Installation of SAP Unifier for R/3

Release 5.0 SP5

Document Version 1.0

Copyright

Icons

Icon	Meaning
	Caution
	Example
	Note
	Recommendation
	Syntax

Typographic Conventions

Type Style	Description
Example text	Words or characters that appear on the screen. These include field names, screen titles, pushbuttons as well as menu names, paths and options.
	Cross-references to other documentation.
Example text	Emphasized words or phrases in body text, titles of graphics and tables.
EXAMPLE TEXT	Names of elements in the system. These include report names, program names, transaction codes, table names, and individual key words of a programming language, when surrounded by body text, for example, SELECT and INCLUDE.
`Example text`	Screen output. This includes file and directory names and their paths, messages, source code, names of variables and parameters as well as names of installation, upgrade and database tools.
`EXAMPLE TEXT`	Keys on the keyboard, for example, function keys (such as `F2`) or the `ENTER` key.
`Example text`	Exact user entry. These are words or characters that you enter in the system exactly as they appear in the documentation.
`<Example text>`	Variable user entry. Pointed brackets indicate that you replace these words and characters with appropriate entries.

Contents

Installation of SAP Unifier for R/3

Purpose

This document details the installation of SAP Unifier for R/3, an application that makes it possible to integrate SAP R/3 systems into SAP Enterprise Portal. As a result, enterprises are able to unify SAP systems with other information sources and integrate business data on a single platform.

Integration

You install SAP Unifier for R/3 on top of SAP Unification Server. After setting the appropriate parameters, you create a project that accesses an R/3 system from the portal. The unifier fields requests launched from the portal, and forwards the requests to R/3 logical systems via the SAP DCOM Component Connector (DCOM CC). It then binds the returned information to the appropriate presentation elements and displays it in the portal.

To integrate an SAP R/3 system into SAP Enterprise Portal through the R/3 Unifier, it is necessary to perform specific configuration tasks that render R/3 data Drag&Relate enabled. Further modifications that extend the Drag&Relate capabilities of the R/3 data in a unifier project can be made in the SAP R/3 system.

See:

Pre-installation Activities

Purpose

SAP Unifier for R/3 runs on SAP Unification Server. Before installing the unifier on Unification Server, you need to install other components, perform certain related activities, and configure specific settings on your system.

Process Flow

1. Install SAP Unification Server 5.0, SP5 Patch 3. *Installing SAP Unification Server 5.0* is located in *SAP Enterprise Portal 6.0 Installation Guide* at
   http://service.sap.com/ep60 → *Documentation & More → Installation.*

   o In order to support SAP Unifier for R/3, your SAP Unification Server configuration should have Windows 2000, SQL Server 2000 and Microsoft FrontPage 2000 server extensions.

   o It is possible to install SAP Unifier for R/3 on a machine that is running SQL Server 2000 with SP1 and SP2.

2. Download and install the Microsoft SQL Patch for dealing with access violations that result from using distributed queries. The patch is available at:

   ftp://ftp.microsoft.com/bussys/sql/transfer/sql80/sqlservr/356450_ENU_i386.exe

   Proceed as follows:

   o Copy the above self-extracting archive file and execute the program

   o When prompted for a password, enter **co-BNuLPO**

   o The extracted **README.TXT** file provides all remaining instructions

3. Install the SAP DCOM Component Connector (DCOM CC) from the Server Components CD (Compilation 4.6D or higher), on the machine on which you intend to generate the unifier project. DCOM Connectors can also be downloaded from *http://service.sap.com/connectors → SAP DCOM Connector.*

   o The SAP DCOM CC serves as the connection layer between SAP Unifier for R/3 and the SAP R/3 system data. It is also necessary for the full support of Single Sign-On 2 (SAP Logon Tickets).

   o During project generation, a linked server and a DCOM destination are created on all the relevant machines: the Unifier machine on which you are running the Project Generator and the SQL server machine.

   o There must be an SAP DCOM CC installed on the R/3 OLE DB Provider machine, as well, if the OLE DB Provider is not on the same machine as the SAP Unifier.

4. Change the activation mode of the SAP DCOM CC to support connection pooling. See [Changing the Activation Mode of the SAP DCOM Connector [Page 8]](#)

5. Install the OLE DB Provider for SAP R/3 on the SQL server machine.

6. Install Microsoft Internet Explorer 5.01 with SP2, or Microsoft Internet Explorer 5.5 with SP2, or Netscape Navigator. **Note that support for Netscape is possible through Java GUI only**.

7. Install the latest version of ITS from the Server Components CD (Compilation 4 or later), to support R/3 access via WEBGUI.

8. Install Microsoft SQL Server 2000.

   During installation, when prompted to select the preferred SQL authentication method, you must choose *Mixed Authentication*.

Distributed System

If you are planning to work on a distributed SQL Server system, install:

- Microsoft Data Access Component (MDAC) 2.6 on the remote SQL Server machine.

- DCOM CC on the SQL Server machine

- OLE DB Provider for SAP R/3 on the remote SQL Server machine

For more information, see .

Result

You are now ready to install SAP Unifier for R/3.

See:

Changing the Activation Mode of the SAP DCOM Connector

Use

To enable SAP Unifier for R/3 to work with no user logged on to the server machine, change the activation mode of the installed SAP DCOM Component Connector (DCOM CC).

Installing SAP DCOM CC automatically installs the package *SAP R/3 DCOM Connector*. By default the activation mode of the package is *Server Application*.

Procedure

To change the SAP DCOM Connector activation mode:

1. From *Windows Start → Programs → Administrative Tools*, select *Component Services* to launch the *Component Services Manager*.

2. Expand *Computers → My Computer → COM+Applications*.

3. Right-click *SAP R/3 DCOM Connector* and select *Properties*.

4. In the displayed *DCOM Connector Properties* screen, select the *Activation* tab.

5. In the *Activation type* frame, select *Library Application* and choose *OK*. If a warning dialog is displayed, choose *OK*.

Result

The SAP R/3 Unifier can now run in Log Off mode.

Support for Distributed SQL (Optional)

The information in this section is relevant if you are planning to run SAP Unifier for R/3 in a distributed SQL server system.

Pre-installation Considerations

You must ensure the following before installing SAP Unifier for R/3:

- The SQL Server machine must have **MDAC 2.6**. This can be installed from: *root folder\Accessories\ODBC Installer\mdac_type.exe*.

- SAP DCOM Component Connector is installed on the SQL server machine.

- The SAP Unifier for R/3 machine must meet all the requirements as specified in the *Prerequisites* section of <u>Running the SAP Unifier for R/3 Setup Wizard [Page 10]</u>.

- When configuring a distributed system, you must run *Setup.exe* from the SAP Unifier for R/3 installation folder\OLEDB Provider Setup, to install the OLE DB Provider for SAP on the SQL Server machine.

Running the SAP Unifier for R/3 Setup Wizard

Prerequisites

You have installed and configured the required supporting software, as described in *Pre-installation Activities*.

Procedure

To install the SAP Unifier for R/3:

1. Insert CD-ROM No.1 into your machine drive.

    The Welcome screen should display automatically. If it does not, from the CD-ROM root folder, double-click the *Start.hta* file, and then click the *SAP Unifier* link to launch the setup wizard.

2. In the *Welcome* screen, choose *Next* to start installing the unifier.

3. After verifying the current selected settings, choose *Next* in the *Start Copying Files* screen to begin installing the program files.

    A *Setup Status* screen appears indicating which files are being installed.

4. After the wizard has finished copying the files, and if it detects an SQL server on the unifier machine, it offers also to install the OLE DB Provider on the same machine.

    o   If you choose to do so, you can now install the OLE DB Provider.

    o   If you choose to install the OLE DB Provider elsewhere, the installation package contains a separate installer just for the OLE DB Provider, enabling you to install it on any other machine that has an SQL server installation.

    o   See the *DCOM CC* entry in Pre-installation Activities [Page 6] for related information.

    o   Choose the appropriate button.

5. Select *Finish* to complete the installation of SAP Unifier for R/3.

Result

The wizard installs an SAP Project Generator icon on the Unifier Management Console, allowing you to create an SAP Unifier for R/3 project.

Integration of SAP Unifier for R/3 into SAP Enterprise Portal

Purpose

SAP Unifier for R/3 can run as a standalone application, allowing you to create any number of projects that retrieve SAP R/3 system information. However, if you want to access your unifier projects through SAP Enterprise Portal, and Drag&Relate the data of various projects, you need to integrate the unifier into the portal.

The portal provides a single gateway to all unifier projects based on different information sources. This section explains how to integrate SAP Unifier for R/3 into SAP Enterprise Portal to enable users to work with SAP business objects from their portal.

Integration

Integrating SAP Unifier for R/3 entails, among other operations, configuring both the portal system and the SAP R/3 system to enable Drag&Relate. For information,see Unifier Integration Workflow [Page 12].

Unifier Integration Workflow

Purpose

You need to properly integrate the SAP Unifier for R/3 into SAP Enterprise Portal to enable an R/3 Unifier project to run successfully and retrieve R/3 data that is Drag&Relate enabled.

Prerequisites

You have installed an SAP Unifier for R/3 on Unification Server.

Process Flow

- Install either SAP Enterprise Portal 5.0 SP4 Patch 2 or higher, or SAP Enterprise Portal 6.0

    The portal does not need to be installed on the same machine as Unification Server. If you choose to install the portal on a different machine, read *Establishing Trust* in Configuring SAP R/3 System Settings for Drag&Relate [Page 19].

    For more information, consult the documentation of the appropriate portal version:

    o   EP 5.0: *SAP Enterprise Portal Installation Guide* at
        http://service.sap.com/epinstall→ *EP 5.0 SPx Roadmap*→ *Installation, Upgrades, and Patches*

    o   EP 6.0: *SAP Enterprise Portal Installation Guide* at
        http://service.sap.com/ep60 → *Documentation & More* → *Installation*.

- If the R/3 server is on a different domain than the portal and the Unification Server, see SAP note 442401.

- Install the SAP Enterprise Portal Plug-in for the R/3 component system

- Configure the portal system for integration of the R/3 Unifier into SAP Enterprise Portal and for Drag&Relate.

    In the portal that lists the organization's available SAP systems and their properties, do one of the following, depending on your portal release.

    o   **Portal 5.0**: Configure the *systems.xml* file

        For information, see Configuring Portal Settings [Page 14].

        Generic details about configuring the *systems.xml* file are available at:
        http://help.sap.com → SAP NetWeaver → SAP Enterprise Portal → Administration Guide → Integration of Applications and Data Sources → Integration of SAP Systems → System Landscape → Defining Component Systems.

    o   **Portal 6.0**: Define the unifier as a portal system in the Portal Content Directory (PCD)

        For information about the PCD see: *http://help.sap.com → SAP NetWeaver → SAP Enterprise Portal → Administration Guide → Integration of Applications and Data Sources → Integration of SAP Systems → System Landscape*.

    You also map users who are not SAP users.

- Configure the SAP R/3 system for Drag&Relate functionality of R/3 data through the R/3 Unifier. This entails:

- o Configuring the R/3 installation to which you are connecting. See Configuring SAP R/3 System Settings for Drag&Relate [Page 19].

- o Configuring the SAP Unifier for R/3 installation. See Drag&Relate with SAP R/3 Systems [Page 21].

- Configure the ITS if you require access to an SAP R/3 system in WEBGUI mode (for Java). It is possible to run either Java or Windows versions of R/3 iViews; see Configuring the Internet Transaction Server [Page 20].

- Configure the Business Object Repository (BOR) in order to enable generating an extended R/3 Unifier project; see Drag&Relate and the Business Object Repository [Page 34] for details.

Result

The end user will have various SAP business objects in the portal iPanel that can be launched to retrieve data from R/3 databases.

Configuring Portal Settings

Use

This section is relevant only if you are working on an SAP Enterprise Portal 5.0 installation.

Systems.xml is the Portal Content Directory (PCD) file that contains the defined attributes of external or back-end systems, such as SAP R/3 or SAP BW, enabling connection to those systems through the portal. The *systems.xml* file stores all the attributes of the SAP R/3 system which you want to access through the SAP Unifier for R/3. This file must be configured before you create an SAP Unifier for R/3 project.

Procedure

The following are the main sections of a portal *systems.xml* file that are relevant for defining an SAP R/3 system. Configure them as described in the linked example.

1. Configure the System section.

2. Configure the Attributes section.

3. Configure the Web Access Points section.

4. Configure any optional sections.

Example

Configuring the Portal Systems.xml File [Page 15]

Configuring the Portal Systems.xml File

This section is relevant only if you are working on an SAP Enterprise Portal 5.0 installation.

Below is an example of portal settings that need to be configured before generating an SAP Unifier for R/3 project.

Structure

The following are the main sections of a *systems.xml* file that are relevant for defining an SAP R/3 system:

- **System**. This section names and describes the SAP R/3 system to which you are connecting.

    o **The *<name>* parameter must be the logical system name as defined in the R/3 system** itself. The logical system is specified in **Table T000** in the SAP R/3 system.

    o The *<TitleText>* parameter is the user-friendly name.

    o The *<Description>* parameters include such information as whether there is multi-language support, and the description texts of the two main languages, English and German.

- **Attributes**. This section defines the connection to the SAP R/3 system. Four types of connection are possible: through the Application Server, with or without a connection string, and through the Message Server, with or without a connection string.

    o A version of each type of connection is provided in the example below.

    o Notice the group of attributes that must be maintained for each version.

    o The R/3 name and language are optional attributes.

    o If you choose to use the connection string option, you do not need the *<ServerPort>* and *<Group>* attributes.

    o The *<value>* of the *<SystemType>* attribute must be defined exactly as written below.

    o The system host value obviates the need for a router.

- **Web Access Points**. This section defines the ITS and the Drag&Relate server (DRS).

    o As in the System section, there are user-friendly name attributes, as well as various description attributes.

    o Other attributes deal with server protocol and the server address.

- There may also be optional sections dealing with issues such as user mapping, and so on.

Example

System：

```
<System name="SP2CLNT800">
   <Title multilingual="true">

      <pcd:TitleText language="DE"> SP2CLNT800</pcd:TitleText>
      <pcd:TitleText language="EN"> SP2CLNT800</pcd:TitleText>
   </Title>

   <Description multilingual="true">

         <pcd:DescriptionText language="de">
      SP2CLNT800</pcd:DescriptionText>
```

```
        <pcd:DescriptionText language="en">
   SP2CLNT800</pcd:DescriptionText>
</Description>
<Accessibility value="true"/>
```

Attributes:

Notice the group of attributes that must be maintained for each version. The *<value>* of the *<SystemType>* attribute must be defined exactly as written below.

Version 1A: Application server, without a connection string. Necessary attributes are *<ApplicationServer>* and *<SystemNumber>*

```
<Attributes>

   <pcd:Attribute name="Type" value="3"/>
   <pcd:Attribute name="ApplicationServer" value="pwdf0647.sap-
   ag.de"/>
```

In the example shown here, the system number is defined by the *<SystemNumber>* attribute and has a value of **00**.

```
   <pcd:Attribute name="Group" value=""/>
   <pcd:Attribute name="Lang" value="EN"/>
   <pcd:Attribute name="Client" value="800"/>
   <pcd:Attribute name="SystemNumber" value="00"/>
   <pcd:Attribute name="R3Name" value="SP2"/>
   <pcd:Attribute name="SystemType" value="SAP_R3"/>
```

Version 1B: Message server, without a connection string. Necessary attributes are *<MessageServer>*, *<ServerPort>* and *<Group>*.

```
<Attributes>

   <pcd:Attribute name="Type" value="3"/>
   <pcd:Attribute name="MessageServer" value="pwdf0647.sap-
   ag.de"/>
```

The *<ServerPort>* value does not need for the last two digits to be the system number.

```
   <pcd:Attribute name="ServerPort" value="3200"/>
   <pcd:Attribute name="Lang" value="EN"/>
   <pcd:Attribute name="R3Name" value="SP2"/>
   <pcd:Attribute name="Client" value="800"/>
   <pcd:Attribute name="Group" value="SPACE"/>
   <pcd:Attribute name="SystemType" value="SAP_R3"/>
```

Version 2A: Connection string for Application server. Necessary attribute is *<ConnectionString>*. In this example of the connection string, the /**H**/ represents the host server and the /**S**/ identifies the server port.

```
    <pcd:Attribute name="Type" value="3"/>
    <pcd:Attribute name="ConnectionString" value="/H/pwdf0647.sap-
    ag.de/S/3200"/>
    <pcd:Attribute name="Lang" value="EN"/>
    <pcd:Attribute name="Client" value="800"/>
    pcd:Attribute name="R3Name" value="SP2"/>
    <pcd:Attribute name="SystemNumber" value="00"/>
    <pcd:Attribute name="SystemType" value="SAP_R3"/>
```

Version 2B: Connection string for Message server. Necessary attribute is
<ConnectionString>. In this example of the connection string, the **/M/** represents the Message
server, the **/S/** identifies the server port, and the **/G/** is the Group attribute.

```
    <Attributes>
        <pcd:Attribute name="Type" value="3" />
        <pcd:Attribute name="ConnectionString" value="/M/pwdf0647.sap-
        ag.de/S/3200/G/SPACE" />
        <pcd:Attribute name="SystemNumber" value="00" />
        <pcd:Attribute name="R3Name" value="SP2"/>
        <pcd:Attribute name="Lang" value="EN" />
        <pcd:Attribute name="Client" value="800" />

        <pcd:Attribute name="SystemType" value="SAP_R3" />
```

The next two attributes enable you to define user mapping to the system to which you are
connecting. They are necessary only if you want to define user mapping through the portal.
The first concerns the entry point to the User Mapping interface of SAP Enterprise Portal. The
second attribute defines a specific user in the SAP R/3 system.

```
        <pcd:Attribute name="UserMappingType" value="admin,user"/>
        <pcd:Attribute name="r3usernamereference" value="1"/>

    </Attributes>
```

WebAccess Points: You must maintain an ITS as a WebAccessPoint.

```
    <WebAccessPoints>

    <WebAccessPoint category="ITS">

        <Title multilingual="true">

            <pcd:TitleText language="de">ITS-
            Webserver</pcd:TitleText>
            <pcd:TitleText language="en"> ITS-
            Webserver</pcd:TitleText>
        </Title>

        <Description multilingual="true">

            <pcd:DescriptionText language="de"> ITS-
            Webserver</pcd:DescriptionText>
            <pcd:DescriptionText language="en"> ITS-
            Webserver/pcd:DescriptionText>

        </Description>
        <Protocol>HTTP</Protocol>
        <HostName>pwdf0647.wdf.sap-ag.de:1080</HostName>
        <Path>/scripts/wgate</Path>
    </WebAccessPoint>


    <WebAccessPoint category="DRS">
```

The Drag&Relate server runs on HTTP or HTTPS protocol.

```
<Protocol>HTTP</Protocol>
```

The following attribute defines the unifier server name and its port. This is the machine on which the unifier project is running.

```
        <HostName>bw-ups.wdf.sap-ag.de:1234</HostName>

    </WebAccessPoint>

  </WebAccessPoints>


  <LogonLanguages>
     <LogonLanguage value="DE"/>
     <LogonLanguage value="EN"/>
  </LogonLanguages>

</System>
```

Configuring SAP R/3 System Settings for Drag&Relate

Use

In order to enable Drag&Relate operations in SAP transactions, you need to access the SAP R/3 component system and prepare the infrastructure for Drag&Relate navigation with R/3 data. This entails modifying the Business Object Repository (BOR) system tables.

You also need to establish trust between the portal and the SAP R/3 system.

Procedure

Modifying the BOR

When you install the appropriate Support Package, you receive a default BOR, with modified system tables and a set of predefined relationships.

- To enable generation of an extended, or customized, SAP Unifier for R/3 project, you need to configure the BOR; see Drag&Relate with SAP R/3 Systems [Page 21] for complete instructions.

- To complete enabling the Drag&Relate functionality:

    a. Configure the ITS; see Configuring the Internet Transaction Server [Page 20]

    b. Import Support Packages

- To ensure that the appropriate support packages that are required for activating HTML link generation have been imported into your component systems, see Setting Up the SAP R/3 System for Drag&Relate [Page 22] for complete information.

Establishing Trust

The portal logon ticket must be trusted in the SAP R/3 system if users are to retrieve R/3 data while working in the portal. A problem arises when SAP Enterprise Portal and Unification Server are installed on separate machines. Resolve the issue as follows:

1. The portal installs with a *verify.der* file, which is the logon ticket to the R/3 system.

    o When working in EP5.0, make sure the R/3 administrator gets the portal parameters defined in the *verify.der* file and that these are uploaded to the SAP back end system.

    o With an EP6.0 installation, it is not necessary to modify the *verify.der* file since its R/3 system parameters are created during the SAPinst installation. It is only possible to change these parameters through the portal UI.

2. Both applications install with a *verify.pse* file. The portal file is the one that needs to establish a trust with the SAP R/3 system. After installing SAP Enterprise Portal and Unification Server, you must go to **C:/Winnt/System32** of the Unification Server machine and manually overwrite the *verify.pse* file with the portal *verify.pse* file. The two files will then be identical, and there will be trust between Unification Server, on which the unifier runs, and the SAP R/3 system.

Configuring the SAP Internet Transaction Server

Use

The SAP Internet Transaction Server (ITS) is an extension of the Microsoft Internet Information Server (IIS), providing a Web interface that enables communication between a portal browser and an SAP R/3 system. The ITS must be running on the IIS machine.

The IIS receives requests from the portal browser, which it passes on to the ITS. The ITS sends the requests to the SAP R/3 system, which returns HTML pages with the relevant data inserted.

In order to support Drag&Relate operations with data originating in SAP logical systems, you first need to add HRNP links to the relevant SAP database properties. This is an ITS configuration called **Link Generation**. For more information, see Activating Link Generation for the SAP GUI for HTML [Page 25].

Drag&Relate with SAP R/3 Systems

The user of the Enterprise Portal can see all the portal contents, irrespective of their type and origin. Drag&Relate technology links information from different sources.
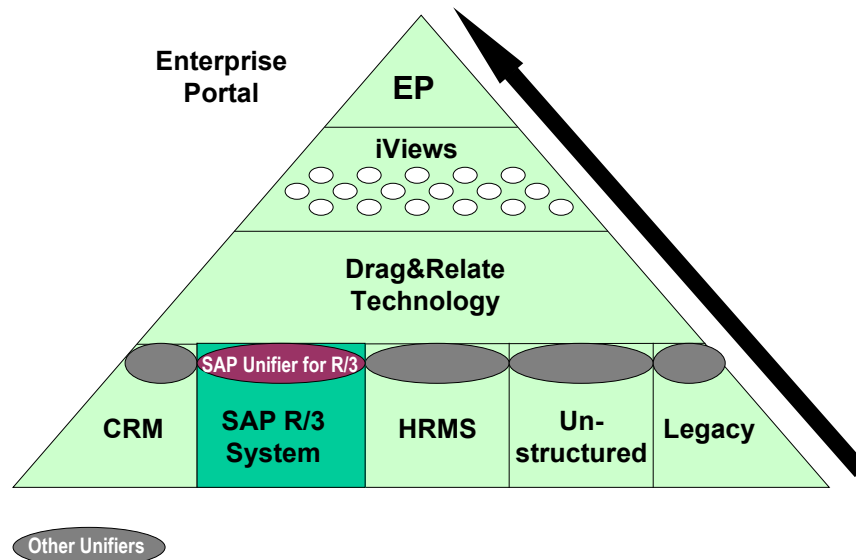
The SAP Unifier for R/3 is a product that permits portal users to see the contents of an SAP R/3 System and to link them to contents in the Web or in other systems in the portal using Drag&Relate.
The contents of external applications are displayed in the form of iViews. The portal displays the information for SAP R/3 transactions in the format of 'External Services'. These are iViews that are displayed on the entire screen.

For more information about Drag&Relate in the Enterprise see The Unification Environment [Ext.].

To implement this scenario you must set up the SAP R/3 System for Drag&Relate [Page 22].

Setting Up the SAP R/3 System for Drag&Relate

Purpose

By setting up the Drag&Relate navigation mechanism you can link the data of the SAP R/3 System with the data of another application in the portal. You can set up this link

- from a non-SAP R/3 System to an SAP R/3 System

- from an SAP R/3 System to a non-SAP R/3 System

- from an SAP R/3 System to a Web component

- from an SAP R/3 System to another SAP R/3 System

- within an SAP R/3 System

Prerequisites

You must satisfy the following requirements:

These requirements must be satisfied for **each** logical SAP R/3 System.

- The SAP R/3 System is registered in the system landscape.
  For more information, see SAP Enterprise Portal Administration Guide, section *Portal Platform → System Administration → System Landscape → System Landscape Editor → Creating a System.*

- The SAP R/3 System has at least Release 4.0B and a Support Package level containing the necessary corrections to the SAP Basis system for using the Drag&Relate functions.
  The following table lists the Support Package levels and their corresponding SAP Basis release (see also Note 211005):

Support Package Level and Corresponding SAP Basis Release for Drag&Relate Functions

SAP Basis Release	Support Package Level
4.0B	42
4.5B	22
4.6A	14
4.6B	06
4.6C	01
as of 4.6D	in the standard system

- The SAP Enterprise Portal plug-in was imported into the SAP R/3 System.
  For more information read Note 0440447.

- The SAP R/3 System is configured for single sign-on with logon tickets.
  For more information see internet address `service.sap.com/securityguide` →
  SAP Enterprise Portal Security Guide → Single Sign On with SAP Logon Tickets.

- The SAP DCOM Component Connector was installed and configured.
  For more information see Running the SAP Unifier for the R/3 Setup Wizard [Page 10].

Process Flow

The SAP R/3 System is configured in a number of steps:

These configuration steps are required for **each** logical SAP R/3 System.

1. You import the Drag&Relate metadata.
   For more information, see Importing Drag&Relate Metadata [Page 24].

   You only have to perform this step for SAP R/3 Systems that contain software component SAP_APPL.

2. You activate link generation for the SAP GUI for HTML
   For more information see Activating Link Generation for the SAP GUI for HTML [Page 25].

3. You assign the users roles. You can create new roles or edit existing roles.
   For more information see Assignment of Users to Roles [Page 26].

Optional (for advanced users only):

4. You define relationships for Drag&Relate. This means that you edit the metadata for the objects in the Business Object Repository (BOR).

   You can only perform this step if:

   you want to continue editing the metadata imported in step 1

   you want to define Drag&Relate metadata manually
   (for example in SAP Basis systems for which SAP does not deliver metadata).

   For more information, see Drag&Relate and the Business Object Repository [Page 34].

Only for SAP R/3 Systems with Release 4.x and higher:

5. You define that system messages should only be displayed each day the first time the user logs on.
   For more information see Configuring System Messages [Page 27].

Result

You set up the SAP R/3 System for Drag&Relate.

Importing Drag&Relate Metadata

Use

Importing the SAP Enterprise Portal plug-in into the SAP R/3 System does not mean that there will be any contents. The tables for the Drag&Relate metadata were created, but not filled with contents. This only happens when the metadata is imported.

Prerequisites

- The SAP Enterprise Portal plug-in was imported into the SAP R/3 System (see Note 440447).

- Software component SAP_APPL exists in the SAP R/3 System.

    You can have a look at the installed components in the detailed display of the system status or check the entries in table CVERS (transaction SE16).

- The metadata must be imported with an up-to-date version of SPAM / SAINT. If necessary, update SPAM.

Procedure

    Also read Notes 335373 and 335369.

1. Choose *Software Distribution Center* on the home page of the SAP Service Marketplace
   (or http://service.sap.com/swcenter).

2. Choose Download → SAP Installations & Upgrades → SAP ENTERPRISE PORTAL → Drag & Relate Meta Data → SAP_WPDRMD.

   The list of available packages with Drag&Relate metadata is displayed. There is a separate package for each Basis release.

    SAP does not currently deliver metadata for Basis releases that are not listed. If you want to implement Drag&Relate scenarios in such systems, you have to define the metadata manually.
    For more information, see Drag&Relate and the Business Object Repository [Page 34].

Result

You imported the Drag&Relate metadata.

    After the import you have to adjust the table structures (see Adjusting Table Structures [Page 56]).

    You can protect existing and modified metadata against overwriting by importing the metadata again. In this case you have to activate the objects manually or use the *Mass Activation* function. For more information, see Drag&Relate and the Business Object Repository [Page 34].

Activating Link Generation for the SAP GUI for HTML

Use

You have to activate HTML link generation in the SAP R/3 System for the hyperrelational navigation mechanism with Drag&Relate.

Prerequisite

To activate HTML link generation you need authorization to change profiles in the SAP R/3 System.

Procedure

1. In the SAP R/3 System choose transaction *RZ10*.

2. Select the instance profile in field *Profile*.

3. In the *Edit profile* area select *Extended maintenance* and choose *Change*.

4. Choose *Parameters → Create*.

5. Enter the following data:

   Parameter name: `wp/pi/enable_drag_and_relate`

   Parameter val.: `1`   (alternatively `yes` or `true`)

   Comment: `Parameter for activating HTML link generation for Drag&Relate`

   > The parameter must be active. If the status is active but the value of the parameter is `0`, `no` or `false`, no HTML links are generated.

6. Choose Enter.

7. Repeat steps 2 to 6 for each instance of the SAP R/3 System.

8. Restart the SAP R/3 System so that the changes can take effect.

Assignment of Users to Roles

Integration

An SAP Unifier for R/3 project that generates a bar in the iPanel is created for each SAP R/3 System registered in the portal. The bar contains a tree structure with branches. Each individual branch represents the role of the user in the SAP R/3 System with its corresponding transactions. If more than one branch appears in the tree structure, the user was assigned more than one role in the SAP R/3 System.

The user can use the Drag&Relate navigation mechanism to click on a field in the content area and execute it with parameters simply by pulling it to a transaction in the iPanel.

Activities

If the bar for an SAP Unifier for R/3 Unifier project is empty in the iPanel, the user must be assigned one or more roles in the SAP R/3 System. There are two possibilities here:

- The user roles in the portal are migrated to the SAP R/3 System.

- The user roles are created or maintained directly in the SAP R/3 System.

Creating and Maintaining User Roles in the SAP R/3 System

To create a new role:

1. Choose Transaction *PFCG*.

2. Enter a name and description for the new role and choose *Create*.

3. Choose the *Menu* tab and then *Transaction*.

4. Enter the transaction codes and choose *Assign transactions*.

5. Choose the *Authorizations* tab and then *Change authorization data.*

6. Maintain the authorizations needed for the role.

7. Choose the *User* tab and assign a user to the role*.*

8. Choose *User compare* and then *Complete compare.*

> The role assigned to the portal user must contain at least the following authorizations in order to have access to the SAP R/3 System from the portal and in order to be able to execute Drag&Relate operations:
>
> Authorization object *S_RFC* with the following field values:
> *ACTVT*: `16`
> *RFC_TYPE*: `FUGR`
> *RFC_NAME*: `RFC1, RSAN, SDIF, SDIFRUNTIME, SDWZ, SKBW, SPR4, SPRT, SRFC, SSCV, SURL, SUSO, SUSW, SU_USER, SWOR, SYST, SYSU`
>
> Authorization object *S_TCODE* with the following field value:
> *TCD*: `SPO1`

You can find detailed information about the individual steps in SAP R/3 Enterprise Documentation, section *mySAP Technology Components → SAP Web Application Server → Security → Users and Roles (BC-SEC-USR).*

Configuring System Messages

Purpose

If an SAP R/3 System with Release 4.x is integrated in the system landscape of SAP Enterprise Portal, the user is logged on again each time he or she accesses the SAP R/3 System for technical reasons. With single sign-on with logon tickets, the portal user is authenticated in the backend system without a further logon, but the SAP R/3 System displays the window with the system messages each time.

Below you can find information about how to configure the SAP R/3 System so that the system messages are only displayed once per calendar day.

> No configuration steps are needed in SAP R/3 Systems with Release 6.10 or higher.

Process Flow

> See also Note 0301563.

You must perform the following steps to configure your system:

1. Create and activate database table TEMSGU.

2. Create an index for table TEMSGU (optional).

   > This is recommended for larger systems with a number of users as it reduces the access times to table entries.

3. Change and activate the report and the function modules

   a. LSM02DEF

   b. SM02_DELOLD_MESSAGE

   c. SM02_GET_UNREAD_MESSAGE

   d. SM02_SET_LAST_USREMSG_ID

Result

Once the changes have been activated, the system messages are only displayed once per day.

Creating Database Table TEMSGU

Procedure

1. Start transaction SE11.

2. Enter **TEMSGU** in field *Database table* and choose *Create*.

3. Enter **Read system messages of a user** in field *Short description*.

4. Choose the tab *Delivery and Maintenance*.

    a. Enter an **L** in field *Delivery Class*.

    b. In field *Data Browser/Table View Maintenance* enter **Display/Maintenance Not Allowed**.

5. Choose tab *Fields*.

    Enter the following 3 fields:

    o **BNAME**
      Data element = **EMEAUTHOR**
      Select *Key* and *Initial value*

    o **BDATE**
      Data element = **EMEDATCRE**
      Select *Initial value*

    o **BID**
      Data element = **EMEID**

6. Select *Technical settings*.

    a. In the area *Logical storage parameters*, enter APPL1 in field *Data Type* and 0 in field *Size Category*.

    b. In area *Buffering,* choose the option *Buffering not allowed.*

7. Choose *Save*.

    Enter **STSK** as package or development class and create a transport request for your SAP System.

8. Activate table TEMSGU.

    In transaction SE16, check if table TEMSGU was created correctly.

Creating an Index for Database Table TEMSGU

Use

In large SAP R/3 Systems with many users, there should be an index on field BDATE of table TEMSGU. This reduces the access time for the table.

Procedure

1. Start transaction SE11.

2. Enter `TEMSGU` in field *Database Table* and choose *Change*.

3. Choose *Indexes*.

4. Enter `DAT` in field *Index name*.

5. Enter `Index of Read System Messages for a User` in field *Short description*.

6. Select the options *Non-unique index* and *Index on all database systems*.

7. Enter `BDATE` in column *Field name* of table *Index flds*.

8. Choose *Save* and then *Activate*.

Changing Report LSM02DEF

Procedure

1. Start transaction *SE38*.

2. Enter **LSM02TOP** in field *Program* and choose *Change*.

3. Add table TEMSGU to the *TABLES:* statement.

```
FUNCTION-POOL SM02.      "MESSAGE-ID

TABLES : TEMSG,TEMSGU,TEMSI,T000,
* start insert langu added
        T002T.
* end insert langu added

INCLUDE LSM02DEF.
INCLUDE TSKHINCL.
```

4. Choose *Save* and then *Activate*.

Changing Function Module SM02_DELOLD_MESSAGE

Procedure

1. Start transaction *SE37*.

2. Enter **SM02_DELOLD_MESSAGE** in field *Function module* and choose *Change*.

3. Enter the following lines at the end of the program code, before the *ENDFUNCTION* statement:

```
* ---------------------------------------------
* Delete your last own ID from yesterday
* ---------------------------------------------

DELETE FROM TEMSGU
  WHERE BNAME = SY-UNAME AND
    BDATE < SY-DATUM

ENDFUNCTION
```

4. Choose *Save* and then *Activate*.

Changing Function Module SM02_GET_UNREAD_MESSAGE

Procedure

1. Start transaction *SE37*.

2. Enter **SM02_GET_UNREAD_MESSAGE** in the *Function module* field and choose *Change*.

3. Enter the following lines after the *IF ENTRY_TYPE = CALL_FROM_DYNP* query:

```
SELECT SINGLE * FROM TEMSGU WHERE BNAME = SY-UNAME
IF SY-SUBRC = 0.
  IF TEMSGU-BDATE = SY-DATUM.
    LAST_USR_EMSG = TEMSGU-BID.
  ENDIF.
ENDIF.

CALL 'C_SAPGPARAM' ID 'NAME'  FIELD 'rdisp/myname'
  ID 'VALUE' FIELD MYNAME.
```

4. Choose *Save* and then *Activate*.

Changing Function Module SM02_SET_LAST_USREMSG_ID

Procedure

1. Start transaction *SE37*.

2. Enter **SM02_SET_LAST_USREMSG_ID** in the *Function module* field and choose *Change*.

3. Enter the following lines at the end of the program code, before the *ENDFUNCTION* statement:

```
* ----------------------------------------------
* Write last message read in the table
* ----------------------------------------------

SELECT SINGLE FOR UPDATE * FROM TEMSGU WHERE BNAME = SY-UNAME.
TEMSGU-BNAME = SY-UNAME.
TEMSGU-BDATE = SY-DATUM.
TEMSGU-BID   = LAST_USR_EMSG.
MODIFY TEMSGU.

ENDFUNCTION
```

4. Choose *Save* and then *Activate*.

Drag&Relate and the Business Object Repository

Use

The context information required for Drag&Relate is maintained as **metadata** for the corresponding business object types in the Business Object Repository (BOR) in the logical SAP R/3 System. This metadata describes the relationship between and business objects and either

- Screen fields

- Possible target transactions

- Other transactions that can be derived from the relationships between business objects.

You can maintain the metadata for Drag&Relate in transaction *SPO0*. You must maintain the Drag&Relate metadata in all the logical system in which the transaction is to be executed with Drag&Relate.

> ⚠
>
> You should only maintain your own BOR objects. If you maintain metadata that was delivered by SAP, you must adjust the objects manually if you will import the metadata again at a later time.

Prerequisites

Before starting to maintain Drag&Relate metadata, you must import the Enterprise Portal plug-in into the SAP R/3 logical system.

If your SAP R/3 logical system is a standard R/3 System, that is, software component SAP_APPL is installed, you should first import the Drag&Relate metadata [Page 24] supplied by SAP through the SAP Service Marketplace.
You can then change the metadata and activate this version of it.

The following table contains an overview of the main tables related to Drag&Relate operations:

Table	Description
SPRTL0	Table in which you maintain Drag&Relate metadata
SPRTL0C	Modified customer version of SPRTL0 used during Drag&Relate operations
SPRTL1	Table containing the object – data element assignments
SPRTL1C	Modified customer version of SPRTL1 used during Drag&Relate operations
SPRTL2	Data element assignment to supertype (active). This runtime table is generated from the data in SPRTL1.

The dependent table *SPRTL2* is not transported. The system generates this table when you activate the metadata. You can activate either the SAP delivery version or the modified customer version (tables ending with C) for tables *SPRTL0* and *SPRTL1*.

> ⚠
>
> Only change objects in the customer tables.

Features

Transaction *SPO0* contains various tools for maintaining metadata. They are all available on the start screen, in the *Tools* area.

You can activate inactive objects or the SAP delivery version to make all saved and inactive metadata available for Drag&Relate operations. There is also a mass activator, in which you can select the exact objects that you want to activate. For more information, see Activating Metadata [Page 55].

You can transport the metadata. This creates a transport request that you can import into other logical systems. This works only in customer modes *Normal* (blank) and *No Automatic Transport* (M), it does not work in the mode *Transport of SAP Tables* (S). In the *Drag&Relate Customizing Check* area you can choose *Settings* to check this.

You can generate a test screen. This calls the display method for an object. If the data element selected is related to a BOR object, you can choose *Enter Key* to enter a value for the object and execute the display method.

You can generate test data derived from the Basis flight data model to test Drag&Relate without affecting the BOR objects of your business transactions.

You can delete the buffer for the ad hoc query to remove inconsistencies in the InfoSets and queries. This is useful if you change InfoSets. For more information, see Ad Hoc Query [Page 54].

Activities

You can maintain metadata for BOR objects, data elements, and Drag&Relate-enabled services. You can switch between these views on the metadata at appropriate points.

> For an SAP R/3 System of Release 4.0B or higher, a Drag&Relate-enabled service is a transaction that has already been assigned to an object. For an SAP R/3 System of Release 4.6C or higher, a service can be a transaction or an ABAP report that has already been assigned to an object. You maintain Drag&Relate metadata from a service perspective for specific data elements that the transaction or report contains.

Specify a relationship between the relevant data element and a BOR object. This relationship is known as a key type. This releases the content of any output fields that use this data element for Drag&Relate. The mouse pointer also changes when the user clicks on an object.

Specify the transactions to which the user can drag the object. The user can see that he or she can drag the object to this particular transaction because the mouse pointer changes.

Derive other target transactions using relations between business objects. Business object types can be linked to other business object types using an **object-to-object relationship** and can then use the target transactions of the other business object as well.

Specify function modules to read additional information that cannot be represented by object-to-object relationships. The SAP Unifier for R/3 can use this information to parameterize URLs that are used as Web components.

Save and activate the metadata that you have entered. Depending on your system settings, a *Transport* dialog box appears so that you can add the changed data to a correction request.

> Once you have saved and activated the metadata, you must restart the SAP Unifier for R/3 and refresh the iPanel in the Enterprise Portal to transfer the metadata changes to the Enterprise Portal.

Maintaining Drag&Relate Metadata in Development Mode

Use

You can use the development mode to display technical information on the fields within a business transaction in the SAP R/3 logical system and maintain Drag&Relate metadata for screen fields to which no BOR object is assigned.

This setting is user-specific and therefore does not apply to all users when set.

Prerequisites

You can use this function in logical SAP R/3 systems of Release 4.0B or higher in which the Enterprise Portal plug-in has been imported.

Setting the Development Mode

1. On the *SAP Logon*, go to the menu in the top left-hand corner of the dialog box and choose *Options*.
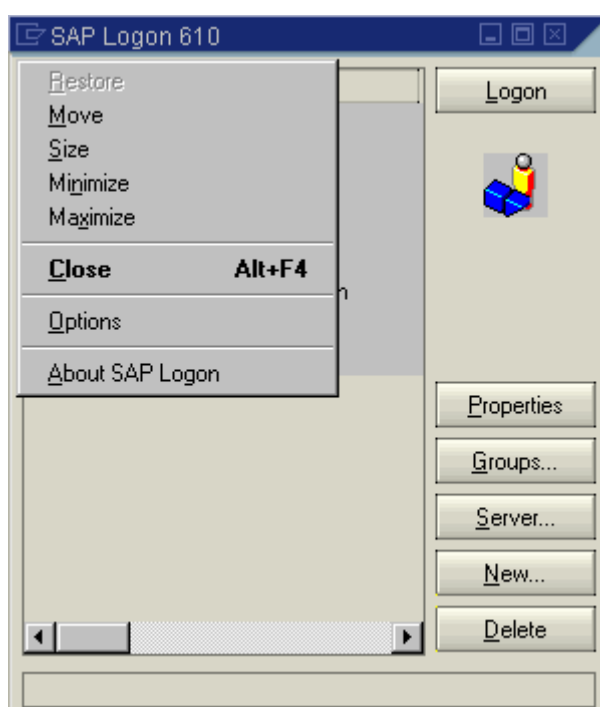
2. In the *Additional command line arguments* field, enter `/WP` .

3. Choose *OK* to save your changes.

4. Log on to the SAP R/3 System and choose transaction *SPO0*.
   You can see that the command line argument in step two has been set through the status *Support Flag Set* in the *Drag&Relate Customizing Check* area. If the support flag has not been set, the status *Support Flag not Set* appears.

5. In the *Drag&Relate Customizing Check* area, choose *Settings*.

6. In the *Settings for Interactive Maintenance of Drag&Relate Data* dialog box, select entry *D* from the possible entries for field *Mode*.

This sets maintenance mode so that links to data elements are highlighted with a vertical blue line in the field.

This mode is set for the duration of the logon session. This means that, the next time you log on to the system and want to use this function, you should check the settings described in steps one to three and you must repeat steps four to six.

Result

You have selected the mode for displaying technical information on the fields in a business transaction in the logical SAP R/3 System. A blue vertical line appears beside fields that have a data element behind them. For more information, see SAP Note 356495.

Using the Development Mode

1. Choose the business transaction for which you want to maintain Drag&Relate metadata.
   You can see which fields have data elements behind them by the blue vertical line on the field.

2. Choose the blue line on the field that you want to maintain.
   The system takes you to the Drag&Relate metadata maintenance transaction for this data element.

3. Maintain the Drag&Relate metadata.

Result

You have maintained Drag&Relate metadata based on the context of the business transaction.

Customizing Check

The customizing check allows you to see whether the prerequisites for configuring and using an SAP R/3 logical system for Drag&Relate have been met.

The system checks the following:

- System Parameter
- Enterprise Portal Plug-In
- Active Data Elements
- SAP GUI Support
- Separate Logical Systems
- ITS for Local URL Generation
- Test Data

The sections below contain more details about the possible options for these customizing settings.

System Parameter

This indicator indicates whether the parameter for Drag&Relate is set.

For more information see Activating Link Generation for the SAP GUI for HTML [Page 25].

Enterprise Portal Plug-In

This indicator shows which plug-in is available in the logical system.

You must have imported the current Enterprise Portal plug-in into your system. The Enterprise Portal plug-in release should match the Enterprise Portal release.
The combination of the current release of the Enterprise Portal and the immediately preceding release of the Enterprise Portal plug-in is also allowed.

Active Data Elements

This indicator shows how many active data elements exist in the logical system.

The screen field and the Drag&Relate object link are connected by data elements. "Active" data elements are data elements that have an assignment to an object type. The system generates object links for all screen fields that reference one of these active data elements.

The Drag&Relate metadata that contains this assignment is not delivered together with the Enterprise Portal plug-in. SAP delivers the data over the SAP Service Marketplace.
For more information, see Importing Drag&Relate Metadata [Page 24].

If there is no metadata in the system or the available objects have not been activated, no "active" data elements are available.
For more information, see Activating Metadata [Page 55].

SAP GUI Support

This indicator shows whether the parameters for link generation in the different flavors of SAP GUI has been set.

SAP GUI for Windows

When using the SAP GUI for Windows, you can activate link generation by setting the command line option /WP, for more information (see Maintaining Drag&Relate Metadata in Development Mode [Page 36]). This option should be set if the SAP GUI for Windows is called from within the Enterprise Portal.
You can also go to this option manually. For more information, see SAP Note 356495.

SAP GUI for HTML

When using the SAP GUI for HTML, the system checks whether the *~navigationenabled* parameter has been set. This parameter should be set automatically for the Enterprise Portal. For more information, see SAP Note 212773.

Separate Logical Systems

This indicator shows the specified name of the logical system as a cross-check for compliance with the naming conventions.

You maintain the logical system in client table T000 in the SAP R/3 logical system. The name of the logical system should match the convention <system name>CLNT<client>, for example, PRDCLNT080 or DEVCLNT000. If you do not stick to this convention, there may be problems with link generation for iViews, because the logical system is not known in the ITS context and is concatenated there in part using the convention of the system name and the client.

ITS for Local URL Generation

This feature is optional, for special cases only.

This indicator shows whether the entry in table *TWPURLSVR* (in Release 4.6C and higher) or in the Drag&Relate customizing (Release 4.0B to 4.6B) has been maintained or not.

The Enterprise Portal Plug-In contains a function module that you can use to locally generate URLs (for example, for Drag&Relate). This functions module requires the information about the ITS on which the system is to generate these URLs.
Choose *Settings* to display the information. Maintain the ITS information in table *TWPURLSVR* in transaction *SM30* or directly in *Settings* (releases earlier than 4.6C)

> ⚠️
>
> As of Basis Release 4.6C, module *CREATE_URL_OBJECT* is used. This accesses the information in table *TWPURLSVR* to determine the ITS. Make sure that this table contains the corresponding entry.

Test Data

This indicator shows whether there is test data in the logical system and whether it is complete.

SAP has created some business objects and metadata based on the Basis flight data model that you can use for testing. The required metadata is generated in transaction SPO0 and the business objects are delivered with the Enterprise Portal plug-in.

There are several business objects in development class *DRSCENARIOS* whose names begin with *DR*, for example, *DRCARR*, *DRSBOOK*, and so on. The system checks whether there is metadata for these objects. If no data is available or data is only available for some of the objects, this is displayed. In this case, you start generating the data.

> 💡
>
> If you change metadata for these objects, you can restore the original data at any time by regenerating the data.

In order to use the test scenarios, the flight data model table must be filled as well. If the tables in your system do not contain any data, you can use the following reports to fill the tables:

- RSBCDAT2 (Basis release 4.0)

- SAPBC_TOOLS_GENERATOR (Basis release 4.5)

- SAPBC_TOOLS_GENERATOR_NEW (Basis release 4.6C)

- SAPBC_DATA_GENERATOR (Basis release 6.10)

For Basis release 4.6B you can get the corresponding development objects in the SAP Knowledge Shop for customers and partners *(http://service.sap.com → Education)* by ordering the "ABAP Training Objects" add-on under Material No. 50039976. Installation of the training objects is described in SAP Note 50039976.

You should use the transactions in development class *DRSCENARIOS* as drop targets for test scenarios. Analog to the business objects, the transaction codes start with *DR*.

Releasing Data Elements for Drag&Relate

Use

To use data elements in Drag&Relate activities, they must be released in the logical system.

Prerequisites

If there are several key fields, the underlying data elements must have a parameter ID so that they can be set automatically (with SET/GET PARAMETER).

If the same data element is used to identify more than one business object type, you must create a Drag&Relate supertype. The Drag&Relate supertype represents a **technical** generalization of business objects that are **semantically** different. For more information see Creating Drag&Relate Supertypes [Page 44].

If you use Drag&Relate with an iView as the drag source in place of a transaction in a logical system, you must generate the object link using method *generateDNRUrl* in service *IUrlGeneratorService* provided by the Enterprise Portal runtime.

See also: Conditions for Releasing Data Elements for Drag&Relate [Page 42]

Procedure

1. Start transaction *SPO0*.

2. To change the definition of an object type, choose the relevant object type in the *Object type* field using the F4 help or by typing the exact name in this field.

3. Choose *Display* to the left of the field.

4. Switch to change mode by choosing *Display/change*. To create a new definition, choose *Create* and enter an object type in the dialog box.

5. Select the relevant key field for the screen, if there is more than one field.

6. To find further data elements (referring to the same domain) for the data element of the relevant key field for the screen, use the functions *Create key* and *Create key: Enhanced*.
   The *Create key* function gives you a list of all the data elements assigned to the same domain as the current key field.
   With the *Create key: Enhanced* function you can select further data elements as keys using the search help.

7. Save your entries.

Result

You have released the data element for use in Drag&Relate actions.

Conditions for Releasing Data Elements for Drag&Relate

An important condition is that only **output** screen fields are drag-enabled.

An instance of a business object is uniquely assigned to each **drag-enabled** output screen field. This is achieved by assigning the data element underlying the output screen field to a key field (the screen-relevant key field) of the business object type. If a business object type has more than one key field, the other keys are supplied with parameter IDs from the transactions.

In a number of cases, fields in different transactions may have different data elements but be used in the same way semantically, in which case they can also be used to identify the business object type. They can be assigned additionally.

The assignments or **context information** are created as *key types*. Each key type contains a complete record with the key fields of the business object and the link information for the data elements and parameter IDs used in the transactions. A field of the key must also be marked as *identifying* in the key type to indicate that there is a relationship between the business object and the data element. This means that all the transactions with screen fields that use this data element and that correspond to the other key type elements are uniquely assigned to this business object type.

Parameter IDs are optional for key type entries marked as having an identifying role (the key value is accessed using the screen field itself). However, they are obligatory for all other key entries so that they can work with SET/GET functions.

Links cannot be generated if the set/get parameters are not set correctly by the transaction.
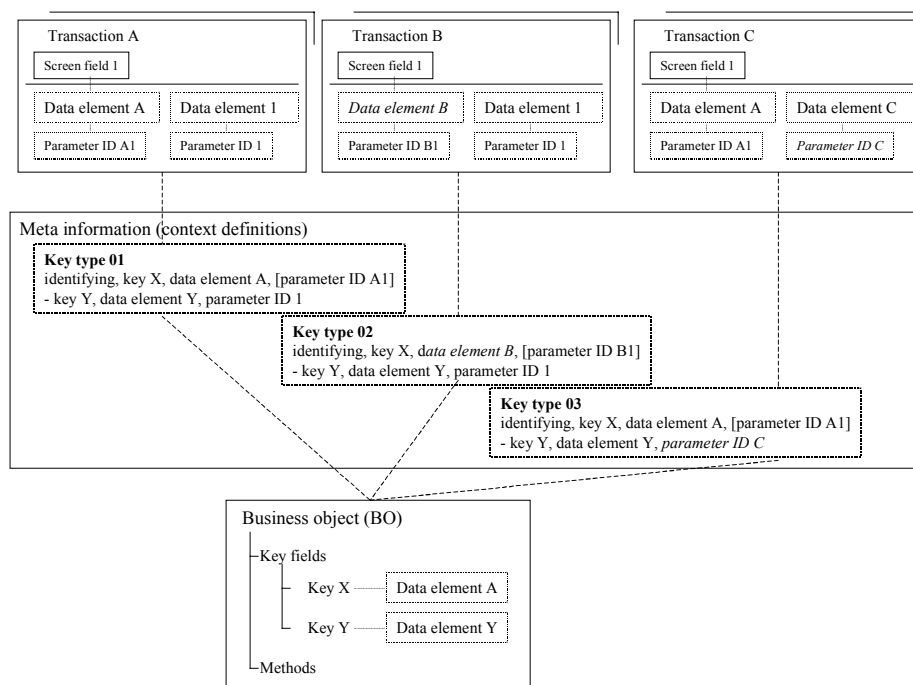
Font size needs to b increased to 16 –18 pt at least

The above graphic shows the connection between the screen field of the source transaction and the business object using key types.

In sample transaction A, key X (with data element A) was marked as identifying in key type 01 since data element A is used by screen field 1 of transaction A. Screen field 1 thus represents the business object and is drag-enabled. The key values are supplied by the screen field and the parameter ID 1. All transactions that support the same key type also become drag-enabled.

It is necessary to specify additional key types if other screen fields that do not support the existing key types need to be drag-enabled. This is the case if data elements or parameter IDs have different names but are used in the same way semantically, as shown in transactions B and C in the diagram.

Creating Drag&Relate Supertypes

Use

In some rare cases, the same data element is used to identify more than one business object type.

As it is no longer possible to uniquely reference a business object type from the data element; these generic data elements need to be assigned to a Drag&Relate supertype, which you must create separately. The Drag&Relate supertype is a **technical** generalization of business objects that are **semantically** different.

A function module that is assigned to the Drag&Relate supertype is used to determine the actual business object type in a certain instance. The actual business object type is then determined by calling this function module when the object is dropped.

> For example, the *Order number* (AUFNR) data element identifies both *internal orders* and *production orders*. A Drag&Relate supertype with the same name as the data element (_AUFNR) is therefore defined in transaction *SPO0.* This Drag&Relate supertype represents both this business object type and others.

You use this procedure to create Drag&Relate supertypes to allow a distinction between the different uses of a data element.

Procedure

1. Start transaction *SPO0.*

2. To change the definition of an object type, choose the relevant object type in the *Object type* field using the F4 help or by typing the exact name in this field.

   > You can only create a Drag&Relate supertype on the *Object View*.

3. Choose *Display* to the left of the field.

4. Choose *D&R Supertypes*.

5. Choose *Create*.

6. Specify the Drag&Relate supertype name.

7. Enter the name of the function module that is to determine the current business object at runtime in the *Func. module* field. This function module must determine which business object is to be called in a Drag&Relate context.

   > The function module must have the same interface as the function module WP_DR_OBJTYPE_GET_EXAMPLE (see F1 help on the *Func. module* field).

8. Choose *Object types* and enter the new supertype in the *Supertype* field of all object types required.

   > Only object type with identical key definitions can be assigned to the same supertype.

9. Assign a given data element to any number of objects that belong to the same supertype. The status indicator informs you whether the data element is assigned to the actual object or the supertype.

10. Save your entries.

Result

You have created a Drag&Relate supertype. A function module can determine the current business object for use in *Drag&Relate* actions.
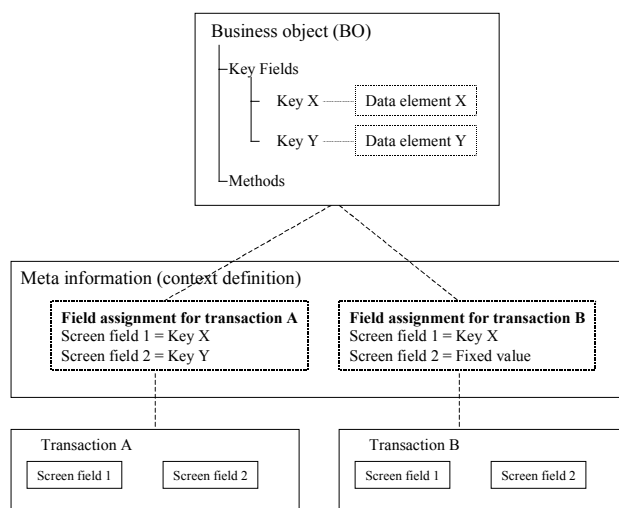
If you have finished maintaining the object metadata, you can continue by <u>activating the metadata [Page 55]</u>.

Assigning Target Transactions

Use

You can assign a whole range of (meaningful) transactions to business object types, which can then be called using Drag&Relate. These transactions can be described as methods of the business object. You can also assign a target transaction to more than one business object type. The transactions that you select depend on the scenarios that you want to support with Drag&Relate and that are useful from a business perspective.

```
┌─────────────────────────────────────────────┐
│ Business object (BO)                          │
│ ┌Key Fields                                   │
│ │     ┌── Key X ─────── Data element X         │
│ │     └── Key Y ─────── Data element Y         │
│ └Methods                                       │
└─────────────────────────────────────────────┘

┌──────────────────────────────────────────────────────────┐
│ Meta information (context definition)                       │
│ ┌─────────────────────────┐  ┌─────────────────────────┐   │
│ │ Field assignment for     │  │ Field assignment for     │   │
│ │ transaction A            │  │ transaction B            │   │
│ │ Screen field 1 = Key X   │  │ Screen field 1 = Key X   │   │
│ │ Screen field 2 = Key Y   │  │ Screen field 2 = Fixed   │   │
│ │                          │  │ value                    │   │
│ └─────────────────────────┘  └─────────────────────────┘   │
└──────────────────────────────────────────────────────────┘

┌─────────────────────────────┐  ┌─────────────────────────────┐
│ Transaction A               │  │ Transaction B               │
│ ┌Screen field 1┐ ┌Screen    │  │ ┌Screen field 1┐ ┌Screen    │
│ └──────────────┘ │field 2┐  │  │ └──────────────┘ │field 2┐  │
│                  └───────┘  │  │                  └───────┘  │
└─────────────────────────────┘  └─────────────────────────────┘
```

Font size needs to be increased in PPT to at least 18pt

The context is added by assigning the business object key fields to the input screen fields of the target transaction. In this way, the chosen target transaction becomes **relate-enabled**. You can have the system transfer a key value, a fixed value, or a system variable when you assign field values to a transaction. This specifies the value that the system passes on to the SAP Unifier for R/3 when the user performs a query.

From a user perspective, relate-enabled means that when a link on the desktop is dragged to a transaction in the iPanel (Drag&Drop), the selected target transaction is started with all the parameters that are uniquely connected with the link (= parameters of the target transaction are set by the business object).

Setting Parameters of the Target Transaction Using a Function Module

For target transactions that are to be processed separately, for example because the initial screen is missing altogether, you can have the system call a function module instead of starting the transaction. This function module is assigned to the target transaction.

Procedure

1. Start Transaction *SPO0*.

2. To change the definition of an existing object type, choose the relevant object type in the *Object type* field using the F4 help or by typing the exact name in this field.

3. Choose *Display* to the left of the field.

4. Switch to change mode by choosing *Display/change*.

5. Choose the *Transactions* tab.

6. A table is displayed in which you can enter the transaction codes for your chosen target transactions that are assigned to the business object type and that are to be relate-enabled.

> If the business object already has dialog methods (for example, *Display*, *Edit*, *Delete*), these provide a sensible starting point for the assignment.

7. To specify that the transaction is to be called using a function module, select the *Function module* column and specify the function module.

> You need to ensure that **exactly the assigned** transaction is processed, otherwise there may be authorization problems. The function module is only intended for more flexible parameterization of the actual target transaction and does not itself perform any authorization checks other than that the user has start authorization for the transaction.

8. The button in the first column of the table takes you to the *Maintain field assignments* menu. Here you specify for each target transaction the fields of the business object type with which the screen input fields of the target transaction are to be automatically set (parameters) for Drag&Relate. You can also specify fixed values.

9. To define a search help for parameterization of the target transaction, enter the search help attachment [Page 48] with the key in the second column. Enter the name of the "target object type" and the key field (for this business object type). In the last column, enter the name of the search help and the search help parameters. You can also specify fixed values here (as for *Maintaining field assignments*).

   A collective search help must have been attached to the input field of the transaction. The search help attachment provides the business environment of an object for Drag&Relate.

> If a search help is specified for the corresponding input field *Delivery* in transaction *Display delivery* and the parameter can be specified with the customer (that is, *Deliveries for customers*), you can call this search help with a specified parameter. This search help must be part of a collective search help that is assigned to the input field *Delivery*.

10. You can test the transaction call.
    To do so, select the appropriate icon in the third column of the table.

11. Save your entries.

Result

You have assigned transactions to the business object types, which can then be called using Drag&Relate.

If you have finished maintaining the object metadata, you can continue by activating the metadata [Page 55].

Search Help Attachment for the Target Transaction

You can provide **search helps** for setting the parameters of the target transaction.

This is important when the object instance that creates the drag-enabled output screen field is linked to more than one *"target instance"* with which the parameters of the *target transaction* can be set (for example there is more than one request (target instance) for a customer (object instance). The selection required here is specified by the search help. The target transaction is always parameterized with exactly one target instance.

For example, if a *Customer no.* field (corresponds to an instance of the business object type *customer*) is dragged to a transaction *Display delivery*, the reference is not unique since one *customer number* may have several *deliveries* (target instances) related to it.

Attaching search help to the target transaction simulates a relationship between the business object on the source and target side (a relationship between the business object types *delivery* and *customer* in the above example).

Simulating an Object-To-Object Relationship using the *Search Help Attachment* of the Target Transaction

The *Search help attachment* of the target transaction allows you to implement a relationship between the business objects on the source and target sides.

The search help attachment is useful in scenarios where the parameters of the target transaction **cannot be uniquely** set using the source instance. In Drag&Relate chains, however, non-unique parameters may only be set at the end of the chain.

For example, the "source instance" *customer no. 0815* could be dragged to the target transaction *Display delivery.*

Activating Relationships for Business Objects

Use

Business object types can be linked to other business object types using an **object-to-object relationship** and can then use the target transactions of the other business object types as well.

You have the following options for simulating this type of relationship:

1. Activate *object attributes*

2. Simulate an object-to-object relationship using the search help attachment [Page 48] for the target transaction.

Activating Object Attributes

This method of implementing object-to-object relationships is useful in scenarios where the parameters of the target transaction can be **uniquely** set using the source instance. An example is when the "source instance" *delivery no. 4711* is dragged to the target transaction *Display customer.* To implement this scenario, you first have to maintain an object-to-object relationship between the business object types *Delivery* (source side) and *Customer* (target side).

You can only do this if the business object types have already been linked to each other using an appropriate attribute in the Business Object Repository (BOR).

If an object-to-object relationship is maintained using this method, all the target transactions assigned to the business object type on the source side (*Customer*) are available for the drag-enabled screen fields assigned to the business object type on the target side (*Delivery*). The object-object relations are transient in this case, that is, if the relationships $A \rightarrow B$ and $B \rightarrow C$ exist, then there is also a relationship $A \rightarrow C$.

Prerequisites

- The appropriate object attributes must be implemented in the BOR for the relevant object relationships. Need t use List Bullet style

- Use of multi-value attributes is not allowed because the action performed with Drag&Relate must be unambiguous.
  This applies for the current Drag&Relate implementation at least, and does not exclude the possibility of multi-value object-to-object relationships in the future.

- Only relationships between Drag&Relate-enabled business object types are supported.

Procedure

1. Start Transaction *SPO0*.

2. To change the definition of an existing object type, choose the relevant object type in the *Object type* field using the F4 help or by typing the exact name in this field.

3. Choose *Display* to the left of the field.

4. Switch to change mode by choosing *Display/change*.

5. Choose the *Object relations* tab. The corresponding relationships to other objects are displayed for the attributes that are implemented on the chosen business object type.

   On the *Object Relations* tab, you can set a priority for each attribute of an object. This is a numeric value, where 1 represents the highest priority. This priority setting specifies which attribute is used for the query.

Some business objects can produce more than one value for a Drag&Relate query, depending on the precise attribute concerned. When the user submits a query based on an **object that can return more than one result**, the SAP Unifier for R/3 needs to know which attribute to use to return the correct result for the query. The user specifies in their personalization settings whether a dialog allows them to select the attribute to be used or whether they want the system to use the attribute with the highest priority.

An order number is dragged and related to the customer address data. In this case, the intended address could be the address of the ordering party (such as the company's head office) or the delivery address (such as a plant in a different location).

6. Activate the object relationships that you consider useful by selecting the corresponding column.

7. Save your entries.

Result

You have activated relationships for business objects.

If you have finished maintaining the object metadata, you can continue by .

Specifying Explicit Attribute Access Using a Function Module

Use

You use this procedure to specify a function module to be used for explicit attribute access in Drag&Relate operations.

Theoretically, this method does not produce an *object-to-object relationship* as it "bypasses" the Business Object Repository (BOR). It can only be used to explicitly specify the parameters of Web components.

These function modules are used to read out the attributes of objects that are not specified as attributes of the BOR at runtime.

Procedure

1. Choose the *Attribute access* tab.

2. Enter the function module that you wish to use for the attribute access.

3. The *prefix* is used to distinguish between function modules if several function modules with the same parameter names are used to read detailed data for an object.
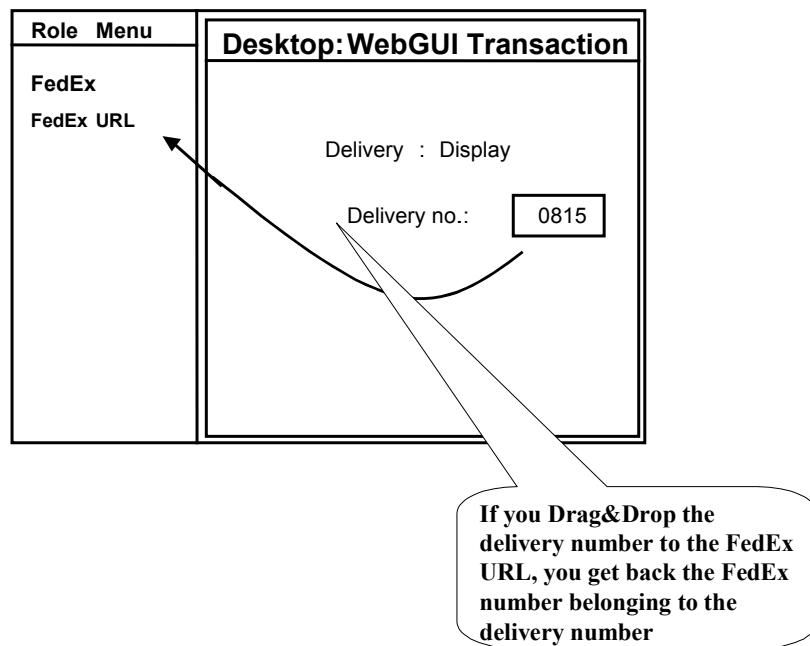
4. Save your entries.

Result

You use this procedure to specify a function module to be used for explicit attribute access in Drag&Relate operations.

Creating Attributes for Specifying Web Component Parameters

Use

You can **specify parameters** for **external Web components** and supply them with information other than the key information of the source object. This procedure does not precisely include the definition of the parameters for a target transaction as in the Drag&Relate variants discussed previously, but means completing a "target URL" of a Web service (such as a transport or shipping service) by adding the information of the (dragged) source object.

```
┌─────────────┬───────────────────────────────┐
│ Role  Menu  │  Desktop: WebGUI Transaction   │
│             ├───────────────────────────────┤
│  FedEx      │                                │
│  FedEx URL  │       Delivery  :  Display     │
│             │                                │
│             │      Delivery no.:   ┌──────┐  │
│             │                      │ 0815 │  │
│             │                      └──────┘  │
│             │                                │
│             │                                │
└─────────────┴───────────────────────────────┘
```

> *If you Drag&Drop the delivery number to the FedEx URL, you get back the FedEx number belonging to the delivery number*

When maintaining the metadata, you can specify function modules for reading additional object attributes that are not specified in the Business Object Repository (BOR). The export parameters of these function modules provide information for parametrizing the "target URLs" of external Web services. These function modules can have any fields as import parameters, but it must be possible to parameterize these fields with key fields of the source business object type.

> In contrast to the normal target transactions, Web components cannot execute collective search helps. Here you can assign any function modules that are called at runtime and that provide their export interface to the SAP Unifier for R/3. Any interface is possible, but it must be specified meaningfully with parameters for the key components of the object or with fixed values.

For more information about Web components, see *Creating Web Components* in the *Help for the Portal Administrator*.

You can also create a suitable ad-hoc query [Page 54] to define the parameters for external Web components.

Procedure

1. Start Transaction *SPO0*.

2. To change the definition of an existing object type, choose the relevant object type in the *Object type* field using the F4 help or by typing the exact name in this field.

3. Choose *Display* to the right of the field.

4. Switch to change mode by choosing *Display/change*.

5. Choose the *Attribute access* tab.

6. Enter the function module that you wish to use for the attribute access.

7. The *prefix* is used to distinguish between function modules if several function modules with the same parameter names are used to read detailed data for an object.

8. The icon in the first column of the table takes you to the *Maintain Parameters* screen. Specify here the values for the import parameters, such as key fields of the source object, fixed values or system variables.

9. Test the access to the attribute. To do so choose *Test* on the *Attribute access* tab.

10. Save your entries.

Result

You have created attributes that can be used to specify parameters for external Web components.

Ad Hoc Query

You can create an appropriate ad hoc query to specify parameters for external Web components.

In logical SAP R/3 Systems of Release 4.6B or higher, transaction *SPO0* contains an *Ad hoc Query* tab. This uses the concept of InfoSets, which are usually based on table joins or logical databases, to create abstract definitions for fields that the system can read. You have to enter parameters for the individual fields. The ad hoc query delivers values that can be passed to the SAP Unifier for R/3.

Transaction *SPO0* also contains a buffer for ad hoc queries. This improves system performance by avoiding the need to put together the query every time it is used.

You can delete the contents of this buffer by choosing *Delete Buffer for AdHoc Query* on the start screen of transaction *SPO0*. This removes the index entry and the generated queries from the system. We recommend doing this if you change the InfoSet that the query was generated from, for example.

For more information about this function, see the documentation on the InfoSet Query in the SAP R/3 System Basis documentation.

Activating Metadata

Before you can work with Drag&Relate, you must activate the metadata in the logical system. The screen field and the Drag&Relate object link are connected by data elements. Active data elements are data elements that have an assignment to an object type. The system generates object links for all screen fields that reference one of these active data elements. During the activation of Drag&Relate supertypes, the system checks the function module assigned to determine the actual business object type.

The system uses a time stamp to check if an object is active or inactive. If the last **activated** time is after the last saved time, the object is active; if the last **saved** time is after the last activated time, the object is inactive.

When you display the metadata for an object, the active version appears. As soon as you change the object, its status reverts to inactive. The last active version continues to be used for any Drag&Relate operations. The active version is the SAP delivery version if the SAP import is more recent than the last customer version maintained.

You can transport modified objects automatically. This makes sense if you transport the metadata from a development system to a test system or from a test system to a production system.
A new import overwrites locally changed objects. Alternatively you can configure the system so that both the import version and the local version are stored in parallel. You can first check the objects and then activate them manually if needed. In the *Drag&Relate Customizing Check* area, choose *Settings*. In the *Settings for Interactive Maintenance of Drag&Relate Data* dialog box, set the *Transport* mode to *M* for *No Automatic Transport*.

> The dependent table *SPRTL2* is not transported. The system generates this table when you activate the metadata. You can activate either the SAP delivery version, tables *SPRTL0* and *SPRTL1*, or the modified customer version, tables *SPRTL0C* and *SPRTL1*C.

You can activate objects simultaneously. When you choose *Mass Activator* a list of all objects in the system appears with their status. You can select the objects to be activated from this list and then activate them at one time.

> Objects created in the customer version are only visible in the SAP delivery version if they are activated from the customer version to the SAP delivery version.

> You cannot maintain the SAP delivery version unless you are in *Transport* mode *S* for *Transport of SAP tables*.

Adjusting Table Structures

Purpose

The table structure for the Drag&Relate metadata has changed between Workplace Plug-In 2.11 and SAP Enterprise Portal Plug-In 5.0. The SAP delivery version of the Drag&Relate metadata still has the old table structure. You therefore must adjust the table structures after importing the metadata.

Process Flow

Start transaction *SPO0*. You are asked in a dialog box whether you want to migrate the data. This transfers the table entries from the obsolete tables to the new tables.

Once you have performed the migration, the objects exist solely in the *SAP delivery version*. Then choose *Activate SAP Version*. This creates a copy of all of these objects in the customer tables and updates certain runtime tables.

Once you have completed the activation step, you can use your existing Drag&Relate data in production operations.

Protecting Modified Drag&Relate Metadata before Overwriting

If you want to protect the existing data against being overwritten by a subsequent (SAP) import, you must choose the option *No automatic activation in your system* (*Settings* function in the maintenance transaction). If you subsequently import Drag&Relate data, the system does not activate the data automatically, and you can activate the objects individually manually or using the mass activator. In addition, you can compare the delivery version or the imported version with the active version before activating it. You must set the import switch, since the objects from the SAP delivery version have the status *not modified* once they have been activated.

If you want to protect only individual objects against being overwritten, for example, because you have actually modified only a few objects, you can also do this by saving the data for these objects and activating them again. This process means that these objects are flagged as *modified* and are then not automatically overwritten during an import (unless, the import switch has been set accordingly).

Transport of Drag&Relate Metadata to Other Component Systems

If you want to use Drag&Relate metadata in more than one component system, it is sufficient to adjust the table structures in one system and to transport this data. Only obsolete tables are then deleted in the target system. Production data is not affected.

In the maintenance transaction choose *Transport Metadata* and select the objects to be transported. In the next step, these are written to a transport request, which you can then distribute using the regular transport tools.

Instead of transporting the data, you can, of course, perform the migration manually in all systems.

Uninstalling the SAP Unifier for R/3

Use

Use this procedure to remove the SAP Unifier for R/3 from your Unification Server installation.

Prerequisites

You have removed any existing projects that you generated with the SAP Unifier for R/3.

Procedure

1. From the Windows Control Panel, choose *Add/Remove → SAP Unifier*.

2. Follow the screen instructions for removing the unifier.

Result

If this is the only instance of SAP Unifier for R/3 that is installed on Unification Server, you can no longer generate an SAP Unifier for R/3 project.

To reinstall the R/3 Unifier, return to .