SIEMENS

Teamcenter Gateway for SAP S/4HANA Configuration Guide

Contents

Document Transfer - Document Info Record

Product Structure Transfer - Bill Of Material

Glossary   A-1

Preface

This documentation cannot be used as a substitute for consulting advice, because it can never consider the individual business processes and configuration. Despite our best efforts it is probable that some information about functionality and coherence may be incomplete.

Issue: July 2018

Legal notice:

Trademark notice:

1. Introduction

The Teamcenter Gateway for SAP S/4HANA **(T4S4)** software solution is a general purpose integration software that provides data and process integration between Teamcenter® by Siemens Product Lifecycle Management Software Inc. and SAP S/4HANA® by SAP AG.

T4S4 provides a wide range of interactive and workflow functions to transfer and synchronize data between Teamcenter and SAP S/4HANA®.

2. Basic Configuration

2.1   The File t4s_mapping_config.sd

The file *t4s_mapping_config.sd* in the directory *<GS_ROOT>\var\mmap\t4s_mapping_config* is the first file that is read by the T4S4 software in the context with mapping functionality. It may contain basic settings as well as variables which are supposed to be used in more than one mapping file (see how to define the direct mapping of constants in the **Teamcenter Gateway - Generic Configuration Guide**).

> Caution:
>
> - The file *t4s_mapping_config.sd* must not be renamed as T4S4 will read it and check there which mapping files to use.
>
> - The generic file *t4s_mapping_config.sd* is only read at T4S4 start up, not in every transaction. Therefore, it is impossible evaluating anything there that changes during T4S4 runtime, because the result will be the same in every run. So for example if you want to get a time stamp and define that in this file, the content of this variable will be the same every time you use it. Please use the object type specific mapping files for this kind of variables. Some Teamcenter related data is not available at runtime, for example the TC DB name: `::TcData(SiteIdName)`

Generic Configuration File t4s_mapping_config.sd

The following basic configuration data can be set in generic mapping configuration file *t4s_mapping_config.sd*:

- System login data, see chapter **Provide SAP Account Information**

- One of the paragraphs in this file is for telling T4S4 which files contain the customer accessible mapping functions.
  Example:

  ```
  source -relax t4s_mm_mapping_template.sd
  ```

  This means T4S4 has to read the content of the file *t4s_mm_mapping_template.sd*. The argument `–relax` means the file is looked up in the memory first and only if its content is not there, the file is loaded from disc (only in the same directory *<GS_ROOT>\var\mmap\t4s_mapping_config*).
  In principle, the file names may be modified freely as long as the file extension *.sd* is kept. Every file stated there that has the file extension *.sd* is actually used for the mapping functionality. In fact the only file T4S4 actually uses for the mapping is the compiled mapping file *t4s_mapping_config.rfdt*. So this is what you need to think about in order to create the compiled mapping file correctly.
  In order to not use a mapping file it is enough to not "source" it in *t4s_mapping_config.sd*. However, we strongly recommend keeping only those mapping files in the *mmap* directory that you really want to use!

> Caution:
>
> Be sure to have the correct file names (the files located in *<GS_ROOT>\var\mmap\t4s_mapping_config*) in the "source" section of this file

- One of the last sections of this file is for sourcing the TCL language files and for activating the preferred language:

```
set strLanguageCode "en_US" ; # default language
::msgcat::mclocale $strLanguageCode
```

In that example, the language is defined in variable `strLanguageCode`. Please check the file names in *<GS_ROOT>/var/lang/T4S/* for getting all supported language codes. The value of variable `strLanguageCode` is overwritten by the value of environment variable `TP_LANGUAGE_CODE`, if defined.

2.2   Provide SAP Account Information

This functionality may be used in order to be able to work without having to enter SAP login data; it is also known as SAP Auto Login.

The settings described here have to be done in the file *t4s_mapping_config.sd*.

Use Cases

- Storing SAP Account Information in Plain Text Format

- Storing SAP Account Information for More Than One SAP Account to Be Used in a Workflow

- Storing SAP Account Information in Encrypted Format

> Caution:
>
> - The SAP Auto Login line must not contain comment signs (#) nor any additional characters.
>
> - Make sure to pair any opening and closing square brackets [ ] properly.
>
> - Do not use a dollar sign in any password because TCL will try and replace the following part by a variable content that does not exist most probably.
>
> - The defined SAP Auto Login is always used for T4x scripts and jobs, even if you did another login (especially another user) by the test login script for ITK or SAP, respectively.
>
> - If any of the strings contains special characters (most likely for the password), enclose the whole string in quotation marks.

Storing SAP Account Information in Plain Text Format

T4S4 Configuration

Use function `::T4S::CONNECTION2SAP::setConnectionInfoPlain2` in order to define a SAP account and tell T4S4 to use it; see the **T4S4 API Reference** for details.

```
# Example for complete Auto Login
::T4S::CONNECTION2SAP::setConnectionInfoPlain2 ET1 800 USER PASSWD EN
::T4S::CONNECTION2SAP::setConnectionInfoPlain2 ET1 800 USER PASSWD EN
SCRIPTING
::T4S::CONNECTION2SAP::selectActiveConnection2SAP "*" ET1 800
```

The call of function `::T4S::CONNECTION2SAP::setConnectionInfoPlain2` defines the connection to SAP; more than one can be defined in the mapping.

The call of function `::T4S::CONNECTION2SAP::selectActiveConnection2SAP` defines which SAP connection T4S4 should use by default, if not specified otherwise (e.g. using a workflow handler argument). It has to be set in order to make sure that this SAP connection should be used by default.

Since the Teamcenter account information should not be accessible to anybody, we recommend storing the account information in encrypted form.

> Caution:
>
> - Only the parameter `Mode` is optional. If omitted, this SAP connection is used for everything but `SCRIPTING` (see below).
>
> - Be sure to write all values in upper case letters (unless the `System` is really defined with lower case letters which is usually not the case).
>
> - If you need more than one SAP Auto Login (e.g. `WORKFLOW` for one user and `SCRIPTING` for another one), define them in separate lines.
>
> - In any case (one or more accounts defined), you have to define the "active" SAP connection (last line in the example above). This one will be used automatically if nothing else is specified, e.g. by manual selection or workflow handler argument.
>
> - Using this Auto Login means that many SAP transactions will be done with the same SAP user (also known as "Technical user", as defined here) although they are triggered from different human users. Please make sure that you are in compliance with SAP licensing requirements when you define communication via a Technical user. Setting up such communication is not meant to circumvent any licensing requirements per your software contract with SAP; it is only meant for user convenience.

The parameter `Mode` determines for which transaction types this account information should be used. The possible key words are as follows:

- `PORTAL`: use this for all SAP transactions where a user the T4S Gateway Menu to initiate a SAP transfer or is using the SAP DataView. Alternatively, you may specify a separate SAP login for handling each portal transaction on each object type separately (the key words are always `PORTAL <object_type> <action>`), e.g.:

Example	Autologin for
PORTAL <object_type> DISPLAY	<object_type> display
PORTAL <object_type> CREATE_HID	<object_type> creation
PORTAL <object_type> CHANGE_HID	<object_type> change
PORTAL BOM DISPLAY	BOM display
PORTAL * SAPDATA_VIEW	all itemtypes for SAP Dataview

- `WORKFLOW`: use this for all T4S4 transactions from a Workflow, see below

- `SCRIPTING`: use this for the T4S4 Admin GUI scripts
  → The scripts using the T4S4 job functionality need `BATCH` instead, see above

- `BATCH`: use this for T4S4 transactions launched from the job server, including T4S4 Admin GUI scripts using the T4S4 job functionality (each script with the word "job" in its name)

- QUERY: use this for the T4S4 user query functionality. In order to use it for specific queries only, add the query name to this call (else it is used for any T4S4 user query), e.g.:

```
::T4S::CONNECTION2SAP::setConnectionInfoPlain2 \
ET1 800 user passwd language QUERY T4S_DEMO_QUERY
```

- (parameter omitted): this is a combination of all the modes besides SCRIPTING

Please consider the following specifics for parameter Mode:

- If you are in doubt which mode is the correct one for your desired functionality, start trying without the last parameter for testing.

- If the value for Mode is none of the listed ones, T4S4 will ignore the line. So if it does not work as expected, please check exactly the number and position of each parameter value and the spelling.

- Any mode that is not covered by such an SAP Auto Login definition will require a manual SAP login. Thus you may have to login manually to the SAP although there was a successful transaction before.

- Those SAP Auto Logins are stored in the tpapps process, only the one for SCRIPTING is stored on the BGS. Therefore, after removing the SCRIPTING SAP Auto Login from the mapping and restarting the GS, T4S4 will not use them any more besides the one for SCRIPTING. This will still be available for the Admin GUI scripts as long as the BGS is running or it is deleted explicitly. This may be done in the test script SAP connection test with the parameter Login set to Delete, or in the mapping:

```
::T4S::CONNECTION2SAP::deleteConnectionInfo $Dest $Client SCRIPTING
```

- When only a SCRIPTING connection is defined (no other modes), you may omit the additional line with selectActiveConnection2SAP; any other usage needs that additional line.

- If such a line is added to the *t4s_mapping_config.sd* file, this account will be used for SAP access and the user will not be able using another SAP connection for the given mode by logging in manually; not even another user name with the same SAP system.

Storing SAP Account Information for More Than One SAP Account to Be Used in a Workflow

T4S4 Configuration

Please make sure is working with all those SAP accounts as expected using the manual SAP login, before configuring the following!

In order to use different SAP systems in workflows (in one workflow job or using different workflow templates) do the following:

- Define each of the needed SAP accounts as described above, e.g.:

```
::T4S::CONNECTION2SAP::setConnectionInfoPlain2 ET1 800 USER PASSWD EN
::T4S::CONNECTION2SAP::setConnectionInfoPlain2 ET1 800 USER PASSWD EN
```

- In each workflow template where one of those SAP accounts should be used, the workflow argument `-use_SAP_system` may be added to the corresponding T4S4 handler, e.g. with the value `ET1` in the handler `T4S-transfer-MaterialMaster`.

- This defined SAP system will be used in this and subsequent workflow tasks and jobs as long as there is no other SAP system definition.

- In order to use the same SAP connection in each task of the same workflow template and checking this connection at the beginning of each workflow job, we recommend setting those arguments in the rule handler `T4S-validate-SAPLogon` instead, i.e. before any T4S4 handler, e.g.:
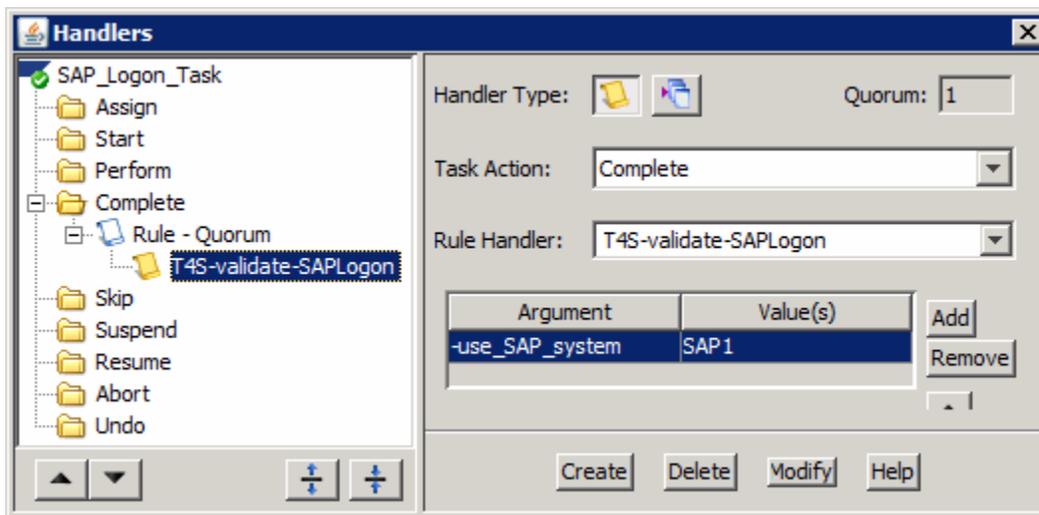
T4S4 determines the SAP system to be used in a workflow handler as follows:

- As long as no T4S4 workflow handler with the argument `-use_SAP_system` was executed in this T4S4 session (i.e. since T4S4 was started), it will use the SAP connection from the last definition with `selectActiveConnection2SAP` in the file *t4s_mapping_config.sd* in every workflow handler.

- As soon as a **T4S4** workflow handler with the argument `-use_SAP_system` was executed, T4S4 will go on using this SAP connection until another connection is defined.

- As this may result in different behavior when executing several jobs of the same workflow templates, we recommend either using the handler argument `-use_SAP_system` in every T4S4 workflow handler, or not using it at all.

Storing SAP Account Information in Encrypted Format

T4S4 Configuration

The following example will define a SAP account with encrypted user name and password:

```
::T4S::CONNECTION2SAP::setConnectionInfo2 ET1 800 341d0a85 14282c82 EN
```

This line is the same as the one using the plain text format, except for the function name is different and the strings for the user name and password are encrypted. Only the user name and password can be stored in an encrypted format.

In order to create the encrypted strings for the user name and password, do the following:

- Open the BGS Admin GUI and select "Script -> Scripts" in the menu.

- Search for the script `T4x Password Crypt`.

- Enter the SAP user name into the input field and run the script.

- You will get an output like `341d0a857eaf9529`.

- Include the output string into the line above in the file *t4s_mapping_config.sd*.

- Do the same for the password.

- Rebuild the mapping

This account will then be used for SAP access. If you need the SAP Auto Login for more than one mode (e.g. `WORKFLOW` and `SCRIPTING`), define them in separate lines.

2.3 Mapping Templates

T4S provides templates Mapping for basic functionality in the folder *<GS_ROOT>\var\mmap \t4s_mapping_config*. Some templates or functionalities are not included in these mapping examples. More advanced templates can be found in *<GS_ROOT>\var\template\t4sdemo\mmap \t4s_mapping_config*. Please note that many examples need the additional T4S demo template (TEM Installation).

The samples in the mapping template files may serve as a guideline to your own customer specific implementation and show the use of the functions provided to make the implementation easier:

- *t4s_bom_item_import_template.sd*
  Mapping Mapping for the BOM import to Teamcenter.

- *t4s_bom_mapping_template.sd*
  Mapping for the Bill of Material.

- *t4s_bp_item_import_template.sd*
  Mapping for the Business Partner import to Teamcenter.

- *t4s_pb_mapping_template.sd*
  Mapping for the Business Partner.

- *t4s_connection_mapping_template.sd*
  Mapping for mapping different SAP connections at runtime.

- *t4s_dir_item_import_template.sd*
  Mapping for the DIR import to Teamcenter.

- *t4s_dir_mapping_template.sd*
  Mapping for the Document Info Record.

- *t4s_dirbom_mapping_template.sd*
  Mapping for the Document Structure.

- *t4s_ecm_item_import_template.sd*
  Mapping for the ECM import to Teamcenter.

- *t4s_ecm_mapping_template.sd*
  Mapping for the Engineering Change Master.

- *t4s_equi_item_import_template.sd*
  Mapping for the Equipment import to Teamcenter.

- *t4s_equi_mapping_template.sd*
  Mapping for the Equipment.

- *t4s_equibom_import_template.sd*
  Mapping for the Equipment BOM import.

- *t4s_equibom_mapping_template.sd*
  Mapping for the Equipment BOM.

- *t4s_funcloc_bom_import_template.sd*
  Mapping for the Functional Location BOM import.

- *t4s_funcloc_bom_mapping_template.sd*
  Mapping for the Functional Location BOM.

- *t4s_funcloc_item_import_template.sd*
  Mapping for the Functional Location import.

- *t4s_funcloc_mapping_template.sd*
  Mapping for the Functional Location.

- *t4s_genbom_mapping_template.sd*
  Mapping for the Generic BOM.

- *t4s_genobj_import_template.sd*
  Mapping for the Generic Object import.

- *t4s_genobj_mapping_template.sd*
  Mapping for the Generic Object.

- *t4s_idgen_mapping_template.sd*
  Mapping for the Item ID generation from SAP objects (IDGEN).

- *t4s_ippe_node_mapping_template.sd*
  Mapping for the SAP iPPE node.

- *t4s_ippe_structure_import_template.sd*
  Mapping for the SAP iPPE structure import.

- *t4s_lov_mapping_template.sd*
  Mapping for the List of Values.

- *t4s_mapping_config.sd*
  The central mapping configuration file.

- *t4s_meta_import_template.sd*
  Mapping for the import of meta data, i.e. the source and target object do not need a 1:1 relationship.

- *t4s_meta_mapping_template.sd*
  Mapping for the generic job creation.

- *t4s_mm_item_import_template.sd*
  Mapping for the MM import to Teamcenter.

- *t4s_mm_mapping_template.sd*
  Mapping for the Material Master.

- *t4s_orderbom_mapping_template.sd*
  Mapping for the Order Bill of Material.

- *t4s_pir_item_import_template.sd*
  Mapping for the Purchase Info Record import to Teamcenter.

- *t4s_pir_mapping_template.sd*
  Mapping for the Purchase Info Record.

- *t4s_plmxml_refsetofoperations_mapping_template.sd*
  Mapping for the routing operations reference set.

- *t4s_plmxml2_bom_mapping_template.sd*
  Mapping for the PLMXML based BOM.

- *t4s_plmxml2_inspectionplan_mapping_template.sd*
  Mapping for the PLMXML based inspection plan.

- *t4s_plmxml2_routing_mapping_template.sd*
  Mapping for the PLMXML based routing.

- *t4s_prodorder_import_template.sd*
  Mapping for the Production Order import to Teamcenter.

- *t4s_project_import_template.sd*
  Mapping for the SAP project import.

- *t4s_project_item_import_template.sd*
  Mapping for the SAP project Item import.

- *t4s_project_item_structure_import_template.sd*
  Mapping for the SAP project Item structure.

- *t4s_project_schedule_import_template.sd*
  Mapping for the SAP project schedule import.

- *t4s_project_schedule_structure_import_template.sd*
  Mapping for the SAP project schedule structure import.

- *t4s_prop_mapping_template.sd*
  Mapping for the SAP Data View.

- *t4s_query_mapping_template.sd*
  Mapping for mapping the SAP query.

- *t4s_rtp_mapping_template.sd*
  Mapping for the Runtime Property.

- *t4s_schedule_mapping_template.sd*
  Mapping for the Schedule Manager integration.

- *t4s_variant_characteristic_mapping_template.sd*
  Mapping for Variant Characteristics.

- *t4s_variant_class_mapping_template.sd*
  Mapping for Variant Classes.

- *t4s_variant_mapping_toolbox_template.sd*
  Functions for the variant mappings.

- *t4s_variant_option_import_template.sd*
  Mapping for the Options of Variants.

- *t4s_variant_value_restriction_mapping_template.sd*
  Functions for the Variants mapping.

- *t4s_wbs_bom_import_template.sd*
  Mapping for the Work Breakdown Structure import.

- *t4s_wbs_bom_ mapping_template.sd*
  Mapping for the Work Breakdown Structure.

- *t4s_workcenter_import_template.sd*
  Mapping for the Equipment based Work Center import.

2.4   Basic T4S4 Preferences

The follwing preference is required to define the selectable SAP connections in the T4S Gateway Menu:
`T4S_SAPLogonSystemClientList`

The value consits of `<System>@<Client>`, e.g:

```
T4S_SAPLogonSystemClientList=
ET1@800
ET2@100
```

For more menu options please see **Configure the T4S Gateway Menu Options**.

3. Part Transfer - Material Master

3.1 Introduction

Please read the **Teamcenter Gateway - Generic Configuration Guide** first to understand the basics for this chapter.

Nomenclature

Please also see the **Glossary**.

Teamcenter	SAP
ItemRevision or PartRevision	Material Master (Material, MM)
Relation	Object Link
Release Status	Cross Plant Material Status
	Plant Specific Material Status

T4S4 Specifics Compared to the Generic Mapping

SAP Transaction Codes	
MM01	Create Material Master
MM02	Change Material Master
MM03	Display Material Master

Export	
TargetTypeName	MaterialMaster
Mapping template	*t4s_mm_mapping_template.sd*
Namespace Mapping	`::T4S::MM::CUSTOM::MAPPING`
Preferences start with	T4S_MaterialMaster
Handler	T4S-transfer-MaterialMaster
	T4S-validate-MaterialMaster
	See **T4S4 API Reference** for more details.
Job Type	`T4S_EXPORT_MM`

Import Job	
Job Type	`T4S_IMPORT_MM`
Mapping template	*t4s_mm_item_import_template.sd*

Relationship between TC and SAP Objects

In SAP there are by default no revisions for a Material Master. Therefore the relation between TC Part and SAP MM is n:1. If your SAP system uses revision levels, please see **Set Revision Level**.

> Caution:
>
> Some customers overcome the n:1 relationship limitation by adding the Revision ID as a suffix to the SAP Material Number. It can be done but is not recommended.

3.2 Global Preferences for MM

To define the data mapping from Teamcenter to SAP the `TargetTypeName` needs to be replaced by `MaterialMaster`. The prefix of the preference is `T4S_`.

Please read the **Teamcenter Gateway - Generic Configuration Guide** first.

- `T4S_MaterialMasterTypeList`

- `T4S_MaterialMasterMapping4`

- `T4S_MaterialMasterMapping2`

- `T4S_MaterialMasterFieldMapping2` (optional)

Optional preference to enable only selected Item Revision Types for the MM Display feature in the T4S Gateway Menu:

```
T4S_MaterialMasterDisplayableTypeList=
ItemRevision
```

Definition of Item Revision Types which should be Supported

The preference defines the Item Revision Types that are configured for the transfer.

```
T4S_MaterialMasterTypeList=
ItemRevision
```

> Caution:
>
> Only Item Revisions (no Items) and Part Revisions can be defined! A Material Master transfer is always based on TC objects of class type `ItemRevision` or its children.
>
> Be aware some custom Item Revision types require a blank before "Revision" (e.g. "`Item Revision`"), standard types do not (e.g. `ItemRevision`). Be careful to use the correct syntax.
>
> Be aware that you should limit the data for processing to the required objects to ensure best performance for transactions.

Definition of Related Teamcenter Objects that contribute Relevant Attributes to the Mapping

This step defines which data needs to be read from the Teamcenter source object. The objects have to be defined for each Item Revision Type. It is important to limit the amount of data to the data that you really need to get best performance for your transaction. Any data that is not required for the transaction should not be read!

Specify the related TC data objects from which you want to pass data to SAP.

```
T4S_MaterialMasterMapping4ItemRevision=
items_tag:Item
items_tag:Item:#__getAllProperties__#:Properties
items_tag:Item:IMAN_master_form:Item Master
#__getAllProperties__#:Properties
IMAN_master_form:ItemRevision Master
IMAN_classification:icm0
```

Definition of Related Teamcenter Objects to store back Information

Example for the configuration of the reverse mapping structure:

```
T4S_MaterialMasterMapping2ItemRevision=
items_tag:Item
items_tag:Item:IMAN_master_form:Item Master
```

Definition of Related Fields on Objects to store back Information

Example for the configuration of the reverse mapping structure:

```
T4S_MaterialMasterFieldMapping2ItemRevision=
object_desc
```

```
T4S_MaterialMasterFieldMapping2items_tag:Item=
item_id

T4S_MaterialMasterFieldMapping2items_tag:Item:IMAN_master_form:Item
Master=
user_data_1
```

3.3   Reserve Material Number

> **Note:**
>
> Reserving a Material Number in SAP does not prevent that anyone creates a SAP MM with this number if he states it explicitly. It only tells SAP not to give this number again if it is asked for a new one:
>
> • Using the SAP number generator to assign TC IDs
>
> • Reserving a Material Number in SAP for future creation of the MM.

> **Caution:**
>
> In both following cases the number will only be reserved in SAP, no Material Master will be created

Using the SAP Number Generator to assign TC IDs

Please see **Using the SAP Number Generator to assign TC IDs**

Using the MM Reserve Functionality

The T4S Gateway Menu entry **MM RESERVE** (see **Configure the T4S Gateway Menu Options**) calls the Material Master Mapping in *t4s_mm_mapping_template.sd*. The number can be written back to Teamcenter as configured in the Mapping.

Steps to enable the functionality:

• Configure the Mapping like you would do to create a Material Master. Define at least the mandatory values for reserving a number:

```
set ::SAPDat(Material:HeadData:IND_SECTOR)  $IndustrySector
set ::SAPDat(Material:HeadData:MATL_TYPE)   $MatlType
```

• Add the value `MM_RESERVE` to the preference `T4S_Gateway_Menu_Explorer_Entries`.

3.4 Create Material Master

Creation of a Material Master for the first time.

T4S4 Configuration

The following example shows the basic data to create a Material Master in SAP. If you want to use a different number in Teamcenter and in SAP, it is useful to fill one of the Teamcenter attributes with the SAP Material Number.

You can find an example in the file *<T4x_GS_ROOT>\var\template\capabilities\T4S \t4s_materialmaster_mapping_doc.sd* under "Create Material Master". This example shows the data mapping where T4S4 first reads the attribute where the SAP Material Number is stored, and only if this is empty, the Item ID is used as the SAP Material Number.

> Caution:
>
> When creating a Material Master Record, SAP automatically converts the number to full upper case. Any further access will fail if the ID is not given exactly as stored in SAP (i.e. in full upper case). That's why the `string toupper` command should always be used.

Check Material Existence

T4S4 provides a function named `::TPSAP::MM::checkMaterialMasterExistence` to check if a Material Master exists.

Codes for Industry Sector and Material Type

When trying to create a SAP Material Master it is required to specify an **Industry sector** in the MM mapping file:

```
set ::SAPDat(Material:HeadData:IND_SECTOR) $::T4S_Defaults(Industry)
```

It does not work if you use the name as you can see it in the list on the "initial screen" when creating the Material Master Record in SAP (e.g. "Mechanical Engineering"), but you have to specify a single uppercase letter. You can get the code table from your SAP system as follows:

- Log in to SAP

- Type "SPRO" into the transaction code field and press Enter

- Click on "SAP Reference IMG"

- Choose: Logistics - General – Material Master – Field Selection – Define Industry Sectors...

- The code here is found in the column "Industry sector"

Change View "Industry Sectors": Overview

New Entries

Industry sector	Industry description	Field reference
1	Retail	A
2	Aerospace & Defense	A
3	Service Provider	A
A	Plant Engin./Construction	A
B	Beverage	A
C	Chemical Industry	C
F	Food and Related Products	P
M	Mechanical Engineering	M
P	Pharmaceuticals	P
W	Retailing	A

Similarly, there is a four letter uppercase code for the "**Material type**", example for the setting in the MM mapping file:

```
set ::SAPDat(Material:HeadData:MATL_TYPE) $::T4S_Defaults(SAPMaterialType)
```

You can obtain the code table as follows:

- Log in to SAP

- Type "SPRO" into the transaction code field and press Enter

- Click on "SAP Reference IMG"

- Choose: Logistics - General – Material Master – Basic Settings - Material Types – Assign Material Types...

- The code here is found in the column "Industry sector"

Number Ranges for Material Master

SAP refuses creating a new MM if an ID is given that is not allowed for the specified Material Type. Check the current Material Number range:

- Log in to SAP

- Type "SPRO" into the transaction code field and press Enter

- Click on "SAP Reference IMG"

- Choose: Logistics-General – Material Master – Basic Settings – Material Types – Define Number Ranges...

- Click on the clock with the green check mark

- On the next screen, click the button "Intervals"

- Check which number ranges are enabled for External use.

Using SAP Material Numbers Longer Than 18 Characters

By default a SAP Material Number has a maximum length of 18 characters for SAP Business Suite® and 18-40 characters for SAP S/4HANA®. Nevertheless it can be configured to a custom length.

The function `::TPSAP::S4::getMatNrLength4TargetSystem` will return the default length of your target SAP system: 18 characters for SAP Business Suite® and 40 characters for SAP S/4HANA®. If you have a different setting, you have to define the length in the *t4s_mapping_config.sd* with the following parameter:

```
set ::T4S_Defaults(MatNrLengthS4:<SAPTargetSystem>) "24"
```

The function `::TPSAP::S4::getMatNrLength4TargetSystem` will then also return the defined value.

> Caution:
>
> As described above, SAP does not handle a MM ID longer than 18 characters at all by default. If that is configured in a customer SAP system, please ask the customer's SAP experts how it works in their SAP system and how T4S4 may use it. Maybe they have to configure their SAP especially so that T4S4 will be allowed using such a special internal SAP field.

3.4.1 Use Reference Material for Creation

Option 1: Using a mapping function

The function `::TPSAP::MM::setMaterialMaster_by_Reference` reads data from an existing SAP MM and sets all the values into the new structure `::SAPDat` which will be used for creating the new MM. Any value may be modified after the call in the mapping. Then the MM is created in SAP.

Example for the usage with plant specific data (the parameter `MaterialMaster_Ref` contains the ID of the SAP MM you want to copy, the parameter `Plant` the ID of the plant from which data have to be read):

```
# Read the reference MM data:
::TPSAP::MM::setMaterialMaster_by_Reference TEST true
$MaterialMaster_Ref $Plant
# Set the desired MM ID for the new MM:
set ::SAPDat(Material:HeadData:MATERIAL) $MaterialMaster_New
```

There are additional parameters to that function that allow to copy more data. Please check the **T4S4 API Reference** for details.

> Caution:
>
> Any changes you want to make to the data that is sent to SAP, you have to implement after the function call. Else the existing data may be overwritten by data from the Reference Material. You also may want to unset data that you don´t want to send to SAP.
>
> This functionality is not exactly the same as the SAP default "Copy from..." when creating a new Material Master; that really creates a complete copy of an existing MM (see option 2). In contrast to that, this T4S4 functionality for creating a Reference Material cannot copy every data from an existing Material. For example, it cannot copy the Material long texts or classification values. If you want to copy all of the attributes you have to read them first from the Reference Material and then set them explicitly in the Mapping. This is possible even if there is plant specific data to be copied for several plants.
>
> If any of the given parameter values does not exist for the given MM, SAP will return "NOT_FOUND" for this function call and not give any data. For example, if there is a MM in Plant 1000 in your SAP: when calling the function `setMaterialMaster_by_Reference` for this existing MM with the value 2000 for the Plant, you will not get empty Plant data for this MM, but not any MM data at all.

Option 2: Create Direct (T4S Gateway Menu)

Only with **"MM Create"** where a SAP window opens (but not with "Create Direct"), T4S4 can use the SAP functionality to create a new MM as a copy of an existing one. To activate this, add the following line in the MM mapping file:

```
set ::SAPDat(Material:Parameter:UseRefMaterial) 12345
```

If you then click **"MM Create"** in the T4S Gateway Menu, you will find the given value "12345" automatically filled in the field "Copy from...". Then you may type a new MM ID (or just press Enter to get one from SAP) and have a new MM with the same data as your referenced MM 12345.

> Caution:
>
> Unfortunately it may be that SAP does not create the new MM with completely "the same data"! Depending on its configuration there may be some data missing in the new MM; especially special data such as "Storage location" etc. Please make sure to check if SAP copies all the needed data to the new MM in such a case before putting that Reference Material feature into the T4S4 installation in the production environment.
>
> As in any other case when creating a new MM, you need to specify the "Industry sector" and "Material type" as well when creating or modifying an object by T4S4 in the SAP GUI, SAP does not allow setting all the same data as without the SAP GUI (started by the T4S Gateway Menu "Create" instead of "Create Direct")

3.4.2 Use Reference Material for Plant Extension

This is similar to using a Reference Material for creation with the function `::TPSAP::MM::setMaterialMaster_by_Reference`. The difference is that you need to provide the optional function parameter Plant too.

3.5 Update Material Master

If the Material Master has already been created in SAP, each following transaction will be MM Update by default. An overview of the maintainable SAP fields can be found in the Material Master Mapping template.

Additional Update of a Material Master.

In some cases it is necessary to update data after the actual transaction was completed. e.g. setting a status to a SAP object may be possible only after all the other data is stored to this object.

A default mapping procedure is available to define post update data for MM. If the optional function `additionalUpdates4MaterialMaster` exists in the MM Mapping (in the same Namespace `::T4S::MM::CUSTOM::`), it is executed after the actual transaction is finished. The post update data has to be defined separately in this additional function. Previously defined `::SAPDat` variables will be unset before the function is called.

You can find an example in the file *<T4x_GS_ROOT>\var\template\capabilities\T4S\t4s_materialmaster_mapping_doc.sd* under "Additional MM Update". This Example shows how to update the document status after the transfer of metadata, Object Links and original files (that has been done during the normal DIR transaction).

> Caution:
>
> The namespace for this function is different to the namespace of the Material Master Mapping.

Troubleshooting

The message *The material cannot be maintained since no maintainable data transferred.* means that you either sent the same data that is already stored in this object or tried to set data that is not allowed there. For example, the SAP target field is a LOV (List of Values) that does not allow the new value. In both cases, SAP does not store anything in this transaction.

3.6 Maintain Unit Of Measure

3.6.1 Maintain Basic Unit of Measure

The basic unit of measure is a required SAP attribute for the Material Master. It can be changed as long as there exists no order for that material in SAP.

You can set the base UOM either by ISO code (`BASE_UOM_ISO`) or depending on your login language (`BASE_UOM`) or both:

```
set ::SAPDat(Material:MaraData:BASE_UOM_ISO) EA
```

```
set ::SAPDat(Material:MaraData:BASE_UOM) PC
```

3.6.2 Maintain Alternate Unit of Measure

A material can have alternate units of measure. E.g. if the basic unit of measure is PC the alternate UOM could be KG.

3.7 Maintain Basic Data

An overview of the maintainable SAP fields can be found in the file *t4s_mm_mapping_template.sd*.

Define The Weight, Volume and Size

By default, none of those fields is mandatory. You can leave out an entire section but if for example you want to specify the weight, you need to specify the weight unit, too.

Setting units of measure (for weight, volume...) might be tricky. If SAP refuses your unit with the error message "****** **not a valid unit of ...**" in a log file (the asterisks are part of the message; the dots represent the type of unit), check in the **currently used SAP system** (this may be different in another SAP system) for the correct string for your desired units of measure with the desired SAP logon language.

- Set your desired unit of measure in an existing MM.

- Check the SAP internal contents with the T4S4 test script "SAP get material master data" (*sap_get_materialdata.tcl*).

- Search the data and use the found string in the mapping file.

Example

When you stored the volume unit "CCM" (cubic centimeter), you get the following output in the test script:

- **...VOLUMEUNIT:1) = CCM**

- **...VOLUMEUNIT_ISO:1) = CMQ**

Set the Net Weight

```
set ::SAPDat(Material:MaraData:NET_WEIGHT) 100
set ::SAPDat(Material:MaraData:UNIT_OF_WT) KG
```

Set the Gross Weight

As a product may be used with different packages, it is possible to define several gross weights. In this case you have to specify additional entire blocks with `:2..n` instead of `:1` as in the example above. Then the `ALT_UNIT` will be different from the `BASE_UOM` and the values for `NUMERATOR` and `DENOMINATOR` define the conversion rule (the quantity relation between net and gross weight, e.g. how many pieces are in a box).

```
set ::SAPDat(Material:MarmData:UNIT_OF_WT:1)  KG
set ::SAPDat(Material:MarmData:ALT_UNIT:1)    ST
set ::SAPDat(Material:MarmData:NUMERATOR:1)   1
set ::SAPDat(Material:MarmData:DENOMINATR:1)  1
set ::SAPDat(Material:MarmData:GROSS_WT:1)    20
```

Set the Volume

```
set ::SAPDat(Material:MarmData:VOLUME:1)      10
set ::SAPDat(Material:MarmData:VOLUMEUNIT:1) CCM
```

Set the Physical Size

```
set ::SAPDat(Material:MarmData:LENGTH:1)    10
set ::SAPDat(Material:MarmData:WIDTH:1)     20
```

```
set ::SAPDat(Material:MarmData:HEIGHT:1)   30
set ::SAPDat(Material:MarmData:UNIT_DIM:1) MM
```

3.7.1   Maintain Cross-Plant Configure Variant Data

A Configured Material (or Variant) is a material that can be kept in stock and that arises from an individual configuration of a configurable product. The Material Master of a Variant is linked to the Configurable Material and configured using the characteristics of the Configurable Material.

Set Material to Configurable

In order to define a SAP Material Master as Configurable Material, add the following line to the MM mapping function:

```
set ::SAPDat(Material:Parameter:SetConfMatFlag) X
```

Depending on the SAP configuration it may need additional data in order to enable it.

3.7.2   Set Revision Level

- Use Case 1: When creating the material.

- Use Case 2: After the material has been created.

> **Caution:**
>
> For creating a Revision Level in SAP (this applies to any object type where SAP allows a Revision Level, not only to Material Masters) be aware of the following: For setting the Revision Level, SAP requires the usage of a change number. This ECM has to be valid for the object type the Revision Level has to be assigned to SAP may allow setting the Revision Level to a given value or not. If you think T4S4 does not set the value correctly, please ask your SAP consultants if your SAP system allows setting the value fix (else SAP generates a value no matter what is transferred).
>
> For setting the Revision Level to a Material Master, SAP requires the usage of a change number that is valid for Materials:
>
>

Use Case 1: When Creating the Material

Add the following example to your MM mapping file (for instance, directly after the `MATL_TYPE` and `IND_SECTOR` section). In this example, the ECM ID is taken from the attribute `SAPChangeNo` in the Item Revision Master Form and the Revision Level is the Item Revision ID. You can find an example in the file *<T4x_GS_ROOT>\var\template\capabilities\T4S\t4s_materialmaster_mapping_doc.sd* under "Set Revision Level".

After a successful MM Create transaction, you should see a new SAP MM with a Revision Level:
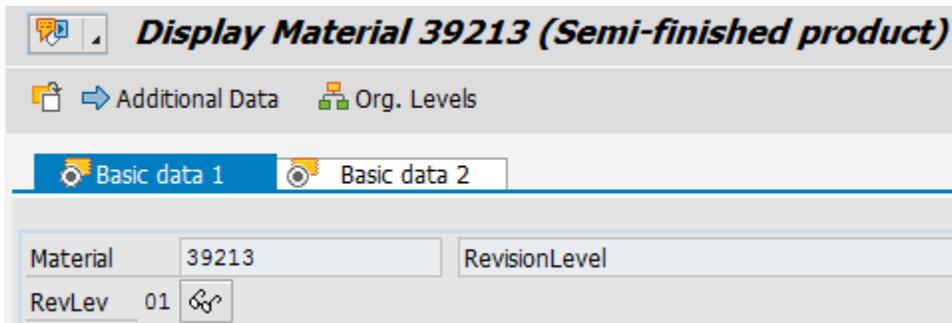
Use Case 2: After the Material Has Been Created

Use the function `::TPSAP::ECM::setRevisionLevel` in the mapping after the Material Master has been created in SAP.

3.7.3 Set X-Plant Status

The X-Plant or Cross Plant Material status is a plant independent status.

Set the following line in your Mapping with a valid 2 digit code:

```
set ::SAPDat(Material:MaraData:PUR_STATUS) XX
```

3.7.4 Maintain Long Text

The following Long Texts are supported:

* Long Text

* Sales Text

* Material Notes

Long Text

In order to see the SAP Material long texts, open transaction **MM03** for a Material in the SAP GUI, click on **Additional Data** and from there select the tab according to your desired MM long text, e.g. **Basic data text** (GRUN), **Inspection text** (PRUE), **Purchase order text** (BEST), or **Internal Comment** (IVER).

Within the Material Master Mapping the function `::TPSAP::MM::setMaterialMasterLongText` is called to transfer SAP MM long texts (same for all but neither the "Material Notes" nor the "Sales text", see below).

The following mapping example shows how German and English Long Texts can be transferred for SAP Materials:

```
::TPSAP::MM::setMaterialMasterLongText 1 EN GRUN "1st line EN" A
::TPSAP::MM::setMaterialMasterLongText 2 EN GRUN "2nd line EN" A
::TPSAP::MM::setMaterialMasterLongText 3 DE GRUN "1st line DE" A
::TPSAP::MM::setMaterialMasterLongText 4 DE GRUN "2nd line DE" A
```

This will transfer two lines for each language. The line index must be contiguous, no matter if the lines are for one language only or for more than one language. The following example shows how to count the lines automatically:

```
set T4S_count  1  ;#  count the lines used for the Long Texts
::TPSAP::MM::setMaterialMasterLongText $T4S_count EN GRUN "Line 1"
incr T4S_count
::TPSAP::MM::setMaterialMasterLongText $T4S_count EN GRUN "Line 2"
incr T4S_count
```

The function `::TPSAP::MM::setMaterialMasterLongText` has an additional optional parameter `FormatColumn` which allows selecting a defined formatting for that long text line in SAP. As this format is defined in SAP internally, it is impossible to provide here a complete list of all those format codes.

> **Caution:**
>
> By default SAP does not handle more than one line per language if the parameter `FormatColumn` is empty. As it allows any character by default (one character only), the easiest way is just setting it to e.g. "A"

Sales Text

The MM long text "Sales text" is shown in the tab **Sales text** in **Main Data**.

Setting the SAP MM long text "Sales text" works a bit different from the other SAP MM long texts. So there is another T4S4 mapping function `::TPSAP::MM::setMaterialMasterSalesLongText` to set it in SAP because it needs different parameters. You can find an example in the file *<T4x_GS_ROOT>\var\template\capabilities\T4S\t4s_materialmaster_mapping_doc.sd* under "Maintain Longtext".

> **Caution:**
>
> The SAP MM long text "Sales text" can only be set if the MM Sales View is activated. The "Sales Organization" and the "Distribution channel" must be set to values that exist in a Sales View.

Material Notes

Additionally there might be a SAP MM long text called "Material Notes" . T4S4 provides the mapping function `::TPSAP::MM::setMaterialMasterNoteLongText`. A correct mapping line for SAP plant 1000 is:

```
::TPSAP::MM::setMaterialMasterNoteLongText 1 EN 1000 "First English line" A
```

Read Long Text from any Object in SAP

In order to read the SAP MM long texts in the default MM mapping function `TC_Object2SAP_MaterialMaster`, use the following example in the file *<T4x_GS_ROOT>\var \template\capabilities\T4S\t4s_materialmaster_mapping_doc.sd* under "Maintain Longtext".

3.7.5 Read Basic Data

To read the data of a Material Master in SAP, you can use the following function:

```
::TPSAP::MM::getMaterialMasterInfo
```

Depending on the parameters you will also get more than the basic information.

There is another function available that uses a different SAP BAPI. If the information you want is not part of the retrieved data you can use this alternate function, although it requires more configuration:

```
::TPSAP::MM::getAllMaterialMasterInfosReadSingle
```

You can use the test script *sap_get_materialdata.tcl* to read the data of a MM. It uses both functions and can serve as an example.

3.8 Maintain Classification

Read the **Teamcenter Gateway - Generic Configuration Guide** first on how to get the TC Classification data.

> Caution:
>
> When the SAP classification is updated, T4S4 will remove all characteristics that have not been set in the Mapping. This can be avoided by **merging the classification**.

Use Cases / Requirements

• Creating or updating a single Material Master classification.

• Creating or updating multiple Material Master classifications.

Creating or Updating a single Material Master Classification

Here is an example on how to set a single classification:

You can find an example in the file *<T4x_GS_ROOT>\var\template\capabilities\T4S \t4s_materialmaster_mapping_doc.sd* under "Maintain Classification".

Creating or Updating Multiple Material Master Classifications

This example shows how to update more than one class. It is similar to single classification. Use the line `::T4S::storeSAPDat2MultiTransferCache Material` at the beginning of the classification data and `::T4S::storeObjectClassification2MultiTransferCache` at the end of each set of those data. You can find the example in the file *<T4x_GS_ROOT>\var\template\capabilities\T4S \t4s_materialmaster_mapping_doc.sd* under "Maintain Classification".

How to find the correct values for the Classification in SAP

If you want to set classification info for a SAP object, you first have to determine how the data objects are called in SAP:

- Log in to SAP

- Type transaction code **CL03** ("Display Class")

- Enter "Class" and "Class type" or choose the values from the lists.
  The "Class" is the name of your class. "Class type" is a reference to the type of data object the class belongs to, e.g. "001" for Material or "017" for Document (may be different on some SAP systems).
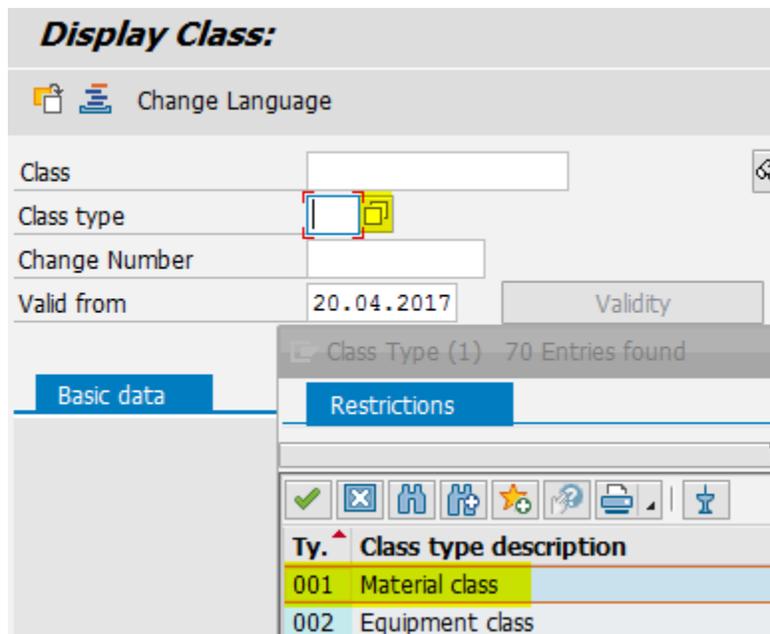
- Open the tab "Char." ("characteristics")
  You will find a list of all attributes defined for this class, including the name (column "Characteristic") and the data type with its length. This name is what you have to use as parameter "CHARACT" in the mapping file

- Check if the characteristic has a LOV (list of values) and what the valid entries are (two reasons: the setting in the mapping is different and SAP will refuse the value if it is invalid): Do for all the characteristic you want to set:

  - Select the line in the list

  - Click on the button "Disp. Vals" below the table.
    The list in the new view shows every possible entry of a LOV; if this list is empty, it is no LOV and allows random values (but dependent on the data type); maybe additionally restricted (see tab "Restrictions")

- Another possibility to check whether it is a LOV or not, is looking at an existing object with classification

  - Open your desired object

  - Click on "Additional data"

  - Maybe you have to select the appropriate tab

  - Click the button that appears at the end of each list when activated

  - If the shown list has only the entry "No entry", it is not a LOV

You can get more technical info of a SAP classification attribute with the SAP transaction **CT03** or **CT04**. Enter the desired "Characteristic", e.g. "release_date". Be aware that such an attribute is unique to the entire SAP classification.

> Caution:
>
> Do not set values to characteristics that are filled automatically by Object Dependencies (DE: "Beziehungswissen") because SAP does not allow it by default.
>
> The values may be language dependent.
>
> SAP may reject the entire transaction if there is a format error in any attribute. Please see **Correct Number and Date Format**.
>
> Errors in the mapping to set SAP classification data are often hard to track because most of the error messages do not contain a hint where they belong to. Neither the classification itself nor the characteristic name is mentioned. For example you get the error *Entry does not have the format date* when you try to set a SAP class attribute that is formatted as date.

3.8.1 Read Classification

For the reading SAP classification, T4S4 provides the following set of functions:

- `set ::SAPDat(Material:Parameter:UpdateClassification) TRUE`

Tell T4S4 to use SAP classification with an object of type Material

- `::TPSAP::CLASS::selectClassificationDefinition`
  Selects the class that T4S4 should use.

- `set ::SAPDat(ObjectClassification:ParameterData:READ_VALUATIONS) X`
  Setting read mode.

- `::TPSAP::CLASS::selectMaterialMasterClassification`
  Select the MM whose classification data should be read.

You can find an example in the file *<T4x_GS_ROOT>\var\template\capabilities\T4S \t4s_materialmaster_mapping_doc.sd* under "Read Classification".

3.8.2 Merge Classification

To avoid unintentional deletion of SAP classification values, it is necessary to read the SAP date before the transfer and reuse it again for setting the SAP classification.

T4S4 can do this automatically using the function `::TPSAP::CLASS::mergeObjectClassification`. The following example shows the usage of that function that reads the values from the defined SAP class and merges them with those set in the mapping from the data of Teamcenter:

```
set Status [::TPSAP::CLASS::selectMaterialMasterClassification
$MaterialNumber]
set ResultList [::TPSAP::CLASS::getDetailedObjectClassification Mapping]
if {[lindex $ResultList 0] == "FOUND"} {
  ::TPSAP::CLASS::mergeObjectClassification
}
```

Disabling Default Values

If you want to allow SAP using default values, omit the switch `NO_DEFAULT_VALUES`.

```
set ::SAPDat(ObjectClassification:ParameterData:NO_DEFAULT_VALUES) X
```

3.8.3 Delete Classification

You can find an example in the file *<T4x_GS_ROOT>\var\template\capabilities\T4S \t4s_materialmaster_mapping_doc.sd* under "Delete Classification".

3.9 Maintain Plant Specific Data

- Creating or Updating a Material Master for one specific plant.

- Creating or Updating a Material Master for multiple plants.

Creating or Updating a Material Master for one specific plant

Set the following in your mapping:

```
set ::SAPDat(Material:MarcData:PLANT) $::T4S_Defaults(PlantId)
```

> Caution:
>
> You also need to maintain a plant specific View to create the Material in the Plant context.

Creating or Updating a Material Master for multiple plants.

SAP only accepts one Plant ID per transaction. In order to set additional data for another plant, you have the following options:

- **Multi Plant Transfer**
  Use the T4S4 functionality allows transferring more than one set of data to SAP where SAP only allows one per transaction. You can find an example in the file *<T4x_GS_ROOT>\var\template\capabilities\T4S\t4s_materialmaster_mapping_doc.sd* under "Maintain Plant Specific Data".

- **In the Workflow via Parameter**
  Attach an additional action handler T4S_update-MaterialMaster-AH. For every plant you must use one instance of this handler in the same or an additional task. In order to have the action handlers perform different things for each plant, add a parameter for the plant to each of those handlers (e.g. "-Plant" as argument name and the needed Plant ID as value). This parameter has to be read in the mapping via
  ```
  set Plant [::T4X::TC::Mappping::getWorkflowArgumentValue –Plant]
  ```

- **Via T4S Gateway Menu**
  Every "Change Direct" can set data for another plant. Therefore it is useful to store the data for the new plant in a Teamcenter attribute that is modifiable in the T4S Gateway Menu.

List of Available Plants in SAP

You can the a list of available plant numbers in SAP:

- Log in to SAP

- Start transaction MM01 (Create Material)

- Type an existing Material number and press Enter

- The fields "Industry sector" and "Material type" are filled from the existing MM

- Click "Select Views"

- Select at least one plant specific view (e.g. MRP or Purchase) and press Enter (or click the green tick icon)

- The window "Organizational Levels" opens

- Click on the Button for the list of values beneath the field "Plant"

- The window "Possible entries: Plants" opens

It shows every plant that is maintained for this SAP system. The plants, for which any data is already stored with the current MM, are marked with an "x"

Check in which Plants a Specific Material Master is Maintained.

In your mapping you can use the function `::TPSAP::MM::getPlantsOfMaterial` to get a list of all plants a specific Material is maintained and the views which exist for each plant. Please see also *List of Maintained MM Views for a Plant*.

The result variable `$::MMResultDat(PLANTS:WERKS:$LineIndex)` contains the Plant ID. The variable `$::MMResultDat(PLANTS:PSTAT:$LineIndex)` contains a string of digits that represent the views that are maintained in that Plant:

- A: Work scheduling

- B: Accounting

- C: Classification

- D: MRP

- E: Purchasing

- F: Production resources/tools

- G: Costing

- K: Basic data

- L: Storage

- P: Forecasting

- Q: Quality management

- S: Warehouse management

- V: Sales

- X: Plant stocks

- Z: Storage location stocks

List of Maintained MM Views for a Plant

Use the function `::TPSAP::MM::getViewList4Plant` that returns a list of maintained Views.

Troubleshooting

The error message *Field WERKS has been transferred inconsistently or is blank* or *WERKS for MARC is ""; previously ""* means that you tried to set plant specific data but did not give a valid Plant ID. Maybe additional plant specific data needs an additional plant ID (although the ID is the same in most cases). E.g. the "Storage Location" requires setting the plant ID for the storage data separately:

```
set ::SAPDat(Material:MardData:PLANT) ...
```

3.9.1 Set Plant Specific Status

If you want to set plant specific status for a MM, you have to specify the Plant ID and at least one plant specific view in the MM mapping. You can find an example in the file *<T4x_GS_ROOT>\var\template \capabilities\T4S\t4s_materialmaster_mapping_doc.sd* under "Set Plant Specific Status".

3.9.2 Maintain Procurement Type

You can find an example in the file *<T4x_GS_ROOT>\var\template\capabilities\T4S \t4s_materialmaster_mapping_doc.sd* under "Maintan Procurement Type". Depending on your SAP system configuration and the values you have set, SAP might need more data to be set than in this example.

3.9.3 Assign Manufacturing Resource Planning (MRP) Area

The MRP Area is an organizational unit and independent from a Plant perspective. It can be assigned to a Material-Plant context.

The SAP Transaction code is `OMIZ`.

T4S4 provides transport packages to read and write SAP MRP Area assignments to a Material-Plant combination:

- `/TESISPLM/MRP_AREA_GET_ALL`

- `/TESISPLM/MRP_LEVEL_CREATE`

Please see **Generic Transfer** and chapter **SAP Transport Packages for Teamcenter Gateway for SAP S/ 4HANA** in the **Teamcenter Gateway for SAP S/4HANA - S/4HANA Preparation Guide**.

3.9.4 Maintain Inspection Control Data

You can find an example in the file *<T4x_GS_ROOT>\var\template\capabilities\T4S \t4s_materialmaster_mapping_doc.sd* under "Maintan Inspection Control Data".

3.9.5 Maintain Batch Data

The transfer of a Material Batch is an extra step of the normal Material Master transfer. Nevertheless the mapping of a Material Batch is part of the normal Material Master Mapping. The Material Batch update is enabled by the following variables in the mapping (first enable the Material Batch transfer as a whole, then set the needed data for it):

You can find an example in the file *<T4x_GS_ROOT>\var\template\capabilities\T4S \t4s_materialmaster_mapping_doc.sd* under "Maintan Batch Data".

3.9.6   Maintain Tax Indicator

T4S4 provides the possibility to create or update the Tax Indicator in a SAP MM.

Although the tax data is part of the SAP MM "Sales View", the "tax indicator" is shown in the "Purchase View".Depending on its configuration SAP may require additional data in order to store the Purchase view and the tax indicator.

> Caution:
>
> In order to set the tax indicator, you have to provide a valid plant ID and a country ID (the so called departure country from which the goods are sent). The country must fit to where the plant is located.
>
> SAP stores the tax indicator internally for the country only, not for each plant. So if the currently processed MM has several plants stored that belong to the same country, and you set the tax indicator in one transaction with one plant, then SAP will use that tax indicator in that MM for every plant which belongs to the same country.
>
> The SAP error message "Tax category X is not defined for country" means that the country is not set.
>
> The SAP error message "Tax category X is not defined for country YY" means that a wrong tax indicator value is provided.

```
set ::SAPDat(Material:HeadData:MATERIAL)       $Material
set ::SAPDat(Material:MarcData:PLANT)          $Plant
set ::SAPDat(Material:HeadData:PURCHASE_VIEW)  X
set ::SAPDat(Material:MaraData:MATL_GROUP)     $MaterialGroup
set ::SAPDat(Material:MlanData:DEPCOUNTRY:1)   $DepartureCountry
set ::SAPDat(Material:MlanData:TAX_IND:1)      $TaxIndicator
```

Mapping function to set the Tax Indicator

The transport packages `TESISPLM_T4S_CALL_TRANSACTION` and `TESISPLM_T4S_MM` need to be imported to SAP. Please see chapter *SAP Transport Packages for T4S4 Functionality* in the **Teamcenter Gateway for SAP S/4HANA - S/4HANA Preparation Guide**.

In order to set a tax indicator from the currently processed MM, set the following mapping line:

```
set rc [::TPSAP::MM::setTaxIndicator4MaterialMaster $MaterialNumber $Plant
$TaxInd]
```

It can be used only after the MM already exists but can be defined it as additional transfer (Mapping function additionalUpdates4MaterialMaster).

The test script *sap_get_mm_tax_ind.tcl* can also be used as an example.

Mapping function to set the Tax Indicator

The transport package `TESISPLM_T4S_MM` needs to be imported to SAP. Please see chapter *SAP Transport Packages for T4S4 Functionality* in the **Teamcenter Gateway for SAP S/4HANA - S/4HANA Preparation Guide**.

The following code can be used:

```
set rc [::TPSAP::MM::getTaxIndicator4MaterialMaster $MaterialNumber $Plant
$TaxInd]
```

If the optional parameter Plant is not set, it returns the Tax Indicator for all the countries from which a plant is defined in that Material Master.

The test script *sap_set_mm_tax_ind.tcl* can also be used as an example.

3.9.7 Maintain Price

Set the following for storing price information in a SAP Material Master. By default, those data are stored on the accounting view (although they may be visible on other views as well):

```
set ::SAPDat(Material:HeadData:ACCOUNT_VIEW)  X
set ::SAPDat(Material:MbewData:VAL_AREA)      $Plant
set ::SAPDat(Material:MbewData:VAL_TYPE)      " "
set ::SAPDat(Material:MbewData:VAL_CLASS)     $ValClass
set ::SAPDat(Material:MbewData:PRICE_UNIT)    "1"
set ::SAPDat(Material:MbewData:PRICE_CTRL)    "S"
if {$::SAPDat(Material:MbewData:PRICE_CTRL) ne "S"} {
  # transfer standard price:
  set ::SAPDat(Material:MbewData:STD_PRICE) $Price
} else {
  # transfer average moving price:
  set ::SAPDat(Material:MbewData:MOVING_PR) $Price
}
```

Troubleshooting

The error *CURRENCY initial v., currency amount 1,2000 in STD_PRICE was transferred without a currency* means that anything in `MARC` or `MBEW` is missing (or inconsistent) in order to set the price information. Unfortunately SAP does not tell which field has to be modified.

The message *Enter a material type* could mean you are transferring valuation data, but miss necessary values for it. In most cases `MbewData:VAL_AREA` or `MbewData:VAL_CLASS` which may be mandatory.

3.9.8   Read Plant Specific Data

To read the plant specific data you can use the functions from chapter **Read Basic Data**. Fill in the plant in the parameters and you will get the plant specific data in addition to the basic data.

3.10   Maintain Custom Fields

T4S4 supports fields of the Material Master which were added as a customer specific solution to the MARA table. Depending on the SAP configuration, the same functionality can work in the same way with other SAP MM tables as well, e.g. MARC or MAKT. Then the additional SAP data structures to be called are BAPI_TE_MARC, BAPI_TE_MAKT ... (and BAPI_TE_*X)

Specific Configuration in SAP

For filling customized fields, the T4S4 Gateway needs an extension of the data structures BAPI_TE_MARA2 and BAPI_TE_MARAX2 in SAP.

- Open SAP GUI

- Open ABAP Dictionary (transaction `SE11`)

- Type in the table name, e.g. BAPI_TE_MARA(X)2.

For the BAPI_TE_MARA(x)2 the ".APPEND" structure must be filled: Component, Component Type, Data Type, Length, Decimal Places, Short Text. For each costumer specific field one entry is necessary.

T4S4 Configuration

The following example is for an ".APPEND" structure for one customized field `EXTWG` of type `CHAR` with a length of `18` in BAPI_TE_MARA. In the BAPI_TE_MARAX structure the component type for each customized field will be `BAPIUPDATE` and each component will be a `CHAR` with length `1`. You can find the example in the file *<T4x_GS_ROOT>\var\template\capabilities\T4S \t4s_materialmaster_mapping_doc.sd* under "Maintain Custom Fields".

If there are several fields defined in BAPI_TE_MARA (or BAPI_TE_MARC, respectively) and T4S4 should only set some of them without modifying the others, be sure to not set the `X` in BAPI_TE_MARAX (or BAPI_TE_MARCX, respectively) for the other fields. Nevertheless you have to provide the data to set in the correct place in the variable `ExtensionInString`.

Example for BAPI_TE_MARA (field name and length):

- MATERIAL, length = 18 or as defined in your SAP system

- Z_1, length = 4

- Z_2, length = 8

- Z_3, length = 5

In order to modify the field Z_3 only (example value: "test"), set the following: first make sure the data to set is on the correct position. Therefore count the lengths of each field before the one to modify together and pass this number of blanks (4 + 8 = 12) in *ExtensionInString*, followed by the value to set:

```
set ExtensionInString [format "%18s%12s%5s" $SAP_MaterialNr "" test]
```

Then tell SAP to only modify the third field there, i.e. pass two blanks and then one "X" in the *ExtensionInXString*:

```
set ExtensionInXString [format "%18s%1s" $SAP_MaterialNr " X"]
```

> **Note:**
>
> This will only work for data defined in the corresponding SAP material table (especially MARA or MARC). In the example above (see screen-shots above), it is the SAP default field `AENNR`. But if the Component Type in BAPI_TE_MARA(X) is anything not defined by SAP itself, you first have to define it in the corresponding SAP table (and check with table T130F).
>
>
>
> Please see corresponding SAP OSS 44410 - Integrating cust.-specific fields in matl master.

3.11 Maintain Object Links

This chapter describes how to maintain Object Links in a Material Master transaction. If you want to create an object link to a MM in a transaction of a different object (e.g. a DIR) you will find the information in the chapter for that object.

3.11.1 Maintain Engineering Change Master (ECM) Link

A link to the Engineering Change Master is automatically created if the change number is set during the ECM transfer:

Specific Configuration in SAP

The ECM needs to support the Material object type. Please see the chapter **Create Engineering Change Master** for more details.

T4S4 Configuration

Set the following line in your mapping:

```
set ::SAPDat(Material:Parameter:CHANGE_NO) $ChangeNumber
```

3.11.2   Maintain Document Info Record (DIR) Link

In most cases the Object Link between MM and DIR is created in the DIR transaction.

- Via MM transfer (recommended)

- Via custom SAP call

> Caution:
>
> You have to ensure that the ECM number has already been created in SAP.

Via MM transfer

A link to the Engineering Change Master is automatically created if the change number is set during the DIR transfer:

Specific Configuration in SAP

The ECN needs to support the Material object type. Please see the chapter **Create Engineering Change Master** for more details.

T4S4 Configuration

Set the following line in your mapping:

```
set ::SAPDat(Material:Parameter:CHANGE_NO) $ChangeNumber
```

Via custom SAP call

By default, SAP does not allow creating an object link to a Document Info Record during the processing of a Material Master. Therefore T4S4 provides an additional SAP function module.

Specific Configuration in SAP

To enable this functionality, import the SAP transport package "T4S_MISC" that is delivered with the T4S GS:

<GS_Root>/var/template/T4S/SAP/TransportPackages/TESISPLM_T4S_MISC.zip

It contains the following SAP function module that can be used by T4S4:

/TESISPLM/MM_OBJ_LINK_MAINTAIN

After the corresponding SAP function module is implemented, make sure your SAP user is allowed to use it. There are more functions included in that transport package. Please read the Readme file in that directory for more information.

T4S4 Configuration

You need to add an additional function call to your mapping procedure. The following functions can be used to create and delete a link to a DIR while processing a material master:

```
::TPSAP::TESISPLM::MM_OBJECT_LINK::MM_DocLinksCreate
::TPSAP::TESISPLM::MM_OBJECT_LINK::MM_DocLinksRead
::TPSAP::TESISPLM::MM_OBJECT_LINK::MM_DocLinksDelete
```

3.12 Maintain Attachment List

By default, SAP does not allow the Attachment List of a Material Master from an external application. Therefore T4S4 provides an additional SAP function module.

> Caution:
>
> Only the creation of Attachments is supported. Change or deletion of Attachments is not possible.

Specific Configuration in SAP

To enable this functionality, import the following SAP transport package, that is delivered with the T4S4 GS:

<GS_Root>/var/template/T4S/SAP/TransportPackages/TESISPLM/T4S_MISC.zip

It contains the following SAP function modules that can be used by T4S4:

/TESISPLM/MM_ATTACH_URL

After the corresponding SAP function module is implemented, make sure your SAP user is allowed to use it. There are more functions included in that transport package. Please read the Readme file in that directory for more information.

T4S4 Configuration

The following function can be used in your mapping:

```
::TPSAP::TESISPLM::MM_ATTACHMENTS::maintainURLs
```

3.13   Maintain Production Version

By default, SAP does not allow to maintain the Production Version of a Material Master from an external application. Therefore T4S4 provides an additional SAP function module.

Specific Configuration in SAP

To enable this functionality, import the following SAP transport packages, that are delivered with the T4S4 GS:

<GS_Root>/var/template/T4S/SAP/TransportPackages/TESISPLM_T4S_CALL_TRANSACTION.zip

<GS_Root>/var/template/T4S/SAP/TransportPackages/TESISPLM_T4S_MM.zip

They contain the following SAP function modules that can be used by T4S4:

- /TESISPLM/T4S_READ_PROD_VERS
  Reads one or all production versions for a given MM.

- /TESISPLM/T4S_CALL_TRANSACTION
  Creates or updates a production version in SAP using CALL TRANSACTION and SAP transaction C223.

After the corresponding SAP function module is implemented, make sure your SAP user is allowed to use it. There are more functions included in that transport package. Please read the Readme file in that directory for more information.

T4S4 Configuration

The following functions can be used in your mapping:

```
::TPSAP::MM::maintainProductionVersion
::TPSAP::MM::checkProductionVersion
::TPSAP::MM::readProductionVersion
::TPSAP::MM::deleteProductionVersion
```

Please check them out in the T4S4 API Reference for more details.

3.14 Maintain Configuration Profile

To create a Configuration Profile, the Material must already exist.

Add the section from the file *<T4x_GS_ROOT>\var\template\capabilities\T4S \t4s_materialmaster_mapping_doc.sd* under "Maintain Configuration Profile" into the T4S4 MM mapping function and fill it with valid data for your SAP system:

The call `::TPSAP::MM::initConfigProfilData` initializes the configuration profile data. Any configuration profile data set before this call will be empty.

In most cases, you will not need all of those parameters. The second example shows how to update the document status after the transfer of metadata, Object Links and original files (that has been done during the normal DIR transaction).

3.15 Maintain Approved Manufacturer Part List (AMPL)

The AMPL is a List of Approved Manufacturer Parts for a given Commercial Part in SAP.

Specific Configuration in SAP

To enable the functionality for maintaining (and reading) the AMPL, the corresponding SAP transport package needs to be imported to SAP:

<GS_Root>/var/template/T4S/SAP/TransportPackages/TESISPLM/T4S_AMPL.zip

It contains the following SAP function modules that can be used by T4S:

- /TESISPLM/CL_EX_AMPL (Class)

- /TESISPLM/IF_EX_AMPL (Interface)

- /TESISPLM/T4S_AMPL_READ

- /TESISPLM/T4S_AMPL_UPDATE

- /TESISPLM/AMPL (Classic BADI Def.)

After the corresponding SAP function module is implemented, make sure your SAP user is allowed to use it. There are more functions included in that transport package. Please read the Readme file in that directory for more information.

T4S Configuration

The following function can be used in your mapping:

```
::TPSAP::VENDOR::readAMPL

::TPSAP::VENDOR::updateAMPL
```

4. Document Transfer - Document Info Record

4.1   Introduction

Please read the **Teamcenter Gateway - Generic Configuration Guide** first to understand the basics for this chapter.

A Document Info Record (DIR) in SAP is very similar to a "Dataset" in Teamcenter: an object with some data that may include physical files which are called Originals in a SAP DIR. Those files can be opened in their original application from the SAP GUI (according to its configuration). Therefore, a Workstation Application has to be configured in SAP for the data type, and the application has to be available locally.

T4S4 supports the transfer of Teamcenter data based on Datasets or Item Revisions. The main difference between both methods is that the transfers can be based on the Item Revision or on the Dataset:

- Using the **Item Revision Method**, T4S4 will create one DIR from this one Item Revision. No matter how many Datasets it contains and how many files they contain all together.

- Using the **Dataset Method**, T4S4 will create by default one DIR from this one Dataset. No matter how many files it contains. It is possible to add the files of one dataset to the DIR of another dataset.

In both cases the workflow target is the Item Revision. To use the options of the T4S Gateway Menu or the SAP Data View, you have to select either the Dataset or the Item Revision depending on the configured method.

Nomenclature

Please also see the **Glossary**.

Teamcenter	SAP
ItemRevision, DocumentRevision or Dataset	Document Info Record (DIR)
Relation	Object Link
Release Status	Status of Document Info Record

T4S4 Specifics Compared to the Generic Mapping

SAP Transaction Codes	
CV01N	Create Document Info Record
CV02N	Change Document Info Record
CV03N	Display Document Info Record

Export	
TargetTypeName	DocumentInfoRecord
Mapping template	*t4s_dir_mapping_template.sd*
Namespace Mapping	`::T4S::DIR::CUSTOM::MAPPING`
Preferences start with	T4S_DocumentInfoRecord
Handler	T4S-transfer-DocumentInfoRecord T4S-validate-DocumentInfoRecord See the **T4S4 API Reference** for more details.
Job Type	`T4S_EXPORT_DIR`

Import Job	
Job Type	`T4S_IMPORT_DIR`
Mapping template	*t4s_dir_item_import_template.sd*

Relationship between TC and SAP Objects

Item Revision Method

The picture below shows object correlation between TC object and SAP Document Info Record (DIR) in case of using T4S4 Item Revision Method.
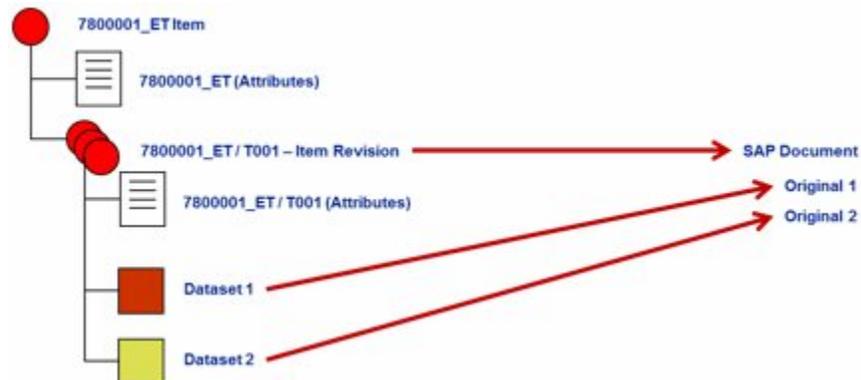
Based on configured TC Item types DIRs are created and configured Named Reference files of datasets are attached to DIR as original file data. Required meta data information can be gathered from Item Revision itself but also from related objects using T4S4 standard data extraction functionality. Please note that if there are more than two original files to be transferred to SAP, SAP KPro is a prerequisite.

Dataset Method

The picture below shows object correlation between TC object and SAP Document Info Record (DIR) in case of using T4S4 Dataset Method.
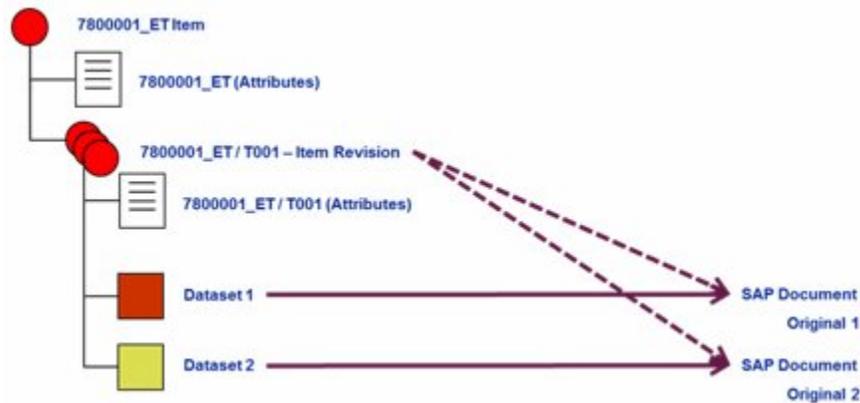
Based on configured TC dataset types DIRs are created and configured Named Reference files are attached to DIR as original file data. Required meta data information can be gathered from dataset itself but also from related objects using T4S4 standard data extraction functionality.

SAP Document Key

In SAP, the unique Key of a Document Info Record is defined by four dimensions:

- Document Number (18 digits): analog to TC Item ID

- Document Type (3 digits): analog to Dataset Type or Item Type (e.g. "DRW")

- Document Part (3 digits): can be used to store each original on a separate part (e.g. "000"). However this is rarely used and not recommended because of practical reasons.

- Document Version (2 digits): analog to TC Revision (e.g. "00" or "AA").

The information to identify this SAP DIR may be stored in a new form attached as Named Reference of the Dataset or in attributes of the Item or Item Revision.

4.2   Global Preferences for Document Info Record

To define the dataset types and data mapping from Teamcenter to SAP the following adaptations in the Teamcenter preferences have to be done. T4S4 will only use the configured Datasets and file types. It may upload all the files contained in the corresponding object (Item Revision or Dataset) to this one DIR as "original files".

> Caution:
>
> If not stated differently, all those keys have to be stated even if they are empty (because the corresponding functionality is not needed).

The following Preferences need to be set:

- `T4S_DocumentInfoRecordTypeList`

- `T4S_DocumentInfoRecordMapping4`

- `T4S_DocumentInfoRecordMapping2`

- `T4S_DocumentInfoRecordFieldMapping2 (optional)`

- `T4S_DocumentInfoRecordDefaultDocumentKey (optional)`

- `T4S_DocumentInfoRecordMappingNamedFileList (file transfer, optional)`

- `T4S_DIR_DataCarrierPath (file transfer, optional)`

- `T4S_DIR_DataCarrierName (file transfer, optional)`

- `T4S_DIR_WorkstationApplication (file transfer, optional)`

- `T4S_DIR_FileExtension (file transfer, optional)`

Optional preference to enable only selected Item Revision Types for the DIR Display feature in the T4S Gateway Menu:

```
T4S_DocumentInfoRecordDisplayableTypeList=
view
```

Definition of Supported Datasets or Item Revision Types

Item Revision Method

```
T4S_DocumentInfoRecordTypeList=
ItemRevision
```

Dataset Method

```
T4S_DocumentInfoRecordTypeList=
UGMASTER
```

Data Objects to be read

Specify the Teamcenter data objects that should be read when a DIR is processed. In this **example**, all the possible settings for Item and Item Revision are done (for the standard "Item" and the "Functionality" Item, too):

Item Revision Method

```
T4S_DocumentInfoRecordMapping4ItemRevision=
#__getAllProperties__#:Properties
items_tag:Item:#__getAllProperties__#:Properties
items_tag:Item:IMAN_master_form:Item Master
IMAN_master_form:ItemRevision Master
IMAN_specification:UGMASTER
IMAN_specification:UGMASTER:ref_list/UGPART:ImanFile
IMAN_specification:T4S_DatasetExtensionForm
```

Dataset Method

```
T4S_DocumentInfoRecordMapping4UGMASTER=
IMAN_specification:ItemRevision:#__getAllProperties__#:Properties
IMAN_specification:ItemRevision:items_tag:Item
IMAN_specification:ItemRevision:IMAN_master_form:ItemRevision Master
ref_list/UGPART:ImanFile
ref_list/propsync:T4S_DatasetExtensionForm
```

Default Document Key

For each Dataset type, specify the default Document Key for the SAP Document Info Record. This is always the Document Type, Part and Version. Be sure to state all those three values for every desired type (if there are less than three values, the key will be treated as if it was empty).

Item Revision Method

```
T4S_DocumentInfoRecordDefaultDocumentKey4ItemRevision=
CAD
001
00
```

Dataset Method

```
T4S_DocumentInfoRecordDefaultDocumentKey4UGMASTER=
CAD
001
00
```

Named References

For each Dataset type / Item Revision, specify a list of Named References. Those are the file types that you import into the Teamcenter Dataset and are attached to the SAP DIR as an "original". The following samples preferences for the *Item Revision Method* and *Dataset Method* show how it is possible to configure the UGPART file extraction in the context of the *Item Revision Method* or *Dataset Method* so that it can be uploaded to the SAP target system.

Item Revision Method

```
T4S_DocumentInfoRecordMappingNamedFileList4ItemRevision=
IMAN_specification:UGMASTER:ref_list/UGPART:ImanFile
```

Dataset Method

```
T4S_DocumentInfoRecordMappingNamedFileList4UGMASTER=
ref_list/UGPART:ImanFile
```

Temporary Files

For each file type, specify the data carrier path, i.e. where the file is saved before sending it to SAP.

Item Revision Method

```
T4S_DIR_DataCarrierPath4IMAN_specification:UGMASTER:ref_list/
UGPART:ImanFile=
C:/temp
```

Dataset Method

```
T4S_DIR_DataCarrierPath4ref_list/UGPART:ImanFile=
C:/temp
```

> Caution:
>
> This preference needs to be filled wither you use KPro or Data Carrier.

If this is a SAP file server, the files should remain there, else they may be deleted.T4S4 deletes them automatically if you set the following in the DIR mapping:

```
set ::SAPDat(Document:Parameter:DeleteOldFileSystemFiles) "TRUE"
```

Additionally, a copy of each file is stored temporarily before sending it to SAP. For determining the directory where the file will be stored, T4S4 checks the following values and uses the *first directory* that exists in the file system of the current TC Server environment *and* is writeable for T4S4 (depending on the operating system user access rights). If a preference or environment variable, respectively, from this list does not exist or is empty, T4S4 will go on checking the next target in the list without any problem:

- Teamcenter preference `TC_transfer_area`

- Operating system environment variable `TC_TMP_DIR`

- Operating system environment variable `TMP`

- Operating system environment variable `TMP`

- Operating system environment variable `TMPDIR`

The preference `TC_transfer_area` may contain several values (e.g. for Windows and for UNIX/Linux) that are checked one after another prior to checking the environment variables in the stated order. The environment variables may contain one value only (i.e. one possible target directory) each. As this check is done at runtime (not during the installation of T4S4) the target directory may be different on different machines or for different users.

Those files are deleted automatically if you set the following in the DIR mapping:

```
set ::SAPDat(Document:Parameter:DeleteTempOriginals) TRUE
```

> Caution:
>
> If T4S4 could not determine a valid directory according to this list, it cannot export the file from Teamcenter and therefore not upload it to SAP as DIR original file. In this case, an error message is written into the T4S4 Session Log.
>
> In UNIX/Linux, it should be something like "/tmp" instead of "C:/temp". A UNIX/Linux system may allow using "C:/temp" as well, but this is not the same as in Windows. Therefore it may confuse users and we do not recommend using it. Example of what will happen:
>
> ```
> $ pwd /home/infodba
> $ md c:/temp
> $ cd c:/temp
> $ pwd
> /home/infodba/c:/temp
> ```

Data Carrier or KPro

Files can either be stored in a Data Carrier (SAP Vault) or in a KPro but not both, so do not use both of the options `T4S_DIR_KproName4*` (with KPro) and `T4S_DIR_DataCarrierName4*` (without KPro)

in the Teamcenter preferences together. At least one of those two has to be stated empty. In either case, the option `T4S_DIR_DataCarrierPath4` has to be set.

Data Carrier - Item Revision Method

```
T4S_DIR_DataCarrierName4IMAN_specification:UGMASTER:ref_list/
UGPART:ImanFile=
SAPVault
```

KPro - Item Revision Method

```
T4S_DIR_KproName4IMAN_specification:UGMASTER:ref_list/UGPART:ImanFile=
DMS_C1_ST
```

Data Carrier - Dataset Method

```
T4S_DIR_DataCarrierName4ref_list/UGPART:ImanFile=
SAPVault
```

KPro - Dataset Method

```
T4S_DIR_KproName4ref_list/UGPART:ImanFile=
DMS_C1_ST
```

For the first tests, you can leave the preferences empty. If not set, the DIR creation is not disturbed (but the file cannot be stored at the correct target)

> Caution:
>
> If set with an incorrect value (not empty), the DIR is not created at all. So it may be hard to find the actual error.

SAP Workstation Application

For each file type (what you see in Teamcenter as "Named Reference"), specify the SAP "Workstation Application" which is used for viewing/printing the file. An incorrect setting does not affect the DIR creation as long as this Workstation Application exists. The only thing is that SAP might not be able to handle the file correctly.

Item Revision Method

```
T4S_DIR_WorkstationApplication4IMAN_specification:UGMASTER:ref_list/
UGPART:ImanFile=
PRT
```

Dataset Method

```
T4S_DIR_WorkstationApplication4ref_list/UGPART:ImanFile=
PRT
```

File Extension

For each file type, specify the file extension that SAP will attach to the file. This extension is not necessarily the file name extension used by the computer operating system (it may be the same accidentally). This also has no effect on the DIR creation itself.

Item Revision Method

```
T4S_DIR_FileExtension4IMAN_specification:UGMASTER:ref_list/
UGPART:ImanFile=
PRT
```

Dataset Method

```
T4S_DIR_FileExtension4ref_list/UGPART:ImanFile=
PRT
```

Simplified DIR data mapping configuration from Teamcenter to SAP

In order to facilitate the DIR preferences setting, T4S4 offers the possibility for specifying the necessary file handling data during DIR processing in the mapping.

Define the needed data by using the following mapping functions:

- `::T4S::DIR::CONFIG::setDocumentInfoRecordOriginalsConfig`
  Use this function to define the needed information to specify an original file transfer to SAP based on a given Teamcenter object type and the relation or "Named Reference" to the file.

- `::T4S::DIR::CONFIG::setDocumentInfoRecordKeyDefaults`
  Same as the preference `T4S_DocumentInfoRecordDefaultDocumentKey4<type>`, this function defines the default SAP Document Key (DIR type, part and version; as the ID will be different in nearly every transaction, it is not useful to define as a default) for a given Teamcenter object type (such as a Dataset type or Item Revision type)

```
::T4S::DIR::CONFIG::setDocumentInfoRecordOriginalsConfig \
MSWord ref_list/word:ImanFile WRD doc MyDatacarrierName "C:/temp" ""

::T4S::DIR::CONFIG::setDocumentInfoRecordKeyDefaults MSWord DRW 000 00
```

> **Caution:**
>
> Do not use the preference and mapping based configuration together because if a modification is required maybe only one of those points is adapted and then maybe that modification will not work as expected

4.3   Reserve Document Number

T4S4 number reservation is included in the "Create Object" wizard of TC. Hitting the "Assign" button, T4S4 creates a new DIR with defined type, part document and version in SAP and returns to TC the SAP DIR number which is used as item ID. Please note that SAP internal numbering has to be activated for this DIR type.

For the configuration, please see **Reserve Material Number**.

4.4   Create Document Info Record

Capability to create Document Info Records in SAP. Document Key and required metadata have to be defined in T4S4 mapping.

T4S4 Configuration

The following example shows the basic data to create a Document Info Record in SAP.

```
set DocumentNumber [::T4X::TC::MAPPING::FieldMapping $Item item_id]
set ::SAPDat(Document:DrawData:DOCUMENTNUMBER)   $DocumentNumber
set ::SAPDat(Document:DrawData:DOCUMENTTYPE)     DRW
set ::SAPDat(Document:DrawData:DOCUMENTPART)     000
set ::SAPDat(Document:DrawData:DOCUMENTVERSION)  00
set ::SAPDat(Document:DrawData:CADINDICATOR)     X
set ::SAPDat(Document:DrawData:DESCRIPTION)      Description
```

Set `::SAPDat(Document:DrawData:DOCUMENTNUMBER)` to an empty value, if you want SAP to use the next free number instead of an ID from Teamcenter. But then you need to store the returned SAP Document Number in a TC attribute in the Reverse Mapping.

Determination of valid Document Types

For creating a SAP Document it is required to specify a "Document type". You can get the list of valid types from your SAP system as follows:

- Log in to SAP

- Start transaction `SPRO`

- Click on "SAP Reference IMG"

- Click through Cross-Application Components > Document Management > Control Data > Define Document Types

- Click on the clock with the green check mark

Check Document Existence

T4S4 provides a function named `::TPSAP::DIR::ExistenceCheck` to check if a Document Info Record exists.

Searching a DIR in a Convenient Way in SAP GUI

In the SAP transaction "Display Document" (cv03n) a SAP Document Info Record may be searched without specifying the complete Document Key. An asterisk in the field "Document Version" and / or "Document Part" will open a list with all matches.

Determine the Latest DIR Version in SAP

Use the function `::TPSAP::DIR::getLatestVersion` to get the latest DIR version.

Use the function `::TPSAP::DIR::getActualVersion` to get a list of released DIR versions (by validity date).

Listing existing DIR Versions in SAP

Please use the following example code:

```
set ActiveVersion [::TPSAP::DIR::getActiveVersion $DIRno $DIRtype
$DIRpart]
foreach ResultElement [lsort [array names ::DIRResultDat \
"DOCUMENTLIST:DOCUMENTVERSION:*"]] {
    tpwrite -logchannel $LogChannelName -mtype DEBUG \
    "Found Version >$::DIRResultDat($ResultElement)<"
}
```

It is also possible to query for documents. You can find an example in the file *<T4x_GS_ROOT>\var \template\capabilities\T4S\t4s_documentinforecord_mapping_doc.sd* under "Create Document Info Record".

4.4.1 Assign Document Type

Capability to assign the Document Type of a DIR during creation. The SAP Document Type is a business critical key attribute of a Document Info Record, because it controls it´s common behavior (numbering, status net, ...).

Please see **Create Document Info Record**.

4.5 Update Document Info Record

Capability covers business requirement to update Document Info Records. Document Key and required metadata have to be defined in T4S4 mapping.

If the Document Info Record has already been created in SAP, each following transaction will be DIR Update by default. An overview of the maintainable SAP fields can be found in the Document Info Record Mapping template.

Additional Update of a Document Info Record.

In some cases it is necessary to update data after the actual transaction was completed. E. g. setting a status to a SAP object may be possible only after all the other data is stored to this object.

A default mapping procedure is available to define post update data for DIR. If the function `::T4S::DIR::CUSTOM::additionalUpdates4DocumentInfoRecord` exists in the DIR Mapping (in the same Namespace), it is executed after the actual transaction is finished. The post update data has to be defined separately in this additional function. Previously defined `::SAPDat` variables will be unset before the function is called.

You can find an example in the file *<T4x_GS_ROOT>\var\template\capabilities\T4S\t4s_documentinforecord_mapping_doc.sd* under "Additional DIR Update". This Example shows how to update the document status after the transfer of metadata, Object Links and original files (that has been done during the normal DIR transaction).

> Caution:
>
> The namespace for this function is different to the namespace of the Document Info Record Mapping.

4.6 Maintain DIR using ECM

SAP Change Management can be used for Document Info Records to control validity of DIR versions. Depending on SAP customization and implementation of SAP change processes Object Management Records are created automatically or by using specific T4S4 functions. The capability is used to cover this business requirement.

This example shows how to set the Change Number in the mapping:

```
set ::SAPDat(Document:DrawData:ECNUMBER) $ChangeNumber
```

4.7 Maintain Basic Data

This capability covers the need to transfer standard attributes which are not specifically named in other DIR related capability sections.

4.7.1 Maintain Longtext

Capability to maintain DIR long texts considering different languages.

Use the following example in the DIR Mapping to fill the DIR Longtext:

```
set Index 1
set ::SAPDat(Document:LONGTEXTS:LANGUAGE:$Index)      ""
set ::SAPDat(Document:LONGTEXTS:LANGUAGE_ISO:$Index) "EN"
set ::SAPDat(Document:LONGTEXTS:TEXTLINE:$Index)     "English"
incr Index
set ::SAPDat(Document:LONGTEXTS:LANGUAGE:$Index)      ""
set ::SAPDat(Document:LONGTEXTS:LANGUAGE_ISO:$Index) "DE"
set ::SAPDat(Document:LONGTEXTS:TEXTLINE:$Index)     "German"
```

To see the SAP DIR Longtext, open a "Display Document" window in the SAP GUI and click on the "Long Text" icon (an icon with a notepad and pencil next to the DIR description field)

> Caution:
>
> The first Longtext line will overwrite the "DIR Description", so you should transfer either the description or the DIR Longtext. If a DIR Longtext exists and you only transfer the "DIR Description", it will overwrite the first 40 characters ("DIR Description" only allows 40 characters) of the first Longtext line.
>
> Maybe you want to be sure and transfer both, so you could set the first Longtext line to the DIR description, e.g.
>
> ```
> set ::SAPDat(Document:LONGTEXTS:TEXTLINE:1) $T4S_DIR_Desc
> ```

SAP additionally offers the key `DELETEVALUE` for the DIR Longtext, but this is not useful in most cases, because SAP will anyway overwrite the complete DIR Longtext with the data you transfer.

4.7.2   Set Revision Level

The capability **Maintain Engineering Change Master (ECM) Link** is a pre-requisite for the "Set Revision Level" capability. The Revision Level of SAP DIR can be assigned to released documents which are referenced to a dedicated Change Master.

Note that SAP requires the usage of a Change Number for setting the Revision Level. This ECM has to be valid for documents.

Add the following sequence to your DIR mapping file. In this example, the ECM ID is taken from the attribute SAPChangeNo in the Item Revision Master Form and the Revision Level is the Item Revision ID:

```
set ::SAPDat(Document:DrawData:ECNUMBER)            $ChangeNumber
set ::SAPDat(Document:Parameter:setNewRevisionLevel)  $ItemRevID
```

After a successful DIR create or update transaction, you should see a new Revision Level in this DIR in SAP:
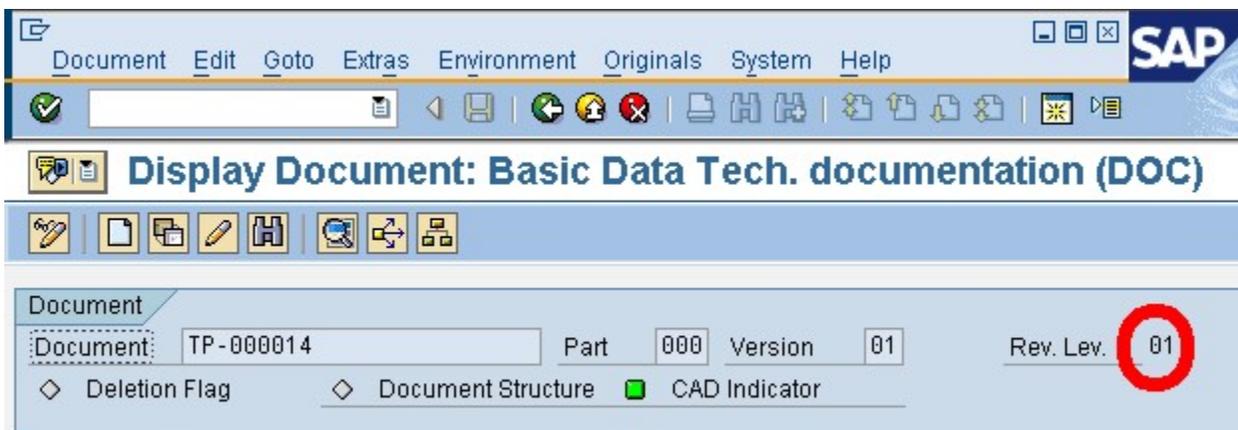
4.7.3   Maintain Status

Capability to maintain the SAP document status. The DIR status is a business critical attribute of a Document Info Record because its behavior (e.g release flag, option to change documents, ...)

Depending on the SAP installation you may either set the internal or external status. If in doubt, you have to check both of them (preferably only one at a time):

External Status

This is what you see it when you open the DIR in the SAP GUI. It works in most cases.

```
::SAPDat(Document:DrawData:STATUSEXTERN)
```

> Caution:
>
> The value may be different if you log in to SAP in different languages. If you connect to SAP in different languages, you need to keep this in mind.

Internal Status

This is the string as SAP stores it internally. This is not language dependent, so it is more fail save. The possible values are less obvious.

```
::SAPDat(Document:DrawData:STATUSINTERN)
```

Determination of valid External Document Status

You can get the possible values as follows:

- Log in to SAP

- Type transaction code `CV02N` for modifying an existing document

- Search for an existing document

- Click into the field "Document status"

- Open the entry help window by clicking the round icon beneath it

Determination of Internal Document Status

In a T4S4 Session Log of a DIR transaction, search for the line `::DIRResultDat(DOCUMENTDATA:STATUSINTERN)` or use the test script *sap_get_dirdata.tcl* to read the document status information from an existing SAP DIR.

Troubleshooting

Message: *When the document is changed, status must be changed* (when creating or updating a DIR) means that this document type requires

- **any** status when the DIR is created.

- **another** status when the DIR is modified.

You can check this in SAP in the definition of your document type, checkbox "Status change". Note that it may be impossible to change from one status to each other status. Maybe a status even prevents any further modification at all.

> Caution:
>
> SAP may allow creating a DIR directly in SAP GUI without asking for a status but it will possibly refuse the DIR creation by T4S4. This is not a bug but a SAP restriction.

4.7.4 Maintain Status On Different DIR

In some cases, after transferring a dedicated version of a document, the status of a predecessor or any other related or unrelated documents have to be changed. The T4S4 capability is used to handle this business requirement.

Use the following function in your mapping:

```
::TPSAP::DIR::setDocumentState
```

4.7.5 Read Document Info Record

Capability to extract metadata information of a DIR with a given Document Key.

T4S4 provides the function `::TPSAP:DIR::getDocumentInfoRecordInfo` to ask SAP for detailed information about a Document Info Record. The return value is `OK` if the DIR exists and SAP provides data for that, else `ERROR`. Note that you will also get `ERROR` if T4S4 does not get any answer from SAP, e.g. because you are not yet logged in.

```
set rc [::TPSAP::DIR::getDocumentInfoRecordInfo 0001 DRW 001 00 "" "" ""
X]
if {$rc eq "OK"} {
  foreach elem [array names ::DIRResultDat *] {
    tpwrite "::DIRResultDat($data): $::DIRResultDat($elem)"
  }
}
```

You can also use the test script *sap_get_dirdata.tcl* (SAP get DIR data) that uses the same function, but only with the mandatory parameters of the Document Key.

4.8   Maintain Classification

The classification of Document Info Records is used to extend metadata information of a DIR by customer specific attributes. Required classification information can extracted out of TC using T4S4 standard functionality. The capability covers the business requirement of DIR classification.

The classification for ECM works the same way like the MM Classification. The difference is the SAP SAP Class Type. It is `017` (Document management) instead of `001` (Material class).

Also the line

```
set ::SAPDat(Material:Parameter:UpdateClassification) "TRUE"
```

needs to be replaced with the line:

```
set ::SAPDat(Document:Parameter:UpdateClassification) "TRUE"
```

4.8.1   Read Classification

Capability to extract classification information of a DIR with a given Document Key.

This example is similar to reading the Classification of a Material Master. The function `::TPSAP::CLASS::selectMaterialMasterClassification` needs to be replaced by:

```
::TPSAP::CLASS::selectDocumentClassification \
$DocumentNumber $DocumentType $DocumentPart $DocumentVersion
```

Also the line

```
set ::SAPDat(Material:Parameter:UpdateClassification) "TRUE"
```

needs to be replaced with the line:

```
set ::SAPDat(Document:Parameter:UpdateClassification) "TRUE"
```

4.8.2   Merge Classification

The Merge Classification capability is related to Maintain Classification capability. In case that some pieces of classification data of a DIR is maintained in SAP (e.g. manually) and other pieces are transferred from TC, rules for merging characteristics and values have to be defined.

Please use the example from **Material Master**. Replace the line

```
set Status [::TPSAP::CLASS::selectMaterialMasterClassification
$MaterialNumber]
```

with:

```
    set Status [::TPSAP::CLASS::selectDocumentClassification \
            $DocumentNumber $DocumentType $DocumentPart $DocumentVersion]
```

4.8.3 Delete Classification

Capability to handle the need to delete document classification data.

Please see **Delete Classification**.

4.9 Maintain Object Links

DIR objects can be linked to other business objects like Material Master, Equipment etc. depending on SAP customization of specific DIR type. T4S4 provides a set of functions to maintain those links. Please note that once a DIR has been blocked for changes, Object Links cannot be changed as well.

Generic Functionality to create Object Links in a DIR Transaction

The subchapters of this chapters will provide examples for the most common cases. In principle the following will work for most SAP Objects.

```
    set ::SAPDat(Document:ObjectLink:<SAP_table>) $TargetID
```

`<SAP_table>` is the internal name of the SAP table for the desired object (or a T4S4 alias for that) and `TargetID` is the identifier for the desired object of this type. In many cases the `TargetID` needs to be the internal SAP ID.

Only update Object Links in a DIR Transfer

If you want your Object Links updated only although no original files are transferred, set the following additionally:

```
    set ::SAPDat(Document:Parameter:UpdateObjectLinks) TRUE
```

Copying Object Links from one DIR to another.

Use the function `::TPSAP::DIR::copyObjectLinks` to copy Object Links from a Document Info Record to another one.

4.9.1 Maintain Document Info Record (DIR) Link

Capability covers business requirement to copy existing links from a source DIR version to a target DIR version.

In order to create a SAP Object Link from the current DIR to another DIR (this target DIR must already exist), add the following line in the DIR mapping. The following example shows how T4S4 can create a SAP Object Link from the current DIR to the existing DIR with the stated `DocumentKey` that has to be created beforehand in the required format:

```
set InternalDocumentNumber  [::TPSAP::DIR::getInternalDocumentNumber \
                             $DocumentNumber]
set DocumentKey [format "%-3.3s%-25.25s%-2.2s%-3.3s" \
                $DocType $InternalDocumentNumber $DocVersion $DocPart]
set ::SAPDat(Document:ObjectLink:DOCUMENT) [list $DocumentKey]
```

> Caution:
>
> Although SAP always shows the Document Version after the Document Part, the order is swapped for the Document Key.
>
> Documents that you want to link this way have to be created first.

If you want to link the DIR to multiple other Document Info Record then you need to add their key as additional list elements:

```
set ::SAPDat(Document:ObjectLink:DOCUMENT) [list $DocumentKey1
$DocumentKey2]
```

For facilitating creating that list of SAP Document Keys, use the function `getDocumentKey4ObjectLink` in the Document Info Record mapping template file that may be adapted freely according to your needs.

4.9.2 Maintain Engineering Change Master (ECM) Link

The capability is a specific occurrence of Maintain Document Info Record (DIR) Link capability, linking DIR to ECM. Caused on different business process needs and data models, T4S4 provides a set of functions to create and maintain DIR/ECM links (link is created withing DIR transfer) and ECM/DIR links (link is created within ECM transfer).

Remarks:

- From a technical point of view the link is always created by changing the DIR.

- Link between ECM and DIR must not be mixed up with "Maintain DIR using ECM" which is dealing with validity and ECM object management records.

> Caution:
>
> Depending on its configuration, SAP may only accept that link setting if you specify the target with its internal number.
>
> Use the function: `::TPSAP::ECM::getInternalChangeNumber` to the internal number.

Please read the Chapters **Maintain Document Info Record (DIR) Link** and **Maintain Object Management Records** to set the Object Link in a MM or ECM transaction.

To set the link from a DIR to the ECM, add the following line in the Mapping:

```
set ChangeNoInt [::TPSAP::ECM::getInternalChangeNumber $ChangeNoExt]
set ::SAPDat(Document:ObjectLink:AENR) $ChangeNoInt
```

Via custom SAP Call

T4S4 allows to create a link between an ECM and a DIR in any transaction.

Specific Configuration in SAP

To enable this functionality, import the SAP transport package "T4S_MISC" that is delivered with the T4S4 GS:

<GS_Root>\var\template\t4s\sap\TransportPackages\TESISPLM_T4S_MISC.zip

It contains the following SAP function module that can be used by T4S4:

/TESISPLM/CM_OBJ_LINK_MAINTAIN

After the corresponding SAP function module is implemented, make sure your SAP user is allowed to use it. There are more functions included in that transport package. Please read the file *<GS_Root>\var \template\t4s\sap\TransportPackages\readme.txt* for more information.

T4S4 Configuration

You need to add an additional function call to your mapping procedure. The following functions can be used to read, create and delete links between DIRs and ECMs:

```
TPSAP::TESISPLM::CM_OBJECT_LINK::CM_DocLinksCreate
TPSAP::TESISPLM::CM_OBJECT_LINK::CM_DocLinksDelete
TPSAP::TESISPLM::CM_OBJECT_LINK::CM_DocLinksRead
```

4.9.3   Maintain Material Master (MM) Link

The capability is linking a Document Info Record to Material Master. To satisfy different business process needs and data models, T4S4 provides a set of functions to create and maintain DIR/MM links (link is created withing DIR transfer) and MM/DIR links (link is created within MM transfer). Please note that from technical point of view, a link is always created by changing the DIR.

In order to create a SAP Object Link from the current DIR to a MM, add the following line in the DIR mapping (from Teamcenter to SAP):

```
set ::SAPDat(Document:ObjectLink:MATERIAL) $MaterialNumber
```

If you want to create SAP Object Links from the current DIR to several MMs, just add their numbers, separated by blanks:

```
set MaterialNumbers "$MaterialNo1 $MaterialNo2 $MaterialNo3"
set ::SAPDat(Document:ObjectLink:MATERIAL) $MaterialNumbers
```

Please also see **Maintain Document Info Record (DIR) Link**.

4.9.4   Maintain Bill Of Material (BOM) Link

The capability is for linking a DIR to BOM Header or BOM Components.

For linking a Material BOM or an Order BOM to a DIR in SAP by T4S4, you first have to read the (Order) BOM in order to hand its data over to SAP during the DIR transaction. The Object Link may point to the BOM Header (shown in the examples below) or to a BOM Position instead. Therefore modify the following line as follows:

BOM Header

```
set ::SAPDat(Document:ObjectLink:STKO_DOC) $Header_GUID
```

BOM Position

```
set ::SAPDat(Document:ObjectLink:STPO_DOC) $Position_GUID
```

> **Caution:**
>
> Depending on the SAP configuration it is possible that SAP does not handle a BOM Object Link at all. Unfortunately SAP must not return an error but just ignores those additional data.
>
> It depends on your SAP configuration what happens if the target (Order) BOM does not exist. If SAP throws an error message, then T4S4 will show it in that transaction. Or SAP will display the Object Link although the BOM does not exist.

Reading the GUID of a Order BOM

Please see **Read Bill Of Material** on how to read a SAP Order BOM.

The required value for the Order BOM Header is returned in the variable `$::BOMResultDat(T_STKOB:GUIDX:1)`.

Reading the GUID of a Bill Of Material

The required value for the BOM Header is returned in the variable `$::BOMResultDat(STKO_API02:ID_GUID:1)`.

4.9.5 Maintain Equipment Link

The capability is linking am Equipment to a Document Info Record.

> **Caution:**
>
> Depending on its configuration, SAP may only accept that link setting if you specify the target with its internal number.
>
> Use the function: `::TPSAP::EQUI::getInternalNumber` to the the internal number.

In order to create a SAP Object Link from the DIR to an Equipment, add the following line in the DIR mapping (from Teamcenter to SAP):

```
set Equipment [::TPSAP::EQUI::getInternalNumber $EquiID]
set ::SAPDat(Document:ObjectLink:EQUI) $Equipment
```

If you want to create SAP Object Links from the current Equipment to several MMs, just add their numbers, separated by blanks:

```
set EquipmentNumbers "$EquipmentNo1 $EquipmentNo2 $EquipmentNo3"
set ::SAPDat(Document:ObjectLink:EQUI) $EquipmentNumbers
```

4.9.6 Delete Object Links

Capability to delete Object Links (specified by type of link) by changing the Document Info Record.

If you want to delete Object Links when updating the document (in order to have only those in SAP that you transfer now, nothing from older transfers), set the following:

```
set ::SAPDat(Document:Parameter:DeleteOldObjectLinks) ALL
```

Allowed values are:

- `FALSE`: default, i.e. it is the same as not setting it at all

- `ALL`: deletes all existing Object Links before storing the ones you transfer now

- `CAD`: deletes the Object Links that do not have a "direction" which is a hint (but not a prove!) that those Object Links were created by an external interface such as T4S4

4.10 Maintain Originals

Capability to maintain Originals on a SAP DIR.

Specific Configuration in TC

The following Preferences need to be configured. An explanation and examples for Item Revision Method and Dataset Method can be found in the chapter **Global Preferences for Document Info Record**.

- `T4S_DocumentInfoRecordMappingNamedFileList`

- `T4S_DIR_DataCarrierPath`

- `T4S_DIR_DataCarrierName`

- `T4S_DIR_WorkstationApplication`

- `T4S_DIR_FileExtension`

Specific Configuration in SAP

There are several obstacles to overcome if you want to transfer originals to SAP. Please read the **Teamcenter Gateway for SAP S/4HANA - S/4HANA Preparation Guide** first and make sure that the SAP system is configured correctly.

> Caution:
>
> The KPro ("Knowledge Provider") is the generic document management service in SAP. For document types without KPro support, only a maximum of two originals is supported for one DIR. And there is a limitation of the physical size of each file. Teamcenter can handle datasets with more than one file and T4S4 will process them all correctly, no matter if they have different "Named References" (independent of their actual file type).

T4S4 Configuration

In order to send the files as originals to SAP, the following line needs to be included in the mapping. The variable `ObjectTypeName` stands for the Item Revision or Dataset that has been configured for the transfer.

```
::T4S::DIR::MAPPING::getDocumentInfoRecordOriginals  $ObjectTypeName
```

After you executed this command, the variable `::SAPDat(Document:Files:NumberOfFiles)` is set to the number of found files on the object. If this number is not equal to the quantity of files you expected, recheck the configuration of your preferences and check your Teamcenter objects. Common errors are that the object relations are incorrect or the Dataset has no Named Reference.

Obtaining Information from Datasets and Files in the Mapping

Examples on how to get data from the Dataset and the File. You may add `#__getAllProperties__#:Properties` to the keys to get more data from the object as explained in the **Teamcenter Gateway - Generic Configuration Guide**.

Item Revision Method

Preference:

```
T4S_DocumentInfoRecordMapping4ItemRevision=
IMAN_specification:MSWord
IMAN_specification:MSWord:ref_list/word:ImanFile
```

Mapping:

```
set DatasetName [::T4X::TC::MAPPING::FieldMapping \
              "$ItemRevisionType:IMAN_specification:MSWord"
object_name]

set FileName [::T4X::TC::MAPPING::FieldMapping \
              $ItemRevisionType:IMAN_specification:MSWord:ref_list/
```

```
word:ImanFile \
           FileName]
```

Dataset Method

Preference:

```
T4S_DocumentInfoRecordMapping4MSWord=
ref_list/word:ImanFile
```

Mapping:

```
set DatasetName [::T4X::TC::MAPPING::FieldMapping $ObjectTypeName
object_name]
set FileName    [::T4X::TC::MAPPING::FieldMapping \
                 $ObjectTypeName:ref_list/word:ImanFile FileName]
```

Handling SAP DIR Original Files in the Mapping

The files shown in a SAP DIR are called original files, no matter if that are just links to the file system or really stored with the DIR, e.g. in a KPro. Whenever modifying anything related to the original files (including file names etc.), be sure to place those lines after the calls to fill missing data with default values (e.g. `getDocumentInfoRecord*` and `checkSAPDocumentKey`) in the T4S4 DIR mapping file, else the default settings requested by those function calls will override your settings.

By default, T4S4 transfers all files that are stored as configured Named References in each configured Dataset as original file to the SAP DIR. If you do not want to transfer files to SAP, add the following line at the end of your DIR mapping:

```
array unset ::SAPDat Document:File*
```

As that unsets all variables `::SAPDat(Document:FileNames:*)`
and `::SAPDat(Document:Files:*)` completely, be sure to place this unset command after any access to any of those variables.

By default, the already existing original files will be deleted from this DIR. If you do not want that, set the following additionally:

```
set ::SAPDat(Document:Parameter:DeleteOldOriginals) FALSE
```

The following line looks rather similar but it has a different meaning. Set this if you do not want to keep the temporary file copies.

```
set ::SAPDat(Document:Parameter:DeleteTempOriginals) TRUE
```

In order to delete files that have been copied to the physical file system (instead of storing them in a SAP Archive or KPro) during the DIR transaction, set the following:

```
set ::SAPDat(Document:Parameter:DeleteOldFileSystemFiles) TRUE
```

Those files may have been stored somewhere in the physical file system if the following was set:

```
set ::SAPDat(Document:FileNames:CopyTargetPath:$FileIndex) $SAP_file_path
```

T4S4 sets this to the same value as the preference `T4S_DIR_DataCarrierPath4` so that the files are only exported to a temporary directory. But if set to another file location explicitly, the files are copied there too. Please see **Global Preferences for Document Info Record**.

Definition of the Workstation Application Description

By default, the the column "Application" in the DIR original table shows the SAP description of the Workstation Application for this original file. Depending on the SAP DIR configuration, this may be modified. If you right-click an original file and select "Details", check if you find a line "Description" there and if you are allowed to modify it as shown in the following screenshot:
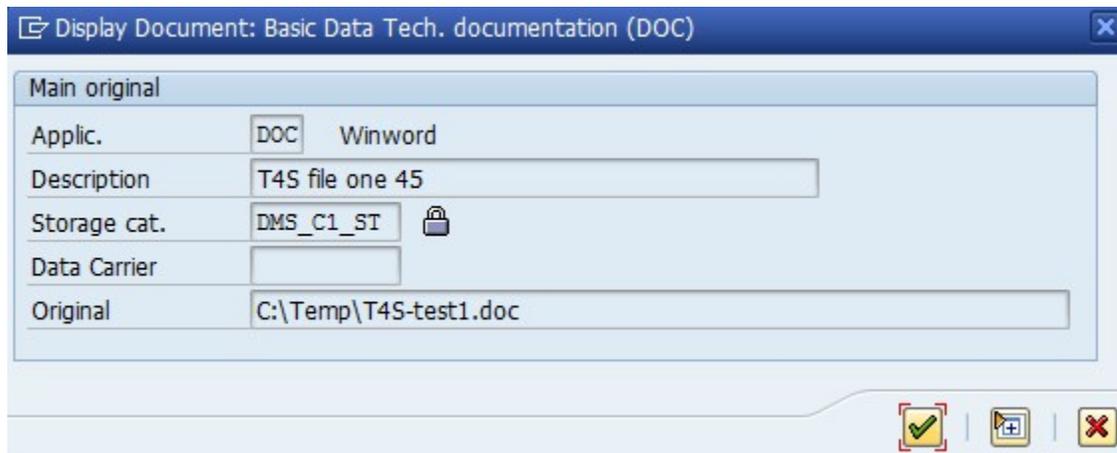
Then this will be shown as "Application" for this original file in the DIR:

If you cannot set it there, T4S4 cannot do it either. If SAP allows it, you can define it in the Mapping. Add the following lines:

```
set Index 1
set ::SAPDat(Document:Files:DESCRIPTION:$Index) "File one"
incr Index
set ::SAPDat(Document:Files:DESCRIPTION:$Index) "File two"
```

> Caution:
>
> This description can only be set in the currently used SAP login language. So when checking the DIR in the default view in SAP GUI (shown in the screenshot above) you will only see it if you are logged in to SAP in the same language as the DIR modification was done.

4.10.1 Maintain Physical File Original

Please see the subchapters **Check In Physical File** and **Store Reference to Physical File**.

Determination of the Workstation Application

You need a Workstation Application for each Named Reference in Teamcenter:

• Log in to SAP.

• Start transaction `SPRO`.

• Click on "SAP Reference IMG".

• Click through Cross-Application Components > Document Management > General Data > Define workstation application.

• Click on the clock with the green check mark.

The values in the first column "Appl" are the ones you are able to use. This screen also contains a column "Suffix" which is used by SAP if no other suffix is given

Determination of Storage Categories

If you are looking for a valid "Storage System" (Vault, Archive, SAP database or KPro) for documents:

- Log in to SAP.

- Type transaction code `CV02N` for modifying an existing document.

- Search for an existing document.

- If there is no "Original" with the open "padlock", create a new "Original".

- Right-click on this new Original and select "Check in under..."

A new window appears which has one of these two titles: "Check in document: KPro" if this Document type is configured for KPro in SAP. Or "Check in document: <Category>" with the name of the category you select in the window (Vault, Archive or SAP database). This window contains a list of the valid names for the given category.

> Caution:
>
> This may be different for every document type.

- Go to "Define Document Types" in SAP (see Document Type).

- Double click on the desired type name in the list.

- If "Use KPro" is active, a file for this document type will be checked in to a KPro, else one of the other categories.

Determination if the Document Type can store Originals with Files

This will show you the main details of all the defined DIR types in that SAP system. The first column "Type" are the ones you have to use as "Document type". When double-clicking a line it will show more details of the selected DIR type, e.g. "File Size". If this is set to "0" (zero), this DIR type does not allow storing any file!

- Log in to SAP.

- Start transaction `SPRO`.

- Click on "SAP Reference IMG".

- Click through Cross-Application Components > Document Management > Control Data > Define Document Types.

- Click on the clock with the green check mark.

4.10.1.1 Check In Physical File

Capability to transfer physical files to SAP DIR as so called original file data. File system, vault, database and KPro storage categories are supported by T4S4.

There are two ways to transfer the physical file to SAP:

Using SAPftp or SAPhttp

By default, the Original file check in is done by using SAPftp and SAPhttp functionality.

> Caution:
>
> Make sure that you obtained and installed the *SAPftp.exe* or *SAPhttp.exe* and further required libraries. Also check if the SAP system needs patches. Please see Chapters *SAPftp and SAPhttpT4S4* and *SAP OSS Notes (Patches)* in the **Teamcenter Gateway for SAP S/4HANA - S/4HANA Preparation Guide** for more details.

Using SAP Filestream

Technical capability to transfer physical files as a data stream instead of using SAPftp, SAPhttp technology.

> Caution:
>
> This method of transfer is only recommended for files smaller than 100 MB. The required disk space and RAM is significantly high for larger file sizes:
>
> | File Size | Disk Space Client | RAM Client | RAM Server |
> |---|---|---|---|
> | 20 MB | 100 MB | 500 MB | 200 MB |
> | 40 MB | 100 MB | 1.0 GB | 400 MB |
> | 50 MB | 1 GB | 1.3 GB | 500 MB |
> | 100 MB | 3 GB | 2.6 GB | 1 GB |

Specific Configuration in SAP

T4S4 requires two SAP Transport Packages to enable this functionality in SAP:

- /TESISPLM/T4S_DIR

- /TESISPLM/T4S_DIR_FILES_IMPORT

Please read the **Teamcenter Gateway for SAP S/4HANA - S/4HANA Preparation Guide**.

T4S4 Configuration

To enable File Stream, set the following parameter in the Mapping:

```
set ::TPSAP::DIR::CtrlDat(UseRfcDirectFileCheckin) "TRUE"
```

Definition of the File Name for the SAP Original

The original file name will be set by a SAP default pattern such as "00000000000100000000022_DOC_000_00_1.DOC". This contains the internal SAP Document Key, a counter and the file extension. In order to influence the original file name, set the following line in the DIR mapping:

```
set ::SAPDat(Document:Parameter:GenerateOriginalFileName) USE_EXTERNAL
```

Then the original file name will be the same but with the "external" DIR ID (i.e. without the leading zeros, e.g. "10000000022_DOC_000_00_1.DOC"). Not setting that line is the same as setting it to the default value `USE_INTERNAL`.

If you do not want that name but the physical file name of what was imported into the Teamcenter Dataset (e.g. *MyFile.doc*), set the following line instead:

```
set ::SAPDat(Document:Parameter:GenerateOriginalFileName)
USE_TC_ORIGINAL_NAME
```

In order to get complete control of the original file names, set that to the value `USE_USER_DEFINED` instead:

```
set ::SAPDat(Document:Parameter:GenerateOriginalFileName)
USE_USER_DEFINED
```

Without any more settings, the file name will be the file name Teamcenter uses internally in the database (which is a generic name in order to keep the name unique, e.g. "000010__wor_o2iO2pw4y9ljw.doc"). Add the following definition in order to set any defined name:

```
set ::SAPDat(Document:FileNames:UserDefinedFilename:$FileIndex) $MyName
```

The counter `FileIndex` means that is for exactly one of maybe several files in the current transaction. So this line has to be set for each file separately like in the following example:

You can find an example in the file *<T4x_GS_ROOT>\var\template\capabilities\T4S\t4s_documentinforecord_mapping_doc.sd* under "Maintain Originals".

Testing Physical File Transfer with SAPhttp or SAPftp

If the file *sapnwrfc.ini* contains the setting `USE_SAPGUI=0` it might have problems uploading "original files" to a SAP DIR, because it cant use SAPftp or SAPhttp correctly. There may be several reasons:

- The SAP system does not allow FTP communication at all (or not for this user)

- The computer operating system (either the one of the SAP server or that on your machine) prevents the FTP communication (port locks, firewall settings...)

- The front end does not have access to the SAP executables SAPftp and SAPhttp

In complex environments (e.g. with a SAP router), it might be impossible to establish the RFC connection to the front end. In order to test the physical possibilities with SAPftp and SAPhttp from your current machine to the desired SAP system with your current user, do the following:

- Start SAP transaction `SE38` (needs additional user rights for the "ABAP Workbench").

- Type the program name `RSFTP005` or `RSHTTP05`, for testing the SAPftp or SAPhttp connection capability.

- Click Execute (F8)

A possible result is shown in the following screenshot:
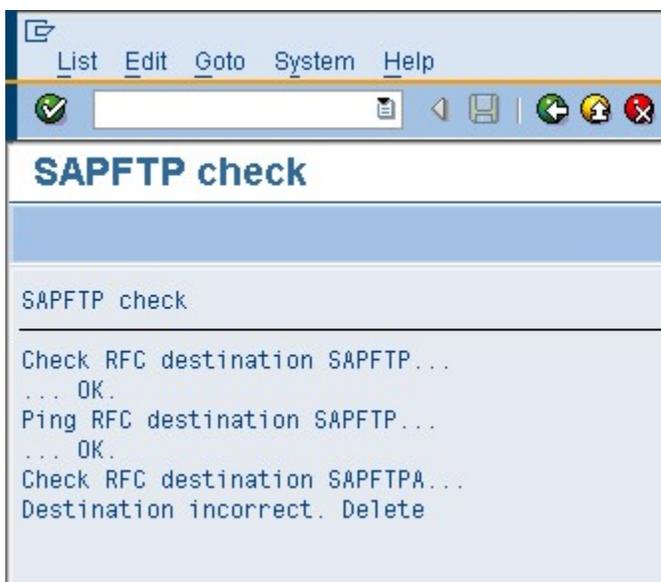
The RFC destination "SAPFTP" starts SAPftp.exe on a SAP front-end Workstation. Whereas RFC destination "SAPFTPA" does the same from an application server. So for your needs it is sufficient if "SAPFTP" is "OK" and you can ignore the "SAPFTPA" message.

If the "SAPFTP" result is not "OK", especially with errors like "timeout during allocate..." or "CM_PRODUCT_SPECIFIC_ERROR", T4S4 will most probably not be able to upload original files to a DIR with the setting `USE_SAPGUI=0` in the config file *sapnwrfc.ini*. The T4S4 settings may be correct, but nevertheless you will not get any original files in a SAP DIR. Depending on your installation, it might work with the setting `USE_SAPGUI=2` if the SAP GUI is installed.

> Caution:
>
> This test can only check if it works in principle, but not if the desired SAP system will accept files which are sent to it using your version of *SAPftp.exe/SAPhttp.exe* and additional RFC libraries.

Troubleshooting

Message: *Network address DEFAULT of your computer is not maintained.*

This is not because of incorrect mapping, but a SAP DIR configuration problem. Add the entry `DEFAULT` for your desired Data Carrier type: *SPRO – Cross-Application Components – Document Management – General Data – Define Data Carrier*. Define at least

- *Data Carrier type "server, front end"*: a type "PC" (selectable from the list, just add a local directory, e.g. *C:\temp*

- Below that in *Identify front end computers*: a type "Default" (selectable from the list, leave all defaults)

This means, you allow any Windows PC to connect to this SAP using the default Data Carrier. Only set an environment variable `hostname` with your computer name (which is recommended in the SAP help) if you need to identify each machine uniquely and do not want to set a default for all the machines.

4.10.1.2 Store Reference to Physical File

In this case the file is not stored in SAP but on a shared directory. The Original in SAP contain a link that points to the file. Please see **Store URL to Other Enterprise System Object** as both methods are almost the same.

4.10.2 Store URL to Other Enterprise System Object

Capability to transfer a TC URL (link to TC objects like revisions and datasets) as DIR original data. Please note that the appropriate workstation application (being able to open TC URL in web browser) has to be defined in SAP.

Specific Configuration in SAP

When trying to open such a URL in SAP, it may show the message "Browser is turned off in customization" instead. If so, modify the SAP DIR customization as follows:

- Start the transaction `SPRO` (normally this is not allowed to normal users).

- Click SAP Reference IMG

- Go to Cross Application Components > Document Management > General Data.

- Click the "execute" button beneath "Define Workstation Application".

- Search "HTM" in the list of Workstation Applications and set the "File Format" to "*.*" (asterisk dot asterisk, no quotation marks), although the default setting is "*.htm"

As this is specific to the "Workstation Application", you may have to do it more than once – once for every "Workstation Application" used with URLs

T4S4 Configuration

Example to transfer a link to a file and store it in the DIR:

```
set LinkUrl "\\\\Fileserver\\ShareDirectory\\MyFile.doc"
set LinkUrl "http://tc_dev_server:7001/tc/webclient/SJK5jWIfoQlgmD"
set ::SAPDat(Document:Parameter:OriginalsUseURL)        TRUE
set FileIndex 1
set ::SAPDat(Document:Files:DOCFILE:$FileIndex)          $LinkUrl
set ::SAPDat(Document:Files:SOURCEDATACARRIER:$FileIndex)   ""
set ::SAPDat(Document:Files:DESCRIPTION:$FileIndex)      "Original URL"
set ::SAPDat(Document:Files:WSAPPLICATION:$FileIndex)    "ACR"
set ::SAPDat(Document:Files:STORAGECATEGORY:$FileIndex)  ""
set ::SAPDat(Document:Files:DATACARRIER:$FileIndex)      ""
set ::SAPDat(Document:Files:ORIGINALTYPE:$FileIndex)     $FileIndex
set ::SAPDat(Document:Files:NumberOfFiles)              $FileIndex
```

Set the whole section again with `FileIndex` "2" instead of "1" for a second link.

Determination of the URL

The URL (in our example stored in `LinkUrl`) is where Teamcenter stores the file. If for example you have a TC Web accessible by the URL **http://tc_dev_server:7001/tc** and the TC internal file tag is **SJK5jWIfoQlgmD** then the correct value for that parameter could be for example **http://tc_dev_server:7001/tc/webclient/SJK5jWIfoQlgmD**. The URL varies depending on Teamcenter internal settings, so it cannot be described exactly.

> **Caution:**
>
> If the file is located in Teamcenter, the Teamcenter Thin Client (also known as TC Web) has to be installed in order to access the file. Note that Teamcenter does not allow accessing a file directly in the volumes.

4.10.3 Delete Originals

Capability to control handling (keep, delete) of

- Original file data of an existing DIR version within change process and

- Temporarily created files to be transferred as originals to SAP

Please read the subsection "Handling SAP DIR Original Files in the Mapping" in the chapter **Maintain Originals**.

4.10.4 Export Physical File

Capability to extract and store original file data of a DIR with a given Document Key (Document Number, Type, Part, and Version).

Use the following code example:

```
set IntDocNo [::TPSAP::DIR::getInternalDocumentNumber $DocumentNumber]
set ::SAPDat(Document:DocumentKey:DOCUMENTNUMBER)  $IntDocNo
set ::SAPDat(Document:DocumentKey:DOCUMENTTYPE)    $DocumentType
set ::SAPDat(Document:DocumentKey:DOCUMENTPART)    $DocumentPart
set ::SAPDat(Document:DocumentKey:DOCUMENTVERSION) $DocumentVersion
set ::SAPDat(Document:Parameter:ORIGINALPATH)      $ExportPath
set Status [::TPSAP::DIR::checkOutView]
set KeyList [lsort [array names ::DIRResultDat]]
foreach Key $KeyList {
   puts "> $Key = '$::DIRResultDat($Key)'"
}
```

You can also use the test script *sap_checkout_dir_view.tcl*.

5. Product Structure Transfer - Bill Of Material

5.1 Introduction

The Teamcenter BOM structure is read using a predefined revision rule. T4S4 can only send one BOM level per transaction to SAP. Sub assemblies have to be transferred in different transactions. T4S4 also allows transferring occurrence notes to BOM Line specific data to SAP.

Please read the Teamcenter Gateway - Generic Configuration Guide first to understand the basics for this chapter.

Nomenclature

Please also see the Glossary.

Teamcenter	SAP
(BOM) view	BOM
BOM Header	BOM Header
Position	Position
BOM Line	BOM Line
Release Status	BOM Status

T4S4 Specifics Compared to the Generic Mapping

SAP Transaction Codes	
CS01	Create Bill Of Material
CS02	Change Bill Of Material
CS03	Display Bill Of Material

Export	
TargetTypeName	BillOfMaterial
Mapping template	*t4s_bom_mapping_template.sd*
Namespace Mapping	`::T4S::BOM::CUSTOM::MAPPING`
Preferences start with	T4S_BillOfMaterial
Handler	T4S-transfer-BillOfMaterial
	T4S-validate-BillOfMaterial

Export	
	See **T4S4 API Reference** for more details.
Job Type	`T4S_EXPORT_BOM`

Import Job	
Job Type	`T4S_IMPORT_BOM`
Mapping template	*t4s_bom_item_import_template.sd*

Relationship between TC and SAP Objects

Teamcenter BOM objects (often called view) are independed objects in Teamcenter. These objects contain a structured list of Teamcenter objects and necessary information of the assembly, mostly of more than one level. Teamcenter BOMs can be loaded differently by applying another Revision Rule to it while opening.

In SAP there are by default no revisions for a Bill Of Material. Therefore the relation between TC Part and SAP Bill Of Material is n:1. If your SAP system uses revision levels, please see **Set Revision Level**.

5.2 Global Preferences for BOM

The extraction of Teamcenter BOM data is configured by preferences. All necessary preferences need to be created in order to trigger a BOM data transfer to SAP and leverage the extracted data into the internal TC Data array.

> Caution:
>
> T4S4 uses the BMIDE type name (e.g. "view") for the BOM handling. This is because "view" is the actual type name whereas "BOMView Revision" is only the display name.

To define the data mapping from Teamcenter to SAP the `TargetTypeName` needs to be replaced by `BillOfMaterial`. The prefix of the preference is `T4S_`. Please see also **Introduction**.

Please read the **Teamcenter Gateway - Generic Configuration Guide** first.

The following adaptations in the Teamcenter preferences have to be done in order to define the BOM view types and data mapping from Teamcenter to SAP:

- `T4S_BillOfMaterialTypeList`

- `T4S_BillOfMaterialMapping4`

- `T4S_BillOfMaterialHeaderTypeList` (optional)

- `T4S_BillOfMaterialHeaderMapping4` (optional)

- `T4S_BillOfMaterialLineMapping4` (optional)

- `T4S_BillOfMaterialOccurrenceNotes4` (optional)

- `T4S_BillOfMaterialMapping2`

- `T4S_BillOfMaterialFieldMapping2`

Optional preference to enable only selected Item Revision Types for the BOM Display feature in the T4S Gateway Menu:

```
T4S_BillOfMaterialDisplayableTypeList=
view
```

Definition of views which should be Supported

This preference contains a list of view objects which are valid for the BOM transfer.

```
T4S_BillOfMaterialTypeList=
view
```

> Caution:
>
> This preference only refers to the BOM view types themselves, no matter which Item Type they belong to. Therefore only View Types can be defined!

Definition of the Revision Rule to be applied to the view.

For each supported View, create a preference with the defined Revision Rule that needs to be applied to read the BOM.

```
T4S_BillOfMaterialMapping4view=
LatestWorking
```

Please read the chapter Applied Revision Rule.

The Revision Rule can also be defined as an argument of the Transaction Handler.

Definition of the Relevant Header Objects

Specify the related TC data objects from which you want to pass data to SAP.

For workflow transactions T4S4 needs to know which Item Revision types are valid as BOM Header types (because a Workflow can be started from the Item Revision only, not from the BOM view). The preference `T4S_BillOfMaterialHeaderTypeList` is optional. If not set all Item Revision types that are configured in `T4S_MaterialMasterTypeList` are valid for a BOM transfer. If you do not want to use the same Item Revision types for MM and BOM processing, set this preference additionally.

```
T4S_BillOfMaterialHeaderTypeList4view=
ItemRevision
```

> Caution:
>
> Like for the Material Master, only Item Revisions (no Items) and Part Revisions can be defined!

Definition of Related Teamcenter Objects that contribute Relevant Attributes to the Mapping

This step defines which data needs to be read from the Teamcenter source objects.

BOM Header

```
T4S_BillOfMaterialHeaderMapping4ItemRevision=
items_tag:Item
items_tag:Item:#__getAllProperties__#:Properties
items_tag:Item:IMAN_master_form:Item Master
#__getAllProperties__#:Properties
IMAN_master_form:ItemRevision Master
IMAN_classification:icm0
```

BOM Line

```
T4S_BillOfMaterialLineMapping4ItemRevision=
items_tag:Item
#__getAllProperties__#:Properties
```

> Caution:
>
> Be aware that you should limit the data for processing to the required objects to ensure best performance for transactions. This is especially important when you read large structures.
>
> It is recommended to use White Lists to limit the amount of data. Please see chapter *Blacklist/ Whitelist* of the **Teamcenter Gateway - Generic Configuration Guide** for more details on this subject.

Occurrence Notes

Occurrence Notes are small texts that may be attached to a Teamcenter BOM position in only this BOM. T4S4 only reads the defined values. Be aware that adding a lot of values to the list has an effect on performance.

```
T4S_BillOfMaterialOccurrenceNotes4view=
bl_has_children
My_OccNote
```

> Note:
>
> T4S4 can only read Occurrence Notes of type "string" (check in BMIDE).
>
> If BOM positions are packed in the Structure Manager, the value will be "Packed Notes". In this case the BOM lines need to be read out by T4S4 as unpacked. See **Maintain Position Data**.
>
> Some Occurrence Notes are read by T4S4 by default without having to configure them explicitly, e.g. `bl_quantity` and `bl_sequence_no`.

Definition of Related Teamcenter Objects to store back Information

T4S4 supports only to write back data to the BOM Header and Related Objects. It is not possible to write back information to Occurrence Notes or BOM Lines.

```
T4S_BillOfMaterialMapping2ItemRevision=
items_tag:Item:IMAN_master_form:Item Master
```

Definition of Related Fields on Objects to store back Information

Example for the configuration of the reverse mapping:

```
T4S_BillOfMaterialFieldMapping2ItemRevision=
object_desc

T4S_BillOfMaterialFieldMapping2items_tag:Item:IMAN_master_form:Item
Master=
user_data_1
```

5.3 Order Bill Of Material

The Order Bill Of Material transfer is very similar to the Bill Of Material transfer. In general the functional follows the same pattern like the normal BOM transfer (especially the handling of the BOMLines), the only difference is that the Order BOM Header is defined by a different set of identifiers:

- `ALTERNATIVE`: BOM Alternative

- `ORDER`: Order Number

- `POSITION`: Order Position Number

- `MATERIAL`: Material Number

- `PLANT`: Plant

- `BOM_USAGE`: BOM Usage

```
set ::SAPDat(OrderBom:Parameter:ALTERNATIVE)  1
set ::SAPDat(OrderBom:Parameter:ORDER)        $OrderNumber
set ::SAPDat(OrderBom:Parameter:POSITION)     $OrderPositionNumber
set ::SAPDat(OrderBom:Parameter:MATERIAL)     $MaterialNumber
set ::SAPDat(OrderBom:Parameter:PLANT)        $Plant
set ::SAPDat(OrderBom:Parameter:BOM_USAGE)    2
```

The test script *sap_get_bomdata_csai.tcl* ("SAP get BOM data using CSAI_BOM_READ") can be used to test the Order BOM read functionality.

SAP Transaction Codes	
cs61	Create Order BOM
cs62	Change Order BOM
cs63	Display Order BOM

Export	
TargetTypeName	OrderBillOfMaterial
Mapping template	*t4s_orderbom_mapping_template.sd*
Namespace Mapping	`::T4S::ORDERBOM::CUSTOM::MAPPING`
Preferences start with	T4S_OrderBillOfMaterial
Handler	T4S-transfer-OrderBillOfMaterial See **T4S4 API Reference** for more details.

No standard functionality for the import is provided.

5.4 Document Structure

The Document Structure transfer is very similar to the Bill Of Material transfer. The only difference is that the header and the positions are documents, which limits mapping possibilities.

In general we have two ways to create or maintain a SAP Document Structure within SAP:

- If the Document Structure is represented by a Teamcenter BOM view Structure, the handler T4S-transfer-DocumentStructure4BomView is the preferred way to maintain the SAP Document Structure in this case. The mapping template (*t4s_dirbom_mapping_template.sd*) is structured in the same way as for a Material BOM.

- If the Document Structure is not represented by a Teamcenter BOM view the standard DIR transfer mapping functions and preferences can be used as described in *Creating a Document Structure in the Document Info Record Mapping* (see below).

SAP Transaction Codes	
cv11	Create Document Structure
cv12	Change Document Structure
cv13	Display Document Structure

No standard functionality for the import is provided.

Creating a Document Structure for a BOM view.

Export	
TargetTypeName	DocumentStructure
Mapping template	*t4s_dirbom_mapping_template.sd*
Namespace Mapping	`::T4S::DIRBOM::CUSTOM::MAPPING`
Preferences start with	T4S_DocumentStructure
Handler	T4S-transfer-DocumentStructure4BomView See **T4S4 API Reference** for more details.

Creating a Document Structure in the Document Info Record Mapping

A document structure can also be created in the DIR Mapping in cases you don´t want to use a BOM view.

Use the workflow handler T4S-transfer-DocumentInfoRecord with the argument – `use_related_objects=true` as described in the **T4S4 API Reference**.

Creating Document Structures can require you to configure both Item Revision and Dataset Method for DIR transactions. In the mapping you need to configure the Dataset and the Item Revision branch too. Be sure that the Item Revision branch contains the line for the DIR structure. Like for a BOM you have to create the structure elements in SAP first.

The function `::T4S::DIR::CUSTOM::MAPPING::getDocumentInfoRecordStructure` can only be found in the demo Mapping template *<GS_Root>\var\template\t4sdemo\mmap\t4s_mapping_config \t4s_dir_mapping_template.sd*. It needs to be included and called in the DIR Mapping.

Check the function `getDocumentInfoRecordStructure` to have the correct settings for the DIR keys of the DIR structure positions. The section `::SAPDat(Document:DOCUMENTSTRUCTURE:*)` has to be empty for cases where you don´t want to create a document structure.

5.5 Maintain BOM for Relation

The T4S4 functionality BOM for Relation (also named "BOM4rel") is a special functionality which can process a SAP BOM based on related Item Revisions. An Item Revision of a defined Item type has some Item Revisions of defined Item types, attached with defined relations. Then all those related Item Revisions can be created as BOM positions in a SAP BOM which T4S4 can maintain in the same way as in the default BOM processing with a real TC BOM view.

> Caution:
>
> This functionality does not work with the T4S Gateway Menu, but with the specific workflow handler `T4S-transfer-BillOfMaterial4Relation` only.

That workflow can be defined to use the Teamcenter BOM positions of the workflow target instead of the defined related objects (see below how to set it), but it cannot use all those Teamcenter objects together for the SAP BOM.

Specific Configuration in TC

The following preferences have to be created additionally:

- `T4S_BOM4RelationTypeList`

- `T4S_BOM4RelationRelatedMappingObjectList4`

- `T4S_BOM4RelationRelatedMappingObject4`

- `T4S_BOM4RelationMapping2`

- `T4S_BOM4RelationFieldMapping2` (optional)

Specify which TC Item Revision types T4S4 should use as BOM4rel Header:

```
T4S_BOM4RelationTypeList =
ItemHeaderRevision
```

Specify for the Teamcenter BOM4rel Header object type which TC Item Revision types T4S4 should use as BOM4rel positions and with which relation it is found at the BOM4rel Header object. The following example shows two Item Revision Types which are related to the current Revision Type "HeaderItemRevision" which is the Teamcenter BOM4rel Header:

```
T4S_BOM4RelationRelatedMappingObjectList4HeaderItemRevision=
IMAN_Motion:ItemRevision
TC_WorkContext_Relation:PartRevision
```

> **Caution:**
>
> By default Teamcenter does not show the internal relation names but only the display name.
>
> Unfortunately Teamcenter does not allow every user seeing the real names of relations as well as object types. Then it is helpful using test script "List object relations and properties for an Item Revision".

Specify the objects T4S4 should read for the BOM4rel for the stated BOM Header object type:

```
T4S_BOM4RelationRelatedMappingObject4HeaderItemRevision=
#__getAllProperties__#:Properties
items_tag:HeaderItem
IMAN_master_form_rev:HeaderItemRevision Master
IMAN_Motion:ItemRevision
IMAN_Motion:ItemRevision:IMAN_master_form_rev:ItemRevision Master
IMAN_Motion:ItemRevision:items_tag:Item
TC_WorkContext_Relation:PartRevision
TC_WorkContext_Relation:PartRevision:IMAN_master_form_rev:PartRevision
Master
TC_WorkContext_Relation:PartRevision:items_tag:Part
```

> **Caution:**
>
> You only need to specify one preference for each BOM Header Item Type that includes the path to the Objects which have the role of the BOM positions.

T4S4 Configuration

As the Teamcenter object relations in a BOM view are different than the defined BOM4rel, T4S4 has to handle those two BOM functionalities differently. The BOM mapping has to be set adapted, especially it has to check if it is in a standard BOM transaction or BOM4rel in the BOM mapping function `TC_Object2SAP_BillOfMaterial`. You can identify the case by the workflow name or by reading a custom handler argument:

```
set ::BOM4relation false
```

```
if {[info exists ::TcData(Workflow:RootTaskName)]} {
  set T4S_WF_name [tpco_formatHEX16 $::TcData(Workflow:RootTaskName)]
  if {$T4S_WF_name eq "T4S_BOM4Rel"} {
    set ::BOM4relation true
  }
}
```

In the BOM position mapping function `TC_Object2SAP_BillOfMaterialPosition` no specific modification has to be done. The BOM4rel handler uses the related objects in the same way as positions from a Teamcenter BOM view. Of course if those are other Item Types or they should be handled differently than in a standard BOM transaction, that has to be defined:

```
if {$::BOM4relation} {
  set Quantity [::T4X::TC::MAPPING::IndexedFieldMapping \
              $PositionIndex $ItemRev cu4_quantity]
} else {
  set Quantity [::T4X::TC::MAPPING::IndexedFieldMapping \
              $PositionIndex BOMLine bl_quantity]
}
```

Using a Teamcenter BOM view

By default the BOM4rel handler does not use a real TC BOM view. If the BOM4rel workflow handler should also use a standard TC BOM, that can be defined with the handler argument –`bom4rel_use_bomview=true`.

If set to `true`, it checks if the workflow target Item Revision has a BOM view. If that is the case, the handler will use it in the same way as by the default BOM handler. This means it does nothing with the directly related objects. If no configured BOM view is attached to that Item Revision, it will do the BOM transfer based on the defined related objects.

Testmapping Script

Same as in nearly every T4x Transaction Type a helpful way to check if the configuration is correct is using the mapping test script. For BOM4rel it is the same test script as for the standard BOM functionality: *test_bom_mapping.tcl* (BOM Mapping Test). But in the first script parameter "SAP Target Object Type" the value "Bill Of Material based on Relation Structure" has to be selected.

5.6   Applied Revision Rule

A Teamcenter BOM is able to load Precise and Imprecise. Precise BOM Header and their positions refers to a dedicated revision state to identify precisely, imprecise BOMs will not.

The preference value can be overruled by the Handler-Arguments –`bom_rev_rule` and –`use_attached_rev_rule`. See **T4S4 API Reference** for more details.

5.7 Create Bill Of Material

Creating a Bill Of Material suggest that there is no BOM created with the key data before. Key data fields are Material Number, BOM Action, Alternative, Application and Usage. Depending on your SAP configuration some of these fields can be left empty.

```
set ::SAPDat(Bom:BomHeader:MaterialNumber)    $MaterialNumber
set ::SAPDat(Bom:BomHeader:BomAction)         5
set ::SAPDat(Bom:BomHeader:BomAlternative)    01
set ::SAPDat(Bom:BomHeader:BomApplication)    PP01
set ::SAPDat(Bom:BomHeader:BomUsage)          2
```

BOM Action

The BOM Action determines how SAP handles the BOM when using the T4S4 function "BOM Display".

The possible settings are:

- 1: Explode BOM level by level (online transaction `CS11` - single-level).

- 2: Explode BOM level by level (online transaction `CS11` - multi-level).

- 3: Explode multi-level BOM (online transaction `CS12`).

- 4: Display summarized BOM (online transaction `CS13`).

- 5: Display single-level BOM (online transaction `CS03`).

In most cases, the value `5` should be used. Then SAP will show the BOM in the same way as if you started transaction `CS03` ("Display material BOM") whereas the setting `1` will open `CS11` instead ("Explode BOM: Level by Level"). From the T4S4 point of view only those two settings (1 and 5) may be used reasonably:

```
set ::T4S_Defaults(BomAction) 5
```

BOM Alternative

Identifies a BOM of a BOM group. When a BOM is created, SAP creates automatically the first alternative.

> Caution:
>
> If no entry is defined, SAP will create the next free alternative.

BOM Application

This parameter defines how the BOM is exploded in SAP for a section of the company (e.g. sales or distribution).

This is how you determine valid parameters:

- Log in to the SAP system.

- Type transaction code `CSM` (Material BOM Browser) or `CS11` (Explode BOM).

- Click into the field "BOM application".

- Open the entry help window by clicking the round icon beneath it and choose the desired value.

BOM Usage

This key defines the area (Engineering/design, Production) where a BOM can be used.

To get the valid values, do the following:

- Log in to SAP.

- Type transaction code `CS01` for creating a new BOM (or `CS03` for displaying an existing BOM).

- Click into the field "BOM usage".

- Open the entry help window by clicking the round icon beneath it and choose the desired value.

Using the Alternative SAP BOM Functions

> Note:
>
> SAP offers different sets of function modules for BOM handling. If you encounter problems during BOM handling, it may be necessary to switch to the older set by setting one of the following switches in the T4S4 BOM mapping:
>
> ```
> set ::SAPDat(Bom:Parameter:useOldBomChangeFunction)  TRUE
> set ::SAPDat(Bom:Parameter:useOldBomCreateFunction)  TRUE
> set ::SAPDat(Bom:Parameter:useOldBomFunctions)       TRUE
> ```
>
> For using all old SAP BOM function modules for the complete BOM handling, set the line with `useOldBomFunctions`. For only using the old ones for creating or updating BOMs, set the line with `useOldBomCreateFunction` or `useOldBomChangeFunction`.
>
> The data SAP requires for creating a BOM using the old functions may be different, e.g. you may have to set the "Base quantity":
>
> ```
> set ::SAPDat(Bom:BomHeader:BMENG)  "1"
> ```

Troubleshooting

The message *Update Bill of Material: Usage of 'LEAVE ON TRANSACTION' in Batch input forbidden* means that the BOM already exists in SAP. Use "BOM Change" instead of "BOM Create".

5.8   Maintain Bill Of Material

Maintaining a Bill Of Material prerequisites that there is a BOM created before and is identifiable uniquely by key fields like Material, BOM Usage and Application. These key fields are not modifiable, otherwise a new BOM will be created.

Important Notes on modifying a SAP BOM

CAD Indicator

If a BOM is created or modified in the SAP GUI and then modified again by T4S4, you may find duplicated lines or other unexpected effects. This is the result of the SAP "CAD Indicator", a flag used by SAP to distinguish if the BOM or BOM Line was created from within the SAP GUI or by an external application such as T4S4. Any BOM Line created manually in the SAP GUI does not have this flag set. Only BOM lines created by T4S4 or similar external applications do.

BOM Lines without a CAD Indicator are handled by T4S4 as if they do not exist. They will not be changed by T4S4 and especially not deleted although they may not exist in the Teamcenter BOM. The reason for that behavior is allowing to add BOM positions manually in SAP that are not reflected in Teamcenter.

Creating and Updating BOMs with and without CAD Indicator

There are three different parameters that control the CAD Indicator:

- You can define the overall default setting for newly created BOM Headers and BOM Lines in the function `TC_Object2SAP_BillOfMaterial`:
  `set ::SAPDat(Bom:Parameter:FL_CAD) D`

  - If you set this parameter to `""` all new headers and all new positions will show up without no CAD Indicator.
    The maintain mechanism will take care of BOM Header with and without CAD Indicator and BOM Line without a CAD Indicator.

  - If you set this parameter to `X` all new headers and all new positions will show up with CAD Indicator.
    The maintain mechanism will take care of BOM Header with CAD Indicator and BOM Line with a CAD Indicator.

  - If you set this parameter to `D` the control of the CAD Indicator for newly created BOM Header and BOM Lines is handed over to the dedicated parameters for header and position. This is the default value.
    The maintain mechanism will take care of BOM Header with and without CAD Indicator and BOM Position with and without a CAD Indicator.

  > Caution:
  >
  > If this parameter is defined in the mapping with an empty value (`""`) or `X`, it will always override the following two parameters.
  >
  > We recommend overwriting the default setting for very special cases only. It can lead to unexpected results if it´s not carefully applied and thoroughly tested.

- For the BOM Header, add the following line in the function `TC_Object2SAP_BillOfMaterial`:
  `set ::SAPDat(Bom:BomHeader:CADKZ) X`
  The value `X` enables the CAD Indicator on the Header and `""` disables it.

- For the BOM Lines, add the following line in the function `TC_Object2SAP_BillOfMaterialPosition`:
  `set ::SAPDat(Bom:Position:CADPO:$BomLineIndex) X`
  The value `X` enables the CAD Indicator on the Position and `""` disables it.

Modification of BOM Lines

Another important point to mention is that SAP does not modify the complete BOM during an update, but only modify lines, add lines, or remove lines. Unchanged lines are not affected. The following screenshot shows an example where a BOM was created without ECM, then modified using ECM 500000000416 and then modified again using ECM 500000000417:

Item	ICt	Component	Compone...	Quantity	U.	A.	S..	Valid From	Valid to	Change No.
0010	T		Test text 1	1	PC	☐	☐	31.10.2008	31.12.9999	500000000417
0020	L	2831	EN: t4s test1	2	EA	☑	☐	30.10.2008	31.12.9999	500000000416
0030	L	2830	EN: test	1	EA	☐	☐	30.10.2008	31.12.9999	

So in both modifications, SAP did not modify the BOM position 0030 with the MM ID 2830. T4S4 always transfers the whole set of BOM data to SAP, then SAP evaluates if a line has to be modified. By default, SAP identifies a BOM position (to compare the existing one with the new set of data coming from T4S4) by the combination of the following data in a BOM Line:

- Position number (shown as "Item")

- Position type (shown as "ICt" = Item Category)

- Component (unique identification of the object represented in this BOM Line, in most cases MM or DIR)

- CAD Indicator (not shown here)

This leads to two important points to keep in mind when modifying a SAP BOM containing two or more positions with the same data in those fields via T4S4:

- SAP may give the error message that the BOM position cannot be uniquely identified and refuse the modification

- In some cases SAP may store a modification nevertheless but only one of those lines will be changed correctly. In this case, you may lose data in the existing SAP BOM!

Support of 40 digit Material Numbers

To use a 40 digit Material Number set the following line in the header mapping:

```
set ::SAPDat(Bom:BomHeader:MATNR_LONG) $MaterialNumber40
```

Analog set the following in the position mapping:

```
set ::SAPDat(Bom:Position:MATNR_LONG:$BomLineIndex) $MaterialNumber40
```

> Caution:
>
> Please ask the SAP Administrator how it is implemented for the customer.

Please also see **Create Material Master**.

5.8.1 Maintain Empty Bill Of Material

Handling of empty BOMs can have several effects on the SAP BOM depending on the setting of the parameter to steer the behavior. For example an empty BOM or unpredictable changes of the BOM can be an effect of it. In order to configure T4S4 to handle the empty BOMs as expected, the set of parameters is necessary. The transfer of an empty BOM is aborted showing an ERROR message like: "Bill Of Material transfer to SAP returns with status: EMPTY_BOM_ERROR"

Enable the transfer of an empty BOM

T4S4 will not allow to transfer an empty Teamcenter BOM by default, but create an error instead. This can happen if the view has no position or you skip every BOM position inside the position mapping logic (`return SKIPPED`). In order to allow the transfer of an empty BOM to SAP, set the following in the BOM Header mapping:

```
set ::SAPDat(Bom:Parameter:allowEmptyBomTransfer) YES
```

Automatically skip the creation of an empty BOM

Teamcenter BOM is empty

The T4S4 BOM Workflow handler argument `-no_transfer_of_empty_bom` will skip the whole transfer, but only if the Teamcenter BOM has no positions.

Resulting SAP BOM will be empty

If you want to skip the whole BOM transaction in the Mapping if the resulting SAP BOM would be empty, set the following parameter instead:

```
set ::SAPDat(Bom:Parameter:skipEmptyBomTransfer) YES
```

> Caution:
>
> The default values (used if they are not set explicitly in the mapping) of those two switches are `NO`. Do not set both of them to `YES` because it is a contradiction.

Delete an Empty BOM in SAP

Set the following parameters in the header mapping to delete an existing SAP BOM if no positions are transferred to SAP:

```
set ::SAPDat(Bom:Parameter:allowEmptyBomTransfer) YES
```

```
set ::SAPDat(Bom:Parameter:deleteEmptyBom)      YES
set ::SAPDat(Bom:Parameter:skipEmptyBomTransfer)  NO ;#default value
```

5.8.2 Maintain Plant Specific Bill Of Material

A Group BOM will extended with a plant specific entry. The Maintenance Group BOM has although an effect on the plant specific BOM. They have the same internal ID.

Configure a valid plant in the header mapping:

```
set ::SAPDat(Bom:BomHeader:Plant) 1000
```

5.8.3 Assign Bill Of Material to Plant

A Group BOM will be extended with a plant specific entry. Changes in the Group BOM will also have an effect on the plant specific BOM. They have the same internal BOM IDs.

Use the function `::TPSAP::BOM::BAPI::assignBillOfMaterial2Plant`.

```
::TPSAP::BOM::BAPI::assignBillOfMaterial2Plant $MatNr $Plant \
  $BomUsage $BomAlternative $NewPlant
```

You can also use the test script *sap_bom_assign2plant.tcl*.

5.8.4 Keep manually added positions

Normally a SAP BOM should be in sync with the TC BOM. But in some cases a SAP BOM may contain positions that should not be changed if the BOM is updated by T4S4.

A common use-case is a manual for a technical product. This is typically not part of the Teamcenter BOM but needed in the SAP BOM for the correct packaging. So it is added manually in SAP. When the product structure itself is modified in the Teamcenter BOM and this BOM is transferred to SAP by T4S4, the BOM position for the manual is not removed from the SAP BOM. SAP checks that this position does not have the CAD Indicator set and prevents its modification. To check this flag, double-click the Item number in the SAP BOM. SAP will open the view "BOM: Item: All data". Activate the tab "Basic Data" there and find the "CAD Indicator" in the section "General Data".

So with the default configuration, T4S4 ignores BOM positions without a CAD Indicator in SAP. Only positions that have a CAD indicator will be deleted or changed. This behavior is configurable.

Please also read **Important Notes on modifying a SAP BOM**.

5.8.5   Read Bill Of Material

Bill Of Material

To read a Bill Of Material, use the function `::TPSAP::BOM::readBillOfMaterial`. The **T4S4 API Reference** contains a code example. If you don´t want to read a plant specific BOM, leave the `Plant` parameter empty.

You can also use the test script *sap_get_bomdata.tcl*.

Order Bill Of Material

To read an Order BOM use the function `::TPSAP::ORDERBOM::readBillOfMaterial`. The **T4S4 API Reference** contains a code example. If you don´t want to read a plant specific BOM, leave the `Plant` parameter empty.

You can also use the test script *sap_get_bomdata_csai.tcl*.

5.8.6   Read Plant Specific Bill Of Material

Please see **Read Bill Of Material**.

5.9   Use Engineering Change Master

Using a Change Master means that Header and Item Records are duplicated and stored separately.

> Caution:
>
> Changes to sub-items are not saved historically, even if you make these changes with reference to a Change Number.

5.9.1   ECM with Date Effectivity

The Date Effectivity is the time period in which the BOM Header or BOM Item is valid. This period is delimited by the Valid-From-date and Valid-To-date in the BOM Header and BOM Item.

To use an ECM Number to maintain a BOM set the following line in your BOM Header Mapping. This is the same not matter if you use an ECM with Date Effectivity or an Effectivity Parameter:

```
set ::SAPDat(Bom:BomHeader:ChangeNumber) $ChangeNumber
```

5.9.2   ECM with Effectivity Parameter

The parameter effectivity is the time period in which the BOM Header or BOM Item is valid. This period is delimited by parameter for example the serial no effectivity.

5.10   Maintain Header Data

The Mapping function `TC_Object2SAP_BillOfMaterial` is executed once for the whole BOM. It maintains settings for the BOM Header. The Header data is applied to all items as well.

5.10.1   Set BOM Status

This indicator controls subsequent processing of the BOM in various other organizational areas (for example, Engineering, Costing, MRP).

```
set ::SAPDat(Bom:BomHeader:BomState) "01"
```

5.10.2   Maintain Longtext

Header Longtext can be written by T4S4 and will hold additional information of an item and for example their functionality and specification in the current use.

For BOM there are different types of long texts available:

- BOM Header Longtext

- BOM Header Alternative Longtext

- Item Longtext

> Caution:
>
> SAP only allows up to 132 characters per line in a long text. If you transfer a longer line, the additional characters will be cut off.

For a BOM position, it will only be saved as long text if it consists of at least three lines. If you want to save two lines with 50 characters each, SAP will cut the last 10 characters of both lines!

BOM Header Longtext

The text can be found in SAP GUI in the BOM Header view "Display material BOM": Header overview > BOM Long Text.

Example for BOM item long texts:

```
lappend Longtext "1. Line"
lappend Longtext "2. Line"
lappend Longtext "3. Line"
```

```
lappend Longtext "4. Line"
::T4S::BOM::MAPPING::setBomLongText $Longtext
```

BOM Header Alternative Longtext

The text can be found in SAP GUI in the BOM Header view "Display material BOM": Header overview > BOM Alternative Text

Example for BOM item long texts:

```
lappend Longtext "1. Line"
lappend Longtext "2. Line"
lappend Longtext "3. Line"
lappend Longtext "4. Line"
::T4S::BOM::MAPPING::setAlternativeLongText $Longtext
```

Item Longtext

To find the text for each BOM position in SAP GUI, go to Item: All data > Long Text

Example for BOM item long texts:

```
lappend Longtext "1. Line"
lappend Longtext "2. Line"
lappend Longtext "3. Line"
lappend Longtext "4. Line"
::T4S::BOM::MAPPING::setPositionLongText $BomLineIndex $Longtext
```

5.10.3 Maintain Custom Fields

BOM Header data can be extended by custom fields to give more space for useful or necessary information.

> Caution:
>
> A prerequisite for this functionality is that this field was defined correctly in the corresponding SAP internal BOM interfaces. See SAP functions `CSAP_MAT_BOM_*` and there the definitions of the BOM position data `STPO_API01`, `STPO_API02` and `STPO_API03`. Every modification should be done with the corresponding include `CI_STPO`.
>
> It might happen that T4S4 can not set a SAP BOM field by the standard call although it looks like a standard field. Then try and transfer it this way as if it was a customer specific SAP BOM field.

Example to transfer a vale to a customer defined SAP field `CUSTOM1`:

BOM Header

```
set ::SAPDat(Bom:CustomFields4Header:CUSTOM1) Value1
```

BOM Position

```
set ::SAPDat(Bom:CustomFields4Position:CUSTOM1:$BomLineIndex) Value1
```

5.11 Maintain Position Data

Positions and their data handled independently from the Header data. The function `TC_Object2SAP_BillOfMaterialPosition` is called once for each BOM position. SAP treats material related BOM positions as references to existing Material Masters. So each of those BOM positions has to exist as Material Master before creating the BOM in SAP or the transaction will fail.

The `PositionIndex` is a parameter to the BOM mapping functions. The variable `BomLineIndex` is used additionally in the position mapping function, because of possible index changes. The following example shows how the position number is set according to the Teamcenter BOM sequence number:

```
set ::SAPDat(Bom:Position:CADPO:$BomLineIndex) X
set ::SAPDat(Bom:Position:POSNR:$BomLineIndex) \
    [::T4X::TC::MAPPING::IndexedFieldMapping $PositionIndex \
    BOMLine bl_sequence_no]
```

the BOM Header has always the `PositionIndex` 0, whereas the BOM Positions have the numbers 1...n.

> Caution:
>
> Although an Item Revision Type may be not specified in the preference `T4S_MaterialMasterTypeList` list or `T4S_BillOfMaterialLineMapping4`, T4S4 will try to read it if it is used as a BOM position!
>
> You should configure the BOM Line Mapping to ignore a position if it is of an invalid type. The following code in the BOM position mapping function (TC_Object2SAP_BillOfMaterialPosition) will just skip the desired position. The transaction will end successfully and that position will just be missing in SAP as if it didn't exist in Teamcenter:
>
> ```
> set ValidBomLineList [list ItemType1 ItemType2]
> if {[lsearch $ValidBomLineList $ItemType] eq "-1"} {
>   # this is not a valid position type and should be ignored
>   return "SKIPPED"
> }
> ```

Export Positions Unpacked

Use the Handler-Argument `–unpack_all_bomlines=TRUE`. Please see the **T4S4 API Reference** for more details.

Export Multiple BOM Levels

Use the Handler-Argument `–scan_max_bom_level` to export more than one level of a BOM. See **T4S4 API Reference** for more details.

> Caution:
>
> Only a flat BOM can be transferred to SAP.

Creating additional Positions in SAP

You can create an additional BOM positions in SAP that don´t have a counterpart in the Teamcenter BOM. This might be helpful for example to avoid empty SAP BOMs or to set specific data additionally.

The following example shows how to create a dummy BOM position of type "T" (text). Define a fix position number for this BOM Line. Of course this must be a number that does not exist in any of the existing Teamcenter BOM lines shown in the Structure Manager. Do not forget to increase the `BomLineIndex` or start with a high default value like 9000.

```
set BomLineIndex "9000"
::TPSAP::BOM::initPositionStructure $BomLineIndex
set ::SAPDat(Bom:Position:POSTP:$BomLineIndex) "T" ;# Text item
set ::SAPDat(Bom:Position:POTX1:$BomLineIndex) "BOM updated \
    [clock format [clock seconds] –format "%Y-%m-%d %T"]"
```

BOM Position Default Values

SAP may use default values on BOM Lines, regardless what has been configured in the Mapping. This is controlled by the following switch:

```
set ::SAPDat(Bom:Parameter:FL_DEFAULT_VALUES) X
```

Set the value to `"  "` instead of the default value `X` to disable this behavior.

Use the transaction code `OS21` to check the SAP default value configuration:

Examples for default values depending on the BOM usage:

- Production relevant

- Engineering/design

- Spare part indicator

- Relevant to sales

- CostingRelevncy

5.11.1 Set Item Category

When you create a new BOM item, you must select an item category for the item. The item category defines the features and functions of an item and determines which specific item data is processed and controls further activities in the system.

5.11.1.1 Maintain Material Master (MM) Item (L, N, ...) Data

BOM Items refers for several Item Categories to Material Masters. Material Master hold more detailed information of the used assembly component. T4S4 is able to update the Material Master specification for different BOM items.

```
set ::SAPDat(Bom:Position:POSTP:$BomLineIndex) L
set ::SAPDat(Bom:Position:IDNRK:$BomLineIndex) $MaterialNumber
set ::SAPDat(Bom:Position:MENGE:$BomLineIndex) $Quantity
set ::SAPDat(Bom:Position:POSNR:$BomLineIndex) $Position
set ::SAPDat(Bom:Position:CADPO:$BomLineIndex) "X"
# Optional if alternate unit is maintained:
set ::SAPDat(Bom:Position:MEINS:$BomLineIndex) $UnitOfMeasure
```

5.11.1.2 Maintain Variable Size Item (R) Data

If you want different-sized sections of a material (raw material) to be represented by one material number in BOM items, you use this item category.

Use the following code example:

```
set ::SAPDat(Bom:Position:POSTP:$BomLineIndex) R
set ::SAPDat(Bom:Position:IDNRK:$BomLineIndex) $MaterialNumber
set ::SAPDat(Bom:Position:ROANZ:$BomLineIndex) 1
set ::SAPDat(Bom:Position:POSNR:$BomLineIndex) $Position
set ::SAPDat(Bom:Position:ROMS1:$BomLineIndex) $Length
set ::SAPDat(Bom:Position:ROMS2:$BomLineIndex) $Width  ;# may be empty
set ::SAPDat(Bom:Position:ROMS3:$BomLineIndex) $Height ;# may be empty
set ::SAPDat(Bom:Position:ROMEI:$BomLineIndex) $UnitOfDimensions
set ::SAPDat(Bom:Position:CADPO:$BomLineIndex) X
```

5.11.1.3 Maintain Document Info Record (DIR) Item (D) Data

Document items are supported in all BOM categories (for example Material BOM and Equipment BOM), so that you can document the product or component in detail. In document structures, you can only enter document items and text items.

Use the following code example:

```
set ::SAPDat(Bom:Position:CADPO:$BomLineIndex) X

set ::SAPDat(Bom:Position:POSNR:$BomLineIndex) $Position
set ::SAPDat(Bom:Position:POSTP:$BomLineIndex) D
set ::SAPDat(Bom:Position:DOKNR:$BomLineIndex) $DocumentNumber
```

```
set ::SAPDat(Bom:Position:DOKTL:$BomLineIndex) $DocumentPart
set ::SAPDat(Bom:Position:DOKAR:$BomLineIndex) $DocumentType
set ::SAPDat(Bom:Position:DOKVR:$BomLineIndex) $DocumentVersion
```

5.11.1.4 Maintain Class Item (K) Data

This item category is only supported in configurable BOMs.

5.11.1.5 Maintain Text Item Data

Text Positions are supported in all BOM categories (for example Material BOM and Document BOM).

Use the following code example:

```
set ::SAPDat(Bom:Position:POSTP:$BomLineIndex) T
set ::SAPDat(Bom:Position:POSNR:$BomLineIndex) $Position
set ::SAPDat(Bom:Position:POTX1:$BomLineIndex) "My Text"
set ::SAPDat(Bom:Position:CADPO:$BomLineIndex) X
```

5.11.2 Set Item No (Find No)

The Item No or Find No is a structural resolution indicator of the whole BOM and also key identifier for BOM Lines in SAP.

Code Example:

```
set ::SAPDat(Bom:Position:POSNR:$BomLineIndex) $Position
```

> Note:
>
> SAP only supports up to four digits for a Position Number.
>
> Avoid using Position Numbers with different lengths (e.g. 10, 20, 100...) because this may lead to a sorting problem in the SAP BOM (e.g. 10, 100, 20... instead of 10, 20, 100...). If this should occur, make sure to send the position numbers with four digits e.g. 0010, 0020, 0100).

Using SAP internal BOM position number

By default, the Teamcenter BOM sequence number (Find number) is transferred to the SAP BOM by setting the variable `::SAPDat(Bom:Position:POSNR:$BomLineIndex)`.

In many SAP system it works by just not setting the variable (comment the line out).

```
# set ::SAPDat(Bom:Position:POSNR:$BomLineIndex) $Position
```

If it doesn´t work, additionally set the following in the BOM Header mapping function:

```
set ::SAPDat(Bom:Parameter:useAutoPositionNumbers) YES
```

If this still does not seem to work use the old BOM functions. The correct value for the Parameter `useAutoPositionNumbers` on is `YES` but for `useOldBomFunctions` it is `TRUE`!

```
set ::SAPDat(Bom:Parameter:useOldBomFunctions)      TRUE
set ::SAPDat(Bom:Parameter:useAutoPositionNumbers) YES
```

If `::SAPDat(Bom:Parameter:useAutoPositionNumbers)` is set to `NO` or `::T4S::BOM::MAPPING::CtrlParam(AutoInitOfPositionData)` is set to `FALSE`, you may have to initialize the BOM positions explicitly by calling the function `initPositionStructure` for each BOM Line:

```
::TPSAP::BOM::initPositionStructure $BomLineIndex
```

> Caution:
>
> Do not use the parameter `::SAPDat(Bom:Parameter:useAutoPositionNumbers)` =TRUE to change an existing BOM. The position number will change for every transfer. Because the position number is an unique identifier for the BOM Line, SAP can´t identify which positions should be changed. Instead it may delete the existing lines and create new ones.

5.11.3 Maintain Sub-Item Data

T4S4 supports the update of a SAP BOM sub-position based on the Teamcenter BOM Line attribute for the Reference Designator (`bl_ref_designator`). The idea is to support the synchronization of the Teamcenter BOM Line Reference Designator information with the SAP sub item data for a BOM Line.

The Reference Designator supports the following integration use cases:

• Transfer of Teamcenter BOM Line Occurrence Note Reference Designator information to the SAP BOM sub item structure for a specific BOM Line

• Import SAP BOMs to create a corresponding Teamcenter BOM with Reference Designator defined by the SAP sub item BOM Line information

The following transfer modes support that BOM sub position functionality:

Transfer Mode	Sub Item information based on unpacked BOMLine	Sub Item information based on packed BOMLine
Workflow	X	X
Transfer Job	X	

Transfer Mode	Sub Item information based on unpacked BOMLine	Sub Item information based on packed BOMLine
Import Job	X	X
Interactive without SAP GUI	X	

Using Packed Teamcenter BOM lines

In this case all Reference Designators are defined in one BOM Line separated by a blank or another separator sign. In the mapping you have to separate the Reference Designators. You can find an example in the file *<T4x_GS_ROOT>\var\template\capabilities\T4S\t4s_billofmaterial_mapping_doc.sd* under "Maintain Sub-Item Data".

Using Unpacked Teamcenter BOM lines

Specific Configuration in TC

If the sub item information is based on unpacked BOM lines, then the workflow argument `unpack_all_bomlines=TRUE` is needed in the BOM transfer handler.

For enabling the Reference Designator information transfer to SAP, add the value `bl_ref_designator` to the preference `T4S_BillOfMaterialMappingOccurrenceNotes4<BomViewType>`.

T4S4 Configuration

In this case all Reference Designators are defined in one BOM Line separated by a blank or another separator sign. In the mapping you have to separate the Reference Designators and create for each of them a sub item line with the new `setSAPBomSubPositionStructure` function.

Import Mapping Functions:

In this case the SAP sub item information is provided be an enhanced version of the `::TPSAP::BOM::BAPI::readBillOfMaterial` function the needed information is part of `::BOMResultDat(T_STPU:*)` structure. An example how to identify the sub items for a BOM Line can be found in the *sap_get_bomdata.tc*l test script.

The following example shows how to get the Reference Designator information and how to set the SAP sub-position information. You can find an example in the file *<T4x_GS_ROOT>\var\template\capabilities\T4S\t4s_billofmaterial_mapping_doc.sd* under "Maintain Sub-Item Data".

5.11.4 Maintain Item Longtext

T4S4 is able to write Item Longtext. Longtexts hold additional information of an item and for example their functionality and specification in the current use.

Please see **Maintain Longtext**.

5.11.5   Maintain Item Alternative Group

T4S4 is able to handle different BOM Alternatives. When you create a new BOM, the first BOM Alternative will be created as well.

5.11.6   Maintain Document Info Record Assignment

Creating Object Links to an existing Document Info Record is supported by T4S4.

Please see Maintain Bill Of Material (BOM) Link.

5.11.7   Maintain Item Object Dependency

T4S4 allows to maintain Item Object Dependencies in SAP:

- Local Selection Conditions

- Global Selection Conditions

The condition needs to be read from the Occurrence Note `bl_variant_condition` or `bl_formula`. Then it has to be converted to a format that is supported by SAP.

The corresponding conversion function depends on the used Teamcenter Options an Variants module:

- Classic/Legacy Variant Expression
  `::T4S::BOM::MAPPING::convertTcClassicSelectionCondition2SAP`

- Modular Variant Expression
  `::T4S::BOM::MAPPING::convertTcVariantSelectionCondition2SAP`

- Product Configurator Variant Formula Expression
  `::T4S::BOM::MAPPING::convertTcSelectionCondition2SAP_2`

The function `::TPSAP::BOM::setSelectionCondition` stores the converted SAP BOM Selection Condition parameters into the the global TCL array `::SAPDat(Bom:Dependency:*)` for the BOM transfer. The same function can be used for local and global Selection Conditions. Please see T4S4 API Reference on how to set the correct parameters.

The function `maintainBOMLineSelectionCondition` in the file *t4s_bom_mapping_template.sd* shows how to use the functions properly. You can also find an example in the file *<T4x_GS_ROOT>\var\template\capabilities\T4S\t4s_changemaster_mapping_doc.sd* under "Maintain Item Object Dependency".

5.11.8   Maintain Custom Position Fields

BOM Position data can be extended by custom fields to give more space for useful or necessary information.

Please see **Maintain Custom Fields**.

5.11.9   Extend Plant Specific Component to a Target Plant

Materials have to maintained in a given plant to use them into a plant specific BOM. In order to use a Material in a plant specific BOM, the extension of the Material to the target plant is necessary.

5.11.10   Read Position Data

Please see **Read Bill Of Material**.

6. Change Transfer - Engineering Change Master

6.1 Introduction

Please read the **Teamcenter Gateway - Generic Configuration Guide** first to understand the basics for this chapter.

Using Teamcenter Change Management, the Teamcenter change revisions can be used as workflow targets. T4S4 can be set to process configured objects in the folders ("impacted items" and "solution items") automatically and transfers all found and configured objects to SAP. This way of working provides a more tight connection between Teamcenter and SAP Change Management. This is described in the **Teamcenter Gateway - Generic Configuration Guide**.

> Caution:
>
> SAP does not allow the transfer of ECM data with the T4S Gateway Menu options "Create" (ECM_CREATE) and "Change" (ECM_CHANGE). If you start those transactions you will not see your Teamcenter data in SAP, the ECM will be created with default values. "Create Direct" and "Change Direct" pass the data correctly.

Nomenclature

Please also see the **Glossary**.

Teamcenter	SAP
ItemRevision or ChangeItemRevision	Change Master (ECM)
Relation	Object Link
Release Status	Status of Change Number
	Release Key

T4S4 Specifics Compared to the Generic Mapping

SAP Transaction Codes	
CC01	Create Change Master
CC02	Change Change Master
CC03	Display Change Master

Export	
TargetTypeName	ChangeMaster
Mapping template	*t4s_ecm_mapping_template.sd*
Namespace Mapping	`::T4S::ECM::CUSTOM::MAPPING`
Preferences start with	T4S_ChangeMaster
Handler	T4S-transfer-ChangeMaster
	T4S-validate-ChangeMaster
	See the **T4S4 API Reference** for more details.
Job Type	`T4S_EXPORT_ECM`

Import Job	
Job Type	`T4S_IMPORT_ECM`
Mapping template	*t4s_ecm_item_import_mapping_template.sd*

Relationship between TC and SAP Objects

Teamcenter and SAP both offer a Change Management as functionality to bundle several changes in one object. The goal is to activate different modifications on several engineering objects (e.g. Bill Of Material) at the same time without having to process them all manually. In SAP there are no revisions for a Change Master. Therefore the relation between TC Item Revision and SAP Change Master is n:1.

6.2 Global Preferences_for_ECM

To define the data mapping from Teamcenter to SAP the `TargetTypeName` needs to be replaced by `ChangeMaster`. The prefix of the preference is `T4S_`. Please see also **Teamcenter Gateway - Generic Configuration Guide**.

- `T4S_ChangeMasterTypeList`

- `T4S_ChangeMaterMapping4`

- `T4S_ChangeMasterMapping2`

- `T4S_ChangeMasterFieldMapping2` (optional)

- Optional preference to enable only selected Item Revision Types for the ECM Display feature in the T4S Gateway Menu:

```
T4S_ChangeMasterDisplayableTypeList=
ItemRevision
```

Definition of Item Revision Types which should be supported

The following preference example shows the complete configuration for the Change Master data mapping:

Example definition for two Item Revision types for usage with T4S4

```
T4S_ChangeMasterTypeList=
SAP2T4S_AttrItemRevision
ItemRevision
```

Definition of Related Teamcenter Objects that contribute Relevant Attributes to the Mapping

This step defines which data needs to be read from the Teamcenter source object. The objects have to be defined for each Item Revision Type. It is important to limit the amount of data to the data that you really need to get best performance for your transaction. Any data that is not required for the transaction should not be read!

Specify the related TC data objects from which you want to pass data to SAP by T4S4.

```
T4S_ChangeMasterMapping4ItemRevision=
items_tag:Item
items_tag:Item:#__getAllProperties__#:Properties
items_tag:Item:IMAN_master_form:Item Master
#__getAllProperties__#:Properties
IMAN_master_form:ItemRevision Master
IMAN_classification:icm0

T4S_ChangeMasterMapping4SAP2T4S_AttrItemRevision=
items_tag:SAP2T4S_AttrItem
IMAN_master_form:SAP2T4S_AttrItemRevision Master
```

Definition of Related Teamcenter Objects to store back Information

Example for the configuration of the reverse mapping structure:

```
T4S_ChangeMasterFieldMapping2IMAN_master_form:ItemRevision Master=
SAPMatNo
SAPMatNoReserved
SentMMToSAP
user_data_1
```

Definition of Related Fields on Objects to store back Information

Example for the configuration of the reverse mapping structure:

```
T4S_ChangeMasterFieldMapping2ItemRevision=
object_desc

T4S_ChangeMasterFieldMapping2items_tag:Item=
item_id

T4S_ChangeMasterFieldMapping2items_tag:Item:IMAN_master_form:Item Master=
user_data_1
```

6.3   Create Engineering Change Master

Creation of a Change Master for the first time. The TC object will always be called "ItemRevisionType", no matter if it is a Teamcenter Change object or an Item Revision.

Set Effectivity Type

The default Effectivity Type for an ECM is Date Effectivity (value `""`).

If you want so set a parameter effectivity instead set the effectivity type with the following code example:

```
set ::SAPDat(ChangeMaster:HeadData:EFFECTIVITY_TYPE) $Effectivity
```

Set Function Type

There is an additional parameter `FUNCTION` which is not needed by default:

```
set ::SAPDat(ChangeMaster:HeadData:FUNCTION) ""
```

It allows the following settings:

- `""` (empty) Without Release Key

- `1` - With Release Key, which has to be defined for the ECM additionally:
  ```
  set ::SAPDat(ChangeMaster:HeadData:RELEASE_KEY) $Release_key
  ```

- `2` - Change Package

- `3` - Change Unit

Create an ECM with Date Effectivity

This is the default case and included in the mapping template.

```
# Date format can be different
set ValidFromDate [clock format [clock seconds] -format "%d.%m.%Y"]
set ::SAPDat(ChangeMaster:HeadData:CHANGE_NO)   $ChangeNumber
set ::SAPDat(ChangeMaster:HeadData:VALID_FROM) $ValidFromDate
set ::SAPDat(ChangeMaster:HeadData:STATUS)      01
set ::SAPDat(ChangeMaster:HeadData:DESCRIPT)    $Description
set ::SAPDat(ChangeMaster:HeadData:REASON_CHG) $ChangeReason
```

> Caution:
>
> SAP will refuse the transaction if you try and set a `VALID_FROM` date in the past.
>
> SAP may require the `VALID_FROM` date in a different format than the example. In such a case (SAP error message "Invalid date") check the SAP internal format by using the test script "SAP get ECM data". Check the result for the line with `ECMResultDat(VALID_FROM)`. If the value is for example `2015-07-29`, then T4S4 has to send the date in the same format, so set the date format code "`%Y-%m-%d`" instead of "`%d.%m.%Y`"
>
> It is not possible to create an ECM with an ECM profile.

Create an ECM with Parameter Effectivity

The existing default types for Parameter Effectivity in SAP are Serial No. (`SERNR`) and Date (`DATE`).

This example shows how to create an ECM with Parameter Effectivity for serial numbers.

```
set ::SAPDat(ChangeMaster:HeadData:EFFECTIVITY_TYPE) SERNR
incr Index
# Material is mandatory if you pass any other value
set ::SAPDat(ChangeMaster:Effectivity:MATERIAL:$Index)      ""
set ::SAPDat(ChangeMaster:Effectivity:SERIALNR_LOW:$Index)  ""
set ::SAPDat(ChangeMaster:Effectivity:SERIALNR_HIGH:$Index) ""
set ::SAPDat(ChangeMaster:Effectivity:SERNR_OI:$Index)      ""
#X for open interval:
set ::SAPDat(ChangeMaster:Effectivity:FL_DELETE:$Index)     ""
# SAP business logic - either set open interval or upper limit
if {$::SAPDat(ChangeMaster:Effectivity:SERIALNR_HIGH:$Index) ne ""} {
   set ::SAPDat(ChangeMaster:Effectivity:SERNR_OI:$Index)      ""
}
```

This example shows how to create an ECM with Parameter Effectivity for dates:

```
set ::SAPDat(ChangeMaster:HeadData:EFFECTIVITY_TYPE) DATE
incr Index
set ValidFromDate [clock format [clock seconds] -format "%d.%m.%Y"]
set ::SAPDat(ChangeMaster:Effectivity:VALID_FROM:$Index)  $ValidFromDate
set ::SAPDat(ChangeMaster:Effectivity:VALID_TO:$Index)    ""
set ::SAPDat(ChangeMaster:Effectivity:DATE_MARK:$Index)   X
#X for open interval:
set ::SAPDat(ChangeMaster:Effectivity:FL_DELETE:$Index)   ""
# SAP business logic - either set open interval or upper limit
if {$::SAPDat(ChangeMaster:Effectivity:VALID_TO:$Index) ne ""} {
   set ::SAPDat(ChangeMaster:Effectivity:DATE_MARK:$Index)   ""
}
```

> **Caution:**
>
> Don´t confuse Parameter Effectivity "Date" with Date Effectivity.

Create the ECM as a copy of an existing ECM

Up to now, there is no special function to support it (similar to the Reference Material functionality). Instead, you may just read all the data of an existing ECM and set those to the new ECM. You can find an example in the file *<T4x_GS_ROOT>\var\template\capabilities\T4S\t4s_changemaster_mapping_doc.sd* under "Create Change Master".

6.4   Update Engineering Change Master (ECM)

If the Change Master has already been created in SAP, each following transaction will be ECM Update by default. An overview of the maintainable SAP fields can be found in the Change Master Mapping template.

6.5   Maintain Header Data

Please see Chapter **Create Engineering Change Master**.

6.6   Maintain Object Types

Unless defined explicitly, T4S4 only creates Engineering Change Masters that are valid for the object types Material, standard BOM and DIR (because those are the most frequently used ones). Call the following function as described in the **T4S4 API Reference**. You need a separate call for each ECM object type:

```
::TPSAP::ECM::setObjectTypeStructure
```

> Caution:
>
> If you specify an object type explicitly, the default configuration (MM, BOM, DIR) is not active any more.

Checking the Configured Object Types of a SAP ECM

You can check them by clicking the "Object Types" button in a SAP ECM Display or Change transaction. In this example Material is the only configured type:

Depending on the version and configuration of your SAP system and SAP GUI, this list may look completely different. For example, if you find the line "Task List" as shown here in the screenshot, there will be no other "Task List" whereas other SAP GUI might show "General Task List" and "Equipment Task List", but not the default "Task List". This is similar with the BOM types. To switch between those two modes, select **Edit – More... – Object Subtypes** in the ECM main menu. With the settings as shown here, you fill find the complete list. If the options are deactivated, SAP will show the list compressed.
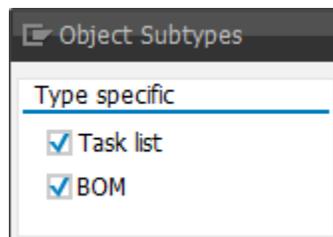
6.7   Maintain Object Management Records

SAP will maintain the Object Management Record between an ECM and another SAP object when it is asked to set the "ECM used object description". This is a convenient way to create such a link between the currently processed object and an ECM without the need to change the current object using the stated ECM. "Using" in that context could mean for example for an MM: set the SAP MM Revision Level. SAP ECM allows storing a description to each used object.

In a SAP transaction to display or change a Change Master, click the button "Objects" to find the list of all the objects that have been modified using this ECM. This shows one line for each used object with the following data (and maybe more columns, depending on your SAP configuration):

- Object Type
  E.g. "BILLOFMATERIAL" for a material BOM , Refer to **T4S4 API Reference** for other Object Types.

- Object
  I.e. the SAP object identifier.

- Description

E.g. "Test ECM item description".

In order to set this description, call the
function `::TPSAP::ECM::updateObjectManagementRecord` in the mapping file of the object type
that is created or modified using an ECM (available for MM, DIR and BOM).

> Caution:
>
> The ECM has to be created in SAP first.

You can find an example for each transfer in the file *<T4x_GS_ROOT>\var\template\capabilities\T4S
\t4s_changemaster_mapping_doc.sd* under "Maintain ECM Data".

In principle there is an easier method to set this ECM object description. But it cannot handle the SAP
object ID separately (as it is necessary to provide it in internal format for this special update but in the
external formal for the standard update). It will only work correctly if the internal and external SAP
object IDs are the same (which is often not the case):

```
# MM Mapping:
set ::SAPDat(Material:AffectedItem:Description) "ECM MM item desc"
# BOM Mapping:
set ::SAPDat(Bom:AffectedItem:Description) "ECM BOM item desc"
# DIR Mapping:
set ::SAPDat(Document:AffectedItem:Description) "ECM DIR item desc"
```

6.7.1 Assign Object Management Records to Alternate Dates

This Chapter shows how to assign an Object Management Record to an Alternate Date.

> Caution:
>
> The Object Management Record has to be created first, e.g. the Object needs to be transferred
> with a Change Number.

Example Code for a Material Master:

```
set ::SAPDat(ChangeMaster:PARAMETER:ALT_DATE)          $AltDate
set ::SAPDat(ChangeMaster:PARAMETER:CHANGE_NO)         $ChangeNo
set ::SAPDat(ChangeMaster:PARAMETER:CHG_OBJECT_TYPE)   "4"
set ::SAPDat(ChangeMaster:PARAMETER:MATERIAL)          $MaterialNumber
::TPSAP::ECM::ChangeMasterAssignObject2AltDate
```

Example Code for a Bill Of Materials

```
set ::SAPDat(ChangeMaster:PARAMETER:ALT_DATE)         $AltDate
set ::SAPDat(ChangeMaster:PARAMETER:CHANGE_NO)        $ChangeNo
set ::SAPDat(ChangeMaster:PARAMETER:CHG_OBJECT_TYPE)  "1"
set ::SAPDat(ChangeMaster:PARAMETER:BOM_CATEGORY)     "M"
set ::SAPDat(ChangeMaster:PARAMETER:MATERIAL)         $sMaterialNumber
set ::SAPDat(ChangeMaster:PARAMETER:BOM_USAGE)        $Usage
set ::SAPDat(ChangeMaster:PARAMETER:PLANT)            $Plant
::TPSAP::ECM::ChangeMasterAssignObject2AltDate
```

Example Code for a Document Info Record:

```
set ::SAPDat(ChangeMaster:PARAMETER:ALT_DATE)         $AltDate
set ::SAPDat(ChangeMaster:PARAMETER:CHANGE_NO)        $ChangeNo
set ::SAPDat(ChangeMaster:PARAMETER:CHG_OBJECT_TYPE)  "3"
set ::SAPDat(ChangeMaster:PARAMETER:DOCUMENT_TYPE)    $DocumentType
set ::SAPDat(ChangeMaster:PARAMETER:DOCUMENT_NUMBER)  $DocumentNumber
set ::SAPDat(ChangeMaster:PARAMETER:DOCUMENT_VERSION) $DocumentVersion
set ::SAPDat(ChangeMaster:PARAMETER:DOCUMENT_PART)    $DocumentPart
::TPSAP::ECM::ChangeMasterAssignObject2AltDate
```

6.7.2 Read Object Management Records

Specific Configuration in SAP

The Transport Package /TESISPLM/T4S_ECM needs to be imported to SAP. Please see **Teamcenter Gateway for SAP S/4HANA - S/4HANA Preparation Guide**.

T4S4 Configuration

Use the following function to get a list of the ECM Object Management Records:

```
::TPSAP::ECM::getChangeMasterObjMgRecList
```

6.8 Maintain Alternate Dates

Multiple Alternate Dates for a given Change Master can be defined.

• Creating an Alternate Date in the Change Master Mapping

• Creating an Alternate Date in a separate transfer

Creating an Alternate Date in the Change Master Mapping

Add the following code example to the Change Master Mapping, increase the index for each new Alternate Date.

```
set Index "1"
set ::SAPDat(ChangeMaster:AltDates:ALT_DATE:$Index)   $AltDateIdent
set ::SAPDat(ChangeMaster:AltDates:VALID_FROM:$Index) $Date
```

Creating an Alternate Date in a Separate Transfer

This code can be executed in a mapping function to create a single Alternate Date.

> Caution:
>
> The Change Master needs to be created first.

```
set ::SAPDat(ChangeMaster:PARAMETER:CHANGE_NO)    $ChangeNumber
set ::SAPDat(ChangeMaster:PARAMETER:ALT_DATE)     $AltDateIdent
set ::SAPDat(ChangeMaster:PARAMETER:VALID_FROM)   $Date
::TPSAP::ECM::ChangeMasterCreateAltDate
```

6.8.1 Read Alternate Dates

This function will return a list of the Alternate Dates attached to a Change Master.

```
::TPSAP::ECM::getChangeMasterAltDatesList
```

6.9 Set Status

The only special value a standard SAP system needs for ECM is the Status:

```
set ::SAPDat(ChangeMaster:HeadData:STATUS) "01"
```

In most SAP systems, "1" (or "01") means "Active" and should be used for ECM transactions. Find this as follows:

- Log in to SAP

- Type transaction code `CC02` for modifying an existing ECM

- Click into the field "Change no. status"

- Open the entry help window by clicking the round icon beneath it.

Chg.no.st ▲	Chg poss	Date chg	Distr.lock	Description
01	✓	✓	☐	Active
02	☐	☐	☐	Inactive

Status of change number (1)   2 Entries found
Restrictions

6.10 Maintain Classification

The classification for ECM works the same way like the MM Classification. The only difference is the SAP Class Type. It is `025` ("Engin. change mgmt") instead of `001` (Material class).

6.11 Maintain Release Key

This is an example on how to maintain the Release Key.

```
set ::SAPDat(ChangeMaster:HeadData:FUNCTION)      1
set ::SAPDat(ChangeMaster:HeadData:RELEASE_KEY)  01
```

6.12 Maintain Custom Fields

T4S4 is able to transfer data to custom fields on SAP Change Master.

Specific Configuration in SAP

The custom fields must be added to the include structure of AENR_API01. In the example the field is called `ZZ0T_CC_PROFILE`
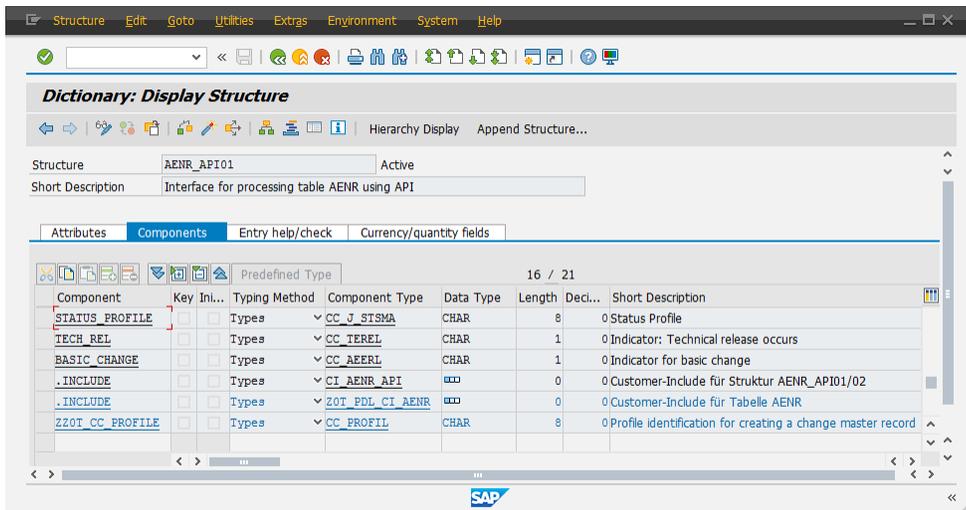
T4S4 Configuration

Now you can set those fields in the mapping file. Example:

```
set ::SAPDat(ChangeMaster:HeadData:ZZ0T_CC_PROFILE) CNCR
```

6.13 Maintain ECM Longtext

SAP will set the ECM description as the long text if it is longer than 40 characters.

```
set ::SAPDat(ChangeMaster:HeadData:DESCRIPT) $MyLongText
```

6.14 Read Engineering Change Master

To read the data of an Engineering Change Master in SAP, you can use the following function:

```
::TPSAP::ECM::getChangeMasterInfo
```

Depending on the parameters you will also get more than the basic information.

You can use the test script *sap_get_ecmdata.tcl* to read the data of an ECM. It uses both functions and can serve as an example.

6.15 Read Effectivity Value

To read the data of an Engineering Change Master in SAP, you can use the following function:

```
::TPSAP::ECM::getChangeMasterEffectivity
```

7. Equipment Transfer

7.1 Introduction

Please read the **Teamcenter Gateway - Generic Configuration Guide** first to understand the basics for this chapter.

Nomenclature

Please also see the **Glossary**.

Teamcenter	SAP
Equipment	Equipment
Relation	Object Link

T4S4 Specifics Compared to the Generic Mapping

SAP Transaction Codes	
IE01	Create Equipment
IE02	Change Equipment
IE03	Display Equipment

Export	
TargetTypeName	EquipmentMaster
Mapping template	*t4s_equi_mapping_template.sd*
Namespace Mapping	`::T4S::EQUI::CUSTOM::MAPPING`
Preferences start with	T4S_EquipmentMaster
Handler	T4S-transfer-EquipmentMaster
	T4S-validate-EquipmentMaster
	See **T4S4 API Reference** for more details.
Job Type	`T4S_EXPORT_EQUI`

Import Job	
Job Type	`T4S_IMPORT_EQUI`
Mapping template	*t4s_equi_item_import_template.sd*

Relationship between TC and SAP Objects

In SAP there are by default no revisions for a Equipment Master. Therefore the relation between TC and SAP Equipment is n:1.

7.2   Export

An Equipment is a physical tool, gadget or machine that is needed for manufacturing. That can be a primitive screw driver as well as a large CNC machine. The SAP Equipment object describes the tools physical characteristics.

> Caution:
>
> The following two issues are SAP internal restrictions for the Equipment transactions that open a SAP window ("Create" and "Change"):
>
> - It is not possible to hand data to SAP. The SAP GUI window is opened but it contains only its own default data (or the ones that were stored there before), not those from Teamcenter.
>
> - When the SAP object is locked (e.g. because it is modified by another user), T4S4 there is no error message and no update.
>
> The Equipment transactions without opening a SAP window ("Create Direct" and "Change Direct") work as expected.

Configuration of Equipment data mapping from Teamcenter to SAP

The following example shows how to configure a Teamcenter Item Revision (here of type "Functionality") that is used for Equipment transfer:

```
T4S_EquipmentMasterTypeList=
FunctionalityRevision
```

Specify the source data objects for the mapping to the SAP Equipment Master:

```
T4S_EquipmentMasterMapping4FunctionalityRevision=
items_tag:Functionality
items_tag:Functionality:IMAN_master_form:Functionality Master
items_tag:Functionality:#__getAllProperties__#:Properties
IMAN_master_form:FunctionalityRevision Master
#__getAllProperties__#:Properties
IMAN_classification:icm0
```

Definition of Equipment data mapping from Teamcenter to SAP

For mapping of Teamcenter data to the SAP Equipment, T4S4 provides a data structure analogous to the SAP Equipment Master. This means that all fields which can be addressed via BAPIs can be filled with values from Teamcenter. T4S4 supports the following SAP data structures:

- Number (ID) of the Equipment Master to be created

- Valid-From Date for Equipment to be created

- Installation Date for Equipment

- General Data for Technical Objects

- Equipment specific data

The examples below show data mapping of Equipment Number and Category:

```
set ::SAPDat(EQUI:PARAMETER:EQUIPMENT)        $Equipment Number
set ::SAPDat(EQUI:DATA_SPECIFIC:EQUICATGRY) "M" ; # Equipment Category
```

If you want to manage an Equipment not only as an independent and individual object, but also as a Material within the framework of Inventory Management, you must provide it with serialization data:

- A unique combination of material and serial number

- Stock information

- Warehouse information

By transferring this information, you describe the piece of equipment as an individual item of a particular material.

8. Functional Location Transfer

8.1 Introduction

Please read the **Teamcenter Gateway - Generic Configuration Guide** first to understand the basics for this chapter.

Nomenclature

Please also see the **Glossary**.

Teamcenter	SAP
Item Revision (e.g. Work Area)	Functional Location
Relation	Object Link

T4S4 Specifics Compared to the Generic Mapping

SAP Transaction Codes	
IL01	Create Functional Location
IL02	Change Functional Location
IL03	Display Functional Location

Export	
TargetTypeName	FunctionalLocation
Mapping template	*t4s_funcloc_mapping_template.sd*
Namespace Mapping	`::T4S::FUNCLOC::CUSTOM::MAPPING`
Preferences start with	T4S_FunctionalLocation
Handler	T4S-transfer-FunctionalLocation T4S-validate-FunctionalLocation See **T4S4 API Reference** for more details.
Job Type	`T4S_EXPORT_FUNCLOC`

Import Job	
Job Type	`T4S_IMPORT_FUNCLOC`
Mapping template	*t4s_funcloc_item_import_template.sd*

Relationship between TC and SAP Objects

In SAP there are by default no revisions for a Functional Location. Therefore the relation between TC Item Revision and SAP Functional Location is n:1.

8.2 Export

A Functional Location describes the location or logical function of a tool (machine...).

> Caution:
>
> The following issues are SAP internal restrictions for the Functional Location transactions that open a SAP window via the T4S Gateway Menu ("Display", "Create" and "Change"):
>
> - It is not possible to hand data to SAP except from the data to identify the object. The SAP window is opened but it contains only its own default data, not those from Teamcenter. The transactions without opening a SAP window ("Create Direct" and "Change Direct") work as expected (similar to what you see with an MM transaction).
>
> - Different from most SAP object types, a "Display" without a valid Functional Location ID (no number given or the Functional Location with this ID does not exist in this SAP system), does not open a SAP GUI at all. The transaction ends with the SAP error message "Functional location <ID> does not exist".

Configuration of Functional Location data mapping from Teamcenter to SAP

The following example shows how to configure a Teamcenter Item Revision (here of type "Functionality") that it is used for a SAP Functional Location:

```
T4S_FunctionalLocationTypeList =
FunctionalityRevision

T4S_FunctionalLocationDisplayableTypeList =
FunctionalityRevision
```

Specify the source data objects for the mapping to the SAP Functional Location:

```
T4S_FunctionalLocationMapping4FunctionalityRevision=
items_tag:Functionality
items_tag:Functionality:IMAN_master_form:Functionality Master
items_tag:Functionality:#__getAllProperties__#:Properties
IMAN_master_form:FunctionalityRevision Master
#__getAllProperties__#:Properties
IMAN_classification:icm0
```

Definition of Functional Location data mapping from Teamcenter to SAP

For data mapping of Teamcenter data to the SAP Functional Location, T4S4 provides a data structure analogous to the SAP Functional Location Master. This means that all fields which can be addressed via BAPIs can be filled with values from Teamcenter. T4S4 supports the following SAP data structures:

- Number (ID) of the Functional Location to be created

- General Data for Technical Objects

- Functional Location specific data

The examples below show data mapping of Functional Location number and category:

```
set ::SAPDat(FUNCLOC:PARAMETER:FUNCLOC)        $FuncLocNumber
set ::SAPDat(FUNCLOC:DATA_SPECIFIC:CATEGORY)  S
```

9. Business Partner Transfer

9.1 Introduction

Please read the **Teamcenter Gateway - Generic Configuration Guide** first to understand the basics for this chapter.

Nomenclature

Please also see the **Glossary**.

Teamcenter	SAP
Vendor	Business Partner
Relation	Object Link

T4S4 Specifics Compared to the Generic Mapping

SAP Transaction Codes	
BP	Business Partner

Export	
TargetTypeName	BusinessPartner
Mapping template	*t4s_bp_mapping_template.sd*
Namespace Mapping	`::T4S::BUSINESS_PARTNER::CUSTOM::MAPPING`
Preferences start with	T4S_BusinessPartner
Handler	T4S-transfer-BusinessPartner
	T4S-validate-BusinessPartner
	See **T4S4 API Reference** for more details.
Job Type	`T4S_EXPORT_BUSINESS_PARTNER`

Import Job	
Job Type	`T4S_IMPORT_BUSINESS_PARTNER`
Mapping template	*t4s_bp_item_import_template.sd*

Relationship between TC and SAP Objects

In SAP there are by default no revisions for a Business Partner. Therefore the relation between TC Item Revision and SAP Business Partner is n:1.

9.2 Export

The term Business Partner refers to a company that sells any items which may be for example used as raw materials in your own production. From an integration point of view, the most important data are their address and contact information.

> **Caution:**
>
> If you migrate from T4S to T4S4: please note that he Vendor module in SAP no longer exists in SAP S/4HANA®. It has been replaced by the Business Partner which can assume the role of a Vendor. You have to migrate the Vendor mapping by manually to the Business Partner Mapping.

Configuration of Business Partner Data Mapping from Teamcenter to SAP

The following example shows the preferences for configuring a Teamcenter Item Revision (here of Item Type "Vendor") that it is used for a SAP Vendor:

```
T4S_Business PartnerTypeList=
VendorRevision
```

Specify the data T4S4 has to read from Teamcenter:

```
T4S_Business PartnerMapping4VendorRevision=
items_tag:Vendor
vendor_role_info:ManufacturerInfo
items_tag:Vendor:#__getAllProperties__#:Properties
```

Definition of Business Partner Data Mapping from Teamcenter to SAP

For setting data to the SAP Business Partner, T4S4 provides amongst others the following:

- Business Partner Number (ID)

- Complete address data (city, street...)

The following example shows the data mapping of the Business Partner:

```
set ::SAPDat(BusinessPartner:PARAMETER:BUSINESSPARTNEREXTERN)
$SapBusinessPartnerId
set ::SAPDat(BusinessPartner:PARAMETER:PARTNERCATEGORY)        "1"
set ::SAPDat(BusinessPartner:CENTRALDATAPERSON:FIRSTNAME)      "Konrad"
set ::SAPDat(BusinessPartner:CENTRALDATAPERSON:LASTNAME)       "Hammerer"
set ::SAPDat(BusinessPartner:CENTRALDATA:SEARCHTERM1)
```

```
"SEARCHTERM1"
set ::SAPDat(BusinessPartner:CENTRALDATA:SEARCHTERM2)
"SEARCHTERM2"
set ::SAPDat(BusinessPartner:CENTRALDATAPERSON:CORRESPONDLANGUAGEISO)
"DE"
set ::SAPDat(BusinessPartner:ADDRESSDATA:STANDARDADDRESS)     "X"
set ::SAPDat(BusinessPartner:ADDRESSDATA:CITY)               "Munich"
set ::SAPDat(BusinessPartner:ADDRESSDATA:POSTL_COD1)         "81739"
set ::SAPDat(BusinessPartner:ADDRESSDATA:STREET)
"Otto-Hahn-Ring"
set ::SAPDat(BusinessPartner:ADDRESSDATA:HOUSE_NO)           "6"
set ::SAPDat(BusinessPartner:ADDRESSDATA:COUNTRYISO)         "DE"
```

> **Caution:**
>
> In order to change the address, you need to maintain a separate array pattern as changing a SAP business partner address is done using a separate call in SAP. This is done in the function `TC_Object2SAP_BusinessPartnerAddress`.
>
> The Vendor role needs to be applied to the Business Partner by the function `TC_Object2SAP_BusinessPartnerRole`.

Please find more predefined Business Partner mapping data in the delivered mapping template file as well as a list of possible data to set at the end of the file.
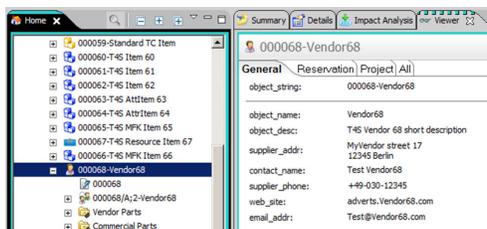
In Teamcenter, the Vendor attribute `supplier_addr` is one field for the whole address. This data has to be separated before you can transfer it to SAP.

Example how the data is stored in SAP:

Business Partner	92	👤 Konrad Hammerer / 81739 Munich
Display in BP role	FLVN00 Vendor	

| Address | Address Overview | Identification | Control | Payment Transactions | Status | Employment Data | Vendor: General Data |

Name

Title	0002 Mr.
First name	Konrad
Last name	Hammerer
Correspondence lang.	DE German

Search Terms

Search Term 1/2	ID12	NO85

Special Customer

☐ VIP
☐ Undesirable Customer   Reason Undes.
Comment

Standard Address

🖨 Print Preview

Street Address

Street/House number	Otto-Hahn-Ring	6
Postal Code/City	81739	Munich
Country	DE Germany Region	

10. Purchase Info Record Transfer

10.1 Introduction

Please read the **Teamcenter Gateway - Generic Configuration Guide** first to understand the basics for this chapter.

Nomenclature

Please also see the **Glossary**.

Teamcenter	SAP
Item Revision	Purchasing Info Record

T4S4 Specifics Compared to the Generic Mapping

SAP Transaction Codes	
ME11	Create Purchasing Info Record
ME12	Change Purchasing Info Record
ME13	Display Purchasing Info Record

Export	
TargetTypeName	PurchaseInfoRecord
Mapping template	*t4s_pir_mapping_template.sd*
Namespace Mapping	`::T4S::PIR::CUSTOM::MAPPING`
Preferences start with	T4S_PurchaseInfoRecord
Handler	T4S-transfer-PIR
	T4S-validate-PIR
	See **T4S4 API Reference** for more details.
Job Type	`T4S_EXPORT_PIR`

Import Job	
Job Type	`T4S_IMPORT_PIR`
Mapping template	*t4s_pir_item_import_template.sd*

Relationship between TC and SAP Objects

In SAP there are by default no revisions for a Purchase Info Record. Therefore the relation between TC Item Revision and SAP Purchase Info Record is n:1.

10.2 Export

In SAP, a Purchase Info Record (PIR) contains purchasing information of specific material: e.g. the Business Partner, the contact person with this Business Partner, the order unit and the typical delivery time.

The starting point in Teamcenter is a Commercial Part and a Manufacturer Part with the Business Partner information. Relate the Commercial Part with the relation `commercialparts` to the Manufacturer Part, so that it looks similar to the following screenshot:
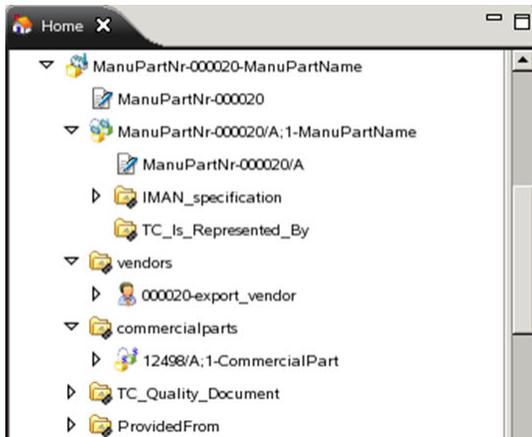
In order to create or change a Purchase Info Record, T4S4 uses the SAP function module `/TESISPLM/ T4S_CALL_TRANSACTION`. You have to deploy the transport package `TESISPLM_T4S_CALL_TRANSACTION`, see chapter *SAP Transport Packages for T4S4 Functionality* in the **Teamcenter Gateway for SAP S/4HANA - S/4HANA Preparation Guide**.

> Caution:
>
> Note that in the Teamcenter dialog "File – New – Part", the type name shown may be "Vendor Part" instead of "Manufacturer Part". They may be used interchangeably.
>
> The PIR processing is supported in Workflows only. There is no PIR functionality to be started from the T4S Gateway Menu in the Teamcenter client.

11. Manufacturing Transfer

11.1 Introduction

Please read the **Teamcenter Gateway - Generic Configuration Guide** first to understand the basics for this chapter.

Nomenclature

Please also see the **Glossary**.

Teamcenter	SAP
Routing	Routing
Production Order	Production Order
Manufacturing BOM	Manufacturing BOM
Inspection Plan	Inspection Plan

T4S4 Specifics Compared to the Generic Mapping

SAP Transaction Codes	
CA01	Create Routing
CA02	Change Routing
CA03	Display Routing
CO01	Create Production Order
CO02	Change Production Order
CO03	Display Production Order
QA01	Create Inspection Plan
QA02	Change Inspection Plan
QST04	Display Inspection Plans

Export	
TargetTypeName	-
Mapping template	*t4s_plmxml2_bom_mapping_template.sd* *t4s_plmxml2_inspectionplan_mapping_template.sd* *t4s_plmxml2_routing_mapping_template.sd*

Namespace Mapping	`::T4S::PLMXML2::BOM::CUSTOM::MAPPING`
	
	`::T4S::PLMXML2::INSPECTIONPLAN::CUSTOM::MAPPING`
	
	`::T4S::PLMXML2::ROUTING::CUSTOM::MAPPING`

Export	
Preferences start with	-
Handler	T4S-add-Form2CC
	T4S-create-AIObject4CC
	T4S-transfer-Routing
	T4S-transfer-ProductionOrder
	T4S-transfer-CC-BOM
	See **T4S4 API Reference** for more details.
Job Type	`T4S_EXPORT_PO`

Import Job (Production Order)	
Job Type	`T4S_IMPORT_PO`
Mapping template	*t4s_prodorder_import_template.sd*

11.2 Export

A comprehensive set of transaction types is supported in the manufacturing area:

- ROUTING: Create, Change, Read, Change header

- INSPECTION PLAN: Create

- MBOM: PLMXML based BOM transactions

- PRODUCTION ORDER: Read Order

Transferring Standard Routing from Teamcenter PLMXML to SAP

In SAP a Routing is a description of work steps uniquely identified by a Routing group and Routing group-counter. It usually has a relation to a manufacturing BOM header material not limited to only one relation. It also contains consumed Materials and Production Resource Tools that are used to manufacture the allocated Material within the process described in the list of operations.

In Teamcenter the corresponding structures and relations are typically created and maintained in the Manufacturing Process Planner application and stored in Collaboration Contexts.

The target object for a Routing transaction is a MECollaborationContext or a subtype of it. For using the standard mapping template you have to include a MEProductContext, a MEProcessContext and a MEPlantContext with ConfigurationContext for each structure including the proper configuration rule. The process should consist of a header node of type MEProcess and a list of operation of type MEOP.

The root object of the MEProductContext reflects the BOM Header Material to which the Routing will be allocated. The MEPlantContext could contain plant and work center information for the Routing and for each operation.

The Routing transfer is based on Teamcenter PLMXML using a so called Application-Interface Object (AI-Object). T4S4 provides an example AI-Object Type with the BMIDE template *t4sdemo_template.zip*, which should only be used for demonstration purposes in a test environment. In a production environment, you should create your own AI-Object Type and Form Type in your database, based on your transfer scenarios and BMIDE project model.
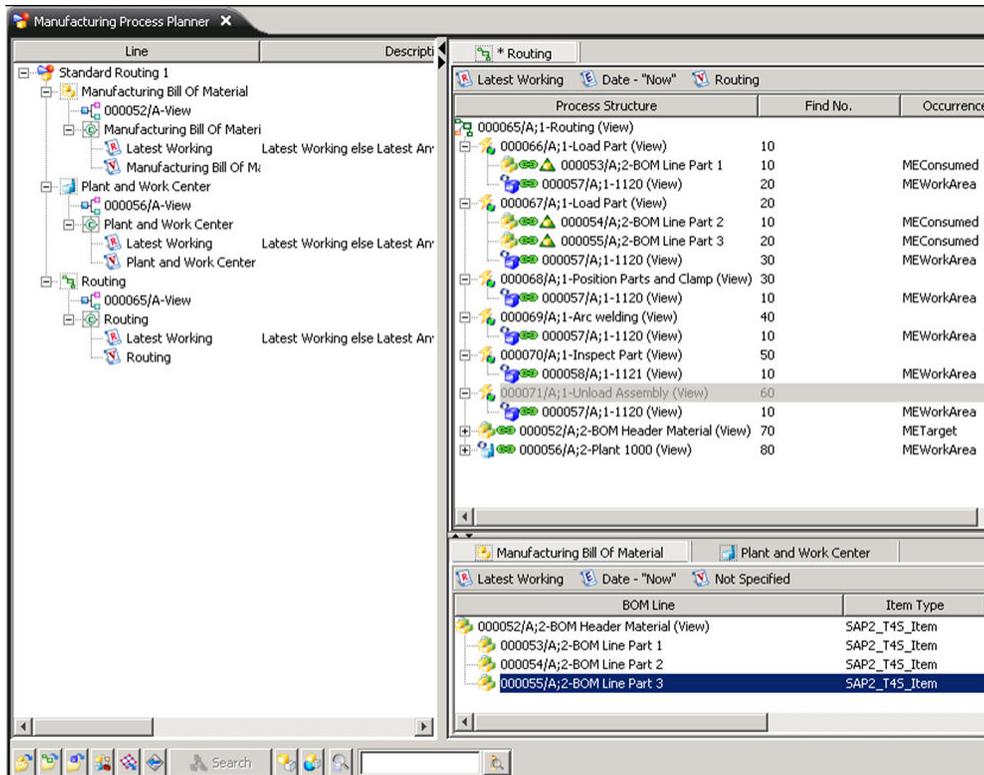
Pick the MECollaborationContext as target and run the template workflow T4S_Routing. A Routing is created in SAP, reflecting the data model in Teamcenter according to your mapping.
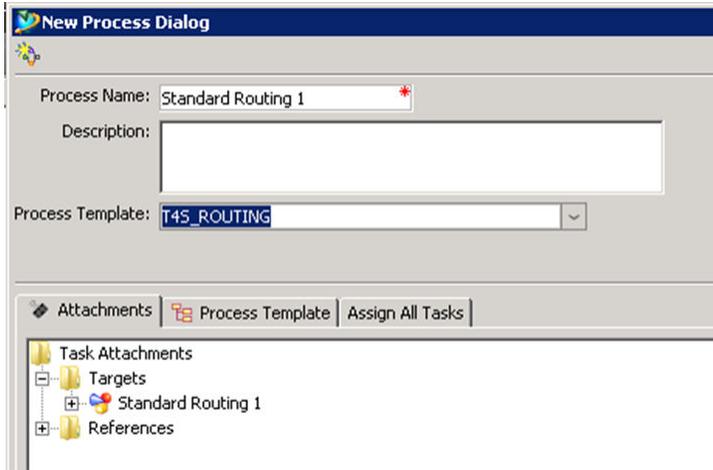
After the transaction returned successfully you will find a new form attached to the Collaboration Context with the group and counter information of the Routing.
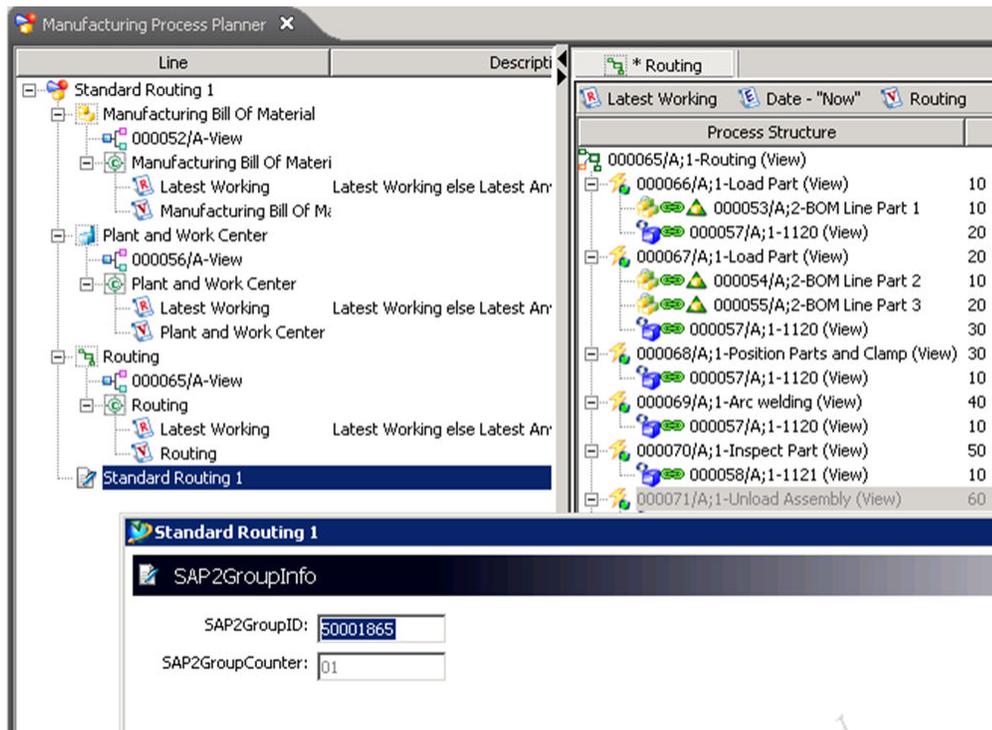
Result of the transaction in SAP:

> Caution:
>
> No specific preferences are needed for Routing.
>
> T4S4 cannot handle an ECM alternative date with Routing transfer.
>
> There is no T4S4 portal mode available. All Routing transactions are Teamcenter workflow based. A workflow template T4S_ROUTING is shipped with the BMIDE template.
>
> Routing can be pretty complex, and transfers may fail if the mapping parameters are not set to match each of the specifics of the Teamcenter and SAP side customizations and/or configuration settings! Critical points are the plant, work center, control key and setup type key. And you should always use a small and simple Routing object initially while testing the mapping configuration so that it is easier to detect potential problems.
>
> If a Routing transfer fails due to data issues and you did correct data in Teamcenter you should create a new workflow to generate a new AI-Object.

Specific Configuration in TC

The preference `TC_customization_libraries` needs to be adapted in order to load the library *libt4s_manufacturing.dll*. The preference value is `libt4s_manufacturing`.

There must be two transfer modes and an AI-Object Type available in the database. The T4S4 BMIDE templates provide an AI-Object Type "SAP2AI4Routing" plus transfer modes "SAP2TransferMode_Export_SAP" and "SAP2TransferMode_Import_SAP" as an example. You have to create your own custom AI-Object Types and transfer modes for a customer scenario. Such a custom AI-Object Type can then be used in a Routing transfer in the handler `T4S-create-AIObject4CC` by adding the workflow argument `–AI_TYPE=<custom AI type>`.

If the default Teamcenter AI-Object Type should be used, but not the default transfer modes. This can also be configured by using the handler arguments `–use_export_transfer_mode` and `–use_import_transfer_mode`.

Specific Configuration in SAP

In order to run transactions you have to deploy the SAP transport package TESISPLM_T4S_ROUTING to your system. Please see chapter *SAP Transport Packages for T4S4 Functionality* in the **Teamcenter Gateway for SAP S/4HANA - S/4HANA Preparation Guide**. Prior to that step you are limited to the Routing create transactions.

T4S4 Configuration

T4S4 is shipped with a Routing mapping template which is able to transfer SAP standard Routing type "N" with Operation, Material-Task-Allocation and Component-Allocation. The general transaction capability is limited by the SAP BAPI. The mapping template is not fully configured to support all possible objects but can be enhanced.

T4S4 has introduced a new XML reader engine and mapping syntax that allows much faster transactions and based on a new mapping language that allows easier customization of the mapping. It is essential to have some PLMXML and general Routing structure knowledge to successfully customize the Routing mapping.

T4S4 ships with two different mapping files and PLMXML reader engines. The new one as mentioned before is highly enhanced but requires a full manual migration of any existing mapping code of previous releases. In order to support existing Routing mapping code we also support the previous PLMXML engine. You can specify the usage of the parser type in your mapping. Default is using the new parser.

To switch on support of previous Routing syntax you should add to your t4s_mapping_config.sd the following line:

```
set ::T4S::PLMXML::ROUTING::CtrlParam(useObjectServerBasedReader) "FALSE"
```

For detailed information about the new object based mapping syntax please refer to the document **Teamcenter Gateway - PLMXML Reader Guide**.

Based on a standard SAP BAPI there is no reverse transaction of Routing data beside the group and counter that will be stored on a form related to the MECollaborationContext. After the transaction you will find a new form attached to the MECollaborationContext that contains the group and counter of the created Routing. In SAP you will find a Routing with the group and counter stored in the form.

Transferring Inspection Plan from Teamcenter PLMXML to SAP

The transfer capabilities of T4S4 include Inspection Plans from Teamcenter to SAP which is a specific type of Routing for quality assurance purposes. Depending on your specific use case the existing capabilities or customer specific enhancements can be provided upon your request.

Mapping files can be provided on a customer project basis.

Transferring Manufacturing BOM from Teamcenter PLMXML to SAP

T4S4 provides the possibility to transfer Bill Of Material based on a PLMXML export file from Teamcenter. Such a solution may be interesting if the goal of your transaction would be to derive several SAP BOM based on a configurable 150% BOM in Teamcenter where you would like to extract the multiple configured BOM in the interface instead of creating them in Teamcenter. Other use cases may be also supported upon your request.

Mapping files can be provided on a customer project basis.

Transferring Production Order data from SAP to Teamcenter

The transfer of Production Order data from SAP to Teamcenter can be used to close the feedback loop in your manufacturing scenario. Depending on the customers use case T4S4 can be individually customized to support the transfer of production order header data and may be further enhanced upon request.

12. Schedule Transfer

12.1 Introduction

Please read the **Teamcenter Gateway - Generic Configuration Guide** first to understand the basics for this chapter.

Nomenclature

Please also see the **Glossary**.

Teamcenter	SAP
Schedule	SAP Network or WBS Element

T4S4 Specifics Compared to the Generic Mapping

SAP Transaction Codes	
SCMA	Schedule Manager

Export	
TargetTypeName	Schedule
Mapping template	*t4s_schedule_mapping_template.sd*
Namespace Mapping	`::T4S::SCHEDULE::CUSTOM::MAPPING`
Preferences start with	T4S_Schedule
Handler	T4S-transfer-Schedule See **T4S4 API Reference** for more details.
Job Type	`T4S_EXPORT_SCHEDULE`

Import Job	

Job Type	`T4S_IMPORT_PROJECT`
	
	 `T4S_IMPORT_PROJECT_ITEM`
	
	 `T4S_IMPORT_PROJECT_SCHEDULE`
	
	 `T4S_IMPORT_PROJECT_SCHEDULE_WBS`
	
	 `T4S_IMPORT_PROJECT_SCHEDULE_NETWORK`
	
	 `T4S_IMPORT_PROJECT_ITEM_STRUCTURE`
	
	 `T4S_IMPORT_PROJECT_SCHEDULE_STRUCTURE`

Mapping template	*t4s_project_item_import_template.sd*

Import Job
t4s_schedule_mapping_template.sd
t4s_project_item_structure_import_template.sd
t4s_project_schedule_structure_import_template.sd

12.2   Export of TC Schedule Information to the SAP Project System

A Teamcenter Schedule object (Schedule or Schedule Task) can be transferred to SAP by T4S4 via workflow or the job processing infrastructure. Depending on the configuration and the mapping it is possible to create or update a SAP Project Definition, Project Network or Work Breakdown Structure Element (WBS Element).

> Caution:
>
> Due to Teamcenter data model changes between Teamcenter 10 and 11 the object transfer implementation will differ in the preference or mapping configuration. So that it might be necessary to migrate the T4S4 configuration after an upgrade to Teamcenter 11 manually.

Requirements for Teamcenter Object related Transfers

All object related transfers are based on the same mapping template named *t4s_schedule_mapping_template.sd*. In the mapping template it is possible to decide which target object should be updated in SAP. This can be controlled by defining a configuration variable (like in the following examples) or a workflow argument or by the Teamcenter object itself.

The T4S_SCHEDULE – Workflow template needs to be imported.

Preferences

- `T4S_ScheduleTypeList`
  List of allowed Teamcenter object types (e.g. Schedule, Schedule Task)

- `T4S_ScheduleMapping4<ObjectType>`
  List of Teamcenter data types to read attributes from

- `T4S_ScheduleMapping2<ObjectType>`
  List of Teamcenter data types to write back during reverse mapping

- `T4S_ScheduleFieldMapping2<ObjectType>` (optional)
  List of Teamcenter attributes to write back during reverse mapping.

Transfer Teamcenter Schedule Data to a SAP Network

The `T4S-transfer-Schedule` action handler (see T4S_SCHEDULE sample workflow) transfers Teamcenter Schedule object information to a SAP Network object if the mapping is defined accordingly.

This means based on the *t4s_schedule_mapping_template.sd* the variable `::T4S_Defaults(TC2SAPType)` is set to `"NETWORK"`. Beside that the Teamcenter Schedule object requires a reference to the SAP Project Definition Id (this could be an extra attribute or the Schedule Object Id) and an attribute that represents the SAP Network Id (used during the reverse mapping).

If the SAP Project Definition doesn't exist yet it will be created otherwise the existing Project Definition will be updated. If the SAP Network doesn't exist yet it will be created otherwise the existing Network will be updated.

Use Defaults in the mapping template for a NETWORK based transfer:

```
# Schedule to NETWORK
set ::T4S_Defaults(TC2SAPType)      "NETWORK"
set ::T4S_Defaults(ProjectProfile)  "U"
set ::T4S_Defaults(NetworkProfile)  "Z000002"
```

Example screenshot for a transfer to a SAP Network:

Transfer Teamcenter Schedule Data to a SAP WBS Element

The `T4S-transfer-Schedule` action handler (see T4S_SCHEDULE sample workflow) transfers Teamcenter Schedule object information to a SAP Work Breakdown Structure Element if the mapping is defined accordingly.

This means based on the *t4s_schedule_mapping_template.sd* the variable `::T4S_Defaults(TC2SAPType)` is set to `"WBS"`. Beside that the Teamcenter Schedule object requires a reference to the SAP Project Definition Id (this could be an extra attribute or the Schedule Object Id) and an attribute that represents the SAP WBS Element Id (used during the reverse mapping).

If the SAP Project Definition doesn't exist yet it will be created otherwise the existing Project Definition will be updated.

If the SAP WBS Element doesn't exist yet it will be created otherwise the existing WBS Element will be updated.

Use Defaults in the mapping template for a WBS Element based transfer:

```
set ::T4S_Defaults(TC2SAPType)      WBS
set ::T4S_Defaults(ProjectProfile)  U
set ::T4S_Defaults(NetworkProfile)  Z000002
```

Example Screenshot for a transfer to a SAP WBS Element:
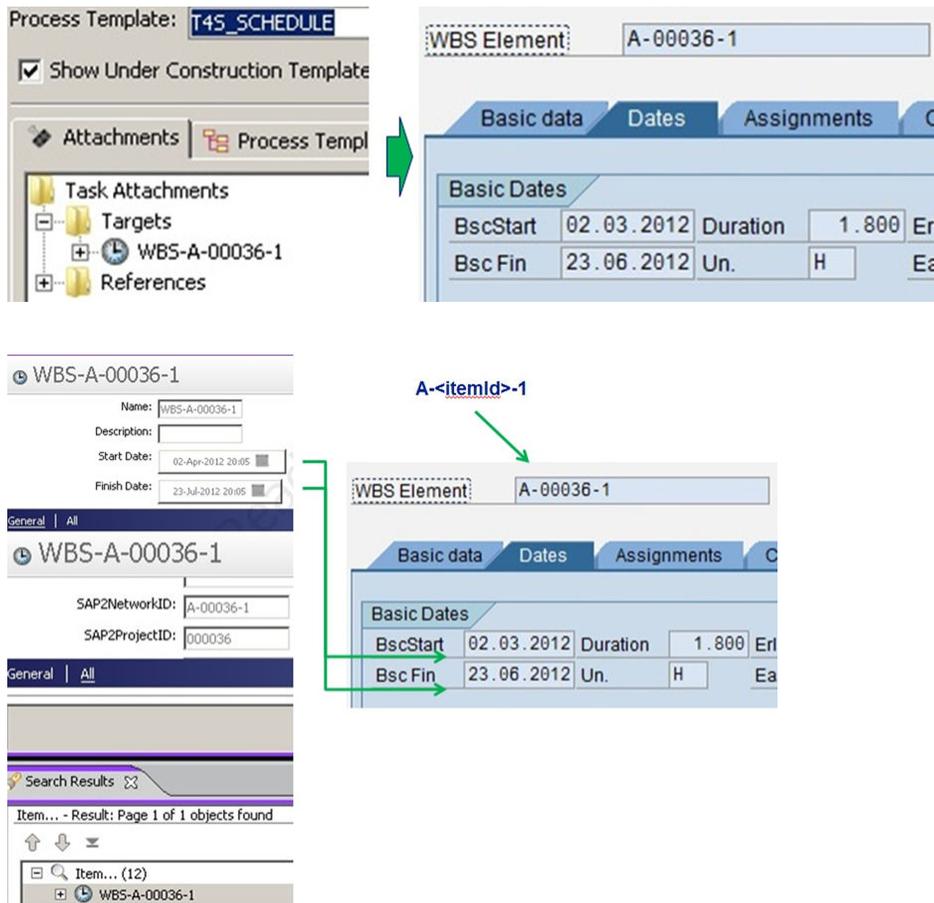
12.3 Import of SAP Project System Information to the TC Schedule

The following use-cases are provided:

- Import SAP Project System Data to Teamcenter Schedule and Item

- Import SAP Project System Data to Teamcenter Schedule

- Import SAP Project System Data to Teamcenter Project Item

- Import SAP Project System Data to Teamcenter Project Item Structure

- Import SAP Project System Data to Teamcenter Schedule Structure

Limitations and Requirements

T4S4 uses the Teamcenter SOA API during the import (create/update) of Teamcenter Schedule objects (like Schedule or Schedule Task). This means it requires a configured Teamcenter Pool Manager and its Web Application Server (e.g. JBOSS) environment.

Beside that, the following additional requirements needs to be fulfilled:

- The Teamcenter Schedule Manager environment needs to be configured (e.g. Timezone, ...).

- A Teamcenter 10 environment requires also an installed Teamcenter Dispatcher with the SchMgrTrans module and Dispatcher Client.

- The Teamcenter SOA policy file *ProjectManagement.xlm* shall be copied from *<GS_Root>\var\template\t4sdemo* into *<TC_Data>\soa\policies* and adapted to the customer needs.

The correct Teamcenter SOA related connection parameters should be defined in the mapping:

```
# URL for accessing Teamcenter SOA services
# ("http(s)://<host>:<port>/<application>")
# use port 7001 for WebLogic or 8080 for WebSphere
set ::T4X::SOA::CtrlParam(TcSoaUrl) "http://localhost:7001/tc"
```

Import SAP Project System Data to Teamcenter Schedule and Item

This function will import a SAP Project Definition or a Work Breakdown Structure Element to the corresponding Teamcenter Item and Schedule based representation. A link between the Schedule and the Item in Teamcenter is created during the import of the both target objects.

The import job is divided into the following three steps:

- Create or Update a Teamcenter Schedule based on the corresponding SAP PS data.

- Create or Update a Teamcenter Item based on the corresponding SAP PS data.

- Create link between the two object created or updated in the previous steps.

The import function itself is implemented by an extra import batch job template (named T4S_IMPORT_PROJECT, see mapping template *t4s_project_import_template.sd*) that is following the same functionality as the normal Meta import batch job (T4S_IMPORT_META).

Import SAP Project System Data to Teamcenter Schedule

This function will import a SAP Project Definition or a Work Breakdown Structure Element to the corresponding Teamcenter Schedule based representation.

This use case it typically used if there is a need to detail the SAP Project data in Teamcenter so that the result of this detailed planning can be used in the Engineering or for transferring the Teamcenter planning results back to the SAP PS.

Depending on the use case it is possible to create a new Schedule from scratch or based on a Schedule template, if needed it is also possible to create a baseline for an existing Schedule.

The import function itself is implemented by an extra import batch job template (named T4S_IMPORT_PROJECT_SCHEDUE, see mapping template *t4s_project_schedule_import_template.sd*) that is following the same functionality as the normal Material Master import job (T4S_IMPORT_MM).

Import SAP Project System Data to Teamcenter Project Item

This function will import a SAP Project Definition or a Work Breakdown Structure Element to the corresponding Teamcenter (Project) Item / Item Revision based representation.

This use case is used if no schedule based planning in Teamcenter is needed, but there is a need to have a container Item that will hold all project related objects (like Datasets, Design Items, ...), via a BOM view or relations.

The import function itself is implemented by an extra import batch job template (named T4S_IMPORT_PROJECT_ITEM, see mapping template *t4s_project_item_import_template.sd*) that is following the same functionality as the normal Material Master import job (T4S_IMPORT_MM).

Import SAP Project System Data to Teamcenter Project Item Structure

This function will import the Work Breakdown Structure (WBS) of a SAP project to the corresponding Teamcenter Item / ItemRevision BOMView. The resulting object structure in Teamcenter is based on a BOM view. During the import it is possible to create or use all needed Teamcenter objects.

This use case is used if no schedule based planning in Teamcenter is needed, but there is a need to have container Items for all WBS elements in the structure so that it is possible to attach all project/WBS related Teamcenter objects (like Datasets, DesignItems, ...).

The import function itself is implemented by an extra import batch job template (named T4S_IMPORT_PROJECT_ITEM_STRUCTURE, see mapping template *t4s_project_item_structure_import_template.sd*) that is following the same functionality as the normal Bill Of Material import job (T4S_IMPORT_BOM).

Import SAP Project System Data to Teamcenter Schedule Structure

This function will import the Work Breakdown Structure (WBS) of a SAP project to the corresponding Teamcenter Schedule. The resulting Teamcenter Schedule structure can be used to start point of a detailed project planning in the context of Teamcenter.

This use case it typically used if there is a need to detail the SAP Project data in Teamcenter. This can be used for planning the Engineering Schedule in Teamcenter only or as base transferring the Teamcenter planning results back to the SAP PS.

The import function itself is implemented by an extra import batch job template (named T4S_IMPORT_PROJECT_SCHEDULE_STRUCTURE, see mapping template *t4s_project_schedule_structure_import_template.sd*) that is following the same functionality as the normal Material Master import job (T4S_IMPORT_MM).

13. IPPE Transfer

13.1 Introduction

Please read the **Teamcenter Gateway - Generic Configuration Guide** first to understand the basics for this chapter.

> Caution:
>
> The overall performance for the synchronization of the iPPE structure with the Teamcenter structure is based on the size of the structures and on the number of structure updates in both systems that have to be synchronized. This kind of performance problem has to be solved by defining the right business workflows during customer project design phase.

Nomenclature

Please also see the **Glossary**.

Teamcenter	SAP
Variant BOM	iPPE

T4S4 Specifics Compared to the Generic Mapping

SAP Transaction Codes	
PDN	Product Designer Workbench

Export	
TargetTypeName	iPPE_Node iPPE_Structure
Mapping template	*t4s_ippe_node_mapping_template.sd*
Namespace Mapping	`::T4S::iPPE::NODE::CUSTOM::MAPPING`
Preferences start with	T4S_iPPE_Node T4S_iPPE_Structure
Handler	T4S-transfer-iPPE-Node4BomLine See **T4S4 API Reference** for more details.
Job Type	`T4S_EXPORT_IPPE_NODES`

Import Job	
Job Type	`T4S_IMPORT_IPPE_NODES`

Import Job	
	`T4S_IMPORT_VARIANT_OPTIONS`
Mapping template	*t4s_ippe_structure_import_template.sd*

Relationship between TC and SAP Objects

In SAP there are by default no revisions for an iPPE. Therefore the relation between TC Item Revision and SAP iPPE is n:1.

## Synchronization between SAP iPPE and Teamcenter

The main use case for that project is:

• The product design starts in SAP

• The SAP iPPE high-level structure includes a set of functional structure nodes for an easier navigation through the complex product structure

• The product design in the iPPE will include the configurability (defined in the attached variant class)

• The product design in the iPPE will include the component variant (including the dependencies, used as variant conditions)

• Based on this iPPE definition a corresponding structure is created in Teamcenter including option definition; see "IPPE Picture 3" below

• In Teamcenter the corresponding items (materials) that represent a specific component variant are attached to the Teamcenter structure

This assignment will be transferred back to the iPPE structure to complete the product design with the corresponding engineering data:
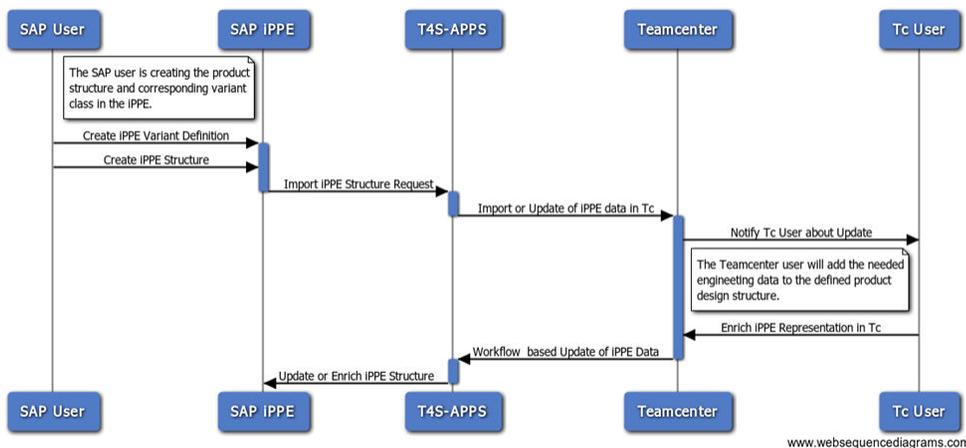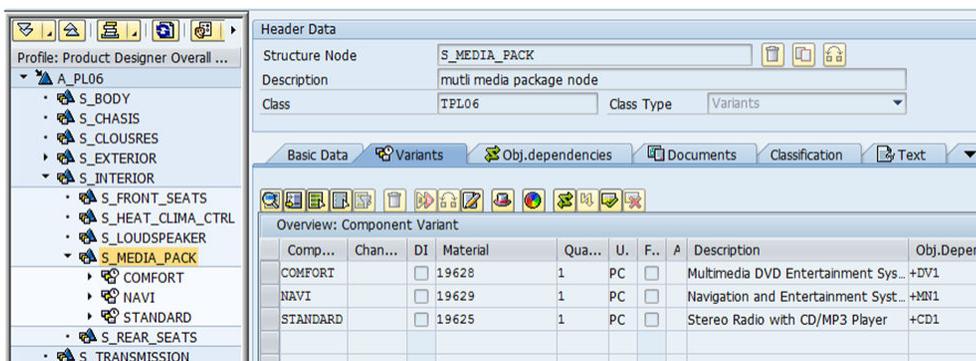
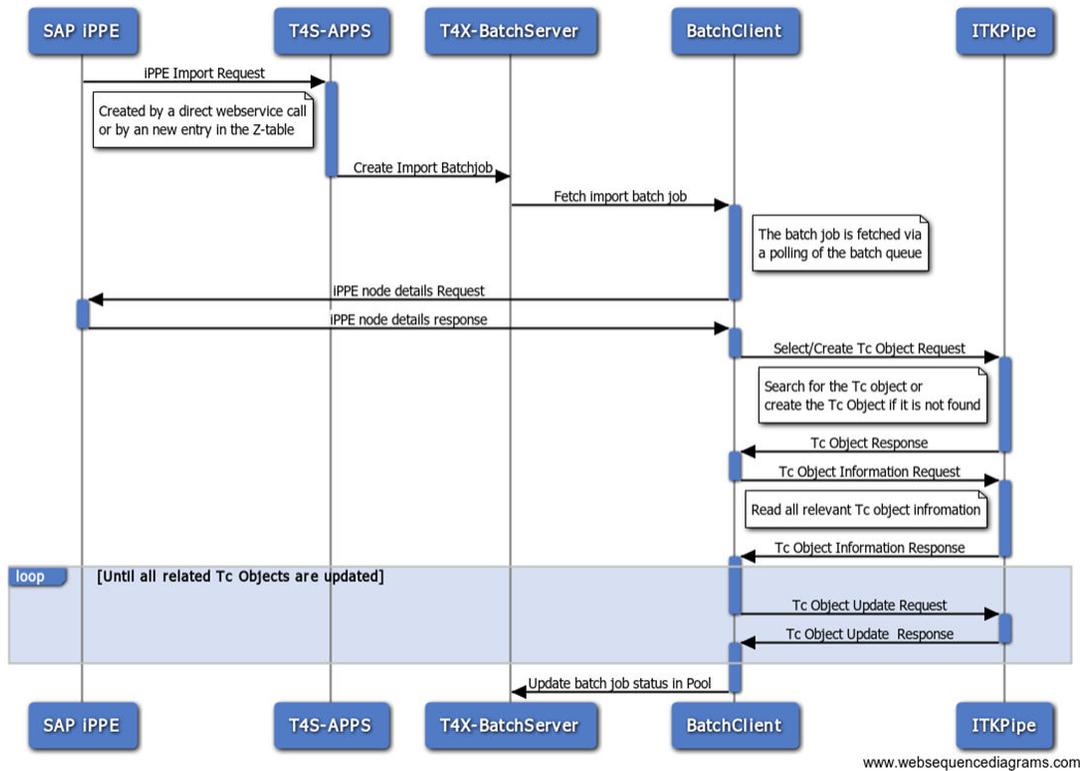Sample SAP iPPE structure:
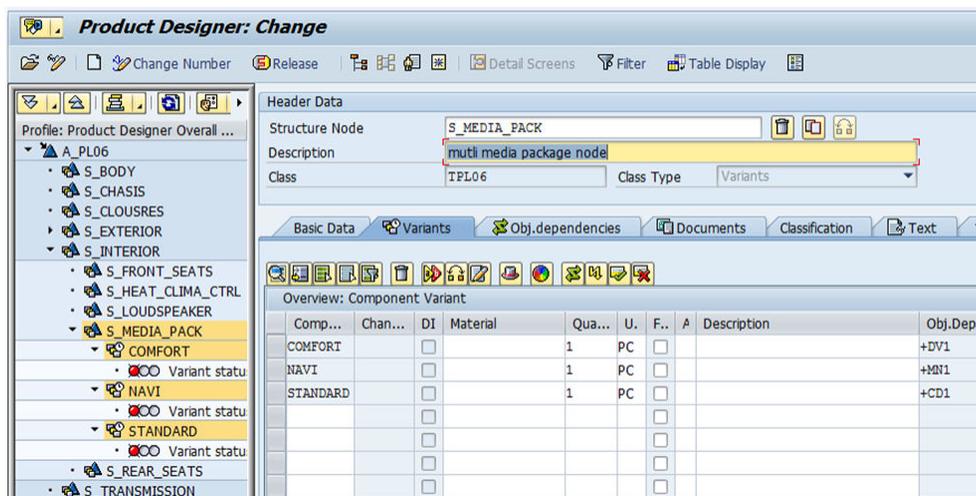
Imported to Teamcenter:
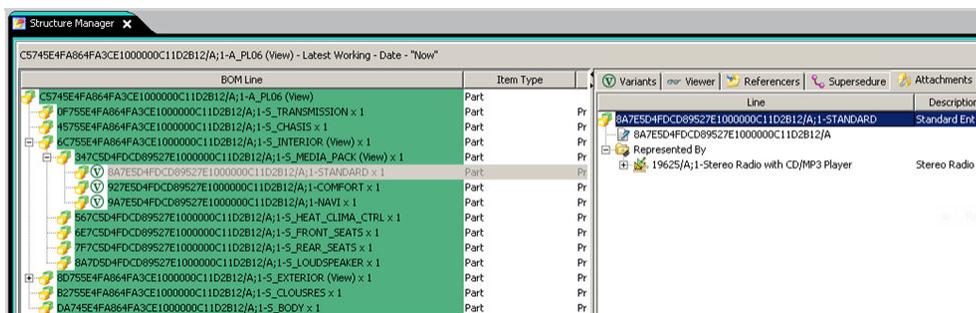
Updated SAP iPPE structure with information from TC:

13.2   Import

SAP iPPE import process flow:

SAP iPPE   T4S-APPS   T4X-BatchServer   BatchClient   ITKPipe

iPPE Import Request

Created by a direct webservice call or by an new entry in the Z-table

Create Import Batchjob

Fetch import batch job

The batch job is fetched via a polling of the batch queue

iPPE node details Request

iPPE node details response

Select/Create Tc Object Request

Search for the Tc object or create the Tc Object if it is not found

Tc Object Response

Tc Object Information Request

Read all relevant Tc object infromation

Tc Object Information Response

loop   [Until all related Tc Objects are updated]

Tc Object Update Request

Tc Object Update Response

Update batch job status in Pool

SAP iPPE   T4S-APPS   T4X-BatchServer   BatchClient   ITKPipe

www.websequencediagrams.com

Import iPPE Structure Node

This function will import the iPPE structure node information only, that means it will create the corresponding object in Teamcenter based on the given SAP iPPE node information.

The following iPPE objects are used as input for the creation or update of the corresponding Teamcenter object:

- Access Nodes

- Structure Nodes

- PVS Nodes

- Color Nodes

The resulting objects in Teamcenter are based on a defined Item (Item Revision) type, for the support of an update of the data it is necessary to store the GUID (32 characters) of the iPPE node into the created Teamcenter object.

The import function itself is implemented by an extra import job template that is based on a normal item import like the Material Master import job.

Import Variant Class Information

This function will import the SAP variant definition of the iPPE structure into the corresponding Teamcenter variant option definition based on Teamcenter classic variants or modular variants. The import job will translate the SAP variant characteristics in the corresponding Teamcenter option definition of the selected variant module.

For the import of the variant option definitions and variant conditions for the Teamcenter modular variants the following set of ITK functions is added to the TCL scripting environment:

- `::ITK::BOM_module_list_options`

- `::ITK::BOM_option_describe`

- `::ITK::BOM_option_ask_logical_default`

- `::ITK::BOM_option_ask_allowed_ints`

- `::ITK::BOM_option_ask_allowed_reals`

- `::ITK::BOM_option_ask_allowed_strings`

- `::ITK::BOM_option_revise`

- `::ITK::BOM_create_option`

- `::ITK::BOM_sos_ask_entries`

- `::ITK::BOM_sos_ask_entry_int`

- `::ITK::BOM_sos_ask_entry_double`

- `::ITK::BOM_sos_ask_entry_logical`

- `::ITK::BOM_sos_ask_entry_display`

- `::ITK::BOM_sos_set_entry_int`

- `::ITK::BOM_sos_set_entry_double`

- `::ITK::BOM_sos_set_entry_logical`

- `::ITK::BOM_sos_set_entry_string`

Import iPPE Structure Node Structure

This function will import the iPPE structure node information and all related children. That means it will create the corresponding structure in Teamcenter based on the given SAP iPPE start node information.

The following iPPE objects and relation between the objects are used as input for the creation or update of the corresponding Teamcenter object:

- Access Nodes

- Structure Nodes

- PVS Nodes

- Color Nodes

The resulting object structure in Teamcenter is based on BOM views or GRM based relations. The import to Teamcenter will also include the variant conditions that are defined for the specific iPPE node in SAP (defined in the object dependency for the iPPE node).

For the import of the variant condition the corresponding Teamcenter structure has to be a configurable structure. The import function supports classic and modular variants.

The import function itself is implemented by an extra import job template that is based on a Material BOM import like the Bill Of Material import job.

Supporting Change Required for using SAP iPPE

During the implementation at the customer it is required that a storage attribute for the iPPE internal GUID is defined in the BMIDE for the Teamcenter Object that represent the corresponding iPPE node.

13.3 Export

SAP iPPE export process flow:

www.websequencediagrams.com
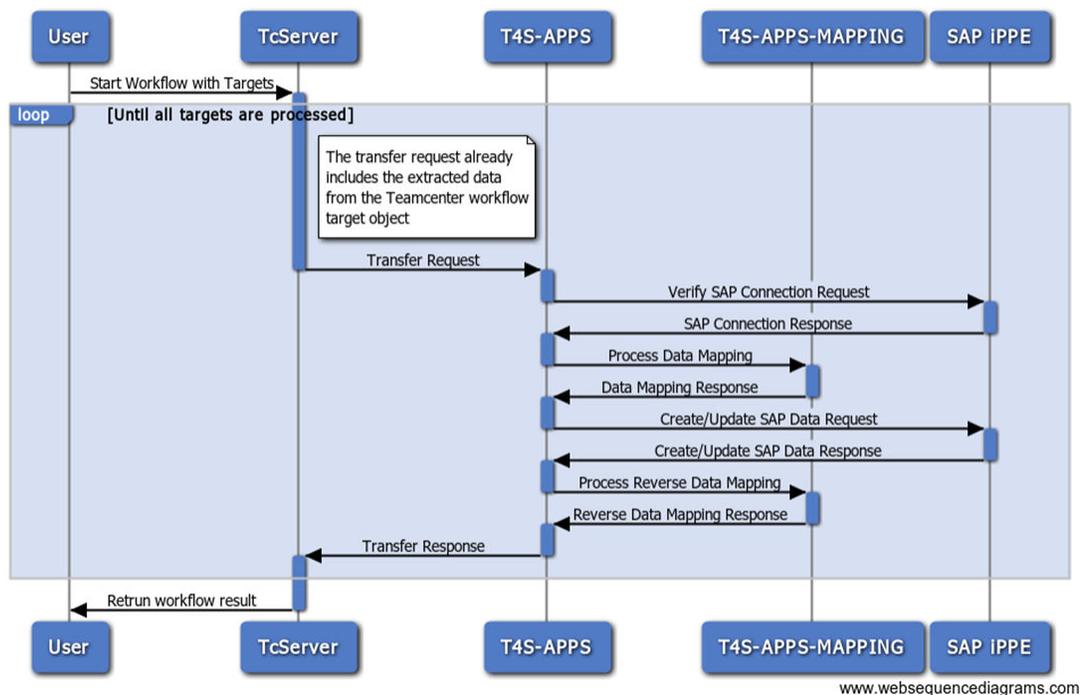
Transfer Teamcenter Data to iPPE Structure Node Structure

This function will extract the Teamcenter BOM object data of a workflow target item (or BOM line item), map the information into an internal update structure for the a specified iPPE target structure node and will update or create the specified iPPE node in the connected SAP system.

The following iPPE objects are used as targets of the workflow transfer for the creation or update:

• Structure Nodes

• PVS Nodes

• Color Nodes

Based on the transfer result the source Teamcenter object is updated with the transfer status and if a new iPPE node is created in SAP the iPPE node GUID (32 characters) is stored back to the Teamcenter object as well.

In addition to the workflow based transfer a job based transfer will also be part of the implementation project. This will reuse the mapping files and the transfer business logic of the workflow transfer.

Supporting Change Required for using SAP iPPE

It is required that a storage attribute for the iPPE internal GUID is defined in the BMIDE for the Teamcenter Object that represents the corresponding iPPE node.

Preferences

For the iPPE information transfer by the handler `T4S-transfer-iPPE-Node4BomLine` the following preferences are defined for the data mapping configuration:

- `T4S_iPPE_NodeTypeList`

- `T4S_iPPE_NodeMapping4<ObjectType>`

- `T4S_iPPE_NodeMapping2<ObjectType>`

- `T4S_iPPE_NodeFieldMapping2<RelationInformation>`

- `T4S_iPPE_StructureTypeList`

- `T4S_iPPE_StructureMapping4<ViewType>`

- `T4S_iPPE_StructureHeaderMapping4<ObjectType>`

- `T4S_iPPE_StructureLineMapping4<ObjectType>`

- `T4S_iPPE_StructureMapping2<ViewType>`

- `T4S_iPPE_StructureFieldMapping2<RelationInformation>`

14. Classification Synchronization

To transfer Classification data of Business Objects between TC and SAP, the Classification must be aligned between both systems. T4S4 has a functionality to synchronize the Classes, Characteristics and allowed Characteristic Values between Teamcenter and SAP.

> Caution:
>
> Although the functionality comes OOTB, it needs to be adapted, as each customer has different requirements and the TC and SAP Classification is not the same. Please get in contact with Siemens PLM Software for integration support and expertise.

Differences between TC and SAP Classification

Teamcenter and SAP name some classification details differently:

Teamcenter	SAP
Class	Class
Classification Attribute	Characteristic
Key LOV	Value Set

- Teamcenter classifies Items and/or Item Revisions, SAP uses MMs, DIRs, etc.

- Same as Teamcenter, SAP does not use classification for every object type

Initiate Class Sync

T4S4 supports the functionality "classification synchronization" which is controlled by scripts:

- Class Synch - Clean

- Class Synch - SAP to TC

- Class Synch - TC to SAP

Use the search string "synch" in the Scripts section of the T4S4 GS Admin GUI, then it shows those scripts. In the used script the "Help" tab can help you with more details how to use it

TC to SAP

The script "Class Synch – TC to SAP" checks the Teamcenter classes, their classification attributes and their allowed value types (e.g. strings or only numbers). Then it checks if it is defined correspondingly in a SAP class. If not, it tries adding that missing classification definition points into the SAP classification definition.

Teamcenter can store in principle the same classification object with different names and/or descriptions in different languages. In order to send them all to SAP, set the parameter "SAP Language Mode" to the value "multiple". By default, Teamcenter does not allow more than one language for a class, but it can be configured internally to use that similar as in SAP.

- Example with the value "single":

  - "TCClass.name" gets synchronized to "SAPClass.description"

  - "TCClass.aliasNames" get synchronized to "SAPClass.keywords"

  The SAP language used for the description and for the keywords is specified by the value of parameter "SAP language"

- Example with the value "multiple":

  - "TCClass.name" (and their locales) get synchronized to "SAPClass.description" (corresponding to their locales)

- "TCClass.aliasNames" get synchronized to "SAPClass.keywords" (for all locales for which values exist in TC)

The SAP language used for the description is the language used in TC for the localized name data. The parameter "SAP language" is only available if the parameter "SAP language mode" has the value "single"; then "SAP language" specifies the SAP language to use for TC data that gets synchronized to SAP multi-language data.

SAP to TC

The script "Class Synch – SAP to TC" does the same in the other way round: first it checks the SAP classes of the stated SAP class type and their classification characteristics with their allowed value types. Then it checks if it is defined correspondingly in the stated Teamcenter class ("TC Target Class ID"). If not, it tries adding that missing classification definition points into the Teamcenter classification definition. That means that the value of that parameter is not the ID of the TC class to be installed, but an existing class to which the new class should be added.

The setting Mapping Mode = dry (not shown here, because the default value is "real") means that nothing will be changed but checked only. The shown value real will actually try modifying the classification definition in Teamcenter and/or SAP.

Clean

The script "Class Synch – Clean" tries removing an internal classification lock that has to be set in order to handle the classification definition correctly. By default that is not needed, but if an error occurs or more than one process tries doing anything there at the same time, then there may be a lock problem and that can be fixed by that script.

That screenshot shows how T4S4 handles that internally:
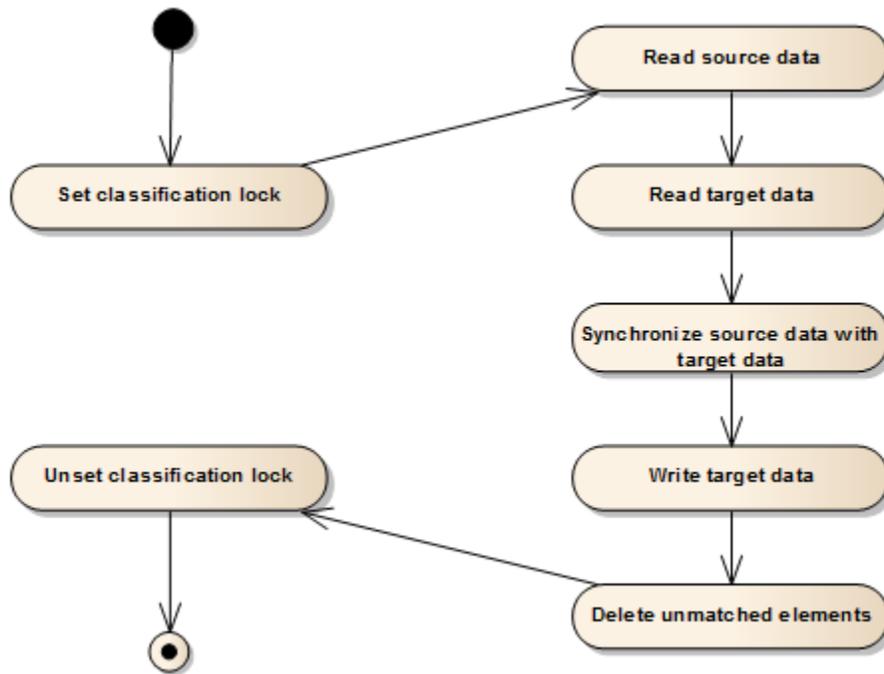
As this is an internal definition handling, you have to execute the script as a Teamcenter user who has not only default user rights but admin user rights. Teamcenter does not even allow the classification check functionality for a non-dba user! So maybe change the Teamcenter user by the script Tc database connection test, e.g. to the default Teamcenter admin user infodba.

> Caution:
>
> Depending on what the script should do in SAP, maybe you have to change the current SAP user. This can be done using the script SAP connection test. Such a user change does not work if an Auto Login is defined for Teamcenter and/or SAP. So it should be removed from the mapping and then the user change done by the corresponding script.
>
> Those classification synchronization scripts can also delete classification definition in Teamcenter and SAP. As such a deletion is not wanted in most cases, just keep then the default values "false" in the parameters with the word "delete"
>
> As the classification generally may work very differently in customer's Teamcenter and SAP depending on the current configuration, maybe the functionality has to be configured in order to use it.

15. Generic Transfer

T4S4 allows you to create your own highly customizable generic transactions to suit your special needs. Please see chapter *T4x Export Concept* and *T4x Import Concept* in the **Teamcenter Gateway - Generic Configuration Guide** for detailed information.

16. Data Transfer

16.1 Configure the T4S Gateway Menu Options

Please also see **Menu-based Triggers** in the **Teamcenter Gateway - Generic Configuration Guide** on how the T4S Gateway Menu works in general.

Modify the preference "`T4S_Gateway_Menu_Explorer_Entries`" in order to set the available functions in the additional menu "T4S Gateway" in "My Teamcenter". This does not affect the workflow functionality but only specifies the functions the user may select from the menu. The same menu will be shown in the TC modules Structure Manager, Multi-Structure Manager and the Manufacturing Process Planer.

Each value for this preference consists of two parts:

- The object type it is used for

- The function for this object type

The possible TargetTypeNames are:

- `MM` – Material Master

- `BOM` – Bill Of Material

- `DIR` – Document Info Record

- `ECM` – Engineering Change Master

- `EQUI` – Equipment Master

- `FCL` – Functional Location

The possible functions are (in brackets: the object type(s) this function works for):

- `CREATE` (all except PIR)
  Creates an object of the desired type in SAP with the SAP GUI (i.e. a SAP GUI opens where you can enter data manually). In this case, you can also use SAP's function to create an object from a template (MM, ECM and Functional Location: "Copy from…"; DIR: "Template"; Equipment: "Reference"; not possible for BOM).

- `CHANGE` (all except PIR)
  Modifies the SAP object with the SAP GUI.

- `CREATE_HID` (all except PIR)
  Creates an object of the desired type in SAP without the SAP GUI (i.e. no SAP GUI opens).

- `CHANGE_HID` (all except PIR)
  Modifies the SAP object without the SAP GUI.

- `DISPLAY` (all except PIR)
  Displays the SAP object in the SAP GUI.

- `HTTP_DISPLAY` (all except PIR)
  Displays the SAP object in the SAP GUI for HTML. The base URL of the SAP system needs to be defined in the mapping.

- `RESERVE` (MM only)
  Gets the next valid Material Master number from SAP's number generator ("internal numbering") and stores this for further usage. This does not prevent that anyone creates a SAP MM with this number if he states it explicitly. But it tells SAP not to give this number again if it is asked for a new one.

- `LINK` (all except BOM and PIR)
  Tells T4S4 to use a corresponding SAP object with a stated identification. Actually, it allows the user to search for a data object in SAP and stores its ID (and any other desired data) in the Teamcenter attribute stated in the mapping file. Same as in any other case with T4S4, SAP does not recognize a relation.

- `LINKVIACLASS` (MM only)
  Same as "Link" but looks for an MM with specifiable characteristic values in a SAP class instead of searching an MM ID. This is done by calling "Find Objects in Classes" in SAP. Please check SAP documentation on details.

- `PSB` (MM, BOM, DIR, ECM)
  Opens the "Product Structure Browser" in SAP.

- `HTTP_PSB` (MM, BOM, DIR, ECM, i.e. all the object types which work with "PSB")
  Opens the "Product Structure Browser" in SAP GUI for HTML. The base URL of the SAP system needs to be defined in the mapping.

Each function uses the Mapping and sets the appropriate data to the SAP object and to the Teamcenter object. If a transaction type is specified for an object type that does not allow that (e.g. PSB for EQUI) this will be shown in the menu nevertheless without showing an error to the Teamcenter user. If this invalid transaction is performed, an error message is shown.

If at least one entry for a specific type (e.g. Material Master) is given, the corresponding sub menu will appear; e.g. the entry `MM_DISPLAY` is sufficient to show the sub menu "Material Master" in the T4S4 Gateway Menu. If that preference does not contain any entry `MM_*`, then the sub menu "Material Master" is not shown at all.

Example:

```
T4S_Gateway_Menu_Explorer_Entries=
SAPCONNECT
MM_CREATE_HID
MM_CHANGE_HID
MM_DISPLAY
MM_RESERVE
MM_LINK
MM_PSB
BOM_DISPLAY
BOM_PSB
DIR_CREATE_HID
DIR_CHANGE_HID
DIR_DISPLAY
VENDOR_DISPLAY
```

In the T4S4 Gateway Menu, you will then find the corresponding entries, e.g. for DIR: "Create Direct", "Change Direct" and "Display". If you click such a menu point after selecting one or multiple valid objects, the T4S Transfer Window that shows the configured data for this object and transaction type and you can start the transaction by clicking "Perform" or "Perform all".

Auto Perform

If you do not want to wait and click, you can configure the software to start some transactions immediately using the preference `T4S_Gateway_AutoPerform`. Whenever a transaction type stated as a value of this preference is selected from the "T4S4 Gateway", the T4S Transfer Window is opened and the transaction will start immediately without waiting for the click on "Perform".

Example as shipped in the template file, i.e. all those Display transactions will be started automatically, any other will wait for the click on "Perform":

```
T4S_Gateway_AutoPerform=
MM_DISPLAY
BOM_DISPLAY
DIR_DISPLAY
ECM_DISPLAY
EQUI_DISPLAY
FCL_DISPLAY
```

Definition of Teamcenter Objects You Want to Support in the T4S4 GUI

This is controlled by the optional preference T4S_<TargetTypeName>DisplayableTypeList. It contains subset of the values of T4S_<TargetTypeName>TypeList and means that T4S4 only has to use those types in the GUI.

```
T4S_<TargetTypeName>DisplayableTypeList=
```

```
SAP2_T4S_Item Revision

T4S_MaterialMasterDisplayableTypeList=
SAP2_T4S_Item Revision
```

> **Caution:**
>
> Leaving the key "T4S_MaterialMasterDisplayableTypeList=" empty is the same as not stating it at all, i.e. then every data type that is configured with "T4S_MaterialMasterTypeList=" is used in the T4S Gateway Menu

Hiding the T4S Gateway Menu

The T4S Gateway menu can be switched off completely by setting the following preference (type logical; by default it does not exist):

```
T4S.UI.GatewayMenu.Hide=true
```

A similar preference exists for each T4x product (e.g. `T4O.UI.GatewayMenu.Hide`). If there is more than one T4x product installed in the same Teamcenter installation, it may be useful to switch all the T4x Gateway menus off and on with the preference `T4X.UI.GatewayMenu.Hide=true`. If this is set, it will deactivate all the T4x Gateway menus, but if the preference `T4S.UI.GatewayMenu.Hide=false` is set additionally, the T4S Gateway Menu will be shown (all other T4x Gateway menus remain hidden).

Using HTTP Display

In principle SAP may allow showing objects in the web browser in nearly the same way as in the SAP GUI. T4S4 can use that with the following configuration steps.

First the used SAP system has to be configured for it because by default it does not allow that functionality. Then T4S4 can use the URL with which SAP will show the needed object data. See details how to define that in SAP in the chapter *SAP Portal iView URL* in the **Teamcenter Gateway for SAP S/4HANA - S/4HANA Preparation Guide**.

- Define the corresponding URL for that SAP HTTP functionality in the mapping configuration, i.e. set the following line (example for a SAP system "ET1" with the client number 800) in the file *t4s_mapping_config/t4s_mapping_config.sd*:

```
catch {
  ::T4S::CONNECTION2SAP::storeWebGuiConfig ET1 800 \
  "http://sapsrv1:8020/sap/bc/gui/sap/its/webgui?~transaction="
}
```

If it is not configured in another special way in SAP, the name "sapsrv1" from that example can be checked in the SAP logon tool directly. Typically it is the same as the first part in the SAP system

definition line "SAP Router String" or if that line is empty then it might be the same as the string in the line "Application Server", or "Message Server", respectively, if that SAP system is defined so.

- Add the value `*_HTTP_DISPLAY` for the requested SAP object type into the preference `T4S_Gateway_Menu_Explorer_Entries` (e.g. `MM_HTTP_DISPLAY` for Material Master display). For checking the correct value for another type, just copy the standard display value and add `HTTP_` into it, e.g. `BOM_DISPLAY` to `BOM_HTTP_DISPLAY`

Then the "T4S4 Gateway" sub menu "Material Master" will show the point "HTTP Display" additionally:

If that point is selected and SAP as well as T4S4 are configured correctly for the SAP HTTP Display functionality, it will not call the SAP GUI for the MM Display, but ask SAP to show it in the web browser.

> Caution:
>
> The transaction does not open a T4S Transfer Window as the standard display transaction does.
>
> As T4S4 does not try sending the SAP login to browser for security reasons, you have to login to SAP in the browser again although you have been logged in already by T4S4.

16.2 Configure the Data Import

Please see chapter *T4x Import Concepts* and *Concept of Triggering Transfer* in the **Teamcenter Gateway - Generic Configuration Guide** for information about the general T4x import concepts.

The following sections only highlight the T4S4 specifics compared to the generic configuration.

In contrast to T4S4 transactions triggered from Teamcenter, all transfers from SAP to Teamcenter are executed asynchronously. This means that no immediate feedback can be delivered to SAP trigger event and further that it takes some time until the data transfer is completed. The following describes the main T4S4 import use case:
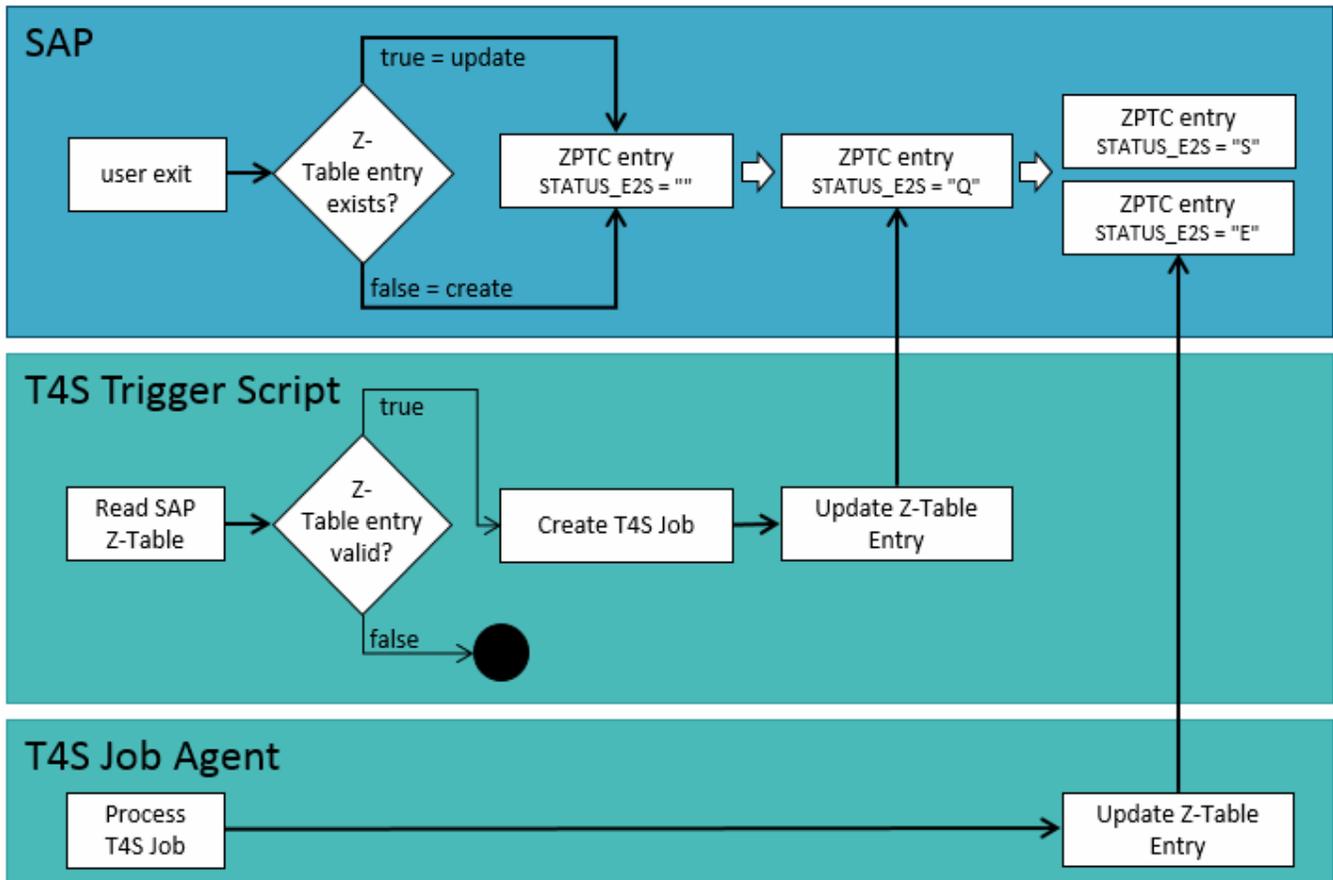
1. A SAP user exit is initiated while saving a SAP object.

2. The user exit creates an entry in custom SAP staging table (ZPTC table).

3. T4S4 scheduled trigger reads entry in staging table and creates a job.

4. The job is executed (as defined in mapping file).

    a. Read SAP data.

    b. Find related TC object.

    c. Create/update TC object.

    d. Update TC attributes.

    e. Additional actions are performed like set status, trigger a TC workflow, send email...

    f. Update entry in staging table (status S for success or E for error).

5. The job status indicates success or failure of import.

Specific Configuration in SAP

The following steps need to be prepared in the SAP:

- Import the ZPTC table and ZPTC functions into SAP with the Transport Package "/TESISPLM/ T4S_ZPTC". See chapter **SAP Transport Packages for T4S Functionality** and Set Up to Trigger Data Transfer From SAP to Teamcenter in the **Teamcenter Gateway for SAP S/4HANA - S/4HANA Preparation Guide**.

- A user exit has to be written in SAP to use the BAPI function /TESISPLM/ZPTC_INSERT to create a new Z-Table entry and populate it with the appropriate data.

T4S4 Configuration

T4S4 offers several pre-defined import jobs and import mapping templates. Be sure to use the correct mapping file to configure the data import for the given object type. Examples are:

SAP object type	ZPTC entry: OBJECT_TYP	Job type
Material Master	MATERIAL	T4S_IMPORT_MM
Document Info Record	DOCUMENT	T4S_IMPORT_DIR
Bill Of Material	BOM	T4S_IMPORT_BOM
Change Master	ECM	T4S_IMPORT_ECM
Production Order	PO	T4S_IMPORT_PO
Work center	WC	T4S_IMPORT_WC
Production Order BOM	POBOM	T4S_IMPORT_POBOM
Equipment	EQUI	T4S_IMPORT_EQUI
Functional Location	FUNCLOC	T4S_IMPORT_FUNCLOC
Vendor	VENDOR	T4S_IMPORT_VENDOR
Purchase Info Record	PIR	T4S_IMPORT_PIR
Single Level iPPE Structure	IPPE_NODES	T4S_IMPORT_IPPE_NODES
Network	NETWORK	T4S_IMPORT_NETWORK
Work Breakdown Structure	WBS	T4S_IMPORT_WBS
Work Breakdown Structure BOM	WBS_BOM	T4S_IMPORT_WBS_BOM
Variant Options	VARIANT_OPTIONS	T4S_IMPORT_VARIANT_OPTIONS

SAP object type	Mapping template
Material Master	*t4s_mm_item_import_template.sd*
Document Info Record	*t4s_dir_item_import_template.sd*
Bill Of Material	*t4s_bom_item_import_template.sd*
Change master	*t4s_ecm_item_import_template.sd*

SAP object type	Mapping template
Production Order	*t4s_prodorder_import_template.sd*
Work center	*t4s_workcenter_import_template.sd*
Production Order BOM	*t4s_prodorder_import_template.sd*
Equipment	*t4s_equi_item_import_template.sd*
Functional Location	*t4s_funcloc_item_import_template*
Vendor	*t4s_vendor_item_import_template*
Purchase Info Record	*t4s_pir_item_import_template.sd*
Single Level iPPE Structure	*t4s_ippe_structure_import_template.sd*
Network	*t4s_network_item_import_template.sd*
Work Breakdown Structure	*t4s_wbs_item_import_template.sd*
Work Breakdown Structure BOM	*t4s_wbs_bom_import_template.sd*
Variant Options	*t4s_variant_option_import_template.sd*

Additionally, T4S4 provides two job types and import mapping templates to support other object types as well. Additional configuration effort will be required to read SAP data within the mapping files:

ZPTC entry: OBJECT_TYP	Job type	Mapping template
OBJECT	`T4S_IMPORT_OBJECT`	*t4s_object_import_template.sd* Support any other SAP business object for import mapping.
META	`T4S_IMPORT_META`	*t4s_meta_import_template.sd* This job type is used for any TC import job that is not necessarily connected to one SAP business object at all.

The following steps need to be performed for T4S4 configuration:

- Prepare the import mapping file *t4s_<object_type>_import_template.sd* and source it in *t4s_mapping_config.sd*.
  For an item import job the following mapping procedures are processed (in that order):

  - `getItemInfo`
    Reads the related SAP Item information. Input parameter `SAPItemNumber` is retrieved from job attribute `ImportObjectId`.

  - `selectTC_Object`
    Identifies the corresponding Teamcenter Item Revision. You can either use a Teamcenter query or if the Material Number matches the Item ID you can use ITK calls to get the latest Item Revision.

  - `createTC_Object`

Creates a new TC object (Item and Item Revision). If you do not want to create a new TC object but continue to the next import step, make sure that the procedure `selectTC_Object` always finds the correct object.

- `SAP_<ObjectType>2TC_Object`
  Updates Teamcenter attributes using the `storeReverseMappingAttribute` function.

- `updateTC_Object`
  Performs any other update function on the Teamcenter Item or Item Revision, i.e. create an alternate ID object, trigger a workflow, set a status, create a form, create a relation…

- `cleanUp`
  Performs any specific action at the end of job execution, e.g. send an email

Refer to chapter *Object Import* in the **Teamcenter Gateway - Generic Configuration Guide** for more details. Also have a look into the **T4S4 API Reference** for details about those mapping procedures.

- If you are using the multi-field key functionality, you need to adapt mapping functions `selectTC_Object` and `createTC_Object` properly; please refer to chapter *Teamcenter Multi-field Key Functionality* in the **Teamcenter Gateway - Generic Configuration Guide** for details.

> Caution:
>
> If the mapping function returns the value `SKIPPED`, do not add a message as second list element because T4S4 will then set the job status to `ApplicationError`. Return the Status `SKIPPED` without a message and the job will finish with status `FINISHED`.

- Configure the T4S4 job agent. See chapter *Job Processing Concept* in the **Teamcenter Gateway - Generic Configuration Guide**.

- Adapt and run the trigger script `Trigger Z-Table import job (Scheduler)` and configure it to run every x minutes. It depends on how often it is necessary to update the TC environment. Provided example: *<T4S_ROOT>/var/test/Jobs/trigger_sap_zjob_scheduler.tcl*. That script will poll for new Z-Table entries and for each found entry it will create a job.
  This example script will check the Z-Table for new entries only. It is an SAP ABAP SQL based query mechanism to read the table. As shown in the example above, only table lines with `Status_E2S` = "" (and `Deleted` = "") are read. Basically, these are new entries, since all other lines which were processed already would have an entry in this field.
  In order to only set the status of an existing ZPTC entry, there is an additional T4S function `::T4S::ZPTC::Set_ZPTC_Status`. Examples in the code to set the ZPTC status to "Q" or "E", respectively, and add a comment:

```
::T4S::ZPTC::Set_ZPTC_Status $el "Q" "JOB ALREADY MOVED TO
T4S-BATCHSERVER-QUEUE"
::T4S::ZPTC::Set_ZPTC_Status $el "E" "Job not processed -
checkJobIsValid returned with $rc!"
```

For information about how to run that script, see chapter *Script-based (scheduled)Triggers* in the **Teamcenter Gateway - Generic Configuration Guide**.
It is possible to run this script like any other T4S test script, or you can add the script to the T4S Scheduler (see *Administration Guide* for details).

16.3 Configure the T4S4 Dataview

Please see chapter **Dataview** in the **Teamcenter Gateway - Generic Configuration Guide** for information about the general T4x Dataview concepts.

The following sections only highlight the T4S4 specifics compared to the generic configuration.

The following SAP object types are supported for the T4S4 SAP Dataview:

- Material Master

- Bill Of Material

- Document Info Record

- Change Master

- Equipment Master

- Functional Location

- Plant

- Workcenter

- Query Result Object

- Generic BOM

Additional Objects can be defined as new functions in the mapping.

In order to open the Dataview select *Window → Show View → Other... → Teamcenter Gateway → SAP Data* from the main menu.

To view the SAP specific information within Teamcenter, select an appropriate object (e.g. an Item Revision) and the SAP object type you want to check. Then T4S4 will process the data for the selected object and the selected SAP object type according to the defined mapping, e.g. a Material Master:
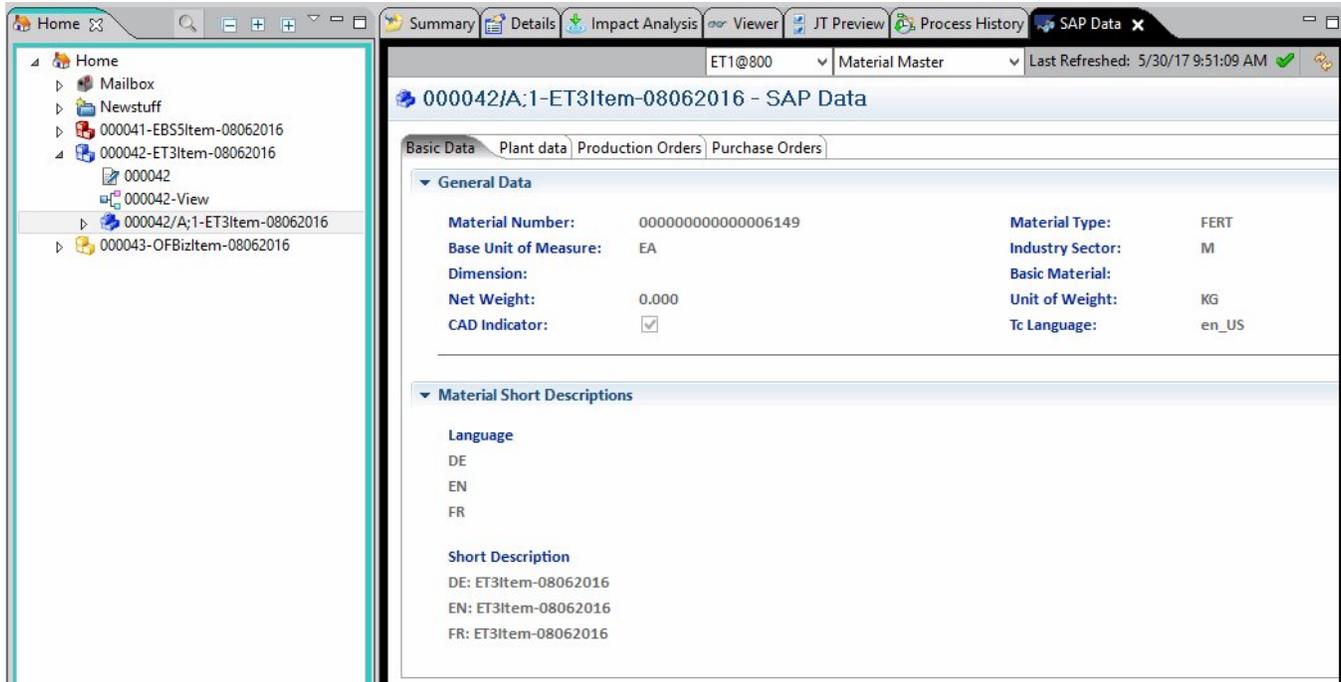
The SAP Dataview control area (the gray part) shows the following information:

- The current SAP connection, e.g. "ET1@800".
  If more than one SAP connection is defined in your T4S4 installation, you can select another from the drop-down list:
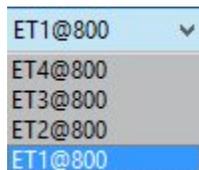
  When selecting a Teamcenter object that has been selected before in the same Teamcenter session, the SAP Dataview will show the data from the same SAP connection as the previous check, no matter if another SAP connection has been used in between.

- The current SAP object type, e.g. "tem". When clicking on the drop-down list, it will show all the SAP object types configured for the selected Teamcenter object:

  When selecting the SAP Dataview, T4S4 will check for which object types the selected Teamcenter object is configured. It checks the `T4S_*TypeList` preferences for the selected Teamcenter object type. The first one it finds is used as the type to show in the SAP Dataview.

However, T4S4 will remember the SAP object type a Teamcenter object has been used with the SAP Dataview. So when selecting a Teamcenter object that has been used with the SAP Dataview before in the same Teamcenter session, T4S4 will use the type that has been opened before.

- "Last Refreshed" is the date and time when the shown data have been processed

- The icon next to the time stamp shows the current processing status or/and result:

   Processing right now

   The processing finished OK

   The processing finished with an error and T4S4 could not get the requested data from SAP.

- Click the refresh button to determine the data again from SAP immediately

The window content shows the desired SAP data as defined in the mapping.

The SAP Dataview is also accessible within the T4S Transfer Window. The following example screenshot shows the SAP Dataview in the T4S Transfer Window after an Item Revision was selected and the T4S Gateway Menu point "Material Master" -> "Change Direct" was activated:

As the T4S Transfer Window is opened for a specific SAP object type, it does not allow switching it.

Specific Configuration in TC

You can use a different set of preferences for the Dataview to increase performance. T4S4 will use the `T4S_<TargetTypeName>DataView*` preferences if they exist. This is an example for Material Master:

- `T4S_MaterialMasterDataViewTypeList`

- `T4S_MaterialMasterDataViewMapping4`

If you do not want to show the tab "SAP Data" in the T4S Transfer Window, disable it by removing the value `CustomData` from the list of tabs in preference `T4S.UI.Transaction.Tabs`. This preference defines which tabs are shown in the T4S Transfer Window. Its default values are:

```
T4S.UI.Transaction.Tabs=
   Attributes
   CustomData
   Status
```

Attributes Shows the Teamcenter data based on a given style sheet

CustomData Shows the SAP Dataview

Status Shows the detailed status messages

T4S4 Configuration

SAP Dataview	
Mapping template	*t4s_prop_mapping_template.sd*
Mapping namespace	`::T4S::PROP::CUSTOM::MAPPING`
Mapping function for handling SAP Material Master	`GetSAPData_MaterialMaster`
Mapping function for handling SAP BOM data	`GetSAPData_BillOfMaterial`
Mapping function for handling SAP DIR data	`GetSAPData_DocumentInfoRecord`
Mapping function for handling SAP ECM	`GetSAPData_ChangeMaster`
The same naming pattern is valid for the remaining object types.	

As the data to be shown needs to be read from SAP, T4S4 uses the same mechanism to get it as in any of the Reverse Mapping functions, so call the same function `SAPFieldMapping` with the same parameters here. Before the call of function `SAPFieldMapping`, make sure you will receive the data for respective object type from SAP by calling function `GetSAPData_<object_type>`, e.g. `GetSAPData_MaterialMaster`.

In order to find the correct call to get the desired SAP data, see the **T4S4 API Reference**.

> Caution:
>
> In order to identify the SAP objects correctly according to your Teamcenter settings, be sure to adapt the corresponding lines in the mapping file `t4s_prop_mapping_template.sd` (e.g. `set MaterialNumber …`) in the same way as in the corresponding lines in the mapping files for those object types.

Please refer to the examples in the *t4s_prop_mapping_template.sd* for getting an overview of the SAP Dataview configuration.

16.4 Configure the Transfer Window

Please see chapter **Configuration of Teamcenter GUI Extensions** in the **Teamcenter Gateway - Generic Configuration Guide** for information about the general configuration of the T4x Transfer Window.

16.5 Configure the SAP Query

Please see chapter *External Queries* in the **Teamcenter Gateway - Generic Configuration Guide** for information about how to configure the external queries in general.

The following section only highlights the T4S4 specifics compared to the generic configuration.

Specific Configuration in TC

The following additional preferences values allow Teamcenter using the T4x user defined search queries (first tell Teamcenter to use the additional lib `libt4x_query`, then the query names):

```
TC_customization_libraries=
libt4s
libt4x_query

T4X_SupportedUserQueries=
T4S_DEMO_QUERY
UserSearchQuery1
```

As you may have installed more than one T4x product in the same Teamcenter installation, you have to tell T4x which product has to execute the queries by setting the following preferences:

```
T4X_QUERY_TargetSystem4T4S_DEMO_QUERY=
T4S

T4X_QUERY_TargetSystem4UserSearchQuery1=
T4S
```

Prepare the following in order to use the T4S4 query functionality:
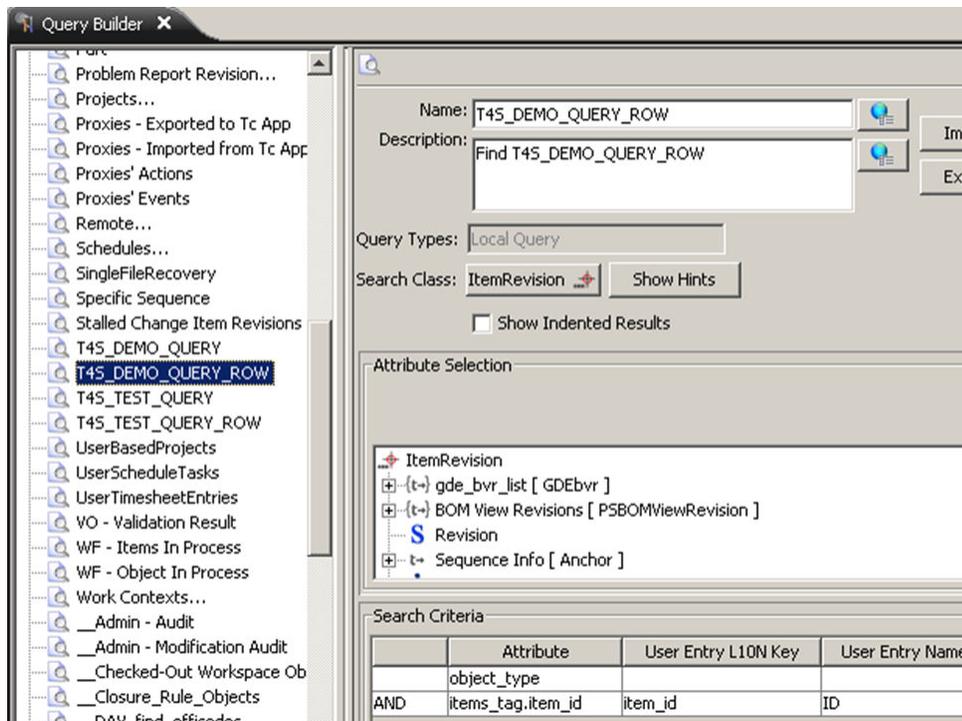
- Use the Teamcenter Query Builder to create user search query (example name T4S_DEMO_QUERY):
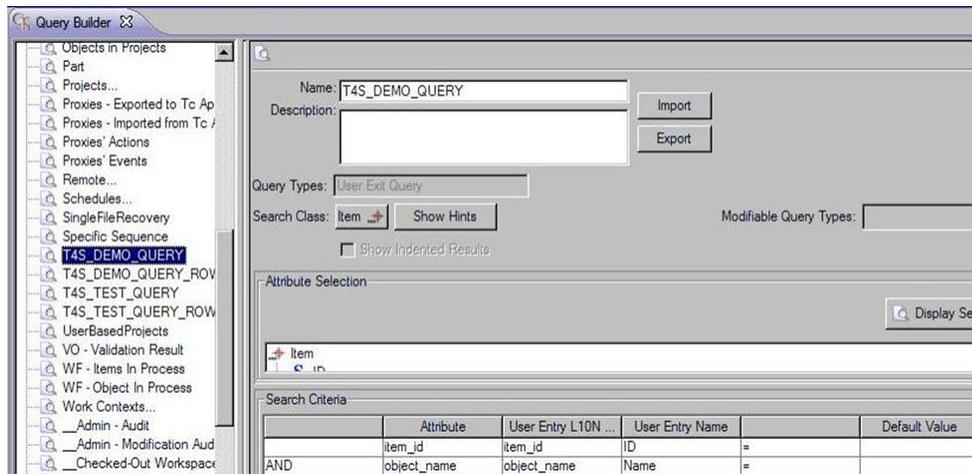
Corresponding XML User Query Import file example:
<t4s_SAP_query_example.xml>

- Use the Teamcenter Query Builder to create row search query (Example name T4S_DEMO_QUERY_ROW):

- Flag the user query for running through the user exit. Therefore, do the following in the Teamcenter command shell:

```
%TC_BIN%/tc_set_query_where_run –u=username –p=userpassword –
query=T4S_DEMO_QUERY –run=user
```

T4S4 Configuration

The mapping file *<GS_Root>/var/template/t4sdemo/mmap/t4s_mapping_config/
t4s_query_mapping_template.sd* provides examples how it needs to be configured and how the results
are given back to Teamcenter.

The actual "external" search runs the function `executeQuery4<object_type>` (e.g. for MM:
`executeQuery4MaterialMaster`). It allows controlling the behavior of a search query. In the
procedure `executeQuery`, you have to specify which search procedure has to be called for which user
search query, e.g. `executeQuery4MaterialMaster`.

In order to search SAP objects from within Teamcenter, execute the defined Teamcenter query, e.g. the
T4S_DEMO_QUERY that allows entering an ID and a name to find matching SAP MMs. This example will
search all the SAP MMs whose ID is starting with "P-3" or with the name (material short description)
starting with "rio" (not case sensitive):
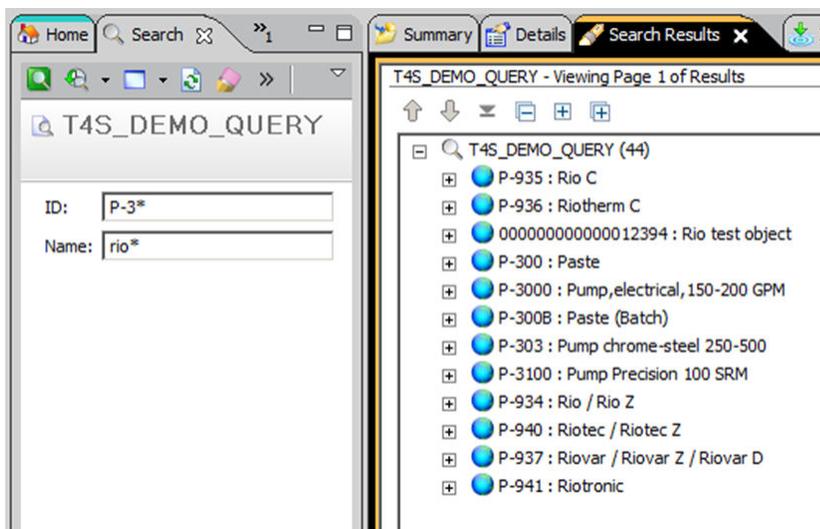
Depending on your SAP system, the result might not show MMs with IDs containing only numbers if the
ID search string does not start with an asterisk (*) because of the leading zeros in the internal SAP
number format. So if you want to find all MMs '1234*' it may show 1234A but not 12345, so maybe
enter *0001234* instead.

In order to modify the shown data in the search result list, adapt the following line in the corresponding
`executeQuery4` function, e.g. `executeQuery4MaterialMaster` for the SAP Material search.
Example:

```
set ::StatusInfo(Query:Row:$RowIndex:1) "$::MMResultDat($RowInfo): My test
text"
```

Then the fixed string "My test text" will be shown in every search result line. Please do not modify the first part of this value ($::MMResultDat($RowInfo)) because this is what T4S4 uses to identify the SAP object.

When selecting any of those result lines and activating the Teamcenter "Viewer" tab, you may see the data of that SAP object as defined in the corresponding query mapping function, e.g. executeQuery4MaterialMaster for a SAP MM query:

```
set ::StatusInfo(Query:Column:1)  MATERIAL-OBJSTRING
set ::StatusInfo(Query:Column:2)  MATERIAL
set ::StatusInfo(Query:Column:3)  MATL_DESC
```
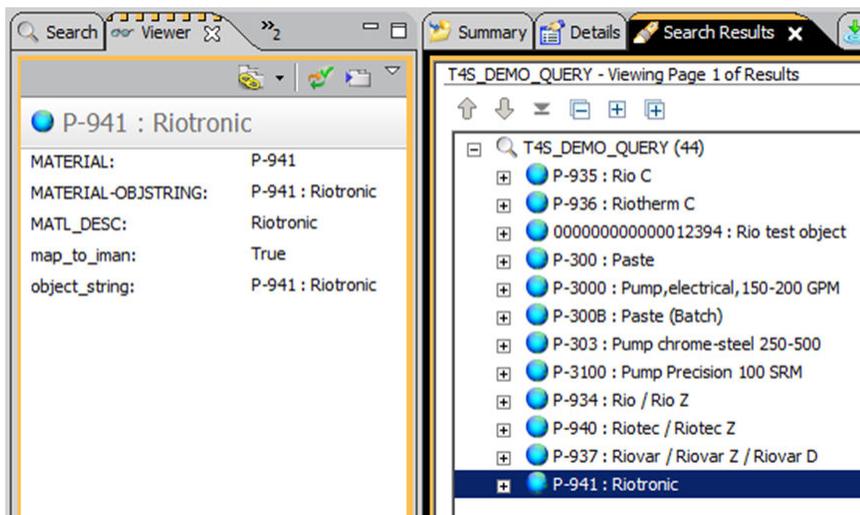

Caution:

Only the USER queries have to be flagged; do not flag the "row" queries! This is because the user does not have to execute those queries explicitly; they are called by T4S4 internally only, see below. If a Query was flagged to run as USER EXIT QUERY, it can´t be modified or deleted anymore. In case you need to do that, it is necessary to flag the query as LOCAL QUERY. The following command will do this:

```
%TC_BIN%/tc_set_query_where_run –u=username –p=userpassword –query=T4S_DEMO_QUERY –run=iman
```

For performance reasons the SAP query is limited to a certain number of results which is defined in the query mapping file, e.g. for a SAP MM query it is limited to 100 by default:

```
set ::SAPDat(MaterialSelection:PARAMETER:MAX_ROWS) 100
```

The example query searches in Teamcenter for objects of type SAP2_T4S_Item Revision (Teamcenter display name: "T4S Item Revision") with **the same Item ID as the queried SAP MM ID**

SAP Data View

Additionally T4S4 allows asking SAP for data from the found objects and showing them in the T4S4 SAP Data View. Select the "SAP Data" tab and find the same data of the object represented by the selected line that you see when selecting the Item Revision that represents the same SAP MM. See **Configure the T4S4 Dataview** for more details.

For adapting the data T4S4 will show in the "SAP Data View" for an SAP object from this query result list, modify the function `GetSAPData_QueryResultObject` in the file *t4s_prop_mapping_template.sd*

By default it checks which query had created the current result list. This may be adapted as well as the data to show:

```
if {$QueryName eq "T4S_DEMO_QUERY"} {}
```

Copying Results to the Clipboard

Up to that point, you only know that the shown objects exist in SAP, but not if there are corresponding objects in Teamcenter (e.g. if any Teamcenter object fulfills the conditions defined in the query, e.g. an Item with the same ID as the SAP MM or an Item Revision with the IRM attribute `SAPMatNo` filled with the SAP MM ID).

In order to find a corresponding Teamcenter object, mark the desired result line and copy it into the Teamcenter clipboard (select Edit – Copy from the main Teamcenter menu or press Ctrl+C). This will trigger the row query defined in the following line in the query mapping function `executeQuery4MaterialMaster` (in this example: T4S_DEMO_QUERY_ROW):

```
set ::StatusInfo(Query:Row:$RowIndex:0) [buildRowQueryDataString
T4S_DEMO_QUERY_ROW ID $::MMResultDat($RowInfo)]
```

The found Teamcenter object will be copied to the Teamcenter clipboard (the Item that fulfills the conditions defined in the query). If the row query could not determine a corresponding Teamcenter object, the result will be the Teamcenter error message "Invalid objects are selected to copy" and nothing is stored in the Teamcenter clipboard.
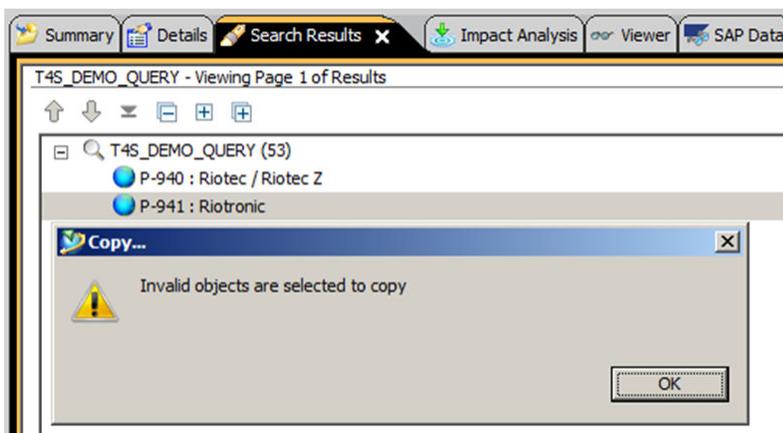
SAP Query Import

In order to import the object, i.e. create or update a corresponding Teamcenter object, right-click the selected line in the Teamcenter search result and select **Import Object to Teamcenter**. This triggers the row query mapping again and starts an Import Job:
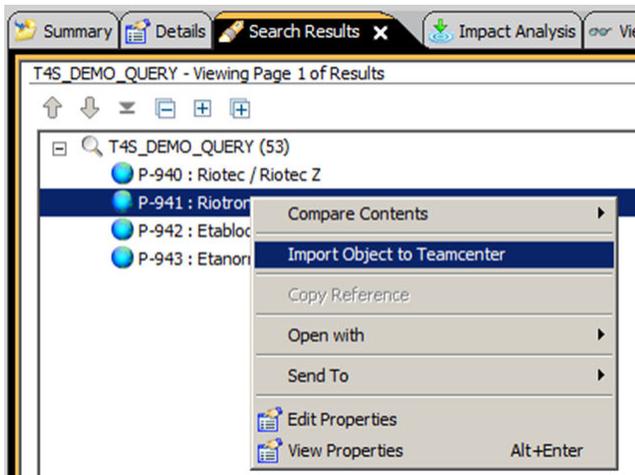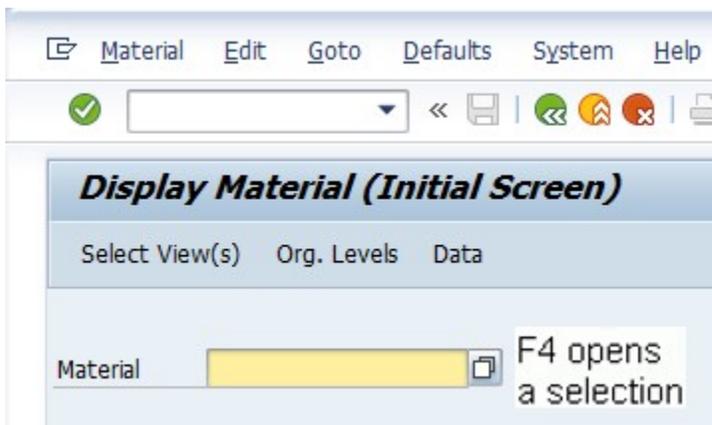
17. Cross-Functional

17.1 Useful SAP Key Shortcuts

Working in SAP GUI can be accelerated by the usage of shortcuts:

- F1: context sensitive help

- F3: Back

- Shift-F3: Exit

- F4: Open selection "Possible entries" (see screenshot)

- F7: Display

- F8: same as the "Execute" button where available

- F12: Cancel current operation

Note that you may configure the way SAP behaves when you press F4 from the main menu. Select Help – Settings – F4 Help.

17.2 Additional SAP Object Update in a Transaction

During a T4S4 transaction the SAP object is created or updated with the data specified in the T4S4 object mapping file. In some cases it is necessary to update data after the actual transaction was completed, e.g. setting a status to a SAP object may be possible only after all the other data is stored to this object.

In order to avoid setting those additional data in several Workflow handlers or even in completely separate transactions that would highly increase the complexity, T4S4 offers another possibility: a default mapping procedure is available to define post update data.

If the corresponding function exists in the current mapping files, it is executed after the actual transaction is finished. Any data stored in the array `::SAPDat` in the regular mapping procedures is used in the actual transaction only. The post update data have to be defined separately in this function.

- MM
  `::T4S::MM::CUSTOM::additionalUpdates4MaterialMaster TransactionId MaterialNumber`

- DIR
  `::T4S::DIR::CUSTOM::additionalUpdates4DocumentInfoRecord TransactionId DocumentNumber DocumentType DocumentPart DocumentVersion`

The valid return values of those functions are:

- In OK case: [list "OK"]

- In error case: [list "ERROR" "" "SomeErrorMessage"]

You can find an example in the file *<T4x_GS_ROOT>\var\template\capabilities\T4S \t4s_documentinforecord_mapping_doc.sd* under "Additional DIR Update".

17.3   Using the SAP Number Generator to assign TC IDs

T4S4 allows the interaction with SAP from Teamcenter "New Item" dialog box when assigning an Item ID. This only works for "Items" and derived sub classes.

The functionality is controlled by preferences and a dedicated mapping file. After T4S4 retrieved the new number it will validate if any object with this ID already exists in the other system. If the new number was retrieved from SAP, T4S4 checks in Teamcenter if an Item with that number already exists. If the number was given from Teamcenter, T4S4 checks against SAP if this number is already used.

> Caution:
>
> If the connection to SAP is not yet established, this will fail and an error message is shown.
>
> For checking the Teamcenter ID against the SAP it can happen that the given number is already taken. Therefore the process runs a loop to draw the number. In default case the loop is limited to 10 attempts – if a free number is found earlier the loop will be discontinued. The number can be de-/increased by setting the preference `T4X_IDGEN_max_loop`.

Specific Configuration in TC

Take the following steps:

- Enable Teamcenter to load the additional T4X library `libt4s_idgen`:

  ```
  TC_customization_libraries=
  libt4s
  libt4x_idgen
  ```

- Define a list of valid ItemTypes that will retrieve the number by T4S4 from SAP:

  ```
  T4X_IDGEN_TypeList=
  SAP2_T4S_Item
  ```

  If that preference is not defined, T4S4 will validate and use the entries of the preference `T4X_IDGEN_MaterialMasterTypeList` instead. If that is not defined either, it will use `T4S_MaterialMasterTypeList`. ItemTypes that are not listed will retrieve the number from Teamcenter naming rules (i.e. they will not use this T4x functionality).

- Optional: Define the correct target system. The default value is `T4S`. Only relevant in a T4S4 installation that is combined with T4O.

  ```
  T4X_IDGEN_TargetSystem4ItemRevision=
  T4S
  ```

- Optional: Define the desired function to use. Two options are available:

  - `T4X_IDGEN_USE_EXTERNAL_NUMBER`
    Default, retrieves number from SAP as defined in the idgen mapping file. This will call the mapping function `getReservedNumber`.

  - `T4X_IDGEN_EXTERNAL_VERIFY_OF_NUMBER`
    Uses Teamcenter naming rules and then validates in SAP if an object already exists with this new ID in Teamcenter. This will call the mapping function `checkIfObjectExists`. A valid Teamcenter naming rule must be configured in BMIDE!

    ```
    T4X_IDGEN_Function4ItemRevision=
    T4X_IDGEN_USE_EXTERNAL_NUMBER
    ```

  - Optional: Define the maximum number of validation checks and repetitions before T4S4 aborts with an error (integer value; default value is 10):

    ```
    T4X_IDGEN_max_loop=
    10
    ```

T4S4 Configuration

The mapping file *t4s_idgen_mapping_template.sd* contains the configuration that needs to be adapted. Please follow the given examples for pulling a number from SAP, changing the number to desired format or validating if an object already exists in SAP for the given number.

Find the procedure `getReservedNumber` where you can define by the given TC ItemType what sub procedure should be called. There are already pre-defined sub procedures for some common cases with the functionality to reserve the numbers (MM) or create the objects (DIR, ECM) in SAP:

- `getReservedNumber4MaterialNumber`

- `getReservedNumber4DocumentInfoRecord`

- `getReservedNumber4EngineeringChangeMaster`

You can create your own procedure and custom calls to support other SAP objects.

Define the mandatory values required to get the number from SAP.

17.4   Using a Custom BAPI for a Standard Transaction

T4S4 uses standard SAP BAPIs for all the transactions. E.g. `BAPI_MATERIAL_SAVEDATA` to create and update Material Masters. In order to call another SAP custom function instead, set the following commands in your *t4s_mapping_config.sd*.

To tell T4S4 for which object type handling you want to use another SAP function, e.g. `::TPSAP::MM::CustomerBAPI` for Material handling.

- State the name of the standard BAPI, e.g. `BAPI_MATERIAL_SAVEDATA`

- State the name of the function to be used instead of the default one, e.g. `My_MM_BAPI`

```
set ::TPSAP::MM::CustomerBAPI(BAPI_MATERIAL_SAVEDATA)   My_MM_BAPI
set ::TPSAP::DIR::CustomerBAPI(BAPI_DOCUMENT_CREATE2)   My_DIR_BAPI
set ::TPSAP::ECM::CustomerBAPI(CCAP_ECN_HEADER_CREATE)  My_ECM_BAPI

# "::TPSAP::BOM" refers to the old SAP BOM BAPIs
# and "::TPSAP::BOM::BAPI" to the new
set ::TPSAP::BOM::CustomerBAPI(BAPI_BOM...)             My_BOM_BAPI
set ::TPSAP::BOM::BAPI::CustomerBAPI(BAPI_BOM...)       My_BOM_BAPI
```

In order to call the SAP BAPI to change the BOM CAD indicator. This functionality is usually included to SAP by importing the transport package that T4S4 provides as *<GS_ROOT>\var\template\t4s\sap \TransportPackages\TESISPLM_T4S_MISC.zip*:

```
set ::TPSAP::TESISPLM::CADKZ::CustomerRfc(ChangeCADKZ) My_CAD_ind_BAPI
```

> **Caution:**
>
> T4S4 does not support this for all the BAPIs yet. Please check if it works for the ones you want to change.

17.5 Technical Name of SAP GUI Fields

To display the name of a field in the SAP GUI, perform the following steps within a SAP GUI window:

- Click into the desired field.

- Press F1 (Help).

- Click the button Technical Information (may be a text field or an icon with hammer and wrench).

- A new window will open.

- The "Screen field" there shows the internal field name.

- To see how this field is formatted (string length, number format...), double click "Data Element" in the section "Field data".

- Another window will open.

- Open the tab "Definition" and see the section "Data Type" there.

The format in which you have to transfer the data may be language dependent (especially "date", e.g. "17.06.2005" or "06/17/2005") but this is not shown here.

- The data type "DATS" is a "char 8" field. To transfer such a date, you have to transfer it as "YYYYMMDD" (`[binary -format [clock seconds] %Y%m%d]`).

- The date type "DATE" is "char 10" ⧠ pass "`[binary -format [clock seconds] %d.%m.%Y]`.

- Both of them "DATS" and "DATE" are shown in the same way in SAP (e.g. "12.05.2006" with German SAP date settings) and may be cleared with the TCL formatted string `[string repeat "0" 10]`x.

If you click on "Field Name" instead of "Data Element", SAP will show the database table as known from the transaction `SE11` ("Dictionary: Display View"). This table additionally shows the field length, e.g. length 13 and 3 decimals for the type QUAN which is used for a BOM component quantity.

17.6 Read SAP Table

T4S4 can read the contents of a SAP table in the database with the function `::TPSAP::MISCELLANEOUS::RfcReadTable`.

Parameters:

- `::SAPDat(ReadTable:PARAMETER:QUERY_TABLE)`
  The table name.

- `::SAPDat(ReadTable:PARAMETER:DELIMITER)`
  Separator for the data in the output result string.

- `::SAPDat(ReadTable:PARAMETER:NO_DATA)`
  If not left empty only table field information is returned but no data.

- `::SAPDat(ReadTable:OPTIONS:TEXT:*)`
  Search parameters in Open SQL (>; <; <>; =; like...)

- `::SAPDat(ReadTable:FIELDS:FIELDNAME:*)`
  You have to specify the columns from the table that shall be in the output of the function.

Mapping Example:

```
# Set parameters
set ::SAPDat(ReadTable:PARAMETER:QUERY_TABLE) /TESISPLM/ZPTC
set ::SAPDat(ReadTable:PARAMETER:DELIMITER)    ;
set ::SAPDat(ReadTable:PARAMETER:NO_DATA)      ""
# Filter (optional, can be empty)
set ::SAPDat(ReadTable:OPTIONS:TEXT:1) "ID >= '10'"
set ::SAPDat(ReadTable:OPTIONS:TEXT:2) "and OBJECT_TYP = 'MATERIAL'"
set ::SAPDat(ReadTable:OPTIONS:TEXT:3) "and OBJECT_ID like '%24728'"
# define output columns
set ::SAPDat(ReadTable:FIELDS:FIELDNAME:1) ID
set ::SAPDat(ReadTable:FIELDS:FIELDNAME:2) OBJECT_ID
# function call and result
set rc [::TPSAP::MISCELLANEOUS::RfcReadTable]
puts "::TPSAP::MISCELLANEOUS::RfcReadTable finished with >$rc<"
foreach elem [lsort -dictionary [array names ::MiscResultDat]] {
  puts "::MiscResultDat($elem) = $::MiscResultDat($elem)"
}
```

The result is stored in the array `::MiscResultDat`. The output of your search in `DATA:WA:*` and the description of the tables columns in `FIELDS:*`.

The output string in `::MiscResultDat(DATA:WA:*)` contains only the fields selected in `::SAPDat(ReadTable:FIELDS:FIELDNAME:*)` by numerical order, separated by the chosen delimiter:

```
::TPSAP::MISCELLANEOUS::RfcReadTable finished with >OK<
::MiscResultDat(DATA:WA:1) =        6743 ;000000000000024728
::MiscResultDat(FIELDS:FIELDNAME:1) = ID
::MiscResultDat(FIELDS:FIELDNAME:2) = OBJECT_ID
::MiscResultDat(FIELDS:FIELDTEXT:1) = Natural Number
::MiscResultDat(FIELDS:FIELDTEXT:2) = Char 80
::MiscResultDat(FIELDS:LENGTH:1) = 000010
::MiscResultDat(FIELDS:LENGTH:2) = 000080
::MiscResultDat(FIELDS:OFFSET:1) = 000000
::MiscResultDat(FIELDS:OFFSET:2) = 000011
::MiscResultDat(FIELDS:TYPE:1) = I
::MiscResultDat(FIELDS:TYPE:2) = C
```

17.7 Installed SAP Languages

This is applicable to every SAP data object; it may be important if you try and set a language specific SAP field by T4S4. SAP will only process the data correctly if the language is shown as available in SAP:

- Log in to SAP

- Start transaction `SMLT` (Language Management)

- To obtain the valid language code for a language, see the list you get when setting a language specific field directly in SAP
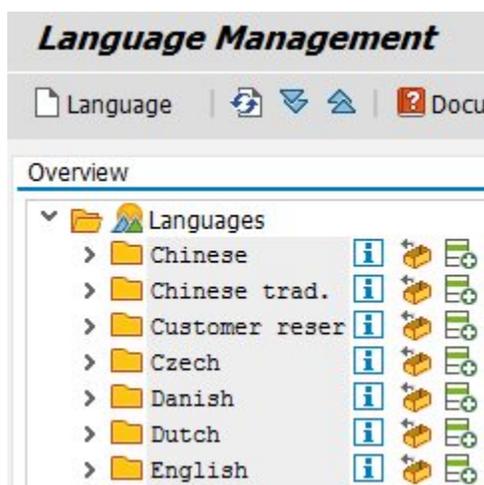
You can also try and login to the SAP system with any language code. If the corresponding language is not installed, SAP will refuse the access with the message "Please enter one of the installed languages. Unfortunately, not every SAP system provides the list of valid language codes here.

> Caution:
>
> When you modify a language specific field (e.g. the "Material Description") and click on the button to select the language for your entry, you might get a list with more languages. This does not mean that SAP accepts data in all these languages by an interface!

17.8 How to Transfer Several Sets of Similar Data in One Transaction

Typical use cases:

- Create/Update SAP Material Masters for multiple plants.

- Create/Update SAP Bill Of Materials for multiple plants and/or usages.

- Create/Update SAP Classification with multiple classes.

T4S4 offers this functionality in order to transfer several sets of data to SAP in one transaction although those data actually need one SAP transaction each. In fact, T4S4 transfers them in several steps to SAP (because else SAP will not store all the data correctly) but from the user's point of view, it appears as one transaction only. The advantage is that all the data can be defined more easily in the mapping without additional steps (e.g. using more than one Workflow handler with different arguments).

Each single transaction will contain all the data set before the line with `::T4S::storeSAPDat2MultiTransferCache`. So, the different data for the different plants needs to be set between the "MultiTransferCache" calls. Put such a "multi transfer" section at the end of the mapping function (e.g. for MM `::T4S::MM::CUSTOM::MAPPING::TC_Object2SAP_MaterialMaster`).

You can find an example in the file *<T4x_GS_ROOT>\var\template\capabilities\T4S\t4s_materialmaster_mapping_doc.sd* under "FAQ - How to transfer several sets of similar data in one transaction".

> Caution:
>
> Up to now T4S4 supports the "multi transfer" for MM and BOM Header only.
>
> Although with MM the multi plant transfer works the same for portal transactions and Workflow, it works in a Workflow only with BOM, not in a portal transaction, i.e. one that has been started from the T4S Gateway menu in Teamcenter.
>
> As T4S4 has to handle such a "multi transfer" with several internal SAP transactions actually (although it looks as one transaction to the user only), it cannot be done with the portal transactions "Create" nor "Change" because those do exactly one SAP transaction. But it works with "Create Direct" and "Change Direct" same as in a Workflow
>
> Besides the multiple classifications for Material Master (see below), do not set any mapping data after the calls `::T4S::storeSAPDat2MultiTransferCache <type>`.
>
> Any data not set after the first "MultiTransferCache" call will be used as they have been set before. If the different transfers have to set different variables, be sure to set all of them in each transfer; maybe some of them have to be empty. Else each variable that is not set explicitly will contain the value from the previous transfer. So you may unset some data for the second transfer.

17.9 Get SAP Connection Data

Use the function `::T4S::CONNECTION2SAP::testSAPLogin` to check your SAP connection. It will return `OK` if the connection is fine.

As T4S4 allows using several SAP systems it might be useful to check which SAP system is connected in a transaction. The function `::TPSAP::RfcGetAttributes` sets the following variables.

- `::TPSAP::RfcAttributes(DEST)`
  SAP target system

- `::TPSAP::RfcAttributes(SYSID)`
  SAP system ID

- `::TPSAP::RfcAttributes(CLIENT)`
  SAP Client

- `::TPSAP::RfcAttributes(KERNEL_REL)`
  SAP Version

> Caution:
>
> These values are only updated if the function is called, so call it every time before you use these variables.

17.10 Correct Number and Date Format

SAP may reject a transaction that contains data in a format that is not expected (mainly for date and decimal numbers).

Using SAP GUI

Look up the data format that SAP expects as follows:

- Log in to the SAP system

- Type transaction code `SU01` for User Maintenance.

- Type the appropriate user ID (the SAP user that will be called by T4S4).

- Click on the button with the glasses symbol ("Display").

- Click on the tab "Defaults"

You should see something like the following:
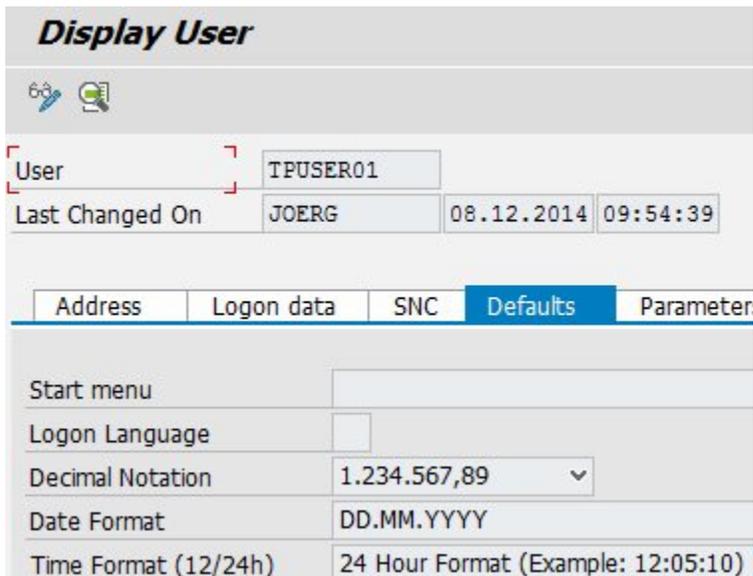
The mapping needs to be set up such that SAP receives the data in the format shown for the specific SAP system.

Using a Mapping Function

The function `::TPSAP::getSystemInfo` retrieves data about the settings of the currently connected user:

- User Name
  `$::TPSAP::SystemInfo(USER_NAME)`

- Decimal Sign
  `$::TPSAP::SystemInfo(DECIMAL_SIGN)`
  You can also use `::TPSAP::getDecimalSigns`.

- Login Language
  `$::TPSAP::SystemInfo(LANGUAGE)`
  You can also use `::TPSAP::getLoginLanguage` and `::TPSAP::getLoginLanguageISO`.

- Date Format
  `$::TPSAP::SystemInfo(DATE_FORMAT)`
  You can also use the functions `::TPSAP::getExternDateString` for external data format
  and `::TPSAP::getInternDateString` for the internal data format.

- SAP System Release Information
  `$::TPSAP::SystemInfo(SAP_SYSTEM_RELEASE)`
  You can also use `::TPSAP::getReleaseInfo`.

17.11   Check if an Object is Locked in SAP

In most cases, the reason for a locked SAP object is that it is being processed. So SAP prevents any other modification on the same object by putting a lock on it. To check such a lock, use the SAP transaction `SM12` "Select Lock Entries". The following screenshot shows it with a lock on the DIR:
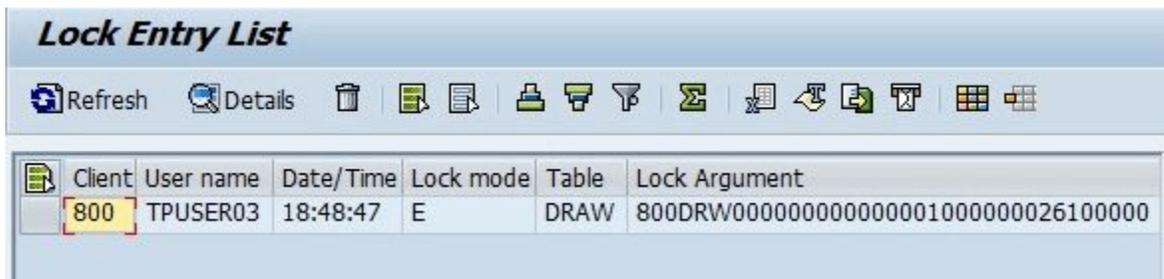
In case of a DIR the "Lock Argument" is
`<SAP_client><DIR_type><internal_DIR_ID><internal_DIR_ID><DIR_part><DIR_version>`.

In order to check such lock, use the T4S4 test script "SAP get ENQUEUE data" (*sap_get_enqueuedata.tcl*) which needs two parameters:

- Object ID
  This is the SAP internal object identifier, so in most cases not just the SAP Object ID itself but a combination of the SAP client, the Object ID and may more data, see below

- Object Table
  This is the SAP internal Object Table name, e.g. MARA for MM, DRAW for DIR

In order to find the lock on that DIR using the T4S4 test script, you have to enter the table name "DRAW" and exactly the same string shown here in the script parameter "Object ID", see screenshot below. So when trying to check locks on other object types, it is helpful to check the correct parameter format for the "Lock Argument" directly in SAP using that transaction SM12:

Caution:

With an incorrect "object ID" (from SAP perspective the "lock argument"), SAP will not answer something like "object not found" but just return no lock information, so the T4S4 script result is "Object is not locked!" So be sure to check that parameter accurately.

17.12 Call Custom SAP BAPI

In some cases a business use case requires functionality that is not provided out of the box by T4S4. If there is a standard or customer provided BAPI function in SAP that provides the required capability, it can be used by T4S4 to close this gap.

There is a special test script that will generate the code example to execute this function. This example can then be tailored with less effort to the customer needs. To generate the code, select the test script *create_code4genericSAP_call.tcl* (Generate code in order to call a generic SAP function).

In the field "SAP Function" you can enter the full name of the BAPI that needs to be called. You also need to select a "Pattern" for the input array (the name can be selected freely). Optionally you also can define the name of the generated file (do not enter a full path):

Caution:

Make sure that your SAP user has the access rights to call the BAPI.

After the script has been executed, you will find the generated file in *<GS_Root>\tmp*.

Depending on your BAPI function, the generated code will include init functions (e.g. `init__TESISPLM_ZPTC_INSERT_PARAMETER`) and the main function that will prepare the BAPI call (e.g. `call_TESISPLM_ZPTC_INSERT`). The BAPI itself is called by the standard function `::TPSAP::performSapCall` with the generated input parameters.

17.13   Basics on SAP Long Text

By default, SAP only allows up to 132 characters per line in a long text. If you transfer a longer line, it will be cut. The line index you set in the mapping has to be contiguous and ascending, even if you transfer a text in different languages.

When setting several lines in more than one language, then all the indexes for the same language must be set contiguously (else SAP will not store all of them).

The index used in the mapping line is not necessarily the line number you will find in SAP.

Depending on the SAP configuration if a Long Text is set, the first line of one language is what SAP will show as "Description" in the same language.

In some cases, SAP will not provide an error message if a requirement for setting a long text is not fulfilled. For example if the SAP MM "Sales View" is not activated, SAP will not store the "Sales text" but the transfer will be successful.

17.14   Basics about SAP Object Links

SAP allows linking different objects together, so you can access the one object directly from the other object without having to know its ID. In most cases, SAP only allows the link creation from a DIR. But you may access the DIR from the linked object as well.

Depending on the SAP configuration, the link creation to an object that does not exist is maybe not allowed (SAP gives an error message and the link is not created). If it is allowed and the target object is created afterwards, the link will be shown.

17.15   Ask SAP GUI for Type Abbreviations

In order to find the abbreviations SAP expects from T4S4 (e.g. the "Material type"), you can configure the SAP GUI to show:

- In the SAP GUI (no active transaction, "SAP easy access"), click the icon "Customize Local Layout" (see red arrow in the following screenshot) or press ALT-F12:

- Select "Options" (in older SAP GUI versions there was tab "Expert" to be activated)

- Open "Interaction Design" and select "Visualization 1"

- Activate the check box "Show Keys within drop-down lists".
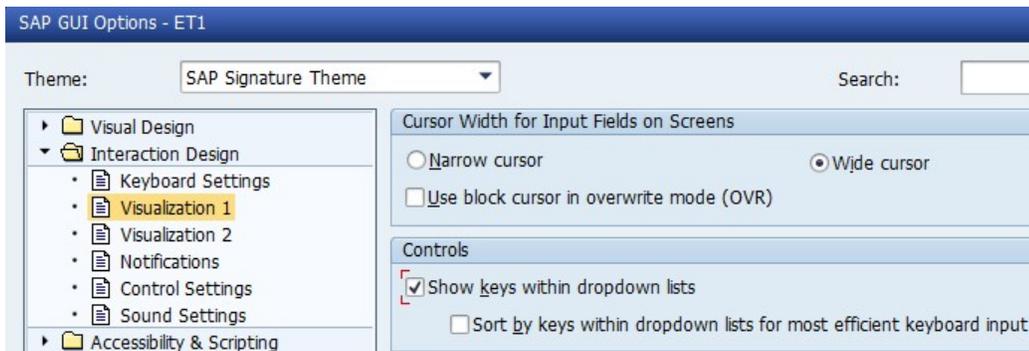
- Then you will find the whole expression and the abbreviation, e.g. "FERT Finished Product" instead of only "Finished Product" for the material type.

A. Glossary

A

Admin

is the term used in this document for people who install and configure Teamcenter and its components. This is in contrast to the "user" role.

Admin GUI

Teamcenter Gateway Administrative User Interface.

AI-Object

Application-Interface Object

Apps

See "GS".

B

BGS

Basic Gateway Service.

BGS_ROOT

The installation directory of the T4S Basic Gateway Service (e.g. *C:\Siemens\T4x_BGS*)

BMIDE

Teamcenter Business Modeler IDE (Integrated Development Environment).

BOM

A Bill Of Materials is a list of the parts or components and their quantities that are required to build a product.

D

Data Carrier

A SAP medium on which originals are stored.

Dataview mark-up

is the language understood by the Dataview. The Dataview receives messages written in this language from the T4x server. Such messages can be formatted as XML or JSON. Normally users do not see such messages. They may however appear in log files or error messages. The so called prop mapping (e.g.

t4s_prop_mapping_template.sd) contains TCL commands that compose messages in the Data View mark-up.

DIR

DIR is the abbreviation for a SAP Document Info Record.

Document Key

A Document Info Record is identified by the combination of Document Type, Document Number, Document Part and Document Version.

Document Structure

A Document Structure is like a Bill Of Materials for Documents.

E

EA

stands for Enterprise Application, any software or set of computer programs used by business users to perform various business functions in context of current integration's portfolio with Teamcenter.

ECN

The Engineering Change Notice can also be called an Engineering Change Note, Engineering Change Order (ECO), or just an Engineering Change (EC).

EPM

Enterprise Process Modeling.

G

GRM

The Generic Relationship Management provides a general way in which two objects can be associated via a relationship.

GS

Gateway Service, manages the communication between Teamcenter and the Enterprise Application.

GS_ROOT

The installation directory of the T4S Gateway Service (e.g. *C:\Siemens\T4S_GS* or *C:\UGS\T4S_Apps*)

GUI

Graphical user interface.

I

ITK

The Integration Toolkit (ITK) is a set of software tools provided by Siemens PLM Software that you can use to integrate third-party or user-developed applications with Teamcenter.

J

JDBC

Java Database Connectivity is an application programming interface (API) for the programming language Java, which defines how a client may access a database.

JSON

JavaScript Object Notation is a lightweight data-interchange format[1].

K

KPro

Kpro stands for Knowledge Provider. See also Data Carrier.

L

LOV

List of Values.

M

Mapping

The mapping is part of the T4x configuration. It contains the code that controls the behavior of the data transfer between Teamcenter and the ERP system.

MFK

Multi-key functionality in Teamcenter.

MM

MM is the abbreviation for a SAP Material Master.

---

1   JSON.org

O

Object Link
A relation between SAP objects like Material Master and Document Info Record.

Object Management Record
Belongs to a SAP Change Number and Documents changes of one particular SAP object like a Material Master.

OOTB
Out of the box.

Original
A representation of a file in SAP.

P

PIR
PIR is an abbreviation for a SAP Purchase Info Record.

Portal Transaction
This means that a T4S4 transfer to SAP that is not triggered by a workflow handler but via the T4S4 Gateway Menu.

R

RAC
stands for Rich Application Client also referred to as rich client or portal.

Revision Level
Used to show changes with reference to a change to a SAP Material Master or Document Info Record.

S

SAP Portal iView URL
Can be used to show sap content in a browser window.

SSL
Secure Sockets Layer.

T

T4S Gateway Menu

A Teamcenter menu item that is provided by T4S4 to enable Portal Transactions to SAP and provide SAP connectivity options.

T4x

The entire Teamcenter Gateway product family.

TargetTypeName

This is the T4x internal name for the transaction type. E.g. `MaterialMaster` or `DocumentInfoRecord`.

TC

Teamcenter

TCL

is a high-level, general-purpose, interpreted, dynamic programming language.

TEM

Teamcenter Environment Manager.

Transport Package

A file that contains functions that can be imported to SAP

U

URI

Unified Resource Identifier: a generalized from of a resource locator (URL) and resource name (URN), which just identifies a resource, but is not necessarily sufficient to locate (find) the resource. URIs are often used to identify configurations in Java and other languages. See **https://en.wikipedia.org/wiki/Uniform_Resource_Identifier** for more details.

URL

Unified Resource Locator: a string with a certain format, allowing to load a resource from a network. URLs are a specific form or URNs.

W

WBS

WBS is an abbreviation for a SAP Work Breakdown Structure.

X

XML

Extensible Markup Language is designed to store and transport data in a format that is both human- and machine-readable.

XRT

stands for XML Rendering Template, also known as XML Rendering Stylesheet. These are XML documents stored in datasets that define how parts of the Teamcenter user interface are rendered. They are used for the Rich Client as well as the Active Workspace.

Z

Z-Table

"Z" is the prefix name for custom tables well-known in SAP world.

Siemens Industry Software

Headquarters

Granite Park One
5800 Granite Parkway
Suite 600
Plano, TX 75024
USA
+1 972 987 3000

Americas

Granite Park One
5800 Granite Parkway
Suite 600
Plano, TX 75024
USA
+1 314 264 8499

Europe

Stephenson House
Sir William Siemens Square
Frimley, Camberley
Surrey, GU16 8QD
+44 (0) 1276 413200

Asia-Pacific

Suites 4301-4302, 43/F
AIA Kowloon Tower, Landmark East
100 How Ming Street
Kwun Tong, Kowloon
Hong Kong
+852 2230 3308

About Siemens PLM Software

Siemens PLM Software, a business unit of the Siemens Industry Automation Division, is a leading global provider of product lifecycle management (PLM) software and services with 7 million licensed seats and 71,000 customers worldwide. Headquartered in Plano, Texas, Siemens PLM Software works collaboratively with companies to deliver open solutions that help them turn more ideas into successful products. For more information on Siemens PLM Software products and services, visit www.siemens.com/plm.