DEXPI
P&ID Specification

Version 1.3

Released by the DEXPI Initiative
on June 1, 2021

DEXPI® Data Exchange in the Process Industry

www.dexpi.org

PROCESSNET
EINE INITIATIVE VON DECHEMA UND VDI-GVC

www.processnet.org

License

Contributors

- Yannik Filke (PTC)
- Luis Gomez (AVEVA)
- Leon Hanke (Aucotec)
- Idar Pe Ingebrigtsen (Equinor)
- David Kochmanski (Covestro)
- Marko Luukkainen (Semantum)
- Reiner Meyer-Rössl (Autodesk)
- Felix Schumacher (Siemens)
- Paul Snijder (Hexagon)
- Heiner Temmen (Evonik)
- Manfred Theißen (pnb plants & bytes/AixCAPE)
- Gregor Tolksdorf (Evonik)
- David Vazquez-Landa (BASF)
- Axel Wagner (Covestro)
- Wolfgang Welscher (X-Visual)
- Michael Wiedau (Evonik)

Contents

Part 1

Overview and Concepts

Introduction 1

The DEXPI Initiative (Data EXchange in the Process Industry) is a working party of ProcessNet, which is a joint initiative of DECHEMA and VDI-GVC:

> “ProcessNet is the German platform for chemical engineering with more than 5,000 members. Experts from the sciences, industry and administration exchange ideas and experience, discuss current topics and identify new scientific trends. ProcessNet is a joint initiative of DECHEMA and VDI-GVC.
>
> ProcessNet organises numerous events targeting the interdisciplinary and cross-sectoral exchange of information. The most prominent conference is the ProcessNet Annual Meeting attracting more than 1,000 participants. The wide variety of thematically structured committees deal with scientific and technical problems and issues of paramount technological and societal relevance, they also trigger funding policy initiatives. ProcessNet is the national contact point for international co-operations. Participation in ProcessNet is open to all members of DECHEMA and/or VDI-GVC.”
>
> —Source: https://www.processnet.org

The DEXPI Initiative is hosted by DECHEMA e.V. and SusChem Deutschland.

1.1. Motivation for DEXPI

Due to the lack of interoperability between CAE[1] (and other) systems, companies today face high efforts in data exchange while working together to execute projects for planning, construction and operation of process plants. Parties typically exchanging data in such projects are e.g. EP/EPCs[2], owner-operators, and vendors, but also site services and authorities. One of the main reasons for this high effort is the lack of an agreed understanding across the different systems, e.g. by means of a commonly used standard for data exchange in the process industry. To become more efficient during planning, construction and operation of plants, a data exchange model based on the ISO 15926 standard shall be established.

1.2. Objectives

The objective is to develop and promote a general method for data exchange, data interoperability and data integration for the process industry covering all phases of the lifecycle of a (petro-)chemical plant, ranging from specification of functional requirements to assets in operation. This method shall cover formats and content to address various problems seen today:

- Avoid format conversions (and thereby data loss) when passing engineering data and documents across CAE system boundaries.
- Make handover of engineering data during and at the end of a project easy and cost-effective.
- Reduce data exchange barriers between different CAE systems or different customizations of the same CAE systems.
- Support long-term storage of plant data in a CAE system independent format. Today’s commonly used standard formats like PDF don’t support value added improvements or at best insufficiently.
- Simplify co-existence of different CAE systems within a company, e.g. due to mergers/acquisitions or different priorities in different business units.

[1] CAE: Computer Aided Engineering

[2] EPC: Engineering-Procurement-Construction

1.3. Expectations

EP/EPCs, suppliers and owner operators want to minimize the cost for handling engineering data during planning, construction and operation of process plants between different CAE systems and they want to create opportunities for new value-added functions base on the available engineering data. Therefore the CAE vendors will implement a valid global standard for data exchange into their CAE systems. In a first phase, data exchange will cover graphics, topology of the full P&ID[3] and attributes of the discrete P&ID components.

The involved owner/operator companies from the DEXPI Initiative will define a common data model which is based on the ISO 15926 standard. The resulting data model will be aligned with other projects in the global ISO 15926 community. The CAE vendors will implement this common data model as the basis for data exchange and will deliver it as part of their default system configuration. In addition, it is expected that CAE vendors agree on a common exchange format for the graphical representation of a P&ID and implement the result in their systems as well. The involved companies expect a constructive team work of the CAE vendors during the definition of the common ISO 15926 conformant data model.

Objective of the first phase of the initiative is the transfer of a P&ID from one P&ID system to another P&ID system. The data transfer must include graphics, symbols, topology, all engineering attributes, enumerations, select lists etc. to enable seamless continuation of work on the P&ID in the destination system. Transfer of engineering data over the full life cycle of a plant between different CAE tools, e.g. from simulation to basic/detail engineering up to operations and maintenance may be covered in subsequent phases.

[3] P&ID: Piping and Instrumentation Diagram

Modeling Concepts 2

This section is a technical introduction to the *DEXPI Information Model*.

2.1. Unified Modeling Language (UML)

The *DEXPI Information Model* is a class model in terms of the *Unified Modeling Language (UML)*. Here, we give an informal overview on the UML concepts used in this specification and on their graphical notation.

2.1.1 Types and Instances

A type specifies a set of allowed values known as the instances of the type [UML:7.5.3.1].

The graphical notation of all types used in DEXPI is a rectangle with the type's name in bold face. Depending on the type, the rectangle may contain additional information. An instance is represented by a rectangle with an underlined string that is composed of the instance's name (if any) and the name of the instance's type, separated by a colon. Depending on the type of the instance, additional information may be shown.

In DEXPI, two kinds of types are used: data types and classes.

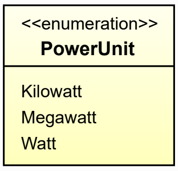
Data types

A data type differs from a class in that instances of a data type are *identified only by their value. All instances of a data type with the same value are considered to be equal instances* [UML:10.2.3.1].

- An *enumeration* is a data type that is specified by a list of its values [UML:10.2.3.3]. These values are called (enumeration) literals. They are identified by a name which must be unique within an enumeration.

  The graphical notation for an enumeration contains contains the keyword `<<enumeration>>` in the name compartment. There is a separate compartment with the names of the enumeration's literals.

Example

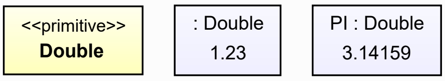


The diagram shows the enumeration *PowerUnit*. The enumeration contains three literals: *Kilowatt*, *Megawatt*, and *Watt*.

- A *primitive type* does not have any substructure (i.e., attributes). Its meaning is defined outside UML [UML:10.2.3.2].

  The graphical notation of a primitive data type contains the keyword `<<primitive>>`. The notation for an instance includes a suitable representation of its value, e.g., a literal.

Example

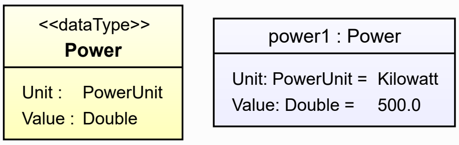


The diagram shows the primitive type *Double*, a type for floating point numbers, and two instances: an anonymous instance with value 1.23 and an instance with name PI and value 3.14159.

- There are further data types in DEXPI that are not primitive types or enumerations. They are either abstract base classes that implement *nullable types*, or they are *structured data types*, i.e., they have attributes.

  The graphical notation for these data types also contains the keyword `<<dataType>>`, and it has a compartment for the data type's attributes, if applicable.

Example

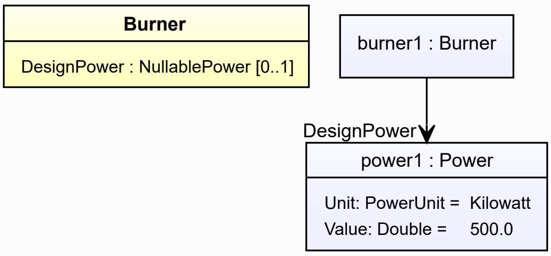


Power is a structured data type with two attributes *Unit* and *Value*. The types of the attributes are *PowerUnit* (an enumeration) and *Double*.
power1 is an instance of *Power*. The *Unit* is *PowerUnit.Kilowatt*, and the *Value* 500.

Classes

The purpose of a class is to specify a classification of objects and to specify [their] features [UML:10.4.1]. As the only features used in this specification are attributes, classes are similar to structured data types. However, the identity of values of classes (which are conventionally called objects) is handled differently: two objects that belong to the same class and that have equal attribute values (e.g, two instances of *Nozzle*, both with the *SubTagName* `N1` and no value for all other attributes of *Nozzle*) are not identical.

Example



Burner is a class with an attribute *DesignPower*. The value of this attribute for the instance burner1 is power1, i.e., a *Power* of 100 kW.

2.1.2 Packages and Models

A package is a namespace for its members [UML:12.2.3.1]. In DEXPI, we use packages as containers for related elements in order to structure the information model. The *DEXPI Information Model* consists of 11 packages. Some of them cover basic data types (e.g., *DataTypes* and *PhysicalQuantities*) while others contain classes to describe various aspects of a P&ID (e.g., *Equipment*, *Piping*, and *Instrumentation*). The package *DexpiModel* provides the class *DexpiModel*, the root of the DEXPI composition-hierarchy.

A *model* is a special kind of package that describes an entire system [UML:12.2.3.11]. Thus, the *DEXPI Information Model* itself is a model in terms of UML. The system that it describes is the engineering and layout information in P&IDs.

Note that the names of all DEXPI data types and classes are unique. In consequence, data types and classes can be identified by their name, and it is not required to give a package name.

Example

There is only one class with name `Tank`. It is not required to give the name of the owning package (`Equipment::Tank`) or even to give a fully qualified name (`Dexpi::Equipment::Tank`) in order to identify the class.

Technical Note

Apart from the case described above, element names in DEXPI are not guaranteed to be unique if the named elements [UML:7.4.3.2] are owned by different name spaces [UML:7.4.3.1]. For example,

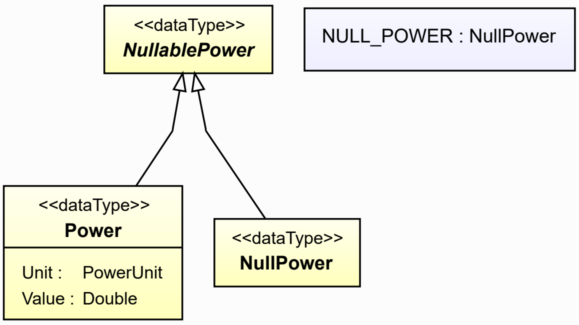
- the package *Equipment* contains the class *Equipment*;
- both the class *Pump* and the class *Compressor* have an attribute called `DesignVolumeFlowRate`;
- all enumerations in the *Enumerations* have an enumeration literal called `NULL`.

2.2. Patterns

2.2.1 Null Values

For several data types, the DEXPI Information Model defines an explicit *null* value.

Example

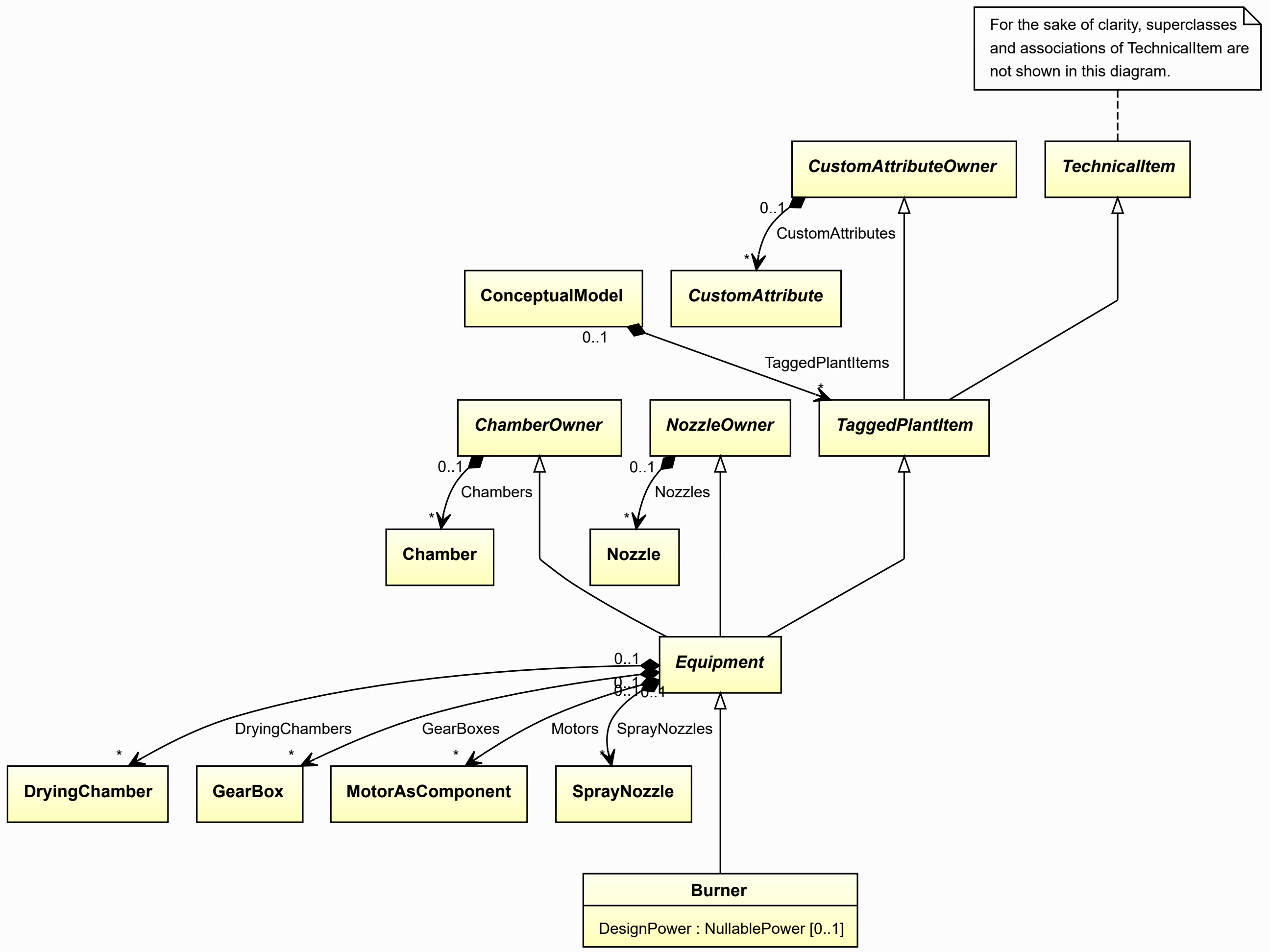


The data type *NullablePower* has two subtypes: *Power* is used to give an *actual* value with a mandatory numerical *Value* and a mandatory *Unit*. *NullPower* is the type of the *null* value *NULL_POWER*.

A *null* value for a data attribute indicates that in a certain information base (e.g., the information available in a P&ID tool) the attribute is known (“declared”), but no value is available.

In contrast, *no* value for a data attribute indicates that in a certain information base, the attribute is not known (“not declared”) - and in consequence, no value is available.

Example



The class *Burner* has a data attribute *DesignPower*. The type of the attribute is *NullablePower*.

- If an exporting tool supports the attribute (e.g., because the tool has been configured accordingly) and a value for the attribute is known (e.g., because the user has entered a value), the DEXPI export should contain the known value.
- If an exporting tool supports the attribute, but no value for the attribute is known (e.g., because the user has never entered a value), the DEXPI export should contain the *null* value.
- If an exporting tool does not support the attribute (e.g., because the tool has not been configured accordingly), the DEXPI export should contain no value.

Implementation in Proteus Schema 3

The exchange format for DEXPI 1.3 is Proteus Schema 4.1 (see https://github.com/ProteusXML). To this end, there is a mapping from each type and attribute in the *DEXPI Information Model* to an XML pattern. Even if there are special cases due to certain design decisions in Proteus Schema, some general guidelines for the DEXPI-Proteus mapping apply.

3.1. Primitive Types

Most primitive types in DEXPI have an equivalent XML data type (see XML Schema Part 2: Datatypes Second Edition). Values of these primitive types are serialized as specified by XML Schema. For example, see *String* or *Double*.

3.2. Classes

Most classes that are used in the *ConceptualModel* are mapped to an XML tag name defined by Proteus Schema and to an RDL reference. Instances of these classes are serialized as an XML element with the appropriate tag name. The RDL reference is given using the `ComponentClass` and `ComponentClassURI` XML attributes.

Technical Note

The mapping is strictly not to tag names, but to the complex XML types defined in Proteus Schema. As there is no ambiguity, we use the tag names to describe the mapping.

Example

In case of the DEXPI class *Pump*, the tag name is `<Equipment>` and the RDL reference is PUMP. Thus, an instance `pump1` is serialized as follows:

```
<Equipment
    ID="pump1"
    ComponentClass="Pump"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS327239" ...>
  ...
</Equipment>
```

3.3. Predefined Data Attributes

Most predefined data attributes in the *ConceptualModel*, i.e., those data attributes that are not custom attributes, are implemented as Proteus `<GenericAttribute>` elements.

`<GenericAttribute>` elements are grouped in a `<GenericAttributes>` element (note the plural-`s`). According to Proteus Schema, an arbitrary number of `<GenericAttributes>` elements can be used as children of several other Proteus elements (e.g., of an `<Equipment>` element). The required `Number` attribute of a `<GenericAttributes>` element gives the number of `<GenericAttribute>` elements. The optional `Set` attribute can be an arbitrary string.

In order to give values for the predefined data attributes, the Set attribute must have the value "DexpiAttributes". For any parent element, there must be at most one <GenericAttributes> child with Set="DexpiAttributes". This <GenericAttributes> element must not contain other content than the predefined data attributes according to this specification.

The DEXPI specification does not forbid other <GenericAttributes> containers. Note that Set="CustomAttributes" is also reserved by DEXPI.

Example

This <Equipment> element for a *Pump* has an <GenericAttributes> element for predefined DEXPI data attributes and another <GenericAttributes> element for arbitrary content.

```
<Equipment
    ID="pump1"
    ComponentClass="Pump"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS327239" ...>
  ...
  <GenericAttributes Number="3" Set="DexpiAttributes">
    <!-- only content according to this specification -->
    <GenericAttribute .../>
    <GenericAttribute .../>
    <GenericAttribute .../>
  </GenericAttributes>
  <GenericAttributes Number="2" Set="SomeOtherContent">
    <!-- arbitrary content -->
    <GenericAttribute .../>
    <GenericAttribute .../>
  </GenericAttributes>
  ...
</Equipment>
```

The XML attributes to be used for each <GenericAttribute> element depend on the type of the data attribute.

3.3.1 Enumerations

For enumeration types, the <GenericAttribute> element must have these XML attributes:

XML Attribute	Description
Name	RDL reference for attribute: name in camel-case
AttributeURI	RDL reference for attribute: URI
Value	RDL reference for enumeration literal: name in camel-case; must be omitted to transfer an enumeration lite
ValueURI	RDL reference for enumeration literal: URI; must be omitted to transfer an enumeration literal that represe
Format	fixed value "anyURI"

Example

Consider the attribute *Location* of *ProcessInstrumentationFunction*. The RDL reference is LOCATION SPECIALIZATION at http://sandbox.dexpi.org/rdl/LocationSpecialization.
Attribute value *Field*; the RDL reference for this literal is FIELD at http://data.posccaesar.org/rdl/RDS409545541.

```
<GenericAttribute
  Name="LocationSpecialization"
  AttributeURI="http://sandbox.dexpi.org/rdl/LocationSpecialization"
  Value="Field"
  ValueURI="http://data.posccaesar.org/rdl/RDS409545541"
  Format="anyURI"/>
```

Attribute value *NULL* (null value of *LocationClassification*).

```
<GenericAttribute
  Name="LocationSpecialization"
  AttributeURI="http://sandbox.dexpi.org/rdl/LocationSpecialization"
  Format="anyURI"/>
```

3.3.2 Integers

For values of type *NullableInteger*, the `<GenericAttribute>` element must have these XML attributes:

XML Attribute	Description
`Name`	RDL reference for attribute: name in camel-case
`AttributeURI`	RDL reference for attribute: URI
`Value`	integer value; must be omitted to transfer the null value *NULL_INTEGER*
`Format`	fixed value `"integer"`

Example

Consider the attribute *NumberOfTubes* of *TubeBundle*. The RDL reference is `NUMBER OF TUBES` at http://data.posccaesar.org/rdl/RDS363959.
Attribute value 36:

```
<GenericAttribute
  Name="NumberOfTubes"
  AttributeURI="http://data.posccaesar.org/rdl/RDS363959"
  Value="36"
  Format="integer"/>
```

Attribute value *NULL_INTEGER*:

```
<GenericAttribute
  Name="NumberOfTubes"
  AttributeURI="http://data.posccaesar.org/rdl/RDS363959"
  Format="integer"/>
```

3.3.3 Multi-Language Strings

For values of type *MultiLanguageString*, a `<GenericAttribute>` element is used for each *SingleLanguageString*. Each `<GenericAttribute>` element must have these attributes:

XML Attribute	Description
`Name`	RDL reference for attribute: name in camel-case
`AttributeURI`	RDL reference for attribute: URI
`Value`	*Value* of the *SingleLanguageString*; must be omitted to transfer the null value *NULL_STRING*
`Format`	fixed value `"string"`
`Language`	*Language* of the *SingleLanguageString*; must be omitted to transfer the null value *NULL_STRING*

For an example, see attribute *ChamberDescription* of *Chamber*.

3.3.4 Physical Quantities

For physical quantities, the `<GenericAttribute>` element must have these XML attributes:

XML Attribute	Description
`Name`	RDL reference for attribute: name in camel-case
`AttributeURI`	RDL reference for attribute: URI
`Value`	numeric value of the physical quantity; must be omitted to transfer a null value
`Units`	RDL reference for unit: name in camel-case; must be omitted to transfer a null value
`UnitsURI`	RDL reference for unit: URI; must be omitted to transfer a null value
`Format`	fixed value `"double"`

Example

Consider the attribute *InsulationThickness* of *PipeFitting*. The RDL reference is `INSULATION THICKNESS` at http://data.posccaesar.org/rdl/RDS4238040.
Attribute value 40 mm: The RDL reference for *Millimetre* is `MILLIMETRE` at http://data.posccaesar.org/rdl/RDS1357739.

```
<GenericAttribute
  Name="InsulationThickness"
  AttributeURI="http://data.posccaesar.org/rdl/RDS4238040"
  Value="40"
  Units="Millimetre"
  UnitsURI="http://data.posccaesar.org/rdl/RDS1357739"
  Format="double"/>
```

Attribute value *NULL_LENGTH*:

```
<GenericAttribute
  Name="InsulationThickness"
  AttributeURI="http://data.posccaesar.org/rdl/RDS4238040"
  Format="double"/>
```

3.3.5 Strings

For values of type *NullableString*, the `<GenericAttribute>` element must have these XML attributes:

XML Attribute	Description
`Name`	RDL reference for attribute: name in camel-case
`AttributeURI`	RDL reference for attribute: URI
`Value`	integer value; must be omitted to transfer the null value *NULL_STRING*
`Format`	fixed value `"string"`

Example

Consider the attribute *LineNumber* of *PipingNetworkSystem*. The RDL reference is `LINE NUMBER ASSIGNMENT CLASS` at http://sandbox.dexpi.org/rdl/LineNumberAssignmentClass.
Attribute value “47121”:

```
<GenericAttribute
  Name="LineNumberAssignmentClass"
  AttributeURI="http://sandbox.dexpi.org/rdl/LineNumberAssignmentClass"
  Value="47121"
  Format="string"/>
```

Attribute value *NULL_STRING*:

```
<GenericAttribute
  Name="LineNumberAssignmentClass"
  AttributeURI="http://sandbox.dexpi.org/rdl/LineNumberAssignmentClass"
  Format="string"/>
```

3.3.6 URIs

For values of type *NullableAnyURI*, the `<GenericAttribute>` element must have these XML attributes:

XML Attribute	Description
`Name`	RDL reference for attribute: name in camel-case
`AttributeURI`	RDL reference for attribute: URI
`Value`	URI value; must be omitted to transfer the null value *NULL_ANY_URI*
`Format`	fixed value `"anyURI"`

Example

Consider the attribute *TypeURI* of *CustomObject*. The RDL reference is `TYPE URI ASSIGNMENT CLASS` at http://sandbox.dexpi.org/rdl/TypeURIAssignmentClass.
Attribute value `http://www.example.org/MicroImpedancePump`:

```
<GenericAttribute
  Name="TypeURIAssignmentClass"
  AttributeURI="http://sandbox.dexpi.org/rdl/TypeURIAssignmentClass"
  Value="http://www.example.org/MicroImpedancePump"
  Format="anyURI"/>
```

Attribute value *NULL_ANY_URI*:

```
<GenericAttribute
  Name="TypeURIAssignmentClass"
  AttributeURI="http://sandbox.dexpi.org/rdl/TypeURIAssignmentClass"
  Format="anyURI"/>
```

3.4. Custom Data Attributes

All custom attributes are implemented using `<GenericAttribute>` elements. The same rules as for predefined attributes apply, with the following modifications:

- Custom attributes are grouped in a `<GenericAttributes>` element with `Set="CustomAttributes"`.
- Each `<GenericAttribute>` element needs a reference to the type of the attribute. To this end, each subclass of *CustomAttribute* is mapped to an RDL reference. This RDL reference is given in the mandatory `Type` (RDL label in camel case) and `TypeURI` attributes of each `<GenericAttribute>` element. For Proteus XML examples, see the subclasses of *CustomAttribute*.

3.5. Reference Attributes

Most reference attributes are implemented using `<Association>` elements.

3.6. Composition Attributes

Most composition attributes correspond to a parent-child relation in the XML hierarchy.

Part 2

DEXPI Information Model

Package DexpiModel 4

4.1. Overview

The *DexpiModel* package contains two classes:

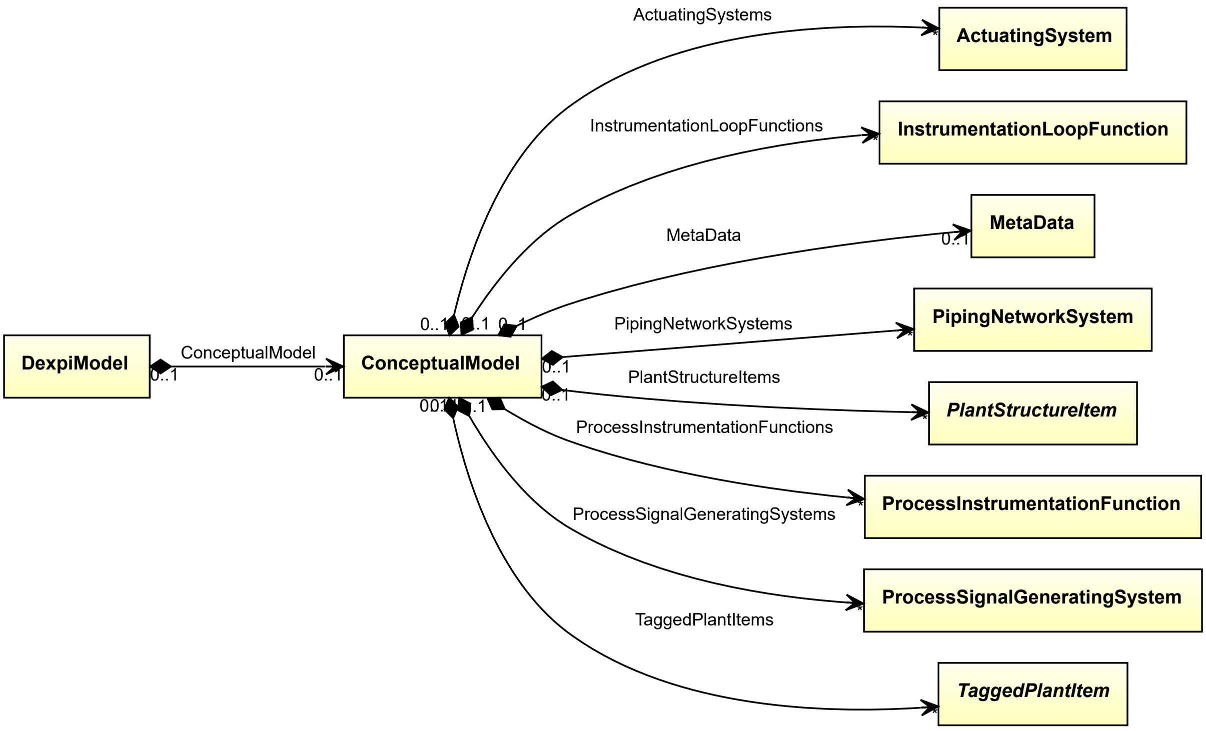
- *DexpiModel*: the root of the composition hierarchy,
- *ConceptualModel*: a container for the conceptual content of a *DexpiModel*, i.e., engineering information independent from its graphical representation.

4.2. ConceptualModel

4.2.1 Overview

Class

The conceptual content of a *DexpiModel*, i.e., engineering information independent from its graphical representation.



Attributes (composition)

Name	Multiplicity	Type
ActuatingSystems	*	*ActuatingSystem*
InstrumentationLoopFunctions	*	*InstrumentationLoopFunction*
MetaData	0..1	*MetaData*
PipingNetworkSystems	*	*PipingNetworkSystem*
PlantStructureItems	*	*PlantStructureItem*
ProcessInstrumentationFunctions	*	*ProcessInstrumentationFunction*
ProcessSignalGeneratingSystems	*	*ProcessSignalGeneratingSystem*
TaggedPlantItems	*	*TaggedPlantItem*

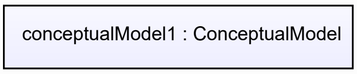
Implementation in Proteus Schema

There is no direct implementation of *ConceptualModel* in Proteus Schema. A *ConceptualModel* is a container for the conceptual information in a *DexpiModel* (as opposed to graphical representation in a *Diagram*), but there is no such distinction in Proteus Schema.
If and only if the top-level `<PlantModel>` element in an XML document contains at least one of these elements,

- `<ActuatingSystem>`
- `<Drawing>`
- `<Equipment>`
- `<InstrumentationLoopFunction>`
- `<MetaData>`
- `<PipingNetworkSystem>`
- `<PlantStructureItem>`
- `<ProcessInstrumentationFunction>`
- `<ProcessSignalGeneratingSystem>`

then the *DexpiModel* corresponding to the `<PlantModel>` contains a *ConceptualModel*.

Example



Example: Implementation in Proteus Schema

The XML fragment contains one of the elements listed above. Hence, *DexpiModel* corresponding to the `<PlantModel>` element contains a *ConceptualModel*, which contains a *PipingNetworkSystem* corresponding to the `<PipingNetworkSystem>` element.

```
<PlantModel>
  ...
  <PipingNetworkSystem ...>
    ...
  </PipingNetworkSystem>
  ...
</PlantModel>
```

4.2.2 ActuatingSystems

Attribute (composition)

The *ActuatingSystems* of the *ConceptualModel*.

Multiplicity: *

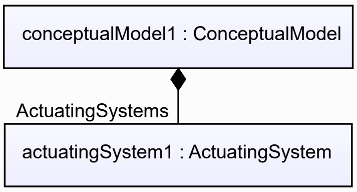
Type: *ActuatingSystem*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

For each *ActuatingSystem*, the corresponding `<ActuatingSystem>` element is a child of the `<PlantModel>` element that corresponds to the *DexpiModel* containing the *ConceptualModel*. See also Proteus Schema Implementation of *ConceptualModel*.

Example


conceptualModel1 : ConceptualModel
ActuatingSystems
actuatingSystem1 : ActuatingSystem

Example: Implementation in Proteus Schema

```
<PlantModel>
  <!--
    The DexpiModel implemented by this PlantModel element implicitly
    contains the ConceptualModel conceptualModel1.
  -->
  ...
  <ActuatingSystem
      ID="actuatingSystem1"
      ComponentClass="ActuatingSystem"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/ActuatingSystem"
      ...>
    ...
  </ActuatingSystem>
  ...
</PlantModel>
```

4.2.3 InstrumentationLoopFunctions

Attribute (composition)

The *InstrumentationLoopFunctions* of the *ConceptualModel*.

Multiplicity: *

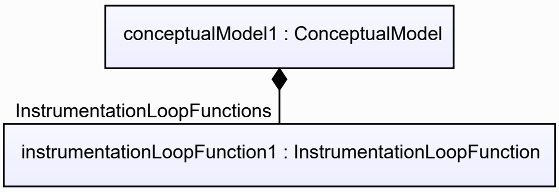
Type: *InstrumentationLoopFunction*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

For each *InstrumentationLoopFunction*, the corresponding `<InstrumentationLoopFunction>` element is a child of the `<PlantModel>` element that corresponds to the *DexpiModel* containing the *ConceptualModel*. See also Proteus Schema Implementation of *ConceptualModel*.

Example


conceptualModel1 : ConceptualModel
InstrumentationLoopFunctions
instrumentationLoopFunction1 : InstrumentationLoopFunction

Example: Implementation in Proteus Schema

```
<PlantModel>
  <!--
    The DexpiModel implemented by this PlantModel element implicitly
    contains the ConceptualModel conceptualModel1.
  -->
  ...
  <InstrumentationLoopFunction
      ID="instrumentationLoopFunction1"
      ComponentClass="InstrumentationLoopFunction"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/InstrumentationLoopFunction"
      ...>
    ...
  </InstrumentationLoopFunction>
  ...
</PlantModel>
```

4.2.4 MetaData

Attribute (composition)

The *MetaData* of the *ConceptualModel*.

Multiplicity: 0..1

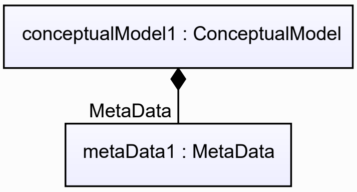
Type: *MetaData*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The `<MetaData>` element corresponding to the *MetaData* is a child of the `<PlantModel>` element that corresponds to the *DexpiModel* containing the *ConceptualModel*. See also Proteus Schema Implementation of *ConceptualModel*.

Example



Example: Implementation in Proteus Schema

```
<PlantModel>
  <!--
    The DexpiModel implemented by this PlantModel element implicitly
    contains the ConceptualModel conceptualModel1.
  -->
  ...
  <MetaData
      ID="metaData1"
      ComponentClass="MetaData"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData"
      ...>
    ...
  </MetaData>
  ...
</PlantModel>
```

4.2.5 PipingNetworkSystems

Attribute (composition)

The *PipingNetworkSystems* of the *ConceptualModel*.

Multiplicity: *

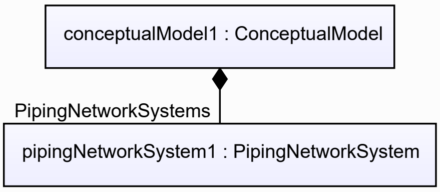
Type: *PipingNetworkSystem*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

For each *PipingNetworkSystem*, the corresponding `<PipingNetworkSystem>` element is a child of the `<PlantModel>` element that corresponds to the *DexpiModel* containing the *ConceptualModel*. See also Proteus Schema Implementation of *ConceptualModel*.

Example


conceptualModel1 : ConceptualModel
PipingNetworkSystems
pipingNetworkSystem1 : PipingNetworkSystem

Example: Implementation in Proteus Schema

```
<PlantModel>
  <!--
    The DexpiModel implemented by this PlantModel element implicitly
    contains the ConceptualModel conceptualModel1.
  -->
  ...
  <PipingNetworkSystem
      ID="pipingNetworkSystem1"
      ComponentClass="PipingNetworkSystem"
      ComponentClassURI="http://data.posccaesar.org/rdl/RDS270359"
      ...>
    ...
  </PipingNetworkSystem>
  ...
</PlantModel>
```

4.2.6 PlantStructureItems

Attribute (composition)

The *PlantStructureItems* of the *ConceptualModel*.

Multiplicity: *

Type: *PlantStructureItem*

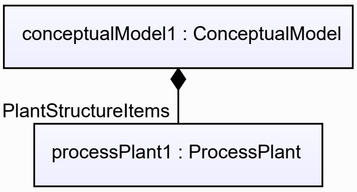
Opposite multiplicity: 0..1

Implementation in Proteus Schema

For each *PlantStructureItem*, the corresponding `<PlantStructureItem>` element is a child of the `<PlantModel>` element that corresponds to the *DexpiModel* containing the *ConceptualModel*. See also Proteus Schema Implementation of *ConceptualModel*.

Example

As the value type *PlantStructureItem* is abstract, we consider *ProcessPlant* as an arbitrary concrete subclass.


conceptualModel1 : ConceptualModel
PlantStructureItems
processPlant1 : ProcessPlant

Example: Implementation in Proteus Schema

```
<PlantModel>
  <!--
    The DexpiModel implemented by this PlantModel element implicitly
    contains the ConceptualModel conceptualModel1.
  -->
  ...
  <PlantStructureItem
      ID="processPlant1"
      ComponentClass="ProcessPlant"
      ComponentClassURI="http://data.posccaesar.org/rdl/RDS7151859"
      ...>
    ...
  </PlantStructureItem>
  ...
</PlantModel>
```

4.2.7 ProcessInstrumentationFunctions

Attribute (composition)

The *ProcessInstrumentationFunctions* of the *ConceptualModel*.

Multiplicity: *

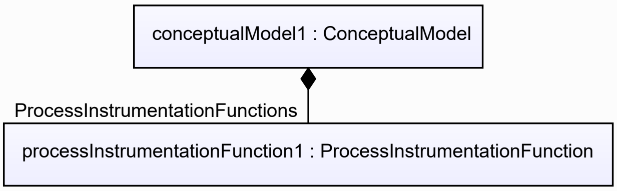
Type: *ProcessInstrumentationFunction*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

For each *ProcessInstrumentationFunction*, the corresponding `<ProcessInstrumentationFunction>` element is a child of the `<PlantModel>` element that corresponds to the *DexpiModel* containing the *ConceptualModel*. See also Proteus Schema Implementation of *ConceptualModel*.

Example


conceptualModel1 : ConceptualModel
ProcessInstrumentationFunctions
processInstrumentationFunction1 : ProcessInstrumentationFunction

Example: Implementation in Proteus Schema

```
<PlantModel>
  <!--
    The DexpiModel implemented by this PlantModel element implicitly
    contains the ConceptualModel conceptualModel1.
  -->
  ...
  <ProcessInstrumentationFunction
      ID="processInstrumentationFunction1"
      ComponentClass="ProcessInstrumentationFunction"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessInstrumentationFunction"
      ...>
    ...
  </ProcessInstrumentationFunction>
  ...
</PlantModel>
```

4.2.8 ProcessSignalGeneratingSystems

Attribute (composition)

The *ProcessSignalGeneratingSystems* of the *ConceptualModel*.

Multiplicity: *

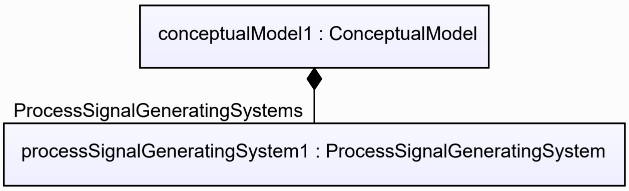
Type: *ProcessSignalGeneratingSystem*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

For each *ProcessSignalGeneratingSystem*, the corresponding `<ProcessSignalGeneratingSystem>` element is a child of the `<PlantModel>` element that corresponds to the *DexpiModel* containing the *ConceptualModel*. See also Proteus Schema Implementation of *ConceptualModel*.

Example


conceptualModel1 : ConceptualModel
ProcessSignalGeneratingSystems
processSignalGeneratingSystem1 : ProcessSignalGeneratingSystem

Example: Implementation in Proteus Schema

```
<PlantModel>
  <!--
    The DexpiModel implemented by this PlantModel element implicitly
    contains the ConceptualModel conceptualModel1.
  -->
  ...
  <ProcessSignalGeneratingSystem
      ID="processSignalGeneratingSystem1"
      ComponentClass="ProcessSignalGeneratingSystem"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessSignalGeneratingSystem"
      ...>
    ...
  </ProcessSignalGeneratingSystem>
  ...
</PlantModel>
```

4.2.9 TaggedPlantItems

Attribute (composition)

The *TaggedPlantItems* of the *ConceptualModel*.

Multiplicity: *

Type: *TaggedPlantItem*

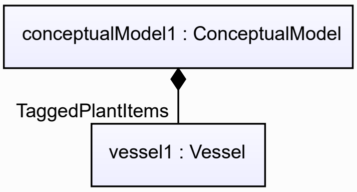
Opposite multiplicity: 0..1

Implementation in Proteus Schema

For each *TaggedPlantItem*, the corresponding `<Equipment>` element is a child of the `<PlantModel>` element that corresponds to the *DexpiModel* containing the *ConceptualModel*. See also Proteus Schema Implementation of *ConceptualModel*.

Example

As the value type *TaggedPlantItem* is abstract, we consider *Vessel* as an arbitrary concrete subclass.



Example: Implementation in Proteus Schema

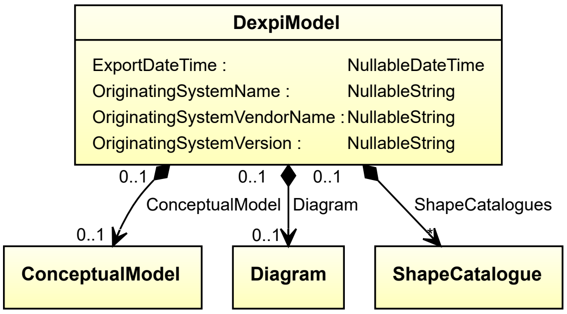
```
<PlantModel>
  <!--
    The DexpiModel implemented by this PlantModel element implicitly
    contains the ConceptualModel conceptualModel1.
  -->
  ...
  <Equipment
      ID="vessel1"
      ComponentClass="Vessel"
      ComponentClassURI="http://data.posccaesar.org/rdl/RDS414674"
      ...>
    ...
  </Equipment>
  ...
</PlantModel>
```

4.3. DexpiModel

4.3.1 Overview

Class

An entire DEXPI model. A *DexpiModel* is the root of the composition hierarchy.


DexpiModel
ExportDateTime : NullableDateTime
OriginatingSystemName : NullableString
OriginatingSystemVendorName : NullableString
OriginatingSystemVersion : NullableString
0..1
ConceptualModel
0..1
0..1
Diagram
0..1
0..1
ShapeCatalogues
ConceptualModel
Diagram
ShapeCatalogue

Attributes (data)

Name	Multiplicity	Type
ExportDateTime	1	*NullableDateTime*
OriginatingSystemName	1	*NullableString*
OriginatingSystemVendorName	1	*NullableString*
OriginatingSystemVersion	1	*NullableString*

Attributes (composition)

Name	Multiplicity	Type
ConceptualModel	0..1	*ConceptualModel*
Diagram	0..1	*Diagram*
ShapeCatalogues	*	*ShapeCatalogue*

Implementation in Proteus Schema

The class is implemented using the Proteus root element `<PlantModel>`, i.e., a *DexpiModel* corresponds to an entire Proteus XML document.
Note that Proteus Schema requires that the `<PlantModel>` element contains a `<PlantInformation>` element. The latter has several required attributes:

- `Application`: fixed value `Dexpi` (optional in Proteus Schema, but required by DEXPI);
- `ApplicationVersion`: fixed value `1.3` (optional in Proteus Schema, but required by DEXPI);
- `Date`: see *ExportDateTime*;
- `Discipline`: fixed value `PID`;
- `Is3D`: fixed value `no`;
- `OriginatingSystem`: see *OriginatingSystemName*;
- `SchemaVersion`: fixed value `4.1`;
- `Time`: see *ExportDateTime*;

The `<PlantInformation>` element must contain a `<UnitsOfMeasure>` element. However, none of the attributes of `<UnitsOfMeasure>` is relevant for DEXPI.

Example

Example: Implementation in Proteus Schema

The XML fragment below only shows the fixed XML attributes of the `<PlantInformation>` element. For the other required XML attributes `Date`, `OriginatingSystem`, `Time`, and `Units`, see the Proteus Schema implementations of the DEXPI attributes given above.

```
<PlantModel>
  <PlantInformation
      Discipline="PID"
      Is3D="no"
      SchemaVersion="4.1"
      ...>
    <UnitsOfMeasure/>
  </PlantInformation>
</PlantModel>
```

4.3.2 ConceptualModel

Attribute (composition)

The conceptual model of the *DexpiModel*.

Multiplicity: 0..1

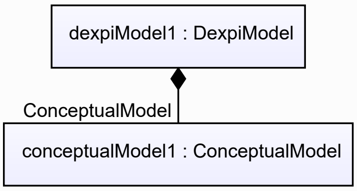
Type: *ConceptualModel*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

See Proteus Schema Implementation of *ConceptualModel*.

Example



4.3.3 Diagram

Attribute (composition)

The diagram of the *DexpiModel*.

Multiplicity: 0..1

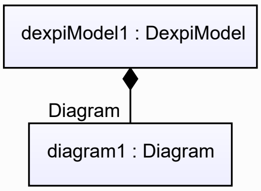
Type: *Diagram*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The `<Drawing>` element that represents the *Diagram* is a child of the `<PlantModel>` element that represents the *DexpiModel*.

Example



Example: Implementation in Proteus Schema

```
<PlantModel>
  <!-- dexpiModel1 -->
  ...
  <Drawing ...>
    <!-- diagram1 -->
    ...
  </Drawing>
  ...
<PlantModel>
```

4.3.4 ExportDateTime

Attribute (data)

The date time at which the *DexpiModel* was exported by the originating system (see *OriginatingSystemName*).

Multiplicity: 1

Type: *NullableDateTime*

Implementation in Proteus Schema

The attribute is implemented using the XML attributes `Date` and `Time` of the `<PlantInformation>` element in the `<PlantModel>` element that corresponds to the *DexpiModel* (see Proteus Schema implementation of *DexpiModel*):

- The date part of the *ExportDateTime* is implemented as the XML attribute `Date` according to the rules for the lexical representation of the XML Schema datatype date as specified by the W3C Recommendation XML Schema Part 2: Datatypes Second Edition from October 28, 2004.
- The time part of the *ExportDateTime* is implemented as the XML attribute `Time` according to the rules for the lexical representation of the XML Schema datatype time as specified by the W3C Recommendation XML Schema Part 2: Datatypes Second Edition from October 28, 2004.

Note that these two attribute are required by Proteus Schema, i.e., it is not possible to transfer the *null value NULL_DATE_TIME*. Also note that the DEXPI type *DateTime* does not allow timezone information. In consequence, the values of the XML attributes `Date` and `Time` must not contain timezone information.

Example

December 7, 2020, 15:32:42 (*DateTime* `"2020-12-07T15:32:42"`)

Example: Implementation in Proteus Schema

The XML fragment below only contains the `Date` and `Time` attributes of the `<PlantInformation>` element. This element has further required attributes (see Proteus Schema implementation of *DexpiModel*).

```
<PlantModel>
  <PlantInformation
      Date="2020-12-07"
      Time="15:32:42"
      ...>
    ...
  </PlantInformation>
</PlantModel>
```

4.3.5 OriginatingSystemName

Attribute (data)

The name of the system from which the *DexpiModel* originates, e.g., the name of a P&ID tool.

Multiplicity: 1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as the XML attribute `OriginatingSystem` of the `<PlantInformation>` element in the `<PlantModel>` element that corresponds to the *DexpiModel* (see Proteus Schema implementation of *DexpiModel*). Note that this attribute is required by Proteus Schema, i.e., it is not possible to transfer the *null value NULL_STRING*.

Example

"PID Kit Professional" (*String*)

Example: Implementation in Proteus Schema

The XML fragment below only contains the `OriginatingSystem` attribute of the `<PlantInformation>` element. This element has further required attributes (see Proteus Schema implementation of *DexpiModel*).

```
<PlantModel>
  <PlantInformation
      OriginatingSystem="PID Kit Professional"
      ...>
    ...
  </PlantInformation>
</PlantModel>
```

4.3.6 OriginatingSystemVendorName

Attribute (data)

The name of the vendor of the system from which the *DexpiModel* originates, e.g., the name of a software company.

Multiplicity: 1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as the XML attribute `OriginatingSystemVendor` of the `<PlantInformation>` element in the `<PlantModel>` element that corresponds to the *DexpiModel* (see Proteus Schema implementation of *DexpiModel*).

Example

"Smart and Clever Systems, Inc." (*String*)

Example: Implementation in Proteus Schema

The XML fragment below only contains the `OriginatingSystemVendor` attribute of the `<PlantInformation>` element. This element has further required attributes (see Proteus Schema implementation of *DexpiModel*).

```
<PlantModel>
  <PlantInformation
      OriginatingSystemVendor="Smart and Clever Systems, Inc."
      ...>
    ...
  </PlantInformation>
</PlantModel>
```

4.3.7 OriginatingSystemVersion

Attribute (data)

The version of the the system from which the *DexpiModel* originates, e.g., the version number of a tool.

Multiplicity: 1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as the XML attribute `OriginatingSystemVersion` of the `<PlantInformation>` element in the `<PlantModel>` element that corresponds to the *DexpiModel* (see Proteus Schema implementation of *DexpiModel*).

Example

“1.1” (*String*)

Example: Implementation in Proteus Schema

The XML fragment below only contains the `OriginatingSystemVersion` attribute of the `<PlantInformation>` element. This element has further required attributes (see Proteus Schema implementation of *DexpiModel*).

```
<PlantModel>
  <PlantInformation
      OriginatingSystemVersion="1.1"
      ...>
    ...
  </PlantInformation>
</PlantModel>
```

4.3.8 ShapeCatalogues

Attribute (composition)

The shape catalogues of the *DexpiModel*.

Multiplicity: *

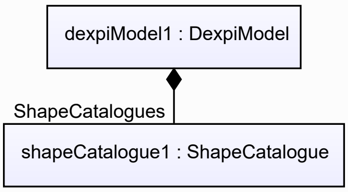
Type: *ShapeCatalogue*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The `<ShapeCatalogue>` element that represents the *ShapeCatalogue* is a child of the `<PlantModel>` element that represents the *DexpiModel*.

Example



Example: Implementation in Proteus Schema

```
<PlantModel>
  <!-- dexpiModel1 -->
  ...
  <ShapeCatalogue ...>
    <!-- shapeCatalogue1 -->
    ...
   </ShapeCatalogue>
  ...
<PlantModel>
```

Package MetaData 5

5.1. Overview

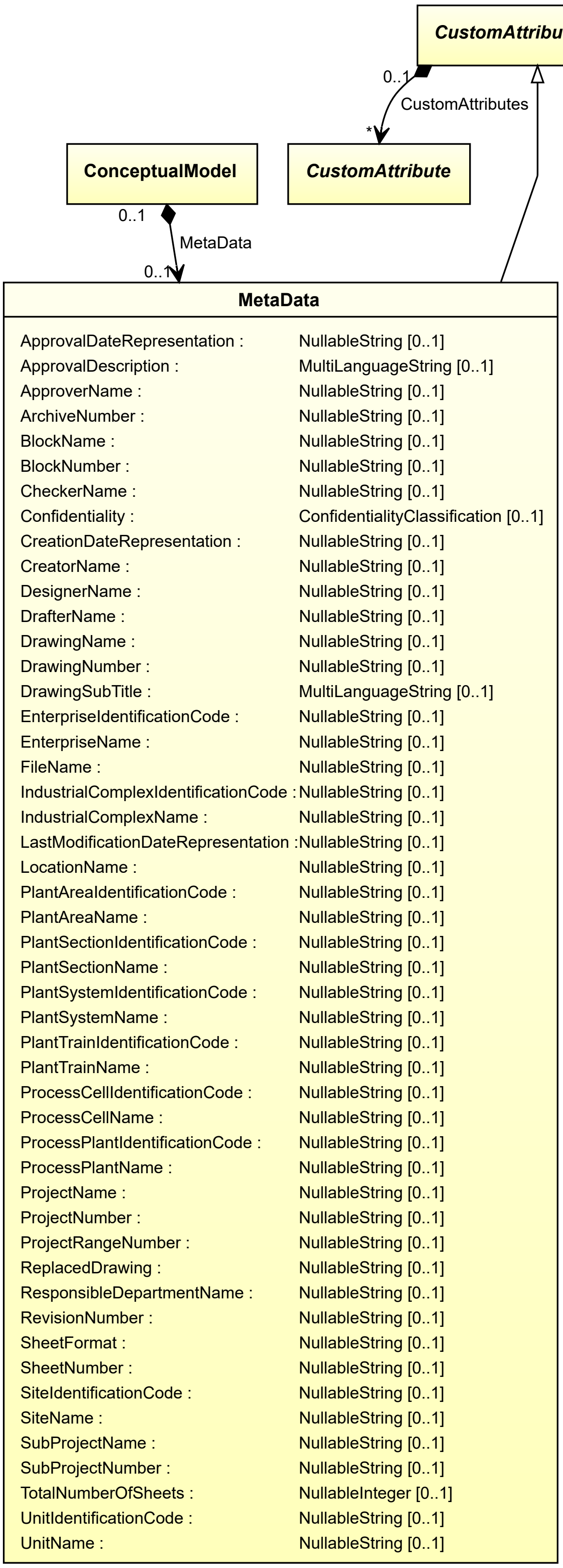
The *MetaData* package provides the *MetaData* class, a container for meta data about a *DexpiModel*.

5.2. MetaData

5.2.1 Overview

Class

A container for meta data about a *DexpiModel*.



ApprovalDateRepresentation :	NullableString [0..1]
ApprovalDescription :	MultiLanguageString [0..1]
ApproverName :	NullableString [0..1]
ArchiveNumber :	NullableString [0..1]
BlockName :	NullableString [0..1]
BlockNumber :	NullableString [0..1]
CheckerName :	NullableString [0..1]
Confidentiality :	ConfidentialityClassification [0..1]
CreationDateRepresentation :	NullableString [0..1]
CreatorName :	NullableString [0..1]
DesignerName :	NullableString [0..1]
DrafterName :	NullableString [0..1]
DrawingName :	NullableString [0..1]
DrawingNumber :	NullableString [0..1]
DrawingSubTitle :	MultiLanguageString [0..1]
EnterpriseIdentificationCode :	NullableString [0..1]
EnterpriseName :	NullableString [0..1]
FileName :	NullableString [0..1]
IndustrialComplexIdentificationCode :	NullableString [0..1]
IndustrialComplexName :	NullableString [0..1]
LastModificationDateRepresentation :	NullableString [0..1]
LocationName :	NullableString [0..1]
PlantAreaIdentificationCode :	NullableString [0..1]
PlantAreaName :	NullableString [0..1]
PlantSectionIdentificationCode :	NullableString [0..1]
PlantSectionName :	NullableString [0..1]
PlantSystemIdentificationCode :	NullableString [0..1]
PlantSystemName :	NullableString [0..1]
PlantTrainIdentificationCode :	NullableString [0..1]
PlantTrainName :	NullableString [0..1]
ProcessCellIdentificationCode :	NullableString [0..1]
ProcessCellName :	NullableString [0..1]
ProcessPlantIdentificationCode :	NullableString [0..1]
ProcessPlantName :	NullableString [0..1]
ProjectName :	NullableString [0..1]
ProjectNumber :	NullableString [0..1]
ProjectRangeNumber :	NullableString [0..1]
ReplacedDrawing :	NullableString [0..1]
ResponsibleDepartmentName :	NullableString [0..1]
RevisionNumber :	NullableString [0..1]
SheetFormat :	NullableString [0..1]
SheetNumber :	NullableString [0..1]
SiteIdentificationCode :	NullableString [0..1]
SiteName :	NullableString [0..1]
SubProjectName :	NullableString [0..1]
SubProjectNumber :	NullableString [0..1]
TotalNumberOfSheets :	NullableInteger [0..1]
UnitIdentificationCode :	NullableString [0..1]
UnitName :	NullableString [0..1]

Supertypes

- *CustomAttributeOwner*

Attributes (data)

Name	Multiplicity	Type
ApprovalDateRepresentation	0..1	*NullableString*
ApprovalDescription	0..1	*MultiLanguageString*
ApproverName	0..1	*NullableString*
ArchiveNumber	0..1	*NullableString*
BlockName	0..1	*NullableString*
BlockNumber	0..1	*NullableString*
CheckerName	0..1	*NullableString*
Confidentiality	0..1	*ConfidentialityClassification*
CreationDateRepresentation	0..1	*NullableString*
CreatorName	0..1	*NullableString*
DesignerName	0..1	*NullableString*
DrafterName	0..1	*NullableString*
DrawingName	0..1	*NullableString*
DrawingNumber	0..1	*NullableString*
DrawingSubTitle	0..1	*MultiLanguageString*
EnterpriseIdentificationCode	0..1	*NullableString*
EnterpriseName	0..1	*NullableString*
FileName	0..1	*NullableString*
IndustrialComplexIdentificationCode	0..1	*NullableString*
IndustrialComplexName	0..1	*NullableString*
LastModificationDateRepresentation	0..1	*NullableString*
LocationName	0..1	*NullableString*
PlantAreaIdentificationCode	0..1	*NullableString*
PlantAreaName	0..1	*NullableString*
PlantSectionIdentificationCode	0..1	*NullableString*
PlantSectionName	0..1	*NullableString*
PlantSystemIdentificationCode	0..1	*NullableString*
PlantSystemName	0..1	*NullableString*
PlantTrainIdentificationCode	0..1	*NullableString*
PlantTrainName	0..1	*NullableString*
ProcessCellIdentificationCode	0..1	*NullableString*
ProcessCellName	0..1	*NullableString*
ProcessPlantIdentificationCode	0..1	*NullableString*
ProcessPlantName	0..1	*NullableString*
ProjectName	0..1	*NullableString*
ProjectNumber	0..1	*NullableString*
ProjectRangeNumber	0..1	*NullableString*

(continued on next page)

Name	Multiplicity	Type
ReplacedDrawing	0..1	*NullableString*
ResponsibleDepartmentName	0..1	*NullableString*
RevisionNumber	0..1	*NullableString*
SheetFormat	0..1	*NullableString*
SheetNumber	0..1	*NullableString*
SiteIdentificationCode	0..1	*NullableString*
SiteName	0..1	*NullableString*
SubProjectName	0..1	*NullableString*
SubProjectNumber	0..1	*NullableString*
TotalNumberOfSheets	0..1	*NullableInteger*
UnitIdentificationCode	0..1	*NullableString*
UnitName	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<MetaData>`
RDL reference: META DATA
ComponentClass: MetaData
ComponentClassURI: http://sandbox.dexpi.org/rdl/MetaData

Example

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
</MetaData>
```

5.2.2 ApprovalDateRepresentation

Attribute (data)

A representation of the approval date of the drawing. The format of the representation is not prescribed.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: APPROVAL DATE REPRESENTATION ASSIGNMENT CLASS
Name: ApprovalDateRepresentationAssignmentClass

AttributeURI: http://sandbox.dexpi.org/rdl/ApprovalDateRepresentationAssignmentClass

Example

"2016-04-01" (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ApprovalDateRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ApprovalDateRepresentationAssignmentClass"
      Format="string"
      Value="2016-04-01" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.3 ApprovalDescription

Attribute (data)

A description of the approval of the drawing.

Multiplicity: 0..1

Type: *MultiLanguageString*

Implementation in Proteus Schema

The attribute is implemented as a *set of DEXPI generic attributes for multi-language string values*.
RDL reference: APPROVAL DESCRIPTION ASSIGNMENT CLASS
Name: ApprovalDescriptionAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ApprovalDescriptionAssignmentClass

Example

Language	Value
en	approved
de	genehmigt

(*MultiLanguageString* with 2 *SingleLanguageString*s)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ApprovalDescriptionAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ApprovalDescriptionAssignmentClass"
      Format="string"
      Language="en"
      Value="approved" />
    <GenericAttribute
      Name="ApprovalDescriptionAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ApprovalDescriptionAssignmentClass"
      Format="string"
      Language="de"
      Value="genehmigt" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.4 ApproverName

Attribute (data)

The name of the approver of the drawing.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: APPROVER NAME ASSIGNMENT CLASS
Name: ApproverNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ApproverNameAssignmentClass

Example

“A. P. Prover” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ApproverNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ApproverNameAssignmentClass"
      Format="string"
      Value="A. P. Prover" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.5 ArchiveNumber

Attribute (data)

The archive number of the drawing.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: ARCHIVE NUMBER ASSIGNMENT CLASS
Name: ArchiveNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ArchiveNumberAssignmentClass

Example

“XY923-463” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ArchiveNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ArchiveNumberAssignmentClass"
      Format="string"
      Value="XY923-463" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.6 BlockName

Attribute (data)

The name of the related block.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: BLOCK NAME ASSIGNMENT CLASS
Name: BlockNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/BlockNameAssignmentClass

Example

"a block" (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="BlockNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/BlockNameAssignmentClass"
      Format="string"
      Value="a block" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.7 BlockNumber

Attribute (data)

The number of the related block.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: BLOCK NUMBER ASSIGNMENT CLASS
Name: BlockNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/BlockNumberAssignmentClass

Example

“B987-654” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="BlockNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/BlockNumberAssignmentClass"
      Format="string"
      Value="B987-654" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.8 CheckerName

Attribute (data)

The name of the checker of the drawing.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: CHECKER NAME ASSIGNMENT CLASS
Name: CheckerNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/CheckerNameAssignmentClass

Example

“C. Hecker” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="CheckerNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/CheckerNameAssignmentClass"
      Format="string"
      Value="C. Hecker" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.9 Confidentiality

Attribute (data)

The confidentiality classification of the drawing.

Multiplicity: 0..1

Type: *ConfidentialityClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: CONFIDENTIALITY SPECIALIZATION
Name: ConfidentialitySpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/ConfidentialitySpecialization

Example

confidential (*ConfidentialityClassification::ConfidentialInformation*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ConfidentialitySpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/ConfidentialitySpecialization"
      Format="anyURI"
      Value="ConfidentialInformation"
      ValueURI="http://data.posccaesar.org/rdl/RDS4316590816" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.10 CreationDateRepresentation

Attribute (data)

A representation of the creation date of the drawing. The format of the representation is not prescribed.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: CREATION DATE REPRESENTATION ASSIGNMENT CLASS
Name: CreationDateRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/CreationDateRepresentationAssignmentClass

Example

“2016-04-01” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="CreationDateRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/CreationDateRepresentationAssignmentClass"
      Format="string"
      Value="2016-04-01" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.11 CreatorName

Attribute (data)

The name of the creator of the drawing.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: CREATOR NAME ASSIGNMENT CLASS
Name: CreatorNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/CreatorNameAssignmentClass

Example

“A. Creator” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="CreatorNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/CreatorNameAssignmentClass"
      Format="string"
      Value="A. Creator" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.12 DesignerName

Attribute (data)

The name of the designer of the drawing.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: DESIGNER NAME ASSIGNMENT CLASS
Name: DesignerNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/DesignerNameAssignmentClass

Example

“D. E. Signer” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignerNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignerNameAssignmentClass"
      Format="string"
      Value="D. E. Signer" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.13 DrafterName

Attribute (data)

The name of the drafter of the drawing.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: DRAFTER NAME ASSIGNMENT CLASS
Name: DrafterNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/DrafterNameAssignmentClass

Example

“D. Rafter” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DrafterNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/DrafterNameAssignmentClass"
      Format="string"
      Value="D. Rafter" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.14 DrawingName

Attribute (data)

The drawing name.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: DRAWING NAME ASSIGNMENT CLASS
Name: DrawingNameAssignmentClass
AttributeURI: http://data.posccaesar.org/rdl/RDS2102503531

Example

“DEXPI example PID” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DrawingNameAssignmentClass"
      AttributeURI="http://data.posccaesar.org/rdl/RDS2102503531"
      Format="string"
      Value="DEXPI example PID" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.15 DrawingNumber

Attribute (data)

The drawing number.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: DRAWING NUMBER ASSIGNMENT CLASS
Name: DrawingNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/DrawingNumberAssignmentClass

Example

“123/A93” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DrawingNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/DrawingNumberAssignmentClass"
      Format="string"
      Value="123/A93" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.16 DrawingSubTitle

Attribute (data)

The subtitle of the drawing.

Multiplicity: 0..1

Type: *MultiLanguageString*

Implementation in Proteus Schema

The attribute is implemented as a *set of DEXPI generic attributes for multi-language string values*.
RDL reference: DRAWING SUB TITLE ASSIGNMENT CLASS
Name: DrawingSubTitleAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/DrawingSubTitleAssignmentClass

Example

Language	Value
en	DEXPI Example PID
de	DEXPI Beispiel-R&I

(*MultiLanguageString* with 2 *SingleLanguageString*s)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DrawingSubTitleAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/DrawingSubTitleAssignmentClass"
      Format="string"
      Language="en"
      Value="DEXPI Example PID" />
    <GenericAttribute
      Name="DrawingSubTitleAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/DrawingSubTitleAssignmentClass"
      Format="string"
      Language="de"
      Value="DEXPI Beispiel-R&amp;I" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.17 EnterpriseIdentificationCode

Attribute (data)

The identification code of the enterprise.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: ENTERPRISE IDENTIFICATION CODE ASSIGNMENT CLASS
Name: EnterpriseIdentificationCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/EnterpriseIdentificationCodeAssignmentClass

Example

"C1248" (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="EnterpriseIdentificationCodeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/EnterpriseIdentificationCodeAssignmentClass"
      Format="string"
      Value="C1248" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.18 EnterpriseName

Attribute (data)

The name of the enterprise.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: ENTERPRISE NAME ASSIGNMENT CLASS
Name: EnterpriseNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/EnterpriseNameAssignmentClass

Example

“CompAny Ltd.” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="EnterpriseNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/EnterpriseNameAssignmentClass"
      Format="string"
      Value="CompAny Ltd." />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.19 FileName

Attribute (data)

The file name of the drawing.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: FILE NAME ASSIGNMENT CLASS
Name: FileNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/FileNameAssignmentClass

Example

“DEXPI_example_PID.xml.” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="FileNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/FileNameAssignmentClass"
      Format="string"
      Value="DEXPI_example_PID.xml." />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.20 IndustrialComplexIdentificationCode

Attribute (data)

The identification code of the industrial complex.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: INDUSTRIAL COMPLEX IDENTIFICATION CODE ASSIGNMENT CLASS
Name: IndustrialComplexIdentificationCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/IndustrialComplexIdentificationCodeAssignmentClass

Example

“I-Chain” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="IndustrialComplexIdentificationCodeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/IndustrialComplexIdentificationCodeAssignmentClass"
      Format="string"
      Value="I-Chain" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.21 IndustrialComplexName

Attribute (data)

The name of the industrial complex.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: INDUSTRIAL COMPLEX NAME ASSIGNMENT CLASS
Name: IndustrialComplexNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/IndustrialComplexNameAssignmentClass

Example

“Isophorone Chain” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="IndustrialComplexNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/IndustrialComplexNameAssignmentClass"
      Format="string"
      Value="Isophorone Chain" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.22 LastModificationDateRepresentation

Attribute (data)

A representation of the last modification date of the drawing. The format of the representation is not prescribed.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: LAST MODIFICATION DATE REPRESENTATION ASSIGNMENT CLASS
Name: LastModificationDateRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/LastModificationDateRepresentationAssignmentClass

Example

“2016-04-02” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="LastModificationDateRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/LastModificationDateRepresentationAssignmentClass"
      Format="string"
      Value="2016-04-02" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.23 LocationName

Attribute (data)

The location name.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: LOCATION NAME ASSIGNMENT CLASS
Name: LocationNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/LocationNameAssignmentClass

Example

“C1248.” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="LocationNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/LocationNameAssignmentClass"
      Format="string"
      Value="C1248." />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.24 PlantAreaIdentificationCode

Attribute (data)

The identification code of the plant area according to ISA-95..

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PLANT AREA IDENTIFICATION CODE ASSIGNMENT CLASS
Name: PlantAreaIdentificationCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PlantAreaIdentificationCodeAssignmentClass

Example

“F4” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PlantAreaIdentificationCodeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PlantAreaIdentificationCodeAssignmentClass"
      Format="string"
      Value="F4" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.25 PlantAreaName

Attribute (data)

The name of the plant area according to ISA-95.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: AREA ISA95 NAME ASSIGNMENT CLASS
Name: AreaIsa95NameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/AreaIsa95NameAssignmentClass

Example

"Area F4" (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="AreaIsa95NameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/AreaIsa95NameAssignmentClass"
      Format="string"
      Value="Area F4" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.26 PlantSectionIdentificationCode

Attribute (data)

The identification code of the plant section.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PLANT SECTION IDENTIFICATION CODE ASSIGNMENT CLASS
Name: PlantSectionIdentificationCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PlantSectionIdentificationCodeAssignmentClass

Example

“10” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PlantSectionIdentificationCodeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PlantSectionIdentificationCodeAssignmentClass"
      Format="string"
      Value="10" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.27 PlantSectionName

Attribute (data)

The name of the plant section.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PLANT SECTION NAME ASSIGNMENT CLASS
Name: PlantSectionNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PlantSectionNameAssignmentClass

Example

“Utilities” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PlantSectionNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PlantSectionNameAssignmentClass"
      Format="string"
      Value="Utilities" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.28 PlantSystemIdentificationCode

Attribute (data)

The identification code of the plant system.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PLANT SYSTEM IDENTIFICATION CODE ASSIGNMENT CLASS
Name: PlantSystemIdentificationCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PlantSystemIdentificationCodeAssignmentClass

Example

“X123” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PlantSystemIdentificationCodeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PlantSystemIdentificationCodeAssignmentClass"
      Format="string"
      Value="X123" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.29 PlantSystemName

Attribute (data)

The name of the plant system.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PLANT SYSTEM NAME ASSIGNMENT CLASS
Name: PlantSystemNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PlantSystemNameAssignmentClass

Example

“System X123” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PlantSystemNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PlantSystemNameAssignmentClass"
      Format="string"
      Value="System X123" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.30 PlantTrainIdentificationCode

Attribute (data)

The identification code of the plant train.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PLANT TRAIN IDENTIFICATION CODE ASSIGNMENT CLASS
Name: PlantTrainIdentificationCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PlantTrainIdentificationCodeAssignmentClass

Example

“T456” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PlantTrainIdentificationCodeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PlantTrainIdentificationCodeAssignmentClass"
      Format="string"
      Value="T456" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.31 PlantTrainName

Attribute (data)

The name of the plant train.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PLANT TRAIN NAME ASSIGNMENT CLASS
Name: PlantTrainNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PlantTrainNameAssignmentClass

Example

“Train T456” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PlantTrainNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PlantTrainNameAssignmentClass"
      Format="string"
      Value="Train T456" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.32 ProcessCellIdentificationCode

Attribute (data)

The identification code of the related process cell according to ISA-95.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PROCESS CELL IDENTIFICATION CODE ASSIGNMENT CLASS
Name: ProcessCellIdentificationCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ProcessCellIdentificationCodeAssignmentClass

Example

"PC123" (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ProcessCellIdentificationCodeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ProcessCellIdentificationCodeAssignmentClass"
      Format="string"
      Value="PC123" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.33 ProcessCellName

Attribute (data)

The name of the related process cell according to ISA-95.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PROCESS CELL NAME ASSIGNMENT CLASS
Name: ProcessCellNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ProcessCellNameAssignmentClass

Example

"a process cell" (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ProcessCellNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ProcessCellNameAssignmentClass"
      Format="string"
      Value="a process cell" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.34 ProcessPlantIdentificationCode

Attribute (data)

The identification code of the process plant.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PROCESS PLANT IDENTIFICATION CODE ASSIGNMENT CLASS
Name: ProcessPlantIdentificationCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ProcessPlantIdentificationCodeAssignmentClass

Example

“ABC” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ProcessPlantIdentificationCodeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ProcessPlantIdentificationCodeAssignmentClass"
      Format="string"
      Value="ABC" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.35 ProcessPlantName

Attribute (data)

The name of the process plant.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PROCESS PLANT NAME ASSIGNMENT CLASS
Name: ProcessPlantNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ProcessPlantNameAssignmentClass

Example

“ABC Plant” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ProcessPlantNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ProcessPlantNameAssignmentClass"
      Format="string"
      Value="ABC Plant" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.36 ProjectName

Attribute (data)

The name of the related project.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PROJECT NAME ASSIGNMENT CLASS
Name: ProjectNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ProjectNameAssignmentClass

Example

“a project” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ProjectNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ProjectNameAssignmentClass"
      Format="string"
      Value="a project" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.37 ProjectNumber

Attribute (data)

The number of the related project.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PROJECT NUMBER ASSIGNMENT CLASS
Name: ProjectNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ProjectNumberAssignmentClass

Example

“P3.1415” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ProjectNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ProjectNumberAssignmentClass"
      Format="string"
      Value="P3.1415" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.38 ProjectRangeNumber

Attribute (data)

The range number of the related project.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PROJECT RANGE NUMBER ASSIGNMENT CLASS
Name: ProjectRangeNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ProjectRangeNumberAssignmentClass

Example

“PR321” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ProjectRangeNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ProjectRangeNumberAssignmentClass"
      Format="string"
      Value="PR321" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.39 ReplacedDrawing

Attribute (data)

The drawing replaced by this drawing.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: REPLACED DRAWING ASSIGNMENT CLASS
Name: ReplacedDrawingAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ReplacedDrawingAssignmentClass

Example

“D321” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ReplacedDrawingAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ReplacedDrawingAssignmentClass"
      Format="string"
      Value="D321" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.40 ResponsibleDepartmentName

Attribute (data)

The name of the department responsible for the drawing.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: RESPONSIBLE DEPARTMENT NAME ASSIGNMENT CLASS
Name: ResponsibleDepartmentNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ResponsibleDepartmentNameAssignmentClass

Example

“R2-D2” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ResponsibleDepartmentNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ResponsibleDepartmentNameAssignmentClass"
      Format="string"
      Value="R2-D2" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.41 RevisionNumber

Attribute (data)

The revision number of the drawing.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: REVISION NUMBER ASSIGNMENT CLASS
Name: RevisionNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/RevisionNumberAssignmentClass

Example

“R2.2” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="RevisionNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/RevisionNumberAssignmentClass"
      Format="string"
      Value="R2.2" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.42 SheetFormat

Attribute (data)

The sheet format.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: SHEET FORMAT ASSIGNMENT CLASS
Name: SheetFormatAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/SheetFormatAssignmentClass

Example

“DIN A3” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="SheetFormatAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/SheetFormatAssignmentClass"
      Format="string"
      Value="DIN A3" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.43 SheetNumber

Attribute (data)

The sheet number of the drawing.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: SHEET NUMBER ASSIGNMENT CLASS
Name: SheetNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/SheetNumberAssignmentClass

Example

“2a” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="SheetNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/SheetNumberAssignmentClass"
      Format="string"
      Value="2a" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.44 SiteIdentificationCode

Attribute (data)

The identification code of the site.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: SITE IDENTIFICATION CODE ASSIGNMENT CLASS
Name: SiteIdentificationCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/SiteIdentificationCodeAssignmentClass

Example

“DC” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="SiteIdentificationCodeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/SiteIdentificationCodeAssignmentClass"
      Format="string"
      Value="DC" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.45 SiteName

Attribute (data)

The name of the site.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: SITE NAME ASSIGNMENT CLASS
Name: SiteNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/SiteNameAssignmentClass

Example

“Dexpi City” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="SiteNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/SiteNameAssignmentClass"
      Format="string"
      Value="Dexpi City" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.46 SubProjectName

Attribute (data)

The name of the related sub-project.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: SUB PROJECT NAME ASSIGNMENT CLASS
Name: SubProjectNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/SubProjectNameAssignmentClass

Example

“a sub-project” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="SubProjectNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/SubProjectNameAssignmentClass"
      Format="string"
      Value="a sub-project" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.47 SubProjectNumber

Attribute (data)

The number of the related sub-project.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: SUB PROJECT NUMBER ASSIGNMENT CLASS
Name: SubProjectNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/SubProjectNumberAssignmentClass

Example

“P3.1415-SP2” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="SubProjectNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/SubProjectNumberAssignmentClass"
      Format="string"
      Value="P3.1415-SP2" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.48 TotalNumberOfSheets

Attribute (data)

The total number of sheets.

Multiplicity: 0..1

Type: *NullableInteger*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for integer values*.
RDL reference: TOTAL NUMBER OF SHEETS
Name: TotalNumberOfSheets
AttributeURI: http://sandbox.dexpi.org/rdl/TotalNumberOfSheets

Example

4 (*Integer*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="TotalNumberOfSheets"
      AttributeURI="http://sandbox.dexpi.org/rdl/TotalNumberOfSheets"
      Format="integer"
      Value="4" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.49 UnitIdentificationCode

Attribute (data)

The identification code of the related unit according to ISA-95.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: UNIT IDENTIFICATION CODE ASSIGNMENT CLASS
Name: UnitIdentificationCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/UnitIdentificationCodeAssignmentClass

Example

“U-923-463” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="UnitIdentificationCodeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/UnitIdentificationCodeAssignmentClass"
      Format="string"
      Value="U-923-463" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

5.2.50 UnitName

Attribute (data)

The name of the related unit according to ISA-95.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: UNIT ISA95 NAME ASSIGNMENT CLASS
Name: UnitIsa95NameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/UnitIsa95NameAssignmentClass

Example

“a unit” (*String*)

Example: Implementation in Proteus Schema

```
<MetaData
    ID="metaData1"
    ComponentClass="MetaData"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MetaData" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="UnitIsa95NameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/UnitIsa95NameAssignmentClass"
      Format="string"
      Value="a unit" />
    ...
  </GenericAttributes>
  ...
</MetaData>
```

Package PlantStructure 6

6.1. Overview

The *PlantStructure* package provides classes to assign *TechnicalItems* (e.g., *TaggedPlantItems* or *PipingNetworkSystems*) to a hierarchy of structures, i.e., technical or organizational groups.

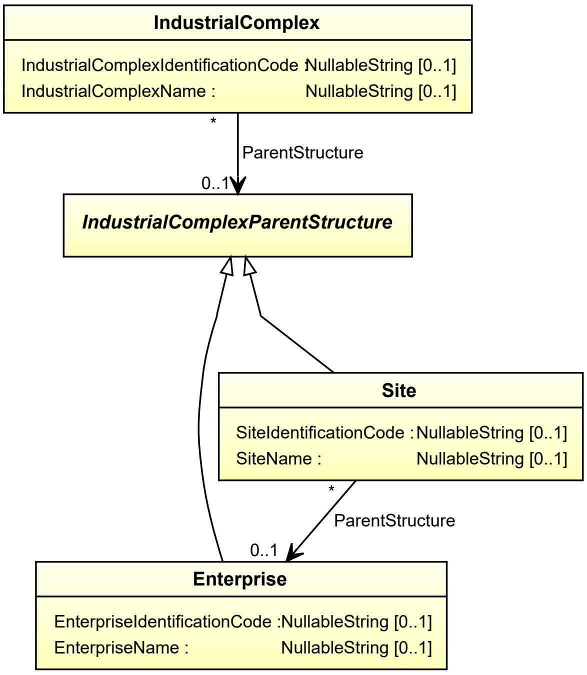
6.1.1 Main Hierarchy

The main hierarchy consists of 6 classes:

- *Enterprise*
- *Site*
- *IndustrialComplex*
- *ProcessPlant*
- *PlantSection*
- *TechnicalItem* (abstract superclass of *TaggedPlantItem*, *PipingNetworkSystem*, etc.)

For each object of one of these classes (except *Enterprise*), a parent structure can be given. A parent structure must be an instance of a class that is on a higher level in the hierarchy. For example, only an *Enterprise* is a suitable parent structure of a *Site*. The parent structure of an *IndustrialComplex* can be an *Enterprise* or a *Site*.

In order to capture these restrictions in the information model, some abstract auxiliary classes are used. For example, *IndustrialComplexParentStructure* is an abstract base class of both *Enterprise* and *Site*. It is used as the type of the *ParentStructure* attribute of *IndustrialComplex*:


IndustrialComplex
IndustrialComplexIdentificationCode :NullableString [0..1]
IndustrialComplexName : NullableString [0..1]
*
ParentStructure
0..1
IndustrialComplexParentStructure
Site
SiteIdentificationCode : NullableString [0..1]
SiteName : NullableString [0..1]
*
ParentStructure
0..1
Enterprise
EnterpriseIdentificationCode :NullableString [0..1]
EnterpriseName : NullableString [0..1]

6.1.2 Side Hierarchies

In addition to the main hierarchy, there are 3 side hierarchies:

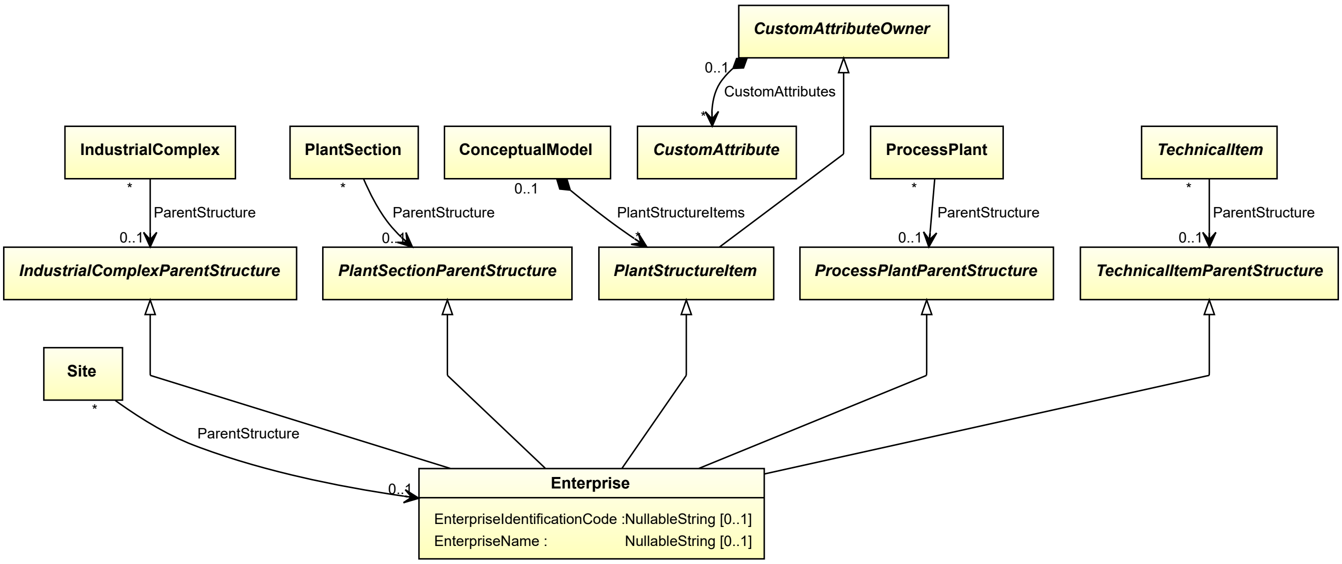
- Instances of *IndustrialComplex*, *ProcessPlant*, *PlantSection*, and *TechnicalItem* can be assigned to an *PlantArea*, see *PlantAreaLocatedStructure*.
- *TechnicalItems* can be assigned to a *PlantSystem*, see *PlantSystemLocatedStructure*.
- *TechnicalItems* can be assigned to a *PlantTrain*, see *PlantTrainLocatedStructure*.

6.2. Enterprise

6.2.1 Overview

Class

An enterprise as defined by ISA 95.



Supertypes

- *IndustrialComplexParentStructure*
- *PlantSectionParentStructure*
- *PlantStructureItem*
- *ProcessPlantParentStructure*
- *TechnicalItemParentStructure*

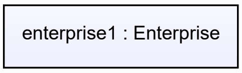
Attributes (data)

Name	Multiplicity	Type
EnterpriseIdentificationCode	0..1	*NullableString*
EnterpriseName	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PlantStructureItem>`
RDL reference: ISA95 ENTERPRISE
ComponentClass: Isa95Enterprise
ComponentClassURI: http://data.posccaesar.org/rdl/RDS10418236543

Example



Example: Implementation in Proteus Schema

```
<PlantStructureItem
    ID="enterprise1"
    ComponentClass="Isa95Enterprise"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS10418236543" ...>
  ...
</PlantStructureItem>
```

6.2.2 EnterpriseIdentificationCode

Attribute (data)

The identification code of the enterprise.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: ENTERPRISE IDENTIFICATION CODE ASSIGNMENT CLASS
Name: EnterpriseIdentificationCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/EnterpriseIdentificationCodeAssignmentClass

Example

“C1248” (*String*)

Example: Implementation in Proteus Schema

```
<PlantStructureItem
    ID="enterprise1"
    ComponentClass="Isa95Enterprise"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS10418236543" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="EnterpriseIdentificationCodeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/EnterpriseIdentificationCodeAssignmentClass"
      Format="string"
      Value="C1248" />
    ...
  </GenericAttributes>
  ...
</PlantStructureItem>
```

6.2.3 EnterpriseName

Attribute (data)

The name of the enterprise.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: ENTERPRISE NAME ASSIGNMENT CLASS
Name: EnterpriseNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/EnterpriseNameAssignmentClass

Example

“CompAny Ltd.” (*String*)

Example: Implementation in Proteus Schema

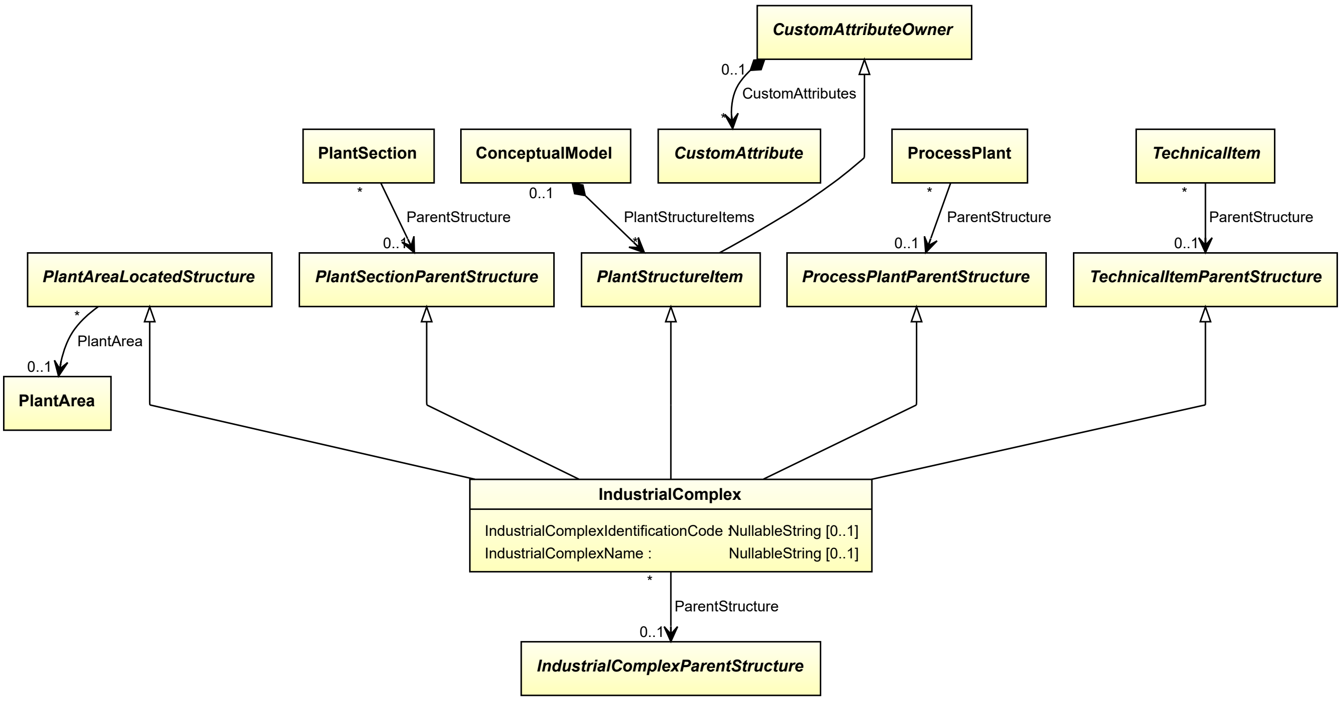
```
<PlantStructureItem
    ID="enterprise1"
    ComponentClass="Isa95Enterprise"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS10418236543" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="EnterpriseNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/EnterpriseNameAssignmentClass"
      Format="string"
      Value="CompAny Ltd." />
    ...
  </GenericAttributes>
  ...
</PlantStructureItem>
```

6.3. IndustrialComplex

6.3.1 Overview

Class

An industrial complex as defined by ISO 10209:2012.



Supertypes

- *PlantAreaLocatedStructure*
- *PlantSectionParentStructure*
- *PlantStructureItem*
- *ProcessPlantParentStructure*
- *TechnicalItemParentStructure*

Attributes (data)

Name	Multiplicity	Type
IndustrialComplexIdentificationCode	0..1	*NullableString*
IndustrialComplexName	0..1	*NullableString*

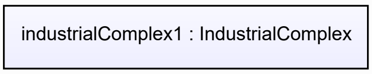
Attributes (reference)

Name	Multiplicity	Type
ParentStructure	0..1	*IndustrialComplexParentStructure*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PlantStructureItem>`
RDL reference: INDUSTRIAL COMPLEX ISO10209 2012
ComponentClass: IndustrialComplexIso102092012
ComponentClassURI: http://sandbox.dexpi.org/rdl/IndustrialComplexIso102092012

Example



Example: Implementation in Proteus Schema

```
<PlantStructureItem
    ID="industrialComplex1"
    ComponentClass="IndustrialComplexIso102092012"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/IndustrialComplexIso102092012" ...>
  ...
</PlantStructureItem>
```

6.3.2 IndustrialComplexIdentificationCode

Attribute (data)

The identification code of the industrial complex.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: INDUSTRIAL COMPLEX IDENTIFICATION CODE ASSIGNMENT CLASS
Name: IndustrialComplexIdentificationCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/IndustrialComplexIdentificationCodeAssignmentClass

Example

“I-Chain” (*String*)

Example: Implementation in Proteus Schema

```
<PlantStructureItem
    ID="industrialComplex1"
    ComponentClass="IndustrialComplexIso102092012"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/IndustrialComplexIso102092012" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="IndustrialComplexIdentificationCodeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/IndustrialComplexIdentificationCodeAssignmentClass"
      Format="string"
      Value="I-Chain" />
    ...
  </GenericAttributes>
  ...
</PlantStructureItem>
```

6.3.3 IndustrialComplexName

Attribute (data)

The name of the industrial complex.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: INDUSTRIAL COMPLEX NAME ASSIGNMENT CLASS
Name: IndustrialComplexNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/IndustrialComplexNameAssignmentClass

Example

“Isophorone Chain” (*String*)

Example: Implementation in Proteus Schema

```
<PlantStructureItem
    ID="industrialComplex1"
    ComponentClass="IndustrialComplexIso102092012"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/IndustrialComplexIso102092012" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="IndustrialComplexNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/IndustrialComplexNameAssignmentClass"
      Format="string"
      Value="Isophorone Chain" />
    ...
  </GenericAttributes>
  ...
</PlantStructureItem>
```

6.3.4 ParentStructure

Attribute (reference)

A superordinate structure of which the *IndustrialComplex* is a part.

Multiplicity: 0..1

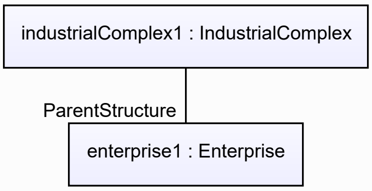
Type: *IndustrialComplexParentStructure*

Opposite multiplicity: 0..*

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: `"is a part of"`
Opposite association type: `"is a collection including"`

Example



Example: Implementation in Proteus Schema

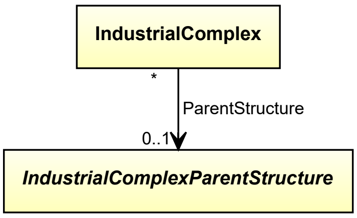
```
<PlantStructureItem
    ID="industrialComplex1"
    ComponentClass="IndustrialComplexIso102092012"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/IndustrialComplexIso102092012" ...>
  ...
  <Association
    Type="is a part of"
    ItemID="enterprise1" />
  ...
<PlantStructureItem />
...
<PlantStructureItem
    ID="enterprise1"
    ComponentClass="Isa95Enterprise"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS10418236543" ...>
  ...
  <Association
    Type="is a collection including"
    ItemID="industrialComplex1" />
  ...
<PlantStructureItem />
```

6.4. IndustrialComplexParentStructure

6.4.1 Overview

Abstract class

A *PlantStructureItem* that is a suitable *ParentStructure* of an *IndustrialComplex*.



Subtypes

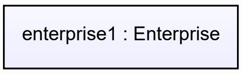
- *Enterprise*
- *Site*

Implementation in Proteus Schema

Implementation is subclass-specific.

Example

As *IndustrialComplexParentStructure* is abstract, we consider *Enterprise* as an arbitrary concrete subclass.



Example: Implementation in Proteus Schema

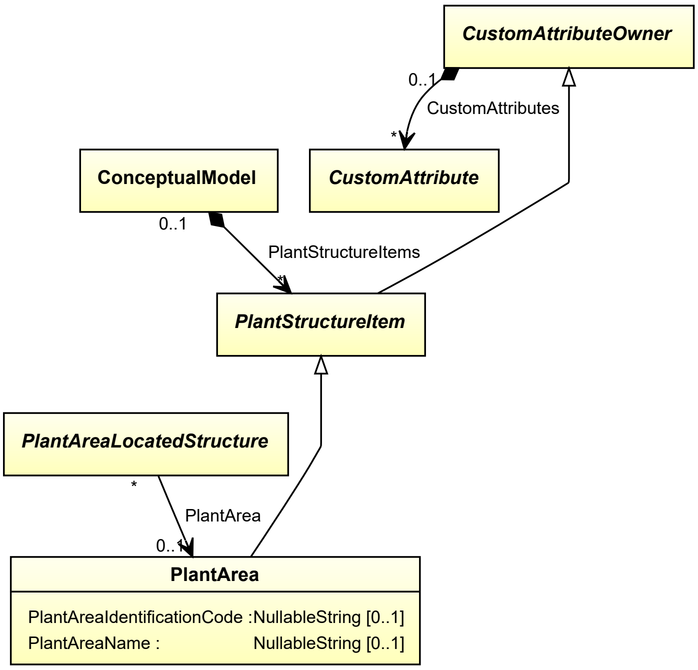
```
<PlantStructureItem
    ID="enterprise1"
    ComponentClass="Isa95Enterprise"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS10418236543" ...>
  ...
</PlantStructureItem>
```

6.5. PlantArea

6.5.1 Overview

Class

An area as defined by ISA 95. The name PlantArea has been chosen to avoid confusion with the data type *Area*.



Supertypes

- *PlantStructureItem*

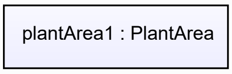
Attributes (data)

Name	Multiplicity	Type
PlantAreaIdentificationCode	0..1	*NullableString*
PlantAreaName	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PlantStructureItem>`
RDL reference: AREA ISA95
ComponentClass: AreaIsa95
ComponentClassURI: http://data.posccaesar.org/rdl/RDS10418236534

Example



Example: Implementation in Proteus Schema

```
<PlantStructureItem
    ID="plantArea1"
    ComponentClass="AreaIsa95"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS10418236534" ...>
  ...
</PlantStructureItem>
```

6.5.2 PlantAreaIdentificationCode

Attribute (data)

The identification code of the plant area according to ISA-95..

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PLANT AREA IDENTIFICATION CODE ASSIGNMENT CLASS
Name: PlantAreaIdentificationCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PlantAreaIdentificationCodeAssignmentClass

Example

"F4" (*String*)

Example: Implementation in Proteus Schema

```
<PlantStructureItem
    ID="plantArea1"
    ComponentClass="AreaIsa95"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS10418236534" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PlantAreaIdentificationCodeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PlantAreaIdentificationCodeAssignmentClass"
      Format="string"
      Value="F4" />
    ...
  </GenericAttributes>
  ...
</PlantStructureItem>
```

6.5.3 PlantAreaName

Attribute (data)

The name of the plant area according to ISA-95.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: AREA ISA95 NAME ASSIGNMENT CLASS
Name: AreaIsa95NameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/AreaIsa95NameAssignmentClass

Example

“Area F4” (*String*)

Example: Implementation in Proteus Schema

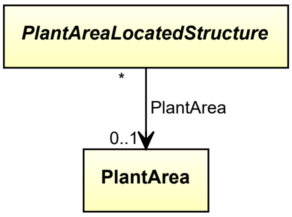
```
<PlantStructureItem
    ID="plantArea1"
    ComponentClass="AreaIsa95"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS10418236534" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="AreaIsa95NameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/AreaIsa95NameAssignmentClass"
      Format="string"
      Value="Area F4" />
    ...
  </GenericAttributes>
  ...
</PlantStructureItem>
```

6.6. PlantAreaLocatedStructure

6.6.1 Overview

Abstract class

A structure that can be located in an *PlantArea*.



Subtypes

- *IndustrialComplex*
- *PlantSection*
- *ProcessPlant*
- *TechnicalItem*

Attributes (reference)

Name	Multiplicity	Type
PlantArea	0..1	*PlantArea*

Implementation in Proteus Schema

Implementation is subclass-specific.

Example

As *PlantAreaLocatedStructure* is abstract, we consider *ProcessPlant* as an arbitrary concrete subclass.

processPlant1 : ProcessPlant

Example: Implementation in Proteus Schema

```
<PlantStructureItem
    ID="processPlant1"
    ComponentClass="ProcessPlant"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS7151859" ...>
  ...
</PlantStructureItem>
```

6.6.2 PlantArea

Attribute (reference)

The *PlantArea* in which the *PlantAreaLocatedStructure* is located.

Multiplicity: 0..1

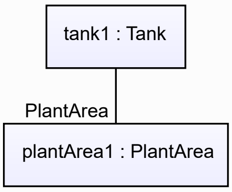
Type: *PlantArea*

Opposite multiplicity: 0..*

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: `"is located in"`
Opposite association type: `"is the location of"`

Example



Example: Implementation in Proteus Schema

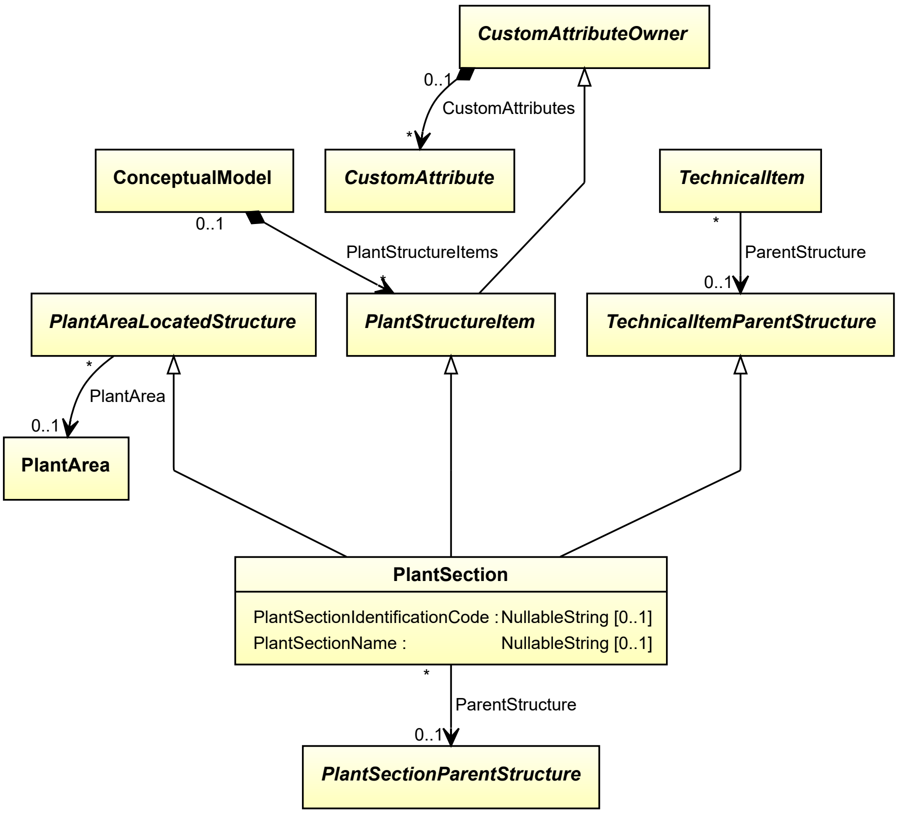
```
<Equipment
    ID="tank1"
    ComponentClass="Tank"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS445139" ...>
  ...
  <Association
    Type="is located in"
    ItemID="plantArea1" />
  ...
<Equipment />
...
<PlantStructureItem
    ID="plantArea1"
    ComponentClass="AreaIsa95"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS10418236534" ...>
  ...
  <Association
    Type="is the location of"
    ItemID="tank1" />
  ...
<PlantStructureItem />
```

6.7. PlantSection

6.7.1 Overview

Class

A plant section as defined by ISO 10209:2012.



Supertypes

- *PlantAreaLocatedStructure*
- *PlantStructureItem*
- *TechnicalItemParentStructure*

Attributes (data)

Name	Multiplicity	Type
PlantSectionIdentificationCode	0..1	*NullableString*
PlantSectionName	0..1	*NullableString*

Attributes (reference)

Name	Multiplicity	Type
ParentStructure	0..1	*PlantSectionParentStructure*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PlantStructureItem>`
RDL reference: PLANT SECTION ISO10209 2012
ComponentClass: PlantSectionIso102092012
ComponentClassURI: http://sandbox.dexpi.org/rdl/PlantSectionIso102092012

Example

Example: Implementation in Proteus Schema

```
<PlantStructureItem
    ID="plantSection1"
    ComponentClass="PlantSectionIso102092012"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PlantSectionIso102092012" ...>
  ...
</PlantStructureItem>
```

6.7.2 ParentStructure

Attribute (reference)

A superordinate structure of which the *PlantSection* is a part.

Multiplicity: 0..1

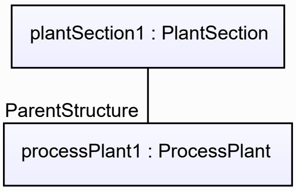
Type: *PlantSectionParentStructure*

Opposite multiplicity: 0..*

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: `"is a part of"`
Opposite association type: `"is a collection including"`

Example



Example: Implementation in Proteus Schema

```
<PlantStructureItem
    ID="plantSection1"
    ComponentClass="PlantSectionIso102092012"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PlantSectionIso102092012" ...>
  ...
  <Association
    Type="is a part of"
    ItemID="processPlant1" />
  ...
<PlantStructureItem />
...
<PlantStructureItem
    ID="processPlant1"
    ComponentClass="ProcessPlant"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS7151859" ...>
  ...
  <Association
    Type="is a collection including"
    ItemID="plantSection1" />
  ...
<PlantStructureItem />
```

6.7.3 PlantSectionIdentificationCode

Attribute (data)

The identification code of the plant section.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PLANT SECTION IDENTIFICATION CODE ASSIGNMENT CLASS
Name: PlantSectionIdentificationCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PlantSectionIdentificationCodeAssignmentClass

Example

“10” (*String*)

Example: Implementation in Proteus Schema

```
<PlantStructureItem
    ID="plantSection1"
    ComponentClass="PlantSectionIso102092012"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PlantSectionIso102092012" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PlantSectionIdentificationCodeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PlantSectionIdentificationCodeAssignmentClass"
      Format="string"
      Value="10" />
    ...
  </GenericAttributes>
  ...
</PlantStructureItem>
```

6.7.4 PlantSectionName

Attribute (data)

The name of the plant section.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PLANT SECTION NAME ASSIGNMENT CLASS
Name: PlantSectionNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PlantSectionNameAssignmentClass

Example

“Utilities” (*String*)

Example: Implementation in Proteus Schema

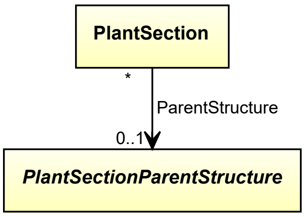
```
<PlantStructureItem
    ID="plantSection1"
    ComponentClass="PlantSectionIso102092012"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PlantSectionIso102092012" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PlantSectionNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PlantSectionNameAssignmentClass"
      Format="string"
      Value="Utilities" />
    ...
  </GenericAttributes>
  ...
</PlantStructureItem>
```

6.8. PlantSectionParentStructure

6.8.1 Overview

Abstract class

A *PlantStructureItem* that is a suitable *ParentStructure* of a *PlantSection*.



Subtypes

- *Enterprise*
- *IndustrialComplex*
- *ProcessPlant*
- *Site*

Implementation in Proteus Schema

Implementation is subclass-specific.

Example

As *PlantSectionParentStructure* is abstract, we consider *ProcessPlant* as an arbitrary concrete subclass.

processPlant1 : ProcessPlant

Example: Implementation in Proteus Schema

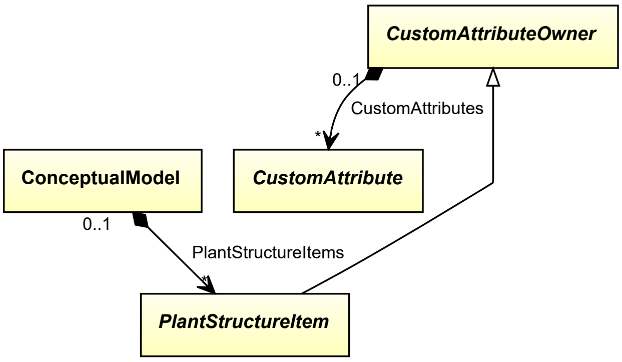
```
<PlantStructureItem
    ID="processPlant1"
    ComponentClass="ProcessPlant"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS7151859" ...>
  ...
</PlantStructureItem>
```

6.9. PlantStructureItem

6.9.1 Overview

Abstract class

Item of the plant break down structure.



Supertypes

- *CustomAttributeOwner*

Subtypes

- *Enterprise*
- *IndustrialComplex*
- *PlantArea*
- *PlantSection*
- *PlantSystem*
- *PlantTrain*
- *ProcessPlant*
- *Site*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*. As *PlantStructureItem* is abstract, there is no RDL reference for the class itself; the RDL reference depends on the concrete subclass.
Tag: `<PlantStructureItem>`
ComponentClass: *depending on subclass*

ComponentClassURI: *depending on subclass*

Example

As *PlantStructureItem* is abstract, we consider *ProcessPlant* as an arbitrary concrete subclass.

processPlant1 : ProcessPlant

Example: Implementation in Proteus Schema

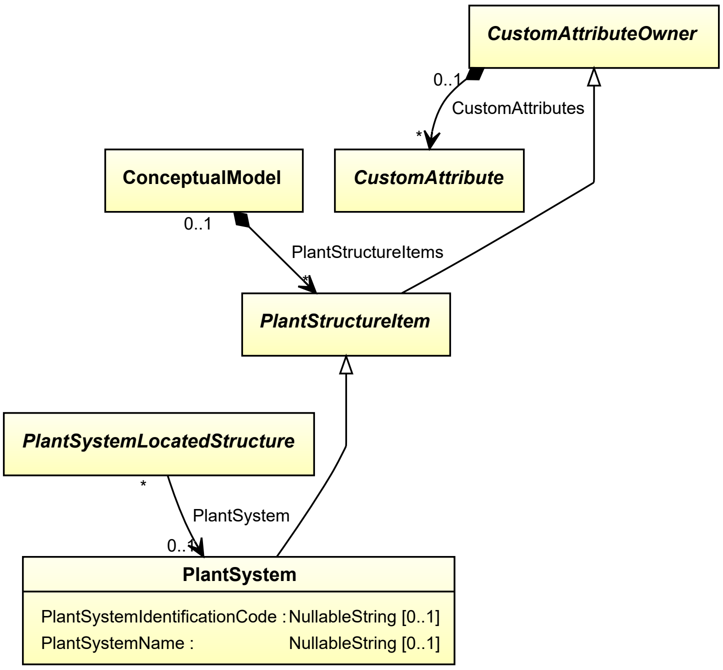
```
<PlantStructureItem
    ID="processPlant1"
    ComponentClass="ProcessPlant"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS7151859" ...>
  ...
</PlantStructureItem>
```

6.10. PlantSystem

6.10.1 Overview

Class

A plant system.



Supertypes

- *PlantStructureItem*

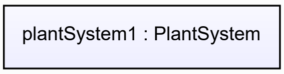
Attributes (data)

Name	Multiplicity	Type
PlantSystemIdentificationCode	0..1	*NullableString*
PlantSystemName	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PlantStructureItem>`
RDL reference: PLANT SYSTEM
ComponentClass: PlantSystem
ComponentClassURI: http://sandbox.dexpi.org/rdl/PlantSystem

Example



Example: Implementation in Proteus Schema

```
<PlantStructureItem
    ID="plantSystem1"
    ComponentClass="PlantSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PlantSystem" ...>
  ...
</PlantStructureItem>
```

6.10.2 PlantSystemIdentificationCode

Attribute (data)

The identification code of the plant system.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PLANT SYSTEM IDENTIFICATION CODE ASSIGNMENT CLASS
Name: PlantSystemIdentificationCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PlantSystemIdentificationCodeAssignmentClass

Example

“X123” (*String*)

Example: Implementation in Proteus Schema

```
<PlantStructureItem
    ID="plantSystem1"
    ComponentClass="PlantSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PlantSystem" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PlantSystemIdentificationCodeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PlantSystemIdentificationCodeAssignmentClass"
      Format="string"
      Value="X123" />
    ...
  </GenericAttributes>
  ...
</PlantStructureItem>
```

6.10.3 PlantSystemName

Attribute (data)

The name of the plant system.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PLANT SYSTEM NAME ASSIGNMENT CLASS
Name: PlantSystemNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PlantSystemNameAssignmentClass

Example

“System X123” (*String*)

Example: Implementation in Proteus Schema

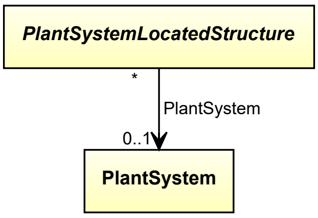
```
<PlantStructureItem
    ID="plantSystem1"
    ComponentClass="PlantSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PlantSystem" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PlantSystemNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PlantSystemNameAssignmentClass"
      Format="string"
      Value="System X123" />
    ...
  </GenericAttributes>
  ...
</PlantStructureItem>
```

6.11. PlantSystemLocatedStructure

6.11.1 Overview

Abstract class

A structure that can be located in a *PlantSystem*.



Subtypes

- *TechnicalItem*

Attributes (reference)

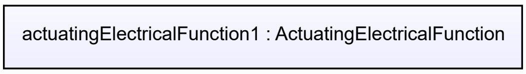
Name	Multiplicity	Type
PlantSystem	0..1	*PlantSystem*

Implementation in Proteus Schema

Implementation is subclass-specific.

Example

As *PlantSystemLocatedStructure* is abstract, we consider *ActuatingElectricalFunction* as an arbitrary concrete subclass.



Example: Implementation in Proteus Schema

```
<ActuatingElectricalFunction
    ID="actuatingElectricalFunction1"
    ComponentClass="ActuatingElectricalFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ActuatingElectricalFunction" ...>
  ...
</ActuatingElectricalFunction>
```

6.11.2 PlantSystem

Attribute (reference)

The *PlantSystem* in which the *PlantSystemLocatedStructure* is located.

Multiplicity: 0..1

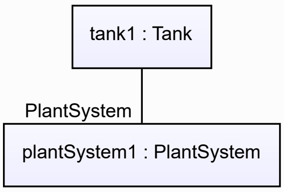
Type: *PlantSystem*

Opposite multiplicity: 0..*

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: `"is located in"`
Opposite association type: `"is the location of"`

Example



Example: Implementation in Proteus Schema

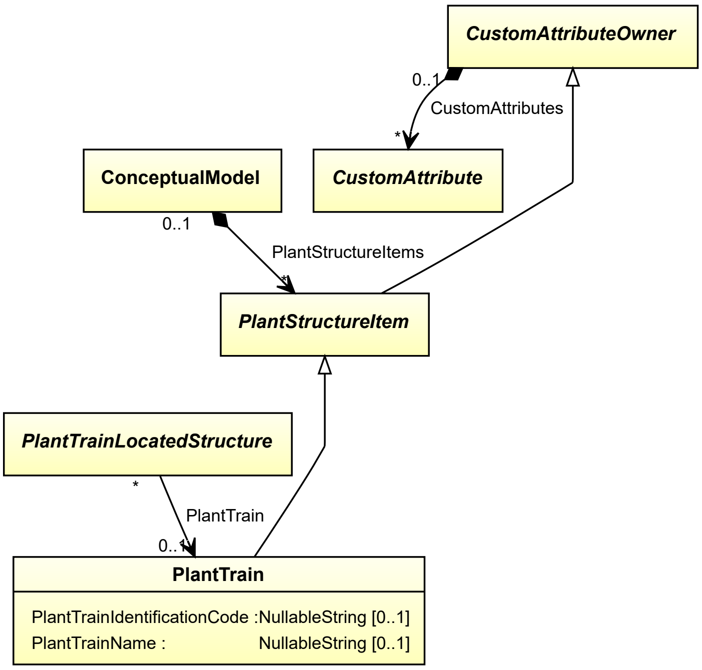
```
<Equipment
    ID="tank1"
    ComponentClass="Tank"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS445139" ...>
  ...
  <Association
    Type="is located in"
    ItemID="plantSystem1" />
  ...
<Equipment />
...
<PlantStructureItem
    ID="plantSystem1"
    ComponentClass="PlantSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PlantSystem" ...>
  ...
  <Association
    Type="is the location of"
    ItemID="tank1" />
  ...
<PlantStructureItem />
```

6.12. PlantTrain

6.12.1 Overview

Class

A plant train.



Supertypes

- *PlantStructureItem*

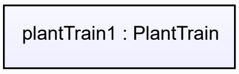
Attributes (data)

Name	Multiplicity	Type
PlantTrainIdentificationCode	0..1	*NullableString*
PlantTrainName	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PlantStructureItem>`
RDL reference: PLANT TRAIN
ComponentClass: PlantTrain
ComponentClassURI: http://sandbox.dexpi.org/rdl/PlantTrain

Example



Example: Implementation in Proteus Schema

```
<PlantStructureItem
    ID="plantTrain1"
    ComponentClass="PlantTrain"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PlantTrain" ...>
  ...
</PlantStructureItem>
```

6.12.2 PlantTrainIdentificationCode

Attribute (data)

The identification code of the plant train.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PLANT TRAIN IDENTIFICATION CODE ASSIGNMENT CLASS
Name: PlantTrainIdentificationCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PlantTrainIdentificationCodeAssignmentClass

Example

“T456” (*String*)

Example: Implementation in Proteus Schema

```
<PlantStructureItem
    ID="plantTrain1"
    ComponentClass="PlantTrain"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PlantTrain" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PlantTrainIdentificationCodeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PlantTrainIdentificationCodeAssignmentClass"
      Format="string"
      Value="T456" />
    ...
  </GenericAttributes>
  ...
</PlantStructureItem>
```

6.12.3 PlantTrainName

Attribute (data)

The name of the plant train.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PLANT TRAIN NAME ASSIGNMENT CLASS
Name: PlantTrainNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PlantTrainNameAssignmentClass

Example

“Train T456” (*String*)

Example: Implementation in Proteus Schema

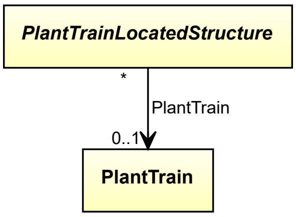
```
<PlantStructureItem
    ID="plantTrain1"
    ComponentClass="PlantTrain"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PlantTrain" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PlantTrainNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PlantTrainNameAssignmentClass"
      Format="string"
      Value="Train T456" />
    ...
  </GenericAttributes>
  ...
</PlantStructureItem>
```

6.13. PlantTrainLocatedStructure

6.13.1 Overview

Abstract class

A structure that can be located in a *PlantTrain*.



Subtypes

- *TechnicalItem*

Attributes (reference)

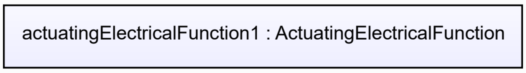
Name	Multiplicity	Type
PlantTrain	0..1	*PlantTrain*

Implementation in Proteus Schema

Implementation is subclass-specific.

Example

As *PlantTrainLocatedStructure* is abstract, we consider *ActuatingElectricalFunction* as an arbitrary concrete subclass.



Example: Implementation in Proteus Schema

```
<ActuatingElectricalFunction
    ID="actuatingElectricalFunction1"
    ComponentClass="ActuatingElectricalFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ActuatingElectricalFunction" ...>
  ...
</ActuatingElectricalFunction>
```

6.13.2 PlantTrain

Attribute (reference)

The *PlantTrain* in which the *PlantTrainLocatedStructure* is located.

Multiplicity: 0..1

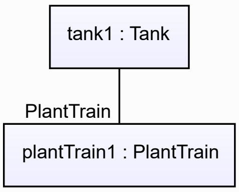
Type: *PlantTrain*

Opposite multiplicity: 0..*

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: `"is located in"`
Opposite association type: `"is the location of"`

Example



Example: Implementation in Proteus Schema

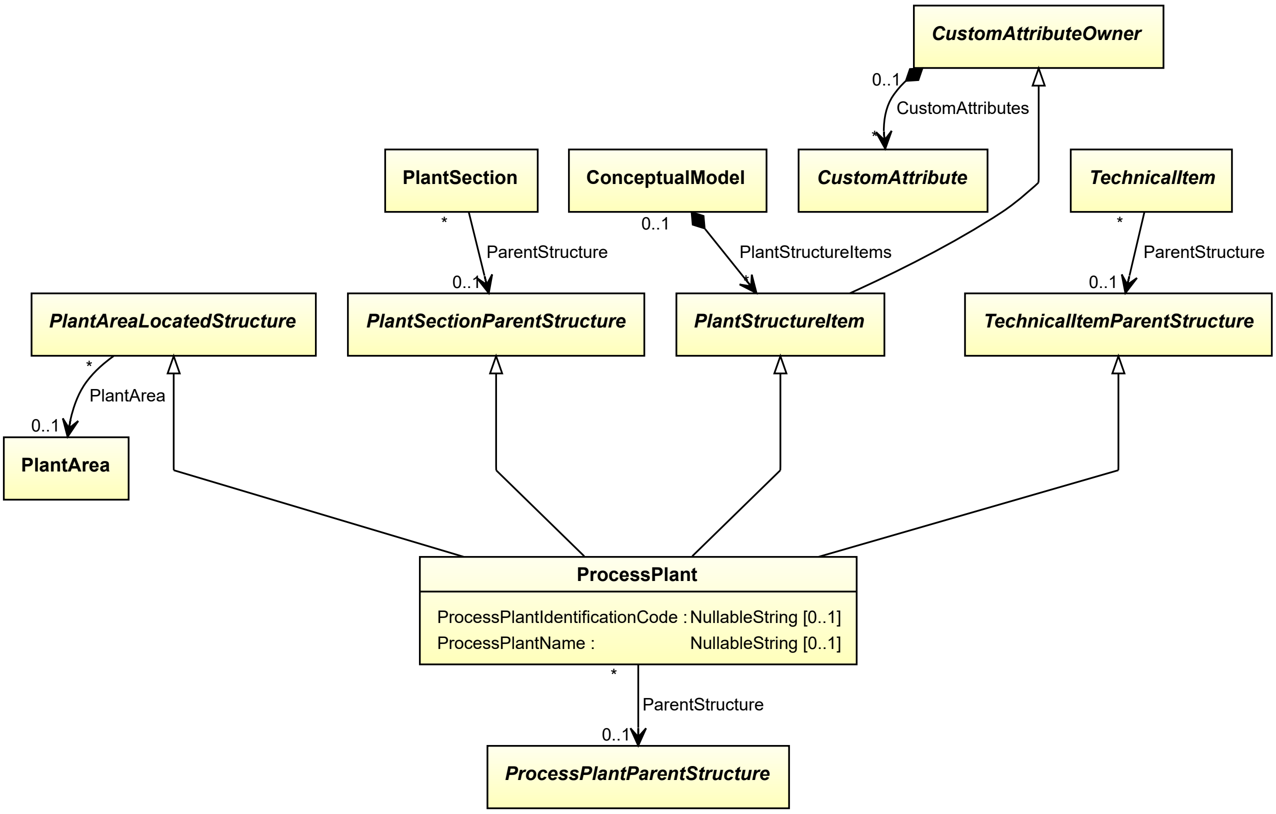
```
<Equipment
    ID="tank1"
    ComponentClass="Tank"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS445139" ...>
  ...
  <Association
    Type="is located in"
    ItemID="plantTrain1" />
  ...
<Equipment />
...
<PlantStructureItem
    ID="plantTrain1"
    ComponentClass="PlantTrain"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PlantTrain" ...>
  ...
  <Association
    Type="is the location of"
    ItemID="tank1" />
  ...
<PlantStructureItem />
```

6.14. ProcessPlant

6.14.1 Overview

Class

A plant employed in carrying out chemical processes, including the required supporting processes (from http://data.posccaesar.org/rdl/RDS7151859).



Supertypes

- *PlantAreaLocatedStructure*
- *PlantSectionParentStructure*
- *PlantStructureItem*
- *TechnicalItemParentStructure*

Attributes (data)

Name	Multiplicity	Type
ProcessPlantIdentificationCode	0..1	*NullableString*
ProcessPlantName	0..1	*NullableString*

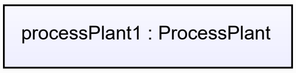
Attributes (reference)

Name	Multiplicity	Type
ParentStructure	0..1	*ProcessPlantParentStructure*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PlantStructureItem>`
RDL reference: PROCESS PLANT
ComponentClass: ProcessPlant
ComponentClassURI: http://data.posccaesar.org/rdl/RDS7151859

Example



Example: Implementation in Proteus Schema

```
<PlantStructureItem
    ID="processPlant1"
    ComponentClass="ProcessPlant"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS7151859" ...>
  ...
</PlantStructureItem>
```

6.14.2 ParentStructure

Attribute (reference)

A superordinate structure of which the *ProcessPlant* is a part.

Multiplicity: 0..1

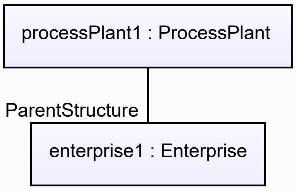
Type: *ProcessPlantParentStructure*

Opposite multiplicity: 0..*

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: `"is a part of"`
Opposite association type: `"is a collection including"`

Example



Example: Implementation in Proteus Schema

```
<PlantStructureItem
    ID="processPlant1"
    ComponentClass="ProcessPlant"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS7151859" ...>
  ...
  <Association
    Type="is a part of"
    ItemID="enterprise1" />
  ...
<PlantStructureItem />
...
<PlantStructureItem
    ID="enterprise1"
    ComponentClass="Isa95Enterprise"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS10418236543" ...>
  ...
  <Association
    Type="is a collection including"
    ItemID="processPlant1" />
  ...
<PlantStructureItem />
```

6.14.3 ProcessPlantIdentificationCode

Attribute (data)

The identification code of the process plant.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PROCESS PLANT IDENTIFICATION CODE ASSIGNMENT CLASS
Name: ProcessPlantIdentificationCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ProcessPlantIdentificationCodeAssignmentClass

Example

"ABC" (*String*)

Example: Implementation in Proteus Schema

```
<PlantStructureItem
    ID="processPlant1"
    ComponentClass="ProcessPlant"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS7151859" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ProcessPlantIdentificationCodeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ProcessPlantIdentificationCodeAssignmentClass"
      Format="string"
      Value="ABC" />
    ...
  </GenericAttributes>
  ...
</PlantStructureItem>
```

6.14.4 ProcessPlantName

Attribute (data)

The name of the process plant.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PROCESS PLANT NAME ASSIGNMENT CLASS
Name: ProcessPlantNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ProcessPlantNameAssignmentClass

Example

“ABC Plant” (*String*)

Example: Implementation in Proteus Schema

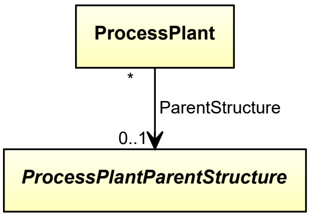
```
<PlantStructureItem
    ID="processPlant1"
    ComponentClass="ProcessPlant"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS7151859" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ProcessPlantNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ProcessPlantNameAssignmentClass"
      Format="string"
      Value="ABC Plant" />
    ...
  </GenericAttributes>
  ...
</PlantStructureItem>
```

6.15. ProcessPlantParentStructure

6.15.1 Overview

Abstract class

A *PlantStructureItem* that is a suitable *ParentStructure* of a *ProcessPlant*.



Subtypes

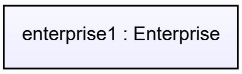
- *Enterprise*
- *IndustrialComplex*
- *Site*

Implementation in Proteus Schema

Implementation is subclass-specific.

Example

As *ProcessPlantParentStructure* is abstract, we consider *Enterprise* as an arbitrary concrete subclass.



Example: Implementation in Proteus Schema

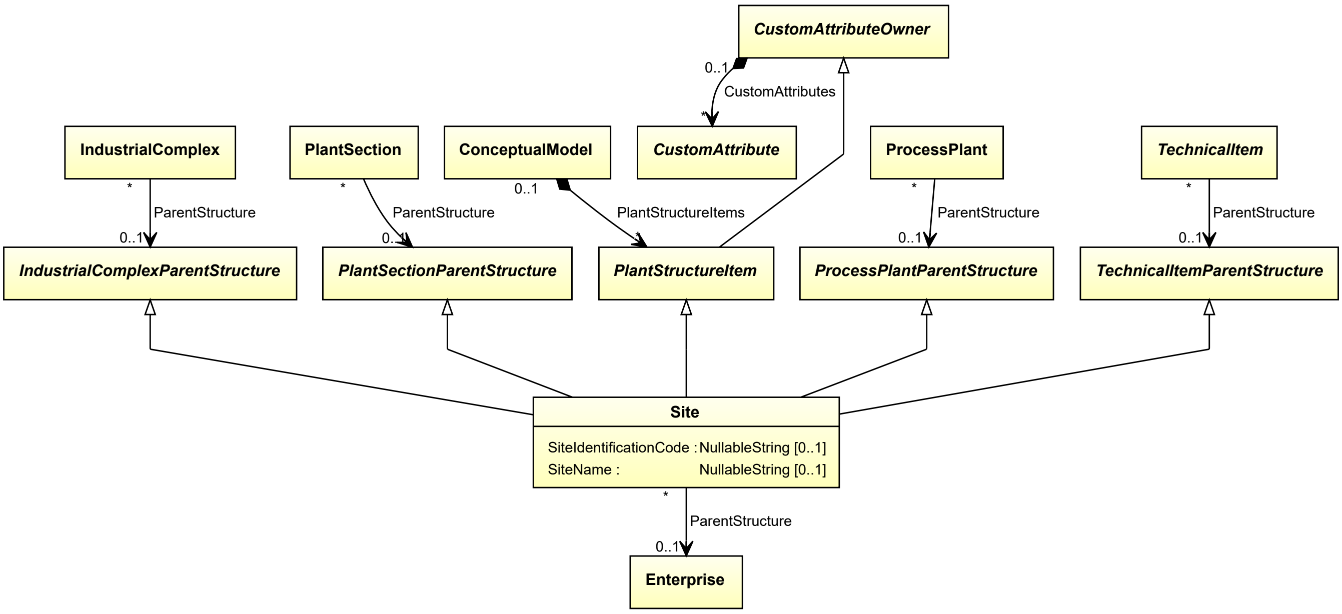
```
<PlantStructureItem
    ID="enterprise1"
    ComponentClass="Isa95Enterprise"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS10418236543" ...>
  ...
</PlantStructureItem>
```

6.16. Site

6.16.1 Overview

Class

A site as defined by ISA 95.



Supertypes

- *IndustrialComplexParentStructure*
- *PlantSectionParentStructure*
- *PlantStructureItem*
- *ProcessPlantParentStructure*
- *TechnicalItemParentStructure*

Attributes (data)

Name	Multiplicity	Type
SiteIdentificationCode	0..1	*NullableString*
SiteName	0..1	*NullableString*

Attributes (reference)

Name	Multiplicity	Type
ParentStructure	0..1	*Enterprise*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PlantStructureItem>`
RDL reference: SITE ISA95
ComponentClass: SiteIsa95
ComponentClassURI: http://data.posccaesar.org/rdl/RDS10418236632

Example

Example: Implementation in Proteus Schema

```
<PlantStructureItem
    ID="site1"
    ComponentClass="SiteIsa95"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS10418236632" ...>
  ...
</PlantStructureItem>
```

6.16.2 ParentStructure

Attribute (reference)

A superordinate structure of which the *Site* is a part.

Multiplicity: 0..1

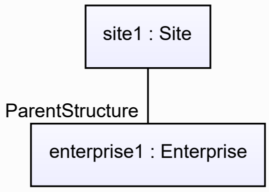
Type: *Enterprise*

Opposite multiplicity: 0..*

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: `"is a part of"`
Opposite association type: `"is a collection including"`

Example



Example: Implementation in Proteus Schema

```
<PlantStructureItem
    ID="site1"
    ComponentClass="SiteIsa95"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS10418236632" ...>
  ...
  <Association
    Type="is a part of"
    ItemID="enterprise1" />
  ...
<PlantStructureItem />
...
<PlantStructureItem
    ID="enterprise1"
    ComponentClass="Isa95Enterprise"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS10418236543" ...>
  ...
  <Association
    Type="is a collection including"
    ItemID="site1" />
  ...
<PlantStructureItem />
```

6.16.3 SiteIdentificationCode

Attribute (data)

The identification code of the site.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: SITE IDENTIFICATION CODE ASSIGNMENT CLASS
Name: SiteIdentificationCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/SiteIdentificationCodeAssignmentClass

Example

“DC” (*String*)

Example: Implementation in Proteus Schema

```
<PlantStructureItem
    ID="site1"
    ComponentClass="SiteIsa95"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS10418236632" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="SiteIdentificationCodeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/SiteIdentificationCodeAssignmentClass"
      Format="string"
      Value="DC" />
    ...
  </GenericAttributes>
  ...
</PlantStructureItem>
```

6.16.4 SiteName

Attribute (data)

The name of the site.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: SITE NAME ASSIGNMENT CLASS
Name: SiteNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/SiteNameAssignmentClass

Example

“Dexpi City” (*String*)

Example: Implementation in Proteus Schema

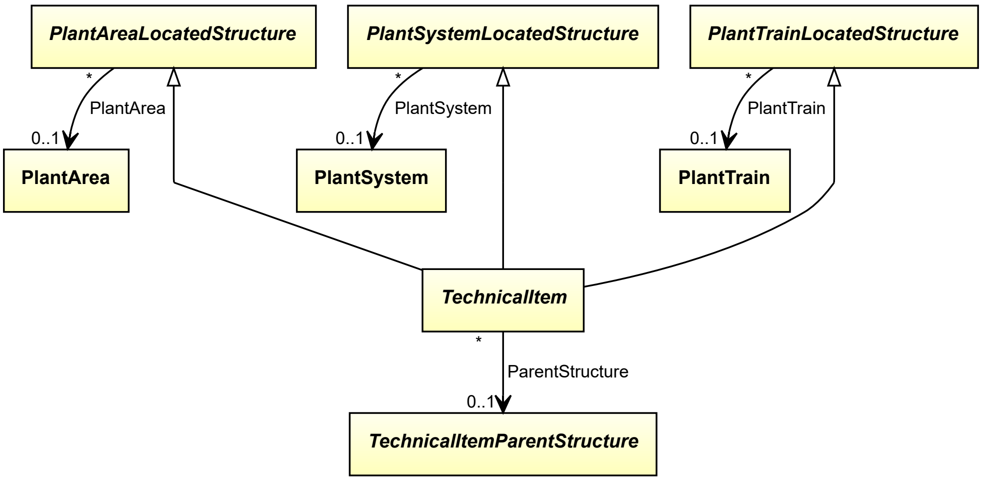
```
<PlantStructureItem
    ID="site1"
    ComponentClass="SiteIsa95"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS10418236632" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="SiteNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/SiteNameAssignmentClass"
      Format="string"
      Value="Dexpi City" />
    ...
  </GenericAttributes>
  ...
</PlantStructureItem>
```

6.17. TechnicalItem

6.17.1 Overview

Abstract class

An item at the lowest level of the plant structure.



Supertypes

- *PlantAreaLocatedStructure*
- *PlantSystemLocatedStructure*
- *PlantTrainLocatedStructure*

Subtypes

- *ActuatingElectricalFunction*
- *ActuatingElectricalSystem*
- *ActuatingFunction*
- *ActuatingSystem*
- *InstrumentationLoopFunction*
- *PipingNetworkSystem*
- *ProcessInstrumentationFunction*
- *ProcessSignalGeneratingFunction*
- *ProcessSignalGeneratingSystem*
- *TaggedPlantItem*

Attributes (reference)

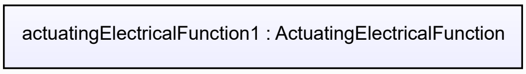
Name	Multiplicity	Type
ParentStructure	0..1	*TechnicalItemParentStructure*

Implementation in Proteus Schema

Implementation is subclass-specific.

Example

As *TechnicalItem* is abstract, we consider *ActuatingElectricalFunction* as an arbitrary concrete subclass.



Example: Implementation in Proteus Schema

```
<ActuatingElectricalFunction
    ID="actuatingElectricalFunction1"
    ComponentClass="ActuatingElectricalFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ActuatingElectricalFunction" ...>
  ...
</ActuatingElectricalFunction>
```

6.17.2 ParentStructure

Attribute (reference)

A superordinate structure of which the *TechnicalItem* is a part.

Multiplicity: 0..1

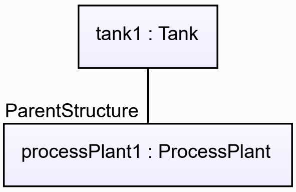
Type: *TechnicalItemParentStructure*

Opposite multiplicity: 0..*

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: `"is a part of"`
Opposite association type: `"is a collection including"`

Example



Example: Implementation in Proteus Schema

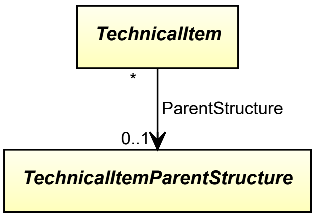
```
<Equipment
    ID="tank1"
    ComponentClass="Tank"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS445139" ...>
  ...
  <Association
    Type="is a part of"
    ItemID="processPlant1" />
  ...
<Equipment />
...
<PlantStructureItem
    ID="processPlant1"
    ComponentClass="ProcessPlant"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS7151859" ...>
  ...
  <Association
    Type="is a collection including"
    ItemID="tank1" />
  ...
<PlantStructureItem />
```

6.18. TechnicalItemParentStructure

6.18.1 Overview

Abstract class

A *PlantStructureItem* that is a suitable *ParentStructure* of a *TechnicalItem*.



Subtypes

- *Enterprise*
- *IndustrialComplex*
- *PlantSection*
- *ProcessPlant*
- *Site*

Implementation in Proteus Schema

Implementation is subclass-specific.

Example

As *TechnicalItemParentStructure* is abstract, we consider *ProcessPlant* as an arbitrary concrete subclass.

processPlant1 : ProcessPlant

Example: Implementation in Proteus Schema

```
<PlantStructureItem
    ID="processPlant1"
    ComponentClass="ProcessPlant"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS7151859" ...>
  ...
</PlantStructureItem>
```

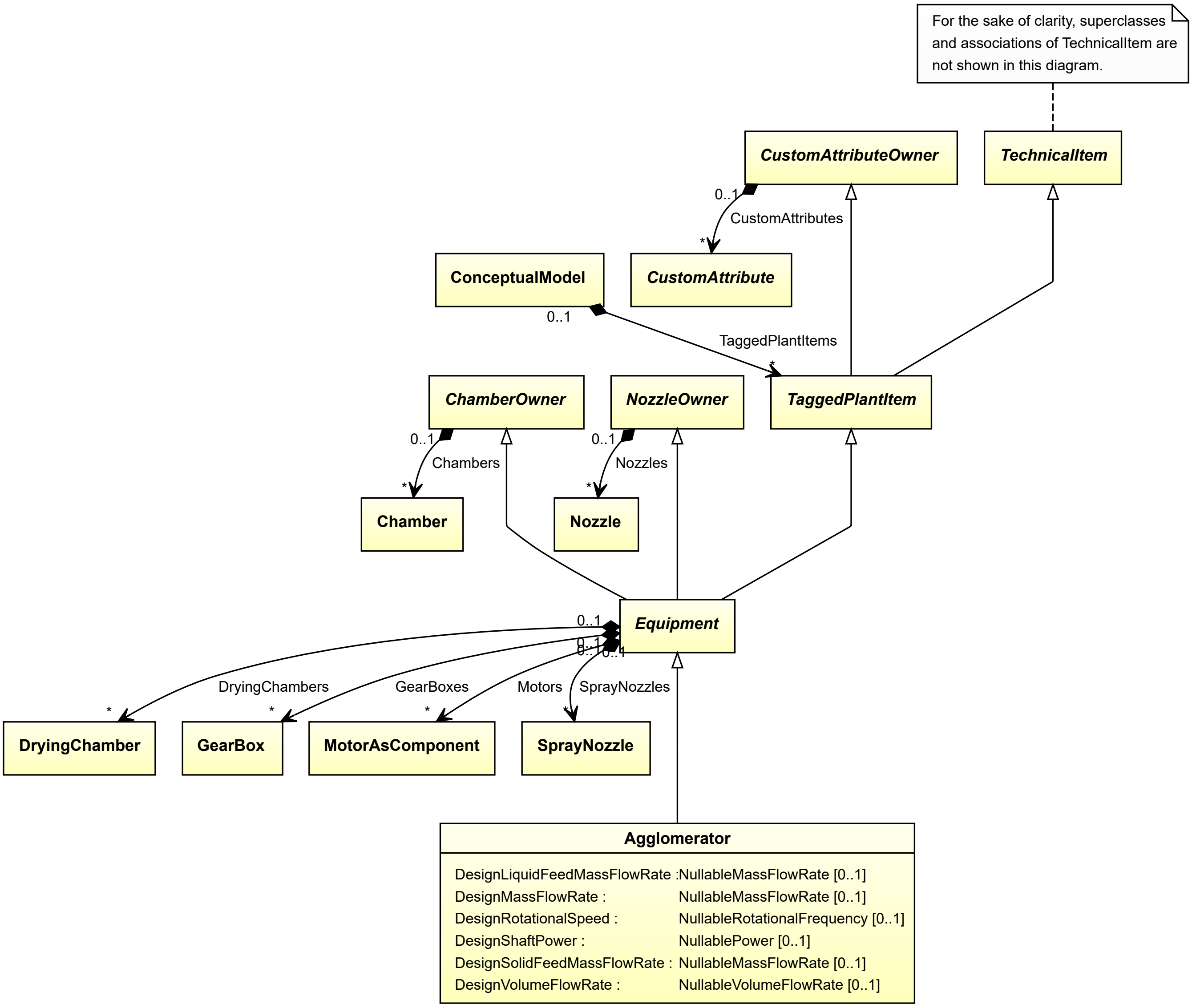
Package Equipment 7

7.1. Agglomerator

7.1.1 Overview

Class

A machine that is capable of agglomerating. It is usually vertically aligned.


For the sake of clarity, superclasses and associations of TechnicalItem are not shown in this diagram.
CustomAttributeOwner
TechnicalItem
0..1
CustomAttributes
*
ConceptualModel
CustomAttribute
0..1
TaggedPlantItems
*
ChamberOwner
NozzleOwner
TaggedPlantItem
0..1
Chambers
*
0..1
Nozzles
*
Chamber
Nozzle
0..1
Equipment
0..1
0..1
0..1
DryingChambers
GearBoxes
Motors
SprayNozzles
*
*
*
*
DryingChamber
GearBox
MotorAsComponent
SprayNozzle
Agglomerator
DesignLiquidFeedMassFlowRate : NullableMassFlowRate [0..1]
DesignMassFlowRate : NullableMassFlowRate [0..1]
DesignRotationalSpeed : NullableRotationalFrequency [0..1]
DesignShaftPower : NullablePower [0..1]
DesignSolidFeedMassFlowRate : NullableMassFlowRate [0..1]
DesignVolumeFlowRate : NullableVolumeFlowRate [0..1]

Supertypes

- *Equipment*

Subtypes

- *CustomAgglomerator*
- *ReciprocatingPressureAgglomerator*
- *RotatingGrowthAgglomerator*
- *RotatingPressureAgglomerator*

Attributes (data)

Name	Multiplicity	Type
DesignLiquidFeedMassFlowRate	0..1	*NullableMassFlowRate*
DesignMassFlowRate	0..1	*NullableMassFlowRate*
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*
DesignShaftPower	0..1	*NullablePower*
DesignSolidFeedMassFlowRate	0..1	*NullableMassFlowRate*
DesignVolumeFlowRate	0..1	*NullableVolumeFlowRate*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: AGGLOMERATOR
ComponentClass: Agglomerator
ComponentClassURI: http://sandbox.dexpi.org/rdl/Agglomerator

Example

Example: Implementation in Proteus Schema

```
<Equipment
    ID="agglomerator1"
    ComponentClass="Agglomerator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Agglomerator" ...>
  ...
</Equipment>
```

7.1.2 DesignLiquidFeedMassFlowRate

Attribute (data)

The liquid feed mass flow rate for which the *Agglomerator* is designed.

Multiplicity: 0..1

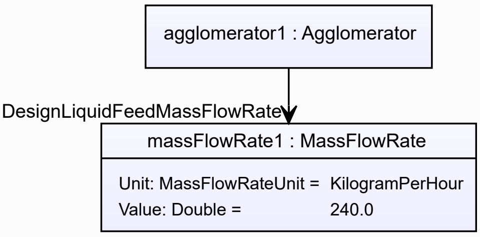
Type: *NullableMassFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN LIQUID FEED MASS FLOW RATE
Name: DesignLiquidFeedMassFlowRate
AttributeURI: http://sandbox.dexpi.org/rdl/DesignLiquidFeedMassFlowRate

Example

The instance agglomerator1 represents an *Agglomerator* with a *DesignLiquidFeedMassFlowRate* of 240.0 kg/h.


agglomerator1 : Agglomerator
DesignLiquidFeedMassFlowRate
massFlowRate1 : MassFlowRate
Unit: MassFlowRateUnit = KilogramPerHour
Value: Double = 240.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="agglomerator1"
    ComponentClass="Agglomerator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Agglomerator" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignLiquidFeedMassFlowRate"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignLiquidFeedMassFlowRate"
      Format="double"
      Value="240.0"
      Units="KilogramPerHour"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1329344" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.1.3 DesignMassFlowRate

Attribute (data)

The mass flow rate for which the *Agglomerator* is designed.

Multiplicity: 0..1

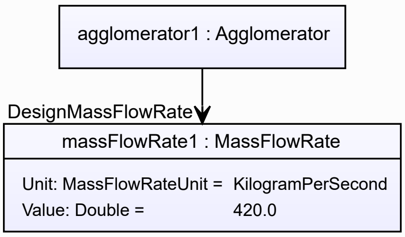
Type: *NullableMassFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN MASS FLOW RATE
Name: DesignMassFlowRate
AttributeURI: http://data.posccaesar.org/rdl/RDS14286182

Example

The instance agglomerator1 represents an *Agglomerator* with a *DesignMassFlowRate* of 420.0 kg/s.


agglomerator1 : Agglomerator
DesignMassFlowRate
massFlowRate1 : MassFlowRate
Unit: MassFlowRateUnit = KilogramPerSecond
Value: Double = 420.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="agglomerator1"
    ComponentClass="Agglomerator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Agglomerator" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignMassFlowRate"
      AttributeURI="http://data.posccaesar.org/rdl/RDS14286182"
      Format="double"
      Value="420.0"
      Units="KilogramPerSecond"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1329659" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.1.4 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *Agglomerator* is designed.

Multiplicity: 0..1

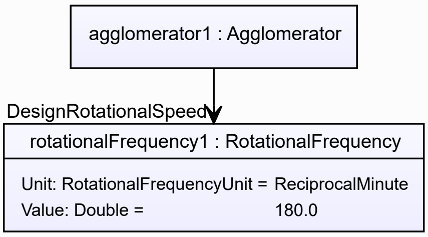
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

Example

The instance agglomerator1 represents an *Agglomerator* with a *DesignRotationalSpeed* of 180.0 min^{-1}.


agglomerator1 : Agglomerator
DesignRotationalSpeed
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 180.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="agglomerator1"
    ComponentClass="Agglomerator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Agglomerator" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignRotationalSpeed"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignRotationalSpeed"
      Format="double"
      Value="180.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.1.5 DesignShaftPower

Attribute (data)

The shaft power for which the *Agglomerator* is designed.

Multiplicity: 0..1

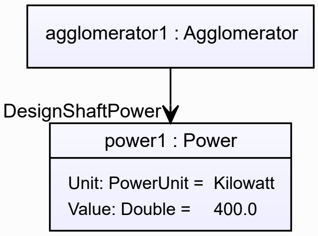
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SHAFT POWER
Name: DesignShaftPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignShaftPower

Example

The instance agglomerator1 represents an *Agglomerator* with a *DesignShaftPower* of 400.0 kW.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="agglomerator1"
    ComponentClass="Agglomerator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Agglomerator" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignShaftPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignShaftPower"
      Format="double"
      Value="400.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.1.6 DesignSolidFeedMassFlowRate

Attribute (data)

The solid feed mass flow rate for which the *Agglomerator* is designed.

Multiplicity: 0..1

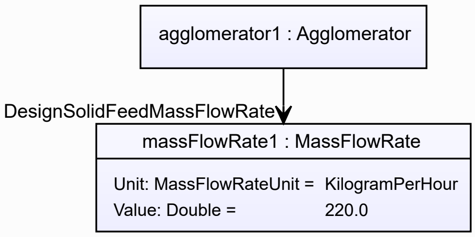
Type: *NullableMassFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SOLID FEED MASS FLOW RATE
Name: DesignSolidFeedMassFlowRate
AttributeURI: http://sandbox.dexpi.org/rdl/DesignSolidFeedMassFlowRate

Example

The instance agglomerator1 represents an *Agglomerator* with a *DesignSolidFeedMassFlowRate* of 220.0 kg/h.


agglomerator1 : Agglomerator
DesignSolidFeedMassFlowRate
massFlowRate1 : MassFlowRate
Unit: MassFlowRateUnit = KilogramPerHour
Value: Double = 220.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="agglomerator1"
    ComponentClass="Agglomerator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Agglomerator" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignSolidFeedMassFlowRate"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignSolidFeedMassFlowRate"
      Format="double"
      Value="220.0"
      Units="KilogramPerHour"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1329344" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.1.7 DesignVolumeFlowRate

Attribute (data)

The volume flow rate for which the *Agglomerator* is designed.

Multiplicity: 0..1

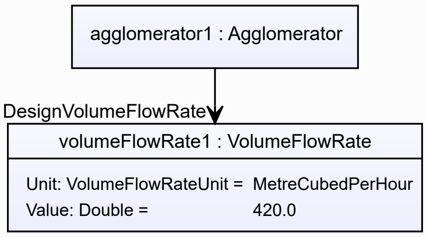
Type: *NullableVolumeFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN VOLUME FLOW RATE
Name: DesignVolumeFlowRate
AttributeURI: http://data.posccaesar.org/rdl/RDS14286227

Example

The instance agglomerator1 represents an *Agglomerator* with a *DesignVolumeFlowRate* of 420.0 m^3/h.


agglomerator1 : Agglomerator
DesignVolumeFlowRate
volumeFlowRate1 : VolumeFlowRate
Unit: VolumeFlowRateUnit = MetreCubedPerHour
Value: Double = 420.0

Example: Implementation in Proteus Schema

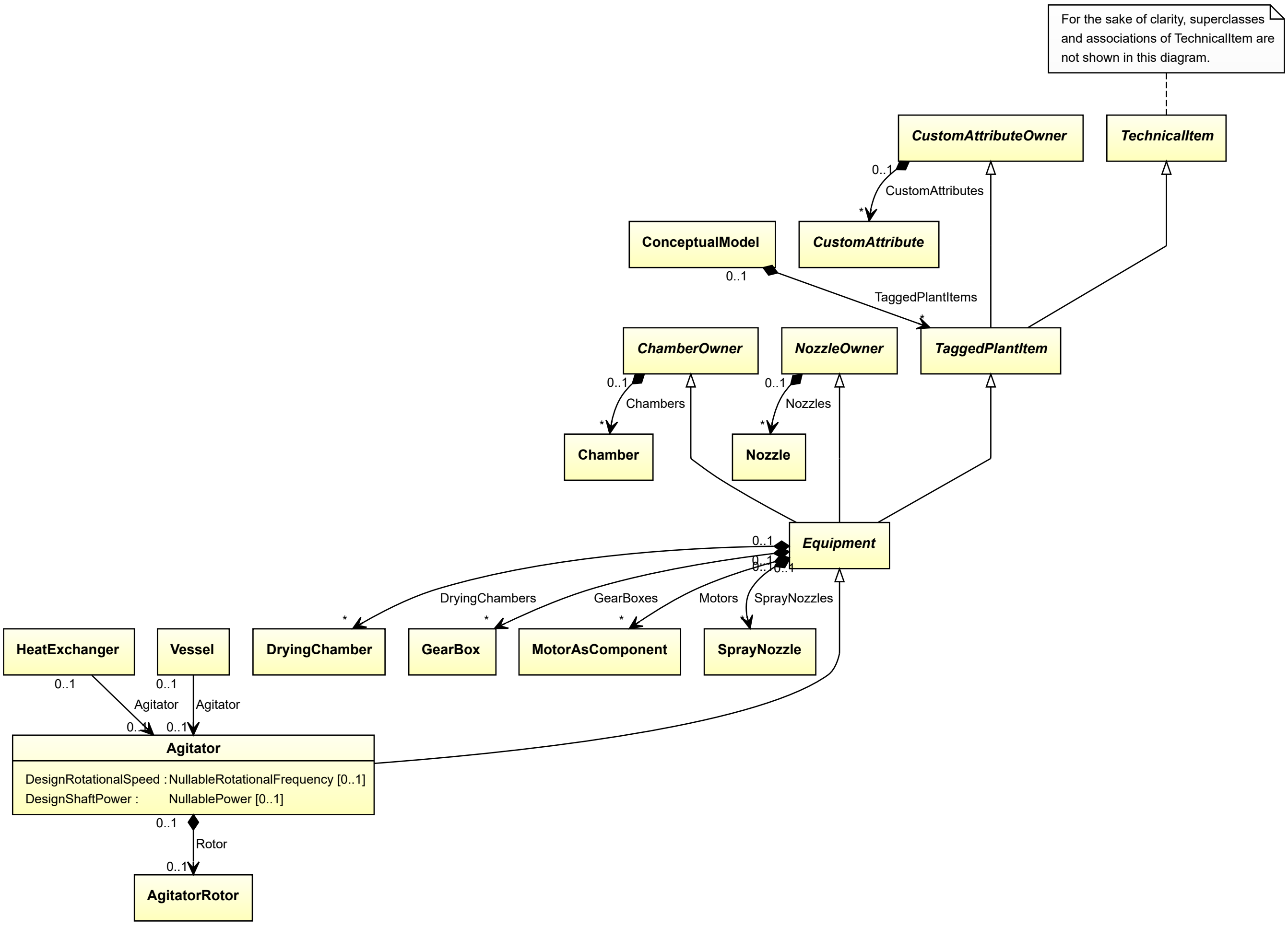
```
<Equipment
    ID="agglomerator1"
    ComponentClass="Agglomerator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Agglomerator" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignVolumeFlowRate"
      AttributeURI="http://data.posccaesar.org/rdl/RDS14286227"
      Format="double"
      Value="420.0"
      Units="MetreCubedPerHour"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1321064" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.2. Agitator

7.2.1 Overview

Class

An Agitator is a dynamic mixer that stirs or shakes fluids by reaction force from moving vanes.



Supertypes

- *Equipment*

Attributes (data)

Name	Multiplicity	Type
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*
DesignShaftPower	0..1	*NullablePower*

Attributes (composition)

Name	Multiplicity	Type
Rotor	0..1	*AgitatorRotor*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: AGITATOR
ComponentClass: Agitator
ComponentClassURI: http://data.posccaesar.org/rdl/RDS16045622

Example

Example: Implementation in Proteus Schema

```
<Equipment
    ID="agitator1"
    ComponentClass="Agitator"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS16045622" ...>
  ...
</Equipment>
```

7.2.2 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *Agitator* is designed.

Multiplicity: 0..1

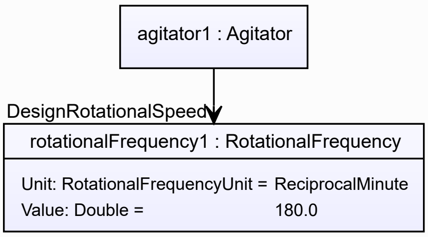
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

Example

The instance agitator1 represents an *Agitator* with a *DesignRotationalSpeed* of 180.0 min^{-1}.


agitator1 : Agitator
DesignRotationalSpeed
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 180.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="agitator1"
    ComponentClass="Agitator"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS16045622" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignRotationalSpeed"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignRotationalSpeed"
      Format="double"
      Value="180.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.2.3 DesignShaftPower

Attribute (data)

The shaft power for which the *Agitator* is designed.

Multiplicity: 0..1

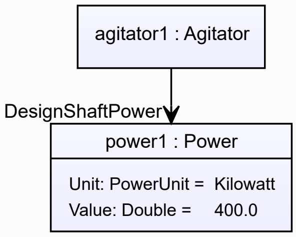
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SHAFT POWER
Name: DesignShaftPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignShaftPower

Example

The instance agitator1 represents an *Agitator* with a *DesignShaftPower* of 400.0 kW.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="agitator1"
    ComponentClass="Agitator"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS16045622" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignShaftPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignShaftPower"
      Format="double"
      Value="400.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.2.4 Rotor

Attribute (composition)

The rotor of the *Agitator*.

Multiplicity: 0..1

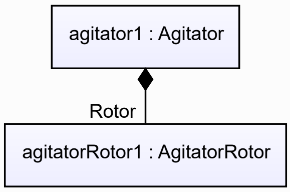
Type: *AgitatorRotor*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (an *AgitatorRotor*) is a child of the `<Equipment>` element for the attribute owner (an *Agitator*).

Example



Example: Implementation in Proteus Schema

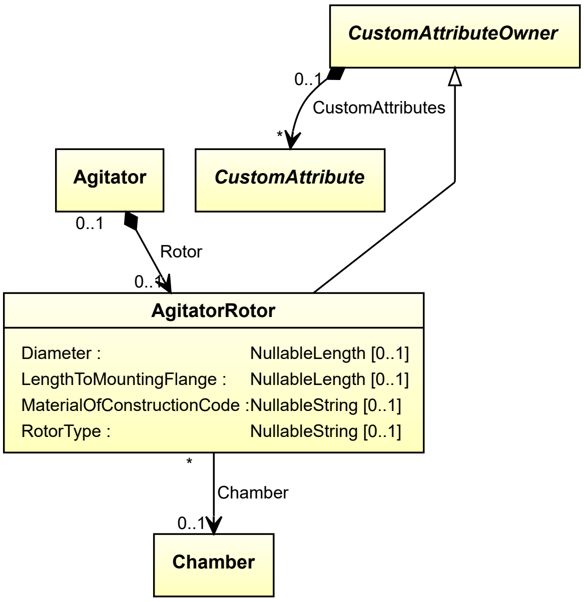
```
<Equipment
  ID="agitator1"
  ComponentClass="Agitator"
  ComponentClassURI="http://data.posccaesar.org/rdl/RDS16045622" ...>
 ...
 <Equipment
   ID="agitatorRotor1"
   ComponentClass="AgitatorRotor"
   ComponentClassURI="http://sandbox.dexpi.org/rdl/AgitatorRotor" ...>
  ...
 <Equipment />
 ...
<Equipment />
```

7.3. AgitatorRotor

7.3.1 Overview

Class

The machine component that is the rotating portion of an *Agitator*.



CustomAttributeOwner
0..1
CustomAttributes
*
Agitator
CustomAttribute
0..1
Rotor
0..1
AgitatorRotor
Diameter : NullableLength [0..1]
LengthToMountingFlange : NullableLength [0..1]
MaterialOfConstructionCode : NullableString [0..1]
RotorType : NullableString [0..1]
*
Chamber
0..1
Chamber

Supertypes

- *CustomAttributeOwner*

Attributes (data)

Name	Multiplicity	Type
Diameter	0..1	*NullableLength*
LengthToMountingFlange	0..1	*NullableLength*
MaterialOfConstructionCode	0..1	*NullableString*
RotorType	0..1	*NullableString*

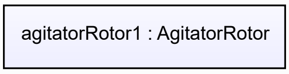
Attributes (reference)

Name	Multiplicity	Type
Chamber	0..1	*Chamber*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: AGITATOR ROTOR
ComponentClass: AgitatorRotor
ComponentClassURI: http://sandbox.dexpi.org/rdl/AgitatorRotor

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="agitatorRotor1"
    ComponentClass="AgitatorRotor"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/AgitatorRotor" ...>
  ...
</Equipment>
```

7.3.2 Chamber

Attribute (reference)

The *Chamber* in which the *AgitatorRotor* is located, if applicable. The Chamber must be a component of the same object as the *AgitatorRotor*.

Multiplicity: 0..1

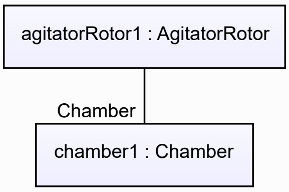
Type: *Chamber*

Opposite multiplicity: 0..*

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: `"is located in"`
Opposite association type: `"is the location of"`

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="agitatorRotor1"
    ComponentClass="AgitatorRotor"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/AgitatorRotor" ...>
  ...
  <Association
    Type="is located in"
    ItemID="chamber1" />
  ...
<Equipment />
...
<Equipment
    ID="chamber1"
    ComponentClass="Chamber"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS903151421" ...>
  ...
  <Association
    Type="is the location of"
    ItemID="agitatorRotor1" />
  ...
<Equipment />
```

7.3.3 Diameter

Attribute (data)

The diameter of the *AgitatorRotor*.

Multiplicity: 0..1

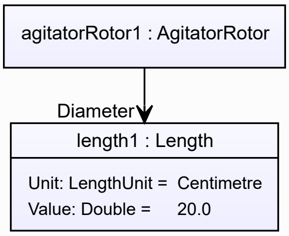
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DIAMETER
Name: Diameter
AttributeURI: http://data.posccaesar.org/rdl/RDS350954

Example

The instance agitatorRotor1 represents an *AgitatorRotor* with a *Diameter* of 20.0 cm.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="agitatorRotor1"
    ComponentClass="AgitatorRotor"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/AgitatorRotor" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="Diameter"
      AttributeURI="http://data.posccaesar.org/rdl/RDS350954"
      Format="double"
      Value="20.0"
      Units="Centimetre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1318004" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.3.4 LengthToMountingFlange

Attribute (data)

The length to the mounting flange of the *AgitatorRotor*.

Multiplicity: 0..1

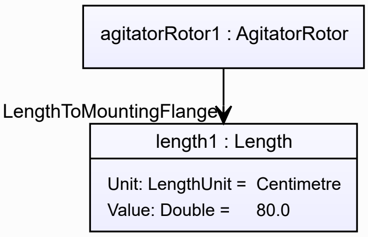
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: LENGTH TO MOUNTING FLANGE
Name: LengthToMountingFlange
AttributeURI: http://sandbox.dexpi.org/rdl/LengthToMountingFlange

Example

The instance agitatorRotor1 represents an *AgitatorRotor* with a *LengthToMountingFlange* of 80.0 cm.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="agitatorRotor1"
    ComponentClass="AgitatorRotor"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/AgitatorRotor" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="LengthToMountingFlange"
      AttributeURI="http://sandbox.dexpi.org/rdl/LengthToMountingFlange"
      Format="double"
      Value="80.0"
      Units="Centimetre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1318004" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.3.5 MaterialOfConstructionCode

Attribute (data)

A code that gives the material of construction of the *AgitatorRotor*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: MATERIAL OF CONSTRUCTION CODE ASSIGNMENT CLASS
Name: MaterialOfConstructionCodeAssignmentClass
AttributeURI: http://data.posccaesar.org/rdl/RDS1460719741

Example

“1.4306” (*String*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="agitatorRotor1"
    ComponentClass="AgitatorRotor"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/AgitatorRotor" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="MaterialOfConstructionCodeAssignmentClass"
      AttributeURI="http://data.posccaesar.org/rdl/RDS1460719741"
      Format="string"
      Value="1.4306" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.3.6 RotorType

Attribute (data)

The rotor type of the *AgitatorRotor*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: ROTOR TYPE ASSIGNMENT CLASS
Name: RotorTypeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/RotorTypeAssignmentClass

Example

“xy1” (*String*)

Example: Implementation in Proteus Schema

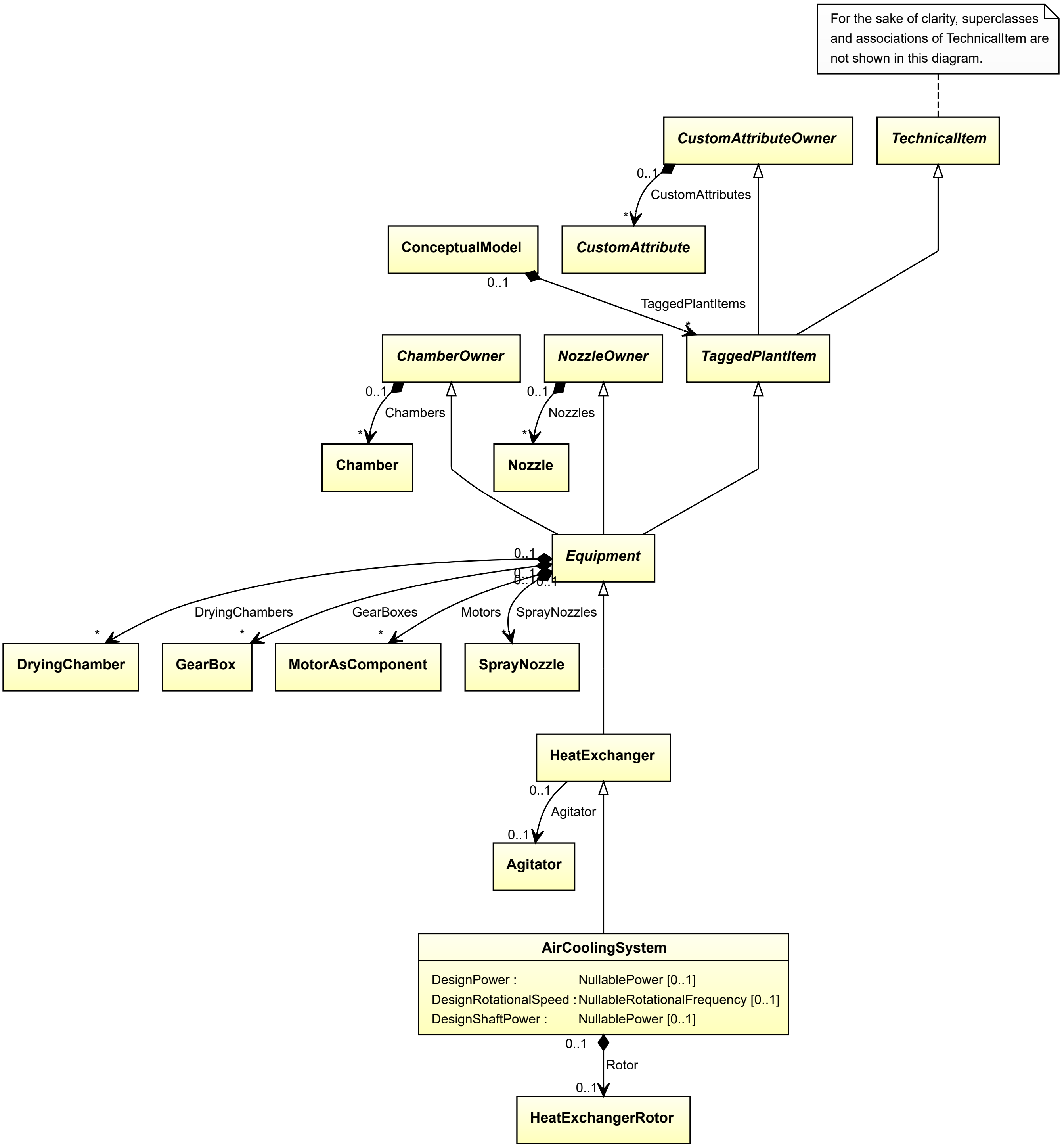
```
<Equipment
    ID="agitatorRotor1"
    ComponentClass="AgitatorRotor"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/AgitatorRotor" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="RotorTypeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/RotorTypeAssignmentClass"
      Format="string"
      Value="xy1" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.4. AirCoolingSystem

7.4.1 Overview

Class

A cooling system which uses air as the cooling medium (from http://data.posccaesar.org/rdl/RDS277379).



Supertypes

- *HeatExchanger*

Attributes (data)

Name	Multiplicity	Type
DesignPower	0..1	*NullablePower*
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*
DesignShaftPower	0..1	*NullablePower*

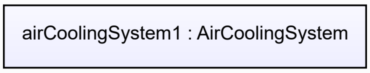
Attributes (composition)

Name	Multiplicity	Type
Rotor	0..1	*HeatExchangerRotor*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: AIR COOLING SYSTEM
ComponentClass: AirCoolingSystem
ComponentClassURI: http://data.posccaesar.org/rdl/RDS277379

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="airCoolingSystem1"
    ComponentClass="AirCoolingSystem"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS277379" ...>
  ...
</Equipment>
```

7.4.2 DesignPower

Attribute (data)

The power for which the *AirCoolingSystem* is designed.

Multiplicity: 0..1

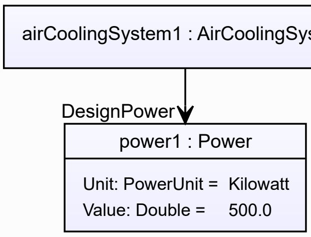
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN POWER
Name: DesignPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignPower

Example

The instance airCoolingSystem1 represents an *AirCoolingSystem* with a *DesignPower* of 500.0 kW.


airCoolingSystem1 : AirCoolingSystem
DesignPower
power1 : Power
Unit: PowerUnit = Kilowatt
Value: Double = 500.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="airCoolingSystem1"
    ComponentClass="AirCoolingSystem"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS277379" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignPower"
      Format="double"
      Value="500.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.4.3 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *AirCoolingSystem* is designed.

Multiplicity: 0..1

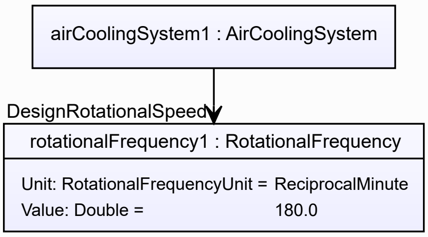
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

Example

The instance airCoolingSystem1 represents an *AirCoolingSystem* with a *DesignRotationalSpeed* of 180.0 min^{-1}.


airCoolingSystem1 : AirCoolingSystem
DesignRotationalSpeed
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 180.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="airCoolingSystem1"
    ComponentClass="AirCoolingSystem"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS277379" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignRotationalSpeed"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignRotationalSpeed"
      Format="double"
      Value="180.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.4.4 DesignShaftPower

Attribute (data)

The shaft power for which the *AirCoolingSystem* is designed.

Multiplicity: 0..1

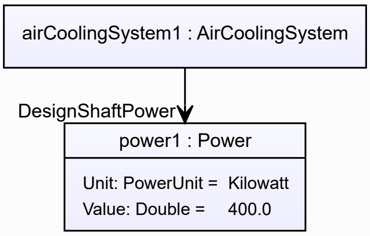
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SHAFT POWER
Name: DesignShaftPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignShaftPower

Example

The instance airCoolingSystem1 represents an *AirCoolingSystem* with a *DesignShaftPower* of 400.0 kW.


airCoolingSystem1 : AirCoolingSystem
DesignShaftPower
power1 : Power
Unit: PowerUnit = Kilowatt
Value: Double = 400.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="airCoolingSystem1"
    ComponentClass="AirCoolingSystem"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS277379" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignShaftPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignShaftPower"
      Format="double"
      Value="400.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.4.5 Rotor

Attribute (composition)

The rotor of the *AirCoolingSystem*.

Multiplicity: 0..1

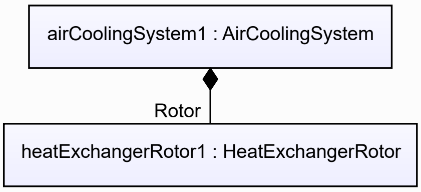
Type: *HeatExchangerRotor*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *HeatExchangerRotor*) is a child of the `<Equipment>` element for the attribute owner (an *AirCoolingSystem*).

Example


airCoolingSystem1 : AirCoolingSystem
Rotor
heatExchangerRotor1 : HeatExchangerRotor

Example: Implementation in Proteus Schema

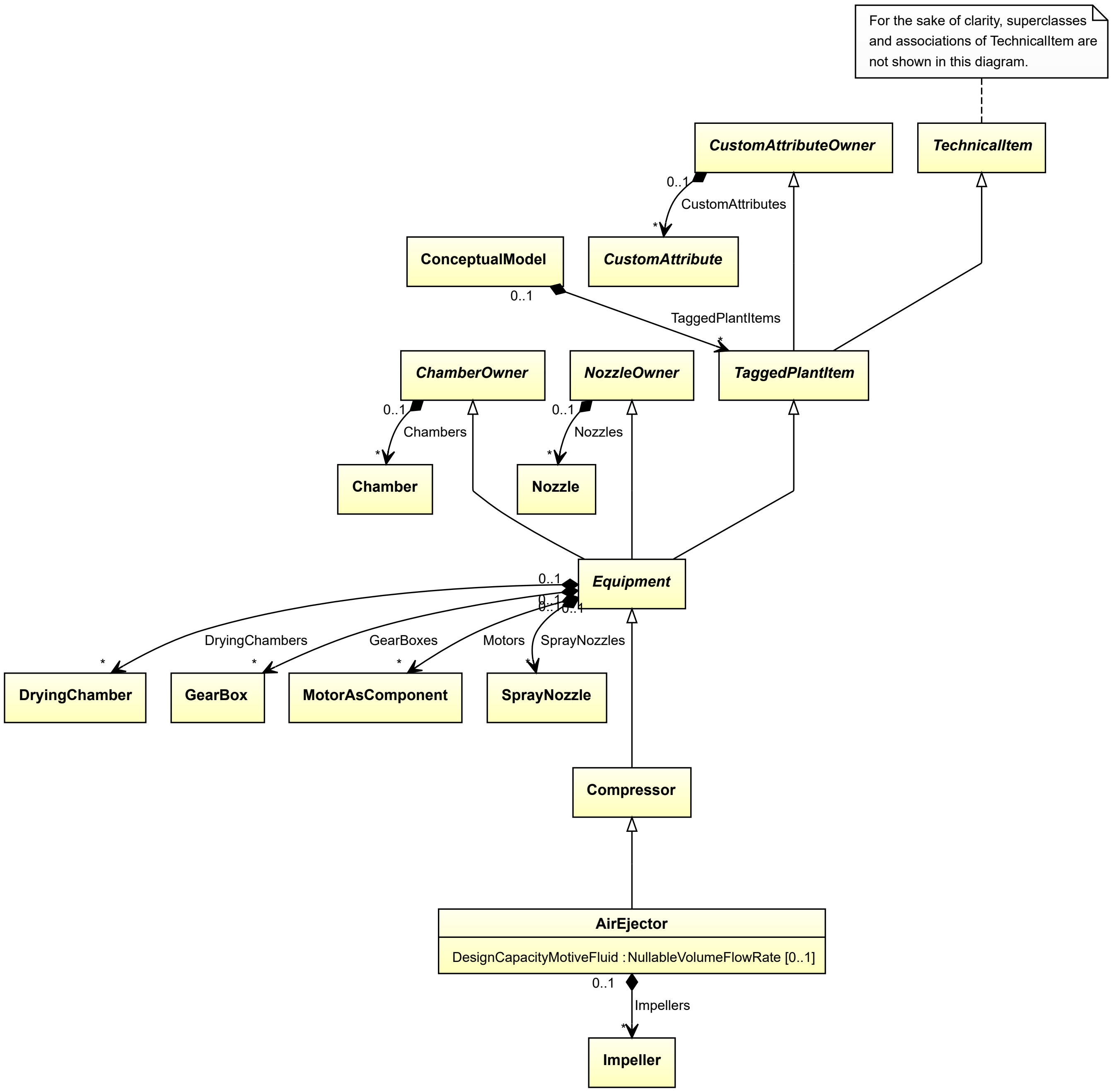
```
<Equipment
    ID="airCoolingSystem1"
    ComponentClass="AirCoolingSystem"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS277379" ...>
  ...
  <Equipment
      ID="heatExchangerRotor1"
      ComponentClass="HeatExchangerRotor"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/HeatExchangerRotor" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.5. AirEjector

7.5.1 Overview

Class

An ejector intended to create vacuum using compressed air (from http://data.posccaesar.org/rdl/RDS5770157).



Supertypes

- *Compressor*

Attributes (data)

Name	Multiplicity	Type
DesignCapacityMotiveFluid	0..1	*NullableVolumeFlowRate*

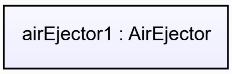
Attributes (composition)

Name	Multiplicity	Type
Impellers	*	*Impeller*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: AIR EJECTOR
ComponentClass: AirEjector
ComponentClassURI: http://data.posccaesar.org/rdl/RDS5770157

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="airEjector1"
    ComponentClass="AirEjector"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS5770157" ...>
  ...
</Equipment>
```

7.5.2 DesignCapacityMotiveFluid

Attribute (data)

The capacity of the volume flow rate for the motive fluid for which the *AirEjector* is designed.

Multiplicity: 0..1

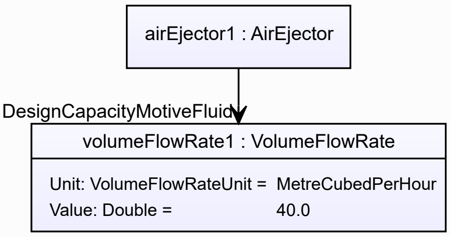
Type: *NullableVolumeFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN CAPACITY MOTIVE FLUID
Name: DesignCapacityMotiveFluid
AttributeURI: http://sandbox.dexpi.org/rdl/DesignCapacityMotiveFluid

Example

The instance airEjector1 represents an *AirEjector* with a *DesignCapacityMotiveFluid* of 40.0 m^3/h.


airEjector1 : AirEjector
DesignCapacityMotiveFluid
volumeFlowRate1 : VolumeFlowRate
Unit: VolumeFlowRateUnit = MetreCubedPerHour
Value: Double = 40.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="airEjector1"
    ComponentClass="AirEjector"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS5770157" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignCapacityMotiveFluid"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignCapacityMotiveFluid"
      Format="double"
      Value="40.0"
      Units="MetreCubedPerHour"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1321064" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.5.3 Impellers

Attribute (composition)

The impellers of the *AirEjector*.

Multiplicity: *

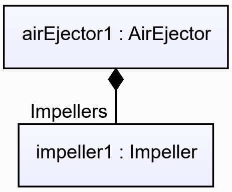
Type: *Impeller*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (an *Impeller*) is a child of the `<Equipment>` element for the attribute owner (an *AirEjector*).

Example



Example: Implementation in Proteus Schema

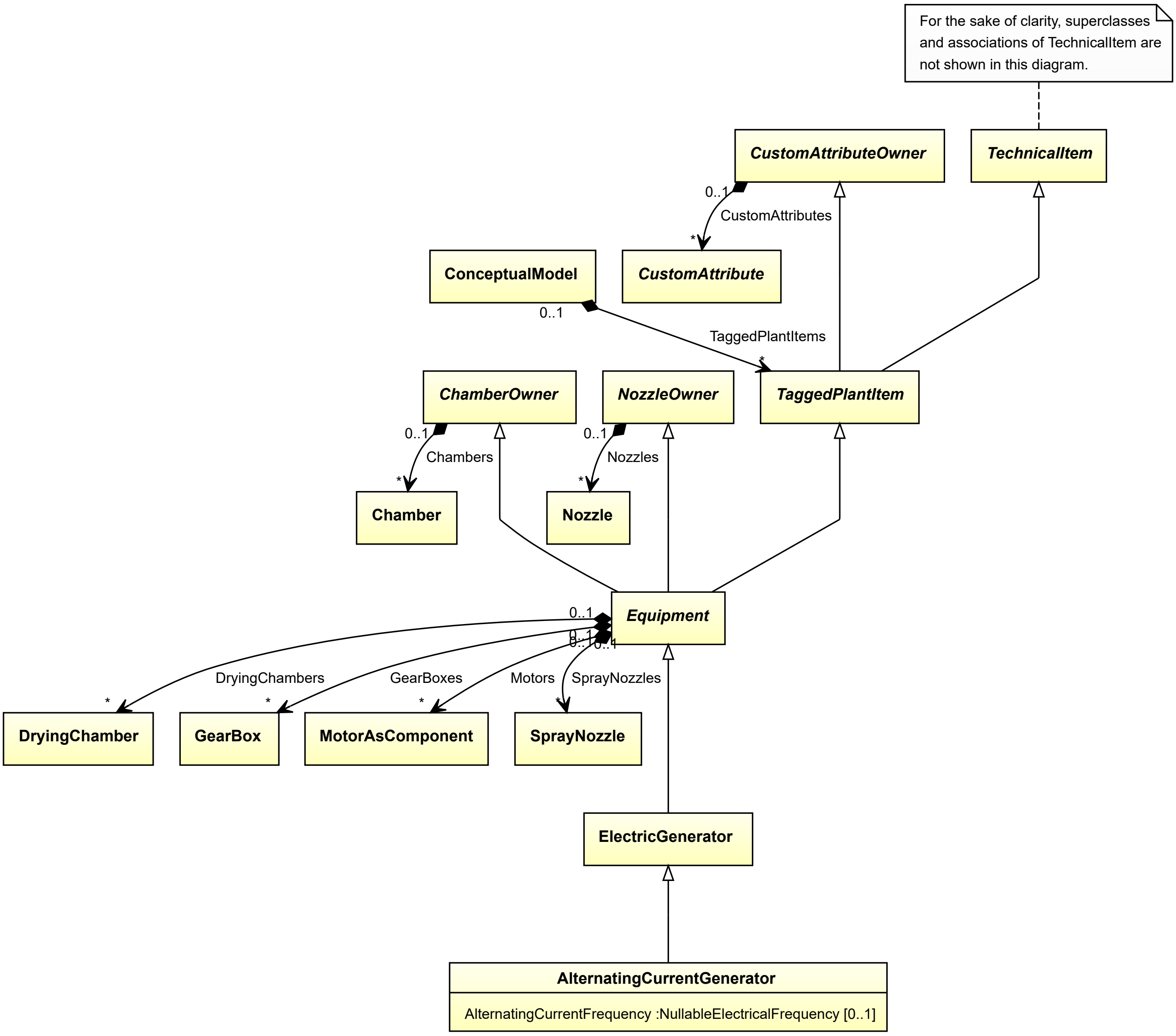
```
<Equipment
    ID="airEjector1"
    ComponentClass="AirEjector"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS5770157" ...>
  ...
  <Equipment
      ID="impeller1"
      ComponentClass="Impeller"
      ComponentClassURI="http://data.posccaesar.org/rdl/RDS414539" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.6. AlternatingCurrentGenerator

7.6.1 Overview

Class

An electric generator for the production of alternating current and voltage (from http://data.posccaesar.org/rdl/RDS873359).



Supertypes

- *ElectricGenerator*

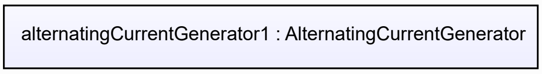
Attributes (data)

Name	Multiplicity	Type
AlternatingCurrentFrequency	0..1	*NullableElectricalFrequency*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: ALTERNATING CURRENT GENERATOR
ComponentClass: AlternatingCurrentGenerator
ComponentClassURI: http://data.posccaesar.org/rdl/RDS873359

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="alternatingCurrentGenerator1"
    ComponentClass="AlternatingCurrentGenerator"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS873359" ...>
  ...
</Equipment>
```

7.6.2 AlternatingCurrentFrequency

Attribute (data)

The alternating current frequency of the *AlternatingCurrentGenerator*.

Multiplicity: 0..1

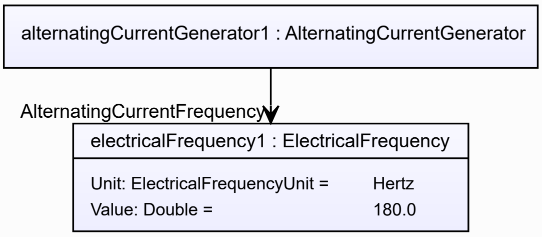
Type: *NullableElectricalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: ALTERNATING CURRENT FREQUENCY
Name: AlternatingCurrentFrequency
AttributeURI: http://sandbox.dexpi.org/rdl/AlternatingCurrentFrequency

Example

The instance alternatingCurrentGenerator1 represents an *AlternatingCurrentGenerator* with an *AlternatingCurrentFrequency* of 180.0 Hz.


alternatingCurrentGenerator1 : AlternatingCurrentGenerator
AlternatingCurrentFrequency
electricalFrequency1 : ElectricalFrequency
Unit: ElectricalFrequencyUnit =
Hertz
Value: Double =
180.0

Example: Implementation in Proteus Schema

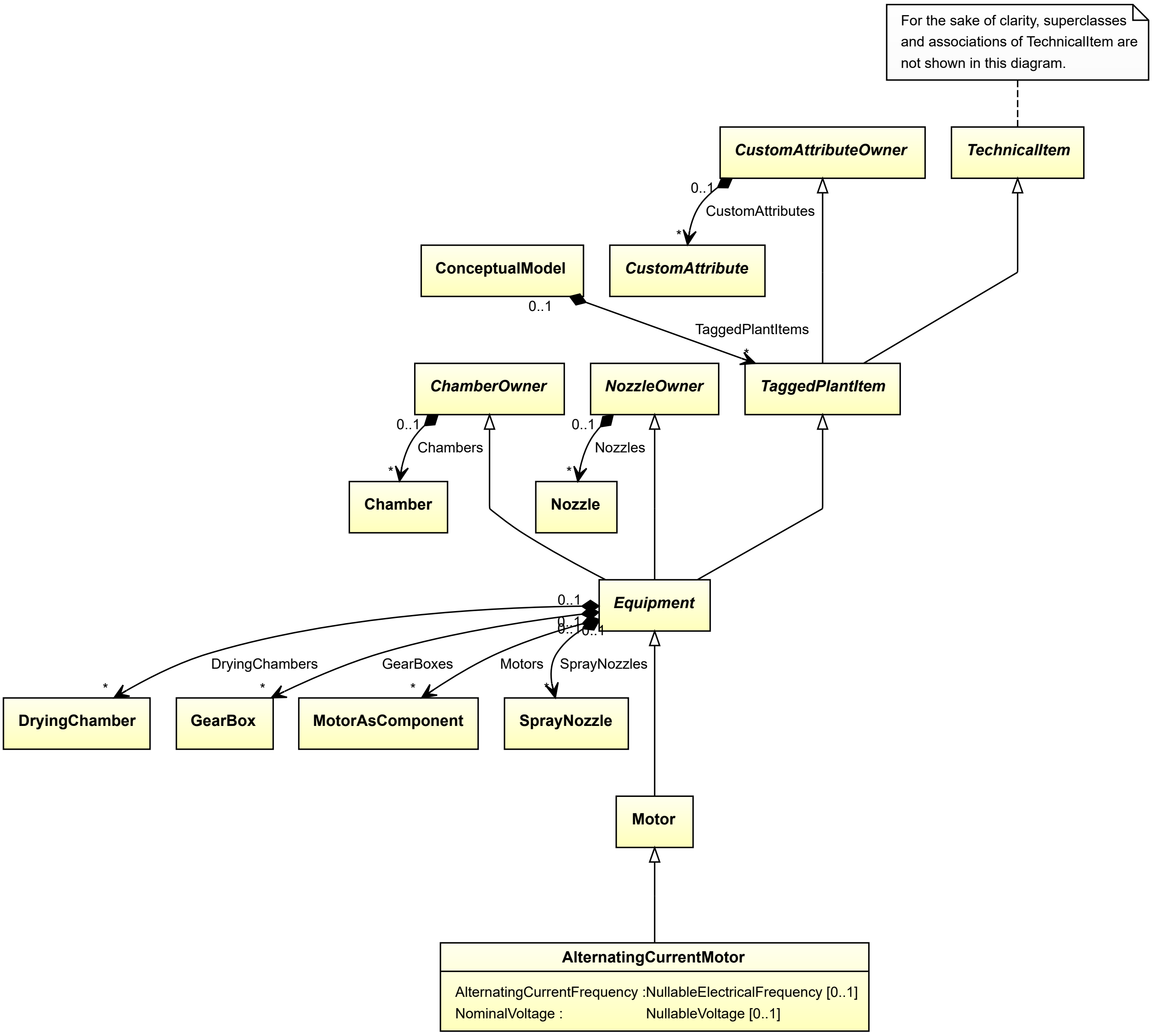
```
<Equipment
    ID="alternatingCurrentGenerator1"
    ComponentClass="AlternatingCurrentGenerator"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS873359" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="AlternatingCurrentFrequency"
      AttributeURI="http://sandbox.dexpi.org/rdl/AlternatingCurrentFrequency"
      Format="double"
      Value="180.0"
      Units="Hertz"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1326464" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.7. AlternatingCurrentMotor

7.7.1 Overview

Class

An electric motor driven by alternating electric current (from http://data.posccaesar.org/rdl/RDS472994).



Supertypes

- *Motor*

Attributes (data)

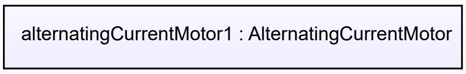
Name	Multiplicity	Type
AlternatingCurrentFrequency	0..1	*NullableElectricalFrequency*
NominalVoltage	0..1	*NullableVoltage*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: ALTERNATING CURRENT MOTOR
ComponentClass: AlternatingCurrentMotor

ComponentClassURI: http://data.posccaesar.org/rdl/RDS472994

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="alternatingCurrentMotor1"
    ComponentClass="AlternatingCurrentMotor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS472994" ...>
  ...
</Equipment>
```

7.7.2 AlternatingCurrentFrequency

Attribute (data)

The alternating current frequency of the *AlternatingCurrentMotor*.

Multiplicity: 0..1

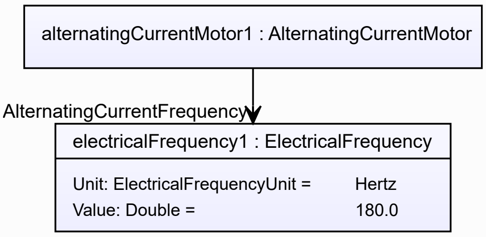
Type: *NullableElectricalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: ALTERNATING CURRENT FREQUENCY
Name: AlternatingCurrentFrequency
AttributeURI: http://sandbox.dexpi.org/rdl/AlternatingCurrentFrequency

Example

The instance alternatingCurrentMotor1 represents an *AlternatingCurrentMotor* with an *AlternatingCurrentFrequency* of 180.0 Hz.


alternatingCurrentMotor1 : AlternatingCurrentMotor
AlternatingCurrentFrequency
electricalFrequency1 : ElectricalFrequency
Unit: ElectricalFrequencyUnit =
Hertz
Value: Double =
180.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="alternatingCurrentMotor1"
    ComponentClass="AlternatingCurrentMotor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS472994" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="AlternatingCurrentFrequency"
      AttributeURI="http://sandbox.dexpi.org/rdl/AlternatingCurrentFrequency"
      Format="double"
      Value="180.0"
      Units="Hertz"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1326464" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.7.3 NominalVoltage

Attribute (data)

The nominal voltage of the *AlternatingCurrentMotor*.

Multiplicity: 0..1

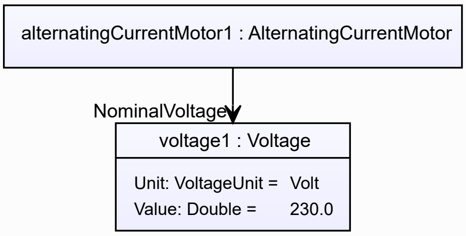
Type: *NullableVoltage*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: NOMINAL VOLTAGE
Name: NominalVoltage
AttributeURI: http://data.posccaesar.org/rdl/RDS369449

Example

The instance alternatingCurrentMotor1 represents an *AlternatingCurrentMotor* with a *NominalVoltage* of 230.0 V.



Example: Implementation in Proteus Schema

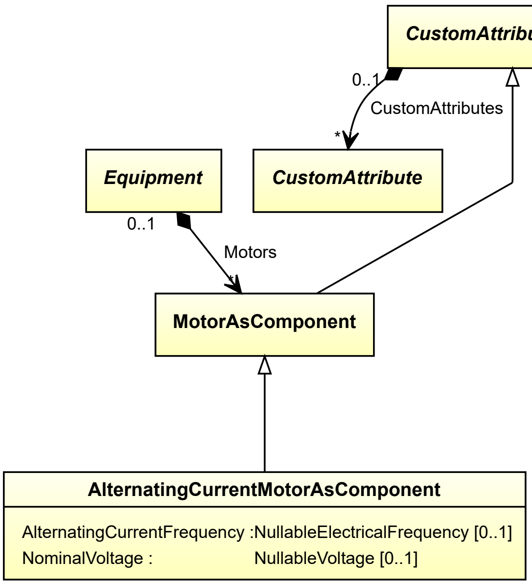
```
<Equipment
    ID="alternatingCurrentMotor1"
    ComponentClass="AlternatingCurrentMotor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS472994" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NominalVoltage"
      AttributeURI="http://data.posccaesar.org/rdl/RDS369449"
      Format="double"
      Value="230.0"
      Units="Volt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1347974" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.8. AlternatingCurrentMotorAsComponent

7.8.1 Overview

Class

An electric motor driven by alternating electric current that is used as a component of an apparatus or of a machine.


CustomAttributeOwner
0..1
CustomAttributes
*
Equipment
CustomAttribute
0..1
Motors
MotorAsComponent
AlternatingCurrentMotorAsComponent
AlternatingCurrentFrequency :NullableElectricalFrequency [0..1]
NominalVoltage : NullableVoltage [0..1]

Supertypes

- *MotorAsComponent*

Attributes (data)

Name	Multiplicity	Type
AlternatingCurrentFrequency	0..1	*NullableElectricalFrequency*
NominalVoltage	0..1	*NullableVoltage*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: ALTERNATING CURRENT MOTOR AS COMPONENT
ComponentClass: AlternatingCurrentMotorAsComponent
ComponentClassURI: http://sandbox.dexpi.org/rdl/AlternatingCurrentMotorAsComponent

Example

alternatingCurrentMotorAsComponent1 : AlternatingCurrentMotorAsComponent

Example: Implementation in Proteus Schema

```
<Equipment
    ID="alternatingCurrentMotorAsComponent1"
    ComponentClass="AlternatingCurrentMotorAsComponent"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/AlternatingCurrentMotorAsComponent" ...>
  ...
</Equipment>
```

7.8.2 AlternatingCurrentFrequency

Attribute (data)

The alternating current frequency of the *AlternatingCurrentMotorAsComponent*.

Multiplicity: 0..1

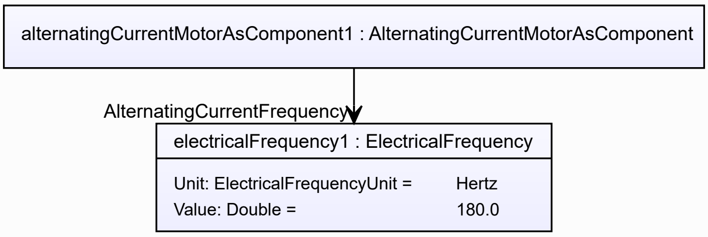
Type: *NullableElectricalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: ALTERNATING CURRENT FREQUENCY
Name: AlternatingCurrentFrequency
AttributeURI: http://sandbox.dexpi.org/rdl/AlternatingCurrentFrequency

Example

The instance alternatingCurrentMotorAsComponent1 represents an *AlternatingCurrentMotorAsComponent* with an *AlternatingCurrentFrequency* of 180.0 Hz.


alternatingCurrentMotorAsComponent1 : AlternatingCurrentMotorAsComponent
AlternatingCurrentFrequency
electricalFrequency1 : ElectricalFrequency
Unit: ElectricalFrequencyUnit =
Hertz
Value: Double =
180.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="alternatingCurrentMotorAsComponent1"
    ComponentClass="AlternatingCurrentMotorAsComponent"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/AlternatingCurrentMotorAsComponent" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="AlternatingCurrentFrequency"
      AttributeURI="http://sandbox.dexpi.org/rdl/AlternatingCurrentFrequency"
      Format="double"
      Value="180.0"
      Units="Hertz"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1326464" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.8.3 NominalVoltage

Attribute (data)

The nominal voltage of the *AlternatingCurrentMotorAsComponent*.

Multiplicity: 0..1

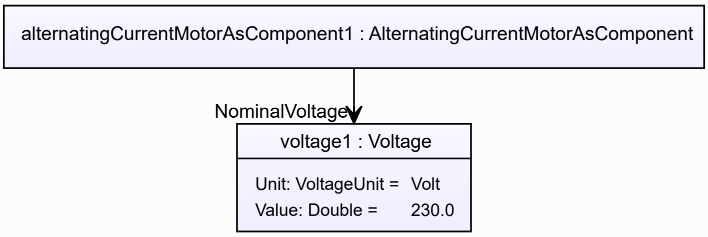
Type: *NullableVoltage*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: NOMINAL VOLTAGE
Name: NominalVoltage
AttributeURI: http://data.posccaesar.org/rdl/RDS369449

Example

The instance alternatingCurrentMotorAsComponent1 represents an *AlternatingCurrentMotorAsComponent* with a *NominalVoltage* of 230.0 V.



Example: Implementation in Proteus Schema

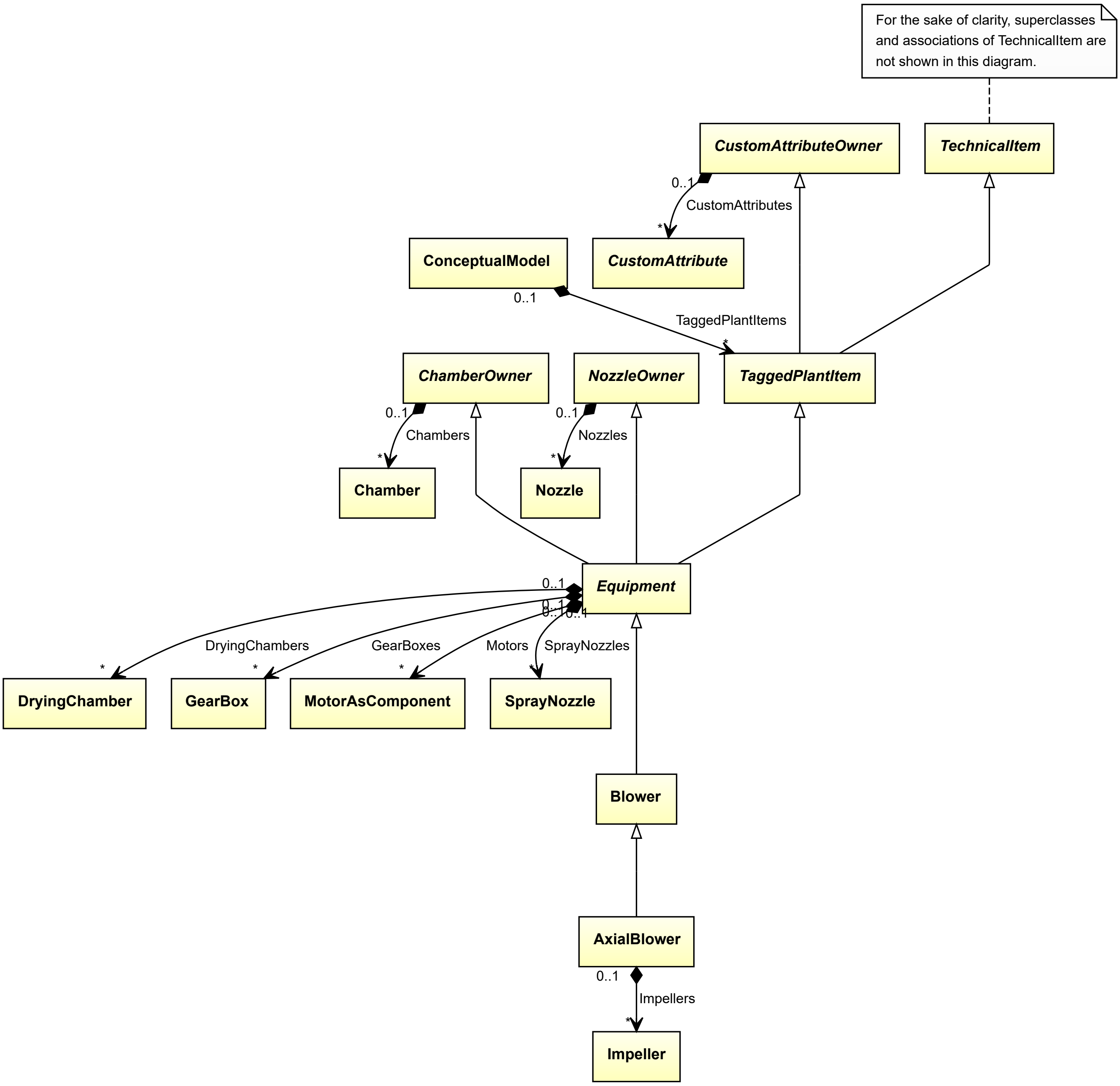
```
<Equipment
    ID="alternatingCurrentMotorAsComponent1"
    ComponentClass="AlternatingCurrentMotorAsComponent"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/AlternatingCurrentMotorAsComponent" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NominalVoltage"
      AttributeURI="http://data.posccaesar.org/rdl/RDS369449"
      Format="double"
      Value="230.0"
      Units="Volt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1347974" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.9. AxialBlower

7.9.1 Overview

Class

A blower in which the flow direction is parallel to the shaft (from http://data.posccaesar.org/rdl/RDS433259).



Supertypes

- *Blower*

Attributes (composition)

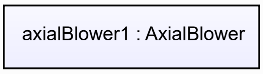
Name	Multiplicity	Type
Impellers	*	*Impeller*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`

RDL reference: AXIAL BLOWER
ComponentClass: AxialBlower
ComponentClassURI: http://data.posccaesar.org/rdl/RDS433259

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="axialBlower1"
    ComponentClass="AxialBlower"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS433259" ...>
  ...
</Equipment>
```

7.9.2 Impellers

Attribute (composition)

The impellers of the *AxialBlower*.

Multiplicity: *

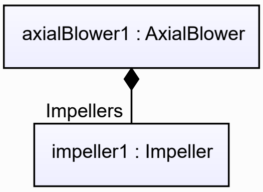
Type: *Impeller*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (an *Impeller*) is a child of the `<Equipment>` element for the attribute owner (an *AxialBlower*).

Example



Example: Implementation in Proteus Schema

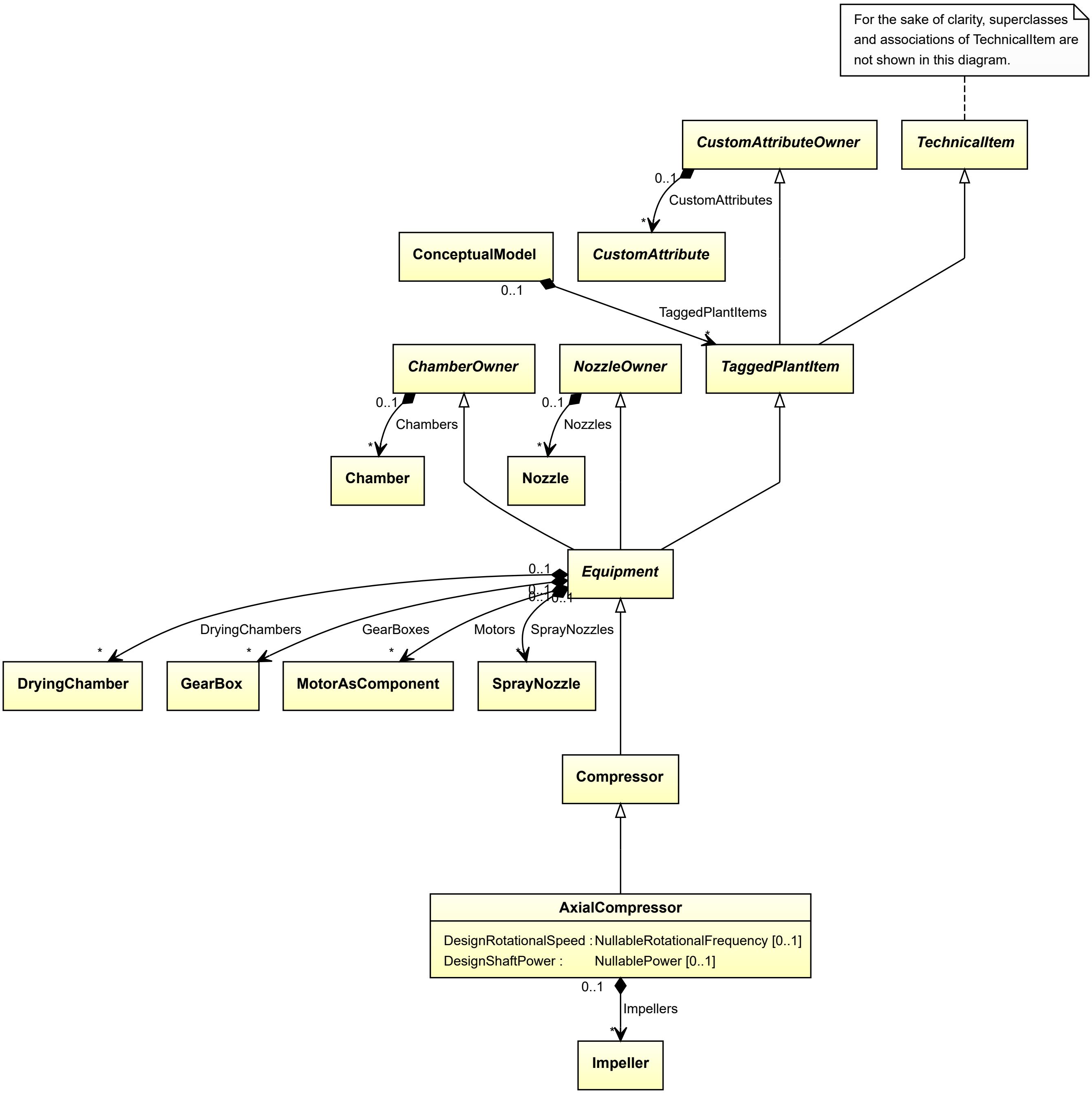
```
<Equipment
    ID="axialBlower1"
    ComponentClass="AxialBlower"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS433259" ...>
  ...
  <Equipment
      ID="impeller1"
      ComponentClass="Impeller"
      ComponentClassURI="http://data.posccaesar.org/rdl/RDS414539" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.10. AxialCompressor

7.10.1 Overview

Class

A *Compressor* in which the gas is accelerated by the action of a bladed rotor and where the main flow is along the rotation axis of the rotor.



Supertypes

- *Compressor*

Attributes (data)

Name	Multiplicity	Type
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*
DesignShaftPower	0..1	*NullablePower*

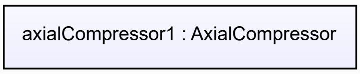
Attributes (composition)

Name	Multiplicity	Type
Impellers	*	*Impeller*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: AXIAL COMPRESSOR
ComponentClass: AxialCompressor
ComponentClassURI: http://data.posccaesar.org/rdl/RDS417239

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="axialCompressor1"
    ComponentClass="AxialCompressor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS417239" ...>
  ...
</Equipment>
```

7.10.2 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *AxialCompressor* is designed.

Multiplicity: 0..1

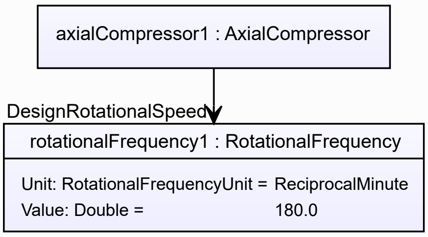
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

Example

The instance axialCompressor1 represents an *AxialCompressor* with a *DesignRotationalSpeed* of 180.0 min^{-1}.


axialCompressor1 : AxialCompressor
DesignRotationalSpeed
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 180.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="axialCompressor1"
    ComponentClass="AxialCompressor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS417239" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignRotationalSpeed"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignRotationalSpeed"
      Format="double"
      Value="180.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.10.3 DesignShaftPower

Attribute (data)

The shaft power for which the *AxialCompressor* is designed.

Multiplicity: 0..1

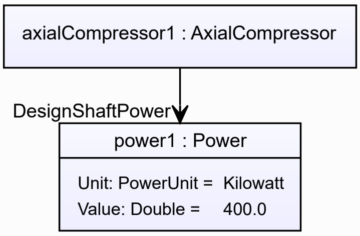
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SHAFT POWER
Name: DesignShaftPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignShaftPower

Example

The instance axialCompressor1 represents an *AxialCompressor* with a *DesignShaftPower* of 400.0 kW.


axialCompressor1 : AxialCompressor
DesignShaftPower
power1 : Power
Unit: PowerUnit = Kilowatt
Value: Double = 400.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="axialCompressor1"
    ComponentClass="AxialCompressor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS417239" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignShaftPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignShaftPower"
      Format="double"
      Value="400.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.10.4 Impellers

Attribute (composition)

The impellers of the *AxialCompressor*.

Multiplicity: *

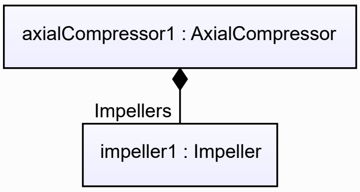
Type: *Impeller*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (an *Impeller*) is a child of the `<Equipment>` element for the attribute owner (an *AxialCompressor*).

Example



Example: Implementation in Proteus Schema

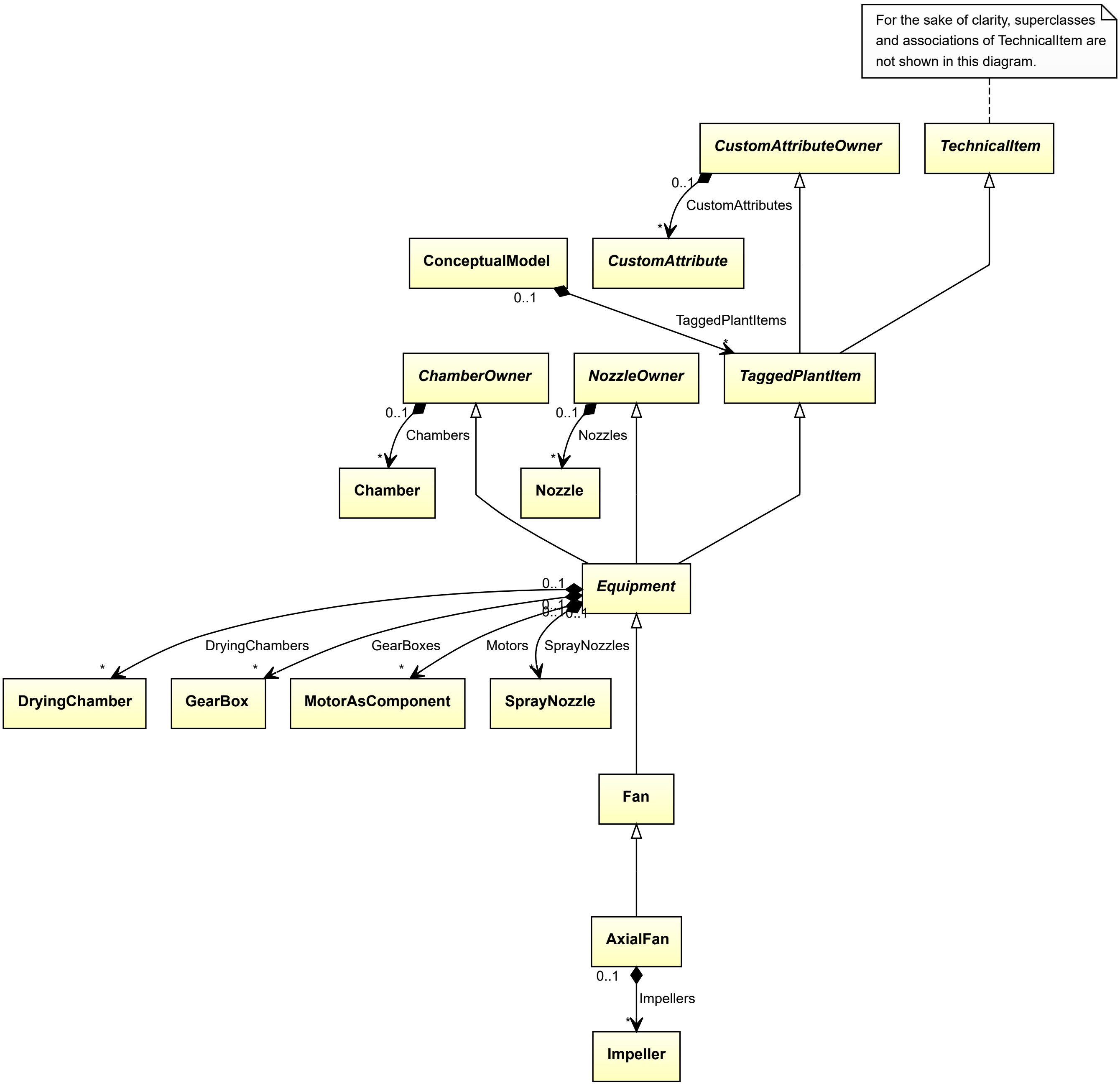
```
<Equipment
    ID="axialCompressor1"
    ComponentClass="AxialCompressor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS417239" ...>
  ...
  <Equipment
      ID="impeller1"
      ComponentClass="Impeller"
      ComponentClassURI="http://data.posccaesar.org/rdl/RDS414539" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.11. AxialFan

7.11.1 Overview

Class

A fan where the flow is along axis of shaft and the pressure ratio is relatively low (from http://data.posccaesar.org/rdl/RDS414044).



Supertypes

- *Fan*

Attributes (composition)

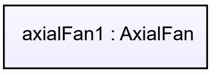
Name	Multiplicity	Type
Impellers	*	*Impeller*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`

RDL reference: AXIAL FAN
ComponentClass: AxialFan
ComponentClassURI: http://data.posccaesar.org/rdl/RDS414044

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="axialFan1"
    ComponentClass="AxialFan"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS414044" ...>
  ...
</Equipment>
```

7.11.2 Impellers

Attribute (composition)

The impellers of the *AxialFan*.

Multiplicity: *

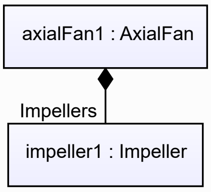
Type: *Impeller*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (an *Impeller*) is a child of the `<Equipment>` element for the attribute owner (an *AxialFan*).

Example



Example: Implementation in Proteus Schema

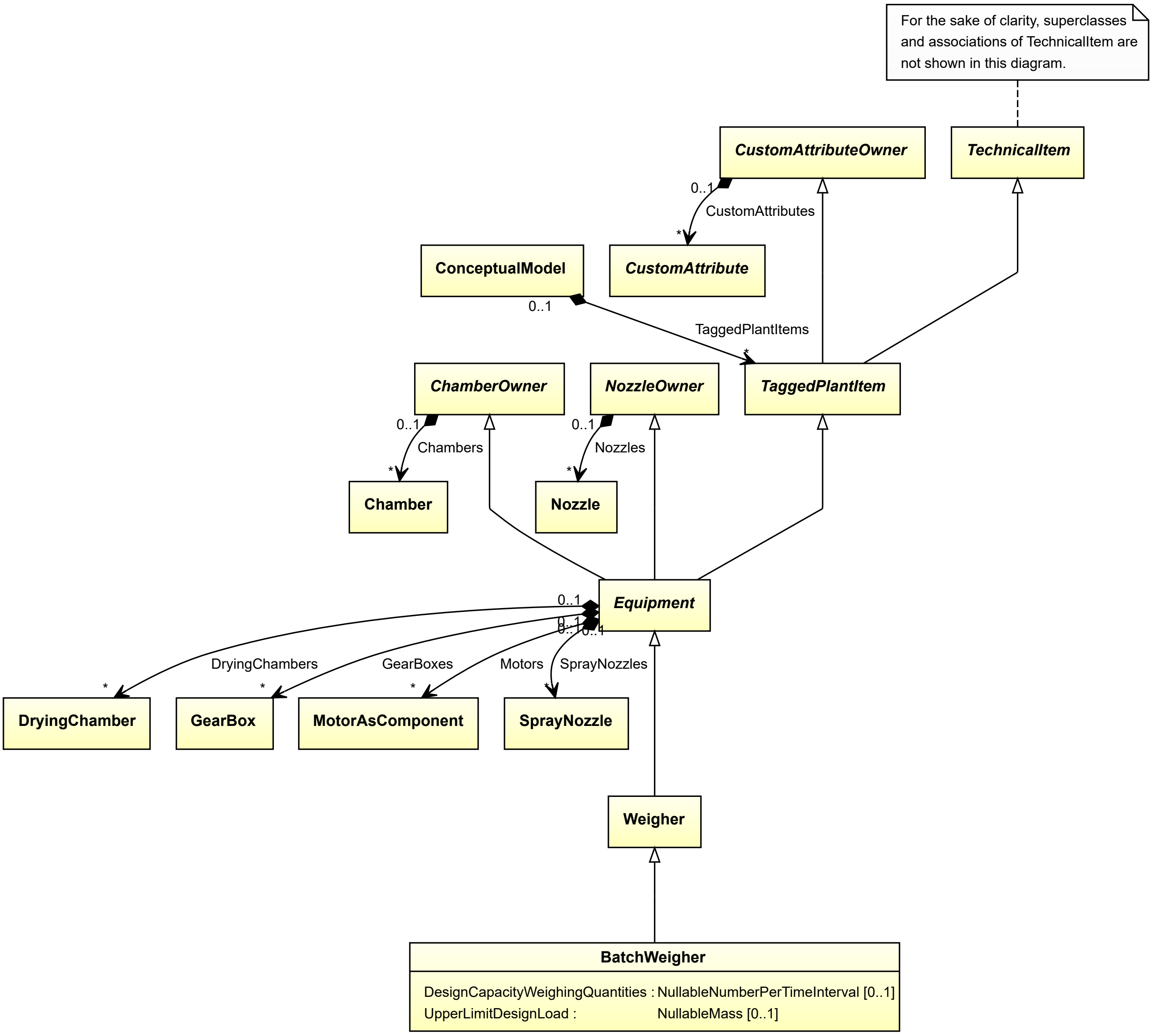
```
<Equipment
    ID="axialFan1"
    ComponentClass="AxialFan"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS414044" ...>
  ...
  <Equipment
      ID="impeller1"
      ComponentClass="Impeller"
      ComponentClassURI="http://data.posccaesar.org/rdl/RDS414539" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.12. BatchWeigher

7.12.1 Overview

Class

A *Weigher* that is operating in batch mode.



Supertypes

- *Weigher*

Attributes (data)

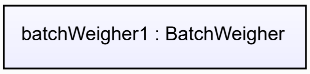
Name	Multiplicity	Type
DesignCapacityWeighingQuantities	0..1	*NullableNumberPerTimeInterval*
UpperLimitDesignLoad	0..1	*NullableMass*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: BATCH WEIGHER
ComponentClass: BatchWeigher

ComponentClassURI: http://sandbox.dexpi.org/rdl/BatchWeigher

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="batchWeigher1"
    ComponentClass="BatchWeigher"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/BatchWeigher" ...>
  ...
</Equipment>
```

7.12.2 DesignCapacityWeighingQuantities

Attribute (data)

The capacity for the number of weighing quantities per time for which the *BatchWeigher* is designed.

Multiplicity: 0..1

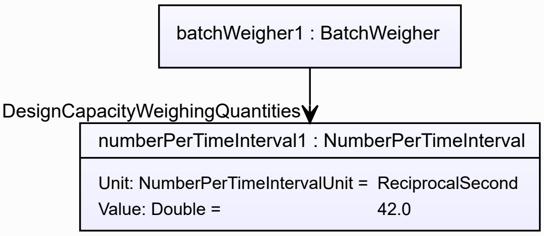
Type: *NullableNumberPerTimeInterval*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN CAPACITY WEIGHING QUANTITIES
Name: DesignCapacityWeighingQuantities
AttributeURI: http://sandbox.dexpi.org/rdl/DesignCapacityWeighingQuantities

Example

The instance batchWeigher1 represents a *BatchWeigher* with a *DesignCapacityWeighingQuantities* of 42.0 s^{-1}.


batchWeigher1 : BatchWeigher
DesignCapacityWeighingQuantities
numberPerTimeInterval1 : NumberPerTimeInterval
Unit: NumberPerTimeIntervalUnit = ReciprocalSecond
Value: Double = 42.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="batchWeigher1"
    ComponentClass="BatchWeigher"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/BatchWeigher" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignCapacityWeighingQuantities"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignCapacityWeighingQuantities"
      Format="double"
      Value="42.0"
      Units="ReciprocalSecond"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1355489" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.12.3 UpperLimitDesignLoad

Attribute (data)

The upper limit for the load for which the *BatchWeigher* is designed.

Multiplicity: 0..1

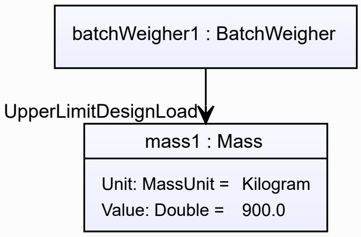
Type: *NullableMass*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT DESIGN LOAD
Name: UpperLimitDesignLoad
AttributeURI: http://sandbox.dexpi.org/rdl/UpperLimitDesignLoad

Example

The instance batchWeigher1 represents a *BatchWeigher* with an *UpperLimitDesignLoad* of 900.0 kg.



Example: Implementation in Proteus Schema

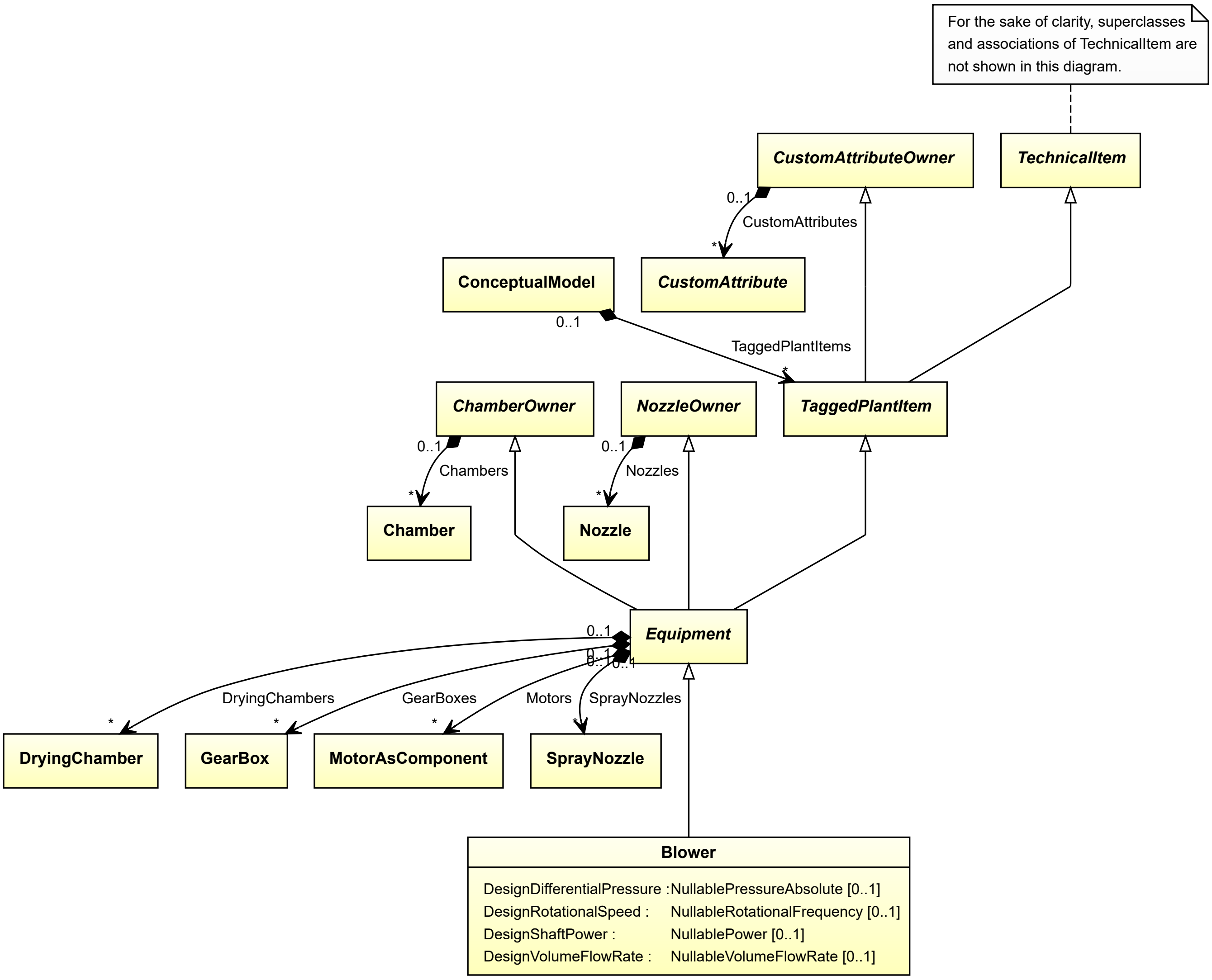
```
<Equipment
    ID="batchWeigher1"
    ComponentClass="BatchWeigher"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/BatchWeigher" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="UpperLimitDesignLoad"
      AttributeURI="http://sandbox.dexpi.org/rdl/UpperLimitDesignLoad"
      Format="double"
      Value="900.0"
      Units="Kilogram"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1328669" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.13. Blower

7.13.1 Overview

Class

A machine that is capable of blowing a medium volume flow.


For the sake of clarity, superclasses and associations of TechnicalItem are not shown in this diagram.
CustomAttributeOwner
TechnicalItem
0..1
CustomAttributes
*
ConceptualModel
CustomAttribute
0..1
TaggedPlantItems
*
ChamberOwner
NozzleOwner
TaggedPlantItem
0..1
Chambers
*
0..1
Nozzles
*
Chamber
Nozzle
Equipment
0..1
0..1
0..1
0..1
DryingChambers
GearBoxes
Motors
SprayNozzles
*
*
*
*
DryingChamber
GearBox
MotorAsComponent
SprayNozzle
Blower
DesignDifferentialPressure :NullablePressureAbsolute [0..1]
DesignRotationalSpeed : NullableRotationalFrequency [0..1]
DesignShaftPower : NullablePower [0..1]
DesignVolumeFlowRate : NullableVolumeFlowRate [0..1]

Supertypes

- *Equipment*

Subtypes

- *AxialBlower*
- *CentrifugalBlower*
- *CustomBlower*

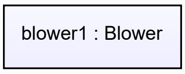
Attributes (data)

Name	Multiplicity	Type
DesignDifferentialPressure	0..1	*NullablePressureAbsolute*
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*
DesignShaftPower	0..1	*NullablePower*
DesignVolumeFlowRate	0..1	*NullableVolumeFlowRate*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: BLOWER FAN
ComponentClass: BlowerFan
ComponentClassURI: http://sandbox.dexpi.org/rdl/BlowerFan

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="blower1"
    ComponentClass="BlowerFan"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/BlowerFan" ...>
  ...
</Equipment>
```

7.13.2 DesignDifferentialPressure

Attribute (data)

The differential pressure for which the *Blower* is designed.

Multiplicity: 0..1

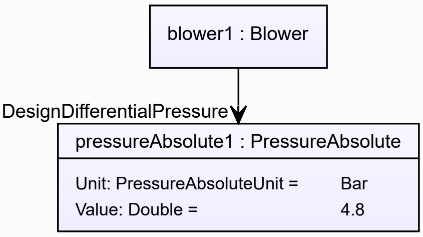
Type: *NullablePressureAbsolute*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN DIFFERENTIAL PRESSURE
Name: DesignDifferentialPressure
AttributeURI: http://sandbox.dexpi.org/rdl/DesignDifferentialPressure

Example

The instance blower1 represents a *Blower* with a *DesignDifferentialPressure* of 4.8 bar.


blower1 : Blower
DesignDifferentialPressure
pressureAbsolute1 : PressureAbsolute
Unit: PressureAbsoluteUnit = Bar
Value: Double = 4.8

Example: Implementation in Proteus Schema

```
<Equipment
    ID="blower1"
    ComponentClass="BlowerFan"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/BlowerFan" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignDifferentialPressure"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignDifferentialPressure"
      Format="double"
      Value="4.8"
      Units="Bar"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1314539" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.13.3 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *Blower* is designed.

Multiplicity: 0..1

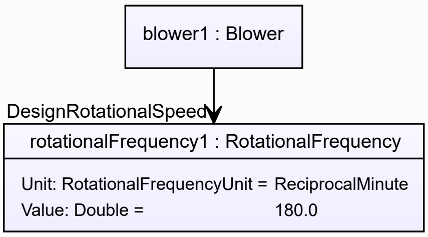
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

Example

The instance blower1 represents a *Blower* with a *DesignRotationalSpeed* of 180.0 min^{-1}.


blower1 : Blower
DesignRotationalSpeed
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 180.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="blower1"
    ComponentClass="BlowerFan"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/BlowerFan" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignRotationalSpeed"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignRotationalSpeed"
      Format="double"
      Value="180.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.13.4 DesignShaftPower

Attribute (data)

The shaft power for which the *Blower* is designed.

Multiplicity: 0..1

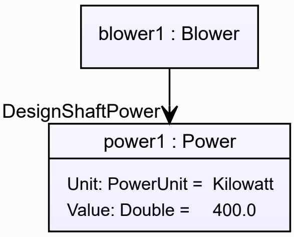
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SHAFT POWER
Name: DesignShaftPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignShaftPower

Example

The instance blower1 represents a *Blower* with a *DesignShaftPower* of 400.0 kW.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="blower1"
    ComponentClass="BlowerFan"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/BlowerFan" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignShaftPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignShaftPower"
      Format="double"
      Value="400.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.13.5 DesignVolumeFlowRate

Attribute (data)

The volume flow rate for which the *Blower* is designed.

Multiplicity: 0..1

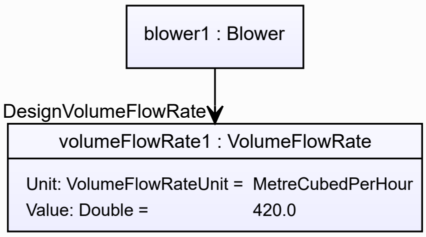
Type: *NullableVolumeFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN VOLUME FLOW RATE
Name: DesignVolumeFlowRate
AttributeURI: http://data.posccaesar.org/rdl/RDS14286227

Example

The instance blower1 represents a *Blower* with a *DesignVolumeFlowRate* of 420.0 m^3/h.


blower1 : Blower
DesignVolumeFlowRate
volumeFlowRate1 : VolumeFlowRate
Unit: VolumeFlowRateUnit = MetreCubedPerHour
Value: Double = 420.0

Example: Implementation in Proteus Schema

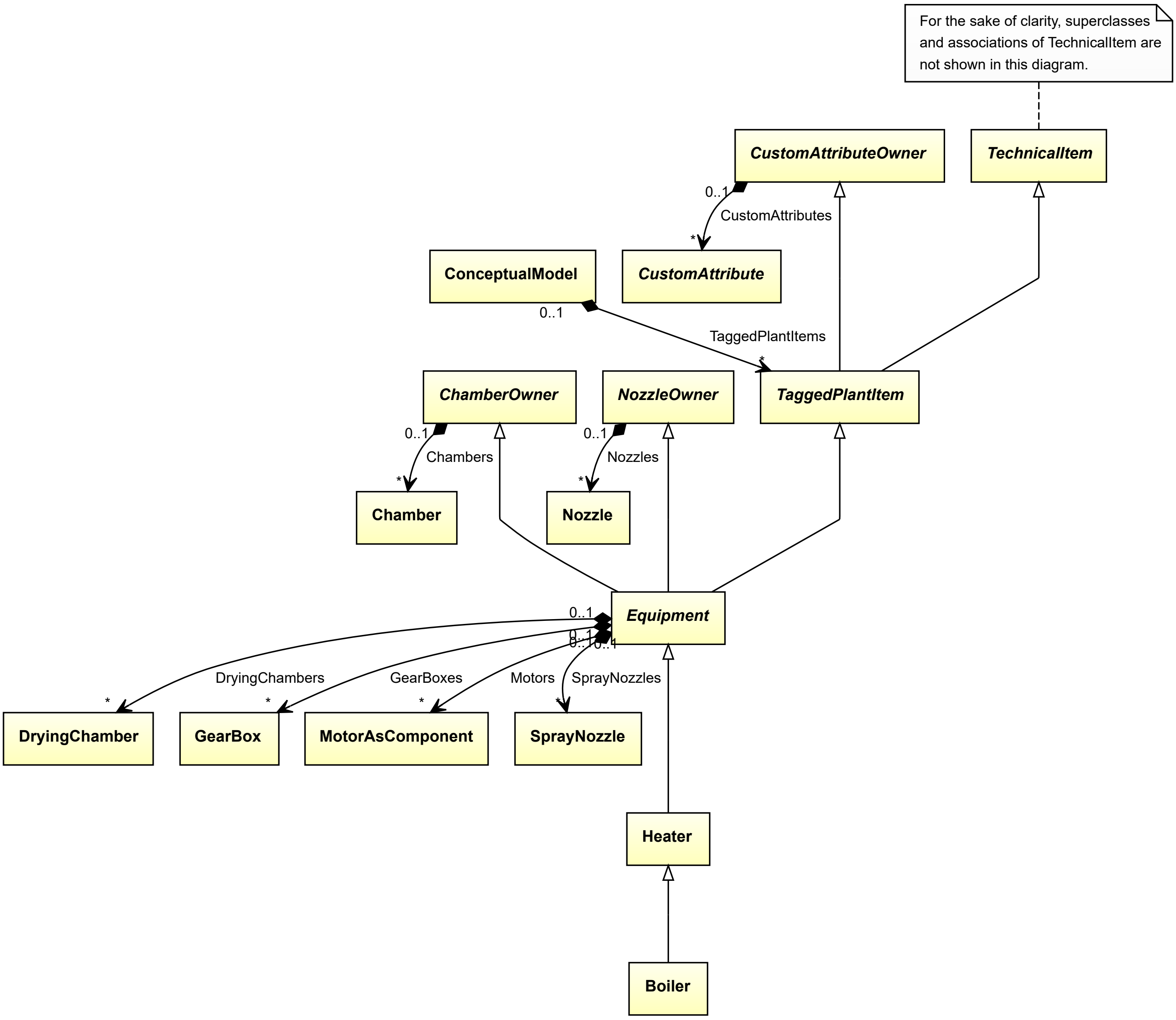
```
<Equipment
    ID="blower1"
    ComponentClass="BlowerFan"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/BlowerFan" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignVolumeFlowRate"
      AttributeURI="http://data.posccaesar.org/rdl/RDS14286227"
      Format="double"
      Value="420.0"
      Units="MetreCubedPerHour"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1321064" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.14. Boiler

7.14.1 Overview

Class

A *Heater* that brings a liquid to its boiling point.



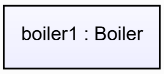
Supertypes

- *Heater*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: BOILER
ComponentClass: Boiler
ComponentClassURI: http://data.posccaesar.org/rdl/RDS14058190

Example



Example: Implementation in Proteus Schema

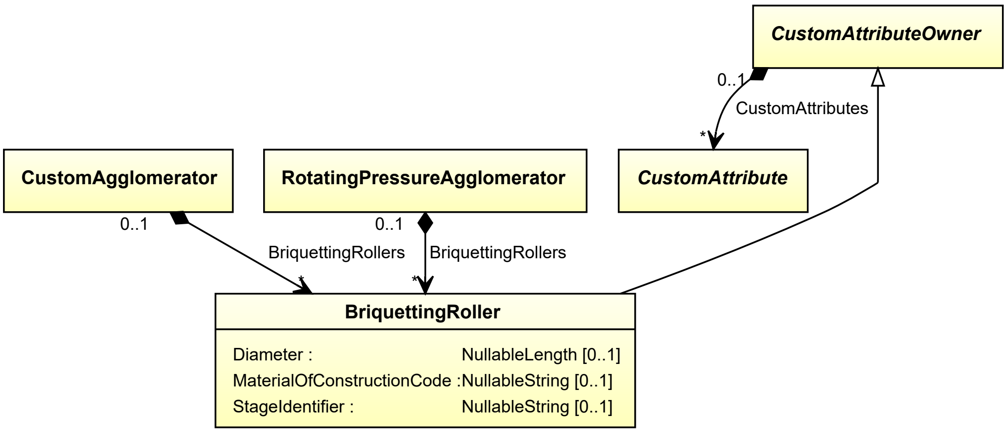
```
<Equipment
    ID="boiler1"
    ComponentClass="Boiler"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS14058190" ...>
  ...
</Equipment>
```

7.15. BriquettingRoller

7.15.1 Overview

Class

An element of an *Agglomerator* that compresses bulk material into briquettes.



CustomAttributeOwner
0..1
CustomAttributes
*
CustomAgglomerator
RotatingPressureAgglomerator
CustomAttribute
0..1
0..1
BriquettingRollers
BriquettingRollers
*
*
BriquettingRoller
Diameter : NullableLength [0..1]
MaterialOfConstructionCode : NullableString [0..1]
StageIdentifier : NullableString [0..1]

Supertypes

- *CustomAttributeOwner*

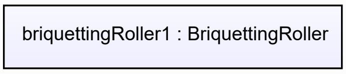
Attributes (data)

Name	Multiplicity	Type
Diameter	0..1	*NullableLength*
MaterialOfConstructionCode	0..1	*NullableString*
StageIdentifier	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: BRIQUETTING ROLLER
ComponentClass: BriquettingRoller
ComponentClassURI: http://sandbox.dexpi.org/rdl/BriquettingRoller

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="briquettingRoller1"
    ComponentClass="BriquettingRoller"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/BriquettingRoller" ...>
  ...
</Equipment>
```

7.15.2 Diameter

Attribute (data)

The diameter of the *BriquettingRoller*.

Multiplicity: 0..1

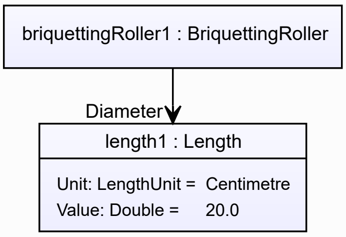
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DIAMETER
Name: Diameter
AttributeURI: http://data.posccaesar.org/rdl/RDS350954

Example

The instance briquettingRoller1 represents a *BriquettingRoller* with a *Diameter* of 20.0 cm.


briquettingRoller1 : BriquettingRoller
Diameter
length1 : Length
Unit: LengthUnit = Centimetre
Value: Double = 20.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="briquettingRoller1"
    ComponentClass="BriquettingRoller"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/BriquettingRoller" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="Diameter"
      AttributeURI="http://data.posccaesar.org/rdl/RDS350954"
      Format="double"
      Value="20.0"
      Units="Centimetre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1318004" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.15.3 MaterialOfConstructionCode

Attribute (data)

A code that gives the material of construction of the *BriquettingRoller*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: MATERIAL OF CONSTRUCTION CODE ASSIGNMENT CLASS
Name: MaterialOfConstructionCodeAssignmentClass
AttributeURI: http://data.posccaesar.org/rdl/RDS1460719741

Example

“1.4306” (*String*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="briquettingRoller1"
    ComponentClass="BriquettingRoller"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/BriquettingRoller" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="MaterialOfConstructionCodeAssignmentClass"
      AttributeURI="http://data.posccaesar.org/rdl/RDS1460719741"
      Format="string"
      Value="1.4306" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.15.4 StageIdentifier

Attribute (data)

The stage identifier of the *BriquettingRoller*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: STAGE IDENTIFIER ASSIGNMENT CLASS
Name: StageIdentifierAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/StageIdentifierAssignmentClass

Example

"s1" (*String*)

Example: Implementation in Proteus Schema

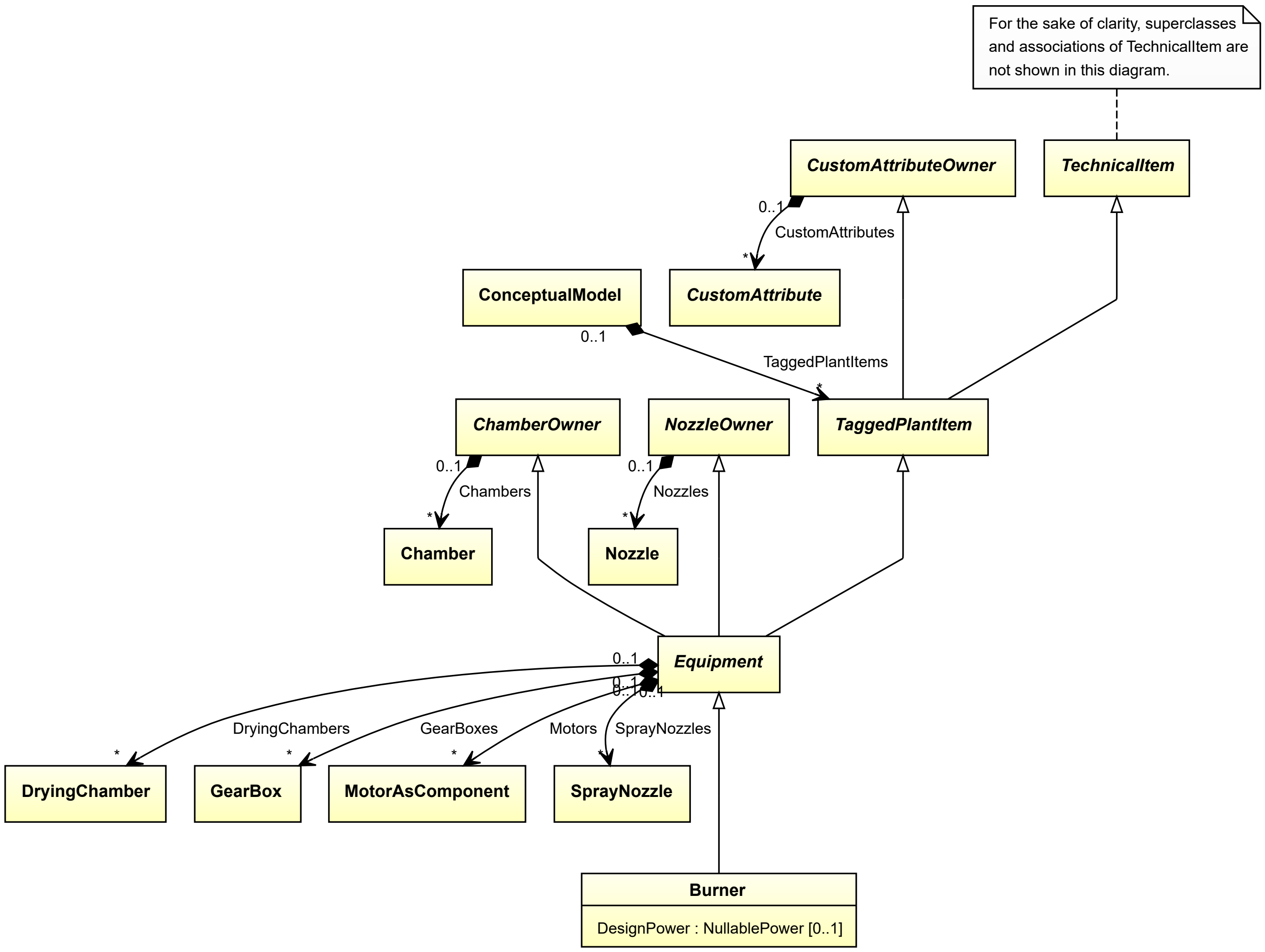
```
<Equipment
    ID="briquettingRoller1"
    ComponentClass="BriquettingRoller"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/BriquettingRoller" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="StageIdentifierAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/StageIdentifierAssignmentClass"
      Format="string"
      Value="s1" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.16. Burner

7.16.1 Overview

Class

A physical object that is intended to release thermal energy by burning a combustible mixture (from http://data.posccaesar.org/rdl/RDS284399).



Supertypes

- *Equipment*

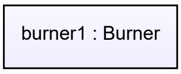
Attributes (data)

Name	Multiplicity	Type
DesignPower	0..1	*NullablePower*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: BURNER
ComponentClass: Burner
ComponentClassURI: http://data.posccaesar.org/rdl/RDS284399

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="burner1"
    ComponentClass="Burner"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS284399" ...>
  ...
</Equipment>
```

7.16.2 DesignPower

Attribute (data)

The power for which the *Burner* is designed.

Multiplicity: 0..1

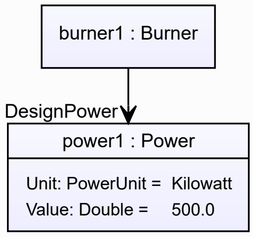
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN POWER
Name: DesignPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignPower

Example

The instance burner1 represents a *Burner* with a *DesignPower* of 500.0 kW.



Example: Implementation in Proteus Schema

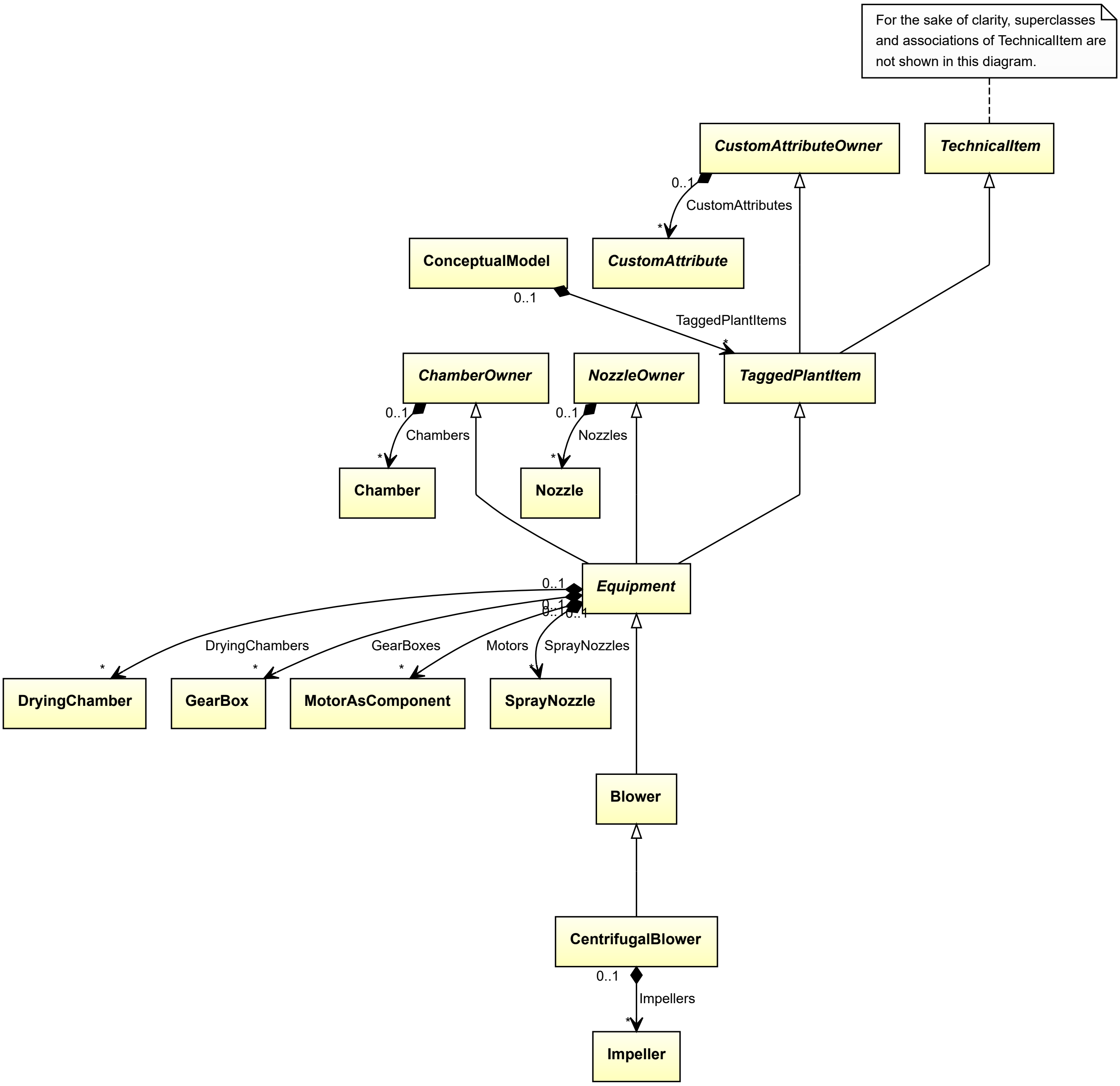
```
<Equipment
    ID="burner1"
    ComponentClass="Burner"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS284399" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignPower"
      Format="double"
      Value="500.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.17. CentrifugalBlower

7.17.1 Overview

Class

A blower in which one ore more impellers accelerate the flow and where the main flow through the impeller is radial.



Supertypes

- *Blower*

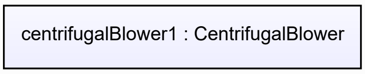
Attributes (composition)

Name	Multiplicity	Type
Impellers	*	*Impeller*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: CENTRIFUGAL BLOWER
ComponentClass: CentrifugalBlower
ComponentClassURI: http://data.posccaesar.org/rdl/RDS16766514

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="centrifugalBlower1"
    ComponentClass="CentrifugalBlower"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS16766514" ...>
  ...
</Equipment>
```

7.17.2 Impellers

Attribute (composition)

The impellers of the *CentrifugalBlower*.

Multiplicity: *

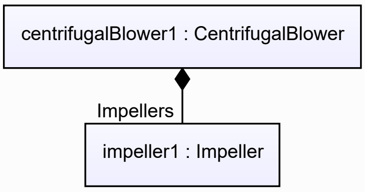
Type: *Impeller*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (an *Impeller*) is a child of the `<Equipment>` element for the attribute owner (a *CentrifugalBlower*).

Example



Example: Implementation in Proteus Schema

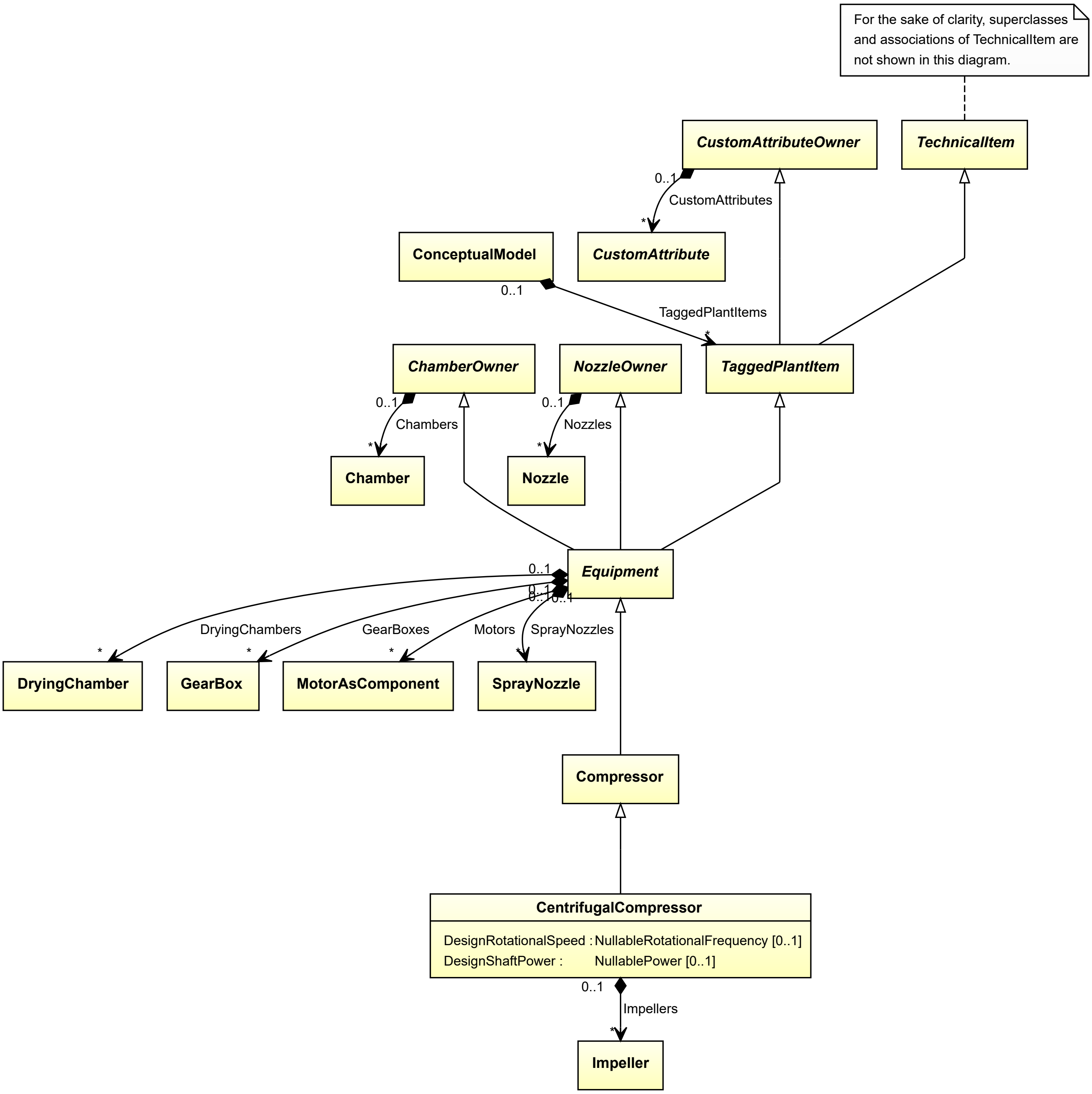
```
<Equipment
    ID="centrifugalBlower1"
    ComponentClass="CentrifugalBlower"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS16766514" ...>
  ...
  <Equipment
      ID="impeller1"
      ComponentClass="Impeller"
      ComponentClassURI="http://data.posccaesar.org/rdl/RDS414539" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.18. CentrifugalCompressor

7.18.1 Overview

Class

A dynamic compressor in which one ore more impellers accelerate the gas and where the main flow through the impeller is radial (from http://data.posccaesar.org/rdl/RDS417194).



Supertypes

- *Compressor*

Attributes (data)

Name	Multiplicity	Type
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*
DesignShaftPower	0..1	*NullablePower*

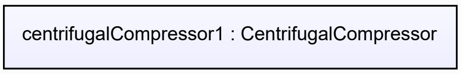
Attributes (composition)

Name	Multiplicity	Type
Impellers	*	*Impeller*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: CENTRIFUGAL COMPRESSOR
ComponentClass: CentrifugalCompressor
ComponentClassURI: http://data.posccaesar.org/rdl/RDS417194

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="centrifugalCompressor1"
    ComponentClass="CentrifugalCompressor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS417194" ...>
  ...
</Equipment>
```

7.18.2 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *CentrifugalCompressor* is designed.

Multiplicity: 0..1

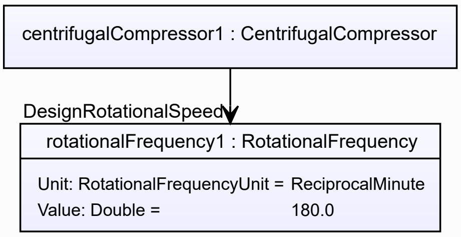
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

Example

The instance centrifugalCompressor1 represents a *CentrifugalCompressor* with a *DesignRotationalSpeed* of 180.0 min^{-1}.


centrifugalCompressor1 : CentrifugalCompressor
DesignRotationalSpeed
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 180.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="centrifugalCompressor1"
    ComponentClass="CentrifugalCompressor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS417194" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignRotationalSpeed"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignRotationalSpeed"
      Format="double"
      Value="180.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.18.3 DesignShaftPower

Attribute (data)

The shaft power for which the *CentrifugalCompressor* is designed.

Multiplicity: 0..1

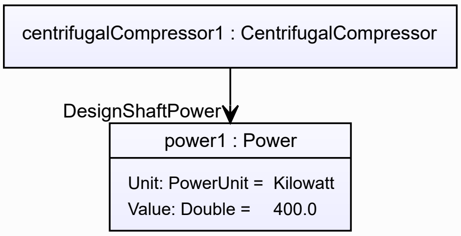
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SHAFT POWER
Name: DesignShaftPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignShaftPower

Example

The instance centrifugalCompressor1 represents a *CentrifugalCompressor* with a *DesignShaftPower* of 400.0 kW.


centrifugalCompressor1 : CentrifugalCompressor
DesignShaftPower
power1 : Power
Unit: PowerUnit = Kilowatt
Value: Double = 400.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="centrifugalCompressor1"
    ComponentClass="CentrifugalCompressor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS417194" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignShaftPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignShaftPower"
      Format="double"
      Value="400.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.18.4 Impellers

Attribute (composition)

The impellers of the *CentrifugalCompressor*.

Multiplicity: *

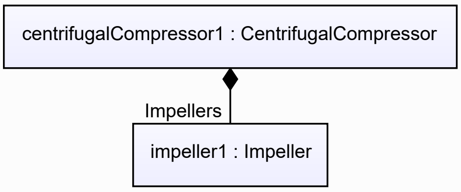
Type: *Impeller*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (an *Impeller*) is a child of the `<Equipment>` element for the attribute owner (a *CentrifugalCompressor*).

Example



Example: Implementation in Proteus Schema

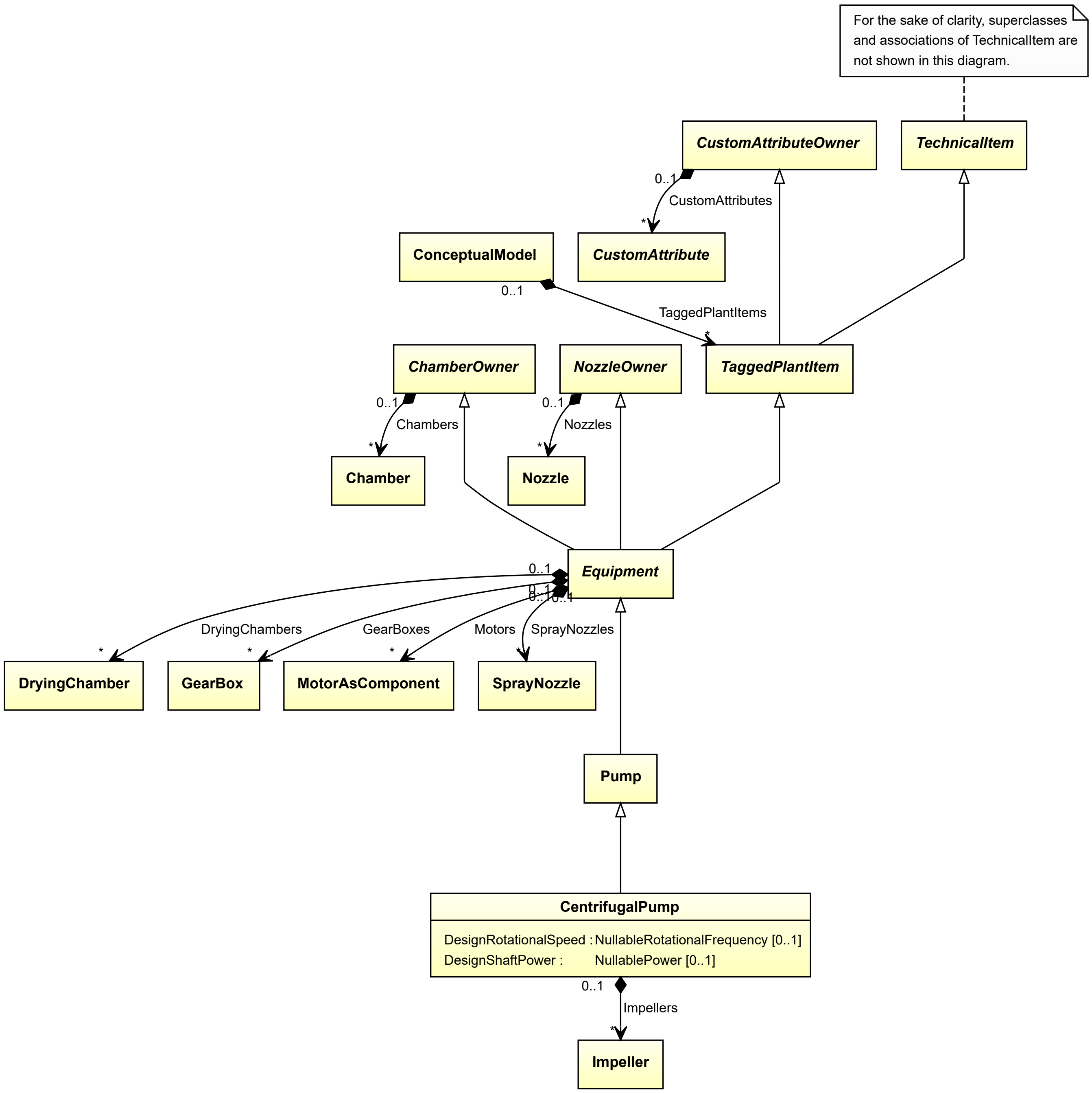
```
<Equipment
    ID="centrifugalCompressor1"
    ComponentClass="CentrifugalCompressor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS417194" ...>
  ...
  <Equipment
      ID="impeller1"
      ComponentClass="Impeller"
      ComponentClassURI="http://data.posccaesar.org/rdl/RDS414539" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.19. CentrifugalPump

7.19.1 Overview

Class

A dynamic pump utilizing impellers provided with vanes generating centrifugal force to achieve the required pressure head (from http://data.posccaesar.org/rdl/RDS416834).



Supertypes

- *Pump*

Attributes (data)

Name	Multiplicity	Type
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*
DesignShaftPower	0..1	*NullablePower*

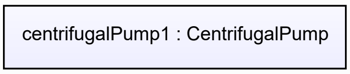
Attributes (composition)

Name	Multiplicity	Type
Impellers	*	*Impeller*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: CENTRIFUGAL PUMP
ComponentClass: CentrifugalPump
ComponentClassURI: http://data.posccaesar.org/rdl/RDS416834

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="centrifugalPump1"
    ComponentClass="CentrifugalPump"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS416834" ...>
  ...
</Equipment>
```

7.19.2 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *CentrifugalPump* is designed.

Multiplicity: 0..1

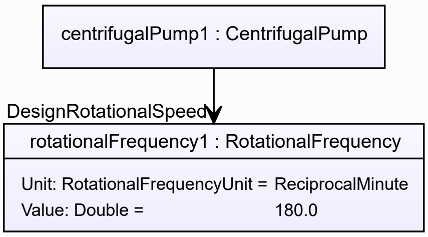
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

Example

The instance centrifugalPump1 represents a *CentrifugalPump* with a *DesignRotationalSpeed* of 180.0 min^{-1}.


centrifugalPump1 : CentrifugalPump
DesignRotationalSpeed
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 180.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="centrifugalPump1"
    ComponentClass="CentrifugalPump"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS416834" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignRotationalSpeed"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignRotationalSpeed"
      Format="double"
      Value="180.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.19.3 DesignShaftPower

Attribute (data)

The shaft power for which the *CentrifugalPump* is designed.

Multiplicity: 0..1

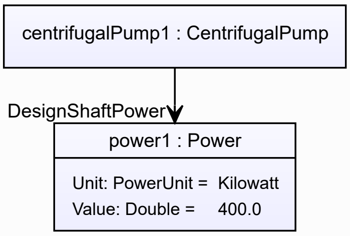
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SHAFT POWER
Name: DesignShaftPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignShaftPower

Example

The instance centrifugalPump1 represents a *CentrifugalPump* with a *DesignShaftPower* of 400.0 kW.


centrifugalPump1 : CentrifugalPump
DesignShaftPower
power1 : Power
Unit: PowerUnit = Kilowatt
Value: Double = 400.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="centrifugalPump1"
    ComponentClass="CentrifugalPump"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS416834" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignShaftPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignShaftPower"
      Format="double"
      Value="400.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.19.4 Impellers

Attribute (composition)

The impellers of the *CentrifugalPump*.

Multiplicity: *

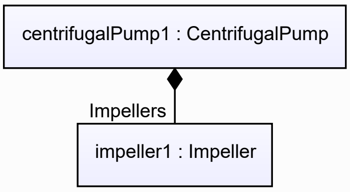
Type: *Impeller*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (an *Impeller*) is a child of the `<Equipment>` element for the attribute owner (a *CentrifugalPump*).

Example



Example: Implementation in Proteus Schema

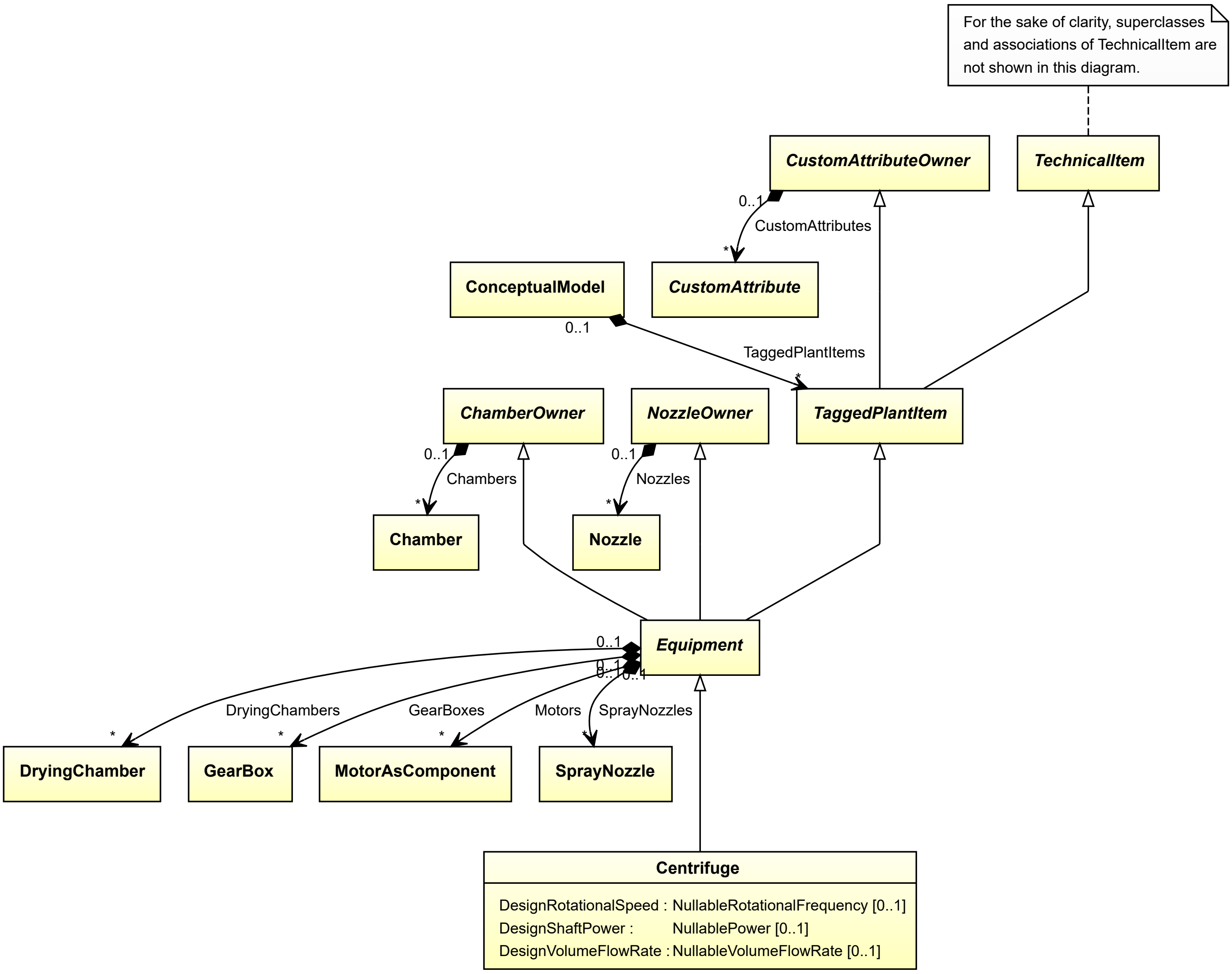
```
<Equipment
    ID="centrifugalPump1"
    ComponentClass="CentrifugalPump"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS416834" ...>
  ...
  <Equipment
      ID="impeller1"
      ComponentClass="Impeller"
      ComponentClassURI="http://data.posccaesar.org/rdl/RDS414539" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.20. Centrifuge

7.20.1 Overview

Class

A ‘separator’ and ‘machine’ that uses centrifugal force to separate phases of different densities (from http://data.posccaesar.org/rdl/RDS420974).


For the sake of clarity, superclasses and associations of TechnicalItem are not shown in this diagram.

Supertypes

- *Equipment*

Subtypes

- *CustomCentrifuge*
- *FilteringCentrifuge*
- *SedimentalCentrifuge*

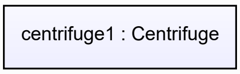
Attributes (data)

Name	Multiplicity	Type
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*
DesignShaftPower	0..1	*NullablePower*
DesignVolumeFlowRate	0..1	*NullableVolumeFlowRate*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: CENTRIFUGE
ComponentClass: Centrifuge
ComponentClassURI: http://data.posccaesar.org/rdl/RDS420974

Example



centrifuge1 : Centrifuge

Example: Implementation in Proteus Schema

```
<Equipment
    ID="centrifuge1"
    ComponentClass="Centrifuge"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS420974" ...>
  ...
</Equipment>
```

7.20.2 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *Centrifuge* is designed.

Multiplicity: 0..1

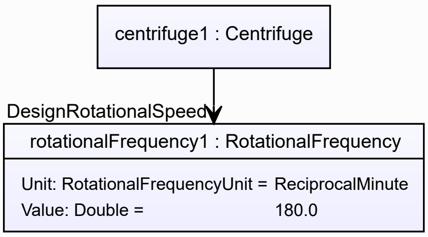
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

Example

The instance centrifuge1 represents a *Centrifuge* with a *DesignRotationalSpeed* of 180.0 min^{-1}.


centrifuge1 : Centrifuge
DesignRotationalSpeed
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 180.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="centrifuge1"
    ComponentClass="Centrifuge"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS420974" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignRotationalSpeed"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignRotationalSpeed"
      Format="double"
      Value="180.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.20.3 DesignShaftPower

Attribute (data)

The shaft power for which the *Centrifuge* is designed.

Multiplicity: 0..1

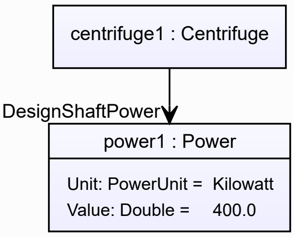
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SHAFT POWER
Name: DesignShaftPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignShaftPower

Example

The instance centrifuge1 represents a *Centrifuge* with a *DesignShaftPower* of 400.0 kW.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="centrifuge1"
    ComponentClass="Centrifuge"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS420974" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignShaftPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignShaftPower"
      Format="double"
      Value="400.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.20.4 DesignVolumeFlowRate

Attribute (data)

The volume flow rate for which the *Centrifuge* is designed.

Multiplicity: 0..1

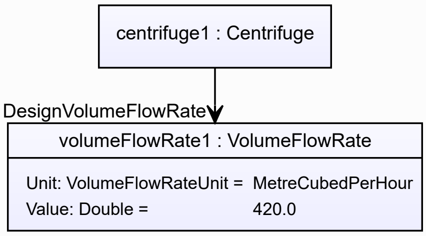
Type: *NullableVolumeFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN VOLUME FLOW RATE
Name: DesignVolumeFlowRate
AttributeURI: http://data.posccaesar.org/rdl/RDS14286227

Example

The instance centrifuge1 represents a *Centrifuge* with a *DesignVolumeFlowRate* of 420.0 m^3/h.


centrifuge1 : Centrifuge
DesignVolumeFlowRate
volumeFlowRate1 : VolumeFlowRate
Unit: VolumeFlowRateUnit = MetreCubedPerHour
Value: Double = 420.0

Example: Implementation in Proteus Schema

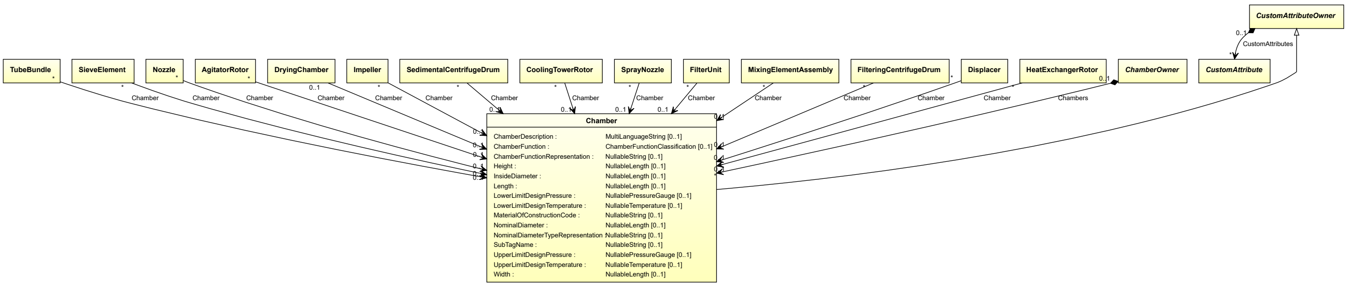
```
<Equipment
    ID="centrifuge1"
    ComponentClass="Centrifuge"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS420974" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignVolumeFlowRate"
      AttributeURI="http://data.posccaesar.org/rdl/RDS14286227"
      Format="double"
      Value="420.0"
      Units="MetreCubedPerHour"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1321064" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.21. Chamber

7.21.1 Overview

Class

A physical object that is an enclosed space (from http://data.posccaesar.org/rdl/RDS903151421).


CustomAttributeOwner
0..1
CustomAttributes
*
TubeBundle
SieveElement
Nozzle
AgitatorRotor
DryingChamber
Impeller
SedimentalCentrifugeDrum
CoolingTowerRotor
SprayNozzle
FilterUnit
MixingElementAssembly
FilteringCentrifugeDrum
Displacer
HeatExchangerRotor
ChamberOwner
CustomAttribute
Chamber
Chambers
0..1
*
Chamber
ChamberDescription : MultiLanguageString [0..1]
ChamberFunction : ChamberFunctionClassification [0..1]
ChamberFunctionRepresentation : NullableString [0..1]
Height : NullableLength [0..1]
InsideDiameter : NullableLength [0..1]
Length : NullableLength [0..1]
LowerLimitDesignPressure : NullablePressureGauge [0..1]
LowerLimitDesignTemperature : NullableTemperature [0..1]
MaterialOfConstructionCode : NullableString [0..1]
NominalDiameter : NullableLength [0..1]
NominalDiameterTypeRepresentation NullableString [0..1]
SubTagName : NullableString [0..1]
UpperLimitDesignPressure : NullablePressureGauge [0..1]
UpperLimitDesignTemperature : NullableTemperature [0..1]
Width : NullableLength [0..1]

Supertypes

- *CustomAttributeOwner*

Attributes (data)

Name	Multiplicity	Type
ChamberDescription	0..1	*MultiLanguageString*
ChamberFunction	0..1	*ChamberFunctionClassification*
ChamberFunctionRepresentation	0..1	*NullableString*
Height	0..1	*NullableLength*
InsideDiameter	0..1	*NullableLength*
Length	0..1	*NullableLength*
LowerLimitDesignPressure	0..1	*NullablePressureGauge*
LowerLimitDesignTemperature	0..1	*NullableTemperature*

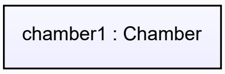
(continued on next page)

Name	Multiplicity	Type
MaterialOfConstructionCode	0..1	*NullableString*
NominalDiameter	0..1	*NullableLength*
NominalDiameterTypeRepresentation	0..1	*NullableString*
SubTagName	0..1	*NullableString*
UpperLimitDesignPressure	0..1	*NullablePressureGauge*
UpperLimitDesignTemperature	0..1	*NullableTemperature*
Width	0..1	*NullableLength*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: CHAMBER
ComponentClass: Chamber
ComponentClassURI: http://data.posccaesar.org/rdl/RDS903151421

Example


chamber1 : Chamber

Example: Implementation in Proteus Schema

```
<Equipment
    ID="chamber1"
    ComponentClass="Chamber"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS903151421" ...>
  ...
</Equipment>
```

7.21.2 ChamberDescription

Attribute (data)

The description of the *Chamber*.

Multiplicity: 0..1

Type: *MultiLanguageString*

Implementation in Proteus Schema

The attribute is implemented as a *set of DEXPI generic attributes for multi-language string values*.
RDL reference: CHAMBER DESCRIPTION ASSIGNMENT CLASS
Name: ChamberDescriptionAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ChamberDescriptionAssignmentClass

Example

Language	Value
en	jacket chamber

(*MultiLanguageString* with 1 *SingleLanguageString*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="chamber1"
    ComponentClass="Chamber"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS903151421" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ChamberDescriptionAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ChamberDescriptionAssignmentClass"
      Format="string"
      Language="en"
      Value="jacket chamber" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.21.3 ChamberFunction

Attribute (data)

A specialization indicating the function of the *Chamber*.

Multiplicity: 0..1

Type: *ChamberFunctionClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: CHAMBER FUNCTION SPECIALIZATION
Name: ChamberFunctionSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/ChamberFunctionSpecialization

Example

heating (*ChamberFunctionClassification::Heating*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="chamber1"
    ComponentClass="Chamber"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS903151421" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ChamberFunctionSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/ChamberFunctionSpecialization"
      Format="anyURI"
      Value="Heating"
      ValueURI="http://data.posccaesar.org/rdl/RDS9666872" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.21.4 ChamberFunctionRepresentation

Attribute (data)

A short textual description of the function of the *Chamber*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: CHAMBER FUNCTION ASSIGNMENT CLASS
Name: ChamberFunctionAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ChamberFunctionAssignmentClass

Example

“cooling” (*String*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="chamber1"
    ComponentClass="Chamber"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS903151421" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ChamberFunctionAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ChamberFunctionAssignmentClass"
      Format="string"
      Value="cooling" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.21.5 Height

Attribute (data)

The height of the *Chamber*.

Multiplicity: 0..1

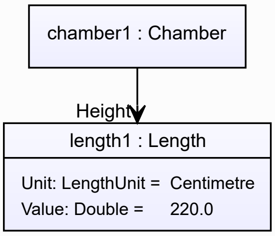
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: HEIGHT
Name: Height
AttributeURI: http://data.posccaesar.org/rdl/RDS357704

Example

The instance chamber1 represents a *Chamber* with a *Height* of 220.0 cm.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="chamber1"
    ComponentClass="Chamber"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS903151421" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="Height"
      AttributeURI="http://data.posccaesar.org/rdl/RDS357704"
      Format="double"
      Value="220.0"
      Units="Centimetre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1318004" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.21.6 InsideDiameter

Attribute (data)

The inside diameter of the *Chamber*.

Multiplicity: 0..1

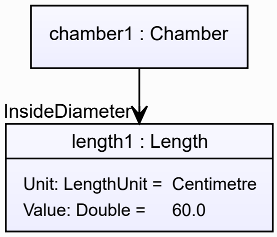
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: INSIDE DIAMETER
Name: InsideDiameter
AttributeURI: http://data.posccaesar.org/rdl/RDS357209

Example

The instance chamber1 represents a *Chamber* with an *InsideDiameter* of 60.0 cm.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="chamber1"
    ComponentClass="Chamber"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS903151421" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="InsideDiameter"
      AttributeURI="http://data.posccaesar.org/rdl/RDS357209"
      Format="double"
      Value="60.0"
      Units="Centimetre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1318004" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.21.7 Length

Attribute (data)

The length of the *Chamber*.

Multiplicity: 0..1

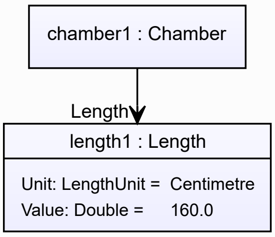
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: LENGTH
Name: Length
AttributeURI: http://data.posccaesar.org/rdl/RDS373094

Example

The instance chamber1 represents a *Chamber* with a *Length* of 160.0 cm.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="chamber1"
    ComponentClass="Chamber"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS903151421" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="Length"
      AttributeURI="http://data.posccaesar.org/rdl/RDS373094"
      Format="double"
      Value="160.0"
      Units="Centimetre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1318004" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.21.8 LowerLimitDesignPressure

Attribute (data)

The lower limit for the pressure for which the *Chamber* is designed.

Multiplicity: 0..1

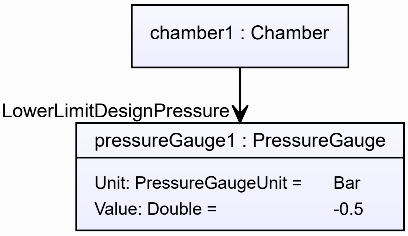
Type: *NullablePressureGauge*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: LOWER LIMIT DESIGN PRESSURE
Name: LowerLimitDesignPressure
AttributeURI: http://data.posccaesar.org/rdl/RDS360674

Example

The instance chamber1 represents a *Chamber* with a *LowerLimitDesignPressure* of -0.5 bar.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="chamber1"
    ComponentClass="Chamber"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS903151421" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="LowerLimitDesignPressure"
      AttributeURI="http://data.posccaesar.org/rdl/RDS360674"
      Format="double"
      Value="-0.5"
      Units="Bar"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1314539" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.21.9 LowerLimitDesignTemperature

Attribute (data)

The lower limit for the temperature for which the *Chamber* is designed.

Multiplicity: 0..1

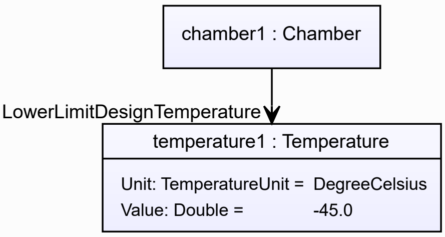
Type: *NullableTemperature*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: LOWER LIMIT DESIGN TEMPERATURE
Name: LowerLimitDesignTemperature
AttributeURI: http://data.posccaesar.org/rdl/RDS360494

Example

The instance chamber1 represents a *Chamber* with a *LowerLimitDesignTemperature* of -45.0 °C.


chamber1 : Chamber
LowerLimitDesignTemperature
temperature1 : Temperature
Unit: TemperatureUnit = DegreeCelsius
Value: Double = -45.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="chamber1"
    ComponentClass="Chamber"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS903151421" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="LowerLimitDesignTemperature"
      AttributeURI="http://data.posccaesar.org/rdl/RDS360494"
      Format="double"
      Value="-45.0"
      Units="DegreeCelsius"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1322684" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.21.10 MaterialOfConstructionCode

Attribute (data)

A code that gives the material of construction of the *Chamber*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: MATERIAL OF CONSTRUCTION CODE ASSIGNMENT CLASS
Name: MaterialOfConstructionCodeAssignmentClass
AttributeURI: http://data.posccaesar.org/rdl/RDS1460719741

Example

“1.4306” (*String*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="chamber1"
    ComponentClass="Chamber"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS903151421" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="MaterialOfConstructionCodeAssignmentClass"
      AttributeURI="http://data.posccaesar.org/rdl/RDS1460719741"
      Format="string"
      Value="1.4306" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.21.11 NominalDiameter

Attribute (data)

The nominal diameter of the *Chamber*, given as a length. See also *NominalDiameterTypeRepresentation*.

Multiplicity: 0..1

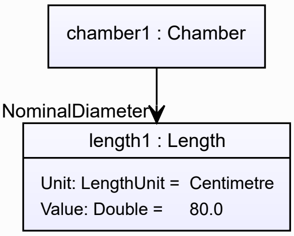
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: NOMINAL DIAMETER
Name: NominalDiameter
AttributeURI: http://data.posccaesar.org/rdl/RDS366794

Example

The instance chamber1 represents a *Chamber* with a *NominalDiameter* of 80.0 cm.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="chamber1"
    ComponentClass="Chamber"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS903151421" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NominalDiameter"
      AttributeURI="http://data.posccaesar.org/rdl/RDS366794"
      Format="double"
      Value="80.0"
      Units="Centimetre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1318004" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.21.12 NominalDiameterTypeRepresentation

Attribute (data)

A readable representation of the type or unit of measure of the nominal diameter of the *Chamber*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: NOMINAL DIAMETER TYPE REPRESENTATION ASSIGNMENT CLASS
Name: NominalDiameterTypeRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/NominalDiameterTypeRepresentationAssignmentClass

Example

“DN” (*String*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="chamber1"
    ComponentClass="Chamber"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS903151421" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NominalDiameterTypeRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/NominalDiameterTypeRepresentationAssignmentClass"
      Format="string"
      Value="DN" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.21.13 SubTagName

Attribute (data)

The sub tag name of the *Chamber*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: SUB TAG NAME ASSIGNMENT CLASS
Name: SubTagNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/SubTagNameAssignmentClass

Example

“ST1” (*String*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="chamber1"
    ComponentClass="Chamber"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS903151421" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="SubTagNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/SubTagNameAssignmentClass"
      Format="string"
      Value="ST1" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.21.14 UpperLimitDesignPressure

Attribute (data)

The upper limit for the pressure for which the *Chamber* is designed.

Multiplicity: 0..1

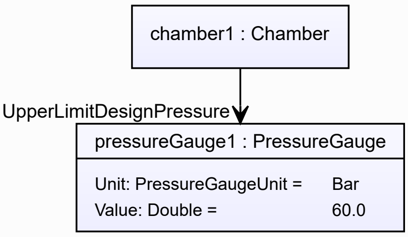
Type: *NullablePressureGauge*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT DESIGN PRESSURE
Name: UpperLimitDesignPressure
AttributeURI: http://data.posccaesar.org/rdl/RDS1470835011

Example

The instance chamber1 represents a *Chamber* with an *UpperLimitDesignPressure* of 60.0 bar.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="chamber1"
    ComponentClass="Chamber"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS903151421" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="UpperLimitDesignPressure"
      AttributeURI="http://data.posccaesar.org/rdl/RDS1470835011"
      Format="double"
      Value="60.0"
      Units="Bar"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1314539" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.21.15 UpperLimitDesignTemperature

Attribute (data)

The upper limit for the temperature for which the *Chamber* is designed.

Multiplicity: 0..1

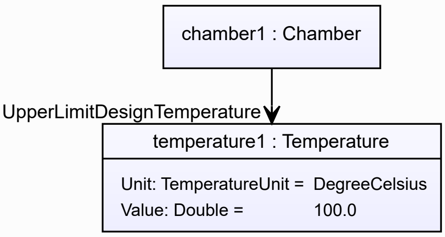
Type: *NullableTemperature*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT DESIGN TEMPERATURE
Name: UpperLimitDesignTemperature
AttributeURI: http://data.posccaesar.org/rdl/RDS360449

Example

The instance chamber1 represents a *Chamber* with an *UpperLimitDesignTemperature* of 100.0 °C.


chamber1 : Chamber
UpperLimitDesignTemperature
temperature1 : Temperature
Unit: TemperatureUnit = DegreeCelsius
Value: Double = 100.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="chamber1"
    ComponentClass="Chamber"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS903151421" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="UpperLimitDesignTemperature"
      AttributeURI="http://data.posccaesar.org/rdl/RDS360449"
      Format="double"
      Value="100.0"
      Units="DegreeCelsius"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1322684" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.21.16 Width

Attribute (data)

The width of the *Chamber*.

Multiplicity: 0..1

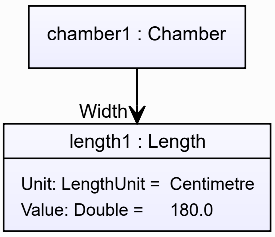
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: WIDTH
Name: Width
AttributeURI: http://data.posccaesar.org/rdl/RDS361709

Example

The instance chamber1 represents a *Chamber* with a *Width* of 180.0 cm.



Example: Implementation in Proteus Schema

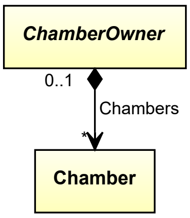
```
<Equipment
    ID="chamber1"
    ComponentClass="Chamber"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS903151421" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="Width"
      AttributeURI="http://data.posccaesar.org/rdl/RDS361709"
      Format="double"
      Value="180.0"
      Units="Centimetre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1318004" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.22. ChamberOwner

7.22.1 Overview

Abstract class

An object that can have chambers.



Subtypes

- *Equipment*

Attributes (composition)

Name	Multiplicity	Type
Chambers	*	*Chamber*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*. As *ChamberOwner* is abstract, there is no RDL reference for the class itself; the RDL reference depends on the concrete subclass.
Tag: `<Equipment>`
ComponentClass: *depending on subclass*
ComponentClassURI: *depending on subclass*

Example

As *ChamberOwner* is abstract, we consider *Vessel* as an arbitrary concrete subclass.

vessel1 : Vessel

Example: Implementation in Proteus Schema

```
<Equipment
    ID="vessel1"
    ComponentClass="Vessel"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS414674" ...>
  ...
</Equipment>
```

7.22.2 Chambers

Attribute (composition)

The Chambers of the *ChamberOwner*.

Multiplicity: *

Type: *Chamber*

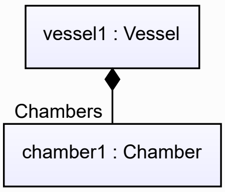
Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *Chamber*) is a child of the `<Equipment>` element for the attribute owner (a *ChamberOwner*).

Example

As the owner type *ChamberOwner* is abstract, we consider *Vessel* as an arbitrary concrete subclass.



Example: Implementation in Proteus Schema

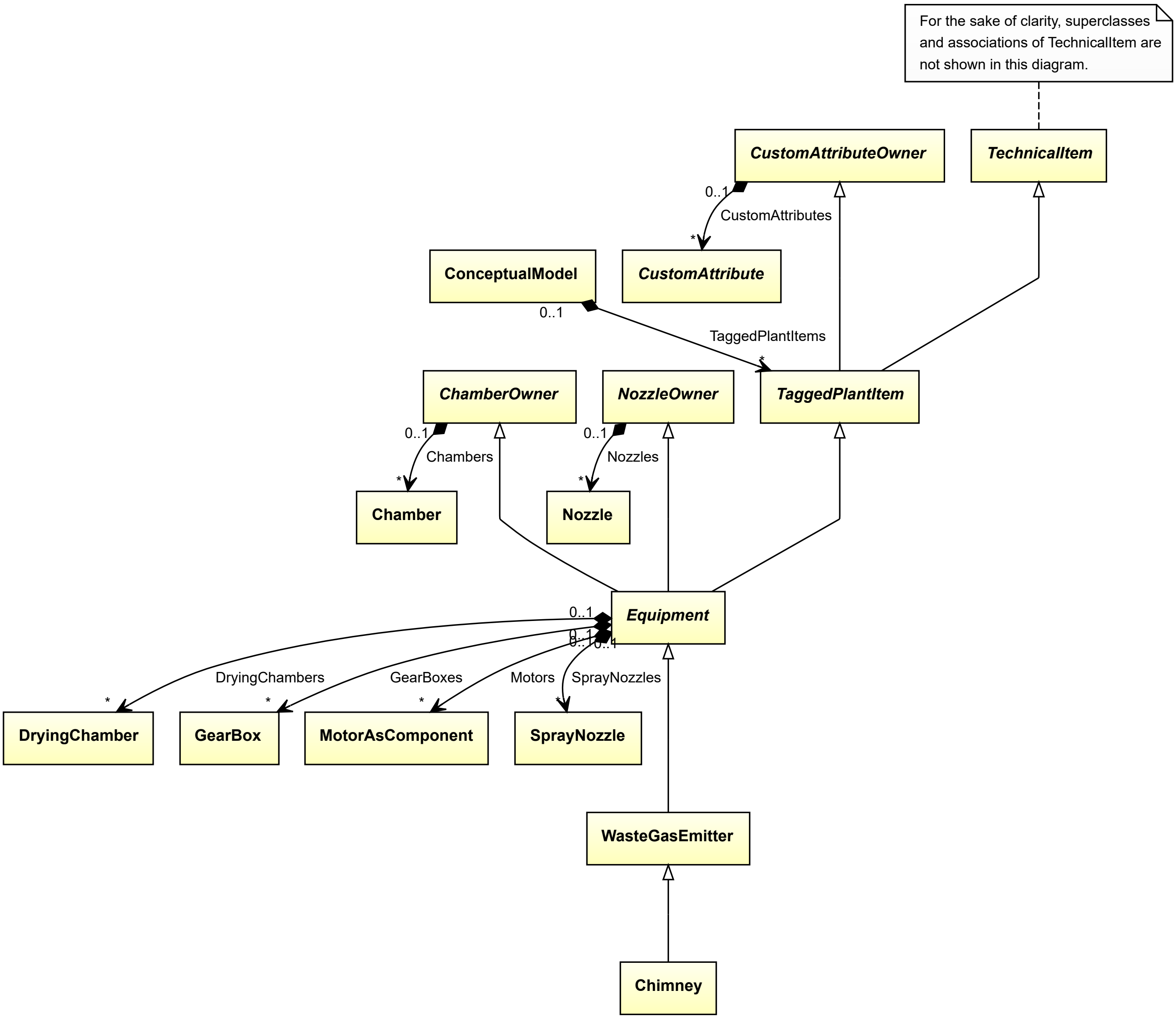
```
<Equipment
    ID="vessel1"
    ComponentClass="Vessel"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS414674" ...>
  ...
  <Equipment
      ID="chamber1"
      ComponentClass="Chamber"
      ComponentClassURI="http://data.posccaesar.org/rdl/RDS903151421" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.23. Chimney

7.23.1 Overview

Class

A *WasteGasEmitter* that is intended to transport waste gas to a high location in the atmosphere.



Supertypes

- *WasteGasEmitter*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: CHIMNEY
ComponentClass: Chimney
ComponentClassURI: http://sandbox.dexpi.org/rdl/Chimney

Example

Example: Implementation in Proteus Schema

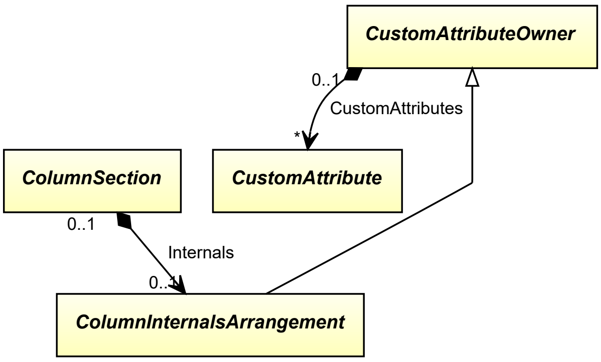
```
<Equipment
    ID="chimney1"
    ComponentClass="Chimney"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Chimney" ...>
  ...
</Equipment>
```

7.24. ColumnInternalsArrangement

7.24.1 Overview

Abstract class

The internals of a column, e.g., trays or packings.



Supertypes

- *CustomAttributeOwner*

Subtypes

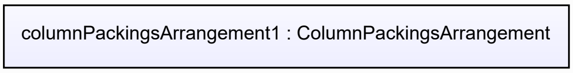
- *ColumnPackingsArrangement*
- *ColumnTraysArrangement*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*. As *ColumnInternalsArrangement* is abstract, there is no RDL reference for the class itself; the RDL reference depends on the concrete subclass.
Tag: `<Equipment>`
ComponentClass: *depending on subclass*
ComponentClassURI: *depending on subclass*

Example

As *ColumnInternalsArrangement* is abstract, we consider *ColumnPackingsArrangement* as an arbitrary concrete subclass.



Example: Implementation in Proteus Schema

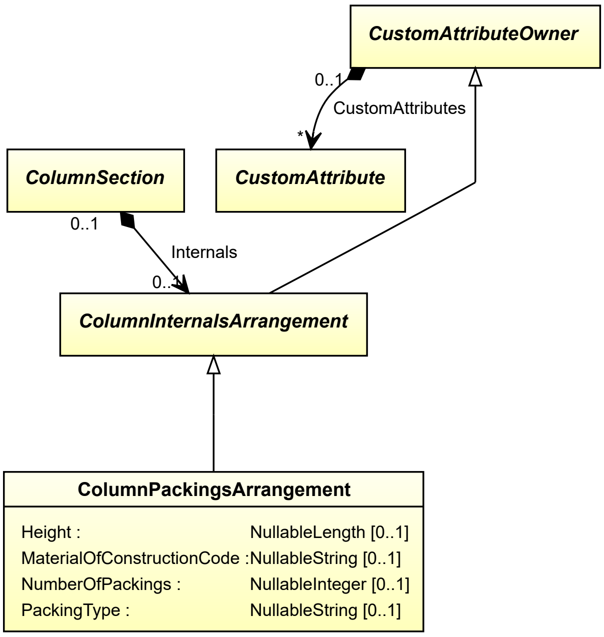
```
<Equipment
    ID="columnPackingsArrangement1"
    ComponentClass="ColumnPackingsArrangement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ColumnPackingsArrangement" ...>
  ...
</Equipment>
```

7.25. ColumnPackingsArrangement

7.25.1 Overview

Class

The packings of a column.


CustomAttributeOwner
0..1
CustomAttributes
*
ColumnSection
CustomAttribute
0..1
Internals
0..1
ColumnInternalsArrangement
ColumnPackingsArrangement
Height : NullableLength [0..1]
MaterialOfConstructionCode : NullableString [0..1]
NumberOfPackings : NullableInteger [0..1]
PackingType : NullableString [0..1]

Supertypes

- *ColumnInternalsArrangement*

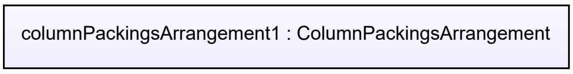
Attributes (data)

Name	Multiplicity	Type
Height	0..1	*NullableLength*
MaterialOfConstructionCode	0..1	*NullableString*
NumberOfPackings	0..1	*NullableInteger*
PackingType	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: COLUMN PACKINGS ARRANGEMENT
ComponentClass: ColumnPackingsArrangement
ComponentClassURI: http://sandbox.dexpi.org/rdl/ColumnPackingsArrangement

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="columnPackingsArrangement1"
    ComponentClass="ColumnPackingsArrangement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ColumnPackingsArrangement" ...>
  ...
</Equipment>
```

7.25.2 Height

Attribute (data)

The height of the *ColumnPackingsArrangement*.

Multiplicity: 0..1

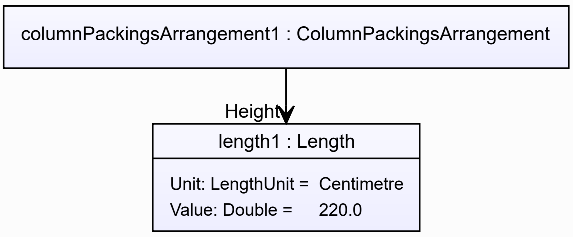
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: HEIGHT
Name: Height
AttributeURI: http://data.posccaesar.org/rdl/RDS357704

Example

The instance columnPackingsArrangement1 represents a *ColumnPackingsArrangement* with a *Height* of 220.0 cm.


columnPackingsArrangement1 : ColumnPackingsArrangement
Height
length1 : Length
Unit: LengthUnit = Centimetre
Value: Double = 220.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="columnPackingsArrangement1"
    ComponentClass="ColumnPackingsArrangement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ColumnPackingsArrangement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="Height"
      AttributeURI="http://data.posccaesar.org/rdl/RDS357704"
      Format="double"
      Value="220.0"
      Units="Centimetre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1318004" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.25.3 MaterialOfConstructionCode

Attribute (data)

A code that gives the material of construction of the *ColumnPackingsArrangement*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: MATERIAL OF CONSTRUCTION CODE ASSIGNMENT CLASS
Name: MaterialOfConstructionCodeAssignmentClass
AttributeURI: http://data.posccaesar.org/rdl/RDS1460719741

Example

“1.4306” (*String*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="columnPackingsArrangement1"
    ComponentClass="ColumnPackingsArrangement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ColumnPackingsArrangement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="MaterialOfConstructionCodeAssignmentClass"
      AttributeURI="http://data.posccaesar.org/rdl/RDS1460719741"
      Format="string"
      Value="1.4306" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.25.4 NumberOfPackings

Attribute (data)

The number of packings in the *ColumnPackingsArrangement*.

Multiplicity: 0..1

Type: *NullableInteger*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for integer values*.
RDL reference: NUMBER OF PACKINGS
Name: NumberOfPackings
AttributeURI: http://sandbox.dexpi.org/rdl/NumberOfPackings

Example

300 (*Integer*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="columnPackingsArrangement1"
    ComponentClass="ColumnPackingsArrangement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ColumnPackingsArrangement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NumberOfPackings"
      AttributeURI="http://sandbox.dexpi.org/rdl/NumberOfPackings"
      Format="integer"
      Value="300" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.25.5 PackingType

Attribute (data)

The type of the packings in the *ColumnPackingsArrangement*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PACKING TYPE ASSIGNMENT CLASS
Name: PackingTypeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PackingTypeAssignmentClass

Example

“rings” (*String*)

Example: Implementation in Proteus Schema

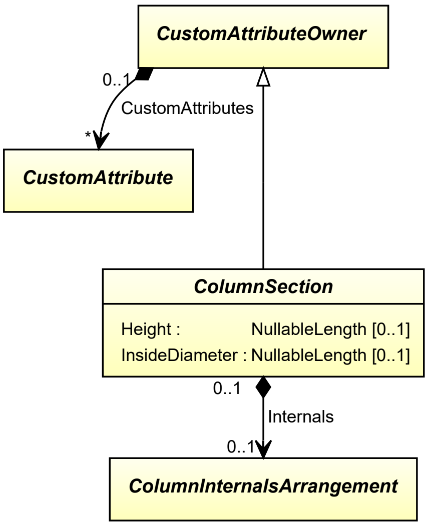
```
<Equipment
    ID="columnPackingsArrangement1"
    ComponentClass="ColumnPackingsArrangement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ColumnPackingsArrangement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PackingTypeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PackingTypeAssignmentClass"
      Format="string"
      Value="rings" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.26. ColumnSection

7.26.1 Overview

Abstract class

A column section.


CustomAttributeOwner
0..1
CustomAttributes
*
CustomAttribute
ColumnSection
Height : NullableLength [0..1]
InsideDiameter : NullableLength [0..1]
0..1
Internals
0..1
ColumnInternalsArrangement

Supertypes

- *CustomAttributeOwner*

Subtypes

- *SubTaggedColumnSection*
- *TaggedColumnSection*

Attributes (data)

Name	Multiplicity	Type
Height	0..1	*NullableLength*
InsideDiameter	0..1	*NullableLength*

Attributes (composition)

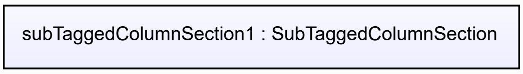
Name	Multiplicity	Type
Internals	0..1	*ColumnInternalsArrangement*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*. As *ColumnSection* is abstract, there is no RDL reference for the class itself; the RDL reference depends on the concrete subclass.
Tag: `<Equipment>`
ComponentClass: *depending on subclass*
ComponentClassURI: *depending on subclass*

Example

As *ColumnSection* is abstract, we consider *SubTaggedColumnSection* as an arbitrary concrete subclass.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="subTaggedColumnSection1"
    ComponentClass="ColumnSection"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ColumnSection" ...>
  ...
</Equipment>
```

7.26.2 Height

Attribute (data)

The height of the *ColumnSection*.

Multiplicity: 0..1

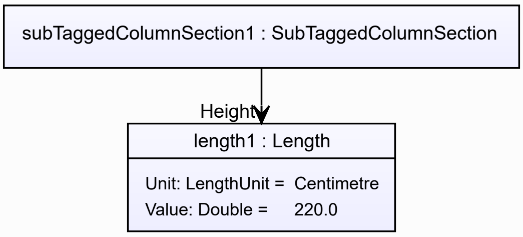
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: HEIGHT
Name: Height
AttributeURI: http://data.posccaesar.org/rdl/RDS357704

Example

As the owning class *ColumnSection* is abstract, we consider *SubTaggedColumnSection* as an arbitrary concrete subclass.
The instance subTaggedColumnSection1 represents a *SubTaggedColumnSection* with a *Height* of 220.0 cm.


subTaggedColumnSection1 : SubTaggedColumnSection
Height
length1 : Length
Unit: LengthUnit = Centimetre
Value: Double = 220.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="subTaggedColumnSection1"
    ComponentClass="ColumnSection"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ColumnSection" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="Height"
      AttributeURI="http://data.posccaesar.org/rdl/RDS357704"
      Format="double"
      Value="220.0"
      Units="Centimetre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1318004" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.26.3 InsideDiameter

Attribute (data)

The inside diameter of the *ColumnSection*.

Multiplicity: 0..1

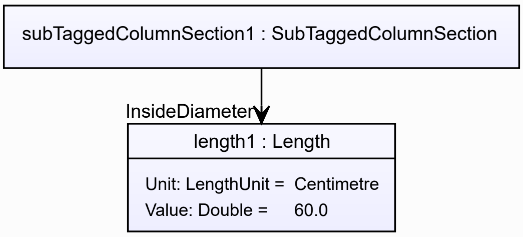
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: INSIDE DIAMETER
Name: InsideDiameter
AttributeURI: http://data.posccaesar.org/rdl/RDS357209

Example

As the owning class *ColumnSection* is abstract, we consider *SubTaggedColumnSection* as an arbitrary concrete subclass.
The instance subTaggedColumnSection1 represents a *SubTaggedColumnSection* with an *InsideDiameter* of 60.0 cm.


subTaggedColumnSection1 : SubTaggedColumnSection
InsideDiameter
length1 : Length
Unit: LengthUnit = Centimetre
Value: Double = 60.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="subTaggedColumnSection1"
    ComponentClass="ColumnSection"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ColumnSection" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="InsideDiameter"
      AttributeURI="http://data.posccaesar.org/rdl/RDS357209"
      Format="double"
      Value="60.0"
      Units="Centimetre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1318004" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.26.4 Internals

Attribute (composition)

The *ColumnInternalsArrangement* of the *ColumnSection*.

Multiplicity: 0..1

Type: *ColumnInternalsArrangement*

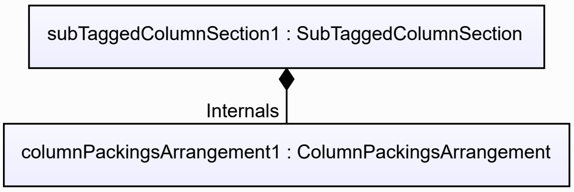
Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *ColumnInternalsArrangement*) is a child of the `<Equipment>` element for the attribute owner (a *ColumnSection*).

Example

As the owner type *ColumnSection* is abstract, we consider *SubTaggedColumnSection* as an arbitrary concrete subclass. As the value type *ColumnInternalsArrangement* is abstract, we consider *ColumnPackingsArrangement* as an arbitrary concrete subclass.


subTaggedColumnSection1 : SubTaggedColumnSection
Internals
columnPackingsArrangement1 : ColumnPackingsArrangement

Example: Implementation in Proteus Schema

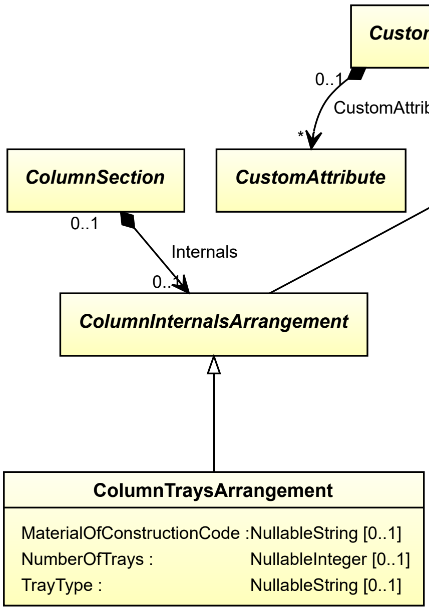
```
<Equipment
    ID="subTaggedColumnSection1"
    ComponentClass="ColumnSection"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ColumnSection" ...>
  ...
  <Equipment
      ID="columnPackingsArrangement1"
      ComponentClass="ColumnPackingsArrangement"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/ColumnPackingsArrangement" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.27. ColumnTraysArrangement

7.27.1 Overview

Class

The trays of a column.


CustomAttributeOwner
0..1
CustomAttributes
*
ColumnSection
CustomAttribute
0..1
Internals
0..1
ColumnInternalsArrangement
ColumnTraysArrangement
MaterialOfConstructionCode :NullableString [0..1]
NumberOfTrays : NullableInteger [0..1]
TrayType : NullableString [0..1]

Supertypes

- *ColumnInternalsArrangement*

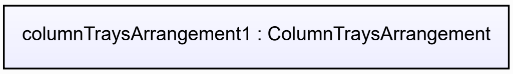
Attributes (data)

Name	Multiplicity	Type
MaterialOfConstructionCode	0..1	*NullableString*
NumberOfTrays	0..1	*NullableInteger*
TrayType	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: COLUMN TRAYS ARRANGEMENT
ComponentClass: ColumnTraysArrangement
ComponentClassURI: http://sandbox.dexpi.org/rdl/ColumnTraysArrangement

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="columnTraysArrangement1"
    ComponentClass="ColumnTraysArrangement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ColumnTraysArrangement" ...>
  ...
</Equipment>
```

7.27.2 MaterialOfConstructionCode

Attribute (data)

A code that gives the material of construction of the *ColumnTraysArrangement*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: MATERIAL OF CONSTRUCTION CODE ASSIGNMENT CLASS
Name: MaterialOfConstructionCodeAssignmentClass
AttributeURI: http://data.posccaesar.org/rdl/RDS1460719741

Example

“1.4306” (*String*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="columnTraysArrangement1"
    ComponentClass="ColumnTraysArrangement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ColumnTraysArrangement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="MaterialOfConstructionCodeAssignmentClass"
      AttributeURI="http://data.posccaesar.org/rdl/RDS1460719741"
      Format="string"
      Value="1.4306" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.27.3 NumberOfTrays

Attribute (data)

The number of trays in the *ColumnTraysArrangement*.

Multiplicity: 0..1

Type: *NullableInteger*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for integer values*.
RDL reference: NUMBER OF TRAYS
Name: NumberOfTrays
AttributeURI: http://sandbox.dexpi.org/rdl/NumberOfTrays

Example

16 (*Integer*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="columnTraysArrangement1"
    ComponentClass="ColumnTraysArrangement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ColumnTraysArrangement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NumberOfTrays"
      AttributeURI="http://sandbox.dexpi.org/rdl/NumberOfTrays"
      Format="integer"
      Value="16" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.27.4 TrayType

Attribute (data)

The type of the trays in the *ColumnTraysArrangement*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: TRAY TYPE ASSIGNMENT CLASS
Name: TrayTypeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/TrayTypeAssignmentClass

Example

“sieve trays” (*String*)

Example: Implementation in Proteus Schema

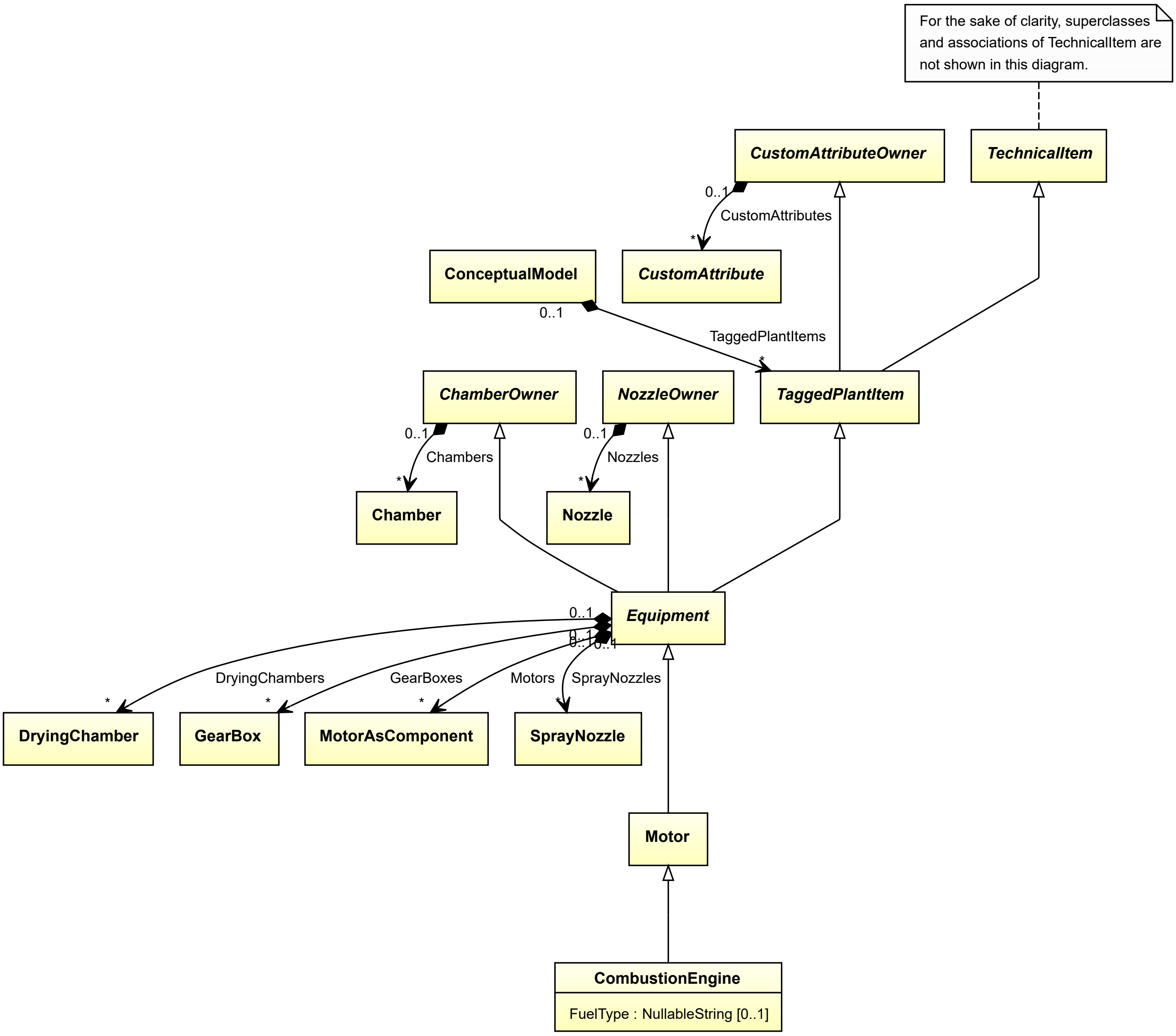
```
<Equipment
    ID="columnTraysArrangement1"
    ComponentClass="ColumnTraysArrangement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ColumnTraysArrangement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="TrayTypeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/TrayTypeAssignmentClass"
      Format="string"
      Value="sieve trays" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.28. CombustionEngine

7.28.1 Overview

Class

An engine intended to deliver power by means of burning fuels (from http://data.posccaesar.org/rdl/RDS1083734).



Supertypes

- *Motor*

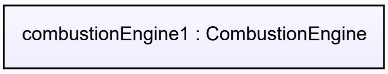
Attributes (data)

Name	Multiplicity	Type
FuelType	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: COMBUSTION ENGINE
ComponentClass: CombustionEngine
ComponentClassURI: http://data.posccaesar.org/rdl/RDS1083734

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="combustionEngine1"
    ComponentClass="CombustionEngine"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS1083734" ...>
  ...
</Equipment>
```

7.28.2 FuelType

Attribute (data)

The fuel type of the *CombustionEngine*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: FUEL TYPE
Name: FuelType
AttributeURI: http://sandbox.dexpi.org/rdl/FuelType

Example

“Diesel fuel” (*String*)

Example: Implementation in Proteus Schema

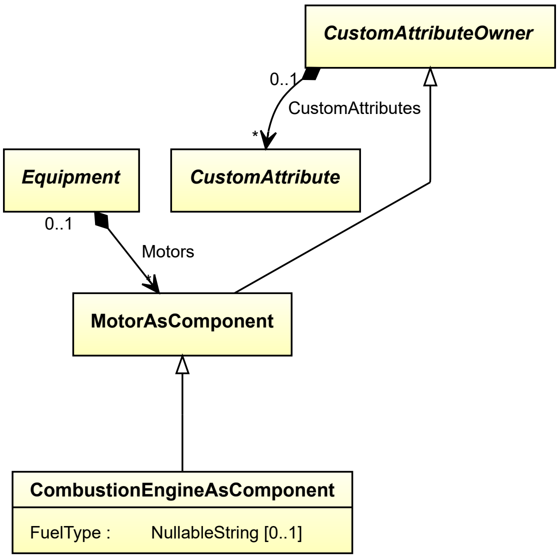
```
<Equipment
    ID="combustionEngine1"
    ComponentClass="CombustionEngine"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS1083734" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="FuelType"
      AttributeURI="http://sandbox.dexpi.org/rdl/FuelType"
      Format="string"
      Value="Diesel fuel" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.29. CombustionEngineAsComponent

7.29.1 Overview

Class

An engine intended to deliver power by means of burning fuels that is used as component of an apparatus or of a machine.



Supertypes

- *MotorAsComponent*

Attributes (data)

Name	Multiplicity	Type
FuelType	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: COMBUSTION ENGINE AS COMPONENT
ComponentClass: CombustionEngineAsComponent
ComponentClassURI: http://sandbox.dexpi.org/rdl/CombustionEngineAsComponent

Example

Example: Implementation in Proteus Schema

```
<Equipment
    ID="combustionEngineAsComponent1"
    ComponentClass="CombustionEngineAsComponent"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CombustionEngineAsComponent" ...>
  ...
</Equipment>
```

7.29.2 FuelType

Attribute (data)

The fuel type of the *CombustionEngineAsComponent*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: FUEL TYPE
Name: FuelType
AttributeURI: http://sandbox.dexpi.org/rdl/FuelType

Example

“Diesel fuel” (*String*)

Example: Implementation in Proteus Schema

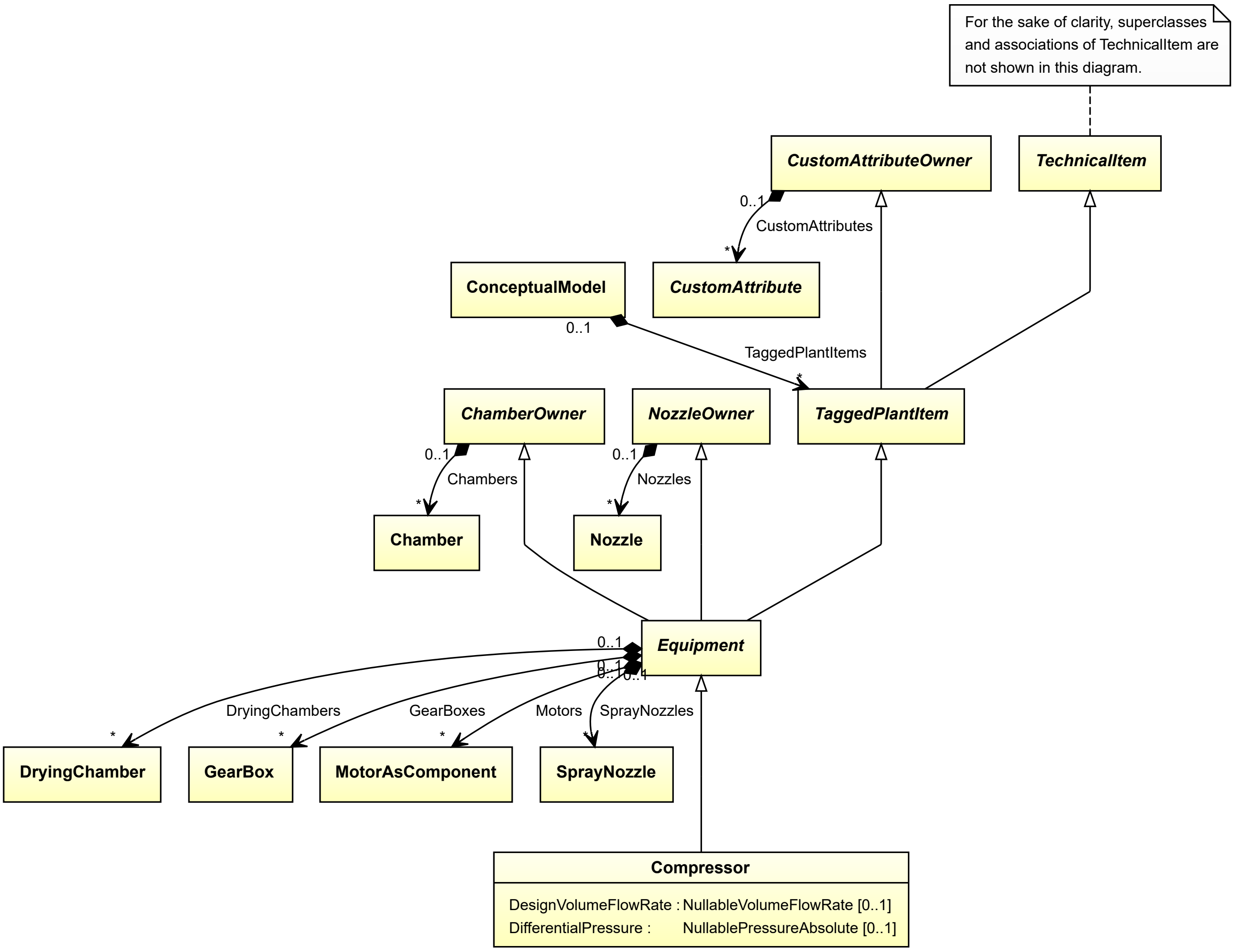
```
<Equipment
    ID="combustionEngineAsComponent1"
    ComponentClass="CombustionEngineAsComponent"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CombustionEngineAsComponent" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="FuelType"
      AttributeURI="http://sandbox.dexpi.org/rdl/FuelType"
      Format="string"
      Value="Diesel fuel" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.30. Compressor

7.30.1 Overview

Class

A machine that has the capability of compressing a gas.



Supertypes

- *Equipment*

Subtypes

- *AirEjector*
- *AxialCompressor*
- *CentrifugalCompressor*
- *CustomCompressor*
- *ReciprocatingCompressor*
- *RotaryCompressor*

Attributes (data)

Name	Multiplicity	Type
DesignVolumeFlowRate	0..1	*NullableVolumeFlowRate*
DifferentialPressure	0..1	*NullablePressureAbsolute*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: COMPRESSOR
ComponentClass: Compressor
ComponentClassURI: http://data.posccaesar.org/rdl/RDS14286497

Example

Example: Implementation in Proteus Schema

```
<Equipment
    ID="compressor1"
    ComponentClass="Compressor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS14286497" ...>
  ...
</Equipment>
```

7.30.2 DesignVolumeFlowRate

Attribute (data)

The volume flow rate for which the *Compressor* is designed.

Multiplicity: 0..1

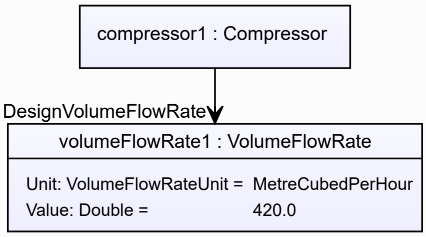
Type: *NullableVolumeFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN VOLUME FLOW RATE
Name: DesignVolumeFlowRate
AttributeURI: http://data.posccaesar.org/rdl/RDS14286227

Example

The instance compressor1 represents a *Compressor* with a *DesignVolumeFlowRate* of 420.0 m^3/h.


compressor1 : Compressor
DesignVolumeFlowRate
volumeFlowRate1 : VolumeFlowRate
Unit: VolumeFlowRateUnit = MetreCubedPerHour
Value: Double = 420.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="compressor1"
    ComponentClass="Compressor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS14286497" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignVolumeFlowRate"
      AttributeURI="http://data.posccaesar.org/rdl/RDS14286227"
      Format="double"
      Value="420.0"
      Units="MetreCubedPerHour"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1321064" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.30.3 DifferentialPressure

Attribute (data)

The differential pressure of the *Compressor*.

Multiplicity: 0..1

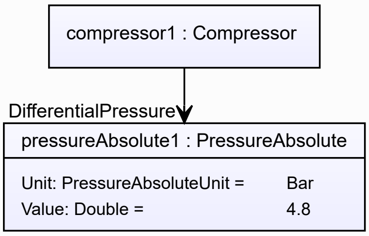
Type: *NullablePressureAbsolute*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DIFFERENTIAL PRESSURE
Name: DifferentialPressure
AttributeURI: http://data.posccaesar.org/rdl/RDS361574

Example

The instance compressor1 represents a *Compressor* with a *DifferentialPressure* of 4.8 bar.


compressor1 : Compressor
DifferentialPressure
pressureAbsolute1 : PressureAbsolute
Unit: PressureAbsoluteUnit = Bar
Value: Double = 4.8

Example: Implementation in Proteus Schema

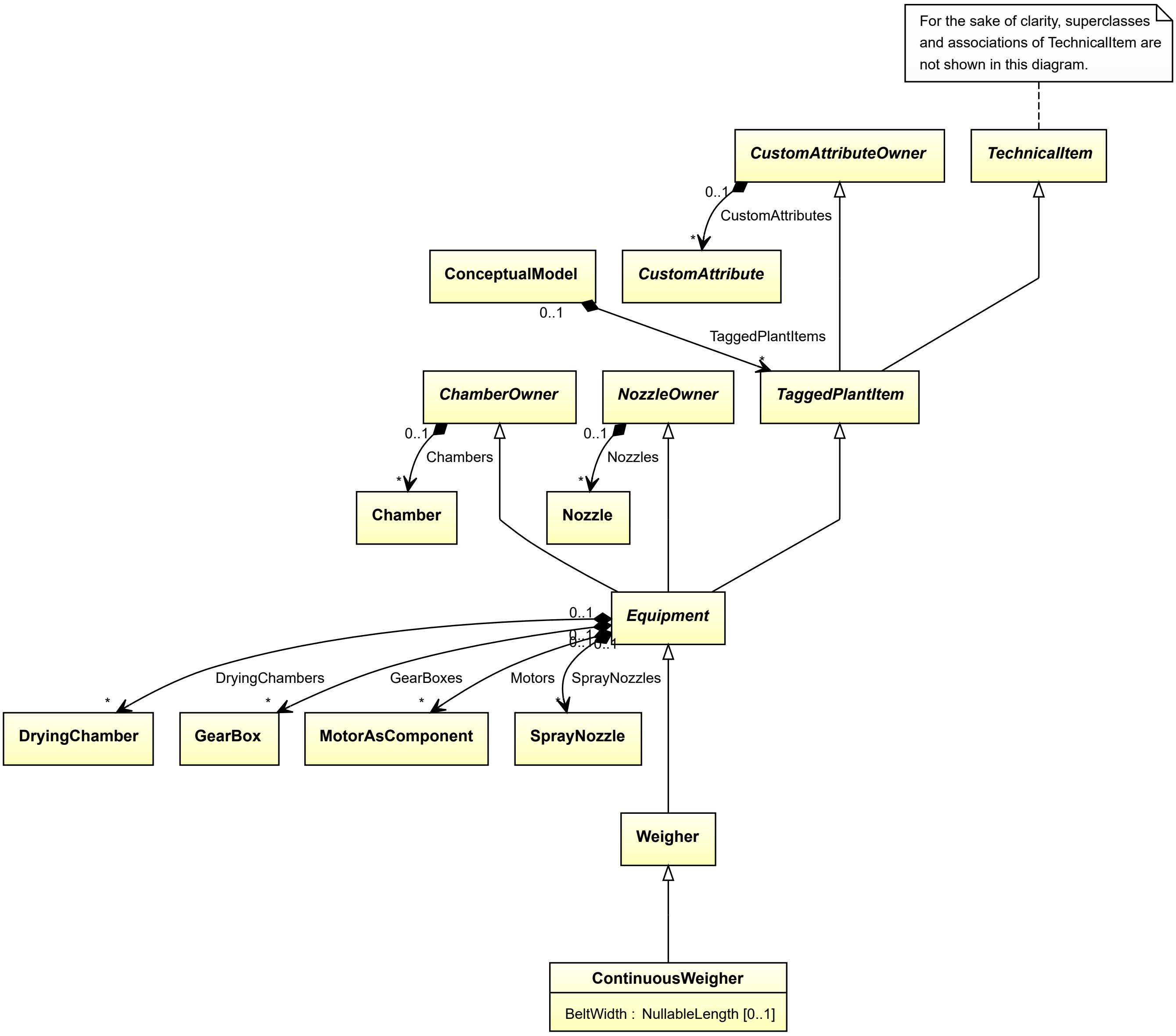
```
<Equipment
    ID="compressor1"
    ComponentClass="Compressor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS14286497" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DifferentialPressure"
      AttributeURI="http://data.posccaesar.org/rdl/RDS361574"
      Format="double"
      Value="4.8"
      Units="Bar"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1314539" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.31. ContinuousWeigher

7.31.1 Overview

Class

A *Weigher* that weighs a mass flow rate in continuous mode.



Supertypes

- *Weigher*

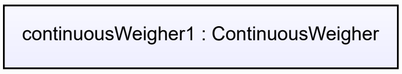
Attributes (data)

Name	Multiplicity	Type
BeltWidth	0..1	*NullableLength*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: CONTINUOUS WEIGHER
ComponentClass: ContinuousWeigher
ComponentClassURI: http://sandbox.dexpi.org/rdl/ContinuousWeigher

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="continuousWeigher1"
    ComponentClass="ContinuousWeigher"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ContinuousWeigher" ...>
  ...
</Equipment>
```

7.31.2 BeltWidth

Attribute (data)

The belt width of the *ContinuousWeigher*.

Multiplicity: 0..1

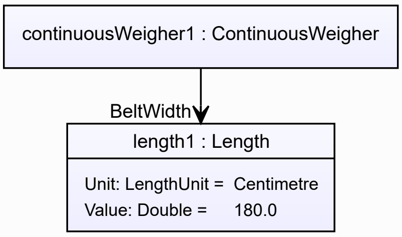
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: BELT WIDTH
Name: BeltWidth
AttributeURI: http://sandbox.dexpi.org/rdl/BeltWidth

Example

The instance continuousWeigher1 represents a *ContinuousWeigher* with a *BeltWidth* of 180.0 cm.


continuousWeigher1 : ContinuousWeigher
BeltWidth
length1 : Length
Unit: LengthUnit = Centimetre
Value: Double = 180.0

Example: Implementation in Proteus Schema

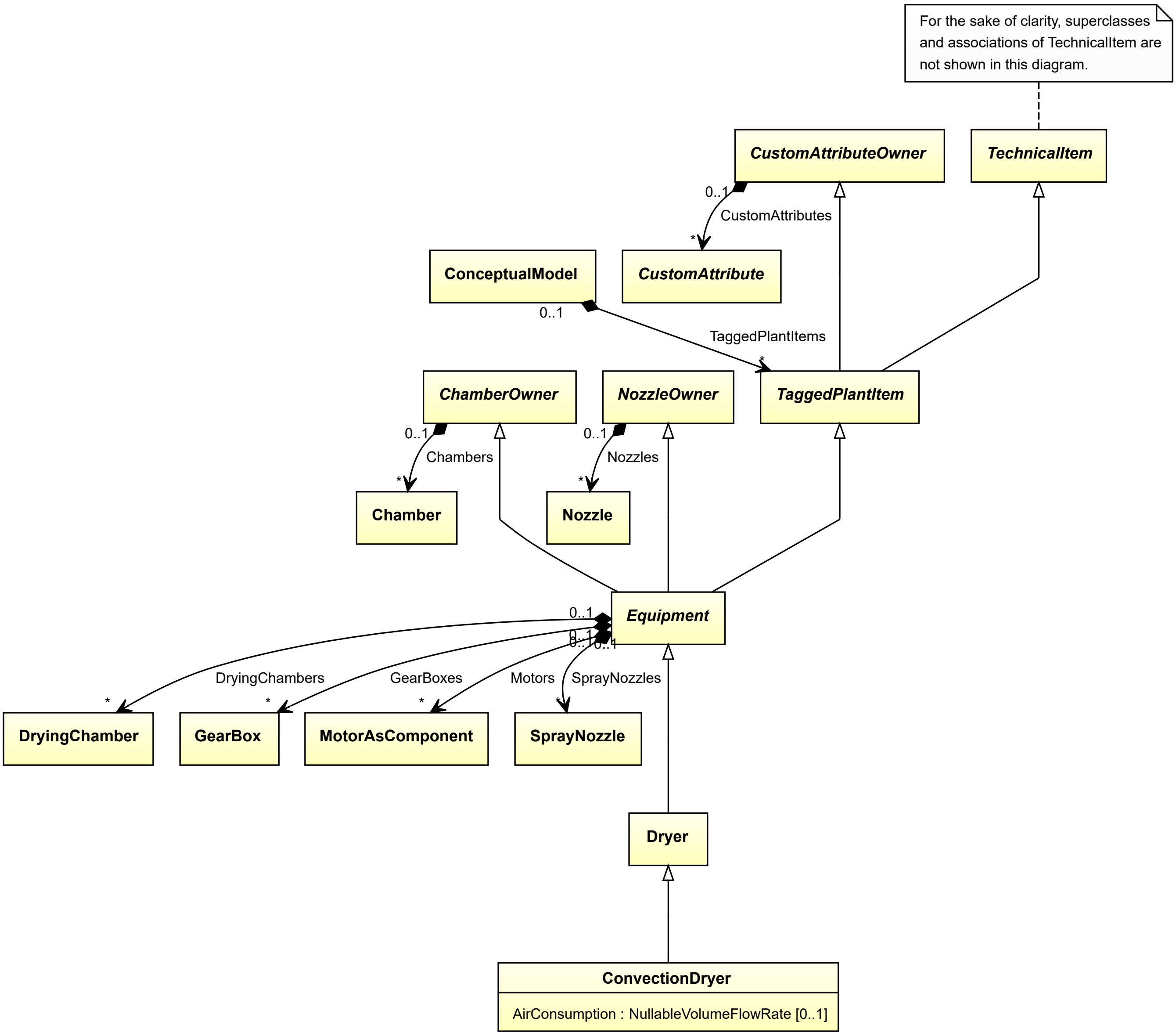
```
<Equipment
    ID="continuousWeigher1"
    ComponentClass="ContinuousWeigher"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ContinuousWeigher" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="BeltWidth"
      AttributeURI="http://sandbox.dexpi.org/rdl/BeltWidth"
      Format="double"
      Value="180.0"
      Units="Centimetre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1318004" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.32. ConvectionDryer

7.32.1 Overview

Class

A *Dryer* that dries a material by bringing it in contact with a drying gas.



Supertypes

- *Dryer*

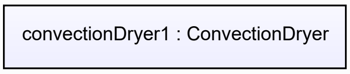
Attributes (data)

Name	Multiplicity	Type
AirConsumption	0..1	*NullableVolumeFlowRate*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: CONVECTION DRYER
ComponentClass: ConvectionDryer
ComponentClassURI: http://sandbox.dexpi.org/rdl/ConvectionDryer

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="convectionDryer1"
    ComponentClass="ConvectionDryer"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ConvectionDryer" ...>
  ...
</Equipment>
```

7.32.2 AirConsumption

Attribute (data)

The consumed air flow of the *ConvectionDryer*.

Multiplicity: 0..1

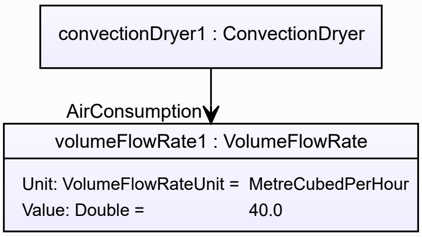
Type: *NullableVolumeFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: AIR CONSUMPTION
Name: AirConsumption
AttributeURI: http://data.posccaesar.org/rdl/RDS5875300

Example

The instance convectionDryer1 represents a *ConvectionDryer* with an *AirConsumption* of 40.0 m^3/h.


convectionDryer1 : ConvectionDryer
AirConsumption
volumeFlowRate1 : VolumeFlowRate
Unit: VolumeFlowRateUnit = MetreCubedPerHour
Value: Double = 40.0

Example: Implementation in Proteus Schema

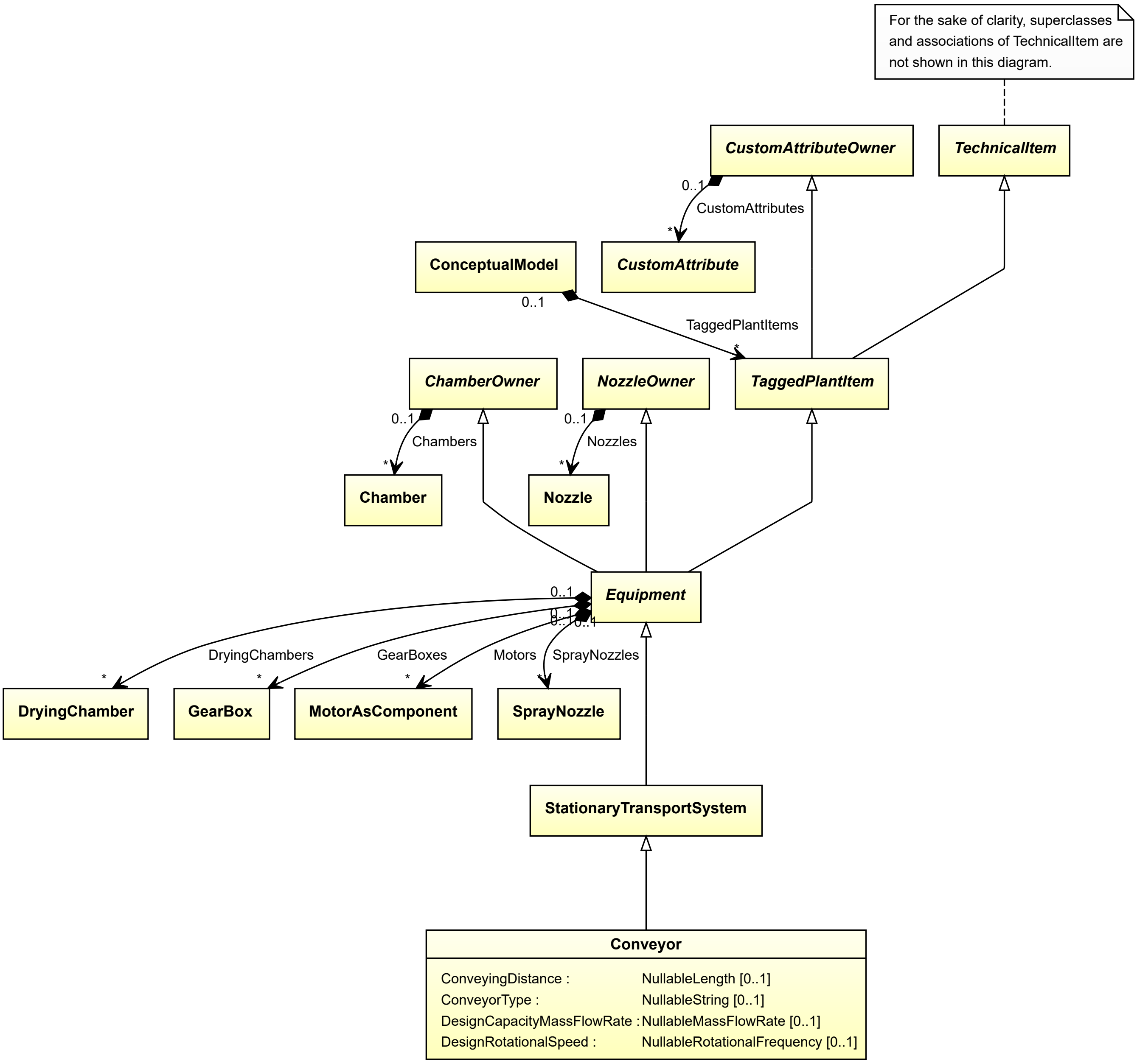
```
<Equipment
    ID="convectionDryer1"
    ComponentClass="ConvectionDryer"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ConvectionDryer" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="AirConsumption"
      AttributeURI="http://data.posccaesar.org/rdl/RDS5875300"
      Format="double"
      Value="40.0"
      Units="MetreCubedPerHour"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1321064" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.33. Conveyor

7.33.1 Overview

Class

A machine that is capable of conveying material.


For the sake of clarity, superclasses and associations of TechnicalItem are not shown in this diagram.
CustomAttributeOwner
TechnicalItem
0..1
CustomAttributes
*
ConceptualModel
CustomAttribute
0..1
TaggedPlantItems
*
ChamberOwner
NozzleOwner
TaggedPlantItem
0..1
Chambers
0..1
Nozzles
*
*
Chamber
Nozzle
Equipment
0..1
0..1
0..1
0..1
DryingChambers
GearBoxes
Motors
SprayNozzles
*
*
*
*
DryingChamber
GearBox
MotorAsComponent
SprayNozzle
StationaryTransportSystem
Conveyor
ConveyingDistance : NullableLength [0..1]
ConveyorType : NullableString [0..1]
DesignCapacityMassFlowRate : NullableMassFlowRate [0..1]
DesignRotationalSpeed : NullableRotationalFrequency [0..1]

Supertypes

- *StationaryTransportSystem*

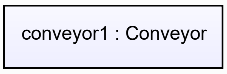
Attributes (data)

Name	Multiplicity	Type
ConveyingDistance	0..1	*NullableLength*
ConveyorType	0..1	*NullableString*
DesignCapacityMassFlowRate	0..1	*NullableMassFlowRate*
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: CONVEYOR
ComponentClass: Conveyor
ComponentClassURI: http://data.posccaesar.org/rdl/RDS11589895

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="conveyor1"
    ComponentClass="Conveyor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11589895" ...>
  ...
</Equipment>
```

7.33.2 ConveyingDistance

Attribute (data)

The conveying distance of the *Conveyor*.

Multiplicity: 0..1

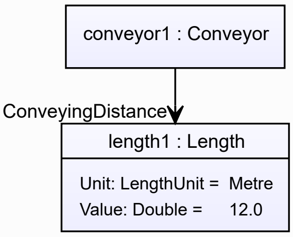
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: CONVEYING DISTANCE
Name: ConveyingDistance
AttributeURI: http://sandbox.dexpi.org/rdl/ConveyingDistance

Example

The instance conveyor1 represents a *Conveyor* with a *ConveyingDistance* of 12.0 m.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="conveyor1"
    ComponentClass="Conveyor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11589895" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ConveyingDistance"
      AttributeURI="http://sandbox.dexpi.org/rdl/ConveyingDistance"
      Format="double"
      Value="12.0"
      Units="Metre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1332674" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.33.3 ConveyorType

Attribute (data)

The type of the conveyor.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: CONVEYOR TYPE ASSIGNMENT CLASS
Name: ConveyorTypeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ConveyorTypeAssignmentClass

Example

“Chain Conveyor” (*String*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="conveyor1"
    ComponentClass="Conveyor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11589895" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ConveyorTypeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ConveyorTypeAssignmentClass"
      Format="string"
      Value="Chain Conveyor" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.33.4 DesignCapacityMassFlowRate

Attribute (data)

The capacity for the mass flow rate for which the *Conveyor* is designed.

Multiplicity: 0..1

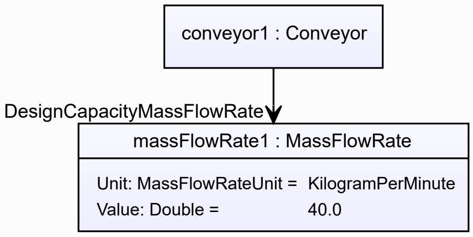
Type: *NullableMassFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN CAPACITY MASS FLOW RATE
Name: DesignCapacityMassFlowRate
AttributeURI: http://sandbox.dexpi.org/rdl/DesignCapacityMassFlowRate

Example

The instance conveyor1 represents a *Conveyor* with a *DesignCapacityMassFlowRate* of 40.0 kg/min.


conveyor1 : Conveyor
DesignCapacityMassFlowRate
massFlowRate1 : MassFlowRate
Unit: MassFlowRateUnit = KilogramPerMinute
Value: Double = 40.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="conveyor1"
    ComponentClass="Conveyor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11589895" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignCapacityMassFlowRate"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignCapacityMassFlowRate"
      Format="double"
      Value="40.0"
      Units="KilogramPerMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1350719" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.33.5 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *Conveyor* is designed.

Multiplicity: 0..1

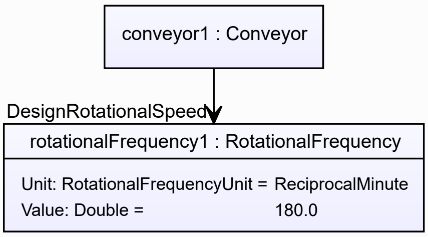
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

Example

The instance conveyor1 represents a *Conveyor* with a *DesignRotationalSpeed* of 180.0 min^{-1}.


conveyor1 : Conveyor
DesignRotationalSpeed
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 180.0

Example: Implementation in Proteus Schema

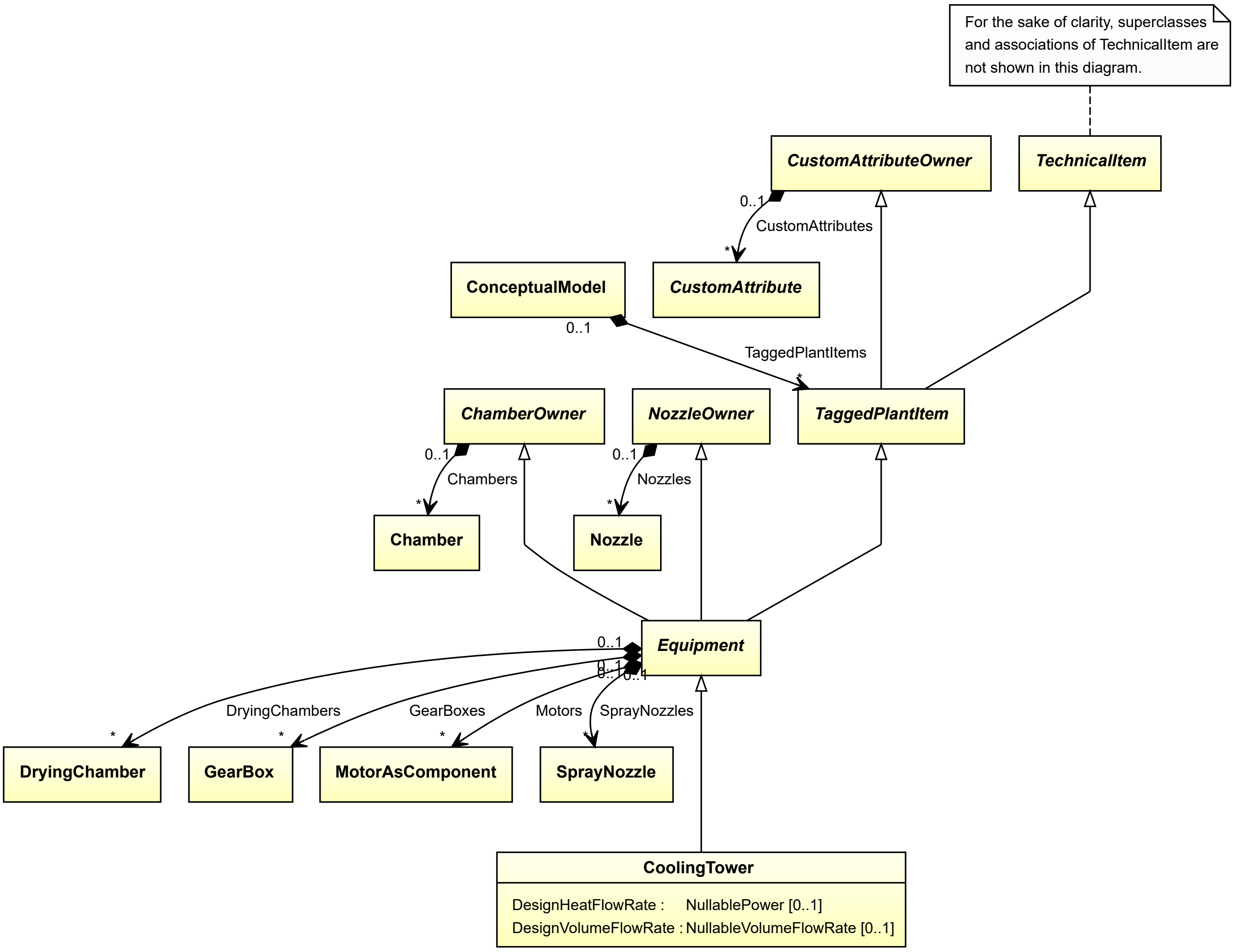
```
<Equipment
    ID="conveyor1"
    ComponentClass="Conveyor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11589895" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignRotationalSpeed"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignRotationalSpeed"
      Format="double"
      Value="180.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.34. CoolingTower

7.34.1 Overview

Class

A cooler and an air cooled heat exchanger that is a tall structure through which air circulates by convection (from http://data.posccaesar.org/rdl/RDS14072341).



Supertypes

- *Equipment*

Subtypes

- *CustomCoolingTower*
- *DryCoolingTower*
- *SprayCooler*
- *WetCoolingTower*

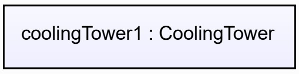
Attributes (data)

Name	Multiplicity	Type
DesignHeatFlowRate	0..1	*NullablePower*
DesignVolumeFlowRate	0..1	*NullableVolumeFlowRate*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: COOLING TOWER
ComponentClass: CoolingTower
ComponentClassURI: http://data.posccaesar.org/rdl/RDS14072341

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="coolingTower1"
    ComponentClass="CoolingTower"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS14072341" ...>
  ...
</Equipment>
```

7.34.2 DesignHeatFlowRate

Attribute (data)

The heat flow rate for which the *CoolingTower* is designed.

Multiplicity: 0..1

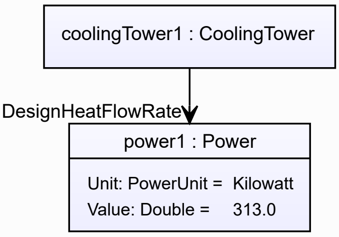
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN HEAT FLOW RATE
Name: DesignHeatFlowRate
AttributeURI: http://sandbox.dexpi.org/rdl/DesignHeatFlowRate

Example

The instance coolingTower1 represents a *CoolingTower* with a *DesignHeatFlowRate* of 313.0 kW.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="coolingTower1"
    ComponentClass="CoolingTower"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS14072341" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignHeatFlowRate"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignHeatFlowRate"
      Format="double"
      Value="313.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.34.3 DesignVolumeFlowRate

Attribute (data)

The volume flow rate for which the *CoolingTower* is designed.

Multiplicity: 0..1

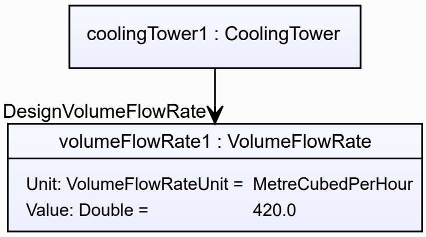
Type: *NullableVolumeFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN VOLUME FLOW RATE
Name: DesignVolumeFlowRate
AttributeURI: http://data.posccaesar.org/rdl/RDS14286227

Example

The instance coolingTower1 represents a *CoolingTower* with a *DesignVolumeFlowRate* of 420.0 m^3/h.


coolingTower1 : CoolingTower
DesignVolumeFlowRate
volumeFlowRate1 : VolumeFlowRate
Unit: VolumeFlowRateUnit = MetreCubedPerHour
Value: Double = 420.0

Example: Implementation in Proteus Schema

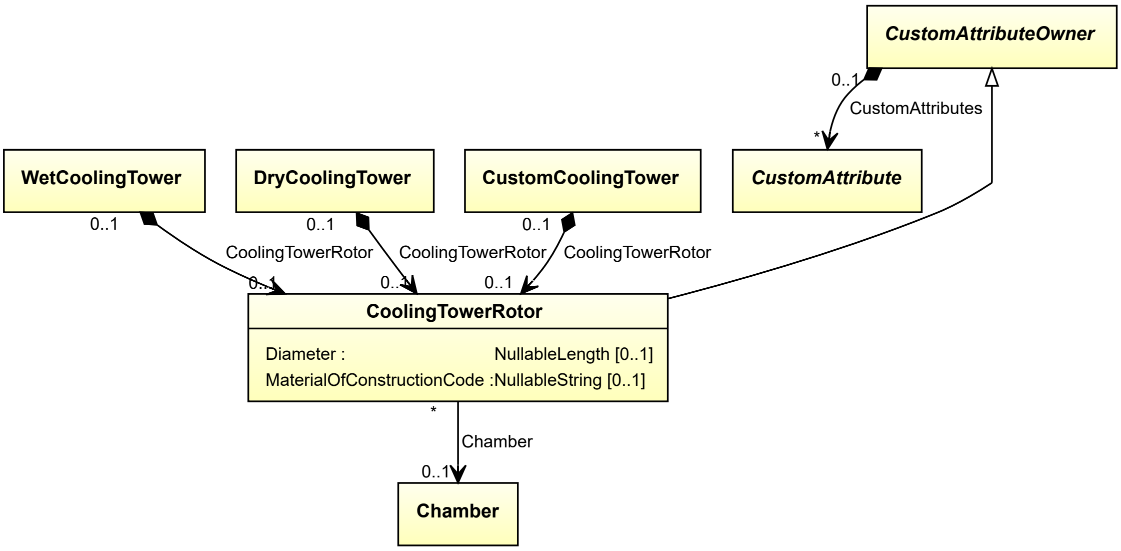
```
<Equipment
    ID="coolingTower1"
    ComponentClass="CoolingTower"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS14072341" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignVolumeFlowRate"
      AttributeURI="http://data.posccaesar.org/rdl/RDS14286227"
      Format="double"
      Value="420.0"
      Units="MetreCubedPerHour"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1321064" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.35. CoolingTowerRotor

7.35.1 Overview

Class

A rotor of a cooling tower.



Supertypes

- *CustomAttributeOwner*

Attributes (data)

Name	Multiplicity	Type
Diameter	0..1	*NullableLength*
MaterialOfConstructionCode	0..1	*NullableString*

Attributes (reference)

Name	Multiplicity	Type
Chamber	0..1	*Chamber*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: COOLING TOWER ROTOR
ComponentClass: CoolingTowerRotor
ComponentClassURI: http://sandbox.dexpi.org/rdl/CoolingTowerRotor

Example

coolingTowerRotor1 : CoolingTowerRotor

Example: Implementation in Proteus Schema

```
<Equipment
    ID="coolingTowerRotor1"
    ComponentClass="CoolingTowerRotor"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CoolingTowerRotor" ...>
  ...
</Equipment>
```

7.35.2 Chamber

Attribute (reference)

The *Chamber* in which the *CoolingTowerRotor* is located, if applicable. The Chamber must be a component of the same object as the *CoolingTowerRotor*.

Multiplicity: 0..1

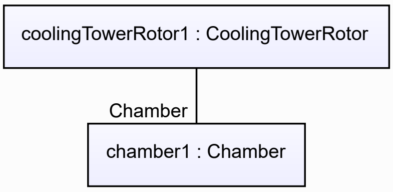
Type: *Chamber*

Opposite multiplicity: 0..*

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: `"is located in"`
Opposite association type: `"is the location of"`

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="coolingTowerRotor1"
    ComponentClass="CoolingTowerRotor"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CoolingTowerRotor" ...>
  ...
  <Association
    Type="is located in"
    ItemID="chamber1" />
  ...
<Equipment />
...
<Equipment
    ID="chamber1"
    ComponentClass="Chamber"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS903151421" ...>
  ...
  <Association
    Type="is the location of"
    ItemID="coolingTowerRotor1" />
  ...
<Equipment />
```

7.35.3 Diameter

Attribute (data)

The diameter of the *CoolingTowerRotor*.

Multiplicity: 0..1

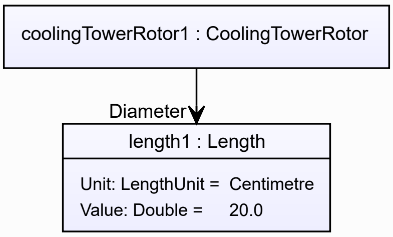
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DIAMETER
Name: Diameter
AttributeURI: http://data.posccaesar.org/rdl/RDS350954

Example

The instance coolingTowerRotor1 represents a *CoolingTowerRotor* with a *Diameter* of 20.0 cm.


coolingTowerRotor1 : CoolingTowerRotor
Diameter
length1 : Length
Unit: LengthUnit = Centimetre
Value: Double = 20.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="coolingTowerRotor1"
    ComponentClass="CoolingTowerRotor"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CoolingTowerRotor" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="Diameter"
      AttributeURI="http://data.posccaesar.org/rdl/RDS350954"
      Format="double"
      Value="20.0"
      Units="Centimetre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1318004" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.35.4 MaterialOfConstructionCode

Attribute (data)

A code that gives the material of construction of the *CoolingTowerRotor*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: MATERIAL OF CONSTRUCTION CODE ASSIGNMENT CLASS
Name: MaterialOfConstructionCodeAssignmentClass
AttributeURI: http://data.posccaesar.org/rdl/RDS1460719741

Example

“1.4306” (*String*)

Example: Implementation in Proteus Schema

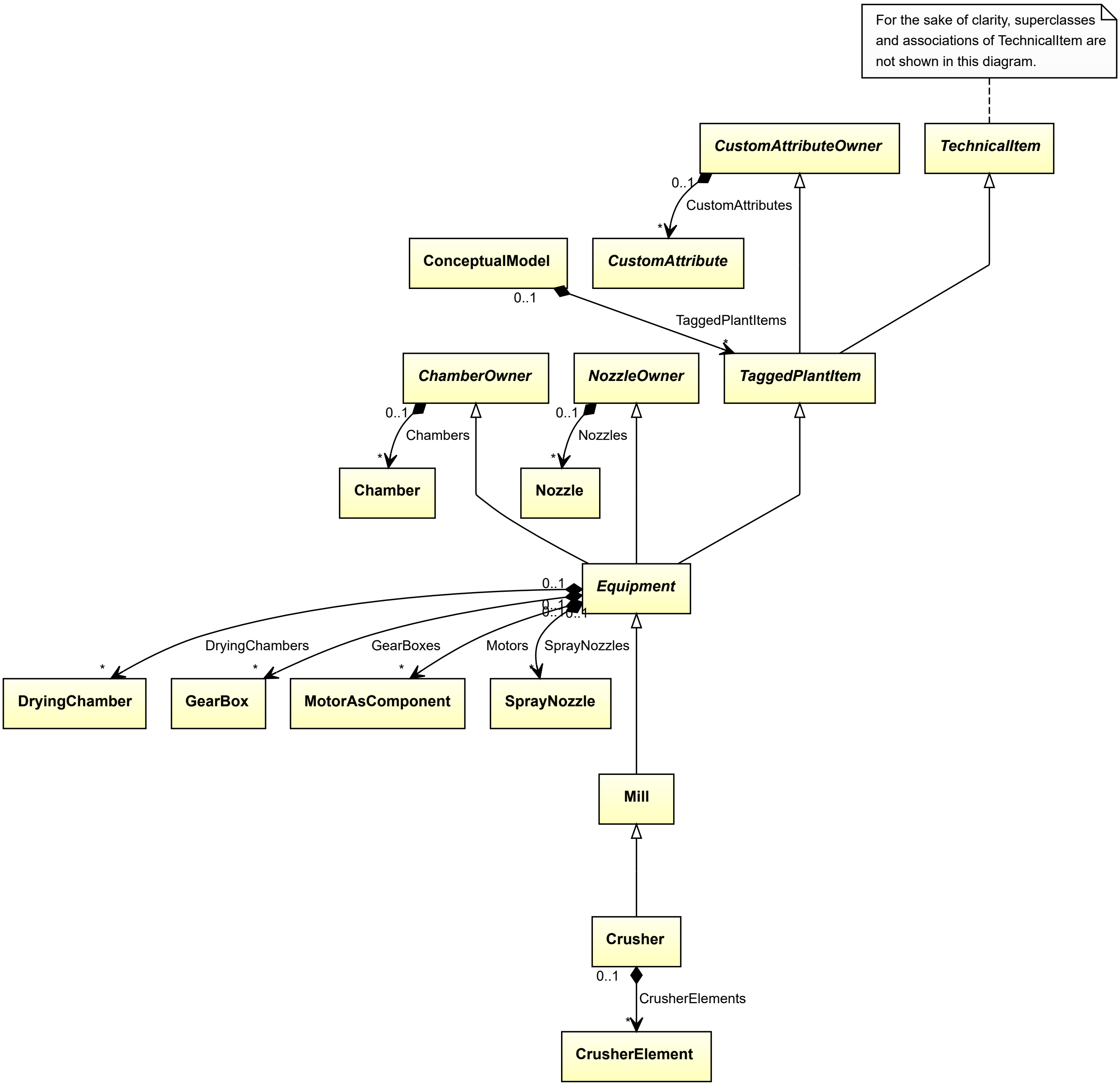
```
<Equipment
    ID="coolingTowerRotor1"
    ComponentClass="CoolingTowerRotor"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CoolingTowerRotor" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="MaterialOfConstructionCodeAssignmentClass"
      AttributeURI="http://data.posccaesar.org/rdl/RDS1460719741"
      Format="string"
      Value="1.4306" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.36. Crusher

7.36.1 Overview

Class

A mill that uses pressure or impact to reduce the particle size of solid materials (from http://data.posccaesar.org/rdl/RDS11589940).



Supertypes

- *Mill*

Attributes (composition)

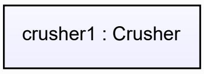
Name	Multiplicity	Type
CrusherElements	*	*CrusherElement*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`

RDL reference: CRUSHER
ComponentClass: Crusher
ComponentClassURI: http://data.posccaesar.org/rdl/RDS11589940

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="crusher1"
    ComponentClass="Crusher"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11589940" ...>
  ...
</Equipment>
```

7.36.2 CrusherElements

Attribute (composition)

The crusher elements of the *Crusher*.

Multiplicity: *

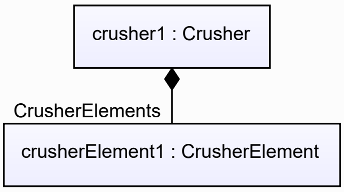
Type: *CrusherElement*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *CrusherElement*) is a child of the `<Equipment>` element for the attribute owner (a *Crusher*).

Example



Example: Implementation in Proteus Schema

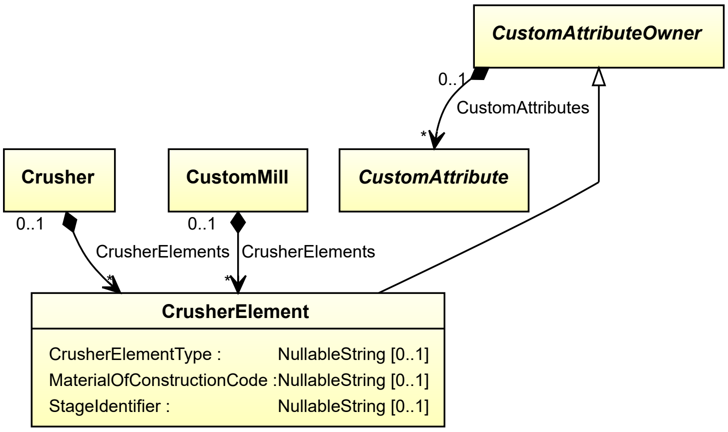
```
<Equipment
    ID="crusher1"
    ComponentClass="Crusher"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11589940" ...>
  ...
  <Equipment
      ID="crusherElement1"
      ComponentClass="CrusherUnit"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/CrusherUnit" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.37. CrusherElement

7.37.1 Overview

Class

A functional component of a *Crusher*.


CustomAttributeOwner
0..1
CustomAttributes
*
Crusher
CustomMill
CustomAttribute
0..1
0..1
CrusherElements
CrusherElements
*
*
CrusherElement
CrusherElementType : NullableString [0..1]
MaterialOfConstructionCode : NullableString [0..1]
StageIdentifier : NullableString [0..1]

Supertypes

- *CustomAttributeOwner*

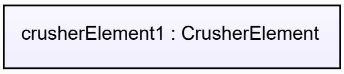
Attributes (data)

Name	Multiplicity	Type
CrusherElementType	0..1	*NullableString*
MaterialOfConstructionCode	0..1	*NullableString*
StageIdentifier	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: CRUSHER UNIT
ComponentClass: CrusherUnit
ComponentClassURI: http://sandbox.dexpi.org/rdl/CrusherUnit

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="crusherElement1"
    ComponentClass="CrusherUnit"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CrusherUnit" ...>
  ...
</Equipment>
```

7.37.2 CrusherElementType

Attribute (data)

The type of the *CrusherElement*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: CRUSHER ELEMENT TYPE ASSIGNMENT CLASS
Name: CrusherElementTypeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/CrusherElementTypeAssignmentClass

Example

“Cone Crusher” (*String*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="crusherElement1"
    ComponentClass="CrusherUnit"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CrusherUnit" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="CrusherElementTypeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/CrusherElementTypeAssignmentClass"
      Format="string"
      Value="Cone Crusher" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.37.3 MaterialOfConstructionCode

Attribute (data)

A code that gives the material of construction of the *CrusherElement*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: MATERIAL OF CONSTRUCTION CODE ASSIGNMENT CLASS
Name: MaterialOfConstructionCodeAssignmentClass
AttributeURI: http://data.posccaesar.org/rdl/RDS1460719741

Example

“1.4306” (*String*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="crusherElement1"
    ComponentClass="CrusherUnit"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CrusherUnit" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="MaterialOfConstructionCodeAssignmentClass"
      AttributeURI="http://data.posccaesar.org/rdl/RDS1460719741"
      Format="string"
      Value="1.4306" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.37.4 StageIdentifier

Attribute (data)

The stage identifier of the *CrusherElement*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: STAGE IDENTIFIER ASSIGNMENT CLASS
Name: StageIdentifierAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/StageIdentifierAssignmentClass

Example

“s1” (*String*)

Example: Implementation in Proteus Schema

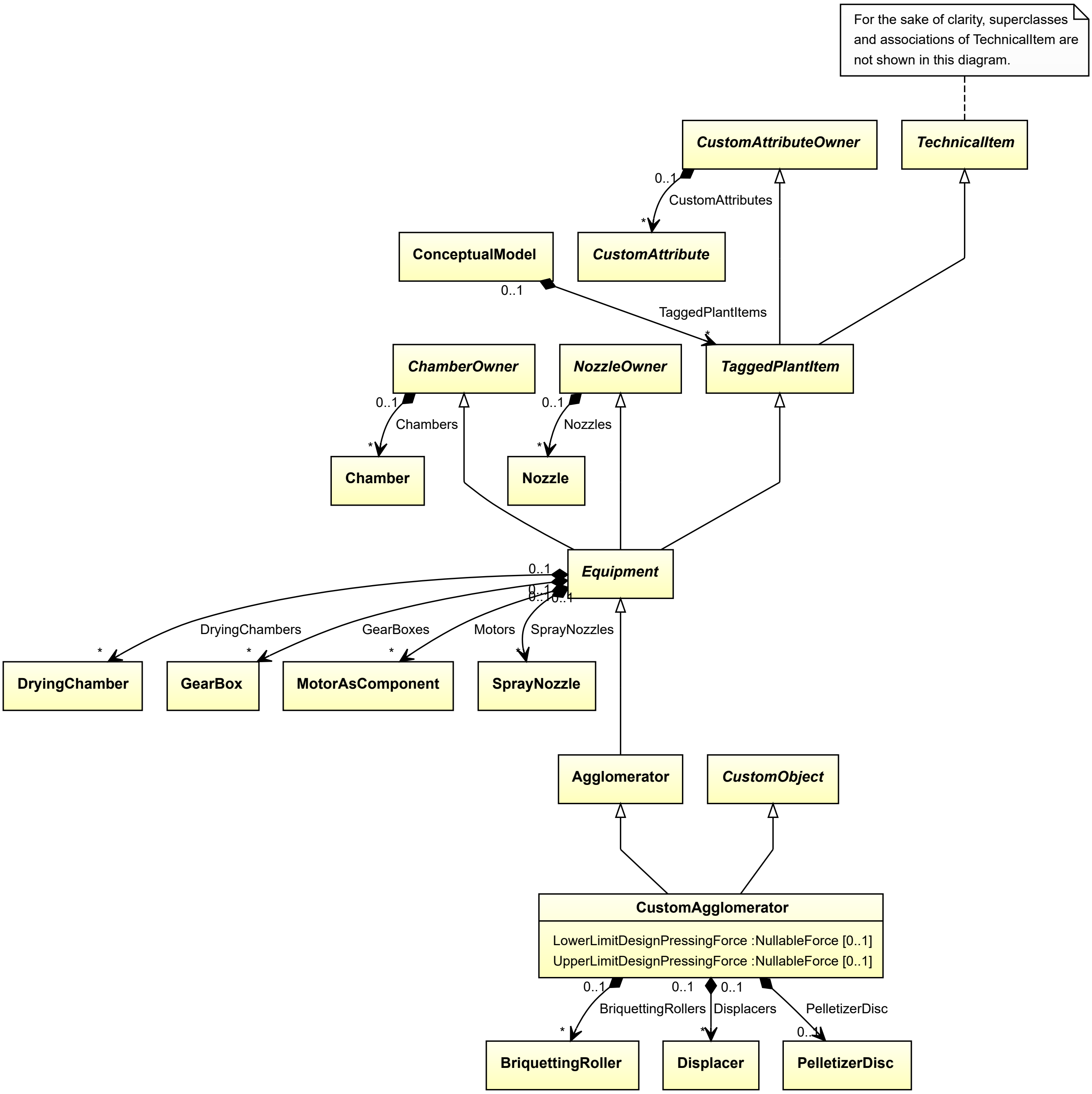
```
<Equipment
    ID="crusherElement1"
    ComponentClass="CrusherUnit"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CrusherUnit" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="StageIdentifierAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/StageIdentifierAssignmentClass"
      Format="string"
      Value="s1" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.38. CustomAgglomerator

7.38.1 Overview

Class

A custom *Agglomerator*, i.e., an *Agglomerator* that is not covered by any of the other subclasses of *Agglomerator* (*ReciprocatingPressureAgglomerator*, *RotatingGrowthAgglomerator*, or *RotatingPressureAgglomerator*).



Supertypes

- *Agglomerator*
- *CustomObject*

Attributes (data)

Name	Multiplicity	Type
LowerLimitDesignPressingForce	0..1	*NullableForce*
UpperLimitDesignPressingForce	0..1	*NullableForce*

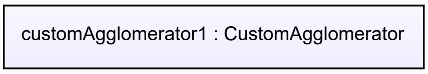
Attributes (composition)

Name	Multiplicity	Type
BriquettingRollers	*	*BriquettingRoller*
Displacers	*	*Displacer*
PelletizerDisc	0..1	*PelletizerDisc*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: CUSTOM AGGLOMERATOR
ComponentClass: CustomAgglomerator
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomAgglomerator

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="customAgglomerator1"
    ComponentClass="CustomAgglomerator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomAgglomerator" ...>
  ...
</Equipment>
```

7.38.2 BriquettingRollers

Attribute (composition)

The briquetting rollers of the *CustomAgglomerator*.

Multiplicity: *

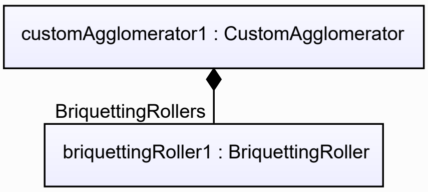
Type: *BriquettingRoller*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *BriquettingRoller*) is a child of the `<Equipment>` element for the attribute owner (a *CustomAgglomerator*).

Example


customAgglomerator1 : CustomAgglomerator
BriquettingRollers
briquettingRoller1 : BriquettingRoller

Example: Implementation in Proteus Schema

```
<Equipment
    ID="customAgglomerator1"
    ComponentClass="CustomAgglomerator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomAgglomerator" ...>
  ...
  <Equipment
      ID="briquettingRoller1"
      ComponentClass="BriquettingRoller"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/BriquettingRoller" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.38.3 Displacers

Attribute (composition)

The displacers of the *CustomAgglomerator*.

Multiplicity: *

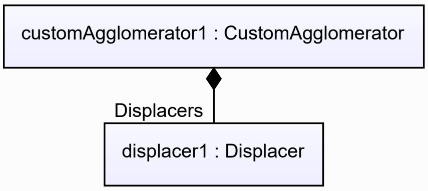
Type: *Displacer*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *Displacer*) is a child of the `<Equipment>` element for the attribute owner (a *CustomAgglomerator*).

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="customAgglomerator1"
    ComponentClass="CustomAgglomerator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomAgglomerator" ...>
  ...
  <Equipment
      ID="displacer1"
      ComponentClass="Displacer"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/Displacer" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.38.4 LowerLimitDesignPressingForce

Attribute (data)

The lower limit for the pressing force for which the *CustomAgglomerator* is designed.

Multiplicity: 0..1

Type: *NullableForce*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: LOWER LIMIT DESIGN PRESSING FORCE
Name: LowerLimitDesignPressingForce
AttributeURI: http://sandbox.dexpi.org/rdl/LowerLimitDesignPressingForce

7.38.5 PelletizerDisc

Attribute (composition)

The pelletizing disc of the *CustomAgglomerator*.

Multiplicity: 0..1

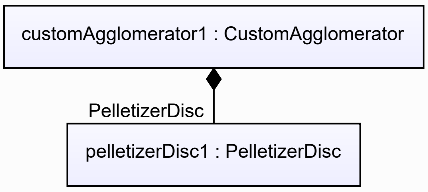
Type: *PelletizerDisc*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *PelletizerDisc*) is a child of the `<Equipment>` element for the attribute owner (a *CustomAgglomerator*).

Example


customAgglomerator1 : CustomAgglomerator
PelletizerDisc
pelletizerDisc1 : PelletizerDisc

Example: Implementation in Proteus Schema

```
<Equipment
    ID="customAgglomerator1"
    ComponentClass="CustomAgglomerator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomAgglomerator" ...>
  ...
  <Equipment
      ID="pelletizerDisc1"
      ComponentClass="PelletingDisc"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/PelletingDisc" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.38.6 UpperLimitDesignPressingForce

Attribute (data)

The upper limit for the pressing force for which the *CustomAgglomerator* is designed.

Multiplicity: 0..1

Type: *NullableForce*

Implementation in Proteus Schema

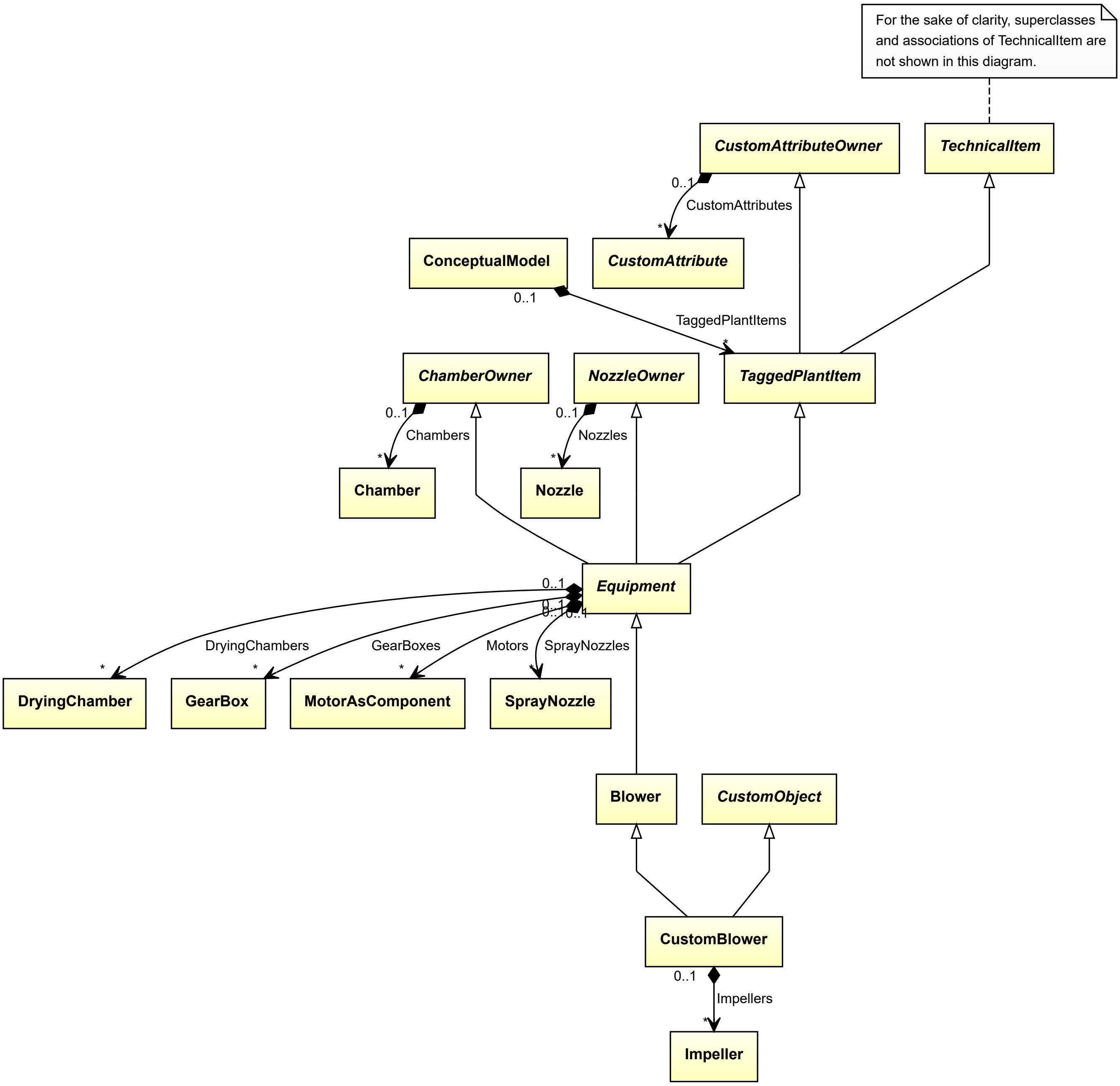
The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT DESIGN PRESSING FORCE
Name: UpperLimitDesignPressingForce
AttributeURI: http://sandbox.dexpi.org/rdl/UpperLimitDesignPressingForce

7.39. CustomBlower

7.39.1 Overview

Class

A custom *Blower*, i.e., a *Blower* that is not covered by any of the other subclasses of *Blower* (*AxialBlower* or *CentrifugalBlower*).



Supertypes

- *Blower*
- *CustomObject*

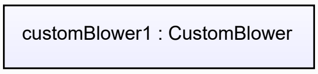
Attributes (composition)

Name	Multiplicity	Type
Impellers	*	*Impeller*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: CUSTOM BLOWER
ComponentClass: CustomBlower
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomBlower

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="customBlower1"
    ComponentClass="CustomBlower"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomBlower" ...>
  ...
</Equipment>
```

7.39.2 Impellers

Attribute (composition)

The impellers of the *CustomBlower*.

Multiplicity: *

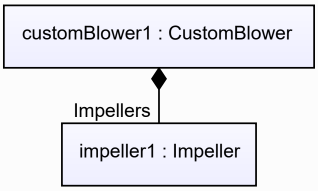
Type: *Impeller*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (an *Impeller*) is a child of the `<Equipment>` element for the attribute owner (a *CustomBlower*).

Example



Example: Implementation in Proteus Schema

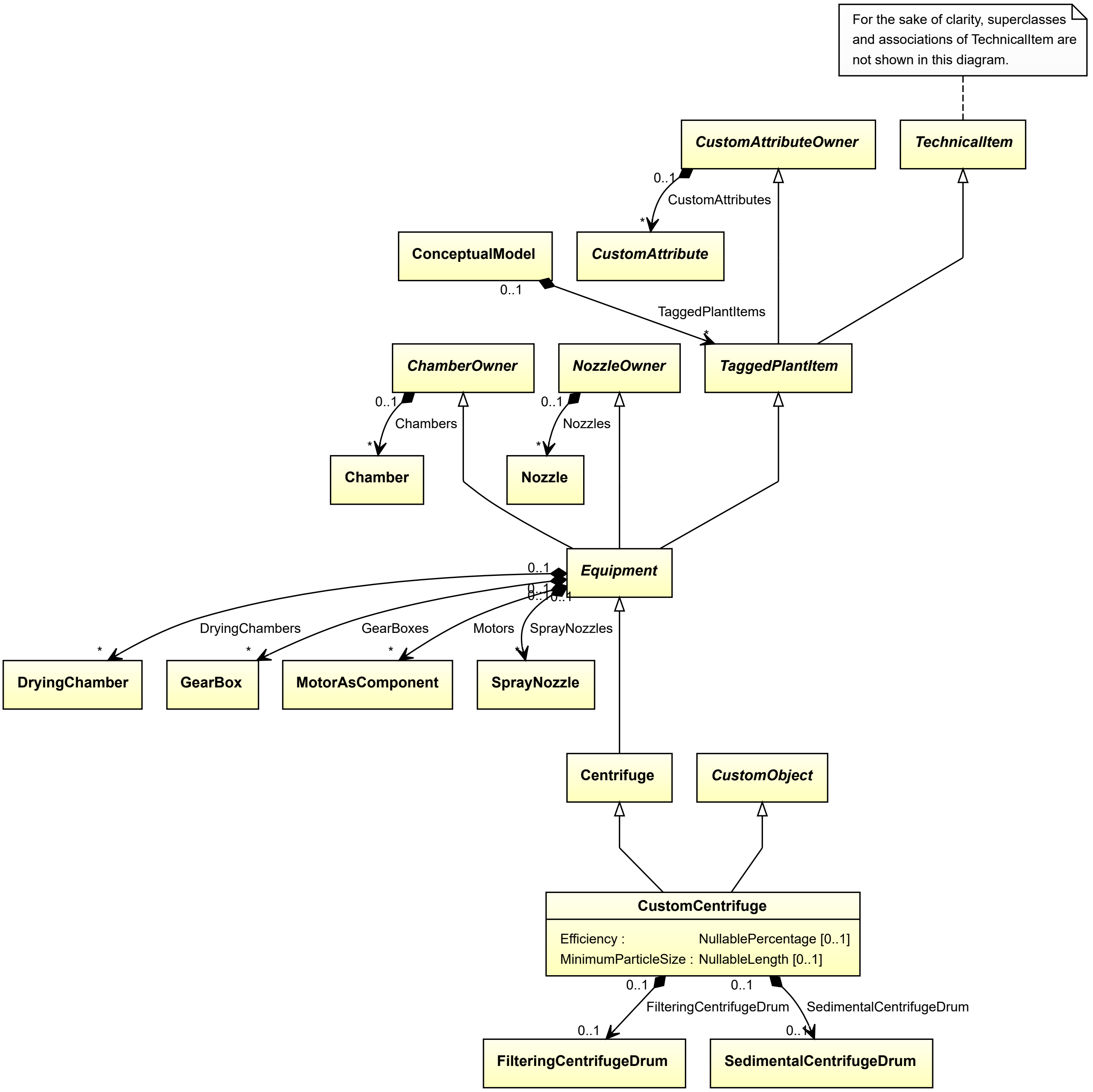
```
<Equipment
    ID="customBlower1"
    ComponentClass="CustomBlower"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomBlower" ...>
  ...
  <Equipment
      ID="impeller1"
      ComponentClass="Impeller"
      ComponentClassURI="http://data.posccaesar.org/rdl/RDS414539" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.40. CustomCentrifuge

7.40.1 Overview

Class

A custom *Centrifuge*, i.e., a *Centrifuge* that is not covered by any of the other subclasses of *Centrifuge* (*FilteringCentrifuge* or *SedimentalCentrifuge*).



Supertypes

- *Centrifuge*
- *CustomObject*

Attributes (data)

Name	Multiplicity	Type
Efficiency	0..1	*NullablePercentage*
MinimumParticleSize	0..1	*NullableLength*

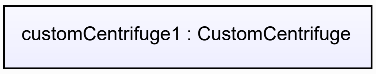
Attributes (composition)

Name	Multiplicity	Type
FilteringCentrifugeDrum	0..1	*FilteringCentrifugeDrum*
SedimentalCentrifugeDrum	0..1	*SedimentalCentrifugeDrum*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: CUSTOM CENTRIFUGE
ComponentClass: CustomCentrifuge
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomCentrifuge

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="customCentrifuge1"
    ComponentClass="CustomCentrifuge"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomCentrifuge" ...>
  ...
</Equipment>
```

7.40.2 Efficiency

Attribute (data)

The efficiency of the *CustomCentrifuge*.

Multiplicity: 0..1

Type: *NullablePercentage*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: EFFICIENCY
Name: Efficiency
AttributeURI: http://data.posccaesar.org/rdl/RDS362654

7.40.3 FilteringCentrifugeDrum

Attribute (composition)

The filtering centrifuge drum of the *CustomCentrifuge*.

Multiplicity: 0..1

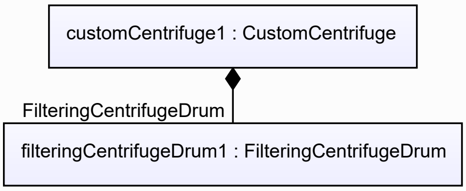
Type: *FilteringCentrifugeDrum*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *FilteringCentrifugeDrum*) is a child of the `<Equipment>` element for the attribute owner (a *CustomCentrifuge*).

Example


customCentrifuge1 : CustomCentrifuge
FilteringCentrifugeDrum
filteringCentrifugeDrum1 : FilteringCentrifugeDrum

Example: Implementation in Proteus Schema

```
<Equipment
    ID="customCentrifuge1"
    ComponentClass="CustomCentrifuge"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomCentrifuge" ...>
  ...
  <Equipment
      ID="filteringCentrifugeDrum1"
      ComponentClass="FilteringCentrifugeDrum"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/FilteringCentrifugeDrum" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.40.4 MinimumParticleSize

Attribute (data)

The minimum particle size of the *CustomCentrifuge*.

Multiplicity: 0..1

Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: MINIMUM PARTICLE SIZE
Name: MinimumParticleSize
AttributeURI: http://sandbox.dexpi.org/rdl/MinimumParticleSize

7.40.5 SedimentalCentrifugeDrum

Attribute (composition)

The sedimental centrifuge drum of the *CustomCentrifuge*.

Multiplicity: 0..1

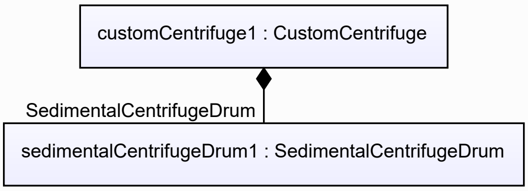
Type: *SedimentalCentrifugeDrum*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *SedimentalCentrifugeDrum*) is a child of the `<Equipment>` element for the attribute owner (a *CustomCentrifuge*).

Example


customCentrifuge1 : CustomCentrifuge
SedimentalCentrifugeDrum
sedimentalCentrifugeDrum1 : SedimentalCentrifugeDrum

Example: Implementation in Proteus Schema

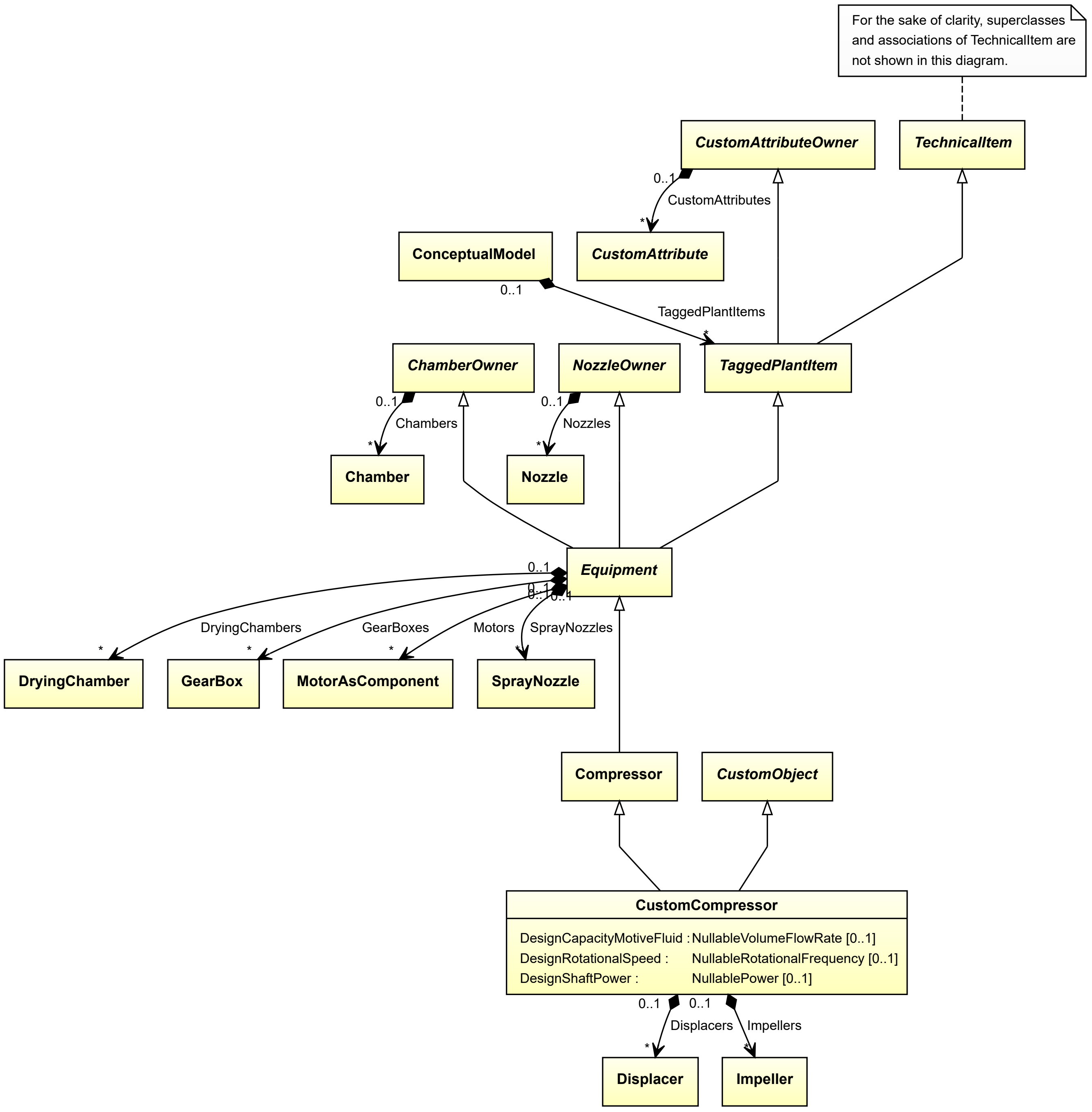
```
<Equipment
    ID="customCentrifuge1"
    ComponentClass="CustomCentrifuge"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomCentrifuge" ...>
  ...
  <Equipment
      ID="sedimentalCentrifugeDrum1"
      ComponentClass="SedimentalCentrifugeDrum"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/SedimentalCentrifugeDrum" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.41. CustomCompressor

7.41.1 Overview

Class

A custom *Compressor*, i.e., a *Compressor* that is not covered by any of the other subclasses of *Compressor* (*AirEjector*, *AxialCompressor*, *CentrifugalCompressor*, *ReciprocatingCompressor*, or *RotaryCompressor*).



Supertypes

- *Compressor*
- *CustomObject*

Attributes (data)

Name	Multiplicity	Type
DesignCapacityMotiveFluid	0..1	*NullableVolumeFlowRate*
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*
DesignShaftPower	0..1	*NullablePower*

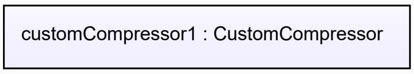
Attributes (composition)

Name	Multiplicity	Type
Displacers	*	*Displacer*
Impellers	*	*Impeller*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: CUSTOM COMPRESSOR
ComponentClass: CustomCompressor
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomCompressor

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="customCompressor1"
    ComponentClass="CustomCompressor"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomCompressor" ...>
  ...
</Equipment>
```

7.41.2 DesignCapacityMotiveFluid

Attribute (data)

The capacity of the volume flow rate for the motive fluid for which the *CustomCompressor* is designed.

Multiplicity: 0..1

Type: *NullableVolumeFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN CAPACITY MOTIVE FLUID

Name: DesignCapacityMotiveFluid
AttributeURI: http://sandbox.dexpi.org/rdl/DesignCapacityMotiveFluid

7.41.3 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *CustomCompressor* is designed.

Multiplicity: 0..1

Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

7.41.4 DesignShaftPower

Attribute (data)

The shaft power for which the *CustomCompressor* is designed.

Multiplicity: 0..1

Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SHAFT POWER
Name: DesignShaftPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignShaftPower

7.41.5 Displacers

Attribute (composition)

The displacers of the *CustomCompressor*.

Multiplicity: *

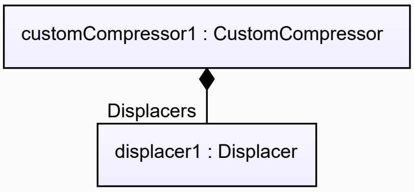
Type: *Displacer*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *Displacer*) is a child of the `<Equipment>` element for the attribute owner (a *CustomCompressor*).

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="customCompressor1"
    ComponentClass="CustomCompressor"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomCompressor" ...>
  ...
  <Equipment
      ID="displacer1"
      ComponentClass="Displacer"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/Displacer" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.41.6 Impellers

Attribute (composition)

The impellers of the *CustomCompressor*.

Multiplicity: *

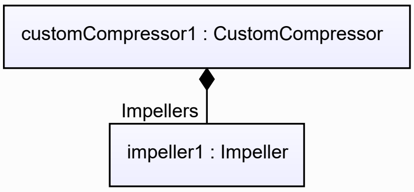
Type: *Impeller*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (an *Impeller*) is a child of the `<Equipment>` element for the attribute owner (a *CustomCompressor*).

Example



Example: Implementation in Proteus Schema

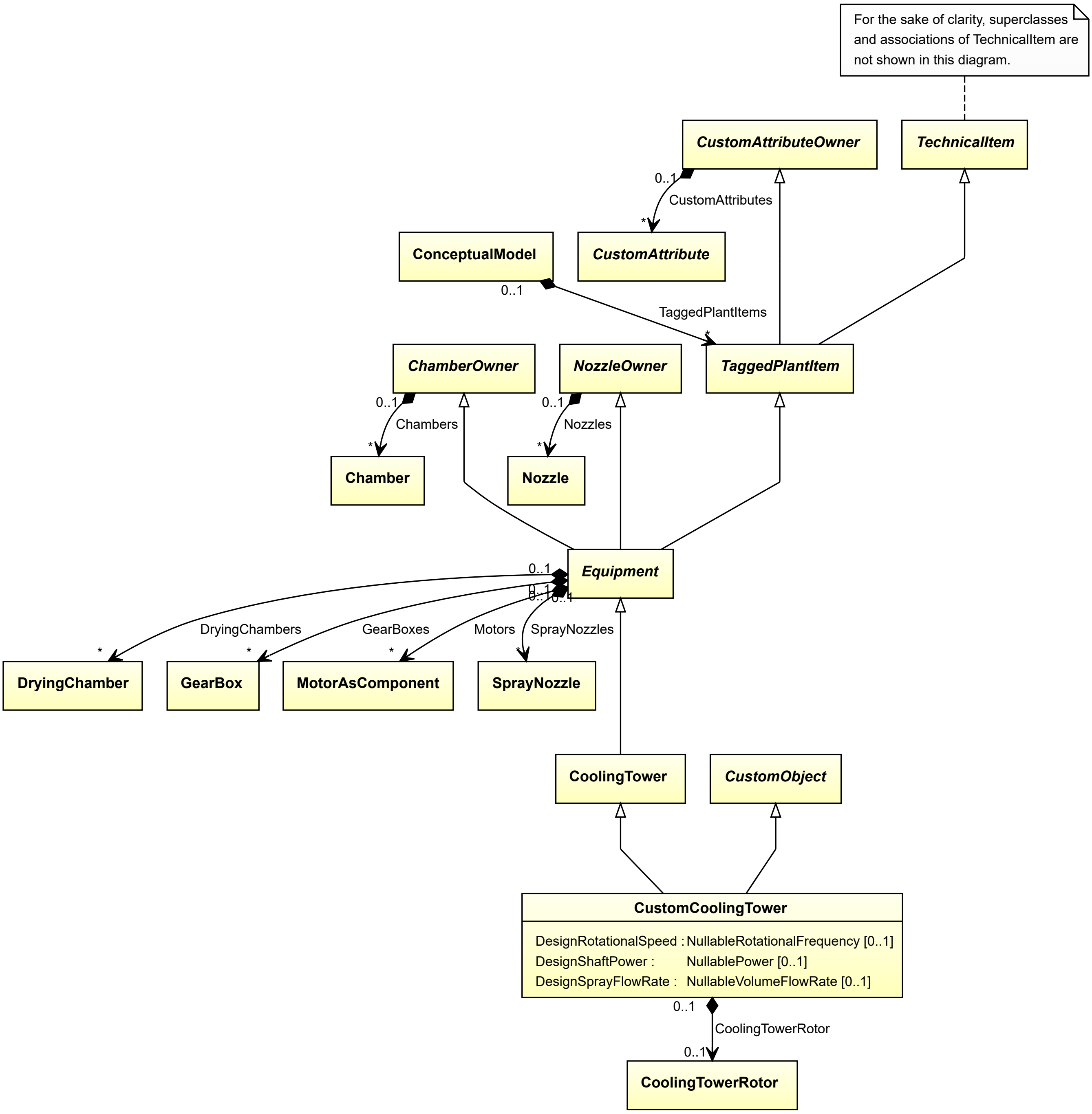
```
<Equipment
    ID="customCompressor1"
    ComponentClass="CustomCompressor"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomCompressor" ...>
  ...
  <Equipment
      ID="impeller1"
      ComponentClass="Impeller"
      ComponentClassURI="http://data.posccaesar.org/rdl/RDS414539" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.42. CustomCoolingTower

7.42.1 Overview

Class

A custom *CoolingTower*, i.e., a *CoolingTower* that is not covered by any of the other subclasses of *CoolingTower* (*DryCoolingTower*, *SprayCooler*, or *WetCoolingTower*).



Supertypes

- *CoolingTower*
- *CustomObject*

Attributes (data)

Name	Multiplicity	Type
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*
DesignShaftPower	0..1	*NullablePower*
DesignSprayFlowRate	0..1	*NullableVolumeFlowRate*

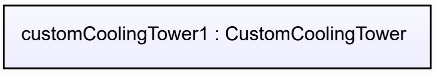
Attributes (composition)

Name	Multiplicity	Type
CoolingTowerRotor	0..1	*CoolingTowerRotor*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: CUSTOM COOLING TOWER
ComponentClass: CustomCoolingTower
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomCoolingTower

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="customCoolingTower1"
    ComponentClass="CustomCoolingTower"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomCoolingTower" ...>
  ...
</Equipment>
```

7.42.2 CoolingTowerRotor

Attribute (composition)

The cooling tower rotor of the *CustomCoolingTower*.

Multiplicity: 0..1

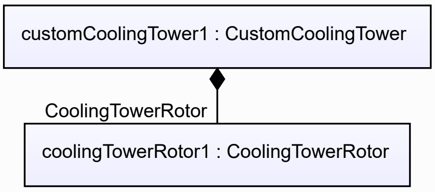
Type: *CoolingTowerRotor*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *CoolingTowerRotor*) is a child of the `<Equipment>` element for the attribute owner (a *CustomCoolingTower*).

Example


customCoolingTower1 : CustomCoolingTower
CoolingTowerRotor
coolingTowerRotor1 : CoolingTowerRotor

Example: Implementation in Proteus Schema

```
<Equipment
    ID="customCoolingTower1"
    ComponentClass="CustomCoolingTower"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomCoolingTower" ...>
  ...
  <Equipment
      ID="coolingTowerRotor1"
      ComponentClass="CoolingTowerRotor"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/CoolingTowerRotor" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.42.3 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *CustomCoolingTower* is designed.

Multiplicity: 0..1

Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

7.42.4 DesignShaftPower

Attribute (data)

The shaft power for which the *CustomCoolingTower* is designed.

Multiplicity: 0..1

Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SHAFT POWER
Name: DesignShaftPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignShaftPower

7.42.5 DesignSprayFlowRate

Attribute (data)

The spray volume flow rate for the motive fluid for which the *CustomCoolingTower* is designed.

Multiplicity: 0..1

Type: *NullableVolumeFlowRate*

Implementation in Proteus Schema

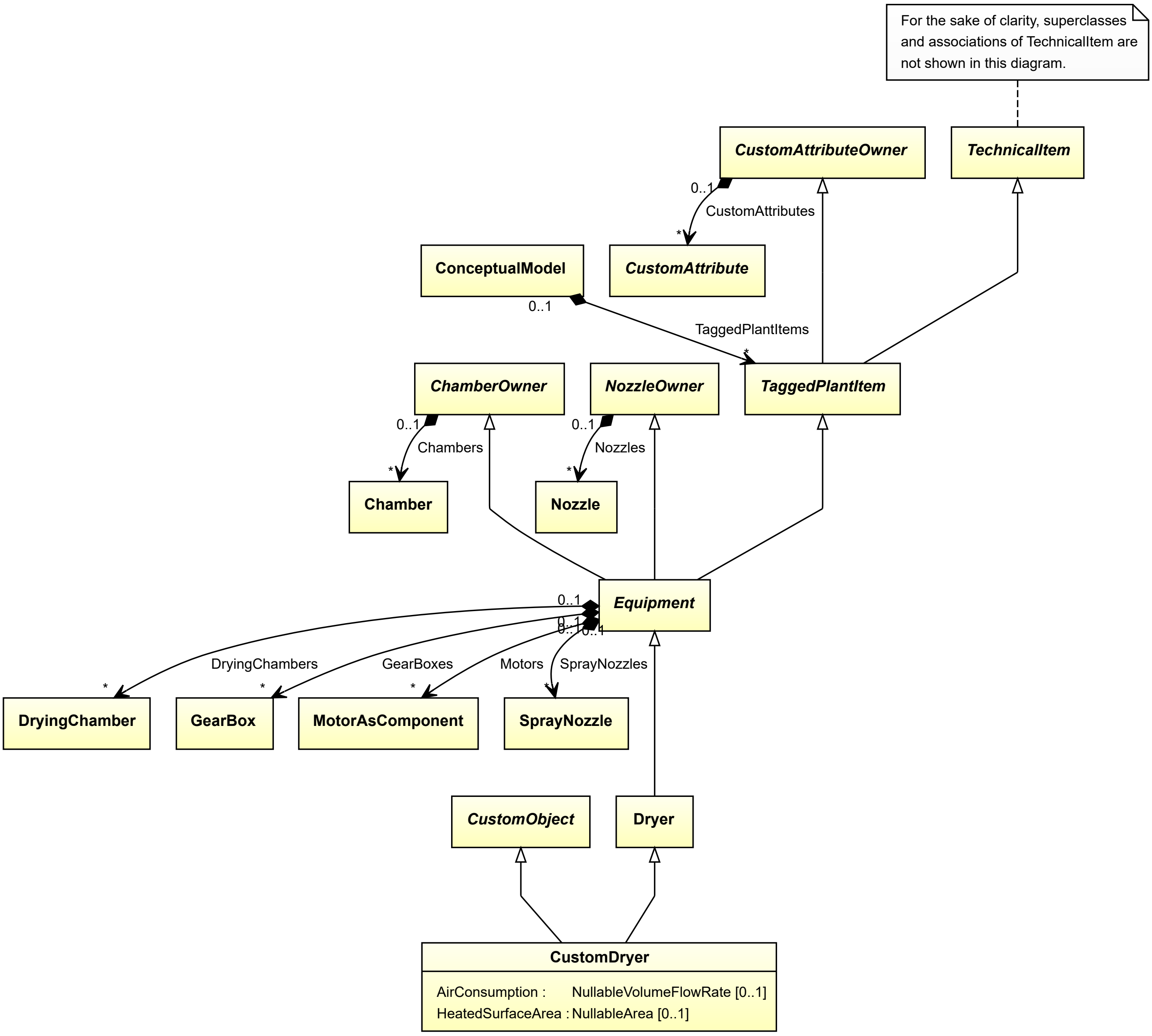
The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SPRAY FLOW RATE
Name: DesignSprayFlowRate
AttributeURI: http://sandbox.dexpi.org/rdl/DesignSprayFlowRate

7.43. CustomDryer

7.43.1 Overview

Class

A custom *Dryer*, i.e., a *Dryer* that is not covered by any of the other subclasses of *Dryer* (*ConvectionDryer* or *HeatedSurfaceDryer*).



Supertypes

- *CustomObject*
- *Dryer*

Attributes (data)

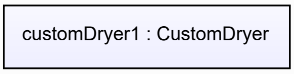
Name	Multiplicity	Type
AirConsumption	0..1	*NullableVolumeFlowRate*
HeatedSurfaceArea	0..1	*NullableArea*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`

RDL reference: CUSTOM DRYER
ComponentClass: CustomDryer
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomDryer

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="customDryer1"
    ComponentClass="CustomDryer"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomDryer" ...>
  ...
</Equipment>
```

7.43.2 AirConsumption

Attribute (data)

The consumed air flow of the *CustomDryer*.

Multiplicity: 0..1

Type: *NullableVolumeFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: AIR CONSUMPTION
Name: AirConsumption
AttributeURI: http://data.posccaesar.org/rdl/RDS5875300

7.43.3 HeatedSurfaceArea

Attribute (data)

The heated surface area of the *CustomDryer*.

Multiplicity: 0..1

Type: *NullableArea*

Implementation in Proteus Schema

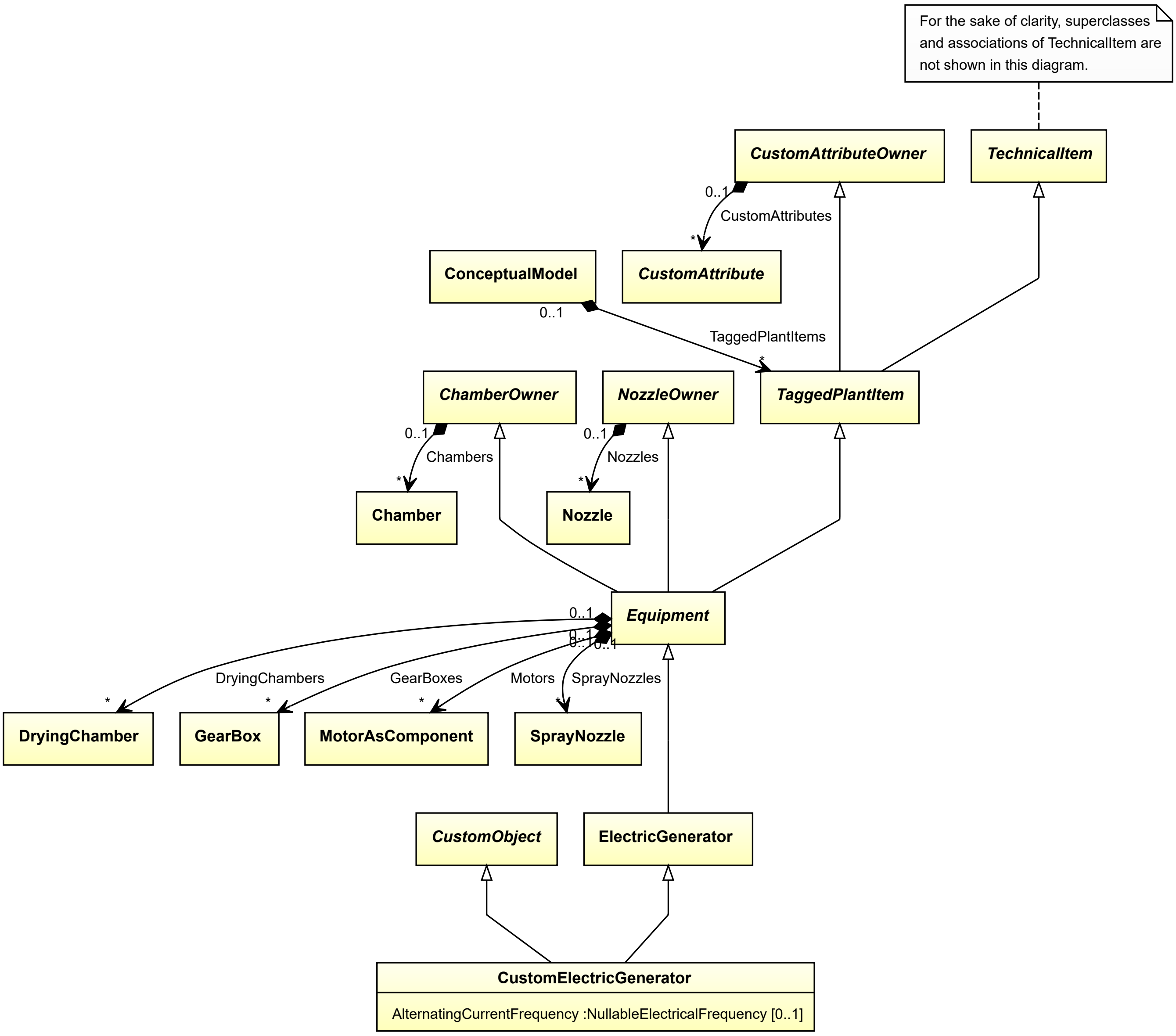
The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: HEATED SURFACE AREA
Name: HeatedSurfaceArea
AttributeURI: http://sandbox.dexpi.org/rdl/HeatedSurfaceArea

7.44. CustomElectricGenerator

7.44.1 Overview

Class

A custom *ElectricGenerator*, i.e., an *ElectricGenerator* that is not covered by any of the other subclasses of *ElectricGenerator* (*AlternatingCurrentGenerator* or *DirectCurrentGenerator*).



Supertypes

- *CustomObject*
- *ElectricGenerator*

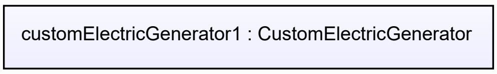
Attributes (data)

Name	Multiplicity	Type
AlternatingCurrentFrequency	0..1	*NullableElectricalFrequency*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: CUSTOM ELECTRIC GENERATOR
ComponentClass: CustomElectricGenerator
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomElectricGenerator

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="customElectricGenerator1"
    ComponentClass="CustomElectricGenerator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomElectricGenerator" ...>
  ...
</Equipment>
```

7.44.2 AlternatingCurrentFrequency

Attribute (data)

The alternating current frequency of the *CustomElectricGenerator*.

Multiplicity: 0..1

Type: *NullableElectricalFrequency*

Implementation in Proteus Schema

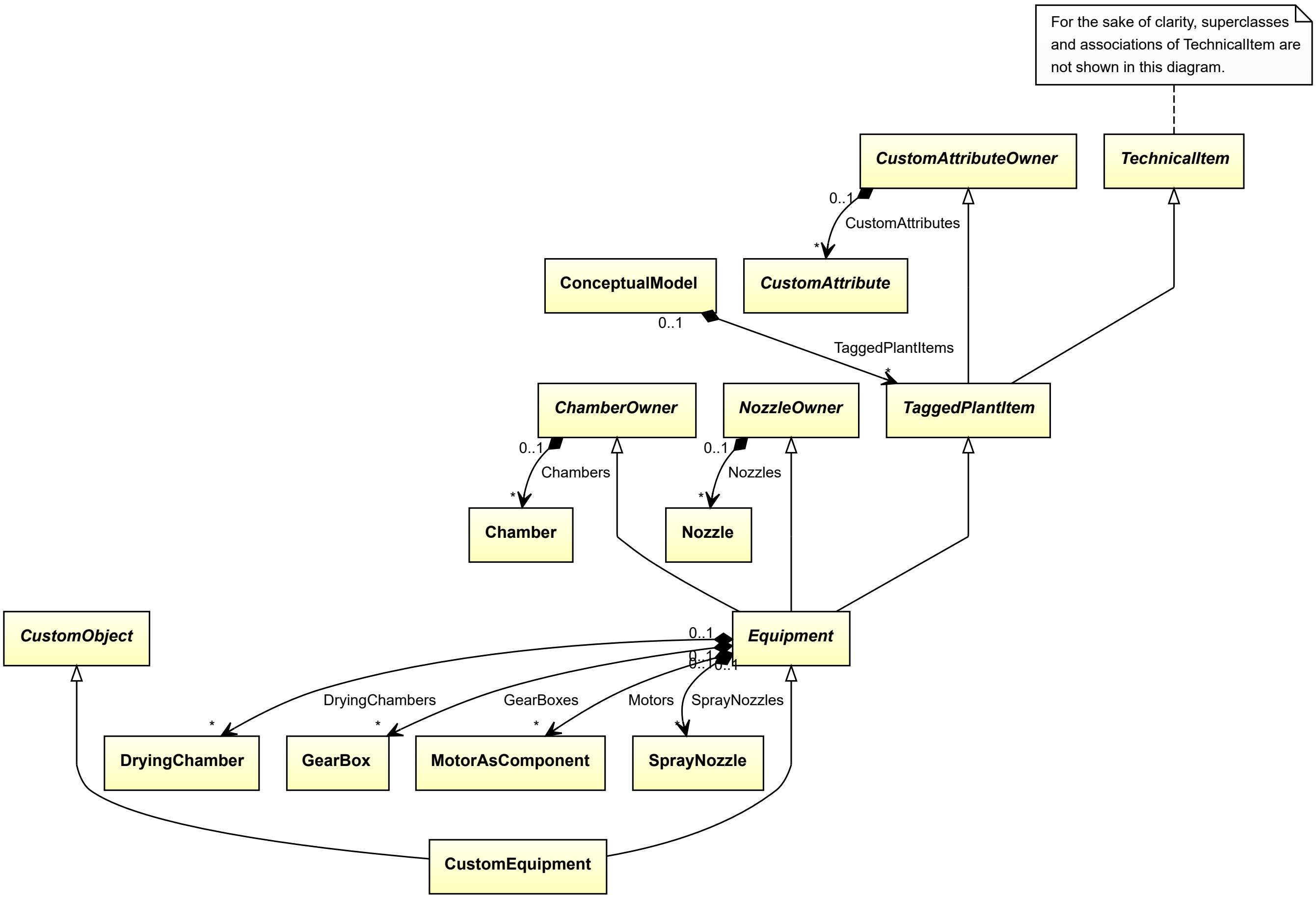
The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: ALTERNATING CURRENT FREQUENCY
Name: AlternatingCurrentFrequency
AttributeURI: http://sandbox.dexpi.org/rdl/AlternatingCurrentFrequency

7.45. CustomEquipment

7.45.1 Overview

Class

A custom *Equipment*, i.e., an *Equipment* that is not covered by any of the other subclasses of *Equipment*.



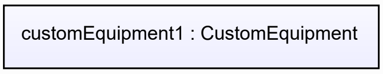
Supertypes

- *CustomObject*
- *Equipment*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: CUSTOM EQUIPMENT
ComponentClass: CustomEquipment
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomEquipment

Example



Example: Implementation in Proteus Schema

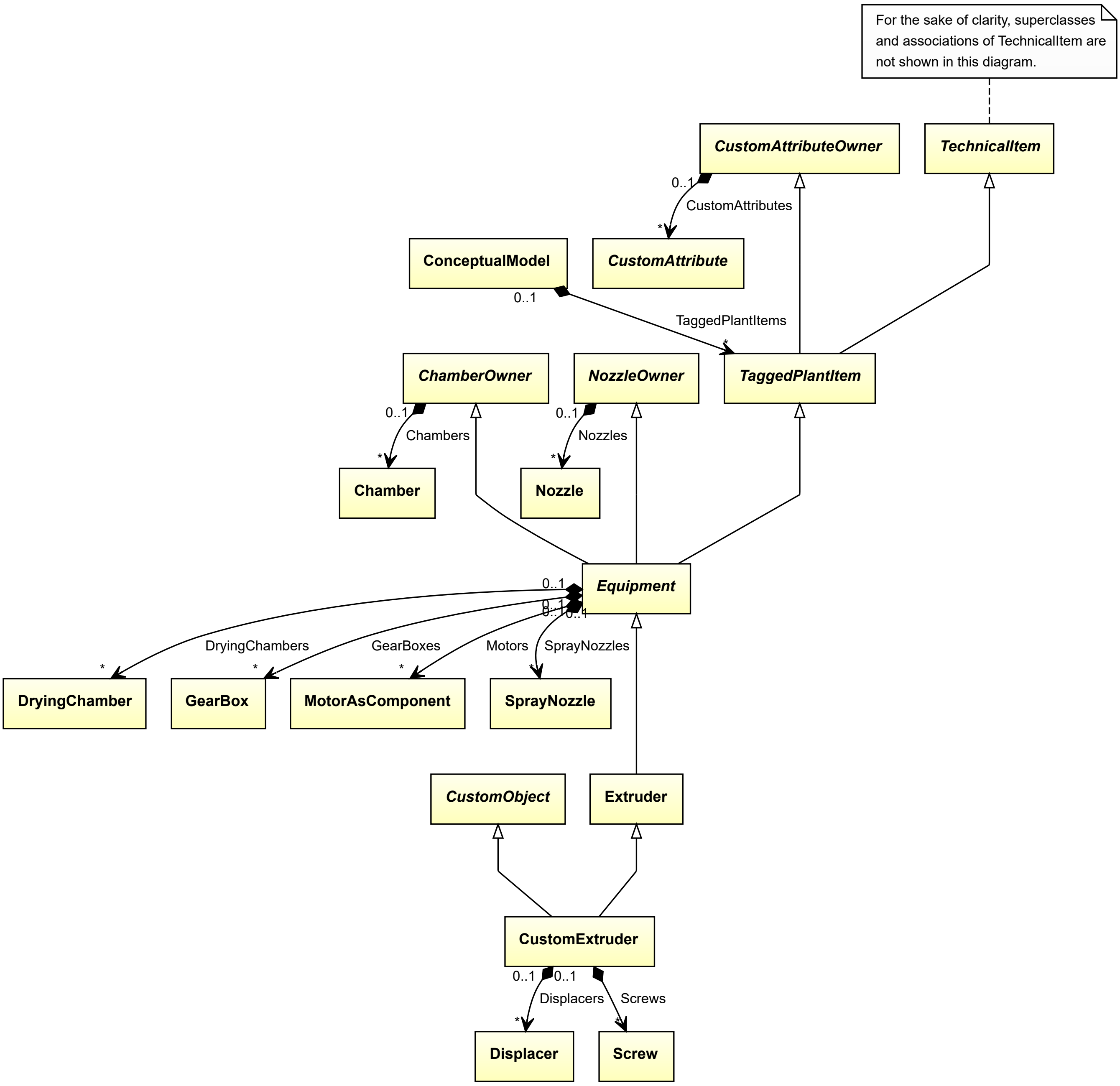
```
<Equipment
    ID="customEquipment1"
    ComponentClass="CustomEquipment"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomEquipment" ...>
  ...
</Equipment>
```

7.46. CustomExtruder

7.46.1 Overview

Class

A custom *Extruder*, i.e., an *Extruder* that is not covered by any of the other subclasses of *Extruder* (*ReciprocatingExtruder* or *RotatingExtruder*).



Supertypes

- *CustomObject*
- *Extruder*

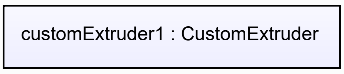
Attributes (composition)

Name	Multiplicity	Type
Displacers	*	*Displacer*
Screws	*	*Screw*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: CUSTOM EXTRUDER
ComponentClass: CustomExtruder
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomExtruder

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="customExtruder1"
    ComponentClass="CustomExtruder"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomExtruder" ...>
  ...
</Equipment>
```

7.46.2 Displacers

Attribute (composition)

The displacers of the *CustomExtruder*.

Multiplicity: *

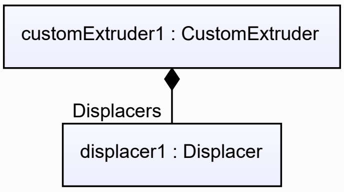
Type: *Displacer*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *Displacer*) is a child of the `<Equipment>` element for the attribute owner (a *CustomExtruder*).

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="customExtruder1"
    ComponentClass="CustomExtruder"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomExtruder" ...>
  ...
  <Equipment
      ID="displacer1"
      ComponentClass="Displacer"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/Displacer" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.46.3 Screws

Attribute (composition)

The screws of the *CustomExtruder*.

Multiplicity: *

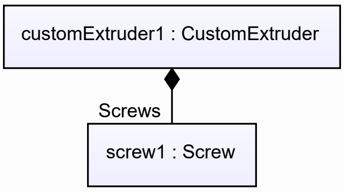
Type: *Screw*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *Screw*) is a child of the `<Equipment>` element for the attribute owner (a *CustomExtruder*).

Example



Example: Implementation in Proteus Schema

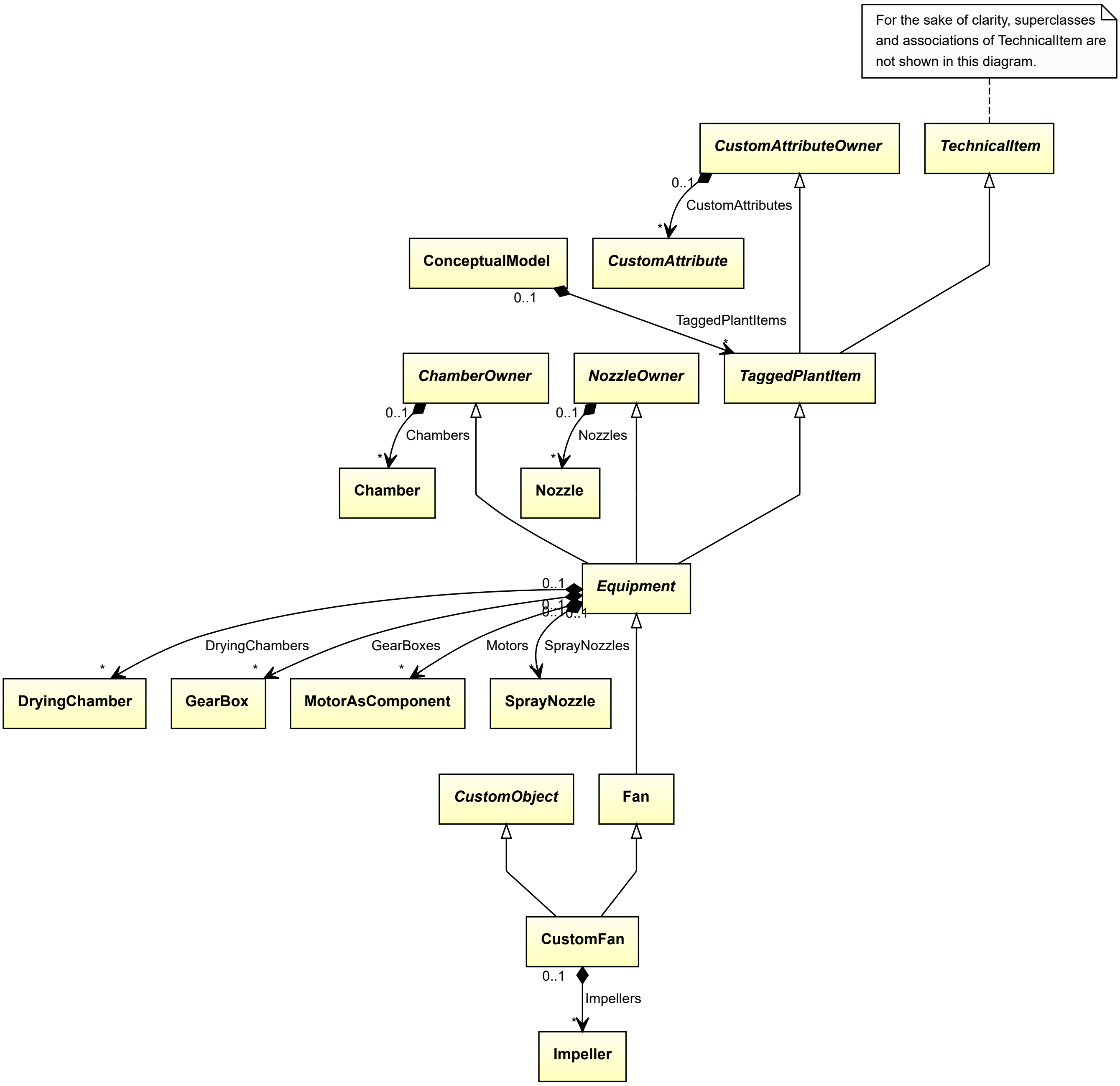
```
<Equipment
    ID="customExtruder1"
    ComponentClass="CustomExtruder"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomExtruder" ...>
  ...
  <Equipment
      ID="screw1"
      ComponentClass="Screw"
      ComponentClassURI="http://data.posccaesar.org/rdl/RDS7219994" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.47. CustomFan

7.47.1 Overview

Class

A custom *Fan*, i.e., a *Fan* that is not covered by any of the other subclasses of *Fan* (*AxialFan* or *RadialFan*).



Supertypes

- *CustomObject*
- *Fan*

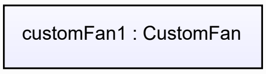
Attributes (composition)

Name	Multiplicity	Type
Impellers	*	*Impeller*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: CUSTOM FAN
ComponentClass: CustomFan
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomFan

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="customFan1"
    ComponentClass="CustomFan"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomFan" ...>
  ...
</Equipment>
```

7.47.2 Impellers

Attribute (composition)

The impellers of the *CustomFan*.

Multiplicity: *

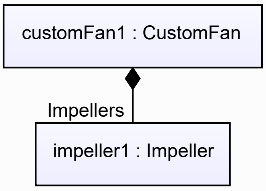
Type: *Impeller*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (an *Impeller*) is a child of the <Equipment> element for the attribute owner (a *CustomFan*).

Example



Example: Implementation in Proteus Schema

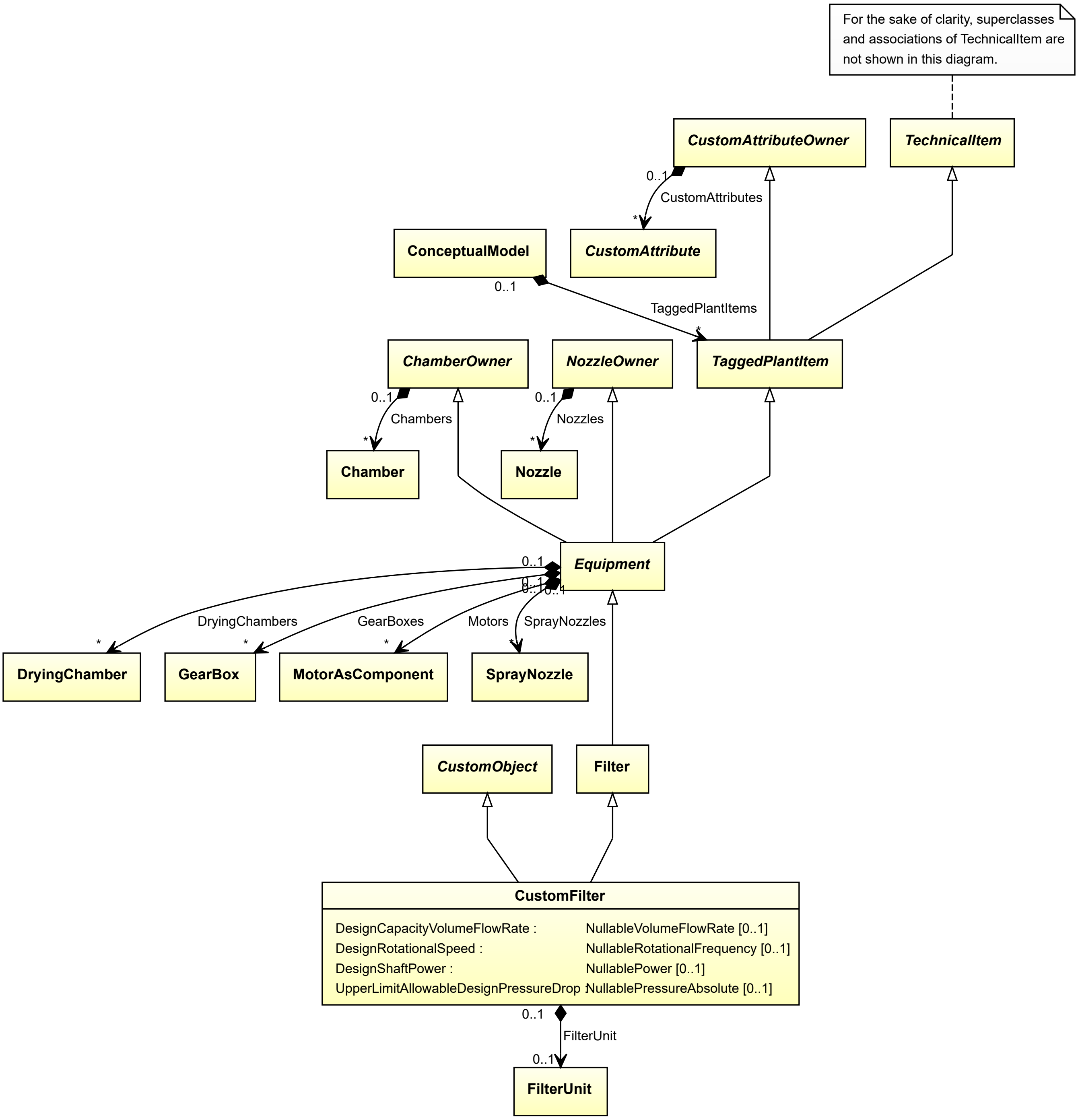
```
<Equipment
    ID="customFan1"
    ComponentClass="CustomFan"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomFan" ...>
  ...
  <Equipment
      ID="impeller1"
      ComponentClass="Impeller"
      ComponentClassURI="http://data.posccaesar.org/rdl/RDS414539" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.48. CustomFilter

7.48.1 Overview

Class

A custom *Filter*, i.e., a *Filter* that is not covered by any of the other subclasses of *Filter* (*GasFilter* or *LiquidFilter*).



Supertypes

- *CustomObject*
- *Filter*

Attributes (data)

Name	Multiplicity	Type
DesignCapacityVolumeFlowRate	0..1	*NullableVolumeFlowRate*
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*
DesignShaftPower	0..1	*NullablePower*
UpperLimitAllowableDesignPressureDrop	0..1	*NullablePressureAbsolute*

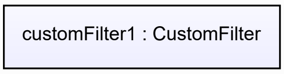
Attributes (composition)

Name	Multiplicity	Type
FilterUnit	0..1	*FilterUnit*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: CUSTOM FILTER
ComponentClass: CustomFilter
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomFilter

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="customFilter1"
    ComponentClass="CustomFilter"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomFilter" ...>
  ...
</Equipment>
```

7.48.2 DesignCapacityVolumeFlowRate

Attribute (data)

The volume flow rate for which the *CustomFilter* is designed.

Multiplicity: 0..1

Type: *NullableVolumeFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN CAPACITY VOLUME FLOW RATE

Name: DesignCapacityVolumeFlowRate
AttributeURI: http://sandbox.dexpi.org/rdl/DesignCapacityVolumeFlowRate

7.48.3 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *CustomFilter* is designed.

Multiplicity: 0..1

Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

7.48.4 DesignShaftPower

Attribute (data)

The shaft power for which the *CustomFilter* is designed.

Multiplicity: 0..1

Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SHAFT POWER
Name: DesignShaftPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignShaftPower

7.48.5 FilterUnit

Attribute (composition)

The filter unit of the *CustomFilter*.

Multiplicity: 0..1

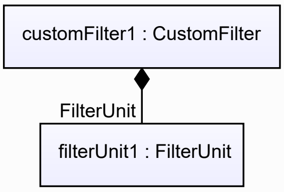
Type: *FilterUnit*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *FilterUnit*) is a child of the `<Equipment>` element for the attribute owner (a *CustomFilter*).

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="customFilter1"
    ComponentClass="CustomFilter"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomFilter" ...>
  ...
  <Equipment
      ID="filterUnit1"
      ComponentClass="FilterUnit"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/FilterUnit" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.48.6 UpperLimitAllowableDesignPressureDrop

Attribute (data)

The upper limit for the pressure drop for which the *CustomFilter* is designed.

Multiplicity: 0..1

Type: *NullablePressureAbsolute*

Implementation in Proteus Schema

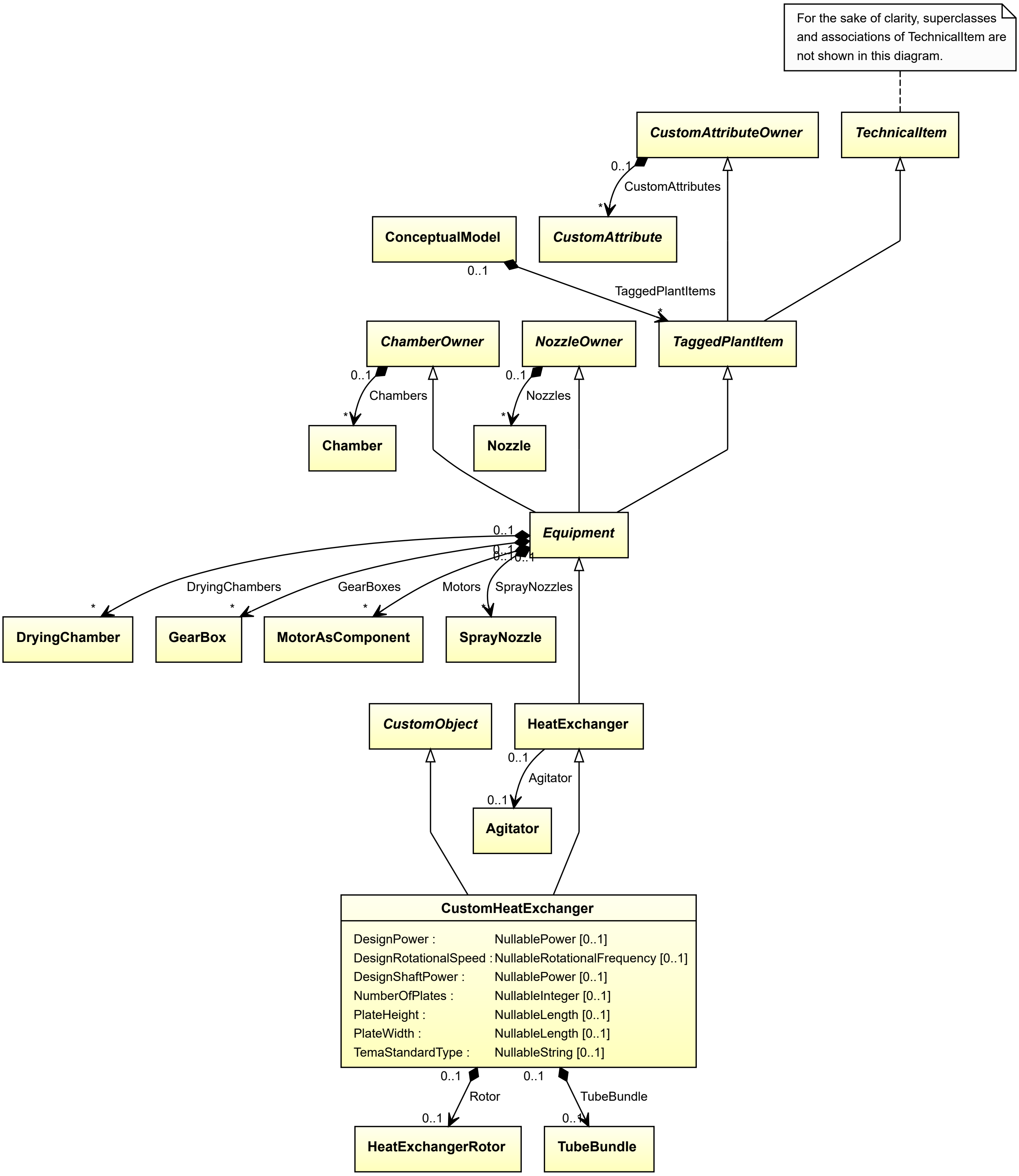
The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT ALLOWABLE DESIGN PRESSURE DROP
Name: UpperLimitAllowableDesignPressureDrop
AttributeURI: http://sandbox.dexpi.org/rdl/UpperLimitAllowableDesignPressureDrop

7.49. CustomHeatExchanger

7.49.1 Overview

Class

A custom *HeatExchanger*, i.e., a *HeatExchanger* that is not covered by any of the other subclasses of *HeatExchanger* (*AirCoolingSystem*, *ElectricHeater*, *PlateHeatExchanger*, *SpiralHeatExchanger*, *ThinFilmEvaporator*, or *TubularHeatExchanger*).


For the sake of clarity, superclasses and associations of TechnicalItem are not shown in this diagram.
CustomAttributeOwner
TechnicalItem
0..1
CustomAttributes
*
ConceptualModel
CustomAttribute
0..1
TaggedPlantItems
*
ChamberOwner
NozzleOwner
TaggedPlantItem
0..1
Chambers
0..1
Nozzles
*
Chamber
*
Nozzle
0..1
Equipment
0..1
0..1
0..1
DryingChambers
GearBoxes
Motors
SprayNozzles
*
*
*
*
DryingChamber
GearBox
MotorAsComponent
SprayNozzle
CustomObject
HeatExchanger
0..1
Agitator
0..1
Agitator
CustomHeatExchanger
DesignPower : NullablePower [0..1]
DesignRotationalSpeed : NullableRotationalFrequency [0..1]
DesignShaftPower : NullablePower [0..1]
NumberOfPlates : NullableInteger [0..1]
PlateHeight : NullableLength [0..1]
PlateWidth : NullableLength [0..1]
TemaStandardType : NullableString [0..1]
0..1
Rotor
0..1
TubeBundle
0..1
0..1
HeatExchangerRotor
TubeBundle

Supertypes

- *CustomObject*
- *HeatExchanger*

Attributes (data)

Name	Multiplicity	Type
DesignPower	0..1	*NullablePower*
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*
DesignShaftPower	0..1	*NullablePower*
NumberOfPlates	0..1	*NullableInteger*
PlateHeight	0..1	*NullableLength*
PlateWidth	0..1	*NullableLength*
TemaStandardType	0..1	*NullableString*

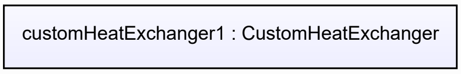
Attributes (composition)

Name	Multiplicity	Type
Rotor	0..1	*HeatExchangerRotor*
TubeBundle	0..1	*TubeBundle*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: CUSTOM HEAT EXCHANGER
ComponentClass: CustomHeatExchanger
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomHeatExchanger

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="customHeatExchanger1"
    ComponentClass="CustomHeatExchanger"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomHeatExchanger" ...>
  ...
</Equipment>
```

7.49.2 DesignPower

Attribute (data)

The power for which the *CustomHeatExchanger* is designed.

Multiplicity: 0..1

Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN POWER
Name: DesignPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignPower

7.49.3 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *CustomHeatExchanger* is designed.

Multiplicity: 0..1

Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

7.49.4 DesignShaftPower

Attribute (data)

The shaft power for which the *CustomHeatExchanger* is designed.

Multiplicity: 0..1

Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SHAFT POWER
Name: DesignShaftPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignShaftPower

7.49.5 NumberOfPlates

Attribute (data)

The number of plates in the *CustomHeatExchanger*.

Multiplicity: 0..1

Type: *NullableInteger*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for integer values*.
RDL reference: NUMBER OF PLATES
Name: NumberOfPlates
AttributeURI: http://data.posccaesar.org/rdl/RDS364229

7.49.6 PlateHeight

Attribute (data)

The height of the plates in the *CustomHeatExchanger*.

Multiplicity: 0..1

Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: PLATE HEIGHT
Name: PlateHeight
AttributeURI: http://sandbox.dexpi.org/rdl/PlateHeight

7.49.7 PlateWidth

Attribute (data)

The width of the plates in the *CustomHeatExchanger*.

Multiplicity: 0..1

Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: PLATE WIDTH
Name: PlateWidth
AttributeURI: http://sandbox.dexpi.org/rdl/PlateWidth

7.49.8 Rotor

Attribute (composition)

The rotor of the *CustomHeatExchanger*.

Multiplicity: 0..1

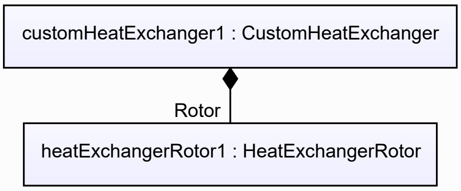
Type: *HeatExchangerRotor*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *HeatExchangerRotor*) is a child of the `<Equipment>` element for the attribute owner (a *CustomHeatExchanger*).

Example


customHeatExchanger1 : CustomHeatExchanger
Rotor
heatExchangerRotor1 : HeatExchangerRotor

Example: Implementation in Proteus Schema

```
<Equipment
    ID="customHeatExchanger1"
    ComponentClass="CustomHeatExchanger"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomHeatExchanger" ...>
  ...
  <Equipment
      ID="heatExchangerRotor1"
      ComponentClass="HeatExchangerRotor"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/HeatExchangerRotor" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.49.9 TemaStandardType

Attribute (data)

The type of the *CustomHeatExchanger* according to the Tubular Exchanger Manufacturers Association, Inc. (TEMA, http://www.tema.org). This is a three-letter code.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: TEMA STANDARD TYPE ASSIGNMENT CLASS
Name: TemaStandardTypeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/TemaStandardTypeAssignmentClass

7.49.10 TubeBundle

Attribute (composition)

The tube bundle of the *CustomHeatExchanger*.

Multiplicity: 0..1

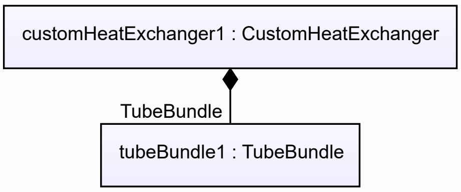
Type: *TubeBundle*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *TubeBundle*) is a child of the `<Equipment>` element for the attribute owner (a *CustomHeatExchanger*).

Example



Example: Implementation in Proteus Schema

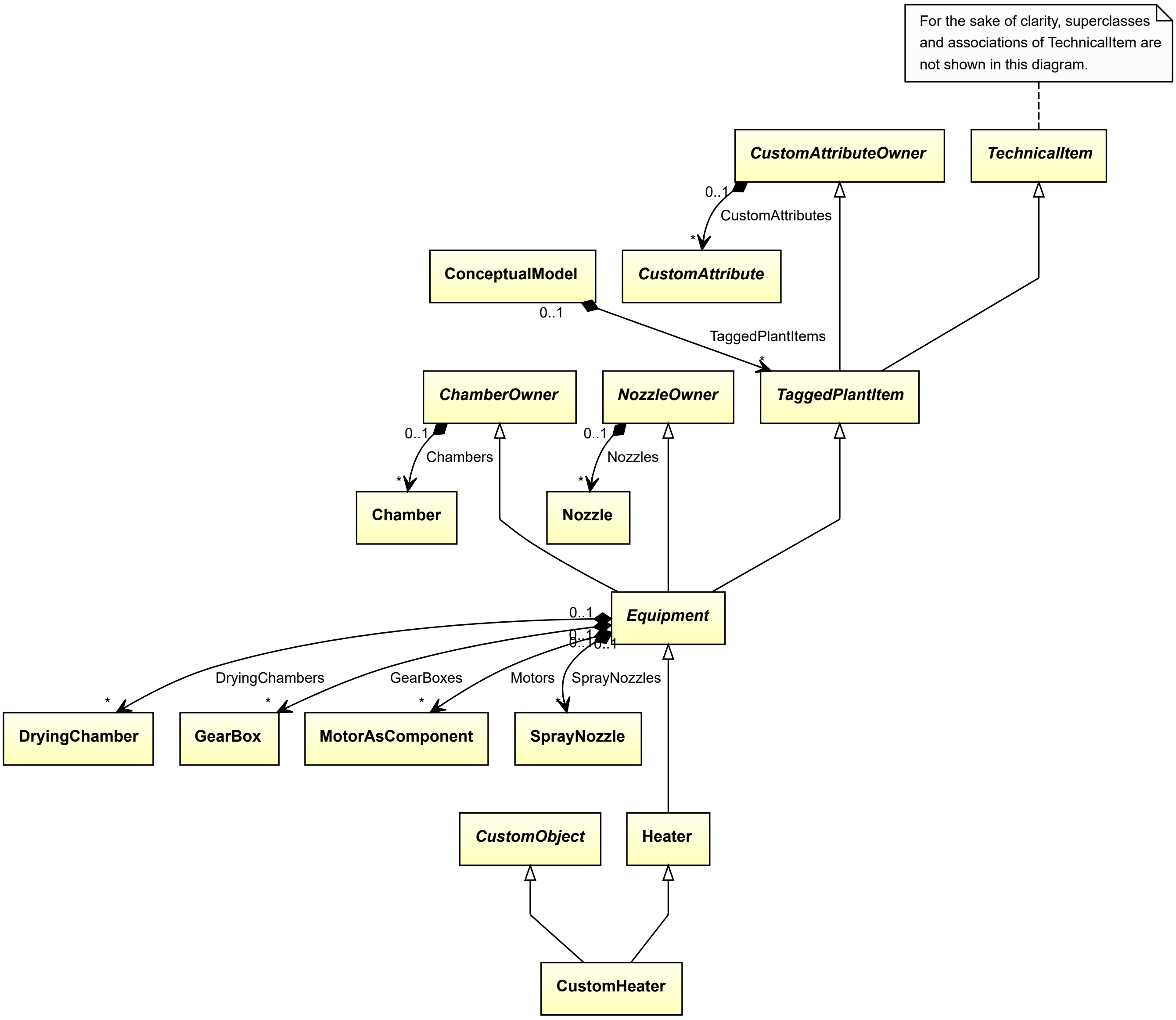
```
<Equipment
    ID="customHeatExchanger1"
    ComponentClass="CustomHeatExchanger"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomHeatExchanger" ...>
  ...
  <Equipment
      ID="tubeBundle1"
      ComponentClass="TubeBundle"
      ComponentClassURI="http://data.posccaesar.org/rdl/RDS415259" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.50. CustomHeater

7.50.1 Overview

Class

A custom *Heater*, i.e., a *Heater* that is not covered by any of the other subclasses of *Heater* (*Boiler*, *Furnace*, or *SteamGenerator*).



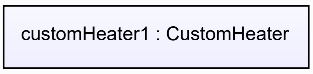
Supertypes

- *CustomObject*
- *Heater*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: CUSTOM HEATER
ComponentClass: CustomHeater
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomHeater

Example



Example: Implementation in Proteus Schema

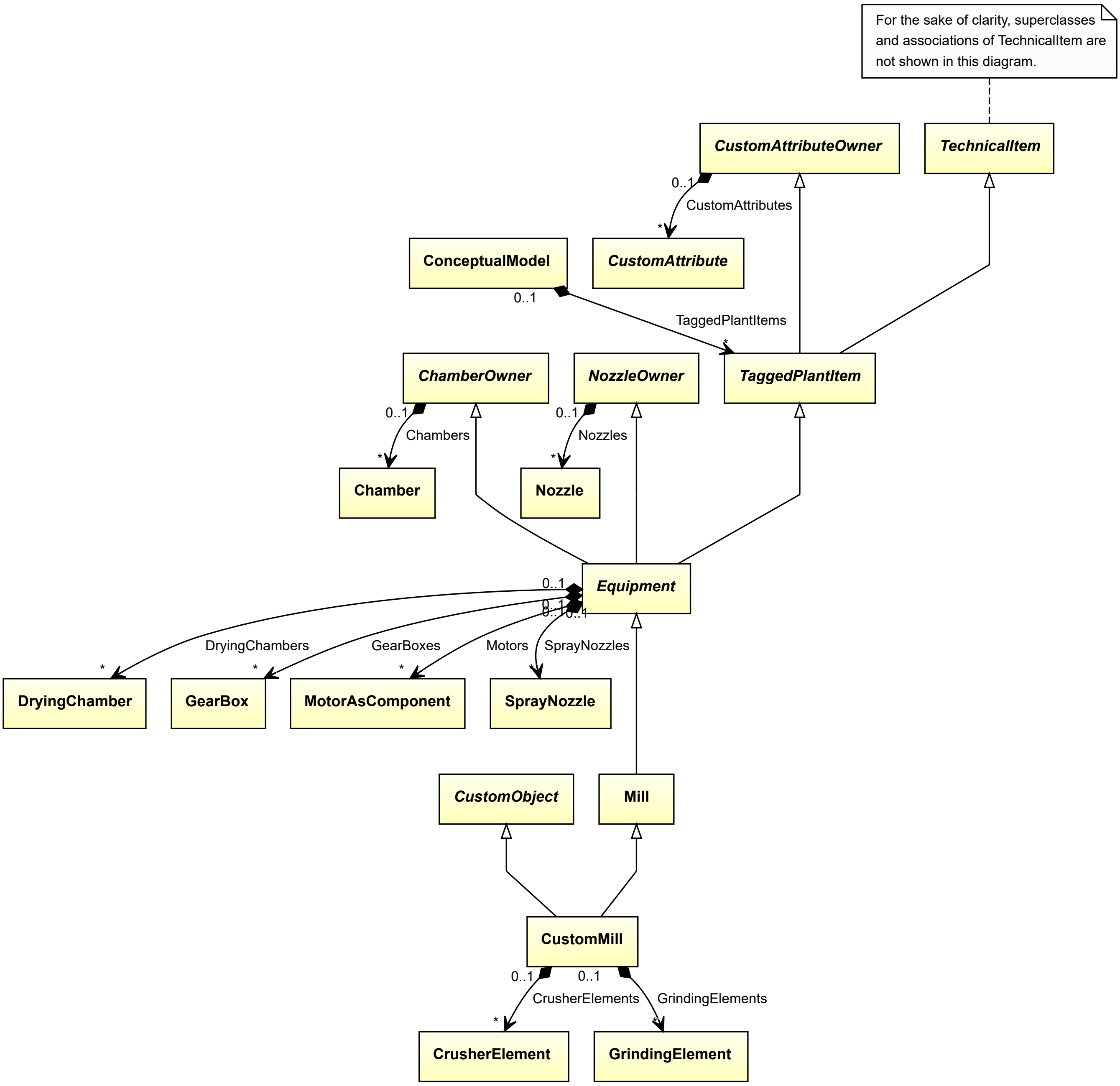
```
<Equipment
    ID="customHeater1"
    ComponentClass="CustomHeater"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomHeater" ...>
  ...
</Equipment>
```

7.51. CustomMill

7.51.1 Overview

Class

A custom *Mill*, i.e., a *Mill* that is not covered by any of the other subclasses of *Mill* (*Crusher* or *Grinder*).



Supertypes

- *CustomObject*
- *Mill*

Attributes (composition)

Name	Multiplicity	Type
CrusherElements	*	*CrusherElement*
GrindingElements	*	*GrindingElement*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: CUSTOM MILL
ComponentClass: CustomMill
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomMill

Example

customMill1 : CustomMill

Example: Implementation in Proteus Schema

```
<Equipment
    ID="customMill1"
    ComponentClass="CustomMill"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomMill" ...>
  ...
</Equipment>
```

7.51.2 CrusherElements

Attribute (composition)

The crusher elements of the *CustomMill*.

Multiplicity: *

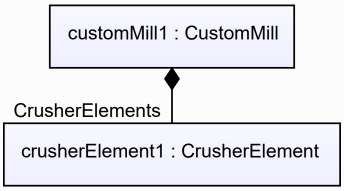
Type: *CrusherElement*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *CrusherElement*) is a child of the `<Equipment>` element for the attribute owner (a *CustomMill*).

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="customMill1"
    ComponentClass="CustomMill"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomMill" ...>
  ...
  <Equipment
      ID="crusherElement1"
      ComponentClass="CrusherUnit"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/CrusherUnit" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.51.3 GrindingElements

Attribute (composition)

The grinding elements of the *CustomMill*.

Multiplicity: *

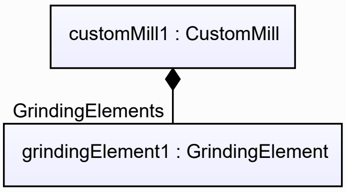
Type: *GrindingElement*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *GrindingElement*) is a child of the `<Equipment>` element for the attribute owner (a *CustomMill*).

Example



Example: Implementation in Proteus Schema

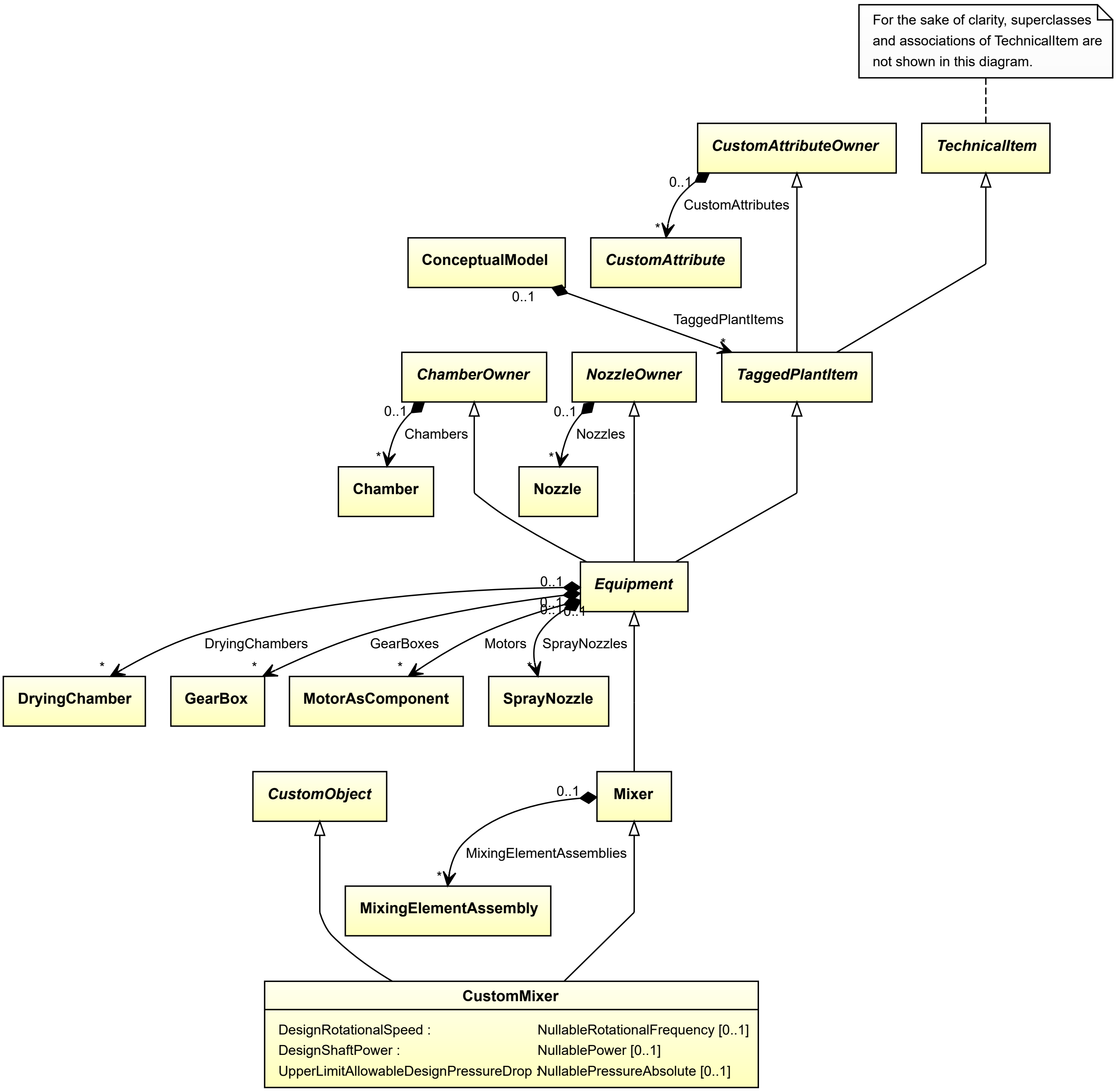
```
<Equipment
    ID="customMill1"
    ComponentClass="CustomMill"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomMill" ...>
  ...
  <Equipment
      ID="grindingElement1"
      ComponentClass="GrindingElement"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/GrindingElement" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.52. CustomMixer

7.52.1 Overview

Class

A custom *Mixer*, i.e., a *Mixer* that is not covered by any of the other subclasses of *Mixer* (*Kneader*, *RotaryMixer*, or *StaticMixer*).



Supertypes

- *CustomObject*
- *Mixer*

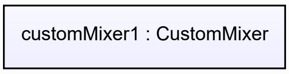
Attributes (data)

Name	Multiplicity	Type
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*
DesignShaftPower	0..1	*NullablePower*
UpperLimitAllowableDesignPressureDrop	0..1	*NullablePressureAbsolute*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: CUSTOM MIXER
ComponentClass: CustomMixer
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomMixer

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="customMixer1"
    ComponentClass="CustomMixer"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomMixer" ...>
  ...
</Equipment>
```

7.52.2 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *CustomMixer* is designed.

Multiplicity: 0..1

Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

7.52.3 DesignShaftPower

Attribute (data)

The shaft power for which the *CustomMixer* is designed.

Multiplicity: 0..1

Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SHAFT POWER
Name: DesignShaftPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignShaftPower

7.52.4 UpperLimitAllowableDesignPressureDrop

Attribute (data)

The upper limit for the pressure drop for which the *CustomMixer* is designed.

Multiplicity: 0..1

Type: *NullablePressureAbsolute*

Implementation in Proteus Schema

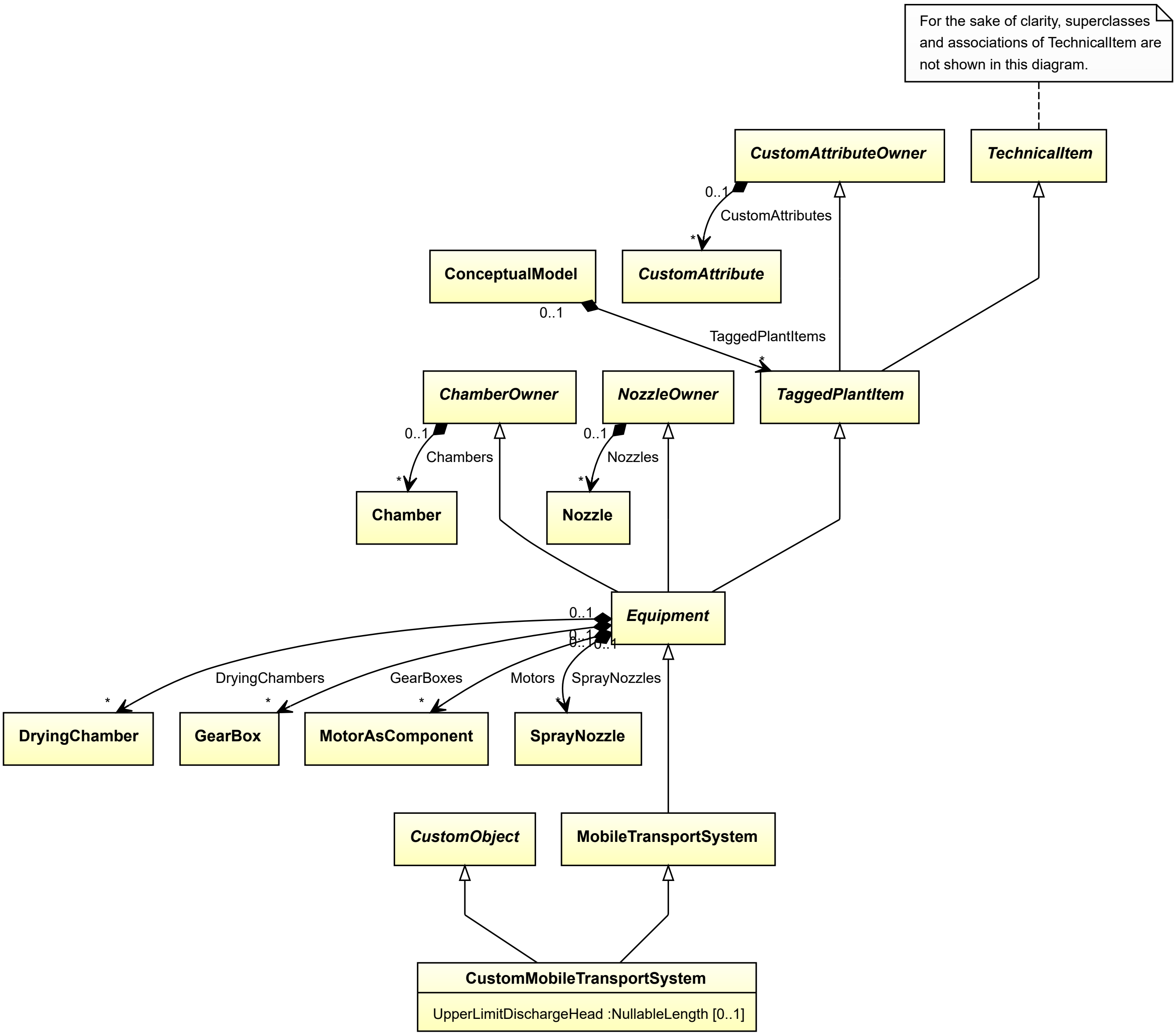
The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT ALLOWABLE DESIGN PRESSURE DROP
Name: UpperLimitAllowableDesignPressureDrop
AttributeURI: http://sandbox.dexpi.org/rdl/UpperLimitAllowableDesignPressureDrop

7.53. CustomMobileTransportSystem

7.53.1 Overview

Class

A custom *MobileTransportSystem*, i.e., a *MobileTransportSystem* that is not covered by any of the other subclasses of *MobileTransportSystem* (*ForkliftTruck*, *RailWaggon*, *Ship*, *TransportableContainer*, or *Truck*).



Supertypes

- *CustomObject*
- *MobileTransportSystem*

Attributes (data)

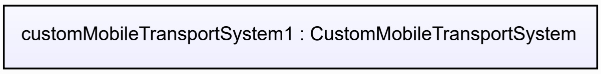
Name	Multiplicity	Type
UpperLimitDischargeHead	0..1	*NullableLength*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: CUSTOM MOBILE TRANSPORT SYSTEM
ComponentClass: CustomMobileTransportSystem

ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomMobileTransportSystem

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="customMobileTransportSystem1"
    ComponentClass="CustomMobileTransportSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomMobileTransportSystem" ...>
  ...
</Equipment>
```

7.53.2 UpperLimitDischargeHead

Attribute (data)

The upper limit for the discharge head of the *CustomMobileTransportSystem*.

Multiplicity: 0..1

Type: *NullableLength*

Implementation in Proteus Schema

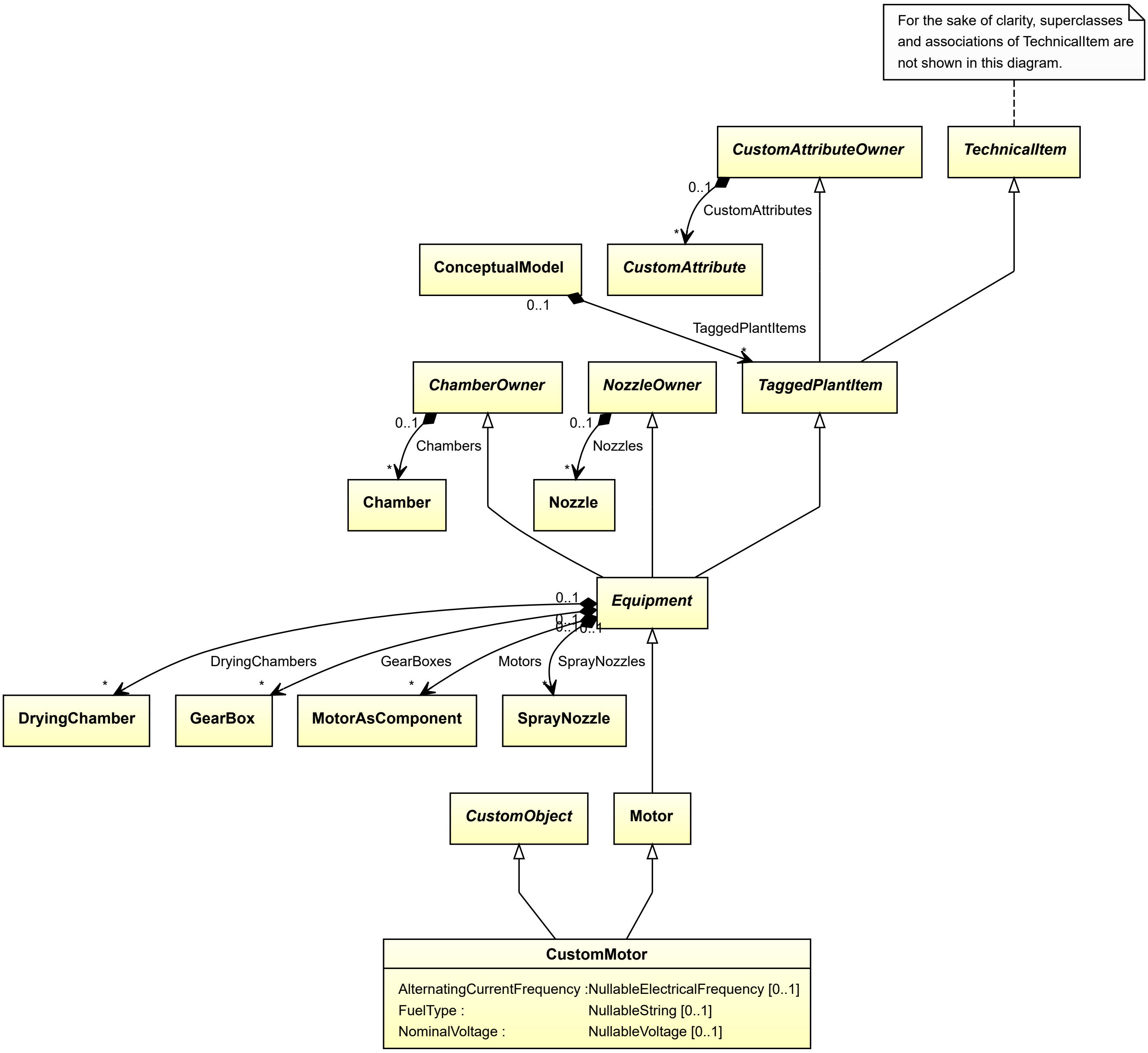
The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT DISCHARGE HEAD
Name: UpperLimitDischargeHead
AttributeURI: http://sandbox.dexpi.org/rdl/UpperLimitDischargeHead

7.54. CustomMotor

7.54.1 Overview

Class

A custom *Motor*, i.e., a *Motor* that is not covered by any of the other subclasses of *Motor* (*AlternatingCurrentMotor*, *CombustionEngine*, or *DirectCurrentMotor*).



Supertypes

- *CustomObject*
- *Motor*

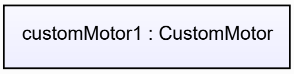
Attributes (data)

Name	Multiplicity	Type
AlternatingCurrentFrequency	0..1	*NullableElectricalFrequency*
FuelType	0..1	*NullableString*
NominalVoltage	0..1	*NullableVoltage*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: CUSTOM MOTOR
ComponentClass: CustomMotor
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomMotor

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="customMotor1"
    ComponentClass="CustomMotor"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomMotor" ...>
  ...
</Equipment>
```

7.54.2 AlternatingCurrentFrequency

Attribute (data)

The alternating current frequency of the *CustomMotor*.

Multiplicity: 0..1

Type: *NullableElectricalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: ALTERNATING CURRENT FREQUENCY
Name: AlternatingCurrentFrequency
AttributeURI: http://sandbox.dexpi.org/rdl/AlternatingCurrentFrequency

7.54.3 FuelType

Attribute (data)

The fuel type of the *CustomMotor*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: FUEL TYPE
Name: FuelType
AttributeURI: http://sandbox.dexpi.org/rdl/FuelType

7.54.4 NominalVoltage

Attribute (data)

The nominal voltage of the *CustomMotor*.

Multiplicity: 0..1

Type: *NullableVoltage*

Implementation in Proteus Schema

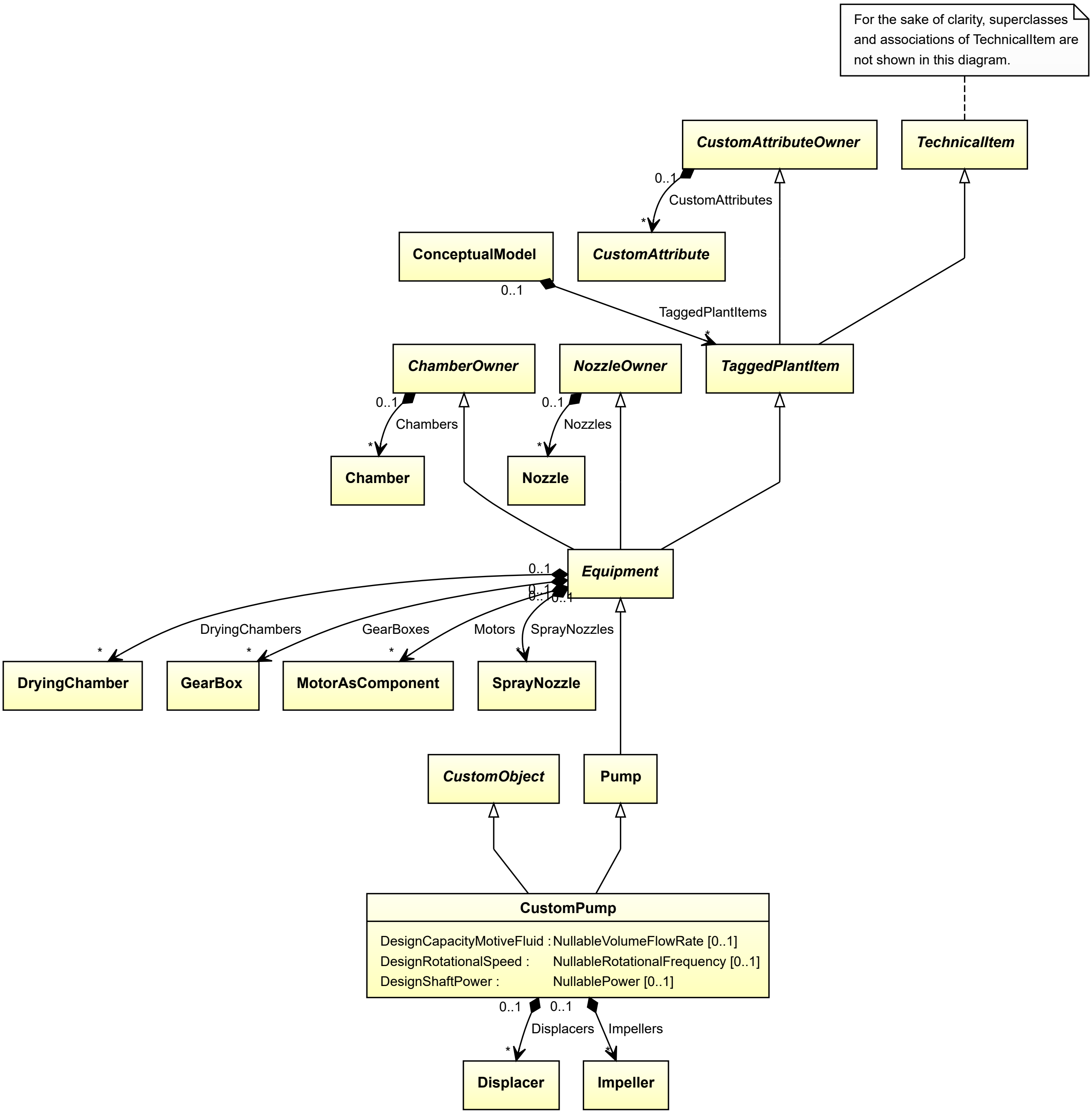
The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: NOMINAL VOLTAGE
Name: NominalVoltage
AttributeURI: http://data.posccaesar.org/rdl/RDS369449

7.55. CustomPump

7.55.1 Overview

Class

A custom *Pump*, i.e., a *Pump* that is not covered by any of the other subclasses of *Pump* (*CentrifugalPump*, *EjectorPump*, *ReciprocatingPump*, or *RotaryPump*).



Supertypes

- *CustomObject*
- *Pump*

Attributes (data)

Name	Multiplicity	Type
DesignCapacityMotiveFluid	0..1	*NullableVolumeFlowRate*
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*
DesignShaftPower	0..1	*NullablePower*

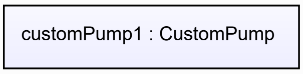
Attributes (composition)

Name	Multiplicity	Type
Displacers	*	*Displacer*
Impellers	*	*Impeller*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: CUSTOM PUMP
ComponentClass: CustomPump
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomPump

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="customPump1"
    ComponentClass="CustomPump"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomPump" ...>
  ...
</Equipment>
```

7.55.2 DesignCapacityMotiveFluid

Attribute (data)

The capacity of the volume flow rate for the motive fluid for which the *CustomPump* is designed.

Multiplicity: 0..1

Type: *NullableVolumeFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN CAPACITY MOTIVE FLUID

Name: DesignCapacityMotiveFluid
AttributeURI: http://sandbox.dexpi.org/rdl/DesignCapacityMotiveFluid

7.55.3 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *CustomPump* is designed.

Multiplicity: 0..1

Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

7.55.4 DesignShaftPower

Attribute (data)

The shaft power for which the *CustomPump* is designed.

Multiplicity: 0..1

Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SHAFT POWER
Name: DesignShaftPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignShaftPower

7.55.5 Displacers

Attribute (composition)

The displacers of the *CustomPump*.

Multiplicity: *

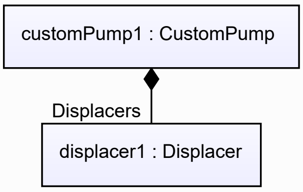
Type: *Displacer*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *Displacer*) is a child of the `<Equipment>` element for the attribute owner (a *CustomPump*).

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="customPump1"
    ComponentClass="CustomPump"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomPump" ...>
  ...
  <Equipment
      ID="displacer1"
      ComponentClass="Displacer"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/Displacer" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.55.6 Impellers

Attribute (composition)

The impellers of the *CustomPump*.

Multiplicity: *

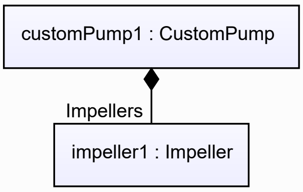
Type: *Impeller*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (an *Impeller*) is a child of the `<Equipment>` element for the attribute owner (a *CustomPump*).

Example



Example: Implementation in Proteus Schema

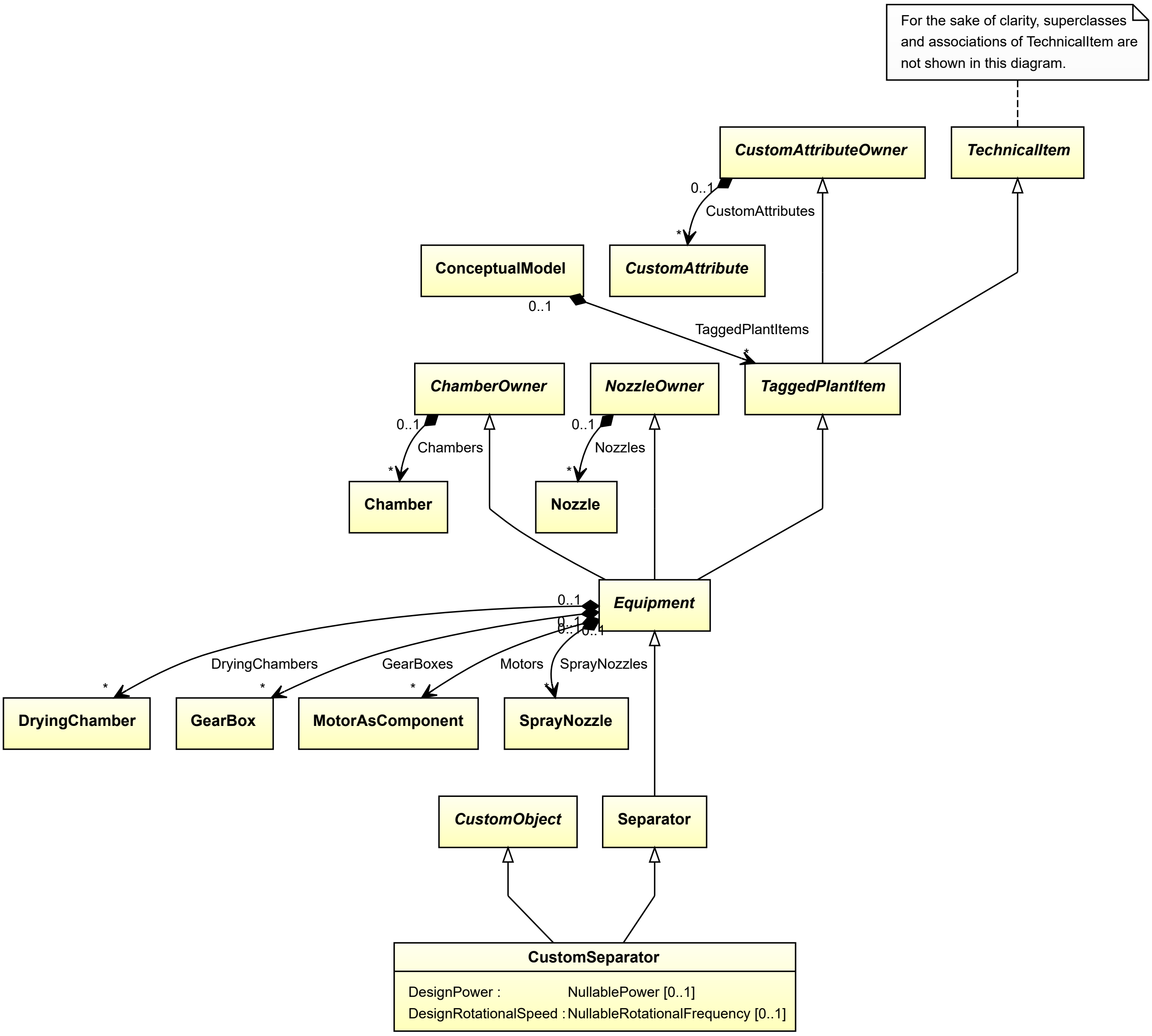
```
<Equipment
    ID="customPump1"
    ComponentClass="CustomPump"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomPump" ...>
  ...
  <Equipment
      ID="impeller1"
      ComponentClass="Impeller"
      ComponentClassURI="http://data.posccaesar.org/rdl/RDS414539" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.56. CustomSeparator

7.56.1 Overview

Class

A custom *Separator*, i.e., a *Separator* that is not covered by any of the other subclasses of *Separator* (*ElectricalSeparator*, *GravitationalSeparator*, *MechanicalSeparator*, or *ScrubbingSeparator*).



Supertypes

- *CustomObject*
- *Separator*

Attributes (data)

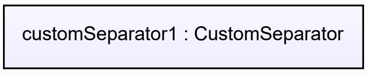
Name	Multiplicity	Type
DesignPower	0..1	*NullablePower*
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`

RDL reference: CUSTOM SEPARATOR
ComponentClass: CustomSeparator
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomSeparator

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="customSeparator1"
    ComponentClass="CustomSeparator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomSeparator" ...>
  ...
</Equipment>
```

7.56.2 DesignPower

Attribute (data)

The power for which the *CustomSeparator* is designed.

Multiplicity: 0..1

Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN POWER
Name: DesignPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignPower

7.56.3 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *CustomSeparator* is designed.

Multiplicity: 0..1

Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

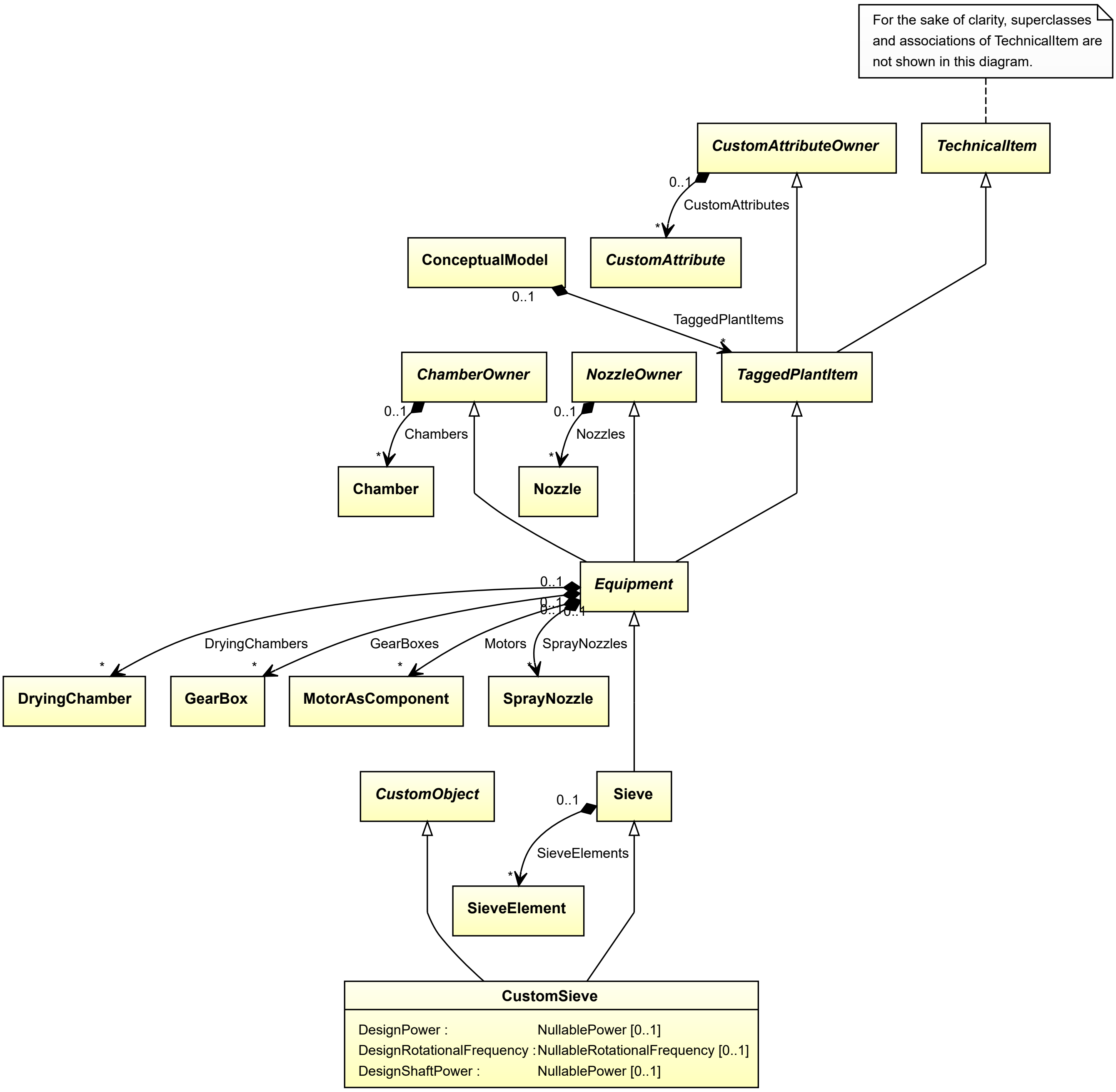
The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

7.57. CustomSieve

7.57.1 Overview

Class

A custom *Sieve*, i.e., a *Sieve* that is not covered by any of the other subclasses of *Sieve* (*RevolvingSieve*, *StationarySieve*, or *VibratingSieve*).



Supertypes

- *CustomObject*
- *Sieve*

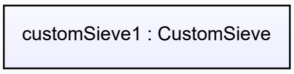
Attributes (data)

Name	Multiplicity	Type
DesignPower	0..1	*NullablePower*
DesignRotationalFrequency	0..1	*NullableRotationalFrequency*
DesignShaftPower	0..1	*NullablePower*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: CUSTOM SIEVE
ComponentClass: CustomSieve
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomSieve

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="customSieve1"
    ComponentClass="CustomSieve"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomSieve" ...>
  ...
</Equipment>
```

7.57.2 DesignPower

Attribute (data)

The power for which the *CustomSieve* is designed.

Multiplicity: 0..1

Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN POWER
Name: DesignPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignPower

7.57.3 DesignRotationalFrequency

Attribute (data)

The rotational frequency for which the *CustomSieve* is designed.

Multiplicity: 0..1

Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL FREQUENCY
Name: DesignRotationalFrequency
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalFrequency

7.57.4 DesignShaftPower

Attribute (data)

The shaft power for which the *CustomSieve* is designed.

Multiplicity: 0..1

Type: *NullablePower*

Implementation in Proteus Schema

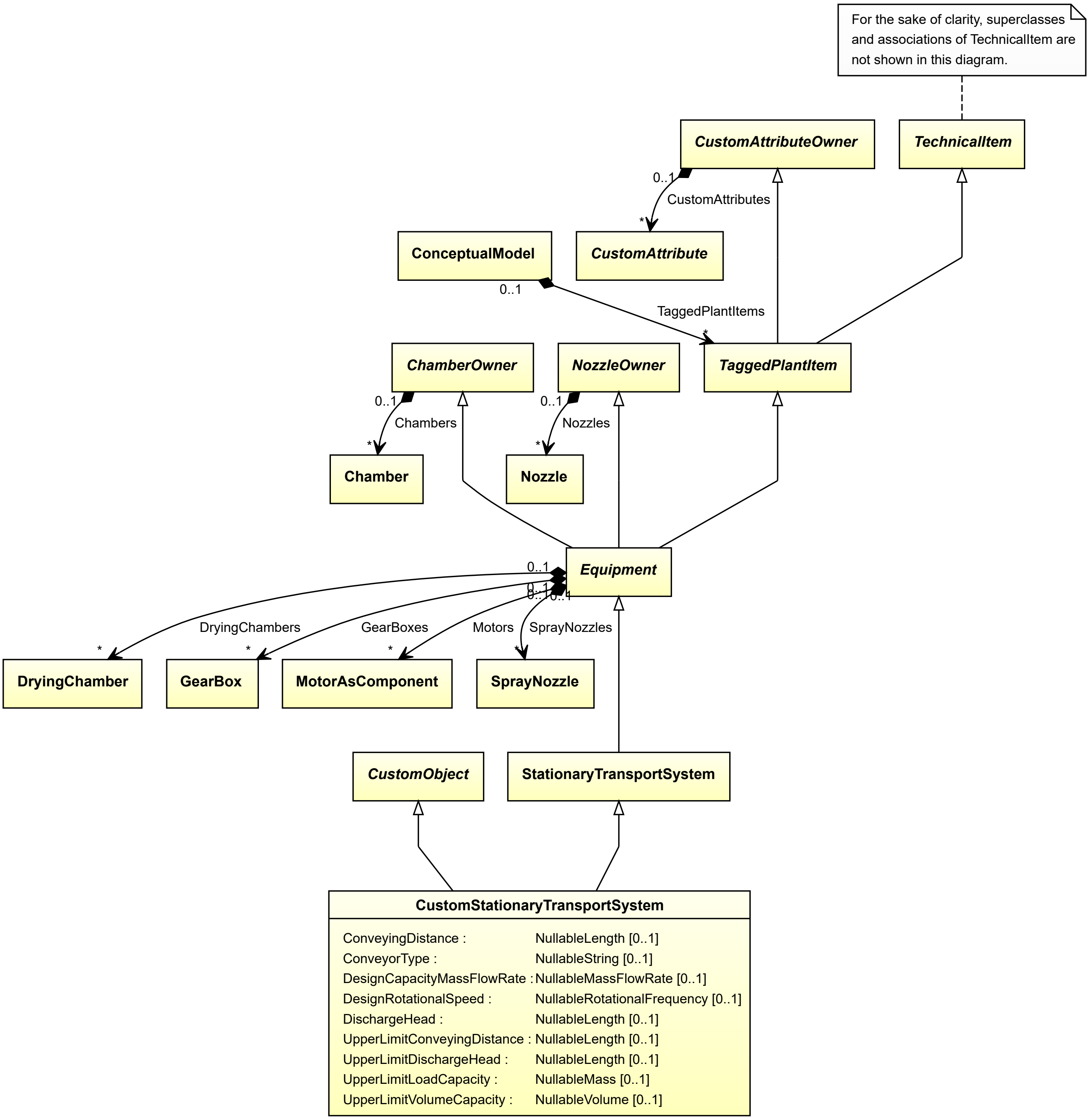
The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SHAFT POWER
Name: DesignShaftPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignShaftPower

7.58. CustomStationaryTransportSystem

7.58.1 Overview

Class

A custom *StationaryTransportSystem*, i.e., a *StationaryTransportSystem* that is not covered by any of the other subclasses of *StationaryTransportSystem* (*Conveyor*, *Lift*, or *LoadingUnloadingSystem*).


For the sake of clarity, superclasses and associations of TechnicalItem are not shown in this diagram.
CustomAttributeOwner
TechnicalItem
0..1
CustomAttributes
*
ConceptualModel
CustomAttribute
0..1
TaggedPlantItems
*
ChamberOwner
NozzleOwner
TaggedPlantItem
0..1
Chambers
*
Chamber
0..1
Nozzles
*
Nozzle
Equipment
0..1
0..1
0..1
0..1
DryingChambers
GearBoxes
Motors
SprayNozzles
*
*
*
*
DryingChamber
GearBox
MotorAsComponent
SprayNozzle
CustomObject
StationaryTransportSystem
CustomStationaryTransportSystem
ConveyingDistance : NullableLength [0..1]
ConveyorType : NullableString [0..1]
DesignCapacityMassFlowRate : NullableMassFlowRate [0..1]
DesignRotationalSpeed : NullableRotationalFrequency [0..1]
DischargeHead : NullableLength [0..1]
UpperLimitConveyingDistance : NullableLength [0..1]
UpperLimitDischargeHead : NullableLength [0..1]
UpperLimitLoadCapacity : NullableMass [0..1]
UpperLimitVolumeCapacity : NullableVolume [0..1]

Supertypes

- *CustomObject*
- *StationaryTransportSystem*

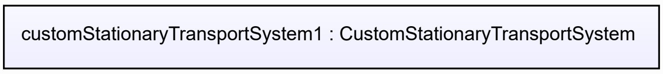
Attributes (data)

Name	Multiplicity	Type
ConveyingDistance	0..1	*NullableLength*
ConveyorType	0..1	*NullableString*
DesignCapacityMassFlowRate	0..1	*NullableMassFlowRate*
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*
DischargeHead	0..1	*NullableLength*
UpperLimitConveyingDistance	0..1	*NullableLength*
UpperLimitDischargeHead	0..1	*NullableLength*
UpperLimitLoadCapacity	0..1	*NullableMass*
UpperLimitVolumeCapacity	0..1	*NullableVolume*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: CUSTOM STATIONARY TRANSPORT SYSTEM
ComponentClass: CustomStationaryTransportSystem
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomStationaryTransportSystem

Example


customStationaryTransportSystem1 : CustomStationaryTransportSystem

Example: Implementation in Proteus Schema

```
<Equipment
    ID="customStationaryTransportSystem1"
    ComponentClass="CustomStationaryTransportSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomStationaryTransportSystem" ...>
  ...
</Equipment>
```

7.58.2 ConveyingDistance

Attribute (data)

The conveying distance of the *CustomStationaryTransportSystem*.

Multiplicity: 0..1

Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: CONVEYING DISTANCE
Name: ConveyingDistance
AttributeURI: http://sandbox.dexpi.org/rdl/ConveyingDistance

7.58.3 ConveyorType

Attribute (data)

The type of the conveyor.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: CONVEYOR TYPE ASSIGNMENT CLASS
Name: ConveyorTypeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ConveyorTypeAssignmentClass

7.58.4 DesignCapacityMassFlowRate

Attribute (data)

The capacity for the mass flow rate for which the *CustomStationaryTransportSystem* is designed.

Multiplicity: 0..1

Type: *NullableMassFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN CAPACITY MASS FLOW RATE
Name: DesignCapacityMassFlowRate
AttributeURI: http://sandbox.dexpi.org/rdl/DesignCapacityMassFlowRate

7.58.5 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *CustomStationaryTransportSystem* is designed.

Multiplicity: 0..1

Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

7.58.6 DischargeHead

Attribute (data)

The length of the *CustomStationaryTransportSystem*.

Multiplicity: 0..1

Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DISCHARGE HEAD
Name: DischargeHead
AttributeURI: http://sandbox.dexpi.org/rdl/DischargeHead

7.58.7 UpperLimitConveyingDistance

Attribute (data)

The upper limit for the conveying distance of the *CustomStationaryTransportSystem*.

Multiplicity: 0..1

Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT CONVEYING DISTANCE
Name: UpperLimitConveyingDistance
AttributeURI: http://sandbox.dexpi.org/rdl/UpperLimitConveyingDistance

7.58.8 UpperLimitDischargeHead

Attribute (data)

The upper limit for the discharge head of the *CustomStationaryTransportSystem*.

Multiplicity: 0..1

Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT DISCHARGE HEAD
Name: UpperLimitDischargeHead
AttributeURI: http://sandbox.dexpi.org/rdl/UpperLimitDischargeHead

7.58.9 UpperLimitLoadCapacity

Attribute (data)

The highest mass to transport for which the *CustomStationaryTransportSystem* is designed.

Multiplicity: 0..1

Type: *NullableMass*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT LOAD CAPACITY
Name: UpperLimitLoadCapacity
AttributeURI: http://sandbox.dexpi.org/rdl/UpperLimitLoadCapacity

7.58.10 UpperLimitVolumeCapacity

Attribute (data)

The highest volume to transport for which the *CustomStationaryTransportSystem* is designed.

Multiplicity: 0..1

Type: *NullableVolume*

Implementation in Proteus Schema

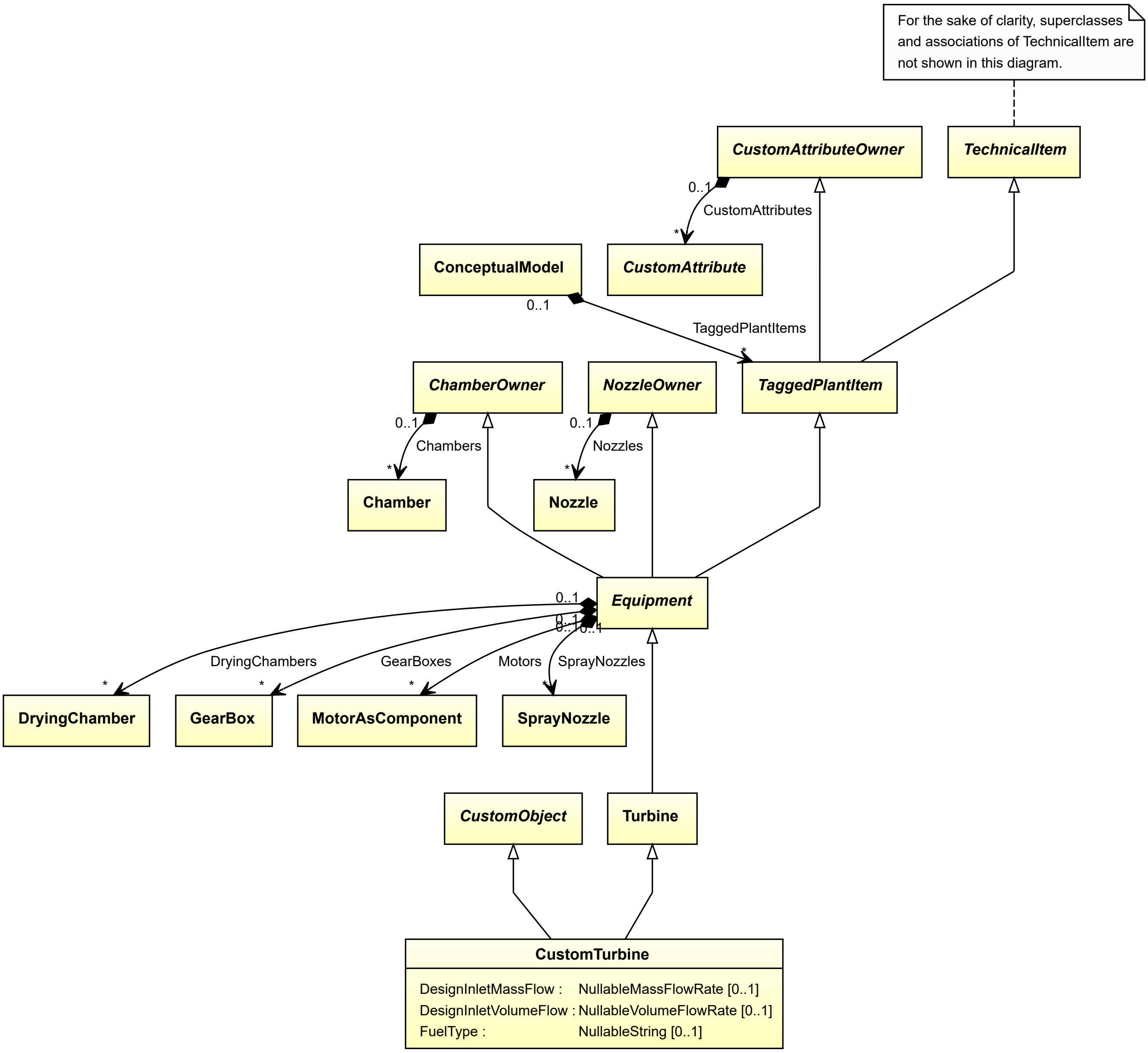
The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT VOLUME CAPACITY
Name: UpperLimitVolumeCapacity
AttributeURI: http://sandbox.dexpi.org/rdl/UpperLimitVolumeCapacity

7.59. CustomTurbine

7.59.1 Overview

Class

A custom *Turbine*, i.e., a *Turbine* that is not covered by any of the other subclasses of *Turbine* (*GasTurbine* or *SteamTurbine*).



Supertypes

- *CustomObject*
- *Turbine*

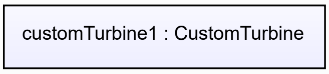
Attributes (data)

Name	Multiplicity	Type
DesignInletMassFlow	0..1	*NullableMassFlowRate*
DesignInletVolumeFlow	0..1	*NullableVolumeFlowRate*
FuelType	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: CUSTOM TURBINE
ComponentClass: CustomTurbine
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomTurbine

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="customTurbine1"
    ComponentClass="CustomTurbine"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomTurbine" ...>
  ...
</Equipment>
```

7.59.2 DesignInletMassFlow

Attribute (data)

The inlet mass flow for which the *CustomTurbine* is designed.

Multiplicity: 0..1

Type: *NullableMassFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN INLET MASS FLOW
Name: DesignInletMassFlow
AttributeURI: http://sandbox.dexpi.org/rdl/DesignInletMassFlow

7.59.3 DesignInletVolumeFlow

Attribute (data)

The inlet volume flow for which the *CustomTurbine* is designed.

Multiplicity: 0..1

Type: *NullableVolumeFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN INLET VOLUME FLOW
Name: DesignInletVolumeFlow
AttributeURI: http://sandbox.dexpi.org/rdl/DesignInletVolumeFlow

7.59.4 FuelType

Attribute (data)

The fuel type of the *CustomTurbine*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

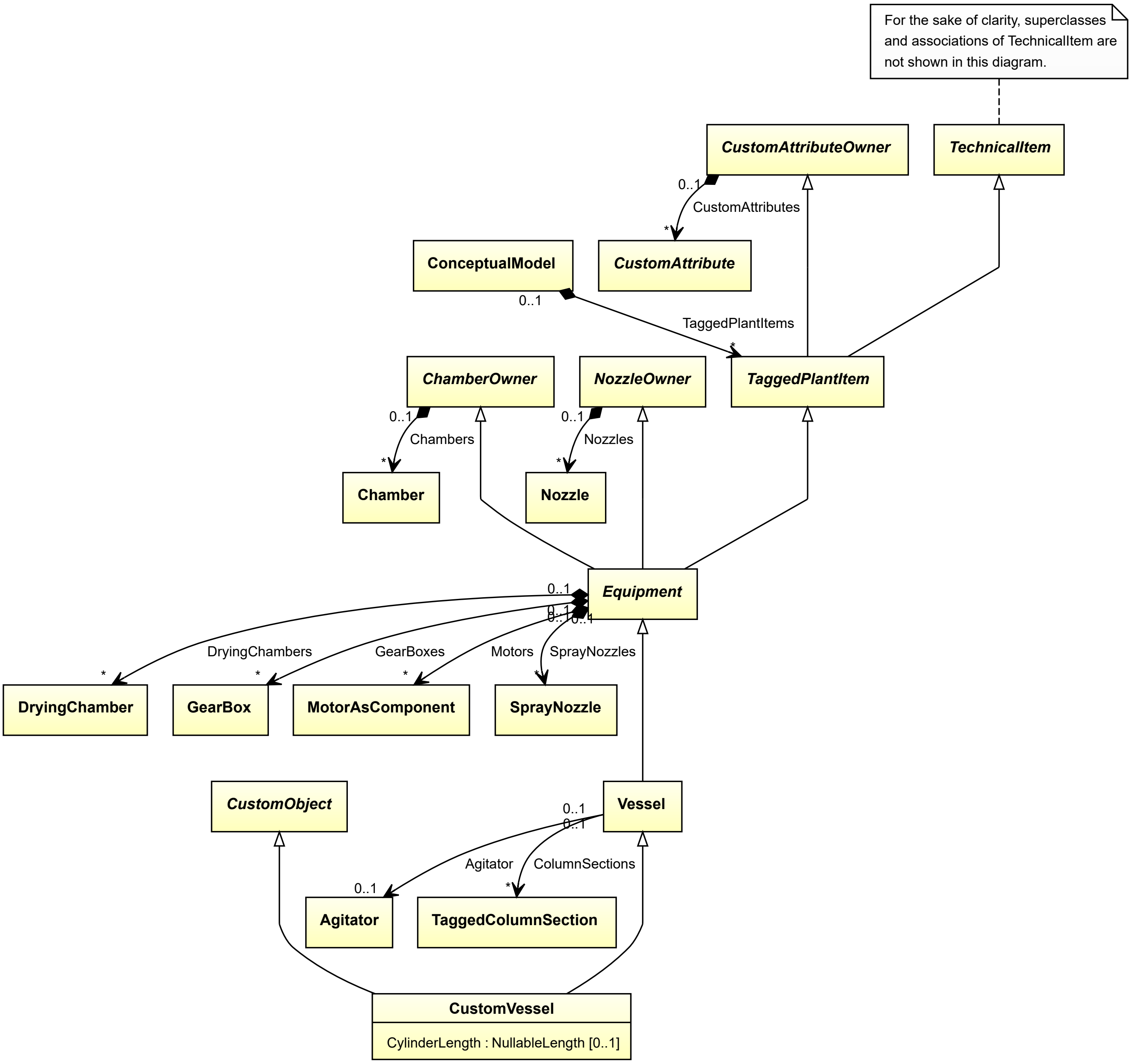
The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: FUEL TYPE
Name: FuelType
AttributeURI: http://sandbox.dexpi.org/rdl/FuelType

7.60. CustomVessel

7.60.1 Overview

Class

A custom *Vessel*, i.e., a *Vessel* that is not covered by any of the other subclasses of *Vessel* (*PressureVessel*, *Silo*, or *Tank*).



Supertypes

- *CustomObject*
- *Vessel*

Attributes (data)

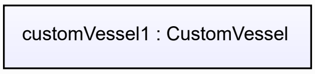
Name	Multiplicity	Type
CylinderLength	0..1	*NullableLength*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`

RDL reference: CUSTOM VESSEL
ComponentClass: CustomVessel
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomVessel

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="customVessel1"
    ComponentClass="CustomVessel"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomVessel" ...>
  ...
</Equipment>
```

7.60.2 CylinderLength

Attribute (data)

The cylinder length of the *CustomVessel*.

Multiplicity: 0..1

Type: *NullableLength*

Implementation in Proteus Schema

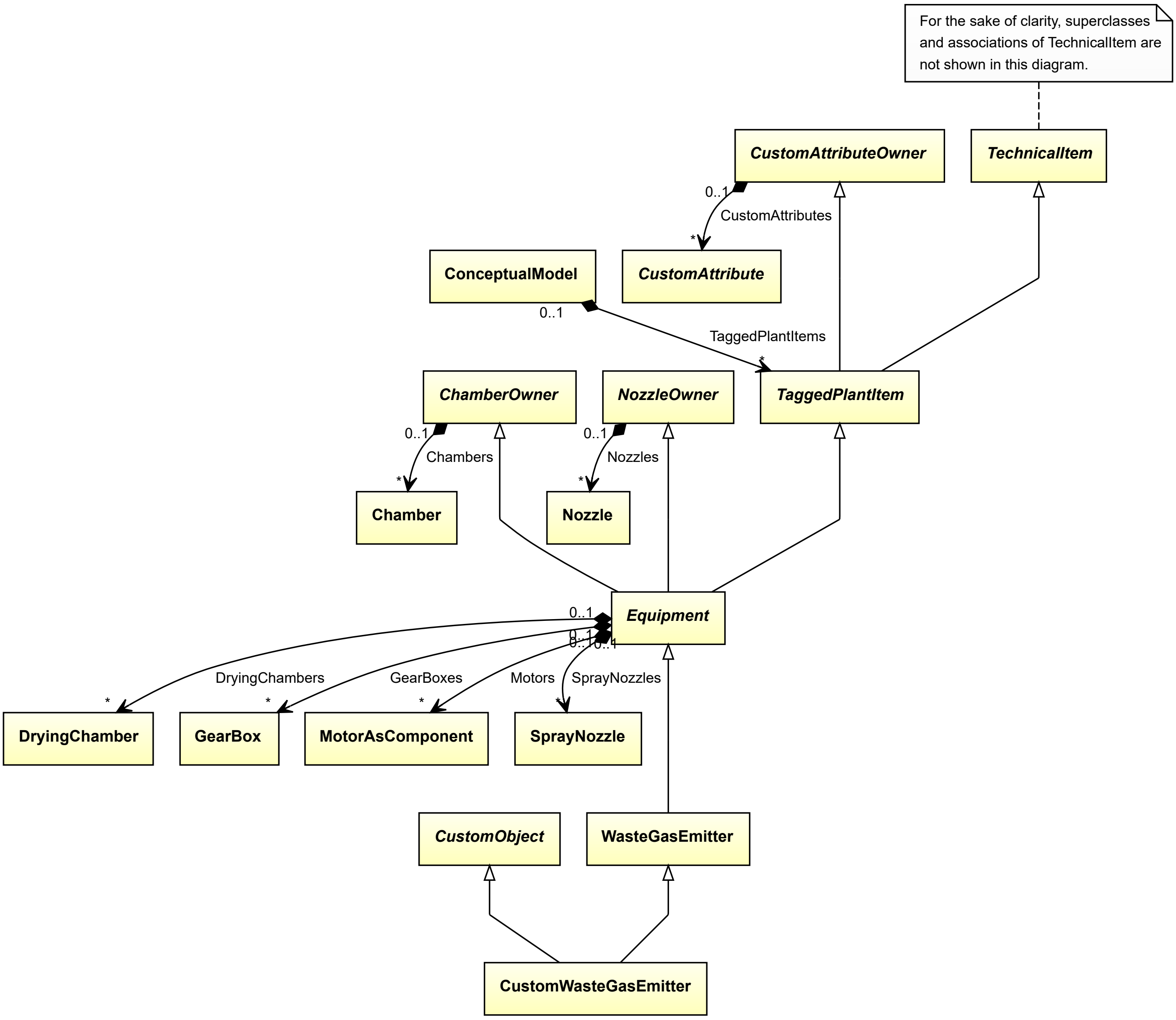
The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: CYLINDER LENGTH
Name: CylinderLength
AttributeURI: http://sandbox.dexpi.org/rdl/CylinderLength

7.61. CustomWasteGasEmitter

7.61.1 Overview

Class

A custom *WasteGasEmitter*, i.e., a *WasteGasEmitter* that is not covered by any of the other subclasses of *WasteGasEmitter* (*Chimney* or *Flare*).



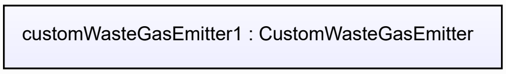
Supertypes

- *CustomObject*
- *WasteGasEmitter*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: CUSTOM WASTE GAS EMITTER
ComponentClass: CustomWasteGasEmitter
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomWasteGasEmitter

Example



Example: Implementation in Proteus Schema

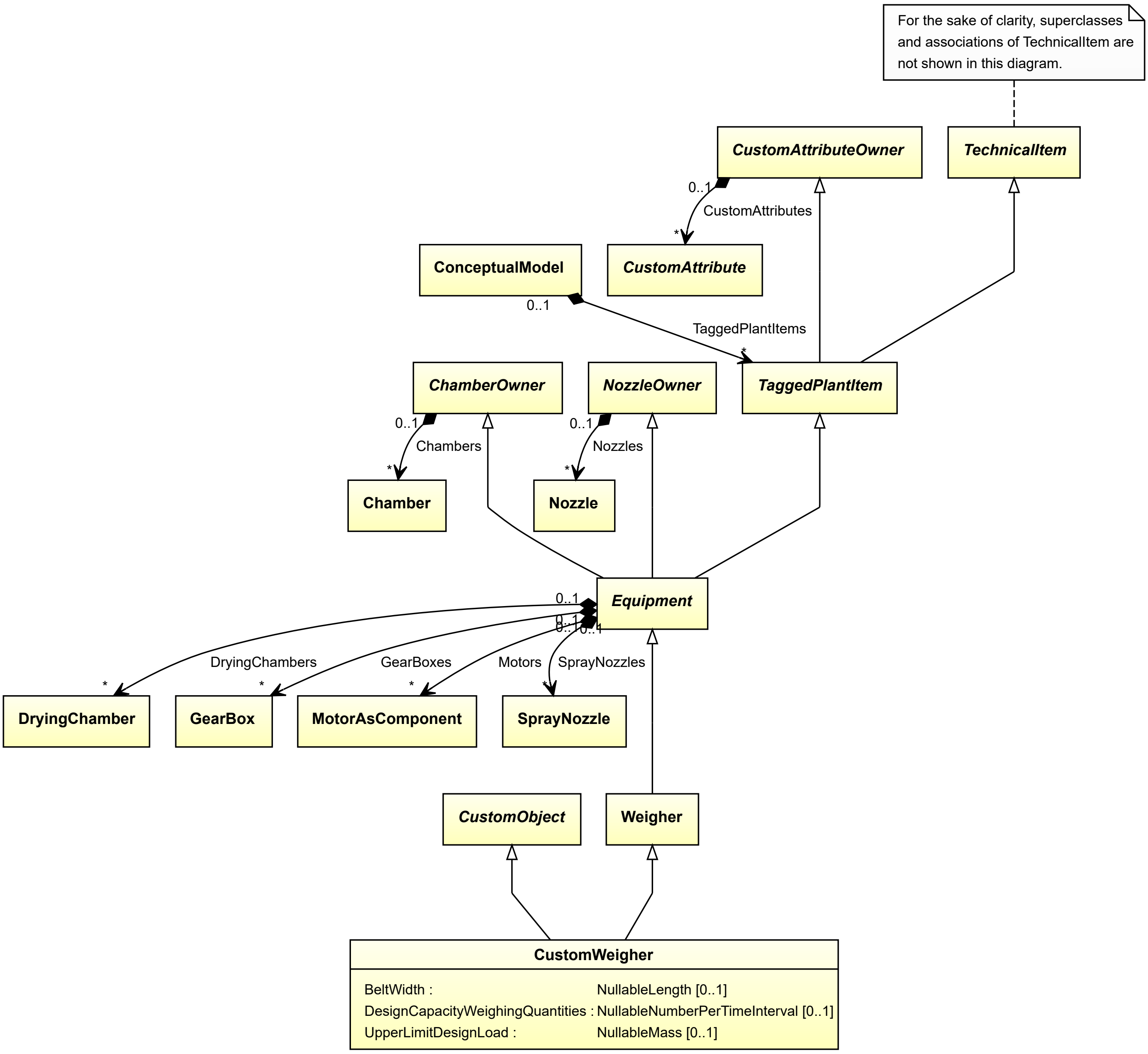
```
<Equipment
    ID="customWasteGasEmitter1"
    ComponentClass="CustomWasteGasEmitter"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomWasteGasEmitter" ...>
  ...
</Equipment>
```

7.62. CustomWeigher

7.62.1 Overview

Class

A custom *Weigher*, i.e., a *Weigher* that is not covered by any of the other subclasses of *Weigher* (*BatchWeigher* or *ContinuousWeigher*).



Supertypes

- *CustomObject*
- *Weigher*

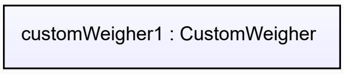
Attributes (data)

Name	Multiplicity	Type
BeltWidth	0..1	*NullableLength*
DesignCapacityWeighingQuantities	0..1	*NullableNumberPerTimeInterval*
UpperLimitDesignLoad	0..1	*NullableMass*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: CUSTOM WEIGHER
ComponentClass: CustomWeigher
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomWeigher

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="customWeigher1"
    ComponentClass="CustomWeigher"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomWeigher" ...>
  ...
</Equipment>
```

7.62.2 BeltWidth

Attribute (data)

The belt width of the *CustomWeigher*.

Multiplicity: 0..1

Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: BELT WIDTH
Name: BeltWidth
AttributeURI: http://sandbox.dexpi.org/rdl/BeltWidth

7.62.3 DesignCapacityWeighingQuantities

Attribute (data)

The capacity for the number of weighing quantities per time for which the *CustomWeigher* is designed.

Multiplicity: 0..1

Type: *NullableNumberPerTimeInterval*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN CAPACITY WEIGHING QUANTITIES
Name: DesignCapacityWeighingQuantities
AttributeURI: http://sandbox.dexpi.org/rdl/DesignCapacityWeighingQuantities

7.62.4 UpperLimitDesignLoad

Attribute (data)

The upper limit for the load for which the *CustomWeigher* is designed.

Multiplicity: 0..1

Type: *NullableMass*

Implementation in Proteus Schema

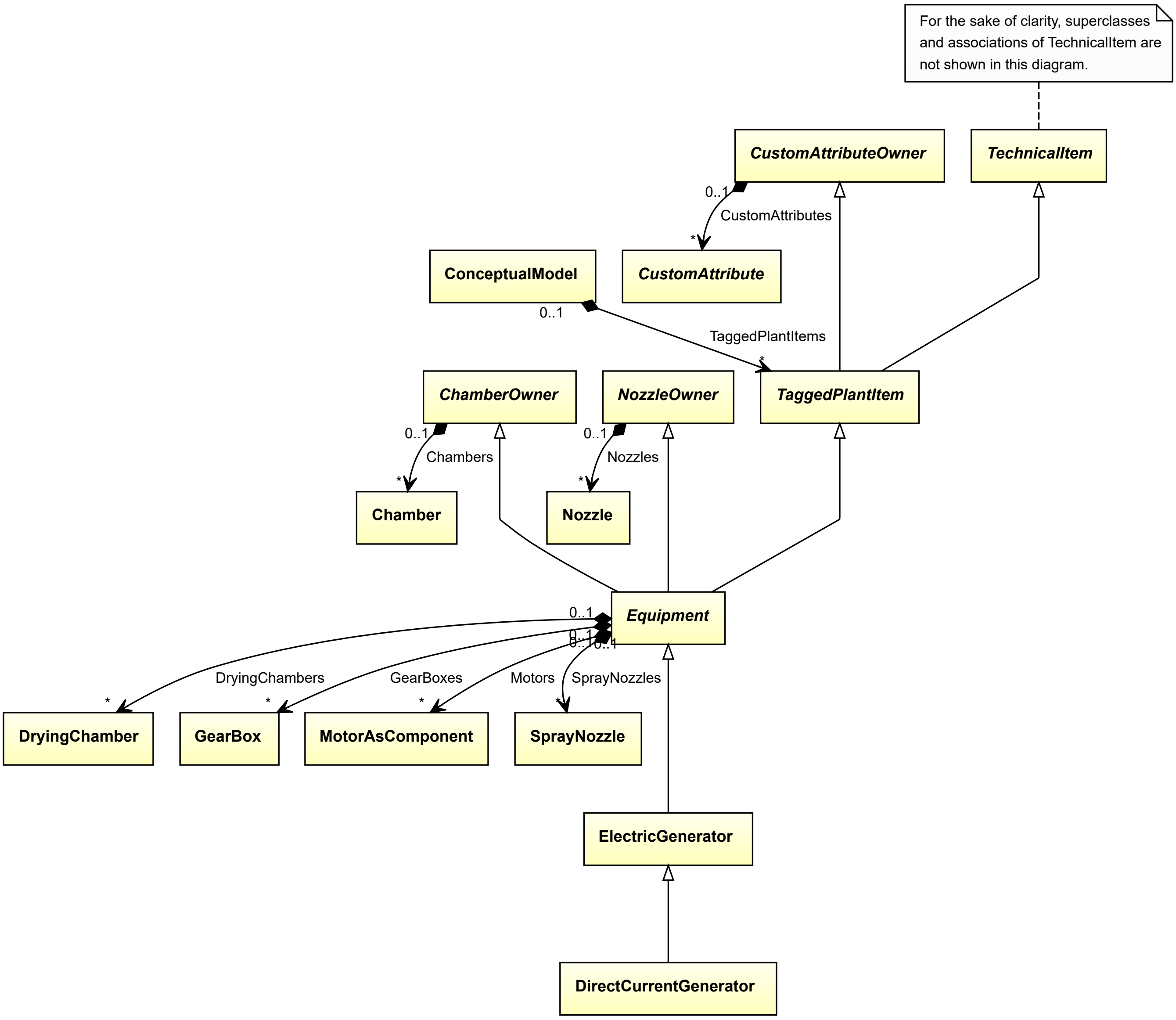
The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT DESIGN LOAD
Name: UpperLimitDesignLoad
AttributeURI: http://sandbox.dexpi.org/rdl/UpperLimitDesignLoad

7.63. DirectCurrentGenerator

7.63.1 Overview

Class

An *ElectricGenerator* and current generator for the production of direct current (DC).



Supertypes

- *ElectricGenerator*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: DIRECT CURRENT GENERATOR
ComponentClass: DirectCurrentGenerator
ComponentClassURI: http://sandbox.dexpi.org/rdl/DirectCurrentGenerator

Example

directCurrentGenerator1 : DirectCurrentGenerator

Example: Implementation in Proteus Schema

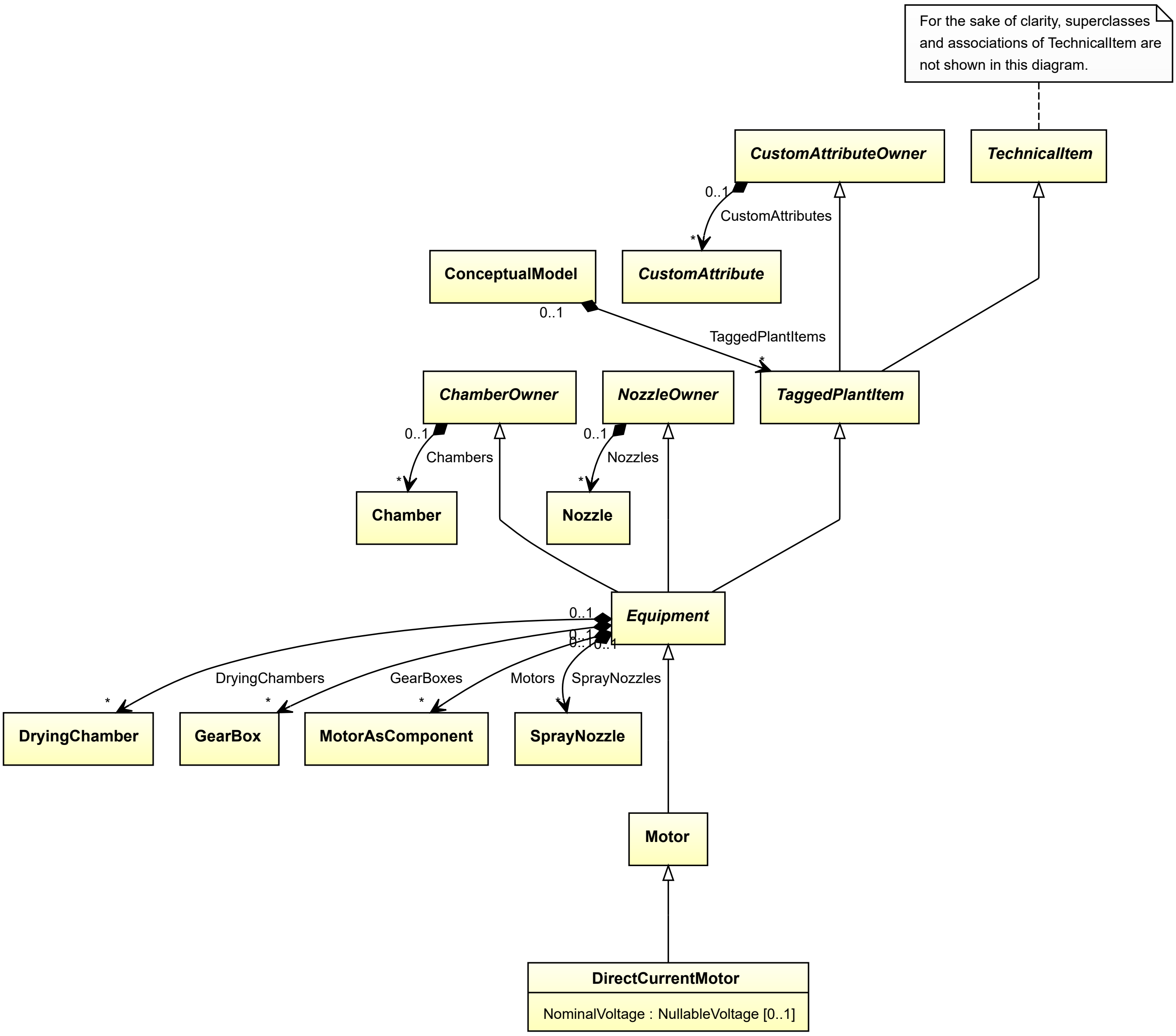
```
<Equipment
    ID="directCurrentGenerator1"
    ComponentClass="DirectCurrentGenerator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/DirectCurrentGenerator" ...>
  ...
</Equipment>
```

7.64. DirectCurrentMotor

7.64.1 Overview

Class

An electric motor for operation by direct current (from http://data.posccaesar.org/rdl/RDS472949).



Supertypes

- *Motor*

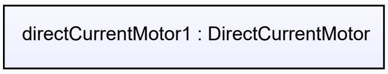
Attributes (data)

Name	Multiplicity	Type
NominalVoltage	0..1	*NullableVoltage*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: DIRECT CURRENT MOTOR
ComponentClass: DirectCurrentMotor
ComponentClassURI: http://data.posccaesar.org/rdl/RDS472949

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="directCurrentMotor1"
    ComponentClass="DirectCurrentMotor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS472949" ...>
  ...
</Equipment>
```

7.64.2 NominalVoltage

Attribute (data)

The nominal voltage of the *DirectCurrentMotor*.

Multiplicity: 0..1

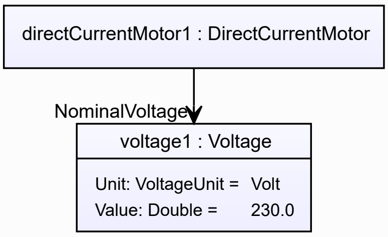
Type: *NullableVoltage*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: NOMINAL VOLTAGE
Name: NominalVoltage
AttributeURI: http://data.posccaesar.org/rdl/RDS369449

Example

The instance directCurrentMotor1 represents a *DirectCurrentMotor* with a *NominalVoltage* of 230.0 V.



Example: Implementation in Proteus Schema

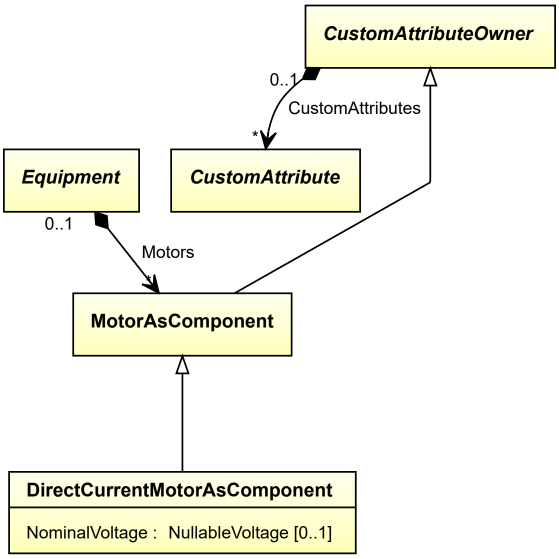
```
<Equipment
    ID="directCurrentMotor1"
    ComponentClass="DirectCurrentMotor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS472949" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NominalVoltage"
      AttributeURI="http://data.posccaesar.org/rdl/RDS369449"
      Format="double"
      Value="230.0"
      Units="Volt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1347974" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.65. DirectCurrentMotorAsComponent

7.65.1 Overview

Class

An electric motor for operation by direct current that is used as component of an apparatus or of a machine.



Supertypes

- *MotorAsComponent*

Attributes (data)

Name	Multiplicity	Type
NominalVoltage	0..1	*NullableVoltage*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: DIRECT CURRENT MOTOR AS COMPONENT
ComponentClass: DirectCurrentMotorAsComponent
ComponentClassURI: http://sandbox.dexpi.org/rdl/DirectCurrentMotorAsComponent

Example

directCurrentMotorAsComponent1 : DirectCurrentMotorAsComponent

Example: Implementation in Proteus Schema

```
<Equipment
    ID="directCurrentMotorAsComponent1"
    ComponentClass="DirectCurrentMotorAsComponent"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/DirectCurrentMotorAsComponent" ...>
  ...
</Equipment>
```

7.65.2 NominalVoltage

Attribute (data)

The nominal voltage of the *DirectCurrentMotorAsComponent*.

Multiplicity: 0..1

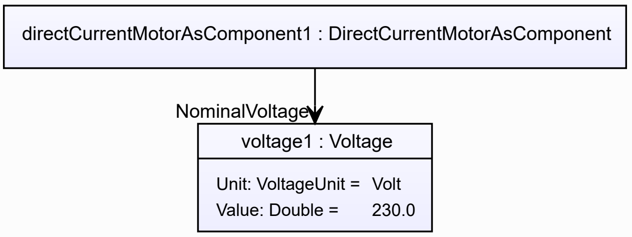
Type: *NullableVoltage*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: NOMINAL VOLTAGE
Name: NominalVoltage
AttributeURI: http://data.posccaesar.org/rdl/RDS369449

Example

The instance directCurrentMotorAsComponent1 represents a *DirectCurrentMotorAsComponent* with a *NominalVoltage* of 230.0 V.



Example: Implementation in Proteus Schema

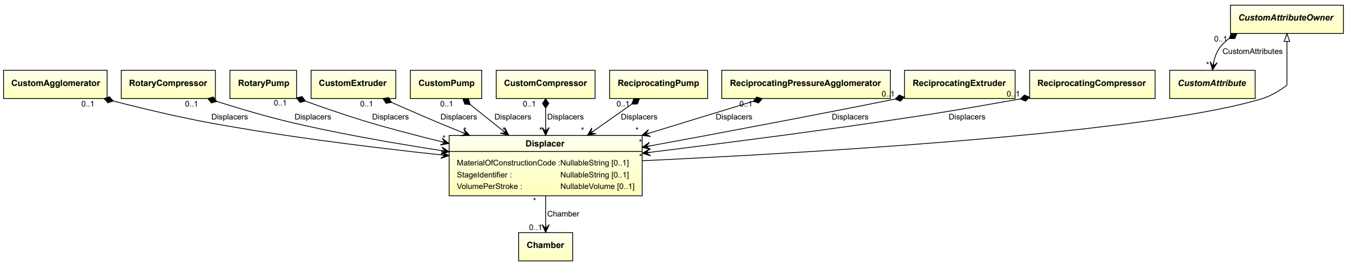
```
<Equipment
    ID="directCurrentMotorAsComponent1"
    ComponentClass="DirectCurrentMotorAsComponent"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/DirectCurrentMotorAsComponent" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NominalVoltage"
      AttributeURI="http://data.posccaesar.org/rdl/RDS369449"
      Format="double"
      Value="230.0"
      Units="Volt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1347974" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.66. Displacer

7.66.1 Overview

Class

An object that has the purpose of displacing a fluid.



Supertypes

- *CustomAttributeOwner*

Attributes (data)

Name	Multiplicity	Type
MaterialOfConstructionCode	0..1	*NullableString*
StageIdentifier	0..1	*NullableString*
VolumePerStroke	0..1	*NullableVolume*

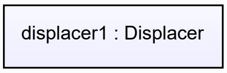
Attributes (reference)

Name	Multiplicity	Type
Chamber	0..1	*Chamber*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: DISPLACER
ComponentClass: Displacer
ComponentClassURI: http://sandbox.dexpi.org/rdl/Displacer

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="displacer1"
    ComponentClass="Displacer"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Displacer" ...>
  ...
</Equipment>
```

7.66.2 Chamber

Attribute (reference)

The *Chamber* in which the *Displacer* is located, if applicable. The Chamber must be a component of the same object as the Displacer.

Multiplicity: 0..1

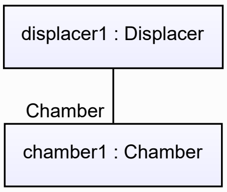
Type: *Chamber*

Opposite multiplicity: 0..*

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: `"is located in"`
Opposite association type: `"is the location of"`

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="displacer1"
    ComponentClass="Displacer"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Displacer" ...>
  ...
  <Association
    Type="is located in"
    ItemID="chamber1" />
  ...
<Equipment />
...
<Equipment
    ID="chamber1"
    ComponentClass="Chamber"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS903151421" ...>
  ...
  <Association
    Type="is the location of"
    ItemID="displacer1" />
  ...
<Equipment />
```

7.66.3 MaterialOfConstructionCode

Attribute (data)

A code that gives the material of construction of the *Displacer*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: MATERIAL OF CONSTRUCTION CODE ASSIGNMENT CLASS
Name: MaterialOfConstructionCodeAssignmentClass
AttributeURI: http://data.posccaesar.org/rdl/RDS1460719741

Example

“1.4306” (*String*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="displacer1"
    ComponentClass="Displacer"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Displacer" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="MaterialOfConstructionCodeAssignmentClass"
      AttributeURI="http://data.posccaesar.org/rdl/RDS1460719741"
      Format="string"
      Value="1.4306" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.66.4 StageIdentifier

Attribute (data)

The stage identifier of the *Displacer*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: STAGE IDENTIFIER ASSIGNMENT CLASS
Name: StageIdentifierAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/StageIdentifierAssignmentClass

Example

“s1” (*String*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="displacer1"
    ComponentClass="Displacer"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Displacer" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="StageIdentifierAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/StageIdentifierAssignmentClass"
      Format="string"
      Value="s1" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.66.5 VolumePerStroke

Attribute (data)

The volume per stroke of the *Displacer*.

Multiplicity: 0..1

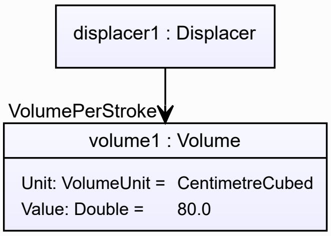
Type: *NullableVolume*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: VOLUME PER STROKE
Name: VolumePerStroke
AttributeURI: http://data.posccaesar.org/rdl/RDS7503244

Example

The instance displacer1 represents a *Displacer* with a *VolumePerStroke* of 80.0 cm^3.



Example: Implementation in Proteus Schema

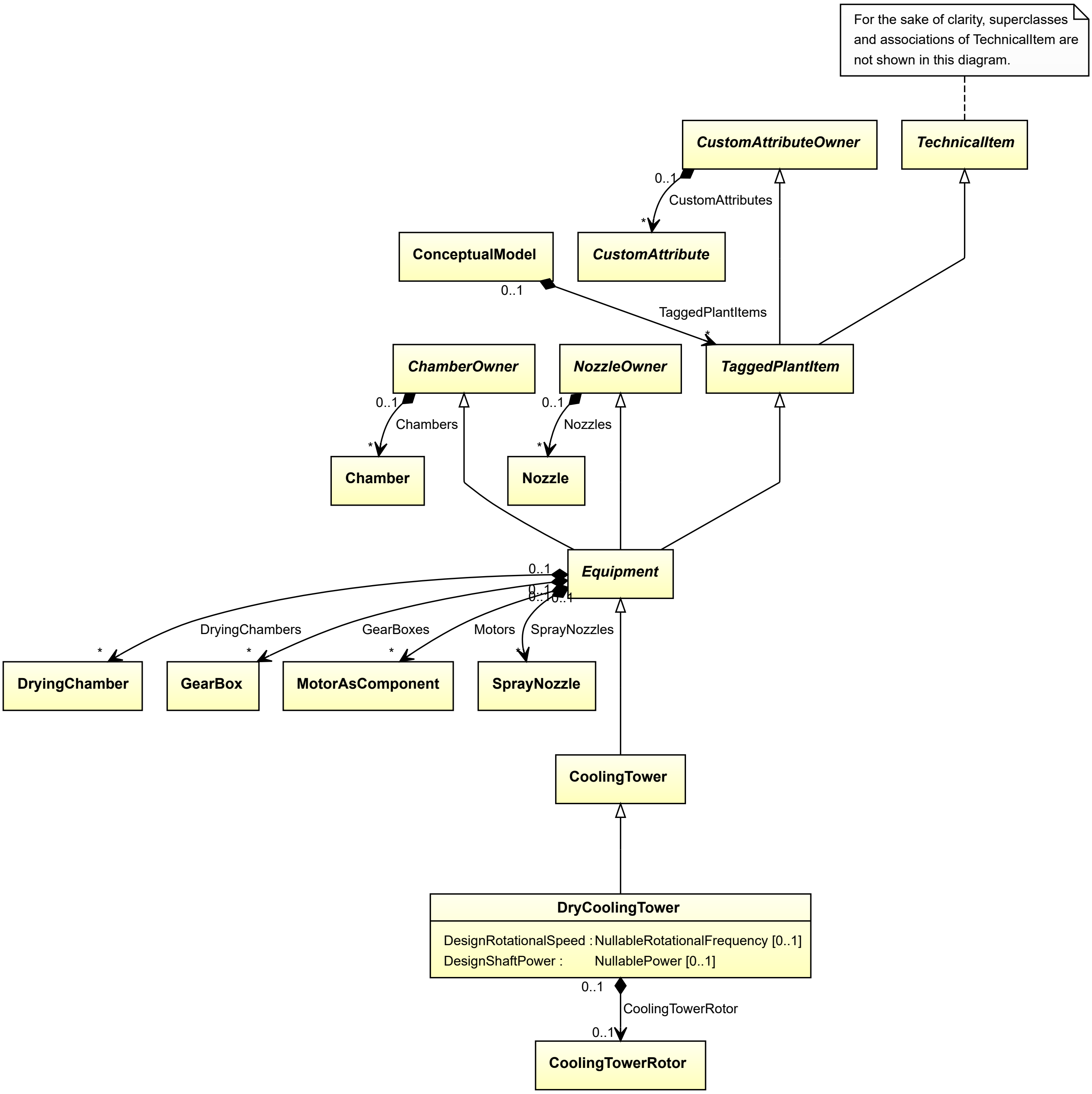
```
<Equipment
    ID="displacer1"
    ComponentClass="Displacer"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Displacer" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="VolumePerStroke"
      AttributeURI="http://data.posccaesar.org/rdl/RDS7503244"
      Format="double"
      Value="80.0"
      Units="CentimetreCubed"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1357874" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.67. DryCoolingTower

7.67.1 Overview

Class

A *CoolingTower* that is an indirect contact heat exchanger where, by full utilization of dry surface coil sections, no direct contact (and no evaporation) occurs between air and water; hence the water is cooled totally by sensible heat transfer (from http://data.15926.org/rdl/RDS14072386).



Supertypes

- *CoolingTower*

Attributes (data)

Name	Multiplicity	Type
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*
DesignShaftPower	0..1	*NullablePower*

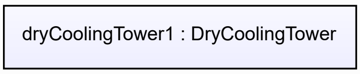
Attributes (composition)

Name	Multiplicity	Type
CoolingTowerRotor	0..1	*CoolingTowerRotor*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: DRY COOLING TOWER
ComponentClass: DryCoolingTower
ComponentClassURI: http://data.posccaesar.org/rdl/RDS14072386

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="dryCoolingTower1"
    ComponentClass="DryCoolingTower"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS14072386" ...>
  ...
</Equipment>
```

7.67.2 CoolingTowerRotor

Attribute (composition)

The cooling tower rotor of the *DryCoolingTower*.

Multiplicity: 0..1

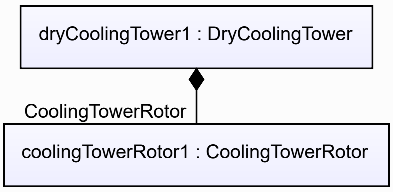
Type: *CoolingTowerRotor*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *CoolingTowerRotor*) is a child of the `<Equipment>` element for the attribute owner (a *DryCoolingTower*).

Example


dryCoolingTower1 : DryCoolingTower
CoolingTowerRotor
coolingTowerRotor1 : CoolingTowerRotor

Example: Implementation in Proteus Schema

```
<Equipment
    ID="dryCoolingTower1"
    ComponentClass="DryCoolingTower"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS14072386" ...>
  ...
  <Equipment
      ID="coolingTowerRotor1"
      ComponentClass="CoolingTowerRotor"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/CoolingTowerRotor" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.67.3 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *DryCoolingTower* is designed.

Multiplicity: 0..1

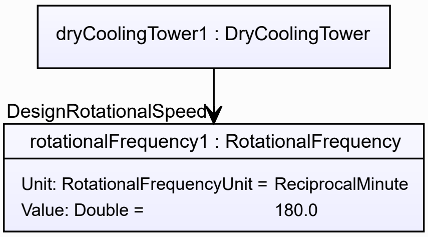
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

Example

The instance dryCoolingTower1 represents a *DryCoolingTower* with a *DesignRotationalSpeed* of 180.0 min^{-1}.


dryCoolingTower1 : DryCoolingTower
DesignRotationalSpeed
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 180.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="dryCoolingTower1"
    ComponentClass="DryCoolingTower"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS14072386" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignRotationalSpeed"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignRotationalSpeed"
      Format="double"
      Value="180.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.67.4 DesignShaftPower

Attribute (data)

The shaft power for which the *DryCoolingTower* is designed.

Multiplicity: 0..1

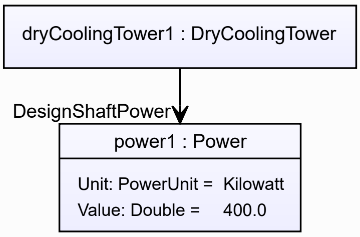
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SHAFT POWER
Name: DesignShaftPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignShaftPower

Example

The instance dryCoolingTower1 represents a *DryCoolingTower* with a *DesignShaftPower* of 400.0 kW.


dryCoolingTower1 : DryCoolingTower
DesignShaftPower
power1 : Power
Unit: PowerUnit = Kilowatt
Value: Double = 400.0

Example: Implementation in Proteus Schema

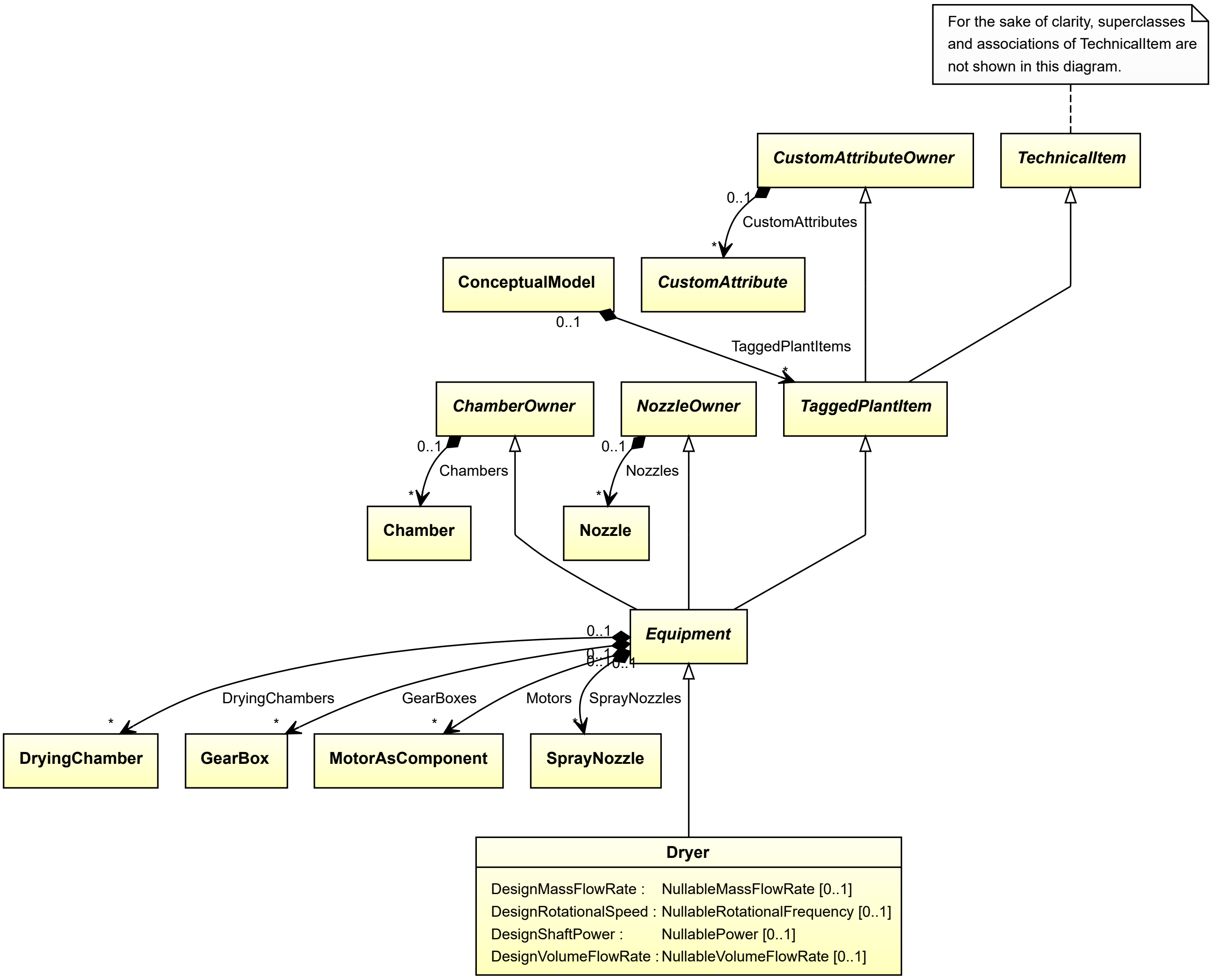
```
<Equipment
    ID="dryCoolingTower1"
    ComponentClass="DryCoolingTower"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS14072386" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignShaftPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignShaftPower"
      Format="double"
      Value="400.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.68. Dryer

7.68.1 Overview

Class

An object that has the capability of drying (from http://data.15926.org/rdl/RDS1066939451).



Supertypes

- *Equipment*

Subtypes

- *ConvectionDryer*
- *CustomDryer*
- *HeatedSurfaceDryer*

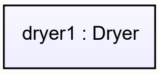
Attributes (data)

Name	Multiplicity	Type
DesignMassFlowRate	0..1	*NullableMassFlowRate*
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*
DesignShaftPower	0..1	*NullablePower*
DesignVolumeFlowRate	0..1	*NullableVolumeFlowRate*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: DRIER
ComponentClass: Drier
ComponentClassURI: http://data.posccaesar.org/rdl/RDS1066939451

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="dryer1"
    ComponentClass="Drier"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS1066939451" ...>
  ...
</Equipment>
```

7.68.2 DesignMassFlowRate

Attribute (data)

The mass flow rate for which the *Dryer* is designed.

Multiplicity: 0..1

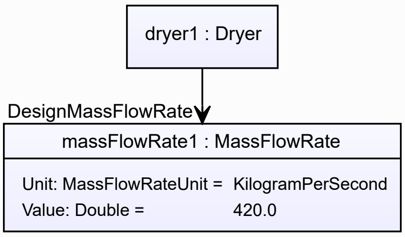
Type: *NullableMassFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN MASS FLOW RATE
Name: DesignMassFlowRate
AttributeURI: http://data.posccaesar.org/rdl/RDS14286182

Example

The instance dryer1 represents a *Dryer* with a *DesignMassFlowRate* of 420.0 kg/s.


dryer1 : Dryer
DesignMassFlowRate
massFlowRate1 : MassFlowRate
Unit: MassFlowRateUnit = KilogramPerSecond
Value: Double = 420.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="dryer1"
    ComponentClass="Drier"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS1066939451" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignMassFlowRate"
      AttributeURI="http://data.posccaesar.org/rdl/RDS14286182"
      Format="double"
      Value="420.0"
      Units="KilogramPerSecond"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1329659" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.68.3 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *Dryer* is designed.

Multiplicity: 0..1

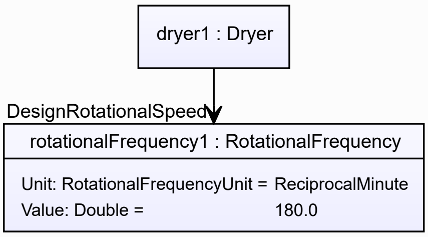
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

Example

The instance dryer1 represents a *Dryer* with a *DesignRotationalSpeed* of 180.0 min^{-1}.


dryer1 : Dryer
DesignRotationalSpeed
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 180.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="dryer1"
    ComponentClass="Drier"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS1066939451" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignRotationalSpeed"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignRotationalSpeed"
      Format="double"
      Value="180.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.68.4 DesignShaftPower

Attribute (data)

The shaft power for which the *Dryer* is designed.

Multiplicity: 0..1

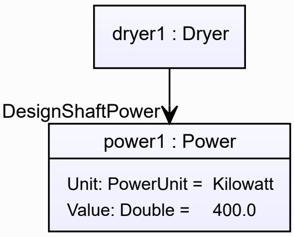
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SHAFT POWER
Name: DesignShaftPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignShaftPower

Example

The instance dryer1 represents a *Dryer* with a *DesignShaftPower* of 400.0 kW.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="dryer1"
    ComponentClass="Drier"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS1066939451" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignShaftPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignShaftPower"
      Format="double"
      Value="400.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.68.5 DesignVolumeFlowRate

Attribute (data)

The volume flow rate for which the *Dryer* is designed.

Multiplicity: 0..1

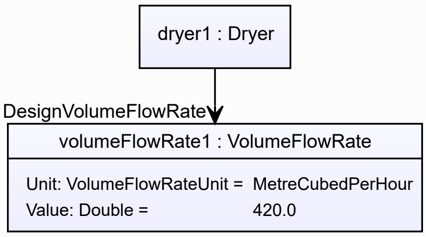
Type: *NullableVolumeFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN VOLUME FLOW RATE
Name: DesignVolumeFlowRate
AttributeURI: http://data.posccaesar.org/rdl/RDS14286227

Example

The instance dryer1 represents a *Dryer* with a *DesignVolumeFlowRate* of 420.0 m^3/h.


dryer1 : Dryer
DesignVolumeFlowRate
volumeFlowRate1 : VolumeFlowRate
Unit: VolumeFlowRateUnit = MetreCubedPerHour
Value: Double = 420.0

Example: Implementation in Proteus Schema

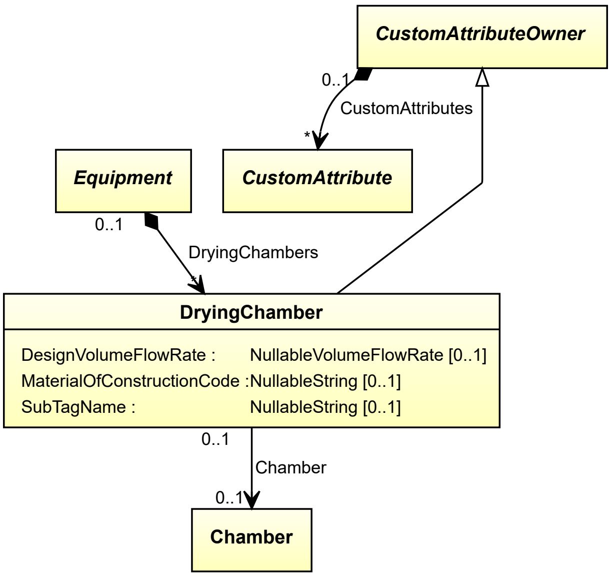
```
<Equipment
    ID="dryer1"
    ComponentClass="Drier"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS1066939451" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignVolumeFlowRate"
      AttributeURI="http://data.posccaesar.org/rdl/RDS14286227"
      Format="double"
      Value="420.0"
      Units="MetreCubedPerHour"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1321064" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.69. DryingChamber

7.69.1 Overview

Class

A device that is a chamber, fixed or portable, for drying used as a component of an apparatus or a machine.


CustomAttributeOwner
0..1
CustomAttributes
*
Equipment
CustomAttribute
0..1
DryingChambers
DryingChamber
DesignVolumeFlowRate : NullableVolumeFlowRate [0..1]
MaterialOfConstructionCode :NullableString [0..1]
SubTagName : NullableString [0..1]
0..1
Chamber
0..1
Chamber

Supertypes

- *CustomAttributeOwner*

Attributes (data)

Name	Multiplicity	Type
DesignVolumeFlowRate	0..1	*NullableVolumeFlowRate*
MaterialOfConstructionCode	0..1	*NullableString*
SubTagName	0..1	*NullableString*

Attributes (reference)

Name	Multiplicity	Type
Chamber	0..1	*Chamber*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: DRYING CHAMBER
ComponentClass: DryingChamber
ComponentClassURI: http://sandbox.dexpi.org/rdl/DryingChamber

Example

dryingChamber1 : DryingChamber

Example: Implementation in Proteus Schema

```
<Equipment
    ID="dryingChamber1"
    ComponentClass="DryingChamber"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/DryingChamber" ...>
  ...
</Equipment>
```

7.69.2 Chamber

Attribute (reference)

The *Chamber* in which the *DryingChamber* is located, if applicable. The Chamber must be a component of the same object as the *DryingChamber*.

Multiplicity: 0..1

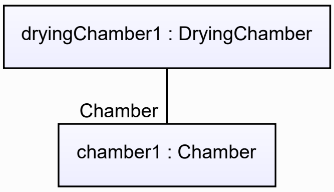
Type: *Chamber*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: `"is located in"`
Opposite association type: `"is the location of"`

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="dryingChamber1"
    ComponentClass="DryingChamber"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/DryingChamber" ...>
  ...
  <Association
    Type="is located in"
    ItemID="chamber1" />
  ...
<Equipment />
...
<Equipment
    ID="chamber1"
    ComponentClass="Chamber"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS903151421" ...>
  ...
  <Association
    Type="is the location of"
    ItemID="dryingChamber1" />
  ...
<Equipment />
```

7.69.3 DesignVolumeFlowRate

Attribute (data)

The volume flow rate for which the *DryingChamber* is designed.

Multiplicity: 0..1

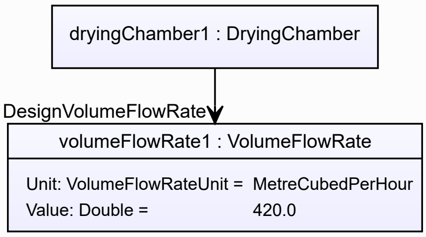
Type: *NullableVolumeFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN VOLUME FLOW RATE
Name: DesignVolumeFlowRate
AttributeURI: http://data.posccaesar.org/rdl/RDS14286227

Example

The instance dryingChamber1 represents a *DryingChamber* with a *DesignVolumeFlowRate* of 420.0 m^3/h.


dryingChamber1 : DryingChamber
DesignVolumeFlowRate
volumeFlowRate1 : VolumeFlowRate
Unit: VolumeFlowRateUnit = MetreCubedPerHour
Value: Double = 420.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="dryingChamber1"
    ComponentClass="DryingChamber"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/DryingChamber" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignVolumeFlowRate"
      AttributeURI="http://data.posccaesar.org/rdl/RDS14286227"
      Format="double"
      Value="420.0"
      Units="MetreCubedPerHour"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1321064" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.69.4 MaterialOfConstructionCode

Attribute (data)

A code that gives the material of construction of the *DryingChamber*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: MATERIAL OF CONSTRUCTION CODE ASSIGNMENT CLASS
Name: MaterialOfConstructionCodeAssignmentClass
AttributeURI: http://data.posccaesar.org/rdl/RDS1460719741

Example

“1.4306” (*String*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="dryingChamber1"
    ComponentClass="DryingChamber"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/DryingChamber" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="MaterialOfConstructionCodeAssignmentClass"
      AttributeURI="http://data.posccaesar.org/rdl/RDS1460719741"
      Format="string"
      Value="1.4306" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.69.5 SubTagName

Attribute (data)

The sub tag name of the *DryingChamber*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: SUB TAG NAME ASSIGNMENT CLASS
Name: SubTagNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/SubTagNameAssignmentClass

Example

“ST1” (*String*)

Example: Implementation in Proteus Schema

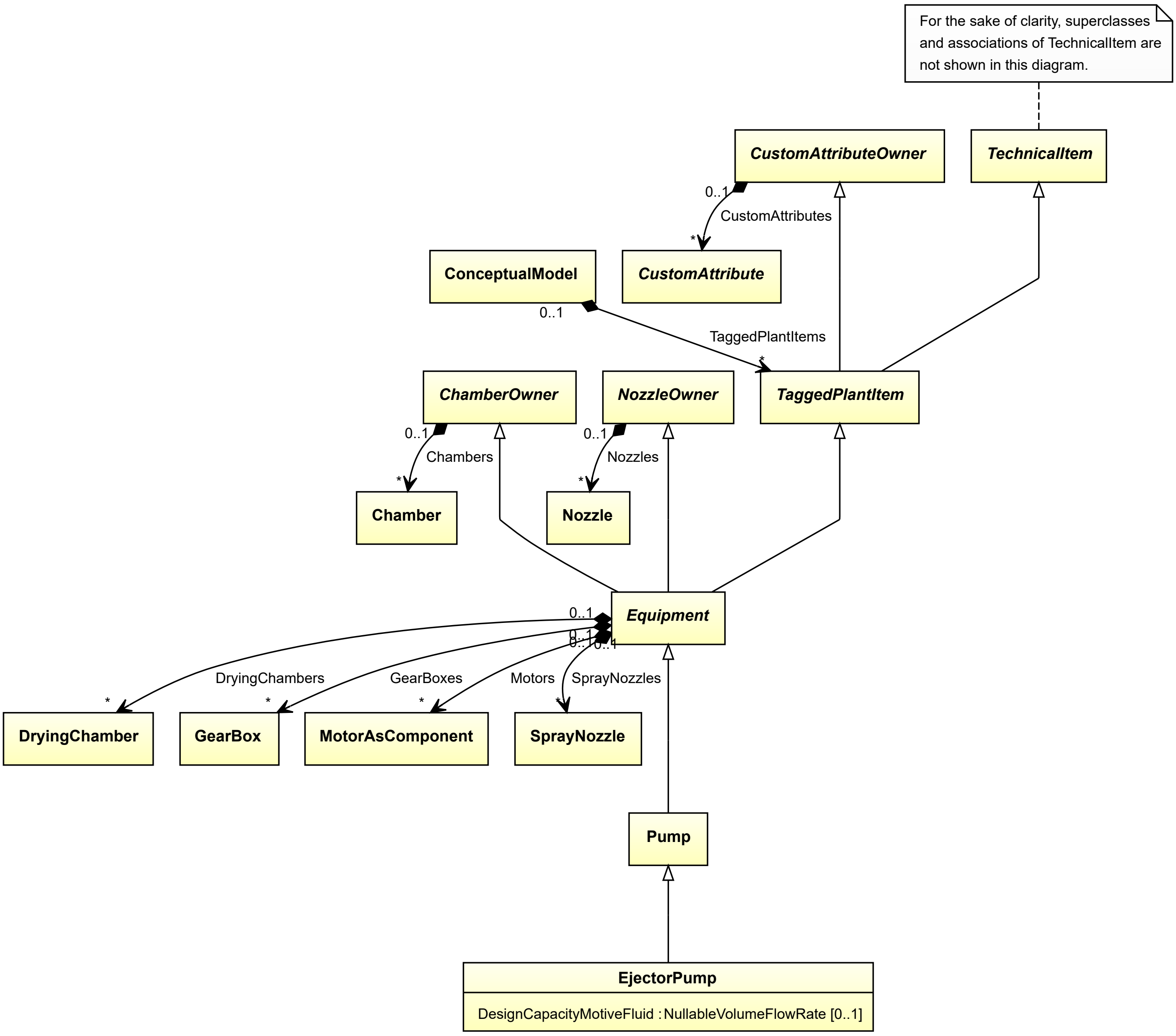
```
<Equipment
    ID="dryingChamber1"
    ComponentClass="DryingChamber"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/DryingChamber" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="SubTagNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/SubTagNameAssignmentClass"
      Format="string"
      Value="ST1" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.70. EjectorPump

7.70.1 Overview

Class

A pump which uses pressurized gas or liquid passing through an ejector to transport liquid (from http://data.posccaesar.org/rdl/RDS860624).



Supertypes

- *Pump*

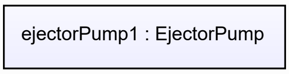
Attributes (data)

Name	Multiplicity	Type
DesignCapacityMotiveFluid	0..1	*NullableVolumeFlowRate*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: EJECTOR PUMP
ComponentClass: EjectorPump
ComponentClassURI: http://data.posccaesar.org/rdl/RDS860624

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="ejectorPump1"
    ComponentClass="EjectorPump"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS860624" ...>
  ...
</Equipment>
```

7.70.2 DesignCapacityMotiveFluid

Attribute (data)

The capacity of the volume flow rate for the motive fluid for which the *EjectorPump* is designed.

Multiplicity: 0..1

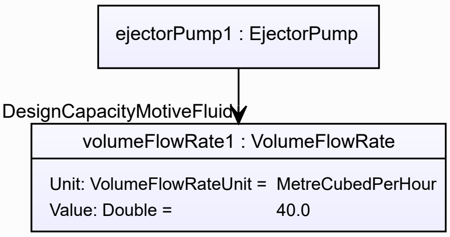
Type: *NullableVolumeFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN CAPACITY MOTIVE FLUID
Name: DesignCapacityMotiveFluid
AttributeURI: http://sandbox.dexpi.org/rdl/DesignCapacityMotiveFluid

Example

The instance ejectorPump1 represents an *EjectorPump* with a *DesignCapacityMotiveFluid* of 40.0 m^3/h.


ejectorPump1 : EjectorPump
DesignCapacityMotiveFluid
volumeFlowRate1 : VolumeFlowRate
Unit: VolumeFlowRateUnit = MetreCubedPerHour
Value: Double = 40.0

Example: Implementation in Proteus Schema

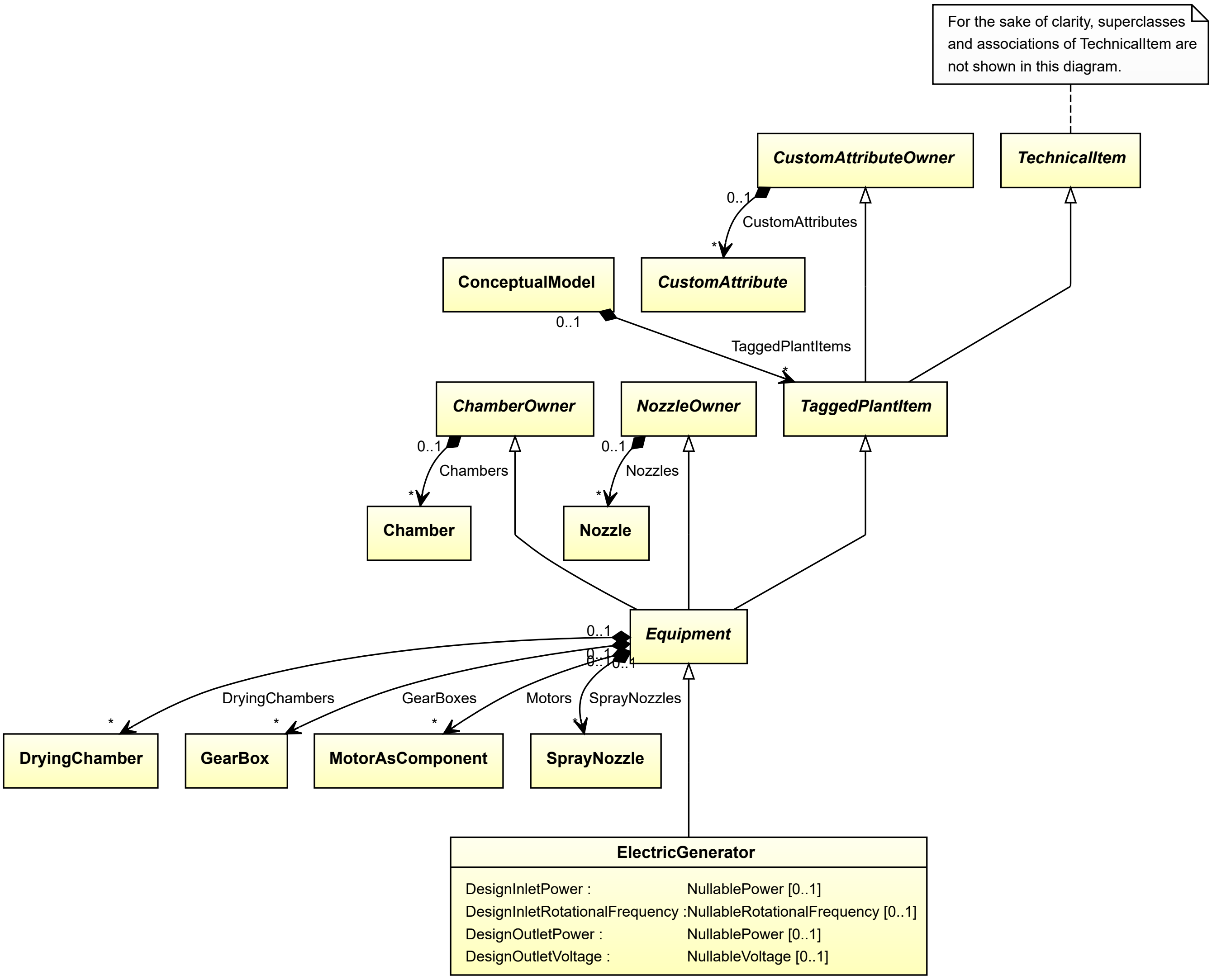
```
<Equipment
    ID="ejectorPump1"
    ComponentClass="EjectorPump"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS860624" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignCapacityMotiveFluid"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignCapacityMotiveFluid"
      Format="double"
      Value="40.0"
      Units="MetreCubedPerHour"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1321064" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.71. ElectricGenerator

7.71.1 Overview

Class

An electric rotating machine that transforms non-electric energy into electric energy (from http://data.posccaesar.org/rdl/RDS415709).



Supertypes

- *Equipment*

Subtypes

- *AlternatingCurrentGenerator*
- *CustomElectricGenerator*
- *DirectCurrentGenerator*

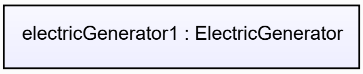
Attributes (data)

Name	Multiplicity	Type
DesignInletPower	0..1	*NullablePower*
DesignInletRotationalFrequency	0..1	*NullableRotationalFrequency*
DesignOutletPower	0..1	*NullablePower*
DesignOutletVoltage	0..1	*NullableVoltage*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: ELECTRIC GENERATOR
ComponentClass: ElectricGenerator
ComponentClassURI: http://data.posccaesar.org/rdl/RDS415709

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="electricGenerator1"
    ComponentClass="ElectricGenerator"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS415709" ...>
  ...
</Equipment>
```

7.71.2 DesignInletPower

Attribute (data)

The inlet power for which the *ElectricGenerator* is designed.

Multiplicity: 0..1

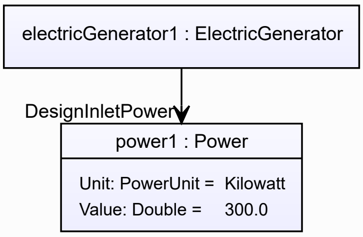
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN INLET POWER
Name: DesignInletPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignInletPower

Example

The instance electricGenerator1 represents an *ElectricGenerator* with a *DesignInletPower* of 300.0 kW.


electricGenerator1 : ElectricGenerator
DesignInletPower
power1 : Power
Unit: PowerUnit = Kilowatt
Value: Double = 300.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="electricGenerator1"
    ComponentClass="ElectricGenerator"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS415709" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignInletPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignInletPower"
      Format="double"
      Value="300.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.71.3 DesignInletRotationalFrequency

Attribute (data)

The inlet rotational frequency for which the *ElectricGenerator* is designed.

Multiplicity: 0..1

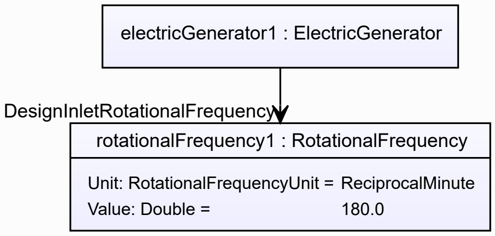
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN INLET ROTATIONAL FREQUENCY
Name: DesignInletRotationalFrequency
AttributeURI: http://sandbox.dexpi.org/rdl/DesignInletRotationalFrequency

Example

The instance electricGenerator1 represents an *ElectricGenerator* with a *DesignInletRotationalFrequency* of 180.0 min^{-1}.


electricGenerator1 : ElectricGenerator
DesignInletRotationalFrequency
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 180.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="electricGenerator1"
    ComponentClass="ElectricGenerator"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS415709" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignInletRotationalFrequency"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignInletRotationalFrequency"
      Format="double"
      Value="180.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.71.4 DesignOutletPower

Attribute (data)

The outlet power for which the *ElectricGenerator* is designed.

Multiplicity: 0..1

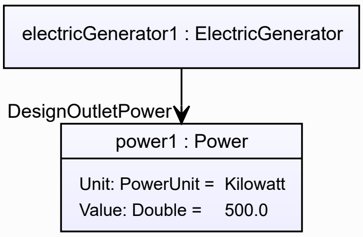
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN OUTLET POWER
Name: DesignOutletPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignOutletPower

Example

The instance electricGenerator1 represents an *ElectricGenerator* with a *DesignOutletPower* of 500.0 kW.


electricGenerator1 : ElectricGenerator
DesignOutletPower
power1 : Power
Unit: PowerUnit = Kilowatt
Value: Double = 500.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="electricGenerator1"
    ComponentClass="ElectricGenerator"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS415709" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignOutletPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignOutletPower"
      Format="double"
      Value="500.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.71.5 DesignOutletVoltage

Attribute (data)

The outlet voltage for which the *ElectricGenerator* is designed.

Multiplicity: 0..1

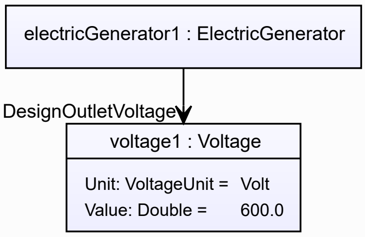
Type: *NullableVoltage*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN OUTLET VOLTAGE
Name: DesignOutletVoltage
AttributeURI: http://sandbox.dexpi.org/rdl/DesignOutletVoltage

Example

The instance electricGenerator1 represents an *ElectricGenerator* with a *DesignOutletVoltage* of 600.0 V.



Example: Implementation in Proteus Schema

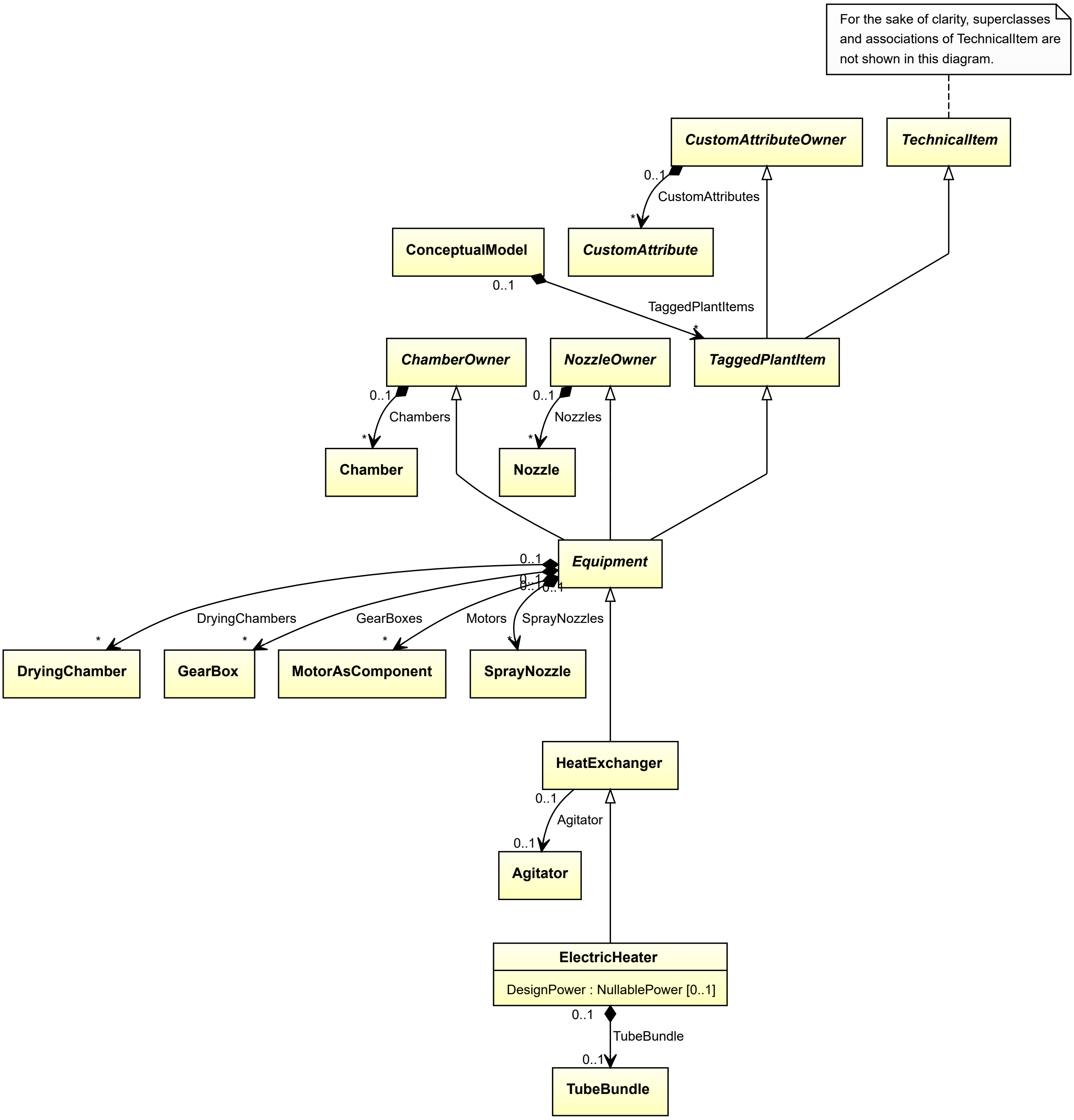
```
<Equipment
    ID="electricGenerator1"
    ComponentClass="ElectricGenerator"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS415709" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignOutletVoltage"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignOutletVoltage"
      Format="double"
      Value="600.0"
      Units="Volt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1347974" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.72. ElectricHeater

7.72.1 Overview

Class

A heater in which electric energy is converted into heat for useful purposes (from http://data.posccaesar.org/rdl/RDS14070475).



Supertypes

- *HeatExchanger*

Attributes (data)

Name	Multiplicity	Type
DesignPower	0..1	*NullablePower*

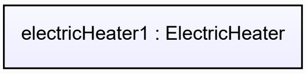
Attributes (composition)

Name	Multiplicity	Type
TubeBundle	0..1	*TubeBundle*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: ELECTRIC HEATER
ComponentClass: ElectricHeater
ComponentClassURI: http://data.posccaesar.org/rdl/RDS14070475

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="electricHeater1"
    ComponentClass="ElectricHeater"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS14070475" ...>
  ...
</Equipment>
```

7.72.2 DesignPower

Attribute (data)

The power for which the *ElectricHeater* is designed.

Multiplicity: 0..1

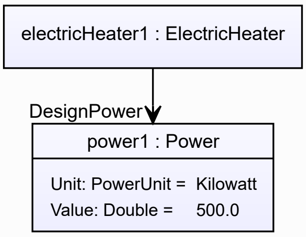
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN POWER
Name: DesignPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignPower

Example

The instance electricHeater1 represents an *ElectricHeater* with a *DesignPower* of 500.0 kW.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="electricHeater1"
    ComponentClass="ElectricHeater"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS14070475" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignPower"
      Format="double"
      Value="500.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.72.3 TubeBundle

Attribute (composition)

The tube bundle of the *ElectricHeater*.

Multiplicity: 0..1

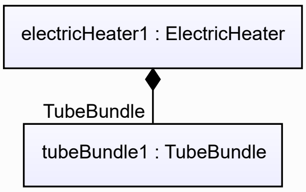
Type: *TubeBundle*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *TubeBundle*) is a child of the `<Equipment>` element for the attribute owner (an *ElectricHeater*).

Example



Example: Implementation in Proteus Schema

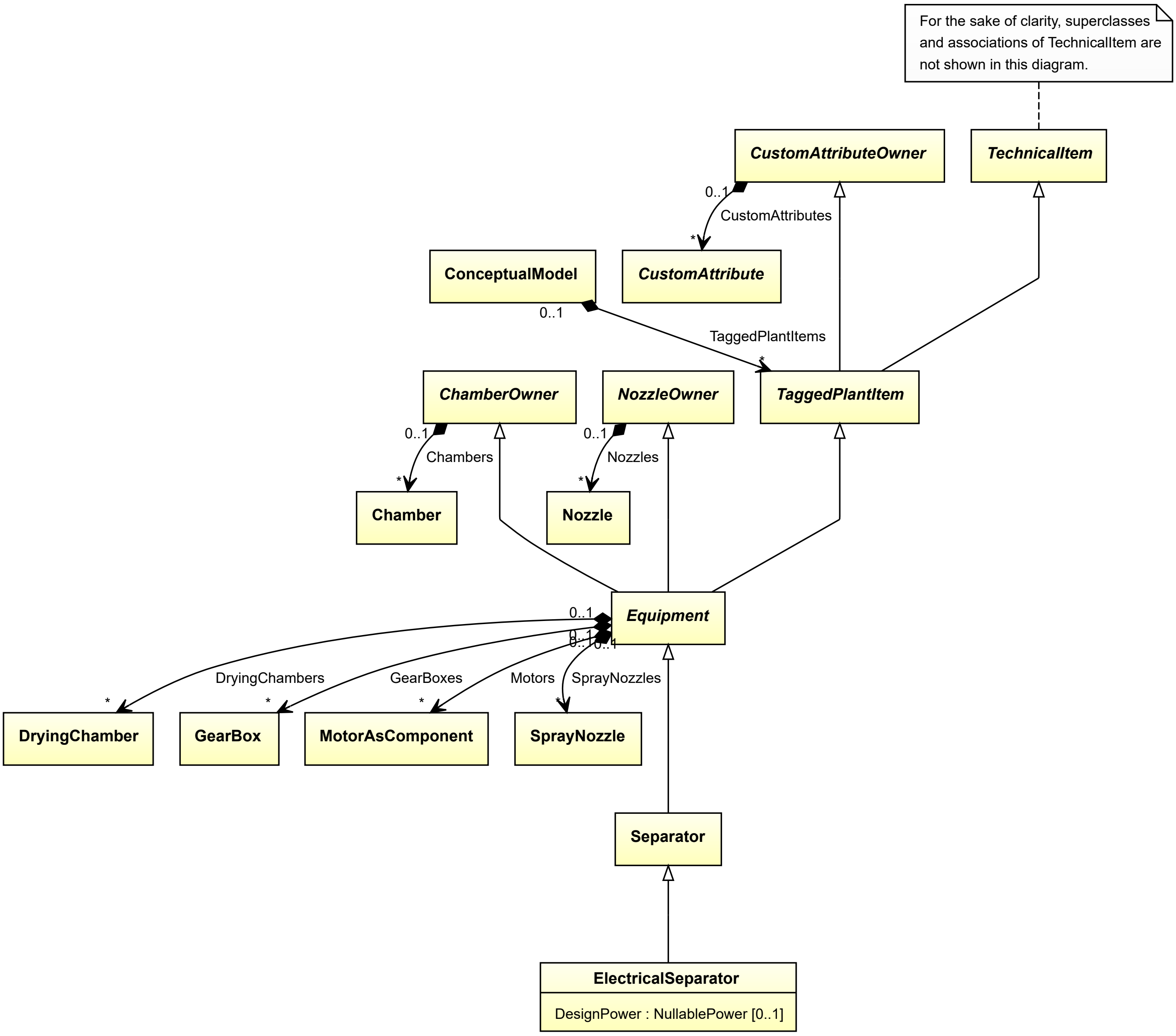
```
<Equipment
    ID="electricHeater1"
    ComponentClass="ElectricHeater"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS14070475" ...>
  ...
  <Equipment
      ID="tubeBundle1"
      ComponentClass="TubeBundle"
      ComponentClassURI="http://data.posccaesar.org/rdl/RDS415259" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.73. ElectricalSeparator

7.73.1 Overview

Class

A separator that uses electromagnetic, magnetic or electrostatic forces to separate phases.



Supertypes

- *Separator*

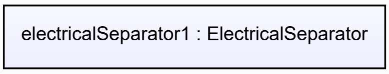
Attributes (data)

Name	Multiplicity	Type
DesignPower	0..1	*NullablePower*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: ELECTRICAL SEPARATOR
ComponentClass: ElectricalSeparator
ComponentClassURI: http://sandbox.dexpi.org/rdl/ElectricalSeparator

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="electricalSeparator1"
    ComponentClass="ElectricalSeparator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ElectricalSeparator" ...>
  ...
</Equipment>
```

7.73.2 DesignPower

Attribute (data)

The power for which the *ElectricalSeparator* is designed.

Multiplicity: 0..1

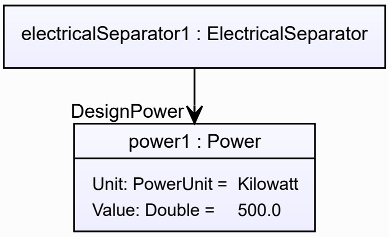
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN POWER
Name: DesignPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignPower

Example

The instance electricalSeparator1 represents an *ElectricalSeparator* with a *DesignPower* of 500.0 kW.


electricalSeparator1 : ElectricalSeparator
DesignPower
power1 : Power
Unit: PowerUnit = Kilowatt
Value: Double = 500.0

Example: Implementation in Proteus Schema

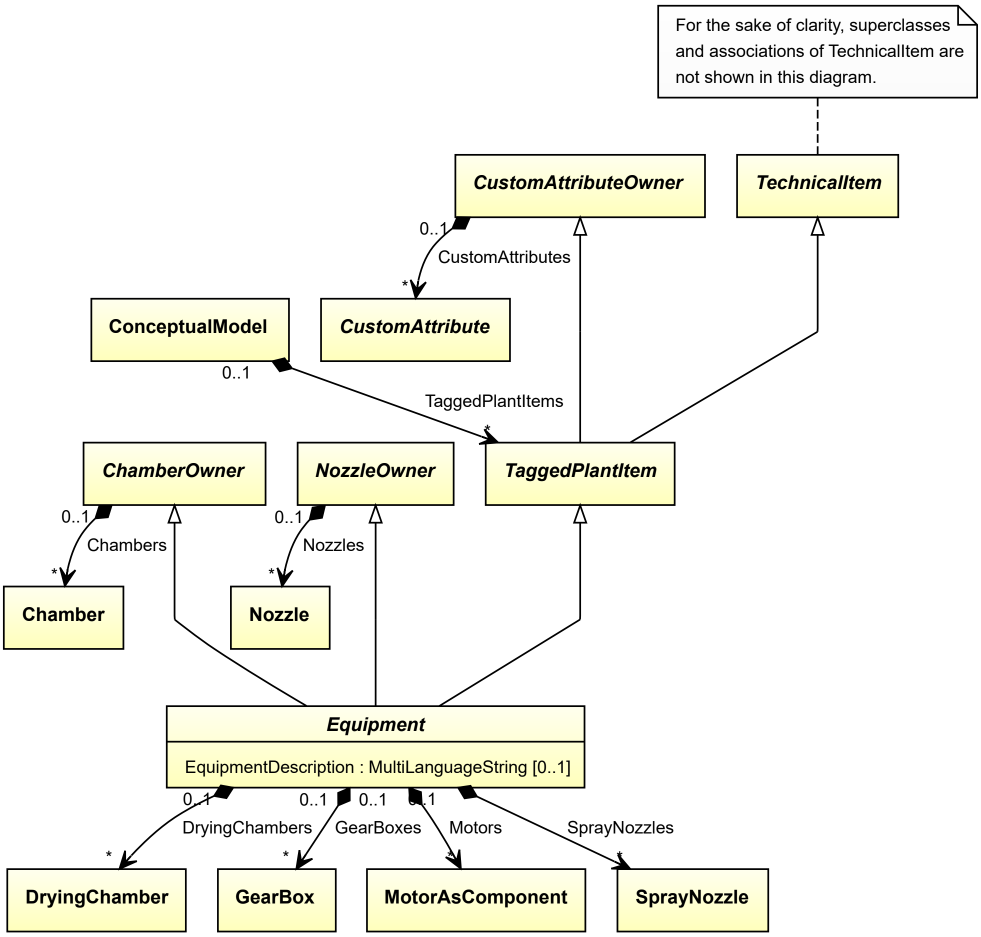
```
<Equipment
    ID="electricalSeparator1"
    ComponentClass="ElectricalSeparator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ElectricalSeparator" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignPower"
      Format="double"
      Value="500.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.74. Equipment

7.74.1 Overview

Abstract class

An apparatus or machine.



For the sake of clarity, superclasses and associations of TechnicalItem are not shown in this diagram.

Supertypes

- *ChamberOwner*
- *NozzleOwner*
- *TaggedPlantItem*

Subtypes

- *Agglomerator*
- *Agitator*
- *Blower*
- *Burner*
- *Centrifuge*
- *Compressor*
- *CoolingTower*
- *CustomEquipment*
- *Dryer*
- *ElectricGenerator*
- *Extruder*
- *Fan*
- *Feeder*
- *Filter*
- *HeatExchanger*
- *Heater*
- *Mill*
- *Mixer*
- *MobileTransportSystem*
- *Motor*
- *PackagingSystem*
- *ProcessColumn*
- *Pump*
- *Separator*
- *Sieve*
- *StationaryTransportSystem*
- *Turbine*
- *Vessel*
- *WasteGasEmitter*
- *Weigher*

Attributes (data)

Name	Multiplicity	Type
EquipmentDescription	0..1	*MultiLanguageString*

Attributes (composition)

Name	Multiplicity	Type
DryingChambers	*	*DryingChamber*
GearBoxes	*	*GearBox*
Motors	*	*MotorAsComponent*
SprayNozzles	*	*SprayNozzle*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*. As *Equipment* is abstract, there is no RDL reference for the class itself; the RDL reference depends on the concrete subclass.
Tag: `<Equipment>`
ComponentClass: *depending on subclass*
ComponentClassURI: *depending on subclass*

Example

As *Equipment* is abstract, we consider *Vessel* as an arbitrary concrete subclass.

Example: Implementation in Proteus Schema

```
<Equipment
    ID="vessel1"
    ComponentClass="Vessel"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS414674" ...>
  ...
</Equipment>
```

7.74.2 DryingChambers

Attribute (composition)

The DryingChambers of the *Equipment*.

Multiplicity: *

Type: *DryingChamber*

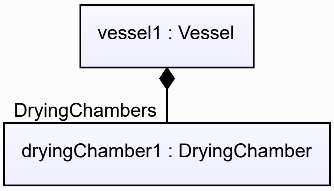
Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *DryingChamber*) is a child of the `<Equipment>` element for the attribute owner (an *Equipment*).

Example

As the owner type *Equipment* is abstract, we consider *Vessel* as an arbitrary concrete subclass.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="vessel1"
    ComponentClass="Vessel"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS414674" ...>
  ...
  <Equipment
      ID="dryingChamber1"
      ComponentClass="DryingChamber"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/DryingChamber" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.74.3 EquipmentDescription

Attribute (data)

A short description of the *Equipment* in natural language.

Multiplicity: 0..1

Type: *MultiLanguageString*

Implementation in Proteus Schema

The attribute is implemented as a *set of DEXPI generic attributes for multi-language string values*.
RDL reference: EQUIPMENT DESCRIPTION ASSIGNMENT CLASS
Name: EquipmentDescriptionAssignmentClass
AttributeURI: http://data.posccaesar.org/rdl/RDS2181987301

Example

As the owning class *Equipment* is abstract, we consider *Vessel* as an arbitrary concrete subclass.

Language	Value
de	Prozessgaskühler
en	process gas cooler

(*MultiLanguageString* with 2 *SingleLanguageString*s)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="vessel1"
    ComponentClass="Vessel"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS414674" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="EquipmentDescriptionAssignmentClass"
      AttributeURI="http://data.posccaesar.org/rdl/RDS2181987301"
      Format="string"
      Language="de"
      Value="Prozessgaskühler" />
    <GenericAttribute
      Name="EquipmentDescriptionAssignmentClass"
      AttributeURI="http://data.posccaesar.org/rdl/RDS2181987301"
      Format="string"
      Language="en"
      Value="process gas cooler" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.74.4 GearBoxes

Attribute (composition)

The gear boxes that are components of the *Equipment*.

Multiplicity: *

Type: *GearBox*

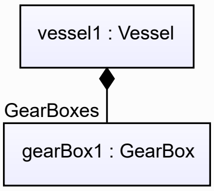
Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *GearBox*) is a child of the `<Equipment>` element for the attribute owner (an *Equipment*).

Example

As the owner type *Equipment* is abstract, we consider *Vessel* as an arbitrary concrete subclass.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="vessel1"
    ComponentClass="Vessel"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS414674" ...>
  ...
  <Equipment
      ID="gearBox1"
      ComponentClass="Gearbox"
      ComponentClassURI="http://data.posccaesar.org/rdl/RDS889514" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.74.5 Motors

Attribute (composition)

The motors that are components of the *Equipment*.

Multiplicity: *

Type: *MotorAsComponent*

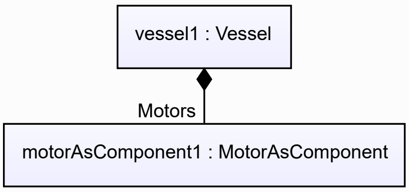
Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *MotorAsComponent*) is a child of the `<Equipment>` element for the attribute owner (an *Equipment*).

Example

As the owner type *Equipment* is abstract, we consider *Vessel* as an arbitrary concrete subclass.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="vessel1"
    ComponentClass="Vessel"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS414674" ...>
  ...
  <Equipment
      ID="motorAsComponent1"
      ComponentClass="MotorAsComponent"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/MotorAsComponent" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.74.6 SprayNozzles

Attribute (composition)

The *SprayNozzles* of the *Equipment*.

Multiplicity: *

Type: *SprayNozzle*

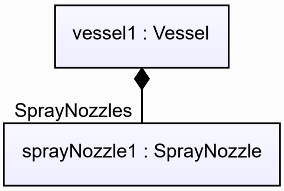
Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *SprayNozzle*) is a child of the `<Equipment>` element for the attribute owner (an *Equipment*).

Example

As the owner type *Equipment* is abstract, we consider *Vessel* as an arbitrary concrete subclass.



Example: Implementation in Proteus Schema

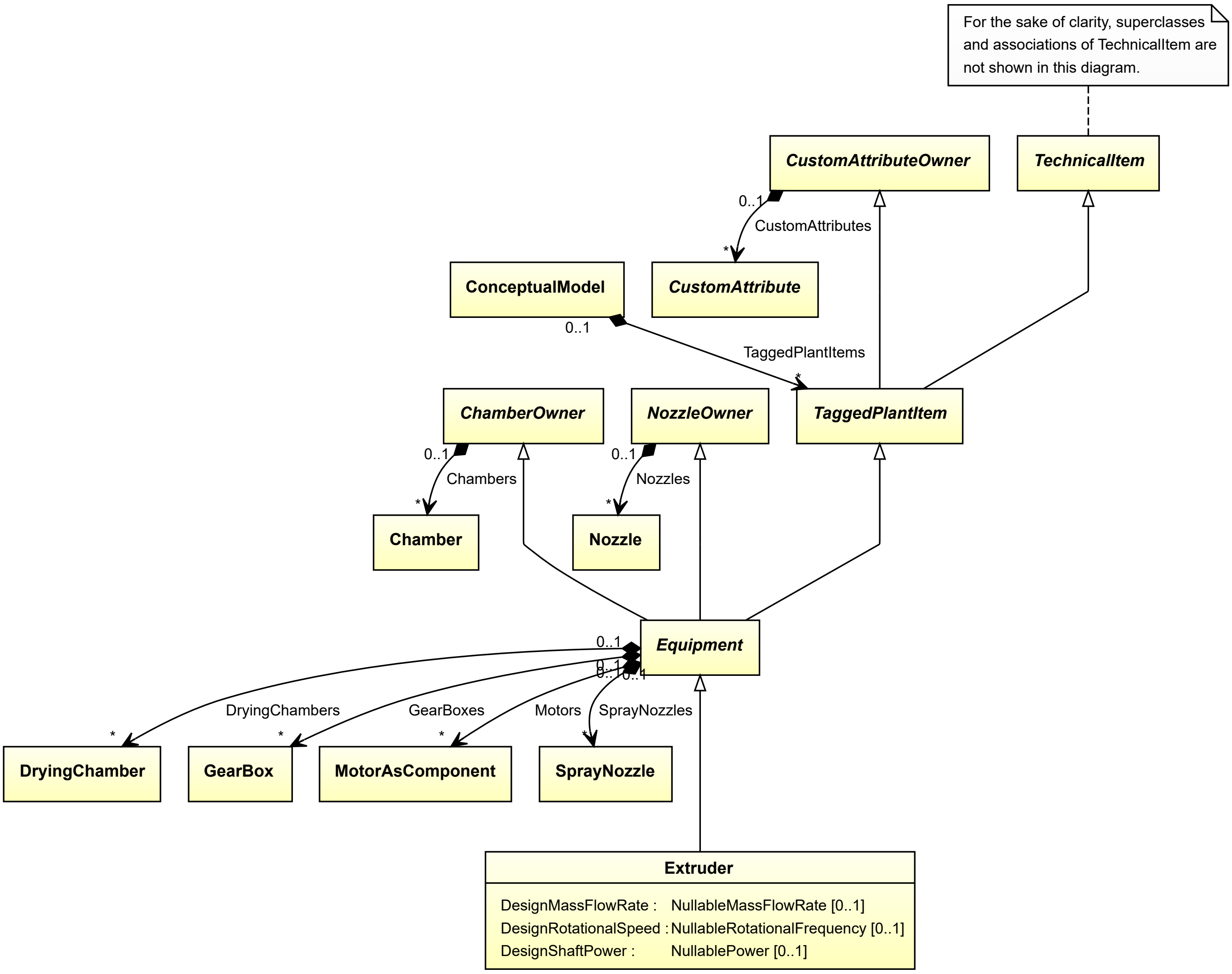
```
<Equipment
    ID="vessel1"
    ComponentClass="Vessel"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS414674" ...>
  ...
  <Equipment
      ID="sprayNozzle1"
      ComponentClass="SprayNozzle"
      ComponentClassURI="http://data.posccaesar.org/rdl/RDS5855670" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.75. Extruder

7.75.1 Overview

Class

A machine that has the capability of extruding (from http://data.15926.org/rdl/RDS394044551).



Supertypes

- *Equipment*

Subtypes

- *CustomExtruder*
- *ReciprocatingExtruder*
- *RotatingExtruder*

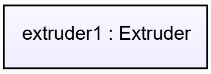
Attributes (data)

Name	Multiplicity	Type
DesignMassFlowRate	0..1	*NullableMassFlowRate*
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*
DesignShaftPower	0..1	*NullablePower*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: EXTRUDER
ComponentClass: Extruder
ComponentClassURI: http://data.posccaesar.org/rdl/RDS394044551

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="extruder1"
    ComponentClass="Extruder"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS394044551" ...>
  ...
</Equipment>
```

7.75.2 DesignMassFlowRate

Attribute (data)

The mass flow rate for which the *Extruder* is designed.

Multiplicity: 0..1

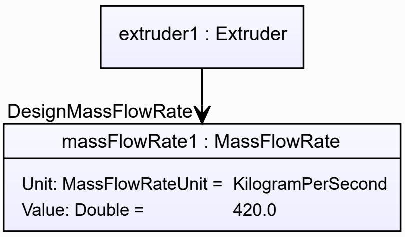
Type: *NullableMassFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN MASS FLOW RATE
Name: DesignMassFlowRate
AttributeURI: http://data.posccaesar.org/rdl/RDS14286182

Example

The instance extruder1 represents an *Extruder* with a *DesignMassFlowRate* of 420.0 kg/s.


extruder1 : Extruder
DesignMassFlowRate
massFlowRate1 : MassFlowRate
Unit: MassFlowRateUnit = KilogramPerSecond
Value: Double = 420.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="extruder1"
    ComponentClass="Extruder"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS394044551" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignMassFlowRate"
      AttributeURI="http://data.posccaesar.org/rdl/RDS14286182"
      Format="double"
      Value="420.0"
      Units="KilogramPerSecond"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1329659" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.75.3 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *Extruder* is designed.

Multiplicity: 0..1

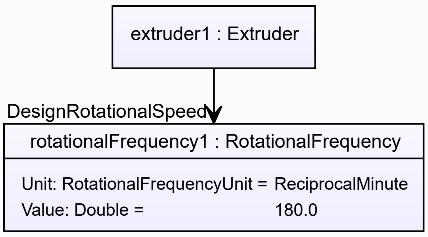
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

Example

The instance extruder1 represents an *Extruder* with a *DesignRotationalSpeed* of 180.0 min^{-1}.


extruder1 : Extruder
DesignRotationalSpeed
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 180.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="extruder1"
    ComponentClass="Extruder"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS394044551" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignRotationalSpeed"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignRotationalSpeed"
      Format="double"
      Value="180.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.75.4 DesignShaftPower

Attribute (data)

The shaft power for which the *Extruder* is designed.

Multiplicity: 0..1

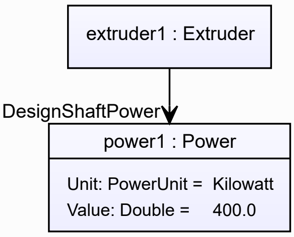
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SHAFT POWER
Name: DesignShaftPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignShaftPower

Example

The instance extruder1 represents an *Extruder* with a *DesignShaftPower* of 400.0 kW.



Example: Implementation in Proteus Schema

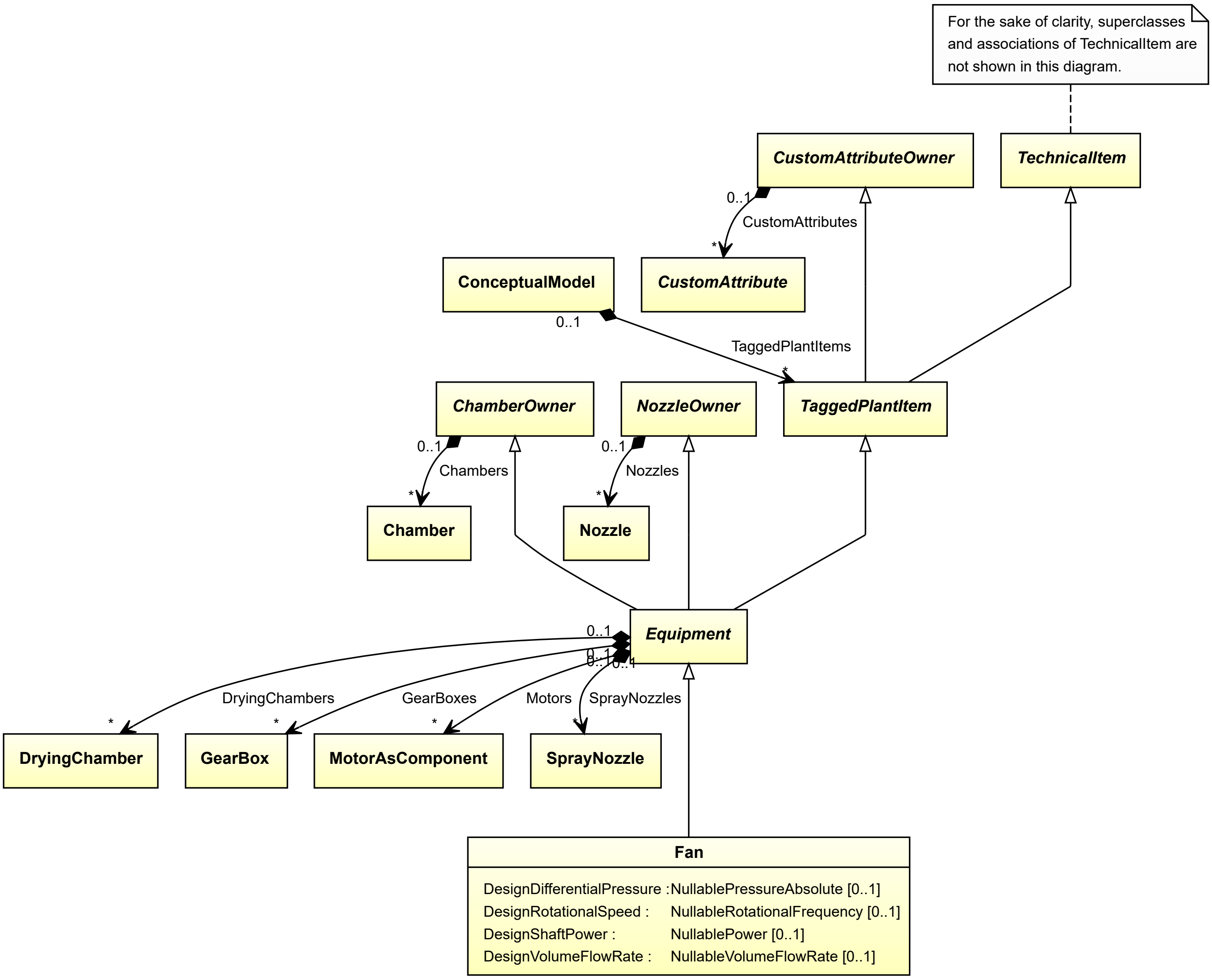
```
<Equipment
    ID="extruder1"
    ComponentClass="Extruder"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS394044551" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignShaftPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignShaftPower"
      Format="double"
      Value="400.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.76. Fan

7.76.1 Overview

Class

An object that is capable of delivering or exhausting volumes of vapour or gas at low differential pressure (from http://data.15926.org/rdl/RDS415169).


For the sake of clarity, superclasses and associations of TechnicalItem are not shown in this diagram.
CustomAttributeOwner
TechnicalItem
0..1
CustomAttributes
*
ConceptualModel
CustomAttribute
0..1
TaggedPlantItems
*
ChamberOwner
NozzleOwner
TaggedPlantItem
0..1
Chambers
*
0..1
Nozzles
*
Chamber
Nozzle
Equipment
0..1
0..1
0..1
0..1
DryingChambers
GearBoxes
Motors
SprayNozzles
*
*
*
*
DryingChamber
GearBox
MotorAsComponent
SprayNozzle
Fan
DesignDifferentialPressure :NullablePressureAbsolute [0..1]
DesignRotationalSpeed : NullableRotationalFrequency [0..1]
DesignShaftPower : NullablePower [0..1]
DesignVolumeFlowRate : NullableVolumeFlowRate [0..1]

Supertypes

- *Equipment*

Subtypes

- *AxialFan*
- *CustomFan*
- *RadialFan*

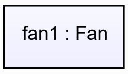
Attributes (data)

Name	Multiplicity	Type
DesignDifferentialPressure	0..1	*NullablePressureAbsolute*
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*
DesignShaftPower	0..1	*NullablePower*
DesignVolumeFlowRate	0..1	*NullableVolumeFlowRate*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: BLOWER FAN
ComponentClass: BlowerFan
ComponentClassURI: http://sandbox.dexpi.org/rdl/BlowerFan

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="fan1"
    ComponentClass="BlowerFan"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/BlowerFan" ...>
  ...
</Equipment>
```

7.76.2 DesignDifferentialPressure

Attribute (data)

The differential pressure for which the *Fan* is designed.

Multiplicity: 0..1

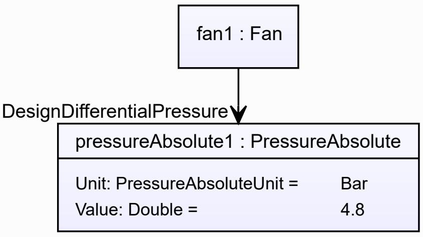
Type: *NullablePressureAbsolute*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN DIFFERENTIAL PRESSURE
Name: DesignDifferentialPressure
AttributeURI: http://sandbox.dexpi.org/rdl/DesignDifferentialPressure

Example

The instance fan1 represents a *Fan* with a *DesignDifferentialPressure* of 4.8 bar.


fan1 : Fan
DesignDifferentialPressure
pressureAbsolute1 : PressureAbsolute
Unit: PressureAbsoluteUnit = Bar
Value: Double = 4.8

Example: Implementation in Proteus Schema

```
<Equipment
    ID="fan1"
    ComponentClass="BlowerFan"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/BlowerFan" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignDifferentialPressure"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignDifferentialPressure"
      Format="double"
      Value="4.8"
      Units="Bar"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1314539" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.76.3 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *Fan* is designed.

Multiplicity: 0..1

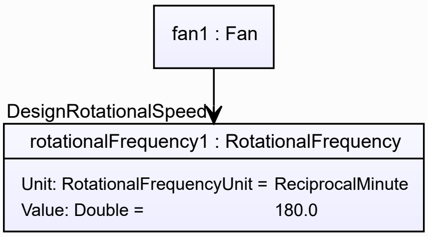
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

Example

The instance fan1 represents a *Fan* with a *DesignRotationalSpeed* of 180.0 min^{-1}.


fan1 : Fan
DesignRotationalSpeed
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 180.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="fan1"
    ComponentClass="BlowerFan"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/BlowerFan" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignRotationalSpeed"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignRotationalSpeed"
      Format="double"
      Value="180.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.76.4 DesignShaftPower

Attribute (data)

The shaft power for which the *Fan* is designed.

Multiplicity: 0..1

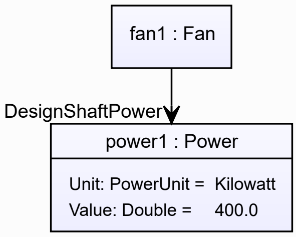
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SHAFT POWER
Name: DesignShaftPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignShaftPower

Example

The instance fan1 represents a *Fan* with a *DesignShaftPower* of 400.0 kW.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="fan1"
    ComponentClass="BlowerFan"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/BlowerFan" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignShaftPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignShaftPower"
      Format="double"
      Value="400.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.76.5 DesignVolumeFlowRate

Attribute (data)

The volume flow rate for which the *Fan* is designed.

Multiplicity: 0..1

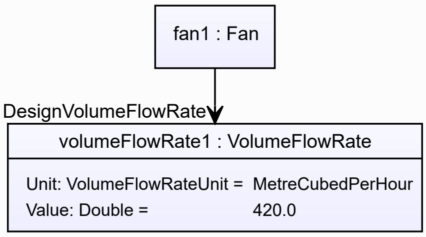
Type: *NullableVolumeFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN VOLUME FLOW RATE
Name: DesignVolumeFlowRate
AttributeURI: http://data.posccaesar.org/rdl/RDS14286227

Example

The instance fan1 represents a *Fan* with a *DesignVolumeFlowRate* of 420.0 m^3/h.


fan1 : Fan
DesignVolumeFlowRate
volumeFlowRate1 : VolumeFlowRate
Unit: VolumeFlowRateUnit = MetreCubedPerHour
Value: Double = 420.0

Example: Implementation in Proteus Schema

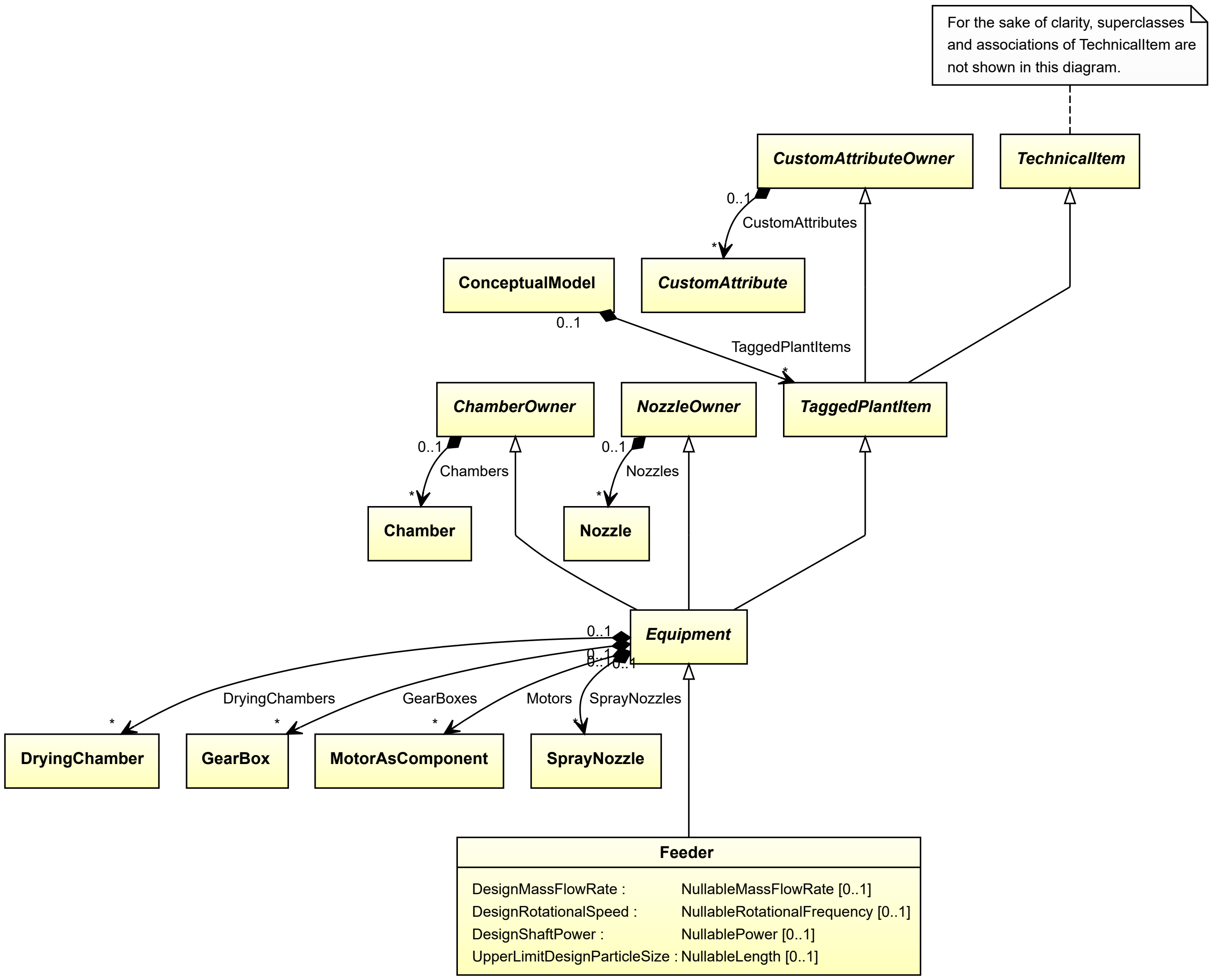
```
<Equipment
    ID="fan1"
    ComponentClass="BlowerFan"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/BlowerFan" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignVolumeFlowRate"
      AttributeURI="http://data.posccaesar.org/rdl/RDS14286227"
      Format="double"
      Value="420.0"
      Units="MetreCubedPerHour"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1321064" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.77. Feeder

7.77.1 Overview

Class

A closed fluid transporter that is a gathering line tied into a trunk line (from http://data.15926.org/rdl/RDS300644).



Supertypes

- *Equipment*

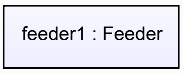
Attributes (data)

Name	Multiplicity	Type
DesignMassFlowRate	0..1	*NullableMassFlowRate*
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*
DesignShaftPower	0..1	*NullablePower*
UpperLimitDesignParticleSize	0..1	*NullableLength*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: FEEDER
ComponentClass: Feeder
ComponentClassURI: http://data.posccaesar.org/rdl/RDS300644

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="feeder1"
    ComponentClass="Feeder"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS300644" ...>
  ...
</Equipment>
```

7.77.2 DesignMassFlowRate

Attribute (data)

The mass flow rate for which the *Feeder* is designed.

Multiplicity: 0..1

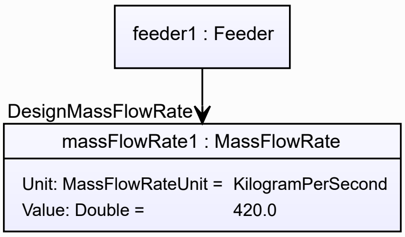
Type: *NullableMassFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN MASS FLOW RATE
Name: DesignMassFlowRate
AttributeURI: http://data.posccaesar.org/rdl/RDS14286182

Example

The instance feeder1 represents a *Feeder* with a *DesignMassFlowRate* of 420.0 kg/s.


feeder1 : Feeder
DesignMassFlowRate
massFlowRate1 : MassFlowRate
Unit: MassFlowRateUnit = KilogramPerSecond
Value: Double = 420.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="feeder1"
    ComponentClass="Feeder"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS300644" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignMassFlowRate"
      AttributeURI="http://data.posccaesar.org/rdl/RDS14286182"
      Format="double"
      Value="420.0"
      Units="KilogramPerSecond"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1329659" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.77.3 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *Feeder* is designed.

Multiplicity: 0..1

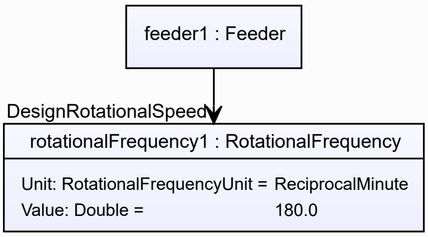
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

Example

The instance feeder1 represents a *Feeder* with a *DesignRotationalSpeed* of 180.0 min^{-1}.


feeder1 : Feeder
DesignRotationalSpeed
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 180.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="feeder1"
    ComponentClass="Feeder"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS300644" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignRotationalSpeed"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignRotationalSpeed"
      Format="double"
      Value="180.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.77.4 DesignShaftPower

Attribute (data)

The shaft power for which the *Feeder* is designed.

Multiplicity: 0..1

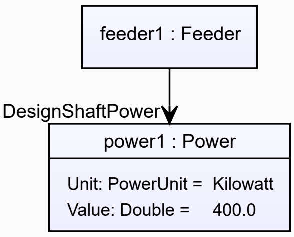
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SHAFT POWER
Name: DesignShaftPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignShaftPower

Example

The instance feeder1 represents a *Feeder* with a *DesignShaftPower* of 400.0 kW.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="feeder1"
    ComponentClass="Feeder"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS300644" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignShaftPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignShaftPower"
      Format="double"
      Value="400.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.77.5 UpperLimitDesignParticleSize

Attribute (data)

The upper limit for the particle size for which the *Feeder* is designed.

Multiplicity: 0..1

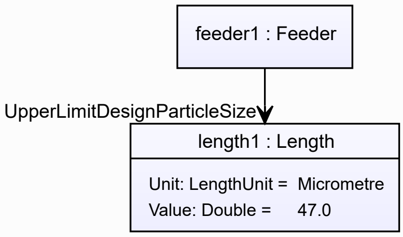
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT DESIGN PARTICLE SIZE
Name: UpperLimitDesignParticleSize
AttributeURI: http://sandbox.dexpi.org/rdl/UpperLimitDesignParticleSize

Example

The instance feeder1 represents a *Feeder* with an *UpperLimitDesignParticleSize* of 47.0 µm.



Example: Implementation in Proteus Schema

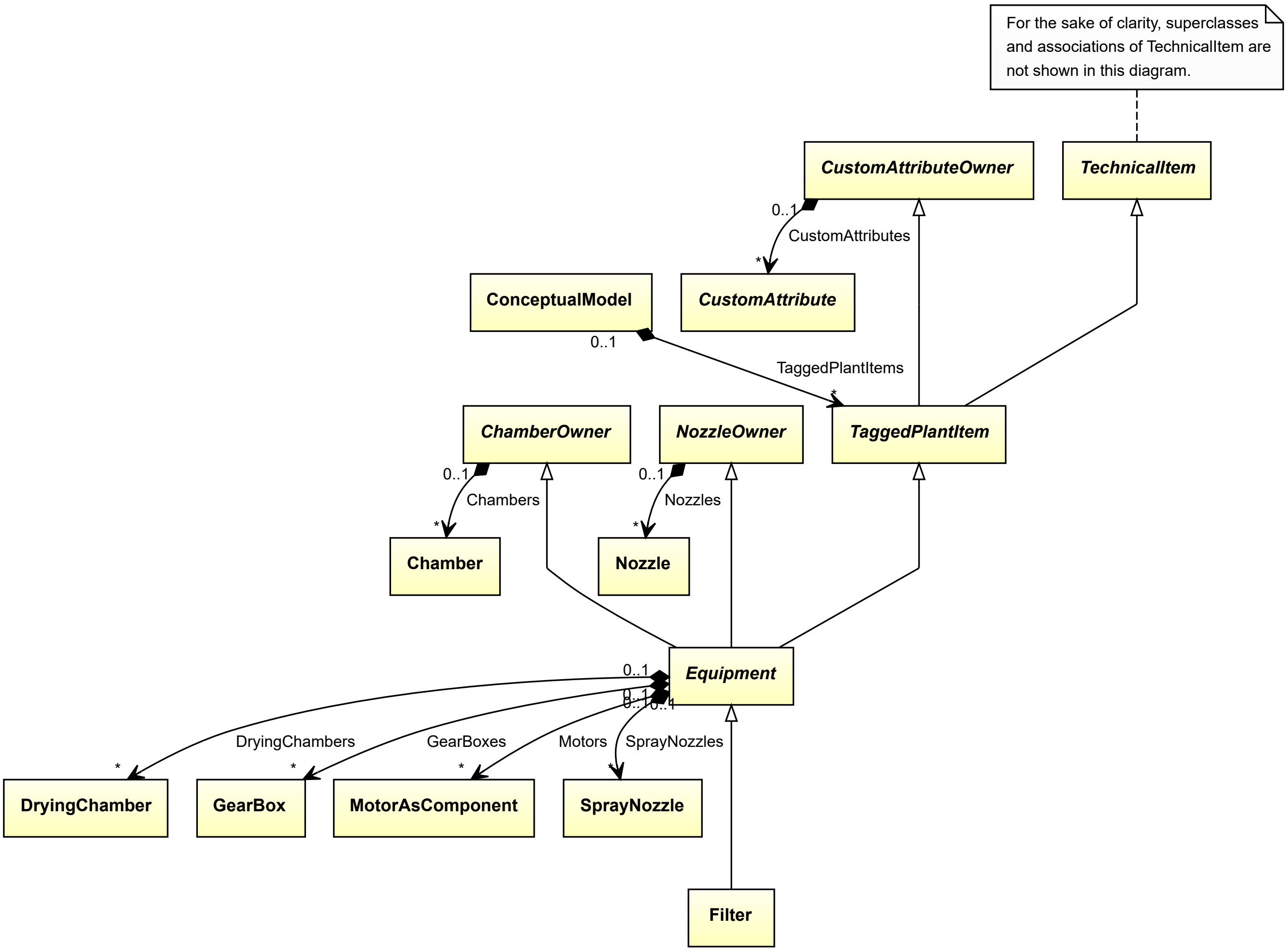
```
<Equipment
    ID="feeder1"
    ComponentClass="Feeder"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS300644" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="UpperLimitDesignParticleSize"
      AttributeURI="http://sandbox.dexpi.org/rdl/UpperLimitDesignParticleSize"
      Format="double"
      Value="47.0"
      Units="Micrometre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1351529" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.78. Filter

7.78.1 Overview

Class

An apparatus or machine that is capable of filtering (from http://data.15926.org/rdl/RDS300689).



For the sake of clarity, superclasses and associations of TechnicalItem are not shown in this diagram.

Supertypes

- *Equipment*

Subtypes

- *CustomFilter*
- *GasFilter*
- *LiquidFilter*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: FILTER
ComponentClass: Filter
ComponentClassURI: http://data.posccaesar.org/rdl/RDS300689

Example

filter1 : Filter

Example: Implementation in Proteus Schema

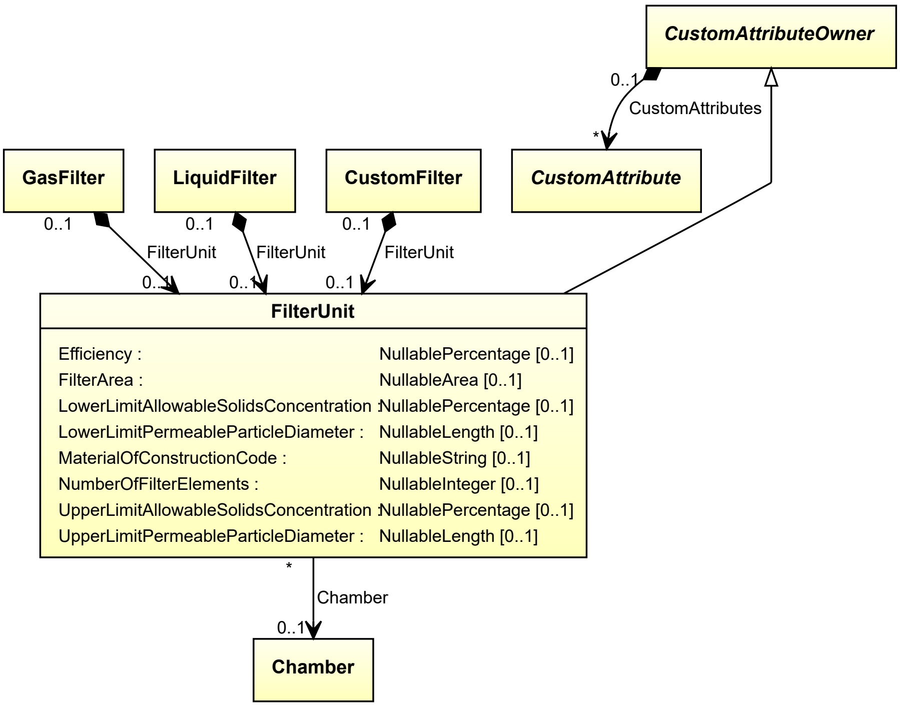
```
<Equipment
    ID="filter1"
    ComponentClass="Filter"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS300689" ...>
  ...
</Equipment>
```

7.79. FilterUnit

7.79.1 Overview

Class

The filtering unit as part of a filter.


CustomAttributeOwner
0..1
CustomAttributes
*
GasFilter
LiquidFilter
CustomFilter
CustomAttribute
0..1
0..1
0..1
FilterUnit
FilterUnit
FilterUnit
0..1
0..1
0..1
FilterUnit
Efficiency : NullablePercentage [0..1]
FilterArea : NullableArea [0..1]
LowerLimitAllowableSolidsConcentration : NullablePercentage [0..1]
LowerLimitPermeableParticleDiameter : NullableLength [0..1]
MaterialOfConstructionCode : NullableString [0..1]
NumberOfFilterElements : NullableInteger [0..1]
UpperLimitAllowableSolidsConcentration : NullablePercentage [0..1]
UpperLimitPermeableParticleDiameter : NullableLength [0..1]
*
Chamber
0..1
Chamber

Supertypes

- *CustomAttributeOwner*

Attributes (data)

Name	Multiplicity	Type
Efficiency	0..1	*NullablePercentage*
FilterArea	0..1	*NullableArea*
LowerLimitAllowableSolidsConcentration	0..1	*NullablePercentage*
LowerLimitPermeableParticleDiameter	0..1	*NullableLength*
MaterialOfConstructionCode	0..1	*NullableString*
NumberOfFilterElements	0..1	*NullableInteger*
UpperLimitAllowableSolidsConcentration	0..1	*NullablePercentage*
UpperLimitPermeableParticleDiameter	0..1	*NullableLength*

Attributes (reference)

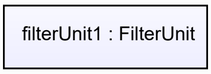
Name	Multiplicity	Type
Chamber	0..1	*Chamber*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: FILTER UNIT

ComponentClass: FilterUnit
ComponentClassURI: http://sandbox.dexpi.org/rdl/FilterUnit

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="filterUnit1"
    ComponentClass="FilterUnit"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/FilterUnit" ...>
  ...
</Equipment>
```

7.79.2 Chamber

Attribute (reference)

The *Chamber* in which the *FilterUnit* is located, if applicable. The Chamber must be a component of the same object as the FilterUnit.

Multiplicity: 0..1

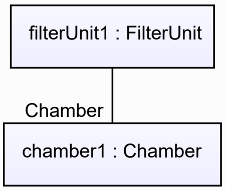
Type: *Chamber*

Opposite multiplicity: 0..*

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: "is located in"
Opposite association type: "is the location of"

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="filterUnit1"
    ComponentClass="FilterUnit"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/FilterUnit" ...>
  ...
  <Association
    Type="is located in"
    ItemID="chamber1" />
  ...
<Equipment />
...
<Equipment
    ID="chamber1"
    ComponentClass="Chamber"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS903151421" ...>
  ...
  <Association
    Type="is the location of"
    ItemID="filterUnit1" />
  ...
<Equipment />
```

7.79.3 Efficiency

Attribute (data)

The efficiency of the *FilterUnit*.

Multiplicity: 0..1

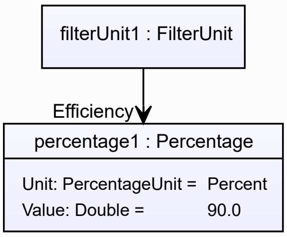
Type: *NullablePercentage*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: EFFICIENCY
Name: Efficiency
AttributeURI: http://data.posccaesar.org/rdl/RDS362654

Example

The instance filterUnit1 represents a *FilterUnit* with an *Efficiency* of 90.0 ???.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="filterUnit1"
    ComponentClass="FilterUnit"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/FilterUnit" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="Efficiency"
      AttributeURI="http://data.posccaesar.org/rdl/RDS362654"
      Format="double"
      Value="90.0"
      Units="Percent"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1317959" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.79.4 FilterArea

Attribute (data)

The filter area of the *FilterUnit*.

Multiplicity: 0..1

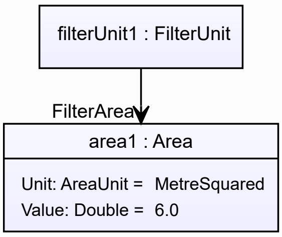
Type: *NullableArea*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: FILTER AREA
Name: FilterArea
AttributeURI: http://sandbox.dexpi.org/rdl/FilterArea

Example

The instance filterUnit1 represents a *FilterUnit* with a *FilterArea* of 6.0 m^2.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="filterUnit1"
    ComponentClass="FilterUnit"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/FilterUnit" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="FilterArea"
      AttributeURI="http://sandbox.dexpi.org/rdl/FilterArea"
      Format="double"
      Value="6.0"
      Units="MetreSquared"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1358009" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.79.5 LowerLimitAllowableSolidsConcentration

Attribute (data)

The lower limit for the concentration for solids.

Multiplicity: 0..1

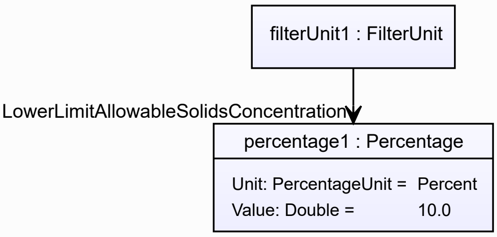
Type: *NullablePercentage*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: LOWER LIMIT ALLOWABLE SOLIDS CONCENTRATION
Name: LowerLimitAllowableSolidsConcentration
AttributeURI: http://sandbox.dexpi.org/rdl/LowerLimitAllowableSolidsConcentration

Example

The instance filterUnit1 represents a *FilterUnit* with a *LowerLimitAllowableSolidsConcentration* of 10.0 ???.


filterUnit1 : FilterUnit
LowerLimitAllowableSolidsConcentration
percentage1 : Percentage
Unit: PercentageUnit = Percent
Value: Double = 10.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="filterUnit1"
    ComponentClass="FilterUnit"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/FilterUnit" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="LowerLimitAllowableSolidsConcentration"
      AttributeURI="http://sandbox.dexpi.org/rdl/LowerLimitAllowableSolidsConcentration"
      Format="double"
      Value="10.0"
      Units="Percent"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1317959" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.79.6 LowerLimitPermeableParticleDiameter

Attribute (data)

The lower limit for the particle size.

Multiplicity: 0..1

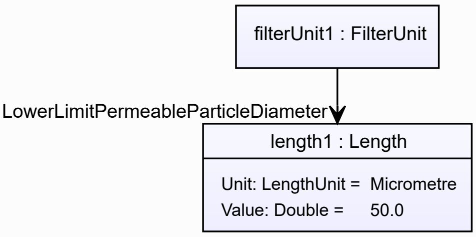
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: LOWER LIMIT PERMEABLE PARTICLE DIAMETER
Name: LowerLimitPermeableParticleDiameter
AttributeURI: http://sandbox.dexpi.org/rdl/LowerLimitPermeableParticleDiameter

Example

The instance filterUnit1 represents a *FilterUnit* with a *LowerLimitPermeableParticleDiameter* of 50.0 µm.


filterUnit1 : FilterUnit
LowerLimitPermeableParticleDiameter
length1 : Length
Unit: LengthUnit = Micrometre
Value: Double = 50.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="filterUnit1"
    ComponentClass="FilterUnit"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/FilterUnit" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="LowerLimitPermeableParticleDiameter"
      AttributeURI="http://sandbox.dexpi.org/rdl/LowerLimitPermeableParticleDiameter"
      Format="double"
      Value="50.0"
      Units="Micrometre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1351529" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.79.7 MaterialOfConstructionCode

Attribute (data)

A code that gives the material of construction of the *FilterUnit*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: MATERIAL OF CONSTRUCTION CODE ASSIGNMENT CLASS
Name: MaterialOfConstructionCodeAssignmentClass
AttributeURI: http://data.posccaesar.org/rdl/RDS1460719741

Example

“1.4306” (*String*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="filterUnit1"
    ComponentClass="FilterUnit"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/FilterUnit" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="MaterialOfConstructionCodeAssignmentClass"
      AttributeURI="http://data.posccaesar.org/rdl/RDS1460719741"
      Format="string"
      Value="1.4306" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.79.8 NumberOfFilterElements

Attribute (data)

The number of filter elements in the *FilterUnit*.

Multiplicity: 0..1

Type: *NullableInteger*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for integer values*.
RDL reference: NUMBER OF FILTER ELEMENTS
Name: NumberOfFilterElements
AttributeURI: http://sandbox.dexpi.org/rdl/NumberOfFilterElements

Example

36 (*Integer*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="filterUnit1"
    ComponentClass="FilterUnit"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/FilterUnit" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NumberOfFilterElements"
      AttributeURI="http://sandbox.dexpi.org/rdl/NumberOfFilterElements"
      Format="integer"
      Value="36" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.79.9 UpperLimitAllowableSolidsConcentration

Attribute (data)

The upper limit for the concentration for solids.

Multiplicity: 0..1

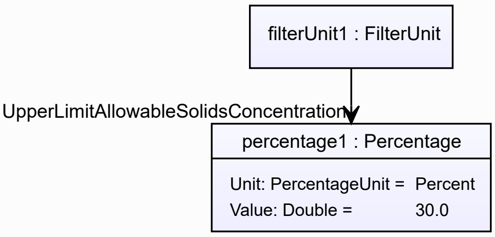
Type: *NullablePercentage*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT ALLOWABLE SOLIDS CONCENTRATION
Name: UpperLimitAllowableSolidsConcentration
AttributeURI: http://sandbox.dexpi.org/rdl/UpperLimitAllowableSolidsConcentration

Example

The instance filterUnit1 represents a *FilterUnit* with an *UpperLimitAllowableSolidsConcentration* of 30.0 ???.


filterUnit1 : FilterUnit
UpperLimitAllowableSolidsConcentration
percentage1 : Percentage
Unit: PercentageUnit = Percent
Value: Double = 30.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="filterUnit1"
    ComponentClass="FilterUnit"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/FilterUnit" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="UpperLimitAllowableSolidsConcentration"
      AttributeURI="http://sandbox.dexpi.org/rdl/UpperLimitAllowableSolidsConcentration"
      Format="double"
      Value="30.0"
      Units="Percent"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1317959" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.79.10 UpperLimitPermeableParticleDiameter

Attribute (data)

The maximum of the particle size passing through the *FilterUnit*.

Multiplicity: 0..1

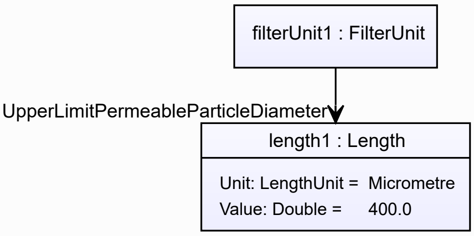
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT PERMEABLE PARTICLE DIAMETER
Name: UpperLimitPermeableParticleDiameter
AttributeURI: http://sandbox.dexpi.org/rdl/UpperLimitPermeableParticleDiameter

Example

The instance filterUnit1 represents a *FilterUnit* with an *UpperLimitPermeableParticleDiameter* of 400.0 µm.


filterUnit1 : FilterUnit
UpperLimitPermeableParticleDiameter
length1 : Length
Unit: LengthUnit = Micrometre
Value: Double = 400.0

Example: Implementation in Proteus Schema

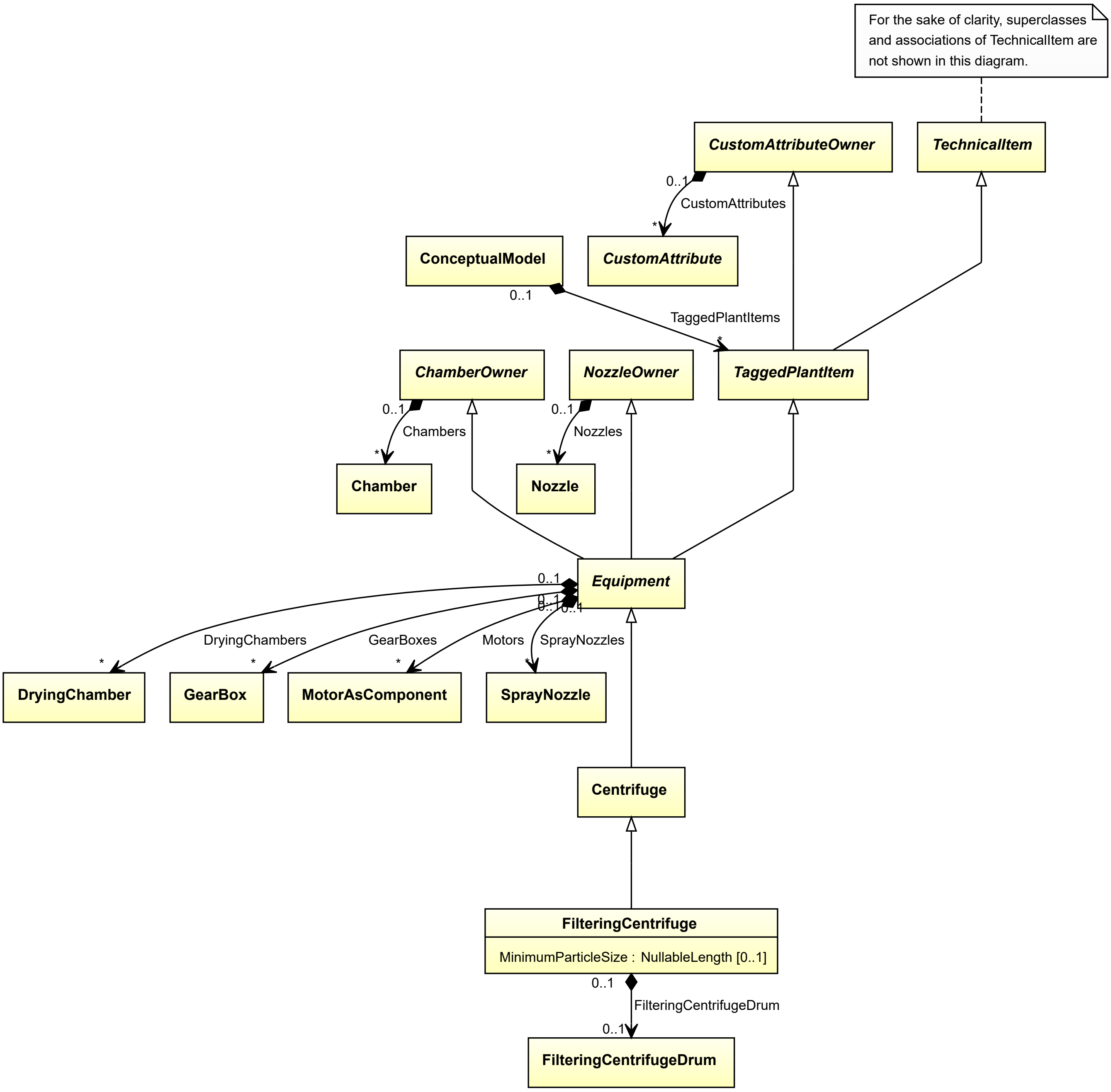
```
<Equipment
    ID="filterUnit1"
    ComponentClass="FilterUnit"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/FilterUnit" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="UpperLimitPermeableParticleDiameter"
      AttributeURI="http://sandbox.dexpi.org/rdl/UpperLimitPermeableParticleDiameter"
      Format="double"
      Value="400.0"
      Units="Micrometre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1351529" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.80. FilteringCentrifuge

7.80.1 Overview

Class

A centrifuge intended to separate solids from liquids by centrifugal process based on particle size.



Supertypes

- *Centrifuge*

Attributes (data)

Name	Multiplicity	Type
MinimumParticleSize	0..1	*NullableLength*

Attributes (composition)

Name	Multiplicity	Type
FilteringCentrifugeDrum	0..1	*FilteringCentrifugeDrum*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: FILTERING CENTRIFUGE
ComponentClass: FilteringCentrifuge
ComponentClassURI: http://sandbox.dexpi.org/rdl/FilteringCentrifuge

Example

filteringCentrifuge1 : FilteringCentrifuge

Example: Implementation in Proteus Schema

```
<Equipment
    ID="filteringCentrifuge1"
    ComponentClass="FilteringCentrifuge"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/FilteringCentrifuge" ...>
  ...
</Equipment>
```

7.80.2 FilteringCentrifugeDrum

Attribute (composition)

The filtering centrifuge drum of the *FilteringCentrifuge*.

Multiplicity: 0..1

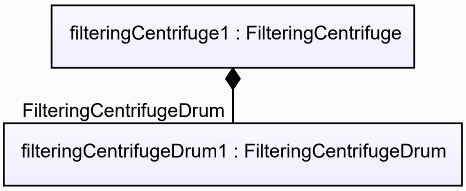
Type: *FilteringCentrifugeDrum*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *FilteringCentrifugeDrum*) is a child of the `<Equipment>` element for the attribute owner (a *FilteringCentrifuge*).

Example


filteringCentrifuge1 : FilteringCentrifuge
FilteringCentrifugeDrum
filteringCentrifugeDrum1 : FilteringCentrifugeDrum

Example: Implementation in Proteus Schema

```
<Equipment
    ID="filteringCentrifuge1"
    ComponentClass="FilteringCentrifuge"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/FilteringCentrifuge" ...>
  ...
  <Equipment
      ID="filteringCentrifugeDrum1"
      ComponentClass="FilteringCentrifugeDrum"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/FilteringCentrifugeDrum" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.80.3 MinimumParticleSize

Attribute (data)

The minimum particle size of the *FilteringCentrifuge*.

Multiplicity: 0..1

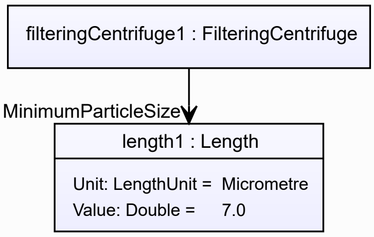
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: MINIMUM PARTICLE SIZE
Name: MinimumParticleSize
AttributeURI: http://sandbox.dexpi.org/rdl/MinimumParticleSize

Example

The instance filteringCentrifuge1 represents a *FilteringCentrifuge* with a *MinimumParticleSize* of 7.0 µm.


filteringCentrifuge1 : FilteringCentrifuge
MinimumParticleSize
length1 : Length
Unit: LengthUnit = Micrometre
Value: Double = 7.0

Example: Implementation in Proteus Schema

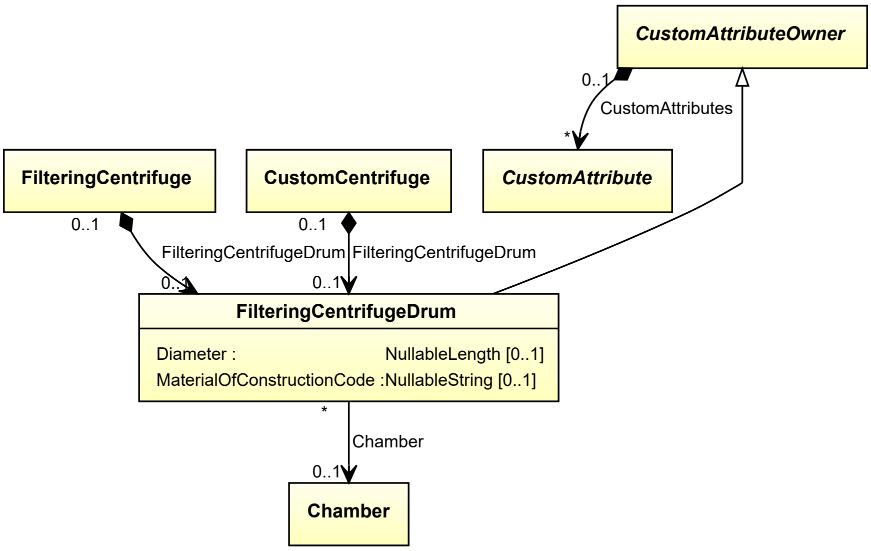
```
<Equipment
    ID="filteringCentrifuge1"
    ComponentClass="FilteringCentrifuge"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/FilteringCentrifuge" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="MinimumParticleSize"
      AttributeURI="http://sandbox.dexpi.org/rdl/MinimumParticleSize"
      Format="double"
      Value="7.0"
      Units="Micrometre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1351529" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.81. FilteringCentrifugeDrum

7.81.1 Overview

Class

A drum being a component of a FilteringCentrifuge.



Supertypes

- *CustomAttributeOwner*

Attributes (data)

Name	Multiplicity	Type
Diameter	0..1	*NullableLength*
MaterialOfConstructionCode	0..1	*NullableString*

Attributes (reference)

Name	Multiplicity	Type
Chamber	0..1	*Chamber*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: FILTERING CENTRIFUGE DRUM
ComponentClass: FilteringCentrifugeDrum
ComponentClassURI: http://sandbox.dexpi.org/rdl/FilteringCentrifugeDrum

Example

Example: Implementation in Proteus Schema

```
<Equipment
    ID="filteringCentrifugeDrum1"
    ComponentClass="FilteringCentrifugeDrum"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/FilteringCentrifugeDrum" ...>
  ...
</Equipment>
```

7.81.2 Chamber

Attribute (reference)

The *Chamber* in which the *FilteringCentrifugeDrum* is located, if applicable. The Chamber must be a component of the same object as the *FilteringCentrifugeDrum*.

Multiplicity: 0..1

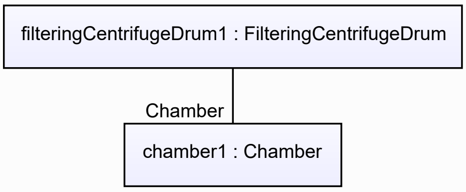
Type: *Chamber*

Opposite multiplicity: 0..*

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: `"is located in"`
Opposite association type: `"is the location of"`

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="filteringCentrifugeDrum1"
    ComponentClass="FilteringCentrifugeDrum"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/FilteringCentrifugeDrum" ...>
  ...
  <Association
    Type="is located in"
    ItemID="chamber1" />
  ...
<Equipment />
...
<Equipment
    ID="chamber1"
    ComponentClass="Chamber"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS903151421" ...>
  ...
  <Association
    Type="is the location of"
    ItemID="filteringCentrifugeDrum1" />
  ...
<Equipment />
```

7.81.3 Diameter

Attribute (data)

The diameter of the *FilteringCentrifugeDrum*.

Multiplicity: 0..1

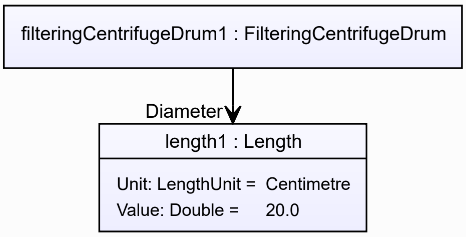
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DIAMETER
Name: Diameter
AttributeURI: http://data.posccaesar.org/rdl/RDS350954

Example

The instance filteringCentrifugeDrum1 represents a *FilteringCentrifugeDrum* with a *Diameter* of 20.0 cm.


filteringCentrifugeDrum1 : FilteringCentrifugeDrum
Diameter
length1 : Length
Unit: LengthUnit = Centimetre
Value: Double = 20.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="filteringCentrifugeDrum1"
    ComponentClass="FilteringCentrifugeDrum"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/FilteringCentrifugeDrum" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="Diameter"
      AttributeURI="http://data.posccaesar.org/rdl/RDS350954"
      Format="double"
      Value="20.0"
      Units="Centimetre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1318004" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.81.4 MaterialOfConstructionCode

Attribute (data)

A code that gives the material of construction of the *FilteringCentrifugeDrum*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: MATERIAL OF CONSTRUCTION CODE ASSIGNMENT CLASS
Name: MaterialOfConstructionCodeAssignmentClass
AttributeURI: http://data.posccaesar.org/rdl/RDS1460719741

Example

“1.4306” (*String*)

Example: Implementation in Proteus Schema

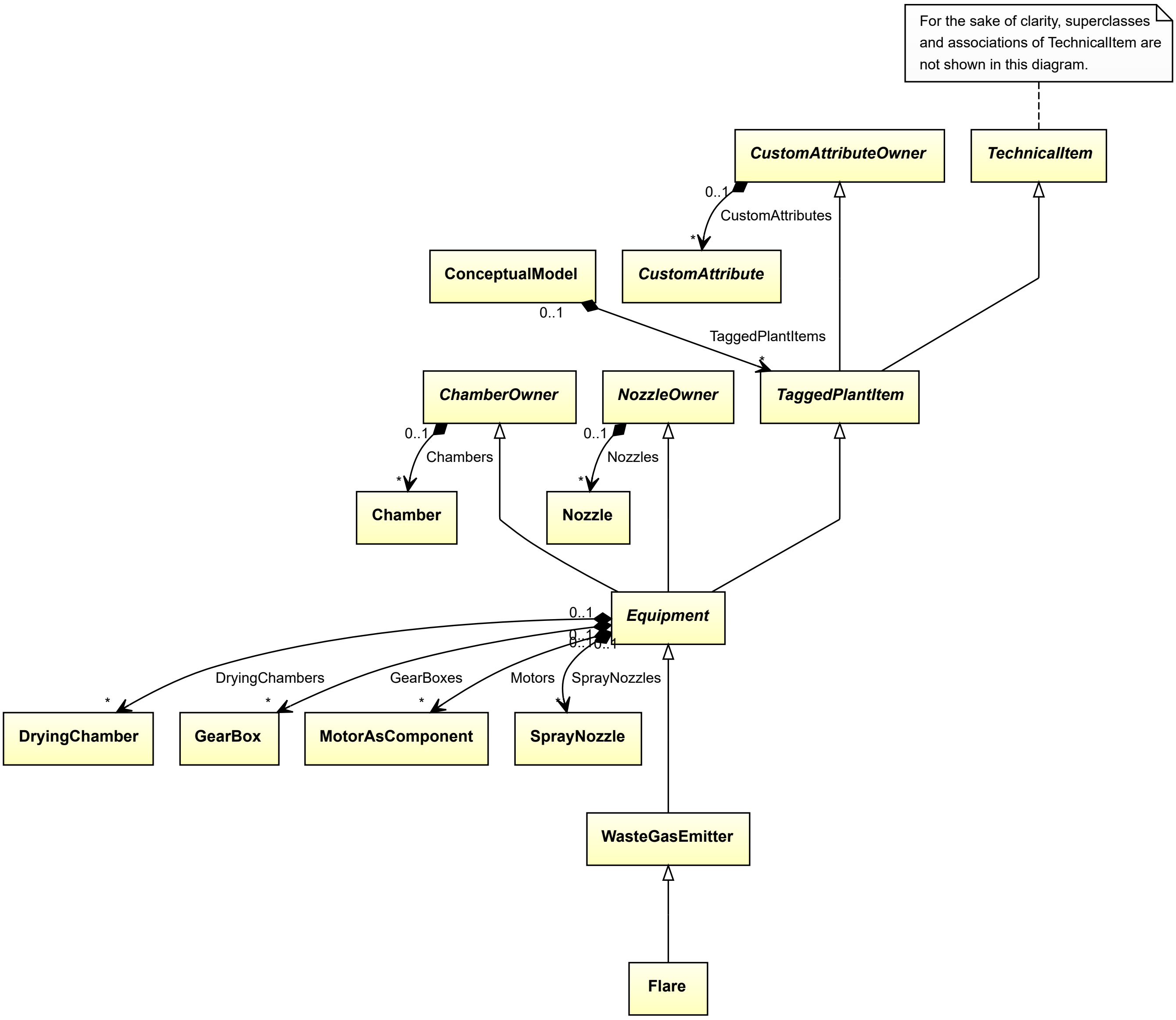
```
<Equipment
    ID="filteringCentrifugeDrum1"
    ComponentClass="FilteringCentrifugeDrum"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/FilteringCentrifugeDrum" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="MaterialOfConstructionCodeAssignmentClass"
      AttributeURI="http://data.posccaesar.org/rdl/RDS1460719741"
      Format="string"
      Value="1.4306" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.82. Flare

7.82.1 Overview

Class

An artefact and waste gas emitter that is intended to burn waste gas in secure distance from the plant or platform.



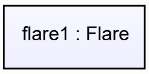
Supertypes

- *WasteGasEmitter*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: FLARE
ComponentClass: Flare
ComponentClassURI: http://sandbox.dexpi.org/rdl/Flare

Example



Example: Implementation in Proteus Schema

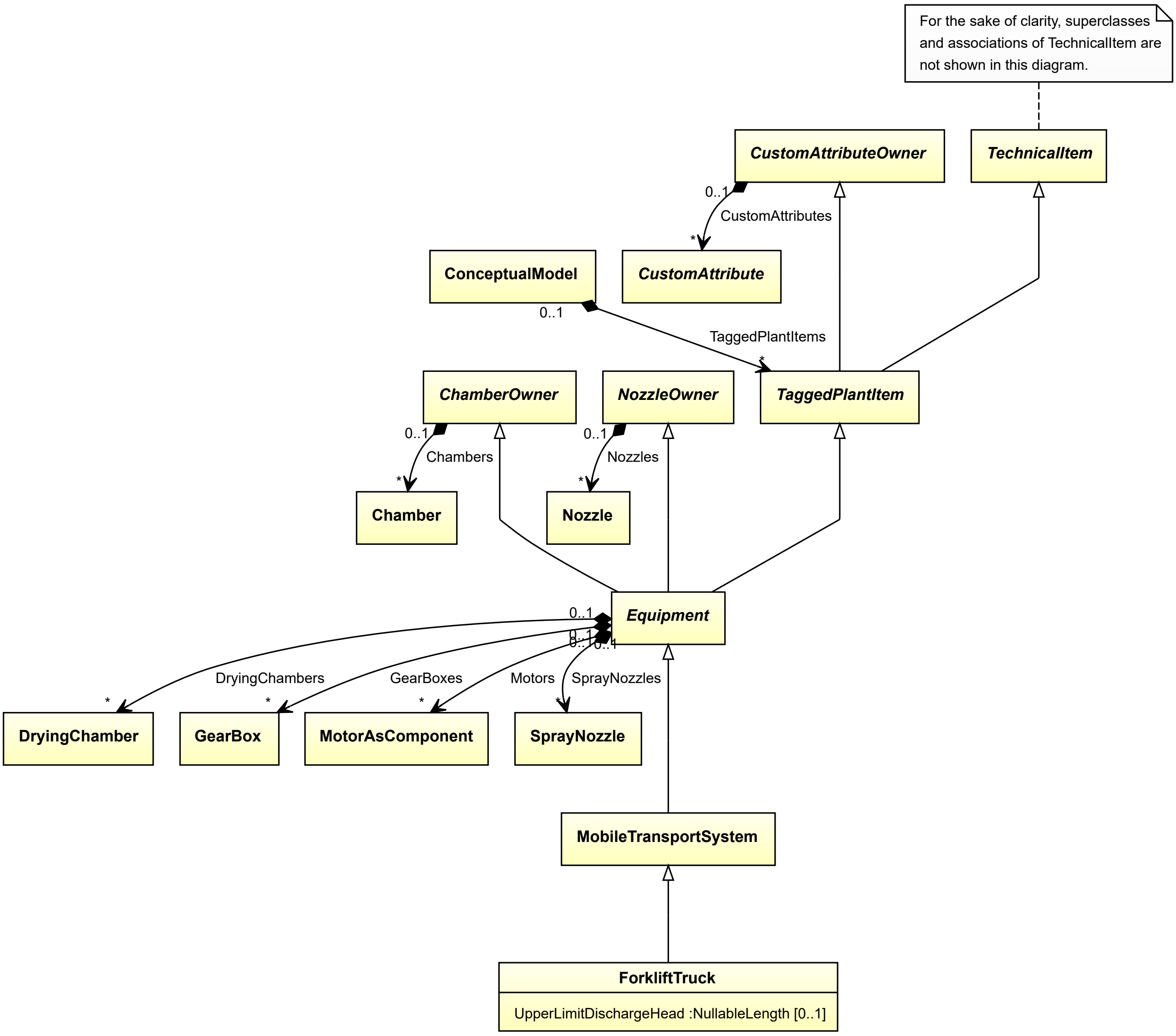
```
<Equipment
    ID="flare1"
    ComponentClass="Flare"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Flare" ...>
  ...
</Equipment>
```

7.83. ForkliftTruck

7.83.1 Overview

Class

A *MobileTransportSystem* and vehicle with power operated prongs that can be raised and lowered by will, for loading, transporting and unloading goods (from http://data.15926.org/rdl/RDS11590075).



Supertypes

- *MobileTransportSystem*

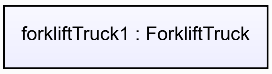
Attributes (data)

Name	Multiplicity	Type
UpperLimitDischargeHead	0..1	*NullableLength*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: FORKLIFT TRUCK
ComponentClass: ForkliftTruck
ComponentClassURI: http://data.posccaesar.org/rdl/RDS11590075

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="forkliftTruck1"
    ComponentClass="ForkliftTruck"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11590075" ...>
  ...
</Equipment>
```

7.83.2 UpperLimitDischargeHead

Attribute (data)

The upper limit for the discharge head of the *ForkliftTruck*.

Multiplicity: 0..1

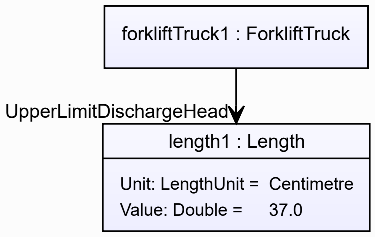
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT DISCHARGE HEAD
Name: UpperLimitDischargeHead
AttributeURI: http://sandbox.dexpi.org/rdl/UpperLimitDischargeHead

Example

The instance forkliftTruck1 represents a *ForkliftTruck* with an *UpperLimitDischargeHead* of 37.0 cm.



Example: Implementation in Proteus Schema

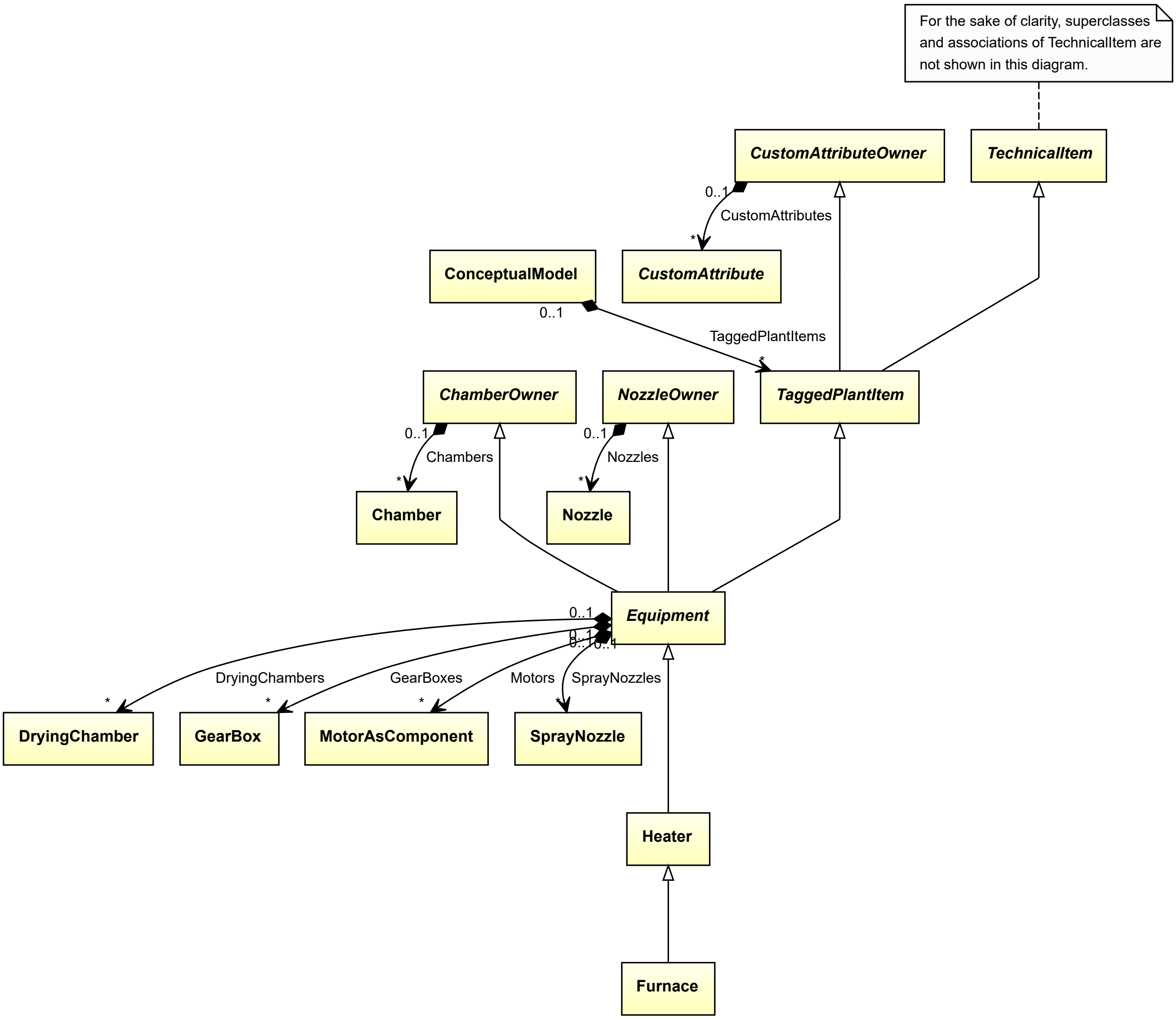
```
<Equipment
    ID="forkliftTruck1"
    ComponentClass="ForkliftTruck"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11590075" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="UpperLimitDischargeHead"
      AttributeURI="http://sandbox.dexpi.org/rdl/UpperLimitDischargeHead"
      Format="double"
      Value="37.0"
      Units="Centimetre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1318004" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.84. Furnace

7.84.1 Overview

Class

A physical object that is intended to induce a reaction in a process fluid by heating it (from http://data.posccaesar.org/rdl/RDS441134).



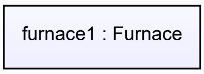
Supertypes

- *Heater*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: FURNACE
ComponentClass: Furnace
ComponentClassURI: http://data.posccaesar.org/rdl/RDS441134

Example



Example: Implementation in Proteus Schema

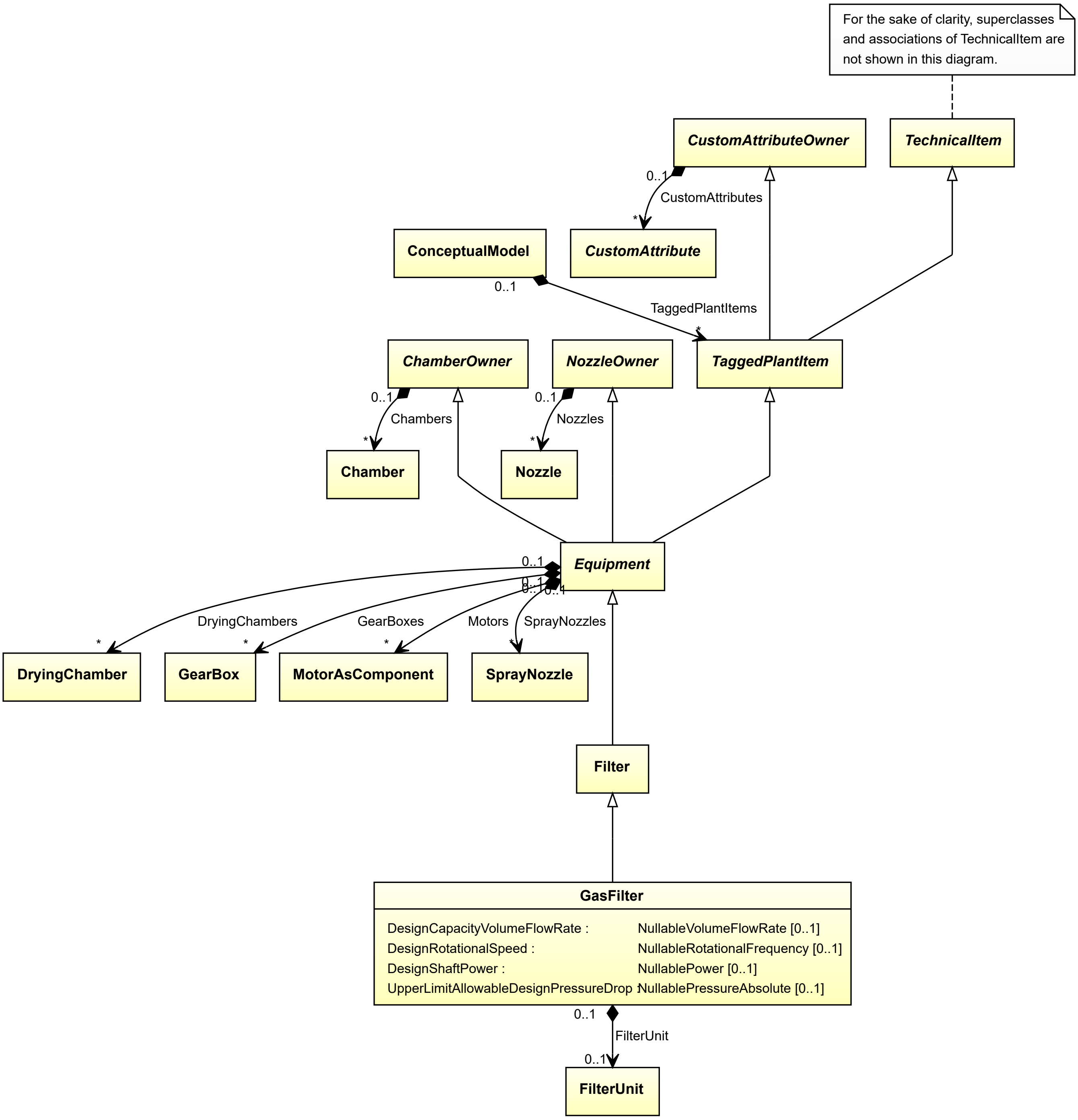
```
<Equipment
    ID="furnace1"
    ComponentClass="Furnace"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS441134" ...>
  ...
</Equipment>
```

7.85. GasFilter

7.85.1 Overview

Class

A filter that is specifically designed to filter a gas.



Supertypes

- *Filter*

Attributes (data)

Name	Multiplicity	Type
DesignCapacityVolumeFlowRate	0..1	*NullableVolumeFlowRate*
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*
DesignShaftPower	0..1	*NullablePower*
UpperLimitAllowableDesignPressureDrop	0..1	*NullablePressureAbsolute*

Attributes (composition)

Name	Multiplicity	Type
FilterUnit	0..1	*FilterUnit*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: GAS FILTER
ComponentClass: GasFilter
ComponentClassURI: http://data.posccaesar.org/rdl/RDS4316755843

Example

gasFilter1 : GasFilter

Example: Implementation in Proteus Schema

```
<Equipment
    ID="gasFilter1"
    ComponentClass="GasFilter"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS4316755843" ...>
  ...
</Equipment>
```

7.85.2 DesignCapacityVolumeFlowRate

Attribute (data)

The volume flow rate for which the *GasFilter* is designed.

Multiplicity: 0..1

Type: *NullableVolumeFlowRate*

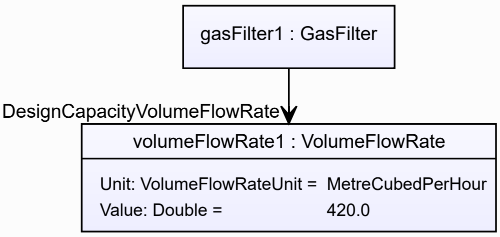
Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN CAPACITY VOLUME FLOW RATE

Name: DesignCapacityVolumeFlowRate
AttributeURI: http://sandbox.dexpi.org/rdl/DesignCapacityVolumeFlowRate

Example

The instance gasFilter1 represents a *GasFilter* with a *DesignCapacityVolumeFlowRate* of 420.0 m^3/h.


gasFilter1 : GasFilter
DesignCapacityVolumeFlowRate
volumeFlowRate1 : VolumeFlowRate
Unit: VolumeFlowRateUnit = MetreCubedPerHour
Value: Double = 420.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="gasFilter1"
    ComponentClass="GasFilter"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS4316755843" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignCapacityVolumeFlowRate"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignCapacityVolumeFlowRate"
      Format="double"
      Value="420.0"
      Units="MetreCubedPerHour"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1321064" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.85.3 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *GasFilter* is designed.

Multiplicity: 0..1

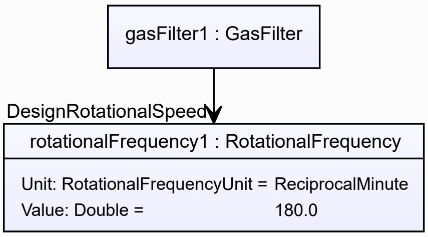
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

Example

The instance gasFilter1 represents a *GasFilter* with a *DesignRotationalSpeed* of 180.0 min^{-1}.


gasFilter1 : GasFilter
DesignRotationalSpeed
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 180.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="gasFilter1"
    ComponentClass="GasFilter"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS4316755843" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignRotationalSpeed"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignRotationalSpeed"
      Format="double"
      Value="180.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.85.4 DesignShaftPower

Attribute (data)

The shaft power for which the *GasFilter* is designed.

Multiplicity: 0..1

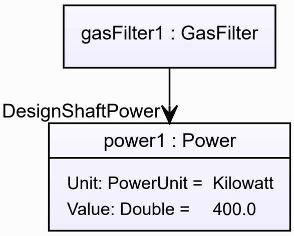
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SHAFT POWER
Name: DesignShaftPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignShaftPower

Example

The instance gasFilter1 represents a *GasFilter* with a *DesignShaftPower* of 400.0 kW.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="gasFilter1"
    ComponentClass="GasFilter"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS4316755843" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignShaftPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignShaftPower"
      Format="double"
      Value="400.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.85.5 FilterUnit

Attribute (composition)

The filter unit of the *GasFilter*.

Multiplicity: 0..1

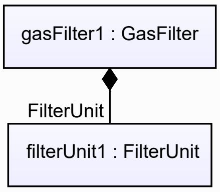
Type: *FilterUnit*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *FilterUnit*) is a child of the `<Equipment>` element for the attribute owner (a *GasFilter*).

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="gasFilter1"
    ComponentClass="GasFilter"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS4316755843" ...>
  ...
  <Equipment
      ID="filterUnit1"
      ComponentClass="FilterUnit"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/FilterUnit" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.85.6 UpperLimitAllowableDesignPressureDrop

Attribute (data)

The upper limit for the pressure drop for which the *GasFilter* is designed.

Multiplicity: 0..1

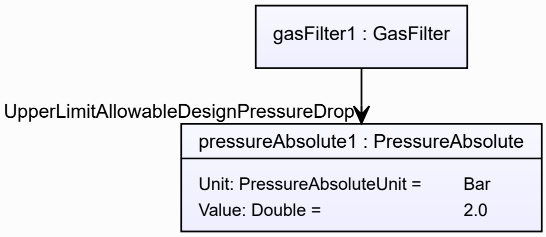
Type: *NullablePressureAbsolute*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT ALLOWABLE DESIGN PRESSURE DROP
Name: UpperLimitAllowableDesignPressureDrop
AttributeURI: http://sandbox.dexpi.org/rdl/UpperLimitAllowableDesignPressureDrop

Example

The instance gasFilter1 represents a *GasFilter* with an *UpperLimitAllowableDesignPressureDrop* of 2.0 bar.


gasFilter1 : GasFilter
UpperLimitAllowableDesignPressureDrop
pressureAbsolute1 : PressureAbsolute
Unit: PressureAbsoluteUnit = Bar
Value: Double = 2.0

Example: Implementation in Proteus Schema

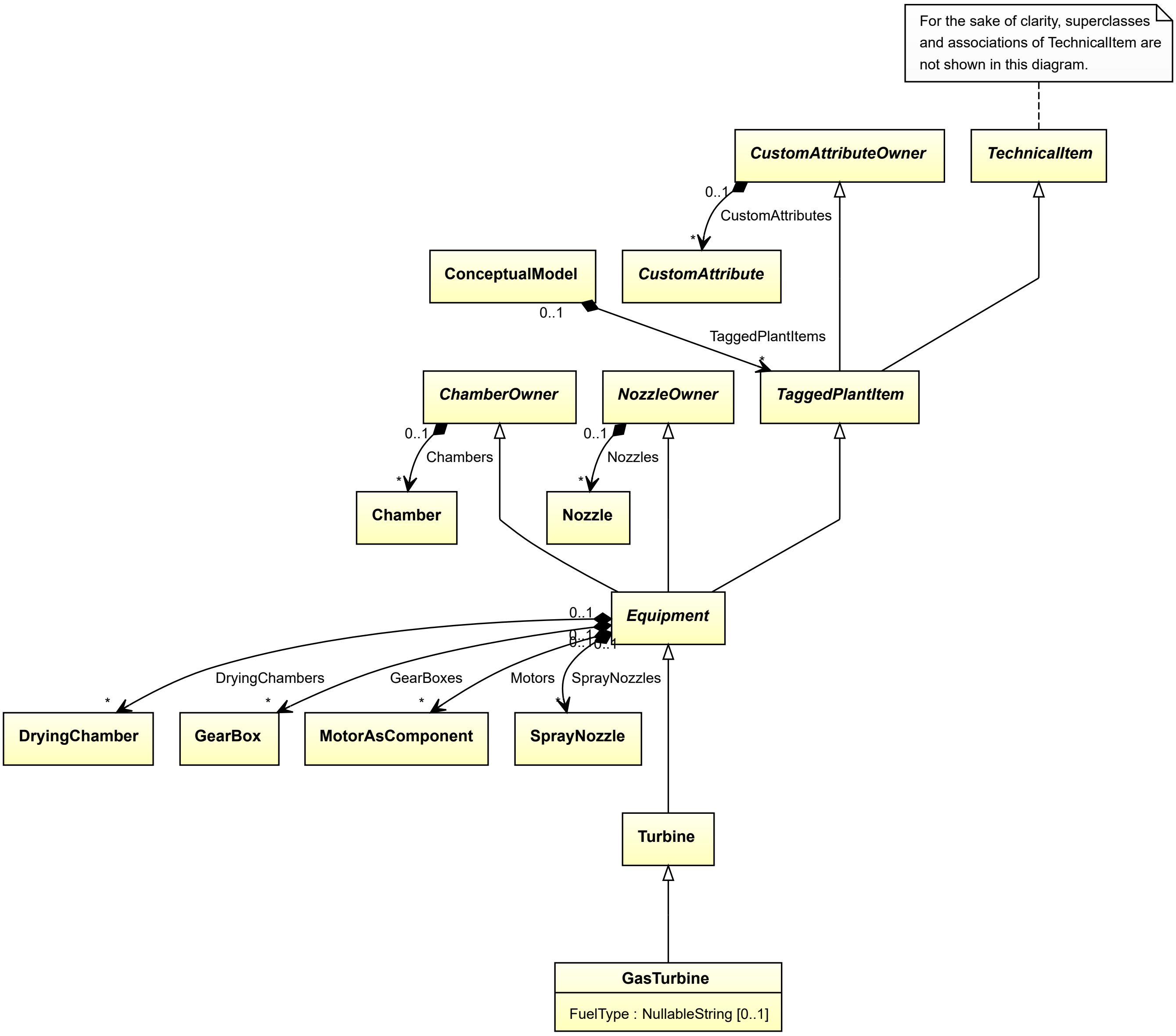
```
<Equipment
    ID="gasFilter1"
    ComponentClass="GasFilter"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS4316755843" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="UpperLimitAllowableDesignPressureDrop"
      AttributeURI="http://sandbox.dexpi.org/rdl/UpperLimitAllowableDesignPressureDrop"
      Format="double"
      Value="2.0"
      Units="Bar"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1314539" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.86. GasTurbine

7.86.1 Overview

Class

A machine that is a rotary mechanical device extracting energy from a gas flow and converting it into useful work.



Supertypes

- *Turbine*

Attributes (data)

Name	Multiplicity	Type
FuelType	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: GAS TURBINE
ComponentClass: GasTurbine
ComponentClassURI: http://sandbox.dexpi.org/rdl/GasTurbine

Example

Example: Implementation in Proteus Schema

```
<Equipment
    ID="gasTurbine1"
    ComponentClass="GasTurbine"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/GasTurbine" ...>
  ...
</Equipment>
```

7.86.2 FuelType

Attribute (data)

The fuel type of the *GasTurbine*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: FUEL TYPE
Name: FuelType
AttributeURI: http://sandbox.dexpi.org/rdl/FuelType

Example

“Diesel fuel” (*String*)

Example: Implementation in Proteus Schema

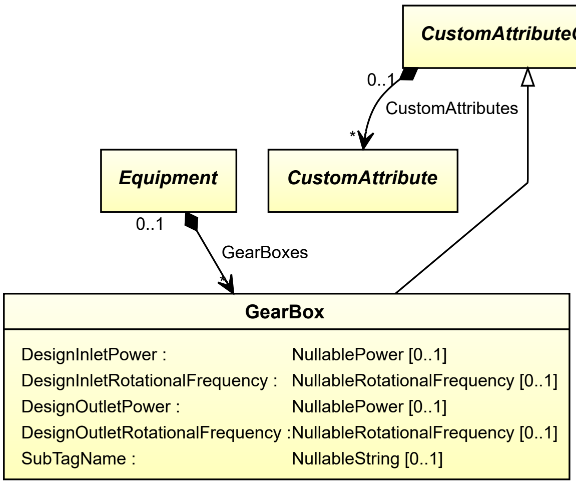
```
<Equipment
    ID="gasTurbine1"
    ComponentClass="GasTurbine"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/GasTurbine" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="FuelType"
      AttributeURI="http://sandbox.dexpi.org/rdl/FuelType"
      Format="string"
      Value="Diesel fuel" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.87. GearBox

7.87.1 Overview

Class

An artefact that consists of a gear casing with an arrangement of two or more gear-wheels transmitting rotating motion from the input shaft to the output shaft.


CustomAttributeOwner
0..1
CustomAttributes
*
Equipment
CustomAttribute
0..1
GearBoxes
*
GearBox
DesignInletPower : NullablePower [0..1]
DesignInletRotationalFrequency : NullableRotationalFrequency [0..1]
DesignOutletPower : NullablePower [0..1]
DesignOutletRotationalFrequency : NullableRotationalFrequency [0..1]
SubTagName : NullableString [0..1]

Supertypes

- *CustomAttributeOwner*

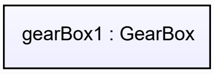
Attributes (data)

Name	Multiplicity	Type
DesignInletPower	0..1	*NullablePower*
DesignInletRotationalFrequency	0..1	*NullableRotationalFrequency*
DesignOutletPower	0..1	*NullablePower*
DesignOutletRotationalFrequency	0..1	*NullableRotationalFrequency*
SubTagName	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: GEARBOX
ComponentClass: Gearbox
ComponentClassURI: http://data.posccaesar.org/rdl/RDS889514

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="gearBox1"
    ComponentClass="Gearbox"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS889514" ...>
  ...
</Equipment>
```

7.87.2 DesignInletPower

Attribute (data)

The inlet power for which the *GearBox* is designed.

Multiplicity: 0..1

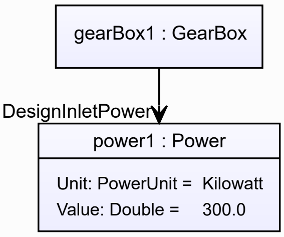
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN INLET POWER
Name: DesignInletPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignInletPower

Example

The instance gearBox1 represents a *GearBox* with a *DesignInletPower* of 300.0 kW.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="gearBox1"
    ComponentClass="Gearbox"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS889514" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignInletPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignInletPower"
      Format="double"
      Value="300.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.87.3 DesignInletRotationalFrequency

Attribute (data)

The inlet rotational frequency for which the *GearBox* is designed.

Multiplicity: 0..1

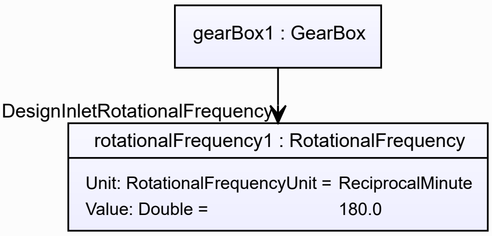
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN INLET ROTATIONAL FREQUENCY
Name: DesignInletRotationalFrequency
AttributeURI: http://sandbox.dexpi.org/rdl/DesignInletRotationalFrequency

Example

The instance gearBox1 represents a *GearBox* with a *DesignInletRotationalFrequency* of 180.0 min^{-1}.


gearBox1 : GearBox
DesignInletRotationalFrequency
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 180.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="gearBox1"
    ComponentClass="Gearbox"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS889514" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignInletRotationalFrequency"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignInletRotationalFrequency"
      Format="double"
      Value="180.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.87.4 DesignOutletPower

Attribute (data)

The outlet power for which the *GearBox* is designed.

Multiplicity: 0..1

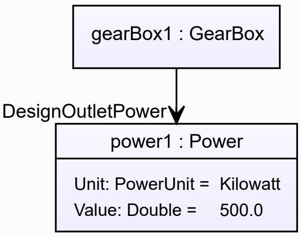
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN OUTLET POWER
Name: DesignOutletPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignOutletPower

Example

The instance gearBox1 represents a *GearBox* with a *DesignOutletPower* of 500.0 kW.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="gearBox1"
    ComponentClass="Gearbox"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS889514" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignOutletPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignOutletPower"
      Format="double"
      Value="500.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.87.5 DesignOutletRotationalFrequency

Attribute (data)

The outlet rotational frequency for which the *GearBox* is designed.

Multiplicity: 0..1

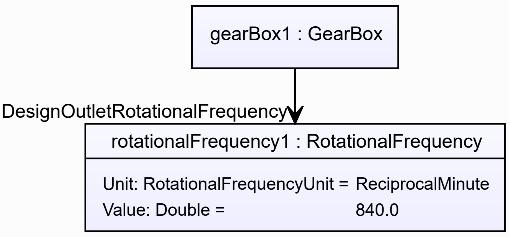
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN OUTLET ROTATIONAL FREQUENCY
Name: DesignOutletRotationalFrequency
AttributeURI: http://sandbox.dexpi.org/rdl/DesignOutletRotationalFrequency

Example

The instance gearBox1 represents a *GearBox* with a *DesignOutletRotationalFrequency* of 840.0 min^{-1}.


gearBox1 : GearBox
DesignOutletRotationalFrequency
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 840.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="gearBox1"
    ComponentClass="Gearbox"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS889514" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignOutletRotationalFrequency"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignOutletRotationalFrequency"
      Format="double"
      Value="840.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.87.6 SubTagName

Attribute (data)

The sub tag name of the *GearBox*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: SUB TAG NAME ASSIGNMENT CLASS
Name: SubTagNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/SubTagNameAssignmentClass

Example

“ST1” (*String*)

Example: Implementation in Proteus Schema

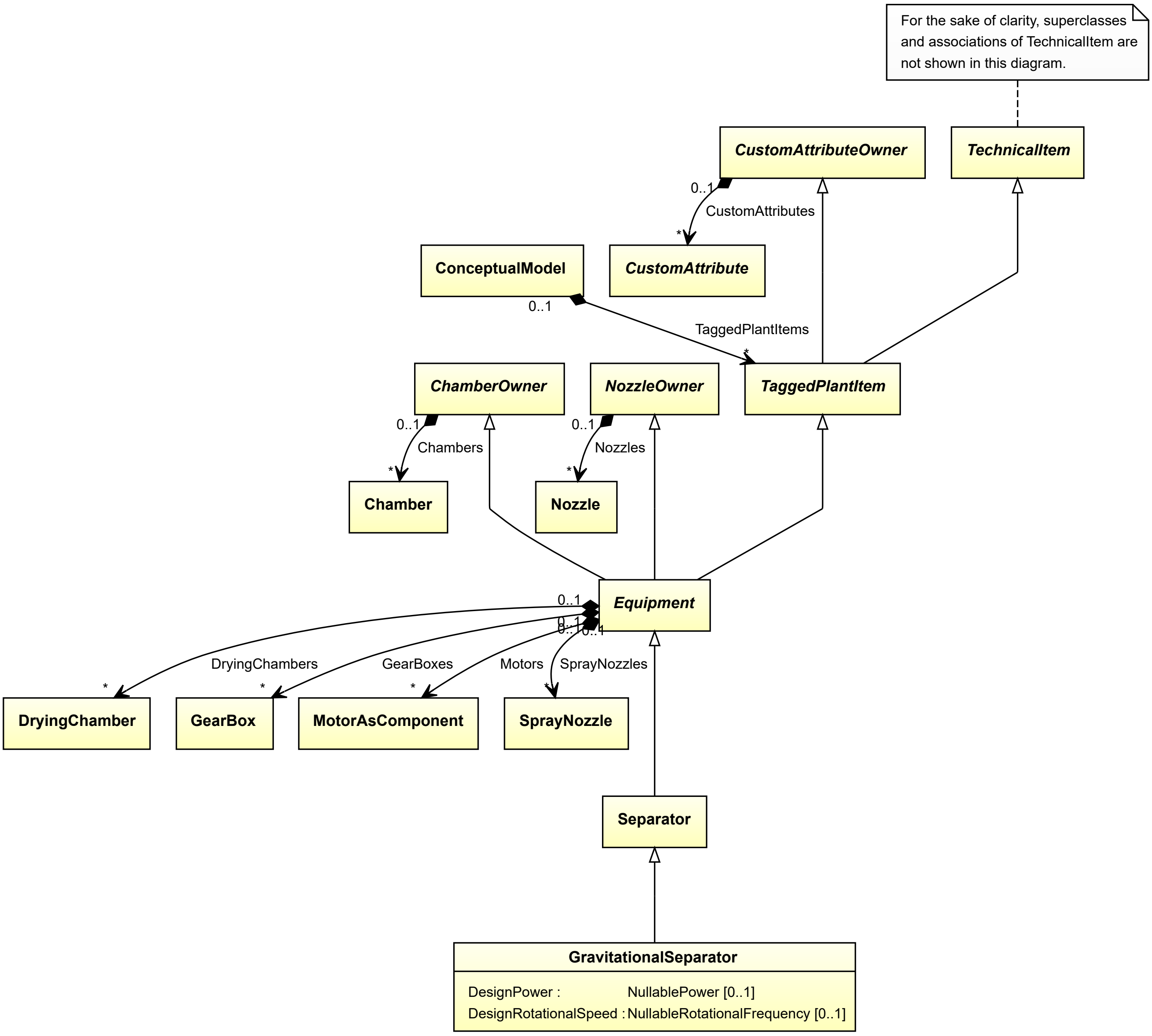
```
<Equipment
    ID="gearBox1"
    ComponentClass="Gearbox"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS889514" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="SubTagNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/SubTagNameAssignmentClass"
      Format="string"
      Value="ST1" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.88. GravitationalSeparator

7.88.1 Overview

Class

A fluid separator that is based on the difference in specific gravity for the substances to be separated (from http://data.15926.org/rdl/RDS16042131).



Supertypes

- *Separator*

Attributes (data)

Name	Multiplicity	Type
DesignPower	0..1	*NullablePower*
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: GRAVITY SEPARATOR
ComponentClass: GravitySeparator

ComponentClassURI: http://data.posccaesar.org/rdl/RDS16042131

Example

gravitationalSeparator1 : GravitationalSeparator

Example: Implementation in Proteus Schema

```
<Equipment
    ID="gravitationalSeparator1"
    ComponentClass="GravitySeparator"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS16042131" ...>
  ...
</Equipment>
```

7.88.2 DesignPower

Attribute (data)

The power for which the *GravitationalSeparator* is designed.

Multiplicity: 0..1

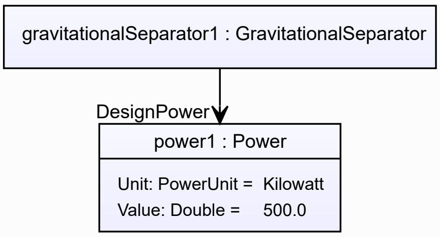
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN POWER
Name: DesignPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignPower

Example

The instance gravitationalSeparator1 represents a *GravitationalSeparator* with a *DesignPower* of 500.0 kW.


gravitationalSeparator1 : GravitationalSeparator
DesignPower
power1 : Power
Unit: PowerUnit = Kilowatt
Value: Double = 500.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="gravitationalSeparator1"
    ComponentClass="GravitySeparator"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS16042131" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignPower"
      Format="double"
      Value="500.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.88.3 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *GravitationalSeparator* is designed.

Multiplicity: 0..1

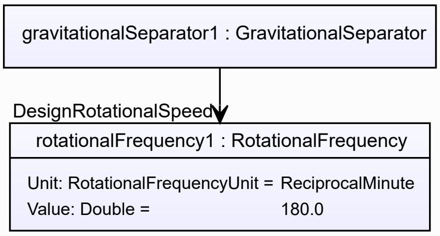
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

Example

The instance gravitationalSeparator1 represents a *GravitationalSeparator* with a *DesignRotationalSpeed* of 180.0 min^{-1}.


gravitationalSeparator1 : GravitationalSeparator
DesignRotationalSpeed
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 180.0

Example: Implementation in Proteus Schema

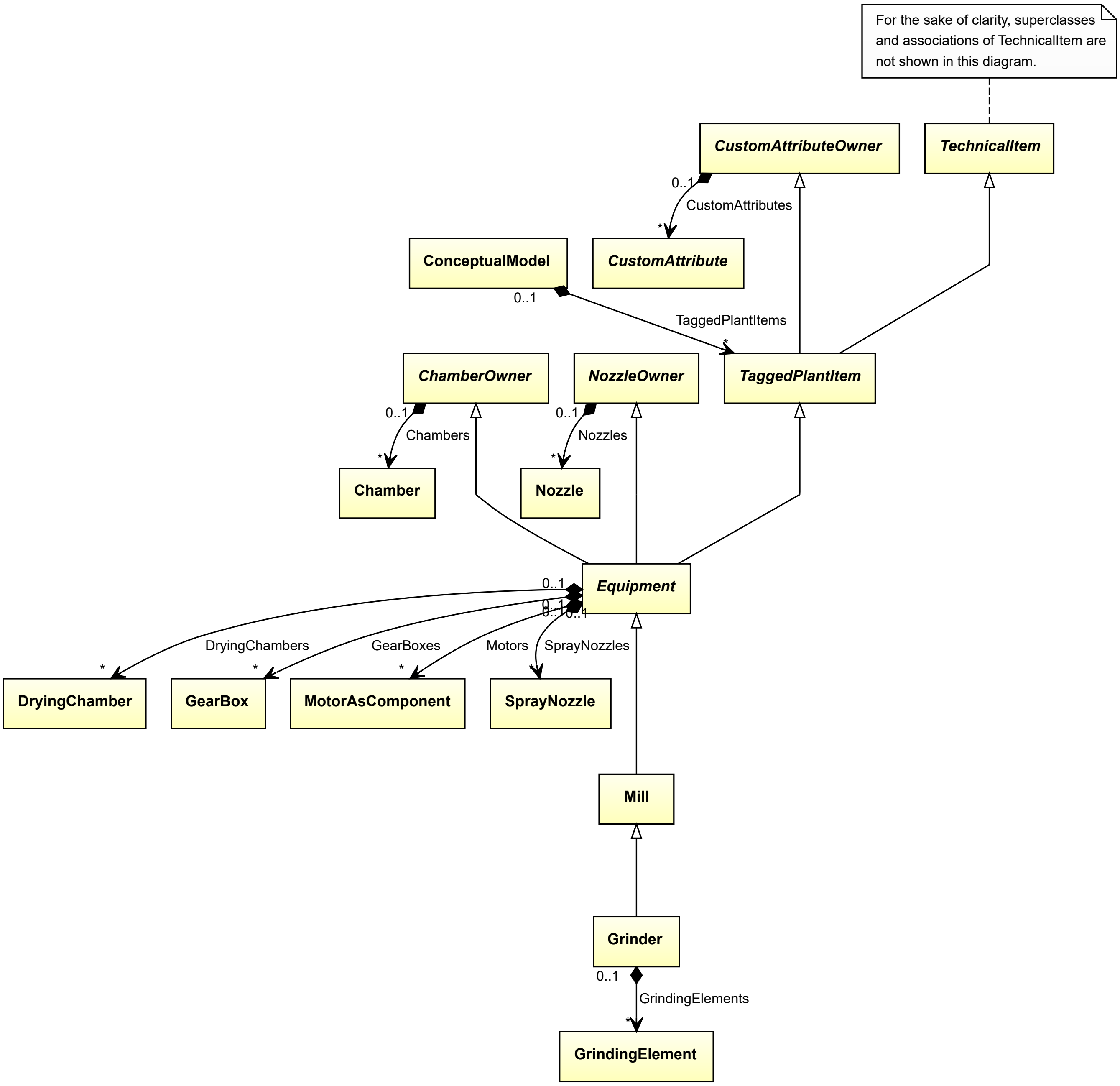
```
<Equipment
    ID="gravitationalSeparator1"
    ComponentClass="GravitySeparator"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS16042131" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignRotationalSpeed"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignRotationalSpeed"
      Format="double"
      Value="180.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.89. Grinder

7.89.1 Overview

Class

A *Mill* that has the capability of grinding,



Supertypes

- *Mill*

Attributes (composition)

Name	Multiplicity	Type
GrindingElements	*	*GrindingElement*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`

RDL reference: GRINDER
ComponentClass: Grinder
ComponentClassURI: http://sandbox.dexpi.org/rdl/Grinder

Example

grinder1 : Grinder

Example: Implementation in Proteus Schema

```
<Equipment
    ID="grinder1"
    ComponentClass="Grinder"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Grinder" ...>
  ...
</Equipment>
```

7.89.2 GrindingElements

Attribute (composition)

The grinding elements of the *Grinder*.

Multiplicity: *

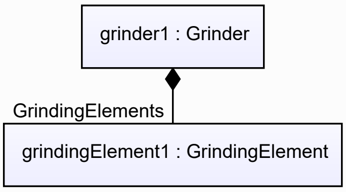
Type: *GrindingElement*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *GrindingElement*) is a child of the `<Equipment>` element for the attribute owner (a *Grinder*).

Example



Example: Implementation in Proteus Schema

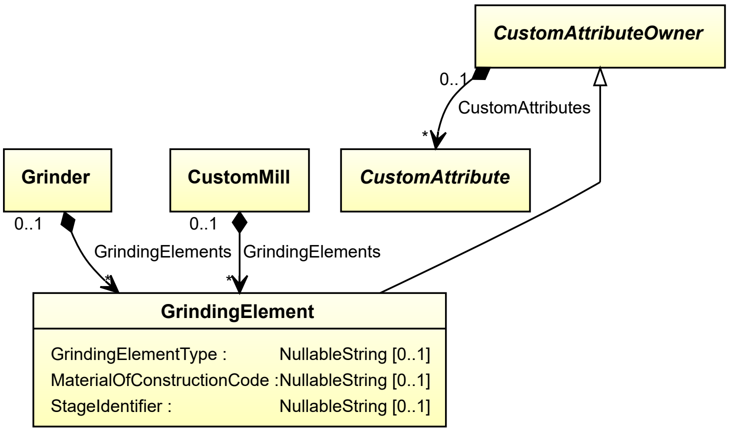
```
<Equipment
    ID="grinder1"
    ComponentClass="Grinder"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Grinder" ...>
  ...
  <Equipment
      ID="grindingElement1"
      ComponentClass="GrindingElement"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/GrindingElement" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.90. GrindingElement

7.90.1 Overview

Class

A functional component of a *Grinder*.


CustomAttributeOwner
0..1
CustomAttributes
*
Grinder
CustomMill
CustomAttribute
0..1
0..1
GrindingElements
GrindingElements
*
*
GrindingElement
GrindingElementType : NullableString [0..1]
MaterialOfConstructionCode : NullableString [0..1]
StageIdentifier : NullableString [0..1]

Supertypes

- *CustomAttributeOwner*

Attributes (data)

Name	Multiplicity	Type
GrindingElementType	0..1	*NullableString*
MaterialOfConstructionCode	0..1	*NullableString*
StageIdentifier	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: GRINDING ELEMENT
ComponentClass: GrindingElement
ComponentClassURI: http://sandbox.dexpi.org/rdl/GrindingElement

Example

grindingElement1 : GrindingElement

Example: Implementation in Proteus Schema

```
<Equipment
    ID="grindingElement1"
    ComponentClass="GrindingElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/GrindingElement" ...>
  ...
</Equipment>
```

7.90.2 GrindingElementType

Attribute (data)

A code that gives the crusher unit type of the *GrindingElement*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: GRINDING ELEMENT TYPE ASSIGNMENT CLASS
Name: GrindingElementTypeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/GrindingElementTypeAssignmentClass

Example

“1.4306” (*String*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="grindingElement1"
    ComponentClass="GrindingElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/GrindingElement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="GrindingElementTypeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/GrindingElementTypeAssignmentClass"
      Format="string"
      Value="1.4306" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.90.3 MaterialOfConstructionCode

Attribute (data)

A code that gives the material of construction of the *GrindingElement*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: MATERIAL OF CONSTRUCTION CODE ASSIGNMENT CLASS
Name: MaterialOfConstructionCodeAssignmentClass
AttributeURI: http://data.posccaesar.org/rdl/RDS1460719741

Example

“1.4306” (*String*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="grindingElement1"
    ComponentClass="GrindingElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/GrindingElement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="MaterialOfConstructionCodeAssignmentClass"
      AttributeURI="http://data.posccaesar.org/rdl/RDS1460719741"
      Format="string"
      Value="1.4306" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.90.4 StageIdentifier

Attribute (data)

The stage identifier of the *GrindingElement*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: STAGE IDENTIFIER ASSIGNMENT CLASS
Name: StageIdentifierAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/StageIdentifierAssignmentClass

Example

“s1” (*String*)

Example: Implementation in Proteus Schema

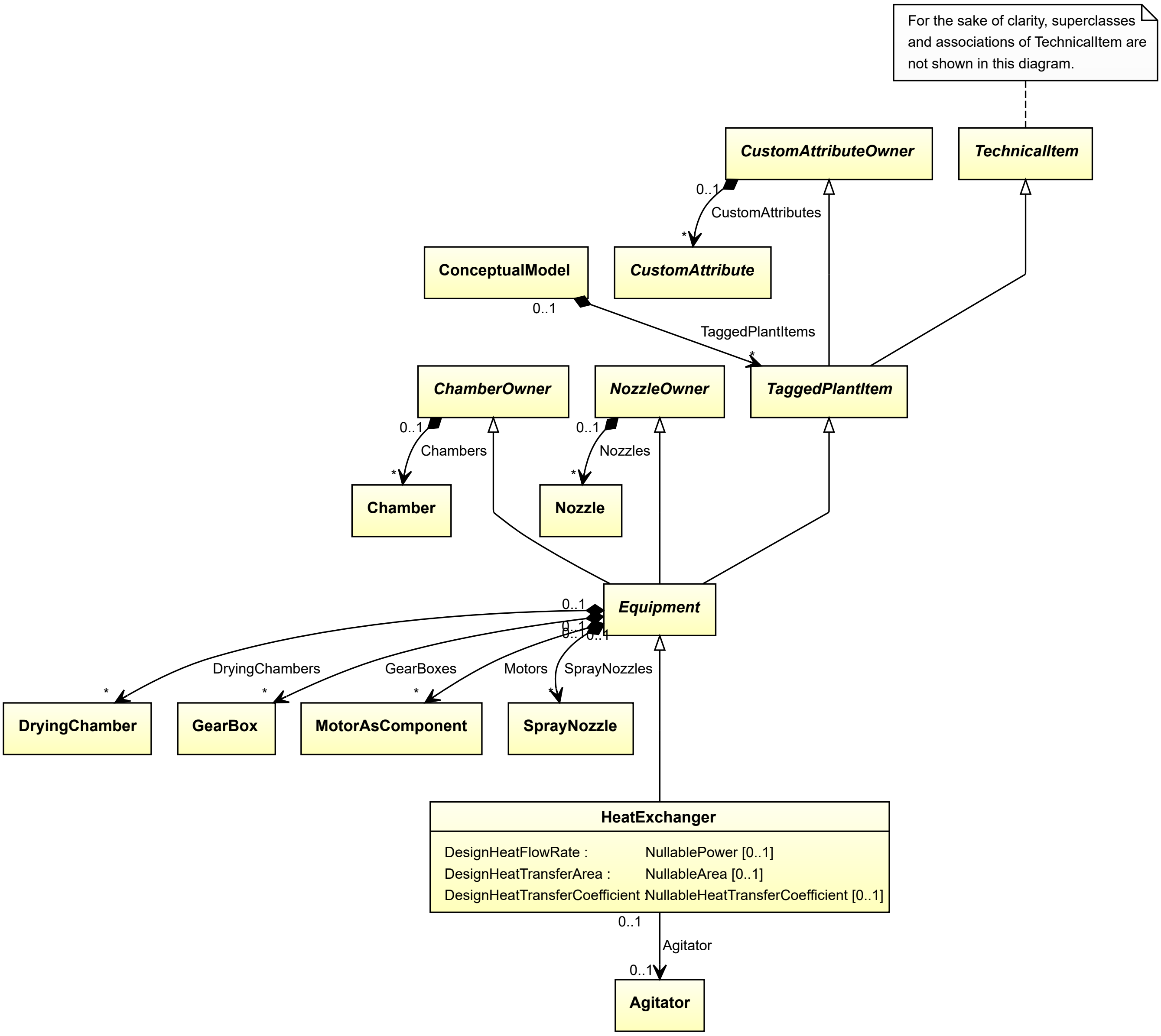
```
<Equipment
    ID="grindingElement1"
    ComponentClass="GrindingElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/GrindingElement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="StageIdentifierAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/StageIdentifierAssignmentClass"
      Format="string"
      Value="s1" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.91. HeatExchanger

7.91.1 Overview

Class

An apparatus or machine that has the capability of heat exchanging (from http://data.15926.org/rdl/RDS304199).



Supertypes

- *Equipment*

Subtypes

- *AirCoolingSystem*
- *CustomHeatExchanger*
- *ElectricHeater*
- *PlateHeatExchanger*
- *SpiralHeatExchanger*
- *ThinFilmEvaporator*
- *TubularHeatExchanger*

Attributes (data)

Name	Multiplicity	Type
DesignHeatFlowRate	0..1	*NullablePower*
DesignHeatTransferArea	0..1	*NullableArea*
DesignHeatTransferCoefficient	0..1	*NullableHeatTransferCoefficient*

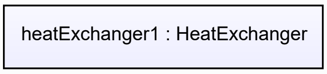
Attributes (reference)

Name	Multiplicity	Type
Agitator	0..1	*Agitator*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: HEAT EXCHANGER
ComponentClass: HeatExchanger
ComponentClassURI: http://data.posccaesar.org/rdl/RDS304199

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="heatExchanger1"
    ComponentClass="HeatExchanger"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS304199" ...>
  ...
</Equipment>
```

7.91.2 Agitator

Attribute (reference)

The *Agitator* of the *HeatExchanger*, if applicable.

Multiplicity: 0..1

Type: *Agitator*

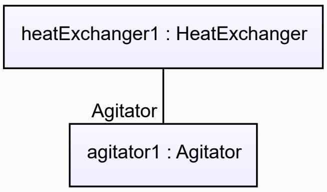
Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: `"is the location of"`

Opposite association type: `"is located in"`

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="heatExchanger1"
    ComponentClass="HeatExchanger"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS304199" ...>
  ...
  <Association
    Type="is the location of"
    ItemID="agitator1" />
  ...
<Equipment />
...
<Equipment
    ID="agitator1"
    ComponentClass="Agitator"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS16045622" ...>
  ...
  <Association
    Type="is located in"
    ItemID="heatExchanger1" />
  ...
<Equipment />
```

7.91.3 DesignHeatFlowRate

Attribute (data)

The heat flow rate for which the *HeatExchanger* is designed.

Multiplicity: 0..1

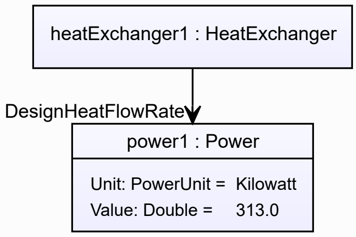
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN HEAT FLOW RATE
Name: DesignHeatFlowRate
AttributeURI: http://sandbox.dexpi.org/rdl/DesignHeatFlowRate

Example

The instance heatExchanger1 represents a *HeatExchanger* with a *DesignHeatFlowRate* of 313.0 kW.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="heatExchanger1"
    ComponentClass="HeatExchanger"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS304199" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignHeatFlowRate"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignHeatFlowRate"
      Format="double"
      Value="313.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.91.4 DesignHeatTransferArea

Attribute (data)

The heat transfer area for which the *HeatExchanger* is designed.

Multiplicity: 0..1

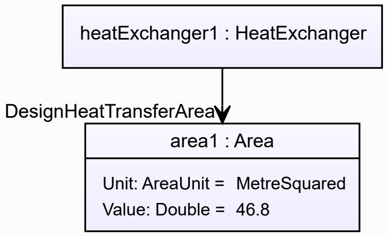
Type: *NullableArea*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN HEAT TRANSFER AREA
Name: DesignHeatTransferArea
AttributeURI: http://sandbox.dexpi.org/rdl/DesignHeatTransferArea

Example

The instance heatExchanger1 represents a *HeatExchanger* with a *DesignHeatTransferArea* of 46.8 m^2.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="heatExchanger1"
    ComponentClass="HeatExchanger"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS304199" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignHeatTransferArea"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignHeatTransferArea"
      Format="double"
      Value="46.8"
      Units="MetreSquared"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1358009" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.91.5 DesignHeatTransferCoefficient

Attribute (data)

The heat transfer coefficient for which the *HeatExchanger* is designed.

Multiplicity: 0..1

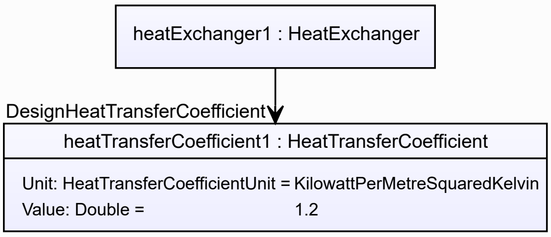
Type: *NullableHeatTransferCoefficient*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN HEAT TRANSFER COEFFICIENT
Name: DesignHeatTransferCoefficient
AttributeURI: http://sandbox.dexpi.org/rdl/DesignHeatTransferCoefficient

Example

The instance heatExchanger1 represents a *HeatExchanger* with a *DesignHeatTransferCoefficient* of 1.2 kW/(m$^2 \cdot$ K).


heatExchanger1 : HeatExchanger
DesignHeatTransferCoefficient
heatTransferCoefficient1 : HeatTransferCoefficient
Unit: HeatTransferCoefficientUnit = KilowattPerMetreSquaredKelvin
Value: Double = 1.2

Example: Implementation in Proteus Schema

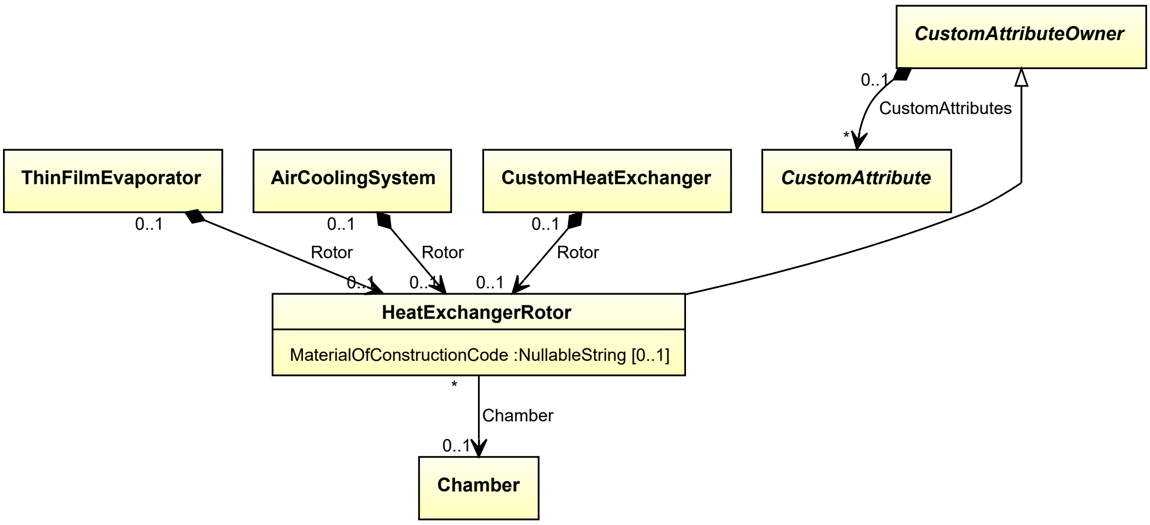
```
<Equipment
    ID="heatExchanger1"
    ComponentClass="HeatExchanger"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS304199" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignHeatTransferCoefficient"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignHeatTransferCoefficient"
      Format="double"
      Value="1.2"
      Units="KilowattPerMetreSquaredKelvin"
      UnitsURI="http://data.posccaesar.org/rdl/RDS43167567170" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.92. HeatExchangerRotor

7.92.1 Overview

Class

A rotor as a component of a *HeatExchanger*.



Supertypes

- *CustomAttributeOwner*

Attributes (data)

Name	Multiplicity	Type
MaterialOfConstructionCode	0..1	*NullableString*

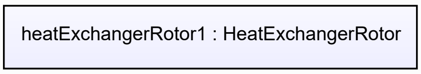
Attributes (reference)

Name	Multiplicity	Type
Chamber	0..1	*Chamber*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: HEAT EXCHANGER ROTOR
ComponentClass: HeatExchangerRotor
ComponentClassURI: http://sandbox.dexpi.org/rdl/HeatExchangerRotor

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="heatExchangerRotor1"
    ComponentClass="HeatExchangerRotor"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/HeatExchangerRotor" ...>
  ...
</Equipment>
```

7.92.2 Chamber

Attribute (reference)

The *Chamber* in which the *HeatExchangerRotor* is located, if applicable. The Chamber must be a component of the same object as the HeatExchangerRotor.

Multiplicity: 0..1

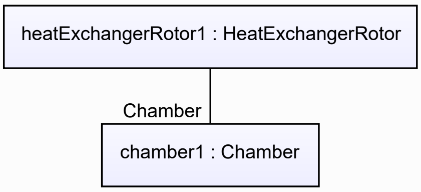
Type: *Chamber*

Opposite multiplicity: 0..*

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: `"is located in"`
Opposite association type: `"is the location of"`

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="heatExchangerRotor1"
    ComponentClass="HeatExchangerRotor"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/HeatExchangerRotor" ...>
  ...
  <Association
    Type="is located in"
    ItemID="chamber1" />
  ...
<Equipment />
...
<Equipment
    ID="chamber1"
    ComponentClass="Chamber"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS903151421" ...>
  ...
  <Association
    Type="is the location of"
    ItemID="heatExchangerRotor1" />
  ...
<Equipment />
```

7.92.3 MaterialOfConstructionCode

Attribute (data)

A code that gives the material of construction of the *HeatExchangerRotor*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: MATERIAL OF CONSTRUCTION CODE ASSIGNMENT CLASS
Name: MaterialOfConstructionCodeAssignmentClass
AttributeURI: http://data.posccaesar.org/rdl/RDS1460719741

Example

“1.4306” (*String*)

Example: Implementation in Proteus Schema

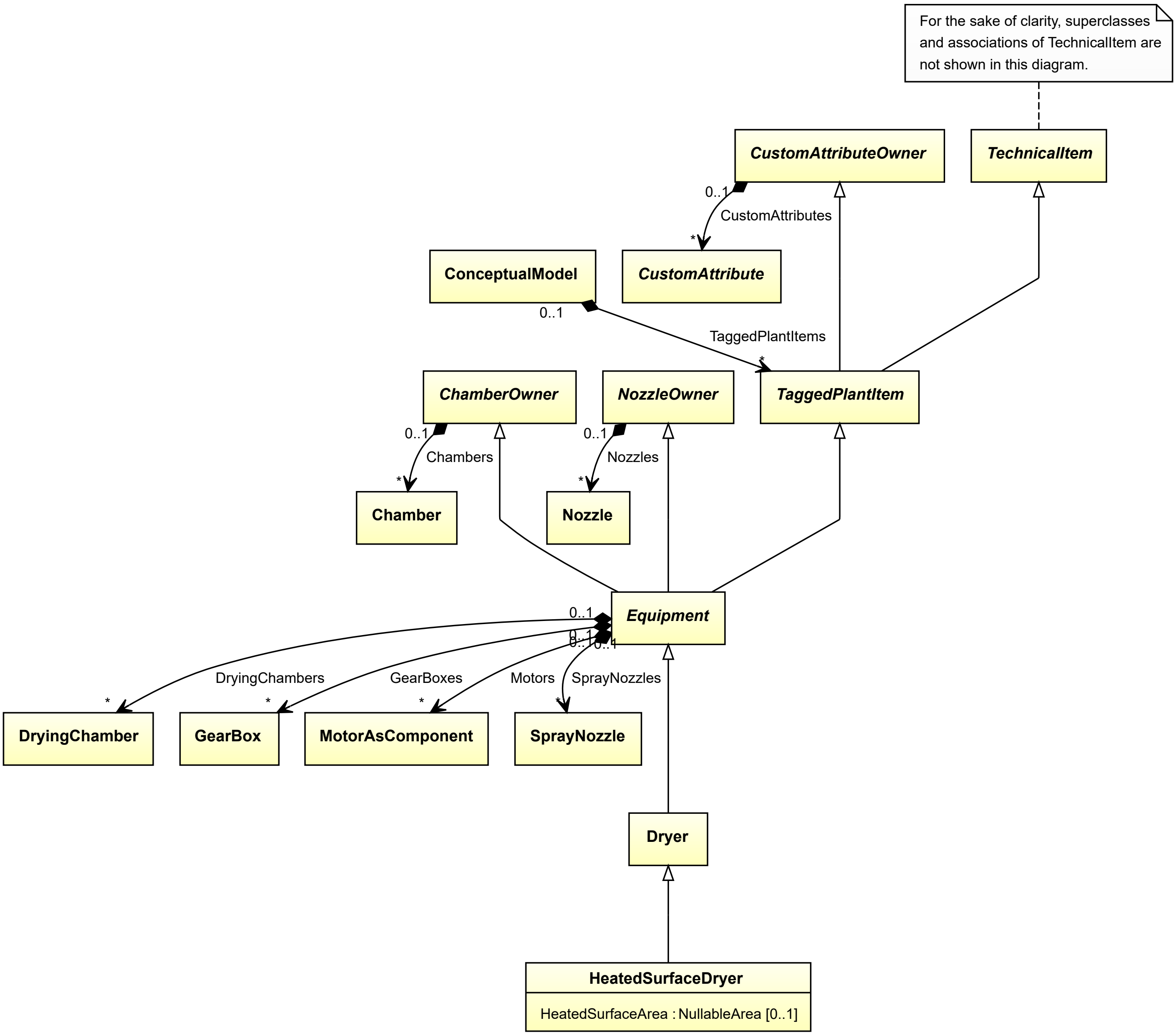
```
<Equipment
    ID="heatExchangerRotor1"
    ComponentClass="HeatExchangerRotor"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/HeatExchangerRotor" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="MaterialOfConstructionCodeAssignmentClass"
      AttributeURI="http://data.posccaesar.org/rdl/RDS1460719741"
      Format="string"
      Value="1.4306" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.93. HeatedSurfaceDryer

7.93.1 Overview

Class

A *Dryer* that dries a material by radiation and/or conduction caused by a heated surface (from http://data.15926.org/rdl/RDS2228449).



Supertypes

- *Dryer*

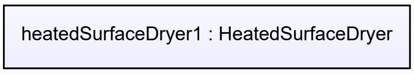
Attributes (data)

Name	Multiplicity	Type
HeatedSurfaceArea	0..1	*NullableArea*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: HEATED SURFACE DRYER
ComponentClass: HeatedSurfaceDryer
ComponentClassURI: http://sandbox.dexpi.org/rdl/HeatedSurfaceDryer

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="heatedSurfaceDryer1"
    ComponentClass="HeatedSurfaceDryer"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/HeatedSurfaceDryer" ...>
  ...
</Equipment>
```

7.93.2 HeatedSurfaceArea

Attribute (data)

The heated surface area of the *HeatedSurfaceDryer*.

Multiplicity: 0..1

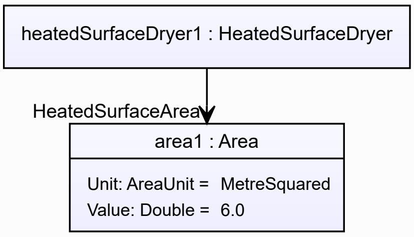
Type: *NullableArea*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: HEATED SURFACE AREA
Name: HeatedSurfaceArea
AttributeURI: http://sandbox.dexpi.org/rdl/HeatedSurfaceArea

Example

The instance heatedSurfaceDryer1 represents a *HeatedSurfaceDryer* with a *HeatedSurfaceArea* of 6.0 m^2.


heatedSurfaceDryer1 : HeatedSurfaceDryer
HeatedSurfaceArea
area1 : Area
Unit: AreaUnit = MetreSquared
Value: Double = 6.0

Example: Implementation in Proteus Schema

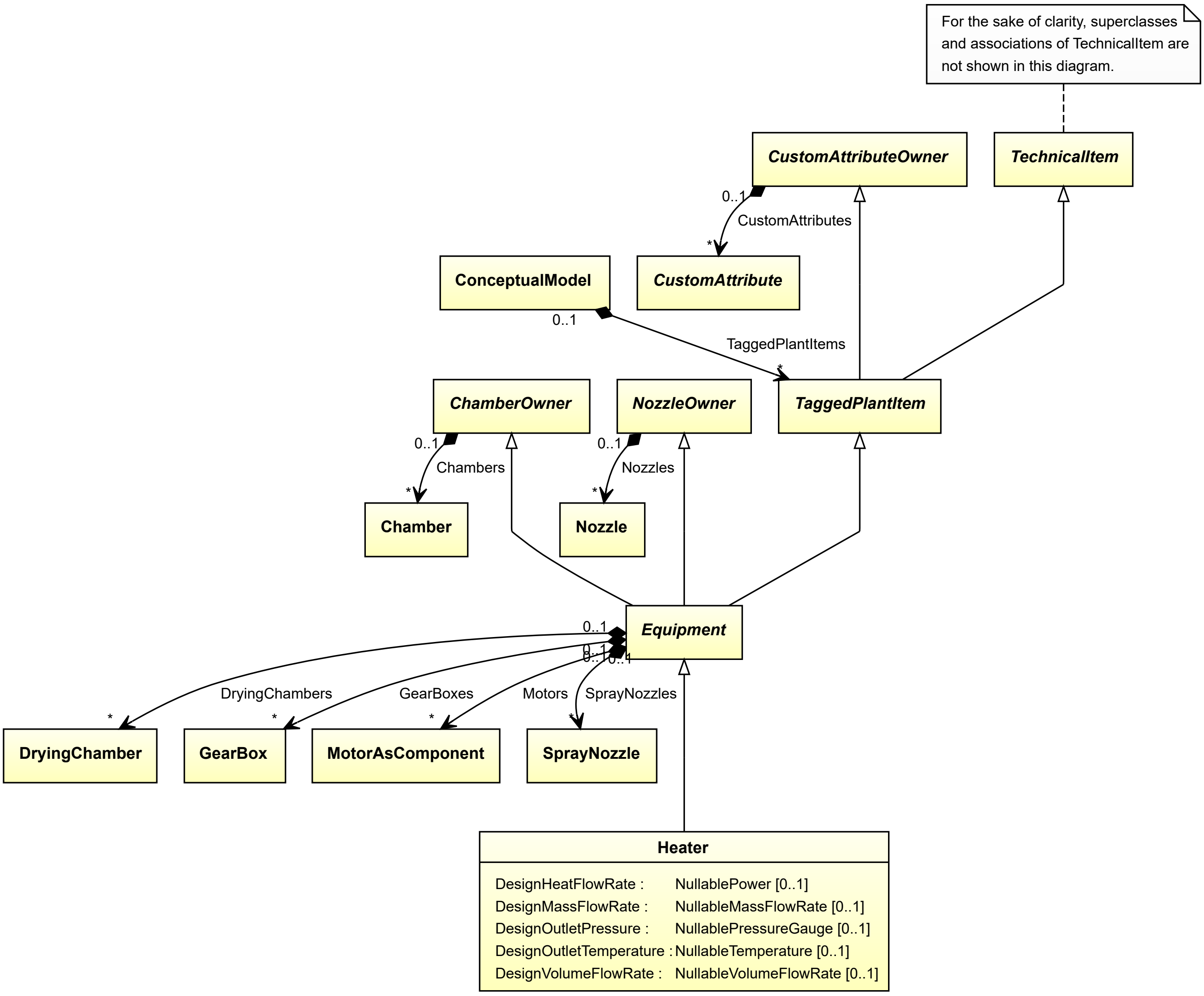
```
<Equipment
    ID="heatedSurfaceDryer1"
    ComponentClass="HeatedSurfaceDryer"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/HeatedSurfaceDryer" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="HeatedSurfaceArea"
      AttributeURI="http://sandbox.dexpi.org/rdl/HeatedSurfaceArea"
      Format="double"
      Value="6.0"
      Units="MetreSquared"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1358009" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.94. Heater

7.94.1 Overview

Class

An apparatus or machine that has the capability of heating.


For the sake of clarity, superclasses and associations of TechnicalItem are not shown in this diagram.
CustomAttributeOwner
TechnicalItem
0..1
CustomAttributes
*
ConceptualModel
CustomAttribute
0..1
TaggedPlantItems
*
ChamberOwner
NozzleOwner
TaggedPlantItem
0..1
Chambers
0..1
Nozzles
*
Chamber
*
Nozzle
Equipment
0..1
0..1
0..1
0..1
DryingChambers
GearBoxes
Motors
SprayNozzles
*
*
*
*
DryingChamber
GearBox
MotorAsComponent
SprayNozzle
Heater
DesignHeatFlowRate : NullablePower [0..1]
DesignMassFlowRate : NullableMassFlowRate [0..1]
DesignOutletPressure : NullablePressureGauge [0..1]
DesignOutletTemperature : NullableTemperature [0..1]
DesignVolumeFlowRate : NullableVolumeFlowRate [0..1]

Supertypes

- *Equipment*

Subtypes

- *Boiler*
- *CustomHeater*
- *Furnace*
- *SteamGenerator*

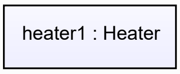
Attributes (data)

Name	Multiplicity	Type
DesignHeatFlowRate	0..1	*NullablePower*
DesignMassFlowRate	0..1	*NullableMassFlowRate*
DesignOutletPressure	0..1	*NullablePressureGauge*
DesignOutletTemperature	0..1	*NullableTemperature*
DesignVolumeFlowRate	0..1	*NullableVolumeFlowRate*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: HEATER
ComponentClass: Heater
ComponentClassURI: http://data.posccaesar.org/rdl/RDS13048646

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="heater1"
    ComponentClass="Heater"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS13048646" ...>
  ...
</Equipment>
```

7.94.2 DesignHeatFlowRate

Attribute (data)

The heat flow rate for which the *Heater* is designed.

Multiplicity: 0..1

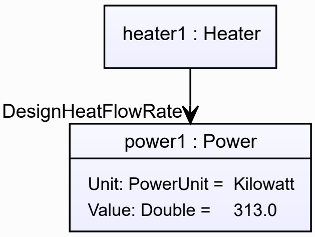
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN HEAT FLOW RATE
Name: DesignHeatFlowRate
AttributeURI: http://sandbox.dexpi.org/rdl/DesignHeatFlowRate

Example

The instance heater1 represents a *Heater* with a *DesignHeatFlowRate* of 313.0 kW.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="heater1"
    ComponentClass="Heater"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS13048646" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignHeatFlowRate"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignHeatFlowRate"
      Format="double"
      Value="313.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.94.3 DesignMassFlowRate

Attribute (data)

The mass flow rate for which the *Heater* is designed.

Multiplicity: 0..1

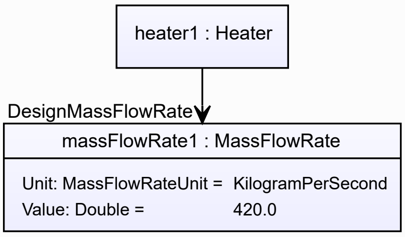
Type: *NullableMassFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN MASS FLOW RATE
Name: DesignMassFlowRate
AttributeURI: http://data.posccaesar.org/rdl/RDS14286182

Example

The instance heater1 represents a *Heater* with a *DesignMassFlowRate* of 420.0 kg/s.


heater1 : Heater
DesignMassFlowRate
massFlowRate1 : MassFlowRate
Unit: MassFlowRateUnit = KilogramPerSecond
Value: Double = 420.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="heater1"
    ComponentClass="Heater"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS13048646" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignMassFlowRate"
      AttributeURI="http://data.posccaesar.org/rdl/RDS14286182"
      Format="double"
      Value="420.0"
      Units="KilogramPerSecond"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1329659" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.94.4 DesignOutletPressure

Attribute (data)

The outlet pressure for which the *Heater* is designed.

Multiplicity: 0..1

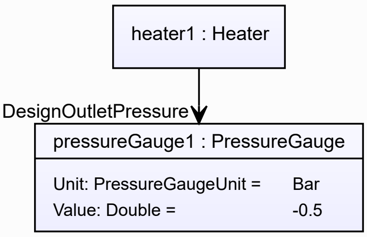
Type: *NullablePressureGauge*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: OUTLET DESIGN PRESSURE
Name: OutletDesignPressure
AttributeURI: http://data.posccaesar.org/rdl/RDS7471401

Example

The instance heater1 represents a *Heater* with a *DesignOutletPressure* of -0.5 bar.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="heater1"
    ComponentClass="Heater"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS13048646" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="OutletDesignPressure"
      AttributeURI="http://data.posccaesar.org/rdl/RDS7471401"
      Format="double"
      Value="-0.5"
      Units="Bar"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1314539" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.94.5 DesignOutletTemperature

Attribute (data)

The outlet temperature for which the *Heater* is designed.

Multiplicity: 0..1

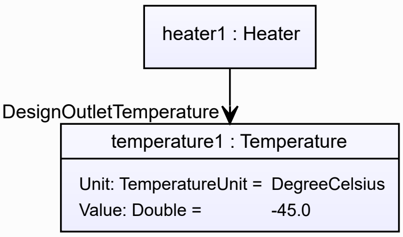
Type: *NullableTemperature*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: OUTLET DESIGN TEMPERATURE
Name: OutletDesignTemperature
AttributeURI: http://data.posccaesar.org/rdl/RDS7471243

Example

The instance heater1 represents a *Heater* with a *DesignOutletTemperature* of -45.0 °C.


heater1 : Heater
DesignOutletTemperature
temperature1 : Temperature
Unit: TemperatureUnit = DegreeCelsius
Value: Double = -45.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="heater1"
    ComponentClass="Heater"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS13048646" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="OutletDesignTemperature"
      AttributeURI="http://data.posccaesar.org/rdl/RDS7471243"
      Format="double"
      Value="-45.0"
      Units="DegreeCelsius"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1322684" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.94.6 DesignVolumeFlowRate

Attribute (data)

The volume flow rate for which the *Heater* is designed.

Multiplicity: 0..1

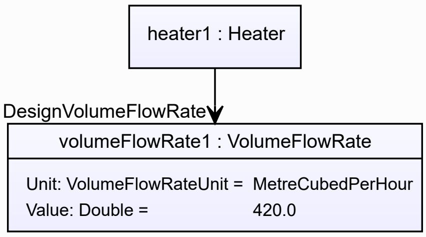
Type: *NullableVolumeFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN VOLUME FLOW RATE
Name: DesignVolumeFlowRate
AttributeURI: http://data.posccaesar.org/rdl/RDS14286227

Example

The instance heater1 represents a *Heater* with a *DesignVolumeFlowRate* of 420.0 m^3/h.


heater1 : Heater
DesignVolumeFlowRate
volumeFlowRate1 : VolumeFlowRate
Unit: VolumeFlowRateUnit = MetreCubedPerHour
Value: Double = 420.0

Example: Implementation in Proteus Schema

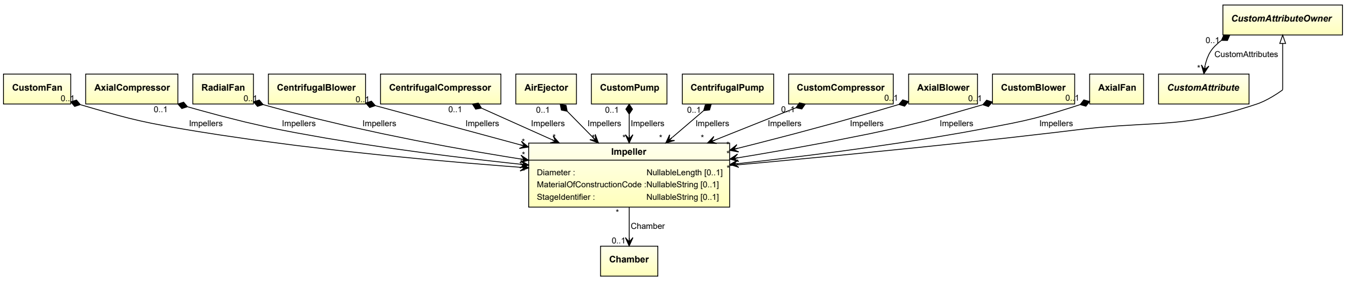
```
<Equipment
    ID="heater1"
    ComponentClass="Heater"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS13048646" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignVolumeFlowRate"
      AttributeURI="http://data.posccaesar.org/rdl/RDS14286227"
      Format="double"
      Value="420.0"
      Units="MetreCubedPerHour"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1321064" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.95. Impeller

7.95.1 Overview

Class

An energy converter component that is an assembly of rotating vanes within an enclosure which is used to impart energy to or derive energy from a fluid through dynamic force (from http://data.posccaesar.org/rdl/RDS414539).



Supertypes

- *CustomAttributeOwner*

Attributes (data)

Name	Multiplicity	Type
Diameter	0..1	*NullableLength*
MaterialOfConstructionCode	0..1	*NullableString*
StageIdentifier	0..1	*NullableString*

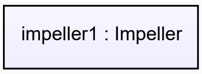
Attributes (reference)

Name	Multiplicity	Type
Chamber	0..1	*Chamber*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: IMPELLER
ComponentClass: Impeller
ComponentClassURI: http://data.posccaesar.org/rdl/RDS414539

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="impeller1"
    ComponentClass="Impeller"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS414539" ...>
  ...
</Equipment>
```

7.95.2 Chamber

Attribute (reference)

The *Chamber* in which the *Impeller* is located, if applicable. The Chamber must be a component of the same object as the Impeller.

Multiplicity: 0..1

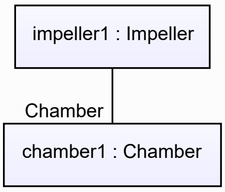
Type: *Chamber*

Opposite multiplicity: 0..*

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: `"is located in"`
Opposite association type: `"is the location of"`

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="impeller1"
    ComponentClass="Impeller"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS414539" ...>
  ...
  <Association
    Type="is located in"
    ItemID="chamber1" />
  ...
<Equipment />
...
<Equipment
    ID="chamber1"
    ComponentClass="Chamber"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS903151421" ...>
  ...
  <Association
    Type="is the location of"
    ItemID="impeller1" />
  ...
<Equipment />
```

7.95.3 Diameter

Attribute (data)

The diameter of the *Impeller*.

Multiplicity: 0..1

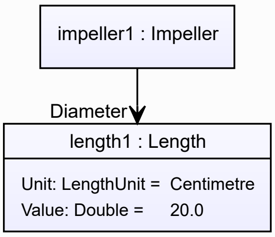
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DIAMETER
Name: Diameter
AttributeURI: http://data.posccaesar.org/rdl/RDS350954

Example

The instance impeller1 represents an *Impeller* with a *Diameter* of 20.0 cm.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="impeller1"
    ComponentClass="Impeller"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS414539" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="Diameter"
      AttributeURI="http://data.posccaesar.org/rdl/RDS350954"
      Format="double"
      Value="20.0"
      Units="Centimetre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1318004" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.95.4 MaterialOfConstructionCode

Attribute (data)

A code that gives the material of construction of the *Impeller*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: MATERIAL OF CONSTRUCTION CODE ASSIGNMENT CLASS
Name: MaterialOfConstructionCodeAssignmentClass
AttributeURI: http://data.posccaesar.org/rdl/RDS1460719741

Example

“1.4306” (*String*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="impeller1"
    ComponentClass="Impeller"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS414539" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="MaterialOfConstructionCodeAssignmentClass"
      AttributeURI="http://data.posccaesar.org/rdl/RDS1460719741"
      Format="string"
      Value="1.4306" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.95.5 StageIdentifier

Attribute (data)

The stage identifier of the *Impeller*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: STAGE IDENTIFIER ASSIGNMENT CLASS
Name: StageIdentifierAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/StageIdentifierAssignmentClass

Example

“s1” (*String*)

Example: Implementation in Proteus Schema

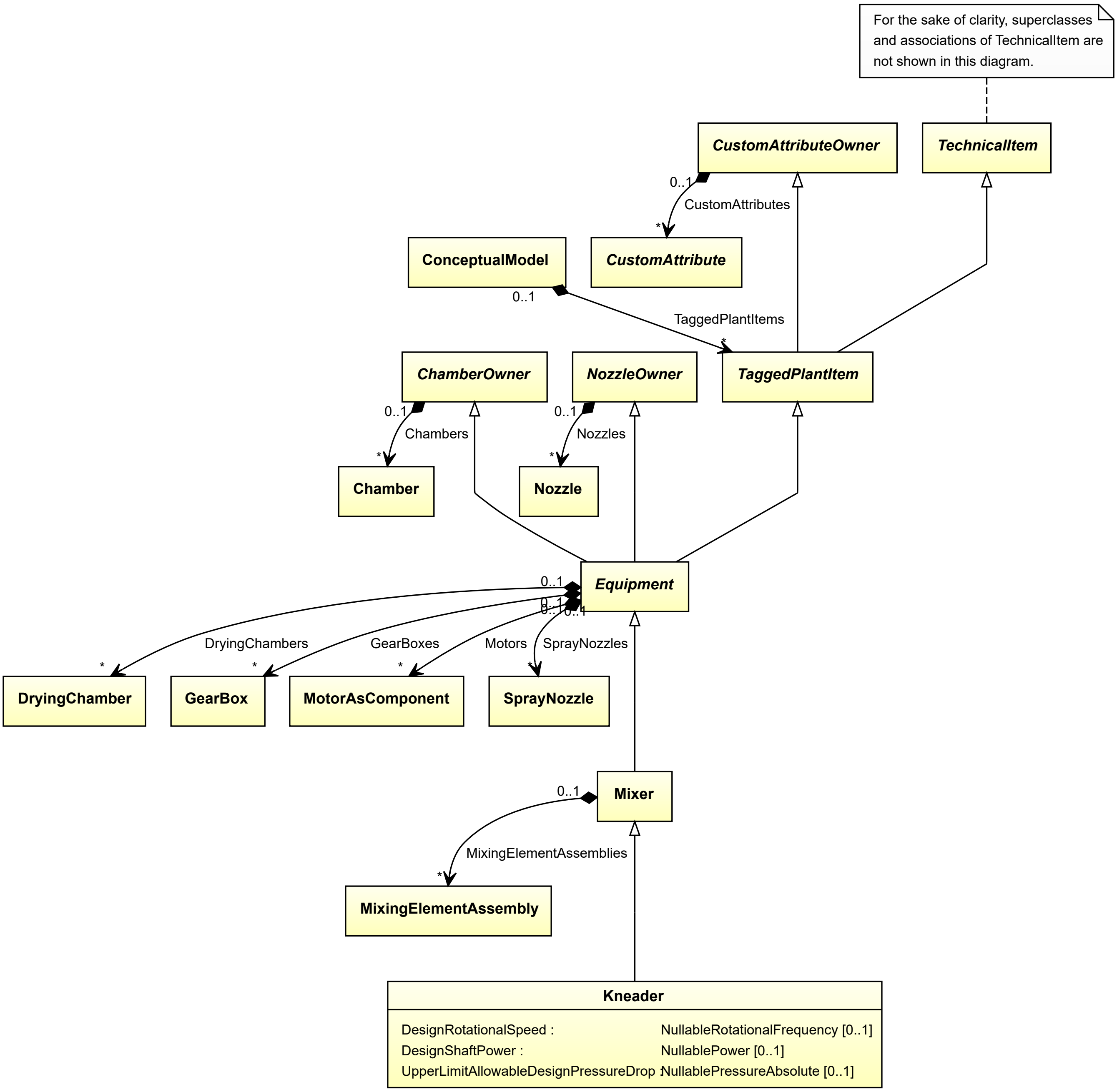
```
<Equipment
    ID="impeller1"
    ComponentClass="Impeller"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS414539" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="StageIdentifierAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/StageIdentifierAssignmentClass"
      Format="string"
      Value="s1" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.96. Kneader

7.96.1 Overview

Class

A machine that is capable of mixing and working into a uniform mass by, or as if by, folding, pressing, and stretching.



Supertypes

- *Mixer*

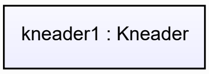
Attributes (data)

Name	Multiplicity	Type
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*
DesignShaftPower	0..1	*NullablePower*
UpperLimitAllowableDesignPressureDrop	0..1	*NullablePressureAbsolute*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: KNEADER
ComponentClass: Kneader
ComponentClassURI: http://sandbox.dexpi.org/rdl/Kneader

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="kneader1"
    ComponentClass="Kneader"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Kneader" ...>
  ...
</Equipment>
```

7.96.2 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *Kneader* is designed.

Multiplicity: 0..1

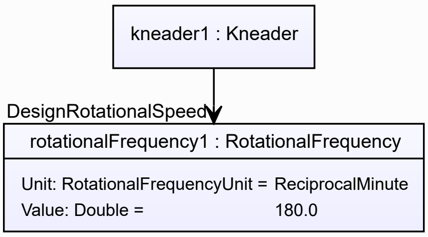
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

Example

The instance kneader1 represents a *Kneader* with a *DesignRotationalSpeed* of 180.0 min^{-1}.


kneader1 : Kneader
DesignRotationalSpeed
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 180.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="kneader1"
    ComponentClass="Kneader"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Kneader" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignRotationalSpeed"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignRotationalSpeed"
      Format="double"
      Value="180.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.96.3 DesignShaftPower

Attribute (data)

The shaft power for which the *Kneader* is designed.

Multiplicity: 0..1

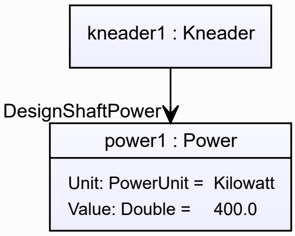
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SHAFT POWER
Name: DesignShaftPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignShaftPower

Example

The instance kneader1 represents a *Kneader* with a *DesignShaftPower* of 400.0 kW.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="kneader1"
    ComponentClass="Kneader"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Kneader" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignShaftPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignShaftPower"
      Format="double"
      Value="400.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.96.4 UpperLimitAllowableDesignPressureDrop

Attribute (data)

The upper limit for the pressure drop for which the *Kneader* is designed.

Multiplicity: 0..1

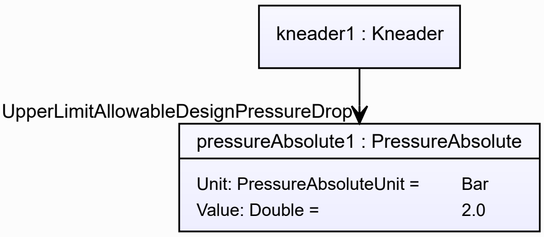
Type: *NullablePressureAbsolute*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT ALLOWABLE DESIGN PRESSURE DROP
Name: UpperLimitAllowableDesignPressureDrop
AttributeURI: http://sandbox.dexpi.org/rdl/UpperLimitAllowableDesignPressureDrop

Example

The instance kneader1 represents a *Kneader* with an *UpperLimitAllowableDesignPressureDrop* of 2.0 bar.


kneader1 : Kneader
UpperLimitAllowableDesignPressureDrop
pressureAbsolute1 : PressureAbsolute
Unit: PressureAbsoluteUnit = Bar
Value: Double = 2.0

Example: Implementation in Proteus Schema

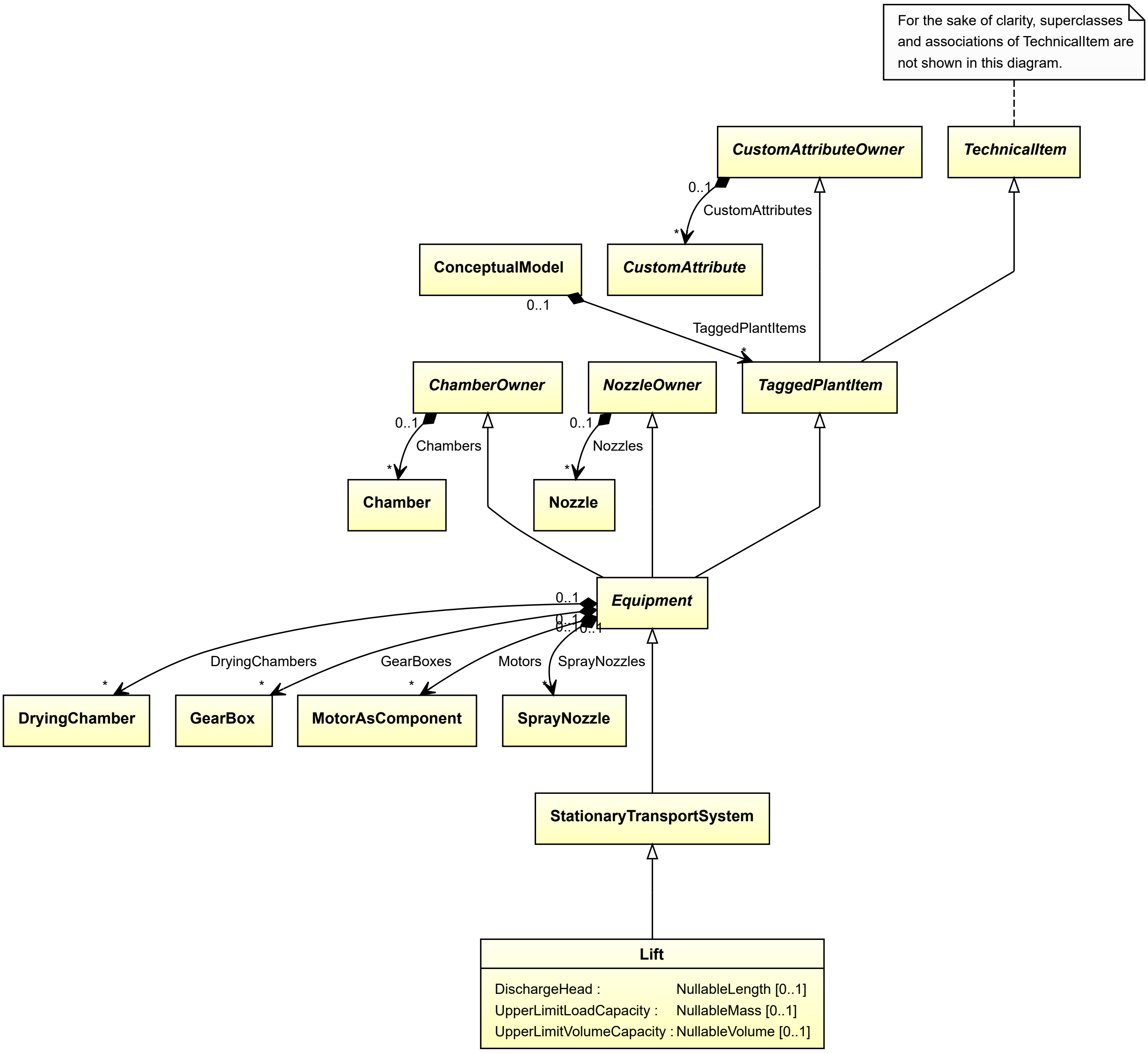
```
<Equipment
    ID="kneader1"
    ComponentClass="Kneader"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Kneader" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="UpperLimitAllowableDesignPressureDrop"
      AttributeURI="http://sandbox.dexpi.org/rdl/UpperLimitAllowableDesignPressureDrop"
      Format="double"
      Value="2.0"
      Units="Bar"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1314539" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.97. Lift

7.97.1 Overview

Class

A *StationaryTransportSystem* for transporting persons or things from one level to another (from http://data.posccaesar.org/rdl/RDS13601120).



Supertypes

- *StationaryTransportSystem*

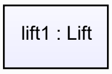
Attributes (data)

Name	Multiplicity	Type
DischargeHead	0..1	*NullableLength*
UpperLimitLoadCapacity	0..1	*NullableMass*
UpperLimitVolumeCapacity	0..1	*NullableVolume*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: LIFT
ComponentClass: Lift
ComponentClassURI: http://data.posccaesar.org/rdl/RDS13601120

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="lift1"
    ComponentClass="Lift"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS13601120" ...>
  ...
</Equipment>
```

7.97.2 DischargeHead

Attribute (data)

The length of the *Lift*.

Multiplicity: 0..1

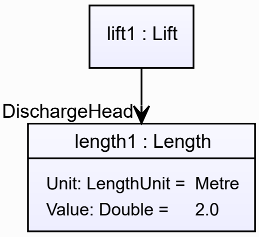
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DISCHARGE HEAD
Name: DischargeHead
AttributeURI: http://sandbox.dexpi.org/rdl/DischargeHead

Example

The instance lift1 represents a *Lift* with a *DischargeHead* of 2.0 m.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="lift1"
    ComponentClass="Lift"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS13601120" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DischargeHead"
      AttributeURI="http://sandbox.dexpi.org/rdl/DischargeHead"
      Format="double"
      Value="2.0"
      Units="Metre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1332674" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.97.3 UpperLimitLoadCapacity

Attribute (data)

The highest mass to transport for which the *Lift* is designed.

Multiplicity: 0..1

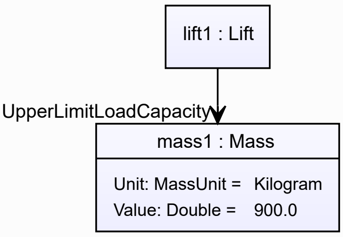
Type: *NullableMass*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT LOAD CAPACITY
Name: UpperLimitLoadCapacity
AttributeURI: http://sandbox.dexpi.org/rdl/UpperLimitLoadCapacity

Example

The instance lift1 represents a *Lift* with an *UpperLimitLoadCapacity* of 900.0 kg.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="lift1"
    ComponentClass="Lift"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS13601120" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="UpperLimitLoadCapacity"
      AttributeURI="http://sandbox.dexpi.org/rdl/UpperLimitLoadCapacity"
      Format="double"
      Value="900.0"
      Units="Kilogram"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1328669" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.97.4 UpperLimitVolumeCapacity

Attribute (data)

The highest volume to transport for which the *Lift* is designed.

Multiplicity: 0..1

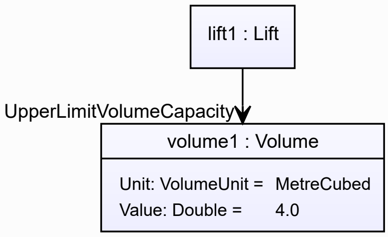
Type: *NullableVolume*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT VOLUME CAPACITY
Name: UpperLimitVolumeCapacity
AttributeURI: http://sandbox.dexpi.org/rdl/UpperLimitVolumeCapacity

Example

The instance lift1 represents a *Lift* with an *UpperLimitVolumeCapacity* of 4.0 m^3.



Example: Implementation in Proteus Schema

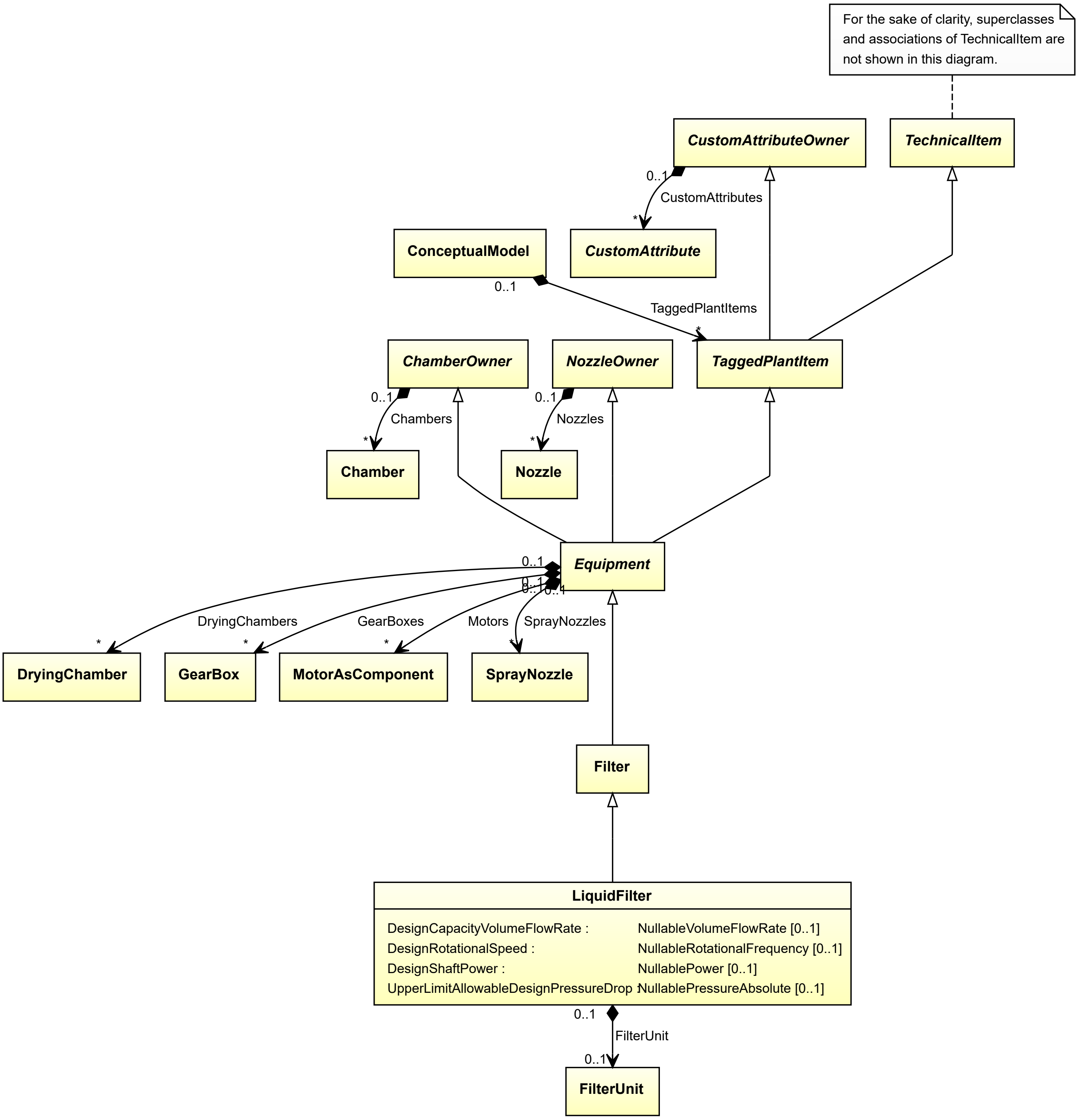
```
<Equipment
    ID="lift1"
    ComponentClass="Lift"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS13601120" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="UpperLimitVolumeCapacity"
      AttributeURI="http://sandbox.dexpi.org/rdl/UpperLimitVolumeCapacity"
      Format="double"
      Value="4.0"
      Units="MetreCubed"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1349099" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.98. LiquidFilter

7.98.1 Overview

Class

A filter that is specifically designed to filter a liquid.



Supertypes

- *Filter*

Attributes (data)

Name	Multiplicity	Type
DesignCapacityVolumeFlowRate	0..1	*NullableVolumeFlowRate*
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*
DesignShaftPower	0..1	*NullablePower*
UpperLimitAllowableDesignPressureDrop	0..1	*NullablePressureAbsolute*

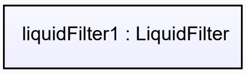
Attributes (composition)

Name	Multiplicity	Type
FilterUnit	0..1	*FilterUnit*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: LIQUID FILTER
ComponentClass: LiquidFilter
ComponentClassURI: http://sandbox.dexpi.org/rdl/LiquidFilter

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="liquidFilter1"
    ComponentClass="LiquidFilter"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/LiquidFilter" ...>
  ...
</Equipment>
```

7.98.2 DesignCapacityVolumeFlowRate

Attribute (data)

The volume flow rate for which the *LiquidFilter* is designed.

Multiplicity: 0..1

Type: *NullableVolumeFlowRate*

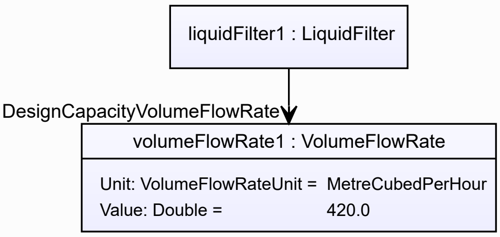
Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN CAPACITY VOLUME FLOW RATE

Name: DesignCapacityVolumeFlowRate
AttributeURI: http://sandbox.dexpi.org/rdl/DesignCapacityVolumeFlowRate

Example

The instance liquidFilter1 represents a *LiquidFilter* with a *DesignCapacityVolumeFlowRate* of 420.0 m^3/h.


liquidFilter1 : LiquidFilter
DesignCapacityVolumeFlowRate
volumeFlowRate1 : VolumeFlowRate
Unit: VolumeFlowRateUnit = MetreCubedPerHour
Value: Double = 420.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="liquidFilter1"
    ComponentClass="LiquidFilter"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/LiquidFilter" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignCapacityVolumeFlowRate"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignCapacityVolumeFlowRate"
      Format="double"
      Value="420.0"
      Units="MetreCubedPerHour"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1321064" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.98.3 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *LiquidFilter* is designed.

Multiplicity: 0..1

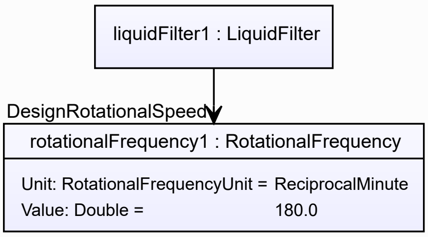
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

Example

The instance liquidFilter1 represents a *LiquidFilter* with a *DesignRotationalSpeed* of 180.0 min^{-1}.


liquidFilter1 : LiquidFilter
DesignRotationalSpeed
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 180.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="liquidFilter1"
    ComponentClass="LiquidFilter"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/LiquidFilter" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignRotationalSpeed"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignRotationalSpeed"
      Format="double"
      Value="180.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.98.4 DesignShaftPower

Attribute (data)

The shaft power for which the *LiquidFilter* is designed.

Multiplicity: 0..1

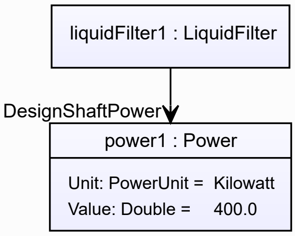
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SHAFT POWER
Name: DesignShaftPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignShaftPower

Example

The instance liquidFilter1 represents a *LiquidFilter* with a *DesignShaftPower* of 400.0 kW.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="liquidFilter1"
    ComponentClass="LiquidFilter"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/LiquidFilter" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignShaftPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignShaftPower"
      Format="double"
      Value="400.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.98.5 FilterUnit

Attribute (composition)

The filter unit of the *LiquidFilter*.

Multiplicity: 0..1

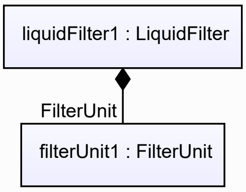
Type: *FilterUnit*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *FilterUnit*) is a child of the `<Equipment>` element for the attribute owner (a *LiquidFilter*).

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="liquidFilter1"
    ComponentClass="LiquidFilter"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/LiquidFilter" ...>
  ...
  <Equipment
      ID="filterUnit1"
      ComponentClass="FilterUnit"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/FilterUnit" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.98.6 UpperLimitAllowableDesignPressureDrop

Attribute (data)

The upper limit for the pressure drop for which the *LiquidFilter* is designed.

Multiplicity: 0..1

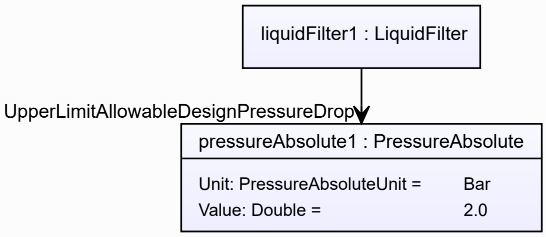
Type: *NullablePressureAbsolute*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT ALLOWABLE DESIGN PRESSURE DROP
Name: UpperLimitAllowableDesignPressureDrop
AttributeURI: http://sandbox.dexpi.org/rdl/UpperLimitAllowableDesignPressureDrop

Example

The instance liquidFilter1 represents a *LiquidFilter* with an *UpperLimitAllowableDesignPressureDrop* of 2.0 bar.


liquidFilter1 : LiquidFilter
UpperLimitAllowableDesignPressureDrop
pressureAbsolute1 : PressureAbsolute
Unit: PressureAbsoluteUnit =
Bar
Value: Double =
2.0

Example: Implementation in Proteus Schema

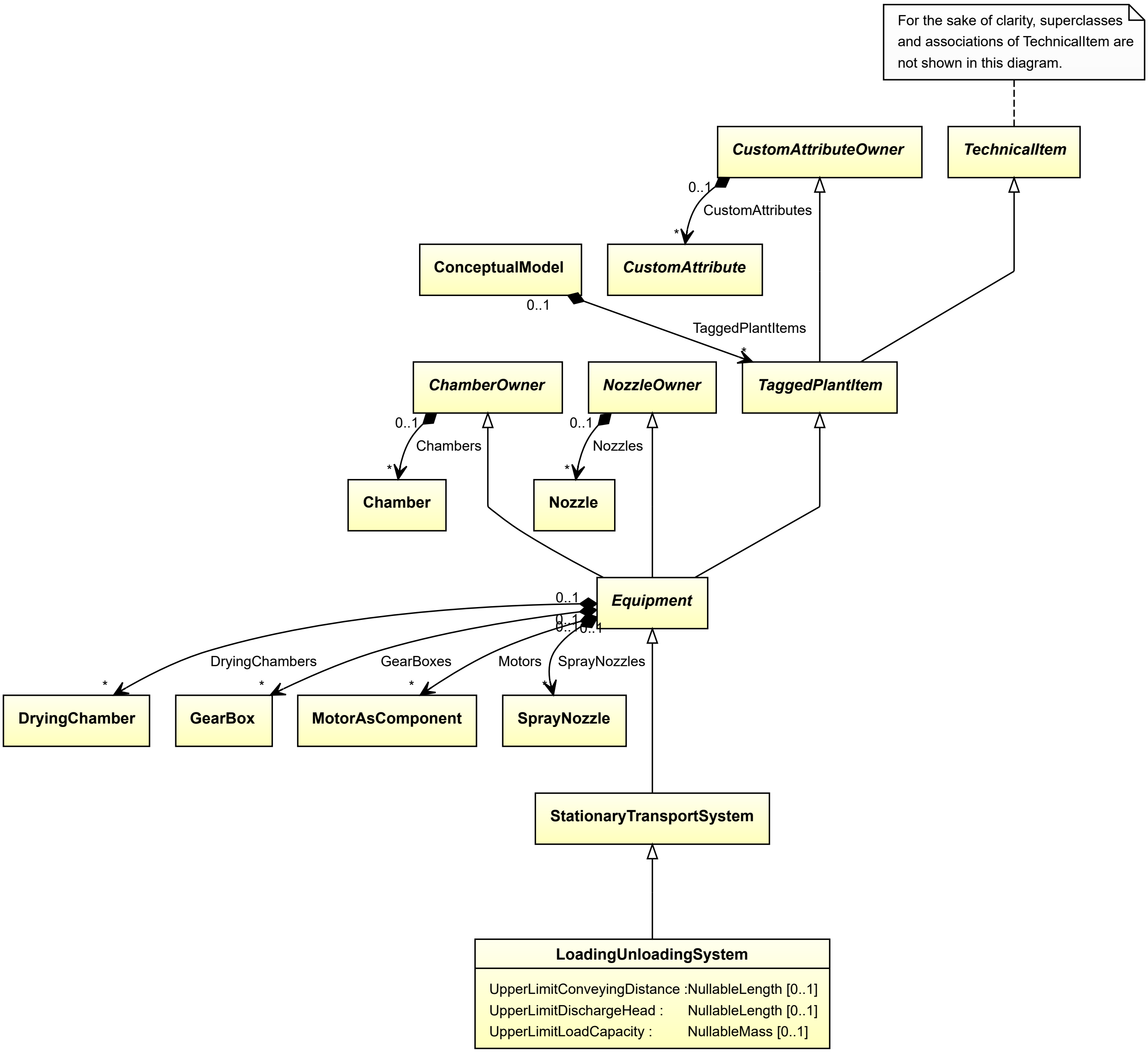
```
<Equipment
    ID="liquidFilter1"
    ComponentClass="LiquidFilter"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/LiquidFilter" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="UpperLimitAllowableDesignPressureDrop"
      AttributeURI="http://sandbox.dexpi.org/rdl/UpperLimitAllowableDesignPressureDrop"
      Format="double"
      Value="2.0"
      Units="Bar"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1314539" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.99. LoadingUnloadingSystem

7.99.1 Overview

Class

A transport system that is intended for loading and/or unloading products into/from vehicles, wagons or vessels (from http://data.posccaesar.org/rdl/RDS11525012).



Supertypes

- *StationaryTransportSystem*

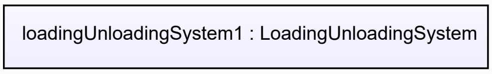
Attributes (data)

Name	Multiplicity	Type
UpperLimitConveyingDistance	0..1	*NullableLength*
UpperLimitDischargeHead	0..1	*NullableLength*
UpperLimitLoadCapacity	0..1	*NullableMass*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: LOADING - UNLOADING SYSTEM
ComponentClass: Loading-UnloadingSystem
ComponentClassURI: http://data.posccaesar.org/rdl/RDS11525012

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="loadingUnloadingSystem1"
    ComponentClass="Loading-UnloadingSystem"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11525012" ...>
  ...
</Equipment>
```

7.99.2 UpperLimitConveyingDistance

Attribute (data)

The upper limit for the conveying distance of the *LoadingUnloadingSystem*.

Multiplicity: 0..1

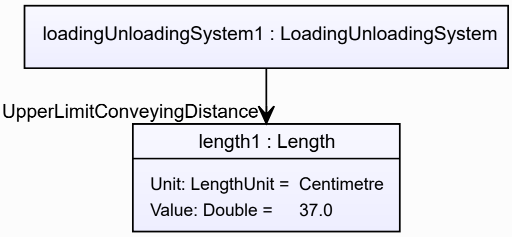
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT CONVEYING DISTANCE
Name: UpperLimitConveyingDistance
AttributeURI: http://sandbox.dexpi.org/rdl/UpperLimitConveyingDistance

Example

The instance loadingUnloadingSystem1 represents a *LoadingUnloadingSystem* with an *UpperLimitConveyingDistance* of 37.0 cm.


loadingUnloadingSystem1 : LoadingUnloadingSystem
UpperLimitConveyingDistance
length1 : Length
Unit: LengthUnit = Centimetre
Value: Double = 37.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="loadingUnloadingSystem1"
    ComponentClass="Loading-UnloadingSystem"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11525012" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="UpperLimitConveyingDistance"
      AttributeURI="http://sandbox.dexpi.org/rdl/UpperLimitConveyingDistance"
      Format="double"
      Value="37.0"
      Units="Centimetre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1318004" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.99.3 UpperLimitDischargeHead

Attribute (data)

The upper limit for the discharge head of the *LoadingUnloadingSystem*.

Multiplicity: 0..1

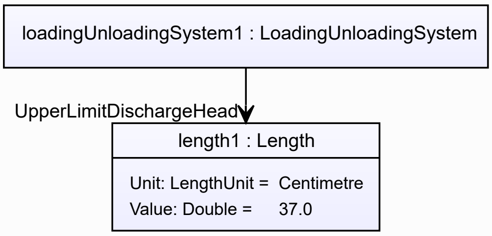
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT DISCHARGE HEAD
Name: UpperLimitDischargeHead
AttributeURI: http://sandbox.dexpi.org/rdl/UpperLimitDischargeHead

Example

The instance loadingUnloadingSystem1 represents a *LoadingUnloadingSystem* with an *UpperLimitDischargeHead* of 37.0 cm.


loadingUnloadingSystem1 : LoadingUnloadingSystem
UpperLimitDischargeHead
length1 : Length
Unit: LengthUnit = Centimetre
Value: Double = 37.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="loadingUnloadingSystem1"
    ComponentClass="Loading-UnloadingSystem"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11525012" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="UpperLimitDischargeHead"
      AttributeURI="http://sandbox.dexpi.org/rdl/UpperLimitDischargeHead"
      Format="double"
      Value="37.0"
      Units="Centimetre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1318004" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.99.4 UpperLimitLoadCapacity

Attribute (data)

The highest mass to transport for which the *LoadingUnloadingSystem* is designed.

Multiplicity: 0..1

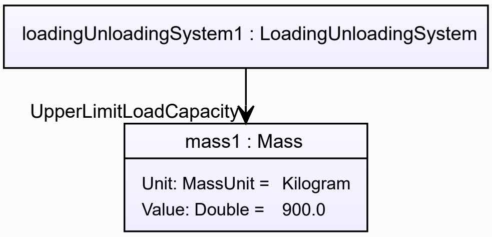
Type: *NullableMass*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT LOAD CAPACITY
Name: UpperLimitLoadCapacity
AttributeURI: http://sandbox.dexpi.org/rdl/UpperLimitLoadCapacity

Example

The instance loadingUnloadingSystem1 represents a *LoadingUnloadingSystem* with an *UpperLimitLoadCapacity* of 900.0 kg.


loadingUnloadingSystem1 : LoadingUnloadingSystem
UpperLimitLoadCapacity
mass1 : Mass
Unit: MassUnit = Kilogram
Value: Double = 900.0

Example: Implementation in Proteus Schema

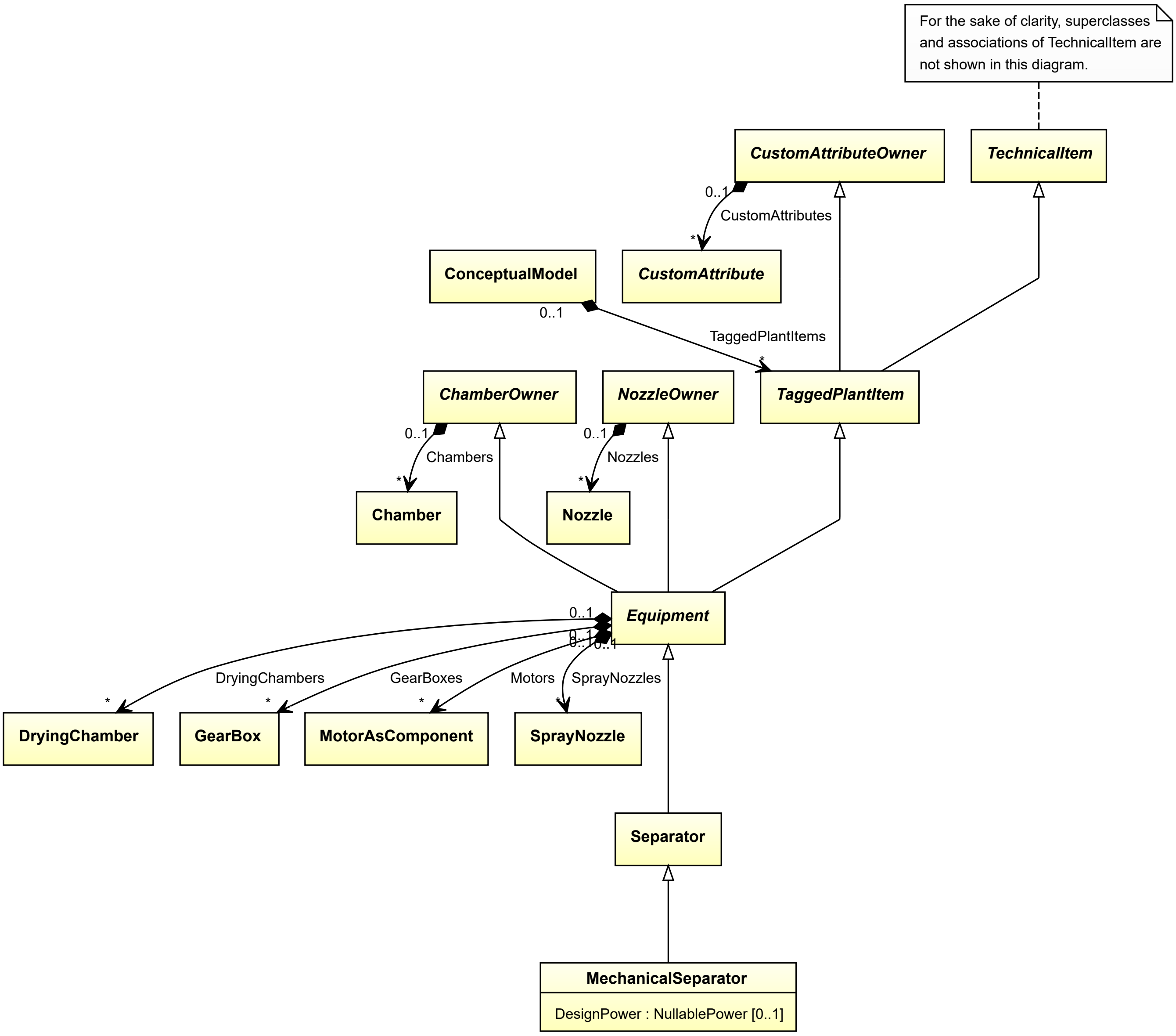
```
<Equipment
    ID="loadingUnloadingSystem1"
    ComponentClass="Loading-UnloadingSystem"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11525012" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="UpperLimitLoadCapacity"
      AttributeURI="http://sandbox.dexpi.org/rdl/UpperLimitLoadCapacity"
      Format="double"
      Value="900.0"
      Units="Kilogram"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1328669" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.100. MechanicalSeparator

7.100.1 Overview

Class

A fluid separator in which mechanical separation of fluids take place (from http://data.posccaesar.org/rdl/RDS279134).



Supertypes

- *Separator*

Attributes (data)

Name	Multiplicity	Type
DesignPower	0..1	*NullablePower*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: MECHANICAL SEPARATOR
ComponentClass: MechanicalSeparator
ComponentClassURI: http://data.posccaesar.org/rdl/RDS279134

Example

mechanicalSeparator1 : MechanicalSeparator

Example: Implementation in Proteus Schema

```
<Equipment
    ID="mechanicalSeparator1"
    ComponentClass="MechanicalSeparator"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS279134" ...>
  ...
</Equipment>
```

7.100.2 DesignPower

Attribute (data)

The power for which the *MechanicalSeparator* is designed.

Multiplicity: 0..1

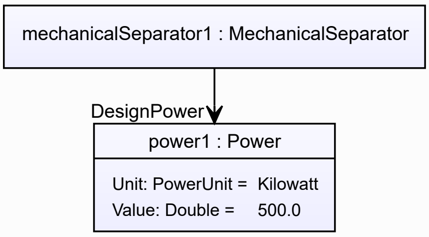
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN POWER
Name: DesignPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignPower

Example

The instance mechanicalSeparator1 represents a *MechanicalSeparator* with a *DesignPower* of 500.0 kW.


mechanicalSeparator1 : MechanicalSeparator
DesignPower
power1 : Power
Unit: PowerUnit = Kilowatt
Value: Double = 500.0

Example: Implementation in Proteus Schema

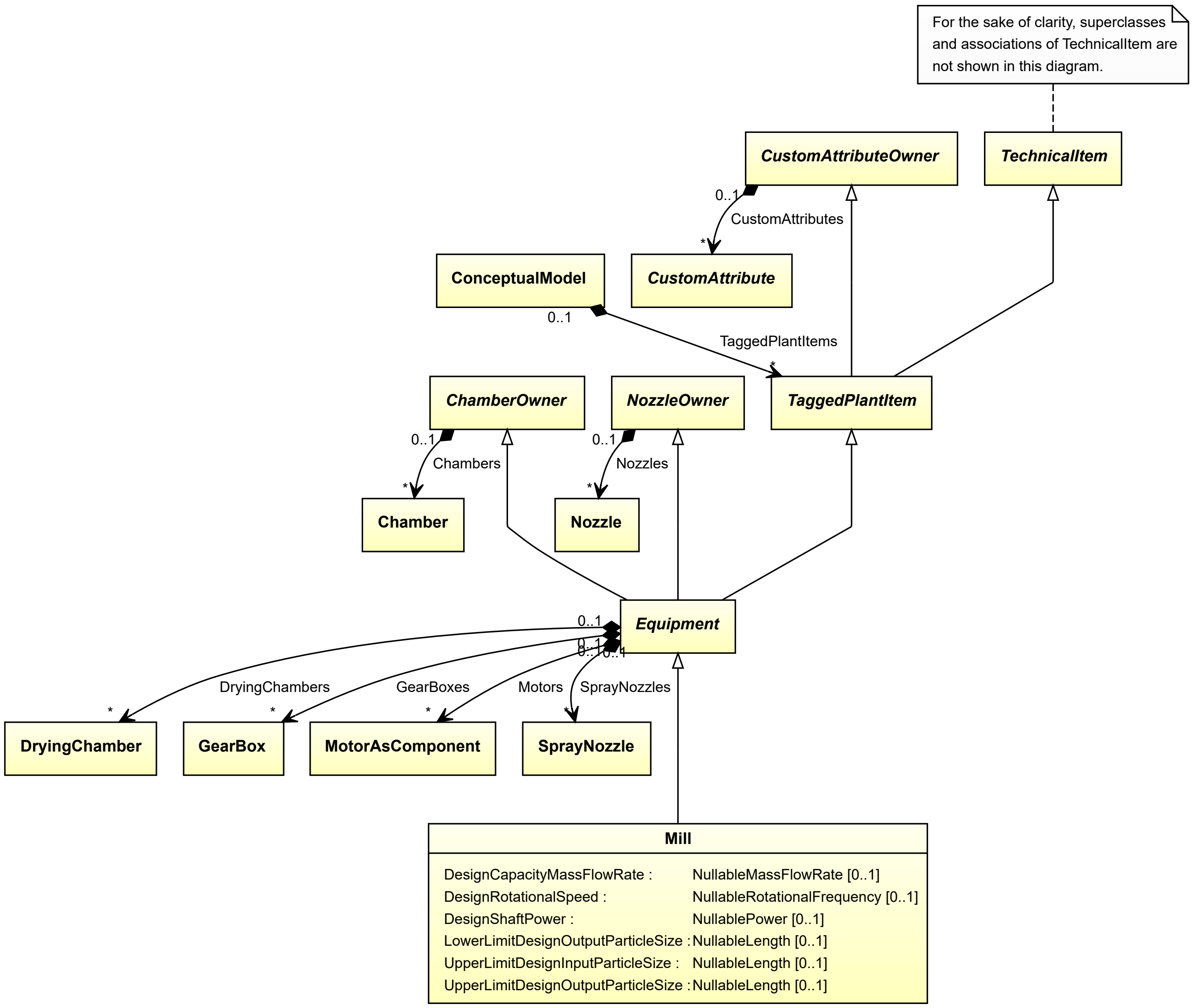
```
<Equipment
    ID="mechanicalSeparator1"
    ComponentClass="MechanicalSeparator"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS279134" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignPower"
      Format="double"
      Value="500.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.101. Mill

7.101.1 Overview

Class

A physical object for grinding or pulverizing materials. Also a machine for shaping metal. In general a machine that manufactures by the continuous repetition of some simple action (from http://data.posccaesar.org/rdl/RDS11589220).


For the sake of clarity, superclasses and associations of TechnicalItem are not shown in this diagram.
CustomAttributeOwner
TechnicalItem
0..1
CustomAttributes
*
ConceptualModel
CustomAttribute
0..1
TaggedPlantItems
*
ChamberOwner
NozzleOwner
TaggedPlantItem
0..1
Chambers
0..1
Nozzles
*
*
Chamber
Nozzle
Equipment
0..1
0..1
0..1
0..1
DryingChambers
GearBoxes
Motors
SprayNozzles
*
*
*
*
DryingChamber
GearBox
MotorAsComponent
SprayNozzle
Mill
DesignCapacityMassFlowRate : NullableMassFlowRate [0..1]
DesignRotationalSpeed : NullableRotationalFrequency [0..1]
DesignShaftPower : NullablePower [0..1]
LowerLimitDesignOutputParticleSize : NullableLength [0..1]
UpperLimitDesignInputParticleSize : NullableLength [0..1]
UpperLimitDesignOutputParticleSize : NullableLength [0..1]

Supertypes

- *Equipment*

Subtypes

- *Crusher*
- *CustomMill*
- *Grinder*

Attributes (data)

Name	Multiplicity	Type
DesignCapacityMassFlowRate	0..1	*NullableMassFlowRate*
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*
DesignShaftPower	0..1	*NullablePower*
LowerLimitDesignOutputParticleSize	0..1	*NullableLength*
UpperLimitDesignInputParticleSize	0..1	*NullableLength*
UpperLimitDesignOutputParticleSize	0..1	*NullableLength*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: MILL
ComponentClass: Mill
ComponentClassURI: http://data.posccaesar.org/rdl/RDS11589220

Example

Example: Implementation in Proteus Schema

```
<Equipment
    ID="mill1"
    ComponentClass="Mill"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11589220" ...>
  ...
</Equipment>
```

7.101.2 DesignCapacityMassFlowRate

Attribute (data)

The capacity for the mass flow rate for which the *Mill* is designed.

Multiplicity: 0..1

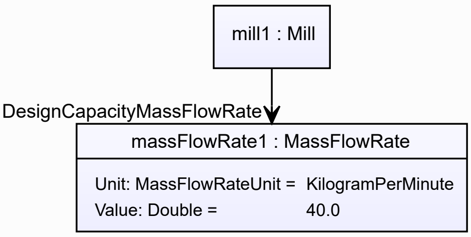
Type: *NullableMassFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN CAPACITY MASS FLOW RATE
Name: DesignCapacityMassFlowRate
AttributeURI: http://sandbox.dexpi.org/rdl/DesignCapacityMassFlowRate

Example

The instance mill1 represents a *Mill* with a *DesignCapacityMassFlowRate* of 40.0 kg/min.


mill1 : Mill
DesignCapacityMassFlowRate
massFlowRate1 : MassFlowRate
Unit: MassFlowRateUnit = KilogramPerMinute
Value: Double = 40.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="mill1"
    ComponentClass="Mill"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11589220" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignCapacityMassFlowRate"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignCapacityMassFlowRate"
      Format="double"
      Value="40.0"
      Units="KilogramPerMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1350719" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.101.3 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *Mill* is designed.

Multiplicity: 0..1

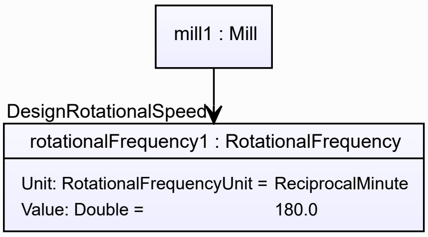
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

Example

The instance mill1 represents a *Mill* with a *DesignRotationalSpeed* of 180.0 min^{-1}.


mill1 : Mill
DesignRotationalSpeed
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 180.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="mill1"
    ComponentClass="Mill"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11589220" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignRotationalSpeed"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignRotationalSpeed"
      Format="double"
      Value="180.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.101.4 DesignShaftPower

Attribute (data)

The shaft power for which the *Mill* is designed.

Multiplicity: 0..1

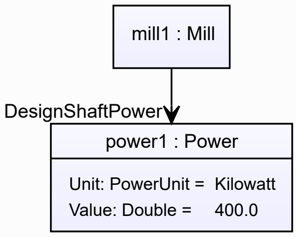
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SHAFT POWER
Name: DesignShaftPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignShaftPower

Example

The instance mill1 represents a *Mill* with a *DesignShaftPower* of 400.0 kW.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="mill1"
    ComponentClass="Mill"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11589220" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignShaftPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignShaftPower"
      Format="double"
      Value="400.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.101.5 LowerLimitDesignOutputParticleSize

Attribute (data)

The lower limit for the output particle size for which the *Mill* is designed.

Multiplicity: 0..1

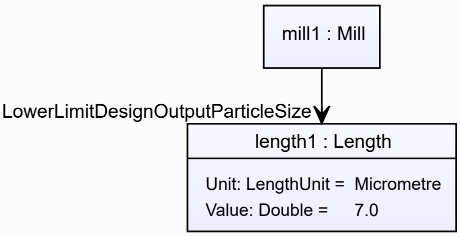
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: LOWER LIMIT DESIGN OUTPUT PARTICLE SIZE
Name: LowerLimitDesignOutputParticleSize
AttributeURI: http://sandbox.dexpi.org/rdl/LowerLimitDesignOutputParticleSize

Example

The instance mill1 represents a *Mill* with a *LowerLimitDesignOutputParticleSize* of 7.0 µm.


mill1 : Mill
LowerLimitDesignOutputParticleSize
length1 : Length
Unit: LengthUnit = Micrometre
Value: Double = 7.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="mill1"
    ComponentClass="Mill"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11589220" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="LowerLimitDesignOutputParticleSize"
      AttributeURI="http://sandbox.dexpi.org/rdl/LowerLimitDesignOutputParticleSize"
      Format="double"
      Value="7.0"
      Units="Micrometre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1351529" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.101.6 UpperLimitDesignInputParticleSize

Attribute (data)

The upper limit for the input particle size for which the *Mill* is designed.

Multiplicity: 0..1

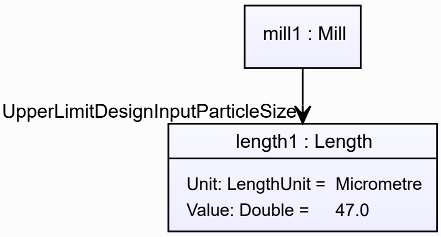
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT DESIGN INPUT PARTICLE SIZE
Name: UpperLimitDesignInputParticleSize
AttributeURI: http://sandbox.dexpi.org/rdl/UpperLimitDesignInputParticleSize

Example

The instance mill1 represents a *Mill* with an *UpperLimitDesignInputParticleSize* of 47.0 µm.


mill1 : Mill
UpperLimitDesignInputParticleSize
length1 : Length
Unit: LengthUnit = Micrometre
Value: Double = 47.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="mill1"
    ComponentClass="Mill"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11589220" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="UpperLimitDesignInputParticleSize"
      AttributeURI="http://sandbox.dexpi.org/rdl/UpperLimitDesignInputParticleSize"
      Format="double"
      Value="47.0"
      Units="Micrometre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1351529" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.101.7 UpperLimitDesignOutputParticleSize

Attribute (data)

The upper limit for the output particle size for which the *Mill* is designed.

Multiplicity: 0..1

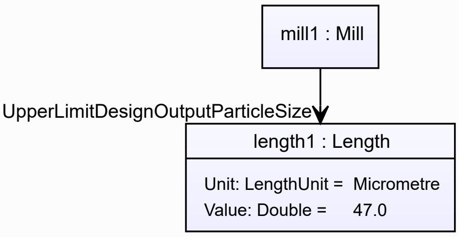
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT DESIGN OUTPUT PARTICLE SIZE
Name: UpperLimitDesignOutputParticleSize
AttributeURI: http://sandbox.dexpi.org/rdl/UpperLimitDesignOutputParticleSize

Example

The instance mill1 represents a *Mill* with an *UpperLimitDesignOutputParticleSize* of 47.0 µm.


mill1 : Mill
UpperLimitDesignOutputParticleSize
length1 : Length
Unit: LengthUnit = Micrometre
Value: Double = 47.0

Example: Implementation in Proteus Schema

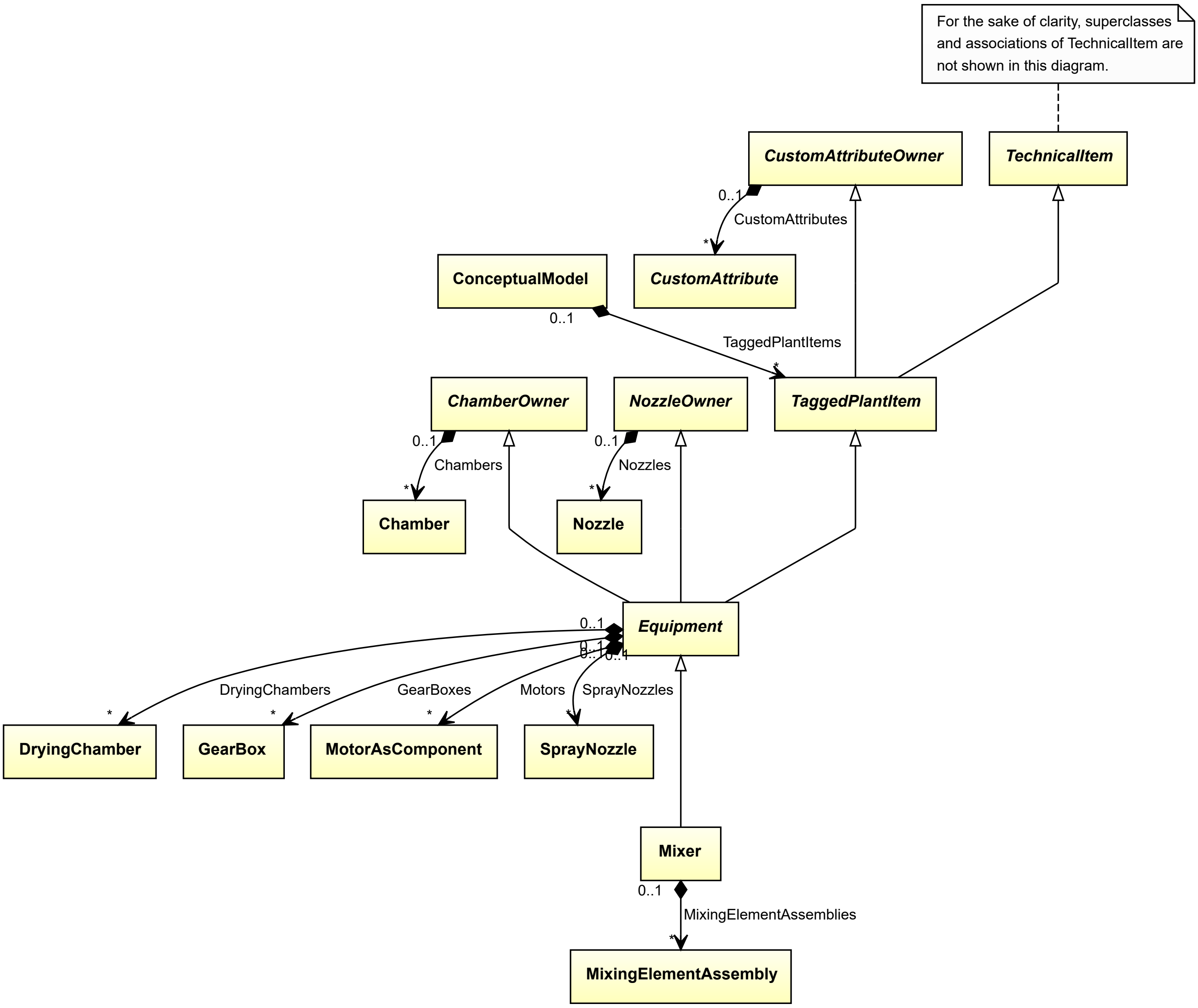
```
<Equipment
    ID="mill1"
    ComponentClass="Mill"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11589220" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="UpperLimitDesignOutputParticleSize"
      AttributeURI="http://sandbox.dexpi.org/rdl/UpperLimitDesignOutputParticleSize"
      Format="double"
      Value="47.0"
      Units="Micrometre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1351529" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.102. Mixer

7.102.1 Overview

Class

An apparatus or machine that has the capability of mixing (from http://data.15926.org/rdl/RDS222370).



Supertypes

- *Equipment*

Subtypes

- *CustomMixer*
- *Kneader*
- *RotaryMixer*
- *StaticMixer*

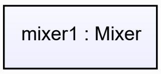
Attributes (composition)

Name	Multiplicity	Type
MixingElementAssemblies	*	*MixingElementAssembly*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: MIXER
ComponentClass: Mixer
ComponentClassURI: http://sandbox.dexpi.org/rdl/Mixer

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="mixer1"
    ComponentClass="Mixer"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Mixer" ...>
  ...
</Equipment>
```

7.102.2 MixingElementAssemblies

Attribute (composition)

The mixing element assemblies of the *Mixer*, if applicable.

Multiplicity: *

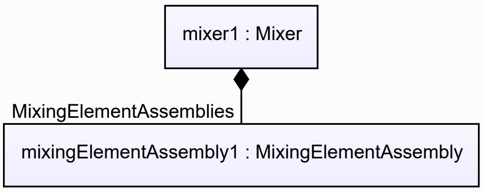
Type: *MixingElementAssembly*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *MixingElementAssembly*) is a child of the `<Equipment>` element for the attribute owner (a *Mixer*).

Example



Example: Implementation in Proteus Schema

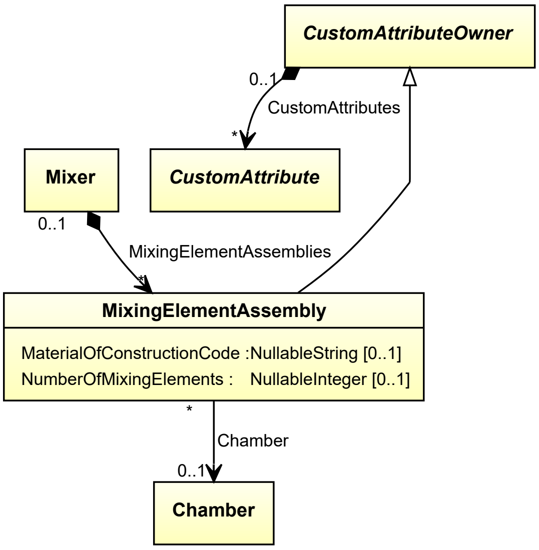
```
<Equipment
    ID="mixer1"
    ComponentClass="Mixer"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Mixer" ...>
  ...
  <Equipment
      ID="mixingElementAssembly1"
      ComponentClass="MixingElementAssembly"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/MixingElementAssembly" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.103. MixingElementAssembly

7.103.1 Overview

Class

Assembly of mixing elements as part of a mixer.



Supertypes

- *CustomAttributeOwner*

Attributes (data)

Name	Multiplicity	Type
MaterialOfConstructionCode	0..1	*NullableString*
NumberOfMixingElements	0..1	*NullableInteger*

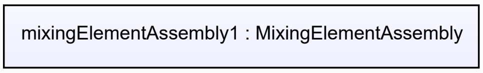
Attributes (reference)

Name	Multiplicity	Type
Chamber	0..1	*Chamber*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: MIXING ELEMENT ASSEMBLY
ComponentClass: MixingElementAssembly
ComponentClassURI: http://sandbox.dexpi.org/rdl/MixingElementAssembly

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="mixingElementAssembly1"
    ComponentClass="MixingElementAssembly"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MixingElementAssembly" ...>
  ...
</Equipment>
```

7.103.2 Chamber

Attribute (reference)

The *Chamber* in which the *MixingElementAssembly* is located, if applicable. The Chamber must be a component of the same object as the MixingElementAssembly.

Multiplicity: 0..1

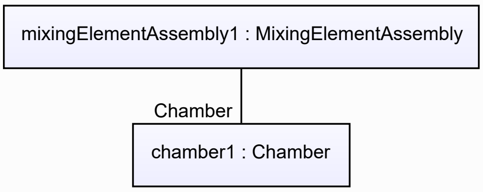
Type: *Chamber*

Opposite multiplicity: 0..*

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: `"is located in"`
Opposite association type: `"is the location of"`

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="mixingElementAssembly1"
    ComponentClass="MixingElementAssembly"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MixingElementAssembly" ...>
  ...
  <Association
    Type="is located in"
    ItemID="chamber1" />
  ...
<Equipment />
...
<Equipment
    ID="chamber1"
    ComponentClass="Chamber"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS903151421" ...>
  ...
  <Association
    Type="is the location of"
    ItemID="mixingElementAssembly1" />
  ...
<Equipment />
```

7.103.3 MaterialOfConstructionCode

Attribute (data)

A code that gives the material of construction of the *MixingElementAssembly*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: MATERIAL OF CONSTRUCTION CODE ASSIGNMENT CLASS
Name: MaterialOfConstructionCodeAssignmentClass
AttributeURI: http://data.posccaesar.org/rdl/RDS1460719741

Example

“1.4306” (*String*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="mixingElementAssembly1"
    ComponentClass="MixingElementAssembly"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MixingElementAssembly" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="MaterialOfConstructionCodeAssignmentClass"
      AttributeURI="http://data.posccaesar.org/rdl/RDS1460719741"
      Format="string"
      Value="1.4306" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.103.4 NumberOfMixingElements

Attribute (data)

The number of mixing elements in the *MixingElementAssembly*.

Multiplicity: 0..1

Type: *NullableInteger*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for integer values*.
RDL reference: NUMBER OF MIXING ELEMENTS
Name: NumberOfMixingElements
AttributeURI: http://sandbox.dexpi.org/rdl/NumberOfMixingElements

Example

5 (*Integer*)

Example: Implementation in Proteus Schema

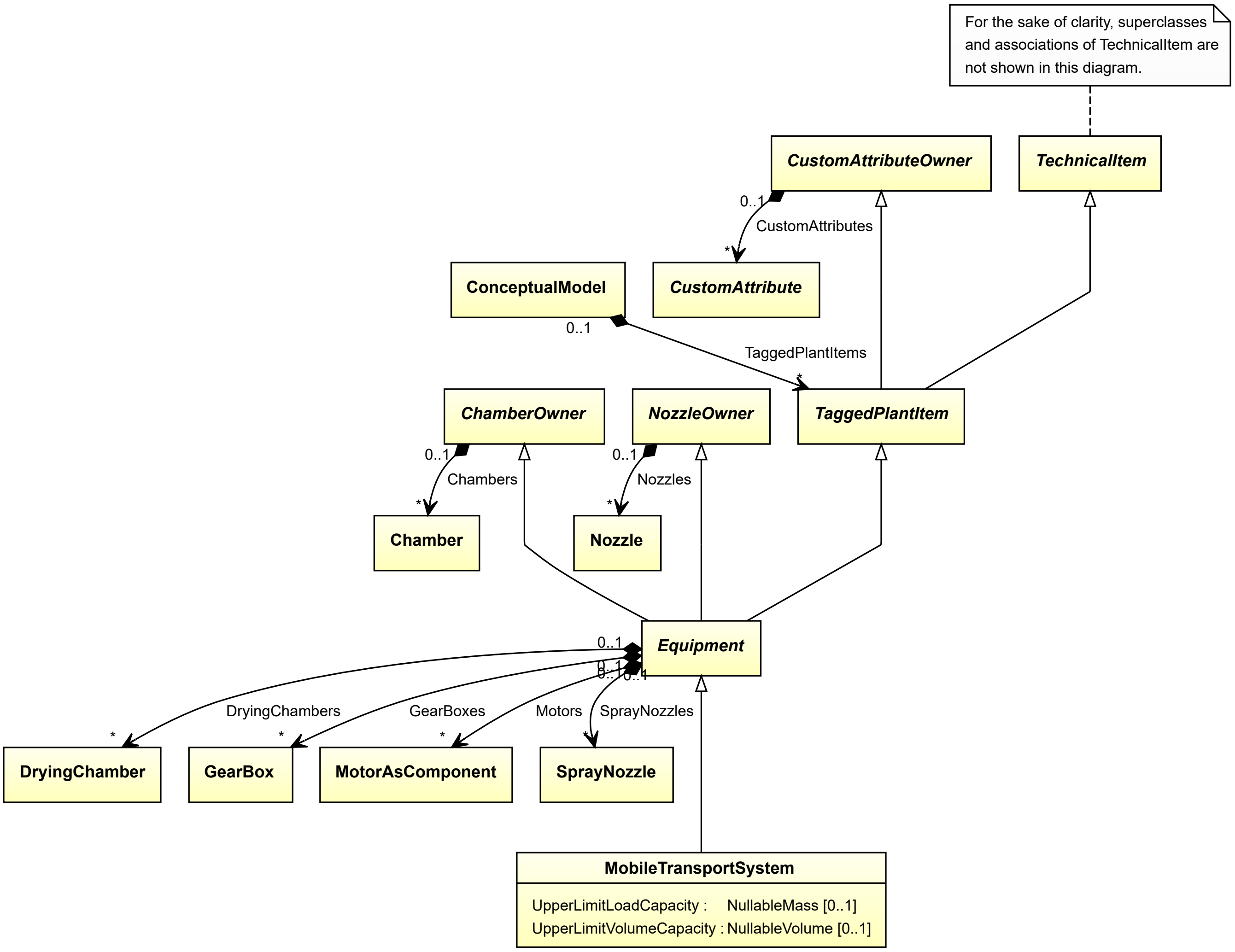
```
<Equipment
    ID="mixingElementAssembly1"
    ComponentClass="MixingElementAssembly"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MixingElementAssembly" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NumberOfMixingElements"
      AttributeURI="http://sandbox.dexpi.org/rdl/NumberOfMixingElements"
      Format="integer"
      Value="5" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.104. MobileTransportSystem

7.104.1 Overview

Class

A mobile system that is intended to transport, store or load/unload material.



Supertypes

- *Equipment*

Subtypes

- *CustomMobileTransportSystem*
- *ForkliftTruck*
- *RailWaggon*
- *Ship*
- *TransportableContainer*
- *Truck*

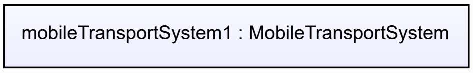
Attributes (data)

Name	Multiplicity	Type
UpperLimitLoadCapacity	0..1	*NullableMass*
UpperLimitVolumeCapacity	0..1	*NullableVolume*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: MOBILE TRANSPORT SYSTEM
ComponentClass: MobileTransportSystem
ComponentClassURI: http://sandbox.dexpi.org/rdl/MobileTransportSystem

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="mobileTransportSystem1"
    ComponentClass="MobileTransportSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MobileTransportSystem" ...>
  ...
</Equipment>
```

7.104.2 UpperLimitLoadCapacity

Attribute (data)

The highest mass to transport for which the *MobileTransportSystem* is designed.

Multiplicity: 0..1

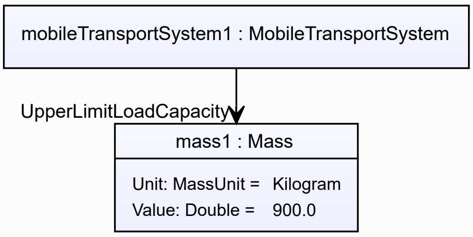
Type: *NullableMass*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT LOAD CAPACITY
Name: UpperLimitLoadCapacity
AttributeURI: http://sandbox.dexpi.org/rdl/UpperLimitLoadCapacity

Example

The instance mobileTransportSystem1 represents a *MobileTransportSystem* with an *UpperLimitLoadCapacity* of 900.0 kg.


mobileTransportSystem1 : MobileTransportSystem
UpperLimitLoadCapacity
mass1 : Mass
Unit: MassUnit = Kilogram
Value: Double = 900.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="mobileTransportSystem1"
    ComponentClass="MobileTransportSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MobileTransportSystem" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="UpperLimitLoadCapacity"
      AttributeURI="http://sandbox.dexpi.org/rdl/UpperLimitLoadCapacity"
      Format="double"
      Value="900.0"
      Units="Kilogram"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1328669" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.104.3 UpperLimitVolumeCapacity

Attribute (data)

The highest volume to transport for which the *MobileTransportSystem* is designed.

Multiplicity: 0..1

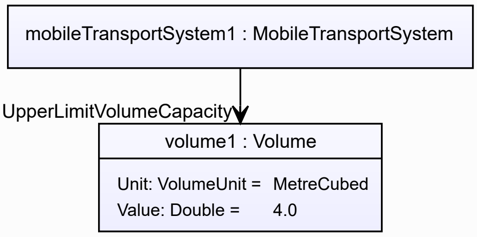
Type: *NullableVolume*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT VOLUME CAPACITY
Name: UpperLimitVolumeCapacity
AttributeURI: http://sandbox.dexpi.org/rdl/UpperLimitVolumeCapacity

Example

The instance mobileTransportSystem1 represents a *MobileTransportSystem* with an *UpperLimitVolumeCapacity* of 4.0 m^3.


mobileTransportSystem1 : MobileTransportSystem
UpperLimitVolumeCapacity
volume1 : Volume
Unit: VolumeUnit = MetreCubed
Value: Double = 4.0

Example: Implementation in Proteus Schema

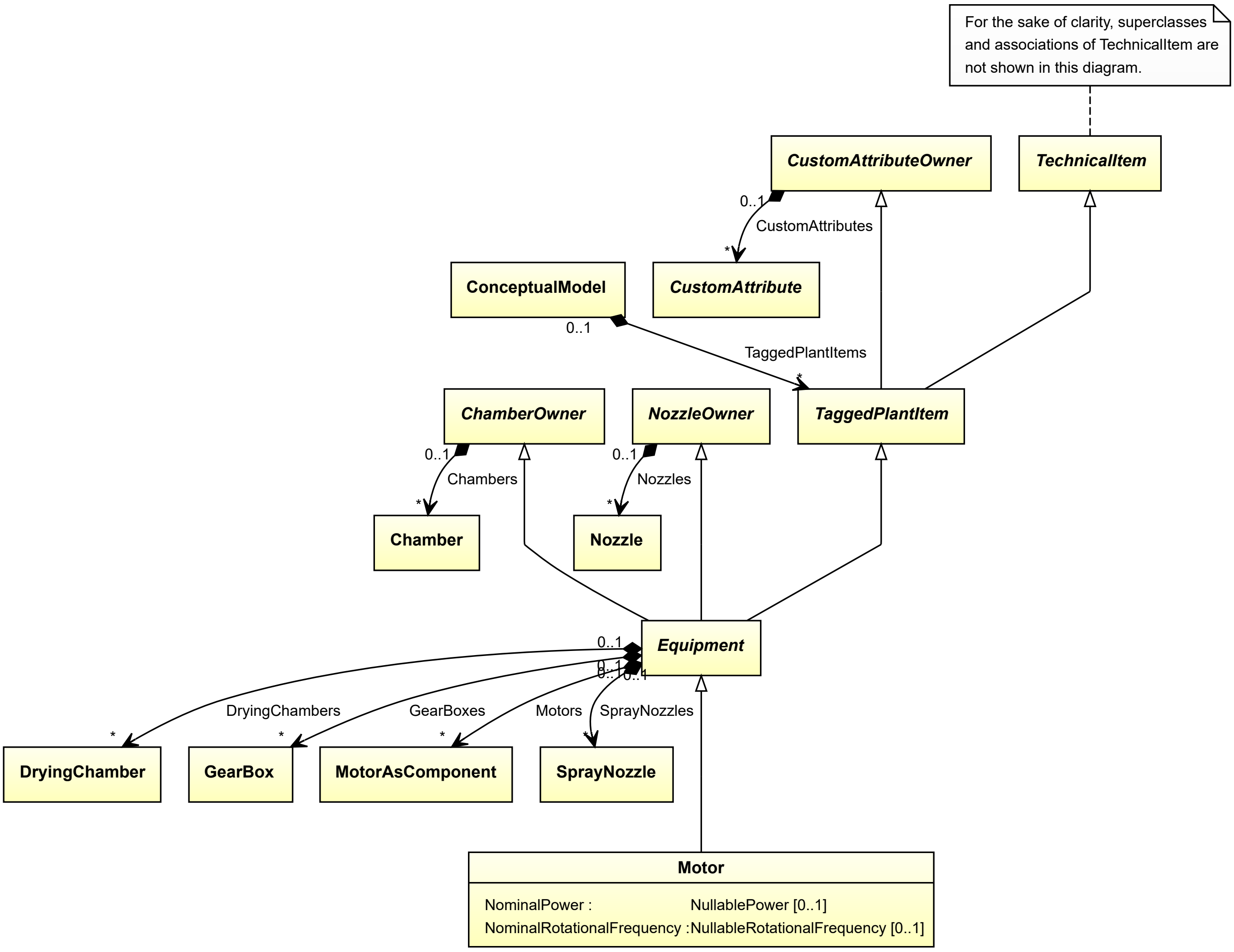
```
<Equipment
    ID="mobileTransportSystem1"
    ComponentClass="MobileTransportSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MobileTransportSystem" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="UpperLimitVolumeCapacity"
      AttributeURI="http://sandbox.dexpi.org/rdl/UpperLimitVolumeCapacity"
      Format="double"
      Value="4.0"
      Units="MetreCubed"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1349099" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.105. Motor

7.105.1 Overview

Class

A driver that is powered by electricity or internal combustion (from http://data.15926.org/rdl/RDS7191198).



Supertypes

- *Equipment*

Subtypes

- *AlternatingCurrentMotor*
- *CombustionEngine*
- *CustomMotor*
- *DirectCurrentMotor*

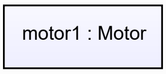
Attributes (data)

Name	Multiplicity	Type
NominalPower	0..1	*NullablePower*
NominalRotationalFrequency	0..1	*NullableRotationalFrequency*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: MOTOR
ComponentClass: Motor
ComponentClassURI: http://data.posccaesar.org/rdl/RDS7191198

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="motor1"
    ComponentClass="Motor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS7191198" ...>
  ...
</Equipment>
```

7.105.2 NominalPower

Attribute (data)

The nominal power of the *Motor*.

Multiplicity: 0..1

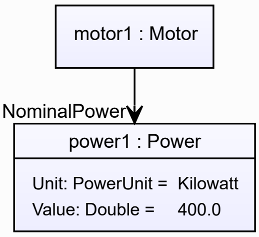
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: NOMINAL POWER
Name: NominalPower
AttributeURI: http://sandbox.dexpi.org/rdl/NominalPower

Example

The instance motor1 represents a *Motor* with a *NominalPower* of 400.0 kW.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="motor1"
    ComponentClass="Motor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS7191198" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NominalPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/NominalPower"
      Format="double"
      Value="400.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.105.3 NominalRotationalFrequency

Attribute (data)

The nominal rotational frequency of the *Motor*.

Multiplicity: 0..1

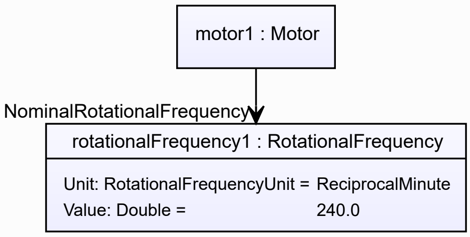
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: NOMINAL ROTATIONAL FREQUENCY
Name: NominalRotationalFrequency
AttributeURI: http://sandbox.dexpi.org/rdl/NominalRotationalFrequency

Example

The instance motor1 represents a *Motor* with a *NominalRotationalFrequency* of 240.0 min^{-1}.


motor1 : Motor
NominalRotationalFrequency
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 240.0

Example: Implementation in Proteus Schema

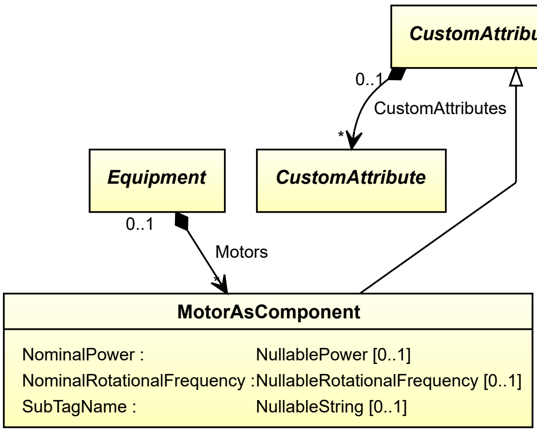
```
<Equipment
    ID="motor1"
    ComponentClass="Motor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS7191198" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NominalRotationalFrequency"
      AttributeURI="http://sandbox.dexpi.org/rdl/NominalRotationalFrequency"
      Format="double"
      Value="240.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.106. MotorAsComponent

7.106.1 Overview

Class

A driver that is powered by electricity or internal combustion and is used as component of an apparatus or of a machine.


CustomAttributeOwner
0..1
CustomAttributes
*
Equipment
CustomAttribute
0..1
Motors
MotorAsComponent
NominalPower : NullablePower [0..1]
NominalRotationalFrequency : NullableRotationalFrequency [0..1]
SubTagName : NullableString [0..1]

Supertypes

- *CustomAttributeOwner*

Subtypes

- *AlternatingCurrentMotorAsComponent*
- *CombustionEngineAsComponent*
- *DirectCurrentMotorAsComponent*

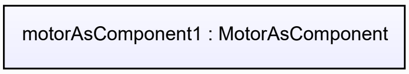
Attributes (data)

Name	Multiplicity	Type
NominalPower	0..1	*NullablePower*
NominalRotationalFrequency	0..1	*NullableRotationalFrequency*
SubTagName	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: MOTOR AS COMPONENT
ComponentClass: MotorAsComponent
ComponentClassURI: http://sandbox.dexpi.org/rdl/MotorAsComponent

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="motorAsComponent1"
    ComponentClass="MotorAsComponent"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MotorAsComponent" ...>
  ...
</Equipment>
```

7.106.2 NominalPower

Attribute (data)

The nominal power of the *MotorAsComponent*.

Multiplicity: 0..1

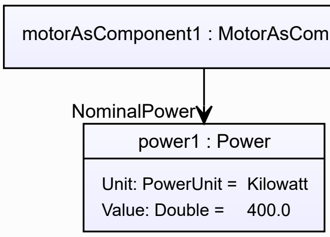
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: NOMINAL POWER
Name: NominalPower
AttributeURI: http://sandbox.dexpi.org/rdl/NominalPower

Example

The instance motorAsComponent1 represents a *MotorAsComponent* with a *NominalPower* of 400.0 kW.


motorAsComponent1 : MotorAsComponent
NominalPower
power1 : Power
Unit: PowerUnit = Kilowatt
Value: Double = 400.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="motorAsComponent1"
    ComponentClass="MotorAsComponent"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MotorAsComponent" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NominalPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/NominalPower"
      Format="double"
      Value="400.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.106.3 NominalRotationalFrequency

Attribute (data)

The nominal rotational frequency of the *MotorAsComponent*.

Multiplicity: 0..1

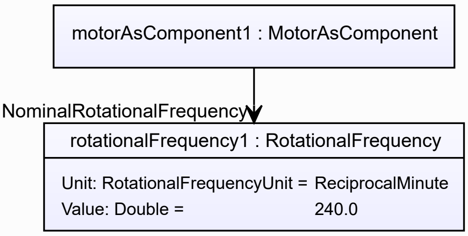
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: NOMINAL ROTATIONAL FREQUENCY
Name: NominalRotationalFrequency
AttributeURI: http://sandbox.dexpi.org/rdl/NominalRotationalFrequency

Example

The instance motorAsComponent1 represents a *MotorAsComponent* with a *NominalRotationalFrequency* of 240.0 min^{-1}.


motorAsComponent1 : MotorAsComponent
NominalRotationalFrequency
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 240.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="motorAsComponent1"
    ComponentClass="MotorAsComponent"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MotorAsComponent" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NominalRotationalFrequency"
      AttributeURI="http://sandbox.dexpi.org/rdl/NominalRotationalFrequency"
      Format="double"
      Value="240.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.106.4 SubTagName

Attribute (data)

The sub tag name of the *MotorAsComponent*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: SUB TAG NAME ASSIGNMENT CLASS
Name: SubTagNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/SubTagNameAssignmentClass

Example

“ST1” (*String*)

Example: Implementation in Proteus Schema

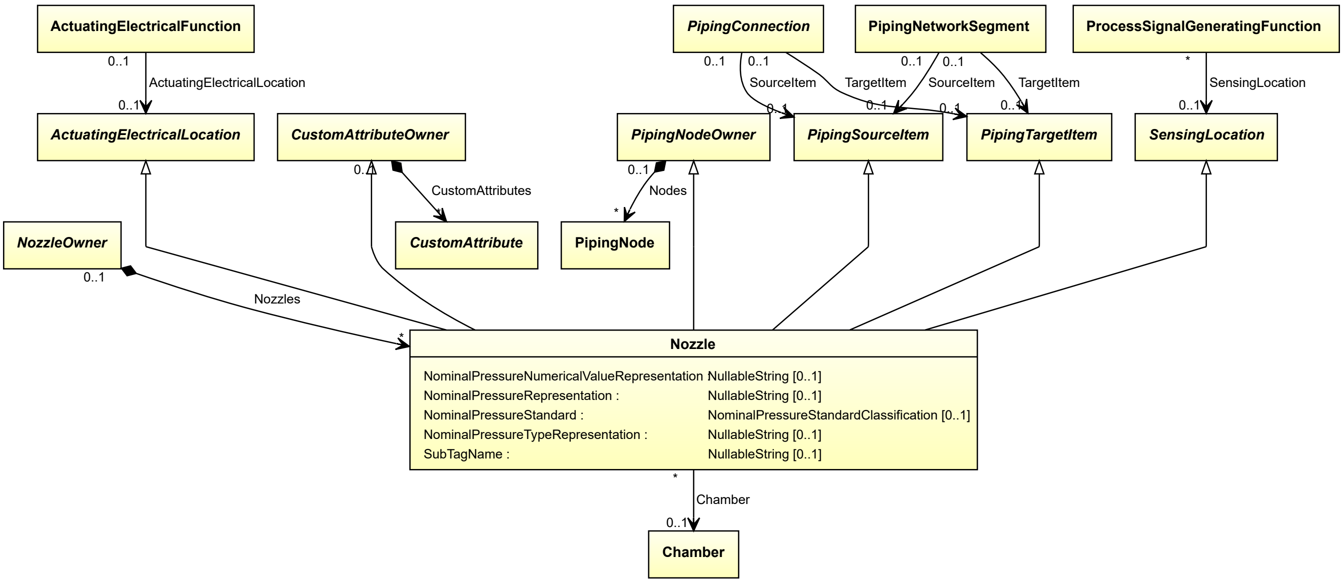
```
<Equipment
    ID="motorAsComponent1"
    ComponentClass="MotorAsComponent"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MotorAsComponent" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="SubTagNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/SubTagNameAssignmentClass"
      Format="string"
      Value="ST1" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.107. Nozzle

7.107.1 Overview

Class

A physical object that has a protruding part through which a stream of fluid is directed (from http://data.posccaesar.org/rdl/RDS415214).


ActuatingElectricalFunction
0..1
ActuatingElectricalLocation
0..1
ActuatingElectricalLocation
CustomAttributeOwner
0..1
CustomAttributes
CustomAttribute
PipingConnection
PipingNetworkSegment
ProcessSignalGeneratingFunction
0..1
0..1
SourceItem
TargetItem
0..1
0..1
SourceItem
TargetItem
*
SensingLocation
PipingNodeOwner
PipingSourceItem
PipingTargetItem
SensingLocation
0..1
0..1
0..1
0..1
0..1
0..1
Nodes
*
PipingNode
NozzleOwner
0..1
Nozzles
*
Nozzle
NominalPressureNumericalValueRepresentation NullableString [0..1]
NominalPressureRepresentation : NullableString [0..1]
NominalPressureStandard : NominalPressureStandardClassification [0..1]
NominalPressureTypeRepresentation : NullableString [0..1]
SubTagName : NullableString [0..1]
*
Chamber
0..1
Chamber

Supertypes

- *ActuatingElectricalLocation*
- *CustomAttributeOwner*
- *PipingNodeOwner*
- *PipingSourceItem*
- *PipingTargetItem*
- *SensingLocation*

Attributes (data)

Name	Multiplicity	Type
NominalPressureNumericalValueRepresentation	0..1	*NullableString*
NominalPressureRepresentation	0..1	*NullableString*
NominalPressureStandard	0..1	*NominalPressureStandardClassification*
NominalPressureTypeRepresentation	0..1	*NullableString*
SubTagName	0..1	*NullableString*

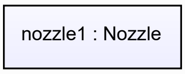
Attributes (reference)

Name	Multiplicity	Type
Chamber	0..1	*Chamber*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Nozzle>`
RDL reference: NOZZLE
ComponentClass: Nozzle
ComponentClassURI: http://data.posccaesar.org/rdl/RDS415214

Example



Example: Implementation in Proteus Schema

```
<Nozzle
    ID="nozzle1"
    ComponentClass="Nozzle"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS415214" ...>
  ...
</Nozzle>
```

7.107.2 Chamber

Attribute (reference)

The *Chamber* at which the *Nozzle* is located, if applicable. The Chamber must be a component of the same object as the Nozzle.

Multiplicity: 0..1

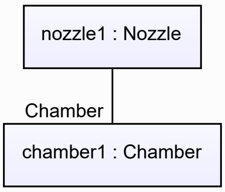
Type: *Chamber*

Opposite multiplicity: 0..*

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: `"is located in"`
Opposite association type: `"is the location of"`

Example



Example: Implementation in Proteus Schema

```
<Nozzle
    ID="nozzle1"
    ComponentClass="Nozzle"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS415214" ...>
  ...
  <Association
    Type="is located in"
    ItemID="chamber1" />
  ...
<Nozzle />
...
<Equipment
    ID="chamber1"
    ComponentClass="Chamber"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS903151421" ...>
  ...
  <Association
    Type="is the location of"
    ItemID="nozzle1" />
  ...
<Equipment />
```

7.107.3 NominalPressureNumericalValueRepresentation

Attribute (data)

A readable representation of the numerical value of the nominal pressure of the *Nozzle*, without any type or unit of measure.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: NOMINAL PRESSURE NUMERICAL VALUE REPRESENTATION ASSIGNMENT CLASS
Name: NominalPressureNumericalValueRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/NominalPressureNumericalValueRepresentationAssignmentClass

Example

“40” (*String*)

Example: Implementation in Proteus Schema

```
<Nozzle
    ID="nozzle1"
    ComponentClass="Nozzle"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS415214" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NominalPressureNumericalValueRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/NominalPressureNumericalValueRepresentationAssignmentClass"
      Format="string"
      Value="40" />
    ...
  </GenericAttributes>
  ...
</Nozzle>
```

7.107.4 NominalPressureRepresentation

Attribute (data)

A readable representation of the nominal pressure of the *Nozzle*. It normally contains a numerical value and a type or unit of measure.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: NOMINAL PRESSURE REPRESENTATION ASSIGNMENT CLASS
Name: NominalPressureRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/NominalPressureRepresentationAssignmentClass

Example

“PN 40” (*String*)

Example: Implementation in Proteus Schema

```
<Nozzle
    ID="nozzle1"
    ComponentClass="Nozzle"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS415214" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NominalPressureRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/NominalPressureRepresentationAssignmentClass"
      Format="string"
      Value="PN 40" />
    ...
  </GenericAttributes>
  ...
</Nozzle>
```

7.107.5 NominalPressureStandard

Attribute (data)

The nominal pressure of the *Nozzle*, given as a reference to a nominal pressure standard and value.

Multiplicity: 0..1

Type: *NominalPressureStandardClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: NOMINAL PRESSURE STANDARD SPECIALIZATION
Name: NominalPressureStandardSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/NominalPressureStandardSpecialization

Example

PN 40 (EN 1333) (*NominalPressureStandardClassification::En1333Pn40Artefact*)

Example: Implementation in Proteus Schema

```
<Nozzle
    ID="nozzle1"
    ComponentClass="Nozzle"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS415214" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NominalPressureStandardSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/NominalPressureStandardSpecialization"
      Format="anyURI"
      Value="En1333Pn40Artefact"
      ValueURI="http://sandbox.dexpi.org/rdl/En1333Pn40Artefact" />
    ...
  </GenericAttributes>
  ...
</Nozzle>
```

7.107.6 NominalPressureTypeRepresentation

Attribute (data)

A readable representation of the type or unit of measure of the nominal pressure of the *Nozzle*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: NOMINAL PRESSURE TYPE REPRESENTATION ASSIGNMENT CLASS
Name: NominalPressureTypeRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/NominalPressureTypeRepresentationAssignmentClass

Example

“PN” (*String*)

Example: Implementation in Proteus Schema

```
<Nozzle
    ID="nozzle1"
    ComponentClass="Nozzle"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS415214" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NominalPressureTypeRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/NominalPressureTypeRepresentationAssignmentClass"
      Format="string"
      Value="PN" />
    ...
  </GenericAttributes>
  ...
</Nozzle>
```

7.107.7 SubTagName

Attribute (data)

The sub tag name of the *Nozzle*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: SUB TAG NAME ASSIGNMENT CLASS
Name: SubTagNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/SubTagNameAssignmentClass

Example

“ST1” (*String*)

Example: Implementation in Proteus Schema

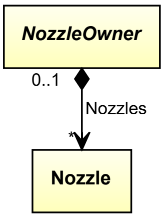
```
<Nozzle
    ID="nozzle1"
    ComponentClass="Nozzle"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS415214" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="SubTagNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/SubTagNameAssignmentClass"
      Format="string"
      Value="ST1" />
    ...
  </GenericAttributes>
  ...
</Nozzle>
```

7.108. NozzleOwner

7.108.1 Overview

Abstract class

An object that can have nozzles.



Subtypes

- *Equipment*

Attributes (composition)

Name	Multiplicity	Type
Nozzles	*	*Nozzle*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*. As *NozzleOwner* is abstract, there is no RDL reference for the class itself; the RDL reference depends on the concrete subclass.
Tag: `<Equipment>`
ComponentClass: *depending on subclass*
ComponentClassURI: *depending on subclass*

Example

As *NozzleOwner* is abstract, we consider *Vessel* as an arbitrary concrete subclass.

Example: Implementation in Proteus Schema

```
<Equipment
    ID="vessel1"
    ComponentClass="Vessel"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS414674" ...>
  ...
</Equipment>
```

7.108.2 Nozzles

Attribute (composition)

The nozzles of the *NozzleOwner*.

Multiplicity: *

Type: *Nozzle*

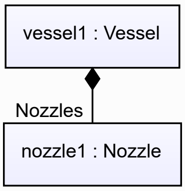
Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *Nozzle*) is a child of the `<Equipment>` element for the attribute owner (a *NozzleOwner*).

Example

As the owner type *NozzleOwner* is abstract, we consider *Vessel* as an arbitrary concrete subclass.



Example: Implementation in Proteus Schema

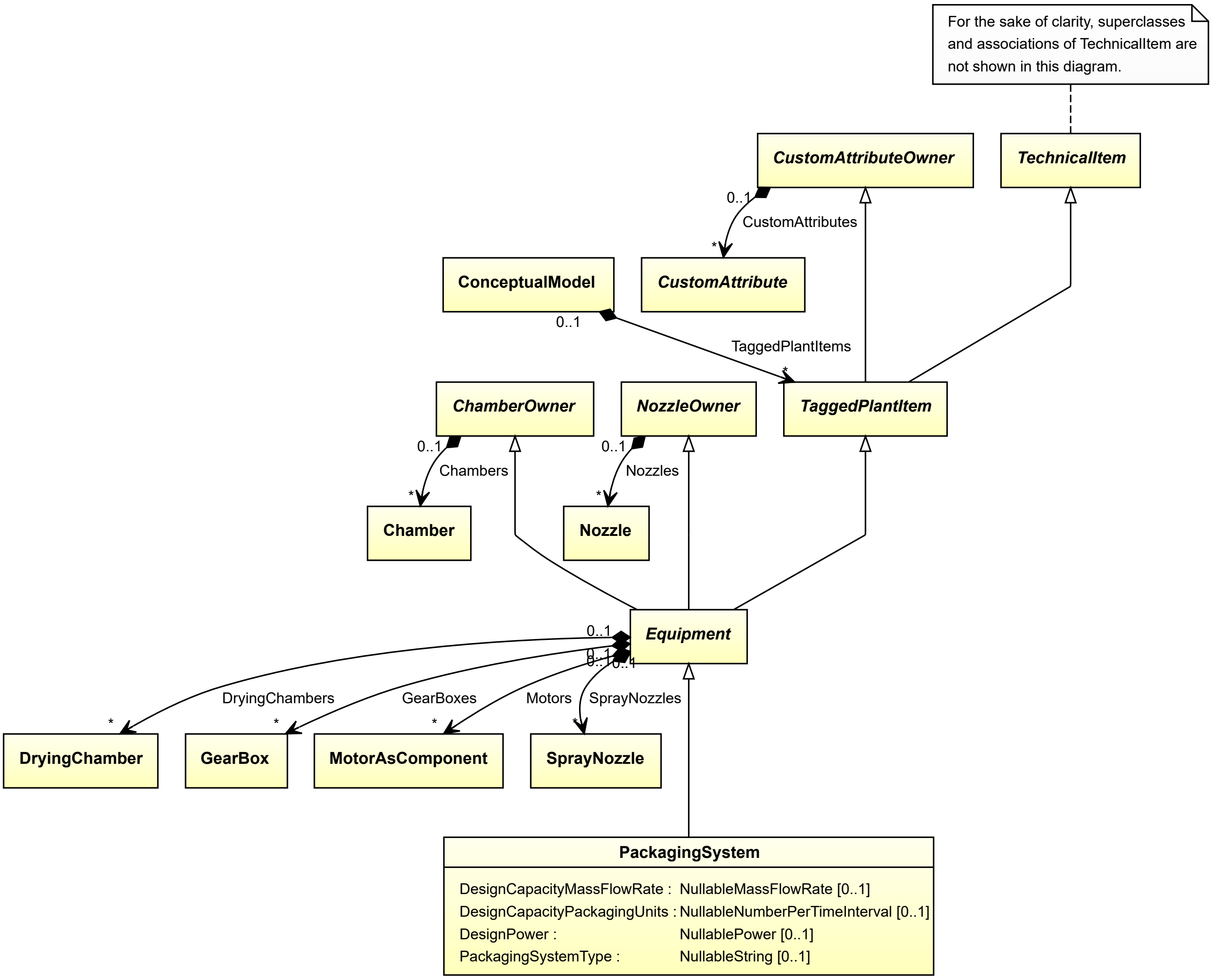
```
<Equipment
    ID="vessel1"
    ComponentClass="Vessel"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS414674" ...>
  ...
  <Nozzle
      ID="nozzle1"
      ComponentClass="Nozzle"
      ComponentClassURI="http://data.posccaesar.org/rdl/RDS415214" ...>
    ...
  <Nozzle />
  ...
<Equipment />
```

7.109. PackagingSystem

7.109.1 Overview

Class

A system that is intended for the preparation of goods for transport, warehousing, logistics, sale, and end use (from http://data.15926.org/rdl/RDS2228725).



Supertypes

- *Equipment*

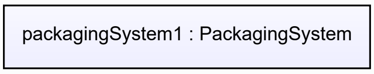
Attributes (data)

Name	Multiplicity	Type
DesignCapacityMassFlowRate	0..1	*NullableMassFlowRate*
DesignCapacityPackagingUnits	0..1	*NullableNumberPerTimeInterval*
DesignPower	0..1	*NullablePower*
PackagingSystemType	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: PACKAGING SYSTEM
ComponentClass: PackagingSystem
ComponentClassURI: http://sandbox.dexpi.org/rdl/PackagingSystem

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="packagingSystem1"
    ComponentClass="PackagingSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PackagingSystem" ...>
  ...
</Equipment>
```

7.109.2 DesignCapacityMassFlowRate

Attribute (data)

The capacity for the mass flow rate for which the *PackagingSystem* is designed.

Multiplicity: 0..1

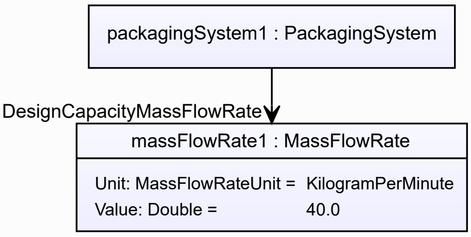
Type: *NullableMassFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN CAPACITY MASS FLOW RATE
Name: DesignCapacityMassFlowRate
AttributeURI: http://sandbox.dexpi.org/rdl/DesignCapacityMassFlowRate

Example

The instance packagingSystem1 represents a *PackagingSystem* with a *DesignCapacityMassFlowRate* of 40.0 kg/min.


packagingSystem1 : PackagingSystem
DesignCapacityMassFlowRate
massFlowRate1 : MassFlowRate
Unit: MassFlowRateUnit = KilogramPerMinute
Value: Double = 40.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="packagingSystem1"
    ComponentClass="PackagingSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PackagingSystem" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignCapacityMassFlowRate"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignCapacityMassFlowRate"
      Format="double"
      Value="40.0"
      Units="KilogramPerMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1350719" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.109.3 DesignCapacityPackagingUnits

Attribute (data)

The capacity for the number of packaging units per time for which the *PackagingSystem* is designed.

Multiplicity: 0..1

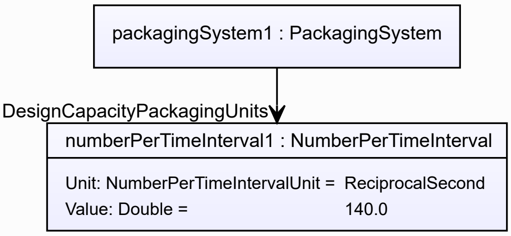
Type: *NullableNumberPerTimeInterval*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN CAPACITY PACKAGING UNITS
Name: DesignCapacityPackagingUnits
AttributeURI: http://sandbox.dexpi.org/rdl/DesignCapacityPackagingUnits

Example

The instance packagingSystem1 represents a *PackagingSystem* with a *DesignCapacityPackagingUnits* of 140.0 s^{-1}.


packagingSystem1 : PackagingSystem
DesignCapacityPackagingUnits
numberPerTimeInterval1 : NumberPerTimeInterval
Unit: NumberPerTimeIntervalUnit = ReciprocalSecond
Value: Double = 140.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="packagingSystem1"
    ComponentClass="PackagingSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PackagingSystem" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignCapacityPackagingUnits"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignCapacityPackagingUnits"
      Format="double"
      Value="140.0"
      Units="ReciprocalSecond"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1355489" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.109.4 DesignPower

Attribute (data)

The power for which the *PackagingSystem* is designed.

Multiplicity: 0..1

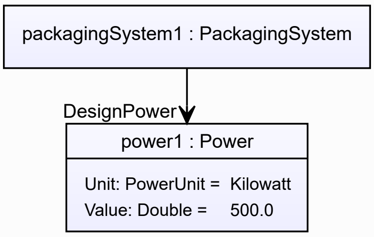
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN POWER
Name: DesignPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignPower

Example

The instance packagingSystem1 represents a *PackagingSystem* with a *DesignPower* of 500.0 kW.


packagingSystem1 : PackagingSystem
DesignPower
power1 : Power
Unit: PowerUnit = Kilowatt
Value: Double = 500.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="packagingSystem1"
    ComponentClass="PackagingSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PackagingSystem" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignPower"
      Format="double"
      Value="500.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.109.5 PackagingSystemType

Attribute (data)

The packaging system type of the *PackagingSystem*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PACKAGING SYSTEM TYPE ASSIGNMENT CLASS
Name: PackagingSystemTypeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PackagingSystemTypeAssignmentClass

Example

“Automated Packaging.” (*String*)

Example: Implementation in Proteus Schema

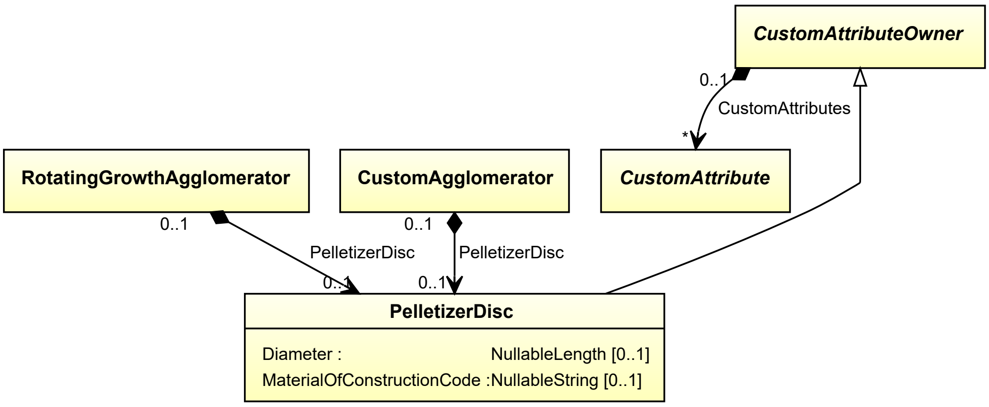
```
<Equipment
    ID="packagingSystem1"
    ComponentClass="PackagingSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PackagingSystem" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PackagingSystemTypeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PackagingSystemTypeAssignmentClass"
      Format="string"
      Value="Automated Packaging." />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.110. PelletizerDisc

7.110.1 Overview

Class

A rotating disc as a component of an *Agglomerator*.



Supertypes

- *CustomAttributeOwner*

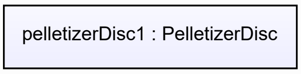
Attributes (data)

Name	Multiplicity	Type
Diameter	0..1	*NullableLength*
MaterialOfConstructionCode	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: PELLETING DISC
ComponentClass: PelletingDisc
ComponentClassURI: http://sandbox.dexpi.org/rdl/PelletingDisc

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="pelletizerDisc1"
    ComponentClass="PelletingDisc"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PelletingDisc" ...>
  ...
</Equipment>
```

7.110.2 Diameter

Attribute (data)

The diameter of the *PelletizerDisc*.

Multiplicity: 0..1

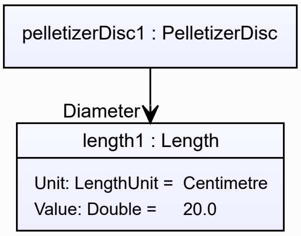
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DIAMETER
Name: Diameter
AttributeURI: http://data.posccaesar.org/rdl/RDS350954

Example

The instance pelletizerDisc1 represents a *PelletizerDisc* with a *Diameter* of 20.0 cm.


pelletizerDisc1 : PelletizerDisc
Diameter
length1 : Length
Unit: LengthUnit = Centimetre
Value: Double = 20.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="pelletizerDisc1"
    ComponentClass="PelletingDisc"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PelletingDisc" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="Diameter"
      AttributeURI="http://data.posccaesar.org/rdl/RDS350954"
      Format="double"
      Value="20.0"
      Units="Centimetre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1318004" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.110.3 MaterialOfConstructionCode

Attribute (data)

A code that gives the material of construction of the *PelletizerDisc*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: MATERIAL OF CONSTRUCTION CODE ASSIGNMENT CLASS
Name: MaterialOfConstructionCodeAssignmentClass
AttributeURI: http://data.posccaesar.org/rdl/RDS1460719741

Example

“1.4306” (*String*)

Example: Implementation in Proteus Schema

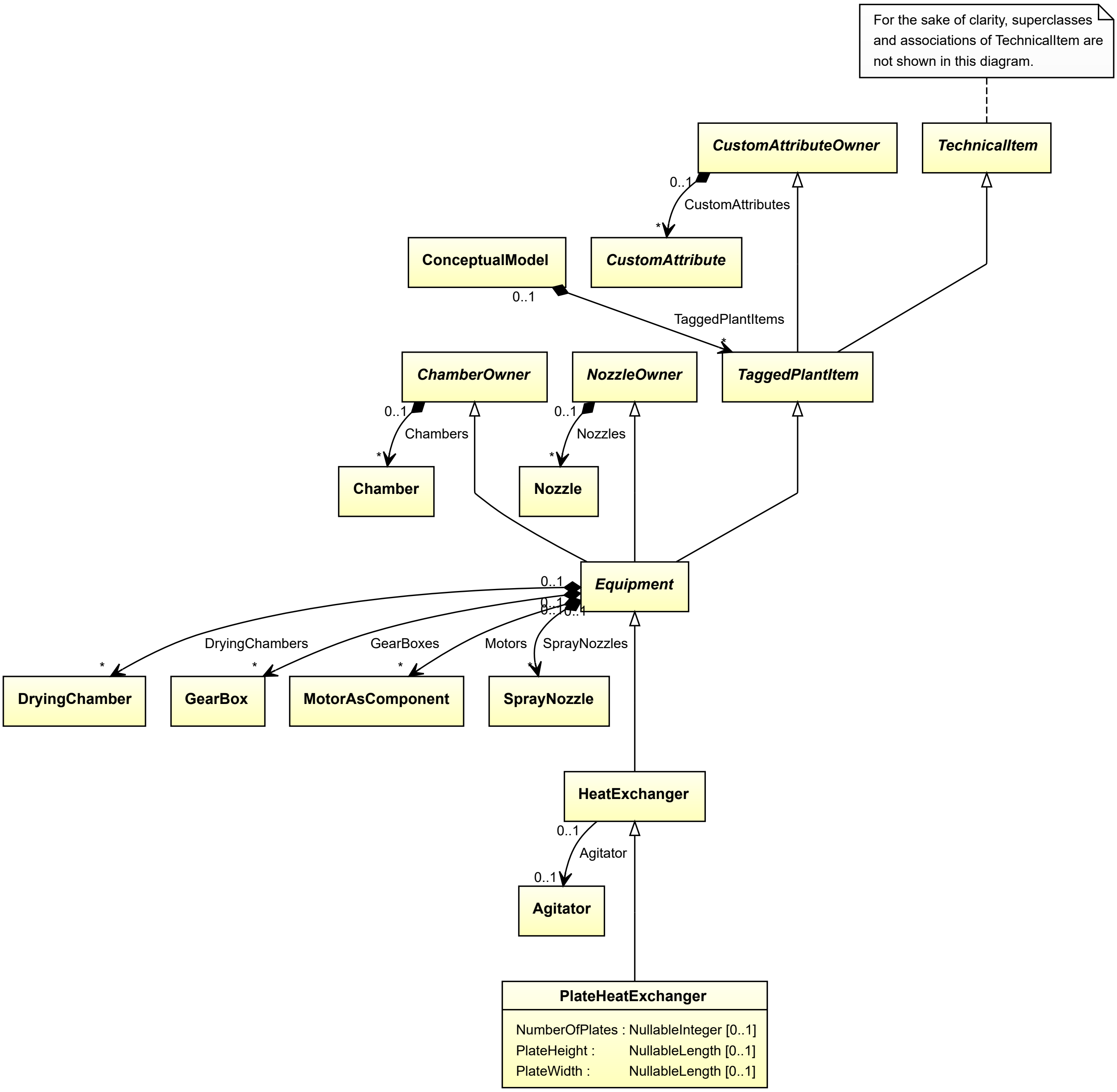
```
<Equipment
    ID="pelletizerDisc1"
    ComponentClass="PelletingDisc"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PelletingDisc" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="MaterialOfConstructionCodeAssignmentClass"
      AttributeURI="http://data.posccaesar.org/rdl/RDS1460719741"
      Format="string"
      Value="1.4306" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.111. PlateHeatExchanger

7.111.1 Overview

Class

A heat exchanger that uses metal plates to transfer heat between two fluids.



Supertypes

- *HeatExchanger*

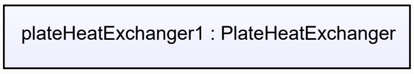
Attributes (data)

Name	Multiplicity	Type
NumberOfPlates	0..1	*NullableInteger*
PlateHeight	0..1	*NullableLength*
PlateWidth	0..1	*NullableLength*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: PLATE HEAT EXCHANGER
ComponentClass: PlateHeatExchanger
ComponentClassURI: http://sandbox.dexpi.org/rdl/PlateHeatExchanger

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="plateHeatExchanger1"
    ComponentClass="PlateHeatExchanger"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PlateHeatExchanger" ...>
  ...
</Equipment>
```

7.111.2 NumberOfPlates

Attribute (data)

The number of plates in the *PlateHeatExchanger*.

Multiplicity: 0..1

Type: *NullableInteger*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for integer values*.
RDL reference: NUMBER OF PLATES
Name: NumberOfPlates
AttributeURI: http://data.posccaesar.org/rdl/RDS364229

Example

20 (*Integer*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="plateHeatExchanger1"
    ComponentClass="PlateHeatExchanger"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PlateHeatExchanger" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NumberOfPlates"
      AttributeURI="http://data.posccaesar.org/rdl/RDS364229"
      Format="integer"
      Value="20" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.111.3 PlateHeight

Attribute (data)

The height of the plates in the *PlateHeatExchanger*.

Multiplicity: 0..1

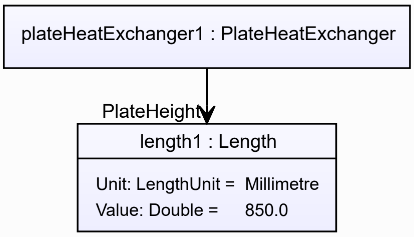
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: PLATE HEIGHT
Name: PlateHeight
AttributeURI: http://sandbox.dexpi.org/rdl/PlateHeight

Example

The instance plateHeatExchanger1 represents a *PlateHeatExchanger* with a *PlateHeight* of 850.0 mm.


plateHeatExchanger1 : PlateHeatExchanger
PlateHeight
length1 : Length
Unit: LengthUnit = Millimetre
Value: Double = 850.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="plateHeatExchanger1"
    ComponentClass="PlateHeatExchanger"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PlateHeatExchanger" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PlateHeight"
      AttributeURI="http://sandbox.dexpi.org/rdl/PlateHeight"
      Format="double"
      Value="850.0"
      Units="Millimetre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1357739" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.111.4 PlateWidth

Attribute (data)

The width of the plates in the *PlateHeatExchanger*.

Multiplicity: 0..1

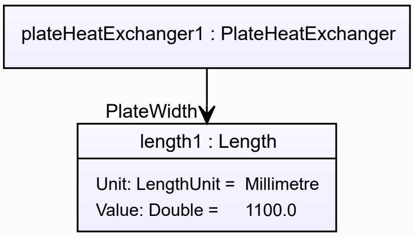
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: PLATE WIDTH
Name: PlateWidth
AttributeURI: http://sandbox.dexpi.org/rdl/PlateWidth

Example

The instance plateHeatExchanger1 represents a *PlateHeatExchanger* with a *PlateWidth* of 1100.0 mm.


plateHeatExchanger1 : PlateHeatExchanger
PlateWidth
length1 : Length
Unit: LengthUnit = Millimetre
Value: Double = 1100.0

Example: Implementation in Proteus Schema

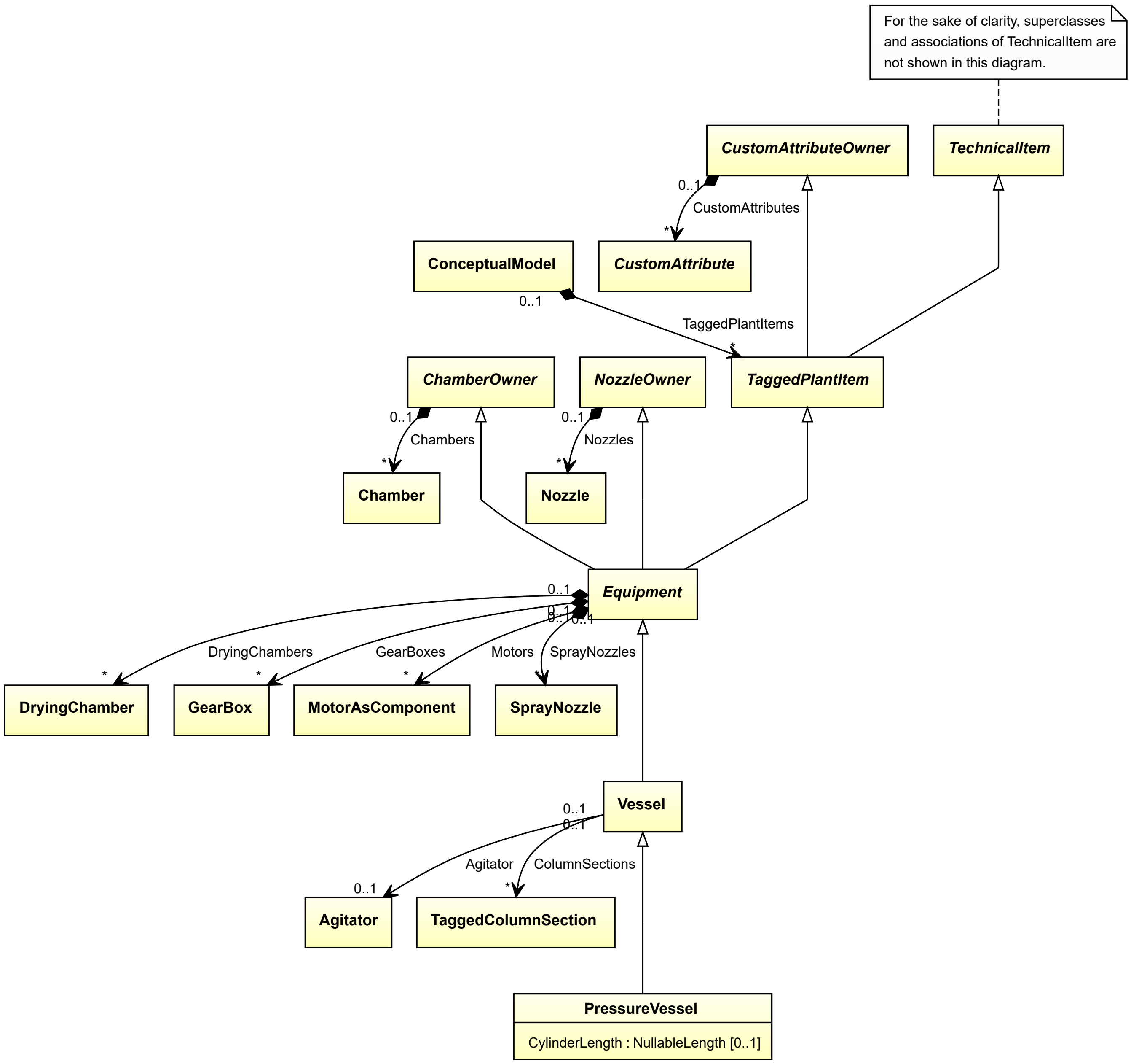
```
<Equipment
    ID="plateHeatExchanger1"
    ComponentClass="PlateHeatExchanger"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PlateHeatExchanger" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PlateWidth"
      AttributeURI="http://sandbox.dexpi.org/rdl/PlateWidth"
      Format="double"
      Value="1100.0"
      Units="Millimetre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1357739" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.112. PressureVessel

7.112.1 Overview

Class

A vessel intended to withstand external and/or internal pressure (from http://data.posccaesar.org/rdl/RDS427229).



Supertypes

- *Vessel*

Attributes (data)

Name	Multiplicity	Type
CylinderLength	0..1	*NullableLength*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: PRESSURE VESSEL

ComponentClass: PressureVessel
ComponentClassURI: http://data.posccaesar.org/rdl/RDS427229

Example

pressureVessel1 : PressureVessel

Example: Implementation in Proteus Schema

```
<Equipment
    ID="pressureVessel1"
    ComponentClass="PressureVessel"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS427229" ...>
  ...
</Equipment>
```

7.112.2 CylinderLength

Attribute (data)

The cylinder length of the *PressureVessel*.

Multiplicity: 0..1

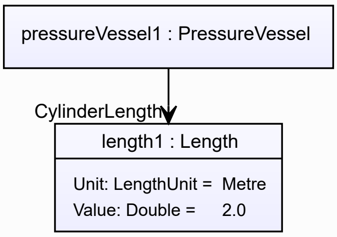
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: CYLINDER LENGTH
Name: CylinderLength
AttributeURI: http://sandbox.dexpi.org/rdl/CylinderLength

Example

The instance pressureVessel1 represents a *PressureVessel* with a *CylinderLength* of 2.0 m.



Example: Implementation in Proteus Schema

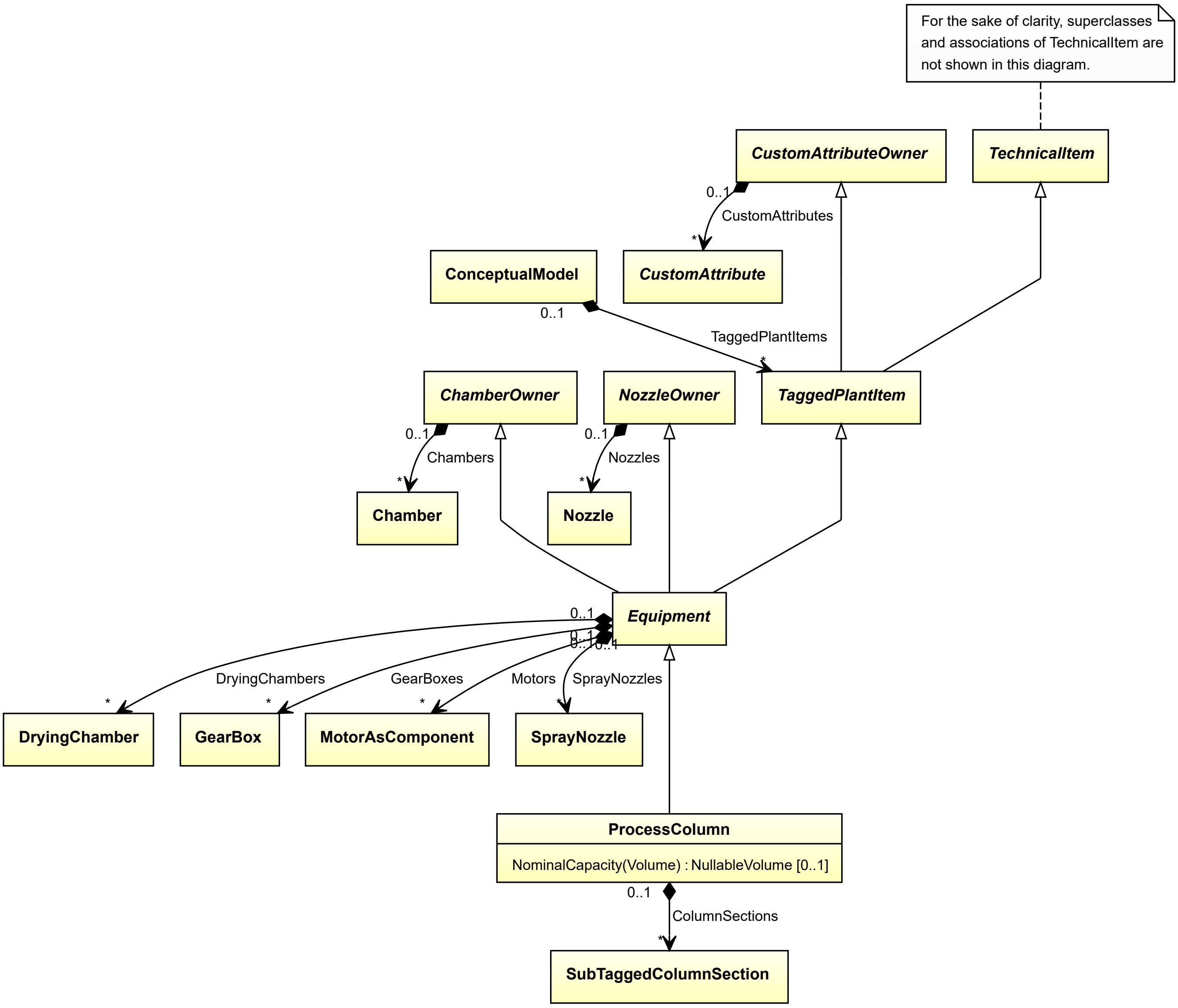
```
<Equipment
    ID="pressureVessel1"
    ComponentClass="PressureVessel"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS427229" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="CylinderLength"
      AttributeURI="http://sandbox.dexpi.org/rdl/CylinderLength"
      Format="double"
      Value="2.0"
      Units="Metre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1332674" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.113. ProcessColumn

7.113.1 Overview

Class

A vertical vessel intended to enable chemical reactions or physical processes utilising differences in density of fluids and/or forced flow of fluid (from http://data.posccaesar.org/rdl/RDS4316825224).


For the sake of clarity, superclasses and associations of TechnicalItem are not shown in this diagram.

Supertypes

- *Equipment*

Attributes (data)

Name	Multiplicity	Type
NominalCapacity(Volume)	0..1	*NullableVolume*

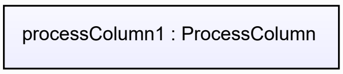
Attributes (composition)

Name	Multiplicity	Type
ColumnSections	*	*SubTaggedColumnSection*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: PROCESS COLUMN
ComponentClass: ProcessColumn
ComponentClassURI: http://data.posccaesar.org/rdl/RDS4316825224

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="processColumn1"
    ComponentClass="ProcessColumn"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS4316825224" ...>
  ...
</Equipment>
```

7.113.2 ColumnSections

Attribute (composition)

The column sections of the *ProcessColumn*.

Multiplicity: *

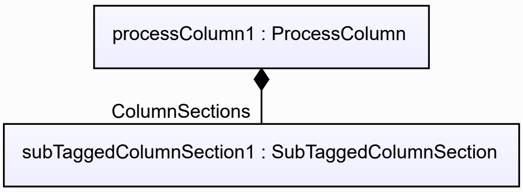
Type: *SubTaggedColumnSection*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *SubTaggedColumnSection*) is a child of the `<Equipment>` element for the attribute owner (a *ProcessColumn*).

Example


processColumn1 : ProcessColumn
ColumnSections
subTaggedColumnSection1 : SubTaggedColumnSection

Example: Implementation in Proteus Schema

```
<Equipment
    ID="processColumn1"
    ComponentClass="ProcessColumn"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS4316825224" ...>
  ...
  <Equipment
      ID="subTaggedColumnSection1"
      ComponentClass="ColumnSection"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/ColumnSection" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.113.3 NominalCapacity(Volume)

Attribute (data)

The nominal volumetric capacity of the *ProcessColumn*.

Multiplicity: 0..1

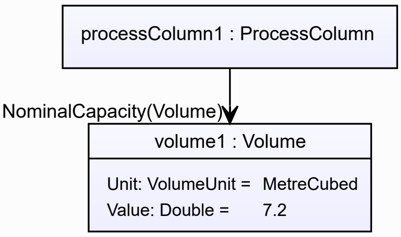
Type: *NullableVolume*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: NOMINAL CAPACITY VOLUME
Name: NominalCapacityVolume
AttributeURI: http://sandbox.dexpi.org/rdl/NominalCapacityVolume

Example

The instance processColumn1 represents a *ProcessColumn* with a *NominalCapacity(Volume)* of 7.2 m^3.



Example: Implementation in Proteus Schema

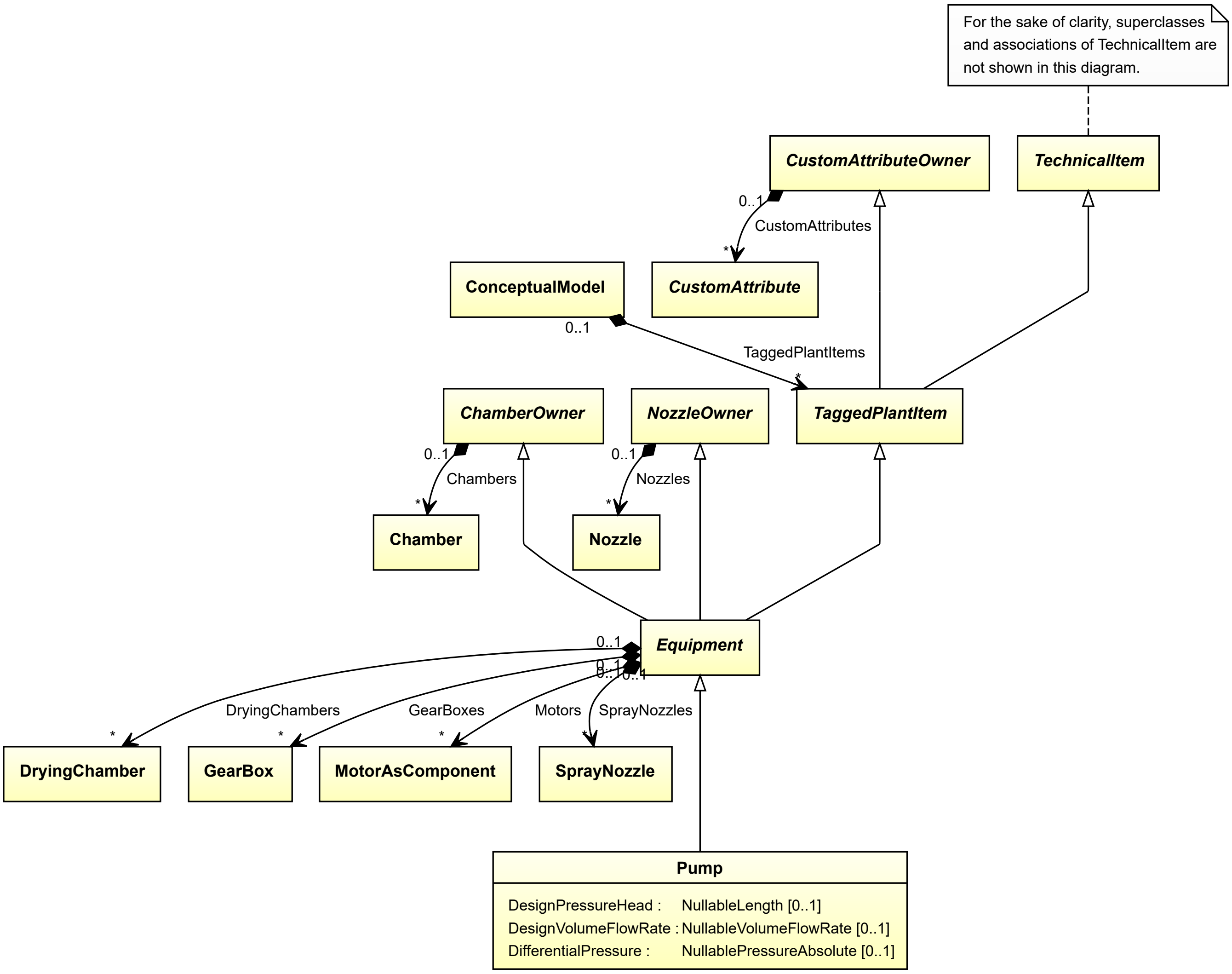
```
<Equipment
    ID="processColumn1"
    ComponentClass="ProcessColumn"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS4316825224" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NominalCapacityVolume"
      AttributeURI="http://sandbox.dexpi.org/rdl/NominalCapacityVolume"
      Format="double"
      Value="7.2"
      Units="MetreCubed"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1349099" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.114. Pump

7.114.1 Overview

Class

A machine that is capable of pumping but may require parts and subsystems for that capability.


For the sake of clarity, superclasses and associations of TechnicalItem are not shown in this diagram.

Supertypes

- *Equipment*

Subtypes

- *CentrifugalPump*
- *CustomPump*
- *EjectorPump*
- *ReciprocatingPump*
- *RotaryPump*

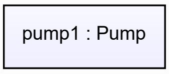
Attributes (data)

Name	Multiplicity	Type
DesignPressureHead	0..1	*NullableLength*
DesignVolumeFlowRate	0..1	*NullableVolumeFlowRate*
DifferentialPressure	0..1	*NullablePressureAbsolute*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: PUMP
ComponentClass: Pump
ComponentClassURI: http://data.posccaesar.org/rdl/RDS327239

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="pump1"
    ComponentClass="Pump"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS327239" ...>
  ...
</Equipment>
```

7.114.2 DesignPressureHead

Attribute (data)

The pressure head for which the *Pump* is designed.

Multiplicity: 0..1

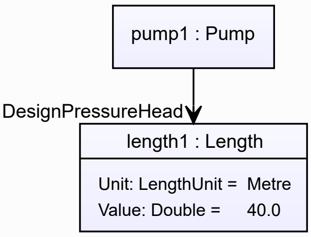
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN PRESSURE HEAD
Name: DesignPressureHead
AttributeURI: http://sandbox.dexpi.org/rdl/DesignPressureHead

Example

The instance pump1 represents a *Pump* with a *DesignPressureHead* of 40.0 m.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="pump1"
    ComponentClass="Pump"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS327239" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignPressureHead"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignPressureHead"
      Format="double"
      Value="40.0"
      Units="Metre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1332674" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.114.3 DesignVolumeFlowRate

Attribute (data)

The volume flow rate for which the *Pump* is designed.

Multiplicity: 0..1

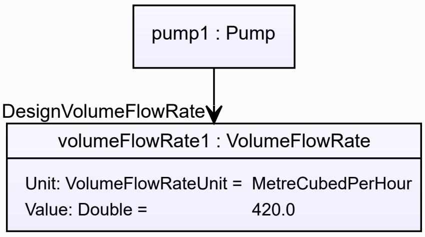
Type: *NullableVolumeFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN VOLUME FLOW RATE
Name: DesignVolumeFlowRate
AttributeURI: http://data.posccaesar.org/rdl/RDS14286227

Example

The instance pump1 represents a *Pump* with a *DesignVolumeFlowRate* of 420.0 m^3/h.


pump1 : Pump
DesignVolumeFlowRate
volumeFlowRate1 : VolumeFlowRate
Unit: VolumeFlowRateUnit = MetreCubedPerHour
Value: Double = 420.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="pump1"
    ComponentClass="Pump"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS327239" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignVolumeFlowRate"
      AttributeURI="http://data.posccaesar.org/rdl/RDS14286227"
      Format="double"
      Value="420.0"
      Units="MetreCubedPerHour"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1321064" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.114.4 DifferentialPressure

Attribute (data)

The differential pressure of the *Pump*.

Multiplicity: 0..1

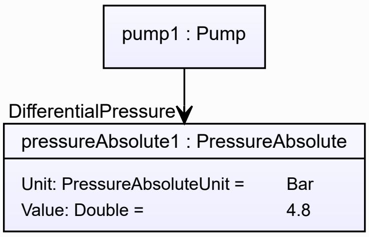
Type: *NullablePressureAbsolute*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DIFFERENTIAL PRESSURE
Name: DifferentialPressure
AttributeURI: http://data.posccaesar.org/rdl/RDS361574

Example

The instance pump1 represents a *Pump* with a *DifferentialPressure* of 4.8 bar.


pump1 : Pump
DifferentialPressure
pressureAbsolute1 : PressureAbsolute
Unit: PressureAbsoluteUnit =
Bar
Value: Double =
4.8

Example: Implementation in Proteus Schema

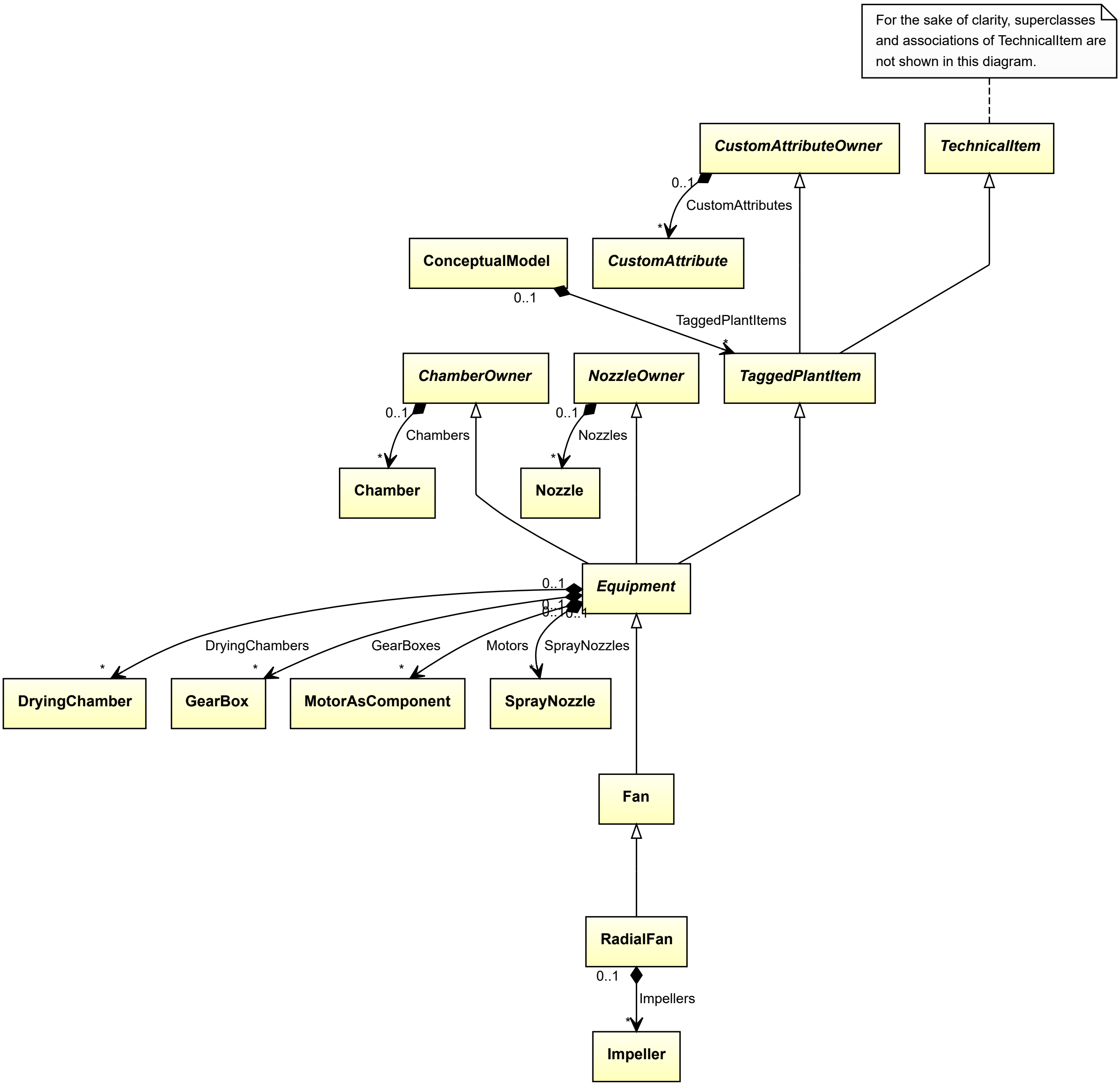
```
<Equipment
    ID="pump1"
    ComponentClass="Pump"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS327239" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DifferentialPressure"
      AttributeURI="http://data.posccaesar.org/rdl/RDS361574"
      Format="double"
      Value="4.8"
      Units="Bar"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1314539" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.115. RadialFan

7.115.1 Overview

Class

A ‘fan’ with axial inlet and radial outlet (from http://data.posccaesar.org/rdl/RDS414089).



Supertypes

- *Fan*

Attributes (composition)

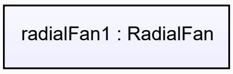
Name	Multiplicity	Type
Impellers	*	*Impeller*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`

RDL reference: RADIAL FAN
ComponentClass: RadialFan
ComponentClassURI: http://data.posccaesar.org/rdl/RDS414089

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="radialFan1"
    ComponentClass="RadialFan"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS414089" ...>
  ...
</Equipment>
```

7.115.2 Impellers

Attribute (composition)

The impellers of the *RadialFan*.

Multiplicity: *

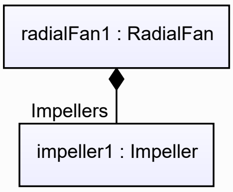
Type: *Impeller*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (an *Impeller*) is a child of the `<Equipment>` element for the attribute owner (a *RadialFan*).

Example



Example: Implementation in Proteus Schema

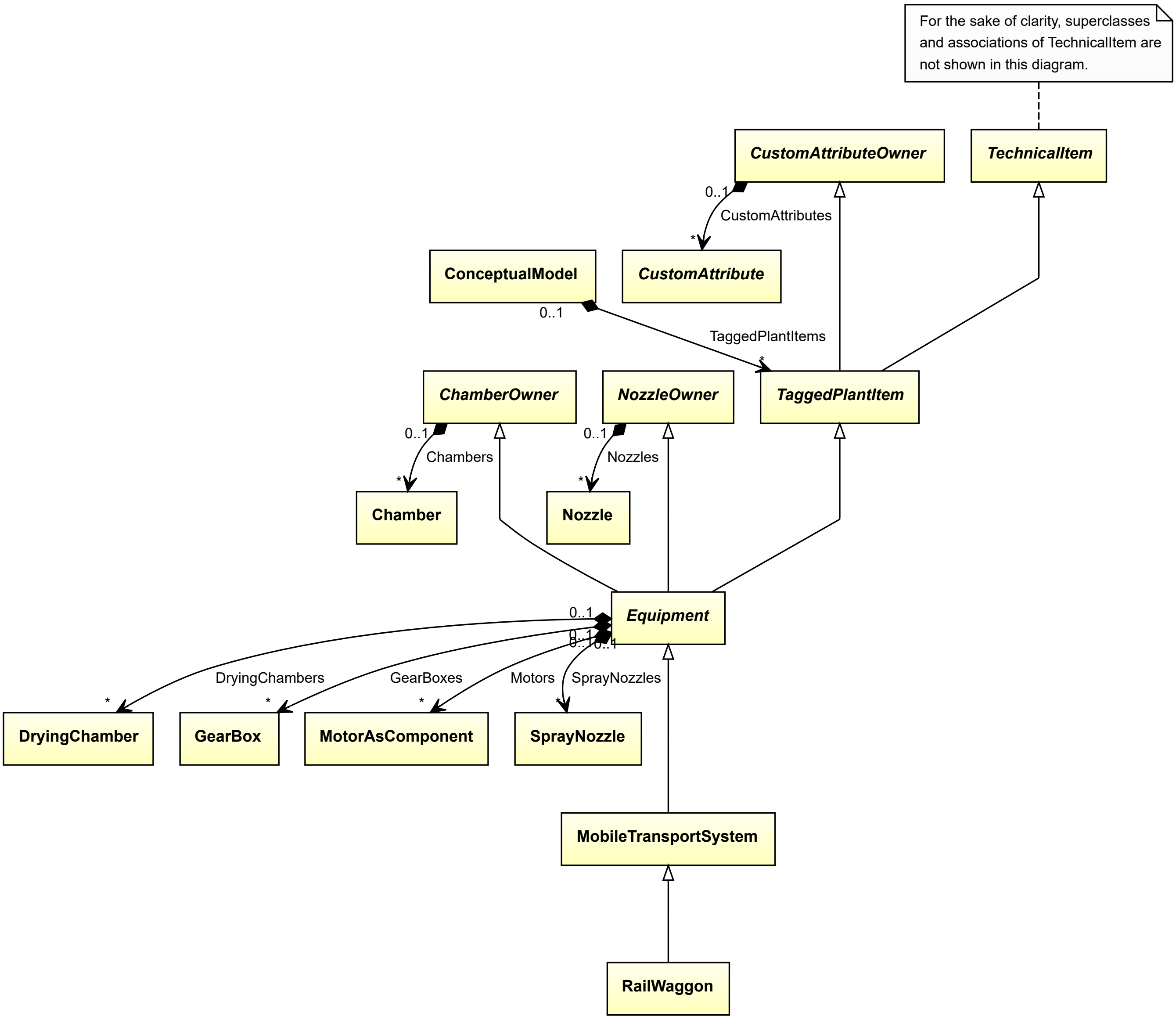
```
<Equipment
    ID="radialFan1"
    ComponentClass="RadialFan"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS414089" ...>
  ...
  <Equipment
      ID="impeller1"
      ComponentClass="Impeller"
      ComponentClassURI="http://data.posccaesar.org/rdl/RDS414539" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.116. RailWaggon

7.116.1 Overview

Class

A non self driving vehicle and mobile transport system intended to ride on rails



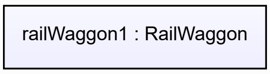
Supertypes

- *MobileTransportSystem*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: RAIL WAGGON
ComponentClass: RailWaggon
ComponentClassURI: http://data.posccaesar.org/rdl/RDS11524697

Example



Example: Implementation in Proteus Schema

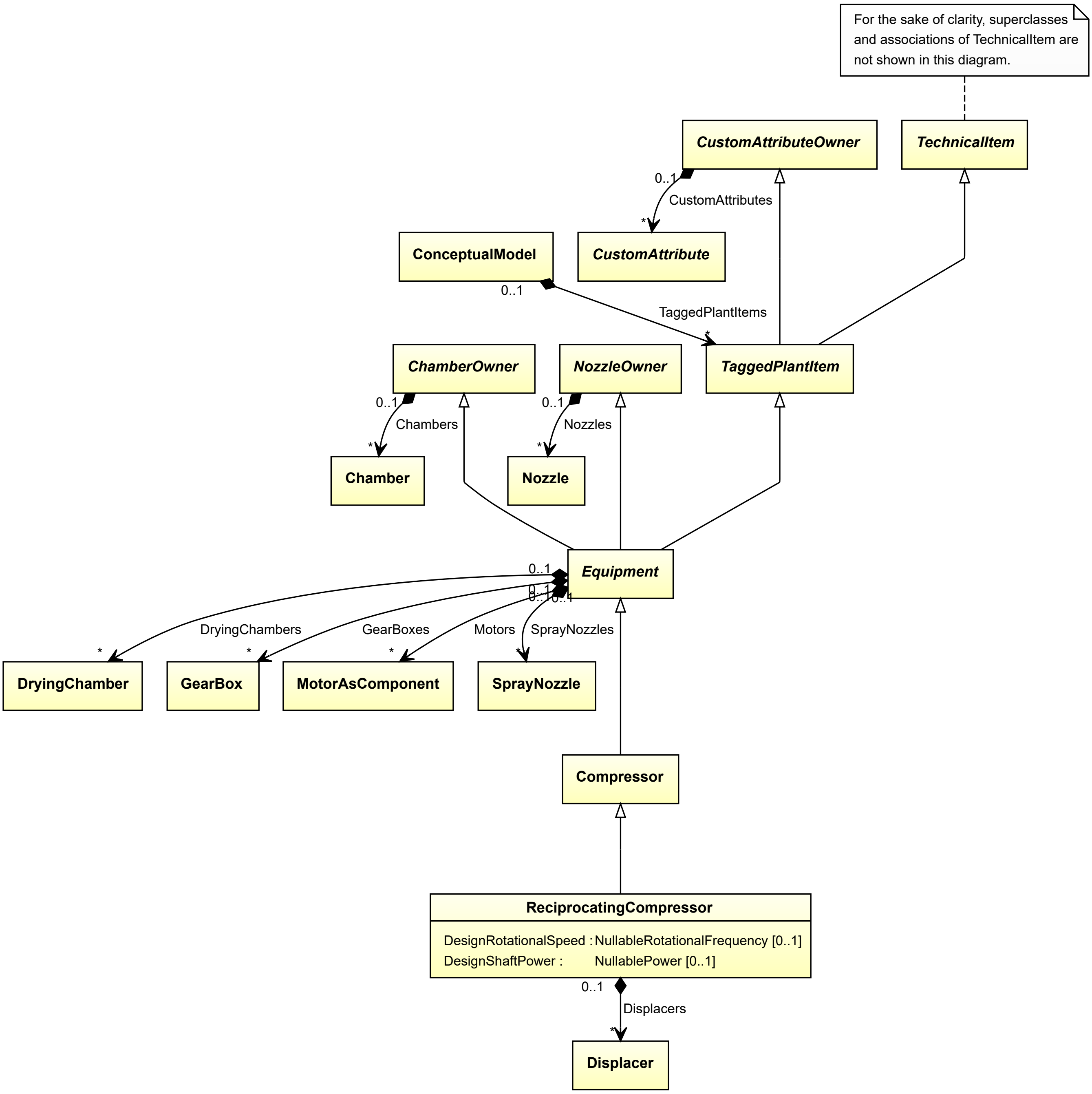
```
<Equipment
    ID="railWaggon1"
    ComponentClass="RailWaggon"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11524697" ...>
  ...
</Equipment>
```

7.117. ReciprocatingCompressor

7.117.1 Overview

Class

A positive displacement compressor in which forced reduction of gas volume takes place by the movement of a displacing element in a cylinder or enclosure (from http://data.posccaesar.org/rdl/RDS417284).



Supertypes

- *Compressor*

Attributes (data)

Name	Multiplicity	Type
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*
DesignShaftPower	0..1	*NullablePower*

Attributes (composition)

Name	Multiplicity	Type
Displacers	*	*Displacer*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: RECIPROCATING COMPRESSOR
ComponentClass: ReciprocatingCompressor
ComponentClassURI: http://data.posccaesar.org/rdl/RDS417284

Example

Example: Implementation in Proteus Schema

```
<Equipment
    ID="reciprocatingCompressor1"
    ComponentClass="ReciprocatingCompressor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS417284" ...>
  ...
</Equipment>
```

7.117.2 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *ReciprocatingCompressor* is designed.

Multiplicity: 0..1

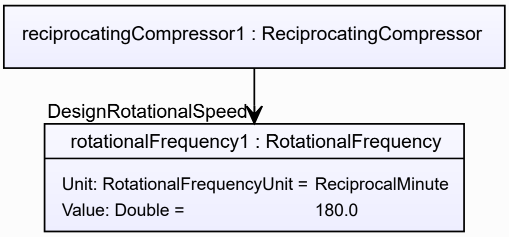
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

Example

The instance reciprocatingCompressor1 represents a *ReciprocatingCompressor* with a *DesignRotationalSpeed* of 180.0 min^{-1}.


reciprocatingCompressor1 : ReciprocatingCompressor
DesignRotationalSpeed
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 180.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="reciprocatingCompressor1"
    ComponentClass="ReciprocatingCompressor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS417284" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignRotationalSpeed"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignRotationalSpeed"
      Format="double"
      Value="180.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.117.3 DesignShaftPower

Attribute (data)

The shaft power for which the *ReciprocatingCompressor* is designed.

Multiplicity: 0..1

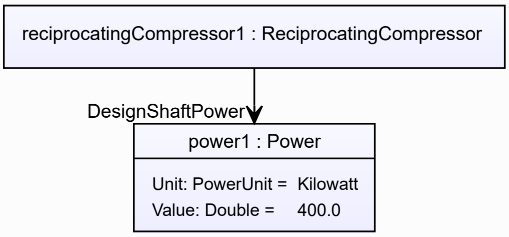
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SHAFT POWER
Name: DesignShaftPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignShaftPower

Example

The instance reciprocatingCompressor1 represents a *ReciprocatingCompressor* with a *DesignShaftPower* of 400.0 kW.


reciprocatingCompressor1 : ReciprocatingCompressor
DesignShaftPower
power1 : Power
Unit: PowerUnit = Kilowatt
Value: Double = 400.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="reciprocatingCompressor1"
    ComponentClass="ReciprocatingCompressor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS417284" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignShaftPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignShaftPower"
      Format="double"
      Value="400.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.117.4 Displacers

Attribute (composition)

The displacers of the *ReciprocatingCompressor*.

Multiplicity: *

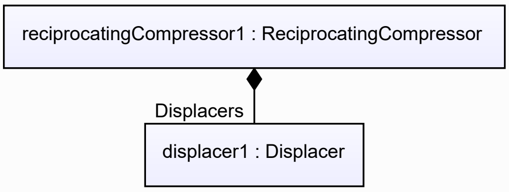
Type: *Displacer*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *Displacer*) is a child of the `<Equipment>` element for the attribute owner (a *ReciprocatingCompressor*).

Example



Example: Implementation in Proteus Schema

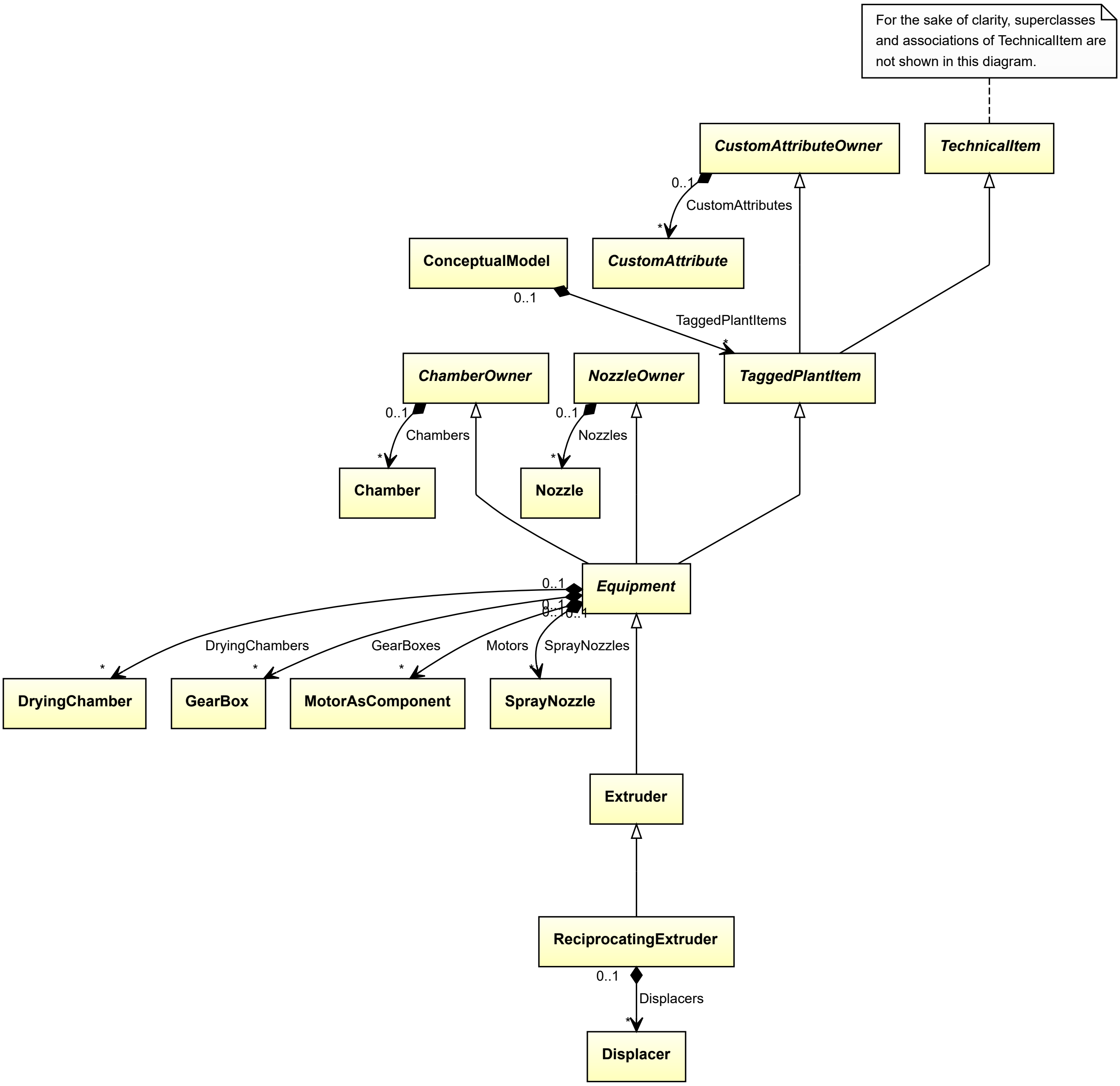
```
<Equipment
    ID="reciprocatingCompressor1"
    ComponentClass="ReciprocatingCompressor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS417284" ...>
  ...
  <Equipment
      ID="displacer1"
      ComponentClass="Displacer"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/Displacer" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.118. ReciprocatingExtruder

7.118.1 Overview

Class

An extruder that uses a piston in a batch process (from http://data.posccaesar.org/rdl/RDS412409911).



Supertypes

- *Extruder*

Attributes (composition)

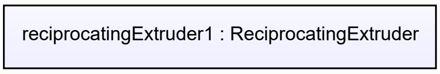
Name	Multiplicity	Type
Displacers	*	*Displacer*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`

RDL reference: PISTON EXTRUDER
ComponentClass: PistonExtruder
ComponentClassURI: http://data.posccaesar.org/rdl/RDS412409911

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="reciprocatingExtruder1"
    ComponentClass="PistonExtruder"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS412409911" ...>
  ...
</Equipment>
```

7.118.2 Displacers

Attribute (composition)

The displacers of the *ReciprocatingExtruder*.

Multiplicity: *

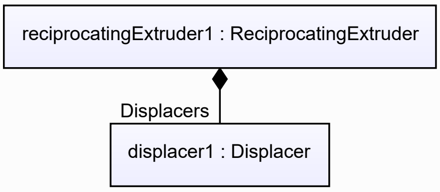
Type: *Displacer*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *Displacer*) is a child of the `<Equipment>` element for the attribute owner (a *ReciprocatingExtruder*).

Example



Example: Implementation in Proteus Schema

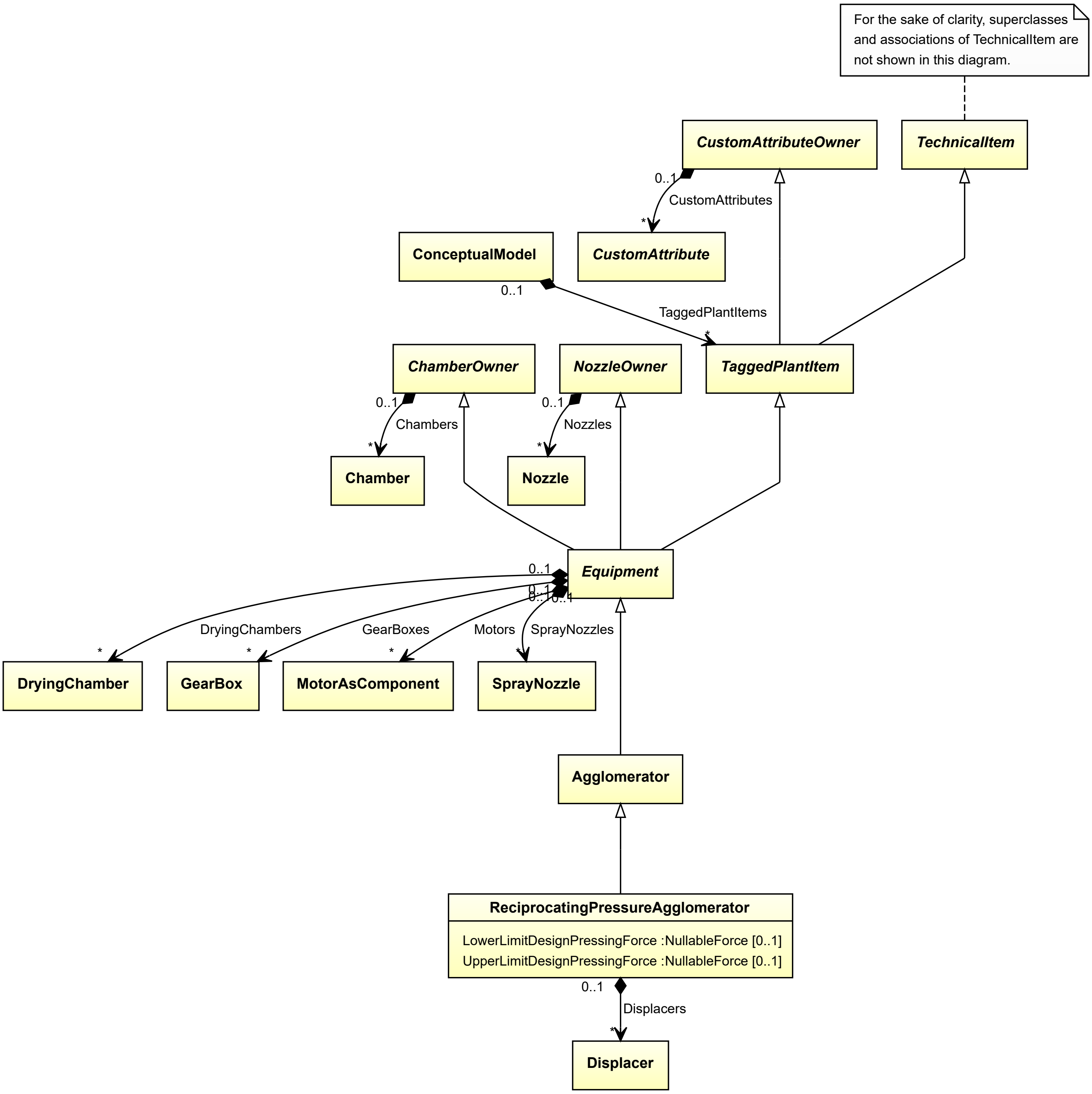
```
<Equipment
    ID="reciprocatingExtruder1"
    ComponentClass="PistonExtruder"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS412409911" ...>
  ...
  <Equipment
      ID="displacer1"
      ComponentClass="Displacer"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/Displacer" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.119. ReciprocatingPressureAgglomerator

7.119.1 Overview

Class

An *Agglomerator* which uses pistons to produce pressure and to form material (from http://data.15926.org/rdl/RDS2228720).



Supertypes

- *Agglomerator*

Attributes (data)

Name	Multiplicity	Type
LowerLimitDesignPressingForce	0..1	*NullableForce*
UpperLimitDesignPressingForce	0..1	*NullableForce*

Attributes (composition)

Name	Multiplicity	Type
Displacers	*	*Displacer*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: RECIPROCATING PRESSURE AGGLOMERATOR
ComponentClass: ReciprocatingPressureAgglomerator
ComponentClassURI: http://sandbox.dexpi.org/rdl/ReciprocatingPressureAgglomerator

Example

reciprocatingPressureAgglomerator1 : ReciprocatingPressureAgglomerator

Example: Implementation in Proteus Schema

```
<Equipment
    ID="reciprocatingPressureAgglomerator1"
    ComponentClass="ReciprocatingPressureAgglomerator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ReciprocatingPressureAgglomerator" ...>
  ...
</Equipment>
```

7.119.2 Displacers

Attribute (composition)

The displacers of the *ReciprocatingPressureAgglomerator*.

Multiplicity: *

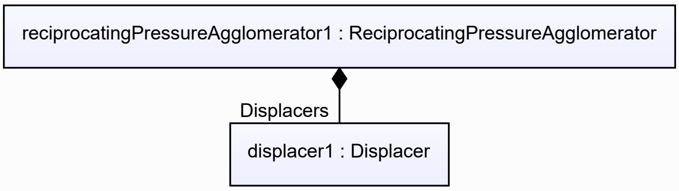
Type: *Displacer*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *Displacer*) is a child of the <Equipment> element for the attribute owner (a *ReciprocatingPressureAgglomerator*).

Example


reciprocatingPressureAgglomerator1 : ReciprocatingPressureAgglomerator
Displacers
displacer1 : Displacer

Example: Implementation in Proteus Schema

```
<Equipment
    ID="reciprocatingPressureAgglomerator1"
    ComponentClass="ReciprocatingPressureAgglomerator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ReciprocatingPressureAgglomerator" ...>
  ...
  <Equipment
      ID="displacer1"
      ComponentClass="Displacer"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/Displacer" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.119.3 LowerLimitDesignPressingForce

Attribute (data)

The lower limit for the pressing force for which the *ReciprocatingPressureAgglomerator* is designed.

Multiplicity: 0..1

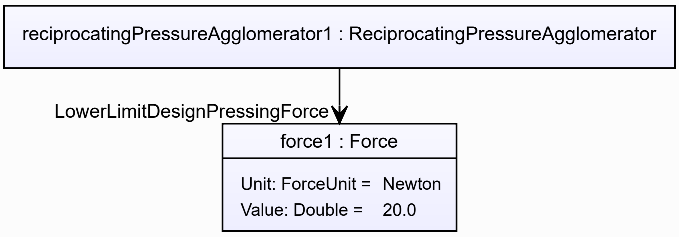
Type: *NullableForce*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: LOWER LIMIT DESIGN PRESSING FORCE
Name: LowerLimitDesignPressingForce
AttributeURI: http://sandbox.dexpi.org/rdl/LowerLimitDesignPressingForce

Example

The instance reciprocatingPressureAgglomerator1 represents a *ReciprocatingPressureAgglomerator* with a *LowerLimitDesignPressingForce* of 20.0 N.


reciprocatingPressureAgglomerator1 : ReciprocatingPressureAgglomerator
LowerLimitDesignPressingForce
force1 : Force
Unit: ForceUnit = Newton
Value: Double = 20.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="reciprocatingPressureAgglomerator1"
    ComponentClass="ReciprocatingPressureAgglomerator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ReciprocatingPressureAgglomerator" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="LowerLimitDesignPressingForce"
      AttributeURI="http://sandbox.dexpi.org/rdl/LowerLimitDesignPressingForce"
      Format="double"
      Value="20.0"
      Units="Newton"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1337939" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.119.4 UpperLimitDesignPressingForce

Attribute (data)

The upper limit for the pressing force for which the *ReciprocatingPressureAgglomerator* is designed.

Multiplicity: 0..1

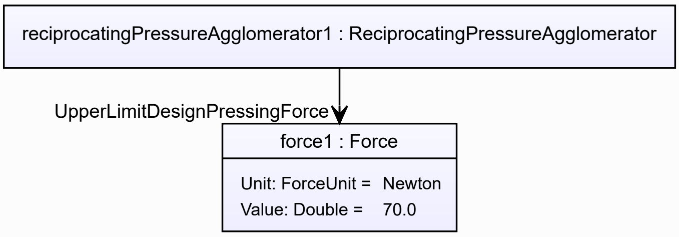
Type: *NullableForce*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT DESIGN PRESSING FORCE
Name: UpperLimitDesignPressingForce
AttributeURI: http://sandbox.dexpi.org/rdl/UpperLimitDesignPressingForce

Example

The instance reciprocatingPressureAgglomerator1 represents a *ReciprocatingPressureAgglomerator* with an *UpperLimitDesignPressingForce* of 70.0 N.


reciprocatingPressureAgglomerator1 : ReciprocatingPressureAgglomerator
UpperLimitDesignPressingForce
force1 : Force
Unit: ForceUnit = Newton
Value: Double = 70.0

Example: Implementation in Proteus Schema

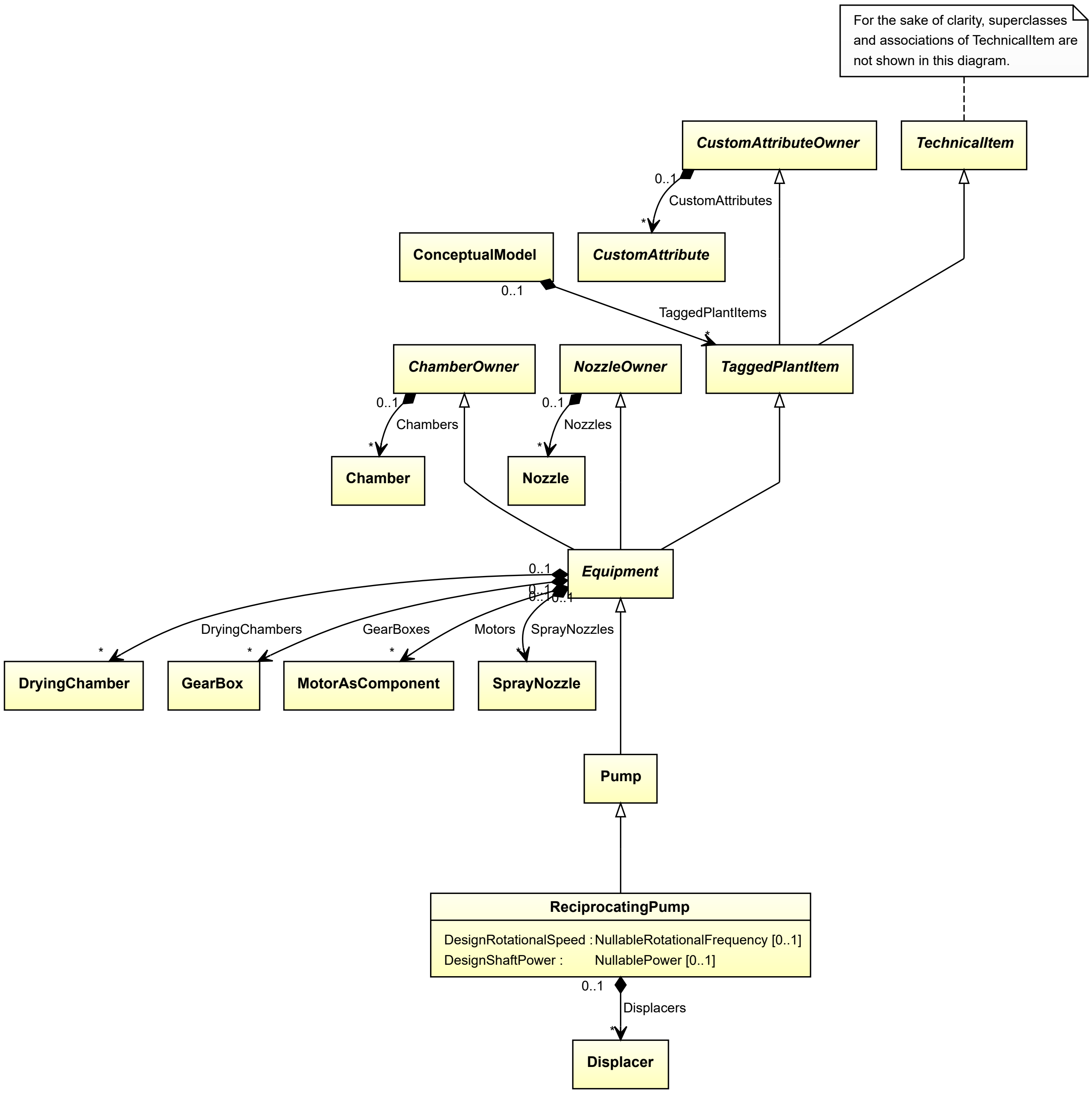
```
<Equipment
    ID="reciprocatingPressureAgglomerator1"
    ComponentClass="ReciprocatingPressureAgglomerator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ReciprocatingPressureAgglomerator" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="UpperLimitDesignPressingForce"
      AttributeURI="http://sandbox.dexpi.org/rdl/UpperLimitDesignPressingForce"
      Format="double"
      Value="70.0"
      Units="Newton"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1337939" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.120. ReciprocatingPump

7.120.1 Overview

Class

A positive displacement pump which contains a displacing element intended to be moved in a reciprocating movement to exert pressure on a fluid, typically moving within a cylindrical space (from http://data.posccaesar.org/rdl/RDS416969).



Supertypes

- *Pump*

Attributes (data)

Name	Multiplicity	Type
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*
DesignShaftPower	0..1	*NullablePower*

Attributes (composition)

Name	Multiplicity	Type
Displacers	*	*Displacer*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: RECIPROCATING PUMP
ComponentClass: ReciprocatingPump
ComponentClassURI: http://data.posccaesar.org/rdl/RDS416969

Example

reciprocatingPump1 : ReciprocatingPump

Example: Implementation in Proteus Schema

```
<Equipment
    ID="reciprocatingPump1"
    ComponentClass="ReciprocatingPump"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS416969" ...>
  ...
</Equipment>
```

7.120.2 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *ReciprocatingPump* is designed.

Multiplicity: 0..1

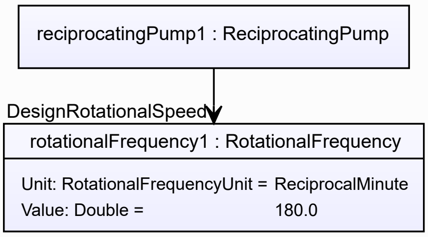
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

Example

The instance reciprocatingPump1 represents a *ReciprocatingPump* with a *DesignRotationalSpeed* of 180.0 min^{-1}.


reciprocatingPump1 : ReciprocatingPump
DesignRotationalSpeed
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 180.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="reciprocatingPump1"
    ComponentClass="ReciprocatingPump"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS416969" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignRotationalSpeed"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignRotationalSpeed"
      Format="double"
      Value="180.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.120.3 DesignShaftPower

Attribute (data)

The shaft power for which the *ReciprocatingPump* is designed.

Multiplicity: 0..1

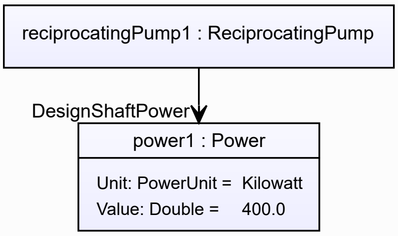
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SHAFT POWER
Name: DesignShaftPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignShaftPower

Example

The instance reciprocatingPump1 represents a *ReciprocatingPump* with a *DesignShaftPower* of 400.0 kW.


reciprocatingPump1 : ReciprocatingPump
DesignShaftPower
power1 : Power
Unit: PowerUnit = Kilowatt
Value: Double = 400.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="reciprocatingPump1"
    ComponentClass="ReciprocatingPump"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS416969" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignShaftPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignShaftPower"
      Format="double"
      Value="400.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.120.4 Displacers

Attribute (composition)

The displacers of the *ReciprocatingPump*.

Multiplicity: *

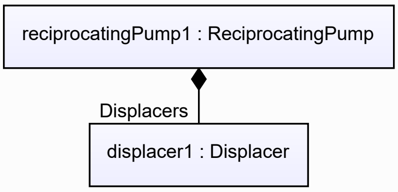
Type: *Displacer*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *Displacer*) is a child of the `<Equipment>` element for the attribute owner (a *ReciprocatingPump*).

Example



Example: Implementation in Proteus Schema

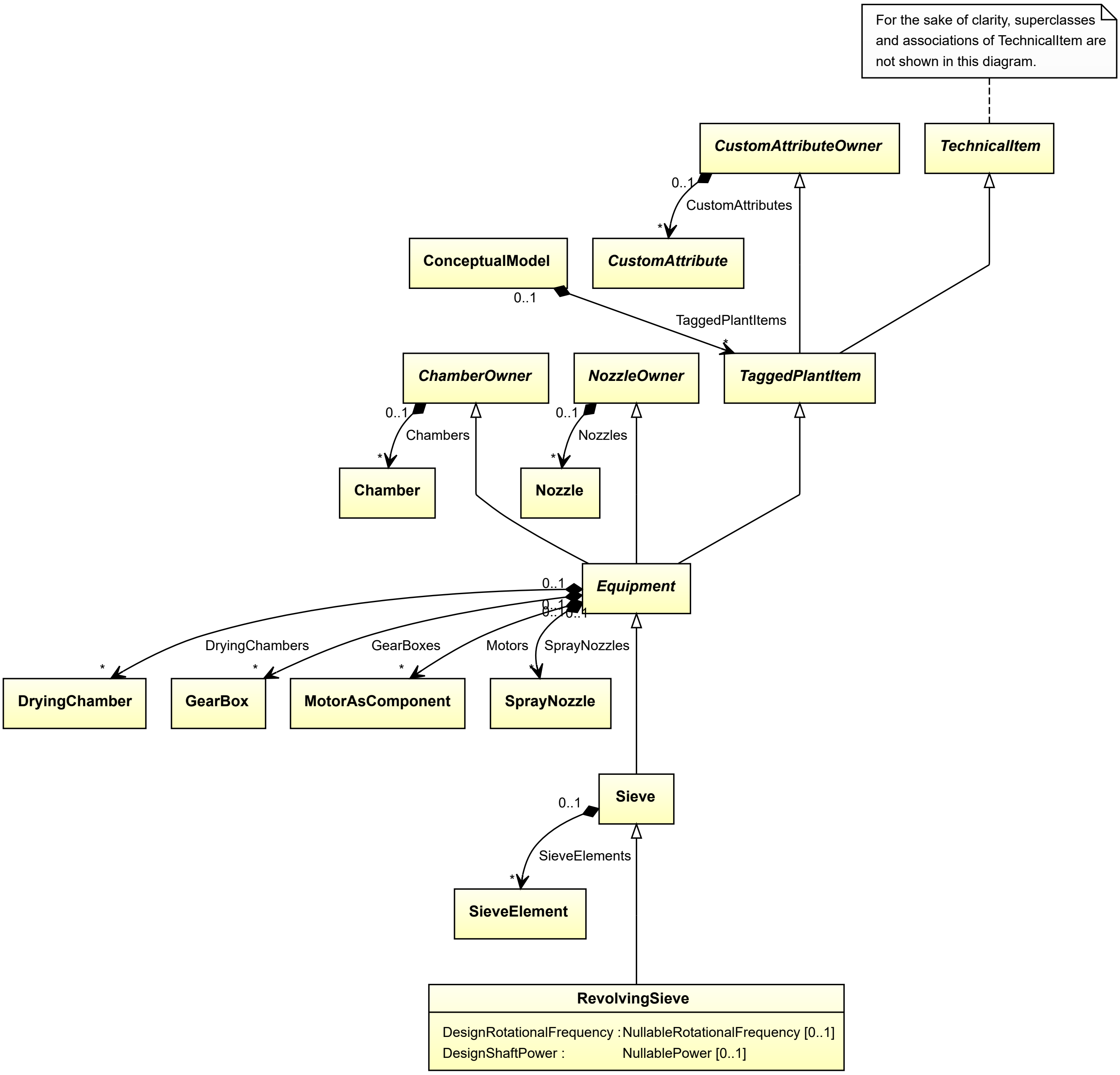
```
<Equipment
    ID="reciprocatingPump1"
    ComponentClass="ReciprocatingPump"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS416969" ...>
  ...
  <Equipment
      ID="displacer1"
      ComponentClass="Displacer"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/Displacer" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.121. RevolvingSieve

7.121.1 Overview

Class

A revolving sieve that intends to sift out finer from coarser parts.



Supertypes

- *Sieve*

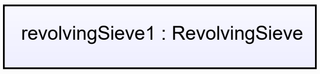
Attributes (data)

Name	Multiplicity	Type
DesignRotationalFrequency	0..1	*NullableRotationalFrequency*
DesignShaftPower	0..1	*NullablePower*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: REVOLVING SIEVE
ComponentClass: RevolvingSieve
ComponentClassURI: http://sandbox.dexpi.org/rdl/RevolvingSieve

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="revolvingSieve1"
    ComponentClass="RevolvingSieve"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/RevolvingSieve" ...>
  ...
</Equipment>
```

7.121.2 DesignRotationalFrequency

Attribute (data)

The rotational frequency for which the *RevolvingSieve* is designed.

Multiplicity: 0..1

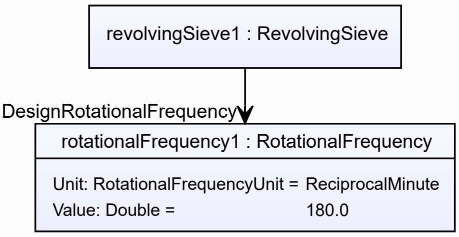
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL FREQUENCY
Name: DesignRotationalFrequency
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalFrequency

Example

The instance revolvingSieve1 represents a *RevolvingSieve* with a *DesignRotationalFrequency* of 180.0 min^{-1}.


revolvingSieve1 : RevolvingSieve
DesignRotationalFrequency
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 180.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="revolvingSieve1"
    ComponentClass="RevolvingSieve"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/RevolvingSieve" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignRotationalFrequency"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignRotationalFrequency"
      Format="double"
      Value="180.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.121.3 DesignShaftPower

Attribute (data)

The shaft power for which the *RevolvingSieve* is designed.

Multiplicity: 0..1

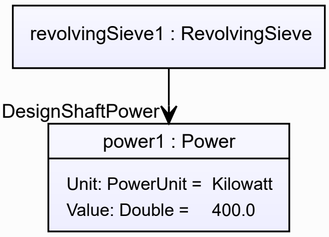
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SHAFT POWER
Name: DesignShaftPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignShaftPower

Example

The instance revolvingSieve1 represents a *RevolvingSieve* with a *DesignShaftPower* of 400.0 kW.



Example: Implementation in Proteus Schema

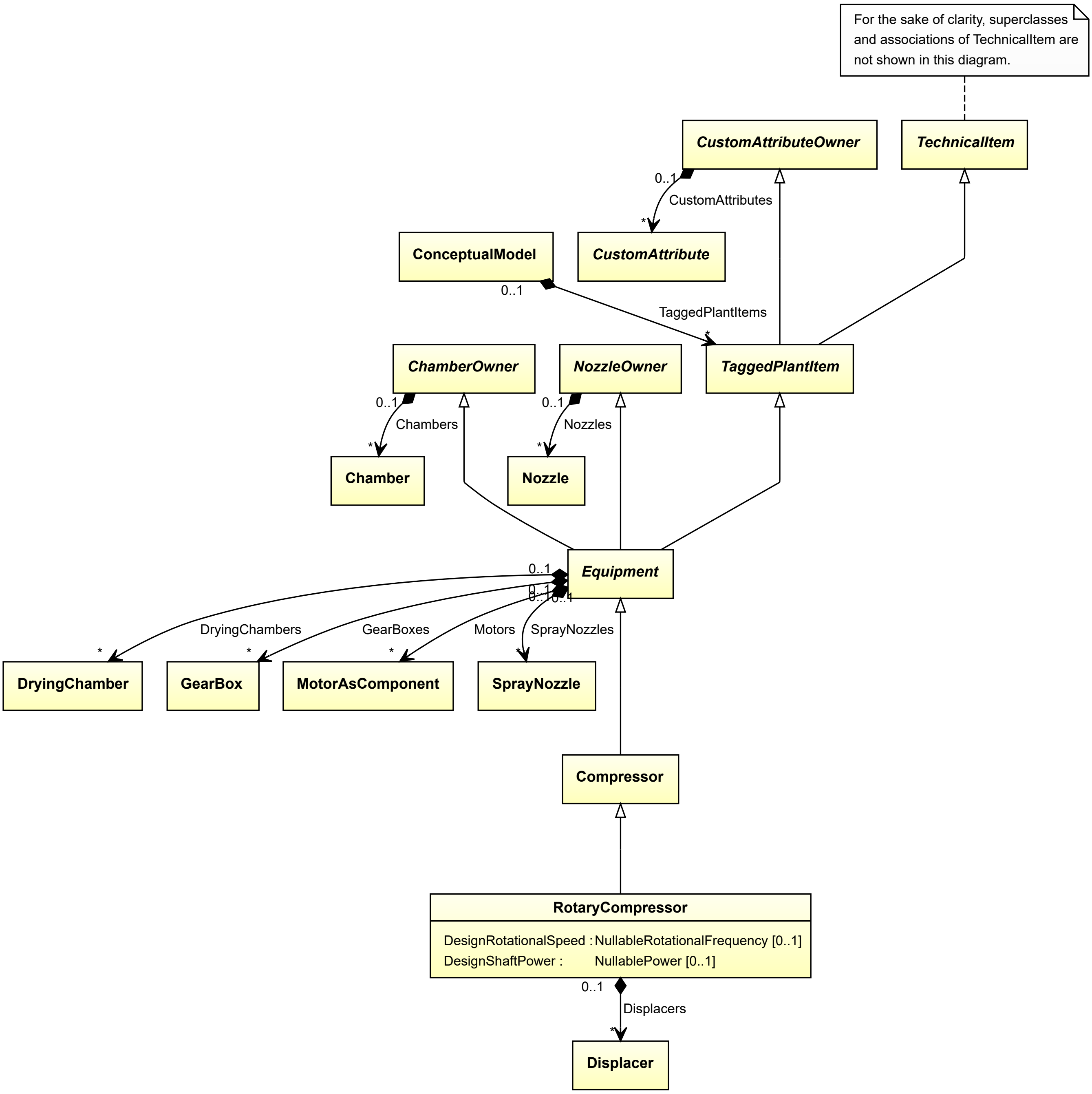
```
<Equipment
    ID="revolvingSieve1"
    ComponentClass="RevolvingSieve"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/RevolvingSieve" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignShaftPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignShaftPower"
      Format="double"
      Value="400.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.122. RotaryCompressor

7.122.1 Overview

Class

A positive displacement compressor in which compression displacement is effected by the positive action of rotating elements (from http://data.posccaesar.org/rdl/RDS435374).



Supertypes

- *Compressor*

Attributes (data)

Name	Multiplicity	Type
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*
DesignShaftPower	0..1	*NullablePower*

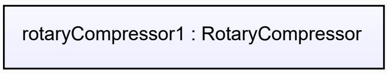
Attributes (composition)

Name	Multiplicity	Type
Displacers	*	*Displacer*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: ROTARY COMPRESSOR
ComponentClass: RotaryCompressor
ComponentClassURI: http://data.posccaesar.org/rdl/RDS435374

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="rotaryCompressor1"
    ComponentClass="RotaryCompressor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS435374" ...>
  ...
</Equipment>
```

7.122.2 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *RotaryCompressor* is designed.

Multiplicity: 0..1

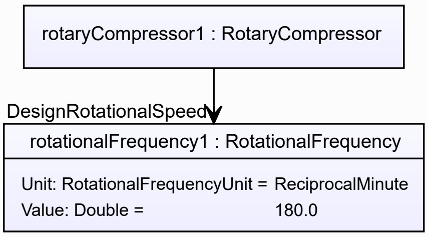
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

Example

The instance rotaryCompressor1 represents a *RotaryCompressor* with a *DesignRotationalSpeed* of 180.0 min^{-1}.


rotaryCompressor1 : RotaryCompressor
DesignRotationalSpeed
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 180.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="rotaryCompressor1"
    ComponentClass="RotaryCompressor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS435374" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignRotationalSpeed"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignRotationalSpeed"
      Format="double"
      Value="180.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.122.3 DesignShaftPower

Attribute (data)

The shaft power for which the *RotaryCompressor* is designed.

Multiplicity: 0..1

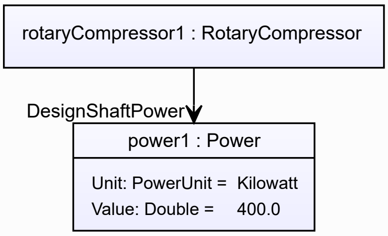
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SHAFT POWER
Name: DesignShaftPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignShaftPower

Example

The instance rotaryCompressor1 represents a *RotaryCompressor* with a *DesignShaftPower* of 400.0 kW.


rotaryCompressor1 : RotaryCompressor
DesignShaftPower
power1 : Power
Unit: PowerUnit = Kilowatt
Value: Double = 400.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="rotaryCompressor1"
    ComponentClass="RotaryCompressor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS435374" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignShaftPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignShaftPower"
      Format="double"
      Value="400.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.122.4 Displacers

Attribute (composition)

The displacers of the *RotaryCompressor*.

Multiplicity: *

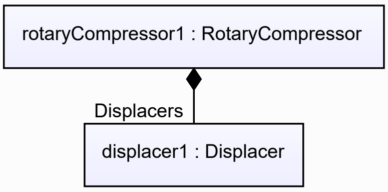
Type: *Displacer*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *Displacer*) is a child of the `<Equipment>` element for the attribute owner (a *RotaryCompressor*).

Example



Example: Implementation in Proteus Schema

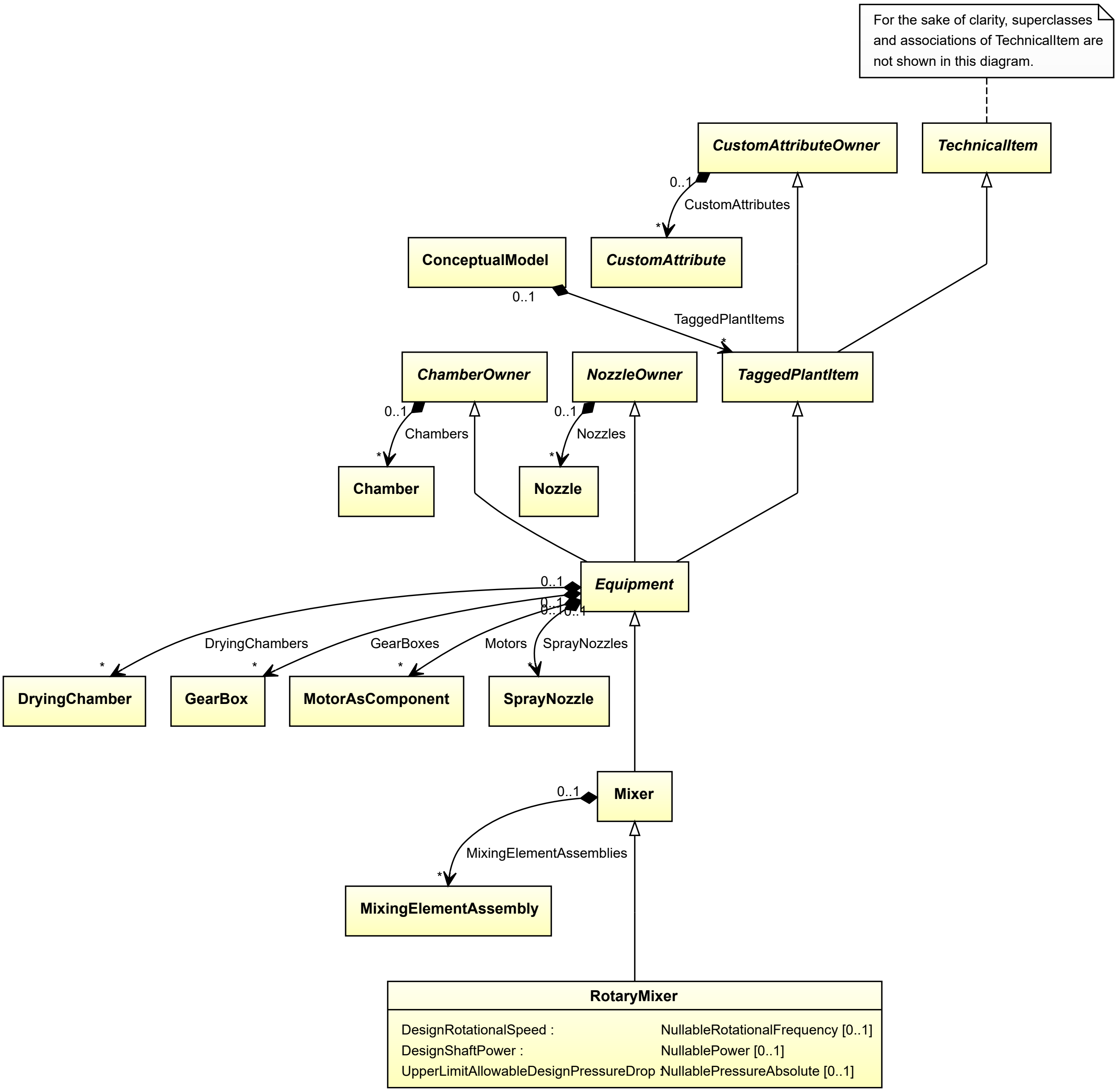
```
<Equipment
    ID="rotaryCompressor1"
    ComponentClass="RotaryCompressor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS435374" ...>
  ...
  <Equipment
      ID="displacer1"
      ComponentClass="Displacer"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/Displacer" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.123. RotaryMixer

7.123.1 Overview

Class

A *Mixer* machine that mixes by means of rotating components.



Supertypes

- *Mixer*

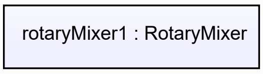
Attributes (data)

Name	Multiplicity	Type
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*
DesignShaftPower	0..1	*NullablePower*
UpperLimitAllowableDesignPressureDrop	0..1	*NullablePressureAbsolute*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: ROTARY MIXER
ComponentClass: RotaryMixer
ComponentClassURI: http://sandbox.dexpi.org/rdl/RotaryMixer

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="rotaryMixer1"
    ComponentClass="RotaryMixer"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/RotaryMixer" ...>
  ...
</Equipment>
```

7.123.2 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *RotaryMixer* is designed.

Multiplicity: 0..1

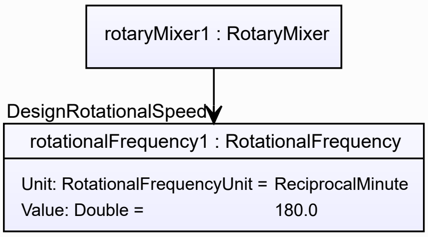
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

Example

The instance rotaryMixer1 represents a *RotaryMixer* with a *DesignRotationalSpeed* of 180.0 min^{-1}.


rotaryMixer1 : RotaryMixer
DesignRotationalSpeed
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 180.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="rotaryMixer1"
    ComponentClass="RotaryMixer"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/RotaryMixer" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignRotationalSpeed"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignRotationalSpeed"
      Format="double"
      Value="180.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.123.3 DesignShaftPower

Attribute (data)

The shaft power for which the *RotaryMixer* is designed.

Multiplicity: 0..1

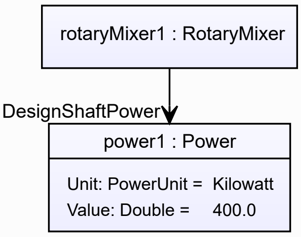
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SHAFT POWER
Name: DesignShaftPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignShaftPower

Example

The instance rotaryMixer1 represents a *RotaryMixer* with a *DesignShaftPower* of 400.0 kW.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="rotaryMixer1"
    ComponentClass="RotaryMixer"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/RotaryMixer" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignShaftPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignShaftPower"
      Format="double"
      Value="400.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.123.4 UpperLimitAllowableDesignPressureDrop

Attribute (data)

The upper limit for the pressure drop for which the *RotaryMixer* is designed.

Multiplicity: 0..1

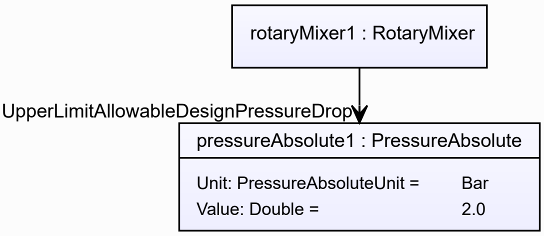
Type: *NullablePressureAbsolute*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT ALLOWABLE DESIGN PRESSURE DROP
Name: UpperLimitAllowableDesignPressureDrop
AttributeURI: http://sandbox.dexpi.org/rdl/UpperLimitAllowableDesignPressureDrop

Example

The instance rotaryMixer1 represents a *RotaryMixer* with an *UpperLimitAllowableDesignPressureDrop* of 2.0 bar.


rotaryMixer1 : RotaryMixer
UpperLimitAllowableDesignPressureDrop
pressureAbsolute1 : PressureAbsolute
Unit: PressureAbsoluteUnit = Bar
Value: Double = 2.0

Example: Implementation in Proteus Schema

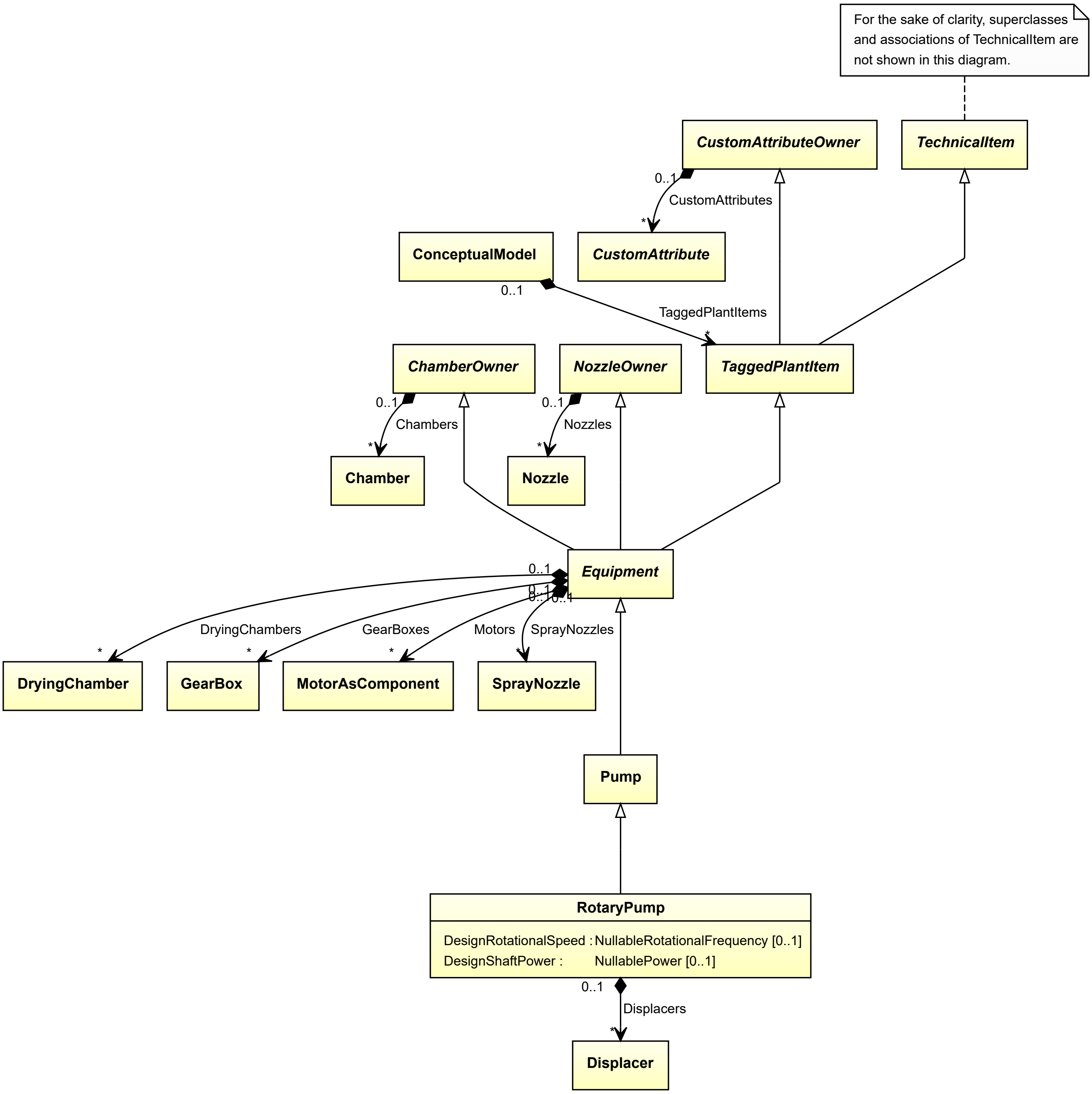
```
<Equipment
    ID="rotaryMixer1"
    ComponentClass="RotaryMixer"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/RotaryMixer" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="UpperLimitAllowableDesignPressureDrop"
      AttributeURI="http://sandbox.dexpi.org/rdl/UpperLimitAllowableDesignPressureDrop"
      Format="double"
      Value="2.0"
      Units="Bar"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1314539" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.124. RotaryPump

7.124.1 Overview

Class

A positive displacement pump that consists of a chamber containing gears, cams, screws, vanes, plungers or similar elements actuated by relative rotation of the drive shaft or casing and which has no separate inlet and outlet valves (from http://data.posccaesar.org/rdl/RDS420749).



Supertypes

- *Pump*

Attributes (data)

Name	Multiplicity	Type
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*
DesignShaftPower	0..1	*NullablePower*

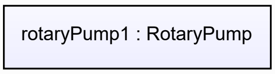
Attributes (composition)

Name	Multiplicity	Type
Displacers	*	*Displacer*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: ROTARY PUMP
ComponentClass: RotaryPump
ComponentClassURI: http://data.posccaesar.org/rdl/RDS420749

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="rotaryPump1"
    ComponentClass="RotaryPump"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS420749" ...>
  ...
</Equipment>
```

7.124.2 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *RotaryPump* is designed.

Multiplicity: 0..1

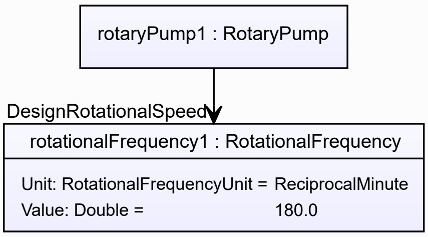
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

Example

The instance rotaryPump1 represents a *RotaryPump* with a *DesignRotationalSpeed* of 180.0 min^{-1}.


rotaryPump1 : RotaryPump
DesignRotationalSpeed
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 180.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="rotaryPump1"
    ComponentClass="RotaryPump"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS420749" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignRotationalSpeed"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignRotationalSpeed"
      Format="double"
      Value="180.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.124.3 DesignShaftPower

Attribute (data)

The shaft power for which the *RotaryPump* is designed.

Multiplicity: 0..1

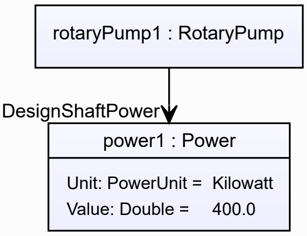
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SHAFT POWER
Name: DesignShaftPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignShaftPower

Example

The instance rotaryPump1 represents a *RotaryPump* with a *DesignShaftPower* of 400.0 kW.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="rotaryPump1"
    ComponentClass="RotaryPump"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS420749" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignShaftPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignShaftPower"
      Format="double"
      Value="400.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.124.4 Displacers

Attribute (composition)

The displacers of the *RotaryPump*.

Multiplicity: *

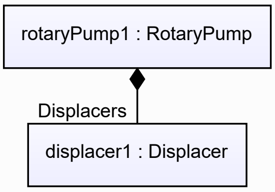
Type: *Displacer*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *Displacer*) is a child of the `<Equipment>` element for the attribute owner (a *RotaryPump*).

Example



Example: Implementation in Proteus Schema

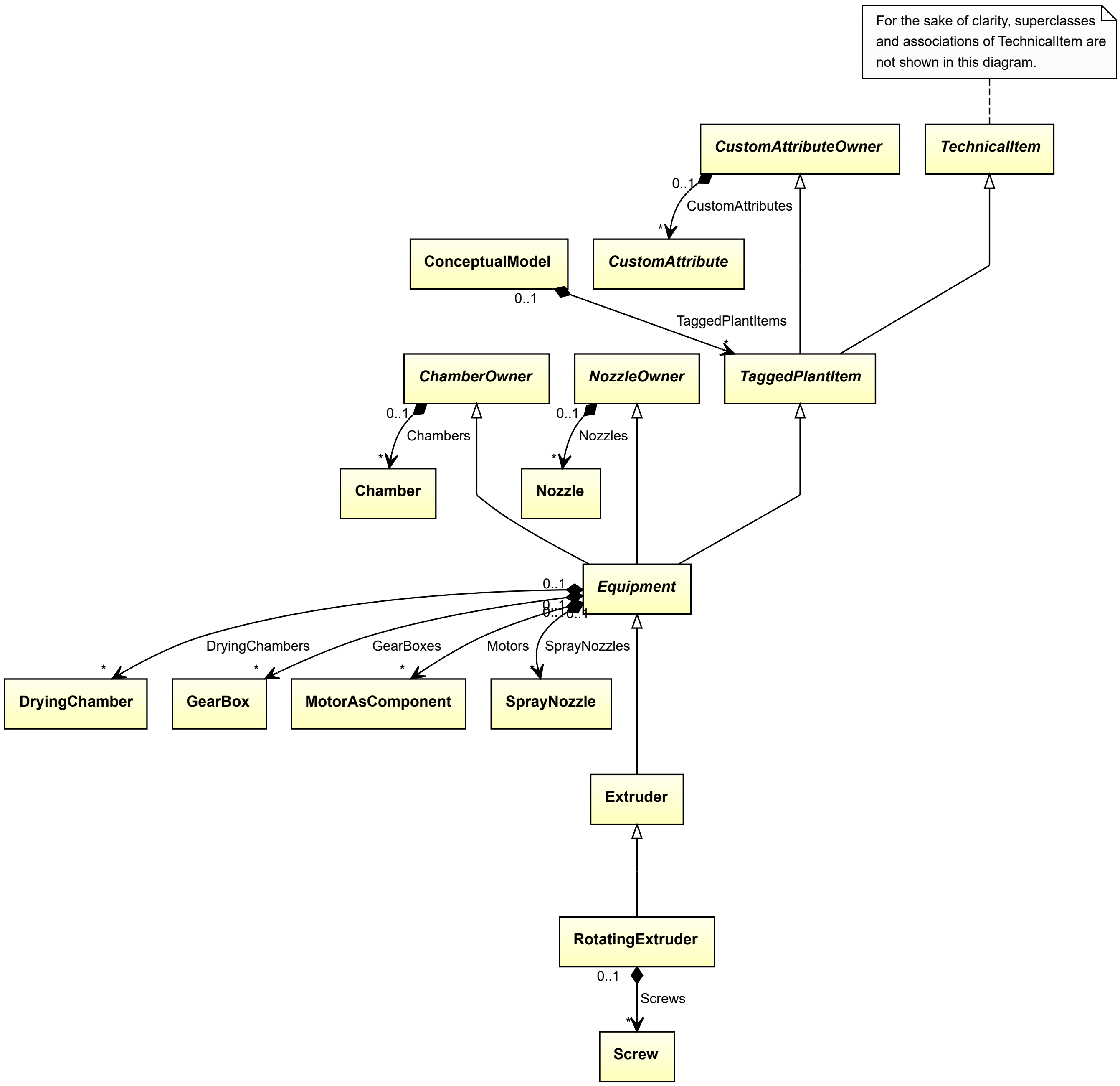
```
<Equipment
    ID="rotaryPump1"
    ComponentClass="RotaryPump"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS420749" ...>
  ...
  <Equipment
      ID="displacer1"
      ComponentClass="Displacer"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/Displacer" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.125. RotatingExtruder

7.125.1 Overview

Class

An extruder that operates in a continuous process. Typically using a screw to build up pressure in the melt. It can incorporate a mixing stage with a forming stage (from http://data.posccaesar.org/rdl/RDS394045941).



Supertypes

- *Extruder*

Attributes (composition)

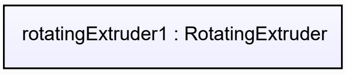
Name	Multiplicity	Type
Screws	*	*Screw*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>

RDL reference: AUGER EXTRUDER
ComponentClass: AugerExtruder
ComponentClassURI: http://data.posccaesar.org/rdl/RDS394045941

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="rotatingExtruder1"
    ComponentClass="AugerExtruder"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS394045941" ...>
  ...
</Equipment>
```

7.125.2 Screws

Attribute (composition)

The screws of the *RotatingExtruder*.

Multiplicity: *

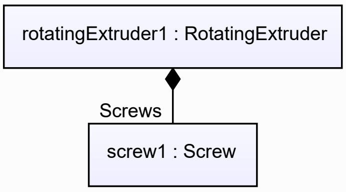
Type: *Screw*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *Screw*) is a child of the `<Equipment>` element for the attribute owner (a *RotatingExtruder*).

Example



Example: Implementation in Proteus Schema

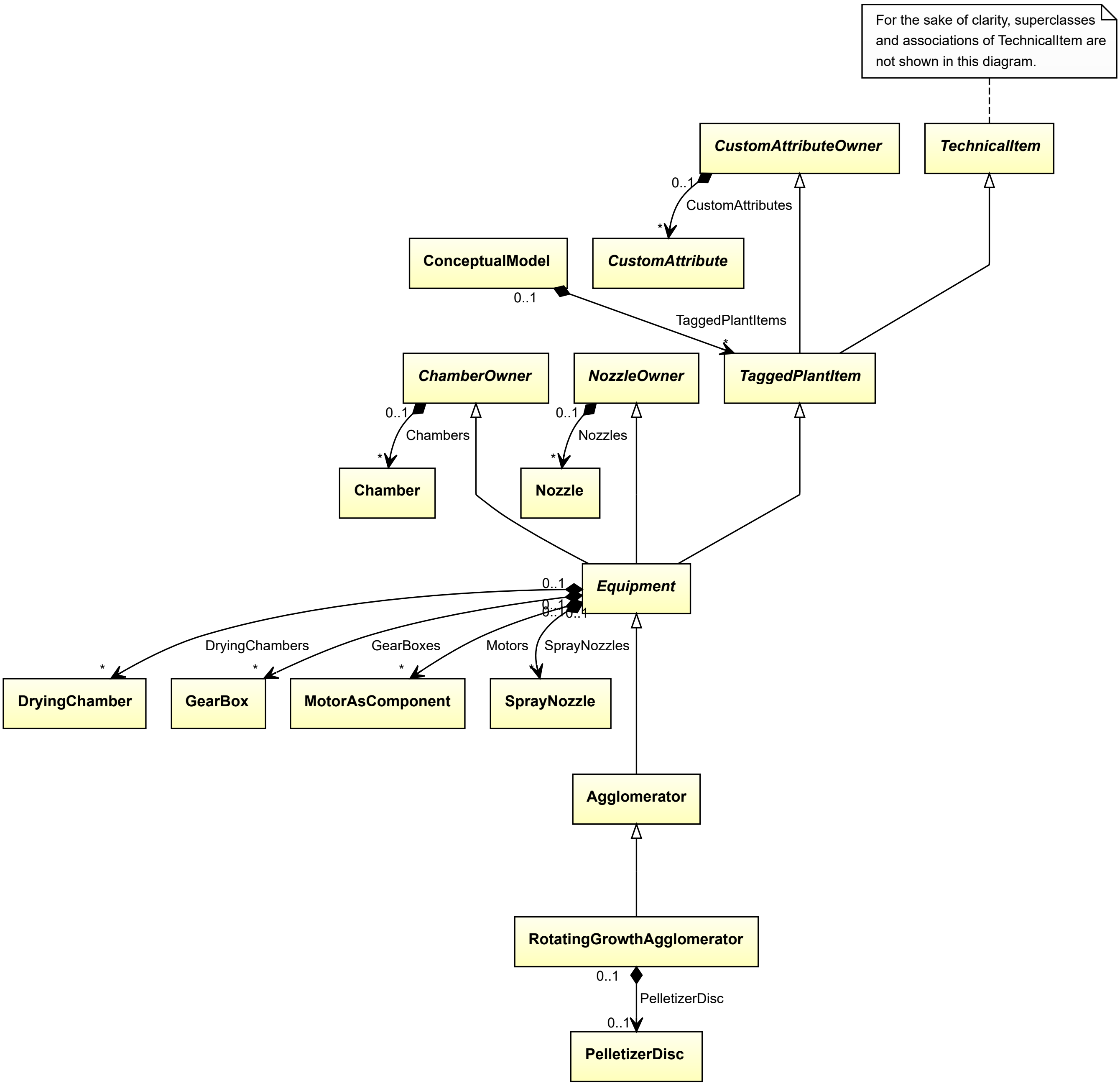
```
<Equipment
    ID="rotatingExtruder1"
    ComponentClass="AugerExtruder"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS394045941" ...>
  ...
  <Equipment
      ID="screw1"
      ComponentClass="Screw"
      ComponentClassURI="http://data.posccaesar.org/rdl/RDS7219994" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.126. RotatingGrowthAgglomerator

7.126.1 Overview

Class

An agglomerator which uses a pelletizer disc to produce pellets.



Supertypes

- *Agglomerator*

Attributes (composition)

Name	Multiplicity	Type
PelletizerDisc	0..1	*PelletizerDisc*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`

RDL reference: ROTATING GROWTH AGGLOMERATOR
ComponentClass: RotatingGrowthAgglomerator
ComponentClassURI: http://sandbox.dexpi.org/rdl/RotatingGrowthAgglomerator

Example

rotatingGrowthAgglomerator1 : RotatingGrowthAgglomerator

Example: Implementation in Proteus Schema

```
<Equipment
    ID="rotatingGrowthAgglomerator1"
    ComponentClass="RotatingGrowthAgglomerator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/RotatingGrowthAgglomerator" ...>
  ...
</Equipment>
```

7.126.2 PelletizerDisc

Attribute (composition)

The pelletizing disc of the *RotatingGrowthAgglomerator*.

Multiplicity: 0..1

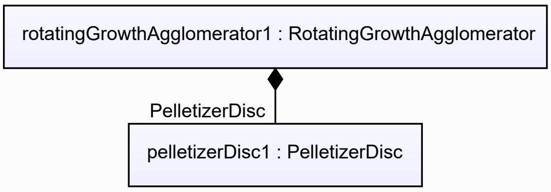
Type: *PelletizerDisc*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *PelletizerDisc*) is a child of the `<Equipment>` element for the attribute owner (a *RotatingGrowthAgglomerator*).

Example


rotatingGrowthAgglomerator1 : RotatingGrowthAgglomerator
PelletizerDisc
pelletizerDisc1 : PelletizerDisc

Example: Implementation in Proteus Schema

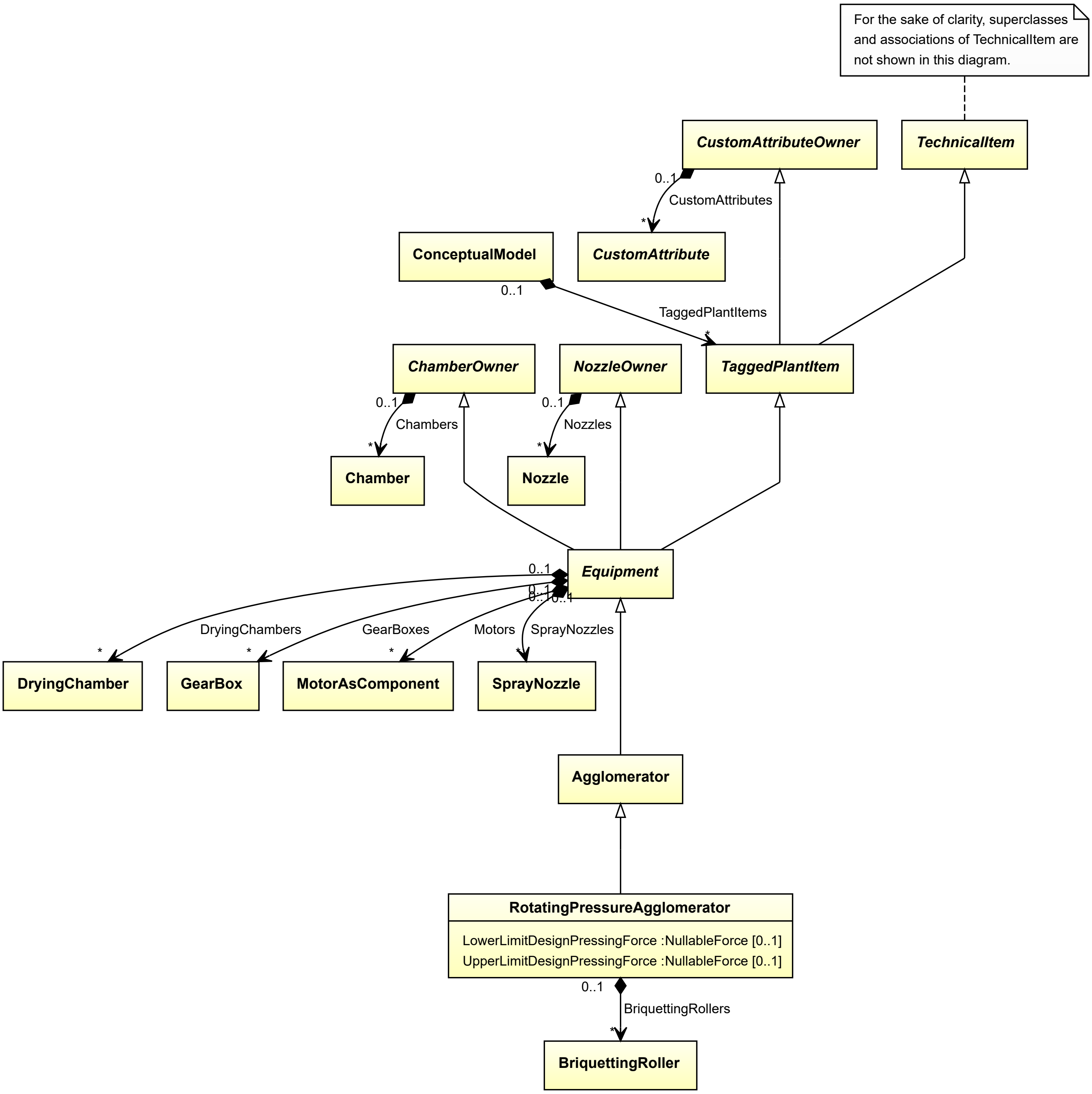
```
<Equipment
    ID="rotatingGrowthAgglomerator1"
    ComponentClass="RotatingGrowthAgglomerator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/RotatingGrowthAgglomerator" ...>
  ...
  <Equipment
      ID="pelletizerDisc1"
      ComponentClass="PelletingDisc"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/PelletingDisc" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.127. RotatingPressureAgglomerator

7.127.1 Overview

Class

An agglomerator which uses briquetting rollers to produce pressure and to form material.



Supertypes

- *Agglomerator*

Attributes (data)

Name	Multiplicity	Type
LowerLimitDesignPressingForce	0..1	*NullableForce*
UpperLimitDesignPressingForce	0..1	*NullableForce*

Attributes (composition)

Name	Multiplicity	Type
BriquettingRollers	*	*BriquettingRoller*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: ROTATING PRESSURE AGGLOMERATOR
ComponentClass: RotatingPressureAgglomerator
ComponentClassURI: http://sandbox.dexpi.org/rdl/RotatingPressureAgglomerator

Example

rotatingPressureAgglomerator1 : RotatingPressureAgglomerator

Example: Implementation in Proteus Schema

```
<Equipment
    ID="rotatingPressureAgglomerator1"
    ComponentClass="RotatingPressureAgglomerator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/RotatingPressureAgglomerator" ...>
  ...
</Equipment>
```

7.127.2 BriquettingRollers

Attribute (composition)

The briquetting rollers of the *RotatingPressureAgglomerator*.

Multiplicity: *

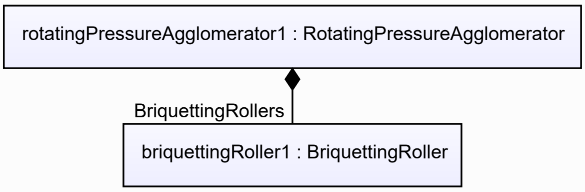
Type: *BriquettingRoller*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *BriquettingRoller*) is a child of the `<Equipment>` element for the attribute owner (a *RotatingPressureAgglomerator*).

Example


rotatingPressureAgglomerator1 : RotatingPressureAgglomerator
BriquettingRollers
briquettingRoller1 : BriquettingRoller

Example: Implementation in Proteus Schema

```
<Equipment
    ID="rotatingPressureAgglomerator1"
    ComponentClass="RotatingPressureAgglomerator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/RotatingPressureAgglomerator" ...>
  ...
  <Equipment
      ID="briquettingRoller1"
      ComponentClass="BriquettingRoller"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/BriquettingRoller" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.127.3 LowerLimitDesignPressingForce

Attribute (data)

The lower limit for the pressing force for which the *RotatingPressureAgglomerator* is designed.

Multiplicity: 0..1

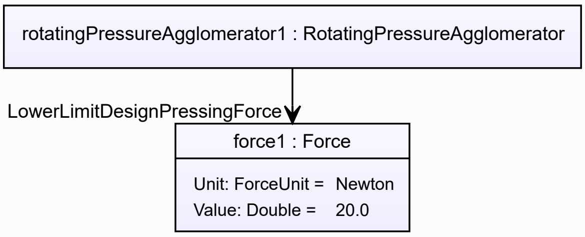
Type: *NullableForce*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: LOWER LIMIT DESIGN PRESSING FORCE
Name: LowerLimitDesignPressingForce
AttributeURI: http://sandbox.dexpi.org/rdl/LowerLimitDesignPressingForce

Example

The instance rotatingPressureAgglomerator1 represents a *RotatingPressureAgglomerator* with a *LowerLimitDesignPressingForce* of 20.0 N.


rotatingPressureAgglomerator1 : RotatingPressureAgglomerator
LowerLimitDesignPressingForce
force1 : Force
Unit: ForceUnit = Newton
Value: Double = 20.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="rotatingPressureAgglomerator1"
    ComponentClass="RotatingPressureAgglomerator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/RotatingPressureAgglomerator" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="LowerLimitDesignPressingForce"
      AttributeURI="http://sandbox.dexpi.org/rdl/LowerLimitDesignPressingForce"
      Format="double"
      Value="20.0"
      Units="Newton"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1337939" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.127.4 UpperLimitDesignPressingForce

Attribute (data)

The upper limit for the pressing force for which the *RotatingPressureAgglomerator* is designed.

Multiplicity: 0..1

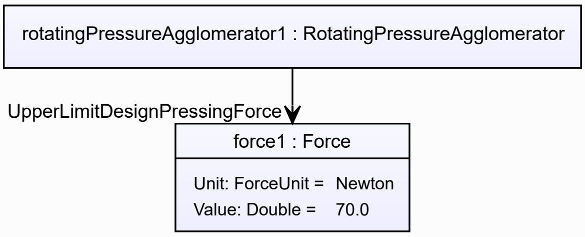
Type: *NullableForce*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT DESIGN PRESSING FORCE
Name: UpperLimitDesignPressingForce
AttributeURI: http://sandbox.dexpi.org/rdl/UpperLimitDesignPressingForce

Example

The instance rotatingPressureAgglomerator1 represents a *RotatingPressureAgglomerator* with an *UpperLimitDesignPressingForce* of 70.0 N.


rotatingPressureAgglomerator1 : RotatingPressureAgglomerator
UpperLimitDesignPressingForce
force1 : Force
Unit: ForceUnit = Newton
Value: Double = 70.0

Example: Implementation in Proteus Schema

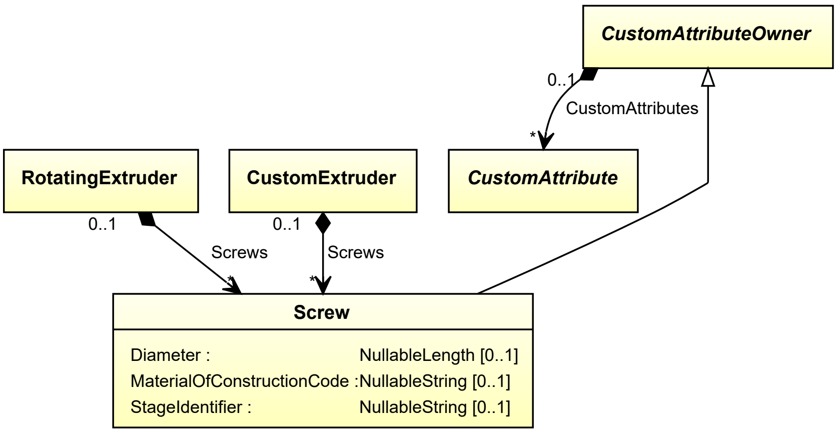
```
<Equipment
    ID="rotatingPressureAgglomerator1"
    ComponentClass="RotatingPressureAgglomerator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/RotatingPressureAgglomerator" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="UpperLimitDesignPressingForce"
      AttributeURI="http://sandbox.dexpi.org/rdl/UpperLimitDesignPressingForce"
      Format="double"
      Value="70.0"
      Units="Newton"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1337939" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.128. Screw

7.128.1 Overview

Class

A shaft with a helical shaped shaft design (from http://data.posccaesar.org/rdl/RDS7219994).


CustomAttributeOwner
0..1
CustomAttributes
*
RotatingExtruder
CustomExtruder
CustomAttribute
0..1
0..1
Screws
Screws
*
*
Screw
Diameter : NullableLength [0..1]
MaterialOfConstructionCode : NullableString [0..1]
StageIdentifier : NullableString [0..1]

Supertypes

- *CustomAttributeOwner*

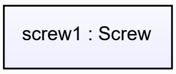
Attributes (data)

Name	Multiplicity	Type
Diameter	0..1	*NullableLength*
MaterialOfConstructionCode	0..1	*NullableString*
StageIdentifier	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: SCREW
ComponentClass: Screw
ComponentClassURI: http://data.posccaesar.org/rdl/RDS7219994

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="screw1"
    ComponentClass="Screw"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS7219994" ...>
  ...
</Equipment>
```

7.128.2 Diameter

Attribute (data)

The diameter of the *Screw*.

Multiplicity: 0..1

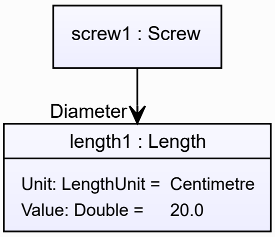
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DIAMETER
Name: Diameter
AttributeURI: http://data.posccaesar.org/rdl/RDS350954

Example

The instance screw1 represents a *Screw* with a *Diameter* of 20.0 cm.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="screw1"
    ComponentClass="Screw"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS7219994" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="Diameter"
      AttributeURI="http://data.posccaesar.org/rdl/RDS350954"
      Format="double"
      Value="20.0"
      Units="Centimetre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1318004" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.128.3 MaterialOfConstructionCode

Attribute (data)

A code that gives the material of construction of the *Screw*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: MATERIAL OF CONSTRUCTION CODE ASSIGNMENT CLASS
Name: MaterialOfConstructionCodeAssignmentClass
AttributeURI: http://data.posccaesar.org/rdl/RDS1460719741

Example

“1.4306” (*String*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="screw1"
    ComponentClass="Screw"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS7219994" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="MaterialOfConstructionCodeAssignmentClass"
      AttributeURI="http://data.posccaesar.org/rdl/RDS1460719741"
      Format="string"
      Value="1.4306" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.128.4 StageIdentifier

Attribute (data)

The stage identifier of the *Screw*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: STAGE IDENTIFIER ASSIGNMENT CLASS
Name: StageIdentifierAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/StageIdentifierAssignmentClass

Example

"s1" (*String*)

Example: Implementation in Proteus Schema

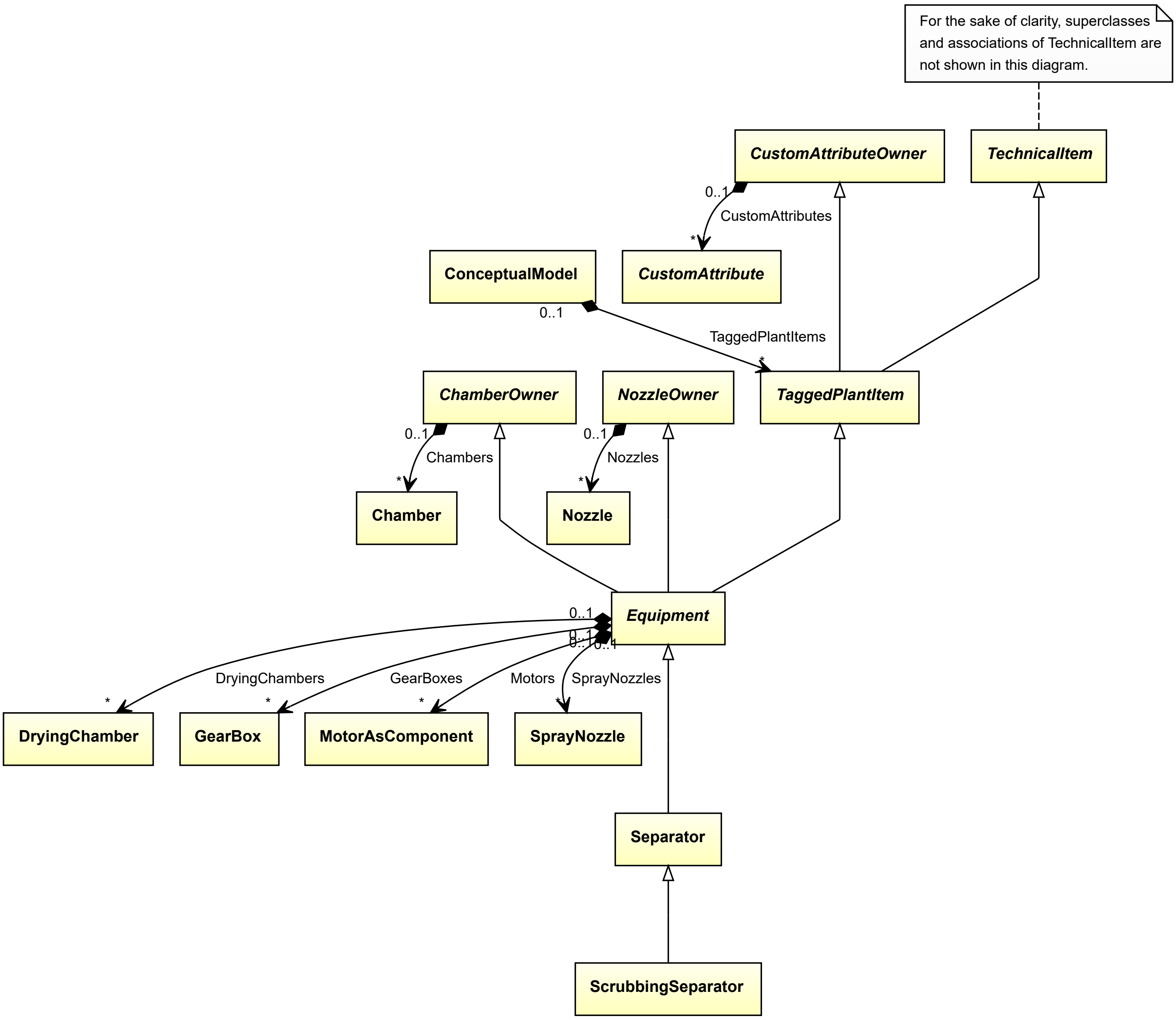
```
<Equipment
    ID="screw1"
    ComponentClass="Screw"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS7219994" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="StageIdentifierAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/StageIdentifierAssignmentClass"
      Format="string"
      Value="s1" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.129. ScrubbingSeparator

7.129.1 Overview

Class

A separator that is intended to clean gas by washing the gas flow with water or with another liquid entering at the top of the vessel.



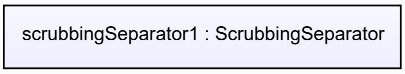
Supertypes

- *Separator*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: SCRUBBING SEPARATOR
ComponentClass: ScrubbingSeparator
ComponentClassURI: http://sandbox.dexpi.org/rdl/ScrubbingSeparator

Example



Example: Implementation in Proteus Schema

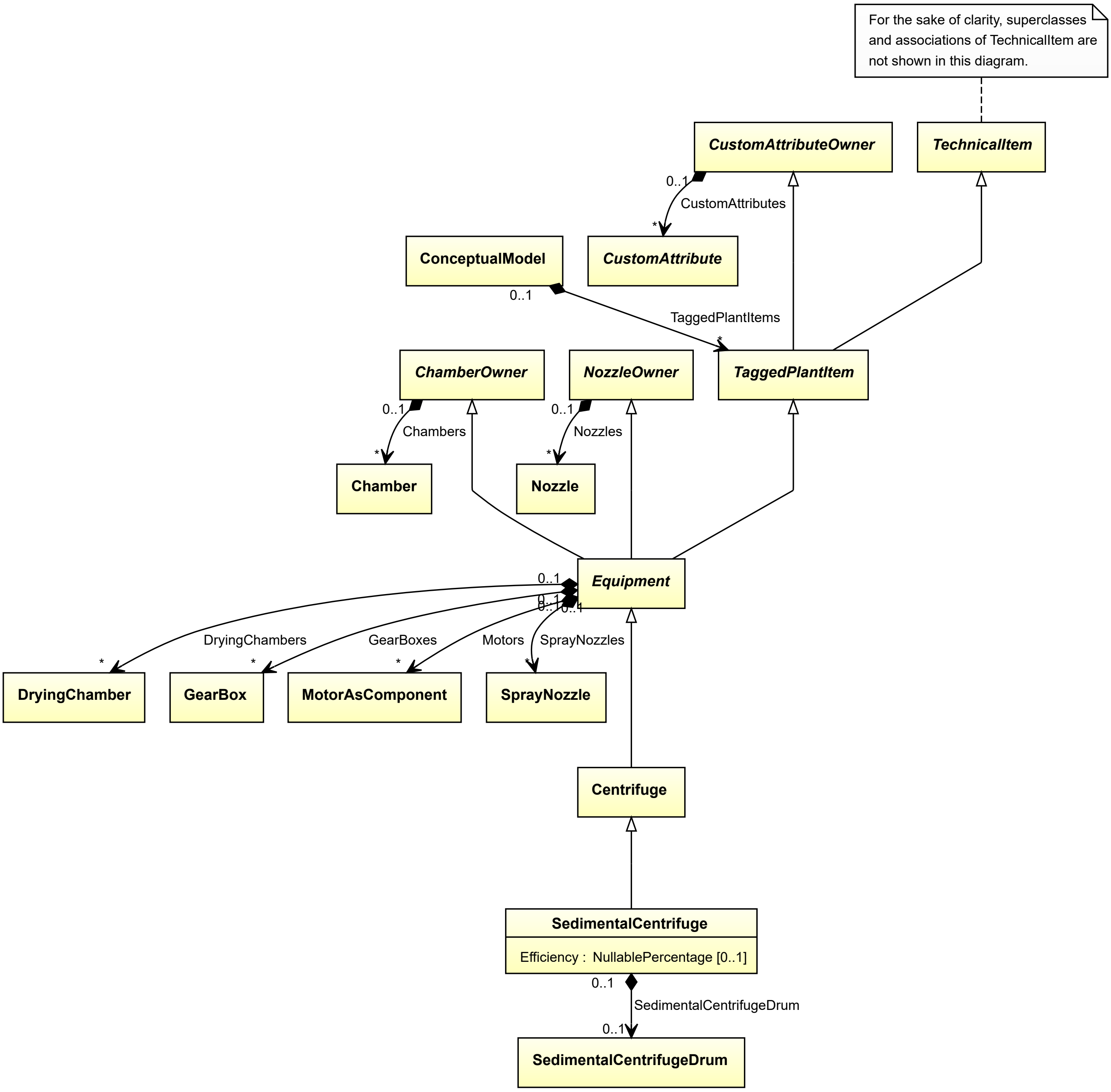
```
<Equipment
    ID="scrubbingSeparator1"
    ComponentClass="ScrubbingSeparator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ScrubbingSeparator" ...>
  ...
</Equipment>
```

7.130. SedimentalCentrifuge

7.130.1 Overview

Class

A centrifuge that is intended to separate solids from liquids by a centrifugal process based on different densities.



Supertypes

- *Centrifuge*

Attributes (data)

Name	Multiplicity	Type
Efficiency	0..1	*NullablePercentage*

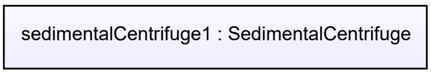
Attributes (composition)

Name	Multiplicity	Type
SedimentalCentrifugeDrum	0..1	*SedimentalCentrifugeDrum*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: SEDIMENTAL CENTRIFUGE
ComponentClass: SedimentalCentrifuge
ComponentClassURI: http://sandbox.dexpi.org/rdl/SedimentalCentrifuge

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="sedimentalCentrifuge1"
    ComponentClass="SedimentalCentrifuge"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SedimentalCentrifuge" ...>
  ...
</Equipment>
```

7.130.2 Efficiency

Attribute (data)

The efficiency of the *SedimentalCentrifuge*.

Multiplicity: 0..1

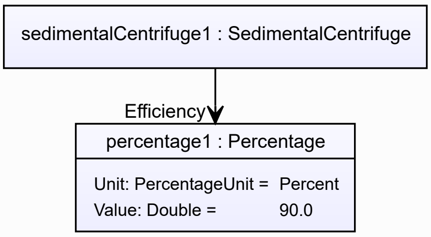
Type: *NullablePercentage*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: EFFICIENCY
Name: Efficiency
AttributeURI: http://data.posccaesar.org/rdl/RDS362654

Example

The instance sedimentalCentrifuge1 represents a *SedimentalCentrifuge* with an *Efficiency* of 90.0 ???.


sedimentalCentrifuge1 : SedimentalCentrifuge
Efficiency
percentage1 : Percentage
Unit: PercentageUnit = Percent
Value: Double = 90.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="sedimentalCentrifuge1"
    ComponentClass="SedimentalCentrifuge"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SedimentalCentrifuge" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="Efficiency"
      AttributeURI="http://data.posccaesar.org/rdl/RDS362654"
      Format="double"
      Value="90.0"
      Units="Percent"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1317959" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.130.3 SedimentalCentrifugeDrum

Attribute (composition)

The sedimental centrifuge drum of the *SedimentalCentrifuge*.

Multiplicity: 0..1

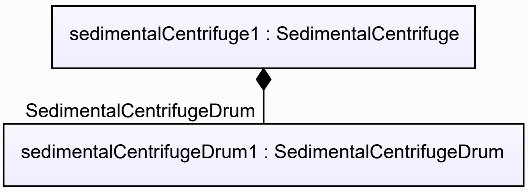
Type: *SedimentalCentrifugeDrum*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *SedimentalCentrifugeDrum*) is a child of the `<Equipment>` element for the attribute owner (a *SedimentalCentrifuge*).

Example


sedimentalCentrifuge1 : SedimentalCentrifuge
SedimentalCentrifugeDrum
sedimentalCentrifugeDrum1 : SedimentalCentrifugeDrum

Example: Implementation in Proteus Schema

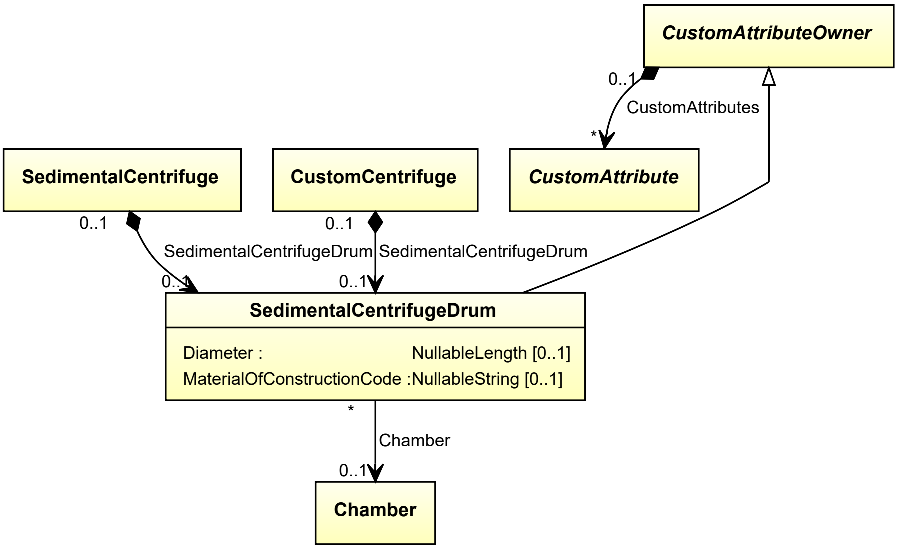
```
<Equipment
    ID="sedimentalCentrifuge1"
    ComponentClass="SedimentalCentrifuge"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SedimentalCentrifuge" ...>
  ...
  <Equipment
      ID="sedimentalCentrifugeDrum1"
      ComponentClass="SedimentalCentrifugeDrum"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/SedimentalCentrifugeDrum" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.131. SedimentalCentrifugeDrum

7.131.1 Overview

Class

A SedimentalCentrifugeDrum is a drum and a component of a *SedimentalCentrifuge*.



Supertypes

- *CustomAttributeOwner*

Attributes (data)

Name	Multiplicity	Type
Diameter	0..1	*NullableLength*
MaterialOfConstructionCode	0..1	*NullableString*

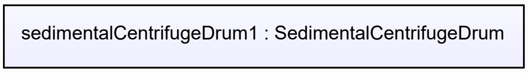
Attributes (reference)

Name	Multiplicity	Type
Chamber	0..1	*Chamber*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: SEDIMENTAL CENTRIFUGE DRUM
ComponentClass: SedimentalCentrifugeDrum
ComponentClassURI: http://sandbox.dexpi.org/rdl/SedimentalCentrifugeDrum

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="sedimentalCentrifugeDrum1"
    ComponentClass="SedimentalCentrifugeDrum"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SedimentalCentrifugeDrum" ...>
  ...
</Equipment>
```

7.131.2 Chamber

Attribute (reference)

The *Chamber* in which the *SedimentalCentrifugeDrum* is located, if applicable. The Chamber must be a component of the same object as the *SedimentalCentrifugeDrum*.

Multiplicity: 0..1

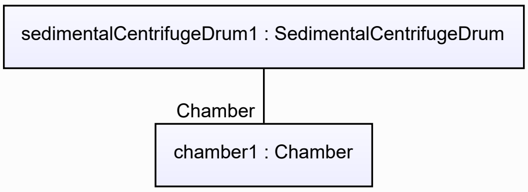
Type: *Chamber*

Opposite multiplicity: 0..*

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: `"is located in"`
Opposite association type: `"is the location of"`

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="sedimentalCentrifugeDrum1"
    ComponentClass="SedimentalCentrifugeDrum"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SedimentalCentrifugeDrum" ...>
  ...
  <Association
    Type="is located in"
    ItemID="chamber1" />
  ...
<Equipment />
...
<Equipment
    ID="chamber1"
    ComponentClass="Chamber"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS903151421" ...>
  ...
  <Association
    Type="is the location of"
    ItemID="sedimentalCentrifugeDrum1" />
  ...
<Equipment />
```

7.131.3 Diameter

Attribute (data)

The diameter of the *SedimentalCentrifugeDrum*.

Multiplicity: 0..1

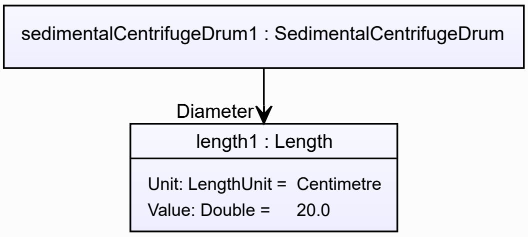
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DIAMETER
Name: Diameter
AttributeURI: http://data.posccaesar.org/rdl/RDS350954

Example

The instance sedimentalCentrifugeDrum1 represents a *SedimentalCentrifugeDrum* with a *Diameter* of 20.0 cm.


sedimentalCentrifugeDrum1 : SedimentalCentrifugeDrum
Diameter
length1 : Length
Unit: LengthUnit = Centimetre
Value: Double = 20.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="sedimentalCentrifugeDrum1"
    ComponentClass="SedimentalCentrifugeDrum"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SedimentalCentrifugeDrum" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="Diameter"
      AttributeURI="http://data.posccaesar.org/rdl/RDS350954"
      Format="double"
      Value="20.0"
      Units="Centimetre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1318004" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.131.4 MaterialOfConstructionCode

Attribute (data)

A code that gives the material of construction of the *SedimentalCentrifugeDrum*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: MATERIAL OF CONSTRUCTION CODE ASSIGNMENT CLASS
Name: MaterialOfConstructionCodeAssignmentClass
AttributeURI: http://data.posccaesar.org/rdl/RDS1460719741

Example

“1.4306” (*String*)

Example: Implementation in Proteus Schema

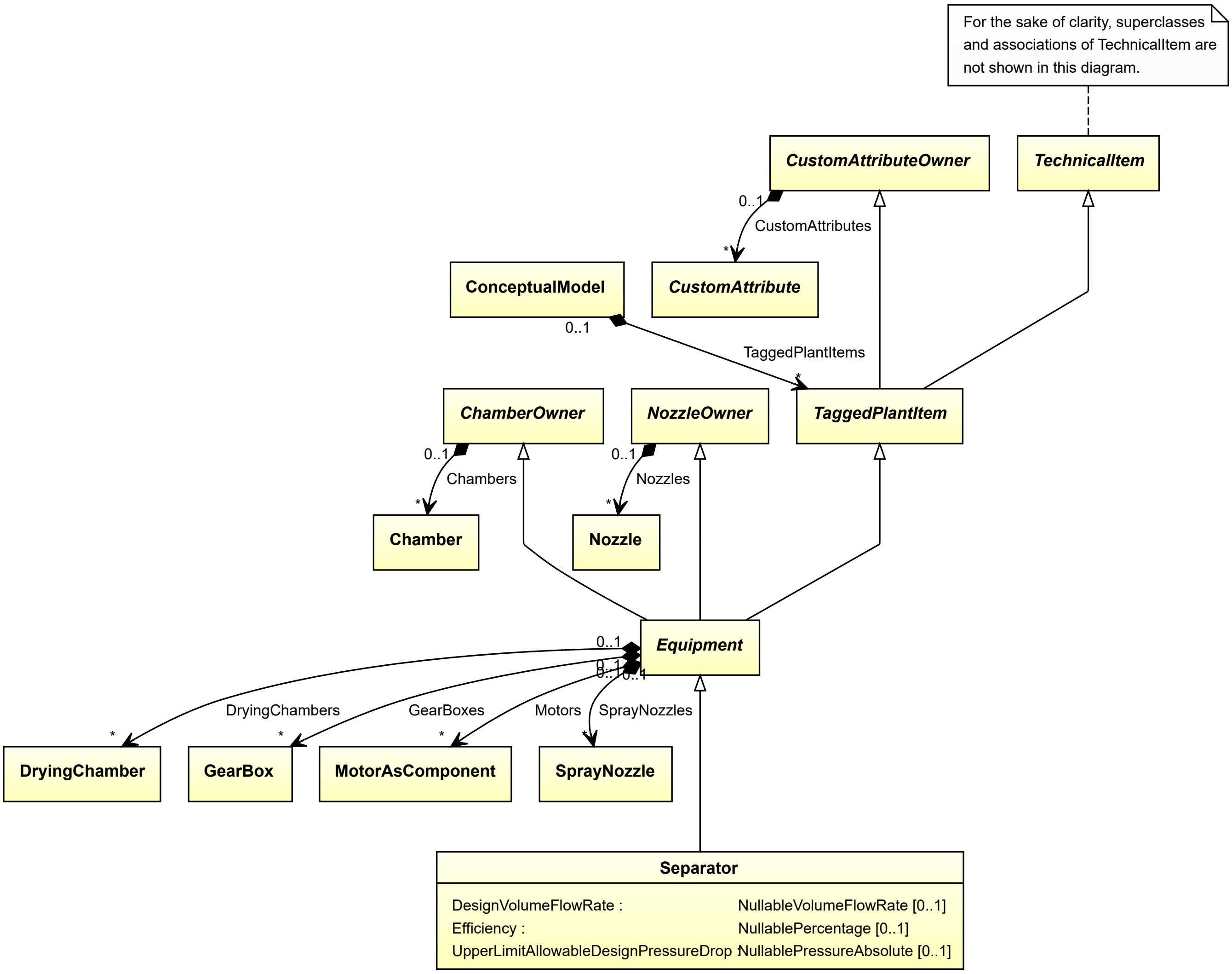
```
<Equipment
    ID="sedimentalCentrifugeDrum1"
    ComponentClass="SedimentalCentrifugeDrum"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SedimentalCentrifugeDrum" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="MaterialOfConstructionCodeAssignmentClass"
      AttributeURI="http://data.posccaesar.org/rdl/RDS1460719741"
      Format="string"
      Value="1.4306" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.132. Separator

7.132.1 Overview

Class

A ‘device’ intended to separate different types of substances (from http://data.posccaesar.org/rdl/RDS2194378711).



Supertypes

- *Equipment*

Subtypes

- *CustomSeparator*
- *ElectricalSeparator*
- *GravitationalSeparator*
- *MechanicalSeparator*
- *ScrubbingSeparator*

Attributes (data)

Name	Multiplicity	Type
DesignVolumeFlowRate	0..1	*NullableVolumeFlowRate*
Efficiency	0..1	*NullablePercentage*
UpperLimitAllowableDesignPressureDrop	0..1	*NullablePressureAbsolute*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: SEPARATOR
ComponentClass: Separator
ComponentClassURI: http://data.posccaesar.org/rdl/RDS2194378711

Example

Example: Implementation in Proteus Schema

```
<Equipment
    ID="separator1"
    ComponentClass="Separator"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS2194378711" ...>
  ...
</Equipment>
```

7.132.2 DesignVolumeFlowRate

Attribute (data)

The volume flow rate for which the *Separator* is designed.

Multiplicity: 0..1

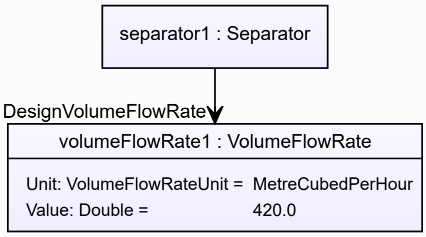
Type: *NullableVolumeFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN VOLUME FLOW RATE
Name: DesignVolumeFlowRate
AttributeURI: http://data.posccaesar.org/rdl/RDS14286227

Example

The instance separator1 represents a *Separator* with a *DesignVolumeFlowRate* of 420.0 m^3/h.


separator1 : Separator
DesignVolumeFlowRate
volumeFlowRate1 : VolumeFlowRate
Unit: VolumeFlowRateUnit = MetreCubedPerHour
Value: Double = 420.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="separator1"
    ComponentClass="Separator"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS2194378711" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignVolumeFlowRate"
      AttributeURI="http://data.posccaesar.org/rdl/RDS14286227"
      Format="double"
      Value="420.0"
      Units="MetreCubedPerHour"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1321064" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.132.3 Efficiency

Attribute (data)

The efficiency of the *Separator*.

Multiplicity: 0..1

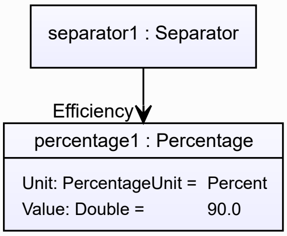
Type: *NullablePercentage*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: EFFICIENCY
Name: Efficiency
AttributeURI: http://data.posccaesar.org/rdl/RDS362654

Example

The instance separator1 represents a *Separator* with an *Efficiency* of 90.0 ???.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="separator1"
    ComponentClass="Separator"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS2194378711" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="Efficiency"
      AttributeURI="http://data.posccaesar.org/rdl/RDS362654"
      Format="double"
      Value="90.0"
      Units="Percent"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1317959" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.132.4 UpperLimitAllowableDesignPressureDrop

Attribute (data)

The upper limit for the pressure drop for which the *Separator* is designed.

Multiplicity: 0..1

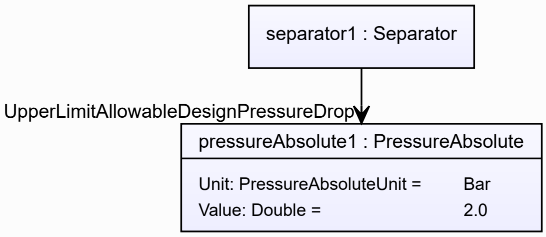
Type: *NullablePressureAbsolute*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT ALLOWABLE DESIGN PRESSURE DROP
Name: UpperLimitAllowableDesignPressureDrop
AttributeURI: http://sandbox.dexpi.org/rdl/UpperLimitAllowableDesignPressureDrop

Example

The instance separator1 represents a *Separator* with an *UpperLimitAllowableDesignPressureDrop* of 2.0 bar.


separator1 : Separator
UpperLimitAllowableDesignPressureDrop
pressureAbsolute1 : PressureAbsolute
Unit: PressureAbsoluteUnit = Bar
Value: Double = 2.0

Example: Implementation in Proteus Schema

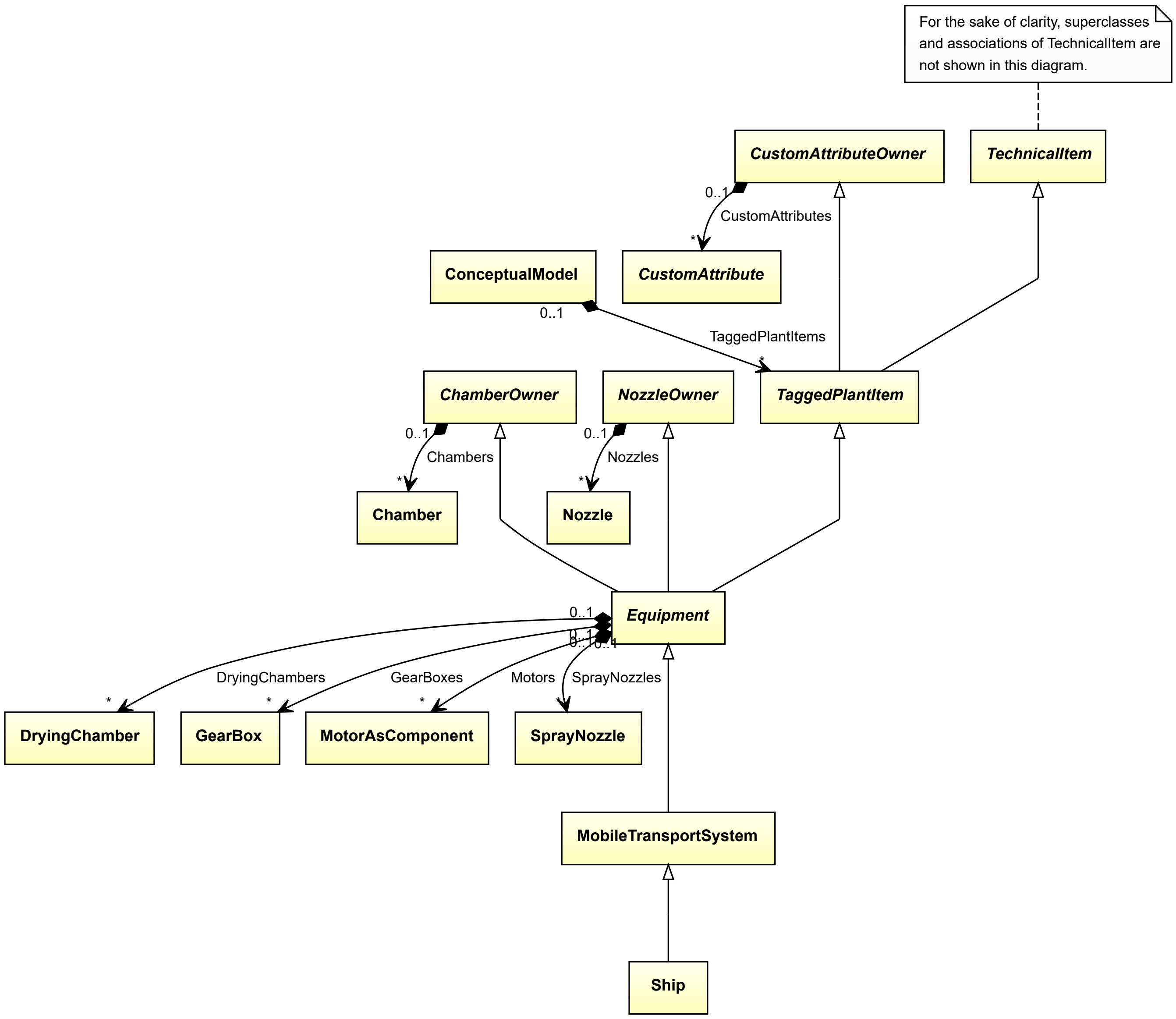
```
<Equipment
    ID="separator1"
    ComponentClass="Separator"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS2194378711" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="UpperLimitAllowableDesignPressureDrop"
      AttributeURI="http://sandbox.dexpi.org/rdl/UpperLimitAllowableDesignPressureDrop"
      Format="double"
      Value="2.0"
      Units="Bar"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1314539" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.133. Ship

7.133.1 Overview

Class

A watercraft and *MobileTransportSystem* that is a sea-going vessel of considerable size.



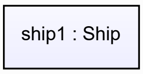
Supertypes

- *MobileTransportSystem*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: SHIP
ComponentClass: Ship
ComponentClassURI: http://data.posccaesar.org/rdl/RDS11523932

Example



Example: Implementation in Proteus Schema

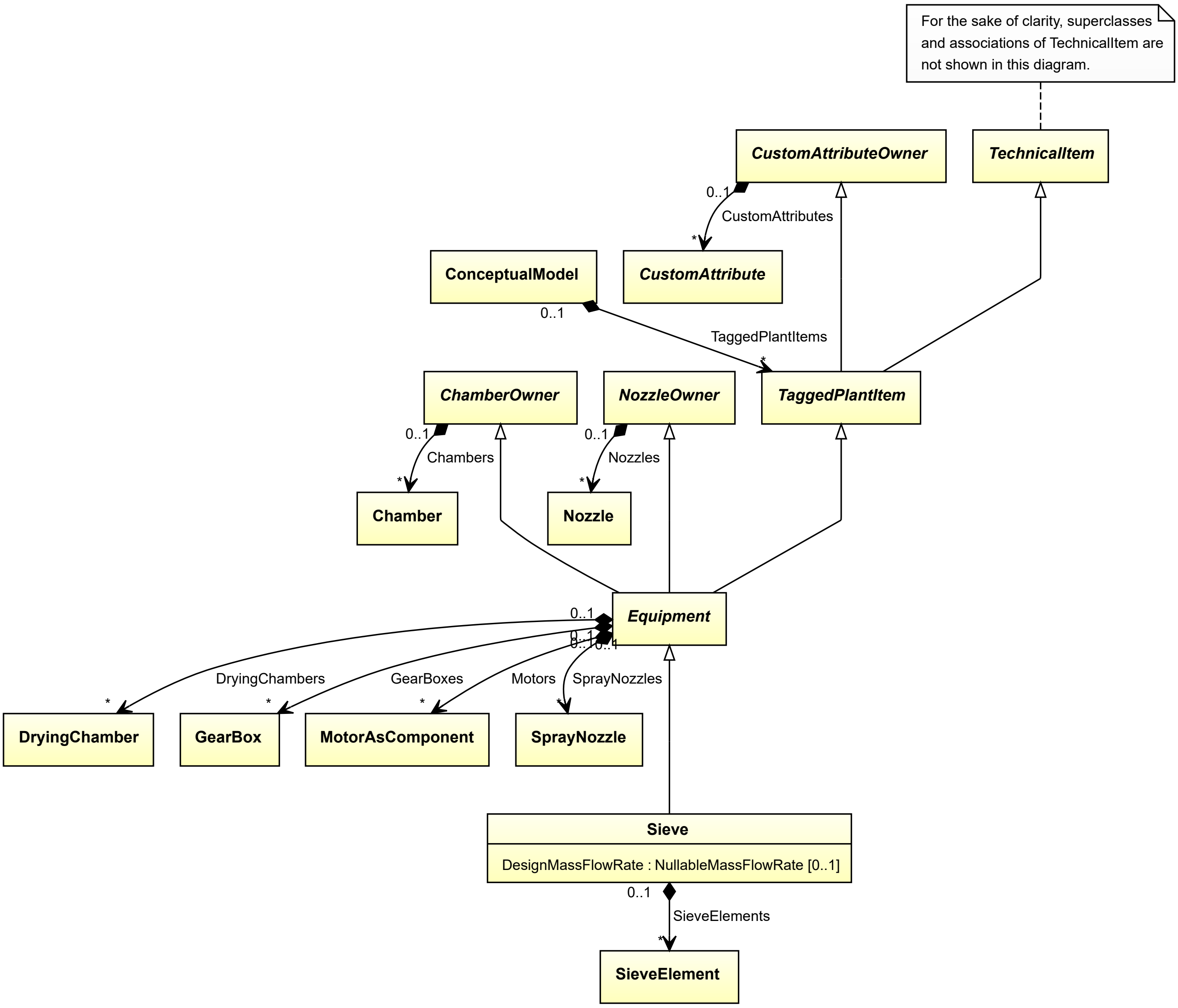
```
<Equipment
    ID="ship1"
    ComponentClass="Ship"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11523932" ...>
  ...
</Equipment>
```

7.134. Sieve

7.134.1 Overview

Class

A device that removes particles from a fluid when the fluid passes through or separates particles or molecules according to their size.



Supertypes

- *Equipment*

Subtypes

- *CustomSieve*
- *RevolvingSieve*
- *StationarySieve*
- *VibratingSieve*

Attributes (data)

Name	Multiplicity	Type
DesignMassFlowRate	0..1	*NullableMassFlowRate*

Attributes (composition)

Name	Multiplicity	Type
SieveElements	*	*SieveElement*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: SIEVE
ComponentClass: Sieve
ComponentClassURI: http://sandbox.dexpi.org/rdl/Sieve

Example

Example: Implementation in Proteus Schema

```
<Equipment
    ID="sieve1"
    ComponentClass="Sieve"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Sieve" ...>
  ...
</Equipment>
```

7.134.2 DesignMassFlowRate

Attribute (data)

The mass flow rate for which the *Sieve* is designed.

Multiplicity: 0..1

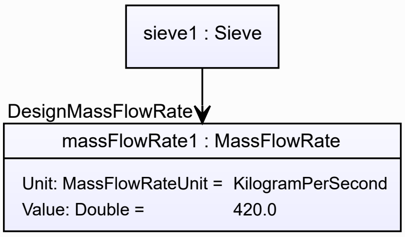
Type: *NullableMassFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN MASS FLOW RATE
Name: DesignMassFlowRate
AttributeURI: http://data.posccaesar.org/rdl/RDS14286182

Example

The instance sieve1 represents a *Sieve* with a *DesignMassFlowRate* of 420.0 kg/s.


sieve1 : Sieve
DesignMassFlowRate
massFlowRate1 : MassFlowRate
Unit: MassFlowRateUnit = KilogramPerSecond
Value: Double = 420.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="sieve1"
    ComponentClass="Sieve"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Sieve" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignMassFlowRate"
      AttributeURI="http://data.posccaesar.org/rdl/RDS14286182"
      Format="double"
      Value="420.0"
      Units="KilogramPerSecond"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1329659" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.134.3 SieveElements

Attribute (composition)

The sieve elements of the *Sieve*.

Multiplicity: *

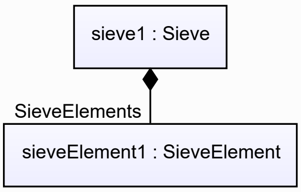
Type: *SieveElement*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *SieveElement*) is a child of the `<Equipment>` element for the attribute owner (a *Sieve*).

Example



Example: Implementation in Proteus Schema

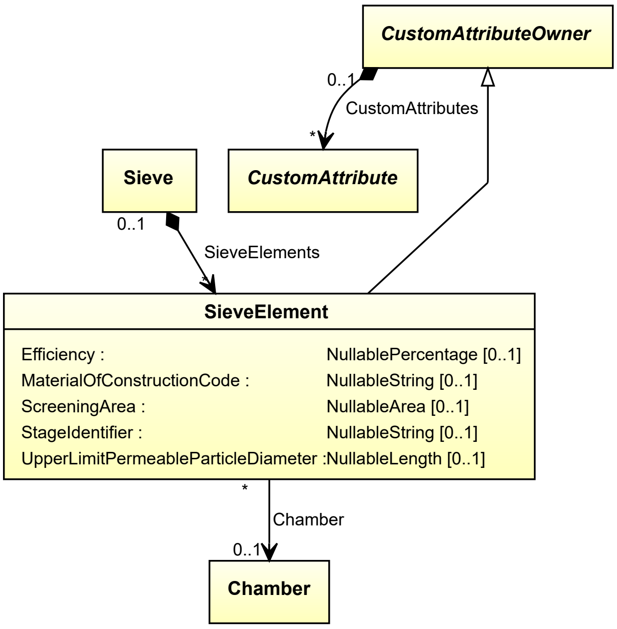
```
<Equipment
    ID="sieve1"
    ComponentClass="Sieve"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Sieve" ...>
  ...
  <Equipment
      ID="sieveElement1"
      ComponentClass="SieveElement"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/SieveElement" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.135. SieveElement

7.135.1 Overview

Class

A screening unit that is a component of a sieve.


CustomAttributeOwner
0..1
CustomAttributes
*
Sieve
CustomAttribute
0..1
SieveElements
SieveElement
Efficiency : NullablePercentage [0..1]
MaterialOfConstructionCode : NullableString [0..1]
ScreeningArea : NullableArea [0..1]
StageIdentifier : NullableString [0..1]
UpperLimitPermeableParticleDiameter :NullableLength [0..1]
*
Chamber
0..1
Chamber

Supertypes

- *CustomAttributeOwner*

Attributes (data)

Name	Multiplicity	Type
Efficiency	0..1	*NullablePercentage*
MaterialOfConstructionCode	0..1	*NullableString*
ScreeningArea	0..1	*NullableArea*
StageIdentifier	0..1	*NullableString*
UpperLimitPermeableParticleDiameter	0..1	*NullableLength*

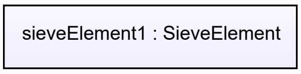
Attributes (reference)

Name	Multiplicity	Type
Chamber	0..1	*Chamber*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: SIEVE ELEMENT
ComponentClass: SieveElement
ComponentClassURI: http://sandbox.dexpi.org/rdl/SieveElement

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="sieveElement1"
    ComponentClass="SieveElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SieveElement" ...>
  ...
</Equipment>
```

7.135.2 Chamber

Attribute (reference)

The *Chamber* in which the *SieveElement* is located, if applicable. The Chamber must be a component of the same object as the *SieveElement*.

Multiplicity: 0..1

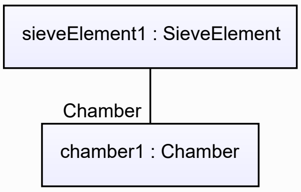
Type: *Chamber*

Opposite multiplicity: 0..*

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: "is located in"
Opposite association type: "is the location of"

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="sieveElement1"
    ComponentClass="SieveElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SieveElement" ...>
  ...
  <Association
    Type="is located in"
    ItemID="chamber1" />
  ...
<Equipment />
...
<Equipment
    ID="chamber1"
    ComponentClass="Chamber"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS903151421" ...>
  ...
  <Association
    Type="is the location of"
    ItemID="sieveElement1" />
  ...
<Equipment />
```

7.135.3 Efficiency

Attribute (data)

The efficiency of the *SieveElement*.

Multiplicity: 0..1

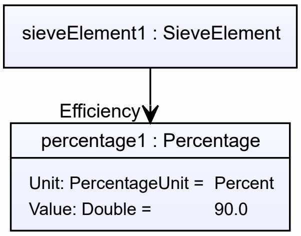
Type: *NullablePercentage*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: EFFICIENCY
Name: Efficiency
AttributeURI: http://data.posccaesar.org/rdl/RDS362654

Example

The instance sieveElement1 represents a *SieveElement* with an *Efficiency* of 90.0 ???.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="sieveElement1"
    ComponentClass="SieveElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SieveElement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="Efficiency"
      AttributeURI="http://data.posccaesar.org/rdl/RDS362654"
      Format="double"
      Value="90.0"
      Units="Percent"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1317959" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.135.4 MaterialOfConstructionCode

Attribute (data)

A code that gives the material of construction of the *SieveElement*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: MATERIAL OF CONSTRUCTION CODE ASSIGNMENT CLASS
Name: MaterialOfConstructionCodeAssignmentClass
AttributeURI: http://data.posccaesar.org/rdl/RDS1460719741

Example

“1.4306” (*String*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="sieveElement1"
    ComponentClass="SieveElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SieveElement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="MaterialOfConstructionCodeAssignmentClass"
      AttributeURI="http://data.posccaesar.org/rdl/RDS1460719741"
      Format="string"
      Value="1.4306" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.135.5 ScreeningArea

Attribute (data)

The filter area of the *SieveElement*.

Multiplicity: 0..1

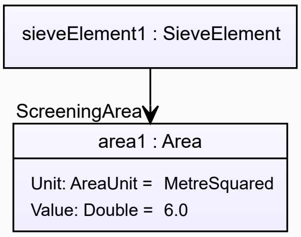
Type: *NullableArea*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: SCREENING AREA
Name: ScreeningArea
AttributeURI: http://sandbox.dexpi.org/rdl/ScreeningArea

Example

The instance sieveElement1 represents a *SieveElement* with a *ScreeningArea* of 6.0 m^2.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="sieveElement1"
    ComponentClass="SieveElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SieveElement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ScreeningArea"
      AttributeURI="http://sandbox.dexpi.org/rdl/ScreeningArea"
      Format="double"
      Value="6.0"
      Units="MetreSquared"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1358009" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.135.6 StageIdentifier

Attribute (data)

The stage identifier of the *SieveElement*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: STAGE IDENTIFIER ASSIGNMENT CLASS
Name: StageIdentifierAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/StageIdentifierAssignmentClass

Example

“s1” (*String*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="sieveElement1"
    ComponentClass="SieveElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SieveElement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="StageIdentifierAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/StageIdentifierAssignmentClass"
      Format="string"
      Value="s1" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.135.7 UpperLimitPermeableParticleDiameter

Attribute (data)

The maximum of the particle size passing through the *SieveElement*.

Multiplicity: 0..1

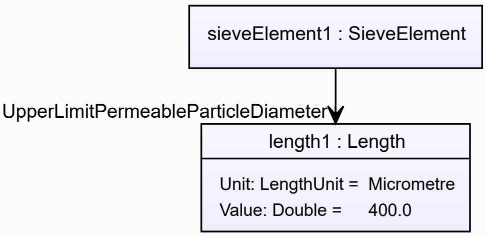
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT PERMEABLE PARTICLE DIAMETER
Name: UpperLimitPermeableParticleDiameter
AttributeURI: http://sandbox.dexpi.org/rdl/UpperLimitPermeableParticleDiameter

Example

The instance sieveElement1 represents a *SieveElement* with an *UpperLimitPermeableParticleDiameter* of 400.0 µm.


sieveElement1 : SieveElement
UpperLimitPermeableParticleDiameter
length1 : Length
Unit: LengthUnit = Micrometre
Value: Double = 400.0

Example: Implementation in Proteus Schema

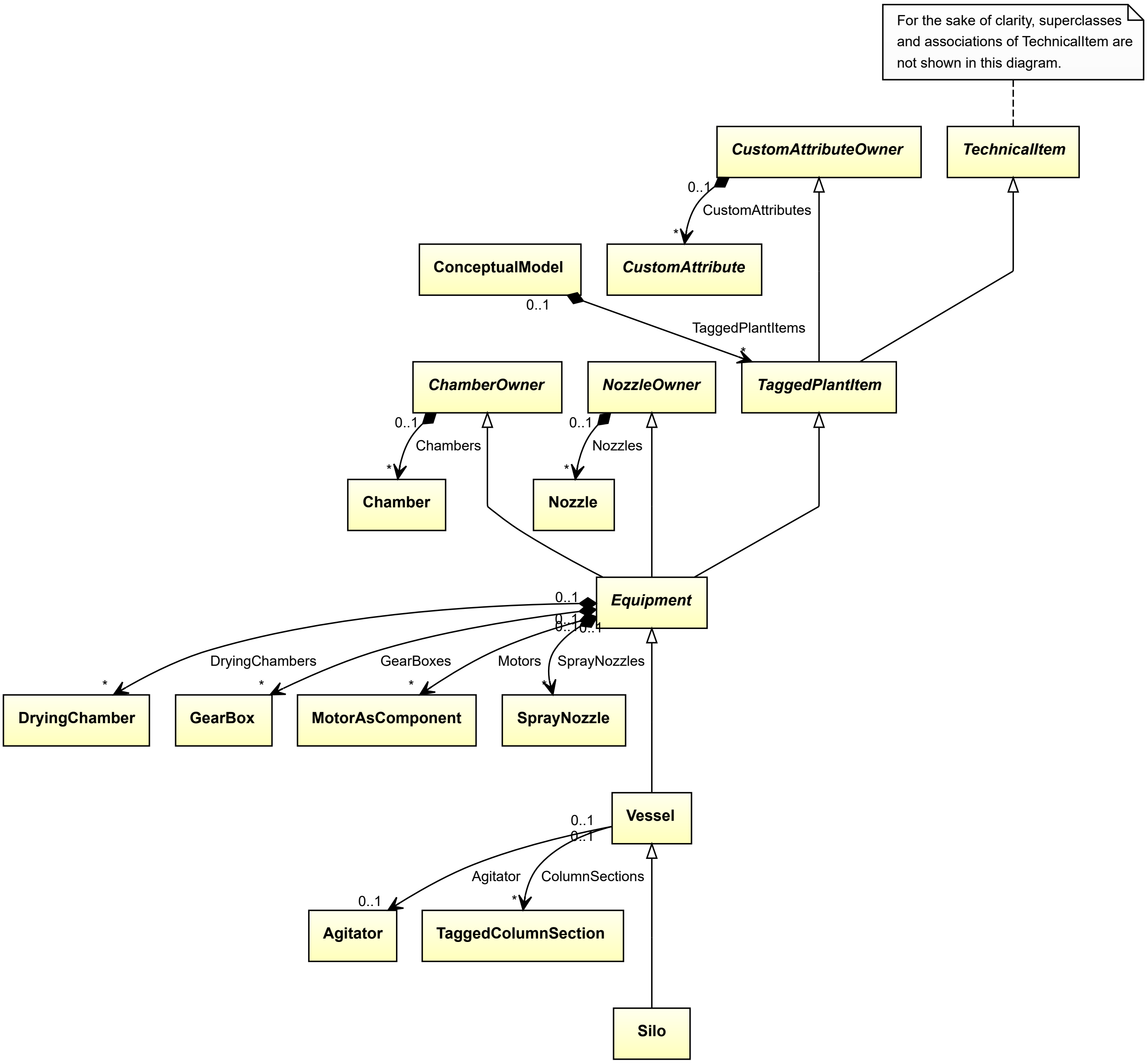
```
<Equipment
    ID="sieveElement1"
    ComponentClass="SieveElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SieveElement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="UpperLimitPermeableParticleDiameter"
      AttributeURI="http://sandbox.dexpi.org/rdl/UpperLimitPermeableParticleDiameter"
      Format="double"
      Value="400.0"
      Units="Micrometre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1351529" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.136. Silo

7.136.1 Overview

Class

A *Vessel* with a conical shape that is intended to store solids in bulk (from http://data.15926.org/rdl/RDS1022399).



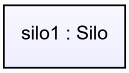
Supertypes

- *Vessel*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: SILO
ComponentClass: Silo
ComponentClassURI: http://data.posccaesar.org/rdl/RDS1022399

Example



Example: Implementation in Proteus Schema

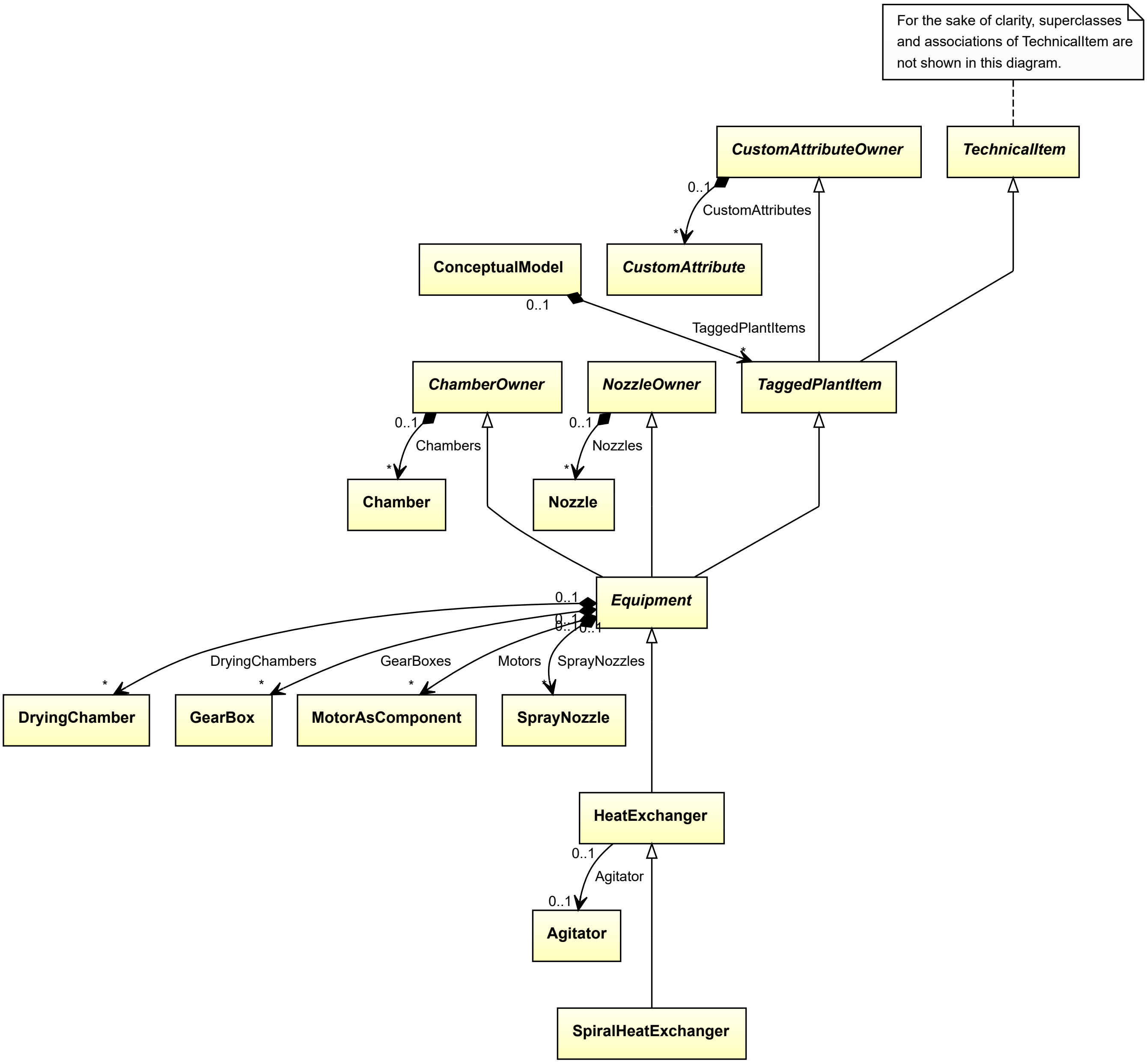
```
<Equipment
    ID="silo1"
    ComponentClass="Silo"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS1022399" ...>
  ...
</Equipment>
```

7.137. SpiralHeatExchanger

7.137.1 Overview

Class

A *HeatExchanger* in which a pair of plates is formed into a spiral.



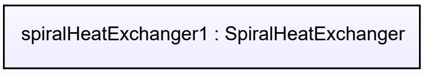
Supertypes

- *HeatExchanger*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: SPIRAL HEAT EXCHANGER
ComponentClass: SpiralHeatExchanger
ComponentClassURI: http://sandbox.dexpi.org/rdl/SpiralHeatExchanger

Example



Example: Implementation in Proteus Schema

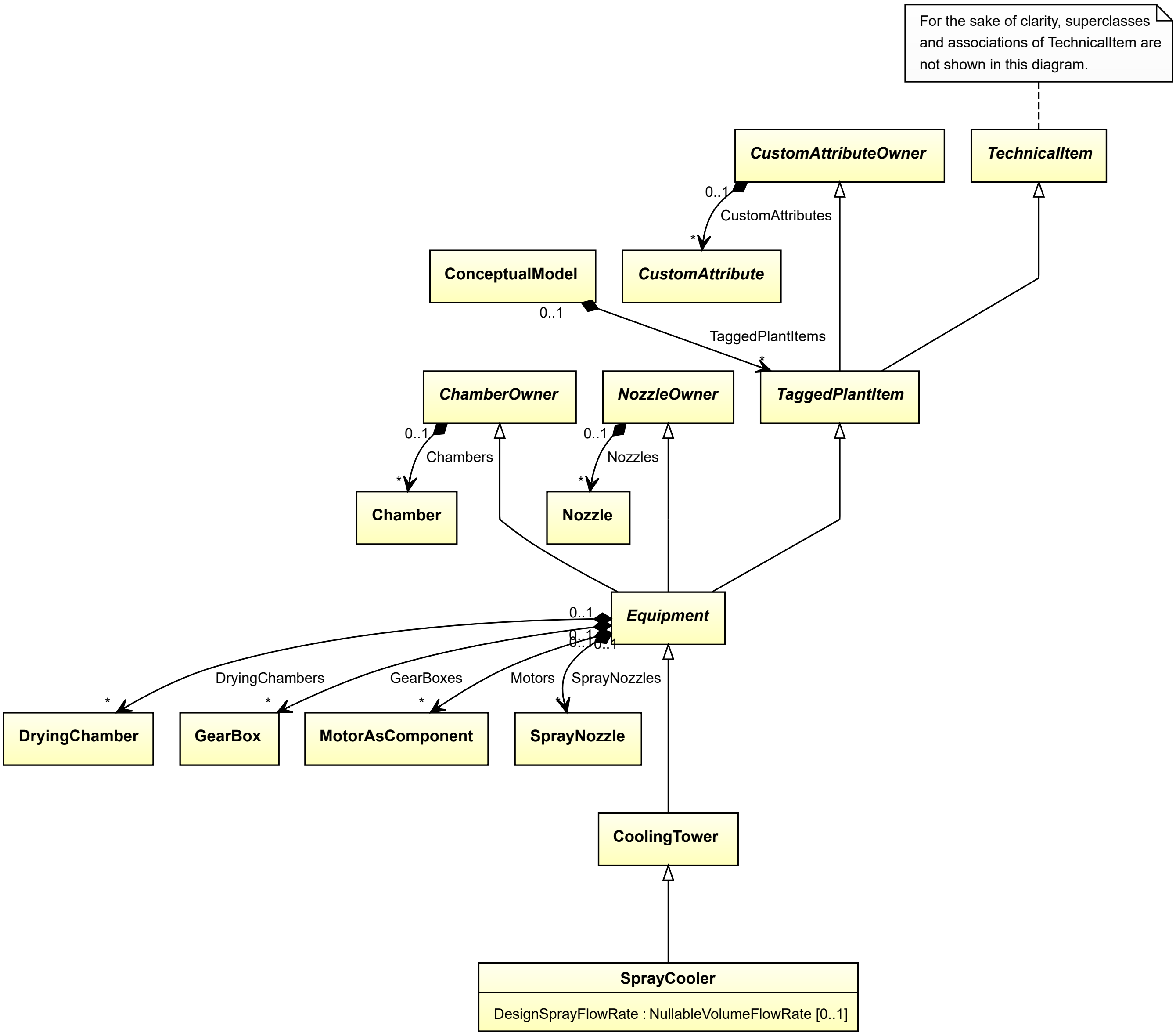
```
<Equipment
    ID="spiralHeatExchanger1"
    ComponentClass="SpiralHeatExchanger"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SpiralHeatExchanger" ...>
  ...
</Equipment>
```

7.138. SprayCooler

7.138.1 Overview

Class

A *CoolingTower* that is based on spraying a coolant on a heated surface to be cooled.



Supertypes

- *CoolingTower*

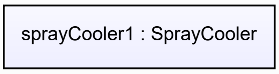
Attributes (data)

Name	Multiplicity	Type
DesignSprayFlowRate	0..1	*NullableVolumeFlowRate*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: SPRAY COOLER
ComponentClass: SprayCooler
ComponentClassURI: http://sandbox.dexpi.org/rdl/SprayCooler

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="sprayCooler1"
    ComponentClass="SprayCooler"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SprayCooler" ...>
  ...
</Equipment>
```

7.138.2 DesignSprayFlowRate

Attribute (data)

The spray volume flow rate for the motive fluid for which the *SprayCooler* is designed.

Multiplicity: 0..1

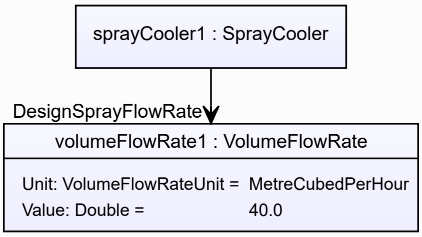
Type: *NullableVolumeFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SPRAY FLOW RATE
Name: DesignSprayFlowRate
AttributeURI: http://sandbox.dexpi.org/rdl/DesignSprayFlowRate

Example

The instance sprayCooler1 represents a *SprayCooler* with a *DesignSprayFlowRate* of 40.0 m^3/h.


sprayCooler1 : SprayCooler
DesignSprayFlowRate
volumeFlowRate1 : VolumeFlowRate
Unit: VolumeFlowRateUnit = MetreCubedPerHour
Value: Double = 40.0

Example: Implementation in Proteus Schema

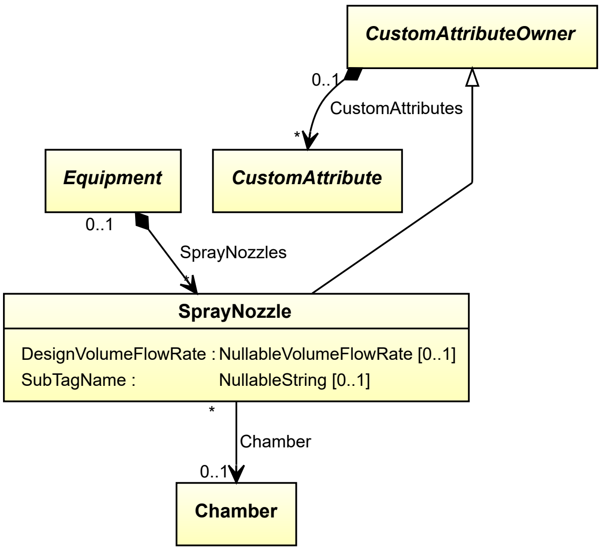
```
<Equipment
    ID="sprayCooler1"
    ComponentClass="SprayCooler"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SprayCooler" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignSprayFlowRate"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignSprayFlowRate"
      Format="double"
      Value="40.0"
      Units="MetreCubedPerHour"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1321064" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.139. SprayNozzle

7.139.1 Overview

Class

A nozzle where liquid is introduced under pressure (from http://data.posccaesar.org/rdl/RDS5855670).



Supertypes

- *CustomAttributeOwner*

Attributes (data)

Name	Multiplicity	Type
DesignVolumeFlowRate	0..1	*NullableVolumeFlowRate*
SubTagName	0..1	*NullableString*

Attributes (reference)

Name	Multiplicity	Type
Chamber	0..1	*Chamber*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: SPRAY NOZZLE
ComponentClass: SprayNozzle
ComponentClassURI: http://data.posccaesar.org/rdl/RDS5855670

Example

sprayNozzle1 : SprayNozzle

Example: Implementation in Proteus Schema

```
<Equipment
    ID="sprayNozzle1"
    ComponentClass="SprayNozzle"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS5855670" ...>
  ...
</Equipment>
```

7.139.2 Chamber

Attribute (reference)

The *Chamber* in which the *SprayNozzle* is located, if applicable. The Chamber must be a component of the same object as the *SprayNozzle*.

Multiplicity: 0..1

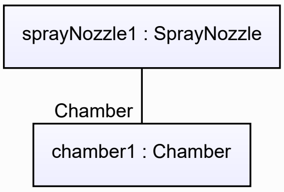
Type: *Chamber*

Opposite multiplicity: 0..*

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: `"is located in"`
Opposite association type: `"is the location of"`

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="sprayNozzle1"
    ComponentClass="SprayNozzle"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS5855670" ...>
  ...
  <Association
    Type="is located in"
    ItemID="chamber1" />
  ...
<Equipment />
...
<Equipment
    ID="chamber1"
    ComponentClass="Chamber"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS903151421" ...>
  ...
  <Association
    Type="is the location of"
    ItemID="sprayNozzle1" />
  ...
<Equipment />
```

7.139.3 DesignVolumeFlowRate

Attribute (data)

The volume flow rate for which the *SprayNozzle* is designed.

Multiplicity: 0..1

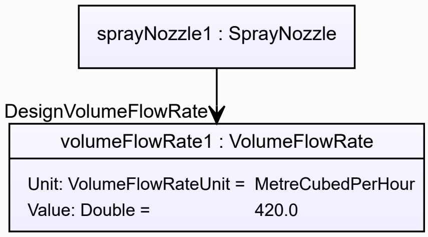
Type: *NullableVolumeFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN VOLUME FLOW RATE
Name: DesignVolumeFlowRate
AttributeURI: http://data.posccaesar.org/rdl/RDS14286227

Example

The instance sprayNozzle1 represents a *SprayNozzle* with a *DesignVolumeFlowRate* of 420.0 m^3/h.


sprayNozzle1 : SprayNozzle
DesignVolumeFlowRate
volumeFlowRate1 : VolumeFlowRate
Unit: VolumeFlowRateUnit = MetreCubedPerHour
Value: Double = 420.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="sprayNozzle1"
    ComponentClass="SprayNozzle"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS5855670" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignVolumeFlowRate"
      AttributeURI="http://data.posccaesar.org/rdl/RDS14286227"
      Format="double"
      Value="420.0"
      Units="MetreCubedPerHour"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1321064" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.139.4 SubTagName

Attribute (data)

The sub tag name of the *SprayNozzle*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: SUB TAG NAME ASSIGNMENT CLASS
Name: SubTagNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/SubTagNameAssignmentClass

Example

“ST1” (*String*)

Example: Implementation in Proteus Schema

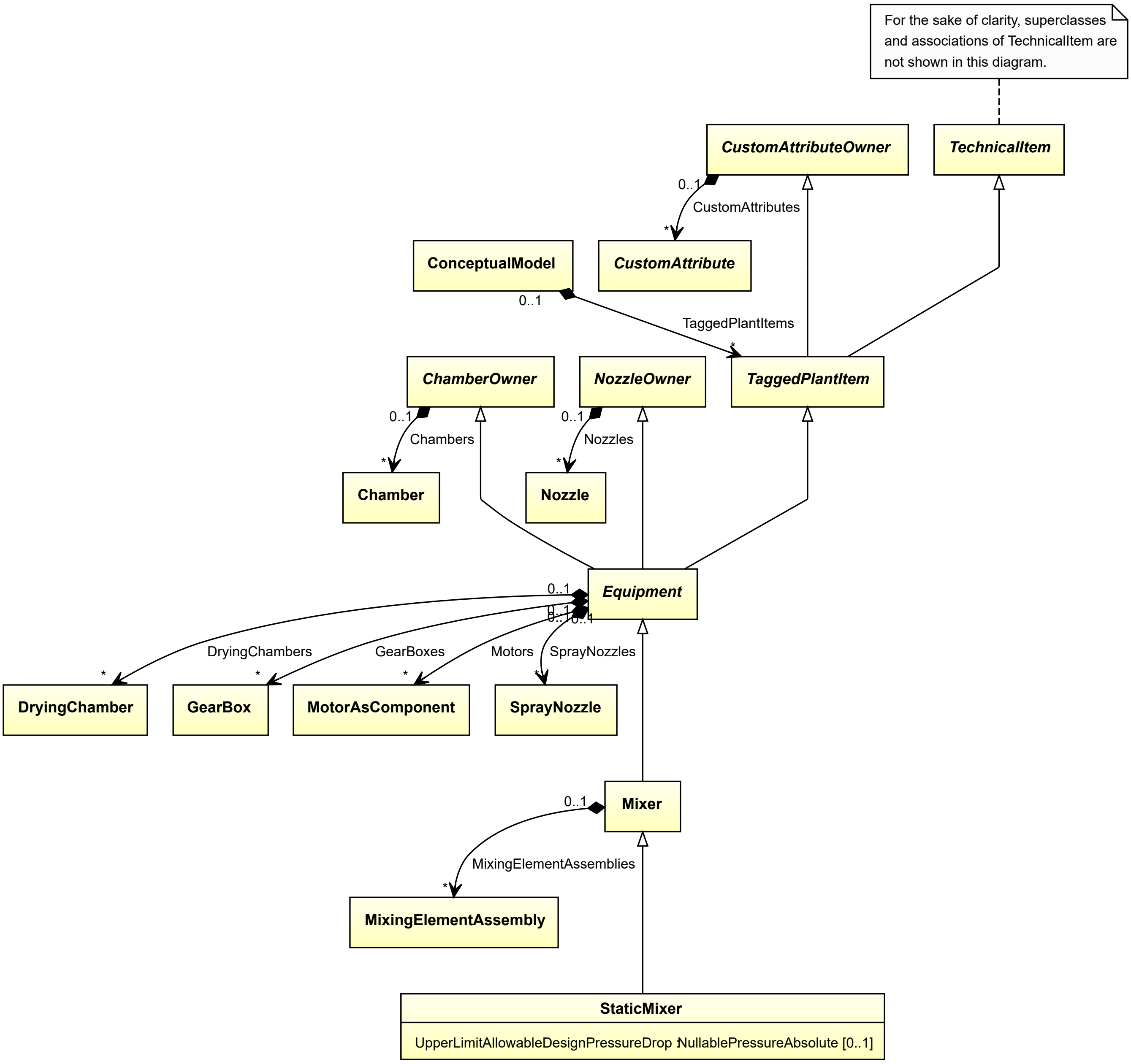
```
<Equipment
    ID="sprayNozzle1"
    ComponentClass="SprayNozzle"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS5855670" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="SubTagNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/SubTagNameAssignmentClass"
      Format="string"
      Value="ST1" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.140. StaticMixer

7.140.1 Overview

Class

A physical object that is intended to mix fluid by means of diverging the flow with static obstacles or by increasing locally the velocity.



Supertypes

- *Mixer*

Attributes (data)

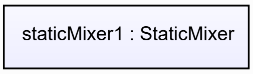
Name	Multiplicity	Type
UpperLimitAllowableDesignPressureDrop	0..1	*NullablePressureAbsolute*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: STATIC MIXER

ComponentClass: StaticMixer
ComponentClassURI: http://data.posccaesar.org/rdl/RDS1016684

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="staticMixer1"
    ComponentClass="StaticMixer"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS1016684" ...>
  ...
</Equipment>
```

7.140.2 UpperLimitAllowableDesignPressureDrop

Attribute (data)

The upper limit for the pressure drop for which the *StaticMixer* is designed.

Multiplicity: 0..1

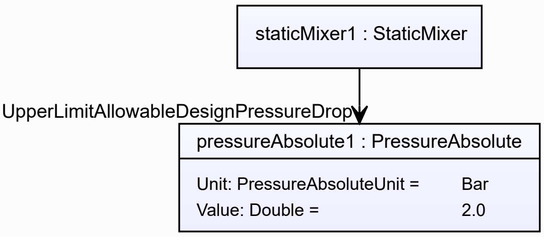
Type: *NullablePressureAbsolute*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: UPPER LIMIT ALLOWABLE DESIGN PRESSURE DROP
Name: UpperLimitAllowableDesignPressureDrop
AttributeURI: http://sandbox.dexpi.org/rdl/UpperLimitAllowableDesignPressureDrop

Example

The instance staticMixer1 represents a *StaticMixer* with an *UpperLimitAllowableDesignPressureDrop* of 2.0 bar.


staticMixer1 : StaticMixer
UpperLimitAllowableDesignPressureDrop
pressureAbsolute1 : PressureAbsolute
Unit: PressureAbsoluteUnit = Bar
Value: Double = 2.0

Example: Implementation in Proteus Schema

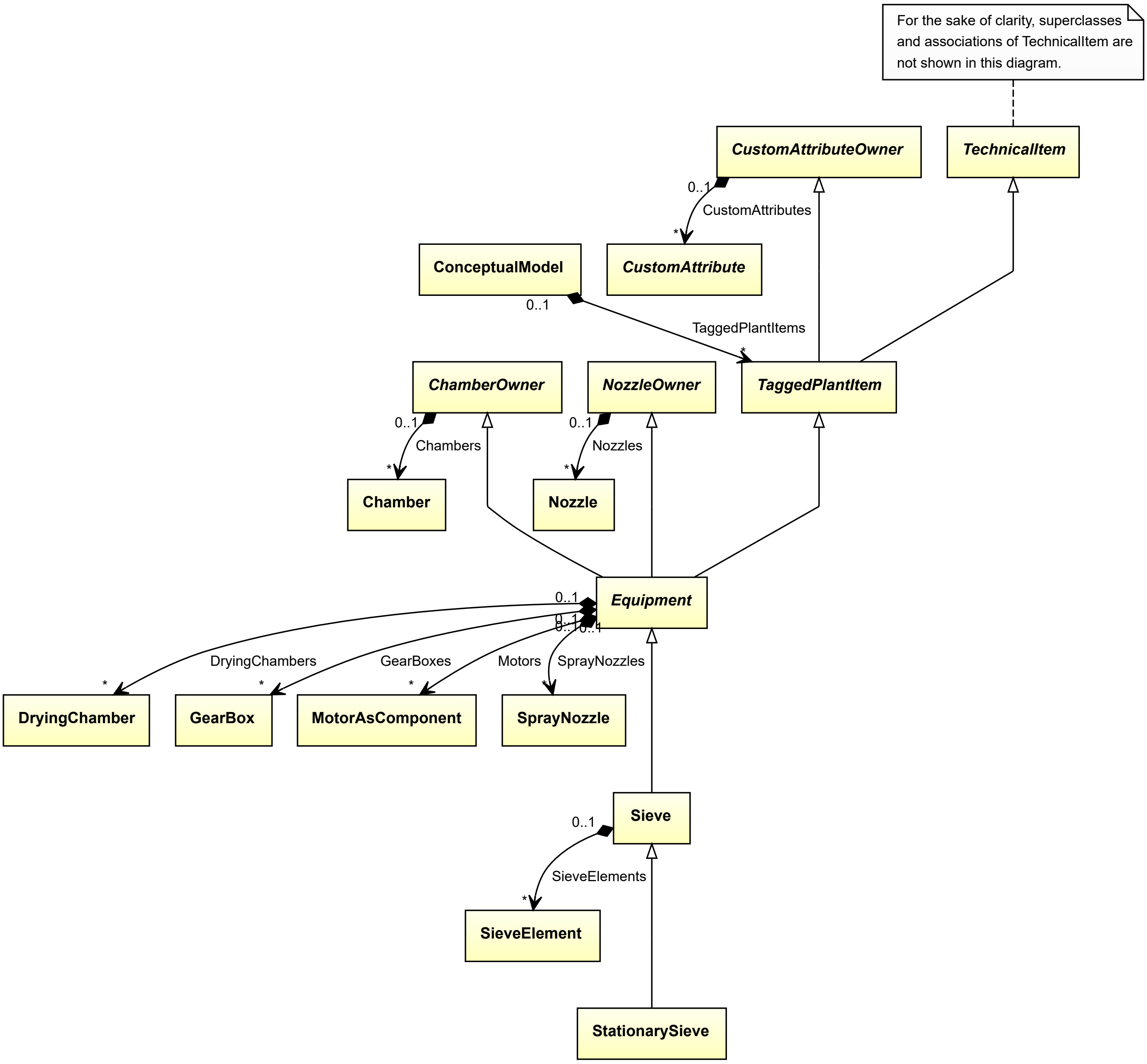
```
<Equipment
    ID="staticMixer1"
    ComponentClass="StaticMixer"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS1016684" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="UpperLimitAllowableDesignPressureDrop"
      AttributeURI="http://sandbox.dexpi.org/rdl/UpperLimitAllowableDesignPressureDrop"
      Format="double"
      Value="2.0"
      Units="Bar"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1314539" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.141. StationarySieve

7.141.1 Overview

Class

A *Sieve* consisting of rakes or sieves, that, during operation, remains in a fixed position (from http://data.15926.org/rdl/RDS2226669).



Supertypes

- *Sieve*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: STATIONARY SCREEN
ComponentClass: StationaryScreen
ComponentClassURI: http://sandbox.dexpi.org/rdl/StationaryScreen

Example

stationarySieve1 : StationarySieve

Example: Implementation in Proteus Schema

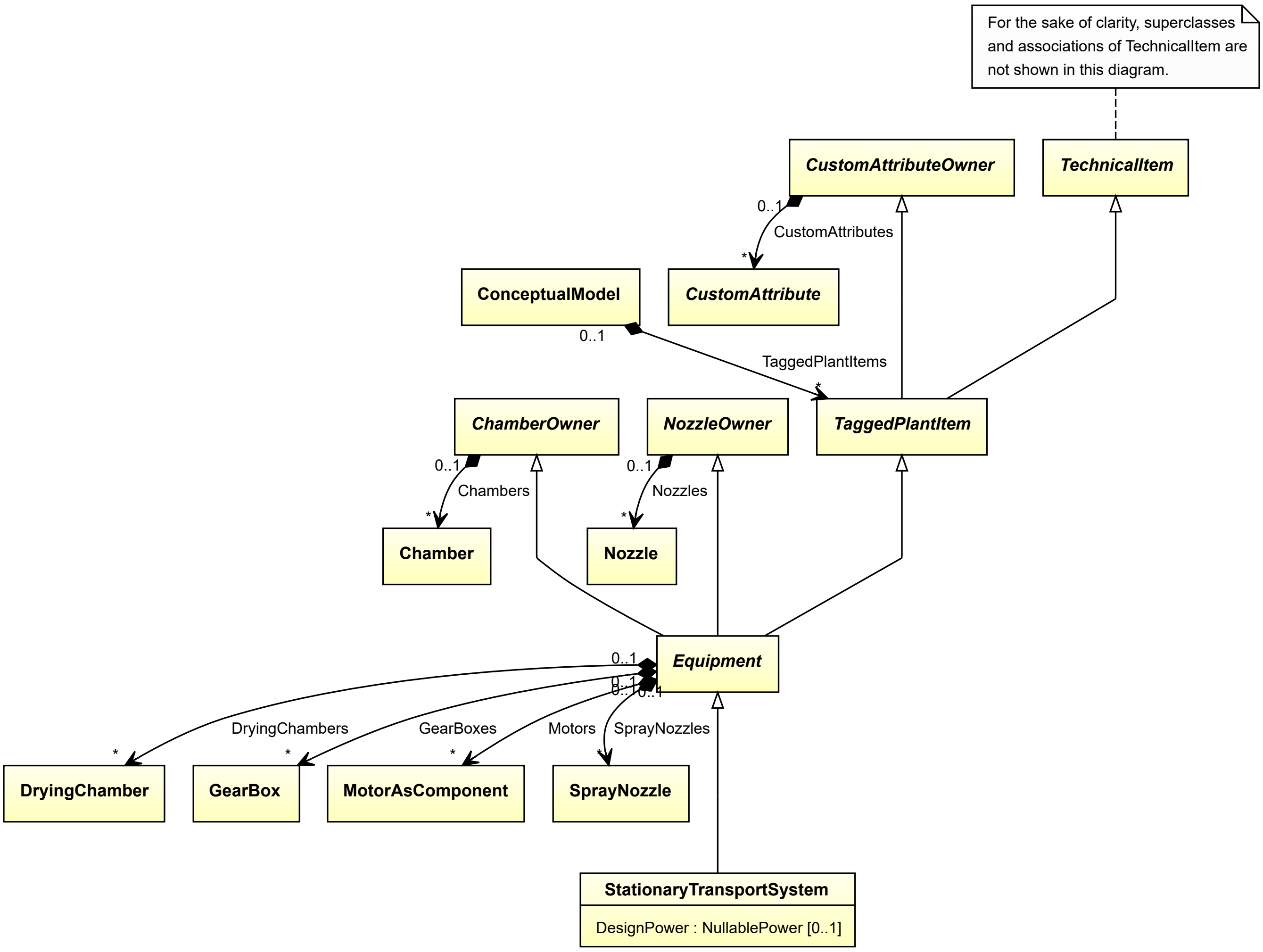
```
<Equipment
    ID="stationarySieve1"
    ComponentClass="StationaryScreen"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/StationaryScreen" ...>
  ...
</Equipment>
```

7.142. StationaryTransportSystem

7.142.1 Overview

Class

A transport system that is intended to transport, store or load/unload material and that, as a whole, remains in one place.



Supertypes

- *Equipment*

Subtypes

- *Conveyor*
- *CustomStationaryTransportSystem*
- *Lift*
- *LoadingUnloadingSystem*

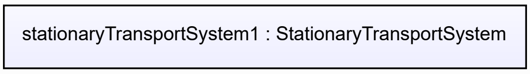
Attributes (data)

Name	Multiplicity	Type
DesignPower	0..1	*NullablePower*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: STATIONARY TRANSPORT SYSTEM
ComponentClass: StationaryTransportSystem
ComponentClassURI: http://sandbox.dexpi.org/rdl/StationaryTransportSystem

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="stationaryTransportSystem1"
    ComponentClass="StationaryTransportSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/StationaryTransportSystem" ...>
  ...
</Equipment>
```

7.142.2 DesignPower

Attribute (data)

The power for which the *StationaryTransportSystem* is designed.

Multiplicity: 0..1

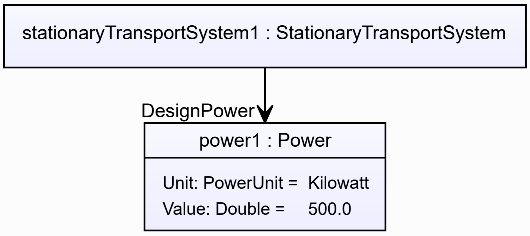
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN POWER
Name: DesignPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignPower

Example

The instance stationaryTransportSystem1 represents a *StationaryTransportSystem* with a *DesignPower* of 500.0 kW.


stationaryTransportSystem1 : StationaryTransportSystem
DesignPower
power1 : Power
Unit: PowerUnit = Kilowatt
Value: Double = 500.0

Example: Implementation in Proteus Schema

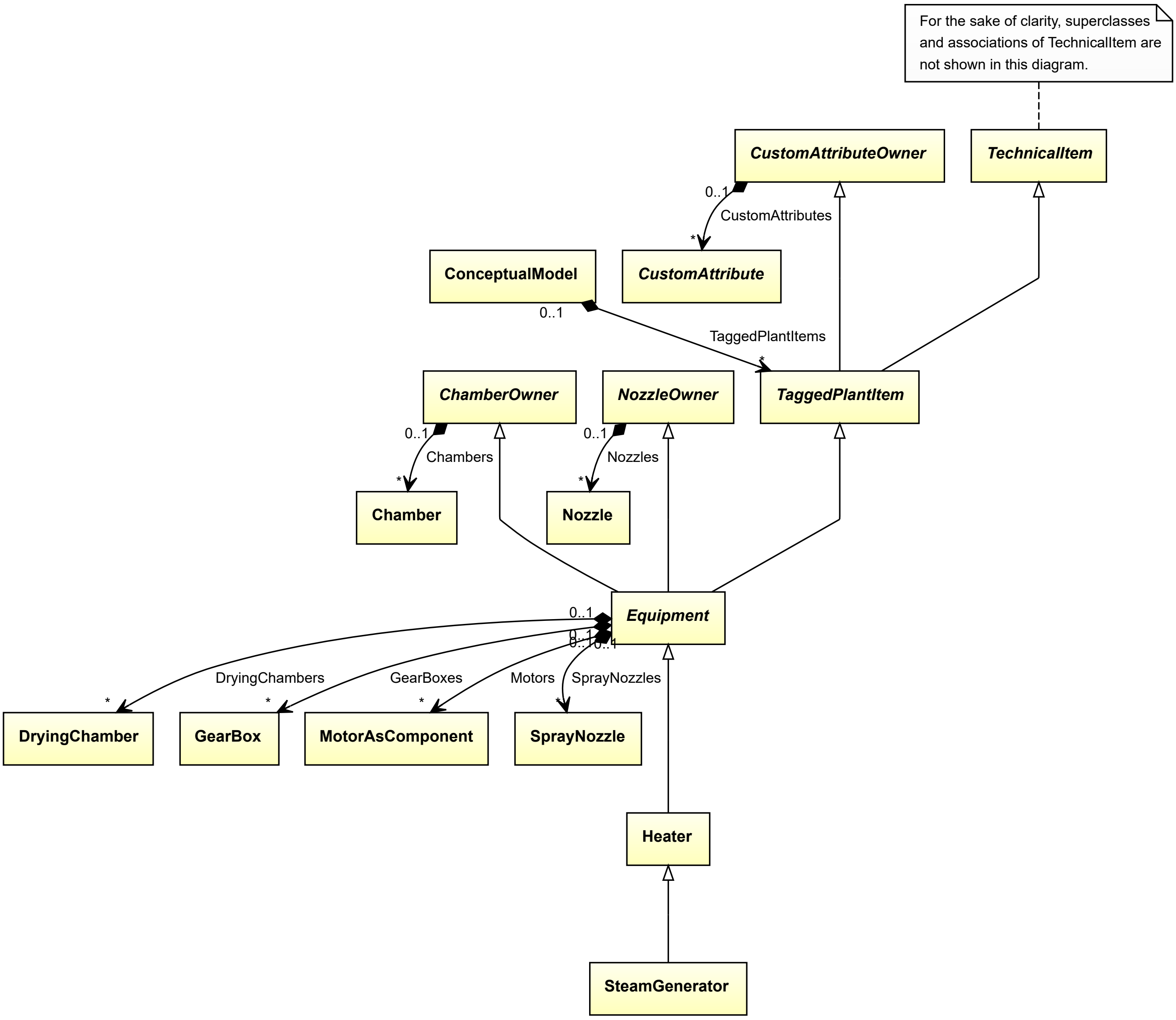
```
<Equipment
    ID="stationaryTransportSystem1"
    ComponentClass="StationaryTransportSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/StationaryTransportSystem" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignPower"
      Format="double"
      Value="500.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.143. SteamGenerator

7.143.1 Overview

Class

A boiler that is intended to generate steam (from http://data.posccaesar.org/rdl/RDS13306207).



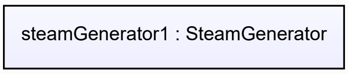
Supertypes

- *Heater*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: STEAM GENERATOR
ComponentClass: SteamGenerator
ComponentClassURI: http://data.posccaesar.org/rdl/RDS13306207

Example



Example: Implementation in Proteus Schema

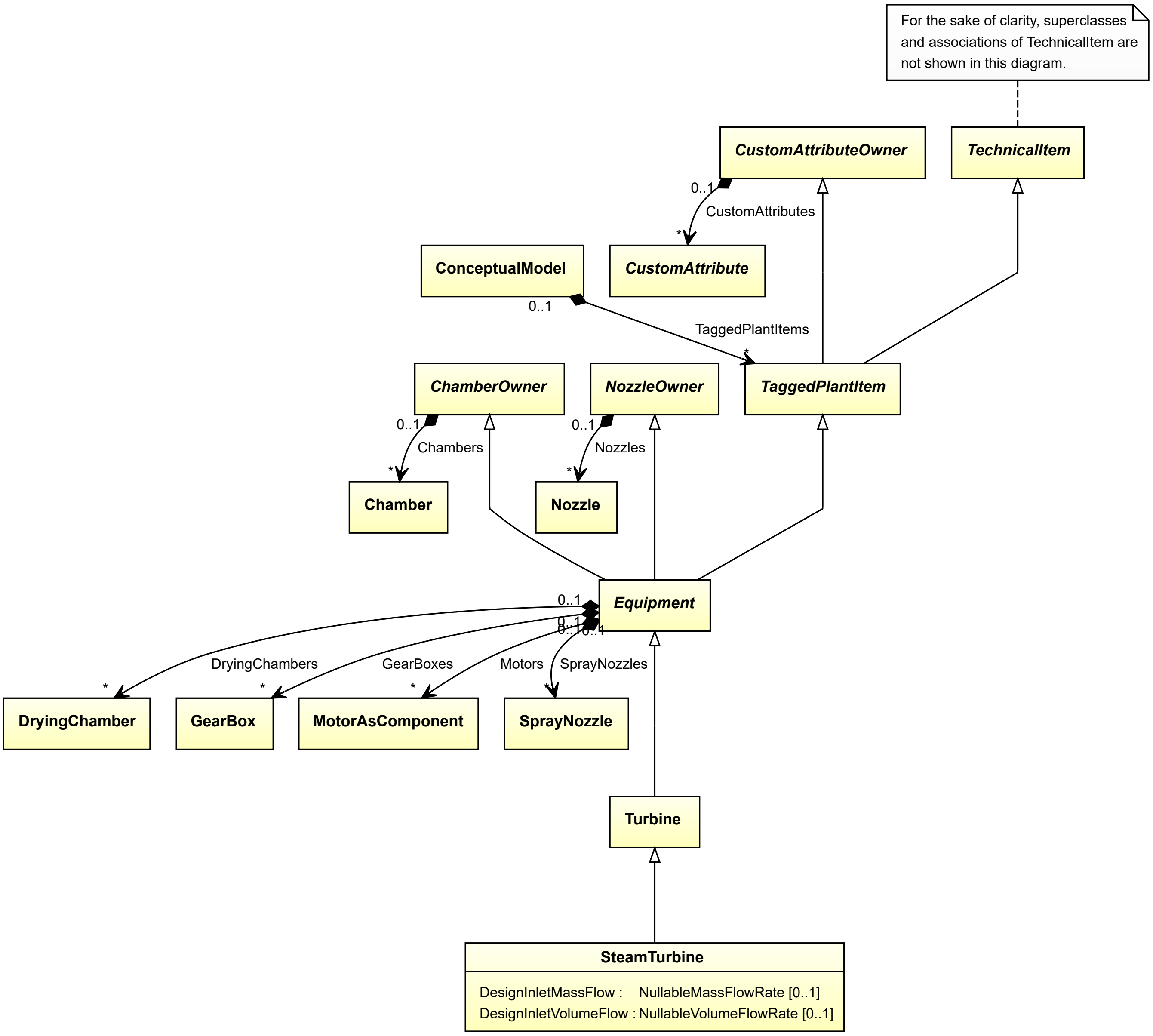
```
<Equipment
    ID="steamGenerator1"
    ComponentClass="SteamGenerator"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS13306207" ...>
  ...
</Equipment>
```

7.144. SteamTurbine

7.144.1 Overview

Class

A turbine that is a heat engine in which energy of steam is transformed into kinetic energy by means of expansion through nozzles and the kinetic energy of the resulting jet is in turn converted into force doing work on rings of blading mounted on a rotating shaft (from http://data.posccaesar.org/rdl/RDS416744).



Supertypes

- *Turbine*

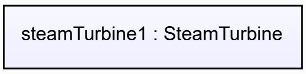
Attributes (data)

Name	Multiplicity	Type
DesignInletMassFlow	0..1	*NullableMassFlowRate*
DesignInletVolumeFlow	0..1	*NullableVolumeFlowRate*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: STEAM TURBINE
ComponentClass: SteamTurbine
ComponentClassURI: http://data.posccaesar.org/rdl/RDS416744

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="steamTurbine1"
    ComponentClass="SteamTurbine"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS416744" ...>
  ...
</Equipment>
```

7.144.2 DesignInletMassFlow

Attribute (data)

The inlet mass flow for which the *SteamTurbine* is designed.

Multiplicity: 0..1

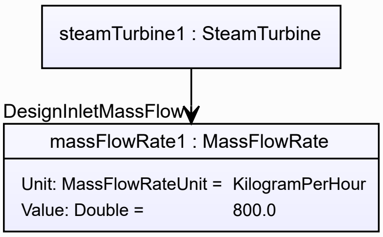
Type: *NullableMassFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN INLET MASS FLOW
Name: DesignInletMassFlow
AttributeURI: http://sandbox.dexpi.org/rdl/DesignInletMassFlow

Example

The instance steamTurbine1 represents a *SteamTurbine* with a *DesignInletMassFlow* of 800.0 kg/h.


steamTurbine1 : SteamTurbine
DesignInletMassFlow
massFlowRate1 : MassFlowRate
Unit: MassFlowRateUnit = KilogramPerHour
Value: Double = 800.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="steamTurbine1"
    ComponentClass="SteamTurbine"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS416744" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignInletMassFlow"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignInletMassFlow"
      Format="double"
      Value="800.0"
      Units="KilogramPerHour"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1329344" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.144.3 DesignInletVolumeFlow

Attribute (data)

The inlet volume flow for which the *SteamTurbine* is designed.

Multiplicity: 0..1

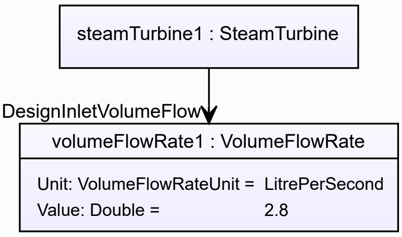
Type: *NullableVolumeFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN INLET VOLUME FLOW
Name: DesignInletVolumeFlow
AttributeURI: http://sandbox.dexpi.org/rdl/DesignInletVolumeFlow

Example

The instance steamTurbine1 represents a *SteamTurbine* with a *DesignInletVolumeFlow* of 2.8 l/s.


steamTurbine1 : SteamTurbine
DesignInletVolumeFlow
volumeFlowRate1 : VolumeFlowRate
Unit: VolumeFlowRateUnit = LitrePerSecond
Value: Double = 2.8

Example: Implementation in Proteus Schema

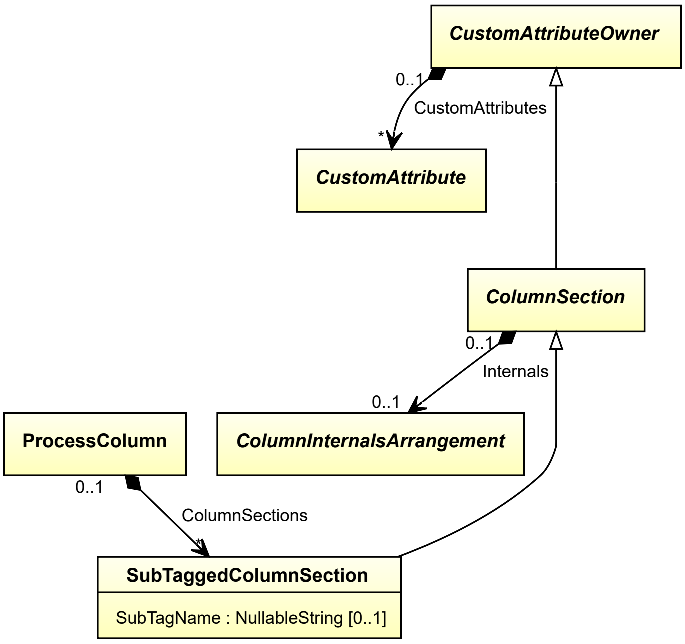
```
<Equipment
    ID="steamTurbine1"
    ComponentClass="SteamTurbine"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS416744" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignInletVolumeFlow"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignInletVolumeFlow"
      Format="double"
      Value="2.8"
      Units="LitrePerSecond"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1331369" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.145. SubTaggedColumnSection

7.145.1 Overview

Class

A sub tagged column section.



Supertypes

- *ColumnSection*

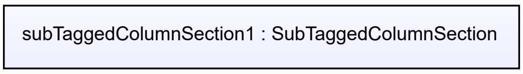
Attributes (data)

Name	Multiplicity	Type
SubTagName	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: COLUMN SECTION
ComponentClass: ColumnSection
ComponentClassURI: http://sandbox.dexpi.org/rdl/ColumnSection

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="subTaggedColumnSection1"
    ComponentClass="ColumnSection"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ColumnSection" ...>
  ...
</Equipment>
```

7.145.2 SubTagName

Attribute (data)

The sub tag name of the *SubTaggedColumnSection*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: SUB TAG NAME ASSIGNMENT CLASS
Name: SubTagNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/SubTagNameAssignmentClass

Example

“ST1” (*String*)

Example: Implementation in Proteus Schema

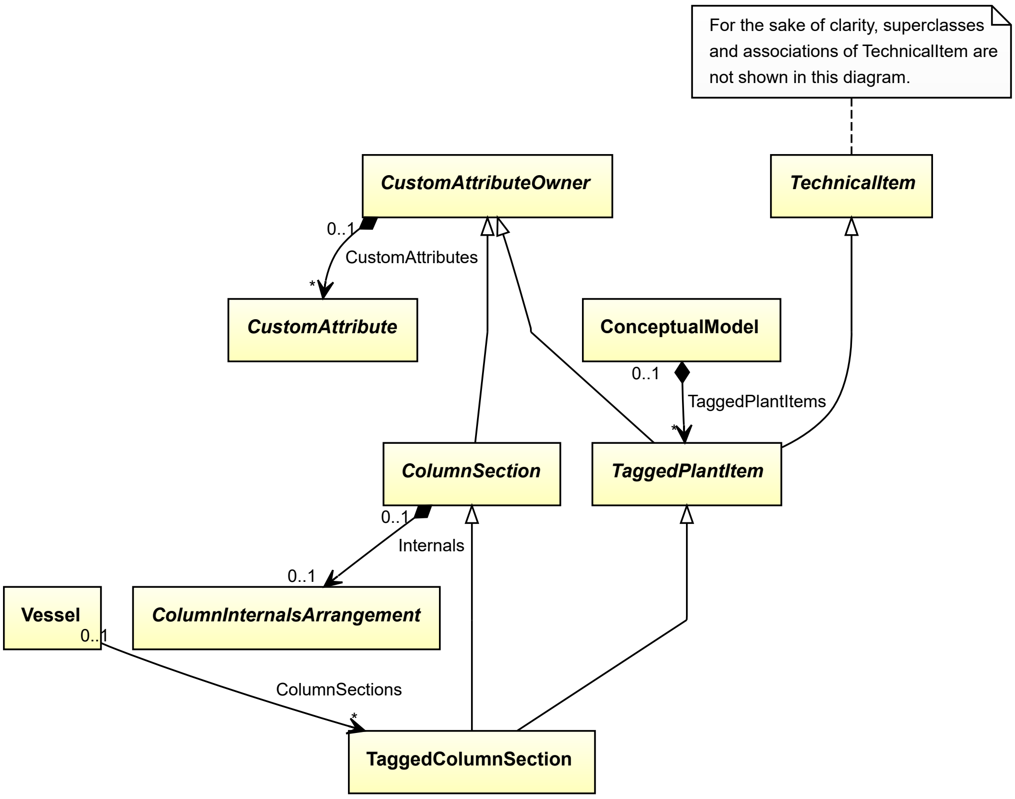
```
<Equipment
    ID="subTaggedColumnSection1"
    ComponentClass="ColumnSection"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ColumnSection" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="SubTagNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/SubTagNameAssignmentClass"
      Format="string"
      Value="ST1" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.146. TaggedColumnSection

7.146.1 Overview

Class

A fully tagged column section.



Supertypes

- *ColumnSection*
- *TaggedPlantItem*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: COLUMN SECTION
ComponentClass: ColumnSection
ComponentClassURI: http://sandbox.dexpi.org/rdl/ColumnSection

Example

taggedColumnSection1 : TaggedColumnSection

Example: Implementation in Proteus Schema

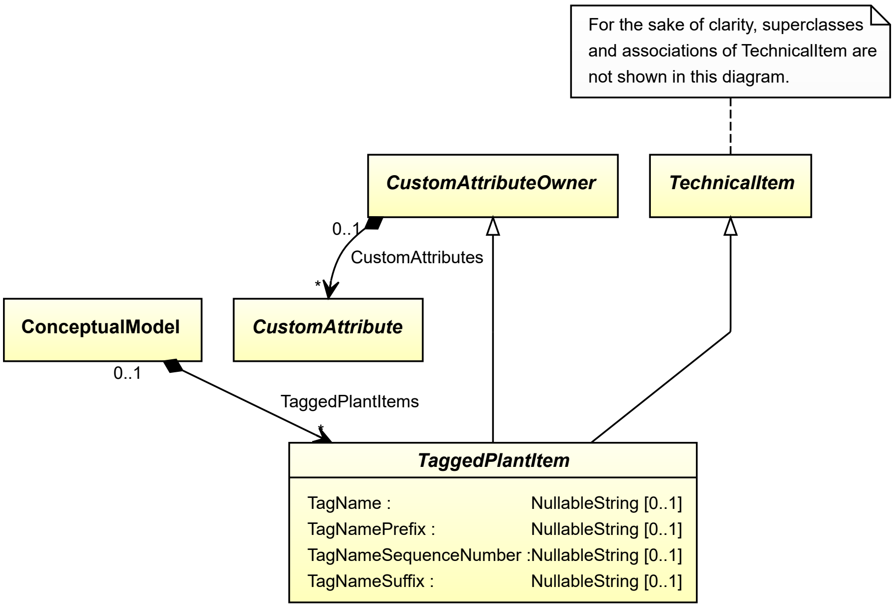
```
<Equipment
    ID="taggedColumnSection1"
    ComponentClass="ColumnSection"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ColumnSection" ...>
  ...
</Equipment>
```

7.147. TaggedPlantItem

7.147.1 Overview

Abstract class

A fully tagged item in a plant.


For the sake of clarity, superclasses and associations of TechnicalItem are not shown in this diagram.
CustomAttributeOwner
TechnicalItem
0..1
CustomAttributes
*
ConceptualModel
CustomAttribute
0..1
TaggedPlantItems
*
TaggedPlantItem
TagName : NullableString [0..1]
TagNamePrefix : NullableString [0..1]
TagNameSequenceNumber : NullableString [0..1]
TagNameSuffix : NullableString [0..1]

Supertypes

- *CustomAttributeOwner*
- *TechnicalItem*

Subtypes

- *Equipment*
- *TaggedColumnSection*

Attributes (data)

Name	Multiplicity	Type
TagName	0..1	*NullableString*
TagNamePrefix	0..1	*NullableString*
TagNameSequenceNumber	0..1	*NullableString*
TagNameSuffix	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*. As *TaggedPlantItem* is abstract, there is no RDL reference for the class itself; the RDL reference depends on the concrete subclass.
Tag: `<Equipment>`
ComponentClass: *depending on subclass*
ComponentClassURI: *depending on subclass*

Example

As *TaggedPlantItem* is abstract, we consider *Vessel* as an arbitrary concrete subclass.

Example: Implementation in Proteus Schema

```
<Equipment
    ID="vessel1"
    ComponentClass="Vessel"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS414674" ...>
  ...
</Equipment>
```

7.147.2 TagName

Attribute (data)

The tag number of the *TaggedPlantItem*. See also *TagNamePrefix*, *TagNameSequenceNumber*, and *TagNameSuffix*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: TAG NAME ASSIGNMENT CLASS
Name: TagNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/TagNameAssignmentClass

Example

As the owning class *TaggedPlantItem* is abstract, we consider *Vessel* as an arbitrary concrete subclass.
“P4714-A” (*String*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="vessel1"
    ComponentClass="Vessel"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS414674" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="TagNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/TagNameAssignmentClass"
      Format="string"
      Value="P4714-A" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.147.3 TagNamePrefix

Attribute (data)

The prefix part of the tag number of the *TaggedPlantItem*. For example, the prefix of the tag number “P4714-A” is “P”. The prefix often indicates the type of the equipment item, e.g., “P” can indicate a pump. See also *TagName*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: TAG NAME PREFIX ASSIGNMENT CLASS
Name: TagNamePrefixAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/TagNamePrefixAssignmentClass

Example

As the owning class *TaggedPlantItem* is abstract, we consider *Vessel* as an arbitrary concrete subclass.
“P” (*String*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="vessel1"
    ComponentClass="Vessel"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS414674" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="TagNamePrefixAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/TagNamePrefixAssignmentClass"
      Format="string"
      Value="P" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.147.4 TagNameSequenceNumber

Attribute (data)

The sequence number part of the tag number of the *TaggedPlantItem*. For example, the sequence number of the tag number “P4714-A” is “4714”.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: TAG NAME SEQUENCE NUMBER ASSIGNMENT CLASS
Name: TagNameSequenceNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/TagNameSequenceNumberAssignmentClass

Example

As the owning class *TaggedPlantItem* is abstract, we consider *Vessel* as an arbitrary concrete subclass.
“4714” (*String*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="vessel1"
    ComponentClass="Vessel"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS414674" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="TagNameSequenceNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/TagNameSequenceNumberAssignmentClass"
      Format="string"
      Value="4714" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.147.5 TagNameSuffix

Attribute (data)

The suffix part of the tag number of an *TaggedPlantItem* item. For example, the suffix of the tag number “P4714-A” is “A”.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: TAG NAME SUFFIX ASSIGNMENT CLASS
Name: TagNameSuffixAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/TagNameSuffixAssignmentClass

Example

As the owning class *TaggedPlantItem* is abstract, we consider *Vessel* as an arbitrary concrete subclass.
“A” (*String*)

Example: Implementation in Proteus Schema

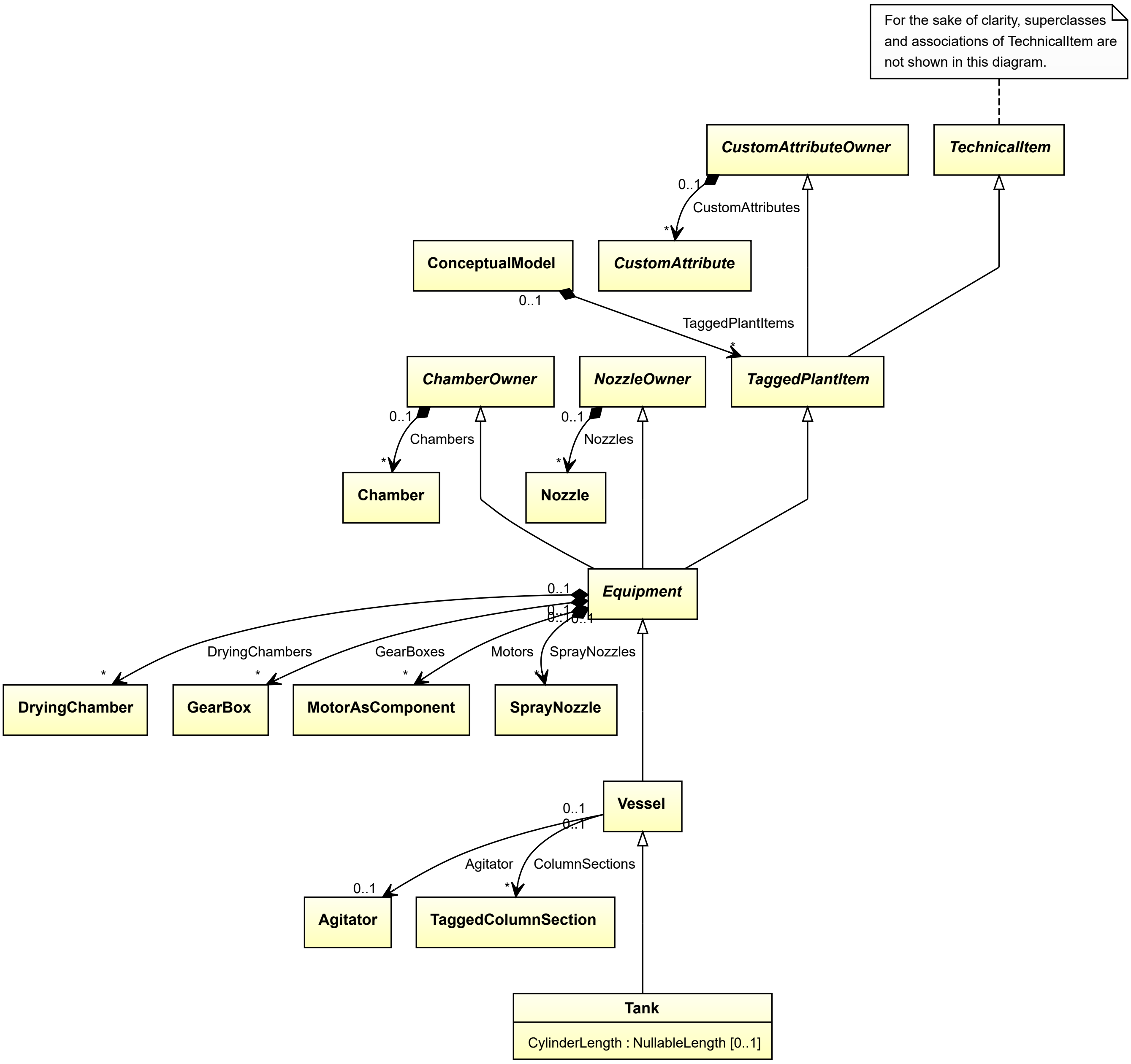
```
<Equipment
    ID="vessel1"
    ComponentClass="Vessel"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS414674" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="TagNameSuffixAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/TagNameSuffixAssignmentClass"
      Format="string"
      Value="A" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.148. Tank

7.148.1 Overview

Class

A vessel intended to contain fluid for storage. Typically a receiving or collecting function for further distribution. Typically with a vertical and cylindrical or square shape and a flat or conical bottom (from http://data.posccaesar.org/rdl/RDS445139).



For the sake of clarity, superclasses and associations of TechnicalItem are not shown in this diagram.

Supertypes

- *Vessel*

Attributes (data)

Name	Multiplicity	Type
CylinderLength	0..1	*NullableLength*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: TANK
ComponentClass: Tank
ComponentClassURI: http://data.posccaesar.org/rdl/RDS445139

Example

Example: Implementation in Proteus Schema

```
<Equipment
    ID="tank1"
    ComponentClass="Tank"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS445139" ...>
  ...
</Equipment>
```

7.148.2 CylinderLength

Attribute (data)

The cylinder length of the *Tank*.

Multiplicity: 0..1

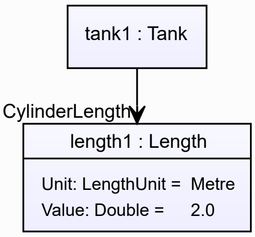
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: CYLINDER LENGTH
Name: CylinderLength
AttributeURI: http://sandbox.dexpi.org/rdl/CylinderLength

Example

The instance tank1 represents a *Tank* with a *CylinderLength* of 2.0 m.



Example: Implementation in Proteus Schema

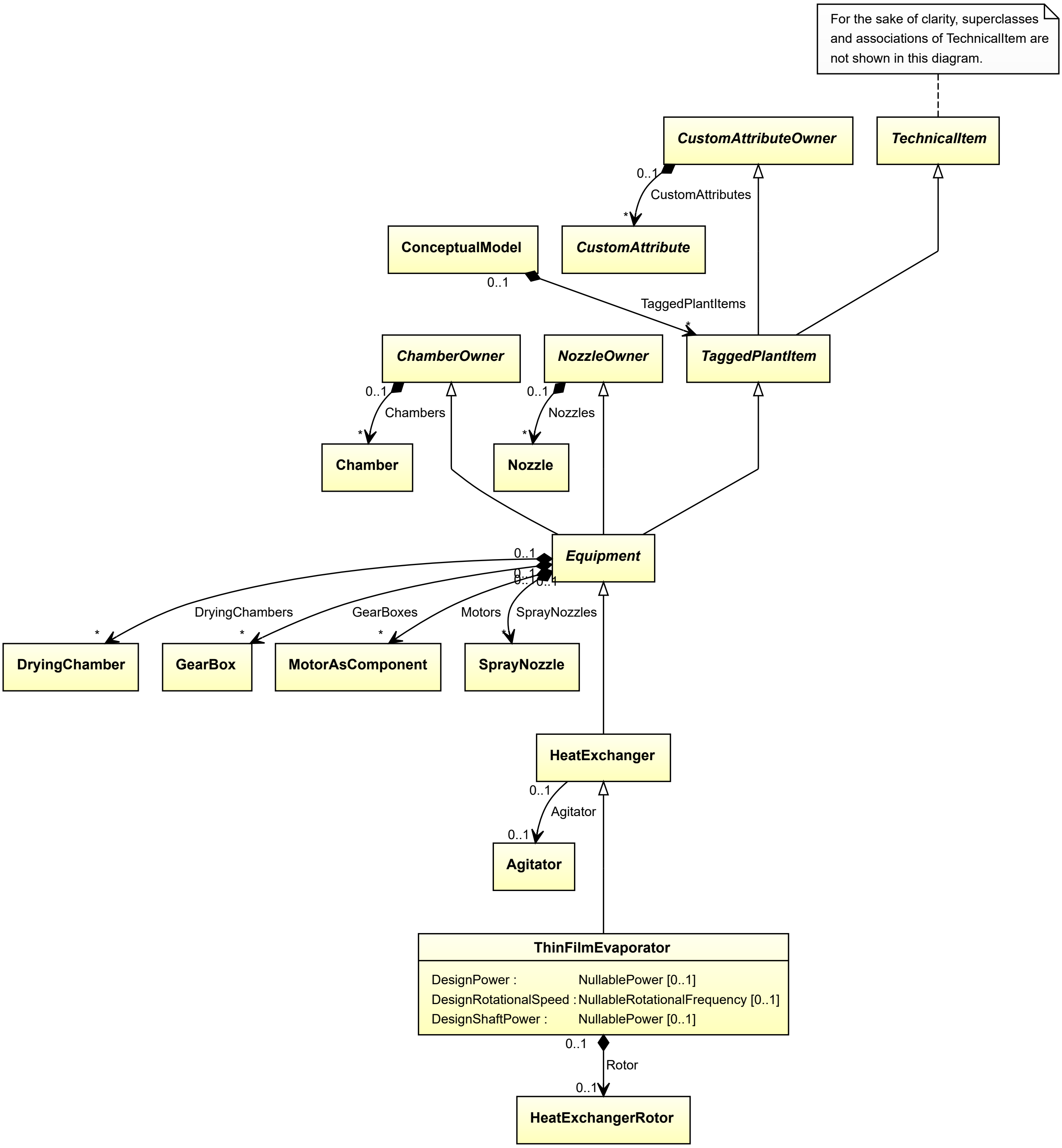
```
<Equipment
    ID="tank1"
    ComponentClass="Tank"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS445139" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="CylinderLength"
      AttributeURI="http://sandbox.dexpi.org/rdl/CylinderLength"
      Format="double"
      Value="2.0"
      Units="Metre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1332674" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.149. ThinFilmEvaporator

7.149.1 Overview

Class

A *HeatExchanger* and evaporator for the purification of temperature-sensitive products by evaporation, where a thin film of the liquid product on the inner side of a vertical evaporation pipe is generated by a rotating wiper system.



Supertypes

- *HeatExchanger*

Attributes (data)

Name	Multiplicity	Type
DesignPower	0..1	*NullablePower*
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*
DesignShaftPower	0..1	*NullablePower*

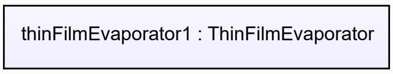
Attributes (composition)

Name	Multiplicity	Type
Rotor	0..1	*HeatExchangerRotor*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: THIN FILM EVAPORATOR
ComponentClass: ThinFilmEvaporator
ComponentClassURI: http://sandbox.dexpi.org/rdl/ThinFilmEvaporator

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="thinFilmEvaporator1"
    ComponentClass="ThinFilmEvaporator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ThinFilmEvaporator" ...>
  ...
</Equipment>
```

7.149.2 DesignPower

Attribute (data)

The power for which the *ThinFilmEvaporator* is designed.

Multiplicity: 0..1

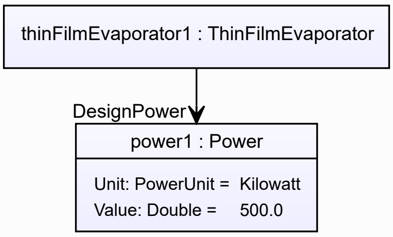
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN POWER
Name: DesignPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignPower

Example

The instance thinFilmEvaporator1 represents a *ThinFilmEvaporator* with a *DesignPower* of 500.0 kW.


thinFilmEvaporator1 : ThinFilmEvaporator
DesignPower
power1 : Power
Unit: PowerUnit = Kilowatt
Value: Double = 500.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="thinFilmEvaporator1"
    ComponentClass="ThinFilmEvaporator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ThinFilmEvaporator" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignPower"
      Format="double"
      Value="500.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.149.3 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *ThinFilmEvaporator* is designed.

Multiplicity: 0..1

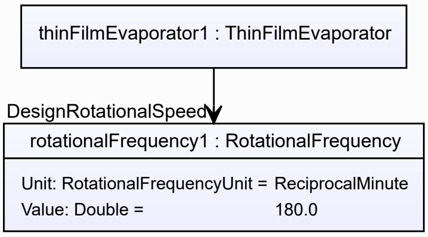
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

Example

The instance thinFilmEvaporator1 represents a *ThinFilmEvaporator* with a *DesignRotationalSpeed* of 180.0 min^{-1}.


thinFilmEvaporator1 : ThinFilmEvaporator
DesignRotationalSpeed
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 180.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="thinFilmEvaporator1"
    ComponentClass="ThinFilmEvaporator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ThinFilmEvaporator" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignRotationalSpeed"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignRotationalSpeed"
      Format="double"
      Value="180.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.149.4 DesignShaftPower

Attribute (data)

The shaft power for which the *ThinFilmEvaporator* is designed.

Multiplicity: 0..1

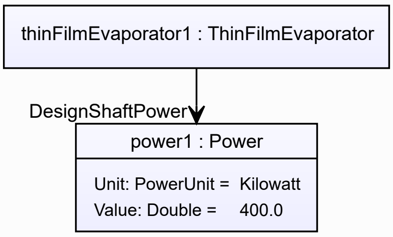
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SHAFT POWER
Name: DesignShaftPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignShaftPower

Example

The instance thinFilmEvaporator1 represents a *ThinFilmEvaporator* with a *DesignShaftPower* of 400.0 kW.


thinFilmEvaporator1 : ThinFilmEvaporator
DesignShaftPower
power1 : Power
Unit: PowerUnit = Kilowatt
Value: Double = 400.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="thinFilmEvaporator1"
    ComponentClass="ThinFilmEvaporator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ThinFilmEvaporator" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignShaftPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignShaftPower"
      Format="double"
      Value="400.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.149.5 Rotor

Attribute (composition)

The rotor of the *ThinFilmEvaporator*.

Multiplicity: 0..1

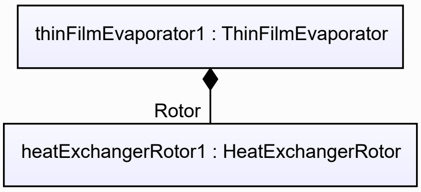
Type: *HeatExchangerRotor*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *HeatExchangerRotor*) is a child of the `<Equipment>` element for the attribute owner (a *ThinFilmEvaporator*).

Example


thinFilmEvaporator1 : ThinFilmEvaporator
Rotor
heatExchangerRotor1 : HeatExchangerRotor

Example: Implementation in Proteus Schema

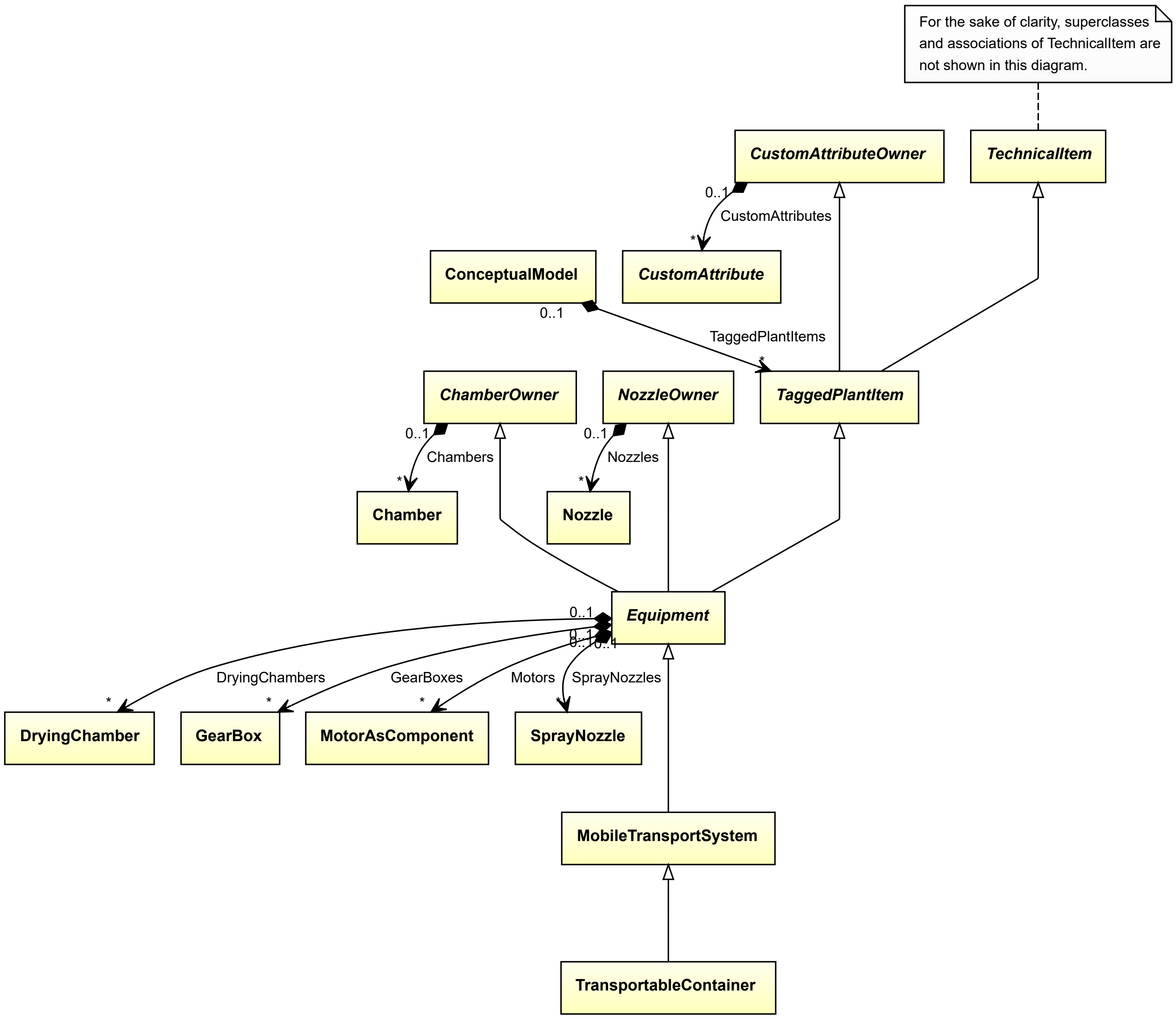
```
<Equipment
    ID="thinFilmEvaporator1"
    ComponentClass="ThinFilmEvaporator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ThinFilmEvaporator" ...>
  ...
  <Equipment
      ID="heatExchangerRotor1"
      ComponentClass="HeatExchangerRotor"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/HeatExchangerRotor" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.150. TransportableContainer

7.150.1 Overview

Class

A ‘container’ that is a transportable, with strength suitable to withstand shipment, storage, and handling (from http://data.posccaesar.org/rdl/RDS22164402859).



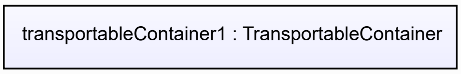
Supertypes

- *MobileTransportSystem*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: TRANSPORTABLE CONTAINER
ComponentClass: TransportableContainer
ComponentClassURI: http://data.posccaesar.org/rdl/RDS22164402859

Example



Example: Implementation in Proteus Schema

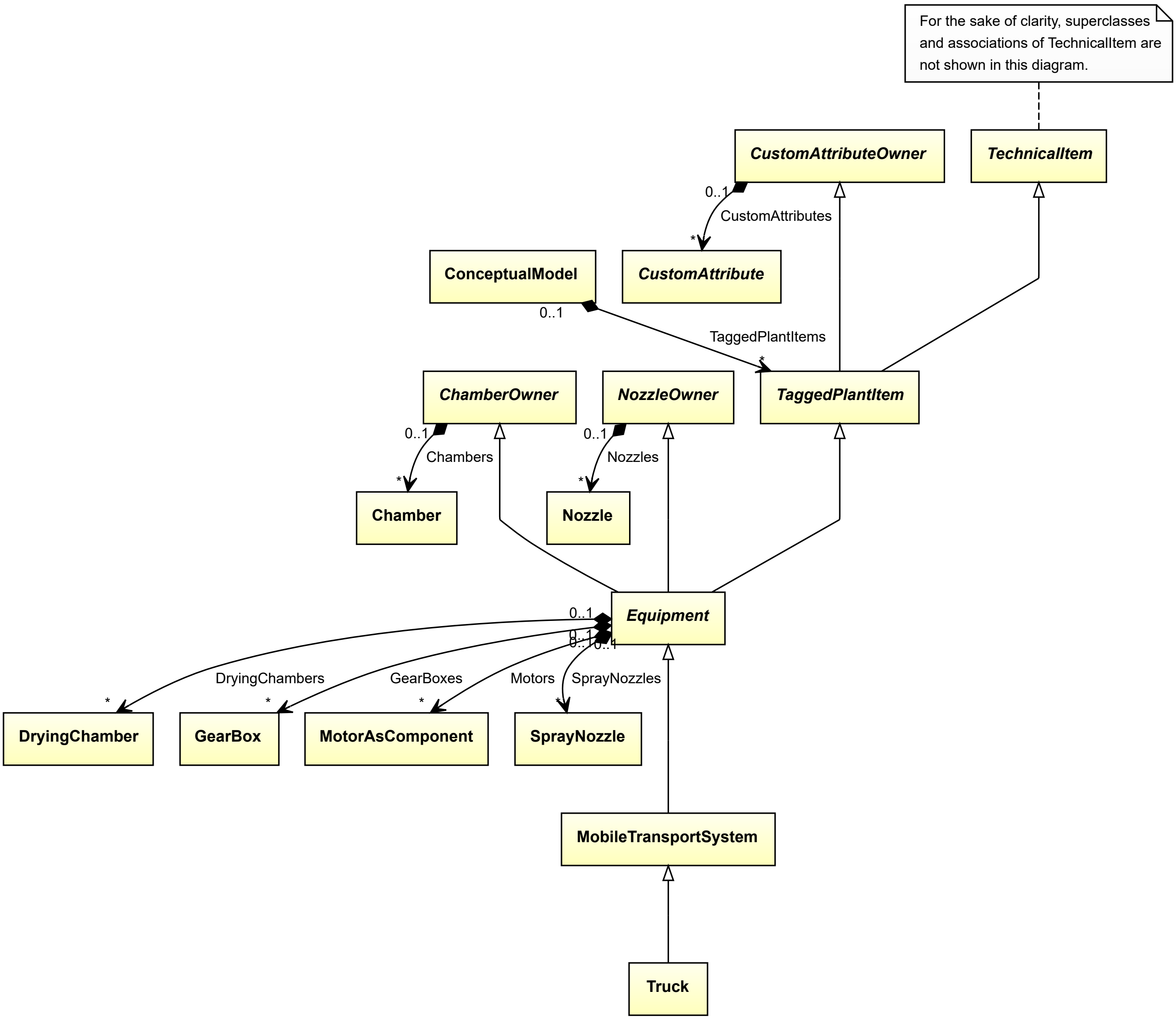
```
<Equipment
    ID="transportableContainer1"
    ComponentClass="TransportableContainer"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS22164402859" ...>
  ...
</Equipment>
```

7.151. Truck

7.151.1 Overview

Class

An automotive vehicle that is long, low and open intended for carrying goods by road (from http://data.posccaesar.org/rdl/RDS11524112).



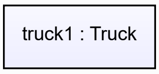
Supertypes

- *MobileTransportSystem*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: TRUCK
ComponentClass: Truck
ComponentClassURI: http://data.posccaesar.org/rdl/RDS11524112

Example



Example: Implementation in Proteus Schema

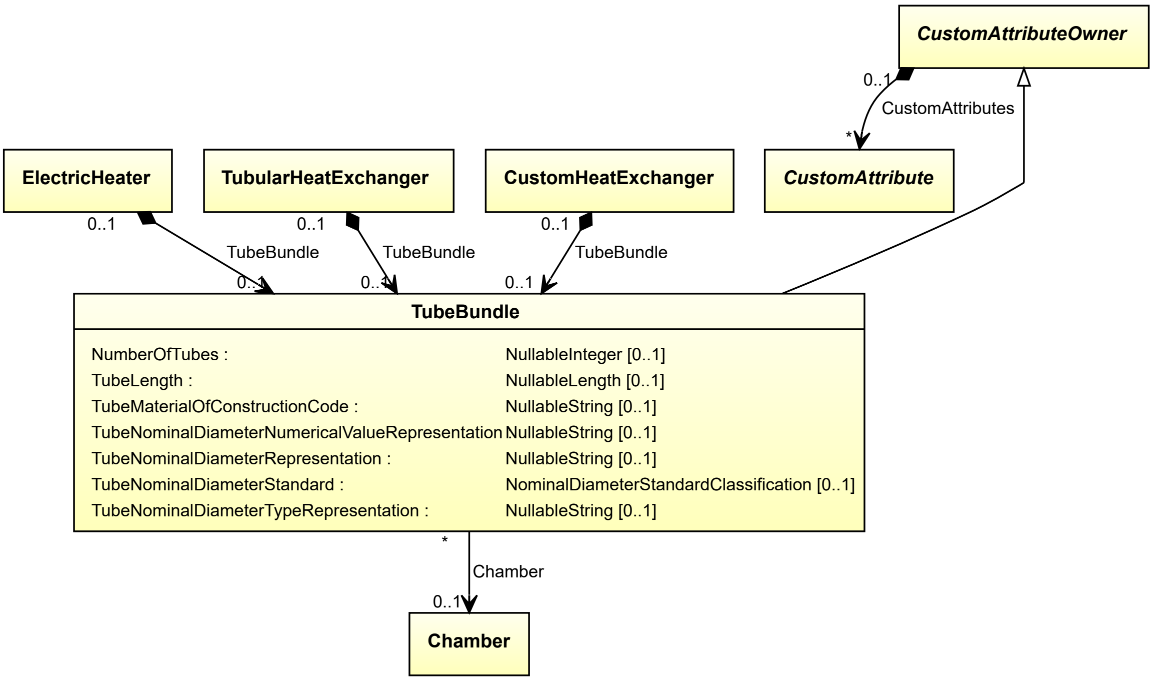
```
<Equipment
    ID="truck1"
    ComponentClass="Truck"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11524112" ...>
  ...
</Equipment>
```

7.152. TubeBundle

7.152.1 Overview

Class

A bundle that consists of several tubes assembled together allowing multiple flow paths from a single source (from http://data.posccaesar.org/rdl/RDS415259).


CustomAttributeOwner
0..1
CustomAttributes
*
ElectricHeater
TubularHeatExchanger
CustomHeatExchanger
CustomAttribute
0..1
TubeBundle
0..1
0..1
TubeBundle
0..1
0..1
TubeBundle
0..1
TubeBundle
NumberOfTubes : NullableInteger [0..1]
TubeLength : NullableLength [0..1]
TubeMaterialOfConstructionCode : NullableString [0..1]
TubeNominalDiameterNumericalValueRepresentation NullableString [0..1]
TubeNominalDiameterRepresentation : NullableString [0..1]
TubeNominalDiameterStandard : NominalDiameterStandardClassification [0..1]
TubeNominalDiameterTypeRepresentation : NullableString [0..1]
*
Chamber
0..1
Chamber

Supertypes

- *CustomAttributeOwner*

Attributes (data)

Name	Multiplicity	Type
NumberOfTubes	0..1	*NullableInteger*
TubeLength	0..1	*NullableLength*
TubeMaterialOfConstructionCode	0..1	*NullableString*
TubeNominalDiameterNumericalValueRepresentation	0..1	*NullableString*
TubeNominalDiameterRepresentation	0..1	*NullableString*
TubeNominalDiameterStandard	0..1	*NominalDiameterStandardClassification*
TubeNominalDiameterTypeRepresentation	0..1	*NullableString*

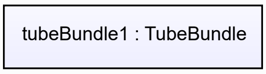
Attributes (reference)

Name	Multiplicity	Type
Chamber	0..1	*Chamber*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: TUBE BUNDLE
ComponentClass: TubeBundle
ComponentClassURI: http://data.posccaesar.org/rdl/RDS415259

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="tubeBundle1"
    ComponentClass="TubeBundle"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS415259" ...>
  ...
</Equipment>
```

7.152.2 Chamber

Attribute (reference)

The *Chamber* in which the *TubeBundle* is located, if applicable. The Chamber must be a component of the same object as the TubeBundle.

Multiplicity: 0..1

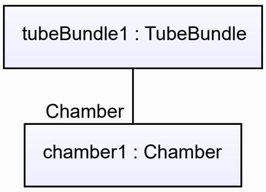
Type: *Chamber*

Opposite multiplicity: 0..*

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: `"is located in"`
Opposite association type: `"is the location of"`

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="tubeBundle1"
    ComponentClass="TubeBundle"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS415259" ...>
  ...
  <Association
    Type="is located in"
    ItemID="chamber1" />
  ...
<Equipment />
...
<Equipment
    ID="chamber1"
    ComponentClass="Chamber"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS903151421" ...>
  ...
  <Association
    Type="is the location of"
    ItemID="tubeBundle1" />
  ...
<Equipment />
```

7.152.3 NumberOfTubes

Attribute (data)

The number of tubes of the *TubeBundle*.

Multiplicity: 0..1

Type: *NullableInteger*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for integer values*.
RDL reference: NUMBER OF TUBES
Name: NumberOfTubes
AttributeURI: http://data.posccaesar.org/rdl/RDS363959

Example

36 (*Integer*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="tubeBundle1"
    ComponentClass="TubeBundle"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS415259" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NumberOfTubes"
      AttributeURI="http://data.posccaesar.org/rdl/RDS363959"
      Format="integer"
      Value="36" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.152.4 TubeLength

Attribute (data)

The length of the tubes of the *TubeBundle*.

Multiplicity: 0..1

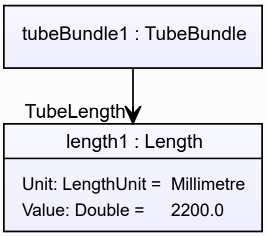
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: TUBE LENGTH
Name: TubeLength
AttributeURI: http://data.posccaesar.org/rdl/RDS363869

Example

The instance tubeBundle1 represents a *TubeBundle* with a *TubeLength* of 2200.0 mm.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="tubeBundle1"
    ComponentClass="TubeBundle"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS415259" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="TubeLength"
      AttributeURI="http://data.posccaesar.org/rdl/RDS363869"
      Format="double"
      Value="2200.0"
      Units="Millimetre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1357739" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.152.5 TubeMaterialOfConstructionCode

Attribute (data)

A code that gives the material of construction of the tubes of the *TubeBundle*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: TUBE MATERIAL OF CONSTRUCTION CODE ASSIGNMENT CLASS
Name: TubeMaterialOfConstructionCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/TubeMaterialOfConstructionCodeAssignmentClass

Example

“1.4306” (*String*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="tubeBundle1"
    ComponentClass="TubeBundle"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS415259" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="TubeMaterialOfConstructionCodeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/TubeMaterialOfConstructionCodeAssignmentClass"
      Format="string"
      Value="1.4306" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.152.6 TubeNominalDiameterNumericalValueRepresentation

Attribute (data)

A readable representation of the numerical value of the nominal diameter of the tubes.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: TUBE NOMINAL DIAMETER NUMERICAL VALUE REPRESENTATION ASSIGNMENT CLASS
Name: TubeNominalDiameterNumericalValueRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/TubeNominalDiameterNumericalValueRepresentationAssignmentClass

Example

“25” (*String*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="tubeBundle1"
    ComponentClass="TubeBundle"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS415259" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="TubeNominalDiameterNumericalValueRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/TubeNominalDiameterNumericalValueRepresentationAssignmentClass
↪"
      Format="string"
      Value="25" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.152.7 TubeNominalDiameterRepresentation

Attribute (data)

A readable representation of the nominal diameter of the tubes.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: TUBE NOMINAL DIAMETER REPRESENTATION ASSIGNMENT CLASS
Name: TubeNominalDiameterRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/TubeNominalDiameterRepresentationAssignmentClass

Example

"DN 25" (*String*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="tubeBundle1"
    ComponentClass="TubeBundle"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS415259" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="TubeNominalDiameterRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/TubeNominalDiameterRepresentationAssignmentClass"
      Format="string"
      Value="DN 25" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.152.8 TubeNominalDiameterStandard

Attribute (data)

The nominal diameter of the tubes, given as a reference to a nominal diameter standard and value.

Multiplicity: 0..1

Type: *NominalDiameterStandardClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: TUBE NOMINAL DIAMETER STANDARD SPECIALIZATION
Name: TubeNominalDiameterStandardSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/TubeNominalDiameterStandardSpecialization

Example

DN 25 (DIN 2448) (*NominalDiameterStandardClassification::Din2448ObjectDn25*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="tubeBundle1"
    ComponentClass="TubeBundle"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS415259" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="TubeNominalDiameterStandardSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/TubeNominalDiameterStandardSpecialization"
      Format="anyURI"
      Value="Din2448ObjectDn25"
      ValueURI="http://sandbox.dexpi.org/rdl/Din2448ObjectDn25" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.152.9 TubeNominalDiameterTypeRepresentation

Attribute (data)

A readable representation of the type of the nominal diameter of the tubes.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: TUBE NOMINAL DIAMETER TYPE REPRESENTATION ASSIGNMENT CLASS
Name: TubeNominalDiameterTypeRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/TubeNominalDiameterTypeRepresentationAssignmentClass

Example

“DN” (*String*)

Example: Implementation in Proteus Schema

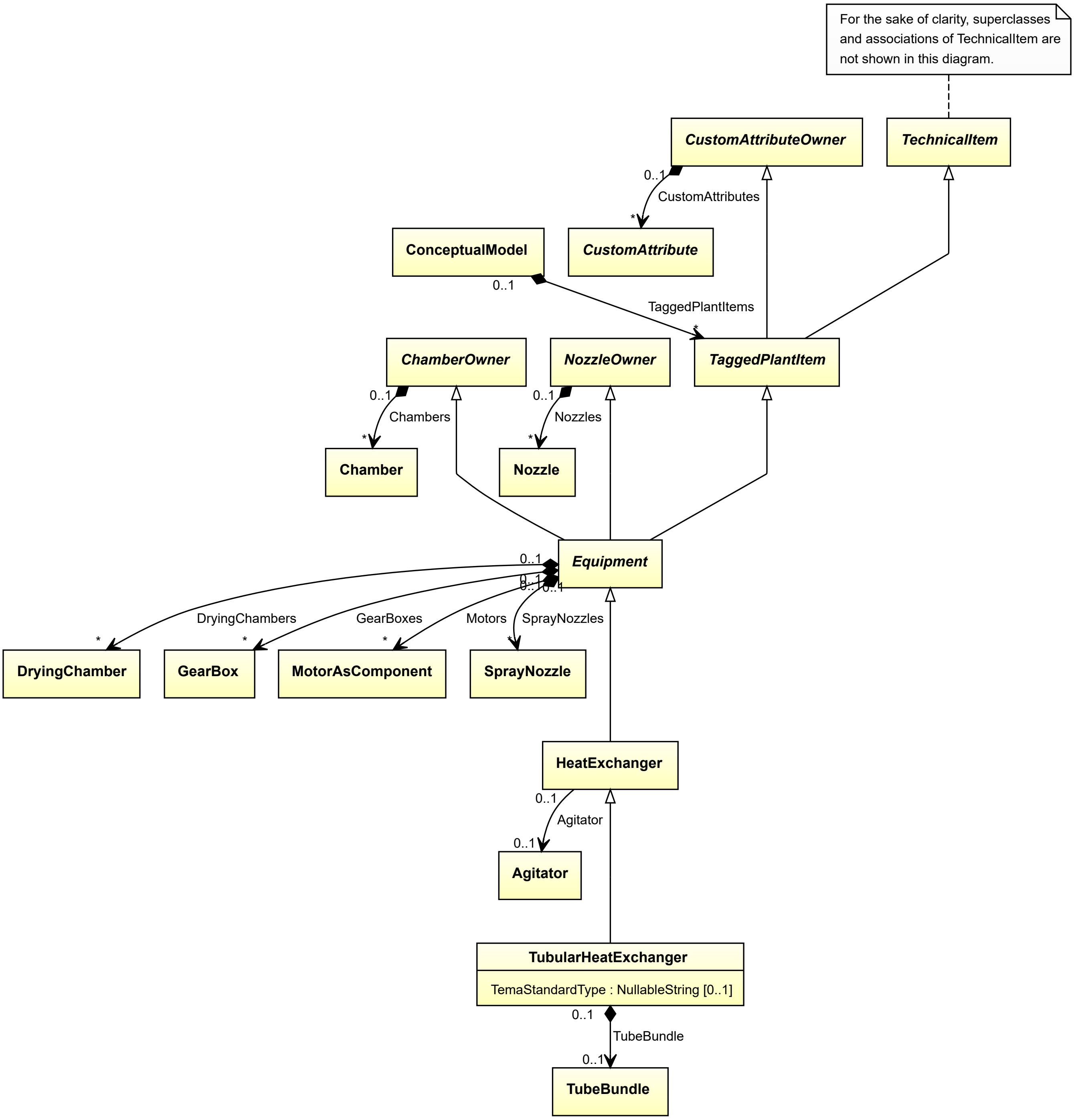
```
<Equipment
    ID="tubeBundle1"
    ComponentClass="TubeBundle"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS415259" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="TubeNominalDiameterTypeRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/TubeNominalDiameterTypeRepresentationAssignmentClass"
      Format="string"
      Value="DN" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.153. TubularHeatExchanger

7.153.1 Overview

Class

An indirect contact heat exchanger that separates the hot and cold fluids by tubes (from http://data.posccaesar.org/rdl/RDS13971182).



Supertypes

- *HeatExchanger*

Attributes (data)

Name	Multiplicity	Type
TemaStandardType	0..1	*NullableString*

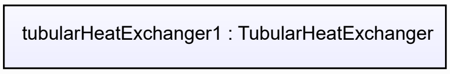
Attributes (composition)

Name	Multiplicity	Type
TubeBundle	0..1	*TubeBundle*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: TUBULAR HEAT EXCHANGER
ComponentClass: TubularHeatExchanger
ComponentClassURI: http://data.posccaesar.org/rdl/RDS13971182

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="tubularHeatExchanger1"
    ComponentClass="TubularHeatExchanger"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS13971182" ...>
  ...
</Equipment>
```

7.153.2 TemaStandardType

Attribute (data)

The type of the *TubularHeatExchanger* according to the Tubular Exchanger Manufacturers Association, Inc. (TEMA, http://www.tema.org). This is a three-letter code.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: TEMA STANDARD TYPE ASSIGNMENT CLASS
Name: TemaStandardTypeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/TemaStandardTypeAssignmentClass

Example

“AEL” (*String*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="tubularHeatExchanger1"
    ComponentClass="TubularHeatExchanger"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS13971182" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="TemaStandardTypeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/TemaStandardTypeAssignmentClass"
      Format="string"
      Value="AEL" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.153.3 TubeBundle

Attribute (composition)

The tube bundle of the *TubularHeatExchanger*.

Multiplicity: 0..1

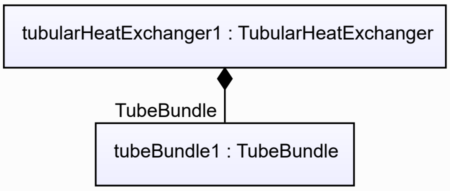
Type: *TubeBundle*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *TubeBundle*) is a child of the `<Equipment>` element for the attribute owner (a *TubularHeatExchanger*).

Example



Example: Implementation in Proteus Schema

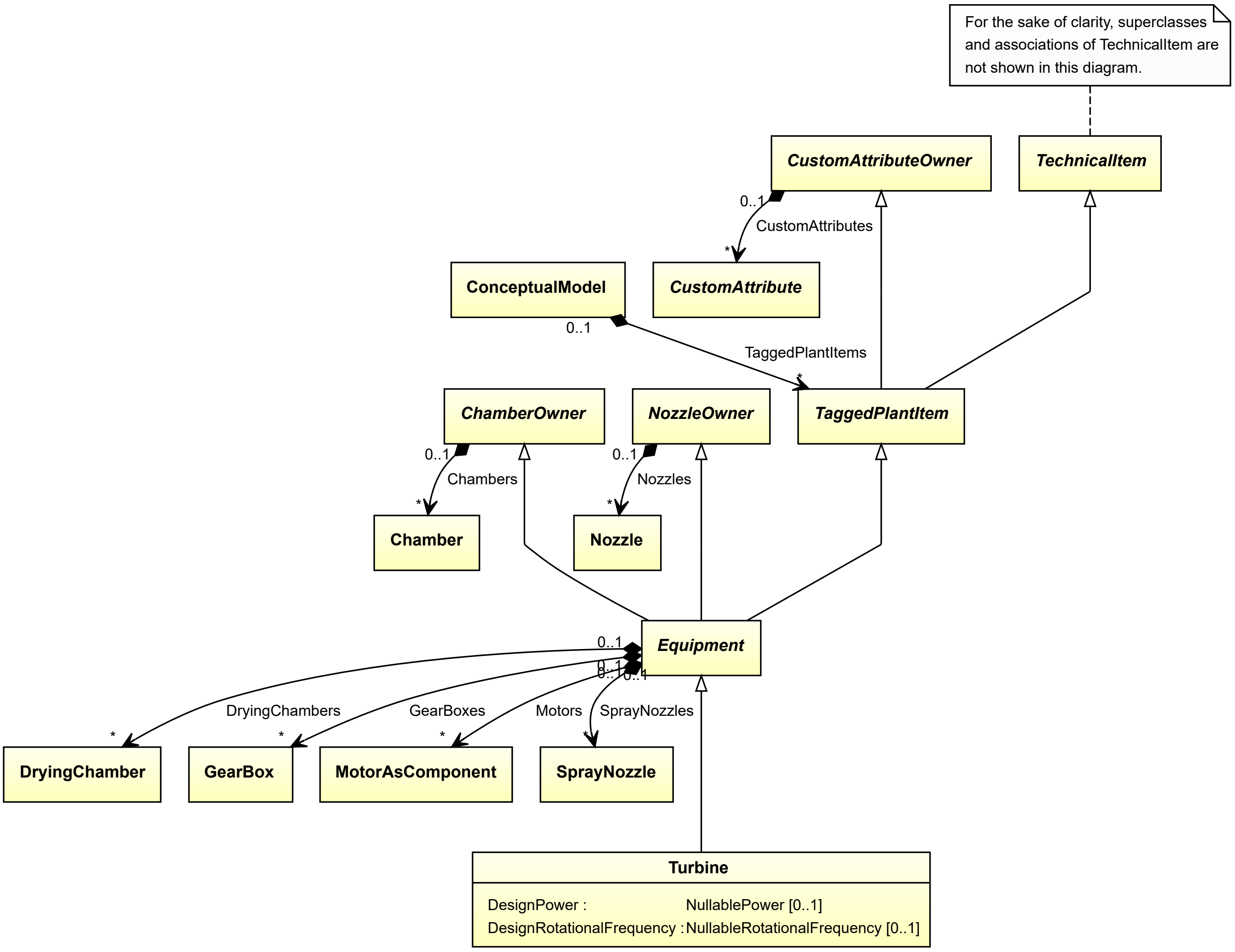
```
<Equipment
    ID="tubularHeatExchanger1"
    ComponentClass="TubularHeatExchanger"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS13971182" ...>
  ...
  <Equipment
      ID="tubeBundle1"
      ComponentClass="TubeBundle"
      ComponentClassURI="http://data.posccaesar.org/rdl/RDS415259" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.154. Turbine

7.154.1 Overview

Class

An object that is a rotary mechanical device that extracts energy from a fluid flow and converts it into useful work (from http://data.15926.org/rdl/RDS313289).



Supertypes

- *Equipment*

Subtypes

- *CustomTurbine*
- *GasTurbine*
- *SteamTurbine*

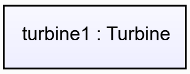
Attributes (data)

Name	Multiplicity	Type
DesignPower	0..1	*NullablePower*
DesignRotationalFrequency	0..1	*NullableRotationalFrequency*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: TURBINE
ComponentClass: Turbine
ComponentClassURI: http://data.posccaesar.org/rdl/RDS313289

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="turbine1"
    ComponentClass="Turbine"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS313289" ...>
  ...
</Equipment>
```

7.154.2 DesignPower

Attribute (data)

The power for which the *Turbine* is designed.

Multiplicity: 0..1

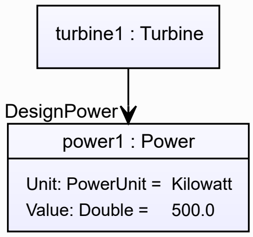
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN POWER
Name: DesignPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignPower

Example

The instance turbine1 represents a *Turbine* with a *DesignPower* of 500.0 kW.



Example: Implementation in Proteus Schema

```
<Equipment
    ID="turbine1"
    ComponentClass="Turbine"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS313289" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignPower"
      Format="double"
      Value="500.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.154.3 DesignRotationalFrequency

Attribute (data)

The rotational frequency for which the *Turbine* is designed.

Multiplicity: 0..1

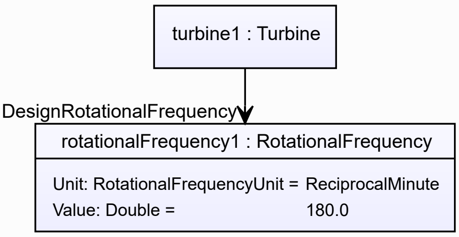
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL FREQUENCY
Name: DesignRotationalFrequency
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalFrequency

Example

The instance turbine1 represents a *Turbine* with a *DesignRotationalFrequency* of 180.0 min^{-1}.


turbine1 : Turbine
DesignRotationalFrequency
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 180.0

Example: Implementation in Proteus Schema

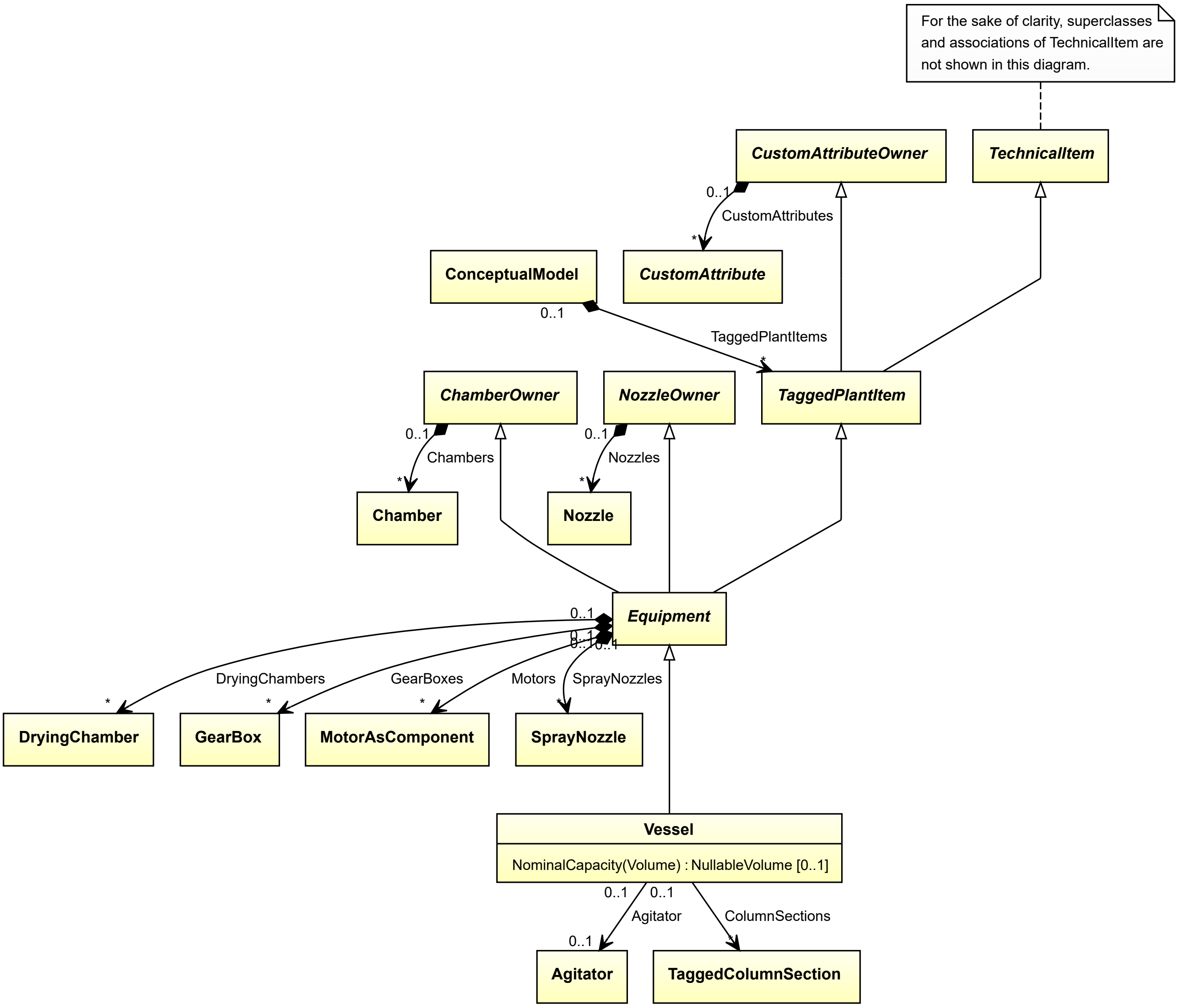
```
<Equipment
    ID="turbine1"
    ComponentClass="Turbine"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS313289" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignRotationalFrequency"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignRotationalFrequency"
      Format="double"
      Value="180.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.155. Vessel

7.155.1 Overview

Class

A container intended for storage and/or processing of fluids or solids.



Supertypes

- *Equipment*

Subtypes

- *CustomVessel*
- *PressureVessel*
- *Silo*
- *Tank*

Attributes (data)

Name	Multiplicity	Type
NominalCapacity(Volume)	0..1	*NullableVolume*

Attributes (reference)

Name	Multiplicity	Type
Agitator	0..1	*Agitator*
ColumnSections	*	*TaggedColumnSection*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: VESSEL
ComponentClass: Vessel
ComponentClassURI: http://data.posccaesar.org/rdl/RDS414674

Example

Example: Implementation in Proteus Schema

```
<Equipment
    ID="vessel1"
    ComponentClass="Vessel"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS414674" ...>
  ...
</Equipment>
```

7.155.2 Agitator

Attribute (reference)

The *Agitator* of the *Vessel*, if applicable.

Multiplicity: 0..1

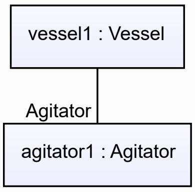
Type: *Agitator*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: `"is the location of"`
Opposite association type: `"is located in"`

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="vessel1"
    ComponentClass="Vessel"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS414674" ...>
  ...
  <Association
    Type="is the location of"
    ItemID="agitator1" />
  ...
<Equipment />
...
<Equipment
    ID="agitator1"
    ComponentClass="Agitator"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS16045622" ...>
  ...
  <Association
    Type="is located in"
    ItemID="vessel1" />
  ...
<Equipment />
```

7.155.3 ColumnSections

Attribute (reference)

The column sections of the *Vessel*, if applicable.

Multiplicity: *

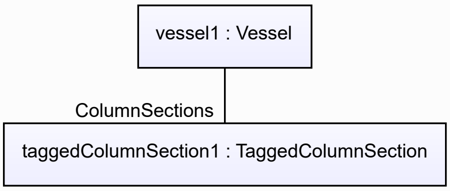
Type: *TaggedColumnSection*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: "is the location of"
Opposite association type: "is located in"

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="vessel1"
    ComponentClass="Vessel"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS414674" ...>
  ...
  <Association
    Type="is the location of"
    ItemID="taggedColumnSection1" />
  ...
<Equipment />
...
<Equipment
    ID="taggedColumnSection1"
    ComponentClass="ColumnSection"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ColumnSection" ...>
  ...
  <Association
    Type="is located in"
    ItemID="vessel1" />
  ...
<Equipment />
```

7.155.4 NominalCapacity(Volume)

Attribute (data)

The nominal volumetric capacity of the *Vessel*.

Multiplicity: 0..1

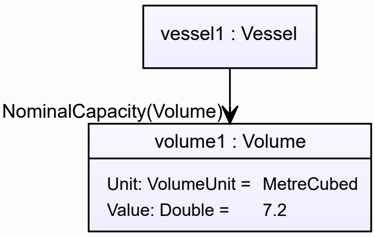
Type: *NullableVolume*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: NOMINAL CAPACITY VOLUME
Name: NominalCapacityVolume
AttributeURI: http://sandbox.dexpi.org/rdl/NominalCapacityVolume

Example

The instance vessel1 represents a *Vessel* with a *NominalCapacity(Volume)* of 7.2 m^3.



Example: Implementation in Proteus Schema

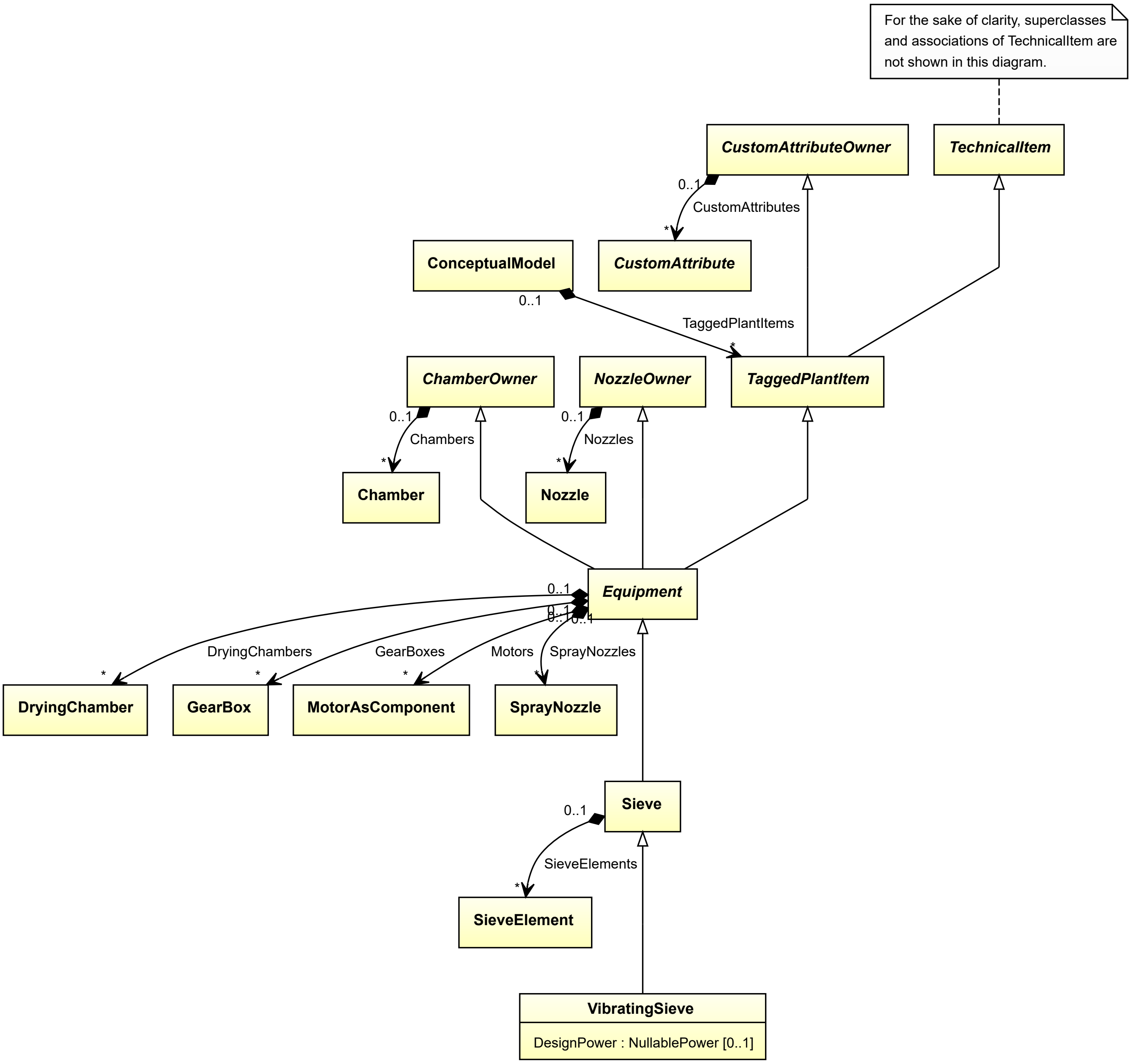
```
<Equipment
    ID="vessel1"
    ComponentClass="Vessel"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS414674" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NominalCapacityVolume"
      AttributeURI="http://sandbox.dexpi.org/rdl/NominalCapacityVolume"
      Format="double"
      Value="7.2"
      Units="MetreCubed"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1349099" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.156. VibratingSieve

7.156.1 Overview

Class

A *Sieve* where the product to be sieved is transported over the mesh by vibration of the latter (from http://data.15926.org/rdl/RDS2226670).



Supertypes

- *Sieve*

Attributes (data)

Name	Multiplicity	Type
DesignPower	0..1	*NullablePower*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <Equipment>
RDL reference: VIBRATING SCREEN

ComponentClass: VibratingScreen
ComponentClassURI: http://sandbox.dexpi.org/rdl/VibratingScreen

Example

vibratingSieve1 : VibratingSieve

Example: Implementation in Proteus Schema

```
<Equipment
    ID="vibratingSieve1"
    ComponentClass="VibratingScreen"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/VibratingScreen" ...>
  ...
</Equipment>
```

7.156.2 DesignPower

Attribute (data)

The power for which the *VibratingSieve* is designed.

Multiplicity: 0..1

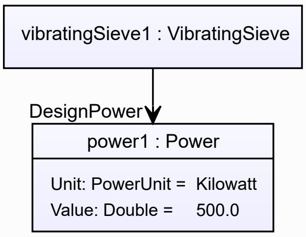
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN POWER
Name: DesignPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignPower

Example

The instance vibratingSieve1 represents a *VibratingSieve* with a *DesignPower* of 500.0 kW.



Example: Implementation in Proteus Schema

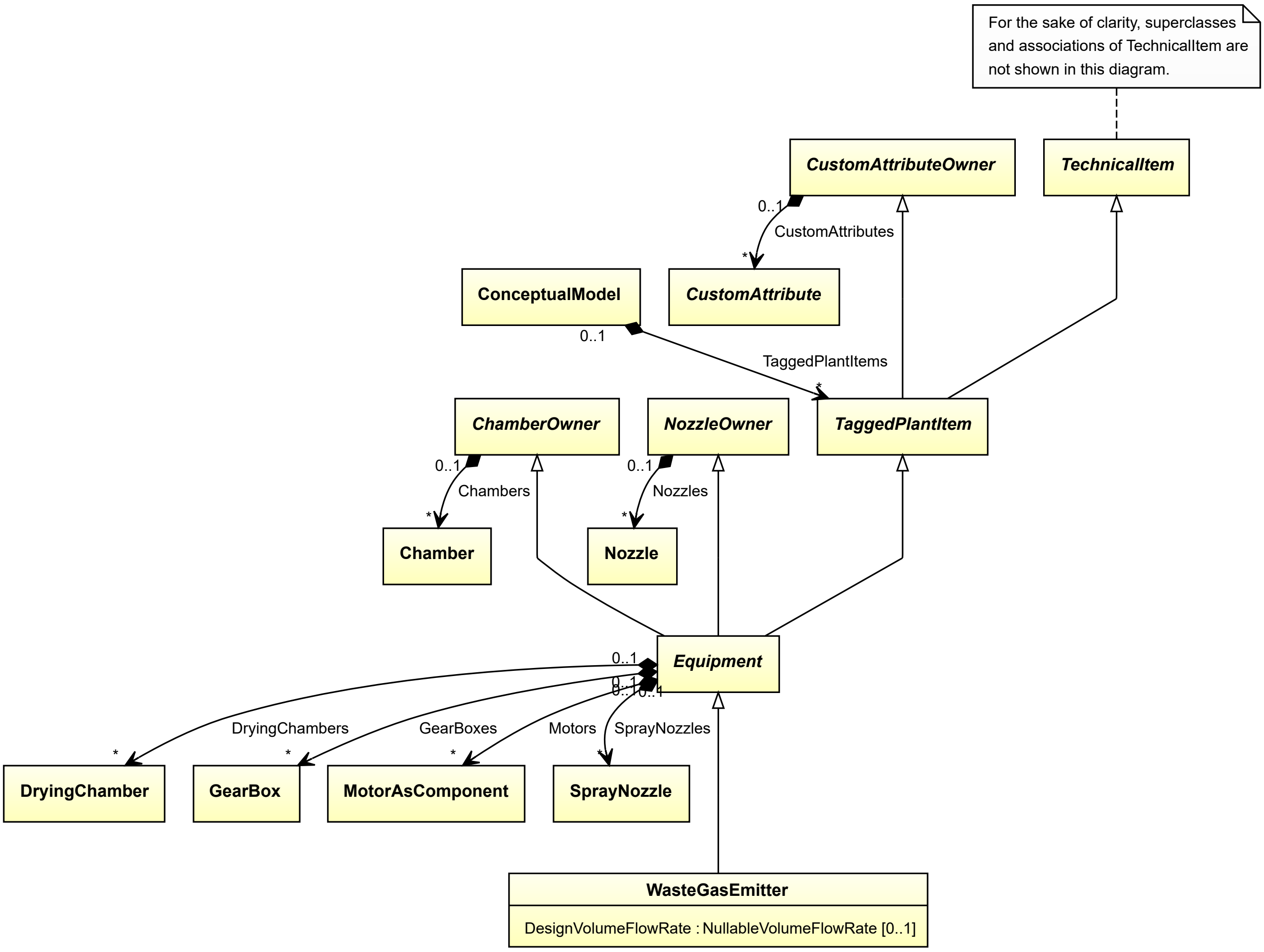
```
<Equipment
    ID="vibratingSieve1"
    ComponentClass="VibratingScreen"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/VibratingScreen" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignPower"
      Format="double"
      Value="500.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.157. WasteGasEmitter

7.157.1 Overview

Class

A physical object that is intended to release/emit waste gas from the process.



Supertypes

- *Equipment*

Subtypes

- *Chimney*
- *CustomWasteGasEmitter*
- *Flare*

Attributes (data)

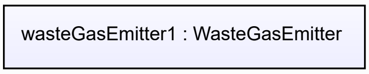
Name	Multiplicity	Type
DesignVolumeFlowRate	0..1	*NullableVolumeFlowRate*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: WASTE GAS EMITTER

ComponentClass: WasteGasEmitter
ComponentClassURI: http://sandbox.dexpi.org/rdl/WasteGasEmitter

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="wasteGasEmitter1"
    ComponentClass="WasteGasEmitter"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/WasteGasEmitter" ...>
  ...
</Equipment>
```

7.157.2 DesignVolumeFlowRate

Attribute (data)

The volume flow rate for which the *WasteGasEmitter* is designed.

Multiplicity: 0..1

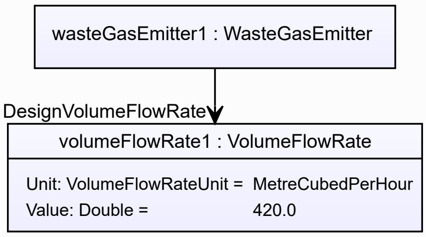
Type: *NullableVolumeFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN VOLUME FLOW RATE
Name: DesignVolumeFlowRate
AttributeURI: http://data.posccaesar.org/rdl/RDS14286227

Example

The instance wasteGasEmitter1 represents a *WasteGasEmitter* with a *DesignVolumeFlowRate* of 420.0 m^3/h.


wasteGasEmitter1 : WasteGasEmitter
DesignVolumeFlowRate
volumeFlowRate1 : VolumeFlowRate
Unit: VolumeFlowRateUnit = MetreCubedPerHour
Value: Double = 420.0

Example: Implementation in Proteus Schema

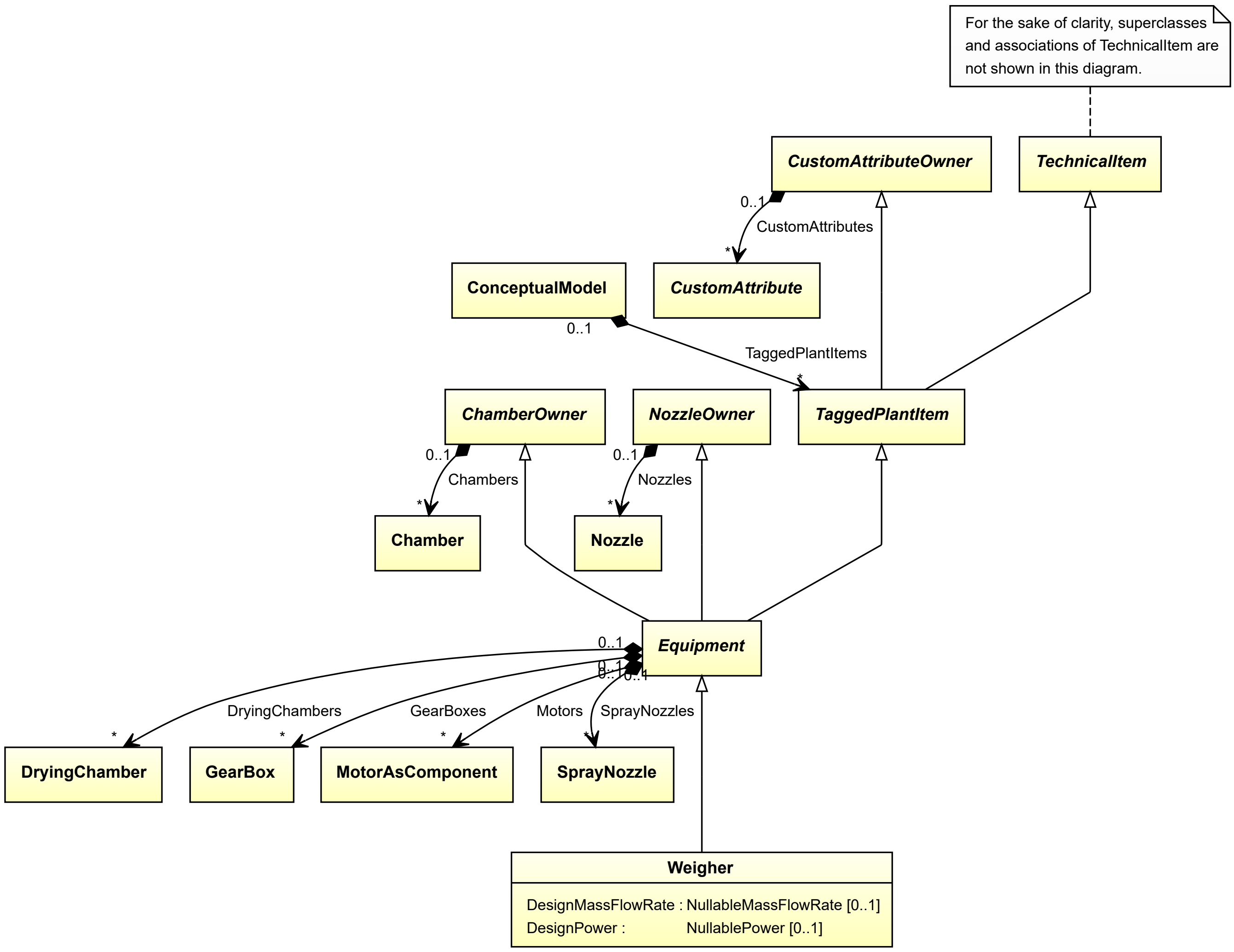
```
<Equipment
    ID="wasteGasEmitter1"
    ComponentClass="WasteGasEmitter"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/WasteGasEmitter" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignVolumeFlowRate"
      AttributeURI="http://data.posccaesar.org/rdl/RDS14286227"
      Format="double"
      Value="420.0"
      Units="MetreCubedPerHour"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1321064" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.158. Weigher

7.158.1 Overview

Class

A functional object that is capable of weighing.



Supertypes

- *Equipment*

Subtypes

- *BatchWeigher*
- *ContinuousWeigher*
- *CustomWeigher*

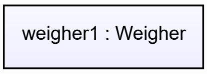
Attributes (data)

Name	Multiplicity	Type
DesignMassFlowRate	0..1	*NullableMassFlowRate*
DesignPower	0..1	*NullablePower*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: WEIGHER
ComponentClass: Weigher
ComponentClassURI: http://sandbox.dexpi.org/rdl/Weigher

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="weigher1"
    ComponentClass="Weigher"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Weigher" ...>
  ...
</Equipment>
```

7.158.2 DesignMassFlowRate

Attribute (data)

The mass flow rate for which the *Weigher* is designed.

Multiplicity: 0..1

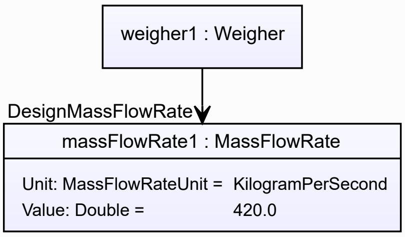
Type: *NullableMassFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN MASS FLOW RATE
Name: DesignMassFlowRate
AttributeURI: http://data.posccaesar.org/rdl/RDS14286182

Example

The instance weigher1 represents a *Weigher* with a *DesignMassFlowRate* of 420.0 kg/s.


weigher1 : Weigher
DesignMassFlowRate
massFlowRate1 : MassFlowRate
Unit: MassFlowRateUnit = KilogramPerSecond
Value: Double = 420.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="weigher1"
    ComponentClass="Weigher"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Weigher" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignMassFlowRate"
      AttributeURI="http://data.posccaesar.org/rdl/RDS14286182"
      Format="double"
      Value="420.0"
      Units="KilogramPerSecond"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1329659" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.158.3 DesignPower

Attribute (data)

The power for which the *Weigher* is designed.

Multiplicity: 0..1

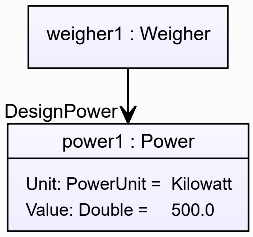
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN POWER
Name: DesignPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignPower

Example

The instance weigher1 represents a *Weigher* with a *DesignPower* of 500.0 kW.



Example: Implementation in Proteus Schema

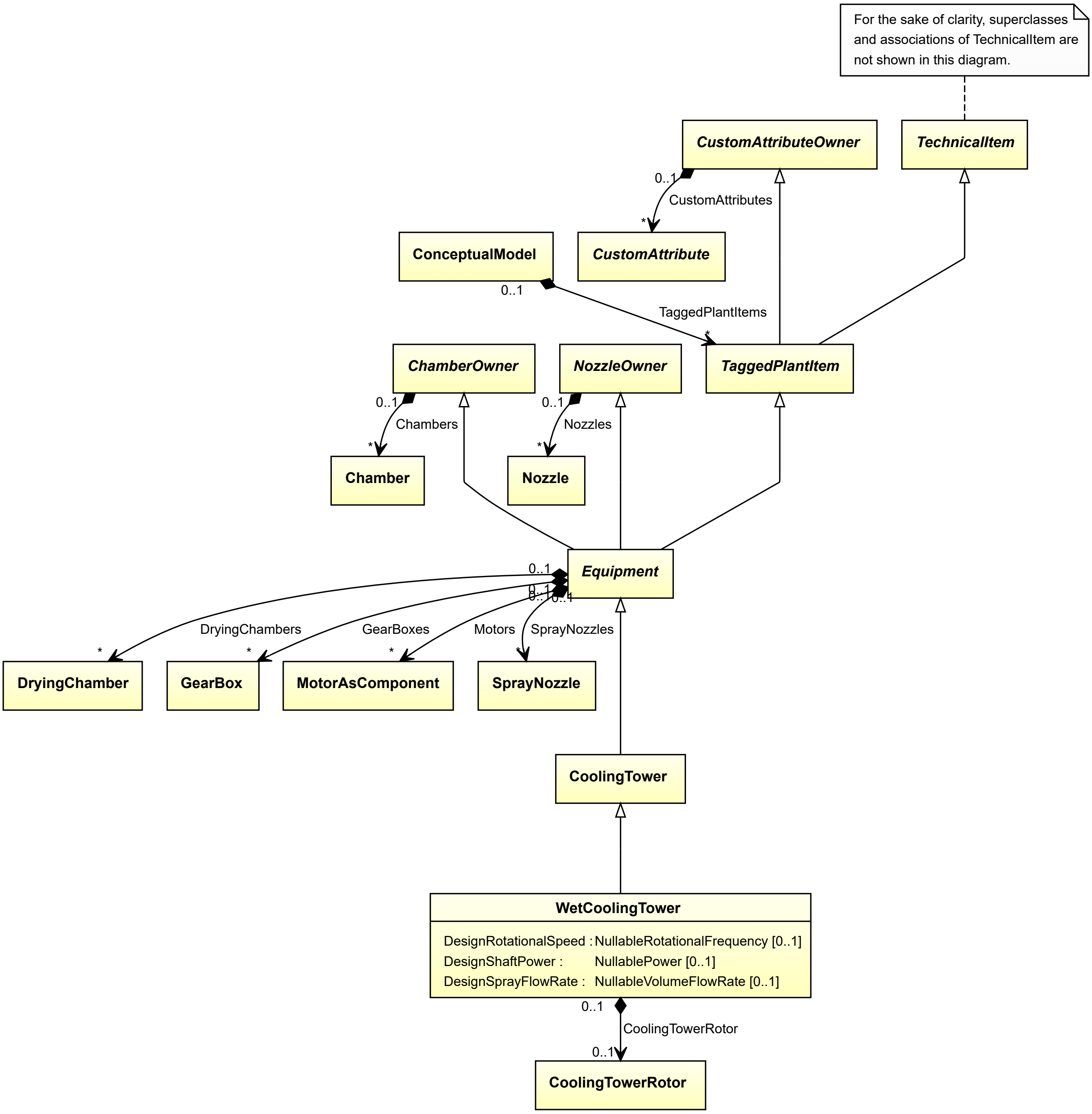
```
<Equipment
    ID="weigher1"
    ComponentClass="Weigher"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Weigher" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignPower"
      Format="double"
      Value="500.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.159. WetCoolingTower

7.159.1 Overview

Class

A *CoolingTower* that derives its primary cooling effect from the evaporation that takes place when air and water are brought into direct contact.



Supertypes

- *CoolingTower*

Attributes (data)

Name	Multiplicity	Type
DesignRotationalSpeed	0..1	*NullableRotationalFrequency*
DesignShaftPower	0..1	*NullablePower*
DesignSprayFlowRate	0..1	*NullableVolumeFlowRate*

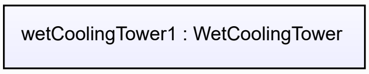
Attributes (composition)

Name	Multiplicity	Type
CoolingTowerRotor	0..1	*CoolingTowerRotor*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<Equipment>`
RDL reference: WET COOLING TOWER
ComponentClass: WetCoolingTower
ComponentClassURI: http://data.posccaesar.org/rdl/RDS14071846

Example



Example: Implementation in Proteus Schema

```
<Equipment
    ID="wetCoolingTower1"
    ComponentClass="WetCoolingTower"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS14071846" ...>
  ...
</Equipment>
```

7.159.2 CoolingTowerRotor

Attribute (composition)

The cooling tower rotor of the *WetCoolingTower*.

Multiplicity: 0..1

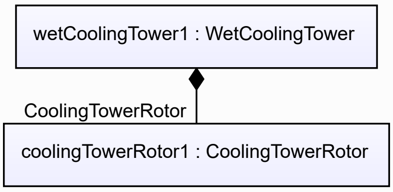
Type: *CoolingTowerRotor*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *CoolingTowerRotor*) is a child of the `<Equipment>` element for the attribute owner (a *WetCoolingTower*).

Example


wetCoolingTower1 : WetCoolingTower
CoolingTowerRotor
coolingTowerRotor1 : CoolingTowerRotor

Example: Implementation in Proteus Schema

```
<Equipment
    ID="wetCoolingTower1"
    ComponentClass="WetCoolingTower"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS14071846" ...>
  ...
  <Equipment
      ID="coolingTowerRotor1"
      ComponentClass="CoolingTowerRotor"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/CoolingTowerRotor" ...>
    ...
  <Equipment />
  ...
<Equipment />
```

7.159.3 DesignRotationalSpeed

Attribute (data)

The rotational speed for which the *WetCoolingTower* is designed.

Multiplicity: 0..1

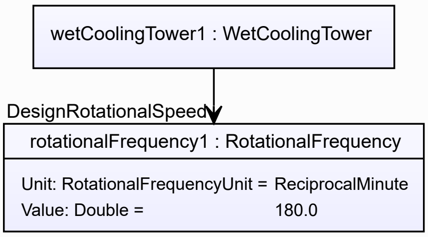
Type: *NullableRotationalFrequency*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN ROTATIONAL SPEED
Name: DesignRotationalSpeed
AttributeURI: http://sandbox.dexpi.org/rdl/DesignRotationalSpeed

Example

The instance wetCoolingTower1 represents a *WetCoolingTower* with a *DesignRotationalSpeed* of 180.0 min^{-1}.


wetCoolingTower1 : WetCoolingTower
DesignRotationalSpeed
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 180.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="wetCoolingTower1"
    ComponentClass="WetCoolingTower"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS14071846" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignRotationalSpeed"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignRotationalSpeed"
      Format="double"
      Value="180.0"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.159.4 DesignShaftPower

Attribute (data)

The shaft power for which the *WetCoolingTower* is designed.

Multiplicity: 0..1

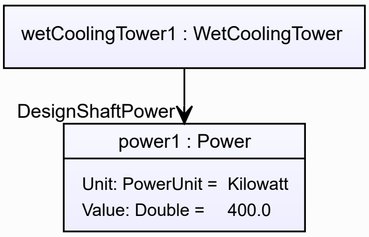
Type: *NullablePower*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SHAFT POWER
Name: DesignShaftPower
AttributeURI: http://sandbox.dexpi.org/rdl/DesignShaftPower

Example

The instance wetCoolingTower1 represents a *WetCoolingTower* with a *DesignShaftPower* of 400.0 kW.


wetCoolingTower1 : WetCoolingTower
DesignShaftPower
power1 : Power
Unit: PowerUnit = Kilowatt
Value: Double = 400.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="wetCoolingTower1"
    ComponentClass="WetCoolingTower"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS14071846" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignShaftPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignShaftPower"
      Format="double"
      Value="400.0"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

7.159.5 DesignSprayFlowRate

Attribute (data)

The spray volume flow rate for the motive fluid for which the *WetCoolingTower* is designed.

Multiplicity: 0..1

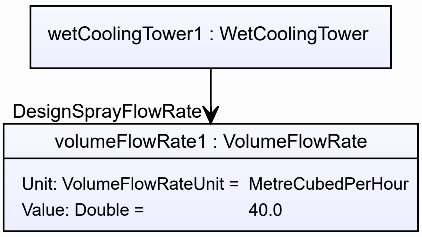
Type: *NullableVolumeFlowRate*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: DESIGN SPRAY FLOW RATE
Name: DesignSprayFlowRate
AttributeURI: http://sandbox.dexpi.org/rdl/DesignSprayFlowRate

Example

The instance wetCoolingTower1 represents a *WetCoolingTower* with a *DesignSprayFlowRate* of 40.0 m^3/h.


wetCoolingTower1 : WetCoolingTower
DesignSprayFlowRate
volumeFlowRate1 : VolumeFlowRate
Unit: VolumeFlowRateUnit = MetreCubedPerHour
Value: Double = 40.0

Example: Implementation in Proteus Schema

```
<Equipment
    ID="wetCoolingTower1"
    ComponentClass="WetCoolingTower"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS14071846" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DesignSprayFlowRate"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignSprayFlowRate"
      Format="double"
      Value="40.0"
      Units="MetreCubedPerHour"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1321064" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

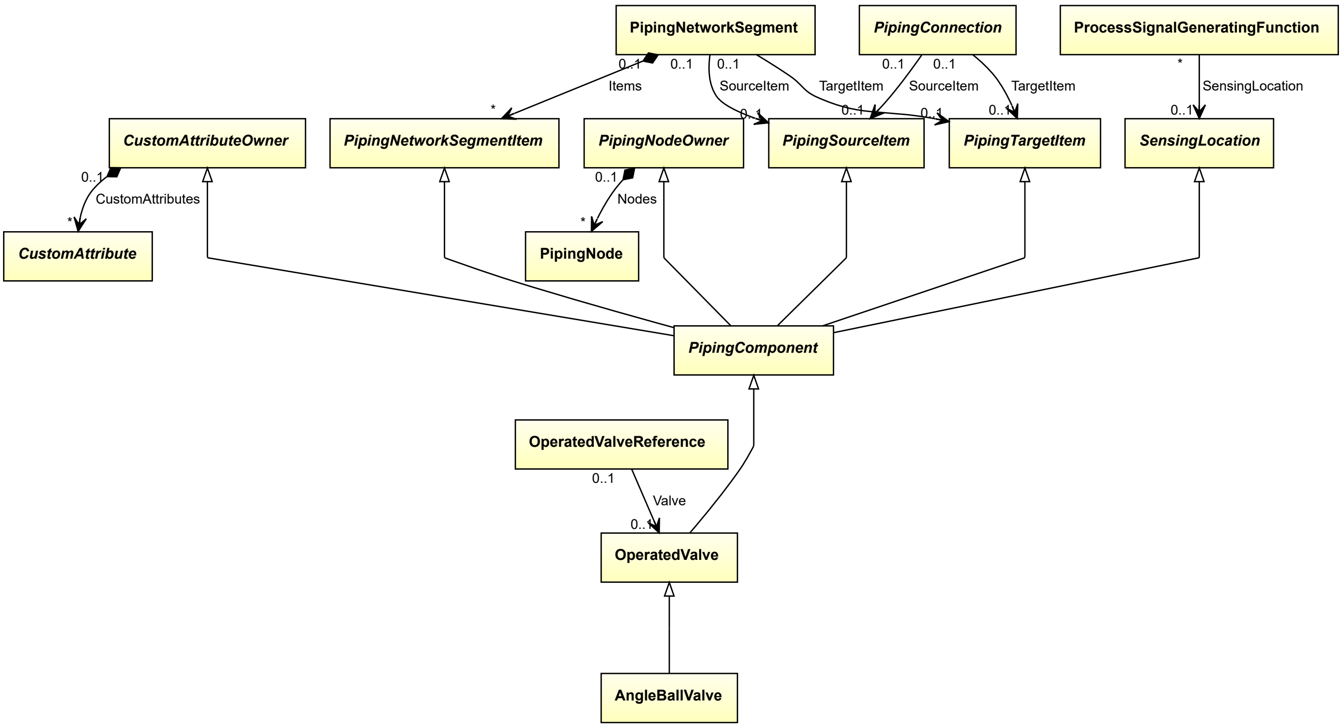
Package Piping 8

8.1. AngleBallValve

8.1.1 Overview

Class

A valve that has valve ports which are not in-line and that has a ball closure member.



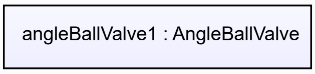
Supertypes

- *OperatedValve*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingComponent>`
RDL reference: ANGLE BALL VALVE
ComponentClass: AngleBallValve
ComponentClassURI: http://sandbox.dexpi.org/rdl/AngleBallValve

Example



Example: Implementation in Proteus Schema

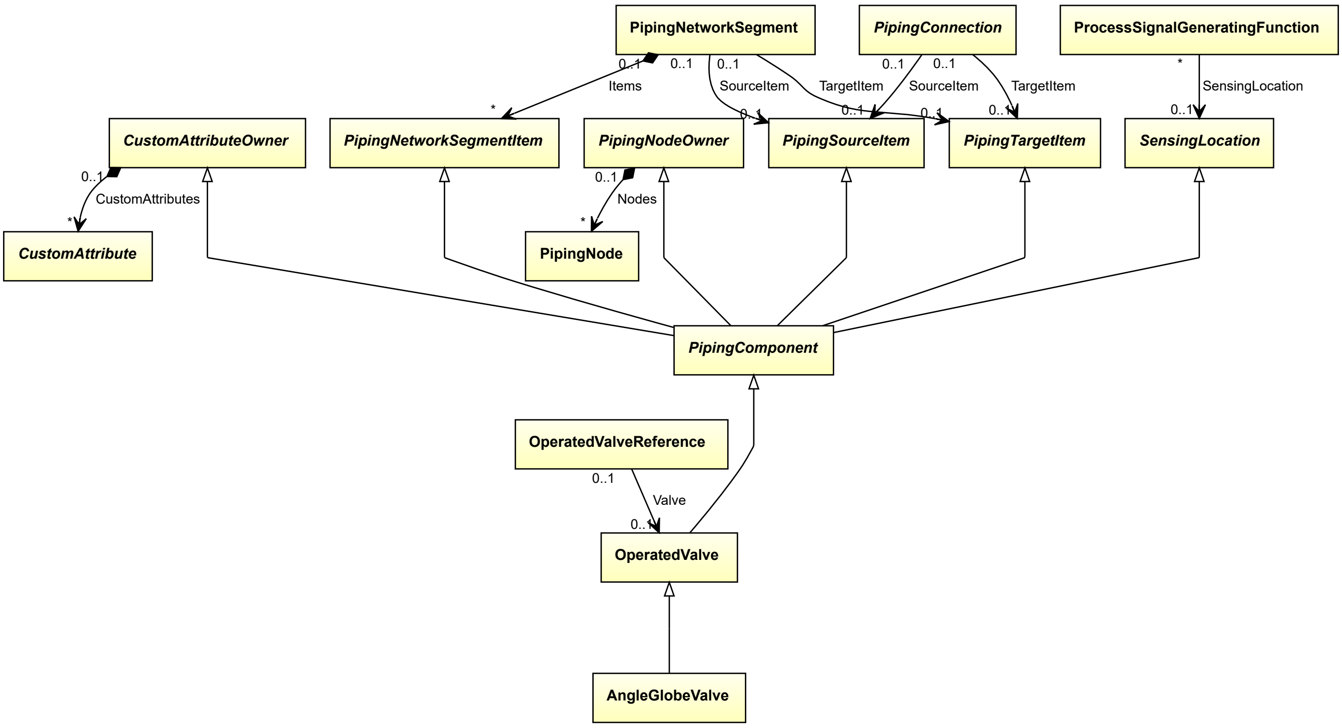
```
<PipingComponent
    ID="angleBallValve1"
    ComponentClass="AngleBallValve"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/AngleBallValve" ...>
  ...
</PipingComponent>
```

8.2. AngleGlobeValve

8.2.1 Overview

Class

A globe valve that deviates from the in-line design, i.e. with a body shape designed to adjust the flow direction with a specified angle relative to the straight through-flow an in-line valve would have provided for (from http://data.posccaesar.org/rdl/RDS882944).



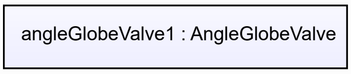
Supertypes

- *OperatedValve*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingComponent>`
RDL reference: ANGLE GLOBE VALVE
ComponentClass: AngleGlobeValve
ComponentClassURI: http://data.posccaesar.org/rdl/RDS882944

Example



Example: Implementation in Proteus Schema

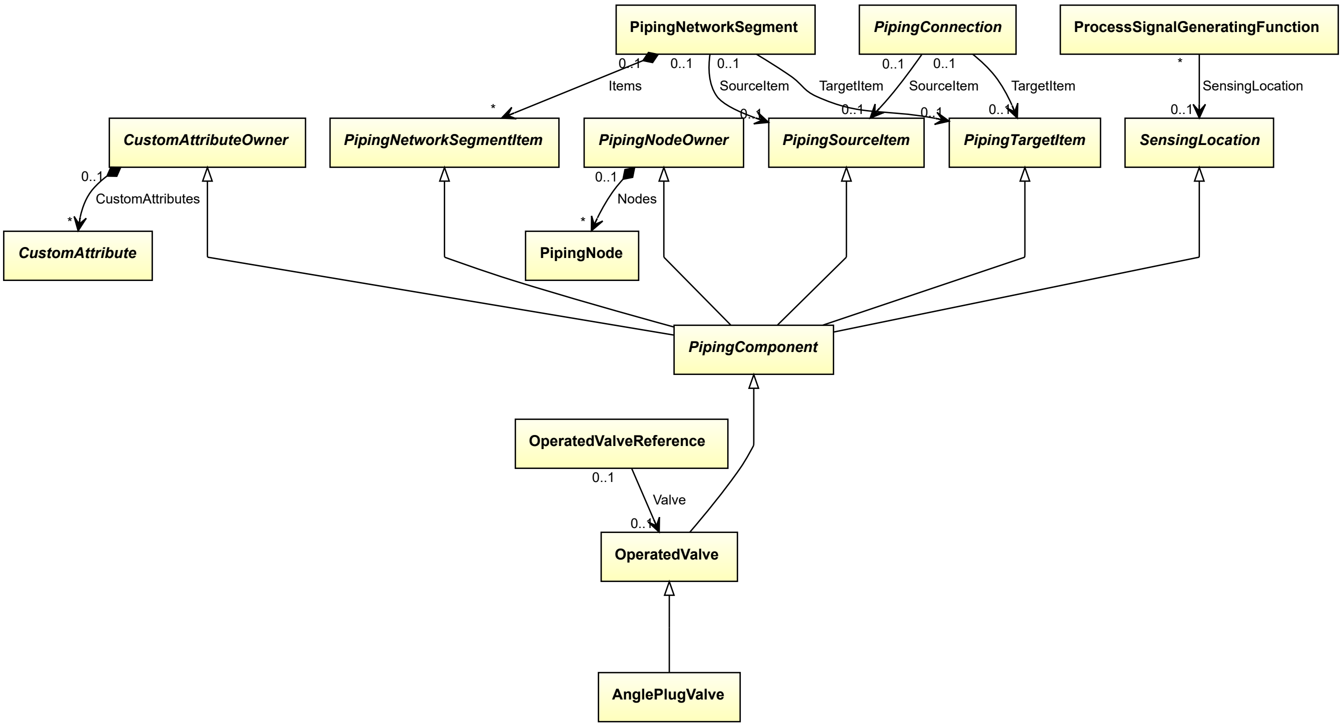
```
<PipingComponent
    ID="angleGlobeValve1"
    ComponentClass="AngleGlobeValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS882944" ...>
  ...
</PipingComponent>
```

8.3. AnglePlugValve

8.3.1 Overview

Class

A valve that has valve ports which are not in-line and that has a quarter turn action in which the closure member is a cylindrical or tapered plug which operates by rotating on its axis and sealing against a downstream seat.



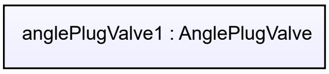
Supertypes

- *OperatedValve*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <PipingComponent>
RDL reference: ANGLE PLUG VALVE
ComponentClass: AnglePlugValve
ComponentClassURI: http://sandbox.dexpi.org/rdl/AnglePlugValve

Example



Example: Implementation in Proteus Schema

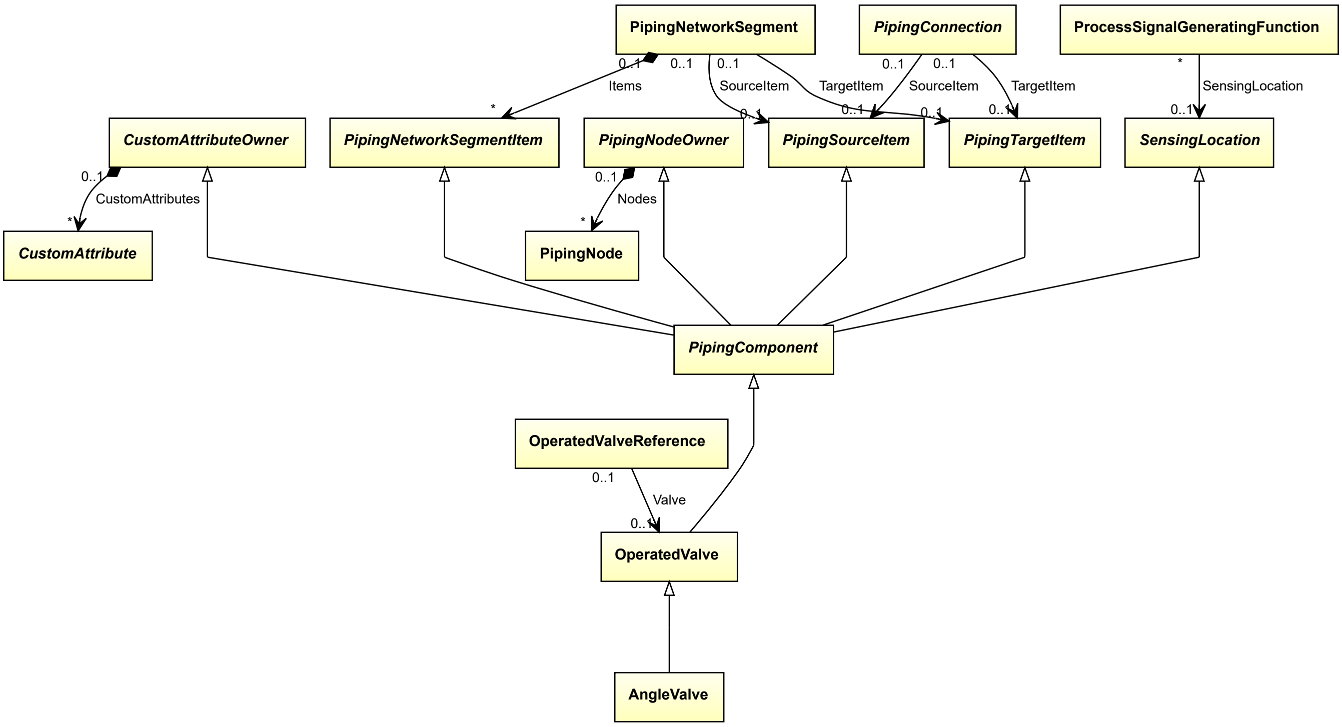
```
<PipingComponent
    ID="anglePlugValve1"
    ComponentClass="AnglePlugValve"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/AnglePlugValve" ...>
  ...
</PipingComponent>
```

8.4. AngleValve

8.4.1 Overview

Class

A valve that has valve ports which are not in-line (from http://data.posccaesar.org/rdl/RDS5789384).



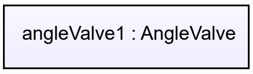
Supertypes

- *OperatedValve*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <PipingComponent>
RDL reference: ANGLE VALVE
ComponentClass: AngleValve
ComponentClassURI: http://data.posccaesar.org/rdl/RDS5789384

Example



Example: Implementation in Proteus Schema

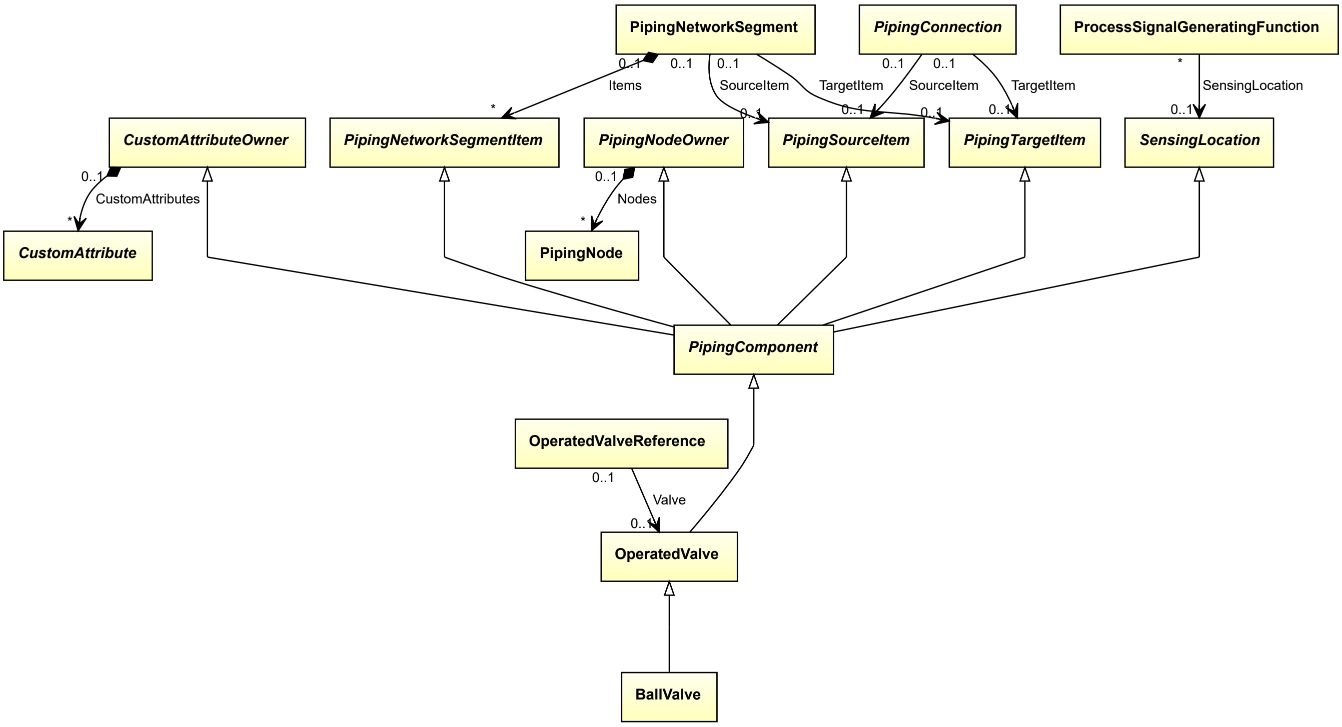
```
<PipingComponent
    ID="angleValve1"
    ComponentClass="AngleValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS5789384" ...>
  ...
</PipingComponent>
```

8.5. BallValve

8.5.1 Overview

Class

A rotary valve that has a ball closure member (from http://data.posccaesar.org/rdl/RDS416654).



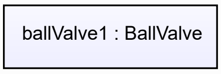
Supertypes

- *OperatedValve*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingComponent>`
RDL reference: BALL VALVE
ComponentClass: BallValve
ComponentClassURI: http://data.posccaesar.org/rdl/RDS416654

Example



Example: Implementation in Proteus Schema

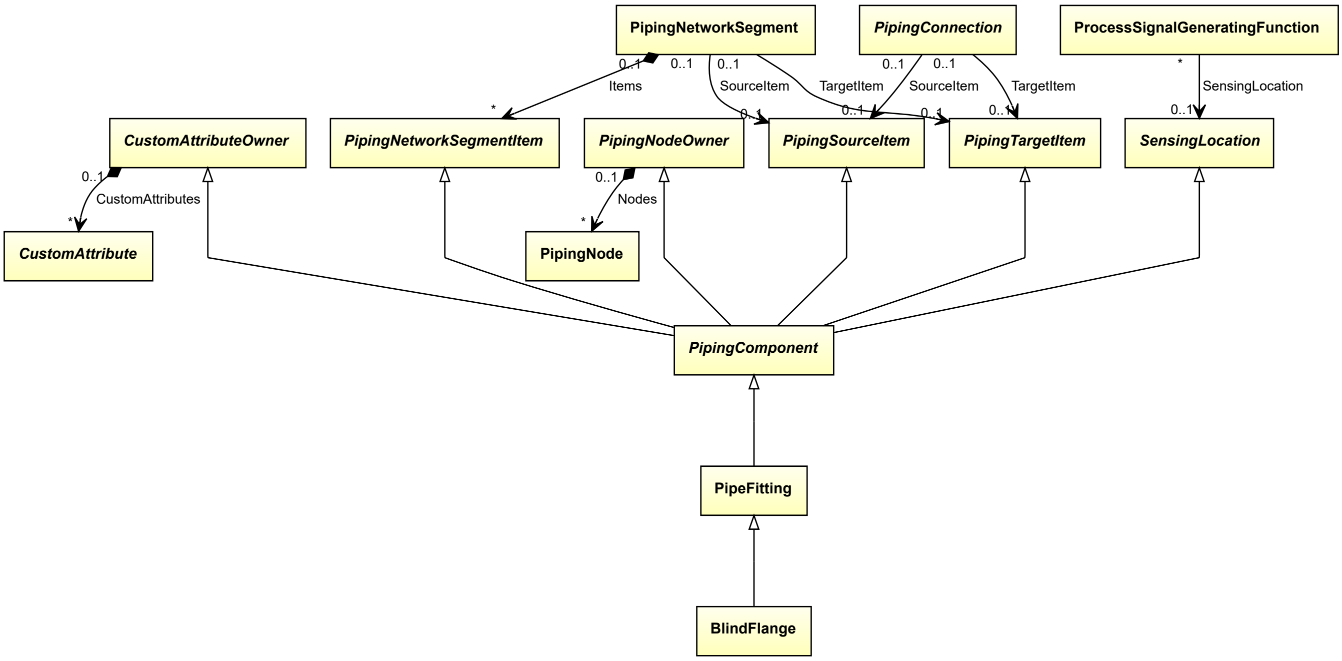
```
<PipingComponent
    ID="ballValve1"
    ComponentClass="BallValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS416654" ...>
  ...
</PipingComponent>
```

8.6. BlindFlange

8.6.1 Overview

Class

A pipe flange that is without a central opening and used to shut off a flanged pipe end (from http://data.posccaesar.org/rdl/RDS414719).



Supertypes

- *PipeFitting*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <PipingComponent>
RDL reference: BLIND FLANGE
ComponentClass: BlindFlange
ComponentClassURI: http://data.posccaesar.org/rdl/RDS414719

Example

Example: Implementation in Proteus Schema

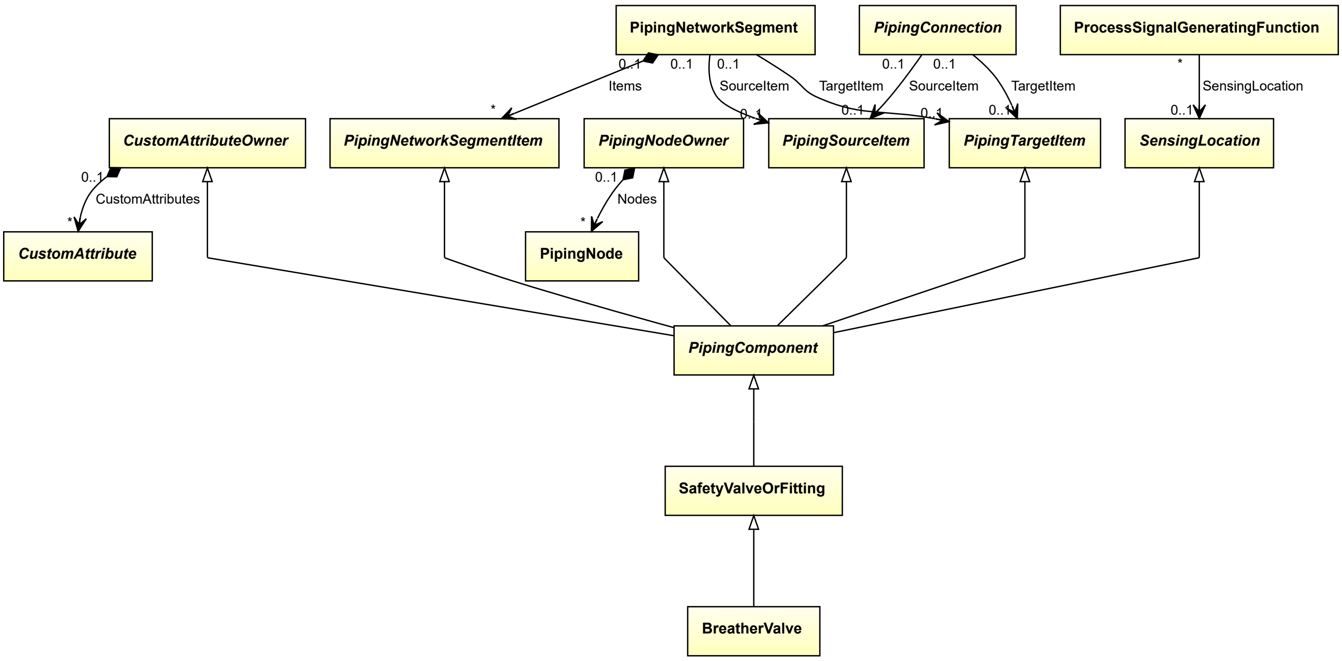
```
<PipingComponent
    ID="blindFlange1"
    ComponentClass="BlindFlange"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS414719" ...>
  ...
</PipingComponent>
```

8.7. BreatherValve

8.7.1 Overview

Class

A breather valve.



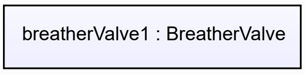
Supertypes

- *SafetyValveOrFitting*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingComponent>`
RDL reference: BREATHER VALVE
ComponentClass: BreatherValve
ComponentClassURI: http://sandbox.dexpi.org/rdl/BreatherValve

Example



Example: Implementation in Proteus Schema

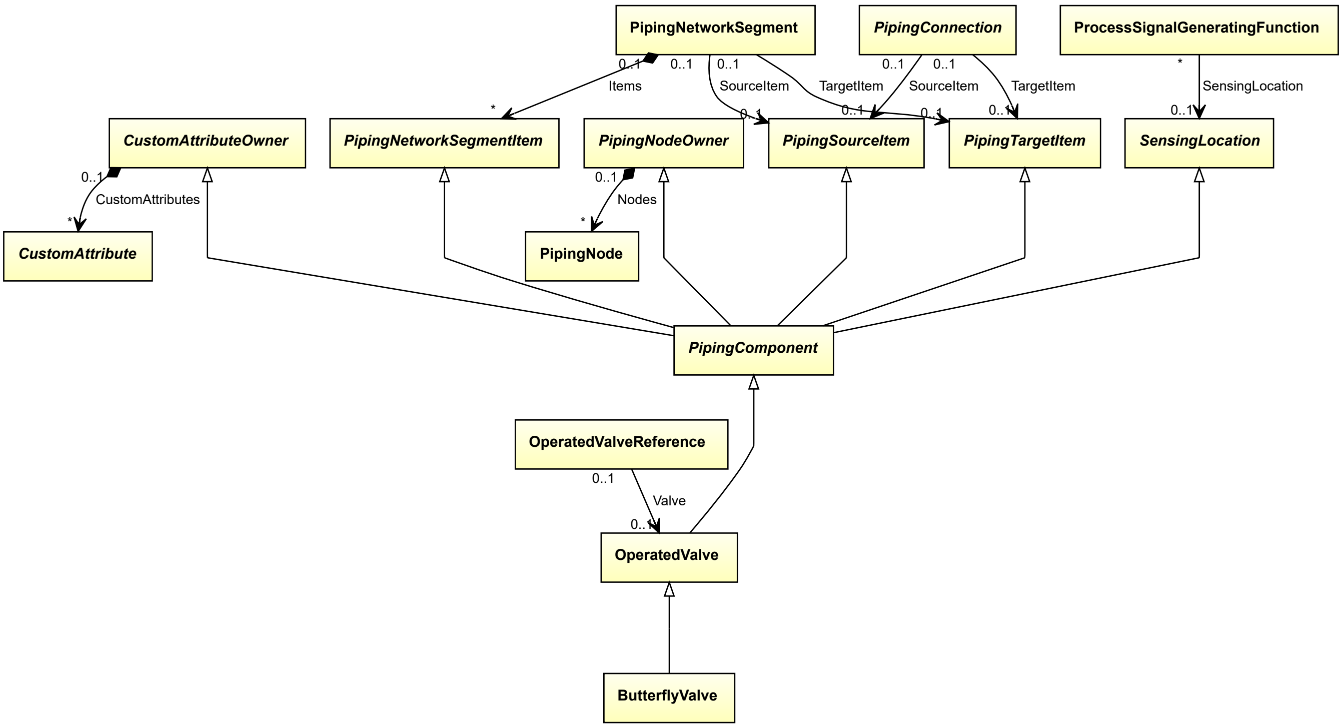
```
<PipingComponent
    ID="breatherValve1"
    ComponentClass="BreatherValve"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/BreatherValve" ...>
  ...
</PipingComponent>
```

8.8. ButterflyValve

8.8.1 Overview

Class

A rotary valve that has a closure member of a disc type with a shaft parallel, or near parallel, to the plane of the disc, with an axis of rotation transverse to the flow direction (from http://data.posccaesar.org/rdl/RDS416609).



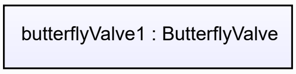
Supertypes

- *OperatedValve*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingComponent>`
RDL reference: BUTTERFLY VALVE
ComponentClass: ButterflyValve
ComponentClassURI: http://data.posccaesar.org/rdl/RDS416609

Example



Example: Implementation in Proteus Schema

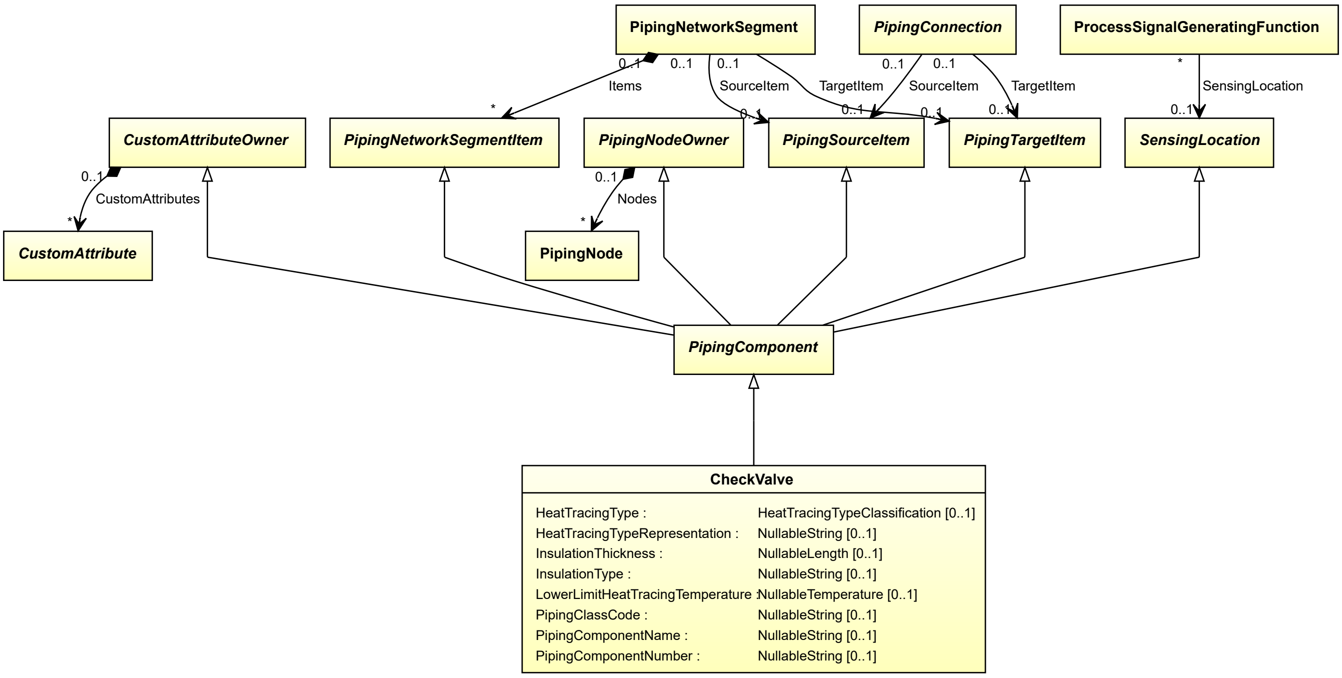
```
<PipingComponent
    ID="butterflyValve1"
    ComponentClass="ButterflyValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS416609" ...>
  ...
</PipingComponent>
```

8.9. CheckValve

8.9.1 Overview

Class

A valve that permits fluid to flow in one direction only (from http://data.posccaesar.org/rdl/RDS292229).


PipingNetworkSegment
PipingConnection
ProcessSignalGeneratingFunction
0..1
0..1
0..1
Items
SourceItem
TargetItem
0..1
0..1
SourceItem
TargetItem
*
SensingLocation
*
0..1
0..1
0..1
0..1
0..1
CustomAttributeOwner
PipingNetworkSegmentItem
PipingNodeOwner
PipingSourceItem
PipingTargetItem
SensingLocation
0..1
CustomAttributes
*
CustomAttribute
0..1
Nodes
*
PipingNode
PipingComponent
CheckValve
HeatTracingType : HeatTracingTypeClassification [0..1]
HeatTracingTypeRepresentation : NullableString [0..1]
InsulationThickness : NullableLength [0..1]
InsulationType : NullableString [0..1]
LowerLimitHeatTracingTemperature :NullableTemperature [0..1]
PipingClassCode : NullableString [0..1]
PipingComponentName : NullableString [0..1]
PipingComponentNumber : NullableString [0..1]

Supertypes

- *PipingComponent*

Subtypes

- *CustomCheckValve*
- *GlobeCheckValve*
- *SwingCheckValve*

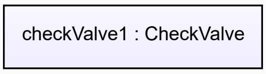
Attributes (data)

Name	Multiplicity	Type
HeatTracingType	0..1	*HeatTracingTypeClassification*
HeatTracingTypeRepresentation	0..1	*NullableString*
InsulationThickness	0..1	*NullableLength*
InsulationType	0..1	*NullableString*
LowerLimitHeatTracingTemperature	0..1	*NullableTemperature*
PipingClassCode	0..1	*NullableString*
PipingComponentName	0..1	*NullableString*
PipingComponentNumber	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <PipingComponent>
RDL reference: CHECK VALVE
ComponentClass: CheckValve
ComponentClassURI: http://data.posccaesar.org/rdl/RDS292229

Example



Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="checkValve1"
    ComponentClass="CheckValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS292229" ...>
  ...
</PipingComponent>
```

8.9.2 HeatTracingType

Attribute (data)

A specialization indicating the heat tracing type related to the *CheckValve*.

Multiplicity: 0..1

Type: *HeatTracingTypeClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: HEAT TRACING TYPE SPECIALIZATION
Name: HeatTracingTypeSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/HeatTracingTypeSpecialization

Example

electrical heat tracing system (*HeatTracingTypeClassification::ElectricalHeatTracingSystem*)

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="checkValve1"
    ComponentClass="CheckValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS292229" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="HeatTracingTypeSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/HeatTracingTypeSpecialization"
      Format="anyURI"
      Value="ElectricalHeatTracingSystem"
      ValueURI="http://data.posccaesar.org/rdl/RDS11854600" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.9.3 HeatTracingTypeRepresentation

Attribute (data)

The heat tracing type related to the *CheckValve*, represented as a string.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: HEAT TRACING TYPE REPRESENTATION ASSIGNMENT CLASS
Name: HeatTracingTypeRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/HeatTracingTypeRepresentationAssignmentClass

Example

“E” (*String*)

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="checkValve1"
    ComponentClass="CheckValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS292229" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="HeatTracingTypeRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/HeatTracingTypeRepresentationAssignmentClass"
      Format="string"
      Value="E" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.9.4 InsulationThickness

Attribute (data)

The insulation thickness of the *CheckValve*.

Multiplicity: 0..1

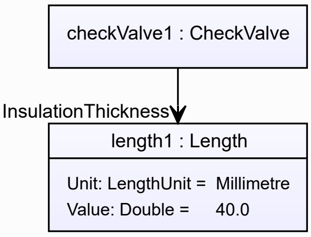
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: INSULATION THICKNESS
Name: InsulationThickness
AttributeURI: http://data.posccaesar.org/rdl/RDS4238040

Example

The instance checkValve1 represents a *CheckValve* with an *InsulationThickness* of 40.0 mm.



Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="checkValve1"
    ComponentClass="CheckValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS292229" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="InsulationThickness"
      AttributeURI="http://data.posccaesar.org/rdl/RDS4238040"
      Format="double"
      Value="40.0"
      Units="Millimetre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1357739" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.9.5 InsulationType

Attribute (data)

The identification code for the insulation type related to the *CheckValve*. So far, DEXPI does not define restrictions for valid values.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: INSULATION TYPE ASSIGNMENT CLASS
Name: InsulationTypeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/InsulationTypeAssignmentClass

Example

"Q" (*String*)

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="checkValve1"
    ComponentClass="CheckValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS292229" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="InsulationTypeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/InsulationTypeAssignmentClass"
      Format="string"
      Value="Q" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.9.6 LowerLimitHeatTracingTemperature

Attribute (data)

The lower limit for the temperature that a heat tracing system must ensure for the *CheckValve*.

Multiplicity: 0..1

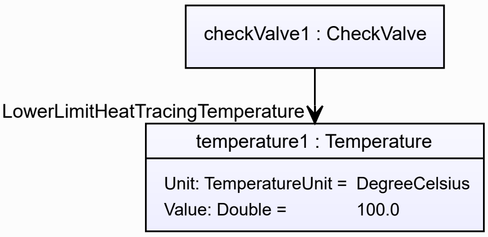
Type: *NullableTemperature*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: LOWER LIMIT HEAT TRACING TEMPERATURE
Name: LowerLimitHeatTracingTemperature
AttributeURI: http://sandbox.dexpi.org/rdl/LowerLimitHeatTracingTemperature

Example

The instance checkValve1 represents a *CheckValve* with a *LowerLimitHeatTracingTemperature* of 100.0 °C.


checkValve1 : CheckValve
LowerLimitHeatTracingTemperature
temperature1 : Temperature
Unit: TemperatureUnit = DegreeCelsius
Value: Double = 100.0

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="checkValve1"
    ComponentClass="CheckValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS292229" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="LowerLimitHeatTracingTemperature"
      AttributeURI="http://sandbox.dexpi.org/rdl/LowerLimitHeatTracingTemperature"
      Format="double"
      Value="100.0"
      Units="DegreeCelsius"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1322684" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.9.7 PipingClassCode

Attribute (data)

The identification code of the piping class of the *CheckValve*. So far, DEXPI does not define restrictions for valid values.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PIPING CLASS CODE ASSIGNMENT CLASS
Name: PipingClassCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PipingClassCodeAssignmentClass

Example

“75HB13” (*String*)

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="checkValve1"
    ComponentClass="CheckValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS292229" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PipingClassCodeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PipingClassCodeAssignmentClass"
      Format="string"
      Value="75HB13" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.9.8 PipingComponentName

Attribute (data)

A string to classify the *CheckValve*. DEXPI does not prescribe the classification system. Typically, company or site site standards are used.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PIPING COMPONENT NAME ASSIGNMENT CLASS
Name: PipingComponentNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PipingComponentNameAssignmentClass

Example

“73KH12” (*String*)

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="checkValve1"
    ComponentClass="CheckValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS292229" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PipingComponentNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PipingComponentNameAssignmentClass"
      Format="string"
      Value="73KH12" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.9.9 PipingComponentNumber

Attribute (data)

An identifier of the *CheckValve*. DEXPI does not prescribe the scope of the identifier, i.e., whether it should be unique in, e.g., a *PipingNetworkSegment* or a *PipingNetworkSystem*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PIPING COMPONENT NUMBER ASSIGNMENT CLASS
Name: PipingComponentNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PipingComponentNumberAssignmentClass

Example

"C2" (*String*)

Example: Implementation in Proteus Schema

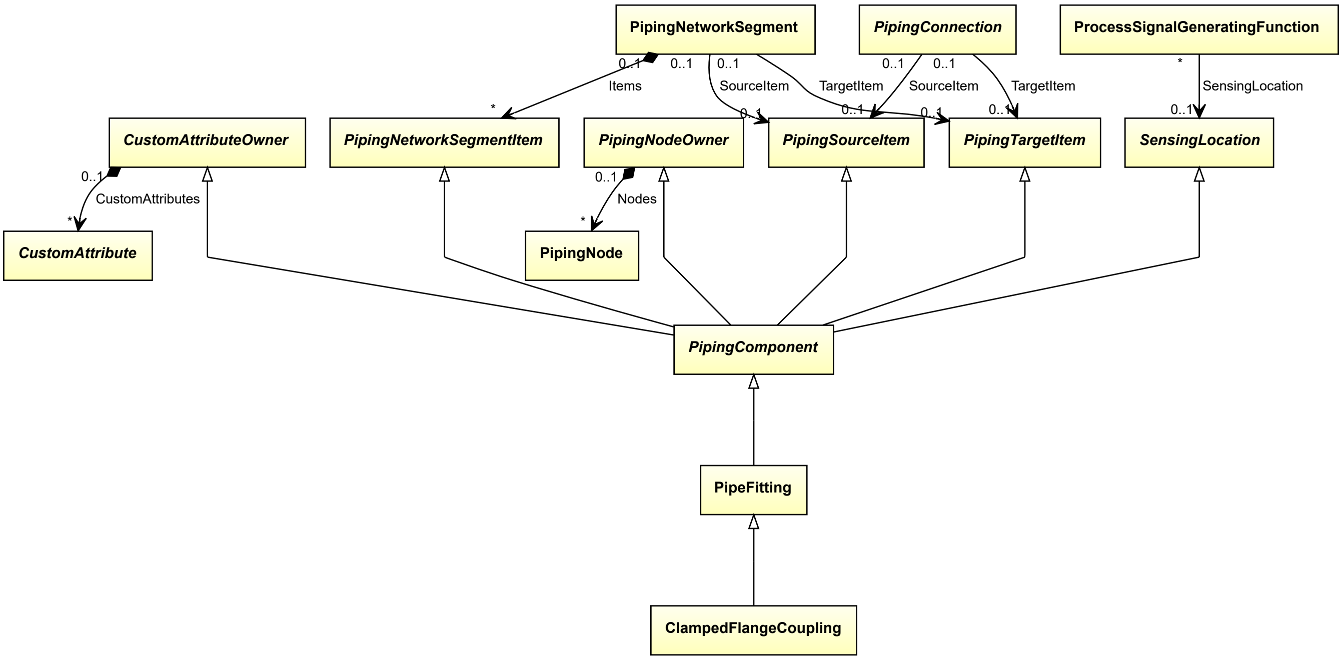
```
<PipingComponent
    ID="checkValve1"
    ComponentClass="CheckValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS292229" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PipingComponentNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PipingComponentNumberAssignmentClass"
      Format="string"
      Value="C2" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.10. ClampedFlangeCoupling

8.10.1 Overview

Class

A clamped flange coupling.



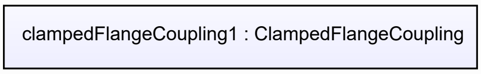
Supertypes

- *PipeFitting*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <PipingComponent>
RDL reference: CLAMPED FLANGE COUPLING
ComponentClass: ClampedFlangeCoupling
ComponentClassURI: http://sandbox.dexpi.org/rdl/ClampedFlangeCoupling

Example



Example: Implementation in Proteus Schema

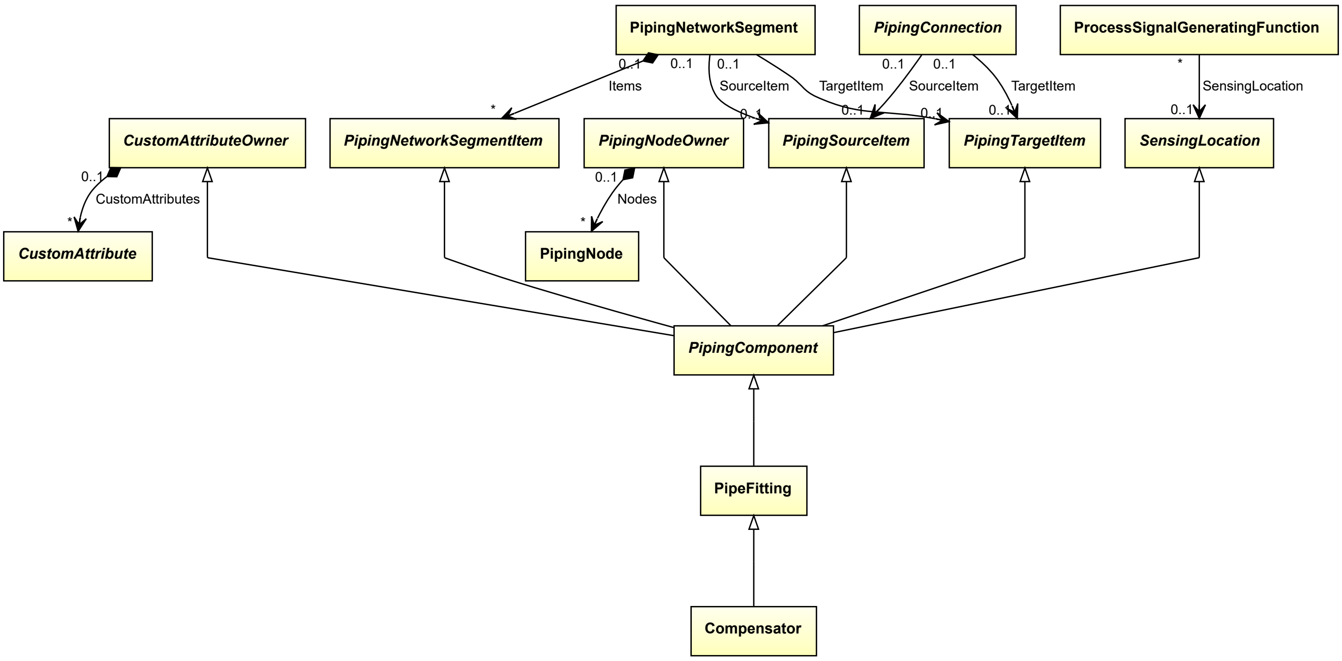
```
<PipingComponent
    ID="clampedFlangeCoupling1"
    ComponentClass="ClampedFlangeCoupling"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ClampedFlangeCoupling" ...>
  ...
</PipingComponent>
```

8.11. Compensator

8.11.1 Overview

Class

A device compensating for axial or radial movement between two elements that is connected (from http://data.posccaesar.org/rdl/RDS1280084541).



Supertypes

- *PipeFitting*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <PipingComponent>
RDL reference: COMPENSATOR
ComponentClass: Compensator
ComponentClassURI: http://data.posccaesar.org/rdl/RDS1280084541

Example

Example: Implementation in Proteus Schema

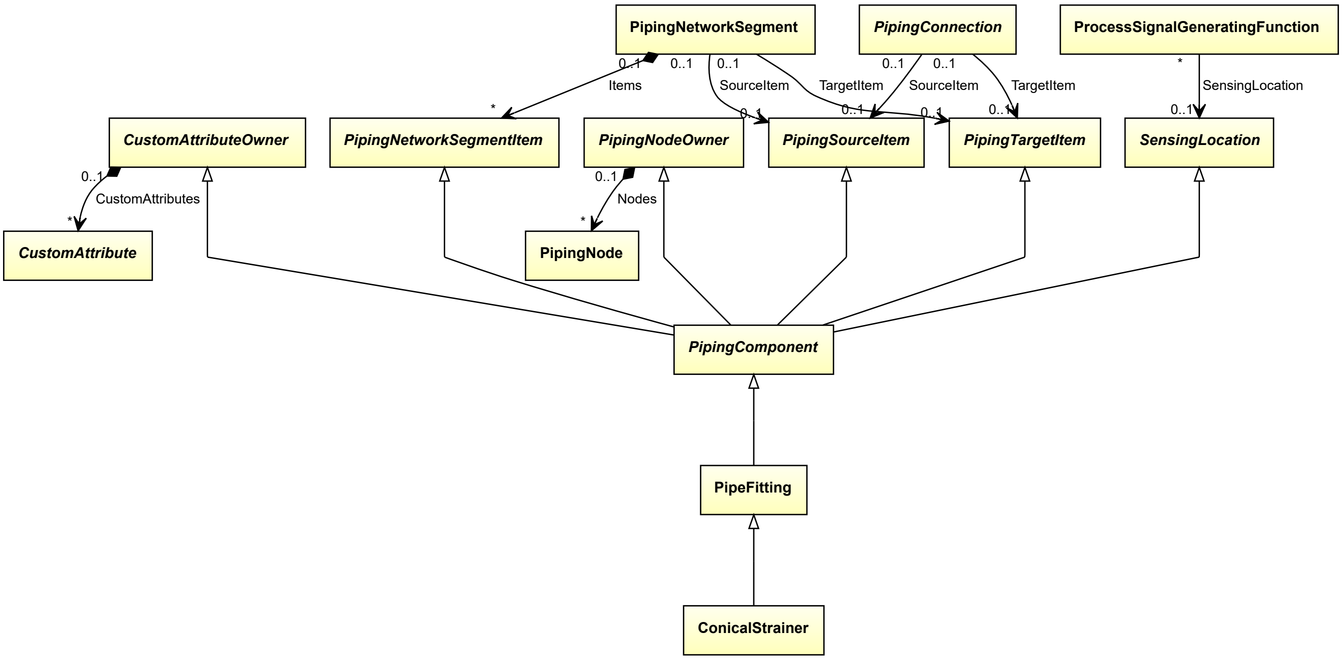
```
<PipingComponent
    ID="compensator1"
    ComponentClass="Compensator"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS1280084541" ...>
  ...
</PipingComponent>
```

8.12. ConicalStrainer

8.12.1 Overview

Class

A strainer where the screen has a conical tubular shape (from http://data.posccaesar.org/rdl/RDS16044540).



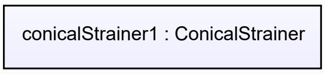
Supertypes

- *PipeFitting*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingComponent>`
RDL reference: CONICAL STRAINER
ComponentClass: ConicalStrainer
ComponentClassURI: http://data.posccaesar.org/rdl/RDS16044540

Example



Example: Implementation in Proteus Schema

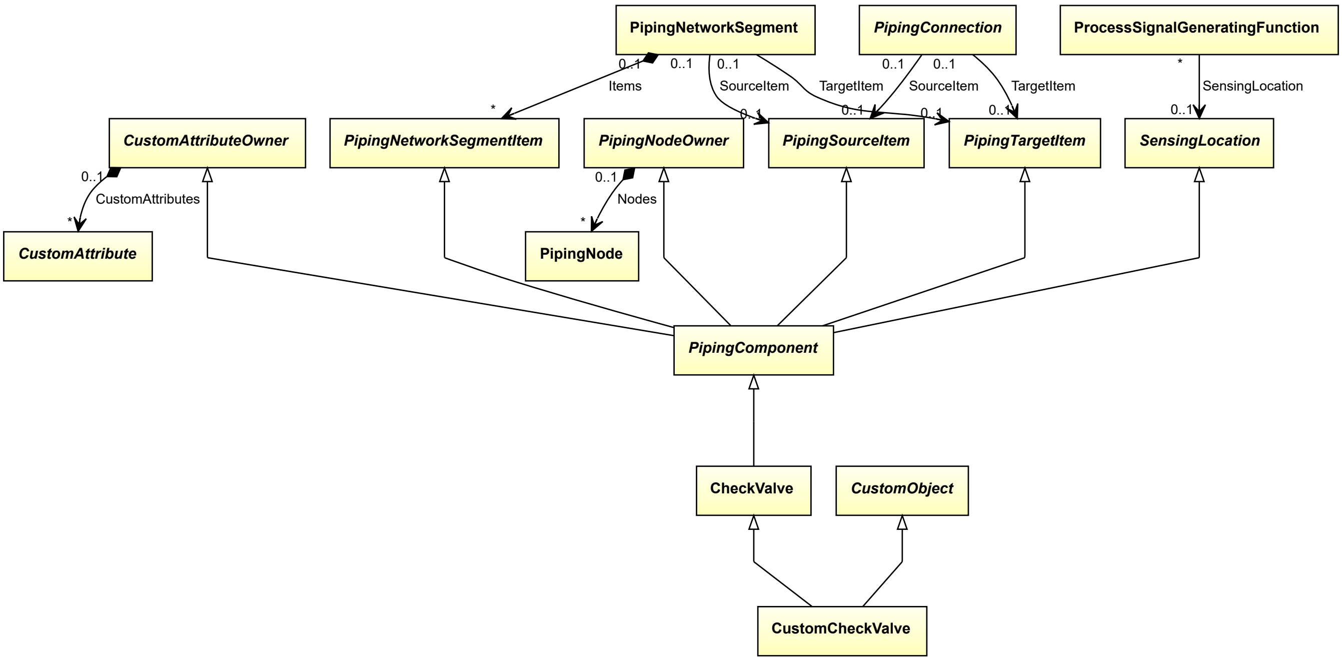
```
<PipingComponent
    ID="conicalStrainer1"
    ComponentClass="ConicalStrainer"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS16044540" ...>
  ...
</PipingComponent>
```

8.13. CustomCheckValve

8.13.1 Overview

Class

A custom *CheckValve*, i.e., a *CheckValve* that is not covered by any of the other subclasses of *CheckValve* (*GlobeCheckValve* or *SwingCheckValve*).



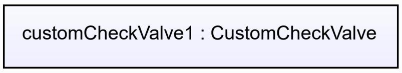
Supertypes

- *CheckValve*
- *CustomObject*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingComponent>`
RDL reference: CUSTOM CHECK VALVE
ComponentClass: CustomCheckValve
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomCheckValve

Example



Example: Implementation in Proteus Schema

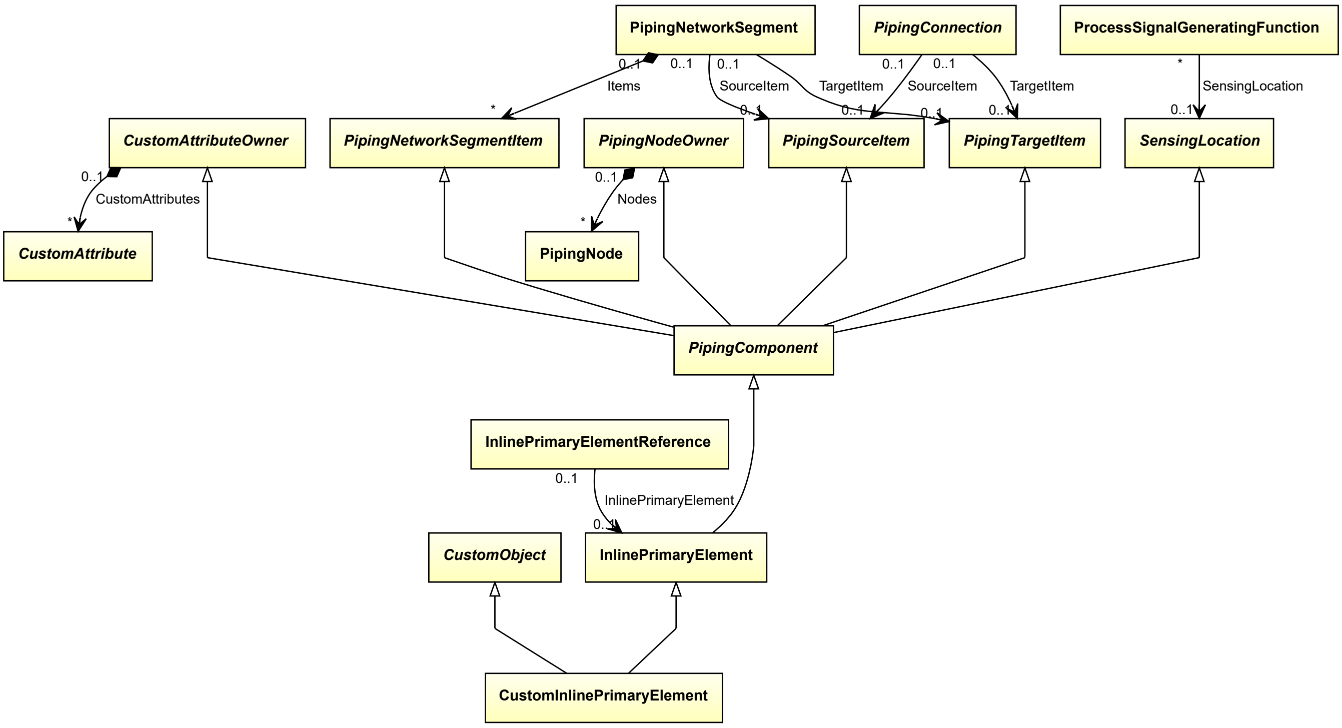
```
<PipingComponent
    ID="customCheckValve1"
    ComponentClass="CustomCheckValve"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomCheckValve" ...>
  ...
</PipingComponent>
```

8.14. CustomInlinePrimaryElement

8.14.1 Overview

Class

A custom *InlinePrimaryElement*, i.e., an *InlinePrimaryElement* that is not covered by any of the other subclasses of *InlinePrimaryElement*.



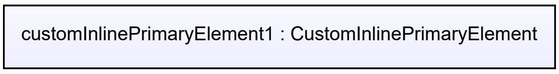
Supertypes

- *CustomObject*
- *InlinePrimaryElement*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingComponent>`
RDL reference: CUSTOM INLINE PRIMARY ELEMENT
ComponentClass: CustomInlinePrimaryElement
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomInlinePrimaryElement

Example



Example: Implementation in Proteus Schema

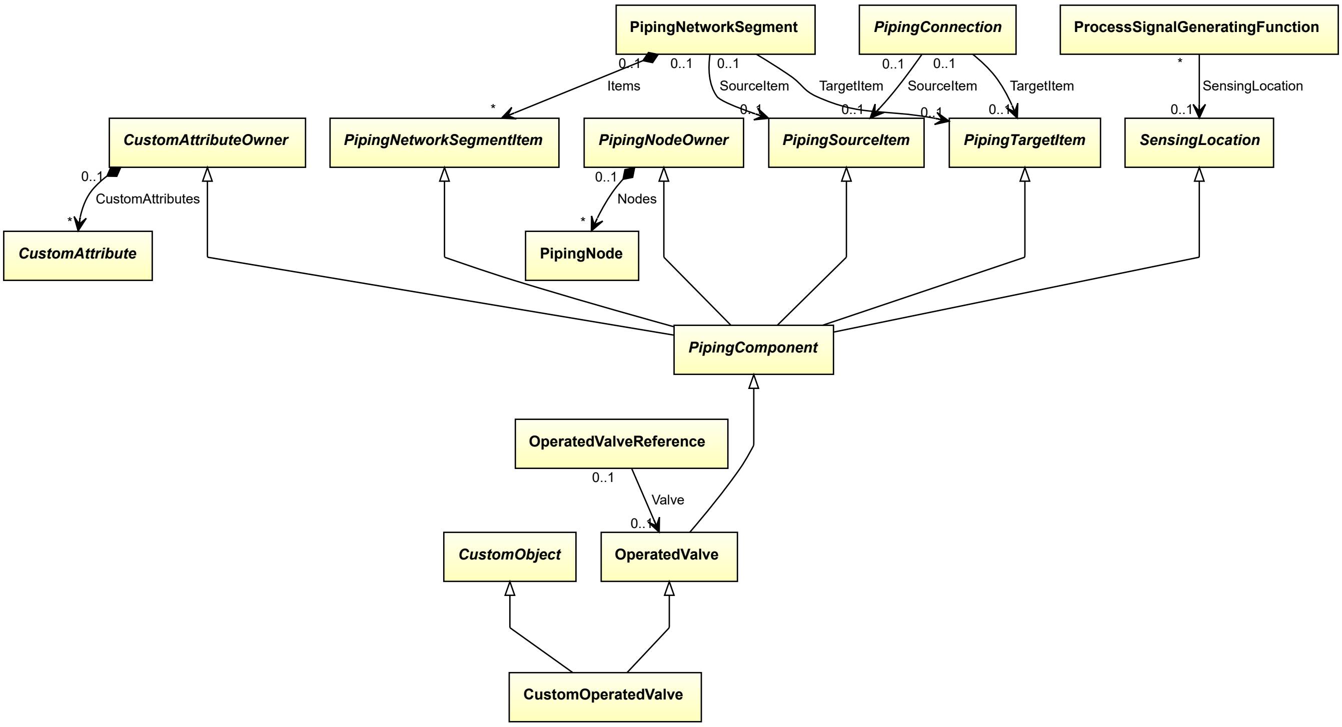
```
<PipingComponent
    ID="customInlinePrimaryElement1"
    ComponentClass="CustomInlinePrimaryElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomInlinePrimaryElement" ...>
  ...
</PipingComponent>
```

8.15. CustomOperatedValve

8.15.1 Overview

Class

A custom *OperatedValve*, i.e., an *OperatedValve* that is not covered by any of the other subclasses of *OperatedValve*.



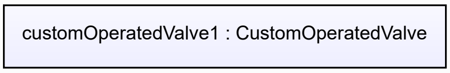
Supertypes

- *CustomObject*
- *OperatedValve*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingComponent>`
RDL reference: CUSTOM OPERATED VALVE
ComponentClass: CustomOperatedValve
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomOperatedValve

Example



Example: Implementation in Proteus Schema

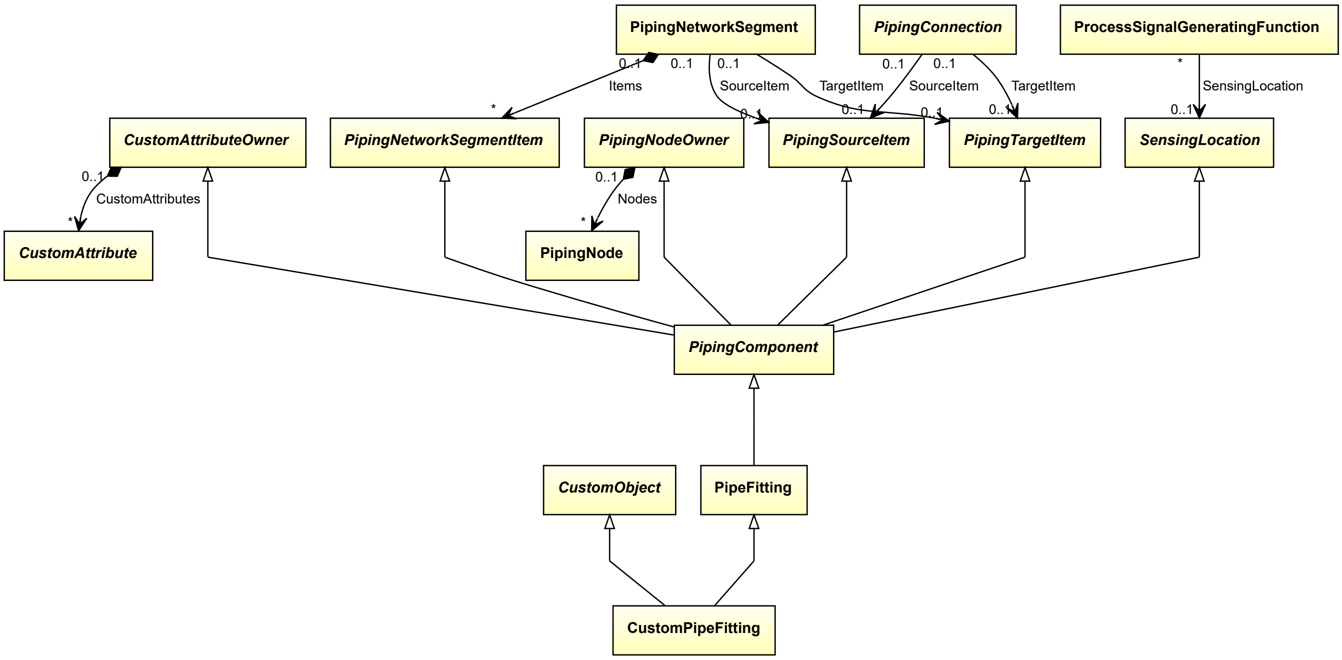
```
<PipingComponent
    ID="customOperatedValve1"
    ComponentClass="CustomOperatedValve"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomOperatedValve" ...>
  ...
</PipingComponent>
```

8.16. CustomPipeFitting

8.16.1 Overview

Class

A custom *PipeFitting*, i.e., a *PipeFitting* that is not covered by any of the other subclasses of *PipeFitting*.



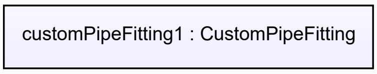
Supertypes

- *CustomObject*
- *PipeFitting*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <PipingComponent>
RDL reference: CUSTOM PIPE FITTING
ComponentClass: CustomPipeFitting
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomPipeFitting

Example



Example: Implementation in Proteus Schema

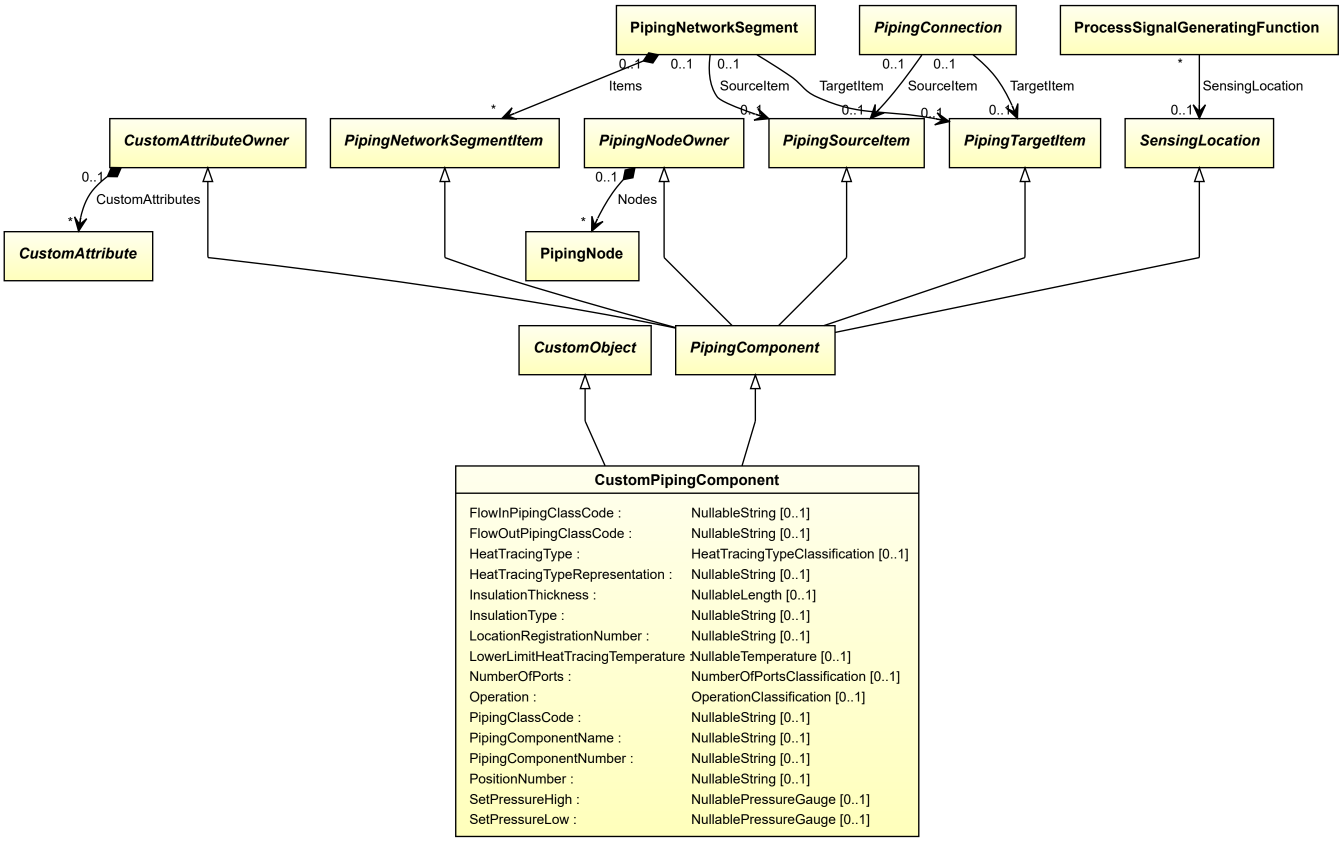
```
<PipingComponent
    ID="customPipeFitting1"
    ComponentClass="CustomPipeFitting"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomPipeFitting" ...>
  ...
</PipingComponent>
```

8.17. CustomPipingComponent

8.17.1 Overview

Class

A custom *PipingComponent*, i.e., a *PipingComponent* that is not covered by any of the other subclasses of *PipingComponent* (*CheckValve*, *InlinePrimaryElement*, *OperatedValve*, *PipeFitting*, or *SafetyValveOrFitting*).


PipingNetworkSegment
PipingConnection
ProcessSignalGeneratingFunction
0..1
0..1
0..1
Items
SourceItem
TargetItem
0..1
0..1
SourceItem
TargetItem
*
SensingLocation
*
0..1
0..1
0..1
0..1
0..1
CustomAttributeOwner
PipingNetworkSegmentItem
PipingNodeOwner
PipingSourceItem
PipingTargetItem
SensingLocation
0..1
CustomAttributes
*
CustomAttribute
0..1
Nodes
*
PipingNode
CustomObject
PipingComponent
CustomPipingComponent
FlowInPipingClassCode : NullableString [0..1]
FlowOutPipingClassCode : NullableString [0..1]
HeatTracingType : HeatTracingTypeClassification [0..1]
HeatTracingTypeRepresentation : NullableString [0..1]
InsulationThickness : NullableLength [0..1]
InsulationType : NullableString [0..1]
LocationRegistrationNumber : NullableString [0..1]
LowerLimitHeatTracingTemperature :NullableTemperature [0..1]
NumberOfPorts : NumberOfPortsClassification [0..1]
Operation : OperationClassification [0..1]
PipingClassCode : NullableString [0..1]
PipingComponentName : NullableString [0..1]
PipingComponentNumber : NullableString [0..1]
PositionNumber : NullableString [0..1]
SetPressureHigh : NullablePressureGauge [0..1]
SetPressureLow : NullablePressureGauge [0..1]

Supertypes

- *CustomObject*
- *PipingComponent*

Attributes (data)

Name	Multiplicity	Type
FlowInPipingClassCode	0..1	*NullableString*
FlowOutPipingClassCode	0..1	*NullableString*
HeatTracingType	0..1	*HeatTracingTypeClassification*
HeatTracingTypeRepresentation	0..1	*NullableString*
InsulationThickness	0..1	*NullableLength*
InsulationType	0..1	*NullableString*
LocationRegistrationNumber	0..1	*NullableString*
LowerLimitHeatTracingTemperature	0..1	*NullableTemperature*
NumberOfPorts	0..1	*NumberOfPortsClassification*
Operation	0..1	*OperationClassification*
PipingClassCode	0..1	*NullableString*
PipingComponentName	0..1	*NullableString*

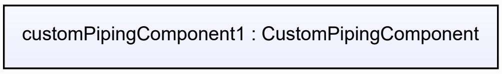
(continued on next page)

Name	Multiplicity	Type
PipingComponentNumber	0..1	*NullableString*
PositionNumber	0..1	*NullableString*
SetPressureHigh	0..1	*NullablePressureGauge*
SetPressureLow	0..1	*NullablePressureGauge*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <PipingComponent>
RDL reference: CUSTOM PIPING COMPONENT
ComponentClass: CustomPipingComponent
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomPipingComponent

Example



Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="customPipingComponent1"
    ComponentClass="CustomPipingComponent"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomPipingComponent" ...>
  ...
</PipingComponent>
```

8.17.2 FlowInPipingClassCode

Attribute (data)

The code of the piping class at the flow in side of *CustomPipingComponent*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: FLOW IN PIPING CLASS CODE ASSIGNMENT CLASS
Name: FlowInPipingClassCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/FlowInPipingClassCodeAssignmentClass

8.17.3 FlowOutPipingClassCode

Attribute (data)

The code of the piping class at the flow out side of *CustomPipingComponent*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: FLOW OUT PIPING CLASS CODE ASSIGNMENT CLASS
Name: FlowOutPipingClassCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/FlowOutPipingClassCodeAssignmentClass

8.17.4 HeatTracingType

Attribute (data)

A specialization indicating the heat tracing type related to the *CustomPipingComponent*.

Multiplicity: 0..1

Type: *HeatTracingTypeClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: HEAT TRACING TYPE SPECIALIZATION
Name: HeatTracingTypeSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/HeatTracingTypeSpecialization

8.17.5 HeatTracingTypeRepresentation

Attribute (data)

The heat tracing type related to the *CustomPipingComponent*, represented as a string.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: HEAT TRACING TYPE REPRESENTATION ASSIGNMENT CLASS
Name: HeatTracingTypeRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/HeatTracingTypeRepresentationAssignmentClass

8.17.6 InsulationThickness

Attribute (data)

The insulation thickness of the *CustomPipingComponent*.

Multiplicity: 0..1

Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: INSULATION THICKNESS
Name: InsulationThickness
AttributeURI: http://data.posccaesar.org/rdl/RDS4238040

8.17.7 InsulationType

Attribute (data)

The identification code for the insulation type related to the *CustomPipingComponent*. So far, DEXPI does not define restrictions for valid values.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: INSULATION TYPE ASSIGNMENT CLASS
Name: InsulationTypeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/InsulationTypeAssignmentClass

8.17.8 LocationRegistrationNumber

Attribute (data)

The location registration number of the *CustomPipingComponent*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: LOCATION REGISTRATION NUMBER ASSIGNMENT CLASS
Name: LocationRegistrationNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/LocationRegistrationNumberAssignmentClass

8.17.9 LowerLimitHeatTracingTemperature

Attribute (data)

The lower limit for the temperature that a heat tracing system must ensure for the *CustomPipingComponent*.

Multiplicity: 0..1

Type: *NullableTemperature*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: LOWER LIMIT HEAT TRACING TEMPERATURE
Name: LowerLimitHeatTracingTemperature
AttributeURI: http://sandbox.dexpi.org/rdl/LowerLimitHeatTracingTemperature

8.17.10 NumberOfPorts

Attribute (data)

A specialization indicating the number of ports of the *CustomPipingComponent*.

Multiplicity: 0..1

Type: *NumberOfPortsClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: NUMBER OF PORTS SPECIALIZATION
Name: NumberOfPortsSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/NumberOfPortsSpecialization

8.17.11 Operation

Attribute (data)

A specialization indicating the operation of the *CustomPipingComponent*.

Multiplicity: 0..1

Type: *OperationClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: OPERATION SPECIALIZATION
Name: OperationSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/OperationSpecialization

8.17.12 PipingClassCode

Attribute (data)

The identification code of the piping class of the *CustomPipingComponent*. So far, DEXPI does not define restrictions for valid values.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PIPING CLASS CODE ASSIGNMENT CLASS
Name: PipingClassCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PipingClassCodeAssignmentClass

8.17.13 PipingComponentName

Attribute (data)

A string to classify the *CustomPipingComponent*. DEXPI does not prescribe the classification system. Typically, company or site site standards are used.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PIPING COMPONENT NAME ASSIGNMENT CLASS
Name: PipingComponentNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PipingComponentNameAssignmentClass

8.17.14 PipingComponentNumber

Attribute (data)

An identifier of the *CustomPipingComponent*. DEXPI does not prescribe the scope of the identifier, i.e., whether it should be unique in, e.g., a *CustomPipingComponent* or a *CustomPipingComponent*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PIPING COMPONENT NUMBER ASSIGNMENT CLASS
Name: PipingComponentNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PipingComponentNumberAssignmentClass

8.17.15 PositionNumber

Attribute (data)

The position number of the *CustomPipingComponent*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: POSITION NUMBER ASSIGNMENT CLASS
Name: PositionNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PositionNumberAssignmentClass

8.17.16 SetPressureHigh

Attribute (data)

The high pressure at which the *CustomPipingComponent* is activated.

Multiplicity: 0..1

Type: *NullablePressureGauge*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: SET PRESSURE HIGH
Name: SetPressureHigh
AttributeURI: http://sandbox.dexpi.org/rdl/SetPressureHigh

8.17.17 SetPressureLow

Attribute (data)

The low pressure at which the *CustomPipingComponent* is activated.

Multiplicity: 0..1

Type: *NullablePressureGauge*

Implementation in Proteus Schema

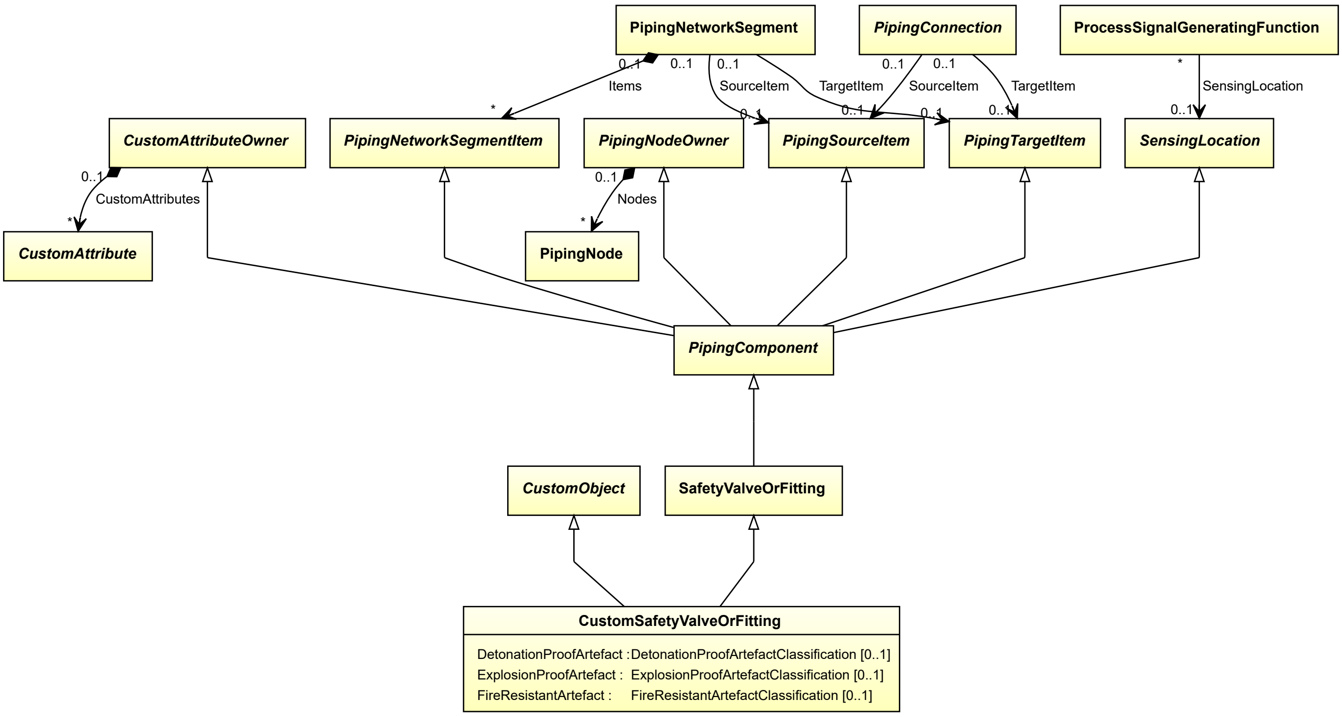
The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: SET PRESSURE LOW
Name: SetPressureLow
AttributeURI: http://sandbox.dexpi.org/rdl/SetPressureLow

8.18. CustomSafetyValveOrFitting

8.18.1 Overview

Class

A custom *SafetyValveOrFitting*, i.e., a *SafetyValveOrFitting* that is not covered by any of the other subclasses of *SafetyValveOrFitting* (*BreatherValve*, *FlameArrestor*, *RuptureDisc*, *SpringLoadedAngleGlobeSafetyValve*, or *SpringLoadedGlobeSafetyValve*).



Supertypes

- *CustomObject*
- *SafetyValveOrFitting*

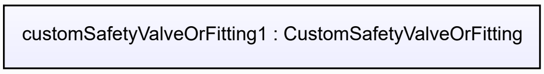
Attributes (data)

Name	Multiplicity	Type
DetonationProofArtefact	0..1	*DetonationProofArtefactClassification*
ExplosionProofArtefact	0..1	*ExplosionProofArtefactClassification*
FireResistantArtefact	0..1	*FireResistantArtefactClassification*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingComponent>`
RDL reference: CUSTOM SAFETY VALVE OR FITTING
ComponentClass: CustomSafetyValveOrFitting
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomSafetyValveOrFitting

Example



Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="customSafetyValveOrFitting1"
    ComponentClass="CustomSafetyValveOrFitting"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomSafetyValveOrFitting" ...>
  ...
</PipingComponent>
```

8.18.2 DetonationProofArtefact

Attribute (data)

A specialization indicating if the *CustomSafetyValveOrFitting* is detonation-proof.

Multiplicity: 0..1

Type: *DetonationProofArtefactClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: DETONATION PROOF ARTEFACT SPECIALIZATION
Name: DetonationProofArtefactSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/DetonationProofArtefactSpecialization

8.18.3 ExplosionProofArtefact

Attribute (data)

A specialization indicating if the *CustomSafetyValveOrFitting* is explosion-proof.

Multiplicity: 0..1

Type: *ExplosionProofArtefactClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: EXPLOSION PROOF ARTEFACT SPECIALIZATION
Name: ExplosionProofArtefactSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/ExplosionProofArtefactSpecialization

8.18.4 FireResistantArtefact

Attribute (data)

A specialization indicating if the *CustomSafetyValveOrFitting* is fire-resistant.

Multiplicity: 0..1

Type: *FireResistantArtefactClassification*

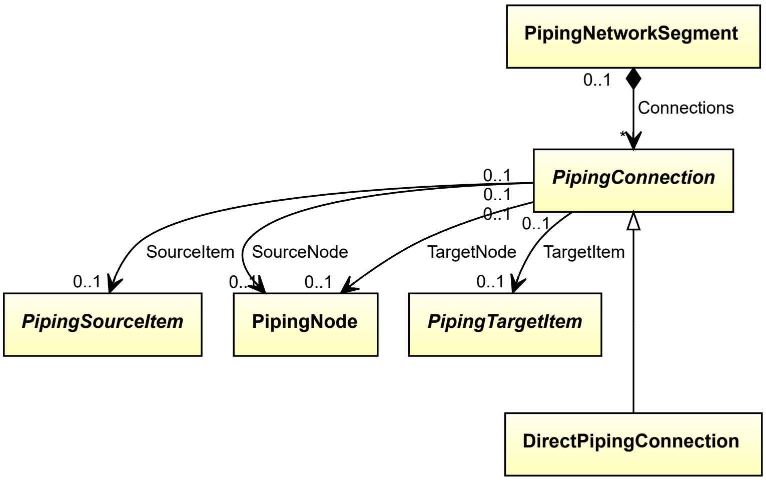
> Implementation in Proteus Schema
>
> The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
> **RDL reference:** FIRE RESISTANT ARTEFACT SPECIALIZATION
> **Name:** FireResistantArtefactSpecialization
> **AttributeURI:** http://sandbox.dexpi.org/rdl/FireResistantArtefactSpecialization

8.19. DirectPipingConnection

8.19.1 Overview

Class

A direct connection between two piping items, i.e. a connection that is not realized by a pipe.



Supertypes

- *PipingConnection*

> Implementation in Proteus Schema
>
> There is no direct implementation of *DirectPipingConnection* in Proteus Schema. A *DirectPipingConnection* rather corresponds to cases where Proteus Schema allows direct connections between piping-related items without a `<CenterLine>` element (corresponding to a *Pipe*) between, e.g., between two *PipingComponents* or between a *PipingComponent* and a *PropertyBreak*.

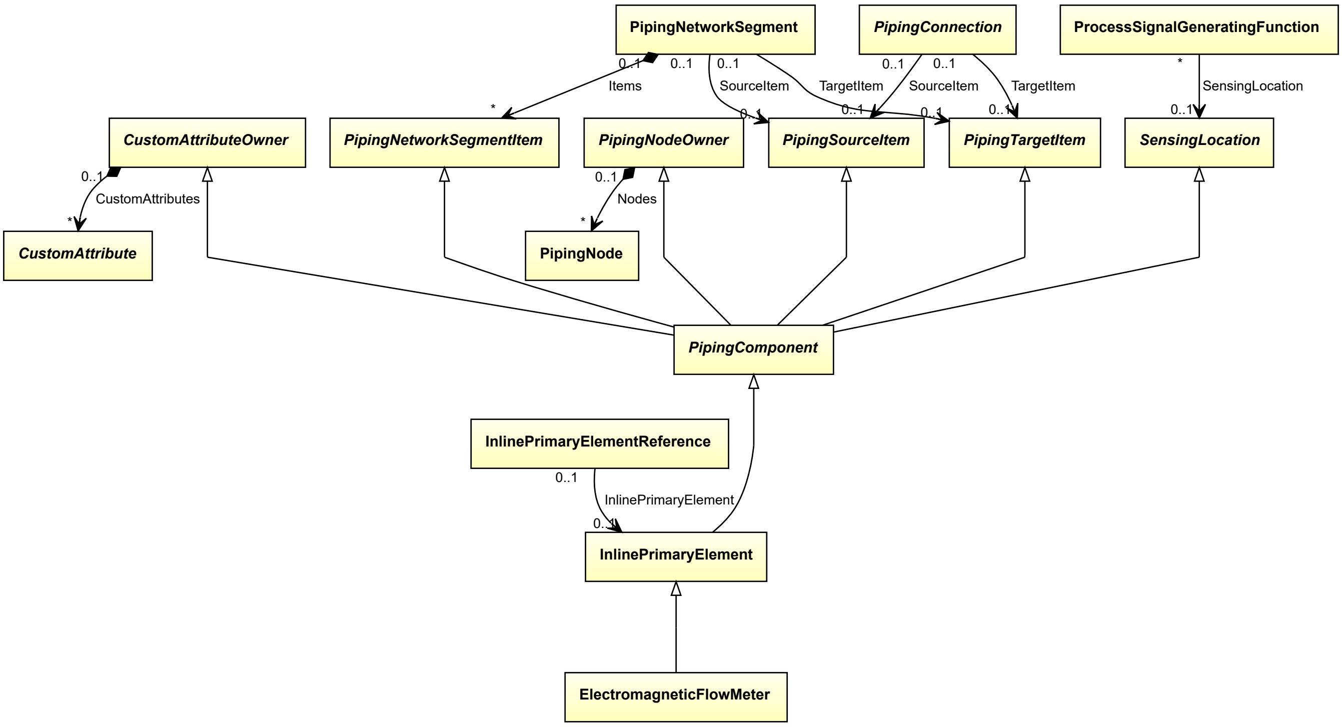
Example

8.20. ElectromagneticFlowMeter

8.20.1 Overview

Class

A velocity flow meter that is measuring flow rate of a conductive fluid running through a magnetic field by measuring the charge created when fluid interacting with the field (from http://data.posccaesar.org/rdl/RDS1009664).



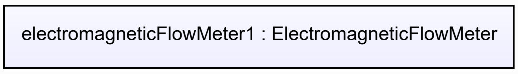
Supertypes

- *InlinePrimaryElement*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingComponent>`
RDL reference: ELECTROMAGNETIC FLOW METER
ComponentClass: ElectromagneticFlowMeter
ComponentClassURI: http://data.posccaesar.org/rdl/RDS1009664

Example



Example: Implementation in Proteus Schema

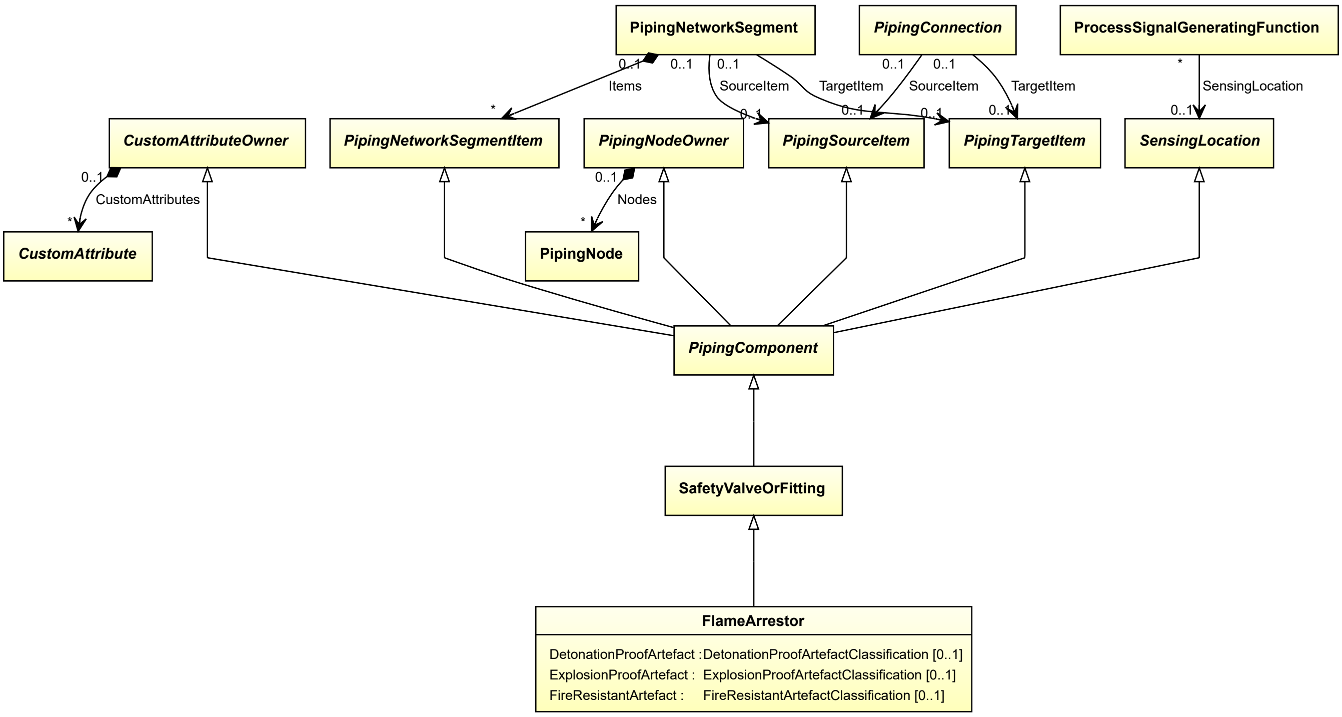
```
<PipingComponent
    ID="electromagneticFlowMeter1"
    ComponentClass="ElectromagneticFlowMeter"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS1009664" ...>
  ...
</PipingComponent>
```

8.21. FlameArrestor

8.21.1 Overview

Class

An ‘arrestor’ which is a trap covering an opening, e.g of a ventilation system or a pipe, to prevent flames from entering the system (from http://data.posccaesar.org/rdl/RDS1325028651).



Supertypes

- *SafetyValveOrFitting*

Attributes (data)

Name	Multiplicity	Type
DetonationProofArtefact	0..1	*DetonationProofArtefactClassification*
ExplosionProofArtefact	0..1	*ExplosionProofArtefactClassification*
FireResistantArtefact	0..1	*FireResistantArtefactClassification*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <PipingComponent>
RDL reference: FLAME ARRESTOR
ComponentClass: FlameArrestor
ComponentClassURI: http://data.posccaesar.org/rdl/RDS1325028651

Example

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="flameArrestor1"
    ComponentClass="FlameArrestor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS1325028651" ...>
  ...
</PipingComponent>
```

8.21.2 DetonationProofArtefact

Attribute (data)

A specialization indicating if the *FlameArrestor* is detonation-proof.

Multiplicity: 0..1

Type: *DetonationProofArtefactClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: DETONATION PROOF ARTEFACT SPECIALIZATION
Name: DetonationProofArtefactSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/DetonationProofArtefactSpecialization

Example

non detonation-proof artefact (*DetonationProofArtefactClassification::NonDetonationProofArtefact*)

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="flameArrestor1"
    ComponentClass="FlameArrestor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS1325028651" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DetonationProofArtefactSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/DetonationProofArtefactSpecialization"
      Format="anyURI"
      Value="NonDetonationProofArtefact"
      ValueURI="http://sandbox.dexpi.org/rdl/NonDetonationProofArtefact" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.21.3 ExplosionProofArtefact

Attribute (data)

A specialization indicating if the *FlameArrestor* is explosion-proof.

Multiplicity: 0..1

Type: *ExplosionProofArtefactClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: EXPLOSION PROOF ARTEFACT SPECIALIZATION
Name: ExplosionProofArtefactSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/ExplosionProofArtefactSpecialization

Example

explosion-proof artefact (*ExplosionProofArtefactClassification::ExplosionProofArtefact*)

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="flameArrestor1"
    ComponentClass="FlameArrestor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS1325028651" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ExplosionProofArtefactSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/ExplosionProofArtefactSpecialization"
      Format="anyURI"
      Value="ExplosionProofArtefact"
      ValueURI="http://sandbox.dexpi.org/rdl/ExplosionProofArtefact" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.21.4 FireResistantArtefact

Attribute (data)

A specialization indicating if the *FlameArrestor* is fire-resistant.

Multiplicity: 0..1

Type: *FireResistantArtefactClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: FIRE RESISTANT ARTEFACT SPECIALIZATION
Name: FireResistantArtefactSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/FireResistantArtefactSpecialization

Example

fire-resistant artefact (*FireResistantArtefactClassification::FireResistantArtefact*)

Example: Implementation in Proteus Schema

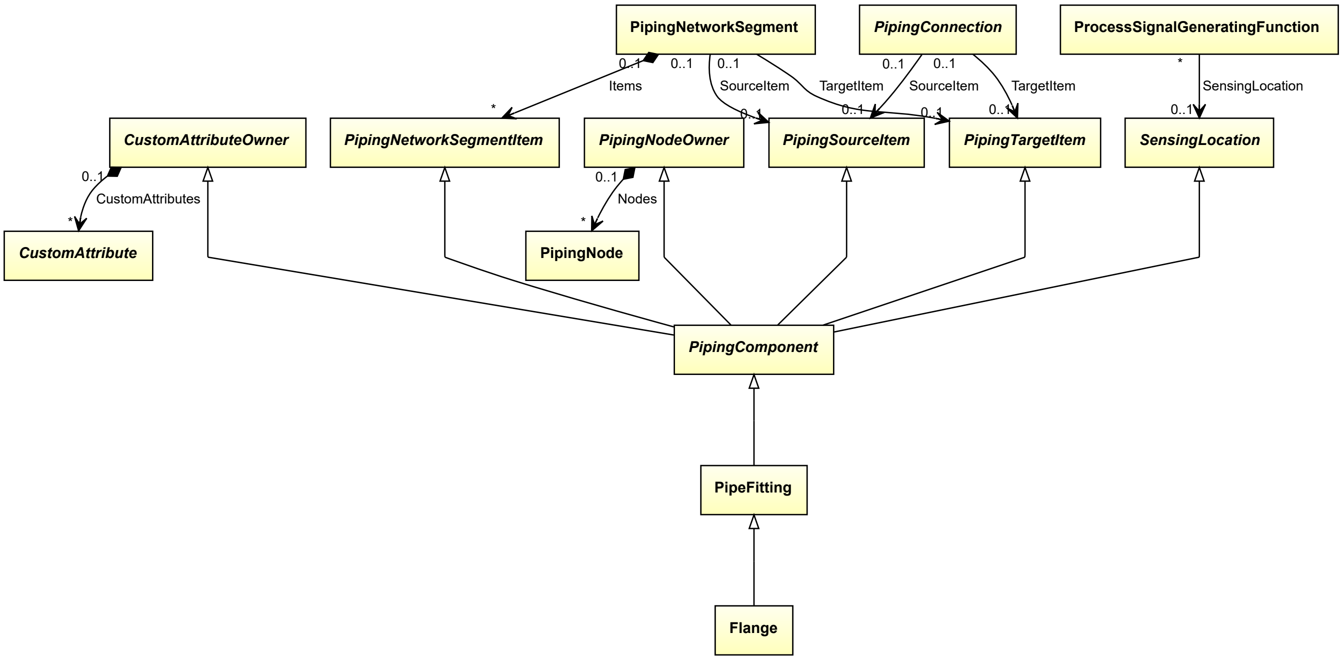
```
<PipingComponent
    ID="flameArrestor1"
    ComponentClass="FlameArrestor"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS1325028651" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="FireResistantArtefactSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/FireResistantArtefactSpecialization"
      Format="anyURI"
      Value="FireResistantArtefact"
      ValueURI="http://data.posccaesar.org/rdl/RDS7907520" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.22. Flange

8.22.1 Overview

Class

A physical object that is a projecting flat rim, plate,collar, or rib (from http://data.posccaesar.org/rdl/RDS13307654).



Supertypes

- *PipeFitting*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingComponent>`
RDL reference: FLANGE
ComponentClass: Flange
ComponentClassURI: http://data.posccaesar.org/rdl/RDS13307654

Example

Example: Implementation in Proteus Schema

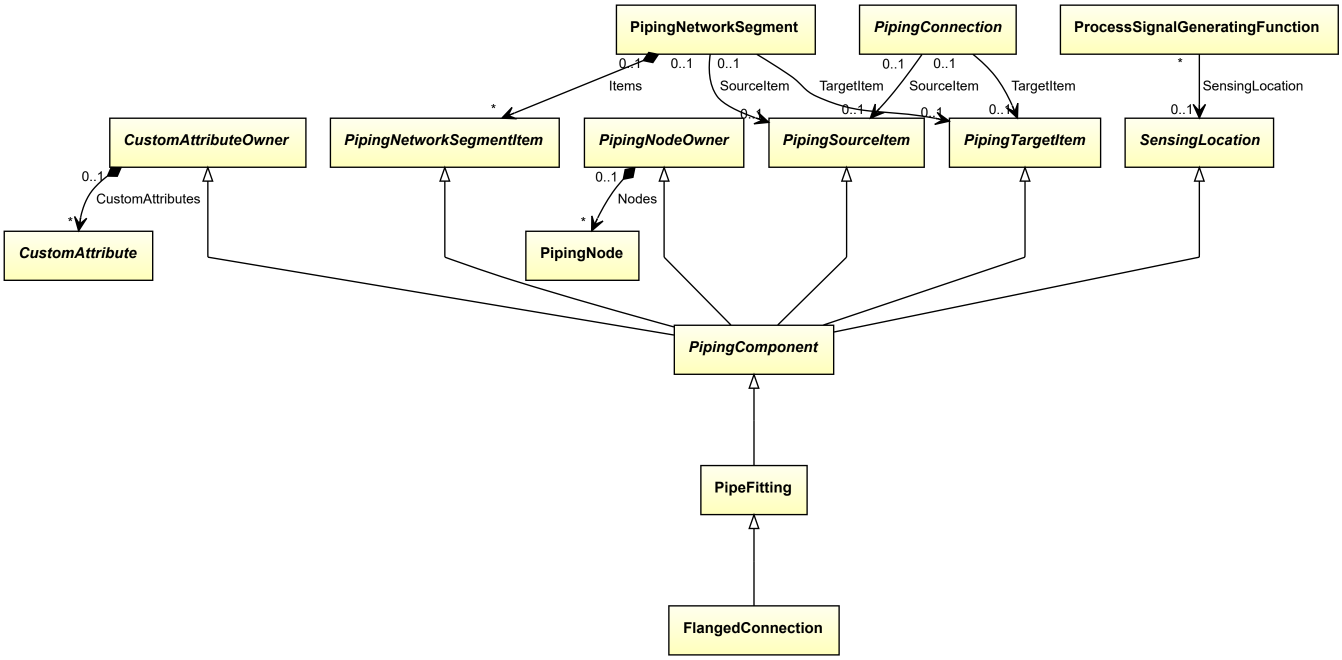
```
<PipingComponent
    ID="flange1"
    ComponentClass="Flange"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS13307654" ...>
  ...
</PipingComponent>
```

8.23. FlangedConnection

8.23.1 Overview

Class

A flanged connection.



Supertypes

- *PipeFitting*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <PipingComponent>
RDL reference: FLANGED CONNECTION
ComponentClass: FlangedConnection
ComponentClassURI: http://sandbox.dexpi.org/rdl/FlangedConnection

Example

flangedConnection1 : FlangedConnection

Example: Implementation in Proteus Schema

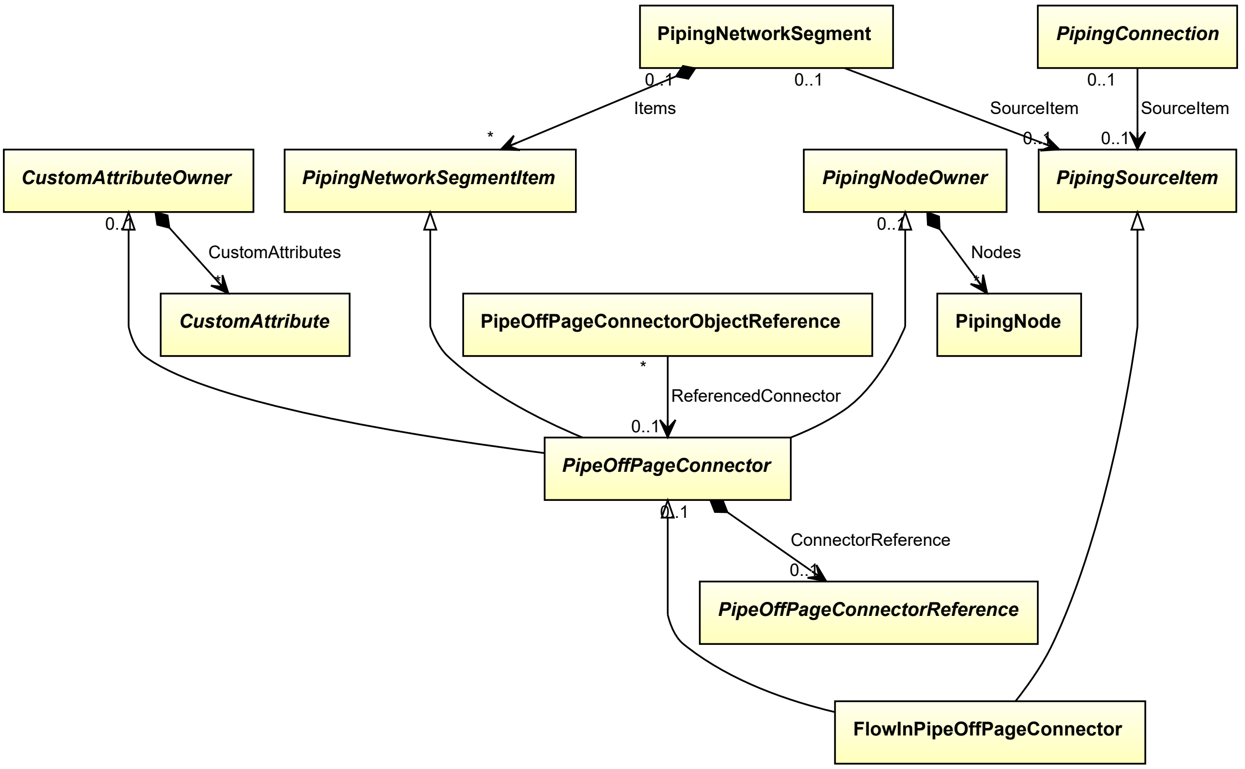
```
<PipingComponent
    ID="flangedConnection1"
    ComponentClass="FlangedConnection"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/FlangedConnection" ...>
  ...
</PipingComponent>
```

8.24. FlowInPipeOffPageConnector

8.24.1 Overview

Class

A pipe connector that indicates that a preceding part of a piping network segment is represented somewhere else, either on the same PID, or on some other PID.



Supertypes

- *PipeOffPageConnector*
- *PipingSourceItem*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <PipeOffPageConnector>
RDL reference: FLOW IN PIPE OFF PAGE CONNECTOR
ComponentClass: FlowInPipeOffPageConnector
ComponentClassURI: http://sandbox.dexpi.org/rdl/FlowInPipeOffPageConnector

Example

Example: Implementation in Proteus Schema

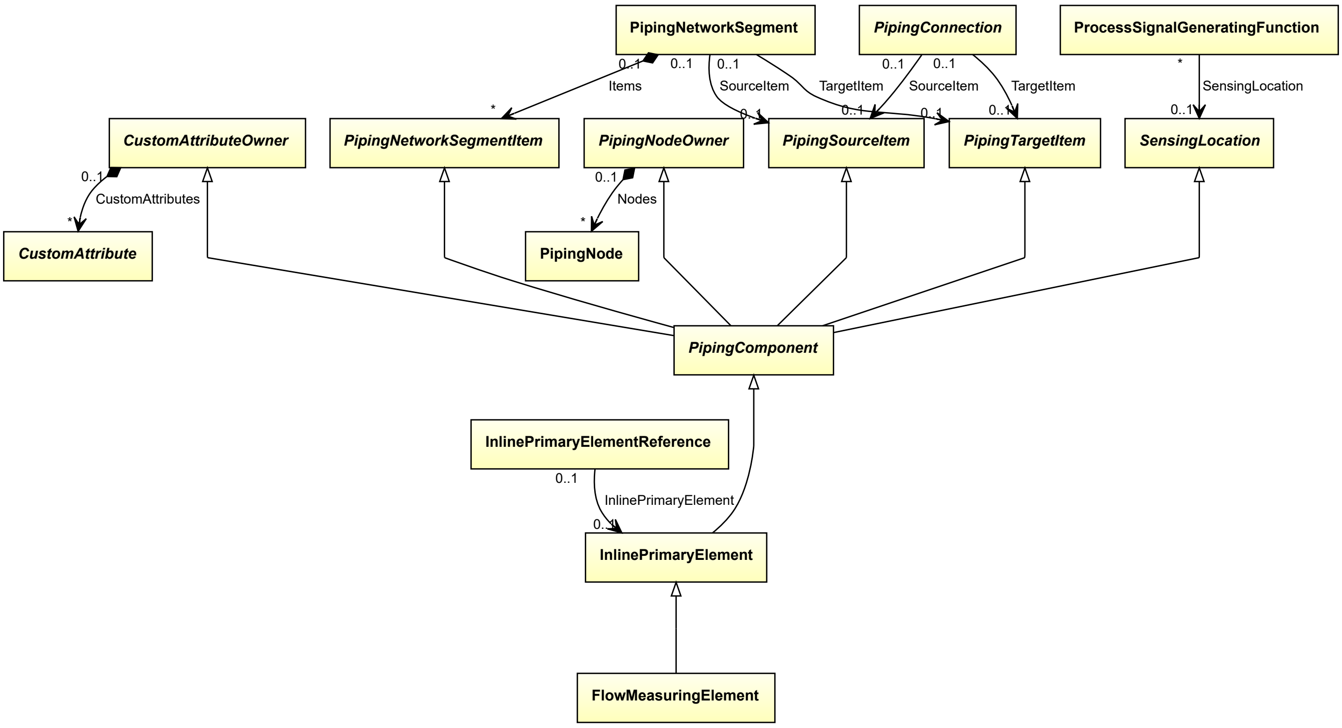
```
<PipeOffPageConnector
    ID="flowInPipeOffPageConnector1"
    ComponentClass="FlowInPipeOffPageConnector"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/FlowInPipeOffPageConnector" ...>
  ...
</PipeOffPageConnector>
```

8.25. FlowMeasuringElement

8.25.1 Overview

Class

A FLOW MEASURING ELEMENT is a MEASURING ELEMENT that is used to measure FLOW RATE.



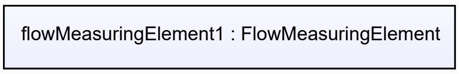
Supertypes

- *InlinePrimaryElement*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingComponent>`
RDL reference: FLOW MEASURING ELEMENT
ComponentClass: FlowMeasuringElement
ComponentClassURI: http://sandbox.dexpi.org/rdl/FlowMeasuringElement

Example



Example: Implementation in Proteus Schema

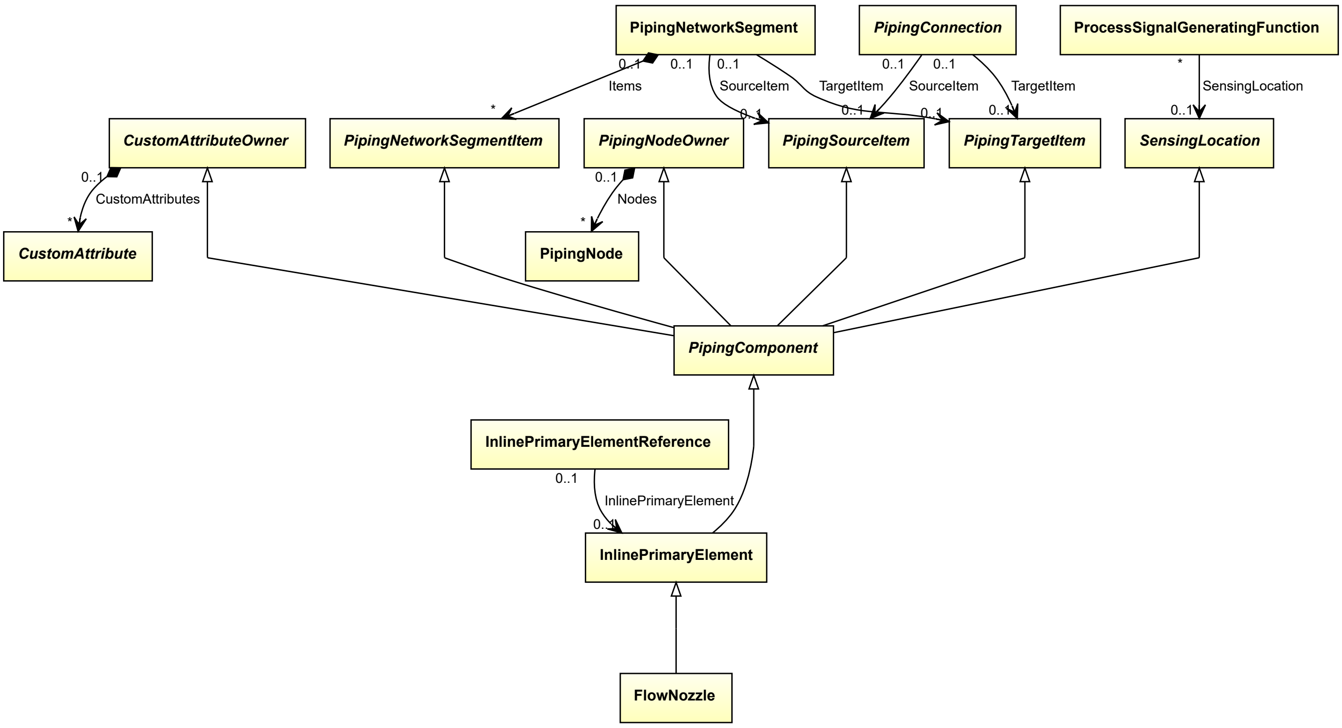
```
<PipingComponent
    ID="flowMeasuringElement1"
    ComponentClass="FlowMeasuringElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/FlowMeasuringElement" ...>
  ...
</PipingComponent>
```

8.26. FlowNozzle

8.26.1 Overview

Class

A nozzle with a smooth entry and a sharp exit (from http://data.posccaesar.org/rdl/RDS821024).



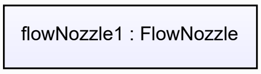
Supertypes

- *InlinePrimaryElement*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <PipingComponent>
RDL reference: FLOW NOZZLE
ComponentClass: FlowNozzle
ComponentClassURI: http://data.posccaesar.org/rdl/RDS821024

Example



Example: Implementation in Proteus Schema

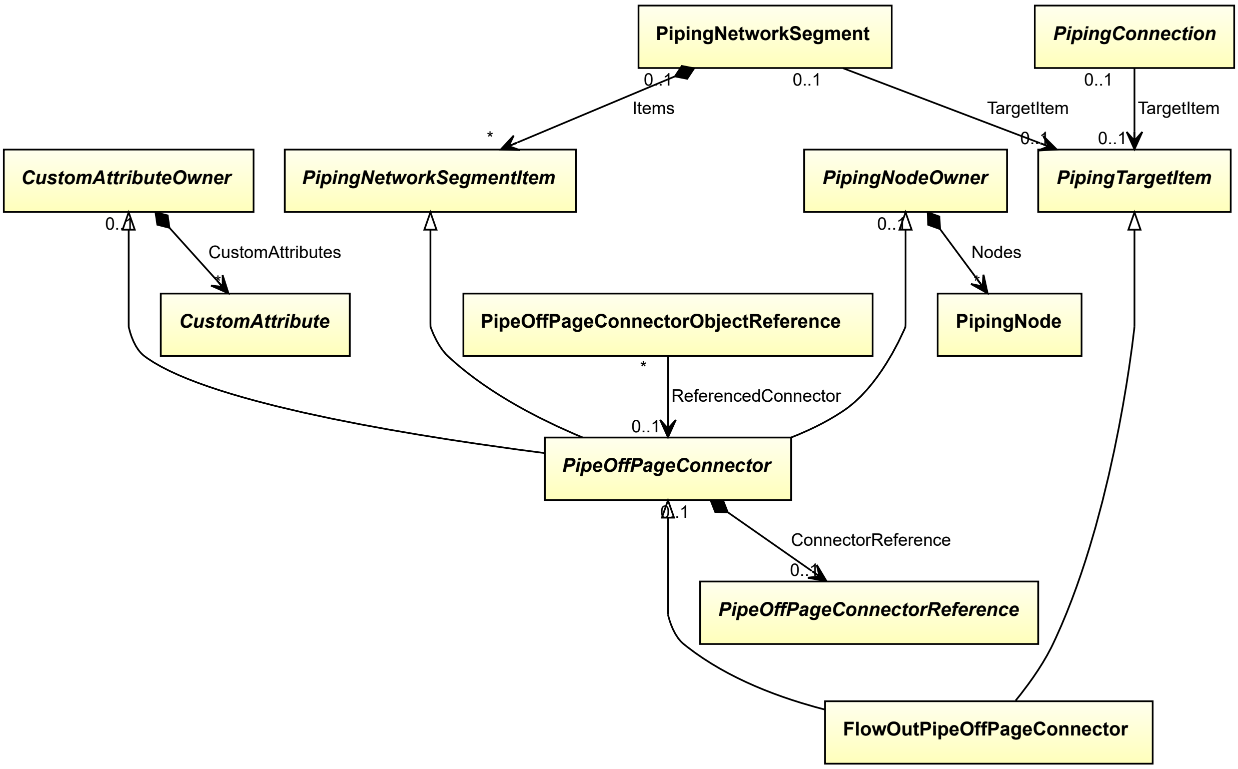
```
<PipingComponent
    ID="flowNozzle1"
    ComponentClass="FlowNozzle"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS821024" ...>
  ...
</PipingComponent>
```

8.27. FlowOutPipeOffPageConnector

8.27.1 Overview

Class

A pipe connector that indicates that a subsequent part of a piping network segment is represented somewhere else, either on the same PID, or on some other PID.



Supertypes

- *PipeOffPageConnector*
- *PipingTargetItem*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <PipeOffPageConnector>
RDL reference: FLOW OUT PIPE OFF PAGE CONNECTOR
ComponentClass: FlowOutPipeOffPageConnector
ComponentClassURI: http://sandbox.dexpi.org/rdl/FlowOutPipeOffPageConnector

Example

Example: Implementation in Proteus Schema

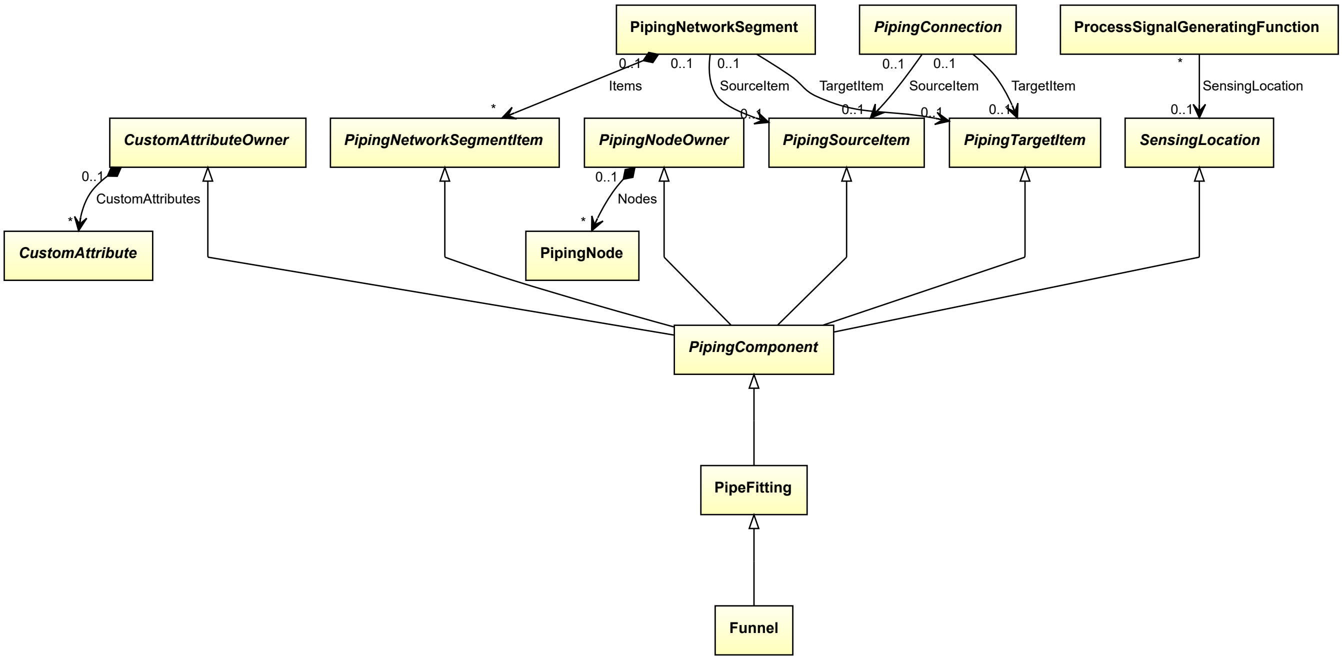
```
<PipeOffPageConnector
    ID="flowOutPipeOffPageConnector1"
    ComponentClass="FlowOutPipeOffPageConnector"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/FlowOutPipeOffPageConnector" ...>
  ...
</PipeOffPageConnector>
```

8.28. Funnel

8.28.1 Overview

Class

A hollow cone with a tube extending from the smaller end and that is designed to catch and direct a downward flow (from http://data.posccaesar.org/rdl/RDS6689917).



Supertypes

- *PipeFitting*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingComponent>`
RDL reference: FUNNEL
ComponentClass: Funnel
ComponentClassURI: http://data.posccaesar.org/rdl/RDS6689917

Example

Example: Implementation in Proteus Schema

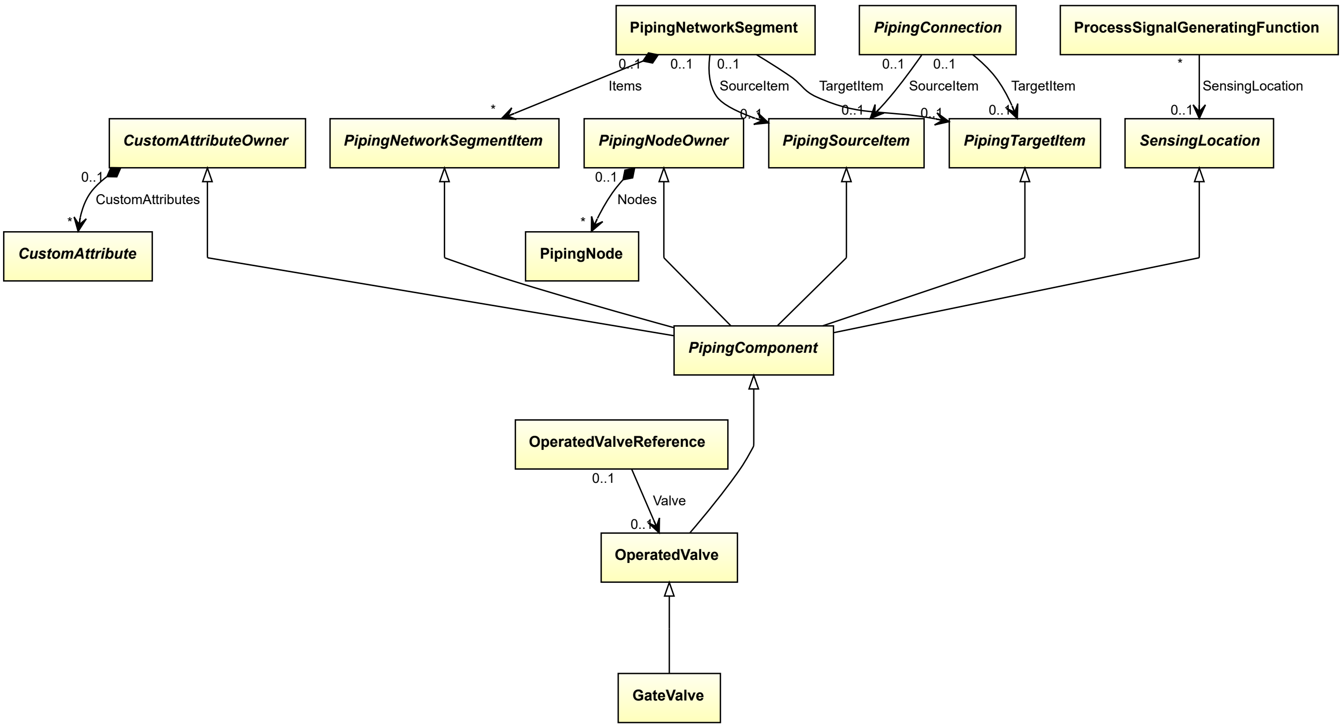
```
<PipingComponent
    ID="funnel1"
    ComponentClass="Funnel"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS6689917" ...>
  ...
</PipingComponent>
```

8.29. GateValve

8.29.1 Overview

Class

A valve that is a valve where the closure member is a gate or disc with a linear motion parallel, or nearly parallel, to the plane of flat seats, which are transverse to the direction of flow (from http://data.posccaesar.org/rdl/RDS416519).



Supertypes

- *OperatedValve*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingComponent>`
RDL reference: GATE VALVE
ComponentClass: GateValve
ComponentClassURI: http://data.posccaesar.org/rdl/RDS416519

Example

gateValve1 : GateValve

Example: Implementation in Proteus Schema

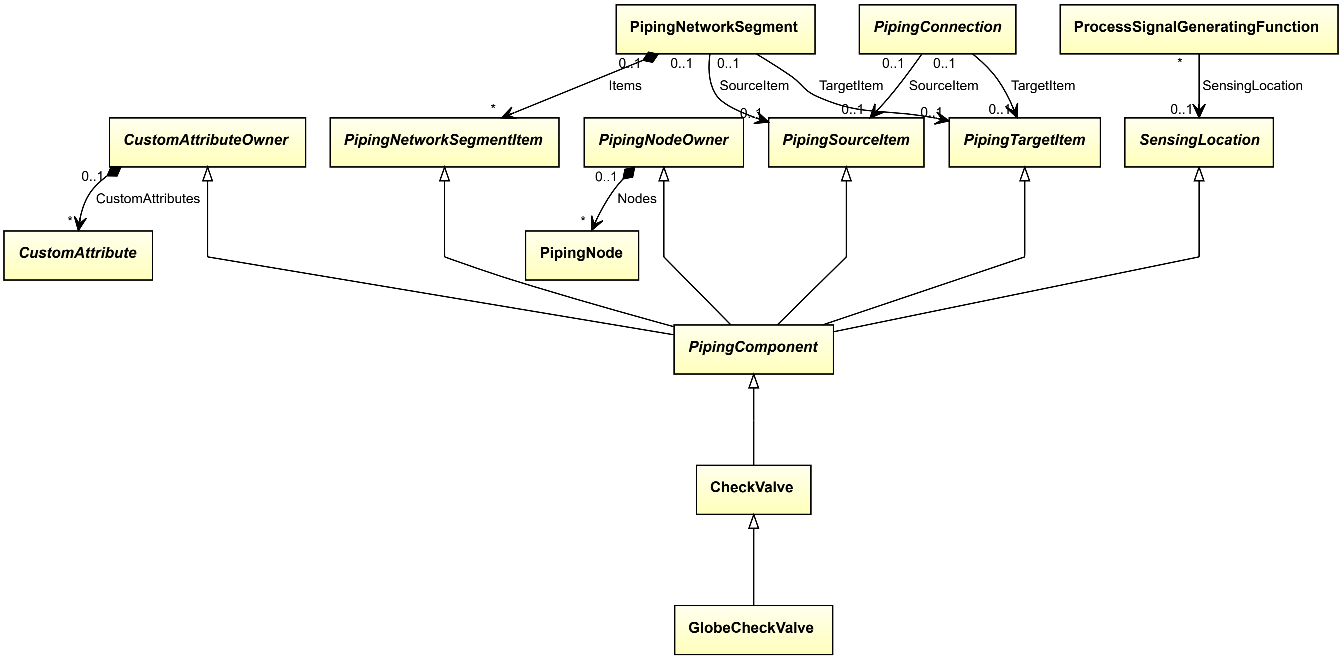
```
<PipingComponent
    ID="gateValve1"
    ComponentClass="GateValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS416519" ...>
  ...
</PipingComponent>
```

8.30. GlobeCheckValve

8.30.1 Overview

Class

A globe chack valve.



Supertypes

- *CheckValve*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <PipingComponent>
RDL reference: GLOBE CHECK VALVE
ComponentClass: GlobeCheckValve
ComponentClassURI: http://sandbox.dexpi.org/rdl/GlobeCheckValve

Example

Example: Implementation in Proteus Schema

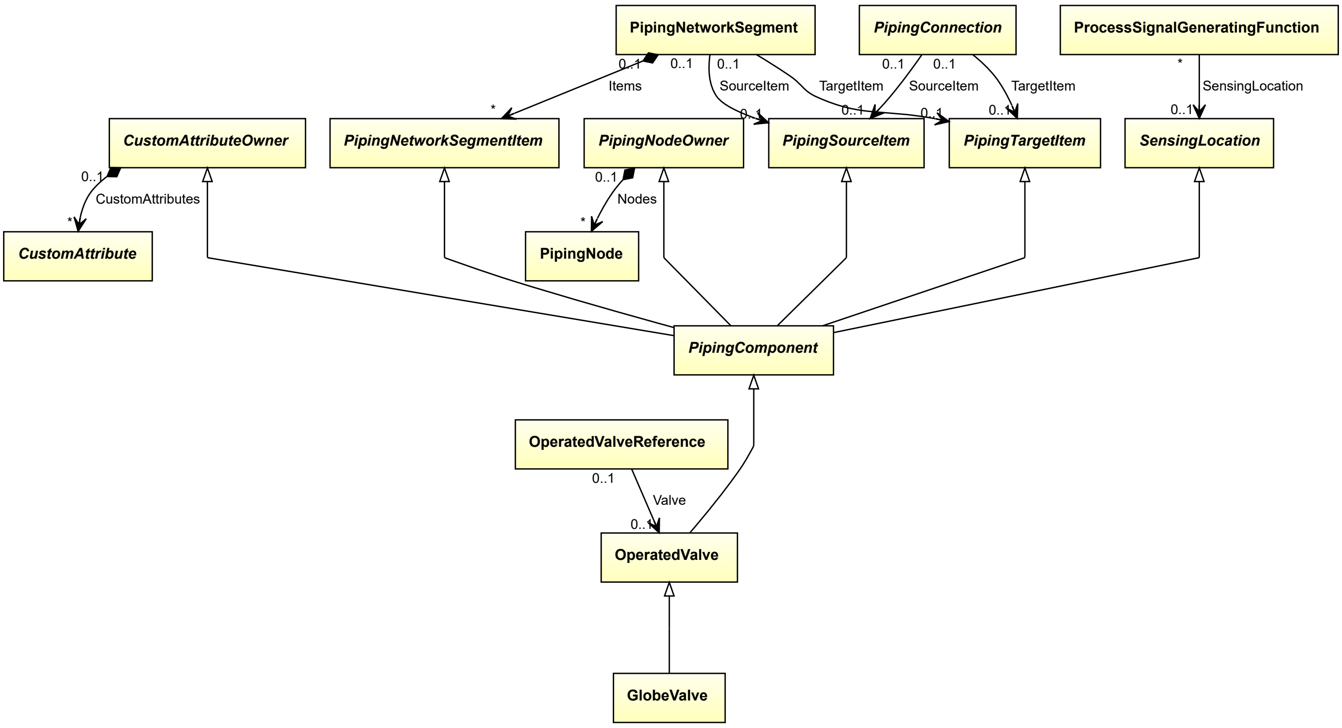
```
<PipingComponent
    ID="globeCheckValve1"
    ComponentClass="GlobeCheckValve"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/GlobeCheckValve" ...>
  ...
</PipingComponent>
```

8.31. GlobeValve

8.31.1 Overview

Class

A valve that is a valve where the closure member is a disc or piston operating with linear motion normal to the flat or shaped seat (from http://data.posccaesar.org/rdl/RDS416204).



Supertypes

- *OperatedValve*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingComponent>`
RDL reference: GLOBE VALVE
ComponentClass: GlobeValve
ComponentClassURI: http://data.posccaesar.org/rdl/RDS416204

Example

globeValve1 : GlobeValve

Example: Implementation in Proteus Schema

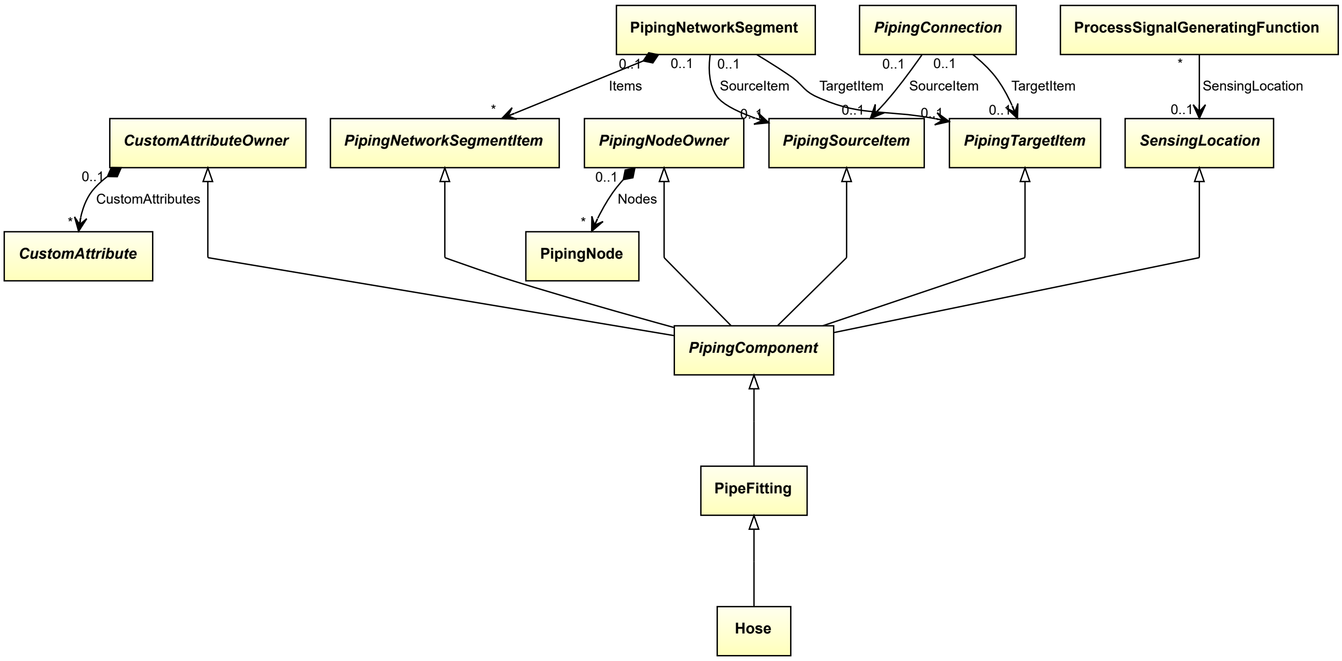
```
<PipingComponent
    ID="globeValve1"
    ComponentClass="GlobeValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS416204" ...>
  ...
</PipingComponent>
```

8.32. Hose

8.32.1 Overview

Class

A tubular which is flexible and capable of conveying liquids under pressure (from http://data.posccaesar.org/rdl/RDS302174).



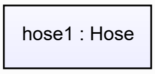
Supertypes

- *PipeFitting*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <PipingComponent>
RDL reference: HOSE
ComponentClass: Hose
ComponentClassURI: http://data.posccaesar.org/rdl/RDS302174

Example



Example: Implementation in Proteus Schema

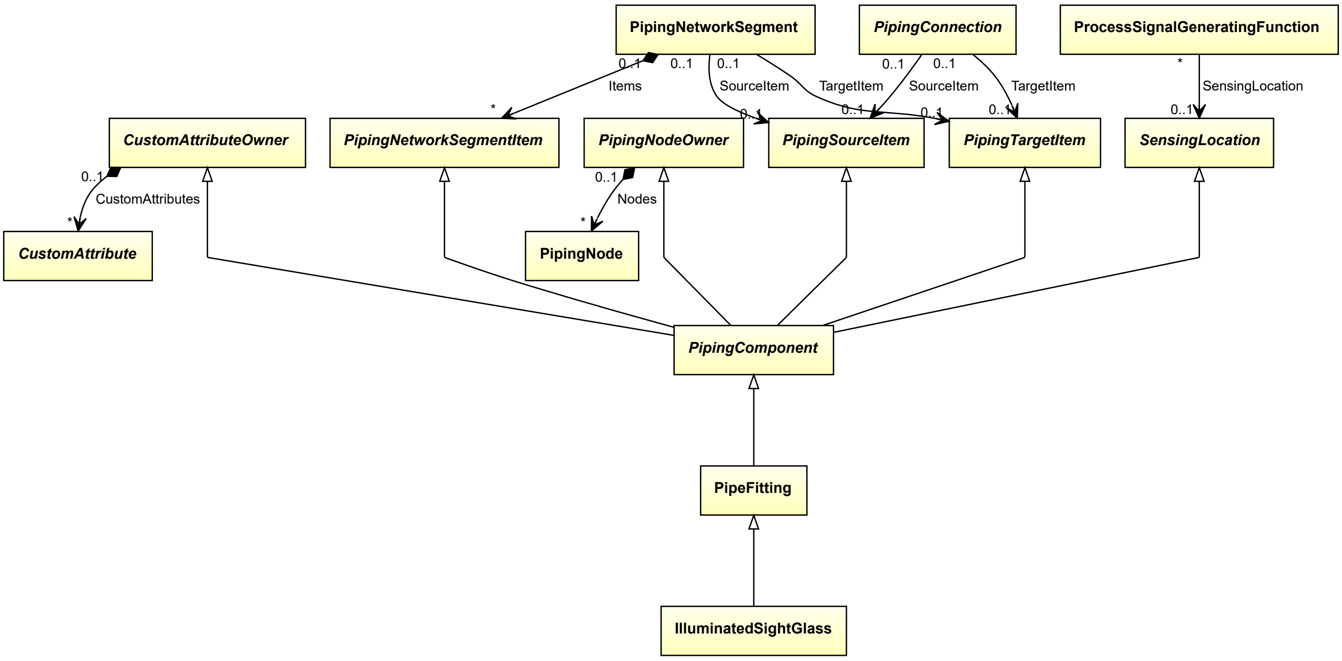
```
<PipingComponent
    ID="hose1"
    ComponentClass="Hose"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS302174" ...>
  ...
</PipingComponent>
```

8.33. IlluminatedSightGlass

8.33.1 Overview

Class

An illuminated sight glass.



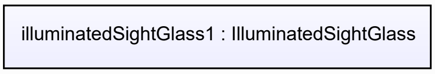
Supertypes

- *PipeFitting*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingComponent>`
RDL reference: ILLUMINATED SIGHT GLASS
ComponentClass: IlluminatedSightGlass
ComponentClassURI: http://sandbox.dexpi.org/rdl/IlluminatedSightGlass

Example



Example: Implementation in Proteus Schema

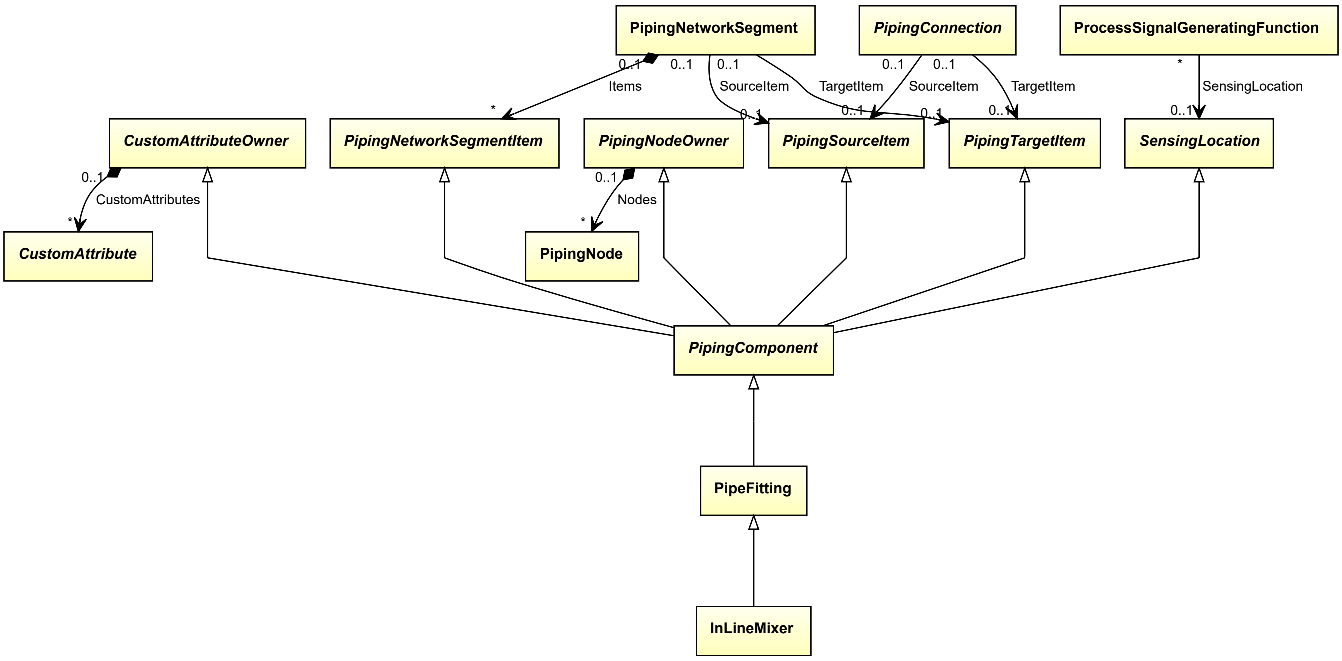
```
<PipingComponent
    ID="illuminatedSightGlass1"
    ComponentClass="IlluminatedSightGlass"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/IlluminatedSightGlass" ...>
  ...
</PipingComponent>
```

8.34. InLineMixer

8.34.1 Overview

Class

A static mixer that is intended to be supported by connected equipment. Typically supported by piping (from http://data.posccaesar.org/rdl/RDS43167562195).



Supertypes

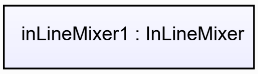
- *PipeFitting*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <PipingComponent>
RDL reference: IN-LINE MIXER
ComponentClass: InLineMixer

ComponentClassURI: http://data.posccaesar.org/rdl/RDS43167562195

Example



Example: Implementation in Proteus Schema

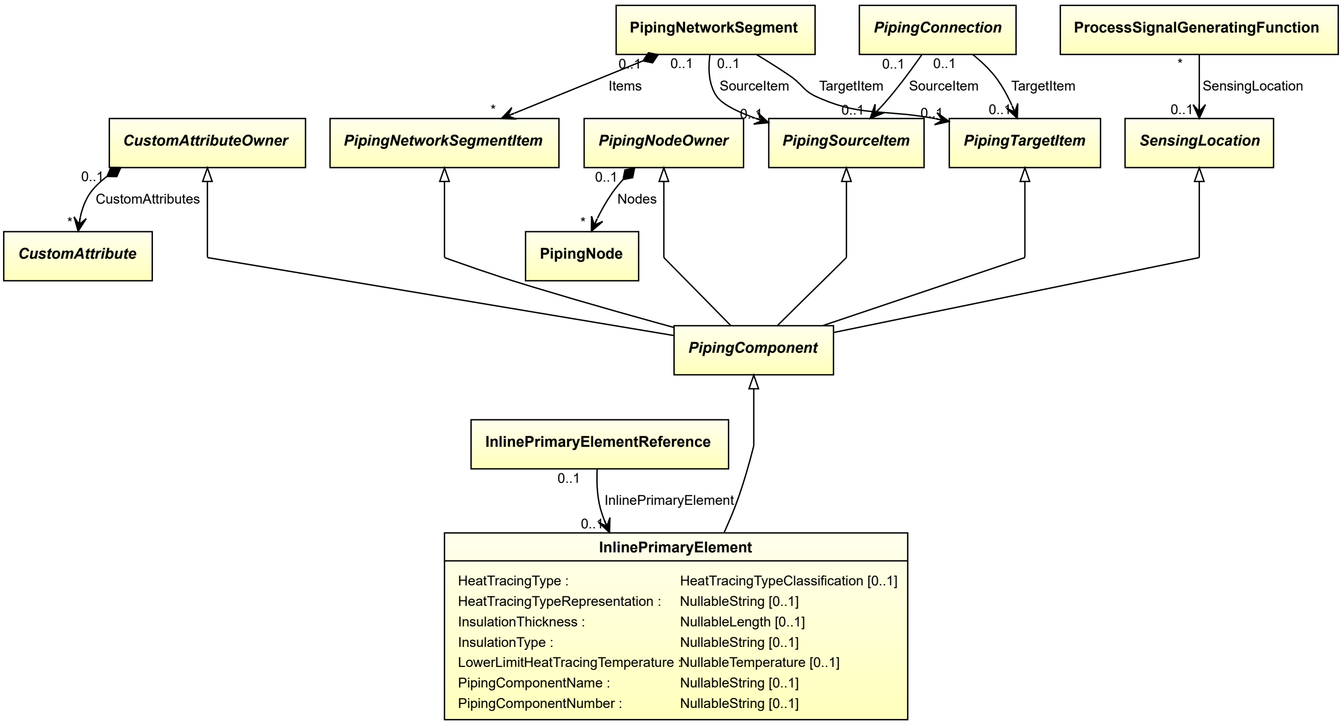
```
<PipingComponent
    ID="inLineMixer1"
    ComponentClass="InLineMixer"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS43167562195" ...>
  ...
</PipingComponent>
```

8.35. InlinePrimaryElement

8.35.1 Overview

Class

An inline primary element.


PipingNetworkSegment
PipingConnection
ProcessSignalGeneratingFunction
0..1
0..1
0..1
Items
SourceItem
TargetItem
0..1
0..1
SourceItem
TargetItem
*
SensingLocation
*
0..1
0..1
0..1
0..1
0..1
CustomAttributeOwner
PipingNetworkSegmentItem
PipingNodeOwner
PipingSourceItem
PipingTargetItem
SensingLocation
0..1
CustomAttributes
*
0..1
Nodes
*
CustomAttribute
PipingNode
PipingComponent
InlinePrimaryElementReference
0..1
InlinePrimaryElement
0..1
InlinePrimaryElement
HeatTracingType : HeatTracingTypeClassification [0..1]
HeatTracingTypeRepresentation : NullableString [0..1]
InsulationThickness : NullableLength [0..1]
InsulationType : NullableString [0..1]
LowerLimitHeatTracingTemperature : NullableTemperature [0..1]
PipingComponentName : NullableString [0..1]
PipingComponentNumber : NullableString [0..1]

Supertypes

- *PipingComponent*

Subtypes

- *CustomInlinePrimaryElement*
- *ElectromagneticFlowMeter*
- *FlowMeasuringElement*
- *FlowNozzle*
- *MassFlowMeasuringElement*
- *PositiveDisplacementFlowMeter*
- *TurbineFlowMeter*
- *VariableAreaFlowMeter*
- *VenturiTube*
- *VolumeFlowMeasuringElement*

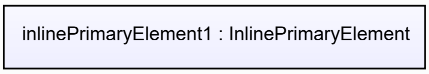
Attributes (data)

Name	Multiplicity	Type
HeatTracingType	0..1	*HeatTracingTypeClassification*
HeatTracingTypeRepresentation	0..1	*NullableString*
InsulationThickness	0..1	*NullableLength*
InsulationType	0..1	*NullableString*
LowerLimitHeatTracingTemperature	0..1	*NullableTemperature*
PipingComponentName	0..1	*NullableString*
PipingComponentNumber	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingComponent>`
RDL reference: INLINE PRIMARY ELEMENT
ComponentClass: InlinePrimaryElement
ComponentClassURI: http://sandbox.dexpi.org/rdl/InlinePrimaryElement

Example



Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="inlinePrimaryElement1"
    ComponentClass="InlinePrimaryElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/InlinePrimaryElement" ...>
  ...
</PipingComponent>
```

8.35.2 HeatTracingType

Attribute (data)

A specialization indicating the heat tracing type related to the *InlinePrimaryElement*.

Multiplicity: 0..1

Type: *HeatTracingTypeClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: HEAT TRACING TYPE SPECIALIZATION
Name: HeatTracingTypeSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/HeatTracingTypeSpecialization

Example

electrical heat tracing system (*HeatTracingTypeClassification::ElectricalHeatTracingSystem*)

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="inlinePrimaryElement1"
    ComponentClass="InlinePrimaryElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/InlinePrimaryElement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="HeatTracingTypeSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/HeatTracingTypeSpecialization"
      Format="anyURI"
      Value="ElectricalHeatTracingSystem"
      ValueURI="http://data.posccaesar.org/rdl/RDS11854600" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.35.3 HeatTracingTypeRepresentation

Attribute (data)

The heat tracing type related to the *InlinePrimaryElement*, represented as a string.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: HEAT TRACING TYPE REPRESENTATION ASSIGNMENT CLASS
Name: HeatTracingTypeRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/HeatTracingTypeRepresentationAssignmentClass

Example

“E” (*String*)

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="inlinePrimaryElement1"
    ComponentClass="InlinePrimaryElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/InlinePrimaryElement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="HeatTracingTypeRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/HeatTracingTypeRepresentationAssignmentClass"
      Format="string"
      Value="E" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.35.4 InsulationThickness

Attribute (data)

The insulation thickness of the *InlinePrimaryElement*.

Multiplicity: 0..1

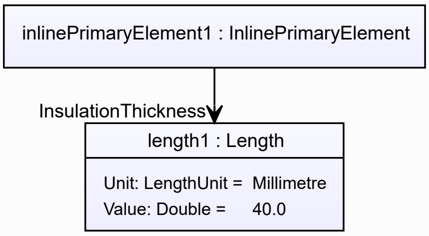
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: INSULATION THICKNESS
Name: InsulationThickness
AttributeURI: http://data.posccaesar.org/rdl/RDS4238040

Example

The instance inlinePrimaryElement1 represents an *InlinePrimaryElement* with an *InsulationThickness* of 40.0 mm.


inlinePrimaryElement1 : InlinePrimaryElement
InsulationThickness
length1 : Length
Unit: LengthUnit = Millimetre
Value: Double = 40.0

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="inlinePrimaryElement1"
    ComponentClass="InlinePrimaryElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/InlinePrimaryElement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="InsulationThickness"
      AttributeURI="http://data.posccaesar.org/rdl/RDS4238040"
      Format="double"
      Value="40.0"
      Units="Millimetre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1357739" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.35.5 InsulationType

Attribute (data)

The identification code for the insulation type related to the *InlinePrimaryElement*. So far, DEXPI does not define restrictions for valid values.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: INSULATION TYPE ASSIGNMENT CLASS
Name: InsulationTypeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/InsulationTypeAssignmentClass

Example

“Q” (*String*)

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="inlinePrimaryElement1"
    ComponentClass="InlinePrimaryElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/InlinePrimaryElement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="InsulationTypeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/InsulationTypeAssignmentClass"
      Format="string"
      Value="Q" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.35.6 LowerLimitHeatTracingTemperature

Attribute (data)

The lower limit for the temperature that a heat tracing system must ensure for the *InlinePrimaryElement*.

Multiplicity: 0..1

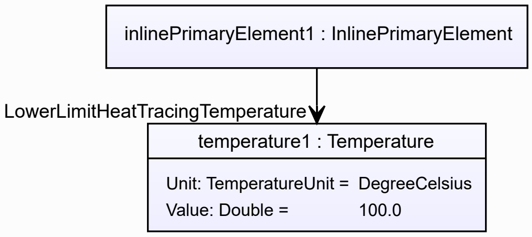
Type: *NullableTemperature*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: LOWER LIMIT HEAT TRACING TEMPERATURE
Name: LowerLimitHeatTracingTemperature
AttributeURI: http://sandbox.dexpi.org/rdl/LowerLimitHeatTracingTemperature

Example

The instance inlinePrimaryElement1 represents an *InlinePrimaryElement* with a *LowerLimitHeatTracingTemperature* of 100.0 °C.


inlinePrimaryElement1 : InlinePrimaryElement
LowerLimitHeatTracingTemperature
temperature1 : Temperature
Unit: TemperatureUnit = DegreeCelsius
Value: Double = 100.0

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="inlinePrimaryElement1"
    ComponentClass="InlinePrimaryElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/InlinePrimaryElement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="LowerLimitHeatTracingTemperature"
      AttributeURI="http://sandbox.dexpi.org/rdl/LowerLimitHeatTracingTemperature"
      Format="double"
      Value="100.0"
      Units="DegreeCelsius"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1322684" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.35.7 PipingComponentName

Attribute (data)

A string to classify the *InlinePrimaryElement*. DEXPI does not prescribe the classification system. Typically, company or site site standards are used.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PIPING COMPONENT NAME ASSIGNMENT CLASS
Name: PipingComponentNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PipingComponentNameAssignmentClass

Example

“73KH12” (*String*)

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="inlinePrimaryElement1"
    ComponentClass="InlinePrimaryElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/InlinePrimaryElement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PipingComponentNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PipingComponentNameAssignmentClass"
      Format="string"
      Value="73KH12" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.35.8 PipingComponentNumber

Attribute (data)

An identifier of the *InlinePrimaryElement*. DEXPI does not prescribe the scope of the identifier, i.e., whether it should be unique in, e.g., a *PipingNetworkSegment* or a *PipingNetworkSystem*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PIPING COMPONENT NUMBER ASSIGNMENT CLASS
Name: PipingComponentNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PipingComponentNumberAssignmentClass

Example

“C2” (*String*)

Example: Implementation in Proteus Schema

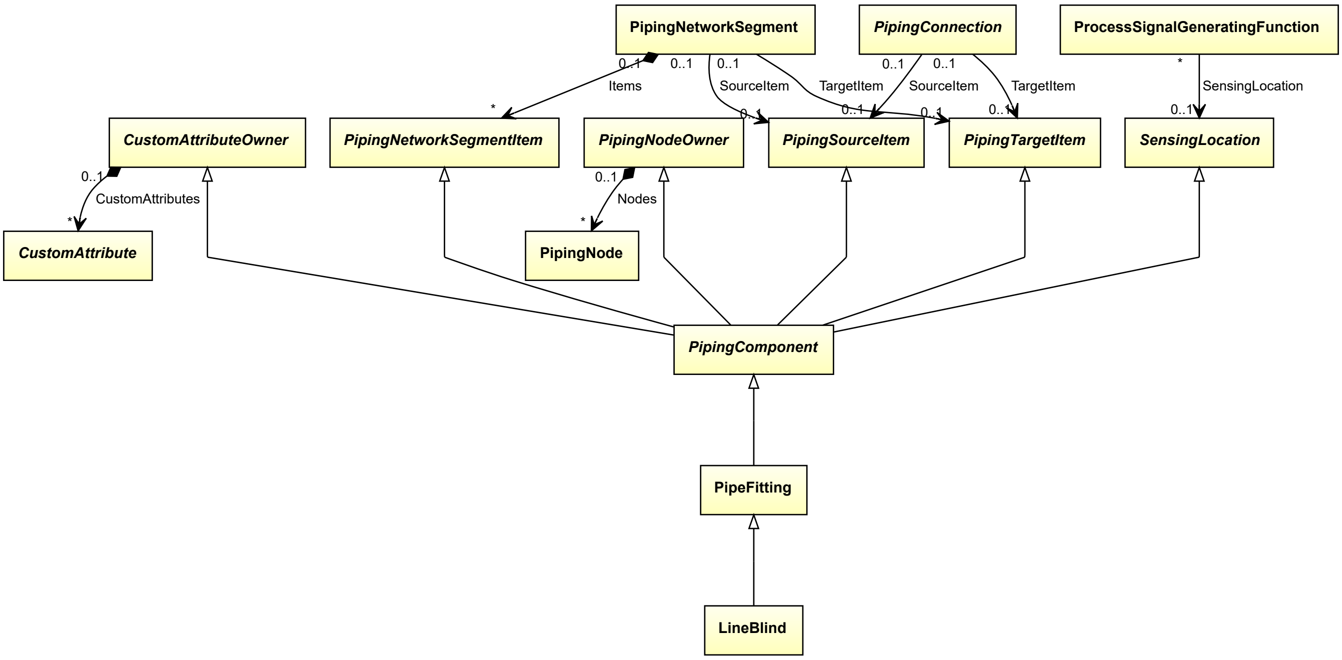
```
<PipingComponent
    ID="inlinePrimaryElement1"
    ComponentClass="InlinePrimaryElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/InlinePrimaryElement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PipingComponentNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PipingComponentNumberAssignmentClass"
      Format="string"
      Value="C2" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.36. LineBlind

8.36.1 Overview

Class

A functional unit used to blind off a process stream (from http://data.posccaesar.org/rdl/RDS280034).



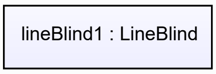
Supertypes

- *PipeFitting*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <PipingComponent>
RDL reference: LINE BLIND
ComponentClass: LineBlind
ComponentClassURI: http://data.posccaesar.org/rdl/RDS280034

Example



Example: Implementation in Proteus Schema

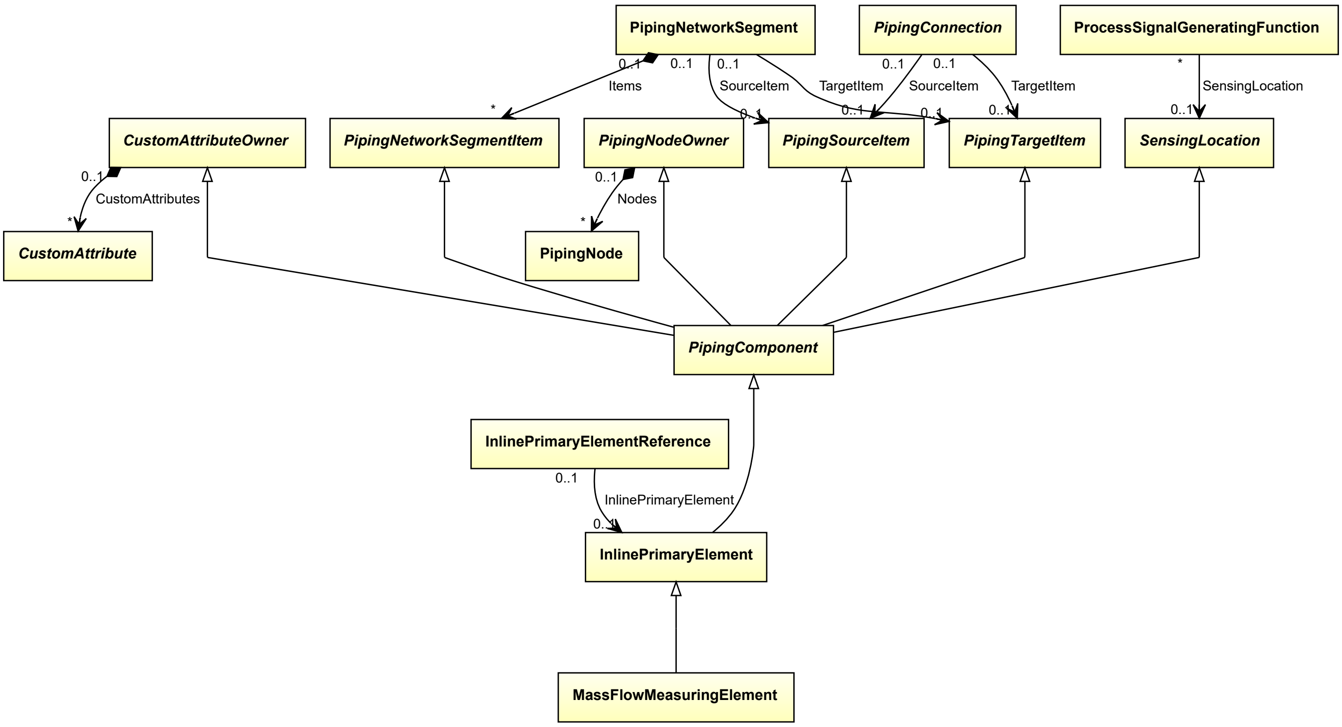
```
<PipingComponent
    ID="lineBlind1"
    ComponentClass="LineBlind"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS280034" ...>
  ...
</PipingComponent>
```

8.37. MassFlowMeasuringElement

8.37.1 Overview

Class

A MASS FLOW MEASURING ELEMENT is a FLOW MEASURING ELEMENT that is used to measure MASS FLOW RATE.



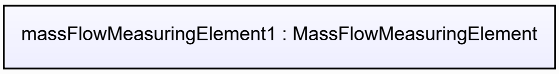
Supertypes

- *InlinePrimaryElement*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <PipingComponent>
RDL reference: MASS FLOW MEASURING ELEMENT
ComponentClass: MassFlowMeasuringElement
ComponentClassURI: http://sandbox.dexpi.org/rdl/MassFlowMeasuringElement

Example



Example: Implementation in Proteus Schema

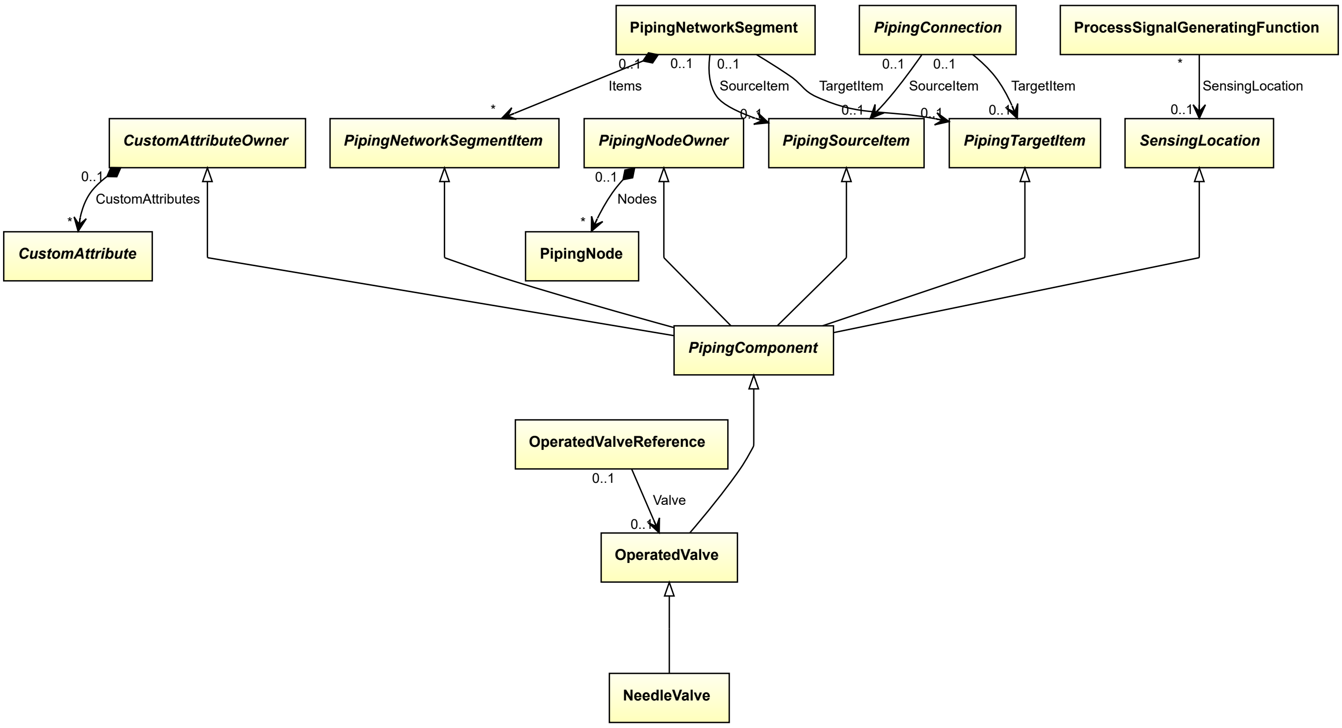
```
<PipingComponent
    ID="massFlowMeasuringElement1"
    ComponentClass="MassFlowMeasuringElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MassFlowMeasuringElement" ...>
  ...
</PipingComponent>
```

8.38. NeedleValve

8.38.1 Overview

Class

A globe valve that has a closure member with the shape of a conical plug (needle) which closes into a small seat (from http://data.posccaesar.org/rdl/RDS421064).



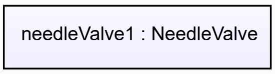
Supertypes

- *OperatedValve*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <PipingComponent>
RDL reference: NEEDLE VALVE
ComponentClass: NeedleValve
ComponentClassURI: http://data.posccaesar.org/rdl/RDS421064

Example



Example: Implementation in Proteus Schema

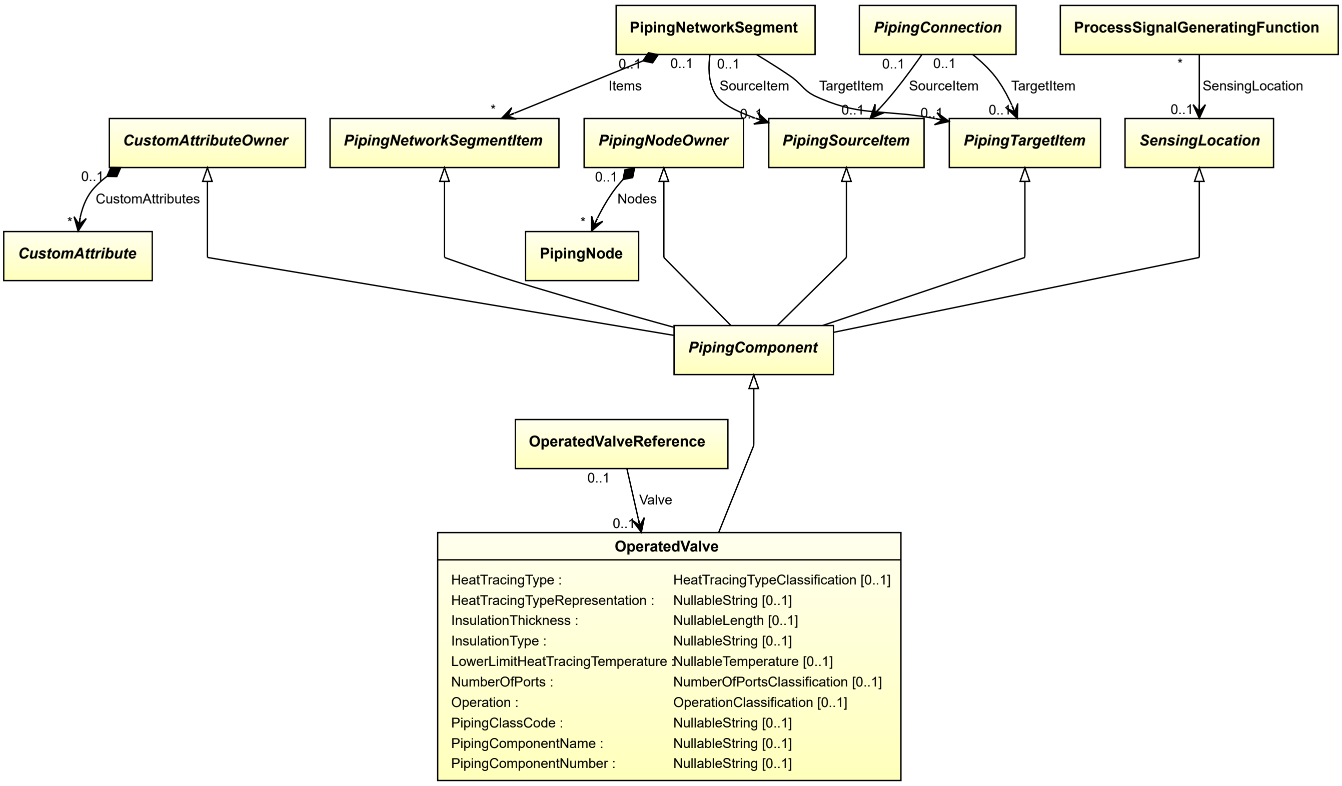
```
<PipingComponent
    ID="needleValve1"
    ComponentClass="NeedleValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS421064" ...>
  ...
</PipingComponent>
```

8.39. OperatedValve

8.39.1 Overview

Class

A valve that includes an external means of operation. (E.g. handwheel / lever / actuator.) (from http://data.posccaesar.org/rdl/RDS11141590).


PipingNetworkSegment
PipingConnection
ProcessSignalGeneratingFunction
0..1
0..1
Items
0..1
SourceItem
TargetItem
0..1
0..1
SourceItem
TargetItem
*
SensingLocation
*
0..1
0..1
0..1
0..1
0..1
CustomAttributeOwner
PipingNetworkSegmentItem
PipingNodeOwner
PipingSourceItem
PipingTargetItem
SensingLocation
0..1
CustomAttributes
0..1
Nodes
*
*
CustomAttribute
PipingNode
PipingComponent
OperatedValveReference
0..1
Valve
0..1
OperatedValve
HeatTracingType : HeatTracingTypeClassification [0..1]
HeatTracingTypeRepresentation : NullableString [0..1]
InsulationThickness : NullableLength [0..1]
InsulationType : NullableString [0..1]
LowerLimitHeatTracingTemperature : NullableTemperature [0..1]
NumberOfPorts : NumberOfPortsClassification [0..1]
Operation : OperationClassification [0..1]
PipingClassCode : NullableString [0..1]
PipingComponentName : NullableString [0..1]
PipingComponentNumber : NullableString [0..1]

Supertypes

- *PipingComponent*

Subtypes

- *AngleBallValve*
- *AngleGlobeValve*
- *AnglePlugValve*
- *AngleValve*
- *BallValve*
- *ButterflyValve*
- *CustomOperatedValve*
- *GateValve*
- *GlobeValve*
- *NeedleValve*
- *PlugValve*
- *StraightwayValve*

Attributes (data)

Name	Multiplicity	Type
HeatTracingType	0..1	*HeatTracingTypeClassification*
HeatTracingTypeRepresentation	0..1	*NullableString*
InsulationThickness	0..1	*NullableLength*
InsulationType	0..1	*NullableString*
LowerLimitHeatTracingTemperature	0..1	*NullableTemperature*
NumberOfPorts	0..1	*NumberOfPortsClassification*
Operation	0..1	*OperationClassification*
PipingClassCode	0..1	*NullableString*
PipingComponentName	0..1	*NullableString*
PipingComponentNumber	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingComponent>`
RDL reference: OPERATED VALVE
ComponentClass: OperatedValve
ComponentClassURI: http://data.posccaesar.org/rdl/RDS11141590

Example

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="operatedValve1"
    ComponentClass="OperatedValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11141590" ...>
  ...
</PipingComponent>
```

8.39.2 HeatTracingType

Attribute (data)

A specialization indicating the heat tracing type related to the *OperatedValve*.

Multiplicity: 0..1

Type: *HeatTracingTypeClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: HEAT TRACING TYPE SPECIALIZATION
Name: HeatTracingTypeSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/HeatTracingTypeSpecialization

Example

electrical heat tracing system (*HeatTracingTypeClassification::ElectricalHeatTracingSystem*)

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="operatedValve1"
    ComponentClass="OperatedValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11141590" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="HeatTracingTypeSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/HeatTracingTypeSpecialization"
      Format="anyURI"
      Value="ElectricalHeatTracingSystem"
      ValueURI="http://data.posccaesar.org/rdl/RDS11854600" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.39.3 HeatTracingTypeRepresentation

Attribute (data)

The heat tracing type related to the *OperatedValve*, represented as a string.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: HEAT TRACING TYPE REPRESENTATION ASSIGNMENT CLASS
Name: HeatTracingTypeRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/HeatTracingTypeRepresentationAssignmentClass

Example

“E” (*String*)

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="operatedValve1"
    ComponentClass="OperatedValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11141590" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="HeatTracingTypeRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/HeatTracingTypeRepresentationAssignmentClass"
      Format="string"
      Value="E" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.39.4 InsulationThickness

Attribute (data)

The insulation thickness of the *OperatedValve*.

Multiplicity: 0..1

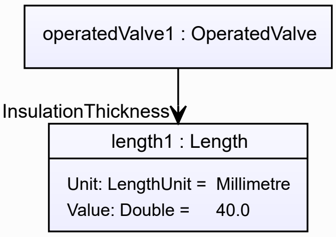
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: INSULATION THICKNESS
Name: InsulationThickness
AttributeURI: http://data.posccaesar.org/rdl/RDS4238040

Example

The instance operatedValve1 represents an *OperatedValve* with an *InsulationThickness* of 40.0 mm.



Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="operatedValve1"
    ComponentClass="OperatedValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11141590" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="InsulationThickness"
      AttributeURI="http://data.posccaesar.org/rdl/RDS4238040"
      Format="double"
      Value="40.0"
      Units="Millimetre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1357739" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.39.5 InsulationType

Attribute (data)

The identification code for the insulation type related to the *OperatedValve*. So far, DEXPI does not define restrictions for valid values.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: INSULATION TYPE ASSIGNMENT CLASS
Name: InsulationTypeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/InsulationTypeAssignmentClass

Example

“Q” (*String*)

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="operatedValve1"
    ComponentClass="OperatedValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11141590" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="InsulationTypeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/InsulationTypeAssignmentClass"
      Format="string"
      Value="Q" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.39.6 LowerLimitHeatTracingTemperature

Attribute (data)

The lower limit for the temperature that a heat tracing system must ensure for the *OperatedValve*.

Multiplicity: 0..1

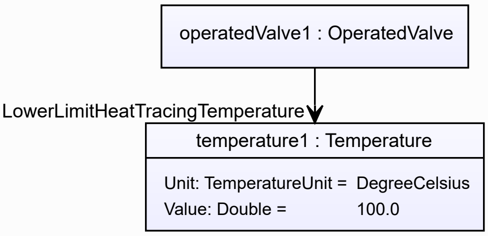
Type: *NullableTemperature*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: LOWER LIMIT HEAT TRACING TEMPERATURE
Name: LowerLimitHeatTracingTemperature
AttributeURI: http://sandbox.dexpi.org/rdl/LowerLimitHeatTracingTemperature

Example

The instance operatedValve1 represents an *OperatedValve* with a *LowerLimitHeatTracingTemperature* of 100.0 °C.


operatedValve1 : OperatedValve
LowerLimitHeatTracingTemperature
temperature1 : Temperature
Unit: TemperatureUnit = DegreeCelsius
Value: Double = 100.0

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="operatedValve1"
    ComponentClass="OperatedValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11141590" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="LowerLimitHeatTracingTemperature"
      AttributeURI="http://sandbox.dexpi.org/rdl/LowerLimitHeatTracingTemperature"
      Format="double"
      Value="100.0"
      Units="DegreeCelsius"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1322684" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.39.7 NumberOfPorts

Attribute (data)

A specialization indicating the number of ports of the *OperatedValve*.

Multiplicity: 0..1

Type: *NumberOfPortsClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: NUMBER OF PORTS SPECIALIZATION
Name: NumberOfPortsSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/NumberOfPortsSpecialization

Example

2 port valve (*NumberOfPortsClassification::TwoPortValve*)

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="operatedValve1"
    ComponentClass="OperatedValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11141590" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NumberOfPortsSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/NumberOfPortsSpecialization"
      Format="anyURI"
      Value="TwoPortValve"
      ValueURI="http://data.posccaesar.org/rdl/RDS11506315" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.39.8 Operation

Attribute (data)

A specialization indicating the operation of the *OperatedValve*.

Multiplicity: 0..1

Type: *OperationClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: OPERATION SPECIALIZATION
Name: OperationSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/OperationSpecialization

Example

continuous operation (*OperationClassification::ContinuousOperation*)

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="operatedValve1"
    ComponentClass="OperatedValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11141590" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="OperationSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/OperationSpecialization"
      Format="anyURI"
      Value="ContinuousOperation"
      ValueURI="http://data.posccaesar.org/rdl/RDS9710162" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.39.9 PipingClassCode

Attribute (data)

The identification code of the piping class of the *OperatedValve*. So far, DEXPI does not define restrictions for valid values.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PIPING CLASS CODE ASSIGNMENT CLASS
Name: PipingClassCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PipingClassCodeAssignmentClass

Example

“75HB13” (*String*)

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="operatedValve1"
    ComponentClass="OperatedValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11141590" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PipingClassCodeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PipingClassCodeAssignmentClass"
      Format="string"
      Value="75HB13" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.39.10 PipingComponentName

Attribute (data)

A string to classify the *OperatedValve*. DEXPI does not prescribe the classification system. Typically, company or site site standards are used.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PIPING COMPONENT NAME ASSIGNMENT CLASS
Name: PipingComponentNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PipingComponentNameAssignmentClass

Example

“73KH12” (*String*)

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="operatedValve1"
    ComponentClass="OperatedValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11141590" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PipingComponentNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PipingComponentNameAssignmentClass"
      Format="string"
      Value="73KH12" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.39.11 PipingComponentNumber

Attribute (data)

An identifier of the *OperatedValve*. DEXPI does not prescribe the scope of the identifier, i.e., whether it should be unique in, e.g., a *PipingNetworkSegment* or a *PipingNetworkSystem*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PIPING COMPONENT NUMBER ASSIGNMENT CLASS
Name: PipingComponentNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PipingComponentNumberAssignmentClass

Example

“C2” (*String*)

Example: Implementation in Proteus Schema

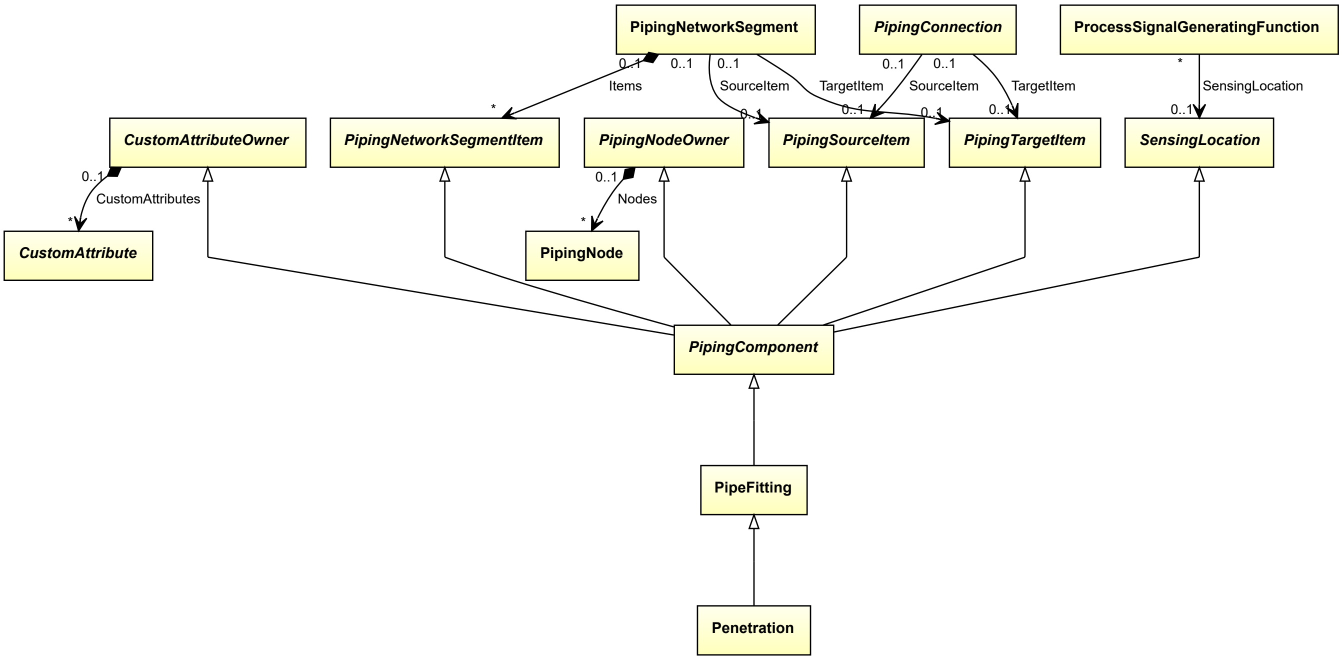
```
<PipingComponent
    ID="operatedValve1"
    ComponentClass="OperatedValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11141590" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PipingComponentNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PipingComponentNumberAssignmentClass"
      Format="string"
      Value="C2" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.40. Penetration

8.40.1 Overview

Class

A device intended to provide a penetration (from http://data.posccaesar.org/rdl/RDS13068275).



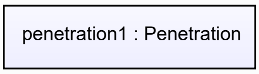
Supertypes

- *PipeFitting*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <PipingComponent>
RDL reference: PENETRATION
ComponentClass: Penetration
ComponentClassURI: http://data.posccaesar.org/rdl/RDS13068275

Example



Example: Implementation in Proteus Schema

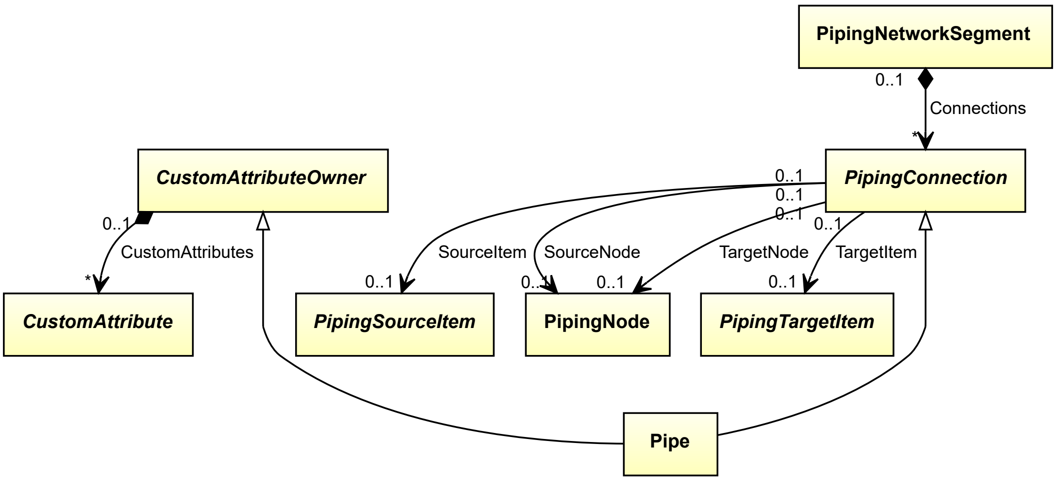
```
<PipingComponent
    ID="penetration1"
    ComponentClass="Penetration"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS13068275" ...>
  ...
</PipingComponent>
```

8.41. Pipe

8.41.1 Overview

Class

An elementary piece of piping, i.e., not interrupted by any item.



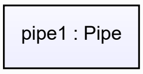
Supertypes

- *CustomAttributeOwner*
- *PipingConnection*

Implementation in Proteus Schema

A *Pipe* is implemented as a `<CenterLine>` element within a `<PipingNetworkSegment>` element. The *SourceItem*, *SourceNode*, *TargetItem*, and *TargetNode* attributes inherited from *PipingConnection* are not directly implemented in Proteus Schema. They are rather given implicitly by the order of `<CenterLine>` and other elements in the `<PipingNetworkSegment>`. For details, see the Proteus Schema specification.

Example



Example: Implementation in Proteus Schema

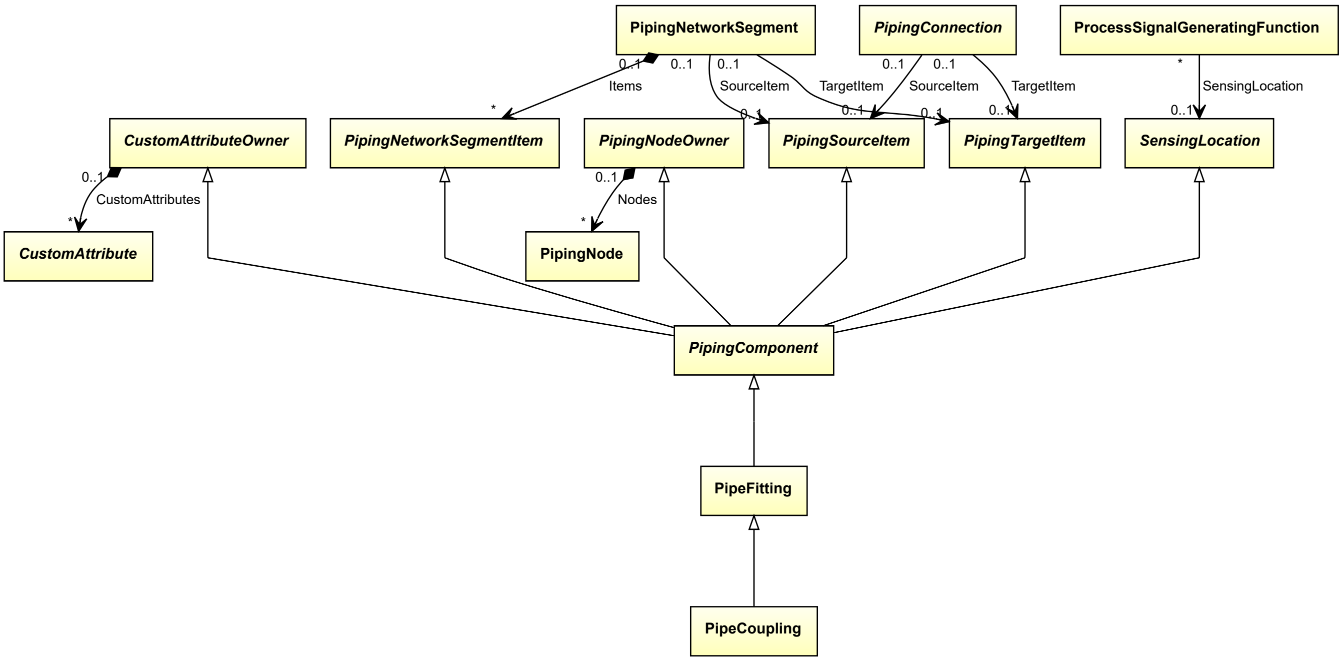
```
<PipingNetworkSegment
    ID="pipingNetworkSegment1"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704">
  ...
  <!--
    Only  a <CenterLine> whose parent is a <PipingNetworkSegment>
    implements a DEXPI Pipe.
  -->
  <CenterLine ...>
    ...
  </CenterLine>
  ...
</PipingNetworkSegment>
```

8.42. PipeCoupling

8.42.1 Overview

Class

An ‘artefact’ that is a one-piece cylindrical section intended to join pipes and/or piping components (from http://data.posccaesar.org/rdl/RDS415664).



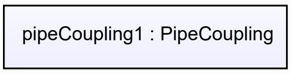
Supertypes

- *PipeFitting*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingComponent>`
RDL reference: PIPE COUPLING
ComponentClass: PipeCoupling
ComponentClassURI: http://data.posccaesar.org/rdl/RDS415664

Example



Example: Implementation in Proteus Schema

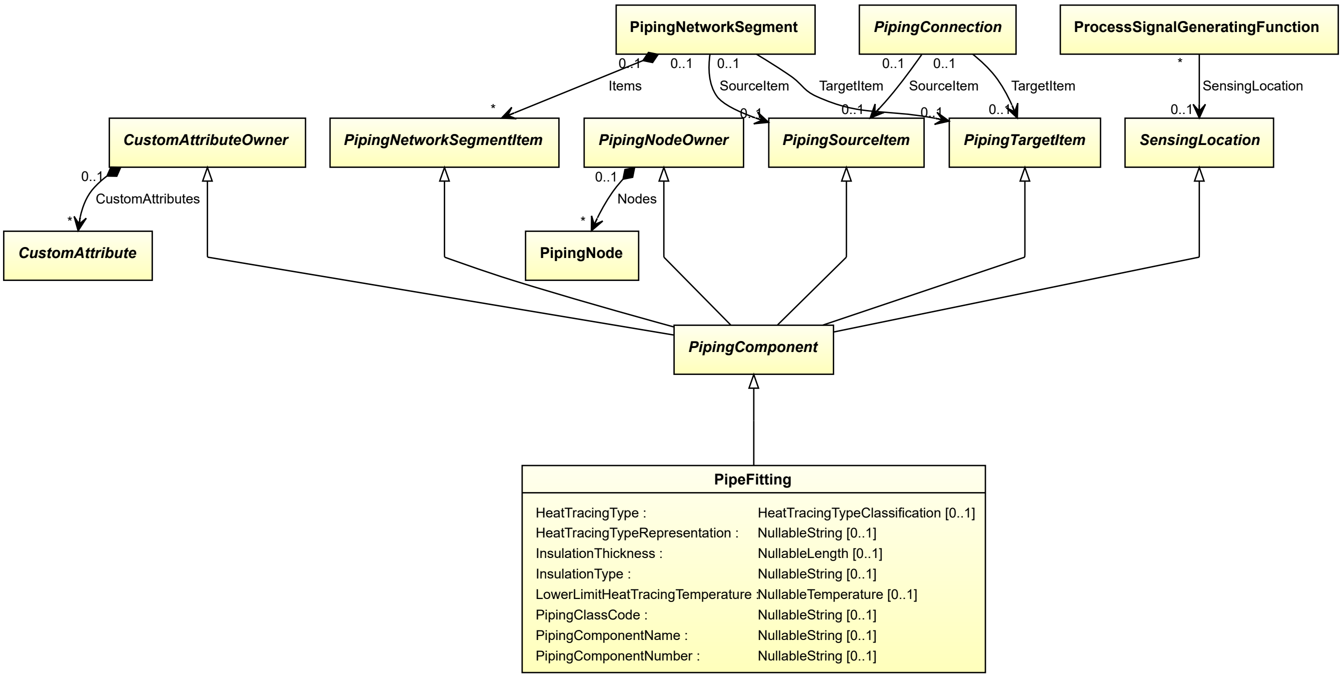
```
<PipingComponent
    ID="pipeCoupling1"
    ComponentClass="PipeCoupling"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS415664" ...>
  ...
</PipingComponent>
```

8.43. PipeFitting

8.43.1 Overview

Class

A pipe fitting.


PipingNetworkSegment
PipingConnection
ProcessSignalGeneratingFunction
0..1
0..1
0..1
Items
SourceItem
TargetItem
0..1
0..1
SourceItem
TargetItem
*
SensingLocation
*
0..1
0..1
0..1
0..1
0..1
CustomAttributeOwner
PipingNetworkSegmentItem
PipingNodeOwner
PipingSourceItem
PipingTargetItem
SensingLocation
0..1
CustomAttributes
0..1
Nodes
*
*
CustomAttribute
PipingNode
PipingComponent
PipeFitting
HeatTracingType : HeatTracingTypeClassification [0..1]
HeatTracingTypeRepresentation : NullableString [0..1]
InsulationThickness : NullableLength [0..1]
InsulationType : NullableString [0..1]
LowerLimitHeatTracingTemperature : NullableTemperature [0..1]
PipingClassCode : NullableString [0..1]
PipingComponentName : NullableString [0..1]
PipingComponentNumber : NullableString [0..1]

Supertypes

- *PipingComponent*

Subtypes

- *BlindFlange*
- *ClampedFlangeCoupling*
- *Compensator*
- *ConicalStrainer*
- *CustomPipeFitting*
- *Flange*
- *FlangedConnection*
- *Funnel*
- *Hose*
- *IlluminatedSightGlass*
- *InLineMixer*
- *LineBlind*
- *Penetration*
- *PipeCoupling*
- *PipeFlangeSpacer*

- *PipeFlangeSpade*
- *PipeReducer*
- *PipeTee*
- *RestrictionOrifice*
- *SightGlass*
- *Silencer*
- *SteamTrap*
- *Strainer*
- *VentilationDevice*

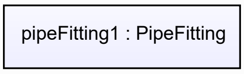
Attributes (data)

Name	Multiplicity	Type
HeatTracingType	0..1	*HeatTracingTypeClassification*
HeatTracingTypeRepresentation	0..1	*NullableString*
InsulationThickness	0..1	*NullableLength*
InsulationType	0..1	*NullableString*
LowerLimitHeatTracingTemperature	0..1	*NullableTemperature*
PipingClassCode	0..1	*NullableString*
PipingComponentName	0..1	*NullableString*
PipingComponentNumber	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <PipingComponent>
RDL reference: PIPE FITTING
ComponentClass: PipeFitting
ComponentClassURI: http://sandbox.dexpi.org/rdl/PipeFitting

Example



Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="pipeFitting1"
    ComponentClass="PipeFitting"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PipeFitting" ...>
  ...
</PipingComponent>
```

8.43.2 HeatTracingType

Attribute (data)

A specialization indicating the heat tracing type related to the *PipeFitting*.

Multiplicity: 0..1

Type: *HeatTracingTypeClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: HEAT TRACING TYPE SPECIALIZATION
Name: HeatTracingTypeSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/HeatTracingTypeSpecialization

Example

electrical heat tracing system (*HeatTracingTypeClassification::ElectricalHeatTracingSystem*)

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="pipeFitting1"
    ComponentClass="PipeFitting"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PipeFitting" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="HeatTracingTypeSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/HeatTracingTypeSpecialization"
      Format="anyURI"
      Value="ElectricalHeatTracingSystem"
      ValueURI="http://data.posccaesar.org/rdl/RDS11854600" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.43.3 HeatTracingTypeRepresentation

Attribute (data)

The heat tracing type related to the *PipeFitting*, represented as a string.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: HEAT TRACING TYPE REPRESENTATION ASSIGNMENT CLASS
Name: HeatTracingTypeRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/HeatTracingTypeRepresentationAssignmentClass

Example

“E” (*String*)

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="pipeFitting1"
    ComponentClass="PipeFitting"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PipeFitting" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="HeatTracingTypeRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/HeatTracingTypeRepresentationAssignmentClass"
      Format="string"
      Value="E" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.43.4 InsulationThickness

Attribute (data)

The insulation thickness of the *PipeFitting*.

Multiplicity: 0..1

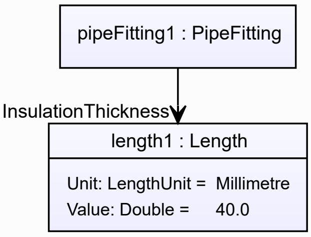
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: INSULATION THICKNESS
Name: InsulationThickness
AttributeURI: http://data.posccaesar.org/rdl/RDS4238040

Example

The instance pipeFitting1 represents a *PipeFitting* with an *InsulationThickness* of 40.0 mm.



Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="pipeFitting1"
    ComponentClass="PipeFitting"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PipeFitting" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="InsulationThickness"
      AttributeURI="http://data.posccaesar.org/rdl/RDS4238040"
      Format="double"
      Value="40.0"
      Units="Millimetre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1357739" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.43.5 InsulationType

Attribute (data)

The identification code for the insulation type related to the *PipeFitting*. So far, DEXPI does not define restrictions for valid values.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: INSULATION TYPE ASSIGNMENT CLASS
Name: InsulationTypeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/InsulationTypeAssignmentClass

Example

“Q” (*String*)

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="pipeFitting1"
    ComponentClass="PipeFitting"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PipeFitting" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="InsulationTypeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/InsulationTypeAssignmentClass"
      Format="string"
      Value="Q" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.43.6 LowerLimitHeatTracingTemperature

Attribute (data)

The lower limit for the temperature that a heat tracing system must ensure for the *PipeFitting*.

Multiplicity: 0..1

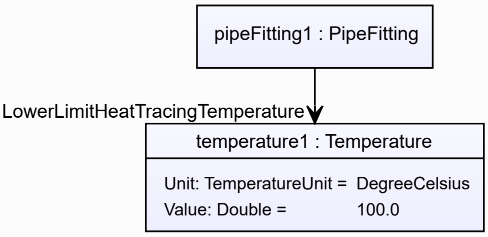
Type: *NullableTemperature*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: LOWER LIMIT HEAT TRACING TEMPERATURE
Name: LowerLimitHeatTracingTemperature
AttributeURI: http://sandbox.dexpi.org/rdl/LowerLimitHeatTracingTemperature

Example

The instance pipeFitting1 represents a *PipeFitting* with a *LowerLimitHeatTracingTemperature* of 100.0 °C.


pipeFitting1 : PipeFitting
LowerLimitHeatTracingTemperature
temperature1 : Temperature
Unit: TemperatureUnit = DegreeCelsius
Value: Double = 100.0

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="pipeFitting1"
    ComponentClass="PipeFitting"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PipeFitting" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="LowerLimitHeatTracingTemperature"
      AttributeURI="http://sandbox.dexpi.org/rdl/LowerLimitHeatTracingTemperature"
      Format="double"
      Value="100.0"
      Units="DegreeCelsius"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1322684" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.43.7 PipingClassCode

Attribute (data)

The identification code of the piping class of the *PipeFitting*. So far, DEXPI does not define restrictions for valid values.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PIPING CLASS CODE ASSIGNMENT CLASS
Name: PipingClassCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PipingClassCodeAssignmentClass

Example

"75HB13" (*String*)

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="pipeFitting1"
    ComponentClass="PipeFitting"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PipeFitting" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PipingClassCodeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PipingClassCodeAssignmentClass"
      Format="string"
      Value="75HB13" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.43.8 PipingComponentName

Attribute (data)

A string to classify the *PipeFitting*. DEXPI does not prescribe the classification system. Typically, company or site site standards are used.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PIPING COMPONENT NAME ASSIGNMENT CLASS
Name: PipingComponentNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PipingComponentNameAssignmentClass

Example

"73KH12" (*String*)

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="pipeFitting1"
    ComponentClass="PipeFitting"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PipeFitting" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PipingComponentNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PipingComponentNameAssignmentClass"
      Format="string"
      Value="73KH12" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.43.9 PipingComponentNumber

Attribute (data)

An identifier of the *PipeFitting*. DEXPI does not prescribe the scope of the identifier, i.e., whether it should be unique in, e.g., a *PipingNetworkSegment* or a *PipingNetworkSystem*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PIPING COMPONENT NUMBER ASSIGNMENT CLASS
Name: PipingComponentNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PipingComponentNumberAssignmentClass

Example

“C2” (*String*)

Example: Implementation in Proteus Schema

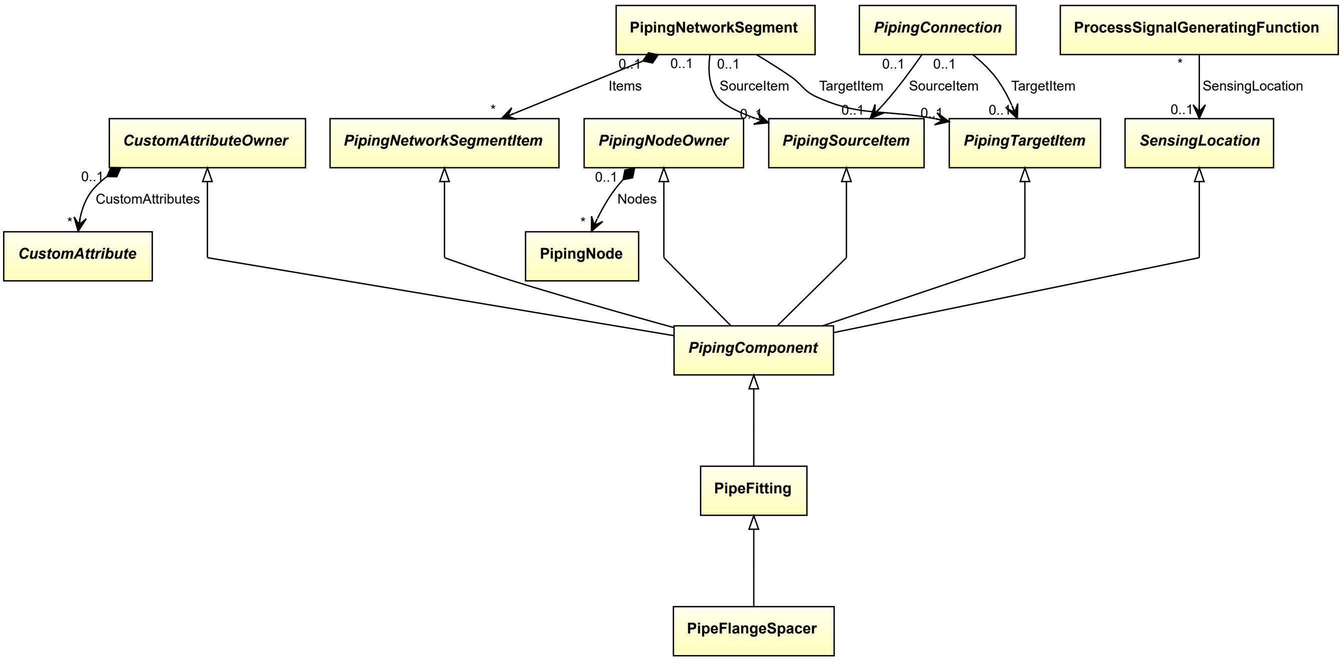
```
<PipingComponent
    ID="pipeFitting1"
    ComponentClass="PipeFitting"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PipeFitting" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PipingComponentNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PipingComponentNumberAssignmentClass"
      Format="string"
      Value="C2" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.44. PipeFlangeSpacer

8.44.1 Overview

Class

A ‘spacer’ and an ‘artefact’ that is intended to be inserted between two pipe flanged ends to provide the distance between the flanges required to insert a ‘pipe flange spade’ (from http://data.posccaesar.org/rdl/RDS472724).



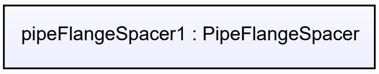
Supertypes

- *PipeFitting*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <PipingComponent>
RDL reference: PIPE FLANGE SPACER
ComponentClass: PipeFlangeSpacer
ComponentClassURI: http://data.posccaesar.org/rdl/RDS472724

Example



Example: Implementation in Proteus Schema

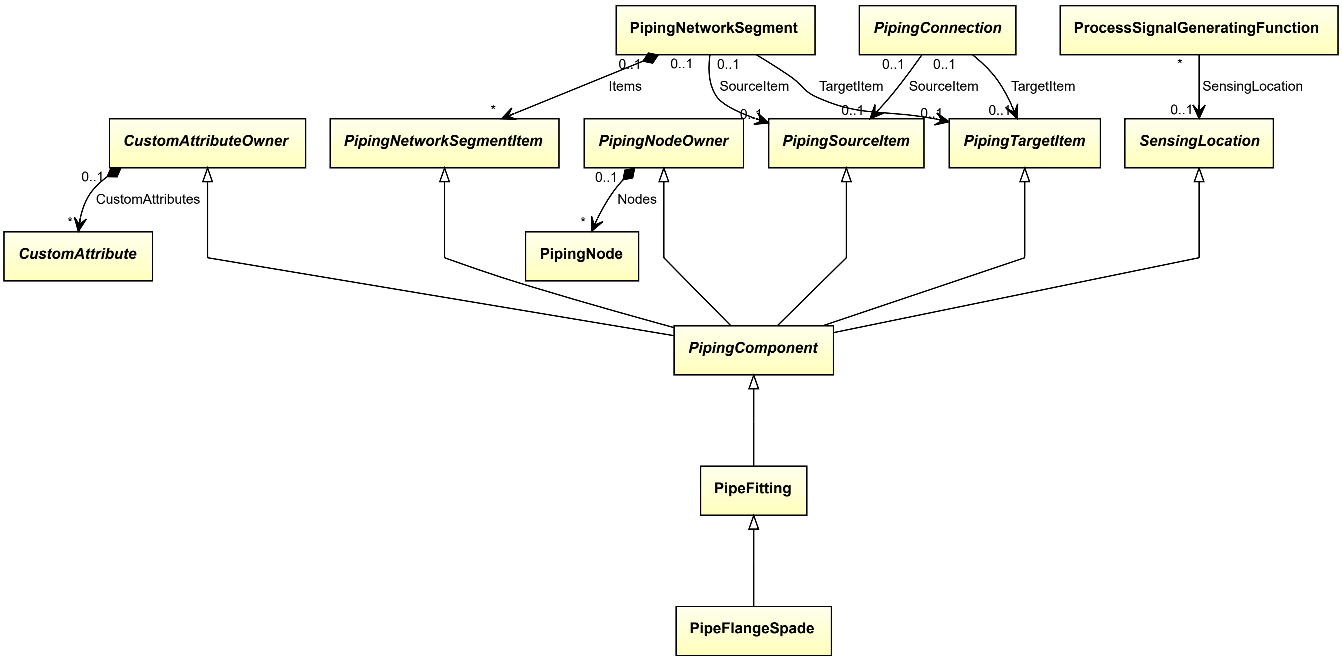
```
<PipingComponent
    ID="pipeFlangeSpacer1"
    ComponentClass="PipeFlangeSpacer"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS472724" ...>
  ...
</PipingComponent>
```

8.45. PipeFlangeSpade

8.45.1 Overview

Class

A ‘line blind’ and an ‘artefact’ that is a circular plate with no central opening and holes to match mating flanged ends. It is also equipped with a handle (from http://data.posccaesar.org/rdl/RDS472679).



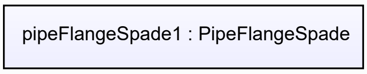
Supertypes

- *PipeFitting*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <PipingComponent>
RDL reference: PIPE FLANGE SPADE
ComponentClass: PipeFlangeSpade
ComponentClassURI: http://data.posccaesar.org/rdl/RDS472679

Example



Example: Implementation in Proteus Schema

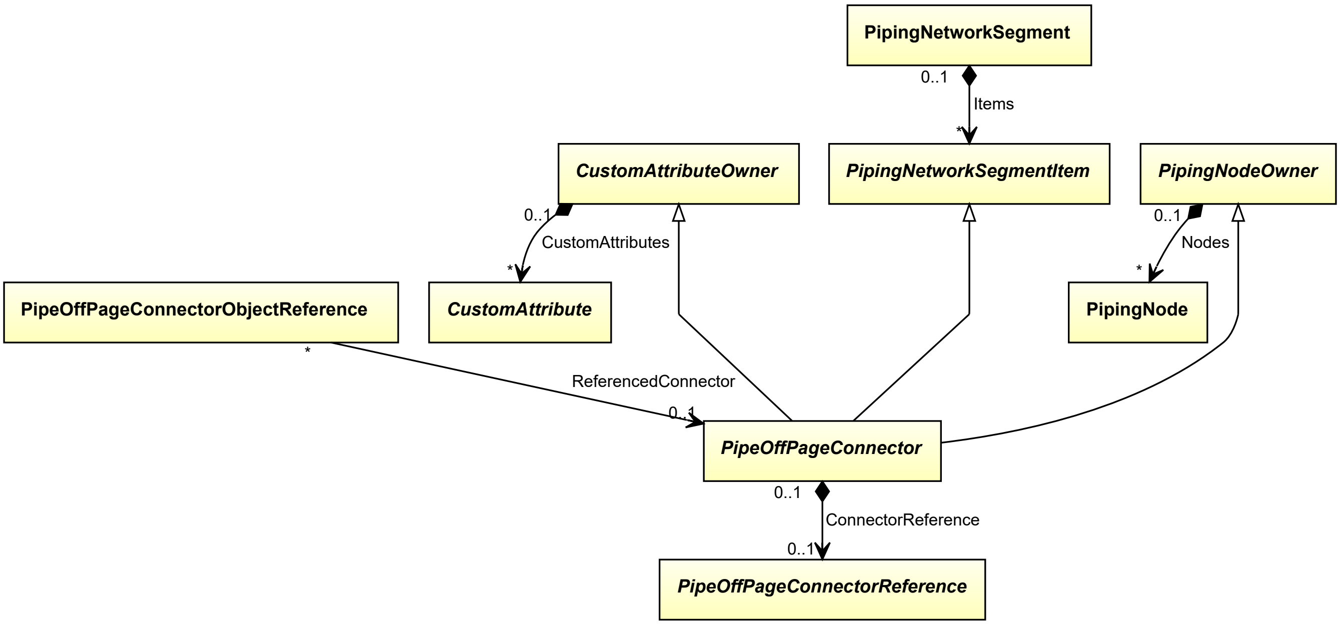
```
<PipingComponent
    ID="pipeFlangeSpade1"
    ComponentClass="PipeFlangeSpade"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS472679" ...>
  ...
</PipingComponent>
```

8.46. PipeOffPageConnector

8.46.1 Overview

Abstract class

A connector that indicates that a piping network segment is continued elsewhere, either on the same PID or on another PID. Graphically, it is usually represented as an arrow.



Supertypes

- *CustomAttributeOwner*
- *PipingNetworkSegmentItem*
- *PipingNodeOwner*

Subtypes

- *FlowInPipeOffPageConnector*
- *FlowOutPipeOffPageConnector*

Attributes (composition)

Name	Multiplicity	Type
ConnectorReference	0..1	*PipeOffPageConnectorReference*

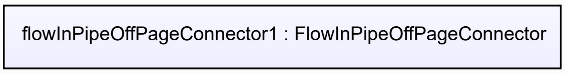
Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*. As *PipeOffPageConnector* is abstract, there is no RDL reference for the class itself; the RDL reference depends on the concrete subclass.
Tag: `<PipeOffPageConnector>`
ComponentClass: *depending on subclass*

ComponentClassURI: *depending on subclass*

Example

As *PipeOffPageConnector* is abstract, we consider *FlowInPipeOffPageConnector* as an arbitrary concrete subclass.



Example: Implementation in Proteus Schema

```
<PipeOffPageConnector
    ID="flowInPipeOffPageConnector1"
    ComponentClass="FlowInPipeOffPageConnector"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/FlowInPipeOffPageConnector" ...>
  ...
</PipeOffPageConnector>
```

8.46.2 ConnectorReference

Attribute (composition)

A reference indicating to which other *PipeOffPageConnector* this *PipeOffPageConnector* is connected.

Multiplicity: 0..1

Type: *PipeOffPageConnectorReference*

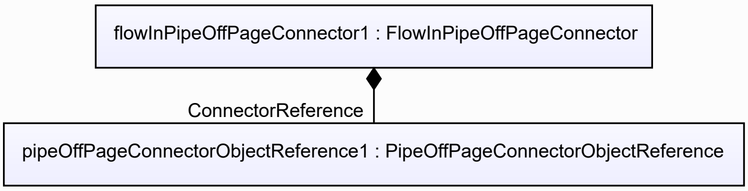
Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *PipeOffPageConnectorReference*) is a child of the `<PipeOffPageConnector>` element for the attribute owner (a *PipeOffPageConnector*).

Example

As the owner type *PipeOffPageConnector* is abstract, we consider *FlowInPipeOffPageConnector* as an arbitrary concrete subclass. As the value type *PipeOffPageConnectorReference* is abstract, we consider *PipeOffPageConnectorObjectReference* as an arbitrary concrete subclass.


flowInPipeOffPageConnector1 : FlowInPipeOffPageConnector
ConnectorReference
pipeOffPageConnectorObjectReference1 : PipeOffPageConnectorObjectReference

Example: Implementation in Proteus Schema

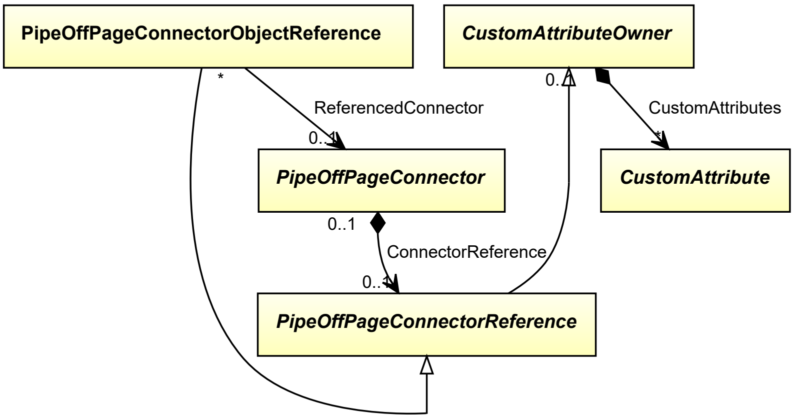
```
<PipeOffPageConnector
    ID="flowInPipeOffPageConnector1"
    ComponentClass="FlowInPipeOffPageConnector"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/FlowInPipeOffPageConnector" ...>
  ...
  <PipeOffPageConnectorReference
      ID="pipeOffPageConnectorObjectReference1"
      ComponentClass="PipeOffPageConnectorObjectReference"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/PipeOffPageConnectorObjectReference" ...>
    ...
  <PipeOffPageConnectorReference />
  ...
<PipeOffPageConnector />
```

8.47. PipeOffPageConnectorObjectReference

8.47.1 Overview

Class

A reference to a *PipeOffPageConnector* by an association.



Supertypes

- *PipeOffPageConnectorReference*

Attributes (reference)

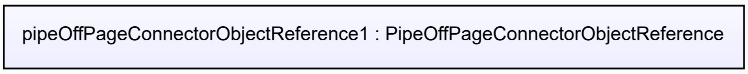
Name	Multiplicity	Type
ReferencedConnector	0..1	*PipeOffPageConnector*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipeOffPageConnectorReference>`
RDL reference: PIPE OFF PAGE CONNECTOR OBJECT REFERENCE
ComponentClass: PipeOffPageConnectorObjectReference

ComponentClassURI: http://sandbox.dexpi.org/rdl/PipeOffPageConnectorObjectReference

Example


pipeOffPageConnectorObjectReference1 : PipeOffPageConnectorObjectReference

Example: Implementation in Proteus Schema

```
<PipeOffPageConnectorReference
    ID="pipeOffPageConnectorObjectReference1"
    ComponentClass="PipeOffPageConnectorObjectReference"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PipeOffPageConnectorObjectReference" ...>
  ...
</PipeOffPageConnectorReference>
```

8.47.2 ReferencedConnector

Attribute (reference)

The *PipeOffPageConnector* referenced.

Multiplicity: 0..1

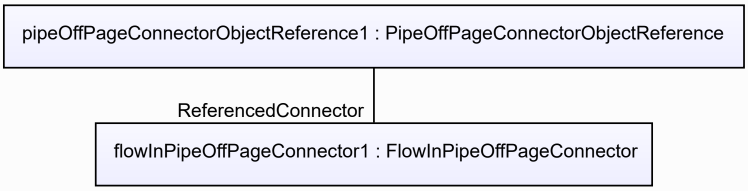
Type: *PipeOffPageConnector*

Opposite multiplicity: 0..*

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: "refers to"
Opposite association type: "is referenced by"

Example


pipeOffPageConnectorObjectReference1 : PipeOffPageConnectorObjectReference
ReferencedConnector
flowInPipeOffPageConnector1 : FlowInPipeOffPageConnector

Example: Implementation in Proteus Schema

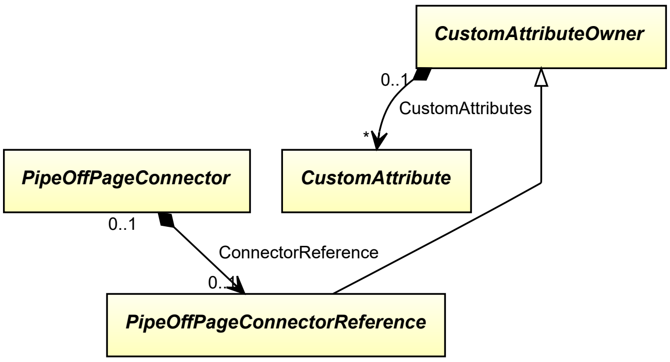
```
<PipeOffPageConnectorReference
    ID="pipeOffPageConnectorObjectReference1"
    ComponentClass="PipeOffPageConnectorObjectReference"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PipeOffPageConnectorObjectReference" ...>
  ...
  <Association
    Type="refers to"
    ItemID="flowInPipeOffPageConnector1" />
  ...
<PipeOffPageConnectorReference />
...
<PipeOffPageConnector
    ID="flowInPipeOffPageConnector1"
    ComponentClass="FlowInPipeOffPageConnector"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/FlowInPipeOffPageConnector" ...>
  ...
  <Association
    Type="is referenced by"
    ItemID="pipeOffPageConnectorObjectReference1" />
  ...
<PipeOffPageConnector />
```

8.48. PipeOffPageConnectorReference

8.48.1 Overview

Abstract class

A reference to a *PipeOffPageConnector*.



Supertypes

- *CustomAttributeOwner*

Subtypes

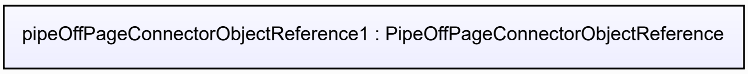
- *PipeOffPageConnectorObjectReference*
- *PipeOffPageConnectorReferenceByNumber*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*. As *PipeOffPageConnectorReference* is abstract, there is no RDL reference for the class itself; the RDL reference depends on the concrete subclass.
Tag: `<PipeOffPageConnectorReference>`
ComponentClass: *depending on subclass*
ComponentClassURI: *depending on subclass*

Example

As *PipeOffPageConnectorReference* is abstract, we consider *PipeOffPageConnectorObjectReference* as an arbitrary concrete subclass.


pipeOffPageConnectorObjectReference1 : PipeOffPageConnectorObjectReference

Example: Implementation in Proteus Schema

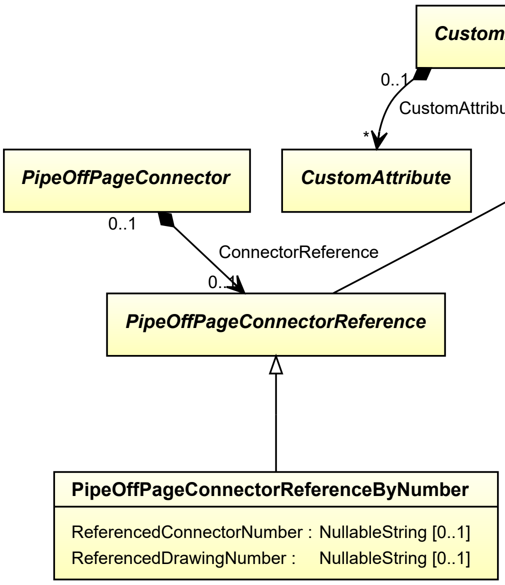
```
<PipeOffPageConnectorReference
    ID="pipeOffPageConnectorObjectReference1"
    ComponentClass="PipeOffPageConnectorObjectReference"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PipeOffPageConnectorObjectReference" ...>
  ...
</PipeOffPageConnectorReference>
```

8.49. PipeOffPageConnectorReferenceByNumber

8.49.1 Overview

Class

A reference to a *PipeOffPageConnector* by drawing and connector number.


CustomAttributeOwner
0..1
CustomAttributes
*
PipeOffPageConnector
CustomAttribute
0..1
ConnectorReference
0..1
PipeOffPageConnectorReference
PipeOffPageConnectorReferenceByNumber
ReferencedConnectorNumber : NullableString [0..1]
ReferencedDrawingNumber : NullableString [0..1]

Supertypes

- *PipeOffPageConnectorReference*

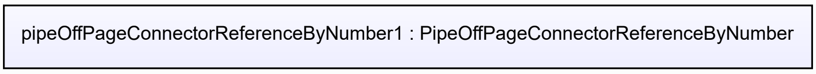
Attributes (data)

Name	Multiplicity	Type
ReferencedConnectorNumber	0..1	*NullableString*
ReferencedDrawingNumber	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <PipeOffPageConnectorReference>
RDL reference: PIPE OFF PAGE CONNECTOR REFERENCE BY NUMBER
ComponentClass: PipeOffPageConnectorReferenceByNumber
ComponentClassURI: http://sandbox.dexpi.org/rdl/PipeOffPageConnectorReferenceByNumber

Example


pipeOffPageConnectorReferenceByNumber1 : PipeOffPageConnectorReferenceByNumber

Example: Implementation in Proteus Schema

```
<PipeOffPageConnectorReference
    ID="pipeOffPageConnectorReferenceByNumber1"
    ComponentClass="PipeOffPageConnectorReferenceByNumber"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PipeOffPageConnectorReferenceByNumber" ...>
  ...
</PipeOffPageConnectorReference>
```

8.49.2 ReferencedConnectorNumber

Attribute (data)

The connector number of the referenced connector.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: REFERENCED CONNECTOR NUMBER ASSIGNMENT CLASS
Name: ReferencedConnectorNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ReferencedConnectorNumberAssignmentClass

Example

“97” (*String*)

Example: Implementation in Proteus Schema

```
<PipeOffPageConnectorReference
    ID="pipeOffPageConnectorReferenceByNumber1"
    ComponentClass="PipeOffPageConnectorReferenceByNumber"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PipeOffPageConnectorReferenceByNumber" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ReferencedConnectorNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ReferencedConnectorNumberAssignmentClass"
      Format="string"
      Value="97" />
    ...
  </GenericAttributes>
  ...
</PipeOffPageConnectorReference>
```

8.49.3 ReferencedDrawingNumber

Attribute (data)

The *DrawingNumber* of the PID that contains the referenced connector.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: REFERENCED DRAWING NUMBER ASSIGNMENT CLASS
Name: ReferencedDrawingNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ReferencedDrawingNumberAssignmentClass

Example

"123/A93" (*String*)

Example: Implementation in Proteus Schema

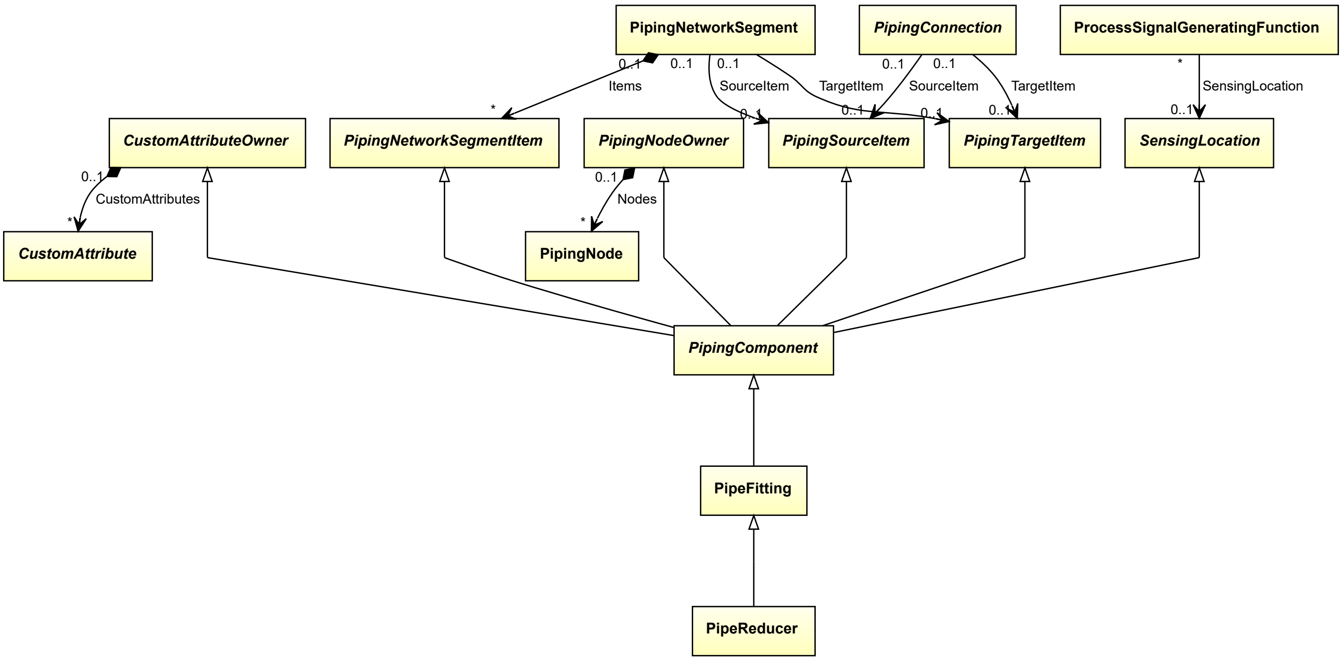
```
<PipeOffPageConnectorReference
    ID="pipeOffPageConnectorReferenceByNumber1"
    ComponentClass="PipeOffPageConnectorReferenceByNumber"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PipeOffPageConnectorReferenceByNumber" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ReferencedDrawingNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ReferencedDrawingNumberAssignmentClass"
      Format="string"
      Value="123/A93" />
    ...
  </GenericAttributes>
  ...
</PipeOffPageConnectorReference>
```

8.50. PipeReducer

8.50.1 Overview

Class

An 'artefact' that has different nominal pipe size at the two ends, intended to connect pipes or piping components (from http://data.posccaesar.org/rdl/RDS416294).



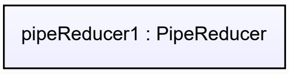
Supertypes

- *PipeFitting*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingComponent>`
RDL reference: PIPE REDUCER
ComponentClass: PipeReducer
ComponentClassURI: http://data.posccaesar.org/rdl/RDS416294

Example



Example: Implementation in Proteus Schema

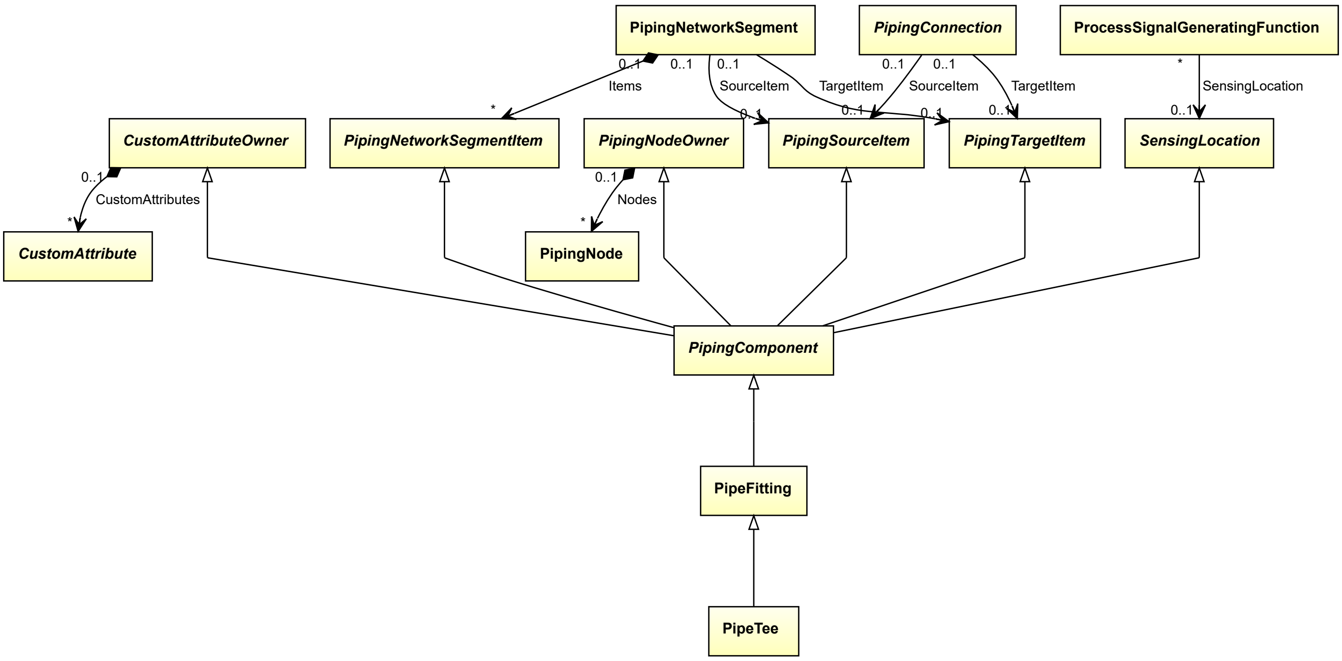
```
<PipingComponent
    ID="pipeReducer1"
    ComponentClass="PipeReducer"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS416294" ...>
  ...
</PipingComponent>
```

8.51. PipeTee

8.51.1 Overview

Class

An ‘artefact’ that has three piping ends in T-shape, including a branch at 90 degrees (from http://data.posccaesar.org/rdl/RDS427724).



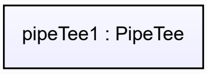
Supertypes

- *PipeFitting*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingComponent>`
RDL reference: PIPE TEE
ComponentClass: PipeTee
ComponentClassURI: http://data.posccaesar.org/rdl/RDS427724

Example



Example: Implementation in Proteus Schema

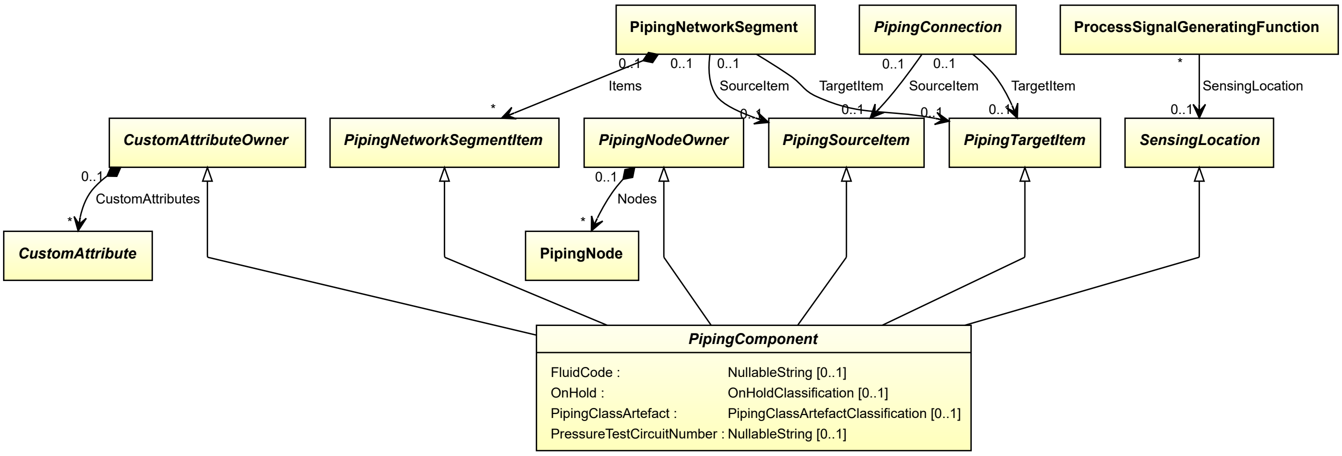
```
<PipingComponent
    ID="pipeTee1"
    ComponentClass="PipeTee"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS427724" ...>
  ...
</PipingComponent>
```

8.52. PipingComponent

8.52.1 Overview

Abstract class

A piping component



Supertypes

- *CustomAttributeOwner*
- *PipingNetworkSegmentItem*
- *PipingNodeOwner*
- *PipingSourceItem*
- *PipingTargetItem*
- *SensingLocation*

Subtypes

- *CheckValve*
- *CustomPipingComponent*
- *InlinePrimaryElement*
- *OperatedValve*
- *PipeFitting*
- *SafetyValveOrFitting*

Attributes (data)

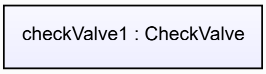
Name	Multiplicity	Type
FluidCode	0..1	*NullableString*
OnHold	0..1	*OnHoldClassification*
PipingClassArtefact	0..1	*PipingClassArtefactClassification*
PressureTestCircuitNumber	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*. As *PipingComponent* is abstract, there is no RDL reference for the class itself; the RDL reference depends on the concrete subclass.
Tag: <PipingComponent>
ComponentClass: *depending on subclass*
ComponentClassURI: *depending on subclass*

Example

As *PipingComponent* is abstract, we consider *CheckValve* as an arbitrary concrete subclass.



Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="checkValve1"
    ComponentClass="CheckValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS292229" ...>
  ...
</PipingComponent>
```

8.52.2 FluidCode

Attribute (data)

The identification code of the fluid related to the *PipingComponent*. So far, DEXPI does not define restrictions for valid values.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: FLUID CODE ASSIGNMENT CLASS
Name: FluidCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/FluidCodeAssignmentClass

Example

As the owning class *PipingComponent* is abstract, we consider *CheckValve* as an arbitrary concrete subclass.
“MNb” (*String*)

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="checkValve1"
    ComponentClass="CheckValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS292229" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="FluidCodeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/FluidCodeAssignmentClass"
      Format="string"
      Value="MNb" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.52.3 OnHold

Attribute (data)

A specialization indicating if the *PipingComponent* is on hold or not.

Multiplicity: 0..1

Type: *OnHoldClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: ON HOLD SPECIALIZATION
Name: OnHoldSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/OnHoldSpecialization

Example

As the owning class *PipingComponent* is abstract, we consider *CheckValve* as an arbitrary concrete subclass. on hold (*OnHoldClassification::OnHold*)

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="checkValve1"
    ComponentClass="CheckValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS292229" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="OnHoldSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/OnHoldSpecialization"
      Format="anyURI"
      Value="OnHold"
      ValueURI="http://sandbox.dexpi.org/rdl/OnHold" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.52.4 PipingClassArtefact

Attribute (data)

A specialization indicating if the *PipingComponent* is an artefact that is described by a piping class.

Multiplicity: 0..1

Type: *PipingClassArtefactClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: PIPING CLASS ARTEFACT SPECIALIZATION
Name: PipingClassArtefactSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/PipingClassArtefactSpecialization

Example

As the owning class *PipingComponent* is abstract, we consider *CheckValve* as an arbitrary concrete subclass. piping class artefact (*PipingClassArtefactClassification::PipingClassArtefact*)

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="checkValve1"
    ComponentClass="CheckValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS292229" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PipingClassArtefactSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/PipingClassArtefactSpecialization"
      Format="anyURI"
      Value="PipingClassArtefact"
      ValueURI="http://sandbox.dexpi.org/rdl/PipingClassArtefact" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.52.5 PressureTestCircuitNumber

Attribute (data)

The number of the pressure test circuit of the *PipingComponent*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PRESSURE TEST CIRCUIT NUMBER ASSIGNMENT CLASS
Name: PressureTestCircuitNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PressureTestCircuitNumberAssignmentClass

Example

As the owning class *PipingComponent* is abstract, we consider *CheckValve* as an arbitrary concrete subclass.
“TC123” (*String*)

Example: Implementation in Proteus Schema

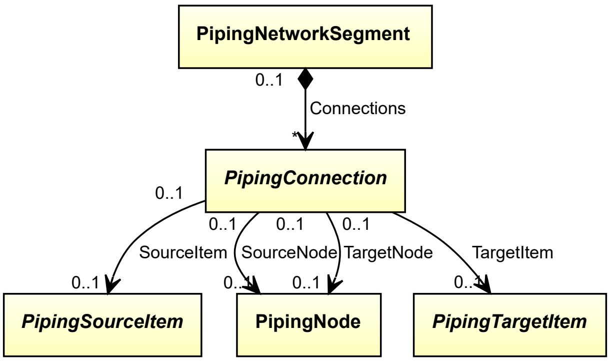
```
<PipingComponent
    ID="checkValve1"
    ComponentClass="CheckValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS292229" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PressureTestCircuitNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PressureTestCircuitNumberAssignmentClass"
      Format="string"
      Value="TC123" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.53. PipingConnection

8.53.1 Overview

Abstract class

An elementary connection between two piping items.



Subtypes

- *DirectPipingConnection*
- *Pipe*

Attributes (reference)

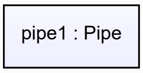
Name	Multiplicity	Type
SourceItem	0..1	*PipingSourceItem*
SourceNode	0..1	*PipingNode*
TargetItem	0..1	*PipingTargetItem*
TargetNode	0..1	*PipingNode*

Implementation in Proteus Schema

Implementation is subclass-specific.

Example

As *PipingConnection* is abstract, we consider *Pipe* as an arbitrary concrete subclass.



Example: Implementation in Proteus Schema

```
<PipingNetworkSegment
    ID="pipingNetworkSegment1"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704">
  ...
  <!--
    Only  a <CenterLine> whose parent is a <PipingNetworkSegment>
    implements a DEXPI Pipe.
  -->
  <CenterLine ...>
    ...
  </CenterLine>
  ...
</PipingNetworkSegment>
```

8.53.2 SourceItem

Attribute (reference)

The *PipingSourceItem* at which the *PipingConnection* starts.

Multiplicity: 0..1

Type: *PipingSourceItem*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

Implementation is subclass-specific.

8.53.3 SourceNode

Attribute (reference)

The *PipingNode* at which the *PipingConnection* starts. The SourceNode must belong to the *SourceItem*.

Multiplicity: 0..1

Type: *PipingNode*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

Implementation is subclass-specific.

8.53.4 TargetItem

Attribute (reference)

The *PipingTargetItem* at which the *PipingConnection* starts.

Multiplicity: 0..1

Type: *PipingTargetItem*

Opposite multiplicity: 0..1

> Implementation in Proteus Schema
>
> Implementation is subclass-specific.

8.53.5 TargetNode

Attribute (reference)

The *PipingNode* at which the *PipingConnection* ends. The TargetNode must belong to the *TargetItem*.

Multiplicity: 0..1

Type: *PipingNode*

Opposite multiplicity: 0..1

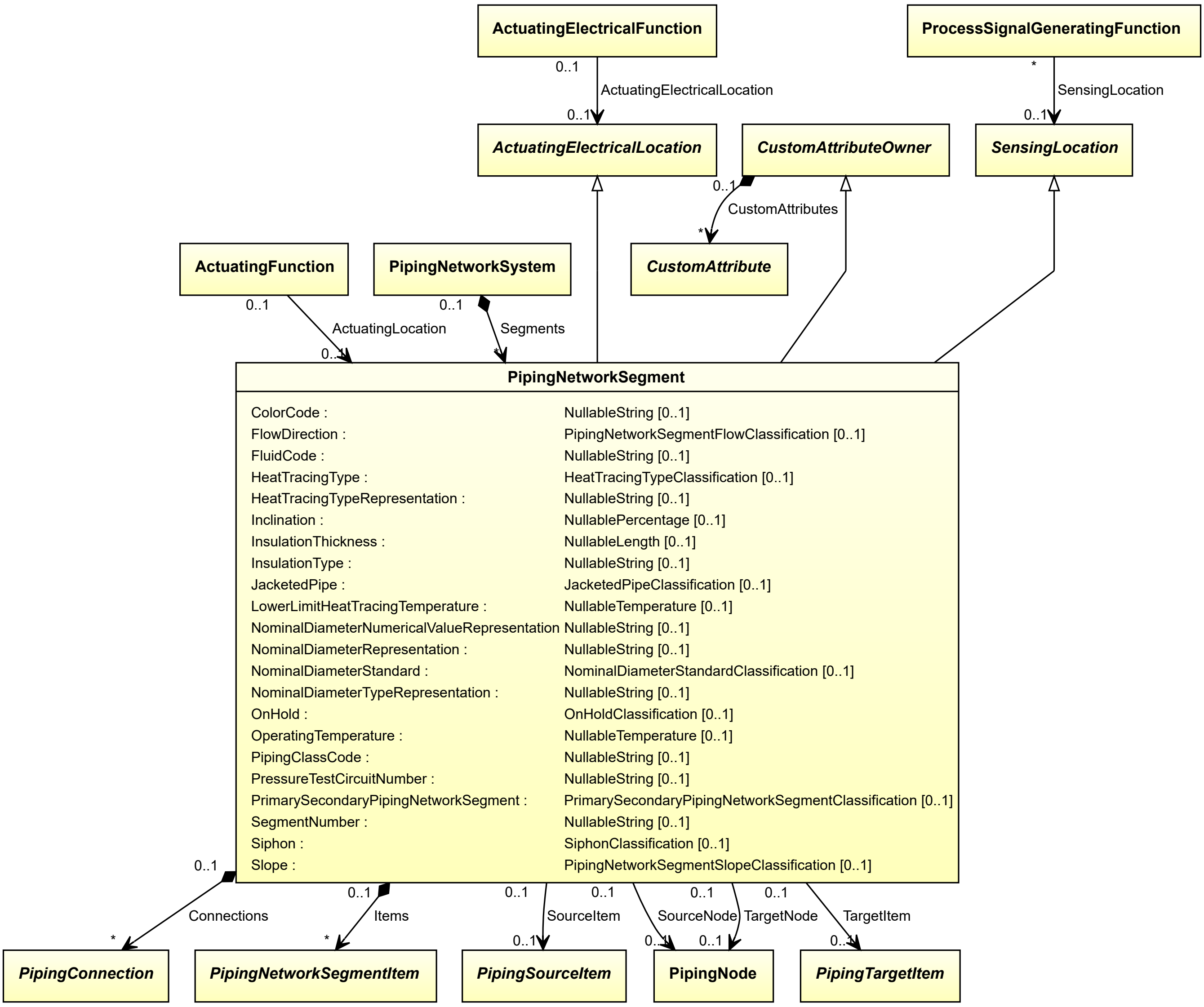
> Implementation in Proteus Schema
>
> Implementation is subclass-specific.

8.54. PipingNetworkSegment

8.54.1 Overview

Class

The piping limited by a Node and a Break, Node and Connector, two Nodes, two Breaks, two Connectors or a Break and a Connector. The last five providing there are no Breaks or Connectors in between. In the last three cases the Segment will coincide with a Piping Branch (from http://data.posccaesar.org/rdl/RDS267704).


ActuatingElectricalFunction
ProcessSignalGeneratingFunction
0..1
ActuatingElectricalLocation
0..1
*
SensingLocation
0..1
ActuatingElectricalLocation
CustomAttributeOwner
SensingLocation
0..1
CustomAttributes
*
ActuatingFunction
PipingNetworkSystem
CustomAttribute
0..1
ActuatingLocation
0..1
0..1
Segments
*
PipingNetworkSegment
ColorCode : NullableString [0..1]
FlowDirection : PipingNetworkSegmentFlowClassification [0..1]
FluidCode : NullableString [0..1]
HeatTracingType : HeatTracingTypeClassification [0..1]
HeatTracingTypeRepresentation : NullableString [0..1]
Inclination : NullablePercentage [0..1]
InsulationThickness : NullableLength [0..1]
InsulationType : NullableString [0..1]
JacketedPipe : JacketedPipeClassification [0..1]
LowerLimitHeatTracingTemperature : NullableTemperature [0..1]
NominalDiameterNumericalValueRepresentation NullableString [0..1]
NominalDiameterRepresentation : NullableString [0..1]
NominalDiameterStandard : NominalDiameterStandardClassification [0..1]
NominalDiameterTypeRepresentation : NullableString [0..1]
OnHold : OnHoldClassification [0..1]
OperatingTemperature : NullableTemperature [0..1]
PipingClassCode : NullableString [0..1]
PressureTestCircuitNumber : NullableString [0..1]
PrimarySecondaryPipingNetworkSegment : PrimarySecondaryPipingNetworkSegmentClassification [0..1]
SegmentNumber : NullableString [0..1]
Siphon : SiphonClassification [0..1]
Slope : PipingNetworkSegmentSlopeClassification [0..1]
0..1
0..1
0..1
0..1
0..1
0..1
Connections
Items
SourceItem
SourceNode
TargetNode
TargetItem
*
*
0..1
0..1
0..1
0..1
PipingConnection
PipingNetworkSegmentItem
PipingSourceItem
PipingNode
PipingTargetItem

Supertypes

- *ActuatingElectricalLocation*
- *CustomAttributeOwner*
- *SensingLocation*

Attributes (data)

Name	Multiplicity	Type
ColorCode	0..1	*NullableString*
FlowDirection	0..1	*PipingNetworkSegmentFlowClassification*
FluidCode	0..1	*NullableString*
HeatTracingType	0..1	*HeatTracingTypeClassification*
HeatTracingTypeRepresentation	0..1	*NullableString*
Inclination	0..1	*NullablePercentage*
InsulationThickness	0..1	*NullableLength*
InsulationType	0..1	*NullableString*

(continued on next page)

Name	Multiplicity	Type
JacketedPipe	0..1	*JacketedPipeClassification*
LowerLimitHeatTracingTemperature	0..1	*NullableTemperature*
NominalDiameterNumericalValueRepresentation	0..1	*NullableString*
NominalDiameterRepresentation	0..1	*NullableString*
NominalDiameterStandard	0..1	*NominalDiameterStandardClassification*
NominalDiameterTypeRepresentation	0..1	*NullableString*
OnHold	0..1	*OnHoldClassification*
OperatingTemperature	0..1	*NullableTemperature*
PipingClassCode	0..1	*NullableString*
PressureTestCircuitNumber	0..1	*NullableString*
PrimarySecondaryPipingNetworkSegment	0..1	*PrimarySecondaryPipingNetworkSegmentClassification*
SegmentNumber	0..1	*NullableString*
Siphon	0..1	*SiphonClassification*
Slope	0..1	*PipingNetworkSegmentSlopeClassification*

Attributes (composition)

Name	Multiplicity	Type
Connections	*	*PipingConnection*
Items	*	*PipingNetworkSegmentItem*

Attributes (reference)

Name	Multiplicity	Type
SourceItem	0..1	*PipingSourceItem*
SourceNode	0..1	*PipingNode*
TargetItem	0..1	*PipingTargetItem*
TargetNode	0..1	*PipingNode*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingNetworkSegment>`
RDL reference: PIPING NETWORK SEGMENT
ComponentClass: PipingNetworkSegment
ComponentClassURI: http://data.posccaesar.org/rdl/RDS267704

Example

pipingNetworkSegment1 : PipingNetworkSegment

Example: Implementation in Proteus Schema

```
<PipingNetworkSegment
    ID="pipingNetworkSegment1"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704" ...>
  ...
</PipingNetworkSegment>
```

8.54.2 ColorCode

Attribute (data)

The color code of the *PipingNetworkSegment*, represented as a string.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: COLOR CODE ASSIGNMENT CLASS
Name: ColorCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ColorCodeAssignmentClass

Example

"C321" (*String*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSegment
    ID="pipingNetworkSegment1"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ColorCodeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ColorCodeAssignmentClass"
      Format="string"
      Value="C321" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSegment>
```

8.54.3 Connections

Attribute (composition)

The connections of the *PipingNetworkSegment*.

Multiplicity: *

Type: *PipingConnection*

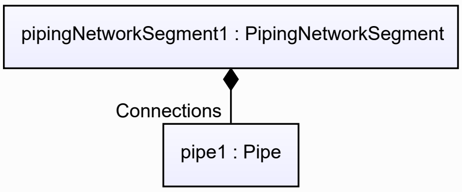
Opposite multiplicity: 0..1

Implementation in Proteus Schema

In case the *PipingConnection* is a *Pipe*, the corresponding `<CenterLine>` element is a child of the `<PipingNetworkSegment>`. Two `<CenterLine>` elements must be separated by at least one element representing a *PipingNetworkSegmentItem*, e.g., a `<PipingComponent>`, in order to implement two *Pipes*; otherwise, the two `<CenterLine>` elements would represent a single *Pipe* whose graphical representation contains a visual "gap", e.g., when another *Pipe* crosses.
In case the *PipingConnection* is a *DirectPipingConnection*, there is no corresponding Proteus element. A *DirectPipingConnection* is rather given implicitly, e.g., by two successive `<PipingComponent>` elements (see Proteus Schema Implementation of *DirectPipingConnection*).

Example

As the value type *PipingConnection* is abstract, we consider *Pipe* as an arbitrary concrete subclass.



Example: Implementation in Proteus Schema

```
<PipingNetworkSegment
    ID="pipingNetworkSegment1"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704" ...>
  ...
  <CenterLine ...>
    <!-- pipe1 -->
    ...
  </CenterLine>
  ...
</PipingNetworkSegment>
```

8.54.4 FlowDirection

Attribute (data)

A specialization indicating if the *PipingNetworkSegment* enables dual flow or not.

Multiplicity: 0..1

Type: *PipingNetworkSegmentFlowClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: FLOW DIRECTION SPECIALIZATION
Name: FlowDirectionSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/FlowDirectionSpecialization

Example

dual flow (*PipingNetworkSegmentFlowClassification::DualFlowPipingNetworkSegment*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSegment
    ID="pipingNetworkSegment1"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="FlowDirectionSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/FlowDirectionSpecialization"
      Format="anyURI"
      Value="DualFlowPipingNetworkSegment"
      ValueURI="http://sandbox.dexpi.org/rdl/DualFlowPipingNetworkSegment" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSegment>
```

8.54.5 FluidCode

Attribute (data)

The identification code of the fluid related to the *PipingNetworkSegment*. So far, DEXPI does not define restrictions for valid values.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: FLUID CODE ASSIGNMENT CLASS
Name: FluidCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/FluidCodeAssignmentClass

Example

“MNb” (*String*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSegment
    ID="pipingNetworkSegment1"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="FluidCodeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/FluidCodeAssignmentClass"
      Format="string"
      Value="MNb" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSegment>
```

8.54.6 HeatTracingType

Attribute (data)

A specialization indicating the heat tracing type related to the *PipingNetworkSegment*.

Multiplicity: 0..1

Type: *HeatTracingTypeClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: HEAT TRACING TYPE SPECIALIZATION
Name: HeatTracingTypeSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/HeatTracingTypeSpecialization

Example

electrical heat tracing system (*HeatTracingTypeClassification::ElectricalHeatTracingSystem*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSegment
    ID="pipingNetworkSegment1"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="HeatTracingTypeSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/HeatTracingTypeSpecialization"
      Format="anyURI"
      Value="ElectricalHeatTracingSystem"
      ValueURI="http://data.posccaesar.org/rdl/RDS11854600" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSegment>
```

8.54.7 HeatTracingTypeRepresentation

Attribute (data)

The heat tracing type related to the *PipingNetworkSegment*, represented as a string.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: HEAT TRACING TYPE REPRESENTATION ASSIGNMENT CLASS
Name: HeatTracingTypeRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/HeatTracingTypeRepresentationAssignmentClass

Example

"E" (*String*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSegment
    ID="pipingNetworkSegment1"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="HeatTracingTypeRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/HeatTracingTypeRepresentationAssignmentClass"
      Format="string"
      Value="E" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSegment>
```

8.54.8 Inclination

Attribute (data)

The inclination (slope) of the *PipingNetworkSegment* in percent.

Multiplicity: 0..1

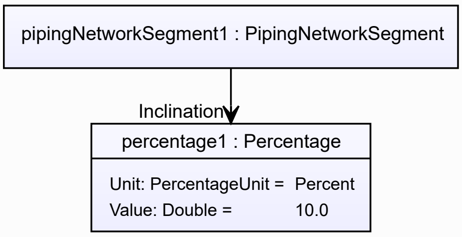
Type: *NullablePercentage*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: INCLINATION
Name: Inclination
AttributeURI: http://data.posccaesar.org/rdl/RDS17688057

Example

The instance pipingNetworkSegment1 represents a *PipingNetworkSegment* with an *Inclination* of 10.0 ???.


pipingNetworkSegment1 : PipingNetworkSegment
Inclination
percentage1 : Percentage
Unit: PercentageUnit = Percent
Value: Double = 10.0

Example: Implementation in Proteus Schema

```
<PipingNetworkSegment
    ID="pipingNetworkSegment1"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="Inclination"
      AttributeURI="http://data.posccaesar.org/rdl/RDS17688057"
      Format="double"
      Value="10.0"
      Units="Percent"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1317959" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSegment>
```

8.54.9 InsulationThickness

Attribute (data)

The insulation thickness of the *PipingNetworkSegment*.

Multiplicity: 0..1

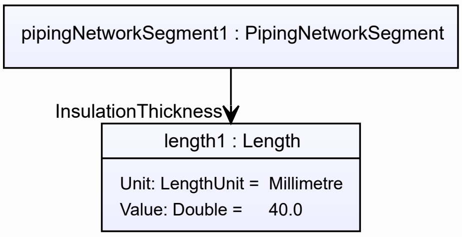
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: INSULATION THICKNESS
Name: InsulationThickness
AttributeURI: http://data.posccaesar.org/rdl/RDS4238040

Example

The instance pipingNetworkSegment1 represents a *PipingNetworkSegment* with an *InsulationThickness* of 40.0 mm.


pipingNetworkSegment1 : PipingNetworkSegment
InsulationThickness
length1 : Length
Unit: LengthUnit = Millimetre
Value: Double = 40.0

Example: Implementation in Proteus Schema

```
<PipingNetworkSegment
    ID="pipingNetworkSegment1"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="InsulationThickness"
      AttributeURI="http://data.posccaesar.org/rdl/RDS4238040"
      Format="double"
      Value="40.0"
      Units="Millimetre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1357739" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSegment>
```

8.54.10 InsulationType

Attribute (data)

The identification code for the insulation type related to the *PipingNetworkSegment*. So far, DEXPI does not define restrictions for valid values.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: INSULATION TYPE ASSIGNMENT CLASS
Name: InsulationTypeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/InsulationTypeAssignmentClass

Example

“Q” (*String*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSegment
    ID="pipingNetworkSegment1"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="InsulationTypeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/InsulationTypeAssignmentClass"
      Format="string"
      Value="Q" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSegment>
```

8.54.11 Items

Attribute (composition)

The items of the *PipingNetworkSegment*.

Multiplicity: *

Type: *PipingNetworkSegmentItem*

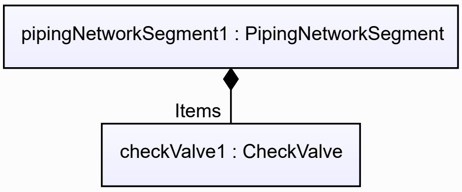
Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The `<PipeOffPageConnector>`, `<PipingComponent>`, or `<PropertyBreak>` elements for the attribute value (a *PipingNetworkSegmentItem*) is a child of the `<PipingNetworkSegment>` element for the attribute owner (a *PipingNetworkSegment*).

Example

As the value type *PipingNetworkSegmentItem* is abstract, we consider *CheckValve* as an arbitrary concrete subclass.



Example: Implementation in Proteus Schema

```
<PipingNetworkSegment
    ID="pipingNetworkSegment1"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704" ...>
  ...
  <PipingComponent
      ID="checkValve1"
      ComponentClass="CheckValve"
      ComponentClassURI="http://data.posccaesar.org/rdl/RDS292229" ...>
    ...
  <PipingComponent />
  ...
<PipingNetworkSegment />
```

8.54.12 JacketedPipe

Attribute (data)

A specialization indicating whether the *PipingNetworkSegment* is jacketed.

Multiplicity: 0..1

Type: *JacketedPipeClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: JACKETED PIPE SPECIALIZATION

Name: JacketedPipeSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/JacketedPipeSpecialization

Example

jacketed (*JacketedPipeClassification::JacketedPipe*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSegment
    ID="pipingNetworkSegment1"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="JacketedPipeSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/JacketedPipeSpecialization"
      Format="anyURI"
      Value="JacketedPipe"
      ValueURI="http://sandbox.dexpi.org/rdl/JacketedPipe" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSegment>
```

8.54.13 LowerLimitHeatTracingTemperature

Attribute (data)

The lower limit for the temperature that a heat tracing system must ensure for the *PipingNetworkSegment*.

Multiplicity: 0..1

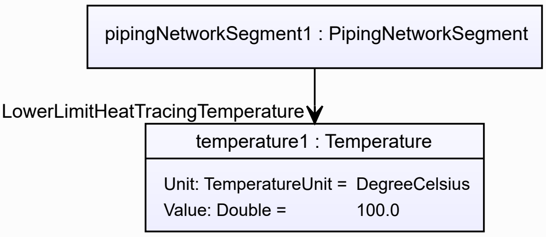
Type: *NullableTemperature*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: LOWER LIMIT HEAT TRACING TEMPERATURE
Name: LowerLimitHeatTracingTemperature
AttributeURI: http://sandbox.dexpi.org/rdl/LowerLimitHeatTracingTemperature

Example

The instance pipingNetworkSegment1 represents a *PipingNetworkSegment* with a *LowerLimitHeatTracingTemperature* of 100.0 °C.


pipingNetworkSegment1 : PipingNetworkSegment
LowerLimitHeatTracingTemperature
temperature1 : Temperature
Unit: TemperatureUnit = DegreeCelsius
Value: Double = 100.0

Example: Implementation in Proteus Schema

```
<PipingNetworkSegment
    ID="pipingNetworkSegment1"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="LowerLimitHeatTracingTemperature"
      AttributeURI="http://sandbox.dexpi.org/rdl/LowerLimitHeatTracingTemperature"
      Format="double"
      Value="100.0"
      Units="DegreeCelsius"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1322684" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSegment>
```

8.54.14 NominalDiameterNumericalValueRepresentation

Attribute (data)

A readable representation of the numerical value of the nominal diameter of the *PipingNetworkSegment*, without any type or unit of measure.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: NOMINAL DIAMETER NUMERICAL VALUE REPRESENTATION ASSIGNMENT CLASS
Name: NominalDiameterNumericalValueRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/NominalDiameterNumericalValueRepresentationAssignmentClass

Example

"25" (*String*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSegment
    ID="pipingNetworkSegment1"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NominalDiameterNumericalValueRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/NominalDiameterNumericalValueRepresentationAssignmentClass"
      Format="string"
      Value="25" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSegment>
```

8.54.15 NominalDiameterRepresentation

Attribute (data)

A readable representation of the nominal diameter of the *PipingNetworkSegment*. It normally contains a numerical value and a type or unit of measure.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: NOMINAL DIAMETER REPRESENTATION ASSIGNMENT CLASS
Name: NominalDiameterRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/NominalDiameterRepresentationAssignmentClass

Example

"DN 25" (*String*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSegment
    ID="pipingNetworkSegment1"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NominalDiameterRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/NominalDiameterRepresentationAssignmentClass"
      Format="string"
      Value="DN 25" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSegment>
```

8.54.16 NominalDiameterStandard

Attribute (data)

The nominal diameter of the *PipingNetworkSegment*, given as a reference to a nominal diameter standard and value.

Multiplicity: 0..1

Type: *NominalDiameterStandardClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: NOMINAL DIAMETER STANDARD SPECIALIZATION
Name: NominalDiameterStandardSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/NominalDiameterStandardSpecialization

Example

DN 25 (DIN 2448) (*NominalDiameterStandardClassification::Din2448ObjectDn25*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSegment
    ID="pipingNetworkSegment1"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NominalDiameterStandardSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/NominalDiameterStandardSpecialization"
      Format="anyURI"
      Value="Din2448ObjectDn25"
      ValueURI="http://sandbox.dexpi.org/rdl/Din2448ObjectDn25" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSegment>
```

8.54.17 NominalDiameterTypeRepresentation

Attribute (data)

A readable representation of the type or unit of measure of the nominal diameter of the *PipingNetworkSegment*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: NOMINAL DIAMETER TYPE REPRESENTATION ASSIGNMENT CLASS
Name: NominalDiameterTypeRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/NominalDiameterTypeRepresentationAssignmentClass

Example

“DN” (*String*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSegment
    ID="pipingNetworkSegment1"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NominalDiameterTypeRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/NominalDiameterTypeRepresentationAssignmentClass"
      Format="string"
      Value="DN" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSegment>
```

8.54.18 OnHold

Attribute (data)

A specialization indicating if the *PipingNetworkSegment* is on hold or not.

Multiplicity: 0..1

Type: *OnHoldClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: ON HOLD SPECIALIZATION
Name: OnHoldSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/OnHoldSpecialization

Example

on hold (*OnHoldClassification::OnHold*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSegment
    ID="pipingNetworkSegment1"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="OnHoldSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/OnHoldSpecialization"
      Format="anyURI"
      Value="OnHold"
      ValueURI="http://sandbox.dexpi.org/rdl/OnHold" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSegment>
```

8.54.19 OperatingTemperature

Attribute (data)

The operating temperature of the *PipingNetworkSegment*.

Multiplicity: 0..1

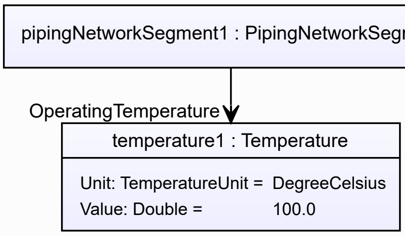
Type: *NullableTemperature*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: OPERATING TEMPERATURE
Name: OperatingTemperature
AttributeURI: http://data.posccaesar.org/rdl/RDS357119

Example

The instance pipingNetworkSegment1 represents a *PipingNetworkSegment* with an *OperatingTemperature* of 100.0 °C.


pipingNetworkSegment1 : PipingNetworkSegment
OperatingTemperature
temperature1 : Temperature
Unit: TemperatureUnit = DegreeCelsius
Value: Double = 100.0

Example: Implementation in Proteus Schema

```
<PipingNetworkSegment
    ID="pipingNetworkSegment1"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="OperatingTemperature"
      AttributeURI="http://data.posccaesar.org/rdl/RDS357119"
      Format="double"
      Value="100.0"
      Units="DegreeCelsius"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1322684" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSegment>
```

8.54.20 PipingClassCode

Attribute (data)

The identification code of the piping class of the *PipingNetworkSegment*. So far, DEXPI does not define restrictions for valid values.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PIPING CLASS CODE ASSIGNMENT CLASS
Name: PipingClassCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PipingClassCodeAssignmentClass

Example

“75HB13” (*String*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSegment
    ID="pipingNetworkSegment1"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PipingClassCodeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PipingClassCodeAssignmentClass"
      Format="string"
      Value="75HB13" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSegment>
```

8.54.21 PressureTestCircuitNumber

Attribute (data)

The number of the pressure test circuit of the *PipingNetworkSegment*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PRESSURE TEST CIRCUIT NUMBER ASSIGNMENT CLASS
Name: PressureTestCircuitNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PressureTestCircuitNumberAssignmentClass

Example

“TC123” (*String*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSegment
    ID="pipingNetworkSegment1"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PressureTestCircuitNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PressureTestCircuitNumberAssignmentClass"
      Format="string"
      Value="TC123" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSegment>
```

8.54.22 PrimarySecondaryPipingNetworkSegment

Attribute (data)

A specialization indicating whether the *PipingNetworkSegment* is a primary or secondary *PipingNetworkSegment*.

Multiplicity: 0..1

Type: *PrimarySecondaryPipingNetworkSegmentClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: PRIMARY SECONDARY PIPING NETWORK SEGMENT SPECIALIZATION
Name: PrimarySecondaryPipingNetworkSegmentSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/PrimarySecondaryPipingNetworkSegmentSpecialization

Example

primary segment (*PrimarySecondaryPipingNetworkSegmentClassification::PrimaryPipingNetworkSegment*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSegment
    ID="pipingNetworkSegment1"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PrimarySecondaryPipingNetworkSegmentSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/PrimarySecondaryPipingNetworkSegmentSpecialization"
      Format="anyURI"
      Value="PrimaryPipingNetworkSegment"
      ValueURI="http://sandbox.dexpi.org/rdl/PrimaryPipingNetworkSegment" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSegment>
```

8.54.23 SegmentNumber

Attribute (data)

The segment number of a *PipingNetworkSegment*. Values are typically (but not necessarily) string representations of numbers with a prefix.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: SEGMENT NUMBER ASSIGNMENT CLASS
Name: SegmentNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/SegmentNumberAssignmentClass

Example

"S3" (*String*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSegment
    ID="pipingNetworkSegment1"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="SegmentNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/SegmentNumberAssignmentClass"
      Format="string"
      Value="S3" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSegment>
```

8.54.24 Siphon

Attribute (data)

A specialization indicating if the *PipingNetworkSegment* is a siphon or not.

Multiplicity: 0..1

Type: *SiphonClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: SIPHON SPECIALIZATION
Name: SiphonSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/SiphonSpecialization

Example

siphon (*SiphonClassification::Siphon*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSegment
    ID="pipingNetworkSegment1"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="SiphonSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/SiphonSpecialization"
      Format="anyURI"
      Value="Siphon"
      ValueURI="http://data.posccaesar.org/rdl/RDS311084" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSegment>
```

8.54.25 Slope

Attribute (data)

A specialization indicating if the *PipingNetworkSegment* is sloped or not.

Multiplicity: 0..1

Type: *PipingNetworkSegmentSlopeClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: SLOPE SPECIALIZATION
Name: SlopeSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/SlopeSpecialization

Example

sloped (*PipingNetworkSegmentSlopeClassification::SlopedPipingNetworkSegment*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSegment
    ID="pipingNetworkSegment1"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="SlopeSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/SlopeSpecialization"
      Format="anyURI"
      Value="SlopedPipingNetworkSegment"
      ValueURI="http://sandbox.dexpi.org/rdl/SlopedPipingNetworkSegment" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSegment>
```

8.54.26 SourceItem

Attribute (reference)

The item at which the PipingNetworkSegment starts.

Multiplicity: 0..1

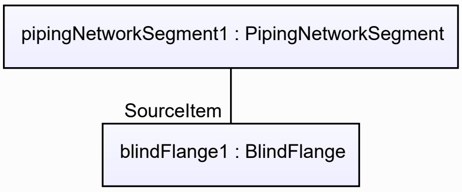
Type: *PipingSourceItem*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The SourceItem is given by means of the `FromID` XML attribute of the `<Connection>` element in the `<PipingNetworkSegment>` element. The value of the `FromID` XML attribute is the XML `ID` of the XML element corresponding to the SourceItem, e.g., a *Nozzle* or a *PipingComponent*.

Example


pipingNetworkSegment1 : PipingNetworkSegment
SourceItem
blindFlange1 : BlindFlange

Example: Implementation in Proteus Schema

The XML fragment demonstrates the case that the SourceItem of the `<PipingNetworkSegment>` is the first item of the segment itself.

```
<PipingNetworkSegment
    ID="pipingNetworkSegment1"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704"
    ...>
  ...
  <PipingComponent
      ID="blindFlange1"
      ComponentClass="BlindFlange"
      ComponentClassURI="http://data.posccaesar.org/rdl/RDS414719"
      ...>
    <!-- This is the first item of the PipingNetworkSegment. -->
    ...
  </PipingComponent>
  ...
  <Connection FromID="blindFlange1" ...>
  ...
</PipingNetworkSegment>
```

8.54.27 SourceNode

Attribute (reference)

The *PipingNode* at which the *PipingNetworkSegment* starts.

Multiplicity: 0..1

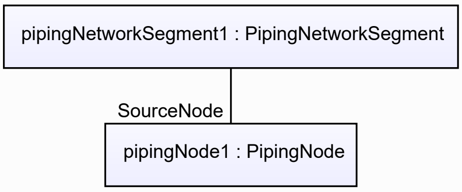
Type: *PipingNode*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The SourceNode is given by means of the `FromNode` XML attribute of the `<Connection>` element in the `<PipingNetworkSegment>` element. The value of the `FromNode` XML attribute is an integer. It refers to the zero-based index of the `<Node>` element within the `<ConnectionPoints>` element associated with the owner of the *PipingNode*. The owner itself is given by means of the `FromID` XML attribute of the `<Connection>` element (see Proteus Schema Implementation of *SourceItem*).
For details, see Proteus P&ID File Specification. Note that in certain cases, Proteus Schema allows to omit the `FromNode` attribute when it is clear from the context.

Example



Example: Implementation in Proteus Schema

The XML fragment demonstrates the case that the SourceNode belongs to a *PipeTee* in another *PipingNetworkSegment*.

```
<PipingNetworkSegment
    ID="pipingNetworkSegment1"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704"
    ...>
  ...
  <Connection FromID="pipeTee1" FromNode="2"/>
  ...
</PipingNetworkSegment>
...
<PipingNetworkSegment
    ID="pipingNetworkSegment2"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704"
    ...>
  ...
  <PipingComponent
      ID="pipeTee1"
      ComponentClass="PipeTee"
      ComponentClassURI="http://data.posccaesar.org/rdl/RDS427724"
      ...>
    ...
    <ConnectionPoints NumPoints="4" ...>
      <Node ...> ... </Node>
      <Node ...> ... </Node>
      <Node ID="pipingNode1" Type="process" ...>
        <!-- This node has index 2. -->
      </Node>
      <Node ...> ... </Node>
    </ConnectionPoints>
    ...
  </PipingComponent>

  ...
</PipingNetworkSegment>
```

8.54.28 TargetItem

Attribute (reference)

The item at which the PipingNetworkSegment ends.

Multiplicity: 0..1

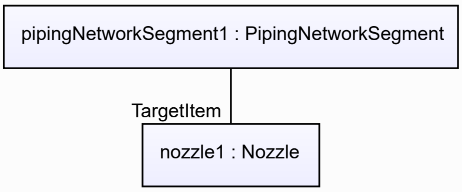
Type: *PipingTargetItem*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The TargetItem is given by means of the ToID XML attribute of the <Connection> element in the <PipingNetworkSegment> element. The value of the ToID XML attribute is the XML ID of the XML element corresponding to the TargetItem, e.g., a *Nozzle* or a *PipingComponent*.

Example



Example: Implementation in Proteus Schema

The XML fragment demonstrates the case that the TargetItem of the <PipingNetworkSegment> is a *Nozzle* of some *Equipment* item.

```
<PipingNetworkSegment
    ID="pipingNetworkSegment1"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704"
    ...>
  ...
  <Connection ToID="nozzle1" ...>
  ...
</PipingNetworkSegment>
...
<Equipment ...>
  ...
  <Nozzle
      ID="nozzle1"
      ComponentClass="Nozzle"
      ComponentClassURI="http://data.posccaesar.org/rdl/RDS415214"
      ...>
    ...
  </Nozzle>
  ...
</Equipment>
```

8.54.29 TargetNode

Attribute (reference)

The Node at which the PipingNetworkSegment ends.

Multiplicity: 0..1

Type: *PipingNode*

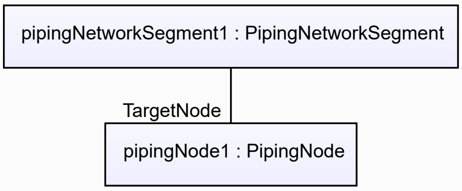
Opposite multiplicity: 0..1

Implementation in Proteus Schema

The TargetNode is given by means of the ToNode XML attribute of the <Connection> element in the <PipingNetworkSegment> element. The value of the ToNode XML attribute is an integer. It refers to the zero-based index of the <Node> element within the <ConnectionPoints> element associated with the owner of the *PipingNode*. The owner itself is given by means of the ToID XML attribute of the <Connection> element (see Proteus Schema Implementation of *TargetItem*).
For details, see Proteus P&ID File Specification. Note that in certain cases, Proteus Schema allows to omit the

`ToNode` attribute when it is clear from the context.

Example



Example: Implementation in Proteus Schema

The XML fragment demonstrates the case that the TargetNode belongs to a *PipeTee* in another *PipingNetworkSegment*.

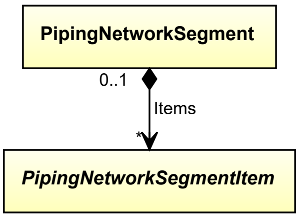
```
<PipingNetworkSegment
    ID="pipingNetworkSegment1"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704"
    ...>
  ...
  <Connection ToID="pipeTee1" ToNode="2"/>
  ...
</PipingNetworkSegment>
...
<PipingNetworkSegment
    ID="pipingNetworkSegment2"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704"
    ...>
  ...
  <PipingComponent
      ID="pipeTee1"
      ComponentClass="PipeTee"
      ComponentClassURI="http://data.posccaesar.org/rdl/RDS427724"
      ...>
    ...
    <ConnectionPoints NumPoints="4" ...>
      <Node ...> ... </Node>
      <Node ...> ... </Node>
      <Node ID="pipingNode1" Type="process" ...>
        <!-- This node has index 2. -->
      </Node>
      <Node ...> ... </Node>
    </ConnectionPoints>
    ...
  </PipingComponent>

  ...
</PipingNetworkSegment>
```

8.55. PipingNetworkSegmentItem

8.55.1 Overview

Abstract class

An item that can be part of a *PipingNetworkSegment*.



Subtypes

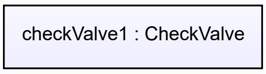
- *PipeOffPageConnector*
- *PipingComponent*
- *PropertyBreak*

Implementation in Proteus Schema

Implementation is subclass-specific.

Example

As *PipingNetworkSegmentItem* is abstract, we consider *CheckValve* as an arbitrary concrete subclass.



Example: Implementation in Proteus Schema

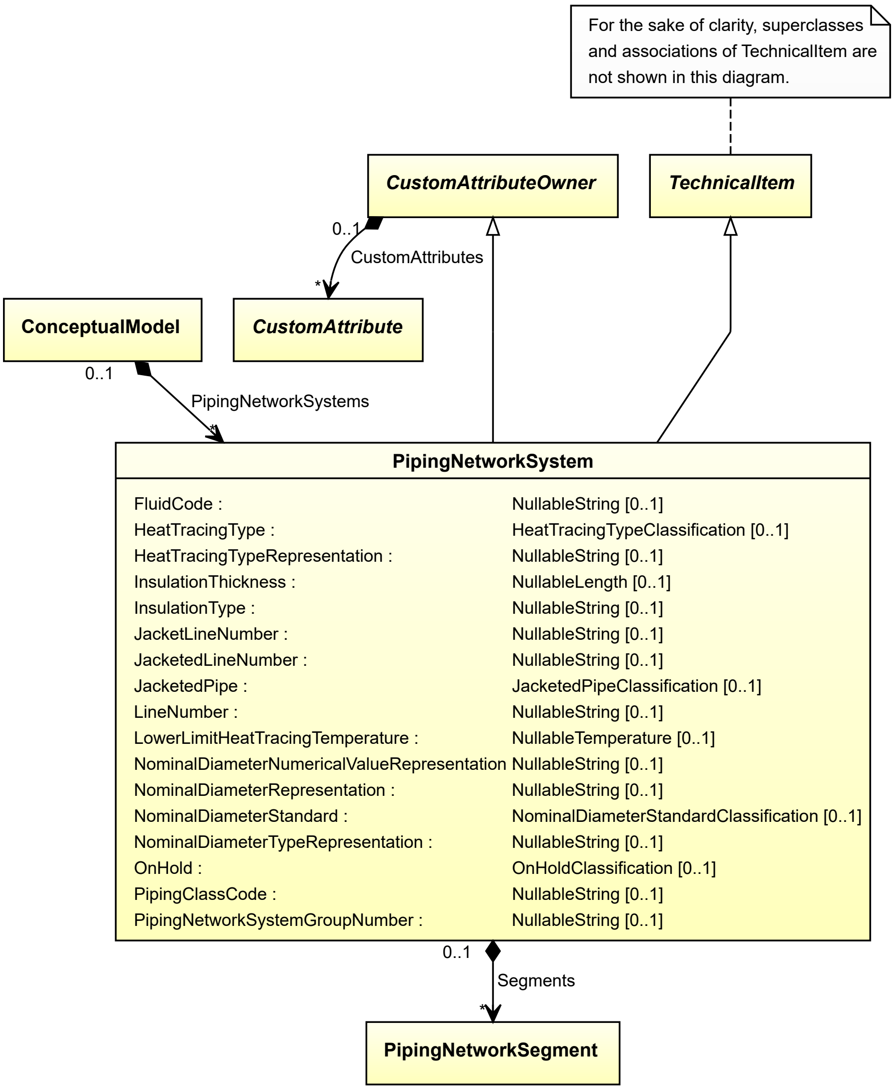
```
<PipingComponent
    ID="checkValve1"
    ComponentClass="CheckValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS292229" ...>
  ...
</PipingComponent>
```

8.56. PipingNetworkSystem

8.56.1 Overview

Class

A fluid system of interconnected piping network branches limited by Unit Operation Inlet/Outlet and Piping Network Terminators. In this context Piping includes e.g. plumbing and tubing (from http://data.posccaesar.org/rdl/RDS270359).


For the sake of clarity, superclasses and associations of TechnicalItem are not shown in this diagram.
CustomAttributeOwner
TechnicalItem
0..1
CustomAttributes
*
ConceptualModel
CustomAttribute
0..1
PipingNetworkSystems
*
PipingNetworkSystem
FluidCode : NullableString [0..1]
HeatTracingType : HeatTracingTypeClassification [0..1]
HeatTracingTypeRepresentation : NullableString [0..1]
InsulationThickness : NullableLength [0..1]
InsulationType : NullableString [0..1]
JacketLineNumber : NullableString [0..1]
JacketedLineNumber : NullableString [0..1]
JacketedPipe : JacketedPipeClassification [0..1]
LineNumber : NullableString [0..1]
LowerLimitHeatTracingTemperature : NullableTemperature [0..1]
NominalDiameterNumericalValueRepresentation NullableString [0..1]
NominalDiameterRepresentation : NullableString [0..1]
NominalDiameterStandard : NominalDiameterStandardClassification [0..1]
NominalDiameterTypeRepresentation : NullableString [0..1]
OnHold : OnHoldClassification [0..1]
PipingClassCode : NullableString [0..1]
PipingNetworkSystemGroupNumber : NullableString [0..1]
0..1
Segments
*
PipingNetworkSegment

Supertypes

- *CustomAttributeOwner*
- *TechnicalItem*

Attributes (data)

Name	Multiplicity	Type
FluidCode	0..1	*NullableString*
HeatTracingType	0..1	*HeatTracingTypeClassification*
HeatTracingTypeRepresentation	0..1	*NullableString*
InsulationThickness	0..1	*NullableLength*
InsulationType	0..1	*NullableString*
JacketLineNumber	0..1	*NullableString*
JacketedLineNumber	0..1	*NullableString*
JacketedPipe	0..1	*JacketedPipeClassification*
LineNumber	0..1	*NullableString*
LowerLimitHeatTracingTemperature	0..1	*NullableTemperature*

(continued on next page)

Name	Multiplicity	Type
NominalDiameterNumericalValueRepresentation	0..1	*NullableString*
NominalDiameterRepresentation	0..1	*NullableString*
NominalDiameterStandard	0..1	*NominalDiameterStandardClassification*
NominalDiameterTypeRepresentation	0..1	*NullableString*
OnHold	0..1	*OnHoldClassification*
PipingClassCode	0..1	*NullableString*
PipingNetworkSystemGroupNumber	0..1	*NullableString*

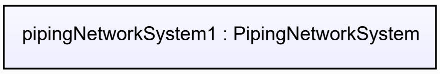
Attributes (composition)

Name	Multiplicity	Type
Segments	*	*PipingNetworkSegment*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingNetworkSystem>`
RDL reference: PIPING NETWORK SYSTEM
ComponentClass: PipingNetworkSystem
ComponentClassURI: http://data.posccaesar.org/rdl/RDS270359

Example



Example: Implementation in Proteus Schema

```
<PipingNetworkSystem
    ID="pipingNetworkSystem1"
    ComponentClass="PipingNetworkSystem"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS270359" ...>
  ...
</PipingNetworkSystem>
```

8.56.2 FluidCode

Attribute (data)

The identification code of the fluid related to the *PipingNetworkSystem*. So far, DEXPI does not define restrictions for valid values.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: FLUID CODE ASSIGNMENT CLASS
Name: FluidCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/FluidCodeAssignmentClass

Example

“MNb” (*String*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSystem
    ID="pipingNetworkSystem1"
    ComponentClass="PipingNetworkSystem"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS270359" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="FluidCodeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/FluidCodeAssignmentClass"
      Format="string"
      Value="MNb" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSystem>
```

8.56.3 HeatTracingType

Attribute (data)

A specialization indicating the heat tracing type related to the *PipingNetworkSystem*.

Multiplicity: 0..1

Type: *HeatTracingTypeClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: HEAT TRACING TYPE SPECIALIZATION
Name: HeatTracingTypeSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/HeatTracingTypeSpecialization

Example

electrical heat tracing system (*HeatTracingTypeClassification::ElectricalHeatTracingSystem*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSystem
    ID="pipingNetworkSystem1"
    ComponentClass="PipingNetworkSystem"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS270359" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="HeatTracingTypeSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/HeatTracingTypeSpecialization"
      Format="anyURI"
      Value="ElectricalHeatTracingSystem"
      ValueURI="http://data.posccaesar.org/rdl/RDS11854600" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSystem>
```

8.56.4 HeatTracingTypeRepresentation

Attribute (data)

The heat tracing type related to the *PipingNetworkSystem*, represented as a string.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: HEAT TRACING TYPE REPRESENTATION ASSIGNMENT CLASS
Name: HeatTracingTypeRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/HeatTracingTypeRepresentationAssignmentClass

Example

"E" (*String*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSystem
    ID="pipingNetworkSystem1"
    ComponentClass="PipingNetworkSystem"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS270359" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="HeatTracingTypeRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/HeatTracingTypeRepresentationAssignmentClass"
      Format="string"
      Value="E" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSystem>
```

8.56.5 InsulationThickness

Attribute (data)

The insulation thickness of the *PipingNetworkSystem*.

Multiplicity: 0..1

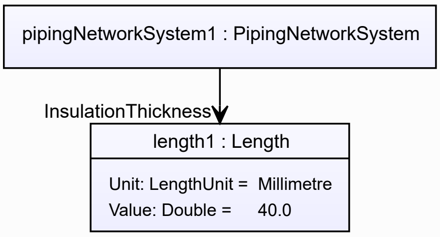
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: INSULATION THICKNESS
Name: InsulationThickness
AttributeURI: http://data.posccaesar.org/rdl/RDS4238040

Example

The instance pipingNetworkSystem1 represents a *PipingNetworkSystem* with an *InsulationThickness* of 40.0 mm.


pipingNetworkSystem1 : PipingNetworkSystem
InsulationThickness
length1 : Length
Unit: LengthUnit = Millimetre
Value: Double = 40.0

Example: Implementation in Proteus Schema

```
<PipingNetworkSystem
    ID="pipingNetworkSystem1"
    ComponentClass="PipingNetworkSystem"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS270359" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="InsulationThickness"
      AttributeURI="http://data.posccaesar.org/rdl/RDS4238040"
      Format="double"
      Value="40.0"
      Units="Millimetre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1357739" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSystem>
```

8.56.6 InsulationType

Attribute (data)

The identification code for the insulation type related to the *PipingNetworkSystem*. So far, DEXPI does not define restrictions for valid values.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: INSULATION TYPE ASSIGNMENT CLASS
Name: InsulationTypeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/InsulationTypeAssignmentClass

Example

“Q” (*String*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSystem
    ID="pipingNetworkSystem1"
    ComponentClass="PipingNetworkSystem"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS270359" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="InsulationTypeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/InsulationTypeAssignmentClass"
      Format="string"
      Value="Q" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSystem>
```

8.56.7 JacketLineNumber

Attribute (data)

The line number of the PipingNetworkSystem that is the jacket of this *PipingNetworkSystem*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: JACKET LINE NUMBER ASSIGNMENT CLASS
Name: JacketLineNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/JacketLineNumberAssignmentClass

Example

“47126J” (*String*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSystem
    ID="pipingNetworkSystem1"
    ComponentClass="PipingNetworkSystem"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS270359" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="JacketLineNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/JacketLineNumberAssignmentClass"
      Format="string"
      Value="47126J" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSystem>
```

8.56.8 JacketedLineNumber

Attribute (data)

The line number of the PipingNetworkSystem for which this *PipingNetworkSystem* is the jacket.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: JACKETED LINE NUMBER ASSIGNMENT CLASS
Name: JacketedLineNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/JacketedLineNumberAssignmentClass

Example

“47126” (*String*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSystem
    ID="pipingNetworkSystem1"
    ComponentClass="PipingNetworkSystem"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS270359" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="JacketedLineNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/JacketedLineNumberAssignmentClass"
      Format="string"
      Value="47126" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSystem>
```

8.56.9 JacketedPipe

Attribute (data)

A specialization indicating whether the *PipingNetworkSystem* is jacketed.

Multiplicity: 0..1

Type: *JacketedPipeClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: JACKETED PIPE SPECIALIZATION
Name: JacketedPipeSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/JacketedPipeSpecialization

Example

jacketed (*JacketedPipeClassification::JacketedPipe*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSystem
    ID="pipingNetworkSystem1"
    ComponentClass="PipingNetworkSystem"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS270359" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="JacketedPipeSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/JacketedPipeSpecialization"
      Format="anyURI"
      Value="JacketedPipe"
      ValueURI="http://sandbox.dexpi.org/rdl/JacketedPipe" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSystem>
```

8.56.10 LineNumber

Attribute (data)

The line number of a *PipingNetworkSystem*. Values are typically (but not necessarily) string representations of numbers.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: LINE NUMBER ASSIGNMENT CLASS
Name: LineNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/LineNumberAssignmentClass

Example

“47126” (*String*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSystem
    ID="pipingNetworkSystem1"
    ComponentClass="PipingNetworkSystem"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS270359" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="LineNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/LineNumberAssignmentClass"
      Format="string"
      Value="47126" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSystem>
```

8.56.11 LowerLimitHeatTracingTemperature

Attribute (data)

The lower limit for the temperature that a heat tracing system must ensure for the *PipingNetworkSystem*.

Multiplicity: 0..1

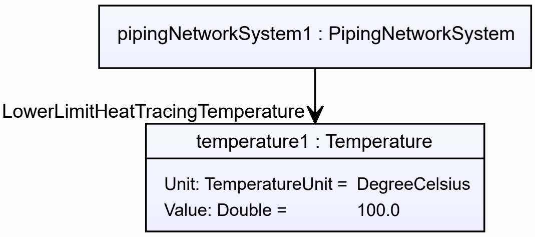
Type: *NullableTemperature*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: LOWER LIMIT HEAT TRACING TEMPERATURE
Name: LowerLimitHeatTracingTemperature
AttributeURI: http://sandbox.dexpi.org/rdl/LowerLimitHeatTracingTemperature

Example

The instance pipingNetworkSystem1 represents a *PipingNetworkSystem* with a *LowerLimitHeatTracingTemperature* of 100.0 °C.


pipingNetworkSystem1 : PipingNetworkSystem
LowerLimitHeatTracingTemperature
temperature1 : Temperature
Unit: TemperatureUnit = DegreeCelsius
Value: Double = 100.0

Example: Implementation in Proteus Schema

```
<PipingNetworkSystem
    ID="pipingNetworkSystem1"
    ComponentClass="PipingNetworkSystem"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS270359" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="LowerLimitHeatTracingTemperature"
      AttributeURI="http://sandbox.dexpi.org/rdl/LowerLimitHeatTracingTemperature"
      Format="double"
      Value="100.0"
      Units="DegreeCelsius"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1322684" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSystem>
```

8.56.12 NominalDiameterNumericalValueRepresentation

Attribute (data)

A readable representation of the numerical value of the nominal diameter of the *PipingNetworkSystem*, without any type or unit of measure.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: NOMINAL DIAMETER NUMERICAL VALUE REPRESENTATION ASSIGNMENT CLASS
Name: NominalDiameterNumericalValueRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/NominalDiameterNumericalValueRepresentationAssignmentClass

Example

“25” (*String*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSystem
    ID="pipingNetworkSystem1"
    ComponentClass="PipingNetworkSystem"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS270359" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NominalDiameterNumericalValueRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/NominalDiameterNumericalValueRepresentationAssignmentClass"
      Format="string"
      Value="25" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSystem>
```

8.56.13 NominalDiameterRepresentation

Attribute (data)

A readable representation of the nominal diameter of the *PipingNetworkSystem*. It normally contains a numerical value and a type or unit of measure.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: NOMINAL DIAMETER REPRESENTATION ASSIGNMENT CLASS
Name: NominalDiameterRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/NominalDiameterRepresentationAssignmentClass

Example

"DN 25" (*String*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSystem
    ID="pipingNetworkSystem1"
    ComponentClass="PipingNetworkSystem"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS270359" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NominalDiameterRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/NominalDiameterRepresentationAssignmentClass"
      Format="string"
      Value="DN 25" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSystem>
```

8.56.14 NominalDiameterStandard

Attribute (data)

The nominal diameter of the *PipingNetworkSystem*, given as a reference to a nominal diameter standard and value.

Multiplicity: 0..1

Type: *NominalDiameterStandardClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: NOMINAL DIAMETER STANDARD SPECIALIZATION
Name: NominalDiameterStandardSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/NominalDiameterStandardSpecialization

Example

DN 25 (DIN 2448) (*NominalDiameterStandardClassification::Din2448ObjectDn25*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSystem
    ID="pipingNetworkSystem1"
    ComponentClass="PipingNetworkSystem"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS270359" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NominalDiameterStandardSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/NominalDiameterStandardSpecialization"
      Format="anyURI"
      Value="Din2448ObjectDn25"
      ValueURI="http://sandbox.dexpi.org/rdl/Din2448ObjectDn25" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSystem>
```

8.56.15 NominalDiameterTypeRepresentation

Attribute (data)

A readable representation of the type or unit of measure of the nominal diameter of the *PipingNetworkSystem*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: NOMINAL DIAMETER TYPE REPRESENTATION ASSIGNMENT CLASS
Name: NominalDiameterTypeRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/NominalDiameterTypeRepresentationAssignmentClass

Example

“DN” (*String*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSystem
    ID="pipingNetworkSystem1"
    ComponentClass="PipingNetworkSystem"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS270359" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NominalDiameterTypeRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/NominalDiameterTypeRepresentationAssignmentClass"
      Format="string"
      Value="DN" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSystem>
```

8.56.16 OnHold

Attribute (data)

A specialization indicating if the *PipingNetworkSystem* is on hold or not.

Multiplicity: 0..1

Type: *OnHoldClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: ON HOLD SPECIALIZATION
Name: OnHoldSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/OnHoldSpecialization

Example

on hold (*OnHoldClassification::OnHold*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSystem
    ID="pipingNetworkSystem1"
    ComponentClass="PipingNetworkSystem"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS270359" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="OnHoldSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/OnHoldSpecialization"
      Format="anyURI"
      Value="OnHold"
      ValueURI="http://sandbox.dexpi.org/rdl/OnHold" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSystem>
```

8.56.17 PipingClassCode

Attribute (data)

The identification code of the piping class of the *PipingNetworkSystem*. So far, DEXPI does not define restrictions for valid values.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PIPING CLASS CODE ASSIGNMENT CLASS
Name: PipingClassCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PipingClassCodeAssignmentClass

Example

“75HB13” (*String*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSystem
    ID="pipingNetworkSystem1"
    ComponentClass="PipingNetworkSystem"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS270359" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PipingClassCodeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PipingClassCodeAssignmentClass"
      Format="string"
      Value="75HB13" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSystem>
```

8.56.18 PipingNetworkSystemGroupNumber

Attribute (data)

The number of the piping network system group of the *PipingNetworkSystem*, represented as a string.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PIPING NETWORK SYSTEM GROUP NUMBER ASSIGNMENT CLASS
Name: PipingNetworkSystemGroupNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PipingNetworkSystemGroupNumberAssignmentClass

Example

“G3” (*String*)

Example: Implementation in Proteus Schema

```
<PipingNetworkSystem
    ID="pipingNetworkSystem1"
    ComponentClass="PipingNetworkSystem"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS270359" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PipingNetworkSystemGroupNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PipingNetworkSystemGroupNumberAssignmentClass"
      Format="string"
      Value="G3" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSystem>
```

8.56.19 Segments

Attribute (composition)

The segments of the *PipingNetworkSystem*.

Multiplicity: *

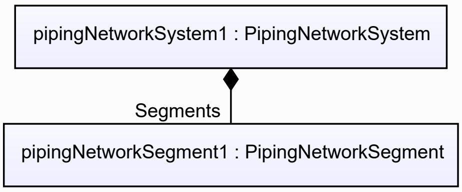
Type: *PipingNetworkSegment*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *PipingNetworkSegment*) is a child of the `<PipingNetworkSystem>` element for the attribute owner (a *PipingNetworkSystem*).

Example


pipingNetworkSystem1 : PipingNetworkSystem
Segments
pipingNetworkSegment1 : PipingNetworkSegment

Example: Implementation in Proteus Schema

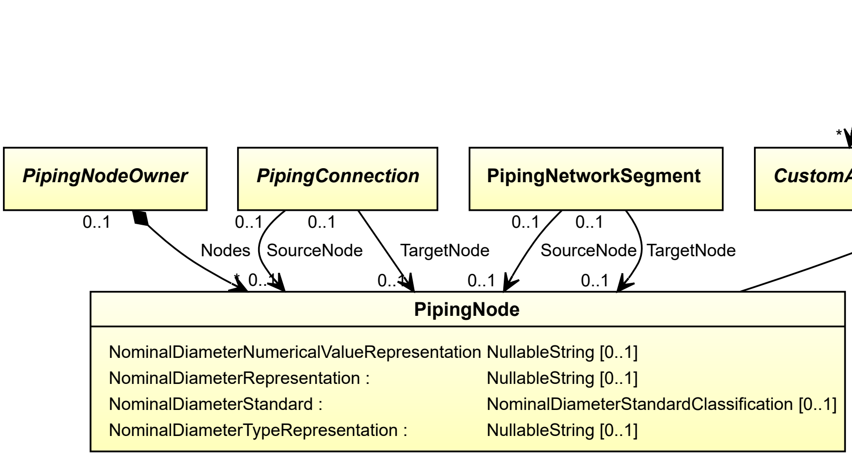
```
<PipingNetworkSystem
    ID="pipingNetworkSystem1"
    ComponentClass="PipingNetworkSystem"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS270359" ...>
  ...
  <PipingNetworkSegment
      ID="pipingNetworkSegment1"
      ComponentClass="PipingNetworkSegment"
      ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704" ...>
    ...
  <PipingNetworkSegment />
  ...
<PipingNetworkSystem />
```

8.57. PipingNode

8.57.1 Overview

Class

A possible connection point for a *PipingConnection*.


CustomAttributeOwner
0..1
CustomAttributes
*
PipingNodeOwner
PipingConnection
PipingNetworkSegment
CustomAttribute
0..1
0..1
0..1
0..1
0..1
Nodes
SourceNode
TargetNode
SourceNode
TargetNode
* 0..1
0..1
0..1
0..1
PipingNode
NominalDiameterNumericalValueRepresentation NullableString [0..1]
NominalDiameterRepresentation : NullableString [0..1]
NominalDiameterStandard : NominalDiameterStandardClassification [0..1]
NominalDiameterTypeRepresentation : NullableString [0..1]

Supertypes

- *CustomAttributeOwner*

Attributes (data)

Name	Multiplicity	Type
NominalDiameterNumericalValueRepresentation	0..1	*NullableString*
NominalDiameterRepresentation	0..1	*NullableString*
NominalDiameterStandard	0..1	*NominalDiameterStandardClassification*
NominalDiameterTypeRepresentation	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented using the Proteus element `<Node>`. The value of the Proteus XML attribute `Type` must be `"process"`.
See also the Proteus implementation of the *Nodes* attribute of *PipingNodeOwner*.

Example

Example: Implementation in Proteus Schema

```
<Node
    ID="pipingNode1"
    Type="process">
  ...
</Node>
```

8.57.2 NominalDiameterNumericalValueRepresentation

Attribute (data)

A readable representation of the numerical value of the nominal diameter of the *PipingNode*, without any type or unit of measure.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: NOMINAL DIAMETER NUMERICAL VALUE REPRESENTATION ASSIGNMENT CLASS
Name: NominalDiameterNumericalValueRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/NominalDiameterNumericalValueRepresentationAssignmentClass

Example

“25” (*String*)

Example: Implementation in Proteus Schema

```
<Node
    ID="pipingNode1"
    Type="process">
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NominalDiameterNumericalValueRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/NominalDiameterNumericalValueRepresentationAssignmentClass"
      Format="string"
      Value="25" />
    ...
  </GenericAttributes>
  ...
</Node>
```

8.57.3 NominalDiameterRepresentation

Attribute (data)

A readable representation of the nominal diameter of the *PipingNode*. It normally contains a numerical value and a type or unit of measure.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: NOMINAL DIAMETER REPRESENTATION ASSIGNMENT CLASS
Name: NominalDiameterRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/NominalDiameterRepresentationAssignmentClass

Example

“DN 25” (*String*)

Example: Implementation in Proteus Schema

```
<Node
    ID="pipingNode1"
    Type="process">
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NominalDiameterRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/NominalDiameterRepresentationAssignmentClass"
      Format="string"
      Value="DN 25" />
    ...
  </GenericAttributes>
  ...
</Node>
```

8.57.4 NominalDiameterStandard

Attribute (data)

The nominal diameter of the *PipingNode*, given as a reference to a nominal diameter standard and value.

Multiplicity: 0..1

Type: *NominalDiameterStandardClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: NOMINAL DIAMETER STANDARD SPECIALIZATION
Name: NominalDiameterStandardSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/NominalDiameterStandardSpecialization

Example

DN 25 (DIN 2448) (*NominalDiameterStandardClassification::Din2448ObjectDn25*)

Example: Implementation in Proteus Schema

```
<Node
    ID="pipingNode1"
    Type="process">
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NominalDiameterStandardSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/NominalDiameterStandardSpecialization"
      Format="anyURI"
      Value="Din2448ObjectDn25"
      ValueURI="http://sandbox.dexpi.org/rdl/Din2448ObjectDn25" />
    ...
  </GenericAttributes>
  ...
</Node>
```

8.57.5 NominalDiameterTypeRepresentation

Attribute (data)

A readable representation of the type or unit of measure of the nominal diameter of the *PipingNode*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: NOMINAL DIAMETER TYPE REPRESENTATION ASSIGNMENT CLASS
Name: NominalDiameterTypeRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/NominalDiameterTypeRepresentationAssignmentClass

Example

"DN" (*String*)

Example: Implementation in Proteus Schema

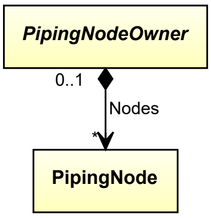
```
<Node
    ID="pipingNode1"
    Type="process">
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NominalDiameterTypeRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/NominalDiameterTypeRepresentationAssignmentClass"
      Format="string"
      Value="DN" />
    ...
  </GenericAttributes>
  ...
</Node>
```

8.58. PipingNodeOwner

8.58.1 Overview

Abstract class

An object that can have *PipingNodes*.



Subtypes

- *Nozzle*
- *PipeOffPageConnector*
- *PipingComponent*
- *PropertyBreak*

Attributes (composition)

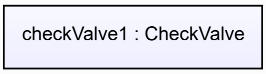
Name	Multiplicity	Type
Nodes	*	*PipingNode*

Implementation in Proteus Schema

Implementation is subclass-specific.

Example

As *PipingNodeOwner* is abstract, we consider *CheckValve* as an arbitrary concrete subclass.



Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="checkValve1"
    ComponentClass="CheckValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS292229" ...>
  ...
</PipingComponent>
```

8.58.2 Nodes

Attribute (composition)

The *PipingNodes* of the *PipingNodeOwner*.

Multiplicity: *

Type: *PipingNode*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the XML hierarchy of the Proteus file. However, the `<Node>` elements for *PipingNodes* are not children of the XML element for the *PipingNodeOwner* itself; they are rather grouped in a single `<ConnectionPoints>` element that is placed in the XML element for the *PipingNodeOwner*.
Note that the first `<Node>` element in the `<ConnectionPoints>` cannot represent a *PipingNode*. In Proteus Schema, the first `<Node>` element corresponds to the *PipingNodeOwner* itself. It is not relevant for DEXPI, but it must be present according to the Proteus Schema specification. It must not have a `Type` attribute.
The further `<Node>` elements with `Type="process"` (cf. Proteus implementation of *PipingNode*) represent the

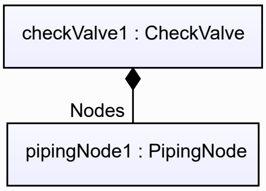
Nodes of the *PipingNodeOwner*.
The following XML fragment shows the case of a *PipingNodeOwner* (more specifically, a *PipingComponent*) that has one *PipingNode*:

```
<PipingComponent ...>
  ...
  <ConnectionPoints NumPoints="4" ...>
    <!-- first Node is never relevant for DEXPI -->
    <Node ID="nonRelevantNode">...</Node>

    <!-- this is a PipingNode because Type is "process" -->
    <Node ID="aPipingNode" Type="process">...</Node>

    <!-- this is not a PipingNode because no Type is given -->
    <Node ID="someNode">...</Node>

    <!-- this is not a PipingNode because Type is not "process" -->
    <Node ID="someSignalNode" Type="signal">...</Node>
  </ConnectionPoints>
  ...
</PipingComponent>
```

In some cases, Proteus Schema uses the *index* of a `<Node>` element to refer to that `<Node>`. The index is the zero-based number of the `<Node>` in the `<ConnectionPoints>`. For example, in the XML fragment above, the index of the `<Node>` with `ID="nonRelevantNode"` is 0 and the index of the `<Node>` with `ID="aPipingNode"` is 1. These indices are an implementation detail of Proteus Schema, and they do not carry any additional semantics. The *Nodes* attribute of *PipingNodeOwner* is *not ordered*, i.e., a *PipingNodeOwner* in DEXPI does not have a *first* or *second PipingNode*.

Example

As the owner type *PipingNodeOwner* is abstract, we consider *CheckValve* as an arbitrary concrete subclass.



Example: Implementation in Proteus Schema

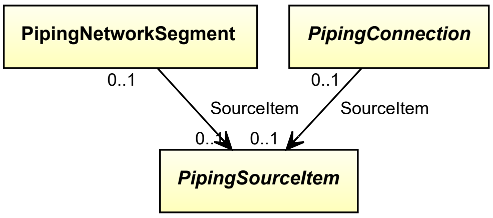
```
<PipingComponent
    ID="checkValve1"
    ComponentClass="CheckValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS292229" ...>
  ...
  <ConnectionPoints ...>
    <Node ID="nonRelevantNode" />
    <Node ID="pipingNode1" Type="process">...</Node>
    ...
  </ConnectionPoints>
  ...
</PipingComponent>
```

8.59. PipingSourceItem

8.59.1 Overview

Abstract class

An item that can be the source of a *PipingConnection* (attribute *SourceItem*) or a *PipingNetworkSegment* (attribute *SourceItem*).



Subtypes

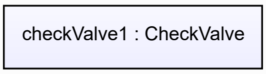
- *FlowInPipeOffPageConnector*
- *Nozzle*
- *PipingComponent*
- *PropertyBreak*

Implementation in Proteus Schema

Implementation is subclass-specific.

Example

As *PipingSourceItem* is abstract, we consider *CheckValve* as an arbitrary concrete subclass.



Example: Implementation in Proteus Schema

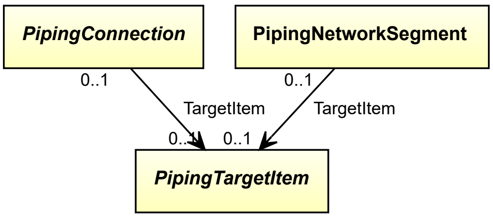
```
<PipingComponent
    ID="checkValve1"
    ComponentClass="CheckValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS292229" ...>
  ...
</PipingComponent>
```

8.60. PipingTargetItem

8.60.1 Overview

Abstract class

An item that can be the target of a *PipingConnection* (attribute *TargetItem*) or a *PipingNetworkSegment* (attribute *TargetItem*).



Subtypes

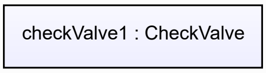
- *FlowOutPipeOffPageConnector*
- *Nozzle*
- *PipingComponent*
- *PropertyBreak*

Implementation in Proteus Schema

Implementation is subclass-specific.

Example

As *PipingTargetItem* is abstract, we consider *CheckValve* as an arbitrary concrete subclass.



Example: Implementation in Proteus Schema

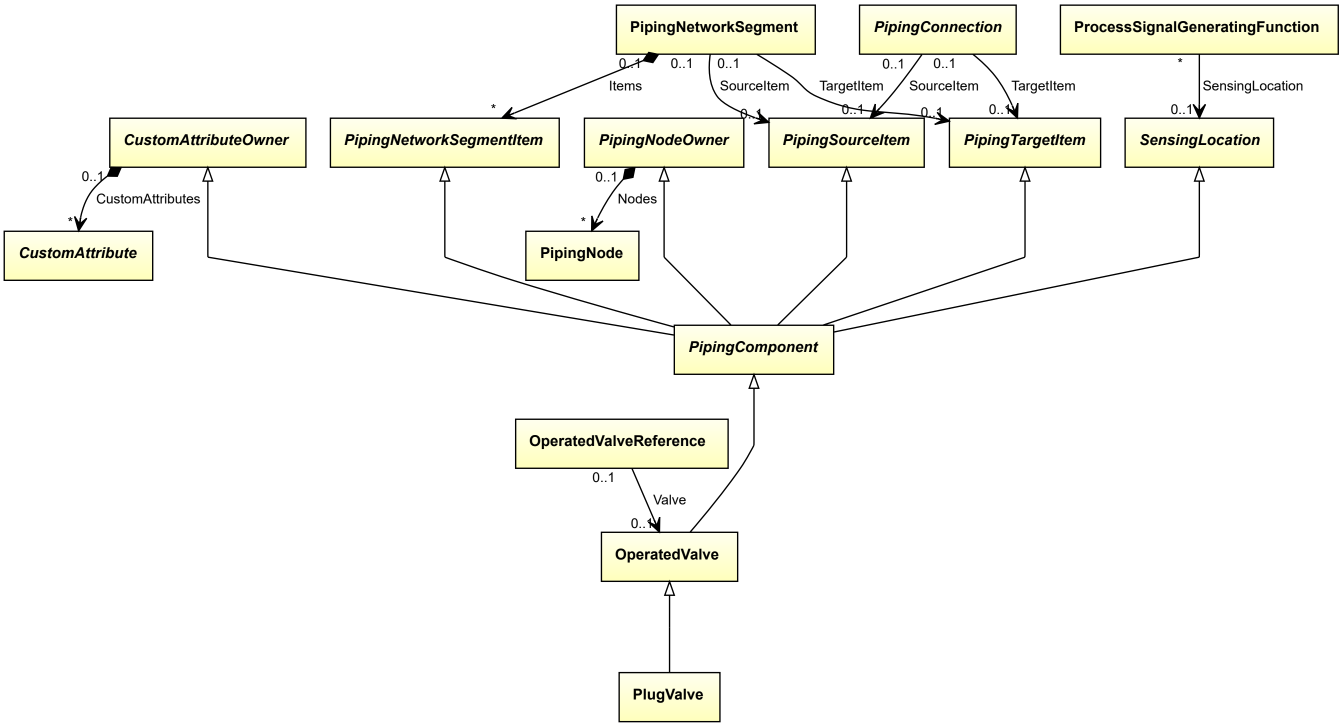
```
<PipingComponent
    ID="checkValve1"
    ComponentClass="CheckValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS292229" ...>
  ...
</PipingComponent>
```

8.61. PlugValve

8.61.1 Overview

Class

A rotary valve that has a quarter turn action in which the closure member is a cylindrical or tapered plug which operates by rotating on its axis and sealing against a downstream seat (from http://data.posccaesar.org/rdl/RDS421109).



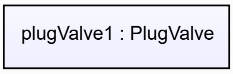
Supertypes

- *OperatedValve*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <PipingComponent>
RDL reference: PLUG VALVE
ComponentClass: PlugValve
ComponentClassURI: http://data.posccaesar.org/rdl/RDS421109

Example



Example: Implementation in Proteus Schema

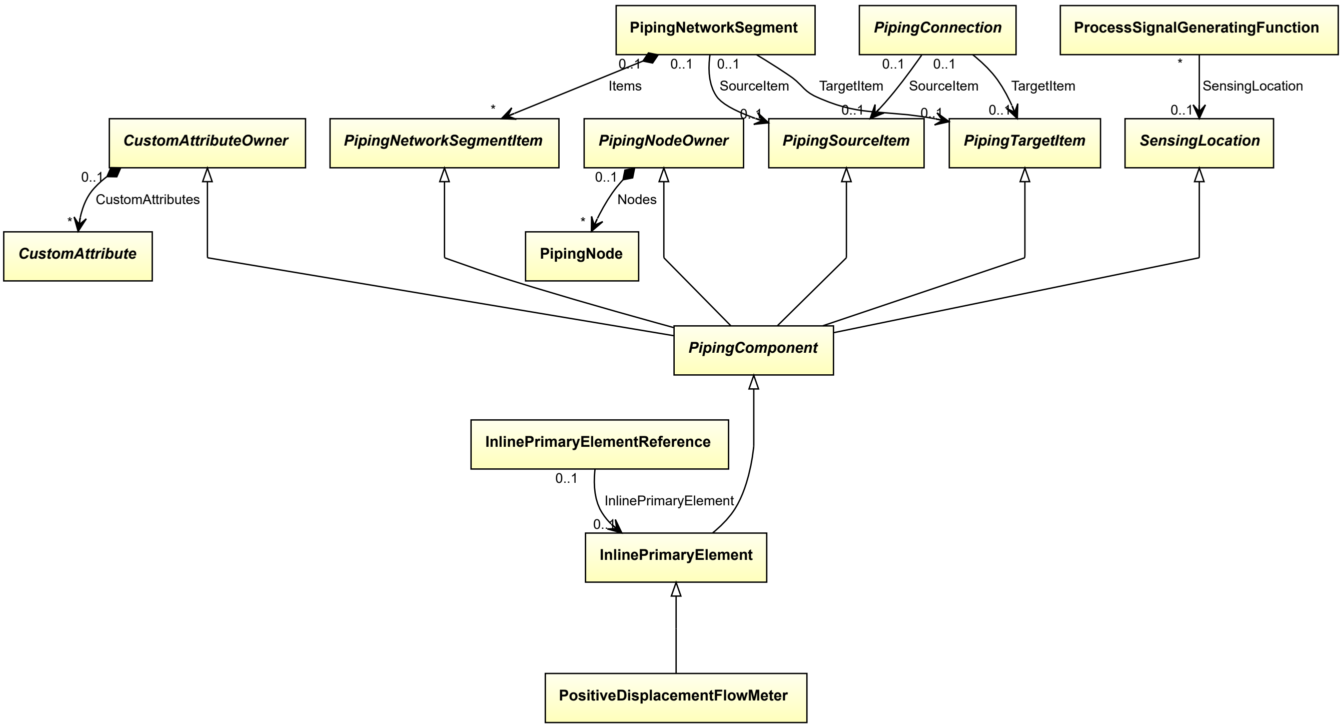
```
<PipingComponent
    ID="plugValve1"
    ComponentClass="PlugValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS421109" ...>
  ...
</PipingComponent>
```

8.62. PositiveDisplacementFlowMeter

8.62.1 Overview

Class

A flow meter that measures the volumetric flow rate of a liquid or gas by separating the flow stream into known volumes and counting them over time (from http://data.posccaesar.org/rdl/RDS418094).



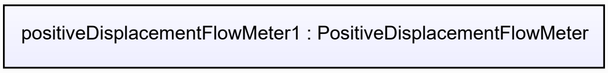
Supertypes

- *InlinePrimaryElement*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingComponent>`
RDL reference: POSITIVE DISPLACEMENT FLOW METER
ComponentClass: PositiveDisplacementFlowMeter
ComponentClassURI: http://data.posccaesar.org/rdl/RDS418094

Example


positiveDisplacementFlowMeter1 : PositiveDisplacementFlowMeter

Example: Implementation in Proteus Schema

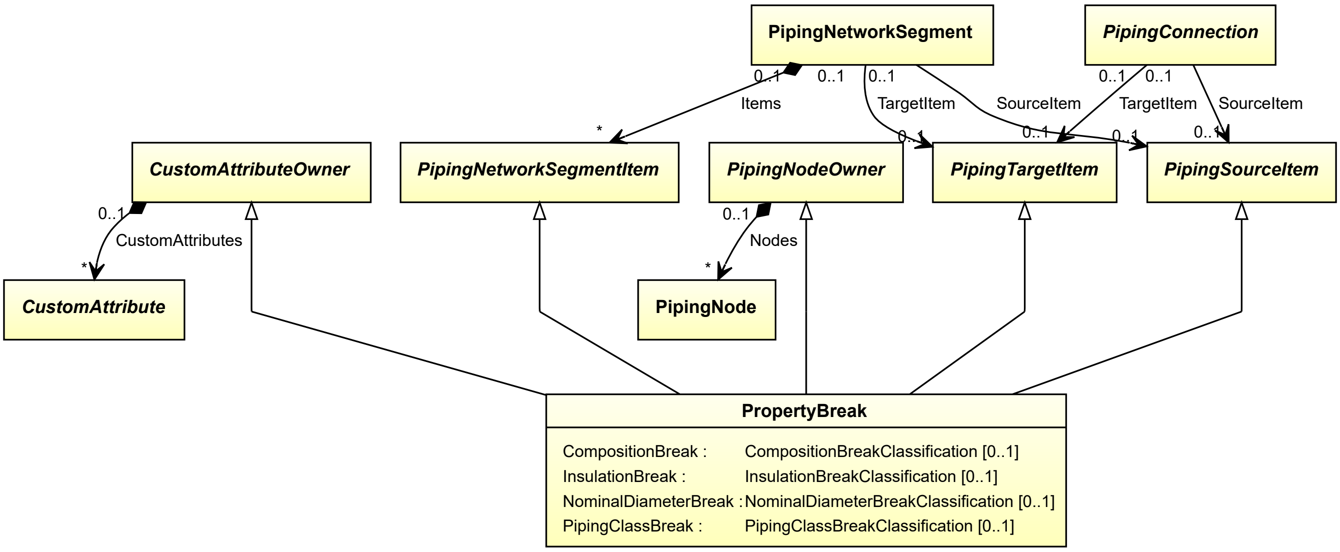
```
<PipingComponent
    ID="positiveDisplacementFlowMeter1"
    ComponentClass="PositiveDisplacementFlowMeter"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS418094" ...>
  ...
</PipingComponent>
```

8.63. PropertyBreak

8.63.1 Overview

Class

A symbol indicating a change in the piping properties.



Supertypes

- *CustomAttributeOwner*
- *PipingNetworkSegmentItem*
- *PipingNodeOwner*
- *PipingSourceItem*
- *PipingTargetItem*

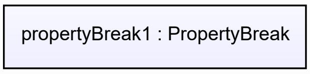
Attributes (data)

Name	Multiplicity	Type
CompositionBreak	0..1	*CompositionBreakClassification*
InsulationBreak	0..1	*InsulationBreakClassification*
NominalDiameterBreak	0..1	*NominalDiameterBreakClassification*
PipingClassBreak	0..1	*PipingClassBreakClassification*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PropertyBreak>`
RDL reference: PROPERTY BREAK
ComponentClass: PropertyBreak
ComponentClassURI: http://sandbox.dexpi.org/rdl/PropertyBreak

Example



Example: Implementation in Proteus Schema

```
<PropertyBreak
    ID="propertyBreak1"
    ComponentClass="PropertyBreak"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PropertyBreak" ...>
  ...
</PropertyBreak>
```

8.63.2 CompositionBreak

Attribute (data)

A specialization indicating if the *PropertyBreak* is a composition break or not.

Multiplicity: 0..1

Type: *CompositionBreakClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: COMPOSITION BREAK SPECIALIZATION
Name: CompositionBreakSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/CompositionBreakSpecialization

Example

no composition break (*CompositionBreakClassification::NoCompositionBreak*)

Example: Implementation in Proteus Schema

```
<PropertyBreak
    ID="propertyBreak1"
    ComponentClass="PropertyBreak"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PropertyBreak" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="CompositionBreakSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/CompositionBreakSpecialization"
      Format="anyURI"
      Value="NoCompositionBreak"
      ValueURI="http://sandbox.dexpi.org/rdl/NoCompositionBreak" />
    ...
  </GenericAttributes>
  ...
</PropertyBreak>
```

8.63.3 InsulationBreak

Attribute (data)

A specialization indicating if the *PropertyBreak* is an insulation break or not.

Multiplicity: 0..1

Type: *InsulationBreakClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: INSULATION BREAK SPECIALIZATION
Name: InsulationBreakSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/InsulationBreakSpecialization

Example

inulation break (*InsulationBreakClassification::InsulationBreak*)

Example: Implementation in Proteus Schema

```
<PropertyBreak
    ID="propertyBreak1"
    ComponentClass="PropertyBreak"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PropertyBreak" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="InsulationBreakSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/InsulationBreakSpecialization"
      Format="anyURI"
      Value="InsulationBreak"
      ValueURI="http://sandbox.dexpi.org/rdl/InsulationBreak" />
    ...
  </GenericAttributes>
  ...
</PropertyBreak>
```

8.63.4 NominalDiameterBreak

Attribute (data)

A specialization indicating if the *PropertyBreak* is a nominal diameter break or not.

Multiplicity: 0..1

Type: *NominalDiameterBreakClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: NOMINAL DIAMETER BREAK SPECIALIZATION
Name: NominalDiameterBreakSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/NominalDiameterBreakSpecialization

Example

no nominal diameter break (*NominalDiameterBreakClassification::NoNominalDiameterBreak*)

Example: Implementation in Proteus Schema

```
<PropertyBreak
    ID="propertyBreak1"
    ComponentClass="PropertyBreak"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PropertyBreak" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="NominalDiameterBreakSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/NominalDiameterBreakSpecialization"
      Format="anyURI"
      Value="NoNominalDiameterBreak"
      ValueURI="http://sandbox.dexpi.org/rdl/NoNominalDiameterBreak" />
    ...
  </GenericAttributes>
  ...
</PropertyBreak>
```

8.63.5 PipingClassBreak

Attribute (data)

A specialization indicating if the *PropertyBreak* is a composition break or not.

Multiplicity: 0..1

Type: *PipingClassBreakClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: PIPING CLASS BREAK SPECIALIZATION
Name: PipingClassBreakSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/PipingClassBreakSpecialization

Example

piping class break (*PipingClassBreakClassification::PipingClassBreak*)

Example: Implementation in Proteus Schema

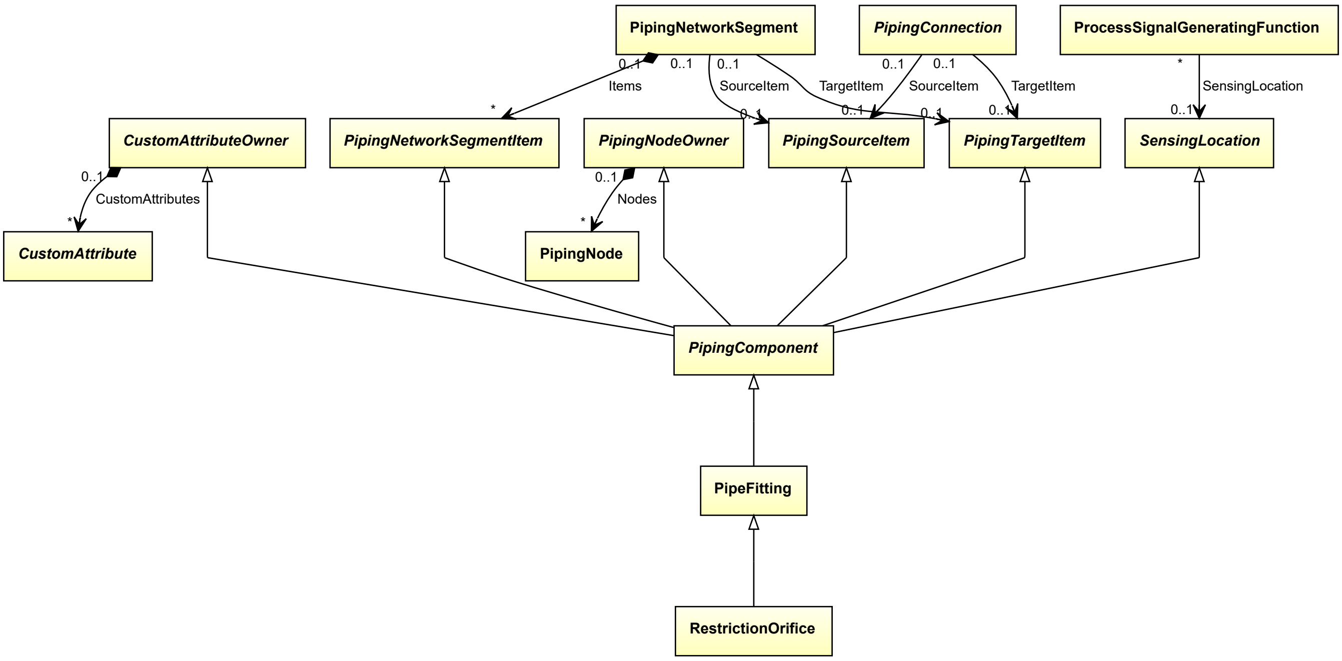
```
<PropertyBreak
    ID="propertyBreak1"
    ComponentClass="PropertyBreak"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PropertyBreak" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PipingClassBreakSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/PipingClassBreakSpecialization"
      Format="anyURI"
      Value="PipingClassBreak"
      ValueURI="http://sandbox.dexpi.org/rdl/PipingClassBreak" />
    ...
  </GenericAttributes>
  ...
</PropertyBreak>
```

8.64. RestrictionOrifice

8.64.1 Overview

Class

A RESTRICTION ORIFICE is an ORIFICE PLATE that is intended for use as a restrictor.



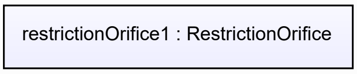
Supertypes

- *PipeFitting*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingComponent>`
RDL reference: RESTRICTION ORIFICE
ComponentClass: RestrictionOrifice
ComponentClassURI: http://sandbox.dexpi.org/rdl/RestrictionOrifice

Example



Example: Implementation in Proteus Schema

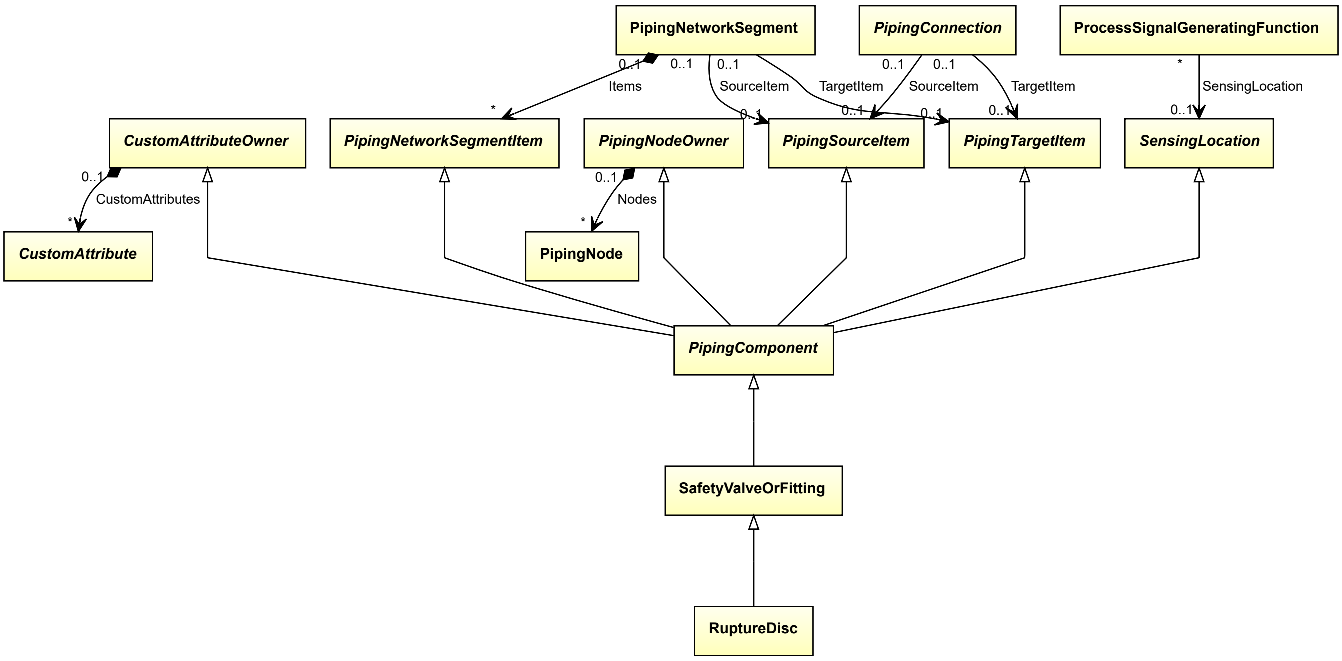
```
<PipingComponent
    ID="restrictionOrifice1"
    ComponentClass="RestrictionOrifice"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/RestrictionOrifice" ...>
  ...
</PipingComponent>
```

8.65. RuptureDisc

8.65.1 Overview

Class

A physical object that is designed to burst at a certain excess pressure. It is part of a rupture disc assembly (from http://data.posccaesar.org/rdl/RDS8372601).



Supertypes

- *SafetyValveOrFitting*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <PipingComponent>
RDL reference: RUPTURE DISC
ComponentClass: RuptureDisc
ComponentClassURI: http://data.posccaesar.org/rdl/RDS8372601

Example

Example: Implementation in Proteus Schema

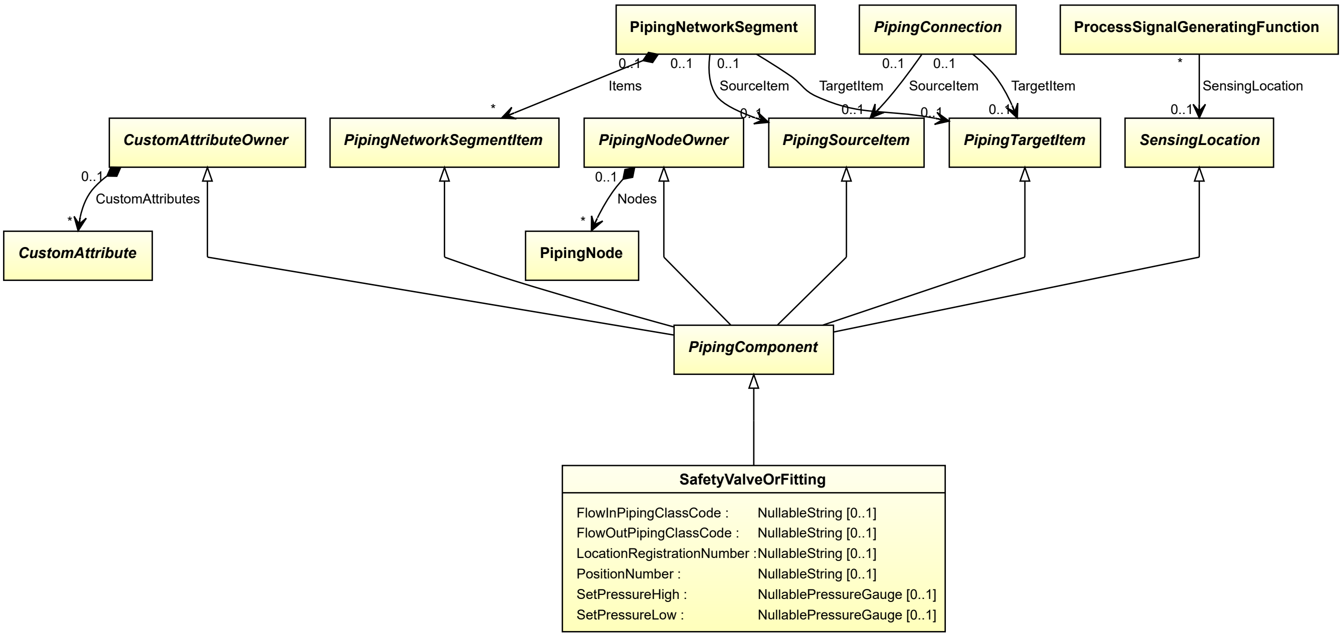
```
<PipingComponent
    ID="ruptureDisc1"
    ComponentClass="RuptureDisc"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS8372601" ...>
  ...
</PipingComponent>
```

8.66. SafetyValveOrFitting

8.66.1 Overview

Class

A safety valve or fitting.



Supertypes

- *PipingComponent*

Subtypes

- *BreatherValve*
- *CustomSafetyValveOrFitting*
- *FlameArrestor*
- *RuptureDisc*
- *SpringLoadedAngleGlobeSafetyValve*
- *SpringLoadedGlobeSafetyValve*

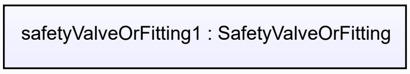
Attributes (data)

Name	Multiplicity	Type
FlowInPipingClassCode	0..1	*NullableString*
FlowOutPipingClassCode	0..1	*NullableString*
LocationRegistrationNumber	0..1	*NullableString*
PositionNumber	0..1	*NullableString*
SetPressureHigh	0..1	*NullablePressureGauge*
SetPressureLow	0..1	*NullablePressureGauge*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <PipingComponent>
RDL reference: SAFETY VALVE OR FITTING
ComponentClass: SafetyValveOrFitting
ComponentClassURI: http://sandbox.dexpi.org/rdl/SafetyValveOrFitting

Example



Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="safetyValveOrFitting1"
    ComponentClass="SafetyValveOrFitting"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SafetyValveOrFitting" ...>
  ...
</PipingComponent>
```

8.66.2 FlowInPipingClassCode

Attribute (data)

The code of the piping class at the flow in side of *SafetyValveOrFitting*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: FLOW IN PIPING CLASS CODE ASSIGNMENT CLASS
Name: FlowInPipingClassCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/FlowInPipingClassCodeAssignmentClass

Example

“75HB13” (*String*)

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="safetyValveOrFitting1"
    ComponentClass="SafetyValveOrFitting"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SafetyValveOrFitting" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="FlowInPipingClassCodeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/FlowInPipingClassCodeAssignmentClass"
      Format="string"
      Value="75HB13" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.66.3 FlowOutPipingClassCode

Attribute (data)

The code of the piping class at the flow out side of *SafetyValveOrFitting*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: FLOW OUT PIPING CLASS CODE ASSIGNMENT CLASS
Name: FlowOutPipingClassCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/FlowOutPipingClassCodeAssignmentClass

Example

“75HB13” (*String*)

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="safetyValveOrFitting1"
    ComponentClass="SafetyValveOrFitting"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SafetyValveOrFitting" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="FlowOutPipingClassCodeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/FlowOutPipingClassCodeAssignmentClass"
      Format="string"
      Value="75HB13" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.66.4 LocationRegistrationNumber

Attribute (data)

The location registration number of the *SafetyValveOrFitting*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: LOCATION REGISTRATION NUMBER ASSIGNMENT CLASS
Name: LocationRegistrationNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/LocationRegistrationNumberAssignmentClass

Example

"L-N123" (*String*)

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="safetyValveOrFitting1"
    ComponentClass="SafetyValveOrFitting"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SafetyValveOrFitting" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="LocationRegistrationNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/LocationRegistrationNumberAssignmentClass"
      Format="string"
      Value="L-N123" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.66.5 PositionNumber

Attribute (data)

The position number of the *SafetyValveOrFitting*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: POSITION NUMBER ASSIGNMENT CLASS
Name: PositionNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PositionNumberAssignmentClass

Example

“SV 104.01” (*String*)

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="safetyValveOrFitting1"
    ComponentClass="SafetyValveOrFitting"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SafetyValveOrFitting" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PositionNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PositionNumberAssignmentClass"
      Format="string"
      Value="SV 104.01" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.66.6 SetPressureHigh

Attribute (data)

The high pressure at which the *SafetyValveOrFitting* is activated.

Multiplicity: 0..1

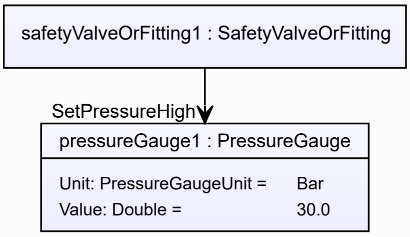
Type: *NullablePressureGauge*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: SET PRESSURE HIGH
Name: SetPressureHigh
AttributeURI: http://sandbox.dexpi.org/rdl/SetPressureHigh

Example

The instance safetyValveOrFitting1 represents a *SafetyValveOrFitting* with a *SetPressureHigh* of 30.0 bar.


safetyValveOrFitting1 : SafetyValveOrFitting
SetPressureHigh
pressureGauge1 : PressureGauge
Unit: PressureGaugeUnit = Bar
Value: Double = 30.0

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="safetyValveOrFitting1"
    ComponentClass="SafetyValveOrFitting"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SafetyValveOrFitting" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="SetPressureHigh"
      AttributeURI="http://sandbox.dexpi.org/rdl/SetPressureHigh"
      Format="double"
      Value="30.0"
      Units="Bar"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1314539" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.66.7 SetPressureLow

Attribute (data)

The low pressure at which the *SafetyValveOrFitting* is activated.

Multiplicity: 0..1

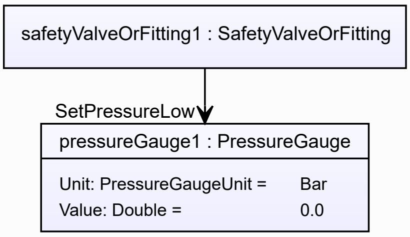
Type: *NullablePressureGauge*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: SET PRESSURE LOW
Name: SetPressureLow
AttributeURI: http://sandbox.dexpi.org/rdl/SetPressureLow

Example

The instance safetyValveOrFitting1 represents a *SafetyValveOrFitting* with a *SetPressureLow* of 0.0 bar.


safetyValveOrFitting1 : SafetyValveOrFitting
SetPressureLow
pressureGauge1 : PressureGauge
Unit: PressureGaugeUnit = Bar
Value: Double = 0.0

Example: Implementation in Proteus Schema

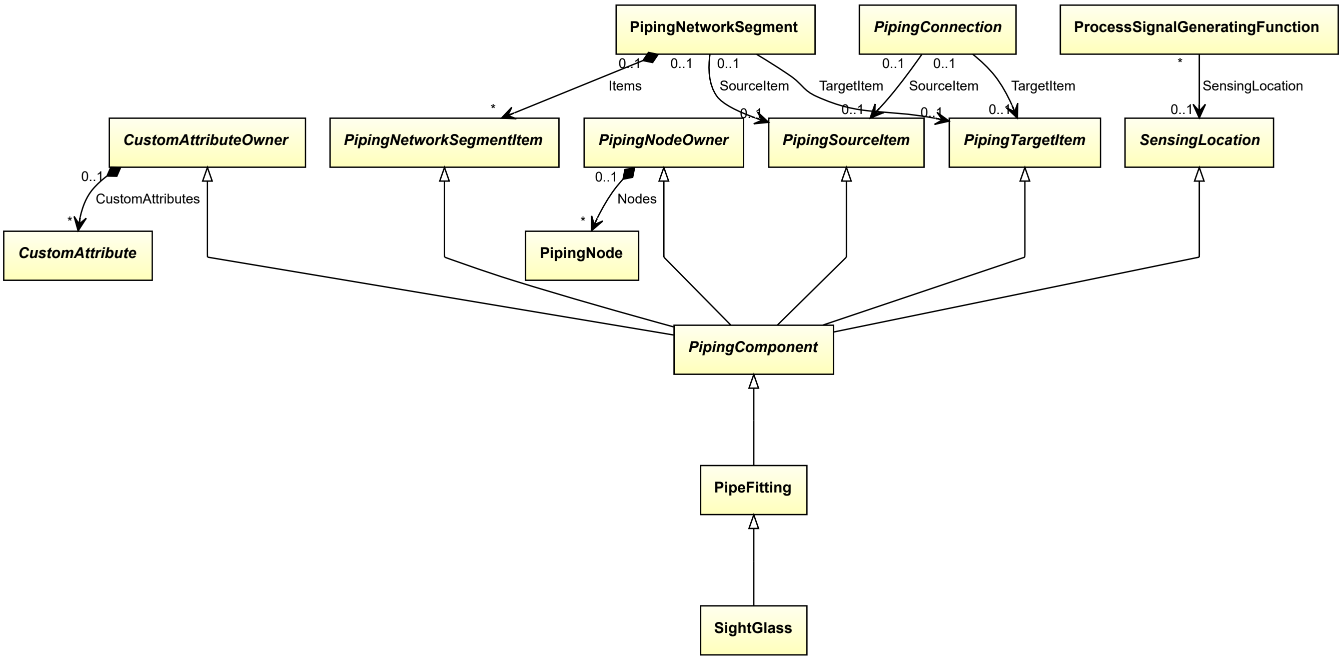
```
<PipingComponent
    ID="safetyValveOrFitting1"
    ComponentClass="SafetyValveOrFitting"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SafetyValveOrFitting" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="SetPressureLow"
      AttributeURI="http://sandbox.dexpi.org/rdl/SetPressureLow"
      Format="double"
      Value="0.0"
      Units="Bar"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1314539" />
    ...
  </GenericAttributes>
  ...
</PipingComponent>
```

8.67. SightGlass

8.67.1 Overview

Class

A physical object that is transparent and intended for viewing a vessel or piping system interior (from http://data.posccaesar.org/rdl/RDS648674).



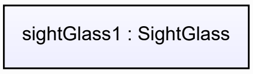
Supertypes

- *PipeFitting*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <PipingComponent>
RDL reference: SIGHT GLASS
ComponentClass: SightGlass
ComponentClassURI: http://data.posccaesar.org/rdl/RDS648674

Example



Example: Implementation in Proteus Schema

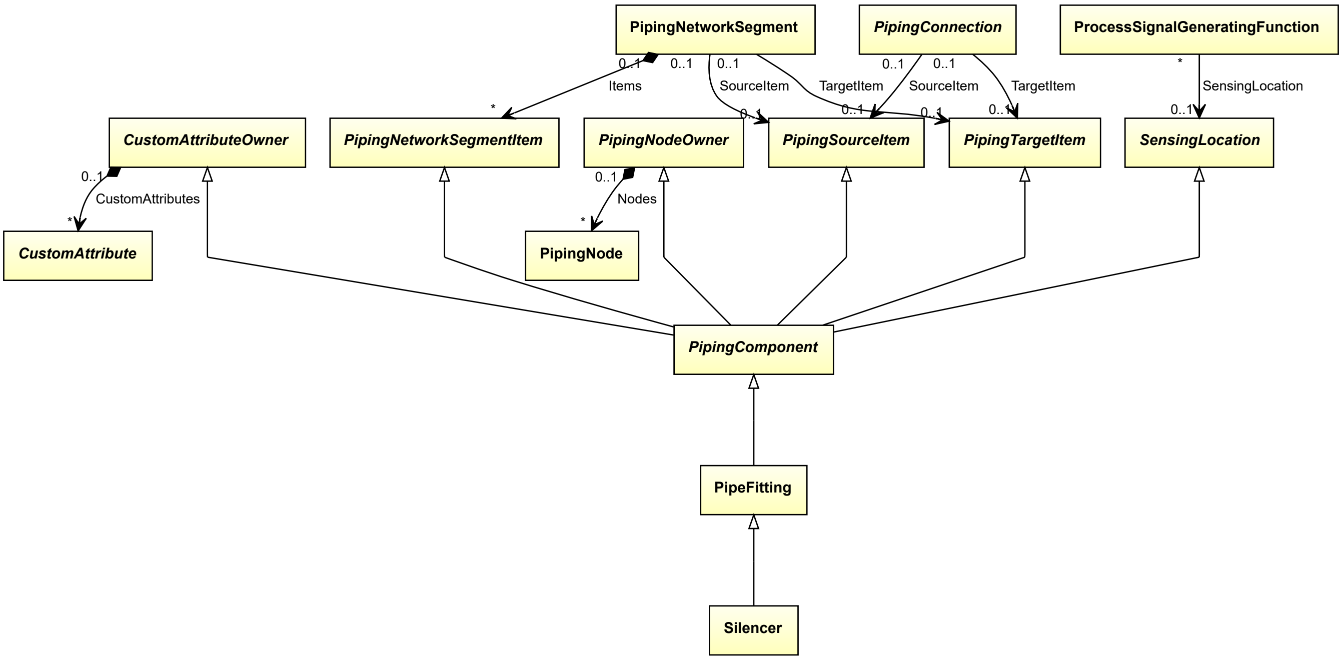
```
<PipingComponent
    ID="sightGlass1"
    ComponentClass="SightGlass"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS648674" ...>
  ...
</PipingComponent>
```

8.68. Silencer

8.68.1 Overview

Class

A device intended to reduce a noise level (from http://data.posccaesar.org/rdl/RDS1049368591).



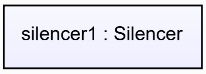
Supertypes

- *PipeFitting*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingComponent>`
RDL reference: SILENCER
ComponentClass: Silencer
ComponentClassURI: http://data.posccaesar.org/rdl/RDS1049368591

Example



Example: Implementation in Proteus Schema

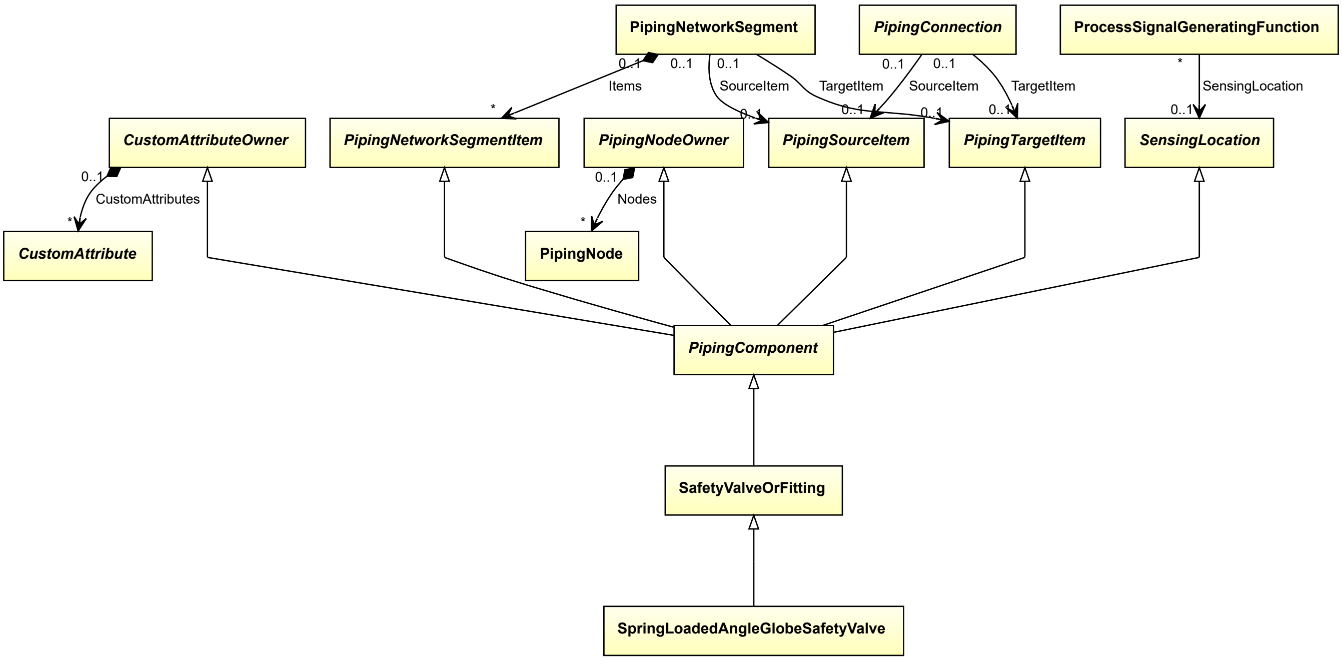
```
<PipingComponent
    ID="silencer1"
    ComponentClass="Silencer"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS1049368591" ...>
  ...
</PipingComponent>
```

8.69. SpringLoadedAngleGlobeSafetyValve

8.69.1 Overview

Class

A spring-loaded angle globe safety valve.



Supertypes

- *SafetyValveOrFitting*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <PipingComponent>
RDL reference: SPRING LOADED ANGLE GLOBE SAFETY VALVE
ComponentClass: SpringLoadedAngleGlobeSafetyValve
ComponentClassURI: http://sandbox.dexpi.org/rdl/SpringLoadedAngleGlobeSafetyValve

Example

springLoadedAngleGlobeSafetyValve1 : SpringLoadedAngleGlobeSafetyValve

Example: Implementation in Proteus Schema

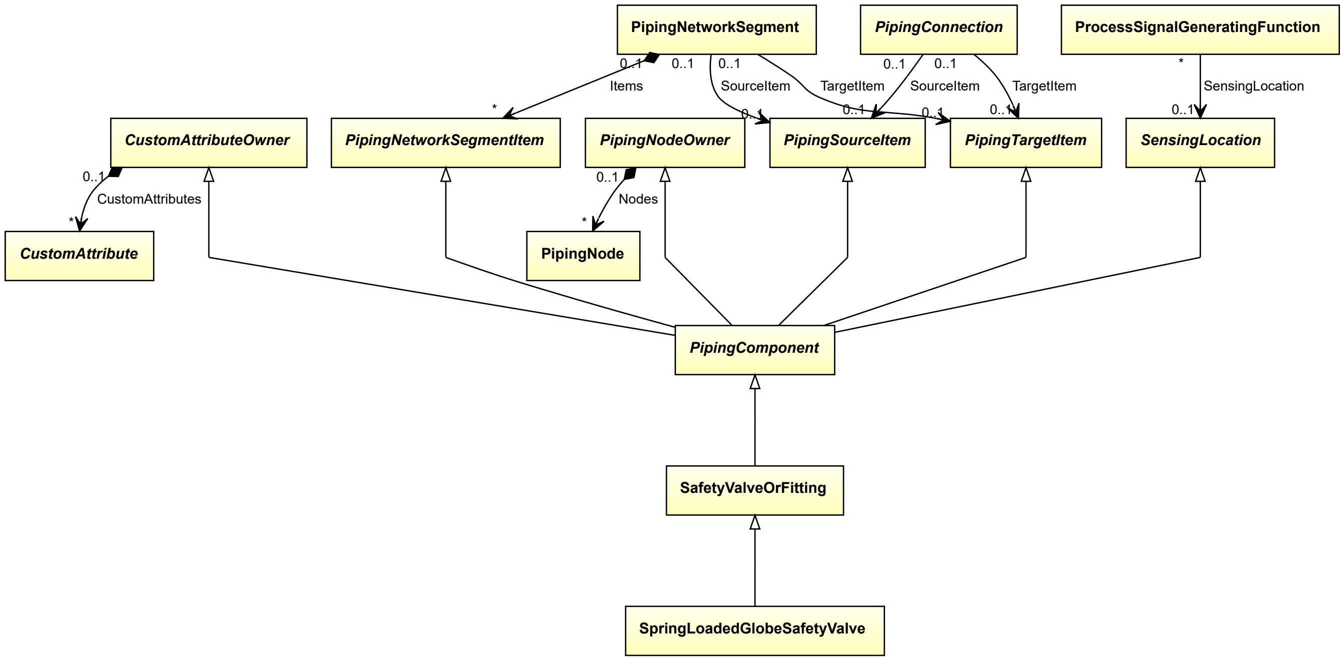
```
<PipingComponent
    ID="springLoadedAngleGlobeSafetyValve1"
    ComponentClass="SpringLoadedAngleGlobeSafetyValve"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SpringLoadedAngleGlobeSafetyValve" ...>
  ...
</PipingComponent>
```

8.70. SpringLoadedGlobeSafetyValve

8.70.1 Overview

Class

A spring-loaded globe safety valve.



Supertypes

- *SafetyValveOrFitting*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingComponent>`
RDL reference: SPRING LOADED GLOBE SAFETY VALVE
ComponentClass: SpringLoadedGlobeSafetyValve
ComponentClassURI: http://sandbox.dexpi.org/rdl/SpringLoadedGlobeSafetyValve

Example

springLoadedGlobeSafetyValve1 : SpringLoadedGlobeSafetyValve

Example: Implementation in Proteus Schema

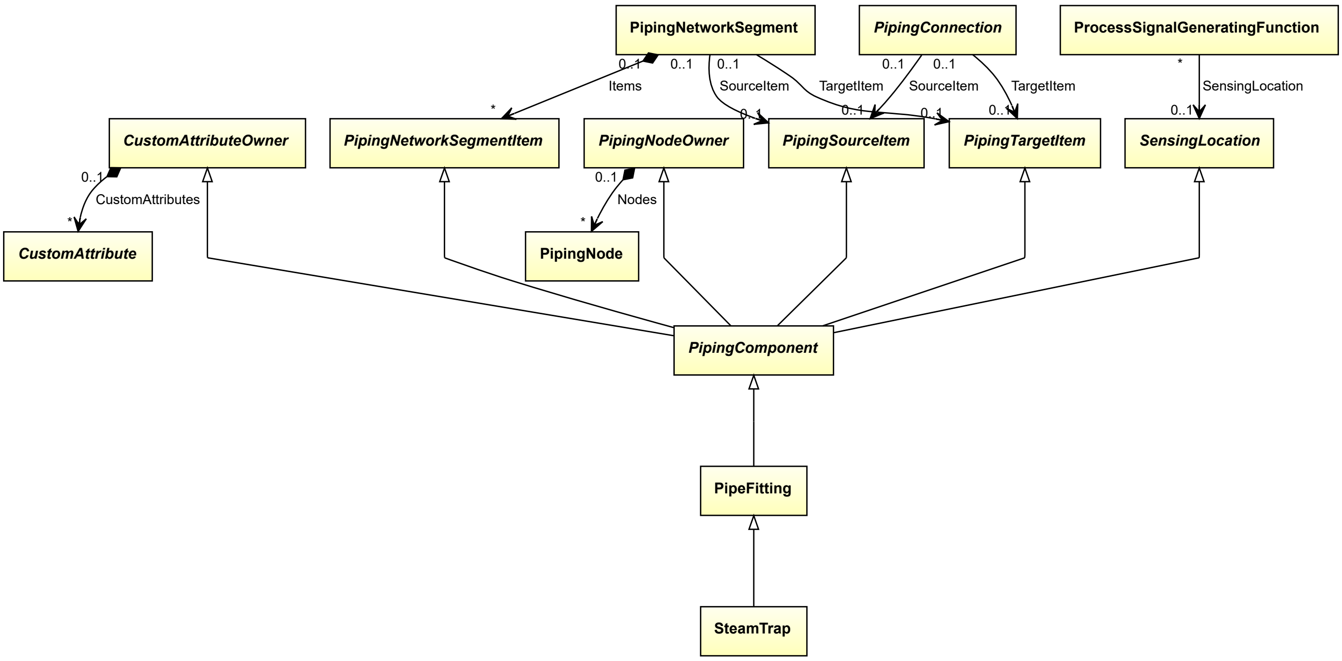
```
<PipingComponent
    ID="springLoadedGlobeSafetyValve1"
    ComponentClass="SpringLoadedGlobeSafetyValve"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SpringLoadedGlobeSafetyValve" ...>
  ...
</PipingComponent>
```

8.71. SteamTrap

8.71.1 Overview

Class

A trap that consists of a chamber into which condensed steam from steam pipes etc. is allowed to drain, and which automatically ejects it without permitting the escape of steam (from http://data.posccaesar.org/rdl/RDS5782388).



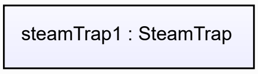
Supertypes

- *PipeFitting*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingComponent>`
RDL reference: STEAM TRAP
ComponentClass: SteamTrap
ComponentClassURI: http://data.posccaesar.org/rdl/RDS5782388

Example



Example: Implementation in Proteus Schema

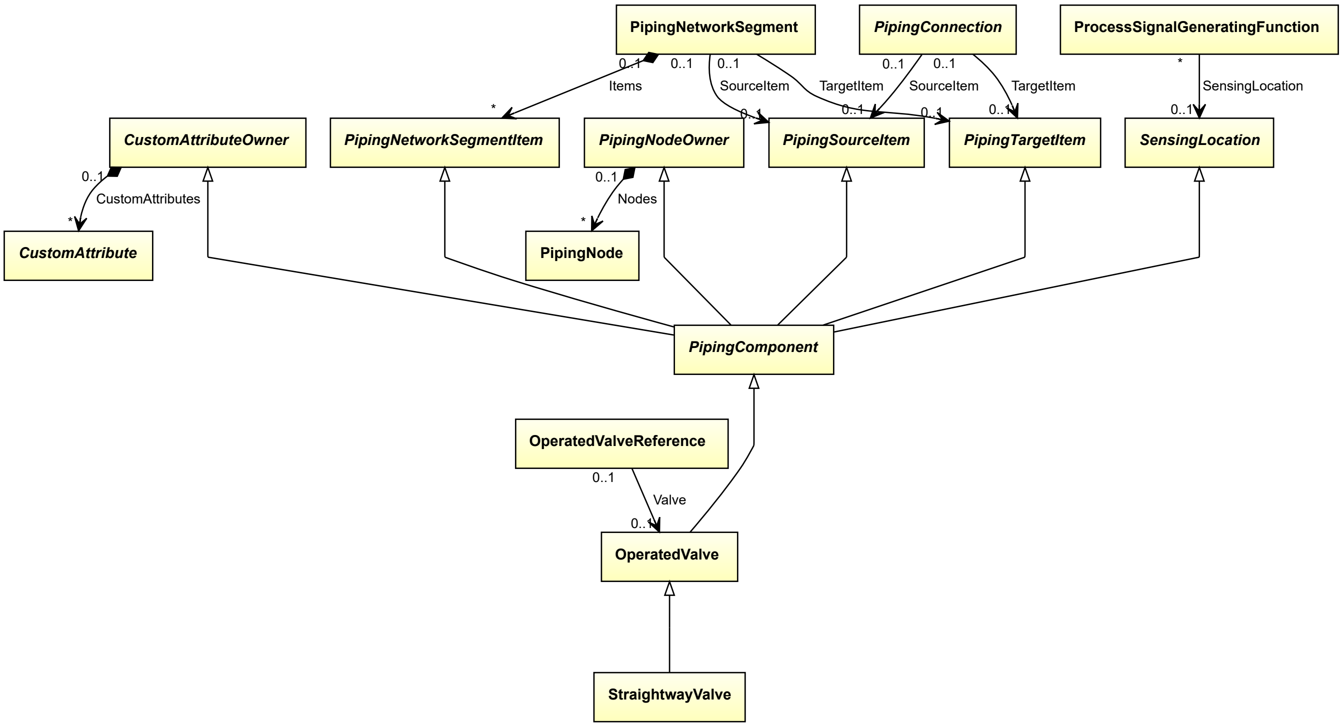
```
<PipingComponent
    ID="steamTrap1"
    ComponentClass="SteamTrap"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS5782388" ...>
  ...
</PipingComponent>
```

8.72. StraightwayValve

8.72.1 Overview

Class

A valve that is straight, i.e. the centerlines perpendicular to the ends are in-line with no offset (from http://data.posccaesar.org/rdl/RDS9390905).



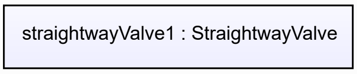
Supertypes

- *OperatedValve*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <PipingComponent>
RDL reference: STRAIGHTWAY VALVE
ComponentClass: StraightwayValve
ComponentClassURI: http://data.posccaesar.org/rdl/RDS9390905

Example



Example: Implementation in Proteus Schema

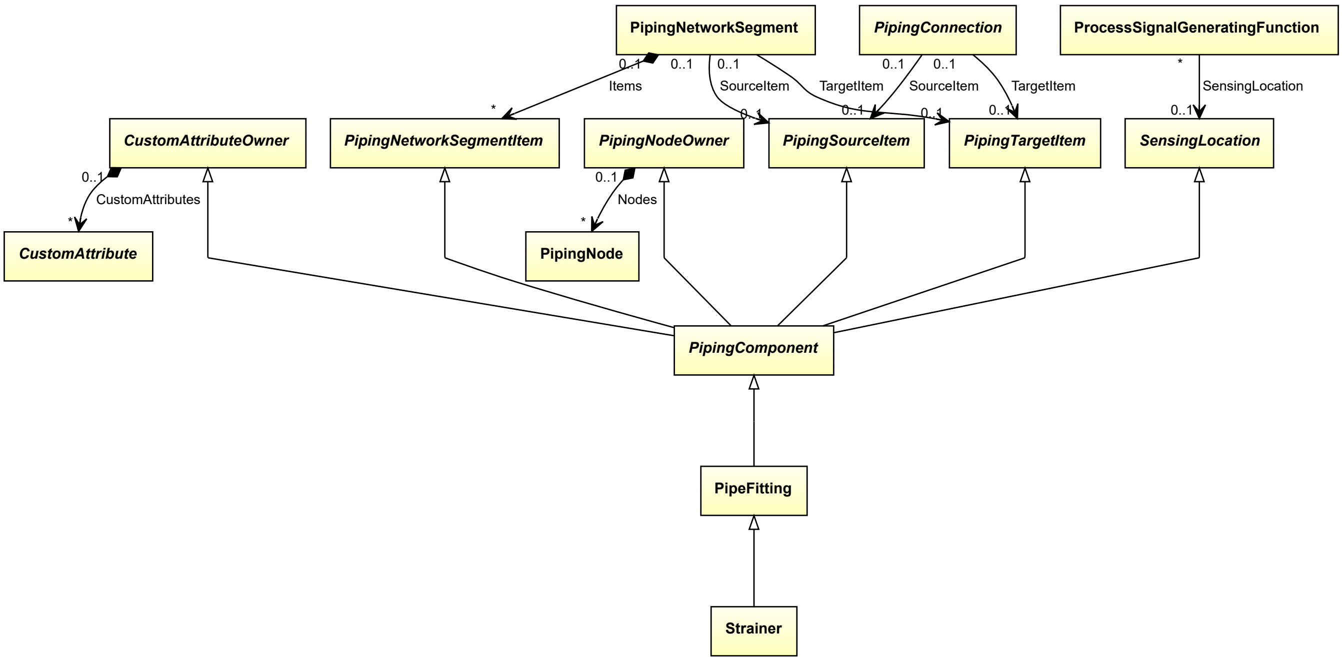
```
<PipingComponent
    ID="straightwayValve1"
    ComponentClass="StraightwayValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS9390905" ...>
  ...
</PipingComponent>
```

8.73. Strainer

8.73.1 Overview

Class

A mechanical separator that is separating solid particles from a fluid by passing the fluid through a wire mesh, screen or metal plates containing perforations or slits (from http://data.posccaesar.org/rdl/RDS422504).



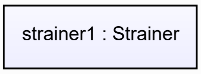
Supertypes

- *PipeFitting*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingComponent>`
RDL reference: STRAINER
ComponentClass: Strainer
ComponentClassURI: http://data.posccaesar.org/rdl/RDS422504

Example



Example: Implementation in Proteus Schema

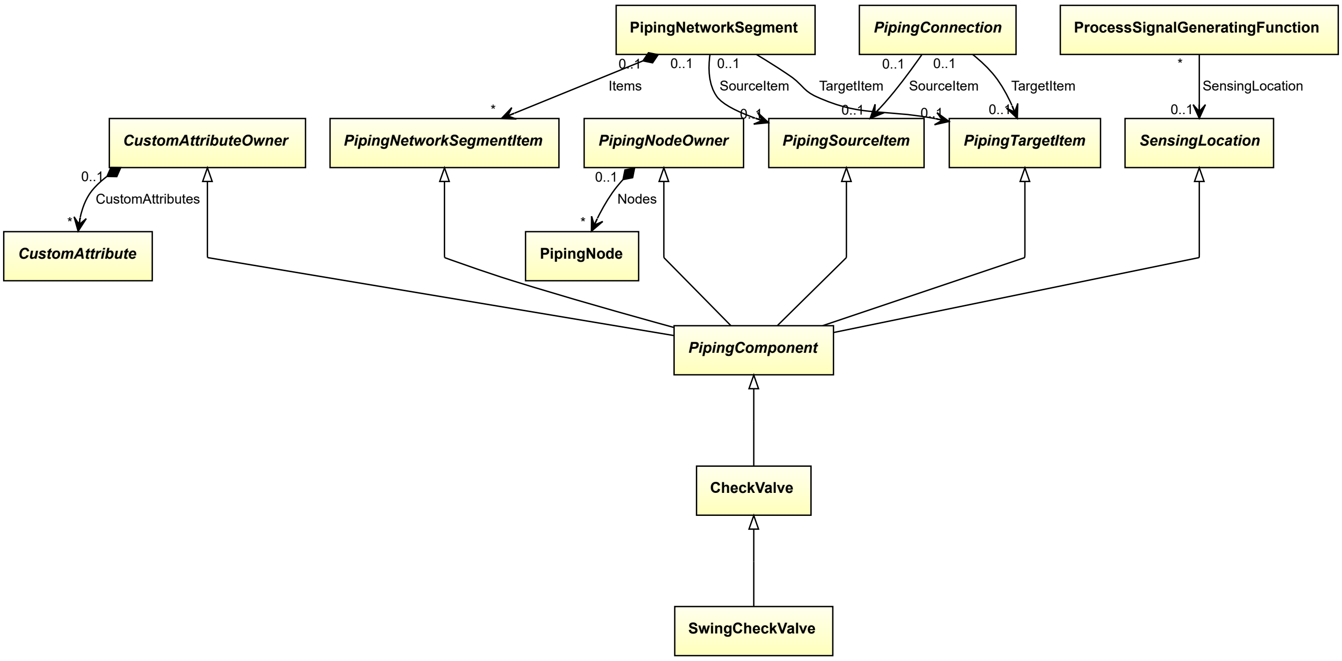
```
<PipingComponent
    ID="strainer1"
    ComponentClass="Strainer"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS422504" ...>
  ...
</PipingComponent>
```

8.74. SwingCheckValve

8.74.1 Overview

Class

A check valve that is a check valve where the closure member is a disc which swings freely on a hinge and which opens automatically when flow is established and closes automatically when flow ceases or is reversed (from http://data.posccaesar.org/rdl/RDS610424).



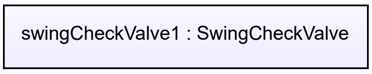
Supertypes

- *CheckValve*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingComponent>`
RDL reference: SWING CHECK VALVE
ComponentClass: SwingCheckValve
ComponentClassURI: http://data.posccaesar.org/rdl/RDS610424

Example



Example: Implementation in Proteus Schema

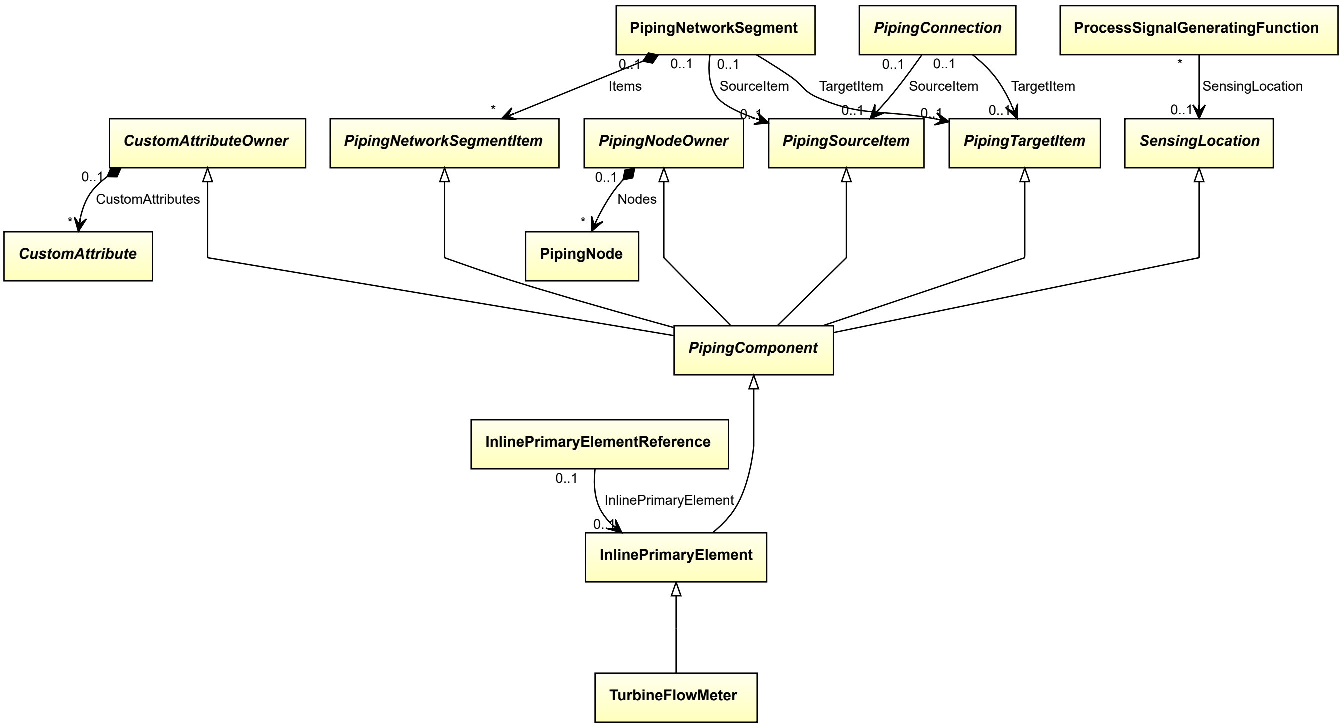
```
<PipingComponent
    ID="swingCheckValve1"
    ComponentClass="SwingCheckValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS610424" ...>
  ...
</PipingComponent>
```

8.75. TurbineFlowMeter

8.75.1 Overview

Class

A velocity flow meter that uses a multi bladed rotor to measure fluid flow rate in units of volumetric flow through a closed conduit (from http://data.posccaesar.org/rdl/RDS417914).



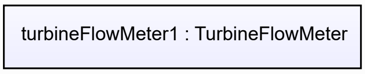
Supertypes

- *InlinePrimaryElement*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingComponent>`
RDL reference: TURBINE FLOW METER
ComponentClass: TurbineFlowMeter
ComponentClassURI: http://data.posccaesar.org/rdl/RDS417914

Example



Example: Implementation in Proteus Schema

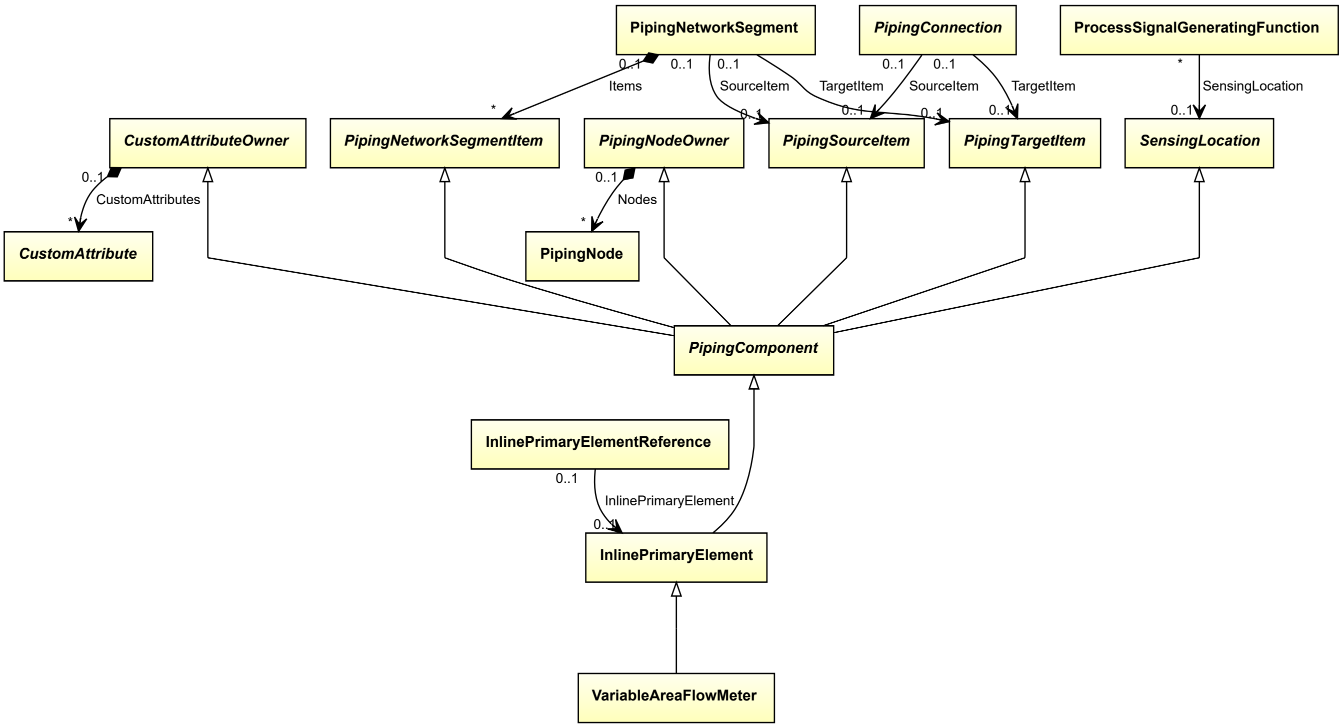
```
<PipingComponent
    ID="turbineFlowMeter1"
    ComponentClass="TurbineFlowMeter"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS417914" ...>
  ...
</PipingComponent>
```

8.76. VariableAreaFlowMeter

8.76.1 Overview

Class

A flow meter consisting of a vertical tube with a conically shaped bore which widens to the top in which a solid body (float) is supported by the force exerted by the fluid stream (from http://data.posccaesar.org/rdl/RDS418229).



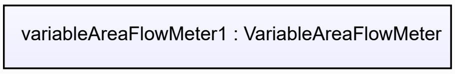
Supertypes

- *InlinePrimaryElement*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingComponent>`
RDL reference: VARIABLE AREA FLOW METER
ComponentClass: VariableAreaFlowMeter
ComponentClassURI: http://data.posccaesar.org/rdl/RDS418229

Example



Example: Implementation in Proteus Schema

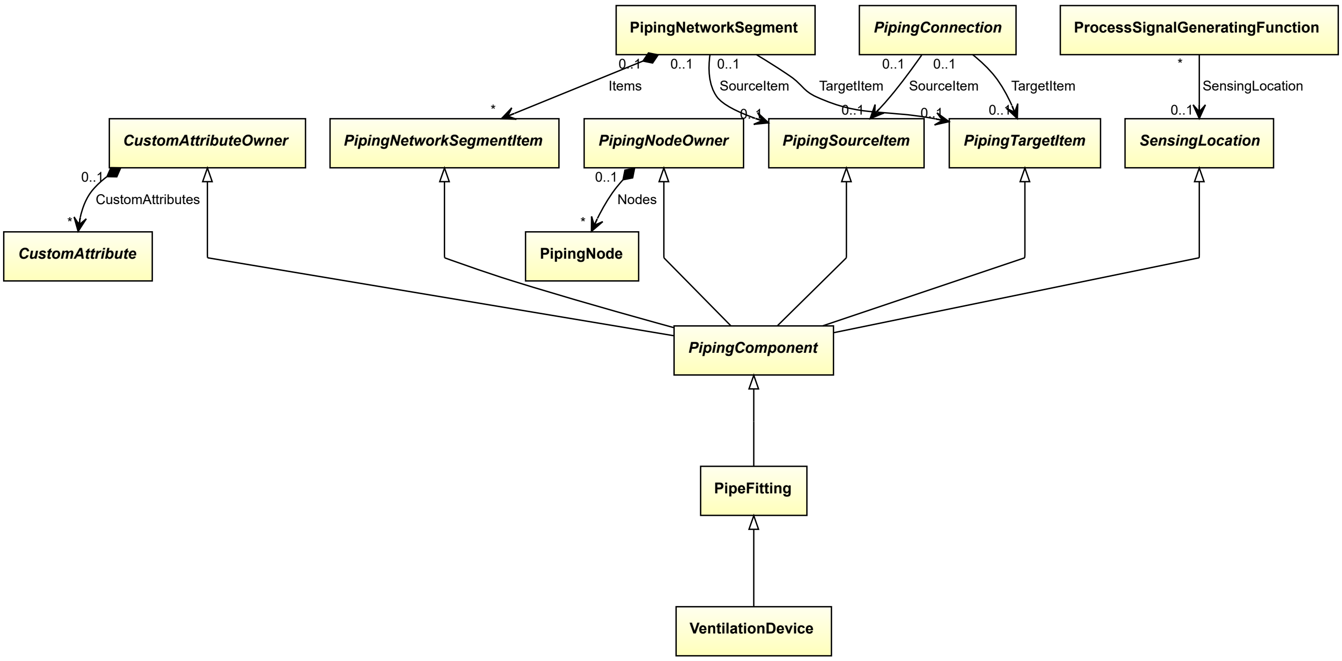
```
<PipingComponent
    ID="variableAreaFlowMeter1"
    ComponentClass="VariableAreaFlowMeter"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS418229" ...>
  ...
</PipingComponent>
```

8.77. VentilationDevice

8.77.1 Overview

Class

A 'device' that allows gas or vapour to leave a container under excess pressure (from http://data.posccaesar.org/rdl/RDS1049335351).



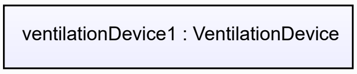
Supertypes

- *PipeFitting*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<PipingComponent>`
RDL reference: VENTILATION DEVICE
ComponentClass: VentilationDevice
ComponentClassURI: http://data.posccaesar.org/rdl/RDS1049335351

Example



Example: Implementation in Proteus Schema

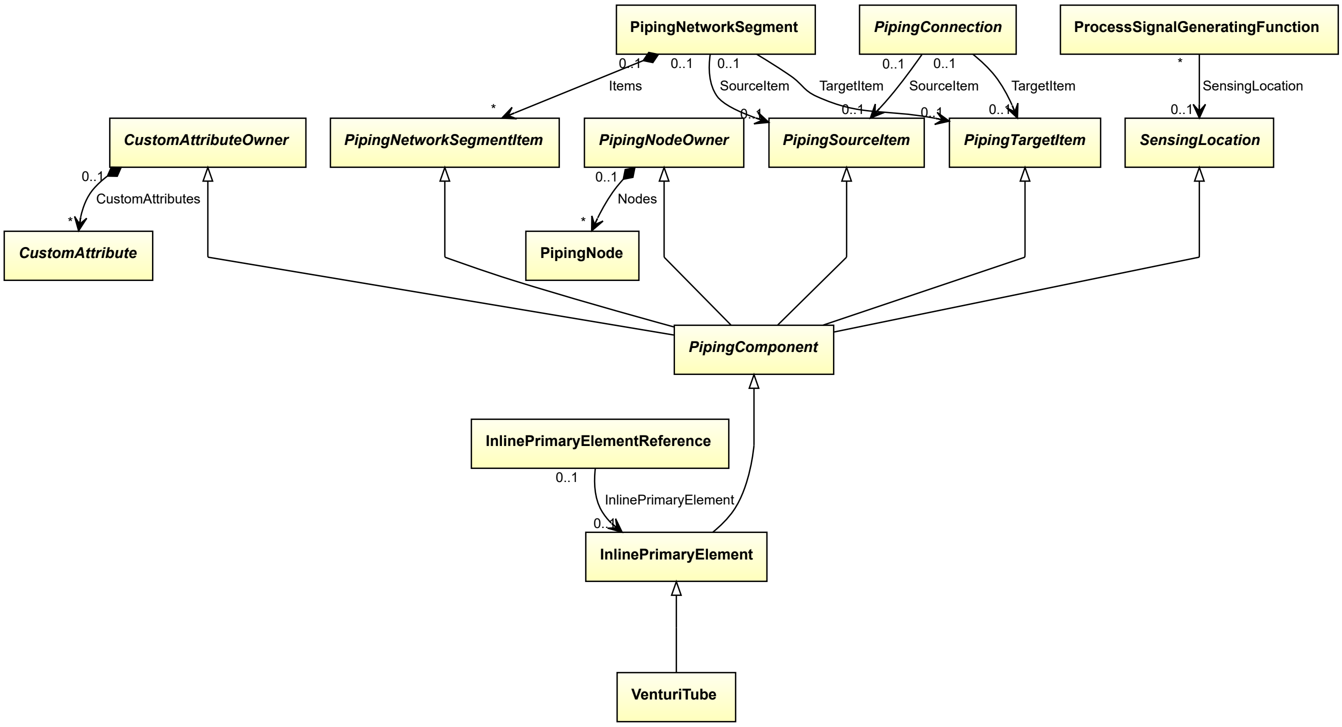
```
<PipingComponent
    ID="ventilationDevice1"
    ComponentClass="VentilationDevice"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS1049335351" ...>
  ...
</PipingComponent>
```

8.78. VenturiTube

8.78.1 Overview

Class

A ‘measuring device’ that has a constriction with a relative long passage with a smooth coned entry and exit (from http://data.posccaesar.org/rdl/RDS648044).



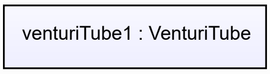
Supertypes

- *InlinePrimaryElement*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <PipingComponent>
RDL reference: VENTURI TUBE
ComponentClass: VenturiTube
ComponentClassURI: http://data.posccaesar.org/rdl/RDS648044

Example



Example: Implementation in Proteus Schema

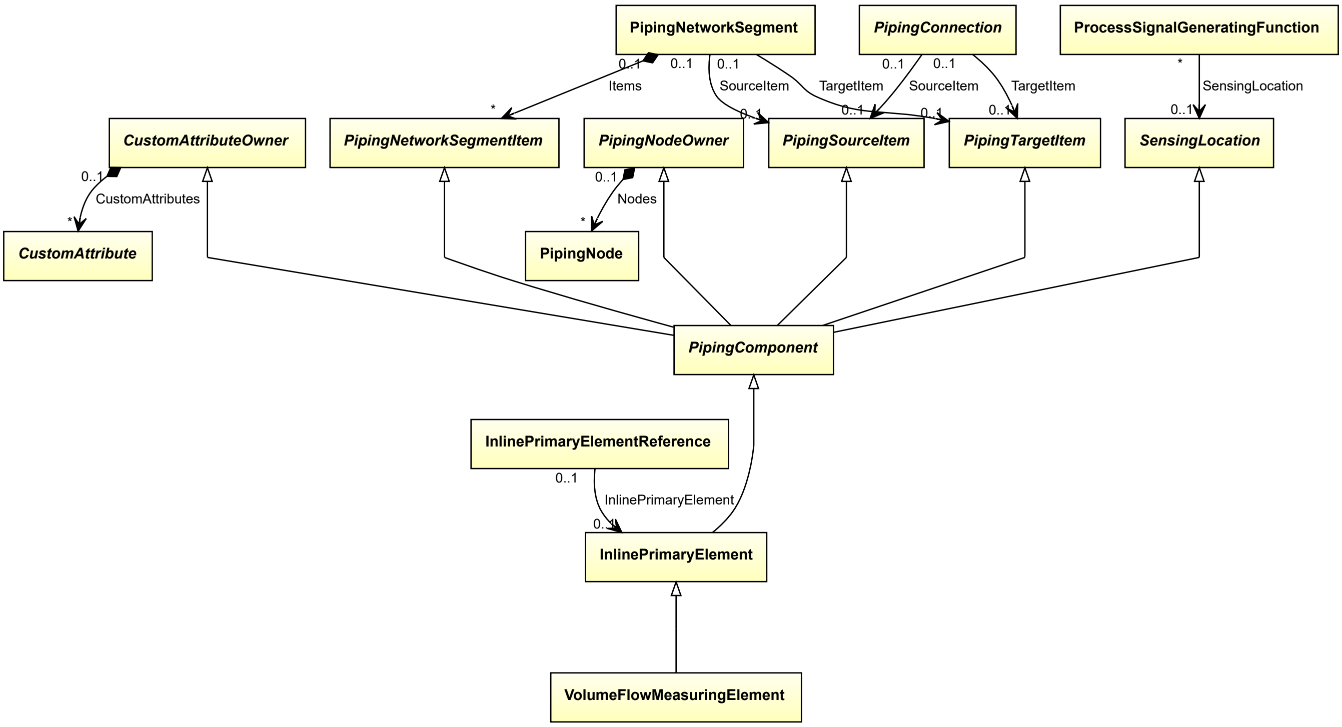
```
<PipingComponent
    ID="venturiTube1"
    ComponentClass="VenturiTube"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS648044" ...>
  ...
</PipingComponent>
```

8.79. VolumeFlowMeasuringElement

8.79.1 Overview

Class

A VOLUME FLOW MEASURING ELEMENT is a FLOW MEASURING ELEMENT that is used to measure VOLUME FLOW RATE.



Supertypes

- *InlinePrimaryElement*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <PipingComponent>
RDL reference: VOLUME FLOW MEASURING ELEMENT
ComponentClass: VolumeFlowMeasuringElement
ComponentClassURI: http://sandbox.dexpi.org/rdl/VolumeFlowMeasuringElement

Example

Example: Implementation in Proteus Schema

```
<PipingComponent
    ID="volumeFlowMeasuringElement1"
    ComponentClass="VolumeFlowMeasuringElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/VolumeFlowMeasuringElement" ...>
  ...
</PipingComponent>
```

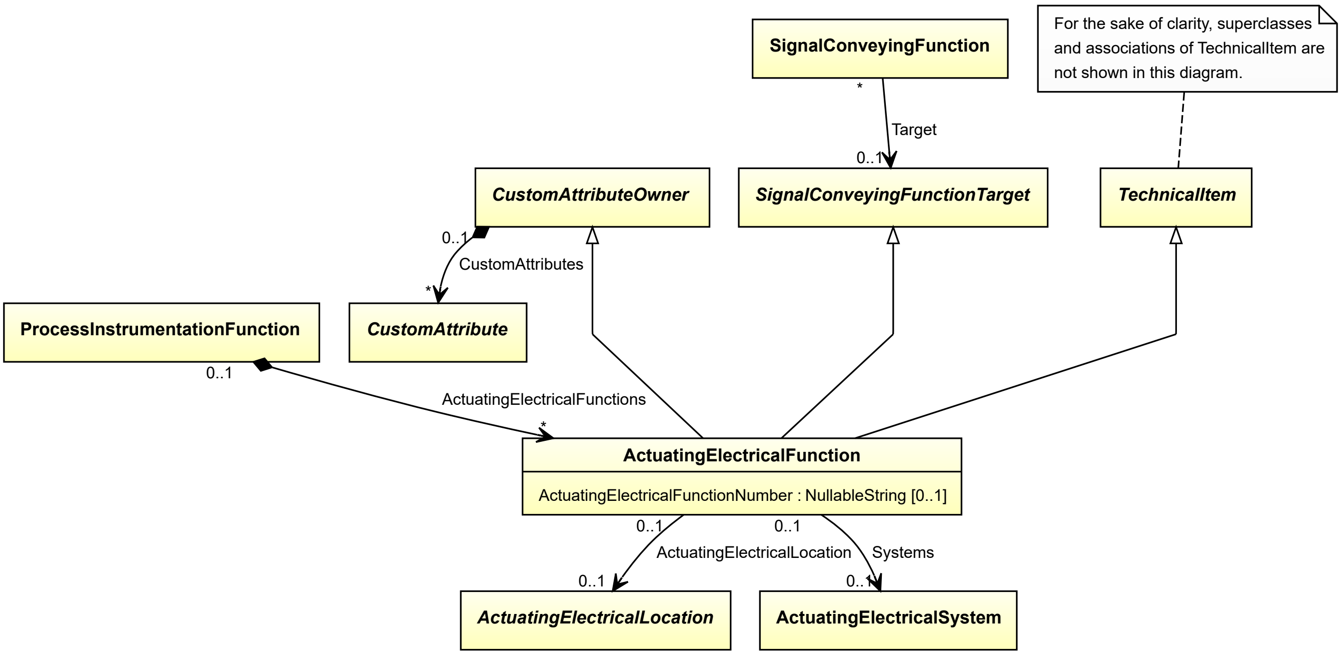
Package Instrumentation 9

9.1. ActuatingElectricalFunction

9.1.1 Overview

Class

An actuation setting electrical function. It covers all types of electrical consumers, e.g., motors and heaters.



SignalConveyingFunction
For the sake of clarity, superclasses and associations of TechnicalItem are not shown in this diagram.
*
Target
0..1
CustomAttributeOwner
SignalConveyingFunctionTarget
TechnicalItem
0..1
CustomAttributes
*
ProcessInstrumentationFunction
CustomAttribute
0..1
ActuatingElectricalFunctions
*
ActuatingElectricalFunction
ActuatingElectricalFunctionNumber : NullableString [0..1]
0..1
0..1
ActuatingElectricalLocation
Systems
0..1
0..1
ActuatingElectricalLocation
ActuatingElectricalSystem

Supertypes

- *CustomAttributeOwner*
- *SignalConveyingFunctionTarget*
- *TechnicalItem*

Attributes (data)

Name	Multiplicity	Type
ActuatingElectricalFunctionNumber	0..1	*NullableString*

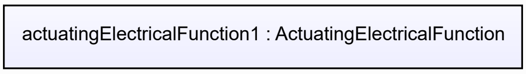
Attributes (reference)

Name	Multiplicity	Type
ActuatingElectricalLocation	0..1	*ActuatingElectricalLocation*
Systems	0..1	*ActuatingElectricalSystem*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <ActuatingElectricalFunction>
RDL reference: ACTUATING ELECTRICAL FUNCTION
ComponentClass: ActuatingElectricalFunction
ComponentClassURI: http://sandbox.dexpi.org/rdl/ActuatingElectricalFunction

Example



Example: Implementation in Proteus Schema

```
<ActuatingElectricalFunction
    ID="actuatingElectricalFunction1"
    ComponentClass="ActuatingElectricalFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ActuatingElectricalFunction" ...>
  ...
</ActuatingElectricalFunction>
```

9.1.2 ActuatingElectricalFunctionNumber

Attribute (data)

An identifier for the *ActuatingElectricalFunction*. It usually contains the identifier of the *ProcessInstrumentationFunction* that includes the *ActuatingElectricalFunction* (see *ProcessInstrumentationFunctionNumber*).

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: ACTUATING ELECTRICAL FUNCTION NUMBER ASSIGNMENT CLASS
Name: ActuatingElectricalFunctionNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ActuatingElectricalFunctionNumberAssignmentClass

Example

"E4750.01" (*String*)

Example: Implementation in Proteus Schema

```
<ActuatingElectricalFunction
    ID="actuatingElectricalFunction1"
    ComponentClass="ActuatingElectricalFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ActuatingElectricalFunction" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ActuatingElectricalFunctionNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ActuatingElectricalFunctionNumberAssignmentClass"
      Format="string"
      Value="E4750.01" />
    ...
  </GenericAttributes>
  ...
</ActuatingElectricalFunction>
```

9.1.3 ActuatingElectricalLocation

Attribute (reference)

The actuating electrical location of the *ActuatingElectricalFunction*.

Multiplicity: 0..1

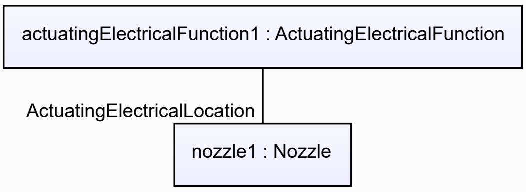
Type: *ActuatingElectricalLocation*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: `"is located in"`
Opposite association type: `"is the location of"`

Example


actuatingElectricalFunction1 : ActuatingElectricalFunction
ActuatingElectricalLocation
nozzle1 : Nozzle

Example: Implementation in Proteus Schema

```
<ActuatingElectricalFunction
    ID="actuatingElectricalFunction1"
    ComponentClass="ActuatingElectricalFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ActuatingElectricalFunction" ...>
  ...
  <Association
    Type="is located in"
    ItemID="nozzle1" />
  ...
<ActuatingElectricalFunction />
...
<Nozzle
    ID="nozzle1"
    ComponentClass="Nozzle"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS415214" ...>
  ...
  <Association
    Type="is the location of"
    ItemID="actuatingElectricalFunction1" />
  ...
<Nozzle />
```

9.1.4 Systems

Attribute (reference)

The ActuatingElectricalSystem that implements the *ActuatingElectricalFunction*.

Multiplicity: 0..1

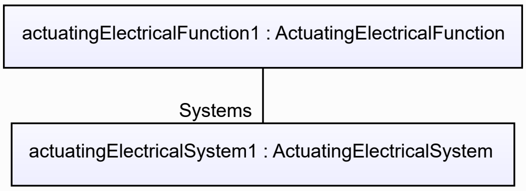
Type: *ActuatingElectricalSystem*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: "is fulfilled by"
Opposite association type: "fulfills"

Example


actuatingElectricalFunction1 : ActuatingElectricalFunction
Systems
actuatingElectricalSystem1 : ActuatingElectricalSystem

Example: Implementation in Proteus Schema

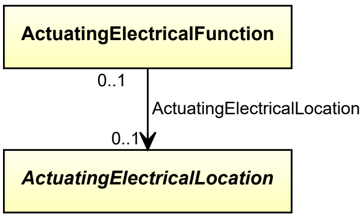
```
<ActuatingElectricalFunction
    ID="actuatingElectricalFunction1"
    ComponentClass="ActuatingElectricalFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ActuatingElectricalFunction" ...>
  ...
  <Association
    Type="is fulfilled by"
    ItemID="actuatingElectricalSystem1" />
  ...
<ActuatingElectricalFunction />
...
<ActuatingElectricalSystem
    ID="actuatingElectricalSystem1"
    ComponentClass="ActuatingElectricalSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ActuatingElectricalSystem" ...>
  ...
  <Association
    Type="fulfills"
    ItemID="actuatingElectricalFunction1" />
  ...
<ActuatingElectricalSystem />
```

9.2. ActuatingElectricalLocation

9.2.1 Overview

Abstract class

An object suitable as the ActuatingElectricalLocation of an *ActuatingElectricalFunction*.


ActuatingElectricalFunction
0..1
ActuatingElectricalLocation
0..1
ActuatingElectricalLocation

Subtypes

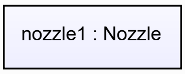
- *Nozzle*
- *PipingNetworkSegment*

Implementation in Proteus Schema

Implementation is subclass-specific.

Example

As *ActuatingElectricalLocation* is abstract, we consider *Nozzle* as an arbitrary concrete subclass.



Example: Implementation in Proteus Schema

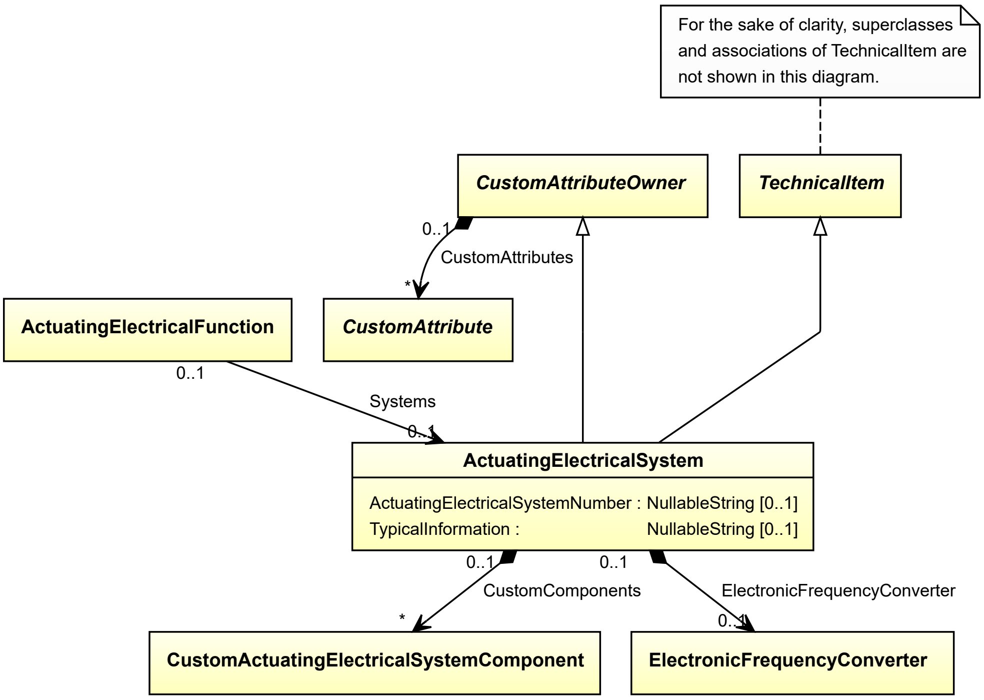
```
<Nozzle
    ID="nozzle1"
    ComponentClass="Nozzle"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS415214" ...>
  ...
</Nozzle>
```

9.3. ActuatingElectricalSystem

9.3.1 Overview

Class

An assembly of artefacts that is designed to fulfill an *ActuatingElectricalFunction*.


For the sake of clarity, superclasses and associations of TechnicalItem are not shown in this diagram.
CustomAttributeOwner
TechnicalItem
0..1
CustomAttributes
*
ActuatingElectricalFunction
CustomAttribute
0..1
Systems
0..1
ActuatingElectricalSystem
ActuatingElectricalSystemNumber : NullableString [0..1]
TypicalInformation : NullableString [0..1]
0..1
0..1
CustomComponents
ElectronicFrequencyConverter
*
0..1
CustomActuatingElectricalSystemComponent
ElectronicFrequencyConverter

Supertypes

- *CustomAttributeOwner*
- *TechnicalItem*

Attributes (data)

Name	Multiplicity	Type
ActuatingElectricalSystemNumber	0..1	*NullableString*
TypicalInformation	0..1	*NullableString*

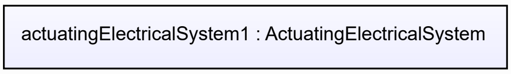
Attributes (composition)

Name	Multiplicity	Type
CustomComponents	*	*CustomActuatingElectricalSystemComponent*
ElectronicFrequencyConverter	0..1	*ElectronicFrequencyConverter*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<ActuatingElectricalSystem>`
RDL reference: ACTUATING ELECTRICAL SYSTEM
ComponentClass: ActuatingElectricalSystem
ComponentClassURI: http://sandbox.dexpi.org/rdl/ActuatingElectricalSystem

Example



Example: Implementation in Proteus Schema

```
<ActuatingElectricalSystem
    ID="actuatingElectricalSystem1"
    ComponentClass="ActuatingElectricalSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ActuatingElectricalSystem" ...>
  ...
</ActuatingElectricalSystem>
```

9.3.2 ActuatingElectricalSystemNumber

Attribute (data)

The number of *ActuatingElectricalSystem*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: ACTUATING SYSTEM NUMBER ASSIGNMENT CLASS
Name: ActuatingSystemNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ActuatingSystemNumberAssignmentClass

Example

“E0001” (*String*)

Example: Implementation in Proteus Schema

```
<ActuatingElectricalSystem
    ID="actuatingElectricalSystem1"
    ComponentClass="ActuatingElectricalSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ActuatingElectricalSystem" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ActuatingSystemNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ActuatingSystemNumberAssignmentClass"
      Format="string"
      Value="E0001" />
    ...
  </GenericAttributes>
  ...
</ActuatingElectricalSystem>
```

9.3.3 CustomComponents

Attribute (composition)

The custom components of the *ActuatingElectricalSystem*.

Multiplicity: *

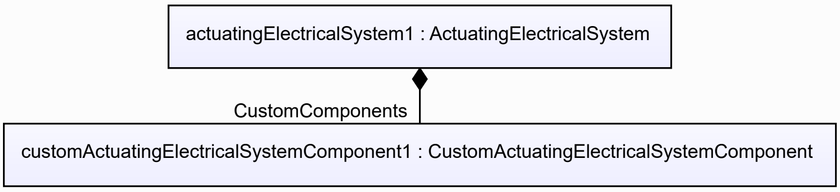
Type: *CustomActuatingElectricalSystemComponent*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *CustomActuatingElectricalSystemComponent*) is a child of the `<ActuatingElectricalSystem>` element for the attribute owner (an *ActuatingElectricalSystem*).

Example


actuatingElectricalSystem1 : ActuatingElectricalSystem
CustomComponents
customActuatingElectricalSystemComponent1 : CustomActuatingElectricalSystemComponent

Example: Implementation in Proteus Schema

```
<ActuatingElectricalSystem
    ID="actuatingElectricalSystem1"
    ComponentClass="ActuatingElectricalSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ActuatingElectricalSystem" ...>
  ...
  <ActuatingElectricalSystemComponent
      ID="customActuatingElectricalSystemComponent1"
      ComponentClass="CustomActuatingElectricalSystemComponent"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomActuatingElectricalSystemComponent" ...>
    ...
  <ActuatingElectricalSystemComponent />
  ...
<ActuatingElectricalSystem />
```

9.3.4 ElectronicFrequencyConverter

Attribute (composition)

The *ElectronicFrequencyConverter* of the *ActuatingElectricalSystem*.

Multiplicity: 0..1

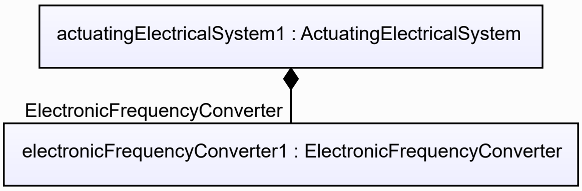
Type: *ElectronicFrequencyConverter*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (an *ElectronicFrequencyConverter*) is a child of the `<ActuatingElectricalSystem>` element for the attribute owner (an *ActuatingElectricalSystem*).

Example


actuatingElectricalSystem1 : ActuatingElectricalSystem
ElectronicFrequencyConverter
electronicFrequencyConverter1 : ElectronicFrequencyConverter

Example: Implementation in Proteus Schema

```
<ActuatingElectricalSystem
    ID="actuatingElectricalSystem1"
    ComponentClass="ActuatingElectricalSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ActuatingElectricalSystem" ...>
  ...
  <ActuatingElectricalSystemComponent
      ID="electronicFrequencyConverter1"
      ComponentClass="ElectronicFrequencyConverter"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/ElectronicFrequencyConverter" ...>
    ...
  <ActuatingElectricalSystemComponent />
  ...
<ActuatingElectricalSystem />
```

9.3.5 TypicalInformation

Attribute (data)

Typical information about the *ActuatingElectricalSystem*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: TYPICAL INFORMATION ASSIGNMENT CLASS
Name: TypicalInformationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/TypicalInformationAssignmentClass

Example

“F4” (*String*)

Example: Implementation in Proteus Schema

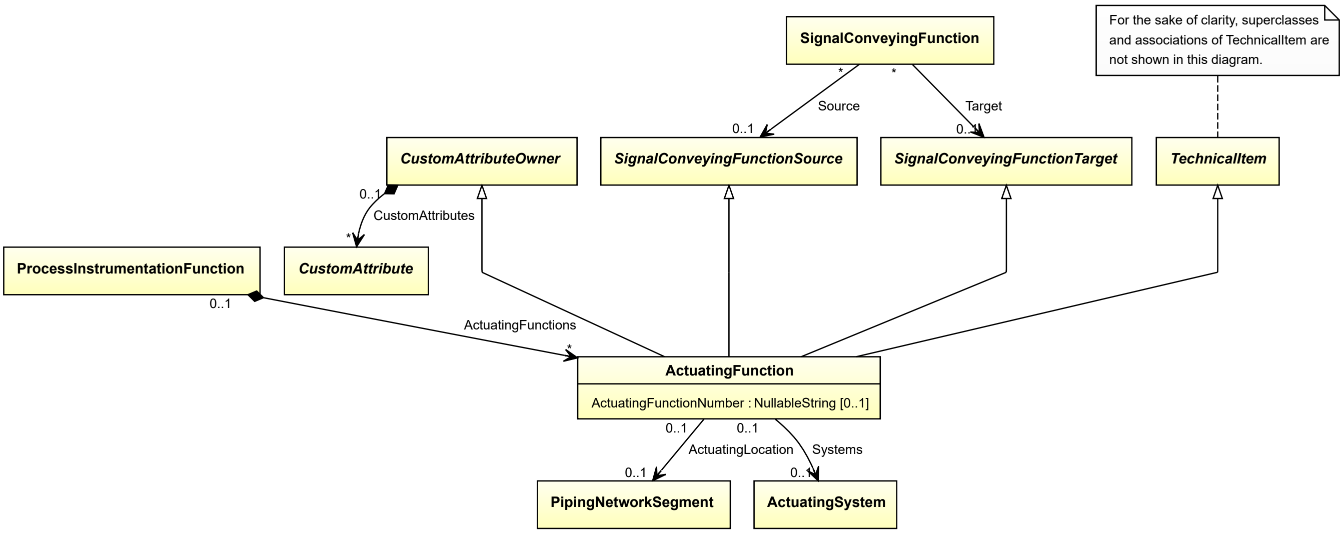
```
<ActuatingElectricalSystem
    ID="actuatingElectricalSystem1"
    ComponentClass="ActuatingElectricalSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ActuatingElectricalSystem" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="TypicalInformationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/TypicalInformationAssignmentClass"
      Format="string"
      Value="F4" />
    ...
  </GenericAttributes>
  ...
</ActuatingElectricalSystem>
```

9.4. ActuatingFunction

9.4.1 Overview

Class

A function for acting control structures relating to the process.


SignalConveyingFunction
For the sake of clarity, superclasses and associations of TechnicalItem are not shown in this diagram.
*
*
Source
Target
0..1
0..1
CustomAttributeOwner
SignalConveyingFunctionSource
SignalConveyingFunctionTarget
TechnicalItem
0..1
CustomAttributes
*
ProcessInstrumentationFunction
CustomAttribute
0..1
ActuatingFunctions
*
ActuatingFunction
ActuatingFunctionNumber : NullableString [0..1]
0..1
0..1
ActuatingLocation
Systems
0..1
0..1
PipingNetworkSegment
ActuatingSystem

Supertypes

- *CustomAttributeOwner*
- *SignalConveyingFunctionSource*
- *SignalConveyingFunctionTarget*
- *TechnicalItem*

Attributes (data)

Name	Multiplicity	Type
ActuatingFunctionNumber	0..1	*NullableString*

Attributes (reference)

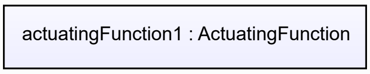
Name	Multiplicity	Type
ActuatingLocation	0..1	*PipingNetworkSegment*
Systems	0..1	*ActuatingSystem*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<ActuatingFunction>`
RDL reference: ACTUATING FUNCTION

ComponentClass: ActuatingFunction
ComponentClassURI: http://sandbox.dexpi.org/rdl/ActuatingFunction

Example



Example: Implementation in Proteus Schema

```
<ActuatingFunction
    ID="actuatingFunction1"
    ComponentClass="ActuatingFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ActuatingFunction" ...>
  ...
</ActuatingFunction>
```

9.4.2 ActuatingFunctionNumber

Attribute (data)

An identifier for the *ActuatingFunction*. It usually contains the identifier of the *ProcessInstrumentationFunction* that includes the *ActuatingFunction* (see *ProcessInstrumentationFunctionNumber*).

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: ACTUATING FUNCTION NUMBER ASSIGNMENT CLASS
Name: ActuatingFunctionNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ActuatingFunctionNumberAssignmentClass

Example

“HV4750.01” (*String*)

Example: Implementation in Proteus Schema

```
<ActuatingFunction
    ID="actuatingFunction1"
    ComponentClass="ActuatingFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ActuatingFunction" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ActuatingFunctionNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ActuatingFunctionNumberAssignmentClass"
      Format="string"
      Value="HV4750.01" />
    ...
  </GenericAttributes>
  ...
</ActuatingFunction>
```

9.4.3 ActuatingLocation

Attribute (reference)

The actuating location of the *ActuatingFunction*.

Multiplicity: 0..1

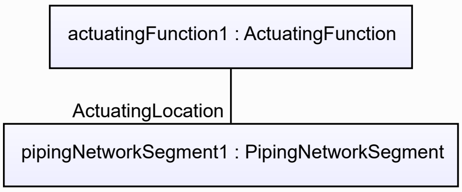
Type: *PipingNetworkSegment*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: `"is located in"`
Opposite association type: `"is the location of"`

Example


actuatingFunction1 : ActuatingFunction
ActuatingLocation
pipingNetworkSegment1 : PipingNetworkSegment

Example: Implementation in Proteus Schema

```
<ActuatingFunction
    ID="actuatingFunction1"
    ComponentClass="ActuatingFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ActuatingFunction" ...>
  ...
  <Association
    Type="is located in"
    ItemID="pipingNetworkSegment1" />
  ...
<ActuatingFunction />
...
<PipingNetworkSegment
    ID="pipingNetworkSegment1"
    ComponentClass="PipingNetworkSegment"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267704" ...>
  ...
  <Association
    Type="is the location of"
    ItemID="actuatingFunction1" />
  ...
<PipingNetworkSegment />
```

9.4.4 Systems

Attribute (reference)

The ActuatingSystem that implements the *ActuatingFunction*.

Multiplicity: 0..1

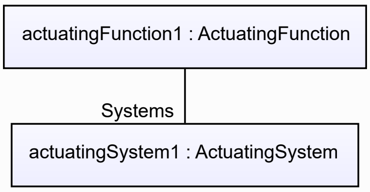
Type: *ActuatingSystem*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: `"is fulfilled by"`
Opposite association type: `"fulfills"`

Example


actuatingFunction1 : ActuatingFunction
Systems
actuatingSystem1 : ActuatingSystem

Example: Implementation in Proteus Schema

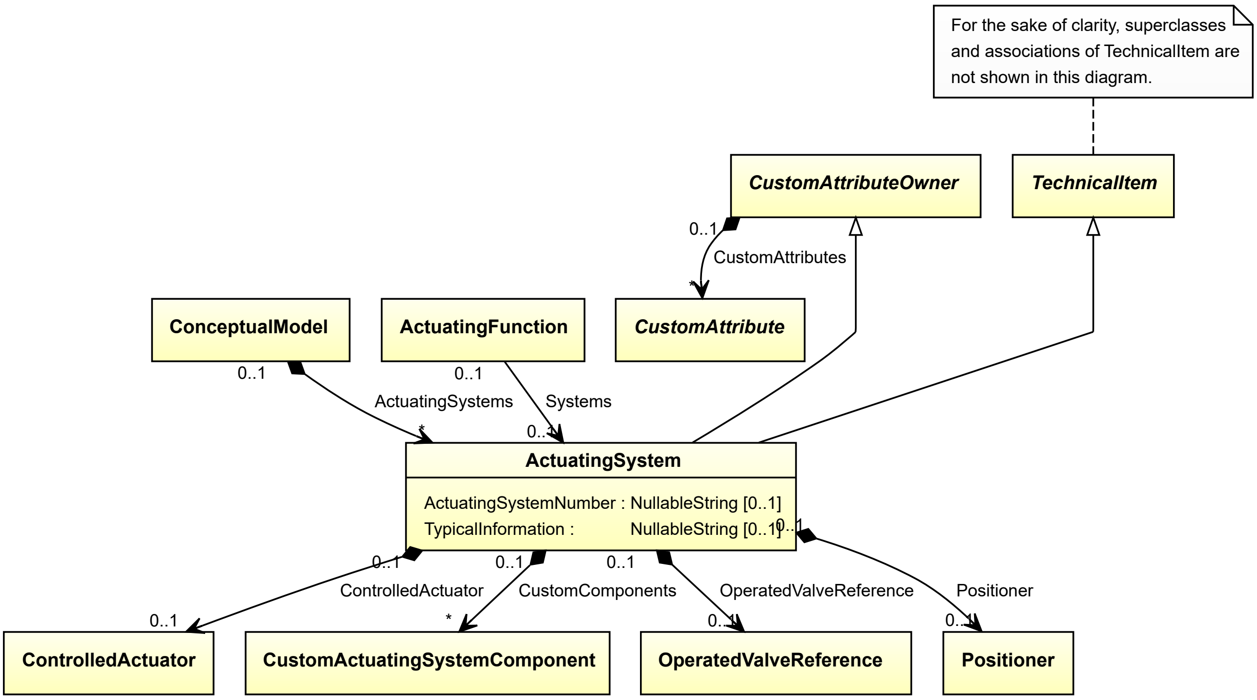
```
<ActuatingFunction
    ID="actuatingFunction1"
    ComponentClass="ActuatingFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ActuatingFunction" ...>
  ...
  <Association
    Type="is fulfilled by"
    ItemID="actuatingSystem1" />
  ...
<ActuatingFunction />
...
<ActuatingSystem
    ID="actuatingSystem1"
    ComponentClass="ActuatingSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ActuatingSystem" ...>
  ...
  <Association
    Type="fulfills"
    ItemID="actuatingFunction1" />
  ...
<ActuatingSystem />
```

9.5. ActuatingSystem

9.5.1 Overview

Class

An assembly of artefacts that is designed to fulfill an *ActuatingFunction*.



Supertypes

- *CustomAttributeOwner*
- *TechnicalItem*

Attributes (data)

Name	Multiplicity	Type
ActuatingSystemNumber	0..1	*NullableString*
TypicalInformation	0..1	*NullableString*

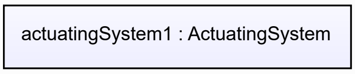
Attributes (composition)

Name	Multiplicity	Type
ControlledActuator	0..1	*ControlledActuator*
CustomComponents	*	*CustomActuatingSystemComponent*
OperatedValveReference	0..1	*OperatedValveReference*
Positioner	0..1	*Positioner*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<ActuatingSystem>`
RDL reference: ACTUATING SYSTEM
ComponentClass: ActuatingSystem
ComponentClassURI: http://sandbox.dexpi.org/rdl/ActuatingSystem

Example



Example: Implementation in Proteus Schema

```
<ActuatingSystem
    ID="actuatingSystem1"
    ComponentClass="ActuatingSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ActuatingSystem" ...>
  ...
</ActuatingSystem>
```

9.5.2 ActuatingSystemNumber

Attribute (data)

The number of the *ActuatingSystem*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: ACTUATING SYSTEM NUMBER ASSIGNMENT CLASS
Name: ActuatingSystemNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ActuatingSystemNumberAssignmentClass

Example

“FT0001” (*String*)

Example: Implementation in Proteus Schema

```
<ActuatingSystem
    ID="actuatingSystem1"
    ComponentClass="ActuatingSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ActuatingSystem" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ActuatingSystemNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ActuatingSystemNumberAssignmentClass"
      Format="string"
      Value="FT0001" />
    ...
  </GenericAttributes>
  ...
</ActuatingSystem>
```

9.5.3 ControlledActuator

Attribute (composition)

The *ControlledActuator* of the *ActuatingSystem*.

Multiplicity: 0..1

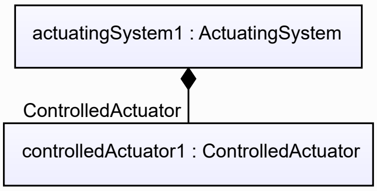
Type: *ControlledActuator*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *ControlledActuator*) is a child of the `<ActuatingSystem>` element for the attribute owner (an *ActuatingSystem*).

Example


actuatingSystem1 : ActuatingSystem
ControlledActuator
controlledActuator1 : ControlledActuator

Example: Implementation in Proteus Schema

```
<ActuatingSystem
    ID="actuatingSystem1"
    ComponentClass="ActuatingSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ActuatingSystem" ...>
  ...
  <ActuatingSystemComponent
      ID="controlledActuator1"
      ComponentClass="ControlledActuator"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/ControlledActuator" ...>
    ...
  <ActuatingSystemComponent />
  ...
<ActuatingSystem />
```

9.5.4 CustomComponents

Attribute (composition)

The custom components of the *ActuatingSystem*.

Multiplicity: *

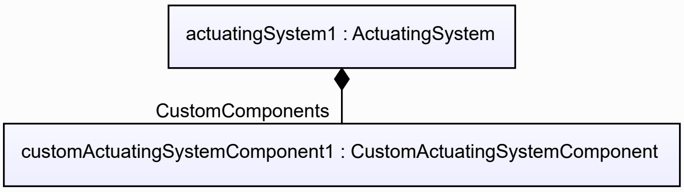
Type: *CustomActuatingSystemComponent*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *CustomActuatingSystemComponent*) is a child of the `<ActuatingSystem>` element for the attribute owner (an *ActuatingSystem*).

Example


actuatingSystem1 : ActuatingSystem
CustomComponents
customActuatingSystemComponent1 : CustomActuatingSystemComponent

Example: Implementation in Proteus Schema

```
<ActuatingSystem
    ID="actuatingSystem1"
    ComponentClass="ActuatingSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ActuatingSystem" ...>
  ...
  <ActuatingSystemComponent
      ID="customActuatingSystemComponent1"
      ComponentClass="CustomActuatingSystemComponent"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomActuatingSystemComponent" ...>
    ...
  <ActuatingSystemComponent />
  ...
<ActuatingSystem />
```

9.5.5 OperatedValveReference

Attribute (composition)

The *OperatedValveReference* of the *ActuatingSystem*.

Multiplicity: 0..1

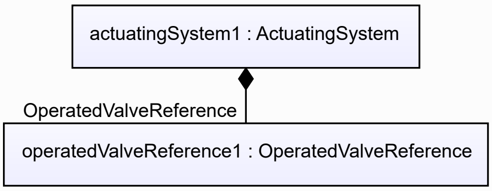
Type: *OperatedValveReference*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (an *OperatedValveReference*) is a child of the `<ActuatingSystem>` element for the attribute owner (an *ActuatingSystem*).

Example


actuatingSystem1 : ActuatingSystem
OperatedValveReference
operatedValveReference1 : OperatedValveReference

Example: Implementation in Proteus Schema

```
<ActuatingSystem
    ID="actuatingSystem1"
    ComponentClass="ActuatingSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ActuatingSystem" ...>
  ...
  <ActuatingSystemComponent
      ID="operatedValveReference1"
      ComponentClass="OperatedValveReference"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/OperatedValveReference" ...>
    ...
  <ActuatingSystemComponent />
  ...
<ActuatingSystem />
```

9.5.6 Positioner

Attribute (composition)

The *Positioner* of the *ActuatingSystem*.

Multiplicity: 0..1

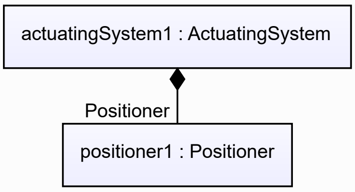
Type: *Positioner*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *Positioner*) is a child of the `<ActuatingSystem>` element for the attribute owner (an *ActuatingSystem*).

Example



Example: Implementation in Proteus Schema

```
<ActuatingSystem
    ID="actuatingSystem1"
    ComponentClass="ActuatingSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ActuatingSystem" ...>
  ...
  <ActuatingSystemComponent
      ID="positioner1"
      ComponentClass="Positioner"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/Positioner" ...>
    ...
  <ActuatingSystemComponent />
  ...
<ActuatingSystem />
```

9.5.7 TypicalInformation

Attribute (data)

Typical information about the *ActuatingSystem*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: TYPICAL INFORMATION ASSIGNMENT CLASS
Name: TypicalInformationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/TypicalInformationAssignmentClass

Example

“F4” (*String*)

Example: Implementation in Proteus Schema

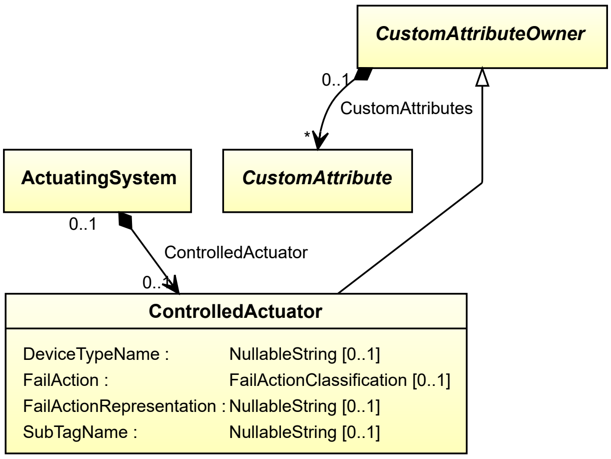
```
<ActuatingSystem
    ID="actuatingSystem1"
    ComponentClass="ActuatingSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ActuatingSystem" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="TypicalInformationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/TypicalInformationAssignmentClass"
      Format="string"
      Value="F4" />
    ...
  </GenericAttributes>
  ...
</ActuatingSystem>
```

9.6. ControlledActuator

9.6.1 Overview

Class

A transducer that is intended to convert energy (electric, mechanical, pneumatic or hydraulic) from an external source into kinetic energy (motion) in response to a signal or power input.


CustomAttributeOwner
0..1
CustomAttributes
*
ActuatingSystem
CustomAttribute
0..1
ControlledActuator
0..1
ControlledActuator
DeviceTypeName : NullableString [0..1]
FailAction : FailActionClassification [0..1]
FailActionRepresentation : NullableString [0..1]
SubTagName : NullableString [0..1]

Supertypes

- *CustomAttributeOwner*

Attributes (data)

Name	Multiplicity	Type
DeviceTypeName	0..1	*NullableString*
FailAction	0..1	*FailActionClassification*
FailActionRepresentation	0..1	*NullableString*
SubTagName	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<ActuatingSystemComponent>`
RDL reference: CONTROLLED ACTUATOR
ComponentClass: ControlledActuator
ComponentClassURI: http://sandbox.dexpi.org/rdl/ControlledActuator

Example

Example: Implementation in Proteus Schema

```
<ActuatingSystemComponent
    ID="controlledActuator1"
    ComponentClass="ControlledActuator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ControlledActuator" ...>
  ...
</ActuatingSystemComponent>
```

9.6.2 DeviceTypeName

Attribute (data)

The device type of the *ControlledActuator*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: DEVICE TYPE NAME ASSIGNMENT CLASS
Name: DeviceTypeNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/DeviceTypeNameAssignmentClass

Example

“pressure transmitter” (*String*)

Example: Implementation in Proteus Schema

```
<ActuatingSystemComponent
    ID="controlledActuator1"
    ComponentClass="ControlledActuator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ControlledActuator" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DeviceTypeNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/DeviceTypeNameAssignmentClass"
      Format="string"
      Value="pressure transmitter" />
    ...
  </GenericAttributes>
  ...
</ActuatingSystemComponent>
```

9.6.3 FailAction

Attribute (data)

The fail action of the *ControlledActuator*.

Multiplicity: 0..1

Type: *FailActionClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: FAIL ACTION SPECIALIZATION
Name: FailActionSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/FailActionSpecialization

Example

fail open (*FailActionClassification::FailOpen*)

Example: Implementation in Proteus Schema

```
<ActuatingSystemComponent
    ID="controlledActuator1"
    ComponentClass="ControlledActuator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ControlledActuator" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="FailActionSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/FailActionSpecialization"
      Format="anyURI"
      Value="FailOpen"
      ValueURI="http://data.posccaesar.org/rdl/RDS5921445" />
    ...
  </GenericAttributes>
  ...
</ActuatingSystemComponent>
```

9.6.4 FailActionRepresentation

Attribute (data)

A readable representation of the fail action of the *ControlledActuator*. This attribute should also be referenced in the graphics if applicable.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: FAIL ACTION REPRESENTATION ASSIGNMENT CLASS
Name: FailActionRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/FailActionRepresentationAssignmentClass

Example

“F.O.” (*String*)

Example: Implementation in Proteus Schema

```
<ActuatingSystemComponent
    ID="controlledActuator1"
    ComponentClass="ControlledActuator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ControlledActuator" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="FailActionRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/FailActionRepresentationAssignmentClass"
      Format="string"
      Value="F.O." />
    ...
  </GenericAttributes>
  ...
</ActuatingSystemComponent>
```

9.6.5 SubTagName

Attribute (data)

The sub tag name of the *ControlledActuator*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: SUB TAG NAME ASSIGNMENT CLASS
Name: SubTagNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/SubTagNameAssignmentClass

Example

“ST1” (*String*)

Example: Implementation in Proteus Schema

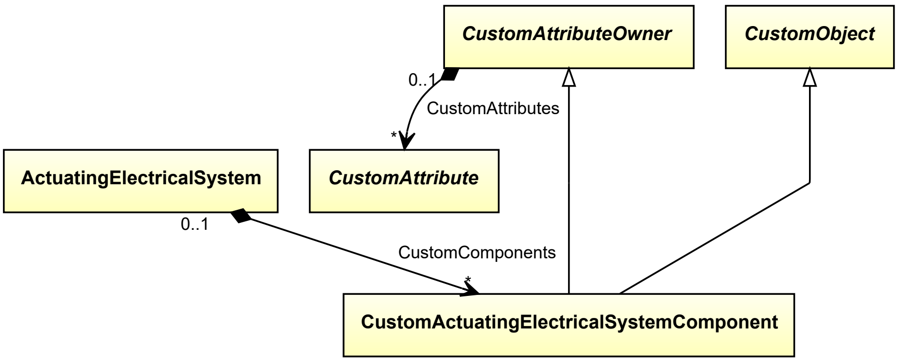
```
<ActuatingSystemComponent
    ID="controlledActuator1"
    ComponentClass="ControlledActuator"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ControlledActuator" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="SubTagNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/SubTagNameAssignmentClass"
      Format="string"
      Value="ST1" />
    ...
  </GenericAttributes>
  ...
</ActuatingSystemComponent>
```

9.7. CustomActuatingElectricalSystemComponent

9.7.1 Overview

Class

A custom component of an *ActuatingElectricalSystem*, i.e., a component other than .



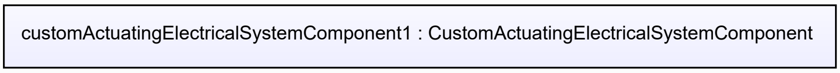
Supertypes

- *CustomAttributeOwner*
- *CustomObject*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<ActuatingElectricalSystemComponent>`
RDL reference: CUSTOM ACTUATING ELECTRICAL SYSTEM COMPONENT
ComponentClass: CustomActuatingElectricalSystemComponent
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomActuatingElectricalSystemComponent

Example


customActuatingElectricalSystemComponent1 : CustomActuatingElectricalSystemComponent

Example: Implementation in Proteus Schema

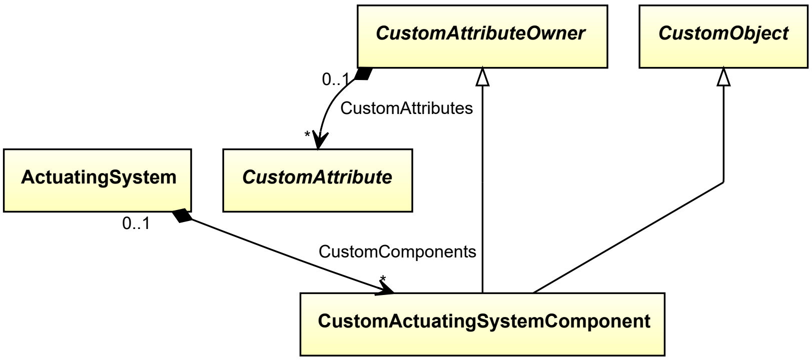
```
<ActuatingElectricalSystemComponent
    ID="customActuatingElectricalSystemComponent1"
    ComponentClass="CustomActuatingElectricalSystemComponent"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomActuatingElectricalSystemComponent" ...>
  ...
</ActuatingElectricalSystemComponent>
```

9.8. CustomActuatingSystemComponent

9.8.1 Overview

Class

A custom component of an *ActuatingSystem*, i.e., a component other than a *ControlledActuator*, an *OperatedValveReference*, or a *Positioner*.



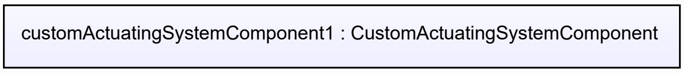
Supertypes

- *CustomAttributeOwner*
- *CustomObject*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<ActuatingSystemComponent>`
RDL reference: CUSTOM ACTUATING SYSTEM COMPONENT
ComponentClass: CustomActuatingSystemComponent
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomActuatingSystemComponent

Example


customActuatingSystemComponent1 : CustomActuatingSystemComponent

Example: Implementation in Proteus Schema

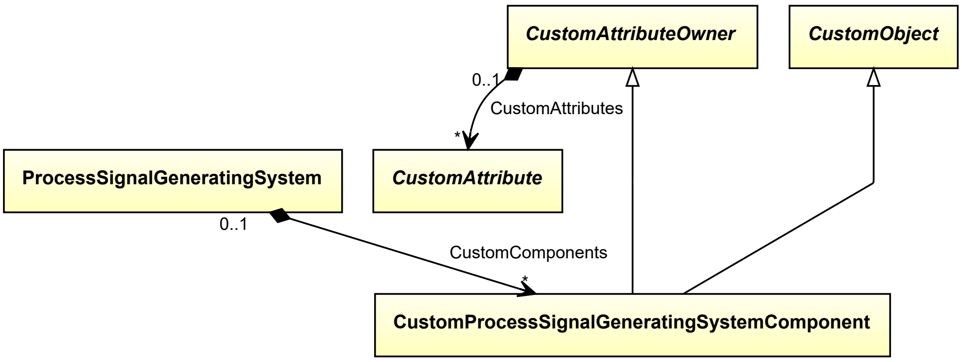
```
<ActuatingSystemComponent
    ID="customActuatingSystemComponent1"
    ComponentClass="CustomActuatingSystemComponent"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomActuatingSystemComponent" ...>
  ...
</ActuatingSystemComponent>
```

9.9. CustomProcessSignalGeneratingSystemComponent

9.9.1 Overview

Class

A custom component of a *ProcessSignalGeneratingSystem*, i.e., a component other than a *PrimaryElement* or a *Transmitter*.



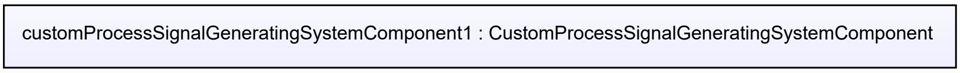
Supertypes

- *CustomAttributeOwner*
- *CustomObject*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<ProcessSignalGeneratingSystemComponent>`
RDL reference: CUSTOM PROCESS SIGNAL GENERATING SYSTEM COMPONENT
ComponentClass: CustomProcessSignalGeneratingSystemComponent
ComponentClassURI: http://sandbox.dexpi.org/rdl/CustomProcessSignalGeneratingSystemComponent

Example


customProcessSignalGeneratingSystemComponent1 : CustomProcessSignalGeneratingSystemComponent

Example: Implementation in Proteus Schema

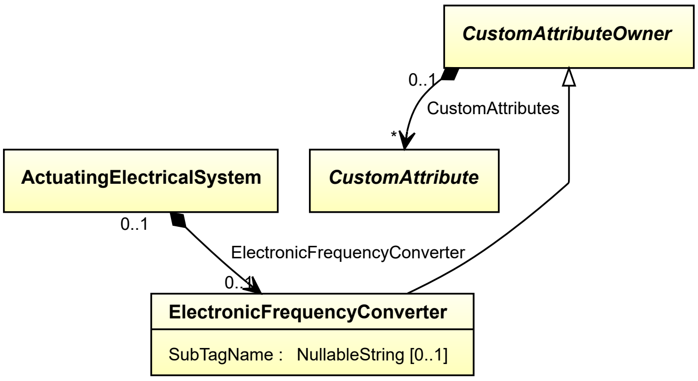
```
<ProcessSignalGeneratingSystemComponent
    ID="customProcessSignalGeneratingSystemComponent1"
    ComponentClass="CustomProcessSignalGeneratingSystemComponent"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomProcessSignalGeneratingSystemComponent" ...>
  ...
</ProcessSignalGeneratingSystemComponent>
```

9.10. ElectronicFrequencyConverter

9.10.1 Overview

Class

An electronic AC converter for changing the frequency


CustomAttributeOwner
0..1
CustomAttributes
*
ActuatingElectricalSystem
CustomAttribute
0..1
ElectronicFrequencyConverter
0..1
ElectronicFrequencyConverter
SubTagName : NullableString [0..1]

Supertypes

- *CustomAttributeOwner*

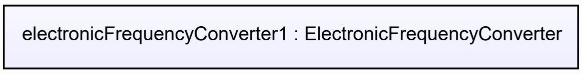
Attributes (data)

Name	Multiplicity	Type
SubTagName	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<ActuatingElectricalSystemComponent>`
RDL reference: ELECTRONIC FREQUENCY CONVERTER
ComponentClass: ElectronicFrequencyConverter
ComponentClassURI: http://sandbox.dexpi.org/rdl/ElectronicFrequencyConverter

Example


electronicFrequencyConverter1 : ElectronicFrequencyConverter

Example: Implementation in Proteus Schema

```
<ActuatingElectricalSystemComponent
    ID="electronicFrequencyConverter1"
    ComponentClass="ElectronicFrequencyConverter"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ElectronicFrequencyConverter" ...>
  ...
</ActuatingElectricalSystemComponent>
```

9.10.2 SubTagName

Attribute (data)

The sub tag name of the *ElectronicFrequencyConverter*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: SUB TAG NAME ASSIGNMENT CLASS
Name: SubTagNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/SubTagNameAssignmentClass

Example

“ST1” (*String*)

Example: Implementation in Proteus Schema

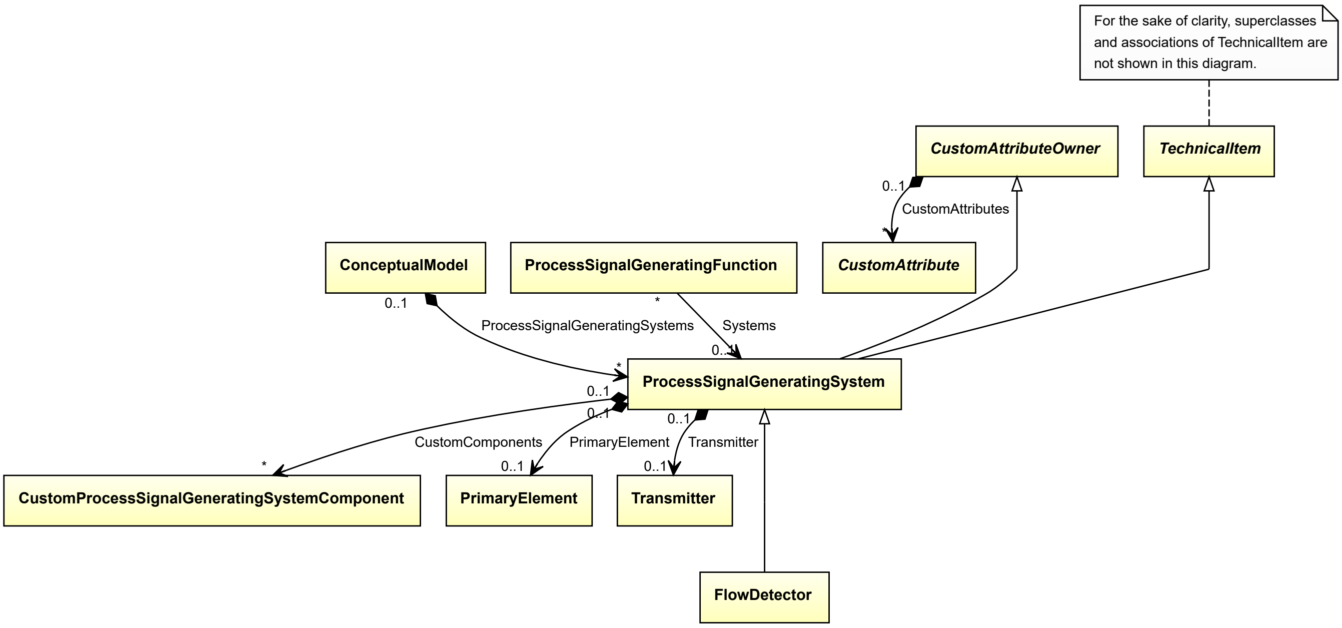
```
<ActuatingElectricalSystemComponent
    ID="electronicFrequencyConverter1"
    ComponentClass="ElectronicFrequencyConverter"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ElectronicFrequencyConverter" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="SubTagNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/SubTagNameAssignmentClass"
      Format="string"
      Value="ST1" />
    ...
  </GenericAttributes>
  ...
</ActuatingElectricalSystemComponent>
```

9.11. FlowDetector

9.11.1 Overview

Class

A detector that is intended to detect whether a fluid flow exists (from http://data.posccaesar.org/rdl/RDS1008719).


For the sake of clarity, superclasses and associations of TechnicalItem are not shown in this diagram.
CustomAttributeOwner
TechnicalItem
0..1
CustomAttributes
ConceptualModel
ProcessSignalGeneratingFunction
CustomAttribute
0..1
*
ProcessSignalGeneratingSystems
Systems
0..1
*
ProcessSignalGeneratingSystem
0..1
0..1
0..1
CustomComponents
PrimaryElement
Transmitter
*
0..1
0..1
CustomProcessSignalGeneratingSystemComponent
PrimaryElement
Transmitter
FlowDetector

Supertypes

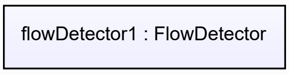
- *ProcessSignalGeneratingSystem*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<ProcessSignalGeneratingSystem>`
RDL reference: FLOW DETECTOR
ComponentClass: FlowDetector

ComponentClassURI: http://data.posccaesar.org/rdl/RDS1008719

Example



Example: Implementation in Proteus Schema

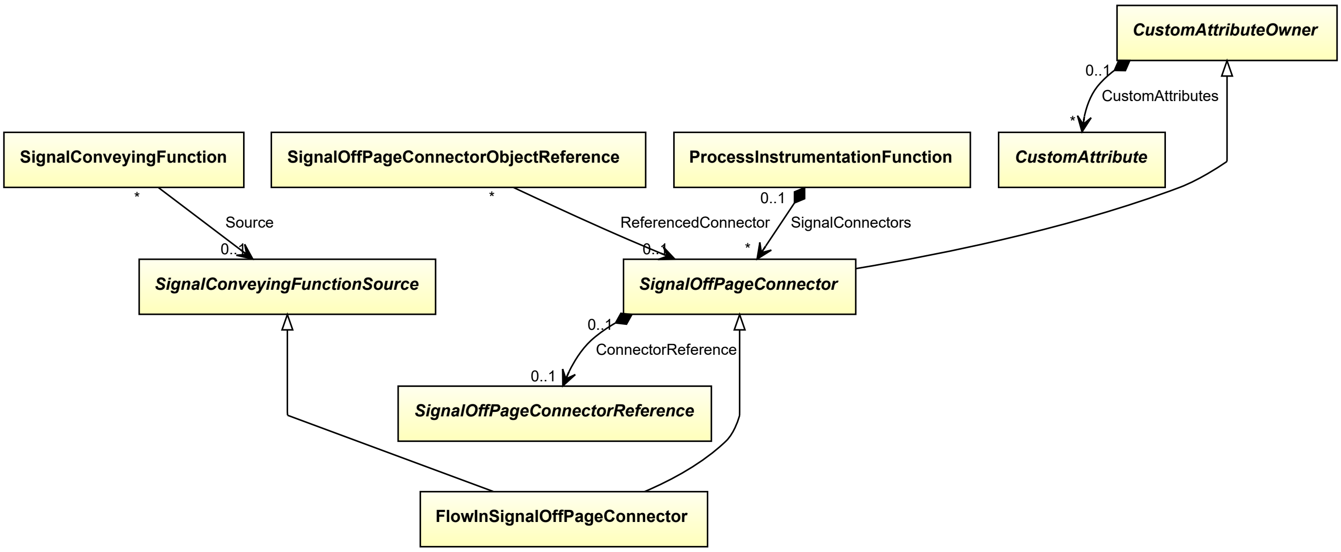
```
<ProcessSignalGeneratingSystem
    ID="flowDetector1"
    ComponentClass="FlowDetector"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS1008719" ...>
  ...
</ProcessSignalGeneratingSystem>
```

9.12. FlowInSignalOffPageConnector

9.12.1 Overview

Class

A signal connector that indicates that a preceding part of a signal conveying function is represented somewhere else, either on the same PID, or on some other PID.



Supertypes

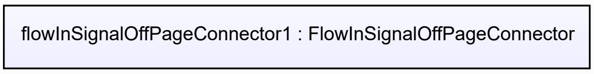
- *SignalConveyingFunctionSource*
- *SignalOffPageConnector*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<InformationFlowOffPageConnector>`
RDL reference: FLOW IN SIGNAL OFF PAGE CONNECTOR

ComponentClass: FlowInSignalOffPageConnector
ComponentClassURI: http://sandbox.dexpi.org/rdl/FlowInSignalOffPageConnector

Example


flowInSignalOffPageConnector1 : FlowInSignalOffPageConnector

Example: Implementation in Proteus Schema

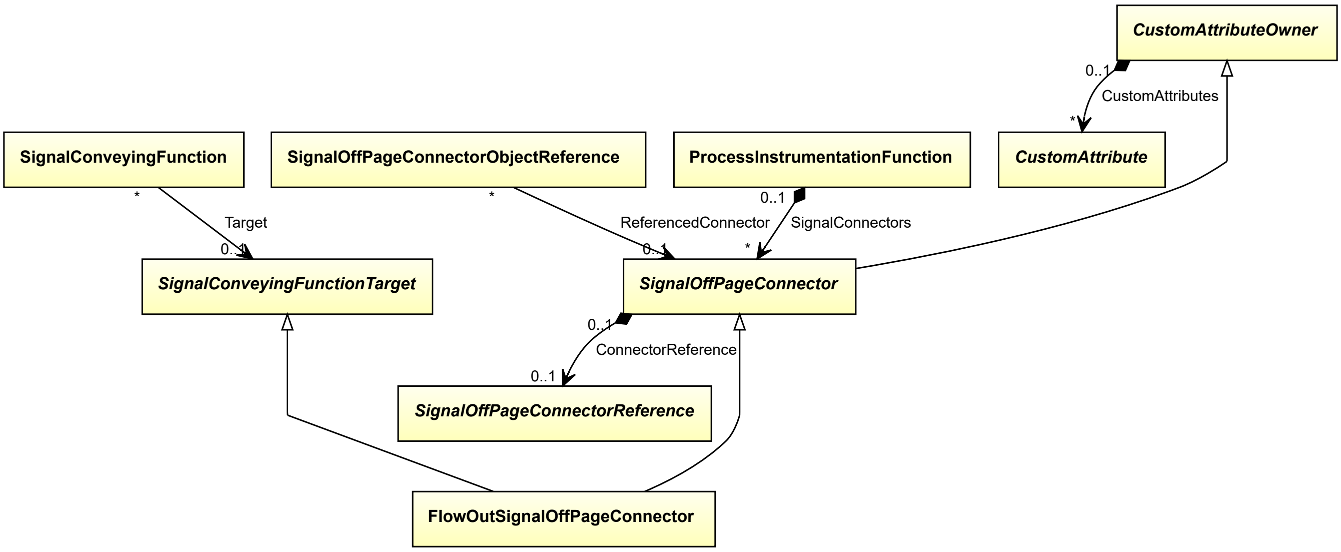
```
<InformationFlowOffPageConnector
    ID="flowInSignalOffPageConnector1"
    ComponentClass="FlowInSignalOffPageConnector"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/FlowInSignalOffPageConnector" ...>
  ...
</InformationFlowOffPageConnector>
```

9.13. FlowOutSignalOffPageConnector

9.13.1 Overview

Class

A signal connector that indicates that a subsequent part of a signal conveying function is represented somewhere else, either on the same PID, or on some other PID.



Supertypes

- *SignalConveyingFunctionTarget*
- *SignalOffPageConnector*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<InformationFlowOffPageConnector>`
RDL reference: FLOW OUT SIGNAL OFF PAGE CONNECTOR
ComponentClass: FlowOutSignalOffPageConnector
ComponentClassURI: http://sandbox.dexpi.org/rdl/FlowOutSignalOffPageConnector

Example

flowOutSignalOffPageConnector1 : FlowOutSignalOffPageConnector

Example: Implementation in Proteus Schema

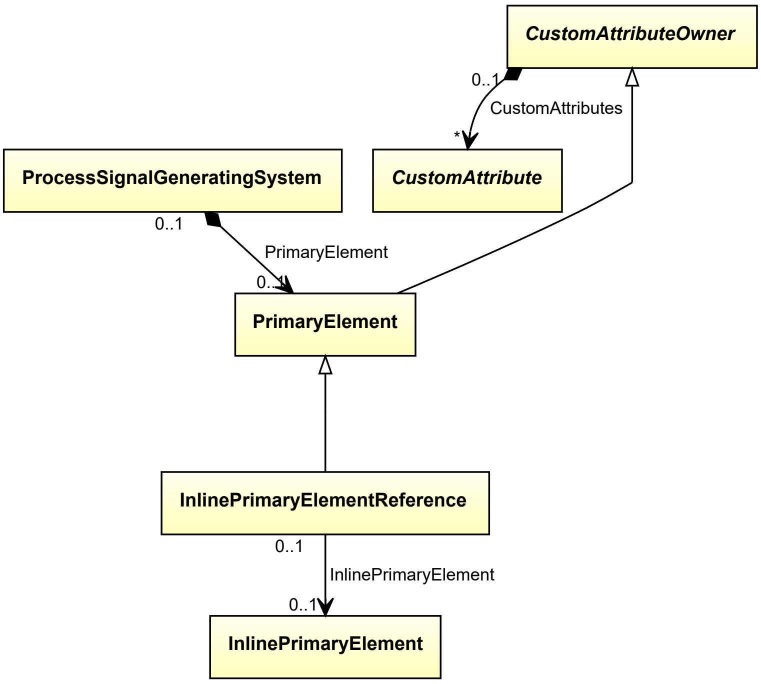
```
<InformationFlowOffPageConnector
    ID="flowOutSignalOffPageConnector1"
    ComponentClass="FlowOutSignalOffPageConnector"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/FlowOutSignalOffPageConnector" ...>
  ...
</InformationFlowOffPageConnector>
```

9.14. InlinePrimaryElementReference

9.14.1 Overview

Class

A reference to an *InlinePrimaryElement* that is part of a *PipingNetworkSegment*.



Supertypes

- *PrimaryElement*

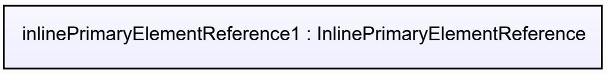
Attributes (reference)

Name	Multiplicity	Type
InlinePrimaryElement	0..1	*InlinePrimaryElement*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<ProcessSignalGeneratingSystemComponent>`
RDL reference: INLINE PRIMARY ELEMENT REFERENCE
ComponentClass: InlinePrimaryElementReference
ComponentClassURI: http://sandbox.dexpi.org/rdl/InlinePrimaryElementReference

Example


inlinePrimaryElementReference1 : InlinePrimaryElementReference

Example: Implementation in Proteus Schema

```
<ProcessSignalGeneratingSystemComponent
    ID="inlinePrimaryElementReference1"
    ComponentClass="InlinePrimaryElementReference"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/InlinePrimaryElementReference" ...>
  ...
</ProcessSignalGeneratingSystemComponent>
```

9.14.2 InlinePrimaryElement

Attribute (reference)

The *InlinePrimaryElement* referenced by the *InlinePrimaryElementReference*.

Multiplicity: 0..1

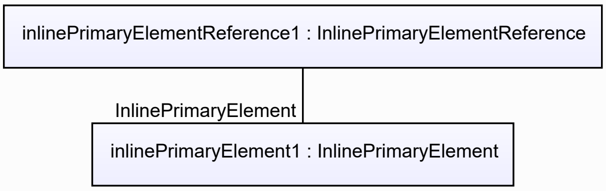
Type: *InlinePrimaryElement*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: "refers to"
Opposite association type: "is referenced by"

Example


inlinePrimaryElementReference1 : InlinePrimaryElementReference
InlinePrimaryElement
inlinePrimaryElement1 : InlinePrimaryElement

Example: Implementation in Proteus Schema

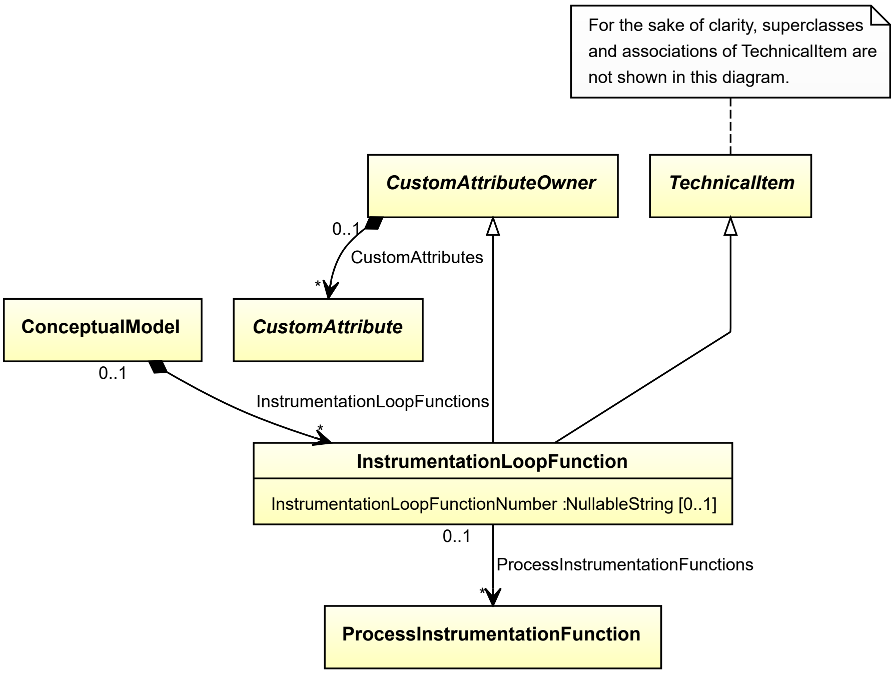
```
<ProcessSignalGeneratingSystemComponent
    ID="inlinePrimaryElementReference1"
    ComponentClass="InlinePrimaryElementReference"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/InlinePrimaryElementReference" ...>
  ...
  <Association
    Type="refers to"
    ItemID="inlinePrimaryElement1" />
  ...
<ProcessSignalGeneratingSystemComponent />
...
<PipingComponent
    ID="inlinePrimaryElement1"
    ComponentClass="InlinePrimaryElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/InlinePrimaryElement" ...>
  ...
  <Association
    Type="is referenced by"
    ItemID="inlinePrimaryElementReference1" />
  ...
<PipingComponent />
```

9.15. InstrumentationLoopFunction

9.15.1 Overview

Class

An identified collection of related *ProcessInstrumentationFunctions* that interact for a known purpose.


For the sake of clarity, superclasses and associations of TechnicalItem are not shown in this diagram.
CustomAttributeOwner
TechnicalItem
0..1
CustomAttributes
*
ConceptualModel
CustomAttribute
0..1
InstrumentationLoopFunctions
*
InstrumentationLoopFunction
InstrumentationLoopFunctionNumber :NullableString [0..1]
0..1
ProcessInstrumentationFunctions
*
ProcessInstrumentationFunction

Supertypes

- *CustomAttributeOwner*
- *TechnicalItem*

Attributes (data)

Name	Multiplicity	Type
InstrumentationLoopFunctionNumber	0..1	*NullableString*

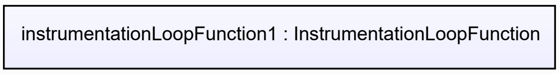
Attributes (reference)

Name	Multiplicity	Type
ProcessInstrumentationFunctions	*	*ProcessInstrumentationFunction*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<InstrumentationLoopFunction>`
RDL reference: INSTRUMENTATION LOOP FUNCTION
ComponentClass: InstrumentationLoopFunction
ComponentClassURI: http://sandbox.dexpi.org/rdl/InstrumentationLoopFunction

Example



Example: Implementation in Proteus Schema

```
<InstrumentationLoopFunction
    ID="instrumentationLoopFunction1"
    ComponentClass="InstrumentationLoopFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/InstrumentationLoopFunction" ...>
  ...
</InstrumentationLoopFunction>
```

9.15.2 InstrumentationLoopFunctionNumber

Attribute (data)

The identification number of the *InstrumentationLoopFunction*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: INSTRUMENTATION LOOP FUNCTION NUMBER ASSIGNMENT CLASS
Name: InstrumentationLoopFunctionNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/InstrumentationLoopFunctionNumberAssignmentClass

Example

"4750.01" (*String*)

Example: Implementation in Proteus Schema

```
<InstrumentationLoopFunction
    ID="instrumentationLoopFunction1"
    ComponentClass="InstrumentationLoopFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/InstrumentationLoopFunction" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="InstrumentationLoopFunctionNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/InstrumentationLoopFunctionNumberAssignmentClass"
      Format="string"
      Value="4750.01" />
    ...
  </GenericAttributes>
  ...
</InstrumentationLoopFunction>
```

9.15.3 ProcessInstrumentationFunctions

Attribute (reference)

The *ProcessInstrumentationFunctions* that constitute this *InstrumentationLoopFunction*.

Multiplicity: *

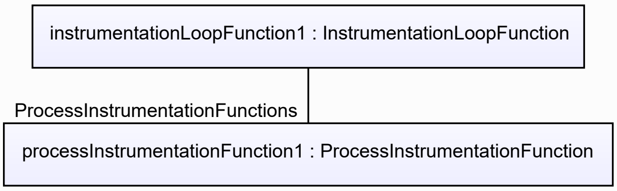
Type: *ProcessInstrumentationFunction*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: "is a collection including"
Opposite association type: "is a part of"

Example


instrumentationLoopFunction1 : InstrumentationLoopFunction
ProcessInstrumentationFunctions
processInstrumentationFunction1 : ProcessInstrumentationFunction

Example: Implementation in Proteus Schema

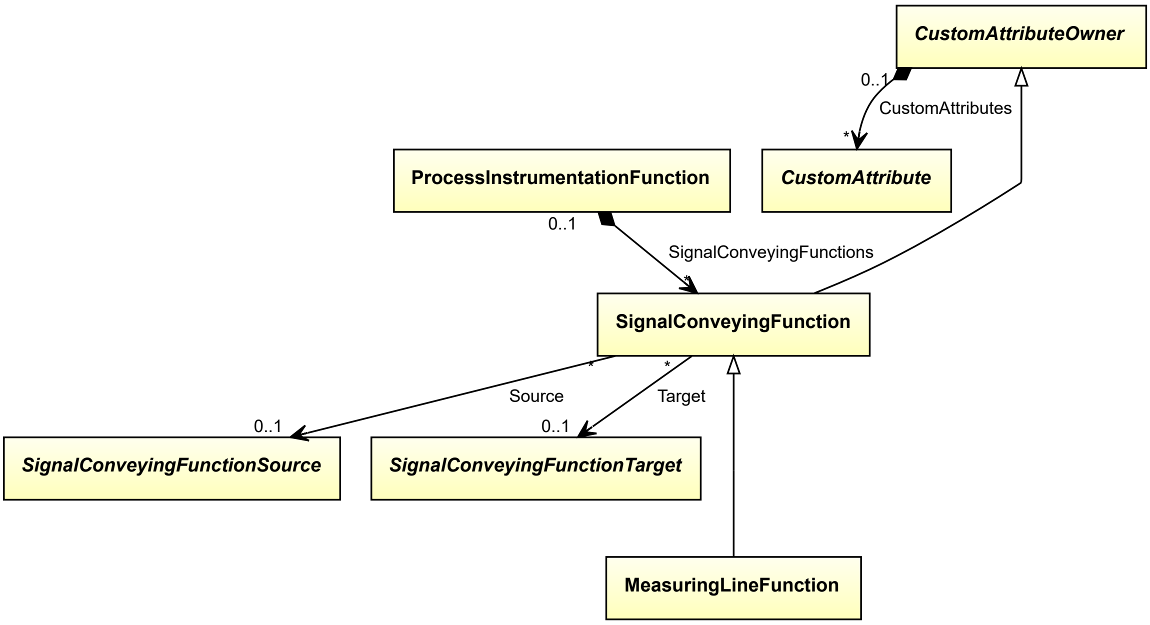
```
<InstrumentationLoopFunction
    ID="instrumentationLoopFunction1"
    ComponentClass="InstrumentationLoopFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/InstrumentationLoopFunction" ...>
  ...
  <Association
    Type="is a collection including"
    ItemID="processInstrumentationFunction1" />
  ...
<InstrumentationLoopFunction />
...
<ProcessInstrumentationFunction
    ID="processInstrumentationFunction1"
    ComponentClass="ProcessInstrumentationFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessInstrumentationFunction" ...>
  ...
  <Association
    Type="is a part of"
    ItemID="instrumentationLoopFunction1" />
  ...
<ProcessInstrumentationFunction />
```

9.16. MeasuringLineFunction

9.16.1 Overview

Class

Information flow function for measured values.



Supertypes

- *SignalConveyingFunction*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<InformationFlow>`
RDL reference: MEASURING LINE FUNCTION
ComponentClass: MeasuringLineFunction
ComponentClassURI: http://sandbox.dexpi.org/rdl/MeasuringLineFunction

Example

Example: Implementation in Proteus Schema

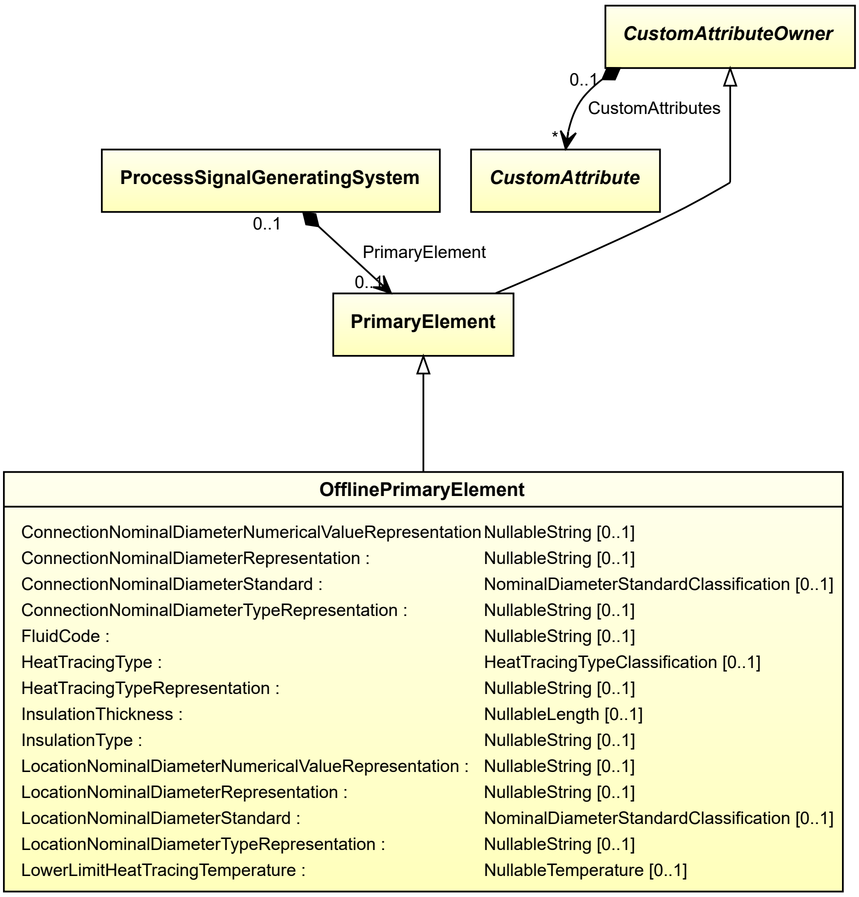
```
<InformationFlow
    ID="measuringLineFunction1"
    ComponentClass="MeasuringLineFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/MeasuringLineFunction" ...>
  ...
</InformationFlow>
```

9.17. OfflinePrimaryElement

9.17.1 Overview

Class

A *PrimaryElement* that is not part of a *PipingNetworkSegment*.


CustomAttributeOwner
0..1
CustomAttributes
*
ProcessSignalGeneratingSystem
CustomAttribute
0..1
PrimaryElement
0..1
PrimaryElement
OfflinePrimaryElement
ConnectionNominalDiameterNumericalValueRepresentation NullableString [0..1]
ConnectionNominalDiameterRepresentation : NullableString [0..1]
ConnectionNominalDiameterStandard : NominalDiameterStandardClassification [0..1]
ConnectionNominalDiameterTypeRepresentation : NullableString [0..1]
FluidCode : NullableString [0..1]
HeatTracingType : HeatTracingTypeClassification [0..1]
HeatTracingTypeRepresentation : NullableString [0..1]
InsulationThickness : NullableLength [0..1]
InsulationType : NullableString [0..1]
LocationNominalDiameterNumericalValueRepresentation : NullableString [0..1]
LocationNominalDiameterRepresentation : NullableString [0..1]
LocationNominalDiameterStandard : NominalDiameterStandardClassification [0..1]
LocationNominalDiameterTypeRepresentation : NullableString [0..1]
LowerLimitHeatTracingTemperature : NullableTemperature [0..1]

Supertypes

- *PrimaryElement*

Attributes (data)

Name	Multiplicity	Type
ConnectionNominalDiameterNumericalValueRepresentation	0..1	*NullableString*
ConnectionNominalDiameterRepresentation	0..1	*NullableString*
ConnectionNominalDiameterStandard	0..1	*NominalDiameterStandardClassification*
ConnectionNominalDiameterTypeRepresentation	0..1	*NullableString*
FluidCode	0..1	*NullableString*
HeatTracingType	0..1	*HeatTracingTypeClassification*
HeatTracingTypeRepresentation	0..1	*NullableString*
InsulationThickness	0..1	*NullableLength*
InsulationType	0..1	*NullableString*
LocationNominalDiameterNumericalValueRepresentation	0..1	*NullableString*
LocationNominalDiameterRepresentation	0..1	*NullableString*
LocationNominalDiameterStandard	0..1	*NominalDiameterStandardClassification*
LocationNominalDiameterTypeRepresentation	0..1	*NullableString*
LowerLimitHeatTracingTemperature	0..1	*NullableTemperature*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<ProcessSignalGeneratingSystemComponent>`
RDL reference: OFFLINE PRIMARY ELEMENT
ComponentClass: OfflinePrimaryElement
ComponentClassURI: http://sandbox.dexpi.org/rdl/OfflinePrimaryElement

Example

Example: Implementation in Proteus Schema

```
<ProcessSignalGeneratingSystemComponent
    ID="offlinePrimaryElement1"
    ComponentClass="OfflinePrimaryElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/OfflinePrimaryElement" ...>
  ...
</ProcessSignalGeneratingSystemComponent>
```

9.17.2 ConnectionNominalDiameterNumericalValueRepresentation

Attribute (data)

A readable representation of the numerical value of the nominal diameter at the device connection of the *OfflinePrimaryElement*. The purpose of this value is to give a textual representation of the nominal diameter to be used in the graphics of a PID.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: CONNECTION NOMINAL DIAMETER NUMERICAL VALUE REPRESENTATION ASSIGNMENT CLASS
Name: ConnectionNominalDiameterNumericalValueRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ConnectionNominalDiameterNumericalValueRepresentationAssignmentClass

Example

“25” (*String*)

Example: Implementation in Proteus Schema

```
<ProcessSignalGeneratingSystemComponent
    ID="offlinePrimaryElement1"
    ComponentClass="OfflinePrimaryElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/OfflinePrimaryElement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ConnectionNominalDiameterNumericalValueRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/
↪ConnectionNominalDiameterNumericalValueRepresentationAssignmentClass"
      Format="string"
      Value="25" />
    ...
  </GenericAttributes>
  ...
</ProcessSignalGeneratingSystemComponent>
```

9.17.3 ConnectionNominalDiameterRepresentation

Attribute (data)

A readable representation of the nominal diameter at the device connection of the *OfflinePrimaryElement*. The purpose of this value is to give a textual representation of the nominal diameter to be used in the graphics of a PID.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: CONNECTION NOMINAL DIAMETER REPRESENTATION ASSIGNMENT CLASS
Name: ConnectionNominalDiameterRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ConnectionNominalDiameterRepresentationAssignmentClass

Example

"DN 25" (*String*)

Example: Implementation in Proteus Schema

```
<ProcessSignalGeneratingSystemComponent
    ID="offlinePrimaryElement1"
    ComponentClass="OfflinePrimaryElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/OfflinePrimaryElement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ConnectionNominalDiameterRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ConnectionNominalDiameterRepresentationAssignmentClass"
      Format="string"
      Value="DN 25" />
    ...
  </GenericAttributes>
  ...
</ProcessSignalGeneratingSystemComponent>
```

9.17.4 ConnectionNominalDiameterStandard

Attribute (data)

The nominal diameter of the device connection of the *OfflinePrimaryElement*, given as a reference to a nominal diameter standard and value.

Multiplicity: 0..1

Type: *NominalDiameterStandardClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: CONNECTION NOMINAL DIAMETER STANDARD SPECIALIZATION
Name: ConnectionNominalDiameterStandardSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/ConnectionNominalDiameterStandardSpecialization

Example

DN 25 (DIN 2448) (*NominalDiameterStandardClassification::Din2448ObjectDn25*)

Example: Implementation in Proteus Schema

```
<ProcessSignalGeneratingSystemComponent
    ID="offlinePrimaryElement1"
    ComponentClass="OfflinePrimaryElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/OfflinePrimaryElement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ConnectionNominalDiameterStandardSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/ConnectionNominalDiameterStandardSpecialization"
      Format="anyURI"
      Value="Din2448ObjectDn25"
      ValueURI="http://sandbox.dexpi.org/rdl/Din2448ObjectDn25" />
    ...
  </GenericAttributes>
  ...
</ProcessSignalGeneratingSystemComponent>
```

9.17.5 ConnectionNominalDiameterTypeRepresentation

Attribute (data)

A readable representation of the type of the nominal diameter at the device connection of the *OfflinePrimaryElement*. The purpose of this value is to give a textual representation of the nominal diameter to be used in the graphics of a PID.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: CONNECTION NOMINAL DIAMETER TYPE REPRESENTATION ASSIGNMENT CLASS
Name: ConnectionNominalDiameterTypeRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ConnectionNominalDiameterTypeRepresentationAssignmentClass

Example

“DN” (*String*)

Example: Implementation in Proteus Schema

```
<ProcessSignalGeneratingSystemComponent
    ID="offlinePrimaryElement1"
    ComponentClass="OfflinePrimaryElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/OfflinePrimaryElement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ConnectionNominalDiameterTypeRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ConnectionNominalDiameterTypeRepresentationAssignmentClass"
      Format="string"
      Value="DN" />
    ...
  </GenericAttributes>
  ...
</ProcessSignalGeneratingSystemComponent>
```

9.17.6 FluidCode

Attribute (data)

The identification code of the fluid related to the *OfflinePrimaryElement*. So far, DEXPI does not define restrictions for valid values.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: FLUID CODE ASSIGNMENT CLASS
Name: FluidCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/FluidCodeAssignmentClass

Example

“MNb” (*String*)

Example: Implementation in Proteus Schema

```
<ProcessSignalGeneratingSystemComponent
    ID="offlinePrimaryElement1"
    ComponentClass="OfflinePrimaryElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/OfflinePrimaryElement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="FluidCodeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/FluidCodeAssignmentClass"
      Format="string"
      Value="MNb" />
    ...
  </GenericAttributes>
  ...
</ProcessSignalGeneratingSystemComponent>
```

9.17.7 HeatTracingType

Attribute (data)

A specialization indicating the heat tracing type related to the *OfflinePrimaryElement*.

Multiplicity: 0..1

Type: *HeatTracingTypeClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: HEAT TRACING TYPE SPECIALIZATION
Name: HeatTracingTypeSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/HeatTracingTypeSpecialization

Example

electrical heat tracing system (*HeatTracingTypeClassification::ElectricalHeatTracingSystem*)

Example: Implementation in Proteus Schema

```
<ProcessSignalGeneratingSystemComponent
    ID="offlinePrimaryElement1"
    ComponentClass="OfflinePrimaryElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/OfflinePrimaryElement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="HeatTracingTypeSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/HeatTracingTypeSpecialization"
      Format="anyURI"
      Value="ElectricalHeatTracingSystem"
      ValueURI="http://data.posccaesar.org/rdl/RDS11854600" />
    ...
  </GenericAttributes>
  ...
</ProcessSignalGeneratingSystemComponent>
```

9.17.8 HeatTracingTypeRepresentation

Attribute (data)

The heat tracing type related to the *OfflinePrimaryElement*, represented as a string.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: HEAT TRACING TYPE REPRESENTATION ASSIGNMENT CLASS
Name: HeatTracingTypeRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/HeatTracingTypeRepresentationAssignmentClass

Example

"E" (*String*)

Example: Implementation in Proteus Schema

```
<ProcessSignalGeneratingSystemComponent
    ID="offlinePrimaryElement1"
    ComponentClass="OfflinePrimaryElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/OfflinePrimaryElement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="HeatTracingTypeRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/HeatTracingTypeRepresentationAssignmentClass"
      Format="string"
      Value="E" />
    ...
  </GenericAttributes>
  ...
</ProcessSignalGeneratingSystemComponent>
```

9.17.9 InsulationThickness

Attribute (data)

The insulation thickness of the *OfflinePrimaryElement*.

Multiplicity: 0..1

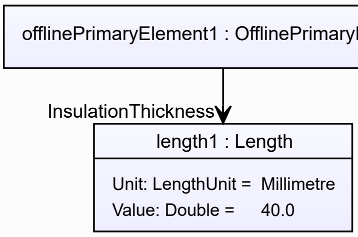
Type: *NullableLength*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: INSULATION THICKNESS
Name: InsulationThickness
AttributeURI: http://data.posccaesar.org/rdl/RDS4238040

Example

The instance offlinePrimaryElement1 represents an *OfflinePrimaryElement* with an *InsulationThickness* of 40.0 mm.


offlinePrimaryElement1 : OfflinePrimaryElement
InsulationThickness
length1 : Length
Unit: LengthUnit = Millimetre
Value: Double = 40.0

Example: Implementation in Proteus Schema

```
<ProcessSignalGeneratingSystemComponent
    ID="offlinePrimaryElement1"
    ComponentClass="OfflinePrimaryElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/OfflinePrimaryElement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="InsulationThickness"
      AttributeURI="http://data.posccaesar.org/rdl/RDS4238040"
      Format="double"
      Value="40.0"
      Units="Millimetre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1357739" />
    ...
  </GenericAttributes>
  ...
</ProcessSignalGeneratingSystemComponent>
```

9.17.10 InsulationType

Attribute (data)

The identification code for the insulation type related to the *OfflinePrimaryElement*. So far, DEXPI does not define restrictions for valid values.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: INSULATION TYPE ASSIGNMENT CLASS
Name: InsulationTypeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/InsulationTypeAssignmentClass

Example

“Q” (*String*)

Example: Implementation in Proteus Schema

```
<ProcessSignalGeneratingSystemComponent
    ID="offlinePrimaryElement1"
    ComponentClass="OfflinePrimaryElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/OfflinePrimaryElement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="InsulationTypeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/InsulationTypeAssignmentClass"
      Format="string"
      Value="Q" />
    ...
  </GenericAttributes>
  ...
</ProcessSignalGeneratingSystemComponent>
```

9.17.11 LocationNominalDiameterNumericalValueRepresentation

Attribute (data)

A readable representation of the numerical value of the nominal diameter at the location of the *OfflinePrimaryElement*. The purpose of this value is to give a textual representation of the nominal diameter to be used in the graphics of a PID.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: LOCATION NOMINAL DIAMETER NUMERICAL VALUE REPRESENTATION ASSIGNMENT CLASS
Name: LocationNominalDiameterNumericalValueRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/LocationNominalDiameterNumericalValueRepresentationAssignmentClass

Example

“25” (*String*)

Example: Implementation in Proteus Schema

```
<ProcessSignalGeneratingSystemComponent
    ID="offlinePrimaryElement1"
    ComponentClass="OfflinePrimaryElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/OfflinePrimaryElement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="LocationNominalDiameterNumericalValueRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/
↪LocationNominalDiameterNumericalValueRepresentationAssignmentClass"
      Format="string"
      Value="25" />
    ...
  </GenericAttributes>
  ...
</ProcessSignalGeneratingSystemComponent>
```

9.17.12 LocationNominalDiameterRepresentation

Attribute (data)

A readable representation of the nominal diameter at the location of the *OfflinePrimaryElement*. The purpose of this value is to give a textual representation of the nominal diameter to be used in the graphics of a PID.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: LOCATION NOMINAL DIAMETER REPRESENTATION ASSIGNMENT CLASS
Name: LocationNominalDiameterRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/LocationNominalDiameterRepresentationAssignmentClass

Example

“DN 25” (*String*)

Example: Implementation in Proteus Schema

```
<ProcessSignalGeneratingSystemComponent
    ID="offlinePrimaryElement1"
    ComponentClass="OfflinePrimaryElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/OfflinePrimaryElement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="LocationNominalDiameterRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/LocationNominalDiameterRepresentationAssignmentClass"
      Format="string"
      Value="DN 25" />
    ...
  </GenericAttributes>
  ...
</ProcessSignalGeneratingSystemComponent>
```

9.17.13 LocationNominalDiameterStandard

Attribute (data)

The nominal diameter of the location of the *OfflinePrimaryElement*, given as a reference to a nominal diameter standard and value.

Multiplicity: 0..1

Type: *NominalDiameterStandardClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: LOCATION NOMINAL DIAMETER STANDARD SPECIALIZATION
Name: LocationNominalDiameterStandardSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/LocationNominalDiameterStandardSpecialization

Example

DN 25 (DIN 2448) (*NominalDiameterStandardClassification::Din2448ObjectDn25*)

Example: Implementation in Proteus Schema

```
<ProcessSignalGeneratingSystemComponent
    ID="offlinePrimaryElement1"
    ComponentClass="OfflinePrimaryElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/OfflinePrimaryElement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="LocationNominalDiameterStandardSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/LocationNominalDiameterStandardSpecialization"
      Format="anyURI"
      Value="Din2448ObjectDn25"
      ValueURI="http://sandbox.dexpi.org/rdl/Din2448ObjectDn25" />
    ...
  </GenericAttributes>
  ...
</ProcessSignalGeneratingSystemComponent>
```

9.17.14 LocationNominalDiameterTypeRepresentation

Attribute (data)

A readable representation of the type of the nominal diameter at the location of the *OfflinePrimaryElement*. The purpose of this value is to give a textual representation of the nominal diameter to be used in the graphics of a PID.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: LOCATION NOMINAL DIAMETER TYPE REPRESENTATION ASSIGNMENT CLASS
Name: LocationNominalDiameterTypeRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/LocationNominalDiameterTypeRepresentationAssignmentClass

Example

“DN” (*String*)

Example: Implementation in Proteus Schema

```
<ProcessSignalGeneratingSystemComponent
    ID="offlinePrimaryElement1"
    ComponentClass="OfflinePrimaryElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/OfflinePrimaryElement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="LocationNominalDiameterTypeRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/LocationNominalDiameterTypeRepresentationAssignmentClass"
      Format="string"
      Value="DN" />
    ...
  </GenericAttributes>
  ...
</ProcessSignalGeneratingSystemComponent>
```

9.17.15 LowerLimitHeatTracingTemperature

Attribute (data)

The lower limit for the temperature that a heat tracing system must ensure for the *OfflinePrimaryElement*.

Multiplicity: 0..1

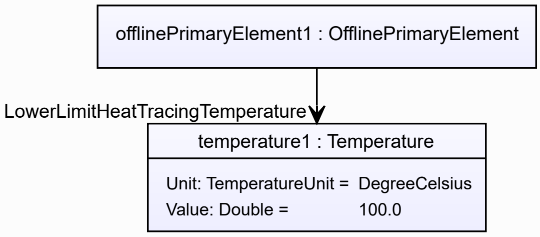
Type: *NullableTemperature*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for physical quantities*.
RDL reference: LOWER LIMIT HEAT TRACING TEMPERATURE
Name: LowerLimitHeatTracingTemperature
AttributeURI: http://sandbox.dexpi.org/rdl/LowerLimitHeatTracingTemperature

Example

The instance offlinePrimaryElement1 represents an *OfflinePrimaryElement* with a *LowerLimitHeatTracingTemperature* of 100.0 °C.


offlinePrimaryElement1 : OfflinePrimaryElement
LowerLimitHeatTracingTemperature
temperature1 : Temperature
Unit: TemperatureUnit = DegreeCelsius
Value: Double = 100.0

Example: Implementation in Proteus Schema

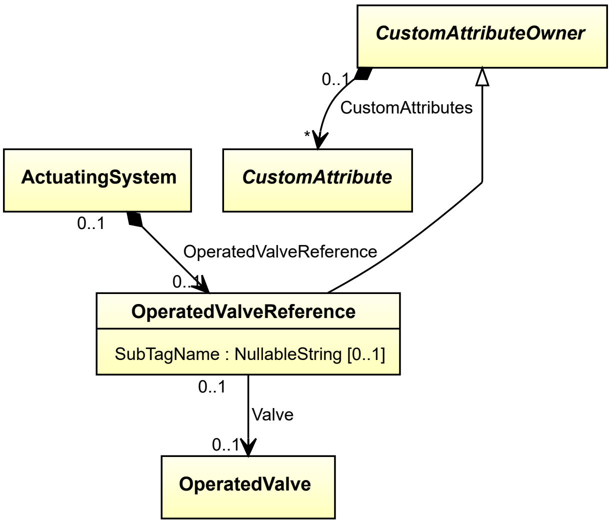
```
<ProcessSignalGeneratingSystemComponent
    ID="offlinePrimaryElement1"
    ComponentClass="OfflinePrimaryElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/OfflinePrimaryElement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="LowerLimitHeatTracingTemperature"
      AttributeURI="http://sandbox.dexpi.org/rdl/LowerLimitHeatTracingTemperature"
      Format="double"
      Value="100.0"
      Units="DegreeCelsius"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1322684" />
    ...
  </GenericAttributes>
  ...
</ProcessSignalGeneratingSystemComponent>
```

9.18. OperatedValveReference

9.18.1 Overview

Class

A reference to an *OperatedValve*.



Supertypes

- *CustomAttributeOwner*

Attributes (data)

Name	Multiplicity	Type
SubTagName	0..1	*NullableString*

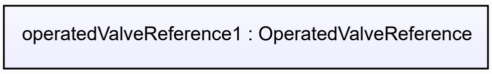
Attributes (reference)

Name	Multiplicity	Type
Valve	0..1	*OperatedValve*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<ActuatingSystemComponent>`
RDL reference: OPERATED VALVE REFERENCE
ComponentClass: OperatedValveReference
ComponentClassURI: http://sandbox.dexpi.org/rdl/OperatedValveReference

Example



Example: Implementation in Proteus Schema

```
<ActuatingSystemComponent
    ID="operatedValveReference1"
    ComponentClass="OperatedValveReference"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/OperatedValveReference" ...>
  ...
</ActuatingSystemComponent>
```

9.18.2 SubTagName

Attribute (data)

The sub tag name of the *OperatedValveReference*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: SUB TAG NAME ASSIGNMENT CLASS
Name: SubTagNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/SubTagNameAssignmentClass

Example

“ST1” (*String*)

Example: Implementation in Proteus Schema

```
<ActuatingSystemComponent
    ID="operatedValveReference1"
    ComponentClass="OperatedValveReference"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/OperatedValveReference" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="SubTagNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/SubTagNameAssignmentClass"
      Format="string"
      Value="ST1" />
    ...
  </GenericAttributes>
  ...
</ActuatingSystemComponent>
```

9.18.3 Valve

Attribute (reference)

The actual valve referenced by the *OperatedValveReference*.

Multiplicity: 0..1

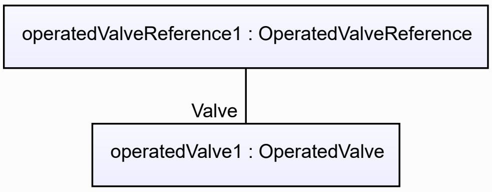
Type: *OperatedValve*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: `"refers to"`
Opposite association type: `"is referenced by"`

Example


operatedValveReference1 : OperatedValveReference
Valve
operatedValve1 : OperatedValve

Example: Implementation in Proteus Schema

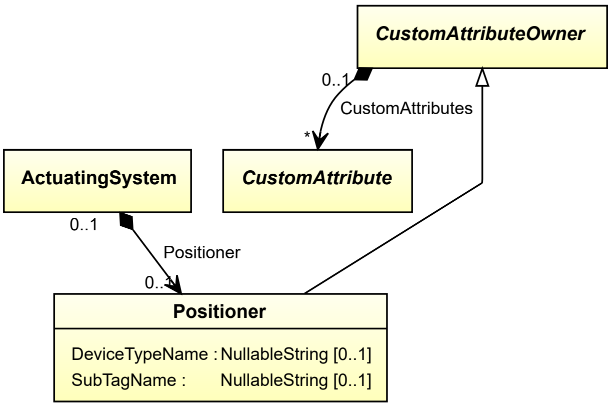
```
<ActuatingSystemComponent
    ID="operatedValveReference1"
    ComponentClass="OperatedValveReference"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/OperatedValveReference" ...>
  ...
  <Association
    Type="refers to"
    ItemID="operatedValve1" />
  ...
<ActuatingSystemComponent />
...
<PipingComponent
    ID="operatedValve1"
    ComponentClass="OperatedValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11141590" ...>
  ...
  <Association
    Type="is referenced by"
    ItemID="operatedValveReference1" />
  ...
<PipingComponent />
```

9.19. Positioner

9.19.1 Overview

Class

A positioner.



Supertypes

- *CustomAttributeOwner*

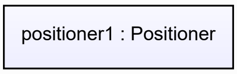
Attributes (data)

Name	Multiplicity	Type
DeviceTypeName	0..1	*NullableString*
SubTagName	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<ActuatingSystemComponent>`
RDL reference: POSITIONER
ComponentClass: Positioner
ComponentClassURI: http://sandbox.dexpi.org/rdl/Positioner

Example



Example: Implementation in Proteus Schema

```
<ActuatingSystemComponent
    ID="positioner1"
    ComponentClass="Positioner"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Positioner" ...>
  ...
</ActuatingSystemComponent>
```

9.19.2 DeviceTypeName

Attribute (data)

The device type of the *Positioner*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: DEVICE TYPE NAME ASSIGNMENT CLASS
Name: DeviceTypeNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/DeviceTypeNameAssignmentClass

Example

“pressure transmitter” (*String*)

Example: Implementation in Proteus Schema

```
<ActuatingSystemComponent
    ID="positioner1"
    ComponentClass="Positioner"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Positioner" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DeviceTypeNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/DeviceTypeNameAssignmentClass"
      Format="string"
      Value="pressure transmitter" />
    ...
  </GenericAttributes>
  ...
</ActuatingSystemComponent>
```

9.19.3 SubTagName

Attribute (data)

The sub tag name of the *Positioner*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: SUB TAG NAME ASSIGNMENT CLASS
Name: SubTagNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/SubTagNameAssignmentClass

Example

“ST1” (*String*)

Example: Implementation in Proteus Schema

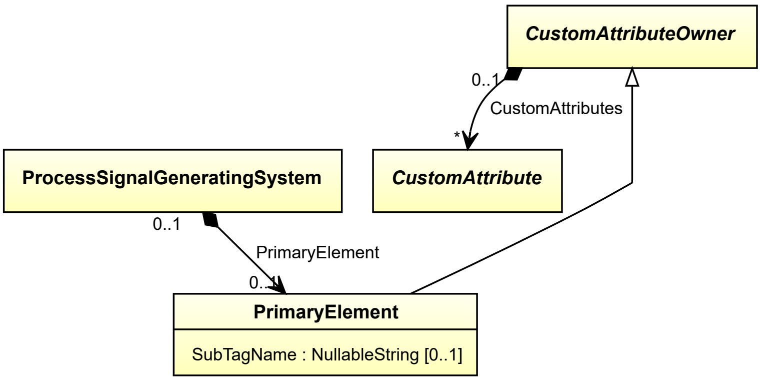
```
<ActuatingSystemComponent
    ID="positioner1"
    ComponentClass="Positioner"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Positioner" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="SubTagNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/SubTagNameAssignmentClass"
      Format="string"
      Value="ST1" />
    ...
  </GenericAttributes>
  ...
</ActuatingSystemComponent>
```

9.20. PrimaryElement

9.20.1 Overview

Class

An artefact that converts the input variable into a signal suitable for measurement.



Supertypes

- *CustomAttributeOwner*

Subtypes

- *InlinePrimaryElementReference*
- *OfflinePrimaryElement*

Attributes (data)

Name	Multiplicity	Type
SubTagName	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<ProcessSignalGeneratingSystemComponent>`
RDL reference: PRIMARY ELEMENT
ComponentClass: PrimaryElement
ComponentClassURI: http://sandbox.dexpi.org/rdl/PrimaryElement

Example

Example: Implementation in Proteus Schema

```
<ProcessSignalGeneratingSystemComponent
    ID="primaryElement1"
    ComponentClass="PrimaryElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PrimaryElement" ...>
  ...
</ProcessSignalGeneratingSystemComponent>
```

9.20.2 SubTagName

Attribute (data)

The sub tag name of the *PrimaryElement*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: SUB TAG NAME ASSIGNMENT CLASS
Name: SubTagNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/SubTagNameAssignmentClass

Example

"ST1" (*String*)

Example: Implementation in Proteus Schema

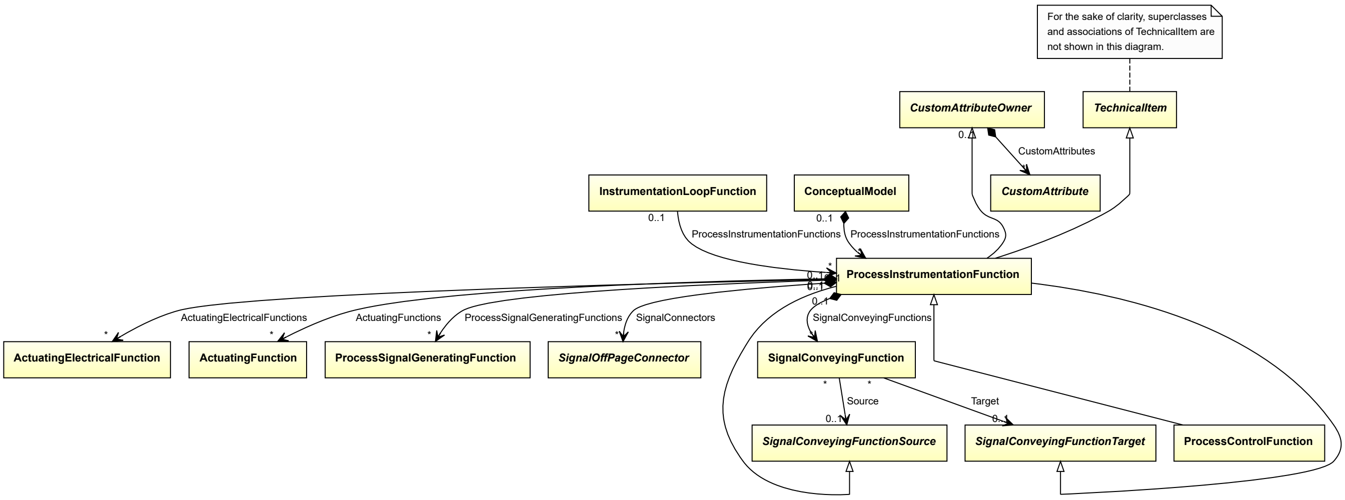
```
<ProcessSignalGeneratingSystemComponent
    ID="primaryElement1"
    ComponentClass="PrimaryElement"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PrimaryElement" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="SubTagNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/SubTagNameAssignmentClass"
      Format="string"
      Value="ST1" />
    ...
  </GenericAttributes>
  ...
</ProcessSignalGeneratingSystemComponent>
```

9.21. ProcessControlFunction

9.21.1 Overview

Class

A requirement for control structures relating to Process Engineering.


For the sake of clarity, superclasses and associations of TechnicalItem are not shown in this diagram.
CustomAttributeOwner
TechnicalItem
CustomAttributes
InstrumentationLoopFunction
ConceptualModel
CustomAttribute
0..1
0..1
ProcessInstrumentationFunctions
ProcessInstrumentationFunctions
ProcessInstrumentationFunction
ActuatingElectricalFunctions
ActuatingFunctions
ProcessSignalGeneratingFunctions
SignalConnectors
SignalConveyingFunctions
ActuatingElectricalFunction
ActuatingFunction
ProcessSignalGeneratingFunction
SignalOffPageConnector
SignalConveyingFunction
Source
Target
SignalConveyingFunctionSource
SignalConveyingFunctionTarget
ProcessControlFunction

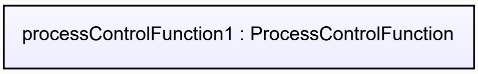
Supertypes

- *ProcessInstrumentationFunction*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<ProcessInstrumentationFunction>`
RDL reference: PROCESS CONTROL FUNCTION
ComponentClass: ProcessControlFunction
ComponentClassURI: http://sandbox.dexpi.org/rdl/ProcessControlFunction

Example



Example: Implementation in Proteus Schema

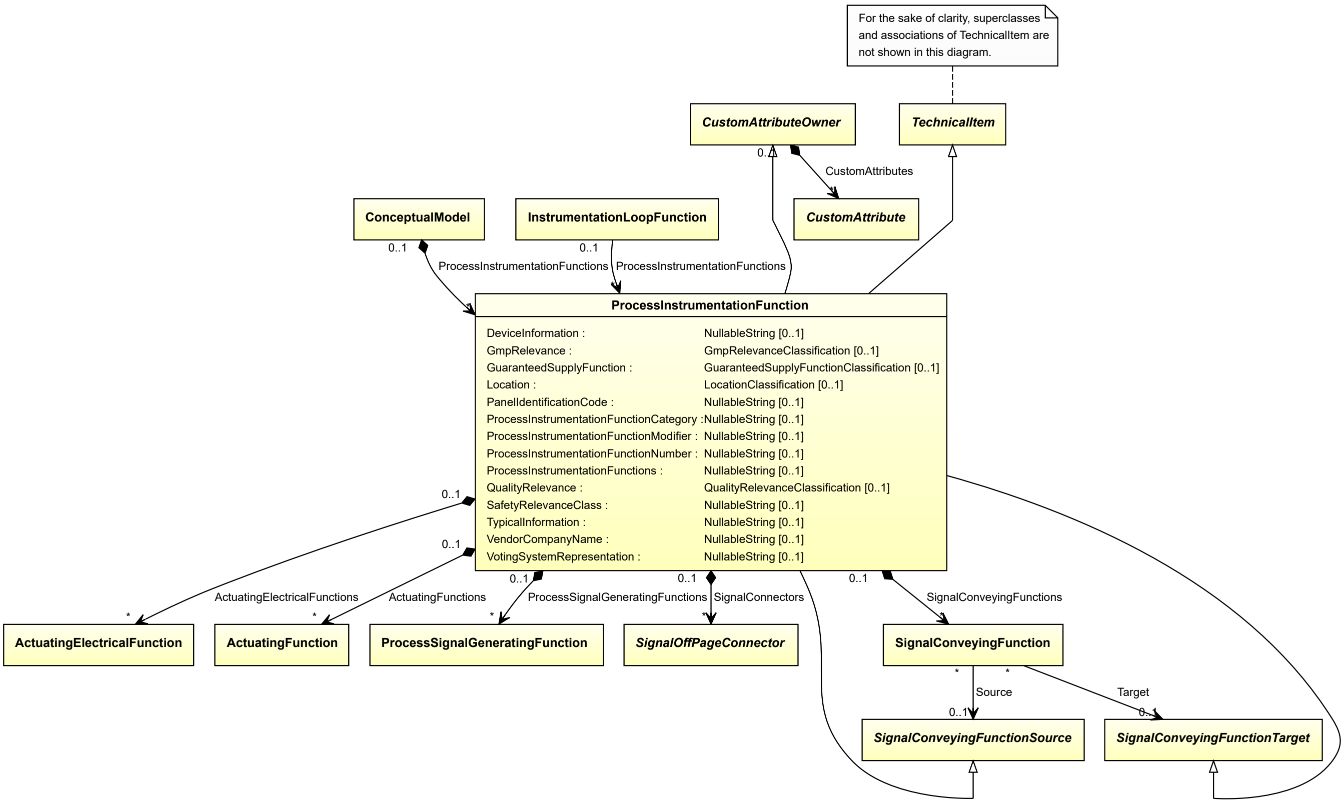
```
<ProcessInstrumentationFunction
    ID="processControlFunction1"
    ComponentClass="ProcessControlFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessControlFunction" ...>
  ...
</ProcessInstrumentationFunction>
```

9.22. ProcessInstrumentationFunction

9.22.1 Overview

Class

A requirement for instrumentation and/or control structures relating to Process Engineering.


For the sake of clarity, superclasses and associations of TechnicalItem are not shown in this diagram.
CustomAttributeOwner
TechnicalItem
0..1
CustomAttributes
CustomAttribute
ConceptualModel
InstrumentationLoopFunction
0..1
ProcessInstrumentationFunctions
0..1
ProcessInstrumentationFunctions
ProcessInstrumentationFunction
DeviceInformation : NullableString [0..1]
GmpRelevance : GmpRelevanceClassification [0..1]
GuaranteedSupplyFunction : GuaranteedSupplyFunctionClassification [0..1]
Location : LocationClassification [0..1]
PanelIdentificationCode : NullableString [0..1]
ProcessInstrumentationFunctionCategory : NullableString [0..1]
ProcessInstrumentationFunctionModifier : NullableString [0..1]
ProcessInstrumentationFunctionNumber : NullableString [0..1]
ProcessInstrumentationFunctions : NullableString [0..1]
QualityRelevance : QualityRelevanceClassification [0..1]
SafetyRelevanceClass : NullableString [0..1]
TypicalInformation : NullableString [0..1]
VendorCompanyName : NullableString [0..1]
VotingSystemRepresentation : NullableString [0..1]
0..1
0..1
0..1
0..1
0..1
ActuatingElectricalFunctions
ActuatingFunctions
ProcessSignalGeneratingFunctions
SignalConnectors
SignalConveyingFunctions
*
*
*
*
ActuatingElectricalFunction
ActuatingFunction
ProcessSignalGeneratingFunction
SignalOffPageConnector
SignalConveyingFunction
*
*
Source
Target
0..1
0..1
SignalConveyingFunctionSource
SignalConveyingFunctionTarget

Supertypes

- *CustomAttributeOwner*
- *SignalConveyingFunctionSource*
- *SignalConveyingFunctionTarget*
- *TechnicalItem*

Subtypes

- *ProcessControlFunction*

Attributes (data)

Name	Multiplicity	Type
DeviceInformation	0..1	*NullableString*
GmpRelevance	0..1	*GmpRelevanceClassification*
GuaranteedSupplyFunction	0..1	*GuaranteedSupplyFunctionClassification*
Location	0..1	*LocationClassification*
PanelIdentificationCode	0..1	*NullableString*
ProcessInstrumentationFunctionCategory	0..1	*NullableString*
ProcessInstrumentationFunctionModifier	0..1	*NullableString*
ProcessInstrumentationFunctionNumber	0..1	*NullableString*
ProcessInstrumentationFunctions	0..1	*NullableString*
QualityRelevance	0..1	*QualityRelevanceClassification*

(continued on next page)

Name	Multiplicity	Type
SafetyRelevanceClass	0..1	*NullableString*
TypicalInformation	0..1	*NullableString*
VendorCompanyName	0..1	*NullableString*
VotingSystemRepresentation	0..1	*NullableString*

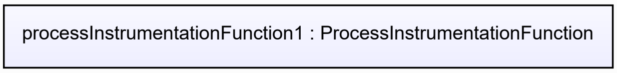
Attributes (composition)

Name	Multiplicity	Type
ActuatingElectricalFunctions	*	*ActuatingElectricalFunction*
ActuatingFunctions	*	*ActuatingFunction*
ProcessSignalGeneratingFunctions	*	*ProcessSignalGeneratingFunction*
SignalConnectors	*	*SignalOffPageConnector*
SignalConveyingFunctions	*	*SignalConveyingFunction*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<ProcessInstrumentationFunction>`
RDL reference: PROCESS INSTRUMENTATION FUNCTION
ComponentClass: ProcessInstrumentationFunction
ComponentClassURI: http://sandbox.dexpi.org/rdl/ProcessInstrumentationFunction

Example


processInstrumentationFunction1 : ProcessInstrumentationFunction

Example: Implementation in Proteus Schema

```
<ProcessInstrumentationFunction
    ID="processInstrumentationFunction1"
    ComponentClass="ProcessInstrumentationFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessInstrumentationFunction" ...>
  ...
</ProcessInstrumentationFunction>
```

9.22.2 ActuatingElectricalFunctions

Attribute (composition)

The *ActuatingElectricalFunctions* that are part of this *ProcessInstrumentationFunction*.

Multiplicity: *

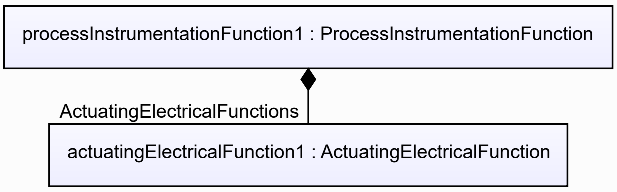
Type: *ActuatingElectricalFunction*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (an *ActuatingElectricalFunction*) is a child of the `<ProcessInstrumentationFunction>` element for the attribute owner (a *ProcessInstrumentationFunction*).

Example


processInstrumentationFunction1 : ProcessInstrumentationFunction
ActuatingElectricalFunctions
actuatingElectricalFunction1 : ActuatingElectricalFunction

Example: Implementation in Proteus Schema

```
<ProcessInstrumentationFunction
    ID="processInstrumentationFunction1"
    ComponentClass="ProcessInstrumentationFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessInstrumentationFunction" ...>
  ...
  <ActuatingElectricalFunction
      ID="actuatingElectricalFunction1"
      ComponentClass="ActuatingElectricalFunction"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/ActuatingElectricalFunction" ...>
     ...
  <ActuatingElectricalFunction />
  ...
<ProcessInstrumentationFunction />
```

9.22.3 ActuatingFunctions

Attribute (composition)

The *ActuatingFunctions* that are part of this *ProcessInstrumentationFunction*.

Multiplicity: *

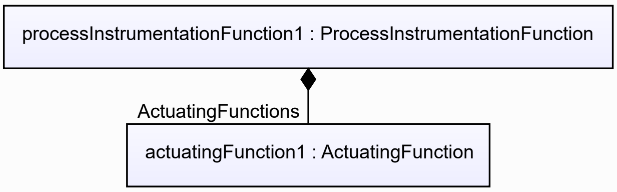
Type: *ActuatingFunction*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (an *ActuatingFunction*) is a child of the `<ProcessInstrumentationFunction>` element for the attribute owner (a *ProcessInstrumentationFunction*).

Example


processInstrumentationFunction1 : ProcessInstrumentationFunction
ActuatingFunctions
actuatingFunction1 : ActuatingFunction

Example: Implementation in Proteus Schema

```
<ProcessInstrumentationFunction
    ID="processInstrumentationFunction1"
    ComponentClass="ProcessInstrumentationFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessInstrumentationFunction" ...>
  ...
  <ActuatingFunction
      ID="actuatingFunction1"
      ComponentClass="ActuatingFunction"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/ActuatingFunction" ...>
    ...
  <ActuatingFunction />
  ...
<ProcessInstrumentationFunction />
```

9.22.4 DeviceInformation

Attribute (data)

Device information the *ProcessInstrumentationFunction*, e.g., for a detector.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: DEVICE INFORMATION ASSIGNMENT CLASS
Name: DeviceInformationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/DeviceInformationAssignmentClass

Example

“MDM” (*String*)

Example: Implementation in Proteus Schema

```
<ProcessInstrumentationFunction
    ID="processInstrumentationFunction1"
    ComponentClass="ProcessInstrumentationFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessInstrumentationFunction" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DeviceInformationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/DeviceInformationAssignmentClass"
      Format="string"
      Value="MDM" />
    ...
  </GenericAttributes>
  ...
</ProcessInstrumentationFunction>
```

9.22.5 GmpRelevance

Attribute (data)

A classification indicating if the *ProcessInstrumentationFunction* is relevant for GMP (good manufacturing practice).

Multiplicity: 0..1

Type: *GmpRelevanceClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: GMP RELEVANCE SPECIALIZATION
Name: GmpRelevanceSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/GmpRelevanceSpecialization

Example

GMP relevant (*GmpRelevanceClassification::GmpRelevantFunction*)

Example: Implementation in Proteus Schema

```
<ProcessInstrumentationFunction
    ID="processInstrumentationFunction1"
    ComponentClass="ProcessInstrumentationFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessInstrumentationFunction" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="GmpRelevanceSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/GmpRelevanceSpecialization"
      Format="anyURI"
      Value="GmpRelevantFunction"
      ValueURI="http://sandbox.dexpi.org/rdl/GmpRelevantFunction" />
    ...
  </GenericAttributes>
  ...
</ProcessInstrumentationFunction>
```

9.22.6 GuaranteedSupplyFunction

Attribute (data)

A classification indicating if the *ProcessInstrumentationFunction* is a guaranteed supply function.

Multiplicity: 0..1

Type: *GuaranteedSupplyFunctionClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: GUARANTEED SUPPLY FUNCTION SPECIALIZATION
Name: GuaranteedSupplyFunctionSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/GuaranteedSupplyFunctionSpecialization

Example

guaranteed supply (*GuaranteedSupplyFunctionClassification::GuaranteedSupplyFunction*)

Example: Implementation in Proteus Schema

```
<ProcessInstrumentationFunction
    ID="processInstrumentationFunction1"
    ComponentClass="ProcessInstrumentationFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessInstrumentationFunction" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="GuaranteedSupplyFunctionSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/GuaranteedSupplyFunctionSpecialization"
      Format="anyURI"
      Value="GuaranteedSupplyFunction"
      ValueURI="http://sandbox.dexpi.org/rdl/GuaranteedSupplyFunction" />
    ...
  </GenericAttributes>
  ...
</ProcessInstrumentationFunction>
```

9.22.7 Location

Attribute (data)

A specialization indicating the location of the *ProcessInstrumentationFunction*.

Multiplicity: 0..1

Type: *LocationClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: LOCATION SPECIALIZATION
Name: LocationSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/LocationSpecialization

Example

field (*LocationClassification::Field*)

Example: Implementation in Proteus Schema

```
<ProcessInstrumentationFunction
    ID="processInstrumentationFunction1"
    ComponentClass="ProcessInstrumentationFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessInstrumentationFunction" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="LocationSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/LocationSpecialization"
      Format="anyURI"
      Value="Field"
      ValueURI="http://data.posccaesar.org/rdl/RDS409545541" />
    ...
  </GenericAttributes>
  ...
</ProcessInstrumentationFunction>
```

9.22.8 PanelIdentificationCode

Attribute (data)

The panel identification code of the *ProcessInstrumentationFunction*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PANEL IDENTIFICATION CODE ASSIGNMENT CLASS
Name: PanelIdentificationCodeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/PanelIdentificationCodeAssignmentClass

Example

“P 3A” (*String*)

Example: Implementation in Proteus Schema

```
<ProcessInstrumentationFunction
    ID="processInstrumentationFunction1"
    ComponentClass="ProcessInstrumentationFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessInstrumentationFunction" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PanelIdentificationCodeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/PanelIdentificationCodeAssignmentClass"
      Format="string"
      Value="P 3A" />
    ...
  </GenericAttributes>
  ...
</ProcessInstrumentationFunction>
```

9.22.9 ProcessInstrumentationFunctionCategory

Attribute (data)

The function category of the *ProcessInstrumentationFunction*. The value is a string, typically one or two letters. Recent standards for PIDs normally enforce a single letter from a fixed list. However, there are no formal DEXPI restrictions for valid strings.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PROCESS INSTRUMENTATION FUNCTION CATEGORY ASSIGNMENT CLASS
Name: ProcessInstrumentationFunctionCategoryAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ProcessInstrumentationFunctionCategoryAssignmentClass

Example

“H” (*String*)

Example: Implementation in Proteus Schema

```
<ProcessInstrumentationFunction
    ID="processInstrumentationFunction1"
    ComponentClass="ProcessInstrumentationFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessInstrumentationFunction" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ProcessInstrumentationFunctionCategoryAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ProcessInstrumentationFunctionCategoryAssignmentClass"
      Format="string"
      Value="H" />
    ...
  </GenericAttributes>
  ...
</ProcessInstrumentationFunction>
```

9.22.10 ProcessInstrumentationFunctionModifier

Attribute (data)

The modifier of the *ProcessInstrumentationFunction*. The value is a string, typically a single letter, e.g., D for difference. So far, there are no formal DEXPI restrictions for valid strings.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PROCESS INSTRUMENTATION FUNCTION MODIFIER ASSIGNMENT CLASS
Name: ProcessInstrumentationFunctionModifierAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ProcessInstrumentationFunctionModifierAssignmentClass

Example

“D” (*String*)

Example: Implementation in Proteus Schema

```
<ProcessInstrumentationFunction
    ID="processInstrumentationFunction1"
    ComponentClass="ProcessInstrumentationFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessInstrumentationFunction" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ProcessInstrumentationFunctionModifierAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ProcessInstrumentationFunctionModifierAssignmentClass"
      Format="string"
      Value="D" />
    ...
  </GenericAttributes>
  ...
</ProcessInstrumentationFunction>
```

9.22.11 ProcessInstrumentationFunctionNumber

Attribute (data)

A unique identifier for the *ProcessInstrumentationFunction*. If the *ProcessInstrumentationFunction* is part of a *InstrumentationLoopFunction*, the identifier of the *ProcessInstrumentationFunction* usually contains the identifier of the *InstrumentationLoopFunction* (see *InstrumentationLoopFunctionNumber*).

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PROCESS INSTRUMENTATION FUNCTION NUMBER ASSIGNMENT CLASS
Name: ProcessInstrumentationFunctionNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ProcessInstrumentationFunctionNumberAssignmentClass

Example

“H4750.01” (*String*)

Example: Implementation in Proteus Schema

```
<ProcessInstrumentationFunction
    ID="processInstrumentationFunction1"
    ComponentClass="ProcessInstrumentationFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessInstrumentationFunction" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ProcessInstrumentationFunctionNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ProcessInstrumentationFunctionNumberAssignmentClass"
      Format="string"
      Value="H4750.01" />
    ...
  </GenericAttributes>
  ...
</ProcessInstrumentationFunction>
```

9.22.12 ProcessInstrumentationFunctions

Attribute (data)

Additional functions of the *ProcessInstrumentationFunction* (i.e., in addition to the function category, see *ProcessInstrumentationFunctionCategory*).

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PROCESS INSTRUMENTATION FUNCTIONS ASSIGNMENT CLASS
Name: ProcessInstrumentationFunctionsAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ProcessInstrumentationFunctionsAssignmentClass

Example

"HS" (*String*)

Example: Implementation in Proteus Schema

```
<ProcessInstrumentationFunction
    ID="processInstrumentationFunction1"
    ComponentClass="ProcessInstrumentationFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessInstrumentationFunction" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ProcessInstrumentationFunctionsAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ProcessInstrumentationFunctionsAssignmentClass"
      Format="string"
      Value="HS" />
    ...
  </GenericAttributes>
  ...
</ProcessInstrumentationFunction>
```

9.22.13 ProcessSignalGeneratingFunctions

Attribute (composition)

The *ProcessSignalGeneratingFunctions* that are part of this *ProcessInstrumentationFunction*.

Multiplicity: *

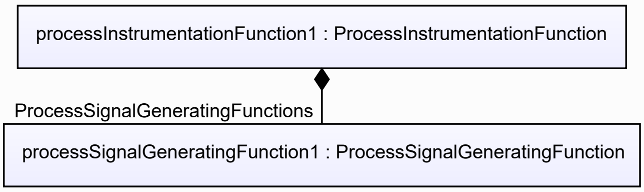
Type: *ProcessSignalGeneratingFunction*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *ProcessSignalGeneratingFunction*) is a child of the `<ProcessInstrumentationFunction>` element for the attribute owner (a *ProcessInstrumentationFunction*).

Example


processInstrumentationFunction1 : ProcessInstrumentationFunction
ProcessSignalGeneratingFunctions
processSignalGeneratingFunction1 : ProcessSignalGeneratingFunction

Example: Implementation in Proteus Schema

```
<ProcessInstrumentationFunction
    ID="processInstrumentationFunction1"
    ComponentClass="ProcessInstrumentationFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessInstrumentationFunction" ...>
  ...
  <ProcessSignalGeneratingFunction
      ID="processSignalGeneratingFunction1"
      ComponentClass="ProcessSignalGeneratingFunction"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessSignalGeneratingFunction" ...>
    ...
  <ProcessSignalGeneratingFunction />
  ...
<ProcessInstrumentationFunction />
```

9.22.14 QualityRelevance

Attribute (data)

A classification indicating if the *ProcessInstrumentationFunction* is quality relevant.

Multiplicity: 0..1

Type: *QualityRelevanceClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: QUALITY RELEVANCE SPECIALIZATION
Name: QualityRelevanceSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/QualityRelevanceSpecialization

Example

quality relevant (*QualityRelevanceClassification::QualityRelevantFunction*)

Example: Implementation in Proteus Schema

```
<ProcessInstrumentationFunction
    ID="processInstrumentationFunction1"
    ComponentClass="ProcessInstrumentationFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessInstrumentationFunction" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="QualityRelevanceSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/QualityRelevanceSpecialization"
      Format="anyURI"
      Value="QualityRelevantFunction"
      ValueURI="http://sandbox.dexpi.org/rdl/QualityRelevantFunction" />
    ...
  </GenericAttributes>
  ...
</ProcessInstrumentationFunction>
```

9.22.15 SafetyRelevanceClass

Attribute (data)

The safety relevance class the *ProcessInstrumentationFunction*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: SAFETY RELEVANCE CLASS ASSIGNMENT CLASS
Name: SafetyRelevanceClassAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/SafetyRelevanceClassAssignmentClass

Example

"SIL3" (*String*)

Example: Implementation in Proteus Schema

```
<ProcessInstrumentationFunction
    ID="processInstrumentationFunction1"
    ComponentClass="ProcessInstrumentationFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessInstrumentationFunction" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="SafetyRelevanceClassAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/SafetyRelevanceClassAssignmentClass"
      Format="string"
      Value="SIL3" />
    ...
  </GenericAttributes>
  ...
</ProcessInstrumentationFunction>
```

9.22.16 SignalConnectors

Attribute (composition)

The *SignalOffPageConnectors* that are part of this *ProcessInstrumentationFunction*.

Multiplicity: *

Type: *SignalOffPageConnector*

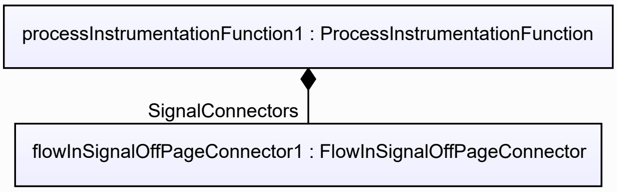
Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *SignalOffPageConnector*) is a child of the `<ProcessInstrumentationFunction>` element for the attribute owner (a *ProcessInstrumentationFunction*).

Example

As the value type *SignalOffPageConnector* is abstract, we consider *FlowInSignalOffPageConnector* as an arbitrary concrete subclass.


processInstrumentationFunction1 : ProcessInstrumentationFunction
SignalConnectors
flowInSignalOffPageConnector1 : FlowInSignalOffPageConnector

Example: Implementation in Proteus Schema

```
<ProcessInstrumentationFunction
    ID="processInstrumentationFunction1"
    ComponentClass="ProcessInstrumentationFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessInstrumentationFunction" ...>
  ...
  <InformationFlowOffPageConnector
      ID="flowInSignalOffPageConnector1"
      ComponentClass="FlowInSignalOffPageConnector"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/FlowInSignalOffPageConnector" ...>
    ...
  <InformationFlowOffPageConnector />
  ...
<ProcessInstrumentationFunction />
```

9.22.17 SignalConveyingFunctions

Attribute (composition)

The *SignalConveyingFunctions* that are part of this *ProcessInstrumentationFunction*.

Multiplicity: *

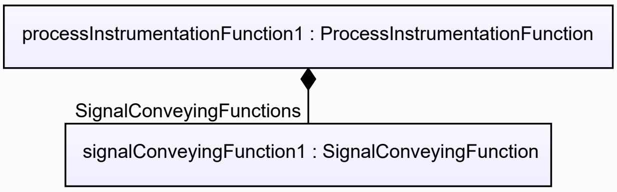
Type: *SignalConveyingFunction*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *SignalConveyingFunction*) is a child of the `<ProcessInstrumentationFunction>` element for the attribute owner (a *ProcessInstrumentationFunction*).

Example


processInstrumentationFunction1 : ProcessInstrumentationFunction
SignalConveyingFunctions
signalConveyingFunction1 : SignalConveyingFunction

Example: Implementation in Proteus Schema

```
<ProcessInstrumentationFunction
    ID="processInstrumentationFunction1"
    ComponentClass="ProcessInstrumentationFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessInstrumentationFunction" ...>
  ...
  <InformationFlow
      ID="signalConveyingFunction1"
      ComponentClass="SignalConveyingFunction"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/SignalConveyingFunction" ...>
    ...
  <InformationFlow />
  ...
<ProcessInstrumentationFunction />
```

9.22.18 TypicalInformation

Attribute (data)

Typical information about the *ProcessInstrumentationFunction*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: TYPICAL INFORMATION ASSIGNMENT CLASS
Name: TypicalInformationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/TypicalInformationAssignmentClass

Example

“F4” (*String*)

Example: Implementation in Proteus Schema

```
<ProcessInstrumentationFunction
    ID="processInstrumentationFunction1"
    ComponentClass="ProcessInstrumentationFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessInstrumentationFunction" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="TypicalInformationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/TypicalInformationAssignmentClass"
      Format="string"
      Value="F4" />
    ...
  </GenericAttributes>
  ...
</ProcessInstrumentationFunction>
```

9.22.19 VendorCompanyName

Attribute (data)

The vendor company name the *ProcessInstrumentationFunction*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: VENDOR COMPANY NAME ASSIGNMENT CLASS
Name: VendorCompanyNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/VendorCompanyNameAssignmentClass

Example

"Emerson" (*String*)

Example: Implementation in Proteus Schema

```
<ProcessInstrumentationFunction
    ID="processInstrumentationFunction1"
    ComponentClass="ProcessInstrumentationFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessInstrumentationFunction" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="VendorCompanyNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/VendorCompanyNameAssignmentClass"
      Format="string"
      Value="Emerson" />
    ...
  </GenericAttributes>
  ...
</ProcessInstrumentationFunction>
```

9.22.20 VotingSystemRepresentation

Attribute (data)

A representation of the voting system of the *ProcessInstrumentationFunction*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: VOTING SYSTEM REPRESENTATION ASSIGNMENT CLASS
Name: VotingSystemRepresentationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/VotingSystemRepresentationAssignmentClass

Example

"1oo2" (*String*)

Example: Implementation in Proteus Schema

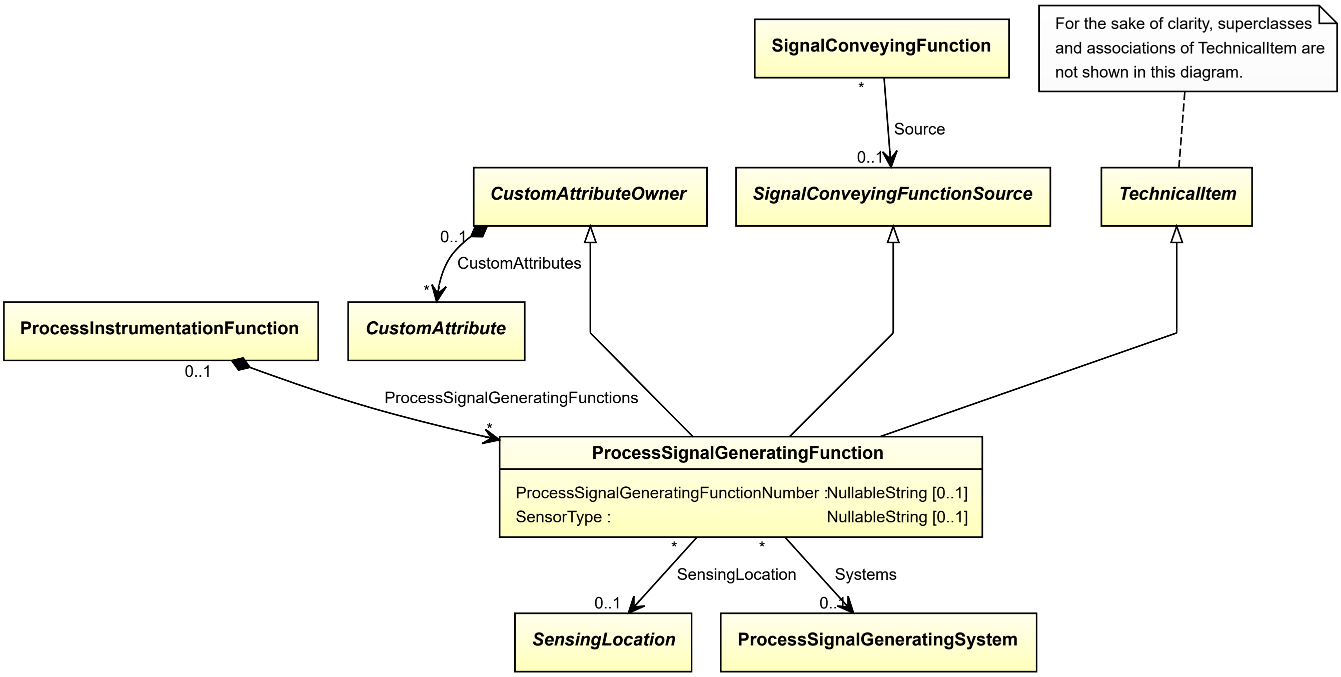
```
<ProcessInstrumentationFunction
    ID="processInstrumentationFunction1"
    ComponentClass="ProcessInstrumentationFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessInstrumentationFunction" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="VotingSystemRepresentationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/VotingSystemRepresentationAssignmentClass"
      Format="string"
      Value="1oo2" />
    ...
  </GenericAttributes>
  ...
</ProcessInstrumentationFunction>
```

9.23. ProcessSignalGeneratingFunction

9.23.1 Overview

Class

A function for instrumentation and/or control structures relating to Process Engineering


SignalConveyingFunction
For the sake of clarity, superclasses and associations of TechnicalItem are not shown in this diagram.
*
Source
0..1
CustomAttributeOwner
SignalConveyingFunctionSource
TechnicalItem
0..1
CustomAttributes
*
ProcessInstrumentationFunction
CustomAttribute
0..1
ProcessSignalGeneratingFunctions
*
ProcessSignalGeneratingFunction
ProcessSignalGeneratingFunctionNumber :NullableString [0..1]
SensorType : NullableString [0..1]
*
*
SensingLocation
Systems
0..1
0..1
SensingLocation
ProcessSignalGeneratingSystem

Supertypes

- *CustomAttributeOwner*
- *SignalConveyingFunctionSource*
- *TechnicalItem*

Attributes (data)

Name	Multiplicity	Type
ProcessSignalGeneratingFunctionNumber	0..1	*NullableString*
SensorType	0..1	*NullableString*

Attributes (reference)

Name	Multiplicity	Type
SensingLocation	0..1	*SensingLocation*
Systems	0..1	*ProcessSignalGeneratingSystem*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<ProcessSignalGeneratingFunction>`
RDL reference: PROCESS SIGNAL GENERATING FUNCTION
ComponentClass: ProcessSignalGeneratingFunction
ComponentClassURI: http://sandbox.dexpi.org/rdl/ProcessSignalGeneratingFunction

Example

processSignalGeneratingFunction1 : ProcessSignalGeneratingFunction

Example: Implementation in Proteus Schema

```
<ProcessSignalGeneratingFunction
    ID="processSignalGeneratingFunction1"
    ComponentClass="ProcessSignalGeneratingFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessSignalGeneratingFunction" ...>
  ...
</ProcessSignalGeneratingFunction>
```

9.23.2 ProcessSignalGeneratingFunctionNumber

Attribute (data)

An identifier for the *ProcessSignalGeneratingFunction*. It usually contains the identifier of the *ProcessInstrumentationFunction* that includes the *ProcessSignalGeneratingFunction* (see *ProcessInstrumentationFunctionNumber*).

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PROCESS SIGNAL GENERATING FUNCTION NUMBER ASSIGNMENT CLASS
Name: ProcessSignalGeneratingFunctionNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ProcessSignalGeneratingFunctionNumberAssignmentClass

Example

“TT4750.03” (*String*)

Example: Implementation in Proteus Schema

```
<ProcessSignalGeneratingFunction
    ID="processSignalGeneratingFunction1"
    ComponentClass="ProcessSignalGeneratingFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessSignalGeneratingFunction" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ProcessSignalGeneratingFunctionNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ProcessSignalGeneratingFunctionNumberAssignmentClass"
      Format="string"
      Value="TT4750.03" />
    ...
  </GenericAttributes>
  ...
</ProcessSignalGeneratingFunction>
```

9.23.3 SensingLocation

Attribute (reference)

The sensing location of the *ProcessSignalGeneratingFunction*.

Multiplicity: 0..1

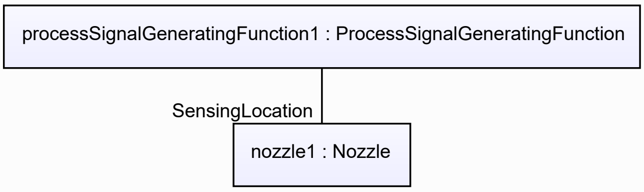
Type: *SensingLocation*

Opposite multiplicity: 0..*

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: `"is located in"`
Opposite association type: `"is the location of"`

Example


processSignalGeneratingFunction1 : ProcessSignalGeneratingFunction
SensingLocation
nozzle1 : Nozzle

Example: Implementation in Proteus Schema

```
<ProcessSignalGeneratingFunction
    ID="processSignalGeneratingFunction1"
    ComponentClass="ProcessSignalGeneratingFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessSignalGeneratingFunction" ...>
  ...
  <Association
    Type="is located in"
    ItemID="nozzle1" />
  ...
<ProcessSignalGeneratingFunction />
...
<Nozzle
    ID="nozzle1"
    ComponentClass="Nozzle"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS415214" ...>
  ...
  <Association
    Type="is the location of"
    ItemID="processSignalGeneratingFunction1" />
  ...
<Nozzle />
```

9.23.4 SensorType

Attribute (data)

The sensor type of the *ProcessSignalGeneratingFunction*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: SENSOR TYPE ASSIGNMENT CLASS
Name: SensorTypeAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/SensorTypeAssignmentClass

Example

“MDM” (*String*)

Example: Implementation in Proteus Schema

```
<ProcessSignalGeneratingFunction
    ID="processSignalGeneratingFunction1"
    ComponentClass="ProcessSignalGeneratingFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessSignalGeneratingFunction" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="SensorTypeAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/SensorTypeAssignmentClass"
      Format="string"
      Value="MDM" />
    ...
  </GenericAttributes>
  ...
</ProcessSignalGeneratingFunction>
```

9.23.5 Systems

Attribute (reference)

The ProcessSignalGeneratingSystem that implements the *ProcessSignalGeneratingFunction*.

Multiplicity: 0..1

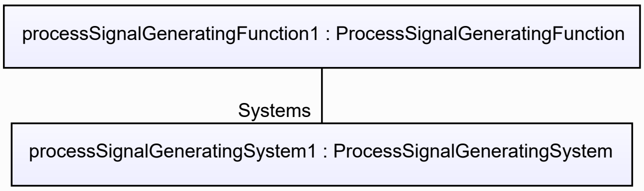
Type: *ProcessSignalGeneratingSystem*

Opposite multiplicity: 0..*

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: `"is fulfilled by"`
Opposite association type: `"fulfills"`

Example


processSignalGeneratingFunction1 : ProcessSignalGeneratingFunction
Systems
processSignalGeneratingSystem1 : ProcessSignalGeneratingSystem

Example: Implementation in Proteus Schema

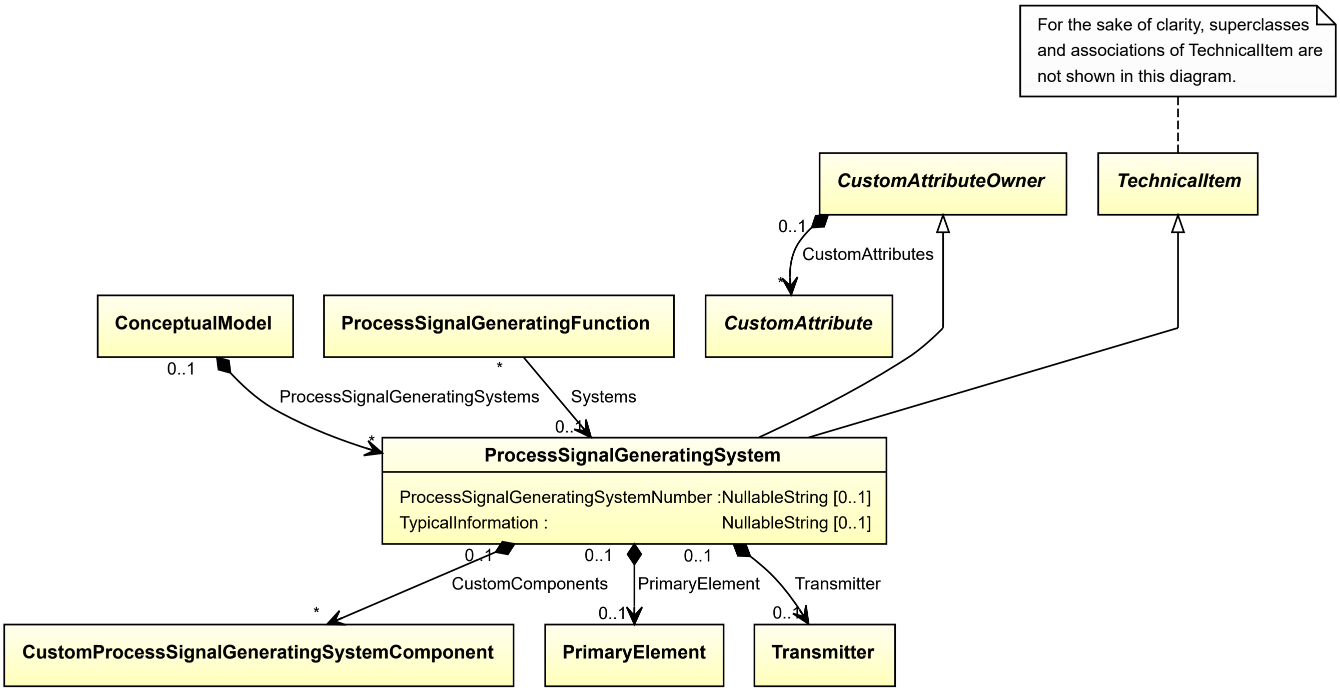
```
<ProcessSignalGeneratingFunction
    ID="processSignalGeneratingFunction1"
    ComponentClass="ProcessSignalGeneratingFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessSignalGeneratingFunction" ...>
  ...
  <Association
    Type="is fulfilled by"
    ItemID="processSignalGeneratingSystem1" />
  ...
<ProcessSignalGeneratingFunction />
...
<ProcessSignalGeneratingSystem
    ID="processSignalGeneratingSystem1"
    ComponentClass="ProcessSignalGeneratingSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessSignalGeneratingSystem" ...>
  ...
  <Association
    Type="fulfills"
    ItemID="processSignalGeneratingFunction1" />
  ...
<ProcessSignalGeneratingSystem />
```

9.24. ProcessSignalGeneratingSystem

9.24.1 Overview

Class

An assembly of artefacts that is designed to fulfill one or more *ProcessSignalGeneratingFunctions*.


For the sake of clarity, superclasses and associations of TechnicalItem are not shown in this diagram.
CustomAttributeOwner
TechnicalItem
0..1
CustomAttributes
*
ConceptualModel
ProcessSignalGeneratingFunction
CustomAttribute
0..1
*
ProcessSignalGeneratingSystems
Systems
0..1
*
ProcessSignalGeneratingSystem
ProcessSignalGeneratingSystemNumber :NullableString [0..1]
TypicalInformation : NullableString [0..1]
0..1
0..1
0..1
CustomComponents
PrimaryElement
Transmitter
*
0..1
0..1
CustomProcessSignalGeneratingSystemComponent
PrimaryElement
Transmitter

Supertypes

- *CustomAttributeOwner*
- *TechnicalItem*

Subtypes

- *FlowDetector*

Attributes (data)

Name	Multiplicity	Type
ProcessSignalGeneratingSystemNumber	0..1	*NullableString*
TypicalInformation	0..1	*NullableString*

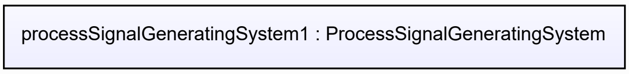
Attributes (composition)

Name	Multiplicity	Type
CustomComponents	*	*CustomProcessSignalGeneratingSystemComponent*
PrimaryElement	0..1	*PrimaryElement*
Transmitter	0..1	*Transmitter*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<ProcessSignalGeneratingSystem>`
RDL reference: PROCESS SIGNAL GENERATING SYSTEM
ComponentClass: ProcessSignalGeneratingSystem
ComponentClassURI: http://sandbox.dexpi.org/rdl/ProcessSignalGeneratingSystem

Example


processSignalGeneratingSystem1 : ProcessSignalGeneratingSystem

Example: Implementation in Proteus Schema

```
<ProcessSignalGeneratingSystem
    ID="processSignalGeneratingSystem1"
    ComponentClass="ProcessSignalGeneratingSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessSignalGeneratingSystem" ...>
  ...
</ProcessSignalGeneratingSystem>
```

9.24.2 CustomComponents

Attribute (composition)

The custom components of the *ProcessSignalGeneratingSystem*.

Multiplicity: *

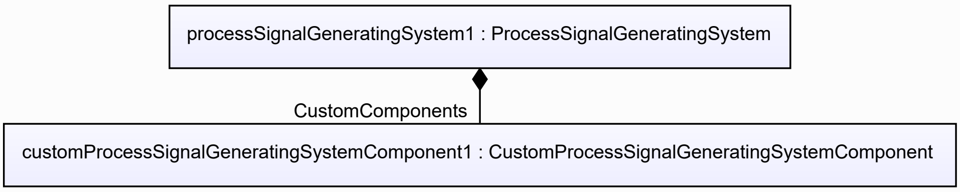
Type: *CustomProcessSignalGeneratingSystemComponent*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *CustomProcessSignalGeneratingSystemComponent*) is a child of the `<ProcessSignalGeneratingSystem>` element for the attribute owner (a *ProcessSignalGeneratingSystem*).

Example


processSignalGeneratingSystem1 : ProcessSignalGeneratingSystem
CustomComponents
customProcessSignalGeneratingSystemComponent1 : CustomProcessSignalGeneratingSystemComponent

Example: Implementation in Proteus Schema

```
<ProcessSignalGeneratingSystem
    ID="processSignalGeneratingSystem1"
    ComponentClass="ProcessSignalGeneratingSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessSignalGeneratingSystem" ...>
  ...
  <ProcessSignalGeneratingSystemComponent
      ID="customProcessSignalGeneratingSystemComponent1"
      ComponentClass="CustomProcessSignalGeneratingSystemComponent"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomProcessSignalGeneratingSystemComponent" ...>
    ...
  <ProcessSignalGeneratingSystemComponent />
  ...
<ProcessSignalGeneratingSystem />
```

9.24.3 PrimaryElement

Attribute (composition)

The *PrimaryElement* of the *ProcessSignalGeneratingSystem*.

Multiplicity: 0..1

Type: *PrimaryElement*

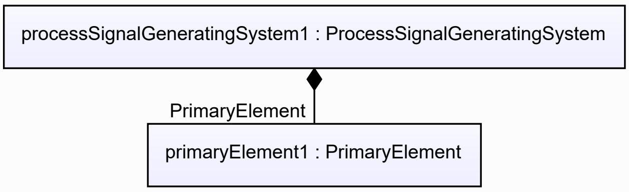
Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *PrimaryElement*) is a child of the `<ProcessSignalGeneratingSystem>` element for the attribute owner (a

ProcessSignalGeneratingSystem).

Example


processSignalGeneratingSystem1 : ProcessSignalGeneratingSystem
PrimaryElement
primaryElement1 : PrimaryElement

Example: Implementation in Proteus Schema

```
<ProcessSignalGeneratingSystem
    ID="processSignalGeneratingSystem1"
    ComponentClass="ProcessSignalGeneratingSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessSignalGeneratingSystem" ...>
  ...
  <ProcessSignalGeneratingSystemComponent
      ID="primaryElement1"
      ComponentClass="PrimaryElement"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/PrimaryElement" ...>
    ...
  <ProcessSignalGeneratingSystemComponent />
  ...
<ProcessSignalGeneratingSystem />
```

9.24.4 ProcessSignalGeneratingSystemNumber

Attribute (data)

The number of the *ProcessSignalGeneratingSystem*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: PROCESS SIGNAL GENERATING SYSTEM NUMBER ASSIGNMENT CLASS
Name: ProcessSignalGeneratingSystemNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ProcessSignalGeneratingSystemNumberAssignmentClass

Example

“FE0001” (*String*)

Example: Implementation in Proteus Schema

```
<ProcessSignalGeneratingSystem
    ID="processSignalGeneratingSystem1"
    ComponentClass="ProcessSignalGeneratingSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessSignalGeneratingSystem" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ProcessSignalGeneratingSystemNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ProcessSignalGeneratingSystemNumberAssignmentClass"
      Format="string"
      Value="FE0001" />
    ...
  </GenericAttributes>
  ...
</ProcessSignalGeneratingSystem>
```

9.24.5 Transmitter

Attribute (composition)

The *Transmitter* of the *ProcessSignalGeneratingSystem*.

Multiplicity: 0..1

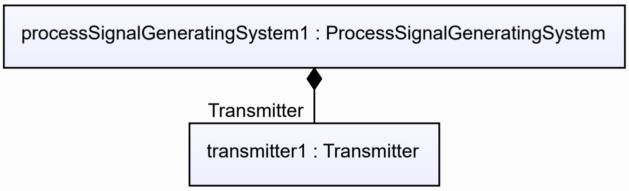
Type: *Transmitter*

Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *Transmitter*) is a child of the `<ProcessSignalGeneratingSystem>` element for the attribute owner (a *ProcessSignalGeneratingSystem*).

Example


processSignalGeneratingSystem1 : ProcessSignalGeneratingSystem
Transmitter
transmitter1 : Transmitter

Example: Implementation in Proteus Schema

```
<ProcessSignalGeneratingSystem
    ID="processSignalGeneratingSystem1"
    ComponentClass="ProcessSignalGeneratingSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessSignalGeneratingSystem" ...>
  ...
  <ProcessSignalGeneratingSystemComponent
      ID="transmitter1"
      ComponentClass="Transmitter"
      ComponentClassURI="http://data.posccaesar.org/rdl/RDS267929" ...>
    ...
  <ProcessSignalGeneratingSystemComponent />
  ...
<ProcessSignalGeneratingSystem />
```

9.24.6 TypicalInformation

Attribute (data)

Typical information about the *ProcessSignalGeneratingSystem*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: TYPICAL INFORMATION ASSIGNMENT CLASS
Name: TypicalInformationAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/TypicalInformationAssignmentClass

Example

“F4” (*String*)

Example: Implementation in Proteus Schema

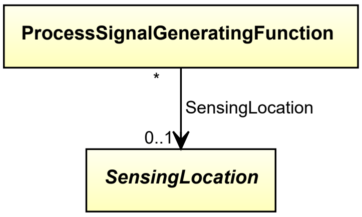
```
<ProcessSignalGeneratingSystem
    ID="processSignalGeneratingSystem1"
    ComponentClass="ProcessSignalGeneratingSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessSignalGeneratingSystem" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="TypicalInformationAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/TypicalInformationAssignmentClass"
      Format="string"
      Value="F4" />
    ...
  </GenericAttributes>
  ...
</ProcessSignalGeneratingSystem>
```

9.25. SensingLocation

9.25.1 Overview

Abstract class

An object than can act as a *SensingLocation* of a *ProcessSignalGeneratingFunction*.



Subtypes

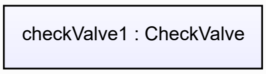
- *Nozzle*
- *PipingComponent*
- *PipingNetworkSegment*

Implementation in Proteus Schema

Implementation is subclass-specific.

Example

As *SensingLocation* is abstract, we consider *CheckValve* as an arbitrary concrete subclass.



Example: Implementation in Proteus Schema

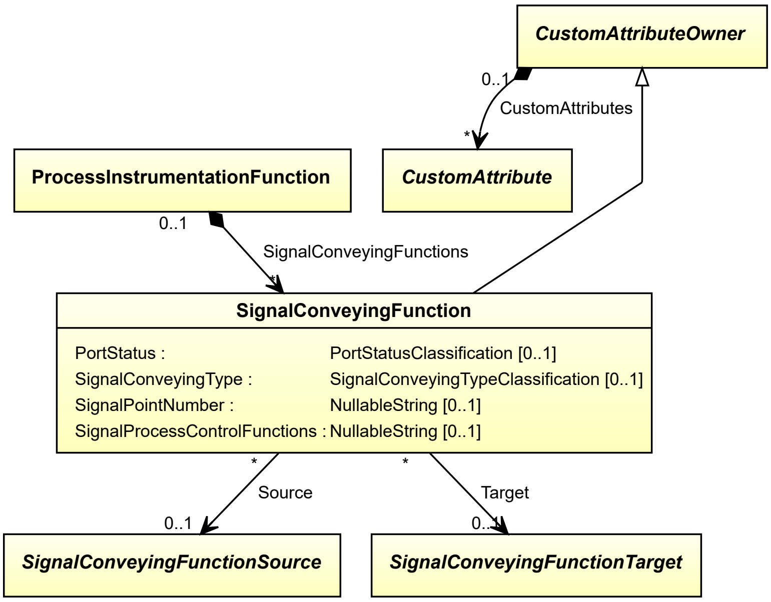
```
<PipingComponent
    ID="checkValve1"
    ComponentClass="CheckValve"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS292229" ...>
  ...
</PipingComponent>
```

9.26. SignalConveyingFunction

9.26.1 Overview

Class

A function for conveying a signal.


CustomAttributeOwner
0..1
CustomAttributes
*
CustomAttribute
ProcessInstrumentationFunction
0..1
SignalConveyingFunctions
*
SignalConveyingFunction
PortStatus : PortStatusClassification [0..1]
SignalConveyingType : SignalConveyingTypeClassification [0..1]
SignalPointNumber : NullableString [0..1]
SignalProcessControlFunctions : NullableString [0..1]
*
Source
0..1
*
Target
0..1
SignalConveyingFunctionSource
SignalConveyingFunctionTarget

Supertypes

- *CustomAttributeOwner*

Subtypes

- *MeasuringLineFunction*
- *SignalLineFunction*

Attributes (data)

Name	Multiplicity	Type
PortStatus	0..1	*PortStatusClassification*
SignalConveyingType	0..1	*SignalConveyingTypeClassification*
SignalPointNumber	0..1	*NullableString*
SignalProcessControlFunctions	0..1	*NullableString*

Attributes (reference)

Name	Multiplicity	Type
Source	0..1	*SignalConveyingFunctionSource*
Target	0..1	*SignalConveyingFunctionTarget*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<InformationFlow>`
RDL reference: SIGNAL CONVEYING FUNCTION
ComponentClass: SignalConveyingFunction
ComponentClassURI: http://sandbox.dexpi.org/rdl/SignalConveyingFunction

Example

signalConveyingFunction1 : SignalConveyingFunction

Example: Implementation in Proteus Schema

```
<InformationFlow
    ID="signalConveyingFunction1"
    ComponentClass="SignalConveyingFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SignalConveyingFunction" ...>
  ...
</InformationFlow>
```

9.26.2 PortStatus

Attribute (data)

A classification indicating the port status of the *SignalConveyingFunction*.

Multiplicity: 0..1

Type: *PortStatusClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: PORT STATUS SPECIALIZATION
Name: PortStatusSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/PortStatusSpecialization

Example

HH (*PortStatusClassification::StatusHighHighPort*)

Example: Implementation in Proteus Schema

```
<InformationFlow
    ID="signalConveyingFunction1"
    ComponentClass="SignalConveyingFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SignalConveyingFunction" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="PortStatusSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/PortStatusSpecialization"
      Format="anyURI"
      Value="StatusHighHighPort"
      ValueURI="http://data.posccaesar.org/rdl/RDS323099" />
    ...
  </GenericAttributes>
  ...
</InformationFlow>
```

9.26.3 SignalConveyingType

Attribute (data)

A classification indicating the signal conveying type of the *SignalConveyingFunction*.

Multiplicity: 0..1

Type: *SignalConveyingTypeClassification*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for enumeration values*.
RDL reference: SIGNAL CONVEYING TYPE SPECIALIZATION
Name: SignalConveyingTypeSpecialization
AttributeURI: http://sandbox.dexpi.org/rdl/SignalConveyingTypeSpecialization

Example

electrical (*SignalConveyingTypeClassification::ElectricalSignalConveying*)

Example: Implementation in Proteus Schema

```
<InformationFlow
    ID="signalConveyingFunction1"
    ComponentClass="SignalConveyingFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SignalConveyingFunction" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="SignalConveyingTypeSpecialization"
      AttributeURI="http://sandbox.dexpi.org/rdl/SignalConveyingTypeSpecialization"
      Format="anyURI"
      Value="ElectricalSignalConveying"
      ValueURI="http://sandbox.dexpi.org/rdl/ElectricalSignalConveying" />
    ...
  </GenericAttributes>
  ...
</InformationFlow>
```

9.26.4 SignalPointNumber

Attribute (data)

The signal point number of the *SignalConveyingFunction*. Typical values are 1 to 6.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: SIGNAL POINT NUMBER ASSIGNMENT CLASS
Name: SignalPointNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/SignalPointNumberAssignmentClass

Example

“2” (*String*)

Example: Implementation in Proteus Schema

```
<InformationFlow
    ID="signalConveyingFunction1"
    ComponentClass="SignalConveyingFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SignalConveyingFunction" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="SignalPointNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/SignalPointNumberAssignmentClass"
      Format="string"
      Value="2" />
    ...
  </GenericAttributes>
  ...
</InformationFlow>
```

9.26.5 SignalProcessControlFunctions

Attribute (data)

The process control functions of the *SignalConveyingFunction*. Values are combinations of characters.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: SIGNAL PROCESS CONTROL FUNCTIONS ASSIGNMENT CLASS
Name: SignalProcessControlFunctionsAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/SignalProcessControlFunctionsAssignmentClass

Example

“SA” (*String*)

Example: Implementation in Proteus Schema

```
<InformationFlow
    ID="signalConveyingFunction1"
    ComponentClass="SignalConveyingFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SignalConveyingFunction" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="SignalProcessControlFunctionsAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/SignalProcessControlFunctionsAssignmentClass"
      Format="string"
      Value="SA" />
    ...
  </GenericAttributes>
  ...
</InformationFlow>
```

9.26.6 Source

Attribute (reference)

The source of the signal conveyed by this *SignalConveyingFunction*.

Multiplicity: 0..1

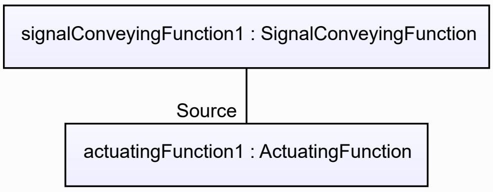
Type: *SignalConveyingFunctionSource*

Opposite multiplicity: 0..*

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: `"has logical start"`
Opposite association type: `"is logical start of"`

Example


signalConveyingFunction1 : SignalConveyingFunction
Source
actuatingFunction1 : ActuatingFunction

Example: Implementation in Proteus Schema

```
<InformationFlow
    ID="signalConveyingFunction1"
    ComponentClass="SignalConveyingFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SignalConveyingFunction" ...>
  ...
  <Association
    Type="has logical start"
    ItemID="actuatingFunction1" />
  ...
<InformationFlow />
...
<ActuatingFunction
    ID="actuatingFunction1"
    ComponentClass="ActuatingFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ActuatingFunction" ...>
  ...
  <Association
    Type="is logical start of"
    ItemID="signalConveyingFunction1" />
  ...
<ActuatingFunction />
```

9.26.7 Target

Attribute (reference)

The target of the signal conveyed by this *SignalConveyingFunction*.

Multiplicity: 0..1

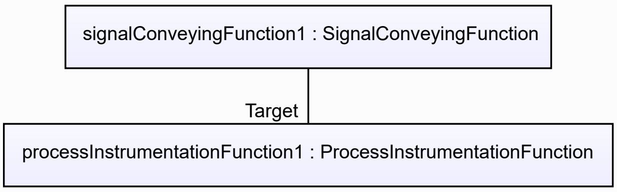
Type: *SignalConveyingFunctionTarget*

Opposite multiplicity: 0..*

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: "has logical end"
Opposite association type: "is logical end of"

Example


signalConveyingFunction1 : SignalConveyingFunction
Target
processInstrumentationFunction1 : ProcessInstrumentationFunction

Example: Implementation in Proteus Schema

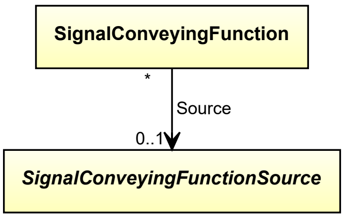
```
<InformationFlow
    ID="signalConveyingFunction1"
    ComponentClass="SignalConveyingFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SignalConveyingFunction" ...>
  ...
  <Association
    Type="has logical end"
    ItemID="processInstrumentationFunction1" />
  ...
<InformationFlow />
...
<ProcessInstrumentationFunction
    ID="processInstrumentationFunction1"
    ComponentClass="ProcessInstrumentationFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ProcessInstrumentationFunction" ...>
  ...
  <Association
    Type="is logical end of"
    ItemID="signalConveyingFunction1" />
  ...
<ProcessInstrumentationFunction />
```

9.27. SignalConveyingFunctionSource

9.27.1 Overview

Abstract class

An object than can act as the *Source* of a *SignalConveyingFunction*.



Subtypes

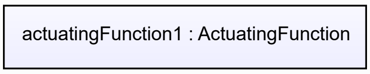
- *ActuatingFunction*
- *FlowInSignalOffPageConnector*
- *ProcessInstrumentationFunction*
- *ProcessSignalGeneratingFunction*

Implementation in Proteus Schema

Implementation is subclass-specific.

Example

As *SignalConveyingFunctionSource* is abstract, we consider *ActuatingFunction* as an arbitrary concrete subclass.



Example: Implementation in Proteus Schema

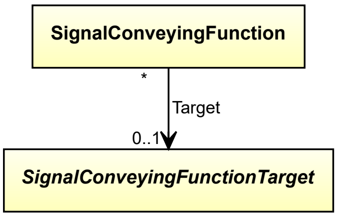
```
<ActuatingFunction
    ID="actuatingFunction1"
    ComponentClass="ActuatingFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ActuatingFunction" ...>
  ...
</ActuatingFunction>
```

9.28. SignalConveyingFunctionTarget

9.28.1 Overview

Abstract class

An object than can act as the *Target* of a *SignalConveyingFunction*.



Subtypes

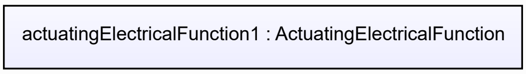
- *ActuatingElectricalFunction*
- *ActuatingFunction*
- *FlowOutSignalOffPageConnector*
- *ProcessInstrumentationFunction*

Implementation in Proteus Schema

Implementation is subclass-specific.

Example

As *SignalConveyingFunctionTarget* is abstract, we consider *ActuatingElectricalFunction* as an arbitrary concrete subclass.



Example: Implementation in Proteus Schema

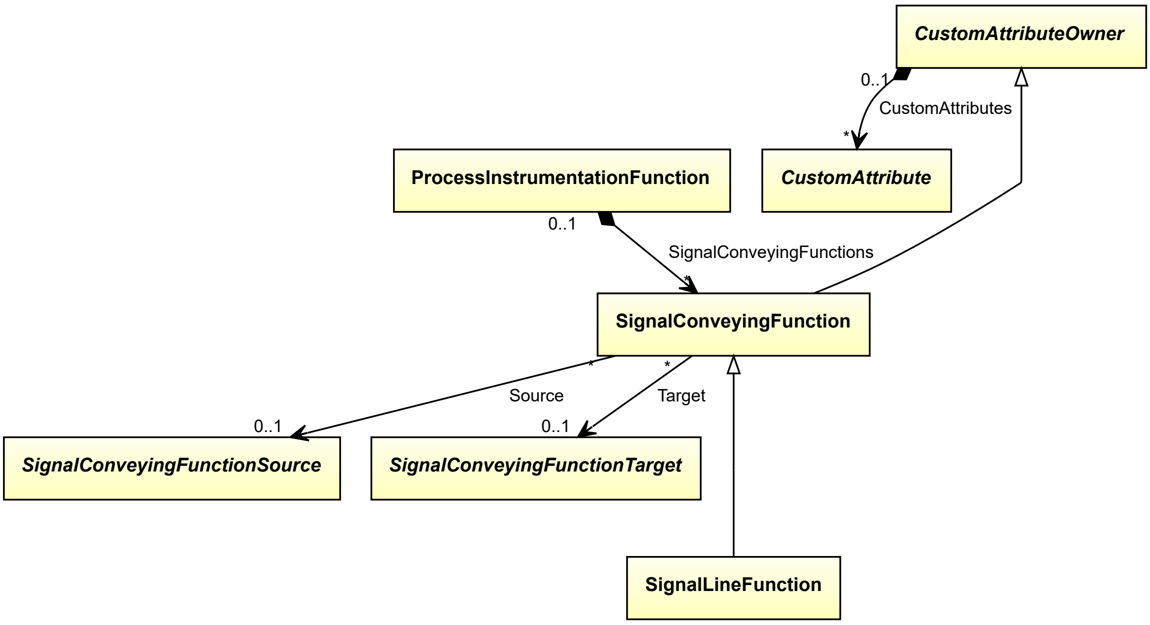
```
<ActuatingElectricalFunction
    ID="actuatingElectricalFunction1"
    ComponentClass="ActuatingElectricalFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/ActuatingElectricalFunction" ...>
  ...
</ActuatingElectricalFunction>
```

9.29. SignalLineFunction

9.29.1 Overview

Class

Information flow function for signals.



Supertypes

- *SignalConveyingFunction*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<InformationFlow>`
RDL reference: SIGNAL LINE FUNCTION
ComponentClass: SignalLineFunction
ComponentClassURI: http://sandbox.dexpi.org/rdl/SignalLineFunction

Example

Example: Implementation in Proteus Schema

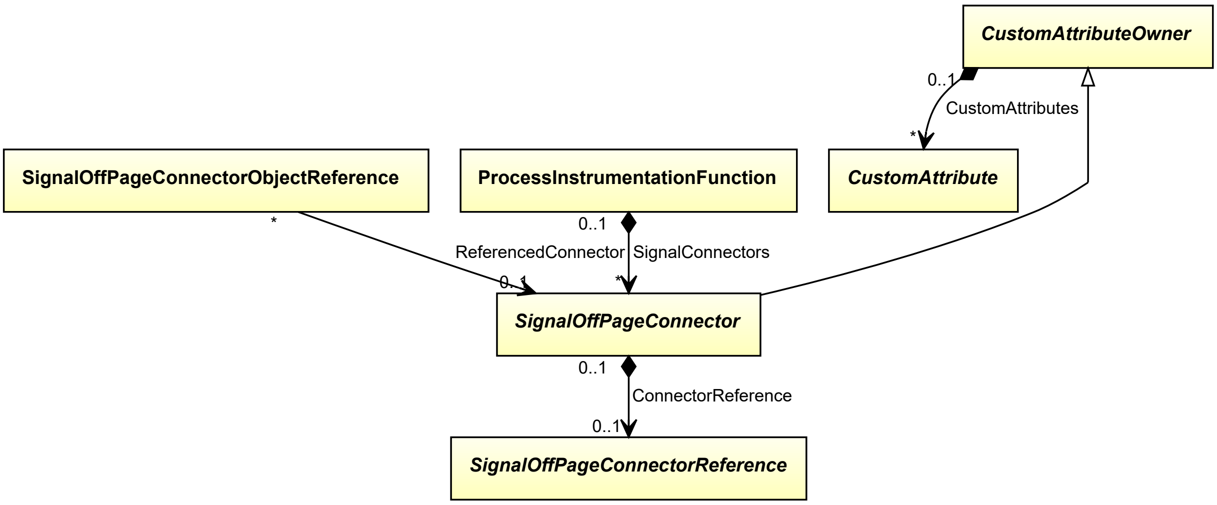
```
<InformationFlow
    ID="signalLineFunction1"
    ComponentClass="SignalLineFunction"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SignalLineFunction" ...>
  ...
</InformationFlow>
```

9.30. SignalOffPageConnector

9.30.1 Overview

Abstract class

A signal connector that indicates that a *SignalConveyingFunction* is continued elsewhere, either on the same P&ID or on another P&ID. Graphically, it is usually represented as an arrow.



Supertypes

- *CustomAttributeOwner*

Subtypes

- *FlowInSignalOffPageConnector*
- *FlowOutSignalOffPageConnector*

Attributes (composition)

Name	Multiplicity	Type
ConnectorReference	0..1	*SignalOffPageConnectorReference*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*. As *SignalOffPageConnector* is abstract, there is no RDL reference for the class itself; the RDL reference depends on the concrete subclass.
Tag: `<InformationFlowOffPageConnector>`
ComponentClass: *depending on subclass*
ComponentClassURI: *depending on subclass*

Example

As *SignalOffPageConnector* is abstract, we consider *FlowInSignalOffPageConnector* as an arbitrary concrete subclass.

flowInSignalOffPageConnector1 : FlowInSignalOffPageConnector

Example: Implementation in Proteus Schema

```
<InformationFlowOffPageConnector
    ID="flowInSignalOffPageConnector1"
    ComponentClass="FlowInSignalOffPageConnector"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/FlowInSignalOffPageConnector" ...>
  ...
</InformationFlowOffPageConnector>
```

9.30.2 ConnectorReference

Attribute (composition)

A reference indicating to which other *SignalOffPageConnector* this *SignalOffPageConnector* is connected.

Multiplicity: 0..1

Type: *SignalOffPageConnectorReference*

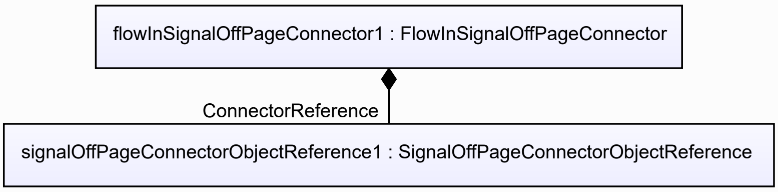
Opposite multiplicity: 0..1

Implementation in Proteus Schema

The attribute is implemented using the *XML hierarchy of the Proteus file*: The element for the attribute value (a *SignalOffPageConnectorReference*) is a child of the `<InformationFlowOffPageConnector>` element for the attribute owner (a *SignalOffPageConnector*).

Example

As the owner type *SignalOffPageConnector* is abstract, we consider *FlowInSignalOffPageConnector* as an arbitrary concrete subclass. As the value type *SignalOffPageConnectorReference* is abstract, we consider *SignalOffPageConnectorObjectReference* as an arbitrary concrete subclass.


flowInSignalOffPageConnector1 : FlowInSignalOffPageConnector
ConnectorReference
signalOffPageConnectorObjectReference1 : SignalOffPageConnectorObjectReference

Example: Implementation in Proteus Schema

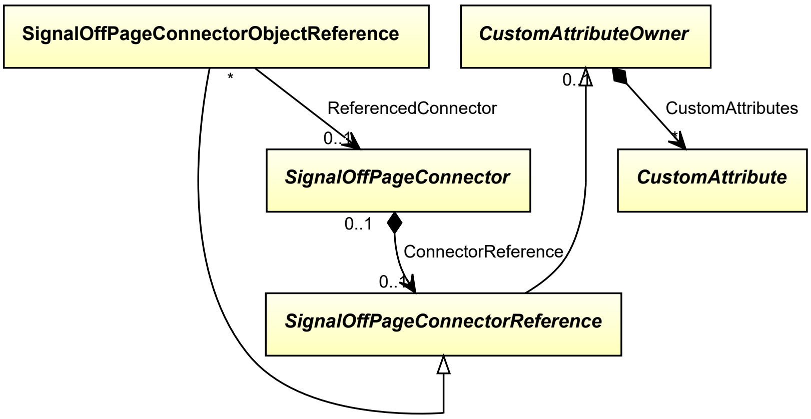
```
<InformationFlowOffPageConnector
    ID="flowInSignalOffPageConnector1"
    ComponentClass="FlowInSignalOffPageConnector"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/FlowInSignalOffPageConnector" ...>
  ...
  <InformationFlowOffPageConnectorReference
      ID="signalOffPageConnectorObjectReference1"
      ComponentClass="SignalOffPageConnectorObjectReference"
      ComponentClassURI="http://sandbox.dexpi.org/rdl/SignalOffPageConnectorObjectReference" ...>
    ...
  <InformationFlowOffPageConnectorReference />
  ...
<InformationFlowOffPageConnector />
```

9.31. SignalOffPageConnectorObjectReference

9.31.1 Overview

Class

A reference to a *SignalOffPageConnector* by an association.



Supertypes

- *SignalOffPageConnectorReference*

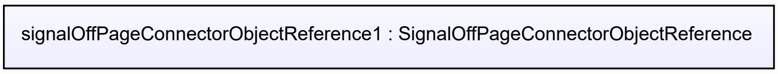
Attributes (reference)

Name	Multiplicity	Type
ReferencedConnector	0..1	*SignalOffPageConnector*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<InformationFlowOffPageConnectorReference>`
RDL reference: SIGNAL OFF PAGE CONNECTOR OBJECT REFERENCE
ComponentClass: SignalOffPageConnectorObjectReference
ComponentClassURI: http://sandbox.dexpi.org/rdl/SignalOffPageConnectorObjectReference

Example


signalOffPageConnectorObjectReference1 : SignalOffPageConnectorObjectReference

Example: Implementation in Proteus Schema

```
<InformationFlowOffPageConnectorReference
    ID="signalOffPageConnectorObjectReference1"
    ComponentClass="SignalOffPageConnectorObjectReference"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SignalOffPageConnectorObjectReference" ...>
  ...
</InformationFlowOffPageConnectorReference>
```

9.31.2 ReferencedConnector

Attribute (reference)

The *SignalOffPageConnector* referenced.

Multiplicity: 0..1

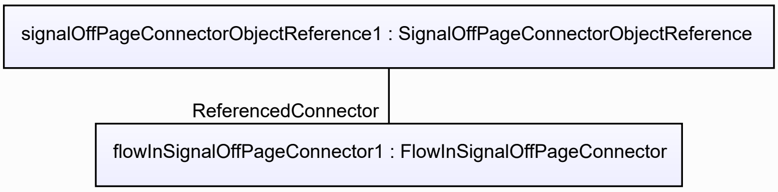
Type: *SignalOffPageConnector*

Opposite multiplicity: 0..*

Implementation in Proteus Schema

The attribute is implemented using *Proteus <Association> elements*.
Association type for the attribute ower: `"refers to"`
Opposite association type: `"is referenced by"`

Example


signalOffPageConnectorObjectReference1 : SignalOffPageConnectorObjectReference
ReferencedConnector
flowInSignalOffPageConnector1 : FlowInSignalOffPageConnector

Example: Implementation in Proteus Schema

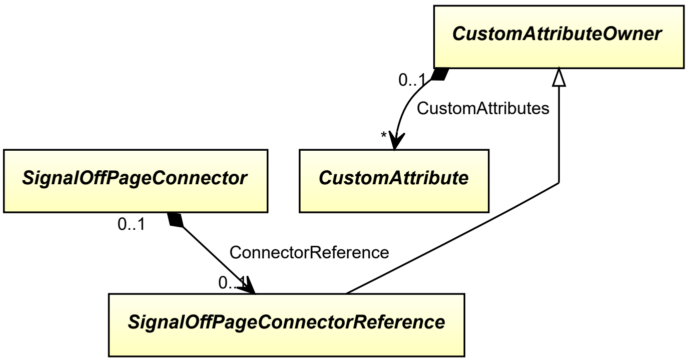
```
<InformationFlowOffPageConnectorReference
    ID="signalOffPageConnectorObjectReference1"
    ComponentClass="SignalOffPageConnectorObjectReference"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SignalOffPageConnectorObjectReference" ...>
  ...
  <Association
    Type="refers to"
    ItemID="flowInSignalOffPageConnector1" />
  ...
<InformationFlowOffPageConnectorReference />
...
<InformationFlowOffPageConnector
    ID="flowInSignalOffPageConnector1"
    ComponentClass="FlowInSignalOffPageConnector"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/FlowInSignalOffPageConnector" ...>
  ...
  <Association
    Type="is referenced by"
    ItemID="signalOffPageConnectorObjectReference1" />
  ...
<InformationFlowOffPageConnector />
```

9.32. SignalOffPageConnectorReference

9.32.1 Overview

Abstract class

A reference to a *SignalOffPageConnector*.



Supertypes

- *CustomAttributeOwner*

Subtypes

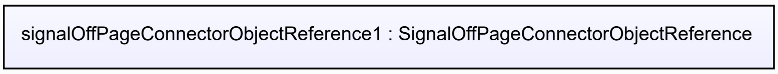
- *SignalOffPageConnectorObjectReference*
- *SignalOffPageConnectorReferenceByNumber*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*. As *SignalOffPageConnectorReference* is abstract, there is no RDL reference for the class itself; the RDL reference depends on the concrete subclass.
Tag: `<InformationFlowOffPageConnectorReference>`
ComponentClass: *depending on subclass*
ComponentClassURI: *depending on subclass*

Example

As *SignalOffPageConnectorReference* is abstract, we consider *SignalOffPageConnectorObjectReference* as an arbitrary concrete subclass.


signalOffPageConnectorObjectReference1 : SignalOffPageConnectorObjectReference

Example: Implementation in Proteus Schema

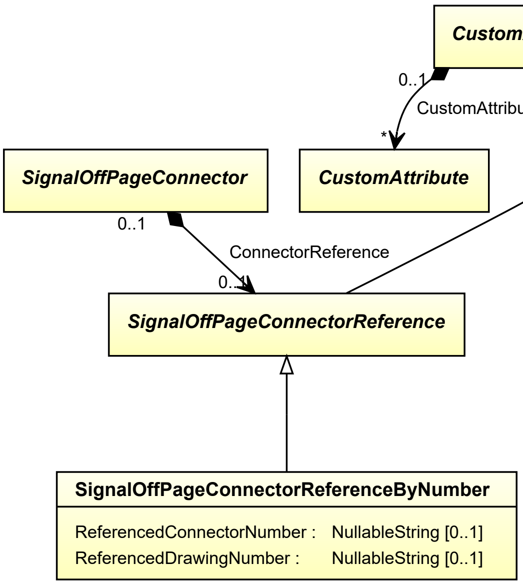
```
<InformationFlowOffPageConnectorReference
    ID="signalOffPageConnectorObjectReference1"
    ComponentClass="SignalOffPageConnectorObjectReference"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SignalOffPageConnectorObjectReference" ...>
  ...
</InformationFlowOffPageConnectorReference>
```

9.33. SignalOffPageConnectorReferenceByNumber

9.33.1 Overview

Class

A reference to a *SignalOffPageConnector* by drawing and connector number.


CustomAttributeOwner
0..1
CustomAttributes
*
SignalOffPageConnector
CustomAttribute
0..1
ConnectorReference
0..1
SignalOffPageConnectorReference
SignalOffPageConnectorReferenceByNumber
ReferencedConnectorNumber : NullableString [0..1]
ReferencedDrawingNumber : NullableString [0..1]

Supertypes

- *SignalOffPageConnectorReference*

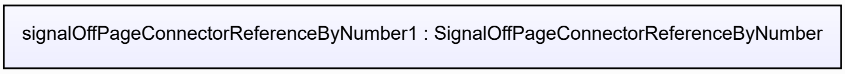
Attributes (data)

Name	Multiplicity	Type
ReferencedConnectorNumber	0..1	*NullableString*
ReferencedDrawingNumber	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: `<InformationFlowOffPageConnectorReference>`
RDL reference: SIGNAL OFF PAGE CONNECTOR REFERENCE BY NUMBER
ComponentClass: SignalOffPageConnectorReferenceByNumber
ComponentClassURI: http://sandbox.dexpi.org/rdl/SignalOffPageConnectorReferenceByNumber

Example


signalOffPageConnectorReferenceByNumber1 : SignalOffPageConnectorReferenceByNumber

Example: Implementation in Proteus Schema

```
<InformationFlowOffPageConnectorReference
    ID="signalOffPageConnectorReferenceByNumber1"
    ComponentClass="SignalOffPageConnectorReferenceByNumber"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SignalOffPageConnectorReferenceByNumber" ...>
  ...
</InformationFlowOffPageConnectorReference>
```

9.33.2 ReferencedConnectorNumber

Attribute (data)

The connector number of the referenced connector.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: REFERENCED CONNECTOR NUMBER ASSIGNMENT CLASS
Name: ReferencedConnectorNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ReferencedConnectorNumberAssignmentClass

Example

"97" (*String*)

Example: Implementation in Proteus Schema

```
<InformationFlowOffPageConnectorReference
    ID="signalOffPageConnectorReferenceByNumber1"
    ComponentClass="SignalOffPageConnectorReferenceByNumber"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SignalOffPageConnectorReferenceByNumber" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ReferencedConnectorNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ReferencedConnectorNumberAssignmentClass"
      Format="string"
      Value="97" />
    ...
  </GenericAttributes>
  ...
</InformationFlowOffPageConnectorReference>
```

9.33.3 ReferencedDrawingNumber

Attribute (data)

The *DrawingNumber* of the PID that contains the referenced connector.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: REFERENCED DRAWING NUMBER ASSIGNMENT CLASS
Name: ReferencedDrawingNumberAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/ReferencedDrawingNumberAssignmentClass

Example

“123/A93” (*String*)

Example: Implementation in Proteus Schema

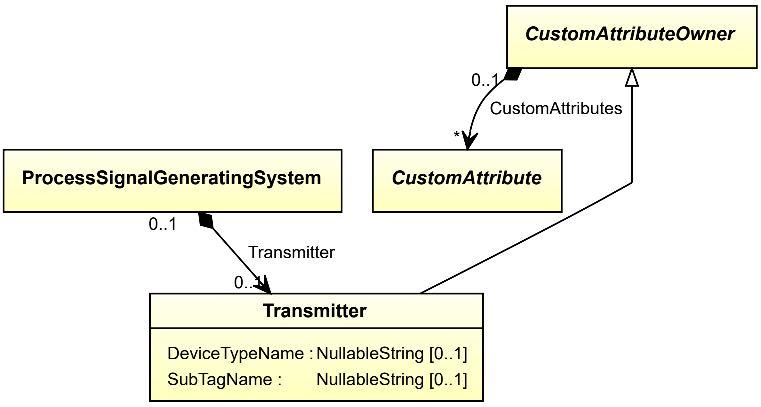
```
<InformationFlowOffPageConnectorReference
    ID="signalOffPageConnectorReferenceByNumber1"
    ComponentClass="SignalOffPageConnectorReferenceByNumber"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/SignalOffPageConnectorReferenceByNumber" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="ReferencedDrawingNumberAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/ReferencedDrawingNumberAssignmentClass"
      Format="string"
      Value="123/A93" />
    ...
  </GenericAttributes>
  ...
</InformationFlowOffPageConnectorReference>
```

9.34. Transmitter

9.34.1 Overview

Class

A detecting instrument that generates a process variable signal and converts it into an output signal.


CustomAttributeOwner
0..1
CustomAttributes
*
ProcessSignalGeneratingSystem
CustomAttribute
0..1
Transmitter
0..1
Transmitter
DeviceTypeName : NullableString [0..1]
SubTagName : NullableString [0..1]

Supertypes

- *CustomAttributeOwner*

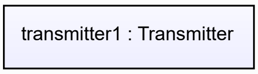
Attributes (data)

Name	Multiplicity	Type
DeviceTypeName	0..1	*NullableString*
SubTagName	0..1	*NullableString*

Implementation in Proteus Schema

The class is implemented as a *Proteus XML element with RDL reference*.
Tag: <ProcessSignalGeneratingSystemComponent>
RDL reference: TRANSMITTER
ComponentClass: Transmitter
ComponentClassURI: http://data.posccaesar.org/rdl/RDS267929

Example



Example: Implementation in Proteus Schema

```
<ProcessSignalGeneratingSystemComponent
    ID="transmitter1"
    ComponentClass="Transmitter"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267929" ...>
  ...
</ProcessSignalGeneratingSystemComponent>
```

9.34.2 DeviceTypeName

Attribute (data)

The device type of the *Transmitter*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: DEVICE TYPE NAME ASSIGNMENT CLASS
Name: DeviceTypeNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/DeviceTypeNameAssignmentClass

Example

“pressure transmitter” (*String*)

Example: Implementation in Proteus Schema

```
<ProcessSignalGeneratingSystemComponent
    ID="transmitter1"
    ComponentClass="Transmitter"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267929" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="DeviceTypeNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/DeviceTypeNameAssignmentClass"
      Format="string"
      Value="pressure transmitter" />
    ...
  </GenericAttributes>
  ...
</ProcessSignalGeneratingSystemComponent>
```

9.34.3 SubTagName

Attribute (data)

The sub tag name of the *Transmitter*.

Multiplicity: 0..1

Type: *NullableString*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: SUB TAG NAME ASSIGNMENT CLASS
Name: SubTagNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/SubTagNameAssignmentClass

Example

“ST1” (*String*)

Example: Implementation in Proteus Schema

```
<ProcessSignalGeneratingSystemComponent
    ID="transmitter1"
    ComponentClass="Transmitter"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS267929" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="SubTagNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/SubTagNameAssignmentClass"
      Format="string"
      Value="ST1" />
    ...
  </GenericAttributes>
  ...
</ProcessSignalGeneratingSystemComponent>
```

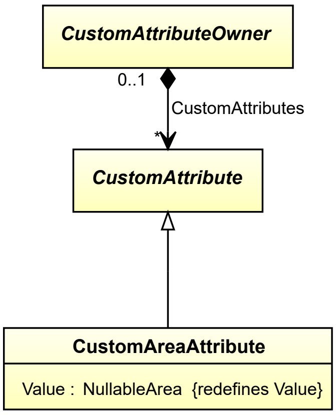
Package Customization 10

10.1. CustomAreaAttribute

10.1.1 Overview

Class

A custom attribute with *Value* type *NullableArea*.



Supertypes

- *CustomAttribute*

Attributes (data)

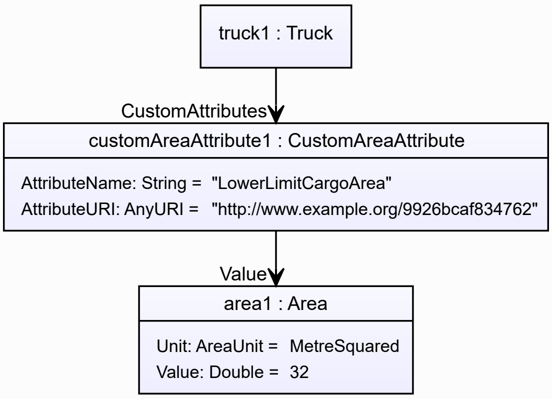
Name	Multiplicity	Type
Value	1	*NullableArea*

Implementation in Proteus Schema

CustomAreaAttribute is implemented as a *custom generic attribute for physical quantities*.
RDL reference: AREA
Type: Area
TypeURI: http://data.posccaesar.org/rdl/RDS349874

Example

As *CustomAttributeOwner* is abstract, we consider *Truck* as an arbitrary concrete subclass.
The *Truck* truck1 has a *CustomAreaAttribute* with *AttributeName* “LowerLimitCargoArea” and an (arbitrary) *AttributeURI* “http://www.example.org/9926bcaf834762”. The *Value* is 32 m^2.


truck1 : Truck
CustomAttributes
customAreaAttribute1 : CustomAreaAttribute
AttributeName: String = "LowerLimitCargoArea"
AttributeURI: AnyURI = "http://www.example.org/9926bcaf834762"
Value
area1 : Area
Unit: AreaUnit = MetreSquared
Value: Double = 32

Example: Implementation in Proteus Schema

```
<Equipment
    ID="truck1"
    ComponentClass="Truck"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11524112" ...>
  ...
  <GenericAttributes Set="DexpiCustomAttributes" ...>
    <GenericAttribute
      Name="LowerLimitCargoArea"
      AttributeURI="http://www.example.org/9926bcaf834762"
      Format="double"
      Type="Area"
      TypeURI="http://data.posccaesar.org/rdl/RDS349874"
      Value="32"
      Units="MetreSquared"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1358009" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

10.1.2 Value

Attribute (data)

The value of the *CustomAreaAttribute*.

Multiplicity: 1

Type: *NullableArea*

Redefines: *Value* (inherited from *CustomAttribute*)

Implementation in Proteus Schema

See implementation of *CustomAreaAttribute*.

Example

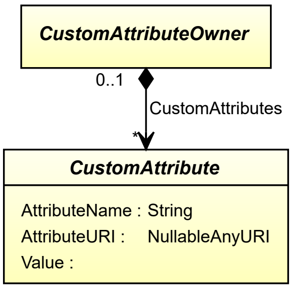
See the example for *CustomAreaAttribute*.

10.2. CustomAttribute

10.2.1 Overview

Abstract class

A custom attribute.



Subtypes

- *CustomAreaAttribute*
- *CustomElectricalFrequencyAttribute*
- *CustomForceAttribute*
- *CustomHeatTransferCoefficientAttribute*
- *CustomIntegerAttribute*
- *CustomLengthAttribute*
- *CustomMassAttribute*
- *CustomMassFlowRateAttribute*
- *CustomMultiLanguageStringAttribute*
- *CustomNumberPerTimeIntervalAttribute*
- *CustomPercentageAttribute*
- *CustomPowerAttribute*
- *CustomPressureAbsoluteAttribute*
- *CustomPressureGaugeAttribute*
- *CustomRotationalFrequencyAttribute*
- *CustomStringAttribute*
- *CustomTemperatureAttribute*
- *CustomVoltageAttribute*
- *CustomVolumeAttribute*
- *CustomVolumeFlowRateAttribute*

Attributes (data)

Name	Multiplicity	Type
AttributeName	1	*String*
AttributeURI	1	*NullableAnyURI*
Value	1	-

Implementation in Proteus Schema

Implementation is subclass-specific.

10.2.2 AttributeName

Attribute (data)

Multiplicity: 1

Type: *String*

10.2.3 AttributeURI

Attribute (data)

Multiplicity: 1

Type: *NullableAnyURI*

10.2.4 Value

Attribute (data)

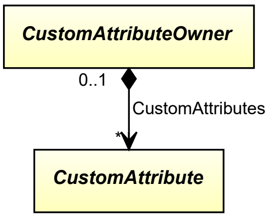
Multiplicity: 1

10.3. CustomAttributeOwner

10.3.1 Overview

Abstract class

An object that can have custom attributes.



Subtypes

- *ActuatingElectricalFunction*
- *ActuatingElectricalSystem*
- *ActuatingFunction*
- *ActuatingSystem*
- *AgitatorRotor*
- *BriquettingRoller*
- *Chamber*
- *ColumnInternalsArrangement*
- *ColumnSection*
- *ControlledActuator*
- *CoolingTowerRotor*
- *CrusherElement*
- *CustomActuatingElectricalSystemComponent*
- *CustomActuatingSystemComponent*
- *CustomProcessSignalGeneratingSystemComponent*
- *Displacer*
- *DryingChamber*
- *ElectronicFrequencyConverter*
- *FilterUnit*
- *FilteringCentrifugeDrum*
- *GearBox*
- *GrindingElement*
- *HeatExchangerRotor*
- *Impeller*
- *InstrumentationLoopFunction*
- *MetaData*
- *MixingElementAssembly*
- *MotorAsComponent*
- *Nozzle*
- *OperatedValveReference*
- *PelletizerDisc*
- *Pipe*
- *PipeOffPageConnector*
- *PipeOffPageConnectorReference*
- *PipingComponent*
- *PipingNetworkSegment*
- *PipingNetworkSystem*
- *PipingNode*
- *PlantStructureItem*

- *Positioner*
- *PrimaryElement*
- *ProcessInstrumentationFunction*
- *ProcessSignalGeneratingFunction*
- *ProcessSignalGeneratingSystem*
- *PropertyBreak*
- *Screw*
- *SedimentalCentrifugeDrum*
- *SieveElement*
- *SignalConveyingFunction*
- *SignalOffPageConnector*
- *SignalOffPageConnectorReference*
- *SprayNozzle*
- *TaggedPlantItem*
- *Transmitter*
- *TubeBundle*

Attributes (composition)

Name	Multiplicity	Type
CustomAttributes	*	*CustomAttribute*

Implementation in Proteus Schema

Implementation is subclass-specific.

Example

See the examples for the subclasses of *CustomAttribute*.

10.3.2 CustomAttributes

Attribute (composition)

Multiplicity: *

Type: *CustomAttribute*

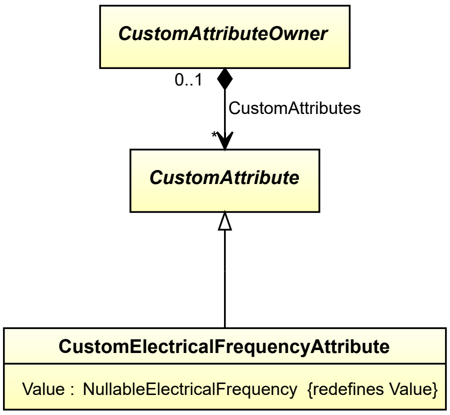
Opposite multiplicity: 0..1

10.4. CustomElectricalFrequencyAttribute

10.4.1 Overview

Class

A custom attribute with *Value* type *NullableElectricalFrequency*.


CustomAttributeOwner
0..1
CustomAttributes
*
CustomAttribute
CustomElectricalFrequencyAttribute
Value : NullableElectricalFrequency {redefines Value}

Supertypes

- *CustomAttribute*

Attributes (data)

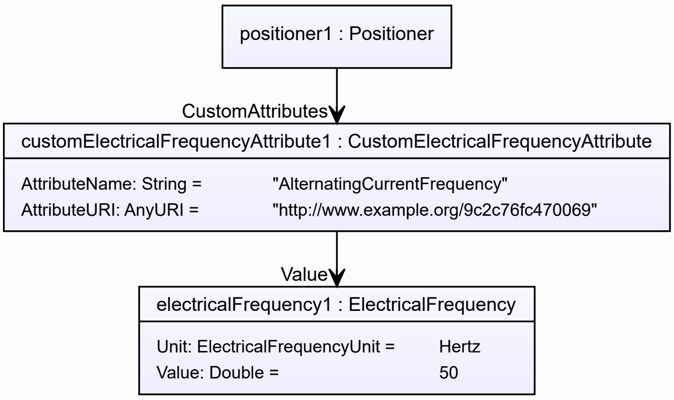
Name	Multiplicity	Type
Value	1	*NullableElectricalFrequency*

Implementation in Proteus Schema

CustomElectricalFrequencyAttribute is implemented as a *custom generic attribute for physical quantities*.
RDL reference: ELECTRICAL FREQUENCY
Type: ElectricalFrequency
TypeURI: http://data.posccaesar.org/rdl/RDS401399

Example

As *CustomAttributeOwner* is abstract, we consider *Positioner* as an arbitrary concrete subclass. The *Positioner* positioner1 has a *CustomElectricalFrequencyAttribute* with *AttributeName* “AlternatingCurrentFrequency” and an (arbitrary) *AttributeURI* “http://www.example.org/9c2c76fc470069”. The *Value* is 50 Hz.


positioner1 : Positioner
CustomAttributes
customElectricalFrequencyAttribute1 : CustomElectricalFrequencyAttribute
AttributeName: String = "AlternatingCurrentFrequency"
AttributeURI: AnyURI = "http://www.example.org/9c2c76fc470069"
Value
electricalFrequency1 : ElectricalFrequency
Unit: ElectricalFrequencyUnit = Hertz
Value: Double = 50

Example: Implementation in Proteus Schema

```
<ActuatingSystemComponent
    ID="positioner1"
    ComponentClass="Positioner"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Positioner" ...>
  ...
  <GenericAttributes Set="DexpiCustomAttributes" ...>
    <GenericAttribute
      Name="AlternatingCurrentFrequency"
      AttributeURI="http://www.example.org/9c2c76fc470069"
      Format="double"
      Type="ElectricalFrequency"
      TypeURI="http://data.posccaesar.org/rdl/RDS401399"
      Value="50"
      Units="Hertz"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1326464" />
    ...
  </GenericAttributes>
  ...
</ActuatingSystemComponent>
```

10.4.2 Value

Attribute (data)

The value of the *CustomElectricalFrequencyAttribute*.

Multiplicity: 1

Type: *NullableElectricalFrequency*

Redefines: *Value* (inherited from *CustomAttribute*)

Implementation in Proteus Schema

See implementation of *CustomElectricalFrequencyAttribute*.

Example

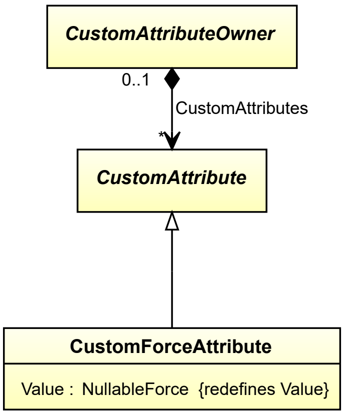
See the example for *CustomElectricalFrequencyAttribute*.

10.5. CustomForceAttribute

10.5.1 Overview

Class

A custom attribute with *Value* type *NullableForce*.



Supertypes

- *CustomAttribute*

Attributes (data)

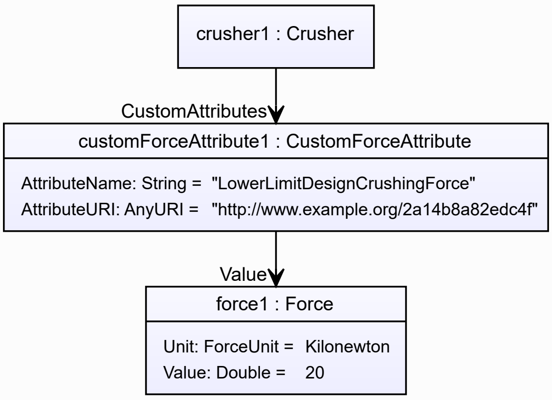
Name	Multiplicity	Type
Value	1	*NullableForce*

Implementation in Proteus Schema

CustomForceAttribute is implemented as a *custom generic attribute for physical quantities*.
RDL reference: FORCE
Type: Force
TypeURI: http://data.posccaesar.org/rdl/RDS351854

Example

As *CustomAttributeOwner* is abstract, we consider *Crusher* as an arbitrary concrete subclass.
The *Crusher* crusher1 has a *CustomForceAttribute* with *AttributeName* "LowerLimitDesignCrushingForce" and an (arbitrary) *AttributeURI* "http://www.example.org/2a14b8a82edc4f". The *Value* is 20 kN.


crusher1 : Crusher
CustomAttributes
customForceAttribute1 : CustomForceAttribute
AttributeName: String = "LowerLimitDesignCrushingForce"
AttributeURI: AnyURI = "http://www.example.org/2a14b8a82edc4f"
Value
force1 : Force
Unit: ForceUnit = Kilonewton
Value: Double = 20

Example: Implementation in Proteus Schema

```
<Equipment
    ID="crusher1"
    ComponentClass="Crusher"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11589940" ...>
  ...
  <GenericAttributes Set="DexpiCustomAttributes" ...>
    <GenericAttribute
      Name="LowerLimitDesignCrushingForce"
      AttributeURI="http://www.example.org/2a14b8a82edc4f"
      Format="double"
      Type="Force"
      TypeURI="http://data.posccaesar.org/rdl/RDS351854"
      Value="20"
      Units="Kilonewton"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1351034" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

10.5.2 Value

Attribute (data)

The value of the *CustomForceAttribute*.

Multiplicity: 1

Type: *NullableForce*

Redefines: *Value* (inherited from *CustomAttribute*)

Implementation in Proteus Schema

See implementation of *CustomForceAttribute*.

Example

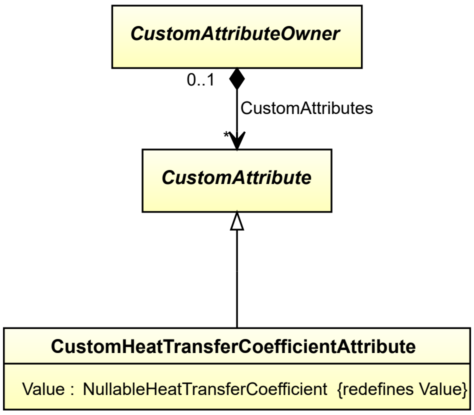
See the example for *CustomForceAttribute*.

10.6. CustomHeatTransferCoefficientAttribute

10.6.1 Overview

Class

A custom attribute with *Value* type *NullableHeatTransferCoefficient*.


CustomAttributeOwner
0..1
CustomAttributes
*
CustomAttribute
CustomHeatTransferCoefficientAttribute
Value : NullableHeatTransferCoefficient {redefines Value}

Supertypes

- *CustomAttribute*

Attributes (data)

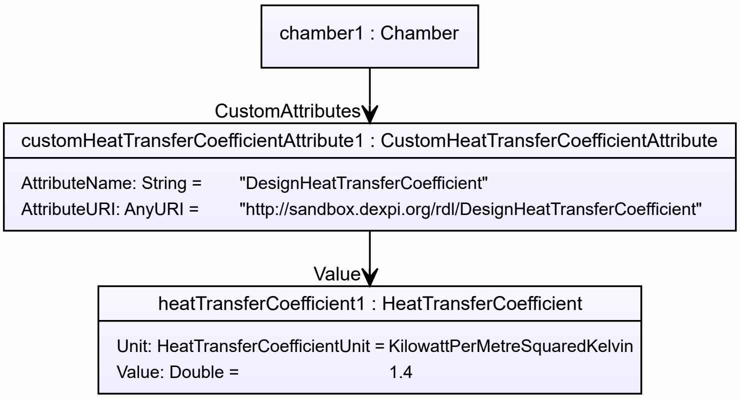
Name	Multiplicity	Type
Value	1	*NullableHeatTransferCoefficient*

Implementation in Proteus Schema

CustomHeatTransferCoefficientAttribute is implemented as a *custom generic attribute for physical quantities*.
RDL reference: HEAT TRANSFER COEFFICIENT
Type: HeatTransferCoefficient
TypeURI: http://data.posccaesar.org/rdl/RDS352304

Example

As *CustomAttributeOwner* is abstract, we consider *Chamber* as an arbitrary concrete subclass.
The *Chamber* chamber1 has a *CustomHeatTransferCoefficientAttribute* with *AttributeName* "DesignHeatTransferCoefficient" and *AttributeURI* "http://sandbox.dexpi.org/rdl/DesignHeatTransferCoefficient". The *Value* is 1.4 kW/(m^2 · K).


chamber1 : Chamber
CustomAttributes
customHeatTransferCoefficientAttribute1 : CustomHeatTransferCoefficientAttribute
AttributeName: String = "DesignHeatTransferCoefficient"
AttributeURI: AnyURI = "http://sandbox.dexpi.org/rdl/DesignHeatTransferCoefficient"
Value
heatTransferCoefficient1 : HeatTransferCoefficient
Unit: HeatTransferCoefficientUnit = KilowattPerMetreSquaredKelvin
Value: Double = 1.4

Example: Implementation in Proteus Schema

```
<Equipment
    ID="chamber1"
    ComponentClass="Chamber"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS903151421" ...>
  ...
  <GenericAttributes Set="DexpiCustomAttributes" ...>
    <GenericAttribute
      Name="DesignHeatTransferCoefficient"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignHeatTransferCoefficient"
      Format="double"
      Type="HeatTransferCoefficient"
      TypeURI="http://data.posccaesar.org/rdl/RDS352304"
      Value="1.4"
      Units="KilowattPerMetreSquaredKelvin"
      UnitsURI="http://data.posccaesar.org/rdl/RDS43167567170" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

10.6.2 Value

Attribute (data)

The value of the *CustomHeatTransferCoefficientAttribute*.

Multiplicity: 1

Type: *NullableHeatTransferCoefficient*

Redefines: *Value* (inherited from *CustomAttribute*)

Implementation in Proteus Schema

See implementation of *CustomHeatTransferCoefficientAttribute*.

Example

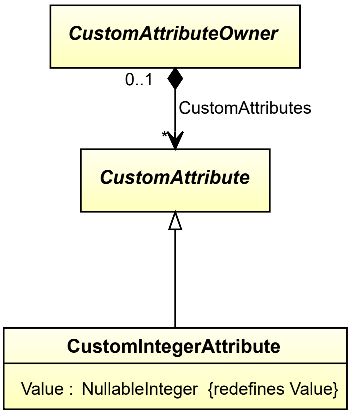
See the example for *CustomHeatTransferCoefficientAttribute*.

10.7. CustomIntegerAttribute

10.7.1 Overview

Class

A custom attribute with *Value* type *NullableInteger*.



Supertypes

- *CustomAttribute*

Attributes (data)

Name	Multiplicity	Type
Value	1	*NullableInteger*

Implementation in Proteus Schema

CustomIntegerAttribute is implemented as a *custom generic attribute for integer values*.
RDL reference: INTEGER NUMBER
Type: IntegerNumber
TypeURI: http://data.posccaesar.org/rdl/RDS47466171135

10.7.2 Value

Attribute (data)

The value of the *CustomIntegerAttribute*.

Multiplicity: 1

Type: *NullableInteger*

Redefines: *Value* (inherited from *CustomAttribute*)

Implementation in Proteus Schema

See implementation of *CustomIntegerAttribute*.

Example

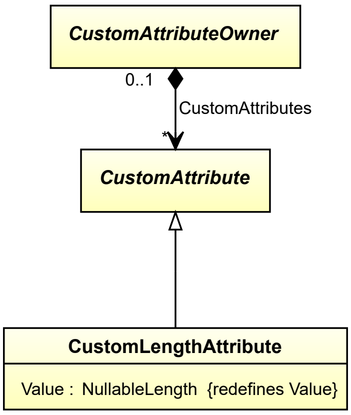
See the example for *CustomIntegerAttribute*.

10.8. CustomLengthAttribute

10.8.1 Overview

Class

A custom attribute with *Value* type *NullableLength*.



Supertypes

- *CustomAttribute*

Attributes (data)

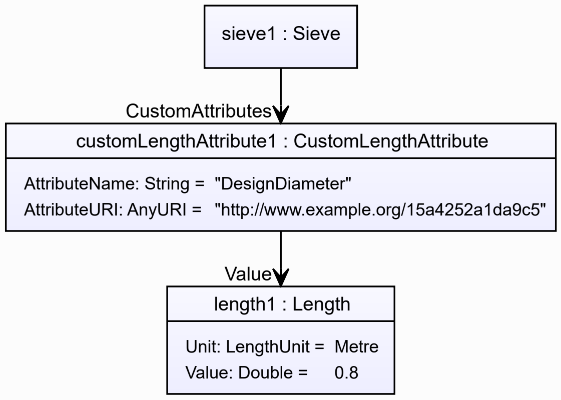
Name	Multiplicity	Type
Value	1	*NullableLength*

Implementation in Proteus Schema

CustomLengthAttribute is implemented as a *custom generic attribute for physical quantities*.
RDL reference: LENGTH
Type: Length
TypeURI: http://data.posccaesar.org/rdl/RDS373094

Example

As *CustomAttributeOwner* is abstract, we consider *Sieve* as an arbitrary concrete subclass.
The *Sieve* sieve1 has a *CustomLengthAttribute* with *AttributeName* “DesignDiameter” and an (arbitrary) *AttributeURI* “http://www.example.org/15a4252a1da9c5”. The *Value* is 0.8 m.


sieve1 : Sieve
CustomAttributes
customLengthAttribute1 : CustomLengthAttribute
AttributeName: String = "DesignDiameter"
AttributeURI: AnyURI = "http://www.example.org/15a4252a1da9c5"
Value
length1 : Length
Unit: LengthUnit = Metre
Value: Double = 0.8

Example: Implementation in Proteus Schema

```
<Equipment
    ID="sieve1"
    ComponentClass="Sieve"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Sieve" ...>
  ...
  <GenericAttributes Set="DexpiCustomAttributes" ...>
    <GenericAttribute
      Name="DesignDiameter"
      AttributeURI="http://www.example.org/15a4252a1da9c5"
      Format="double"
      Type="Length"
      TypeURI="http://data.posccaesar.org/rdl/RDS373094"
      Value="0.8"
      Units="Metre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1332674" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

10.8.2 Value

Attribute (data)

The value of the *CustomLengthAttribute*.

Multiplicity: 1

Type: *NullableLength*

Redefines: *Value* (inherited from *CustomAttribute*)

Implementation in Proteus Schema

See implementation of *CustomLengthAttribute*.

Example

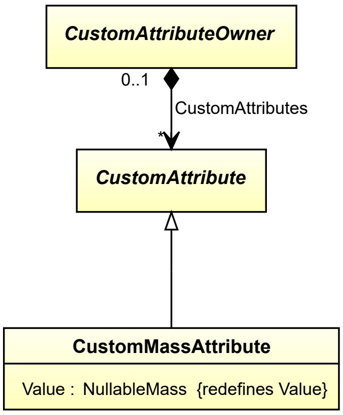
See the example for *CustomLengthAttribute*.

10.9. CustomMassAttribute

10.9.1 Overview

Class

A custom attribute with *Value* type *NullableMass*.



Supertypes

- *CustomAttribute*

Attributes (data)

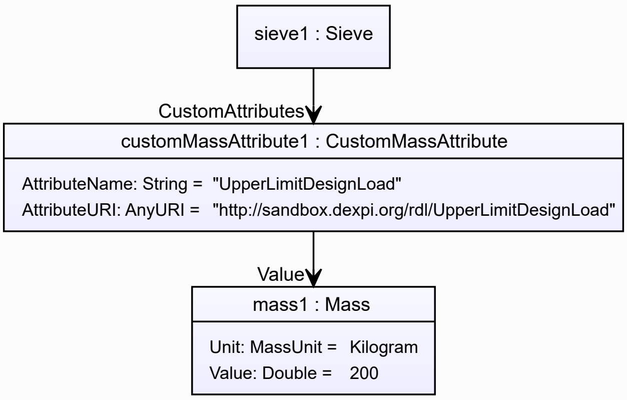
Name	Multiplicity	Type
Value	1	*NullableMass*

Implementation in Proteus Schema

CustomMassAttribute is implemented as a *custom generic attribute for physical quantities*.
RDL reference: MASS
Type: Mass
TypeURI: http://data.posccaesar.org/rdl/RDS353339

Example

As *CustomAttributeOwner* is abstract, we consider *Sieve* as an arbitrary concrete subclass.
The *Sieve* sieve1 has a *CustomMassAttribute* with *AttributeName* “UpperLimitDesignLoad” and *AttributeURI* “http://sandbox.dexpi.org/rdl/UpperLimitDesignLoad”. The *Value* is 200 kg.


sieve1 : Sieve
CustomAttributes
customMassAttribute1 : CustomMassAttribute
AttributeName: String = "UpperLimitDesignLoad"
AttributeURI: AnyURI = "http://sandbox.dexpi.org/rdl/UpperLimitDesignLoad"
Value
mass1 : Mass
Unit: MassUnit = Kilogram
Value: Double = 200

Example: Implementation in Proteus Schema

```
<Equipment
    ID="sieve1"
    ComponentClass="Sieve"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/Sieve" ...>
  ...
  <GenericAttributes Set="DexpiCustomAttributes" ...>
    <GenericAttribute
      Name="UpperLimitDesignLoad"
      AttributeURI="http://sandbox.dexpi.org/rdl/UpperLimitDesignLoad"
      Format="double"
      Type="Mass"
      TypeURI="http://data.posccaesar.org/rdl/RDS353339"
      Value="200"
      Units="Kilogram"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1328669" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

10.9.2 Value

Attribute (data)

The value of the *CustomMassAttribute*.

Multiplicity: 1

Type: *NullableMass*

Redefines: *Value* (inherited from *CustomAttribute*)

Implementation in Proteus Schema

See implementation of *CustomMassAttribute*.

Example

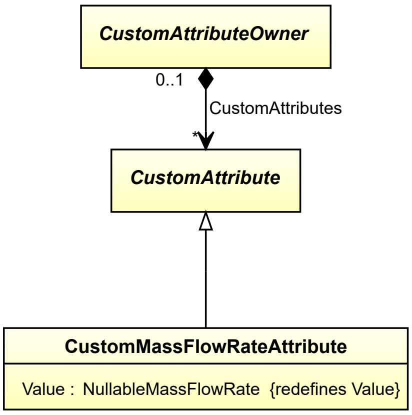
See the example for *CustomMassAttribute*.

10.10. CustomMassFlowRateAttribute

10.10.1 Overview

Class

A custom attribute with *Value* type *NullableMassFlowRate*.


CustomAttributeOwner
0..1
CustomAttributes
*
CustomAttribute
CustomMassFlowRateAttribute
Value : NullableMassFlowRate {redefines Value}

Supertypes

- *CustomAttribute*

Attributes (data)

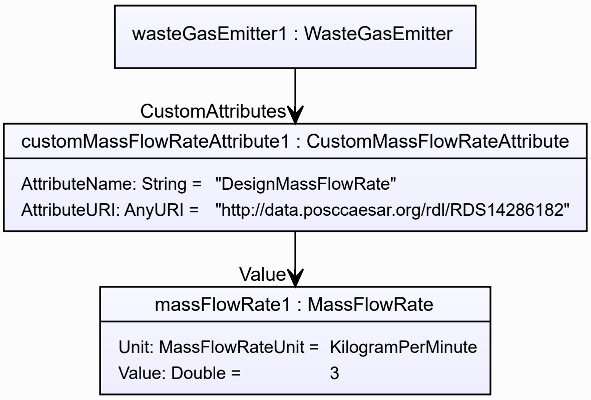
Name	Multiplicity	Type
Value	1	*NullableMassFlowRate*

Implementation in Proteus Schema

CustomMassFlowRateAttribute is implemented as a *custom generic attribute for physical quantities*.
RDL reference: MASS FLOW RATE
Type: MassFlowRate
TypeURI: http://data.posccaesar.org/rdl/RDS380789

Example

As *CustomAttributeOwner* is abstract, we consider *WasteGasEmitter* as an arbitrary concrete subclass.
The *WasteGasEmitter* wasteGasEmitter1 has a *CustomMassFlowRateAttribute* with *AttributeName* “DesignMassFlowRate” and *AttributeURI* “http://data.posccaesar.org/rdl/RDS14286182”. The *Value* is 3 kg/min.


wasteGasEmitter1 : WasteGasEmitter
CustomAttributes
customMassFlowRateAttribute1 : CustomMassFlowRateAttribute
AttributeName: String = "DesignMassFlowRate"
AttributeURI: AnyURI = "http://data.posccaesar.org/rdl/RDS14286182"
Value
massFlowRate1 : MassFlowRate
Unit: MassFlowRateUnit = KilogramPerMinute
Value: Double = 3

Example: Implementation in Proteus Schema

```
<Equipment
    ID="wasteGasEmitter1"
    ComponentClass="WasteGasEmitter"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/WasteGasEmitter" ...>
  ...
  <GenericAttributes Set="DexpiCustomAttributes" ...>
    <GenericAttribute
      Name="DesignMassFlowRate"
      AttributeURI="http://data.posccaesar.org/rdl/RDS14286182"
      Format="double"
      Type="MassFlowRate"
      TypeURI="http://data.posccaesar.org/rdl/RDS380789"
      Value="3"
      Units="KilogramPerMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1350719" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

10.10.2 Value

Attribute (data)

The value of the *CustomMassFlowRateAttribute*.

Multiplicity: 1

Type: *NullableMassFlowRate*

Redefines: *Value* (inherited from *CustomAttribute*)

Implementation in Proteus Schema

See implementation of *CustomMassFlowRateAttribute*.

Example

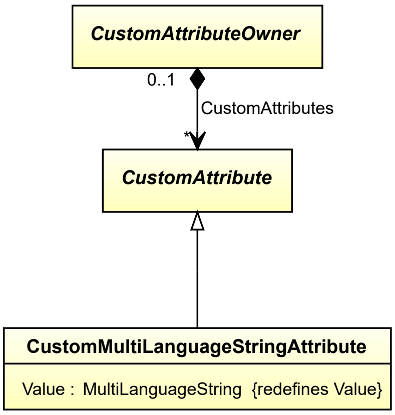
See the example for *CustomMassFlowRateAttribute*.

10.11. CustomMultiLanguageStringAttribute

10.11.1 Overview

Class

A custom attribute with *Value* type *MultiLanguageString*.


CustomAttributeOwner
0..1
CustomAttributes
*
CustomAttribute
CustomMultiLanguageStringAttribute
Value : MultiLanguageString {redefines Value}

Supertypes

- *CustomAttribute*

Attributes (data)

Name	Multiplicity	Type
Value	1	*MultiLanguageString*

Implementation in Proteus Schema

CustomMultiLanguageStringAttribute is implemented as a *set of DEXPI generic attributes for multi-language string values*.
RDL reference: MULTI LANGUAGE STRING
Type: MultiLanguageString
TypeURI: http://sandbox.dexpi.org/rdl/MultiLanguageString

10.11.2 Value

Attribute (data)

The value of the *CustomMultiLanguageStringAttribute*.

Multiplicity: 1

Type: *MultiLanguageString*

Redefines: *Value* (inherited from *CustomAttribute*)

Implementation in Proteus Schema

See implementation of *CustomMultiLanguageStringAttribute*.

Example

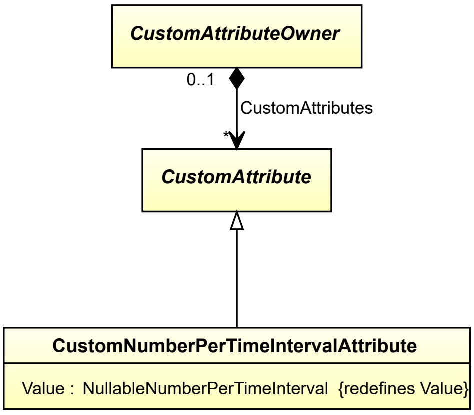
See the example for *CustomMultiLanguageStringAttribute*.

10.12. CustomNumberPerTimeIntervalAttribute

10.12.1 Overview

Class

A custom attribute with *Value* type *NullableNumberPerTimeInterval*.


CustomAttributeOwner
0..1
CustomAttributes
*
CustomAttribute
CustomNumberPerTimeIntervalAttribute
Value : NullableNumberPerTimeInterval {redefines Value}

Supertypes

- *CustomAttribute*

Attributes (data)

Name	Multiplicity	Type
Value	1	*NullableNumberPerTimeInterval*

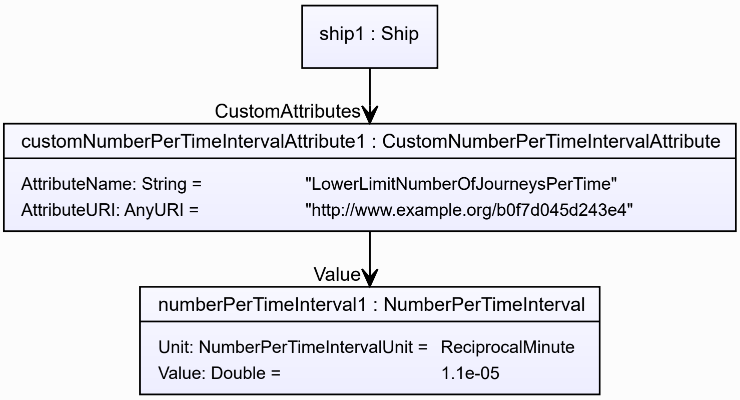
Implementation in Proteus Schema

CustomNumberPerTimeIntervalAttribute is implemented as a *custom generic attribute for physical quantities*.
RDL reference: NUMBER PER TIME INTERVAL
Type: NumberPerTimeInterval
TypeURI: http://sandbox.dexpi.org/rdl/NumberPerTimeInterval

Example

As *CustomAttributeOwner* is abstract, we consider *Ship* as an arbitrary concrete subclass.
The *Ship* ship1 has a *CustomNumberPerTimeIntervalAttribute* with *AttributeName* “LowerLimitNumberOfJour-

neysPerTime" and an (arbitrary) *AttributeURI* "http://www.example.org/b0f7d045d243e4". The *Value* is 1.1e-05 min^{-1}.


ship1 : Ship
CustomAttributes
customNumberPerTimeIntervalAttribute1 : CustomNumberPerTimeIntervalAttribute
AttributeName: String = "LowerLimitNumberOfJourneysPerTime"
AttributeURI: AnyURI = "http://www.example.org/b0f7d045d243e4"
Value
numberPerTimeInterval1 : NumberPerTimeInterval
Unit: NumberPerTimeIntervalUnit = ReciprocalMinute
Value: Double = 1.1e-05

Example: Implementation in Proteus Schema

```
<Equipment
    ID="ship1"
    ComponentClass="Ship"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS11523932" ...>
  ...
  <GenericAttributes Set="DexpiCustomAttributes" ...>
    <GenericAttribute
      Name="LowerLimitNumberOfJourneysPerTime"
      AttributeURI="http://www.example.org/b0f7d045d243e4"
      Format="double"
      Type="NumberPerTimeInterval"
      TypeURI="http://sandbox.dexpi.org/rdl/NumberPerTimeInterval"
      Value="1.1e-05"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

10.12.2 Value

Attribute (data)

The value of the *CustomNumberPerTimeIntervalAttribute*.

Multiplicity: 1

Type: *NullableNumberPerTimeInterval*

Redefines: *Value* (inherited from *CustomAttribute*)

Implementation in Proteus Schema

See implementation of *CustomNumberPerTimeIntervalAttribute*.

Example

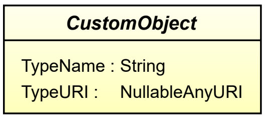
See the example for *CustomNumberPerTimeIntervalAttribute*.

10.13. CustomObject

10.13.1 Overview

Abstract class

The abstract base class of all custom classes.



Subtypes

- *CustomActuatingElectricalSystemComponent*
- *CustomActuatingSystemComponent*
- *CustomAgglomerator*
- *CustomBlower*
- *CustomCentrifuge*
- *CustomCheckValve*
- *CustomCompressor*
- *CustomCoolingTower*
- *CustomDryer*
- *CustomElectricGenerator*
- *CustomEquipment*
- *CustomExtruder*
- *CustomFan*
- *CustomFilter*
- *CustomHeatExchanger*
- *CustomHeater*
- *CustomInlinePrimaryElement*
- *CustomMill*
- *CustomMixer*
- *CustomMobileTransportSystem*
- *CustomMotor*
- *CustomOperatedValve*
- *CustomPipeFitting*
- *CustomPipingComponent*
- *CustomProcessSignalGeneratingSystemComponent*
- *CustomPump*
- *CustomSafetyValveOrFitting*
- *CustomSeparator*
- *CustomSieve*

- *CustomStationaryTransportSystem*
- *CustomTurbine*
- *CustomVessel*
- *CustomWasteGasEmitter*
- *CustomWeigher*

Attributes (data)

Name	Multiplicity	Type
TypeName	1	*String*
TypeURI	1	*NullableAnyURI*

Implementation in Proteus Schema

Implementation is subclass-specific.

10.13.2 TypeName

Attribute (data)

A name that identifies the type of the *CustomObject*.

Multiplicity: 1

Type: *String*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for string values*.
RDL reference: TYPE NAME ASSIGNMENT CLASS
Name: TypeNameAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/TypeNameAssignmentClass

Example

“micro impedance pump” (*String*)

Example: Implementation in Proteus Schema

```
<Equipment
    ID="customPump1"
    ComponentClass="CustomPump"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomPump" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="TypeNameAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/TypeNameAssignmentClass"
      Format="string"
      Value="micro impedance pump" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

10.13.3 TypeURI

Attribute (data)

A URI that identifies the type of the *CustomObject*.

Multiplicity: 1

Type: *NullableAnyURI*

Implementation in Proteus Schema

The attribute is implemented as a *DEXPI generic attribute for URI values*.
RDL reference: TYPE URI ASSIGNMENT CLASS
Name: TypeURIAssignmentClass
AttributeURI: http://sandbox.dexpi.org/rdl/TypeURIAssignmentClass

Example

`http://www.example.org/MicroImpedancePump` (*AnyURI*)

Example: Implementation in Proteus Schema

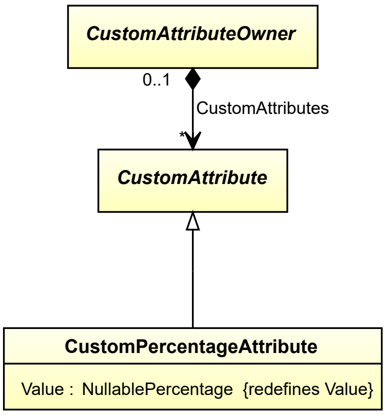
```
<Equipment
    ID="customPump1"
    ComponentClass="CustomPump"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomPump" ...>
  ...
  <GenericAttributes Set="DexpiAttributes" ...>
    <GenericAttribute
      Name="TypeURIAssignmentClass"
      AttributeURI="http://sandbox.dexpi.org/rdl/TypeURIAssignmentClass"
      Format="anyURI"
      Value="http://www.example.org/MicroImpedancePump" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

10.14. CustomPercentageAttribute

10.14.1 Overview

Class

A custom attribute with *Value* type *NullablePercentage*.


CustomAttributeOwner
0..1
CustomAttributes
*
CustomAttribute
CustomPercentageAttribute
Value : NullablePercentage {redefines Value}

Supertypes

- *CustomAttribute*

Attributes (data)

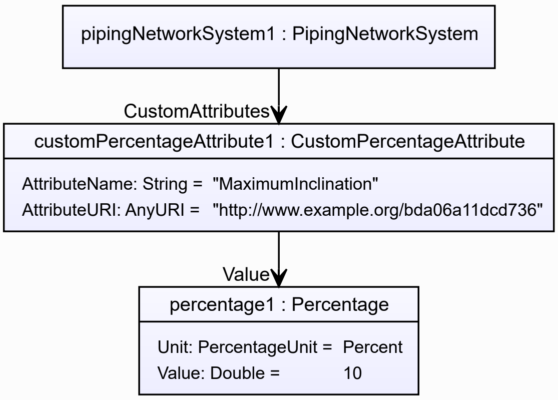
Name	Multiplicity	Type
Value	1	*NullablePercentage*

Implementation in Proteus Schema

CustomPercentageAttribute is implemented as a *custom generic attribute for physical quantities*.
RDL reference: PERCENTAGE
Type: Percentage
TypeURI: http://data.posccaesar.org/rdl/RDS13657820

Example

As *CustomAttributeOwner* is abstract, we consider *PipingNetworkSystem* as an arbitrary concrete subclass. The *PipingNetworkSystem* pipingNetworkSystem1 has a *CustomPercentageAttribute* with *AttributeName* “MaximumInclination” and an (arbitrary) *AttributeURI* “http://www.example.org/bda06a11dcd736”. The *Value* is 10 ???.


pipingNetworkSystem1 : PipingNetworkSystem
CustomAttributes
customPercentageAttribute1 : CustomPercentageAttribute
AttributeName: String = "MaximumInclination"
AttributeURI: AnyURI = "http://www.example.org/bda06a11dcd736"
Value
percentage1 : Percentage
Unit: PercentageUnit = Percent
Value: Double = 10

Example: Implementation in Proteus Schema

```
<PipingNetworkSystem
    ID="pipingNetworkSystem1"
    ComponentClass="PipingNetworkSystem"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS270359" ...>
  ...
  <GenericAttributes Set="DexpiCustomAttributes" ...>
    <GenericAttribute
      Name="MaximumInclination"
      AttributeURI="http://www.example.org/bda06a11dcd736"
      Format="double"
      Type="Percentage"
      TypeURI="http://data.posccaesar.org/rdl/RDS13657820"
      Value="10"
      Units="Percent"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1317959" />
    ...
  </GenericAttributes>
  ...
</PipingNetworkSystem>
```

10.14.2 Value

Attribute (data)

The value of the *CustomPercentageAttribute*.

Multiplicity: 1

Type: *NullablePercentage*

Redefines: *Value* (inherited from *CustomAttribute*)

Implementation in Proteus Schema

See implementation of *CustomPercentageAttribute*.

Example

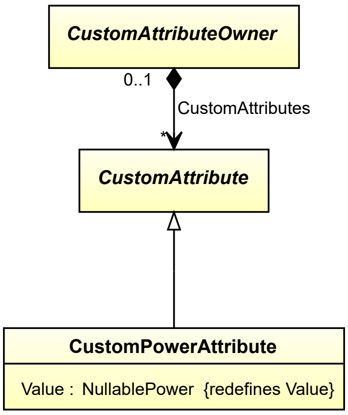
See the example for *CustomPercentageAttribute*.

10.15. CustomPowerAttribute

10.15.1 Overview

Class

A custom attribute with *Value* type *NullablePower*.



Supertypes

- *CustomAttribute*

Attributes (data)

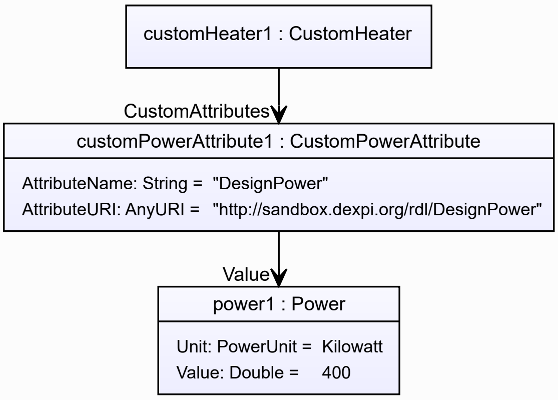
Name	Multiplicity	Type
Value	1	*NullablePower*

Implementation in Proteus Schema

CustomPowerAttribute is implemented as a *custom generic attribute for physical quantities*.
RDL reference: POWER
Type: Power
TypeURI: http://data.posccaesar.org/rdl/RDS354104

Example

As *CustomAttributeOwner* is abstract, we consider *CustomHeater* as an arbitrary concrete subclass.
The *CustomHeater* customHeater1 has a *CustomPowerAttribute* with *AttributeName* “DesignPower” and *AttributeURI* “http://sandbox.dexpi.org/rdl/DesignPower”. The *Value* is 400 kW.


customHeater1 : CustomHeater
CustomAttributes
customPowerAttribute1 : CustomPowerAttribute
AttributeName: String = "DesignPower"
AttributeURI: AnyURI = "http://sandbox.dexpi.org/rdl/DesignPower"
Value
power1 : Power
Unit: PowerUnit = Kilowatt
Value: Double = 400

Example: Implementation in Proteus Schema

```
<Equipment
    ID="customHeater1"
    ComponentClass="CustomHeater"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomHeater" ...>
  ...
  <GenericAttributes Set="DexpiCustomAttributes" ...>
    <GenericAttribute
      Name="DesignPower"
      AttributeURI="http://sandbox.dexpi.org/rdl/DesignPower"
      Format="double"
      Type="Power"
      TypeURI="http://data.posccaesar.org/rdl/RDS354104"
      Value="400"
      Units="Kilowatt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

10.15.2 Value

Attribute (data)

The value of the *CustomPowerAttribute*.

Multiplicity: 1

Type: *NullablePower*

Redefines: *Value* (inherited from *CustomAttribute*)

Implementation in Proteus Schema

See implementation of *CustomPowerAttribute*.

Example

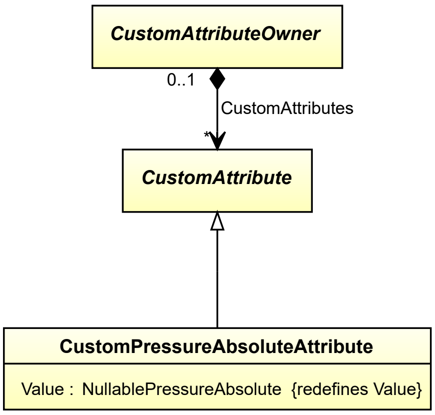
See the example for *CustomPowerAttribute*.

10.16. CustomPressureAbsoluteAttribute

10.16.1 Overview

Class

A custom attribute with *Value* type *NullablePressureAbsolute*.


CustomAttributeOwner
0..1
CustomAttributes
*
CustomAttribute
CustomPressureAbsoluteAttribute
Value : NullablePressureAbsolute {redefines Value}

Supertypes

- *CustomAttribute*

Attributes (data)

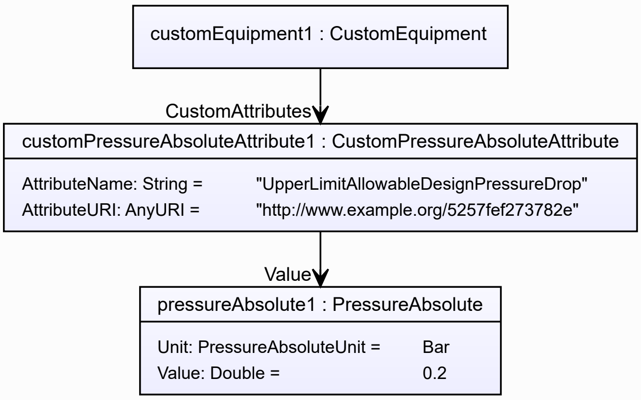
Name	Multiplicity	Type
Value	1	*NullablePressureAbsolute*

Implementation in Proteus Schema

CustomPressureAbsoluteAttribute is implemented as a *custom generic attribute for physical quantities*.
RDL reference: PRESSURE ABSOLUTE
Type: PressureAbsolute
TypeURI: http://sandbox.dexpi.org/rdl/PressureAbsolute

Example

As *CustomAttributeOwner* is abstract, we consider *CustomEquipment* as an arbitrary concrete subclass. The *CustomEquipment* customEquipment1 has a *CustomPressureAbsoluteAttribute* with *AttributeName* “UpperLimitAllowableDesignPressureDrop” and an (arbitrary) *AttributeURI* “http://www.example.org/5257fef273782e”. The *Value* is 0.2 bar.


customEquipment1 : CustomEquipment
CustomAttributes
customPressureAbsoluteAttribute1 : CustomPressureAbsoluteAttribute
AttributeName: String = "UpperLimitAllowableDesignPressureDrop"
AttributeURI: AnyURI = "http://www.example.org/5257fef273782e"
Value
pressureAbsolute1 : PressureAbsolute
Unit: PressureAbsoluteUnit = Bar
Value: Double = 0.2

Example: Implementation in Proteus Schema

```
<Equipment
    ID="customEquipment1"
    ComponentClass="CustomEquipment"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomEquipment" ...>
  ...
  <GenericAttributes Set="DexpiCustomAttributes" ...>
    <GenericAttribute
      Name="UpperLimitAllowableDesignPressureDrop"
      AttributeURI="http://www.example.org/5257fef273782e"
      Format="double"
      Type="PressureAbsolute"
      TypeURI="http://sandbox.dexpi.org/rdl/PressureAbsolute"
      Value="0.2"
      Units="Bar"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1314539" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

10.16.2 Value

Attribute (data)

The value of the *CustomPressureAbsoluteAttribute*.

Multiplicity: 1

Type: *NullablePressureAbsolute*

Redefines: *Value* (inherited from *CustomAttribute*)

Implementation in Proteus Schema

See implementation of *CustomPressureAbsoluteAttribute*.

Example

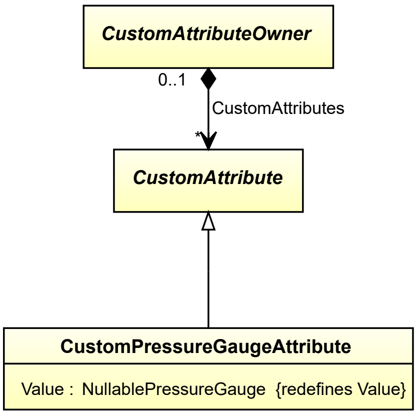
See the example for *CustomPressureAbsoluteAttribute*.

10.17. CustomPressureGaugeAttribute

10.17.1 Overview

Class

A custom attribute with *Value* type *NullablePressureGauge*.


CustomAttributeOwner
0..1
CustomAttributes
*
CustomAttribute
CustomPressureGaugeAttribute
Value : NullablePressureGauge {redefines Value}

Supertypes

- *CustomAttribute*

Attributes (data)

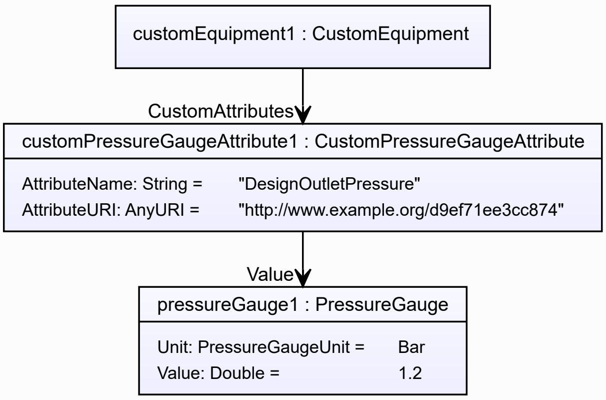
Name	Multiplicity	Type
Value	1	*NullablePressureGauge*

Implementation in Proteus Schema

CustomPressureGaugeAttribute is implemented as a *custom generic attribute for physical quantities*.
RDL reference: PRESSURE GAUGE
Type: PressureGauge
TypeURI: http://data.posccaesar.org/rdl/RDS416159

Example

As *CustomAttributeOwner* is abstract, we consider *CustomEquipment* as an arbitrary concrete subclass. The *CustomEquipment* customEquipment1 has a *CustomPressureGaugeAttribute* with *AttributeName* "DesignOutletPressure" and an (arbitrary) *AttributeURI* "http://www.example.org/d9ef71ee3cc874". The *Value* is 1.2 bar.


customEquipment1 : CustomEquipment
CustomAttributes
customPressureGaugeAttribute1 : CustomPressureGaugeAttribute
AttributeName: String = "DesignOutletPressure"
AttributeURI: AnyURI = "http://www.example.org/d9ef71ee3cc874"
Value
pressureGauge1 : PressureGauge
Unit: PressureGaugeUnit = Bar
Value: Double = 1.2

Example: Implementation in Proteus Schema

```
<Equipment
    ID="customEquipment1"
    ComponentClass="CustomEquipment"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomEquipment" ...>
  ...
  <GenericAttributes Set="DexpiCustomAttributes" ...>
    <GenericAttribute
      Name="DesignOutletPressure"
      AttributeURI="http://www.example.org/d9ef71ee3cc874"
      Format="double"
      Type="PressureGauge"
      TypeURI="http://data.posccaesar.org/rdl/RDS416159"
      Value="1.2"
      Units="Bar"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1314539" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

10.17.2 Value

Attribute (data)

The value of the *CustomPressureGaugeAttribute*.

Multiplicity: 1

Type: *NullablePressureGauge*

Redefines: *Value* (inherited from *CustomAttribute*)

Implementation in Proteus Schema

See implementation of *CustomPressureGaugeAttribute*.

Example

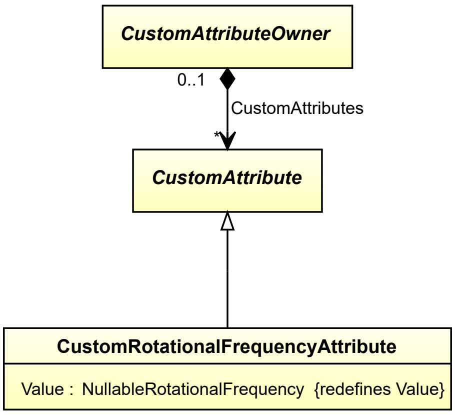
See the example for *CustomPressureGaugeAttribute*.

10.18. CustomRotationalFrequencyAttribute

10.18.1 Overview

Class

A custom attribute with *Value* type *NullableRotationalFrequency*.


CustomAttributeOwner
0..1
CustomAttributes
*
CustomAttribute
CustomRotationalFrequencyAttribute
Value : NullableRotationalFrequency {redefines Value}

Supertypes

- *CustomAttribute*

Attributes (data)

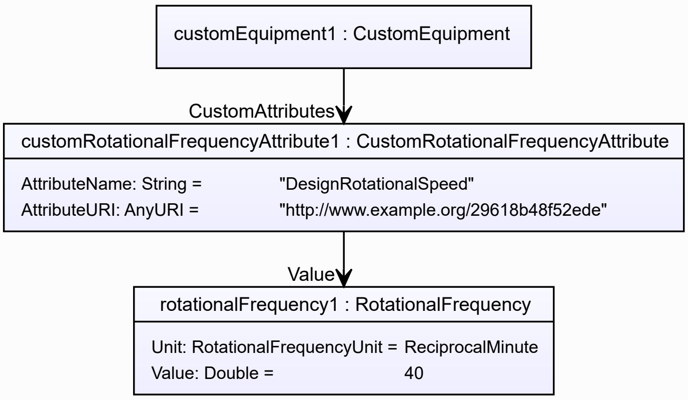
Name	Multiplicity	Type
Value	1	*NullableRotationalFrequency*

Implementation in Proteus Schema

CustomRotationalFrequencyAttribute is implemented as a *custom generic attribute for physical quantities*.
RDL reference: ROTATIONAL FREQUENCY
Type: RotationalFrequency
TypeURI: http://data.posccaesar.org/rdl/RDS354734

Example

As *CustomAttributeOwner* is abstract, we consider *CustomEquipment* as an arbitrary concrete subclass. The *CustomEquipment* customEquipment1 has a *CustomRotationalFrequencyAttribute* with *AttributeName* “DesignRotationalSpeed” and an (arbitrary) *AttributeURI* “http://www.example.org/29618b48f52ede”. The *Value* is 40 min^{-1}.


customEquipment1 : CustomEquipment
CustomAttributes
customRotationalFrequencyAttribute1 : CustomRotationalFrequencyAttribute
AttributeName: String = "DesignRotationalSpeed"
AttributeURI: AnyURI = "http://www.example.org/29618b48f52ede"
Value
rotationalFrequency1 : RotationalFrequency
Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 40

Example: Implementation in Proteus Schema

```
<Equipment
    ID="customEquipment1"
    ComponentClass="CustomEquipment"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomEquipment" ...>
  ...
  <GenericAttributes Set="DexpiCustomAttributes" ...>
    <GenericAttribute
      Name="DesignRotationalSpeed"
      AttributeURI="http://www.example.org/29618b48f52ede"
      Format="double"
      Type="RotationalFrequency"
      TypeURI="http://data.posccaesar.org/rdl/RDS354734"
      Value="40"
      Units="ReciprocalMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

10.18.2 Value

Attribute (data)

The value of the *CustomRotationalFrequencyAttribute*.

Multiplicity: 1

Type: *NullableRotationalFrequency*

Redefines: *Value* (inherited from *CustomAttribute*)

Implementation in Proteus Schema

See implementation of *CustomRotationalFrequencyAttribute*.

Example

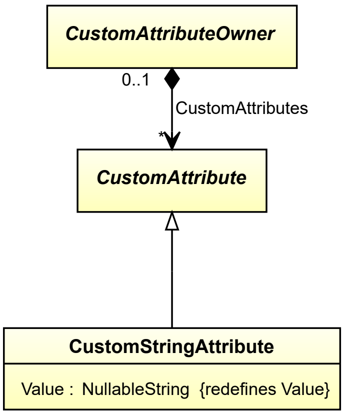
See the example for *CustomRotationalFrequencyAttribute*.

10.19. CustomStringAttribute

10.19.1 Overview

Class

A custom attribute with *Value* type *NullableString*.



Supertypes

- *CustomAttribute*

Attributes (data)

Name	Multiplicity	Type
Value	1	*NullableString*

Implementation in Proteus Schema

CustomStringAttribute is implemented as a *custom generic attribute for string values*.
RDL reference: STRING
Type: String
TypeURI: http://sandbox.dexpi.org/rdl/String

10.19.2 Value

Attribute (data)

The value of the *CustomStringAttribute*.

Multiplicity: 1

Type: *NullableString*

Redefines: *Value* (inherited from *CustomAttribute*)

Implementation in Proteus Schema

See implementation of *CustomStringAttribute*.

Example

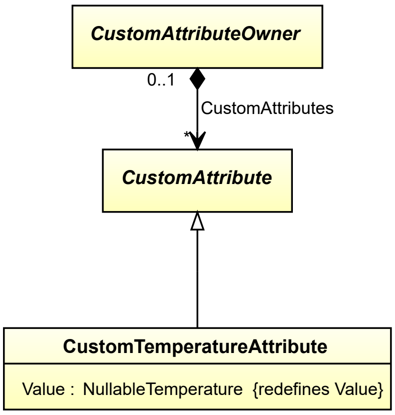
See the example for *CustomStringAttribute*.

10.20. CustomTemperatureAttribute

10.20.1 Overview

Class

A custom attribute with *Value* type *NullableTemperature*.


CustomAttributeOwner
0..1
CustomAttributes
*
CustomAttribute
CustomTemperatureAttribute
Value : NullableTemperature {redefines Value}

Supertypes

- *CustomAttribute*

Attributes (data)

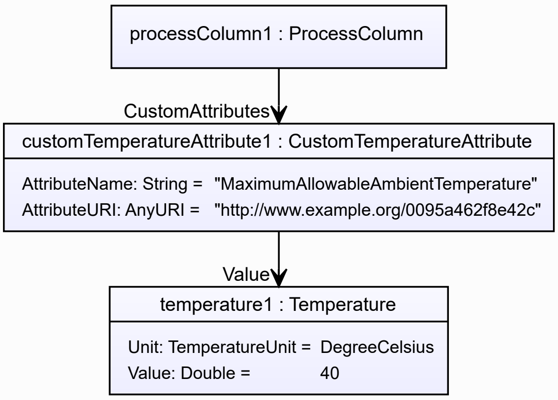
Name	Multiplicity	Type
Value	1	*NullableTemperature*

Implementation in Proteus Schema

CustomTemperatureAttribute is implemented as a *custom generic attribute for physical quantities*.
RDL reference: TEMPERATURE
Type: Temperature
TypeURI: http://data.posccaesar.org/rdl/RDS355859

Example

As *CustomAttributeOwner* is abstract, we consider *ProcessColumn* as an arbitrary concrete subclass. The *ProcessColumn* processColumn1 has a *CustomTemperatureAttribute* with *AttributeName* “MaximumAllowableAmbientTemperature” and an (arbitrary) *AttributeURI* “http://www.example.org/0095a462f8e42c”. The *Value* is 40 °C.


processColumn1 : ProcessColumn
CustomAttributes
customTemperatureAttribute1 : CustomTemperatureAttribute
AttributeName: String = "MaximumAllowableAmbientTemperature"
AttributeURI: AnyURI = "http://www.example.org/0095a462f8e42c"
Value
temperature1 : Temperature
Unit: TemperatureUnit = DegreeCelsius
Value: Double = 40

Example: Implementation in Proteus Schema

```
<Equipment
    ID="processColumn1"
    ComponentClass="ProcessColumn"
    ComponentClassURI="http://data.posccaesar.org/rdl/RDS4316825224" ...>
  ...
  <GenericAttributes Set="DexpiCustomAttributes" ...>
    <GenericAttribute
      Name="MaximumAllowableAmbientTemperature"
      AttributeURI="http://www.example.org/0095a462f8e42c"
      Format="double"
      Type="Temperature"
      TypeURI="http://data.posccaesar.org/rdl/RDS355859"
      Value="40"
      Units="DegreeCelsius"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1322684" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

10.20.2 Value

Attribute (data)

The value of the *CustomTemperatureAttribute*.

Multiplicity: 1

Type: *NullableTemperature*

Redefines: *Value* (inherited from *CustomAttribute*)

Implementation in Proteus Schema

See implementation of *CustomTemperatureAttribute*.

Example

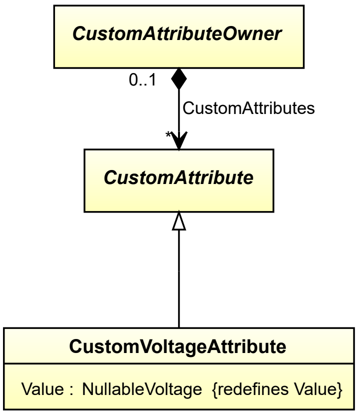
See the example for *CustomTemperatureAttribute*.

10.21. CustomVoltageAttribute

10.21.1 Overview

Class

A custom attribute with *Value* type *NullableVoltage*.



Supertypes

- *CustomAttribute*

Attributes (data)

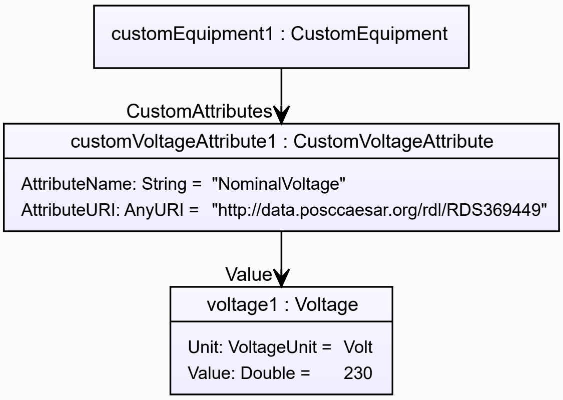
Name	Multiplicity	Type
Value	1	*NullableVoltage*

Implementation in Proteus Schema

CustomVoltageAttribute is implemented as a *custom generic attribute for physical quantities*.
RDL reference: VOLTAGE
Type: Voltage
TypeURI: http://data.posccaesar.org/rdl/RDS372374

Example

As *CustomAttributeOwner* is abstract, we consider *CustomEquipment* as an arbitrary concrete subclass. The *CustomEquipment* customEquipment1 has a *CustomVoltageAttribute* with *AttributeName* “NominalVoltage” and *AttributeURI* “http://data.posccaesar.org/rdl/RDS369449”. The *Value* is 230 V.


customEquipment1 : CustomEquipment
CustomAttributes
customVoltageAttribute1 : CustomVoltageAttribute
AttributeName: String = "NominalVoltage"
AttributeURI: AnyURI = "http://data.posccaesar.org/rdl/RDS369449"
Value
voltage1 : Voltage
Unit: VoltageUnit = Volt
Value: Double = 230

Example: Implementation in Proteus Schema

```
<Equipment
    ID="customEquipment1"
    ComponentClass="CustomEquipment"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomEquipment" ...>
  ...
  <GenericAttributes Set="DexpiCustomAttributes" ...>
    <GenericAttribute
      Name="NominalVoltage"
      AttributeURI="http://data.posccaesar.org/rdl/RDS369449"
      Format="double"
      Type="Voltage"
      TypeURI="http://data.posccaesar.org/rdl/RDS372374"
      Value="230"
      Units="Volt"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1347974" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

10.21.2 Value

Attribute (data)

The value of the *CustomVoltageAttribute*.

Multiplicity: 1

Type: *NullableVoltage*

Redefines: *Value* (inherited from *CustomAttribute*)

Implementation in Proteus Schema

See implementation of *CustomVoltageAttribute*.

Example

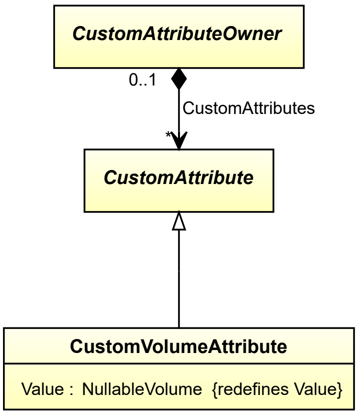
See the example for *CustomVoltageAttribute*.

10.22. CustomVolumeAttribute

10.22.1 Overview

Class

A custom attribute with *Value* type *NullableVolume*.



Supertypes

- *CustomAttribute*

Attributes (data)

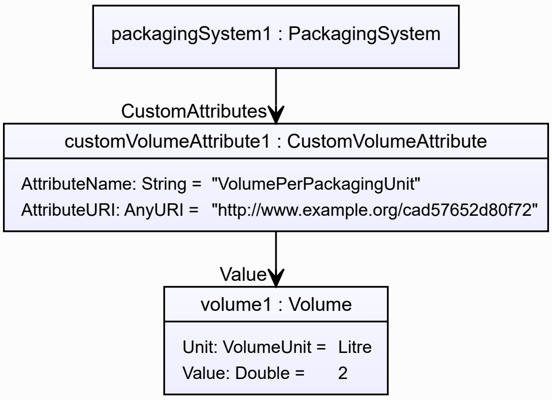
Name	Multiplicity	Type
Value	1	*NullableVolume*

Implementation in Proteus Schema

CustomVolumeAttribute is implemented as a *custom generic attribute for physical quantities*.
RDL reference: VOLUME
Type: Volume
TypeURI: http://data.posccaesar.org/rdl/RDS356444

Example

As *CustomAttributeOwner* is abstract, we consider *PackagingSystem* as an arbitrary concrete subclass. The *PackagingSystem* packagingSystem1 has a *CustomVolumeAttribute* with *AttributeName* “VolumePerPackagingUnit” and an (arbitrary) *AttributeURI* “http://www.example.org/cad57652d80f72”. The *Value* is 2 l.


packagingSystem1 : PackagingSystem
CustomAttributes
customVolumeAttribute1 : CustomVolumeAttribute
AttributeName: String = "VolumePerPackagingUnit"
AttributeURI: AnyURI = "http://www.example.org/cad57652d80f72"
Value
volume1 : Volume
Unit: VolumeUnit = Litre
Value: Double = 2

Example: Implementation in Proteus Schema

```
<Equipment
    ID="packagingSystem1"
    ComponentClass="PackagingSystem"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/PackagingSystem" ...>
  ...
  <GenericAttributes Set="DexpiCustomAttributes" ...>
    <GenericAttribute
      Name="VolumePerPackagingUnit"
      AttributeURI="http://www.example.org/cad57652d80f72"
      Format="double"
      Type="Volume"
      TypeURI="http://data.posccaesar.org/rdl/RDS356444"
      Value="2"
      Units="Litre"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1331144" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

10.22.2 Value

Attribute (data)

The value of the *CustomVolumeAttribute*.

Multiplicity: 1

Type: *NullableVolume*

Redefines: *Value* (inherited from *CustomAttribute*)

Implementation in Proteus Schema

See implementation of *CustomVolumeAttribute*.

Example

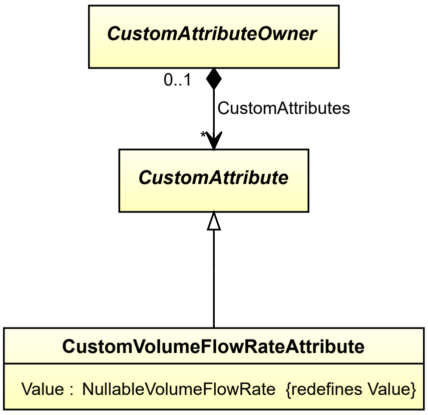
See the example for *CustomVolumeAttribute*.

10.23. CustomVolumeFlowRateAttribute

10.23.1 Overview

Class

A custom attribute with *Value* type *NullableVolumeFlowRate*.


CustomAttributeOwner
0..1
CustomAttributes
*
CustomAttribute
CustomVolumeFlowRateAttribute
Value : NullableVolumeFlowRate {redefines Value}

Supertypes

- *CustomAttribute*

Attributes (data)

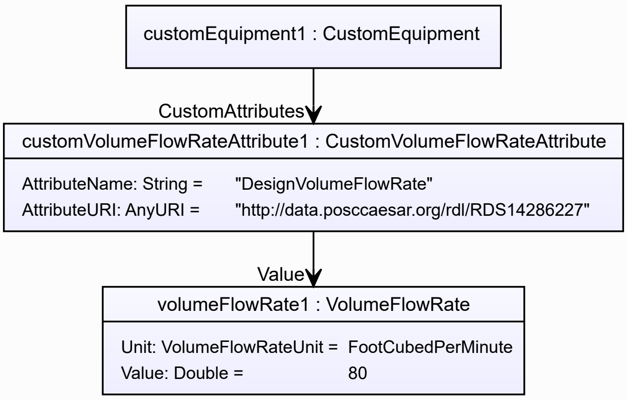
Name	Multiplicity	Type
Value	1	*NullableVolumeFlowRate*

Implementation in Proteus Schema

CustomVolumeFlowRateAttribute is implemented as a *custom generic attribute for physical quantities*.
RDL reference: VOLUME FLOW RATE
Type: VolumeFlowRate
TypeURI: http://data.posccaesar.org/rdl/RDS380834

Example

As *CustomAttributeOwner* is abstract, we consider *CustomEquipment* as an arbitrary concrete subclass. The *CustomEquipment* customEquipment1 has a *CustomVolumeFlowRateAttribute* with *AttributeName* “DesignVolumeFlowRate” and *AttributeURI* “http://data.posccaesar.org/rdl/RDS14286227”. The *Value* is 80 ft^3/min.


customEquipment1 : CustomEquipment
CustomAttributes
customVolumeFlowRateAttribute1 : CustomVolumeFlowRateAttribute
AttributeName: String = "DesignVolumeFlowRate"
AttributeURI: AnyURI = "http://data.posccaesar.org/rdl/RDS14286227"
Value
volumeFlowRate1 : VolumeFlowRate
Unit: VolumeFlowRateUnit = FootCubedPerMinute
Value: Double = 80

Example: Implementation in Proteus Schema

```
<Equipment
    ID="customEquipment1"
    ComponentClass="CustomEquipment"
    ComponentClassURI="http://sandbox.dexpi.org/rdl/CustomEquipment" ...>
  ...
  <GenericAttributes Set="DexpiCustomAttributes" ...>
    <GenericAttribute
      Name="DesignVolumeFlowRate"
      AttributeURI="http://data.posccaesar.org/rdl/RDS14286227"
      Format="double"
      Type="VolumeFlowRate"
      TypeURI="http://data.posccaesar.org/rdl/RDS380834"
      Value="80"
      Units="FootCubedPerMinute"
      UnitsURI="http://data.posccaesar.org/rdl/RDS1320164" />
    ...
  </GenericAttributes>
  ...
</Equipment>
```

10.23.2 Value

Attribute (data)

The value of the *CustomVolumeFlowRateAttribute*.

Multiplicity: 1

Type: *NullableVolumeFlowRate*

Redefines: *Value* (inherited from *CustomAttribute*)

Implementation in Proteus Schema

See implementation of *CustomVolumeFlowRateAttribute*.

Example

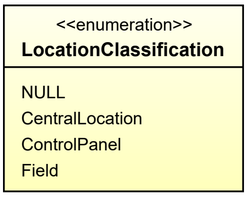
See the example for *CustomVolumeFlowRateAttribute*.

Package Enumerations 11

11.1. Overview

The *Enumerations* package contains enumerations for various aspects of engineering information in a P&ID. Enumerations that are relevant for P&ID graphics only are part of the *Graphics* package.

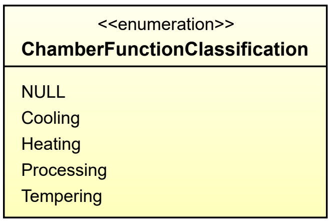
For example, the *LocationClassification* enumeration provides four enumeration literals: *NULL* (the *null value* for this enumeration), *CentralLocation*, *ControlPanel*, and *Field*. *LocationClassification* is used as the type of the *Location* attribute of *ProcessInstrumentationFunction*.



11.2. ChamberFunctionClassification

11.2.1 Overview

Enumeration



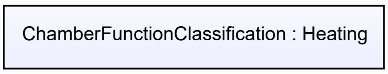
Literals

Name	Symbol	RDL Reference
NULL	*null*	
Cooling	cooling	COOLING http://data.posccaesar.org/rdl/RDS9684422
Heating	heating	HEATING http://data.posccaesar.org/rdl/RDS9666872
Processing	processing	PROCESSING http://data.posccaesar.org/rdl/RDS9658367
Tempering	tempering	TEMPERING http://sandbox.dexpi.org/rdl/Tempering

Implementation in Proteus Schema

All data attributes with type *ChamberFunctionClassification* are implemented as *DEXPI generic attributes for enumeration values*. In a <GenericAttribute> element, the *ChamberFunctionClassification* literal is given by means of its RDL reference in the table above. The Value attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The ValueURI attribute of the element is the URI of the RDL reference.

Example



Example: Implementation in Proteus Schema

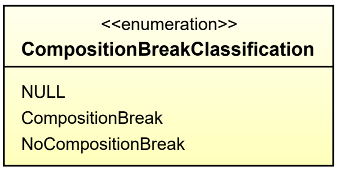
```
<GenericAttribute
  Format="anyURI"
  Value="Heating"
  ValueURI="http://data.posccaesar.org/rdl/RDS9666872" ...>
```

Note that the <GenericAttribute> element must have a Name and an AttributeURI attribute. They depend on the data type attribute of the DEXPI class that owns the instance *Heating*. For a complete example, see the attribute *ChamberFunction* of the DEXPI class *Chamber*.

11.3. CompositionBreakClassification

11.3.1 Overview

Enumeration



Literals

Name	Symbol	RDL Reference
NULL	*null*	
CompositionBreak	composition break	COMPOSITION BREAK http://sandbox.dexpi.org/rdl/CompositionBreak
NoCompositionBreak	no composition break	NO COMPOSITION BREAK http://sandbox.dexpi.org/rdl/NoCompositionBreak

Implementation in Proteus Schema

All data attributes with type *CompositionBreakClassification* are implemented as *DEXPI generic attributes for enumeration values*. In a <GenericAttribute> element, the *CompositionBreakClassification* literal is given by means of its RDL reference in the table above. The Value attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The ValueURI attribute of the element is the URI of the RDL reference.

Example

Example: Implementation in Proteus Schema

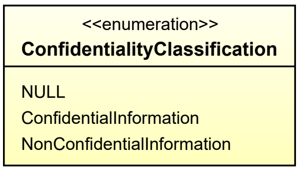
```
<GenericAttribute
  Format="anyURI"
  Value="NoCompositionBreak"
  ValueURI="http://sandbox.dexpi.org/rdl/NoCompositionBreak" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance *NoCompositionBreak*. For a complete example, see the attribute *CompositionBreak* of the DEXPI class *PropertyBreak*.

11.4. ConfidentialityClassification

11.4.1 Overview

Enumeration



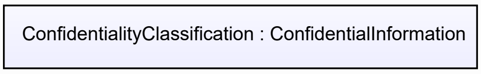
Literals

Name	Symbol	RDL Reference
NULL	*null*	
ConfidentialInformation	confidential	CONFIDENTIAL INFORMATION http://data.posccaesar.org/rdl/RDS4316590816
NonConfidentialInformation	not confidential	NON CONFIDENTIAL INFORMATION http://sandbox.dexpi.org/rdl/NonConfidentialInformation

Implementation in Proteus Schema

All data attributes with type *ConfidentialityClassification* are implemented as *DEXPI generic attributes for enumeration values*. In a `<GenericAttribute>` element, the *ConfidentialityClassification* literal is given by means of its RDL reference in the table above. The `Value` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `ValueURI` attribute of the element is the URI of the RDL reference.

Example



Example: Implementation in Proteus Schema

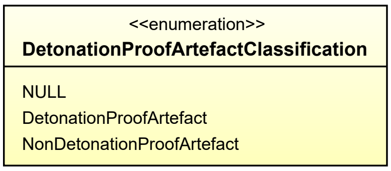
```
<GenericAttribute
  Format="anyURI"
  Value="ConfidentialInformation"
  ValueURI="http://data.posccaesar.org/rdl/RDS4316590816" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance *ConfidentialInformation*. For a complete example, see the attribute *Confidentiality* of the DEXPI class *MetaData*.

11.5. DetonationProofArtefactClassification

11.5.1 Overview

Enumeration


<<enumeration>>
DetonationProofArtefactClassification
NULL
DetonationProofArtefact
NonDetonationProofArtefact

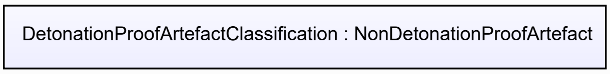
Literals

Name	Symbol	RDL Reference
NULL	*null*	
DetonationProofArtefact	detonation-proof artefact	DETONATION PROOF ARTEFACT http://sandbox.dexpi.org/rdl/DetonationProofArtefact
NonDetonationProofArtefact	non detonation-proof artefact	NON DETONATION PROOF ARTEFACT http://sandbox.dexpi.org/rdl/NonDetonationProofArtefact

Implementation in Proteus Schema

All data attributes with type *DetonationProofArtefactClassification* are implemented as *DEXPI generic attributes for enumeration values*. In a `<GenericAttribute>` element, the *DetonationProofArtefactClassification* literal is given by means of its RDL reference in the table above. The `Value` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `ValueURI` attribute of the element is the URI of the RDL reference.

Example


DetonationProofArtefactClassification : NonDetonationProofArtefact

Example: Implementation in Proteus Schema

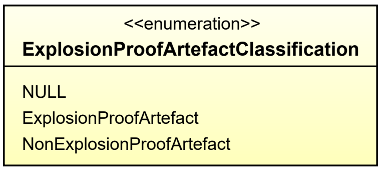
```
<GenericAttribute
  Format="anyURI"
  Value="NonDetonationProofArtefact"
  ValueURI="http://sandbox.dexpi.org/rdl/NonDetonationProofArtefact" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance *NonDetonationProofArtefact*. For a complete example, see the attribute *DetonationProofArtefact* of the DEXPI class *FlameArrestor*.

11.6. ExplosionProofArtefactClassification

11.6.1 Overview

Enumeration


<<enumeration>>
ExplosionProofArtefactClassification
NULL
ExplosionProofArtefact
NonExplosionProofArtefact

Literals

Name	Symbol	RDL Reference
NULL	*null*	
ExplosionProofArtefact	explosion-proof artefact	EXPLOSION PROOF ARTEFACT http://sandbox.dexpi.org/rdl/ExplosionProofArtefact
NonExplosionProofArtefact	non explosion-proof artefact	NON EXPLOSION PROOF ARTEFACT http://sandbox.dexpi.org/rdl/NonExplosionProofArtefact

Implementation in Proteus Schema

All data attributes with type *ExplosionProofArtefactClassification* are implemented as *DEXPI generic attributes for enumeration values*. In a `<GenericAttribute>` element, the *ExplosionProofArtefactClassification* literal is given by means of its RDL reference in the table above. The `Value` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `ValueURI` attribute of the element is the URI of the RDL reference.

Example

ExplosionProofArtefactClassification : ExplosionProofArtefact

Example: Implementation in Proteus Schema

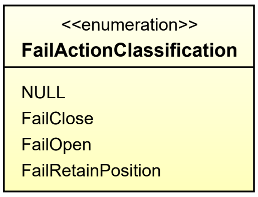
```
<GenericAttribute
  Format="anyURI"
  Value="ExplosionProofArtefact"
  ValueURI="http://sandbox.dexpi.org/rdl/ExplosionProofArtefact" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance *ExplosionProofArtefact*. For a complete example, see the attribute *ExplosionProofArtefact* of the DEXPI class *FlameArrestor*.

11.7. FailActionClassification

11.7.1 Overview

Enumeration



Literals

Name	Symbol	RDL Reference
NULL	*null*	
FailClose	fail close	FAIL CLOSE http://data.posccaesar.org/rdl/RDS5921400
FailOpen	fail open	FAIL OPEN http://data.posccaesar.org/rdl/RDS5921445
FailRetainPosition	fail retain position	FAIL RETAIN POSITION http://sandbox.dexpi.org/rdl/FailRetainPosition

Implementation in Proteus Schema

All data attributes with type *FailActionClassification* are implemented as *DEXPI generic attributes for enumeration values*. In a `<GenericAttribute>` element, the *FailActionClassification* literal is given by means of its RDL reference in the table above. The `Value` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `ValueURI` attribute of the element is the URI of the RDL reference.

Example

FailActionClassification : FailOpen

Example: Implementation in Proteus Schema

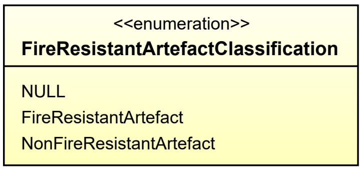
```
<GenericAttribute
  Format="anyURI"
  Value="FailOpen"
  ValueURI="http://data.posccaesar.org/rdl/RDS5921445" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance *FailOpen*. For a complete example, see the attribute *FailAction* of the DEXPI class *ControlledActuator*.

11.8. FireResistantArtefactClassification

11.8.1 Overview

Enumeration



Literals

Name	Symbol	RDL Reference
NULL	*null*	
FireResistantArtefact	fire-resistant artefact	FIRE RESISTANT ARTEFACT http://data.posccaesar.org/rdl/RDS7907520
NonFireResistantArtefact	non fire-resistant artefact	NON FIRE RESISTANT ARTEFACT http://sandbox.dexpi.org/rdl/NonFireResistantArtefact

Implementation in Proteus Schema

All data attributes with type *FireResistantArtefactClassification* are implemented as *DEXPI generic attributes for enumeration values*. In a `<GenericAttribute>` element, the *FireResistantArtefactClassification* literal is given by means of its RDL reference in the table above. The `Value` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `ValueURI` attribute of the element is the URI of the RDL reference.

Example

FireResistantArtefactClassification : FireResistantArtefact

Example: Implementation in Proteus Schema

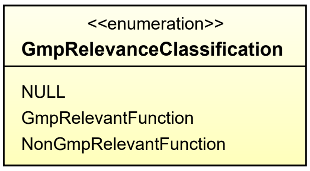
```
<GenericAttribute
  Format="anyURI"
  Value="FireResistantArtefact"
  ValueURI="http://data.posccaesar.org/rdl/RDS7907520" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance *FireResistantArtefact*. For a complete example, see the attribute *FireResistantArtefact* of the DEXPI class *FlameArrestor*.

11.9. GmpRelevanceClassification

11.9.1 Overview

Enumeration



Literals

Name	Symbol	RDL Reference
NULL	*null*	
GmpRelevantFunction	GMP relevant	GMP RELEVANT FUNCTION http://sandbox.dexpi.org/rdl/GmpRelevantFunction
NonGmpRelevantFunction	not GMP relevant	NON GMP RELEVANT FUNCTION http://sandbox.dexpi.org/rdl/NonGmpRelevantFunction

Implementation in Proteus Schema

All data attributes with type *GmpRelevanceClassification* are implemented as *DEXPI generic attributes for enumeration values*. In a `<GenericAttribute>` element, the *GmpRelevanceClassification* literal is given by means of its RDL reference in the table above. The `Value` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `ValueURI` attribute of the element is the URI of the RDL reference.

Example

GmpRelevanceClassification : GmpRelevantFunction

Example: Implementation in Proteus Schema

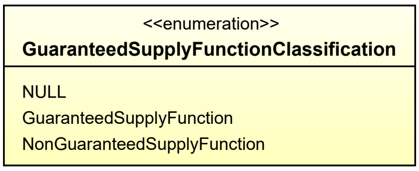
```
<GenericAttribute
  Format="anyURI"
  Value="GmpRelevantFunction"
  ValueURI="http://sandbox.dexpi.org/rdl/GmpRelevantFunction" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance *GmpRelevantFunction*. For a complete example, see the attribute *GmpRelevance* of the DEXPI class *ProcessInstrumentationFunction*.

11.10. GuaranteedSupplyFunctionClassification

11.10.1 Overview

Enumeration



<<enumeration>>
GuaranteedSupplyFunctionClassification
NULL
GuaranteedSupplyFunction
NonGuaranteedSupplyFunction

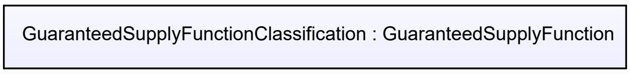
Literals

Name	Symbol	RDL Reference
NULL	*null*	
GuaranteedSupplyFunction	guaranteed supply	GUARANTEED SUPPLY FUNCTION http://sandbox.dexpi.org/rdl/GuaranteedSupplyFunction
NonGuaranteedSupplyFunction	no guaranteed supply	NON GUARANTEED SUPPLY FUNCTION http://sandbox.dexpi.org/rdl/NonGuaranteedSupplyFunction

Implementation in Proteus Schema

All data attributes with type *GuaranteedSupplyFunctionClassification* are implemented as *DEXPI generic attributes for enumeration values*. In a `<GenericAttribute>` element, the *GuaranteedSupplyFunctionClassification* literal is given by means of its RDL reference in the table above. The `Value` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `ValueURI` attribute of the element is the URI of the RDL reference.

Example



GuaranteedSupplyFunctionClassification : GuaranteedSupplyFunction

Example: Implementation in Proteus Schema

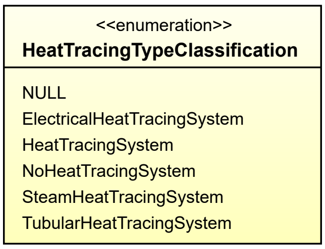
```
<GenericAttribute
  Format="anyURI"
  Value="GuaranteedSupplyFunction"
  ValueURI="http://sandbox.dexpi.org/rdl/GuaranteedSupplyFunction" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance *GuaranteedSupplyFunction*. For a complete example, see the attribute *GuaranteedSupplyFunction* of the DEXPI class *ProcessInstrumentationFunction*.

11.11. HeatTracingTypeClassification

11.11.1 Overview

Enumeration



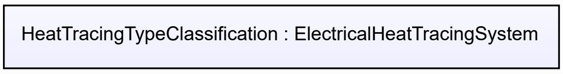
Literals

Name	Symbol	RDL Reference
NULL	*null*	
ElectricalHeatTracingSystem	electrical heat tracing system	ELECTRICAL HEAT TRACING SYSTEM http://data.posccaesar.org/rdl/RDS11854600
HeatTracingSystem	heat tracing system	HEAT TRACING SYSTEM http://data.posccaesar.org/rdl/RDS267434
NoHeatTracingSystem	no heat tracing system	NO HEAT TRACING SYSTEM http://sandbox.dexpi.org/rdl/NoHeatTracingSystem
SteamHeatTracingSystem	steam heat tracing system	STEAM HEAT TRACING SYSTEM http://data.posccaesar.org/rdl/RDS11854690
TubularHeatTracingSystem	tubular heat tracing system	TUBULAR HEAT TRACING SYSTEM http://data.posccaesar.org/rdl/RDS11854645

Implementation in Proteus Schema

All data attributes with type *HeatTracingTypeClassification* are implemented as *DEXPI generic attributes for enumeration values*. In a `<GenericAttribute>` element, the *HeatTracingTypeClassification* literal is given by means of its RDL reference in the table above. The `Value` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `ValueURI` attribute of the element is the URI of the RDL reference.

Example



Example: Implementation in Proteus Schema

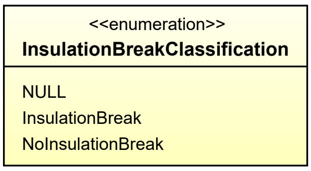
```
<GenericAttribute
  Format="anyURI"
  Value="ElectricalHeatTracingSystem"
  ValueURI="http://data.posccaesar.org/rdl/RDS11854600" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance *ElectricalHeatTracingSystem*. For a complete example, see the attribute *HeatTracingType* of the DEXPI class *OfflinePrimaryElement*.

11.12. InsulationBreakClassification

11.12.1 Overview

Enumeration



Literals

Name	Symbol	RDL Reference
NULL	*null*	
InsulationBreak	inulation break	INSULATION BREAK http://sandbox.dexpi.org/rdl/InsulationBreak
NoInsulationBreak	no insulation break	NO INSULATION BREAK http://sandbox.dexpi.org/rdl/NoInsulationBreak

Implementation in Proteus Schema

All data attributes with type *InsulationBreakClassification* are implemented as *DEXPI generic attributes for enumeration values*. In a `<GenericAttribute>` element, the *InsulationBreakClassification* literal is given by means of its RDL reference in the table above. The `Value` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `ValueURI` attribute of the element is the URI of the RDL reference.

Example

InsulationBreakClassification : InsulationBreak

Example: Implementation in Proteus Schema

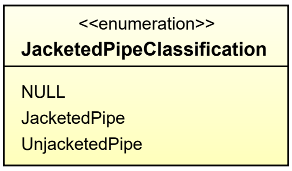
```
<GenericAttribute
  Format="anyURI"
  Value="InsulationBreak"
  ValueURI="http://sandbox.dexpi.org/rdl/InsulationBreak" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance *InsulationBreak*. For a complete example, see the attribute *InsulationBreak* of the DEXPI class *PropertyBreak*.

11.13. JacketedPipeClassification

11.13.1 Overview

Enumeration



Literals

Name	Symbol	RDL Reference
NULL	*null*	
JacketedPipe	jacketed	JACKETED PIPE http://sandbox.dexpi.org/rdl/JacketedPipe
UnjacketedPipe	not jacketed	UNJACKETED PIPE http://sandbox.dexpi.org/rdl/UnjacketedPipe

Implementation in Proteus Schema

All data attributes with type *JacketedPipeClassification* are implemented as *DEXPI generic attributes for enumeration values*. In a `<GenericAttribute>` element, the *JacketedPipeClassification* literal is given by means of its RDL reference in the table above. The `Value` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `ValueURI` attribute of the element is the URI of the RDL reference.

Example

JacketedPipeClassification : JacketedPipe

Example: Implementation in Proteus Schema

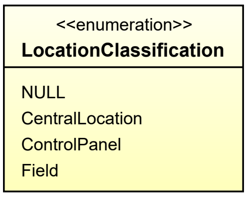
```
<GenericAttribute
  Format="anyURI"
  Value="JacketedPipe"
  ValueURI="http://sandbox.dexpi.org/rdl/JacketedPipe" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance *JacketedPipe*. For a complete example, see the attribute *JacketedPipe* of the DEXPI class *PipingNetworkSegment*.

11.14. LocationClassification

11.14.1 Overview

Enumeration



Literals

Name	Symbol	RDL Reference
NULL	*null*	
CentralLocation	central	CENTRAL LOCATION http://sandbox.dexpi.org/rdl/CentralLocation
ControlPanel	panel	CONTROL PANEL http://data.posccaesar.org/rdl/RDS874124
Field	field	FIELD http://data.posccaesar.org/rdl/RDS409545541

Implementation in Proteus Schema

All data attributes with type *LocationClassification* are implemented as *DEXPI generic attributes for enumeration values*. In a `<GenericAttribute>` element, the *LocationClassification* literal is given by means of its RDL reference in the table above. The `Value` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `ValueURI` attribute of the element is the URI of the RDL reference.

Example

LocationClassification : Field

Example: Implementation in Proteus Schema

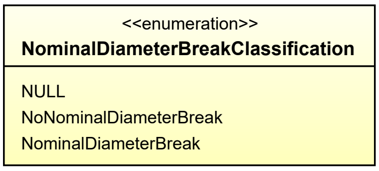
```
<GenericAttribute
  Format="anyURI"
  Value="Field"
  ValueURI="http://data.posccaesar.org/rdl/RDS409545541" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance *Field*. For a complete example, see the attribute *Location* of the DEXPI class *ProcessInstrumentationFunction*.

11.15. NominalDiameterBreakClassification

11.15.1 Overview

Enumeration



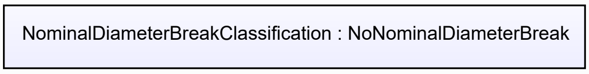
Literals

Name	Symbol	RDL Reference
NULL	*null*	
NoNominalDiameterBreak	no nominal diameter break	NO NOMINAL DIAMETER BREAK http://sandbox.dexpi.org/rdl/NoNominalDiameterBreak
NominalDiameterBreak	nominal diameter break	NOMINAL DIAMETER BREAK http://sandbox.dexpi.org/rdl/NominalDiameterBreak

Implementation in Proteus Schema

All data attributes with type *NominalDiameterBreakClassification* are implemented as *DEXPI generic attributes for enumeration values*. In a `<GenericAttribute>` element, the *NominalDiameterBreakClassification* literal is given by means of its RDL reference in the table above. The `Value` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `ValueURI` attribute of the element is the URI of the RDL reference.

Example



Example: Implementation in Proteus Schema

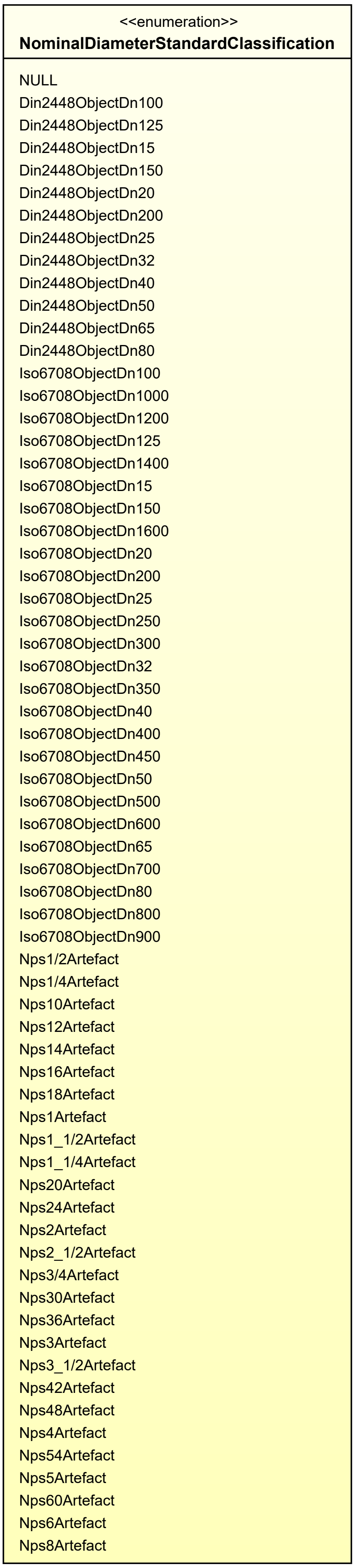
```
<GenericAttribute
  Format="anyURI"
  Value="NoNominalDiameterBreak"
  ValueURI="http://sandbox.dexpi.org/rdl/NoNominalDiameterBreak" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance *NoNominalDiameterBreak*. For a complete example, see the attribute *NominalDiameterBreak* of the DEXPI class *PropertyBreak*.

11.16. NominalDiameterStandardClassification

11.16.1 Overview

Enumeration



<<enumeration>>
NominalDiameterStandardClassification

NULL
Din2448ObjectDn100
Din2448ObjectDn125
Din2448ObjectDn15
Din2448ObjectDn150
Din2448ObjectDn20
Din2448ObjectDn200
Din2448ObjectDn25
Din2448ObjectDn32
Din2448ObjectDn40
Din2448ObjectDn50
Din2448ObjectDn65
Din2448ObjectDn80
Iso6708ObjectDn100
Iso6708ObjectDn1000
Iso6708ObjectDn1200
Iso6708ObjectDn125
Iso6708ObjectDn1400
Iso6708ObjectDn15
Iso6708ObjectDn150
Iso6708ObjectDn1600
Iso6708ObjectDn20
Iso6708ObjectDn200
Iso6708ObjectDn25
Iso6708ObjectDn250
Iso6708ObjectDn300
Iso6708ObjectDn32
Iso6708ObjectDn350
Iso6708ObjectDn40
Iso6708ObjectDn400
Iso6708ObjectDn450
Iso6708ObjectDn50
Iso6708ObjectDn500
Iso6708ObjectDn600
Iso6708ObjectDn65
Iso6708ObjectDn700
Iso6708ObjectDn80
Iso6708ObjectDn800
Iso6708ObjectDn900
Nps1/2Artefact
Nps1/4Artefact
Nps10Artefact
Nps12Artefact
Nps14Artefact
Nps16Artefact
Nps18Artefact
Nps1Artefact
Nps1_1/2Artefact
Nps1_1/4Artefact
Nps20Artefact
Nps24Artefact
Nps2Artefact
Nps2_1/2Artefact
Nps3/4Artefact
Nps30Artefact
Nps36Artefact
Nps3Artefact
Nps3_1/2Artefact
Nps42Artefact
Nps48Artefact
Nps4Artefact
Nps54Artefact
Nps5Artefact
Nps60Artefact
Nps6Artefact
Nps8Artefact

Literals

Name	Symbol	RDL Reference
NULL	*null*	
Din2448ObjectDn100	DN 100 (DIN 2448)	DIN 2448 OBJECT DN 100 http://sandbox.dexpi.org/rdl/Din2448ObjectDn100
Din2448ObjectDn125	DN 125 (DIN 2448)	DIN 2448 OBJECT DN 125 http://sandbox.dexpi.org/rdl/Din2448ObjectDn125
Din2448ObjectDn15	DN 15 (DIN 2448)	DIN 2448 OBJECT DN 15 http://sandbox.dexpi.org/rdl/Din2448ObjectDn15
Din2448ObjectDn150	DN 150 (DIN 2448)	DIN 2448 OBJECT DN 150 http://sandbox.dexpi.org/rdl/Din2448ObjectDn150
Din2448ObjectDn20	DN 20 (DIN 2448)	DIN 2448 OBJECT DN 20 http://sandbox.dexpi.org/rdl/Din2448ObjectDn20
Din2448ObjectDn200	DN 200 (DIN 2448)	DIN 2448 OBJECT DN 200 http://sandbox.dexpi.org/rdl/Din2448ObjectDn200
Din2448ObjectDn25	DN 25 (DIN 2448)	DIN 2448 OBJECT DN 25 http://sandbox.dexpi.org/rdl/Din2448ObjectDn25
Din2448ObjectDn32	DN 32 (DIN 2448)	DIN 2448 OBJECT DN 32 http://sandbox.dexpi.org/rdl/Din2448ObjectDn32
Din2448ObjectDn40	DN 40 (DIN 2448)	DIN 2448 OBJECT DN 40 http://sandbox.dexpi.org/rdl/Din2448ObjectDn40
Din2448ObjectDn50	DN 50 (DIN 2448)	DIN 2448 OBJECT DN 50 http://sandbox.dexpi.org/rdl/Din2448ObjectDn50
Din2448ObjectDn65	DN 65 (DIN 2448)	DIN 2448 OBJECT DN 65 http://sandbox.dexpi.org/rdl/Din2448ObjectDn65
Din2448ObjectDn80	DN 80 (DIN 2448)	DIN 2448 OBJECT DN 80 http://sandbox.dexpi.org/rdl/Din2448ObjectDn80
Iso6708ObjectDn100	DN 100 (ISO 6708)	ISO 6708 OBJECT DN 100 http://sandbox.dexpi.org/rdl/Iso6708ObjectDn100
Iso6708ObjectDn1000	DN 1000 (ISO 6708)	ISO 6708 OBJECT DN 1000 http://sandbox.dexpi.org/rdl/Iso6708ObjectDn1000
Iso6708ObjectDn1200	DN 1200 (ISO 6708)	ISO 6708 OBJECT DN 1200 http://sandbox.dexpi.org/rdl/Iso6708ObjectDn1200
Iso6708ObjectDn125	DN 125 (ISO 6708)	ISO 6708 OBJECT DN 125 http://sandbox.dexpi.org/rdl/Iso6708ObjectDn125
Iso6708ObjectDn1400	DN 1400 (ISO 6708)	ISO 6708 OBJECT DN 1400 http://sandbox.dexpi.org/rdl/Iso6708ObjectDn1400
Iso6708ObjectDn15	DN 15 (ISO 6708)	ISO 6708 OBJECT DN 15 http://sandbox.dexpi.org/rdl/Iso6708ObjectDn15
Iso6708ObjectDn150	DN 150 (ISO 6708)	ISO 6708 OBJECT DN 150 http://sandbox.dexpi.org/rdl/Iso6708ObjectDn150
Iso6708ObjectDn1600	DN 1600 (ISO 6708)	ISO 6708 OBJECT DN 1600 http://sandbox.dexpi.org/rdl/Iso6708ObjectDn1600
Iso6708ObjectDn20	DN 20 (ISO 6708)	ISO 6708 OBJECT DN 20 http://sandbox.dexpi.org/rdl/Iso6708ObjectDn20
Iso6708ObjectDn200	DN 200 (ISO 6708)	ISO 6708 OBJECT DN 200 http://sandbox.dexpi.org/rdl/Iso6708ObjectDn200
Iso6708ObjectDn25	DN 25 (ISO 6708)	ISO 6708 OBJECT DN 25 http://sandbox.dexpi.org/rdl/Iso6708ObjectDn25
Iso6708ObjectDn250	DN 250 (ISO 6708)	ISO 6708 OBJECT DN 250 http://sandbox.dexpi.org/rdl/Iso6708ObjectDn250

(continued on next page)

Name	Symbol	RDL Reference
Iso6708ObjectDn300	DN 300 (ISO 6708)	ISO 6708 OBJECT DN 300 http://sandbox.dexpi.org/rdl/Iso6708ObjectDn300
Iso6708ObjectDn32	DN 32 (ISO 6708)	ISO 6708 OBJECT DN 32 http://sandbox.dexpi.org/rdl/Iso6708ObjectDn32
Iso6708ObjectDn350	DN 350 (ISO 6708)	ISO 6708 OBJECT DN 350 http://sandbox.dexpi.org/rdl/Iso6708ObjectDn350
Iso6708ObjectDn40	DN 40 (ISO 6708)	ISO 6708 OBJECT DN 40 http://sandbox.dexpi.org/rdl/Iso6708ObjectDn40
Iso6708ObjectDn400	DN 400 (ISO 6708)	ISO 6708 OBJECT DN 400 http://sandbox.dexpi.org/rdl/Iso6708ObjectDn400
Iso6708ObjectDn450	DN 450 (ISO 6708)	ISO 6708 OBJECT DN 450 http://sandbox.dexpi.org/rdl/Iso6708ObjectDn450
Iso6708ObjectDn50	DN 50 (ISO 6708)	ISO 6708 OBJECT DN 50 http://sandbox.dexpi.org/rdl/Iso6708ObjectDn50
Iso6708ObjectDn500	DN 500 (ISO 6708)	ISO 6708 OBJECT DN 500 http://sandbox.dexpi.org/rdl/Iso6708ObjectDn500
Iso6708ObjectDn600	DN 600 (ISO 6708)	ISO 6708 OBJECT DN 600 http://sandbox.dexpi.org/rdl/Iso6708ObjectDn600
Iso6708ObjectDn65	DN 65 (ISO 6708)	ISO 6708 OBJECT DN 65 http://sandbox.dexpi.org/rdl/Iso6708ObjectDn65
Iso6708ObjectDn700	DN 700 (ISO 6708)	ISO 6708 OBJECT DN 700 http://sandbox.dexpi.org/rdl/Iso6708ObjectDn700
Iso6708ObjectDn80	DN 80 (ISO 6708)	ISO 6708 OBJECT DN 80 http://sandbox.dexpi.org/rdl/Iso6708ObjectDn80
Iso6708ObjectDn800	DN 800 (ISO 6708)	ISO 6708 OBJECT DN 800 http://sandbox.dexpi.org/rdl/Iso6708ObjectDn800
Iso6708ObjectDn900	DN 900 (ISO 6708)	ISO 6708 OBJECT DN 900 http://sandbox.dexpi.org/rdl/Iso6708ObjectDn900
Nps1/2Artefact	NPS 1/2	NPS 1/2 ARTEFACT http://data.posccaesar.org/rdl/RDS20863408113
Nps1/4Artefact	NPS 1/4	NPS 1/4 ARTEFACT http://data.posccaesar.org/rdl/RDS2086340839
Nps10Artefact	NPS 10	NPS 10 ARTEFACT http://data.posccaesar.org/rdl/RDS20863408298
Nps12Artefact	NPS 12	NPS 12 ARTEFACT http://data.posccaesar.org/rdl/RDS208634082110
Nps14Artefact	NPS 14	NPS 14 ARTEFACT http://data.posccaesar.org/rdl/RDS208634082122
Nps16Artefact	NPS 16	NPS 16 ARTEFACT http://data.posccaesar.org/rdl/RDS208634082134
Nps18Artefact	NPS 18	NPS 18 ARTEFACT http://data.posccaesar.org/rdl/RDS208634082146
Nps1Artefact	NPS 1	NPS 1 ARTEFACT http://data.posccaesar.org/rdl/RDS20863408137
Nps1_1/2Artefact	NPS 1 1/2	NPS 1 1/2 ARTEFACT http://data.posccaesar.org/rdl/RDS2086340822
Nps1_1/4Artefact	NPS 1 1/4	NPS 1 1/4 ARTEFACT http://data.posccaesar.org/rdl/RDS20863408321
Nps20Artefact	NPS 20	NPS 20 ARTEFACT http://data.posccaesar.org/rdl/RDS208634082158
Nps24Artefact	NPS 24	NPS 24 ARTEFACT http://data.posccaesar.org/rdl/RDS208634082170

(continued on next page)

Name	Symbol	RDL Reference
Nps2Artefact	NPS 2	NPS 2 ARTEFACT http://data.posccaesar.org/rdl/RDS20863408214
Nps2_1/2Artefact	NPS 2 1/2	NPS 2 1/2 ARTEFACT http://data.posccaesar.org/rdl/RDS20863408226
Nps3/4Artefact	NPS 3/4	NPS 3/4 ARTEFACT http://data.posccaesar.org/rdl/RDS20863408125
Nps30Artefact	NPS 30	NPS 30 ARTEFACT http://data.posccaesar.org/rdl/RDS208634082182
Nps36Artefact	NPS 36	NPS 36 ARTEFACT http://data.posccaesar.org/rdl/RDS208634082194
Nps3Artefact	NPS 3	NPS 3 ARTEFACT http://data.posccaesar.org/rdl/RDS20863408238
Nps3_1/2Artefact	NPS 3 1/2	NPS 3 1/2 ARTEFACT http://data.posccaesar.org/rdl/RDS20863408333
Nps42Artefact	NPS 42	NPS 42 ARTEFACT http://data.posccaesar.org/rdl/RDS208634082206
Nps48Artefact	NPS 48	NPS 48 ARTEFACT http://data.posccaesar.org/rdl/RDS208634082218
Nps4Artefact	NPS 4	NPS 4 ARTEFACT http://data.posccaesar.org/rdl/RDS20863408250
Nps54Artefact	NPS 54	NPS 54 ARTEFACT http://data.posccaesar.org/rdl/RDS208634082230
Nps5Artefact	NPS 5	NPS 5 ARTEFACT http://data.posccaesar.org/rdl/RDS20863408262
Nps60Artefact	NPS 60	NPS 60 ARTEFACT http://data.posccaesar.org/rdl/RDS208634082242
Nps6Artefact	NPS 6	NPS 6 ARTEFACT http://data.posccaesar.org/rdl/RDS20863408274
Nps8Artefact	NPS 8	NPS 8 ARTEFACT http://data.posccaesar.org/rdl/RDS20863408286

Implementation in Proteus Schema

All data attributes with type *NominalDiameterStandardClassification* are implemented as *DEXPI generic attributes for enumeration values*. In a `<GenericAttribute>` element, the *NominalDiameterStandardClassification* literal is given by means of its RDL reference in the table above. The `Value` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `ValueURI` attribute of the element is the URI of the RDL reference.

Example

NominalDiameterStandardClassification : Din2448ObjectDn25

Example: Implementation in Proteus Schema

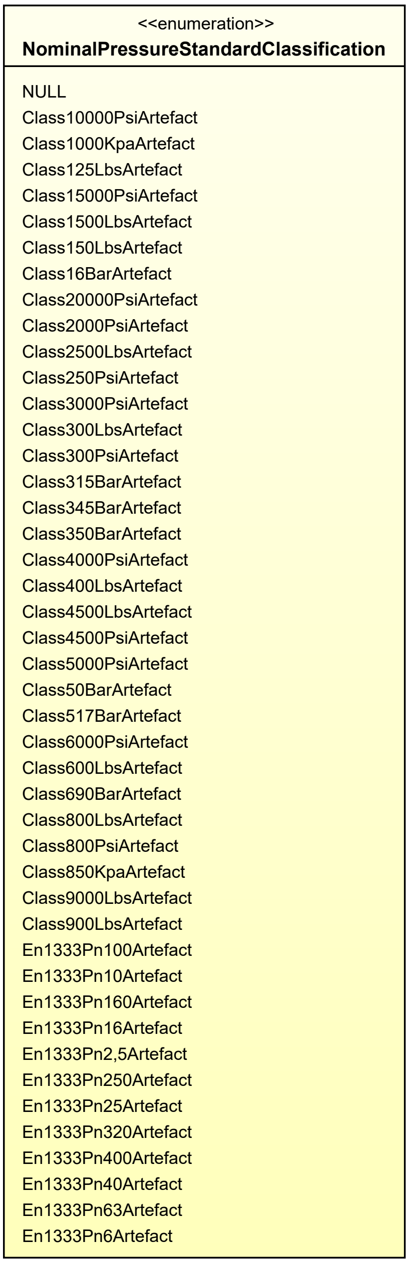
```
<GenericAttribute
  Format="anyURI"
  Value="Din2448ObjectDn25"
  ValueURI="http://sandbox.dexpi.org/rdl/Din2448ObjectDn25" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance *Din2448ObjectDn25*. For a complete example, see the attribute *TubeNominalDiameterStandard* of the DEXPI class *TubeBundle*.

11.17. NominalPressureStandardClassification

11.17.1 Overview

Enumeration



Literals

Name	Symbol	RDL Reference
NULL	*null*	
Class10000PsiArtefact	Class 10000 psi	CLASS 10000 PSI ARTEFACT http://sandbox.dexpi.org/rdl/Class10000PsiArtefact
Class1000KpaArtefact	Class 1000 kpa	CLASS 1000 KPA ARTEFACT http://sandbox.dexpi.org/rdl/Class1000KpaArtefact
Class125LbsArtefact	Class 125 lbs	CLASS 125 LBS ARTEFACT http://sandbox.dexpi.org/rdl/Class125LbsArtefact
Class15000PsiArtefact	Class 15000 psi	CLASS 15000 PSI ARTEFACT http://sandbox.dexpi.org/rdl/Class15000PsiArtefact
Class1500LbsArtefact	Class 1500 lbs	CLASS 1500 LBS ARTEFACT http://sandbox.dexpi.org/rdl/Class1500LbsArtefact
Class150LbsArtefact	Class 150 lbs	CLASS 150 LBS ARTEFACT http://sandbox.dexpi.org/rdl/Class150LbsArtefact
Class16BarArtefact	Class 16 bar	CLASS 16 BAR ARTEFACT http://sandbox.dexpi.org/rdl/Class16BarArtefact
Class20000PsiArtefact	Class 20000 psi	CLASS 20000 PSI ARTEFACT http://sandbox.dexpi.org/rdl/Class20000PsiArtefact
Class2000PsiArtefact	Class 2000 psi	CLASS 2000 PSI ARTEFACT http://sandbox.dexpi.org/rdl/Class2000PsiArtefact
Class2500LbsArtefact	Class 2500 lbs	CLASS 2500 LBS ARTEFACT http://sandbox.dexpi.org/rdl/Class2500LbsArtefact
Class250PsiArtefact	Class 250 psi	CLASS 250 PSI ARTEFACT http://sandbox.dexpi.org/rdl/Class250PsiArtefact
Class3000PsiArtefact	Class 3000 psi	CLASS 3000 PSI ARTEFACT http://sandbox.dexpi.org/rdl/Class3000PsiArtefact
Class300LbsArtefact	Class 300 lbs	CLASS 300 LBS ARTEFACT http://sandbox.dexpi.org/rdl/Class300LbsArtefact
Class300PsiArtefact	Class 300 psi	CLASS 300 PSI ARTEFACT http://sandbox.dexpi.org/rdl/Class300PsiArtefact
Class315BarArtefact	Class 315 bar	CLASS 315 BAR ARTEFACT http://sandbox.dexpi.org/rdl/Class315BarArtefact
Class345BarArtefact	Class 345 bar	CLASS 345 BAR ARTEFACT http://sandbox.dexpi.org/rdl/Class345BarArtefact
Class350BarArtefact	Class 350 bar	CLASS 350 BAR ARTEFACT http://sandbox.dexpi.org/rdl/Class350BarArtefact
Class4000PsiArtefact	Class 4000 psi	CLASS 4000 PSI ARTEFACT http://sandbox.dexpi.org/rdl/Class4000PsiArtefact
Class400LbsArtefact	Class 400 lbs	CLASS 400 LBS ARTEFACT http://sandbox.dexpi.org/rdl/Class400LbsArtefact
Class4500LbsArtefact	Class 4500 lbs	CLASS 4500 LBS ARTEFACT http://sandbox.dexpi.org/rdl/Class4500LbsArtefact
Class4500PsiArtefact	Class 4500 psi	CLASS 4500 PSI ARTEFACT http://sandbox.dexpi.org/rdl/Class4500PsiArtefact
Class5000PsiArtefact	Class 5000 psi	CLASS 5000 PSI ARTEFACT http://sandbox.dexpi.org/rdl/Class5000PsiArtefact
Class50BarArtefact	Class 50 bar	CLASS 50 BAR ARTEFACT http://sandbox.dexpi.org/rdl/Class50BarArtefact
Class517BarArtefact	Class 517 bar	CLASS 517 BAR ARTEFACT http://sandbox.dexpi.org/rdl/Class517BarArtefact

(continued on next page)

Name	Symbol	RDL Reference
Class6000PsiArtefact	Class 6000 psi	CLASS 6000 PSI ARTEFACT http://sandbox.dexpi.org/rdl/Class6000PsiArtefact
Class600LbsArtefact	Class 600 lbs	CLASS 600 LBS ARTEFACT http://sandbox.dexpi.org/rdl/Class600LbsArtefact
Class690BarArtefact	Class 690 bar	CLASS 690 BAR ARTEFACT http://sandbox.dexpi.org/rdl/Class690BarArtefact
Class800LbsArtefact	Class 800 lbs	CLASS 800 LBS ARTEFACT http://sandbox.dexpi.org/rdl/Class800LbsArtefact
Class800PsiArtefact	Class 800 psi	CLASS 800 PSI ARTEFACT http://sandbox.dexpi.org/rdl/Class800PsiArtefact
Class850KpaArtefact	Class 850 kpa	CLASS 850 KPA ARTEFACT http://sandbox.dexpi.org/rdl/Class850KpaArtefact
Class9000LbsArtefact	Class 9000 lbs	CLASS 9000 LBS ARTEFACT http://sandbox.dexpi.org/rdl/Class9000LbsArtefact
Class900LbsArtefact	Class 900 lbs	CLASS 900 LBS ARTEFACT http://sandbox.dexpi.org/rdl/Class900LbsArtefact
En1333Pn100Artefact	PN 100 (EN 1333)	EN 1333 PN 100 ARTEFACT http://sandbox.dexpi.org/rdl/En1333Pn100Artefact
En1333Pn10Artefact	PN 10 (EN 1333)	EN 1333 PN 10 ARTEFACT http://sandbox.dexpi.org/rdl/En1333Pn10Artefact
En1333Pn160Artefact	PN 160 (EN 1333)	EN 1333 PN 160 ARTEFACT http://sandbox.dexpi.org/rdl/En1333Pn160Artefact
En1333Pn16Artefact	PN 16 (EN 1333)	EN 1333 PN 16 ARTEFACT http://sandbox.dexpi.org/rdl/En1333Pn16Artefact
En1333Pn2,5Artefact	PN 2,5 (EN 1333)	EN 1333 PN 2,5 ARTEFACT http://sandbox.dexpi.org/rdl/En1333Pn2,5Artefact
En1333Pn250Artefact	PN 250 (EN 1333)	EN 1333 PN 250 ARTEFACT http://sandbox.dexpi.org/rdl/En1333Pn250Artefact
En1333Pn25Artefact	PN 25 (EN 1333)	EN 1333 PN 25 ARTEFACT http://sandbox.dexpi.org/rdl/En1333Pn25Artefact
En1333Pn320Artefact	PN 320 (EN 1333)	EN 1333 PN 320 ARTEFACT http://sandbox.dexpi.org/rdl/En1333Pn320Artefact
En1333Pn400Artefact	PN 400 (EN 1333)	EN 1333 PN 400 ARTEFACT http://sandbox.dexpi.org/rdl/En1333Pn400Artefact
En1333Pn40Artefact	PN 40 (EN 1333)	EN 1333 PN 40 ARTEFACT http://sandbox.dexpi.org/rdl/En1333Pn40Artefact
En1333Pn63Artefact	PN 63 (EN 1333)	EN 1333 PN 63 ARTEFACT http://sandbox.dexpi.org/rdl/En1333Pn63Artefact
En1333Pn6Artefact	PN 6 (EN 1333)	EN 1333 PN 6 ARTEFACT http://sandbox.dexpi.org/rdl/En1333Pn6Artefact

Implementation in Proteus Schema

All data attributes with type *NominalPressureStandardClassification* are implemented as *DEXPI generic attributes for enumeration values*. In a `<GenericAttribute>` element, the *NominalPressureStandardClassification* literal is given by means of its RDL reference in the table above. The `Value` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `ValueURI` attribute of the element is the URI of the RDL reference.

Example

NominalPressureStandardClassification : En1333Pn40Artefact

Example: Implementation in Proteus Schema

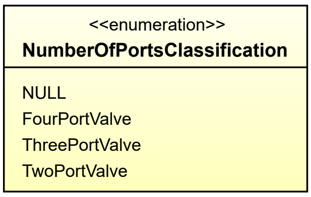
```
<GenericAttribute
  Format="anyURI"
  Value="En1333Pn40Artefact"
  ValueURI="http://sandbox.dexpi.org/rdl/En1333Pn40Artefact" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance *En1333Pn40Artefact*. For a complete example, see the attribute *NominalPressureStandard* of the DEXPI class *Nozzle*.

11.18. NumberOfPortsClassification

11.18.1 Overview

Enumeration



Literals

Name	Symbol	RDL Reference
NULL	*null*	
FourPortValve	4 port valve	FOUR PORT VALVE http://data.posccaesar.org/rdl/RDS6330166
ThreePortValve	3 port valve	THREE PORT VALVE http://data.posccaesar.org/rdl/RDS6331437
TwoPortValve	2 port valve	TWO PORT VALVE http://data.posccaesar.org/rdl/RDS11506315

Implementation in Proteus Schema

All data attributes with type *NumberOfPortsClassification* are implemented as *DEXPI generic attributes for enumeration values*. In a `<GenericAttribute>` element, the *NumberOfPortsClassification* literal is given by means of its RDL reference in the table above. The `Value` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `ValueURI` attribute of the element is the URI of the RDL reference.

Example

NumberOfPortsClassification : TwoPortValve

Example: Implementation in Proteus Schema

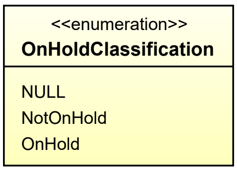
```
<GenericAttribute
  Format="anyURI"
  Value="TwoPortValve"
  ValueURI="http://data.posccaesar.org/rdl/RDS11506315" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance *TwoPortValve*. For a complete example, see the attribute *NumberOfPorts* of the DEXPI class *OperatedValve*.

11.19. OnHoldClassification

11.19.1 Overview

Enumeration



Literals

Name	Symbol	RDL Reference
NULL	*null*	
NotOnHold	not on hold	NOT ON HOLD http://sandbox.dexpi.org/rdl/NotOnHold
OnHold	on hold	ON HOLD http://sandbox.dexpi.org/rdl/OnHold

Implementation in Proteus Schema

All data attributes with type *OnHoldClassification* are implemented as *DEXPI generic attributes for enumeration values*. In a `<GenericAttribute>` element, the *OnHoldClassification* literal is given by means of its RDL reference in the table above. The `Value` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `ValueURI` attribute of the element is the URI of the RDL reference.

Example

OnHoldClassification : OnHold

Example: Implementation in Proteus Schema

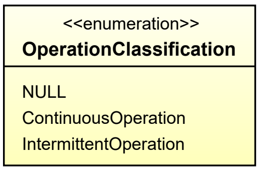
```
<GenericAttribute
  Format="anyURI"
  Value="OnHold"
  ValueURI="http://sandbox.dexpi.org/rdl/OnHold" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance *OnHold*. For a complete example, see the attribute *OnHold* of the DEXPI class *PipingComponent*.

11.20. OperationClassification

11.20.1 Overview

Enumeration



Literals

Name	Symbol	RDL Reference
NULL	*null*	
ContinuousOperation	continuous operation	CONTINUOUS OPERATION http://data.posccaesar.org/rdl/RDS9710162
IntermittentOperation	intermittent operation	INTERMITTENT OPERATION http://data.posccaesar.org/rdl/RDS9705752

Implementation in Proteus Schema

All data attributes with type *OperationClassification* are implemented as *DEXPI generic attributes for enumeration values*. In a `<GenericAttribute>` element, the *OperationClassification* literal is given by means of its RDL reference in the table above. The `Value` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `ValueURI` attribute of the element is the URI of the RDL reference.

Example

OperationClassification : ContinuousOperation

Example: Implementation in Proteus Schema

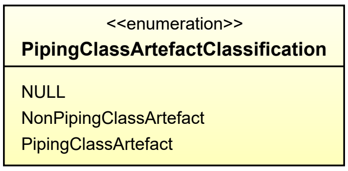
```
<GenericAttribute
  Format="anyURI"
  Value="ContinuousOperation"
  ValueURI="http://data.posccaesar.org/rdl/RDS9710162" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance *ContinuousOperation*. For a complete example, see the attribute *Operation* of the DEXPI class *OperatedValve*.

11.21. PipingClassArtefactClassification

11.21.1 Overview

Enumeration



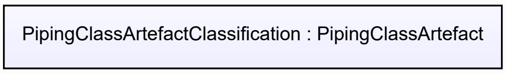
Literals

Name	Symbol	RDL Reference
NULL	*null*	
NonPipingClassArtefact	non-piping-class artefact	NON PIPING CLASS ARTEFACT http://sandbox.dexpi.org/rdl/NonPipingClassArtefact
PipingClassArtefact	piping class artefact	PIPING CLASS ARTEFACT http://sandbox.dexpi.org/rdl/PipingClassArtefact

Implementation in Proteus Schema

All data attributes with type *PipingClassArtefactClassification* are implemented as *DEXPI generic attributes for enumeration values*. In a `<GenericAttribute>` element, the *PipingClassArtefactClassification* literal is given by means of its RDL reference in the table above. The `Value` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `ValueURI` attribute of the element is the URI of the RDL reference.

Example



Example: Implementation in Proteus Schema

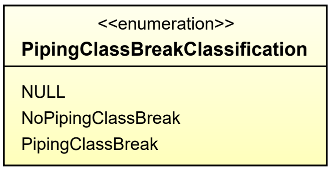
```
<GenericAttribute
  Format="anyURI"
  Value="PipingClassArtefact"
  ValueURI="http://sandbox.dexpi.org/rdl/PipingClassArtefact" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance *PipingClassArtefact*. For a complete example, see the attribute *PipingClassArtefact* of the DEXPI class *PipingComponent*.

11.22. PipingClassBreakClassification

11.22.1 Overview

Enumeration



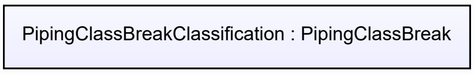
Literals

Name	Symbol	RDL Reference
NULL	*null*	
NoPipingClassBreak	no piping class break	NO PIPING CLASS BREAK http://sandbox.dexpi.org/rdl/NoPipingClassBreak
PipingClassBreak	piping class break	PIPING CLASS BREAK http://sandbox.dexpi.org/rdl/PipingClassBreak

Implementation in Proteus Schema

All data attributes with type *PipingClassBreakClassification* are implemented as *DEXPI generic attributes for enumeration values*. In a `<GenericAttribute>` element, the *PipingClassBreakClassification* literal is given by means of its RDL reference in the table above. The `Value` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `ValueURI` attribute of the element is the URI of the RDL reference.

Example



Example: Implementation in Proteus Schema

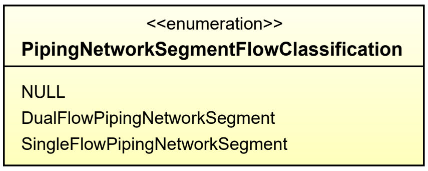
```
<GenericAttribute
  Format="anyURI"
  Value="PipingClassBreak"
  ValueURI="http://sandbox.dexpi.org/rdl/PipingClassBreak" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance *PipingClassBreak*. For a complete example, see the attribute *PipingClassBreak* of the DEXPI class *PropertyBreak*.

11.23. PipingNetworkSegmentFlowClassification

11.23.1 Overview

Enumeration


<<enumeration>>
PipingNetworkSegmentFlowClassification
NULL
DualFlowPipingNetworkSegment
SingleFlowPipingNetworkSegment

Literals

Name	Symbol	RDL Reference
NULL	*null*	
DualFlowPipingNetworkSegment	dual flow	DUAL FLOW PIPING NETWORK SEGMENT http://sandbox.dexpi.org/rdl/DualFlowPipingNetworkSegment
SingleFlowPipingNetworkSegment	single flow	SINGLE FLOW PIPING NETWORK SEGMENT http://sandbox.dexpi.org/rdl/SingleFlowPipingNetworkSegment

Implementation in Proteus Schema

All data attributes with type *PipingNetworkSegmentFlowClassification* are implemented as *DEXPI generic attributes for enumeration values*. In a `<GenericAttribute>` element, the *PipingNetworkSegmentFlowClassification* literal is given by means of its RDL reference in the table above. The `Value` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `ValueURI` attribute of the element is the URI of the RDL reference.

Example

PipingNetworkSegmentFlowClassification : DualFlowPipingNetworkSegment

Example: Implementation in Proteus Schema

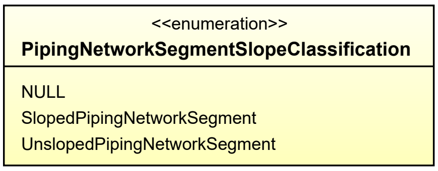
```
<GenericAttribute
  Format="anyURI"
  Value="DualFlowPipingNetworkSegment"
  ValueURI="http://sandbox.dexpi.org/rdl/DualFlowPipingNetworkSegment" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance *DualFlowPipingNetworkSegment*. For a complete example, see the attribute *FlowDirection* of the DEXPI class *PipingNetworkSegment*.

11.24. PipingNetworkSegmentSlopeClassification

11.24.1 Overview

Enumeration


<<enumeration>>
PipingNetworkSegmentSlopeClassification
NULL
SlopedPipingNetworkSegment
UnslopedPipingNetworkSegment

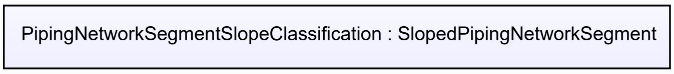
Literals

Name	Symbol	RDL Reference
NULL	*null*	
SlopedPipingNetworkSegment	sloped	SLOPED PIPING NETWORK SEGMENT http://sandbox.dexpi.org/rdl/SlopedPipingNetworkSegment
UnslopedPipingNetworkSegment	not sloped	UNSLOPED PIPING NETWORK SEGMENT http://sandbox.dexpi.org/rdl/UnslopedPipingNetworkSegment

Implementation in Proteus Schema

All data attributes with type *PipingNetworkSegmentSlopeClassification* are implemented as *DEXPI generic attributes for enumeration values*. In a `<GenericAttribute>` element, the *PipingNetworkSegmentSlopeClassification* literal is given by means of its RDL reference in the table above. The `Value` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `ValueURI` attribute of the element is the URI of the RDL reference.

Example


PipingNetworkSegmentSlopeClassification : SlopedPipingNetworkSegment

Example: Implementation in Proteus Schema

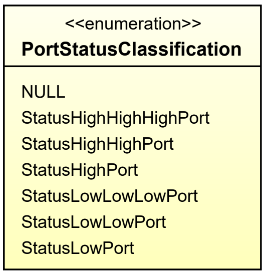
```
<GenericAttribute
  Format="anyURI"
  Value="SlopedPipingNetworkSegment"
  ValueURI="http://sandbox.dexpi.org/rdl/SlopedPipingNetworkSegment" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance *SlopedPipingNetworkSegment*. For a complete example, see the attribute *Slope* of the DEXPI class *PipingNetworkSegment*.

11.25. PortStatusClassification

11.25.1 Overview

Enumeration



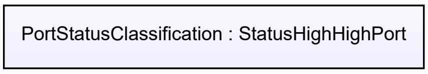
Literals

Name	Symbol	RDL Reference
NULL	*null*	
StatusHighHighHighPort	HHH	STATUS HIGH HIGH HIGH PORT http://sandbox.dexpi.org/rdl/StatusHighHighHighPort
StatusHighHighPort	HH	STATUS HIGH HIGH PORT http://data.posccaesar.org/rdl/RDS323099
StatusHighPort	H	STATUS HIGH PORT http://data.posccaesar.org/rdl/RDS323144
StatusLowLowLowPort	LLL	STATUS LOW LOW LOW PORT http://sandbox.dexpi.org/rdl/StatusLowLowLowPort
StatusLowLowPort	LL	STATUS LOW LOW PORT http://data.posccaesar.org/rdl/RDS323189
StatusLowPort	L	STATUS LOW PORT http://data.posccaesar.org/rdl/RDS323234

Implementation in Proteus Schema

All data attributes with type *PortStatusClassification* are implemented as *DEXPI generic attributes for enumeration values*. In a `<GenericAttribute>` element, the *PortStatusClassification* literal is given by means of its RDL reference in the table above. The `Value` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `ValueURI` attribute of the element is the URI of the RDL reference.

Example



Example: Implementation in Proteus Schema

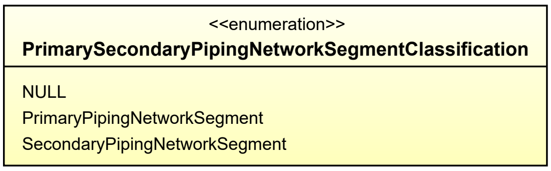
```
<GenericAttribute
  Format="anyURI"
  Value="StatusHighHighPort"
  ValueURI="http://data.posccaesar.org/rdl/RDS323099" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance *StatusHighHighPort*. For a complete example, see the attribute *PortStatus* of the DEXPI class *SignalConveyingFunction*.

11.26. PrimarySecondaryPipingNetworkSegmentClassification

11.26.1 Overview

Enumeration


<<enumeration>>
PrimarySecondaryPipingNetworkSegmentClassification
NULL
PrimaryPipingNetworkSegment
SecondaryPipingNetworkSegment

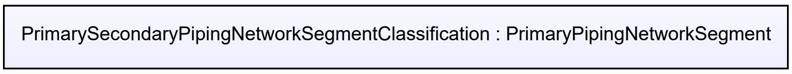
Literals

Name	Symbol	RDL Reference
NULL	*null*	
PrimaryPipingNetworkSegment	primary segment	PRIMARY PIPING NETWORK SEGMENT http://sandbox.dexpi.org/rdl/PrimaryPipingNetworkSegment
SecondaryPipingNetworkSegment	secondary segment	SECONDARY PIPING NETWORK SEGMENT http://sandbox.dexpi.org/rdl/SecondaryPipingNetworkSegment

Implementation in Proteus Schema

All data attributes with type *PrimarySecondaryPipingNetworkSegmentClassification* are implemented as *DEXPI generic attributes for enumeration values*. In a `<GenericAttribute>` element, the *PrimarySecondaryPipingNetworkSegmentClassification* literal is given by means of its RDL reference in the table above. The `Value` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `ValueURI` attribute of the element is the URI of the RDL reference.

Example


PrimarySecondaryPipingNetworkSegmentClassification : PrimaryPipingNetworkSegment

Example: Implementation in Proteus Schema

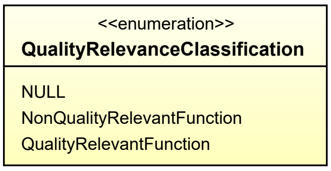
```
<GenericAttribute
  Format="anyURI"
  Value="PrimaryPipingNetworkSegment"
  ValueURI="http://sandbox.dexpi.org/rdl/PrimaryPipingNetworkSegment" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance *PrimaryPipingNetworkSegment*. For a complete example, see the attribute *PrimarySecondaryPipingNetworkSegment* of the DEXPI class *PipingNetworkSegment*.

11.27. QualityRelevanceClassification

11.27.1 Overview

Enumeration



Literals

Name	Symbol	RDL Reference
NULL	*null*	
NonQualityRelevantFunction	not quality relevant	NON QUALITY RELEVANT FUNCTION http://sandbox.dexpi.org/rdl/NonQualityRelevantFunction
QualityRelevantFunction	quality relevant	QUALITY RELEVANT FUNCTION http://sandbox.dexpi.org/rdl/QualityRelevantFunction

Implementation in Proteus Schema

All data attributes with type *QualityRelevanceClassification* are implemented as *DEXPI generic attributes for enumeration values*. In a `<GenericAttribute>` element, the *QualityRelevanceClassification* literal is given by means of its RDL reference in the table above. The `Value` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `ValueURI` attribute of the element is the URI of the RDL reference.

Example

QualityRelevanceClassification : QualityRelevantFunction

Example: Implementation in Proteus Schema

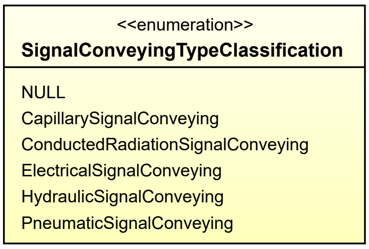
```
<GenericAttribute
  Format="anyURI"
  Value="QualityRelevantFunction"
  ValueURI="http://sandbox.dexpi.org/rdl/QualityRelevantFunction" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance *QualityRelevantFunction*. For a complete example, see the attribute *QualityRelevance* of the DEXPI class *ProcessInstrumentationFunction*.

11.28. SignalConveyingTypeClassification

11.28.1 Overview

Enumeration


<<enumeration>>
SignalConveyingTypeClassification
NULL
CapillarySignalConveying
ConductedRadiationSignalConveying
ElectricalSignalConveying
HydraulicSignalConveying
PneumaticSignalConveying

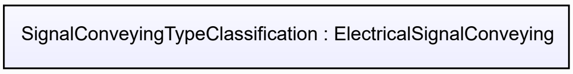
Literals

Name	Symbol	RDL Reference
NULL	*null*	
CapillarySignalConveying	capillary	CAPILLARY SIGNAL CONVEYING http://sandbox.dexpi.org/rdl/CapillarySignalConveying
ConductedRadiationSignalConveying	conducted radiation	CONDUCTED RADIATION SIGNAL CONVEYING http://sandbox.dexpi.org/rdl/ConductedRadiationSignalConv
ElectricalSignalConveying	electrical	ELECTRICAL SIGNAL CONVEYING http://sandbox.dexpi.org/rdl/ElectricalSignalConveying
HydraulicSignalConveying	hydraulic	HYDRAULIC SIGNAL CONVEYING http://sandbox.dexpi.org/rdl/HydraulicSignalConveying
PneumaticSignalConveying	pneumatic	PNEUMATIC SIGNAL CONVEYING http://sandbox.dexpi.org/rdl/PneumaticSignalConveying

Implementation in Proteus Schema

All data attributes with type *SignalConveyingTypeClassification* are implemented as *DEXPI generic attributes for enumeration values*. In a `<GenericAttribute>` element, the *SignalConveyingTypeClassification* literal is given by means of its RDL reference in the table above. The `Value` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `ValueURI` attribute of the element is the URI of the RDL reference.

Example


SignalConveyingTypeClassification : ElectricalSignalConveying

Example: Implementation in Proteus Schema

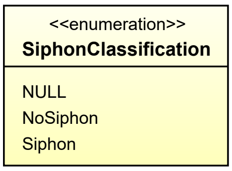
```
<GenericAttribute
  Format="anyURI"
  Value="ElectricalSignalConveying"
  ValueURI="http://sandbox.dexpi.org/rdl/ElectricalSignalConveying" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance *ElectricalSignalConveying*. For a complete example, see the attribute *SignalConveyingType* of the DEXPI class *SignalConveyingFunction*.

11.29. SiphonClassification

11.29.1 Overview

Enumeration



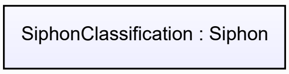
Literals

Name	Symbol	RDL Reference
NULL	*null*	
NoSiphon	no siphon	NO SIPHON http://sandbox.dexpi.org/rdl/NoSiphon
Siphon	siphon	SIPHON http://data.posccaesar.org/rdl/RDS311084

Implementation in Proteus Schema

All data attributes with type *SiphonClassification* are implemented as *DEXPI generic attributes for enumeration values*. In a `<GenericAttribute>` element, the *SiphonClassification* literal is given by means of its RDL reference in the table above. The `Value` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `ValueURI` attribute of the element is the URI of the RDL reference.

Example



Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="anyURI"
  Value="Siphon"
  ValueURI="http://data.posccaesar.org/rdl/RDS311084" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance *Siphon*. For a complete example, see the attribute *Siphon* of the DEXPI class *PipingNetworkSegment*.

Package PhysicalQuantities 12

12.1. Overview

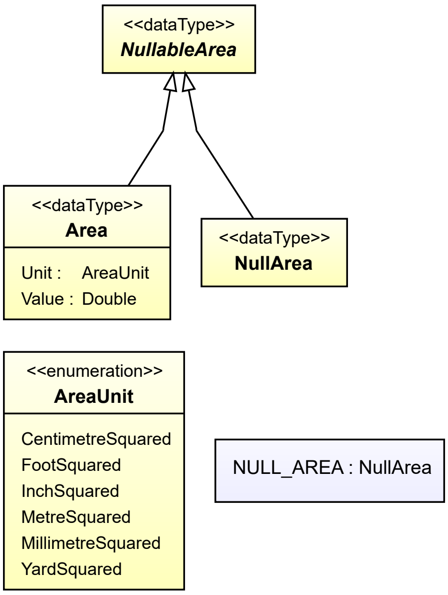
The *PhysicalQuantities* package provides data types to represent physical quantities such as the *area* of a surface or the *frequency* of a repeating event.

In DEXPI, there is a distinction between two types of physical quantities: *simple physical quantity types* and *application-dependent physical quantity types*.

12.1.1 Simple Physical Quantity Types

Simple physical quantity types are characterized by a *physical dimension*, e.g., L^2 in case of *area*, and a set of units of measurement, e.g., m^2 (meter squared), cm^2 (centimeter squared) and some more for *area*.

For illustration, we discuss the information model for *area*:



- The data type *NullableArea* is abstract. It has two concrete sub types: *Area* is used for *actual area* values, and *NullArea* is the type of the explicit *null* value *NULL_AREA*.
- An *Area* has a mandatory numerical *Value* of type *Double*. The mandatory *Unit* is one of the literals of the enumeration *AreaUnit*, for example *MetreSquared*.
- Like any enumeration literal in DEXPI, a literal for a unit of measurement has a symbol, e.g., m^2 in case of *MetreSquared*. In addition, it is identified by a UN Code. For example, the UN Code of *MetreSquared* is MTK.

There are 12 simple physical quantity types:

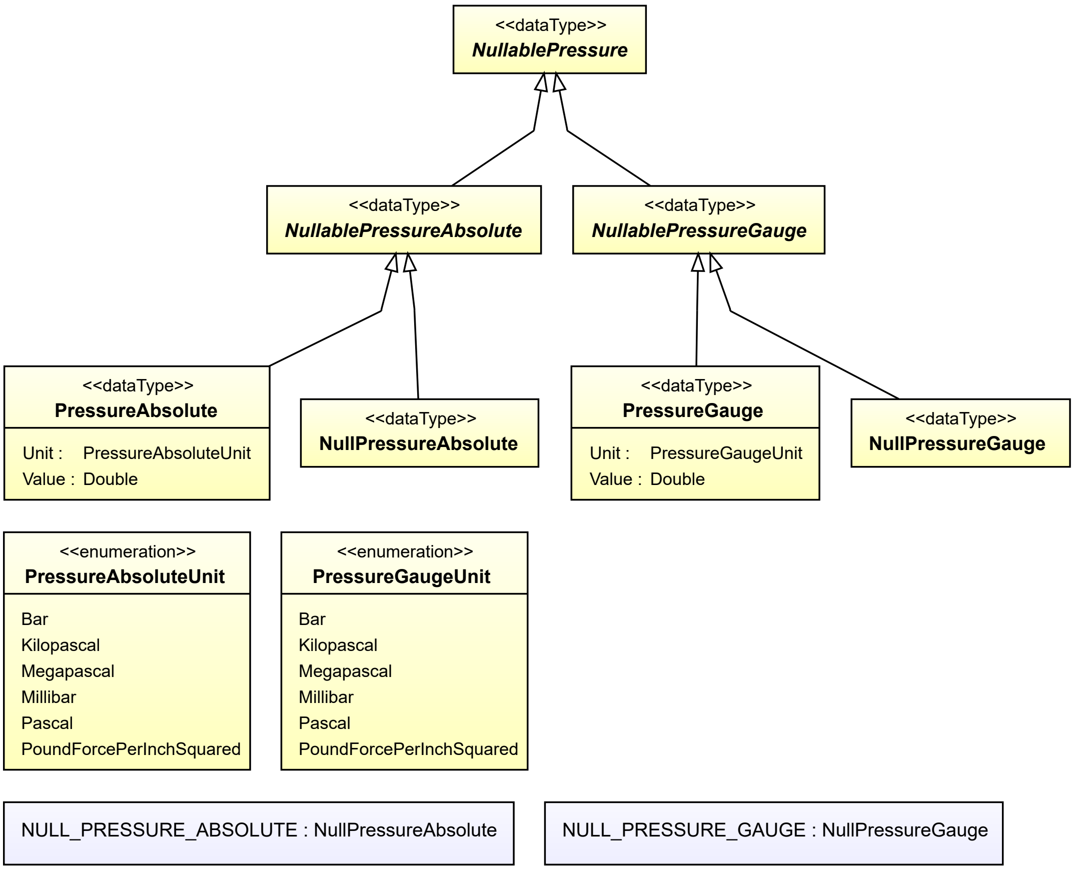
- *NullableArea*
- *NullableForce*
- *NullableHeatTransferCoefficient*

- *NullableLength*
- *NullableMass*
- *NullableMassFlowRate*
- *NullablePercentage*
- *NullablePower*
- *NullableTemperature*
- *NullableVoltage*
- *NullableVolume*
- *NullableVolumeFlowRate*

12.1.2 Application-Dependent Physical Quantity Types

An application-dependent physical quantity type such as *pressure* is characterized by a *physical dimension*, e.g., L^{-1}MT^{-2}. For an application-dependent physical quantity type, there are specializations that are intended for different application areas. These application areas may have different physical or technical meanings. The allowed units of measurement may also differ among the application areas. For example, in case of *pressure*, DEXPI distinguishes between an *absolute pressure* and a *gauge pressure*.

To illustrate the information model for application-dependent physical quantity types, we consider *pressure*:



- The abstract data type *NullablePressure* has two abstract sub types that correspond to the two application areas: *NullablePressureAbsolute* and *NullablePressureGauge*.
- Each application area is modeled in the same way as a *simple physical quantity type*. For example, *NullablePressureAbsolute* has a concrete sub type *PressureAbsolute* for *actual absolute pressure* values. Another concrete sub type is *NullPressureAbsolute*, a singleton type whose instance *NULL_PRESSURE_ABSOLUTE* is a *null* value.

- A *PressureAbsolute* has a mandatory numerical *Value* of type *Double* and a mandatory *Unit* of type *PressureAbsoluteUnit*.
- Each enumeration literal of *PressureAbsoluteUnit* has a symbol and is identified by a UN Code. For example, in case of *Bar* the symbol is bar and the UN Code is BAR.

Note that the literals of *PressureAbsoluteUnit* of *PressureGaugeUnit* have the same names, symbols, and UN Codes. From a user's perspective, this means that the same units can be used for an *absolute pressure* and for a *gauge pressure*. From the model's perspective, these are still distinct literals.

See the unit enumerations for the application areas of *frequency* for the case when the allowed units actually depend on the application area. For example, *ElectricalFrequencyUnit* contains the literal *Hertz*, whereas *RotationalFrequencyUnit* and *NumberPerTimeIntervalUnit* have no literal with name Hertz.

There are 2 application-dependent physical quantity types:

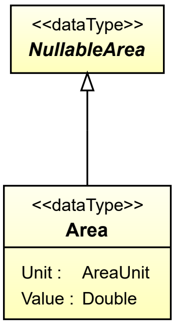
- *NullableFrequency*
- *NullablePressure*

12.2. Area

12.2.1 Overview

Data type

An *actual value* for a physical quantity of type *NullableArea*, i.e., a physical quantity that has a numerical value and a unit of measurement.



Supertypes

- *NullableArea*

Attributes (data)

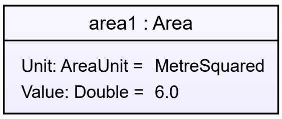
Name	Multiplicity	Type
Unit	1	*AreaUnit*
Value	1	*Double*

Implementation in Proteus Schema

All data attributes with type *NullableArea* (the base type of *Area*) are implemented as *DEXPI generic attributes for physical quantities*.

Example

The instance area1 represents an *Area* of 6.0 m^2.



Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double"
  Value="6.0"
  Units="MetreSquared"
  UnitsURI="http://data.posccaesar.org/rdl/RDS1358009" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance area1. For a complete example, see the attribute *FilterArea* of the DEXPI class *FilterUnit*.

12.2.2 Unit

Attribute (data)

The unit of measurement of the *Area*.

Multiplicity: 1

Type: *AreaUnit*

Implementation in Proteus Schema

See implementation of *Area*.

Example

See example for *Area*.

12.2.3 Value

Attribute (data)

The numerical value of the *Area*.

Multiplicity: 1

Type: *Double*

Implementation in Proteus Schema

See implementation of *Area*.

Example

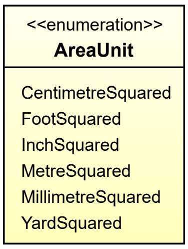
See example for *Area*.

12.3. AreaUnit

12.3.1 Overview

Enumeration

A unit of measurement for a physical quantity of type *NullableArea* with *dimension* L^2.



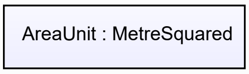
Literals

Name	Symbol	UN Code	RDL Reference
CentimetreSquared	cm^2	CMK	CENTIMETRE SQUARED http://data.posccaesar.org/rdl/RDS1357829
FootSquared	ft^2	FTK	FOOT SQUARED http://data.posccaesar.org/rdl/RDS1342934
InchSquared	in^2	INK	INCH SQUARED http://data.posccaesar.org/rdl/RDS1342979
MetreSquared	m^2	MTK	METRE SQUARED http://data.posccaesar.org/rdl/RDS1358009
MillimetreSquared	mm^2	MMK	MILLIMETRE SQUARED http://data.posccaesar.org/rdl/RDS1358189
YardSquared	yd^2	YDK	YARD SQUARED http://data.posccaesar.org/rdl/RDS1343744

Implementation in Proteus Schema

AreaUnit is only used as the type of the *Unit* attribute of *Area*. *Area* is implemented using *DEXPI generic attributes for physical quantities*. In a `<GenericAttribute>` element, the *AreaUnit* literal is given by means of its RDL reference in the table above. The `Units` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `UnitsURI` attribute of the element is the URI of the RDL reference.

Example



Example: Implementation in Proteus Schema

```
<GenericAttribute
  Units="MetreSquared"
  UnitsURI="http://data.posccaesar.org/rdl/RDS1358009" ...>
```

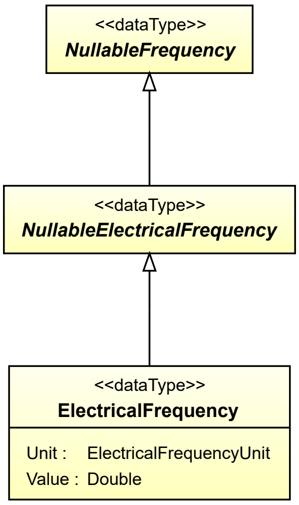
Note that the `<GenericAttribute>` element must have further attributes (`Format`, `Value`, `Name`, and `AttributeURI`). See the implementation examples for *Area* and for the *FilterArea* attribute of *FilterUnit*.

12.4. ElectricalFrequency

12.4.1 Overview

Data type

An *actual value* for a physical quantity of type *NullableElectricalFrequency*, i.e., a physical quantity that has a numerical value and a unit of measurement.



Supertypes

- *NullableElectricalFrequency*

Attributes (data)

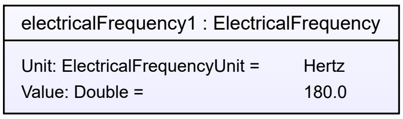
Name	Multiplicity	Type
Unit	1	*ElectricalFrequencyUnit*
Value	1	*Double*

Implementation in Proteus Schema

All data attributes with type *NullableElectricalFrequency* (the base type of *ElectricalFrequency*) are implemented as *DEXPI generic attributes for physical quantities*.

Example

The instance electricalFrequency1 represents an *ElectricalFrequency* of 180.0 Hz.



electricalFrequency1 : ElectricalFrequency	
Unit: ElectricalFrequencyUnit =	Hertz
Value: Double =	180.0

Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double"
  Value="180.0"
  Units="Hertz"
  UnitsURI="http://data.posccaesar.org/rdl/RDS1326464" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance electricalFrequency1. For a complete example, see the attribute *AlternatingCurrentFrequency* of the DEXPI class *AlternatingCurrentGenerator*.

12.4.2 Unit

Attribute (data)

The unit of measurement of the *ElectricalFrequency*.

Multiplicity: 1

Type: *ElectricalFrequencyUnit*

Implementation in Proteus Schema

See implementation of *ElectricalFrequency*.

Example

See example for *ElectricalFrequency*.

12.4.3 Value

Attribute (data)

The numerical value of the *ElectricalFrequency*.

Multiplicity: 1

Type: *Double*

Implementation in Proteus Schema

See implementation of *ElectricalFrequency*.

Example

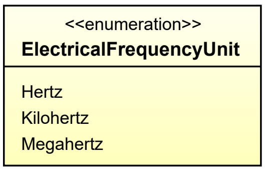
See example for *ElectricalFrequency*.

12.5. ElectricalFrequencyUnit

12.5.1 Overview

Enumeration

A unit of measurement for a physical quantity of application type *NullableElectricalFrequency* with dimension T^{-1}.



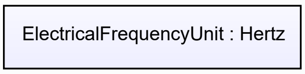
Literals

Name	Symbol	UN Code	RDL Reference
Hertz	Hz	HTZ	HERTZ http://data.posccaesar.org/rdl/RDS1326464
Kilohertz	kHz	KHZ	KILOHERTZ http://data.posccaesar.org/rdl/RDS4316756612
Megahertz	MHz	MHZ	MEGAHERTZ http://data.posccaesar.org/rdl/RDS4316806716

Implementation in Proteus Schema

ElectricalFrequencyUnit is only used as the type of the *Unit* attribute of *ElectricalFrequency*. *ElectricalFrequency* is implemented using *DEXPI generic attributes for physical quantities*. In a `<GenericAttribute>` element, the *ElectricalFrequencyUnit* literal is given by means of its RDL reference in the table above. The `Units` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `UnitsURI` attribute of the element is the URI of the RDL reference.

Example



Example: Implementation in Proteus Schema

```
<GenericAttribute
  Units="Hertz"
  UnitsURI="http://data.posccaesar.org/rdl/RDS1326464" ...>
```

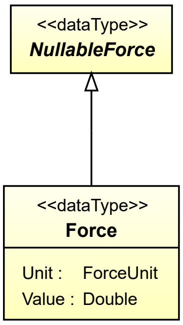
Note that the `<GenericAttribute>` element must have further attributes (`Format`, `Value`, `Name`, and `AttributeURI`). See the implementation examples for *ElectricalFrequency* and for the *AlternatingCurrentFrequency* attribute of *AlternatingCurrentGenerator*.

12.6. Force

12.6.1 Overview

Data type

An *actual value* for a physical quantity of type *NullableForce*, i.e., a physical quantity that has a numerical value and a unit of measurement.



Supertypes

- *NullableForce*

Attributes (data)

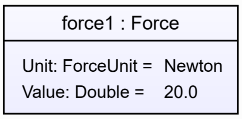
Name	Multiplicity	Type
Unit	1	*ForceUnit*
Value	1	*Double*

Implementation in Proteus Schema

All data attributes with type *NullableForce* (the base type of *Force*) are implemented as *DEXPI generic attributes for physical quantities*.

Example

The instance force1 represents a *Force* of 20.0 N.



Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double"
  Value="20.0"
  Units="Newton"
  UnitsURI="http://data.posccaesar.org/rdl/RDS1337939" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance force1. For a complete example, see the attribute *LowerLimitDesignPressingForce* of the DEXPI class *ReciprocatingPressureAgglomerator*.

12.6.2 Unit

Attribute (data)

The unit of measurement of the *Force*.

Multiplicity: 1

Type: *ForceUnit*

Implementation in Proteus Schema

See implementation of *Force*.

Example

See example for *Force*.

12.6.3 Value

Attribute (data)

The numerical value of the *Force*.

Multiplicity: 1

Type: *Double*

Implementation in Proteus Schema

See implementation of *Force*.

Example

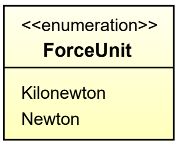
See example for *Force*.

12.7. ForceUnit

12.7.1 Overview

Enumeration

A unit of measurement for a physical quantity of type *NullableForce* with *dimension* LMT^{-2}.



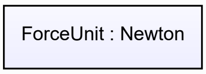
Literals

Name	Symbol	UN Code	RDL Reference
Kilonewton	kN	B47	KILONEWTON http://data.posccaesar.org/rdl/RDS1351034
Newton	N	NEW	NEWTON http://data.posccaesar.org/rdl/RDS1337939

Implementation in Proteus Schema

ForceUnit is only used as the type of the *Unit* attribute of *Force*. *Force* is implemented using *DEXPI generic attributes for physical quantities*. In a `<GenericAttribute>` element, the *ForceUnit* literal is given by means of its RDL reference in the table above. The `Units` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `UnitsURI` attribute of the element is the URI of the RDL reference.

Example



Example: Implementation in Proteus Schema

```
<GenericAttribute
  Units="Newton"
  UnitsURI="http://data.posccaesar.org/rdl/RDS1337939" ...>
```

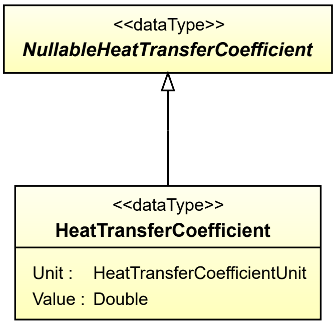
Note that the `<GenericAttribute>` element must have further attributes (`Format`, `Value`, `Name`, and `AttributeURI`). See the implementation examples for *Force* and for the *LowerLimitDesignPressingForce* attribute of *ReciprocatingPressureAgglomerator*.

12.8. HeatTransferCoefficient

12.8.1 Overview

Data type

An *actual value* for a physical quantity of type *NullableHeatTransferCoefficient*, i.e., a physical quantity that has a numerical value and a unit of measurement.


<<dataType>>
NullableHeatTransferCoefficient
<<dataType>>
HeatTransferCoefficient
Unit : HeatTransferCoefficientUnit
Value : Double

Supertypes

- *NullableHeatTransferCoefficient*

Attributes (data)

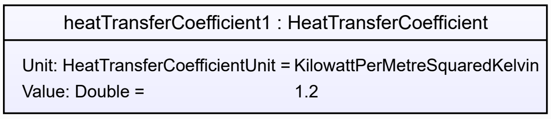
Name	Multiplicity	Type
Unit	1	*HeatTransferCoefficientUnit*
Value	1	*Double*

Implementation in Proteus Schema

All data attributes with type *NullableHeatTransferCoefficient* (the base type of *HeatTransferCoefficient*) are implemented as *DEXPI generic attributes for physical quantities*.

Example

The instance heatTransferCoefficient1 represents a *HeatTransferCoefficient* of 1.2 kW/(m^2 · K).



heatTransferCoefficient1 : HeatTransferCoefficient

Unit: HeatTransferCoefficientUnit = KilowattPerMetreSquaredKelvin
Value: Double = 1.2

Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double"
  Value="1.2"
  Units="KilowattPerMetreSquaredKelvin"
  UnitsURI="http://data.posccaesar.org/rdl/RDS43167567170" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance heatTransferCoefficient1. For a complete example, see the attribute *DesignHeatTransferCoefficient* of the DEXPI class *HeatExchanger*.

12.8.2 Unit

Attribute (data)

The unit of measurement of the *HeatTransferCoefficient*.

Multiplicity: 1

Type: *HeatTransferCoefficientUnit*

Implementation in Proteus Schema

See implementation of *HeatTransferCoefficient*.

Example

See example for *HeatTransferCoefficient*.

12.8.3 Value

Attribute (data)

The numerical value of the *HeatTransferCoefficient*.

Multiplicity: 1

Type: *Double*

Implementation in Proteus Schema

See implementation of *HeatTransferCoefficient*.

Example

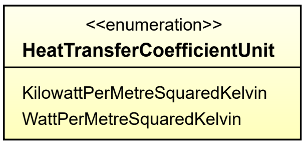
See example for *HeatTransferCoefficient*.

12.9. HeatTransferCoefficientUnit

12.9.1 Overview

Enumeration

A unit of measurement for a physical quantity of type *NullableHeatTransferCoefficient* with *dimension* MT$^{-3}\Theta$.


<<enumeration>>
HeatTransferCoefficientUnit
KilowattPerMetreSquaredKelvin
WattPerMetreSquaredKelvin

Literals

Name	Symbol	UN Code	RDL Reference
KilowattPerMetreSquaredKelvin	$\mathrm{kW}/(\mathrm{m}^2 \cdot \mathrm{K})$	N78	KILOWATT PER METRE SQUARED KELVIN http://data.posccaesar.org/rdl/RDS43167567170
WattPerMetreSquaredKelvin	$\mathrm{W}/(\mathrm{m}^2 \cdot \mathrm{K})$	D55	WATT PER METRE SQUARED KELVIN http://data.posccaesar.org/rdl/RDS1348424

Implementation in Proteus Schema

HeatTransferCoefficientUnit is only used as the type of the *Unit* attribute of *HeatTransferCoefficient*. *HeatTransferCoefficient* is implemented using *DEXPI generic attributes for physical quantities*. In a `<GenericAttribute>` element, the *HeatTransferCoefficientUnit* literal is given by means of its RDL reference in the table above. The `Units` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `UnitsURI` attribute of the element is the URI of the RDL reference.

Example

HeatTransferCoefficientUnit : KilowattPerMetreSquaredKelvin

Example: Implementation in Proteus Schema

```
<GenericAttribute
  Units="KilowattPerMetreSquaredKelvin"
  UnitsURI="http://data.posccaesar.org/rdl/RDS43167567170" ...>
```

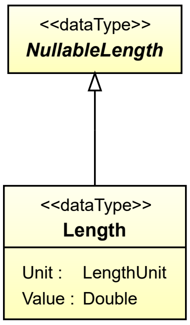
Note that the `<GenericAttribute>` element must have further attributes (`Format`, `Value`, `Name`, and `AttributeURI`). See the implementation examples for *HeatTransferCoefficient* and for the *DesignHeatTransferCoefficient* attribute of *HeatExchanger*.

12.10. Length

12.10.1 Overview

Data type

An *actual value* for a physical quantity of type *NullableLength*, i.e., a physical quantity that has a numerical value and a unit of measurement.



Supertypes

- *NullableLength*

Attributes (data)

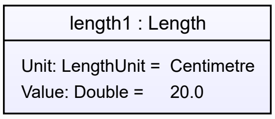
Name	Multiplicity	Type
Unit	1	*LengthUnit*
Value	1	*Double*

Implementation in Proteus Schema

All data attributes with type *NullableLength* (the base type of *Length*) are implemented as *DEXPI generic attributes for physical quantities*.

Example

The instance length1 represents a *Length* of 20.0 cm.



length1 : Length

Unit: LengthUnit = Centimetre
Value: Double = 20.0

Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double"
  Value="20.0"
  Units="Centimetre"
  UnitsURI="http://data.posccaesar.org/rdl/RDS1318004" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance length1. For a complete example, see the attribute *Diameter* of the DEXPI class *AgitatorRotor*.

12.10.2 Unit

Attribute (data)

The unit of measurement of the *Length*.

Multiplicity: 1

Type: *LengthUnit*

Implementation in Proteus Schema

See implementation of *Length*.

Example

See example for *Length*.

12.10.3 Value

Attribute (data)

The numerical value of the *Length*.

Multiplicity: 1

Type: *Double*

Implementation in Proteus Schema

See implementation of *Length*.

Example

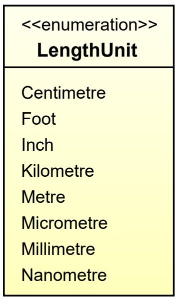
See example for *Length*.

12.11. LengthUnit

12.11.1 Overview

Enumeration

A unit of measurement for a physical quantity of type *NullableLength* with *dimension* L.



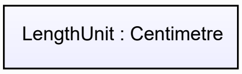
Literals

Name	Symbol	UN Code	RDL Reference
Centimetre	cm	CMT	CENTIMETRE http://data.posccaesar.org/rdl/RDS1318004
Foot	ft	FOT	FOOT http://data.posccaesar.org/rdl/RDS1324664
Inch	in	INH	INCH http://data.posccaesar.org/rdl/RDS1326959
Kilometre	km	KMT	KILOMETRE http://data.posccaesar.org/rdl/RDS1330199
Metre	m	MTR	METRE http://data.posccaesar.org/rdl/RDS1332674
Micrometre	µm	4H	MICROMETRE http://data.posccaesar.org/rdl/RDS1351529
Millimetre	mm	MMT	MILLIMETRE http://data.posccaesar.org/rdl/RDS1357739
Nanometre	nm	C45	NANOMETRE http://data.posccaesar.org/rdl/RDS1337669

Implementation in Proteus Schema

LengthUnit is only used as the type of the *Unit* attribute of *Length*. *Length* is implemented using *DEXPI generic attributes for physical quantities*. In a `<GenericAttribute>` element, the *LengthUnit* literal is given by means of its RDL reference in the table above. The `Units` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `UnitsURI` attribute of the element is the URI of the RDL reference.

Example



Example: Implementation in Proteus Schema

```
<GenericAttribute
  Units="Centimetre"
  UnitsURI="http://data.posccaesar.org/rdl/RDS1318004" ...>
```

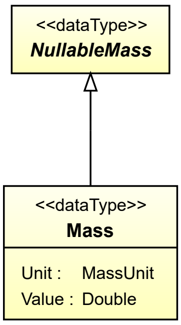
Note that the `<GenericAttribute>` element must have further attributes (`Format`, `Value`, `Name`, and `AttributeURI`). See the implementation examples for *Length* and for the *Diameter* attribute of *AgitatorRotor*.

12.12. Mass

12.12.1 Overview

Data type

An *actual value* for a physical quantity of type *NullableMass*, i.e., a physical quantity that has a numerical value and a unit of measurement.



Supertypes

- *NullableMass*

Attributes (data)

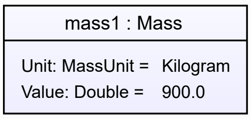
Name	Multiplicity	Type
Unit	1	*MassUnit*
Value	1	*Double*

Implementation in Proteus Schema

All data attributes with type *NullableMass* (the base type of *Mass*) are implemented as *DEXPI generic attributes for physical quantities*.

Example

The instance mass1 represents a *Mass* of 900.0 kg.



Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double"
  Value="900.0"
  Units="Kilogram"
  UnitsURI="http://data.posccaesar.org/rdl/RDS1328669" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance mass1. For a complete example, see the attribute *UpperLimitDesignLoad* of the DEXPI class *BatchWeigher*.

12.12.2 Unit

Attribute (data)

The unit of measurement of the *Mass*.

Multiplicity: 1

Type: *MassUnit*

Implementation in Proteus Schema

See implementation of *Mass*.

Example

See example for *Mass*.

12.12.3 Value

Attribute (data)

The numerical value of the *Mass*.

Multiplicity: 1

Type: *Double*

Implementation in Proteus Schema

See implementation of *Mass*.

Example

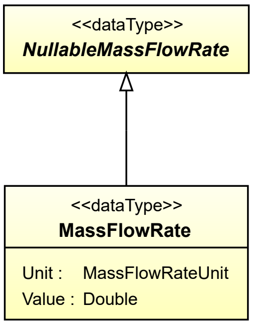
See example for *Mass*.

12.13. MassFlowRate

12.13.1 Overview

Data type

An *actual value* for a physical quantity of type *NullableMassFlowRate*, i.e., a physical quantity that has a numerical value and a unit of measurement.



Supertypes

- *NullableMassFlowRate*

Attributes (data)

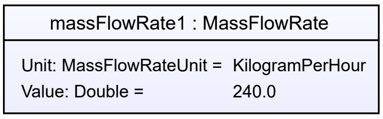
Name	Multiplicity	Type
Unit	1	*MassFlowRateUnit*
Value	1	*Double*

Implementation in Proteus Schema

All data attributes with type *NullableMassFlowRate* (the base type of *MassFlowRate*) are implemented as *DEXPI generic attributes for physical quantities*.

Example

The instance massFlowRate1 represents a *MassFlowRate* of 240.0 kg/h.


massFlowRate1 : MassFlowRate
Unit: MassFlowRateUnit = KilogramPerHour
Value: Double = 240.0

Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double"
  Value="240.0"
  Units="KilogramPerHour"
  UnitsURI="http://data.posccaesar.org/rdl/RDS1329344" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance massFlowRate1. For a complete example, see the attribute *DesignLiquidFeedMassFlowRate* of the DEXPI class *Agglomerator*.

12.13.2 Unit

Attribute (data)

The unit of measurement of the *MassFlowRate*.

Multiplicity: 1

Type: *MassFlowRateUnit*

Implementation in Proteus Schema

See implementation of *MassFlowRate*.

Example

See example for *MassFlowRate*.

12.13.3 Value

Attribute (data)

The numerical value of the *MassFlowRate*.

Multiplicity: 1

Type: *Double*

Implementation in Proteus Schema

See implementation of *MassFlowRate*.

Example

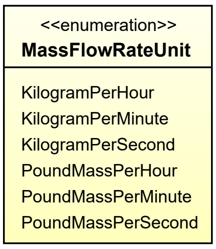
See example for *MassFlowRate*.

12.14. MassFlowRateUnit

12.14.1 Overview

Enumeration

A unit of measurement for a physical quantity of type *NullableMassFlowRate* with *dimension* MT^{-1}.



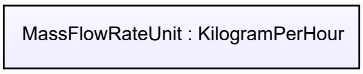
Literals

Name	Symbol	UN Code	RDL Reference
KilogramPerHour	kg/h	E93	KILOGRAM PER HOUR http://data.posccaesar.org/rdl/RDS1329344
KilogramPerMinute	kg/min	F31	KILOGRAM PER MINUTE http://data.posccaesar.org/rdl/RDS1350719
KilogramPerSecond	kg/s	KGS	KILOGRAM PER SECOND http://data.posccaesar.org/rdl/RDS1329659
PoundMassPerHour	lb/h	4U	POUND MASS PER HOUR http://data.posccaesar.org/rdl/RDS43168250123
PoundMassPerMinute	lb/min	K78	POUND MASS PER MINUTE http://data.posccaesar.org/rdl/RDS43168250156
PoundMassPerSecond	lb/s	K81	POUND MASS PER SECOND http://data.posccaesar.org/rdl/RDS43168250167

Implementation in Proteus Schema

MassFlowRateUnit is only used as the type of the *Unit* attribute of *MassFlowRate*. *MassFlowRate* is implemented using *DEXPI generic attributes for physical quantities*. In a `<GenericAttribute>` element, the *MassFlowRateUnit* literal is given by means of its RDL reference in the table above. The `Units` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `UnitsURI` attribute of the element is the URI of the RDL reference.

Example



Example: Implementation in Proteus Schema

```
<GenericAttribute
  Units="KilogramPerHour"
  UnitsURI="http://data.posccaesar.org/rdl/RDS1329344" ...>
```

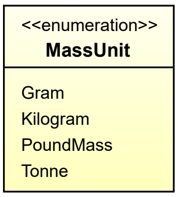
Note that the `<GenericAttribute>` element must have further attributes (`Format`, `Value`, `Name`, and `AttributeURI`). See the implementation examples for *MassFlowRate* and for the *DesignLiquidFeedMassFlowRate* attribute of *Agglomerator*.

12.15. MassUnit

12.15.1 Overview

Enumeration

A unit of measurement for a physical quantity of type *NullableMass* with *dimension* M.



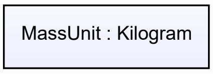
Literals

Name	Symbol	UN Code	RDL Reference
Gram	g	GRM	GRAM http://data.posccaesar.org/rdl/RDS1325789
Kilogram	kg	KGM	KILOGRAM http://data.posccaesar.org/rdl/RDS1328669
PoundMass	lb	LBR	POUND MASS http://data.posccaesar.org/rdl/RDS11617515
Tonne	t	TNE	TONNE http://data.posccaesar.org/rdl/RDS1344689

Implementation in Proteus Schema

MassUnit is only used as the type of the *Unit* attribute of *Mass*. *Mass* is implemented using *DEXPI generic attributes for physical quantities*. In a `<GenericAttribute>` element, the *MassUnit* literal is given by means of its RDL reference in the table above. The `Units` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `UnitsURI` attribute of the element is the URI of the RDL reference.

Example



Example: Implementation in Proteus Schema

```
<GenericAttribute
  Units="Kilogram"
  UnitsURI="http://data.posccaesar.org/rdl/RDS1328669" ...>
```

Note that the `<GenericAttribute>` element must have further attributes (`Format`, `Value`, `Name`, and `AttributeURI`). See the implementation examples for *Mass* and for the *UpperLimitDesignLoad* attribute of *BatchWeigher*.

12.16. NULL_AREA

12.16.1 Overview

Instance

The only instance of the singleton type *NullArea*. This instance represents the *null value* of the physical quantity type *NullableArea*.

Type

- *NullArea*

Implementation in Proteus Schema

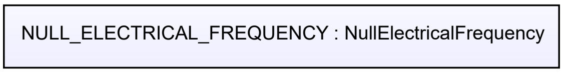
See implementation of *NullArea*.

12.17. NULL_ELECTRICAL_FREQUENCY

12.17.1 Overview

Instance

The only instance of the singleton type *NullElectricalFrequency*. This instance represents the *null value* of the physical quantity type *NullableElectricalFrequency*.



Type

- *NullElectricalFrequency*

Implementation in Proteus Schema

See implementation of *NullElectricalFrequency*.

12.18. NULL_FORCE

12.18.1 Overview

Instance

The only instance of the singleton type *NullForce*. This instance represents the *null value* of the physical quantity type *NullableForce*.

Type

- *NullForce*

Implementation in Proteus Schema

See implementation of *NullForce*.

12.19. NULL_HEAT_TRANSFER_COEFFICIENT

12.19.1 Overview

Instance

The only instance of the singleton type *NullHeatTransferCoefficient*. This instance represents the *null value* of the physical quantity type *NullableHeatTransferCoefficient*.

NULL_HEAT_TRANSFER_COEFFICIENT : NullHeatTransferCoefficient

Type

- *NullHeatTransferCoefficient*

Implementation in Proteus Schema

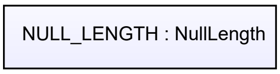
See implementation of *NullHeatTransferCoefficient*.

12.20. NULL_LENGTH

12.20.1 Overview

Instance

The only instance of the singleton type *NullLength*. This instance represents the *null value* of the physical quantity type *NullableLength*.



Type

- *NullLength*

Implementation in Proteus Schema

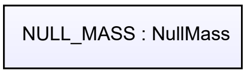
See implementation of *NullLength*.

12.21. NULL_MASS

12.21.1 Overview

Instance

The only instance of the singleton type *NullMass*. This instance represents the *null value* of the physical quantity type *NullableMass*.



Type

- *NullMass*

Implementation in Proteus Schema

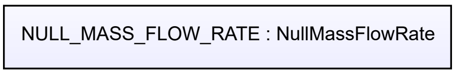
See implementation of *NullMass*.

12.22. NULL_MASS_FLOW_RATE

12.22.1 Overview

Instance

The only instance of the singleton type *NullMassFlowRate*. This instance represents the *null value* of the physical quantity type *NullableMassFlowRate*.



Type

- *NullMassFlowRate*

Implementation in Proteus Schema

See implementation of *NullMassFlowRate*.

12.23. NULL_NUMBER_PER_TIME_INTERVAL

12.23.1 Overview

Instance

The only instance of the singleton type *NullNumberPerTimeInterval*. This instance represents the *null value* of the physical quantity type *NullableNumberPerTimeInterval*.

Type

- *NullNumberPerTimeInterval*

> **Implementation in Proteus Schema**
>
> See implementation of *NullNumberPerTimeInterval*.

12.24. NULL_PERCENTAGE

12.24.1 Overview

Instance

The only instance of the singleton type *NullPercentage*. This instance represents the *null value* of the physical quantity type *NullablePercentage*.

Type

- *NullPercentage*

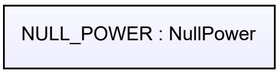
> **Implementation in Proteus Schema**
>
> See implementation of *NullPercentage*.

12.25. NULL_POWER

12.25.1 Overview

Instance

The only instance of the singleton type *NullPower*. This instance represents the *null value* of the physical quantity type *NullablePower*.



Type

- *NullPower*

> **Implementation in Proteus Schema**
>
> See implementation of *NullPower*.

12.26. NULL_PRESSURE_ABSOLUTE

12.26.1 Overview

Instance

The only instance of the singleton type *NullPressureAbsolute*. This instance represents the *null value* of the physical quantity type *NullablePressureAbsolute*.

Type

- *NullPressureAbsolute*

Implementation in Proteus Schema

See implementation of *NullPressureAbsolute*.

12.27. NULL_PRESSURE_GAUGE

12.27.1 Overview

Instance

The only instance of the singleton type *NullPressureGauge*. This instance represents the *null value* of the physical quantity type *NullablePressureGauge*.

Type

- *NullPressureGauge*

Implementation in Proteus Schema

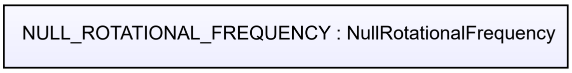
See implementation of *NullPressureGauge*.

12.28. NULL_ROTATIONAL_FREQUENCY

12.28.1 Overview

Instance

The only instance of the singleton type *NullRotationalFrequency*. This instance represents the *null value* of the physical quantity type *NullableRotationalFrequency*.



Type

- *NullRotationalFrequency*

Implementation in Proteus Schema

See implementation of *NullRotationalFrequency*.

12.29. NULL_TEMPERATURE

12.29.1 Overview

Instance

The only instance of the singleton type *NullTemperature*. This instance represents the *null value* of the physical quantity type *NullableTemperature*.

Type

- *NullTemperature*

Implementation in Proteus Schema

See implementation of *NullTemperature*.

12.30. NULL_VOLTAGE

12.30.1 Overview

Instance

The only instance of the singleton type *NullVoltage*. This instance represents the *null value* of the physical quantity type *NullableVoltage*.

Type

- *NullVoltage*

Implementation in Proteus Schema

See implementation of *NullVoltage*.

12.31. NULL_VOLUME

12.31.1 Overview

Instance

The only instance of the singleton type *NullVolume*. This instance represents the *null value* of the physical quantity type *NullableVolume*.

Type

- *NullVolume*

Implementation in Proteus Schema

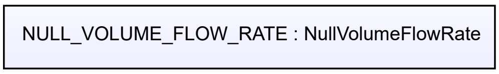
See implementation of *NullVolume*.

12.32. NULL_VOLUME_FLOW_RATE

12.32.1 Overview

Instance

The only instance of the singleton type *NullVolumeFlowRate*. This instance represents the *null value* of the physical quantity type *NullableVolumeFlowRate*.



Type

- *NullVolumeFlowRate*

Implementation in Proteus Schema

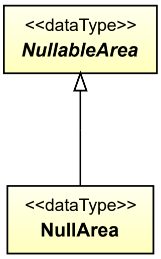
See implementation of *NullVolumeFlowRate*.

12.33. NullArea

12.33.1 Overview

Data type

A *null value* for a physical quantity of type *NullableArea*. The only instance of this singleton type is *NULL_AREA*.



Supertypes

- *NullableArea*

Implementation in Proteus Schema

All data attributes with type *NullableArea* (the base type of *NullArea*) are implemented as *DEXPI generic attributes for physical quantities*.

Example

NULL_AREA : NullArea

Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double" ...>
```

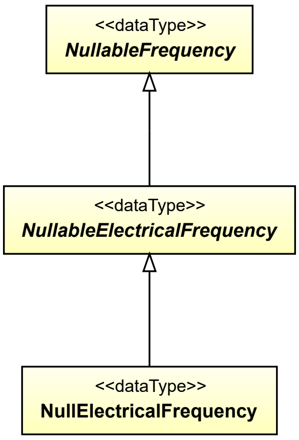
Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the null instance *NULL_AREA* (e.g., see the implementation example for the attribute *FilterArea* of the DEXPI class *FilterUnit*). For a null value, the attributes `Units`, `UnitsURI`, and `Value` *must not be used*.

12.34. NullElectricalFrequency

12.34.1 Overview

Data type

A *null value* for a physical quantity of application type *NullableElectricalFrequency*. The only instance of this singleton type is *NULL_ELECTRICAL_FREQUENCY*.



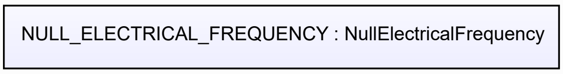
Supertypes

- *NullableElectricalFrequency*

Implementation in Proteus Schema

All data attributes with type *NullableElectricalFrequency* (the base type of *NullElectricalFrequency*) are implemented as *DEXPI generic attributes for physical quantities*.

Example



Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double" ...>
```

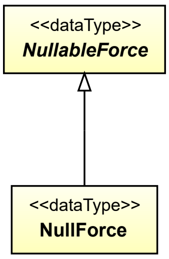
Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the null instance *NULL_ELECTRICAL_FREQUENCY* (e.g., see the implementation example for the attribute *AlternatingCurrentFrequency* of the DEXPI class *AlternatingCurrentGenerator*). For a null value, the attributes `Units`, `UnitsURI`, and `Value` *must not be used*.

12.35. NullForce

12.35.1 Overview

Data type

A *null value* for a physical quantity of type *NullableForce*. The only instance of this singleton type is *NULL_FORCE*.



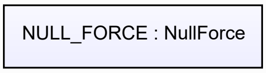
Supertypes

- *NullableForce*

Implementation in Proteus Schema

All data attributes with type *NullableForce* (the base type of *NullForce*) are implemented as *DEXPI generic attributes for physical quantities*.

Example



Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double" ...>
```

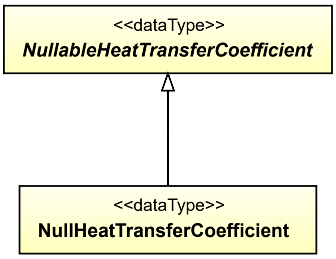
Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the null instance *NULL_FORCE* (e.g., see the implementation example for the attribute *LowerLimitDesignPressingForce* of the DEXPI class *ReciprocatingPressureAgglomerator*). For a null value, the attributes `Units`, `UnitsURI`, and `Value` *must not be used*.

12.36. NullHeatTransferCoefficient

12.36.1 Overview

Data type

A *null value* for a physical quantity of type *NullableHeatTransferCoefficient*. The only instance of this singleton type is *NULL_HEAT_TRANSFER_COEFFICIENT*.



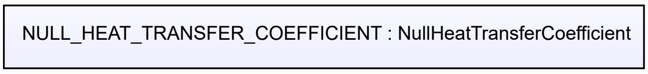
Supertypes

- *NullableHeatTransferCoefficient*

Implementation in Proteus Schema

All data attributes with type *NullableHeatTransferCoefficient* (the base type of *NullHeatTransferCoefficient*) are implemented as *DEXPI generic attributes for physical quantities*.

Example


NULL_HEAT_TRANSFER_COEFFICIENT : NullHeatTransferCoefficient

Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double" ...>
```

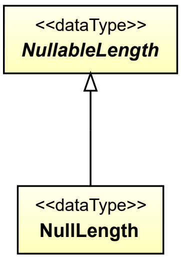
Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the null instance *NULL_HEAT_TRANSFER_COEFFICIENT* (e.g., see the implementation example for the attribute *DesignHeatTransferCoefficient* of the DEXPI class *HeatExchanger*). For a null value, the attributes `Units`, `UnitsURI`, and `Value` *must not be used*.

12.37. NullLength

12.37.1 Overview

Data type

A *null value* for a physical quantity of type *NullableLength*. The only instance of this singleton type is *NULL_LENGTH*.



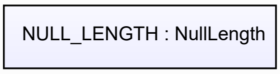
Supertypes

- *NullableLength*

Implementation in Proteus Schema

All data attributes with type *NullableLength* (the base type of *NullLength*) are implemented as *DEXPI generic attributes for physical quantities*.

Example



Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double" ...>
```

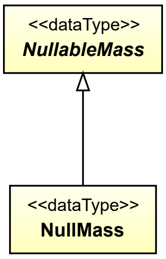
Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the null instance *NULL_LENGTH* (e.g., see the implementation example for the attribute *Diameter* of the DEXPI class *AgitatorRotor*). For a null value, the attributes `Units`, `UnitsURI`, and `Value` *must not be used*.

12.38. NullMass

12.38.1 Overview

Data type

A *null value* for a physical quantity of type *NullableMass*. The only instance of this singleton type is *NULL_MASS*.



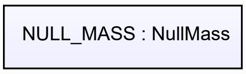
Supertypes

- *NullableMass*

Implementation in Proteus Schema

All data attributes with type *NullableMass* (the base type of *NullMass*) are implemented as *DEXPI generic attributes for physical quantities*.

Example



Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double" ...>
```

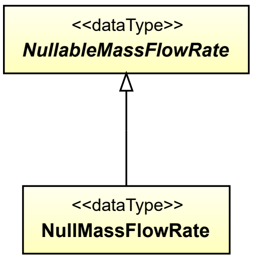
Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the null instance *NULL_MASS* (e.g., see the implementation example for the attribute *UpperLimitDesignLoad* of the DEXPI class *BatchWeigher*). For a null value, the attributes `Units`, `UnitsURI`, and `Value` *must not be used*.

12.39. NullMassFlowRate

12.39.1 Overview

Data type

A *null value* for a physical quantity of type *NullableMassFlowRate*. The only instance of this singleton type is *NULL_MASS_FLOW_RATE*.



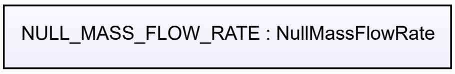
Supertypes

- *NullableMassFlowRate*

Implementation in Proteus Schema

All data attributes with type *NullableMassFlowRate* (the base type of *NullMassFlowRate*) are implemented as *DEXPI generic attributes for physical quantities*.

Example



Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double" ...>
```

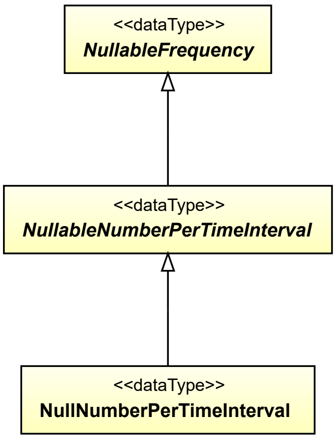
Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the null instance *NULL_MASS_FLOW_RATE* (e.g., see the implementation example for the attribute *DesignLiquidFeedMassFlowRate* of the DEXPI class *Agglomerator*). For a null value, the attributes `Units`, `UnitsURI`, and `Value` *must not be used*.

12.40. NullNumberPerTimeInterval

12.40.1 Overview

Data type

A *null value* for a physical quantity of application type *NullableNumberPerTimeInterval*. The only instance of this singleton type is *NULL_NUMBER_PER_TIME_INTERVAL*.



Supertypes

- *NullableNumberPerTimeInterval*

Implementation in Proteus Schema

All data attributes with type *NullableNumberPerTimeInterval* (the base type of *NullNumberPerTimeInterval*) are implemented as *DEXPI generic attributes for physical quantities*.

Example

NULL_NUMBER_PER_TIME_INTERVAL : NullNumberPerTimeInterval

Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double" ...>
```

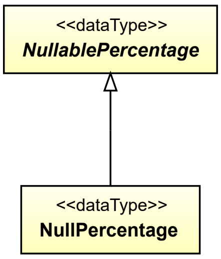
Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the null instance *NULL_NUMBER_PER_TIME_INTERVAL* (e.g., see the implementation example for the attribute *DesignCapacityWeighingQuantities* of the DEXPI class *BatchWeigher*). For a null value, the attributes `Units`, `UnitsURI`, and `Value` *must not be used*.

12.41. NullPercentage

12.41.1 Overview

Data type

A *null value* for a physical quantity of type *NullablePercentage*. The only instance of this singleton type is *NULL_PERCENTAGE*.



Supertypes

- *NullablePercentage*

Implementation in Proteus Schema

All data attributes with type *NullablePercentage* (the base type of *NullPercentage*) are implemented as *DEXPI generic attributes for physical quantities*.

Example

NULL_PERCENTAGE : NullPercentage

Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double" ...>
```

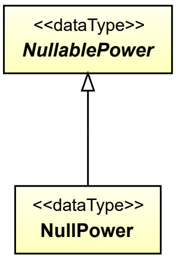
Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the null instance *NULL_PERCENTAGE* (e.g., see the implementation example for the attribute *Efficiency* of the DEXPI class *FilterUnit*). For a null value, the attributes `Units`, `UnitsURI`, and `Value` *must not be used*.

12.42. NullPower

12.42.1 Overview

Data type

A *null value* for a physical quantity of type *NullablePower*. The only instance of this singleton type is *NULL_POWER*.



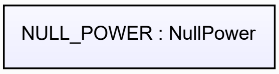
Supertypes

- *NullablePower*

Implementation in Proteus Schema

All data attributes with type *NullablePower* (the base type of *NullPower*) are implemented as *DEXPI generic attributes for physical quantities*.

Example



Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double" ...>
```

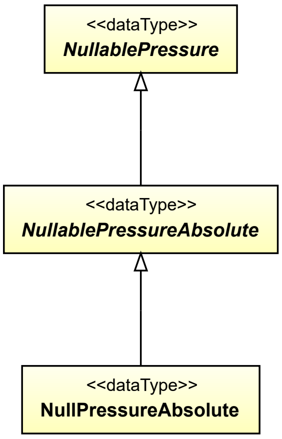
Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the null instance *NULL_POWER* (e.g., see the implementation example for the attribute *DesignShaftPower* of the DEXPI class *Agglomerator*). For a null value, the attributes `Units`, `UnitsURI`, and `Value` *must not be used*.

12.43. NullPressureAbsolute

12.43.1 Overview

Data type

A *null value* for a physical quantity of application type *NullablePressureAbsolute*. The only instance of this singleton type is *NULL_PRESSURE_ABSOLUTE*.



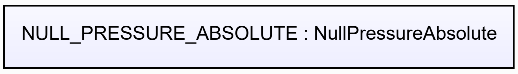
Supertypes

- *NullablePressureAbsolute*

Implementation in Proteus Schema

All data attributes with type *NullablePressureAbsolute* (the base type of *NullPressureAbsolute*) are implemented as *DEXPI generic attributes for physical quantities*.

Example



Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double" ...>
```

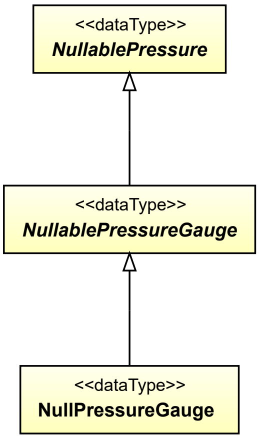
Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the null instance *NULL_PRESSURE_ABSOLUTE* (e.g., see the implementation example for the attribute *DesignDifferentialPressure* of the DEXPI class *Blower*). For a null value, the attributes `Units`, `UnitsURI`, and `Value` *must not be used*.

12.44. NullPressureGauge

12.44.1 Overview

Data type

A *null value* for a physical quantity of application type *NullablePressureGauge*. The only instance of this singleton type is *NULL_PRESSURE_GAUGE*.



Supertypes

- *NullablePressureGauge*

Implementation in Proteus Schema

All data attributes with type *NullablePressureGauge* (the base type of *NullPressureGauge*) are implemented as *DEXPI generic attributes for physical quantities*.

Example

Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double" ...>
```

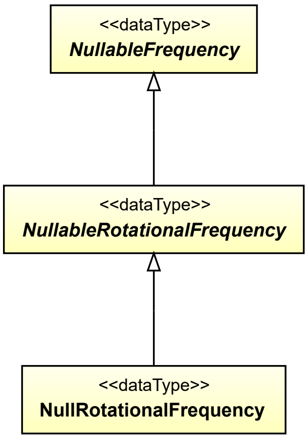
Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the null instance *NULL_PRESSURE_GAUGE* (e.g., see the implementation example for the attribute *LowerLimitDesignPressure* of the DEXPI class *Chamber*). For a null value, the attributes `Units`, `UnitsURI`, and `Value` *must not be used*.

12.45. NullRotationalFrequency

12.45.1 Overview

Data type

A *null value* for a physical quantity of application type *NullableRotationalFrequency*. The only instance of this singleton type is *NULL_ROTATIONAL_FREQUENCY*.



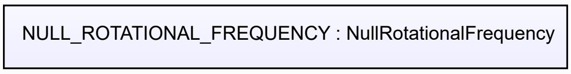
Supertypes

- *NullableRotationalFrequency*

Implementation in Proteus Schema

All data attributes with type *NullableRotationalFrequency* (the base type of *NullRotationalFrequency*) are implemented as *DEXPI generic attributes for physical quantities*.

Example



Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double" ...>
```

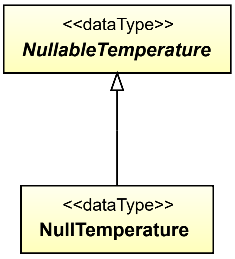
Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the null instance *NULL_ROTATIONAL_FREQUENCY* (e.g., see the implementation example for the attribute *DesignRotationalSpeed* of the DEXPI class *Agglomerator*). For a null value, the attributes `Units`, `UnitsURI`, and `Value` *must not be used.*

12.46. NullTemperature

12.46.1 Overview

Data type

A *null value* for a physical quantity of type *NullableTemperature*. The only instance of this singleton type is *NULL_TEMPERATURE*.



Supertypes

- *NullableTemperature*

Implementation in Proteus Schema

All data attributes with type *NullableTemperature* (the base type of *NullTemperature*) are implemented as *DEXPI generic attributes for physical quantities*.

Example

Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double" ...>
```

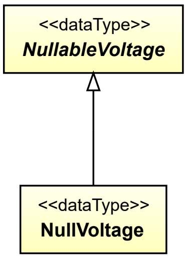
Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the null instance *NULL_TEMPERATURE* (e.g., see the implementation example for the attribute *LowerLimitDesignTemperature* of the DEXPI class *Chamber*). For a null value, the attributes `Units`, `UnitsURI`, and `Value` *must not be used*.

12.47. NullVoltage

12.47.1 Overview

Data type

A *null value* for a physical quantity of type *NullableVoltage*. The only instance of this singleton type is *NULL_VOLTAGE*.



Supertypes

- *NullableVoltage*

Implementation in Proteus Schema

All data attributes with type *NullableVoltage* (the base type of *NullVoltage*) are implemented as *DEXPI generic attributes for physical quantities*.

Example

Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double" ...>
```

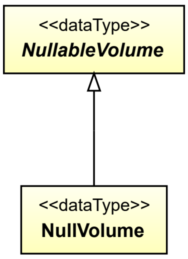
Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the null instance *NULL_VOLTAGE* (e.g., see the implementation example for the attribute *NominalVoltage* of the DEXPI class *AlternatingCurrentMotor*). For a null value, the attributes `Units`, `UnitsURI`, and `Value` *must not be used*.

12.48. NullVolume

12.48.1 Overview

Data type

A *null value* for a physical quantity of type *NullableVolume*. The only instance of this singleton type is *NULL_VOLUME*.



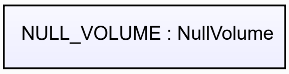
Supertypes

- *NullableVolume*

Implementation in Proteus Schema

All data attributes with type *NullableVolume* (the base type of *NullVolume*) are implemented as *DEXPI generic attributes for physical quantities*.

Example



Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double" ...>
```

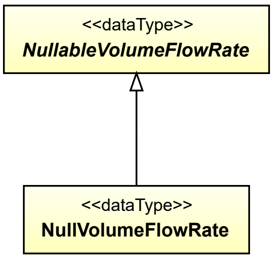
Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the null instance *NULL_VOLUME* (e.g., see the implementation example for the attribute *VolumePerStroke* of the DEXPI class *Displacer*). For a null value, the attributes `Units`, `UnitsURI`, and `Value` *must not be used*.

12.49. NullVolumeFlowRate

12.49.1 Overview

Data type

A *null value* for a physical quantity of type *NullableVolumeFlowRate*. The only instance of this singleton type is *NULL_VOLUME_FLOW_RATE*.



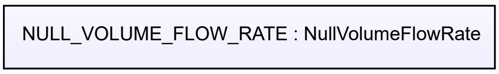
Supertypes

- *NullableVolumeFlowRate*

Implementation in Proteus Schema

All data attributes with type *NullableVolumeFlowRate* (the base type of *NullVolumeFlowRate*) are implemented as *DEXPI generic attributes for physical quantities*.

Example



Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the null instance *NULL_VOLUME_FLOW_RATE* (e.g., see the implementation example for the attribute *DesignVolumeFlowRate* of the DEXPI class *Agglomerator*). For a null value, the attributes `Units`, `UnitsURI`, and `Value` *must not be used*.

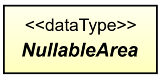
12.50. NullableArea

12.50.1 Overview

Abstract data type

NullableArea is a *simple physical quantity type* for the dimension L^2. *NullableArea* is abstract and has two concrete subtypes:

- an *Area* is an *actual value* for a physical quantity with a numerical value and a unit of measurement;
- a *NullArea* is a *null value* that explicitly indicates the absence of an actual physical quantity.



Subtypes

- *Area*
- *NullArea*

Implementation in Proteus Schema

All data attributes with type *NullableArea* are implemented as *DEXPI generic attributes for physical quantities*.

Example

See the examples for *Area* and *NullArea*.

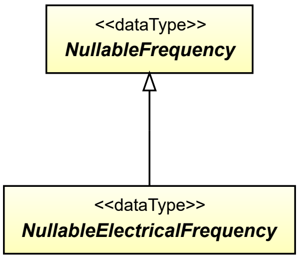
12.51. NullableElectricalFrequency

12.51.1 Overview

Abstract data type

A physical quantity of dimension T^{-1} (inherited from *NullableFrequency*) for the application type *electrical frequency*. *NullableElectricalFrequency* is abstract and has two concrete subtypes:

- an *ElectricalFrequency* is an *actual value* for a physical quantity with a numerical value and a unit of measurement;
- a *NullElectricalFrequency* is a *null value* that explicitly indicates the absence of an actual physical quantity.



Supertypes

- *NullableFrequency*

Subtypes

- *ElectricalFrequency*
- *NullElectricalFrequency*

Implementation in Proteus Schema

All data attributes with type *NullableElectricalFrequency* are implemented as *DEXPI generic attributes for physical quantities*.

Example

See the examples for *ElectricalFrequency* and *NullElectricalFrequency*.

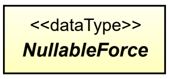
12.52. NullableForce

12.52.1 Overview

Abstract data type

NullableForce is a *simple physical quantity type* for the dimension LMT^{-2}. *NullableForce* is abstract and has two concrete subtypes:

- a *Force* is an *actual value* for a physical quantity with a numerical value and a unit of measurement;
- a *NullForce* is a *null value* that explicitly indicates the absence of an actual physical quantity.



Subtypes

- *Force*
- *NullForce*

Implementation in Proteus Schema

All data attributes with type *NullableForce* are implemented as *DEXPI generic attributes for physical quantities*.

Example

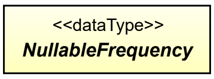
See the examples for *Force* and *NullForce*.

12.53. NullableFrequency

12.53.1 Overview

Abstract data type

NullableFrequency is an *application-dependent physical quantity type* for the dimension T^{-1}. It has 3 subtypes for different application areas.



Subtypes

- *NullableElectricalFrequency*
- *NullableNumberPerTimeInterval*
- *NullableRotationalFrequency*

Implementation in Proteus Schema

All data attributes with a type derived from *NullableFrequency* are implemented as *DEXPI generic attributes for physical quantities*.

Example

See the examples for *ElectricalFrequency*, *NullElectricalFrequency*, *NumberPerTimeInterval*, *NullNumberPerTimeInterval*, *RotationalFrequency*, and *NullRotationalFrequency*.

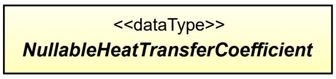
12.54. NullableHeatTransferCoefficient

12.54.1 Overview

Abstract data type

NullableHeatTransferCoefficient is a *simple physical quantity type* for the dimension $MT^{-3}\Theta$. *NullableHeatTransferCoefficient* is abstract and has two concrete subtypes:

- a *HeatTransferCoefficient* is an *actual value* for a physical quantity with a numerical value and a unit of measurement;
- a *NullHeatTransferCoefficient* is a *null value* that explicitly indicates the absence of an actual physical quantity.



Subtypes

- *HeatTransferCoefficient*
- *NullHeatTransferCoefficient*

Implementation in Proteus Schema

All data attributes with type *NullableHeatTransferCoefficient* are implemented as *DEXPI generic attributes for physical quantities*.

Example

See the examples for *HeatTransferCoefficient* and *NullHeatTransferCoefficient*.

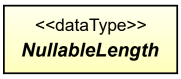
12.55. NullableLength

12.55.1 Overview

Abstract data type

NullableLength is a *simple physical quantity type* for the dimension L. *NullableLength* is abstract and has two concrete subtypes:

- a *Length* is an *actual value* for a physical quantity with a numerical value and a unit of measurement;
- a *NullLength* is a *null value* that explicitly indicates the absence of an actual physical quantity.



Subtypes

- *Length*
- *NullLength*

Implementation in Proteus Schema

All data attributes with type *NullableLength* are implemented as *DEXPI generic attributes for physical quantities*.

Example

See the examples for *Length* and *NullLength*.

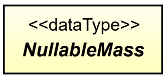
12.56. NullableMass

12.56.1 Overview

Abstract data type

NullableMass is a *simple physical quantity type* for the dimension M. *NullableMass* is abstract and has two concrete subtypes:

- a *Mass* is an *actual value* for a physical quantity with a numerical value and a unit of measurement;
- a *NullMass* is a *null value* that explicitly indicates the absence of an actual physical quantity.



Subtypes

- *Mass*
- *NullMass*

Implementation in Proteus Schema

All data attributes with type *NullableMass* are implemented as *DEXPI generic attributes for physical quantities*.

Example

See the examples for *Mass* and *NullMass*.

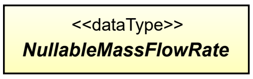
12.57. NullableMassFlowRate

12.57.1 Overview

Abstract data type

NullableMassFlowRate is a *simple physical quantity type* for the dimension MT^{-1}. *NullableMassFlowRate* is abstract and has two concrete subtypes:

- a *MassFlowRate* is an *actual value* for a physical quantity with a numerical value and a unit of measurement;
- a *NullMassFlowRate* is a *null value* that explicitly indicates the absence of an actual physical quantity.



Subtypes

- *MassFlowRate*
- *NullMassFlowRate*

Implementation in Proteus Schema

All data attributes with type *NullableMassFlowRate* are implemented as *DEXPI generic attributes for physical quantities*.

Example

See the examples for *MassFlowRate* and *NullMassFlowRate*.

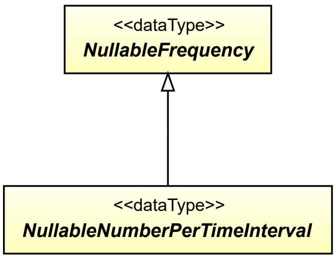
12.58. NullableNumberPerTimeInterval

12.58.1 Overview

Abstract data type

A physical quantity of dimension T^{-1} (inherited from *NullableFrequency*) for the application type *number per time interval*. *NullableNumberPerTimeInterval* is abstract and has two concrete subtypes:

- a *NumberPerTimeInterval* is an *actual value* for a physical quantity with a numerical value and a unit of measurement;
- a *NullNumberPerTimeInterval* is a *null value* that explicitly indicates the absence of an actual physical quantity.



Supertypes

- *NullableFrequency*

Subtypes

- *NullNumberPerTimeInterval*
- *NumberPerTimeInterval*

Implementation in Proteus Schema

All data attributes with type *NullableNumberPerTimeInterval* are implemented as *DEXPI generic attributes for physical quantities*.

Example

See the examples for *NumberPerTimeInterval* and *NullNumberPerTimeInterval*.

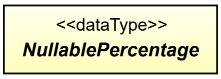
12.59. NullablePercentage

12.59.1 Overview

Abstract data type

A quantity given as a percentage. Although percentage is not a physical quantity type in the strict sense, it is modeled using the same pattern as for *simple physical quantity types*. *NullablePercentage* is abstract and has two concrete subtypes:

- a *Percentage* is an *actual value* for a quantity with a numerical value and a unit of measurement;
- a *NullPercentage* is a *null value* that explicitly indicates the absence of an actual quantity.



Subtypes

- *NullPercentage*
- *Percentage*

Implementation in Proteus Schema

All data attributes with type *NullablePercentage* are implemented as *DEXPI generic attributes for physical quantities*.

Example

See the examples for *Percentage* and *NullPercentage*.

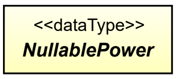
12.60. NullablePower

12.60.1 Overview

Abstract data type

NullablePower is a *simple physical quantity type* for the dimension L^2MT^{-3}. *NullablePower* is abstract and has two concrete subtypes:

- a *Power* is an *actual value* for a physical quantity with a numerical value and a unit of measurement;
- a *NullPower* is a *null value* that explicitly indicates the absence of an actual physical quantity.



Subtypes

- *NullPower*
- *Power*

Implementation in Proteus Schema

All data attributes with type *NullablePower* are implemented as *DEXPI generic attributes for physical quantities*.

Example

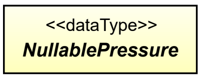
See the examples for *Power* and *NullPower*.

12.61. NullablePressure

12.61.1 Overview

Abstract data type

NullablePressure is an *application-dependent physical quantity type* for the dimension $L^{-1}MT^{-2}$. It has 2 subtypes for different application areas.



Subtypes

- *NullablePressureAbsolute*
- *NullablePressureGauge*

Implementation in Proteus Schema

All data attributes with a type derived from *NullablePressure* are implemented as *DEXPI generic attributes for physical quantities*.

Example

See the examples for *PressureAbsolute*, *NullPressureAbsolute*, *PressureGauge*, and *NullPressureGauge*.

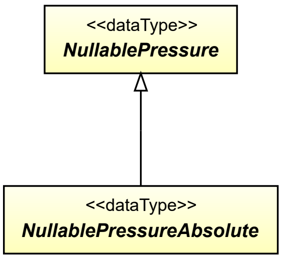
12.62. NullablePressureAbsolute

12.62.1 Overview

Abstract data type

A physical quantity of dimension $L^{-1}MT^{-2}$ (inherited from *NullablePressure*) for the application type *absolute pressure*. A *pressure absolute* is a pressure relative to a perfect vacuum. This data type is also used for the *difference between two pressures* other than atmospheric pressure. *NullablePressureAbsolute* is abstract and has two concrete subtypes:

- a *PressureAbsolute* is an *actual value* for a physical quantity with a numerical value and a unit of measurement;
- a *NullPressureAbsolute* is a *null value* that explicitly indicates the absence of an actual physical quantity.



Supertypes

- *NullablePressure*

Subtypes

- *NullPressureAbsolute*
- *PressureAbsolute*

Implementation in Proteus Schema

All data attributes with type *NullablePressureAbsolute* are implemented as *DEXPI generic attributes for physical quantities*.

Example

See the examples for *PressureAbsolute* and *NullPressureAbsolute*.

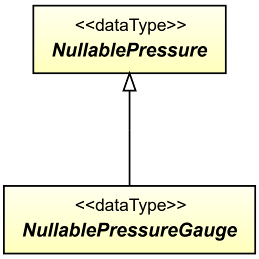
12.63. NullablePressureGauge

12.63.1 Overview

Abstract data type

A physical quantity of dimension $L^{-1}MT^{-2}$ (inherited from *NullablePressure*) for the application type *pressure gauge*. A *pressure gauge* is a pressure relative to atmospheric pressure. *NullablePressureGauge* is abstract and has two concrete subtypes:

- a *PressureGauge* is an *actual value* for a physical quantity with a numerical value and a unit of measurement;
- a *NullPressureGauge* is a *null value* that explicitly indicates the absence of an actual physical quantity.



Supertypes

- *NullablePressure*

Subtypes

- *NullPressureGauge*
- *PressureGauge*

Implementation in Proteus Schema

All data attributes with type *NullablePressureGauge* are implemented as *DEXPI generic attributes for physical quantities*.

Example

See the examples for *PressureGauge* and *NullPressureGauge*.

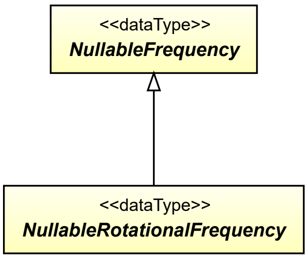
12.64. NullableRotationalFrequency

12.64.1 Overview

Abstract data type

A physical quantity of dimension T^{-1} (inherited from *NullableFrequency*) for the application type *rotational frequency*. *NullableRotationalFrequency* is abstract and has two concrete subtypes:

- a *RotationalFrequency* is an *actual value* for a physical quantity with a numerical value and a unit of measurement;
- a *NullRotationalFrequency* is a *null value* that explicitly indicates the absence of an actual physical quantity.



Supertypes

- *NullableFrequency*

Subtypes

- *NullRotationalFrequency*
- *RotationalFrequency*

Implementation in Proteus Schema

All data attributes with type *NullableRotationalFrequency* are implemented as *DEXPI generic attributes for physical quantities*.

Example

See the examples for *RotationalFrequency* and *NullRotationalFrequency*.

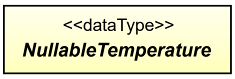
12.65. NullableTemperature

12.65.1 Overview

Abstract data type

NullableTemperature is a *simple physical quantity type* for the dimension Θ. *NullableTemperature* is abstract and has two concrete subtypes:

- a *Temperature* is an *actual value* for a physical quantity with a numerical value and a unit of measurement;
- a *NullTemperature* is a *null value* that explicitly indicates the absence of an actual physical quantity.



Subtypes

- *NullTemperature*
- *Temperature*

Implementation in Proteus Schema

All data attributes with type *NullableTemperature* are implemented as *DEXPI generic attributes for physical quantities*.

Example

See the examples for *Temperature* and *NullTemperature*.

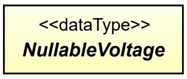
12.66. NullableVoltage

12.66.1 Overview

Abstract data type

NullableVoltage is a *simple physical quantity type* for the dimension $L^2MT^{-3}I^{-1}$. *NullableVoltage* is abstract and has two concrete subtypes:

- a *Voltage* is an *actual value* for a physical quantity with a numerical value and a unit of measurement;
- a *NullVoltage* is a *null value* that explicitly indicates the absence of an actual physical quantity.



Subtypes

- *NullVoltage*
- *Voltage*

Implementation in Proteus Schema

All data attributes with type *NullableVoltage* are implemented as *DEXPI generic attributes for physical quantities*.

Example

See the examples for *Voltage* and *NullVoltage*.

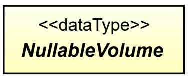
12.67. NullableVolume

12.67.1 Overview

Abstract data type

NullableVolume is a *simple physical quantity type* for the dimension L^3. *NullableVolume* is abstract and has two concrete subtypes:

- a *Volume* is an *actual value* for a physical quantity with a numerical value and a unit of measurement;
- a *NullVolume* is a *null value* that explicitly indicates the absence of an actual physical quantity.



Subtypes

- *NullVolume*
- *Volume*

Implementation in Proteus Schema

All data attributes with type *NullableVolume* are implemented as *DEXPI generic attributes for physical quantities*.

Example

See the examples for *Volume* and *NullVolume*.

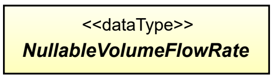
12.68. NullableVolumeFlowRate

12.68.1 Overview

Abstract data type

NullableVolumeFlowRate is a *simple physical quantity type* for the dimension L^3T^{-1}. *NullableVolumeFlowRate* is abstract and has two concrete subtypes:

- a *VolumeFlowRate* is an *actual value* for a physical quantity with a numerical value and a unit of measurement;
- a *NullVolumeFlowRate* is a *null value* that explicitly indicates the absence of an actual physical quantity.



Subtypes

- *NullVolumeFlowRate*
- *VolumeFlowRate*

Implementation in Proteus Schema

All data attributes with type *NullableVolumeFlowRate* are implemented as *DEXPI generic attributes for physical quantities*.

Example

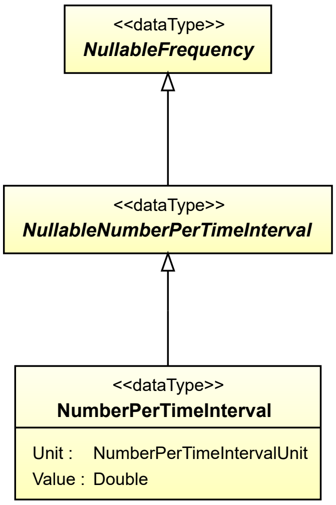
See the examples for *VolumeFlowRate* and *NullVolumeFlowRate*.

12.69. NumberPerTimeInterval

12.69.1 Overview

Data type

An *actual value* for a physical quantity of type *NullableNumberPerTimeInterval*, i.e., a physical quantity that has a numerical value and a unit of measurement.



Supertypes

- *NullableNumberPerTimeInterval*

Attributes (data)

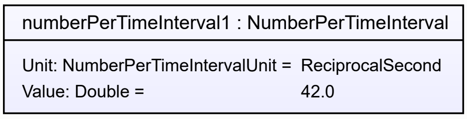
Name	Multiplicity	Type
Unit	1	*NumberPerTimeIntervalUnit*
Value	1	*Double*

Implementation in Proteus Schema

All data attributes with type *NullableNumberPerTimeInterval* (the base type of *NumberPerTimeInterval*) are implemented as *DEXPI generic attributes for physical quantities*.

Example

The instance numberPerTimeInterval1 represents a *NumberPerTimeInterval* of 42.0 s^{-1}.



numberPerTimeInterval1 : NumberPerTimeInterval

Unit: NumberPerTimeIntervalUnit = ReciprocalSecond
Value: Double = 42.0

Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double"
  Value="42.0"
  Units="ReciprocalSecond"
  UnitsURI="http://data.posccaesar.org/rdl/RDS1355489" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance numberPerTimeInterval1. For a complete example, see the attribute *DesignCapacityWeighingQuantities* of the DEXPI class *BatchWeigher*.

12.69.2 Unit

Attribute (data)

The unit of measurement of the *NumberPerTimeInterval*.

Multiplicity: 1

Type: *NumberPerTimeIntervalUnit*

Implementation in Proteus Schema

See implementation of *NumberPerTimeInterval*.

Example

See example for *NumberPerTimeInterval*.

12.69.3 Value

Attribute (data)

The numerical value of the *NumberPerTimeInterval*.

Multiplicity: 1

Type: *Double*

Implementation in Proteus Schema

See implementation of *NumberPerTimeInterval*.

Example

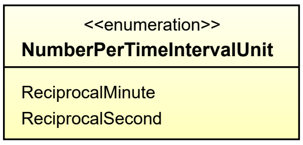
See example for *NumberPerTimeInterval*.

12.70. NumberPerTimeIntervalUnit

12.70.1 Overview

Enumeration

A unit of measurement for a physical quantity of application type *NullableNumberPerTimeInterval* with dimension T^{-1}.



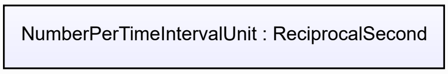
Literals

Name	Symbol	UN Code	RDL Reference
ReciprocalMinute	min^{-1}	C94	RECIPROCAL MINUTE http://data.posccaesar.org/rdl/RDS4316851589
ReciprocalSecond	s^{-1}	C97	RECIPROCAL SECOND http://data.posccaesar.org/rdl/RDS1355489

Implementation in Proteus Schema

NumberPerTimeIntervalUnit is only used as the type of the *Unit* attribute of *NumberPerTimeInterval*. *NumberPerTimeInterval* is implemented using *DEXPI generic attributes for physical quantities*. In a `<GenericAttribute>` element, the *NumberPerTimeIntervalUnit* literal is given by means of its RDL reference in the table above. The `Units` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `UnitsURI` attribute of the element is the URI of the RDL reference.

Example



Example: Implementation in Proteus Schema

```
<GenericAttribute
  Units="ReciprocalSecond"
  UnitsURI="http://data.posccaesar.org/rdl/RDS1355489" ...>
```

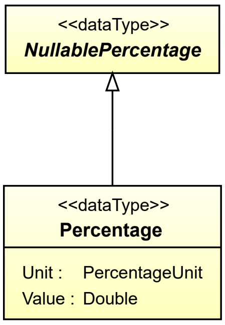
Note that the `<GenericAttribute>` element must have further attributes (`Format`, `Value`, `Name`, and `AttributeURI`). See the implementation examples for *NumberPerTimeInterval* and for the *DesignCapacityWeighingQuantities* attribute of *BatchWeigher*.

12.71. Percentage

12.71.1 Overview

Data type

An *actual value* for a physical quantity of type *NullablePercentage*, i.e., a physical quantity that has a numerical value and a unit of measurement.



Supertypes

- *NullablePercentage*

Attributes (data)

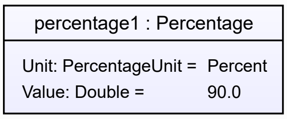
Name	Multiplicity	Type
Unit	1	*PercentageUnit*
Value	1	*Double*

Implementation in Proteus Schema

All data attributes with type *NullablePercentage* (the base type of *Percentage*) are implemented as *DEXPI generic attributes for physical quantities*.

Example

The instance percentage1 represents a *Percentage* of 90.0 ???.



Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double"
  Value="90.0"
  Units="Percent"
  UnitsURI="http://data.posccaesar.org/rdl/RDS1317959" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance percentage1. For a complete example, see the attribute *Efficiency* of the DEXPI class *FilterUnit*.

12.71.2 Unit

Attribute (data)

The unit of measurement of the *Percentage*.

Multiplicity: 1

Type: *PercentageUnit*

Implementation in Proteus Schema

See implementation of *Percentage*.

Example

See example for *Percentage*.

12.71.3 Value

Attribute (data)

The numerical value of the *Percentage*.

Multiplicity: 1

Type: *Double*

Implementation in Proteus Schema

See implementation of *Percentage*.

Example

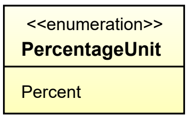
See example for *Percentage*.

12.72. PercentageUnit

12.72.1 Overview

Enumeration

A unit of measurement for a quantity of type *NullablePercentage*.



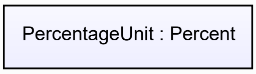
Literals

Name	Symbol	UN Code	RDL Reference
Percent	???	-	PERCENT http://data.posccaesar.org/rdl/RDS1317959

Implementation in Proteus Schema

PercentageUnit is only used as the type of the *Unit* attribute of *Percentage*. *Percentage* is implemented using *DEXPI generic attributes for physical quantities*. In a `<GenericAttribute>` element, the *PercentageUnit* literal is given by means of its RDL reference in the table above. The `Units` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `UnitsURI` attribute of the element is the URI of the RDL reference.

Example



Example: Implementation in Proteus Schema

```
<GenericAttribute
  Units="Percent"
  UnitsURI="http://data.posccaesar.org/rdl/RDS1317959" ...>
```

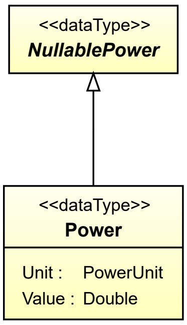
Note that the `<GenericAttribute>` element must have further attributes (`Format`, `Value`, `Name`, and `AttributeURI`). See the implementation examples for *Percentage* and for the *Efficiency* attribute of *FilterUnit*.

12.73. Power

12.73.1 Overview

Data type

An *actual value* for a physical quantity of type *NullablePower*, i.e., a physical quantity that has a numerical value and a unit of measurement.



Supertypes

- *NullablePower*

Attributes (data)

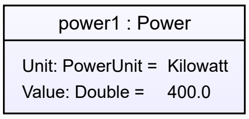
Name	Multiplicity	Type
Unit	1	*PowerUnit*
Value	1	*Double*

Implementation in Proteus Schema

All data attributes with type *NullablePower* (the base type of *Power*) are implemented as *DEXPI generic attributes for physical quantities*.

Example

The instance power1 represents a *Power* of 400.0 kW.



Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double"
  Value="400.0"
  Units="Kilowatt"
  UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance power1. For a complete example, see the attribute *DesignShaftPower* of the DEXPI class *Agglomerator*.

12.73.2 Unit

Attribute (data)

The unit of measurement of the *Power*.

Multiplicity: 1

Type: *PowerUnit*

Implementation in Proteus Schema

See implementation of *Power*.

Example

See example for *Power*.

12.73.3 Value

Attribute (data)

The numerical value of the *Power*.

Multiplicity: 1

Type: *Double*

Implementation in Proteus Schema

See implementation of *Power*.

Example

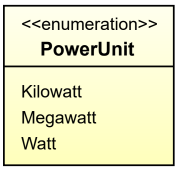
See example for *Power*.

12.74. PowerUnit

12.74.1 Overview

Enumeration

A unit of measurement for a physical quantity of type *NullablePower* with *dimension* L^{2}MT^{-3}.



Literals

Name	Symbol	UN Code	RDL Reference
Kilowatt	kW	-	KILOWATT http://data.posccaesar.org/rdl/RDS1330919
Megawatt	MW	-	MEGAWATT http://data.posccaesar.org/rdl/RDS1332584
Watt	W	-	WATT http://data.posccaesar.org/rdl/RDS1348154

Implementation in Proteus Schema

PowerUnit is only used as the type of the *Unit* attribute of *Power*. *Power* is implemented using *DEXPI generic attributes for physical quantities*. In a `<GenericAttribute>` element, the *PowerUnit* literal is given by means of its RDL reference in the table above. The `Units` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `UnitsURI` attribute of the element is the URI of the RDL reference.

Example

PowerUnit : Kilowatt

Example: Implementation in Proteus Schema

```
<GenericAttribute
  Units="Kilowatt"
  UnitsURI="http://data.posccaesar.org/rdl/RDS1330919" ...>
```

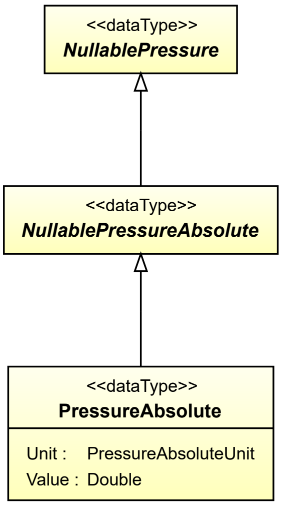
Note that the `<GenericAttribute>` element must have further attributes (`Format`, `Value`, `Name`, and `AttributeURI`). See the implementation examples for *Power* and for the *DesignShaftPower* attribute of *Agglomerator*.

12.75. PressureAbsolute

12.75.1 Overview

Data type

An *actual value* for a physical quantity of type *NullablePressureAbsolute*, i.e., a physical quantity that has a numerical value and a unit of measurement.



Supertypes

- *NullablePressureAbsolute*

Attributes (data)

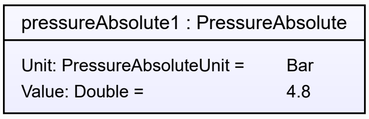
Name	Multiplicity	Type
Unit	1	*PressureAbsoluteUnit*
Value	1	*Double*

Implementation in Proteus Schema

All data attributes with type *NullablePressureAbsolute* (the base type of *PressureAbsolute*) are implemented as *DEXPI generic attributes for physical quantities*.

Example

The instance pressureAbsolute1 represents a *PressureAbsolute* of 4.8 bar.


pressureAbsolute1 : PressureAbsolute
Unit: PressureAbsoluteUnit = Bar
Value: Double = 4.8

Example: Implementation in Proteus Schema

```xml
<GenericAttribute
  Format="double"
  Value="4.8"
  Units="Bar"
  UnitsURI="http://data.posccaesar.org/rdl/RDS1314539" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance pressureAbsolute1. For a complete example, see the attribute *DesignDifferentialPressure* of the DEXPI class *Blower*.

12.75.2 Unit

Attribute (data)

The unit of measurement of the *PressureAbsolute*.

Multiplicity: 1

Type: *PressureAbsoluteUnit*

Implementation in Proteus Schema

See implementation of *PressureAbsolute*.

Example

See example for *PressureAbsolute*.

12.75.3 Value

Attribute (data)

The numerical value of the *PressureAbsolute*.

Multiplicity: 1

Type: *Double*

Implementation in Proteus Schema

See implementation of *PressureAbsolute*.

Example

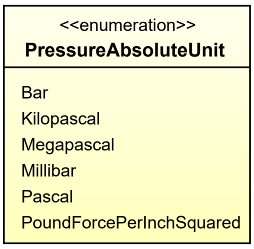
See example for *PressureAbsolute*.

12.76. PressureAbsoluteUnit

12.76.1 Overview

Enumeration

A unit of measurement for a physical quantity of application type *NullablePressureAbsolute* with dimension L^{-1}MT^{-2}.



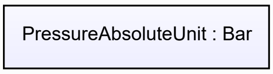
Literals

Name	Symbol	UN Code	RDL Reference
Bar	bar	BAR	BAR http://data.posccaesar.org/rdl/RDS1314539
Kilopascal	kPa	KPA	KILOPASCAL http://data.posccaesar.org/rdl/RDS1330559
Megapascal	MPa	MPA	MEGAPASCAL http://data.posccaesar.org/rdl/RDS1332404
Millibar	mbar	MBR	MILLIBAR http://data.posccaesar.org/rdl/RDS11617875
Pascal	Pa	PAL	PASCAL http://data.posccaesar.org/rdl/RDS1338749
PoundForcePerInchSquared	lbf/in^2	PS	POUND FORCE PER INCH SQUARED http://data.posccaesar.org/rdl/RDS1341809

Implementation in Proteus Schema

PressureAbsoluteUnit is only used as the type of the *Unit* attribute of *PressureAbsolute*. *PressureAbsolute* is implemented using *DEXPI generic attributes for physical quantities*. In a <GenericAttribute> element, the *PressureAbsoluteUnit* literal is given by means of its RDL reference in the table above. The Units attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The UnitsURI attribute of the element is the URI of the RDL reference.

Example



Example: Implementation in Proteus Schema

```
<GenericAttribute
  Units="Bar"
  UnitsURI="http://data.posccaesar.org/rdl/RDS1314539" ...>
```

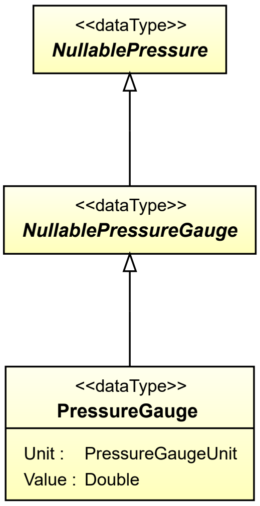
Note that the <GenericAttribute> element must have further attributes (Format, Value, Name, and AttributeURI). See the implementation examples for *PressureAbsolute* and for the *DesignDifferentialPressure* attribute of *Blower*.

12.77. PressureGauge

12.77.1 Overview

Data type

An *actual value* for a physical quantity of type *NullablePressureGauge*, i.e., a physical quantity that has a numerical value and a unit of measurement.



Supertypes

- *NullablePressureGauge*

Attributes (data)

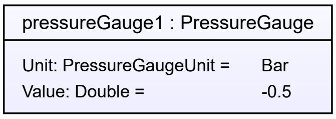
Name	Multiplicity	Type
Unit	1	*PressureGaugeUnit*
Value	1	*Double*

Implementation in Proteus Schema

All data attributes with type *NullablePressureGauge* (the base type of *PressureGauge*) are implemented as *DEXPI generic attributes for physical quantities*.

Example

The instance pressureGauge1 represents a *PressureGauge* of -0.5 bar.



Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double"
  Value="-0.5"
  Units="Bar"
  UnitsURI="http://data.posccaesar.org/rdl/RDS1314539" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance pressureGauge1. For a complete example, see the attribute *LowerLimitDesignPressure* of the DEXPI class *Chamber*.

12.77.2 Unit

Attribute (data)

The unit of measurement of the *PressureGauge*.

Multiplicity: 1

Type: *PressureGaugeUnit*

Implementation in Proteus Schema

See implementation of *PressureGauge*.

Example

See example for *PressureGauge*.

12.77.3 Value

Attribute (data)

The numerical value of the *PressureGauge*.

Multiplicity: 1

Type: *Double*

Implementation in Proteus Schema

See implementation of *PressureGauge*.

Example

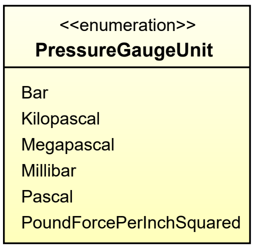
See example for *PressureGauge*.

12.78. PressureGaugeUnit

12.78.1 Overview

Enumeration

A unit of measurement for a physical quantity of application type *NullablePressureGauge* with dimension $L^{-1}MT^{-2}$.



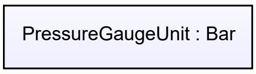
Literals

Name	Symbol	UN Code	RDL Reference
Bar	bar	BAR	BAR http://data.posccaesar.org/rdl/RDS1314539
Kilopascal	kPa	KPA	KILOPASCAL http://data.posccaesar.org/rdl/RDS1330559
Megapascal	MPa	MPA	MEGAPASCAL http://data.posccaesar.org/rdl/RDS1332404
Millibar	mbar	MBR	MILLIBAR http://data.posccaesar.org/rdl/RDS11617875
Pascal	Pa	PAL	PASCAL http://data.posccaesar.org/rdl/RDS1338749
PoundForcePerInchSquared	lbf/in^2	PS	POUND FORCE PER INCH SQUARED http://data.posccaesar.org/rdl/RDS1341809

Implementation in Proteus Schema

PressureGaugeUnit is only used as the type of the *Unit* attribute of *PressureGauge*. *PressureGauge* is implemented using *DEXPI generic attributes for physical quantities*. In a `<GenericAttribute>` element, the *PressureGaugeUnit* literal is given by means of its RDL reference in the table above. The `Units` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `UnitsURI` attribute of the element is the URI of the RDL reference.

Example


PressureGaugeUnit : Bar

Example: Implementation in Proteus Schema

```
<GenericAttribute
  Units="Bar"
  UnitsURI="http://data.posccaesar.org/rdl/RDS1314539" ...>
```

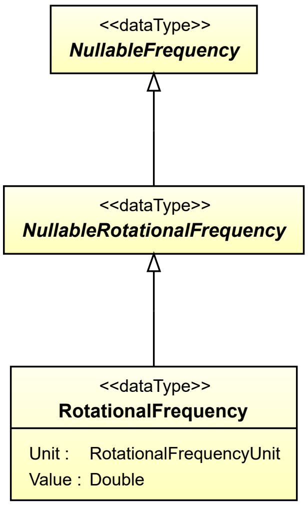
Note that the `<GenericAttribute>` element must have further attributes (`Format`, `Value`, `Name`, and `AttributeURI`). See the implementation examples for *PressureGauge* and for the *LowerLimitDesignPressure* attribute of *Chamber*.

12.79. RotationalFrequency

12.79.1 Overview

Data type

An *actual value* for a physical quantity of type *NullableRotationalFrequency*, i.e., a physical quantity that has a numerical value and a unit of measurement.



Supertypes

- *NullableRotationalFrequency*

Attributes (data)

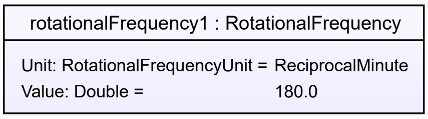
Name	Multiplicity	Type
Unit	1	*RotationalFrequencyUnit*
Value	1	*Double*

Implementation in Proteus Schema

All data attributes with type *NullableRotationalFrequency* (the base type of *RotationalFrequency*) are implemented as *DEXPI generic attributes for physical quantities*.

Example

The instance rotationalFrequency1 represents a *RotationalFrequency* of 180.0 min^{-1}.


rotationalFrequency1 : RotationalFrequency

Unit: RotationalFrequencyUnit = ReciprocalMinute
Value: Double = 180.0

Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double"
  Value="180.0"
  Units="ReciprocalMinute"
  UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance rotationalFrequency1. For a complete example, see the attribute *DesignRotationalSpeed* of the DEXPI class *Agglomerator*.

12.79.2 Unit

Attribute (data)

The unit of measurement of the *RotationalFrequency*.

Multiplicity: 1

Type: *RotationalFrequencyUnit*

Implementation in Proteus Schema

See implementation of *RotationalFrequency*.

Example

See example for *RotationalFrequency*.

12.79.3 Value

Attribute (data)

The numerical value of the *RotationalFrequency*.

Multiplicity: 1

Type: *Double*

Implementation in Proteus Schema

See implementation of *RotationalFrequency*.

Example

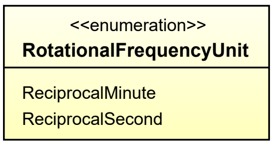
See example for *RotationalFrequency*.

12.80. RotationalFrequencyUnit

12.80.1 Overview

Enumeration

A unit of measurement for a physical quantity of application type *NullableRotationalFrequency* with dimension T^{-1}.



Literals

Name	Symbol	UN Code	RDL Reference
ReciprocalMinute	min^{-1}	C94	RECIPROCAL MINUTE http://data.posccaesar.org/rdl/RDS4316851589
ReciprocalSecond	s^{-1}	C97	RECIPROCAL SECOND http://data.posccaesar.org/rdl/RDS1355489

Implementation in Proteus Schema

RotationalFrequencyUnit is only used as the type of the *Unit* attribute of *RotationalFrequency*. *RotationalFrequency* is implemented using *DEXPI generic attributes for physical quantities*. In a `<GenericAttribute>` element, the *RotationalFrequencyUnit* literal is given by means of its RDL reference in the table above. The `Units` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `UnitsURI` attribute of the element is the URI of the RDL reference.

Example

RotationalFrequencyUnit : ReciprocalMinute

Example: Implementation in Proteus Schema

```
<GenericAttribute
  Units="ReciprocalMinute"
  UnitsURI="http://data.posccaesar.org/rdl/RDS4316851589" ...>
```

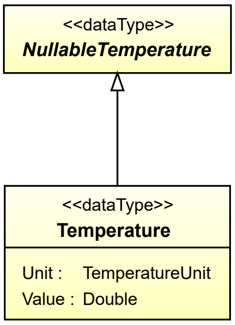
Note that the `<GenericAttribute>` element must have further attributes (`Format`, `Value`, `Name`, and `AttributeURI`). See the implementation examples for *RotationalFrequency* and for the *DesignRotationalSpeed* attribute of *Agglomerator*.

12.81. Temperature

12.81.1 Overview

Data type

An *actual value* for a physical quantity of type *NullableTemperature*, i.e., a physical quantity that has a numerical value and a unit of measurement.



Supertypes

- *NullableTemperature*

Attributes (data)

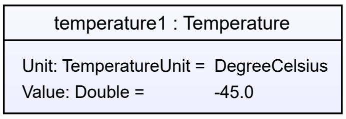
Name	Multiplicity	Type
Unit	1	*TemperatureUnit*
Value	1	*Double*

Implementation in Proteus Schema

All data attributes with type *NullableTemperature* (the base type of *Temperature*) are implemented as *DEXPI generic attributes for physical quantities*.

Example

The instance temperature1 represents a *Temperature* of -45.0 °C.


temperature1 : Temperature
Unit: TemperatureUnit = DegreeCelsius
Value: Double = -45.0

Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double"
  Value="-45.0"
  Units="DegreeCelsius"
  UnitsURI="http://data.posccaesar.org/rdl/RDS1322684" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance temperature1. For a complete example, see the attribute *LowerLimitDesignTemperature* of the DEXPI class *Chamber*.

12.81.2 Unit

Attribute (data)

The unit of measurement of the *Temperature*.

Multiplicity: 1

Type: *TemperatureUnit*

Implementation in Proteus Schema

See implementation of *Temperature*.

Example

See example for *Temperature*.

12.81.3 Value

Attribute (data)

The numerical value of the *Temperature*.

Multiplicity: 1

Type: *Double*

Implementation in Proteus Schema

See implementation of *Temperature*.

Example

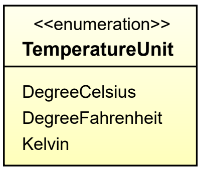
See example for *Temperature*.

12.82. TemperatureUnit

12.82.1 Overview

Enumeration

A unit of measurement for a physical quantity of type *NullableTemperature* with *dimension* Θ.



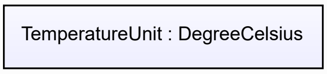
Literals

Name	Symbol	UN Code	RDL Reference
DegreeCelsius	°C	CEL	DEGREE CELSIUS http://data.posccaesar.org/rdl/RDS1322684
DegreeFahrenheit	°F	FAH	DEGREE FAHRENHEIT http://data.posccaesar.org/rdl/RDS1322549
Kelvin	K	KEL	KELVIN http://data.posccaesar.org/rdl/RDS1327904

Implementation in Proteus Schema

TemperatureUnit is only used as the type of the *Unit* attribute of *Temperature*. *Temperature* is implemented using *DEXPI generic attributes for physical quantities*. In a `<GenericAttribute>` element, the *TemperatureUnit* literal is given by means of its RDL reference in the table above. The `Units` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `UnitsURI` attribute of the element is the URI of the RDL reference.

Example



Example: Implementation in Proteus Schema

```
<GenericAttribute
  Units="DegreeCelsius"
  UnitsURI="http://data.posccaesar.org/rdl/RDS1322684" ...>
```

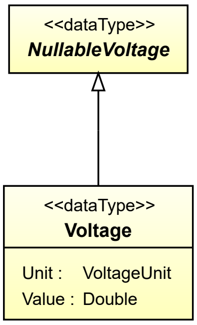
Note that the `<GenericAttribute>` element must have further attributes (`Format`, `Value`, `Name`, and `AttributeURI`). See the implementation examples for *Temperature* and for the *LowerLimitDesignTemperature* attribute of *Chamber*.

12.83. Voltage

12.83.1 Overview

Data type

An *actual value* for a physical quantity of type *NullableVoltage*, i.e., a physical quantity that has a numerical value and a unit of measurement.



Supertypes

- *NullableVoltage*

Attributes (data)

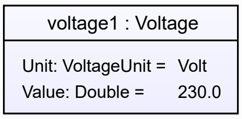
Name	Multiplicity	Type
Unit	1	*VoltageUnit*
Value	1	*Double*

Implementation in Proteus Schema

All data attributes with type *NullableVoltage* (the base type of *Voltage*) are implemented as *DEXPI generic attributes for physical quantities*.

Example

The instance voltage1 represents a *Voltage* of 230.0 V.



Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double"
  Value="230.0"
  Units="Volt"
  UnitsURI="http://data.posccaesar.org/rdl/RDS1347974" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance voltage1. For a complete example, see the attribute *NominalVoltage* of the DEXPI class *AlternatingCurrentMotor*.

12.83.2 Unit

Attribute (data)

The unit of measurement of the *Voltage*.

Multiplicity: 1

Type: *VoltageUnit*

Implementation in Proteus Schema

See implementation of *Voltage*.

Example

See example for *Voltage*.

12.83.3 Value

Attribute (data)

The numerical value of the *Voltage*.

Multiplicity: 1

Type: *Double*

Implementation in Proteus Schema

See implementation of *Voltage*.

Example

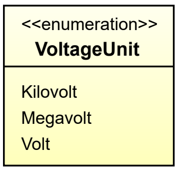
See example for *Voltage*.

12.84. VoltageUnit

12.84.1 Overview

Enumeration

A unit of measurement for a physical quantity of type *NullableVoltage* with *dimension* L^{2}MT^{-3}I^{-1}.



Literals

Name	Symbol	UN Code	RDL Reference
Kilovolt	kV	KVT	KILOVOLT http://data.posccaesar.org/rdl/RDS1359653041
Megavolt	MV	B78	MEGAVOLT http://data.posccaesar.org/rdl/RDS1359661910
Volt	V	VLT	VOLT http://data.posccaesar.org/rdl/RDS1347974

Implementation in Proteus Schema

VoltageUnit is only used as the type of the *Unit* attribute of *Voltage*. *Voltage* is implemented using *DEXPI generic attributes for physical quantities*. In a `<GenericAttribute>` element, the *VoltageUnit* literal is given by means of its RDL reference in the table above. The `Units` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `UnitsURI` attribute of the element is the URI of the RDL reference.

Example

Example: Implementation in Proteus Schema

```
<GenericAttribute
  Units="Volt"
  UnitsURI="http://data.posccaesar.org/rdl/RDS1347974" ...>
```

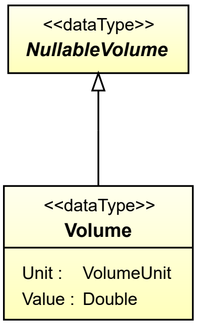
Note that the `<GenericAttribute>` element must have further attributes (`Format`, `Value`, `Name`, and `AttributeURI`). See the implementation examples for *Voltage* and for the *NominalVoltage* attribute of *AlternatingCurrentMotor*.

12.85. Volume

12.85.1 Overview

Data type

An *actual value* for a physical quantity of type *NullableVolume*, i.e., a physical quantity that has a numerical value and a unit of measurement.



Supertypes

- *NullableVolume*

Attributes (data)

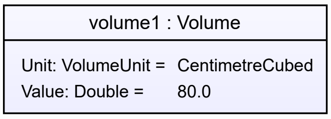
Name	Multiplicity	Type
Unit	1	*VolumeUnit*
Value	1	*Double*

Implementation in Proteus Schema

All data attributes with type *NullableVolume* (the base type of *Volume*) are implemented as *DEXPI generic attributes for physical quantities*.

Example

The instance volume1 represents a *Volume* of 80.0 cm^3.



Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double"
  Value="80.0"
  Units="CentimetreCubed"
  UnitsURI="http://data.posccaesar.org/rdl/RDS1357874" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance volume1. For a complete example, see the attribute *VolumePerStroke* of the DEXPI class *Displacer*.

12.85.2 Unit

Attribute (data)

The unit of measurement of the *Volume*.

Multiplicity: 1

Type: *VolumeUnit*

Implementation in Proteus Schema

See implementation of *Volume*.

Example

See example for *Volume*.

12.85.3 Value

Attribute (data)

The numerical value of the *Volume*.

Multiplicity: 1

Type: *Double*

Implementation in Proteus Schema

See implementation of *Volume*.

Example

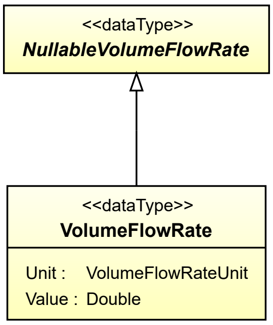
See example for *Volume*.

12.86. VolumeFlowRate

12.86.1 Overview

Data type

An *actual value* for a physical quantity of type *NullableVolumeFlowRate*, i.e., a physical quantity that has a numerical value and a unit of measurement.



Supertypes

- *NullableVolumeFlowRate*

Attributes (data)

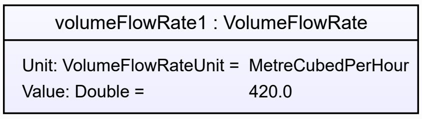
Name	Multiplicity	Type
Unit	1	*VolumeFlowRateUnit*
Value	1	*Double*

Implementation in Proteus Schema

All data attributes with type *NullableVolumeFlowRate* (the base type of *VolumeFlowRate*) are implemented as *DEXPI generic attributes for physical quantities*.

Example

The instance volumeFlowRate1 represents a *VolumeFlowRate* of 420.0 m^3/h.


volumeFlowRate1 : VolumeFlowRate
Unit: VolumeFlowRateUnit = MetreCubedPerHour
Value: Double = 420.0

Example: Implementation in Proteus Schema

```
<GenericAttribute
  Format="double"
  Value="420.0"
  Units="MetreCubedPerHour"
  UnitsURI="http://data.posccaesar.org/rdl/RDS1321064" ...>
```

Note that the `<GenericAttribute>` element must have a `Name` and an `AttributeURI` attribute. They depend on the data type attribute of the DEXPI class that owns the instance volumeFlowRate1. For a complete example, see the attribute *DesignVolumeFlowRate* of the DEXPI class *Agglomerator*.

12.86.2 Unit

Attribute (data)

The unit of measurement of the *VolumeFlowRate*.

Multiplicity: 1

Type: *VolumeFlowRateUnit*

Implementation in Proteus Schema

See implementation of *VolumeFlowRate*.

Example

See example for *VolumeFlowRate*.

12.86.3 Value

Attribute (data)

The numerical value of the *VolumeFlowRate*.

Multiplicity: 1

Type: *Double*

Implementation in Proteus Schema

See implementation of *VolumeFlowRate*.

Example

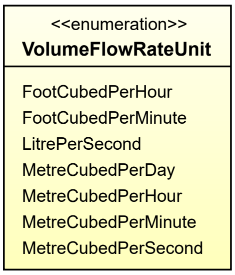
See example for *VolumeFlowRate*.

12.87. VolumeFlowRateUnit

12.87.1 Overview

Enumeration

A unit of measurement for a physical quantity of type *NullableVolumeFlowRate* with *dimension* L^3T^{-1}.



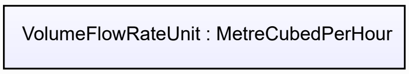
Literals

Name	Symbol	UN Code	RDL Reference
FootCubedPerHour	ft^3/h	2K	FOOT CUBED PER HOUR http://data.posccaesar.org/rdl/RDS1320029
FootCubedPerMinute	ft^3/min	2L	FOOT CUBED PER MINUTE http://data.posccaesar.org/rdl/RDS1320164
LitrePerSecond	l/s	G51	LITRE PER SECOND http://data.posccaesar.org/rdl/RDS1331369
MetreCubedPerDay	m^3/d	G52	METRE CUBED PER DAY http://data.posccaesar.org/rdl/RDS1320839
MetreCubedPerHour	m^3/h	MQH	METRE CUBED PER HOUR http://data.posccaesar.org/rdl/RDS1321064
MetreCubedPerMinute	m^3/min	G53	METRE CUBED PER MINUTE http://data.posccaesar.org/rdl/RDS1349909
MetreCubedPerSecond	m^3/s	MQS	METRE CUBED PER SECOND http://data.posccaesar.org/rdl/RDS1321379

Implementation in Proteus Schema

VolumeFlowRateUnit is only used as the type of the *Unit* attribute of *VolumeFlowRate*. *VolumeFlowRate* is implemented using *DEXPI generic attributes for physical quantities*. In a `<GenericAttribute>` element, the *VolumeFlowRateUnit* literal is given by means of its RDL reference in the table above. The `Units` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `UnitsURI` attribute of the element is the URI of the RDL reference.

Example



Example: Implementation in Proteus Schema

```
<GenericAttribute
  Units="MetreCubedPerHour"
  UnitsURI="http://data.posccaesar.org/rdl/RDS1321064" ...>
```

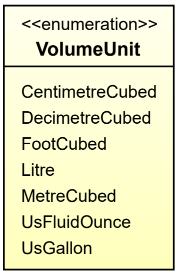
Note that the `<GenericAttribute>` element must have further attributes (`Format`, `Value`, `Name`, and `AttributeURI`). See the implementation examples for *VolumeFlowRate* and for the *DesignVolumeFlowRate* attribute of *Agglomerator*.

12.88. VolumeUnit

12.88.1 Overview

Enumeration

A unit of measurement for a physical quantity of type *NullableVolume* with *dimension* L^3.



Literals

Name	Symbol	UN Code	RDL Reference
CentimetreCubed	cm^3	CMQ	CENTIMETRE CUBED http://data.posccaesar.org/rdl/RDS1357874
DecimetreCubed	dm^3	DMQ	DECIMETRE CUBED http://data.posccaesar.org/rdl/RDS1319174
FootCubed	ft^3	FTQ	FOOT CUBED http://data.posccaesar.org/rdl/RDS1319669
Litre	l	LTR	LITRE http://data.posccaesar.org/rdl/RDS1331144

(continued on next page)

Name	Symbol	UN Code	RDL Reference
MetreCubed	m^3	MTQ	METRE CUBED http://data.posccaesar.org/rdl/RDS1349099
UsFluidOunce	fl oz (US)	OZA	US FLUID OUNCE http://data.posccaesar.org/rdl/RDS11619315
UsGallon	gal (US)	GLL	US GALLON http://data.posccaesar.org/rdl/RDS11615400

Implementation in Proteus Schema

VolumeUnit is only used as the type of the *Unit* attribute of *Volume*. *Volume* is implemented using *DEXPI generic attributes for physical quantities*. In a `<GenericAttribute>` element, the *VolumeUnit* literal is given by means of its RDL reference in the table above. The `Units` attribute of the element is the name of the RDL reference in camel-case; it equals the name of the literal. The `UnitsURI` attribute of the element is the URI of the RDL reference.

Example

Example: Implementation in Proteus Schema

```
<GenericAttribute
  Units="CentimetreCubed"
  UnitsURI="http://data.posccaesar.org/rdl/RDS1357874" ...>
```

Note that the `<GenericAttribute>` element must have further attributes (`Format`, `Value`, `Name`, and `AttributeURI`). See the implementation examples for *Volume* and for the *VolumePerStroke* attribute of *Displacer*.

Package Graphics 13

13.1. Overview

Until DEXPI 1.2, graphics have not been in the scope of the DEXPI Information Model, but only of Proteus Schema. The purpose of the Graphics package in DEXPI 1.3 is to complete DEXPI Information Model to cover conceptual and graphics information in a single comprehensive model.

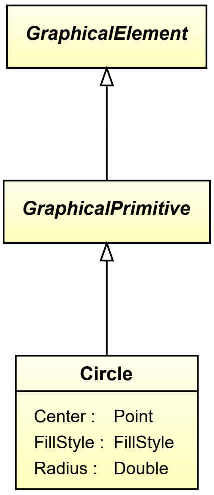
As this is a substantially new approach, the graphics package in DEXPI 1.3 is still informative, i.e., not normative as the other packages. It serves as a documentation of the way how graphics is represented in the official DEXPI Verificator 1.0 for DEXPI 1.3.

13.2. Circle

13.2.1 Overview

Class

A circle.



Supertypes

- *GraphicalPrimitive*

Attributes (data)

Name	Multiplicity	Type
Center	1	*Point*
FillStyle	1	*FillStyle*
Radius	1	*Double*

13.2.2 Center

Attribute (data)

The center position of the *Circle*.

Multiplicity: 1

Type: *Point*

13.2.3 FillStyle

Attribute (data)

The fill style of the *Circle*.

Multiplicity: 1

Type: *FillStyle*

13.2.4 Radius

Attribute (data)

The radius of the *Circle* in mm.

Multiplicity: 1

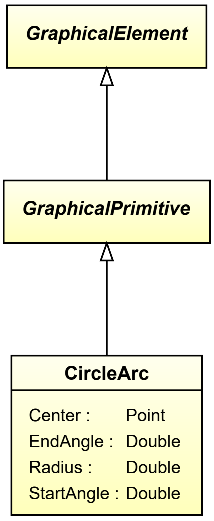
Type: *Double*

13.3. CircleArc

13.3.1 Overview

Class

A circle arc.



Supertypes

- *GraphicalPrimitive*

Attributes (data)

Name	Multiplicity	Type
Center	1	*Point*
EndAngle	1	*Double*
Radius	1	*Double*
StartAngle	1	*Double*

13.3.2 Center

Attribute (data)

The center position of the *CircleArc*.

Multiplicity: 1

Type: *Point*

13.3.3 EndAngle

Attribute (data)

The end angle of the *CircleArc*.

Multiplicity: 1

Type: *Double*

13.3.4 Radius

Attribute (data)

The radius of the *CircleArc* in mm.

Multiplicity: 1

Type: *Double*

13.3.5 StartAngle

Attribute (data)

The start angle of the *CircleArc*.

Multiplicity: 1

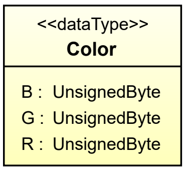
Type: *Double*

13.4. Color

13.4.1 Overview

Data type

A color. It is defined using the RDF color model.



Attributes (data)

Name	Multiplicity	Type
B	1	*UnsignedByte*
G	1	*UnsignedByte*
R	1	*UnsignedByte*

13.4.2 B

Attribute (data)

The intensity of the blue component of the *Color*.

Multiplicity: 1

Type: *UnsignedByte*

13.4.3 G

Attribute (data)

The intensity of the green component of the *Color*.

Multiplicity: 1

Type: *UnsignedByte*

13.4.4 R

Attribute (data)

The intensity of the red component of the *Color*.

Multiplicity: 1

Type: *UnsignedByte*

13.5. DashStyle

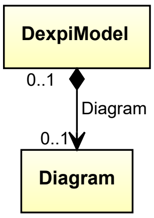
13.5.1 Overview

Enumeration

13.6. Diagram

13.6.1 Overview

Class

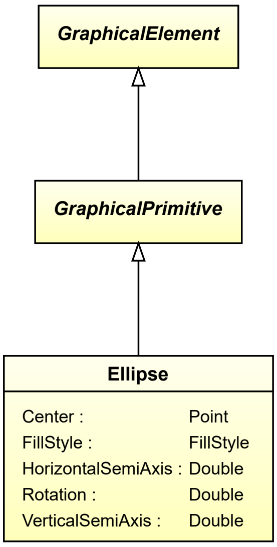


13.7. Ellipse

13.7.1 Overview

Class

An ellipse.



Supertypes

- *GraphicalPrimitive*

Attributes (data)

Name	Multiplicity	Type
Center	1	*Point*
FillStyle	1	*FillStyle*
HorizontalSemiAxis	1	*Double*
Rotation	1	*Double*
VerticalSemiAxis	1	*Double*

13.7.2 Center

Attribute (data)

The center position of the *Ellipse*.

Multiplicity: 1

Type: *Point*

13.7.3 FillStyle

Attribute (data)

The fill style of the *Ellipse*.

Multiplicity: 1

Type: *FillStyle*

13.7.4 HorizontalSemiAxis

Attribute (data)

The length of the horizontal semi-axis of the *Ellipse* in mm.

Multiplicity: 1

Type: *Double*

13.7.5 Rotation

Attribute (data)

The rotation of the *Ellipse* around its center in degrees.

Multiplicity: 1

Type: *Double*

13.7.6 VerticalSemiAxis

Attribute (data)

The length of the vertical semi-axis of the *Ellipse* in mm.

Multiplicity: 1

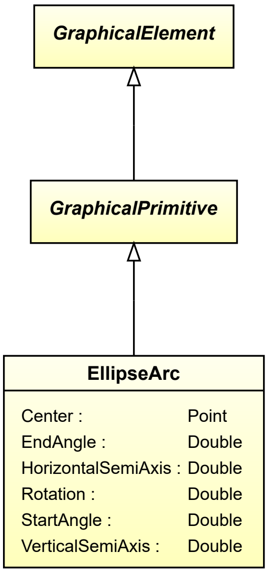
Type: *Double*

13.8. EllipseArc

13.8.1 Overview

Class

A ellipse arc.



Supertypes

- *GraphicalPrimitive*

Attributes (data)

Name	Multiplicity	Type
Center	1	*Point*
EndAngle	1	*Double*
HorizontalSemiAxis	1	*Double*
Rotation	1	*Double*
StartAngle	1	*Double*
VerticalSemiAxis	1	*Double*

13.8.2 Center

Attribute (data)

The center position of the *EllipseArc*.

Multiplicity: 1

Type: *Point*

13.8.3 EndAngle

Attribute (data)

The end angle of the *EllipseArc*.

Multiplicity: 1

Type: *Double*

13.8.4 HorizontalSemiAxis

Attribute (data)

The length of the horizontal semi-axis of the *EllipseArc* in mm.

Multiplicity: 1

Type: *Double*

13.8.5 Rotation

Attribute (data)

The rotation of the *EllipseArc* around its center in degrees.

Multiplicity: 1

Type: *Double*

13.8.6 StartAngle

Attribute (data)

The start angle of the *EllipseArc*.

Multiplicity: 1

Type: *Double*

13.8.7 VerticalSemiAxis

Attribute (data)

The length of the vertical semi-axis of the *EllipseArc* in mm.

Multiplicity: 1

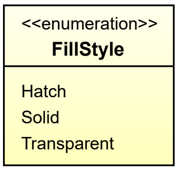
Type: *Double*

13.9. FillStyle

13.9.1 Overview

Enumeration

A fill style for a graphical element.



Literals

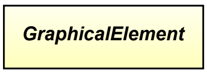
Name	Symbol
Hatch	Hatch
Solid	Solid
Transparent	Transparent

13.10. GraphicalElement

13.10.1 Overview

Abstract class

A graphical element.



Subtypes

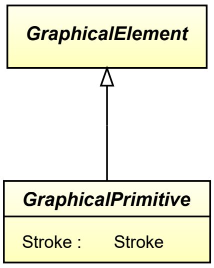
- *GraphicalPrimitive*

13.11. GraphicalPrimitive

13.11.1 Overview

Abstract class

A primitive graphical element.



Supertypes

- *GraphicalElement*

Subtypes

- *Circle*
- *CircleArc*
- *Ellipse*
- *EllipseArc*
- *PolyLine*
- *Polygon*

Attributes (data)

Name	Multiplicity	Type
Stroke	1	*Stroke*

13.11.2 Stroke

Attribute (data)

The stroke of the *GraphicalPrimitive*.

Multiplicity: 1

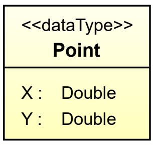
Type: *Stroke*

13.12. Point

13.12.1 Overview

Data type

A point in the X-Y-plane.



Attributes (data)

Name	Multiplicity	Type
X	1	*Double*
Y	1	*Double*

13.12.2 X

Attribute (data)

The X (horizontal) coordinate of the *Point* in mm.

Multiplicity: 1

Type: *Double*

13.12.3 Y

Attribute (data)

The Y (horizontal) coordinate of the *Point* in mm.

Multiplicity: 1

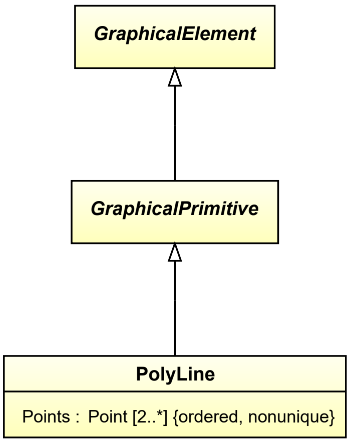
Type: *Double*

13.13. PolyLine

13.13.1 Overview

Class

A poly line, i.e., a line that consists of 1 or more straight sections.



Supertypes

- *GraphicalPrimitive*

Attributes (data)

Name	Multiplicity	Type
Points	2..*	*Point*

13.13.2 Points

Attribute (data)

The points of the *PolyLine*.

Multiplicity: 2..*

Type: *Point*

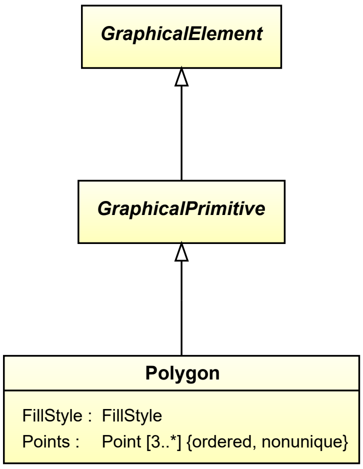
Modifiers: ordered, nonunique

13.14. Polygon

13.14.1 Overview

Class

A polygon.



Supertypes

- *GraphicalPrimitive*

Attributes (data)

Name	Multiplicity	Type
FillStyle	1	*FillStyle*
Points	3..*	*Point*

13.14.2 FillStyle

Attribute (data)

The fill style of the *Polygon*.

Multiplicity: 1

Type: *FillStyle*

13.14.3 Points

Attribute (data)

The points of the vertices of the *Polygon*.

Multiplicity: 3..*

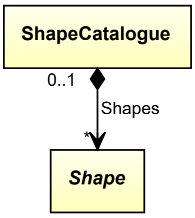
Type: *Point*

Modifiers: ordered, nonunique

13.15. Shape

13.15.1 Overview

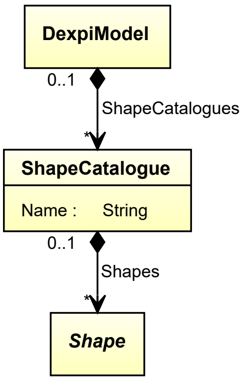
Abstract class



13.16. ShapeCatalogue

13.16.1 Overview

Class



Attributes (data)

Name	Multiplicity	Type
Name	1	*String*

Attributes (composition)

Name	Multiplicity	Type
Shapes	*	*Shape*

13.16.2 Name

Attribute (data)

The name of the *ShapeCatalogue*.

Multiplicity: 1

Type: *String*

13.16.3 Shapes

Attribute (composition)

The shapes of the *ShapeCatalogue*.

Multiplicity: *

Type: *Shape*

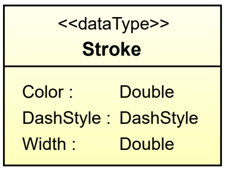
Opposite multiplicity: 0..1

13.17. Stroke

13.17.1 Overview

Data type

A stroke style.



Attributes (data)

Name	Multiplicity	Type
Color	1	*Double*
DashStyle	1	*DashStyle*
Width	1	*Double*

13.17.2 Color

Attribute (data)

The color of the *Stroke*.

Multiplicity: 1

Type: *Double*

13.17.3 DashStyle

Attribute (data)

The dash style of the *Stroke*.

Multiplicity: 1

Type: *DashStyle*

13.17.4 Width

Attribute (data)

The width of the *Stroke* in mm.

Multiplicity: 1

Type: *Double*

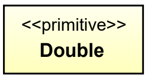
Package DataTypes 14

14.1. Overview

The *DataTypes* package contains fundamental data types. Note that further data types that are specific to engineering information in P&IDs are provided by the *Enumerations* and *PhysicalQuantities* packages. Finally, the *Graphics* package contains data types required for P&ID graphics.

14.1.1 Simple Data Types

Simple data types such as *Double* correspond to conventional data types in programming languages or data formats.

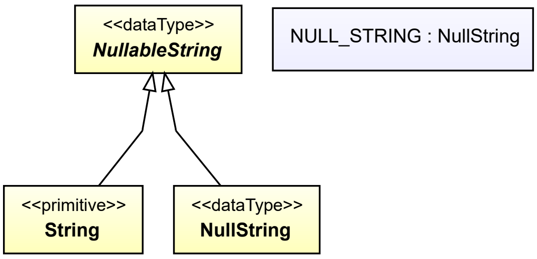


There are 2 simple data types:

- *Double*
- *UnsignedByte*

14.1.2 Nullable Data Types

Nullable data types are similar to simple data types, but they provide an additional *null value*. A nullable data type such as *NullableString* is always abstract. It has two concrete sub types: a type for actual values (e.g., *String*), and a type for the *null value* (e.g., *NullString*). The latter is a singleton type, i.e., there is only one instance of this type (e.g., *NULL_STRING*).

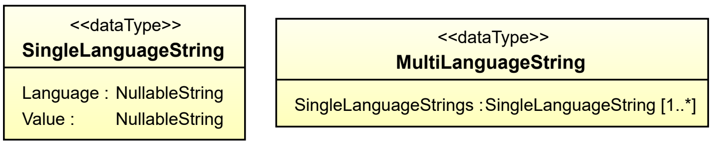


There are 4 nullable data types:

- *NullableAnyURI*
- *NullableDateTime*
- *NullableInteger*
- *NullableString*

14.1.3 Complex Data Types

Complex data types are aggregations of other data types.



There are 2 complex data types:

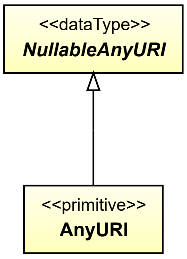
- *MultiLanguageString*
- *SingleLanguageString*

14.2. AnyURI

14.2.1 Overview

Data type

A Uniform Resource Identifier (URI). The value space of *AnyURI* is the same as that of the XML Schema data type anyURI as specified by the W3C Recommendation XML Schema Part 2: Datatypes Second Edition from October 28, 2004.



Supertypes

- *NullableAnyURI*

Implementation in Proteus Schema

An *AnyURI* is implemented as a literal for the XML Schema data type anyURI.
The way this literal is used in a Proteus XML document depends on the Proteus Schema implementations of the data attributes with type *NullableAnyURI*. There is no attribute with direct type *AnyURI*.

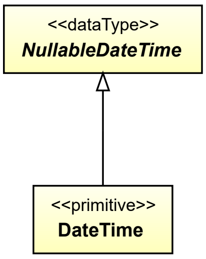
- *NullableAnyURI* is the type of the *AttributeURI* attribute of *CustomAttribute*. See Proteus Schema implementation of *AttributeURI*.
- *NullableAnyURI* is the type of the *TypeURI* attribute of *CustomObject*. See Proteus Schema implementation of *TypeURI*.

14.3. DateTime

14.3.1 Overview

Data type

A date time. The value space of *DateTime* is the same as that of the XML Schema data type dateTime as specified by the W3C Recommendation XML Schema Part 2: Datatypes Second Edition from October 28, 2004, with the exception that *DateTime* values do not carry any timezone information.



Supertypes

- *NullableDateTime*

Implementation in Proteus Schema

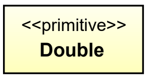
The base type *NullableDateTime* is only used as the type of the *ExportDateTime* attribute of *DexpiModel*. For this attribute, special rules apply; see Proteus Schema implementation of *ExportDateTime*. There are no attributes with type *DateTime*.

14.4. Double

14.4.1 Overview

Data type

A double-precision 64-bit floating point number. The value space of *Double* is the same as that of the XML Schema data type double as specified by the W3C Recommendation XML Schema Part 2: Datatypes Second Edition from October 28, 2004, with the exception that the special values *positive infinity* (`INF`), *negative infinity* (`-INF`), and *not-a-number* (`NaN`) are excluded from the value space of *Double*.



Implementation in Proteus Schema

A *Double* is implemented as a literal for the XML Schema data type double.
The way this literal is used in a Proteus XML document depends on the Proteus Schema implementations of the data attributes with type *Double*.

- *Double* is the type of the *Value* attributes of the physical quantity types and physical quantity application types in the *PhysicalQuantities* package. For example, see the Proteus Schema implementations of *Area*::*Value* and *ElectricalFrequency*::*Value*.
- *Double* is the type of several data attributes of classes and data types in the *Graphics* package. See the Proteus Schema implementations of these attributes:
  - *Circle*::*Radius*
  - *CircleArc*::*EndAngle*
  - *CircleArc*::*Radius*
  - *CircleArc*::*StartAngle*
  - *Ellipse*::*HorizontalSemiAxis*

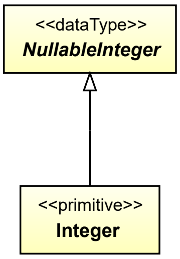
- *Ellipse*::*Rotation*
- *Ellipse*::*VerticalSemiAxis*
- *EllipseArc*::*EndAngle*
- *EllipseArc*::*HorizontalSemiAxis*
- *EllipseArc*::*Rotation*
- *EllipseArc*::*StartAngle*
- *EllipseArc*::*VerticalSemiAxis*
- *Point*::*X*
- *Point*::*Y*
- *Stroke*::*Color*
- *Stroke*::*Width*

14.5. Integer

14.5.1 Overview

Data type

An integer number. The value space of *Integer* is the same as that of the XML Schema data type integer as specified by the W3C Recommendation XML Schema Part 2: Datatypes Second Edition from October 28, 2004.



Supertypes

- *NullableInteger*

Implementation in Proteus Schema

An *Integer* is implemented as a literal for the XML Schema data type integer.
The way this literal is used in a Proteus XML document depends on the Proteus Schema implementations of the data attributes with type *NullableInteger*. There is no attribute with direct type *Integer*.

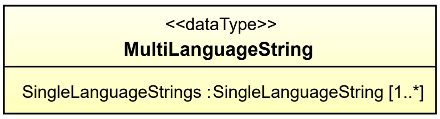
- *NullableInteger* is the type of the *Value* attribute of *CustomIntegerAttribute*. *CustomIntegerAttribute* is implemented as a *DEXPI custom generic attribute*.
- All other attributes with type *NullableInteger* are implemented as *DEXPI generic attributes*. For example, see the *NumberOfPackings* attribute of *ColumnPackingsArrangement*.

14.6. MultiLanguageString

14.6.1 Overview

Data type

A container for one or more *SingleLanguageString*s. *MultiLanguageString* is used as the type of data attributes which have language-dependent string values: Each *SingleLanguageString* contains a *NullableString Value* and a *Language* tag.



Attributes (data)

Name	Multiplicity	Type
SingleLanguageStrings	1..*	*SingleLanguageString*

Implementation in Proteus Schema

The Proteus Schema implementation of *MultiLanguageString* depends on the Proteus Schema implementations of the data attributes with type *MultiLanguageString*.

- *MultiLanguageString* is the type of the *Value* attribute of *CustomMultiLanguageStringAttribute*. *CustomMultiLanguageStringAttribute* is implemented as a *set of custom generic attributes for multi-language string values*.
- All other data attributes with type *MultiLanguageString* are implemented as *sets of DEXPI generic attributes for multi-language string values*. For example, see the *ChamberDescription* attribute of *Chamber*.

14.6.2 SingleLanguageStrings

Attribute (data)

The *SingleLanguageString*s with language-dependent values for the *MultiLanguageString*.

Multiplicity: 1..*

Type: *SingleLanguageString*

Implementation in Proteus Schema

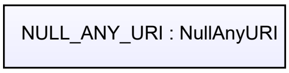
See Proteus Schema implementation of *MultiLanguageString*.

14.7. NULL_ANY_URI

14.7.1 Overview

Instance

The only instance of the singleton type *NullAnyURI*. This instance represents the *null value* of type *NullableAnyURI*.



Type

- *NullAnyURI*

Implementation in Proteus Schema

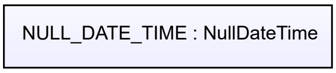
See Proteus Schema implementation of *NullableAnyURI*.

14.8. NULL_DATE_TIME

14.8.1 Overview

Instance

The only instance of the singleton type *NullDateTime*. This instance represents the *null value* of type *NullableDateTime*.



Type

- *NullDateTime*

Implementation in Proteus Schema

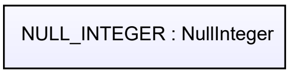
See Proteus Schema implementation of *NullableDateTime*.

14.9. NULL_INTEGER

14.9.1 Overview

Instance

The only instance of the singleton type *NullInteger*. This instance represents the *null value* of type *NullableInteger*.



Type

- *NullInteger*

Implementation in Proteus Schema

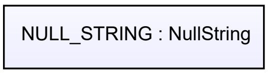
See Proteus Schema implementation of *NullableInteger*.

14.10. NULL_STRING

14.10.1 Overview

Instance

The only instance of the singleton type *NullString*. This instance represents the *null value* of type *NullableString*.



Type

- *NullString*

Implementation in Proteus Schema

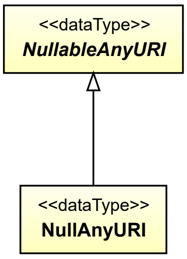
See Proteus Schema implementation of *NullableString*.

14.11. NullAnyURI

14.11.1 Overview

Data type

A *null value* for a value of type *NullableAnyURI*. The only instance of this singleton type is *NULL_ANY_URI*.



Supertypes

- *NullableAnyURI*

Implementation in Proteus Schema

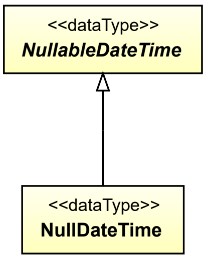
See Proteus Schema implementation of the base type *NullableAnyURI*

14.12. NullDateTime

14.12.1 Overview

Data type

A *null value* for a value of type *NullableDateTime*. The only instance of this singleton type is *NULL_DATE_TIME*.



Supertypes

- *NullableDateTime*

Implementation in Proteus Schema

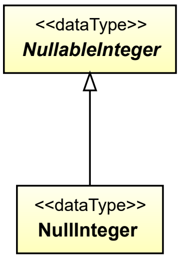
See Proteus Schema implementation of the base type *NullableDateTime*

14.13. NullInteger

14.13.1 Overview

Data type

A *null value* for a value of type *NullableInteger*. The only instance of this singleton type is *NULL_INTEGER*.



Supertypes

- *NullableInteger*

Implementation in Proteus Schema

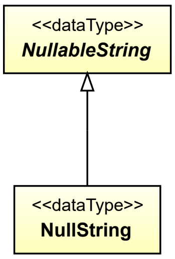
See Proteus Schema implementation of the base type *NullableInteger*

14.14. NullString

14.14.1 Overview

Data type

A *null value* for a value of type *NullableString*. The only instance of this singleton type is *NULL_STRING*.



Supertypes

- *NullableString*

Implementation in Proteus Schema

See Proteus Schema implementation of the base type *NullableString*

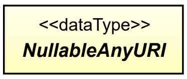
14.15. NullableAnyURI

14.15.1 Overview

Abstract data type

A Uniform Resource Identifier (URI), or a null value. *NullableAnyURI* is abstract and has two concrete subtypes:

- an *AnyURI* is an *actual value* for a Uniform Resource Identifier (URI);
- a *NullAnyURI* is a *null value* that explicitly indicates the absence of a Uniform Resource Identifier (URI).



Subtypes

- *AnyURI*
- *NullAnyURI*

Implementation in Proteus Schema

The Proteus Schema implementation of *NullableAnyURI* depends on the Proteus Schema implementations of the data attributes with type *NullableAnyURI*.

- *NullableAnyURI* is the type of the *AttributeURI* attribute of *CustomAttribute*. See Proteus Schema implementation of *AttributeURI*.
- *NullableAnyURI* is the type of the *TypeURI* attribute of *CustomObject*. See Proteus Schema implementation of *TypeURI*.

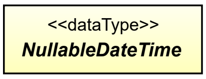
14.16. NullableDateTime

14.16.1 Overview

Abstract data type

A date time, or a null value. *NullableDateTime* is abstract and has two concrete subtypes:

- a *DateTime* is an *actual value* for a date time;
- a *NullDateTime* is a *null value* that explicitly indicates the absence of a date time.



Subtypes

- *DateTime*
- *NullDateTime*

Implementation in Proteus Schema

NullableDateTime is only used as the type of the *ExportDateTime* attribute of *DexpiModel*. For this attribute, special rules apply; see Proteus Schema implementation of *ExportDateTime*.

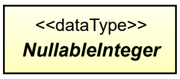
14.17. NullableInteger

14.17.1 Overview

Abstract data type

An integer number, or a null value. *NullableInteger* is abstract and has two concrete subtypes:

- an *Integer* is an *actual value* for an integer number;
- a *NullInteger* is a *null value* that explicitly indicates the absence of an integer number.



Subtypes

- *Integer*
- *NullInteger*

Implementation in Proteus Schema

The Proteus Schema implementation of *NullableInteger* depends on the Proteus Schema implementations of the data attributes with type *NullableInteger*.

- *NullableInteger* is the type of the *Value* attribute of *CustomIntegerAttribute*. *CustomIntegerAttribute* is implemented as a *DEXPI custom generic attribute*.
- All other data attributes with type *NullableInteger* are implemented as *DEXPI generic attributes*. For example, see the *NumberOfPackings* attribute of *ColumnPackingsArrangement*.

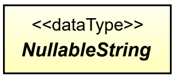
14.18. NullableString

14.18.1 Overview

Abstract data type

A sequence of characters, or a null value. *NullableString* is abstract and has two concrete subtypes:

- a *String* is an *actual value* for a sequence of characters;
- a *NullString* is a *null value* that explicitly indicates the absence of a sequence of characters.



Subtypes

- *NullString*
- *String*

Implementation in Proteus Schema

The Proteus Schema implementation of *NullableString* depends on the Proteus Schema implementations of the data attributes with type *NullableString*.

- *NullableString* is the type of the *Value* attribute of *CustomStringAttribute*. *CustomStringAttribute* is implemented as a *DEXPI custom generic attribute*.
- All other data attributes with type *NullableString* are implemented as *DEXPI generic attributes*. For example, see the *MaterialOfConstructionCode* attribute of *AgitatorRotor*.

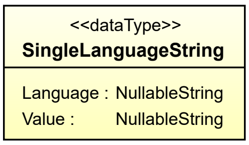
Note that there are further data attributes with type *String* instead of *NullableString*. See the Proteus Schema implementation of *String* for a list of these special cases.

14.19. SingleLanguageString

14.19.1 Overview

Data type

A *SingleLanguageString* contains a *NullableString* as its *Value* and a *Language* tag. *SingleLanguageString* is only used within *MultiLanguageString*. See the latter data type for more details.



Attributes (data)

Name	Multiplicity	Type
Language	1	*NullableString*
Value	1	*NullableString*

Implementation in Proteus Schema

See Proteus Schema implementation of *MultiLanguageString*.

14.19.2 Language

Attribute (data)

The language of the *Value* of the *SingleLanguageString*. If not *NULL_STRING*, the language must be given as a language tag standardized by IETF BCP 47 (*Best Current Practice 47* by the *Internet Engineering Task Force*). IETF BCP 47 is based on ISO 639.

In DEXPI, only those language tags are allowed

- that are listed in the normative registry file at https://www.iana.org/assignments/language-subtag-registry/language-subtag-registry from September 29, 2020,
- that consist of exactly two letters (in particular extended tags such as *en-US* or *de-CH-1996* are not allowed),
- that refer to an actual language (the *type* defined by IETF BCP 47 must be *language*),
- and that are not deprecated.

The table below contains all permitted language tags.

Tag	Language
aa	Afar
ab	Abkhazian
ae	Avestan
af	Afrikaans
ak	Akan
am	Amharic
an	Aragonese
ar	Arabic
as	Assamese
av	Avaric
ay	Aymara
az	Azerbaijani
ba	Bashkir
be	Belarusian
bg	Bulgarian
bh	Bihari languages
bi	Bislama
bm	Bambara

(continued on next page)

Tag	Language
bn	Bengali/Bangla
bo	Tibetan
br	Breton
bs	Bosnian
ca	Catalan/Valencian
ce	Chechen
ch	Chamorro
co	Corsican
cr	Cree
cs	Czech
cu	Church Slavic/Church Slavonic/Old Bulgarian/Old Church Slavonic/Old Slavonic
cv	Chuvash
cy	Welsh
da	Danish
de	German
dv	Dhivehi/Divehi/Maldivian
dz	Dzongkha
ee	Ewe
el	Modern Greek (1453-)
en	English
eo	Esperanto
es	Spanish/Castilian
et	Estonian
eu	Basque
fa	Persian
ff	Fulah
fi	Finnish
fj	Fijian
fo	Faroese
fr	French
fy	Western Frisian
ga	Irish
gd	Scottish Gaelic/Gaelic
gl	Galician
gn	Guarani
gu	Gujarati
gv	Manx
ha	Hausa
he	Hebrew
hi	Hindi
ho	Hiri Motu
hr	Croatian
ht	Haitian/Haitian Creole

(continued on next page)

Tag	Language
hu	Hungarian
hy	Armenian
hz	Herero
ia	Interlingua (International Auxiliary Language Association)
id	Indonesian
ie	Interlingue/Occidental
ig	Igbo
ii	Sichuan Yi/Nuosu
ik	Inupiaq
io	Ido
is	Icelandic
it	Italian
iu	Inuktitut
ja	Japanese
jv	Javanese
ka	Georgian
kg	Kongo
ki	Kikuyu/Gikuyu
kj	Kuanyama/Kwanyama
kk	Kazakh
kl	Kalaallisut/Greenlandic
km	Khmer/Central Khmer
kn	Kannada
ko	Korean
kr	Kanuri
ks	Kashmiri
ku	Kurdish
kv	Komi
kw	Cornish
ky	Kirghiz/Kyrgyz
la	Latin
lb	Luxembourgish/Letzeburgesch
lg	Ganda/Luganda
li	Limburgan/Limburger/Limburgish
ln	Lingala
lo	Lao
lt	Lithuanian
lu	Luba-Katanga
lv	Latvian
mg	Malagasy
mh	Marshallese
mi	Maori
mk	Macedonian

(continued on next page)

Tag	Language
ml	Malayalam
mn	Mongolian
mr	Marathi
ms	Malay (macrolanguage)
mt	Maltese
my	Burmese
na	Nauru
nb	Norwegian Bokmål
nd	North Ndebele
ne	Nepali (macrolanguage)
ng	Ndonga
nl	Dutch/Flemish
nn	Norwegian Nynorsk
no	Norwegian
nr	South Ndebele
nv	Navajo/Navaho
ny	Nyanja/Chewa/Chichewa
oc	Occitan (post 1500)
oj	Ojibwa
om	Oromo
or	Oriya (macrolanguage)/Odia (macrolanguage)
os	Ossetian/Ossetic
pa	Panjabi/Punjabi
pi	Pali
pl	Polish
ps	Pushto/Pashto
pt	Portuguese
qu	Quechua
rm	Romansh
rn	Rundi
ro	Romanian/Moldavian/Moldovan
ru	Russian
rw	Kinyarwanda
sa	Sanskrit
sc	Sardinian
sd	Sindhi
se	Northern Sami
sg	Sango
sh	Serbo-Croatian
si	Sinhala/Sinhalese
sk	Slovak
sl	Slovenian
sm	Samoan

(continued on next page)

Tag	Language
sn	Shona
so	Somali
sq	Albanian
sr	Serbian
ss	Swati
st	Southern Sotho
su	Sundanese
sv	Swedish
sw	Swahili (macrolanguage)
ta	Tamil
te	Telugu
tg	Tajik
th	Thai
ti	Tigrinya
tk	Turkmen
tl	Tagalog
tn	Tswana
to	Tonga (Tonga Islands)
tr	Turkish
ts	Tsonga
tt	Tatar
tw	Twi
ty	Tahitian
ug	Uighur/Uyghur
uk	Ukrainian
ur	Urdu
uz	Uzbek
ve	Venda
vi	Vietnamese
vo	Volapük
wa	Walloon
wo	Wolof
xh	Xhosa
yi	Yiddish
yo	Yoruba
za	Zhuang/Chuang
zh	Chinese
zu	Zulu

Multiplicity: 1

Type: *NullableString*

Implementation in Proteus Schema

See Proteus Schema implementation of *MultiLanguageString*.

14.19.3 Value

Attribute (data)

The actual *NullableString* value of the *SingleLanguageString*.

Multiplicity: 1

Type: *NullableString*

Implementation in Proteus Schema

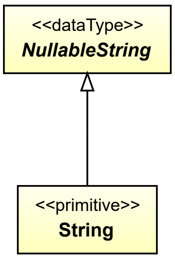
See Proteus Schema implementation of *MultiLanguageString*.

14.20. String

14.20.1 Overview

Data type

A sequence of characters. The value space of *String* is the same as that of the XML Schema data type string as specified by the W3C Recommendation XML Schema Part 2: Datatypes Second Edition from October 28, 2004.



Supertypes

- *NullableString*

Implementation in Proteus Schema

A *String* is implemented as a literal for the XML Schema data type string.
The way this literal is used in a Proteus XML document depends on the Proteus Schema implementations of the data attributes with type *NullableString* or *String*.

- *NullableString* is the type of the *Value* attribute of *CustomStringAttribute*. *CustomStringAttribute* is implemented as a *DEXPI custom generic attribute*.
- All other attributes with type *NullableString* are implemented as *DEXPI generic attributes*. For example, see the *MaterialOfConstructionCode* attribute of *AgitatorRotor*.
- For the attributes with direct type *String* special rules apply. These attributes are:
  - *CustomAttribute*::*AttributeName*
  - *CustomObject*::*TypeName*
  - *ShapeCatalogue*::*Name*

14.21. UnsignedByte

14.21.1 Overview

Data type

An integer number in the range from 0 to 255 (inclusively). The value space of *UnsignedByte* is the same as that of the XML Schema data type unsignedByte as specified by the W3C Recommendation XML Schema Part 2: Datatypes Second Edition from October 28, 2004.

Implementation in Proteus Schema

UnsignedByte is only used as the type of the *R*, *G*, and *B* attributes of *Color*. For these attributes, special rules apply; see Proteus Schema implementation of *Color*.

Appendix

Glossary A

A.1. Dimension of a Physical Quantity Type

A physical quantity type such as *mass*, *length* or *area* is characterized by its dimension (for example, see https://en.wikipedia.org/wiki/Physical_quantity). It is possible to express the dimension of any physical quantity type using the dimensions of some base quantity types. The choice of the base quantity types is arbitrary to a certain extent. The base quantity types that underly the International System of Units are listed below.

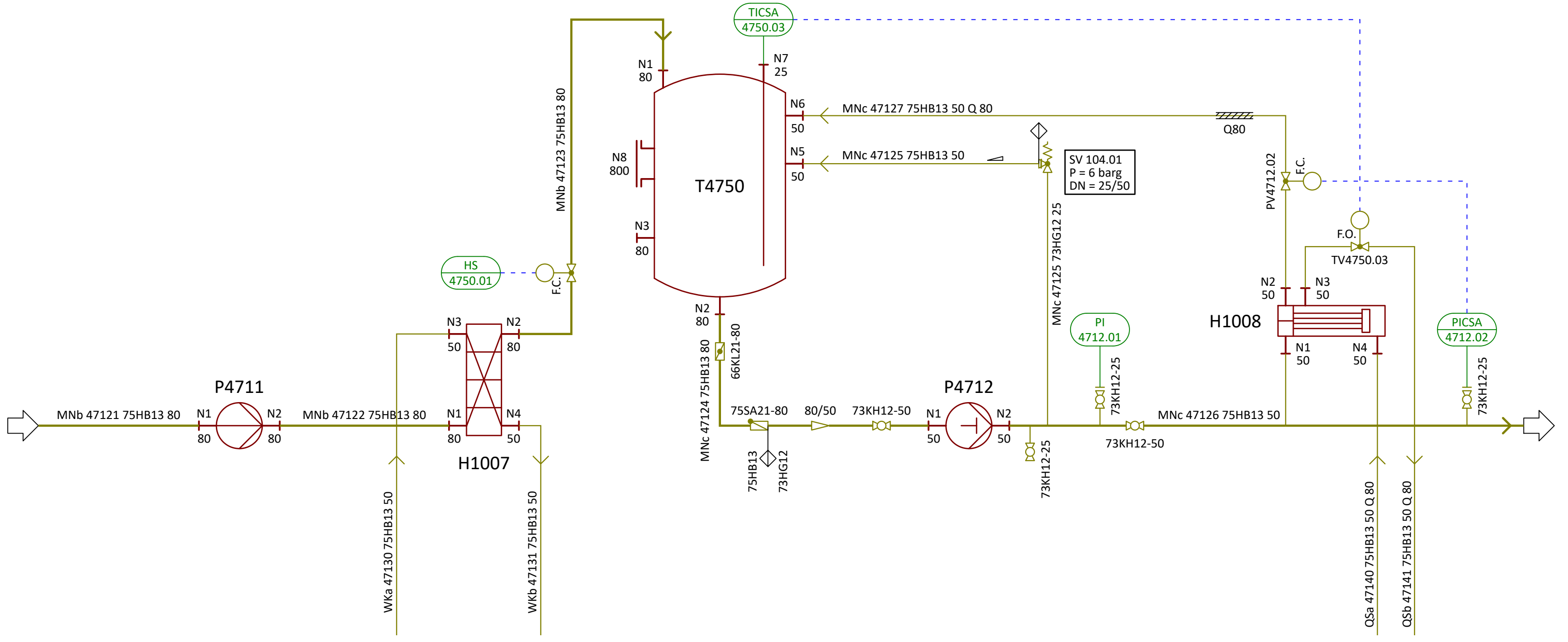
Base Quantity Type	Dimension Symbol
amount of substance	N
electric current	I
length	L
luminous intensity	J
mass	M
thermodynamic temperature	Θ (Greek capital letter Theta)
time	T

Using the symbols given in the table, the dimension of *length* is L, and the dimension of *area* is L·L, or L^2.

A.2. Unified Modeling Language

The Unified Modeling Language (UML) is a general-purpose modeling language for the *analysis, design, and implementation of software-based systems* [UML:1]. Within this specification, we use the notation [UML:x] to refer to section x of *OMG® Unified Modeling Language® (OMG UML®), Version 2.5.1* (see https://www.omg.org/spec/UML/2.5.1).

Reference P&ID B



Ident	H1007	
Design Press. min. Chamber 1 / 2	-1 barg	-1 barg
Design Press. max. Chamber 1 / 2	60 barg	30 barg
Design Temp. min. Chamber 1 / 2	-45 °C	-45 °C
Design Temp. max. Chamber 1 / 2	100 °C	100 °C
Design Duty / Design Heat Transfer Area	313 kW	46.8 m²
DesignPlate Height / Plate Width	850 mm	1100 mm
MaterialPlates	1.4306	

Ident	H1008	
Design Press. min. Chamber 1 / 2	-1 barg	-1 barg
Design Press. max. Chamber 1 / 2	60 barg	30 barg
Design Temp. min. Chamber 1 / 2	-45 °C	-45 °C
Design Temp. max. Chamber 1 / 2	100 °C	100 °C
Design Duty / Design Heat Transfer Area	313 kW	46.8 m²
Nominal Diameter / Tube Length	DN 800	2200 mm
MaterialTubes / Material Shell	1.4306	1.4308

Ident	P4711	
Design Press. Casing min.	-0.5 barg	
Design Press. Casing max.	60 barg	
Design Temp. Casing min.	-45 °C	
Design Temp. Casing max.	80 °C	
Design Capacity / Design Press. Head	200 m³/h	10 m
Design Speed / Design Power max. Diam. Wheel	600 1/min	60 kW
Material Case Press. Side / Material Impeller	1.4306	1.4308

Ident	P4712	
Design Press. Casing min.	-1 barg	
Design Press. Casing max.	60 barg	
Design Temp. Casing min.	-45 °C	
Design Temp. Casing max.	100 °C	
Design Capacity / Design Press. Head	420 m³/h	40 m
Design Speed / Design Power max. Diam. Wheel	1400 1/min	84 kW
Material Case Press. Side / Material Impeller	1.4306	1.4305

Ident	T4750	
Design Press. min. Chamber 1 / 2	-0.1 barg	-0.1 barg
Design Press. max. Chamber 1 / 2	0.1 barg	0.05 barg
Design Temp. min. Chamber 1 / 2	-45 °C	-45 °C
Design Temp. max. Chamber 1 / 2	100 °C	100 °C
Nominal Diameter / Overall Height/Length	DN 20000	4 m
Nominal Capacity	26 m³	
Material Chamber 1 / 2	1.4306	1.4305