Applies to:
SAP NetWeaver Process Integration (PI) in releases 7.00, 7.10, 7.11, 7.30 & above

Summary
This guide provides additional advice for using the Enhanced Change and Transport System (CTS+) to deal with PI Software logistics issues. The main scope is on the concept of “how to use” PI Software logistics and not on “how to configure” a transport landscape. This document relies on an existing CTS+ setup already finished successfully and does NOT cover any technical details how to configure CTS+ landscapes.

Author(s):  Andreas Böhme, Michael Graf
Company:  SAP Deutschland AG & Co. KG
Created on: 1 October 2011

Author(s) Bio

Andreas Böhme is a Senior Technology Consultant at SAP with main focus on SAP NW Process Integration (PI) and all SAP J2EE Technology- and Performance-related topics. He holds a degree in Commercial Information Technology (Technical University Ilmenau) and has been a Core member and Team lead within SAP’s NW PI Best Practices Community for many years.

Michael Graf is a Technology Consultant at SAP and focuses on SAP NetWeaver Process Integration, SAP NetWeaver Development Infrastructure and other Java-related SAP products. He holds a degree in Computer Science (University of Tübingen) and is experienced in software engineering and software development.
Table of Contents

(1) Overview and fundamentals: PI Transports in general 3
   Introduction .............................................................................................................................................. 3
   Transport strategies for PI Objects ........................................................................................................... 4
   Specific characteristics for PI Directory transports .................................................................................. 4

(2) PI Transports using Enhanced Change and Transport System (CTS+) ................................................. 6
   Scenario description .................................................................................................................................. 6
   Alternative object lists for transports ....................................................................................................... 7
   Reasons for NOT transporting change lists: ............................................................................................. 8
   Recommended PI roles and responsibilities ............................................................................................. 8

(3) Detailed step-by-step solution for PI Transports using CTS+ ............................................................. 11
   a) SLD Objects ....................................................................................................................................... 11
   b) Design Objects (Enterprise Service Repository) .................................................................................. 13
   c) Configuration Objects (Integration Directory) .................................................................................... 16

Related Content ........................................................................................................................................... 20

Copyright ..................................................................................................................................................... 21
(1) Overview and fundamentals: PI Transports in general

Introduction

For any SAP PI development activities we usually separate between different types of development objects:

1. SAP PI specific objects within the System Landscape Directory (SLD) = **SLD Objects**
   e.g. Technical Systems, Business Systems, Products and Software Component Versions (SWCVs)

2. SAP PI design objects within the Enterprise Service Repository (ESR) = **Repository Objects**
   e.g. SWCVs, Namespaces, Data types, Interfaces, Mappings, Integration Processes, etc.

3. SAP PI configuration objects within the Integration Directory (ID) = **Directory Objects**
   e.g. Configuration scenarios, Receiver & Interface Determinations, Communication channels, etc.

4. Development objects within several SAP application systems (e.g. SAP ERP, CRM, SCM,...) that implement business logic (out of PI scope) = **SAP Backend Objects**
   e.g. JAVA- or ABAP classes, Client- and Server-Proxies generated out of the Proxy Framework, ABAP programs, function modules etc.

Within this document we focus on the SAP PI Objects (1) to (3) only [SLD, Rep, Dir]. All other objects developed within SAP application systems [incl. ABAP Proxy objects] must be delivered within their own infrastructure. There may be dependencies between Repository objects and SAP Backend objects, in particular generated ABAP proxies, for which you will have to make sure they get synchronized during transportation. One possible approach – out of scope for SAP PI – would be SAP’s Quality Gate Management (QGM)
Transport strategies for PI Objects

PI relevant SLD Objects can be transported via

- Manual Export / Import via ZIP files using local Frontend-PC
  Selection of individual content from Source-SLD (DEV) and client-based transfer to Target-SLD (PRD) has to be done by developer / administrator → error-prone → Manually
- Enhanced Change and Transport System (CTS+)

PI Repository and Directory Objects can be transported via

- Manual Export / Import via TPZ files using Filesystem access of SAP PI server or local Frontend-PC (since SAP PI 7.10) → error-prone → Manually
  - /usr/sap/<DEV>/SYS/global/xi/repository_server/export
  - /usr/sap/<PRD>/SYS/global/xi/repository_server/import
- Change Management Service (CMS) component of SAP NWDI → JAVA transport mechanism
- Enhanced Change and Transport System (CTS+) → ABAP transport mechanism

**Note:** SAP recommends using CTS+ for all PI components (SLD, Enterprise Service Repository, Integration Directory) to make sure all Interface-related objects can be transported throughout a SAP PI landscape in a controlled and documented manner.

Specific characteristics for PI Directory transports

Independent from the chosen Transport strategy (TPZs, CMS, CTS+), there are the following specific characteristics for Directory transports

- Business System names (e.g. Sender / Receiver systems) will be mapped over automatically during the import into the next following SAP PI system according to their transport target defined in the SLD (predecessor <> successor relation)

  ![SAP ECC_1EE_200 Details](image)

- Connection parameters like Users, Passwords, Hostnames, URLs inside Communication Channel objects will be invalidated and replaced by empty fields during the initial import into the next system (and only this time!)
  - Additional manual effort required before change lists can be activated
  - This is no bug, but intended behavior!
Your SAP PI Transport strategy depends from your PI Transport landscape (2-, 3- or even 4-tier PI landscape) and the number of connected SAP Backend systems in each “stage” DEV, QAS, PRD

Some common “challenges” in SAP PI - SAP ERP interconnections we face often:

- Multi-Client-connection in Development or QA stage (e.g. DEV_100 & DEV_200), but only Single-Client-connection in Production (PRD_100)
- Automatic mapping of Business System names fails during import if not all of the included objects have defined a transport target before
- Possible solutions or workarounds:
  - General restriction on 1-client-connection for each stage → preferable approach
  - Using of “dummy” entries for Business Systems (without any technical connection behind) to map multiple DEV Business Systems to virtual ones in the next stage → not Best-practice, but possible workaround to enable continuous transports
  - Build different transports that go different routes (e.g. some transports stop in QA and never reach production) → methodologically flawed, NOT advisable due to complexity and data inconsistency
(2) PI Transports using Enhanced Change and Transport System (CTS+)

Scenario description
The main idea behind CTS+ is simple: Perform combined ABAP+JAVA transports in so-called “close coupling” scenarios and use it for all kind of SAP PI objects (since PI 7.00 SP14 / 7.10 SP06)

- Development objects will be exported from DEV system via regular tools (SLD/Rep/Dir) and attached to an existing / newly created ABAP transport request
- Transport Organizer Web UI on DEV system can be used to create, manage & release transport requests (alternative: “classical” TMS inside ABAP can be used as well for release)
- During the import into the target system (QAS, PRD), an automatic deployment method using the CTS Deploy Web-Service will be triggered
- Deployment service knows the Import methods SDM, XI, SLD, File

The tools used for SAP PI transports are:

- Transport Management System (ABAP TMS) on every involved PI system
- Transport Organizer Web UI on the development SAP PI System
- CTS Deploy Web-Service on each delivery system (QAS, PRD)
- System Landscape Directory (SLD), to export SLD-Content from DEV system
- Enterprise Services Builder (ESR), to export Design-Content from DEV system and to track performed transports later
- Integration Builder for Integration Directory transports (ID), to export Configuration-Content from DEV system and to track performed transports later

The relevant objects for SAP PI transports are usually:

- SLD objects like Technical Systems (3rd party), Business Systems, Products and SWCVs
- Design objects from ES Repository (incl. all kind of Mappings)
Configuration objects from Integration Directory
Possibly ABAP-Mappings (but NOT ABAP Proxies!)

**Alternative object lists for transports**

A change list enables a collective release of changes to various objects in the ES Repository or Integration Directory. With regard to transports, it stands to reason that those objects that have been changed are exactly those objects that you want to transport. Transporting on the change list level therefore simplifies the selection of changed objects during development – as long as the number of change lists is still manageable!

Alternatively, you can group transport lists containing object sets that are independent of a change list together in the transport wizard.

**Change Lists**

- The Integration Builder tools create a standard change list automatically as soon as an object is saved or deleted by a user who has not yet been assigned a change list
- Once activated, an open change list has the status transportable → can be transferred to CTS
- Exports the changes contained in the change list [only]. This corresponds to the **object versions** at the time of release of the change list

**Transport Lists**

- Manual selection of objects can be done in the Integration Builder tools independent from change lists (“Right mouse click > Export”)
- Will be transferred to CTS and waits in status *Waiting for Export* as long as this operation takes to finish (asynchronous)
- Exports the **object versions active** at the time of compilation of transport list (may be a later version than your change list!)

The following figure provides an overview of the different statuses change lists and transport lists can have:
**Note:** Use exclusively Transport Lists and try to avoid transporting Change Lists due to complexity reasons!

The PI developer himself should be responsible for the compilation of “his” objects AFTER his development activities have been finished (This is a big difference compared to classical ABAP transports!)

**Reasons for NOT transporting change lists:**

a) **Complexity**

During regular PI development activities there will usually be created hundreds of change list, too many to deal with each single change.

b) **Versioning**

You can make new changes to objects in transportable change lists before the transport took place. The result of this is that different versions of an object can exist in different transportable change lists. The following figure provides an example:

![Object Versions Diagram](image)

The transportable change list CL1 contains three objects. At the time CL1 was released, both objects Object2 and Object3 had version 1 and Object1 had version 2. After this, the objects Object2 and Object3 were changed again using other change lists and have now newer version 3. You now have the following options:

- To transport the object versions in change list CL1, you could use the *Release for CTS Transport* function on the Change Lists tab page (blue line) → usually not reasonable to transport elder versions!
- To transport the current object versions, create a transport list manually by using the transport wizard → recommended to have control and clarification which version exactly should be transported → most latest versions contained in transport list (green line)

**Recommended PI roles and responsibilities**

For having a proper SAP PI Software Logistics concept in place we recommend setting up several roles and assigning tasks inside the SAP PI project team. A detailed description of how to establish a special organizational unit called “Integration Competency Center (ICC)” can be found on SDN (see related documents).
(1) Role **SAP BASIS / PI System Administrator**
- Creates **one single ABAP Transport Request** (Type Workbench) per PI Interface for each of:
  - SLD Objects
  - Design Objects (ESR)
  - Configuration Objects (ID)
- That usually makes 3 different Transport Requests for each SAP PI interface-pair / scenario
- Imports the released transport requests later into the next system after their approval

(2) Role **SAP PI Transport Manager / Technical Project Lead**
- Uses the given ABAP Transport Request, creates several Transport Tasks below and assigns all PI Developers to individual sub tasks (ABAP TMS Function Add User)
- Checks and releases the overall Transport Request (with all developer Tasks inside) after development has been finished and transport lists attached
- Adopts the imported Directory Objects (Dir) after import into next systems [according to Post-Transport documentation] and activates the change list(s)

(3) Role **SAP PI Developer**
- Creates all his development objects and builds Transport Lists (Export) containing all objects relevant for the certain PI scenario
- Uses the assigned ABAP Transport Task (inside the Transport Request) and attaches his compiled Transport list (individual objects) → Export will be only out of DEV system, the same objects pass up to PRD!
- Creates basic documentation for Post-Transport configuration of Directory objects, mainly Communication channels ("What parameters have to be modified after import"?)

Picture of an example workflow (proposal)
Possible Troubleshooting activities may be:

- Activation of Transport List 'XI 7.1 Import' after the Directory import fails due to in-complete Post-Transport configuration
- Check for existing version conflicts

* Explicit release of single tasks not required in CTS+
(3) Detailed step-by-step solution for PI Transports using CTS+

a) SLD Objects

The need for transporting SLD Objects heavily depends on the current SLD strategy in place:

- In case of one single SLD for multiple SAP PI systems there are NO transports needed at all.
- In case of multiple SLDs (local for each SAP PI system, Distributed Production / Non-Production) you might have to have setup different transport strategies for SLD Objects.

In the following description we assume a distributed SLD topology with one central “Master-SLD” consolidating and distributing all SAP Technical Systems and multiple “Runtime-SLDS” each locally directly on its own SAP PI system. In this case, the local SLDs must be supplied via CTS+ transports.

Step-by-Step description for transports in this SLD scenario:

- All self-registered SLD Objects (Technical Systems of Type SAP AS ABAP / JAVA) will be collected centrally and distributed automatically by the SLD concept → no need for transports here
- SAP PI-specific SLD Objects (Technical Systems 3rd party, Business Systems, Products and SWCVs) will be maintained inside DEV-SLD (even for PRD-systems!) by the PI Developer and transferred manually (selective) into CTS+ Transport Requests
- That means the PI Developer is responsible for “his“ Business Systems (DEV, QAS, PRD) and SWCVs
• The DEV-SLD already knows all landscape data incl. PRD Business Systems and the transport groups between!

• Requirement: The SLD Transport interconnections between Business Systems (transport targets) have to be maintained [by the PI Transport Manager] in the following matter:
  o DEV_BS1 → QAS_BS1
  o DEV_BS1 → PRD_BS1

• This is due to the fact, that the original objects from the DEV export stay the same unchanged throughout the complete Software logistics cycle

• The compiled CTS+ Transport with its content [Business Systems, Products & SWCVs] will be imported after its release into the Target-SLD

Recommended sequence for transporting SLD objects:
  1. Technical Systems (if not already there via SLD Bridge), e.g. Systems of type 3rd Party
  2. Products and underlying Software Components Versions (SWCVs)
  3. Business Systems of Sender / Receiver in interface scenario

[should be consolidated into one single Transport Order]
b) Design Objects (Enterprise Service Repository)

Idea: All SAP PI related development activities will [only] be performed on PI DEV system and transported via a CTS+ Transport Request tool-based directly out of Enterprise Service Repository (ESR)

No direct changes to any Design objects on QAS- and PRD- system are allowed!

Prerequisites:
SAP Basis creates **one** ABAP Transport Request per SAP PI interface pair / scenario and SAP PI Transport Manager assigns his staff members in terms of Transport Tasks

[\` to be more precise: one TR for Design objects (ESR) & one TR for configuration objects (ID)]

- Creation of Transport Request and tasks can be done via “normal” CTS in ABAP and/or Web UI (ABAP TMS Function Add User) → SE09/10

Check and flag for “Standard Request” should be done via Transport Organizer Web UI

- The “Standard Request” is the one shown as proposal in the ESR Transport Wizard

How to bundle Design objects:

a) For the initial transport to the next system stage, the set of objects depends on the chosen component structure

- Usually we recommend to select a single **Namespace** with all objects below

- Alternative: complete **Software Component Version (SWCV)** may be transported if the amount of objects inside is still manageable
If the customer follows SAP’s **Design recommendations** and **Naming conventions**, several Namespaces that belong together logically for one certain scenario, can be distributed over 3 different SWCVs → customer example:

SWCV “DTO_I_LEGACY_SHIPMENT 1.0” Namespace urn:mycompany.com:DTO_SHIPMENT:CustomsData (Sender)
SWCV “DTO_I_SAPECC_RECEIVE 1.0” Namespace urn:mycompany.com:DTO_SHIPMENT:CustomsData (Receiver)
SWCV “DTO_A_SAPPI 1.0” Namespace urn:mycompany.com:DTO_SHIPMENT:CustomsData:Global (PI Generic)

In this case, the PI developer has to select all objects belonging to “his” interface (Namespaces from 3 different SWCVs) and attaches this 3 Transport Lists to his assigned Transport Request.

Please wait for the [asynchronous] system message inside ESR before releasing any request!
b) For any corrections / later transports the set of objects is usually smaller and can be done via single selection

- The PI Developer selects the changed objects and attaches this Transport List to a given Request
- Sequence problems with wrong transport order in case of multiple changes to the same object will be prevented automatically by internal versioning

Design recommendation:

- Try to use Integration Scenarios inside ESR for modeling because it will be easier to transfer it to Configuration Scenarios later on in Configuration (ID), where we need scenarios for a proper transport strategy
c) Configuration Objects (Integration Directory)

Idea: All SAP PI related development activities will [only] be performed on PI DEV system and transported via a CTS+ Transport Request tool-based directly out of Integration Directory (ID)

No direct changes to any Configuration objects on QAS- and PRD-system are allowed -- with exception of Post-Transport activities for Communication channels (connection parameters)!

Prerequisites:
SAP Basis creates one ABAP Transport Request per SAP PI interface pair / scenario and SAP PI Transport Manager assigns his staff members in terms of Transport Tasks

[`same as above for the Design objects out of Repository]

How to bundle Configuration objects:

a) For the initial transport the set of objects should be bundled by a Configuration Scenario

- Usually we recommend to select a complete Configuration Scenario with all objects below
- PI Developer has to maintain Name and description of his Transport List according to Naming conventions (see chapter below)
- The feature of Folders as additional order criteria introduced with PI 7.10, & changed with PI 7.11 can be used in addition, but just for better visibility → optional
- The object assignment to an Configuration Scenario must be used → mandatory
b) For any corrections / later transports the set of objects is usually smaller and can be done via single selection
   - The PI Developer selects the changed objects and attaches this Transport List to a given Request
   - Sequence problems with wrong transport order in case of multiple changes to the same object will be prevented automatically by internal versioning

Configuration characteristics:
   - After successful import to the target system a new Change List will be created automatically in the Target-Directory with the default name ‘XI 7.1 Import’ and the owner NWDL_CTSADM (or corresponding Communication user)
   - This Change list must be transferred to your own User-ID & Post-transport configuration must be performed for all Communication channels and the Change list must be activated before a new import of the same Configuration object can take place
   - Otherwise you’ll get an error message at the next deployment similar to “[Object xyz] (DirContext) since object is already in your change list ‘XI 7.1 Import’”

Important restriction:

One Transport Request must NOT contain one and the same object multiple times
   - This is esp. essential for Business Systems that might be reused in different scenarios
   - If you transport larger selections inside one transport request (not recommended!), you’ll get the same error message to “[Business System XYZ] (DirContext) since object is already in your change list ‘XI 7.1 Import’”
     - The Business System name mapping (DEV → QAS) works for the first time
     - The 2nd mapping [inside the same import] will fail because the 1st change is still not active – as it is recorded within the same Change List
Note: SAP Notes 1547414 & 1548233 deliver an useful change that after import, only one changelist will be created in Integration Directory, even if the transport request contains more than one changelist. This is to avoid the failure of import into directory when an object has undergone multiple modifications and more than one changelists are attached to the same transport request. All imported versions of the object will be added to the single import changelist and while adding any object, if there is an open version already present in the changelist, it will be replaced with the new open version.

Note: For Directory objects the transport sequence is crucial! There is the danger of overwriting newer version of an object with an elder one, if the import sequence differs from the original export. For directory objects, the Version-ID of an object will be ignored while importing.
**Naming conventions for PI Transports**

Each CTS+ Transport Request should be named after an agreed convention starting with a fixed prefix followed by an [optional] Interface-ID and a free text description. The respective convention can be defined Customer- and Project-specific, but should make sure that CTS+ transports can be identified easily by there description. One possible approach would be for example:

<table>
<thead>
<tr>
<th>PI_&lt;ObjectType&gt;_&lt;InterfaceID&gt;: &lt;short text&gt;</th>
</tr>
</thead>
</table>

- **ObjectType** [SLD, REP, DIR]
- **InterfaceID** Numeric value as unique identifier → usually defined earlier in Blueprint phase
- **Short text** optional: free text with additional description what objects are inside

Valid example Transport Requests:
- SIDK900045  PI_SLD_S702: SWCVs_TS3rdParty_BS
- SIDK900049  PI_DIR_S701: ConfigScenario AussenhandelsdatenVerteilen

Besides this, each Transport List for the export from Integration Directory (ID) should have a unique description as well for easier identification. Esp. the responsible developer should be stated clearly by adding the User-ID. One possible approach would be for example:

| Name = PI_DIR_<InterfaceID>_<Username>: <short text> |
| Description = <Interface description> |

![Export Configuration Objects](image)
Related Content
Please see the following additional SAP PI Guidelines regarding Design and Naming recommendations.

SAP PI Design recommendations

SAP PI Naming conventions

SAP recommends using CTS+ for all PI components (SLD, Enterprise Service Repository, Integration Directory) to make sure all Interface-related objects can be transported throughout a SAP PI landscape in a controlled and documented manner. For details regarding the CTS+ configuration itself, please check:

How to configure CTS+ for PI 7.1 >= SP6

Best Practices for implementing CTS+

Integration Competency Centers (ICC) are an organizational unit within an enterprise, aiming at the reduction of IT costs and the establishment of an adaptive enterprise allowing to react fast in a rapidly changing IT landscape. This article shows up a methodology for establishing an ICC in an SAP environment, with a special focus on SAP integration platforms like SAP NetWeaver Process Integration (PI) or SAP BusinessObjects Data Services

How to Define and Establish an SAP Integration Competency Center (ICC)