Capgemini

SAP BASIS Introductory Training Program

Day 8 - Agenda

- Concepts of User & Authorization – AS JAVA
- Break
- User and Role Management - AS JAVA
- Lunch Break
- CTS: Overview & Architecture
- Break
- CTS: Transport Management System
- Exercise Break Out Session

User and Authorization Concepts – AS JAVA

User and Authorization Concepts – AS JAVA

AS Java provides an open architecture supported by service providers for the storage of user and group data. The AS Java is supplied with the following service providers which are also referred to as a user store:

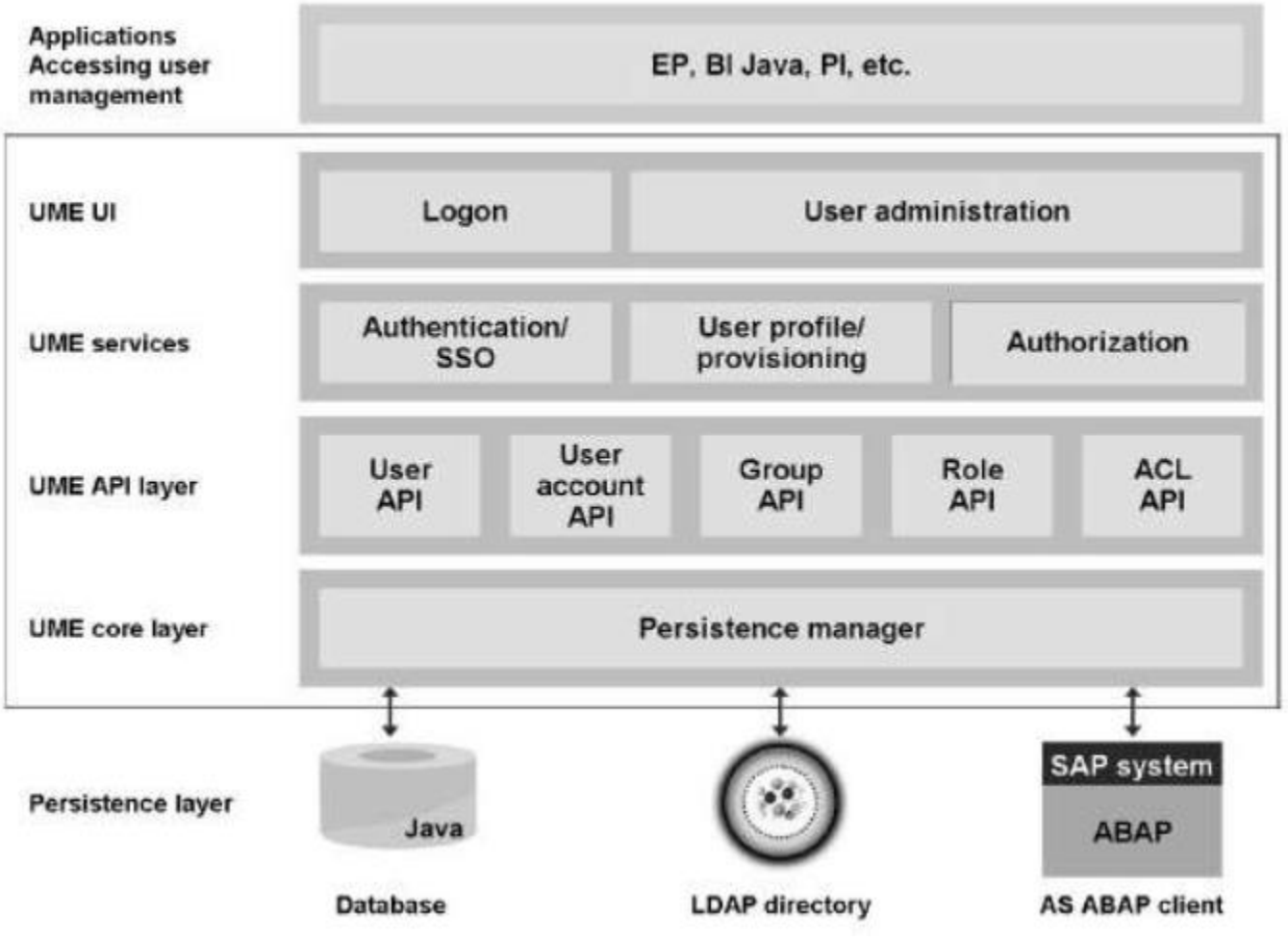
- DBMS provider: storage in the system database
- UDDI provider: storage via external service providers (Universal Description, Discovery and Integration)
- UME provider: Connection of the integrated User Management Engine

The DBMS and UDDI providers implement standards and therefore ensure that AS Java is J2EE-compliant. When AS Java is installed, SAP's own User Management Engine (UME) is always set up as the user store and is the correct choice for most SAP customers. The UME is the only way to flexibly set up and operate user and authorization concepts.

Important Features of the UME

- The UME has its own administration console for administering users. It allows the administrator to perform the routine tasks of user administration,such as creating users and groups, role assignment, and other actions.
- Security settings can be used to define password policies, such as minimum password length and the number of incorrect logon attempts before a user is locked.
- The UME provides different self-service scenarios that can be used by applications. For example, a user can change his or her data, or register as a new user. Newly-created users can be approved using a workflow.
- User data can be exchanged with other (AS Java or external) systems using an export/import mechanism.
- The UME logs important security events, such as a user's successful logons or incorrect logon attempts, and changes to user data, groups, and roles.

UME Architecture

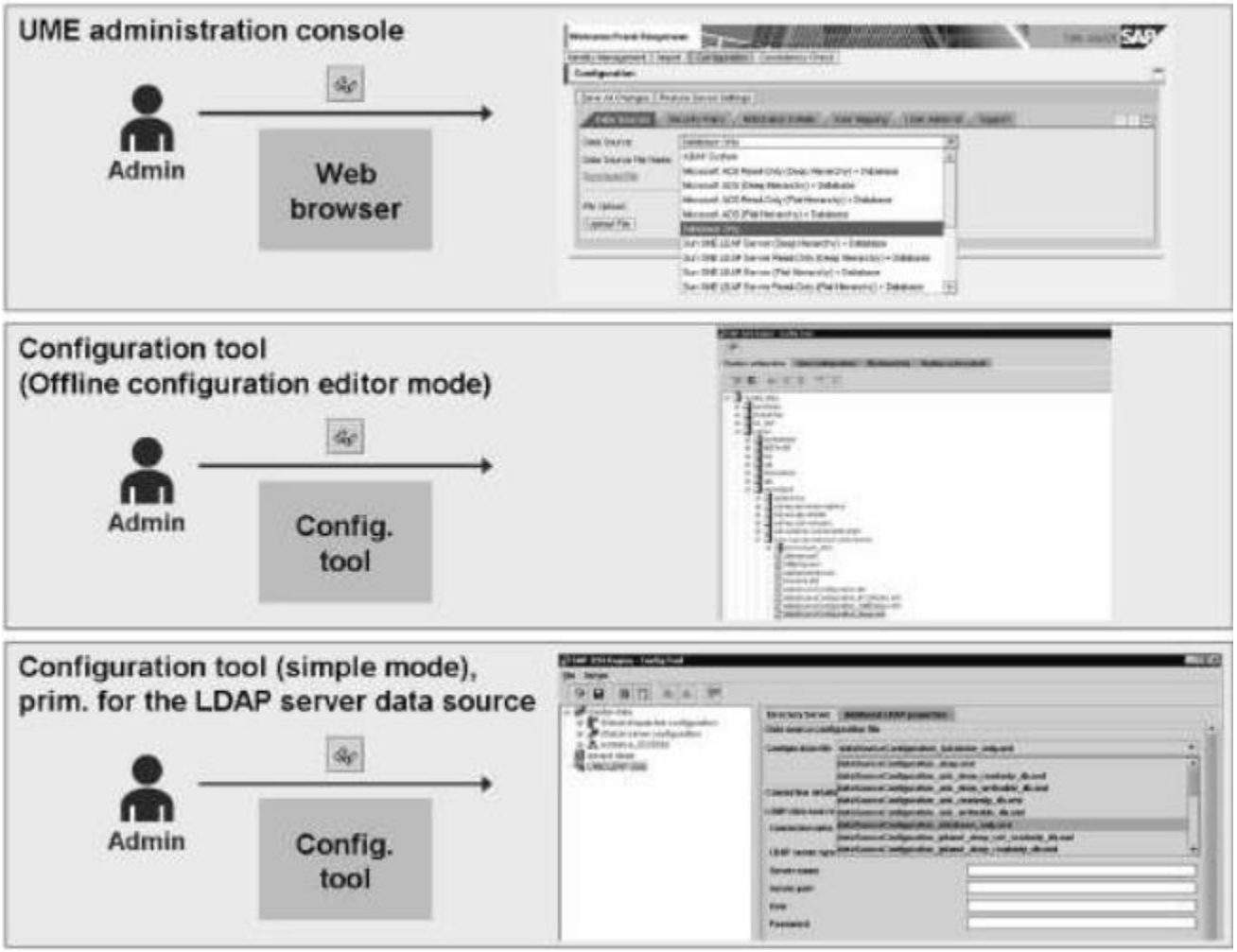


UME Architecture Details

The UME is a Java application which runs on SAP NetWeaver AS Java and which covers the following functional areas:

- UME Core Layer: Provides persistence managers between the application programming interface and the user management data sources - these control where user data such as users, user accounts, groups, roles and their assignments are read from or written to, with the result that applications which use the API do not have to know where the user management data is stored.
- UME API Layer: This layer provides programming interfaces (APIs) not just for UME developers but also for customers and partners. This means that you can access the UME functions with the Java programs which you develop yourself.
- UME services: The UME provides the following services to higher-level software layers:
  - Log-on procedure and Single Sign-On (log-on to AS Java is taken over for other systems and vice versa)
  - Provisioning processes via user master data Authorization Concept
- UME UI: The UME is responsible for the user interface which, in some log-on procedures, appears in the Web browser, as well as for the UME Administration Console

Tools for UME Configuration



Break

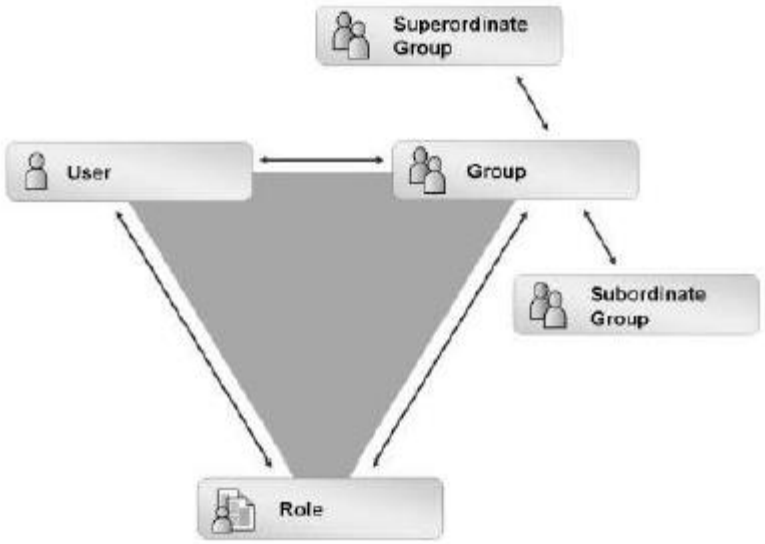
User and Role Management – AS JAVA

User and Group Administration

In the UME environment, the term Principle designates the following central "objects":

Principle	Meaning
User	General properties of a user (such as name, e-mail, telephone number etc.)
User account	Logon-related properties of a user (such as password, validity, lock indicator etc.)
Group	Set of user and/or groups
Role	Set of (Java) authorizations

The figure on the right hand side shows how principles are assigned



Assigning Roles

It is also possible to assign roles to users directly. The Principle group supports hierarchies of groups. A group may also possess superordinate and subordinate groups. Users actually possess the roles which

- are directly assigned to them
- are assigned to the groups to which they belong
- are assigned to the superordinate group of the groups to which they belong

When performing a search in the UME Administration Console, you must check the Search Recursively field if you want to see indirectly assigned principles.

Special features of the ABAP Data Source

If you use a client of the ABAP system as the data source , then UME behaves as follows

- The ABAP users are visible in AS Java and can log onto AS Java with their ABAP passwords.
- The ABAP roles are depicted in AS Java as UME groups of the same name.
- In AS Java, the assignment of ABAP users to ABAP (composite) roles appears as the assignment of UME users to UME groups.

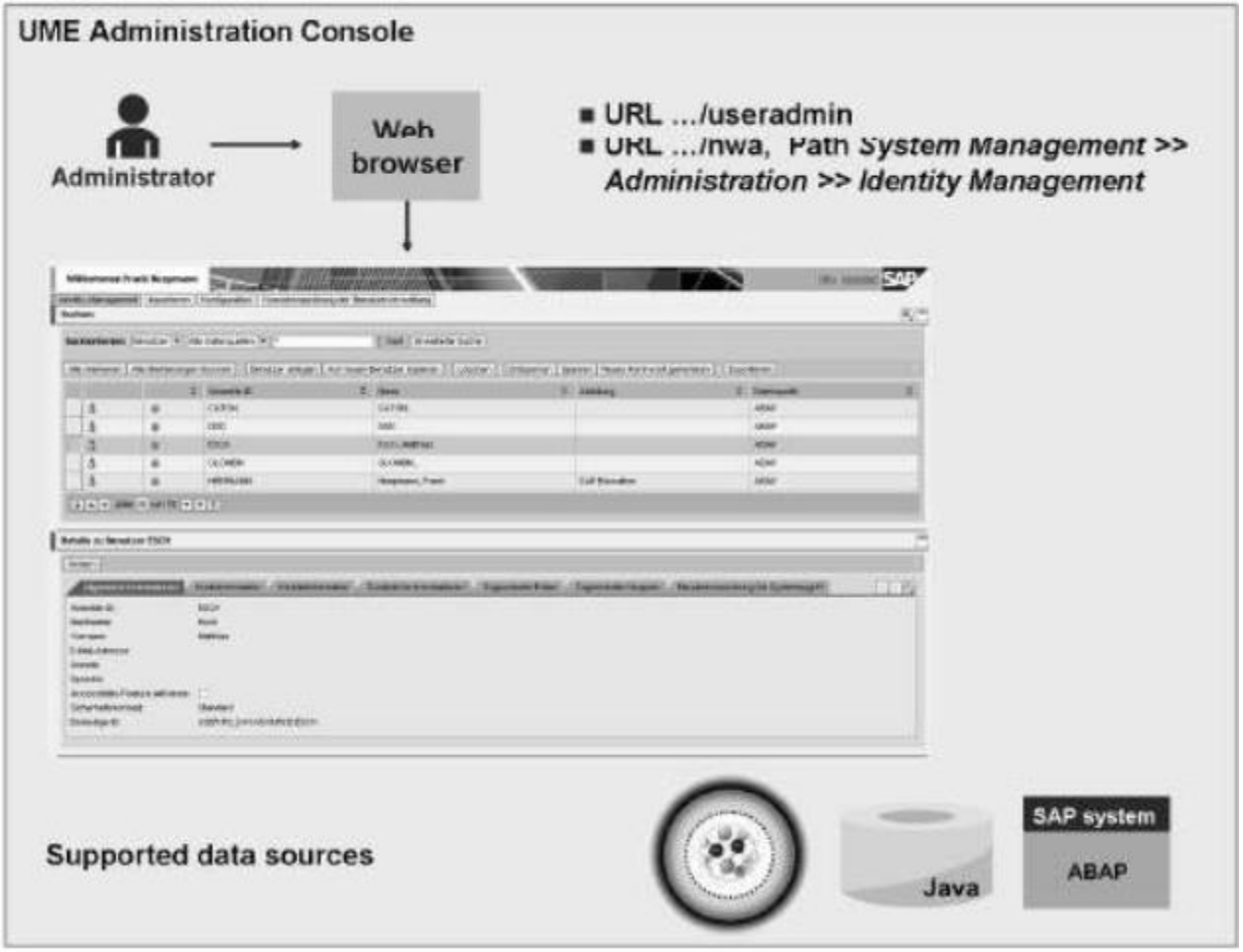
Using ABAP as a Data Source

The reason for this group administration concept is the shared authorization administration for applications that have both ABAP and Java components. Applications such as PI, for example, are made of both ABAP and Java components. The ABAP authorizations are mapped with PFCG roles. The J2EE authorizations are realized using UME roles. A user should be assigned a PFCG role in the ABAP system and a UME role on the Java side for the user to have both ABAP and Java authorizations. To avoid this, the PFCG roles are visible as groups in the UME. The PFCG role (a group) can be assigned a UME role in the UME. If a user is assigned the PFCG role in the ABAP system, he or she automatically also receives the authorizations from the UME role. Assigning authorizations therefore becomes simpler.

The connection between the UME in an AS Java and user management in an AS ABAP is established via the Java Connector (JCo). A communication user existing in ABAP is stored as a UME parameter (this usually has SAPJSF in its name). This communication user's ABAP authorization determines whether it is possible to modify ABAP user master records using UME resources.

- The role SAP_BC_JSF_COMMUNICATION_RO gives the UME read access to the user data in the AS ABAP.
- The role SAP_BC_JSF_COMMUNICATION gives the UME write access to the user data in the AS ABAP

UME Administration Console



User Types in AS JAVA

UME User Types

User Type	Logon to AS Java	Password Rules	Mapped ABAP user types (with ABAP system as data source)
Standard	possible	applies	*Dialog*
Technical users	possible	does not apply	*System*
Internal service user	not possible	applies	–
Unknown	possible	applies	*Communication*, *Service* and *Reference*

Authorization Concept in AS JAVA

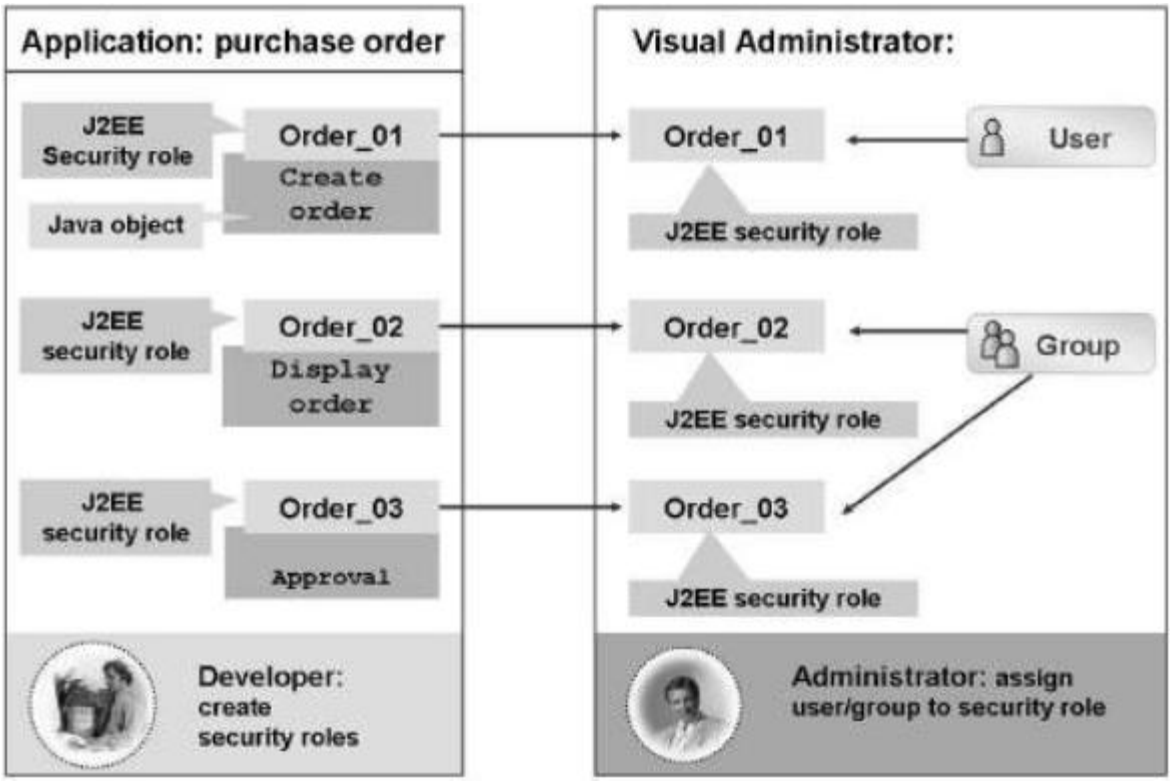
- You can use authorizations to control which users can access a Java applications,and which users are permitted for a user. Authorizations are combined as roles and then assigned to a user or a user group by an administrator. The UME administration console and Visual Administrator tools are used to assign authorizations.Authorization checks are built into a Java application. You must distinguish between the following authorization checks:
  - J2EE security roles
  - UME roles

- With both types of authorization check, the developer needs to define the authorizations query in the application. The developer decides which type of authorization check is to be used. This means in practice that whether J2EE security roles or UME roles are used depends on the application.

- J2EE security roles are part of the J2EE standard. UME roles are an (SAP) extension of the J2EE security roles. You can define the same authorization checks with J2EE security roles and UME roles. However, it is easier and more precise to assign authorizations with UME roles. A J2EE security role comprises one object and UME roles many authorization objects (known as actions). This means that many J2EE security roles but perhaps only one UME role need to be assigned for the same authorizations. It is recommended that you always use UME roles, except in cases in which J2EE security roles are sufficient.

- **Note:** A role in the ABAP environment is roughly equivalent to a UME role. An authorization object in the ABAP environment can be compared to a security role.

Structure of a J2EE Security Role



The figure shows the Order application as an example. For this application, a developer creates objects such as Create order, Approve order, and so on. If you are using J2EE security roles, a security role must be created for each object. The role is defined in the deployment descriptor (XML file) of a specific application.If the application is made available on the J2EE server, the administrator must add user names or user groups to each of these security roles for the users that are to use this application. The administrator must assign each single authorization/J2EE security role individually to a user or a group.

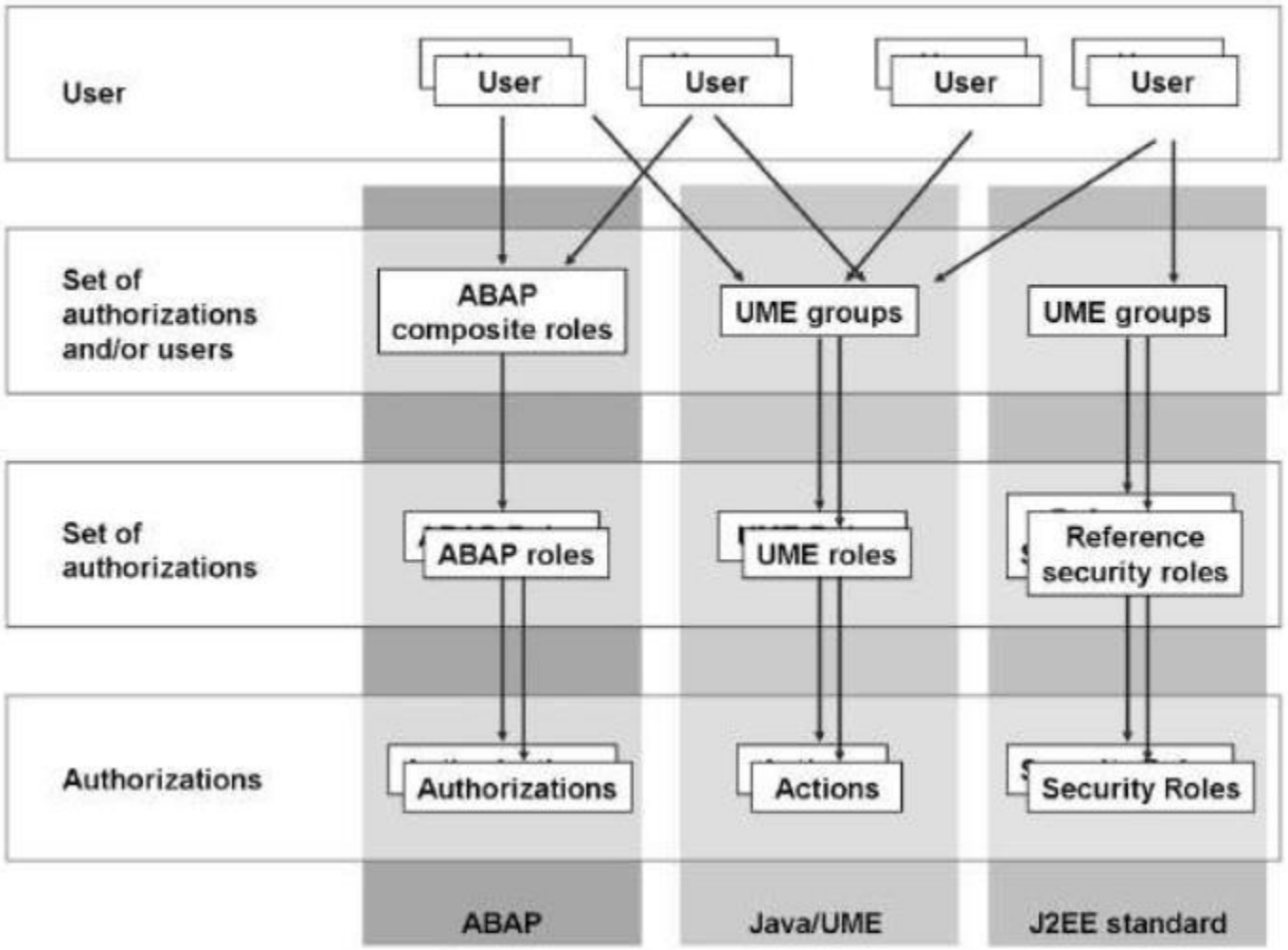
Using Visual Admin to assign a security role

- You can use the Visual Administrator to assign security role to a user or group. The Security Provider service of SAP NetWeaver AS Java must be running, and the user that wants to make the assignment must have administration authorizations.
- A J2EE security role can be assigned
  - either directly to users and/or groups
  - or as a so-called reference role to precisely one J2EE security role in the component SAP-J2EE-Engine
- To assign security roles, proceed as follows:

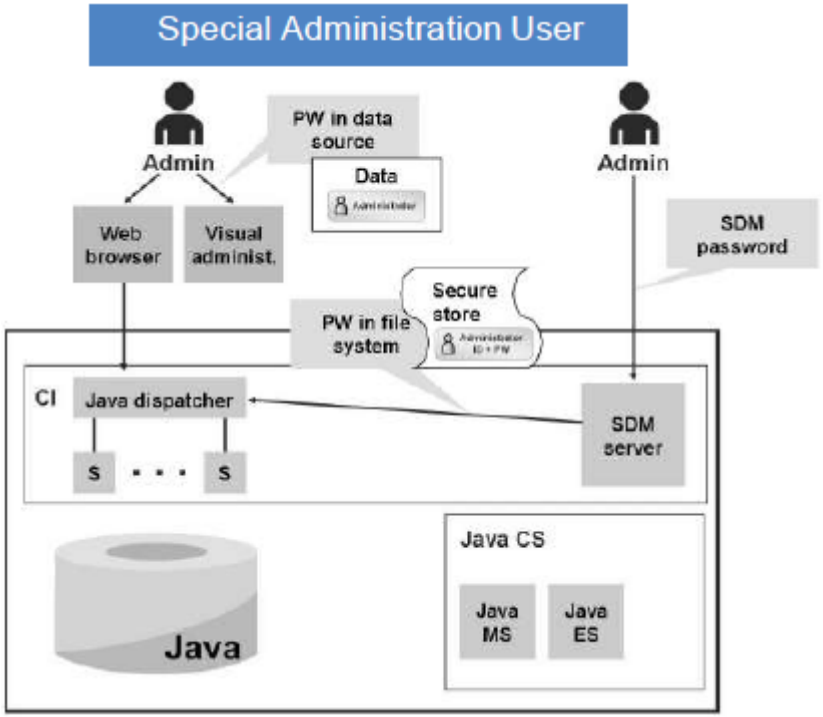
1. Start the Visual Administrator (\usr\sap\<SID>\<instance>\j2ee\admin\go).
2. Navigate to Server → Services → Security Provider → Runtime → Policy Configurations.
3. In the Components area, select the application (or service).
4. Choose the Security Roles tab page.
5. In the Security Roles area, select the security role that you want to assign.
6. Switch to change mode if necessary.
7. Depending on the type of J2EE security role, you either
   - perform assignment directly to users and/or groups
   - perform assignment to a reference security role

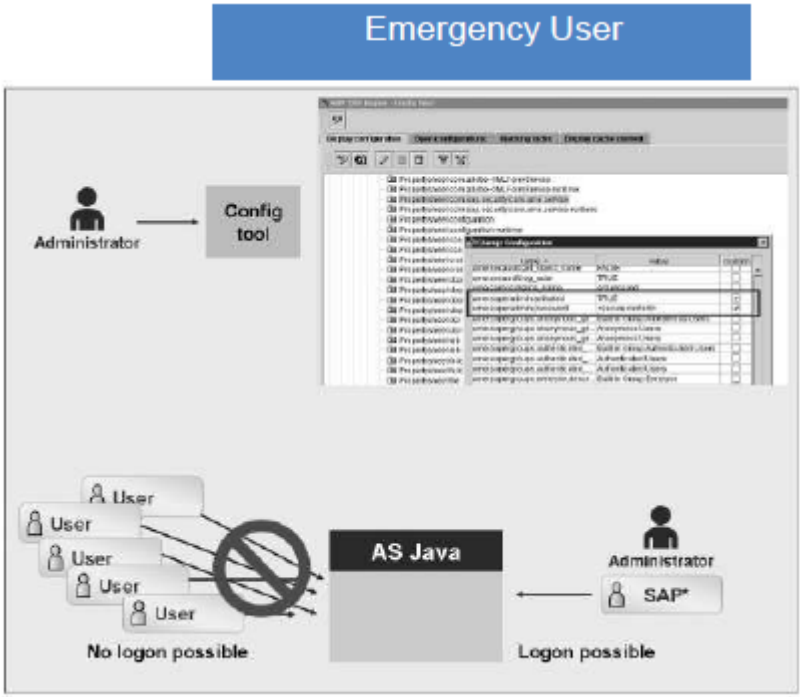
Using UME Console to assign roles

Comparison of Authorization Concept – ABAP/JAVA



Special Users in AS JAVA





You define the password for the administration user when installing an AS Java. After the installation, you can, of course, create other users with the same authorizations. However, the one and only administration user is special because this is not only used by the administrator in person but is also used for deployment via the SDM server

You need to activate an emergency user for the UME if the user management has been incorrectly configured and no one can log on to an application, or all administration users are locked. This emergency user is called SAP* and can log on to any application and to the configuration tools. The SAP* user has full administration authorizations and, for security reasons, does not have a default password. You set the password as part of emergency user activation.

Lunch Break

Change Transport System – Overview & Architecture

CTS Overview

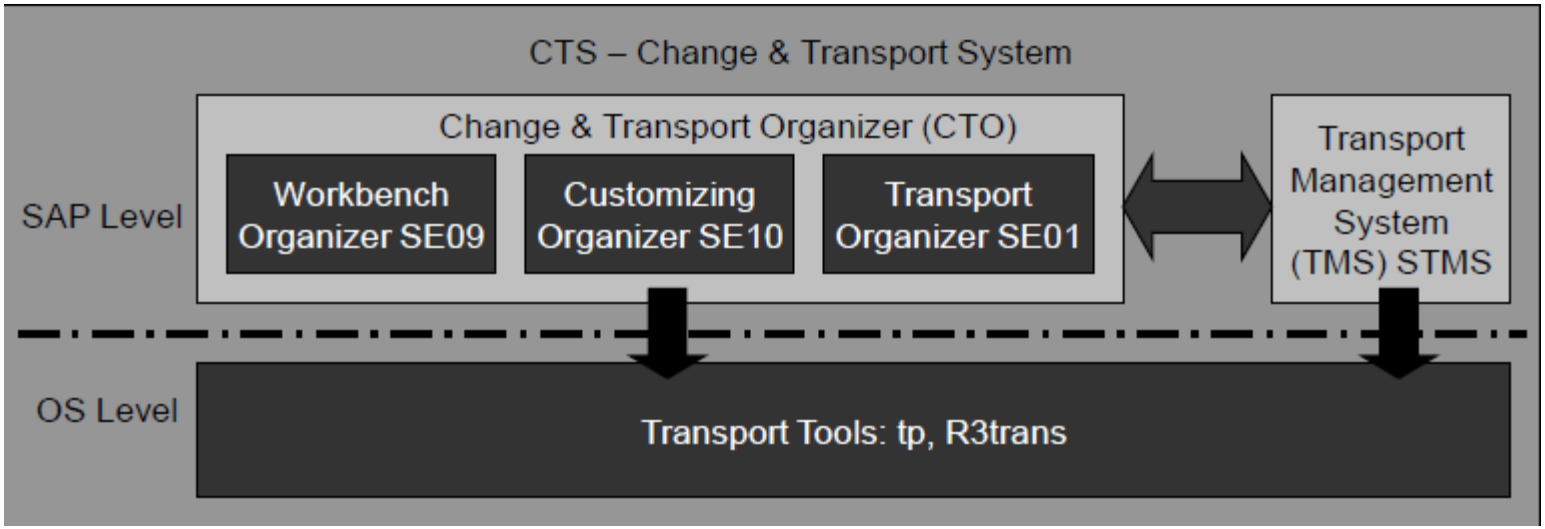
The Change and Transport System (CTS) is a tool that helps you to organize development projects in the ABAP Workbench and in Customizing, and then transport the changes between the SAP Systems in your systems landscape.

- **Repairs**

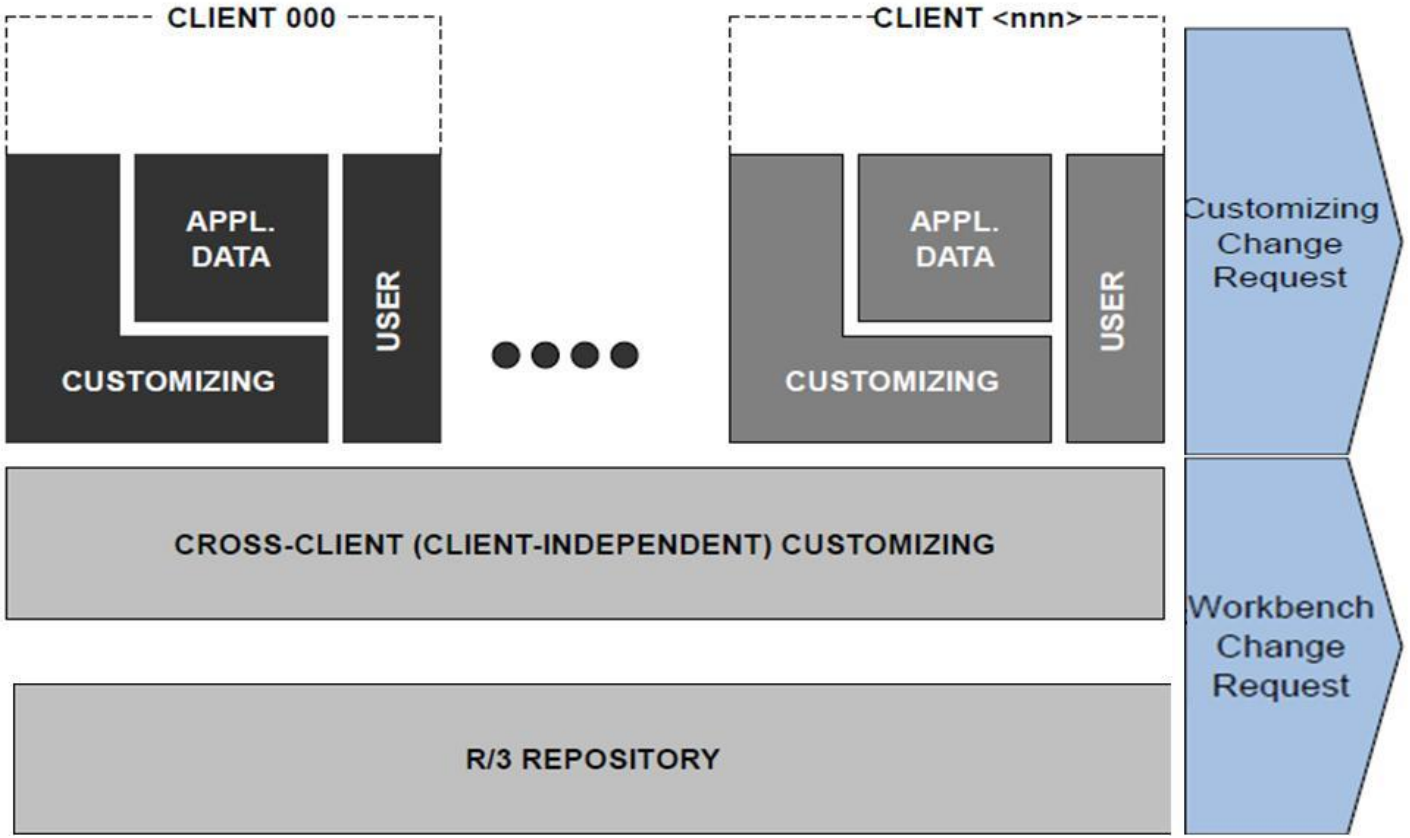
A repair is a change to an object that is not owned by the current system. Repairs are normally done on SAP owned objects

- **Corrections**

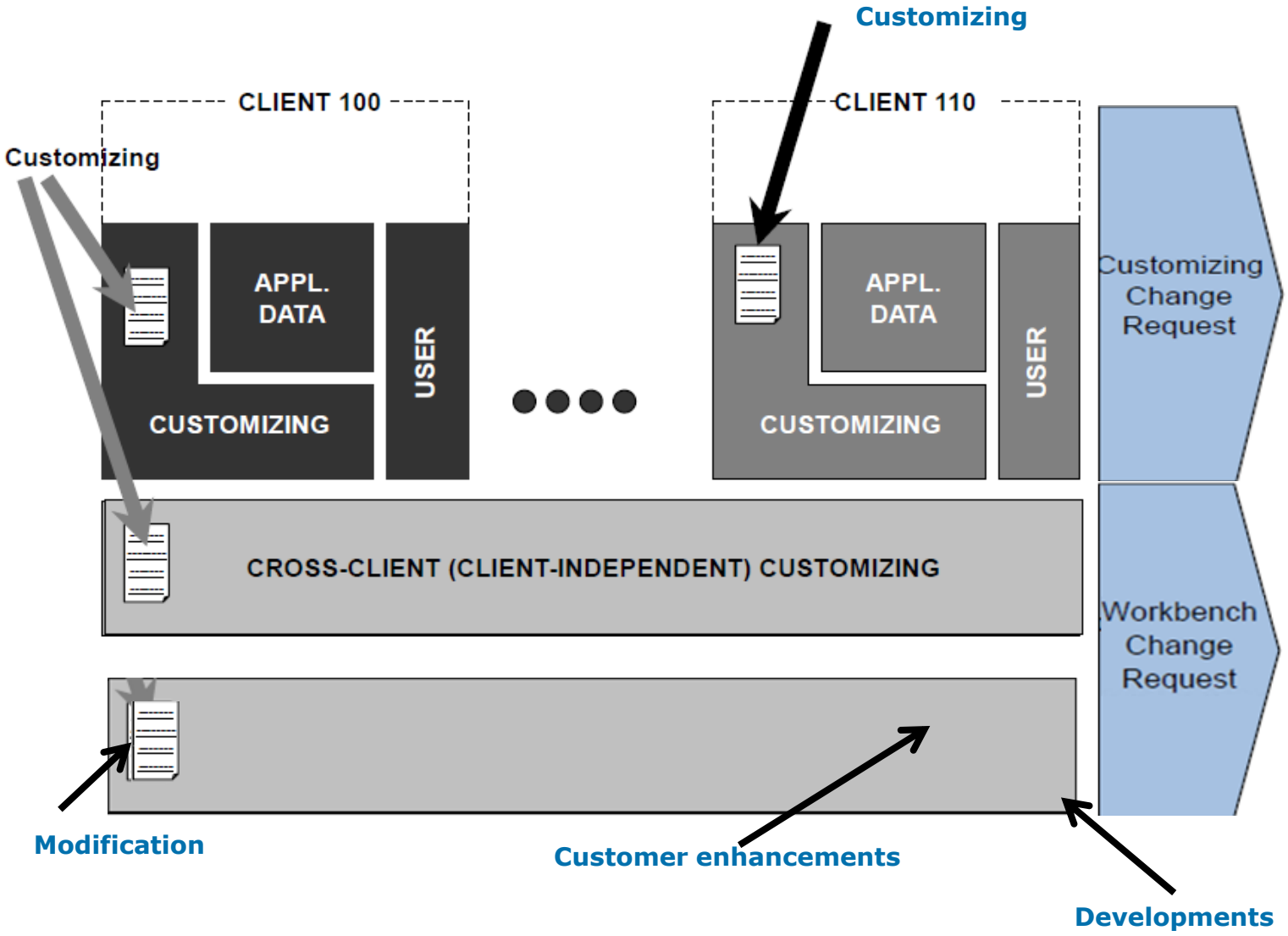
A correction is a change to an object that is owned by the current system



DATA Structure of R/3 System

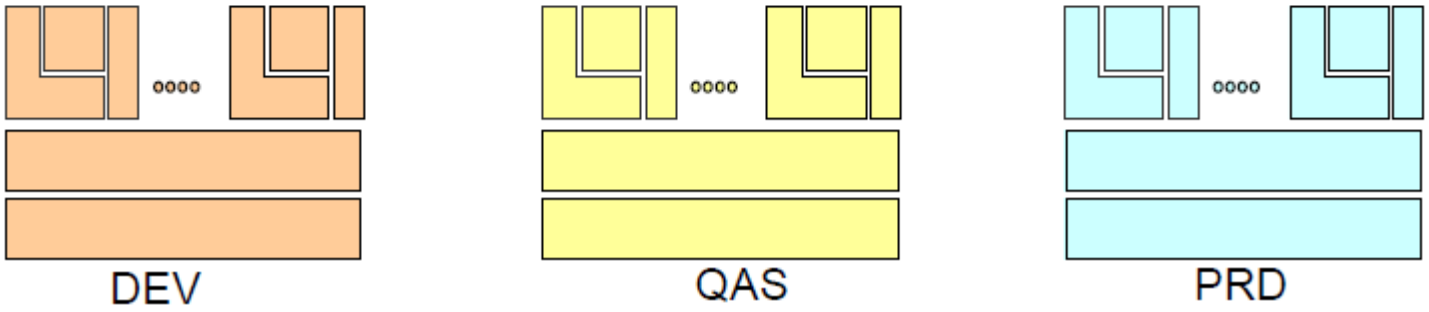


Types of Changes in SAP System



R/3 Client Roles

- Critical Client Roles
  - Customizing & Development – Customizing and Development work in ABAP workbench
  - Quality Assurance – Environment for testing and verifying new and existing Customizing settings and Business Application functionality
  - Production – Production system with real data

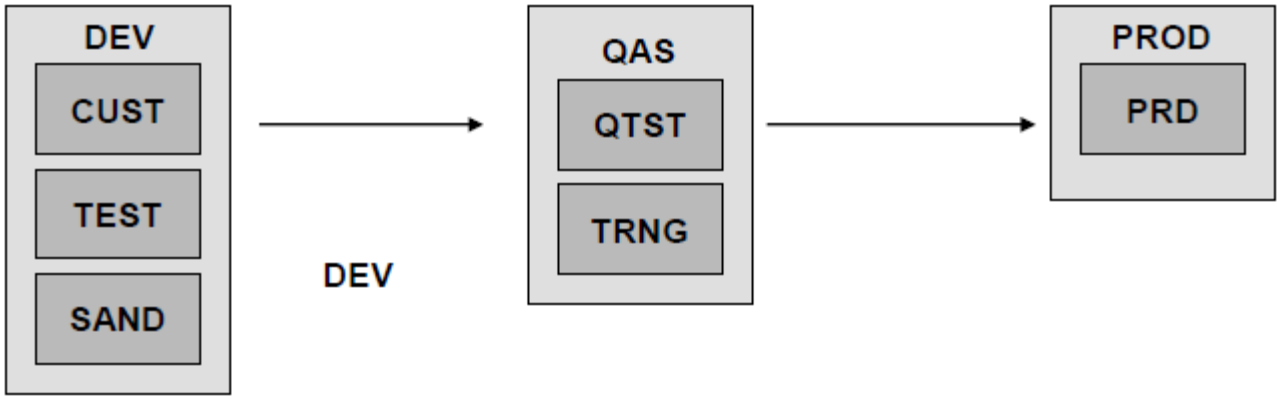


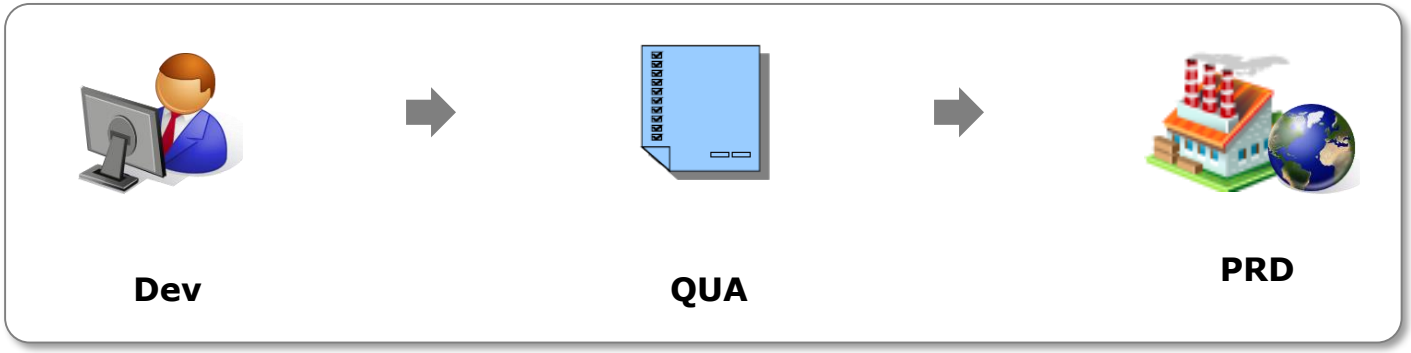
System & Client Change Options

- System Change Options
  - The system change option defines whether or not Repository objects and client-independent customizing objects are globally modifiable.
  - To reach the system change option use transaction SE06 and choose System Change Option.
  - Four settings
    - Modifiable
    - Restricted modifiability
    - Not modifiable; enhance able only
    - Not modifiable; not enhance able

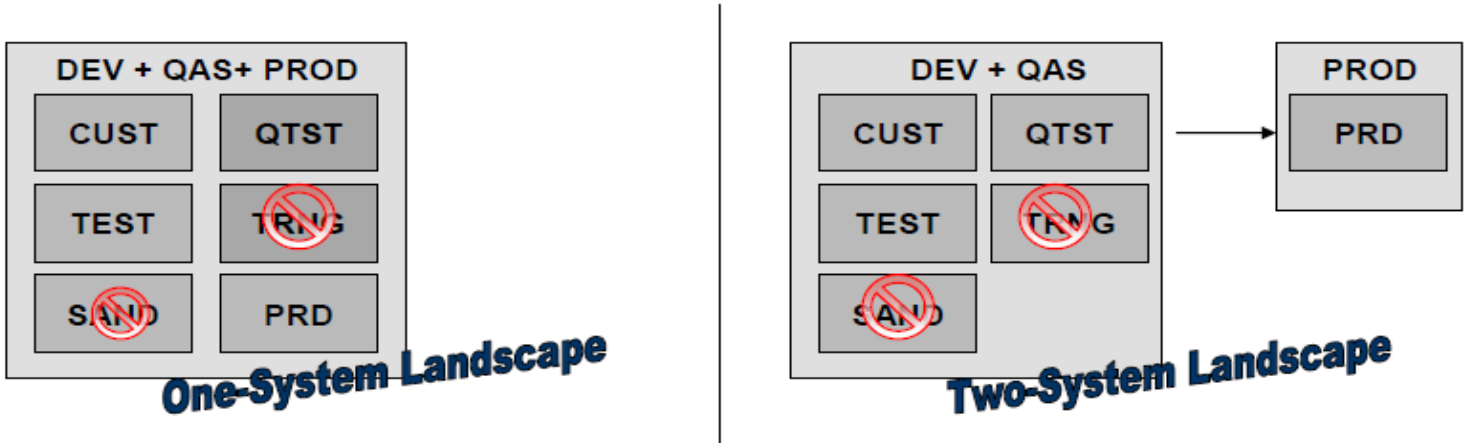
- Client Change Options
  - Client change options that are found in the clients master table T000, can be maintained by using transaction SCC4.
  - The two settings that must be maintained to implement controls on where changes are made and enforce the changes being recorded to change requests are:
    - Changes and transports for client-specific objects
      - Changes without automatic recording
      - Automatic recording of changes
      - No changes allowed
      - Changes w/o automatic recording, no transports allowed
    - Cross-client object changes
      - Changes to repository and cross-client customizing allowed
      - No changes to cross-client customizing objects
      - No changes to repository objects
      - No changes to repository and cross-client customizing objects

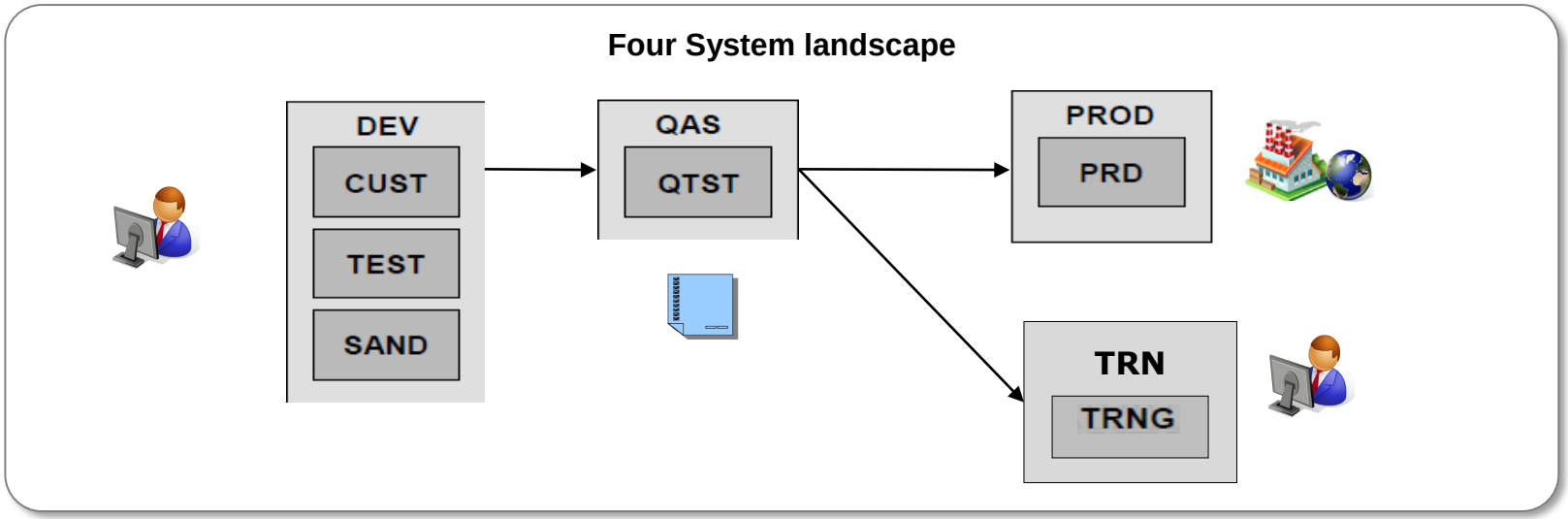
SAP Standard 3 – System Landscape





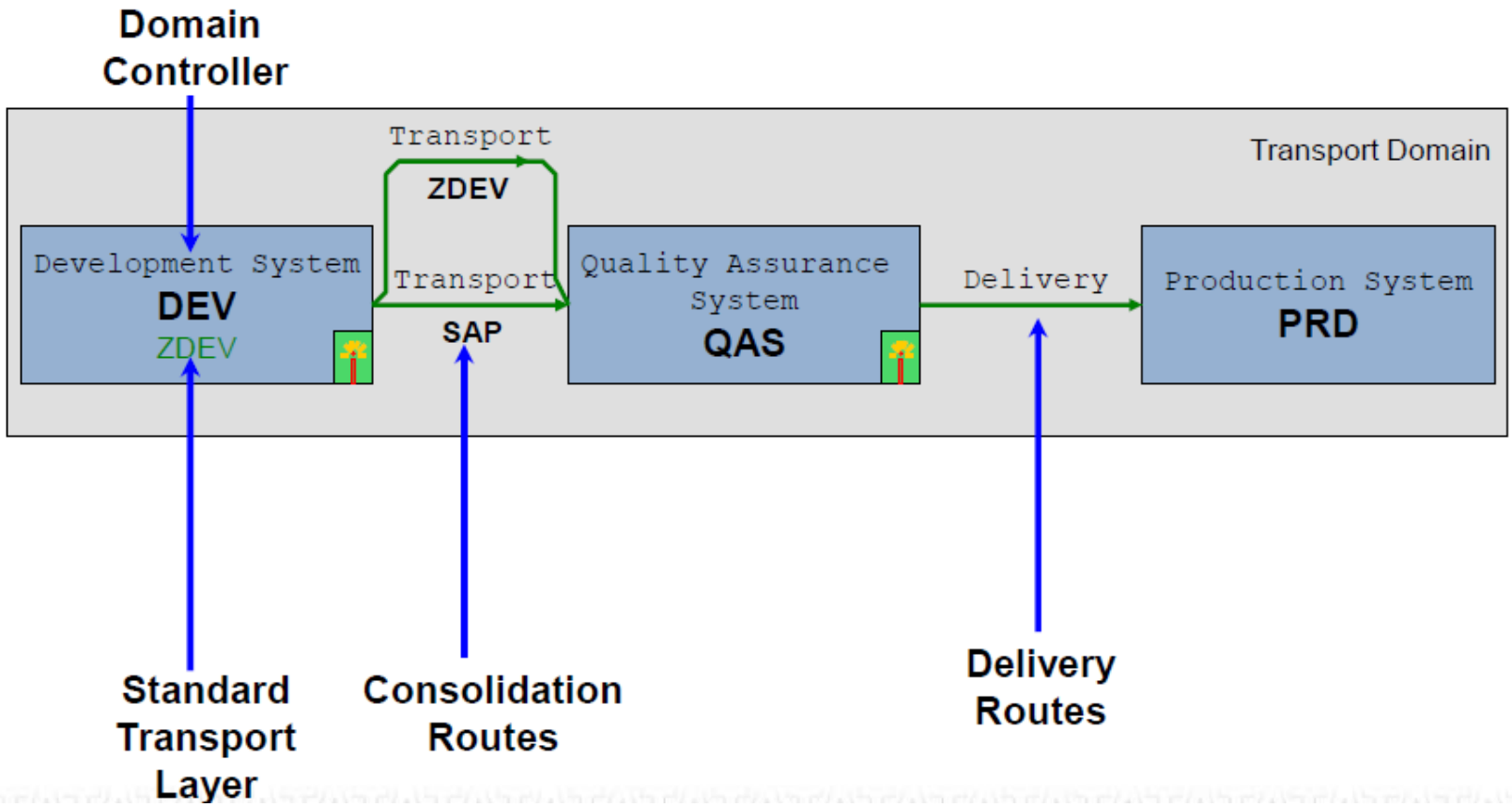
Alternative System Landscapes





Change Transport System - Configuration

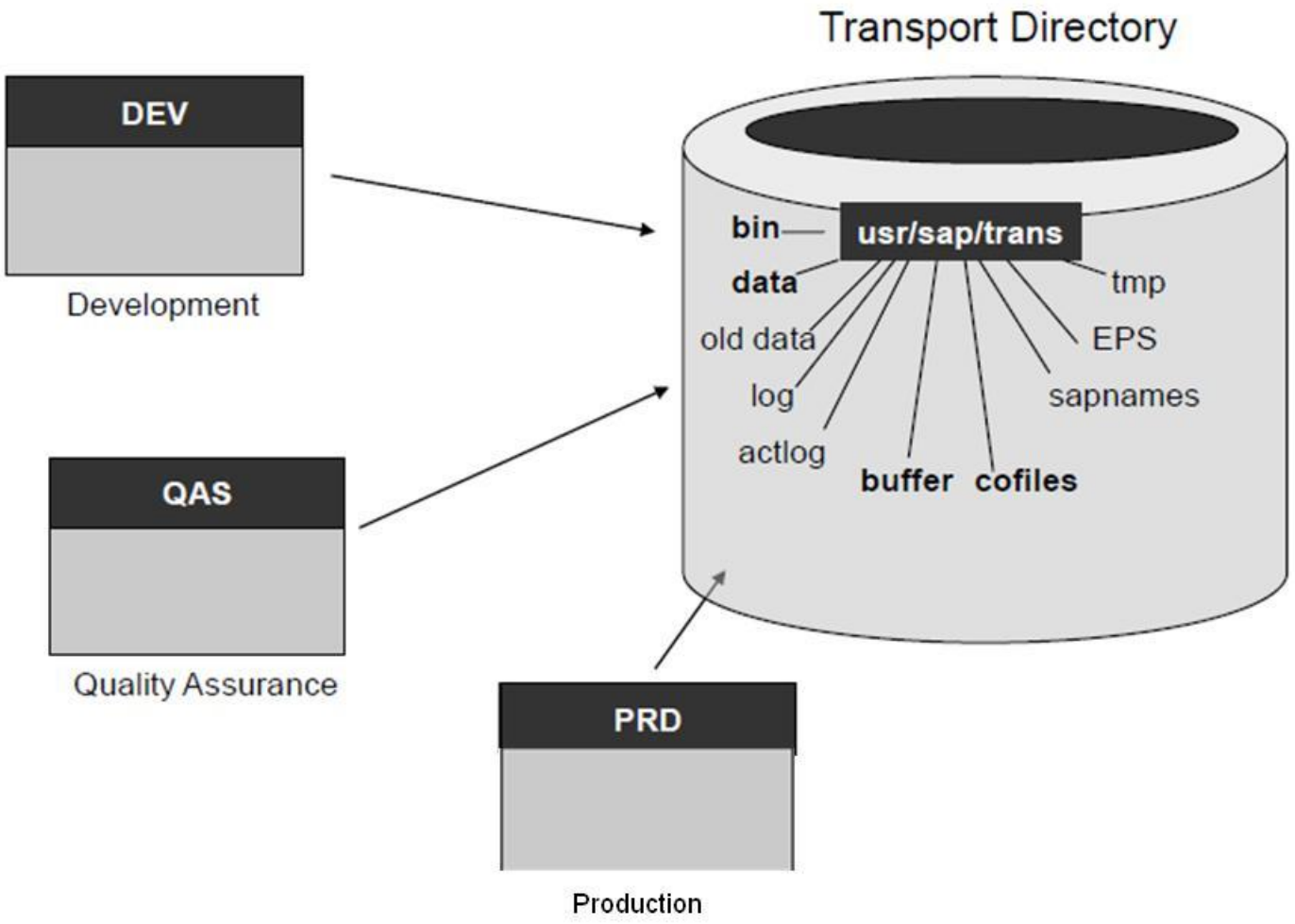
Configuring Transport Routes (STMS)



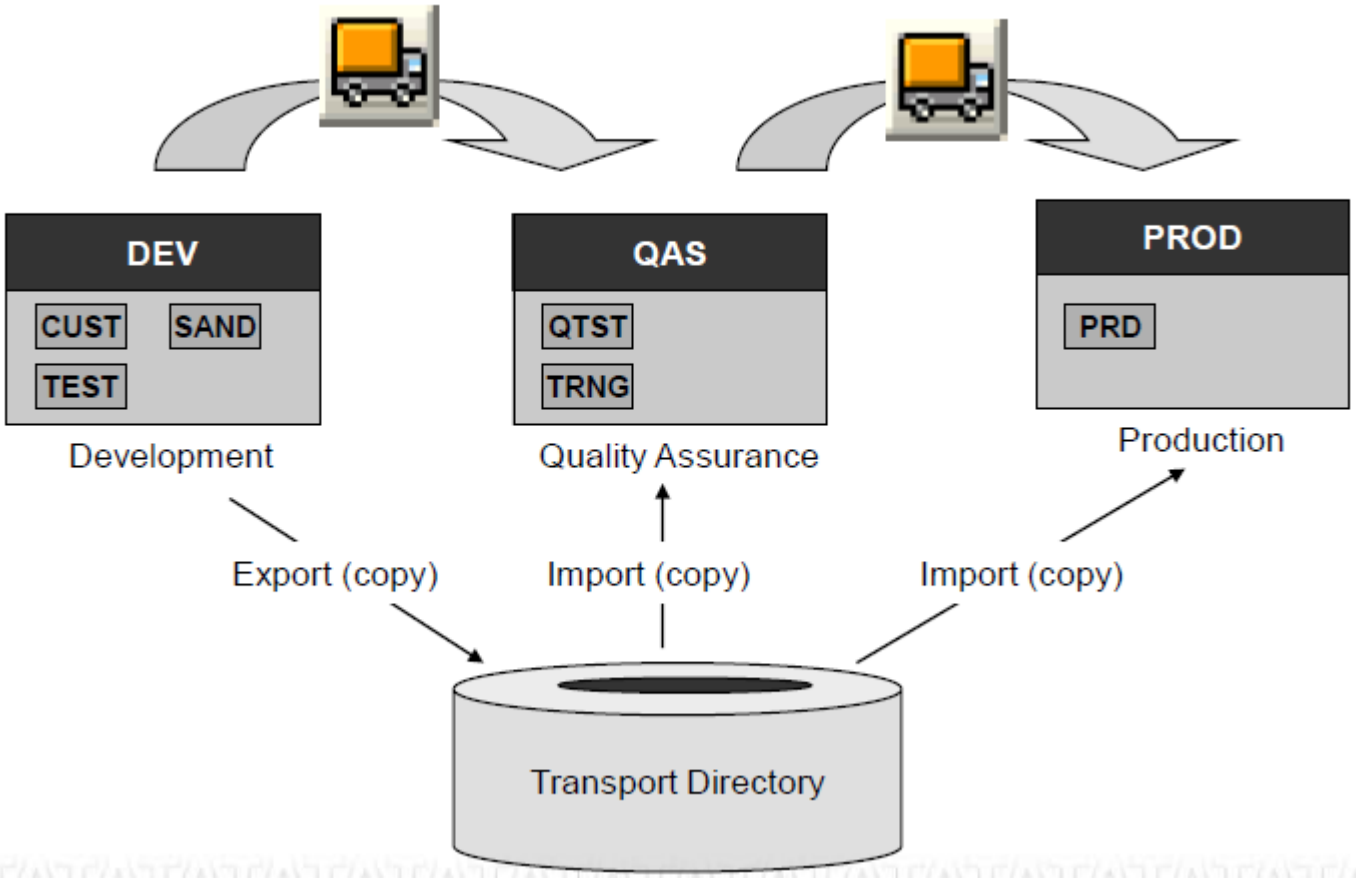
Configuring R/3 Transport

- Make the transport directory available.
- Configure the transport domain controller and define the domain.
- Configuration of the transport program (tp) is done automatically and must not be done at OS level.
- In the TMS:
  - Include all remaining systems in the domain
  - Define the transport routes
  - Define QA approval procedure
- Set the system change options according to the role of the R/3 System.
- Create clients and set the client change options for the production system, development system, and so on.

Transport Directory

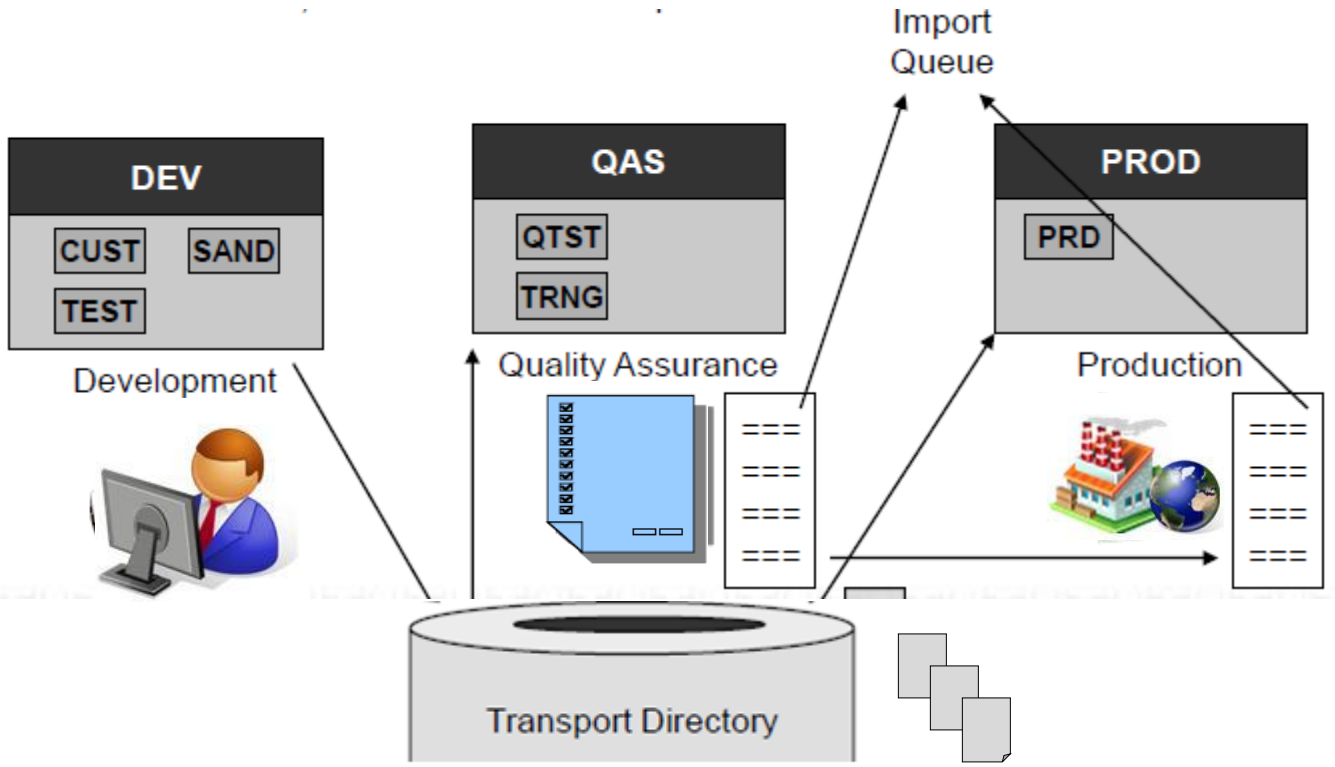


System Landscape Logical and Physical Transport Sequence



Transport Management System

- Exporting Changes
  - Releasing change request
  - Physical copying of recorded object from database to OS file in transport directory
- Importing Changes
  - Import queue of TMS
  - Transaction STMS, choose Overview - Imports



Exercise & Break Out Session

Exercise

Login into the SAP system with the userid/password provided by your instructor

Steps for instructor

- Go to transaction STMS , add a new system called ID1 with the same host name as the training server. This will be a virtual host
- Go to define transport layer , and create a new layer called ZID1
- Create a transport route , ZTR1 from the training server to the ID1 virtual host
- Create a new package called ZPK1 using SE80
- Associate the package with the ZPK1 with ZID1 transport layer
- Explain the relationship between Package – Route – Layer – Object Type

Steps for Trainees

- Go to transaction SE01 , and create a transport of type “Workbench” , and take care to mention the target host ID1
- Add any non-SAP standard ABAP object in the transport
- Save the transport
- Now release the transport
- Note the data & cofile that has been created and their names