

# **ADM325**

**Software Logistics for SAP  
S/4HANA and SAP Business Suite**

**PARTICIPANT HANDBOOK  
INSTRUCTOR-LED TRAINING**

Course Version: 24  
Course Duration: 5 Days

# SAP Copyrights, Trademarks and Disclaimers

© 2025 SAP SE or an SAP affiliate company. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP SE or an SAP affiliate company.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE (or an SAP affiliate company) in Germany and other countries. Please see <https://www.sap.com/corporate/en/legal/copyright.html> for additional trademark information and notices.

Some software products marketed by SAP SE and its distributors contain proprietary software components of other software vendors.

National product specifications may vary.

These materials may have been machine translated and may contain grammatical errors or inaccuracies.

These materials are provided by SAP SE or an SAP affiliate company for informational purposes only, without representation or warranty of any kind, and SAP SE or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP SE or SAP affiliate company products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

In particular, SAP SE or its affiliated companies have no obligation to pursue any course of business outlined in this document or any related presentation, or to develop or release any functionality mentioned therein. This document, or any related presentation, and SAP SE's or its affiliated companies' strategy and possible future developments, products, and/or platform directions and functionality are all subject to change and may be changed by SAP SE or its affiliated companies at any time for any reason without notice. The information in this document is not a commitment, promise, or legal obligation to deliver any material, code, or functionality. All forward-looking statements are subject to various risks and uncertainties that could cause actual results to differ materially from expectations. Readers are cautioned not to place undue reliance on these forward-looking statements, which speak only as of their dates, and they should not be relied upon in making purchasing decisions.

# Typographic Conventions

American English is the standard used in this handbook.

The following typographic conventions are also used.

This information is displayed in the instructor's presentation



Demonstration



Procedure



Warning or Caution



Hint



Related or Additional Information



Facilitated Discussion



User interface control

*Example text*

Window title

*Example text*



# Contents

vii	<b>Course Overview</b>
1	<b>Unit 1: Introduction to SAP Software Logistics</b>
3	Lesson: SAP System Landscape
9	Lesson: Client Concept
17	Lesson: Describing System Change Options and Client Change Options
27	<b>Unit 2: SAP System Landscape Set Up</b>
29	Lesson: Setting Up the Transport Management System (TMS)
51	Lesson: Configuring Extended Transport Control
65	<b>Unit 3: Creating and Exporting Transport Requests</b>
67	Lesson: Customizing and Customizing Projects
77	Lesson: Transport Requests for Customizing
85	Lesson: Customizing Procedure
95	Lesson: Transport Requests for Development
107	Lesson: Customer Development
115	Lesson: Modifying SAP Objects
131	<b>Unit 4: Importing Transport Requests</b>
133	Lesson: The Transport Process
139	Lesson: Imports Using TMS
155	Lesson: QA Approval Procedure and Transport Proposals
165	Lesson: Import Process
181	Lesson: Monitoring Tools
195	Lesson: Cleaning up the Transport Directory
209	<b>Unit 5: Software Change Strategies</b>
211	Lesson: System Landscape Options
227	<b>Unit 6: Client Tools</b>
229	Lesson: Client Copy and Client Transport Tools
259	Lesson: Client Compare and Client Maintenance Tools
273	<b>Unit 7: SAP Note Assistant, SAP Support Packages, SAP System Upgrades, SAP S/4HANA Conversion</b>
275	Lesson: SAP Note Assistant
293	Lesson: SAP Support Packages
309	Lesson: SAP System Upgrade and SAP S/4HANA Conversion

**327      Unit 8:     Appendix: Enhancements to the Change and Transport System**

- |     |  |
|-----|--|
| 329 | Lesson: Enhanced CTS - The Basic Idea  |
| 337 | Lesson: Configuration of the Change and Transport System for the enhanced CTS                          |
| 355 | Lesson: Transport of non-ABAP Objects  |
| 369 | Lesson: Outlook: Integration of ABAP Development Into CI / CD (DevOps) Processes with the Help of gCTS |

**384      Glossary**

# Course Overview

## TARGET AUDIENCE

This course is intended for the following audiences:

- Technology Consultant
- System Administrator



# UNIT 1

# Introduction to SAP Software Logistics

## Lesson 1

SAP System Landscape

3

## Lesson 2

Client Concept

9

## Lesson 3

Describing System Change Options and Client Change Options

17

## UNIT OBJECTIVES

- Describe the difference between an application and an SAP system
- Explain the need for a multi-system landscape
- Describe the data structure of an AS ABAP based SAP system
- List different client roles and explain their use
- Illustrate an example of a multi-system landscape
- Use system change options and client change options for your transport landscape
- Outline the idea of Customizing using current settings



# Unit 1

## Lesson 1

# SAP System Landscape

## LESSON OVERVIEW

In this lesson, you will learn about the SAP product portfolio, SAP Business Suite, SAP NetWeaver, SAP S/4HANA, SAP systems, and instances. You will also identify the components of the SAP system landscape that SAP recommends.

## Business Example

As an SAP system administrator, you need to understand the roles of different SAP systems in the SAP system landscape for on-premise systems before deciding how to configure the Transport Management System.

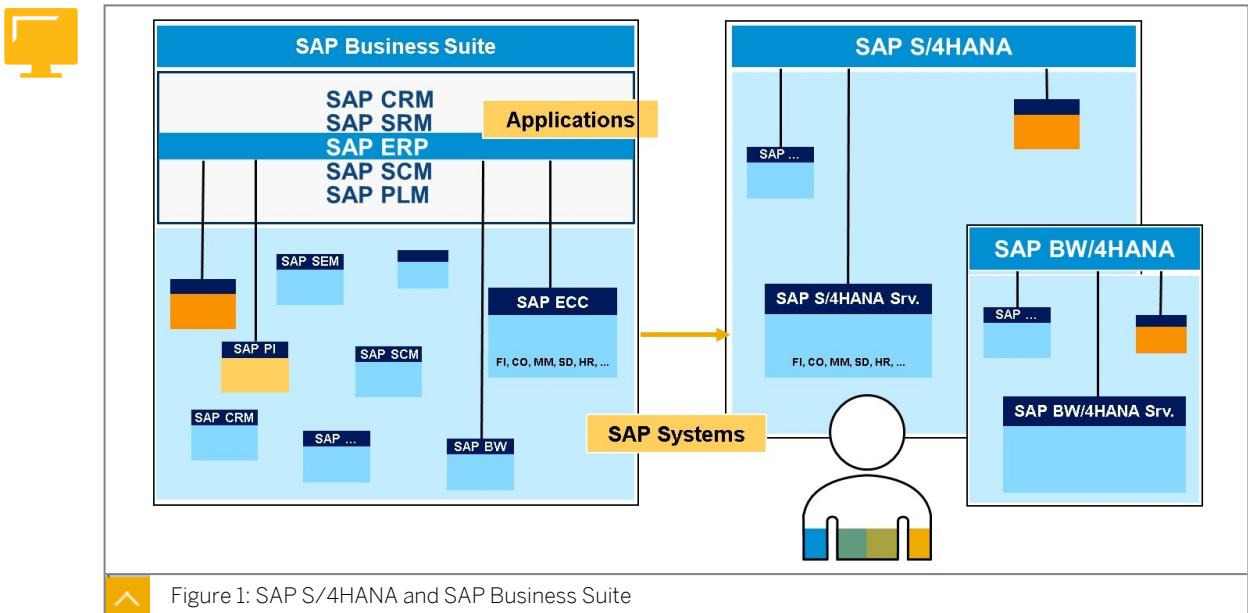


## LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Describe the difference between an application and an SAP system
- Explain the need for a multi-system landscape

## SAP System Components



In the context of *SAP Business Suite*, the SAP Product Portfolio offers SAP applications for the following areas:

- Enterprise Resource Planning (SAP ERP)
- Customer Relationship Management (SAP CRM)

- Supplier Relationship Management (SAP SRM)
- Supply Chain Management (SAP SCM)
- Product Life Cycle Management (SAP PLM)

These applications are part of the *SAP Business Suite*. As a result, a customer with an SAP Business Suite license also has a license for each application in the SAP Business Suite.



Note:  
Sometimes the terms “SAP solution” and “SAP application” are used interchangeably.

An SAP application is a virtual bundle of business functions. You cannot install, patch, upgrade, or administrate an SAP application. You cannot log onto an SAP application because an SAP application consists of several SAP components, such as SAP ECC (SAP ERP Central Component), SAP SCM server, and so on. From a technology point of view, it is the SAP components that a customer installs or upgrades. You can log on and work with a component. A common term for an SAP component is an “SAP system”.

SAP ECC is the central component of *SAP ERP*, which in turn is the central application of the SAP Business Suite.

However, SAP now offers *SAP S/4HANA Suite*, a new generation of business applications built on the in-memory platform SAP HANA. SAP S/4HANA Server is a new SAP system in the new SAP S/4HANA Suite. You could say that SAP S/4HANA Server is “the new SAP ECC system”.



Note:  
It is possible to convert your existing SAP ECC system to an SAP S/4HANA Server system, by performing an SAP S/4HANA conversion.

Another important product to mention is *SAP NetWeaver*, which represents the technology stack provided by SAP to support the SAP Business Suite. Again, you cannot install SAP NetWeaver, or patch or update SAP NetWeaver as a whole. SAP NetWeaver is a bundle of several SAP systems, for example:

- SAP Business Warehouse (SAP BW)
- SAP Enterprise Portal (SAP EP)
- SAP Process Integration (SAP PI)
- SAP Master Data Management (SAP MDM)

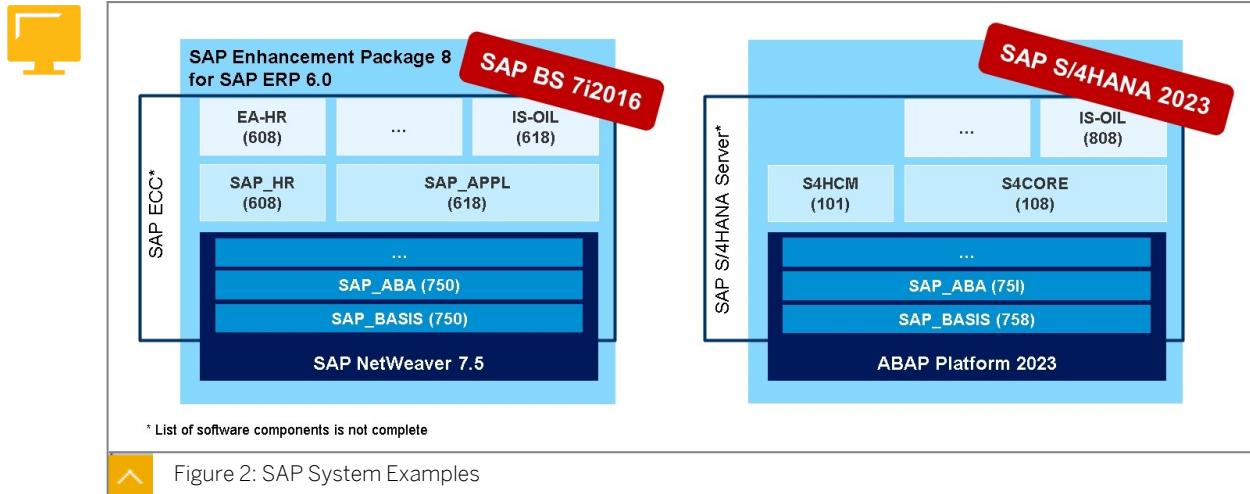
These SAP systems can be installed as needed to serve different requirements.

SAP systems for SAP Business Suite are based on SAP NetWeaver Application Server (SAP NetWeaver AS) – either on ABAP (SAP NetWeaver AS ABAP – or for short: AS ABAP) or on JAVA (SAP NetWeaver AS Java – or for short: AS JAVA).

*ABAP Platform* is the technology platform underlying SAP S/4HANA and is shipped as part of SAP S/4HANA.

In contrast to SAP NetWeaver AS ABAP, ABAP Platform is not shipped as a standalone product. ABAP Platform consists of the AS ABAP, the ABAP Development Tools (ADT) in

Eclipse and several additional technology components which are required to run SAP S/4HANA.

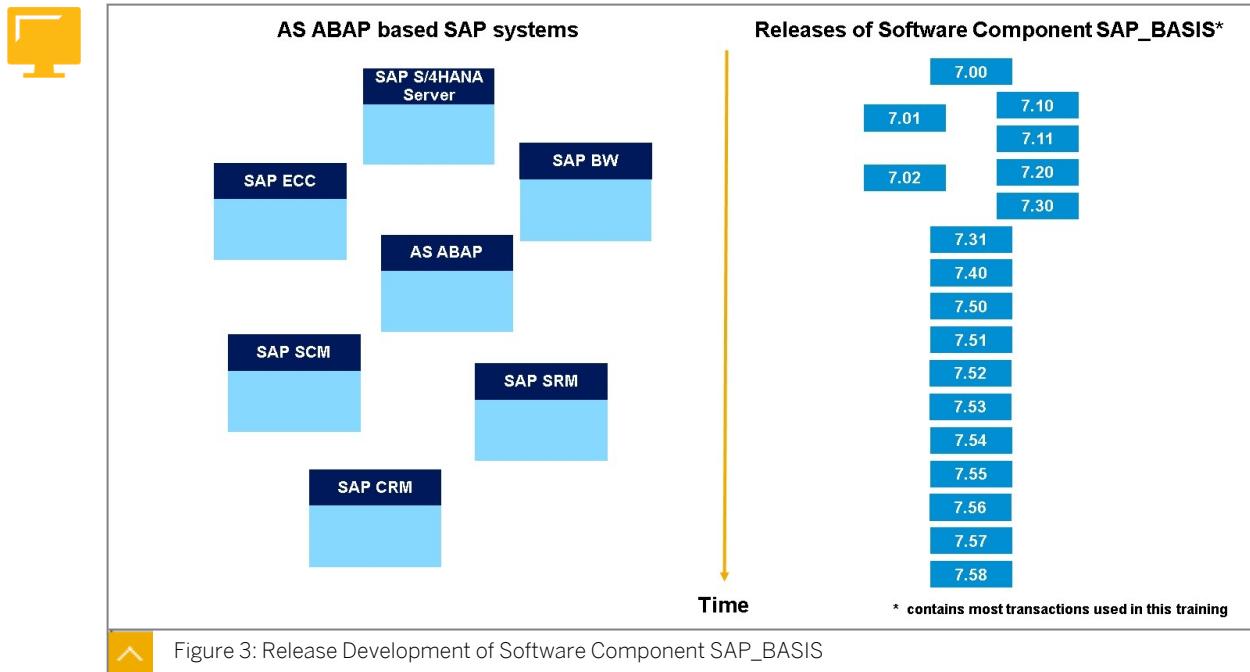


The figure above shows two examples of SAP systems:

- An SAP ECC 6.08 system (as part of *SAP Business Suite 7 Innovations 2016 (BS7i2016)*)
- An SAP S/4HANA 2021 Server system

In this training course, SAP S/4HANA 2021 Server systems are used.

## AS ABAP Based SAP Systems



This course describes how to set up and transport within an SAP system landscape of AS ABAP based SAP systems.

The figure above shows a history of some of the releases of the central software component SAP\_BASIS. Among others, this software component contains the coding for the transactions that form the Transport Management System.

## SAP System Landscape Recommendations



### The SAP system landscape should enable you to:

Protect application data using:

- Client concept and client roles
- Authorization concepts

Perform development, quality assurance, and production activities in separate environments to ensure:

- Consistency of repository objects
- A secure and stable production environment

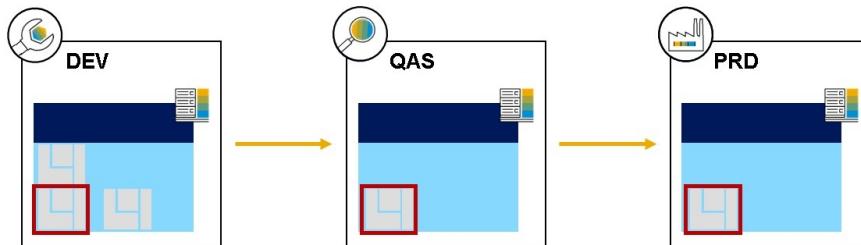


Figure 4: Aims of SAP the System Landscape

To maintain data security, you should protect application data by using the client concept, which separates data according to clients, and the authorization concept, which separates data within one client according to the users.

In addition, you should strictly separate the environments for development, quality assurance, and production. The stability and performance of the production client should not be affected by development and test activities in other clients.

Changes to repository objects are cross-client and immediately affect the business functions and processes. Therefore, changes have to be tested before being transported to the production system.

If you had only one SAP system, no further development would be possible after production had started. Changes to repository objects could only be made when production operations were stopped for development and testing.

A two-system landscape is not optimal, because both development and quality assurance testing take place in the development system.

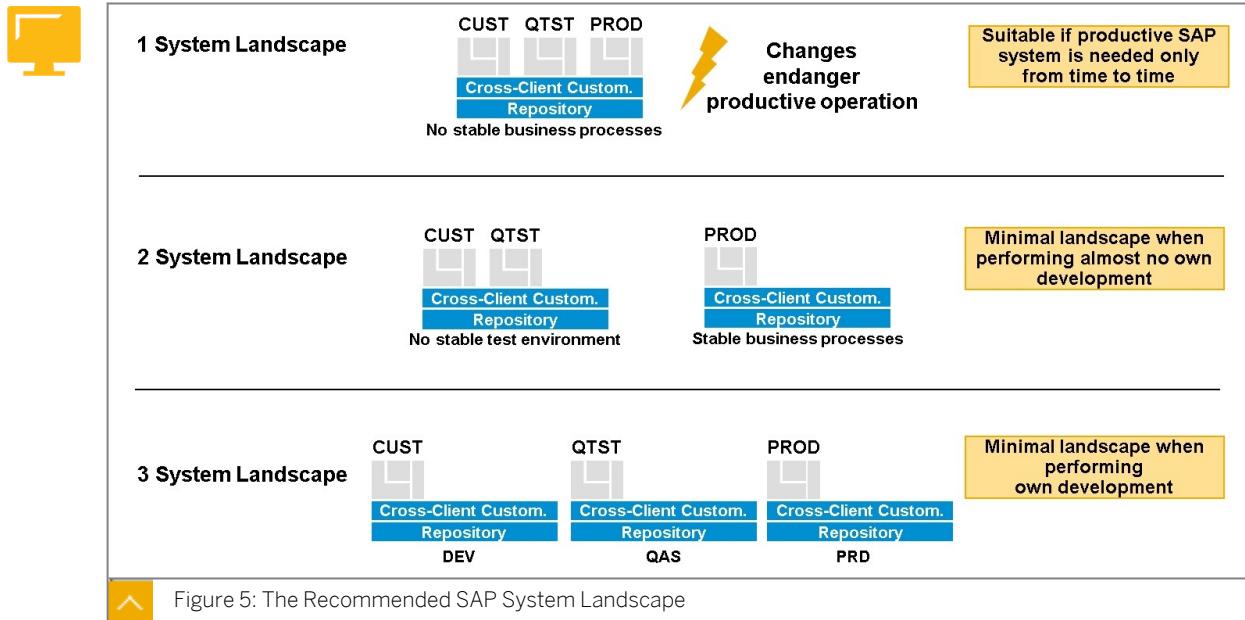


Figure 5: The Recommended SAP System Landscape

To ensure smooth production operation, an SAP system landscape should contain the three SAP systems from the figure above, the standard clients, and any required additional clients.

A three-system landscape meets the needs of many customers. It has the following features:

- Development takes place in the development system. Changed objects are then released to the quality assurance system.
- In the quality assurance system, you can test new developments without affecting the production environment.



Note:

A quality assurance system enables you to integrate development efforts and to check the validity and consistency of transported objects before moving the objects into production.

- All objects imported into the quality assurance system should also be delivered to one or more production systems.

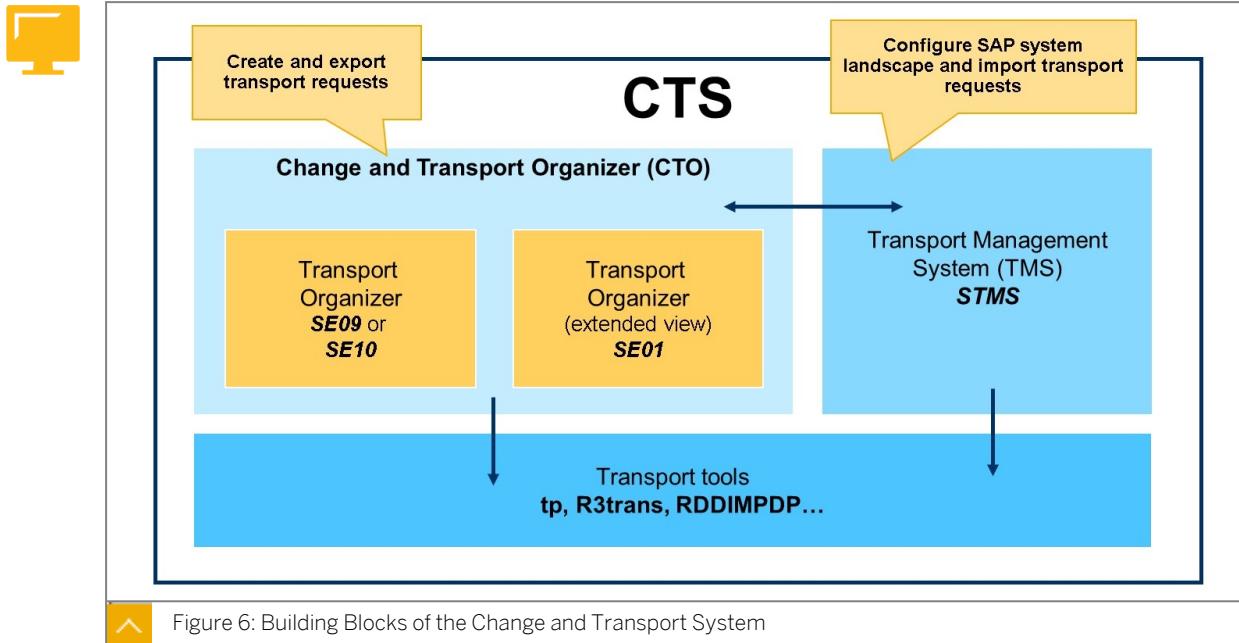
The three-system landscape is also a basis to apply SAP Support Packages or to perform a release upgrade.

SAP systems in the same SAP system landscape require unique SAP system names, known as the *System ID* (SID).

SAP provides the tools required for the creation, documentation, and distribution of changes within an SAP system landscape. It also provides the tools to set up the SAP system landscape so that it supports the management, verification, and testing of all changes:

- A single client is recommended for all customizing work. Set the *Client change options* appropriately before using your clients.
- A single SAP system (the development system) is recommended for all development work. Set the *system change options* appropriately.

- Assign appropriate user authorizations to developers and the customizing team members to control whether a user can create, modify, or release tasks and transport requests.
- Record and document all changes to a client or SAP system.



The *Change and Transport System (CTS)* comprises the following tools:

- The *Change and Transport Organizer (CTO)* provides functions for organizing software development projects. It is designed to support projects of all sizes, whether they are carried out centrally or in a distributed environment.
- The *Transport Management System (TMS)* organizes, monitors, and performs transports for all SAP systems within an SAP system landscape. In addition, TMS is used to configure and manage the setting up of SAP systems and transport routes within an SAP system landscape.
- The *transport tools* are executables and programs that communicate with the SAP system, the database, and files generated during the transport process.



## LESSON SUMMARY

You should now be able to:

- Describe the difference between an application and an SAP system
- Explain the need for a multi-system landscape

## Client Concept

### LESSON OVERVIEW

In this lesson, you will learn how to identify client roles in the SAP system. In addition, you will learn about the concepts of change management within an SAP system environment.

### Business Example

As an SAP system administrator or project team leader, you should monitor software configuration changes and program code changes to protect your production system from incorrect configuration settings or program bugs. You therefore need to know which types of changes can occur in AS ABAP-based SAP system landscapes. In addition, you need to understand the client concept for AS ABAP based SAP systems.



### LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Describe the data structure of an AS ABAP based SAP system
- List different client roles and explain their use
- Illustrate an example of a multi-system landscape

### Data in an SAP System

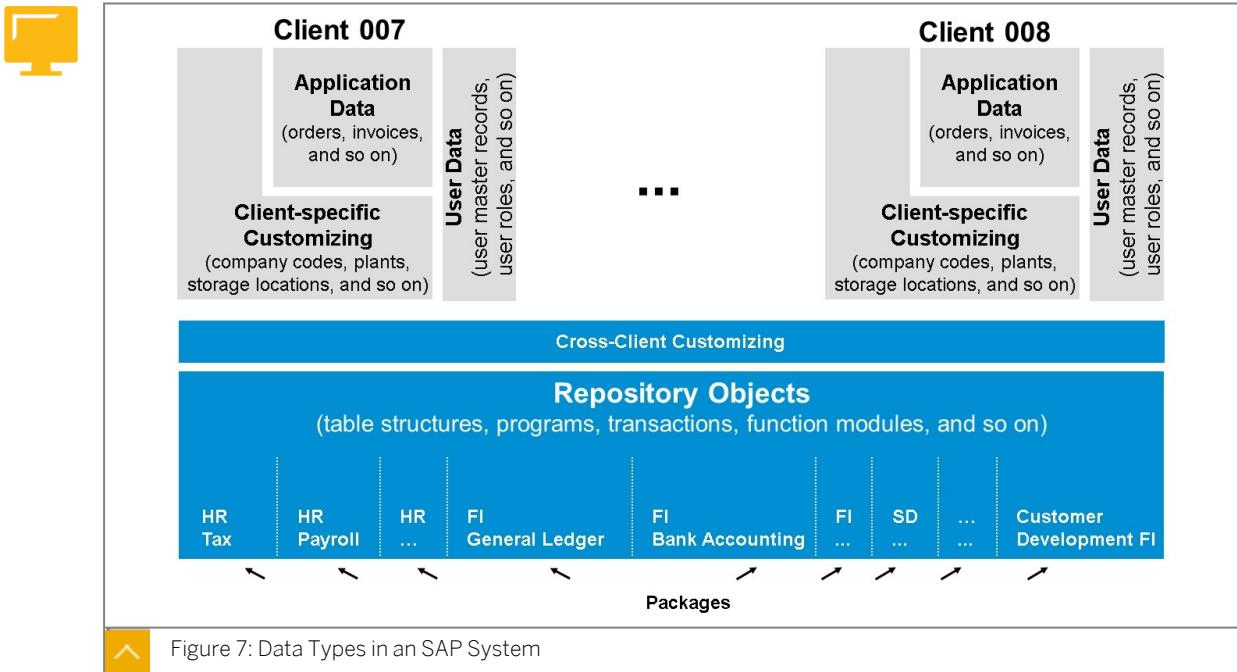
Data in an SAP system can be divided into two categories:

**1. Client-specific data:**

Application, customizing, and user master data, which only affect one client.

**2. Cross-client data:**

Customizing data and all repository objects, which affect the whole SAP system environment.

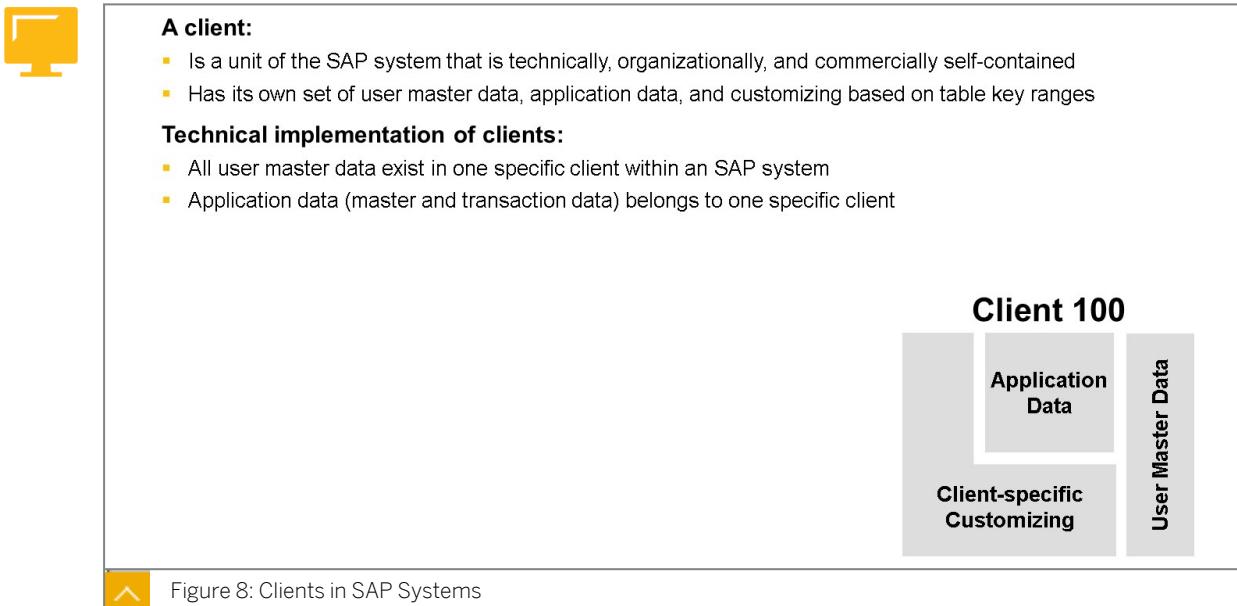


**Note:**

The ABAP dictionary is a data dictionary that is part of the ABAP repository.

## Clients in an SAP System

A client is a self-contained unit in commercial, organizational, and technical terms. It has its own user master data and set of table key ranges.



The data of different clients is separated both in the database and at the kernel level: Open SQL statements executed by an SAP application use the client number in the WHERE clause. A

table may contain data from several different clients, however the WHERE clause limits access to particular clients.

Client-specific data types are as follows:

- User master data, such as parameters and user groups.
- Customizing data, such as organizational units, assignments, and document types.
- Application data, such as business transaction data and material master data.

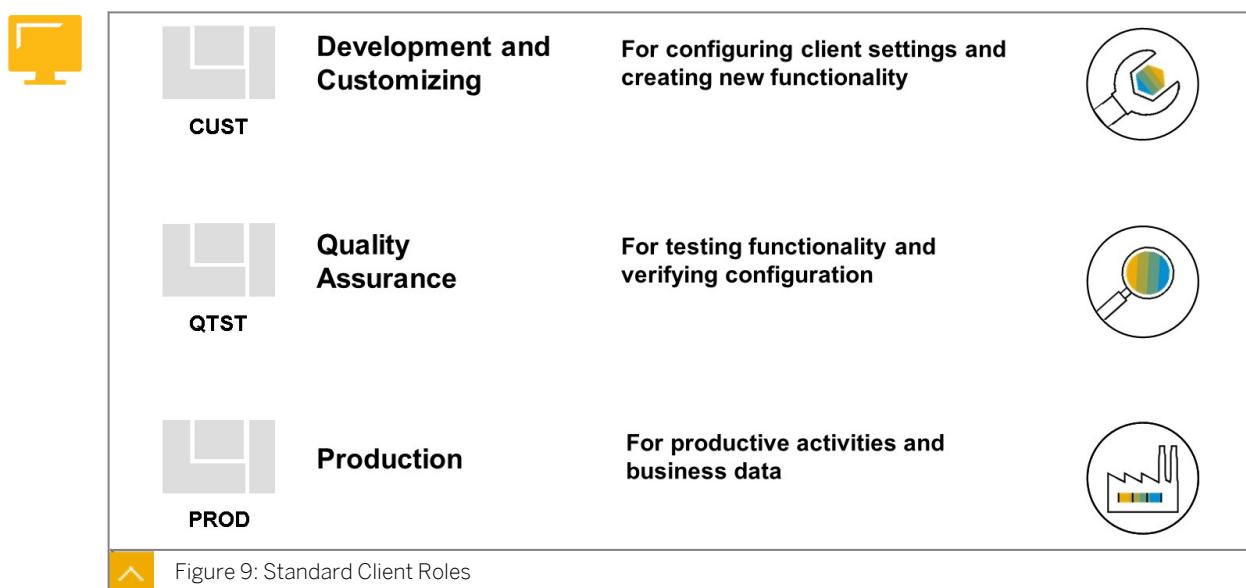
The SAP client concept can integrate several companies or subsidiaries in a single client. It does so using the following methods:

- Company Codes

Company codes define the smallest corporate organizational units, for which a complete self-contained set of accounts can be drawn up for external reporting.

- SAP Authorization Concept

The SAP authorization concept enables the parent company to access all subsidiaries for report purposes, while subsidiary-specific data is protected against access from other subsidiaries through company code definition.



Standard client roles fulfill the optimal minimum requirements of your SAP system.

The standard clients are as follows:

- Client **CUST: Development and Customizing**

The central customizing client, where complete adaptation of the SAP system to customer-specific needs takes place. All changes performed in this client are recorded so that they can be supplied to other clients using the Transport Management System (TMS).

- Client **QTST: Quality Assurance**

Used to test and verify the new customizing settings in the application.

- Client **PROD: Production**

The client for production activities, that is, where your company's business is carried out. Customizing changes that are imported into this client have to first be tested carefully in the QTST client. This ensures that production operation is free of disruption.

To realize the full benefits of a multi-system landscape, every critical client should be located in a separate SAP system.



**Note:**

The (logical) roles shown here are not identical to the (technical) client roles that can be chosen in transaction SCC4 (*Client Maintenance*). For example, the (logical) role QTST corresponds to the (technical) role Test. The (technical) client roles that can be selected in transaction SCC4 are *Customizing, Test, Production, Demo, Training/Education and SAP Reference*.



 <b>PREP</b>	<b>Pre-Production</b>	<b>For final integration test (release testing) as well as performance and volume test</b>	
 <b>TRNM</b>	<b>Training Master</b>	<b>For building all training courses and refreshing training execution client</b>	
 <b>TRNG</b>	<b>Training Execution</b>	<b>For end-user training</b>	

Figure 10: Additional Client Roles (1/2)

Additional clients within an SAP system landscape may include:

- Client *PREP: Pre-Production*

Client for the final integration and validation test, located on a separate pre-production system: The pre-production system is the environment for the *final integration test* once the scope of the release has been fixed (*release test*), for *regression test*, *technical system test* including *performance test* and *volume test* and *user acceptance test*. Furthermore, production support changes are tested here (particularly while the QTST client is being used for *integration testing* of a new development release).



**Note:**

The pre-production environment typically is owned by the production support organization.

**Caution:**

The pre-production system must not be “corrupted” with new release functionality or new Support Packages until the appropriate time (ideally as close to Go Live as possible).

All pre-production systems should be regularly refreshed from the production system, if possible.

The overall setup of all pre-production systems should mimic the production as closely as possible, including hardware-architecture, to ensure that the testing performed is valid. In detail this means:

- Same high-availability and fail-over solutions
- Same hardware architecture, processor types and storage subsystem
- Same disk size and data volume (due to system copy)
- The number of additional application servers can be reduced and possibly enhanced on demand by means of virtualization (only needed when a full and exact simulation of the productive workload is possible).
- Client *TRNM: Training Master*

Client to build training sources and to act as source client for the client copy of the *training execution* client.

In this client, exercises and demos are prepared including sample data.

- Client *TRNG: Training Execution*

An end-user training environment. This client will be refreshed from the *training master* client. Training takes place here.

- Client *SAND: Sandbox*

A *sandbox* client allows you to experiment with transaction data and configuration settings. This client contains application data.

Sandbox environments are made available for testing specific functionality and in particular system-wide or “non-removable” changes (such as the activation of *Business Functions*).

**Caution:**

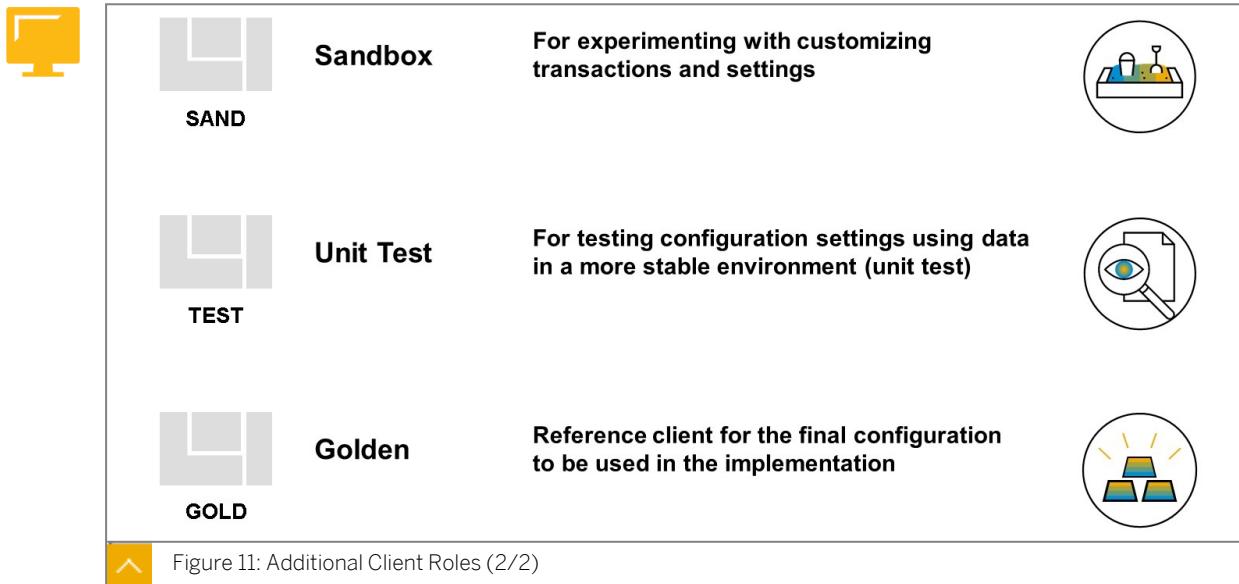
The transport of changes out of the sandbox should **not** be permitted. The sandbox is a purely standalone landscape with no connections to other environments. However, the sandbox may receive changes from the development in order to stay up-to-date.

- Client *TEST: Unit Test*

The goal of a *unit test* client is to verify customizing changes against sample data in a more stable environment. This can be done by copying customizing settings from the customizing client to the *TEST* client. This is called a *unit test*. In case that the test data has been destroyed during the tests, this client can be refreshed from the *test data backup* client (not included in this list).

- Client GOLD: *Golden Client*

The *golden client* is the “ultimate” reference client for all the good, complete and final configuration that is being used in the implementation. This client does not contain any application data, so you cannot perform any transactions here. This is a sensitive client and access needs to be restricted.



### Example for a Transport Landscape

The following figure shows a four-system landscape including two supporting systems together with the clients in each of the systems as an example. Let us assume that the customer performs agile software development with the help of *sprints*.

In this example, development and customizing is performed in the *DEV* system, client 100. A separate client (300) in the *DEV* system is used for unit tests by the developers (who test code changes) and consultants (who test customizing changes). Transport requests are released only after a successful unit test and a sprint review meeting.

Client 100 of system *QAS* then is used for functional tests of the new functions from the sprint. In addition,

- *functional integration tests* are executed to check the new functions in the context of the overall business process.
- *regression tests* can be executed to test whether or not the new functions will have no negative impact.
- *user acceptance tests* can be performed by the business key users.

Client 100 of system *PRE* then can be used for *volume tests*, *performance tests* of the business processing including the new functions, additional *user acceptance tests* and for the *final integration test* before – after final sign-off / approval – the changes are imported into the productive client in the *PRD* system.

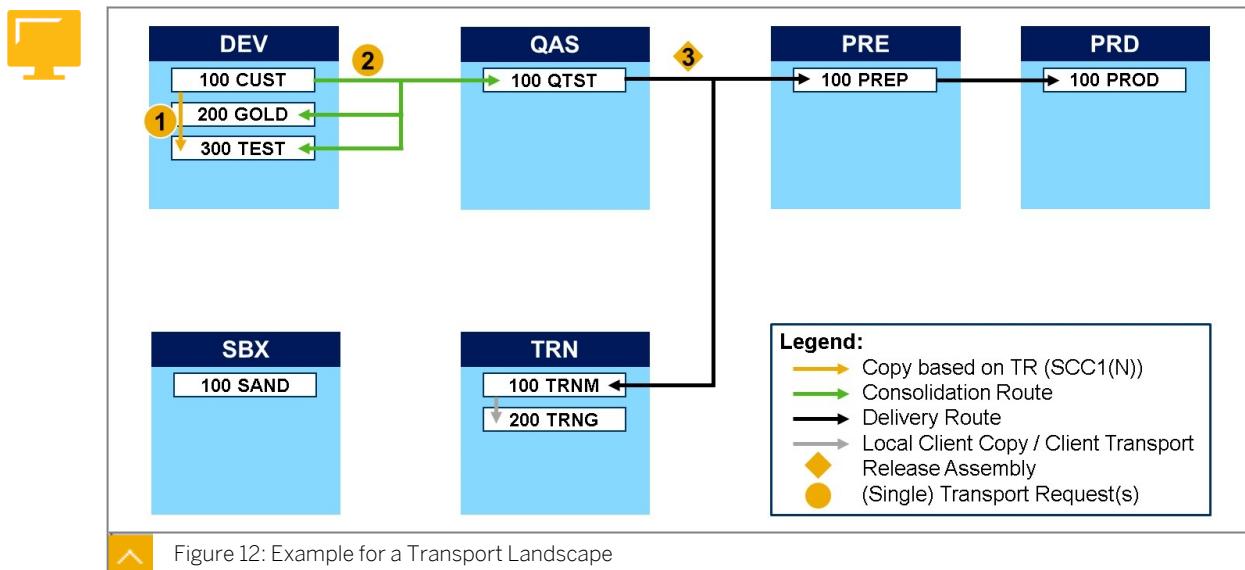


Figure 12: Example for a Transport Landscape

The export of transport requests from the *DEV* system must be planned carefully both because this fills the import queues of both the *QAS* system and the *DEV* system and because transport requests should not be deleted from any import queue as this may result in inconsistencies.



#### Note:

The content of the transport requests should at least be tested in the *TEST* client before they are released. To transfer the content of the transport request(s) to be tested to the *TEST* client, transaction *scc1* (*Copy by Transport Request*) or *SCC1N* (*Copy Data from Transports* – available as of SAP S/4HANA 1909) can be used.

In this four-system landscape, once a transport request has been released from the development system, it can be imported more or less immediately into the clients of the *QAS* system – provided that there is no conflict with ongoing (functional) tests.



#### Caution:

The import into the pre-production system should take place as late as possible, however. Only changes that are supposed to go to production with the next *Go Live* should be allowed to enter the pre-production system.

The *SBX* system is required for the initial testing of Support Packages or other solution components (such as the activation of *Business Functions*) without impacting the development environment.



## LESSON SUMMARY

You should now be able to:

- Describe the data structure of an AS ABAP based SAP system
- List different client roles and explain their use
- Illustrate an example of a multi-system landscape



# Unit 1

## Lesson 3

# Describing System Change Options and Client Change Options

## LESSON OVERVIEW

In this lesson, you will learn about the system change options and client change options of an AS ABAP-based SAP system.

## Business Example

As a transport administrator, you need to set the correct system change options for each SAP system and the correct client change options for all clients to prevent – for example – that changes to the repository can be performed directly in the production environment.



## LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Use system change options and client change options for your transport landscape
- Outline the idea of Customizing using current settings

## System Change Options



Software Component	Technical Name	Modifiable
S4FND	Modifiable	
SAP_ABA	restricted modifiability	
SAP_BASIS	restricted modifiability	
SAP_BW	Modifiable	
SAP_GWFND	Modifiable	
SAP_HR	restricted modifiability	
Subcomponent SAP_HRCAE of SAP_HR	Modifiable	

Namespace/Name Range	Prefix	Modifiable	Tech. ID
/0SJ1G/		Modifiable	/0CUST/
/0SJ1U/		Modifiable	/0SAP/
/0SJ2G/		Modifiable	/0SJ1G/
/0SJ3K/		Modifiable	/0SJ1U/
/0SJ5K/		Modifiable	/0SJ2G/
Recycling Administration		Modifiable	/0SJ3G/

Figure 13: System Change Options

The system change options define whether repository objects and cross-client customizing objects can be globally modified. If they are globally modifiable, you can specify whether or not each software component and namespace can be modified.

To reach the system change option, go to transaction SE06 and choose *System Change Option*.

A software component is a set of dedicated packages.

For the software components, there are four different possible settings:

- Modifiable
- Restricted Modifiability
- This means that you can only create objects as non-originals.
- Not Modifiable (Enhanceable Only)

    This means that changes are not allowed. Objects can be enhanced using the *Enhancement Framework* only.

- Not Modifiable (Not Enhanceable)

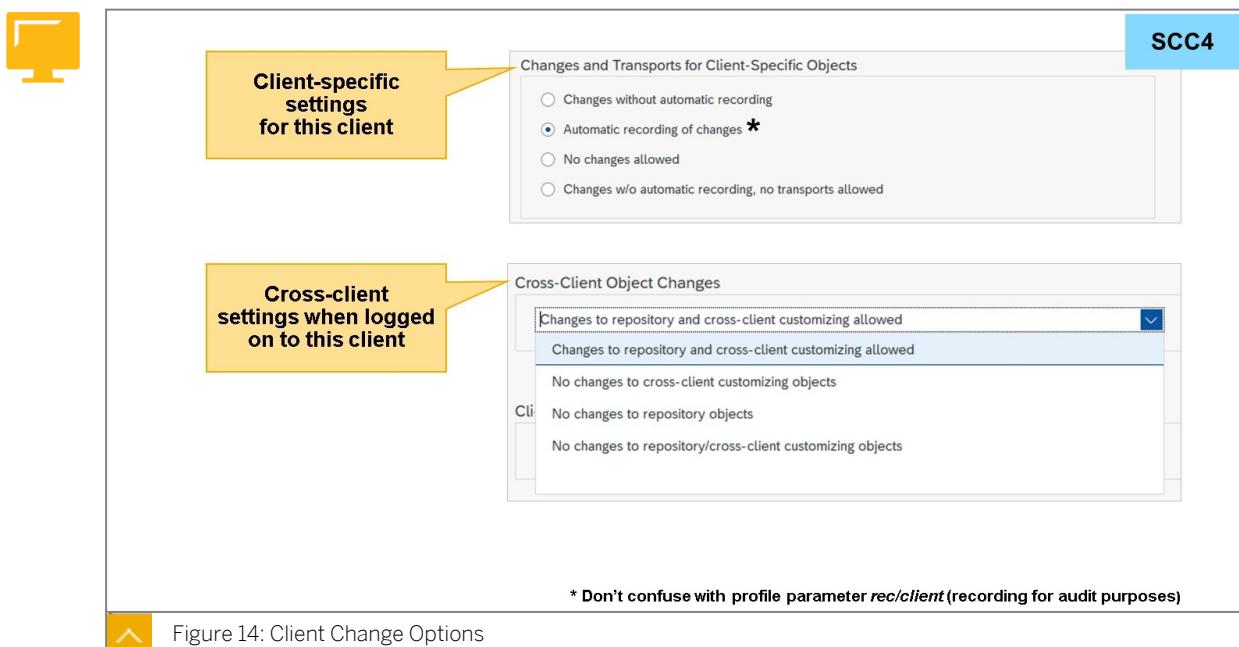
    This means that changes and enhancements are not allowed.

You can reserve name spaces for software developed by SAP customers and partners. These name spaces are designed for large-scale customer enhancements, as well as add-ons developed by partners.

The syntax for objects from reserved name spaces is as follows: /<name\_space>/<object\_name>.

For each SAP system, the customer can define whether or not objects contained in the name spaces and name ranges can be changed. To enable you to change objects, the *Global Setting* of this SAP system in transaction SE06 must be *Modifiable*.

## Client Change Options



Client change options, found in the client's T000 master table, can be maintained using transaction SCC4.

Two settings must be maintained to implement controls on where changes are made, and to enforce the changes being recorded to transport requests.

These settings are as follows:

- Changes and transports for client-specific objects.
- Cross-client object changes.

The following lists provide a brief description of what each option allows.

### **Changes and Transports for Client-Specific Objects**

#### **Changes without automatic recording**

This flag means that changes in the customizing settings of the client are permitted. The changes are not automatically recorded in a transport request.

Customizing settings can be exported from the client by manually creating a transport request and adding the objects in question to this transport request.

This variant makes sense for pure test, training or demonstration clients.

#### **Automatic recording of changes**

This flag means that changes are permitted in the customizing settings of the client. Changes are automatically recorded in a transport request.

You can also export customizing settings independently of the automatic recording from the client by manually creating a transport request and adding objects to it.

This variant can be used for customizing clients.



#### Hint:

In a productive client, settings that can be maintained as “current settings” are excluded from automatic recording. Current settings (such as exchange rates, posting periods, and so on) are maintained without recording the changes.



#### Note:

*Automatic recording of changes* does not refer to recording for audit purposes. Recording for audit purposes is also possible.

- To record data changes from within the SAP system, you can set the profile parameter `rec/client`, via transaction `RZ10`.
- To record data changes caused by importing transport requests, you can set the `tp` parameter `RECCCLIENT`, using transaction `STMS`.

With new installations of SAP S/4HANA 2021 or conversions to SAP S/4HANA 2021, table data change logging is enabled by default. For details, see SAP Note [3093760 –Table Data Change Logs In ABAP Platform](#).

### **No changes allowed**

This flag means that changes in the customizing settings of the client are not permitted.

However, (existing) customizing settings can be exported from the client by creating a transport request and manually attaching objects to it.

In a productive client, customizing settings that can be maintained as “current settings” are excluded from the client lock. This means that current settings (such as exchange rates, posting periods, and so on) can always be maintained in the productive client.

This variant can be used for productive clients, demonstration clients, and training clients.

#### **Changes without automatic recording, no transports allowed**

This flag means that changes are permitted in the customizing settings of the client. However, the changes are not automatically recorded in a transport request.

In contrast to the *Changes without automatic recording* option, no manual transport of customizing settings is possible from the client.

This variant can be used for pure test clients or sandbox clients.

#### **Cross-Client Object Changes**

This feature controls the clients in which you are allowed to maintain cross-client objects.

It makes sense not to allow cross-client-maintenance if the objects to be maintained are exclusively application objects (all client-specific) or client-related customizing objects. In this case (for example, in a test client), you must ensure that cross-client functionality can't be changed.

Cross-client-functions include all cross-client-customizing objects (for example, factory calendar, definition of price list conditions, printer controls). In particular, they also include all objects of the SAP repository (programs, function modules, screens, dictionary object, and so on).



##### Note:

Changes to cross-client objects are allowed only when **both** the settings in the *Client maintenance* (transaction `SCC4`) **and** the *System change options* (transaction `SE06`) don't prevent them.

#### **Changes to repository and cross-client customizing allowed**

There are no restrictions on the maintenance of cross-client objects for the client when this setting is used. Both cross-client customizing objects and objects of the SAP repository can be maintained.

Use this setting in the customizing and development client only.

#### **No changes to cross-client customizing objects**

Cross-client customizing objects can't be maintained in the client with this setting.

Only use this if the development client is separated from the customizing client.

#### **No changes to repository objects**

With this setting, objects of the SAP repository can't be maintained in the client.

Only use this if the customizing client is separated from the development client.

#### **No changes to repository and cross-client customizing objects**

This is a combination of both restrictions: Neither cross-client customizing objects nor objects of the SAP repository can be maintained in the client.

Use this in every client in the landscape, except for the customizing and development clients.



Hint:

If you use a job step user for the *Technical Job Repository* in SAP S/4HANA (transaction `SJOBREPO_STEPUSER`), you may enter an entry in table `CLMS_TENANT` (*Tenant Information (Extension to T000)*), which may result in additional fields in transaction `SCC4` (such as *Tenant ID*). See SAP Note [2449125](#) – *Create and assign job step user for Technical Job Repository*.

## Customizing Using Current Settings

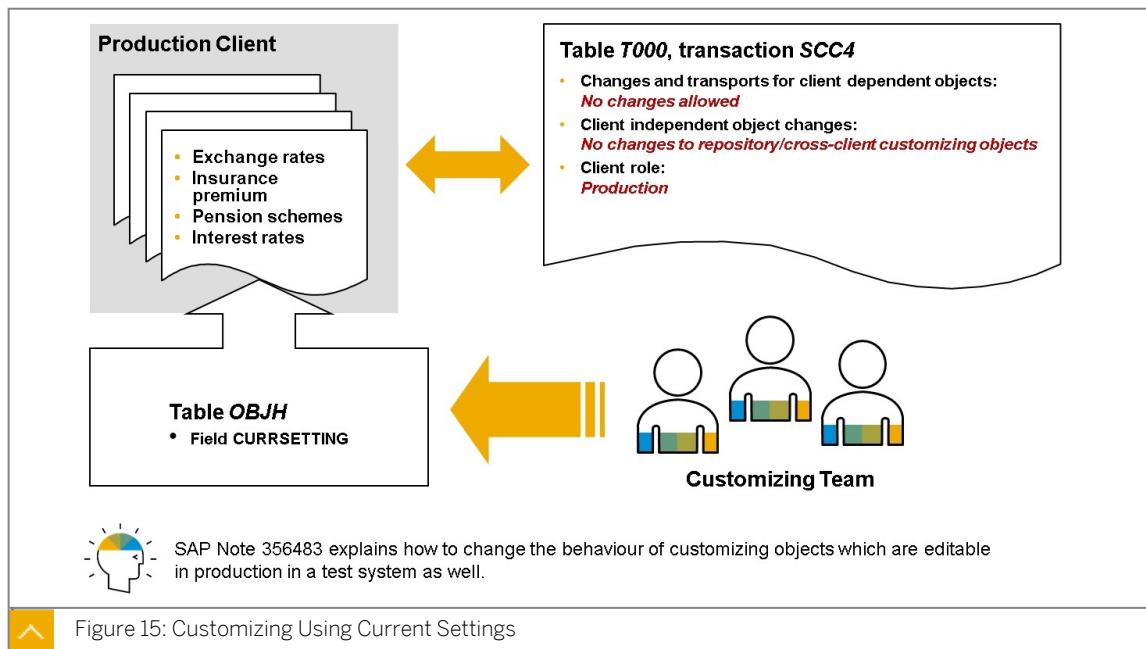


Figure 15: Customizing Using Current Settings

In a customizing project, and also during the maintenance of an SAP system, all customizing changes should be transported and tested in the quality assurance system. But certain kinds of customizing changes, or *Data-only customizing changes*, need to be carried out in a production client, without being saved as transport requests. Examples of such data, which may require frequent adjustment in SAP systems, include

- Interest Rates,
- Health Insurance Premiums,
- Pension Schemes,
- Tax Schemes, and
- Currency Exchange Rates,

because these types of changes have a frequent change rate, and they are not subject to extensive testing like other customizing changes. These changes are done directly in the productive client in the production system. To avoid having to use transport requests for these changes, the *Current Settings* function was introduced.

The *Current Settings* function may be used within a production client, without impacting business flow-related customizing objects.

SAP-approved *Current Settings* for customizing objects are kept in the *CURSETTING* field, in the *OBJH* table.

When using *Current Settings* in a production environment, the following settings need to be maintained:

- The client role is set to: *Production*.
- Cross-client object changes are set to: *No changes to Repository and cross-client Customizing objects*.
- Changes and transports for client-specific objects are set to: *No changes allowed*.



Note:

If you want to know how the object attribute *Current Settings* can be assigned to a maintenance object, see SAP Note [2442887](#) – SOBJ | *How to assign object attribute Current Settings to a maintenance object*. To deactivate the transport connection for certain customizing settings, follow SAP Note [2444878](#) – SE54 | *How to deactivate the transport connection*.

For view maintenance transactions, you have the option of maintaining current settings in test clients that are “not modifiable” also. For more information, see SAP Note [356483](#) – *In test system, behavior of customizing objects which are editable in production i.e. Current Settings*.



## LESSON SUMMARY

You should now be able to:

- Use system change options and client change options for your transport landscape
- Outline the idea of Customizing using current settings

## Learning Assessment

1. Which of the following are SAP systems?

*Choose the correct answers.*

- A SAP ECC
- B SAP ERP
- C SAP S/4HANA Server
- D SAP Business Suite
- E SAP BW

2. SAP recommends a three-system landscape as the minimal landscape when performing own development

*Determine whether this statement is true or false.*

- True
- False

3. Which of the following is always cross-client?

*Choose the correct answer.*

- A Repository objects
- B Application data
- C Customizing
- D User master records

4. Which of the following are standard (logical) client roles?

*Choose the correct answers.*

- A Development and Customizing
- B Monitoring
- C Quality Assurance
- D Importing
- E Production

5. If you set the client change options (transaction SCC4) for client 100 to *Changes to repository and cross-client customizing allowed* and the system change options (transaction SE06) *global settings* to *Not Modifiable*, you can then change repository objects from within client 100.

*Determine whether this statement is true or false.*

- True
- False

6. Which of the following client change options allow you to change client-specific customizing within the specific client?

*Choose the correct answers.*

- A Changes without automatic recording
- B Automatic recording of changes
- C No changes allowed
- D Changes w/o automatic recording, no transports allowed

## Learning Assessment - Answers

1. Which of the following are SAP systems?

*Choose the correct answers.*

- A SAP ECC
- B SAP ERP
- C SAP S/4HANA Server
- D SAP Business Suite
- E SAP BW

You are correct! The SAP ECC, SAP S/4HANA Server and SAP BW are SAP systems. SAP ERP is not an SAP system, it is an application. SAP Business Suite is not an SAP system, it is the bundle of the applications SAP ERP, SAP CRM, SAP SCM, SAP SRM and SAP PLM.

2. SAP recommends a three-system landscape as the minimal landscape when performing own development

*Determine whether this statement is true or false.*

- True
- False

You are correct! A three-system landscape is the recommended minimal landscape when performing own development.

3. Which of the following is always cross-client?

*Choose the correct answer.*

- A Repository objects
- B Application data
- C Customizing
- D User master records

You are correct! Repository objects are always cross-client. Application data is client-specific. Customizing is partly client-specific and partly cross-client. User master records are client-specific.

4. Which of the following are standard (logical) client roles?

*Choose the correct answers.*

- A Development and Customizing
- B Monitoring
- C Quality Assurance
- D Importing
- E Production

You are correct! *Development and Customizing, Quality Assurance and Production* are standard (logical) client roles. *Monitoring* and *Importing* are not client roles.

5. If you set the client change options (transaction SCC4) for client 100 to *Changes to repository and cross-client customizing allowed* and the system change options (transaction SE06) *global settings* to *Not Modifiable*, you can then change repository objects from within client 100.

*Determine whether this statement is true or false.*

- True
- False

You are correct! It is not possible to change repository objects if the system change options, *global settings* are set to *Not Modifiable*. It does not matter how the client change options are set in this case.

6. Which of the following client change options allow you to change client-specific customizing within the specific client?

*Choose the correct answers.*

- A Changes without automatic recording
- B Automatic recording of changes
- C No changes allowed
- D Changes w/o automatic recording, no transports allowed

You are correct! *Changes without automatic recording, Automatic recording of changes* and *Changes w/o automatic recording, no transports allowed* allow you to change client-specific customizing within the specific client. *No changes allowed* does not allow you to change client-specific customizing within the specific client.

### Lesson 1

Setting Up the Transport Management System (TMS)

29

### Lesson 2

Configuring Extended Transport Control

51

### UNIT OBJECTIVES

- List the main steps to set up a system landscape
- Explain the use of the transport directory
- Define the terms transport domain, system landscape, and transport group
- Create a transport domain and add additional systems to it
- Outline how to configure tp from within the Transport Management System
- Configure transport routes
- Configure the quality assurance approval procedure
- Check the setup of the Transport Management System (TMS)
- Explain the use of transport target groups and client-specific transport routes
- Describe a suitable system landscape for parallel project development and maintenance
- Outline the transport process between different transport groups and transport domains



# Setting Up the Transport Management System (TMS)

## LESSON OVERVIEW

In this lesson, you will learn how to set up the *Transport Management System (TMS)*. First, some basic terms, such as *SAP system landscape*, *transport domain*, and *transport group* are introduced. Then, you will learn how to set up a transport domain and how to configure a transport domain controller. After that, transport routes are defined. Finally, the QA approval procedure will be introduced and the TMS setup will be checked.

### Business Example

An SAP system administrator needs to understand the roles of different SAP systems and clients in the SAP system landscape before making decisions on how to configure the Transport Management System (TMS). After you have installed the first SAP system, but before you perform customizing or development, you need to configure the TMS so that it will record, track and control customizing and development changes for the subsequent transports to other SAP systems in the SAP system landscape.



## LESSON OBJECTIVES

After completing this lesson, you will be able to:

- List the main steps to set up a system landscape
- Explain the use of the transport directory
- Define the terms transport domain, system landscape, and transport group
- Create a transport domain and add additional systems to it
- Outline how to configure tp from within the Transport Management System
- Configure transport routes
- Configure the quality assurance approval procedure
- Check the setup of the Transport Management System (TMS)

## Overview and Prerequisites

The following list provides an overview of what has to be considered before, during, and after installation of an SAP system.

### Setting Up the SAP System Landscape: Overview



- Before SAP system and database installation:

- Define the network structure for the SAP system landscape.
- Install the hardware and the operating systems for the SAP systems.
- During SAP system and database installation:
  - Create one common transport directory.
  - Create additional transport directories for external systems – if they exist.
- After SAP system and database installation:
  - If you have set up the SAP system using a database copy, initialize the *Change and Transport Organizer* (CTO) (by choosing *Perform Post-Installation Actions* in transaction SE06).
  - Set up the Transport Management System (TMS).

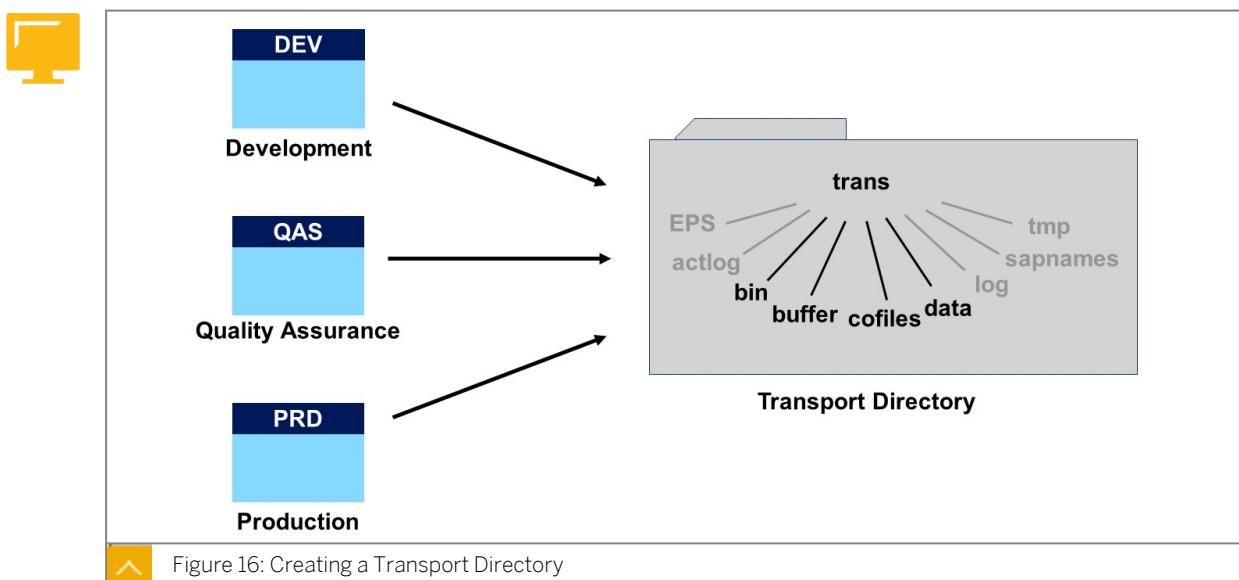
**Hint:**

If you have set up the SAP system using a database copy: To initialize the *Change and Transport Organizer* (CTO), go to transaction SE06 and choose *Database Copy or Database Migration* → *Perform Post-Installation Actions*.

To configure the SAP system landscape using the TMS, go to transaction STMS. This process will be described in detail in this lesson.

## Introduction to the Transport Directory

To set up your SAP system landscape, it is sufficient to start with an existing SAP system as a development system. The quality assurance and production systems are not required at this stage. All future SAP systems can be represented at this stage as “virtual” systems. You also need to create a transport directory at file system level for all SAP systems. This directory is required for the TMS.



**Note:**

Depending on your SAP system, the global transport directory and all necessary subdirectories may be created automatically during the installation of the SAP system. See the installation guide of your SAP system for more details.

The SAP profile parameter *DIR\_TRANS* has to point to the path of the transport directory. For Linux and UNIX, the default path is */usr/sap/trans*. For Microsoft Windows, the default path is *\\$\\$(SAPGLOBALHOST)\sapmnt\trans* or *\\$\\$(SAPTRANSHOST)\sapmnt\trans*.

**Note:**

For more information about the profile parameter *DIR\_TRANS*, see SAP Note [2506805 – Transport Directory DIR\\_TRANS](#).

The subdirectories required in the common transport directory include (among others):

- *EPS*: download directory for SAP Support Packages (Electronic Parcel Service).
- *actlog*: information about SAP user per transport request and transport task.
- *bin*: configuration files for *tp* (*TP\_<domain name>.PFL*) and TMS (*DOMAIN.CFG*).
- *buffer*: transport buffer file for each SAP system, indicating which transport requests are to be imported in which sequence.
- *cofiles*: command or transport request information files, which include information on the transport type, object classes, required import steps, and post-processing exit codes.
- *data*: data part of the exported transport requests.
- *log*: transport logs, trace files, and statistics.
- *sapnames*: information about transport requests per SAP user.
- *tmp*: temporary data and log files.

Transporting enables you to synchronize customizing and development in multiple SAP systems, through the transfer of changes from the development system to subsequent SAP systems. Transports along the transport routes must occur in only one direction.

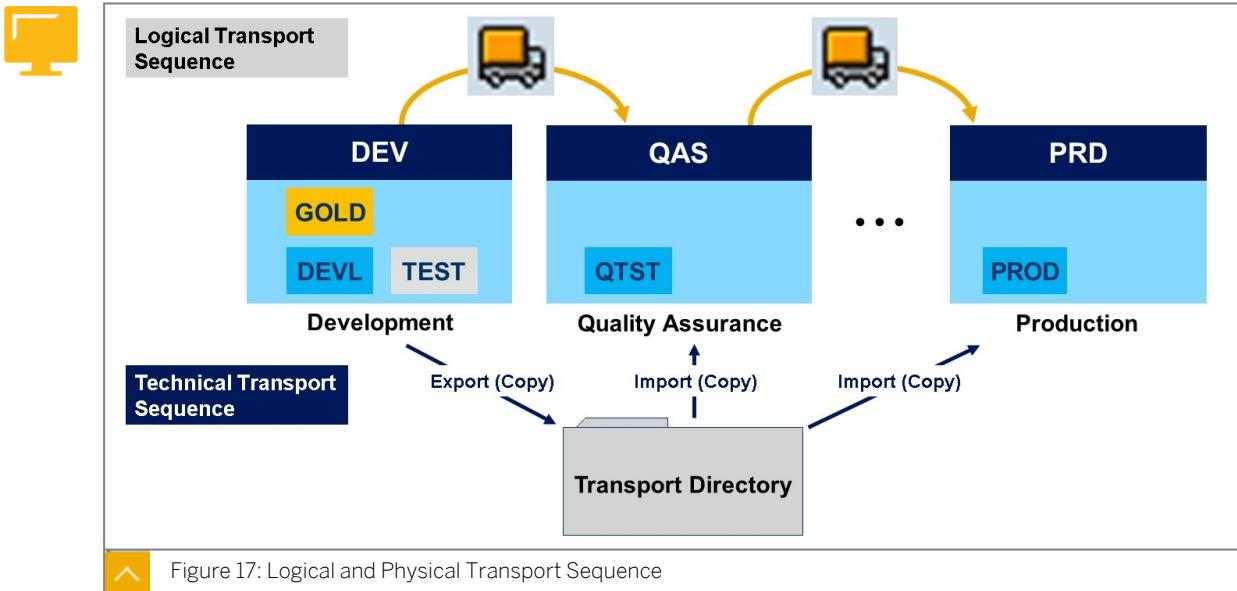


Figure 17: Logical and Physical Transport Sequence

As mentioned above, transporting requires a transport directory to enable SAP systems in the SAP system landscape to store and access various data files, command files, and log files.

Physically, objects in a three-system landscape are transported in three steps:

1. All objects in a (transportable) transport request, that are to be released, are exported by being copied from the database of the source system to the transport directory.
2. These objects are imported from the transport directory into the database of the quality assurance system. Here, they are tested and validated.
3. After testing and verification, the objects can be imported from the transport directory into the database of the production system.



#### Note:

The terms “export” and “import” are meant as *copy*, not as *move*.

At the end of the process, the objects exist four times: in the database of DEV, QAS, and PRD, and in the transport directory.

## TMS: Concepts and Terminology

Depending on the size of an organization and the scope of its SAP implementation, there may be many different individuals carrying out customizing and development projects. Some will be involved in customizing specific application areas, others may be involved in the development of new ABAP programs, others in the QA testing and acceptance of changes. There can be hundreds or even thousands of changes that have to be transported through the SAP systems in the landscape. The transport of these changes will be done using the functions of the TMS.

The concepts behind TMS are as follows:

- Centralized configuration of Change and Transport System (CTS) for all SAP systems.
- Centralized management of transport requests, especially the import process.

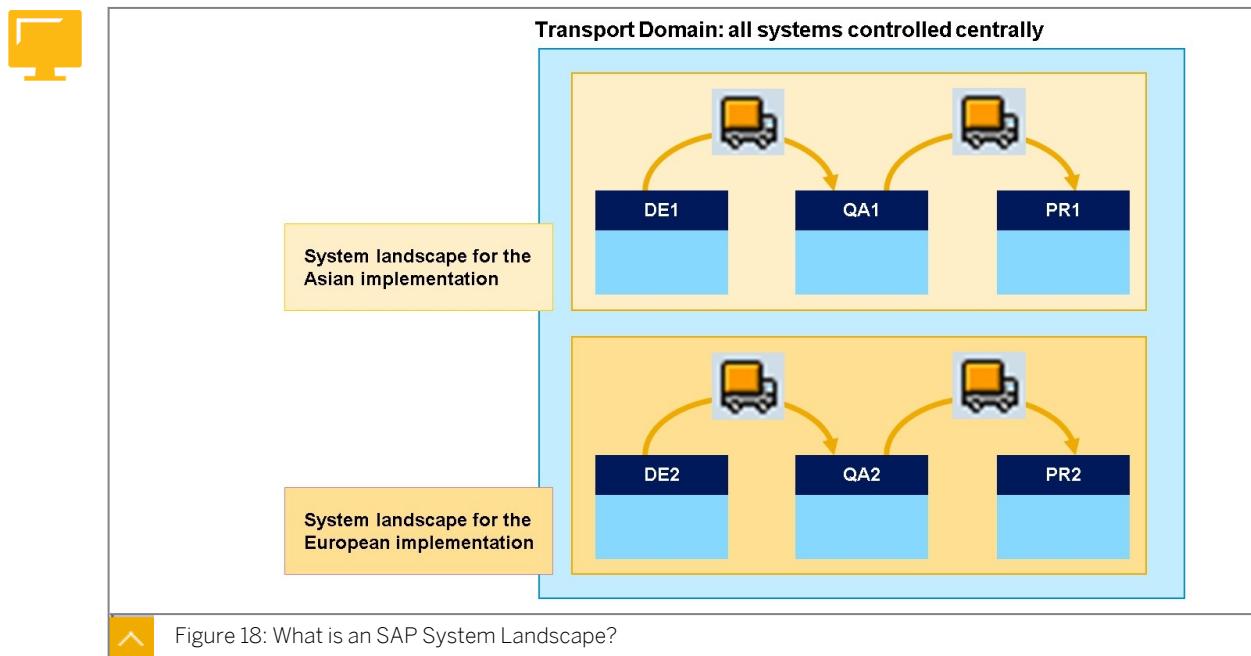
- Transport strategy based on predefined transport routes.

The purpose of the TMS, accessed via transaction STMS, is to centrally control the propagation of changes through the SAP system landscape based on predefined paths. This is designed to ensure the consistency of the SAP repository and the contents of the customizing tables in all SAP systems in the landscape. All necessary activities can be done from within the SAP system (using the SAP authorization concept) and there is no need to manually execute scripts at operating system level.

With TMS you are able to perform the following activities:

- Define the SAP system role within an SAP system landscape or transport domain.
- Configure the transport routes, using either an editor or delivered standard configuration settings.
- Configure the transport tool program's (*tp*) parameter profile.
- Display the import queues of all SAP systems in the transport domain.
- Define quality assurance and approval procedures in the QA system.
- Schedule the import of transport requests in an import queue.
- Perform transports between SAP systems without a common transport directory.
- Handle transport proposals of developers.

TMS enables SAP system administrators to centrally manage the transport configuration of multiple SAP systems by defining transport domains, assigning transport domain controllers, and defining transport routes.



A **transport domain** consists of all SAP systems that you plan to manage centrally. Within the transport domain, all SAP systems must have unique SAP system IDs (SIDs). Only one of these SAP systems is identified as the (transport) domain controller.

**Note:**

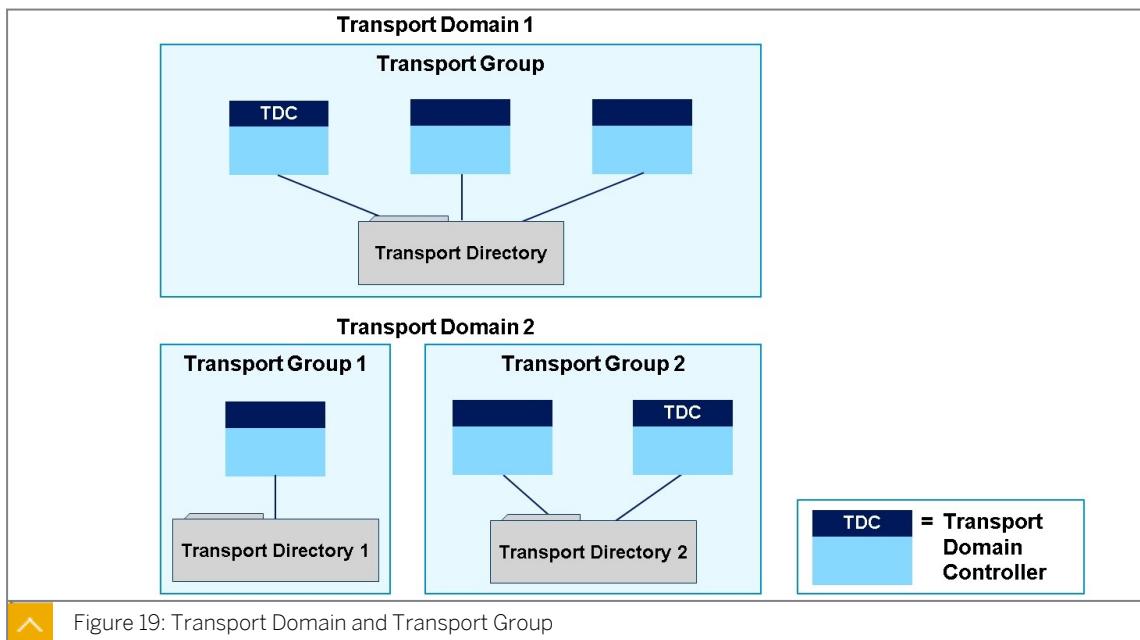
The **transport domain controller** is the SAP system where all TMS configuration settings are maintained. Any changes to the configuration settings are distributed to all SAP systems in the landscape. This ensures that the TMS configuration settings are consistent throughout the transport domain. The transport domain controller stores the reference configuration, and all other SAP systems receive a copy of the reference configuration.

An **SAP system landscape** is a set of SAP systems that share customizing and repository objects via transported transport requests. A typical landscape is made up of, but not limited to, a development, a quality assurance, and a production system. In most cases, the SAP system landscape and the transport domain are made up of the same SAP systems, but it is not uncommon to have multiple system landscapes within one transport domain.

Examples of a single transport domain with multiple system landscapes include:

- A multinational company may have separate landscapes for each subsidiary. DE1, QA1, and PR1 may be the landscape for the Asian implementation, and DE2, QA2, and PR2 may be the landscape for the European implementation (see the figure “What is an SAP System Landscape?”). Even though they are separate system landscapes, both can still be controlled centrally in one transport domain.
- An SAP customer may have multiple system landscapes for different kinds of SAP systems. DEV, QAS, and PRD may be the landscape for the SAP ECC systems, while DBW, QBW, and PBW may be the landscape for the SAP BW systems. Again, separate system landscapes are controlled centrally using one transport domain.

A transport domain contains at least one transport group. A **transport group** consists of one or more SAP systems that share a common transport directory. The following figure shows the relationship between a transport domain and a transport group.



TMS supports several transport directories within a single transport domain. This might be the case, for example, for the multinational company mentioned above where each subsidiary may have its own (local) transport directory. Or, it could be a transport domain containing a

landscape that has a DEV system with its own transport directory, and QAS and PRD systems sharing a common transport directory, for security reasons. In this case, the transport domain would consist of two transport groups.



Note:

The terms *transport domain*, *transport domain controller*, and *transport group* concern only the SAP systems in the environment. They don't include the transport relationship between SAP systems, which is defined by the transport routes.

## Establishing a Transport Domain

To set up a transport domain, first determine which SAP systems should be included in the transport domain. The transport domain should contain all SAP systems that will be centrally administered using TMS.



Note:

Not all SAP systems need to be installed and present when the transport domain is configured for the first time, but planning how the landscape will look like is necessary. These "future" SAP systems can be represented by *virtual systems*.

## Overview

The configuration of TMS can be broken down into three steps:

1. Configuring the transport domain defines which SAP systems will be included in the domain.
2. Configuring the transport routes defines the SAP system and client roles within the landscape(s).
3. Optional: Configuring the QA approval procedure defines who is responsible for approving the changes and the promotion of those changes to the delivery system or systems.

The figure "Setting Up the Transport Management System" shows the three steps. Note that the first two steps are mandatory for setting up an SAP system landscape. The last step (QA approval procedure configuration) is optional.

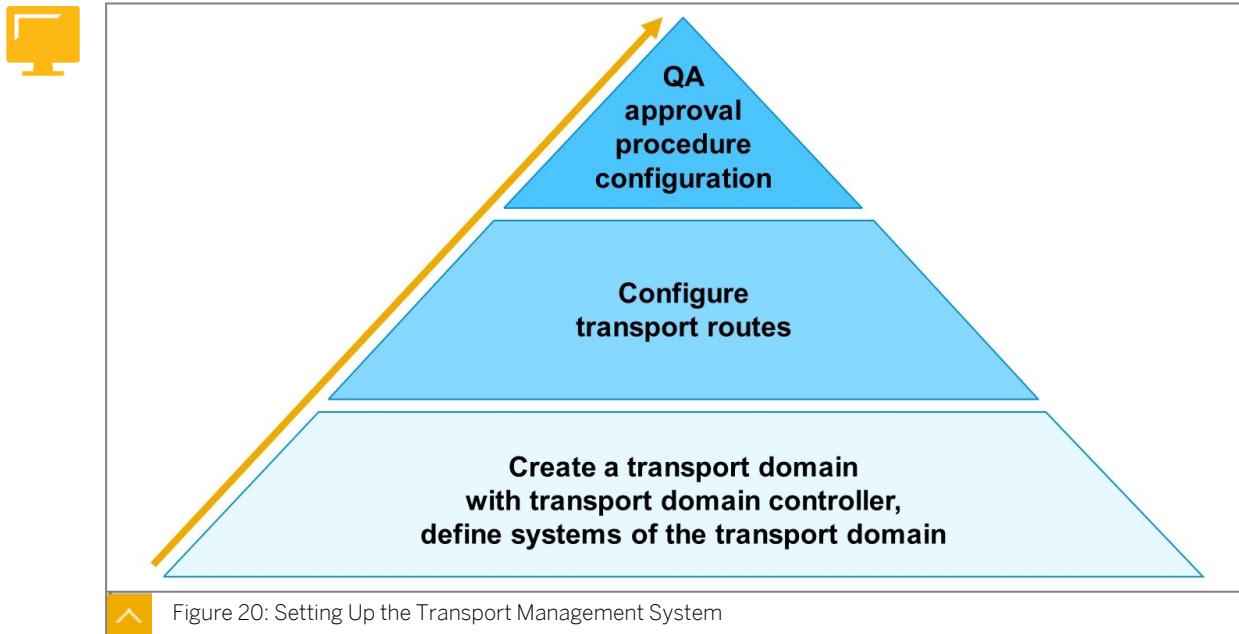


Figure 20: Setting Up the Transport Management System

### Initializing the Transport Domain Controller

The first SAP system you configure is automatically selected as the transport domain controller. However, you can later switch the role of the transport domain controller to a different SAP system.

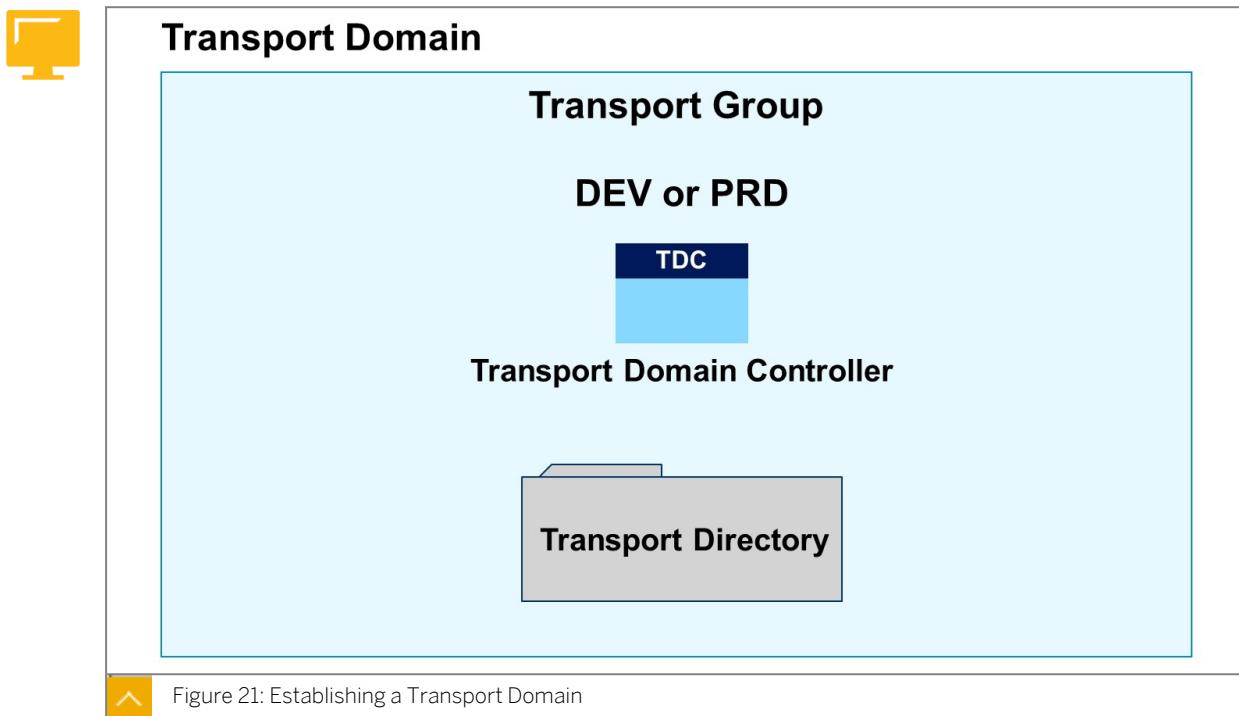


Figure 21: Establishing a Transport Domain

As certain configuration tasks that are relevant to the entire transport domain (for example, creating transport routes) can only be carried out on the transport domain controller, SAP recommends that the SAP system chosen to be the transport domain controller has the following attributes:

- High availability

- High security precautions
- Highest level of maintenance

A production system might be the ideal choice to be transport domain controller. As the development system is usually installed before the quality assurance and production systems, the common practice is to configure the development system as the transport domain controller first and later move the assignment of the transport domain controller to the production system. Another scenario might be to use an SAP system that you already use for central monitoring and central user administration as the transport domain controller.



**Note:**

The system load on the SAP system caused by work on the transport domain controller is very low.

When using TMS (transaction STMS) for the first time after a new SAP system installation, you are automatically prompted to initialize the TMS. You must initialize the TMS while logged on to client 000. To configure TMS, you need the authorization S\_CTS\_ADMIN (for authorization object S\_CTS\_ADM).

### Initializing the Transport Domain Controller

When there is no transport domain configured yet, the start of transaction STMS in client 000 will perform the following activities:



- Assign the SAP system as the transport domain controller.
- Create the transport domain name *DOMAIN\_<SID>*.
- Create the transport group *GROUP\_<SID>*.
- Create the system user *TMSADM* in client 000.
- Create RFC destinations.
- Set up the file *DOMAIN.CFG* and the *TP\_<domain name>.PFL*, in directory *bin* of the transport directory.

When you call transaction STMS in client 000 on the transport domain controller system for the first time, the following actions will automatically be carried out by the SAP system:

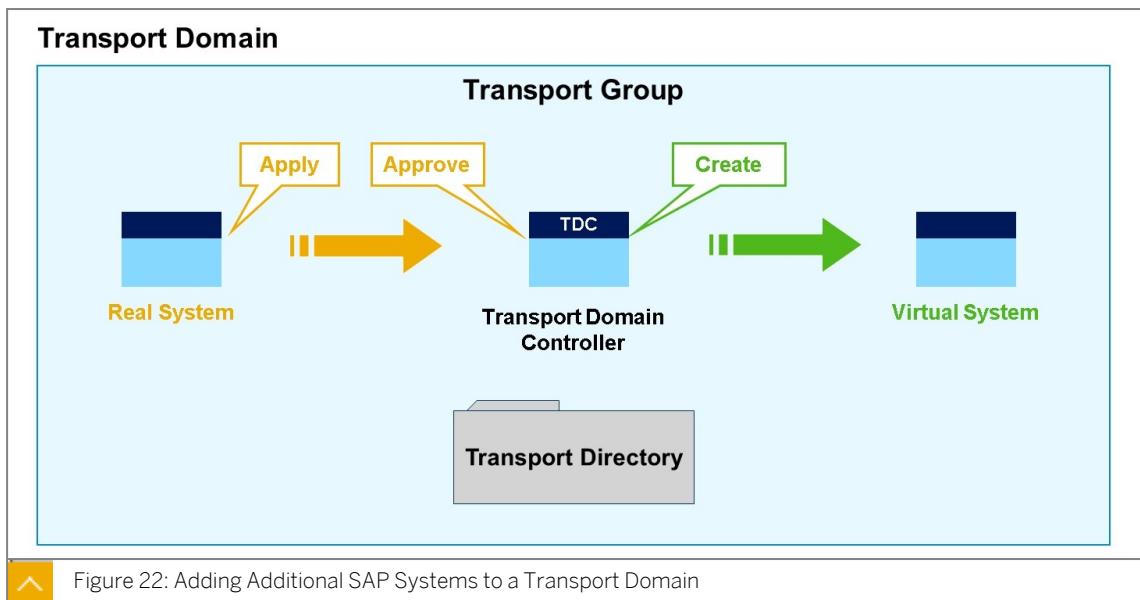
- A transport group is created with the name *GROUP\_<SID>*.
- In client 000, the system user *TMSADM* is created with assigned profile *S.A\_TMSADM*.
- The RFC destinations required for the TMS are generated.
- The TMS configuration file *DOMAIN.CFG* is stored in the transport subdirectory *bin*. This file contains the transport domain name and description, as well as the transport domain controller's host name, instance number, SID, and transport group.
- The transport profile for the transport control program *tp* is generated and stored in the transport subdirectory *bin*, under the name *TP\_<domain name>.PFL*. The parameters in this profile are maintained using transaction STMS.

**Note:**

The name of the transport domain can't contain blanks, and can't be changed later without reconfiguring the transport domain controller. By default, a transport domain will have the name *DOMAIN\_<SID>*, where *<SID>* is the SAP system ID of the transport domain controller system.

## Adding Additional SAP Systems to a Transport Domain

To add a new SAP system to an existing transport domain, you must perform configuration activities both on the new SAP system and on the transport domain controller.



Each additional SAP system must apply to the transport domain controller for inclusion to the transport domain. As the configuration of the TMS can only be done on the transport domain controller, the transport domain controller must confirm inclusion for each SAP system. The initial screen of transaction STMS at the applicant system indicates if the SAP system is waiting for inclusion in the transport domain.

The description of the transport domain is stored in the file *DOMAIN.CFG* in the *bin* subdirectory of the common transport directory. All other SAP systems connected to this transport directory read the file *DOMAIN.CFG* during STMS initialization to identify the transport domain controller and transport group address.

**Note:**

If your SAP systems don't have a common transport directory, you can manually configure TMS with the address of the transport domain controller. In this case, choose *Other Configuration → Include System in Domain* when entering transaction STMS for the first time. In the next dialog box, enter the *Target host* and the *Instance number* of the transport domain controller system.

When an SAP system has been configured for the TMS and has been approved in the transport domain, you'll find a newly created user TMSADM in client 000 and generated RFC destinations required for the TMS.



## Hint:

Due to the created RFC destinations, the password of user *TMSADM* needs to be consistent within the whole transport domain. For more information, see the following:

- SAP Note [761637](#) – Logon restrictions prevent *TMSADM* logon
- SAP Note [1568362](#) – *TMSADM* password change
- SAP Note [1801805](#) – Introduction of new destinations and patches in *TMS\_UPDATE\_PWD\_OF\_TMSADM*
- SAP Note [2493023](#) – *TMSADM* Problems: Required notes
- SAP Note [2816097](#) – User *TMSADM* is constantly locked
- Wiki *Changing the TMSADM password* (<https://wiki.scn.sap.com/wiki/display/SL/Changing+the+TMSADM+password>)



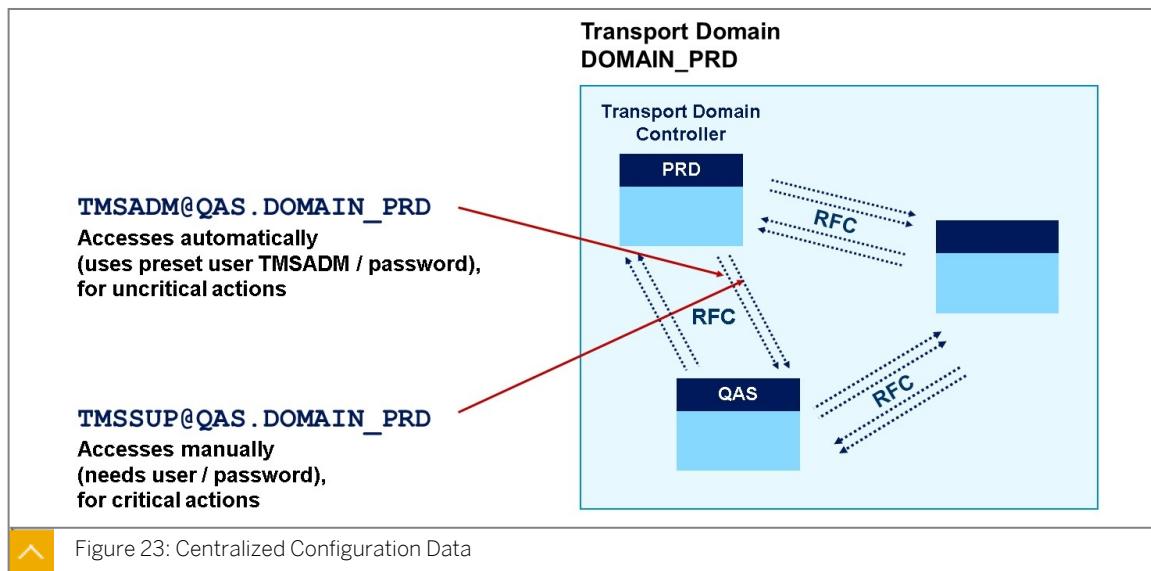
## Hint:

When configuring the TMS on an SAP system, you can specify the application server (that is, the instance) to be used for all TMS functions. Choose the instance with the highest availability.



## Caution:

To exchange data and objects between the SAP systems of an SAP system landscape, all SAP systems should have the same release and SAP Support Package level. For more information on transporting between different releases, see the composite SAP Note [1090842](#) – Cross-Release Transports.



The SAP systems within a transport domain communicate with each other using Remote Function Calls (RFCs). RFC communication requires user IDs to access target SAP systems.

When SAP systems are added to a transport domain, the necessary RFC destinations and user IDs are automatically configured by the TMS tool. The transport domain configuration settings are distributed throughout the transport domain using RFC communication.

Changes to the transport domain configuration are made on the transport domain controller system, and are then distributed to all SAP systems in the transport domain. Each time you make a change on the transport domain controller system, a dialog box is displayed asking whether to distribute the change. You can distribute several changes in one distribution process.

When SAP systems are initially added to the TMS configuration, the system user *TMSADM* is configured with limited authorizations. This user is used when distributing and activating the TMS configurations.

There are two types of RFC destinations created by the TMS:

- *TMSADM@<SID>.<domain name>*
- *TMSSUP@<SID>.<domain name>*

The user *TMSADM* is inserted as a system user in the *TMSADM@<SID>.<domain name>* destination, but there is no RFC user inserted into the *TMSSUP@<SID>.<domain name>* destination. Therefore, if you try to execute a function that goes beyond the limited authorizations of the *TMSADM* user, for example, scheduling an import, TMS uses the *TMSSUP* destination. As a result, the target SAP system prompts you for credentials of a user that does have the necessary authorizations.

### Virtual Systems

Extending a transport domain is not restricted to physically installed SAP systems. Virtual SAP systems are often included as place holders for planned SAP systems and are replaced by the planned SAP system after it has been physically installed. In addition, you can extend the transport domain to include external SAP systems, for example, an SAP system from a different transport domain.

By creating virtual SAP systems, you can model the transport routes of the planned SAP system landscape to ensure that the import queues of subsequent SAP systems will already be created and maintained for later use. In the beginning stages of a new SAP implementation, customers frequently have only the development system physically installed and store the development and customizing work in the transport directory and import queues of the respective planned SAP systems.



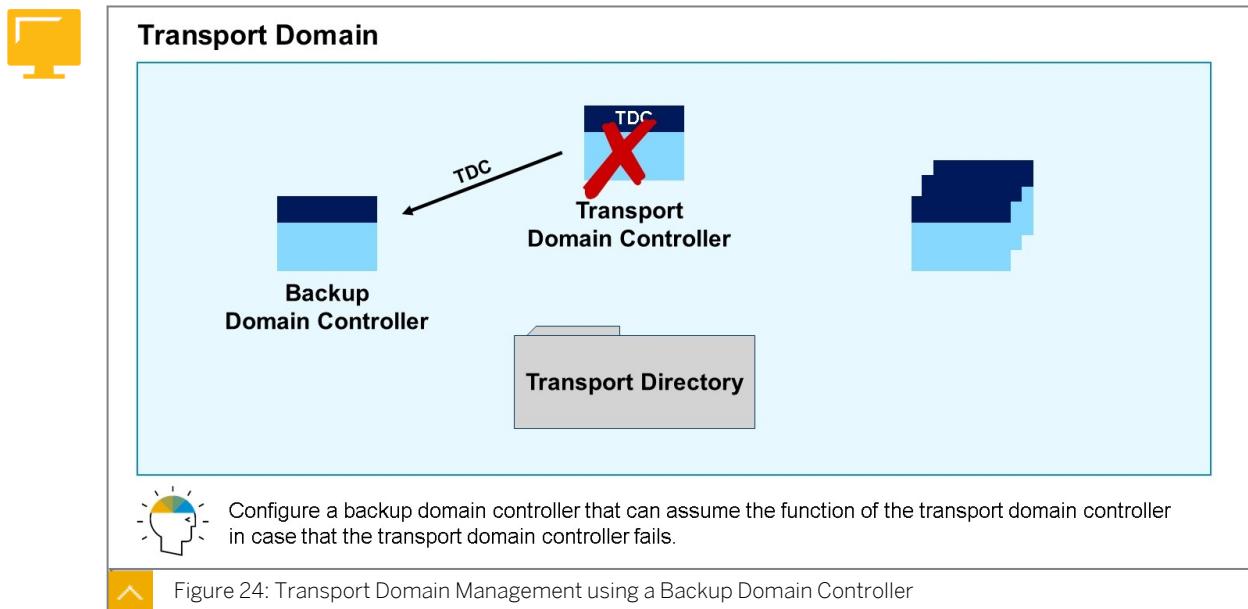
#### Note:

Because no RFC address can be created for a virtual system, files on the transport directory are accessed via an already existing SAP system. This SAP system acts as the **communication system** which must be an active SAP system in the transport domain. The default for this will be the transport domain controller system. This should only be changed if the planned SAP system is going to share a transport directory other than the one used by the transport domain controller and there is an active SAP system available in that target transport group.

### Backup Domain Controller

If the SAP system that is acting as the transport domain controller fails, no changes can be made to the TMS configuration. Therefore, SAP recommends that you configure a **backup domain controller** that can take over the function of the transport domain controller when required.

It may also be necessary to move the transport domain controller to another SAP system. In this case, you must activate the backup domain controller. Then, the backup domain controller becomes the transport domain controller, and the transport domain controller becomes the backup domain controller.



#### Hint:

The SAP system that you select as a backup domain controller must be an existing SAP system, it can't be a virtual or an external SAP system. In addition, the SAP system that you want to use as the backup domain controller must have the same release version as the transport domain controller. Otherwise, configuration information may be lost when changing the domain controller.

To check the current status of the transport domain configuration for each SAP system in the transport domain, access the TMS System overview. From the TMS initial screen, use the menu to choose *Overview → Systems*. In this overview, you can also see whether the configuration is up to date, and whether any errors occurred when distributing the configuration.

#### Configuring tp

The transport control program *tp* requires a transport profile. This profile provides information on establishing a database connection for all SAP systems in the transport domain, as well as other technical information on performing the transport process. TMS generates and manages this transport profile as a part of the transport domain configuration.

#### Caution:

Do not adjust the transport profile using a text editor at operating system level.

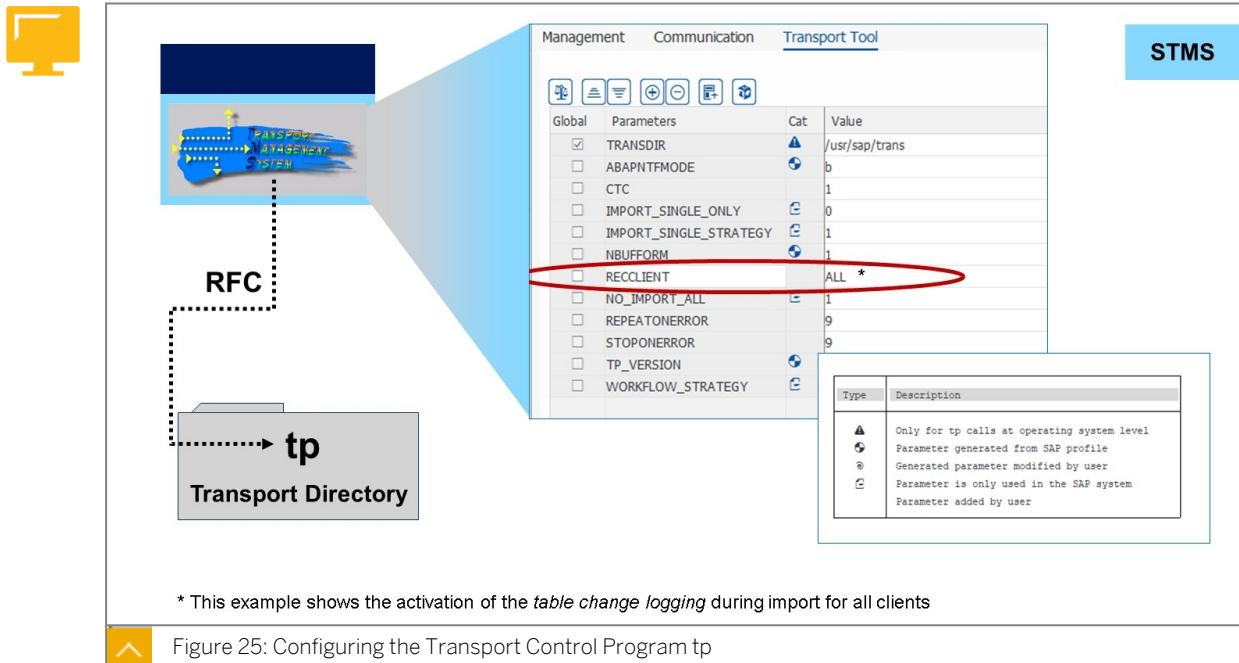


Figure 25: Configuring the Transport Control Program tp

To display the *tp* parameters of an SAP system, call transaction STMS. From the menu, choose Overview → Systems. Mark one SAP system and choose SAP System → Display. Choose the tab *Transport Tool*. From the menu, choose Goto → *tp* Parameters. This displays the parameters in *TP\_<domain name>.PFL*, as well as the default value of other parameters used by the programs *tp* and *R3trans*. If a parameter is not specified in the *TP\_<domain name>.PFL* file, a default value will be used. Global parameters override defaults. Local parameters override global parameters. For example, you can specify a global parameter value for all but one SAP system, if a local parameter applies to that SAP system.

**Hint:**

You can switch between the *Display All Parameters* and *Display Substituted Parameters Only* views, by using the *Display Less* and *Display More* buttons.

**Note:**

The example in the figure “Configuring the Transport Control Program tp” shows the setting of the *tp* parameter *RECCCLIENT* to *ALL* (all clients). This can be used to record changes to data **caused by the import of transport requests**. This may be necessary for audit purposes. To record data changes done **from within the SAP system**, you can use profile parameter *rec/client*.

## Configuring Transport Routes

Transport routes indicate the role of each SAP system and the flow of transport requests. The transport routes are what actually define your SAP system landscape.

### Overview

The initialization of TMS and the setup of the transport domain defines the systems in the environment in terms of the transport domain, transport domain controller, and transport group.

In addition, you must define the transport relationship between these SAP systems. Even though TMS has been initialized, you cannot perform transports until the transport routes have been configured and distributed.

After establishing a transport domain, you need to perform the following activities:

### Configuring Transport Routes



1. Model transport routes from the transport domain controller, using:
  - Default standard configurations (one-system, two-system, and three-system landscapes)
  - Graphical editor for non-standard configurations
2. Distribute and activate the new configuration data for all SAP systems within the transport domain.

To reduce the effort of specifying individual transport routes, you can use standard configurations. Transport routes for the standard configurations are generated automatically.

You can choose from the following standard configurations:

- *Single System*
- *Development and Production System*
- *Three Systems in Group*

If you use the standard configuration function, the current configurations for all SAP systems involved are replaced by the standard settings. Existing packages, transport layers, or objects created in the SAP system are not deleted.

For complex SAP system landscapes, define additional consolidation and delivery routes after setting up the transport route configuration on the basis of one of the standard configurations.

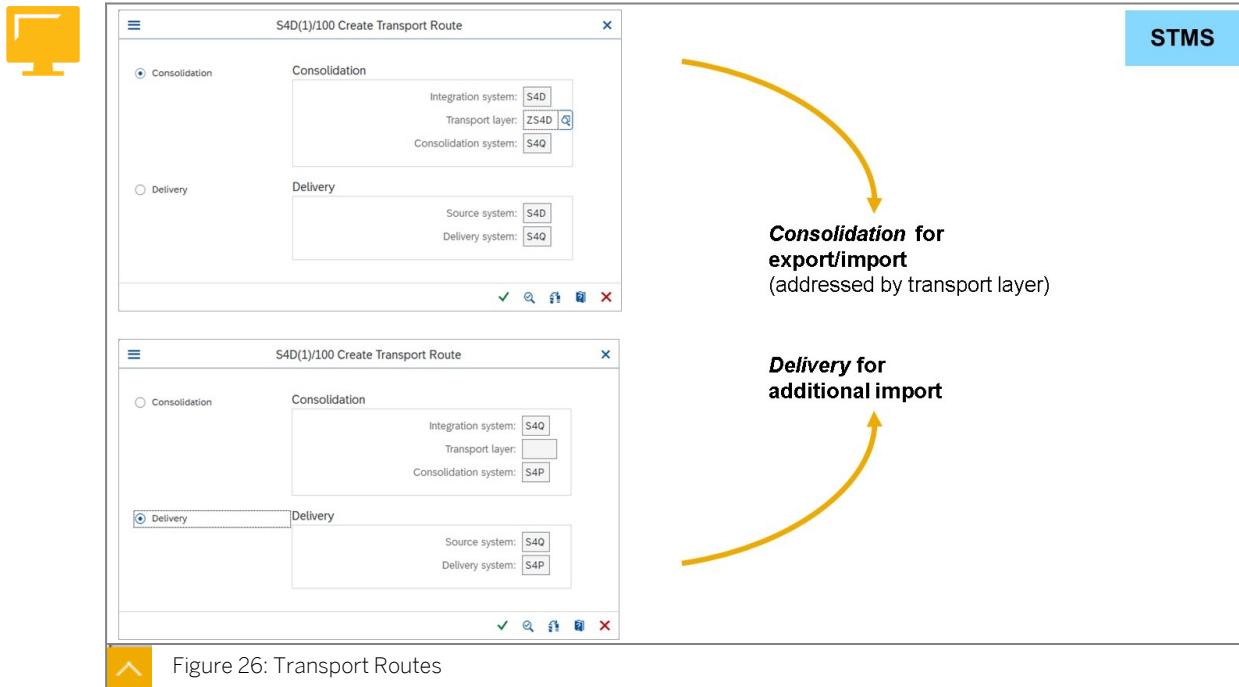
After you have defined transport routes, you must activate the changes and distribute them to all SAP systems in the transport domain.

There is a version management of the stored configurations. If you modify and save an active configuration, the version counter is increased. To activate a previous version, from the start screen of transaction STMS use the menu to choose *Overview* → *Transport Routes* and then *Configuration* → *Get Other Version*. This displays a list of all versions of the transport route configuration. Select the version you want from the list. To activate an older version, use the *distribute and activate* procedure.

To ensure consistency, transport routes can be configured only on the transport domain controller system. To help define a transport route, the TMS provides a graphical editor and a hierarchical list editor, which can be used interchangeably, as well as standard configurations for a one-system, two-system, and three-system landscape.

### Transport Layers and Transport Routes

Transport routes define the flow of the transport requests from one SAP system to the next. These routes are called either *Consolidation* or *Delivery* routes.



A **consolidation route** is an “export/ import” route. In a standard three-system landscape, the consolidation route typically proceeds from the development system (where the transport request is exported from) to the quality assurance system (where the transport requested is imported into). Consolidation routes have *transport layers* in order to be addressed.

A **delivery route** is an “another import” route. In a standard three-system landscape, the delivery route is specified between the quality assurance system and the production system. This is because there is no additional export from the quality assurance system, but another import in the production system. Delivery routes have no names.

All repository objects are grouped into logical units called *packages* (formerly: development classes). The definition of each package contains an assignment to a transport layer. The **transport layer** is the “name” of a consolidation route. Repository objects, via the assignment to the package, inherit this transport layer assignment.

All SAP delivered objects are assigned (according to their packages) to the transport route SAP.

Customizing objects (and others) are not grouped into packages. They follow the transport route for the standard transport layer. This standard transport layer is named *Z<SID>* by default. The SID is the SAP system ID of the development system.

### SAP System Roles within Transport Routes

In the context of transport routes, an SAP system may play the following roles:

#### Integration system

The source of a consolidation route, that is the SAP system where changes originate and are assigned to transport requests.

The SAP system where customer changes are integrated with the SAP standard, customer developments and modifications take place.

#### Consolidation system

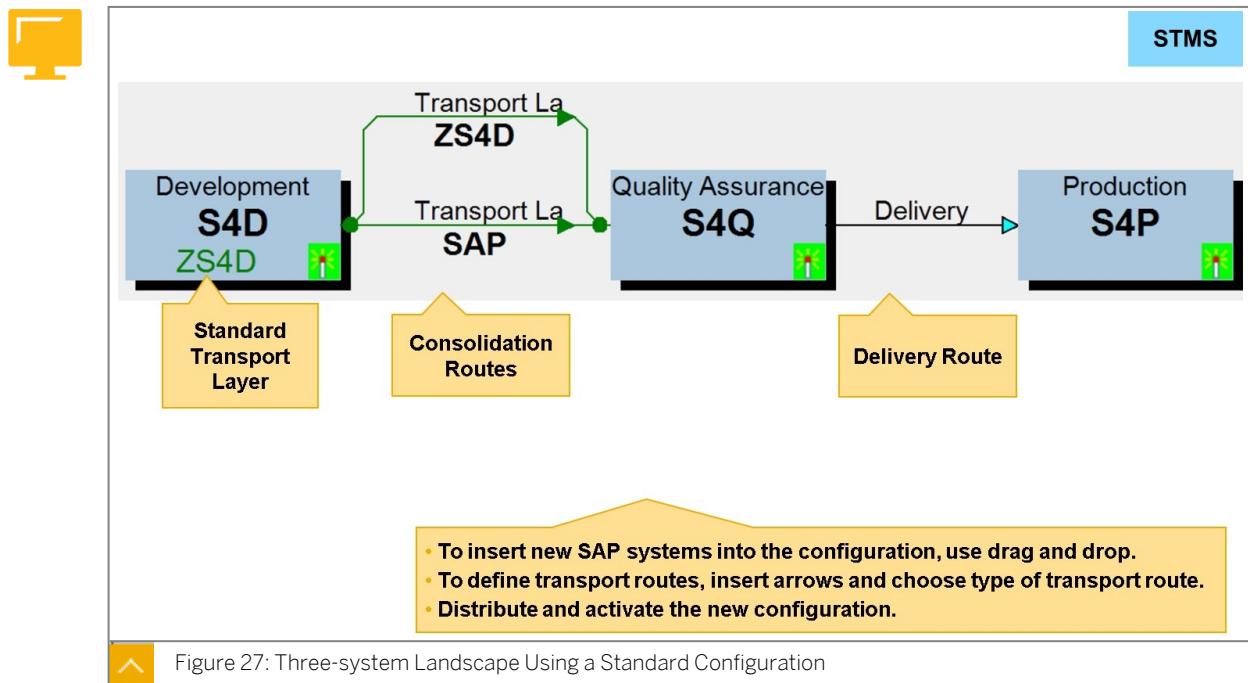
The target SAP system of a consolidation route.

#### Delivery system

The target SAP system of a delivery route.

### Source System

The source SAP system of a delivery route.



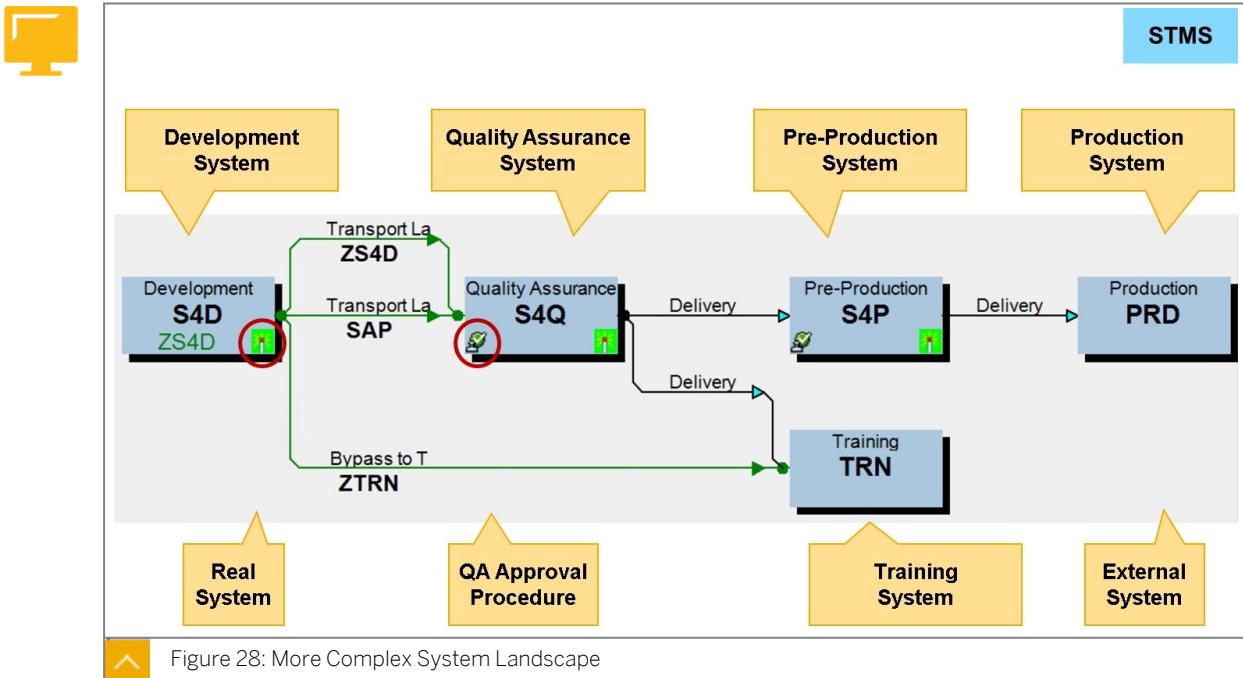
If you have a more complex landscape, you need to configure more transport layers and transport routes. You also need to reroute certain objects away from the standard transport routes, for example, if a separate training system exists and there are certain programs that are to run there, but you don't want those programs getting into the quality assurance system or into the production system.

If the standard configurations delivered by SAP don't meet your landscape needs, TMS has editors that allow you to maintain the landscape.

The graphical editor delivers a visual drag and drop interface, where the SAP systems in the transport domain can be positioned and linked using the mouse. This is the default editor in the *Transport Routes* area of transaction STMS.

To create transport routes, click the appropriate SAP systems from the node area, and drop them in the display area (by clicking again). To create a transport route between these SAP systems, from the screen *Change Transport Routes*, choose the *Add Transport Route* button. The mouse pointer becomes a stylus. Using the stylus, draw a line connecting the two SAP systems between which a transport route is desired. In the *Create Transport Route* dialog box, enter the required information for either the consolidation or the delivery route. In case of a consolidation route you need to assign a transport layer. You can create the transport layer right here. Finally, save your changes.

The following figure shows a more complex SAP system landscape.



### Distribution and Activation of Transport Routes

Changes in the transport route configuration are not valid until they are distributed and activated. To distribute and activate the changes from the TMS editor of the transport domain controller, choose the *Distribute and Activate* icon and activate, or choose the menu to select *Configuration → Distribute and Activate*. For activation the RFC destinations that have been created during the set up of the transport domain, are used.

To avoid any possible inconsistencies, especially if there are any released transport requests waiting in any import queues, TMS performs a number of consistency checks on the version of the transport routes being activated, for example, whether the settings of the transport control program *tp* are consistent with the transport route configuration. The activation also triggers checks to verify that new transport routes do not affect existing transport requests. If any problems exist, the configuration will not be activated and you will receive an error message indicating the nature of the error.



#### Note:

Once you have configured the transport routes correctly, the only reason that changes would need to be made would relate to a change in your SAP system landscape, for example, adding a new SAP system to the landscape.

In the setup shown in the figure above, the SAP system *PRD* is treated as the production system with a different transport directory. The SAP system *TRN* is used as a training system that needs its own consolidation route.

So the former three system landscape is now extended to a four system landscape - development, quality assurance, pre-production, and production. Plus a training system.

### Configuring the QA Approval Procedure

When a transport request is released and exported from the development system (in the figure above: *S4D*), the import buffer for the suitable consolidation system (for example, *S4Q*) is populated. The import buffer is a list of transport requests waiting for import. Once the

transport request is imported into the consolidation system, the import buffer for all delivery systems (for example, S4P) is populated.

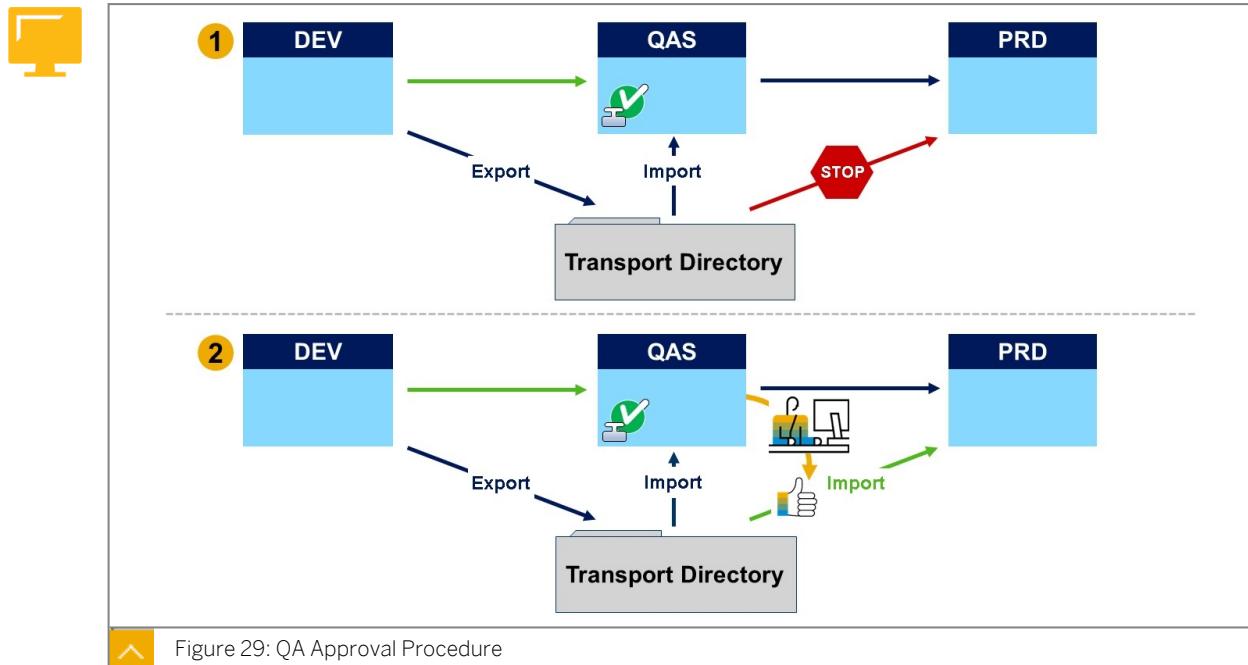
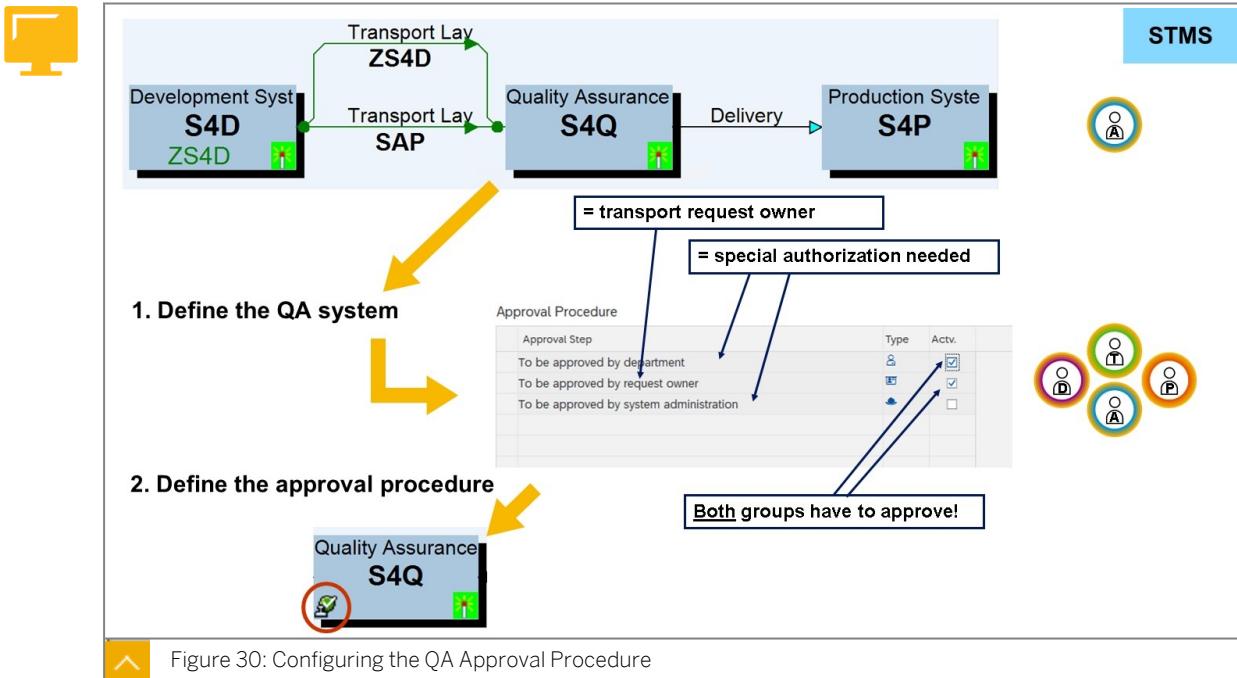


Figure 29: QA Approval Procedure

With the **QA approval procedure**, the buffer of the delivery systems is populated but the entries are flagged as *inactive*. In other words, the transport requests can't be imported until the responsible person flags the transport request as approved, which results in activating the entries in the delivery systems' buffers.

**TMS Quality Assurance** increases the quality and the availability of the production systems by letting you check transport requests in the QA system before they are delivered to subsequent SAP systems. The SAP system for which the QA approval procedure is activated is called the **QA system**. When the QA approval procedure is activated, a transport request will only be imported into the delivery system(s) if all the QA approval steps are processed in the QA system and the request has been approved. When you configure the QA system, you determine how many QA approval steps have to be processed for each transport request.



The figure above shows you how to activate the QA approval process from the graphical editor and how to define the approval steps.



#### Hint:

The authorization object `S_CTS_ADMI` (field `CTS_ADMFCT`) defines whether you belong to the department (value `QTEA`) or to the SAP system administration (value `TADM`). The corresponding authorizations are `S_CTS_QATEST` for the department and `S_CTS_ADMIN` for the SAP system administration.

## Verifying the TMS Setup

Once you have configured the TMS, there are some tests that you can perform to verify that there are no technical problems with the TMS configuration:

### Verifying TMS Setup



- RFC connection test
- Transport directory check
- Transport control program (`tp`) check

As the transport domain controller communicates with other SAP systems in the transport domain using RFCS, you should check the RFC destinations between the SAP systems in the transport domain. To do this, from the *System Overview* screen of the STMS, select an SAP system from the list and use the menu to choose *SAP System → Check → Connection Test*. This verifies communications in both directions, to and from the transport domain controller.

*Exports* physically write files into the file system level. *Imports* read and update these files. Therefore, you should verify that the transport directory is available and has the appropriate file system level permissions. To do this, from the *System Overview* screen of the STMS, use the menu to choose *SAP System → Check → Transport Directory*. This action creates, reads,

and deletes a file to the most important subdirectories of the transport directory and returns a report on the success or failure of all subdirectories. If you receive any errors, the appropriate corrections need to be made at the file system level.

Because exports and imports are physically executed by the operating system level program *tp*, you should verify that the *tp* program and its parameter configuration are consistent. To do this, from the *System Overview* screen of the STMS, use the menu to choose *SAP System → Check → Transport Tool*. This executes tests verifying the *tp* interface, the transport profile, the RFC destinations, as well as the connection to the databases, and returns a report of the results.

## SAP Notes for the Setup of the Transport Management System

The following SAP Notes might be helpful:

- SAP Note [1536753](#) – Configuration of the Transport Management System
- SAP Note [556734](#) – FAQ Transport: Setup and further information
- SAP Note [28781](#) – Central transport directory NT/UNIX
- SAP Note [83327](#) – Setting up transport system in heterogeneous SAP system group
- SAP Note [2506805](#) – Transport Directory DIR\_TRANS
- SAP Note [2701630](#) – Best practices for configuring transport directory
- SAP Note [1541076](#) – Preliminary checks and troubleshooting for transport issues
- SAP Note [1090842](#) – Cross-Release Transports
- SAP Note [761637](#) – Logon restrictions prevent TMSADM logon
- SAP Note [1568362](#) – TMSADM password change
- SAP Note [2816097](#) – User TMSADM is constantly locked
- SAP Note [2493023](#) – TMSADM Problems: Required notes
- SAP Note [2920928](#) – Transport Groups in STMS
- SAP Note [3201742](#) – How to switch to hierarchical view for transport routes



## LESSON SUMMARY

You should now be able to:

- List the main steps to set up a system landscape
- Explain the use of the transport directory
- Define the terms transport domain, system landscape, and transport group
- Create a transport domain and add additional systems to it
- Outline how to configure *tp* from within the Transport Management System
- Configure transport routes
- Configure the quality assurance approval procedure
- Check the setup of the Transport Management System (TMS)



## Configuring Extended Transport Control

### LESSON OVERVIEW

In this lesson, you will learn about extended transport control. This covers the use of both transport target groups and client dependent transport routes. Transports between transport groups and between different transport domains are also discussed.

### Business Example

As an SAP system administrator, you are asked to ensure that certain development objects or customizing objects can be imported into more than one subsequent SAP system or into more than one specific client of a Quality Assurance (QA) system.



### LESSON OBJECTIVES

After completing this lesson, you will be able to:

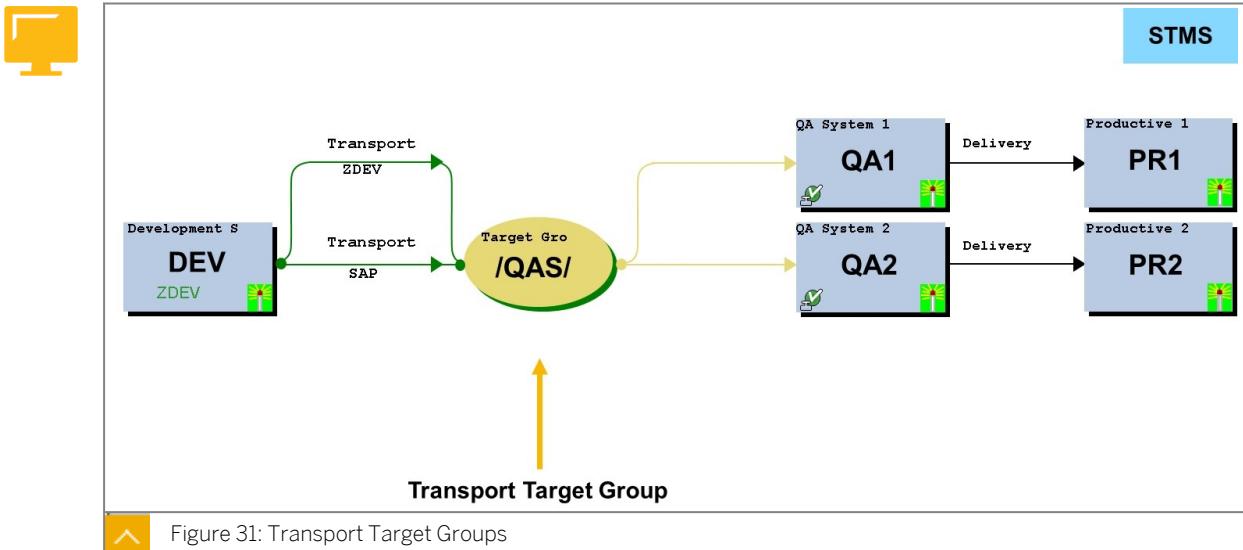
- Explain the use of transport target groups and client-specific transport routes
- Describe a suitable system landscape for parallel project development and maintenance
- Outline the transport process between different transport groups and transport domains

### Extended Transport Control

#### Transport Target Groups

Using SAP standard transport control, it is not possible to transport a transport request from one development system to multiple quality assurance systems in an easy way. The reason for that is that it is not possible to create two consolidation routes with the same transport layer from one and the same SAP system. As every repository object is assigned (via a package) to a certain transport layer, it follows that a transport request seems to have only one target system.

The solution for the problem stated above is to create a *transport target group*, which may lead to one or more SAP systems.



To create a transport target group, use the menu path *Overview → Transport Routes* in transaction STMS and from there (in change mode) choose the menu path *Edit → Transport Target Group → Create*.



#### Note:

The name of the transport target group must start and end with “/”.

When a transport request, which has a transport target group as target, is released, it will fill the import queue(s) of all SAP systems in this transport target group. In the figure above, all released transport requests that contain objects pointing to the transport layer ZDEV are ready for import into both the QA1 system and the QA2 system.

#### Client Transport Control

Because some SAP system landscapes contain multiple clients in the development and quality assurance systems, it is a challenge for the transport administrator to maintain consistent (client-specific) customizing across the landscape. Different SAP systems and different clients within an SAP system may need to receive changes at different times, depending on quality assurance approval and acceptance procedures. Communication errors between customizing project leaders and transport administrators can inadvertently cause inconsistencies in the configuration settings of certain clients.

Because the majority of customizing is client-specific, during the scheduling of an import process, the import scheduler prompts the transport administrator for a target client. The administrator would need to manually schedule the imports for the different clients, possibly based on instructions from customizing project leaders, and would also have to keep track of which transport requests have and have not been imported into which clients.

TMS offers the **Extended Transport Control** (also known as *Client Transport Control (CTC)*), where the administrator can automate the process by:

- Client-specific transport target groups
- Client-specific consolidation routes
- Client-specific delivery routes.

#### *Client-specific transport targets*

The transport targets of consolidation and delivery routes do not just specify an SAP system, they also specify a client. Client-specific transport targets are entered in the form: <SID>.<client> (for example, **S4Q.100**). Transport target groups combine several client-specific transport targets under a symbolic name. You can specify transport target groups when you define consolidation routes or delivery routes. To differentiate them from traditional transport targets, transport target groups must start and end their names with "/" (for example, **/TTG01/**).



#### Note:

In the context of cross-client transport routes, transport target groups have already been discussed in the previous subsection. The concept of transport target groups and client-specific transport routes can easily be combined.

#### *Client-specific consolidation routes*

For each transport layer, the consolidation routes determine where changes made in the SAP system are transported to after the transport request has been released. If you have activated extended transport control, then the transport target can be a specific client in a target system or a transport target group. If you do not activate extended transport control, the transport administrator has to specify the correct target client at the time of import.

#### *Client-Specific Delivery Routes*

Delivery routes determine whether transport requests are to be flagged for import into subsequent SAP systems/clients, after they have been imported into an SAP system. If you have activated extended transport control, then you can set the delivery routes as client-specific. This makes it possible to supply several clients in one SAP system in sequence. You can also specify a target group as the target of a delivery route.

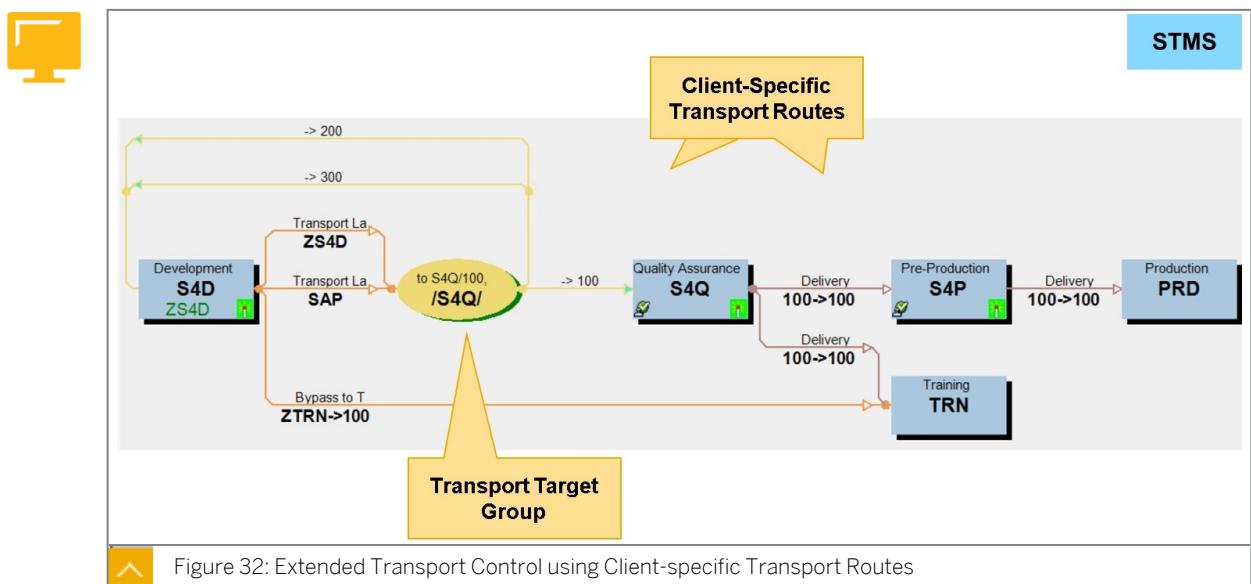


Figure 32: Extended Transport Control using Client-specific Transport Routes

In the figure, objects pointing to the transport layer ZS4D will be imported (using transport target group /S4Q/) into S4Q client 100, and S4D client 200 (golden client), and S4D client 300 (client for functional tests).

After QA approval in the S4Q system (client 100), the objects are forwarded to S4P client 100, and TRN client 100.

After QA approval in the S4P system (client 100), the objects are forwarded to PRD client 100.

Extended transport control makes daily transport tasks easier and increases security.

Extended transport control also reduces the need for communication between project leaders and SAP system administrators, because the transport routes can now be configured completely. No additional details about the target client need to be given at the time of import.

To take advantage of this function, in the transport program profile, you must set the *tp* parameter *CTC* (Client Transport Control) to *TRUE* (value of **1**). The default value is *FALSE* (value of **0**), which deactivates the extended transport control.



**Note:**

When inserting a client-specific transport target group, the *tp* parameter *CTC* is set to **1** automatically.



**Caution:**

You can use either the normal system-to-system transport or client-specific transport routes, but not a mixture of both types of connections in the same SAP system landscape. When using client-specific transport routes, you must specify the target client or clients (for consolidation routes) / the source client and the target client or clients (for delivery routes) when defining the transport route.

The **standard transport layer** (default: *Z<SID>*) determines the default transport target of the transport requests. When you use extended transport control, you can set a different standard transport layer for individual clients in the SAP system. This means that you can forward customizing requests from different clients to different transport targets. The client-specific standard transport layer is also the default transport layer for new packages that have been created in a client. If you accept this default, then the cross-client objects that have been created in cross-client customizing are transported along the same route as the corresponding client-specific customizing.



**Note:**

This scenario would contradict our recommendation that all customizing/development changes originate in one single client. Client-specific transport layers should only be used under certain circumstances, for example, when you have multiple quality assurance systems leading to multiple production systems as shown in the figure “Transport Target Groups”.

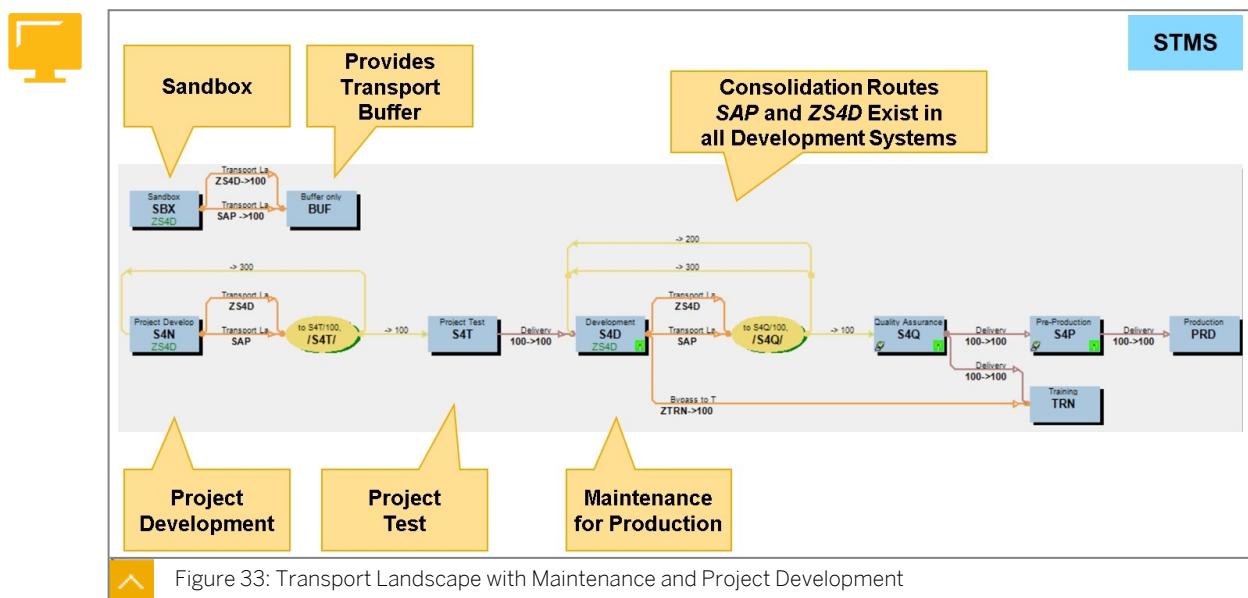
## Project Development and Maintenance

When developing in large projects, it can be necessary to have two development and two quality assurance systems:

1. One development system and one quality assurance systems is used for daily error fixing, small developments, minor customizing changes, that is, a system for maintaining the production system.
2. A different development system and a different quality assurance (or test) system is used for performing a large customer development. This can be necessary because this large

customer development would take a long time and would change the development system in a way that it can't be used as a maintenance system for the production system any longer. Some customers call very large customer development projects a "release", which, in this context, is not an SAP term.

Changes in the maintenance development system have to be replicated (not transported) in the project development system. If the large customer development project is ready, it has to be forwarded into the maintenance development system. Here, the objects from the large customer development project may need to be re-packed into new transport requests to be transported to the quality assurance system.



In the scenario in the figure above and in this course, S4N (project development) and S4T (project test) are used for developing large customer projects. S4D and S4Q and S4P are used for maintaining PRD on a daily basis.

In the figure, there is no delivery route from S4D to S4Q. This means that the objects from the large customer development project have to be re-packed into new transport requests in the S4D system. The alternative would be to create a delivery route from S4D to S4Q and just forward the transport requests from the large customer development project without changing them.

The transport layer SAP originates in both development systems. This means that you change SAP standard objects in both development systems. In addition, there is only one transport layer for own development, ZS4D. It is used by the consolidation route that originates both in S4N and S4D. Otherwise, there would be problems when re-packing the own developed objects from S4N in S4D. Other options include:

- Two transport layers, for example, ZS4N and ZS4D, both used in S4N and S4D.
- Only transport layer ZS4N in S4N and one transport layer ZS4D in S4D. In this case, the customer packages in S4N and S4D would have to point to the corresponding, different transport layers.

SAP system SBX is the sandbox system. It is used to test and change customizing and development objects apart from the real SAP systems. This can be used in the context of large development projects or SAP system upgrade projects.

The corresponding consolidation routes in SBX and the SAP system *BUF* are needed in order to export transport requests from SBX. This is needed if these transport requests should be used for an import in any other SAP system. *BUF* will remain as a virtual system – there will never be an SAP system installed named *BUF*.

## Transport Between Transport Groups and Transport Domains

If you configure different transport groups in your SAP system landscape, there will be the need to perform transports between these different transport groups. If there is more than one transport domain in your company, there might be the need to perform a transport from one transport domain to another domain.

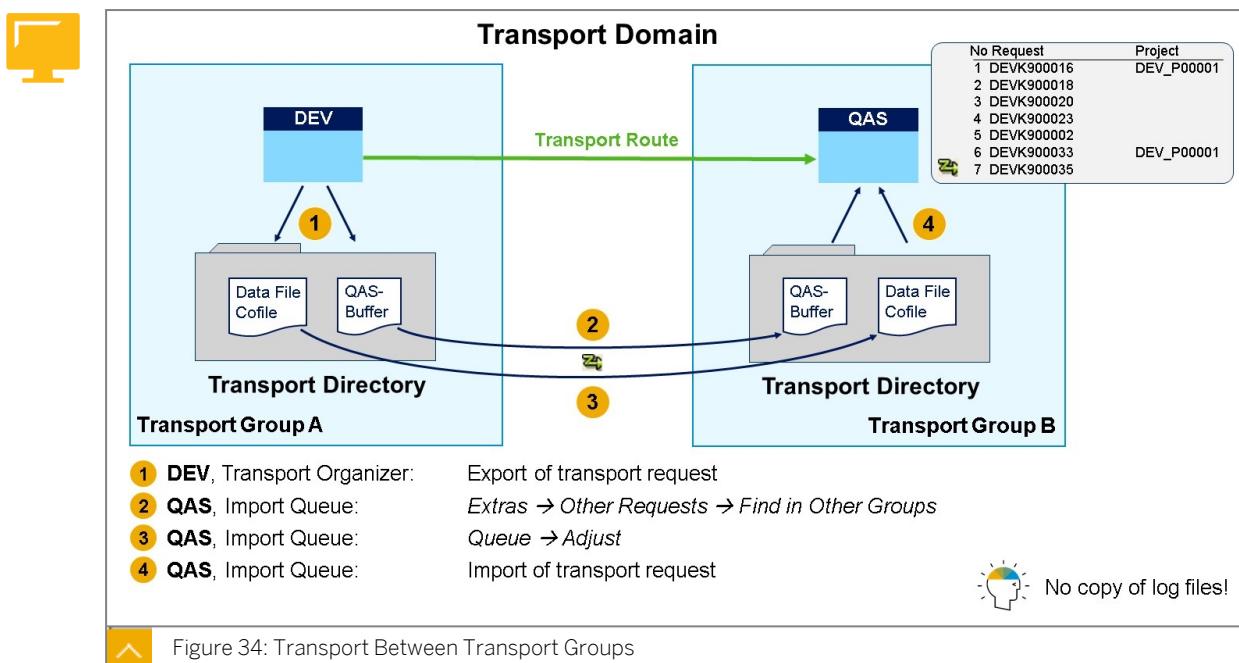
### Transports Between Different Transport Groups

Usually all the SAP systems in a transport domain share the same common transport directory. There are situations, however, in which separate transport directories are set up for different parts of the same domain.

Multiple transport directories are used, for example, when:

- An SAP system is connected to the domain through a slow or expensive network connection.
- Strict security measures prevent allowing direct file system access from other SAP systems.
- Dissimilar hardware platforms prevent the use of a common transport directory.

Within a transport domain, SAP systems that share a common transport directory form a *transport group*. The TMS supports transports between transport groups. The following figure outlines the procedure on how to transport requests between different transport groups.



After a transport request has been released from DEV, the request is marked for import into the target system. Be aware that this happens in the transport directory of the DEV system (step 1). If the source and target systems are in different transport groups, however, the import queue of the target system must be adjusted from the *Import Queue* screen in the target system group choosing *Extras → Other Requests → Find in Other Groups* (this

corresponds to step 2 in the figure above). TMS searches for requests for the selected SAP system in the import buffers (on file system level) of all transport groups in the transport domain, and (depending on the selected options) transfers the data files and cofiles belonging to the transport requests (step 3). Before the transfer of the data file, the transport request is marked in the import queue with a “lightning” icon that disappears after adjusting the import queue of the target system.



**Note:**

You can schedule program *RSTMSTIQ* periodically in the target systems to automatically adjust the queue. For more information, see SAP Note [2030463](#) – *Automatic adjustment of import queue with RSTMSTIQ*.



**Note:**

There are limitations to transporting between different transport groups:

- The transport logs displayed are specific to the transport group of the SAP system you are using, that is, they are not copied to the other transport directory.
- Transports displayed in the *Transport Organizer* are also specific to the transport group of the SAP system you are using.

If you have configured multiple transport domains and want to perform transports between SAP systems in different domains, you can use *domain links* to link the two domains or external SAP systems to transport between different transport domains.

### Transport Between Different Transport Domains



- Domain link (needs a permanent network connection)
- External systems

For a **domain link**, there must be a permanent network connection between the SAP systems in the two domains, similar to the connection between SAP systems within the same transport domain.

Linking two transport domains with a domain link involves two steps:

1. One transport domain controller has to request a link between the two transport domains.
2. The transport domain controller of the other transport domain has to confirm the link between the two transport domains.

To request a transport domain link, enter the *System Overview* area in transaction **STMS**. Here, choose the menu path *SAP System* → *Create* → *Domain Link*. Enter the SAP *system name*, *host name*, and *instance number* of the transport domain controller you want to link to, and then confirm your entries. The SAP system generates the required RFC destinations automatically and sends the address data of the local transport domain controller to the transport domain controller of the other transport domain.

To confirm a link between two transport domains, enter the *System Overview* area in transaction **STMS**. Position the cursor on the line for the transport domain controller system

where you have requested the domain link and choose *SAP System → Approve* from the menu. Confirm the prompt and distribute the configuration.

After you have established a domain link:

- You can perform transports between SAP systems in different transport domains in the same way as you make transports between SAP systems in different transport groups; RFC is used to transfer transport files between the transport directories involved.
- You can display transport logs of SAP systems in the other transport domain.



Note:

Transport domains are independent administrative units. Transport routes are not distributed between transport domains. You can, however, configure a transport route between SAP systems in different domains, but must configure it twice, once in each transport domain. The transport domains that are linked by a domain link must have different names.



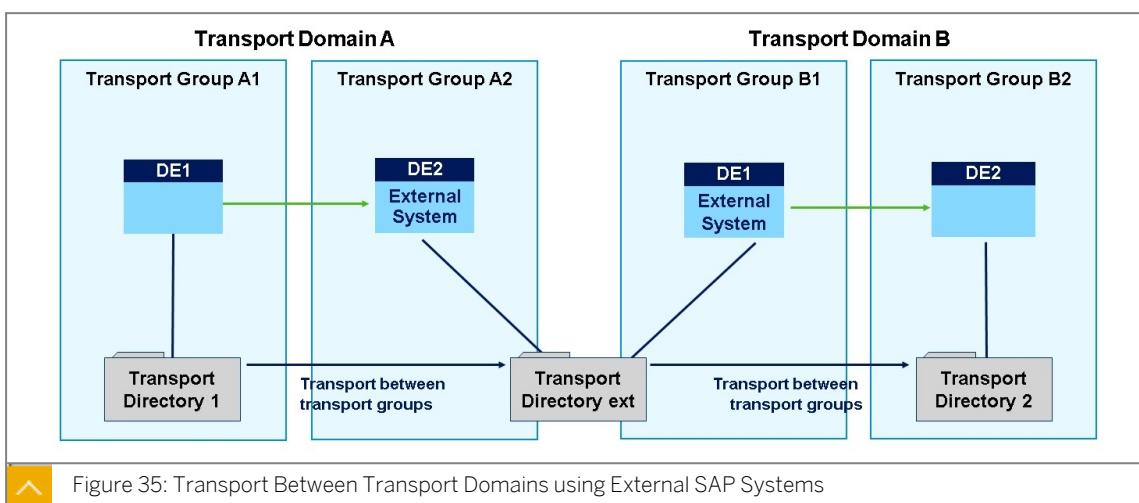
Hint:

The administration of the TMS configuration is simpler if the *TMSADM* password remains the same across the domains. As older releases do not implement a flexible standard password, domain links to older system landscapes are either not possible or you can choose one of the two recognized standard passwords. See also SAP Notes [1414256](#) – *Changing TMSADM password is too complex* and [761637](#) – *Logon restrictions prevent TMSADM logon*.

If you can't operate a permanent RFC connection between SAP systems in the two transport domains, you can use **external systems** to perform transports between the two domains.

External systems are like virtual systems. However, for this type of SAP system, a separate transport directory is also defined. This directory can either be located on a disk partition that can be accessed by an SAP system in the other domain, or it is on a replaceable data medium, for example a DVD, memory stick or portable hard disk.

The idea behind external systems is to reduce the problem of transporting between **different** transport domains by transporting between different transport groups in the **same** transport domain.



In the figure above, you want to transport between the SAP systems *DE1* in transport domain *A* and *DE2* in transport domain *B*. In transport domain *A*, you create an external system called **DE2** (pointing to the transport directory *Transport Directory ext*). In transport domain *B*, you create an external system called **DE1** (pointing to the transport directory *Transport Directory ext* as well). In addition, in both transport domains you need to define a transport route between these two SAP systems. You must also configure the host of the SAP system *DE1* so that it can access the directory *Transport Directory ext*. In the same way, the SAP system *DE2* in transport domain *B* must be able to access the directory *Transport Directory ext*.

After you export a transport request from *DE1* to *Transport Directory 1*, you perform a transport between transport groups in transport domain *A*. As a result, data files, cofiles and the buffer entries for system *DE2* exist in *Transport Directory ext*. Next, you perform a transport between transport groups in transport domain *B* so that all necessary data is sent to the transport directory *Transport Directory 2*. Then you can perform the import into *DE2*.

To configure an external system, log on to the SAP system that is the transport domain controller and call transaction **STMS**. In the *System Overview* area, choose the menu path *SAP System → Create → External System*. Enter the *System ID* (together with a *Description*) and the *Path* and a *Description* of the transport directory. You also need both to enter the communication system. The transport domain controller is proposed as the communication system. Save your entries and confirm the prompt for the distribution / activation of the changes.



## LESSON SUMMARY

You should now be able to:

- Explain the use of transport target groups and client-specific transport routes
- Describe a suitable system landscape for parallel project development and maintenance
- Outline the transport process between different transport groups and transport domains



# Learning Assessment

1. A transport domain always consists of only one transport directory.

*Determine whether this statement is true or false.*

- True
- False

2. When setting up the Transport Management System (TMS), some steps must be performed in a certain sequence. Bring the following activities into the right sequence, regarding the setup of TMS.

*Arrange these steps into the correct sequence.*

- Configure the Quality Assurance (QA) procedure
- Create a transport directory
- Define the SAP systems of the transport domain
- Create a transport domain
- Configure the transport routes

3. Which actions are performed automatically when creating a new transport domain in Transport Management System (transaction STMS)?

*Choose the correct answers.*

- A The SAP system is assigned to the transport domain
- B The standard transport layer is created
- C A transport group is created
- D The Quality Assurance (QA) procedure is set up
- E User TMSADM is created in client 000

4. You want transport requests that are exported from the development system to be added to the import queues of multiple quality assurance systems simultaneously. Which technique should be used for this?

*Choose the correct answer.*

- A Transport target groups
- B Client-specific transport routes
- C Quality Assurance (QA) procedure
- D External systems

# Learning Assessment - Answers

1. A transport domain always consists of only one transport directory.

*Determine whether this statement is true or false.*

True

False

You are correct! A transport domain can consist of several transport directories, whereby each transport directory is part of exactly one transport group.

2. When setting up the Transport Management System (TMS), some steps must be performed in a certain sequence. Bring the following activities into the right sequence, regarding the setup of TMS.

*Arrange these steps into the correct sequence.*

**5** Configure the Quality Assurance (QA) procedure

**1** Create a transport directory

**3** Define the SAP systems of the transport domain

**2** Create a transport domain

**4** Configure the transport routes

You are correct! First, you need to create a transport directory, and then, create a transport domain. You then need to define the SAP systems of the transport domain, configure the transport routes, and finally configure the Quality Assurance (QA) procedure (optional).

3. Which actions are performed automatically when creating a new transport domain in Transport Management System (transaction STMS)?

*Choose the correct answers.*

- A The SAP system is assigned to the transport domain
- B The standard transport layer is created
- C A transport group is created
- D The Quality Assurance (QA) procedure is set up
- E User TMSADM is created in client 000

You are correct! When creating a new transport domain in Transport Management System (transaction STMS), the SAP system is assigned to the transport domain, a transport group is created, and user TMSADM is created in client 000. The standard transport layer is defined when defining the transport routes. The Quality Assurance (QA) procedure must be set up manually (if needed).

4. You want transport requests that are exported from the development system to be added to the import queues of multiple quality assurance systems simultaneously. Which technique should be used for this?

*Choose the correct answer.*

- A Transport target groups
- B Client-specific transport routes
- C Quality Assurance (QA) procedure
- D External systems

You are correct! You should use transport target groups for this. Client-specific transport routes allow you to define the transport routes per client. The Quality Assurance (QA) procedure can prevent untested transport requests to be imported into the production system. External systems are used to transport via multiple transport directories.

# UNIT 3

# Creating and Exporting Transport Requests

## Lesson 1

Customizing and Customizing Projects

67

## Lesson 2

Transport Requests for Customizing

77

## Lesson 3

Customizing Procedure

85

## Lesson 4

Transport Requests for Development

95

## Lesson 5

Customer Development

107

## Lesson 6

Modifying SAP Objects

115

## UNIT OBJECTIVES

- Explain the meaning of Customizing
- Illustrate the use of Implementation Guides for Customizing
- Outline the Role of SAP Solution Manager in implementation projects
- Create transport requests for Customizing
- Explain the use of CTS projects
- Describe the content of a transport request for Customizing changes
- Describe the Customizing procedure from the creation to the release of a transport request
- Explain the difference between a workbench request and a customizing request

- List Best Practices for Customizing
- List Best Practices for Development
- List differences between Customizing and development
- Outline the idea of SAP Software Change Registration
- Outline the idea of naming concepts and the repository object directory
- Explain the concept of packages
- Explain object locking and version management
- Describe the process of releasing a transport request that contains repository objects
- List selected Transport Organizer tools and explain their use
- List authorizations for software logistics
- Explain the difference between an original and a copy and the consequences for transport tasks
- Explain the process flow for modifications

# Unit 3

## Lesson 1

# Customizing and Customizing Projects

## LESSON OVERVIEW

In this lesson, you will get an overview on customizing of SAP systems to meet individual business requirements. You will also learn how the *SAP Implementation Guide* (IMG) and the SAP Solution Manager system can be used in the customizing process.

## Business Example

Customizing settings adapt SAP systems to meet a company's business requirements by defining the logic paths that the application programs follow. As a TMS administrator, you want to learn about the roles and responsibilities of the team that will do the customizing. This is necessary for supporting the customizing teams during the complete life cycle of the project. With that knowledge, you can recommend the right tools for the efficient realization of the customizing project.



## LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Explain the meaning of Customizing
- Illustrate the use of Implementation Guides for Customizing
- Outline the Role of SAP Solution Manager in implementation projects

## Definition of Customizing

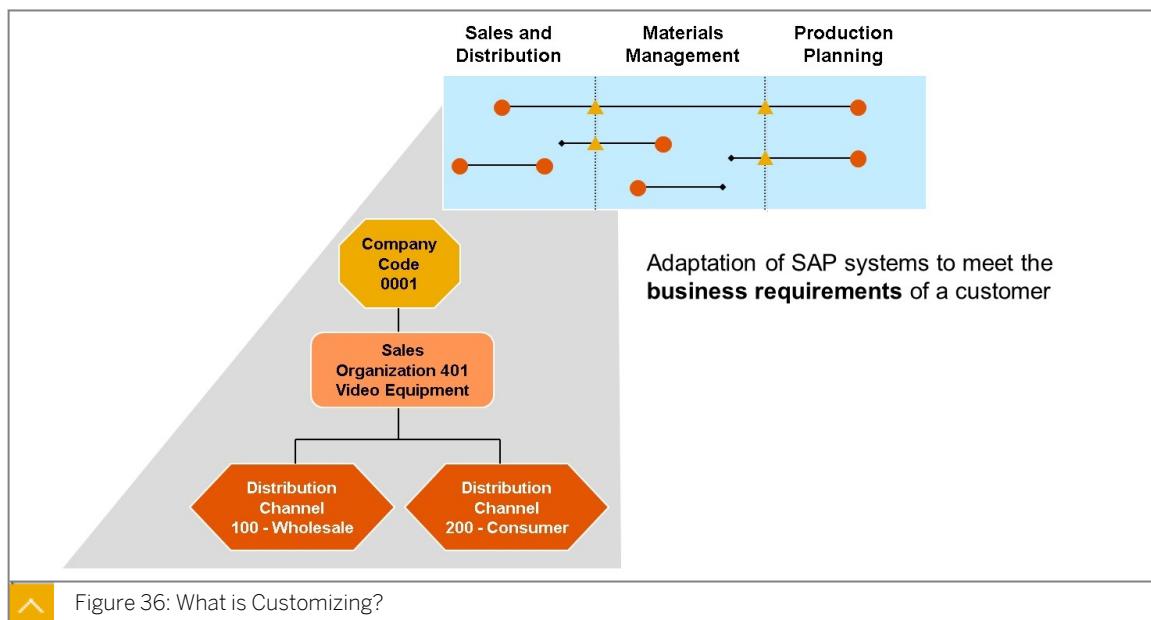


Figure 36: What is Customizing?

The term **customizing** describes the overall procedure to set up one or more SAP systems. In particular, the aim of the procedure is to:

- Adapt the non-company specific and industry-specific functions shipped in an SAP system to the specific business requirements of your company
- Enhance SAP functions in your company
- Implement SAP solutions in your company in a fast, reliable, and cost-effective manner.

In short, customizing adapts the SAP software to meet a company's individual business requirements by setting up the business transactions the company requires in the SAP system.

To show the kinds of business transactions, an enterprise may need to customize in an SAP system, consider the example of an international company that sells video equipment wholesale. To enable customers at a particular outlet to place a wholesale order for video equipment, the company must customize the SAP system to recognize the company's structure. This could include:

- Defining a sales organization
- Defining distribution channels
- Assigning distribution channels to the sales organization

The example from the figure "What is Customizing?" describes the definition of a sales organization and the distribution channels. This customizing is an example of the definition of the individual organizational structure of a company. This kind of customizing is typically stored in client-specific customizing tables. In general, this customizing is also known as a *Client-specific customizing*. Client-specific customizing affects only one particular client.

Another type of customizing is customizing that adjusts the execution of programs. This customizing is mostly also client-specific, but some customizing activities insert entries in cross-client tables or generate some repository objects (tables or programs). In general, this customizing is known as *cross-client customizing*. Cross-client customizing affects all clients in the SAP system.

### **Customizing from a Technical Perspective**

The following figure shows the technical representation of customizing in the SAP system.

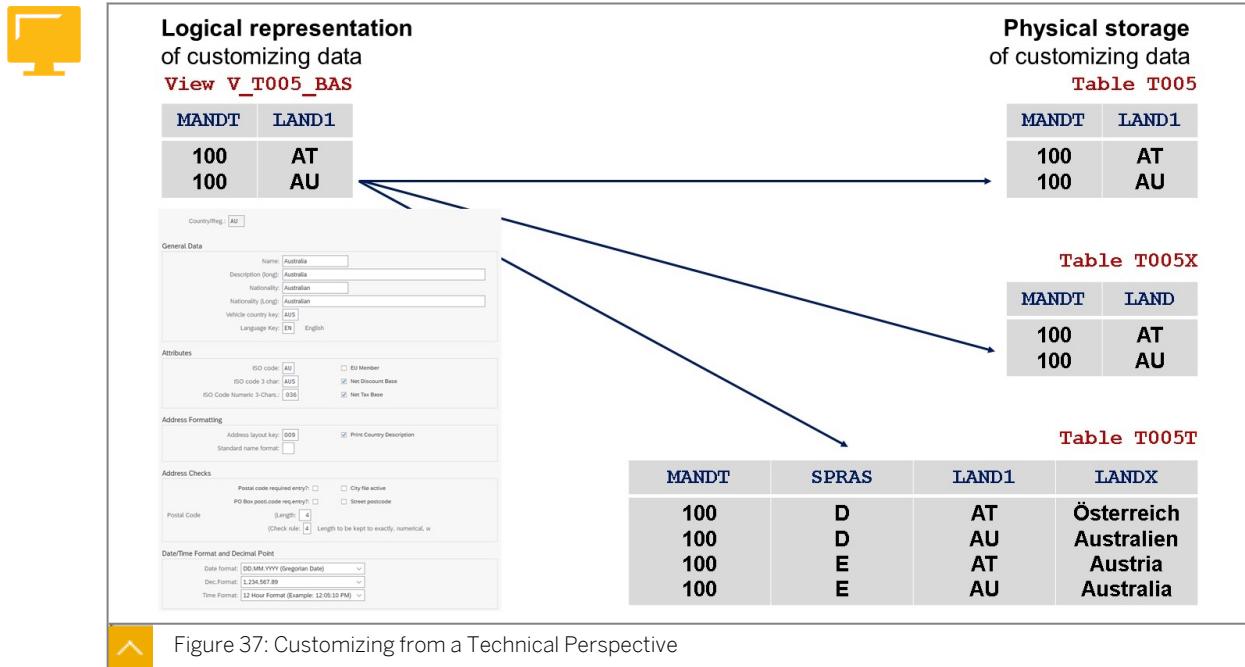


Figure 37: Customizing from a Technical Perspective

Customizing activities typically create or change entries in multiple tables. This is often done through table views.

A **table view** is a virtual table, which presents data that is physically stored in one or multiple tables. You can compare a table view with a view that is used in an SQL database. The different customizing tables of a table view can be related to each other by foreign keys.

The table entries created with the help of table views are identified by the key fields of the respective table.

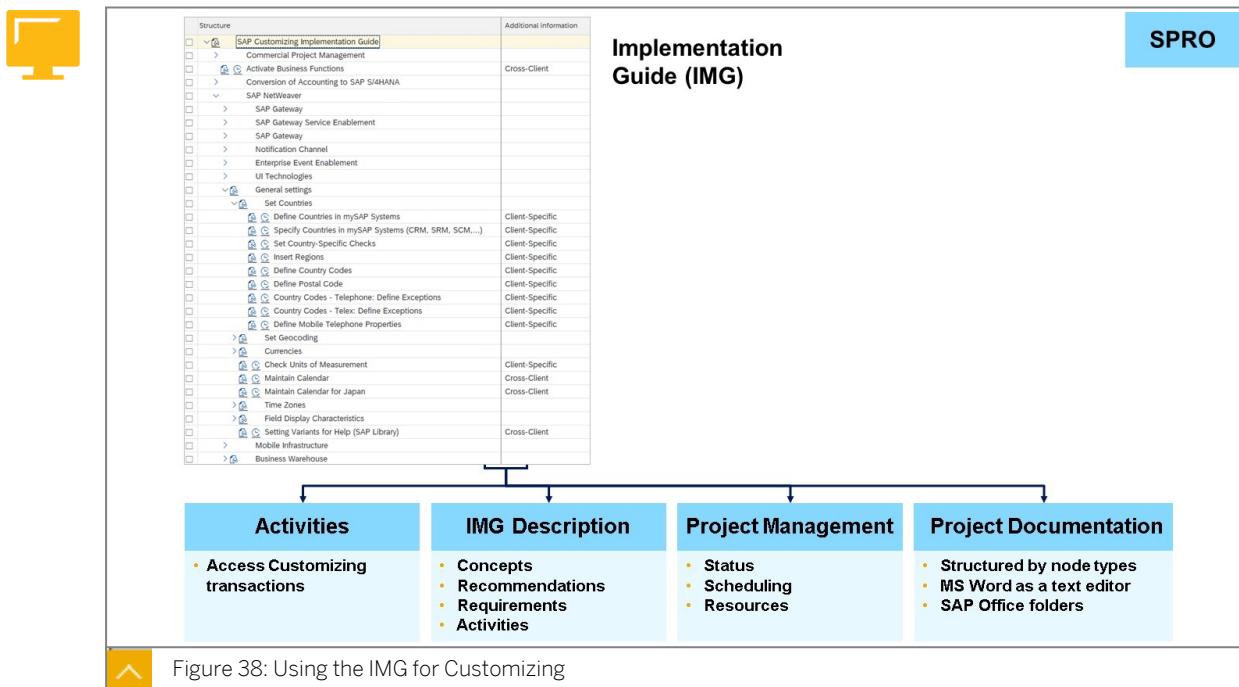
In the figure above, the table view for maintenance *Global Parameters of Countries (New Dimension Systems)* is shown. The country parameters are maintained with the help of view *V\_T005\_BAS*. All the different attributes for a country, for example, *Name*, *Nationality*, *ISO code*, *Date format*, and *Decimal format* are maintained. These parameters are stored physically in different tables. For example, general country values are stored in table *T005*, the country names in different languages are stored in table *T005T* and the decimal format and date format are stored in table *T005X*.

The customizing team does not need to care about the technical storage of the customizing, they only have to maintain the information in the table view.

The table maintenance can be accessed directly with the IMG or even with a generic table maintenance transaction (transaction SM30).

## Customizing Using the SAP Implementation Guide

An SAP system provides a huge variety of different business processes that can be adapted to the individual needs of a customer by customizing. An SAP S/4HANA Server 2021 for example contains many tens of thousands of customizing tables. The customer is guided through this customizing with the help of the *Implementation guide* (IMG). An SAP system is delivered with a complete IMG for all processes available in the SAP system. This IMG is known as the **SAP Reference Implementation Guide**. The IMG can be accessed using transaction SPRO.



The IMG provides a hierarchical list with customizing activities in that SAP system. This customizing is grouped by applications and is ordered in the sequence in which the individual activities have to be performed.

The IMG doesn't just show and group all customizing activities. The IMG also provides documentation on each customizing activity. This documentation explains the customizing activity that can be executed in this IMG node. Also each customizing activity is classified into different areas. For example, if you are interested in which customizing activity is cross-client, you can use SAP Reference IMG in transaction SPRO and choose the menu path *Additional Information → Technical Data → Client Dependence*. There is also a classification on *language dependency* and on *transport type* that can be displayed.

As well as these tools for accessing customizing activities, displaying customizing documentation, and displaying customizing classification, the IMG also provides tools for project management and project documentation. These tools support the project administration and project control out of the IMG.

The SAP Reference IMG is split into different areas.

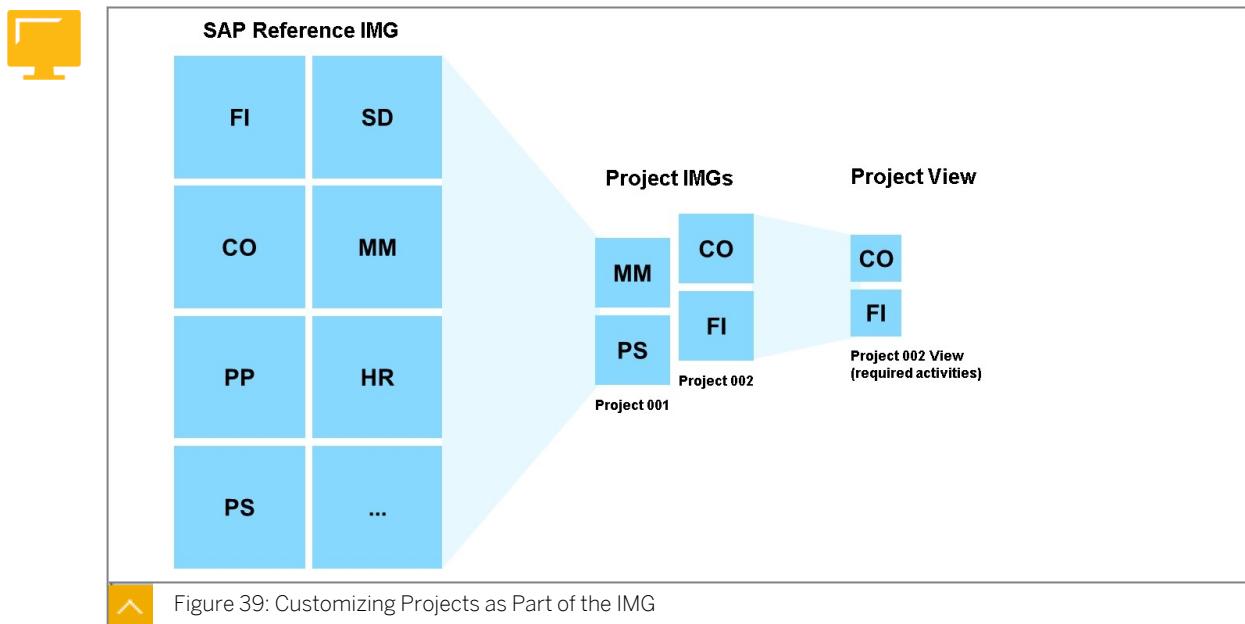


Figure 39: Customizing Projects as Part of the IMG

If a customer is setting up a new project for adoption of the SAP system, only some areas are in the scope of this adoption project. For this, the customer can use a **Project IMG**. A Project IMG is the implementation guide for a specific customizing project. It contains all the customizing activities which have to be performed in that project. You can determine the project scope of a Project IMG by countries, components, or customizing activities. A Project IMG is based on the *SAP Reference IMG*, which can, for example, be reduced to the project scope by selecting countries and components.

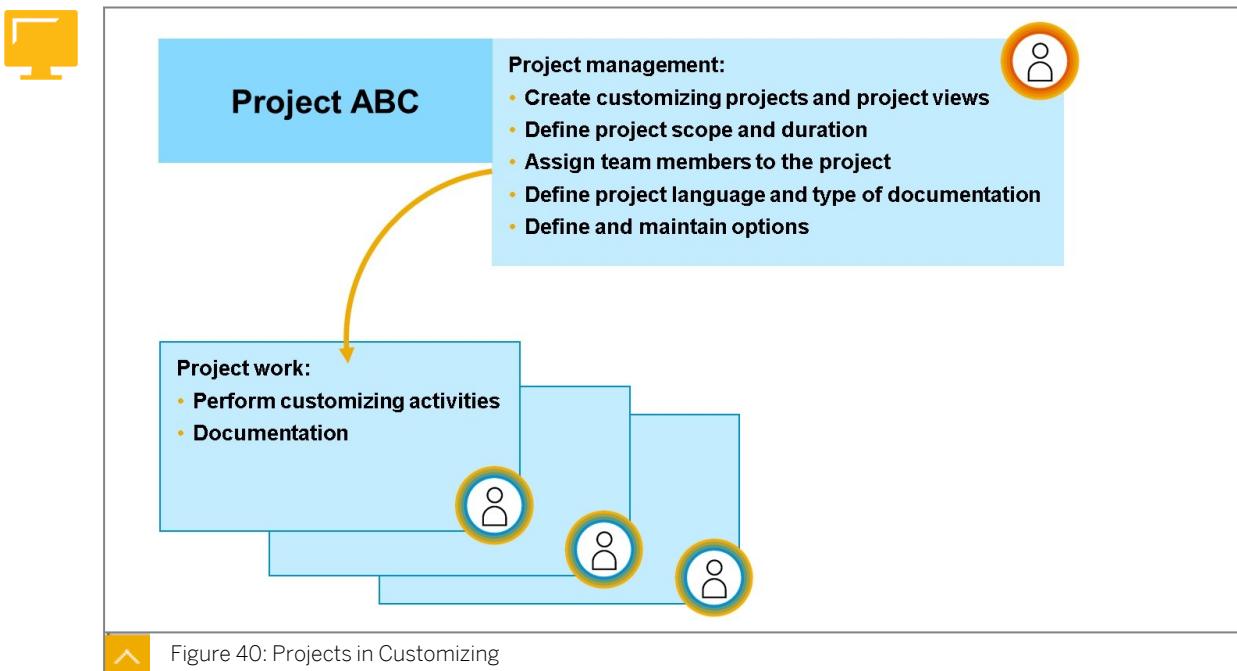
When defining a customizing project, customers select the functions they require in the different SAP areas and components. These functions are used to generate the Project IMG. All relevant customizing transactions, project-specific or cross-project documentation, and project-management information are then located in the subsets of the *SAP Reference IMG*, the Project IMGs. All customizing activities should be done within the Project IMG.

The Project IMGs can be changed at any time through regeneration. The respective previous IMG is then overwritten. However, status information and project documentation are not lost.

Within a Project IMG it is possible to generate *project views* that restrict the extent of customizing, for example, to different levels of necessary activities. So it is possible to show within one project view only the activities that are classified as mandatory and in a second project view the activities that are classified as optional.

The Project IMGs are cross-client, that means the Project IMG is visible in all clients.

To access the IMG project definition you can use the transaction code `SPRO_ADMIN`.



Depending on the scope of a customizing project, a lot of different customizing activities have to be done. These customizing activities are usually done by different project members that are working on the implementation of a specific area. To make sure that everyone is working efficiently together, customizing projects need a clear organizational structure and responsibilities.

The customizing and transport tools support the task sharing between the project leader (indicated by the icon in the upper right on the figure above) and the project members (indicated by the three icons in the lower left). The customizing process is based on this with clearly-defined user roles. Each role has its specific areas of responsibility. This structures and organizes the implementation activities:

The project leader is responsible for defining and managing the project. If IMG projects are used, this includes:

- Create the customizing IMG projects and the project views
- Define the project scope and duration of the IMG project
- Assign team members to the IMG project (define who is supposed to check the customizing settings and perform changes)
- Define the IMG project language and type of project documentation
- Define and maintain IMG project options

As you can see, with the use of IMG projects, many general project administration tasks can be done inside the SAP system.

The customizing team members:

- Perform customizing activities within the Project IMG
- Keep records inside the Project IMG

Any number of project team members can be assigned to each project. The project leader makes this assignment. The assigned project team members and new project team member assignments can be managed in transaction `SPRO_ADMIN`.

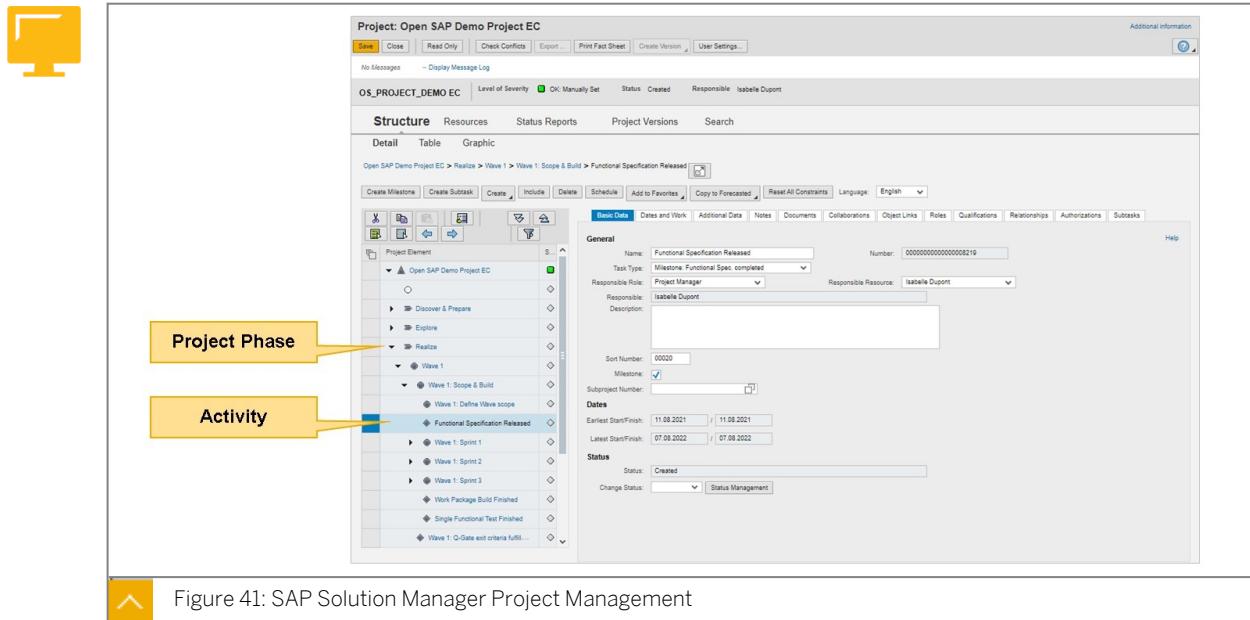
## SAP Solution Manager in Implementation Projects

As you have seen in this lesson so far, the IMG offers a lot of tools and documentations that can be used for supporting the implementation process. But if you set up and plan an implementation project, you will have to manage all the different project phases which are:

- Project preparation
- Business blueprint
- Realization
- Final preparation
- Go-live support

The process of customizing the SAP system can be controlled and accelerated using the integration of *SAP Portfolio and Project Management* into SAP Solution Manager.

Consequently, the SAP Solution Manager system can act as the platform for the entire project team.



SAP Solution Manager provides the following features for implementation projects:

- *Project Administration/Project Definition* supports you initially in setting up your project during project preparation and enables you to carry out major administrative tasks such as the definition of project standards during the entire project.
- In SAP Solution Manager, you can centrally define and manage your SAP system landscape, relevant for your implementation or template projects, which allows the interaction and navigation into a related SAP system landscape; for example, during configuration and testing.
- The definition and documentation of your project scope in business blueprint is accelerated through the *Business Process Repository* providing latest available implementation contents. The purpose of the business blueprint is to document in detail

the scope of business scenarios, business processes, and process steps of your implementation project from a business and technical perspective.

- During configuration, you have to configure the business requirements specified in the business blueprint phase in the related SAP system landscape. SAP Solution Manager as central platform for your project team provides the access to the related project implementation guides (IMGs) and provides integrated use of other customizing technologies such as *Business Configuration Sets* (BC Sets) and customizing distribution.
- The *customizing distribution* used with the SAP Solution Manager helps to centrally control customizing changes across the related SAP system landscape of your implementation project.
- *Testing functions* support test coordinators to centrally create test cases and test plans reflecting sequence and integration tests. Testers can centrally execute their test packages, also leveraging existing test technologies for automated testing such as eCATT.
- The latest available roadmaps representing the standard SAP implementation methodologies are provided through the *Roadmap Viewer* (<https://go.support.sap.com/roadmapviewer/>). With the SAP Solution Manager system, you can track status, issues and/or upload your project-specific accelerators along the work packages and tasks of your implementation project.

Implementation projects in an SAP system landscape today normally affect more than one SAP system (for example, SAP SCM Server and SAP ECC). During these kind of projects, customizing in more than one SAP systems has to be done. The SAP Solution Manager system supports project maintenance and administration in an SAP system landscape. Implementation projects that affect a special SAP system landscape can be defined centrally for this SAP system landscape and then the project IMGs can be created in the different (managed) SAP systems.

The SAP Solution Manager system provides the tools for supporting customizing projects in an SAP system landscape of multiple SAP systems. This function is extended, for example, with the *Change Request Management*, a collection of tools that enables you to manage your SAP Solution Manager projects from end to end: From change management and project planning, through resource management, to physical transports of changes from the development environment into the productive environment.

SAP Solution Manager tools include the transport tracking and approval tasks that are necessary in a productive system landscape with on-going implementation project and maintenance tasks.

For the implementation of SAP S/4HANA, SAP Activate was introduced. **SAP Activate** methodology replaces ASAP as SAP's implementation methodology for SAP S/4HANA and all other SAP solutions. SAP Activate is a framework consisting of the following closely integrated components:

- *SAP Best Practices* that deliver ready-to-run business processes as well as best practices for migration and integration.
- *Guided Configurations* (both in the cloud and on-premise).
- *SAP Activate methodology* that uses agile techniques to implement the SAP S/4HANA solution in the cloud or on-premise.

### SAP Cloud ALM

Besides SAP Solution Manager, you can also use SAP Cloud ALM to support the implementation of business processes.

SAP Cloud ALM is an offering for Application Lifecycle Management (ALM). It's intended for customers who use solutions provided by SAP, and who do not want to use their own ALM on-premise platform to manage those solutions.

SAP Cloud ALM is designed to support ALM for the Intelligent, Sustainable Enterprise, including SAP S/4HANA (all deployments). It is the next generation ALM platform for SAP customers built on modern cloud technology. For more information visit *SAP Support Portal* area *ALM (Application Lifecycle Management)* → *SAP Cloud ALM*.

SAP Solution Manager follows the maintenance strategy of the SAP Business Suite. SAP Cloud ALM is not dependent on SAP Business Suite, allowing mainstream maintenance beyond 2030. SAP plans no new on-premise products in the ALM portfolio and no further releases of SAP Solution Manager after SAP Solution Manager 7.2. For more information, see SAP Note [3255311](#) – *SAP Solution Manager 7.2 mainstream maintenance*.



Note:

This means that SAP Cloud ALM is the go-to platform for all new customers and – over time – all SAP Solution Manager customers.



## LESSON SUMMARY

You should now be able to:

- Explain the meaning of Customizing
- Illustrate the use of Implementation Guides for Customizing
- Outline the Role of SAP Solution Manager in implementation projects



# Unit 3

## Lesson 2

# Transport Requests for Customizing

## LESSON OVERVIEW

This lesson introduces the tools that SAP delivers to assist in the management of implementing and maintaining customizing changes to an SAP system landscape. This lesson will explain the tools for creating and working with transport requests and tasks.

## Business Example

Customizing activities are performed in the development system. The customizing changes must be recorded for transferring these changes to both the quality assurance and the production system. For the recording of customizing changes, transport requests must be created. In addition, it is necessary to assign tasks for the different members of the customizing team in the transport request. As a project leader, you therefore need to become familiar with the concept of transport requests and tasks.



## LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Create transport requests for Customizing
- Explain the use of CTS projects
- Describe the content of a transport request for Customizing changes

## Tools for Managing Transport Requests



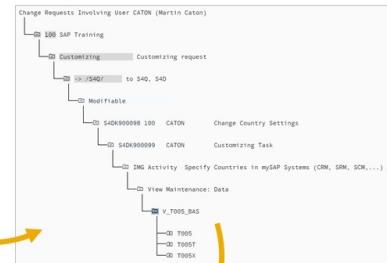
SPRO

### Implementation Guide (IMG)

Structure	Additional information
SAP Customizing Implementation Guide	Cross-Client
Commercial Project Management	
Activate Business Functions	
Conversion of Accounting to SAP S/4HANA	
SAP NetWeaver	
SAP Gateway	
SAP Gateway Service Enablement	
SAP Gateway	
Notification Channel	
Enterprise Event Enablement	
UI Technologies	
General settings	
Set Countries	
Define Countries in mySAP Systems	Client-Specific
Specify Countries in mySAP Systems (CRM, SRM, SCM,...)	Client-Specific
Set Country-Specific Checks	Client-Specific
Insert Regions	Client-Specific
Define Country Codes	Client-Specific
Define Postal Code	Client-Specific

SE09

### Transport Organizer



### Transport Management System (TMS)



Figure 42: Customizing Tools

SAP provides implementation tools for customizing.

- Implementation Guide (IMG)
- Transport Organizer

The **Implementation Guide (IMG)** is the main customizing tool. It represents the central entry point for all project members for carrying out the customizing activities. Customizing takes place out of the created IMG project, which contains all the customizing activities for a specific project scope in a hierarchical list. If the customizing is changed, these changes must be recorded. The **Transport Organizer** (transaction SE09/SE10) records the customizing changes in transport requests.

The *transport request* is an information source in the *Transport Organizer* that records all changes and modifications made to repository objects and customizing settings during a development project.

If the customizing is finished, the transport request can be released to the *Transport Management System* (TMS) for transferring the changes to other SAP systems in the SAP system landscape.

The function for recording the changes made during customizing is integrated in the customizing and table maintenance transactions. The *Transport Organizer* is called from within these customizing transactions for recording the changes into transport requests. The *Transport Organizer* is fully integrated in the TMS.

**The Transport Organizer provides information about:**

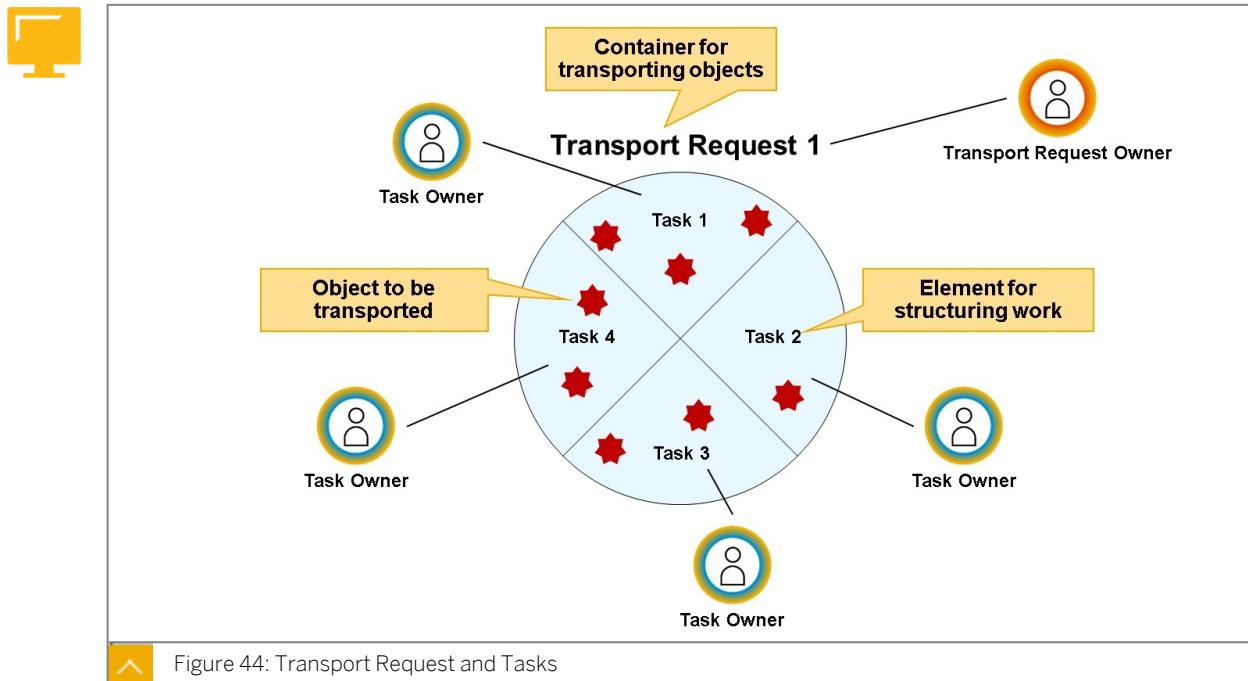
- Transport requests for customizing
- Transport requests for repository objects
- Global transport information

The *Transport Organizer* is used to create, manage, release, and analyze transport requests that record customizing or repository objects. To access the *Transport Organizer*, you can use the transaction code **SE09** or **SE10**, which both lead to the same screen.

All customizing requests that belong to a specific user are displayed according to a set of standard selection criteria in the *Transport Organizer*. The selection options include user, transport request type, transport request status, and date. An example of these selection criteria is shown in the figure above. In the *Transport Organizer*, the screen area *Global Information* is displayed. In this area, you can get a quick cross-system overview of the status of transported transport requests.

## Creating Transport Requests and Tasks

Customizing mainly consists of client-specific table entries which in turn are saved to transport requests of type *Customizing*. A transport request belongs only to one user who owns the transport request. It is possible that other users are also storing their objects in the same transport request. The owner of a transport request can administer and release the transport request. For example, the owner can assign other users to the transport requests, can release the transport request, and so on.



The *Transport Organizer* allows you to work with a team on a development or customizing project, and allows the changes made to the SAP system to be recorded.

A **transport request** is used for transferring objects that belong together into another SAP system simultaneously. As long as it hasn't been released, you can compare a transport request with a folder in the *File Explorer*, which contains and subfolders (the tasks), which in turn contain (references to) different documents (the customizing objects). A transport request initially contains one task per member assigned to the transport request.

**Tasks** are smaller units that are used by team members or customizers for recording the objects they want to transport. The changes a project member has made are stored in the tasks that the project member owns.

In general, a customizing object can be listed in several tasks belonging to the same transport request. Likewise, a team member can own several tasks belonging to a transport request.

The recorded objects are only transported in the context of the entire transport request. This function is used to manage and control the distribution of changes within an SAP system landscape.

SAP recommends the following use of the transport function:

- The project manager or responsible person creates a transport request.
- By creating a task inside the transport request for each of the project members, the project members are assigned to the transport request.

- The project members record changes to their objects to their tasks.

This leads to the following advantages for the management of transport requests:

- The project managers have control of all changes that are recorded in their transport requests.
- Objects in tasks can't be transported individually. They are only transported with the whole transport request.
- With the release of the transport request, the project leader is able to control when the objects are to be transferred to the other SAP systems in the SAP system landscape.



**Hint:**

Create as few transport requests as possible by grouping the developers and customizers. Try to avoid creating a new transport request for each little change. It is much easier to handle transport requests if there are only a few hundred per year, instead of several thousand per year.

## Managing Transport Requests in Projects

In this lesson, you have seen how customizing changes are recorded to tasks and transport requests and how this is done technically. You also now have an idea of which organizational procedures during the creation and recording of transport request can help you to get an overview of when the changes are transported.

First, let's introduce the team. There is the project leader, the developer, the tester, and the transport administrator. All of these are roles. They may be combined in fewer people, based on security and necessary knowledge.



### The roles involved in developing and transporting



**Development Project Team Leader:** manages transport requests



**Developer:** develops repository objects and performs customizing



**Tester:** tests newly developed or changed functions

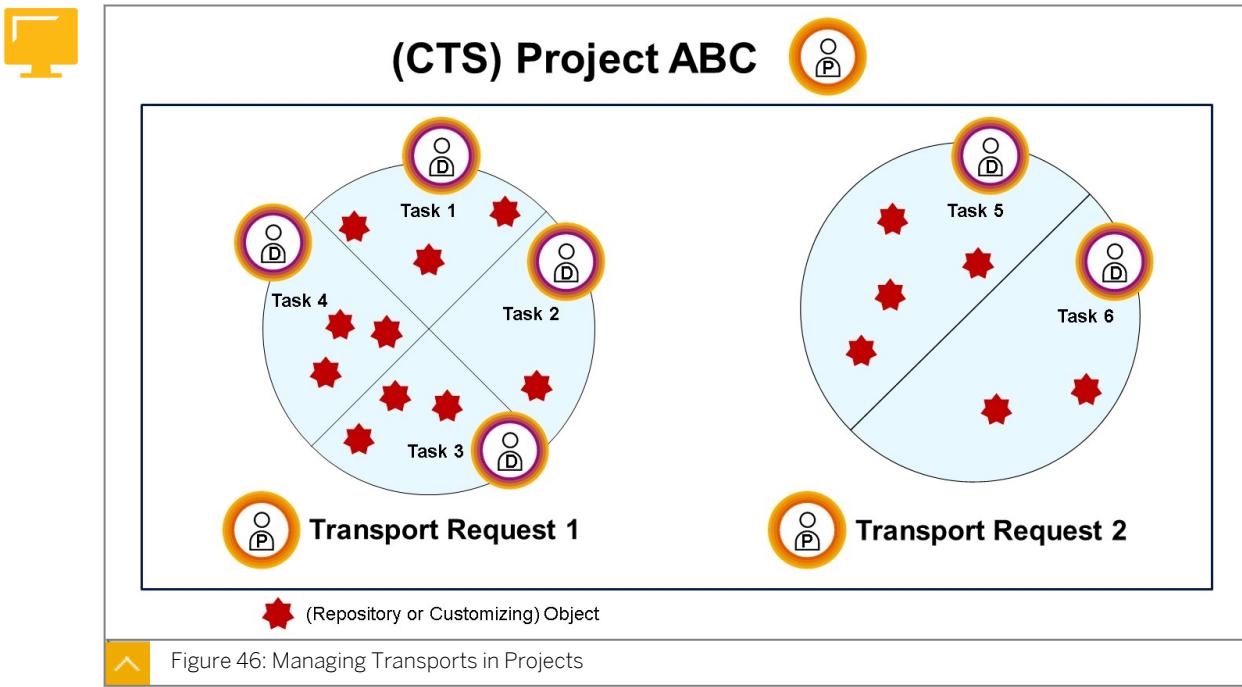


**Transport Administrator:** responsible for software logistics



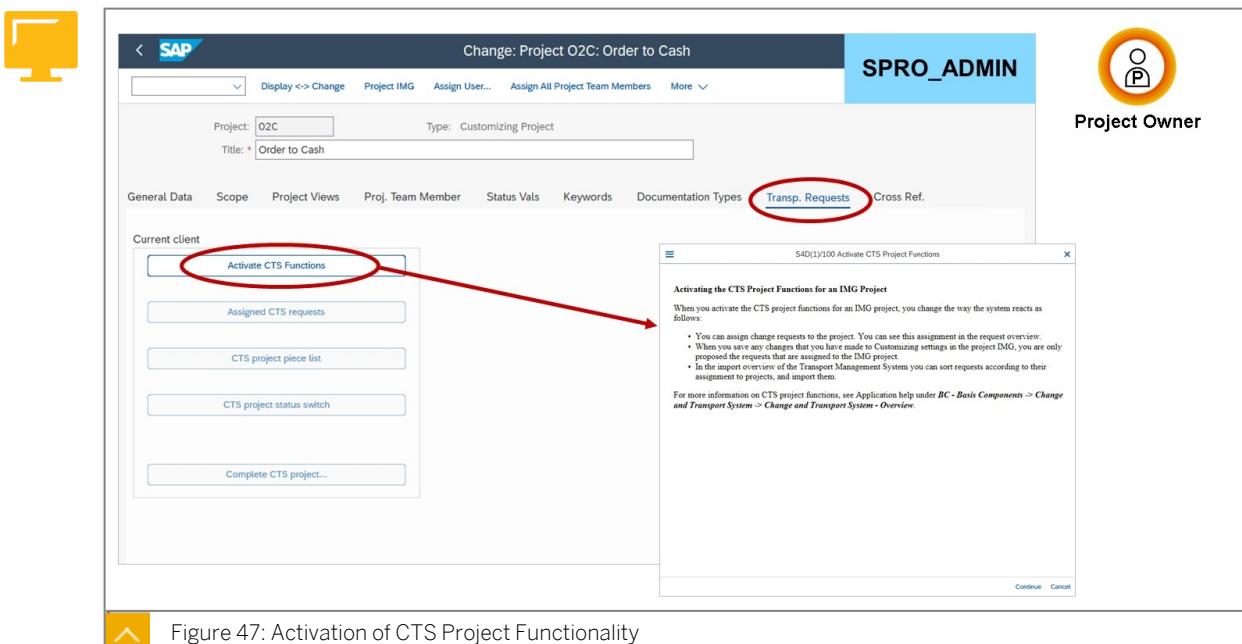
Figure 45: The Roles Involved

Project IMGs may be linked to the *Change and Transport System* (CTS). The transport requests created and assigned by the project leader contain smaller units called tasks that have been allocated to the project team members. In these tasks, the changes made by the team members are recorded. A team member can also own several tasks belonging to a transport request.



With the CTS functionality in the IMG projects, it is possible to link the transport projects (CTS projects) to the Project IMG in which the changes are done.

If you want to use the CTS functionality in IMG projects, you have to activate the functionality in the IMG project first. IMG projects are cross-client. As a result, the CTS function may be separately enabled in different clients for the same IMG project. The following figure shows how to activate the CTS project functions for an IMG project. This activation can be done by changing the IMG project in transaction SPRO\_ADMIN.



When you activate the CTS project functions for an IMG project, you change the way the SAP system reacts as follows:

- The SAP system creates a CTS project and relates it to the IMG project.

- You then can assign transport requests to the CTS project.



Note:

You can see this assignment in the *Transport Organizer* as well.

- When you perform customizing in the Project IMG, you can record the changes only in transport requests that are assigned to the assigned CTS project.

The assignment of transport requests to a CTS project can be done in two different ways:

- The first way is to assign the transport requests created from the project administration (in transaction SPRO\_ADMIN). This is shown in the figure “Assignment of Transports to CTS Projects”.
- The second way is to assign a transport request to a CTS project and to use the *Transport Organizer* (transaction SE09 or SE10) and to put the CTS project into the properties of the transport request. This can be done in one of two ways:
  - By displaying the transport requests in the *Transport Organizer*, double-clicking the modifiable transport request and then switching to tab *Properties*.
  - By filling out the *Project* field when creating transport requests in the *Transport Organizer*.



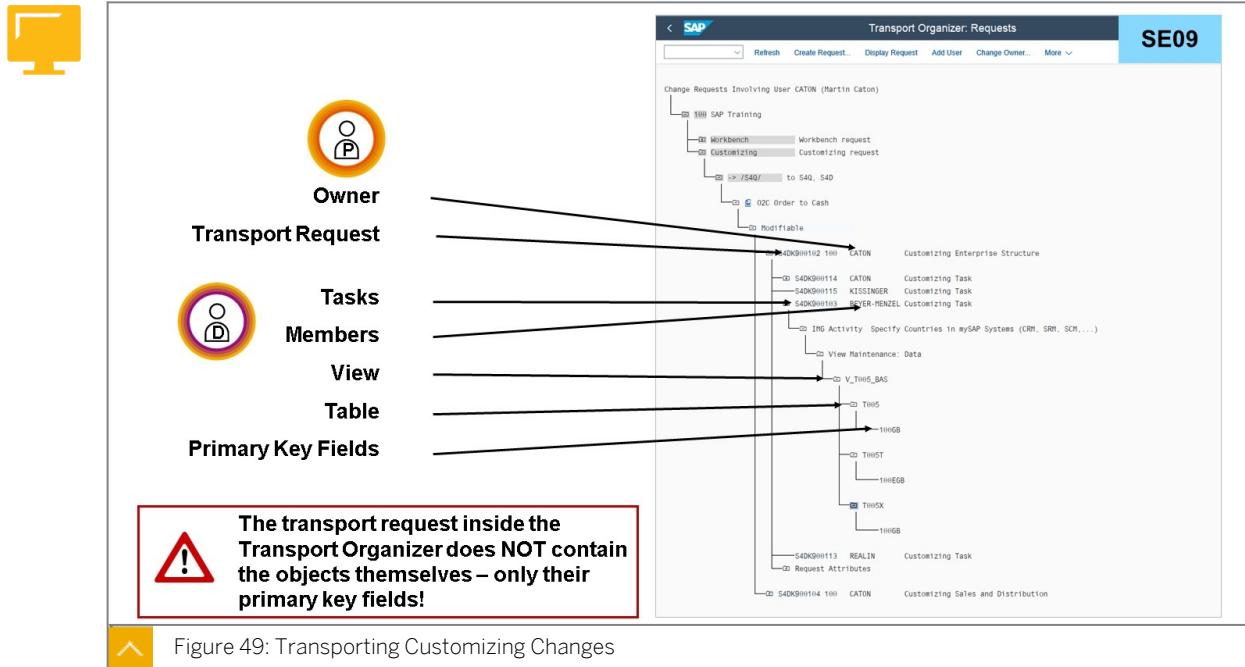
Figure 48: Assignment of Transports to CTS Projects

In the *Import Queue* screen of the TMS for the system in question (transaction STMS, area *Import Overview*), you can filter requests according to their assignment to CTS projects, and import them. Using this function it is possible to transport CTS projects individually as a unit containing several transport requests.

**Caution:**

In general, customizing projects should not overlap. During the release of transport requests, dependencies of requests belonging to different projects should be recorded, and projects should be checked for overlapping.

## Transporting Customizing Changes



**⚠ The transport request inside the Transport Organizer does NOT contain the objects themselves – only their primary key fields!**

Figure 49: Transporting Customizing Changes

Customizing transport requests usually contain changes to customizing. At the time the change is made in the SAP system, the **keys** that are identifying the table entry are stored in the transport request's task. At the time when the transport request is exported, this transport request will extract the relevant **table entries** from the database of the SAP system and copies them to the transport directory.

**Note:**

The contents of the transport request always represents the customizing entries at the time when the transport request was released.

Because of the dynamic access to table entries during customizing, both client-specific and cross-client customizing are not protected from being overwritten. Table entries are locked while the customizing transaction is being used (by the enqueue service), but they are unlocked as soon as the changes are completed and saved to a transport request.

Customizing changes can be logged. This is often used for documentation, such as who has changed a certain customizing setting. Transport requests only record the table keys at the time when a customizing entry is changed. From a transport request, you can only obtain information on who has made the changes but no information about the state before and after the change. This information can be stored by activating the logging of customizing tables.

To analyze the logging of changes to customizing tables, use IMG logging through transaction SCU3. The prerequisites for *IMG Logging* are:

- In the *Technical settings* in ABAP Dictionary maintenance (transaction SE11), the flag *Log Changes* has been selected for the respective customizing table or view.
- The profile parameter *rec/client* is set for the appropriate clients.



Note:

For additional information, see SAP Note [1916 – Logging table changes in R/3](#) and SAP Note [2490201 – R3trans default setting for table logging](#).

Generally, all objects are transported to the target system in the state in which they exist in the source system. Objects transported from the source system overwrite objects in the target system that have the same names. Objects are deleted in the target system if they do not exist in the source system.

For productive systems, there are also legal requirements in some countries that make it necessary to activate customizing table logging. Some legal requirements for activating the table logging can be found in SAP Note [112388 – SAP system audit | Tables requiring logging](#).

If not otherwise specified in the TMS configuration, the customizing settings are transported using the standard transport layer.



## LESSON SUMMARY

You should now be able to:

- Create transport requests for Customizing
- Explain the use of CTS projects
- Describe the content of a transport request for Customizing changes

## Customizing Procedure

### LESSON OVERVIEW

In this lesson, you will learn how customizing procedures should be set up. You will also identify how to test *customizing* transport requests. In addition, you will get an idea about cross-client Customizing.

### Business Example

Changes to Customizing require testing and approval before being promoted to production. These implementations require communication between project leaders, who create transport requests, and the TMS administrator who is responsible for importing the transport requests. As a project leader, you want to know how the customizing procedure needs to be defined to keep the propagation of transport requests through the landscape consistent with the transport strategy.



### LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Describe the Customizing procedure from the creation to the release of a transport request
- Explain the difference between a workbench request and a customizing request
- List Best Practices for Customizing

### Steps in the Customizing Procedure

In an SAP implementation, where customizing and development changes are integral to the SAP system being available, projects must be executed in a structured environment using defined procedures, to minimize the threat of downtime caused by bugs.

The goal of your project organization must be to divide the large number of activities among the project team so that the team members do not interfere with each other's work. You must make sure that work that logically belongs together but is being performed by different team members is still connected. This is carried out by dividing the tasks in a customizing project among three roles, each of which is responsible for performing certain tasks.

#### Project Team Leader

The project team leader creates the transport requests and assigns the appropriate team members to them. When a team member is assigned to a transport request, the *Transport Organizer* creates a task. The settings for each team member are recorded in this task. For the transport to other SAP systems, the project team leader can release the created transport requests.

#### Customizer/Developer

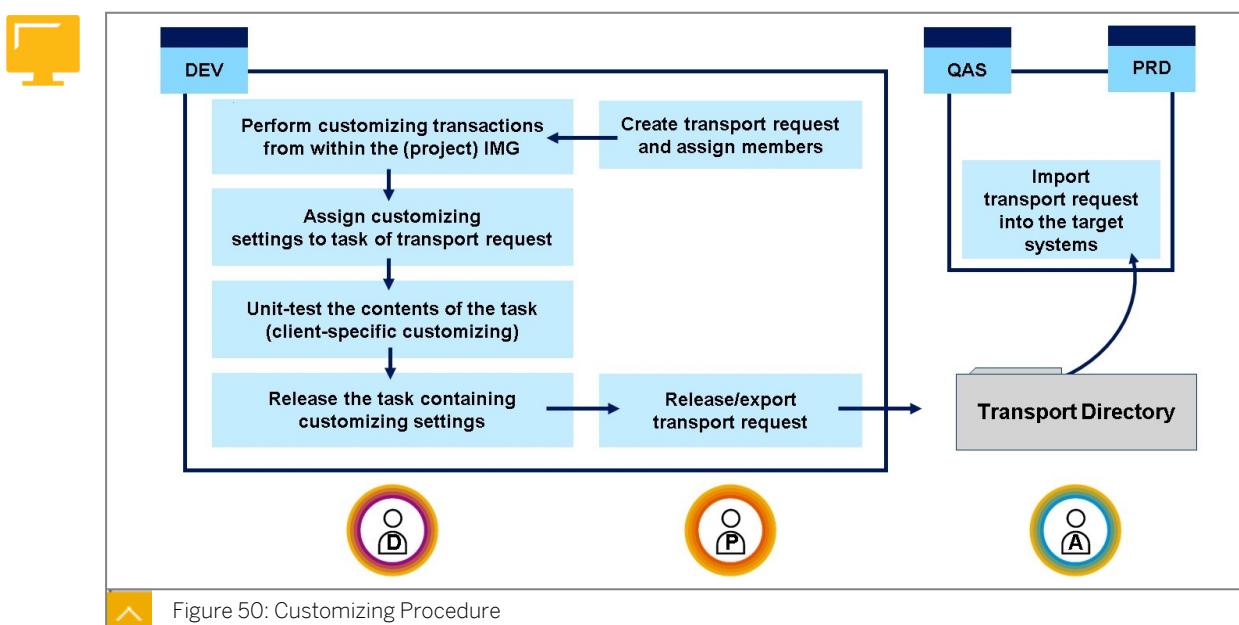
The customizers or developers perform their customizing from the IMG or their development and assign their settings to a transport request and thus, to their individual

task. The customizers can copy their settings to a *TEST* client to test them before the transport request is released (this, of course, only makes sense for client-specific data). They are authorized to release their own tasks in a transport request but are not allowed to release the transport request.

### TMS Administrator

The TMS administrator uses the TMS to transport released transport requests to subsequent SAP systems in the SAP system landscape using the predefined transport routes.

Usually, application consultants and employees in the department handle the roles of the project team leader and the customizers and developers. The project team leader decides which customizing settings to perform and how to divide the necessary changes among the project team members, who in turn execute the customizing transactions. The TMS administrator is responsible for the transport between the SAP systems along the transport routes, after the transport requests are released to the TMS.



The *Transport Organizer* and the *Transport Management System* are designed for supporting this task split. The sequence of the different customizing steps is shown in the figure above. During usual customizing work, the sequence of the steps is as follows:

1. The project leader first assigns a transport request to a (CTS) project and assigns the subsidiary tasks to the members involved. These members perform customizing changes that are recorded in the transport request.
2. After customizing is completed, the members must release their tasks so that the transport request can then be released from the source system for export to the file system.
3. The transport request can now be imported into the subsequent SAP systems.



## Hint:

Experience has shown that, by using these three, clearly-defined roles and instituting strict guidelines for the procedures and documentation of customizing, the overall customizing process is easier to manage and the risk of errors in production is significantly reduced.



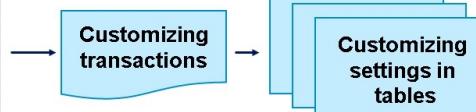
## Note:

The authorization (security) administrator enforces these roles by assigning the appropriate authorizations to each user master record. For example, the customizer must be able to execute the assigned customizing transactions and be able to release their own tasks, but not be able to create or release transport requests. SAP delivers standard roles for the customizing team leaders, customizing team members, and CTS administrators.

For more information on authorization and role management, see SAP course *ADM940: Authorization Concept for SAP S/4HANA and SAP Business Suite*.

So far, you have seen how the creation and assignment of transport requests can be done. In the following sections, you will see how the other steps work.

### Recording of Customizing Changes



Automatic recording of changes



Transport Request

Changes without automatic recording



Transport Request

manually



Figure 51: Recording of Customizing Changes

A customizing transaction is a transaction for setting customizing table entries. To use a customizing transaction, you do not need to know about the technical aspects of where and how a business object is maintained, or which transactions are used to access and change certain fields in specific tables.

After the customizing settings have been changed using a customizing transaction, the settings should be saved as follows:

- When a client is configured to automatically record changes, the settings are automatically saved to a transport request managed by the *Transport Organizer*.
- If the client is not configured for automatic recording of changes, the settings are saved but not recorded in a transport request. However, they can be included in a transport request manually. This can be done from the specific customizing transaction (for example, using the path *Table View → Transport*) or from within the *Transport Organizer*.

All customizing transactions in the IMG also allow entries to be manually saved to a transport request.



#### Note:

Some customizing transactions are classified as *Manual transport*. Changes done in these transactions must be manually added to a transport request for transport to their target system. In addition, some customizing transactions have transport steps which differ for those indicated in the figure above.

To view transport dependencies for items within the IMG: from the initial screen of the SAP system, enter *SAP Reference IMG* in transaction `SPRO` and choose *Additional Information → Technical Data → Transport Type*.

SAP recommends that all customizing changes originate in one client only and all changes be saved to transport requests. This control is put in place by the SAP system administrator via use of the client settings in the *Client Administration* client change options (transaction `SCC4`).

## Testing of Customizing

Testing of changes in the development system is very important. Only tested and error-free transport requests should be released in the development system. The advantage of these procedure is that the amount of transport requests (fewer correction transports will be needed) and error corrections between the development system and the quality assurance system is dramatically reduced, because most errors are detected early before releasing a transport request and not later during testing in the quality assurance system.

### Testing Customizing Transport Requests

- Before releasing a customizing transport request, perform a *unit test* to:
  - Test the functionality of the customizing within the transport request.
  - Verify that the transport request is complete.
- Maintaining a separate client for testing allows:
  - Real unit testing.
  - Maintenance of test data without the risk of creating customizing-dependent data.

All customizing must be tested prior to production in two ways: unit testing and quality assurance testing. Perform unit testing first.

**Unit testing** is the testing of individual customizing settings. **Quality assurance testing** is the testing of all customizing settings together. Unit testing should be performed by the customizer and be completed before the release of the transport requests.

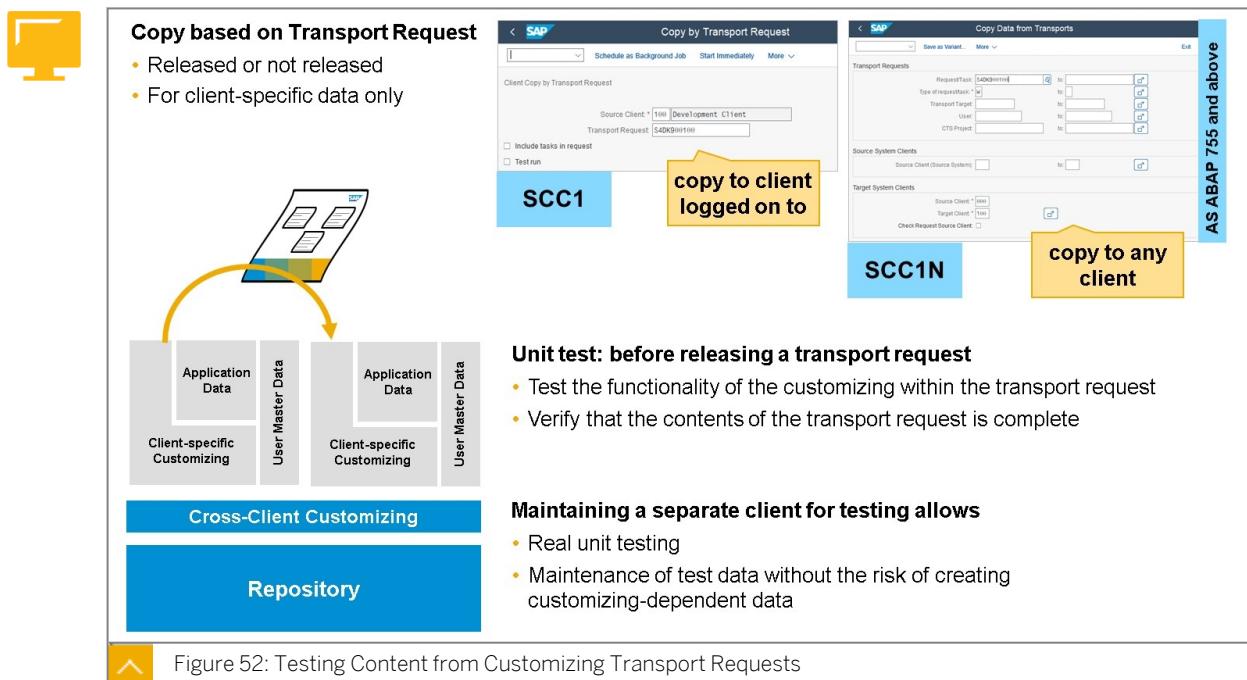
Unit testing typically requires application data. Because many customers find it advantageous to keep their customizing client free of application data, another client is created with the necessary application data for unit testing.

Before releasing a transport request, copy the recorded changes to a separate client for unit testing.

Transaction SCC1 (or its successor SCC1N, which is available as of SAP S/4HANA 2020), copies changes from one client to another based on:

- A task
- A transport request
- A transport request including its tasks

After unit testing the tasks, a transport request can be released. Unit testing alone, however, is not sufficient for transporting customizing changes to production. After unit testing, the change needs to be tested with all other customizing settings in quality assurance testing in a dedicated SAP system. This is done to ensure that all SAP system settings work properly together.



#### Note:

When copying the contents of a task to the unit test client, the task does not need to be released. Not releasing the task allows any errors identified during the unit test process to be corrected and assigned to the same task. Once a task is released, no further changes can be recorded to it, and a new task has to be created in the transport request.

To copy the contents of a transport request from one client to another client, use transaction SCC1. Log on to the target client, that is, the unit test client. Enter the source client and the transport request to be copied.



#### Note:

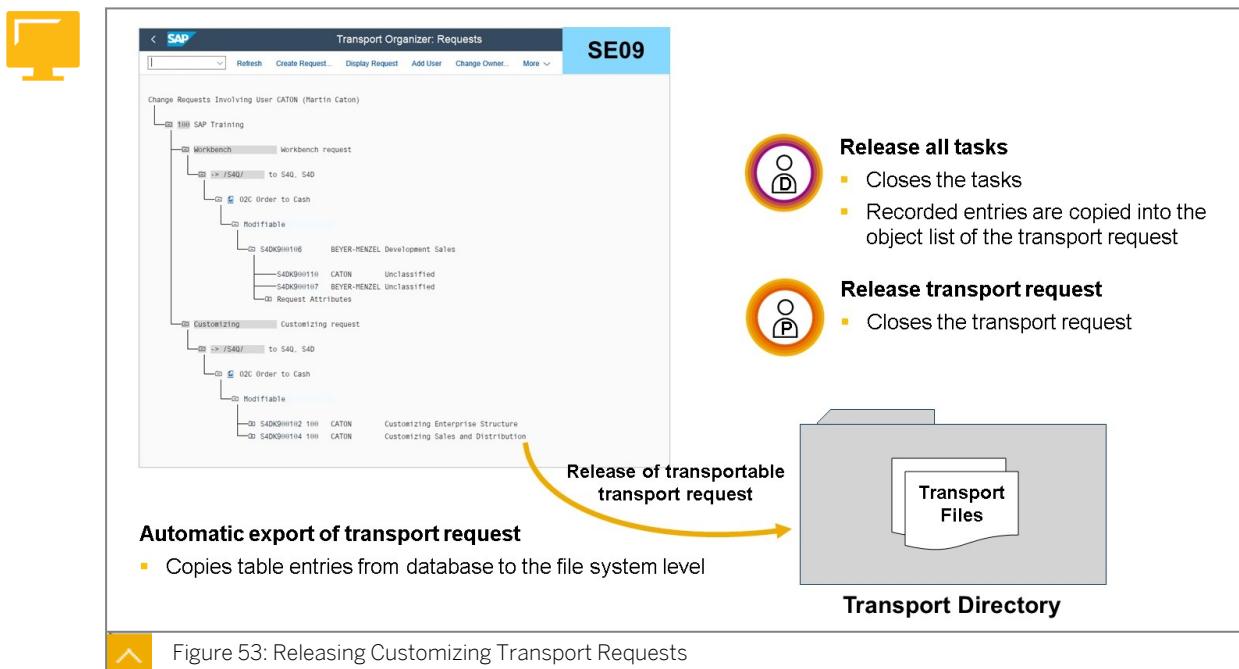
If a transport request contains cross-client objects, these objects are not copied.

With FP01 for SAP S/4HANA 2020 (that is, SAP\_BASIS 755 SP01), SAP ships transaction SCC1N as successor for transaction scc1. With the help of this new transaction it is possible to copy customizing objects recorded in transport requests to several target clients. This can be a local transport request or an imported transport request from another system. In contrast to transaction scc1, transaction scc1N can be executed in any client. In addition, a large number of new parameters are available.

If you want to copy transport requests that have not been released with transaction scc1N, proceed as follows:

- In the *Export / Import Time of the Transport Request* area, select the *Local System Import Date* radio button.
- If it is a mandatory field (this depends on the release / SP level), enter a date that is far in the past in the *Export / Import Date* field.

## Releasing Customizing Transport Requests



Promoting changes recorded in a transport request begins with releasing the relevant tasks. Releasing a task indicates that the owner of the task has completed customizing or development work, that the unit testing was successful, and that the appropriate documentation is complete. This means:

- The task contains a recorded object.
- The task has been documented.
- The task is owned by the person releasing it or the person releasing it has the appropriate SAP system authorizations.

To release a task:

1. Enter the *Transport Organizer* by executing transaction **SE09**.
2. Check that your user ID is selected in field *User*, that *Modifiable* is selected as *Request Status* and choose *Display*. The request overview displays.

3. To view all tasks in a specified transport request, expand the tree structure.
4. Position the cursor on the task you wish to release and choose *Release Directly* in the application toolbar.

By releasing a transport request, you indicate that it has sufficient documentation, the changes recorded in it have been tested, and the changes are ready to be transported using the TMS transport routes. During the export process triggered by the release, the objects recorded in the transport request are copied from the SAP database to operating system level files in the transport directory. In addition, a record of the transport request is automatically added to the appropriate import queues of the SAP systems defined in the TMS.



#### Hint:

Only transportable transport requests are exported when released.

Releasing and exporting a single transport request generates export logs and import logs when importing it into subsequent SAP systems. Testing in the quality assurance system and QA approval sign off are necessary before the import into the production system takes place. To support the validation process and limit the technical and administrative overhead, SAP recommends that you transport changes that belong to the same project together in a limited number of transport requests. This is ideally done by assignment of tasks within the transport requests by the project team leader or the person responsible for the transport.



#### Hint:

To keep control, you should always assign transport requests to a project. This makes it easier to import and approve the project.

Sometimes you want to merge transport requests. You can combine multiple transport requests into one single transport request. Merging transport requests can be done explicitly in the *Transport Organizer* by choosing *Utilities* → → *Reorganize* → *Merge Requests...* from the menu.

## Cross-client Customizing

We have learned that the database of the AS ABAP-based SAP system contains not only application data, user data and client-specific customizing, but also repository objects and cross-client customizing.

### Client-specific customizing

These are typically entries in customizing tables where the client field is on the first position in the table key. An example of this is the customizing table *T005* with the country customizing. The entries in that table only affect the client that is specified in the key field of the entry.

### Cross-client customizing

These are typically entries in customizing tables without a client field in the key. An example of a client-specific customizing table is the table *T000*, in which all clients that exist in an SAP system are stored.

### Repository objects

Repository objects are, for example, table definitions in the ABAP dictionary or programs and function modules.

You must distinguish between client-specific and cross-client customizing.

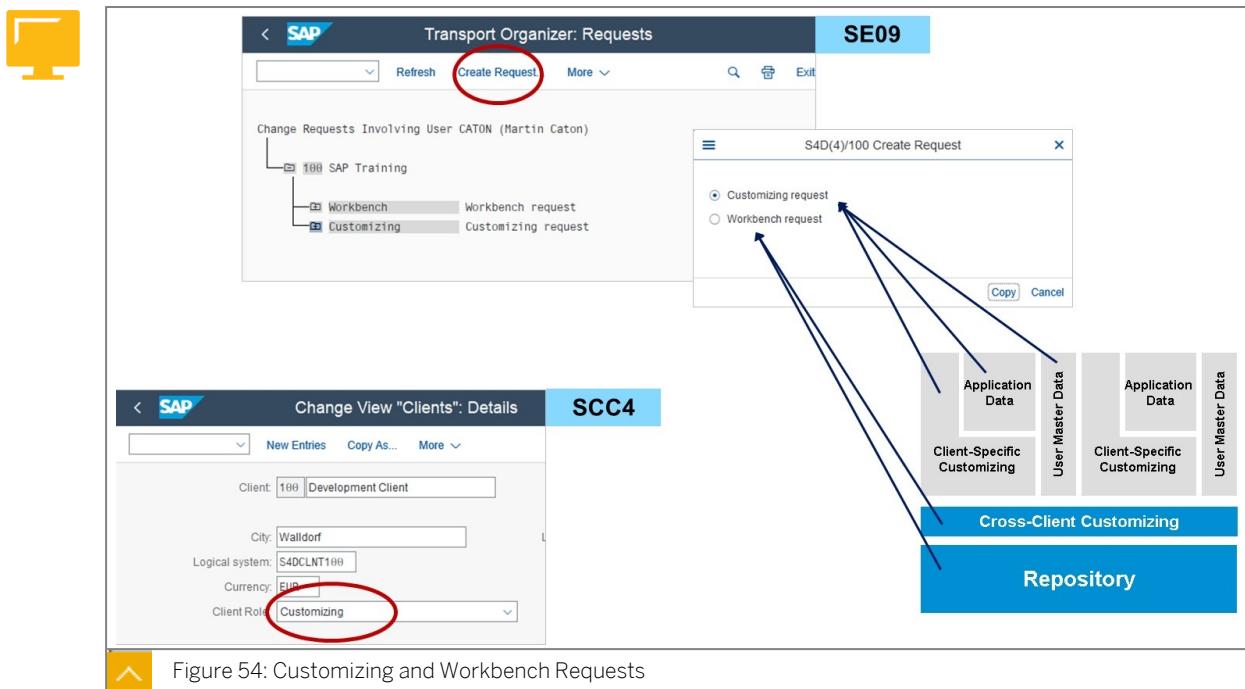
To determine the client dependency of IMG transactions in a project IMG, from the IMG menu, choose (*View → Additional information → Technical Data → Client-Dependence*).

Cross-client customizing affects one of the following:

- **Cross-client customizing objects**, which are repository objects generated by customizing demands. To ensure proper transport, assign these repository objects to a customer package. Examples of these objects are search helps, condition tables, and hierarchies.
- **Global customizing settings** are the standard SAP system settings and configurations in various tables whose key value does not contain the client. Examples of these settings are calendars, online help settings, printer settings, communication settings, and schedules.

While client-specific customizing changes are recorded in *Customizing* transport requests, cross-client changes must be saved to *Workbench* transport requests. Therefore, changes to cross-client customizing objects, global settings, and repository objects require a *Workbench* transport request.

## Customizing Request versus Workbench Requests



There are different types of transport requests with special properties, including:

- The transport request of type *Customizing*. This is a transport request for transporting settings from client-specific tables. Cross-client customizing or workbench objects can't be assigned to this kind of transport request.
- The transport request is of type *Workbench*. This is a transport request for transporting repository objects and settings from cross-client tables.

With this, there is a split between client-specific data that can be recorded in transport requests of type *Customizing* and the transport request of type *Workbench*. A project leader or a person responsible for the transport can control who is allowed to transfer cross-client changes by assigning tasks inside the workbench requests.



Hint:

Some cross-client customizing settings are only connected to the TMS if the client role is set to *Customizing* in SCC4.

## Planning Customizing Change Management

Different customizing tools and the technical settings that are required for the usage of these tools. We will now discuss what is important for planning the customizing change management.

From a process point of view, it is essential to establish policies on how customizing and development will be carried out and by whom.



### Define customizing policies:

- Define policies for creating the project IMGs and views within a project, and the use of project status and documentation
- Establish a single source client for all customizing

**Use a single client for all customizing**

### Restrict access to transport request management tasks:

- Determine change management roles
- Assign appropriate user authorizations

**Use projects to group transport requests**

### Establish project teams:

- Establish project team leaders to define projects, and to assign tasks within transport requests to team members
- Provide training on the use of change management tools to all project teams

**CTS Project**

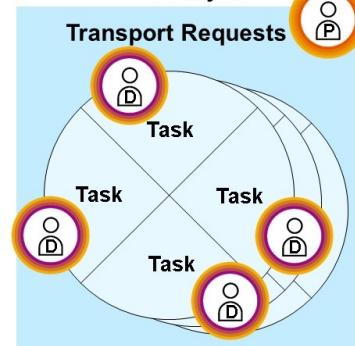


Figure 55: Planning Customizing Change Management

Customizing is the standard way of adapting the functions of the delivered SAP components to the requirements of a company. Project control tools allow easy documentation and monitoring of each phase of the SAP implementation and customizing.

Customizing is required for implementations of SAP systems. Because customizing is a highly integrated process, both between functional areas and within the areas themselves, all customizing activities must be performed in a single client, usually known as the *development client*. Development tasks should also be restricted to this client, ensuring that there is a single environment for all implementation efforts.

The project team leader is responsible for creating transport requests for all project team members at the start of a development or customizing project.

After project team members complete their work, they save their changes in the tasks that comprise the transport request, and document the changes in the task. When all project team members have released their tasks, the project team leader releases the transport request for transport to other SAP systems within the SAP system landscape.

The project managers or team leaders are responsible for creating the appropriate project IMGs for the specific business areas and assigning who is responsible for executing the specific customizing transactions. The project management function enables you to assign

tasks in the *Transport Organizer*. The project team leader should activate the CTS functions of the project, create transport requests for recording and transporting the customizing settings, and add users to the transport requests.

The project managers or team leaders are also responsible for training the team members on the tools and processes, as well as establishing and enforcing documentation and unit testing standards.



Note:

SAP recommends that transport requests should contain testable units of work, so it is best to have a minimum number of transport requests with many tasks. This reduces the number of transport requests moving through the transport system and makes it easier to troubleshoot problems when they arise.



## LESSON SUMMARY

You should now be able to:

- Describe the Customizing procedure from the creation to the release of a transport request
- Explain the difference between a workbench request and a customizing request
- List Best Practices for Customizing

# Transport Requests for Development

## LESSON OVERVIEW

In this lesson, you will learn about planning change management for development. You will identify the different implementation tools provided by SAP for customizing and development. You will also identify the role of *Transport Organizer* in recording the transport requests.

## Business Example

As the transport administrator for your company, you must ensure that changes to programs and other development objects are distributed to all SAP systems in the SAP system landscape. Distributing these changes to all SAP systems in the SAP system landscape will provide a consistent and stable implementation of business processes and customer development. To streamline the process of distributing changes, the transport administrator must be familiar with the *Transport Organizer*, which controls the distribution of changes throughout the landscape.



## LESSON OBJECTIVES

After completing this lesson, you will be able to:

- List Best Practices for Development
- List differences between Customizing and development
- Outline the idea of SAP Software Change Registration
- Outline the idea of naming concepts and the repository object directory
- Explain the concept of packages

## Planning Change Management for Development

Before starting the development in an SAP system, you should plan how the change management for development is done.

The areas which have to be described in the change management procedure for development are the following:

### Planning Change Management for Development

- Restrict repository object changes:
  - Create a single SAP system for all developments.



#### Note:

Usually, there is one development system per SAP system landscape.

- Ensure proper system and client change options.
- Assign appropriate user authorizations.
- Define development standards:
  - Use packages to group repository objects.
  - Establish standards for development and documentation.
  - Maintain versions.
- Establish project teams:
  - Provide all project teams with training on change management tools.
  - Assign team leaders to projects and assign tasks within transport requests to team members.
- Use projects to group transport requests:
  - Use the project assignment to import and approve whole projects.



Hint:

Importing and approving complete projects reduces the potential of errors in many ways.

- Do not import single transport requests – except for emergency repairs.

The tools that the SAP systems provide for change management are based on creating, documenting, and distributing transport requests. The customer must set up the infrastructure and procedures for the management, verification, and testing of these development changes.

Recommendations for development change management include the following:

- Perform development efforts in a single environment only: the development system. Set the system change options accordingly.
- Use packages to group functionally related repository objects. The transport layer assigned to the package enables the same predefined transport route to be used for all objects in the package.
- When releasing a transport request, document the purpose and the status of the changes.
- To maintain security, use authorizations to control which users can create, modify, or release transport requests (authorization object S\_TRANSPRT). SAP delivers sample authorization profiles that provide the SAP system access required for various levels of responsibility in change management.

It is useful to define development rules, customizing rules, and transport rules that describe:

- How and where changes are made.
- How and where these changes are tested.
- How the quality assurance is done.

- How and who creates, releases and imports transport requests in the SAP system landscape.

## Customizing versus Development

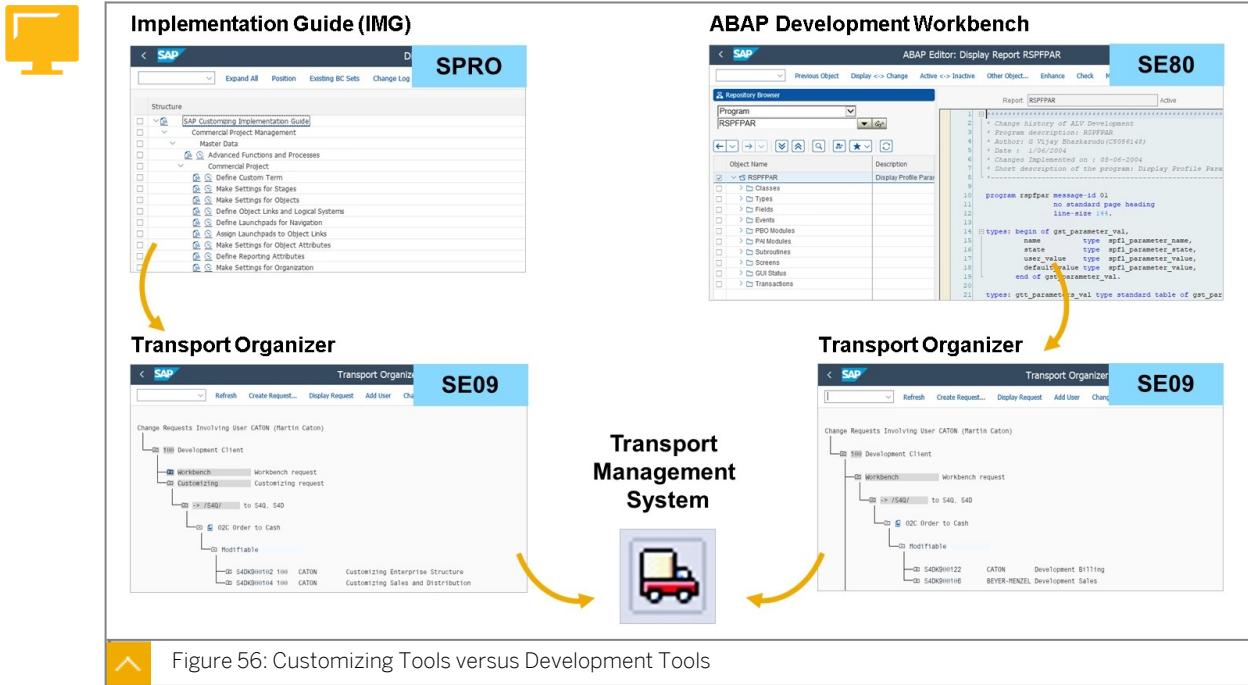


Figure 56: Customizing Tools versus Development Tools

SAP provides various implementation tools for customizing and development.

- For customizing:

The *Implementation Guide (IMG)* is the main customizing tool. Once you decide which business functions you require, the IMG automatically generates a hierarchical list of steps or customizing transactions for customizing.

The *Transport Organizer* (transaction **SE09**) records customizing changes in transport requests, which can be released to the transport system for export to other SAP systems in the SAP system landscape.

- For development:

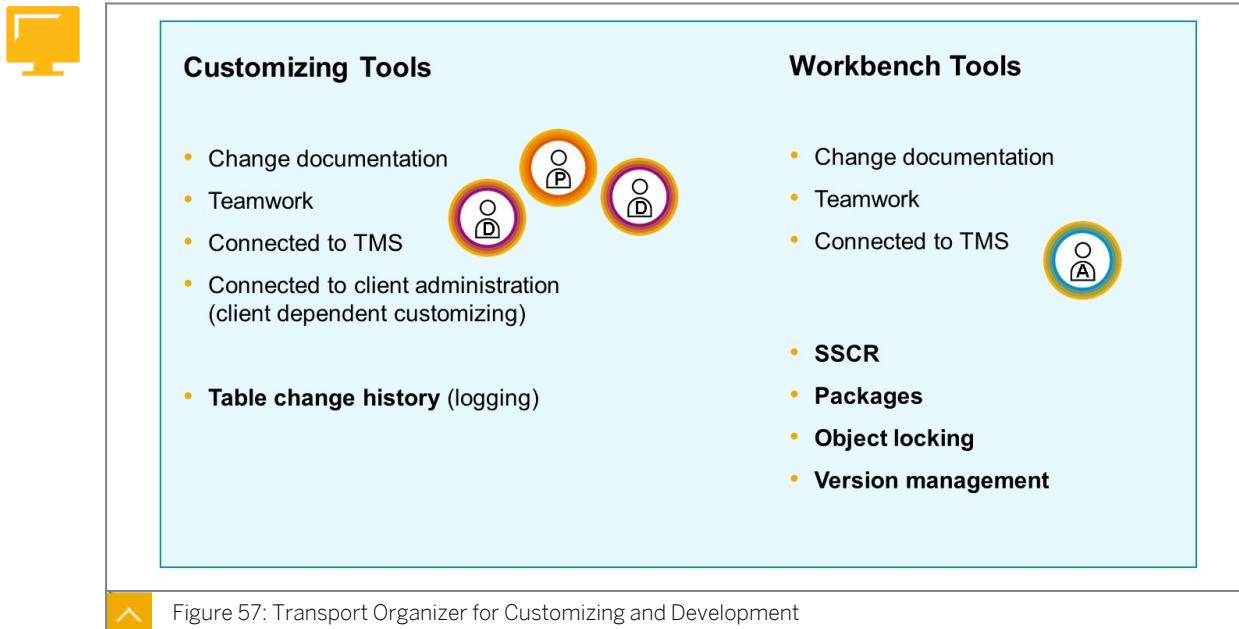
The *ABAP Development Workbench* (as well as the *ABAP Development Tools*) provides access to development tools, which cover the entire software development cycle. You can use these tools for customer-specific development and for SAP enhancements of your business processes.

Note:

*ABAP Development Tools* are an ABAP-integrated development environment built on top of the Eclipse platform. Its main objective is to support developers in today's increasingly complex development environments by offering state-of-the-art ABAP development tools. These tools include strong and proven ABAP lifecycle management on the open Eclipse platform with powerful UI capabilities.

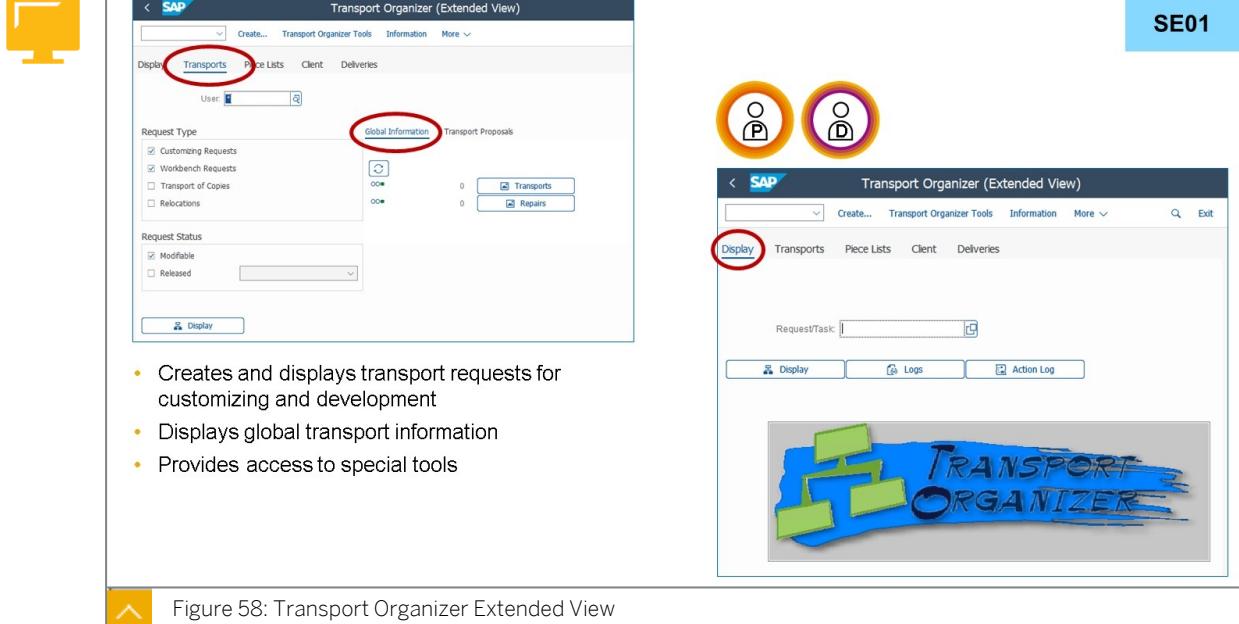
The *Transport Organizer* (transaction SE09) also records ABAP Development Workbench changes in transport requests, which can then be released to the transport system for export to other SAP systems in the SAP system landscape. The *Transport Organizer* is completely integrated with the Transport Management System (TMS).

The differences between these tools are shown in the following figure.



The *Transport Organizer* records the customizing changes and ABAP Development Workbench changes in two types of transport requests: Client-specific objects are saved to customizing transport requests. Cross-client objects are saved to workbench transport requests.

Customizing changes consist of table entries.



The *Transport Organizer* can be used for customizing and workbench transport requests. The *Transport Organizer*:

- Displays transport requests.
- Displays global transport information.
- Provides access to special tools.

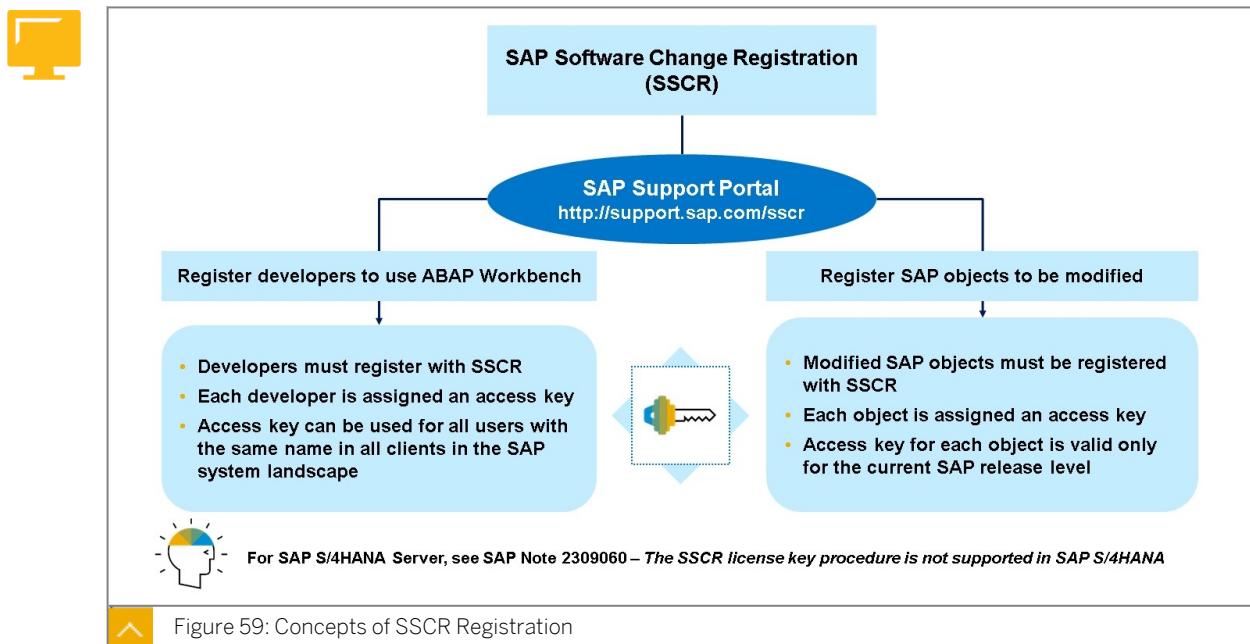
The *Transport Organizer* creates, manages, releases, and analyzes transport requests for development.

To access the *Transport Organizer*, use transaction SE09/SE10 or SE01 for extended functionality.

All workbench transport requests are displayed according to the selection criteria. Selection options include *user* (owner of either the transport request or one of its tasks), *transport request type*, *transport request status*, and date.

The *Global Information* screen area provides a quick overview of the status of transported transport requests and repairs.

## SSCR Registration Concepts



SAP Software Change Registration (SSCR) process has certain key concepts:

In ABAP-based SAP systems of SAP Business Suite, any user in an SAP system who wishes to create, modify, or delete repository objects, including customer objects, must be registered using the SAP SSCR process. Such users are often referred to as “development users” or (for short) “developers”.

As a result of the registration process, an access key is assigned to each developer. The access key is entered and saved on the developer's SAP system in the table *DEVACCESS*.

The access key is associated with the developer's logon ID and the SAP systems license number. The developer is prompted for the access key during the initial attempt to create or change a repository object.

You must register developers and all SAP repository objects (not customer objects) that are to be modified. When registering an object, you must supply the object program ID, object type, object name, the SAP system's license number, and the release of your SAP system. After registering an SAP object in an SAP system and applying the access key, the key is stored in the database table *ADIRACCESS*. This will ensure that further changes to the object do not require another key.



**Note:**  
Registered object keys become invalid after a release upgrade.

The registration can be done from the SSCR application at <https://launchpad.support.sap.com/#/sscr>, which can also be accessed from SAP Support Portal, quick link /sscr (<https://support.sap.com/sscr>).

SSCR provides development reliability, rapid error correction, and high system availability. This can be done by limiting the access to the development and object keys. Most customers are doing the registration of objects and developers centrally.

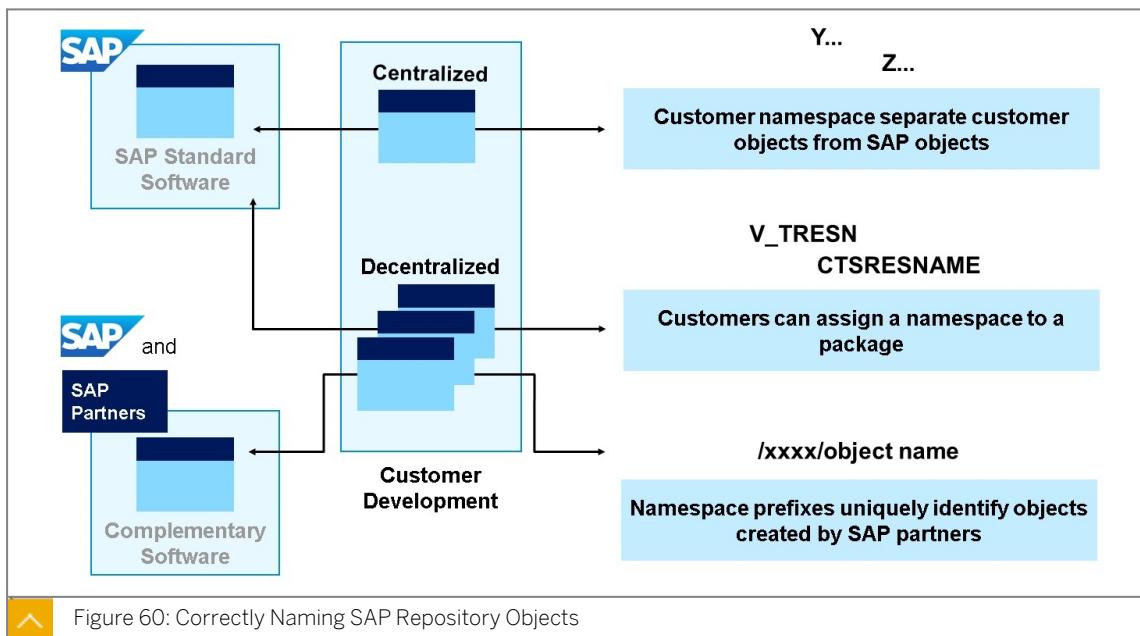


**Note:**  
In SAP S/4HANA Server systems, these keys are not checked, and, therefore not necessary. For details, see SAP Note [2309060 – The SSCR license key procedure is not supported in SAP S/4 HANA](#).

SAP Note [2501703 – Frequently asked questions about SAP Software Change Registration \(SSCR\)](#) provides additional technical information about SAP Software Change Registration (SSCR).

## Repository Objects and Attributes

The naming conventions for SAP repository objects are listed in the figure “Correctly Naming SAP Repository Objects”.



To prevent conflicts with object names when creating repository objects, developers must follow naming conventions. Extended name lengths allow you to use descriptive names. Object names should clearly describe the function of the repository object.

Namespaces differentiate between SAP repository objects and customer repository objects. Customer repository object names must begin with a Y or Z.



**Hint:**

There are also other naming conventions. For example, customer fields in an SAP table definition must begin with ZZ, **not** with Z. If the object name is a number, the customer namespace usually begins with 9.

For an overview of all of the current naming conventions for repository objects, see SAP Note [16466 – Customer name range for SAP objects](#).

Using customer namespaces correctly prevents object name conflicts between customer objects and SAP objects. If a customer has a more complex SAP system landscape with different SAP system lines and more than one development system, they require additional naming conventions to prevent object name conflicts.

In the SAP system, customer namespaces can be reserved using the view `V_TRESN`, which enables developers to assign a specific namespace to a package. For example: you are developing in the SAP system `DE1` for a project within the package `ZPROJECT1`. You have chosen the naming convention `ZPROJ1` for this project and maintained it in all your development systems. A developer in the SAP system `DE2`, working on another project within another package, now tries to create a program with the name `ZPROJ1PROGRAM`. When saving, the developer gets the message that this name is reserved for the package `ZPROJECT1`. In addition, the developer is prevented from assigning the object to another package. The developer therefore has to choose a different program name to assign the program to their package. This avoids naming conflicts right from the start when objects are created.

The view `CTSRESNAME` offers a simplified view maintenance compared to `V_TRESN`. This naming convention is according to the program ID and object type.

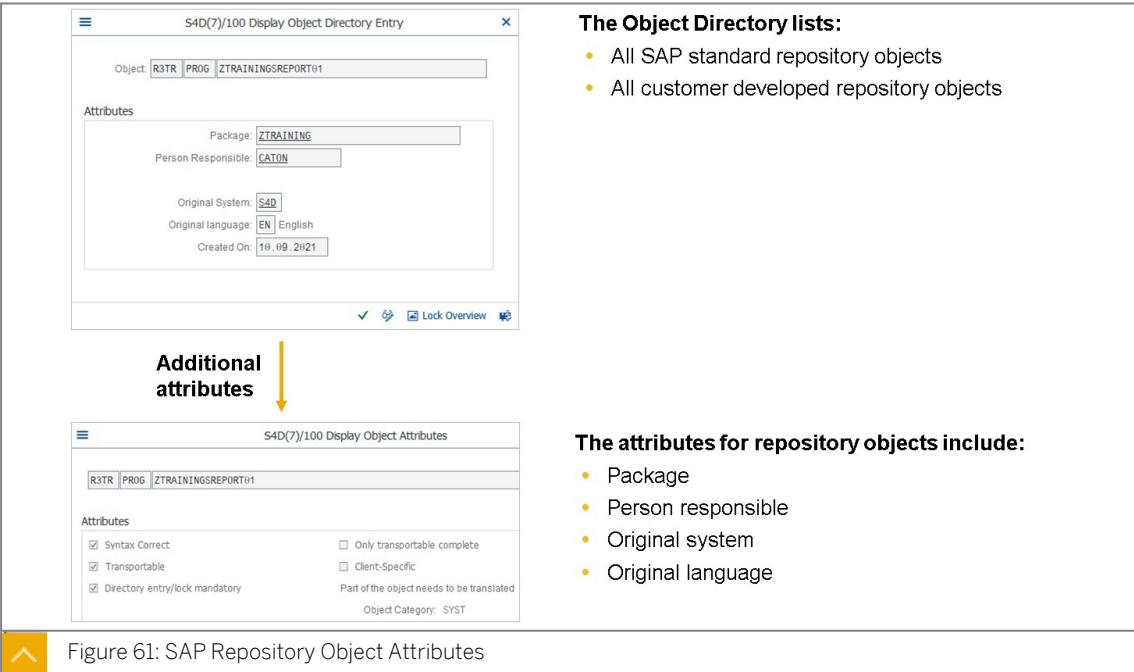


**Note:**

View `CTSRESNAME` can only be used for development namespace with reserved namespace prefixes. This means that you can only use the `V_TRESN` view maintenance to reserve naming conventions for the customer namespace `Y*/Z*` (object type-specific reservations).

Reserving namespaces is the only way to prevent naming conflicts that occur with objects created in complementary software from SAP partners. SAP partners and customer companies can apply for a name space prefix through the *Development Namespaces* application in *SAP ONE Support Launchpad* (<https://launchpad.support.sap.com/#/namespaces>) which can also be started from *SAP Support Portal*, quick link `/namespaces` (<http://support.sap.com/namespaces>). For more information concerning namespaces and reservation use SAP Note [84282 – Development namespaces for customers and partners](#).

SAP repository objects are listed in the object directory. The **object directory** is a catalog of all repository objects in the SAP system, including all standard repository objects that are delivered with the SAP system, and all repository objects created by the customer using the ABAP Workbench.



The screenshot shows two SAP dialog boxes. The top box is titled "S4D(7)/100 Display Object Directory Entry" and displays object details: Object R3TR PROG ZTRAININGSREPORT01, Package ZTRAINING, Person Responsible CATON, Original System S4D, Original language EN English, and Created On 19.09.2021. The bottom box is titled "S4D(7)/100 Display Object Attributes" and lists additional attributes: Syntax Correct (checked), Transportable (checked), Directory entry/lock mandatory (checked), Only transportable complete (unchecked), Client-Specific (unchecked), and Part of the object needs to be translated (unchecked). An arrow points from the text "Additional attributes" to the bottom dialog box.

**The Object Directory lists:**

- All SAP standard repository objects
- All customer developed repository objects

**The attributes for repository objects include:**

- Package
- Person responsible
- Original system
- Original language

Figure 61: SAP Repository Object Attributes

The object directory lists:

- All SAP standard repository objects
- All customer-developed repository objects

The attributes for repository objects include:

- Package
- Person responsible
- Original system
- Original language

Attributes for each repository object are assigned by the SAP system. The object directory is stored in table *TADIR*. This table is very central to your SAP system's consistency. To change entries in *TADIR*, use only the standard functions that SAP provides.

With the appropriate authorization, you can change the package and person responsible for the object. To change object directory entries from the *Transport Organizer*, choose transaction *SE09* and choose the menu path *Goto → Transport Organizer Tools*. Alternatively, use transaction *SE03*. Here, choose *Transport Organizer Tools → Object Directory → Change Object Directory Entries*.

Some repository objects may be generated automatically by the SAP system as a result of customizing activities. In the object directory, these SAP system-created objects are flagged as "generated".

For each entry in the object directory, the primary key is comprised of the following fields: program identification (*PGMID*), object type, and object name. The program identification is usually *R3TR*. Examples of object types are *PROG* (ABAP program), *DEVC* (package), *TABL* (table definition).

## Packages

The repository is organized by packages. Repository objects are assigned to a package, which formerly has been known as development class. The package:

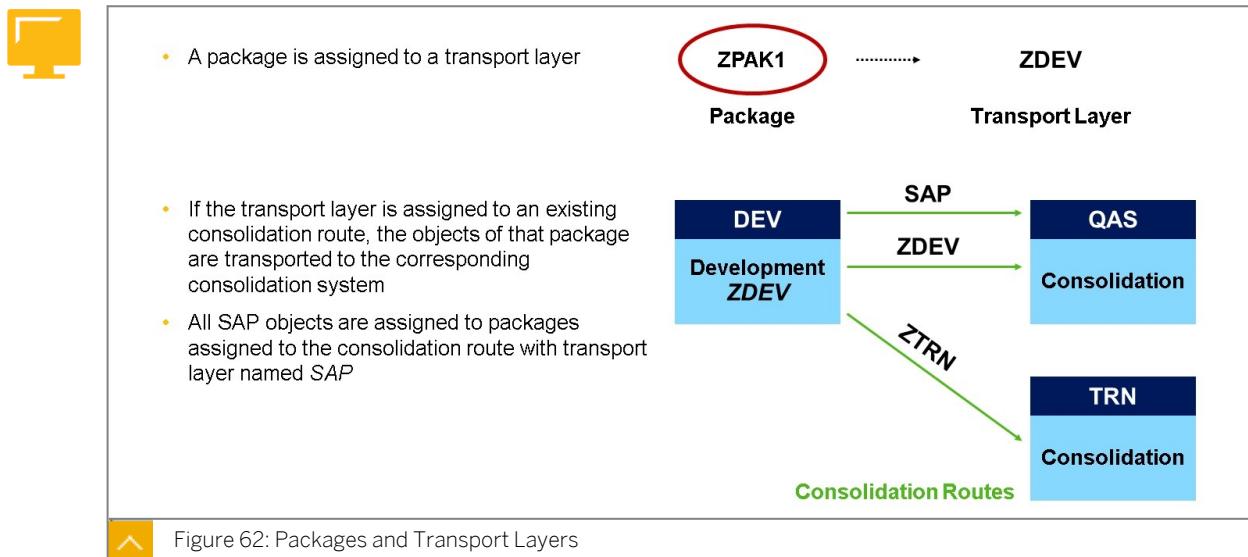
- Provides a logical grouping of objects for coordination of development efforts.
- Defines a repository object's transport layer.
- Can control the naming of objects.

Examples of package definitions are shown in the figure below.

### Assigning SAP Repository Objects Packages to Transport Layers

A package is assigned to a transport layer. If the transport layer is assigned to an existing consolidation route, the objects of that package are transported to the consolidation system using this consolidation route.

SAP objects modified in the customer integration system follow the consolidation route that is assigned to the transport layer SAP.



When a package is assigned to a transport layer, all objects belonging to that package follow the same predefined consolidation route, usually pointing from the development system to the quality assurance system.

In the figure above, both the SAP transport layer and the standard ZDEV transport layer are assigned to a consolidation route from **DEV** to **QAS**, that is, from the development system to the quality assurance system. All repository objects assigned to a package whose transport layer is ZDEV are transportable. In the figure, objects that are assigned to package **ZPAK1**, for example, use the consolidation route with transport layer **ZDEV**.

In this example, the transport layer **ZTRN** is assigned to a consolidation route from **DEV** to **TRN**, that is from the development system to the training system. All repository objects assigned to a package whose transport layer is **ZTRN** are transportable.

SAP objects belong to the preinstalled transport layer **SAP**. In the figure above, SAP objects that are modified in **DEV** will follow the consolidation route assigned to transport layer **SAP** and therefore will be promoted to **QAS**. There is only one SAP transport layer in a standard SAP three system landscape.

In this example, you can transport only those repository objects (via a *Workbench* transport request) whose package is assigned to the transport layer SAP, ZDEV, or ZTRN. If an object is assigned to a transport layer that is either blank or for which there is no transport route, you cannot transport the object via a *Workbench* transport request.

Packages are also objects of the ABAP Workbench and can be created by the *Repository Browser* (which is part of the *Object Navigator*, transaction SE80). When creating a package in the customer namespace, by default the standard transport layer is assigned to the package. Users may also choose an alternative transport layer. The transport layer will be used by the TMS tools for the determination of the consolidation route for the objects inside the package.

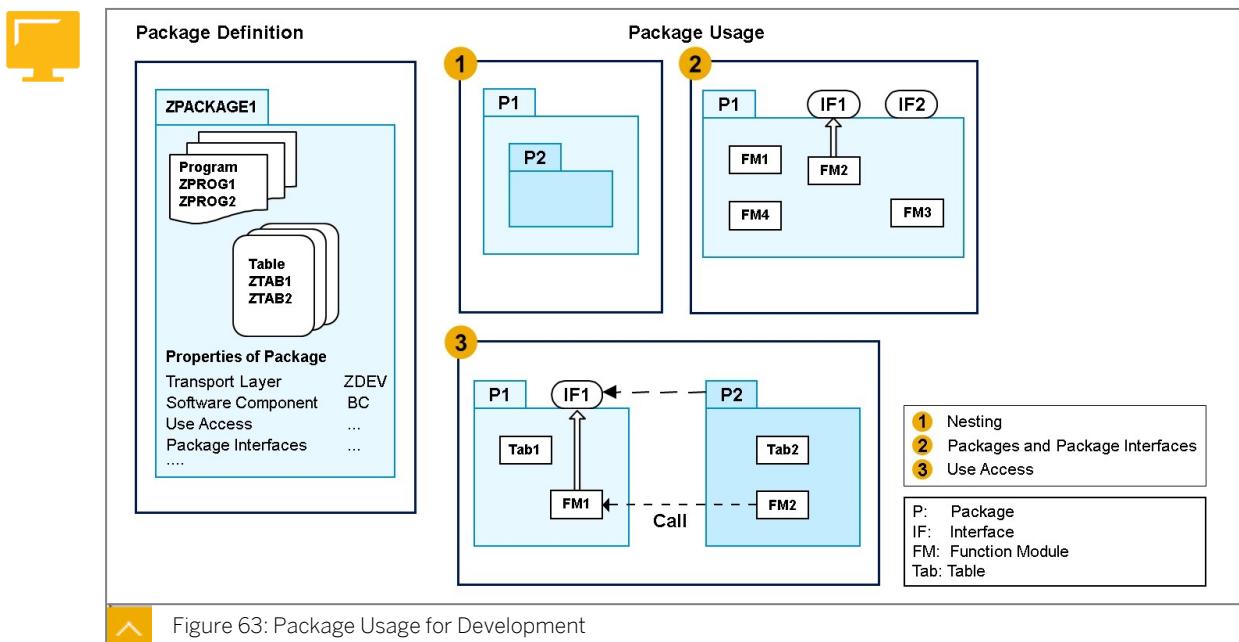
Customer created packages may begin with the following letters:

- Y or Z indicates that the package is for customer objects that are to be transported.
- \$ indicates a package for temporary objects that are not to be transported and therefore do not require a transport layer.
- TEST indicates a package for local objects which provides version management – but no transporting.

The package \$TMP is used when a repository object is saved as a *local object* and is not assigned to a transport request.

You can use view V\_TRESN or view CTSRESNAME to specify which packages can be assigned to a namespace. For example, program names starting with the string ZABC can be prevented from being assigned to a package other than the one defined for this namespace in V\_TRESN.

The V\_TDEVC view holds all packages in the SAP system, including all SAP packages.



As of SAP Web Application Server 6.10, the concept of packages was introduced. Previously existing development classes simply had been containers for development objects with a transport layer that determines how the objects will be transported. Packages extend the concept of development classes with the addition of new attributes: nesting, interfaces, visibility, and use accesses (see the figure “Package Usage for Development”).

1. Nesting of packages, defining the package hierarchy:

Nesting allows you to split the larger units of the SAP system into a hierarchical structure. The combination of interfaces and use accesses allows developers to hide package elements, and protect them from unauthorized use.

## 2. Definition of package interface:

Packages use interfaces and visibility to make their services known to other packages. All the visible elements in a package can, potentially, be used by other packages; invisible elements cannot. This allows the package to encapsulate its contents and protect its elements from being used by unspecified external packages.

## 3. Definition of use access:

Use access is the right of one package to use the visible elements in the interface of a second package (but not the other way round).

The package concept also offers the option of splitting and encapsulating the SAP system into technical units (the packages), reducing high levels of dependency and decoupling the SAP system on a large and small scale.



## LESSON SUMMARY

You should now be able to:

- List Best Practices for Development
- List differences between Customizing and development
- Outline the idea of SAP Software Change Registration
- Outline the idea of naming concepts and the repository object directory
- Explain the concept of packages



# Customer Development

## LESSON OVERVIEW

In this lesson, you will learn the concepts of customer development.

### Business Example

As the transport administrator, you must work together with the development teams to ensure that the system requirements for the development projects are fulfilled. Therefore, you need to know the concepts of customer development.



## LESSON OBJECTIVES

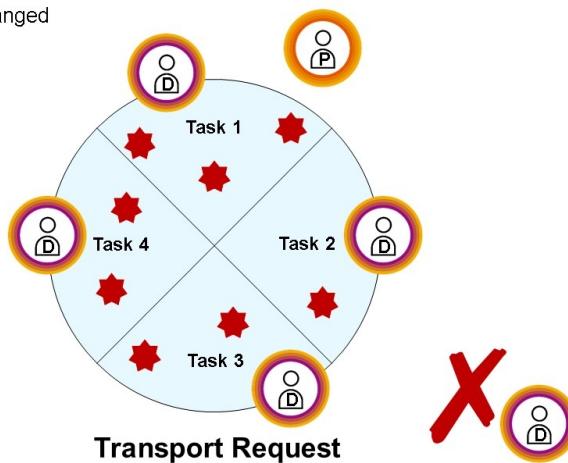
After completing this lesson, you will be able to:

- Explain object locking and version management
- Describe the process of releasing a transport request that contains repository objects
- List selected Transport Organizer tools and explain their use
- List authorizations for software logistics

## Object Locking



Repository objects may be changed by developers assigned to the transport request.



★ Repository Object

Only developers assigned to the transport request may change repository objects locked by that transport request.



Figure 64: SAP Repository Object Locking

There are two locking mechanisms when repository objects are being changed:

### Enqueue Lock

The editor program, which works with the enqueue service ensures that only one user at a time can change an object in the SAP system.

### Repository Lock

The workbench transport request ensures that the developer changing the object is assigned to a valid task within the workbench request.

When a repository object is assigned to a task within a workbench request, that object can only be changed by the developers associated with that transport request. This prevents users outside the development team from making changes to any of the objects in the transport request before the development project is completed and the associated transport requests are released.

An *object list* is associated with each task. Each user working on an object has a corresponding entry in the object list of their task. The object list records which users have actually edited the object.

Objects may be manually entered in the object list of a task or transport request. For this, in the *Transport Organizer* (transaction SE09) choose *Display*. You can manually add objects, for example, by double-clicking the transport request and adding the desired objects. Or you can use the menu path *Request/Task* → *Object List* → *Include Objects....*

These objects are not automatically locked if they are entered manually. To manually lock the object, in the *Transport Organizer* initial screen (transaction SE09), choose *Display*, select the transport request or task in question and then choose *Request/Task* → *Object List* → *Lock Objects* from the menu.

## Version Management

**Versions of SAP repository objects can be:**

- displayed
- compared
- restored

Versions of Object ZREPORTING01 of Type Report Source Code									
<a href="#">Display</a> <a href="#">Compare</a> <a href="#">Retrieve</a> <a href="#">Fetch Remote Versions</a> <a href="#">Search</a> <a href="#">Documentation</a> <a href="#">More</a>									
	Status	Version	Compare	Cat.	Flag	SAP Rel.	Arch.	Request	Request Text
	modified	755				S4DK900142	Correction: Returnable Packaging Processing	755	10.09.2021 14:32:11 BEYER-MENZEL
	active	755				S4DK900142	Correction: Returnable Packaging Processing	755	10.09.2021 14:31:46 BEYER-MENZEL
	6	755		U		S4DK900142	Correction: Returnable Packaging Processing	755	10.09.2021 14:31:59 BEYER-MENZEL
	2	755		U		S4DK900142	Correction: Returnable Packaging Processing	755	10.09.2021 14:31:56 BEYER-MENZEL
	4	755				S4DK900138	Development Returnable Packaging Processing	S4D_P00002	10.09.2021 14:29:56 CATON
	2	755				S4DK900138	Development Advanced Return Management	S4D_P00002	10.09.2021 14:14:14 CATON
	2	755				S4DK900130	Development Rebate Processing	S4D_P00002	10.09.2021 13:42:51 BEYER-MENZEL
	1	755				S4DK900127	Development Payment Cards	S4D_P00002	10.09.2021 13:31:48 BEYER-MENZEL

Use toolbar Search button to search for changes within versions

**Category Flag (1) 4 Entries found**

Category Short Descrip.
I Version created by import
S Version created due to System request
U Version created due to a User request
V Version created at request release

4 Entries found

**DB**

**Development DB**  
Active and modified versions

**Version DB**  
Versions saved by version management

Versions of SAP repository objects can be compared or restored.

When a workbench transport request is released, a new version of each repository object in the transport request is written to the version database, which contains a complete change history of all repository objects.

In addition to the versions created automatically by the release of transport requests, users can also create temporary versions at any time. To do this, in the maintenance transaction for

the repository object, for example transaction SE38 (*ABAP Editor*), choose the menu path *Utilities* → *Versions* → *Generate Version*.

You can access the version management by choosing the menu path *Utilities* → *Versions* → *Version Management* from:

- *Object Navigator / Repository Browser* (transaction SE80) when displaying / changing a repository object.
- Display and maintenance transactions for all repository objects, for example *ABAP Editor* (transaction SE38), *ABAP Dictionary* (transaction SE11) or *Function Builder* (transaction SE37).

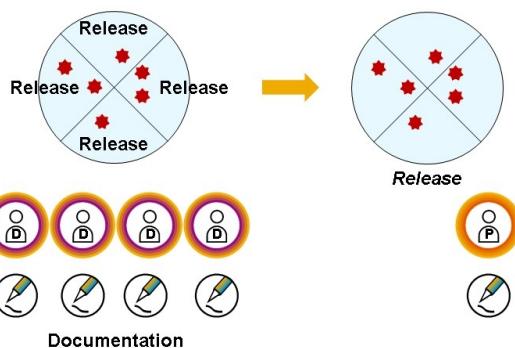
In the version overview, the active and modified (that is: inactive) versions are displayed in the **development database**, and versions saved as a result of released transport requests are displayed in the **version database**.



#### Note:

The version database resides in the development system. Versions cannot easily be transported between SAP systems. If the development system is removed from the SAP system landscape, all versions in the version database are lost.

## Releasing Transport Requests



#### Releasing the transport request:

- Writes a version to the version database for the contained repository objects
- Unlocks the repository objects
- Exports contained objects to the file system – only if the transport request is a transportable transport request

#### Releasing a task requires:

- Contents in the task

#### Releasing a transport request requires:

- Release of all not-empty tasks of the transport request
- Documentation (if mandatory)

Figure 66: Releasing Transport Requests

After the development tasks are completed, a transport request is released so that the contents can be transported to subsequent SAP systems.



#### Note:

Releasing a task or transport request requires the developer to have the necessary authorization.

Releasing a transport request requires that all (non-empty) tasks are documented (if mandatory) and released.

A released task can no longer be changed, but additional tasks for the same transport request can be created to change the objects in the released task, as long as the transport request itself is not released. A released task cannot be deleted. If a task is empty, it is deleted when the transport request is released.

- Releasing a **transportable** transport request records a version of all the repository objects included in the transport request (for object types that are included in the version management), and then exports the objects, that is, copies the objects from the database to a file in the transport directory.
- Releasing a **local** transport request records a version of the repository objects included in the transport request (for object types that are included in the version management), but does **not** export them to file system level.



#### Note:

As of some patch level for *tp* version 381.183.11 and kernel 773 or higher, cofiles are written on operating system level even for local transport requests.

## Export Logs

After releasing a transportable transport request, review the transport logs to ensure that the export was successful.

**Log Overview for S4DK900134**

- S4DK900134 Development Contracts
  - S4D Development System
    - Checks at Operating System Level 10.09.2021 08:27:20 (0) Completed
    - Pre-Export Methods 10.09.2021 08:27:23 (0) Completed
    - 200 Selection for Import 10.09.2021 08:27:25 (0) Completed
    - 100 Export 10.09.2021 08:27:25 (0) Completed
    - Import steps not specific to transport request
  - S4Q Quality Assurance System
    - 100 Selection for Import 10.09.2021 08:27:25 (0) Completed
    - Import steps not specific to transport request

**Return code (0) means "everything OK"**

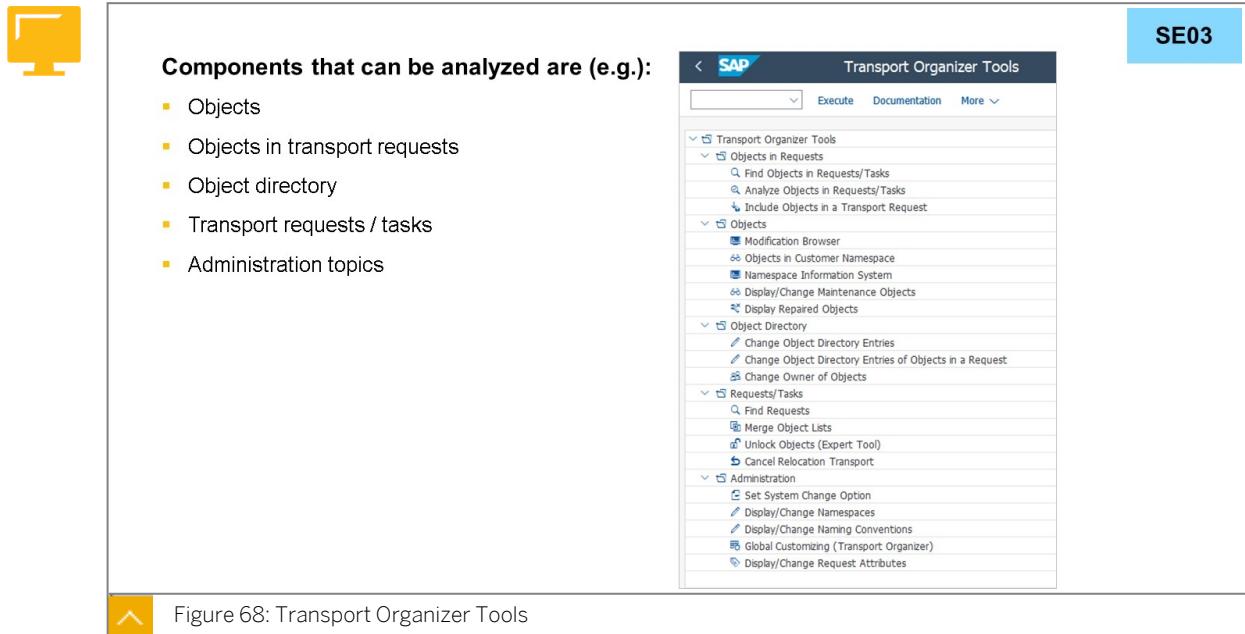
To access the transport logs from the initial screen of the *Transport Organizer* (transaction code **SE09**), choose *Display*, select a transport request, and choose *Goto → Transport Logs*.

In the transport logs, the following return codes indicate the success or failure of the export:

- 0: The export was successful.
- 4: A warning was issued but all objects were exported successfully.

- 8: An error in an object occurred. Whether the objects are considered as exported successfully depends on *tp* parameter settings.
- 12 or higher: A critical error has occurred, generally not caused by the objects in the transport request, but by the transport tools themselves.

## Transport Organizer Settings



The *Transport Organizer Tools* are a collection of tools that support the work with the *Change and Transport System (CTS)*. These tools can be accessed either by the corresponding button in the *Transport Organizer* (for example, transaction **SE09**) or by starting transaction **SE03** directly.

You can find documentation of the functions by selecting the corresponding line and using the *Documentation* button, shown in the figure above. A documentation of the tools can also be accessed by right-clicking them.

The options available under *Global Customizing (Transport Organizer)* provide customers with several options that can be used for:

- Displaying transport errors at log on.  
To cause transport errors to be displayed to transport request owners when they log on, set *Transport error display at logon* to *globally activated* or to *set by user*.
- Applying objects checks to objects contained in transport requests before the transport request is exported to a file in the transport directory.

Object checks identify and display errors found in workbench transport requests. Errors, such as program syntax errors, are identified and displayed when a transport request is released, but not yet exported. More detailed information about the error can be found by simply double-clicking the error on the displayed report.

The screenshot shows two SAP application windows side-by-side. The left window is titled 'General Settings for Transport Organizer' under 'Global Customizing (Transport Organizer)'. It contains several configuration sections: 'Transport error display at logon' (with options 'globally activated', 'globally deactivated', and 'set by user'), 'Object checks at request or task release' (with options 'globally activated', 'globally deactivated', and 'set by user'), 'Object checks (if activated) at release of' (with checkboxes for 'Workbench Request', 'Task', and 'Transport of Copies'), 'Check existence of task documentation at task release' (with options 'globally activated', 'globally deactivated', and 'set by user'), and 'Request Release' (with a checkbox 'Release in background only:' and a field 'Server for Background Release:'). Below these settings are two circular icons: one with a person icon and another with a document icon.

The right window is titled 'General Settings for Transport Request Attributes' under 'Global Customizing (Transport Organizer)'. It displays a table of attributes with their descriptions and mandatory status. The table includes columns for 'Attribute', 'Short Description', 'Mandatory Attribute', and 'Mandatory Value'. Some attributes listed include 'EXPORT\_TIMESTAMP', 'GIT\_BRANCH', 'ONLINE\_IMPORT\_COMPATIBLE', 'SAPCOMPONENT', 'SAPCORR', 'SAPIMG', 'SAPHNOTE', 'SAPOSS', 'SAPTEST', 'SAP\_CCTS', 'SAP\_CWB\_USER', 'SAP\_CTS\_AVCS\_TARGET', 'SAP\_CTS\_PROJECT', 'SAP\_DEVIC', 'SAP\_KW\_AREA', 'SAP\_KW\_FOLDER', 'SAP\_REPLACE\_TRANSPORT\_REQUEST', 'SAP\_SCTS\_AVCS\_AUTHOR', 'SAP\_SCTS\_AVCS\_AUTHOR\_MAIL', 'SAP\_SCTS\_AVCS\_TARGET', 'SAP\_SCTS\_CLOUD', 'SAP\_SCTS\_CLOUD\_COPY', 'SAP\_TMS\_BS\_PLUGIN\_ID', 'SAP\_TMS\_BS\_PLUGIN\_MAX\_PATOLV', 'SAP\_TMS\_BS\_PLUGIN\_MRP\_PATOLV', 'SAP\_TMS\_BS\_PLUGIN\_PEO\_SC', 'SAP\_TMS\_BS\_PLUGIN\_SC\_REL', 'SAP\_TMS\_BS\_PLUGIN\_VERSION', 'SAP\_TMFLOW', 'SCRM\_TEAM', 'SMB\_BUSINESS\_CHANGE\_PROJECT', 'SMB\_LIFECYCLE\_PROJECT', 'SMB\_NOT\_RELEVANT', 'TAKT', 'TAKT\_END', 'TAKT\_START', and 'TCI'. The 'Mandatory Value' column contains 'X' for several entries.

A blue box labeled 'SE03' is positioned in the top right corner of the right window.

Figure 69: Transport Organizer Settings

You can activate or deactivate the display of transport errors at logon and object checks in two different ways:

- Globally: For this, enter transaction SE03, choose *Administration → Global Customizing (Transport Organizer)* and select *globally (de)activated* for the corresponding area.
- For a specific user: For this, enter transaction SE03, choose *Administration → Global Customizing (Transport Organizer)* and select the option *Set by user* for the corresponding area.

Then, for a logged-on user, enter the *Transport Organizer* (for example, transaction SE09) and choose *Settings → Transport Organizer* from the menu. Here, the options *Display Transport Errors at Logon to SAP System*, *Check Objects at Request Release*, and *Check for Task Documentation at Release* can be set by the user that is logged on (provided the settings in transaction SE03 allow this).

You can use the *Transport Organizer Tools* for many different tasks concerning objects in tasks and requests. These tools also allow access to several administrative transactions, which can also be accessed in other ways.

## Authorizations for Change Management



Profile	Authorization
	<b>S_TMW_ADMIN</b> <b>S_TMW_OPERA</b> <b>S_TMW_PRJALL</b> <b>S_TMW_IMPORT</b> Administrator in CTS (All Authorizations) Operator in CTS (All Transport Authorizations) Release Any Transport Tasks and Requests Import Transport Requests
	<b>S_TMW_CREATE</b> <b>S_TMW_PROJECT</b> Create Transport Requests Release Own Transport Tasks and Requests
	<b>S_TMW_DEVELO</b> Release Own Transport Tasks

Figure 70: Authorizations for Change Management

As we have seen before, different tools and concepts exist for managing customizing and workbench transport requests in a controlled way. By specifying the customizing and development roles for that SAP system, you usually define how these tools are used as well as the responsibility for changing customizing and repository objects in an SAP system.

In the most cases, the authorization split can be done as shown in the figure above:

**Super user (for example, SAP system administrator)**

Has all authorizations related to transport requests and tasks.

**Project leader**

Can create and manage transport requests and tasks.

**Developer**

Can use only existing transport requests.

**End user (not shown in the figure above)**

Has display authorization only.



### LESSON SUMMARY

You should now be able to:

- Explain object locking and version management
- Describe the process of releasing a transport request that contains repository objects
- List selected Transport Organizer tools and explain their use
- List authorizations for software logistics



# Modifying SAP Objects

## LESSON OVERVIEW

In this lesson, you will learn about modifying repository objects from the SAP standard and the factors that have to be taken care of when making changes to these objects.

### Business Example

The purpose of your development work is to introduce new or improved functions into the production system. This may require the creation of new objects and the modification of existing objects in the SAP system. Developers can use the *Modification Assistant* in the ABAP editor to make changes to repository objects of the SAP standard, such as ABAP programs. The *Modification Assistant* reduces the amount of effort required when applying SAP Support Packages and when performing an upgrade of an SAP system.



## LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Explain the difference between an original and a copy and the consequences for transport tasks
- Explain the process flow for modifications

## Creating and Modifying Objects

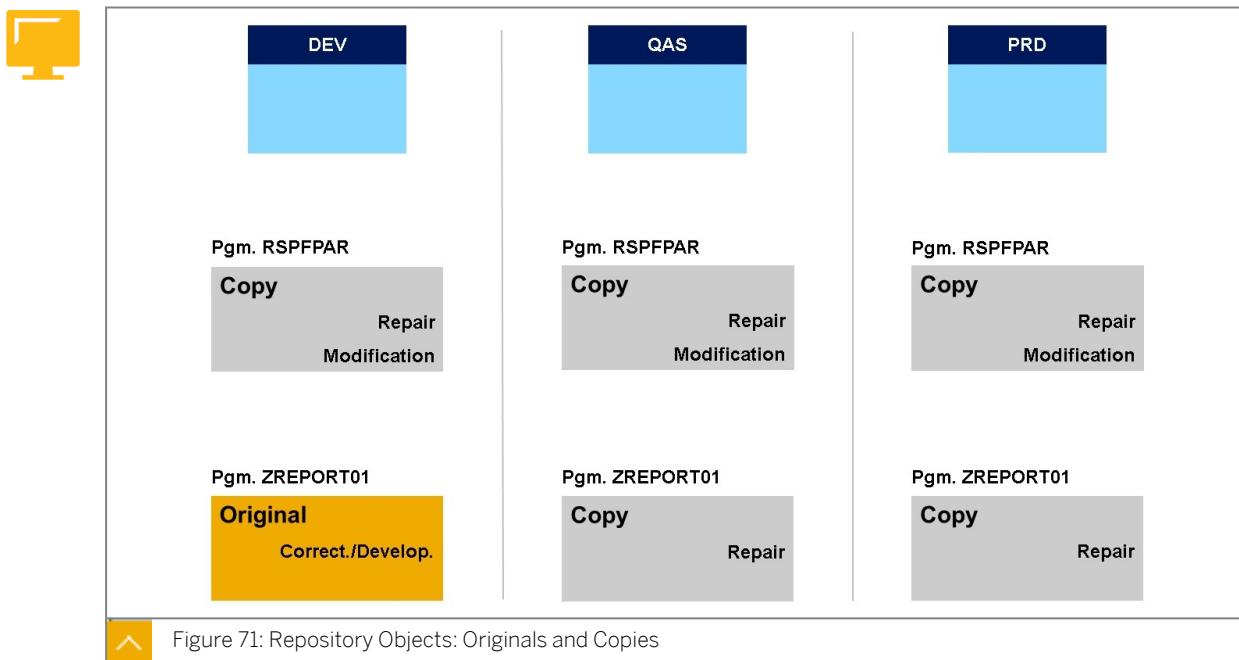
Repository objects should only be changed in their original system. An object is the **original** in only one SAP system: the SAP system where it has been created. All other SAP systems may only contain **copies** of the object. This ensures that changes to repository objects are only made in the respective development system.

When a repository object is transported to a subsequent SAP system, it exists there as a copy. Changes to copies, which are referred to as **repairs**, should be made only in exceptional cases. In all customer SAP systems, including the customer's development system, SAP-delivered repository objects are copies. The original of those objects exists at SAP in the development system. Repairs to SAP objects are referred to as **modifications**.



### Note:

The system change options (in transaction SE06) control which repository objects can be modified in the SAP system. Only in the development system, the system change option should be set to permit changes to customer-developed objects and, if necessary, SAP objects.



When changing a copy, a *Repair flag* is set for this object to prevent the object from being overwritten by imports into this SAP system before the assigned transport request has been released. The object is locked exclusively in a task of type *Repair*.

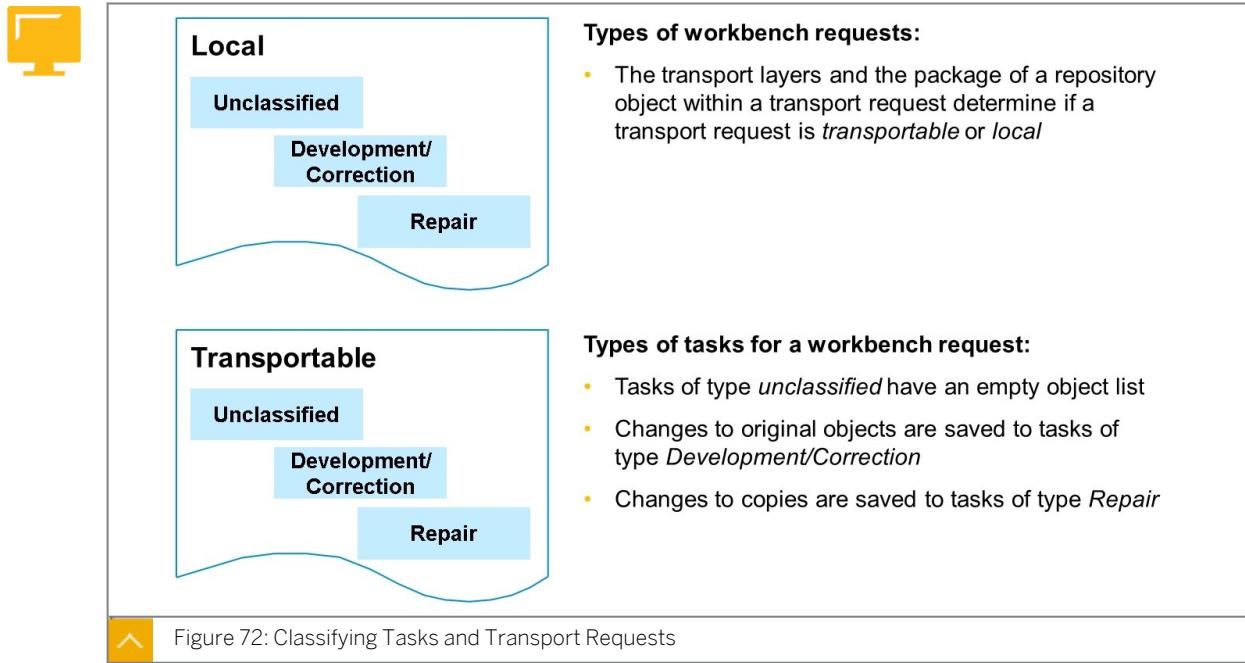
### Classification of Tasks and Transport Requests

A distinction is made between two types of development tasks: *development/correction*, that is changes in the original system, and *repair*, that is changes in an SAP system other than the original system.

The package of the repository object determines whether the transport request type is *transportable* or *local*. If the assigned package is assigned to a valid transport layer, that is, a transport layer which is assigned to an existing consolidation route, a transport request of type *transportable* is used. Otherwise, a *local* transport request is used.

Objects saved as local objects cannot be transported. The only way to transport such an object is to assign it to another package: one that has an existing transport layer assigned.

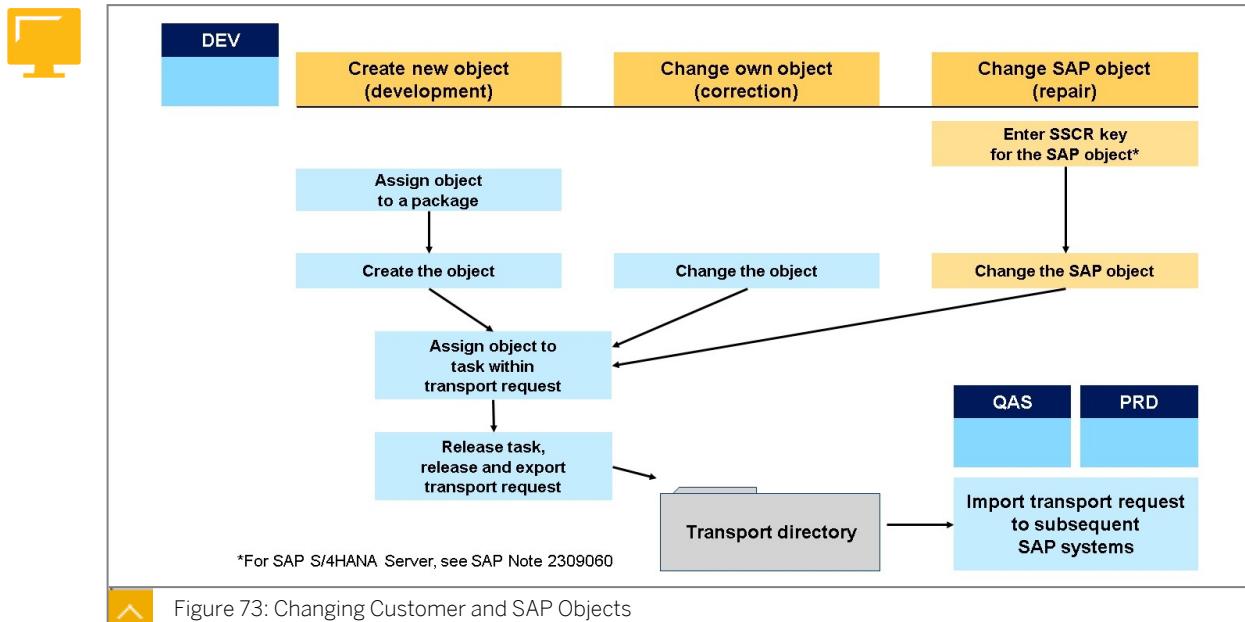
Therefore, the package of a repository object determines whether a transport request of type *Transportable* or *Local* is required.



There are transportable and local workbench requests, depending on the package assignment of the objects inside. They may contain the following three types of tasks:

- Tasks of type *Unclassified* have empty object lists.
- Tasks of type *Development/Correction* contain originals.
- Tasks of type *Repair* contain copies.

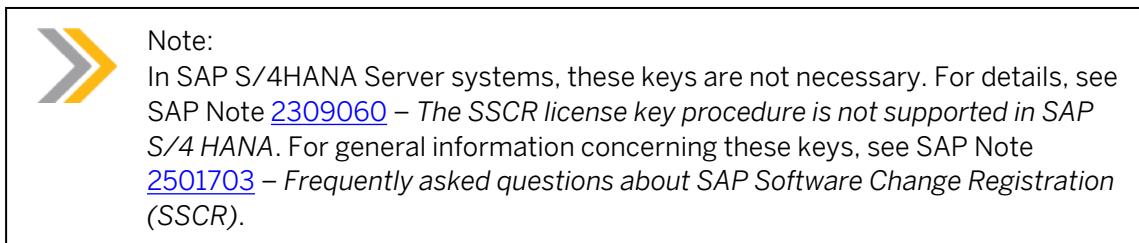
## Changing SAP Objects



Before starting a **modification**, you should ask if it is really necessary to modify the standard programs or if an enhancement can also be done by means of SAP extension techniques (for example, BAdI, appends, user exits, enhancements). Note that the modification may have to be adjusted if SAP delivers this object in a newer version, for example, in an SAP Support

Package or during a release change. That will cause additional effort for maintaining the modification.

Before changing any repository object (customer and SAP objects), you have to be registered as a developer in SAP *Software Change Registration* (SSCR). In case you want to perform a modification (SAP standard object), you have to register each SAP object that you intend to change as well. Registering the object gives you an access key which you apply to the object.

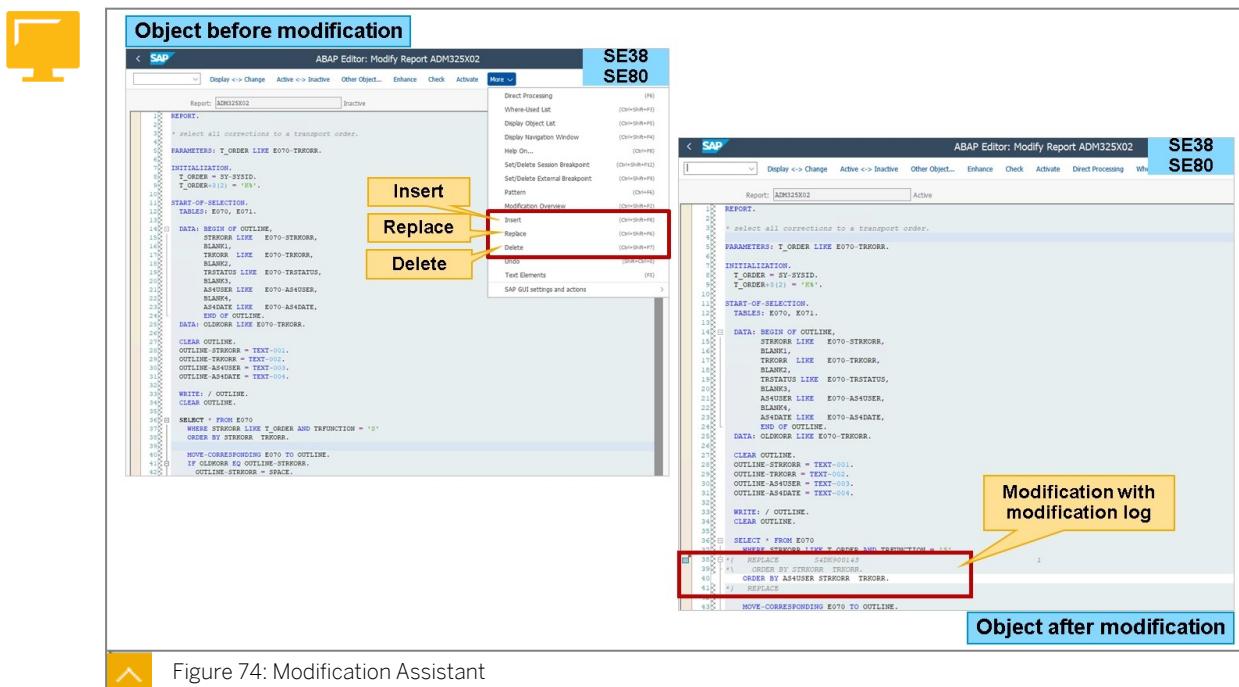


When performing a modification, the *Transport Organizer* prompts you for a transport request in the same way that it does when you make changes to customer-owned objects. Because the object is not an original in your SAP system, it will be assigned to a task of type *Repair*.

All SAP repository objects are assigned to SAP-defined packages. All new customer repository objects must be assigned to a customer-created package. Packages are used to group objects in the project and transport them along the same transport route.

When the developer releases the task of a transport request, the lock on the object is transferred to the transport request.

At the end of the project development, the transport request is released. This releases all locks and records a version of the changed objects, both SAP and customer objects, in the version database.



To support the different levels of modification, SAP provides tools such as the IMG, Business Add-Ins (BAdIs), transaction CMOD (*Project Management of SAP Enhancements*), and the ABAP development environment.

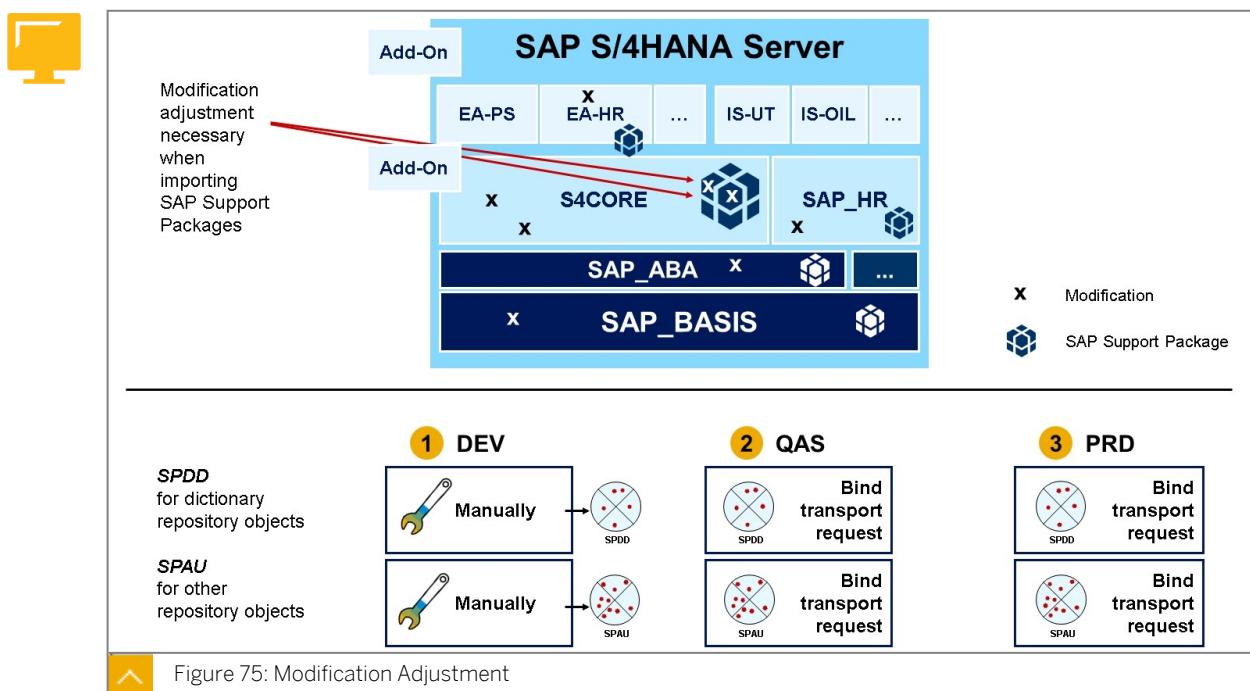
The *Modification Assistant* helps you to modify SAP standard delivered objects and to simplify the modification adjustment. When you change SAP objects in an ABAP Workbench editor, you branch into a special modification mode. Originals are initially protected in this mode and are only changed when you click the corresponding buttons (see the figure above).

The *Modification Assistant* logs all modifications that are made to the object. The log is easy to read and dramatically reduces the amount of effort needed for a modification adjustment.

In addition, the *Modification Assistant* works with the most commonly used ABAP development tools such as *ABAP Editor*, *Screen Painter*, *Menu Painter* and *Text Elements* maintenance.

The *Modification Assistant* is **not** used for changes made to dictionary objects, objects such as global interfaces and classes and their components. These objects need to be adjusted manually during modification adjustment.

A list of all modified objects in the SAP system can be shown with the *Modification Browser* (transaction **SE95**).



If a new version of a modified SAP repository object is imported into the customer SAP system, as part of an SAP Support Package or an SAP system upgrade, adjustments must be made to the modified SAP object.

Before applying an SAP Support Package or SAP enhancement package or performing an SAP system upgrade, all open repairs must be confirmed and released. This check is performed by the update / upgrade check routines of the *SAP Support Package Manager / SAP Software Update Manager*.

To adjust the ABAP Dictionary repository objects, use transaction **SPDD**. To adjust other repository objects, use transaction **SPAU**.



**Caution:**

The adjustment of modifications is neither automatic nor trivial. Customers must decide which development efforts and modifications they wish to retain, bearing in mind that the new SAP functions may quite possibly make retaining the modification unnecessary.



### LESSON SUMMARY

You should now be able to:

- Explain the difference between an original and a copy and the consequences for transport tasks
- Explain the process flow for modifications

## Learning Assessment

- From a technical point of view, a customizing setting is table contents in one or more tables.

*Determine whether this statement is true or false.*

- True
- False

- Which of the following statements are true concerning transport requests for customizing?

*Choose the correct answers.*

- A A transport request can consist of one or more tasks.
- B All tasks of a transport requests must be assigned to the same owner.
- C You can use the *Transport Organizer* (transaction SE09) to create transport requests.
- D The owner of the transport request must be the owner of at least one task of the transport request.
- E A transport request always has exactly one owner.

- Where is a CTS project mainly used in the context of transport requests?

*Choose the correct answer.*

- A To group transport requests
- B To define the target system of transport requests
- C To assign tasks to transport requests
- D To release transport requests

4. When performing customizing that should be transported, steps must be performed in a certain sequence. Bring the following activities into the right sequence, regarding the customizing procedure.

*Arrange these steps into the correct sequence.*

- Release the transport request
- Assign customizing settings to a task of a transport request
- Start the customizing transaction
- Release all tasks of the transport request
- Import the transport request into subsequent SAP systems

5. To transport customizing settings, a *customizing request* is always used.

*Determine whether this statement is true or false.*

- True
- False

6. Which of the following statements are correct for packages in the context of development and transport?

*Choose the correct answers.*

- A Repository objects can be assigned to a package
- B A package can be assigned to a software component
- C A package can be assigned to a transport layer
- D Customizing objects can be assigned to a package
- E A repository object can be assigned to several packages

7. Which object combinations can be transported within one single workbench request?

*Choose the correct answers.*

- A Any custom developed objects assigned to different packages if the packages are assigned to the same transport layer
- B Custom developed objects and SAP standard objects and if the packages of all objects are assigned to transport layers that point to the same target system (and client)
- C Any custom developed objects if they are assigned to custom developed packages
- D Any SAP standard objects (independent of the software components their packages are assigned to)

8. You are changing a customer developed program. The program is assigned to a transport request. When is the object lock released?

*Choose the correct answer.*

- A When the transport request is released
- B When the task which the program is assigned to is released
- C When the program is activated
- D When the program is generated

9. Which of the following actions can trigger the creation of a version for a repository object in the version database?

*Choose the correct answers.*

- A Releasing the corresponding transport request ('' version)
- B Importing the corresponding transport request ('I' version), if the corresponding tp parameter is set correctly
- C Creating a (temporary) version manually ('U' version)
- D Saving and activating the object manually ('A' version)

10. What tasks can you perform by using the *Transport Organizer Tools* (transaction SE03)?

*Choose the correct answers.*

- A Search for objects in transport requests
- B Unlock objects
- C Set general transport request attributes
- D Create transport requests
- E Release transport requests

11. Creating and changing a customer repository object in the (original) development system is recorded in a task of type *Development/Correction*, changing an SAP standard object in the development system is recorded in a task of type *Repair*.

*Determine whether this statement is true or false.*

- True
- False

12. Which status can a task have within a transportable transport request depending on its contents?

*Choose the correct answers.*

- A Unclassified
- B Development/Correction
- C Repair
- D Local

13. When modifying an SAP standard object, modification adjustment is always necessary when applying the latest SAP Support Packages.

*Determine whether this statement is true or false.*

- True
- False

14. When modifying an SAP standard object, this is always done under the control of the *Modification Assistant*.

*Determine whether this statement is true or false.*

- True
- False

## Learning Assessment - Answers

- From a technical point of view, a customizing setting is table contents in one or more tables.

Determine whether this statement is true or false.

- True  
 False

You are correct! From a technical point of view, a customizing setting is table contents in one or more tables.

- Which of the following statements are true concerning transport requests for customizing?

Choose the correct answers.

- A A transport request can consist of one or more tasks.  
 B All tasks of a transport requests must be assigned to the same owner.  
 C You can use the *Transport Organizer* (transaction SE09) to create transport requests.  
 D The owner of the transport request must be the owner of at least one task of the transport request.  
 E A transport request always has exactly one owner.

You are correct! A transport request for customizing can consist of one or more tasks, it can be created with the *Transport Organizer*, and it always has exactly one owner. The different tasks of a transport request can be assigned to different owners. The owner of the transport request does not need to own a task in the transport request.

3. Where is a CTS project mainly used in the context of transport requests?

*Choose the correct answer.*

- A To group transport requests
- B To define the target system of transport requests
- C To assign tasks to transport requests
- D To release transport requests

You are correct! A CTS project is used to group transport requests, for example, to import them together or to perform the Quality Assurance (QA) procedure for one whole project. The target system is defined within the transport request itself. Tasks are defined within the transport request. Transport requests are released by using the *Transport Organizer*.

4. When performing customizing that should be transported, steps must be performed in a certain sequence. Bring the following activities into the right sequence, regarding the customizing procedure.

*Arrange these steps into the correct sequence.*

- 4** Release the transport request
- 2** Assign customizing settings to a task of a transport request
- 1** Start the customizing transaction
- 3** Release all tasks of the transport request
- 5** Import the transport request into subsequent SAP systems

You are correct! First you start the customizing transaction and then you assign the customizing settings to a task of a transport request (that should have been created before). You then release all tasks of this transport request, followed by releasing the transport request, and then you can import the transport request into the subsequent SAP systems.

5. To transport customizing settings, a *customizing request* is always used.

*Determine whether this statement is true or false.*

- True
- False

You are correct! Depending on the type of customizing, a *customizing request* is used for client-specific customizing and a *workbench request* is used for cross-client customizing.

6. Which of the following statements are correct for packages in the context of development and transport?

*Choose the correct answers.*

- A Repository objects can be assigned to a package
- B A package can be assigned to a software component
- C A package can be assigned to a transport layer
- D Customizing objects can be assigned to a package
- E A repository object can be assigned to several packages

You are correct! Repository objects can be assigned to a package, a package can be assigned to a software component, and a package can be assigned to a transport layer. Customizing objects can not be assigned to a package. A repository object can not be assigned to several packages.

7. Which object combinations can be transported within one single workbench request?

*Choose the correct answers.*

- A Any custom developed objects assigned to different packages if the packages are assigned to the same transport layer
- B Custom developed objects and SAP standard objects and if the packages of all objects are assigned to transport layers that point to the same target system (and client)
- C Any custom developed objects if they are assigned to custom developed packages
- D Any SAP standard objects (independent of the software components their packages are assigned to)

You are correct! You can transport with one single workbench request the following: Any custom developed objects assigned to different packages if the packages are assigned to the same transport layer, custom developed objects and SAP standard objects if the packages of all objects are assigned to transport layers that point to the same target system, and any SAP standard objects independent of the software components their packages are assigned to. You cannot transport any custom developed objects with one single workbench request if they are assigned to custom developed packages, or if their packages are assigned to different transport layers pointing to different transport target systems (and clients).

8. You are changing a customer developed program. The program is assigned to a transport request. When is the object lock released?

*Choose the correct answer.*

- A When the transport request is released
- B When the task which the program is assigned to is released
- C When the program is activated
- D When the program is generated

You are correct! The object lock is released at the time when the transport request is released. Releasing the task which the program is assigned to, activating the program, or generating the program does not release the object lock.

9. Which of the following actions can trigger the creation of a version for a repository object in the version database?

*Choose the correct answers.*

- A Releasing the corresponding transport request ('' version)
- B Importing the corresponding transport request ('I' version), if the corresponding tp parameter is set correctly
- C Creating a (temporary) version manually ('U' version)
- D Saving and activating the object manually ('A' version)

You are correct! A version for a repository object in the version database can be created by releasing the corresponding transport request, by importing the corresponding transport request, and by creating a (temporary) version manually. No version is created just by saving and activating the object manually.

10. What tasks can you perform by using the *Transport Organizer Tools* (transaction SE03)?

*Choose the correct answers.*

- A Search for objects in transport requests
- B Unlock objects
- C Set general transport request attributes
- D Create transport requests
- E Release transport requests

You are correct! You use the *Transport Organizer Tools* for searching for objects in transport requests, unlocking objects, and setting general transport request attributes. You use the *Transport Organizer* (transaction SE09) for creating transport requests and releasing transport requests.

11. Creating and changing a customer repository object in the (original) development system is recorded in a task of type *Development/Correction*, changing an SAP standard object in the development system is recorded in a task of type *Repair*.

*Determine whether this statement is true or false.*

- True  
 False

You are correct! Creating and changing a customer repository object in the (original) development system is recorded in a task of type *Development/Correction*, changing an SAP standard object in the development system is recorded in a task of type *Repair*.

12. Which status can a task have within a transportable transport request depending on its contents?

*Choose the correct answers.*

- A Unclassified  
 B Development/Correction  
 C Repair  
 D Local

You are correct! A task within a transportable transport request can have the status *Unclassified* (no objects contained), *Development/Correction* (typically: own development contained), and *Repair* (typically: SAP standard object contained). There is no status *Local* for a task, but a transport request can be *local* or *transportable*.

13. When modifying an SAP standard object, modification adjustment is always necessary when applying the latest SAP Support Packages.

*Determine whether this statement is true or false.*

- True  
 False

You are correct! A modification adjustment is necessary only if the SAP Support Packages applied also contain the modified object.

14. When modifying an SAP standard object, this is always done under the control of the *Modification Assistant*.

*Determine whether this statement is true or false.*

- True  
 False

You are correct! It depends on the object type, if the *Modification Assistant* is available.



### Lesson 1

The Transport Process

133

### Lesson 2

Imports Using TMS

139

### Lesson 3

QA Approval Procedure and Transport Proposals

155

### Lesson 4

Import Process

165

### Lesson 5

Monitoring Tools

181

### Lesson 6

Cleaning up the Transport Directory

195

## UNIT OBJECTIVES

- Illustrate the transport process with QA approval procedure
- Compare the different queue-based options to import transport requests
- Explain how to time imports and to define maintenance periods
- Describe the different transport strategies
- Outline the use of Transport of Copies and Relocations
- Use the QA approval procedure
- Explain the idea of the transport workflow
- Analyze the different steps during the import of transport requests
- Explain the use of tp commands

- List selected monitoring tools and explain their use
- Explain the content of the transport directory
- Troubleshoot typical import errors
- Name tp commands to clean up the transport directory

# Unit 4

## Lesson 1

# The Transport Process

## LESSON OVERVIEW

In this lesson, you will learn about the Transport Management System (TMS) in the SAP system and learn how it is involved in the transport process.

## Business Example

As the transport administrator, you should ensure the correct propagation of changes throughout the SAP system landscape by managing the flow of the transport process.

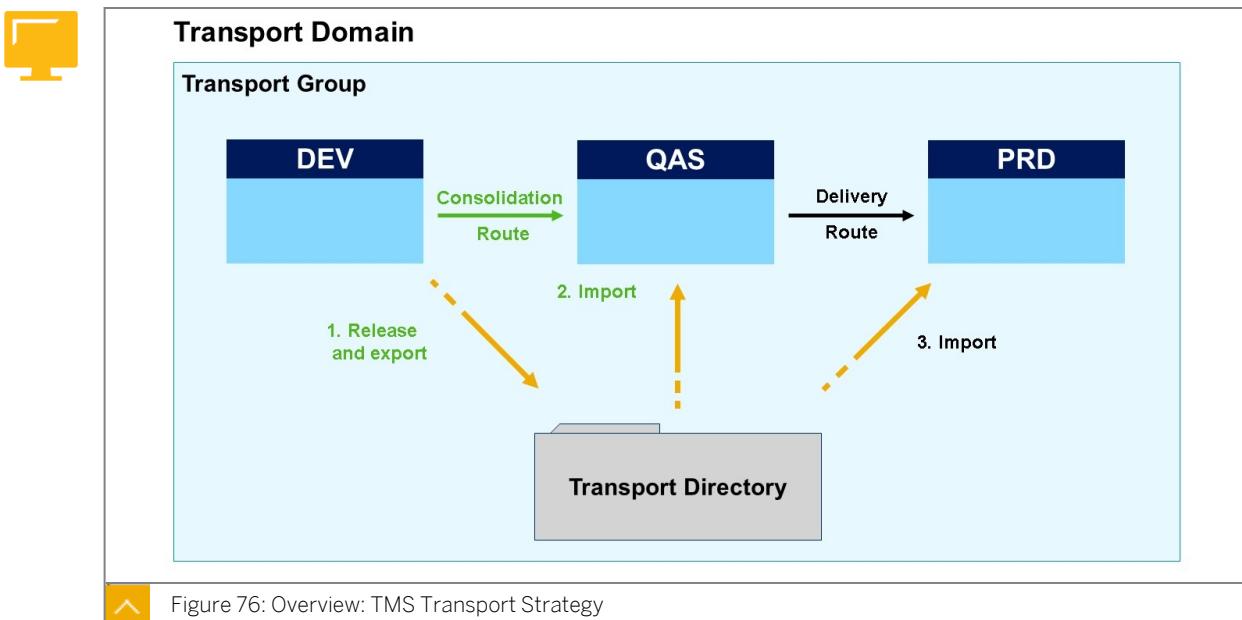


## LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Illustrate the transport process with QA approval procedure

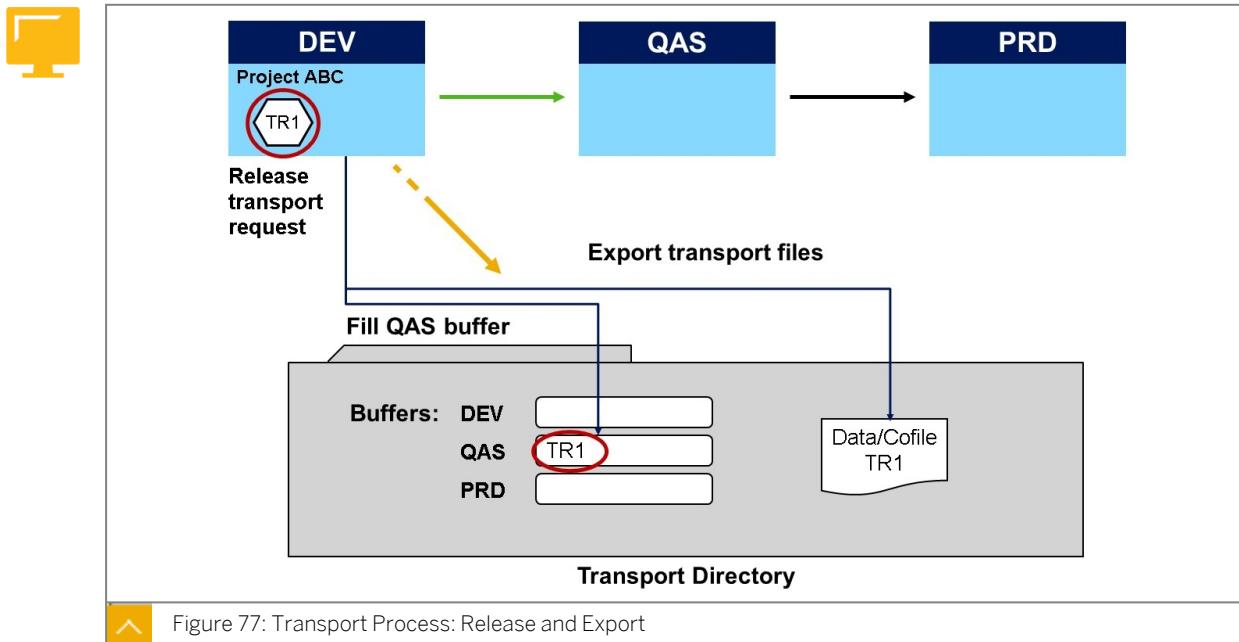
## Steps in Transport Process



In the Transport Management System (TMS), transport requests are propagated along predefined transport routes. You can define multiple consolidation and delivery routes. The import procedure can be performed by any authorized user from within the SAP system. However, most functions in the transaction for the Transport Management System (STMS) are executed by *tp* commands at operating system level, which someone with technical knowledge can also perform manually.

Transport requests to be imported are displayed in the import queue of the target SAP system.

Using TMS, you can import a complete import queue, that is, all transport requests that have been exported from the development system. This ensures that no import errors occur due to missing objects and that newer versions of an object do not get overwritten by older versions.



Let's take a three-system landscape as an example. The first step in the transport process is to release a transport request and export all the associated objects from the database of the development system (DEV) to files in the common transport directory at file system level. For each released transport request, the data is exported to a data file in the subdirectory *data* and a control file is written to the subdirectory *cfiles*.

During export, the entries required for the subsequent import are created in the import buffers of the target system(s), and a test import can be performed.

In the directory *buffer* at file system level, there is an import buffer file for each SAP system in the transport domain. The file is named after the corresponding SAP system ID and contains control information regarding the transport requests to be imported and the order of import.

Several transport control commands can be used to manage the import buffer files at the operating system level. The control information in the import buffer files is read and represented in import queues that can be accessed from the *Transport Management System* (transaction STMS) within the SAP system. An import queue shows the transport requests that are listed in the corresponding buffer file.

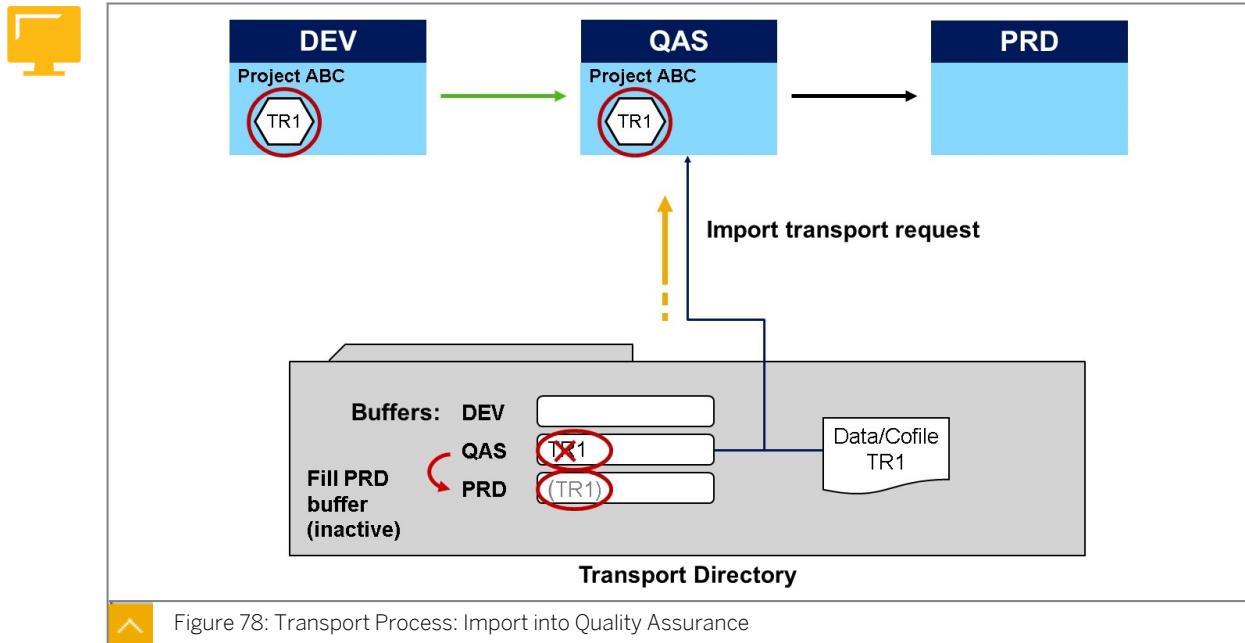


Figure 78: Transport Process: Import into Quality Assurance

Using TMS from within the SAP system, the second step in the transport process is to import all transport requests listed in the import queue of the quality assurance system (QAS). TMS performs the import by starting the transport control program *tp* at operating system level.

After the successful import into the quality assurance system, the transport requests will be placed in the import buffer and import queue of the production system (PRD) and any other delivery system.

**Note:**

If the *Quality Assurance (QA) approval procedure* is activated, the entries will be placed in the import queue of PRD in a way that they cannot be imported without approval.

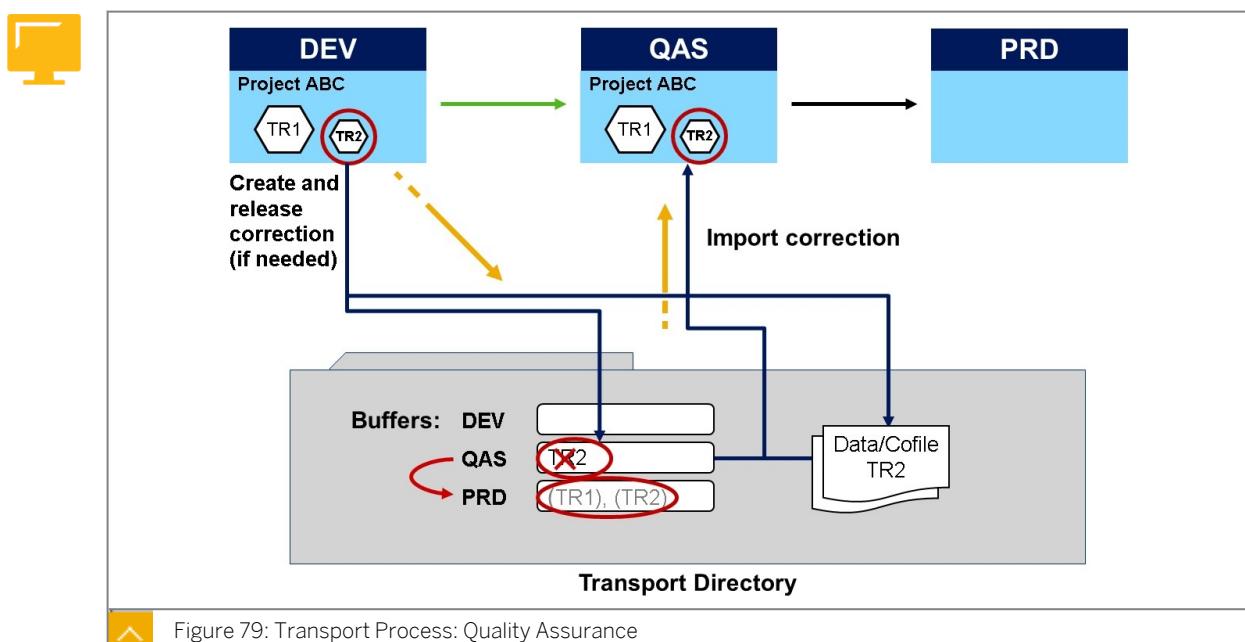


Figure 79: Transport Process: Quality Assurance

After import to the quality assurance system, for example, QAS, the objects need to be tested for possible errors. Errors should be corrected in the development system, for example, DEV, and the changes should again be imported into QAS (see the figure “Transport Process: Quality Assurance”). During the import into QAS, the additional transport request is added to the buffer of the production system PRD.

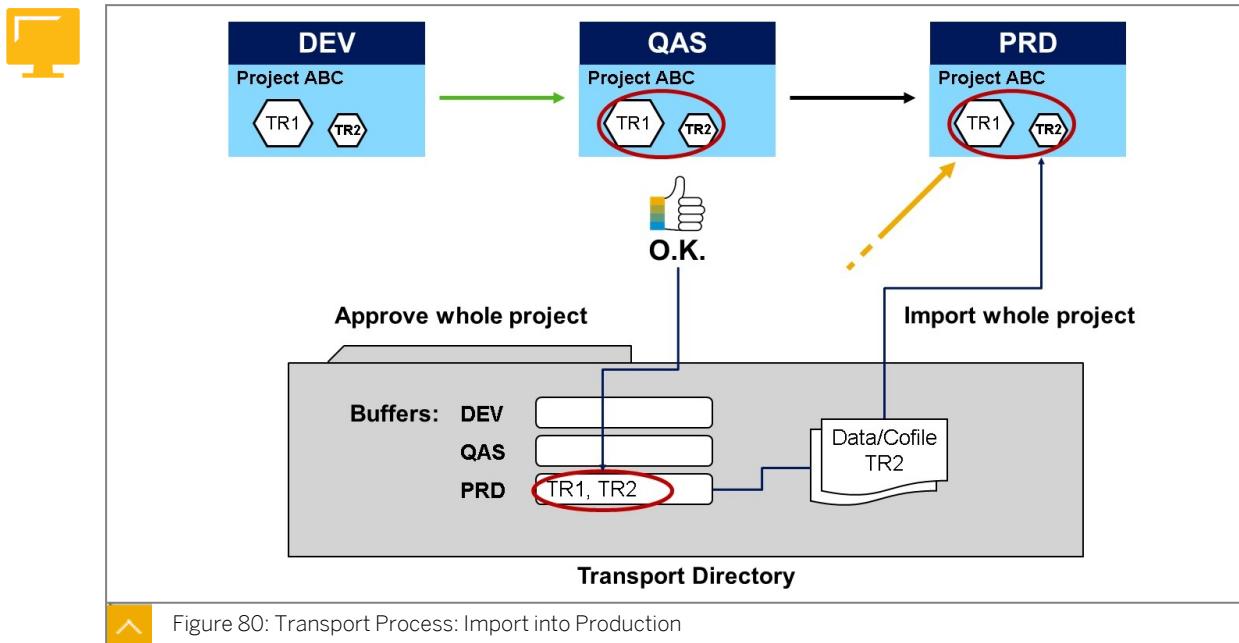


Figure 80: Transport Process: Import into Production

After all the transport requests that were imported into QAS have been thoroughly tested and verified, the transport requests must be approved.



#### Note:

If the quality assurance (QA) procedure is activated, the status of the entries in the import queue of PRD then changes from *inactive* to *active* and the transport requests are ready for import into the PRD.

Using TMS, you can import all transport requests, or simply a first set of approved transport requests, listed in the production system import queue in the given, not manipulated, sequence.



#### Caution:

To ensure that there is no negative effect on production activities in PRD, ensure that the transport requests are imported in the correct sequence.



#### Hint:

You should always try to import complete CTS projects. By doing this, you minimize the risk of forgetting single transport requests and you keep the correct sequence of the transport requests.



## LESSON SUMMARY

You should now be able to:

- Illustrate the transport process with QA approval procedure



# Imports Using TMS

## LESSON OVERVIEW

In this lesson, you will learn about the tools for performing imports using the Transport Management System (TMS). In addition, the different transport strategies will be introduced.

### Business Example

After the project team leader has confirmed and released all required transport requests from the development system, the transport administrator must import the transport requests into the target systems. The transport administrator must follow the guidelines established for the transport strategy and the transport schedule. In this way, the transport administrator can ensure that changes are consistently distributed to all SAP systems in the landscape.



## LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Compare the different queue-based options to import transport requests
- Explain how to time imports and to define maintenance periods
- Describe the different transport strategies
- Outline the use of Transport of Copies and Relocations

## Importing Transport Requests Using TMS

The most important tools used to perform imports using TMS are the import queues that reflect the SAP system-specific import buffers at file system level. The import queues display the transport requests that are to be imported, in the correct order. The import queues of all SAP systems are displayed in each SAP system of the transport domain. You can perform imports to all the SAP systems from any SAP system in the domain.



### Note:

When importing a transport request into a different SAP system than the one you are logged on to, you may need to provide credentials for the SAP system to be imported in.

To access the TMS import overview, use transaction STMS and choose *Overview → Imports* from the menu. The import overview shows the current status of the import queue of each SAP system of the transport domain. If you navigate into an import queue of one SAP system, you can see all transport requests that are to be imported.

The screenshot shows the SAP Import Queue interface for System S4Q. The toolbar includes buttons for Refresh, Sort Ascending, Sort Descending, Select/Deselect Request, and Display Less (which is circled in red). The main area displays a table of transport requests with columns: Number, Request, T, QM, Clt, RC, I, UMS, Owner, Project, Short Text, Ac, CV, ES, and RI. The table contains 11 rows of data. Below the table, several properties are highlighted with arrows pointing to callout boxes: Name, Type, Order for Import, Source Client, Target Client, Return Code, Import Flag, Unconditional Mode, Active, Component Check, Runtime Information, and Queue Entry Status.

Figure 81: Import Queue Information

By choosing *Display More* in the application toolbar, additional columns will be displayed.

The column *CV* (Version Check) indicates whether or not the transport request fits the release and SAP Support Package level of all software components of the target system. If the transport request doesn't fit, it should not be imported. It can be imported at your own risk after analyzing its contents carefully.



**Note:**  
SAP Note [1090842](#) – Composite SAP Note: Cross-release transports lists some technical problems that may occur during a transport between systems with different SAP BASIS releases.



**Hint:**  
The version check would show a wrong release or SAP Support Package level if there is an additional software component, for example an additional plug in, in the export or import system. If this different set of software components is intended, and can't be resolved otherwise, there is a way to exclude a specific software component from the version check. For details, see SAP Note [1742547](#) – Information about component version check in TMS.

With the extended transport control, you could have multiple target clients for one target system. You can choose to see either one row for each target client or only one row per transport request.

To improve performance, data is read from the transport directory only the first time you start TMS. After that, the information shown is buffered in the database. The time stamp in the import overview indicates how recent the data is. The internal buffers of TMS become invalid at midnight. To refresh the data, from with any import queue choose *Edit → Refresh*. It may be more convenient to have the refresh performed periodically in the background. To do this, choose the menu path *Extras → Update All Import Queues* from the *Import Overview* screen. SAP recommends running this refresh on an hourly basis.



Note:

If it takes a long time to finish refreshing the import queue, SAP Note [1924741](#) – *It takes a long time to refresh a system's import queue* may help. For an automating manual process of deleting imported requests from the import buffer, see SAP Note [2461665](#) – *Automating manual process of deleting imported requests from import queue*.

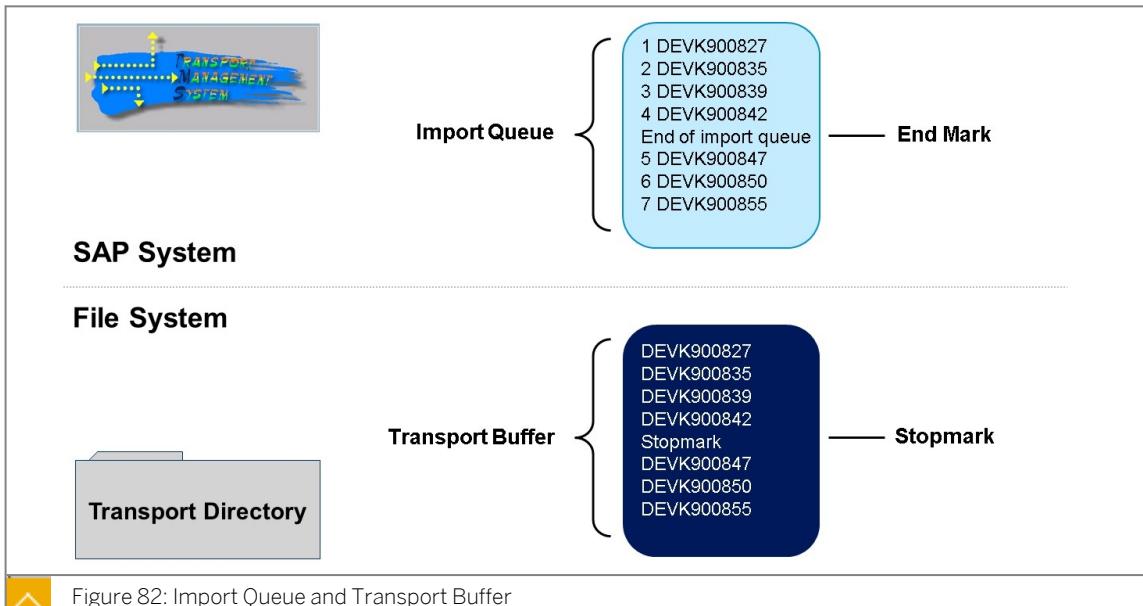


Figure 82: Import Queue and Transport Buffer

The terms *transport buffer* and *import queues* are related. The import queue in the SAP system represents the transport buffer file located in the transport directory. The import queue highlights the requests that will be imported during the next complete import (*import all*). Because of end marks, there may be more transport requests in the transport buffer than those highlighted in the import queue.

The end mark and stopmark are the corresponding markers in the import queue and transport buffer. They indicate that only those transport requests before the mark will be imported by an *import all*. Regardless of how the end mark or stopmark is created, the mark is set both in the transport buffer and in the import queue. In an import queue, an end mark is indicated through the statement “End of import queue”. In a transport buffer, the term **stopmark** is visible. There can only be one end mark or stopmark in each import queue or transport buffer.

To set an end mark/stopmark by closing an import queue, in the import queue choose *Queue → Close* from the menu. This is analogous to the operating system command `tp setstopmark`. The status bar shows the action performed.

To remove an end mark/stopmark (by opening an import queue manually, normally not needed) in the import queue screen, choose the menu path *Queue → Open*. Opening an import queue is analogous to the operating system command `tp delstopmark`.

Using TMS, you can move end marks to any position in the import queue in front of a transport request (by choosing *Queue → Move End Mark* from the menu). This is analogous to the operating system command `tp mvstopmark`.

 **Use the import queue to:**

- **View the status of transport requests and access:**
  - Object lists
  - Documentation
  - Transport logs
- **Close and open the queue, move the end mark**
- **Import:**
  - All transport requests
  - Complete projects
  - Preliminary transport requests
  - Transport requests according to filter settings
- **Add, delete and forward transport requests**



 Figure 83: Import Queue

You can use the import queue to:

- View the status of transport requests.
- Access object lists, documentation, and transport logs.
- Close and open the queue, and move the end mark.
- Import all transport requests, complete projects, preliminary transport requests, and selected transport requests according to filter settings.
- Add, delete, and forward requests.

To keep target SAP systems consistent, you need deadlines to coordinate the release of transport requests by developers. To prevent transport requests released after the deadline from being imported, the import queue of the quality assurance system can be closed. As a consequence, transport requests released after deadline are positioned after an end mark in the queue for the next import. Only the requests before the end mark are imported in the next import. The same is valid for the production system correspondingly.

In exceptional cases, you can forward a transport request to another SAP system before being imported into the defined target system. For example, before being imported into the quality assurance system, a request may need to be rushed to a training system. To prepare the import to a target system outside the predefined transport routes, in the *Import Queue* screen, choose the menu path *Request → Forward → System*.

You can also delete transport requests from or add them to an import queue. But then, object dependencies may cause inconsistencies in the target system after the next import. For example, if you delete a request containing a new data element, the import of all other transport requests containing tables that depend on that data element will fail.



**Caution:**

To avoid these inconsistencies, you are strongly advised not to delete individual transport requests from the import queue. Make your corrections in the development system and release a new transport request.

**Hint:**

Always import all transport requests into the production system, even those with errors, together with their corrections.

**Note:**

If you wish to simulate the import of a transport request, you can perform a *test import*. For more information, see SAP Note [2510475 – Simulate Transport](#).

## Import Strategy

There are different strategies for how to import transport requests. These will be discussed in the following pages:

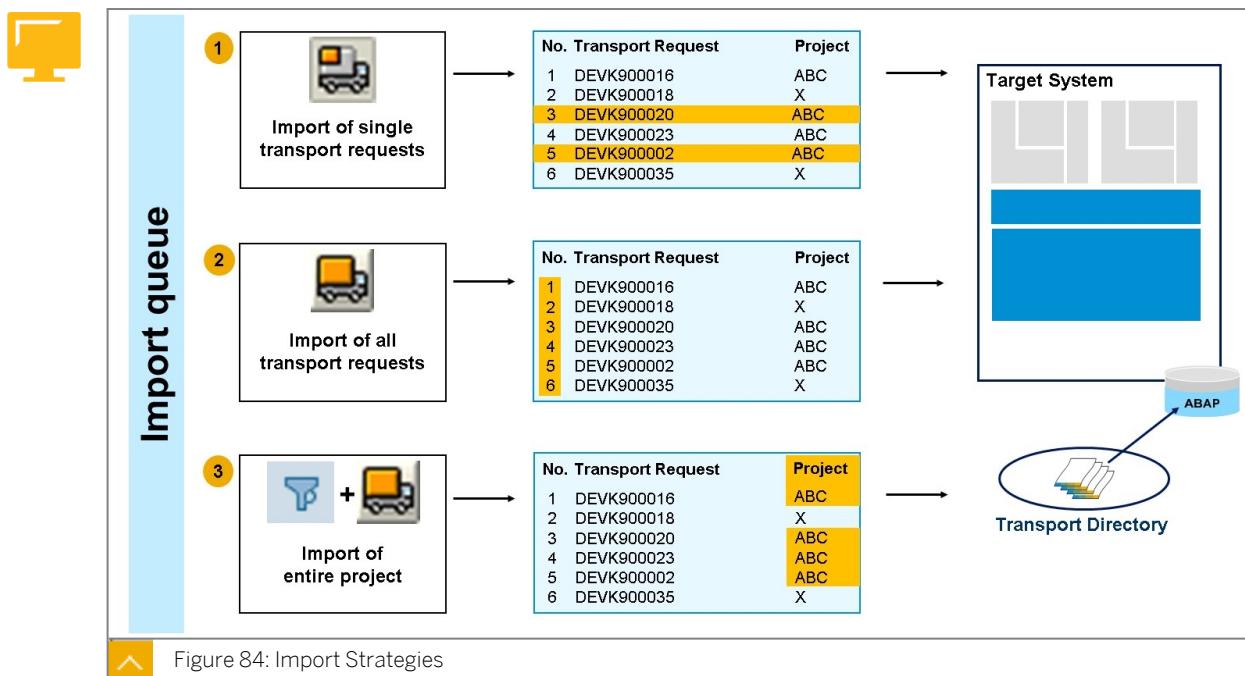


Figure 84: Import Strategies

## Import All Transport Requests

To import all transport requests in the queue (performing what is known as an **import all**), choose the *Import All Requests* button (the “fully loaded truck” icon, see the figure above). The *Start Import* dialog box appears.

If you have configured the *Extended Transport Control*, the target client is fixed. Otherwise you can either choose a target client or keep the default. The number of the default target client is identical to the number of the source client (this is one reason why you should use the same numbers for related “major” clients in all SAP systems).

**Hint:**

If you start the import from an SAP system different from the target system, a logon window of the target system may be displayed and you need to enter credentials for the target system.

In the *Start Import* dialog box, you have several options to control the import:

- In the *Date* tab, you can schedule the import.
- In the *Execution* tab, you can select whether TMS starts *tp* synchronously or asynchronously. Selecting *asynchronously* means *tp* works in the background so that your user session is not blocked for the duration of the import.
- In the *Options* tab, you can select “expert options”, the so-called unconditional modes. The options and their defaults vary according to the selected import method and the configured transport strategy.

The *Import Overview* screen indicates whether the import is running. After the import, the end mark is removed and the queue is opened again automatically. After transport requests have been imported successfully, they are automatically added to the import queue of further target systems (systems that are connected with the help of a delivery route). The configured transport routes therefore specify which transport requests are automatically forwarded to which target systems.

When using the *QA approval procedure* function in this SAP system, all transport requests in the import queue of the subsequent SAP systems are set to *inactive*. If an import containing one or more inactive requests is triggered, the TMS will not perform the import.



**Note:**

When using the *QA approval procedure*, you can only import **all** transport requests into the delivery systems if **all** the transport requests that are ready for import have been checked for all the applicable approval steps (*approved* or *rejected*).

If all the transport requests for one CTS project have been approved, you can import them into the delivery system even if there are still unprocessed or rejected requests for other CTS projects in the work list.

If you perform an import through an *import all*, objects are imported in the correct sequence in which they are listed in the import buffer file on file system. This means that if transport requests near the start of the list and those near the end of the list affect the same objects, the final versions of the objects after import will represent the latest changes. As a result, the incorrect objects do not affect your production environment, they are not really imported at all.



**Note:**

You can deactivate the ability to perform a complete import (*import all*) for each SAP system using the *tp* parameter *NO\_IMPORT\_ALL* or by changing the transport strategy (see the “Transport Strategies” section, later in this lesson).

### Import Complete Project

Before performing the import, SAP recommends setting the end mark to close the import queue. This avoids unintentionally importing other transport requests that may appear in the import queue.

You can set a filter on the import queue to limit the displayed transport requests to transport requests with specific properties so that you can see only those transport requests that belong to a specific project. To set a filter, place the cursor in a row of the import queue and press the *Filters* button in the application toolbar.

To prevent transport requests being imported from unapproved projects, use the filter for column *Project* to import only the transport requests belonging to approved projects.

### Import Single Transport Requests (Preliminary Import)

In contrast to standard imports, preliminary imports are imports of selected single transport requests. SAP strongly recommends only using project-specific imports or full imports because of object dependencies and the risk of inconsistencies when importing individual transport requests. For example, an ABAP program in one transport request can be successfully imported, but the table that it refers to can be in another transport request that has not yet been imported. Until the table is imported, executing the program generates short dumps. Therefore, use preliminary imports only in exceptional cases.

To import single transport requests choose the *Import Request* button on the application toolbar (related to the “partly loaded truck”, see the figure above).

To minimize the risks associated with preliminary imports, the transport request remains in the import queue after the import and is re-imported the next time the entire import queue or the corresponding CTS project is imported. This guarantees the correct import sequences and is defined by the import option *Leave Transport Request in Queue for Later Import*, which, depending on the transport strategy, may be automatically selected.

By default, the TMS will check if transport requests in the import queue depend on transport requests in other projects. The import will only be possible if the predecessor's relationships is not violated. It may be necessary to specify additional options when performing a preliminary import:

- *Leave Transport Request in Queue for Later Import* – this is the default when using transport strategy *mass transports*
- *Import Transport Request Again* – ignore that the transport request has already been imported
- *Overwrite Originals*
- *Overwrite Objects in Unconfirmed Repairs*
- *Ignore invalid transport type*
- *Ignore invalid table class*
- *Skip predecessor relationships*
- *Ignore Invalid Component Version* (a wrong SAP Support Package level or a wrong release)

### Sequence of Import of Objects in Transport Requests

The objects from the transport requests marked for import will be imported as follows:



- All objects of all selected transport requests are merged together.
- Objects are sorted first according to their level (for example, table definitions before programs).
- If an object is included in more than one transport request, only the version in the last transport request is kept after import (according to the sequence in the import queue).



**Hint:**  
This sequence is used for *import all*, *import project* and *import single*.



**Note:**  
In a three-system landscape:

- The import queue of QAS reflects the order of export from DEV
  - The import queue of PRD reflects the order of import into QAS
- This is not identical in all cases, but it is the correct sequence.

## Timing Imports and Maintenance Mode

### Timing Imports

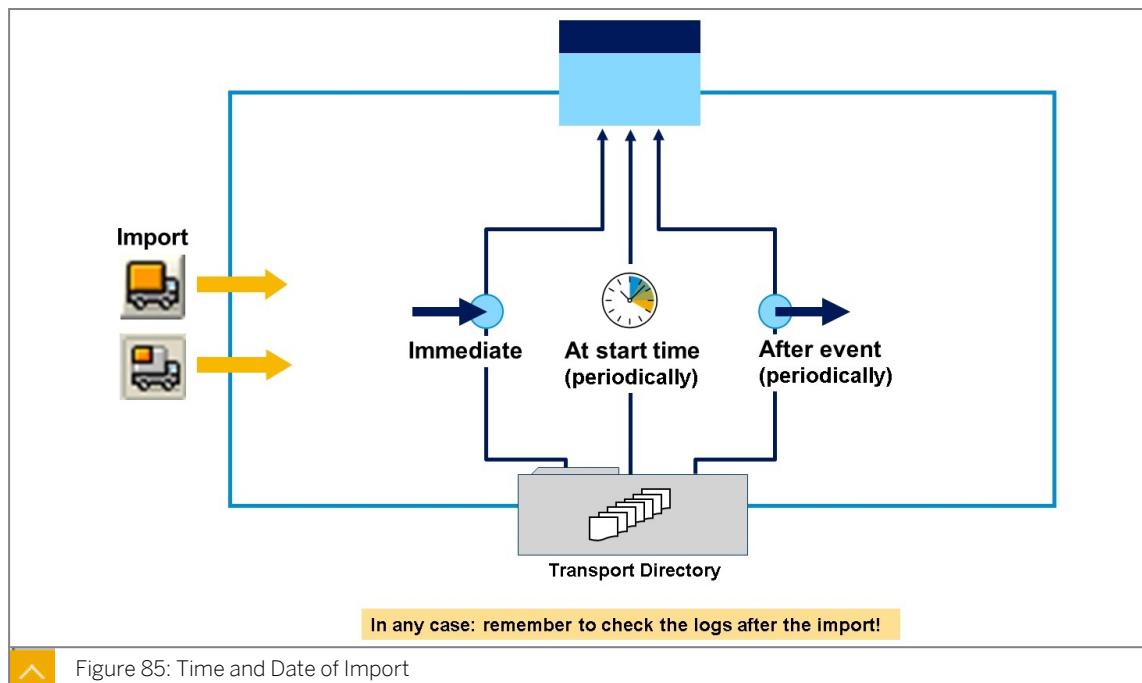


Figure 85: Time and Date of Import

Depending on both the import strategy (*import project*, individual import, *import all*, or special transport workflow) and the SAP release / SAP Support Package level, the available options may vary. When starting an import, you can select the following options in the *Date* tab:

#### Immediate

Selecting this option immediately starts the import.

#### At Start Time

Selecting this option starts the import at the specified time. The import is scheduled as a background job in the target system. If you also enter a date and time in the *No Start After* field, the import is started in the time frame between the times entered in *Planned Start* and *No Start After*. If there is no background process available during this time, an

import does not occur. If you want the import to be performed regularly, you must select a period in the *Period* field.

### After Event

Selecting this option starts the import only after a specified event is triggered. If you select *Execute Import Periodically*, the import is started each time the specified event is triggered. Otherwise, the import is started only the first time the event occurs.



#### Note:

Note that the *Period* field and the *Execute Import Periodically* check box don't exist for individual transports and the special transport workflow.

From the import queue of each SAP system, you can monitor and maintain all planned imports by choosing *Goto → Job Monitor* from the menu.

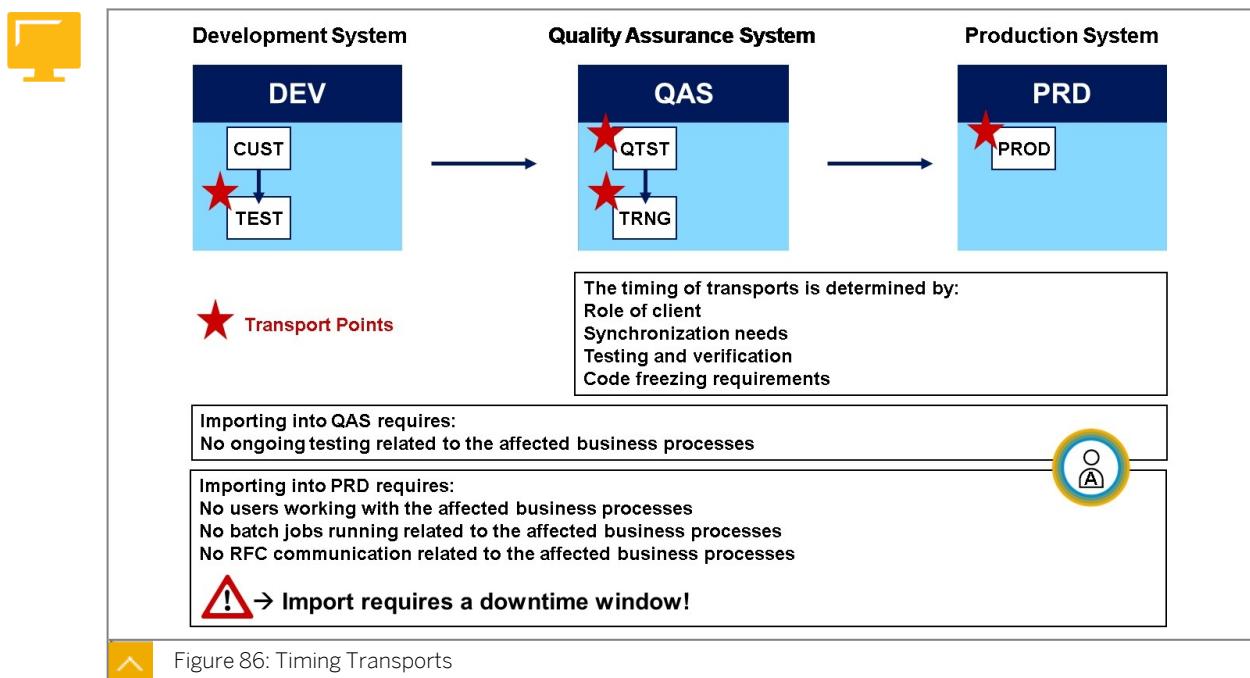


Figure 86: Timing Transports

After the export, a transport request is not automatically imported. It must be manually imported. When planning imports, include enough time to accommodate post-import tasks such as the quality assurance testing. SAP recommends planning imports at regular intervals, such as monthly, weekly, or daily, using *import all* or *import project* into the target system. Frequent imports “on demand” are not recommended.

The following actions have to be considered:

1. Copy the contents of the transport requests into a client in the same SAP system using transaction SCC1 or, as of SAP S/4HANA 2020, SCC1N (unit testing).
2. Release of transport requests.
3. Imports into clients in subsequent SAP systems.

The timing of transport requests is based on the following factors:

- Clients and their roles in your SAP system landscape

- Synchronization requirements, that is, when changes are required in different SAP systems
- Backup prior to the import
- Code freezing requirements

Change management includes creating, releasing, distributing, and verifying transport requests for all SAP systems in the SAP system landscape. Ensure that your distribution procedures are supported by the TMS setup. Determine release, import, and verification requirements for all transport points in your SAP system landscape. For each transport point, define the following:

- When will transports take place?
- Who is responsible for a transport request during its various transport phases?
- How will transports be reviewed before being distributed or redistributed?
- What happens if a transport is successful, but testing proves that its contents are incorrect?
- Are sign-offs required for transporting?

Assign persons to be responsible for all transport steps. Use authorizations to restrict access to transport management tasks and to operating system level.

SAP provides techniques for the distribution and management of changes using transport buffers. These transport buffers indicate the changes to be transported to the target system and the transport order. You must control the transport buffers and the distribution or redistribution process. To verify all imports, review transport logs, establish testing standards and quality assurance procedures, and define sign-off procedures.

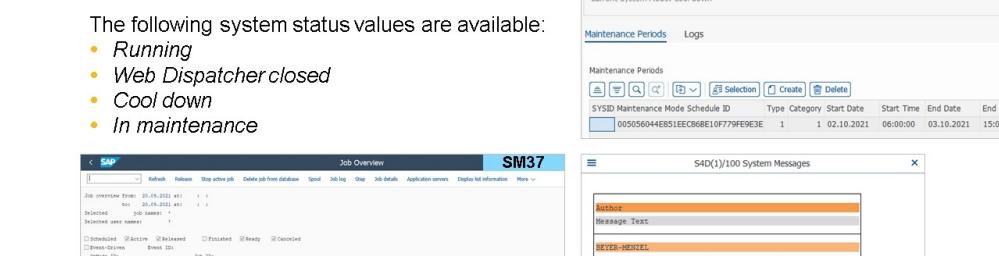
### Maintenance Mode

With SAP S/4HANA, transaction SMAINTENANCE allows you to define a maintenance period. During this maintenance period, normal users will not be able to log on. Only system administrators with a special security policy can connect (transaction SECPOL, attribute *TENANT\_RUNLEVEL\_LOGON\_PRIVILEGE*). In addition, only admin batch jobs will be executed (all other jobs will be on hold).

- The action *Switch to Maintenance* starts a workflow which sets the system to mode *In Maintenance*. The workflow consists of the following modes: *Running* → *Web dispatcher closed* → *Cool down* → *In Maintenance* as follows:
  - *Running*: The system is fully operational for business end users and administrator.
  - *Web dispatcher Closed*: The system is fully operational for business end users and administrators but connections to the Web dispatcher from outside are blocked. That is users connected via the Web dispatcher cannot work any longer.
  - *Cool down*: The system is fully operational for administrators only. Business users are expected to finish their work and log out. At the end of the cool down phase business end users' sessions are terminated.
  - *In Maintenance*: The system is fully operational for administrators only (users who have a special security policy assigned as described above). There are no business end users logged onto the system.

The current duration of the whole workflow is 10 minutes. Switching back to *Running* in this period may lead to errors.

- The action *Switch to Running* starts a workflow which sets the system to mode *Running*. The workflow consists of the modes *In Maintenance* → *Running*. The switch to *Running* is performed immediately.



The screenshot shows the SAP transaction SM37 interface. At the top, there's a navigation bar with 'SAP' and tabs for 'Switch to Maintenance', 'Switch to Running', 'Refresh Display', 'Documentation', 'More', and 'Exit'. Below the navigation bar, the 'System status' section indicates 'Current System Mode: Cool down'. The 'Maintenance Periods' tab is selected, showing a table with columns: Maintenance Schedule ID, Type, Category, Start Date, Start Time, End Date, End Time, and SAP System. One row is visible: '005056044E851EE0968E10F779FE9E3E', Type '1', Category '1', Start Date '02.10.2021', Start Time '06:00:00', End Date '03.10.2021', End Time '15:00:00', and SAP System 'S4D'. The 'Logs' tab is also present. In the main area, the 'Job Overview' section shows a table of scheduled jobs. The 'S4D(1)/100 System Messages' section displays a message: 'The system is set to a maintenance mode. Please log off soon.' with buttons 'Exit' and 'Continue'.

For more information, see the blog *ABAP Platform – Part 3 – what's new for the basis administrators* (<https://blogs.sap.com/2020/06/18/abap-platform-part-3-whats-new-for-the-basis-administrators/>) and SAP Online Documentation for SAP S/4HANA (*Product Assistance*), area *Enterprise Technology* → *ABAP Platform* → *Administrating the ABAP Platform* → *Administration Concepts and Tools* → *Administration of Application Server ABAP* → *Maintenance Mode*.

## Transport Strategies

There are three different transport strategies available:

## Transport Strategies

- Queue-Driven Transports, Mass transports
  - Queue-Driven Transports, Single transports
  - Workflow-driven transports



**Hint:**

By default, the transport strategy is set to *Queue-Driven Transports, Mass transports*.

Queue-Driven Transports, Mass Transports

Mass transports are a good solution if you have a large number of transport requests to administrate and want to automate the process as much as possible. The continuous use of mass transports is the most secure way of keeping your SAP systems synchronized

and consistent. Before you perform a mass transport into your production system, you must check all transport requests in the quality assurance system and confirm their transport into other SAP systems. Use the *Quality Assurance approval procedure* when doing this.

You define *mass transports* as the import method for the relevant SAP systems by choosing the transport strategy *Queue-Driven Transports, Mass transports*.

The administrator can schedule the imports periodically in TMS, or start each import manually. Only import single transport requests (*single imports*) before others in the import queue in special cases.

Transport requests that are imported in advance by the TMS will be imported again during the regular import. You can also use the *transport workflow* to import single imports in advance.

### Queue-Driven Transports, Single Transports

If you want to maintain a production system with only a small amount of specific transport requests, it is best to import single transport requests rather than importing all transport requests waiting for import. Use single transport requests if you have fewer changes to transport and your organization prevents you from having a fixed transport schedule.

This method usually entails extra work for the administrators compared to periodic imports. Developers need to pay extra attention to the consistency and the import order of their transport requests. If a small number of developers are working on a project, or if the developers work very closely with the administrator, they often perform their own single transports.



#### Note:

You can use this import strategy as well if you use project import:

- By removing the *import all* option, you prevent the administrator from accidentally importing all transport request independent of the CTS project.
- You can filter with respect to the CTS project to use import all transport requests belonging to one specific CTS project together.

### Workflow-driven Transports

If you want to perform specific single transports into your SAP systems, but would rather have this done by the SAP system administrator, you may want to use the transport workflow. This method automatically triggers a workflow when you release a transport request. The workflow ensures close communication between development and administration.



#### Hint:

As prerequisite for this, you need to have configured the transport workflow for your SAP system (see lesson “QA Approval Procedure and Transport Proposals”).

## Maintain Transport Strategy

If you want to work with queue driven single transports or workflow-driven transports instead, you will need to change the configuration as follows:

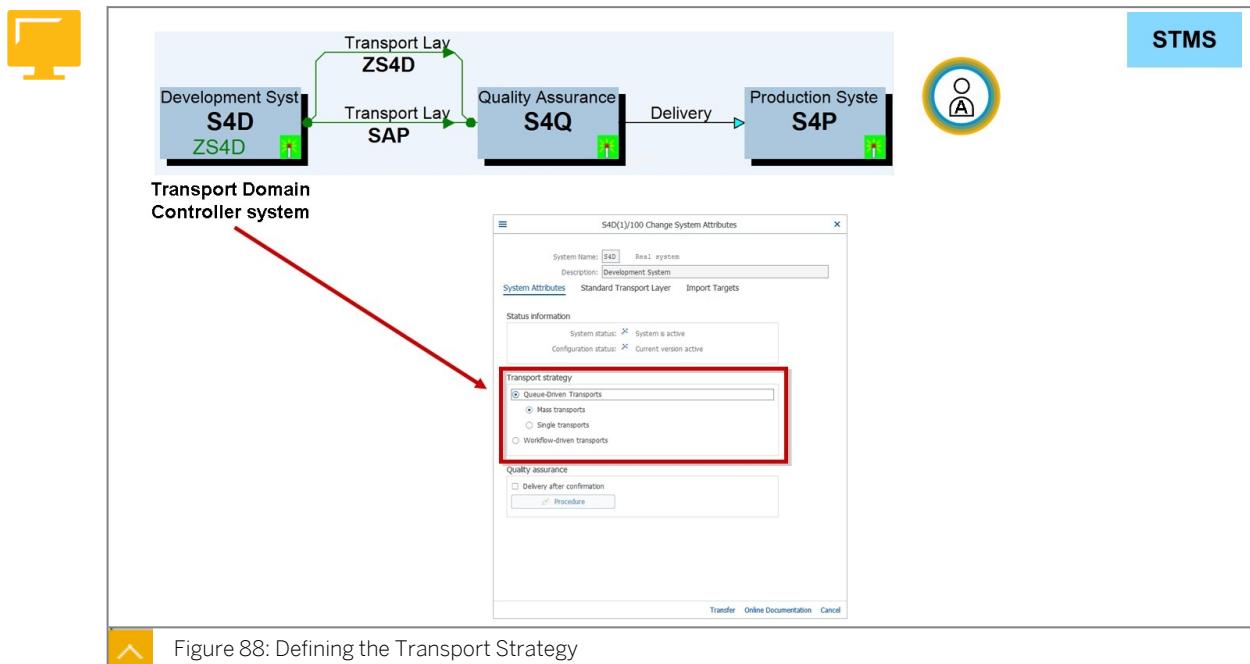


Figure 88: Defining the Transport Strategy

### Procedure

1. Start transaction STMS on the Transport Domain Controller system and choose Overview → *Transport Routes* from the menu. The screen *Display Transport Routes* appears displaying the existing transport routes in the transport domain.
2. Switch to the change mode.
3. Double-click one of the SAP systems of the system landscape. The *Change System Attributes* dialog box appears.
4. Select the *System Attributes* tab and choose your transport strategy.
5. Choose *Transfer*.
6. Save your settings and confirm to activate and distribute the configuration across all systems (if necessary).

### Settings in TMS Depending on the Transport Strategy

Some settings in the *Transport Management System* depend on the import strategy you have chosen:

#### Queue-Driven Transports, Mass Transports

By default, the import option *Leave Transport Request in Queue for Later Import* is activated, when doing an *import single*.



## Hint:

The import option *Leave Transport Request in Queue for Later Import* causes transport requests that have been imported as single transport requests to be imported in the correct order in the next import of all transport requests. This option is useful if you have to make preliminary imports for individual transport requests as it prevents older objects from being imported at the next regular import of all the transport requests (overtaker problem).

### Queue-Driven Transports, Single Transports

By default, the import option *Leave Transport Request in Queue for Later Import* is deactivated.



## Hint:

If you supply an SAP system only with single transport requests (without using the project import), this option would not be useful since the transport requests would remain in the queue after the import with status *Request is ready for import again* and would have to be manually deleted from the import queue.

The buttons in the application toolbar on the *Import Queue* screen change according to the requirements of the single import strategy.

In the *Import All Requests* function (“fully loaded truck”) is only available if you have selected one or more CTS projects using the *Filter* function. This prevents you from accidentally importing **all** transport requests in the queue.

### Workflow-Driven Transports

Transport proposals are created automatically when transport requests are exported.

The import options correspond to those for single transport requests.

Imports become linked to the processing of transport proposals in the *TMS Worklist*.

A warning appears in the import queue if you try to import transport requests without using the transport workflow.

The following parameters for the transport control program *tp* and the Change and Transport System (CTS) are set according to the chosen transport strategy:



Table 1: tp Parameters of Transport Strategies

Parameter	Value for Queue-Driven Mass Transports	Value for Queue-Driven Single Transports	Value for Workflow-Driven Transports
IMPORT_SINGLED_ONLY	0	1	1
NO_IMPORT_ALL	0	1	1
IMPORT_SINGLED_STRATEGY	0	1	0

Parameter	Value for Queue-Driven Mass Transports	Value for Queue-Driven Single Transports	Value for Workflow-Driven Transports
WORKFLOW_STRATEGY	0	0	1
REPEATONERROR	9	8	8
STOPONERROR	9	9	9

The *tp* parameter **STOPONERROR** defines from which return code on the import immediately stops. **REPEATONERROR** defines from which return code on the import is not classified as successful and has to be repeated (therefore, the transport request remains in the import queue). For example, with *Single Transports*, return code 8 is classified as an unsuccessful import and has to be repeated. With *Mass Transports*, the same return code 8 would be a successful import and the transport request is deleted from the import queue.



#### Hint:

Don't manually change the parameters that are relevant to the transport strategy. TMS generates these parameters each time the transport route configuration is changed.

## Transport of Copies and Relocation

Figure 89: Transport of Copies and Relocation

You can use *transport of copies* to transport objects to another SAP system of your choice, especially, if there is no consolidation route pointing from the SAP system you created the transport request to the SAP system you want to import the transport request. The objects are transported in the version they have in the SAP system, the transport request was exported from. The original location of the objects is not changed. In contrast to workbench requests or customizing requests, transports of copies will not be added to the import queue of a subsequent delivery system.



Note:

The term *copies* in *Transport of copies* is not related to the word “copying” but to the fixed term “copy” as opposite of “original”.

You can use *Relocations of objects w/o package change* if development work on objects is to take place in another SAP system **temporarily**. Special developments may be carried out in a separate SAP system, for example, so as not to interfere with the development process. This transport request type allows you to move the original location of objects to the target system. It also can be used, if an object was created, for example, in the quality assurance system, and the original system entry should now be moved to the development system.

Use *Relocations of objects with package change* when the development system of individual objects is to be changed on a **permanent** basis. This request type allows you to change the original location of objects to the target system and to change the package assignment of the objects at the same time. Due to the package being changed automatically, the objects have the desired transport attributes immediately after being imported into the target system of the transport request.

Use *Move Complete Package* when the development system of a complete package is to be changed on a permanent basis.



### LESSON SUMMARY

You should now be able to:

- Compare the different queue-based options to import transport requests
- Explain how to time imports and to define maintenance periods
- Describe the different transport strategies
- Outline the use of Transport of Copies and Relocations

## QA Approval Procedure and Transport Proposals

### LESSON OVERVIEW

In this lesson, you will learn the use of the *Quality Assurance (QA) approval* procedure. You will also learn the use of the special transport workflow and its configuration procedure. In addition, you will see why creating transport proposals can be helpful.

### Business Example

Transport requests are to be imported into the production system only after they have been approved in the quality assurance system. Therefore, you need a defined *QA approval procedure*.

In some situations, however, it may be necessary to transport an urgent correction into the production system directly, bypassing the configured transport routes. You can use the special transport workflow to perform this type of transport.



### LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Use the QA approval procedure
- Explain the idea of the transport workflow

### TMS Quality Assurance

The *TMS Quality Assurance (QA) approval procedure* increases the quality and the availability of the production systems by letting you check transport requests in the quality assurance system before importing them to subsequent SAP systems.

When you activate the *QA approval procedure*, transport requests are only ready to be imported into the delivery systems if all the QA approval steps are processed for the transport request in the QA system and the transport request has been approved. When you configure the *QA approval procedure*, you determine how many QA approval steps have to be processed for each transport request. If one check for an approval step is not successful, the entire transport request cannot be approved. Therefore, you can only import completely approved transport requests into the delivery systems.

Rejected transport requests are not imported into the delivery systems of the QA system.

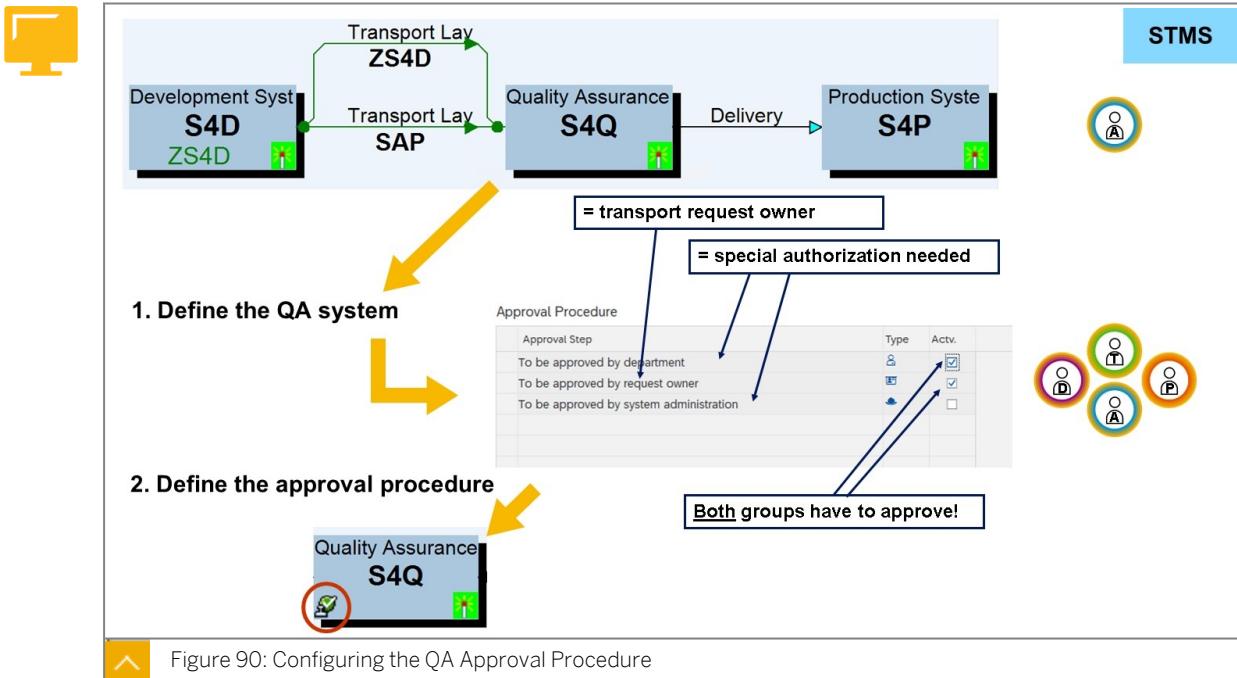


Figure 90: Configuring the QA Approval Procedure

Before you can process transport requests, you should configure the *QA approval procedure*. For this, ensure that the SAP system landscape and/or transport domain is set up so that there is at least one development system, one quality assurance system, and one production system. The SAP system to be configured as the QA system must have the following attributes:

- It must be the target of at least one transport route (either a consolidation route or a delivery route).
- It must be the source of at least one delivery route.

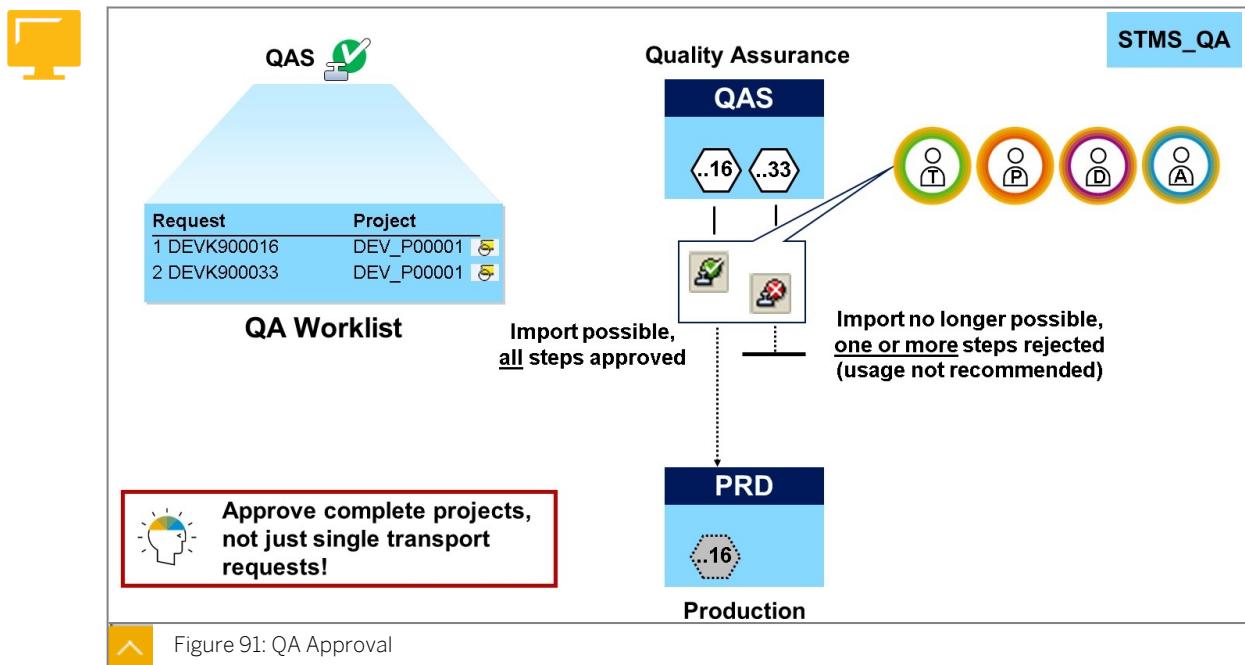
In the SAP system attributes for the chosen SAP system, the *Delivery after confirmation* option must be set. In the approval procedure, you can define which users must approve so that the transport can be imported into the delivery system (see the figure “Configuring the QA Approval Procedure”).

After configuration, the *QA worklist* is automatically set up. All the transport requests imported into the QA system are included in the *QA worklist*.

### Steps in the QA Approval Procedure

To display the *QA worklist*, use transaction STMS\_QA. The timestamp at the upper-right side of the screen indicates when the *QA worklist* was last updated. The entry in the upper-left side indicates how many transport requests still need to be processed.

The list displays the transport requests corresponding to the selected approval steps. By default, the transport requests corresponding to **all** approval steps are shown. To select the approval step whose corresponding transport requests you want to see, choose *Worklist → Select Approval Step* from the menu. By double-clicking various items in the table listing the transport requests, you obtain further information on those items.

**Hint:**

You could also access the QA *worklist* using transaction STMS, but, from a security point of view, you might want to allow only administrators to call transaction STMS. Check out the sub-transactions of STMS by searching for entries in table TSTC (using transaction SE16) beginning with **STMS\***. Here, you can find, for example, **STMS\_QA**.

You should test the transport requests listed in a QA *worklist* before importing them into the delivery systems.

The QA status *Rejected* means that one or more approval steps of a transport request have been rejected by a person responsible. A request is approved only if all the approval steps have the status *Approved*.

In the QA *worklist*, you can see:

- The QA status (*St*)
- The overall status (*GS*)
- The number of steps (*Nr*)

Transport requests can be imported into the delivery systems only if all of them have received approval according to the various approval steps.

**Note:**

If all the transport requests for one project have been approved, they can be imported into the delivery system even if other projects still have unprocessed or rejected requests in the QA *worklist*.

Transport requests with the QA status *Rejected* as well as unprocessed transport requests in the QA *worklist* will not be imported into the delivery systems.

**Hint:**

The approval or rejection of a single step can be changed, as long as not all approval steps have been performed. As soon as all defined approval steps have been performed, however, it is no longer possible to change the decision.

**Note:**

SAP recommends that you do **not** reject transport requests containing errors, but correct the error using subsequent transport requests and then approve the affected transport requests as an entire package.



**QA Worklist: System S4Q**

Object List Object List - Web UI **QA History** More ▾

13.09.2021 11:25:53

Number	Request	Mdt	Owner	Project	Short Text	St	GS	Nr
1	S4DK900130	100	BEYER-MENZEL	S4D_P00002	Development Rebate Processing	⚡	⚡	1
2	S4DK900134	100	CATON	S4D_P00001	Development Contracts	⚡	⚡	—
3	S4DK900106	100	BEYER-MENZEL	S4D_P00001	Development Sales	⚡	⚡	—
4	S4DK900143	100	CATON	S4D_P00001	Modification Sales Document	⚡	⚡	1
5	S4DK900102	100	CATON	S4D_P00001	Customizing Enterprise Structure	⚡	⚡	—
6	S4DK900132	100	CATON	S4D_P00002	Change to Development Rebate Process	⚡	⚡	—

**QA history**

**Who approved quality assurance testing for this transport request?**

Type	Description
⚡	Distribution required
∅	New
⌚	In Process
🔴	Request rejected
🟢	Request approved

Figure 92: QA History

The meaning of the different icons for the approval status are as follows:

- *Distribution required*

The QA status for this transport request could not be distributed. After correcting errors, you must repeat the distribution.

- *New*

The transport request is new in the QA worklist.

- *In Process*

There are still some approval steps for this transport request that have to be processed.

- *Request rejected*

The transport request was rejected for an individual step or for all steps (depending on the view displayed in the QA worklist).

- *Request approved*

The transport request was approved for an individual step or for all steps (depending on the view displayed in the QA worklist).

From the QA work list screen, you can access the QA history by choosing the menu path *Goto → QA History*.

The QA worklist history (or QA history) shows all the transport requests for a specific period that are no longer displayed in the QA worklist. Transport requests are no longer displayed in the QA worklist when they have been approved or deleted. The default period for the QA history is 30 days but this time period can be changed in the *Confirm time interval* dialog box.

To determine who was responsible for approving a transport request, choose *Request → Display → QA Status*.



#### Note:

The QA history is stored in the database of the QA system (the QAS system). If you perform a database or SAP system copy from PRD to QAS, the history will be lost. To prevent the QA history from being deleted in this way, see SAP Notes [397138 – TMS QA and system copies](#) and [1864549 – STMS QA saving administration data more easily](#).

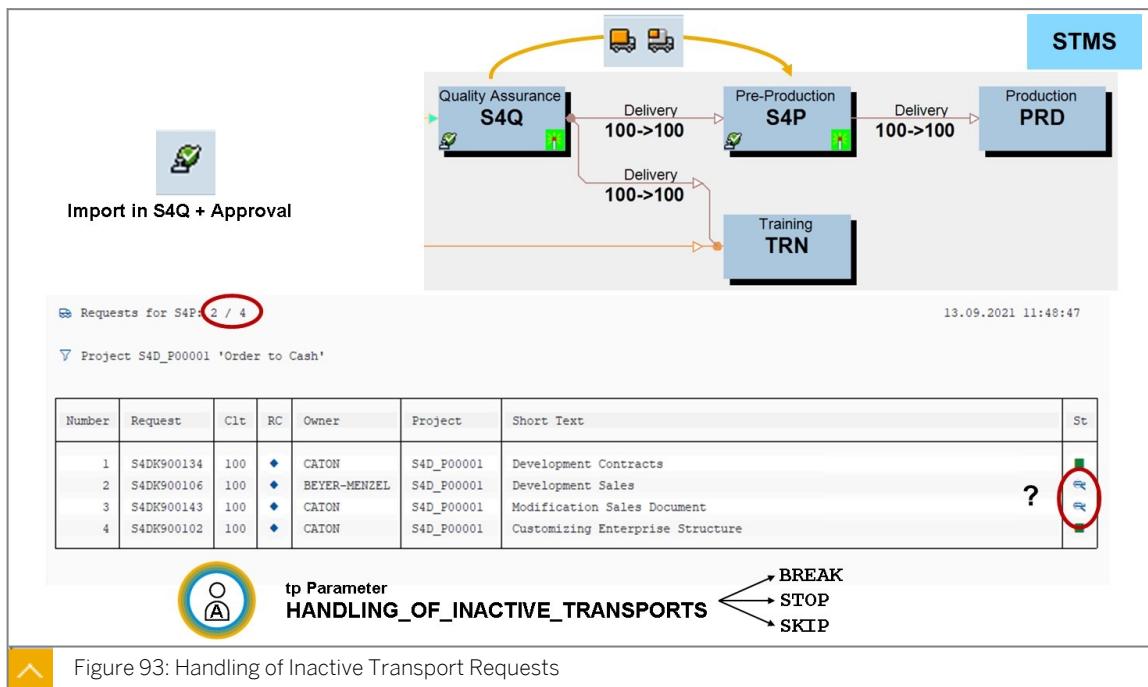


Figure 93: Handling of Inactive Transport Requests

The setting of the *tp* parameter **HANDLING\_OF\_INACTIVE\_TRANSPORTS** controls how a mass import (or an import of all transport requests belonging to the same CTS project) into a subsequent SAP system should react if inactive transport requests are in the import queue. For this parameter, the following values are possible:

#### BREAK

The import terminates if the import queue contains at least one inactive transport request. This is the default value for this parameter.

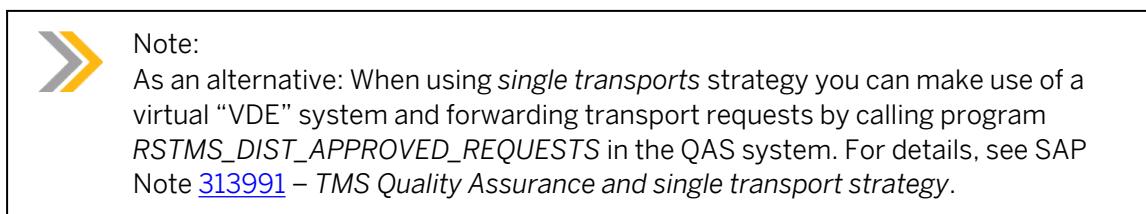
#### STOP

During the import, the system imports all of the confirmed requests that are in the queue before the first unconfirmed request.

### SKIP

During the import the system imports all of the confirmed requests and skips the unconfirmed requests (and the rejected requests). When you use this setting, sequence problems with regard to the involved transport requests may occur. Therefore, this setting is not recommended.

For details, see SAP Note [1372301 – Handling unconfirmed transport requests](#).



## Special Transport Workflow

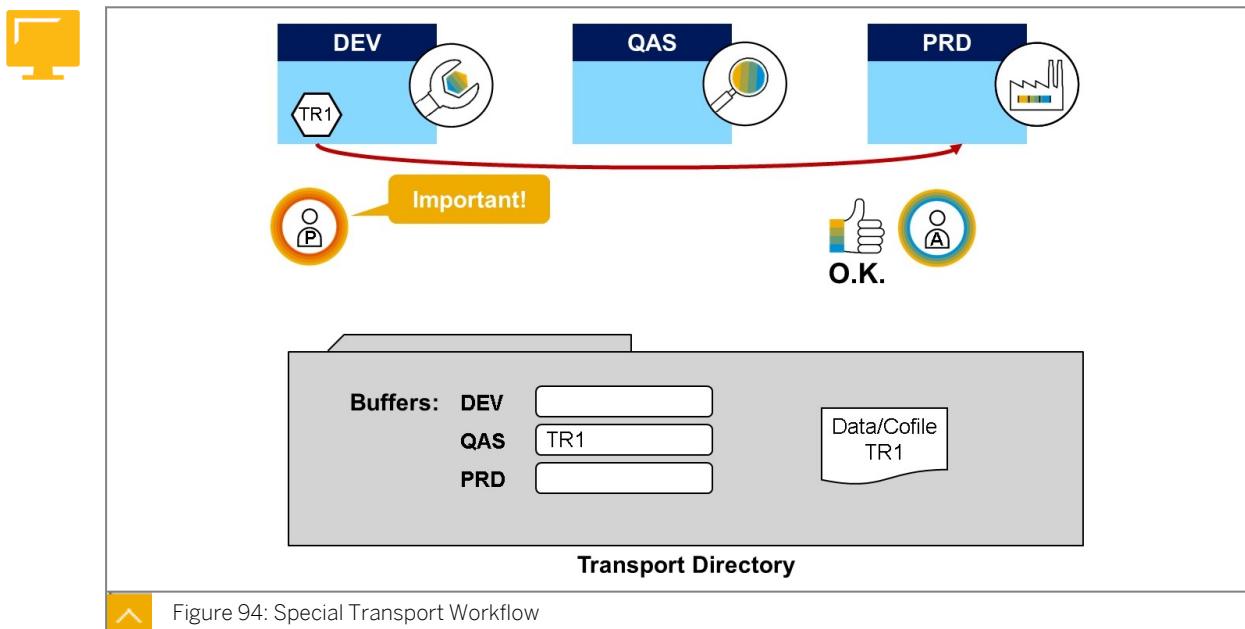


Figure 94: Special Transport Workflow

When you use a *queue-driven transport* strategy, you can use the **special transport workflow** if transport requests containing corrections are needed urgently, or if transports are required that do not follow the defined transport routes.

Before you can use the special transport workflow, you must configure one client in one SAP system as the so-called *Workflow Engine*.

The system on which the *Workflow Engine* is to be configured should have the following criteria, which are listed in order of importance:

1. High availability
2. High release

### 3. Low or medium system load

 Note:  
These prerequisites are usually met by the productive system.

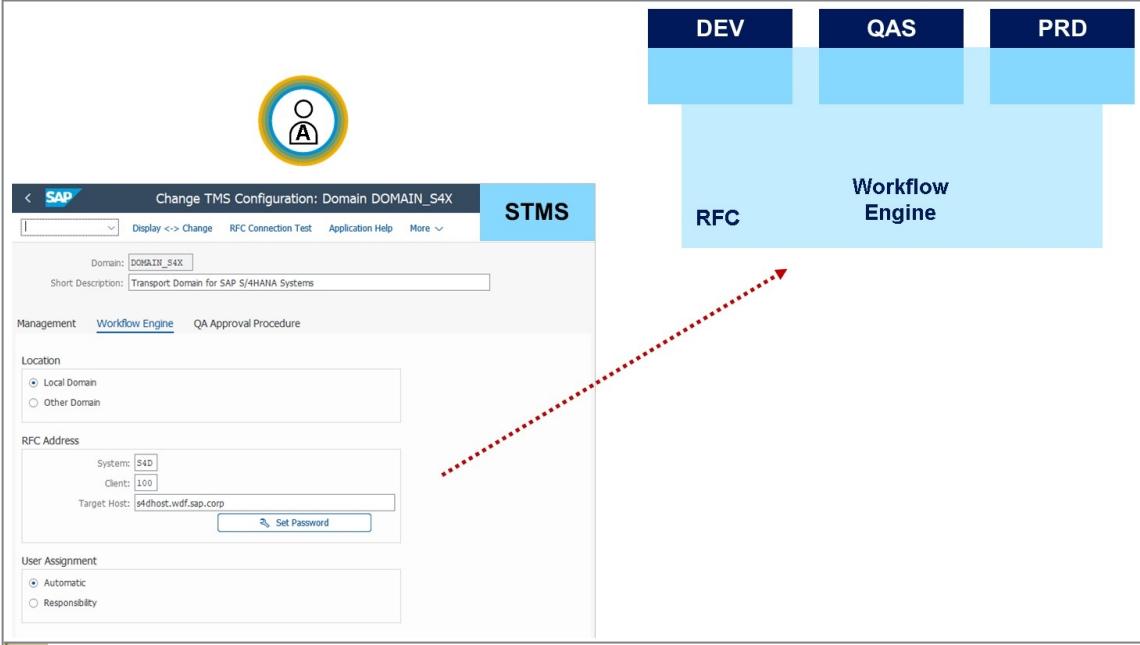


Figure 95: Configuring the Special Transport Workflow

To configure the special transport workflow, do the following:

1. Log on to the SAP system acting as the transport domain controller.
2. Start transaction STMS, choose the menu path *Overview → Systems* and then the menu path *Goto → Transport Domain*.
3. Choose the *Workflow Engine* tab.
4. Switch to change mode. Enter the SAP system, the client, and the target host for your *Workflow Engine*. Choose Save. Confirm the dialog box to distribute the configuration change.
5. On the *Display TMS Configuration: Domain <domain>* screen, choose *Set (TMSADM\_WF) Password* (if it exists) and enter a password for user *TMSADM\_WF* (you may need to choose *Execute* and then confirm a dialog box). This creates the system user *TMSADM\_WF* in this client.



#### Hint:

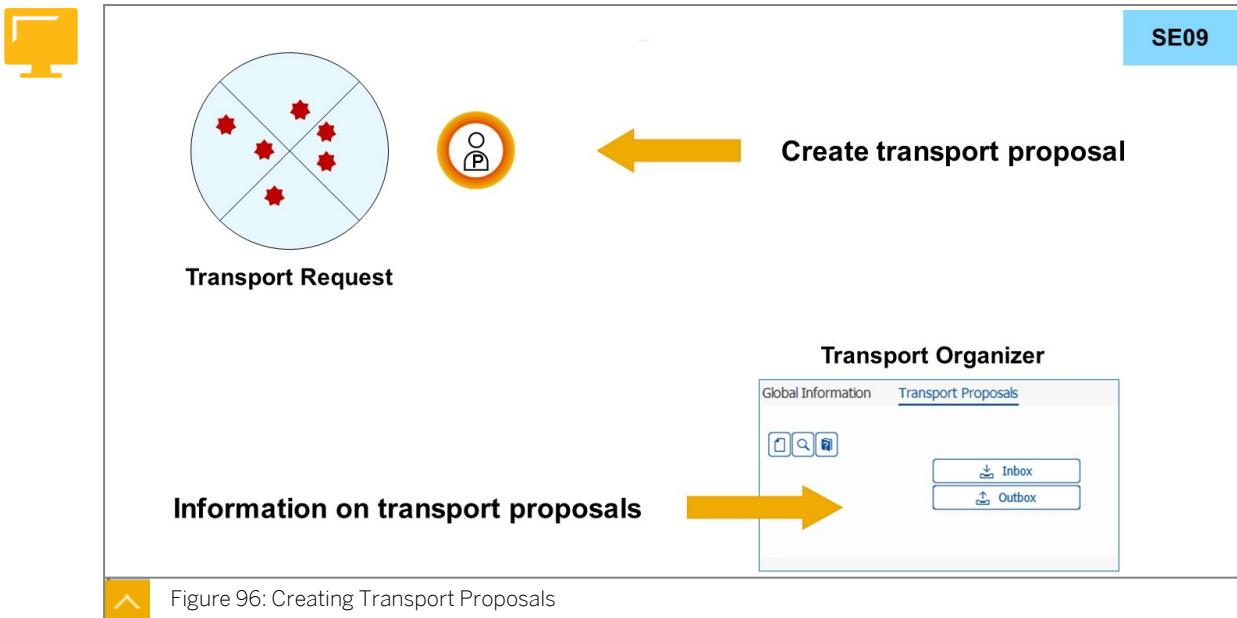
In the case of an error message that this user cannot be created, follow SAP Note [2191190](#) – *Could not create user TMSADM\_WF error configuring workflow*.

6. You may need to log on to every system in the system landscape. After finishing you may get the message that (among others) the RFC destinations *TMSADM\_WF* required for the

*Workflow Engine* in the connected systems have been updated. In addition, the SAP system automatically:

- Sends the address data of the *Workflow Engine* to all SAP systems in the transport domain.
- Performs the workflow related customizing on the *Workflow Engine*.

7. Go Back to the *Change TMS Configuration: Domain <domain>* screen and choose *Check Workflow Customizing* from the application toolbar.



To use the special transport workflow, you must create a *transport proposal*. To do this, go to the *Transport Organizer* (Transaction SE09) and select *released* requests. Choose *Display*. Position the cursor on a transport request that you want to transport and choose *Utilities* → *Create Transport Proposal* from the menu. The *Create Transport Proposal* dialog box appears. Enter a *Short text*, the *Target system*, and (optional) other transport requests. The target system applies to all transport requests that you enter. Then choose *Create and Send Proposal*.



#### Note:

When a transport proposal is created in the Transport Workflow, you can specify all the systems of the transport landscape as target system. If only the target systems defined directly in the transport routes should be available to the person placing the proposal, see SAP Note [374043 – Use only direct transport rows in Transport Workflow](#).

Any transport requests that you place in the transport proposal must be released.

When a transport proposal is created, the SAP system assigns a proposal number to this transport proposal, and then places it in the specific *TMS worklist* for the transport administrator.

If the transport administrator rejects your transport proposal, it reappears in your transport proposal inbox. You can cancel the transport proposal or revise it and send it back to the transport administrator.

After the transport administrator has approved your transport proposal, the import of the corresponding transport requests will be started and the transport proposal will reappear in your transport proposal inbox. Check that the transport requests are being imported correctly into the target system or systems, and confirm the transport proposal.



To approve or reject a transport proposal, start transaction STMS in the system / client specified as *Workflow Engine*. To display the *TMS worklist* (see the figure “Transport Proposal Worklist”), choose the menu path *Overview → Worklist*. Double-click the transport proposal you want to process. Check if the transport requests, the list of target systems, and the import times and import options for the transport proposal are correct. You can display the object list of the transport requests by choosing *Display Requests*. Switch to change mode if you want to make changes to the transport requests, transport targets, import times, or import options.

You can create a message for a developer by choosing the *Create Attachment* icon.

To process the transport proposal, choose the respective icon to approve or reject the transport proposal.

- If you approve the transport proposal, the import into the specified SAP systems will start automatically. The transport proposal is placed in the developer's transport proposal inbox and the developer finally confirms it.
- If the transport proposal is rejected, it will be placed in the developer's transport proposal inbox for modification.

### More Information

For more information on the transport workflow, see the online documentation for SAP S/4HANA (*Product Assistance*), area *Enterprise Technology → ABAP Platform → Administrating the ABAP Platform → Administration Concepts and Tools → Solution Life Cycle Management → Software Logistics → Change and Transport System → Transport Management System (BC-CTS-TMS) → Transport Workflow (Administration)* and area *Enterprise Technology → ABAP Platform → Administrating the ABAP Platform → Administration Concepts and Tools → Solution Life Cycle Management → Software*

*Logistics → Change and Transport System → Transport Management System (BC-CTS-TMS) → Configuring TMS → Configuring the Transport Workflow.*



### LESSON SUMMARY

You should now be able to:

- Use the QA approval procedure
- Explain the idea of the transport workflow

# Unit 4

## Lesson 4

# Import Process

## LESSON OVERVIEW

In this lesson, you will learn about the technical steps involved in the import process.

### Business Example

To be able to analyze and resolve problems that may arise during imports, you must understand the steps in the import process.



### LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Analyze the different steps during the import of transport requests
- Explain the use of tp commands

## Steps in the Import Process



### When importing several transport requests simultaneously

- tp collectively processes each import step for all transport requests before proceeding with the next import step
- tp does **NOT** process all import steps for only a single transport request before proceeding with the next transport request

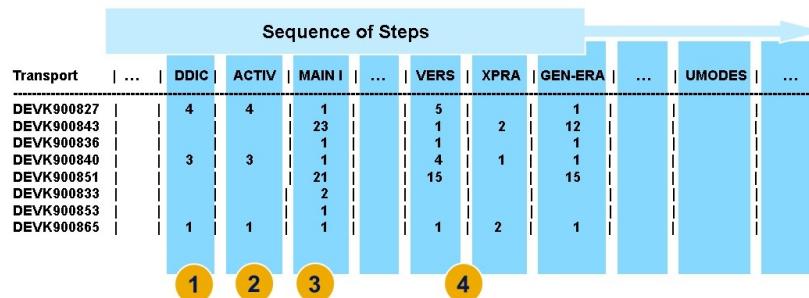
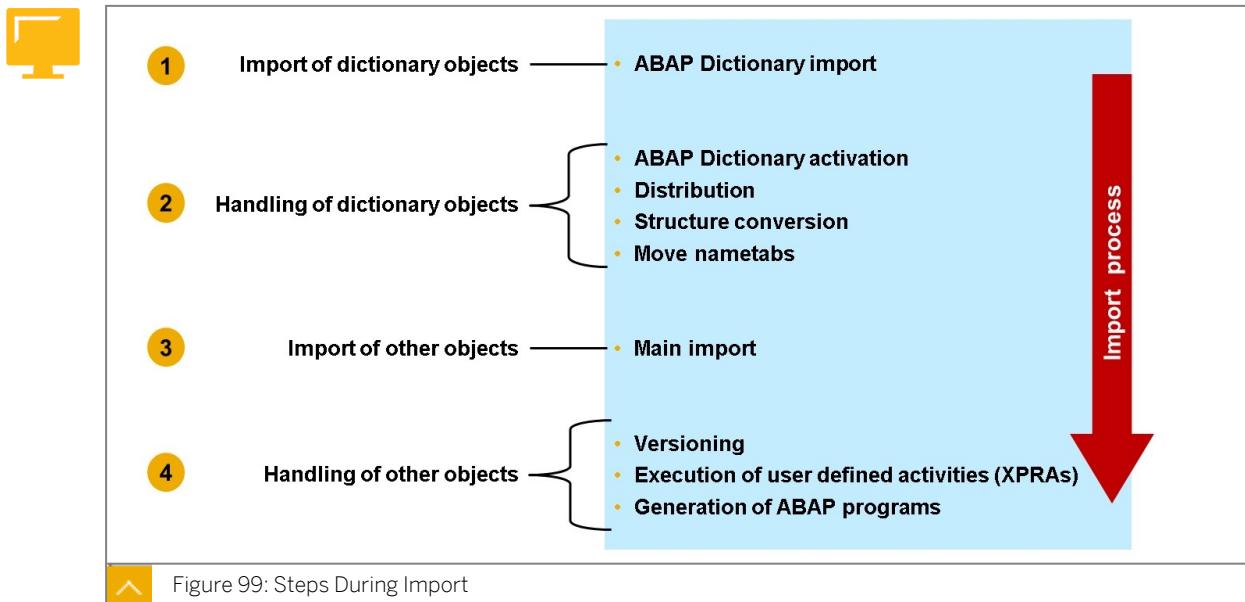


Figure 98: tp Processing Sequence

When importing several transport requests simultaneously, *tp* processes each import step collectively for all transport requests to be imported. The required steps for each request are listed in the transport buffer file. The contents of the transport buffer file are organized as a table in which each column represents an import step. The numbers in the columns indicate either whether the import step is necessary, or the number of objects in the request that require the specific step.

When you import all transport requests listed in the figure “*tp* Processing Sequence”, together, the first import step *DDIC* is processed for transport requests where it is required (DEVK900827, DEVK900840 and DEVK900865), then the *ACTIV* step is processed (for the same transport requests), then the *MAIN I* step for all transport requests, and so on.

The `tp import all` command, therefore, does **not** process all steps for one transport request before moving on to the steps for the next transport request. For example, if you detect an error in a program that has already been exported, you will need to correct the program and release the corresponding transport request for the correction. *Import all* requests then will import the transport requests of the whole transport buffer in the correct sequence and the faulty program will be overwritten. Because the generation step is the last step and is performed only once for all programs in the transport requests, the faulty program does not affect your production system. Only the correct version of the program will be generated.



There are four major steps when importing transport requests (see the figure “Steps During Import”):

1. Importing dictionary objects
2. Handling of dictionary objects
3. Importing “everything else”
4. Handling of “everything else”

These steps in the import process are, in detail:

#### DDIC: ABAP Dictionary import

In an import with *R3trans*, all ABAP Dictionary structural data is imported inactively, thus enabling you to perform this import step in an SAP system that is not yet in downtime.

- Log file: <source SID>H9<number>. <target SID>
- Transport tool: *R3trans*

#### ACTIV: ABAP Dictionary activation

Runtime descriptions (*nametabs*) are generated. This is the beginning of the downtime.

- Log file: <source SID>A9<number>. <target SID>
- Transport tool: Job *RDDMASGL* (starting report *RDDMASGL*)

#### ACTIV: ABAP Dictionary distribution

After activation and running logical checks for the new dictionary structures, the distribution program decides what actions are required to adopt the object on database level.

- Log file: *DS<date>.<target SID>*
- Transport tool: Job *RDDISOL* (starting report *RDDGENBB* with variant *DIST*)

#### **ACTIV: Structure conversion**

If necessary, changes are made to table structures.

- Log file: *N<date>.<target SID>*
- Transport tool: Job *RDDGENOL* (starting report *RDDGENBB* with variant *CONV*)

#### **ACTIV: Move nametabs**

The new ABAP runtime objects are put into the active runtime environment, database structures are adjusted, if necessary.

- Log file: *P<date>.<target SID>*
- Transport tool: Job *RDDMNTAB* (starting report *RDDMNTAB*)

#### **MAIN I: Main import**

All other objects and data are imported.

- Log file: *<source SID>I9<number>.<target SID>*
- Transport tool: Job *R3trans*

#### **VERS: Versioning**

Versions of repository objects are only created if *tp* parameter *VERS\_AT\_IMP* is active.

- Log file: *<source SID>V9<number>.<target SID>*
- Transport tool: Job *RDDVERS\** (starting reports *RDDVERS\**)

#### **XPRA: Execution of Reports after import and of "After Import Methods"**

XPRAs are programs that are started during import in the target system. The XPRA object has the same name as the program.

- Log file: *<source SID>R9<number>.<target SID>*
- Transport tool: Job *RDDEXECL* (starting report *RDDEXECL*)

#### **GENERA: Generation of ABAP programs**

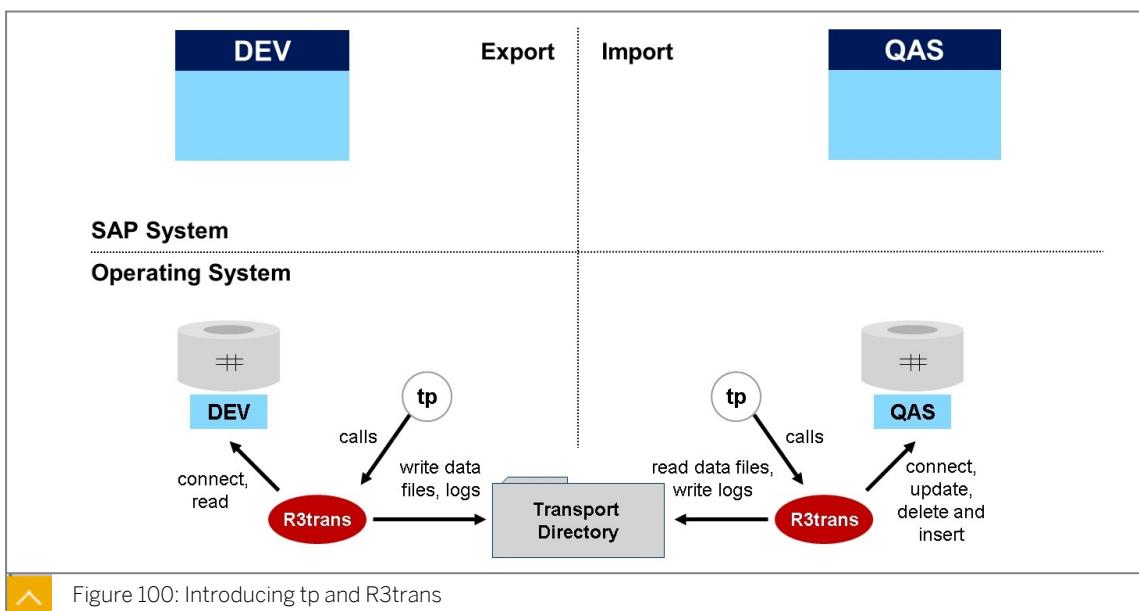
During this step, the ABAP programs are generated.

- Log file: *<source SID>G9<number>.<target SID>*
- Transport tool: Job *RDDDIC3L* (starting report *RDDDIC3L*)

**Note:**

It is also possible to import non-ABAP objects using the enhanced Change and Transport System (*enhanced CTS*, formerly known as *CTS+*). In this case, an additional deployment step for importing the objects into the non-ABAP system takes place after the main import. The corresponding log file is <source SID>T9<number>.<target SID>, the transport tool is job *RDDEPLOY*.

When transporting SAP HANA content with the help of *SAP HANA Transport for ABAP* (HTA), the additional job *RDDHANAD* for SAP HANA deployment is triggered. This job creates the log file <source SID>C9<number>.<target SID>.



*R3trans* is a transport tool at operating system level used to transport data between SAP systems. *R3trans* is usually called by other programs such as *tp*, the transport control program.

For transports between SAP systems, to access the database, *tp* indirectly calls *R3trans* by causing UNIX to issue a `fork()`, Windows to issue a `CreateProcess()`, and AS/400 to issue a `spawn()`. During export, *R3trans* stores the object data extracted from the database in data files in the transport subdirectory *data*. The format of these data files, *R3trans* format, is independent of the platform. During import, *R3trans* reuses these data files.

Direct use of *R3trans* is not supported but may be required in exceptional cases. In case of transports, *R3trans* should always be used via *tp*.

**Note:**

Import steps differ for the different object types. Further activities may be required in addition to *R3trans* activities. *tp* ensures that all export and import steps, including *R3trans* activities, are completed successfully.

*R3trans* writes data using a standard transport format. Thus you can export data with an old *R3trans* version and import data with a new version of *R3trans*. You can also transport between different databases or operating systems.

**Note:**

Although exports and imports are independent of the *R3trans* version, the database platform, or the operating system, SAP does not support using *tp* or *R3trans* for transports between different SAP system releases. See also SAP Note [1090842 – Composite SAP Note: Cross-release transports.](#)

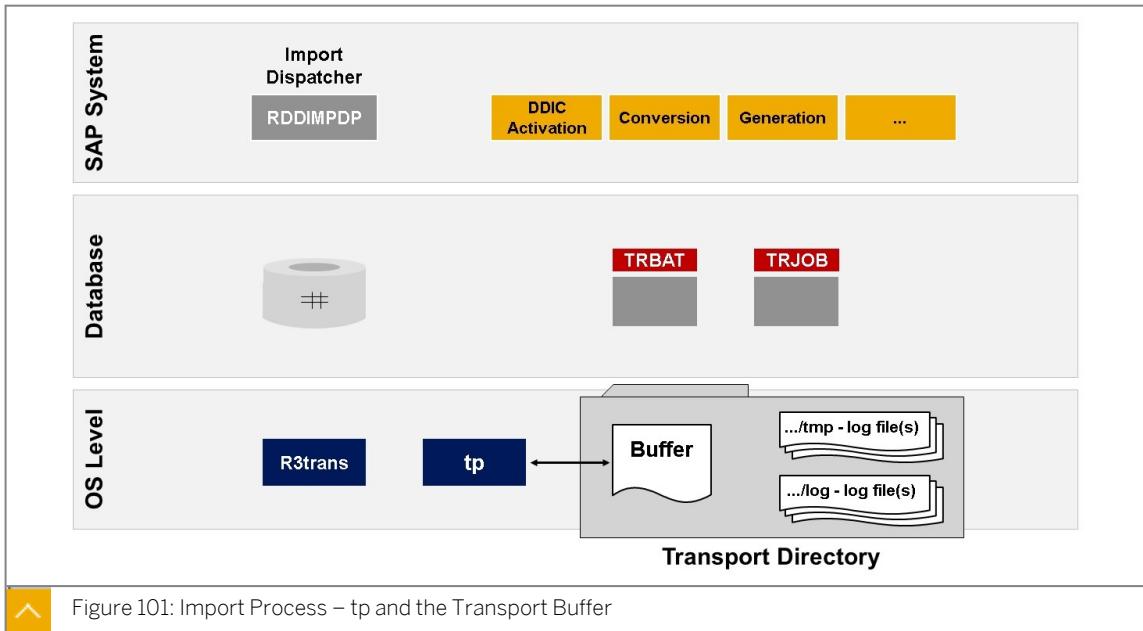


Figure 101: Import Process – *tp* and the Transport Buffer

The first step of an import process is the *tp* call, triggered by starting an import through TMS or by a *tp import* command at operating system level.

During the whole import process, *tp* reads the transport buffer file that includes all the necessary steps for the specific request.

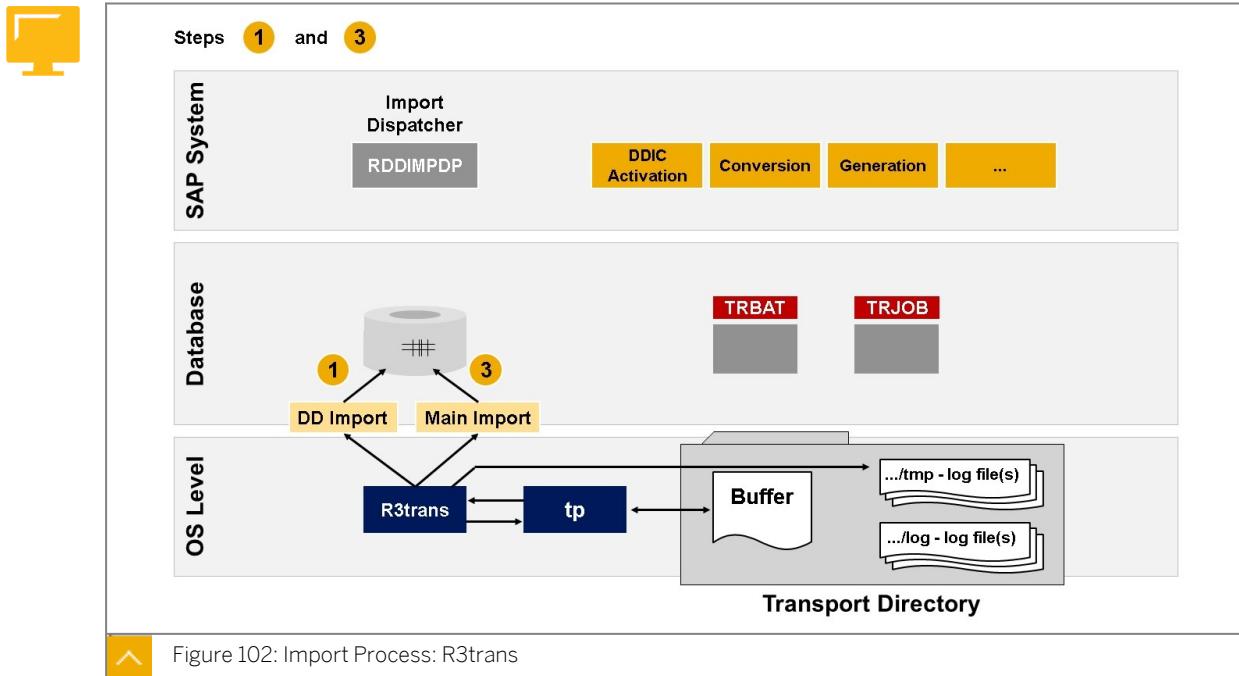
If you trigger a *tp import all*, *tp* has to ensure that only all transport requests are imported that are stored in the transport buffer **at the beginning** of the import process. This is done by executing command *tp setstopmark* each time a *tp import all* process is started. After the steps of the import process are completed, the command *tp delstopmark* is performed automatically, and a *tp cleanbuffer* deletes the transport requests from the transport buffer.

After all involved tools have finished their work, *tp* exits to operating system level and writes a return code to the appropriate log file for the activity. For example, *tp import* commands are recorded in the *ULOG* file.

**Note:**

The command *tp import* can be restarted. If an error occurs during import, after you eliminate the error condition and restart *tp*, *tp* finds the correct point to restart.

By default, *tp* aborts if one import phase receives a return code larger than 8. The transport profile parameter *STOPONERROR* defines what return code value should cause *tp* to abort.



*tp* reads the transport buffer file that includes all the necessary steps for the specific transport request(s) and calls *R3trans* at operating system level.

For each import step, *tp* passes information from the transport buffer file to *R3trans*. *R3trans* reads the corresponding data files in the transport subdirectory *data* and connects directly to the database to perform inserts or updates to the included objects. After *R3trans* finishes performing inserts or updates, it passes an exit code (*return code*) to *tp*.

For each transport action, *R3trans* writes a log file in transport subdirectory *tmp*. After *R3trans* has completed its work, *tp* moves these log files to the transport subdirectory *log*.

During the import process, *R3trans* is executed by the import steps *ABAP dictionary import* (for the import of ABAP dictionary definitions) and *main import* (for the import of other repository objects and table contents).

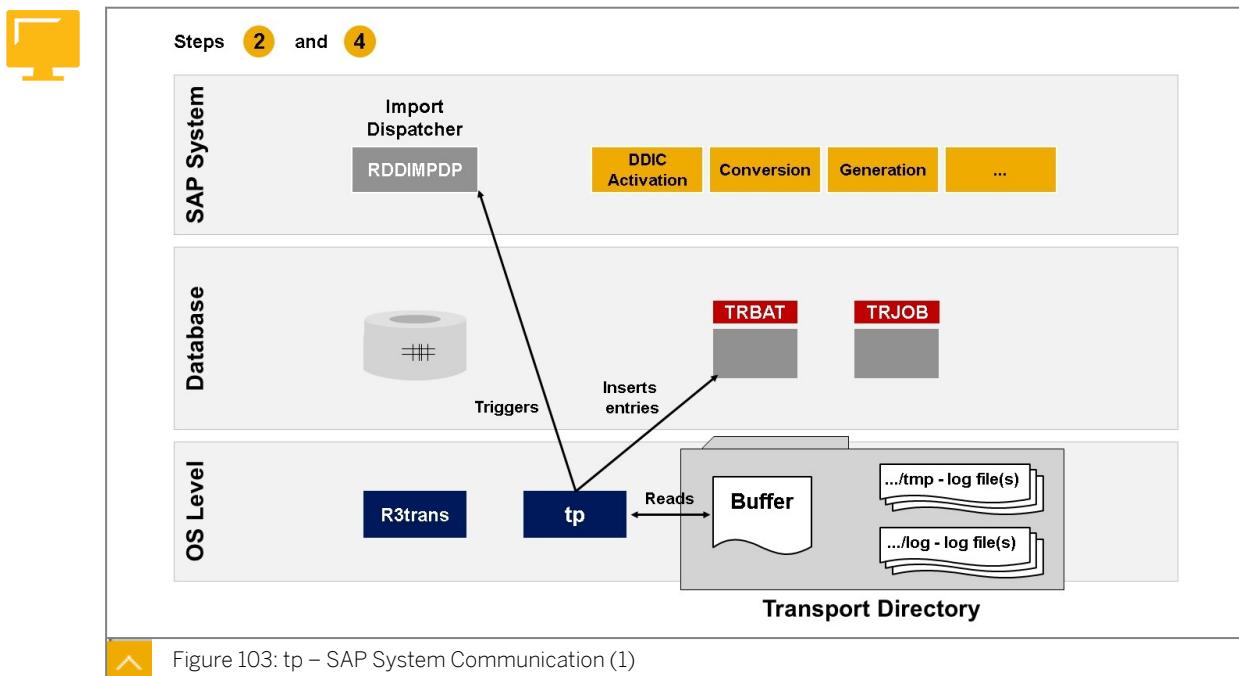


Figure 103: tp – SAP System Communication (1)

In addition to the steps performed by *R3trans*, *tp* triggers the so-called import dispatcher. The **import dispatcher** is a background job with the name *RDDIMPDP* in the SAP system, which performs steps in the import process.

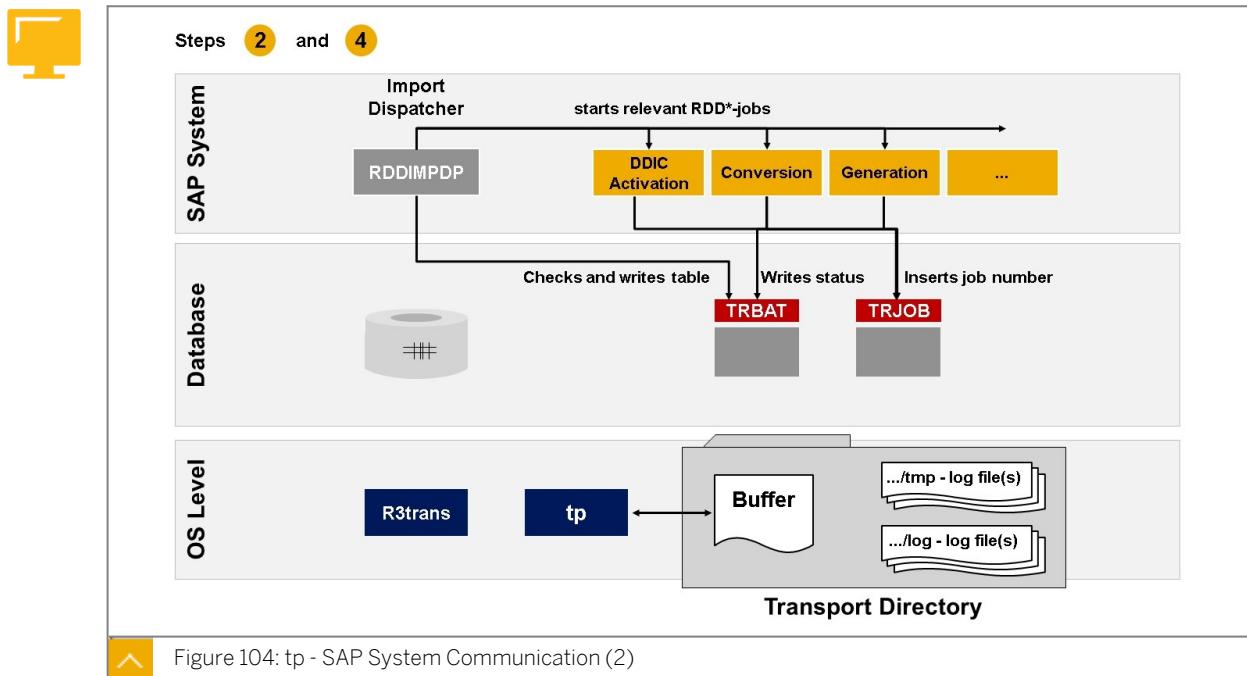
*tp* and *RDDIMPDP* communicate via table *TRBAT*. For every transport request, *tp* writes an entry in table *TRBAT*. The import function currently being performed for the request is represented by a character.

In the example below, there are three transport requests waiting for activation of DDIC objects (*Function = J*) in the table *TRBAT*.

Request	Function	Return Code	Timestamp
DEVK904711	J	9999	00000001
DEVK904714	J	9999	00000002
DEVK904718	J	9999	00000003
HEADER	J	B	20221112181000

*tp* inserts a header entry to request *RDDIMPDP* to start processing. Some activities that are independent of transport requests, such as *distribution* and *structure conversion*, only have a header entry in table *TRBAT*. Return code 9999 indicates that the step is waiting to be performed. For the header entry, *tp* inserts a *B* (for “begin”) as return code.

To trigger the job *RDDIMPDP* in the SAP System, *tp* uses the operating system level tool *sapevt*.



When *RDDIMPDP* is started, it checks the table *TRBAT* to find out if there is an action to be performed for the transport requests such as *mass activation*, *distribution*, or *table conversion*. It sets the *header* entry to *R* (for run), and starts the appropriate *RDD\** program as a background task, reschedules itself, and exits.



Note:  
For example, *RDDMASGL* is for mass activation, *RDDEXECL* for execution of reports and “after import methods” after import, and *RDDVERS1* for versioning. Each *RDD\** job receives a job number that is recorded in table *TRJOB*. The jobs report their current status back to *TRJOB*.

The activated program (in this example, the mass activator for dictionary objects) sets the status of the first entry in *TRBAT* to active (return code 8888):

Request	Function	Return Code	Timestamp
DEVK904711	J	8888	00000001
DEVK904714	J	9999	00000002
DEVK904718	J	9999	00000003
HEADER	J	R	20221112181005

Each of the required background tasks receives a job number generated by background processing. This job number and the step ID are inserted into table *TRJOB* by the *RDD\** jobs.

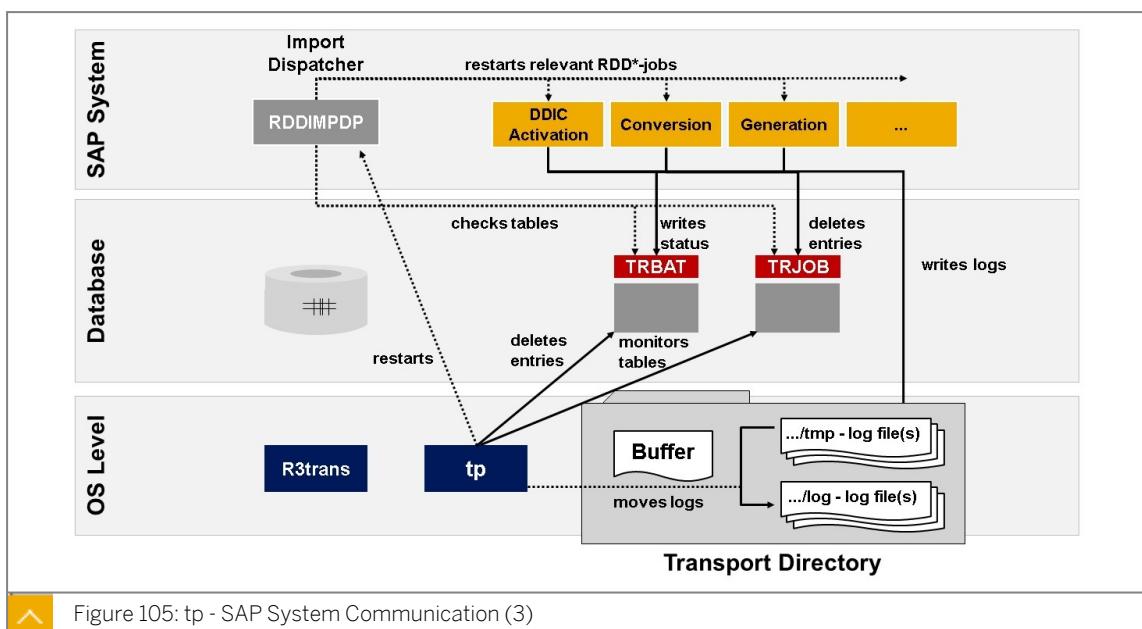


Figure 105: tp - SAP System Communication (3)

The background tasks write their return codes in table *TRBAT* and delete the corresponding entry in table *TRJOB*. Return codes of 12 or less indicate that the step is finished. In *TRBAT*, the column *TIMESTAMP* contains the finishing time. When all the necessary actions are performed for all transport requests, the header entry is set to F (for “finished”) by the *RDD\** jobs.

Request	Function	Return Code	Timestamp
DEVK904711	J	4	20221112181039
DEVK904714	J	0	20221112181041
DEVK904718	J	0	20221112181045

Request	Function	Return Code	Timestamp
HEADER	J	F	20221112181045

All the background jobs log the performed steps either in the database or in transport subdirectory *tmp*.

*tp* monitors the entries in table *TRBAT* and table *TRJOB*. When the header entry in table *TRBAT* is set to *F* and table *TRJOB* is empty, *tp* copies the logs of the completed steps from directory *tmp* to directory *log* and deletes the corresponding entries in table *TRBAT*.

If problems are detected by *tp* when monitoring the *TRBAT* and *TRJOB* tables, *tp* re-triggers *RDDIMPDP* through *sapevt*. *RDDIMPDP* then automatically recognizes if a previous step is still active or was aborted by checking the *TRJOB* and *TRBAT* tables. If a step was aborted, *RDDIMPDP* restarts this step.

*tp* uses the tables *TRBAT* and *TRJOB* to communicate with the various ABAP programs started by the background jobs. Note that there must be at least two background work processes configured in the SAP system.

When imports are performed, *tp* triggers the import dispatcher *RDDIMPDP* by sending the event *SAP\_TRIGGER\_RDDIMPDP* with the help of the tool *sapevt*. In client 000, user DDIC must schedule the job *RDDIMPDP* with event-based scheduling.

For AS ABAP based systems of SAP Business Suite, *RDDIMPDP* can be scheduled by running the ABAP program *RDDNEWPP* in client 000 with a user that has the CTS administration authorization *S\_CTS\_ADMIN*.



Hint:

Executing *RDDNEWPP* in clients different from 000 schedules job *RDDIMPDP\_CLIENT\_<####>*. Job *RDDIMPDP\_CLIENT\_<####>* (which has been scheduled automatically after a local or remote client copy with transaction *SCCL* or *SCC9*) is not needed anymore, however. See SAP Note [2687484](#) – About job *RDDIMPDP\_CLIENT\_nnn*.



Note:

SAP Note [3035580](#) – Job *RDDIMPDP* running as DDIC ships a new report (*RDDNEWPP2*) which enhances its predecessor (*RDDNEWPP*) with the option to explicitly specify the user chosen for the scheduled executions of the background job *RDDIMPDP*. In addition, see SAP Note [3217799](#) – Use a different user for the job *RDDIMPDP* in S/4HANA - SJOBREPO.

Due to the Job Repository and the profile parameter *rdisp/job\_repo\_activate\_time*, for SAP S/4HANA Server systems, it is not necessary to run report *RDDNEWPP* in client 000, but it might take some time until the job is scheduled again.

**Note:**

You can also import non-ABAP objects using the enhanced Change and Transport System (*enhanced CTS*, formerly known as *CTS+*). In this case, table *TRBATS* is used. *TRBATS* works like *TRBAT*, but *TRBATS* includes the SID of the non-ABAP system in its primary key. So it is possible to trigger imports in multiple systems. The AS ABAP based SAP system on which *tp* is called is used as *communication system* for the non-ABAP system.

### More Information

For more information, see the online documentation for SAP S/4HANA (*Product Assistance*), area *Enterprise Technology* → *ABAP Platform* → *Administrating the ABAP Platform* → *Administration Concepts and Tools* → *Solution Life Cycle Management* → *Software Logistics* → *Change and Transport System* → *Transport Tools (BC-CTS-TLS)* → *Transport Control Program tp* → *How tp Works* → *Communication Between tp and ABAP*.

### Transport Commands

The transport control program **tp** is used for controlling transports between SAP systems, applying SAP Support Packages, and for performing release upgrades. The *tp* program tracks transports by controlling exports and imports of objects between SAP systems. The program also ensures that the steps for exporting and importing objects are performed in the correct sequence and that imports into a target system are done in the same sequence as the exports from the source system.

*tp* is an operating system level program, which uses special programs, such as C programs, operating system commands, and ABAP programs in the SAP system.

Exports and imports are performed separately:

- In the export phase, the objects to be transported are extracted from the database of the source system and are stored in files in the transport directory. An export always takes place immediately after the transport request has been released so that the objects are freed for further changes. *tp* uses the program **R3trans** at operating system level to establish a connection to the database and to perform the necessary steps in the database.
- In the import phase, the objects are added to the database of the target system.

**Note:**

There is no automatic mechanism that imports a transport request into a target system immediately after export.

Although most transport activities can be performed using TMS, you may find the need to use *tp* commands for exceptional cases. Because the import queues are the representation of the import buffer file in the SAP system, the use of *tp* on operating system level and from within TMS produces consistent results.



**Usage: `tp <command> [argument(s)] [option(s)]`, where <command> may be one of:**

Exporting		I Buffer Actions	I Disk Space	
EXPORT	I EXPWBO	I ADDTOBUFFER I DELFROMBUFFER I CLEAROLD		
R3E	I VERSE	I SETSYNCMARK I ADDSYNCMARK I DELSYNCMARK		
CHECK	I SDE	I SETSTOPMARK I DELSTOPMARK I TESTOLD		
TST	I	I MARK I UNMARK I		
	I	I CLEANBUFFER I	I	
Importing		I Special Functions	I Information	
PUT	I	I LOCKSYS I UNLOCKSYS I SHOWBUFFER		
IMPORT	I	I LOCK_EU I UNLOCK_EU I COUNT		
CMD	I R3I	I WRITELOG I GETPROTS I SHOWPARAMS		
	I	I SAPSTART I SAPSTOP I GO		
R3H	I	I GETTBATGENT I GETTRBATENT I CONNECT		
ACT	I MVKERNEL	I GETDDXTTENT I GETDDXTFENT I CHECKIMPDP		
SXPRA	I SDI	I GETOPENCORR I CREATECOFILE I SHOWINFO		
GENTB	I GENT1	I PREPAREBUFFER I CREATEINFO I GETOBJLIST		
GENT2	I GEN	I CONTP I EXPCHK I EXTRDOCU		
DIST	I XPA	I SHOWSEMAPHORE I SHOWSEMACLIENT I GETCONVENT		
MVNTABS	I VERSI	I SETSEMAPHORE I DELSEMACLIENT I EXPLAINRC		
IMPSYNC	I DELIVER	I DELSEMAPHORE I HANADEPLOY I GETTPSTATENT		
DEPLOY	I IMPORTPREVIEW I CHECKIN	I CHECKINMOVE I CHECKOUT		
CHECKDEPLOY	I GETKEYTRACE I MERGELOGS	I MEPP I VCSCommit		



For more details call `tp` with the desired command (and no other parameters)

Figure 106: Overview of tp Commands (tp help)

`tp` commands are executed in the transport subdirectory `bin` by user `<SID>adm`. `tp` commands are called using syntax `tp <command> [argument(s)] [option(s)]`.

- To display general information on the syntax of all `tp` calls, use `tp help`. To display a description of the syntax and function of a specific command, use `tp <command>` without any options.
- To count the number of all transport requests registered for import into a specific SAP system, use `tp count <SID>`.
- To test whether a connection to an SAP system's database was successful, use `tp connect <SID>`.
- To display how the import dispatcher `RDDIMPDP` is scheduled for a specific SAP system, use `tp checkimpdp <SID>`.
- To display all current settings of the `tp` parameter file for a specific SAP system, use `tp showparams <SID>`.



#### Hint:

You always have to specify the full qualified path to the `tp` parameter file `TP_<domain name>.PFL` by using the parameter `pf=<(path to )TP_<domain name>.PFL`.

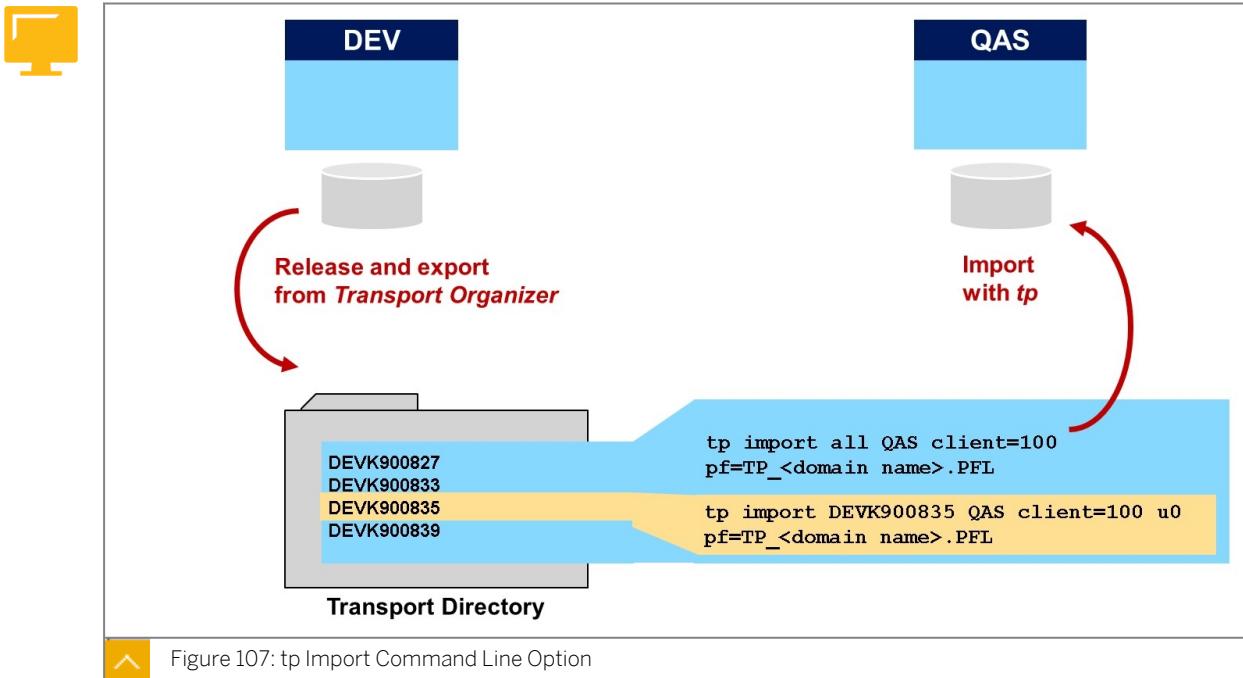


Figure 107: tp Import Command Line Option

Imports performed using *tp* at operating system level should be restricted to exceptional cases, for example if the TMS is not available or special *tp* commands are needed.

There are two commands to start an import using *tp* at the operating system level:

- `tp import all <target SID> client=<client number> pf=TP_<domain name>.PFL`

Analogous to the use of the *Import All Requests* button in TMS, this command imports all waiting transport requests in the correct sequence.

- `tp import <transport request> <target SID> client=<client number> u0 pf=TP_<domain name>.PFL`

Analogous to the use of the *Import Request* button in TMS, this command imports a single transport request into the target system (and leaves it in the queue for later import).

Importing individual transport requests should only be performed in exceptional cases as the correct sequence of transport requests is not necessarily maintained and thus newer versions of objects may be overwritten by older versions through the regular import of all waiting transport requests. To ensure that the objects imported through an individual transport request are not overwritten by an older version, make sure that you use unconditional mode 0, that is, the option *u0*, when starting individual imports (see figure above). Using this mode is analogous to the TMS preliminary import and causes the transport request to remain in the list of transport requests to be imported. When the regular (mass) transport takes place, the transport request will automatically be imported again in the sequence of the import queue.

**Caution:**

Processing imports in the wrong order can result in severe inconsistencies in the target system. These are hard to diagnose. If you do not want to import single transport requests as preliminary import (**NOT recommended** in case of transport strategy *Queue-Driven Transports, Mass transports*), you would use `tp import <transport request> <target SID> client=<client number> pf=TP_<domain name>.PFL`, without the option `u0`.

The contents of transport buffer files in the transport subdirectory *buffer* are organized as a table. Each line contains information on a specific transport request. One column includes the unconditional mode that is linked to the transport request. Other columns specify import actions.

`tp` offers several commands for accessing buffers:

- `tp showbuffer <SID>`

This command displays transport buffer entries for system `<SID>`.

- `tp addtobuffer <transport request> <target SID>`

This command registers the transport request `<transport request>` at the end of the import queue of system `<target SID>`.

- `tp delfrombuffer <transport request> <target SID>`

This command deletes the single transport request `<transport request>` from the transport buffer file of system `<target SID>`.

- `tp cleanbuffer <SID>`

This command removes successfully imported transport requests from the transport buffer file of system `<SID>`. This function is included in the command `tp import all <target SID>`. This action can be started in transaction `STMS` by choosing the menu paths *Overview → Imports* and then (after selecting a queue) *Import Queue → Display* and finally *Extras → Delete Imported Requests*.

- `tp setstopmark <SID>`

This command places a stopmark at the end of the transport buffer for system `<SID>` so that subsequent `tp import` commands only process the transport requests that are located before the stopmark.

- `tp mvstopmark <transport request> <SID>`

This command moves the stopmark in front of the transport request `<transport request>` in the transport buffer file for system `<SID>`.

- `tp delstopmark <SID>`

This command deletes the stopmark from a transport buffer file for system `<SID>`.

**Hint:**

You always have to specify the full qualified path to the `tp` parameter file `TP_<domain name>.PFL` by using the parameter `pf=<(path to )TP_<domain name>.PFL`.

## Unconditional Modes

Unconditional modes are among the options that can be assigned to *tp* commands. They are used to cause specific rules of the Change and Transport System (CTS) to be ignored.

Some examples of *tp* unconditional import modes are:



Table 2: Unconditional Modes

0	During Import	Leave transport request in queue for later import: Import from transport buffer without deleting from transport buffer and set unconditional mode I in transport buffer to trigger another import
1	During export	Ignore incorrect status of a transport request.
1/I	During import	Import transport request again: ignore that transport request was already imported into SAP system and import again from beginning
2	During import	Overwrite originals
3	During import	Ignore predecessor relations
4	During import	Ignore invalid component version
6	During import	Overwrite objects in unconfirmed repairs
8	During export	Ignore invalid table class: allow selection of tables, some tables are part of complex objects and generally can't be exported alone
8	During import	Ignore invalid table class: ignore the restriction resulting from table classifications and import all table entries into specified clients
9	During import	Ignore invalid transport type



Hint:

The unconditional modes may vary in dependence of the SAP system and *tp* release.

Each unconditional mode is represented by a digit or a character. To use an unconditional mode, add a **u** to the *tp* command line and add the required digit(s). For example, if you want to import request DEVK900132 into the quality assurance system and overwrite originals, enter the following command: `tp import DEVK900132 QAS u2`.



Caution:

Use unconditional modes carefully. It is preferable to transport according to the rules of the *Change and Transport System* (CTS).

Some unconditional modes can be used directly in TMS.



## LESSON SUMMARY

You should now be able to:

- Analyze the different steps during the import of transport requests

- Explain the use of tp commands



# Monitoring Tools

## LESSON OVERVIEW

In this lesson, you will learn about the tools provided by the Transport Management System (TMS) for monitoring transport activity in your transport domain. You will also learn ways to ensure a stable test environment and a stable development environment.

### Business Example

As the administrator who is responsible for software logistics, you must be able to track the progress of the imports and to make sure that only the required transport requests are being transported. The Transport Management System provides the administrator with a variety of tools to monitor all aspects of the import process and to ensure that the import is being performed with few or no errors.



## LESSON OBJECTIVES

After completing this lesson, you will be able to:

- List selected monitoring tools and explain their use
- Explain the content of the transport directory
- Troubleshoot typical import errors

## Monitoring Tools

The Transport Management System (TMS) provides various tools with which you can monitor transport activities in your transport domain.

## Monitoring Tools

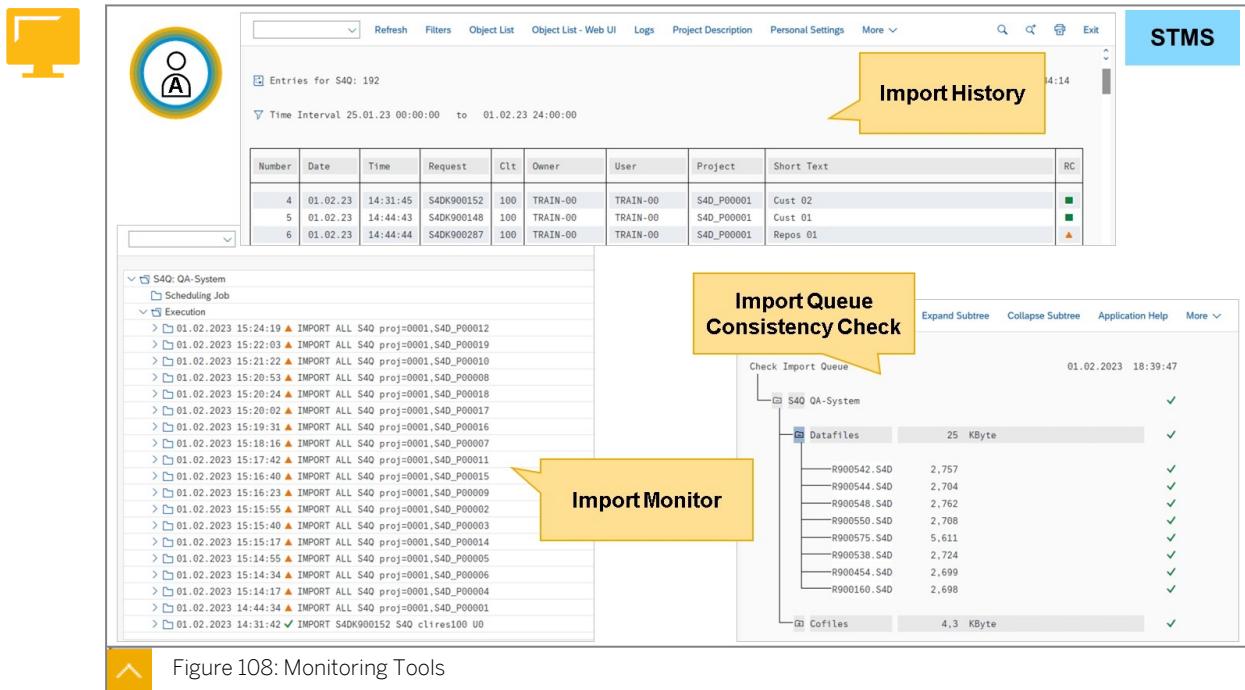


Figure 108: Monitoring Tools

The tools can be accessed as follows (some of them are shown in the figure above):

- In transaction STMS, choose *Overview → Systems* from the menu and mark one or more SAP systems:
  - *RFC connection test:* To check RFC destinations for all or just one system within the transport domain in both directions, choose *SAP System → Check → Connection Test*.
  - *Transport directory check:* To verify the availability of the transport directories for all or just one system within the transport domain, choose *SAP System → Check → Transport Directory* → *Transport Directory*.
  - *Transport control program check:* To check the transport control program *tp* for all or just one specific SAP system within the transport domain, choose *SAP → System → Check → Transport Tool*.
- In transaction STMS choose *Overview → Imports* from the menu and double-click the SAP system in question:
  - *Import history:* To display the import history, choose the menu path *Goto → Import History*.

### Note:

The import history is collected from the log file *ALOG* and is kept in the tables *TPALOG* and *TPALOGHDR*. For details, see SAP Note [375230](#) – TMS: incomplete import history.

- *Import monitor:* The transport control program *tp* stores status information in the database before and after each import step. This status information is read and

displayed by the *import monitor* which can be accessed by choosing the menu path *Goto → Import Monitor*.

- *Import queue consistency check*: To check if the data files and cofiles of transport requests in an import queue exist in buffers of the transport directory and can be read, choose the menu path *Queue → Check → Consistency*.

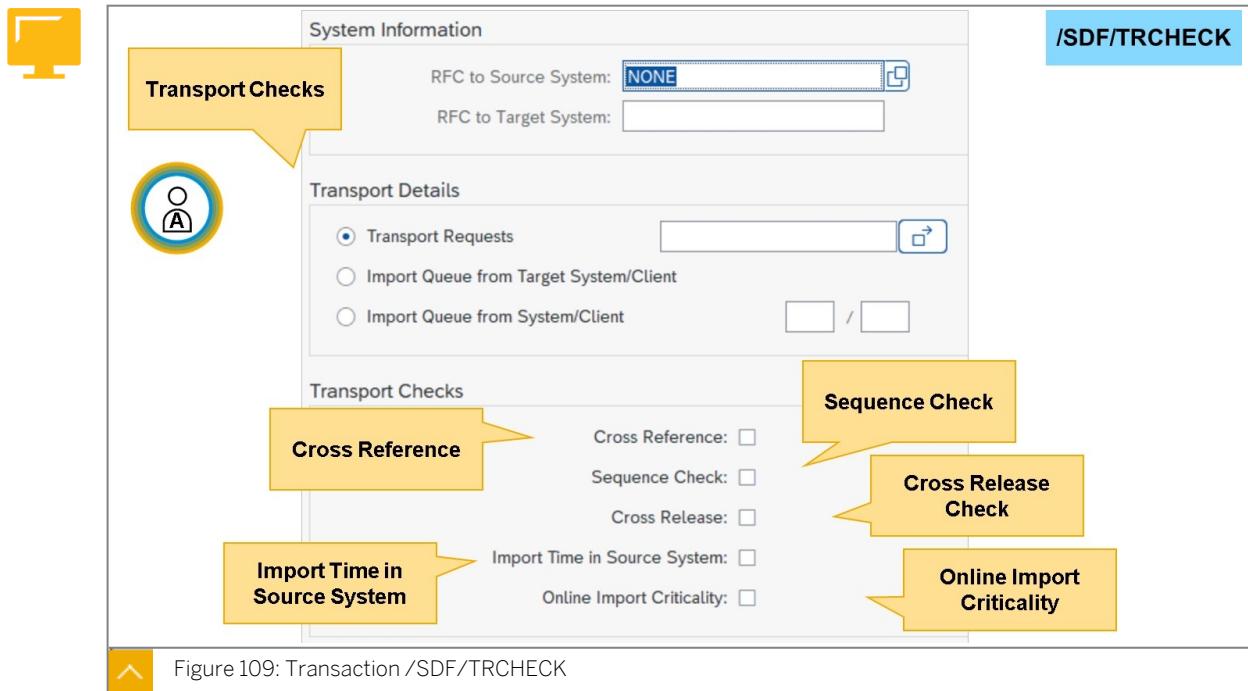


Figure 109: Transaction /SDF/TRCHECK

Transaction /SDF/TRCHECK or report /SDF/CMO\_TR\_CHECK provides various proactive checks for objects in transport requests. It predicts transport related errors before the requests are imported into a target system. Typical use cases are that developers check their transport request in the development system, before they release it and import it into the test system. Or a transport manager checks a number of transport requests in a test system, before they will be imported into the preproduction or production system. The checks are delivered with the ST-PI Plugin. For more information, see SAP Note [2475591 – Transport Check Report](#), which also lists required SAP Notes.

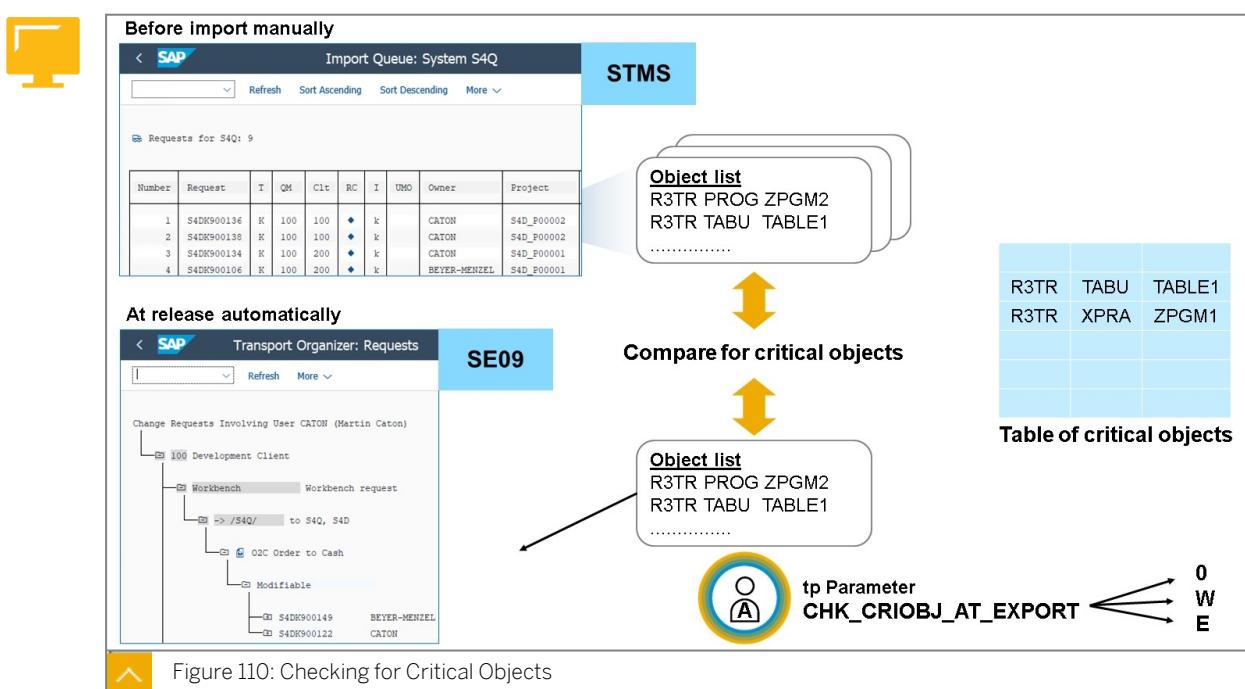
These checks are available:

- *Cross Reference*: For all objects in the selected transport requests, the referenced objects are identified by a where-used-analysis. If the referenced objects are not included in the transport requests, we compare their versions between the reference and target system. If the versions are different or if the referenced objects do not exist in the target system, it will be highlighted as a potential error. In addition we show the last transport requests for the missing object versions. This check works for ABAP repository, data dictionary, customizing objects, SAP Notes and BW objects.
- *Sequence Check*: The sequence check identifies other transport requests with identical objects which have been released in the analysis period, but have not yet been imported into the target system.
- *Cross Release Check*: If the current system and the target system are on different support package levels, this check identifies critical objects in the selected transport request, which belong to inconsistent software components and should not be imported into the

target system - for example SAP Notes. For customizing objects we compare in addition if the table structure is different in the reference system and target system.

- *Import Time in Source System:* The import time of the selected transport requests in the source system is summed up. For this check the source system should be a test system in which the transport requests have already been imported.
- *Online Import Criticality:* This check estimates the criticality of an import when the end users are working in the production system. As a prerequisite you must first collect the table call statistics and the report execution statistics in the production system for one week. The collection of the usage statistics must be triggered in the production system with the report /SDF/OI\_ADMIN. You must first activate the *Usage and Procedure Logging (UPL)* in the production system. The report identifies the dependent objects of the transported objects and checks the usage profile of all objects. For tables, the number of table reads per hour, table writes per hour and the table size in KB is shown in the output. For reports, the number of report execution steps per hour based on the UPL data is shown. In addition, you can maintain a list of critical objects with regard to online import in table /SDF/OI\_CRIOBJ in the production system. These objects are then also shown in the result.

## Checking for Critical Objects



There are two options to check for critical objects:

### Before the import of transport requests into the target system:

This option has to be performed manually and is only a display of the list of transport requests that contain critical objects.

### During the release of the transport request:

This option is performed automatically, if *tp* parameter *CHK\_CRIOBJ\_AT\_EXPORT* is set to *W* (warning) or *E* (error).

To check if the transport requests in an import queue contain critical objects that should not be imported into the target system, go to the *Import Overview* (transaction *STMS*, menu path *Overview* → *Imports*), double-click the system in question and choose

Queue → Check → Critical Objects from the menu. The object lists of the transport requests are checked to see if they contain critical objects. The result is displayed in a hierarchical list. Transport requests containing critical objects are marked with the appropriate icon. The critical objects are highlighted in color.

Before performing this check, you must have maintained which objects are classified as critical. For this, you must be logged on to the transport domain controller system. The information about critical objects is distributed to the whole transport domain when you save your changes.

To display or maintain the critical transport objects defined for the transport domain, call transaction STMS on the transport domain controller system. Choose the menu path Overview → Imports and then the menu path Extras → Critical Transport Objects.



Note:

You can also check requests for critical objects in the QA worklist. This supports the decisions you make on approving or rejecting transport requests.

If you want to check the whole QA worklist for critical objects, enter QA worklist and choose the menu path Worklist → Check → Critical Objects. You may need to confirm a dialog box. An overview of the transport requests with critical objects appears. If you only want to check certain transport requests for critical objects, you may filter them in the display.

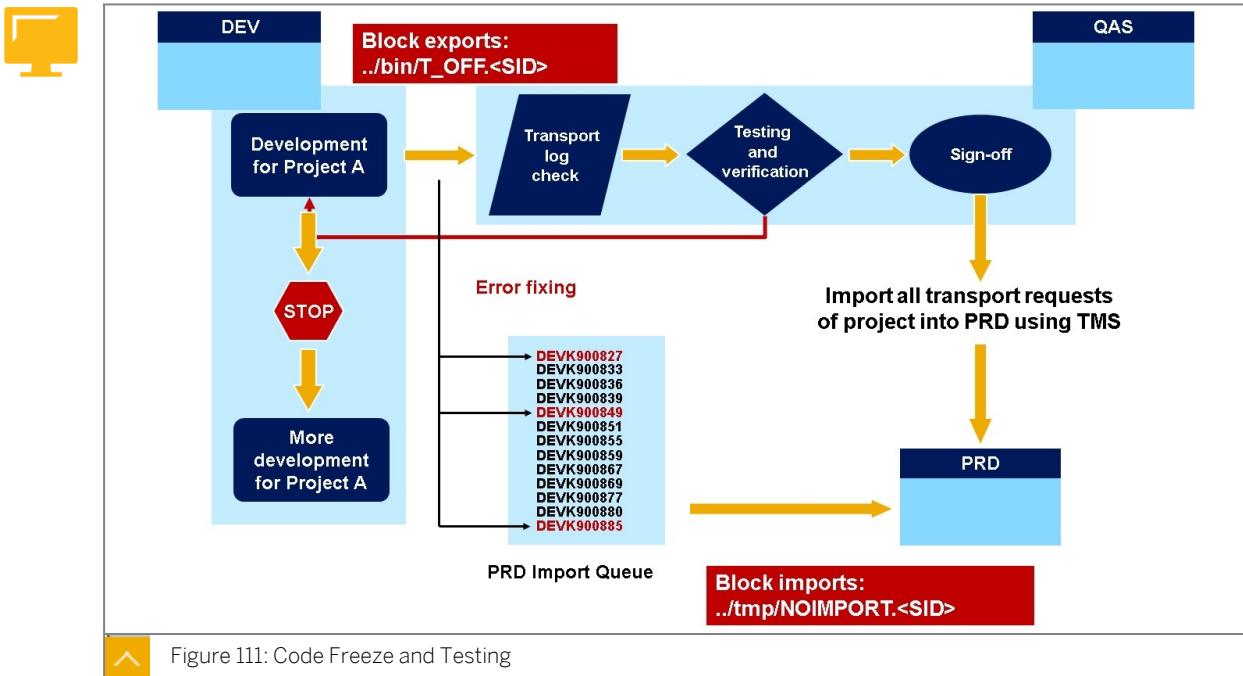
Only objects of the form *Object type = R3TR* are checked as transport objects. Transport objects that start with *LIMU* are sub-objects of a repository object with an object directory entry. To check these, you must find out the object directory entry, and enter it in the table of critical objects.



Hint:

Interesting objects to be classified as critical, would be, for example, *R3TR XPRA* (XPRAs) or *R3TR TABL* (table definitions) or *R3TR TABU* (table contents, for interesting tables).

## Code Freezing and Testing



To ensure a stable test and development environment, use a development deadline to freeze work on objects in the development system until the completion of the quality assurance verification.

Follow this procedure during object development and testing:

1. Release the transport requests containing the developed objects.
2. Freeze development of the objects in the development system.
3. Import the objects and verify the changes in the quality assurance environment.
4. Sign off the changes.
5. If necessary, allow further development on the objects in the development system.

With the following tools, an administrator can force code freezing for an SAP system landscape. There are two main options:

- **Stop exports** by creating a file `T_OFF.<SID>` in transport directory `/usr/sap/trans/bin`. The first line of this file will be displayed when you start a release of a transport request from the *Transport Organizer*.
- **Stop imports** by creating a file `NOIMPORT.<SID>` in transport directory `/usr/sap/trans/tmp`. The contents of this file are not evaluated.

When an object has gone through the error fixing cycle, it is included in at least two different transport requests within the import queue of the production system: the original transport request and the transport request containing the fix. If you import the complete import queue, using *Import all requests* into the production system after sign-off, the object from the original transport request has no impact on your production environment.

**Note:**

SAP recommends transporting all transport requests of a whole CTS project together in a single step when all objects are in an acceptable state rather than each single object as soon it is ready.

## Transport Directory Naming Conventions

Because the transport control program *tp* runs on many different operating systems, restrictive naming conventions are required. Transport requests are always represented in the following format <source SID>K9<5 digits or characters>, where <source SID> is the SAP system in which the transport request was created, and K9 indicates a customer transport request. The five digits form a serial number, which can be expanded using characters.

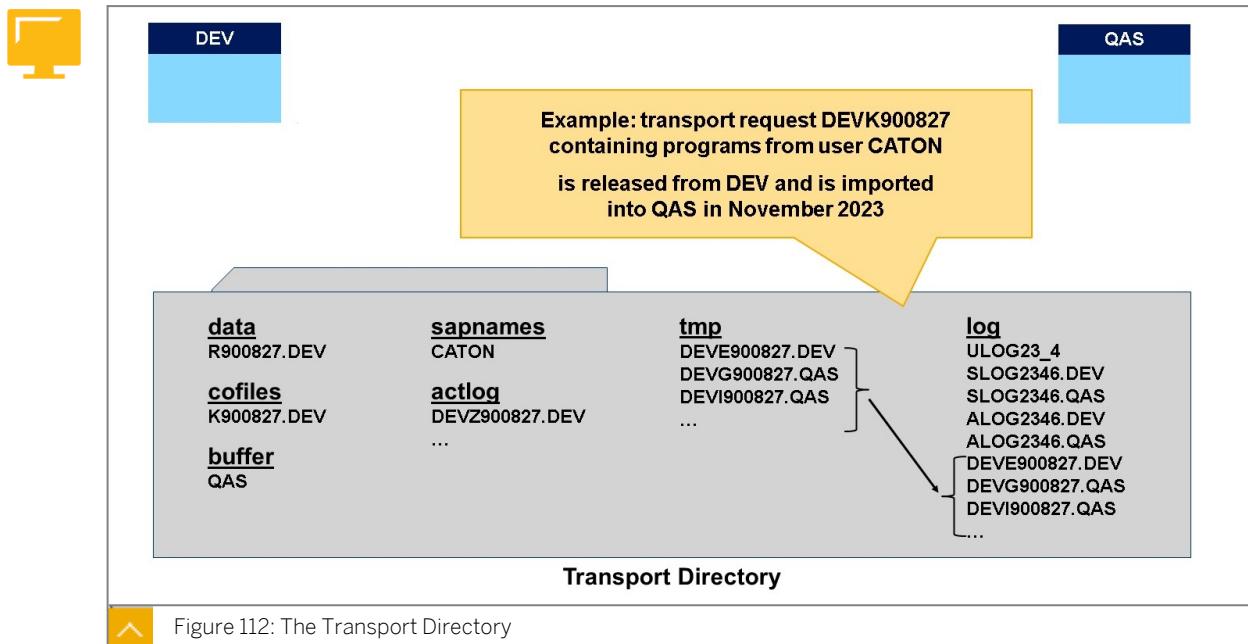


Figure 112: The Transport Directory

The transport directory contains subdirectories, such as *actlog*, *bin*, *buffer*, *cofiles*, *data*, *log*, *sapnames*, *tmp*, and *EPS*.

### actlog

This subdirectory contains files for the tasks of the transport requests and the transport requests itself. A file is created for each transport request and for each task for which actions have taken place, and is updated when a new action takes place, for example, the creation or release of a transport request.

### bin

This subdirectory contains the configuration files for the transport domain.

### buffer

This subdirectory contains a transport buffer file <SID> for each SAP system. When a transport request is released, the transport buffer file of the target systems is updated.

### cofiles

This subdirectory contains command files named *K9<5 digits>.<source SID>*. They contain, for example, a list of the performed import steps.

#### data

This subdirectory contains files named *R9<5 digits >.<source SID>* containing the exported objects.

#### log

This subdirectory contains all log files, such as *ULOGs*, *ALOGs*, *SLOGs*, and log files named

- *<source SID><action>9<5 digits>.<action SID>* for each executed step (for example, with *<action> = I* for *main import* or *A* for *activation*) and
- *<action><date>.<action SID>* for steps that are collectively executed, for example for *step structure conversion (N)* or *step move nametabs (P)*.

#### sapnames

This subdirectory contains files named after the user's log on name (without any non-standard characters). A file is created for each SAP system user, who works with the CTS, and is updated when a transport request is released. Note that for repair-related transport requests, a file for the owner of the repaired objects is created as well. For more information, see SAP Note [2379949 – sapnames logging](#).

#### tmp

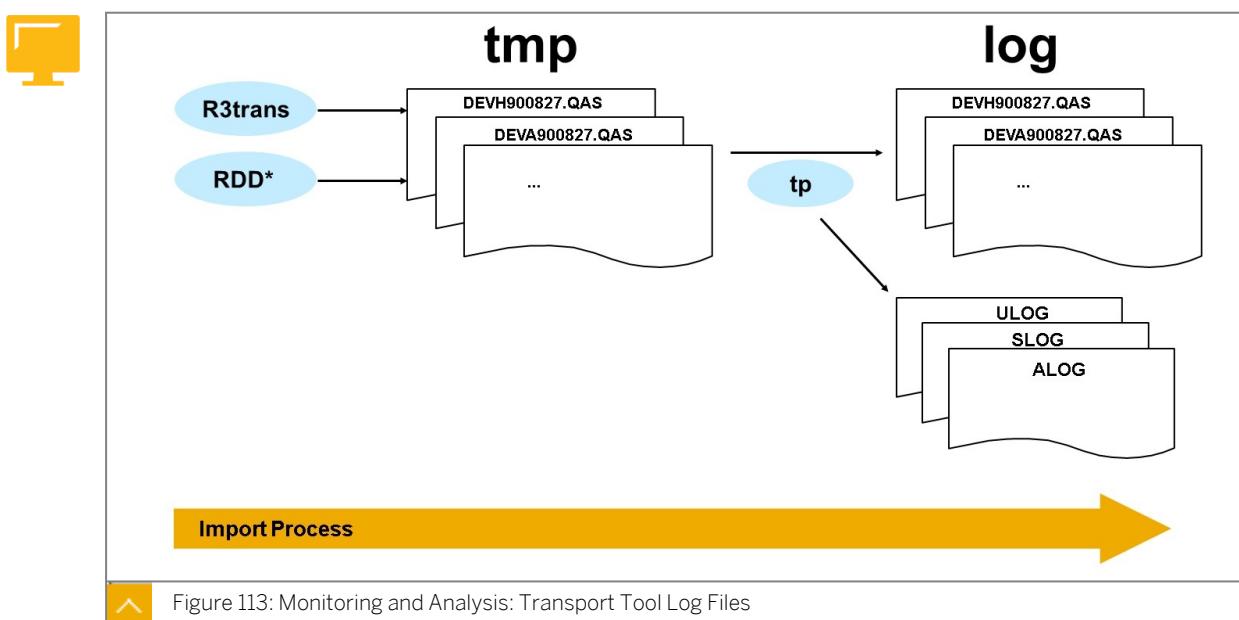
This subdirectory contains log files before they are moved to the *log* directory.

#### EPS

This subdirectory (Electronic Parcel Service) (among others) contains the subdirectories *in* and (optional) *download*, to which SAP Support Packages can be copied in order to apply them with the SAP Support Package Manager (transaction *SPAM*).

## Troubleshooting Steps

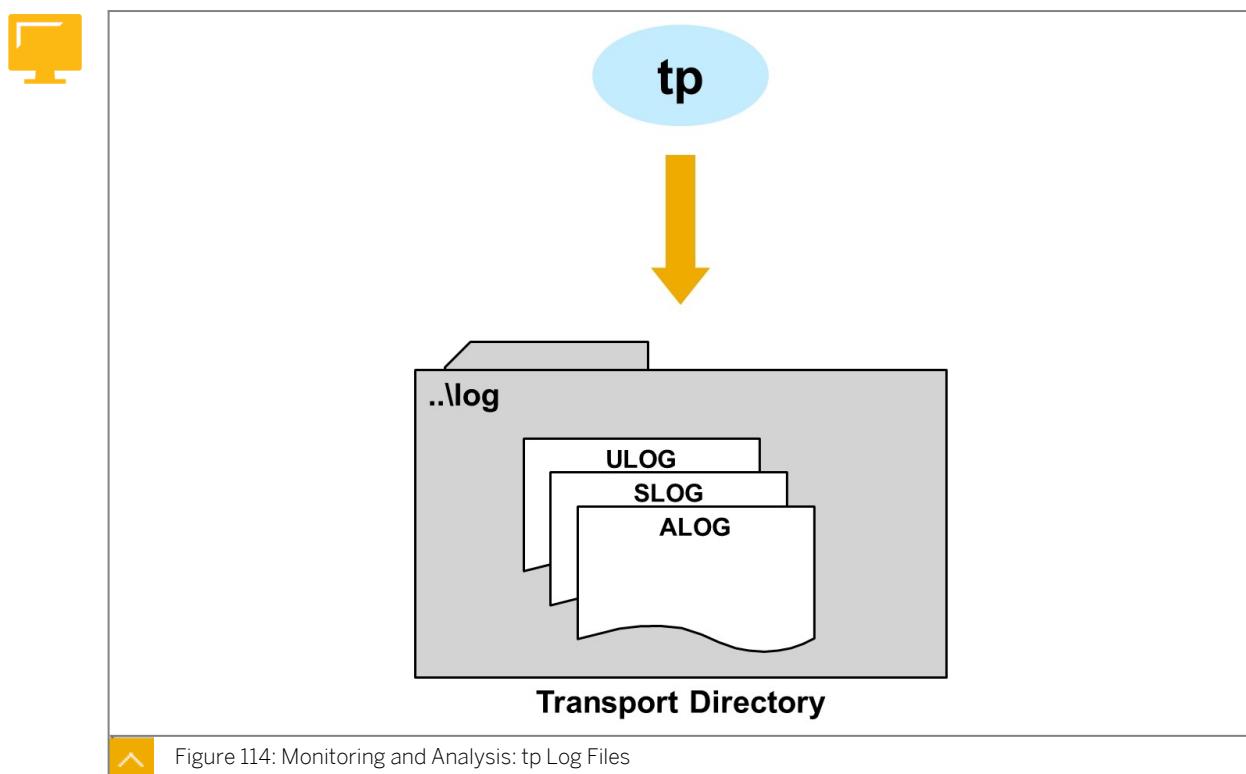
### Log Files



The various transport tools write a log for each transport action to the transport subdirectory `tmp`. After completion of this step, `tp` moves these logs from `tmp` to `log`. The log files are named `<source SID><action>9<5 digits>.<target SID>`, where `<action>` is represented by a single character and `<5 digits>` is taken from the corresponding transport request.

Each of those log file can be displayed from within in the SAP system. By expanding the log display, you can choose different detail-levels in the log file. The possible levels are as follows:

- Performed actions and return code
- Additional error messages
- End-user logs
- Details for developers and hotline



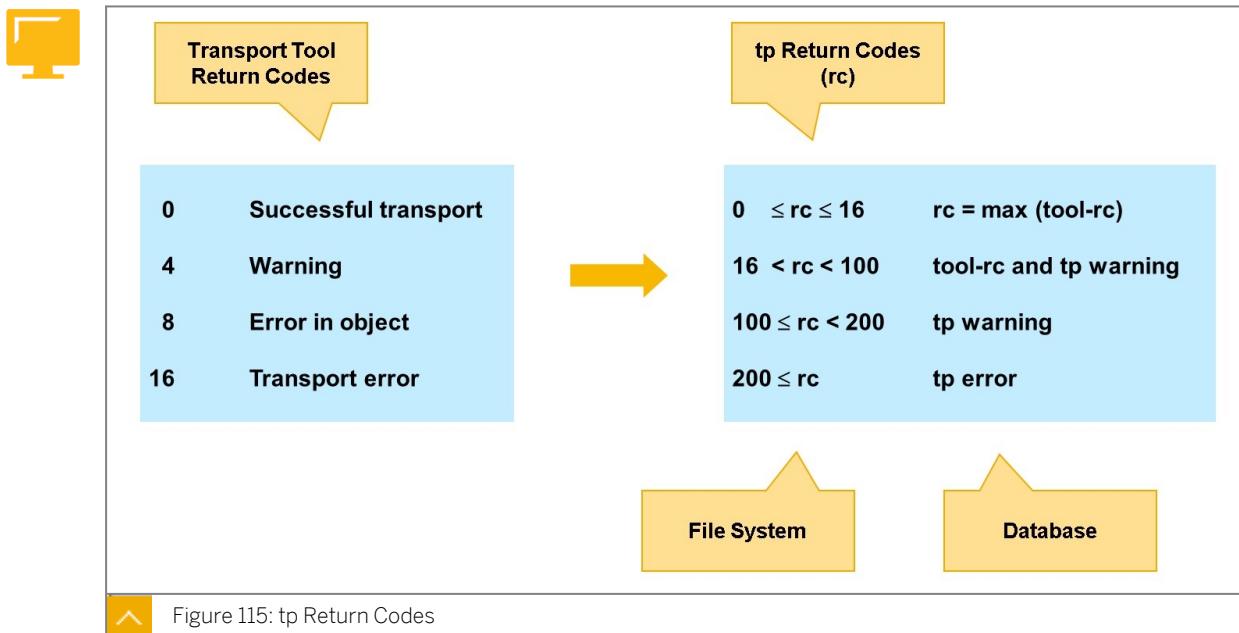
For long-running imports, it may be helpful to monitor the log files written at the operating system level. All logs in the transport environment are stored in the transport subdirectory `log`. These logs include logs created by `tp` (`ULOG`, `SLOG` and `ALOG`), and logs created by the various transport tools.

The current `ULOG` file records all `tp` commands that are free of syntax errors, and is named using the name convention `ULOG<YY>_<Q>` (YY is the year, Q is the quarter of the year). Each line in the `ULOG` file represents a `tp` command.

The `SLOG` file is used to monitor the transport activities of a specific SAP system. It contains a general overview of performed transports indicating the return code and thus the success of each transport. The name of the `SLOG` file can be set as `tp` parameter using the global `tp` parameter `SYSLOG`. The default setting is `SLOG<YY><WW>.<SID>` (YY is the year, WW the calendar week).

The `ALOG` file records the return code for all transport steps handled in the common transport directory. The name of the `ALOG` file can be set as `tp` parameter using the global `tp` parameter `ALLLOG`. The default value is `ALOG<YY><WW>.<SID>`.

## Return Codes



*tp* receives return codes from all the transport tools involved in an import process. *tp*'s own return code is interpreted as follows:

- 0 to 16 indicate the maximum value of all return codes from transport tools.
- 17 to 99 are values that are calculated from the return codes of the transport tools and a *tp* warning, for example, the transport buffer of the target system has no write permission.
- 100 to 199 indicates *tp* warnings. *tp* warnings mean that something went wrong and *tp* could not perform all the tasks. 100 to 149 are normal *tp* warnings, for example, *RDDIMPDP* cannot be triggered by *sapevt*. Return codes of 150 to 199 are rare and indicate incorrect operation by a user. For example, a return code of 152 is received if *tp* tries to import a transport request that is not included in the transport buffer.
- 200 or more indicates *tp* errors. For example, if a file could not be accessed as required by the import process, the return code is 212.

The accompanying text is more significant than the value of a return code. To display the text of a specific *tp* return code, use the *tp* command `tp explainrc <value of return code>`.



Note:

For more information, see SAP Note [2878102 – Meaning of tp return code](#).

## Troubleshooting

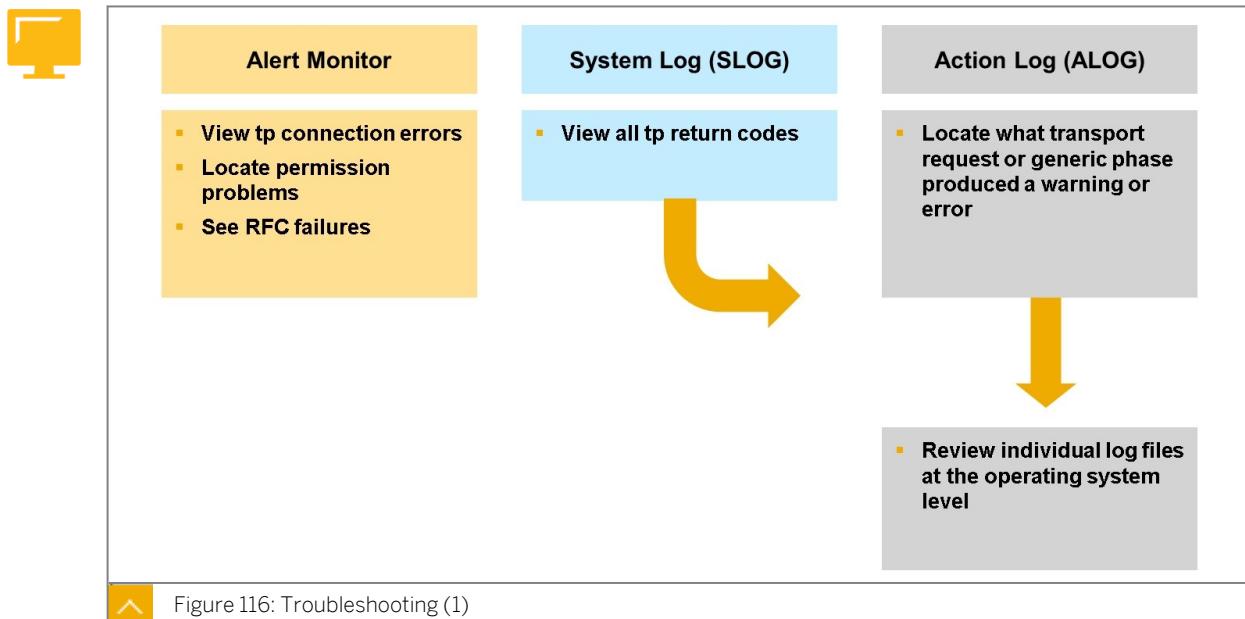


Figure 116: Troubleshooting (1)

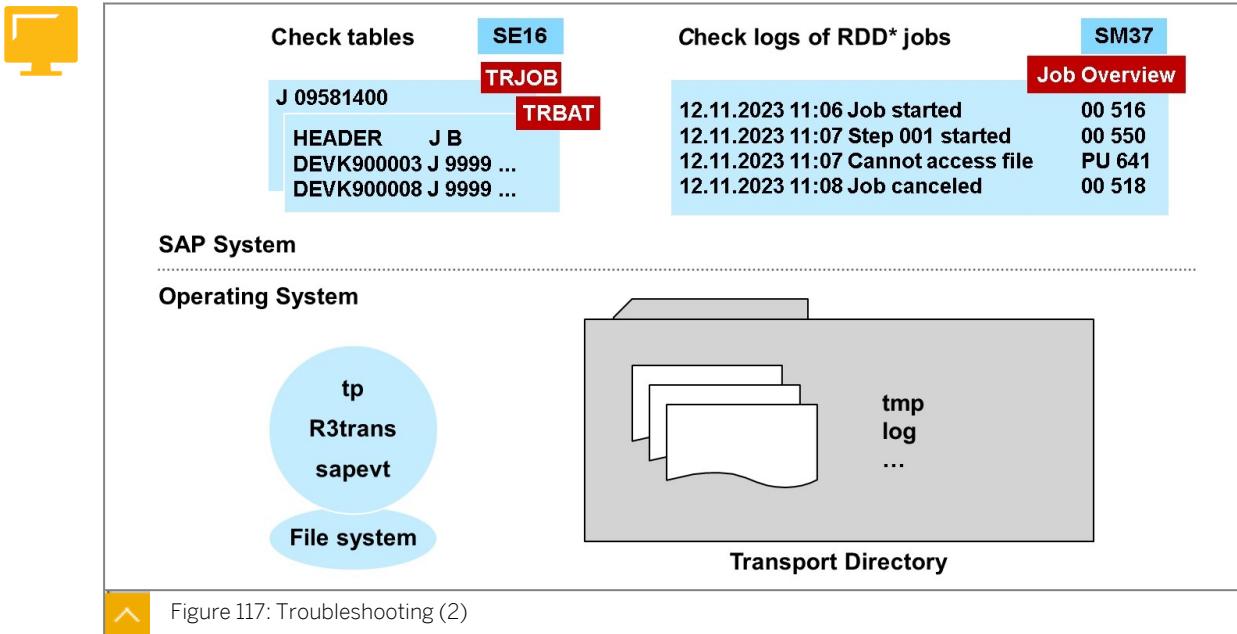
The first step in troubleshooting transport errors is to use the *alert monitor*, which records all TMS transport actions. The alert monitor is available in transaction STMS by choosing the menu path *Monitor → TMS Alerts → TMS Alert Viewer*. The information displays the date and time, the user name, the TMS status message, and the target SAP system. To display the full text of an error message, double-click the error message.

More detailed information can be seen in the *SLOG* file, which is used to monitor the transport activities of SAP systems and to determine the success of import requests.

If import failures are recorded in the *SLOG* file, drill down to the *ALOG* file and locate the import step that produced the return code listed in *SLOG*.

Use the *ALOG* file to identify the detailed log file, which is written for each step of a transport request. These log files can also be accessed from the Transport Management System (transaction STMS). To locate the log file for the transport request that produced an error, use the *ALOG* file.

All log files that are not dependent on specific transport requests, such as the log files for *structure conversion* and *move nametabs* can be accessed both from the operating system level in the transport directory and from the import queue of the SAP system in question (in transaction STMS) by selecting a transport request, choosing *Logs* and then expanding the folder *Import steps not specific to transport request*.



Additionally, you can check whether the import dispatcher *RDDIMPDP* is scheduled correctly and is event-triggered. Use the job overview **SM37** to monitor the related background jobs (*RDD\**). Here, enter **RDD\*** in the *Job name* field and an asterisk **\*** both in the *User Name* and in the *Or after event* field.

Problems might result from:

- Wrong versions of *tp* or *R3trans*
- *tp* not running, as in UNIX (`ps -ef | grep tp`)
- Permission or share problems with the common transport directory
- No free disk space

When analyzing a problem, compare the logs and transport buffer entries with the entries in tables *TRBAT* and *TRJOB* (using transaction **SE16**). If required, insert the transport request or header in *TRBAT* and restart *RDDIMPDP*.



#### Hint:

If a communication problem between *tp* and the SAP system is indicated, try to start *sapevt* at operating system level to trigger *RDDIMPDP*.



#### Note:

Because *sapevt* communicates in an unauthorized way with the message server, the program may no longer function in general if secure message server communication is active. An exception is when *sapevt* is called on the application server using the operating system user that owns the SAP system. In this case, the parameter `pf=<instance profile>` must also be passed. For more information, see SAP Note [2000417 – Problems with SAPEVT as of kernel release 7.40](#).



## LESSON SUMMARY

You should now be able to:

- List selected monitoring tools and explain their use
- Explain the content of the transport directory
- Troubleshoot typical import errors



# Cleaning up the Transport Directory

## LESSON OVERVIEW

In this lesson, you will learn how to clean up the transport directory.

### Business Example

From time to time, it makes sense to clean up the transport directory.



## LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Name tp commands to clean up the transport directory

## Cleaning up the Transport Directory

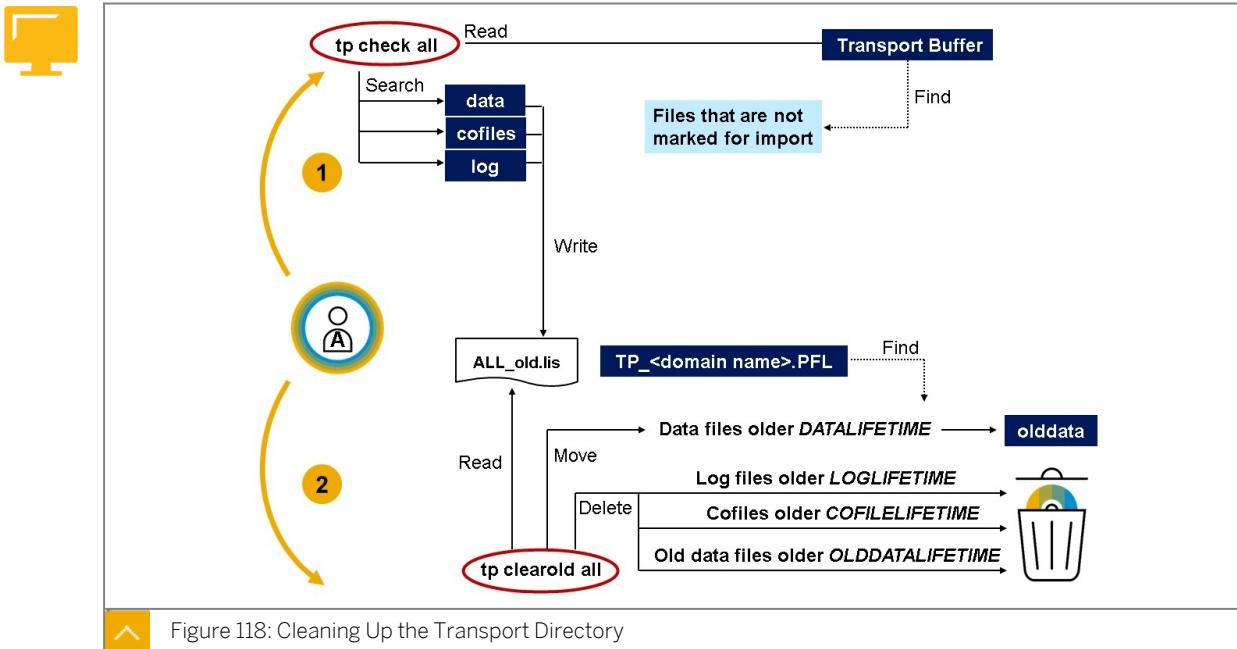
Data files, cofiles, and different log files are written to the transport directory. These files are not deleted, overwritten, or reorganized automatically, so the transport administrator has to take care of them.

To clean up the transport directory, use the commands `tp check all` and `tp clearold all`. Over time, many files accumulate in the transport directories. These files contain important information, but they also use up disk space and become obsolete.



Hint:

Before cleaning up the transport directory, SAP recommends making a copy of the transport directory and saving it for audit purposes.



The command `tp check all` searches for transport requests that are not marked for import into any SAP system, as revealed by the transport buffer entries. The transport requests are listed in file `ALL_old.ls`, which is located in transport subdirectory `tmp`.

The command `tp clearold all` uses the result list of the command `tp check all` to find files that are not needed anymore for import and that have exceeded a maximum age. To specify the maximum age, use the `tp` parameters `DATALIFETIME`, `OLDDATALIFETIME`, `COFILELIFETIME`, and `LOGLIFETIME`. Data files in the transport subdirectory `data` that are older than the parameter `DATALIFETIME` are moved to the transport subdirectory `olldata`. Files in the subdirectories `log` and `cofiles` are immediately deleted if they exceed the age specified by `LOGLIFETIME` or `COFILELIFETIME` respectively. Files in `olldata` are deleted if they are older than `OLDDATALIFETIME`.



#### Hint:

With the command `tp testold all`, it is possible to create a list of files which will be deleted by the command `tp clearold all`. The result is located in subdirectory `tmp`.



#### Note:

The files in the `EPS` subdirectory of the transport directory are not affected by this clean-up. The transport administrator has to care about them manually.

## More Information

For more information, see:

- SAP Note [41732](#) – *Deletion of data in transport directory*.
- SAP Note [1694830](#) – *tp clearold extension for cts+ directories and tmp directory*.



## LESSON SUMMARY

You should now be able to:

- Name tp commands to clean up the transport directory



# Learning Assessment

1. You are running a development project in a three-system landscape using the *Quality Assurance (QA) approval procedure*. After fixing some errors that have been detected during the test, quality assurance testing in the quality assurance system for the project finally is positive. Which transport requests should you import into the production system?

*Choose the correct answer.*

- A The entire project which includes all approved transport requests assigned to the project
- B Exactly those transport requests that contain only error-free objects
- C If the same objects are contained in multiple transport requests, only import the newest transport requests
- D All transport requests from the import queue of the production system

2. You are using a standard three-system landscape: When releasing a transportable transport request from the development system, it is being exported from the development system and immediately imported into the quality assurance system.

*Determine whether this statement is true or false.*

- True
- False

3. You are using client independent transport routes. Which information can be seen in the import queue (transaction STMS) for transport requests that have not imported yet?

*Choose the correct answers.*

- A The owner
- B The source client
- C The component check result
- D The type
- E The target client

4. You are using a standard three-system landscape. You did not add or delete any transport requests manually from the import queues. What is the sequence of the transport requests in the import queue of the production system?

*Choose the correct answer.*

- A The sequence in which the transport requests were imported into the quality assurance system
- B The sequence in which the transport requests were released from the development system
- C The sequence of the time stamp of the creation of the transport requests
- D The sequence of the 5-digit number of the transport requests

5. Which of the following are transport strategies of the Transport Management System (TMS)?

*Choose the correct answers.*

- A Queue-Driven Transports, Mass transports
- B Queue-Driven Transports, Single transports
- C Workflow-driven transports
- D Queue-Driven Transports, Project based transports

6. What is different when using a transport of copies instead of a workbench request?

*Choose the correct answers.*

- A You can choose any SAP system of the transport domain as the target system
- B You do not need any consolidation route to the target system
- C You can transport together any transportable objects
- D Copies of the objects are being transported, not the originals
- E The objects are being copied from the source system

7. You are using a standard three-system landscape. By using the Transport Management System (transaction STMS), it is possible to perform an automatic, periodic import of all transport requests of the import queue into the quality assurance system, for example once every 24 hours.

*Determine whether this statement is true or false.*

- True
- False

8. You are using a standard three-system landscape with standard transport routes: For increased transport security, it makes sense to define the *Quality Assurance (QA) approval procedure* not only in the quality assurance system but also in the production system.

*Determine whether this statement is true or false.*

- True  
 False

9. You are defining the *Quality Assurance (QA) approval* procedure. Which standard approval steps are selectable?

*Choose the correct answers.*

- A To be approved by department  
 B To be approved by request owner  
 C To be approved by system administration  
 D To be approved by developer  
 E To be approved by task owner

10. You are using a standard three-system landscape with *Quality Assurance (QA) approval procedure*. There are five transport requests in the import queue of the production system. The second transport request has not been approved yet, the others are approved. What happens in general, when you start an import all (import of all transport requests) into the production system?

*Choose the correct answer.*

- A The behavior depends on the setting of *tp parameter HANDLING\_OF\_INACTIVE\_TRANSPORTS*  
 B All approved transport requests are imported  
 C Only the first transport request is being imported  
 D The behavior depends on the setting of the transport strategy

11. During the import of transport requests, several steps are performed. Bring the following steps into the right sequence, regarding the import procedure.

*Arrange these steps into the correct sequence.*

- Handling of dictionary objects (for example, activation)
- Import of other objects (for example, programs)
- Import of dictionary objects (for example, table structures)
- Handling of other objects (for example, generation)

12. Which tool is performing the dictionary import during the import of transport requests?

*Choose the correct answer.*

- A R3trans
- B RDDIMPDP
- C RDDMASGL
- D R3load

13. Which of the following are monitoring tools when importing transport requests?

*Choose the correct answers.*

- A Check for critical objects
- B Import monitor
- C Import history
- D Performance monitor

14. What is the SAP standard procedure to clean up the transport directory?

*Choose the correct answer.*

- A Using command line options of *tp*
- B Scheduling one of the standard batch jobs
- C Switching on the automatic cleanup in TMS
- D Setting the corresponding attribute for transport requests

## Learning Assessment - Answers

1. You are running a development project in a three-system landscape using the *Quality Assurance (QA) approval procedure*. After fixing some errors that have been detected during the test, quality assurance testing in the quality assurance system for the project finally is positive. Which transport requests should you import into the production system?

*Choose the correct answer.*

- A The entire project which includes all approved transport requests assigned to the project
- B Exactly those transport requests that contain only error-free objects
- C If the same objects are contained in multiple transport requests, only import the newest transport requests
- D All transport requests from the import queue of the production system

You are correct! You should import the entire project which causes all approved transport requests assigned to the project to be imported. You should not leave away any approved transport requests of the project or import additional transport requests.

2. You are using a standard three-system landscape: When releasing a transportable transport request from the development system, it is being exported from the development system and immediately imported into the quality assurance system.

*Determine whether this statement is true or false.*

- True
- False

You are correct! Releasing a transportable transport request does not cause an immediate import into the subsequent system(s).

3. You are using client independent transport routes. Which information can be seen in the import queue (transaction STMS) for transport requests that have not imported been yet?

*Choose the correct answers.*

- A The owner
- B The source client
- C The component check result
- D The type
- E The target client

You are correct! The owner, source client, component check result, and type of the transport request can be seen in the import queue. The target client is not set and therefore not visible in case of client independent transport routes.

4. You are using a standard three-system landscape. You did not add or delete any transport requests manually from the import queues. What is the sequence of the transport requests in the import queue of the production system?

*Choose the correct answer.*

- A The sequence in which the transport requests were imported into the quality assurance system
- B The sequence in which the transport requests were released from the development system
- C The sequence of the time stamp of the creation of the transport requests
- D The sequence of the 5-digit number of the transport requests

You are correct! The sequence of the transport requests in the import queue of the production system is the sequence the transport requests were imported into the quality assurance system. .

5. Which of the following are transport strategies of the Transport Management System (TMS)?

*Choose the correct answers.*

- A Queue-Driven Transports, Mass transports
- B Queue-Driven Transports, Single transports
- C Workflow-driven transports
- D Queue-Driven Transports, Project based transports

You are correct! Queue-Driven Transports, Mass transports as well as Queue-Driven Transports, Single transports, and Workflow-driven transports are transport strategies. Queue-Driven Transports, Project based transports is not a transport strategy.

6. What is different when using a transport of copies instead of a workbench request?

*Choose the correct answers.*

- A You can choose any SAP system of the transport domain as the target system
- B You do not need any consolidation route to the target system
- C You can transport together any transportable objects
- D Copies of the objects are being transported, not the originals
- E The objects are being copied from the source system

You are correct! The differences are that you can choose any SAP system of the transport domain as the target system, you do not need any consolidation route to the target system, and you can transport together any transportable objects. There is no difference in copies of the objects are being transported, or objects are being copied from the source system. This is the same as with workbench requests.

7. You are using a standard three-system landscape. By using the Transport Management System (transaction STMS), it is possible to perform an automatic, periodic import of all transport requests of the import queue into the quality assurance system, for example once every 24 hours.

*Determine whether this statement is true or false.*

- True
- False

You are correct! It is possible to perform an automatic, periodic import of all transport requests of the import queue into the quality assurance system.

8. You are using a standard three-system landscape with standard transport routes: For increased transport security, it makes sense to define the *Quality Assurance (QA) approval procedure* not only in the quality assurance system but also in the production system.

*Determine whether this statement is true or false.*

- True
- False

You are correct! It makes no sense to define the *QA approval procedure* in the production system because there are no subsequent systems to be secured.

9. You are defining the *Quality Assurance (QA) approval* procedure. Which standard approval steps are selectable?

Choose the correct answers.

- A To be approved by department
- B To be approved by request owner
- C To be approved by system administration
- D To be approved by developer
- E To be approved by task owner

You are correct! The selectable standard approval steps are: To be approved by department, to be approved by request owner, and to be approved by system administration. To be approved by developer and to be approved by task owner are no standard approval steps.

10. You are using a standard three-system landscape with *Quality Assurance (QA) approval procedure*. There are five transport requests in the import queue of the production system. The second transport request has not been approved yet, the others are approved. What happens in general, when you start an import all (import of all transport requests) into the production system?

Choose the correct answer.

- A The behavior depends on the setting of *tp* parameter  
*HANDLING\_OF\_INACTIVE\_TRANSPORTS*
- B All approved transport requests are imported
- C Only the first transport request is being imported
- D The behavior depends on the setting of the transport strategy

You are correct! The behavior depends on the setting of *tp* parameter *HANDLING\_OF\_INACTIVE\_TRANSPORTS* (*BREAK*, *STOP*, *Skip*). In general, not all approved transport requests are imported and not only the first transport request is being imported. The behavior does not depend on the setting of the transport strategy.

11. During the import of transport requests, several steps are performed. Bring the following steps into the right sequence, regarding the import procedure.

*Arrange these steps into the correct sequence.*

- 2** Handling of dictionary objects (for example, activation)
- 3** Import of other objects (for example, programs)
- 1** Import of dictionary objects (for example, table structures)
- 4** Handling of other objects (for example, generation)

You are correct! The import of the dictionary objects takes place first, then the dictionary objects are handled, followed by the other objects being imported, and finally the other objects are being handled.

12. Which tool is performing the dictionary import during the import of transport requests?

*Choose the correct answer.*

- A R3trans
- B RDDIMPDP
- C RDDMASGL
- D R3load

You are correct! *R3trans* performs the dictionary import. *RDDIMPDP* is the import dispatcher, *RDDMASGL* is responsible for the activation of dictionary objects, and *R3load* is not used in the context of importing transport requests.

13. Which of the following are monitoring tools when importing transport requests?

*Choose the correct answers.*

- A Check for critical objects
- B Import monitor
- C Import history
- D Performance monitor

You are correct! Check for critical objects, import monitor, and import history are monitoring tools. A performance monitor does not exist for importing transport requests.

14. What is the SAP standard procedure to clean up the transport directory?

*Choose the correct answer.*

- A Using command line options of *tp*
- B Scheduling one of the standard batch jobs
- C Switching on the automatic cleanup in TMS
- D Setting the corresponding attribute for transport requests

You are correct! Command line options of *tp* are used (*tp check all* and *tp clearold all*). For this there is no standard batch job, no automatic cleanup in TMS, and no corresponding attribute for transport requests.

## Lesson 1

System Landscape Options

211

### UNIT OBJECTIVES

- Analyze different system landscape options and explain their pros and cons



## System Landscape Options



### LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Analyze different system landscape options and explain their pros and cons

### Introduction

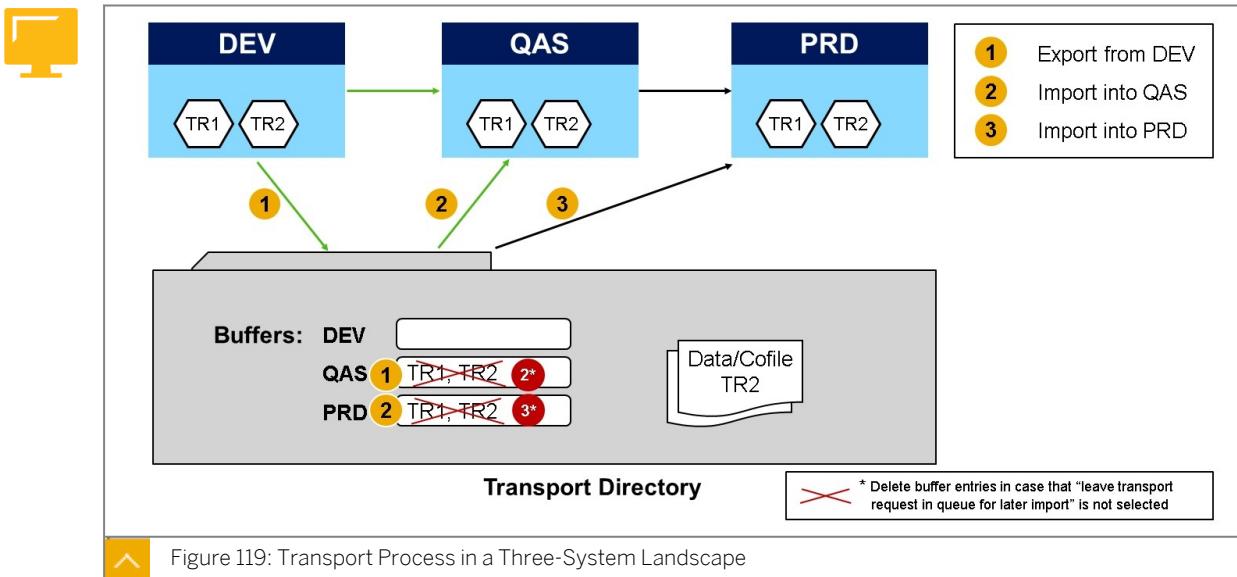
You need to consider which landscape fits best to your change control processes. The processes are critical for organizations that require the delivery of regular software updates (via major or minor releases) while simultaneously providing a secure and stable production environment for end users – especially in a scenario for SAP customers that run development initiatives (innovation) and operation support (maintenance) in parallel.

This lesson provides an overview of the main system landscape options that are available together with their pros and cons. It starts with the basic three-system landscape and then moves forward to more complex scenarios.

A three-system-landscape consists of a development system (DEV), a quality assurance system (QAS), and a productive system (PRD).

- The development system DEV is needed to implement changes.
- The quality assurance system QAS is needed to perform functional tests and integration tests with production-like data. QAS is also needed to test the import of the transport requests. Without a quality assurance system, the import of transports cannot be tested before it is executed in production.

The following figure recaps the transport process in a three-system-landscape with a *consolidation* route (indicated by a green arrow in the following figures) between the development system and the quality assurance system and a *delivery* route (indicated by a black arrow in the following figures) between the quality assurance system and the production system:



1. In the first step, the first transport request (TR1) is released. As a consequence, development objects that are locked are released and the transport request is added to the import queue of the quality assurance system.

**Note:**

The import queue has two functions. The first function is to contain all transport requests that have been released, and the second is to remember the sequence in which the transport requests have been released.

Next the second transport request (TR2) is released as well.

2. In the second step, the transport requests contained in the QAS import queue are imported into the quality assurance system. System QAS is used for first tests with production-like data.  
The transport requests that are imported into the QAS system are also added to the import queue of the productive system PRD. This guarantees that the transport requests and their import sequence to the quality assurance system QAS are remembered. When you use the *Import Project* or the *Import ALL* strategy, the transport requests are also deleted from the import queue of the quality assurance system.
3. In the third step, the transport requests listed in the import buffer of the productive system are imported into production. When you use the *Import Project* or the *Import ALL* strategy, the transport requests are also deleted from the import queue of the production system.

### Three-System-Landscapes

Using the *Single Transport* strategy in a three-system landscape leads to the following issues and risks:

- Transport requests are imported in an individual sequence
- Risk of forgotten transports
- Risk of version downgrades by transport sequence violations

These risks are discussed in more detail in the following sections.



Note:

SAP recommends that you use the *Import Project* or *Import ALL* strategy.



- **Problem:** Importing Transport Requests in an individual sequence

- Risk of forgotten transports
- Risk of version downgrades by transport sequence violation

- **Solution:** Use *Import Project* or *Import ALL* strategy

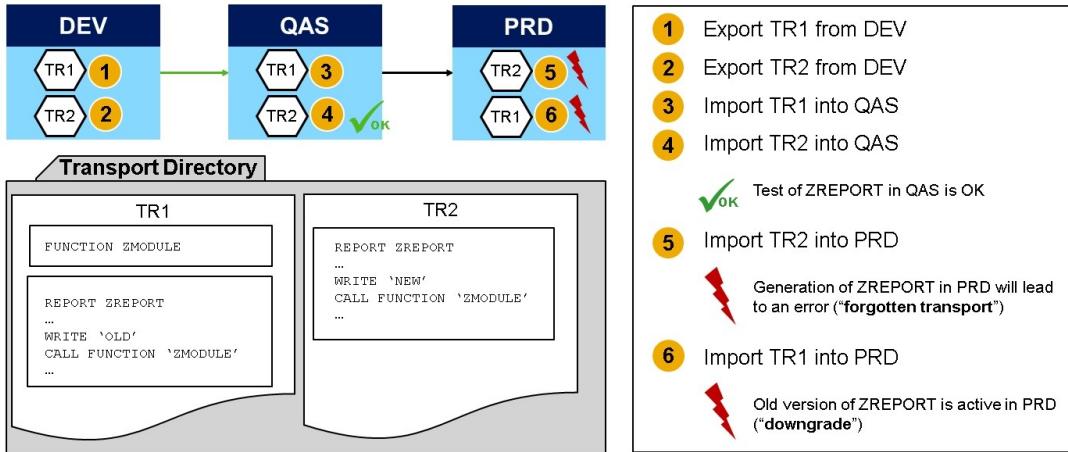


Figure 120: Transport Risks

Transport request TR1 contains a report, ZREPORT, which calls the function module, ZMODULE which has also been created in transport request TR1. This transport request is released in the development system (step 1 in the figure above). Then transport request TR2 changes the report ZREPORT but leaves the function module ZMODULE untouched. TR2 is then released as well (step 2).

In QAS, TR1 and TR2 are imported one after the other (steps 3,4) and then tested together. The test is OK.

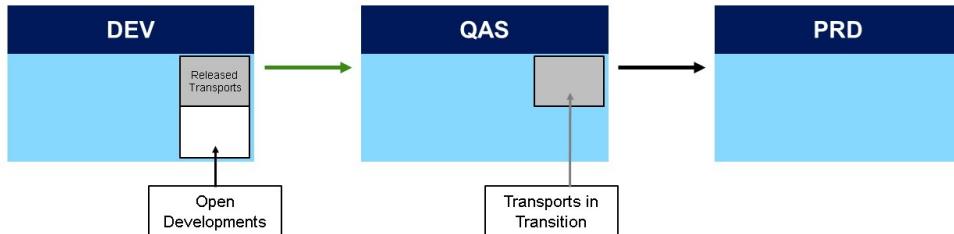
However, when now only TR2 is imported into PRD (step 5), the generation of ZREPORT fails, because TR1 was forgotten and function module ZMODULE therefore does not exist in PRD and cannot be called.

If then TR1 (without TR2) is also imported into PRD (step 6), the test in PRD fails again because now the old version of ZREPORT (version of TR1) has overwritten the newer version (version of TR2).

The following figure explains some reasons for these inconsistencies in a more general way:



- **Inconsistencies between DEV, QAS and PRD:**
  - Open developments in DEV which have not yet been exported
  - Transports in QAS which have not yet been imported to PRD (“transports in transition”)



- **Reasons:**
  - Transport sequence violations
  - Transports which were not transported completely into all systems
  - Transports stay in the import buffer of QAS or PRD too long
  - Several parallel projects with different project timelines
- **Solution:**
  - Forward all transport requests that are released in the development system quickly into the test and production system

Figure 121: Inconsistencies in the Transport Landscape – Reasons

The development system DEV contains all developments (both open developments and developments that have been recorded in released transport requests). The quality assurance system QAS contains transports that have already been imported into PRD, and also transports that have not yet been imported into PRD (labelled as “transports in transition” in the figure above). Therefore, all three systems are on different software levels.

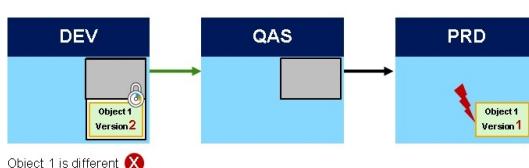


#### Caution:

These different software levels in a three-system-landscape cause risks. One big risk is that programs that have been tested successfully in QAS might fail in PRD.



- **Version Inconsistencies:**
  - Error in PRD, object 1 must be maintained
  - Object 1 is already changed in DEV and cannot be maintained



- **Cross Reference Errors:**

- Error in PRD, object 1 must be maintained
- Object 1 uses object 2, object 2 was changed by project
- Test in QAS is OK
- Error in PRD because object 2 is still in old version



- **Solution:**

- Comparing active DEV object version with active PRD object version
- Intermediate switch DEV object version for fix

- **Solution:**

- Additional pre-production System PRE with software / configuration state of the PRD system

Figure 122: Inconsistencies During Bug Fixing – Examples

Objects that have been changed in a project that is currently in transition (either in open transport requests or in already released transport requests) cannot be maintained anymore during bug-fixing.

As soon as a change to a repository object is performed in the development system, this repository object is locked in the transport request (see the left part of the figure above). The lock is deleted when the transport request is released from the development system.

You can compare the object versions in DEV and PRD before starting a maintenance task. If the versions are different, then you have to temporarily switch back an old object version from the version history in the development system. This object version is corrected and transported to production. In the next step, the correction is made again with the latest object version.



**Note:**

This procedure is not possible for BW objects, because they do not have a version history.

The procedure to maintain two different versions of one object via versions, and to transport these versions through the system landscape is error-prone. Alternatively, the developer can search for objects in transition with transaction SREPO which – however – requires an RFC destination between the SAP system that you want to compare the production system with and the production system itself.

The right part of the figure above shows the reasons for *cross reference* errors. Here the transport request for object 2 has already been released in the development system and therefore the lock has already been deleted – object 2 belongs to a transport in transition.

In this example, object 1 was not changed by the project. But object 1 uses object 2, and – as stated above – object 2 was changed (from version 1 to version 2).

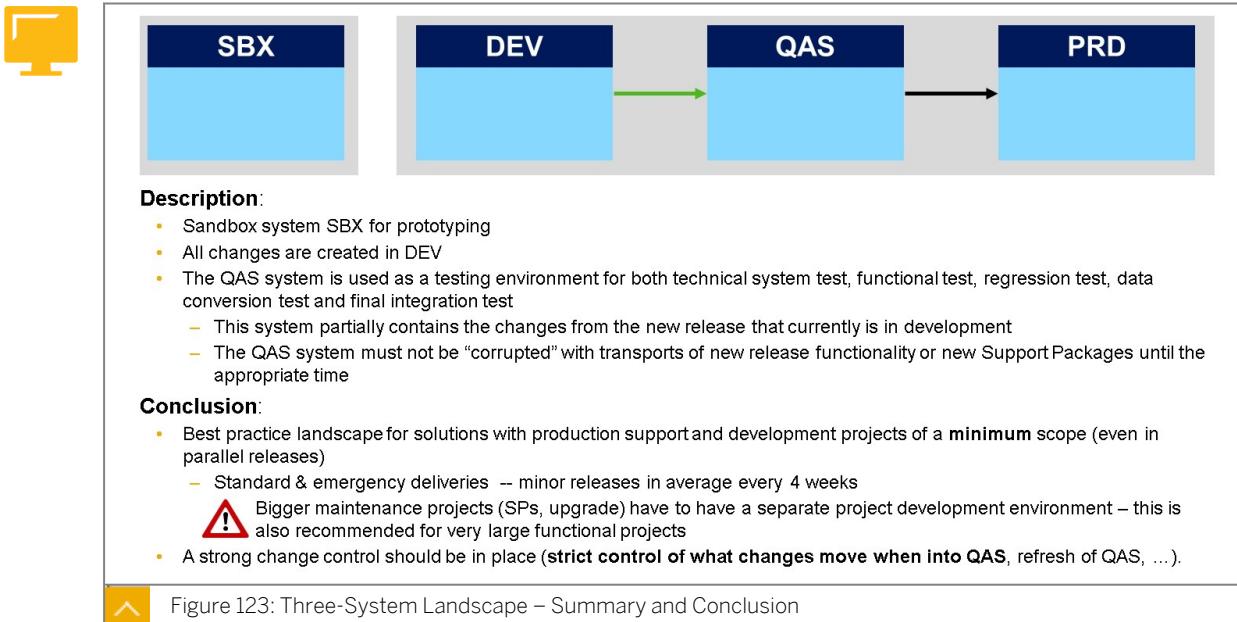
Initially, object 1 worked, but now object 1 has to be corrected. The new version of object 1 still works with the new version of object 2, but the new version of object 1 does not work with the old version of object 2 and this is only detected in PRD.

The system landscape topology, which can be used to detect cross-reference inconsistencies is a pre-production system. This fourth – pre-production – system is brought between the QAS and the PRD. If this pre-production system has the current software and configuration state of production, a cross-reference error can already be detected during tests in pre-production.



**Note:**

You can also use the report `/SDF/CMO_TR_CHECK` to check for cross system issues before transporting into the production environment.



Assume an SAP ECC system landscape on an old release serving one or several countries. The major development cycle and the rollout are complete. Ongoing changes are limited to Support Packages, operating and database patches. SAP configuration changes are required to support minor business requirements.

In this case, a three-system-landscape might be sufficient: The maintenance transports are bundled into packages, so that they can be tested together.

This scenario works well if the number of urgent corrections is low. Non-urgent corrections should be released later so that QAS stays close to PRD for the test of urgent corrections.

In this scenario, all changes are created in the development system DEV (maybe after prototyping in the sandbox system SBX).

The QAS system is used as a testing environment for integration testing and data conversion testing. This system partially contains the changes from the new release that currently is in development. The QAS system is the environment for the final integration test (once the scope of the release has been fixed – release test), regression test and technical system test. It should represent the status in the PRD systems in the following way:

- *Infrastructure:* the setup of the QAS system should mimic PRD as closely as possible and it should be a full copy of the production system. This is the prerequisite for realistic volume tests.
- *Transports:* The QAS system must not be “corrupted” with new release functionality or new maintenance packages until the appropriate time (ideally as close to go-live as possible).
- *Data:* The QAS system should be regularly refreshed from PRD, if possible.

This landscape is the best practice landscape for solutions with production support and development projects of a **minimum** scope (even in parallel releases).

Recommendations to make best use of this landscape:

- Strict change management for production support and project development, and/or same development team for both types of changes.

- A strong change control should be in place (strict control of what changes move when into QAS, refresh of QAS, ...).

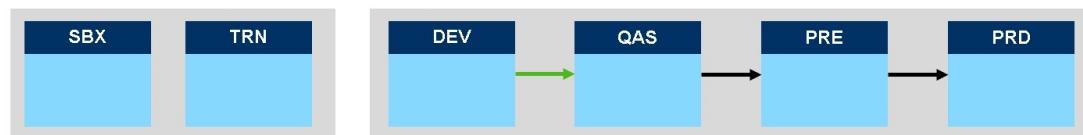


#### Note:

Larger maintenance projects (Support Packages, upgrades) have to have a separate project development environment (see the section dealing with the dual track landscape later in this lesson). This is also recommended for very large functional projects.

## Four-System-Landscapes

In times of invasive changes (for example, during implementation of Support Packages), it might be useful to introduce a fourth system to support the maintenance of the production system.



#### Description:

- The pre-production system PRE has the current software / configuration state of the production system
- The next urgent correction or project cycle can be isolated in PRE
- Cross reference errors can be deleted before they do into production
- Optional: Training system TRN for training end users on the new release

#### Conclusion:

- Best practice landscape for solutions with production support and development projects of a **medium scope** (even in parallel releases)
  - It allows for a staged testing
  - Bigger maintenance projects (SPs, upgrade) have to have a separate project development environment – this is also recommended for very large functional projects
- Recommendations to make best use of this landscape:
  - Strict change management for production support and project development, and/or same development team for both types of changes
  - A strong change control should be in place (**strict control of what changes move when into PRE**, refresh of PRE, ...).

Figure 124: Four-System Landscape – The Basic Idea

- All changes are created in the development system DEV. Any invasive changes, such as Support Packages, are tested in the sandbox system SBX before they are implemented in DEV.
- The QAS system is used as a testing environment for integration testing and data conversion testing. This system partially contains the changes from the new release that currently is in development.
- The pre-production system PRE is the environment for the final integration test (once the scope of the release has been fixed – release test), regression test, user acceptance test and technical system test. It should represent the status in the PRD systems with respect to:
  - Infrastructure*: the setup of the PRE system should mimic PRD as closely as possible and it should be a full copy of the production system. This is the prerequisite for realistic volume tests.

- *Transports*: The PRE system must not be “corrupted” with new release functionality or new maintenance packages until the appropriate time (ideally as close to go-live as possible).
- *Data*: The PRE system should be regularly refreshed from PRD, if possible.

During the implementation of Support Packages (or other minor releases), in DEV only urgent changes, standard changes and smaller non-invasive functional enhancements are developed. Here, PRE is required, for example, for testing of urgent changes during the time periods while the Support Package org the minor release is “blocking” QAS.

It is important to move the Support Packages fast through the landscape in order to limit the time period of inconsistent releases in the landscape. A reduction of the transition time for the Support Packages can be achieved by thorough testing in SBX, by a clear definition of the test scope (reflecting the changes of the concrete Support Package), by test automation and by ensured availability of test resources



Note:

If the number of changes is very small, a 3-system landscape might be sufficient.

The required system sizes are as follows:

- The DEV and the QAS system may be of small data volume, no high availability setup, small CPU and small RAM.
- The PRE system may have the full data volume, a reduced high availability setup, and reduced CPU and RAM.
- The PRD system of course should have full data volume, a full high availability setup, as well as full CPU and RAM.

The four-system landscape with the pre-production system is the best practice landscape for solutions with production support and development projects of a **medium** scope (even in parallel releases) – as it allows for a staged testing. Recommendations to make best use of this landscape are :

- Strict change management for production support and project development, and/or same development team for both types of changes.
- A strong change control should be in place (strict control of what changes move when into PRE, refresh of PRE, ...).



Note:

Larger maintenance projects have to have a separate project development environment (see the section dealing with dual system landscape below). This is also recommended for very large functional projects.

### Variants of the Four-System Landscape

Besides the four-system landscape with a pre-production system (as discussed above), other variants of a four-system landscape are possible as well. Two of them are presented in the following. There is no ideal system landscape, all have their pros and cons.

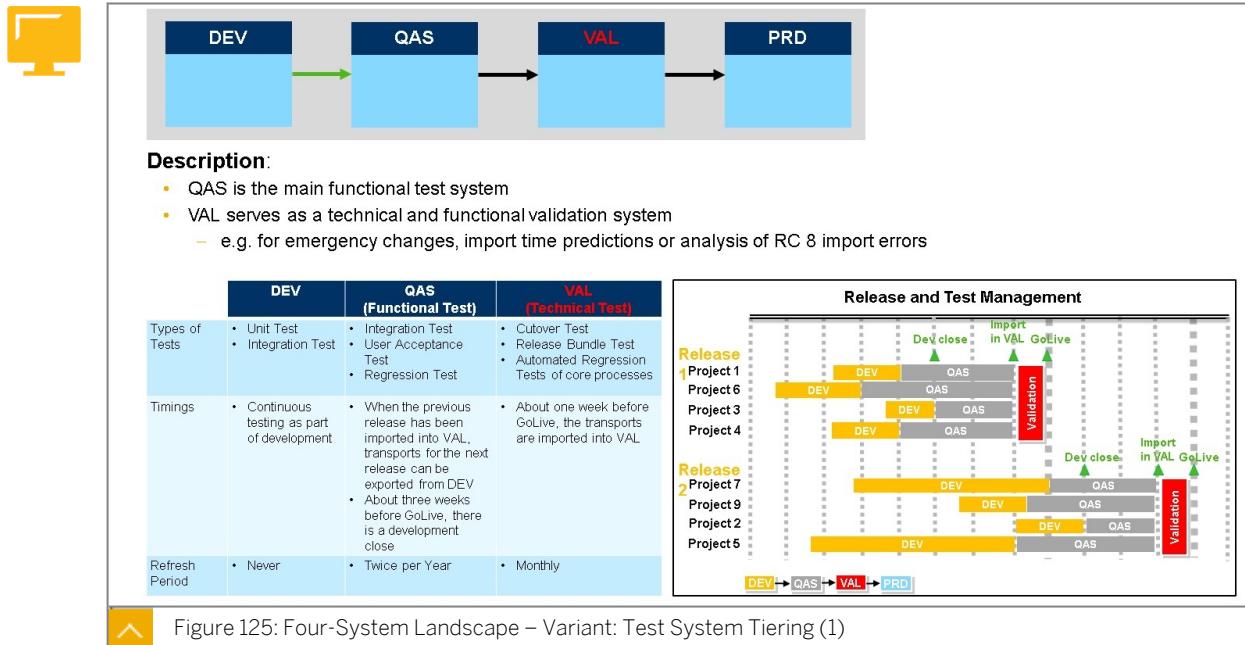


Figure 125: Four-System Landscape – Variant: Test System Tiering (1)

In the scenario of the figure above, QAS is the main functional test system. All functional tests (integration tests, user acceptance tests, regression tests) take place in QAS. As a consequence, only one release can be tested in QAS (for the next go-live). Future releases can already be developed and tested in DEV.



#### Hint:

A test client and core interfaces can be set up in DEV to allow early integration testing.

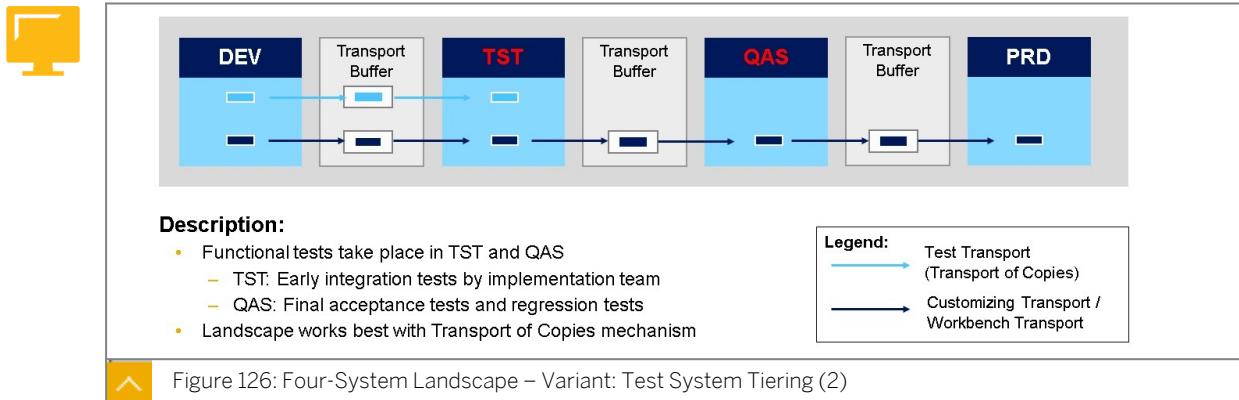
The validation system VAL serves as a technical and functional validation system for the following purposes:

- Test of emergency changes when the next release is already sitting in QAS
- Technical cutover tests and import time predictions, in particular for upgrades or big release imports
- Technical validation of the release bundle if certain transports must be taken out of the release (last minute scope changes), for example if they do not meet the test criteria

The technical validation can comprise the analysis of RC8 during the import and an extended syntax check (ATC) of the imported code. The functional validation can comprise a number of automated test scripts for the core processes and some manual spot checks as needed.

In addition, the VAL system allows a better release management, as multiple projects can be validated together allowing a common go-live.

Another variant of the four-system landscape is shown in the following figure. Here, functional tests are not restricted to the second system of the system landscape but takes place in both the second and the third system:



In this scenario, functional tests take place in TST and QAS. So interfaces and appropriate test data must be provided for both systems. This is sometimes a problem when the SAP solution is heavily integrated with legacy systems which have only one test tier.

Typically early integration tests are done by the implementation teams and vendors in TST. Final integration tests, user acceptance tests and regression tests are done in QAS in the final preparation phase of a release.

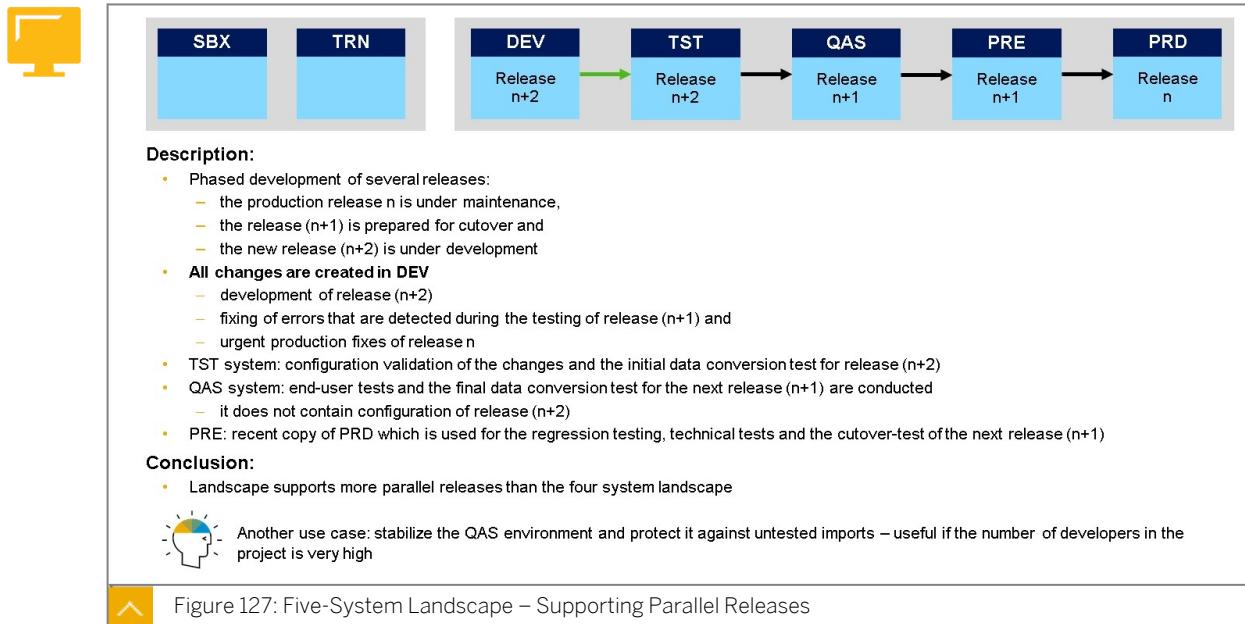
This landscape allows to test for future releases in TST, even though they are not yet assigned to the next upcoming release. The transport bundle for the next release must be assembled only before the final preparation phase starts.

This landscape works best with the *Transport of Copies* mechanism which – for example – is used in SAP Solution Manager *Change Request Management*.

In this option, QAS partly takes over the role of the technical cutover system VAL from the variant above. However there are some restrictions:

- Transports are already imported into QAS, before the transport bundle is finalized.
- Emergency transports cannot be tested in a clean environment while the release is tested in QAS.
- Repeated dress rehearsals and cutover optimizations are not possible.

## Advanced Landscapes to Support Parallel Releases



A five-system landscape shown in the figure above allows a phased development of several releases: the production release n is under maintenance, the release (n+1) is prepared for cutover and the new release (n+2) is under development.

- In this scenario, **all** changes are created in DEV (both development of release (n+2) and fixing of errors that are detected during the testing of release (n+1) and urgent production fixes of release n).
- The TST system is used for configuration validation of the changes (mostly developments of release n+2) and the initial data conversion test for release (n+2).
- In the quality assurance system QAS, the end-user tests and the final data conversion test for the next release (n+1) are conducted, it does not contain configuration of release (n+2).
- The Pre-production system PRE is a recent copy of the production system and is used for the regression testing, technical tests and the cutover-test of the next release (release (n +1)).

A single track landscape in principle does work. But it comes with the following limitations and risks:

- Challenges and risks arise mainly because all types of changes need to be managed in a single development system and in a single transport track (in a best practices environment, there are 5 types of changes – emergencies, minor releases, major releases, standard changes, and SAP maintenance).
- There are conflicts that will need to be managed explicitly by respective processes. These conflicts will lead to limitation in either the project development or the creation of corrections (one has to wait).
- Limited flexibility for developing new functions.
- Limited test capabilities for testing new functions (time and environment).

- Risk for urgent corrections (the release or development status of the DEV system is different to the status of the production system; potentially, in DEV, there are new developments active, the objects that need to be corrected have been changed by a new development project).
  - Urgent corrections potentially take longer to be implemented.
  - The test environment for urgent corrections potentially is "far from production" (only the pre-production system reflects the status of production).

This is where a dual-system transport landscape comes into play (see the following figure). This landscape, also known as “N+1 landscape”, was designed for customers who need to continuously release a significant number of software updates, regularly, and provide a secure and stable production end user environment.

The main principle of this design is the decoupling of production support and project development:

- In the *Production Support* track, all urgent changes are created as well as standard changes and “minor releases” consisting of uncritical changes.
- In the *Project Development* track, the development of larger new functionality is done (such as “major releases” and SAP Support Packages)

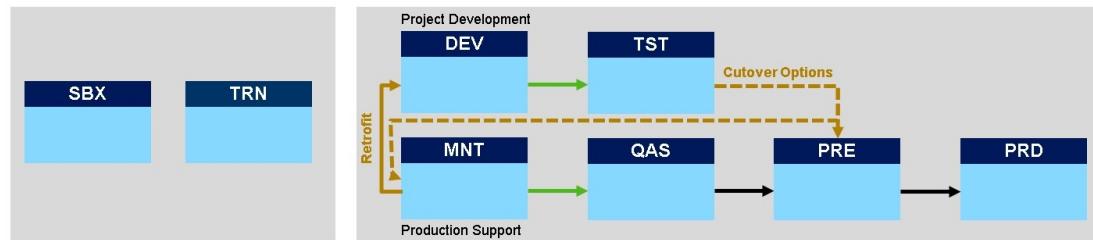
Only at well defined points in time, the new developments are taken over into the *Production Support* track. This is done via an explicit **cutover** after a sufficient testing period in the TST system.

The new release needs to be accepted by the business as well as by the owner of the *Production Support* track.

Changes done in the *Production Support* track are retrofitted into the *Project Development* track (in a timely manner).

The required system sizes are as follows:

- The DEV, MNT and the QAS system may be of small data volume, no high availability setup, small CPU and small RAM.
- The TST and the PRE system may have the full data volume, a reduced high availability setup, and reduced CPU and RAM.
- The PRD system of course should have full data volume, a full high availability setup, as well as full CPU and RAM.

**Description:**

- Decoupling of production support and project development:
  - In the production support track, all Urgent Changes are created as well as Standard Changes and "Minor Releases" consisting of uncritical changes
  - In the project development track, the development of larger new functionality is done – Major Releases and SAP Support Packages
- Only at well defined points in time, the new developments are taken over into the Production Support track ("Cutover")
- Changes done in the Production Support track are retrofitted into the project track (in a timely manner)

**Conclusion:**

- Best practice in case of **large** implementation projects (either maintenance or functional projects) in parallel to production support
- Project changes do not affect the production support landscape – therefore it allows for save and fast production support at all times

Figure 128: Dual System Landscape – The Basic Idea

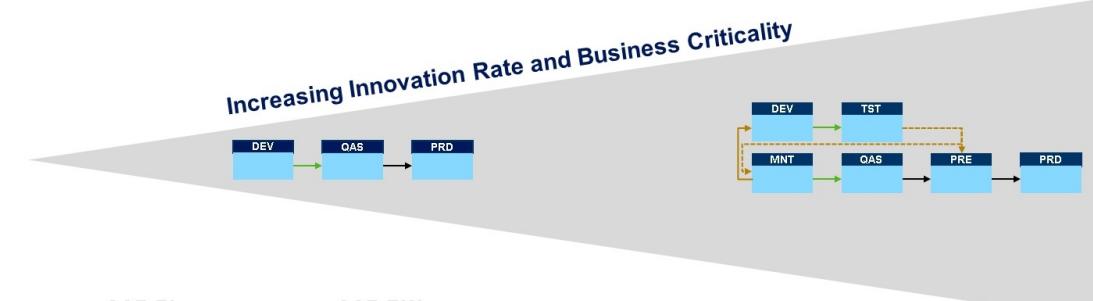
This landscape or its variants are best practice in case of **large** implementation projects (either maintenance or functional projects) in parallel to production support – project changes do not affect the production support landscape. Therefore it allows for save and fast production support at all times.

Note:

More than one *Project Development* track is usually not recommended.



**Only for your most innovative and business-critical systems**

**Deciding factors:**

- Major implementation projects (Major Releases) or upgrade projects are in the pipeline – these projects typically have a long runtime ( $\geq 3$  months)
- Production support and projects are changing the same business processes – that means a significant number of conflicts (parallel changes of the same objects) are expected
- There is a high risk awareness in the company or for the particular solution
- Separated responsibilities of maintenance and implementation in the organization

Figure 129: When to Use the Dual Landscape?

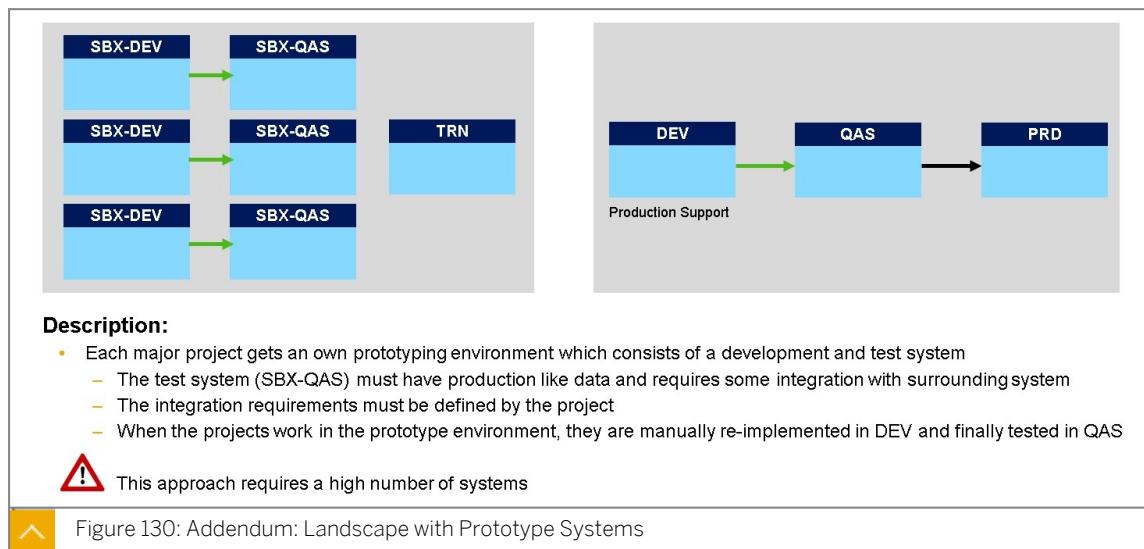
The figure above shows some deciding factors for a dual track transport landscape.

**Note:**

Note that if Support Packages are included in the project in a dual track transport landscape, then all SAP Notes that must be implemented in production must be handled as conflicts.

### Addendum: Landscape with Prototype Systems

The following figure finally shows a different approach how to handle multiple projects for the same production system:



### LESSON SUMMARY

You should now be able to:

- Analyze different system landscape options and explain their pros and cons

# Learning Assessment

1. Which landscape is best practice for solutions with production support and development projects of a **medium** scope?

*Choose the correct answer.*

- A Three-System Landscape
- B Four-System Landscape
- C Five-System Landscape

# Learning Assessment - Answers

1. Which landscape is best practice for solutions with production support and development projects of a **medium** scope?

*Choose the correct answer.*

- A Three-System Landscape
- B Four-System Landscape
- C Five-System Landscape

You are correct. The four-system landscape is best practice for solutions with production support and development projects of a **medium** scope. Three-system landscapes are best suited for solutions with production support and development projects of a **minimum** scope, five-system landscapes are used in **large** implementation projects parallel to production support.

## Lesson 1

Client Copy and Client Transport Tools

229

## Lesson 2

Client Compare and Client Maintenance Tools

259

## UNIT OBJECTIVES

- List client copy and client transport tools
- Outline the idea of copies based on transport requests
- Create a new client with the help of a local client copy
- Illustrate the use of remote client copies, client transports and client deletions
- Describe the monitoring of client copies
- Perform a Client Comparison
- Outline the idea of client adjustments with respect to Customizing
- Describe how to compare two repositories



# Client Copy and Client Transport Tools

## LESSON OVERVIEW

In this lesson, you will get an overview of the client tools available in an SAP system landscape. You will learn the processes of client copy and client transport and you will explore SAP system requirements for processing a client copy.

## Business Example

For operating an AS ABAP based SAP system, several clients are required. Your job as an SAP system administrator is to create and set up clients and then manage all the clients across the SAP system landscape.



## LESSON OBJECTIVES

After completing this lesson, you will be able to:

- List client copy and client transport tools
- Outline the idea of copies based on transport requests
- Create a new client with the help of a local client copy
- Illustrate the use of remote client copies, client transports and client deletions
- Describe the monitoring of client copies

## Using Client Copy and Client Transport Tools

SAP offers client copy and client transport tools.

You can use the client copy to create, for example, the following clients:

- New clients from the SAP reference client 000, during initial implementation of an SAP system
- Training clients
- Demonstration clients
- Test clients
- Production clients

The target client can be copied from a source client that is located either in the same or in another system.

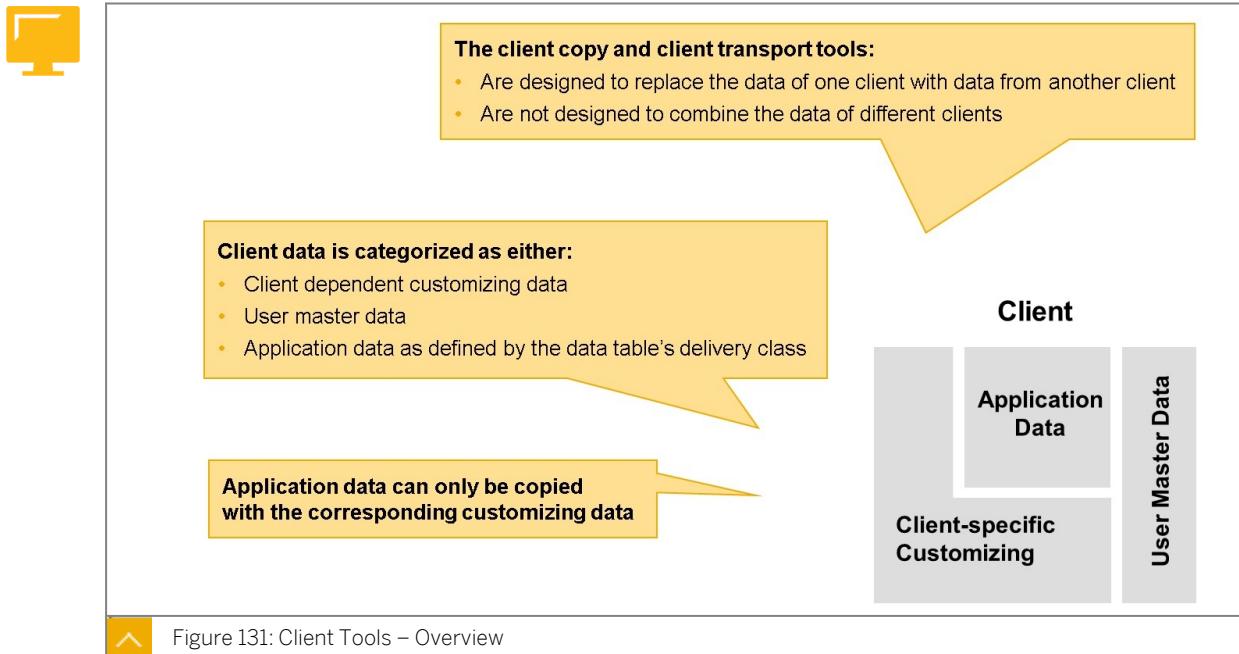


Figure 131: Client Tools – Overview

(Copy) profiles define the tables to be copied through the delivery class of the table. Special data such as user master data can be selected by specifying criteria.



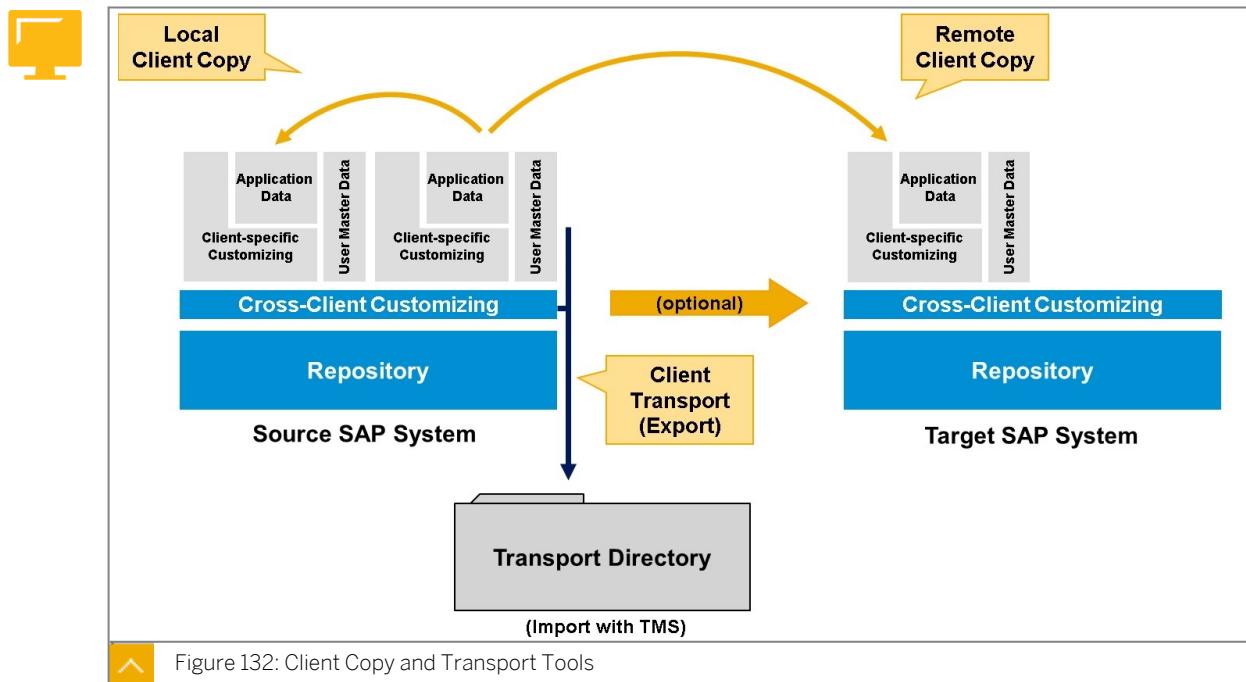
Note:  
These *delivery classes* are explained later in this lesson.

With the client copy tools, you can copy selected parts of an existing client into another client, for example, the user master data with the copy profile *SAP\_USER*.

To construct a client, use the (copy) profile *SAP\_CUST*, which copies client-specific customizing only, to copy client 000, because the consistency of the application data in the SAP delivery client 000 can't be guaranteed.



Note:  
You need to create a new client in transaction *scc4 (Client Administration)*, before you can perform the client copy.



The client copy tools can copy the following components of the source client into the target client:

- *User master data*: user master data is only deleted in the target system if a (client copy) profile with user master data is copied. Authorization profiles and roles belong to customizing and are, therefore, always copied together with the customizing. Copying users without authorization profiles is problematic. The (client copy) profile `SAP_USER` therefore also contains authorization profiles and roles.
- *Customizing*: many client copy profiles contain customizing. Customizing data is generally stored in tables with delivery classes C,G,E, and S.
- *Cross-client customizing*: when performing a remote client copy or a client transport, also cross-client customizing can be copied.



**Caution:**

Inconsistencies can arise in the target system after copying cross-client tables. Copying cross-client customizing can only be used to create a new SAP system because existing clients can be corrupted by changes in their context.

- *Master/transaction (application) data*: select this option, for example, if you want to set up a test client from the production client.

**Note:**

Application data depends on customizing data, so it can only exist consistently together with it. Existing application data is always deleted from the target client, except for copies with SAP\_USER and SAP\_UONL. Application data is generally stored in tables with delivery class A.

If you have maintained a logical system in each of the source and target clients (transaction BD54) and have assigned it to the client (transaction SCC4), the logical system names in the target client application data are automatically converted after copying the application data (for example, with copy profile SAP\_ALL).

Copying clients requires a large amount of system resources. To avoid premature termination due to bottlenecks, ensure that enough resources are available by:

- *Database storage space*: Perform a *test run* before copying a client. The log contains the total copied and deleted data (in MB). Storage requirements can only be estimated, because space already allocated, but not yet used, is not taken into account.
- *Runtime*: Depending on the volume of application data, copying a client can take several hours or even days. Active users or background processes in clients other than the source or target clients can extend this time even more. For example, locks in a third client in the same system can delay the processing of objects. In principle, work in the system is technically possible during the client copy. However, it is strongly recommended that you do not do this (or only in exceptional cases), since this can lead to inconsistencies. If you use parallel processes for client copy, dialog processes are used even if the client copy job is scheduled in the background.

**Hint:**

The standard timeout value is usually sufficient. If the database is used by additional processes, it can be advisable to increase the profile parameter.

- *System load*: Copying or transporting a client can take a long time because large amounts of data are moved. One or more dialog processes are occupied for this time. The database interface is heavily used.
- *Protecting clients against user log ons*: You must ensure that no users log on to the system during the copy. For technical reasons, the source client (optional) and target client (mandatory) are locked.

**Note:**

The source client lock can be removed in the *expert settings*. For the new client copy tools available as of SAP\_BASIS 754, it can be removed on the selection screen. SAP recommends that you unlock the source client in exceptional situations only (when constructing sandbox or test systems).

The logon lock also applies to HTTP and not just to SAP GUI log ons. The only way to access a locked client is to use RFC. Users who have been logged on to the target client before the start of the copy cannot be locked automatically, so you must ensure that they leave the system. The source and target clients should both be additionally protected by a

system message (SM02). Monitor compliance in both clients (for example, in transaction SM04). You should not work in the source client either during the copy.

- *Clean Up:* If you want to use SAP Query in the new client, you must do some post processing.
- *Constraints:* You can't access archived data in the target client if the target client number is not the same as the source client number.
- *Change documents (tables CDHDR, PCDHDR, CDPOS, and PCDPOS):* Change documents from user administration and logs from the generic log repository (application log tables BAL\*) are not copied.
- *Effect on Number Ranges:* If customizing and application data is copied, the number ranges are copied with the data because they are required by the application data. If only customizing data is copied, the number ranges are reset because the application data that references them is deleted and then reconstructed.



#### Note:

The amount of data and, therefore, the memory required and copy time for productive clients can be considerable. In this case you should not copy application data, you might think about creating the required test data, for example, with extended Computer Aided Test Tools (eCATTs).

### Authorizations

To copy or transport clients, you need the appropriate authorizations.

Table 3: General Authorization Objects for Client Copy

Authorization Objects	Action
S_TABU_CLI	Maintain cross-client tables
S_TABU_DIS	Maintain system tables
S_CLNT_IMP	Import data in client copy
S_DATASET	Access the file system

Table 4: Authorizations for Copying User Profiles and User Master Records

Authorization Objects	Action
S_USER_AGR	Copy roles
S_USER_PRO	Copy authorization profiles
S_USER_GRP	Copy user master records

Table 5: Authorizations for Transporting Clients

Authorization Object	Action
S_TRANSPRT with TTYPE 'CLCP' and ACTVT '01'	Create object lists for client transports and copying into another client

Table 6: Authorizations for Remote Copy (Mandatory Authorization for RFC User in Source Client)

Authorization	Action
S_TABU_RFC	Remote access to tables in target system

### Changes with SAP\_BASIS 754



Note:

This section is valid only for SAP systems based on SAP\_BASIS 754 or higher.

Starting with SAP\_BASIS 754, the client copy procedures were changed and enhanced. See SAP Note [2962811 – New Client Copy Tool: General Information](#) for details.

The new Client Copy Tool comes with new transaction codes and APIs (RFC). Furthermore the tool now offers task lists for execution in STC01.

New authorization objects S\_CLNT\_CPY (*Client Copy Process Execution*, activity 60 for client copy, activity 03 for client comparison) and S\_CLNT\_EXI (*Client Copy Exit Execution in RFC* for exits in remote systems) and are available.

The figure “Client Copy Tools – Overview” shows both tools that have been re-factored as successors of old transactions and new tools. These tools can be accessed either using transaction codes or with the help of task lists (transaction STC01).



Process/Tasks	Old Transaction	New Transaction	Task List	Released with	Description
Local Client Copy	SCCL	SCCLN	SAP_CLIENT_COPY_LOCAL	SAP_BASIS 754 SP 0	Copy a client within the same SAP system
Remote Client Copy	SCC9	SCC9N	SAP_CLIENT_COPY_REMOTE	SAP_BASIS 754 SP 1	Copy a client between two SAP systems using an RFC destination
Client Deletion	SCC5	SCC5N	SAP_CLIENT_DELETION	SAP_BASIS 754 SP 0	Delete a local client
Client Export	SCC8	SCC8N	SAP_CLIENT_EXPORT	SAP_BASIS 754 SP 2	Export a client to a transport request
Client Import	SCC7	SCC7N	SAP_CLIENT_IMPORT_POSTPROCE SSING	SAP_BASIS 754 SP 2	Run client import post processing after import
Client Copy Log	SCC3	SCC3 (enhanced)	n/a	SAP_BASIS 754 SP 1	Displays logs of client copies
Client Copy - Special Selection	SCC1	SCC1N	SAP_CLIENT_COPY_BY_TRANSPORT	SAP_BASIS 755 SP 1	Copies the content of transport requests / tasks to a different client in the same SAP system
Client/Container Comparison	N/A	SCC_COMPARE	SAP_CLIENT_COMPARISON	SAP_BASIS 755 SP 0	Compares two clients in the same or in different SAP systems (replaces SCC_COMP_LOCAL and SCC_COMP_REMOTE)
Client Size Determination	N/A	SCC_CLIENT_SIZE	SAP_CLIENT_SIZE	SAP_BASIS 754 SP 0, tasklist available with SAP_BASIS 755 SP 1	Determines the size of a client or of specified tables – only approximation if more than one client is filled



Figure 133: Client Copy Tools – Overview

The Client Copy Tool contains the following improvements:

- *Improved performance:* Up to 10 times faster than before by using native SAP HANA DB capabilities. The Remote Client Copy tool is up to 5 times faster than before.
- *Improved security and reduced manual effort:* User SAP\* is no longer needed to perform client copies so that system does not need to be restarted. Task lists are available for automation.
- *Improved stability:* The Client Copy application exits now run in isolated environments. Failed exits and tables are stored in the log.
- *Improved log:* The Client Copy log has been transformed from a file-based to a table-based UI with additional information and better persistence. The tool offers multiple tabs for different sections of the log, such as *Header Data*, *Processed Tables*, *Application Exit Messages*, and *Runtime Measurements*.

 Note:

Client copies need no longer be executed from the target client. Instead, SAP recommends that you run the Client Copy tool from a third client that is not affected by the tool. For example, when copying from client 100 to 200, you could run the tool in client 000.

The client copy always locks the target client. By default, the source client is also locked. You may choose **not** to lock the source client at the risk of concurrent write access to the source client resulting in inconsistent copies. Note that a client lock now not only prevents new log ons using SAP GUI, but also HTTP access.

You can search for task lists using transaction STC01. You have two options:

- Enter a task list and choose *Display Task List*.
- Open the drop-down list and select *Task List*, *Description*, or *Task List Group*.

Once you have completed one of these actions, choose *Search*.

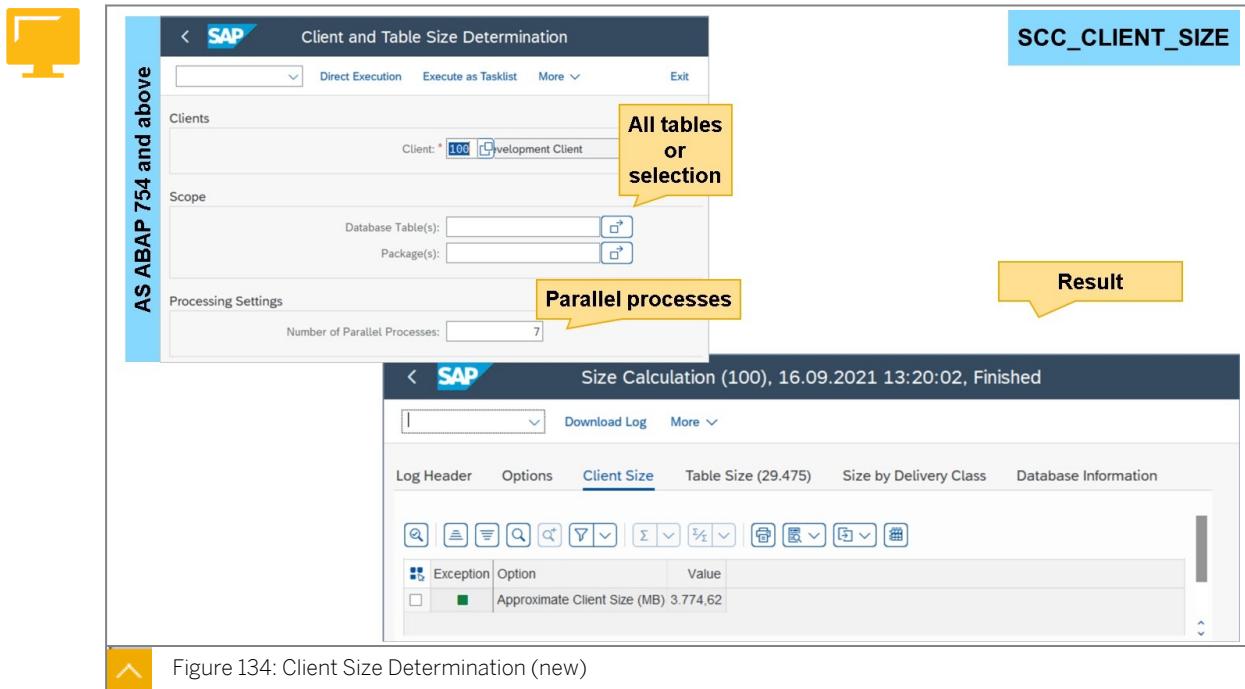
 Note:

To avoid a long process duration when copying large amounts of data, execute the client copy as task list.

Before copying clients or tables from one system to another, you can determine their size and how much space is required for them in the memory. To do this, use transaction SCC\_CLIENT\_SIZE.

For this, proceed as follows:

1. Call transaction SCC\_CLIENT\_SIZE.
2. Enter a client and the number of processes that are to run in parallel.
3. To select a single table, enter a client-specific table or use the *search* function.
4. Execute the transaction.



## Testing Customizing Transport Requests

Transport requests for customizing are usually created in the development system. They are then released in the development system and imported into the quality assurance system. It can be a good idea to additionally perform some pre-testing in a separate client of the development system. This should be done before releasing the transport request.

Before releasing a customizing transport request, perform a unit test to:

- Test the functionality of the transport request.
- Verify that the contents of the transport request is complete.

Maintaining a separate client for testing allows both true unit testing and the maintenance of test data without the risk of creating customizing-dependent data.

Transaction SCC1 copies changes from one client to another based on either:

- A task
- A transport request
- A transport request and its tasks

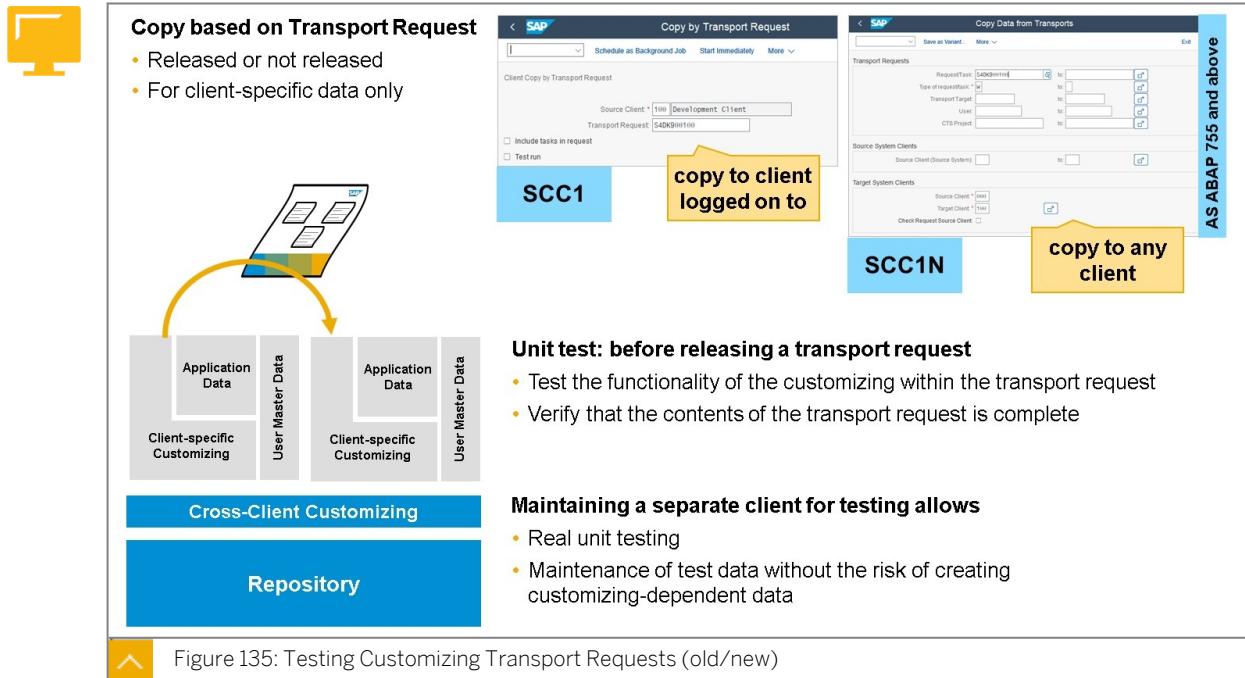


Figure 135: Testing Customizing Transport Requests (old/new)

To copy the contents of a transport request from one client to another client, use transaction **SCC1** from within the target client. Enter both the source client and the transport request of which the objects should be copied.



#### Hint:

Only objects of one transport request may be copied at a time. As copying objects of each transport request individually may be time consuming, you can use the *Include Objects* option in transaction **SE09** to bundle the objects of several transport requests into one transport request. The resulting objects of this larger transport request can then be copied to the target client, saving time.



#### Note:

If the transport request contains cross-client objects, these objects are not copied.

With FP01 for SAP S/4HANA 2020 (that is SAP\_BASIS 755 SP01), SAP ships transaction **SCC1N** as successor for transaction **SCC1**. With the help of this new transaction, you can copy customizing objects recorded in transport requests to several target clients. A transport request in this context can also be a local transport request or an imported transport request from another system. In contrast to transaction **SCC1**, **SCC1N** can be executed in any client. In addition, a large number of new parameters are available.

If you want to copy transport requests with transaction **SCC1N** that have not been released, proceed as follows:

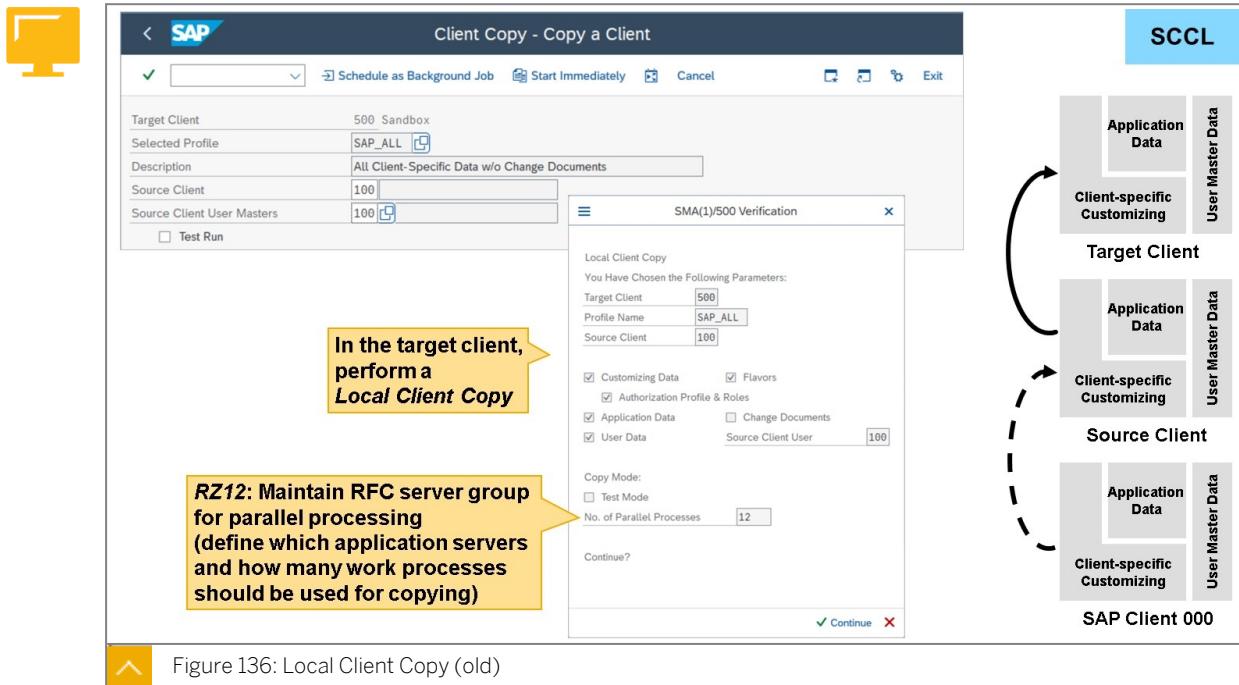
- In the *Export / Import Time of the Transport Request* area, select the *Local Import/Update Date* radio button.
- If it is a mandatory field (this depends on the release / SP level), enter a date that is far in the past in the *Export/Import Date* field.

For more information, see the blog *New Tool to Copy Configuration Data to Clients (SCC1N)* (available at <https://blogs.sap.com/2022/02/15/new-transport-based-client-copy-scc1n/>)

## Local Client Copy

A local client copy copies data between clients within the same SAP system.

### Transaction SCCL



When using transaction SCCL and / or transaction SCC8 (local and remote client copy using the “classical” tools), the client copy must be initiated from the target client using the following steps:

1. Create an entry for the target client in the client maintenance table. For this, choose transaction SCC4.
2. Because no user exists in the new target client right now, no SAP\* user exists. In this case, you can log on to the target client as **SAP\*** with the hard coded password **pass**. Perform the client copy using transaction SCCL. Select the data to be copied using a profile.

**Note:**

The user name SAP\* with the password pass is only active if the profile parameter *login/no\_automatic\_user\_sapstar* is set to the value 0.

3. Assign the source client(s) for customizing data, application data, and user master records.
4. Start the client copy. Because copying takes a long time, use background processing.

**Caution:**

To ensure data consistency, you can't work in the target client during a client copy. SAP recommends that you don't work in the source client during a client copy.

**Note:**

SAP delivers the software with standard client 000. You can't work in client 000, but you can use client 001 as your productive client, if it exists. Client 001 is a copy of client 000. However, if you don't want to use client 001, SAP recommends that you begin the SAP system implementation by creating a new client as a copy of client 000. Don't forget to delete client 001 in this case.

You can improve the performance of the client copy, for example, by excluding tables or packages, with *Edit → Expert Settings*. You can exclude tables from the client copy, for example if they are not relevant for the target client, in the *Tables* tab. For further information about excluding tables from the client copy, choose the *Documentation* button in the application toolbar of the expert settings. For further information about expert settings in the client copy, see SAP Note [446485](#) – CC-ADMIN: Special copying options.

**Transaction SCCLN (as of SAP\_BASIS 754)**

Starting with SAP\_BASIS 754, the local client copy procedure was changed and enhanced. To perform a local client copy with transaction SCCLN, proceed as follows:

1. Call transaction SCCLN.
2. Select a source client and a target client.
3. Define the scope for the client copy. You can use either select a client copy profile or you can specify selected tables.

**Note:**

SAP recommends that you copy individual tables only in exceptional circumstances.

4. Define the processing setting, for example, check the copy by selecting *Test Mode*, lock the source client for logons, define the number of processes running in parallel.
5. Specify whether you want to schedule the client copy as a task list (either in dialog box or as a background job) or to start it immediately.

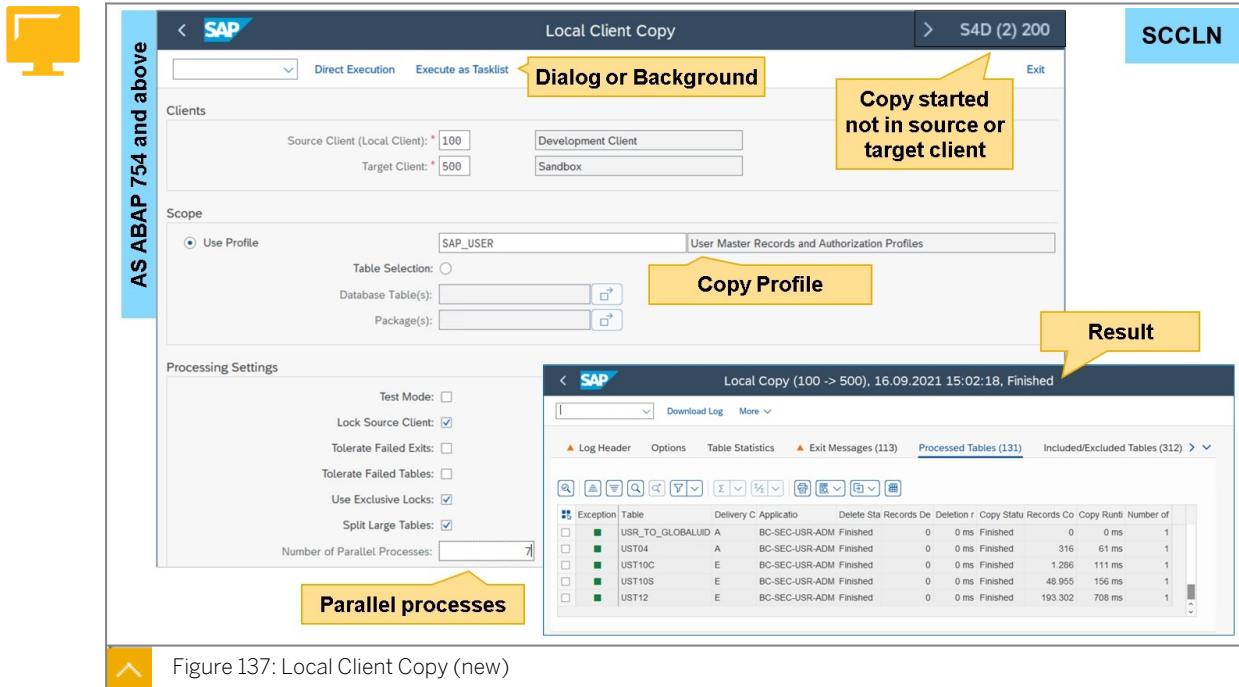


Figure 137: Local Client Copy (new)

### Using Parallel Processes

You can use parallel processes for remote and local copies and to delete clients, to exploit the capacity of your database better.

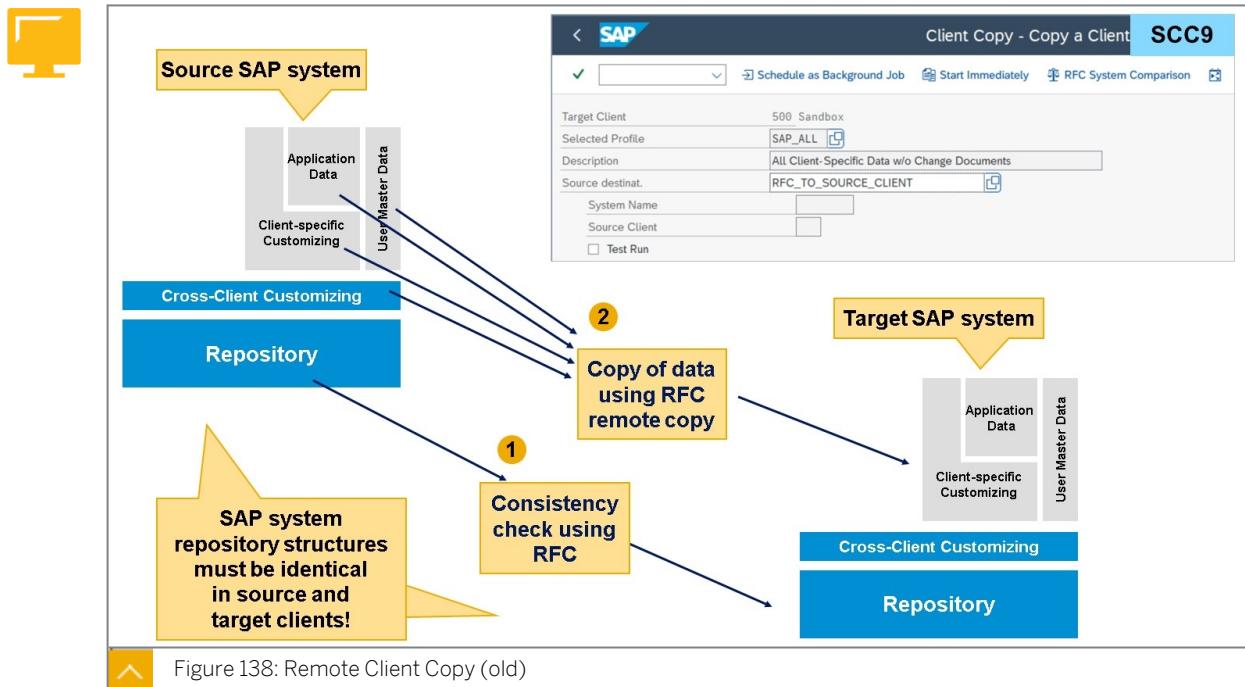
**Note:**  
Valid for SCCLN only: Two processes per available database CPU is a good guideline. The number of application servers is unrestricted.  
Parallel processes are only used during the actual copy phase, not during the analysis and postprocessing phases.  
The resource management may only assign a limited number of processes to the client copy.

### Remote Client Copy

A remote client copy allows you to copy data between clients of different SAP systems. You can use a remote client copy to, for example, copy client-specific and cross-client customizing data between SAP systems.

A remote client copy proceeds in the same way as a local client copy, but gets the data from a *Remote Function Call* (RFC) destination to the source client.

## Remote Copy with Transaction SCC9

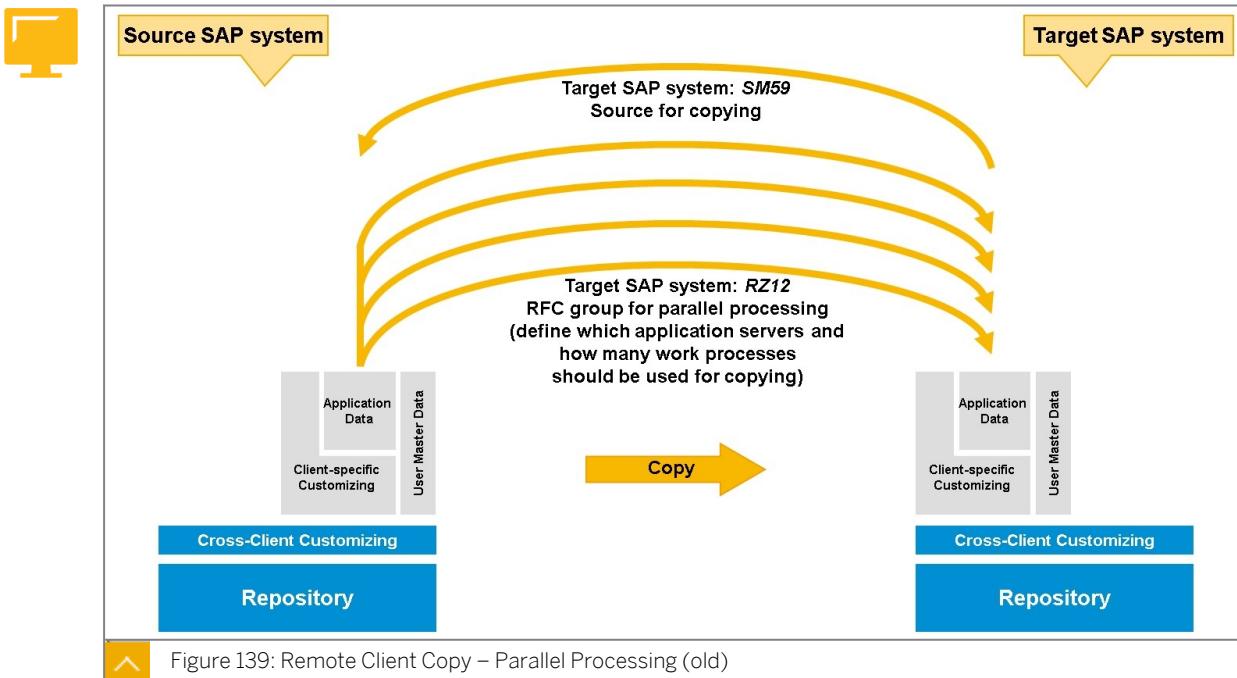


A remote client copy is easy to use and does not require file system space. Consider the following:

- Cross-client customizing can be copied during a remote client copy, depending on the client copy profile.
- To be able to copy all data during the client copy, the structures of all copied tables in both SAP systems must be identical. During remote client copy, an automatic repository consistency check is performed. If inconsistencies are detected, the client copy is terminated and an error message is displayed. You can also perform the consistency check yourself with the *RFC System Comparison* button on the start screen of transaction SCC9.

If you experience problems copying from another SAP system, see SAP Note [557132](#) – CC-TOPIC: *Remote client copy*.

For more information about remote client copies, see also SAP Note [47502](#) – CC-REMOTE: *Messages of the Remote Clientcopy*. For additional information on client copy in general, see SAP Note [1537913](#) – Client Copy: Local, Remote, Export/Import & deletion.



The client copy function can be used with multiple parallel processes, which speeds up the copying process. Other factors that should be taken into consideration to optimize a client copy include the network performance and the database performance.

The client copy processes are generated at run time. They are parallel RFC processes, managed by *RFC server groups*. You need to specify the maximum number of processes you want to use in the client copy program.



**Note:**  
Parallel processing can only be used when performing a local or remote client copy, it can't be used for client transports.

To specify the maximum number of client copy processes, use transaction **SCCI** for a local copy or transaction **SCC9** for a remote copy and then choose *Goto → Parallel Processes*.

To define an RFC server group, use transaction **RZ12**. For RFC server groups, the limitations on the SAP system parameters are explained in SAP Note [99284 – RFC exception: RESOURCE\\_FAILURE](#). See SAP Note [2001276 – Changed configuration as of 7.40 SP2](#) for changes as of AS ABAP 7.40.

During the execution of a client copy using parallel processing, each parallel process checks the other to see if it is still running. If a process has failed, it will be restarted automatically.



**Hint:**  
In an existing client, application data, client-specific customizing, cross-client customizing, and the repository rely on each other. Even when including the cross-client customizing, a remote client copy will only be 100% consistent if the complete repository of the source and target system is identical. However, this will never be the case in “real” SAP systems.

Finally, note the following:

- When you copy a client from one system to another, the data is transferred directly using the RFC interface – there is no intermediate storage on hard disk.
- The systems can be on different platforms.
- The target client number can be different from the source client number.
- Code page conversions are performed automatically, as far as technically possible.
- Do not use the same client as the source for multiple copies or client transports at the same time.

### Remote Client Copy with Transaction SCC9N (as of SAP\_BASIS 754)



Note:

This section is valid only for SAP systems based on SAP\_BASIS 754 or higher.

Starting with SAP\_BASIS 754, the remote client copy procedure was changed and enhanced.

For a remote copy with transaction SCC9N, proceed as follows:

1. Call transaction SCC9N in the target system on a client different from the target client.
2. Select or enter an RFC destination to the source system and source client and select a target client.
3. Define the scope for the client copy. You can use either select a client copy profile from the existing profile of the client copy from the list or select specific database tables or packages.
4. Make the *processing settings* as in a local copy, for example, check the copy by selecting *Test Mode*, lock the source client for logons, specify whether you want to split large tables, specify whether you want to copy incompatible tables (RFC). See the figure, Remote Client Copy with Transaction SCC9N.
5. Define the number of processes running in parallel.
6. Specify whether you want to schedule the client copy as a background job (*Execute as Tasklist*) or to start it immediately (*Direct Execution*).
7. Start the copy.

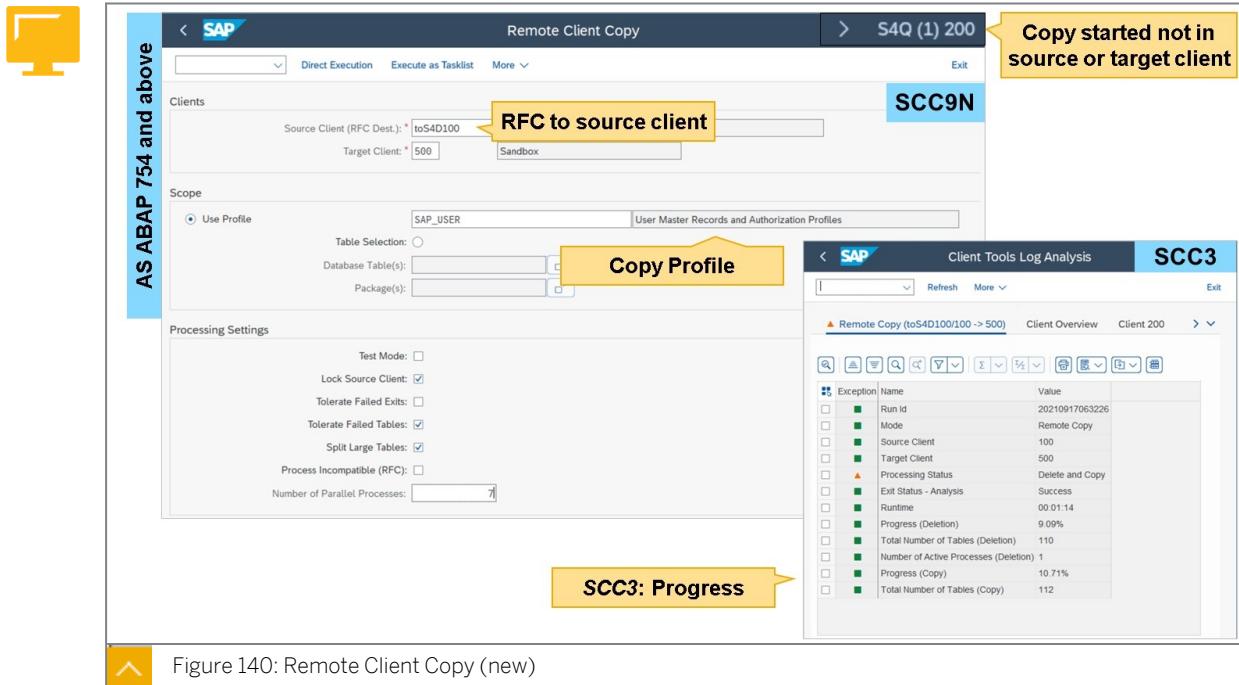


Figure 140: Remote Client Copy (new)

**Hint:**

When you are preparing to perform a remote client copy in an SAP S/4HANA environment and you are looking for performance recommendations, see SAP Note [2953662](#) – *Recommendations for remote client copy performance improvements in S/4HANA* for more information.

## Client Transport

A **client transport** differs from a client copy in that it does not use RFC. Like a client copy, however, a client transport is used to copy data between different clients. A client transport can be used to replace a local or a remote client copy.

A client transport consists of two steps. First, a client export extracts client data from the source client to files at the operating system level. Then the data is imported from these files into the target client.

## Client Transport with the Help of Transactions SCC8 and STMS

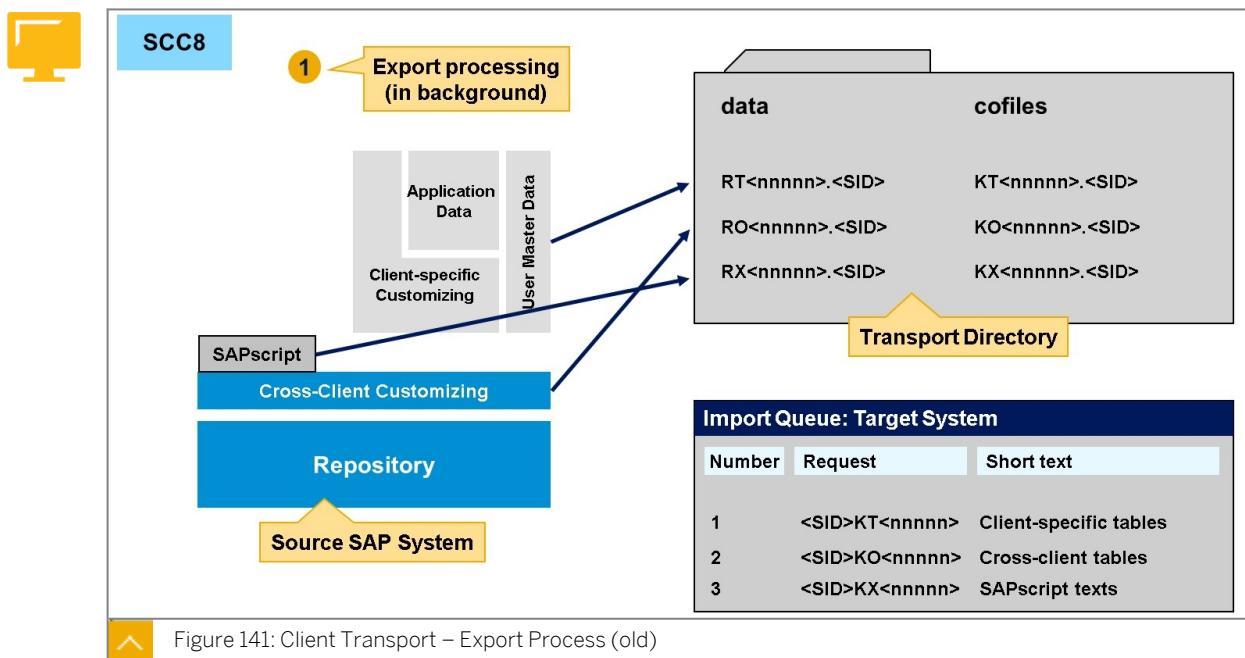


Figure 141: Client Transport – Export Process (old)

To perform a client export with the help of transaction SCC8, follow these steps:

1. Log on to the source client. Choose transaction SCC8 (client export). Select the data to be copied using a (client copy) profile.
2. Specify the target system to which the client will be copied. The target system must be defined in the Transport Management System (TMS) as part of the transport domain.
3. Start the client export. Because copying may take a long time, use scheduled background processing.

The client export performed in the source system <SID> exports the client data asynchronously by calling the transport program *tp* at operating system level. This export process will generate up to three data files at the operating system level:

- *RO< number >.<SID>*: this file contains cross-client data
- *RT< number >.<SID>*: this file contains client-specific data
- *RX< number >.<SID>*: this file contains SAPscript texts

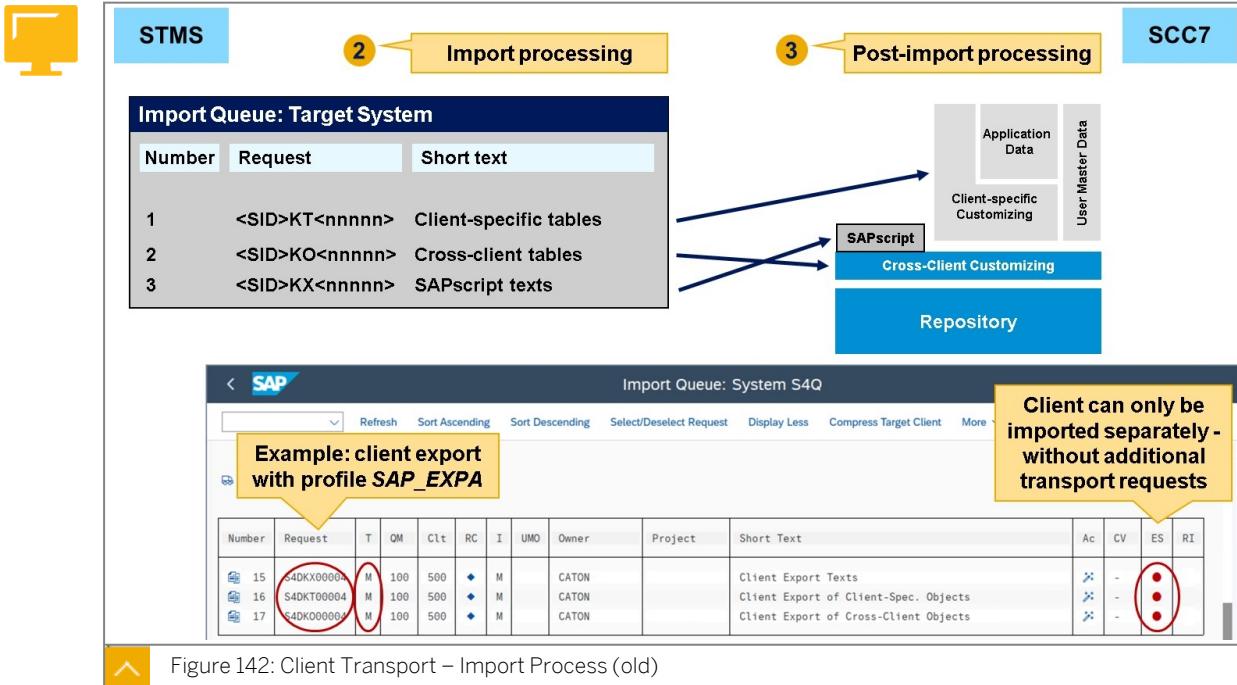


**Hint:**

The data export is asynchronous and still runs even after transaction SCC8 has already finished. Don't run any other client copy tool before the data export is finished. As usual, you can check the status of the export with transaction SCC3. You can also check the status of the client export with the transaction SE01. Here, display the logs for the transport request <SID>KT<number> on the *Client* tab.

**Note:**

During the client export, there might be no information available in transaction SCC3, (*Client*) Exports area, because the export is done with *R3trans*. Depending on the release of SAP\_BASIS it may say *Aborted* in transaction SCC3 while the client export is running , instead of *Processing...*, as you would expect.



Depending on the type of data selected through the client transport profile, up to three client copy transport files are added to the buffer of the target system. Those files are related to the data files on operating system level described earlier.

The client export transport requests are not imported when an *Import all* takes place. You can import these requests into the target client using *import single* in TMS: Choose one of the transport requests of the client transport in the *Transport Management System* (TMS). The other transport requests belonging to this client transport are then automatically imported in the correct order.

After the import process has completed, post-import activities are required for object generation steps. For this, after completing the import, log on to the target client. Choose transaction SCC7 for the post-processing steps.

To display client transport logs, use the *Transport Organizer – Extended View* (transaction SE01), *Client* tab.

During client export, a repository consistency check can be performed by choosing *RFC system check* in transaction SCC8. If inconsistencies are detected, a list of the ABAP Dictionary table definitions missing or different in the target system will be generated. This will help you to recognize in advance problems that may occur during the import of the source data.

### Client Export with Transaction SCC8N (as of SAP\_BASIS 754)

Starting with SAP\_BASIS 754, the client transport procedure was changed and enhanced.

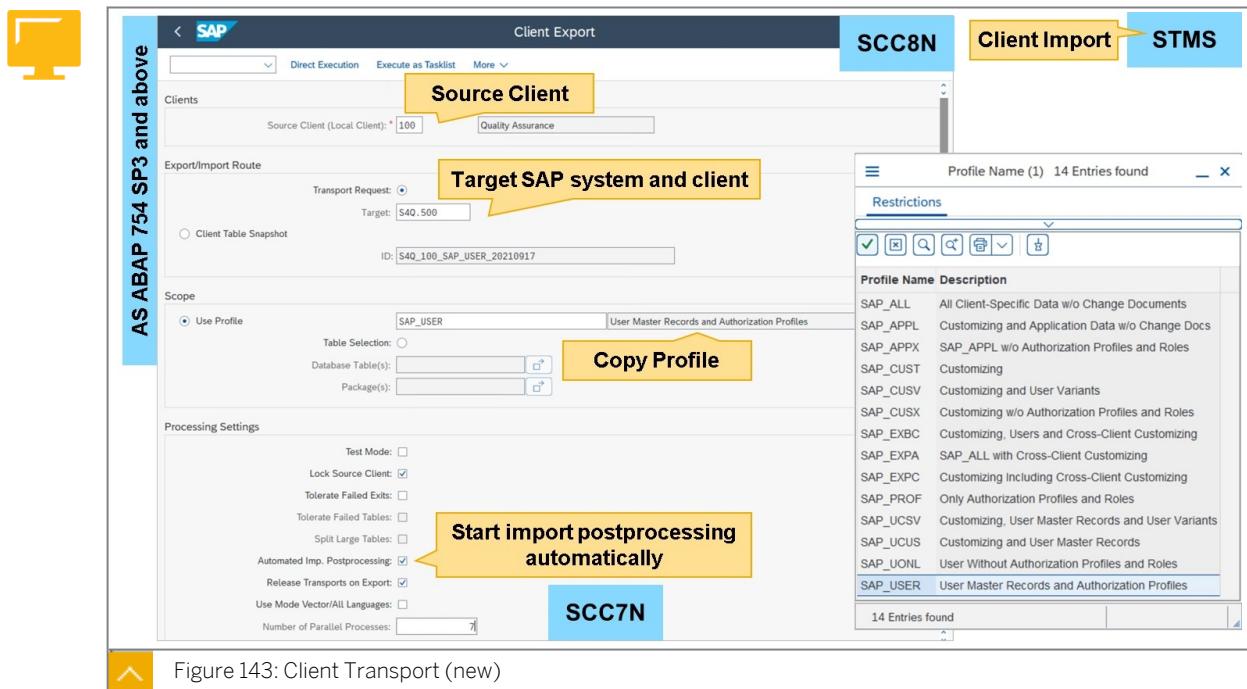


Figure 143: Client Transport (new)

A client transport with the help of the new client tool comprises three steps:

1. Client export (SCC8N):
  - a. Call transaction SCC8N.
  - b. Select the source client.
  - c. Select the *Export/Import Route*. To do this, you can, for example, select the target of a transport request.
  - d. Define the scope for the client copy. You can either use the previously defined profile of the client copy (direct entry or selection from the drop down list) or select table(s) or package(s).
  - e. Define the processing settings, for example, *Test Mode*, *Tolerate Failed Exits*, *Automated Import Post Processing* or *Release Transports on Exports*.
  - f. Specify whether you want to schedule the client copy as a background job (*Execute as Tasklist*) or to start it immediately (*Direct Execution*).
2. Client import (STMS):
  - a. Choose one of the transport requests of the client transport in the *Transport Management System* (transaction STMS, *Import Queue Overview*). The other transport requests belonging to this client transport are then automatically added in the correct order.
  - b. Import these transport requests into the target client. For this, you may need to be logged on to the target client.
3. Client import post processing (SCC7N):



Note:

You need to perform post processing activities to adapt the runtime environment to the current state of the data.

- a. Call transaction SCC7N.
- b. Select the *Target Client of the Import*.
- c. Select the *Export/Import Route*. To do this, you can for example select the target of a transport request.
- d. Check the *Client Import Information: Client-Specific Transport, Cross-Client Transport, Profile, Source System, and Source Client*.
- e. Define the *Processing Settings: Test Mode, Tolerate Failed Exits, Tolerate Failed Tables, Use Exclusive Locks, and Number of Parallel Processes*.
- f. Specify whether you want to schedule the client copy as a background job (*Execute as Tasklist*) or to start it immediately (*Direct Execution*).



Note:

Client import post processing is always required and must be performed in the target client after the import of the transport requests. When you have selected *Automated Imp. Postprocessing* in the client export (transaction SCC8N), it is not necessary to start the client import post processing manually.

Don't use the same client as the source for multiple copies or client transports at the same time.

## Client Delete



Hint:

For deleting clients 001 and 066, see SAP Note [1749142](#) – How to remove unused clients including client 001 and 066.

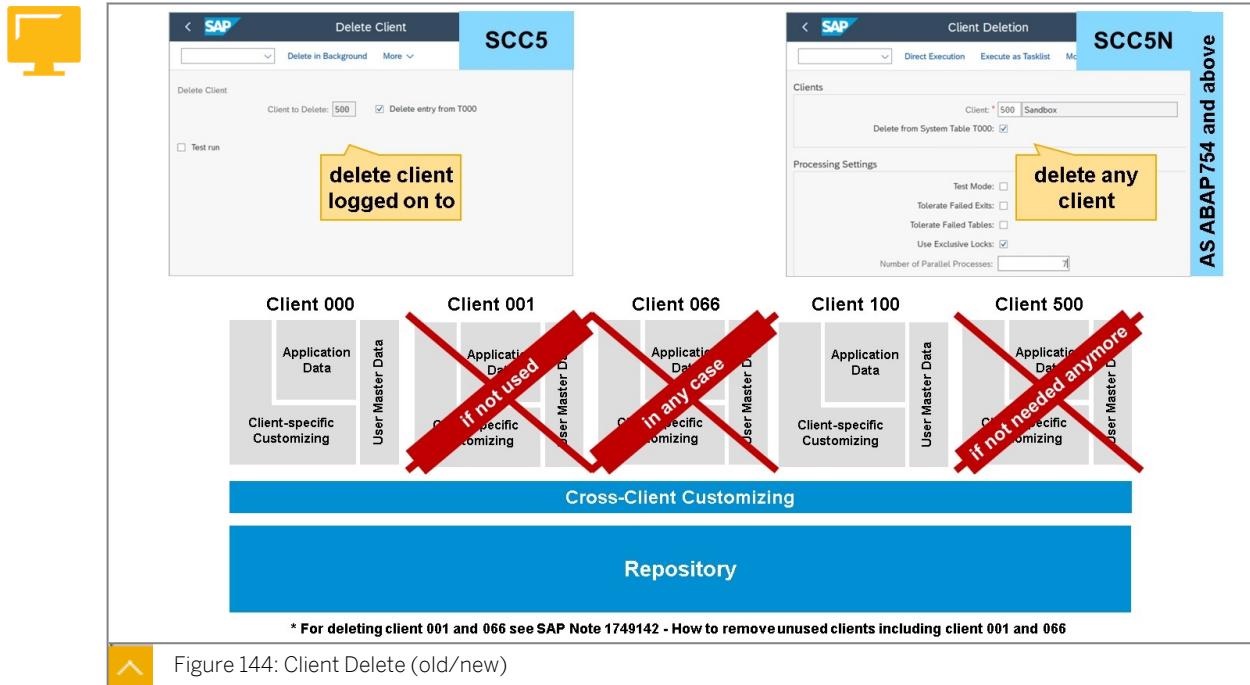


Figure 144: Client Delete (old/new)

### Deleting a Client with Transaction SCC5

To delete a client from within an SAP system with the help of transaction SCC5, proceed as follows:

- Log on to the client to be deleted.
- Use transaction SCC5.
- Start the deletion of the client, preferably using background processing.

When you delete a client entry from table *T000* with *Client Maintenance* (transaction SCC4), you can no longer log on to the client or update it using transport requests. This, however, does not eliminate the data that belongs to the client. This means that the client specific data remains in your SAP system, occupying space in the database. Therefore, to eliminate an SAP client entirely, that is, to delete both the client and the client specific data, use the *Delete Client* function, transaction SCC5.



#### Hint:

Deleting a client entry with *Client Maintenance*, transaction SCC4, allows you to temporarily lock the client. This deletion procedure preserves the data for the client but prevents users from logging on to the client or accessing the data belonging to the client. To restore the client and allow logon, recreate the client entry using *Client Maintenance*.

The time required for the deletion of a client can be reduced by performing the deletion using parallel processes.

### Deleting a Client with Transaction SCC5N (as of SAP\_BASIS 754)

Starting with SAP\_BASIS 754 the local client copy procedure was changed and enhanced.

To delete a client with the help of transaction SCC5N, proceed as follows:

1. Call transaction SCC5N. You do not need to do this from within the client you want to delete.
2. Select the client from the list that you want to delete. Optional, you can specify whether you want to delete the client from client administration (table T000).
3. Define the processing settings, for example, check the deletion by selecting *Test Mode* or define the number of processes running in parallel.
4. Specify whether you want to schedule the deletion as a background job (*Execute as Tasklist*) or to start it immediately (*Direct Execution*).
5. Start the deletion.

### **Client Copy Profiles**

When copying, you can use copy profiles that simplify selection and combination of the objects to be copied. SAP delivers the following copy profiles in the table below. The customizing and application data is deleted in the target client before copying for all profiles except SAP\_USER and SAP\_UONL. This is, technically, unavoidable.

Roles are also copied with the user master records using the profile SAP\_USER.

Table 7: Overview of Copy Profiles (General)

Copy Profile	Meaning
SAP_USER	User master data, user roles, and authorization profiles are copied. The client is not reset.
SAP_UONL	User master data without authorization profiles and roles
SAP_PROF	Only authorization profiles and roles
SAP_CUST	Client-specific customizing including the authorization profiles is copied. The application data is deleted and the user master data of the target client is retained.
SAP_CUSV	SAP_CUST with variants
SAP_CUSX	SAP_CUST without authorization profiles and roles
SAP_UCUS	SAP_CUST with user master data
SAP_UCSV	SAP_UCUS with variants
SAP_ALL	All client data except change documents (see Note 180949) and local data is copied.
SAP_APPL	SAP_ALL without user master data
SAP_APPX	SAP_APPL without authorization profiles and roles

Table 8: Additional Copy Profiles for Remote Copies (SCC9(N))

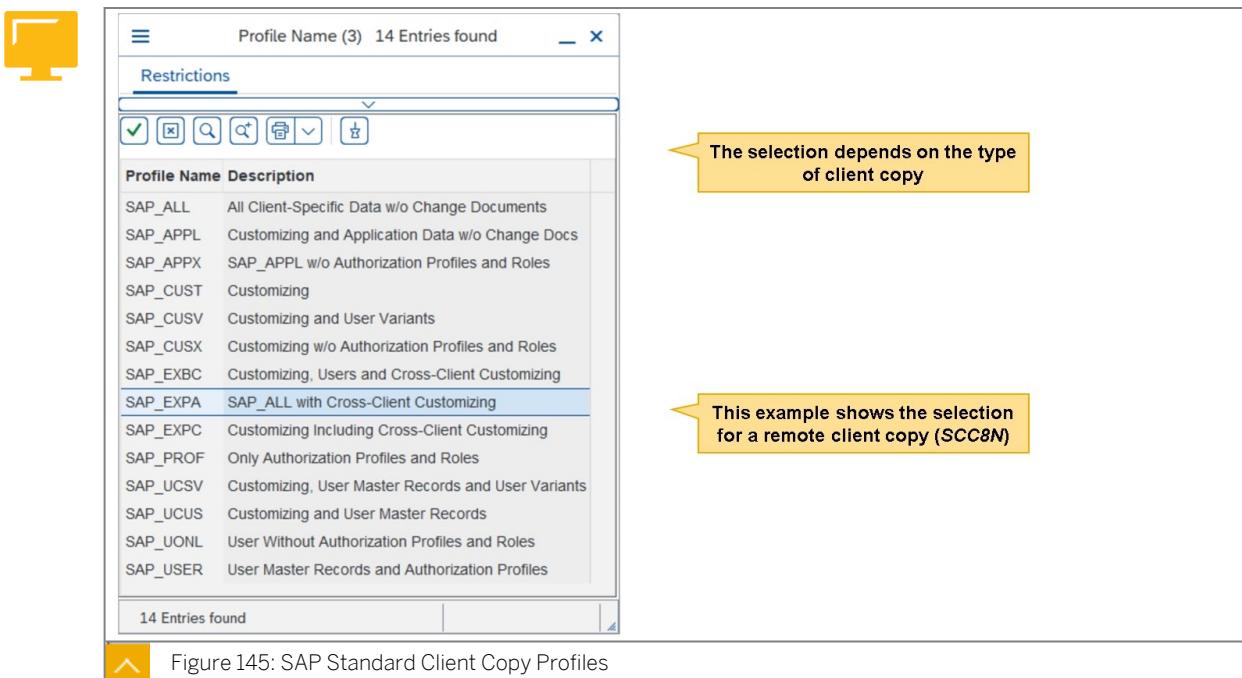
Copy Profile	Meaning
SAP_RMBC	SAP_UCSV with cross-client customizing
SAP_RMPA	SAP_ALL with cross-client customizing
SAP_RMPC	SAP_CUSV with cross-client customizing

Table 9: Additional Copy Profiles for Client Transports (SSC8(N))

Copy Profile	Meaning
SAP_EXBC	SAP_UCSV with cross-client customizing
SAP_EXPA	SAP_ALL with cross-client customizing
SAP_EXPC	SAP_CUSV with cross-client customizing

Table 10: Special Profiles (SCC8 and SCC9(N) Only)

Copy Profile	Meaning
SAP_RECO	This profile is only for recovering an accidentally deleted client (see SAP Note <a href="#">31496</a> ). It contains local tables of delivery classes L and W and the change documents, as well as SAP_ALL.



The screenshot shows a SAP interface titled "Profile Name (3) 14 Entries found". The list includes:

- SAP\_ALL All Client-Specific Data w/o Change Documents
- SAP\_APPL Customizing and Application Data w/o Change Docs
- SAP\_APPX SAP\_APPL w/o Authorization Profiles and Roles
- SAP\_CUST Customizing
- SAP\_CUSV Customizing and User Variants
- SAP\_CUSX Customizing w/o Authorization Profiles and Roles
- SAP\_EXBC Customizing, Users and Cross-Client Customizing
- SAP\_EXPA SAP\_ALL with Cross-Client Customizing**
- SAP\_EXPC Customizing Including Cross-Client Customizing
- SAP\_PROF Only Authorization Profiles and Roles
- SAP\_UCSV Customizing, User Master Records and User Variants
- SAP\_UCUS Customizing and User Master Records
- SAP\_UONL User Without Authorization Profiles and Roles
- SAP\_USER User Master Records and Authorization Profiles

Annotations:

- A yellow callout points to SAP\_EXPA with the text: "The selection depends on the type of client copy"
- A yellow callout points to the bottom of the list with the text: "This example shows the selection for a remote client copy (SCC8N)"

For all typical client copy scenarios, SAP provides profiles affecting various kinds of data in the SAP system and using various copy functions (as shown above).

Depending on the client copy option you have selected, the client copy tool determines the purpose of the table in the SAP system when the table is copied. For example, if you selected the *Customizing* profile and the table belongs to customizing, then the table would be copied to the target client. If the table contains application data, however, the table contents will not be copied (and existing data will be deleted in the target client).



#### Note:

To determine the purpose of a table, the client copy tool uses the *table delivery class* assigned. The client copy tool assumes that all tables belong to customizing except those that have table delivery classes *L*, *A*, and *W*.

Figure 146: Table Delivery Classes

To understand the effect of table delivery classes during upgrading, transport, or client copy, especially if there are development projects that create new tables, see the SAP Online Documentation and SAP Note [2857 – What table delivery classes exist? What do they mean?](#).

Table delivery classes are assigned to tables as follows:

- In general, tables containing application data should have delivery class A.
- Tables containing customizing data have different delivery classes, depending on how SAP changes the customizing settings in them.
- Tables of delivery class C may not contain new entries from SAP.
- Tables of delivery class G may be changed by SAP.
- Delivery classes S, E, and W are for tables containing control data and SAP system data, such as tables TADIR, VARI, TSTC, and CCCEXIT.



**Note:**

W tables are SAP system tables that are filled internally. Their contents are purely technical. W tables are not taken into account by the client copy tool.

- Delivery class L is for tables for temporary data such as spool data and background processing data, for example, tables TSP\* and TBTCJOB\*.



**Note:**

These tables should be empty in a new SAP system and its contents should be deleted in the target client.

## Identity of Table Structures When Performing Cross-System Client Copies

To set up a new client by client copy, the structures of all tables to be copied must be identical in the source and the target system. The table structures are flagged as incompatible in the following cases:

- If the target system has an additional key field.
- If the table structures have different field names, data types, or key fields.
- If a field in a table has a shorter field length in the target system than in the source system since this can lead to data loss.

For local client copies, the table structure for the source client and the target client are identical. for cross-system client copies, the client copy program compares the table structures in the source and target system. Any incompatible tables are excluded from the copy. The client copy continues to run.

- *Remote client copy*: If inconsistencies are found, the incompatible tables are excluded from the client copy. SAP recommends that you first run the remote client copy in test mode. You can then analyze the error messages in the log and decide whether you still want to make the client copy.
- *Client transport*: If an RFC connection can be established between the SAP systems, you can use the program *RSCLICCHK* to perform a client transport. As of SAP\_BASIS 754 SP3, you can start the program directly from transaction *SCC8N* by starting a client copy in test mode.

To ensure data consistency in the target client, users, except for SAP\* and DDIC, can't log on to the target client during copying. Similarly, no one should work in the source client during the client copy.



### Note:

Tables with the delivery class L and W are ignored by the copy. Exceptions are the deletion of clients (transaction *SCC5 /SCC5N*) or when you use the copy profile *SAP\_RECO*.

When you start a client copy, take runtime restrictions and database space into account. Storage requirements can only be estimated, because space already allocated, but not yet used, is not taken into account.



### Note:

Depending on the type of databases, space which has become free is available only after a reorganization.

To test whether there is sufficient database space, perform a test client copy. Using the classical transactions, this can be executed either as a *Simulation* or as a considerably faster *Resource check*; the resulting display shows the database memory requirements. The new client copy tools offers to select the *Test mode* on the selection screen.

**Hint:**

To estimate the size of a client you can also schedule a *Test run* with option *Resource check*.

On how to copy large production clients, see SAP Note [489690](#) – CC INFO: *Copying large production clients*.

**Note:**

For performance aspects, see SAP Note [2163425](#) – *Recommendations for client copy performance improvement*.

## Monitoring

To display all log files for a completed local or remote client copy, choose transaction SCC3. Here you can select the relevant target client and the relevant copying process. These copy logs provide the following information:

- Table statistics
- Control information
- Information about each table copied, including the link to the appropriate IMG component

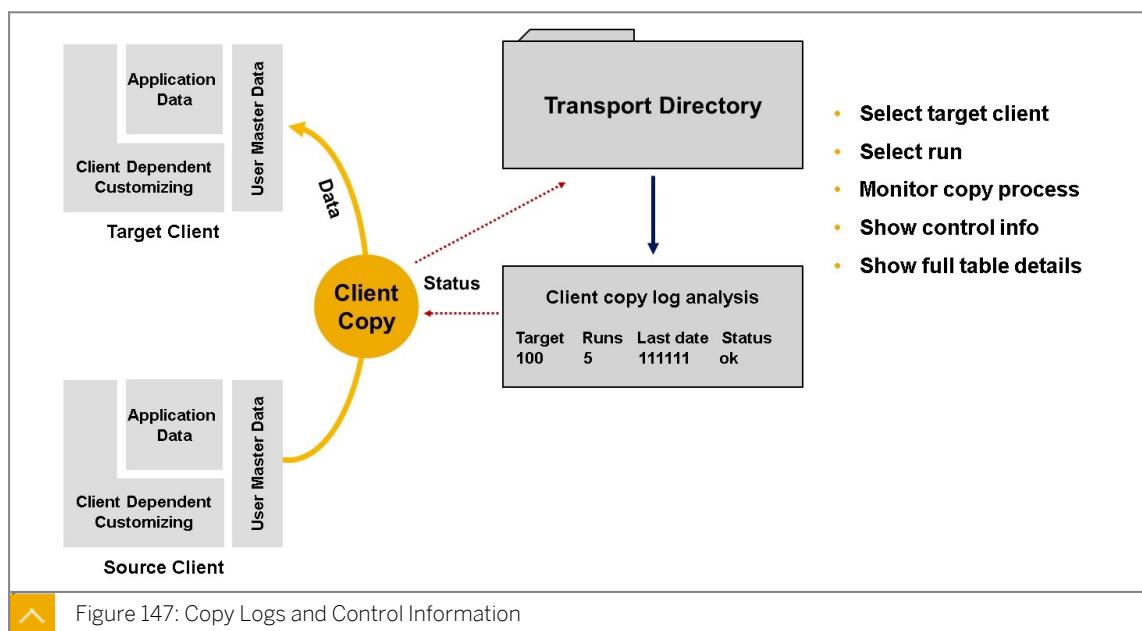


Figure 147: Copy Logs and Control Information

Up to (and including) SAP\_BASIS 753, double-click the highlighted entries for further information. You get a selection list in which you can see information about the copied tables (for example, runtime and the number of entries copied) and the exit programs from other application components used. If you had tables with copy problems in a local or remote client copy, you can recopy them with the *Recopy Errors* function. You can display the original log in the file system with the *File Log* button. You can also display the export and import logs of the main transport request.

To display client transport logs, use the *Transport Organizer (Extended View)*, transaction SE01, tab *Client*.



**Hint:**

Technically, copy log information is located in the view *V\_CCCFLOW*. This view contains, for example, the run time and processing status of a local client copy, the number of already copied tables, and the name of the table being currently copied.

All log files are physically stored in the *<transport directory>/log* directory at operating system level. Log files are named *CC<number>.<SID>*, where *<number>* is the serial client copy number, and *<SID>* is the source system ID.

If copying is unexpectedly terminated, the SAP system uses the name of the table being currently copied to restart the copy process.

Figure 148: Client Copy – Monitoring and Troubleshooting (old)

Target Client	500		
Source Client (incl. Auth.)	100		
Source Client User Master	100		
Copy Type	Local Copy		
Profile	SAP_ALL		
Status	Processing...		
User	SAP*		
Start on	17.09.2021 / 11:48:41		
Last Entry on	17.09.2021 / 11:58:21		
Current Action:	Copy/Delete Tables		
Process	Server		
00001	smhost	T024	11:58:20
00002	smhost	TNHCF_BO	11:58:20
00003	smhost	TTEPD0_COUNTRIES	11:58:20
00004	smhost	T77GMSC1	11:58:18
00005	smhost	WRMA_SHRP_1	11:58:20
00006	smhost	WSRMB_SEQ_HDR_C	11:58:20
00007	smhost	SXMSALERTRELMSG	11:58:20
00008	smhost	T770MATGTC	11:58:20
00009	smhost	TUWS_DATA	11:58:20
00010	smhost	T77PO	11:58:21
00011	smhost	T778U	11:58:20
00012	smhost	T77HAP_C_WFE	11:58:20
Statistics for this Run			
- No. of Tables	26819	of	26819
- Deleted Lines	6		
- Copied Lines	3107690		

When performing client copy error analysis, check not only the copy log, but also the SAP system log (transaction SM21), which tells you whether database problems are responsible for the client copy error. Correct any database problems before restarting the client copy.



**Hint:**

SAP system log messages like *Syn. MC maintenance deactivated fully*, *Buffer TABL/TABLP reset*, and *Table logging* don't indicate an error. These messages document special functions that are used to improve performance and guarantee consistency.

When a client copy run terminates abnormally, the *Restart* option is proposed by default, which continues the run from the point at which it was terminated. If the original run was recent, as indicated by the status line, it is advisable to choose *Restart*. Alternatively, to start the run from the beginning, choose *Restart → New Start*.

**Note:**

If a copy terminates for technical reasons, for example, due to a database shutdown, you can restart it with the same settings. All exits are processed again. Any tables already copied in the client copy are skipped if they are unchanged.

The last step is restarted. You cannot continue to copy an incompletely copied table, the table is reinitialized and recopied.

If the restart fails, the log displays possible reasons for the error. Before you try to restart the program again, eliminate the error.

If a client copy or client import post-processing did not finish, the SAP system automatically proposes the restart mode when you call the transaction. The same parameter settings are used automatically.

To avoid the problem of insufficient free space in the database, perform a simulation test run or a resource check.

- A *simulation test run* estimates the space required by reading all records to be copied without updating them in the database.
- A *resource check* estimates the space required by counting the records to be copied.

**Hint:**

Client copies ignore tables in the local development class \$TMP. If you want to copy these tables, modify the development class in the object directory.

For more information about error analysis with client copy tools, see SAP Note [22514](#) – CC-INFO: Error analysis for client copy.

If you discover a write error in the target client, it is usually due to insufficient space in the database or simultaneous processing in the target client. In this case, check the SAP system log (transaction SM21) to determine the error. Don't work in either the source or target client during the copy.

If a termination in an exit program (the termination is explained in an ABAP runtime error) occurs, run the log display program RSCCPROT to determine the name of the last exit program that has caused the termination. The client copy program has not crashed; this is an application error.

If you discover an unexplained cancellation or error, check the:

- Copy log (transaction SCC3)
- SAP system log (transaction SM21)
- Dump analysis (transaction ST22)
- Job overview (transaction SM37)
- Spool requests (transaction SP01)

### **Enhancements as of SAP\_BASIS 754**

Starting with SAP\_BASIS 754, the copy logs and control information was changed and enhanced (see the following figure):

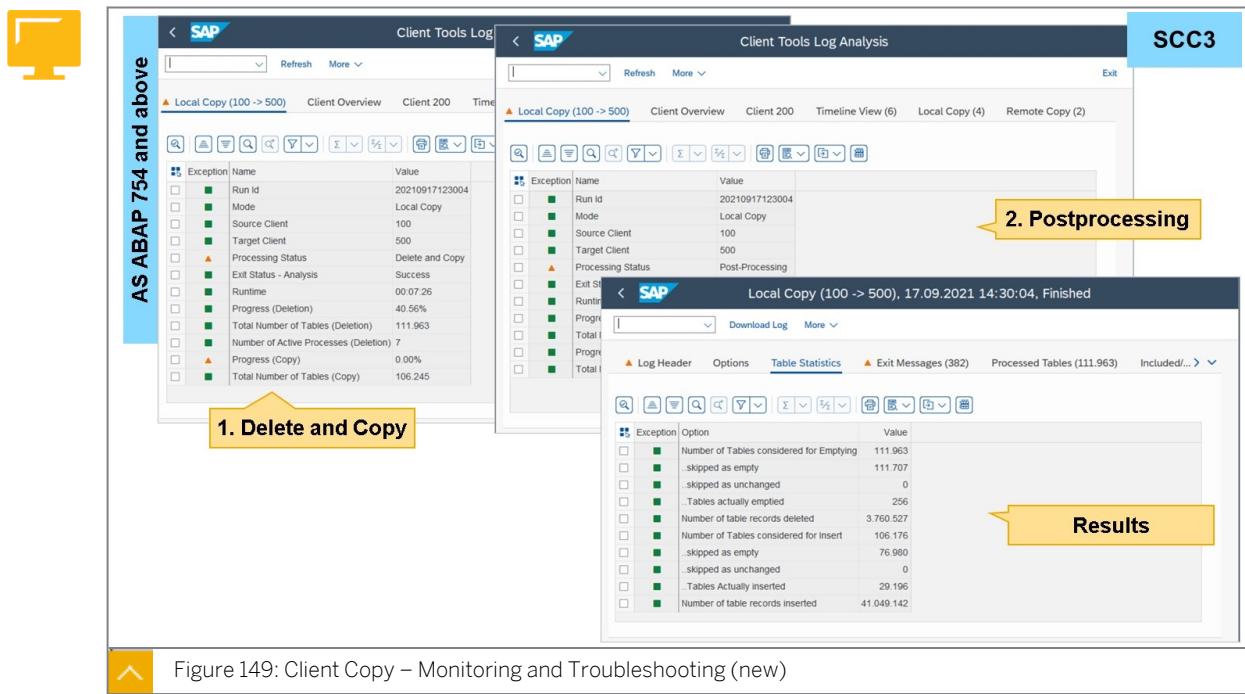


Figure 149: Client Copy – Monitoring and Troubleshooting (new)

The client copy log has been transformed from a file-based to a table-based UI with additional information and better persistence.

To access the *Client Tools Log Analysis*, which contains an overview of all client copies that have been performed, call transaction scc3. The start page of this transaction contains the following tabs (in case that the respective tasks have already been performed in the system):

- *Active Process*: Displays the currently running processes of a client.
- *Client Overview*: Lists all clients available in the SAP system.
- *Client <nnn>*: Displays the overview of client copy tool processes when a client was target for another client.
- *Timeline View*: Total number of timelines of all client copies.
- *Local Copy*: Total number of local copies performed.
- *Remote Copy*: Total number of remote copies performed.
- *Client Deletion*: Total number of client deletions performed.
- *Client Export*: Total number of client exports performed.
- *Client Import*: Total number of client imports performed.
- *Copy by Transport*: Total number of copies performed using a transport request.
- *Comparison*: Total number of client table comparisons performed.
- *Client Table Snapshot*: Total number of snapshots created (if available).



#### Note:

You can download the log to your local front end.



## LESSON SUMMARY

You should now be able to:

- List client copy and client transport tools
- Outline the idea of copies based on transport requests
- Create a new client with the help of a local client copy
- Illustrate the use of remote client copies, client transports and client deletions
- Describe the monitoring of client copies

# Client Compare and Client Maintenance Tools

## LESSON OVERVIEW

In this lesson, you will learn how to perform a client compare and how to maintain clients with the help of a client adjustment.

### Business Example

In one of your production systems, you encounter a specific problem related to a customizing table. Assume that you have multiple SAP production systems, and the customizing settings work in one SAP system but not in another. Therefore, you would like to compare a specific customizing table across multiple SAP systems to help you find the root cause of the problem.



## LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Perform a Client Comparison
- Outline the idea of client adjustments with respect to Customizing
- Describe how to compare two repositories

## Using Client Compare Tools

When several SAP systems and clients are being implemented, it may be necessary to compare and adjust customizing settings between different SAP systems and clients. The *Client compare* function enables you to compare and adjust the contents of a table or view in two different clients, using RFC destinations.

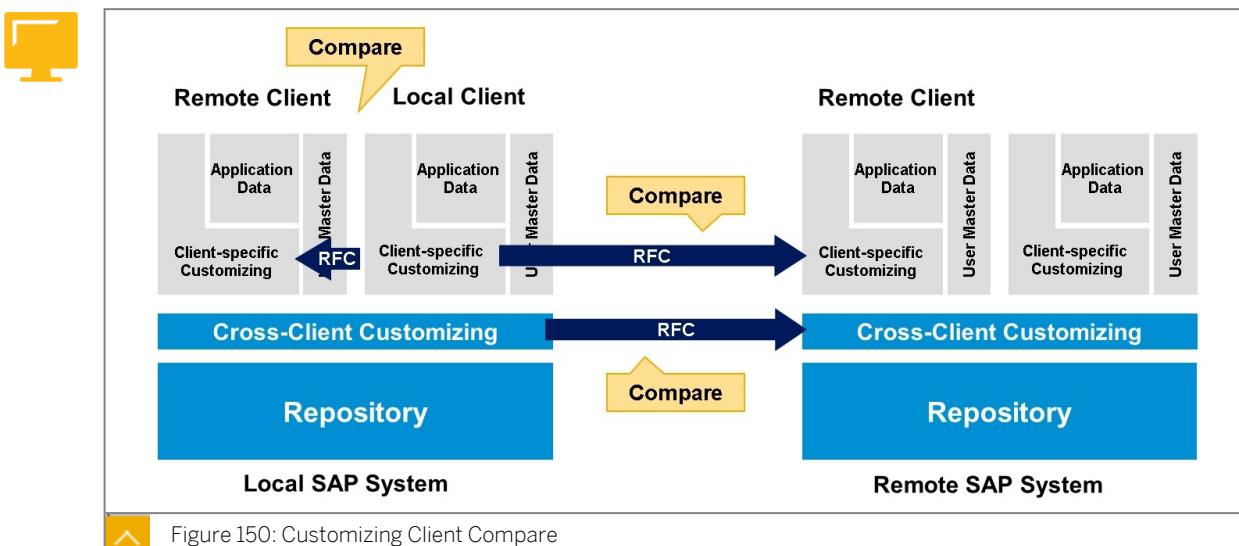


Figure 150: Customizing Client Compare

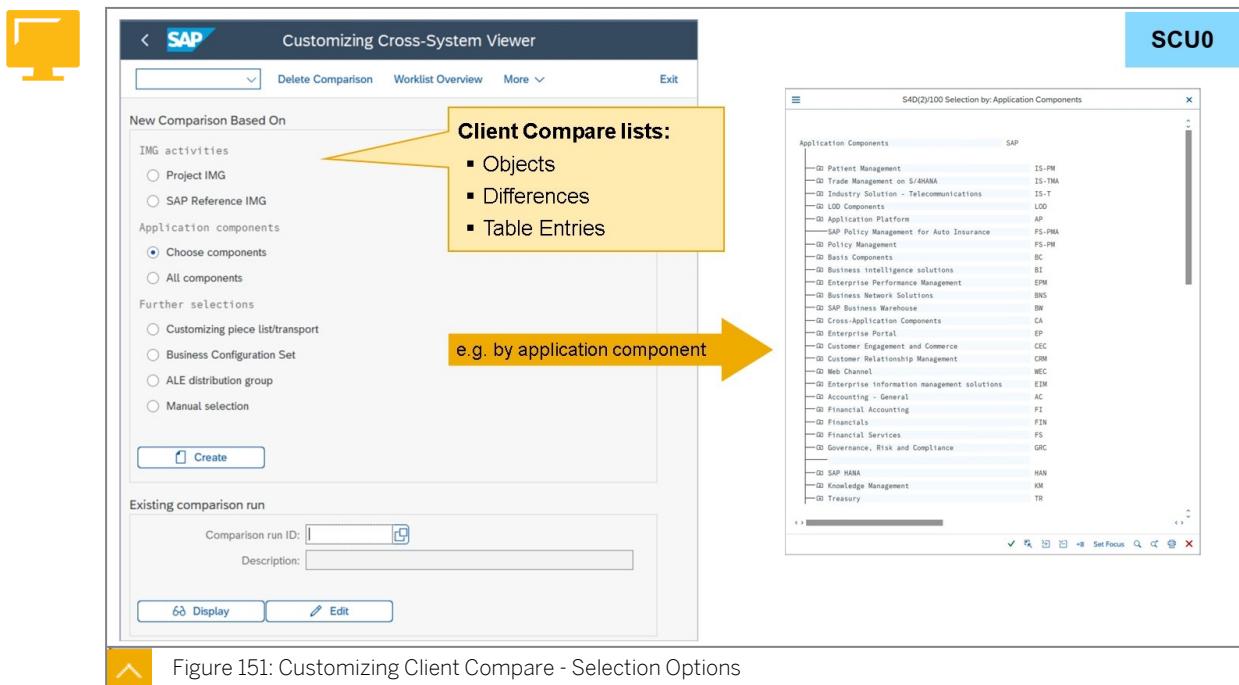
You can use the client compare for example to:

- Compare a client with a reference client, such as a customer-owned client or the SAP delivery client 000. This is especially useful after an SAP system upgrade or a language import, since only client 000 is provided with data from tables belonging to *delivery class C*.
- Compare clients during a roll-out scenario. For example, if subsidiaries want to adjust their customizing with respect to the reference customizing of a master client.
- Compare cross-client customizing before combining different clients into one SAP system in a roll-in scenario. For example, if subsidiaries want to receive customizing changes from the reference client which might be the master client of their parent organization.

### Customizing Client Compare with Transaction SCU0

To use the compare function, choose transaction SCU0.

To select the objects to be compared in a client compare, you can use a project-oriented approach using the *Project Implementation Guide* (Project IMG) to define the objects. You can also select from application components' customizing object lists and from transport requests, or select the objects manually.



To start a client compare, use transaction SCU0 in background processing. From the initial screen, select what the comparison will be based on and choose *Create* (see the figure above). In the dialog boxes that appear, make the appropriate entries. The client compare can be restricted to, for example, client-specific data.

Transaction SCU0 first displays an overview of the tables belonging to views for the selected IMG, application component or transport request. Next, the comparison itself is performed. The SAP system creates a list of differences and indicates whether these differences are encountered in the table structure or in the table contents (see the figure "Customizing Client Compare – Difference List").

To display the differing table entries, select an object. This enables you to perform a detailed comparison.



Note:

The comparison client is generally an external environment, so you must access it through an RFC destination.



Hint:

For information on client compares, see also SAP Note [91096 – Table Compare: Info about Cust. Cross System Check](#).



The screenshot shows the SAP Customizing Cross-System Viewer in Change mode. The main window displays a difference list for object V\_T005\_BAS. The table has columns for Stat., Object, Description, Comp, Entries, and Component. A callout box labeled "Notes on Performance" points to a separate window titled "S4D(1)/100 Overview: Comparison" which contains notes on performance.

Stat.	Object	Description	Comp	Entries	Entries	Entries	Entries	Component
	Subobject		LogonSys	Only in	Changed	Only in		
	Subobject		Total	LogonSys		CompSys.		
<input checked="" type="checkbox"/>	V_T005_BAS	V Global Parameters of Countries (New Dimension)		255	1	0	0	BC-SRV-ADR

Processing status

Comparison status

Notes on Performance

Restrict selection?

Figure 152: Customizing Client Compare – Difference List

The result of each comparison run is an overview of existing differences between the log-on client and the comparison client. This overview serves as a starting point for subsequent processing of differences. The comparison runs are stored in a work list or difference list.

### Enhancements as of SAP\_BASIS 754

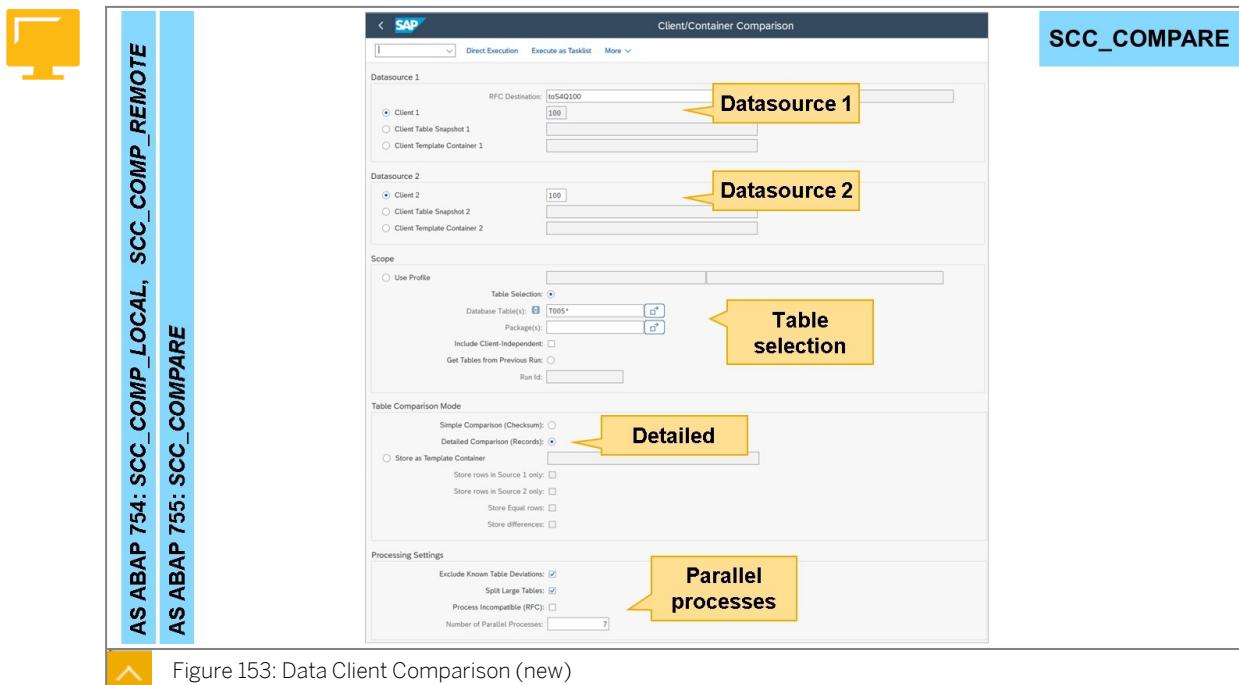
Starting with SAP\_BASIS 754, the client copy procedures have been changed and enhanced.

To set up a new client by client copy, the structures of all tables to be copied must be identical in the source and the target system. To perform a consistency check, you can perform a cross-table comparison with transaction `SCC_COMP_LOCAL` (if both clients are based on the same SAP system) or a cross-system check with transaction `SCC_COMP_REMOTE`. With SAP\_BASIS 754 SP 03, the new transaction `SCC_COMPARE` replaces transactions `SCC_COMP_LOCAL` and `SCC_COMP_REMOTE`.

To compare tables, proceed as follows:

1. Call transaction `SCC_COMP_LOCAL` / `SCC_COMP_REMOTE` / `SCC_COMPARE`.
2. Enter the clients that you want to compare and an RFC destination (in case of a remote comparison).
3. Select the tables that you want to compare, such as:

- Explicit table selection by table name or package.
  - Explicit selection of tables using the time stamp of a previous client copy.
4. Define the table comparison mode (*Simple* or *Detailed comparison*).
  5. Adjust the processing settings, for example, *Exclude Known Table Deviations* or *Split Large Tables*.
  6. Specify the number of parallel processes.
  7. Decide whether you want to execute the client copy as batch job (*Execute as Tasklist*) or start it immediately (*Direct Execution*).



After running the comparison, you can check the result (see the figure “Data Client Comparison – Result (new)”). The log provides *Log Header* information, *Comparison Results*, a list of *Compared Tables* together with the result of the comparison, *Runtime* of the Exit-Methods, and the selected *Options*.

The screenshot shows the SAP Data Client Comparison Result screen. At the top right, there is a yellow callout box labeled "Detailed Comparison: Result". Below the header, the transaction code "SCC3" is visible. The main area displays a table comparing "Datasource 1" and "Datasource 2". A vertical blue bar on the left is labeled "AS ABAP 754 and above". A yellow callout box labeled "Difference" points to the first row of the table, which shows a red dot next to "T005" and a green square next to "T005A". Another yellow callout box labeled "Details" points to the right side of the table where runtime and ms values are listed.

Exception	Table Name	Delivery Class	Application	Result	Rows (100)	Rows (100)	Different Rows	Equal Rows	Rows (Client 100 only)	Rows (Client 100 only)	Runtime (ms)
	T005	C	BC-SRV-ADR	Differences found	254	255	0	254	0	1	57 ms
	T005A	E	BC-SRV-ADR	Data Equal	29	29	0	29	0	0	5 ms
			BC-SRV-ADR	Data Equal	240	240	0	240	0	0	5 ms
			BC-SRV-ADR	Data Equal	641	641	0	641	0	0	13 ms
	T005F	C	BC-SRV-ADR	Data Equal	6.922	6.922	0	6.922	0	0	53 ms
	T005G	C	CA-GTF-TS	Data Equal	333	333	0	333	0	0	12 ms
	T005H	C	CA-GTF-TS	Data Equal	3.996	3.996	0	3.996	0	0	36 ms
	T005I	C	BC-SRV-COM	Data Equal	0	0	0	0	0	0	0 ms

Figure 154: Data Client Comparison – Result (new)

**Note:**

You can access the client comparison result of all runs from transaction SCC3.

## Client Adjustment

For each compared object in a client comparison run, the objects compared are listed along with a *Description*. The most important information is the status indicators. The *comparison status* indicates the existence and nature of any differences and is set automatically.

In transaction SCU0, the *Processing status* allows you to distinguish between the objects that have already been processed, that is, have been made identical in both clients, and those that have not. This kind of processing is also referred to as **adjusting** the respective object. The processing status is indicated by a traffic light, where red indicates not processed (*Open*), yellow indicates the objects in both clients are *In process* of being made identical, and green indicates the completion of this adjustment process (*Completed*) – see figure “Customizing Client Compare – Difference List” above.



**Hint:**

For an explanation of the possible statuses under *Comparison status* and *Process status*, choose the *Legend* icon.

For a detailed analysis of a comparison run, choose *Comparison*. This leads to a screen similar to the one shown in the following figure.

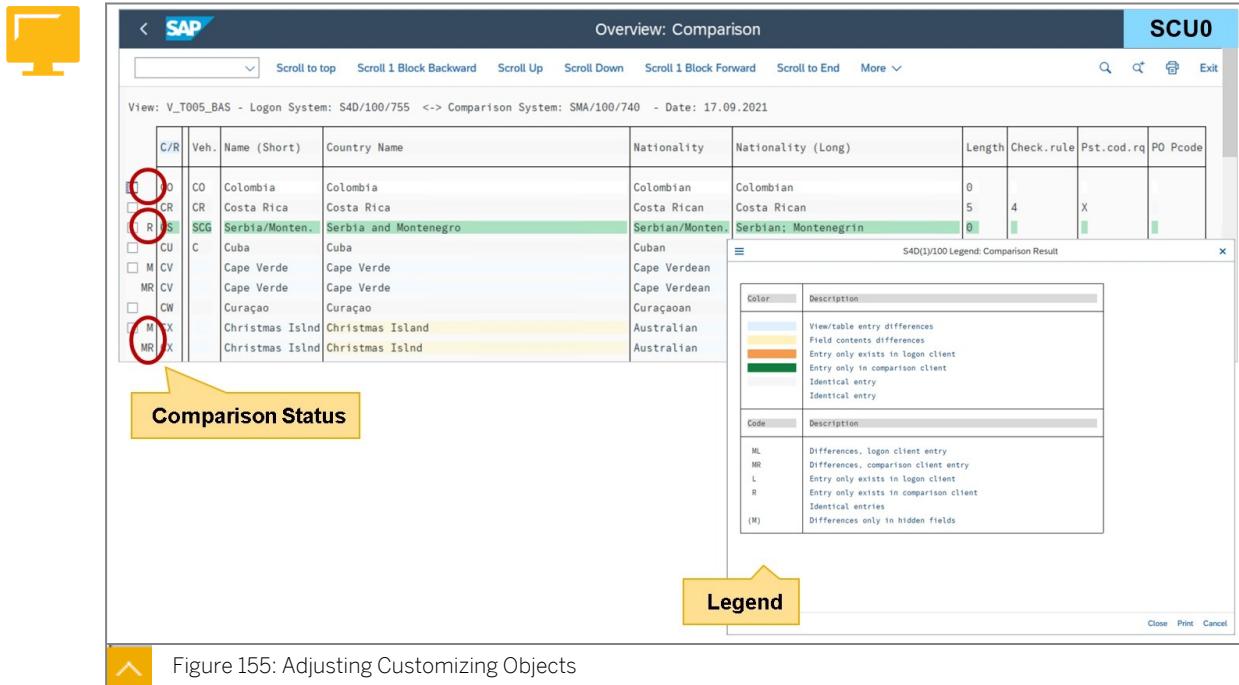


Figure 155: Adjusting Customizing Objects

You can adjust one object at a time. The objects that can be adjusted are some of the tables and views that can be maintained using transaction SM30. You can also use the SAP Reference IMG (transaction SPRO) or a Project IMG. All other objects can only be compared.

To perform an adjustment, from the Comparison screen of transaction SCU0, choose *Edit → Interact. Copy*. The Overview Adjustment screen is displayed, showing the details of the differences between the two clients, record by record. To the left of each record in this adjustment work list there is the comparison status, which indicates whether or not each respective entry in the record exists in the comparison client and the log-on client (see the figure above).



#### Caution:

Be careful when adjusting customizing. Customizing usually is related to other customizing. You can easily create inconsistencies when copying customizing from the source client without caring for related customizing in the target client.

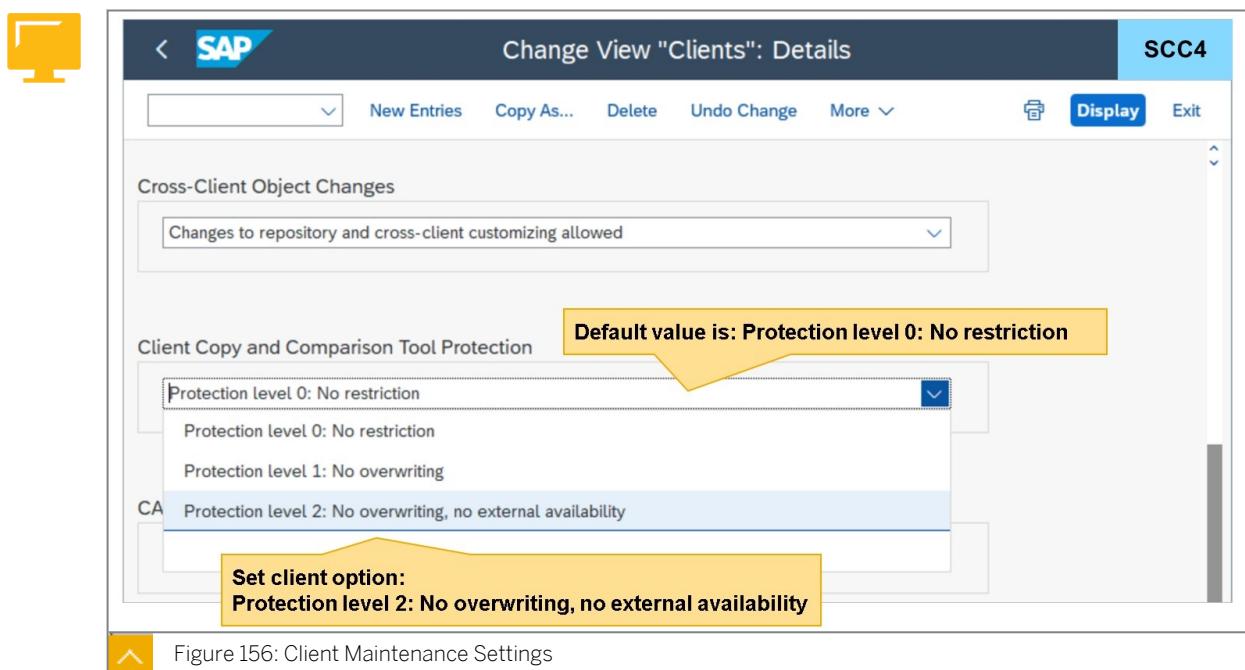


Figure 156: Client Maintenance Settings

The client option *Protection level* regarding client copy and client comparison can be set to prevent a client from being overwritten by the client copy or from a client compare and client adjustment. It can also be set to ensure that sensitive data is not viewable from another client during a client compare.

To set this option, choose transaction **SCC4** and select the client in question in change mode. Select a protection level:

#### **Protection level 1: No overwriting**

This protection level ensures that the client cannot be overwritten by the client copy tools. This setting should be used if customizing is being performed in this client or the client contains critical settings or data that should not be overwritten.

#### **Protection level 2: No overwriting, no external availability**

This protection level also protects the client against overwriting. In addition, it protects the client against read access from another client during a client copy or customizing comparison. This setting should be used if the client contains sensitive data, for example, all production clients should use this setting.

## **Comparing the Repository**

There are several reasons why you might want to compare the repository of two SAP systems. For example, in the one SAP system business processes are working in a different way than in a second SAP system, although the business processes should work the same. You have already compared the corresponding customizing but didn't find the difference. This might be because of differences in the repository.

Another reason for comparing the repository would be that, before performing a remote client copy, you should compare the repository of the source and the target SAP system. If the two repositories differ too much, data will be corrupted after the remote client copy in the target system from point of view of the repository of the target system.

With transaction **SREPO**, you can compare the repository of two SAP systems. **SREPO** performs a mass comparison of the repository objects between two SAP systems.

Prerequisites for a comparison are:

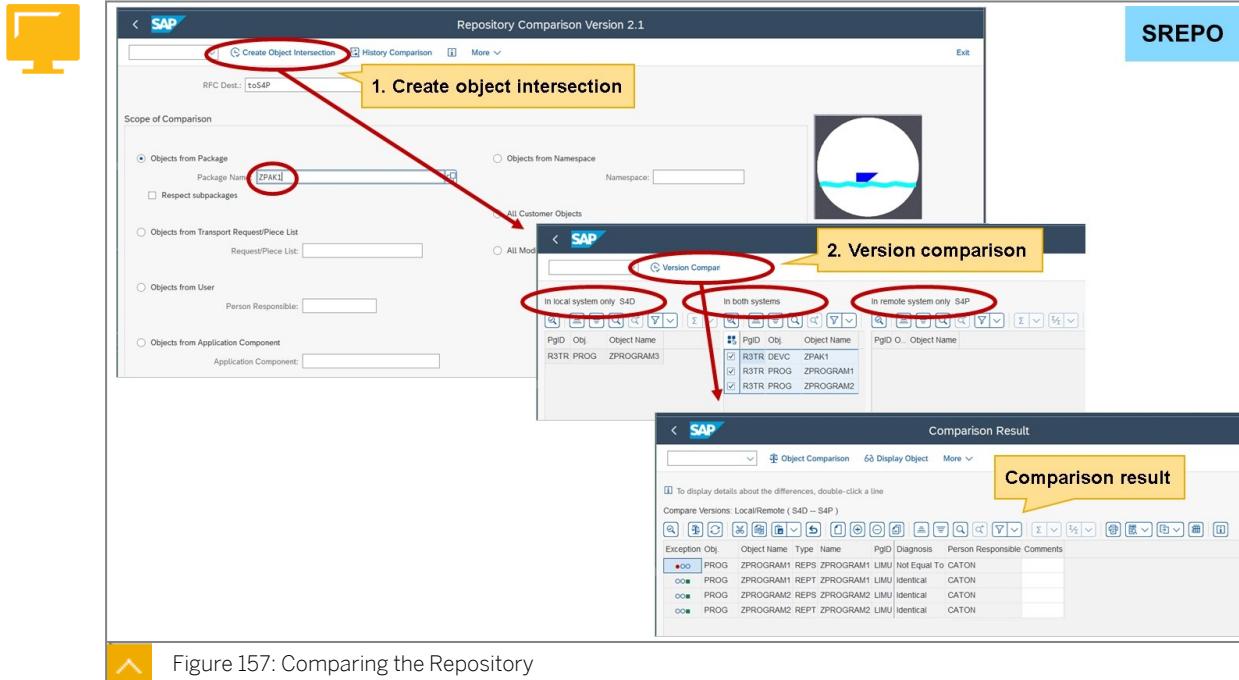
- Both SAP systems must have the same release and SAP Support Package level
- The package **SCTS\_COMP\_SYS** must exist in both SAP systems
- An RFC destination must be configured (transaction SM59)

You can compare the following sets of objects:

- Objects from a package
- Objects in a transport request or piece list
- Objects of a user
- Objects in an application component
- Objects in a name space
- All customer objects
- All modified SAP objects
- Freely selected objects

The comparison is done in two steps (see the figure “Comparing the Repository”):

1. Intersection is created. The objects are selected and their existence in the two SAP systems is checked.
2. Versions are compared. The objects are compared.



The object list contains information about the version comparison, with indicators as follows:

- **Green** indicates that the object is identical in both SAP systems,

- *Yellow* indicates that the objects can't be compared directly. This is because certain objects can't be compared by transaction SREPO.
- *Red* indicates that the objects are different.

More details and instructions are available in SAP Note [1529391](#) – *Revision of comparison tool SREPO*.



## LESSON SUMMARY

You should now be able to:

- Perform a Client Comparison
- Outline the idea of client adjustments with respect to Customizing
- Describe how to compare two repositories



## Learning Assessment

1. You want to perform a remote client copy of client 100 of your development system to client 200 of your quality assurance system, using transaction scc9. You want to use 16 parallel processes. Which of the following statements are correct regarding the client copy?

*Choose the correct answers.*

- A An RFC server group (RZ12) must exist in the quality assurance system
- B An RFC server group (RZ12) must exist in the development system
- C You start the remote client copy (scc9) in the quality assurance system
- D You start the remote client copy (scc9) in the development system
- E An RFC destination (SM59) must exist in the quality assurance system, pointing to client 100 of the development system
- F An RFC destination (SM59) must exist in the development system, pointing to client 200 of the quality assurance system

2. If you want to delete client 001 via transaction SCC5 in your SAP ECC development system using 16 parallel processes, you need an RFC server group (RZ12) in the development system for this.

*Determine whether this statement is true or false.*

- True
- False

3. You want to perform a client compare with the *Customizing Cross-System Viewer* (transaction SCU0) from client 100 of the development system to client 100 of the quality assurance system. With the help of this tool, you can also compare the repository doing this.

*Determine whether this statement is true or false.*

- True
- False

4. You are logged on in client 100 of the development system DEV and want to perform a client compare with the *Customizing Cross-System Viewer* (SCU0) with client 100 of the quality assurance system QAS. You do not want to adjust, only compare. Which of the following client change options (SCC4) would make the client compare impossible?

Choose the correct answer.

- A In QAS, client 100: Protection level 2: No overwriting, no external availability
- B In QAS, client 100: Protection level 1: No overwriting
- C In DEV, client 100: Protection level 2: No overwriting, no external availability
- D In DEV, client 100: Protection level 1: No overwriting

## Learning Assessment - Answers

1. You want to perform a remote client copy of client 100 of your development system to client 200 of your quality assurance system, using transaction scc9. You want to use 16 parallel processes. Which of the following statements are correct regarding the client copy?

*Choose the correct answers.*

- A An RFC server group (RZ12) must exist in the quality assurance system
- B An RFC server group (RZ12) must exist in the development system
- C You start the remote client copy (scc9) in the quality assurance system
- D You start the remote client copy (scc9) in the development system
- E An RFC destination (SM59) must exist in the quality assurance system, pointing to client 100 of the development system
- F An RFC destination (SM59) must exist in the development system, pointing to client 200 of the quality assurance system

You are correct! An RFC server group (RZ12) must exist in the quality assurance system, you start the remote client copy (scc9) in the quality assurance system, and an RFC destination (SM59) must exist in the quality assurance system, pointing to client 100 of the development system. You do not need an RFC server group (RZ12) in the development system, you do not start the remote client copy (scc9) in the development system, and no RFC destination (SM59) needs to exist in the development system, pointing to client 200 of the quality assurance system.

2. If you want to delete client 001 via transaction scc5 in your SAP ECC development system using 16 parallel processes, you need an RFC server group (RZ12) in the development system for this.

*Determine whether this statement is true or false.*

- True
- False

You are correct! You need an RFC server group (RZ12) in the development system if you want to run parallel processes in transaction SCC5.

3. You want to perform a client compare with the *Customizing Cross-System Viewer* (transaction SCU0) from client 100 of the development system to client 100 of the quality assurance system. With the help of this tool, you can also compare the repository doing this.

*Determine whether this statement is true or false.*

- True  
 False

You are correct! A client compare with transaction SCU0 is designed only for comparing customizing settings, no repository comparison. To compare the repository, you can use *Repository Comparison* (transaction SREPO).

4. You are logged on in client 100 of the development system DEV and want to perform a client compare with the *Customizing Cross-System Viewer* (SCU0) with client 100 of the quality assurance system QAS. You do not want to adjust, only compare. Which of the following client change options (SCC4) would make the client compare impossible?

*Choose the correct answer.*

- A In QAS, client 100: Protection level 2: No overwriting, no external availability  
 B In QAS, client 100: Protection level 1: No overwriting  
 C In DEV, client 100: Protection level 2: No overwriting, no external availability  
 D In DEV, client 100: Protection level 1: No overwriting

You are correct! In QAS, client 100: Protection level 2: No overwriting, no external availability would make the client comparison impossible. The other settings would not influence this client compare.

# UNIT 7

# SAP Note Assistant, SAP Support Packages, SAP System Upgrades, SAP S/4HANA Conversion

## Lesson 1

SAP Note Assistant

275

## Lesson 2

SAP Support Packages

293

## Lesson 3

SAP System Upgrade and SAP S/4HANA Conversion

309

## UNIT OBJECTIVES

- List the advantages of automatically implementing SAP Notes
- List options to find relevant SAP Notes and explain how to distribute them in your system landscape
- Implement SAP Notes with SAP Note Assistant
- Describe the difference between SAP Notes with correction instructions and SAP Notes with transport-based correction instructions
- Outlook: Find SAP Notes automatically
- Describe the prerequisites for requesting an SAP Support Package Stack
- Import an SAP Support Package that requires a modification adjustment
- Explain the difference between SAP Support Package Stacks and Feature Package Stacks
- List capabilities of SUM
- Describe important steps for planning and performing an SAP System Upgrade or an SAP S/4HANA Conversion
- Outline the idea of business functions



# SAP Note Assistant

## LESSON OVERVIEW

In this lesson, you will learn about SAP Notes and the advantages of implementing an SAP Note automatically with the *SAP Note Assistant*. You will also learn about the goals, functions, and the features of the *SAP Note Assistant*. Additionally, you will learn about how to reset an SAP Note implementation and about the preconditions for handling SAP Notes. Finally, there is a section on *Transport-Based Correction Instructions* (TCI).

## Business Example

As the SAP system administrator, you should know the tools used for applying error fixes, legal changes, security fixes, and new functions to the SAP software components.



## LESSON OBJECTIVES

After completing this lesson, you will be able to:

- List the advantages of automatically implementing SAP Notes
- List options to find relevant SAP Notes and explain how to distribute them in your system landscape
- Implement SAP Notes with SAP Note Assistant
- Describe the difference between SAP Notes with correction instructions and SAP Notes with transport-based correction instructions
- Outlook: Find SAP Notes automatically

## Introduction to SAP Note Assistant

There are SAP Notes for different reasons. SAP Notes can provide information of any kind for customers and partners of SAP, but often also contain solutions for known issues within an SAP system.



### Note:

When talking about SAP Notes in this lesson, only those SAP Notes are referred to that contain a change of a repository object. Not all changes can be handled with the *SAP Note Assistant*. In this case, manual steps are necessary.

To overcome most of these limitations, the *Transport-Based Corrections Instruction* (TCI) has been introduced that allows for example also the handling of dictionary objects. For details, see the last part of this lesson.



### Example: SAP S/4HANA Server

**SAP Feature Package / Support Package Stack:**  
Targeted combination of SAP Support Packages



#### SAP Support Package:

Group of objects  
for one software component

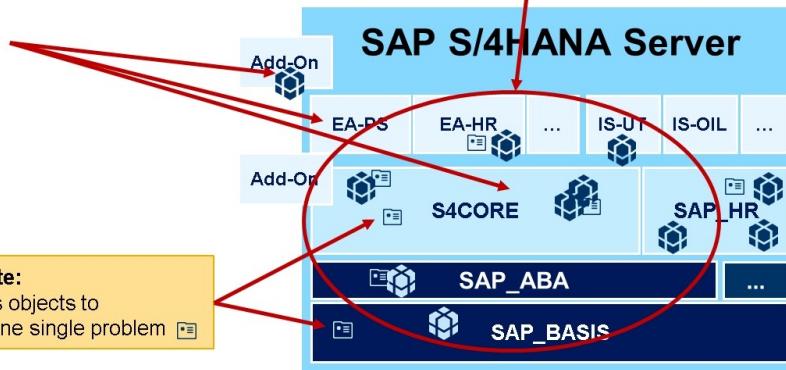


Figure 158: What is an SAP Note?

They include a description of the following:

- The symptoms of the issue
- The cause of the issue
- The solution (for example, descriptions of how to correct the source code)
- The SAP release and SAP Support Package status for which the SAP Note is valid
- Links to SAP Support Packages that correct the issue



#### Caution:

Carefully read the SAP Note before you implement it. The SAP Note can contain prerequisites, interactions, and references to post processing activities, such as making changes to a table structure that you must consider when you implement it.

Before making any modification based on SAP Notes, verify that the SAP Note is applicable to your SAP system release and that the symptoms in the SAP Note are the same as the symptoms apparent in your SAP system. If you're not sure, contact SAP.

All modifications of SAP objects should be performed in the development system, tested in the quality assurance system, and distributed to the production system only after thorough testing and verification.



#### Hint:

Because modifications often lead to the need for a modification adjustment during a later SAP system upgrade or when applying SAP Support Packages, document the modification changes to help to speed up the modification adjustment process. When documenting, be sure to include the SAP Note number and SAP system release dependencies.

The automatic implementation of SAP Notes contributes to enhanced quality, because it reduces errors due to manual implementation. Manually implementing an SAP Note can lead to high costs because of the resources, including personnel and time, needed to implement the SAP Note.

If an SAP Note is implemented incorrectly, the quality of the SAP system suffers.

### Configuration Steps for SAP Note Assistant

Before you can start using SAP Note Assistant, you need to make a number of settings in your system:

- Define the system change option (transaction SE06). For the software components in which you want to implement the SAP Notes, set the *Status* to *restricted modifiability* or *Modifiable*. In addition, for the general SAP name range, set the *Status* to *Modifiable*.
- There are different ways in which you can load SAP Notes into your system. You can download SAP Notes directly using an HTTP destination to SAP, or you can use SAP Note Assistant to upload SAP Notes that you have already loaded from SAP Support Portal, or you can use the ABAP Download Service. If you want to download an SAP Note with HTTP, you first need to create two RFC destinations of type H (recommendation: SAP-SUPPORT\_PORTAL) and G (recommendation: SAP-SUPPORT\_NOTE\_DOWNLOAD) for connecting to SAP Support Portal and for SAP Note download.

**Note:**

The use of RFC destination SAPOSS (of type 3) is no longer supported. See SAP Note [2923799 – Final Shutdown of RFC Connections From Customer Systems to SAP](#) for more information. For the download procedure, you need to choose *HTTP Protocol* or *Download Service Application*.

- *HTTP Protocol* is available as of SAP\_BASIS 740.
- *ABAP Download Service* is available for SAP\_BASIS release 700 onwards. Any ABAP system having the download service (for example the SAP Solution Manager 7.2 System) can be used as download system. For more information concerning this scenario, see SAP Note [2554853 – SAP NetWeaver download service for SAP Notes](#).

Because there is a risk that the SAP Notes files can be maliciously modified and customers can unknowingly upload the maliciously modified SAP Notes files into their ABAP systems, SAP delivers all SAP Notes with digital signature to protect SAP Notes files with increased authenticity and improved security.

To consume digitally signed SAP Notes from SAP ONE Support Launchpad (<https://launchpad.support.sap.com>), you need to follow SAP Note [2836302 – Automated guided steps for enabling Note Assistant for TCI and Digitally Signed SAP Notes](#) and the steps listed in the PDFs attached to this SAP Note and to SAP Note [2508268 – Download of Digitally Signed SAP Notes in SNOTE](#).

SAP Note [2827658 – Automated Configuration of new Support Backbone Communication - Update 02](#) provides a task list SAP\_BASIS\_CONFIG\_OSS\_COMM to configure and check the connection to SAP new Support Backbone.

For selecting and customizing the procedure, you can also use SAP Reference IMG (transaction SPRO), area SAP Customizing Implementation Guide → ABAP Platform → Application Server → Basis Services → SNOTE → Download of SAP Notes in SNOTE, or run report RCWB\_SNOTE\_DWNLD\_PROC\_CONFIG. This is a one-time set up. If required, you can change the settings in this report at any given point in time.

For more information, see SAP Note [2537133 – FAQ - Digitally Signed SAP Notes](#).

- To view the Note text correctly in SAP Note Assistant, install Adobe Reader Version 7.0.1 or higher on the computer from which transaction SNOTE is started.
- Before you implement notes with SAP Note Assistant, you should upgrade to the latest version of SAP Note Assistant. Follow SAP Note [1668882 – Note Assistant: Important notes for SAP\\_BASIS 730,731,740,750,751,752,753,754, 755, 756](#) for details.

### **SAP Note Assistant Goals**

The goals of SAP Note Assistant are as follows:

- Automatically implement SAP Notes which handle repository object source code corrections
- Handle dependencies with other SAP Notes, SAP Support Packages, and modifications

- Display all SAP Notes implemented in the SAP system
- Handle updates (SAP Support Packages, SAP enhancement packages, SAP system upgrades)
- Correct only a single problem, not a replacement for SAP Support Packages

The *SAP Note Assistant* includes the following functions:

- *Searching*: You can search valid SAP Notes for your system and download them.
- *Reporting*: You can display an overview of all existing SAP Notes and their processing status, and all implemented source code corrections.
- *Project Administration*: You can assign SAP Notes to users who then specify a processing status for the SAP Notes. The SAP system informs you if an SAP Note has an inconsistent status.
- *Logging*: The SAP system logs all processing steps.
- *Corrections*: You can implement the source code corrections (correction instructions) contained in SAP Notes into your SAP system automatically.
- *Integration*: When you apply SAP Support Packages or upgrade your SAP system, the SAP system automatically recognizes which SAP Notes have already been completed by an imported SAP Support Package or SAP system upgrade, and which correction instructions still need to be implemented.

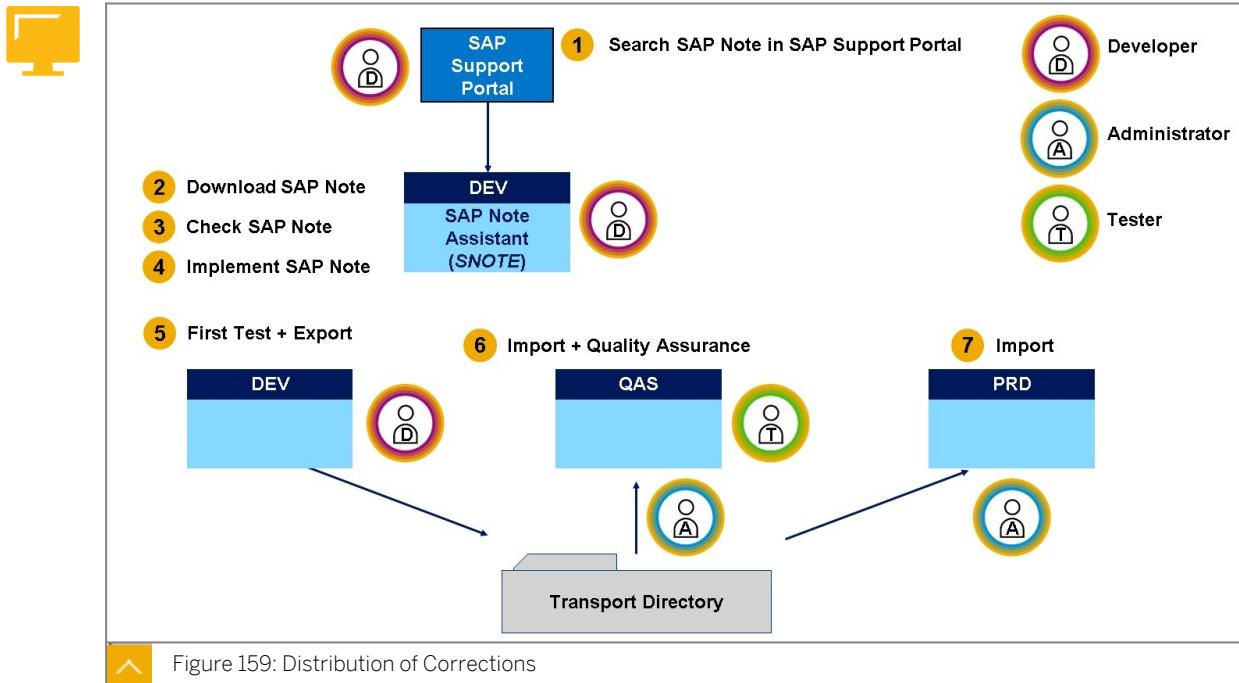
You can apply the SAP Notes in different ways with *SAP Note Assistant*:

1. If a connection to the SAP Support Portal is configured in the SAP system (see previous section), you can directly download the SAP Notes to your SAP system.
2. You can also download the SAP Notes from SAP Support Portal (for example, with the *Download Manager*) to your PC and then upload them to the SAP system.

The upload then takes place from within the *SAP Note Assistant*.

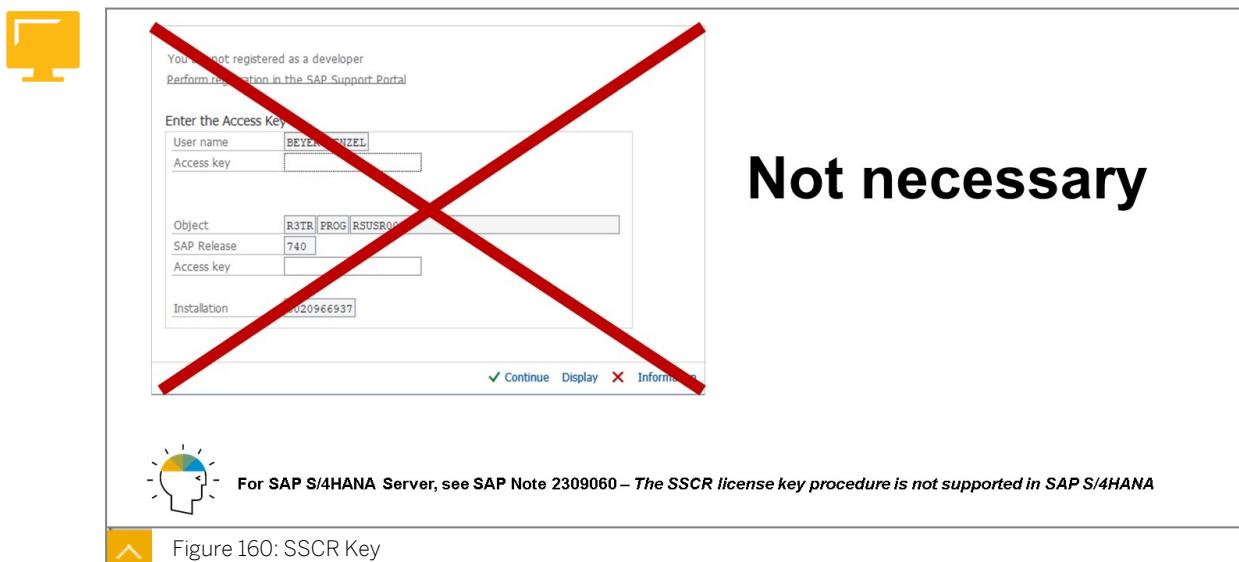
## Finding and Distribution of SAP Notes

Once the required SAP Notes have been identified, they have to be applied and distributed among the SAP system landscape.



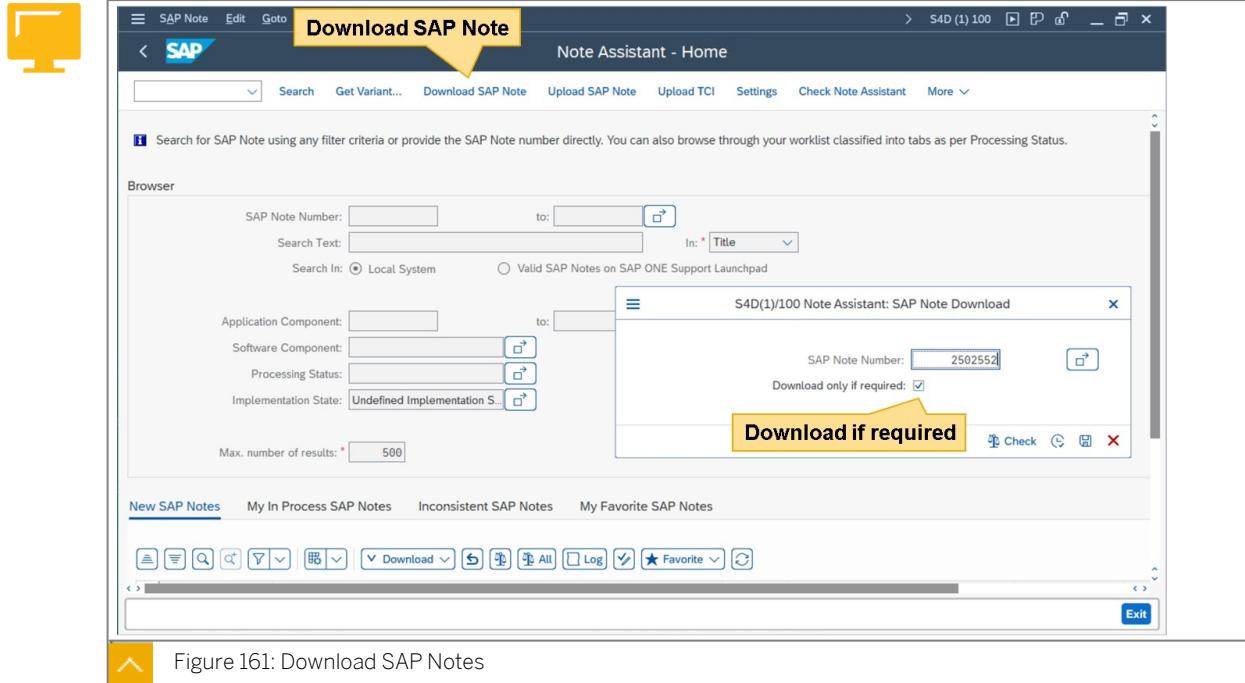
Follow these steps to apply SAP Notes in your SAP system landscape:

1. Implement the SAP Notes with the SAP Note Assistant in the development system (DEV).
2. Transport the SAP Notes from the development system to the quality assurance system (QAS) with the help of a transport request that contains the corrected objects. Don't implement the SAP Note with the SAP Note Assistant in QAS. Now you can test these corrections in QAS.
3. Import the SAP Notes into the production system (PRD). Use the transport request that contains the corrected objects. Do not implement the SAP Note with the SAP Note Assistant in PRD.



Changes to SAP repository objects that are introduced using the SAP Note Assistant don't require an SSCR key.

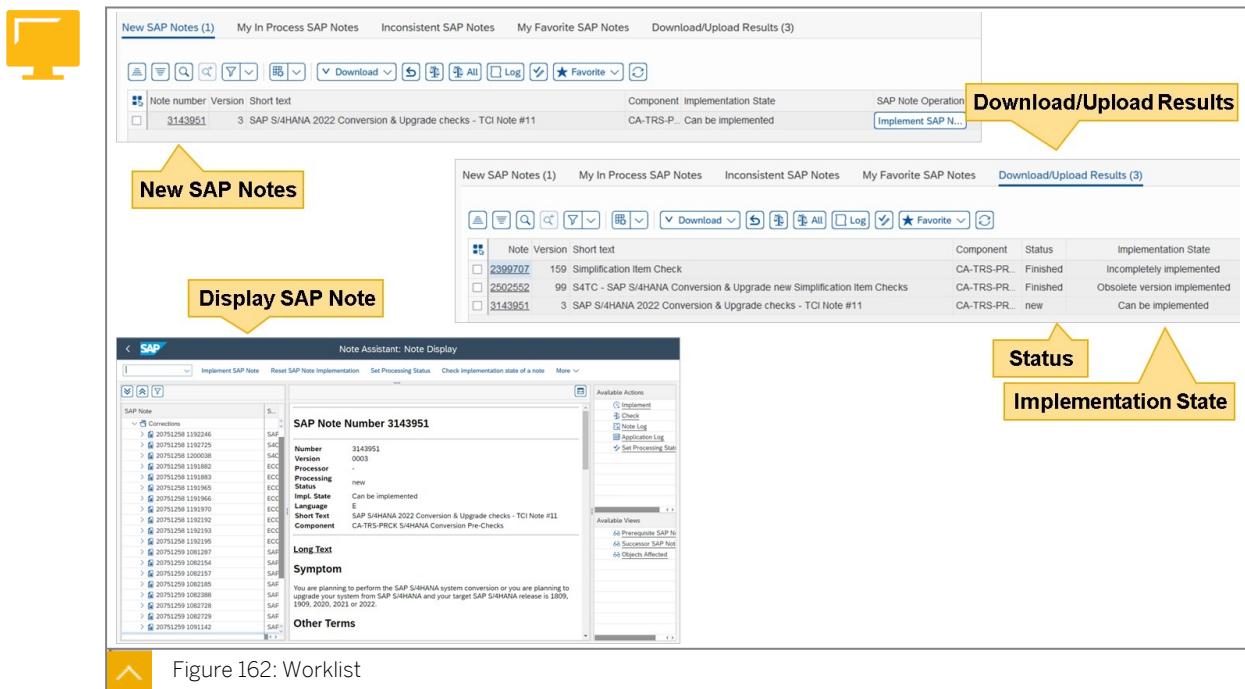
## Handling the SAP Note Assistant



SAP Note [3051466](#) – *Note Assistant Made Easy : Revamped Note Assistant* completely revamps SAP Note Assistant with an aim of making it more simplified, intuitive, customizable and scalable, along with providing a new user experience. Among others, it offers a new *Home* screen, where the user worklist and the *Note Browser* is integrated in a single screen. In addition, the worklist and search results are displayed in tabs to improve usability (see the figure above)

To download SAP Notes, open SAP Note Assistant (transaction SNOTE) and choose *Goto* → *Download SAP Note* from the menu or use the corresponding button in the application toolbar.

Another option is to download an SAP Note from the SAP Support Portal to your front end and then choose *Goto* → *Upload SAP Note* from the menu in transaction SNOTE to load the SAP Note into the SAP Note Assistant.



SAP Note Assistant is started with transaction SNOTE. The *Worklist* (lower part of the *Home* screen) is your personal view on the SAP Notes.

Inconsistent and new SAP Notes appear in the worklists of every user. Under the category *My In Process SAP Notes*, you only see the SAP Notes that you process.

- From your worklist, you can perform all actions such as download, implement, and display SAP Notes.
- You can check the implementation status of an SAP Note.
- You can classify an SAP Note or set the processing status:
  - If the SAP Note doesn't correspond to your situation, you can set it to *Not Relevant*.
  - After implementing an SAP Note, you can set the status to *Completed*.

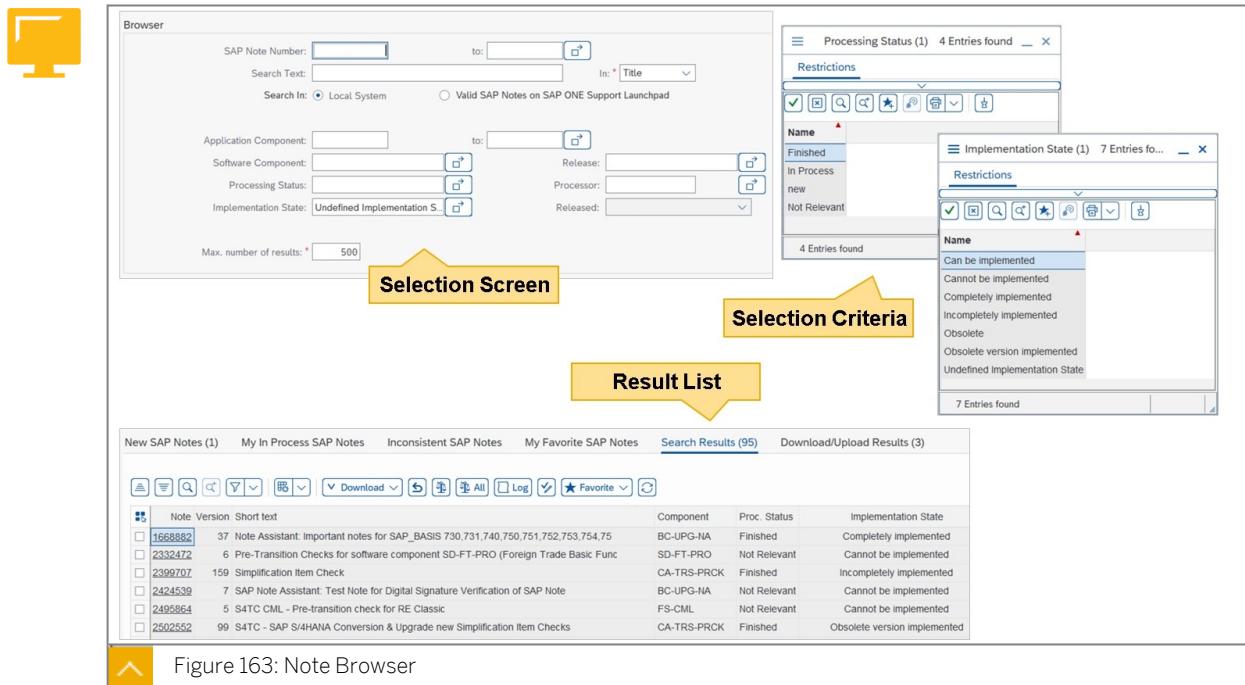


Figure 163: Note Browser

You can use the *Note Browser* to find any SAP Note that is available in the SAP system. Select an SAP Note in your worklist and then choose *SAP Note → Display SAP Note* from the menu to read through the SAP Note and see the source code changes.

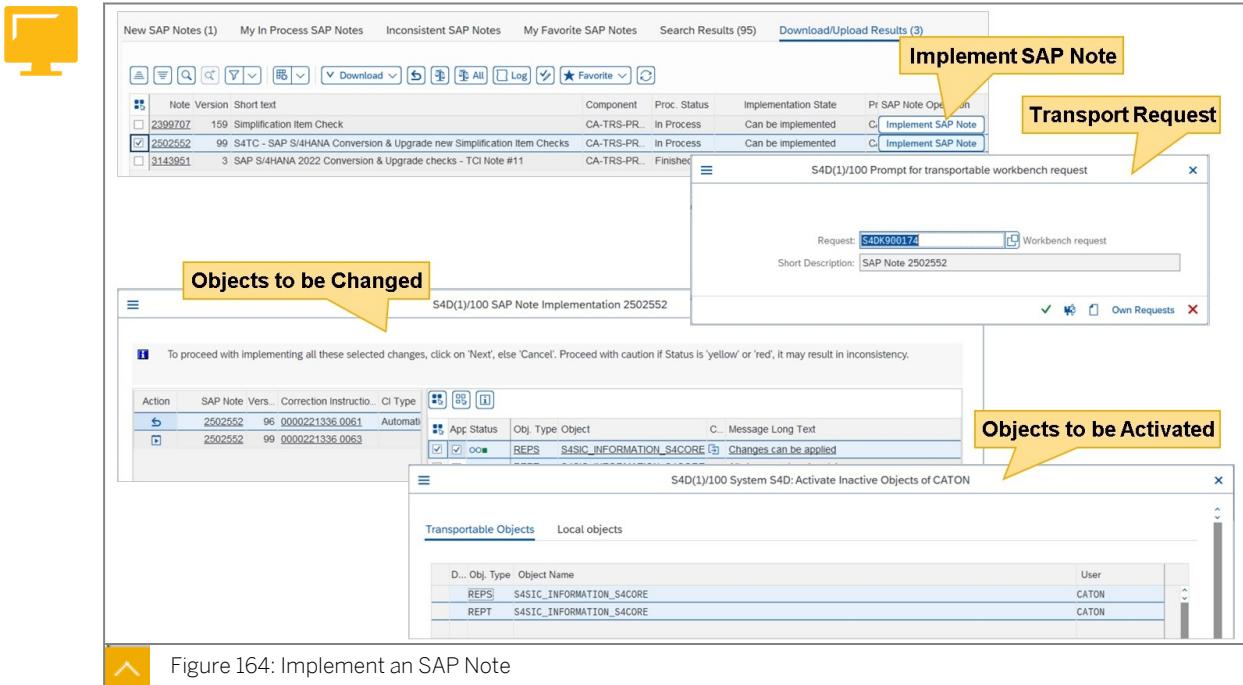


#### Note:

To search for SAP Notes in the SAP ONE Support Launchpad, the *ANST Note WebService* has to be implemented. For details, see SAP Note [2730525 – Consuming the Note Search Webservice](#).

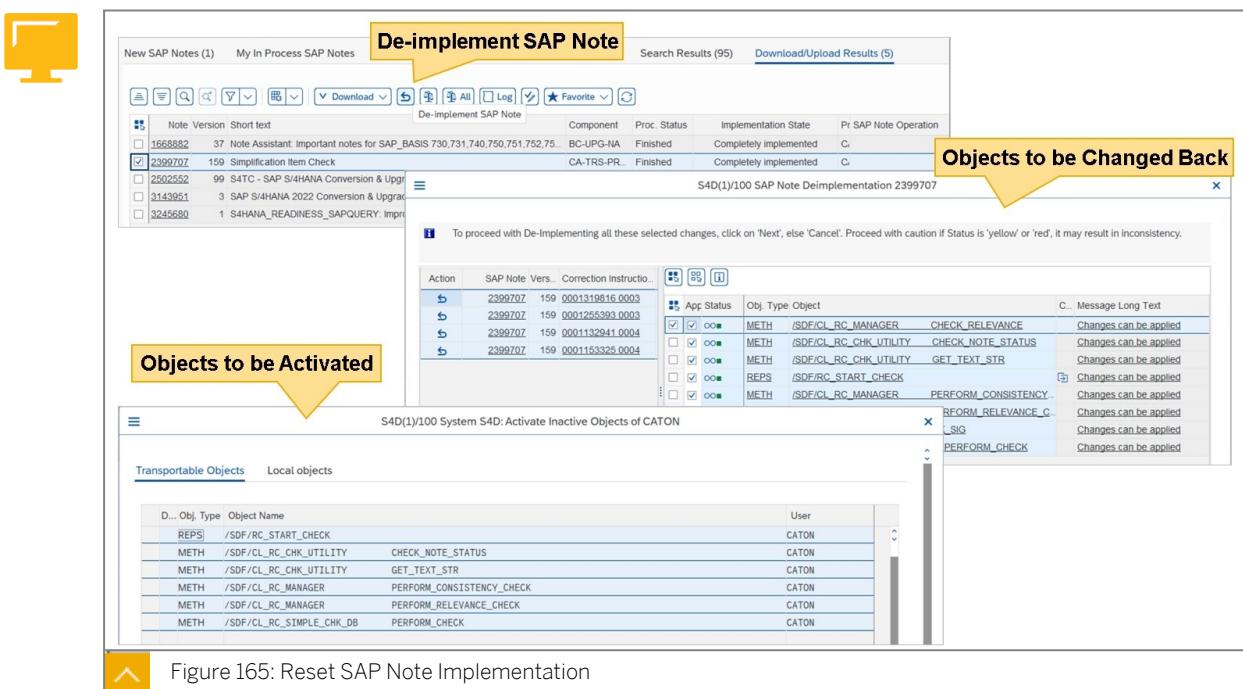
To reduce the number of SAP Notes that are displayed in the results list, you can specify various criteria in the selection screen (processing status, application component, implementation status).

From the worklist, you can also implement the correction instructions that are contained in the SAP Note.



To implement the SAP Note, select the respective SAP Note and choose the button *Implement SAP Note*.

Traffic light indicates whether the changes can be copied, that is, whether the correction instructions can be implemented (see the figure above). Double-clicking the message provides more information.



In exceptional cases, you may want to de-implement an SAP Note that you have implemented. You can do this by choosing the button *De-implement SAP Note*.

If you do this, the changes that have been introduced with the implementation of the SAP Note will be reset and the SAP Note will be reset to the status that it had before you implemented it.

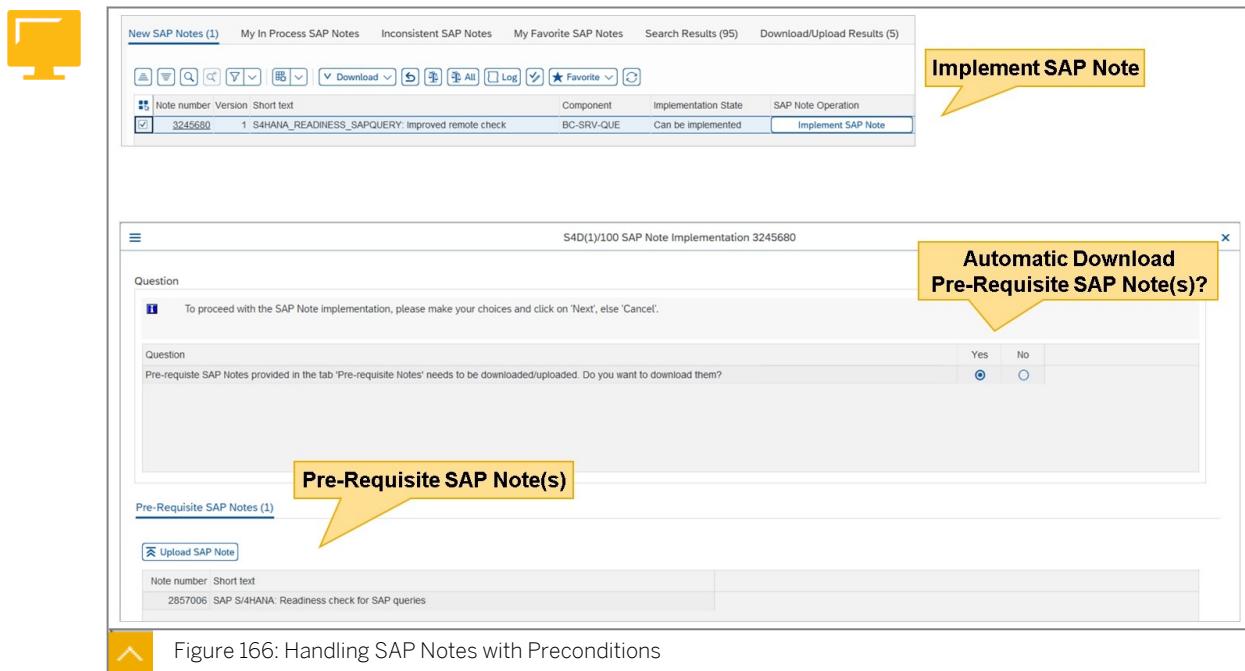


Figure 166: Handling SAP Notes with Preconditions

Sometimes, correction instructions can only be implemented if you have implemented other correction instructions before. From a technical perspective, these preconditions are included in the SAP Note as well. Because *SAP Note Assistant* strictly follows the information in the SAP Note, you must implement the required SAP Notes first.

The required SAP Notes can be automatically downloaded by *SAP Note Assistant*.



#### Note:

Make sure that you carefully read the prerequisite SAP Notes, too.

The *SAP Note Assistant* then tries to implement as many SAP Notes as possible in one step.



#### Note:

*SAP Note Assistant* is closely related to the *Modification Assistant*. From within the *SAP Note Assistant*, you can directly switch to the *Modification Browser* (transaction SE95) by choosing the menu path *Environment* → *Modification Browser*. The *Modification Browser* displays an overview of all objects that are modified or enhanced with this SAP Note.

If you have implemented corrections from SAP Notes manually before (or without) using the *SAP Note Assistant*, you must make these SAP Notes known to the *SAP Note Assistant*. Otherwise, the *SAP Note Assistant* does not know that this correction has already been implemented in your SAP system and implements it again. This may lead to inconsistencies.

For this, you should first try to simply download the SAP Note with the *SAP Note Assistant*. If this doesn't work, for example, because the SAP Note is no longer accessible, you can manually register the SAP Note using the program *SCWN\_REGISTER\_NOTES*.

Registration offers the following benefits:

- All the SAP Notes that have been implemented in your SAP system are displayed in the *Note Browser*.
- If a manually implemented SAP Note is a prerequisite for an SAP Note that you want to implement using *SAP Note Assistant*, *SAP Note Assistant* recognizes that the prerequisite SAP Note has already been implemented and doesn't prompt you to implement it again.
- For future SAP Support Packages or upgrades, the manually implemented SAP Notes are displayed during the modification adjustment in the category *SAP Note corrections*. The SAP system determines whether these SAP Notes have to be implemented again or reset to their original status.

To register the SAP Notes, the SAP system downloads the SAP Note and checks whether it is valid for your release and SAP Support Package level.



#### Hint:

The report *SCWN\_TRANSPORT\_NOTES* allows you to transport SAP Notes that can't be implemented, for example, SAP Notes that are obsolete. The use of this report may be helpful if update tools (such as *SAP Update Manager*) require some SAP Notes in the SAP system before the update can take place. For additional information, see SAP Note [1788379 – Transport of SAP NOTES](#).

#### Problems with the SAP Note Assistant:

- If implementation fails, report an incident in SAP Support Portal (using *SAP ONE Support Launchpad*) and assign it to the component of the SAP Note (for example, *FI-AR-CR*).
- If you experience any other problems with *SAP Note Assistant*, report an incident and assign it to the component *BC-UPG-NA*.

#### Transport-based Correction Instructions

Extensive changes to one functional area typically require a huge amount of SAP Notes or multiple manual changes. *SAP Note Transport-Based Correction Instruction* (TCI) is a new way to deliver ABAP correction instructions in a flexible manner.

From a technical point of view, there are the following types of SAP Notes:

- SAP Notes without any correction instructions (such as *Consulting notes* or *Information for Installation notes*).
- SAP Notes with correction instructions.
- *SAP Note Transport-Based Correction Instructions* (TCIs).

*SAP Notes with correction instructions* contain specific, **single** corrections. They can be implemented automatically for all supported objects. However, objects that are not supported must be implemented manually, which, in turn, might require developer knowledge.

*SAP Note Transport-Based Correction Instructions* (TCIs), in contrast, contain corrections of only one specific functional area, which is an encapsulated sub component. They can contain **multiple** corrections, which reduces dependencies to other corrections and update artifacts. Technically, they contain ABAP transport requests similar to SAP Support Packages deliveries, but they are delivered together with SAP Notes as a new type of correction instructions.



Hint:

SAP Note [2489679](#) – How to identify TCI Note helps to differentiate a TCI Note from a non-TCI Note.

A TCI Note is a note that contains ABAP corrections in a package (such as *K10600ECPS4CORE.SAR*). In contrast, a non-TCI Note is a note that contains ABAP corrections to an object or new object (such as *REPS RSBDCOSO*). To identify the kind of note, you need to check the delivered correction type.

TCIs support all ABAP objects that have a transport connection. A single TCI consists of exactly one ABAP transport request and a set of installation attributes, for example, software component version, minimal SAP Support Package, languages.

The validity of a TCI is defined by the software component version and a range of SAP Support Package levels.



Note:

As prerequisite, the system landscape has to be enabled by following the information in SAP Note [2187425](#) – *Information about SAP Note Transport based Correction Instructions (TCI)*. This SAP note contains a PDF as attachment that explains the configuration process in detail.

## Benefits

Compared to SAP Notes with correction instructions, SAP Note *Transport-Based Correction Instructions (TCI)* support all transport-enabled SAP ABAP objects including Data Dictionary objects, table contents and MIME objects. In addition, they allow a fast consumption of consolidated correction instructions and no development skills are needed for implementation due to an automated installation process. If needed, TCIs can also be de-implemented or rolled back. For details, see SAP Note [2408383](#) – *TCI - Enabling System for SAP Note Transport-Based Correction Instruction (TCI) Rollback*.

## Implementation Process

The following figure outlines the implementation process for TCIs.

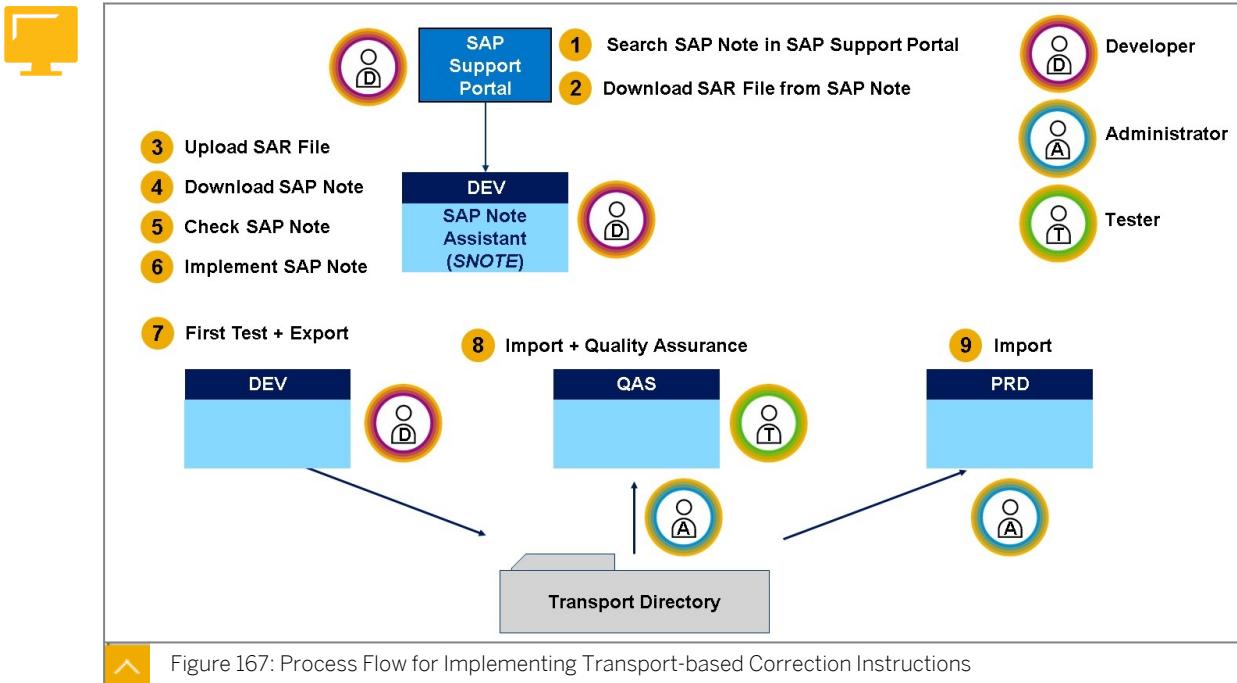


Figure 167: Process Flow for Implementing Transport-based Correction Instructions

1. In SAP ONE Support Launchpad (<https://launchpad.support.sap.com>) or from SAP Support Portal, Quick Link /notes. Open the SAP Note that contains the TCI.
2. Enter the *Correction Instructions* area and select the relevant software component.
3. Choose *Download*. The system provides a SAR file. You have to save the downloaded SAR file into a directory of your front-end computer or on operating system level of your SAP system.
4. To implement the SAP Note with the corresponding TCI in your development system, use *SAP Note Assistant* through transaction *SNOTE*. In *SAP Note Assistant*, choose *Goto* → *Upload TCI* and select the SAR file that you have downloaded.

**Note:**

The manual download and upload of the SAR file is not necessary if you use the ABAP Download Service to download SAP Notes with all of their dependencies and relevant transport-based correction instructions. See SAP Note [2554853 – SAP NetWeaver download service for SAP Notes](#).

5. Choose *Goto* → *Download SAP Note* from the menu and enter the SAP Note number that matches the SAR file. Choose *Execute* and proceed as usual (you may need additional authorizations for importing SAP Notes that contain TCIs).
6. Record your changes in a transport request. After a first test in the development system, release the transport request.

**Hint:**

Warnings about objects without directory entry can be ignored, because TCIs can contain deletions.

7. Import the transport request into the quality assurance system and perform an (integration) test.
8. After the integration test has successfully been passed, you can import the transport request into the production system.



Note:

The implementation process is also described in SAP Note [2543372 – How to implement a Transport-based Correction Instruction](#).

For more information on TCIs, see the PDF that is attached in SAP Note [2187425 – Information about SAP Note Transport based Correction Instructions \(TCI\)](#) and the online documentation for SAP S/4HANA (Product Assistance), area *Enterprise Technology → ABAP Platform → Administrating the ABAP Platform → Administration Concepts and Tools → Solution Life Cycle Management → Software Logistics → Note Assistant → SAP Note Transport-Based Correction Instructions (TCI)*.

Information on the ABAP Download Service can be found in the online documentation for SAP S/4HANA (Product Assistance), area *Enterprise Technology → ABAP Platform → Administrating the ABAP Platform → Administration Concepts and Tools → Solution Life Cycle Management → Software Logistics → ABAP Download Service*.

### SAP System Recommendations

If you want to correct an error using an SAP Note, you must know the number of the SAP Note that corrects the error.

SAP Notes that contain corrections for application transactions can be found with the help of the *Automated Note Search Tool*, transaction ANST or ANST\_SEARCH\_TOOL in the SAP system. For more information on this tool see SAP Notes [1778716 – Advanced Note Search Tool](#) and [SAP Note 1818192 – FAQ: Automated Note Search Tool](#). There is also an ANST Guide available as attachment to SAP Note [1778716](#). You can start the *Automated Note Search Tool* directly from SAP Note Assistant by choosing *Goto → (Other Tools → ) Launch ANST*.



Hint:

SAP Note [2730525 – Consuming the Note Search Webservice](#) provides a new API, valid for AS ABAP 740 and above, that calls the SOAP based SAP Note search web service in the SAP Support Backbone. The *Automated Note Search Tool* has been adapted to this change by replacing the call to the old RFC API with the call to the new API provided by this SAP Note. This change is available in SAP Note [2732094 – ANST- Implementing SOAP Based ANST Note Search](#) and applicable for AS ABAP 740 and above.

SAP Notes containing security relevant topics can be found either using the **security notes** search at <https://launchpad.support.sap.com/#/securitynotes>, which can be accessed from SAP Support Portal, quick link /securitynotes.

You can also find relevant SAP Notes by configuring and using *System Recommendations* in SAP Solution Manager (not part of this course)

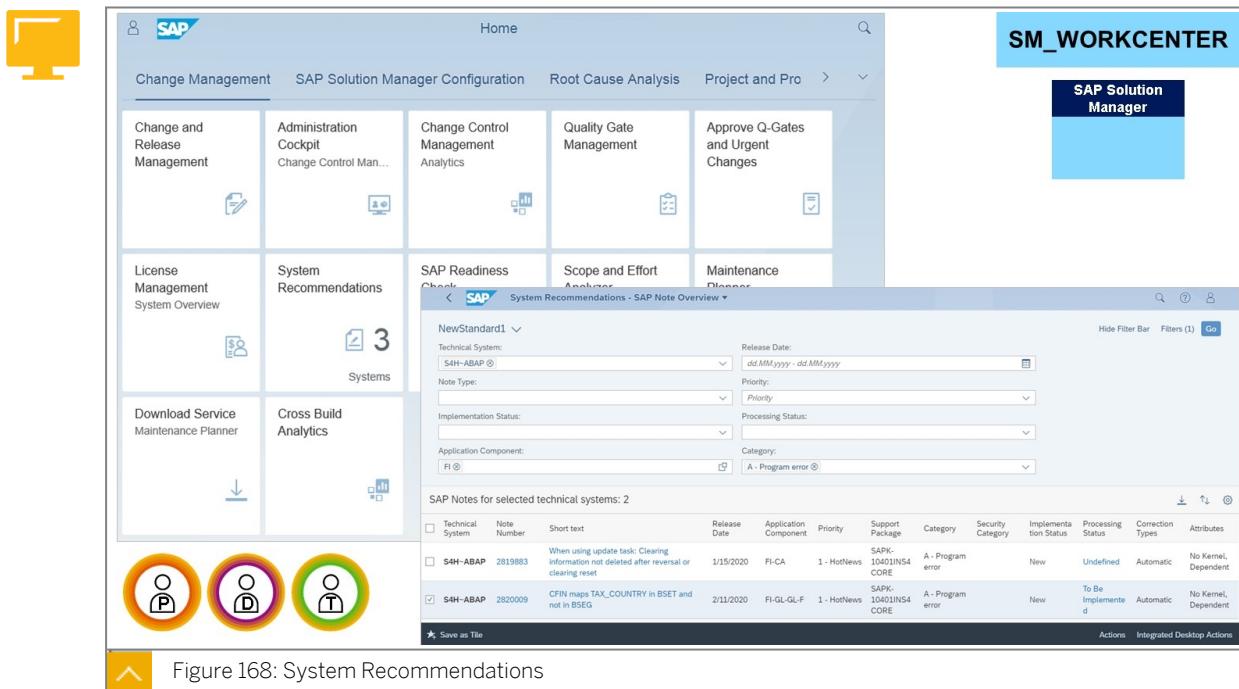


Figure 168: System Recommendations

SAP Solution Manager hosts an app called *System Recommendations*. This app allows you to manage the SAP Notes and Support Package Patches that are not yet implemented and installed on your technical systems. Once an SAP Note has reached the completely Implemented status or a Support Package Patch has been registered as being fully installed by SAP Solution Manager's *Landscape Management Database* (LMDB), you can no longer see them in the application. The app automatically calculates the SAP Notes and Support Package Patches for the desired technical system.

The following features are available:

- Calculate, display, and assign statuses to SAP Notes that are available for installation on one or more of your technical systems
- Display SAP Notes and their object list as well as the prerequisites SAP Notes
- Manage and track the workflow of an SAP Note implementation
- Analyze the effect that implementing an SAP Note has on your systems and business processes
- Create a request for change
- Select a Java patch and add it to the Download Basket
- Enable background services to automatically update SAP Note information



#### Note:

In this section, we focus only on the SAP Notes for ABAP based Systems.

The *System Recommendation* comprises the following SAP Notes categories:

Table 11: Examples of SAP Notes which are supported

SAP Note type	Explanation
---------------	-------------

SAP Security Notes	Important SAP Notes in the security category.
HotNews	SAP HotNews, which are SAP Notes with priority 1 (very high). These SAP Notes provide information to help avoid and/or solve problems that can result in data loss or crashes of the SAP system.
Performance-Relevant Notes	SAP Notes from the performance category improve the performance of your system.
Legal Change Notes	SAP Notes from the Legal Change, Announcement of Legal Change, and Correction of Legal Function categories respond to changes in legal requirements.
System Measurement (License Auditing Notes)	SAP Notes related to Licensing SAP Software, like new license checks, usage information, etc.
SAP Notes for Corrections/Patch Notes	All SAP Notes that contain corrections to program source code (ABAP) or contain patches (Java).

To start the *System Recommendations* start the transaction `SM_WORKCENTER` within the SAP Solution Manager system to start the SAP Solution Manager *Launchpad*. Then navigate to the *Change Management* area and select *System Recommendations* to start the application.



#### Note:

To find the system in the list the following activities including their required activities need to be performed: *SAP Solution Manager Mandatory Configuration* and the *Managed System Configuration*.

When you start the application, the *System Overview* is displayed. On the *System Overview* you can select the system and choose *Display SAP Notes*, the *SAP Note Overview* screen displays, as shown in the figure above.

Use the filter options to find the right SAP Note to fix your issue. By choosing *Display Detail Page*, the *Show SAP Note Details* screen displays.

With the *Integrated Desktop Actions* you could, for example *Download the SAP Note* or choose *Create Request for Change*.

For more information, see the *System Recommendation* area on SAP Support Portal (<https://support.sap.com/sysrec>) and the SAP training class SM100 – SAP Solution Manager *Configuration for Operations*.



## LESSON SUMMARY

You should now be able to:

- List the advantages of automatically implementing SAP Notes
- List options to find relevant SAP Notes and explain how to distribute them in your system landscape
- Implement SAP Notes with SAP Note Assistant
- Describe the difference between SAP Notes with correction instructions and SAP Notes with transport-based correction instructions

- Outlook: Find SAP Notes automatically

## SAP Support Packages

### LESSON OVERVIEW

In this lesson, you will learn about applying SAP Support Packages in a multi-system landscape. In addition, you will learn about modification adjustment and downtime when applying SAP Support Packages. Finally, there is a short section on feature packages.

### Business Example

As an SAP system administrator, you want to import functional enhancements into your SAP system and to correct issues in advance. You also want to apply legal changes and security improvements. For this purpose, SAP provides SAP Support Packages. Therefore you need to know how to apply Support Packages for AS ABAP based SAP systems.



### LESSON OBJECTIVES

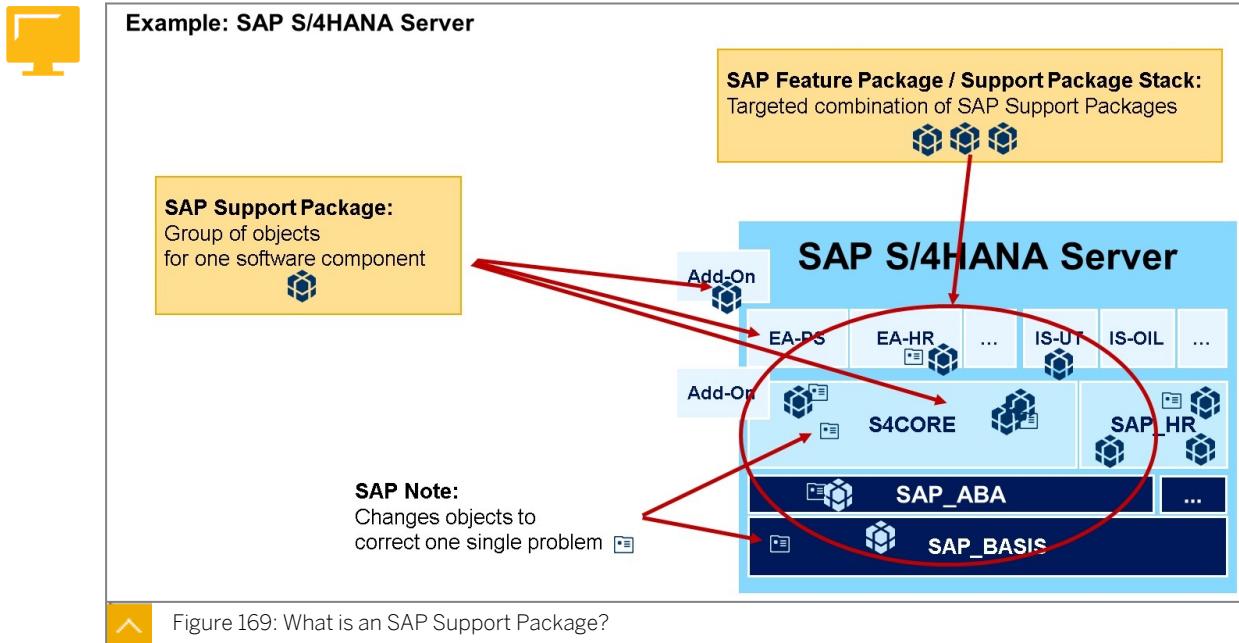
After completing this lesson, you will be able to:

- Describe the prerequisites for requesting an SAP Support Package Stack
- Import an SAP Support Package that requires a modification adjustment
- Explain the difference between SAP Support Package Stacks and Feature Package Stacks

### Introduction to SAP Support Packages and SAP Support Package Stacks

An AS ABAP based SAP system consists of different software components (see the following figure). All of these software components are regularly updated using SAP Support Packages. With the help of SAP Support Packages, issues are solved, legal changes are shipped, security issues are solved, and new functions are provided.

The SAP system should always be kept at an up-to-date level so that errors in the standard release are proactively removed.



### What is an SAP Support Package?

An **SAP Support Package** is a group of corrected or new SAP objects. SAP Support Packages are required to correct issues in various components. This is done by replacing erroneous objects with corrected versions of these objects.

Each software component (such as *SAP\_BASIS*) has a separate sequence of SAP Support Packages.

An **SAP Support Package Stack** is a set of SAP Support Packages for a product version that are best implemented together. SAP Support Package Stacks allow you to keep productive applications up to date on a regular basis with a minimal cost of ownership.



#### Hint:

In contrast to SAP Support Packages, *Feature Packages (FPs)* might include innovations, product features, functions, or functionality in addition to corrections. With regards to technology, upgrade, and administration, *Feature Packages* and SAP Support Packages are handled in the same manner, however. For details, see the end of this lesson.

## Requesting SAP Support Packages

**SAP Feature Package / Support Package Stack**

**SAP S/4HANA Server**

- EA-PS
- EA-HR
- IS-UT
- IS-OIL
- S4CORE
- SAP\_HR
- SAP\_ABA
- SAP\_BASIC

**Maintenance Planner**

Requesting SAP Support Packages and SAP Support Package Stacks requires:

- SAP Solution Manager, properly configured
- Usage of Maintenance Planner in SAP Support Portal
- Valid maintenance certificate in the SAP system

Schedule a system deployment of the calculated archives at a convenient date.

Plan a software change on your system, including download of files.

Click here to explore the system details.

Verify automatically or edit manually the software details of a system and correct errors - more.

Attribute Change Package 41 for ST-PI 740  
ST-PI 740: SP 0014  
ST-PI 740: SP 0015

SAP\_BASIS\_754\_SP\_0003  
SAP\_BASIS\_754\_SP\_0004

SAP\_BW\_754\_SP\_0003  
SAP\_BW\_754\_SP\_0004

SAP\_NW\_GATEWAY\_FOUNDATION\_754  
SAP\_NW\_GATEWAY\_FOUNDATION\_754

SAP\_GWINTO\_754\_SP\_0003  
SAP\_GWINTO\_754\_SP\_0004

SAP\_ABA\_754\_SP\_0003  
SAP\_ABA\_754\_SP\_0004

Figure 170: Requesting SAP Support Packages

Before you can apply SAP Support Packages, you must download them from the SAP Support Portal. You should apply whole SAP Support Package stacks rather than other combinations of SAP Support Packages.

For requesting SAP Support Packages, you need your SAP system registered in your SAP Solution Manager system. This SAP system information is brought to SAP Support Portal. Here, you start the *Maintenance Planner* to calculate the SAP Support Packages.

1. Register your SAP system to be patched in a central System Landscape Directory (SLD). In this SLD technical information about your SAP system is stored.
2. Transfer the information from SLD via batch job to your central SAP Solution Manager system, into the *Landscape Management Database* (LMDB).
3. Manually check or define a *Technical system* in LMDB.
4. Transfer the information from LMDB using batch job to SAP Support Portal.
5. In SAP Support Portal, start the *Maintenance Planner*. Define SAP System dependencies and perform some verification.
6. Use the *Maintenance Planner* to calculate the list of SAP Support Packages to be applied.
7. From *Maintenance Planner*, download an XML file containing the description of the calculated SAP Support Packages to be applied. Use *Maintenance Planner* to push the calculated SAP Support Packages to your download basket. Use the *Download Manager* to download the calculated SAP Support Packages from your download basket.



### Note:

You can also decide to send the system data directly to the LMDB and not through the SLD.

8. Start the tool to apply the SAP Support Packages, either `SPAM` or `SUM`. When using transaction `SPAM`, this transaction needs to be executed in client 000 of the system in question.



Note:

You can also use *Data Supplier Processing*, where the managed systems send data directly to the LMDB and not through the SLD. For details, see SAP Note [2183995](#) – *Data Supplier Processing in SAP Solution Manager 7.2 in LMDB*.

You can also send the data from the Data Supplier directly to SAP Support Portal (without involving SAP Solution Manager). For more information, see SAP Note [2855951](#) – *How to Configure SLD Data Supplier to Directly Send Data to SAP's Support Backbone*.

You also need a valid *maintenance certificate* in order to apply SAP Support Packages.

SAP recommends that you create the maintenance certificates automatically with SAP Solution Manager. This takes much less effort than creating them manually, especially if maintenance certificates for several SAP systems are needed. A maintenance certificate can also be created manually using the SAP Support Portal (using the *License Key* application).



Hint:

A maintenance certificate is valid for three months only. For more information, see the SAP Support Portal, area *Maintenance* → *Maintenance Information* → *Maintenance Certificate*.

## Importing SAP Support Packages

### Prerequisite: SPAM/SAINT Update

To import SAP Support Packages, you can use *SAP Support Package Manager* (transaction `SPAM`).

Before using the *SAP Support Package Manager*, you should patch the tool itself with a *SPAM/SAINT update*. This is a patch to enhance the functions of transactions `SPAM` and `SAINT`, for example, when dealing with new attributes from new SAP Support Packages.



Note:

Transaction `SAINT` (*Software Add-On Installation Tool*) allows you to install and upgrade add-ons.

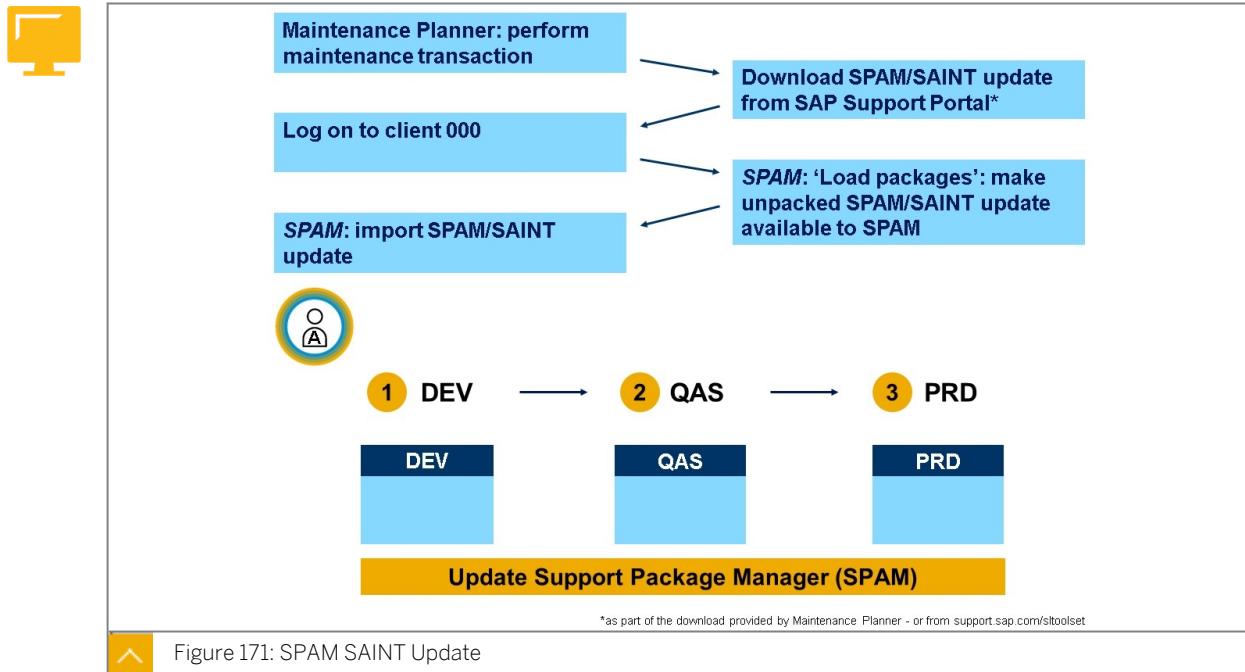


Figure 171: SPAM SAINT Update

**Note:**

SPAM/SAINT updates are in German and English only. See SAP Note [170543](#) – *Transaction SPAM only useable in language D and E.* If you work in another language, new or updated user interface elements and texts might not be properly displayed. SAP recommends that you log on in German or English if you want to use the Support Package Manager.

After you have updated transactions SPAM and SAINT to the latest patch level, from a technical perspective, you're ready to import an SAP Support Package stack.

## Process Flow

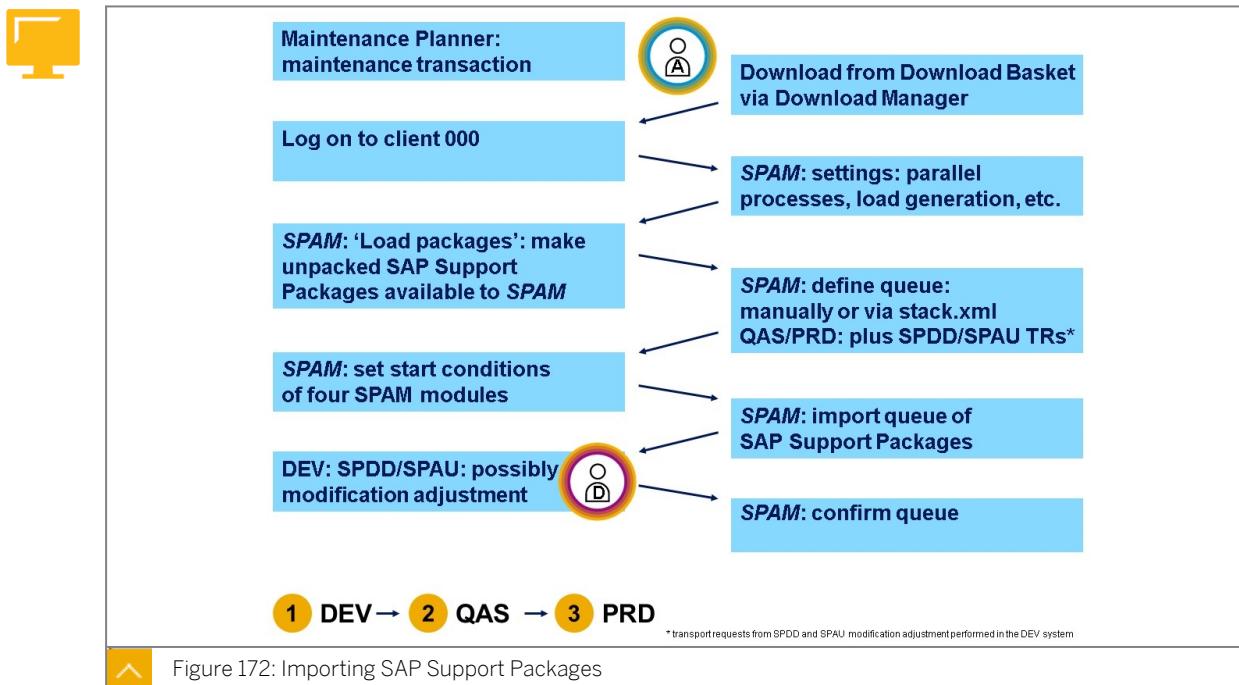


Figure 172: Importing SAP Support Packages

SAP Support Package Manager ensures that only SAP Support Packages that match your SAP system are displayed in the queue. SAP Support Packages that are intended for a different release or an add-on that is not installed don't appear in the queue.

You can compile a shared queue for all software components in the SAP system, by choosing *All Components*.

The queue contains the SAP Support Packages available for the selected component(s) in your SAP system and any required *Conflict Resolution Transports* (CRTs) and associated Add-on SAP Support Packages.

The following rules apply when you create a queue:

- SAP Support Packages for a selected software component are placed in the queue according to their sequence.
- If SAP Support Packages in the queue are related to SAP Support Packages for another software component (predecessor relationship, CRT required), the extra SAP Support Packages will be added to the queue until all predecessor relationships are included.

If the SAP system has calculated a consistent queue, you can display it and check it by choosing the *Calculated Queue* tab.

### Import Process

Transaction SPAM offers different scenarios to import the Support Packages. To set the desired scenario, choose *Extras → Settings*.

- Use the *test* scenario to check whether a modification adjustment is necessary, or whether there are any conflicts or problems (such as unreleased repairs), before you actually import the SAP Support Package. Use this scenario to estimate and possibly minimize the effort needed to import the SAP Support Packages. This scenario imports no data or

objects into your SAP system. If errors occur, you can continue the import without the need to remove the errors.



Note:

You need to select the *test* scenario explicitly. Note that the queue is empty after you run the *test* scenario and you need to define it again. You then need to select the *standard* scenario explicitly.

- The *standard* scenario imports all SAP Support Packages in the queue completely. If errors occur, you can only continue and complete the import after you have removed the errors. If you selected the *standard* scenario, you could choose between the *standard* import mode and the *downtime-minimized* import mode to reduce downtime.

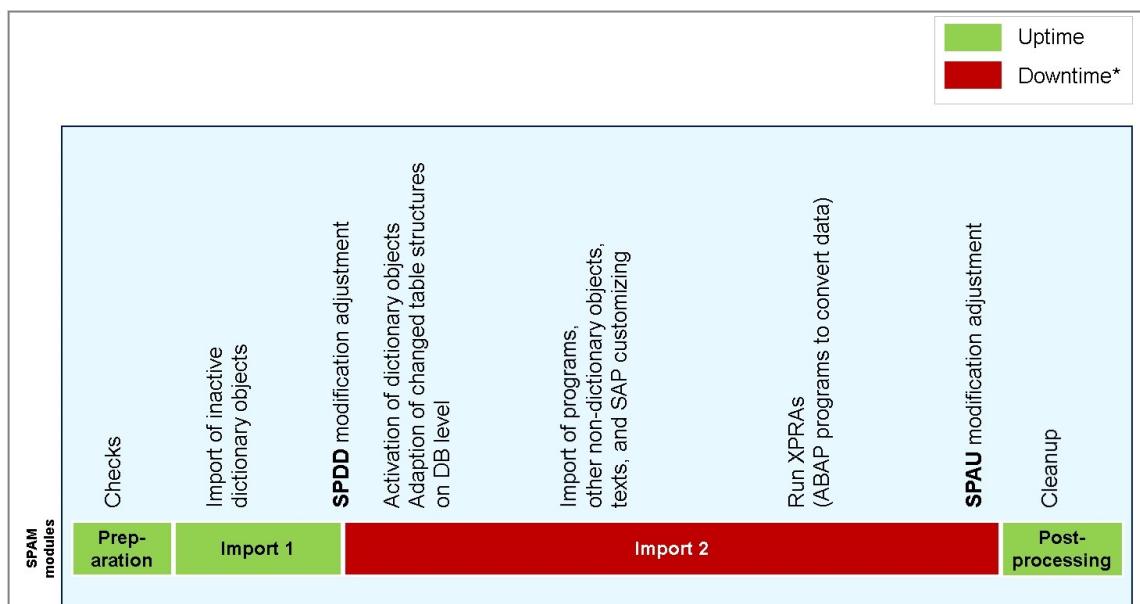


Figure 173: Support Package Manager: Import Process



Note:

During the time needed for importing SAP Support Packages, no imports can take place, and no manual changes can be made to repository objects (ABAP programs, dictionary objects). The transport requests for modification adjustment are handled in a special way, so the time for importing SAP Support Packages should be kept as short as possible.



Hint:

When importing SAP Support Packages, the *SAP Support Package Manager* will check if there are still any transport requests in the import queue for the old SAP Support Package level. In this case, the *SAP Support Package Manager* returns a warning. You have the option to ignore the warning and still import the SAP Support Packages.

Due to the size and scope of current SAP Support Packages, importing SAP Support Packages requires longer SAP system downtime. The SAP system is not stopped and restarted during the import process, but it cannot be used productively during this time. This restriction is a disadvantage for many production systems.

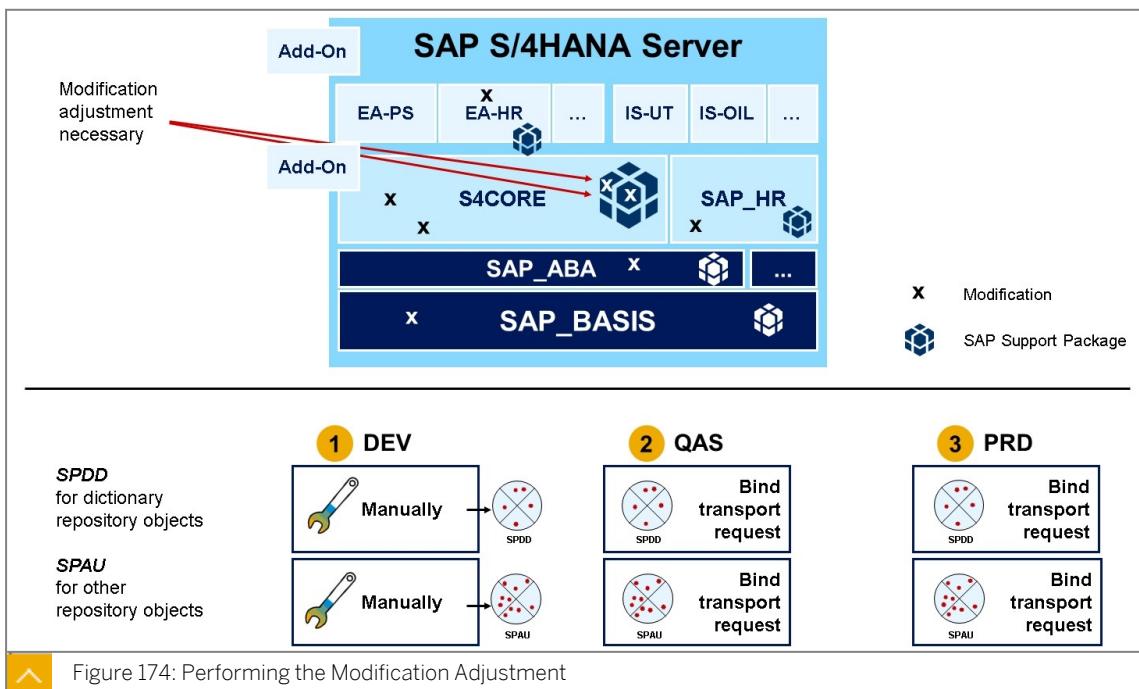
To reduce the downtime during the import of SAP Support Packages, an import procedure has been developed that allows you to import a large portion of the objects to be imported during production operation. These objects are mainly programs. The downtime can be significantly reduced if an SAP Support Package contains a high portion of programs (the portion might be up to 80%). With the *downtime-minimized* import mode (not shown in the figure above), the objects are imported into the database in an inactive state and are, therefore, “invisible” to the SAP system and you can continue to use the SAP system productively for a longer time.



#### Hint:

It is also possible to import SAP Support Packages with the *Software Update Manager* (SUM). Importing SAP Support Packages with SUM is more difficult for the SAP system administrator, but can lead to a shorter downtime if configured correctly.

## Modification Adjustment



The modification adjustment works as follows:

1. The SAP Support Package Manager stops in order to allow the modification adjustment.
2. Create one transport request in the *Transport Organizer* and tasks inside the transport request for the developers so that your developers can perform the modification adjustment.

3. Ask the developers to perform the modification adjustment for their objects. After the adjustment is complete, the developers must release their tasks and inform you. **The adjustment should be performed in the development client, not client 000**, of course.
4. If you left the transaction, call the Support Package Manager (transaction SPAM) again, and choose *Support Package → Import Queue* from the menu. Then the SAP system prompts you again to perform the modification adjustment. As it has already been completed, ignore the message and choose *Continue*. The Support Package Manager then completes the processing and displays the status.

The *import logs of the queue* function displays logs for steps of the Support Package Manager that use the transport control program *tp*. After the queue has been successfully imported, you should always check these logs. To jump to the log display, choose the menu path *Goto → Import Logs → Queue* in the initial screen of the Support Package Manager.

Finally, you need to confirm the successful import of the queue into your SAP system. This ensures that you can import other SAP Support Packages in future.

If you have not yet confirmed successfully imported SAP Support Packages, you will be prompted to confirm these packages during the next update of the SAP system.



**Note:**

Without this confirmation, you can't import additional SAP Support Packages or complete even other maintenance-related tasks, such as the implementation of SAP Notes in transaction SNOTE.

If you have a multi-system landscape, you only need to perform the modification adjustment once. The system stores the data for the modification adjustment in transport requests that you can include in the import process when importing the Support Package queue into the follow-on systems.



**Hint:**

When a modification adjustment transport is imported as part of a Support Package queue, it is deleted from the normal transport flow for workbench transport requests. Modification adjustment requests are not forwarded to follow-on systems automatically.

If you are working with the classic three-system landscape comprising a development system (*DEV*), quality assurance system (*QAS*) and production system (*PRD*), the modification adjustment transport is put into the *QAS* import queue after being exported from the *DEV* system. Including the adjustment transport in a Support Package queue in system *QAS* deletes it from the *QAS* import queue. Because no transport forwarding takes place when importing a Support Package queue, the adjustment transport is not forwarded into the import queue of the *PRD* system. This means that the modification adjustment will not be imported to your *PRD* system with the Transport Management System.

As a consequence, you need to import the adjustment transport into the *PRD* system as part of a Support Package queue using the same procedure as in the *QAS* system.

## SAP Support Packages and TMS



### Note:

In the remaining part of this lesson, for the sake of simplicity, the term *SAP Support Package* is used for both *SAP Support Packages* and *SAP Support Package stacks*.

A transport request that has been released from an SAP system with a certain release should only be imported into an SAP system with the same release. This is obvious when transporting objects from the SAP standard, for example modifications and customizing. But this restriction should also be considered, when transporting customer developments, because these can be related to SAP standard objects.

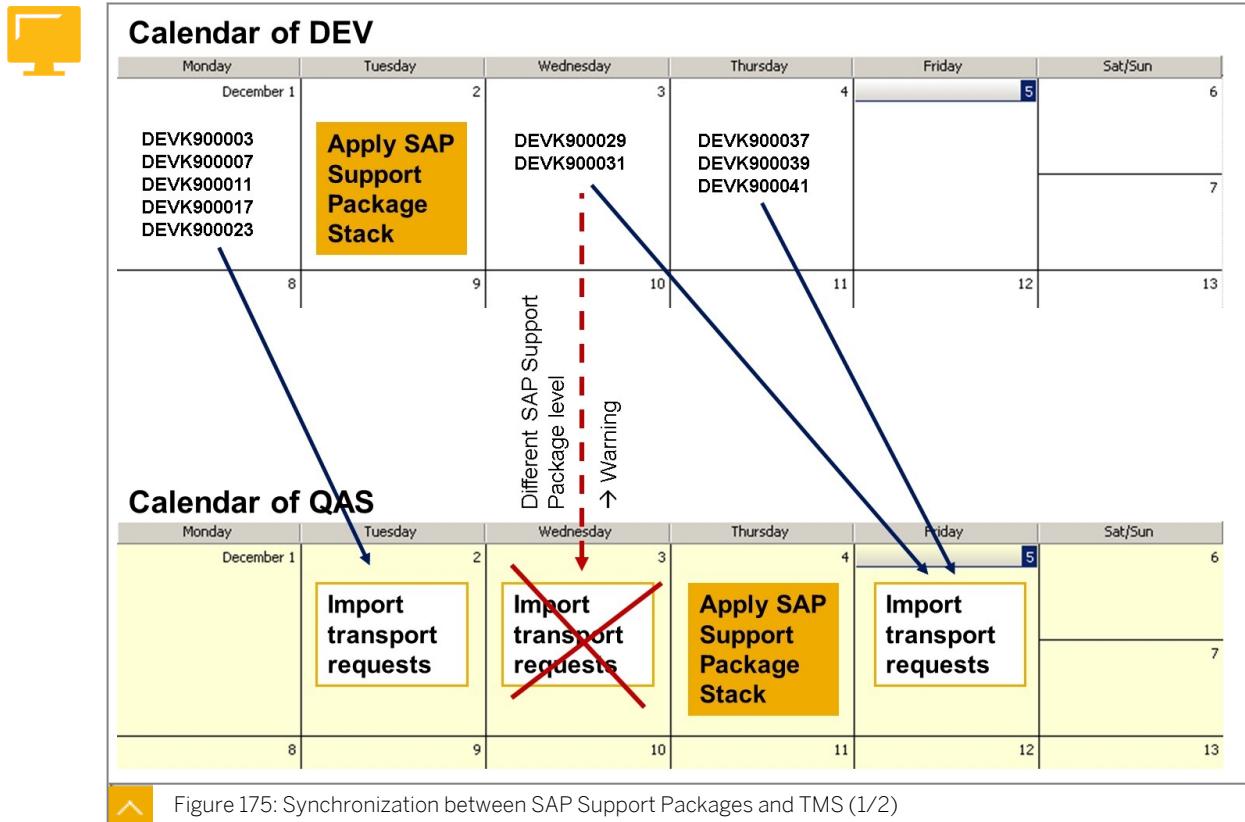
You must consider not just the release, for example SAP S/4HANA Server 2020 versus SAP S/4HANA Server 2021, but also the level of the corresponding software components and the SAP Support Package level. For example, a transport request that has been exported from an SAP S/4HANA 2021 Server with S4CORE Support Package 2 should not be imported into an SAP S/4HANA 2021 Server with S4CORE Support Package 1.



### Hint:

The export release should be equal to the import release. Transport requests usually rely on a certain release of the corresponding software components and their SAP Support Package level. They should not be imported to an SAP system with a different level than the one from which they were exported.

As a result of this, the applying of SAP Support Packages and the importing of transport requests must be coordinated (see the following figure).



The *Transport Management System* (TMS) checks if a transport request fits the software component version and SAP Support Package level of the target SAP system. The result of this check can be seen in the import queue, in the CV column (to see this column, choose *Display More* in the application toolbar). This is done by comparing the software component vector of the transport request with the software components of the target system. If the check is negative, the *Status* column shows the icon for *Request does not match component version*. For details, see the following figure.



Hint:

A transport request that does not fit the target system can still be imported choosing the import option *Ignore Invalid Component Version*.

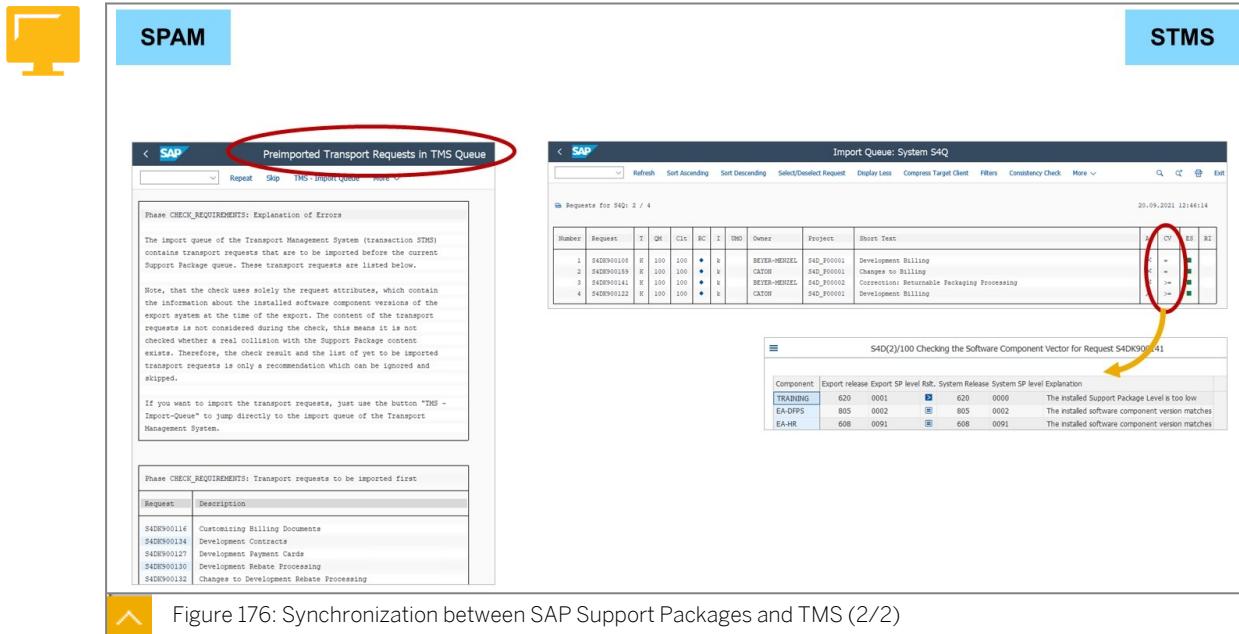


Figure 176: Synchronization between SAP Support Packages and TMS (2/2)

**Hint:**

The version check will be performed only if the *tp* parameter *SP\_TRANS\_SYNC* is set to *ON* for the SAP system to be imported in. For more information, see SAP Notes [1742547](#) – *Information about component version check in TMS* and [1688610](#) – *TMS import queue warning message : 'Does not match component version' or 'Checking components of the requests'*.

A corresponding check is also performed when importing SAP Support Packages. If there are still any transport requests in the import queue for the old SAP Support Package level, SAP Support Package Manager will return a warning (see the figure above). However, you have the option to ignore the warning and still import the SAP Support Packages.

### Importing SAP Support Packages with SUM

It is also possible to import SAP Support Packages with *Software Update Manager (SUM)*. *SUM* is the tool used for an SAP system upgrade, but it can also be used to import SAP Support Packages. *SUM* is more complicated to use than the *SAP Support Package Manager*, transaction *SPAM*, but it can generate a much shorter downtime than *SPAM* when configured appropriate.

**Hint:**

If downtime is an issue, consider using *SUM* instead of *SPAM*.

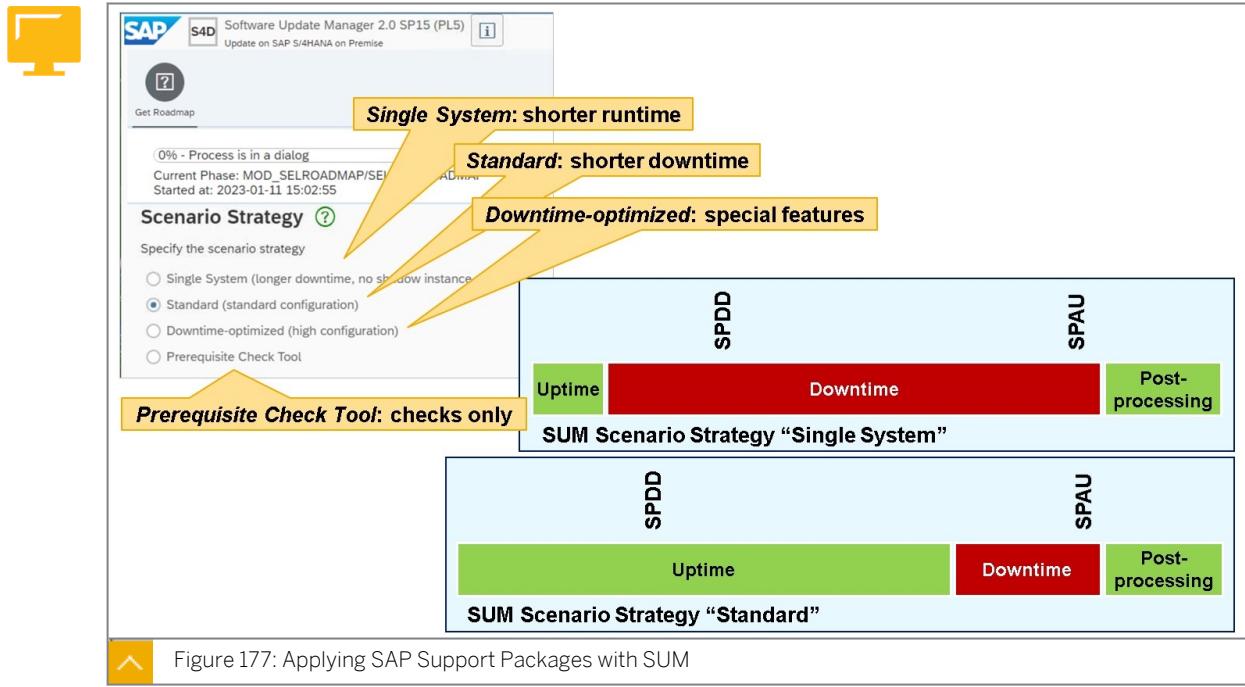


Figure 177: Applying SAP Support Packages with SUM

The **SUM Standard** mode uses a shadow system in which several activities take place during the preparation phase thus reducing the downtime.

Both, **SPAM** and **SUM** can be used now and in the future. Both tools have scenarios in which they are the preferred choice.

**Note:**

Both tools, **SPAM** and **SUM**, can be used to import SAP Support Packages. In many cases you would choose the **SPAM**, but you should choose **SUM** for importing a large number of SAP Support Packages in system landscapes with downtime critical SAP systems. For more information, see SAP Note [1803986 – Rules to use SUM or SPAM/SAINT to apply SPs for ABAP stacks](#).

The following figure provides advantages of both methods:

SPAM/SAINT	SUM
Easy handling	Short downtime possible, depending on scenario strategy
Short runtime	Shows point of no return with reset option
Test mode, showing modification preview	Performs several checks, e.g. database free space
Does not require Maintenance Planner	Stack.xml from Maintenance Planner required for most scenarios
Runs as transaction, inside SAP system, possible as dialog or batch	Runs separately, outside SAP system
Can work with third party add-ons unknown to Maintenance Planner	Can restart the SAP system, can update the kernel

Figure 178: SPAM versus SUM

## Feature Package Stacks and SAP Support Package Stacks

SAP continuously improves SAP software through the SAP Notes service and Feature Packages and Support Packages.

For SAP systems based on AS ABAP, an **SAP Support Package** is a bundle of software corrections available as a Support Package for the ABAP programming language.

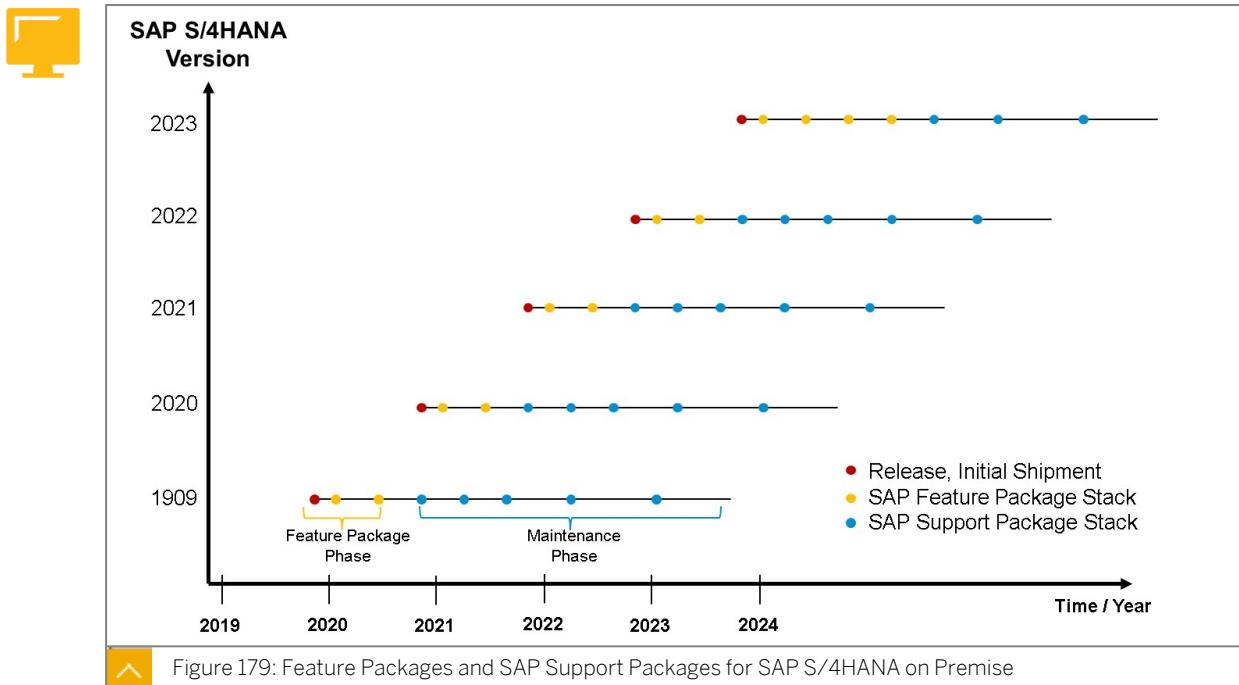
**Feature Packages** are packages that deliver non-disruptive innovation for generally available product versions, also bundling software corrections and legal changes. A Feature Package delivery comes as a *Feature Package stack*. Like an *SAP Support Package stack*, it must be used in the given combination. *Feature Package stacks* should be seen as an entity in themselves. Customers must obey the minimum requirements and dependencies between individual components and apply the *Feature Packages* and patches specified in the *Feature Package stack* together.



### Note:

*Feature Packages* are not provided for all product versions. SAP decides case by case whether a *Feature Package* phase is offered after general availability of a product version for a defined period.

For on-premise software products, SAP S/4HANA provides innovations on a regular basis through releases and *Feature Packages*.



The last *Feature Package* stack for a given release before a new release is shipped is usually the starting point of the maintenance-only phase. This *Feature Package* stack is also called *Go-to Feature Package*. During a maintenance-only phase, SAP provides corrections but no new innovations for the release.



## LESSON SUMMARY

You should now be able to:

- Describe the prerequisites for requesting an SAP Support Package Stack
- Import an SAP Support Package that requires a modification adjustment
- Explain the difference between SAP Support Package Stacks and Feature Package Stacks



## SAP System Upgrade and SAP S/4HANA Conversion

### LESSON OVERVIEW

In this lesson, you will get a brief overview about planning and performing an SAP system upgrade and a conversion to SAP S/4HANA.

### Business Example

As an SAP administrator of your company, you need a basic understanding of how an SAP system upgrade or an SAP S/4HANA conversion works.



### LESSON OBJECTIVES

After completing this lesson, you will be able to:

- List capabilities of SUM
- Describe important steps for planning and performing an SAP System Upgrade or an SAP S/4HANA Conversion
- Outline the idea of business functions

### Capabilities of SUM



#### Note:

When performing a release change of an SAP system, sometimes the term *installing parts of an SAP Enhancement Package* is used (for example, when going from SAP ECC 6.05 to SAP ECC 6.07) and sometimes the term *upgrade* is used (for example, when going from SAP ECC 6.05 to SAP ECC 6.08). This depends on the start and the target release. There are two different terms, because the technique used differs at one point. A release change of an SAP S/4HANA Server system is always called an *upgrade*.

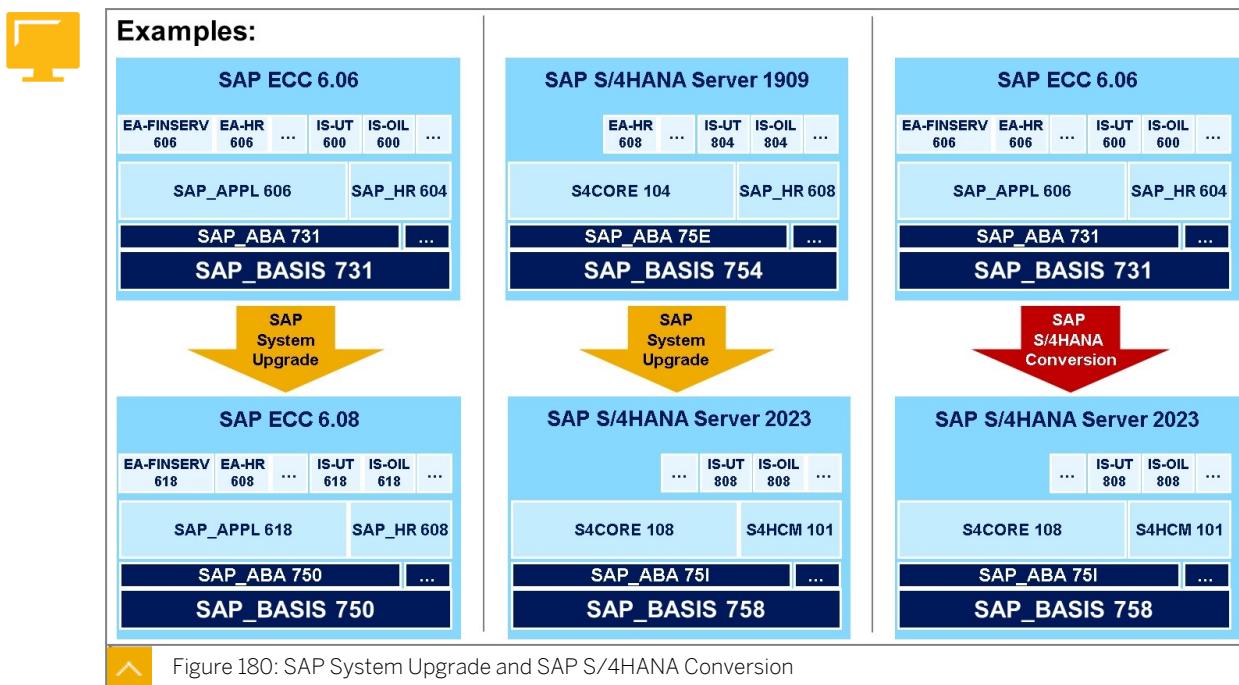
Changing the product from SAP ECC to SAP S/4HANA Server is called *SAP S/4HANA Conversion*. This procedure is technically related to an *upgrade* procedure, but needs many additional manual steps to be performed.

The central tool used is the *Software Update Manager (SUM)*. The SUM can be used - besides others - for one of the following:



- Import customer transport requests
- Import SAP Support Packages
- Install and upgrade add-ons

- Perform an SAP system upgrade
- Migrate the database via DMO (*Database Migration Option*) to:
  - SAP HANA DB
  - SAP ASE
  - MS SQL Server
  - IBM DB2
  - SAP MaxDB
- Perform an SAP S/4HANA Conversion:
  - Without DMO
  - Including DMO



#### Note:

An SAP system upgrade and an SAP S/4HANA Conversion exchange software components of the old release with software components of the new release. In the case of an SAP system upgrade, the type of the SAP system remains as it is (for example, SAP ECC or SAP S/4HANA Server). In an SAP S/4HANA conversion, the type of the SAP system is exchanged from SAP ECC to SAP S/4HANA Server.



#### Hint:

For detailed information, see Quick Link /s/toolset on SAP Support Portal (<https://support.sap.com/s/toolset>), area System Maintenance.

## Planning and Performing an SAP System Upgrade

The first thing that happens is that the need for an upgrade is recognized. This recognition of the need to upgrade could be at a number of different levels of upgrade recognition triggers; some are very operational in nature, while others are much more strategic. On the operational side, you may realize that you have a serious technical SAP system limitation within your current environment.

At some time, an SAP release reaches its end of maintenance. This means that SAP no longer delivers SAP Support Packages that adapt your processes for legal requirements and correct errors. See Quick Link */pam* (*Product Availability Matrix*) in the SAP Support Portal.

The new release can contain many new functions that your company requires. New features can be added on the **basis** side (new administration tools) or on the **application** side (new and enhanced business functionality).

The following factors are also important when deciding whether to upgrade:

- *Costs*: what will the upgrade cost me in total?
- *Payback/ROI*: will the upgrade have financial advantages for me?
- *Benefits*: what advantages will the upgrade have?
- *Risks*: are there risks involved in the upgrade?

Change and improvements are integral parts of today's business environment. SAP supports and embraces the process of continuous change as:

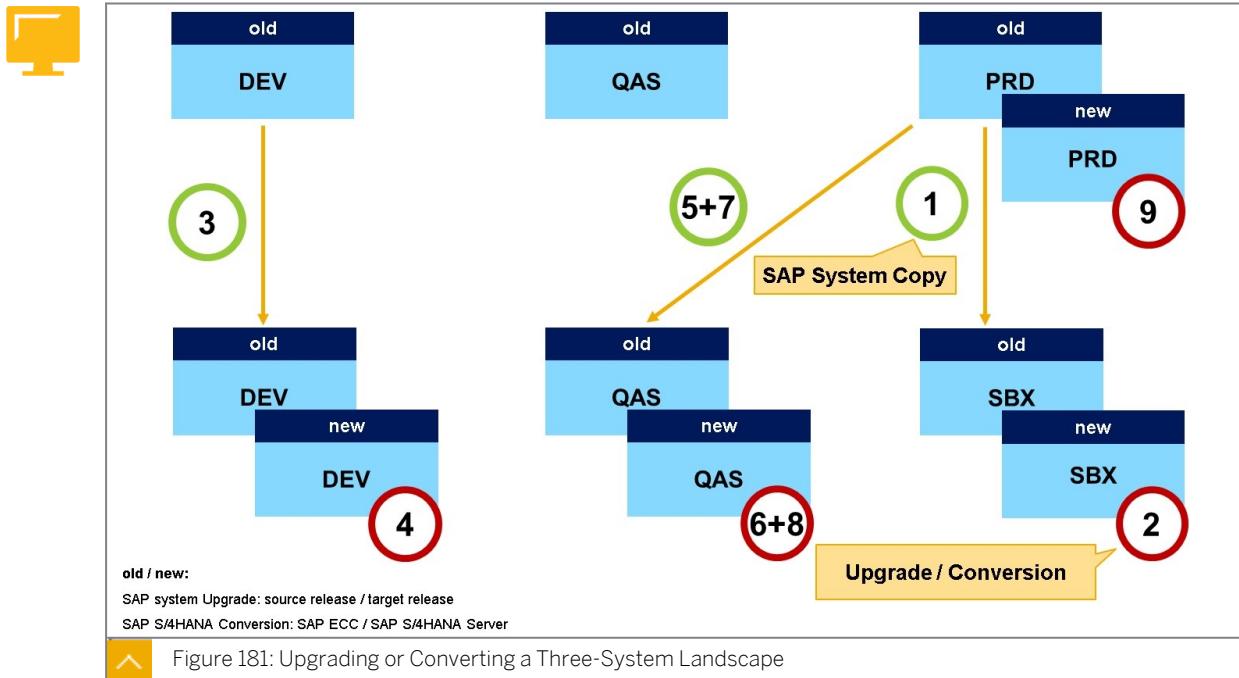
- New SAP functions becomes available.
- Market technology changes.
- New business requirements are defined.



### Note:

Focus on the typical problem areas, such as solution gaps. Assuming that you have documented your initial implementation, one of the easiest things to evaluate during an upgrade planning is whether or not any of the original gaps can be replaced by standard functions in the target release.

Review each component of new release functions. Determine how these enhancements can add value to your company's operations.



When planning an SAP system upgrade or an SAP S/4HANA Conversion, consider the following topics:

- Release customizing
- Modification adjustment and migration of modifications into SAP standard
- Adaptation and testing of customer development and enhancements
- Adaptation and testing of interfaces
- End-user training
- Validation and testing of the new release

The biggest part of the upgrade project represents the adaptation of the current business processes to the new features of the new SAP system. There are not only changes to the Customizing and the repository. There are also changes for the end users how to work with the new applications. When the development system has been upgraded to the new release, it can still be necessary to develop bug fixes for the productive system which is still working with the start release.

Some additional technical upgrade process related issues are:

- Hardware requirement on new release: server, front-end, network
- Sizing forecast and SAP system configuration for new release
- Planning of OS, DB, and SAP system upgrade
- Testing and validating a backup strategy for the upgrade and on the new release
- Performing technical update/upgrade on the whole SAP system landscape (transport landscape)
- Post-upgrade activities including performance monitoring

Since the requirements of the new release are changing, this requires changes to the configuration which in turn may need a sizing forecast to run. Finally, the whole environment is concerned, for example, the server hardware, the client hardware, the network, the operating system, and the database. Especially when upgrading the operating system, other software on this computer has to be tested. For example, the backup software has to be tested.

Because of the new requirements, performance monitoring is more important after the upgrade.

When deciding which SAP release you want to upgrade to, consider:

- The SAP release strategy
- IT costs for upgrade and maintenance
- Costs for adapting business processes
- Business requirements versus SAP functions
- Costs versus Payback/ROI



**Note:**

When considering an upgrade of an AS ABAP based SAP systems or an SAP S/4HANA Conversion, many aspects stated here depend on the activation of business functions. There is a separate section on this later in this lesson.

An SAP system upgrade is more than just a technical upgrade. The upgrade of an SAP system landscape should be executed as a project. A significant amount of planning and lots of steps are required in order to perform a release change for the SAP system landscape. Customer upgrade projects last several months. Update projects last about half as long. Don't forget to consider:

- Budgeting and resource planning
- Upgrade project in relation to other implementation/roll out activities
- Availability of resources
- End of maintenance of the current release

A technical upgrade of one SAP system involves the following steps (see the following figure):

1. When planning the upgrade, you should make decisions about the upgrade strategy. An exact upgrade schedule should be drawn up.

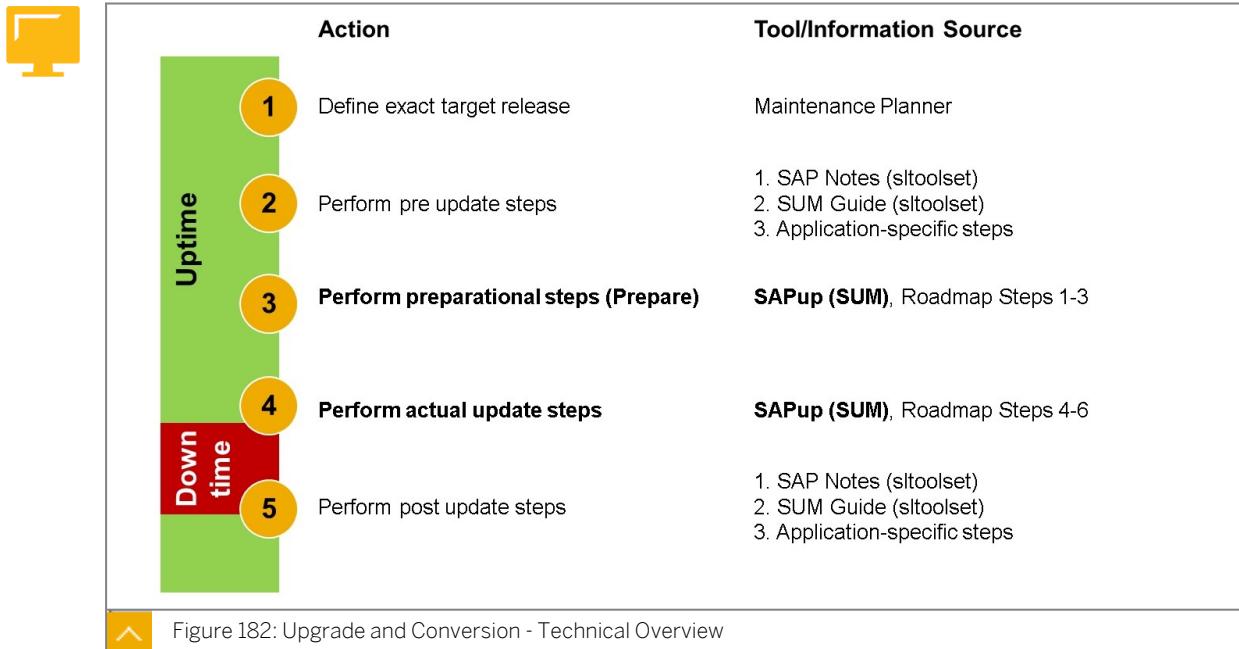


**Caution:**

It is important to read the SAP Notes about the upgrade. If you don't, your SAP system upgrade likely will fail.

2. Certain requirements (software, hardware, operating system, database) must be met before you can upgrade the SAP system. Careful preparation of the upgrade is the best guarantee that it will run without errors.
3. After you have performed the preparations, you can start.

4. The *prepare* part of the upgrade automatically checks if your SAP system is properly configured for performing the upgrade. However, you must perform many tests and actions manually before starting the upgrade.
5. Perform the upgrade.
6. Follow-up activities include all actions necessary for completing the upgrade. SAP recommends completing the actions in the order given in the upgrade guides.



Note:  
*Software Update Manager (SUM)* is the “complete package”; *SAPup* is the main tool inside.

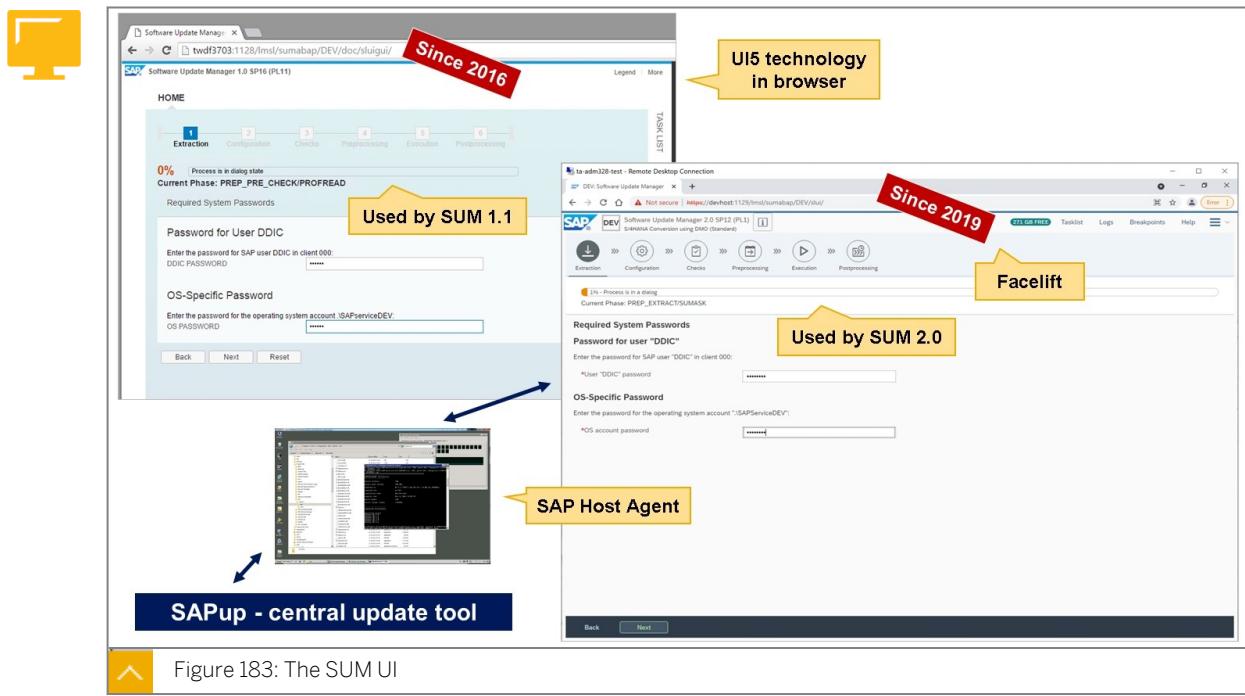


Figure 183: The SUM UI

Several tools help to perform the upgrade or conversion. They are part of the technical upgrade process.

- The *prepare* part of SUM has to run before the SAP system upgrade. You have to repeat it until it is error-free. The error messages are written to the log file *CHECKS.LOG*.

Checks are performed on the source release. For example, it checks whether the source release of the SAP system, the database, and the operating system is sufficient for this upgrade, whether there is enough space in the database available, and whether modified SAP objects are still in unreleased transport requests.

SAP Support Packages and add-ons are collected for binding them to the upgrade or conversion process. This is very important. If, for example, you don't bind enough SAP Support Packages to the upgrade or conversion, this will result in a loss of data during the procedure. If you don't maintain your add-ons, the whole SAP system can become unstable and inconsistent.



#### Note:

This is done with the help of the *Maintenance Planner* and by adding extra Support Packages.

Tools are imported in the source SAP system that are needed for the procedure.

- With the SUM UI, the procedure runs independently from a dedicated front end server, so you can control and monitor the progress of the procedure from a number of different places (see the figure above).

This provides optimal support for a remote procedure. The SUM provides an alert mechanism that lets you start an external program (for example: sending an SMS to your mobile) if it breaks down.

- With the new release, ABAP programs of the new software components are exchanged. They are delivered in their source code, but not with their load. This is not possible,

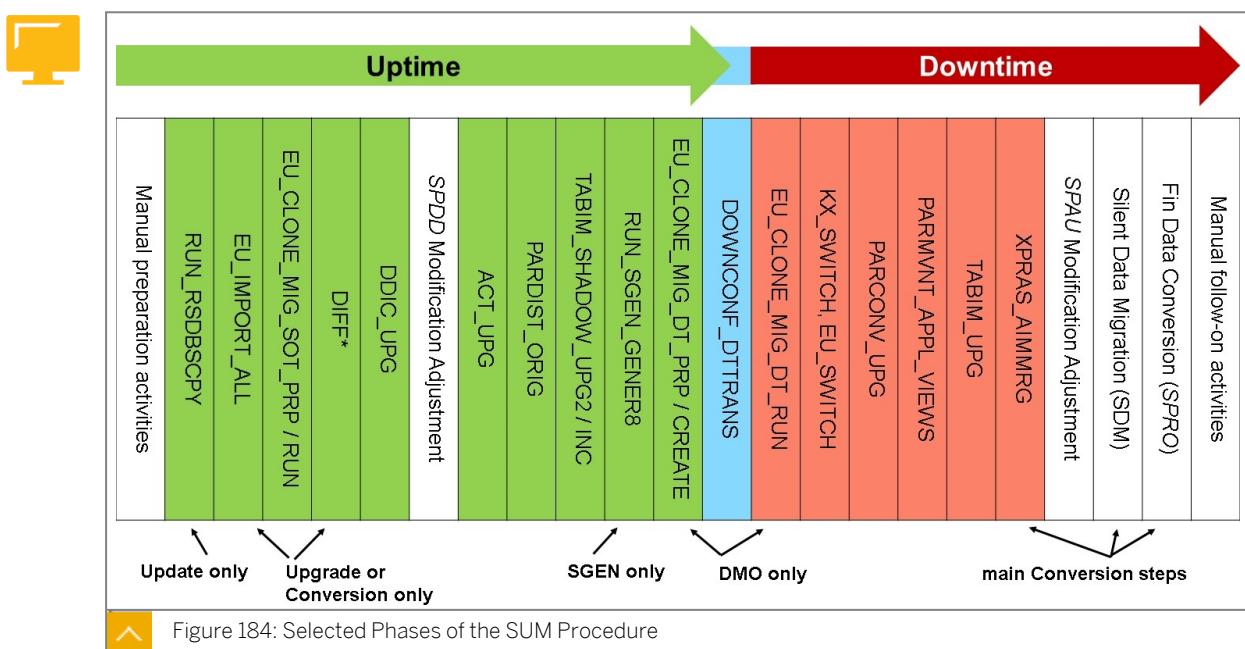
because the load depends on the local environment (kernel, operating system, and hardware). When you call a program the first time, its load will automatically be generated if it does not already exist. This may, however, reduce production system performance for some time after the update/upgrade. To avoid this, you can use transaction SGEN to generate the loads at the end of the technical downtime, during the downtime – or let SUM do the generation before the begin of the downtime, during productive operation.

The latest SUM archive can be downloaded from SAP Support Portal, Quick Link /sltoolset (<https://support.sap.com/sltoolset>) in the System Maintenance area.

Perform the following steps to start SUM (actually, start SAPup):

1. Copy the SUM archive to the host, the PAS of the SAP system runs on, for example, to directory `usr/sap/<SID>`, or any other directory. This directory should have about 50 to 200 GB of free space.
  2. Unpack the SUM archive. This results in around 15.000 files and several directories.
  3. Update the SAP Host Agent, if necessary. It is used to connect the SUM UI to SAPup. SAP Host Agent should be up to date.
  4. Configure the SAP Host Agent from within the SUM/abap directory: execute the command **SUMSTART confighostagent <SID>**. On a Windows operating system, you have to perform this step as user `<sid>adm`, on a Linux/Unix operating system as user `root`.
  5. Connect a UI5 enabled Web browser via `https://<hostname>:1129/lmsl/sumabap/<SID>/slui`. This connects the SUM UI (the Web browser) to the SUM tool SAPup.
  6. Log on to SUM with user `<sid>adm`.

When using SUM, just as SPAM or SAINT, these steps are performed first on the development system, then on the quality assurance system, then on the productive system.



The figure above shows the main phases of the SUM procedure.

- Manual preparation activities have to be performed.

- *RUN\_RSDBSCPY* clones tables from the original to the shadow repository (update only).
- *EU\_IMPORT\_ALL* creates the shadow repository from so called upgrade DVDs (upgrade and conversion only).
- *EU\_CLONE\_MIG\_SOT\_\** creates the shadow data for the shadow system (DMO only).
- *DIFF...* copies customer-specific objects from the original to the shadow repository (upgrade and conversion only).
- *DDIC\_UPG* imports dictionary objects from the download directory to the shadow repository.
- SPDD modification adjustment takes place (in development system, only)
- *ACT\_UPG* activates all ABAP dictionary objects that are not delivered activated.
- *PARDIST\_ORIG* starts the distribution.
- *TABIM\_SHADOW...* imports non-dictionary objects from the download directory to the shadow repository, also copies upgrade and language data from the download directory, only if it is inserted into new tables.
- *RUN\_SGEN\_GENER8* runs SGEN, if selected.
- *EU\_CLONE\_MIG\_\** preparations for downtime migration (DMO only).
- *DOWNCONF\_DTTRANS* begin of downtime, ramp down is performed.
- *EU\_CLONE\_MIG\_DT\_RUN* performs the downtime migration (DMO only).
- *KX\_SWITCH* switches to new kernel.
- *EU\_SWITCH* switches to new repository.
- *PARCONV\_UPG* converts application tables.
- *PARMVNT\_APPL\_VIEWS* moves the nametab of application tables.
- *TABIM\_UPG* imports upgrade and language data from the download directory.
- *XPRAS\_AIMMRG* executes XPRAs and *After Import Methods* (AIMs) (both are ABAP programs executed to adopt data to the new release).
- SPAU modification adjustment takes place (in development system, only)
- In case of an upgrade of an SAP S/4HANA server system or an SAP S/4HANA conversion: silent data migration was initialized by SUM and is started now.
- In case of an SAP S/4HANA conversion: the FIN data conversion has to be performed.
- Manual follow-on activities have to be performed.

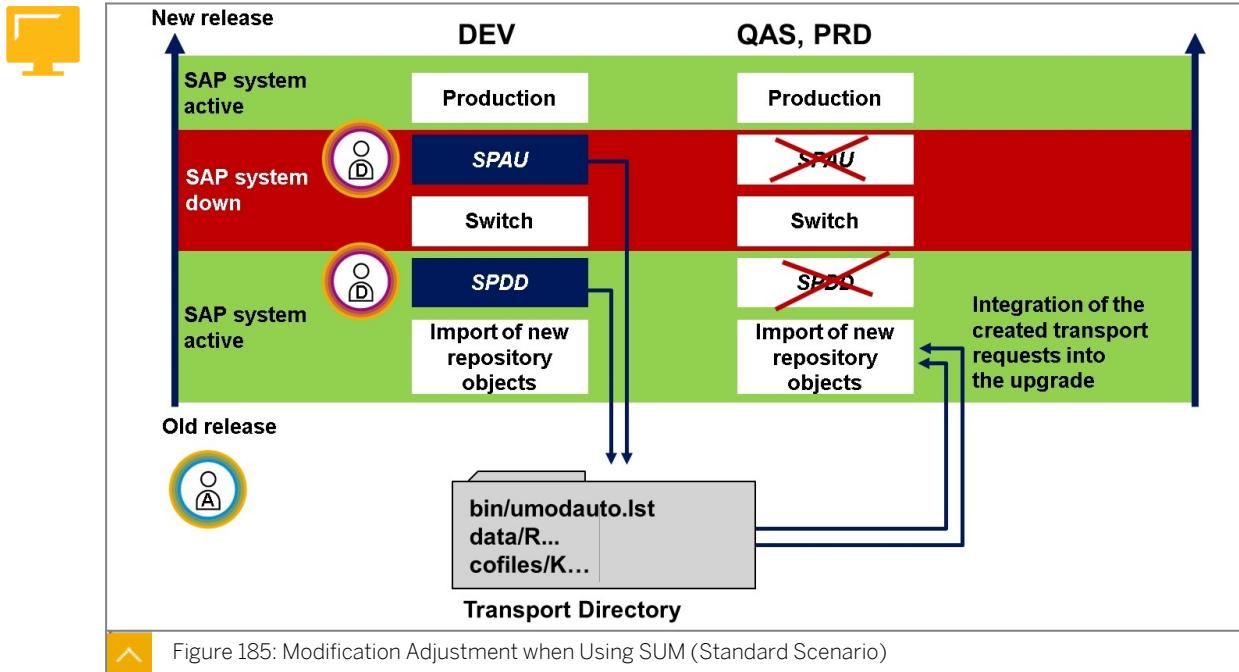


Figure 185: Modification Adjustment when Using SUM (Standard Scenario)

**Note:**

The terms *SAP system active* and *SAP system down* in this figure refers to the SUM configuration *Standard*. When choosing SUM configuration *Single System*, almost the entire procedure runs in downtime.

The start of the update process involves the transfer of new data to the SAP system. SAP repository objects are imported into the SAP system. All repository objects that have been modified by customers must be compared to the new SAP standard during the update process.

In order to avoid loss of data and table fields that customers may have created, conflicting table structures must be merged before the activation of dictionary objects in the update process.

If objects need to be adjusted, use the transactions SPAU and SPDD. All modifications made by customers are then merged with the new SAP object versions to retain data; otherwise, the new SAP version will be activated and data may be lost.

When choosing the *Standard* mode, the SAP system is available during the activation phase. The activation takes place via the shadow instance via the shadow system in the shadow repository.

When the update is completed, the SAP system is successfully running at the new release level. Customer-developed objects and modifications have been preserved.

**Hint:**

The two transport requests from SPDD and SPAU (one from SPDD and one from SPAU) can be included (**not imported**) into the update of the subsequent SAP systems in the same SAP system landscape. By doing this, the modification adjustment is performed automatically in this subsequent SAP systems. Here you only have to adjust those objects manually that were modified in the subsequent SAP systems, but not in the SAP system in which you have performed SPDD and SPAU.

The ABAP Dictionary objects (tables, data elements, domains, and so on) are adjusted during productive operation, before the activation of the ABAP Dictionary. The adjusted objects are collected in a transport request. You don't release this transport request; instead it must be flagged for export in transaction SPDD. Towards the end of the upgrade, SUM exports this transport request into the transport directory, and then registers it for transport in the file <transport directory>/bin/umodauto.lst.

Non-dictionary repository objects (reports, screens, and so on) are adjusted towards the end of the upgrade during downtime. At this stage, the import of SAP objects has already been completed. However, the old, modified version is still available in the version database. As with ABAP Dictionary objects, all adjustments are released to a transport request that is flagged and then exported and registered by SUM.

**Caution:**

Do not attempt to import adjustment transport requests into the SAP system manually during SPDD. This can lead to a loss of data.

Do not activate any objects during SPDD. Activation is carried out automatically after the modification adjustment.

For further information, see the upgrade manual and online application help for SPDD/SPAU.

## Activating Business Functions

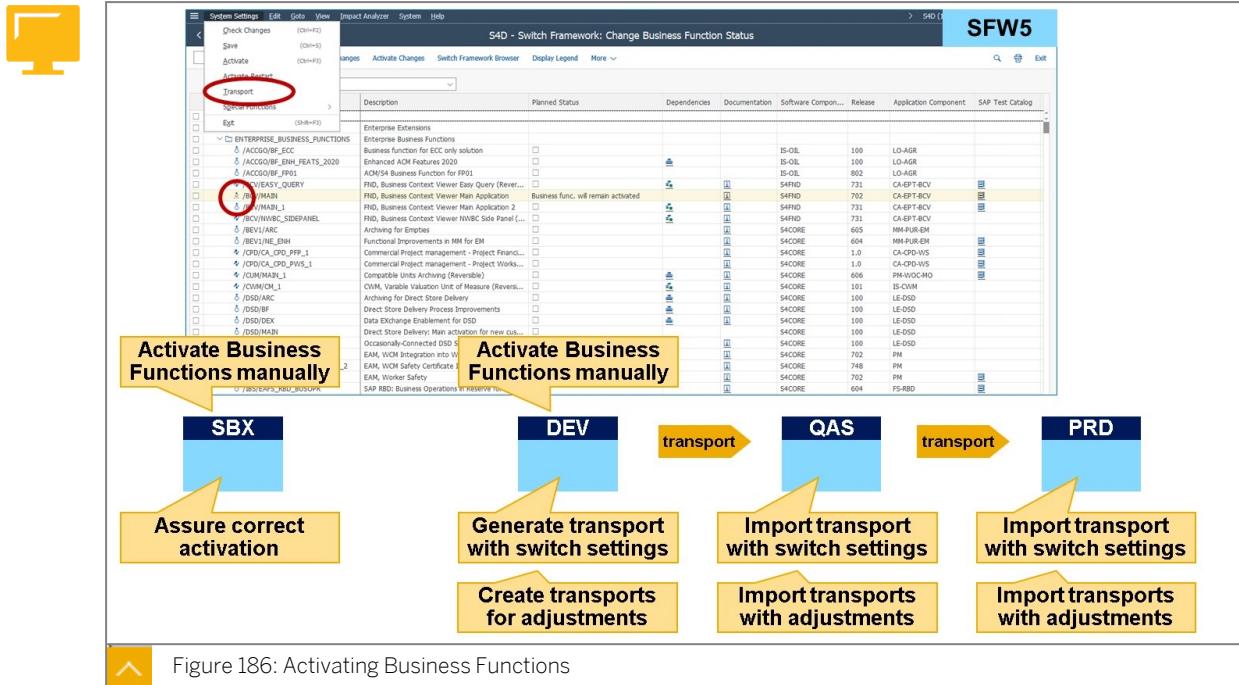
You can activate new business functions by using the *Switch Framework*. Without activation of the business functions, the system would behave as in the source release of the upgrade. Only required business functions should be activated. This significantly reduces the effort of testing, adjusting customizing, adjusting own development, training end users, and others.

**Hint:**

This technique is available in SAP S/4HANA Server, SAP ECC, and other AS ABAP-based SAP systems.

Because most business functions can't be de-activated once they are activated, the activation and first testing should be done in a sandbox system (if the wrong business function is activated, the sandbox system can be reset). The required business functions can then be activated in the development system using transaction SFW5. Now, in transaction SFW5, a transport request should be created which contains the activated switches. This transport request is then transported to the quality assurance system. Here, testing takes place. After adjusting customizing and adjusting development via the development system, the transport

request containing the switch settings can be imported together with the adjustment transport requests into the production system.



The transactions used for handling the *Switch Framework* are:

- SFW5 – activate/deactivate business functions
- SFW\_BROWSER – investigate, which enhancements are related to the business function in question
- SFW1 – create switches (only customer enhancements)
- SFW2 – create business functions (only customer enhancements)
- SFW3 – create business function sets for industry solutions (only customer enhancements)

The *Switch Framework* enables optional activation of business functions. With the *Switch Framework*, you can control the activation of repository objects. Activating a business function triggers switches, which then control the execution of the code enhancements. These switches ensure that you only work with the new functions if you have activated them. All functional changes and the impact of an activated business function are made transparent in advance by the documentation. Note that, once a business function is activated you usually can't reverse it. The activation process starts a batch job in your SAP system that automatically performs all changes in the SAP system.



#### Note:

The *Switch Framework* is a proven concept; it was already used to retrofit industry solutions in SAP ECC 6.00.



## LESSON SUMMARY

You should now be able to:

- List capabilities of SUM
- Describe important steps for planning and performing an SAP System Upgrade or an SAP S/4HANA Conversion
- Outline the idea of business functions



## Learning Assessment

- When applying an SAP Note with the SAP Note Assistant, neither a developer key nor an object key is necessary for applying the SAP Note itself but both keys can be necessary in systems of SAP Business Suite for manual activities related to the SAP Note.

*Determine whether this statement is true or false.*

- True
- False

- Which tools can be used to apply SAP Support Packages?

*Choose the correct answers.*

- A Support Package Manager (SPAM)
- B Software Update Manager (SUM)
- C Transport Management System (TMS)
- D SAP Note Assistant (SNOTE)

- It is possible to perform an SAP system upgrade from SAP ECC 6.05 to SAP S/4HANA Server 2022.

*Determine whether this statement is true or false.*

- True
- False

- Modification adjustments should be performed by SAP system administrators.

*Determine whether this statement is true or false.*

- True
- False

5. You are running a three-system landscape. You want to activate a business function with the Switch Framework (transaction SFW5). Bring the following steps into the right sequence, regarding the activation procedure.

*Arrange these steps into the correct sequence.*

- In the development system, activate the correct business functions with SFW5.
- In the quality assurance system, import the transport request with STMS and perform quality assurance testing.
- In the production system, import the transport request with STMS.
- In the development system, create a transport request containing the activated business functions with SFW5 and release the transport request.
- In a temporary sandbox system, activate the business functions for test purposes with SFW5.

## Learning Assessment - Answers

- When applying an SAP Note with the *SAP Note Assistant*, neither a developer key nor an object key is necessary for applying the SAP Note itself but both keys can be necessary in systems of SAP Business Suite for manual activities related to the SAP Note.

*Determine whether this statement is true or false.*

- True  
 False

You are correct! When applying an SAP Note with the *SAP Note Assistant*, neither a developer key nor an object key is necessary for applying the SAP Note itself but both keys can be necessary for manual activities related to the SAP Note, for example, for creating a package or changing a table structure in systems of SAP Business Suite.

- Which tools can be used to apply SAP Support Packages?

*Choose the correct answers.*

- A Support Package Manager (SPAM)  
 B Software Update Manager (SUM)  
 C Transport Management System (TMS)  
 D SAP Note Assistant (SNOTE)

You are correct! You can apply SAP Support Packages using SPAM or SUM. TMS is used to import transport requests. SNOTE is used to apply SAP Notes.

- It is possible to perform an SAP system upgrade from SAP ECC 6.05 to SAP S/4HANA Server 2022.

*Determine whether this statement is true or false.*

- True  
 False

You are correct! You can't perform an upgrade from SAP ECC 6.05 to SAP S/4HANA Server 2022, but you can perform an **SAP S/4HANA Conversion** from SAP ECC 6.05 to SAP S/4HANA Server 2022.

4. Modification adjustments should be performed by SAP system administrators.

*Determine whether this statement is true or false.*

True

False

You are correct! Modification adjustments should be performed by developers and not by SAP system administrators.

5. You are running a three-system landscape. You want to activate a business function with the Switch Framework (transaction SFW5). Bring the following steps into the right sequence, regarding the activation procedure.

*Arrange these steps into the correct sequence.*

**2** In the development system, activate the correct business functions with SFW5.

**4** In the quality assurance system, import the transport request with STMS and perform quality assurance testing.

**5** In the production system, import the transport request with STMS.

**3** In the development system, create a transport request containing the activated business functions with SFW5 and release the transport request.

**1** In a temporary sandbox system, activate the business functions for test purposes with SFW5.

You are correct! First you activate the business functions for test purposes with transaction SFW5 in a temporary sandbox system, then you activate the correct business functions with transaction SFW5 in the development system. You then create a transport request containing the activated business functions with transaction SFW5 in the development system and release the transport request. Next, you import the transport request with transaction STMS in the quality assurance system and perform quality assurance testing, and finally, you import the transport request with transaction STMS in the production system.

# UNIT 8

# Appendix: Enhancements to the Change and Transport System

## Lesson 1

Enhanced CTS - The Basic Idea

329

## Lesson 2

Configuration of the Change and Transport System for the enhanced CTS

337

## Lesson 3

Transport of non-ABAP Objects

355

## Lesson 4

Outlook: Integration of ABAP Development Into CI / CD (DevOps) Processes with the Help of gCTS 369

## UNIT OBJECTIVES

- Explain the basic idea of the enhanced CTS
- Explain the configuration steps for the enhanced CTS
- Perform transports of non-ABAP objects with the help of the ABAP Change and Transport System
- List use cases for the enhanced CTS
- Outline the transport options of SAP HANA content
- Outline the basic idea of DevOps and gCTS



# Enhanced CTS - The Basic Idea

## LESSON OVERVIEW

The ABAP *Change and Transport System* (CTS) also lets you transport Java objects (J2EE, JEE) and SAP-specific non-ABAP technologies (such as SAP HANA Delivery Units or SAP NetWeaver Portal iViews) in your landscape. This capability is known as enhanced CTS.

This lesson describes the basic idea of the enhanced CTS.

## Business Example

Your company uses the ABAP *Change and Transport System* (CTS) to perform transports in transport landscapes based on SAP systems with AS ABAP. In addition, your company is performing some development on applications for SAP HANA databases.

As a member of the transport administration team, you want to know if you can transport SAP HANA Delivery Units (TGZ files) with the help of CTS as well.



## LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Explain the basic idea of the enhanced CTS

## Enhanced CTS - The Basic Idea

### Classical Transport Scenario

In an SAP system landscape, you have the option to develop in ABAP, according to the J2EE / JEE standard, or to use SAP-specific non-ABAP technology, such as Web Dynpro Java, SAP Enterprise Portal, SAP HANA content or even to use SAP Business Technology Platform to extend your SAP applications. This is not a fight one technology versus the other, but different approaches to a solution for business needs. You choose from these options based on your preferences, knowledge that is already available in one of the technologies, or with respect to specific advantages of one technology in certain areas. Therefore, in any bigger landscape you will find many or even all of these objects. When you, for example, use SAP Enterprise Portal to provide users with role-based access to a new functionality in your SAP ECC Server backend system, you have to synchronize the update of both the SAP Enterprise Portal and the SAP ECC Server runtime, which even might be connected to other SAP systems with the help of SAP Process Orchestration. Or imagine an SAP S/4HANA Server system for which you want to extend existing business processes with the help of SAP Business Technology Platform.

So there are tools for ABAP and non-ABAP transports available, but (up to some SAP Support Package stack for SAP NetWeaver 7.0) there was no central control of non-ABAP transports and there was no central control of all transports into productive SAP systems.

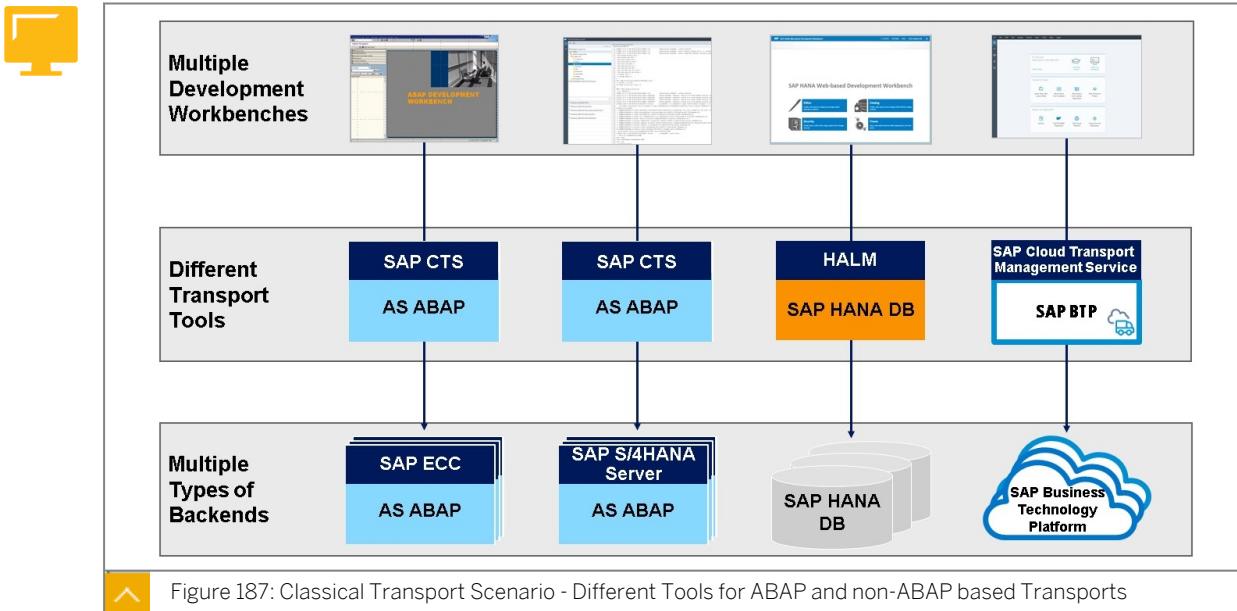


Figure 187: Classical Transport Scenario - Different Tools for ABAP and non-ABAP based Transports

### The New Option

With the enhancements of the ABAP Change and Transport System (CTS for short, or *enhanced CTS*, sometimes also abbreviated as *CTS+*, respectively, if specifically the enhanced capabilities are meant), available with some SP Stack of SAP NetWeaver 7.0, the CTS has also been enhanced to handle the transport of non-ABAP objects.

Enhanced CTS provides the option to transport objects which have been created in programming languages other than ABAP via the *ABAP Change and Transport System*. The goal is to administer non-ABAP systems in a CTS transport domain in Application Server ABAP and, therefore, to have one transport tool (the CTS) that supports all workbenches and applications when it comes to transports. The tools for creating applications and creating content remain the same. Options to attach applications or contents to a CTS transport request are integrated into the different workbenches. The level of integration is different for each workbench.

In this way, the enhanced *Change and Transport System* provides a unified transport tool and makes the work of administrators easier when executing imports. You only need access to one tool to execute imports for different backend systems. The deployment tools are called automatically. CTS allows you to monitor imports and analyze errors. The import queues and import histories in CTS give an overview of imports that have already been executed or which are planned.

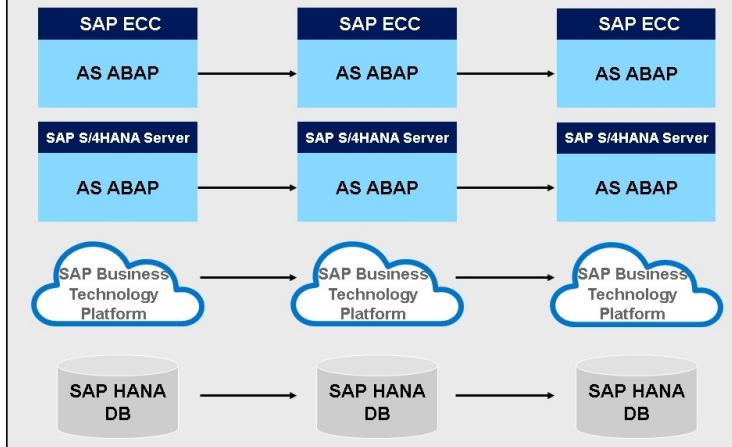
SAP provides a close integration into CTS for some SAP applications. This means that non-ABAP objects can be attached to transport requests directly from the SAP application. Examples of closely integrated applications are:

- SAP HANA Database
- Lifecycle Management Console (LCM) of SAP BusinessObjects Business Intelligence (BI)
- SAP NetWeaver Development Infrastructure (NWDI)
- SAP Enterprise Portal
- Process Integration (PI)

**Note:**

In some scenarios, you need to perform manual actions when you add objects to transport requests ("loose coupling"). Some object types require you to perform manual actions after the deployment as well.

Having this in mind, you can integrate additional applications into CTS such as *SAP Business Technology Platform* (as long as you only have to transport contents in the form of Multitarget Application archives) or *SAP Mobile Platform*. For more details and for a list of applications that can be integrated, see the section *Enhanced Change and Transport System* on SAP Support Portal, area *Tools* → *Software Logistics Tools* → *Enhanced Change and Transport System (CTS+)* and the wiki on software logistics (<https://wiki.scn.sap.com/wiki/display/SL>), area *Change and Transport System* → *Enhanced Change & Transport System (BC-CTS-PLS)*.

**One central transport tool****Enhanced CTS (CTS+)**

Multiple backends and transport landscapes

Figure 188: The New Option – Using the Enhanced CTS

An administrator of your AS ABAP system can now also manage the transports for the non-ABAP parts of your SAP system landscape. By combining and synchronizing transports in a comprehensive approach, the enhanced Change and Transport System therefore solves transport issues.

**Note:**

The enhanced CTS can also be used in the SAP Solution Manager *Change Request Management* scenario. Therefore, the enhanced CTS enables *Change Request Management* to support the entire Business Process Platform for SAP Business Suite.

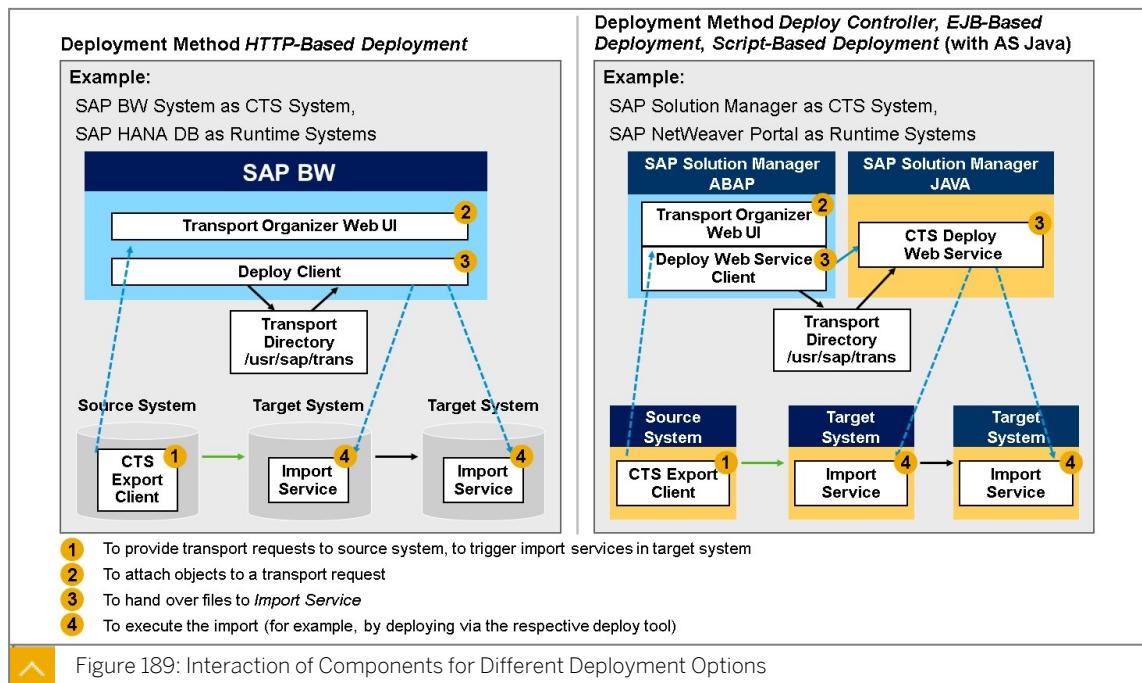
## System Requirements

To use the enhancements of the *Change and Transport System*, both the CTS system(s) and the runtime SAP systems (source system, target systems) must fulfill certain requirements, which are listed in the remaining part of this lesson.

**Note:**

In this lesson, the term *CTS system* refers to an SAP system based on AS ABAP which acts either as transport domain controller system or as communication system for the non-ABAP systems in question.

The main concepts of the *Change and Transport System* (such as TMS systems, transport layers, transport routes) are still valid and required for a landscape with the enhanced CTS. The main parts of the enhanced CTS are shown in the following figure and are explained below. The decision where to configure these components depends on the use case (such as SAP HANA, SAP Enterprise Portal, SAP NetWeaver Development Infrastructure, SAP Business Technology Platform) and on the system landscape, especially when a complex system landscape needs to be enabled for the (enhanced) CTS functionality.



Dependent on the deployment method (see the figure “Interaction of Components for Different Deployment Options”), the following components must be configured before you can use the *enhanced Change and Transport System*:

- *CTS Deploy Web Service* (AS Java)

This is a Java Web service responsible for communicating with the deployment tools in non-ABAP systems. The transport control program *tp* communicates with the *CTS Deploy Web Service* when deploying the non-ABAP objects.

- *Deploy Web Service Client* (AS ABAP)

This client on the ABAP side makes it possible for the transport control program *tp* on the AS ABAP to communicate with the *CTS Deploy Web Service* on the AS Java.

The Deploy Web service client is comprised of the logical port *CTSDEPLOY* (which contains the name of the *CTS Deploy Web Service* and additional parameters) and the HTTP connection *CTSDEPLOY* (which contains the destination of the *CTS Deploy Web Service*).

**Note:**

The *CTS Deploy Web Service* and the *Deploy Web Service Client* are only required if the application that you want to transport using CTS uses one of the following deployment methods:

- *Software Deployment Manager or Deploy Controller*
- *EJB-Based Deployment (application-specific)*
- *Script-Based Deployment (application-specific)*

If the application uses *HTTP-Based Deployment (application-specific)* as deployment method, the *CTS Deploy Web Service* and the *Deploy Web Service Client* are not required.

- *Deploy Client (AS ABAP)*

The *Deploy Client* (not to be mixed with the *Deploy Web Service Client* from the item above) consists of an HTTP destination on the CTS system for every target system (that is to every system where you would like to execute imports).

**Note:**

The *Deploy Client* is only required if the application that you want to transport uses the deployment method *HTTP-Based Deployment (application-specific)*.

- *Transport Organizer Web UI (AS ABAP)*

The *Transport Organizer Web UI* is an ABAP Web Dynpro application that allows you to create and edit transport requests for non-ABAP systems. The communication system where the *Transport Organizer Web UI* is located must be a Unicode system.

**Note:**

The *communication system* is an AS ABAP-based SAP system on which certain transport tools are called that do not exist on the non-ABAP system in question:

- For source systems, the communication system covers the *Transport Organizer Web UI*.
- For target systems, the communication system triggers *tp* together with the *Deploy (Web Service) Client*.

By default, the transport domain controller is used as communication system.

In the context of the enhanced CTS, the communication system for a non-ABAP system is also labeled as *CTS system*.

- *Transport Directory*

Consistent transport directory for AS Java and AS ABAP.



Note:

*R3trans* creates a subfolder in the <DIR\_TRANS>/data directory with the name of the transport request. All attached non-ABAP files of that transport request are copied into that folder. During import, these non-ABAP files are transferred to the *Import Service* (for example, the *Deploy Controller* in case of AS Java based SAP systems) of the target system.

If you can't use a consistent transport directory, you can alternatively use a *Java Connector* destination for the data transfer.

- *CTS Export Client* (source system)

The objects to be transported are created or changed in the source system. The different export mechanisms in the workbenches of the applications and the export service take over the role of the *CTS Export Client*. The *CTS Export Client* communicates with the transport system by proposing a transport request for selection and attaching non-ABAP objects from the application to the transport request. This is called "close coupling".



Note:

For more information, see the documentation or the guide on CTS integration of each application which is available on SAP Support Portal, area *Tools* → *Software Logistics Tools* → *Enhanced Change and Transport System (CTS+)*.

- *Import Service* (target system)

The objects from the source system are imported into the target system. The various deployment tools (*SDM*, *Deploy Controller*, and so on) take on the role of the import service on the target system.



Note:

For more information, see the documentation or the guide on CTS integration of each application which is available on SAP Support Portal, area *Tools* → *Software Logistics Tools* → *Enhanced Change and Transport System (CTS+)*.

Even if you want to use enhanced CTS only for non-ABAP systems, you still require an AS ABAP based SAP system for the transport domain controller / CTS system.

Some configuration steps are required to enable your systems to transport non-ABAP objects using the *Change and Transport System*. Since transporting non-ABAP objects in CTS always means connecting an ABAP system with non-ABAP systems, you must make sure that the communication between these systems is set up in a secure way.

**Note:**

In former releases, some enhancements of the CTS functions (such as the *Transport Organizer Web UI – Web Dynpro Application CTS\_ORGANIZER*) have been part of the CTS Plug-In 2.0 (software component version *CTS\_PLUG 200*). Starting with SAP NetWeaver 7.40 SP10, changes to the enhanced CTS functions are delivered with Support Packages for SAP NetWeaver. For details, see SAP Notes [1665940](#) – *Installing/Updating SAP CTS Plug-In 2.0* and [2630240](#) – *Uninstalling CTS\_PLUG 200*.

### Example for an SAP System Landscape with the Enhanced CTS

The following figure shows an example of an SAP system landscape for the use with the *enhanced Change and Transport System*.

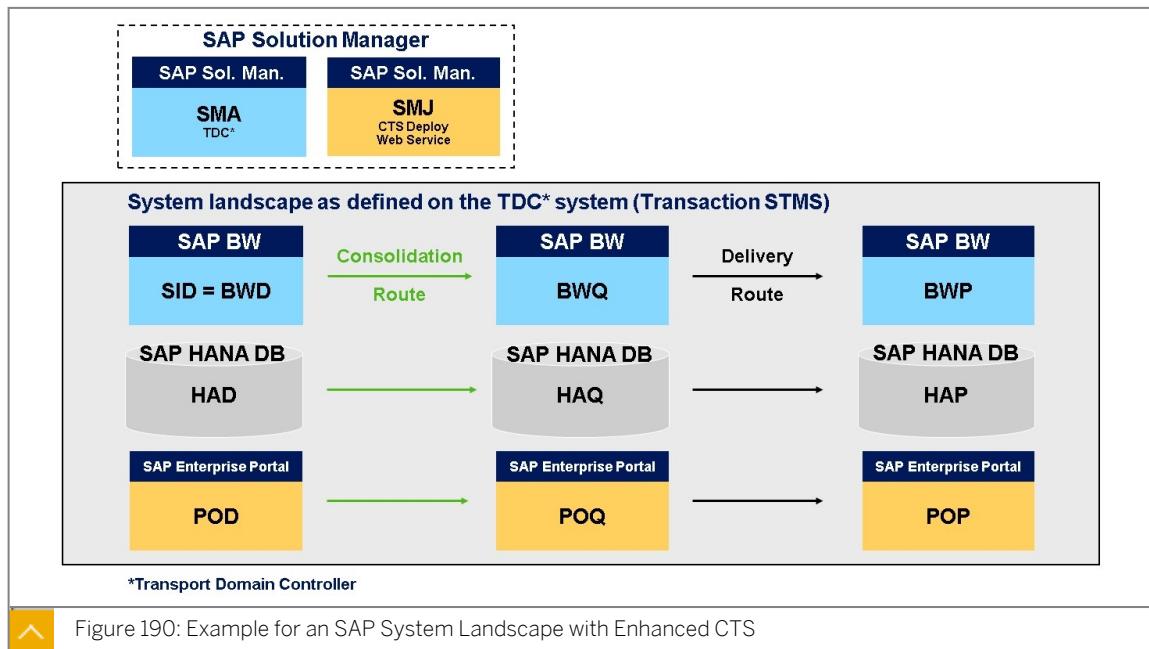


Figure 190: Example for an SAP System Landscape with Enhanced CTS

In this example, the transport domain controller (TDC) is located on an SAP Solution Manager ABAP system. The SAP Solution Manager ABAP system acts a CTS system.

Within transaction STMS on the TDC system (*System Overview* area), you can define both ABAP and *non-ABAP systems* (SAP systems) as well as configure Java Stacks for existing AS ABAP based SAP systems. The SAP system landscape shown in the figure above contains a three-system landscape for SAP Business Warehouse (SAP BW) systems. In addition, it contains a three-system landscape for SAP HANA databases (without any SAP system on top of them) in which native SAP HANA applications are to be developed. And finally there is a three-system landscape of SAP Enterprise Portal systems.



### LESSON SUMMARY

You should now be able to:

- Explain the basic idea of the enhanced CTS



## Configuration of the Change and Transport System for the enhanced CTS

### LESSON OVERVIEW

This lesson describes how to configure both the *Change and Transport System* and the runtime systems (source system, target systems) for transporting non-ABAP objects. First, the prerequisites for the CTS system and the runtime SAP systems are reviewed. Then the configuration steps that are needed to set up the CTS system accordingly are listed. Next, the creation of non-ABAP systems and the definition of transport routes between them are discussed. This lesson ends with an appendix about the configuration of various runtime systems for the use with the *enhanced CTS*.



#### Note:

This lesson uses terms and transactions that have not yet been explained in lessons of this class, but that are well known from preceding classes (SAPTEC, ADM100 or classes derived from these classes). The instructor won't have the time to explain all these terms and transactions in detail in this class.

### Business Example

Your company uses the ABAP *Change and Transport System* (CTS) to perform transports in transport landscapes based on SAP systems with AS ABAP. In addition, your company is performing some *SAP NetWeaver Portal* development and some development of *SAP HANA* content.

As a member of the transport administration team, you want to know if you can transport *SAP NetWeaver Portal* content files (*EPA* files) and *SAP HANA* content files (*TGZ* files) with the help of CTS as well.



### LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Explain the configuration steps for the enhanced CTS

### Configuration of the Change and Transport System for the Enhanced CTS

The following figure lists the main configuration steps that are necessary on the transport domain controller (TDC) system and the other CTS systems.

**Note:**

In this lesson, the term *CTS system* refers to an SAP system based on AS ABAP which acts either as transport domain controller system or as communication system for the non-ABAP systems in question.

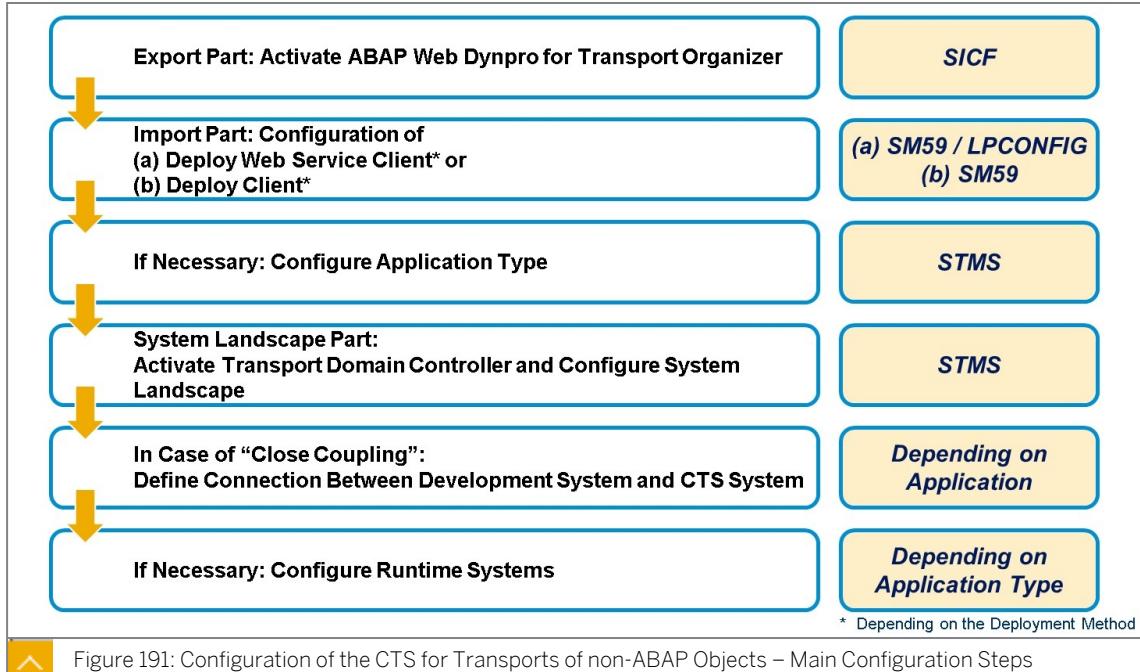


Figure 191: Configuration of the CTS for Transports of non-ABAP Objects – Main Configuration Steps

Note that the *CTS Deploy Web Service* runs on an AS Java but is called from the *Deploy Web Service Client*, which needs to be configured on the CTS system (because the CTS system needs to know how to connect to the *CTS Deploy Web Service*).

**Hint:**

This *Deploy Web Service Client* needs to be configured on every SAP system on which the *tp* program uses the *CTS Deploy Web Service* to trigger the deployment (for example: deployment using the *Deploy Controller*). For target systems based on AS Java (only), this will be the *communication system* for this target system as defined in the *System Overview* area of transaction STMS. Usually, this communication system will be the transport domain controller system.

For target systems based on SAP HANA DB (without any SAP system on top), the *Deploy Web Service Client* is not needed, however, you need to configure the *Deploy Client* instead.

All of these main steps will be discussed in more detail in the following subsections.

### The SAP System Landscape in this Class as an Example

The following figure points out the SAP system landscape (simplified) in this class.

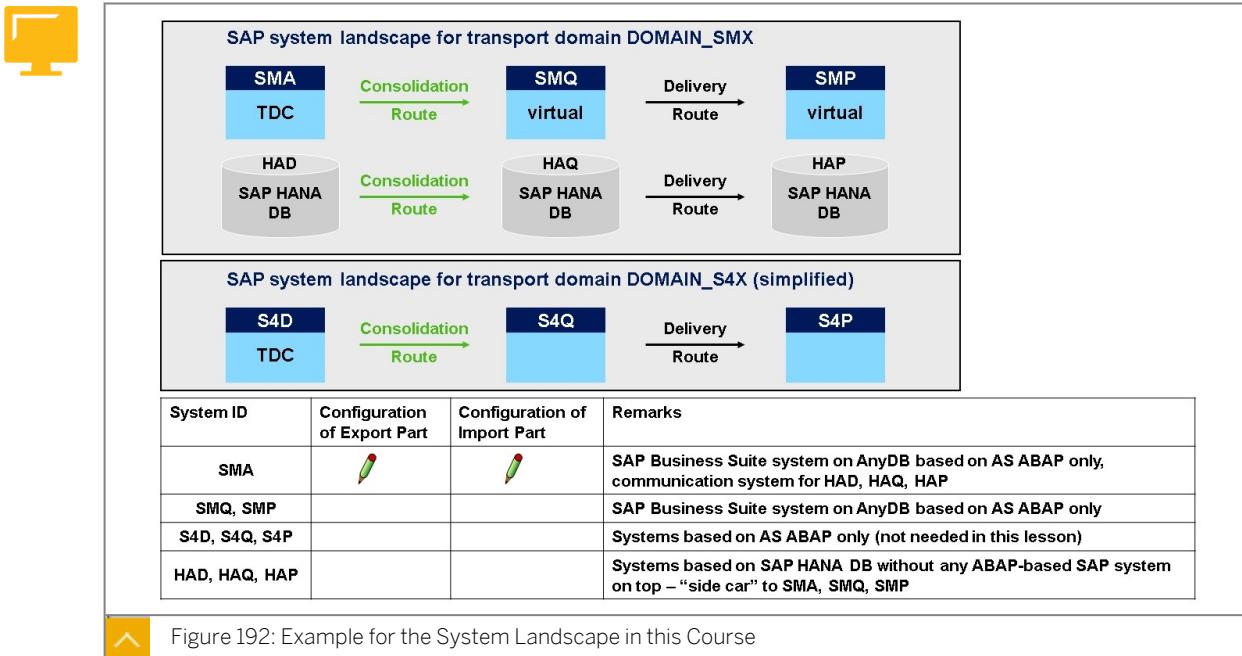


Figure 192: Example for the System Landscape in this Course

In this scenario, there is a three-SAP system landscape  $S4D \rightarrow S4Q \rightarrow S4P$  of SAP systems based on AS ABAP in transport domain *DOMAIN\_S4X*. This system landscape is not considered in this lesson, however.

In addition, there is a transport domain *DOMAIN\_SMX*. This transport domain is considered in this lesson. In this transport domain, there is an SAP system landscape  $SMA \rightarrow SMQ \rightarrow SMP$  of SAP systems based on AS ABAP. These systems are running on a database different from *SAP HANA DB*. And there is a three *SAP HANA DB* system landscape  $HAD \rightarrow HAQ \rightarrow HAP$  in this transport domain *DOMAIN\_SMX* without any SAP system on top. These three SAP systems will have the transport domain controller system *SMA* as communication system (defined in transaction *STMS*, area System Overview). *SMA*, therefore, acts as CTS system for these three *SAP HANA DB* based systems.

In this case, the import part needs to be configured on the *SMA* system, because *SMA* plays the role of the communication system for the *HAQ* and *HAP* system and the import into *HAQ*, for example, will, therefore, use *tp* on the *SMA* system.

The export part needs to be configured on the *SMA* system as well, because this SAP system acts as communication system for the *HAD* system so that transport requests for *HAD* (transport requests named *HADK9<number>*) will be created with the help of the *Transport Organizer Web UI* that is located on the *SMA* system.

## Export Part: Activation of the Service for the Transport Organizer Web UI

The screenshot shows the SAP Define Services interface. At the top, there is a header bar with the title "Export Part: Activate ABAP Web Dynpro for Transport Organizer" and a button labeled "SICF". Below the header, there is a "Filter Details" section with fields for "Virtual Host", "Service Path", "ServiceName" (set to "CTS\_ORGANIZER"), "Description", "Lang." (set to "English"), and buttons for "Apply", "Reset", and "Fine-Tune". The main area displays a tree view of "Virtual Hosts/Services" under "default\_host". The tree structure is as follows:

- default\_host
  - sap
    - bc
    - webdynpro
      - sap
      - CTS\_ORGANIZER**

Next to each node, there is a "Documentation" column and a "Reference Service" column. For "CTS\_ORGANIZER", the documentation says "VIRTUAL DEFAULT HOST" and the reference service is "Transport: Organizer".

**Activate Service CTS\_ORGANIZER**

Figure 193: Configuration of the ABAP Web Dynpro Application for CTS (Export Part)

SAP provides an ABAP Web Dynpro application (*CTS\_ORGANIZER*) that helps you to create transport orders and to attach objects. Before you can use the *Transport Organizer Web UI*, you need to activate the *CTS\_ORGANIZER* service in transaction *SICF* on the CTS system (AS ABAP part).

## Import Part: Configuration of CTS Deploy Web Client or the Deploy Client

The *CTS Deploy Web Service* (which is part of the AS Java) connects the ABAP transport tools with the deployment tools in the non-ABAP applications. When deploying the non-ABAP objects, the transport control program *tp* communicates with the *CTS Deploy Web Service* with the help of the *Deploy Web Service Client*.

The following figure shows the necessary steps to configure the import part on the CTS system. It depends on the deployment method whether you need to configure the *Deploy Web Service Client* or the *Deploy Client*.



### Note:

The *CTS Deploy Web Service* and *Deploy Web Service Client* are only required if you use an AS Java for one of the following deployment methods:

- Software Deployment Manager or Deploy Controller
- EJB-Based Deployment (application-specific)
- Script-Based Deployment (application-specific)

If you use *HTTP-Based Deployment* (application-specific) as deployment method, however, you need to configure the *Deploy Client* (that is, create an HTTP destination that points directly to the target system). For details about this HTTP destination, refer to a following section of this lesson and to the application-specific (configuration) guide.

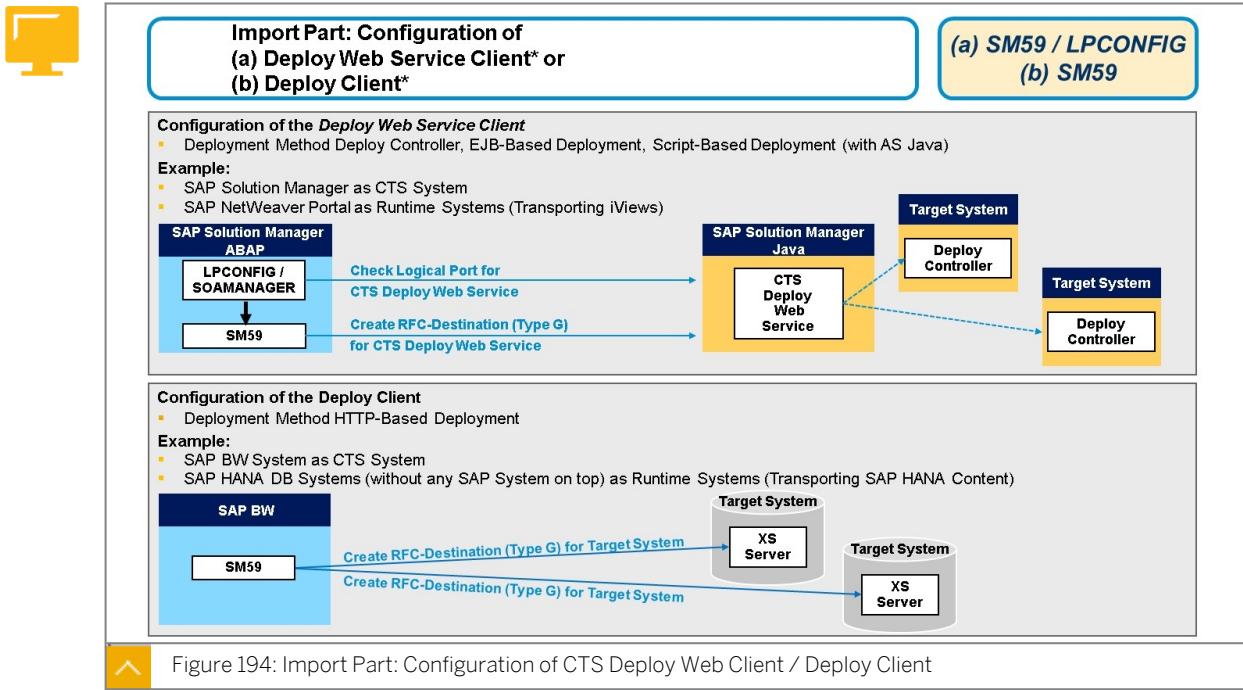


Figure 194: Import Part: Configuration of CTS Deploy Web Client / Deploy Client

### Configuration of the Deploy Web Service Client

If you need to configure the *Deploy Web Service Client* to connect to the *CTS Deploy Web Service*, proceed as follows (see figure above):

1. Check the *CTS Deploy Web Service*. To do this, open a Web browser, enter the URL for the *SAP NetWeaver Administrator* on the SAP system on which the *CTS Deploy Web Service* is running. In the *Single Service Administration: Service Definitions* application, search for the *DeployProxy* and check that it is available on that server.
2. Configure a logical port for the *CTS Deploy Web Service*. As of enhancement package 1 for SAP NetWeaver 7.0, SAP ships the logical port *CTSDEPLOY* for the proxy class *CO\_TFLDEPLOY\_PROXY\_VI\_DOCUMENT* in client 000 with the standard installation. This logical port refers to the RFC destination *CTSDEPLOY*. Therefore, you only need to check in transaction *LPCONFIG* (in client 000) of the CTS system that this logical port exists.
3. Create a new HTTP destination with the name **CTSDEPLOY** as RFC destination of connection type G in transaction *SM59*. The name of this RFC destination will be needed for the logical port of the *CTS Deploy Web Service*. On the *Technical Settings* tab, enter the host name and the port of the AS Java on which the *CTS Deploy Web Service* is running. On the tab *Logon & Security*, choose *Basic Authentication* as log on procedure (you may need to confirm a dialog box). As log on user, enter a user ID of the AS Java on which the *CTS Deploy Web Service* is used. This user must have the authorization to use the *CTS Deploy Web Service*.

Large objects can cause timeouts in deployments. Therefore, you may need to set a timeout parameter for the Internet Communication Manager (ICM) on the CTS system (AS ABAP part), and on target systems based on AS ABAP + Java, to be longer than the longest expected deployment time. This can be done in the definition of the RFC destination (transaction *SM59*, tab *Special Options*) as well.

### Configuration of the Deploy Client

For deployment method *HTTP-Based Deployment (application-specific)*, you need to configure the *Deploy Client*. For this, on the CTS system you have to create an HTTP

connection as RFC destination for every target system, that is for every system where you would like to execute imports (see the figure “Import Part: Configuration of CTS Deploy Web Client / Deploy Client”).

For this, proceed as follows:

1. Create a new HTTP connection to an external server with a self-explaining name (such as **<SID of the target system>\_DESTINATION**) as RFC destination of connection type G in transaction SM59. The name of this RFC destination will be needed later during the definition of the target system in transaction STMS.
  - On the *Technical Settings* tab, enter the host name and the port of the target host – depending on your security settings you may enter the HTTP or HTTPS-Port as Service No. The *Path Prefix* depends on the application type (for example **/sap/hana/xs/lm/slpslp\_xsjs** for SAP HANA DB content transports).
  - On the tab *Logon & Security*, choose *Basic Authentication* as *logon procedure* (you may need to confirm a dialog box). As logon user, enter a user ID on the target system.



**Note:**

This user must have the authorization to process the import (for example, for SAP HANA DB content transports, the user needs to be assigned to the role *sap.hana.xs.lm.roles::SLP\_CTS\_deploy\_admin* on the SAP HANA DB system).

- Large objects can cause timeouts in deployments. Therefore, you may need to set a timeout parameter for the Internet Communication Manager (ICM) on the CTS system (AS ABAP part), and on target systems based on AS ABAP + Java, to be longer than the longest expected deployment time. This can be done in the definition of the RFC destination (transaction SM59, tab *Special Options*) as well.
2. Repeat this step for all target systems.
  3. Depending on the network topology, the Application Server ABAP system might not be able to directly connect to the Internet. In this case, a proxy needs to be configured that routes the download requests to the Internet. As the communication uses client 000, make sure that the *HTTP Proxy* settings for these RFC destinations in client 000 are configured according to your network configuration. For this, choose the menu path *Extras → HTTP Proxy Configuration* in transaction SM59 and:
    - on the *Global Settings* tab page, make sure that the *proxy setting is active* and specify addresses that should not be accessed using the proxy.
    - on the *HTTPS Protocol* tab page, enter the connection information for the proxy server and choose OK.



**Note:**

When you use the enhanced CTS for the integration of *SAP Business Technology Platform* (SAP BTP), in addition the CTS system must validate the server certificate in order to ensure the server identity and to prevent “man-in-the-middle” attacks. This means that the root certificate of the certificate authority that was used to sign the *SAP BTP Cloud Foundry* server certificate needs to be imported into transaction *STRUST* of the CTS system.

## Configuration of the Application Type

The enhanced *Change and Transport System* (CTS) supports the transport of objects of specific non-ABAP applications. The supported objects can vary from release to release. You have the option to connect further non-ABAP applications (also non-SAP applications) with the CTS. In this way, objects from the connected applications can also be transported with the CTS.

In order to use CTS with your application, it might be necessary that you have to make the application known in CTS. For this, you need an *application type* which will then be used as unique identifier for your application (such as *SAP HANA DB*) in CTS.

You can connect these applications with CTS for each domain in the following way:

1. Configure the source systems for exporting objects of the application to be connected.

This is necessary so that objects in the source systems can be exported using the *Change and Transport System*.

In the *Transport Management System* (transaction code STMS) on the transport domain controller system, enter the *System Overview* area and choose *Extras → Application Types → Configure*. Here, you can create a new entry for your application type.



**Note:**

For the SAP HANA DB integration with CTS, the *application type HDBLM* is used.

For the integration of *Multi-Target Applications* (MTAs) on *SAP Business Technology Platform (Cloud Foundry environment)* into CTS, the *application type SCP\_CF* with *Support Details BC-XS-SL-DS* is used (for which you may need to create an entry manually).

As a result, the application is connected with the *Change and Transport System*. You can now select the application in the *Transport Organizer Web UI* when choosing the object (file) that is to be attached to a transport request.

2. Configure the target systems for transporting the objects of this application.

This is necessary so that the objects of the application can be imported into the target systems (this procedure assumes that the target systems of the application are already configured in your transport landscape):

In the *Transport Management System* (transaction code STMS) on the transport domain controller system, enter the *System Overview* area. Select the target system in question with a double click on it. On the resulting screen choose the menu path *Goto → Application Types → Deployment Method*.

Create a new entry and select the *Application Type* (or *Application ID*) using the value help. Select the required *Deploy method* from the selection list.

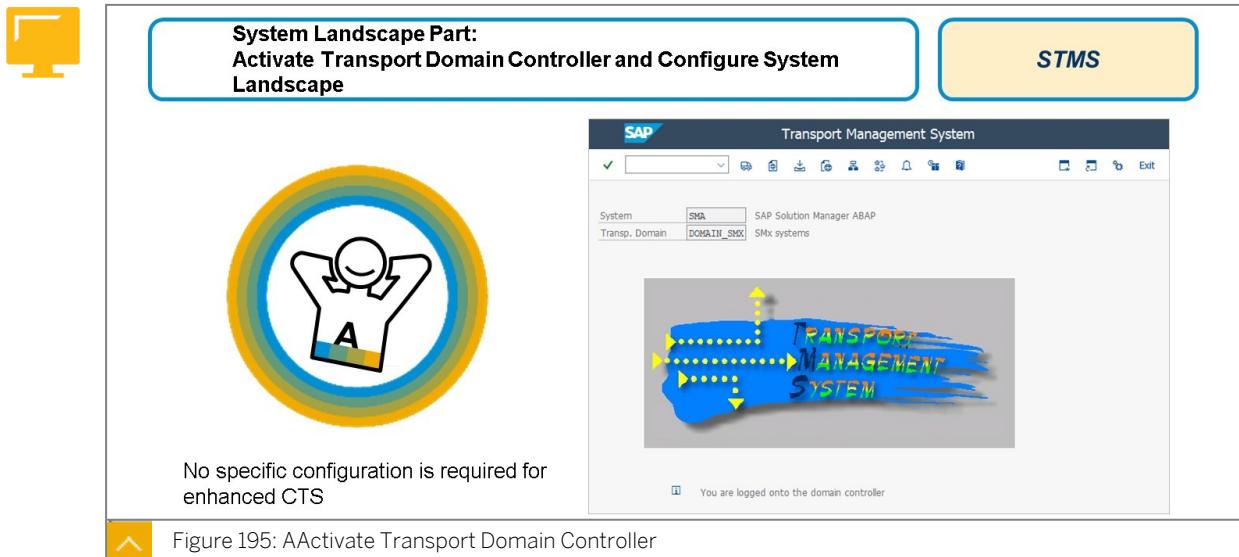
**Hint:**

The *HTTP destination* field contains the name of the HTTP destination (as defined before in transaction SM59) that is used for deployment. This field is used only for *Deploy method HTTP-based Deployment (application-specific)*.

The *Deploy URI*, *User* (for a technical deploy user) and *Password* fields are optional for *HTTP-based Deployment (application-specific)*.

After these steps, you can transport objects of the connected application in the *Change and Transport System*.

### SAP System Landscape Part I: Activate Transport Domain Controller



Before you can work with the *Transport Management System* (TMS), you must configure it for all SAP systems in your SAP system landscape.

The TMS configuration includes the configuration of the transport domain. In this step, you define which SAP systems in your SAP system landscape form a transport domain, and which SAP system is to be the transport domain controller.

**Note:**

The procedure to activate the TMS is the same as for “normal” ABAP transports, therefore, in this step, there is no specific configuration required for the enhanced functionality of CTS.

### SAP System Landscape Part II: Configuration of SAP System Landscape

The following roadmap shows the necessary steps to configure the SAP system landscape on the transport domain controller system.

**Note:**

The configuration steps differ in detail with respect to both the SP Stack level and the type (*SAP PI* versus *SAP Enterprise Portal* versus *SAP HANA DB* content versus *SAP Business Technology Platform integration*) of the SAP systems in question.

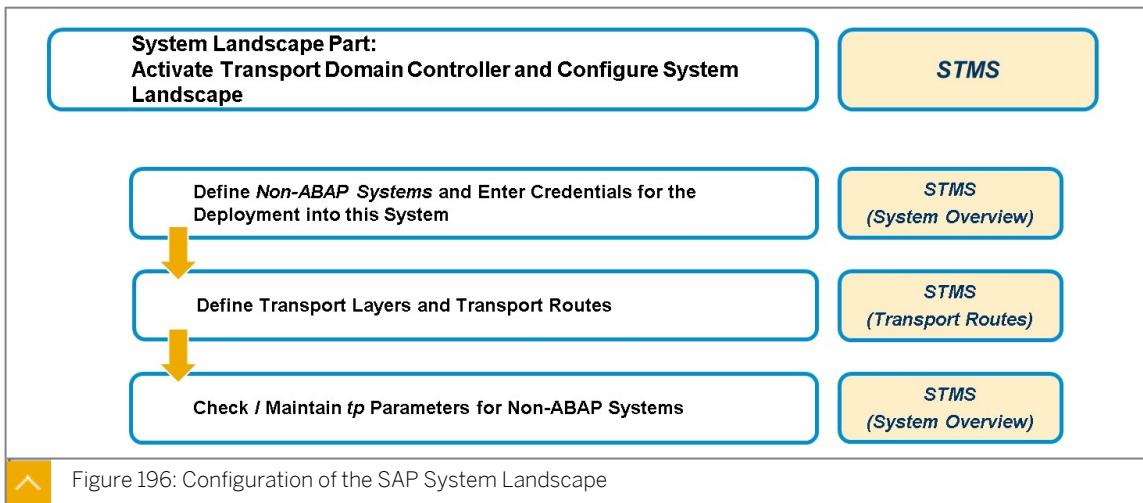


Figure 196: Configuration of the SAP System Landscape

### **Define non-ABAP Systems / Java Stack Configurations and Enter Credentials for the Deployment into this System**

Consider a three-system landscape as an example. Depending on the usage type of the runtime system(s), you need to create the following *non-ABAP systems* or perform *Java Stack Configurations* in transaction **STMS** (*System Overview* area) on the transport domain controller system:

- For transports between SAP systems based on AS ABAP (only), you don't need any non-ABAP Systems in transaction **STMS**.
- For transports between non-ABAP systems (such as *SAP HANA DB* systems, SAP systems based on AS Java only or *SAP Business Technology Platform integration*), you need to define three non-ABAP systems in transaction **STMS** representing your development system, the quality assurance system and the production system. Together with the CTS system, you need four SAP systems in transaction **STMS**.

**Note:**

For any non-ABAP System, choose the transport domain controller system (TDC system) to be the *communication system*.

### **Define Transport Layers and Transport Routes**

Next you need to define transport layers and transport routes (*consolidation routes* and *delivery routes*) between the non-ABAP SAP systems in question. The transport layers and the transport routes that are needed depend on the SAP systems in question.

The definition of transport layers and transport routes is done on the transport domain controller system (TDC system) in transaction **STMS** (*Transport Routes* area).

## Check / Maintain tp Parameters for Non-ABAP Systems

The following figure shows the necessary *tp* parameters that need to be maintained on the transport domain controller system for the non-ABAP SAP systems in question.

**System Landscape Part:**  
Activate Transport Domain Controller and Configure System Landscape

**Check / Maintain tp Parameters for Non-ABAP Systems**

**STMS**

**STMS (System Overview)**

**Check the following tp parameters on the CTS system for the (non-ABAP) systems in question:**

	tp parameters
<b>Source System (Development System or Transport Domain Controller)</b>	NON_ABAP_SYSTEM* NON_ABAP_WBO_CLIENT WBO_GET_REQ_STRATEGY WBO_REL_REQ_STRATEGY NON_ABAP_WBO_INBOX++ WBO_FILE_TRANSFER
<b>Target system (Quality Assurance System and Production System)</b>	NON_ABAP_SYSTEM* DEPLOY_DATA_SHARE DEPLOY_WEB_SERVICE DEPLOY_URL (or DEPLOY_XI_URL or DEPLOY_SLD_URL)

\* needed only for systems without AS ABAP  
++ needed only in case of "loose coupling"

Figure 197: Adjustment of tp Parameters for non-ABAP Systems

The necessary *tp* parameters differ between the source system on the one hand and the target systems on the other hand. To check or maintain these parameters, use transaction STMS of the transport domain controller system (System Overview area, here double-click on the SAP system in question and then switch to the *Transport Tool* tab).

- *NON\_ABAP\_WBO\_CLIENT* specifies the client (on the CTS system or on the development system – if it contains an AS ABAP stack) in which transport requests can be created and released. This parameter is needed to enable the *Transport Organizer Web UI* and to restrict this Web UI for non-ABAP SAP systems to one client.
- The parameter *WBO\_GET\_REQ\_STRATEGY* (values: **SMART**, **TAGGED** or **CREATE**) defines the automatic **creation** of transport requests during the export process. The parameter *WBO\_REL\_REQ\_STRATEGY* (values **AUTO** or **MANUAL**) defines the automatic **release** of transport requests.
- The parameter *NON\_ABAP\_WBO\_INBOX* points to the directory in which the *Transport Organizer* looks for files that should be transported. This parameter is used for “loose coupling” only (file transfer using a mount directory or a share directory).
- If your SAP application is closely integrated with CTS (“close coupling”), you can attach objects directly from the application by default. The data is then transferred to the CTS system using *SAP Java Connector (JCo)*. In this case, the parameter *WBO\_FILE\_TRANSFER* is set to **STREAM** (this is the default value). If you need to transfer files using a share or a *Network File System* however, you must set the parameter *WBO\_FILE\_TRANSFER* to **SHARE**.
- The parameter *DEPLOY\_DATA\_SHARE* points to the directory where *tp* stores the data and the *CTS Deploy Web Service / Deploy Client* takes it for deployment to the runtime SAP systems (it is not needed if you use *HTTP-Based Deployment*). For SAP systems

based on AS ABAP (or AS ABAP + Java), the value of parameter *DEPLOY\_DATA\_SHARE* should point to the *data* subdirectory of the directory specified in profile parameter *DIR\_TRANS* on the SAP system in question. This parameter will be generated automatically but you may need to adapt the value of this parameter manually.



Note:

Instead of the parameter *DEPLOY\_DATA\_SHARE*, you can also use the parameter *CTS\_FILE\_PROVIDER\_URI*. With this parameter, you need to define an SAP Java Connector connection (JCo connection) for transferring data to the target system (instead of using a directory on operating system level). This parameter will only be evaluated if the parameter *DEPLOY\_DATA\_SHARE* is not set.

- The value of parameter *DEPLOY\_WEB\_SERVICE* must be the name for the logical port in transaction *LPCONFIG* (for example, *CTSDEPLOY*). This parameter is generated automatically. It will not be evaluated in case of *HTTP-based deployments*.
- If you want to trigger the deployment in the target systems, you need to specify the parameter *DEPLOY\_URL*. This parameter is generated automatically and should point to the URL of the deploy tool (for example, *Deploy Controller*) in the target system. This parameter is not used for *HTTP-based deployments*.



Note:

For deploying into an SAP PI system, the URL of the PI system is specified by parameter *DEPLOY\_XI\_URL*, for deploying into a *System Landscape Directory* (SLD) system, the parameter *DEPLOY\_SLD\_URL* is used.

- The parameter *CTS\_SYSTEM\_TYPE* classifies the systems in transaction *LMDB* (*Landscape Management Database*) for SAP Solution Manager scenarios. For non-ABAP source systems used in SAP Solution Manager scenarios, you must manually add this parameter as required by your scenario. For non-ABAP target systems, the parameter is generated automatically.

The import of Java applications into subsequent SAP systems will be performed by the ABAP transport control program *tp*. *tp* calls the *Deploy Client* or the *CTS Deploy Web Service* according to the *tp* parameters and depending on the deployment method.

In case of using the *Deploy Client* (deployment method *HTTP-based Deployment*), the credentials of the deploy user are stored in the definition of the RFC destination.

If using the *Deploy Web Service Client*, the *CTS Deploy Web Service* calls the deploy tool (for example the *Deploy Controller* (DC) for AS Java based SAP systems). For this, you need to specify a user with appropriate permissions to perform the import. This user must be specified on the transport domain controller system of the transport domain to which the SAP system belongs: in the *System Overview* area of transaction *STMS* on the transport domain controller system, double-click the SAP system in question and switch to the *Transport Tool* tab.

- If your SAP system is an SAP PI system, choose *Goto → XI User/ Password* and enter the credentials of the user who will start the deployment. This user must exist on the SAP PI system in question and the PFCG role *SAP\_XI\_CMS\_SERV\_USER* must be assigned to this user.

- If your SAP system is based on an AS Java only (for example an *SAP Enterprise Portal* system), the deployment will be performed by the *SDM* (SAP systems based on AS Java 6.40 or AS Java 7.0x) or by the *Deploy Controller* (SAP systems based on AS Java 7.10 or higher). You need to choose *Goto → SDM/DC User/ Password* and enter the appropriate credentials for the Java-based SAP system in question (either the SDM password or a user in the Java-based SAP system in question who is allowed to trigger the deployment).
- For the *System Landscape Directory* (SLD), the deployment procedure is a little bit different, however. Therefore, you need to specify a different user on the SLD application.

**Note:**

You need to perform this step for every SAP system in which you want to import non-ABAP objects. When you activate the *Deployment Service* for target systems (during the creation of a *non-ABAP system* or a *Java Stack Configuration* in transaction STMS), these credentials are maintained/stored automatically, however.

### Check TMS Settings

Once you have configured the TMS, there are some tests that you can perform to verify that there are no technical issues with the TMS configuration. For example, you should verify that the *tp* program and its parameter configuration are consistent. To do this, from the *System Overview* area of transaction STMS, follow the menu path *SAP System → Check → Transport Tool*. This executes tests verifying the *tp* interface, the transport profile, the RFC destinations, and connection to the databases. It returns a report of the results.

### Configuration of Runtime SAP Systems

In addition to the configuration of the transport domain controller and the CTS systems, you also may need to configure your runtime SAP systems (that means the SAP systems between which the export and import take place).

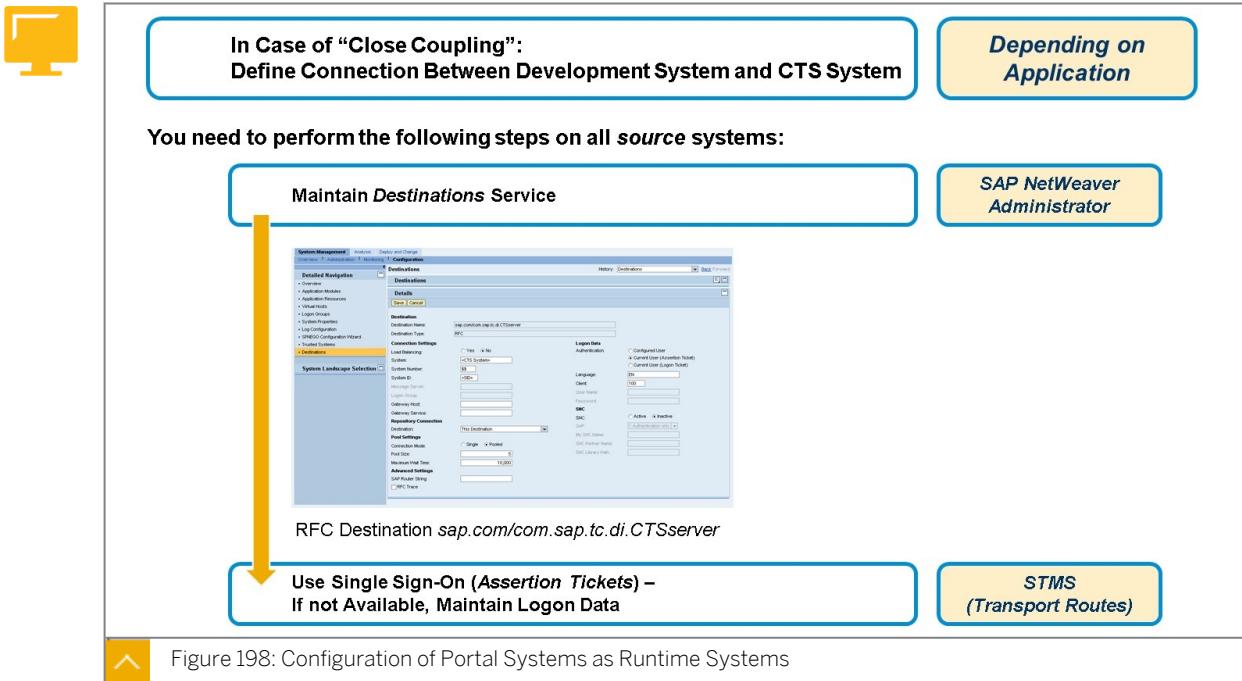
The necessary steps for the configuration of runtime SAP systems differ with respect to the type of these SAP systems.

**Note:**

For target systems based on AS ABAP only, no specific preparation steps are required for the use of enhanced CTS. You simply configure your SAP systems for the use with CTS "as usual". In addition, no specific release of the SAP systems is required, the CTS compatibility is sufficient in this case.

### Example: Configuration of SAP Enterprise Portal Systems as Runtime Systems

If you want to transport *SAP Enterprise Portal* content with the help of the enhanced CTS, however, you need to configure the *SAP Enterprise Portal* systems accordingly. The main steps are listed on the following roadmap:



Applications or services can establish connections to other services. When using such connections, you need to specify the remote service's address and the user authentication information to use for the connection. Many applications use the *Destinations* service for this purpose. You can manually create HTTP destinations and RFC destinations using the *Destinations* service in *SAP NetWeaver Administrator* (see figure above) or (for SAP systems based on AS Java 7.0x) in the *Visual Administrator* tool.

Before you can attach objects to transport requests directly in the application, you must first use RFC to connect your SAP application system (such as *SAP Enterprise Portal*) to the CTS system (to be more precise: to the SAP system that acts as *Communication System* for your *SAP Enterprise Portal* system). This enables your application to communicate with the transport system.

To create this RFC connection, open *SAP NetWeaver Administrator* of your SAP application development system and choose (*System Management* → ) *Configuration* → *Security* → *Destinations*. Create a destination of *Destination Type RFC* (for the development system as *Hosting system*) called (exactly) **`sap.com/com.sap.tc.di.CTSserver`**.

- In the *Connection and Transport Security Settings* step, enter the connection data of your CTS system.
- In the *Logon Data* step, select the *Current User (Assertion Ticket)* option as authentication for the connection to your CTS system. SAP recommends that you use Single Sign-On (SSO). If you cannot use SSO, select the *Technical User* option and enter your preferred language, a client, and an appropriate user name and password. Enter the client from the *tp* parameter *NON\_ABAP\_WBO\_CLIENT* on the CTS system. Use the value help for field *Destination Name* in area *Repository Connection*.
- In the *Specific Settings* step, use the default values for the *Pool Settings*. Finally choose *Finish* to save the destination.



Note:

In SAP PI runtime SAP systems, you need to create the Service user **NWDI\_CTSADMIN** and to assign the role **SAP\_XI\_CMS\_SERV\_USER** to this user. The user that you have created will be needed in the transport domain controller system in the TMS System Overview. There (on the *Transport Tool* tab for the SAP system in question), you need to choose *Goto → XI User/Password* from the menu and enter the credentials as you have maintained it in the SAP PI system.

### Example: Configuration of SAP HANA DB Systems as Runtime Systems

If you want to transport SAP HANA DB content with the help of the enhanced CTS, you need to configure both the CTS system and the SAP HANA DB systems accordingly. The main steps are listed on the following roadmap:



In case of transporting HANA content with the help of the enhanced CTS, you need to perform the following steps:

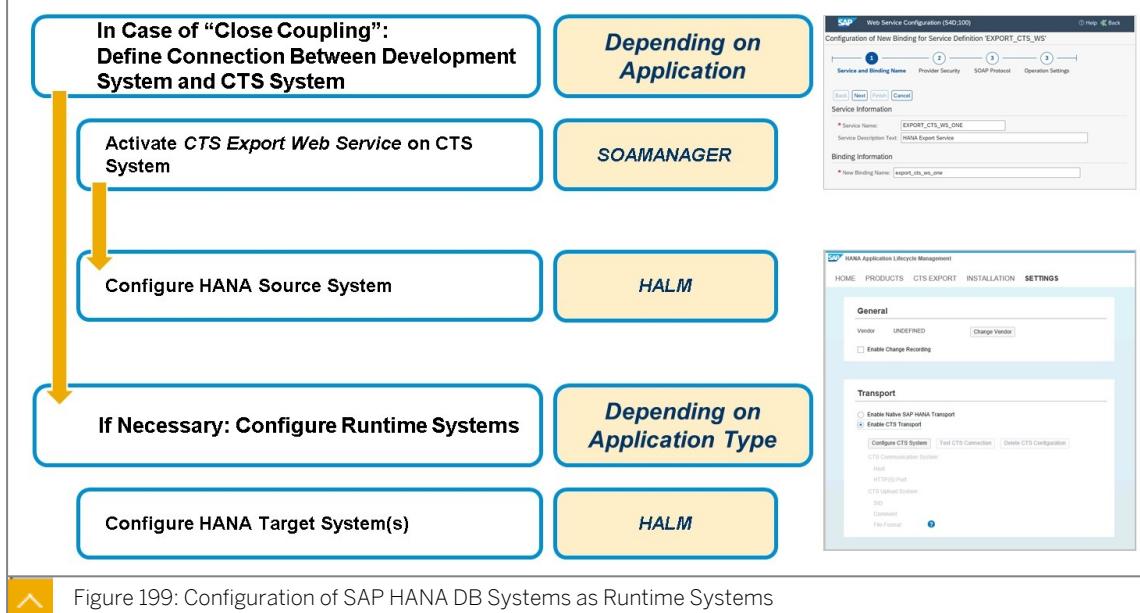


Figure 199: Configuration of SAP HANA DB Systems as Runtime Systems

### Activate CTS Export Web Service (CTS Export Client) on CTS System

On the SAP HANA DB development system, you have to configure the remote connection to the CTS system. This is done with the help of the *CTS Export Web Service (CTS Export Client)*. This Web Service runs on the AS ABAP of your CTS system and needs to be activated. The SID under which the system is known in CTS has to be forwarded to the CTS system whenever a transport request is needed or created. The CTS system has to know for which system it has to create or look for a transport request.



Note:

The “name” of a transport request in TMS starts with the SID of the development (source) system (as defined in the TMS) and thereby identifies (among others) the transport route to which it belongs.

The activation takes place in transaction SOAMANAGER (*SOA Management*) on the CTS system. Here you need to create/define the service **EXPORT\_CTS\_WS\_ONE** for the existing service definition *EXPORT\_CTS\_WS*. You can also define/edit the binding here (the *binding* contains a runtime configuration which is needed to implement the service). As soon as the binding is defined, you can configure the service as desired.

The most important configuration settings are defined on the *Provider Security* step of the *Guided Configuration* (for example, *Transport Level Security*, *Authentication Settings*). The *Authentication Settings* that you use in here has to fit with what you configure for the destination to the CTS system in *SAP HANA Application Lifecycle Management* (HALM) later (see below).



#### Hint:

The use of Single Sign-On using SAP Assertion Ticket is recommended (see SAP Note [2117110](#) – Recommendation to Replace SAP Logon Tickets with Single Sign-On Solution).

## Configure SAP HANA DB Source System

The configuration in the CTS system is ready now. Next you have to configure the connection from your SAP HANA DB development (source) system to the CTS system.

This configuration is done in *SAP HANA DB Application Lifecycle Management* (HALM; URL `http://<Host of SAP HANA DB>:<port>/sap/hana/xs/lm`) with a user that is assigned to (at least) the (HANA) roles:

- `sap.hana.xs.admin.roles::HTTPDestAdministrator`
- `sap.hana.xs.admin.roles::RuntimeConfAdministrator`
- `sap.hana.xs.lm.roles::Administrator`



#### Note:

If you use Multi-tenant Database Containers, make sure that you are logged on to the correct tenant. Settings for the enhanced CTS in HALM are tenant specific. This means that different tenants may also have different settings.

On HALM, enter the *SETTINGS* area, select *Enable CTS Transport* and click *Configure CTS System*.

- Here you first need to provide system details about the CTS system and the SID of your development system (as defined in the *System Overview* area of transaction STMS on the CTS system).
- Next, you need to maintain the (HTTP) destination to the CTS system. Here you need to enter the alias that you have configured during the activation of the *CTS Export Web Service* before (for example, `/<client>/export_cts_ws`) as the *Path Prefix*. In addition, the authentication settings need to be maintained according to your needs. The *Authentication Type* that you use has to fit with what you have configured for the *CTS Export Web Service* before.

**Hint:**

SAP Note [2097341](#) – *Issues related to HALM Integration with CTS* lists some basic requirements which should be satisfied to have the integration between HALM and the CTS system successful. The documents attached to this SAP Note may also be helpful for troubleshooting.

### Configure HANA Target System(s)

You also need to enable the enhanced CTS on all target systems. This configuration is also performed in *SAP HANA Application Lifecycle Management* (HALM).

For this, log on to the HALM of the target system in question with a user that is assigned to (at least) the (HANA) roles

- `sap.hana.xs.admin.roles::HTTPDestAdministrator`
- `sap.hana.xs.admin.roles::RuntimeConfAdministrator` and
- `sap.hana.xs.lm.roles::Administrator`.

**Note:**

If you use Multitenant Database Containers, make sure that you are logged on to the correct tenant. Settings for the enhanced CTS in HALM are tenant specific. This means that different tenants may also have different settings.

Enter the *SETTINGS* area, select *Enable CTS Transport* and click *Configure CTS System*. You do not need to enter any details for the CTS system or the connection.

### Addendum: Configure SAP Business Technology Platform as Runtime Environment

In the *Cloud Foundry* environment of *SAP Business Technology Platform* (SAP BTP), you can transport *Cloud Foundry* applications bundled in *Multi-Target Application* (MTA) archives to *Cloud Foundry* spaces. With the integration of SAP BTP into the *Change and Transport System* (CTS), you can manage the transport of MTA archives with the same tool as for SAP BW ABAP objects or SAP Enterprise Portal content. You can model a landscape for your *Cloud Foundry* spaces in CTS like for any other non-ABAP application supported by CTS. When doing that, you associate logical systems in TMS (3-digit-SIDs in TMS that represent the *Cloud Foundry* spaces) to the *Cloud Foundry* spaces.

In order to trigger the transport, you have to create a transport request for the first system in the system landscape (according to the transport routes in transaction STMS) and attach the MTA archives that you want to transport to it using the *Transport Organizer Web UI* tool in CTS. Afterward, you need to release the transport request. You can then start the import into the next logical system from the transport route, which will trigger the import into the corresponding *Cloud Foundry* space on SAP BTP.

**Note:**

All SAP BTP users that should be allowed to attach BTP content to transport requests have to have a corresponding user in the client of the CTS system that you are using for transports (that is, the client where you have activated the *Transport Organizer Web UI*). To transport non-ABAP objects, you can use the authorizations of the delivered role *SAP\_CTS\_PLUS*.

If you want to transport applications running on the *SAP BTP Cloud Foundry Environment*, the guide *How To... Configure SAP Cloud Platform Cloud Foundry for CTS* (available in SAP Support Portal (<https://support.sap.com>), area *Tools* → *Software Logistics Tools* → *Change and Transport: Change and Transport for SAP Business Technology Platform* → *(Link to) How to... Configure SAP BTP for CTS*) might be helpful.

### Related Information

Additional information for the transport of non-ABAP-objects with the help of the *Change and Transport Systems* can be found in the online documentation of SAP S/4HANA (Product Assistance) area *SAP S/4HANA* → *Enterprise Technology* → *ABAP Platform* → *Administrating the ABAP Platform* → *Administration Concepts and Tools* → *Solution Life Cycle Management* → *Software Logistics* → *Change and Transport System* → *Change and Transport System - Overview* → *Transporting Non-ABAP Objects in Change and Transport System*.

Information on transporting SAP HANA DB content in particular can be found in area *SAP S/4HANA* → *Enterprise Technology* → *ABAP Platform* → *Administrating the ABAP Platform* → *Administration Concepts and Tools* → *Solution Life Cycle Management* → *Software Logistics* → *Transport Scenarios for SAP HANA Content*.

In addition, there are some articles regarding the enhanced functionality of the CTS. These are available on SAP Community, area *Software Logistics* (<https://www.sap.com/community/topic/software-logistics.html>). From, here you also can follow the links in the *Expert Content* area, section *Change and Transport*, for example, *Enhanced Change and Transport System (CTS+)* (<https://support.sap.com/en/tools/software-logistics-tools/enhanced-change-and-transport-system.html>), *Transports in SAP HANA / HALM* (<https://support.sap.com/en/tools/software-logistics-tools/hana-halm.html>) or *Change and Transport for SAP Cloud (Business Technology) Platform* (<https://support.sap.com/en/tools/software-logistics-tools/sl-for-sap-btp.html>).

Finally, the following SAP Notes might be helpful:

- SAP Note [1003674](#) – *Enhancement for non-ABAP systems in CTS* is a composite note that provides additional information about the enhanced CTS.
- SAP Note [2286312](#) – *CTS+, Export Web Service and SLG1: Troubleshooting* provides information when the export calls against the ABAP communication system fails in case of using the *Export Web Service* (*EXPORT\_CTS\_WS*). SAP Notes [2097341](#) – *Issues related to HALM Integration with CTS* (including its attachments) and [2505430](#) – *CTS+ for HANA: Error when configuring the HTTP Destination* particularly provide information in case of problems when registering a CTS system in *HANA Application Lifecycle Manager (HALM)*.



### LESSON SUMMARY

You should now be able to:

- Explain the configuration steps for the enhanced CTS



## Transport of non-ABAP Objects

### LESSON OVERVIEW

After you have run through all necessary configuration steps on both the transport domain controller (TDC) system / the CTS system and the SAP systems that are included in the transport landscape (runtime SAP systems), you can transport non-ABAP objects into subsequent target systems.

This lesson starts with a section that contains the “big picture”. The main steps for transporting non-ABAP objects are listed here. In the following sections, these steps are presented in more detail with respect to the usage type of the SAP systems in question.

### Business Example

Your company uses the *Change and Transport System* (CTS) to perform transports in transport landscapes based on SAP systems with AS ABAP. In addition, your company is performing some development on an application for *SAP HANA* datamarts.

As a member of the transport administration team, you want to know how to transport content transport files (*TGZ* files) with the help of CTS as well.



### LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Perform transports of non-ABAP objects with the help of the ABAP Change and Transport System
- List use cases for the enhanced CTS
- Outline the transport options of SAP HANA content

### Transport of non-ABAP Objects

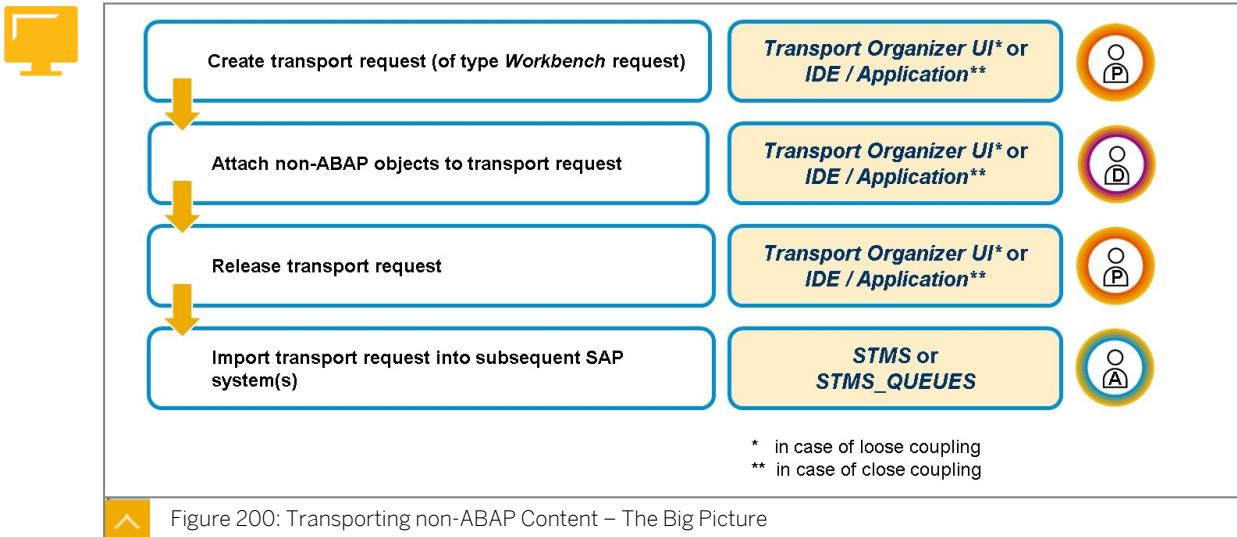
#### The “Big Picture”

After you have performed all necessary configuration steps in both the CTS system(s) and the SAP systems that are included in the transport landscape (source system, target systems), you can transport non-ABAP objects into subsequent target systems. The following roadmap contains the main steps to transport non-ABAP content with the help of the *enhanced Change and Transport System* (CTS).



#### Note:

In this lesson, the term *CTS system* refers to an SAP system based on AS ABAP which acts either as transport domain controller system or as communication system for the non-ABAP systems in question.



First, you need to create a suitable transport request (this transport requests needs to be of type *workbench request* because there is no client-specific data in non-ABAP system). This step is performed either:

- on the communication system with the help of the *Transport Organizer Web UI* (for non-ABAP development systems) in case of “loose coupling” or
- directly by the Integrated Development Environment (IDE; such as *SAP NetWeaver Developer Studio* or *SAP Web IDE for SAP HANA*) or the application (such as *Package Export Editor* in *SAP Enterprise Portal*) in case of “close coupling”.



#### Hint:

Depending on the *tp* parameter *WBO\_GET\_REQ\_STRATEGY*, you can configure “close coupling” or “loose coupling”, that is, whether the application triggers the *Change and Transport System* to create a transport request automatically.

Next, you can attach non-ABAP objects (for example SCAs, EARs, SDAs, EPAs, TPZs, TGZs, MTA archives) to this transport request. This can also be done either with the help of the *Transport Organizer Web UI* in the communication system (in case of a non-ABAP development system when using “loose coupling”) or directly in the IDE / application (in case of “close coupling”).

Then, you may release the transport request. Again, this step can be done either with the help of the *Transport Organizer Web UI* in the communication system or directly in the application / IDE.



#### Hint:

Depending on the *tp* parameter *WBO\_REL\_REQ\_STRATEGY*, you can configure the *Transport Management System* so that a transport request will be released automatically as well (“close coupling”).

After the export of the transport request has finished successfully, you may want to import the transport request to a subsequent SAP system (for example: the quality assurance *SAP Enterprise Portal* system). The import can be triggered with transaction *STMS\_QUEUES* on

either the target system (if it is based on AS ABAP) or on the CTS system (for non-ABAP target systems).



Hint:

Transaction STMS\_QUEUES offers a subset of the functions of transaction STMS.



Note:

For *Integration Builder Content* transports (TPZ files), depending on the release and SP stack level, it might be necessary to activate the changes manually in the *Integration Directory* of the target system.

### Generic Steps for Performing Transports

Before you can perform the integrated transport scenarios, a transport request (of type *Workbench Request*) is needed.

#### Creating Transport Requests and Attaching non-ABAP Objects

This transport request can be created manually in the *Transport Organizer Web UI* (based on the *CTS\_ORGANIZER* Web Dynpro application) that runs on the AS ABAP part of the communication system. Or this transport request can be created automatically by the application (by setting the *tp* parameter *WBO\_GET\_REQ\_STRATEGY* to **SMART** or **CREATE** for the source system).



Hint:

By choosing the value *CREATE* for the *tp* parameter *WBO\_GET\_REQ\_STRATEGY*, you get a new transport request, whether a *preselected request* already exists or not, and select it as the default request. This can be used in combination with *WBO\_REL\_REQ\_STRATEGY* = **AUTO**. Then developers don't need to know anything about transport requests: a transport request is created automatically for them when exporting and is then released.

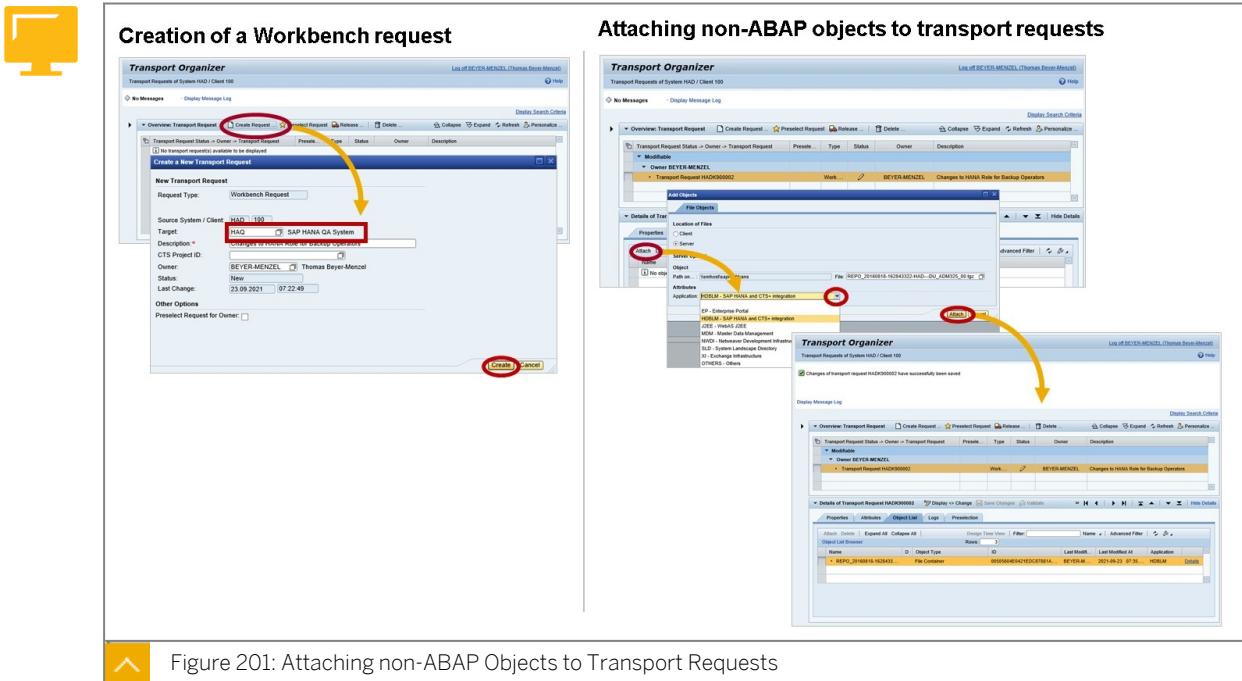


Figure 201: Attaching non-ABAP Objects to Transport Requests



Note:

The CTS\_ORGANIZER Web Dynpro application for the *Transport Organizer Web UI* can be accessed directly from the entry screen of transaction STMS using the *Transport Organizer Web UI* button.



Hint:

For non-ABAP transport, tasks of a transport request are not used (only the transport request itself is needed).

The transport request that you create with the help of this *Transport Organizer Web UI* on the communication system will be named <SID>K9<number> where <number> is a running number and <SID> is the name of the development system (source system) according to your transport landscape.

To attach non-ABAP transport objects (for example, an *EAR* file or a *TGZ* file) to existing workbench requests, you can use the coupling to the *Transport Organizer* using the *Transport Organizer Web UI*. Here, press the corresponding button on the start screen of transaction STMS. On the following log on screen, you may need to enter the client for which you want to perform the transport and your credentials on the communication system. Within that *Transport Organizer Web UI*, you can create a new transport request or select an existing transport request that you want to use for attaching files from your file system to it.



Note:

For certain transport scenarios it might be necessary to mark this transport request as *Preselected Transport Request*.

There are two options to attach non-ABAP transport objects (for example, an *EAR* file or a *TGZ* file) to existing workbench requests:

- You can attach the file directly and automatically from within the application or the IDE in the case of “close coupling”.
- Or you use the coupling to the *Transport Organizer* using the *Transport Organizer Web UI*. In this case, press the corresponding button on the start screen of transaction STMS. On the following log on screen, you may need to enter the client for which you want to perform the transport and your credentials on the communication system. Within that *Transport Organizer Web UI*, you can create a new transport request or select an existing transport request that you want to use for attaching files from your file system to it (“loose coupling” scenario).

**Note:**

For certain transport scenarios it might be necessary to mark this transport request as *Preselected Transport Request*.

There are several types of objects that can be attached to a transport request. You can browse your file system and choose the file that you want to attach to the transport request. When you have found the file in question, choose the *Attach* button on the *Object List* tab. You might need to switch into change mode for this.

### Releasing Transport Requests

After assigning non-ABAP objects to a transport request, you can release this transport request. The release of a transport request triggers the export of the files to file system level (transport directory).

After releasing the transport request, this transport request is ready for import into a subsequent SAP system (according to the definition of the transport routes in transaction STMS on the transport domain controller system).

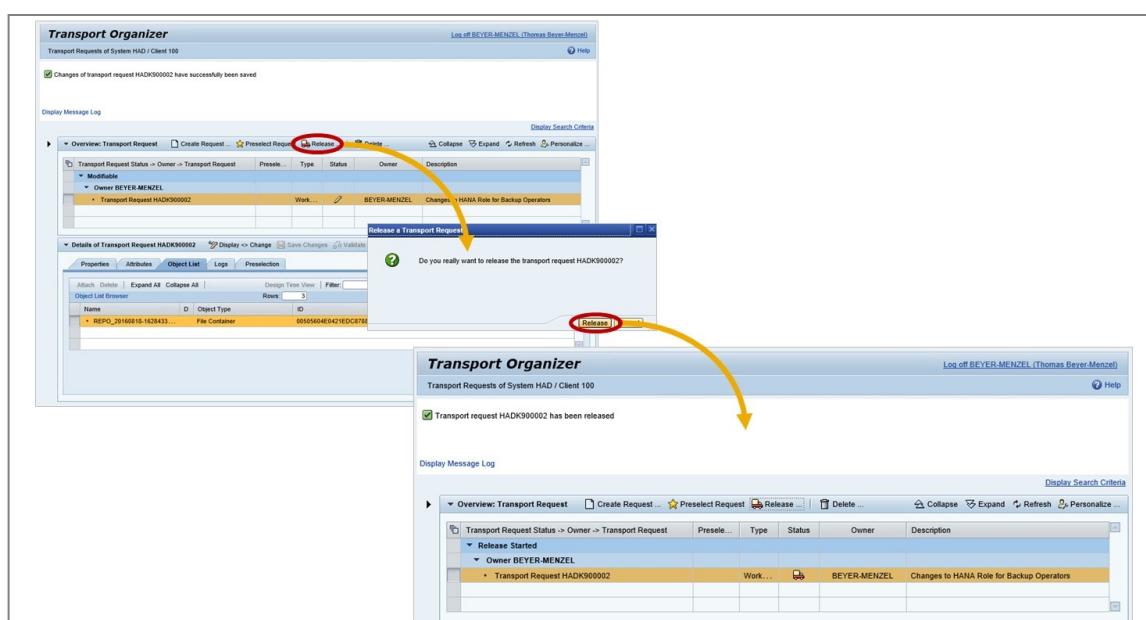


Figure 202: Releasing a Transport Request

The release of the transport request <SID>K9<number> can either be done manually using the *Transport Organizer Web UI* (as shown in the figure above), or it can be done automatically in the IDE / application (in this case, the *tp* parameter *WBO\_REL\_REQ\_STRATEGY* needs to be set to **AUTO** for the source system).

## Importing Transport Requests into a Subsequent SAP System

The import of transport requests into subsequent SAP systems can be performed in transaction **STMS\_QUEUES** (which is a subset of transaction **STMS**) on the CTS system.

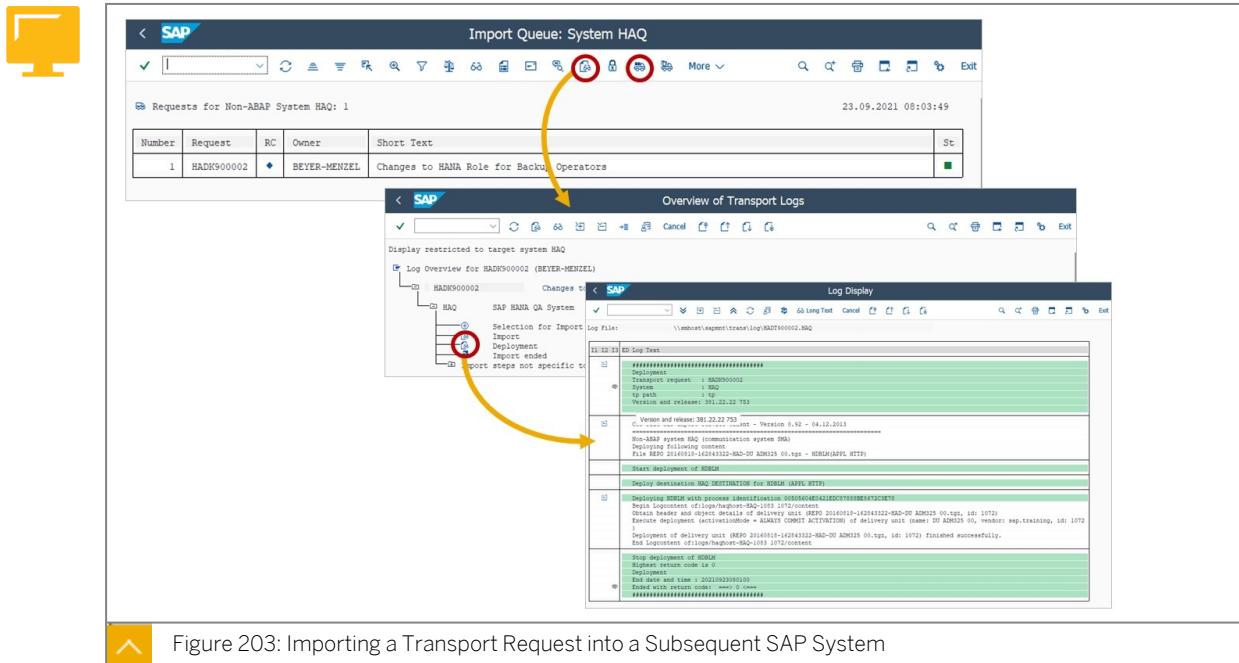


Figure 203: Importing a Transport Request into a Subsequent SAP System

In transaction **STMS\_QUEUES**, double-click the subsequent system in which you want to import the transport request (this is the SAP system which was assigned as *Target (System)* in the transport request before). Select the transport request in question and choose the *Import Request* button on the application toolbar. After the import has finished, you may want to check the *Deployment log* (see the figure “Importing a Transport Request into a Subsequent SAP System”).

During the import, the transport control program *tp* triggers the import dispatcher *RDDIMPDP* that in turn triggers the job *RDDEPLOY* on the CTS system. Depending on the deployment method, this job opens a connection to the *CTS Deploy Web Service* or the *Deploy Client*. This connection needs to be kept open during the complete deployment process. The *CTS Deploy Web Service / Deploy Client* finally calls the import tool on the non-ABAP system in question (for example, the *Deploy Controller* in the case of target systems based on AS Java 7.1 and higher). As a result, the contents of the transport request has been deployed to the non-ABAP system in question and the transport request is now listed in the import queue of the follow-up SAP system (according to the definition of the transport landscape in transaction **STMS**, area *Transport Routes* on the transport domain controller system).



**Hint:**

If your import queue isn't refreshed automatically, choose *Refresh*.

**Note:**

As of enhancement package 2 for SAP NetWeaver 7.0, SAP provides predefined roles for assigning SAP system-specific authorizations for non-ABAP SAP systems:

- You can use the predefined role *SAP\_CTS\_PLUS\_ORG\_TEMPLATE* as a template to create a new role restricting permissions to **create** transport requests to certain SAP system(s). It contains the authorization object *S\_SYS\_RWBO* (authorization object for granting system-specific authorization for the *Transport Organizer*).
- You can use the predefined role *SAP\_CTS\_PLUS\_TRANSPRT\_TEMPLATE* as a template to create a new role restricting permissions to **import** transport requests into certain system(s). It contains the authorization object *S\_CTS\_SADM* (authorization object for granting transport authorizations for individual systems in the *Change and Transport System*).

## Use Cases for the Enhanced Change and Transport System

### Appendix: Transport for Objects from SAP NetWeaver Development Infrastructure (NWDI)

*SAP NetWeaver Development Infrastructure (NWDI)* provides an infrastructure for developing Java based applications on the SAP NetWeaver platform and is responsible for versioning, build and lifecycle management of these applications. SAP NWDI takes care of all parts of the development process in a project-specific way:

- Central source file management:

The *Design Time Repository* (DTR) offers a file storage in a database with check-out / check-in mechanisms that allows versioning of source files.

- Central build and archive management:

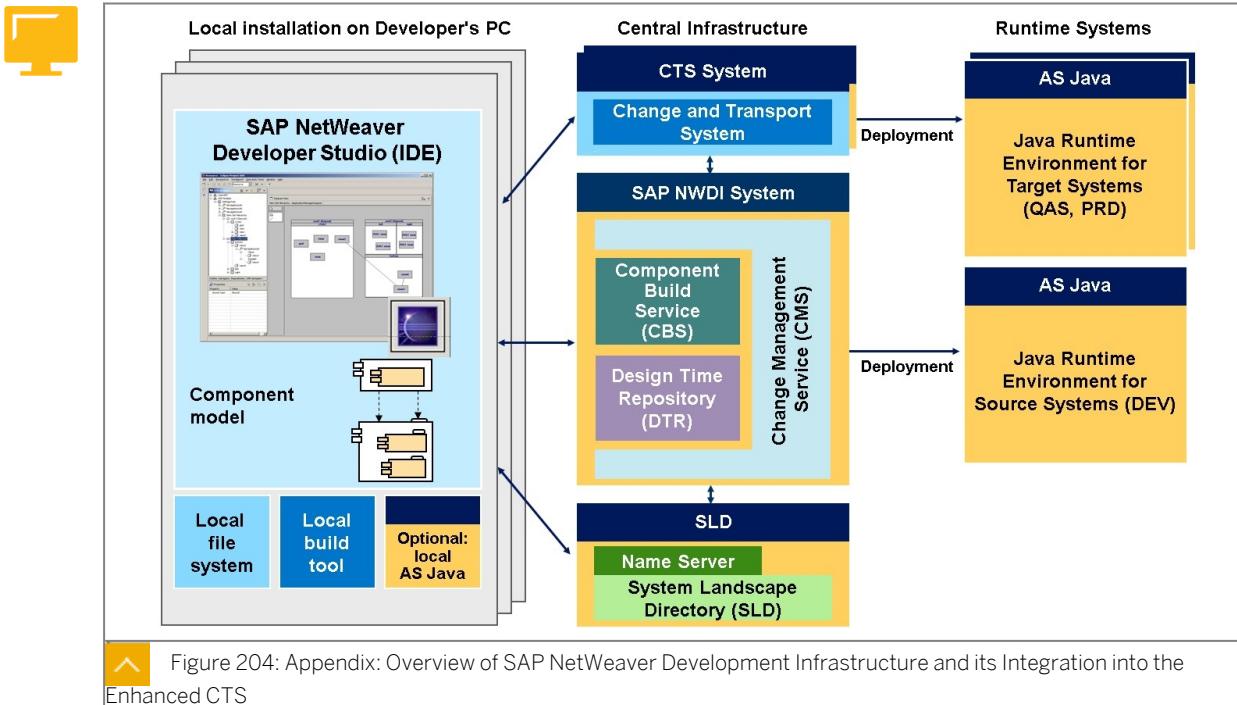
The *Component Build Service* (CBS) gives developers access to the latest archive versions in a central database storage and a central build triggered by the developer.

- Central landscape and transport management:

The *Change Management Service* (CMS) and the *enhanced Change and Transport System* are used for the central administration of the Java transport landscape. The functions of the CMS are closely interlinked with the DTR, the CBS and the *System Landscape Directory* (SLD) which contains a description of your SAP system landscape.

All the development processes in SAP NWDI are based on the SAP component model, which enhances the public/private concept of Java by metadata regarding the use of objects without implying any changes to the development objects themselves: a Java interface stays an interface, a public class stays a public class. By clearly defining the visibility of and the dependencies between objects, the component model helps you structure applications into reusable components.

The figure below shows the integration of *SAP NetWeaver Development Infrastructure* with the *enhanced Change and Transport System*.



An important feature of SAP NWDI is the ability to develop and build software that belongs to different releases and SAP Support Package Stacks. To do this, you need only a single SAP NWDI system, because the *Component Build Service* (CBS) creates the software in accordance with the release and SAP Support Package Stack.

*SAP NetWeaver Developer Studio* is the SAP own environment for developing multi-level Java-based applications. The development environment is based on the open source product *Eclipse*, whose open plug-in architecture provides a suitable platform that can be enhanced with special functions.

On the side of the central infrastructure, there are, among other things, the services shown in the figure above: the *Design Time Repository*, *Component Build Service*, and the *Name Server* as part of the *System Landscape Directory*.

Developers use *SAP NetWeaver Developer Studio* to initiate the transfer of local development work to the central development system. *SAP NetWeaver Developer Studio* uses CBS and CMS to start the deployment into the central development system. The import into all the following SAP systems can be controlled exclusively via the *Change and Transport System* (CTS) and it is the administrator who starts the imports via transaction STMS.

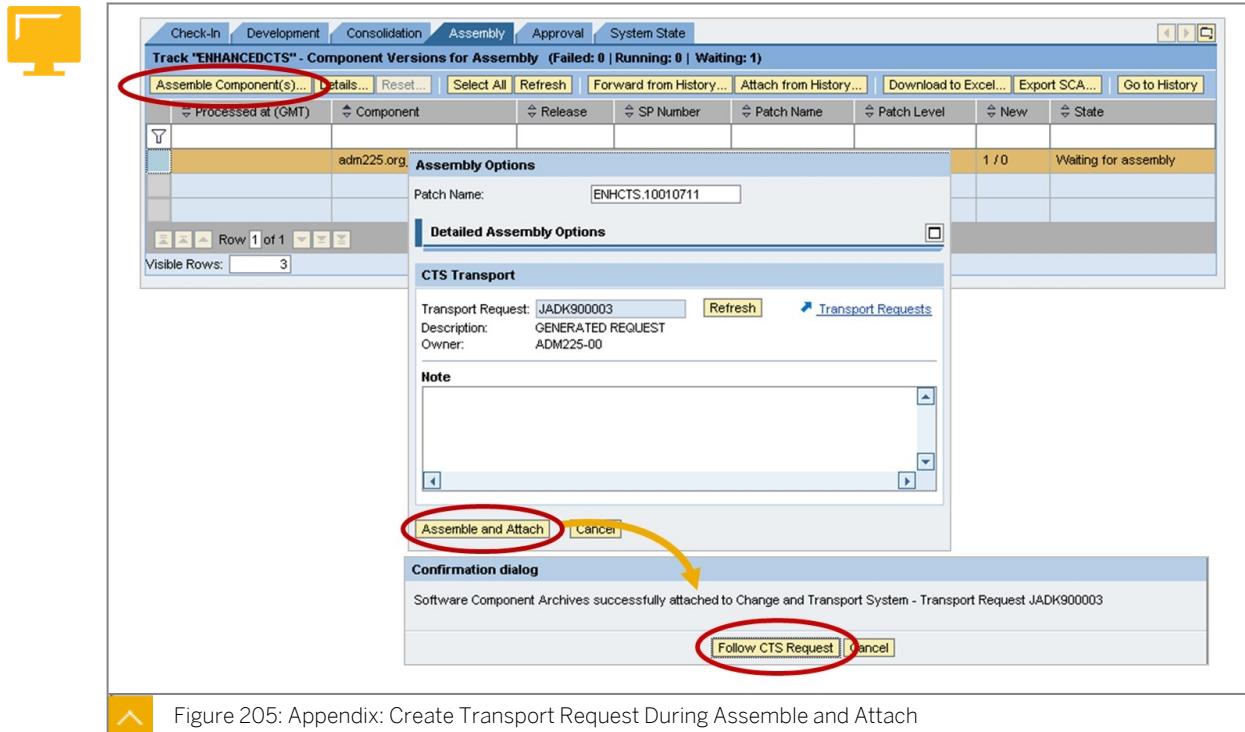


Figure 205: Appendix: Create Transport Request During Assemble and Attach

Before you can transport objects from SAP NWDI, you need a transport request of type *Workbench request*. This transport request can be created in different ways:

- You can create a transport request in the *Transport Organizer Web UI* on the CTS system (“loose coupling”).
- A transport request can be generated and released automatically from SAP NetWeaver Developer Studio or from CMS. Then this transport request is handed over to CTS (“close coupling”).

As a prerequisite, the following *tp* parameters must be set for the source system on the transport domain controller system (in transaction STMS, System Overview area, *Transport Tool* tab) – these two parameters allow the automatic creation and automatic release of a transport request:

- *WBO\_GET\_REQ\_STRATEGY* = **SMART** or **CREATE**
- *WBO\_REL\_REQ\_STRATEGY* = **AUTO**

After releasing the transport request, this transport request appears in the import queue of the subsequent SAP system (according to the transport routes defined in the *Transport Routes Editor* of transaction STMS). It can be imported to the subsequent SAP system with transaction *STMS\_QUEUES* as usual for ABAP transport requests.

**Hint:**

You need to configure the SAP NWDI system for the integration with the CTS system first. To do this, you need to maintain both connection data to the communication system and the SID of the corresponding Java development system. These settings can be done within *SAP NetWeaver Administrator* (area *Configuration → Security → Destinations*). Create a connection of *Destination Type RFC* with *Destination Name (exactly)* **sap.com/com.sap.tc.di.CTSserver** by choosing *Create....*

- In the *Connection and Transport Security Settings* step, enter the connection data of your CTS system (to be more precise: the communication system of the Java development system in the TMS system landscape).
- In the *Logon Data* step, select the *Current User (Assertion Ticket)* option as authentication for the connection to your CTS system. SAP recommends that you use Single Sign-On (SSO). If you cannot use SSO, select the *Technical User* option and enter your preferred *language*, a *client*, and an appropriate *user and password*. Enter the *client* from the *tp* parameter **NON\_ABAP\_WBO\_CLIENT** on the CTS system.
- Use the default settings under *Repository Connection* (*Logon Data* step) and *Pool Settings* (*Specific Settings* step) and save your entries.

This hint also applies for the configuration of *SAP Enterprise Portal* systems for the use with the enhanced capabilities of CTS.

## Appendix: Transport for SAP Enterprise Portal Systems

Between the communication system and the *SAP Enterprise Portal* system, a “close coupling” is available.

With “close coupling”, it is possible to create and release transport requests automatically and directly from the *Package Export Editor* in the development *SAP Enterprise Portal* system (to enter this tool, switch to area *System Administration → Transport → Transport Packages → Export*). In this scenario, you need to create an object (*transport package*) that can be transported within the *SAP Enterprise Portal* system. Then you include your changes to the *SAP Enterprise Portal* objects in question in this transport package.

For “close coupling”, the following *tp* parameters can be set for the source system on the transport domain controller system (in transaction *STMS*) to allow the automatic creation and automatic release of a transport request:

- *WBO\_GET\_REQ\_STRATEGY* = **SMART** or **CREATE**
- *WBO\_REL\_REQ\_STRATEGY* = **AUTO**

In contrast, “loose coupling” means that transport requests need to be created manually. In addition, in this case, non-ABAP objects need to be exported to file system level first before they can be attached to the transport request.

In the “close coupling” scenario, you need to select the *Transport Method CTS*. If you press *Start Export*, the objects that are included in the transport package will be attached to a transport request. This transport request either must exist already as *Preselected Transport Request* (in case that *tp* parameter *WBO\_GET\_REQ\_STRATEGY* is set to **TAGGED**) or is created

automatically (in case that *tp* parameter *WBO\_GET\_REQ\_STRATEGY* is set to **SMART** or **CREATE**).

The release of this transport request can then be done either manually using the *Transport Organizer Web UI* (in case that *tp* parameter *WBO\_REL\_REQ\_STRATEGY* is set to **MANUAL**) or automatically after attaching the objects to the transport request (in case that *tp* parameter *WBO\_REL\_REQ\_STRATEGY* is set to **AUTO**).

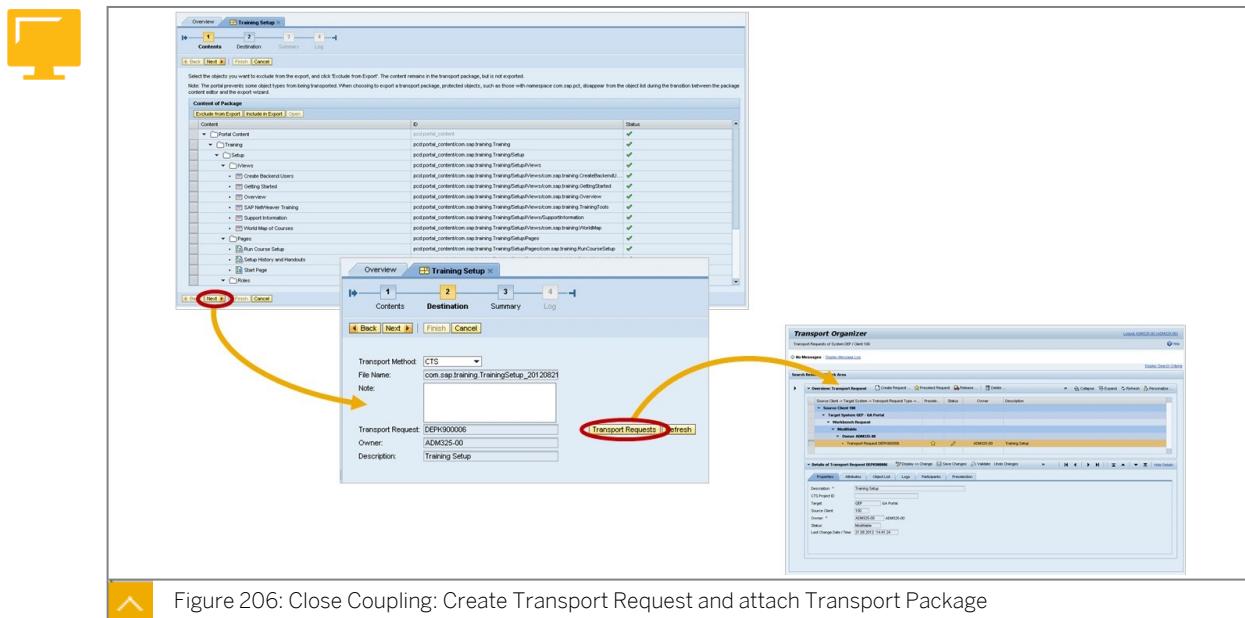


Figure 206: Close Coupling: Create Transport Request and attach Transport Package

To import the transport request into a subsequent SAP system, you can use transaction *STMS\_QUEUES* on the CTS system.

### Further Information on the Enhanced CTS

Further Information on the enhancements of the ABAP Change and Transport System can be found on SAP Community (area Software Logistics: <https://www.sap.com/community/topic/software-logistics.html>). From here you may follow the links in area *Expert Content* (for example, *Enhanced Change and Transport System (CTS+)* <https://support.sap.com/en/tools/software-logistics-tools/enhanced-change-and-transport-system.html> or *Transports in SAP HANA/HALM* <https://support.sap.com/en/tools/software-logistics-tools/hana-halm.html>).

For information how to use CTS for transporting applications running on the SAP Business Technology Platform (SAP BTP) Cloud Foundry Environment, see the blog *Setting up a CTS+ enabled transport landscape in SAP Cloud Platform* (<https://blogs.sap.com/2017/03/29/setting-up-a-cts-enabled-transport-landscape-in-sap-cloud-platform/>) and the guide *How To... Configure SAP Cloud Platform Cloud Foundry for CTS* (available in SAP Support Portal (<https://support.sap.com>), area Tools → Software Logistics Tools → Change and Transport: Change and Transport for SAP Business Technology Platform → (Link to) How to... Configure SAP BTP for CTS).

### Transport Options for SAP HANA Content

There are a number of different transport scenarios for SAP HANA Content , depending on how the SAP HANA Content is used.

- For SAP HANA Content for native SAP HANA applications, you can use the extended Change and Transport System or SAP HANA Application Lifecycle Management (HALM).
- For SAP HANA Content for ABAP for SAP HANA applications, you can use SAP HANA Transport for ABAP (HTA).

**Note:**

SAP HANA Content refers to SAP HANA development objects that are created in SAP HANA-based development projects. SAP HANA Content is not part of the SAP HANA database installation itself. It is stored in the repository of the SAP HANA appliance software and can include different objects, such as schemas and table definitions, information views (attribute views and analysis views), SQL scripts, or roles and authorizations.

A typical system landscape for SAP HANA development consists of a development system, a quality assurance system, and a production system. To transport SAP HANA development objects from a system to a successor system, there are a number of transport scenarios available. The transport scenario that is suitable for your situation depends on the way that the SAP HANA Content to be transported is used, and on your system landscape (see the figure “Transport Options for SAP HANA Content”).

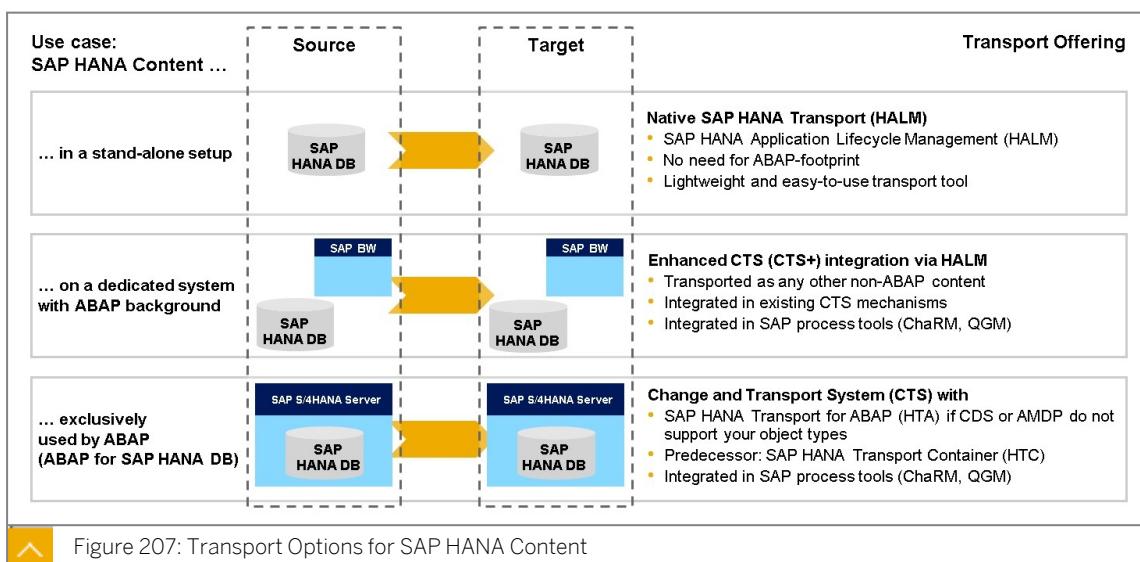


Figure 207: Transport Options for SAP HANA Content

## 1. SAP HANA content for ABAP for SAP HANA applications

SAP HANA content for ABAP for SAP HANA applications is usually transported together with ABAP objects, since the development objects are closely connected in terms of contents.

SAP HANA Transport for ABAP (HTA) can be used to synchronize objects and packages from the SAP HANA repository to ABAP and add them to a default (ABAP) transport request. Before synchronizing them, you select SAP HANA packages and objects that you can transport together with the associated ABAP objects. When the transport request is imported into the target system:

- The ABAP objects in the transport request (including the HTA objects) are imported into the ABAP based SAP system.

- The SAP HANA packages and objects are imported into the SAP HANA repository of the target system and are activated there.



**Hint:**

Until SAP NetWeaver 7.4 SPS 10, *HANA Transport Container* (HTC) was the method used to transport SAP HANA content for ABAP for SAP HANA applications. In contrast to HTA, an HTC always contains **all** SAP HANA repository objects of a delivery unit. From SAP NetWeaver 7.4 SPS 11, HTA is used to transport SAP HANA content for ABAP for SAP HANA applications.

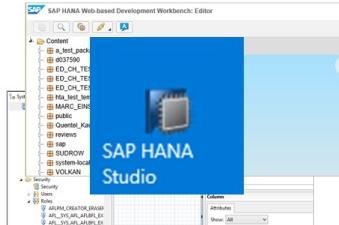


**Note:**

As of Feature Package Stack 02 for SAP S/4HANA 1809, transaction `SCTS_HTA_TOOLS` offers you an overview of tools and transactions that supports you when working with SAP HANA Transport for ABAP.

You can also transport native SAP HANA objects created using HDI (SAP HANA Deployment Infrastructure) with HTA. For more information, see SAP Note [2569651](#) – Configure your ABAP Development System for Development of HDI Objects and SAP Note [2493223](#) – HTA for HDI: Release Note.

**1 Develop your application**



- SAP HANA Studio or
- SAP Web IDE for SAP HANA

**2 Select objects / packages to be transported**



- Transaction `SCTS_HTA`

**3 Manage your transports**



- Transaction `SE09` on source system
- Transaction `STMS` on target system

 Figure 208: SAP HANA Transport for ABAP

## 2. SAP HANA content for native SAP HANA applications or non-ABAP applications that are based on SAP HANA, if the enhanced CTS is in use

Content for native SAP HANA applications is SAP HANA content that is developed exclusively for SAP HANA systems. You can use the *enhanced Change and Transport System* (CTS+) to transport SAP HANA content for native SAP HANA applications, or for non-ABAP applications that are based on SAP HANA. The enhanced CTS is particularly well-suited to heterogeneous transport landscapes in which the enhanced CTS is already in use. From an administrative point of view, the SAP HANA content is another object type that can be transported with the CTS in the same way as other non-ABAP objects. The effort required for configuration is low, and the transport tools are already familiar. In particular, you can use the enhanced CTS if you want to transport SAP HANA content as part of an SAP solution (such as SAP BI, Mobile, and so on), since in this case, you often need to create a link between the SAP HANA content and other non-ABAP development objects and non-SAP HANA development objects.

This transport scenario is integrated into the existing CTS transport landscape and into the SAP tools for change control (*Change Request Management* and *Quality Gate Management* in SAP Solution Manager). You can configure the transport routes for the systems in question in the *Transport Management System* (TMS). The transport is performed in the same way as for any other non-ABAP application that is integrated into the enhanced CTS.

### 3. SAP HANA content for native SAP HANA applications, if the enhanced CTS is not in use.

If the enhanced CTS is not in use, you can use *SAP HANA Application Lifecycle Management* (HALM) to transport SAP HANA content for native SAP HANA applications. You can use HALM to perform transports and monitor transport processes.

HALM is particularly well-suited to SAP HANA transport landscapes with no ABAP development (without enhanced CTS) and with no requirement for synchronizing transports with other non-HANA content.

#### **Additional Information**

For more information, which option to choose, see the blog *CTS+ or HTA?*, available at <https://blogs.sap.com/2015/06/11/cts-or-hta/>.

In addition, the following SAP Notes are related to SAP HANA Transport for ABAP:

- SAP Note [1990798](#) – SAP HANA Transport for ABAP (HTA): Release Information
- SAP Note [2290536](#) – Deployment with SCTS\_HTA
- SAP Note [2321486](#) – Troubleshooting SAP HANA Transport for ABAP (HTA) Deployment
- SAP Note [2493223](#) – HTA for HDI: Release Note
- SAP Note [2569651](#) – Configure your ABAP Development System for Development of HDI Objects
- SAP Note [2682272](#) – Troubleshooting and FAQs when using HTA for HDI



#### **LESSON SUMMARY**

You should now be able to:

- Perform transports of non-ABAP objects with the help of the ABAP Change and Transport System
- List use cases for the enhanced CTS
- Outline the transport options of SAP HANA content

## Outlook: Integration of ABAP Development Into CI / CD (DevOps) Processes with the Help of gCTS



### LESSON OBJECTIVES

After completing this lesson, you will be able to:

- Outline the basic idea of DevOps and gCTS

### Introduction to DevOps

#### The Change and Transport System

In AS ABAP based SAP systems, the *Change and Transport System* (CTS) is the central tool for managing changes to customizing and repository data that you make in customizing (*SAP Reference IMG*, transaction *SPRO*) or with the help of the ABAP Workbench (for example, transaction *SE80* or *ABAP Development Tools*). The CTS records all changes in transport requests.

By defining transport routes, you can assure that changes are transported into a test environment before they can be imported (copied) to the production environment. The *QA approval procedure* can be used to assure that the changes have been tested successfully before they can be imported into the production environment. In addition, all transports are logged, so that you can see when a transport request was imported into a client or system, and whether there were any errors.

With the help of CTS projects, the *Change and Transport System* also supports project-based development using the waterfall approach.



#### Note:

In a waterfall model, project requirements are established up front, followed by design, then implementation, verification and maintenance. It culminates in a “big bang” release. Ongoing business needs are assumed to remain similar throughout the process. Teams operate in silos, led by comprehensive documentation.

Automation is often not involved, and quality testing is delayed until the end.

In the SAP world, given the high level of integration and large number of dependencies inherent in the software, waterfall development has become the accepted practice.

With the advent of non-ABAP technologies in SAP environments (such as the development for AS Java based systems), the *Change and Transport System* was enhanced so that non-ABAP changes can be transported with the help of the CTS as well (*enhanced CTS* or for short: “CTS +”). Together with the capabilities of SAP Solution Manager, this offers:

- Defined transport routes for non-ABAP changes (such as Java development or native SAP HANA content)
- One central tool to control different types of transports within an SAP system landscape
- The option to include a predefined set of workflows and processes which are all compliant with the recommendations provided by the IT infrastructure library (*Change Request Management*).

## Agile Development and DevOps

Over the years, companies face increasing pressure to adapt much more quickly thanks to new architectures and technologies, including the cloud. Adages such as “never touch a running system” are outdated. Companies continuously want to create innovative applications or services to overcome their business constraints. However, they need the capability to move fast to respond to shifting market conditions in a strategic way. It is no longer about making technology choices you’ll stick with for decades. Rather, it’s about having the ability to make technology changes quickly and gracefully.

In this context, there has been a change of paradigm in software development. A number of agile development methods (such as *Scrum*) evolved in reaction to the waterfall methods that critics described as slow, error-prone and inflexible.

According to the *Manifesto for Agile Software Development* (<http://agilemanifesto.org/principles.html>), agile software development practices are based on principles such as:

- Customer satisfaction by early and continuous delivery of valuable software
- Welcome changing requirements, even late in development
- Deliver working software frequently (weeks rather than months)
- Close cooperation between business people and developers
- Best architectures, requirements, and designs emerge from cross-functional, self-organizing teams

One of the differences between agile software development methods and waterfall is the approach to quality and testing: Unlike waterfall, agile development and testing occur concurrently. Teams communicate closely with business owners and customers, involving stakeholders at every step. Documentation is light. Feedback is solicited frequently, and quality issues are addressed after every sprint.



### Hint:

A *sprint* is a repeatable fixed time-box (typically one month or less) during which a “done” product of the highest possible value is created. In this context, “done” refers to a set of items that have to be fulfilled before a project can be considered complete.

Agile's iterative delivery approach enables prioritization, proving value weekly, or even daily, while significantly reducing project risk. As opposed to the typical six-month waterfall delivery timetable that waits until the end for course corrections, agile responds to specific, evolving needs with great agility in an iterative, ongoing fashion.

The term **DevOps** was originally introduced to describe a strategy for improving development and operations collaboration. *DevOps* is neither a methodology nor a framework. It is a culture, a movement, and a philosophy that describes how to cooperate and shorten time to

market while staying up to date with market trends. The five principles of culture, automation, lean, measurement, and sharing (abbreviated to as CALMS) summarize this approach. It is about transforming the organization by bringing cross-functional developers and operations together with an emphasis on better collaboration and tighter integration.

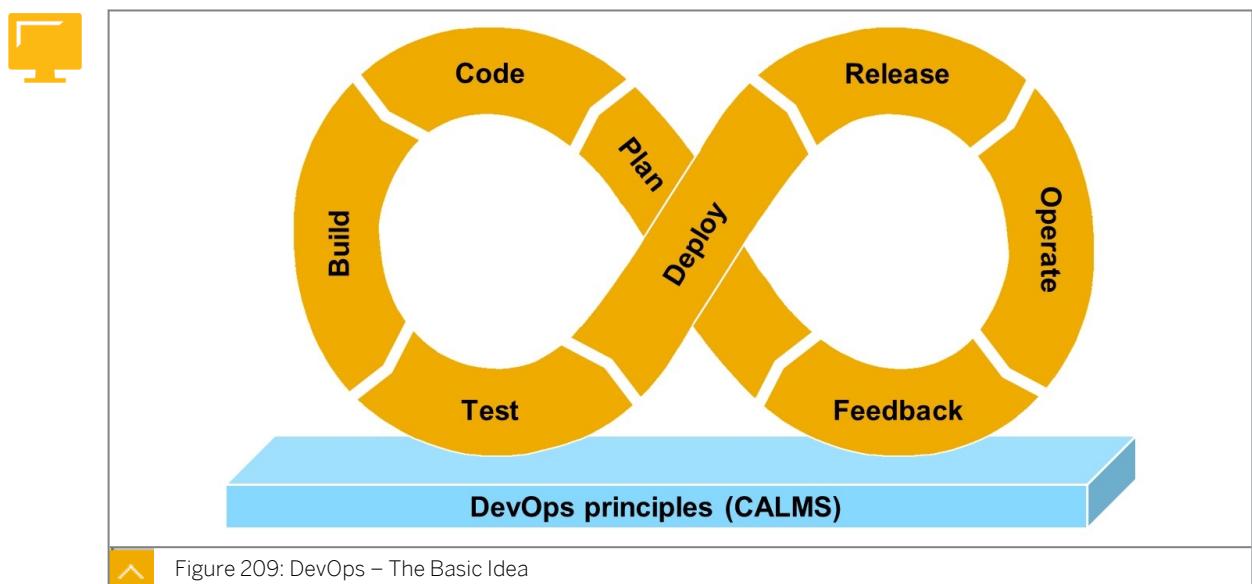


Figure 209: DevOps – The Basic Idea

### Continuous Integration

The goal of *DevOps* is to support continuous testing, integration, delivery and deployment of stable, high quality software, guided by continuous feedback, and enhanced by automation.

Agile development is just a first step on the journey to *DevOps* and its continuous approach to integrate and deliver, which requires a much higher level of automation and collaboration between the teams. In practice, the *DevOps* lifecycle is a continuous cycle due to the feedback and collaboration by operations and development, which serve as an input channel to planning. That's why more frequently the *DevOps* cycle is represented with the infinity symbol to represent the *DevOps* approach for a continuous, ongoing development and operations cycle (see the figure above).

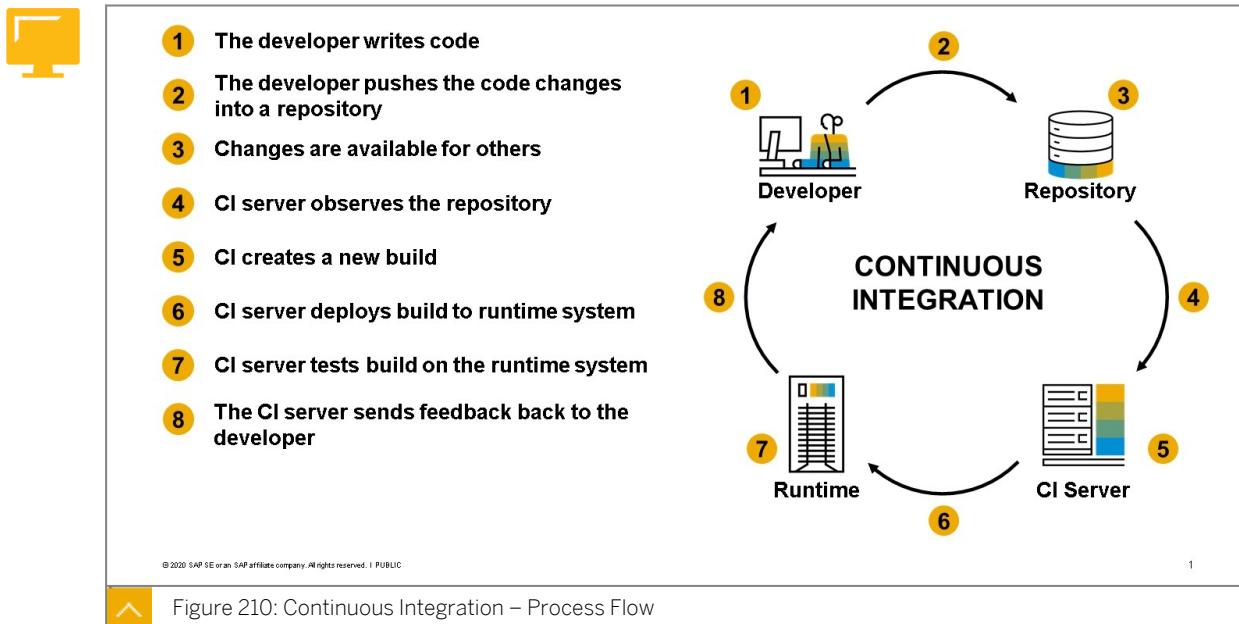
Agile development methods such as *Scrum* aim at enabling teams to deliver feature increments in short cycles. One of the key elements to doing this efficiently and in high quality is the practice of test automation, which ensures that all functional and regression testing is automated. These tests are written by developers, ideally at the same time as (or even before) the product code ("test-driven development"). The key to good test automation is learning to write testable code. There are many benefits in following this development approach, such as better quality code in less time, reduced bug analysis and fix efforts, and being more responsive to changing requirements.



#### Note:

For more information on writing testable code, see the openSAP course *Writing Testable Code for ABAP* (<https://open.sap.com/courses/wtc1>).

The following figure outlines the continuous integration process:

**Note:**

The following text assumes that *Git* is used as source code management system. For more information on *Git*, see a following section later in this lesson.

The developer writes code (they can use the known editors). Then the developer commits and pushes the code changes into a repository (for example by transferring objects in (open) tasks to *Git*). Technically, this results in a *commit*. As a result, the changes are stored as files in the *Git* repository and are visible to others. The *CI* server (if installed and configured) monitors the repository and creates a new build. Next, the *CI* server deploys the build to the runtime system. Then, the *CI* server deploys the build to an acceptance system, where the *CI* server verifies the build quality and sends feedback to the developer (if the pipeline on the *CI* server is set up in that way – see the note below). Finally, if tests have passed, the *CI* server propagates the validated changes towards the subsequent runtime system.

**Hint:**

You can connect multiple runtime systems with one and the same *Git* repository. When you start the deployment to a runtime system, you can select one commit (out of the list of available commits) which results in the deployment of all objects in the state that they have at the time of this commit.

**Note:**

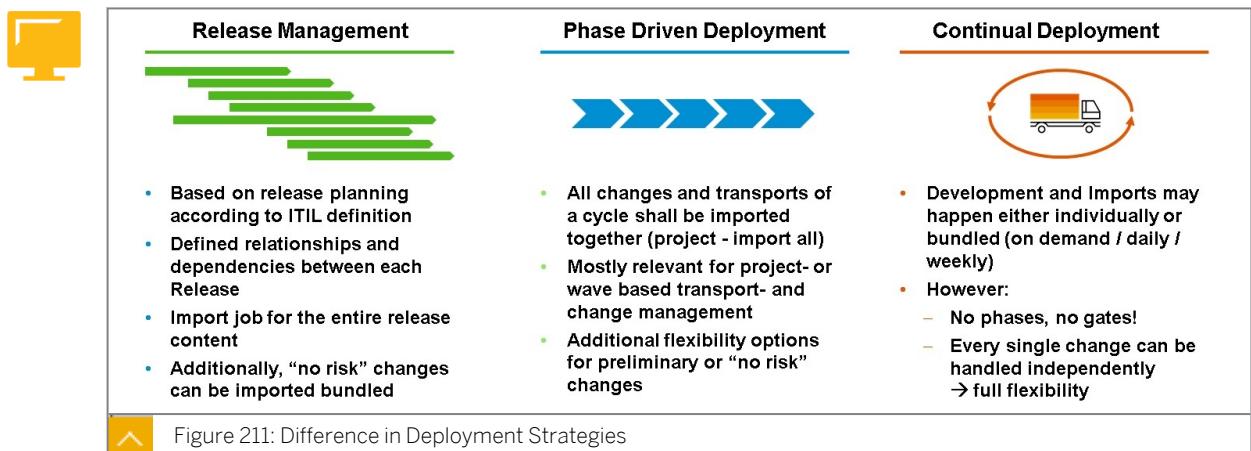
On a CI server, *pipelines* can be used to manage this process or to automate it. For example, on a CI server, a pipeline is set up to take care of the following steps:

- It monitors the repository, and whenever a new commit is available, the changed objects are extracted from the repository and deployed to an integration test system.
- After a successful deployment, some automated tests may run.
- Feedback on the results of these tests is sent to the developers.

The pipeline can be set up in a way that, if the test fails, the previous commit that had passed the tests successfully ("last known good commit") can be deployed again to the integration test system. The integration test system is then always in a usable state and ready for further commits to be deployed. The developers can then do fixes and commit them. Then, the pipeline starts again. This aggregates all of the changes made since the last "good" commit – the newest version of the changed objects – to the integration test system. With this, the integration test system always shows the last known good software state.

## Differences in Deployment Strategies

If we have a closer look at the deployment strategy, we see a major change coming with DevOps. In the world of the ABAP programming language, it was a best practice to recommend release bundling and a predefined deployment maybe once or a couple of times per year, or at least a project-oriented bundling of changes. DevOps comes with a continuous integration and delivery approach. For the different deployment strategies, see the figure "Difference in Deployment Strategies".



DevOps pushes for a higher sequence of deployments to adopt applications quickly. Ideally, the deployment of innovations is continuous. It is a logical conclusion that you must align your innovation and change planning accordingly. This means that you plan for smaller, ideally separated but frequent changes instead of only a few large change packages or releases for a time period.

According to DevOps, instead of planning for a "big bang" innovation step, you should split an innovation into smaller, encapsulated innovation steps according to change packages. Those packages will be continuously implemented and delivered as part of a continuous business planning effort.

**Hint:**

As outlined later in this lesson in more detail, the idea of *Git-enabled CTS* is to extend the idea of continuous development to the ABAP world as well. You then have the option to choose between the different deployment strategies for ABAP based development. All of these options have their pros and cons. Release management and the phase driven deployment still remain valid options, in particular, when the focus is on transporting changes into production systems.

## Git

Although the approach of *DevOps* is more common in cloud-based and Java-based solutions, most of the SAP customers have other technologies in place such as AS ABAP-based and on-premise solutions. As a result, there are the following challenges:

- Establish *DevOps* principles for all technologies as much as possible.
- Integrate different technologies to cover the whole solution landscape (hybrid).

In the formerly ABAP-centric world, SAP standards and tools were based on SAP software, such as the Transport Management System (TMS). You then needed to integrate third-party tools and systems. With the new variety in solutions and technology with *DevOps*, you now have a higher variety of tools. You can build an SAP software-centric, managed *DevOps* tool landscape and integrate ABAP and non-ABAP landscapes.

In the process of continuous integration (as outlined in a previous figure), a source code management system such as *Git* is needed.

**Hint:**

**Git** is a distributed source code management system for tracking changes in source code during software development. It was created by Linus Torvalds in 2005 for the development of the Linux kernel and is designed for coordinating work among programmers, but it can be used to track changes in any set of files. Its goals include speed, data integrity, and support for distributed, non-linear workflows.

*Git* supports branching and merging, and includes specific tools for visualizing and navigating a non-linear development history. In *Git*, a core assumption is that a change will be merged more often than it is written, as it is passed around to various reviewers. In *Git*, branches are very lightweight: a branch is only a reference to one commit. The *Git* history is stored in such a way that the ID of a particular version (a *commit* in *Git* terms) depends upon the complete development history leading up to that commit. Once it is published, it is not possible to change the old versions without it being noticed.

## The Idea of gCTS

The idea of the *Git-enabled Change and Transport System* (gCTS) is to enable you to manage your ABAP change and transport management processes using *Git* as an external version control system (VCS). See the figure “gCTS – The Basic Idea”.

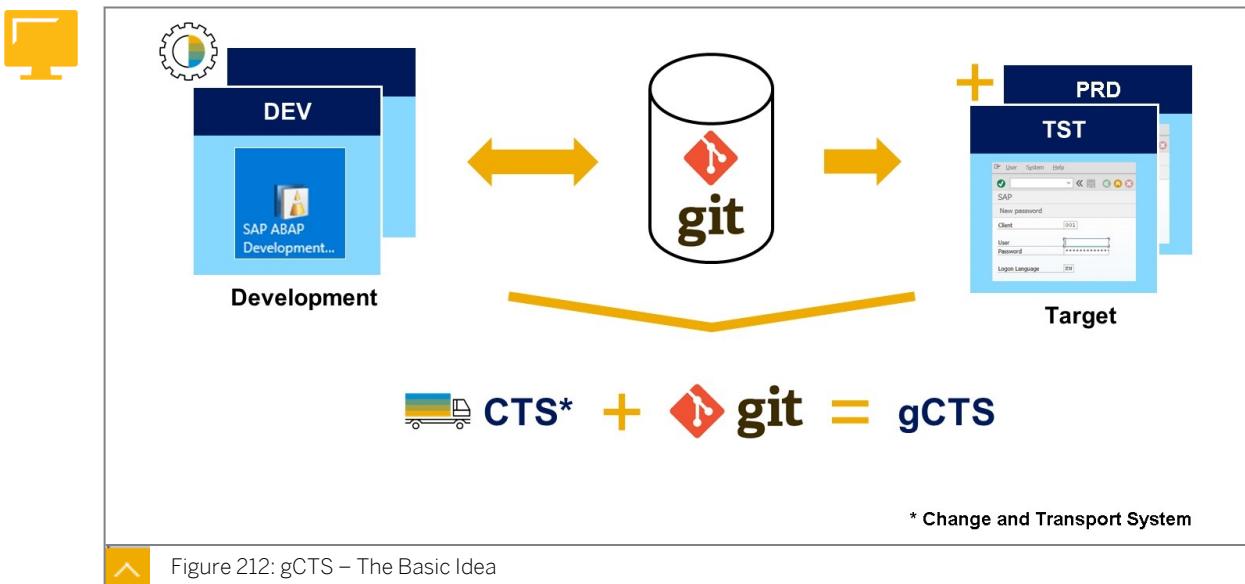


Figure 212: gCTS – The Basic Idea

You can use gCTS for the following situations:

- If you are accustomed to *Git* as a version control system, and now you also want to use it for your ABAP developments and continuous integration processes.
- If you are accustomed to ABAP development, and you want to align the processes with your non-ABAP development processes.

**Note:**

The use of gCTS is optional. For the latest information about *Git-enabled Change and Transport System*, see SAP Note [2821718](#) – Central Note for *Git-enabled Change and Transport System* (gCTS).

*Git-enabled Change and Transport System* (gCTS) lets you store all versions of ABAP development and customizing objects in *Git* repositories.

gCTS makes use of *Git* features, such as working on local copies of a central remote *Git* repository. Software development takes place in the usual editors, in *ABAP Development Tools* (ADT), for example. Changes to objects are still recorded in transport requests. Current states of objects can be transferred to remote *Git* repositories. This process is called “pushing objects to the remote repository”, for example, by releasing a transport request or to perform an initial filling of the *Git* repository.

ABAP objects are copied into a local repository on the ABAP system. In this process, they are transformed from database table contents to file and folder structures on the operating system level. These are stored in the local file system of the application server, the *local repository*, and are accessible from any command-line versioning tool, such as *Git*. This way, the ABAP file objects can be pushed to a central remote *Git* repository. This allows you to distribute software between systems using *Git*. You can retrieve the software from the central remote *Git* repository to another ABAP system. This process is called “pulling” objects from the remote repository.

**Note:**

As a prerequisite, you need to define which ABAP packages in an ABAP system are managed by which repository in gCTS and assign a gCTS-specific transport layer to the packages.

With gCTS, the process for continuous development and continuous integration is as follows:

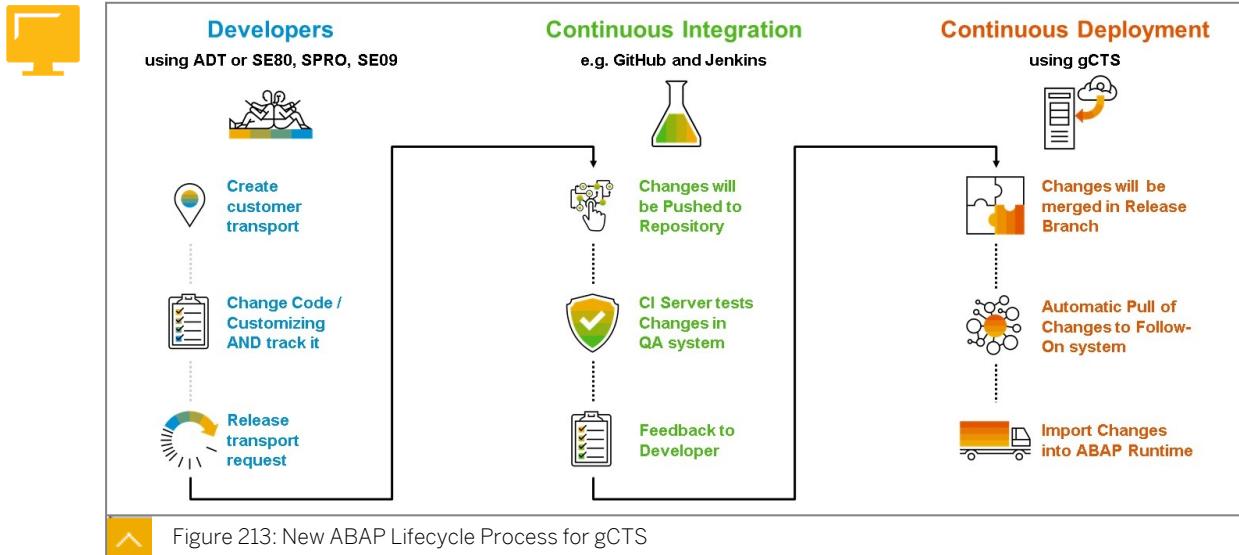


Figure 213: New ABAP Lifecycle Process for gCTS

As of SAP S/4HANA Server 2020, any type of ABAP development and customizing object can be stored in a *Git* repository. Whenever objects are changed in a development system, the classic CTS records them in transport requests. gCTS uses the transport requests to transfer and commit these objects to a repository. From there, the commits are pushed to a central *Git* repository, the *remote repository*.

The objects in the remote repository can then be transferred into the repository of any target system. From there, they can be imported into any other ABAP system. Internally, the well-known transport tools provide fast import, automatic object activation, and execution of after-import methods.

This enables you to perform rollbacks directly after importing a certain commit.

**Caution:**

The *rollback* feature should not be used in production systems. (Application) data that had been created based on newer coding is not protected.

gCTS prepares ABAP for continuous integration and provides options to set up similar development processes for ABAP as for non-ABAP platforms.

You can set up *continuous integration (CI) pipelines* that update your target systems automatically. CI processes involve using a pipeline that defines a sequence of steps to be performed and conditions that control the pipeline.

SAP provides a sample CI/CD pipeline scenario for *Git-enabled Change and Transport Management* (gCTS) using project "Piper". You can use it as a template and adapt it to your needs.



Hint:  
For example,

- The command `gctsCloneRepository` clones a *Git* repository from a remote repository to a local repository on an ABAP system.
- The command `gctsDeploy` pulls a commit from the corresponding remote *Git* repository to a specified local repository on an ABAP system.
- The command `gctsExecuteABAPUnitTests` runs ABAP unit tests for all packages of the specified repository.
- The command `gctsRollback` performs a rollback of commit(s) in a local ABAP system repository.

For more information on this command, see the project “Piper” pages on *GitHub* (for example, <https://sap.github.io/jenkins-library/steps/gctsDeploy/>).



Note:

Background information: When the CI server creates a new build, gCTS determines changed objects since the last deployment and creates a R3trans *data* file. When the CI server deploys the build to the runtime system, technically the previously created R3trans *data* file is deployed.

## Infrastructure for gCTS

The components needed to *Git-enabled Change and Transport System* (see the following figure) are:

- *Git Server*: In addition to the other components that are delivered by SAP, access to *Git* repositories on a *Git* server (*GitHub*, for example) is required.
- *Git Client*: The *Git* client is the java executable `abap2vcs.jar` that comes with the SAP Kernel. It communicates with the *Git* server. To run the *Git* client, a Java runtime is required that is not part of the standard installation. A Java runtime version of at least 1.8 is required. Ports 22 and 443 are used for outgoing communication.
- *ABAP Runtime*: gCTS is an extension of the classic ABAP *Change and Transport System*. You perform your software development in the ABAP runtime as usual, and you assign your changes to transport requests. When you release a transport request, a new *commit* is created, and the contents of the transport request is pushed to the *Git* repository. All the objects that are part of the released transport request are converted into files that are stored in the *Git* repository on the *Git* server. From there, they can be pulled to any target ABAP system that is connected to the respective kernel.

**Note:**

You can also configure the SAP system in a way that a commit is created when the task of the transport request (and not the transport request itself) is released. For more information, see the blog *Create a commit in Git when an ABAP task is released* (<https://blogs.sap.com/2020/08/05/create-a-commit-in-git-when-an-abap-task-is-released/>).

- **SAP Fiori App for Git-Enabled Change and Transport System (gCTS app):** The gCTS app is a standard SAP Fiori app that is integrated in *SAP Fiori Launchpad*. This app is a prerequisite for working with gCTS.

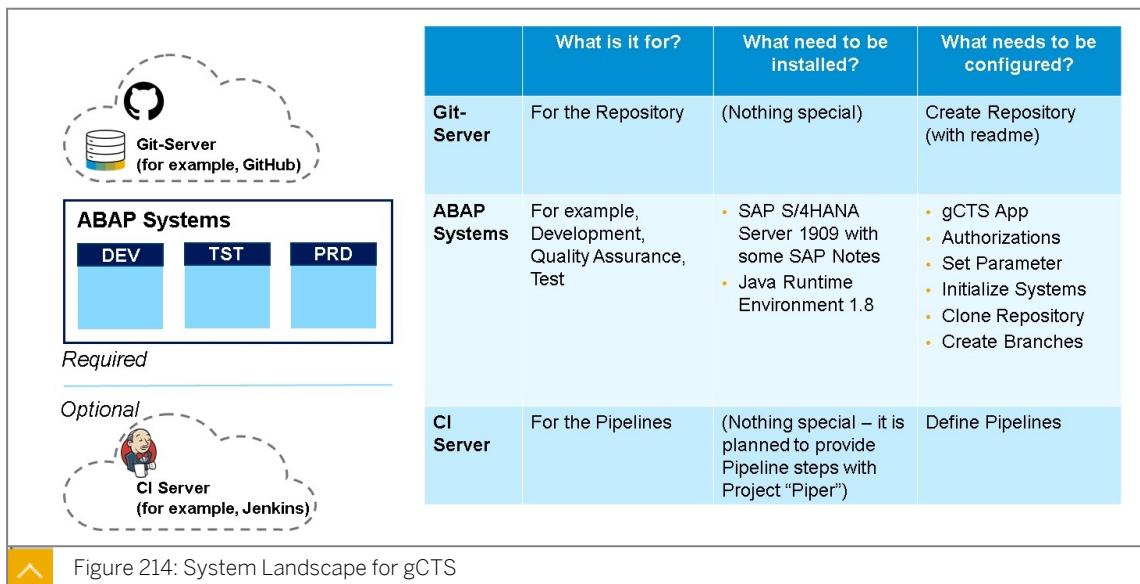


Figure 214: System Landscape for gCTS

To use gCTS, you need to configure the following (see figure above):

- Implement the gCTS app in your ABAP system: The gCTS SAP Fiori app is part of the *SAP\_BASIS\_TCR\_T* (*SAP: Application Services*) catalog and is installed as part of the UI for Basis Applications, software component *UIBAS001*. You need to add this catalog to a role. To use the gCTS app, among other things, you need to activate the OData Service *SCTS\_GCTS\_SRV* and the ICF Services */sap/bc/ui5\_ui5/sap/bc\_cts\_git* and */sap/bc/cts\_abapvcs*.

**Note:**

SAP ships the role *SAP\_BC\_GCTS\_ADMIN* as a template.

- Enable gCTS in your ABAP system: Use the *Enable gCTS* wizard within the gCTS SAP Fiori app to set mandatory parameters and initialize the system. The steps of the wizard include (among others):
  - A path to a working directory to store files in local repositories in the file system of the application server

- The path to the Java runtime on the file system of the application server to run the *Git* client

As the last step of the wizard, use the *Health Check* to verify the configuration status. Parameters can be set in the gCTS App on the *Configurations* tab.

- To work with Git repositories, you need to configure Git repositories on the ABAP system: In this way, you create a connection between your ABAP system and the remote Git repository on the Git server. By cloning the remote repository to the ABAP system, you make the contents of the remote repository accessible for your ABAP system by importing the contents of the repository to the ABAP system. For this you need the gCTS app, *Repositories* tab.
- For the use of *branches* for development in general (without the gCTS context), see the blog *A successful Git branching model* (<https://nvie.com/posts/a-successful-git-branching-model/>) as an introduction.

When you use gCTS, you need to define which objects or packages you want to manage using gCTS. You must decide whether you want to use gCTS only for new development projects, or whether you also want to switch existing developments to gCTS. It's important that you assign the ABAP packages that you use with gCTS to transport layers that are different from the transport layers used for classic CTS. The transport layer for gCTS is created when the repository is registered.



#### Note:

SAP S/4HANA 2020 FPS 02 (based on SAP\_BASIS 7.55 FPS02), the gCTS registry enables you to define a specific *Git* repository as the original repository of an ABAP object. This might be useful, if you work in a distributed development environment, for example, or for customizing projects. For details, see SAP Note [3046346 – gCTS Registry: Central Note](#) and the blog *Integrating gCTS with Transport Organizer processes* (<https://blogs.sap.com/2022/05/24/integrating-gcts-with-transport-organizer-processes/>).

## Additional Information on gCTS

For an introduction on agile software development, see the blog *Introducing Agile Software Engineering in development* (<https://blogs.sap.com/2018/05/02/introducing-agile-software-engineering-in-development/>).

For more information on gCTS, see:

- the online documentation of SAP S/4HANA (*Product Assistance*) area SAP S/4HANA → *Enterprise Technology* → *ABAP Platform* → *Administrating the ABAP Platform* → *Administration Concepts and Tools* → *Solution Life Cycle Management* → *Software Logistics* → *Change and Transport System* → *Change and Transport System - Overview* → *Git-Enabled Change and Transport System*.
- the blog *How Change Request Management (ChaRM) Leverages Git-enabled CTS (gCTS)* (<https://blogs.sap.com/2021/09/05/how-change-request-management-charm-leverages-git-enabled-cts-gcts/>) on how to use SAP Solution Manager *Change Request Management* with gCTS.
- SAP Note [2821718 – Central Note for Git-enabled Change and Transport System \(gCTS\)](#) as the central point of entry for information related to gCTS.

- SAP Note [2919181](#) – *Troubleshooting Git-enabled Change and Transport System (gCTS)* with a link to a guided answer on troubleshooting.
- SAP Note [3046346](#) – *gCTS Registry: Central Note* as central entry point on information how to you define a specific *Git* repository as the original repository of an ABAP object.



## LESSON SUMMARY

You should now be able to:

- Outline the basic idea of DevOps and gCTS

## Learning Assessment

1. You want to use the *enhanced Change and Transport System* (enhanced CTS). Which of the following are deployment methods?

*Choose the correct answers.*

- A HTTP-Based Deployment
- B Deploy controller, EJB-based deployment, script-based deployment (with AS Java)
- C RFC-based deployment
- D Client-based deployment

2. You want to configure the *enhanced Change and Transport System* (enhanced CTS). Which of the following steps must be performed?

*Choose the correct answers.*

- A Create an HTTP connection in the CTS system (transaction SM59)
- B Configure the back-end systems in *Transport Management System* of the CTS system (transaction STMS)
- C Install an additional add-on in the CTS system (transaction SAINT)
- D Create a second database schema in the database of the back-end systems (transaction DBACOCKPIT)

3. You have configured the *enhanced Change and Transport System* (enhanced CTS) with stand-alone SAP HANA databases as back-end systems. The CTS system has the SID SMA, the SAP HANA databases have the SIDs HAD, HAQ, and HAP. Now you want to create a transport request to transport among the SAP HANA databases: You can create the transport requests in the *Transport Organizer* (transaction SE09) in system SMA.

*Determine whether this statement is true or false.*

- True
- False

## Learning Assessment - Answers

1. You want to use the *enhanced Change and Transport System* (enhanced CTS). Which of the following are deployment methods?

Choose the correct answers.

- A HTTP-Based Deployment
- B Deploy controller, EJB-based deployment, script-based deployment (with AS Java)
- C RFC-based deployment
- D Client-based deployment

You are correct! The two deployment methods are HTTP-Based Deployment and Deploy controller, EJB-based deployment, script-based deployment (with AS Java). There is no RFC-based deployment and Client-based deployment.

2. You want to configure the *enhanced Change and Transport System* (enhanced CTS). Which of the following steps must be performed?

Choose the correct answers.

- A Create an HTTP connection in the CTS system (transaction SM59)
- B Configure the back-end systems in *Transport Management System* of the CTS system (transaction STMS)
- C Install an additional add-on in the CTS system (transaction SAINT)
- D Create a second database schema in the database of the back-end systems (transaction DBACOCKPIT)

You are correct! You must create an HTTP connection in the enhanced CTS system and configure the back ends in *Transport Management System* of the CTS system. You do not need to install an additional add-on in the CTS system or to create a second database schema in the database of the back ends.

3. You have configured the *enhanced Change and Transport System* (enhanced CTS) with stand-alone SAP HANA databases as back-end systems. The CTS system has the SID SMA, the SAP HANA databases have the SIDs HAD, HAQ, and HAP. Now you want to create a transport request to transport among the SAP HANA databases: You can create the transport requests in the *Transport Organizer* (transaction SE09) in system SMA.

Determine whether this statement is true or false.

- True  
 False

You are correct! You cannot create the transport requests in the *Transport Organizer* (transaction SE09), but rather in the *Transport Organizer Web UI* (transaction STMS) of system SMA, because the transport request must have HAD as the source system. This is possible to choose in the *Transport Organizer Web UI* only.

# Glossary

## **Backup Domain Controller**

An SAP system that can assume the functions of the transport domain controller if the transport domain controller fails.

## **Client**

A client is a self-contained unit in an SAP system, in commercial, organizational, and technical terms, with its own user master data and set of table key ranges.

## **Client Change Options**

The client change options define which parts of the SAP system are modifiable when working in this client.

## **Client Compare**

Compare customizing settings of two clients via RFC.

## **Client Copy**

Copying the contents of a whole client or parts of it (according to the client copy profile used) from one client to another; the two clients can be in the same SAP system (local client copy), in different SAP systems (remote client copy) or the copy process can be performed by creating a transport request (client transport)

## **Client Copy Profiles**

Definition of which part of the client should be copied in a client copy or client transport process

## **Client Maintenance**

Maintaining customizing settings of one client according to the customizing settings of another client, based on a client compare via RFC.

## **Client Transport**

Creating a transport request out of the contents of a whole client or parts of it (according to the client copy profile used) and then importing this transport request into a client either of the same SAP system or of a different SAP system

## **Client-Specific Data**

Client-specific data, such as user master data, application data, and customizing settings, which affects only one client.

## **Consolidation Route**

The regular transport path of a repository object from the development system to the consolidation system. Technically this is an “export / import” route.

## **Correction**

Change to the original of an object

## **Cross-Client Data**

Cross-client data, such as cross-client customizing data and all repository objects, which affect the whole SAP system environment.

## **CTO (Change and Transport Organizer)**

This is part of the Change and Transport System (CTS). It provides functions for organizing software development projects. It is designed to support projects of all sizes, whether they are carried out centrally or in a distributed environment.

## **CTS (Change and Transport System)**

These are tools for creating and organizing transport requests, exporting, distributing and importing them. The CTS consists of the Change and Transport Organizer (CTO) and the Transport Management System (TMS)

## **CTS Project**

Customizing should be organized in projects. The project then contains all the customizing activities, modifications, and development. A project should be used for grouping transport requests.

## **Current Settings**

Customizing settings which can be performed directly in the productive client in the productive system

## **Customizing**

Overall procedure to set up one or more SAP systems. In particular, the aim of the procedure is to adapt the non-company specific and industry-specific functions shipped in an SAP system to the specific business requirements of your company.

## **Customizing Transport Request**

Transport requests that contain client-dependent data, such as client dependent customizing

## **Development System**

The SAP system of an SAP system landscape in which development and customizing take place.

## **Domain Link**

A link between different transport domains. A domain link enables transparent access to all linked SAP systems, display of import queues in all SAP systems and configuration of transport routes between SAP systems in different domains.

## **Enhanced CTS**

An extension of Change and Transport System (CTS), used to execute and monitor the transports between non-ABAP systems.

### **Extended Transport Control**

Extended Transport Control enables you to define client-dependent transport routes (consolidation, delivery), transport target groups and the assignment of clients to transport layers.

### **External System**

External systems are like virtual systems. This means that they are accessed using a communication system. However, for this type of SAP system, a separate transport directory is also defined.

### **Implementation Guide (IMG)**

The IMG provides a hierarchical list with customizing activities in an SAP system. This customizing is grouped by modules and ordered in the sequence in which the individual activity has to be performed.

### **Import Monitor**

Monitor used for tracking the import of transport requests

### **Import Queue**

The import queue displays the transport requests that are to be imported in a specific SAP system, in the correct order.

### **Local Client Copy**

Copying the contents of a hole client or parts of it (according to the client copy profile used) from one client to another client in the same SAP system

### **Modification**

Repair of an SAP object

### **Modification Assistant**

The Modification Assistant helps to modify SAP standard objects and simplify the modification adjustment process.

### **Note Browser**

The Note Browser gives an overview of all SAP Notes available in an SAP system

### **Package**

Formerly known as development class. A package provides a logical grouping of repository objects for coordination of development efforts, defines a

repository object's transport layer and can control naming objects

### **Pre-production System**

The (optional) SAP system for the final integration test once the scope of the release has been fixed (release test).

### **Production System**

The SAP system, of an SAP system landscape, in which end users carry out productive work.

### **QA Approval Procedure**

The TMS Quality Assurance (QA) approval procedure increases the quality and the availability of the production systems by letting you check transport requests in the quality assurance system before they are delivered to subsequent SAP systems.

### **Quality Assurance (QA) Approval Procedure**

The TMS Quality Assurance (QA) approval procedure increases the quality and the availability of the production systems by letting check transport requests in the quality assurance system before delivering them to subsequent SAP systems. Transport requests are only forwarded to the delivery systems if all the QA approval steps are processed for the request in the QA system and the request has been approved.

### **Quality Assurance System**

The SAP system, of an SAP system landscape, in which integrated testing and quality assurance take place.

### **R3trans**

Program at operating system level to establish a connection to the database and to perform the necessary steps in the database for exports and imports

### **RDDIMPDP (Import Dispatcher)**

Job and program in the SAP system, that controls post import activities, such as table structure activation and conversion, program generation and versioning

### **Remote Client Copy**

Copying the contents of a hole client or parts of it (according to the client copy profile used) from one client to another client in a different SAP system via RFC

## **Repair**

Change to a copy of an object

## **Repository**

The collectivity of all programs, function modules, table structures, data elements, domains, and so on that belong to an SAP system.

## **SAP Activate**

Innovation adoption framework introduced for SAP S/4HANA that combines SAP Best Practices, methodology, and guided configuration delivered with a reference landscape. The SAP Activate methodology is SAP guidance for implementation, enhancements, upgrades or co-innovation of SAP solutions starting with SAP S/4HANA. It enables costeffective, agile, and fast delivery of the SAP solution to the customer and supports deployments in the cloud, on premise, or in hybrid deployment.

## **SAP ECC**

SAP Enterprise Central Component. Formerly known as SAP R/3. As of release 5.0, SAP R/3 was renamed to SAP ECC. SAP ECC is the main SAP component and, the main component in SAP ERP.

## **SAP ERP**

SAP ERP is an SAP solution. It contains several SAP components. The main component is the SAP ECC (formerly known as SAP R/3).

## **SAP NetWeaver**

SAP NetWeaver is comprised of different components, such as SAP BW, SAP EP, SAP PI, and so on. These technical components of SAP NetWeaver can be installed as needed to serve different requirements.

## **SAP Note Assistant**

Tool for automatically applying SAP Notes concerning repository object changes in an SAP system

## **SAP S/4HANA Server**

SAP S/4HANA (SAP S/4HANA Server) is the newly developed successor of SAP ECC.

## **SAP Software Change Registration (SSCR)**

Process for tracking, which users are allowed to develop and which SAP objects are changed in an SAP system

## **SAP Solution Manger**

The SAP Solution Manager is the entry point for implementation, operation and continuous business improvement for the SAP applications.

## **SAP Support Package**

Kind of transport request, for problem fixing, legal changes, solving security issues and new functionality for a software component of an SAP system

## **SAP System Landscape**

Any required SAP systems and clients, their meaning and the transport routes for implementation and maintenance processes. The SAP system landscape may, for example, comprise a development system, a quality assurance system, and a production system.

## **SAP System Update/Upgrade**

Procedure for exchanging the older repository of an SAP system with a newer repository; this includes changing table structures at SAP system and database level, converting application data, exchanging the kernel executables

## **Special Transport Workflow**

The Special Transport Workflow can be used, if transport corrections are needed urgently, or if transport requests are required that do not follow the defined transport routes. This process works with a workflow and each step is documented.

## **Support Package Manager (SPAM)**

Tool in the SAP system for applying SAP Support Packages

## **System Change Options**

The system change options define whether or not repository objects, and cross-client customizing objects, are globally modifiable.

## **Task**

Organizational part of a transport request.

## **Transport Control Program (tp)**

tp is a program at operating system level that performs exports and imports for transporting objects between SAP systems. The program also ensures that the steps in exporting and importing objects are performed in the correct order.

## **Transport Directory**

A directory that manages all data to be transported between SAP systems.

## **Transport Domain**

All SAP systems that are managed jointly by the TMS.

**Transport Domain Controller**

SAP system in which the transport route configuration is maintained centrally for all SAP systems in the same transport domain.

**Transport Group**

All SAP systems that access a common transport directory.

**Transport Management System (TMS)**

Set of all tools in the SAP system for organizing, performing, and monitoring transports between SAP systems. The TMS is part of the Change and Transport System.

**Transport Organizer**

See: Change and Transport Organizer (CTO)

**Transport Organizer Tools**

Collection of tools that support the work with the Change and Transport System (CTS)

**Transport Request**

A transport request is used for transferring objects together into another SAP system.

**Transport Target Group**

Transport target groups can serve as transport targets from consolidations and deliveries for simultaneously servicing different SAP systems and/or clients.

**Workbench Transport Request**

Transport request that contain client-independent data, such as repository objects or client independent customizing