# Developing and Testing a Common Space Systems Ontology using the Ontological Modeling Language

**Hamilton E. Johnson**
**University of Alabama in Huntsville**
**320 Sparkman Drive**
**Huntsville, AL 35899**
**hamilton.johnson@uah.edu**

**L. Dale Thomas**
**Department of ISEEM**
**University of Alabama in Huntsville**
**320 Sparkman Drive**
**Huntsville, AL 35899**
**dale.thomas@uah.edu**

**Manuel J. Diaz**
**Advanced Concepts Office**
**Marshall Space Flight Center**
**Huntsville, AL 35812**
**manuel.j.diaz@nasa.gov**

*Abstract*—**This paper describes the development and testing of the initial version of a common space systems ontology (CoSSO) for use by the Advanced Concepts Office (ACO) at NASA's Marshall Space Flight Center. The ontology provides a shared conceptualization of concepts of interest to the ACO for modeling aerospace systems concepts in a pre-phase A context to aid with the transition to a more model-based paradigm. The ontological concepts and relations, as well as the anticipated use cases, were developed through interactions with the relevant subject matter experts at the ACO and implemented in the Ontological Modeling Language (OML). The ontology builds on the Basic Formal Ontology (BFO) and the Common Core Ontologies (CCO).**

**While most of the ontology is still in the initial stages, an Environmental Control and Life Support System (ECLSS) ontology is being built on top of the main CoSSO and heavily developed as a proof of concept. The ECLSS ontology is designed with different use cases in mind, namely predicting and diagnosing errors in ECLS systems on long-duration missions, with a focus on the Four-Bed $CO_2$ carbon dioxide scrubber currently on board the ISS. The ECLSS ontology is being developed in a similar manner to the CoSSO, and designed to be compatible with it. The current state of both ontologies is presented and discussed, along with plans for future development and testing.**

## TABLE OF CONTENTS

## 1. INTRODUCTION

A large amount of information is gathered and created during the development of a complex aerospace system. This information is in a variety of forms, including reports, CAD files, simulation results, etc. Model Based Systems Engineering seeks to allow information from disparate models to be integrated, allowing for more seamless development. Ontologies are central to this effort. Ontologies are a shared representation of the terms of interests in a given domain and how they interact, i.e. the relations between the terms. An ontology for a domain allows the output of various models to be combined together by converting them to the same set of definitions and restrictions, greatly improving the ability of the development team to coordinate and share information.

The Advanced Concepts Office (ACO) at Marshall Space Flight Center (MSFC) is interesting in developing an ontology for modeling aerospace systems concepts in a pre-phase A context. In addition, the Environmental and Life Support System (ECLSS) group at MSFC is interested in an ECLSS ontology which builds off of the ACO effort for use in ontologically modeling the Four Bed Carbon Dioxide scrubber ($4BCO_2$) to aid diagnostics. This paper presents the current state of this effort. Section 2 describes the ontology development methodology, including the desired use cases of the ontology. Sections 6 and 7 give an overview of the current state of the ACO and ECLSS ontologies.

## 2. ONTOLOGY DEVELOPMENT METHODOLOGY

A number of different ontology development methodologies have been proposed by different sources[1], [2], [3], [4], [5], [6], [7], [8]. Attributes of several of these were used to develop a methodology involving several steps, as shown in Fig. 1. These steps are run in parallel and feed into each other. The steps are summarized below.

### *Requirements Definition*

The requirement definition step outlines the motivation and scope of the ontology, including intended use cases and verification strategies.

### *Knowledge Elicitation*

In the knowledge elicitation step, knowledge from different sources including textbooks, documents, and domain experts is compiled and reconciled. This is done in parallel with the conceptualization and formalization steps.

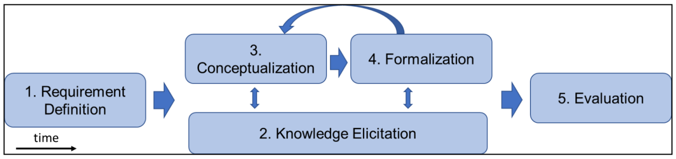


**Figure 1. Ontology development methodology**

*Conceptualization*

The conceptualization step consists of creating the ontology by first forming a taxonomy using the information gathered in the knowledge elicitation step and resolving any contradictions or incompatibilities which may arise. The terms in the taxonomy are defined in a glossary, and have simple ”is-a” or ”has-a” relationships between them. The axioms of the ontology then have to be defined. These contain information about the relations between different terms as well as their object properties.

*Formalization*

Once a conceptualization is defined, it can be transferred into a formal ontology language. The ontology language and tool must first be determined. After this, the definitions of classes, relation properties, data properties, and axioms can be established based on the conceptualization step. The result of this step is a machine-readable ontology.

*Evaluation*

Once the ontology is in a somewhat usable state, it can be evaluated. A few methods of evaluation have been performed or are planned. The simplest is modeling a system using the current version of the taxonomy/ontology to test for completeness in the required domain, with the results being used to inform the knowledge elicitation, conceptualization, and formalization. As the ontology nears completion, competency questions will be used to ensure it meets the desired use cases.

Details of how these steps were applied for the ACO and ECLSS ontologies and the results of the steps are given in the following sections.

## 3. REQUIREMENTS DEFINITION

The requirements for the ontology were determined through interactions with the primary stakeholders at MSFC. It was decided to model terms of interest for modeling concepts of aerospace systems in a pre-Phase A context. The two envisioned use cases are given below. Knowledge elicitation and verification plans were also developed during this step.

The ACO has several use cases in mind for the ontology. These were divided into priority one and two use cases, as shown in Figs. 2 and 3, where the priority one use cases are the initial targets for the ontology to meet, and the priority two use cases are envisioned for future development. The priority one use case allows the system architect to generate a SysML model of the system which is consistent with the ontology, while disciplinarians use the ontology to exchange, annotate, and query data.

The priority two use case includes using the ontology as a schema for a database which can be used to store mission concepts, allowing previous concepts to be queried, compared, and reused. This would allow previous mission concepts to be used as a starting point if they have similar requirements or restrictions.

## 4. KNOWLEDGE ELICITATION

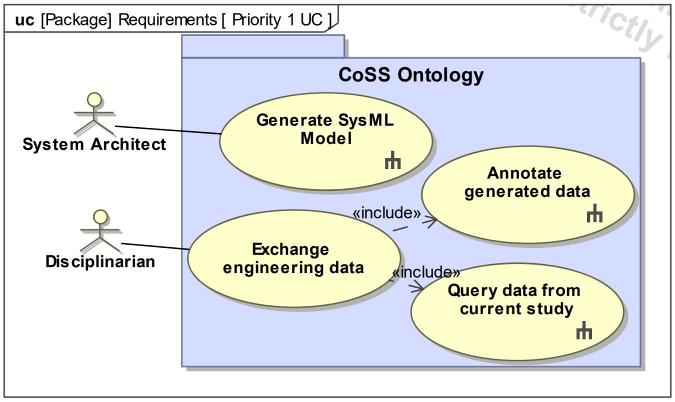


**Figure 2**. **Priority one use case**

The knowledge elicitation plan is shown in Fig. 4. It consists of first preemptively building a taxonomy using terms and concepts found through discussions and technical meetings with the stakeholders and other sources. Classes pertaining to a specific domain are then grouped together into different ”views,” which are then discussed with Subject Matter Experts (SMEs) and modified to fit their conceptualization.

During an ontology survey, it was determined that no publicly available ontology met ACO’s needs. The Basic Formal Ontology (BFO)[2] and Common Core Ontologies (CCO)[9] were selected as top-level and mid-level ontologies, respectively. Using these and various references, including spacecraft design references and past studies, a draft taxonomy was created.

The taxonomy was used to create a set of views corresponding to the main technical domains at the ACO: Avionics, Configuration, Mission Analysis, Power, Propulsion, Structures, Thermal, Nuclear Propulsion, and Mass Properties. These views were discussed with domain experts and amended to match their conceptualizations. Any contradiction with other views was noted to be evaluated during the reconciliation step of the knowledge elicitation plan.

## 5. CONCEPTUALIZATION

The conceptualization of the ACO ontology was done using SysML as a representation and discussion tool, while the conceptualization of the ECLSS ontology was done in the Ontological Modeling Language (OML), which is discussed in Section 6. In the SysML conceptualization, SysML entities and relationships are used to represent ontological entities and relationships — blocks for classes, generalizations for ”is-a” relationships, composition for ”has-a” relationships, and stereotypes extended from an association for any other object property. Examples of the resulting diagrams are shown in Figs. 5 and 6.

## 6. FORMALIZATION

The formalization process consists of taking the information represented in the SysML model and transferring it into a formal ontology language in an ontology editor. The first step was the selection of a formal ontology modeling language.

---

[2] https://basic-formal-ontology.org/

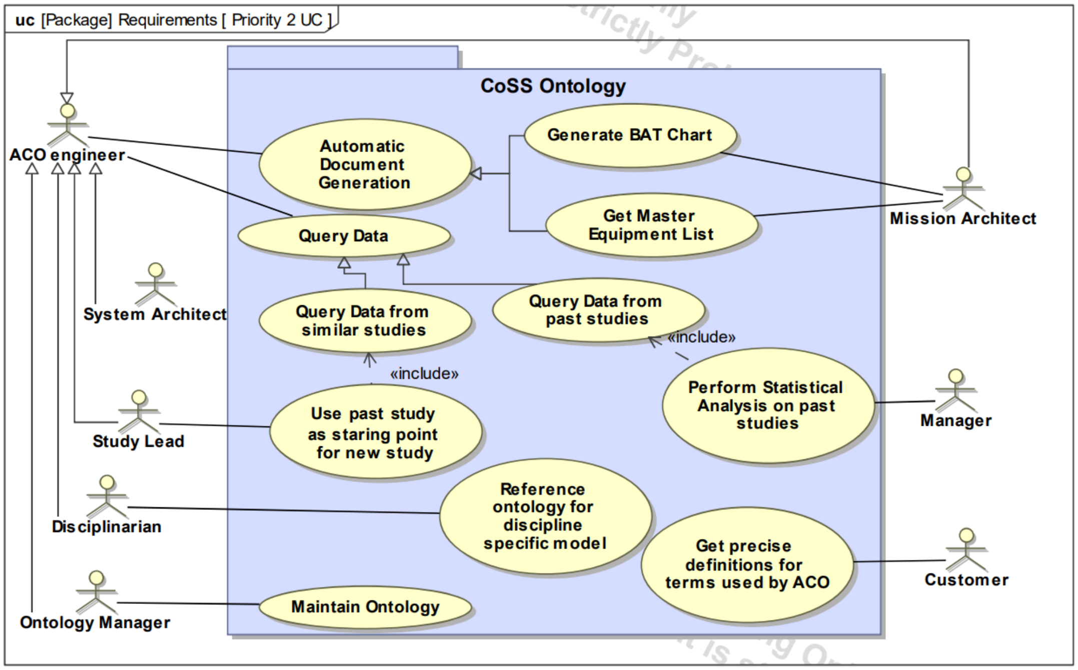


**Figure 3. Priority two use case**

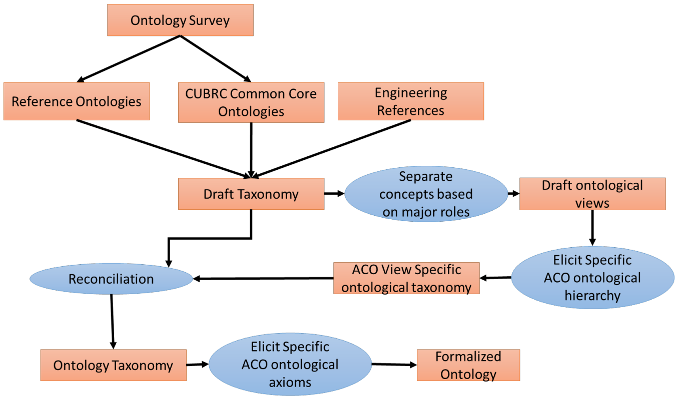


**Figure 4. Knowledge elicitation plan**

### *Ontology Modeling Language Selection*

A number of potential ontology modeling languages were identified, including the Web Ontology Language v2 (OWL 2), the Ontological Modeling Language (OML), the Shapes Constraint Language (SHACL)[3], DAML+OIL, and many others [10], [11], [12]. Out of these, OWL 2[4] and OML[5] were selected as the two most promising candidates — OWL 2 because it is the de facto standard ontological modeling language, as well as a W3C standard, and therefore has widespread resources and tool support available, and OML because it is built on OWL 2, but with some useful differences. While OWL 2 is widely used, it incorporates some assumptions less suited to systems engineering such as the open world assumption, which is useful in systems with incomplete information, but can cause reasoning to behave in a non-intuitive manner when dealing with a complete model of a system. OML, by contrast, is being developed specifically for use in aerospace systems by the OpenCAESAR group at NASA's Jet Propulsion Laboratory (JPL) [13], [14], [15]. Among its benefits are support for open or closed world assumptions and better support for versioning control using git, as well as envisioned use cases that match closely with those of the ACO[13]. OML also has the capability to output OWL 2 Description Logic (OWL 2 DL) ontologies, allowing for an easy transition to OWL if OML proves to be unsuitable. For these reasons, OML was selected for modeling the ontology.

### *Building the Ontologies*

The ontology was built by transitioning terms from the taxonomy to the OML version using OpenCAESAR's Rosetta workbench[6]. Figure 7 shows the OML version of the SysML diagram shown in Fig. 5. This transition has been completed using simple inheritance and containment relations, while more involved relations and data properties are still being investigated. In the ontology's current state, queries and reasoning engines can be used to determine a concept's ancestors or children, as well as whether a concept contains or is contained by any other concept. More complex reasoning will be possible as the relations and object properties are filled in.

Some more examples of the current version of the CoSSO can be seen in Figs. 8–9. A full listing of terms included in the ontology is given in Appendix A.

## 7. ECLSS ONTOLOGY

After the conceptualization of the ACO ontology was well under way, work began on the ECLSS ontology. This development followed the same steps as the overall ontology, but at a smaller scale. This process is summarized below.

---

[3] https://spinrdf.org/shacl-and-owl.html

[4] https://www.w3.org/TR/owl2-overview/

[5] https://github.com/opencaesar/oml

[6] https://github.com/opencaesar/oml-rosetta/

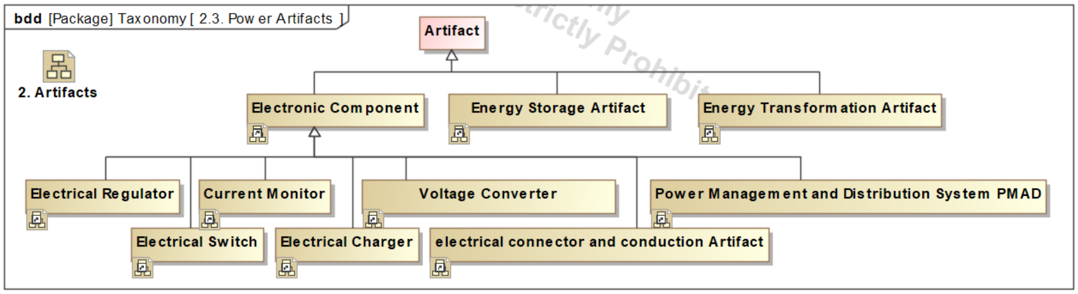


**Figure 5.** Top level power artifacts in the ACO taxonomy. Arrows represent ”is-a” relations.

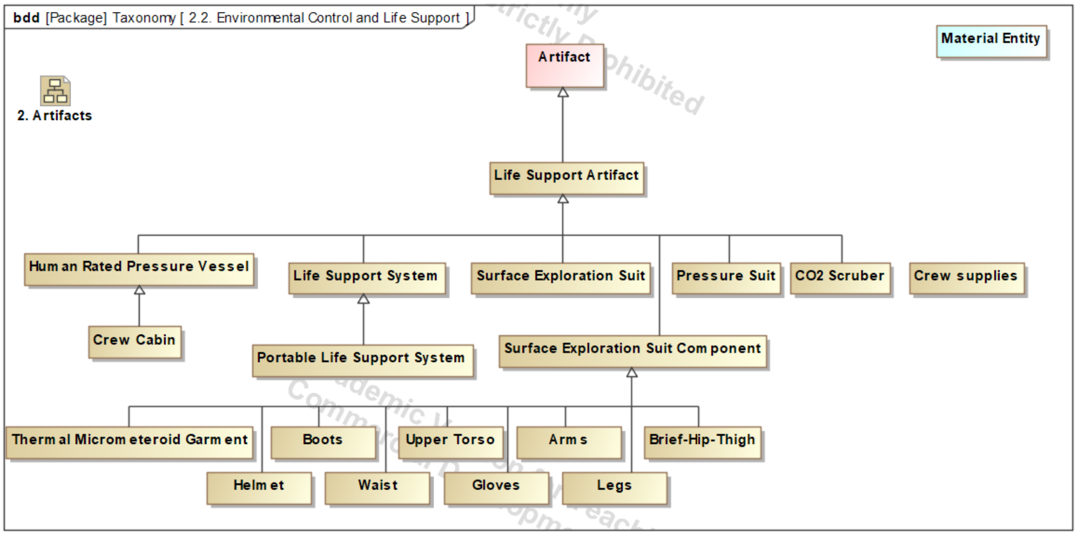


**Figure 6.** ECLSS artifacts in the ACO taxonomy, before expansion into the ECLSS ontology. Arrows represent ”is-a” relations.

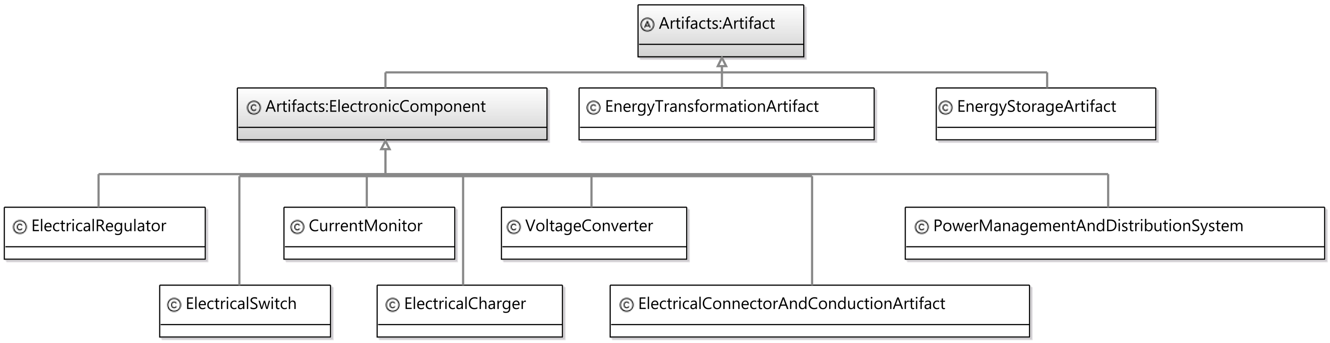


**Figure 7.** Top level power artifacts in the ACO ontology.

### *Requirements Definition*

The ECLSS ontology is a separate project which is envisioned as being a separate ontology built on top of the overall ACO ontology. The primary envisioned use case for the ECLSS ontology was defined through interactions with stakeholders as providing a model of the Four Bed $CO_2$ system ($4BCO_2$) [16] to aid diagnostics on long-duration crewed missions. It would do this by providing an ontological model of the $4BCO_2$ system, including possible errors and their relations to faults and system components. This is a test case to determine the utility of an overall ontology for ECLSS.

### *Knowledge Elicitation*

The ECLSS ontology was developed similarly to the ACO ontology, with the taxonomy focusing on the components needed to model the $4BCO_2$ system on the ISS as shown in [17]. Components which were already included in the ACO ontology were imported into the ECLSS ontology.

### *Conceptualization*

Since OML had already been selected as the ontological modeling language of choice, the conceptualization of the ECLSS ontology was done in OML. The terms gathered during the knowledge elicitation stage were arranged in a taxonomy involving simple inheritance and containment rela-

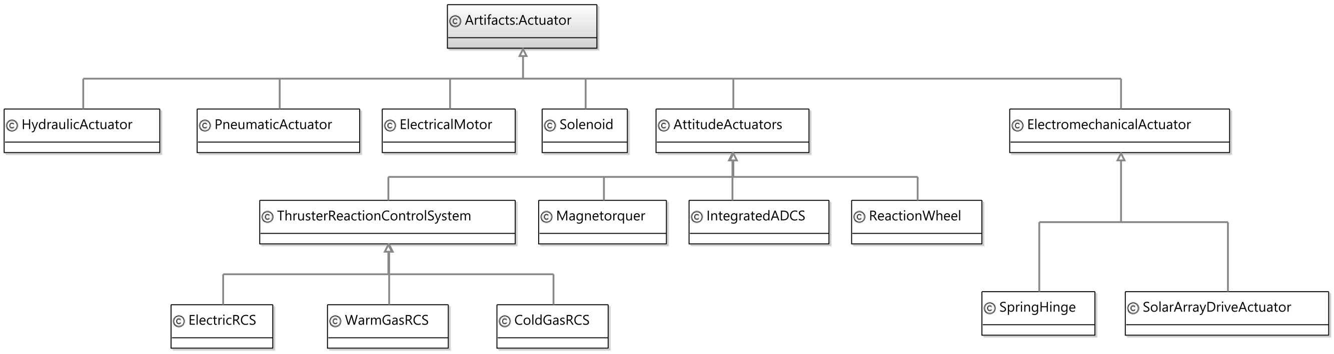


**Figure 8**. **Actuator view of the ACO ontology in OML.**

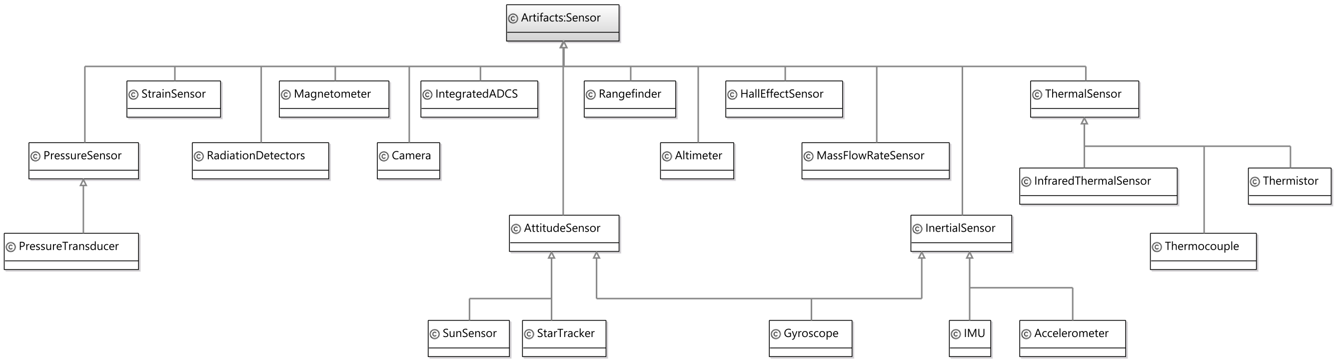


**Figure 9**. **Sensor view of the ACO ontology in OML.**

tions. Terms which overlapped with the ACO ontology were imported when possible, or arranged in such a way that they fit in with the overall hierarchy.

Appendix B shows the current version of the taxonomy. The hierarchy begins with *Life Support Artifact*, which is a child of *Artifact* from the overall ontology. From there the artifacts are split by subsystem into Water Revitalization, Atmospheric, etc., largely based on the systems on the ISS. Since the current focus of the ontology is modeling the $4BCO_2$ system, the Atmospheric system is the most heavily developed, going down to the $4BCO_2$ system, which is defined as containing all of its major component parts, as given in [17].

### *Formalization*

The formalization of the ECLSS ontology is underway. Since the conceptualization was done in OML, formalization mainly consists of deciding on and implementing the properties and relations between the concepts.

## 8. PLANNED EVALUATION METHODS

The evaluation plan for the ontology consists of first attempting to use it to model a diverse set of aerospace systems of interest to the ACO using data from previous ACO concepts. This will test whether the ontology includes all the terms and relationships required or if there are still gaps to be filled. The first systems to be modeled will be a nuclear thermal propulsion system and a space observatory. Later in the development competency questions will be finalized and used to evaluate the ontology's capabilities.

The initial version of the ECLSS ontology will be evaluated through comparison with known faults that the $4BCO_2$ system has experienced on orbit. The ontology and models built on it will be examined to determine whether they include all necessary information for diagnosing these errors and providing preliminary troubleshooting advice. An instance of the ISS ECLSS is in development (Fig. 10) for this purpose. The main physical components of the system have been added. Work is ongoing on modeling the faults and relations between components. A sample program using the ontology as a base will be developed and evaluated to confirm that the ontology achieves its goals.

## 9. CONCLUSIONS

An ontology including components and basic relations has been created for use by ACO. It is still in early development, but shows promise as a method of unifying ACO's development methodology. In addition, an ECLSS ontology has been developed as an extension of the ACO ontology to model the $4BCO_2$ system and related faults. This shows promise as a method of diagnosing faults in the $4BCO_2$ system. The model currently includes all the major components of the system as well as preliminary faults and relations.

There are two main limitations to the ontology as it currently stands. It is has not been fully tested yet via modeling example systems. This testing is the next step for the ontology and will help ensure it is complete and useful for modeling aerospace concepts.

It also does not fully conform to the standards for domain ontologies built on BFO and CCO[18], which limits the

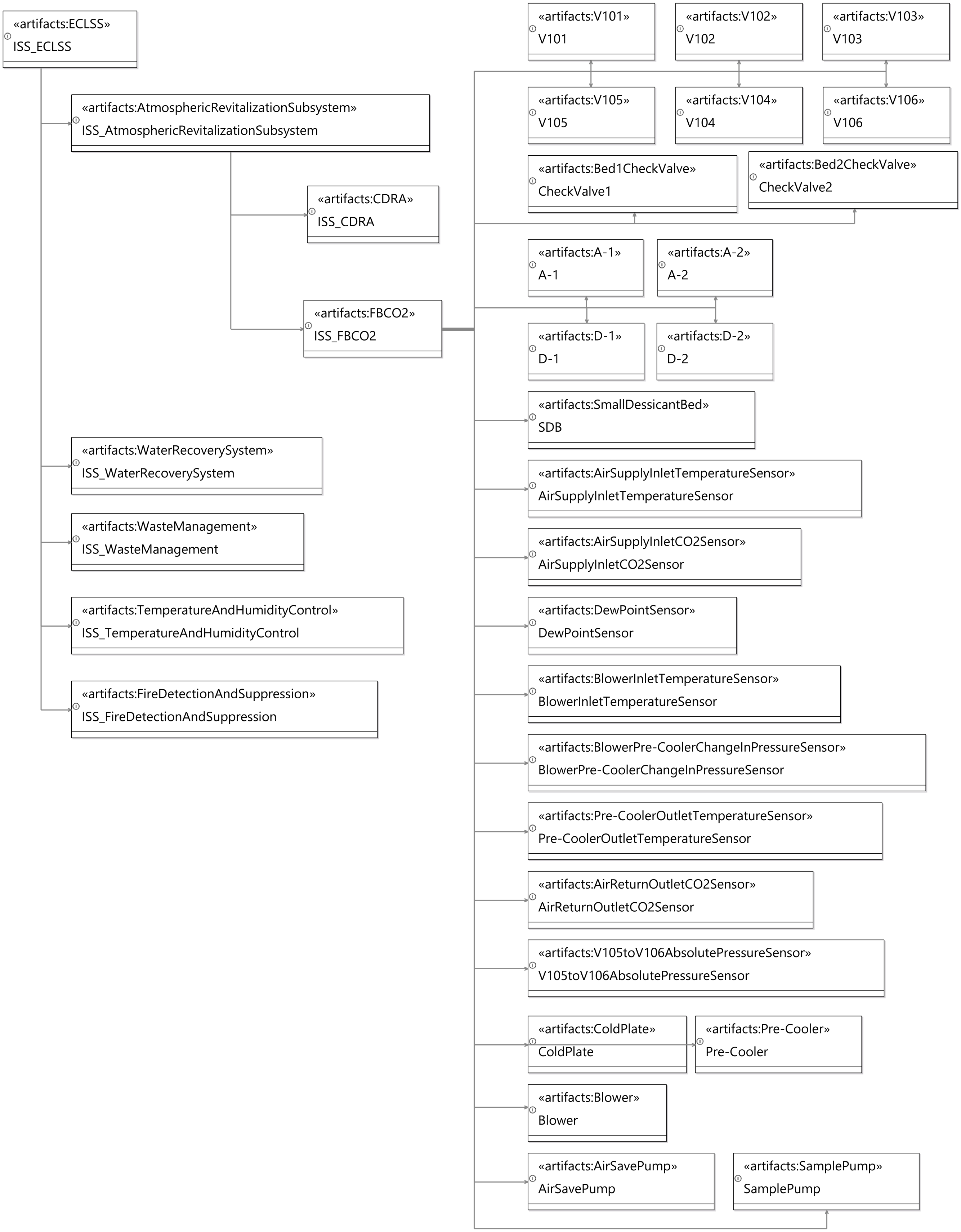


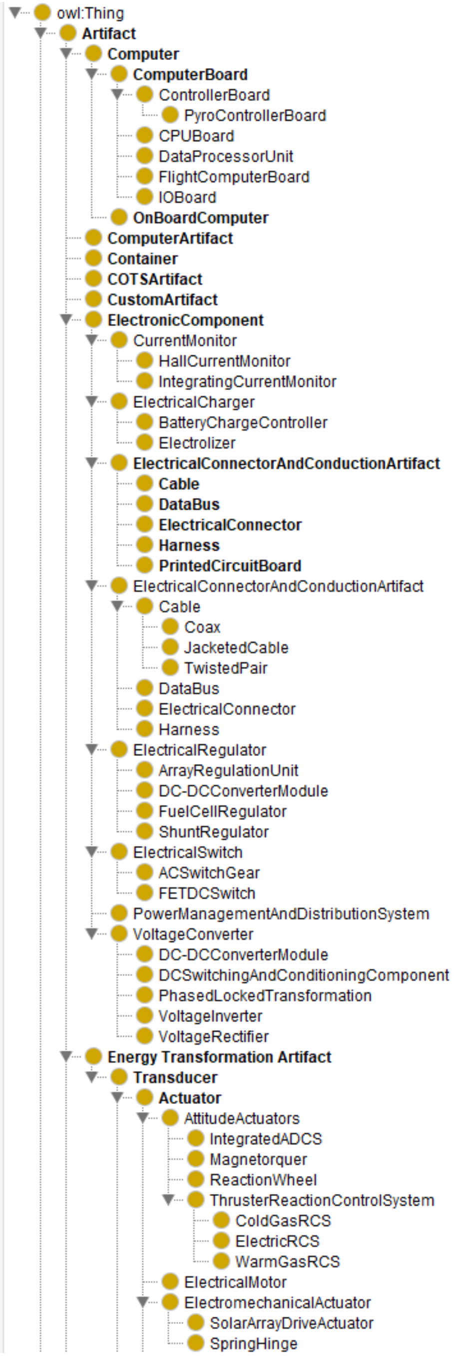
**Figure 10. Instance of the ISS ECLSS in OML. The arrows represent containment relations.**

potential for reuse outside ACO. The primary point where it diverges with the standard is the use of multiple inheritance for some classes. An investigation is underway to determine whether a refactor to remove the instances of multiple inheritance should be conducted. Reuse in the ECLSS ontology has not been hampered by this divergence from the standard, but reuse by other organizations could be.

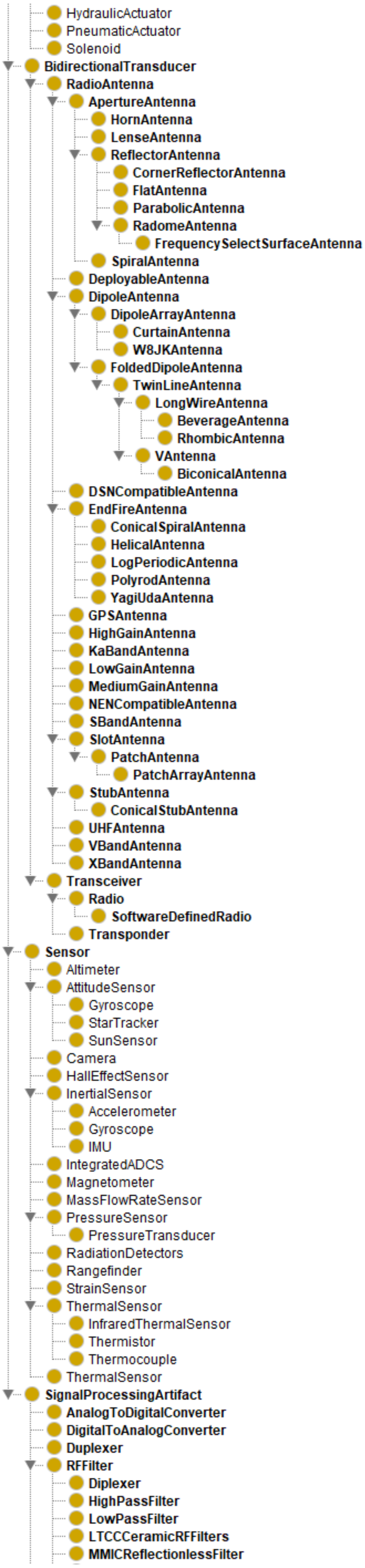
Beyond investigating these caveats, future work on the both ontologies includes expanding them by incorporating more involved relations between the various components and continuing to add relevant terms as they are identified.

# APPENDICES

## A. TERMS IN THE COSSO

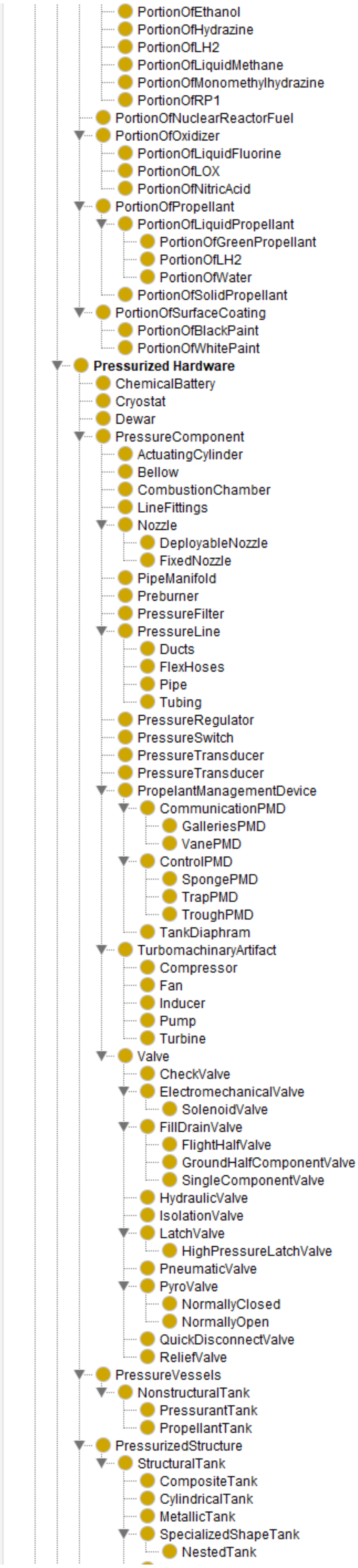


- owl:Thing
  - **Artifact**
    - **Computer**
      - **ComputerBoard**
        - ControllerBoard
          - PyroControllerBoard
        - CPUBoard
        - DataProcessorUnit
        - FlightComputerBoard
        - IOBoard
      - **OnBoardComputer**
    - **ComputerArtifact**
    - **Container**
    - **COTSArtifact**
    - **CustomArtifact**
    - **ElectronicComponent**
      - CurrentMonitor
        - HallCurrentMonitor
        - IntegratingCurrentMonitor
      - ElectricalCharger
        - BatteryChargeController
        - Electrolizer
      - **ElectricalConnectorAndConductionArtifact**
        - **Cable**
        - **DataBus**
        - **ElectricalConnector**
        - **Harness**
        - **PrintedCircuitBoard**
      - ElectricalConnectorAndConductionArtifact
        - Cable
          - Coax
          - JacketedCable
          - TwistedPair
        - DataBus
        - ElectricalConnector
        - Harness
      - ElectricalRegulator
        - ArrayRegulationUnit
        - DC-DCConverterModule
        - FuelCellRegulator
        - ShuntRegulator
      - ElectricalSwitch
        - ACSwitchGear
        - FETDCSwitch
      - PowerManagementAndDistributionSystem
      - VoltageConverter
        - DC-DCConverterModule
        - DCSwitchingAndConditioningComponent
        - PhasedLockedTransformation
        - VoltageInverter
        - VoltageRectifier
    - **Energy Transformation Artifact**
      - **Transducer**
        - **Actuator**
          - AttitudeActuators
            - IntegratedADCS
            - Magnetorquer
            - ReactionWheel
            - ThrusterReactionControlSystem
              - ColdGasRCS
              - ElectricRCS
              - WarmGasRCS
          - ElectricalMotor
          - ElectromechanicalActuator
            - SolarArrayDriveActuator
            - SpringHinge



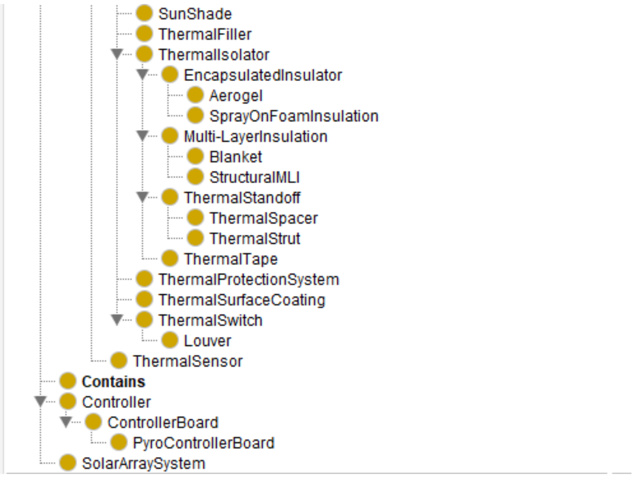
- HydraulicActuator
- PneumaticActuator
- Solenoid
- **BidirectionalTransducer**
  - **RadioAntenna**
    - **ApertureAntenna**
      - **HornAntenna**
      - **LenseAntenna**
      - **ReflectorAntenna**
        - **CornerReflectorAntenna**
        - **FlatAntenna**
        - **ParabolicAntenna**
        - **RadomeAntenna**
          - **FrequencySelectSurfaceAntenna**
      - **SpiralAntenna**
    - **DeployableAntenna**
    - **DipoleAntenna**
      - **DipoleArrayAntenna**
        - **CurtainAntenna**
        - **W8JKAntenna**
      - **FoldedDipoleAntenna**
        - **TwinLineAntenna**
          - **LongWireAntenna**
            - **BeverageAntenna**
            - **RhombicAntenna**
          - **VAntenna**
            - **BiconicalAntenna**
    - **DSNCompatibleAntenna**
    - **EndFireAntenna**
      - **ConicalSpiralAntenna**
      - **HelicalAntenna**
      - **LogPeriodicAntenna**
      - **PolyrodAntenna**
      - **YagiUdaAntenna**
    - **GPSAntenna**
    - **HighGainAntenna**
    - **KaBandAntenna**
    - **LowGainAntenna**
    - **MediumGainAntenna**
    - **NENCompatibleAntenna**
    - **SBandAntenna**
    - **SlotAntenna**
      - **PatchAntenna**
        - **PatchArrayAntenna**
    - **StubAntenna**
      - **ConicalStubAntenna**
    - **UHFAntenna**
    - **VBandAntenna**
    - **XBandAntenna**
  - **Transceiver**
    - **Radio**
      - **SoftwareDefinedRadio**
    - **Transponder**
- **Sensor**
  - Altimeter
  - AttitudeSensor
    - Gyroscope
    - StarTracker
    - SunSensor
  - Camera
  - HallEffectSensor
  - InertialSensor
    - Accelerometer
    - Gyroscope
    - IMU
  - IntegratedADCS
  - Magnetometer
  - MassFlowRateSensor
  - PressureSensor
    - PressureTransducer
  - RadiationDetectors
  - Rangefinder
  - StrainSensor
  - ThermalSensor
    - InfraredThermalSensor
    - Thermistor
    - Thermocouple
  - ThermalSensor
- **SignalProcessingArtifact**
  - **AnalogToDigitalConverter**
  - **DigitalToAnalogConverter**
  - **Duplexer**
  - **RFFilter**
    - **Diplexer**
    - **HighPassFilter**
    - **LowPassFilter**
    - **LTCCCeramicRFFilters**
    - **MMICReflectionlessFilter**

                - **RFBandpassFilter**
                - **Triplexer**
                - **WaveguideFilter**
            - **RFSwitch**
                - **CoaxialRFSwitch**
                    - **CoaxialDPDTSwitches**
                    - **CoaxialTSwitch**
                - **ReliantRFSwitch**
                - **SurfaceMountSwitch**
                - **WaveguideRFSwitch**
- **EnergyStorageArtifact**
- EnergyStorageArtifact
    - ElectricalEnergyStorageArtifact
        - ChemicalBattery
            - Li-FeBattery
            - Li-ionBattery
            - Li-poBattery
        - ChemicalBatteryPack
        - RegenerativeFuelCell
        - Supercapacitor
    - MechanicalEnergyStorageArtifact
        - MomentumStorageArtifact
            - ReactionWheel
    - ThermalEnergyStorageArtifact
        - ThermalBattery
        - ThermalStabilizer
- EnergyTransformationArtifact
    - PowerSource
        - ElectricalPowerSource
            - FuelCell
                - ProtonExchangeMembraneFuelCell
                - SolidOxideFuelCell
            - SolarArrayWing
            - SolarCell
            - ThermionicGenerator
            - ThermoelectricGenerator
                - BraytonGenerator
                - RankineGenerator
                - StirlingGenerator
        - HydraulicPowerSource
        - MechanicalPowerSource
        - PneumaticPowerSource
        - ThermalPowerSource
            - ChemicalReactor
            - NuclearReactor
- **EngineeredSystem**
- **Facility**
- **LifeSupportArtifact**
    - CO2Scrubber
    - HumanRatedPressureVessel
        - CrewCabin
    - LifeSupportSystem
        - PortableLifeSupportSystem
    - PressureSuit
    - SurfaceExplorationSuit
    - SurfaceExplorationSuitComponent
        - Arms
        - Boots
        - Brief-Hip-Thigh
        - Gloves
        - Helmet
        - Legs
        - ThermalMicrometeroidGarment
        - UpperTorso
        - Waist
- **LiftingSurface**
- **Mechanism**
    - ArticulationMechanism
        - SolarArrayMechanism
    - DeploymentMechanism
- **NuclearReactorComponent**
    - CriticalityControlSystem
        - DrumControlSystem
    - NeutronModerator
    - NuclearFuelElement
    - NuclearReactorStructure
        - DrumStructure
        - NuclearReactorShielding
            - NuclearReactorExternalShielding
            - NuclearReactorInternalShielding
    - ReactorWorkingFluidInterface
        - FlowChannel
        - FlowPlate
            - CylindricalFlowPlate
            - FlatFlowPlate
    - Reflector
- **PortionOfProcessedMaterial**
    - PortionOfFuel



        - PortionOfEthanol
        - PortionOfHydrazine
        - PortionOfLH2
        - PortionOfLiquidMethane
        - PortionOfMonomethylhydrazine
        - PortionOfRP1
    - PortionOfNuclearReactorFuel
    - PortionOfOxidizer
        - PortionOfLiquidFluorine
        - PortionOfLOX
        - PortionOfNitricAcid
    - PortionOfPropellant
        - PortionOfLiquidPropellant
            - PortionOfGreenPropellant
            - PortionOfLH2
            - PortionOfWater
        - PortionOfSolidPropellant
    - PortionOfSurfaceCoating
        - PortionOfBlackPaint
        - PortionOfWhitePaint
- **Pressurized Hardware**
    - ChemicalBattery
    - Cryostat
    - Dewar
    - PressureComponent
        - ActuatingCylinder
        - Bellow
        - CombustionChamber
        - LineFittings
        - Nozzle
            - DeployableNozzle
            - FixedNozzle
        - PipeManifold
        - Preburner
        - PressureFilter
        - PressureLine
            - Ducts
            - FlexHoses
            - Pipe
            - Tubing
        - PressureRegulator
        - PressureSwitch
        - PressureTransducer
        - PressureTransducer
        - PropelantManagementDevice
            - CommunicationPMD
                - GalleriesPMD
                - VanePMD
            - ControlPMD
                - SpongePMD
                - TrapPMD
                - TroughPMD
            - TankDiaphram
        - TurbomachinaryArtifact
            - Compressor
            - Fan
            - Inducer
            - Pump
            - Turbine
        - Valve
            - CheckValve
            - ElectromechanicalValve
                - SolenoidValve
            - FillDrainValve
                - FlightHalfValve
                - GroundHalfComponentValve
                - SingleComponentValve
            - HydraulicValve
            - IsolationValve
            - LatchValve
                - HighPressureLatchValve
            - PneumaticValve
            - PyroValve
                - NormallyClosed
                - NormallyOpen
            - QuickDisconnectValve
            - ReliefValve
    - PressureVessels
        - NonstructuralTank
            - PressurantTank
            - PropellantTank
    - PressurizedStructure
        - StructuralTank
            - CompositeTank
            - CylindricalTank
            - MetallicTank
            - SpecializedShapeTank
                - NestedTank

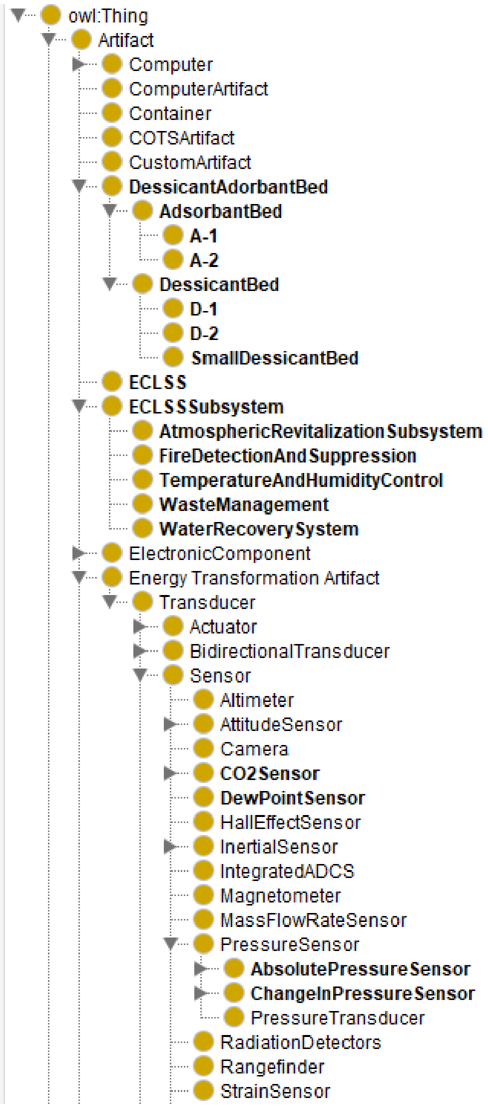
      - SphericalTank
  - SealedContainer
- **PropulsionSystem**
  - JetEngine
    - AirbreathingJetEngine
    - RocketEngine
      - ChemicalRocket
        - LiquidEngine
          - BipropellantEngine
            - CryoBiprop
              - FluorineEngine
              - LOXHydrogen
              - LOXMethane
              - LOXRP1
            - HypergolBiprop
              - HydrazineNTO
              - MMHNTO
          - CryogenicallyFueledRocketEngine
            - CryoBiprop
              - FluorineEngine
              - LOXHydrogen
              - LOXMethane
              - LOXRP1
          - HypergolicallyFueledRocketEngine
          - MonopropellantEngine
            - GreenPropellantEngine
            - HydrazineEngine
            - HydrogenPeroxideEngine
          - PressureFedLiquidRocketEngine
            - BlowDown
            - PressureStabilized
          - PumpFedLiquidRocketEngine
      - ColdGasPropulsionSystem
      - ElectricalThruster
        - ElectromagneticThruster
          - MagnetoplasmadynamicThruster
          - PulsePlasmaThruster
        - ElectrostaticThruster
          - ElectrosprayThruster
          - GriddedIonThruster
            - HighPowerElectricPropulsion
            - NASASolarTechnologyApplicationReadiness_NSTAR
            - RadioFrequencyIonThruster
          - HallThruster
        - ElectrothermalThruster
          - Arcjet
          - MicrowaveThruster
          - Resistojet
          - VariableSpecificImpulseMagnetoplasmaRocket_VASIMR
      - NuclearPropulsionSystem
    - TVCEngine
  - PoweredWheels
  - RotorSystem
  - SolarSail
- **Scintillator**
- **StructuralArtifact**
  - PrimaryStructure
    - SolarArrayStructure
    - SolarArrayWingStructure
  - SecondaryStructure
  - Shielding
  - StructuralTank
  - Truss
  - TrussMember
    - Strut
    - Tie
- **ThermalManagementComponent**
  - HeatRejectionAndCollection
    - Heater
    - PhaseChangeDevice
      - Sublimator
    - Radiator
  - HeatTransportDevice
    - ColdPlate
    - Doubler
    - FluidLoop
      - MechanicallyPumpedFluidLoop
    - HeatExchanger
    - HeatPipe
      - CapillaryPumpedLoop
      - LoopHeatPipe
    - HeatPump
      - FluidHeatPump
        - Cryocooler
      - ThermoelectricHeatPump
    - ThermalStrap
  - ThermalControlSystem
  - ThermalInterfaceComponent



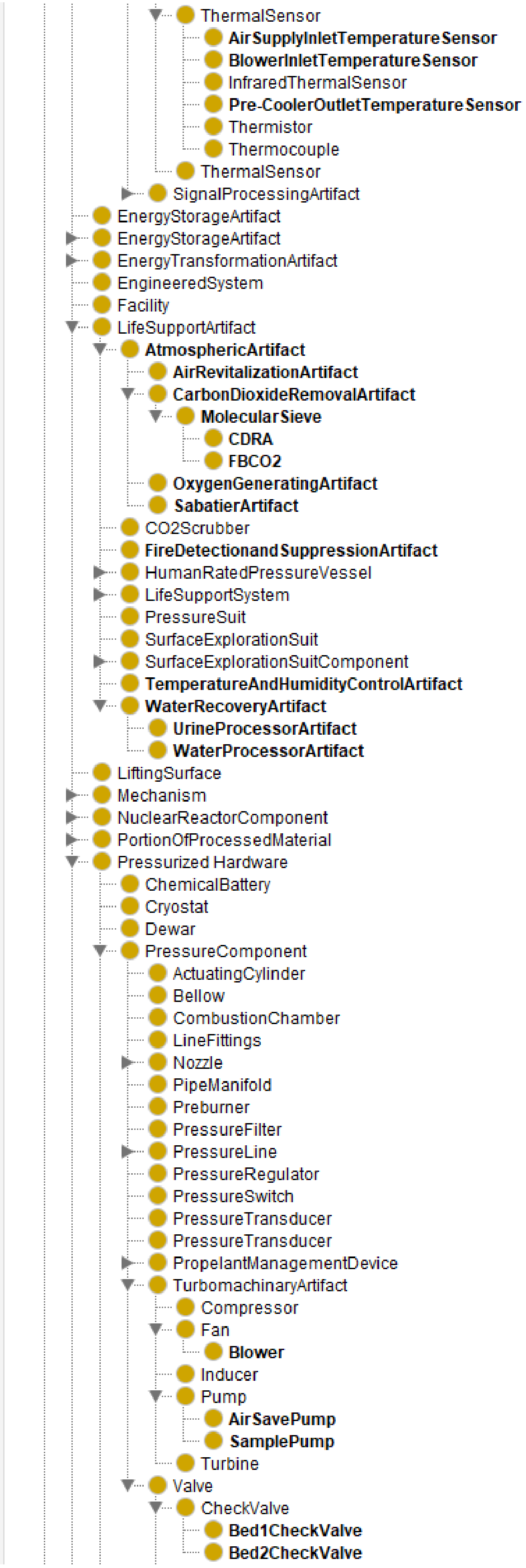
      - SunShade
      - ThermalFiller
      - ThermalIsolator
        - EncapsulatedInsulator
          - Aerogel
          - SprayOnFoamInsulation
        - Multi-LayerInsulation
          - Blanket
          - StructuralMLI
        - ThermalStandoff
          - ThermalSpacer
          - ThermalStrut
        - ThermalTape
      - ThermalProtectionSystem
      - ThermalSurfaceCoating
      - ThermalSwitch
        - Louver
    - ThermalSensor
- **Contains**
- Controller
  - ControllerBoard
    - PyroControllerBoard
- SolarArraySystem

## B. Terms in the ECLSS Ontology

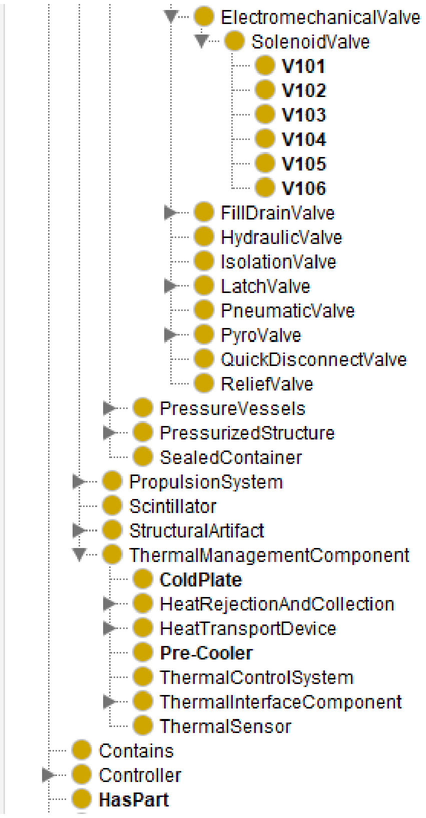
Terms in bold are defined in the ECLSS ontology, while the rest are imports from the ACO ontology.



- owl:Thing
  - Artifact
    - Computer
    - ComputerArtifact
    - Container
    - COTSArtifact
    - CustomArtifact
    - **DessicantAdorbantBed**
      - **AdsorbantBed**
        - **A-1**
        - **A-2**
      - **DessicantBed**
        - **D-1**
        - **D-2**
        - **SmallDessicantBed**
    - **ECLSS**
    - **ECLSSSubsystem**
      - **AtmosphericRevitalizationSubsystem**
      - **FireDetectionAndSuppression**
      - **TemperatureAndHumidityControl**
      - **WasteManagement**
      - **WaterRecoverySystem**
    - ElectronicComponent
    - Energy Transformation Artifact
      - Transducer
        - Actuator
        - BidirectionalTransducer
        - Sensor
          - Altimeter
          - AttitudeSensor
          - Camera
          - **CO2Sensor**
          - **DewPointSensor**
          - HallEffectSensor
          - InertialSensor
          - IntegratedADCS
          - Magnetometer
          - MassFlowRateSensor
          - PressureSensor
            - **AbsolutePressureSensor**
            - **ChangeInPressureSensor**
            - PressureTransducer
          - RadiationDetectors
          - Rangefinder
          - StrainSensor



ThermalSensor
**AirSupplyInletTemperatureSensor**
**BlowerInletTemperatureSensor**
InfraredThermalSensor
**Pre-CoolerOutletTemperatureSensor**
Thermistor
Thermocouple
ThermalSensor
SignalProcessingArtifact
EnergyStorageArtifact
EnergyStorageArtifact
EnergyTransformationArtifact
EngineeredSystem
Facility
LifeSupportArtifact
**AtmosphericArtifact**
**AirRevitalizationArtifact**
**CarbonDioxideRemovalArtifact**
**MolecularSieve**
**CDRA**
**FBCO2**
**OxygenGeneratingArtifact**
**SabatierArtifact**
CO2Scrubber
**FireDetectionandSuppressionArtifact**
HumanRatedPressureVessel
LifeSupportSystem
PressureSuit
SurfaceExplorationSuit
SurfaceExplorationSuitComponent
**TemperatureAndHumidityControlArtifact**
**WaterRecoveryArtifact**
**UrineProcessorArtifact**
**WaterProcessorArtifact**
LiftingSurface
Mechanism
NuclearReactorComponent
PortionOfProcessedMaterial
Pressurized Hardware
ChemicalBattery
Cryostat
Dewar
PressureComponent
ActuatingCylinder
Bellow
CombustionChamber
LineFittings
Nozzle
PipeManifold
Preburner
PressureFilter
PressureLine
PressureRegulator
PressureSwitch
PressureTransducer
PressureTransducer
PropelantManagementDevice
TurbomachinaryArtifact
Compressor
Fan
**Blower**
Inducer
Pump
**AirSavePump**
**SamplePump**
Turbine
Valve
CheckValve
**Bed1CheckValve**
**Bed2CheckValve**



ElectromechanicalValve
SolenoidValve
**V101**
**V102**
**V103**
**V104**
**V105**
**V106**
FillDrainValve
HydraulicValve
IsolationValve
LatchValve
PneumaticValve
PyroValve
QuickDisconnectValve
ReliefValve
PressureVessels
PressurizedStructure
SealedContainer
PropulsionSystem
Scintillator
StructuralArtifact
ThermalManagementComponent
**ColdPlate**
HeatRejectionAndCollection
HeatTransportDevice
**Pre-Cooler**
ThermalControlSystem
ThermalInterfaceComponent
ThermalSensor
Contains
Controller
**HasPart**

## ACKNOWLEDGMENTS

The authors thank the Advanced Concepts Office at NASA's Marshall Space Flight Center for funding this project.

## REFERENCES

[1] R. Iqbal, M. A. A. Murad, A. Mustapha, and N. M. Sharef, "An analysis of ontology engineering methodologies," *Research Journal of Applied Sciences, Engineering and Technology*, vol. 6, no. 16, pp. 2993–3000, 2013.

[2] F. López, "Overview of methodologies for building ontologies," *Proceedings of the IJCAI-99 workshop on Ontologies and Problem-Solving Methods*, p. 13, 1999.

[3] A. De Nicola, M. Missikoff, and R. Navigli, "A proposal for a unified process for ontology building: Upon," in *Database and Expert Systems Applications*, K. V. Andersen, J. Debenham, and R. Wagner, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 655–664.

[4] A. D. Nicola, M. Missikoff, and R. Navigli, "A software engineering approach to ontology building," *Information Systems*, vol. 34, no. 2, pp. 258–275, 2009.

[5] M. Fernández-López, A. Gomez-Perez, and N. Juristo, "Methontology: from ontological art towards ontological engineering," *Engineering Workshop on Ontological Engineering (AAAI97)*, 03 1997.

[6] N. F. Noy and D. L. McGuinness, "Ontology development 101: A guide to creating your first ontology," Online, retrieved: May 29, 2022. [Online]. Available: https://protege.stanford.edu/publications/ontologydevelopment/ontology101.pdf

[7] M. Bravo, L. F. H. Reyes, and J. A. R. Ortiz, "Methodology for ontology design and construction," *Contaduría y Administración*, no. 64, p. 24, 2019.

[8] R. Arp, B. Smith, and A. Spear, *Building Ontologies with Basic Formal Ontology*. The MIT Press, 2015.

[9] R. Rudnicki, "An overview of the common core ontologies." 4455 Genesee St., Buffalo, NY 14225: CUBRC, Inc., 2019. [Online]. Available: https://www.nist.gov/system/files/documents/2021/10/14/nist-ai-rfi-cubrc_inc_004.pdf

[10] D. Kalibatiene and O. Vasilecas, "Survey on ontology languages," vol. 90, 01 2011, pp. 124–141.

[11] A. Fensel and U. Keller, "Choosing an ontology language." 01 2005, pp. 47–50.

[12] V. Maniraj and S. Ramakrishnan, "Ontology languages – a review," *International Journal of Computer Theory and Engineering*, vol. 2, pp. 887–891, 01 2010.

[13] D. Wagner, S. Kim, A. Jimemez, M. Elaasar, N. Rouquette, and S. Jenkins, "Caesar model-based approach to harness design," in *2020 IEEE Aerospace Conference*. IEEE, 2020, pp. 1–13.

[14] S. Jenkinis, "Ontological modeling language: Origin and rationale," 2021. [Online]. Available: https://www.opencaesar.io/imce/2021/06/19/OML-Origin-and-Rationale.html

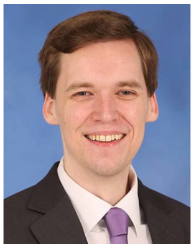
[15] H. P. de Koning, "Learning from nasa jpl's ontology development using owl and oml," in *Space Systems Ontology – Brainstorming Workshop*, 2019. [Online]. Available: https://indico.esa.int/event/310/contributions/4581/attachments/3514/4653/Presentation_-_1430_-_Hans-Peter_de_g.pdf

[16] G. Cmarik, W. Peters, and J. Knox, "4-bed co2 scrubber — from design to build," *2020 International Conference on Environmental Systems*, 07 2020. [Online]. Available: https://hdl.handle.net/2346/86440

[17] G. Cmarik, J. Knox, and J. Garr, "Progress of four bed carbon dioxide scrubber," *2022 International Conference on Environmental Systems*, 07 2022. [Online]. Available: https://hdl.handle.net/2346/89582

[18] R. Rudnicki, B. Smith, T. Malyuta, and W. Mandrick, "Best practices of ontology development," 11 2016. [Online]. Available: https://www.nist.gov/system/files/documents/2021/10/14/nist-ai-rfi-cubrc_inc_002.pdf

## BIOGRAPHY



***Hamilton Johnson*** *received his B.S. in Physics and Computer Science at Ouachita Baptist University and his M.S. in Aerospace Systems Engineering at the University of Alabama in Huntsville, where he is currently pursuing a Ph.D. in Aerospace Systems Engineering. He performs research at the Complex Systems Integration Laboratory, focusing on the application of MBSE to modeling aerospace systems concepts.*

***L. Dale Thomas*** *currently serves as a Professor and Eminent Scholar of Systems Engineering in the Department of Industrial and Systems Engineering and Engineering Management at the University of Alabama in Huntsville (UAH). He teaches system engineering students in the art and science of systems architecture and design, systems integration, test, and verification, and systems management. Dale also serves as director of the Alabama Space Grant Consortium and as deputy director of the UAH Propulsion Research Center. Prior to his retirement from NASA in July 2015, Dale served as the Associate Center Director (Technical) for the NASA Marshall Space Flight Center (MSFC) in Huntsville, Alabama, providing technical leadership for all MSFC spaceflight projects. He had previously served as the NASA Constellation Program Manager, leading the Constellation Program Office at Johnson Space Center in Houston, Texas, leading a nationwide team including all NASA field centers and five prime contractors.*

***Manuel J. Diaz*** *is an Aerospace Engineer at the Advanced Concepts Office at NASA MSFC. Mr. Diaz performs studies for the conceptual design and analysis of space systems and manages several investments in model-based and digital engineering capabilities and technologies to push the paradigm of conceptual design. Mr. Diaz holds the roles of the Research and Technology lead for the NASA Safety and Engineering Center's (NESC) Systems Engineering (SE) Technical Discipline Team (TDT), as well as serves as MSFC's Small Business Technical Coordinator, Advanced Concepts. Mr. Diaz is currently a Ph.D. candidate at the Georgia Institute of Technology. He holds an MS and BS in Aerospace Engineering also from the Georgia Institute of Technology. He is a Member of AIAA and a member of the Space Integration and Digital Engineering Integration and Outreach Committees.*