

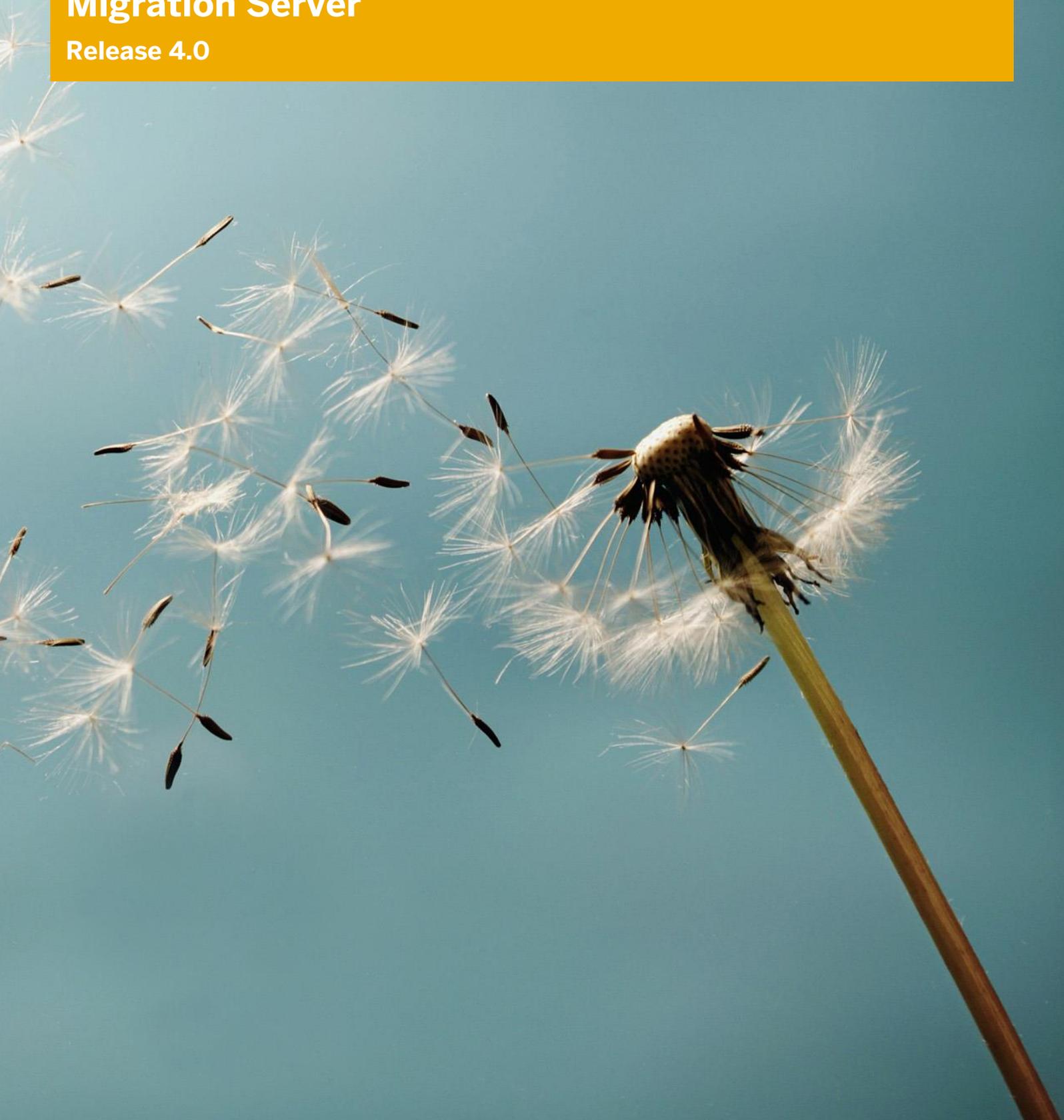
How-To Guide

Document Version: 1.4 – 2015-02-16

CUSTOMER

How to Scramble Data Using SAP Test Data Migration Server

Release 4.0



Typographic Conventions

Type Style	Description
<i>Example</i>	Words or characters quoted from the screen. These include field names, screen titles, pushbuttons labels, menu names, menu paths, and menu options. Textual cross-references to other documents.
Example	Emphasized words or expressions.
EXAMPLE	Technical names of system objects. These include report names, program names, transaction codes, table names, and key concepts of a programming language when they are surrounded by body text, for example, SELECT and INCLUDE.
Example	Output on the screen. This includes file and directory names and their paths, messages, names of variables and parameters, source text, and names of installation, upgrade and database tools.
Example	Exact user entry. These are words or characters that you enter in the system exactly as they appear in the documentation.
<Example>	Variable user entry. Angle brackets indicate that you replace these words and characters with appropriate entries to make entries in the system.
EXAMPLE	Keys on the keyboard, for example, F2 or ENTER.

Document History

Version	Date	Change
1.0	October 2011	First Version of this guide
1.1	March 2013	Extensive reworking of the structure and information flow
1.2	August 2013	Scrambling simulation and stand-alone scrambling
1.3	February 2014	Import-Export of Data from Files for Manual 1:1 Mapping
1.4	February 2015	Transformation of information and information architecture to the people-centric format.

Table of Contents

1	Business Scenario	6
2	Getting Started.....	7
2.1	Navigating to the TDMS Work Center.....	7
2.2	Roles and Authorizations.....	7
2.3	System Landscape for Data Scrambling.....	8
2.3.1	Scrambling System Requirements	8
2.3.2	Landscape for the Stand-Alone Scrambling Scenario	8
2.3.3	Landscape for Scrambling During Data Transfer	9
2.3.4	Landscape for Stand-Alone Conversion Package.....	10
2.4	Taking the First Steps	11
3	Scrambling Objects.....	12
3.1	Solution Category.....	12
3.2	Scrambling Supergroups and Scrambling Groups	13
3.3	Scrambling Rules.....	14
3.4	Global Mapping	14
4	Stand-Alone Scrambling	16
5	Scrambling During Data Transfer	18
6	Stand-Alone Conversion.....	20
7	Working with Standard Scrambling Content.....	21
7.1	Preparing the Ready-to-Use Content	21
7.1.1	Ready-to-Use Standard Content from SAP TDMS	22
8	Working with Custom Requirements.....	23
8.1	Understanding the Process Flow Graphically	23
8.2	Working with Multiple Product Levels	27
8.3	Working with the User Interface to Create Scrambling Objects	27
8.4	Working with the Solution Category	28
8.4.1	Creating a Solution Category.....	28
8.4.2	Assigning the TDMS Migration Solution to a Solution Category (Required).....	29
8.5	Working with the Scrambling Supergroup	31
8.6	Working with the Scrambling Group.....	32
8.7	Copying a Scrambling Rule.....	33
9	Working with Scrambling Rules	35
9.1	Creating Rules as a Standard User	35
9.1.1	Step One: Adding a Scrambling Type (Required)	36
9.1.2	Delete Value.....	36
9.1.3	Fixed Value.....	38

9.1.4	Manual 1 to 1 Mapping.....	39
9.1.5	Number Conversion	40
9.1.6	Random Table with One Column	41
9.1.7	Value Range Table.....	43
9.1.8	Random Table for Time Periods	44
9.1.9	Custom Mapping	45
9.1.10	Step Two: Adding Table-Field Combinations (Required).....	51
9.1.11	Step Three: Setting the Primary Table-Field (Required).....	53
9.1.12	Step Four: Using a Key-Set.....	55
9.1.13	Step Five: Using a Domain.....	56
9.2	Additional Features for an Expert User.....	57
9.2.1	Step Six: Assigning Routine Names.....	57
9.2.2	Step Seven: Assigning Includes	58
9.2.3	Step Eight: Using the Identifier and Scrambling Function Modules	59
9.2.4	Step Nine: Using Condition-Sets	62
10	Consistent Scrambling of Data	64
10.1	Global Mapping	65
11	Additional Features in the Scrambling Workbench.....	67
11.1	Validation Feature for Scrambling Rules	67
11.2	Simulation for Scrambling Rules	67
11.3	Search Feature for Scrambling Rules	70
11.4	Import-Export of Data from Files for Manual 1:1 Mapping	71
11.4.1	Prerequisites.....	71
11.4.2	Exporting Data to the .CSV File.....	71
11.4.3	Importing the Mapping Values to a Scrambling Rule.....	73
12	Using Scrambling Objects at the Project and Package Levels.....	75
12.1	Copy Option at Project Level	75
12.2	Activating Scrambling Customizing at Project Level	76
12.3	Copy Option at the Migration Package Level	77
12.4	Activating Scrambling Customizing at the Migration Package Level	78
13	Example Rule Creation	79
14	Troubleshooting	88
15	FAQs.....	91
15.1	Types of Data that Can Be Scrambled.....	91
15.2	Solution Category	91
15.3	Scrambling Rules: Basic Technical Information	91
15.4	Scrambling Rules: Condition Sets.....	92
15.5	Scrambling Rules: Expert Technical Information	92
15.6	Miscellaneous.....	93
16	The Scrambling Configuration After Upgrade Tool.....	94

1 Business Scenario

Your organization has a lot of sensitive data in the systems and you want to scramble the data before making it available to testers, developers, and others.

You can use SAP TDMS 4.0 to make this possible by scrambling data before making it available in the non-production system.

1 Note

Data scrambling does not scramble the actual data residing in the production system.

You can scramble data in the following contexts:

- **Stand-Alone Scrambling:** You can scramble data residing in a particular system using this scenario. Typically, you have set up your non-production system with data copied from the production system or a quality system. You want to scramble the data to prevent the developers and testers from viewing the original data of the production system.

- **Scrambling During Data Transfer:** You can scramble data selected for transfer across systems by SAP TDMS.

SAP TDMS selects the data for transfer and stores it in a temporary cluster in the sender system. SAP TDMS next scrambles the data in the cluster before transferring it to the receiver system.

- **Stand-Alone Conversion:** You have built a test system by copying the data from the production system or a quality system. During the copy, the production system name is also copied to the test system. To uniquely identify the test system, you want to rename the logical system name of the test system.

You can rename the logical system names residing in the test system to your preferred list of names.

You can use the stand-alone conversion as an advanced parallel processing technique that serves as an alternative to BDLS - the standard system copy process for logical system rename.

2 Getting Started

2.1 Navigating to the TDMS Work Center

Method 1

Install the SAP NetWeaver Business Client 3.5 or 4.0. 1.

To add your system to the NWBC, enter the following information:

- System name
- URL for the system
- System type
- Client
- Language

You can get this information from the transaction TDMS_CFGCHK in your SAP Solution Manager system. After you enter the transaction, select your NWBC version and execute to view the system information.

Method 2

Add your SAP logon system to your SAP logon pad by entering the following information:

- System ID
- Message server URL
- group 1.server
- System number

Use transaction TDMS to enter the TDMS work center.

2.2 Roles and Authorizations

Assign appropriate roles to your users from among these roles:

- SAP_TDMS_SCRAMBLING_DISPL
A user with this role only has Display authorizations for scrambling objects.
- SAP_TDMS_SCRAMBLING_ADMIN
A user with this role has Create and Edit authorizations.
This role is a part of the composite role SAP_TDMS_MASTER_CR.
- SAP_TDMS_SCRAMBLING_USER

A user with this role has Copy and Activate authorizations for the scrambling objects created at the package level.

This role is a part of the composite role SAP_TDMS_TECH_EXECUTION_USER.

- SAP_TDMS_SCRAMBLING_EXPERT

A user with this role has Copy and Activate authorizations for scrambling objects created at both the project and package levels.

This role is a part of the composite role SAP_TDMS_TECH_EXECUTION_EXPERT.

- SAP_TDMS_SCRAMBLING_BUS_EXPERT

A user with this role has Create, Change, Display, Copy, and Activate authorizations for all scrambling objects created at both the project and package levels.

This role is a part of the composite role SAP_TDMS_BUSINESS_EXPERT.

- SAP_SLOP_MASTER

This rule should be assigned to the RFC user that is used in the RFC connection.

2.3 System Landscape for Data Scrambling

This section describes the system landscape requirements for the scrambling scenarios.

For information about the system landscape for SAP TDMS, see the Master Guide for SAP TDMS 4.0 on SAP Service Marketplace at <http://service.sap.com/tdms>.

2.3.1 Scrambling System Requirements

For information about the scrambling system requirements, see the landscape requirements outlined in the Master Guide for SAP TDMS and the detailed information in the Operations Guide for SAP TDMS on SAP Service Marketplace at <http://service.sap.com/tdms>.

For the Execution System:

You have installed SAP Basis Release 620 or above.

2.3.2 Landscape for the Stand-Alone Scrambling Scenario

This figure shows a typical landscape and communication flow among the systems in a stand-alone scrambling scenario.

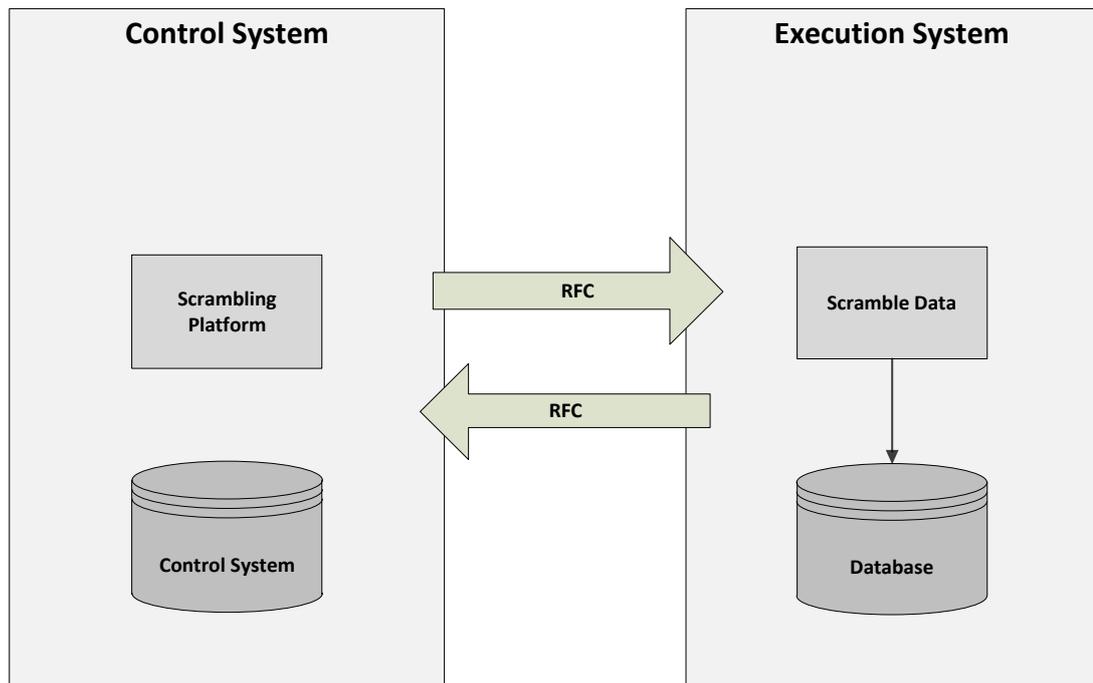


Figure 1: System landscape for the stand-alone scrambling scenario

Control System

All TDMS scrambling-specific settings and customizing are stored in this system. This system triggers and monitors the activities in the process tree.

Execution System

The execution system is a non-production system that has already been set up using a client copy, system copy, or SAP TDMS (without scrambling). This is the system where you would like to scramble the sensitive content before users can handle the data.

2.3.3 Landscape for Scrambling During Data Transfer

Here's a diagram of a typical landscape and the communication flow among the systems for scrambling during data transfer.

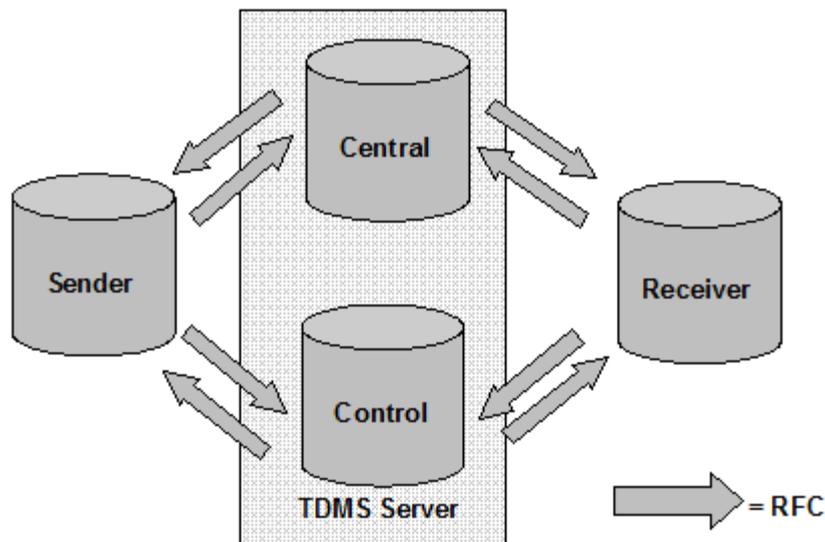


Figure 2: System landscape for the transfer scenario

Sender System

This supplies data to the non-production system that you want to set up using SAP TDMS. The sender system is usually a production system. SAP TDMS selects data from the sender system based on your selection criteria and stores the data in a cluster table in the sender system. Before transfer, SAP TDMS reads the data from this cluster table, scrambles the data, and stores the data in the same cluster table. During transfer, SAP TDMS transfers the scrambled content to the receiver system.

Receiver System

This is the target system in the SAP TDMS data migration process.

SAP TDMS transfers the scrambled data from the sender system to the receiver system.

The TDMS Server

- **Control System:** All TDMS-specific settings and Customizing are stored in the control system. The control system triggers the TDMS activities in the process tree.
- **Central System:** The back-end processing for data migration takes place in the central system.

2.3.4 Landscape for Stand-Alone Conversion Package

For information about the landscape for Logical System Rename package, take a look at Section 2.2.2 [Landscape for the Stand-Alone Scrambling scenario](#).

2.4 Taking the First Steps

We provide you with a scrambling workbench to design and develop the objects to scramble data. This workbench is used across all applications running SAP TDMS and can be accessed through the TDMS work center under the transaction TDMS. Navigate to the Data Scrambling Workbench from the TDMS Work center.

To start using TDMS Scrambling, copy the scrambling content we deliver to your work center by choosing the [Copy from Template](#) pushbutton. During the first run, this option copies the objects provided by SAP to the work center.

After you copy the initial content from the template to the work center, you can copy any additional content available with support packages using these copy options:

- **Copy Enhanced Scrambling Objects:** Copies an enhancement provided by SAP but not yet available with the customers.
- **Append Scrambling Objects:** Appends the scrambling objects delivered by SAP to the work center.

3 Scrambling Objects

SAP TDMS provides standard scrambling content so that you can carry out data scrambling. If you find that you need additional content, you can define the following TDMS scrambling objects:

- Solution Category
- Supergroup
- Group
- Scrambling Rule
- Global Mapping
-
- Keep reading for a detailed description of each scrambling object.

3.1 Solution Category

You can use the solution categories to group scrambling objects (groups, rules and mappings) according to specific TDMS migration solutions (TDMS packages). The Master Guide for SAP TDMS 4.0 contains more information about TDMS migration solutions. You can find it in the SAP Service Marketplace at <http://service.sap.com.tdms>.

Example

If, for example, you want to scramble vendor names (Table LFA1 Field NAME1) in the SAP ERP application, SAP TDMS provides the solution category SAP_ERP that groups all the TDMS ERP migration solutions.

You have created a scrambling group called `VENDOR_GROUP`, a scrambling rule called `VENDOR_NAME` and a mapping table called `VENDOR_MAPPING`. You should assign all these objects to the solution category `SAP_ERP` to ensure that the objects are only used for the TDMS ERP migration solutions. If you assign the objects to `SAP_CRM`, errors result during the scrambling process because the vendor name table `LFA1` does not exist in the CRM system.

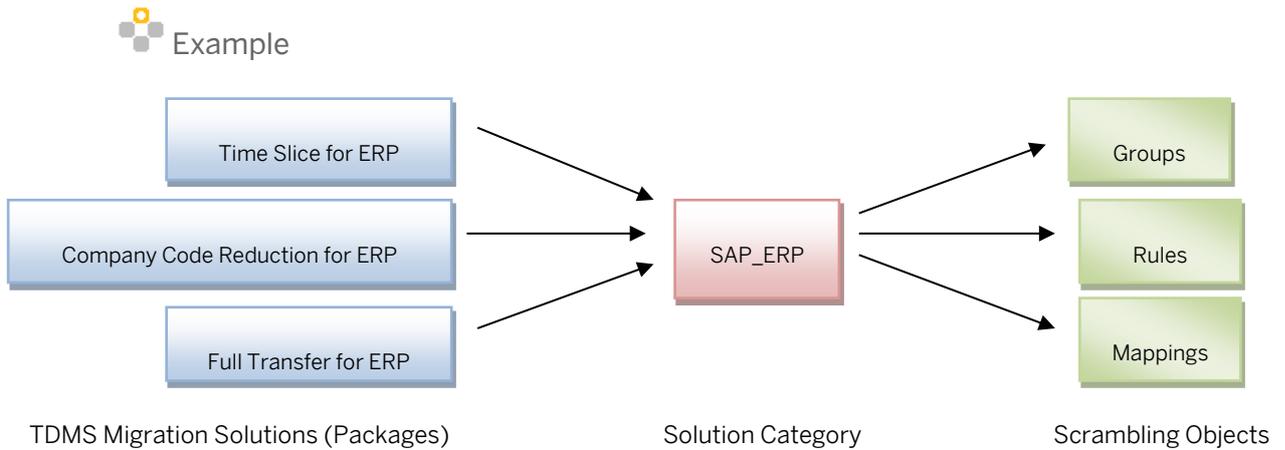


Figure 3: Using the Solution Category: An Example

This table displays the complete list of solution categories provided by SAP TDMS 4.0:

Maintain Migration Solution Categories	
<input type="button" value="New"/>	<input type="button" value="Rename"/>
<input type="button" value="Delete"/>	<input type="button" value="Assign Solution"/>
	Solution Category
<input type="checkbox"/>	SAP_ALL
<input type="checkbox"/>	SAP_ERP
<input type="checkbox"/>	SAP_CRM
<input type="checkbox"/>	SAP_HCM
<input type="checkbox"/>	SAP_SRM
<input type="checkbox"/>	SAP_SCM
<input type="checkbox"/>	SAP_BI

Figure 4: Solution Categories Available with SAP TDMS

3.2 Scrambling Supergroups and Scrambling Groups

You can use Scrambling supergroups and groups to facilitate the grouping of related rules for easy execution of actions such as activating or deactivating a group of rules together. Supergroups are the root (top) elements of the hierarchy and can contain both groups and rules.

You can correlate supergroups to the top-level folders in a computing environment and groups to the subfolders in the same environment.

A multi-level nesting of groups is possible, where groups can contain groups that in turn contain further groups.

Example

Among the ready-to-use scrambling objects provided by SAP, Vendor Address and Vendor Number are groups under the Vendor supergroup. If you want to activate rules related to Vendor, you just need to activate the Vendor supergroup.

Scrambling Overview			
Scrambling Objects	Transport	Status	Object Type
▶ Vendor	<input type="checkbox"/>	○○●	Supergroup
▶ Customer	<input type="checkbox"/>	○○●	Supergroup
▶ Material	<input type="checkbox"/>	○○●	Supergroup
▼ Personnel Data	<input type="checkbox"/>	○△○	Supergroup
▼ General Data	<input type="checkbox"/>	○○●	Group
▼ Personnel OM Data	<input type="checkbox"/>	○○●	Group
▼ Personnel OM Position	<input type="checkbox"/>	○○●	Group
▪ Rule Personnel Position3	<input type="checkbox"/>	○○●	Rule
▪ Rule Personnel Position1	<input type="checkbox"/>	○○●	Rule
▪ Rule Personnel Position2	<input type="checkbox"/>	○○●	Rule
▶ Personnel OM Job	<input type="checkbox"/>	○○●	Group
▶ Personnel OM Org ID	<input type="checkbox"/>	○○●	Group
▶ Personnel Name	<input type="checkbox"/>	○○●	Group
▶ Personnel Birth Data	<input type="checkbox"/>	○○●	Group
▶ Personnel Address	<input type="checkbox"/>	○○●	Group
▶ Personnel Other Data	<input type="checkbox"/>	○○●	Group
▶ Personnel Bank Data	<input type="checkbox"/>	○○●	Group
▶ Personal Generic Data Reset	<input type="checkbox"/>	○○●	Group
▶ Personnel CATS Data	<input type="checkbox"/>	○○●	Group
▶ Country Specific	<input type="checkbox"/>	○△○	Group

Figure 5: Using Supergroups to Categorize Scrambling Objects

3.3 Scrambling Rules

A scrambling rule contains:

- Logic to derive the scrambled values
- A list of table-fields and domains to be scrambled
- Conditions (available in the Expert mode)
- Includes (available in the Expert mode)

You can use scrambling rule to provide scrambling logic and other technical information to scramble data.

You can select a scrambling logic available with standard scrambling types. For more information about scrambling types, see Section 9.1.1 [Scrambling Types](#).

After you select the scrambling type, you need to specify a list of table and field combinations and domains that you want scrambled. For more information, take a look at Chapter 9: [Working with Scrambling Rules](#).

3.4 Global Mapping

If you want to ensure cultural consistency of names after scrambling or geographical consistency of addresses after scrambling, you can use Global Mapping scrambling objects.

For more information, take a look at section 10.1 [Global Mapping](#).

Maintain Global Mapping

Mapping Name Address Mapping Type Random Selection Table With 4 Columns

Save Cancel

Global Mapping Data

Solution Category: SAP_ERP

Created By: KURMADAS Changed By: KURMADAS

Created On: 29.01.2013 Changed On: 29.01.2013

Scrambling Column Number: 00

Maintain Scrambling Values

Append Row Insert Row Delete Row Edit Column Text Import Export Filter

Street	City	State	Country
Kent Street	Sydney	NSW	Australia
Whitefield	Bangalore	Karnataka	India
St Peter Road	Chicago	Illinois	US

Figure 6: Maintaining Global Mapping in the work center

4 Stand-Alone Scrambling

You use this scenario to scramble data residing in a system and to store the scrambled data in the same system.

Prerequisites

You have created the required scrambling content in the Scrambling workbench in the TDMS work center.

Procedure

Here's what you need to do to run a Stand-Alone scrambling package:

1. Navigate to the TDMS Work Center and create a project with the template `TDMS_TEMPLATE_STANDALONE_SCR` for stand-alone scrambling package execution.
For more information about creating projects, see the SAP Help Portal at <http://help.sap.com> and go to Application Lifecycle Management → SAP Test Data Migration Server → SAP Test Data Migration Server 4.0 → Application Help → English → Projects.
2. Switch to the Edit mode on the Project screen, assign a system landscape (type: Stand-Alone Scrambling), and save your entry.
3. Create a package for Stand-alone Scrambling by choosing the *Create Package* pushbutton.
4. In the process tree for the package, you can view all the activities for the Stand-Alone scrambling package grouped under the phases Package Settings, Data Scrambling, and Post-Transfer Processing.
5. Execute the background activities in the phase Package Settings until you reach the dialog activity Define Scrambling Rules.
 1. Execute the dialog activity Define Scrambling Rules.
 2. On the Scrambling Overview screen, choose the Copy pushbutton. Choose an option: *Copy from Project* or *Copy from Work Center* to copy the scrambling objects from the Project or Scrambling workbench respectively.
 3. Activate the scrambling objects relevant for scrambling in the current package by choosing the *Activate* checkbox next to each object.
6. Execute the remaining activities under the phase Package Settings. You can also view the activity documentation for each activity by choosing the activity name.
7. Monitor the status of the activities from the process tree.
You can find more information about monitoring in the Operations Guide for SAP TDMS 4.0 on SAP Service Marketplace at <http://service.sap.com/tdms>.
8. After you successfully complete the activities in the Package Settings phase, execute the activity *Start Data Scrambling/Conversion* in the Data Scrambling phase to start the data scrambling process.
Choose the *Ext. Process Monitor* pushbutton to see how data scrambling is progressing.
9. After you successfully complete the scrambling activity *Start Data Scrambling/Conversion*, you can validate the data in the execution system.



Recommendation

Validate the converted data before executing the activities in the Post-Transfer Processing phase, as executing the activities in the Post-Transfer Processing phase clears any temporary data stored in the execution system used for data scrambling.

-
10. After you successfully validate the scrambled data, execute the activities in the Post-Transfer Processing phase.

5 Scrambling During Data Transfer

You use this scenario to scramble data selected for transfer from a sender system to a receiver system.

Prerequisites

You have created the required scrambling content in the Scrambling workbench in the TDMS work center.

Procedure

Here's what you need to do to run a data transfer package for scrambling:

1. Navigate to the TDMS work center and create a project with the template `TDMS_TEMPLATE_TIME`, for example, to create a time-based reduction package for SAP ERP.
For more information about creating projects, see the SAP Help Portal at <http://help.sap.com> and go to Application Lifecycle Management → SAP Test Data Migration Server → SAP Test Data Migration Server 4.0 → Application Help → English → Projects.
2. Switch to the Edit mode on the Project screen and assign a system landscape (Type: TDMS Landscape Template) and save your entry.
3. Create a package by choosing the *Create Package* pushbutton.
4. In the process tree for the package, you can view all the activities for the ERP transfer package grouped under the following phases:
 - Package Settings
 - System Analysis
 - Data Transfer
 - Post-Transfer Processing
5. Execute all activities in the phase Package Settings until you reach the dialog activity *Define Scrambling Rules*.
 4. Execute the Define Scrambling Rules activity. On the dialog box asking you whether you want to scramble the data, choose *Yes* and proceed.
 5. On the Scrambling Overview screen, choose the *Copy* pushbutton. Choose an option: *Copy from Project* or *Copy from Work Center* to copy the scrambling objects from the Project or Scrambling workbench respectively.
 6. Activate the scrambling objects relevant for scrambling in the current package by choosing the *Activate* checkbox next to each object.
6. Execute the remaining activities in the Package Settings phase. You can also view the activity documentation for each activity by choosing the activity name.
7. Monitor the status of the activities from the process tree.
The Operations Guide for SAP TDMS 4.0 contains more information about monitoring SAP TDMS. You can find it in the SAP Service Marketplace at <http://service.sap.com/tdms>.
8. After you successfully complete the activities in the Package Settings phase, execute all activities in the phase System Analysis.
9. After you successfully complete the activities in the System Analysis phase, execute all activities in the phase Data Transfer until you reach the activity *Data Transfer Start*.

-
10. Execute the *Data Transfer Start* activity .After you successfully complete the *Data Transfer Start* activity in the Data Transfer phase, go to the receiver system and validate the scrambled data.

 Recommendation

Validate the scrambled data before executing the activities in the Post-Transfer Processing phase, as executing the activities in the phase Post-Transfer Processing phase clears any temporary data stored in the execution system used for data scrambling.

11. Execute the remaining activities in the Data Transfer phase.
12. After you successfully validate the scrambled data, execute the activities in the Post-Transfer Processing phase.

6 Stand-Alone Conversion

You use this scenario to convert the logical system names of the target system to your preferred list of names.

Procedure

Here's what you need to do to run a stand-alone package for data conversion:

1. Navigate to the TDMS work center and create a project with the template `TDMS_TEMPLATE_STANDALONE_LOGSYS` to execute the stand-alone conversion package for Logical System Rename.
For more information about creating projects, see the SAP Help Portal at <http://help.sap.com> and go to Application Lifecycle Management → SAP Test Data Migration Server → SAP Test Data Migration Server 4.0 → Application Help → English → Projects.
2. Switch to the Edit mode on the Project screen, assign a system landscape (type: TDMS Landscape Template for Stand-Alone Scrambling), and save your entry.
3. Create a package for Logical System Rename by choosing the *Create Package* pushbutton.
4. In the process tree for the package, you can view all the activities for the Logical System Rename package grouped under a single phase: *Logical System Rename*. You can view the activity documentation for each activity by choosing the activity name.
5. Execute the background activities in the phase Logical System Rename until you reach the dialog activity *Maintain Settings for Logical System Rename*.
 7. Execute the activity *Maintain Settings for Logical System Rename*.
 8. On the Overview screen, choose the tab page *Map Logical System Names* and enter the new name for the old logical system name.
 9. Navigate to the tab pages *Table-Fields* and *Domains* and specify any additional table-field combinations or domains relevant for logical system rename.
 10. Save your entries and close the screen.
6. Execute the remaining activities in the phase Logical System Rename until you reach activity *Start Data Scrambling/Conversion*.
7. Execute activity *Start Data Scrambling/Conversion* to start the data conversion.
8. Monitor the status of the activities from the process tree.
 - o Choose the *Ext. Process Monitor* pushbutton to see how data scrambling is progressing.
 - o For more information about monitoring, take a look at the chapter on monitoring in the Operations Guide for SAP TDMS 4.0 on SAP Service Marketplace at <http://service.sap.com/tdms>.
9. After you successfully complete the *Start Data Scrambling/Conversion* activity, go to the execution system and validate the converted content.
10. Execute the remaining activities in the Logical System Rename phase.

7 Working with Standard Scrambling Content

If you want to use the ready-to-use scrambling content provided by SAP TDMS, you can use the Scrambling workbench in the TDMS work center.

7.1 Preparing the Ready-to-Use Content

We provide some content ready for use in data scrambling. To view the content provided by SAP TDMS, you have to copy the content from the standard templates.

Recommendation

You need to copy the standard content from the templates at the start of each support package to update the content in your workbench to the latest version.

Copying Ready-to Use Content for First Use

Here's what you do to copy the ready-to-use content to your workbench:

1. Navigate to the Data Scrambling workbench from the TDMS work center.
2. Copy the delivered content to your work center by choosing the option *Copy from Template* pushbutton in the Scrambling Overview screen. During the first run, this option copies the objects provided by SAP to the Work Center.
3. You can now see the standard content copied into the work center.

Copying Ready-to-Use Content for Subsequent Use

If you have already copied the standard content into the Scrambling Workbench and you select the Copy from Template pushbutton again, you need to choose an appropriate copy method. The following methods are available to you:

- **Copy Enhanced Scrambling Objects:** Copies an enhancement provided by SAP but not yet available with the customers.
- **Append Scrambling Objects:** Appends the scrambling objects delivered by SAP to the work center.

Choose the desired method and proceed.

7.1.1 Ready-to-Use Standard Content from SAP TDMS

We provide the following content ready for use in data scrambling:

Ready-to-Use Content	Application	Use this to:
Vendor	SAP ERP	Scramble the vendor number and address details
Customer	SAP ERP	Scramble the customer number and address details
Material	SAP ERP	Scramble the material number
Personnel Data	SAP HCM	Scramble the personnel name, address, birth data and other country-specific data
Personal and Address Data	SAP ERP	Scramble personal data like VAT registration number, bank details and address details
Credit card	SAP ERP	Scramble a credit card number
CRM Data	SAP CRM	Scramble the business partner personnel number, name and address
Personnel_Number	SAP_ERP	Scramble personnel numbers

Scrambling Overview					
Assign/Unassign Objects Copy From Template Expand All Collapse All Refresh Search Find					
Scrambling Objects	Transport	Status	Object Type	Solution Category	
▶ Personal Data and Address Data	<input type="checkbox"/>	○○○	Supergroup	SAP_ERP	
▶ Personnel_Number	<input type="checkbox"/>	●○○	Supergroup	SAP_ERP	
▶ Creditcard_Data	<input type="checkbox"/>	○△○	Supergroup	SAP_ERP	
▶ Vendor	<input type="checkbox"/>	○○○	Supergroup	SAP_ERP	
▶ Customer	<input type="checkbox"/>	○△○	Supergroup	SAP_ERP	
▶ Material	<input type="checkbox"/>	○○○	Supergroup	SAP_ERP	
▶ Personnel_Data	<input type="checkbox"/>	●○○	Supergroup	SAP_HCM	
▶ CRM Data	<input type="checkbox"/>	○△○	Supergroup	SAP_CRM	

Figure 7: Standard content available for data scrambling

After you copy the content to the workbench, you can use the scrambling objects in your package. For more information, see the sequence of steps in Chapters 4, 5, and 6.

8 Working with Custom Requirements

If you want to extend the ready-to-use scrambling content provided by SAP TDMS or create new scrambling content for your custom requirements, you can use the Scrambling workbench in the TDMS work center.

8.1 Understanding the Process Flow Graphically

The following set of flowcharts shows you how to proceed systematically with creating scrambling objects, including scrambling rules, and assigning the rules to a TDMS package.

When creating a new rule, you can use an existing solution category or create a new solution category as shown in the flowchart.

Creating a Solution Category

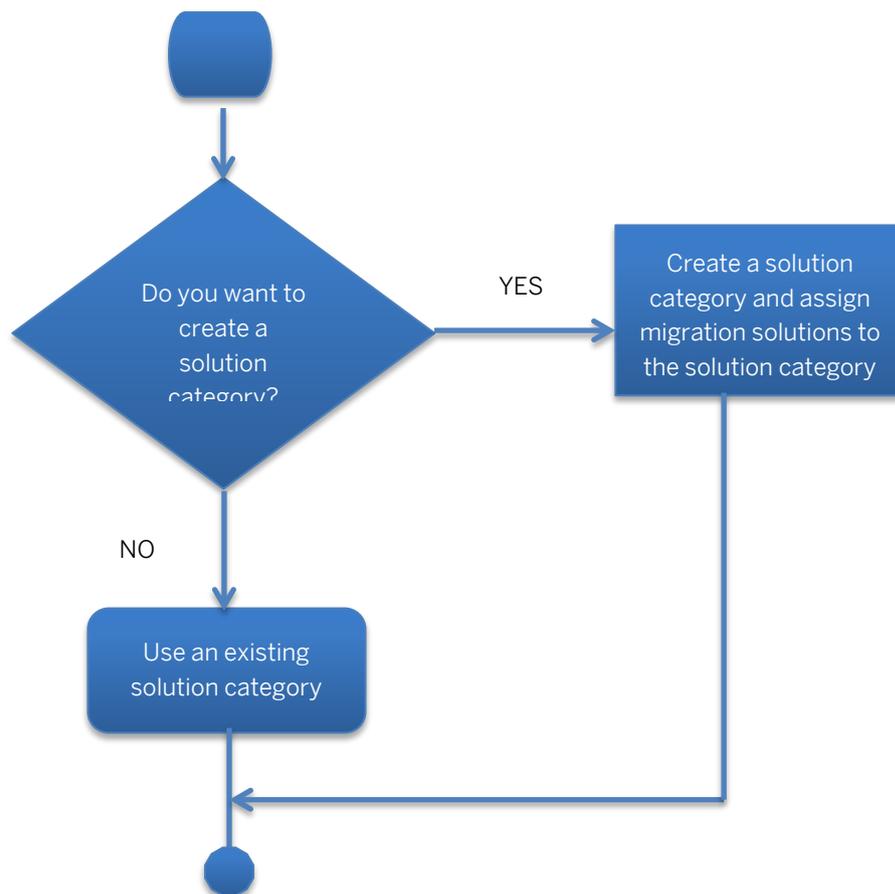


Figure 8: Creating a Solution Category

Creating a Supergroup

When creating a new rule, you can use an existing supergroup or create a new supergroup as shown in the flowchart.

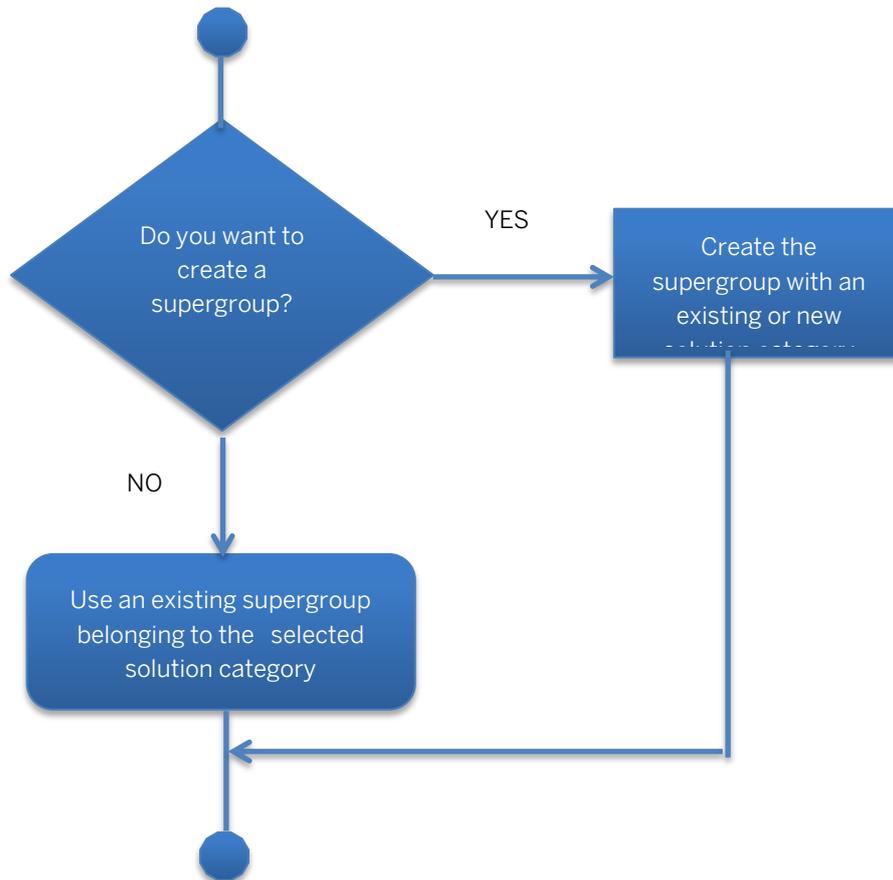


Figure 9: Creating a Supergroup

Creating a Group

When creating a new rule, you can use an existing group or create a new group as shown in the flowchart.

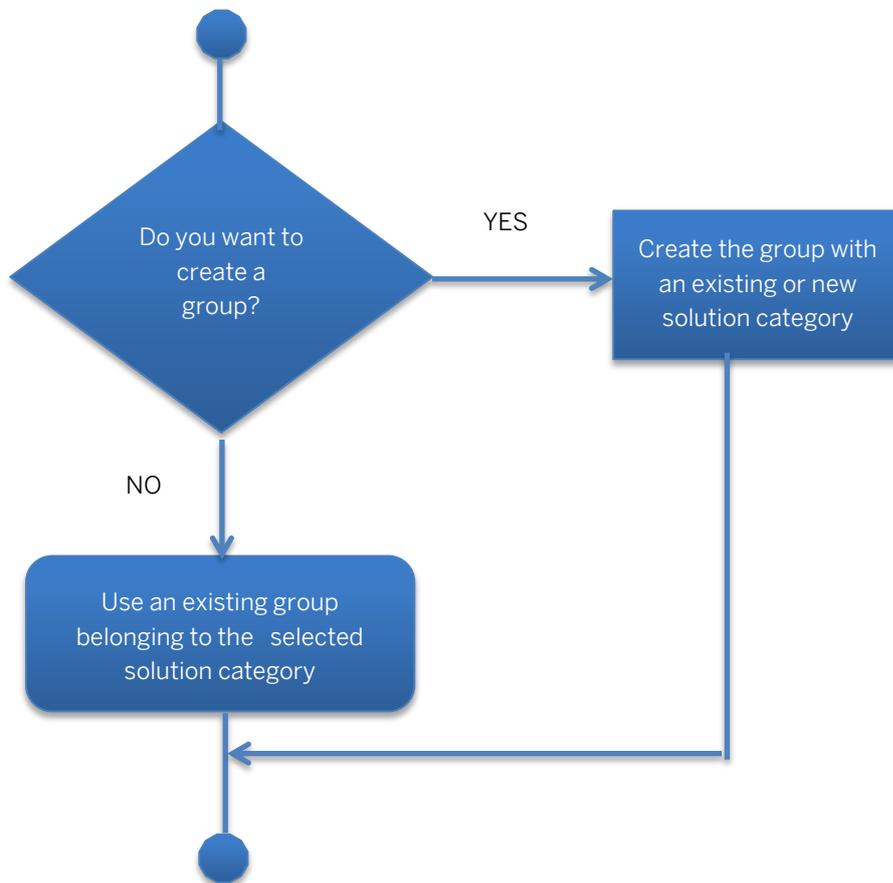


Figure 10: Creating a Group

Creating and Activating a Rule

To realize your custom requirements, you can create a rule or copy and modify an existing rule as shown in the flow chart. As the final step, you need to activate the rule at the project or package level.

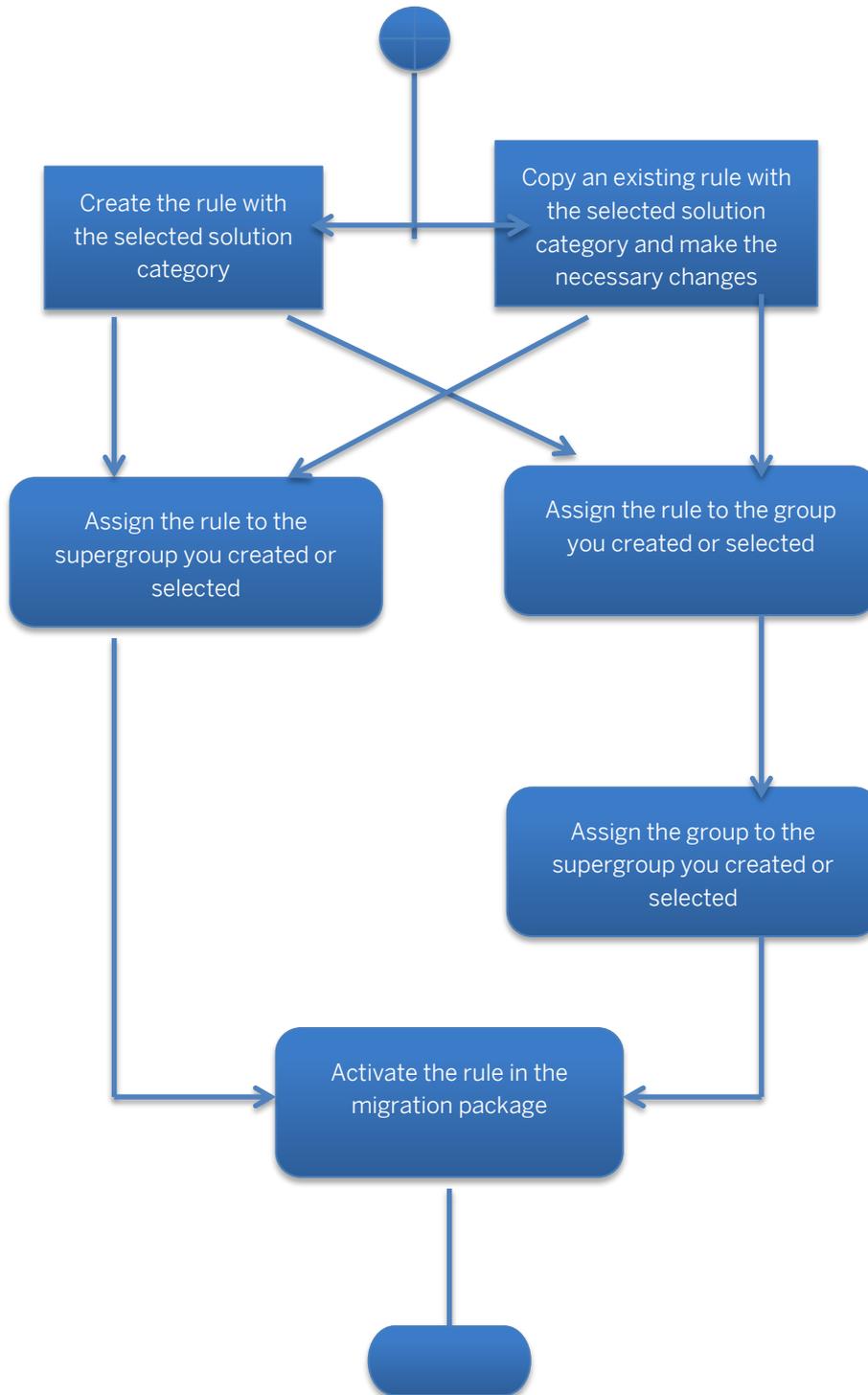


Figure 11: Creating and Activating a Rule

8.2 Working with Multiple Product Levels

Creation

You can create custom scrambling objects at the following levels:

- Scrambling Workbench
- Project
- Package (where you want to activate scrambling)

Copy

Any objects you create at a particular level can be copied to the levels below.

Modification

You can modify the objects at any level.

Activation

You can activate the scrambling objects at the project level or at the package level.

8.3 Working with the User Interface to Create Scrambling Objects

If the SAP provided Standard Content for Scrambling does not meet your business requirements, then you can create your own Scrambling Objects by following the below steps.

1. Navigate to the Data Scrambling Workbench from the TDMS Work center.
2. To start using SAP TDMS Scrambling, copy the SAP standard content to your work center by choosing the option [Copy from Template](#) pushbutton in the Scrambling Overview screen. During the first run, this option copies the objects provided by SAP to the Work Center.
3. Navigate to the Scrambling Groups tab page and create a scrambling supergroup and a scrambling group. If you need more details about creating a scrambling supergroup and group, take a look at Sections 8.4 [Working with the Scrambling Supergroup](#) and Section 8.5 [Working with the Scrambling Group](#).
4. After creating the supergroup and group, create a rule. Navigate to the Scrambling Rules tab page to create a scrambling rule. For details of the step-by-step approach to creating a rule, take a look at Chapter 9 [Working with Scrambling Rules](#).
5. After creating the scrambling rule, assign the scrambling Rule to the Scrambling Supergroup and Scrambling Group. You can do this by choosing Assign on the Scrambling Rule tab page.

Your scrambling objects are now ready for use in the required project or package. To copy your scrambling objects from the work center to the project level or the package level, take a look at Chapter 12 section.

8.4 Working with the Solution Category

8.4.1 Creating a Solution Category

To create a solution category:

1. On the Solution Category tab page, choose the New pushbutton.

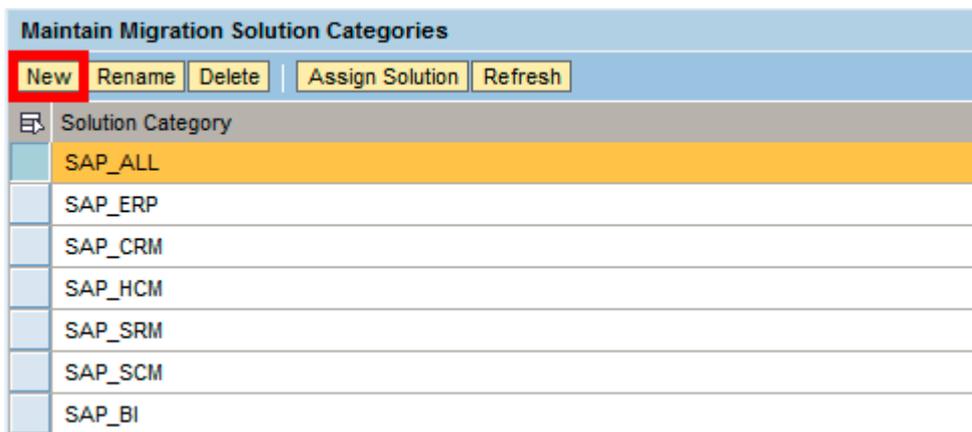


Figure 12: Creating a New Solution Category

The Create a Migration Solution Category dialog box appears.

2. Enter a name for the solution category and save your entry.

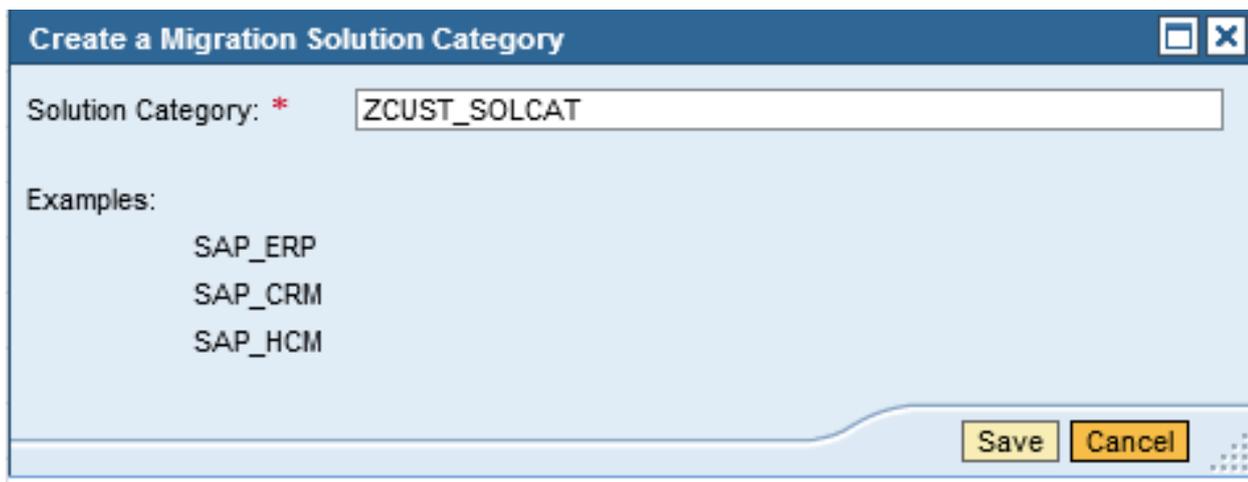


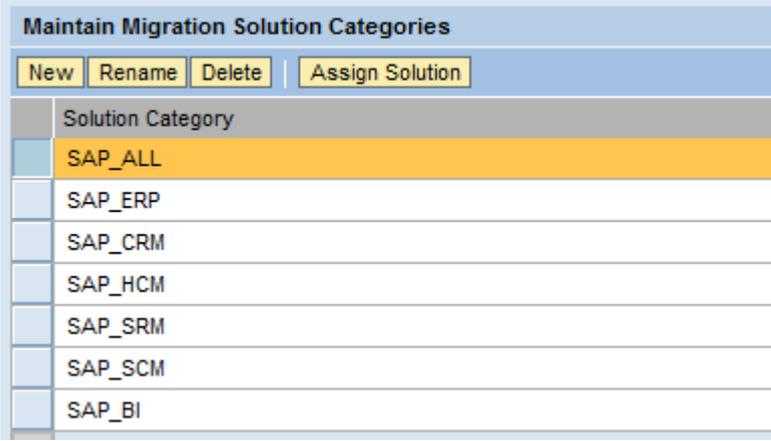
Figure 13: Provide a Name to the Solution Category

8.4.2 Assigning the TDMS Migration Solution to a Solution Category (Required)

If you use an SAP solution category, you can:

- Use the default migration solutions available with the solution category
- Add a migration solution
- Delete a migration solution

SAP provides the following standard solution categories.



Maintain Migration Solution Categories	
New Rename Delete Assign Solution	
Solution Category	
<input checked="" type="checkbox"/>	SAP_ALL
<input type="checkbox"/>	SAP_ERP
<input type="checkbox"/>	SAP_CRM
<input type="checkbox"/>	SAP_HCM
<input type="checkbox"/>	SAP_SRM
<input type="checkbox"/>	SAP_SCM
<input type="checkbox"/>	SAP_BI

Figure 14: Solution Categories Available with SAP TDMS

If you create a new solution category, you are required to assign the appropriate TDMS migration solutions. Here's how you proceed:

1. Select the *Solution Categories* tab page in the Scrambling workbench..

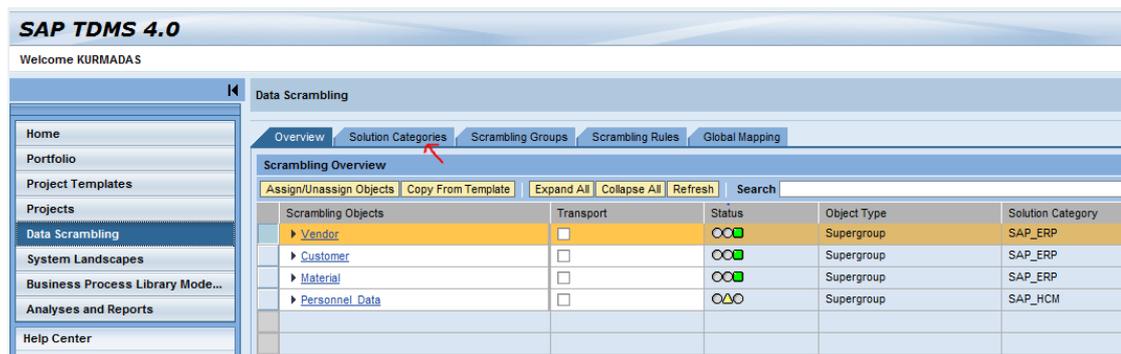


Figure 15: Selecting the Solution Categories tab page

2. Choose the *Assign Solution* pushbutton on the *Solution Categories* tab page to assign new migration solutions to a solution category.

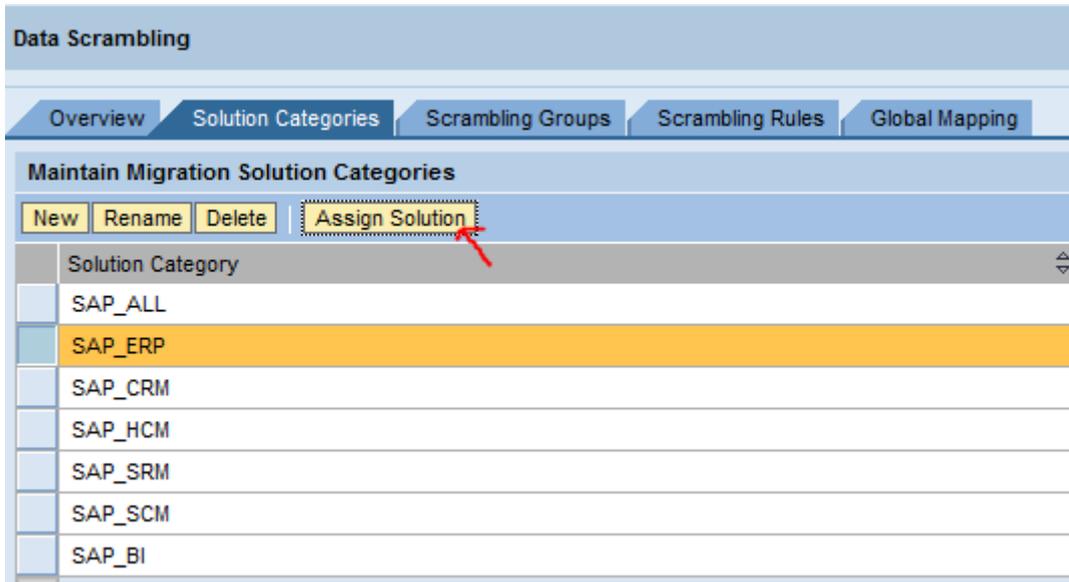


Figure 16: Choosing the Assign Solution Pushbutton

3. Assign a TDMS migration solution to the solution category,

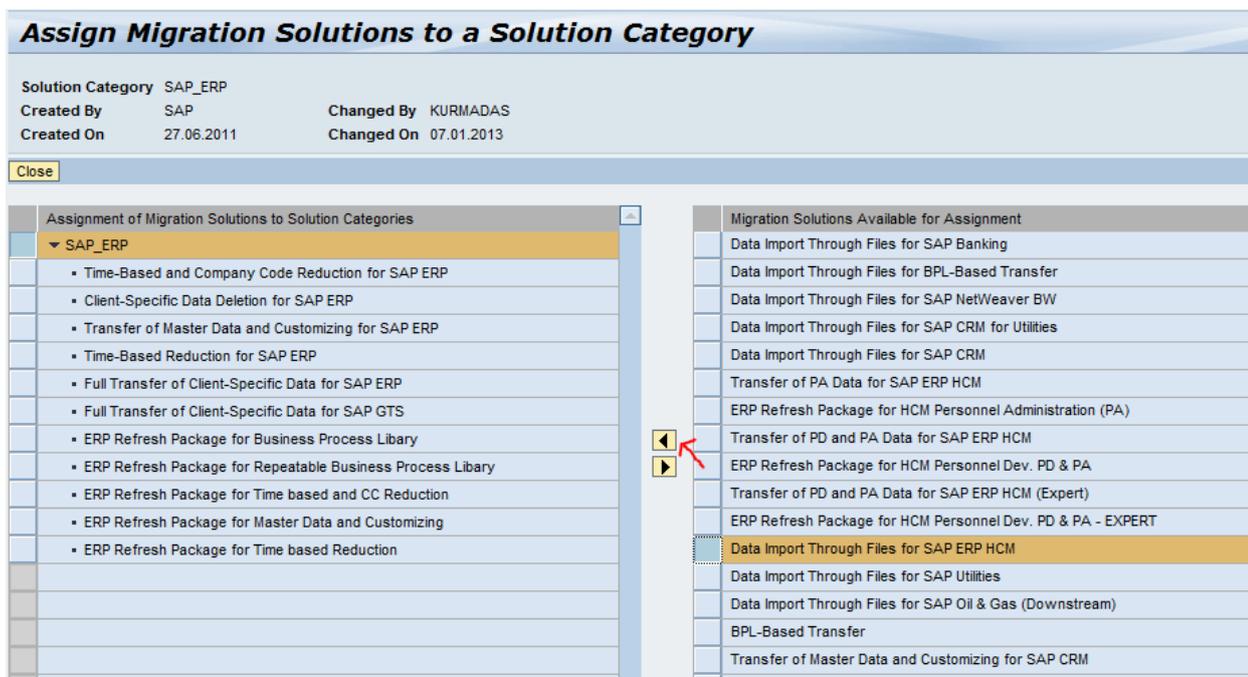


Figure 17: Selecting Migration Solutions for the Solution Category

8.5 Working with the Scrambling Supergroup

Here's what you do to create a scrambling supergroup:

1. Choose *Data Scrambling* in the TDMS work center.

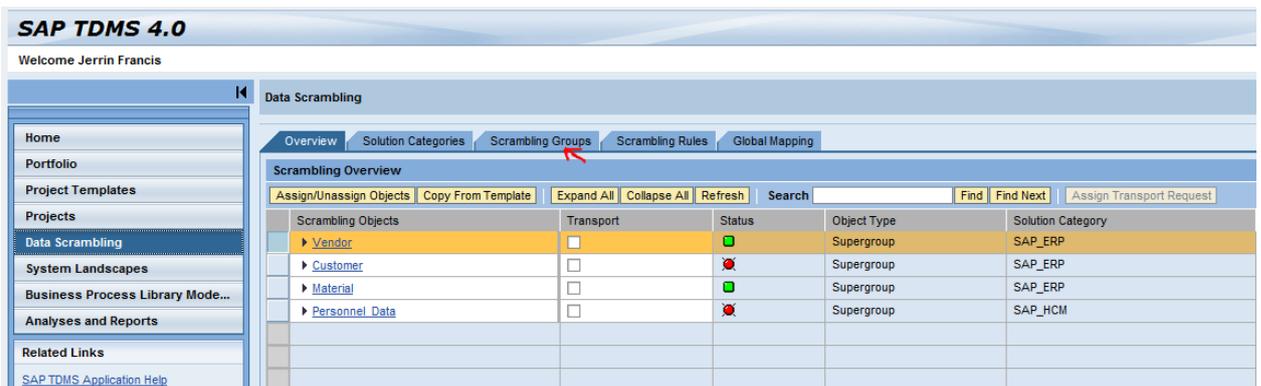


Figure 18: Choosing the Scrambling Groups tab page

2. On the *Scrambling Groups* tab page, choose the *New* pushbutton.

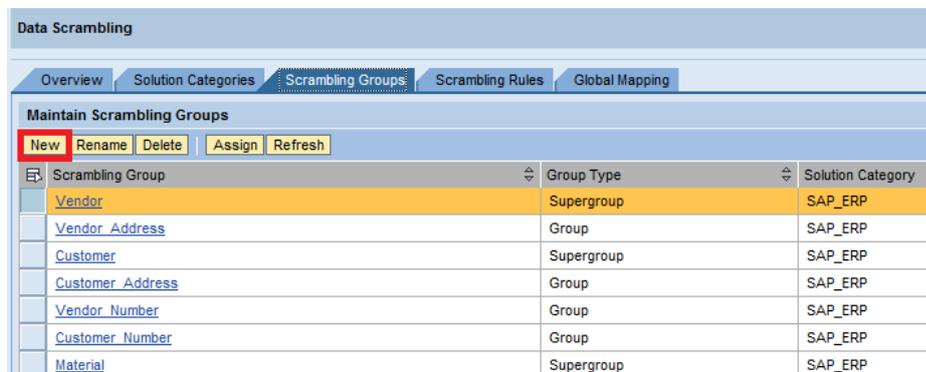


Figure 19: Creating a New Supergroup

3. Enter the name of the Scrambling Supergroup and the solution category. Select the *Supergroup* checkbox.

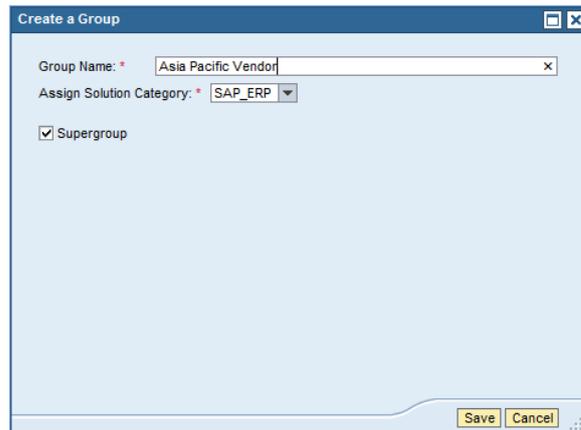


Figure 20: Entering the Supergroup Parameters

8.6 Working with the Scrambling Group

Here's what you do to create a Scrambling Group:

1. Choose *Data Scrambling* in the TDMS work center.

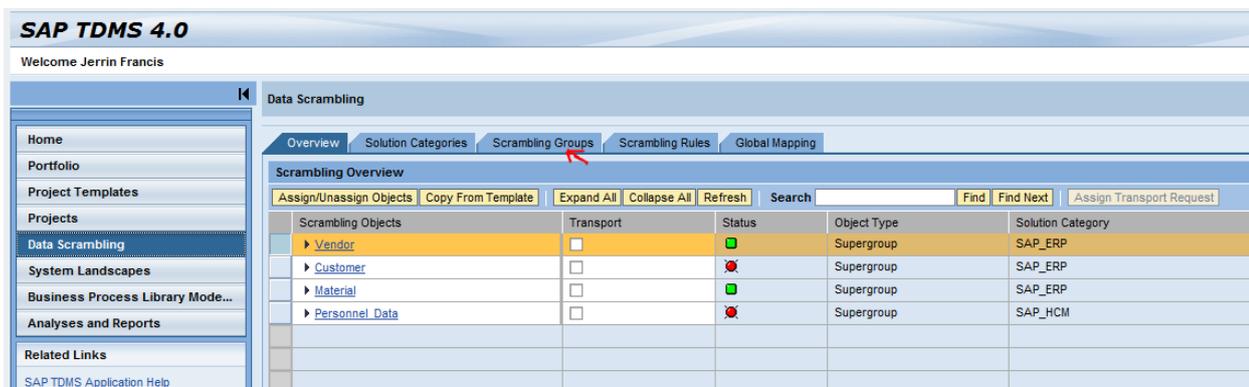


Figure 21: Choosing the Scrambling Groups tab page

2. On the *Scrambling Groups* tab page, choose the *New* pushbutton.

Data Scrambling		
Overview Solution Categories Scrambling Groups Scrambling Rules Global Mapping		
Maintain Scrambling Groups		
New Rename Delete Assign Refresh		
Scrambling Group	Group Type	Solution Category
Vendor	Supergroup	SAP_ERP
Vendor Address	Group	SAP_ERP
Customer	Supergroup	SAP_ERP
Customer Address	Group	SAP_ERP
Vendor Number	Group	SAP_ERP
Customer Number	Group	SAP_ERP
Material	Supergroup	SAP_ERP

Figure 22: Creating a New Group

3. Enter the name of the scrambling group and the solution category. Do not select the *Supergroup* checkbox.

Create a Group [Close] [X]

Group Name: *

Assign Solution Category: *

Supergroup

Figure 23: Entering the Group Parameters

8.7 Copying a Scrambling Rule

When to Use

Do you want to modify standard scrambling rule provided by SAP?

If you want to modify the standard scrambling rule, you can do this by copying the standard rule to your custom namespace and editing the rule.

1. Navigate to the Scrambling Rules tab page.

-
2. Select the rule that you want to enhance and select the Copy Rule pushbutton.
 3. On the dialog box, enter the prefix for the rule name and choose OK.

The rule is now available in your custom namespace and you can further enhance the rule.

9 Working with Scrambling Rules

A scrambling rule contains the complete technical information required to scramble the data.

If you want to enhance any standard scrambling rule provided by SAP, you can do so by copying the standard rule to your custom namespace and enhancing the rule created under the custom namespace.

For details on the steps, take a look at section 8.6 [Copying a Scrambling Rule](#).

A scrambling rule contains the following basic and expert technical information:

Basic Technical Information

- Scrambling type
- Table name/field name
- Primary table field
- Key-sets
- Domains

Expert Technical Information

- Routine names
- Include names
- Condition sets
- Technical identifier
- Scrambling function module

9.1 Creating Rules in the Standard Mode

When creating a scrambling rule, you need to maintain the following attributes:

Attribute	Required	Default Options provided
Rule Name	X	
Solution Category	X	
Scrambling Supergroup		
Scrambling Group		
Scrambling Type	X	
Table Name/Field Name	X	
Primary Table-Field	X	

Attribute	Required	Default Options provided
Routine Name	X	SCRAMBLE_DATA
Key-sets		
Domains		

9.1.1 Step One: Adding a Scrambling Type (Required)

You can use various scrambling types to scramble the data. Here is a list of scrambling types that SAP TDMS provides.

- Delete Value
- Fixed Value
- Manual 1 To 1 Mapping
- Number Conversion
- Random Table with One column
- Value Range Table
- Random Table for Time Periods
- Custom Mapping

The Lookup Table

In all the examples described in this document, the term **lookup table** refers to the table where you enter the mapping values for a scrambling type.

The lookup table appears when you use the following scrambling types:

- Random Table with N Columns
- Manual 1 To 1 Mapping
- Random Table for Time Periods
- Value Range Table

Keep reading for a detailed description of each scrambling object.

9.1.2 Delete Value

When to Use
Do you want the system to delete specific data across all selected records during scrambling?

You can use this scrambling type to clear the value in a particular field for all the records of the tables specified in the rule.

 **Caution**

Do not use this scrambling type to scramble key fields in a table to avoid a possible loss of data.

 **Example**

You want to clear the email address of all employees before data transfer.

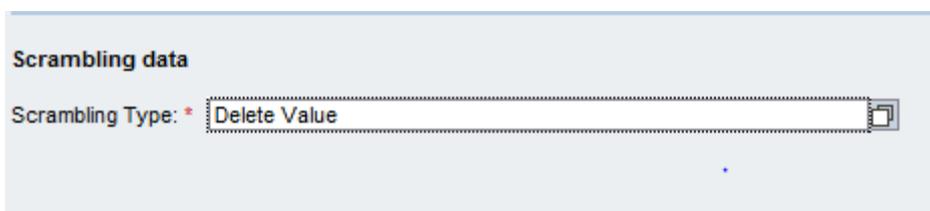


Figure 24: Assign the Delete Value Scrambling Type to a Scrambling Rule

The values in the table before data scrambling:

Employee Number	Email Address
00001	var_dena@webmine.com
00002	woodh_am@mail.com
00003	tob_saleh@myspace.com
00004	leonine-rio@todles.com

The values in the table after data scrambling:

Employee Number	Email Address
00001	
00002	
00003	
00004	

In this example, the email addresses of all employees are cleared from the table before data transfer.

9.1.3 Fixed Value

When to Use

Do you want the system to fix a specific value for all selected records during scrambling?

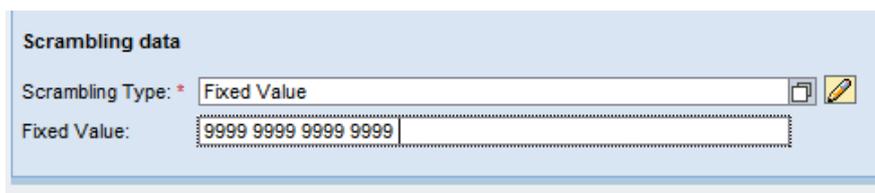
You can use this scrambling type to fix the value of a particular field for all the records of the tables in the rule.

Caution

Do not use this scrambling type to scramble key fields in a table to avoid a possible loss of data.

Example

You want the credit card number to be set to 9999-9999-9999-9999 for all customers.



The screenshot shows a configuration window titled "Scrambling data". It contains two fields: "Scrambling Type:" with a dropdown menu set to "Fixed Value", and "Fixed Value:" with a text input field containing "9999 9999 9999 9999". There are also small icons for copy and edit next to the dropdown.

Figure 25: Assign the Fixed Value Scrambling Type to a Scrambling Rule

The values in the table before scrambling:

Customer Number	Credit Card Number
00001	4129 3456 5412 2345
00002	5674 2890 0312 8257
00003	2095 6899 0023 1412
00004	8479 9031 4567 7828

The values in the table after scrambling:

Customer Number	Credit Card Number
00001	9999 9999 9999 9999
00002	9999 9999 9999 9999
00003	9999 9999 9999 9999
00004	9999 9999 9999 9999

In this example, the credit card number is scrambled to the value 9999 9999 9999 9999 in all the records of the table.

9.1.4 Manual 1 to 1 Mapping

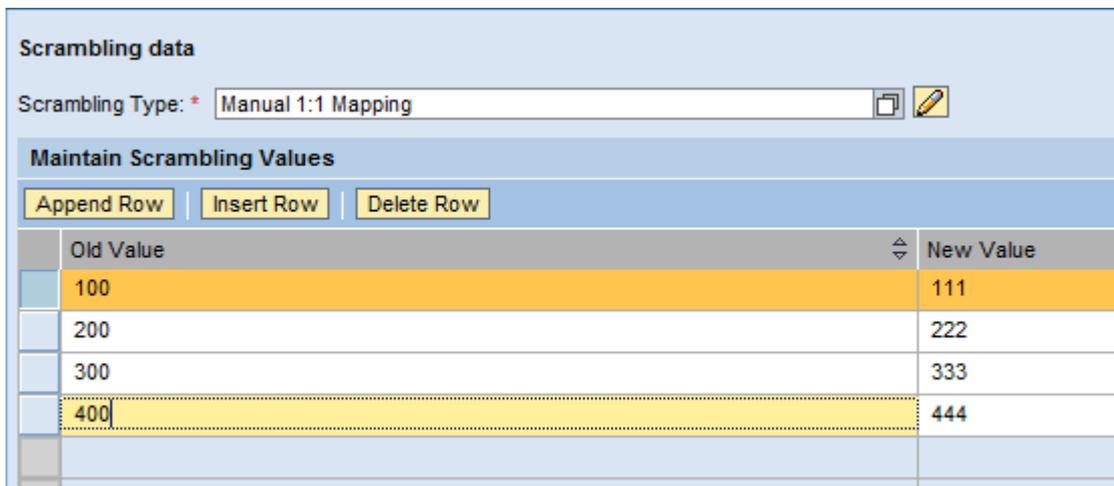
When to Use

Do you want to manually specify fixed values for a set of data across all selected records during scrambling?

You can use this scrambling type to manually map the new value to each old value of a given table/tables in the rule.

Example

You want to scramble the material price from 100 to 111, 200 to 222, 300 to 333, and 400 to 444.



Old Value	New Value
100	111
200	222
300	333
400	444

Figure 26: Assign the Manual 1 to 1 Scrambling Type to a Scrambling Rule

The values in the table before scrambling:

Material Number	Price
00001	100
00002	200
00003	300
00004	400
00005	500

The values in table after scrambling:

Material Number	Price
00001	111

Material Number	Price
00002	222
00003	333
00004	444
00005	500

In this example, the material prices are scrambled as mapped in the lookup table of the rule. Notice that material 00005 is not scrambled because the lookup table does not have a new value for the price 500.

1 Note

In a lookup table where the mapping values are in the lookup table format, you can also provide the inputs in the form of a .CSV file.

9.1.5 Number Conversion

When to Use

Do you want the system to scramble a set of numbers to another set of unique numbers where you specify the starting number?

You can use this scrambling type to scramble table records with a new unique number for each unique old value of a field.

This scrambling type is ideal for the scrambling of key fields such as customer number, vendor number, or employee number.

Example

The customer numbers range from 1 to 1000 in a table and you want to scramble these numbers to a new number range starting with the value 10000.

Scrambling data

Scrambling Type: *  

Number Conversion Starting Value:

Figure 27: Assign the Number Conversion Scrambling Type to a Scrambling Rule

The values in the table before scrambling:

Customer Number
00001

Customer Number
00002
00003
00004

The values in the table after scrambling:

Customer Number
10096
10134
10003
10872

In this example, the new customer numbers generated are unique and random in the range 10000 to 11000 for every old customer number identified in the table. The end value of the range is dynamically determined by the Scrambling platform based on the number of records relevant for scrambling.

9.1.6 Random Table with One Column

When to Use

Do you want the system to scramble data randomly from a set of values you specify?

You can use this scrambling type to assign random values from a lookup table assigned for all the records of a table or tables given in a rule.

Caution

Do not use this scrambling type to scramble key fields in a table to avoid a possible loss of data.

Example

You want to scramble the 100,000 employee names in a table by providing a few lookup values.

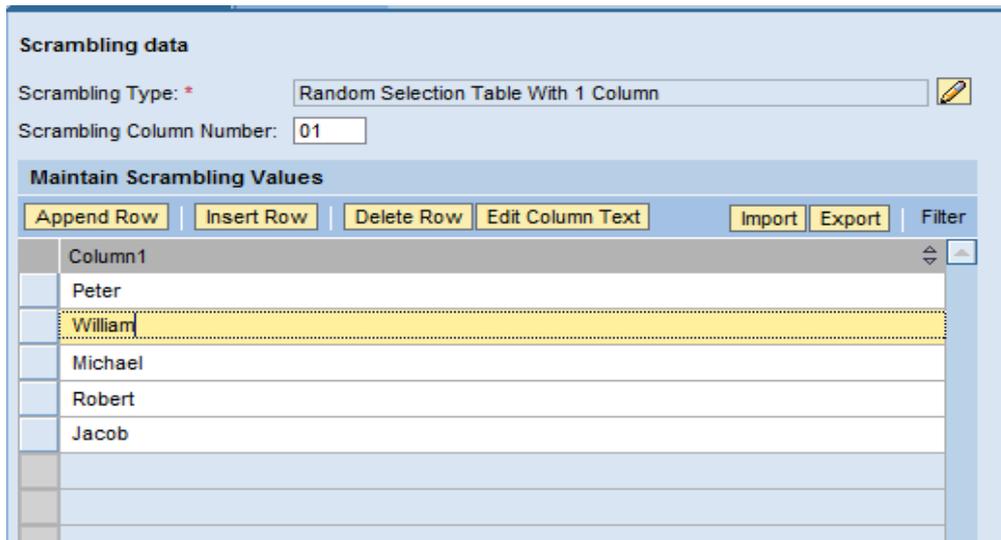


Figure 28: Assign the Random Table with One Column Scrambling Type to a Scrambling Rule

The values in the table before scrambling:

Employee Number	First Name
00001	Martin
00002	David
00003	Alex
00004	Henry

The values in the table after scrambling:

Employee Number	First Name
00001	William
00002	Robert
00003	William
00004	Jacob

In this example, employee first names are scrambled in all the records of the table from the lookup values provided in the rule.

1 Note

A few names could repeat several times since the lookup set is smaller than the actual number of records scrambled.

9.1.7 Value Range Table

When to Use

Do you want the system to scramble data randomly from a range of values you specify?

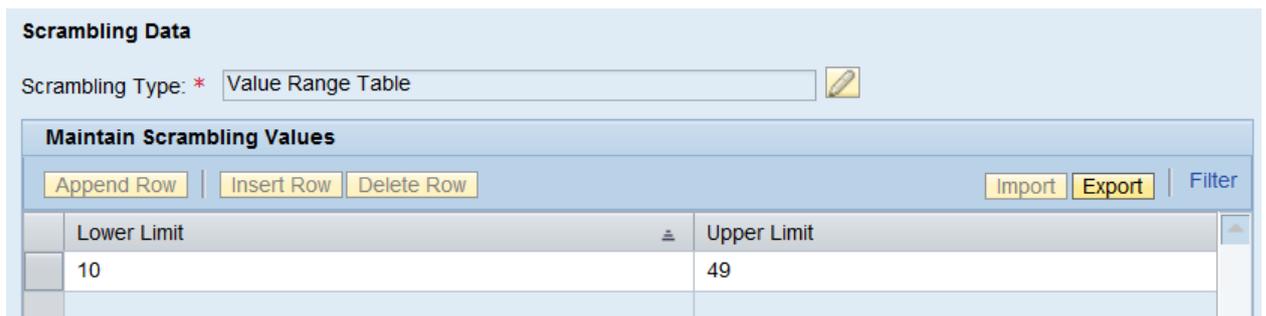
You can use this scrambling type to scramble a field value randomly within a value range.

Caution

Do not use this scrambling type to scramble key fields in a table to avoid a possible loss of data.

Example

You want to scramble the commute distance data (in kilometers) for all employees and you want the new values to range from 10 to 49.



The screenshot shows the 'Scrambling Data' configuration window. The 'Scrambling Type' is set to 'Value Range Table'. Below this is a table titled 'Maintain Scrambling Values' with columns for 'Lower Limit' and 'Upper Limit'. The current values are 10 and 49 respectively. The interface includes buttons for 'Append Row', 'Insert Row', 'Delete Row', 'Import', 'Export', and 'Filter'.

Figure 29: Assign a Value Range Table Scrambling Type to a Scrambling Rule

Values in the table before scrambling:

Employee Number	Commute Distance in Kilometers
00001	800
00002	300
00003	500
00004	600

Values in the table after scrambling:

Employee Number	Commute Distance in Kilometers
00001	10

Employee Number	Commute Distance in Kilometers
00002	5
00003	25
00004	20

In this example, the new values assigned after scrambling are randomly generated during conversion by scrambling programs from the range provided in the rule.

1 Note

The same value could be generated more than once by the random number generator. Hence, this scrambling type must not be used for the scrambling of key fields as it could lead to a duplicate key error.

9.1.8 Random Table for Time Periods

When to Use

Do you want the system to scramble dates randomly from a time range you specify?

You can use this scrambling type to scramble date fields randomly with the new dates in a specified time range.

 **Caution**

Do not use this scrambling type to scramble key fields in a table to avoid a possible loss of data.

 **Example**

You want to scramble the date of birth of employees with new dates ranging from Jan 1, 1970 to Jan 1, 1990.

Scrambling data

Scrambling Type: *  

Maintain Scrambling Values

	From Day	To Day	From Month	To Month	From Year	To Year
	1	28	1	12	1970	1990

Figure 30: Assign the Random Table for Time Periods Scrambling Type to a Scrambling Rule

Values in the table before scrambling:

Employee Number	Birth Date
00001	19841112
00002	19770410
00003	19710702
00004	19830118

Values in table after scrambling:

Employee Number	Birth Date
00001	19790206
00002	19891228
00003	19820712
00004	19720529

Here, the new birth dates are randomly generated from the date ranges provided in the rule.

1 Note

Leave the From value and the To value blank if you do not want to scramble any one of the following fields: Day, Month or Year.

If you want to scramble the day and the month and leave the year unscrambled, maintain the lookup values as shown below.

Scrambling data

Scrambling Type: * Random Table for Time Periods

Maintain Scrambling Values

Append Row | Insert Row | Delete Row | Import | Export | Filter

	From Day	To Day	From Month	To Month	From Year	To Year
	1	28	1	12		

Figure 31: Scrambling the Day and Month but Not the Year

9.1.9 Custom Mapping

When to Use
Do you want the system to scramble data using your custom logic?

You can use this scrambling type when none of the standard scrambling types described above meets your requirements. You can write the ABAP code with the logic you want to scramble in the scrambling routine and assign it to the rule.

By default, SAP assigns the routine SCRAMBLE_DATA for simple rules. Such routines can be replaced or overridden by custom coded routines and assigned to the table fields of the scrambling rule.

Always create the custom routines with this interface:

USING p_rule
CHANGING p_value
 p_changed

Example

```

*-----
*&      Form  example_routine
*&-----
*      text
*-----
*      -->P_RULE      text
*      -->P_VALUE     text
*      -->P_CHANGED  text
*-----
FORM example_routine USING p_rule
                        CHANGING p_value
                        p_changed.
ENDFORM.                "example routine
  
```

Figure 32: A sample custom code

In this section, we will look at the different ways of developing ABAP code for custom developments.

- Custom Code for Scrambled Data
- Custom Code for Data Before Scrambling

Custom Code for Scrambled Data

In certain cases, you may want to change the data that was already scrambled using standard scrambling types.

To meet this requirement, you can call the SAP delivered standard routine SCRAMBLE_DATA in your custom routines to obtain the scrambled value. You can then modify the scrambled value according to your requirements.

Example

You want to scramble an employee name in two tables: PA0002-VORNA and PA0001-VORNA. You want the new name assigned for PA0002-VORNA to be in title case and the new name assigned for PA0001-ENAME to be in the upper case.

Add custom code to modify the already-scrambled name in title case to upper case.

Proceed as follows:

1. Maintain the lookup values in the rule in title case.

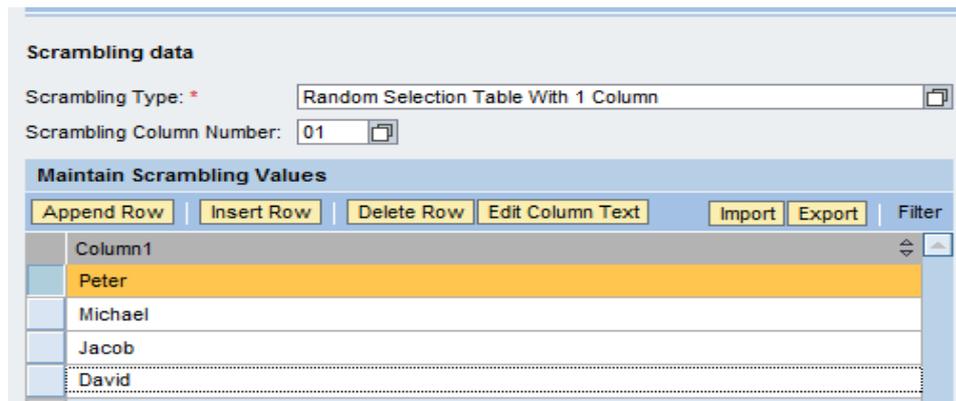


Figure 33: Maintaining Lookup values for the scrambling rule

2. Replace the routine name with a custom routine name for those table-fields where you want the scrambled output to be in upper case. Hence, we change the routine name for PA0001-ENAME to ZDATA_UPPERCASE and add custom code to it.

Table-Fields					
Append Row Insert Row Delete Row Keysets					
Table Name	Field Name	Primary Table-field	Routine Name	Is a Cluster Table	
PA0002	VORNA	<input checked="" type="checkbox"/>	SCRAMBLE_DATA	<input type="checkbox"/>	
PA0001	ENAME	<input type="checkbox"/>	ZDATA_UPPERCASE	<input type="checkbox"/>	

Figure 34: Assigning Custom routine to table-field

3. Add the custom include name in the rule where you have created your custom routines.

Append Row Insert Row Delete Row	
Program	Include Type
CNV_TDMS_SCR_70010_FORMS	P
CNV_TDMS_SCR_70010_TOP	D
ZCUSTOM_INCLUDE	P

Figure 35: Assigning Custom include to scrambling rule

4. Create a new routine ZDATA_UPPERCASE and call the standard routine SCRAMBLE_DATA to get the scrambled value in title case.
5. After you get the value, convert the data (P_VALUE) to the upper case as shown below.

```

L *-----
FORM zdata_uppercase USING    p_rule
                        CHANGING p_value
                        p_changed.

* call the standard routine to get the newly assigned
* by scrambling type 'random table with 1 column'
PERFORM scramble_data USING    p_value
                        CHANGING p_value
                        p_changed.

* Here P_value is the scrambled value
TRANSLATE p_value TO UPPER CASE.
ENDFORM.                    "zdata_uppercase

```

Figure 36: Custom routine code

After scrambling is complete, the names in the PA0002 table are in the title case and the names in the PA0001 table in the upper case.

6. You can similarly modify the already-scrambled value according to your requirements as shown in the example above.

Custom Code for Data Before Scrambling

You can use this scrambling type when none of the standard scrambling types meets your requirements. You can use the scrambling type Custom Mapping and assign custom routines to all the table fields of a rule.

Example

You want to multiply the currency value of the salary of all employees in a custom table by a factor of 5.

Here's what to do:

1. Assign the scrambling type *Custom Mapping* to the rule.



Figure 37: Assigning No Mapping scrambling type to scrambling rule

2. Add the custom routine name MULTIPLY_CURRENCY to the table field in the rule.



Figure 38: Assigning a custom routine to the table-field

3. Add the custom include name to the rule where you coded your custom routines.

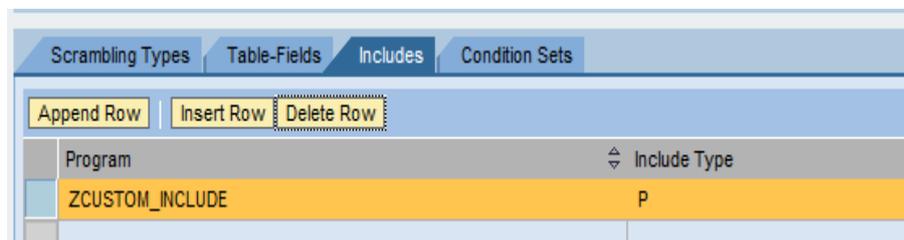


Figure 39: Assigning a custom include to the scrambling rule

4. Create a new routine `MULTIPLY_CURRENCY` with the standard interface. The parameter `P_VALUE` contains the value before scrambling. You can modify `P_VALUE` according to your requirements.
5. Add the code (shown below) in the custom routine to multiply the currency value by a factor of 5.

```

*&-----*
*&      Form multiply_currency
*&-----*
*      text
*-----*
*      -->P_RULE      text
*      -->P_VALUE     text
*      -->P_CHANGED   text
*-----*
FORM multiply_currency USING p_rule
                        CHANGING p_value
                        p_changed.

      p_value = p_value * 5.
      p_changed = 'X'.

ENDFORM.              "multiply_currency

```

Figure 40: Custom routine code

We have learned how to write custom code in the example above.

Next, we will explore how to code conditions when using the scrambling type Custom Mapping since we cannot use the condition-sets of the rule for the *Custom Mapping* scrambling type.

Example

You want to multiply the currency value by a factor of 5 for the salary of employees whose employee number occurs between 100000 and 101000.

Here's what to do:

1. Assign the Custom Mapping scrambling type to the rule.



Figure 41: Assigning the scrambling type No Mapping to the rule

2. Add the custom routine name MULTIPLY_CURRENCY to the table field.



Figure 42: Assigning Custom routine to table-field

3. Add the custom include name to the rule where you coded your custom routines.

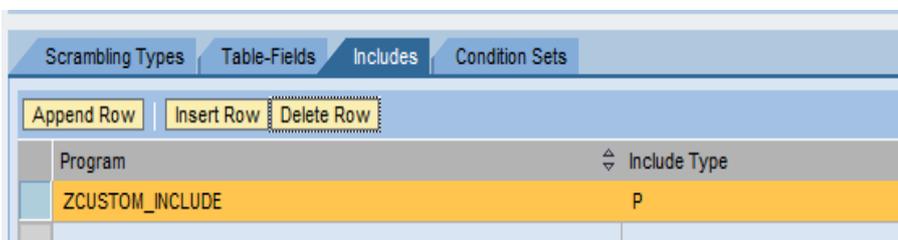


Figure 43: Assigning Custom include to scrambling rule

4. Add the code (shown below) in the custom routine to multiply the currency value by a factor of 5 for the salary of only those employees whose numbers occur between 100000 and 101000.

```

FORM multiply_currency USING p_rule
                        CHANGING p_value
                        p_changed.

FIELD-SYMBOLS: <lv_empno> TYPE ANY.

ASSIGN COMPONENT 'EMPNO' OF STRUCTURE gd_original
                TO <lv_empno>.

IF sy-subrc EQ 0 AND
   <lv_empno> BETWEEN '100000' AND '101000'.

   p_value = p_value * 5.
   p_changed = 'X'.

ENDIF.

ENDFORM.                                "multiply_currency

```

Figure 44: Custom routine code

Global Variables for Custom Coding

We provide a few global variables that can be used in custom routine code.

The variable `GD_ORIGINAL` contains the complete row of the current record of the table being scrambled. (In this example, `GD_ORIGINAL` contains the current record of table `PA0008`).

The following global variables can be used in custom coding:

- `GC_PACKID`: Contains the TDMS package number
- `GC_TABNAME`: Contains the table name.
- `GD_FIELDNAME`: Contains the field name.
- `GD_ORIGINAL`: Contains the complete row of the current record of the table (only for transparent tables).
- `GD_KEYTAB`: Contains the complete row of the current record of the table (only for cluster tables).

i Note

Always try to use the global variables with field symbols, as they are not statically declared.

9.1.10 Step Two: Adding Table-Field Combinations (Required)

This section describes how you can add table-field combinations to the scrambling rule.

You can enter the list of table names and the corresponding fields that you want to scramble on the [Table-Fields](#) tab page..

The table-field combination refers to the field that has to be scrambled in a particular table.

Example

For example, if you want to scramble the KUNNR (Customer Number) field in the KNA1 (Customer Master) table, the Table-Field combination is KNA1-KUNNR.

When you specify on the *Table-Field* tab page that KUNNR must be scrambled in the table KNA1 based on the conversion logic defined in the rule, the Scrambling workbench sets these table-field combinations as conversion relevant tables and conversion relevant fields for later processing.

In the case of cluster tables, you can enter the table name and field name in the following format:

Table name: 'Cluster table name' - 'RELID'

Field name: 'Structure/component of the RELID' - 'Field to be converted of the component'

Example

You want to scramble the first name of an American employee, and create a scrambling rule using the following values:

Table Name: **PCL2-RU**

Field Name: **NAME-VORNA**

Is Cluster Table field: **Selected**

Table Name	Field Name	Primary Table-field	Routine Name	Is a Cluster Table	Do Not Scramble	Keysets
PCL2-RU	NAME-VORNA	<input type="checkbox"/>	SCRAMBLE_DATA_CLUSTER	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

Figure 45: Assigning a Table-Field Combination for Scrambling

Note

Do not set a cluster table as a primary table field, which is always required to be a transparent table.

To maintain multiple table-fields for a rule:

1. Choose the *Export* pushbutton to export the data in the table to a .CSV file.
2. Maintain additional table-field entries in the file.
3. Import the new set of table-fields from the file to the *Table-Fields* tab page belonging to the scrambling rule.

Table Name	Field Name	Primary Table-field	Keysets
ADRC	NAME1	<input checked="" type="checkbox"/>	
LFA1	MCOD1	<input type="checkbox"/>	
LFA1	NAME1	<input type="checkbox"/>	

Figure 46: The *Table-Fields Combination* Tab Page

9.1.11 Step Three: Setting the Primary Table-Field (Required)

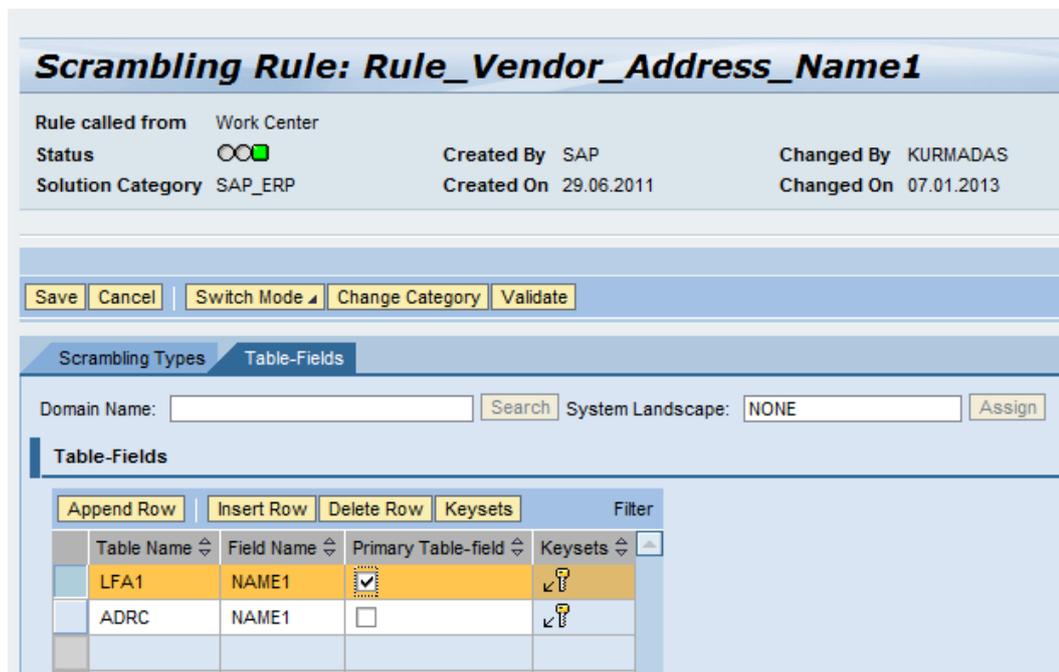
When to Use

Do you want to set the Primary table-field to activate the scrambling rule? (Required)

The primary table-field combination indicates a combination of the master table (that contains the superset of field values specified in table-field combination) and the field that needs to be scrambled. The primary table field is a required attribute of a rule. The Scrambling platform assigns new values for all the records of the primary table field selected for the rule. The platform uses these results to scramble the other table of the rule.

Example

In the rule described, LFA1-NAME1 is set as the primary table-field.



Scrambling Rule: Rule_Vendor_Address_Name1

Rule called from: Work Center
 Status: ○○●
 Solution Category: SAP_ERP

Created By: SAP
 Created On: 29.06.2011

Changed By: KURMADAS
 Changed On: 07.01.2013

Buttons: Save, Cancel, Switch Mode, Change Category, Validate

Scrambling Types: Table-Fields

Domain Name: Search System Landscape: NONE Assign

Table Name	Field Name	Primary Table-field	Keysets
LFA1	NAME1	<input checked="" type="checkbox"/>	
ADRC	NAME1	<input type="checkbox"/>	

Figure 47: Selecting a Table-Field Combination as a Primary Table-Field for Scrambling

Assume the following data in the LFA1 and ADRC tables before scrambling:

LFA1

LIFNR	NAME1
1000	SAP AG
1001	Microsoft
1002	Google

ADRC

ADDRNUMBER	NAME1
00001	SAP AG
00002	Microsoft
00003	Google
00004	Audi
00005	BMW

The Scrambling platform locates a new value for every record of the LFA1 table since it was set as a primary table field.

The following data shows the post-scrambling situation for both tables:

LFA1

LIFNR	NAME1
1000	Abc1
1001	Group1
1002	Vanilla1

In this rule, LFA1-NAME1 is the primary table field. All the records in LFA1 are scrambled as the Scrambling platform finds new value for the primary table.

ADRC:

ADDRNUMBER	NAME1
00001	Abc1
00002	Group1
00003	Vanilla1
00004	Audi
00005	BMW

In the ADRC table, the last two records are not scrambled as LFA1 does not have these records and the Scrambling platform did not find the result set to be able to scramble the values.

Note

Always make sure to set the master table that contains all the records as the primary table to enable a complete scrambling of records in all the tables for the rule.

9.1.12 Step Four: Using a Key-Set

When to Use

Do you want the system to scramble data consistently across all the tables in a particular SAP application such as SAP ERP or SAP SCM?

You use key sets to uniquely identify a record for scrambling in a table. Key sets are a single field or a combination of fields by which a record can be uniquely identified in a table.

You would require key sets to:

- Consistently scramble data across table and fields
- Modify the new value assignment based on particular fields of a table

While working with multiple tables in a rule, the key set defined across the tables must contain the same set of fields.

While using global mapping for scrambling, you have to define the key set in the same sequence for all rules that use the same global mapping.

9.1.12.1 Modifying a New Value Assignment Based on the Key-Set

When to Use

Do you want to define mapping values for a key field or a set of key fields in a primary table?

You use the key set to modify the new values assignment by identifying the table record uniquely with a different set of key fields.

Example

Here you can see how key sets are used to scramble the same old value differently by assigning a variety of key sets.

Let's assume you are scrambling the first name of an employee in table PA0002 (sample non-scrambled data of PA0002 table shown below).

Employee Number	Subtype	First Name	Last Name
000001	10	Peter	Spierig
000002	10	Peter	Farely
000003	11	Peter	Jackson

Let's see how the Scrambling platform assigns the new value based on the different key sets when *Random table with 1 column* is assigned as a scrambling type. The results after scrambling are shown below:

Case 1: When no key set is assigned

Employee Number	Subtype	First Name	Last Name
000001	10	Martin	Spierig
000002	10	Martin	Farely
000003	11	Martin	Jackson

Here, all Peters are scrambled to Martin as no key set is assigned. A new value is assigned to every unique old value. Since all old first names are *Peter*, all new first names are scrambled to *Martin*.

Case 2: When the key set is the Subtype field

Employee Number	Subtype	First Name	Last Name
000001	10	Michael	Spierig
000002	10	Michael	Farely
000003	11	Roger	Jackson

The Scrambling platform scrambles the subtypes 10 to Michael and the subtypes 11 to Roger. Hence, the old value Peter is not of consequence here. The platform takes into account the subtype and assigns a new value for every unique subtype value.

Case 3: When the key set is the Employee Number field

Employee Number	Subtype	First Name	Last Name
000001	10	Martin	Spierig
000002	10	Roger	Farely
000003	11	Michael	Jackson

Here, the Scrambling platform assigns a new value based on the employee number, so every instance of Peter is scrambled to a new value. This is because the employee numbers are unique for all three records.

9.1.13 Step Five: Using a Domain

When to Use
Do you want to scramble all table-field combinations belonging to a domain?
Do you want the system to identify the table-fields belonging to specific domains for scrambling?

In certain scenarios, you may want to scramble all the tables and fields that belong to specific domains. In such a case, it can be a tedious exercise to manually enter all the table names and field names. Instead you can use the Domains option available on the Scrambling interface.

The Scrambling platform identifies all the tables and fields of the domains you maintain and includes them for scrambling.

You can also provide the roll name (data element) to restrict the tables and fields of a domain to specific data elements.

Example

You want to scramble the vendor numbers of all the tables that use the domain LIFNR.

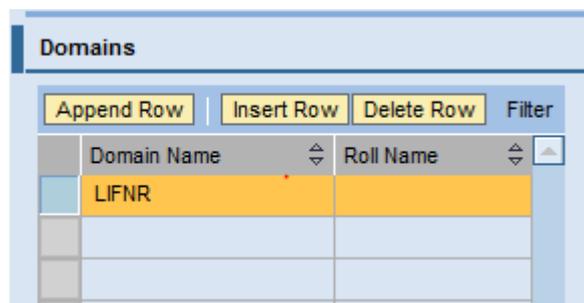


Figure 48: Assigning Domain to a Scrambling Rule

9.2 Additional Features in the Expert Mode

9.2.1 Step Six: Assigning Routine Names

When to Use

Do you want to modify the default scrambling routine to a different standard routine or a custom routine?

The routine contains the logic (ABAP code) to scramble the table-fields provided in the rule. You need to assign routine name for every table-field in the rule.

SAP provides you the following standard routines:

- **SCRAMBLE_DATA**: Use only for transparent tables.
- **SCRAMBLE_DATA_UPPERCASE**: Use only for transparent tables when you want scrambled data in the upper case.
- **SCRAMBLE_DATA_CLUSTER**: Use only for cluster tables.
- **SCRAMBLE_NON_INITIAL**: Use this subroutine instead of the routine SCRAMBLE_DATA to only scramble the data that is not Initial. (Blank fields are not assigned new values.)

Table Name	Field Name	Primary Table-field	Routine Name
P01T_LST	VORNA	<input type="checkbox"/>	SCRAMBLE_DATA
PA0002	VNAMC	<input type="checkbox"/>	SCRAMBLE_TO_UPPERCASE
PA0002	VORNA	<input checked="" type="checkbox"/>	SCRAMBLE_DATA
PA0021	FAVOR	<input type="checkbox"/>	SCRAMBLE_DATA
PA0111	GLVOR	<input type="checkbox"/>	SCRAMBLE_DATA
PA0148	FAVOR	<input type="checkbox"/>	SCRAMBLE_DATA
PCL2-RD	NAME-VORNA	<input type="checkbox"/>	SCRAMBLE_DATA_CLUSTER
PCL2-RJ	NAME-VORNA	<input type="checkbox"/>	SCRAMBLE_DATA_CLUSTER
PCL2-RU	NAME-VORNA	<input type="checkbox"/>	SCRAMBLE_DATA_CLUSTER

Figure 49: Assigning Subroutine to a Scrambling Rule

Points to Remember

- The routine name **SCRAMBLE_DATA** is automatically assigned to every table-field of a rule in the standard mode. You can modify them only in the expert mode.
- For custom requirements, code your logic in custom subroutines and assign them to the rule.

9.2.2 Step Seven: Assigning Includes

You need to code all the custom routines assigned to a rule in an include program.

You can assign the following types of includes to a rule:

- **P**: Includes with FORM subroutines
- **D**: Includes with global data declarations

Scrambling Types	Table-Fields	Includes	Condition Sets
Append Row Insert Row Delete Row			
Program			Include Type
CNV_TDMS_SCR_70010_FORMS			P
CNV_TDMS_SCR_70010_TOP			D
ZCUSTOM_INCLUDE			P

Figure 50: Assigning Includes to a Scrambling Rule

The standard includes are automatically assigned to the scrambling rule.

Standard Includes

SAP provides the following standard includes:

Includes for transparent tables, when the routine SCRAMBLE_DATA is used:

- **CNV_TDMS_SCR_70010_FORMS** (Include type 'P')
- **CNV_TDMS_SCR_70010_TOP** (Include type 'D')

Includes for cluster tables, when the routine **SCRAMBLE_CLUSTER_DATA** is used,

- **CNV_TDMS_SCR_HCM_FORMS** (Include type 'P')
- **CNV_TDMS_SCR_HCM_TOP** (Include type 'D')

Points to Remember

- Includes can only be maintained in a rule in the Expert mode
- In the standard mode, the following includes are automatically assigned to a rule:
 - **CNV_TDMS_SCR_70010_FORMS**
 - **CNV_TDMS_SCR_70010_TOP**

9.2.3 Step Eight: Using the Identifier and Scrambling Function Modules

When to Use

Do you want the system to filter the data for mapping from a multi-column random table based on a key field defined in the mapping table?

You can use identifiers and the scrambling function modules to handle special case scrambling scenarios that cannot be handled by a basic rule.

You can use:

- Identifiers to tag a rule by a unique name for special case handling.
- Scrambling function modules to override the normal mapping logic with custom logic for the special case scrambling scenarios.

SAP has used identifiers and scrambling function modules in Vendor/Customer address and HCM data scrambling rules.

Address Scrambling

In address scrambling, only scramble the address data of vendor and customer objects in the address tables. SAP identifies these objects through the following identifiers and scrambling function modules in the address rules:

Identifiers: VENDNO, CUSTNO

Scrambling function module: CNV_TDMS_SCR_PRESEL_ADDRESS

Scrambling Rule: Rule_Vendor_Address_Name1

Rule called from: Work Center
 Status: ●
 Solution Category: SAP_ERP

Created By: SAP
 Created On: 29.06.2011

Changed By: KURMADAS
 Changed On: 07.01.2013

Scrambling data

Scrambling Type: *

Scrambling Column Number:

Scrambling Function Module:

Identifier:

Maintain Scrambling Values

Address Object	Name1	Name2	Name3
VENDNO	Group1	Group2	Group3
VENDNO	abc1	abc2	abc3
VENDNO	Vanilla1	Vanilla2	Vanilla3
VENDNO	Apple1	Apple2	Apple3
VENDNO	Merge1	Merge2	Merge3
VENDNO	Art1	Art2	Art3
VENDNO	Promo1	Promo2	Promo3
VENDNO	Exit1	Exit2	Exit3
VENDNO	AUTOPress1	AUTOPress2	AUTOPress3
VENDNO	Index1	Index2	Index3
VENDNO	Community1	Community2	Community3
VENDNO	Fifa1	Fifa2	Fifa3
VENDNO	Text1	Text2	Text3
CUSTNO	Version1	Version2	Version3
CUSTNO	Excel1	Excel2	Excel3
CUSTNO	Profile1	Profile2	Profile3
CUSTNO	Finance1	Finance2	Finance3

Figure 51: Identifiers and scrambling function modules in Vendor/Customer address

HCM Rules

Similarly, in HCM rules, data is scrambled based on the country of the employee. The Scrambling Engine maintains the following identifiers and scrambling function modules in the HCM rules:

Identifiers: MOLGA

Scrambling function module: CNV_TDMS_SCR_PRESEL_HCM_DATA

Scrambling Rule: Rule_Personnel_First_Name

Rule called from Work Center
 Status Created By SAP Changed By KURMA
 Solution Category SAP_HCM Created On 12.07.2011 Changed On 07.01.2

Edit Close Switch Mode Change Category Validate

Scrambling Types Table-Fields Includes Condition Sets

Scrambling data

Scrambling Type: * Mapping_Personnel_General_Data

Scrambling Column Number: 03

Scrambling Function Module: CNV_TDMS_SCR_PRESEL_HCM_DATA

Identifier: MOLGA

Maintain Scrambling Values

Append Row Insert Row Delete Row Edit Column Text

	Country	Gender	First Name	Last Name	Full Name
	01	2	Katja	Lehmann	Katja Lehmann
	01	2	Anna	Schmidt	Anna Schmidt
	01	2	Isabel	Kunze	Isabel Kunze
	01	2	Regina	Hinz	Regina Hinz
	01	2	Simone	Schulze	Simone Schulze
	01	2	Carmen	MÃf#ÃÄ¼ller	Carmen MÃf#ÃÄ¼ller
	01	2	Christine	Bach	Christine Bach
	01	2	Beate	Schiller	Beate Schiller
	01	1	Bernd	Kant	Bernd Kant
	01	1	Paul	Heine	Paul Heine
	01	1	Peter	Lehmann	Peter Lehmann
	01	1	Andreas	Schmidt	Andreas Schmidt
	01	2	Elisabeth	Breitner	Elisabeth Breitner
	01	2	Gerda	Schulze	Gerda Schulze

Figure 52: Identifiers and scrambling function modules in Personnel name

Technical Background Information

The scrambling activity *Preparation of Data for Mapping* identifies the rules where the identifiers are maintained and prepares the table CNVTDMSSCR_KEYS with the information that is needed for scrambling in the special scenarios.

If VENDNO is maintained in a rule, this activity identifies the vendor addresses from the LFA1 table and stores the data in table CNVTDMSSCR_KEYS. Similarly the activity stores employee number and country information in the case of HCM rules where MOLGA is maintained as the identifier.

Data Browser: Table CNVTDMSSCR_KEYS Select Entries 6						
Table: CNVTDMSSCR_KEYS						
Displayed Fields: 7 of 8 Fixed Columns: 5 List Width 0250						
MANDT	PROJECT	PACKID	IDENTIFIER	VARKEY	SUBKEY1	
<input type="checkbox"/>	810	ZNAV_HCM	9EOHY	MOLGA	00000010	08
<input type="checkbox"/>	810	ZNAV_HCM	9EOHY	MOLGA	00001000	01
<input type="checkbox"/>	810	ZSP4_REP	9FOCL	VENDNO	0000006944	
<input type="checkbox"/>	810	ZSP4_REP	9FOCL	VENDNO	0000006945	
<input type="checkbox"/>	810	ZTSTERP	9FO2B	CUSTNO	0000005362	
<input type="checkbox"/>	810	ZTSTERP	9FO2F	CUSTNO	0000005362	

Figure 53: Table CNVTDMSSCR_KEYS after executing Preparation of Data for Mapping activity

IDENTIFIER: Stores the identifier maintained in the rule.

VARKEY: Stores the object keys such as vendor address, customer address number and personnel number.

SUBKEY1: Stores the country information (MOLGA) of an employee.

During the execution of the activity *Mapping of Data for Scrambling*, the scrambling function modules maintained for the rules (activated for the package) use the table CNVTDMSSCR_KEYS. The function modules map the data appropriately to handle the special case scenarios of address data scrambling and HCM data scrambling.

Special scenario: HCM Data Scrambling



The rule RULE_PERSONNEL_FIRST_NAME does not provide mapping data for all countries. When you activate the scrambling rule, personnel name data is scrambled according to the mapping value for countries in the rule. For those countries that don't have a mapping value, the default mapping information ('99') is set.

9.2.4 Step Nine: Using Condition-Sets

When to Use

Do you want to filter the records for scrambling based on a field in the primary table?

You use condition sets to filter the table records for scrambling.

A condition set is always applicable to just the primary table maintained in the rule. You can use the result set to scramble the other tables in the rule.

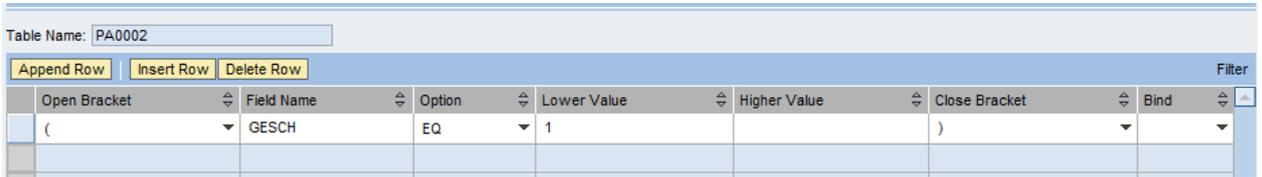
You can set simple or complex conditions on single or multiple fields of the primary table. Be sure to design your condition-set carefully by using the brackets and bind (AND, OR) options provided on the user interface.

Note

Condition sets do not work in rules that use the Custom Mapping scrambling type.

Example

You want to scramble the first name of the male employees in your company.



The screenshot shows a configuration window for a scrambling rule. At the top, the 'Table Name' is set to 'PA0002'. Below this are three buttons: 'Append Row', 'Insert Row', and 'Delete Row', along with a 'Filter' button. The main area is a table with the following columns: 'Open Bracket', 'Field Name', 'Option', 'Lower Value', 'Higher Value', 'Close Bracket', and 'Bind'. The first row is highlighted and contains the following values: '(', 'GESCH', 'EQ', '1', an empty field, and ')'. The 'Bind' column has a dropdown menu with 'AND' selected.

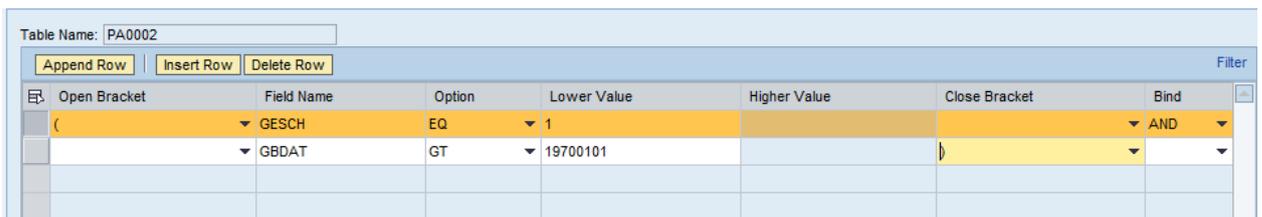
Open Bracket	Field Name	Option	Lower Value	Higher Value	Close Bracket	Bind
(GESCH	EQ	1)	AND

Figure 54: Assigning Condition Sets to scramble first name

In this example, table PA0002 contains the employee first names for both male and female employees. According to our scrambling condition in the scrambling rule, we want to scramble only those records in table PA0002 where the value of the field GESCH is '1'. (GESCH is a gender field in PA0002 table. The field stores '1' for Male and '2' for Female.)

Example

You want to scramble the first name of male employees born after January 1, 1970.



The screenshot shows a configuration window for a scrambling rule. At the top, the 'Table Name' is set to 'PA0002'. Below this are three buttons: 'Append Row', 'Insert Row', and 'Delete Row', along with a 'Filter' button. The main area is a table with the following columns: 'Open Bracket', 'Field Name', 'Option', 'Lower Value', 'Higher Value', 'Close Bracket', and 'Bind'. The first row is highlighted and contains the following values: '(', 'GESCH', 'EQ', '1', an empty field, and ')'. The 'Bind' column has a dropdown menu with 'AND' selected. The second row is also highlighted and contains the following values: 'GBDAT', 'GT', '19700101', an empty field, and ')'. The 'Bind' column has a dropdown menu with 'AND' selected.

Open Bracket	Field Name	Option	Lower Value	Higher Value	Close Bracket	Bind
(GESCH	EQ	1)	AND
	GBDAT	GT	19700101)	AND

Figure 55: Assigning Condition Sets to scramble first name along with a date

In the example above, the Scrambling Engine checks for the Birth Date fields GBDAT having a value greater than '19700101' and the Gender field GESCH having a value equal to '1'.

The fields GBDAT and GESCH belong to the PA0002 table, which is marked as a primary table in this rule.

Note

When you define a condition set, ensure that the conditions are on the fields of the primary table.

10 Consistent Scrambling of Data

When to Use

Do you want to ensure cultural and geographical consistency of names and addresses during scrambling?

You need the data to be scrambled consistently across the different related tables and fields.

Example

In address scrambling, I want to scramble the NAME1, NAME2, NAME3, STREET and CITY fields of table ADRC consistently by assigning a scrambling type Random Table with Multiple Columns as shown below.

Name1	Name2	Name3	Name4	Name text	City1
Group1	Group2	Group3	Group4	Grp1	Dundee
abc1	abc2	abc3	abc4	Grp2	Worcester
Vanilla1	Vanilla2	Vanilla3	Vanilla4	Van1	Edinburgh
Applep	Appleq	Applr	Apples	Appl	Norwich
Merge1	Merge2	Merge3	Merge4	Merg	Chichester

Figure 56: Assigning the Random table with multiple columns Scrambling Type to a Scrambling Rule

To scramble data consistently, the Scrambling platform requires key field information by which the NAME1 and NAME2 fields can be uniquely identified in a table. Hence, assign ADRC-ADDRNUMBER as a key set, so that the scrambling programs can assign a new value consistently based on the address number field.

Example mapping for ADRC-NAME1

Key (address number)	Old value (Non-scrambled NAME1 value)	New Value (scrambled NAME1 value)
1000	SAP	abc1
1001	Microsoft	Group1
1002	Google	Vanilla1

Example mapping for ADRC-NAME2

Key (address number)	Old value (Non-scrambled NAME2 value)	New Value (scrambled NAME2 value)
1000	AG	Abc2
1001	Limited	Group2
1002	Incorporations	Vanilla2

The Scrambling platform assigns the new value from the same row of the look up values maintained in the random table with multi columns (as shown in the figure above) based on a key.

10.1 Global Mapping

You can use Global Mapping to scramble data from a random selection table with multiple columns where you want multiple rules to use the same mapping to scramble data consistently. You can assign a global mapping to several scrambling rules.

You can use this scrambling type to assign random, meaningful values to more than one related field.

You could have a random table with 2, 3, 4, 5, 10, 15, 20 or 30 columns. You can choose any of them based on the number of columns you want to maintain in the lookup table.

Points to Remember

- Create a separate rule for every column field involved in scrambling and maintain the scrambling column number in the rule.
- Key sets are required attributes for every table field defined in a rule when the scrambling type Random table with more than 1 column is used.
- Always create the Random table with more than 1 column scrambling type globally on the Global Mapping tab page, as the scrambling type would be used in multiple rules.



Example

You want to scramble address data randomly looking into the values of a lookup table and you also want this new address to be consistent with other related attributes. When the country is changed from United States to India, the state should be changed from New York to Delhi.

Values in the table before scrambling:

Address Number	Street	City	State	Country
1000	101st Avenue	San Diego	CA	US
1001	Lonsdale Street	Melbourne	Victoria	Australia
1002	3010, Wilson	Chicago	Illinois	US
1003	Vasant Kunj	Firozabad	New Delhi	India

Values in the table after scrambling:

Address Number	Street	City	State	Country
1000	Whitefield	Bangalore	Karnataka	India
1001	Kent Street	Sydney	NSW	Australia
1002	St Peter Road	Chicago	Illinois	US
1003	Whitefield	Bangalore	Karnataka	India

In the example above, all the fields of the table or tables are scrambled using the lookup values provided in the Global scrambling type. You need a single rule for a particular field.

In this example, we require a rule each for the street, city, state, and country fields. The fields can belong to a single table or to many different tables.

i Note

A few names could repeat several times since the lookup set is smaller than the actual number of records scrambled.

4. Refer to Chapter 13 [Example Rule Creation](#) to follow the steps for rule creation.

11 Additional Features in the Scrambling Workbench

You can work with several easy-to-use features to streamline your experience in the workbench.

11.1 Validation Feature for Scrambling Rules

When to Use
Do you want to validate the technical details of a rule before using the rule in a package?

We recommend checking the consistency of a rule before using the rule in a package. You can carry out the consistency check by choosing the Validate pushbutton on the Scrambling Rule Maintenance screen.

Prerequisite

You have assigned a landscape to the rule.

Validation Checks

The validate feature carries out the following checks:

- The table-fields are maintained for the rule
- The table-fields and/or domains entered on the Table-Fields tab page are valid
- The primary table-field is maintained for the rule
- The specified Routine name is valid and exists in the Include maintained for the rule.
- The Include information is maintained correctly for the rule
- The entered Include exists in the sender system or the execution system
- The scrambling values are assigned to the rule
- The key scrambling field and scrambling program are maintained for the rule on the Scrambling Type tab page if the scrambling type used is *Global Mapping with Random table with multiple columns*

11.2 Simulation for Scrambling Rules

On the *Scrambling Rule Maintenance* screen, you can simulate the data scrambling for the primary table-field combination by choosing the *Simulate* pushbutton. After you select *Simulate*, choose one of the following options:

- Simulate from Sender or Execution System
- Simulate Using User-Defined Data

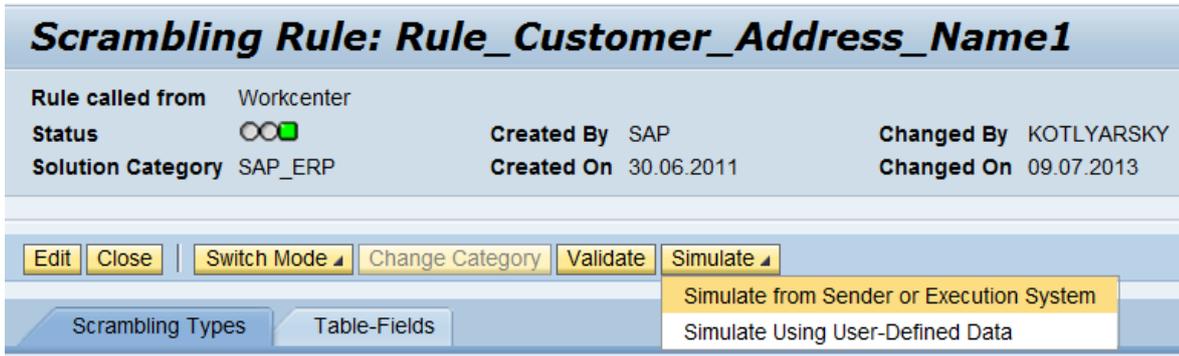


Figure 57: Selecting the Simulation Option

Simulation from the Sender System or the Execution System

When you choose the *Simulate from Sender or Execution System* menu option, the SAPGUI logon screen for the sender system or the execution system appears.

- If you create the landscape based on the TDMS Landscape Template and assign the landscape to a scrambling rule, the SAPGUI logon screen for the sender system appears.
 - If you create the landscape based on the TDMS Landscape Template for Standalone Scrambling and assign the landscape to a scrambling rule, the SAPGUI logon screen for the execution system appears.
1. After you log in, the *Selection Criteria for Data* screen appears. This screen displays the data relevant to the primary table-field combination for your scrambling rule.
 2. Enter the selection criteria and execute.

The *Simulation* screen appears where you can view the data before scrambling and the data after scrambling for the selection criteria.

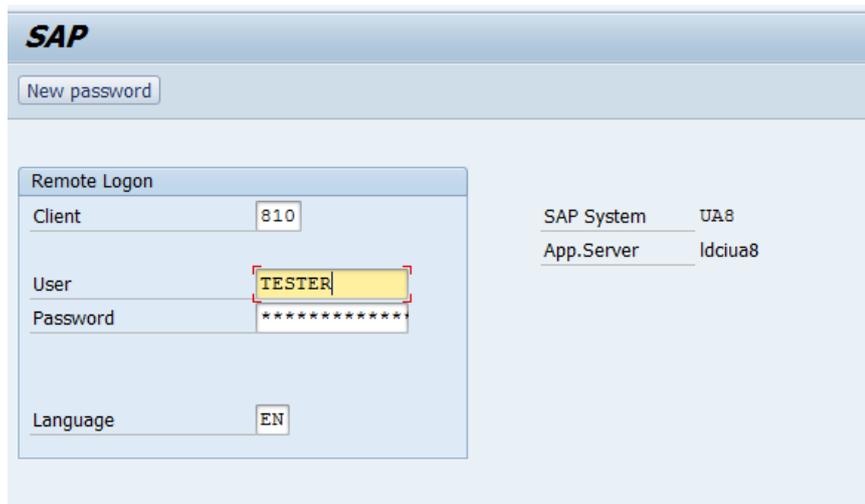


Figure 58: The SAPGUI logon screen

Selection Criteria for Data: Table ADRC

Field for selection

Address number: 121 to

From: to

Address version: to

Maximum No. of Hits: 99

Figure 59: Entering your selection criteria for simulation

Simulation:Rule_Customer_Address_Name1

Choose Fields

Data Before Scrambling

Addr. no.	From	V	Name
121	01.01.0001		Ides AG

Data After Scrambling

Addr. no.	From	V	Name
121	01.01.0001		Search1

Figure 60: The *Simulation* Screen

Simulation Using User-Defined Data:

After you select the *Simulate Using User-Defined Data* menu option, the *Simulation of Scrambling Rule* screen appears.

- To fill the content in the *Data Before Scrambling* table, you can carry out the following actions:
 - Manually append rows
 - Autofill data
 - Import data from a .CSV file

2. After carrying out the desired action, select the *Simulate* pushbutton to view the simulation of the data scrambling for the entered data.
3. You can view the scrambled content in the *Data After Scrambling* table.

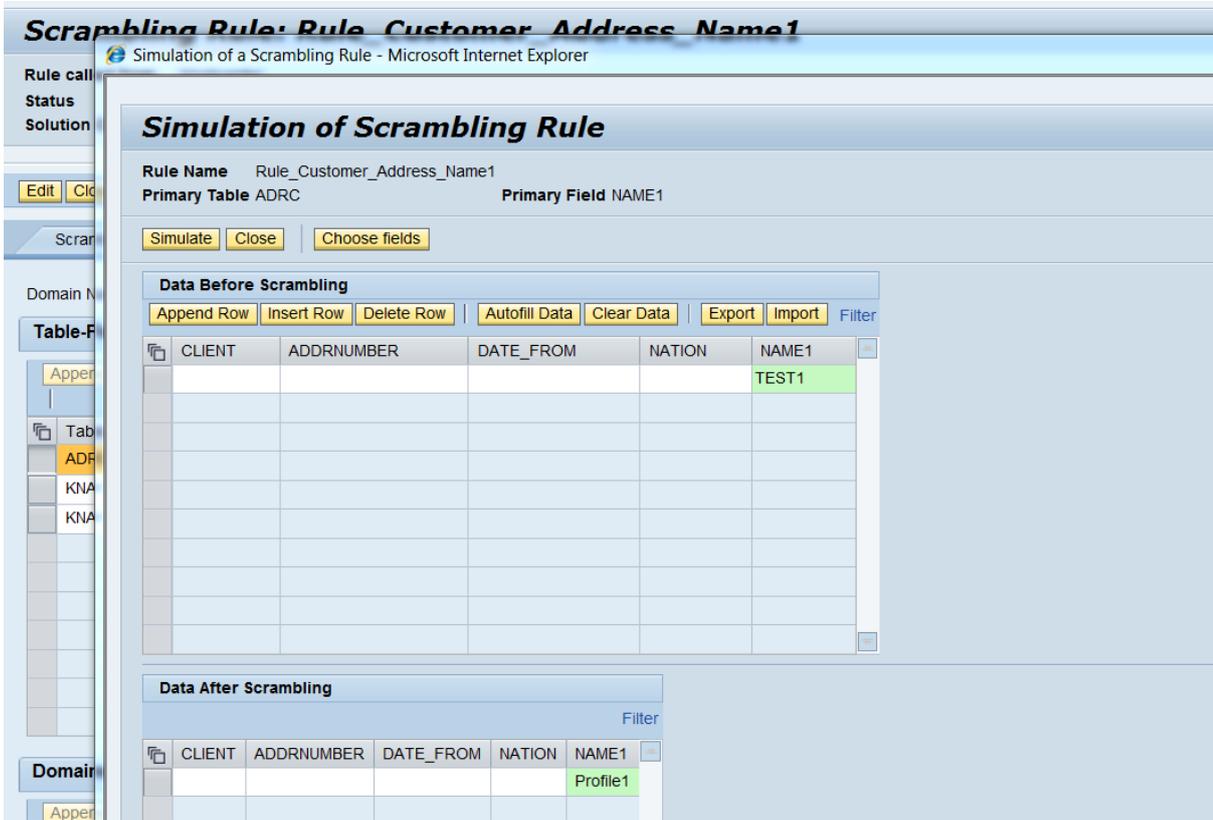


Figure 61: Viewing scrambled content in the *Data After Scrambling* table

11.3 Search Feature for Scrambling Rules

On the *Overview* screen of the Scrambling platform, you can search for a rule using the rule name or table or field name as search criteria. The search result is highlighted on the *Overview* tab page.

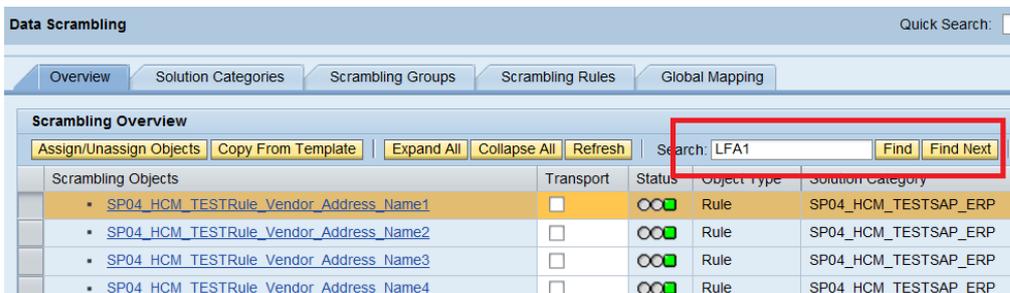


Figure 62: Searching for scrambling rules

11.4 Import-Export of Data from Files for Manual 1:1 Mapping

You can import mapping values and table-field combinations into a rule.

You can export mapping values and table-field combinations to a .CSV file.

11.4.1 Prerequisites

- You are familiar with:
 - The TDMS Work Center
 - The Data Scrambling workbench
 - Concepts for Manual 1:1 Mapping, mapping values and the scrambling rule
- To import mapping values, you have stored the values in a .CSV file.

11.4.2 Exporting Data to the .CSV File

1. Choose the *Scrambling Rules* tab page in the Data Scrambling workbench.
2. Click a rule that you want to use to export data, for example, ZTEST_MAPPING.



Figure 63: Selecting a Scrambling Rule

The *Scrambling Rule* screen opens.

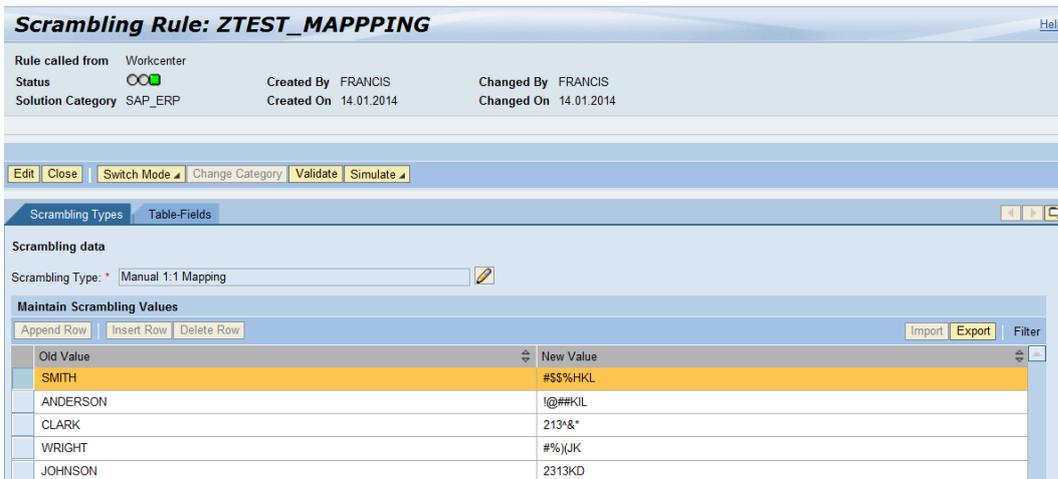


Figure 64: The Scrambling Rule Screen

3. To save the mapping values of the rule to a .CSV file, choose the *Export* pushbutton.

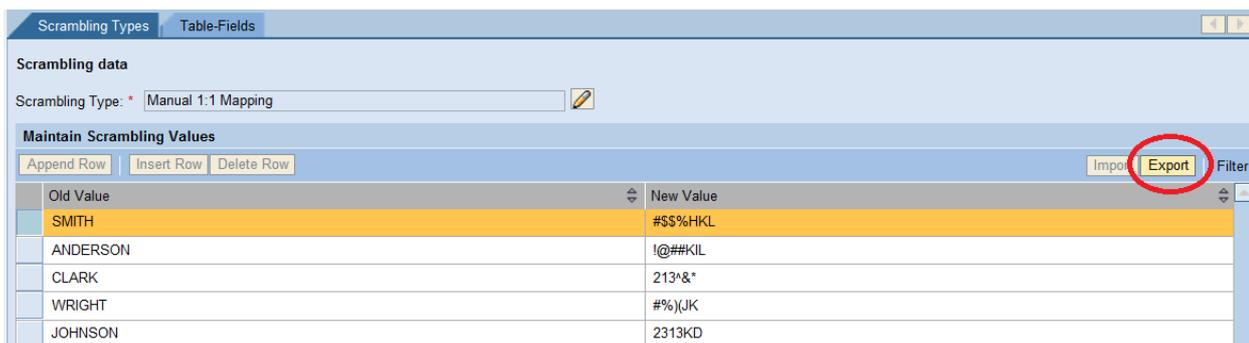


Figure 65: Choosing the Export pushbutton

The *File Download* dialog box opens.

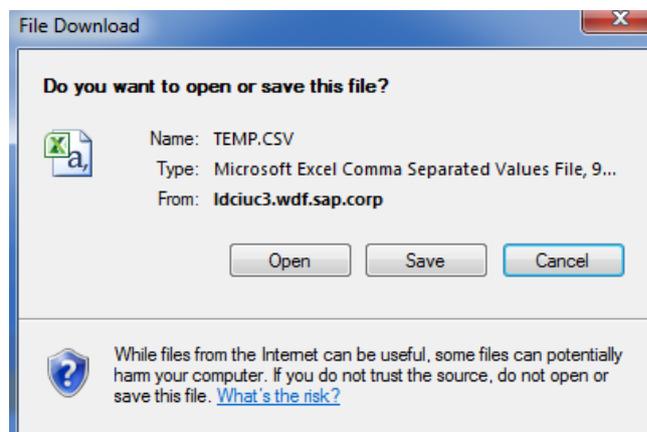


Figure 66: The File Download Dialog Box

4. Save your file to the desired location.

11.4.3 Importing the Mapping Values to a Scrambling Rule

i Note

You can import mapping values to an existing scrambling rule or when creating a scrambling rule.

1. Choose the *Import* pushbutton to import mapping values from a .CSV file during the Assign Scrambling Type step.

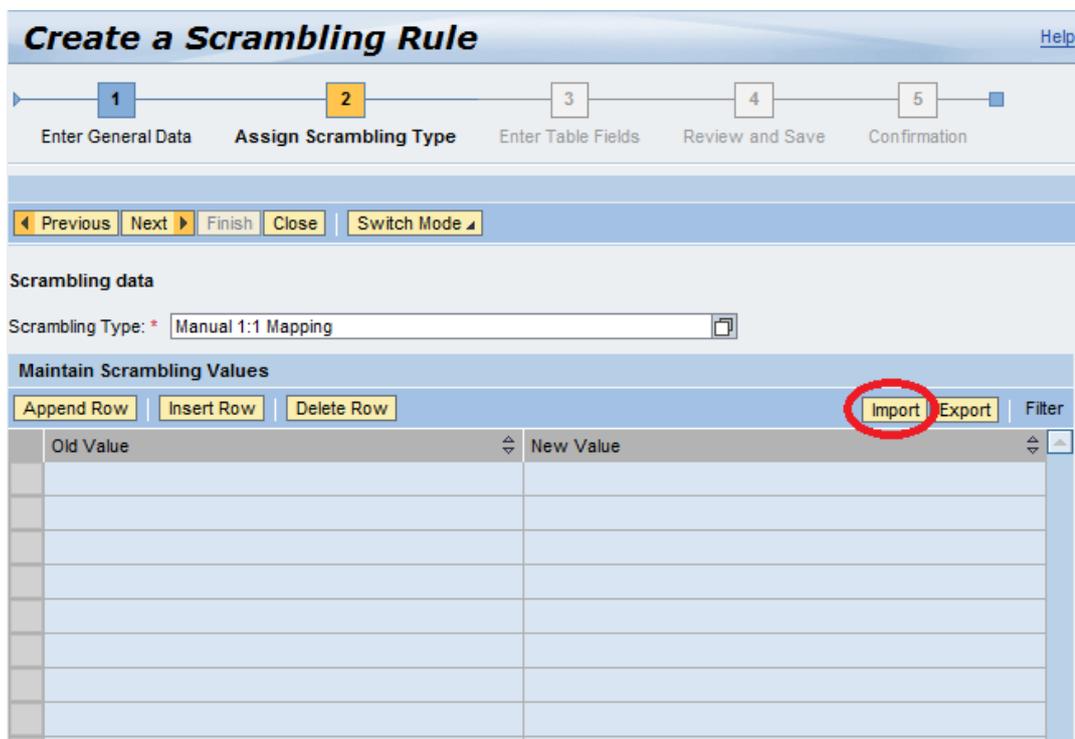


Figure 67: Choosing the Import Pushbutton

The Import Mapping Values dialog box opens.

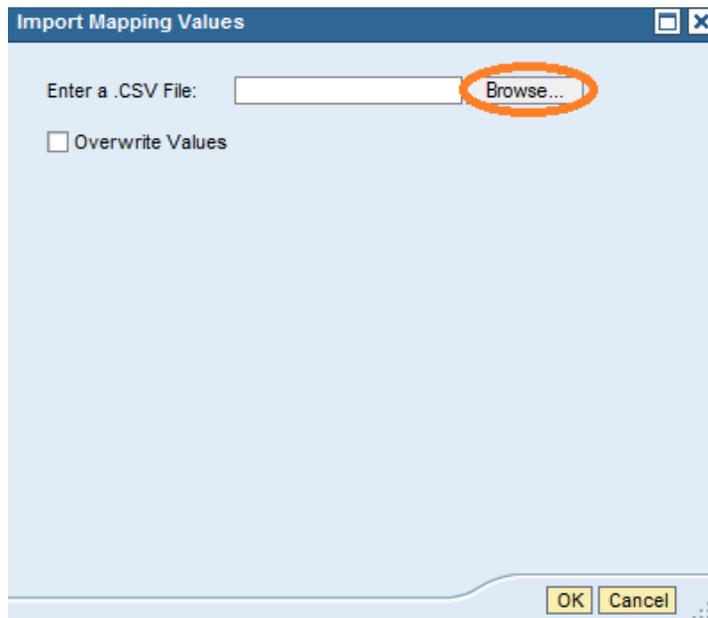


Figure 68: The Import Mapping Values Dialog Box

2. Browse for and select the appropriate .CSV file.
3. If you want to overwrite the values in a scrambling rule with the values in your file, select the *Overwrite Values* checkbox.

12 Using Scrambling Objects at the Project and Package Levels

To use the scrambling objects we deliver or the custom content you created in the Scrambling work center at the project level, you need to first copy the objects from the Scrambling work center to the Project work center.

When you use the scrambling objects available at the workbench level or the project level in your package, you need to first copy the objects from the Scrambling work center or from the project work center to the Package work center.

Unless you perform Copy from Work center or Copy from Project, you cannot view and use the scrambling objects created at the Scrambling workbench level or at the project level.

12.1 Copy Option at Project Level

You can copy the scrambling objects that are available in the work center to your project.

Use Case: You can copy the objects to the project and create or modify the objects for use ONLY in the packages under the project.

Navigate to the *Scrambling* tab page and choose the *Edit* button. This enables the Copy pushbutton in the work center

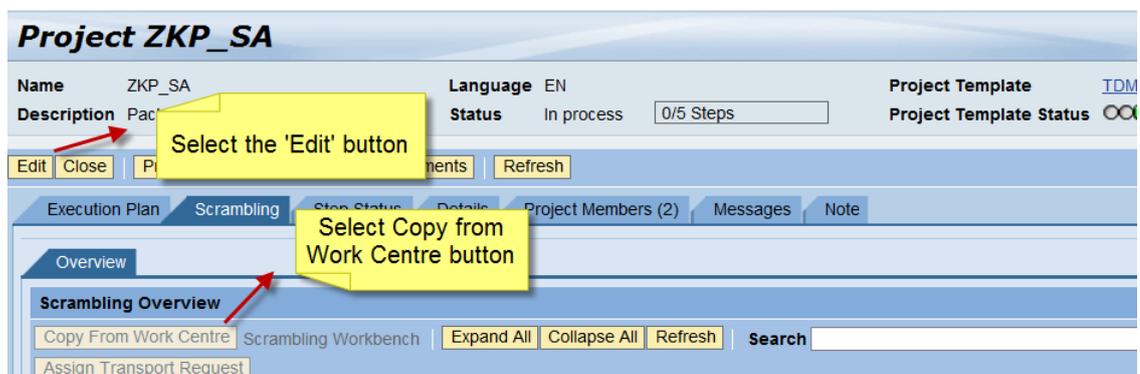


Figure 69: Copying Scrambling customizing at the project level

After you select the *Copy from Work Center* pushbutton for the first time in the project, all available scrambling objects present in the work_center are copied to the project.

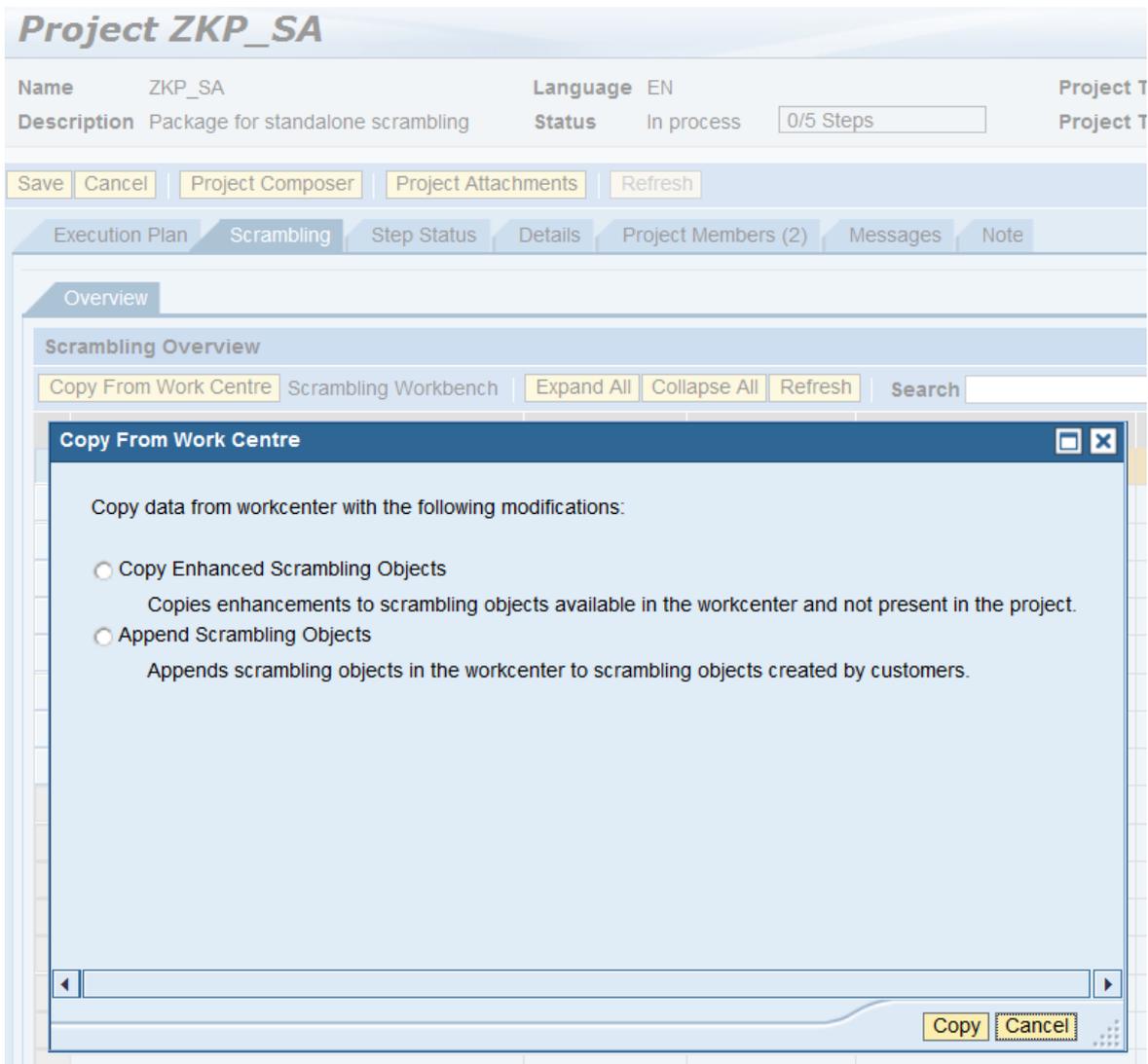


Figure 70: Options available to copy scrambling customizing

From the second time on, when you select the *Copy from Work Center* pushbutton, you have the following options:

- **Copy Enhanced Scrambling Objects:** This option copies enhancements available in the work center but not present in the project.
- **Append Scrambling Objects:** This option appends the scrambling objects available in the work center to the project.

12.2 Activating Scrambling Customizing at Project Level

If you want to scramble some table fields as part of a package, you need to first activate those rules that contain the conversion logic to scramble these table fields at the package level. Unless you activate the scrambling objects, you cannot use them in the package.

1. On the Scrambling tab page, select the scrambling object you want to activate: supergroup, scrambling group, or scrambling rule.
2. Select the *Active* checkbox to activate the scrambling rules that enable scrambling.
3. Save your entries.



Caution

If two scrambling rules contain the same table-field combination, you can activate the rules only if both the rules contain condition sets.

12.3 Copy Option at the Migration Package Level

You can copy the scrambling objects that are available in the work center or project.

Execute the activity *Define Scrambling Rules* to define or activate the scrambling rules for your package.

After you choose the *Copy* pushbutton, you have the following options:

- Copy From Project
- Copy From Work Center

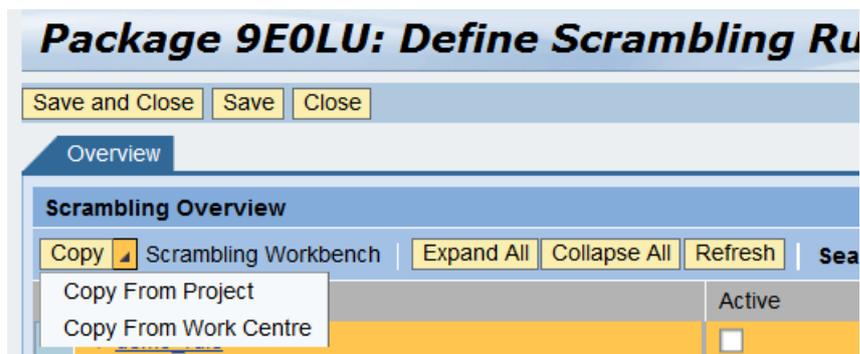


Figure 71: Copying Scrambling customizing at the package level

- During the first run, selection of either option enables the copy of all objects in the work center or project to the migration package.
- From the second run on, after you select an option, the following dialog box appears:

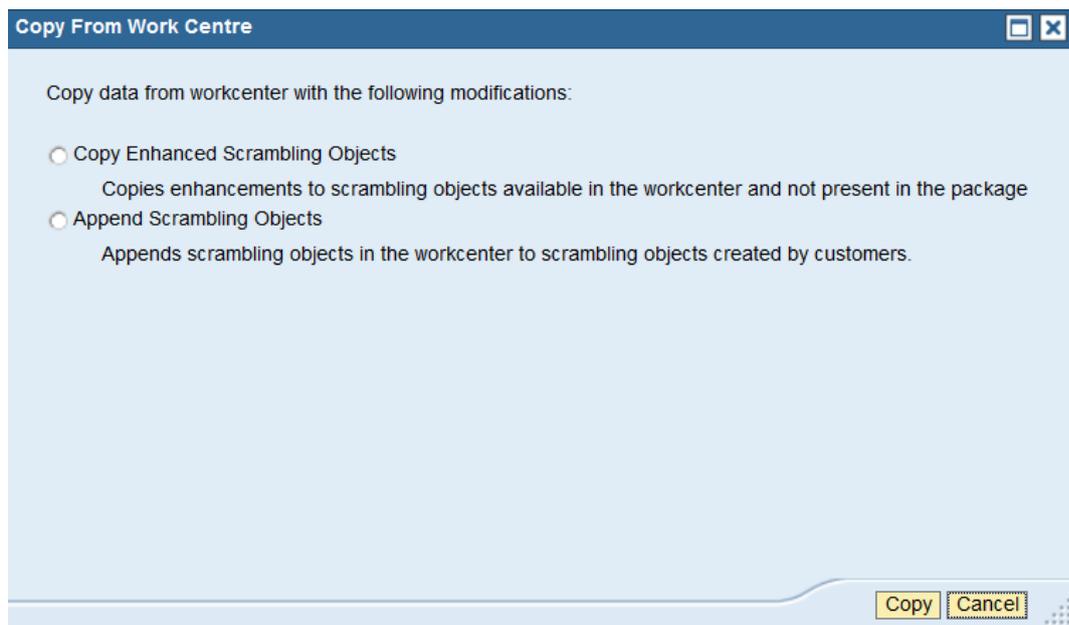


Figure 72: Options available to copy scrambling customizing

- **Copy Enhanced Scrambling Objects:** This option copies enhancements available in the work center or project but not present in the package. Hence, during the first run, this option enables the copy of all objects in the work center or project to the migration package.
- **Append Scrambling Objects:** This option appends the scrambling objects available in the work center or project to the package.

12.4 Activating Scrambling Customizing at the Migration Package Level

If you want to scramble some table fields as part of a package, you need to first activate those rules that contain the conversion logic to scramble these table fields at the package level. Unless you activate the scrambling objects, you cannot use them in the package.

1. On the *Scrambling* tab page, select the scrambling object you want to activate: supergroup, scrambling group, or scrambling rule.
2. Select the *Active* checkbox to activate the scrambling rules that enable scrambling.
3. Save your entries.

13 Example Rule Creation

The example below shows how to create a custom rule for credit card scrambling based on the card type.

1. Select *Data Scrambling* from the TDMS work center.

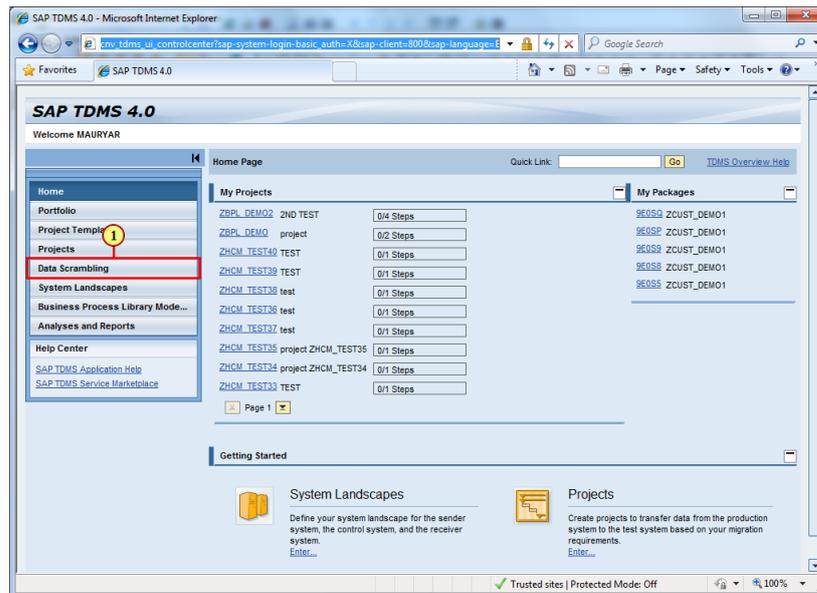


Figure 73: Selecting Data Scrambling tab in the work center

2. Choose the *Scrambling Rules* tab page in the *Data Scrambling* work center.

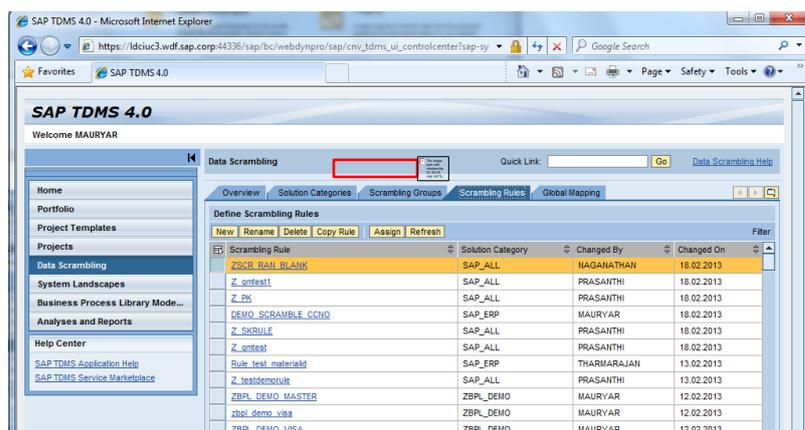


Figure 74: Selecting Scrambling Rules tab page in the work center

3. Choose the *New* pushbutton.

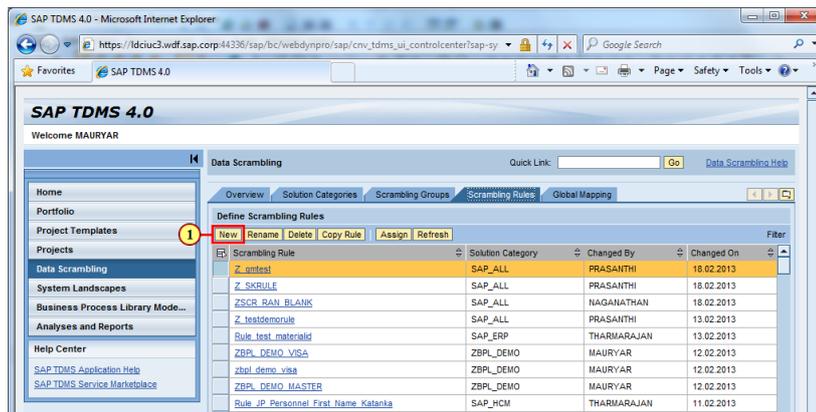


Figure 75: Selecting the *New* pushbutton

- On the *Create a Scrambling Rule* window, enter the rule name, solution category, scrambling supergroup and scrambling group.

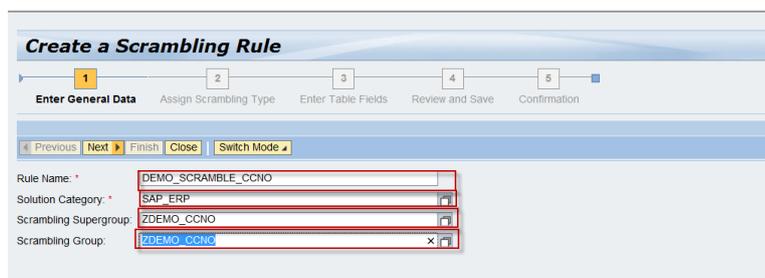


Figure 76: Entering information to create a scrambling rule

- Choose the *Next* pushbutton to navigate to the *Scrambling Type* screen.
- Use the input help and select *No Mapping* as the scrambling type.
- Choose the *Select* pushbutton.

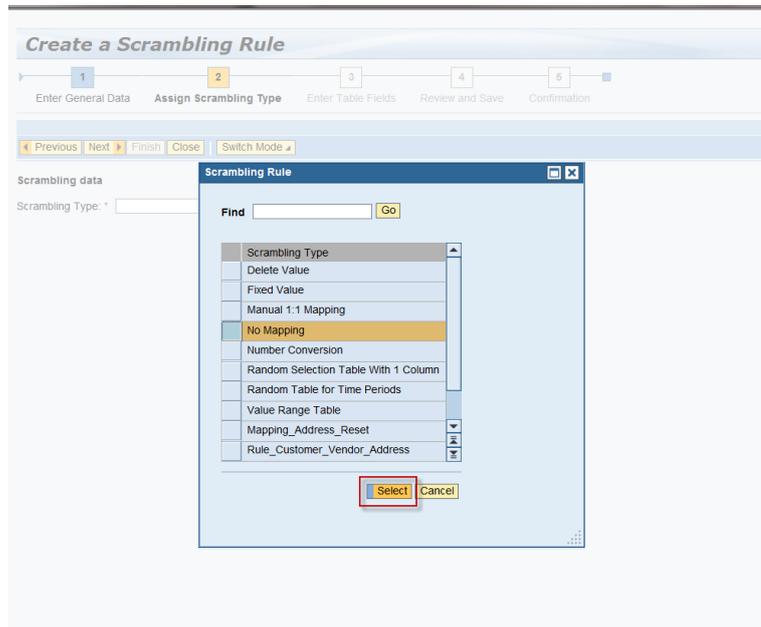


Figure 77: Selecting a mapping type for the scrambling rule

- Choose the *Next* pushbutton.



Figure 78: Selecting Next pushbutton

You are now in the step: *Enter Table Fields*.

- Change the mode to Expert mode.

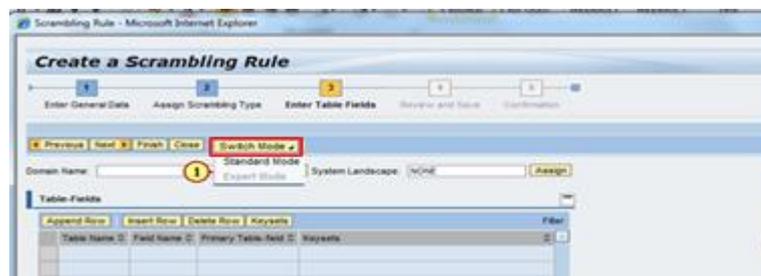


Figure 79: Changing the mode of scrambling rule

- Select the *Append Row* push button.

A new entry appears in the *Table-Fields* entries.

11. Enter the following data:

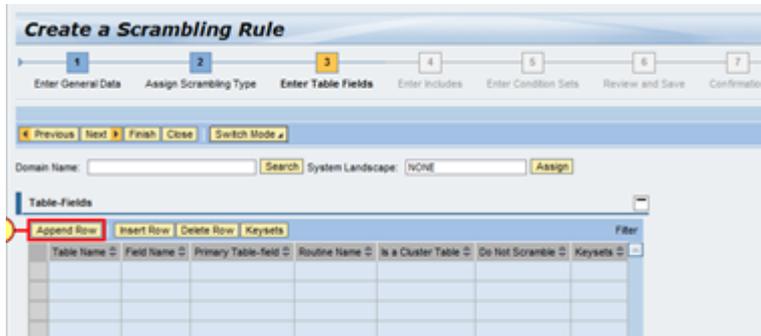


Figure 80: Creating a new scrambling rule

Table Name: ZCUSTOMER

Field Name: CCNO

Routine Name: ZSCRAMBLE_DATA

Primary Table-Field: Selection option should be on

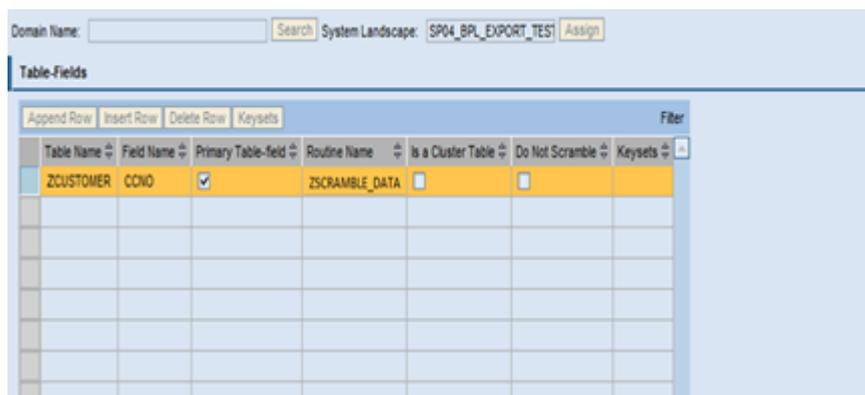


Figure 81: Entering data in the table-fields

12. After entering data, choose the *Next* pushbutton.

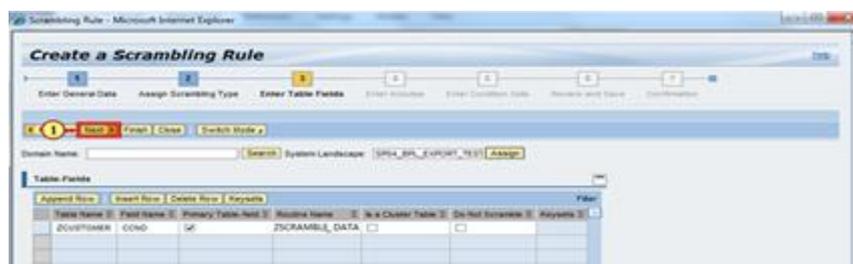


Figure 82: Selecting *Next* pushbutton

13. On the Includes tab page, choose the *Append Row* pushbutton to enter Include data.

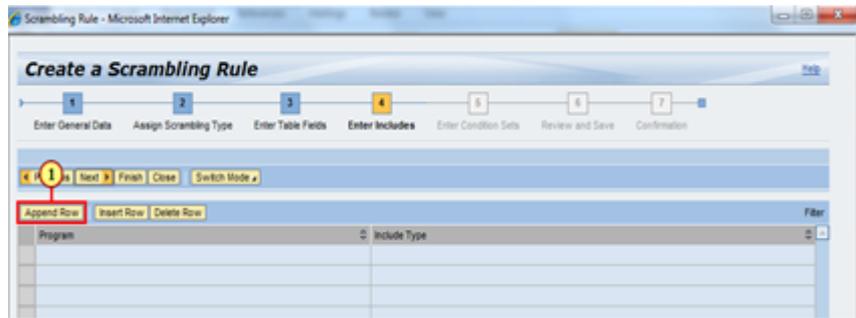


Figure 83: Selecting *Append Row* pushbutton

14. Enter the following include with the include types:

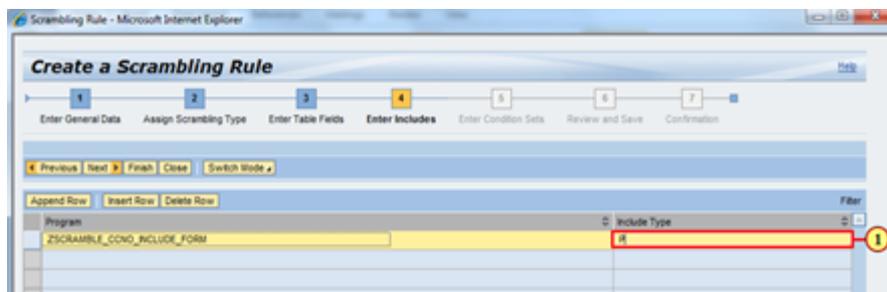


Figure 84: Entering the Include Type

Program name: ZSCRAMBLE_CCNO_INCLUDE_FORMS; Include Type 'P'

15. Choose the *Finish* pushbutton to go to the *Review and Save* step.



Figure 85: Finishing and Saving the Scrambling Rule

16. Create an Include by carrying out the following steps in your central/control system.

17. Create an Include as ZSCRAMBLE_CCNO_INCLUDE_FORM and write your code (example below). Save and activate the Include.

Syntax

```
FORM zscramble_data USING      p_rule_id
                                CHANGING p_value
                                p_changed.
```

```
FIELD-SYMBOLS : <lv_iss_type> TYPE ANY.
```

* Assigning the field

```
ASSIGN COMPONENT 'ISS_TYPE'
OF STRUCTURE gd_original
TO <lv_iss_type>.
```

* if its visa or master scramble the cardno.

```
IF <lv_iss_type> = 'VISA'.
    p_value = '1111111111111111'.
    p_changed = 'X'.
ELSEIF <lv_iss_type> = 'MASTER'.
    p_value = '2222222222222222'.
    p_changed = 'X'.
ENDIF.
```

```
ENDFORM.                    "scramble_data
```

18. Assign the rule to a supergroup. Choose the [Overview](#) tab page.

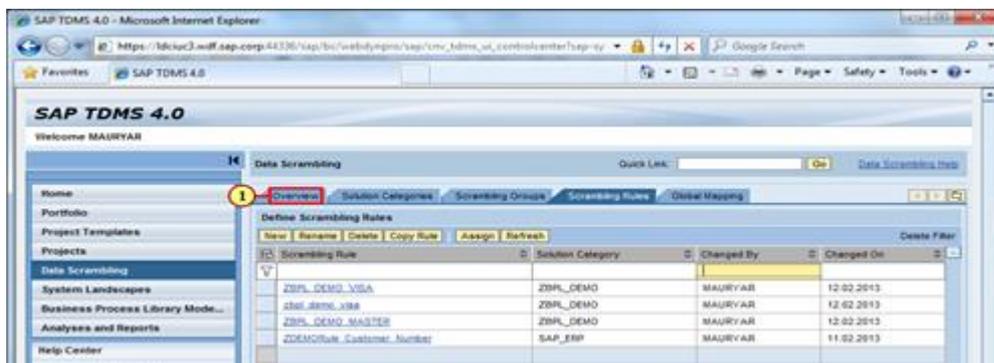


Figure 86: Assigning the rule to a group

19. Choose the [Assign/Unassign Objects](#) pushbutton.

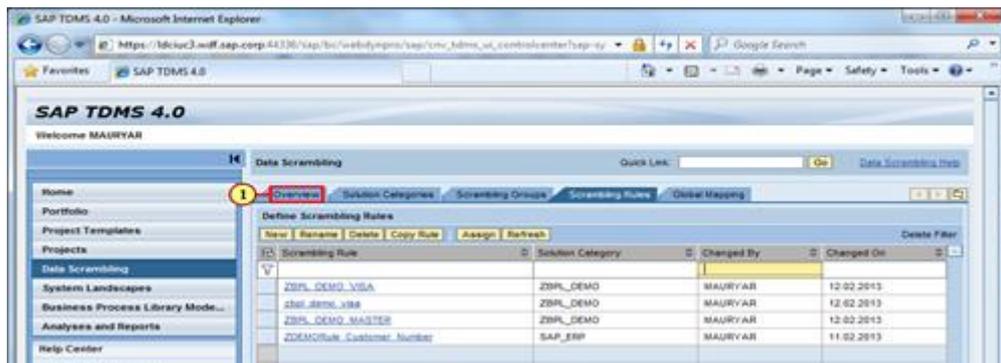


Figure 87: Choosing *Assign/Unassign Objects* pushbutton

20. Select the supergroup name.

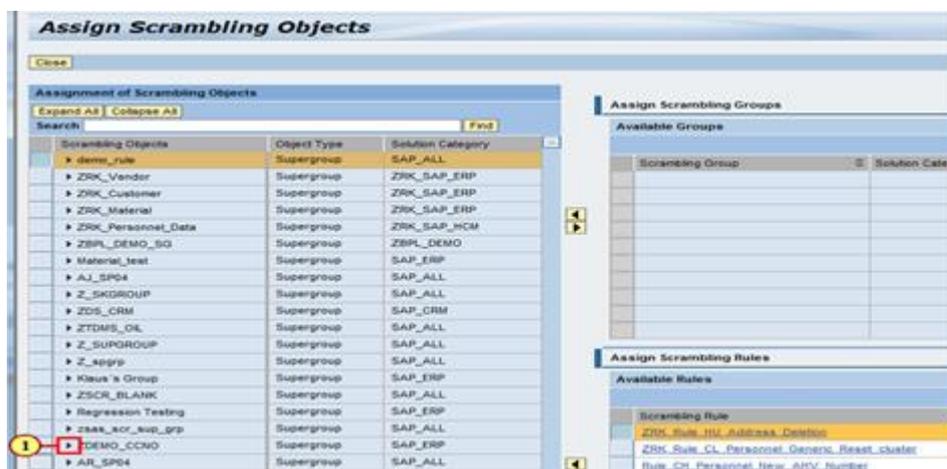


Figure 88: Selecting the supergroup to assign the rule

21. Select the group name under the selected supergroup name.

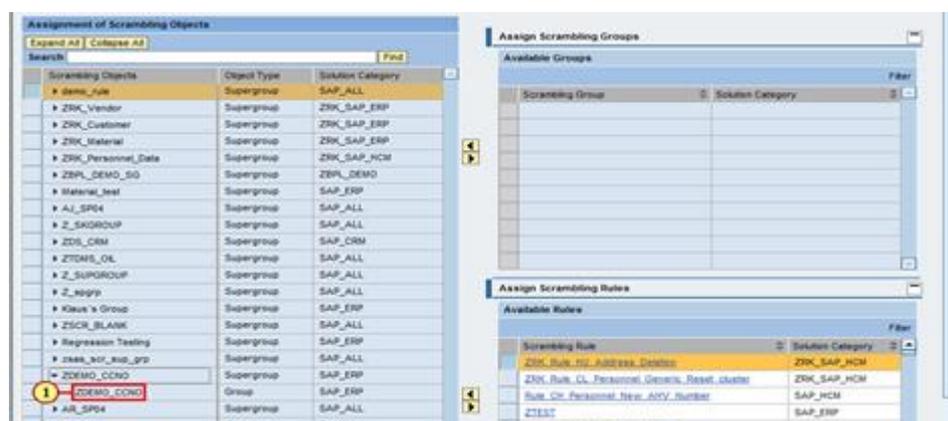


Figure 89: Selecting the group to assign the rule

22. In the *Assign Scrambling Groups* section on the top right, select the rule that you created.

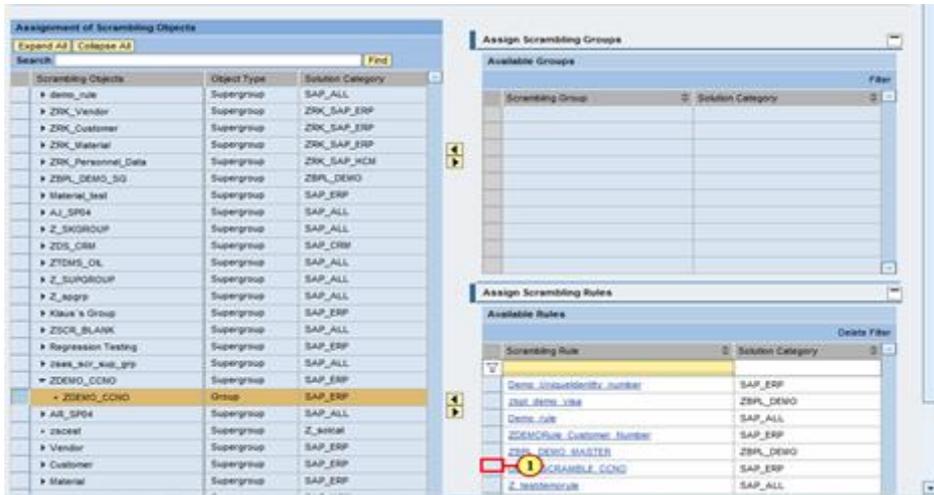


Figure 90: Assigning rules to a scrambling group

23. After you select the rule, choose the back arrow button to assign the scrambling rule to the group.

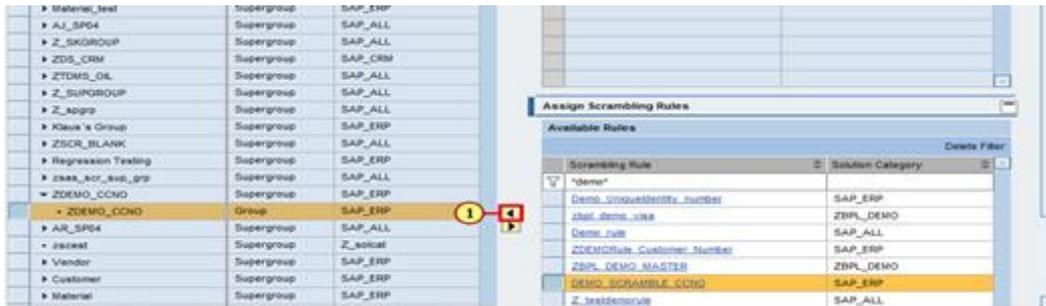


Figure 91: Selecting the back arrow button to assign rule to the group

24. Choose the *Close* pushbutton.

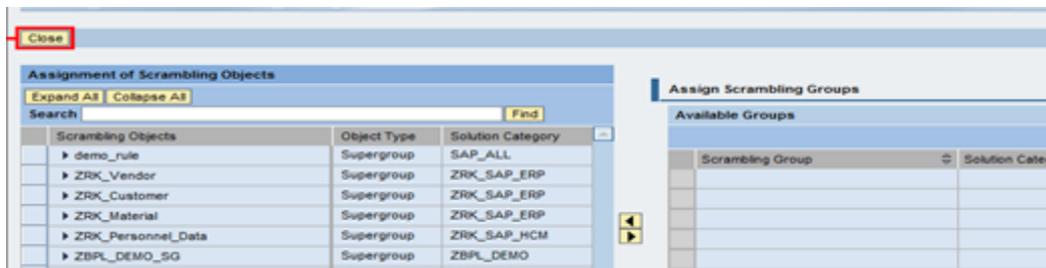


Figure 92: Selecting the *Close* pushbutton

25. Choose the *Refresh* pushbutton from the main window. The supergroup status changes to green.

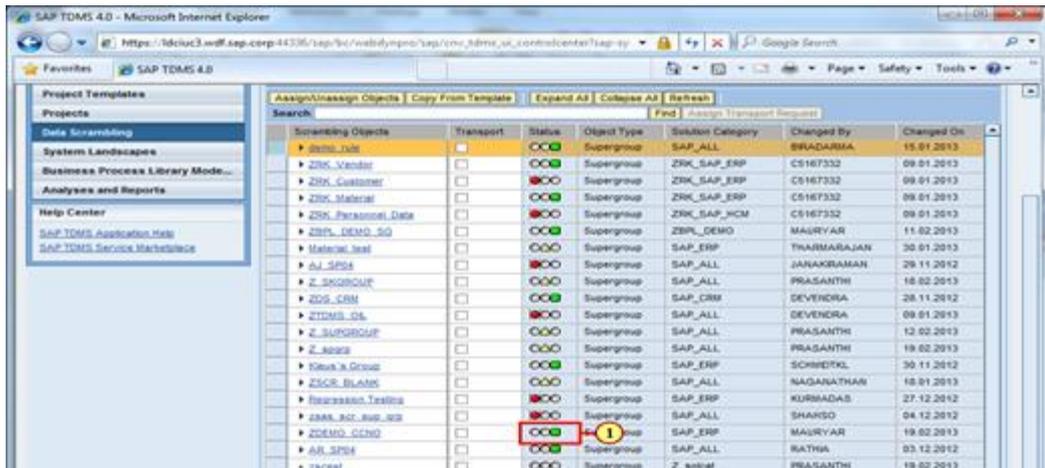


Figure 93: Selecting the Refresh pushbutton to see the changes

14 Troubleshooting

Issue

You have defined your own custom include and custom subroutine. When you execute the Convert Migration Objects activity, the following error appears: *Form: <Your_Form_Name> does not exist.*

Solution

Check whether any statement exceeds 72 characters in the Include in your central system. The ABAP technical limitation for dynamic program generation requires that all ABAP statements have a maximum length of 72 characters.

Carry out the check as follows:

11. Choose the Edit mode for the Include.
12. Select *Settings* under the *Utilities* menu.

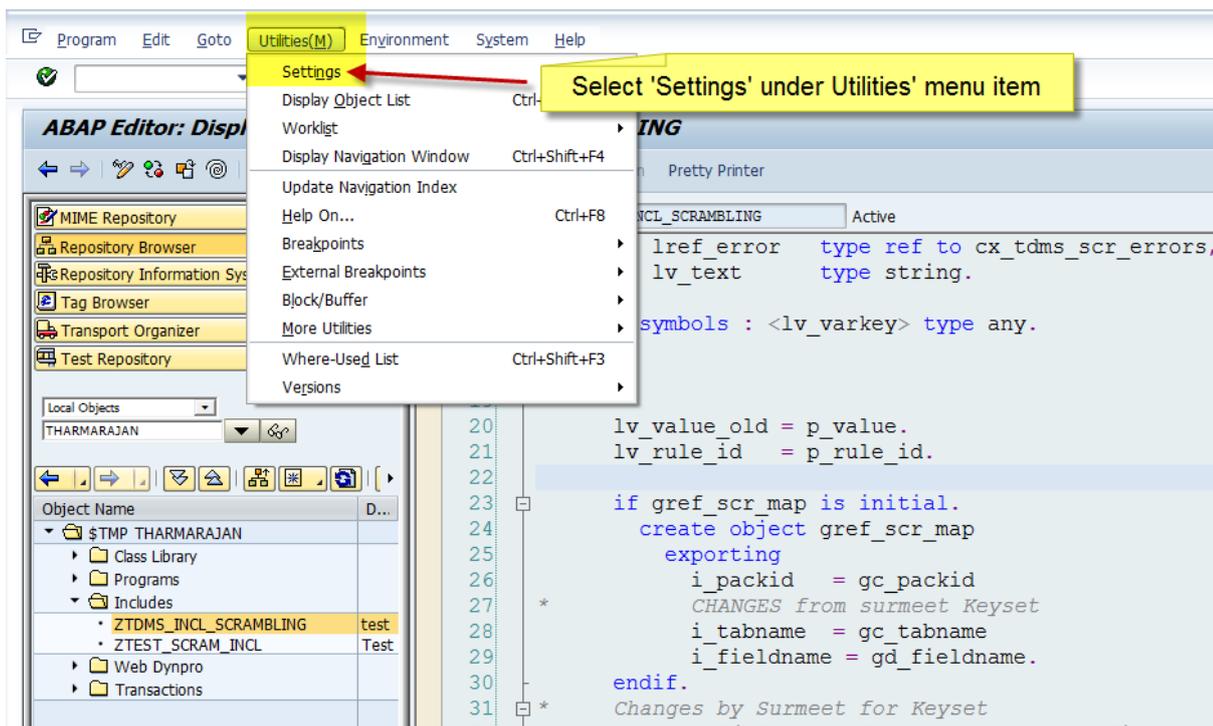


Figure 94: Selecting Settings under the *Utilities* menu

13. On the next screen, select *Downwards-Compatible Line Length* (72) as shown below:

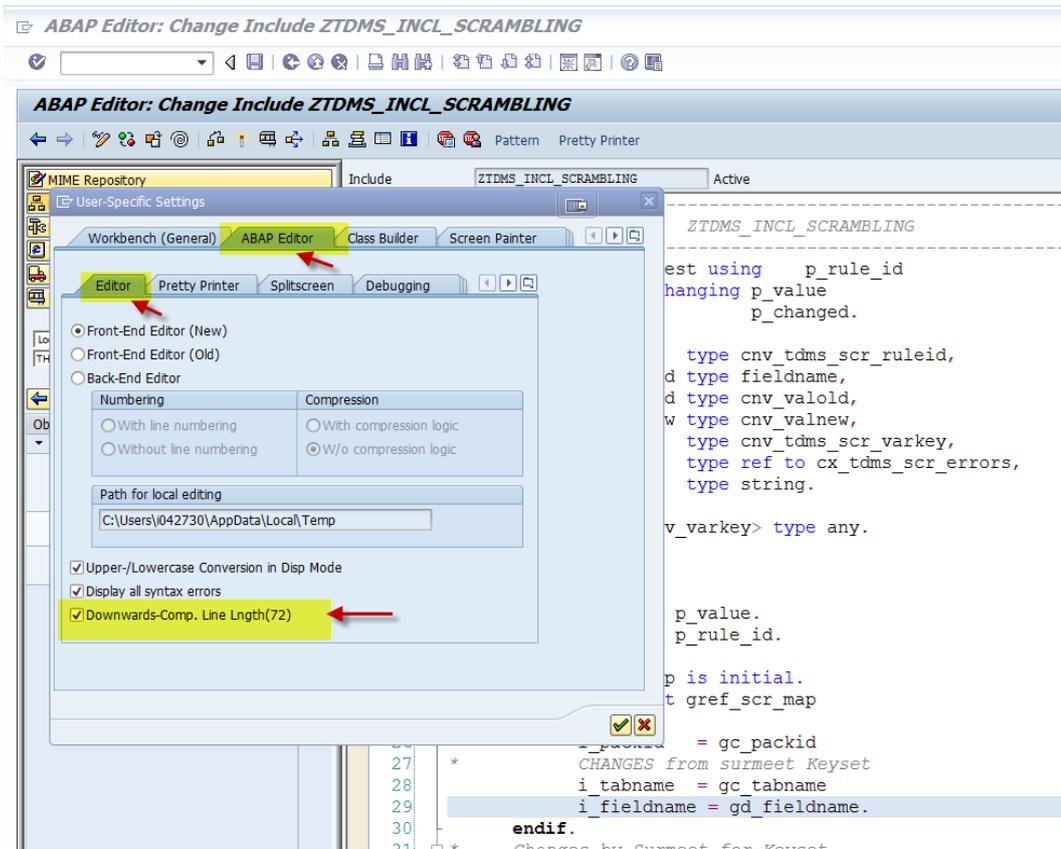


Figure 95: Selecting the Downwards-Compatible Line Length

After you make your selection, a red line appears to the right of the screen. Ensure that your statements are inside the red line.

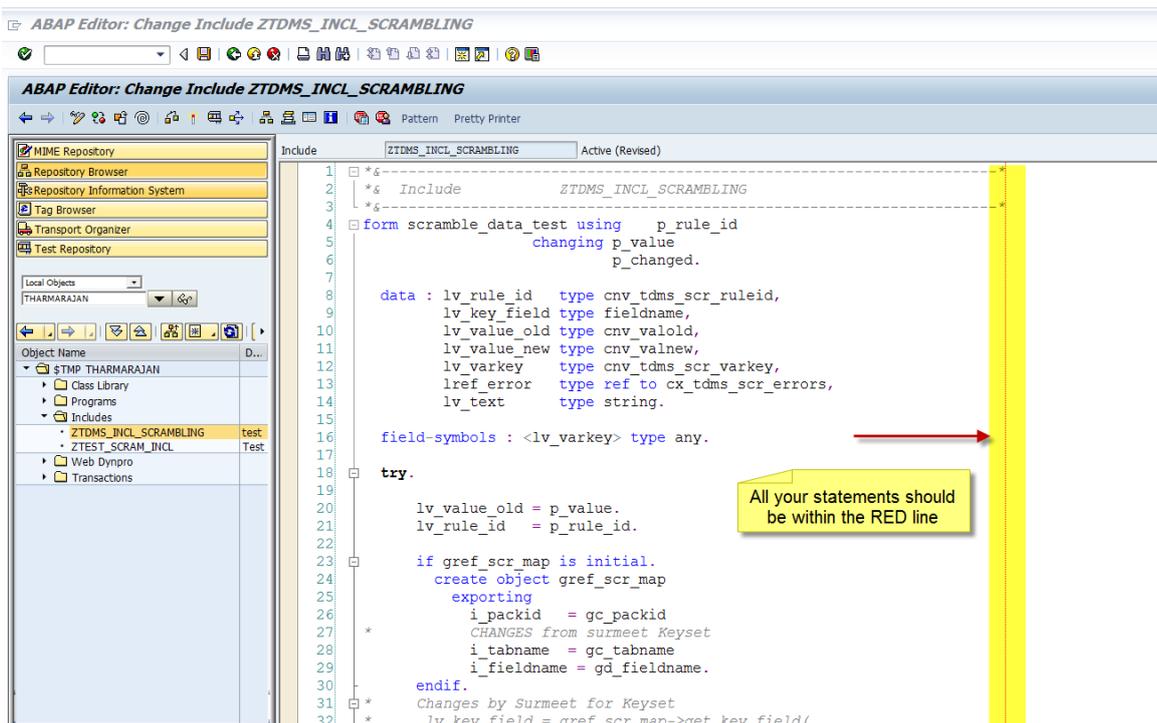


Figure 96: Checking that all statements are inside the red line

15 FAQs

15.1 Types of Data that Can Be Scrambled

Can I scramble cluster data in the Transfer scenario?

No. You can only scramble cluster data in the stand-alone scrambling scenario.

Example:

Cluster Table BSEG. BSEG is a part of the RFBLG table cluster. Using TDTIM, we can transfer RFBLG table cluster, but not the cluster table BSEG. Hence, if you define scrambling rules for BSEG and want to perform a transfer with scrambling, scrambling will not happen for BSEG.

15.2 Solution Category

What can I use the solution category SAP_ALL for?

This solution category can be used when you want to create or access scrambling objects common across several different migration packages. Currently, no standard content is provided by SAP under the solution category SAP_ALL. You can create your custom scrambling objects under this category.

Can I enhance or modify the solution categories SAP delivers?

Yes, you can modify and enhance these solution categories after copying the objects from the template to the TMDS work center.

Example: If you want to use the ERP rules in the TDMS Banking package, add the TDMS Banking migration solutions to the existing SAP_ERP solution category.

15.3 Scrambling Rules: Basic Technical Information

Can I have multiple primary table fields for a rule?

No, you can have only 1 primary table field for a rule.

How does a primary table field help me when the No Mapping scrambling type is used?

The primary table field is a required attribute for every rule regardless of the scrambling type used, but the Scrambling platform does not require a primary table field for the following scrambling types:

- No Mapping
- Delete Value
- Fixed value
- Manual 1:1 Mapping

Use the primary table to maintain the condition sets for the rule in all the cases listed above except for No Mapping.

15.4 Scrambling Rules: Condition Sets

Can I create a condition set for a table using the fields of other tables?

No, a condition set is always applicable to the primary table.

Can I design condition sets for the offset value of a field?

No, a condition set always works on a complete value. You cannot design a condition set for the offset of a value.

However, you can use options such as CP.

Field Name	Option	Lower Value
VORNA	CP	N*

Figure 97: Using CP option in the condition sets

15.5 Scrambling Rules: Expert Technical Information

Which ready-to-use identifiers does SAP make available?

We provide provides the following identifiers:

- o **VENDNO** (Vendor objects scrambling)
- o **CUSTNO** (Customer objects scrambling)
- o **MOLGA** (AP HCM data scrambling based on the country of the employee)

Can I use identifiers delivered by SAP rules?

Yes, these identifiers can be used in customer-created rules in the following cases:

- o Vendor/customer address data scrambling rules
- o HCM data scrambling based on country rules

How do I maintain custom identifiers and use them in rules?

As of SP04, SAP does not enable the creation of custom identifiers. Only VENDNO, CUSTNO, and MOLGA can be used as identifiers.

What scrambling function modules are provided in the standard?

CNV_TDMS_SCR_PRESEL_ADDRESS (for vendor and customer address scrambling)

CNV_TDMS_SCR_PRESEL_HCM_DATA (for HCM scrambling based on country)

How can I use the standard scrambling function modules in their rules?

You can use the standard scrambling function module (FM) along with the identifiers for address and HCM data scrambling. Make sure that these FMs are not used in any other rules to avoid runtime errors and data inconsistencies.

How can I create a new scrambling function module and use it in a rule?

SAP provides a template function module `CNV_TDMS_SCR_PRESEL_TEMPLATE`. You can create a new FM by copying the interface of this standard template.

Why do I have to create a new scrambling function module?

If you want to override the standard mapping feature with your own custom logic, write the mapping logic in a Z function module and maintain the FM name in the scrambling rule.

See the standard scrambling FMs for more technical information about coding in the scrambling FM.

15.6 Miscellaneous

Can I execute the scrambling activities simultaneously in multiple packages in the Transfer scenario?

Yes, this type of execution is possible even if the different packages are using the same system and client as the sender system.

Can I execute the scrambling activities simultaneously in multiple packages in the Stand-Alone scenario?

No, this type of execution is not possible.

16 The Scrambling Configuration After Upgrade Tool

The Scrambling Configuration After Upgrade tool enables the migration of the scrambling rules you created in SAP TDMS 3.0 to the Scrambling workbench of SAP TDMS 4.0 with reduced effort.

For more information about how to configure scrambling rules created in SAP TDMS 3.0 for use in SAP TDMS 4.0, see the Upgrade Master Guide for SAP TDMS 4.0 on the Service Marketplace at <http://service.sap.com/instguides>

www.sap.com/contactsap

Material Number

© 2012 SAP AG. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

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

Microsoft, Windows, Excel, Outlook, and PowerPoint are registered trademarks of Microsoft Corporation.

IBM, DB2, DB2 Universal Database, System ads, System i5, System p, System p5, System x, System z, System z10, System z9, z10, z9, iSeries, pSeries, xSeries, zSeries, eServer, z/VM, z/OS, i5/OS, S/390, OS/390, OS/400, AS/400, S/390 Parallel Enterprise Server, PowerVM, Power Architecture, POWER6+, POWER6, POWER5+, POWER5, POWER, OpenPower, PowerPC, BatchPipes, BladeCenter, System Storage, GPFS, HACMP, RETAIN, DB2 Connect, RACF, Redbooks, OS/2, Parallel Sysplex, MVS/ESA, AIX, Intelligent Miner, WebSphere, Netfinity, Tivoli and Informix are trademarks or registered trademarks of IBM Corporation. Linux is the registered trademark of Linus Torvalds in the U.S. and other countries.

Adobe, the Adobe logo, Acrobat, PostScript, and Reader are either trademarks or registered trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Oracle is a registered trademark of Oracle Corporation.

UNIX, X/Open, OSF/1, and Motif are registered trademarks of the Open Group.

Citrix, ICA, Program Neighborhood, MetaFrame, WinFrame, VideoFrame, and MultiWin are trademarks or registered trademarks of Citrix Systems, Inc.

HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

Java is a registered trademark of Sun Microsystems, Inc.

JavaScript is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

SAP, R/3, xApps, xApp, SAP NetWeaver, Duet, PartnerEdge, ByDesign, SAP Business ByDesign, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary.

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