

SAP Identity  
Management 8.0  
SP7 and higher  
March 2021

## **SAP Identity Management 8.0 – Setup and Provisioning SP4 (DL1)**

Building Block Configuration Guide

SAP SE  
Dietmar-Hopp-Allee 16  
69190 Walldorf  
Germany

## Copyright

© 2016 SAP SE or an SAP affiliate company. All rights reserved.

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

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






National product specifications may vary.

These materials are provided by SAP SE 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.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE in Germany and other countries. Please see <http://www.sap.com/corporate-en/legal/copyright/index.epx#trademark> for additional trademark information and notices.

Revision	Change Date	Description
0	2016-03-15	NN: Initial Creation
1	2016-03-29	RF: High level review, small updates + comments
2	2016-07-26	RF: Final Developer Review
3	2018-02-13	NN: Updates SP2
4	2018-08-03	NN: Updates SP2 patch 2
5	2020-01-23	NN: Updates for SP3
6	2020-11-25	NN: Updates for SP4
7	2021-03-02	NN: Updates for Mass Admin Configuration of Custom Jobs

## Icons

Icon	Meaning
	Caution
	Example
	Note
	Recommendation
	Syntax

## Typographic Conventions

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

## Content

1	Purpose .....	8
2	Preparation .....	8
2.1	Prerequisites .....	8
2.1.1	Import of SAP Standard Packages.....	8
2.2	Unzip the Solution Package .....	9
2.3	Import the IDM RDS Packages .....	9
3	Initial Configuration.....	11
3.1	Initial System Setup.....	11
3.1.1	Changing Settings of Standard Attributes.....	11
3.1.2	Initial Configuration of Folder Structure .....	12
3.1.3	Setting IDM RDS Package Constants .....	13
3.1.4	Create Minimum Data in the System.....	15
3.1.5	Import Additional Configuration Packages.....	18
3.1.6	Web UI Login .....	18
3.1.7	Import of Database enhancements .....	19
3.1.8	Modify Grants on Database Tables.....	21
3.1.9	Enable Attribute Eventing .....	22
3.1.10	Check and Update Special Attribute Settings.....	22
3.1.11	Configure Password Policy .....	24
3.2	E-Mail Notification Configuration .....	24
3.2.1	Prerequisites .....	24
3.2.2	The Script sapc_sendNotification.....	25
3.2.3	Customize E-Mail Templates .....	25
3.2.4	Job Error Handler .....	26
3.2.5	Job Result Notification .....	26
3.2.6	SAP PF Notifications .....	27
4	System Connectivity – Source Systems.....	28
4.1	SAP HCM Integration .....	28
4.2	Microsoft Active Directory / LDS.....	28
4.3	SAP AS ABAP .....	28
4.4	SAP AS Java.....	29
4.5	SAP SuccessFactors .....	29
4.6	Text File Based Upload .....	29
5	System Connectivity – Backend Systems .....	30
5.1	General Rules .....	30
5.1.1	Job Execution General .....	30
5.1.2	Initial Load Job .....	30
5.1.3	Update Load Job .....	31
5.1.4	SYSTEM Privilege Settings .....	32
5.1.5	Handling of System Specific Attributes.....	32
5.2	Connecting an SAP AS ABAP System.....	33

5.2.1	General Information.....	33
5.2.2	Repository Type ABAP and ABAP Business Suite .....	33
5.2.3	Preparation Steps.....	34
5.2.4	Cleanup of Inconsistencies in the ABAP System.....	35
5.2.5	Creating the SAP AS ABAP Repository .....	35
5.2.6	Repository Jobs for SAP AS ABAP .....	37
5.2.7	Post Load Configuration Steps .....	39
5.3	Connecting an SAP AS Java System.....	41
5.3.1	General Information.....	41
5.3.2	Preparation Steps.....	41
5.3.3	Creating the SAP AS Java Repository .....	42
5.3.4	Repository Jobs for SAP AS Java.....	43
5.3.5	Post Load Configuration Steps .....	44
5.4	Connecting an SAP HANA System .....	46
5.4.1	General Information.....	46
5.4.2	Preparation Steps.....	46
5.4.3	Creating the SAP HANA Repository .....	48
5.4.4	Repository Jobs for SAP HANA .....	49
5.4.5	Post Load Configuration Steps .....	50
5.5	Connecting an SAP SuccessFactors System.....	51
5.5.1	General Information.....	51
5.6	Connecting Microsoft Active Directory / LDS .....	52
5.6.1	General Information.....	52
5.6.2	Preparation Steps.....	53
5.6.3	Creating the AD / LDS Repository.....	53
5.6.4	Repository Jobs for AD / LDS .....	54
5.6.5	Post Load Configuration Steps .....	55
5.7	Post System Connection Steps.....	57
5.7.1	UI Forms for System Specific Attributes.....	57
5.8	Removing a Backend System .....	61
6	Approval Configuration .....	66
6.1	Approval Workflows of IDM RDS Package.....	66
6.2	Required Master Data for Approval Workflows.....	66
6.3	Enable Approval Workflow .....	66
6.4	Configuration of Approval Steps.....	68
7	Further Configuration Details .....	69
7.1	Script sapc_localStandardScriptsContainer.....	69
7.2	Configuration of System Specific Attributes.....	69
7.3	Presentation Types for Attributes .....	70
7.4	Attribute Value Help for Attribute SAPC_ROLE_CATEGORY_SYS_LANDSCAPE	70
8	Mass Administration Jobs.....	72

8.1	The Request Object.....	72
8.1.1	Attributes SAPC_REQ_STATE & STATE_INFO .....	72
8.1.2	Error Mail Template SAPC_MASS_JOB_ERROR.....	72
8.1.3	The Process / Workflow.....	73
8.2	Setting Up a Custom Mass Administration Job.....	74
9	Transport.....	77
9.1	Considerations Regarding System Specific Attributes .....	77
10	Appendix .....	78
10.1	Overview of System Specific Attributes.....	78
10.1.1	System Specific Attributes for SAP AS ABAP .....	78
10.1.2	System Specific Attributes for SAP AS Java .....	78
10.1.3	System Specific Attributes for SAP HANA.....	79
10.1.4	System Specific Attributes for AD / LDS.....	79

# Identity Management 8.0 - Setup and Provisioning: Configuration Guide

## 1 Purpose

The purpose of this document is to describe the general configuration steps required to manually set up the configuration for the SAP IDM 8.0 rapid-deployment solution within the system landscape that has already been installed using the corresponding installation or configuration guides for installation.

## 2 Preparation

### 2.1 Prerequisites

#### 2.1.1 Import of SAP Standard Packages

The SAP Provisioning Framework and its Schema are required to be installed on the Identity Management system before installing the content of this rapid deployment solution. In version SAP IDM 8.0 the SAP Provisioning Framework does not consist of only one single package but is a combination of several standard packages provided with the SAP IDM 8.0 standard product.

Following packages need to be installed (in the below order):

- SAP Provisioning Framework:  
[http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/7a/a8af2aa16e4996973ad7df64b8b569/content.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/7a/a8af2aa16e4996973ad7df64b8b569/content.htm)
  - com.sap.idm.provisioning.engine
  - com.sap.idm.util.notification
  - com.sap.idm.connector.custom
  - Further connectors according to the project requirements (e.g. package com.sap.idm.abap if you intend to connect SAP AS ABAP systems to the Identity Management solution).

Additionally those packages can be installed:

- com.sap.idm.forms.default
- com.sap.idm.forms.html5

Further explanations about the package structure of the SAP Provisioning Framework you can find here:

[http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/19/0596659b1245399f0fed52b4349154/content.htm?frameset=/en/19/0596659b1245399f0fed52b4349154/frameset.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/19/0596659b1245399f0fed52b4349154/content.htm?frameset=/en/19/0596659b1245399f0fed52b4349154/frameset.htm).

It is recommended to put all the packages of the SAP Provisioning Framework into a subfolder like shown on the picture below.





**Figure 1 - Recommended Folder Structure in Eclipse - SAP**

An explanation about the package structure of the rapid deployment solution can be found in Figure 2 - Recommended Folder Structure in Eclipse.

## 2.2 Unzip the Solution Package

The content of the solution package is shipped in a ZIP file. Unzip the file in the file system of your SAP Identity Management system.

The solution package contains the following file structure (which is referenced in subsequent chapters):

- Database Related Files
  - Contains new views and stored procedures for Microsoft SQL Server and Oracle



Please have a look at SAP note 2297163 - *SAP NetWeaver Identity Management rapid-deployment solution V2.80 (IDM 8.0)*

(<https://launchpad.support.sap.com/#/notes/1691375/E>) for more details on database specific limitations and specialties.

- RDS Configuration Packages
  - Contains the files needed for the import of the solution package into your system
- Templates
  - Contains various templates shipped with the solution package, such as e-mail notification templates, and \*.csv file templates for the upload jobs
- Examples
  - Contains \*.csv upload files with sample data

## 2.3 Import the IDM RDS Packages

In this step, you import the following files into your SAP Identity Management system:

- *IDM80\_Identity\_Store\_Schema\_v<xx>.idmschema*

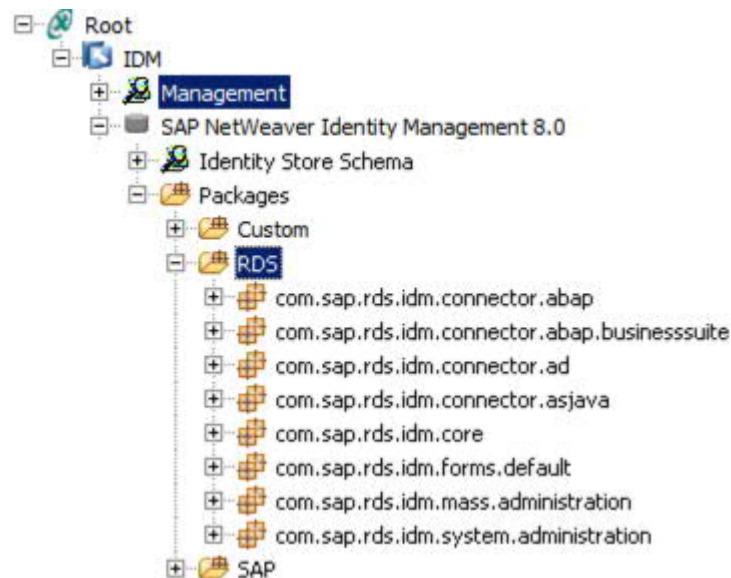
Import the identity store schema of the rapid-deployment solution:

1. Open Eclipse.
2. Go to your identity store where you want to import the solution package.

3. Right click on *Identity store schema*.
4. From the context menu, choose *Import...*
5. Select *IDM80\_RDS\_Identity\_Store\_Schema\_V<xx>.idmschema* (always use the latest version) file and choose *Open*.

Import the rapid-deployment solution packages:

1. Open Eclipse.
2. Go to your identity store where you want to import the solution package.
3. It is recommended to create a separate subfolder for the packages of the rapid-deployment solution as shown in the picture below:



**Figure 2 - Recommended Folder Structure in Eclipse - RDS**

4. In the context menu of the (sub-) folder where you want to import the packages, choose *Import...*
5. Select the packages to import and repeat the process for all required packages in the following order:
  - 5.1. `com.sap.rds.idm.core`
  - 5.2. `com.sap.rds.idm.system.administration`
  - 5.3. `com.sap.rds.idm.configitem`
  - 5.4. `com.sap.rds.idm.mass.administration`



Other packages (for example the forms and approval packages) must only be imported after other initial configuration steps have been performed successfully (See section 3.1.5 - [ImportAdditionalConfigurationPackages](#)).



There might be missing dependencies during the package-import due to dependency-circles. Those will be resolved later, when all packages are imported.

## 3 Initial Configuration

### 3.1 Initial System Setup

#### 3.1.1 Changing Settings of Standard Attributes

It is recommended to check (and if needed change) some settings of standard SAP IDM attributes for better usability.

- Open the *Identity Management Developer Studio* (Eclipse)
- Go to your active identity store, choose the following navigation path: *Identity Store Schema → Attributes* or *Identity Store Schema → Entry Types*
- Open the attributes / entry types as listed below and change the recommended settings as shown in the table below
- For more details about the different settings please have a look at the official documentation in the SAP Help Portal

Attribute / Entry Type	Recommended Settings
MXREF_MX_PRIVILEGE	Tab <i>Presentation</i> : <ul style="list-style-type: none"> <li>• Activate <i>Show Search Field</i></li> <li>• If needed activate <i>List entries on Load</i></li> </ul> Tab <i>Assignments</i> : <ul style="list-style-type: none"> <li>• Activate <i>Show Advanced Search</i></li> <li>• Set <i>Status Filter</i> to <i>All</i></li> </ul>
MXREF_MX_ROLE	Same as MXREF_MX_PRIVILEGE
MXREF_MX_GROUP	Tab <i>Presentation</i> : <ul style="list-style-type: none"> <li>• Activate <i>Show Search Field</i></li> <li>• If needed activate <i>List entries on Load</i></li> </ul>
MXREF_MX_DYNAMIC_GROUP	Same as MXREF_MX_GROUP
MXMEMBER_MX_PERSON	Same as MXREF_MX_PRIVILEGE
MX_ASSIGNMENT	Tab <i>Presentation</i> : <ul style="list-style-type: none"> <li>• Activate <i>Show Search Field</i></li> <li>• If needed activate <i>List entries on Load</i></li> <li>• <i>Number of Lines</i>: 10</li> </ul> Tab <i>Assignments</i> : <ul style="list-style-type: none"> <li>• Activate <i>Show Advanced Search</i></li> <li>• Set <i>Status Filter</i> to <i>All</i></li> <li>• Activate <i>Include Direct Assignments Only</i></li> </ul>
MX_PERSON	Tab <i>Attributes</i> : <ul style="list-style-type: none"> <li>• MXREF_* attributes → button <i>Edit Property</i></li> <li>• Check and adapt the settings based on your requirements</li> <li>• Please be aware that not all MXREF_* attributes have the <i>Edit Property</i> button enabled</li> </ul>
MX_ROLE	Same as MX_PERSON



The option *List entries on Load* can lead to negative performance impacts, especially on the attribute MX\_ASSIGNMENT, because this attribute is calculating the whole assignment hierarchy on load. It is still recommended to activate this option but in case the performance of the user interface is influenced negatively when loading UI forms, showing one of the attributes mentioned above, it might need to be deactivated again. In this case the attribute values are only loaded when clicking the *Search* button in the according UI forms.



The *Status Filter* setting allows you to see assignments that are not in status OK (for example pending and failed assignments). It might be confusing for the end user to not see them, for example if they want to apply for a role that they don't see (because the state is pending) but can also not request because they are requested already. Thus why it is recommended to set the *Status Filter* always to *All*, on all reference attributes used.

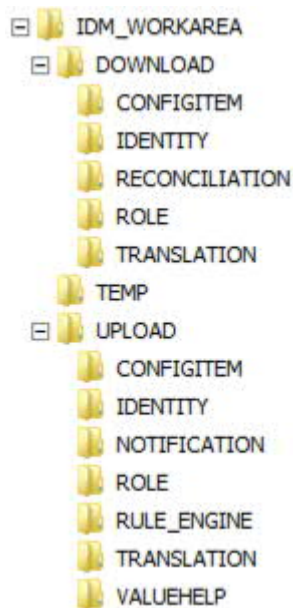
### 3.1.2 Initial Configuration of Folder Structure

To be able to execute mass administration jobs and certain system setup jobs, it is required to create folders in the file system that the dispatcher can access and configure the package constants of package *com.sap.rds.idm.mass.administration* accordingly (see section 3.1.3 [SettingIDMRDSPackageConstants](#)**Error! Not a valid bookmark self-reference.**).

1. Create a folder in your file system for file downloads (either on the SAP Identity Management server directly, or as a network share that is accessible from the SAP Identity Management server) (for example, *D:\IDM\_WORKAREA\DOWNLOAD*)
2. Create a folder in your file system for file uploads (either on the SAP Identity Management server directly, or as a network share that is accessible from the SAP Identity Management server) (for example, *D:\IDM\_WORKAREA\UPLOAD*)
3. Create a folder to store mail attachments temporarily (either on the SAP Identity Management server directly, or as a network share that is accessible from the SAP Identity Management server) (for example, *D:\IDM\_WORKAREA\TEMP*)
4. Create the necessary folder structure as shown on figure Figure 3 - Folder Structure on File below (for example, *D:\IDM\_WORKAREA\DOWNLOAD\IDENTITY*).

Necessary folders are:

- a. <download-folder>\IDENTITY
- b. <download-folder>\ROLE
- c. <download-folder>\RECONCILIATION
- d. <download-folder>\TRANSLATION
- e. <download-folder>\NOTIFICATION
- f. <download-folder>\CONFIGITEM
- g. <upload-folder>\IDENTITY
- h. <upload-folder>\ROLE
- i. <upload-folder>\RULE\_ENGINE
- j. <upload-folder>\TRANSLATION
- k. <upload-folder>\NOTIFICATION
- l. <upload -folder>\CONFIGITEM
- m. <upload -folder>\VALUEHELP



**Figure 3 - Folder Structure on File System**



The operating system user in which name the dispatchers are executed, needs to have full access to those folders in order to create, manipulate or read files (for example the windows system user of the system the dispatchers are running on).



Be aware of the different syntax of file system paths on Windows and UNIX based systems.



Other sub folders might be required later, if custom mass administration jobs are created for other entry types (more about custom mass administration jobs in section.

### 3.1.3 Setting IDM RDS Package Constants

The following package constants are required to be maintained in the SAP IDM Administration and Monitoring interface. To do that, open the Administration and Monitoring interface ([http\(s\)://<host>:<port>/idm/admin](http(s)://<host>:<port>/idm/admin)), change to tab *System Configuration*, select *Packages* in the left-hand pane, click *Go* (or filter first for your desired package) and select the package to set the constants as described below. Please be aware that you have to maintain constants of standard packages and IDM RDS packages.

- com.sap.idm.util.notification:** If you want to use E-Mail notification, please make sure you maintained the necessary details for the notification package of the standard SAP Provisioning Framework.  
 A detailed explanation of the standard package constants can be found here [http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/66/b9be6cac10458b8bd5aeaae1ef276f/content.htm?frameset=/en/66/b9be6cac10458b8bd5aeaae1ef276f/frameset.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/66/b9be6cac10458b8bd5aeaae1ef276f/content.htm?frameset=/en/66/b9be6cac10458b8bd5aeaae1ef276f/frameset.htm).
- com.sap.rds.idm.core:**

Constant	Default / Suggestion	Description
SAPC_MASTER_IDS_ID	1	ID of Master Identity Store for IDM RDS Package

Constant	Default / Suggestion	Description
SAPC_LOG_LEVEL	2 for development systems, 1 elsewhere	Log level used in tasks and jobs (0=off, 1=error, 2=info, 3=debug)
SAPC_ERROR_MAIL_SMTP_HOST		SMTP Settings for error mail notifications (only required if error mail notification is to be used)
SAPC_ERROR_MAIL_SMTP_PORT		
SAPC_ERROR_MAIL_ORIGINATOR	SAP IDM RDS 8.0 <donotreply@yourcompany.com>	
SAPC_ERROR_MAIL_RECEIVER		Semicolon separated list of error notification receivers


- **com.sap.rds.idm.system.administration:**

Constant	Default / Suggestion	Description
SAPC_ADMINISTRATOR_USER_NAME	Administrator	The name of the development administrator user (only single value is supported here; if more administrative users need to be setup this has to be done after RDS system setup)

- **com.sap.rds.idm.approval:**

Constant	Default / Suggestion	Description
SAPC_APPROVAL_ROLE	IDM_APPROVERS	(SAPC) Approval role. Members are used as approvers in "Role Member" approval step

- **com.sap.rds.idm.mass.administration:** package constants that are not mentioned in the table don't necessarily need to be changed. Please note that the folders set in these constants need to exist on the file system where the dispatcher(s) are running that will execute the mass administration and system setup jobs. More details about the mass administration in section [InitialConfigurationOfFolderStructure](#).

Constant	Default / Suggestion	Description
SAPC_PATH_UPLOAD	<drive>:/IDM_WORKAREA/UPLOAD	Master Folder for File Upload
SAPC_PATH_DOWNLOAD	<drive>:/IDM_WORKAREA/DOWNLOAD	Master Folder for File Download
SAPC_UPLOAD_USER	IDM_UPLOAD	MSKEYVALUE of user that is used for audit information of upload jobs. The default value IDM_UPLOAD is created by system setup job.  If this value needs to be adapted, the entered user needs to exist in SAP IDM database.

It is not necessary to change the other path variables. This is only necessary, if the folder structure and file names have to be different than the default values.



Be aware of the different syntax of file system paths on Windows and UNIX based systems.

### 3.1.4 Create Minimum Data in the System

#### Create Minimum Data

Execute the job *SAPC - Create Minimum Data* to create basic data required for next steps.

1. Copy the file *SAPC\_Form\_Privileges.csv* from the RDS templates folder to the *UPLOAD* folder created in step 3.1.2.
2. Go to the Monitoring and Administration Interface ([http\(s\)://<host>:<port>/idm/admin](http(s)://<host>:<port>/idm/admin)) and chose tab *System Configuration*.
3. In the left hand pane select *Packages* and search for package *com.sap.rds.idm.system.administration*.
4. In the lower section of the page select the tab *Jobs*.
5. Select job *SAPC Create Minimum Data* from the table shown in the left panel of the lower section, check that the dispatcher settings for the job are fine and run the job by clicking *Run Now* from the information section.

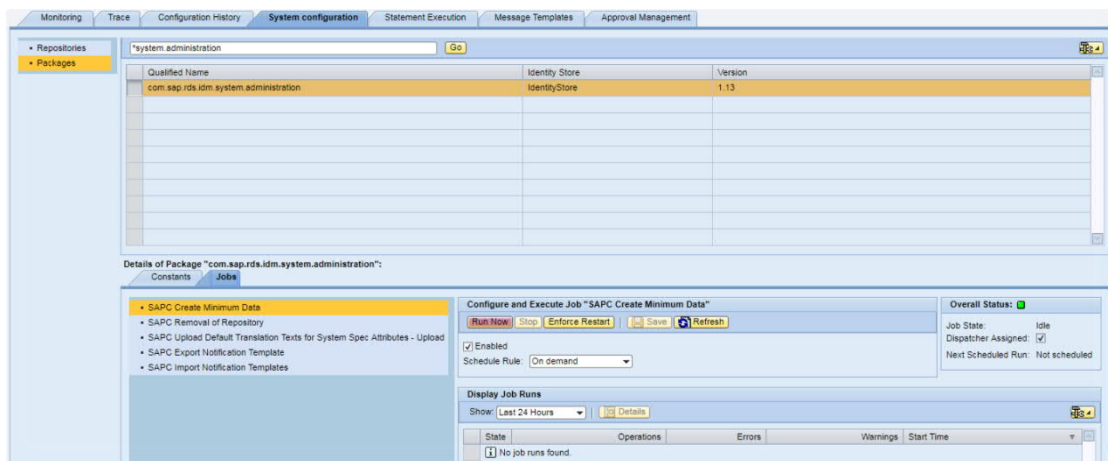


Figure 4 - Run Job from Admin UI

6. Check the job log for potential issues by selecting the job log on the lower section of the information section and clicking on details.

The result of the job is:

- MX\_PERSON IDM\_UPLOAD is created
- The following MX\_ROLE objects are created:
  - SAPC\_IDM\_ADMINISTRATORS
  - SAPC\_IDM\_DISPLAY\_ALL
  - SAPC\_IDM\_ESCALATION\_APPROVERS
  - SAPC\_IDM\_REPORT\_ALL
  - SAPC\_IDM\_SYSTEM\_ADMINISTRATORS
  - The privileges granting access to the RDS Forms will be created (*SAPC\_PRIV:FORM.\**) and assigned to the roles *SAPC\_IDM\_ADMINISTRATORS* and *SAPC\_IDM\_SYSTEM\_ADMINISTRATORS*

#### Upload Default Translation Texts

In order to set translation texts for attributes created in the schema only at runtime, when a system is being connected to Identity Management (for example License Type – HCM as the



English display value for the license type attribute for the repository HCM), it is necessary to upload a set of predefined translation values into the database.



The translation texts can be adapted in the file mentioned below before being uploaded into the database.

Execute the job *SAPC – Upload Default Translation Texts for System Spec Attributes* in order to have language translation templates in the database, which will be used later when the systems are connected.

1. Copy the file *SAPC\_Translations\_Upload\_SystemSpecificAttributes.csv* from the templates delivered with the RDS to the folder <your upload>\TRANSLATION and adapt it if necessary (for example if further languages are required).
2. Navigate to the jobs of the package *com.sap.rds.idm.system.administration* as shown in Figure 4 - Run Job from Admin UI.
3. Select the job *SAPC Upload Default Translation Texts for System Spec Attributes* and click *Run Now*.
4. Check the job log for potential issues.

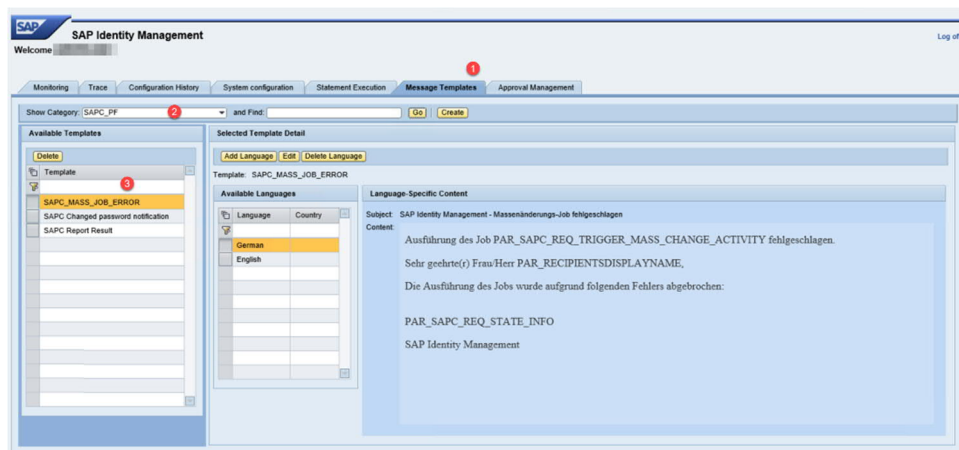
## Run Mass Administration Job Initialization

With SP2 for RDS for SAP IDM 8.0 the mass administration jobs have to be started as requests (SAPC Request) from the IDM UI. In order to proceed with the configuration, an initial load job has to be executed.

1. Navigate to the jobs of the package *com.sap.rds.idm.mass.administration* as shown in Figure 4 - Run Job from Admin UI.
2. Select the job *SAPC Setup Mass Operations* and click *Run Now*.
3. Check the job log for potential issues.

## Import Notification Templates

The notification templates of the IdM Business Extensions Service are supposed to be imported with the core package. Nevertheless, sometimes this import does not work as expected. Please check in the IdM Administration UI, whether the SAPC-Template class exist, and the templates are imported before executing the following steps. If the class and the templates are there already, please skip the following steps.



The former RDS comes with its own notification templates and its own template class. Those templates must be uploaded.

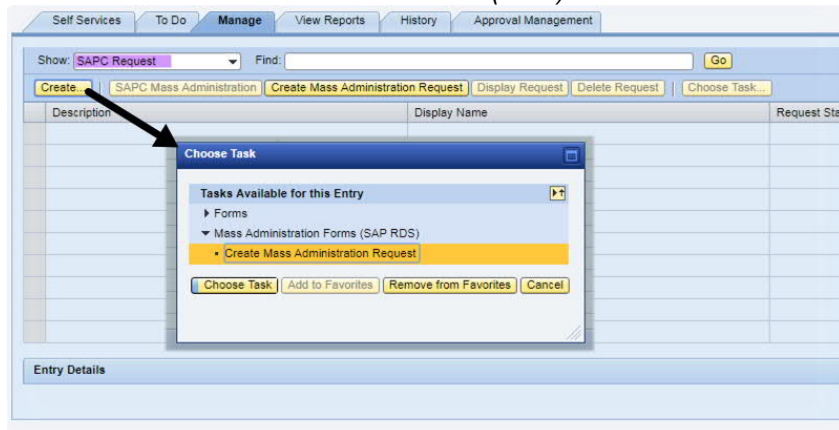
1. Copy the file *SAPC\_NotificationTemplateImport.csv* from the folder Templates/NOTIFICATION to the NOTIFICATION folder in your upload folder, created in step 3.1.2.
2. Navigate to the jobs of the package *com.sap.rds.idm.system.administration* as shown in Figure 4 - Run Job from Admin UI.
3. Select the job *SAPC Import Notification Templates* and click *Run Now*.
4. Check the job log for potential issues.



## Upload Attribute Value Help Data

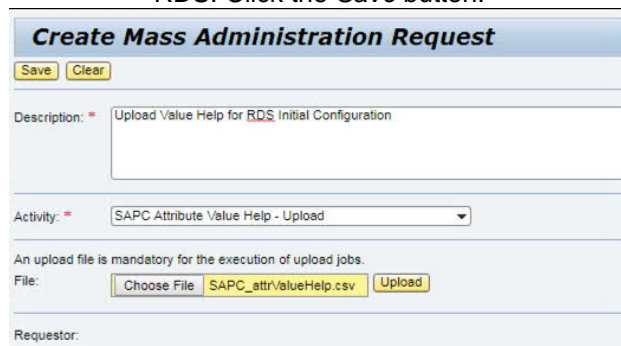
Certain attributes in the schema require value help information in the database. The job *SAPC Read AttrValueHelp* can be utilized to upload value help information for the attributes of the RDS and can also be used to upload custom attribute value help information, for custom attributes.

1. Log in to SAP Identity Management User Interface (IDM UI on *http(s)://<J2EE-host>:<J2EE-port>/idm*) and chose the *Manage* tab entry type *SAPC Request*.
2. Click the *Create...* button and chose the task *Create Mass Administration Request* from folder *Mass Administration Forms (RDS)*. Click on *Chose Task*.



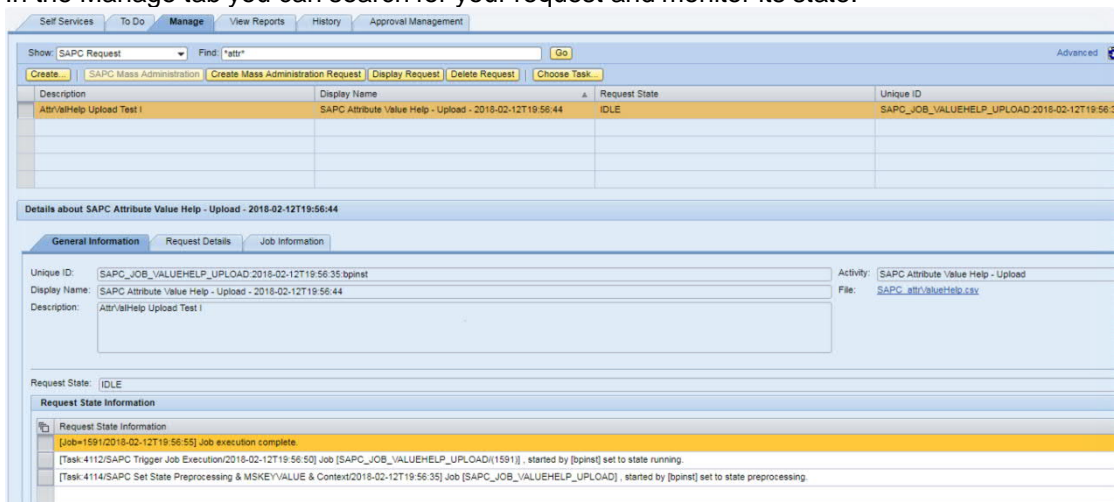
**Figure 5 - Run Mass Administration Job II**

3. In the opened form enter a description (for example Upload Value Help for RDS Initial Configuration). Choose activity *SAPC Attribute Value Help – Upload*. Click on *Choose File* and select the file *SAPC\_attrValueHelp.csv* from the templates delivered with the RDS. Click the *Save* button.



**Figure 6 - Run Mass Administration Job II**

4. In the Manage tab you can search for your request and monitor its state.




The screenshot shows the SAP Identity Management 'Manage' tab. At the top, there are tabs for 'Self Services', 'To Do', 'Manage', 'View Reports', 'History', and 'Approval Management'. Below these, there is a search bar with 'SAPC Request' and a 'Find' button. A table lists requests with columns for Description, Display Name, Request State, and Unique ID. The first row shows 'Attr/Help Upload Test I' with a display name of 'SAPC Attribute Value Help - Upload - 2018-02-12T19:56:44' and a state of 'IDLE'. Below the table, there is a section for 'Details about SAPC Attribute Value Help - Upload - 2018-02-12T19:56:44' with tabs for 'General Information', 'Request Details', and 'Job Information'. The 'General Information' tab is active, showing fields for Unique ID, Display Name, Description, Activity, and File. Below this, the 'Request State Information' section shows a log of events, including 'Job execution complete' and 'Job started by [bpinst] set to state running'.

5. Check the job log for potential issues.

### 3.1.5 Import Additional Configuration Packages

After the necessary roles and users have been created by the job *SAPC – Create Minimum Data*, you can import the remaining necessary packages. Those are:

- **com.sap.rds.forms.default:** All IDM RDS package forms for the SAP Identity SAP IDM UI.  
 Maintain constant `SAPC_NOTIFICATION_TEMP_FOLDER` of that package in order to be able to send mail attachments like report results.  
 Maintain constant `SAPC_SEND_PW_RESET_NOTIFICATION` of that package. If this constant is flagged true, a mail will be send to the person, a password reset has been requested for. It is recommend to be disabled if it is planned to use the standard notifications. The difference between standard notification mails and the RDS mail is that the RDS mail is one collective mail for all repositories, send at the time of requesting password reset, while the standard password reset notification is send after a successful password reset has been executed for a certain system. Nevertheless it is possible to use both notifications in parallel.
- **com.sap.rds.idm.rule.engine:** See section [RuleEngine](#) (if required).
- **com.sap.rds.idm.approval:** See section [ApprovalConfiguration](#) (if required).
- The required connector packages **com.sap.rds.idm.connector.\*** according to the project requirements.
- **com.sap.rds.idm.forms.systemspecific:** This package contains form templates for system specific attributes, which need to be customized for the customer implementation. It is recommended to import this package into another subfolder than the other packages (for example a folder called *Custom* like shown in the picture Figure 2 - Recommended Folder Structure in Eclipse).

### 3.1.6 Web UI Login

Use the following URLs to log on to the SAP Identity Management Web UI:

IDM 8.0 Web UI: `http(s):// <SAP Netweaver Java Server>:<port>/idm`

IDM 8.0 Web UI for Admin: `http(s)://<SAP Netweaver Java Server>:<port>/idm/admin`

### Prerequisites

- *Administrator* user

Has been created in SAP Identity Management with the *SAPC - Create Minimum Data* job and has role *SAPC\_IDM\_ADMINISTRATORS* assigned.

- Your own *UI user administration* user

You have to assign the role *SAPC\_IDM\_ADMINISTRATORS* to your user administration user to see all Web enabled tasks of this solution package.

### 3.1.7 Import of Database enhancements

There are two enhancements delivered with the rapid-deployment solution that you may need to import into your database. Determine which of those you need by the explanation below and perform the steps described next for the ones needed in your implementation:

- Database View *SAPC\_LINK\_EXT\_HYBRID*: A view that allows you to analyze user to privilege assignments in an enhanced view that shows not only assignments the user has, but also where the assignments are inherited from. This information is not visible in the standard *idmv\_link\_\** views. This view is not required for any of the RDS functionality but recommended to be used for data analysis and custom reporting.
- Stored Procedure *SAPC\_MC\_REPOSITORY\_DELETE*: A stored procedure allowing IDM to call standard procedure *mc\_repository\_delete* in order to allow job SAPC Removal of Repository to delete a repository.
- Stored Procedure *SAPC\_MXP\_ATTR\_DELETE*: A stored procedure allowing IDM to call standard procedure *mxp\_attr\_delete* in order to allow job SAPC Removal of Repository to delete system specific attributes.
- Stored Procedure *sapc\_mxp\_xadd\_link\_audit*: A stored procedure allowing the IDM RDS Approval Framework to store additional auditing information while executing the approval process. It is called in the processes *SAPC – Write Link Audit Information*: \* as shown in the picture below.

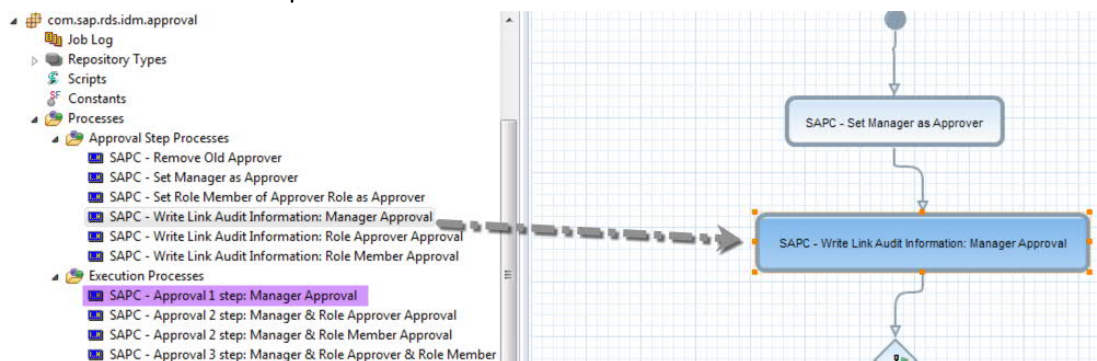


Figure 7 - Usage of Procedure *sapc\_mxp\_xadd\_link\_audit*

- This stored procedure is required to be deployed on the database, if the approval package of the solution is intended to be used, or if the same kind of enhanced link audit information is supposed to be stored in a custom approval workflow.

#### In case of Microsoft SQL Server:

1. Log on to the database using the *SQL Server Management Studio* tool for Microsoft SQL Server. For this operation, you need a user with administrative access rights (for example, user *mxmc\_oper*).
2. To open a new editor window, choose *New Query*.
3. Make sure that you are connected to the proper database of your SAP Identity Management installation (standard database is *mxmc\_db*).
4. Paste the content of the files *IDM\_BE\_sapc\_mxp\_add\_link\_audit\_MS-SQL.sql*, *IDM\_BE\_sapc\_link\_ext\_hybrid\_<lang>\_MS-SQL.sql*,

*IDM\_BE\_sapc\_mc\_repository\_delete\_MS-SQL.sql* and *IDM\_BE\_sapc\_mxp\_attr\_delete\_MS-SQL.sql* into the editor window. The scripts should be executed separately.

5. Execute the SQL code.
6. Verify that the code was executed without errors.

In case of Sybase ASE:

1. Log on to the database using the *SQL Server Management Studio* tool for Sybase ASE (for example iSQL or Squirrel). For this operation, you need a user with administrative access rights (for example, user *mxmc\_oper*).
2. To open a new editor window, choose *New Query*.
3. Make sure that you are connected to the database schema of your SAP Identity Management installation (standard database is *mxmc\_db*).
4. Paste the content of the files *IDM\_BE\_sapc\_mxp\_add\_link\_audit\_ASE.sql*, *IDM\_BE\_sapc\_link\_ext\_hybrid\_<lang>\_ASE.sql*, *IDM\_BE\_sapc\_mc\_repository\_delete\_ASE.sql* and *IDM\_BE\_sapc\_mxp\_attr\_delete\_ASE.sql* into the editor window. The scripts should be executed separately.  
Execute the SQL code.
5. Verify that the code was executed without errors.

In case of Oracle:

1. Log on to the database using the *Oracle SQL Developer*. For this operation, you need a user with administrative access rights (for example, user *mxmc\_oper*, for DB2: *IC\_OPER*).
2. To open a new editor window, choose *New Query*.
3. Make sure that you are connected to the proper database of your SAP Identity Management installation (standard database is *mxmc\_db*; for DB2 it is called *IC*).
4. Paste the content of the files *IDM\_BE\_sapc\_mxp\_add\_link\_audit\_ORACLE.sql*, *IDM\_BE\_sapc\_link\_ext\_hybrid\_<lang>\_ORACLE.sql*, *IDM\_BE\_sapc\_mc\_repository\_delete\_ORACLE.sql* and *IDM\_BE\_sapc\_mxp\_attr\_delete\_ORACLE.sql* into the editor window. The scripts should be executed separately.
5. Replace *&&PREFIX* with your database prefix (for example, **MXMC**).
6. Execute the SQL code.
7. Verify that the code was executed without errors.

In case of DB2:

1. Log on to the database using the *DB2 Data Studio*. For this operation, you need a user with administrative access rights (for example, user *mxmc\_oper*, for DB2: *IC\_OPER*).
2. Open a new project or use an existing one.
3. Right click *Stored Procedures* and chose *New -> Stored Procedure*. Chose *Language=SQL* and *Template=Deploy & Run IN/OUTparameters*
4. Enter the name for the stored procedure *sapc\_mxp\_xadd\_link\_audit*. Paste the content of the file *IDM\_BE\_sapc\_mxp\_link\_audit\_DB2.sql* into the editor pane.
5. Right click on the new procedure in the left handed navigation panel and chose *Deploy* from the context menu. Make sure that you deploy to the database schema of your SAP Identity Management installation (standard database for DB2 it is called *IC*).

6. Repeat steps 3 to 5 for script *IDM\_BE\_sapc\_mc\_repository\_delete\_DB2.sql* and *IDM\_BE\_sapc\_mxp\_attr\_delete\_DB2.sql*.
7. Open a query editor in DB2 Data Studio and connect to the Identity Center database.
8. Paste the content of the file *IDM\_BE\_sapc\_link\_ext\_hybrid\_<lang>\_DB2.sql* into the editor window.
9. Replace *&&PREFIX* with your database prefix (for example, *MXMC*).
10. Execute the SQL code.
11. Verify that the code was executed without errors.

### 3.1.8 Modify Grants on Database Tables

To enable following functionality, you need to add permissions to the runtime user on the database.

- Upload of translation values for attributes: This, for example, will be performed by the initial load jobs in order to set the language translation for the system specific attributes.
- Delete Repositories: This is required for the Job *SAPC Administration - Removal Of Repository* of package *com.sap.rds.idm.system.administration*.
- Add additional link audit information on approval workflows (see Figure 7 - Usage of Procedure *sapc\_mxp\_xadd\_link\_audit*)

Follow the steps below to perform those changes.

In case of Microsoft SQL Server or Sybase ASE:

1. Log on to the database using the *SQL Server Management Studio* tool for Microsoft SQL Server or the correlating tool for Sybase ASE (for example *Squirrel*). For this, you need a user with administrative access rights (user *sa* or the operating system user).
2. To open a new editor window, choose *New Query*.
3. Make sure that you are connected to the proper database of your SAP Identity Management installation (standard database is *mxmc\_db*).
4. Execute the following SQL code:
 

```
GRANT select,insert,update,delete ON mc_language_translations
TO <your db prefix>_rt_role
GRANT select on SAPC_LINK_EXT_HYBRID TO <your db
prefix>_rt_role
GRANT select on idmv_packages TO <your db prefix>_rt_role
GRANT execute on sapc_mxp_xadd_link_audit TO <your db
prefix>_rt_role
grant select on mcv_syslog_ext to <your db prefix> _rt_role
```
5. Verify that the code was executed without errors.

In case of Oracle or IBM DB2:

1. Log on to the database using the *Oracle SQL Developer* tool (or a correlating tool for DB2). For this, you need a user with administrative access rights (user *mxmc\_oper* or *IC\_OPER* for DB2 or the operating system user).
2. Make sure that you are connected to the proper database of your SAP Identity Management installation (standard database is *mxmc\_db*).  
Execute the following SQL code:
 

```
GRANT select,insert,update,delete ON mc_language_translations
```

```

TO <your db prefix>_rt_role;
GRANT select on SAPC_LINK_EXT_HYBRID TO <your db
prefix>_rt_role;
GRANT select on idmv_packages TO <your db prefix>_rt_role;
GRANT execute on sapc_mxp_xadd_link_audit TO <your db
prefix>_rt_role;
grant select on mcv_syslog_ext to <your db prefix> _rt_role
COMMIT;

```

3. Verify that the code was executed without errors.

### 3.1.9 Enable Attribute Eventing

You need to assign certain attributes in the system to an event task that inherits changes of the global attribute to the existing system-specific attributes.



For example if the global validity of a user gets extended, but the validity on the local system remains unchanged, the user will not be extended on the connected systems, because the validity on the system overrules the global validity. Assigning this event process as described in the following, will automate inheritance of changes to the global valid to date `MX_VALIDTO` to the valid to dates for the connected systems (`SAPC_IDEN_VALIDTO_$$$rep.NAME%`).

1. In your identity store, choose the following navigation path: *Identity store schema* → *Attributes*.
2. Open the attributes listed below, navigate to the *Event tasks* tab and link the following processes for the events *Add*, *Modify*, and *Delete* according to the event: `SAPC_Handle_System_Specific_Attribute_<operation>` (located in package `com.sap.rds.idm.core`)
  - `MX_VALIDFROM`
  - `MX_VALIDTO`

### 3.1.10 Check and Update Special Attribute Settings

The attributes `SAPC_IDEN_TEMP_RESET_PW_REPOSITORIES` and `SAPC_IDEN_REPOSITORIES_DISABLED_UI` contain a database-specific SQL statement, which, if necessary, has to be checked and adapted.

In your identity store, choose the following navigation path: *Identity store schema* → *Attributes*.

Open the attribute `SAPC_IDEN_TEMP_RESET_PW_REPOSITORIES` and choose the *Attribute values* tab and choose SQL Query.

Enter the following SQL statement or check and compare the existing one:

- For Microsoft SQL and Sybase ASE:
 

```

select rep_name, CASE WHEN displayName = '' THEN rep_name ELSE
(CASE WHEN displayName is null THEN rep_name ELSE displayName
END) END as displayName from
(select rep_id, rep_name from mc_repository, mc_repository_vars
where rep_id = Repository and varName = 'SAPC_REP_PRODUCTIVE'
and VarValue = '1'
and rep_name in (SELECT SUBSTRING(attrname,8,30) FROM
idmv_value_basic WHERE attrname LIKE 'ACCOUNT%' AND mskey =
%USERMSKEY%)) as repName
LEFT JOIN

```

```
(select Repository as rep_id, VarValue as displayName from
mc_repository_vars where VarName = 'SAPC_REP_DISPLAYNAME') as
displayName
on repName.rep_id = displayName.rep_id
order by displayName
```

- For Oracle and IBM DB2:

```
select rep_name, CASE WHEN displayName = '' THEN rep_name ELSE
(CASE WHEN displayName is null THEN rep_name ELSE displayName
END) END as displayName from
(select rep_id, rep_name from mc_repository, mc_repository_vars
where rep_id = Repository and varName = 'SAPC_REP_PRODUCTIVE'
and VarValue = '1'
and rep_name in (SELECT SUBSTR(attrname,8,30) FROM
idmv_value_basic WHERE attrname LIKE 'ACCOUNT%' AND mskey =
%USERMSKEY%)) repName
LEFT JOIN
(select Repository as rep_id, VarValue as displayName from
mc_repository_vars where VarName = 'SAPC_REP_DISPLAYNAME')
displayName
on repName.rep_id = displayName.rep_id
order by displayName
```

Repeat the above procedure for attribute *SAPC\_IDEN\_REPOSITORIES\_DISABLED\_UI*.

Open the attribute *SAPC\_IDEN\_TEMP\_JOB\_GUID* and choose the *Attribute values* tab and choose SQL Query.

Enter the following SQL statement or check and compare the existing one:

- For Microsoft SQL and Sybase ASE:

```
select jobs_.JobGuid,
CASE WHEN tasks_.TaskId is null AND jobs_.Repository is null
THEN CAST(jobs_.JobId as VARCHAR) + '/' + jobs_.Name
WHEN tasks_.TaskId is null AND jobs_.Repository is not null
THEN CAST(jobs_.JobId as VARCHAR) + '/' + jobs_.Name + ' (' +
jobs_.Repository + ')'
WHEN tasks_.TaskId is not null AND jobs_.Repository is null
THEN CAST(jobs_.JobId as VARCHAR) + '/' + jobs_.Name + ' (Task:
' + CAST(tasks_.TaskId as VARCHAR) + '/' + tasks_.TaskName +
')'
ELSE CAST(jobs_.JobId as VARCHAR) + '/' + jobs_.Name + '
(Task: ' + CAST(tasks_.TaskId as VARCHAR) + '/' +
tasks_.TaskName + ')' + ' (' + jobs_.Repository + ')'
END AS JobDescr
from
(select JobGuid, JobId, Name, mcPackageID, Repository
from mc_jobs where mcObsoletedTime is null )
jobs_
LEFT JOIN
(select TaskId, TaskName, JobGuid from mxp_tasks where
mcObsoletedGUID is null) tasks_
ON jobs_.JobGuid = tasks_.JobGuid
```

- For Oracle and IBM DB2:

```
select jobs_.JobGuid,
CASE WHEN tasks_.TaskId is null AND jobs_.Repository is null
THEN CAST(jobs_.JobId as VARCHAR(255)) || '/' || jobs_.Name
WHEN tasks_.TaskId is null AND jobs_.Repository is not null
THEN CAST(jobs_.JobId as VARCHAR(255)) || '/' || jobs_.Name ||
' (' || jobs_.Repository || ')'
```



```

        WHEN tasks_.TaskId is not null AND jobs_.Repository is null
        THEN CAST(jobs_.JobId as VARCHAR(255)) || '/' || jobs_.Name ||
        ' (Task: ' || CAST(tasks_.TaskId as VARCHAR(255)) || '/' ||
        tasks_.TaskName || ')'
        ELSE CAST(jobs_.JobId as VARCHAR(255)) || '/' || jobs_.Name
        || ' (Task: ' || CAST(tasks_.TaskId as VARCHAR(255)) || '/' ||
        tasks_.TaskName || ')' || ' (' || jobs_.Repository || ')'
        END AS JobDescr
    from
    (select JobGuid, JobId, Name, mcPackageID, Repository
        from mc_jobs where mcObsoletedTime is null )
    jobs_
LEFT JOIN
    (select TaskId, TaskName, JobGuid from mxp_tasks where
        mcObsoletedGUID is null) tasks_
ON jobs_.JobGuid = tasks_.JobGuid

```

Open the attribute `MX_MODIFYTASK_ATTR` and choose the *Attribute values* tab and choose SQL Query.

Enter the following SQL statement or check and compare the existing one:

- For Microsoft SQL and Sybase ASE:
 

```
select Attr_ID, CAST(Attr_ID AS VARCHAR) + '/' + AttrName
displayname from MXI_Attributes
```
- For Oracle and IBM DB2:
 

```
select Attr_ID, CAST(Attr_ID AS VARCHAR) || '/' || AttrName
displayname from MXI_Attributes
```

### 3.1.11 Configure Password Policy

It is recommended to adapt password requirements according to the needs by the system landscape to be provisioned. To do so please follow the procedure described here: [http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/b7/27a46854e242e2859cf1061befb10f/content.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/b7/27a46854e242e2859cf1061befb10f/content.htm).



The password policy has to be strong enough to cover the requirements of all systems to be provisioned. If the password policy is weaker than any system provisioned by SAP IDM, the provision actions might fail towards that system.

## 3.2 E-Mail Notification Configuration

A notification framework is provided by SAP Identity Management allowing to send mails on occurrence of provisioning events (for example creation of user or assignment of authorizations to a user) as well as approval and attestation events like a notification for approvers about a new approval item.

In addition to the standard notification abilities of the product, the rapid-deployment solution package comes with additional notification templates and abilities for further notification events that will be described in this section.

### 3.2.1 Prerequisites

The notification settings of the standard product are set up and maintained correctly. In order to perform the required actions, please follow the steps on [http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/66/b9be6cac10458b8bd5aeaae1ef276f/content.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/66/b9be6cac10458b8bd5aeaae1ef276f/content.htm).



When using the IDM RDS package you have to consider carefully which notifications you want to use for which events during provisioning. There are lots of standard notifications provided by the standard SAP Provisioning Framework as described in above documentation.

The notification package of the rapid-deployment solution is deployed on the system and the notification templates have been uploaded like described in **Error! Reference source not found..**

### 3.2.2 ~~Maintain the notification constants of package com.sap.provisioning~~ **section Changing Settings of Standard Attributes**

It is recommended to check (and if needed change) some settings of standard SAP IDM attributes for better usability.

- Open the *Identity Management Developer Studio* (Eclipse)
- Go to your active identity store, choose the following navigation path: *Identity Store Schema* → *Attributes* or *Identity Store Schema* → *Entry Types*
- Open the attributes / entry types as listed below and change the recommended settings as shown in the table below
- For more details about the different settings please have a look at the official documentation in the SAP Help Portal

Attribute / Entry Type	Recommended Settings
MXREF_MX_PRIVILEGE	Tab <i>Presentation</i> : <ul style="list-style-type: none"> <li>• Activate <i>Show Search Field</i></li> <li>• If needed activate <i>List entries on Load</i></li> </ul> Tab <i>Assignments</i> : <ul style="list-style-type: none"> <li>• Activate <i>Show Advanced Search</i></li> <li>• Set <i>Status Filter</i> to <i>All</i></li> </ul>
MXREF_MX_ROLE	Same as MXREF_MX_PRIVILEGE
MXREF_MX_GROUP	Tab <i>Presentation</i> : <ul style="list-style-type: none"> <li>• Activate <i>Show Search Field</i></li> <li>• If needed activate <i>List entries on Load</i></li> </ul>
MXREF_MX_DYNAMIC_GROUP	Same as MXREF_MX_GROUP
MXMEMBER_MX_PERSON	Same as MXREF_MX_PRIVILEGE
MX_ASSIGNMENT	Tab <i>Presentation</i> : <ul style="list-style-type: none"> <li>• Activate <i>Show Search Field</i></li> <li>• If needed activate <i>List entries on Load</i></li> <li>• <i>Number of Lines</i>: 10</li> </ul> Tab <i>Assignments</i> : <ul style="list-style-type: none"> <li>• Activate <i>Show Advanced Search</i></li> <li>• Set <i>Status Filter</i> to <i>All</i></li> <li>• Activate <i>Include Direct Assignments Only</i></li> </ul>
MX_PERSON	Tab <i>Attributes</i> : <ul style="list-style-type: none"> <li>• MXREF_* attributes → button <i>Edit Property</i></li> <li>• Check and adapt the settings based on your requirements</li> <li>• Please be aware that not all MXREF_* attributes have the <i>Edit Property</i> button enabled</li> </ul>
MX_ROLE	Same as MX_PERSON



The option *List entries on Load* can lead to negative performance impacts, especially on the attribute MX\_ASSIGNMENT, because this attribute is calculating the whole assignment hierarchy on load. It is still recommended to activate this option but in case the performance of the user interface is influenced negatively when loading UI forms, showing one of the attributes mentioned above, it might need to be

deactivated again. In this case the attribute values are only loaded when clicking the *Search* button in the according UI forms.



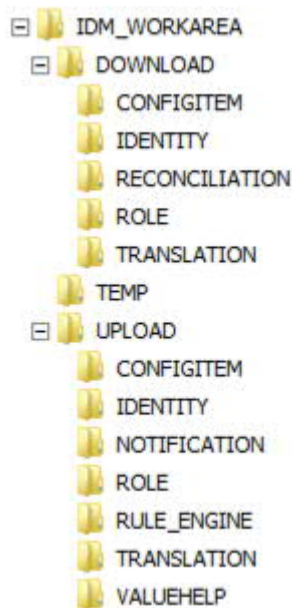
The *Status Filter* setting allows you to see assignments that are not in status OK (for example pending and failed assignments). It might be confusing for the end user to not see them, for example if they want to apply for a role that they don't see (because the state is pending) but can also not request because they are requested already. Thus why it is recommended to set the *Status Filter* always to *All*, on all reference attributes used.

### 3.2.3 Initial Configuration of Folder Structure

To be able to execute mass administration jobs and certain system setup jobs, it is required to create folders in the file system that the dispatcher can access and configure the package constants of package *com.sap.rds.idm.mass.administration* accordingly (see section 3.1.3 SettingIDMRDSPackageConstants**Error! Not a valid bookmark self-reference.**).

5. Create a folder in your file system for file downloads (either on the SAP Identity Management server directly, or as a network share that is accessible from the SAP Identity Management server) (for example, *D:\IDM\_WORKAREA\DOWNLOAD*)
6. Create a folder in your file system for file uploads (either on the SAP Identity Management server directly, or as a network share that is accessible from the SAP Identity Management server) (for example, *D:\IDM\_WORKAREA\UPLOAD*)
7. Create a folder to store mail attachments temporarily (either on the SAP Identity Management server directly, or as a network share that is accessible from the SAP Identity Management server) (for example, *D:\IDM\_WORKAREA\TEMP*)
8. Create the necessary folder structure as shown on figure Figure 3 - Folder Structure on File below (for example, *D:\IDM\_WORKAREA\DOWNLOAD\IDENTITY*).  
Necessary folders are:

- a. <download-folder>\IDENTITY
- b. <download-folder>\ROLE
- c. <download-folder>\RECONCILIATION
- d. <download-folder>\TRANSLATION
- e. <download-folder>\NOTIFICATION
- f. <download-folder>\CONFIGITEM
- g. <upload-folder>\IDENTITY
- h. <upload-folder>\ROLE
- i. <upload-folder>\RULE\_ENGINE
- j. <upload-folder>\TRANSLATION
- k. <upload-folder>\NOTIFICATION
- l. <upload -folder>\CONFIGITEM
- m. <upload -folder>\VALUEHELP



**Figure 3 - Folder Structure on File System**



The operating system user in which name the dispatchers are executed, needs to have full access to those folders in order to create, manipulate or read files (for example the windows system user of the system the dispatchers are running on).



Be aware of the different syntax of file system paths on Windows and UNIX based systems.



Other sub folders might be required later, if custom mass administration jobs are created for other entry types (more about custom mass administration jobs in section.

Setting IDM RDS Package Constants.

### 3.2.4 The Script `sapc_sendNotification`

The script `sapc_sendNotification` (content of the package `com.sap.rds.idm.core`) is an enhanced version of the script `sap_sendNotification` of the standard notification package. The enhancements are:

- Ability for context variables bigger than 2000 characters (`uGetContextVar` and `uSetContextVar` only support up to 2000 characters).
- Skipping approval complete notification if further approval tasks are outstanding for the current assignment. This is useful if the approval workflow is dynamic and can end at different points.
- Possibility to replace host and port in approval URL
- `sapc_sendNotification` supports notification parameters containing HTML coding. Those parameters will have same naming convention as others in the template (`PAR_...`) but the context variable has to be of name `#MSG_HTML_PAR_...` instead of `#MSG_PAR_...`.

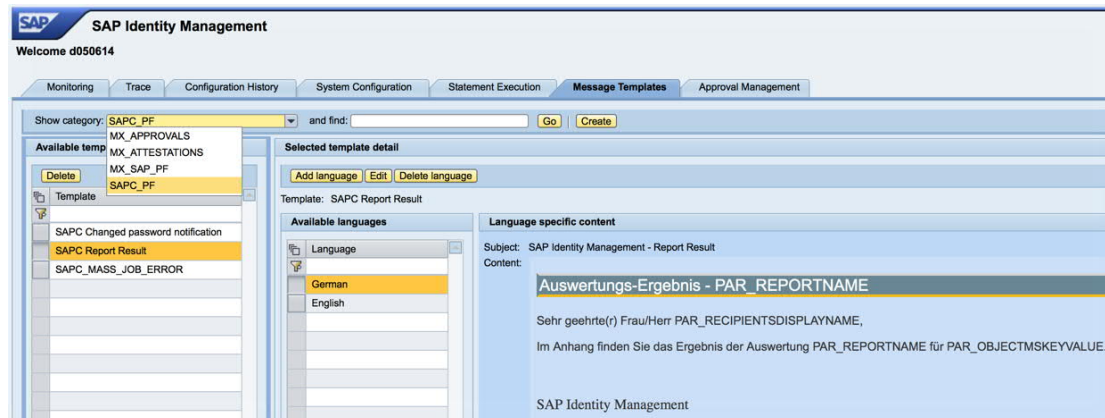
It is not possible to change the notification process for standard provisioning notifications (for example user creation or authorization assignments). If one of the mentioned enhancements is required in the standard notification on provisioning events, it is necessary to replace the

content of the script `sap_sendNotification` with the content of `sapc_sendNotification`. Thus but requires a change in the standard package and should only be performed if necessary.

On all other events like approval processes the notification process can be maintained and thus set to the notification process of the rapid-deployment solution in order to make use of those enhancements.

### 3.2.5 Customize E-Mail Templates

Notification templates can be maintained in the Monitoring and Administration interface of SAP Identity Management in the tab *Message Template*. The template categories and templates created by the steps executed in [UploadNotificationTemplates](#) are visible here and can be adapted to project needs.



Own notification templates can be created as well if this is required. The context variable `#MSG_TEMPLATE` needs to be set to the name of the template (example: `uSetContextVar("#MSG_TEMPLATE", "SAPC Report Result")`). A detailed example of the adaption of notification templates and the usage of its context variables can be found here:

<http://scn.sap.com/community/idm/blog/2014/03/06/using-the-assignment-notification-template-for-non-assignment-messages>.



The notification context variable `PAR_SAPC_PASSWORD` can also be used in standard notification templates, if the password reset has been initialized by the RDS process of package `com.sap.rds.forms.default`.

### 3.2.6 Job Error Handler

The Job error handler is a script that is writing job errors only into a file that will not be reset automatically on every job execution. The file name is `SAPC_Error_Log.txt` and it is located in the runtime folder of the job (`/%$ddm.ddm$path%`).

If that file exists it will be used later for the notification (see Job Result Notification).

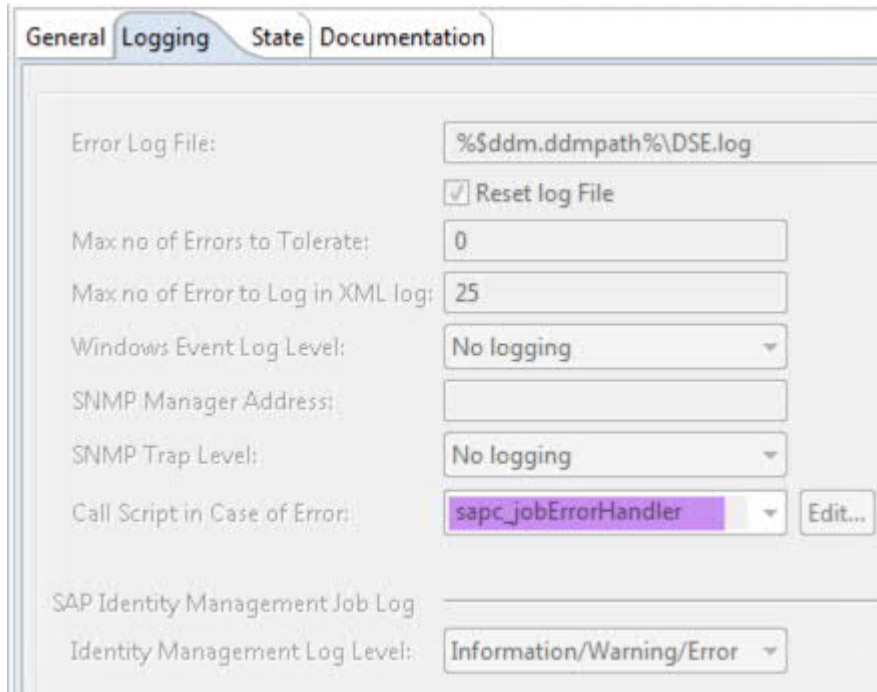


The reason why mail notifications are not send by that script itself is, that notifications would be send as much as errors occur during execution of the job.



Even though those files only capture errors, they will keep growing over time if job errors occur frequently. In that case, a frequent cleanup has to be setup, either as an SAP IDM job or as an operating system script).

In order to make use of that script in your job, it needs to be linked into the job scripts and set as script to be called in case of error on *Logging* section of the job properties.



Field	Value
Error Log File:	;%\$ddm.ddmpath%\DSE.log
Reset log File	<input checked="" type="checkbox"/>
Max no of Errors to Tolerate:	0
Max no of Error to Log in XML log:	25
Windows Event Log Level:	No logging
SNMP Manager Address:	
SNMP Trap Level:	No logging
Call Script in Case of Error:	sapc_jobErrorHandler
SAP Identity Management Job Log	<input checked="" type="checkbox"/>
Identity Management Log Level:	Information/Warning/Error

**Figure 8 - Set Job Error Handler Script**

### 3.2.7 Job Result Notification

After the completion of a job, a notification can be send to an administrator user, informing about the job result and errors by sending the log files (DSE log, Job log and SAPC\_Error\_Log).

In order to activate the notification on job results, link the script *sapc\_sendJobLog* of package *com.sap.rds.idm.core* into the scripts of your job. Copy the pass *SAPC Send Final Job Error Mail* from job *SAPC Send Final Job Error Mail* of package *com.sap.rds.idm.core* at the end of the job to be monitored and remove the prefix *Copy of*.

### 3.2.8 SAP PF Notifications

The standard notifications of the SAP Provisioning Framework can be used and might be combined with the IDM RDS package notification for the Password Reset scenario.

An overview about the SAP IDM standard notifications is available at the following link:

[http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/66/b9be6cac10458b8bd5aeaae1ef276f/content.htm?frameset=/en/b0/7fcc98fc24bb4aa7079da64754126/frameset.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/66/b9be6cac10458b8bd5aeaae1ef276f/content.htm?frameset=/en/b0/7fcc98fc24bb4aa7079da64754126/frameset.htm)

When you are using the RDS notifications you should disable the respective standard notification (*NOTIFYEVENT\_PASSWORD\_CHANGED* → *PF Changed password notification*).

## 4 System Connectivity – Source Systems

### 4.1 SAP HCM Integration

If you want to connect an SAP HCM System as your source system for personnel records being loaded as identity data into the SAP IDM system you have to configure the integration based on the standard SAP Provisioning Framework and following the standard documentation available at:

[http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/0c/0075a4a025422587257d16b22461ea/content.htm?frameset=/en/66/b9be6cac10458b8bd5aeaae1ef276f/frameset.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/0c/0075a4a025422587257d16b22461ea/content.htm?frameset=/en/66/b9be6cac10458b8bd5aeaae1ef276f/frameset.htm)



It is possible to handle the persons of the HCM system and the SU01 ABAP users separately. This might be necessary if not all persons have an SU01 account on the HCM system. This can be achieved by creating two separate repositories, one representing the HCM system (must be of type ABAP Business Suite) and one representing the underlying ABAP stack for the SU01 users, whereby the constant `NO_USER_ACCOUNT` is set to 1 on the repository representing the HCM system.



Manager information are stored in the attribute `MX_PERSONNEL_NUMBER_OF_MANAGER`. It is recommended to create an event process resolving the person to that personnel number and storing it on the attribute `MX_MANAGER`. An example implementation can be found in the folder *Attribute Modification Handler* of package `com.sap.rds.idm.core` with the name `SAPC Handle MX_FS_PERSONNEL_NUMBER_OF_MANAGER`.

### 4.2 Microsoft Active Directory / LDS

If you want to connect Microsoft Active Directory or LDS as your source system for user / identity data please follow the configuration steps to connect the system as backend system as described in chapter [Connecting Microsoft Active Directory / LDS](#).

The difference compared to using this system just as a backend system is that you clearly have to configure the *Initial Load Job* / *Update Job* accordingly to meet the projects requirements and allow local changes on the system to be reflected into SAP IDM. When using the system just as backend system you normally would not consider / allow local changes (or just in a limited manner).

Typical examples for such modifications are:

- Modify trigger on SYSTEM privilege
- Creation and/or assignment(s) of ACCOUNT and SYSTEM privilege
- Creation and/or assignment(s) of privileges
- Setting of attributes (such as ACCOUNT attribute)
- Eventing on privileges

### 4.3 SAP AS ABAP

If you want to connect an SAP AS ABAP system (special case: SAP CUA – Central User Administration) as your source system for user / identity data please follow the configuration steps to connect the system as backend system as described in chapter [Connecting an SAP AS ABAP System](#).

The difference compared to using this system just as a backend system is that you clearly have to configure the *Initial Load Job* / *Update Job* accordingly to meet the projects requirements and allow local changes on the system to be reflected into SAP IDM. When



using the system just as backend system you normally would not consider / allow local changes (or just in a limited manner).

Typical examples for such modifications are:

- Modify trigger on SYSTEM privilege
- Creation and/or assignment(s) of ACCOUNT and SYSTEM privilege
- Creation and/or assignment(s) of privileges
- Setting of attributes (such as ACCOUNT attribute)
- Eventing on privileges

## 4.4 SAP AS Java

If you want to connect an SAP AS Java system / SAP Enterprise Portal as your source system for user / identity data please follow the configuration steps to connect the system as backend system as described in chapter [Connecting an SAP AS Java System](#).

The difference compared to using this system just as a backend system is that you clearly have to configure the *Initial Load Job* / *Update Job* accordingly to meet the projects requirements and allow local changes on the system to be reflected into SAP IDM. When using the system just as backend system you normally would not consider / allow local changes (or just in a limited manner).

Typical examples for such modifications are:

- Modify trigger on SYSTEM privilege
- Creation and/or assignment(s) of ACCOUNT and SYSTEM privilege
- Creation and/or assignment(s) of privileges
- Setting of attributes (such as ACCOUNT attribute)
- Eventing on privileges

## 4.5 SAP SuccessFactors

To use SAP SuccessFactors as your source system please follow the configuration steps of the standard documentation:

[http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/bc/7c98740824425494df38ec8a428e97/content.htm?frameset=/en/f9/0d3de3dc0f47c6bbcc17c4e83f27f7/frameset.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/bc/7c98740824425494df38ec8a428e97/content.htm?frameset=/en/f9/0d3de3dc0f47c6bbcc17c4e83f27f7/frameset.htm)

## 4.6 Text File Based Upload

If you want to use a text file as the source of user / identity information (for instance based on the file extract from another system that cannot easily be integrated directly with SAP IDM) you can either use and adapt the already existing IDM RDS mass administration job *SAPC Identity Upload – Creation* or you have to create the respective upload job(s) yourself.

## 5 System Connectivity – Backend Systems

The following chapters provide details on how to connect the different supported backend systems to the SAP IDM system.

### 5.1 General Rules

#### 5.1.1 Job Execution General

Please be aware of the following exceptions and special cases when executing the jobs described in the next chapters:

- In certain jobs a cleanup of existing temporary database tables is performed at the beginning. During the first execution of such jobs (when those temporary database tables do not exist yet) you might see error messages like this:
  - SQL Update failed. SQL:delete from sapc\_recon\_ADDEMO\_idm\_users  
java.lang.Throwable: Invalid object name 'sapc\_recon\_ADDEMO\_idm\_users'.

#### 5.1.2 Initial Load Job

The following specialties have to be considered when executing the *Initial Load* job of any given backend system.



Certain information will be stored globally (repository independent) but might differ between the repositories. An identity, for example, can have different display names on different systems. Global information will only be set on new users (users that will be created in SAP IDM by the load job itself). Existing users will only be updated with their system specific attributes. That means, the information loaded from the first backend system has higher priority than information from systems that are connected to SAP IDM later. The system with the most accurate data should typically be chosen to be connected first.



The *Initial Load* job has been designed to be executed once only. If it is required to synchronize information from the backend system into SAP IDM more than once or regularly, the *Update Job* needs to be utilized.



It is possible that the *Initial Load* job needs to be adapted. If this is required, it is recommended to create a copy of the job in a custom package. In order to be able to copy the job to your custom package it is required to check out the RDS package where the job needs to be copied from. Nevertheless, that checkout can



be reverted by choosing the option *Revert Package* from the context menu of the package after the job has been copied.

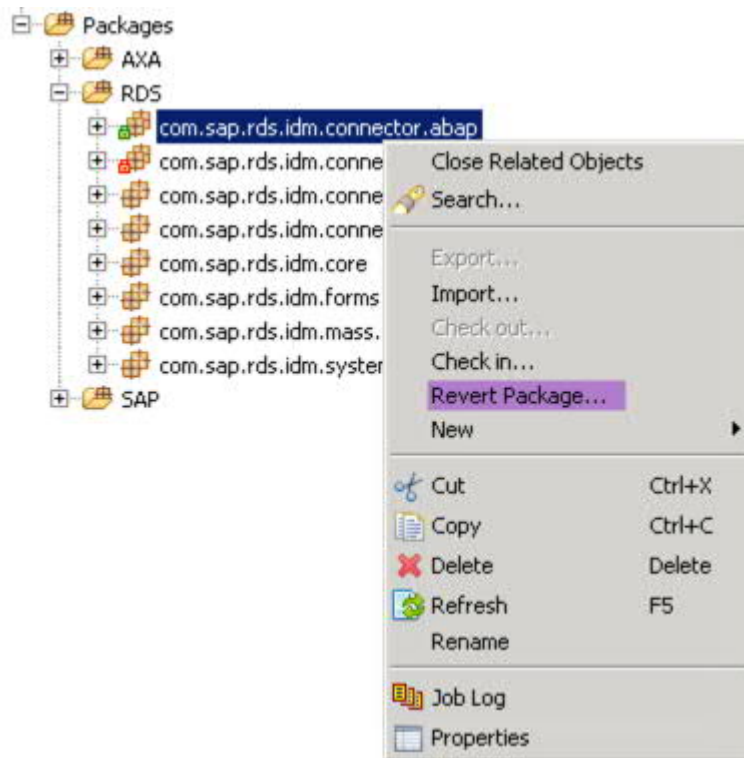


Figure 9 - Reverting a Package



The *Initial Load* job can be executed in several separate steps. This can be done by enabling the pass *Enable this pass to interrupt the execution of the job at this point*, to stop the execution in between. This requires the package that contains the job to be checked out.



Check which value help data is required for your scenarios and enable or disable the passes as required. This requires the package that contains the job to be checked out. Example for SAP AS ABAP:

- Groups
- Printer
- License Type
- Start Menu

### 5.1.3 Update Load Job

The following specialties have to be considered when executing the *Update* job of any given backend system.



The *Update* job is utilizing the delta mechanism of SAP IDM, which automates handling of removed items like privileges and privilege assignments from the target system. It not only creates new privileges but also removes those that have been deleted from the target system.



The delta mechanism has a security function to prevent malfunction. It has a maximum of items to be deleted. For example if no roles could be read from a connected system, IDM will not delete all roles for that system but print a warning message in the job log informing about the not performed delete operations due to overstepping delta limitations. If this is the case you can either temporarily increase the limit of items allowed to be deleted, or clean up removed items manually (for example by deleting a bunch of roles with a custom job). The chance of this issue to arise is high during authorization management projects.



The *Update* job is utilizing the build in delta mechanism of SAP IDM. This allows IDM to automatically delete privileges and privilege assignments in the database, if they have been deleted on the target system during the execution of the *Update Job*. If the execution fails, it is recommended to reset the delta in order to avoid malfunction of the job. If this is required to be done, the job “*SAPC – <repository type> Reset Delta*” needs to be executed from the *Administration and Monitoring Interface* as described in [ExecuteRepositoryJob](#).



The *Update Job* can be enabled to read modified user attributes (that have been changed locally on the backend system) into SAP IDM. This can be done in the pass that is writing the user data to the SAP IDM identity store (example for SAP AS ABAP: *WriteABAPUsers [modified]*). By default all attributes are configured in a way that only new entries will be written but already existing entries will not be updated. You have to enable each attribute (by removing the dot in front of the attribute) that is needed based on customer requirements.

### 5.1.4 SYSTEM Privilege Settings

The system privilege, which is created by the Initial Load job is holding a reference to all attributes that are update relevant (MODIFY event) for the according system. This means, if one of those attributes is changed at an identity, SAP IDM will trigger a modify event to synchronize that change to the connected system. It is possible to adapt the list of preconfigured attributes in the pass *Update System Privilege trigger attributes [modified]*. This requires the package that contains the job to be checked out.



You can either include a list of attributes by using the script *sap\_core\_setPrivilegeModifyTriggerAttributes*, or you can exclude a list of attributes by using the script *sap\_core\_setPrivilegeModifyTriggerIgnoreAttributes*. A good approach is to use as few as possible to avoid unnecessary events being triggered in the system.

An overview about the system specific attributes of the IDM RDS package and a copy and configuration template for each supported repository type can be found in document *IDM\_RDS\_List of system specific attributes.xlsx* available in IDM RDS package folder “*Examples*”.

### 5.1.5 Handling of System Specific Attributes

The IDM RDS package supports the handling of system specific attributes for each connected backend system. This means that you can have a global attribute value for a given attribute and a system specific attribute value for the same attribute.

A good example for this logic is the validity and the lock status of a user:

- The user can have a global validity like this:
  - Valid from (MX\_VALIDFROM): 01.01.2016
  - Valid to (MX\_VALIDTO): 31.12.2020

- Lock status (MX\_LOCKED): unlocked
- But for one special consolidation system the user has to be locked and limited for a given time period; to support such use cases the system specific attributes can be used. Let's say the consolidation system is called BST100 and the system specific attributes for this system can have the following values:
  - Valid From BST100: 01.01.2016
    - Attribute Name: SAPC\_IDEN\_REP\_VALIDFROM\_BST100
  - Valid To BST100: 31.05.2016
    - Attribute Name: SAPC\_IDEN\_REP\_VALIDTO\_BST100
  - Lock status BST100: locked
    - Attribute Name: SAPC\_IDEN\_REP\_DISABLED\_BST100

When SAP IDM provisions the user data to the backend system the logic how to retrieve which attribute values should be written works like this:

- First IDM checks if there is a system specific attribute value for the given attribute and the given system; if yes this system specific value is used
- If no system specific value is available it checks for the respective global attribute value; if no global attribute value is available nothing will be written
- That means the system specific attribute value has a higher priority than the global attribute value

What happens if global attribute values are changed in SAP IDM?

- You can configure certain global attributes (depending on your requirements) in a way that a global attribute value is automatically inherited to the system specific attributes
- How to configure this eventing is described in chapter 3.1.9 - Enable Attribute Eventing

The system specific attributes are created by the Initial Load job for each repository.

The list of system specific attributes that are created and supported by the standard IDM RDS package configuration are described in document *IDM\_RDS\_List of system specific attributes.xlsx* provided in IDM RDS Package folder "Examples".

## 5.2 Connecting an SAP AS ABAP System

### 5.2.1 General Information

The connectivity to an SAP AS ABAP system is based on the standard SAP Provisioning Framework containing specific enhancements. For the general usage of the SAP Provisioning Framework, you can refer to the standard documentation that describes the architecture, technical overview, and detailed configuration of SAP Provisioning Framework here:

[http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/ce/3f5d445c824350b8a3a436ff7ede3f/content.htm?frameset=/en/ce/3f5d445c824350b8a3a436ff7ede3f/frameset.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/ce/3f5d445c824350b8a3a436ff7ede3f/content.htm?frameset=/en/ce/3f5d445c824350b8a3a436ff7ede3f/frameset.htm)

The subsequent sections in this guide contain detailed information about settings specific to this solution package. When going through these steps, the reader is assumed to be familiar with the standard SAP Provisioning Framework.

Please make yourself familiar with the restrictions that apply to all connectors and to SAP AS ABAP connector only.

Restrictions that Apply to All Connectors:

[http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/e5/4b384d2cc14813891fb687bbe3d587/content.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/e5/4b384d2cc14813891fb687bbe3d587/content.htm)

Restrictions for SAP AS ABAP System Connectors:

[http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/20/14e54bf1e2424dbd70e1b51fab6cd3/content.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/20/14e54bf1e2424dbd70e1b51fab6cd3/content.htm)

### 5.2.2 Repository Type ABAP and ABAP Business Suite

Repository types are similar to what was called repository templates in previous versions of SAP Identity Management.

There are two different repository types that can be used in order to connect an ABAP system to SAP Identity Management: *SAPC\_ABAPSystem* and *SAPC\_ABAPBusinessSuite*. A brief explanation about the differences between those packages can be found in the following section. A more detailed explanation can be found in the document *DL0\_IDM80\_BB\_ConfigGuide\_EN\_XX.docx* (Master Guide).

Both types support dual stack systems and connection via message server as well as connection via load balancing and come with the same set of jobs and functionalities.

## Repository Type *SAPC\_ABAPSystem*

- Using Java Class toSAP which is calling RFCs in the connected ABAP system.
- Backwards compatible to all R/3 versions.
- Incapable to handle:
  - Delta role and profile assignment (to only add or remove roles or profiles that have been changed, without full overwrite of the user).
  - BAPI filters for user exits and Business Suite capabilities (for example to maintain HCM communication information or CRM business partner via Identity Management)

## Repository type *SAPC\_ABAPBusinessSuite*

Using Java Class toSAPidentity which is calling BAPIs in the connected SAP AS ABAP systems. It is required to be chosen if you want to utilize business suite capabilities or provision delta role or profile assignments.

It is required for the connected system to be of version NetWeaver 7.0 EHP 1 (or complementary) at least if this connector is chosen. A full list of required SP levels for the different NetWeaver versions can be found in version note 1469551.

This connector cannot be used connecting a CUA source system to SAP Identity Management.

### 5.2.3 Preparation Steps

Before you can connect an SAP AS ABAP system to SAP Identity Management, you need a service user that is used to establish the connection.

To create the service user, you create a custom copy of the PFCG role *SAP\_BC\_SEC\_IDM\_COMMUNICATION* and rename it to a custom role, such as *Z\_SAP\_BC\_SEC\_IDM\_COMMUNICATION*.

### Procedure

1. In the SAP Easy Access menu, choose the *Create role* pushbutton or access transaction **PFCG**.
2. Enter the name of the delivered standard role: **SAP\_BC\_SEC\_IDM\_COMMUNICATION**.
3. To copy the standard role, choose *Copy role* and enter a new name: **Z\_SAP\_BC\_SEC\_IDM\_COMMUNICATION**.



Do not change the delivered standard roles (SAP\_\*), but rather only the copies of these roles (Z\_\*). Otherwise, the standard roles that you have modified will be overwritten by newly delivered standard roles during a later upgrade or release change.

4. Choose *Change*.
5. On the *Authorizations* tab, choose *Change authorization data*.
6. To display the names of authorization objects, choose: *Utilities* → *Technical Names On*.
7. Choose *Change Authorization Data*.

8. Choose *Manually* and add the authority object `S_TABU_NAM`.
9. Enter the following values:

Field Name	User Action and Values
<i>Activity</i>	03 (Display)
<i>Table names</i>	TUTYPNOW, USR02, USR04, USH02, USH04, AGR_AGRS, AGR_DEFINE, AGR_1252

10. Add `NC` to *Table Authorization Group* of authorization object `S_TABU_DIS`.
11. Navigation path to `S_TABU_DIS` → open node *Basis: Administration* → *Table Maintenance* (via standard tools such as `SM30`)
12. Save and generate the profile for the role.
13. To create the service user, access transaction `SU01` and assign the new custom role `Z_SAP_BC_SEC_IDM_COMMUNICATION`. You should select user type *Service* for this user to exclude it from password change intervals.



If the communication user is supposed to write back Infotype 105 information system user name and mail address (0001 and 0010), please assign a copy of the role `SAP_HR_COMM_IDM_INTEGRATION` to the communication user as well.



If the assigned role does not provide all necessary authorizations, check SAP Note [1691375](#) for additional information and necessary enhancements of the authorizations (which are dependent on the back-end system release).

## 5.2.4 Cleanup of Inconsistencies in the ABAP System

SAP Identity Management is loading user management data from the back-end system. In some cases there are inconsistencies on the SAP AS ABAP system, which need to be corrected before running the initial load job.

If these inconsistencies are not corrected, the initial load job will still run, but error messages will be raised for certain issues. Potential issues are:

- Users are assigned to license types that do not exist anymore in the back-end system
- Users are assigned to roles that do not exist anymore in the back-end system
- Users have the same role assigned several times with overlapping assignment dates

Inconsistencies related to role assignments can be corrected as described here

[http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/8c/ab0b1b96b8492688f8e76114f347a6/content.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/8c/ab0b1b96b8492688f8e76114f347a6/content.htm).

Inconsistencies related to license types can be corrected using transaction `/nUSMM`.

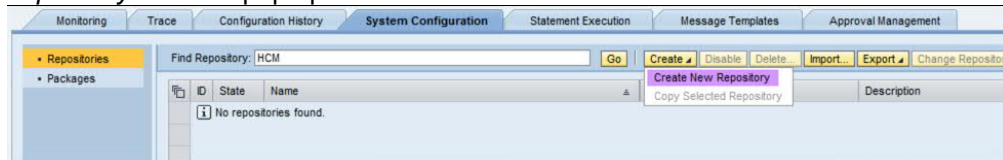
## 5.2.5 Creating the SAP AS ABAP Repository

Before connecting an SAP AS ABAP system to SAP Identity Management, the notes and remarks on the official help page regarding creation of repositories should be considered. This information can be found here:

[http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/9d/a17e94cf624066a5958322c07139a6/content.htm?frameset=/en/e1/312e41e4a34f09a43924c49ca30ea6/frameset.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/9d/a17e94cf624066a5958322c07139a6/content.htm?frameset=/en/e1/312e41e4a34f09a43924c49ca30ea6/frameset.htm)

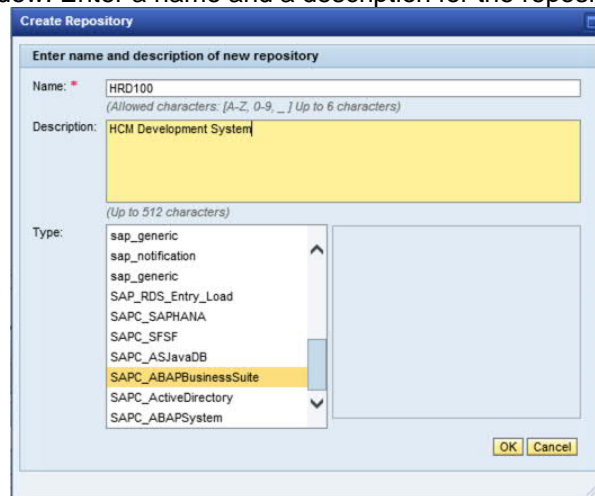
## Procedure

1. In the *Administration and Monitoring* interface for Identity Management ([http\(s\)://<host>:<port>/idm/admin](http(s)://<host>:<port>/idm/admin)) open the tab *System Configuration*.
2. Make sure *Repositories* is selected in the left hand pane of the screen.
3. Click the *Create* button in the second level navigation and chose *Create New Repository* from the pop up menu.



**Figure 10 - Create SAP AS ABAP Repository**

4. Select *SAPC\_ABAPSystem* or *SAPC\_ABAPBusinessSuite* from the selection menu in the popup window. Enter a name and a description for the repository to be created.




**Figure 11 - Define SAP AS ABAP Repository**

5. Click *OK*.
6. Select the repository created by you in the upper table and set the connection information in the table in the bottom of the page. Necessary information that are required to be maintained are described here (sections about repository constants): [http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/fe/47443874bb47a3b313c48cd68ad21c/content.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/fe/47443874bb47a3b313c48cd68ad21c/content.htm).
7. Maintain additional (RDS specific) information as described in the table below:

Name	Current Value	Description
MX_REQ_PRIV_N OMASTER_TASK	SAPC No Master Process	Process that will be called if the account privilege is missing on an identity. The default process will assign the missing account privilege automatically and thus trigger the creation of the user after an assignment has been added. Remove the process if that is not wanted.
SAPC_REP_PROD UCTIVE	<input checked="" type="checkbox"/>	Enable if repository should be active in the system (for instance to be displayed in forms: <i>Reset Password, Enable/Disable Identity</i> )
SAPC_REP_DISPL AYNAME		Display name of the repository, used in user interfaces for end users.
SAPC_LOAD_FILT ER_*		Those constants can be used to filter objects to be read. If for example only Z_* roles should be read a filter value for SAPC_LOAD_FILTER_ROLE would be "where uniqueness like 'Z_%'" Can also be used like blacklist, if for examples test



		<p>users shall not be loaded the filter value for SAPC_LOAD_FILTER_USERS would be "where logonuid not like 'TMP_%' or logonuid not like 'Z_TEST%'"</p>  <p>Depending on your database type and the way you loaded your data into SAP IDM please be aware of lowercase / uppercase during filtering and adapt your filter accordingly.</p>
SAPC_LOAD_SKIP_CHANGES_FROM_BE	true	This flag is used to control whether the update load job is loading changes on user from the connected systems (false). If it is true (default), only roles, profiles and additional user information will be read from the SAP AS ABAP system. If it is false, also changes on attributes and role or profile assignments will be synchronized back into SAP IDM.
SAPC_PROVISION_DELTA	false	<p>Only available for SAP AS ABAP Business Suite connector.</p> <p>This flag is used when provisioning authorization assignments, to determine whether only the change (added removed assignment) will be provisioned (true) or all user assignments will be overwritten on the target system (default behavior).</p>
SAPC_PROVISION_VALIDITY	true	Determine whether or not validity dates on assignments should be provisioned into the target systems.
SAPC_NO_CREATE_ON_MODIFY	false	Determine whether the user creation should fail, if a user already exists. For more information please refer to note <a href="#">2940838</a>
SAPC_RFC_TRACE_ENABLED	false	Enable RFC trace as described in note <a href="#">1642374</a> without the need to check out the package

## 5.2.6 Repository Jobs for SAP AS ABAP

In the following section the repository type jobs are described in the order that they are supposed to be executed. All jobs exist for both SAP AS ABAP repository types (only the name differs whereby "AS ABAP" is replaced by "BusinessSuite" for the Business Suite connector).

- *SAPC AS ABAP - Connection Test*: This job will check the connection to the system to be connected.
- *SAPC AS ABAP - Read Help Values GLOBAL*: This job will load attribute value help information from the system and add them to the global attribute value help information in the database. This could be for example valid values for salutation like *Mr.*
- *SAPC AS ABAP - Read Help Values SYSTEM SPECIFIC*: This job will load attribute value help information from the system and add them to the system specific attribute value help information in the database.
- *SAPC AS ABAP – Initial Load*: This job synchronizes the information between the system to be connected and SAP IDM. Additionally, it will create system specific attributes and objects.
- *SAPC AS ABAP - Read Authorization Details*: This job reads additional information about ABAP authorization objects like the relation between single roles and composite roles.

- **SAPC AS ABAP - Last Logon & Creation:** This job will read the following additional user information:
  - Last logon time of users.
  - Creation date of users.
  - Modify date and last modifying user of users.
  - User lock due to too many failed logon attempts.

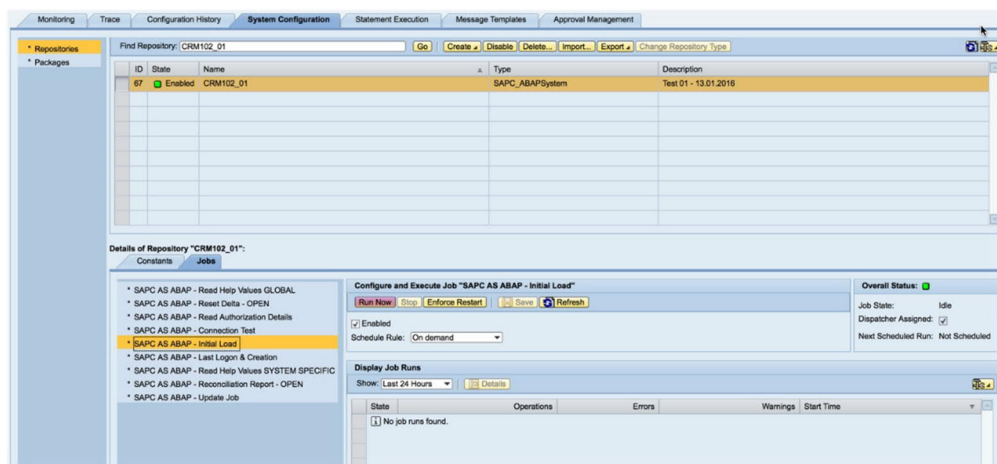


Especially the user lock information are important to be synchronized as they are required to be known to SAP IDM for the user unlock and password reset scenario.

- **SAPC AS ABAP – Update Job:** This job will read updated information from the SAP AS ABAP system to SAP IDM. This can be roles and profiles but also changes on the identities. Using the repository constant `SAPC_LOAD_SKIP_CHANGES_FROM_BE` it can be defined whether changes on the identities should be synchronized back from the connected SAP AS ABAP system to SAP IDM or not. If not only master data like added and deleted privileges and value help data will be synchronized.
- **SAPC AS ABAP – Reconciliation Report:** This job compares the data between SAP IDM and the given backend system and creates an HTML based report showing the differences.
- **SAPC AS ABAP – Reset Delta:** This job resets the SAP IDM internal delta key information for the given backend system. It might be required after job failures occurred (for further information see chapter 5.1.3 - [UpdateLoadJob](#)).

It is recommended to execute the jobs from the administration interface ([http\(s\)://<host>:<port>/idm/admin](http(s)://<host>:<port>/idm/admin)) as described in the following:

1. Select the tab *System Configuration*.
2. Assure the item *Repositories* is selected in the left pane.
3. Select the repository the job is to be executed for from the table in the upper middle section.
4. Select the tab *Jobs* from the lower middle section.
5. Select the job you want to execute and click on the button Run now.
6. Check the job log after the job has been executed. Use the *Details* button to see the messages in the log after selecting it from the table in the lower middle section.



**Figure 12 - Execute Repository Job in Admin UI**





When executing the job *SAPC AS ABAP - Read Authorization Details* it might happen that you receive error messages such as “Referenced value does not exist”. This is typically caused by inconsistent entries in the SAP AS ABAP system. Such inconsistencies have to be manually corrected in the backend system. More details and a description how to fix the issue can be found in SAP note 2108838.

## 5.2.7 Post Load Configuration Steps

### Manual Configuration of System Specific Attributes

The following steps are required to be performed manually:

Changing of attribute *SAPC\_IDEN\_REP\_ENCRYPTED\_PASSWORD\_<repository>*:

- Navigate to your Identity Store → *Identity Store Schema* → *Attributes* and open attribute *SAPC\_IDEN\_REP\_ENCRYPTED\_PASSWORD\_<repository>*.
- Choose the *Presentation* tab.
- Select the *Hide input* checkbox and save the changes.

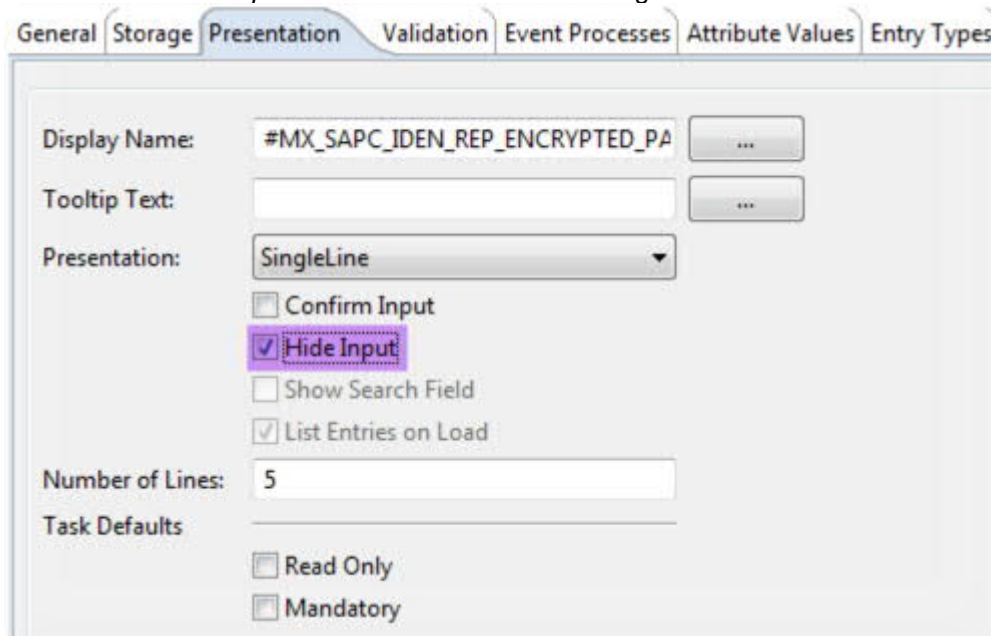
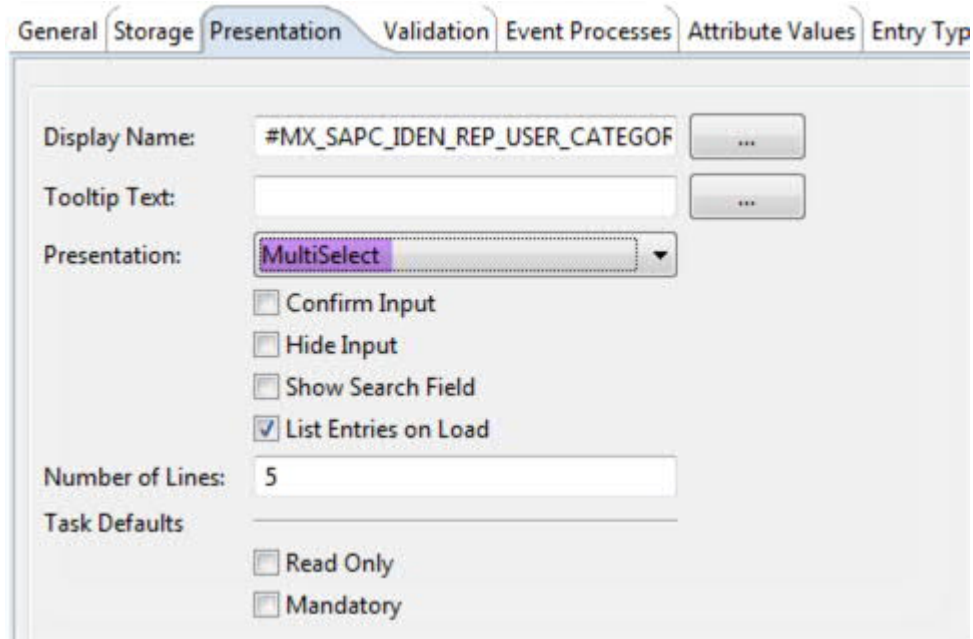


Figure 13 - Hide Encrypted Password

Changing of attribute *SAPC\_IDEN\_REP\_USER\_CATEGORY\_<repository>*:

- Navigate to your Identity Store → *Identity Store Schema* → *Attributes* and open attribute *SAPC\_IDEN\_REP\_USER\_CATEGORY\_<repository>*.
- Choose the *Presentation* tab.
- Change the presentation type to *MultiSelect* and save the changes.



General Storage **Presentation** Validation Event Processes Attribute Values Entry Typ

Display Name: #MX\_SAPC\_IDEN\_REP\_USER\_CATEGOF ...

Tooltip Text: ...

Presentation: MultiSelect

☐ Confirm Input

☐ Hide Input

☐ Show Search Field

☒ List Entries on Load

Number of Lines: 5

Task Defaults

☐ Read Only

☐ Mandatory

**Figure 14 - Set Presentation Type MultiSelect**

The following jobs should be scheduled to run on a frequent basis:

- *SAPC AS ABAP – Update Job*: Depending on what changes are to be synchronized it is recommended to run the job every time there is changes in authorization objects expected (for example after role transport). If changes made to identities on the ABAP system itself need to be synchronized to SAP IDM, it is recommended to run the job more frequent (for example every night).
- *SAPC AS ABAP - Read Authorization Details*: It is recommended to be executed after every run of the update job.
- *SAPC AS ABAP - Last Logon / Creation*: Due to the importance of the synchronization of the lock state and its relatively low system load, it is recommended to run this job also during business hours (for example. every 2 hours).

It is mandatory to schedule those jobs for every repository. Follow the instructions:

7. Open the *IDM Administration and Monitoring Interface* on [http\(s\)://<host>:<port>/idm/admin](http(s)://<host>:<port>/idm/admin).
8. Go to tab *System Configuration*.
9. In the left hand pane select *Repositories* and select the repository to schedule the job for from the upper table.
10. In the lower section of the page select the tabs *Jobs* and chose the job to be scheduled.
11. Select a scheduling rule from the *Schedule Rule* drop down menu that is set to *On Demand* by default.
12. Save the setting and repeat it for other jobs and repositories.

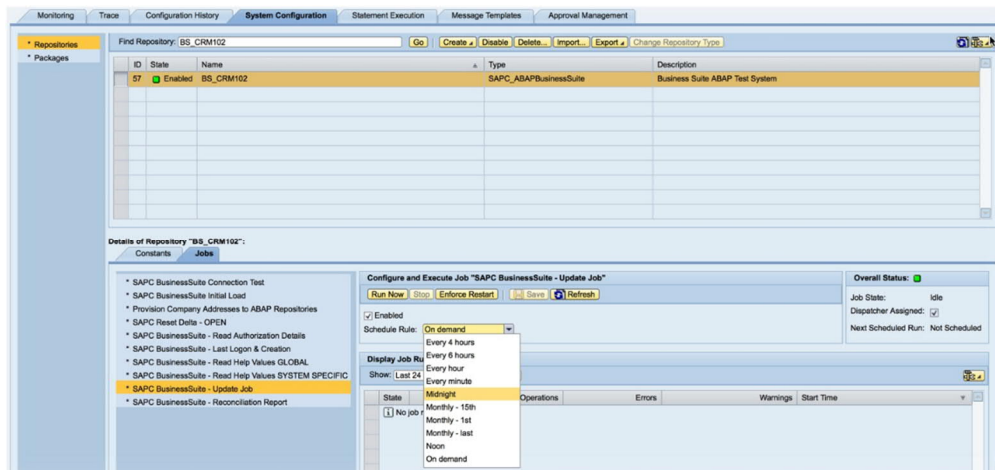


Figure 15 - Scheduling of Repository Job

## 5.3 Connecting an SAP AS Java System

### 5.3.1 General Information

The connectivity to an SAP AS Java system is based on the standard SAP Provisioning Framework containing specific enhancements. For the general usage of the SAP Provisioning Framework, you can refer to the standard documentation that describes the architecture, technical overview, and detailed configuration of SAP Provisioning Framework here:

[http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/f9/0d3de3dc0f47c6bbcc17c4e83f27f7/content.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/f9/0d3de3dc0f47c6bbcc17c4e83f27f7/content.htm)

The subsequent sections in this guide contains detailed information about settings specific to this solution package. When going through these steps, the reader is assumed to be familiar with the standard SAP Provisioning Framework.

Please make yourself familiar with the restrictions that apply to all connectors and to SAP AS Java connector only.

Restrictions that Apply to All Connectors:

[http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/e5/4b384d2cc14813891fb687bbe3d587/content.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/e5/4b384d2cc14813891fb687bbe3d587/content.htm)

Restrictions for SAP AS Java System Connectors:

[http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/e1/8645dd31a44bdd95b0148cab621415/content.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/e1/8645dd31a44bdd95b0148cab621415/content.htm)

### 5.3.2 Preparation Steps

To connect to an AS Java system, you need a technical user that is used to establish the connection. This technical user should only have the necessary authorizations in the back-end system. As of AS Java Release 6.40, the appropriate authorizations are provided with the UME action *UME.SpmI\_Write\_Action*. There is also an action called *UME.SpmI\_Read\_Action* for read-only access. Prior to Release 6.40, the action to use is *MANAGE\_ALL\_COMPANIES*.

### Procedure

Creating the service user:

1. In the UME (User Management Engine) of the AS Java, create a new role. Select *Role* → *Create Role*.
2. Select a name for your new role (for example, *Z\_IDM\_CONNECT*).

3. Go to the *Assigned Actions* tab.
4. Search for **\*spml\***.
5. Select *Spml\_Write\_Action* and *Spml\_Read\_Action*, and choose *Add*.
6. Choose *Save*.
7. In the UME, create a new user. Select *User* → *Create User*.
8. Enter a name, password, and e-mail address.
9. Set the *Security Policy* to *Technical User*.
10. Go to the *Assigned Roles* tab and assign your role that you created in step 2.
11. Choose *Save*.

### 5.3.3 Creating the SAP AS Java Repository

Before connecting an SAP AS Java system to SAP Identity Management, the notes and remarks on the official help page regarding creation of repositories should be considered. This information can be found here:

[http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/9d/a17e94cf624066a5958322c07139a6/content.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/9d/a17e94cf624066a5958322c07139a6/content.htm)

#### Procedure

1. In the *Administration and Monitoring* interface for Identity Management ([http\(s\)://<host>:<port>/idm/admin](http(s)://<host>:<port>/idm/admin)) open the tab *System Configuration*.
2. Make sure *Repositories* is selected in the left hand pane of the screen.
3. Click the *Create* button in the second level navigation and chose *Create New Repository* from the pop up menu.

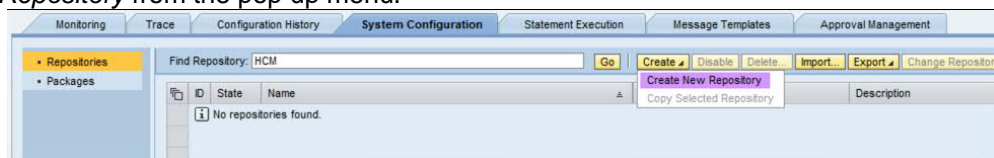


Figure 16 - Create SAP AS Java Repository

4. Select *SAPC\_ASJavaDB* from the selection menu in the popup window. Enter a name and a description for the repository to be created.

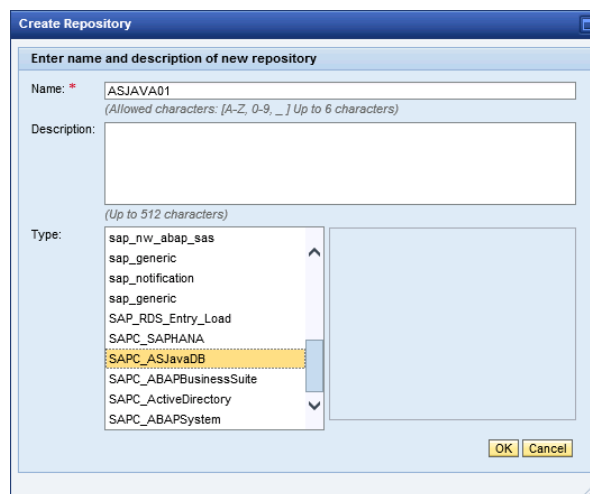



Figure 17 - Define SAP AS Java Repository

5. Click *OK*.

6. Select the repository created by you in the upper table and set the connection information in the table in the bottom of the page. Necessary information that is required to be maintained is described here (sections about repository constants): [http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/f5/b72224836b41d19a70d99adf498cac/content.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/f5/b72224836b41d19a70d99adf498cac/content.htm)
7. Maintain additional (RDS specific) information as described in the table below:

Name	Current Value	Description
MX_REQ_PRIV_NOMASTER_TASK	SAPC No Master Process	Process that will be called if the account privilege is missing on an identity. The default process will assign the missing account privilege automatically and thus trigger the creation of the user after an assignment has been added. Remove the process if that is not wanted.
SAPC_REPOSITORY_ACTIVE	<input checked="" type="checkbox"/>	Enable if repository should be active in the system (for instance to be displayed in forms: <i>Reset Password, Enable/Disable Identity</i> )
SAPC_REPOSITORY_DISPLAYNAME		Display name of the repository, used in user interfaces for end users.
SAPC_LOAD_FILTER_*		<p>Those constants can be used to filter objects to be read. If for example only Z_* roles should be read a filter value for SAPC_LOAD_FILTER_ROLE would be "where uniqueness like 'Z_%'"</p> <p>Can also be used like blacklist, if for examples test users shall not be loaded the filter value for SAPC_LOAD_FILTER_USER would be "where logonuid not like 'TMP_%' or logonuid not like 'Z_TEST%'"</p>  <p>Depending on your database type and the way you loaded your data into SAP IDM please be aware of lowercase / uppercase during filtering and adapt your filter accordingly.</p>
SAPC_LOAD_SKIP_CHANGES_FROM_BE	TRUE	This flag is used to control whether the update load job is loading changes on user from the connected systems (false). If it is true (default), only roles, profiles and additional user information will be read from the SAP AS Java system. If it is false, also changes on attributes and role or profile assignments will be synchronized back into SAP IDM.

### 5.3.4 Repository Jobs for SAP AS Java

In the following section the repository type jobs are described in the order that they are supposed to be executed.

- *SAPC AS Java (Database) - Connection Test*: This job will check the connection to the system to be connected.
- *SAPC AS Java (Database) – Initial Load*: This job synchronizes the information between the system to be connected and SAP IDM. Additionally it will create system specific attributes and objects.
- *SAPC AS Java (Database) – Update Job*: This job will read updated information from the SAP AS Java system to SAP IDM. This can be roles and groups but also changes on the identities. Using the repository constant *SAPC\_LOAD\_SKIP\_CHANGES\_FROM\_BE* it can be defined whether changes on the identities should be synchronized back from the connected SAP AS Java system to SAP IDM.

- *SAPC AS Java (Database) – Reconciliation Report*: This job compares the data in SAP IDM and the given backend system and creates an HTML based report showing the differences.
- *SAPC AS Java (Database) – Reset Delta*: This job resets the SAP IDM internal delta key information for the given backend system. It might be required after job failures occurred (for further information see chapter 5.1.3 - [UpdateLoadJob](#)).

It is recommended to execute the jobs from the administration interface ([http\(s\)://<host>:<port>/idm/admin](http(s)://<host>:<port>/idm/admin)) as described in the following:

1. Select the tab *System Configuration*.
2. Assure the item *Repositories* is selected in the left pane.
3. Select the repository the job is to be executed for from the table in the upper middle section.
4. Select the tab *Jobs* from the lower middle section.
5. Select the job you want to execute and click on the button *Run now*.
6. Check the job log after the job has been executed. Use the *Details* button to see the messages in the log after selecting it from the table in the lower middle section.

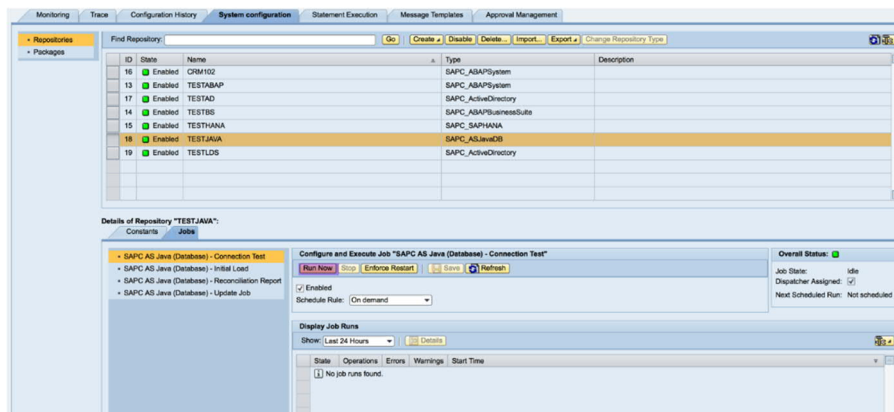


Figure 18 - Execute Repository Job in Admin UI

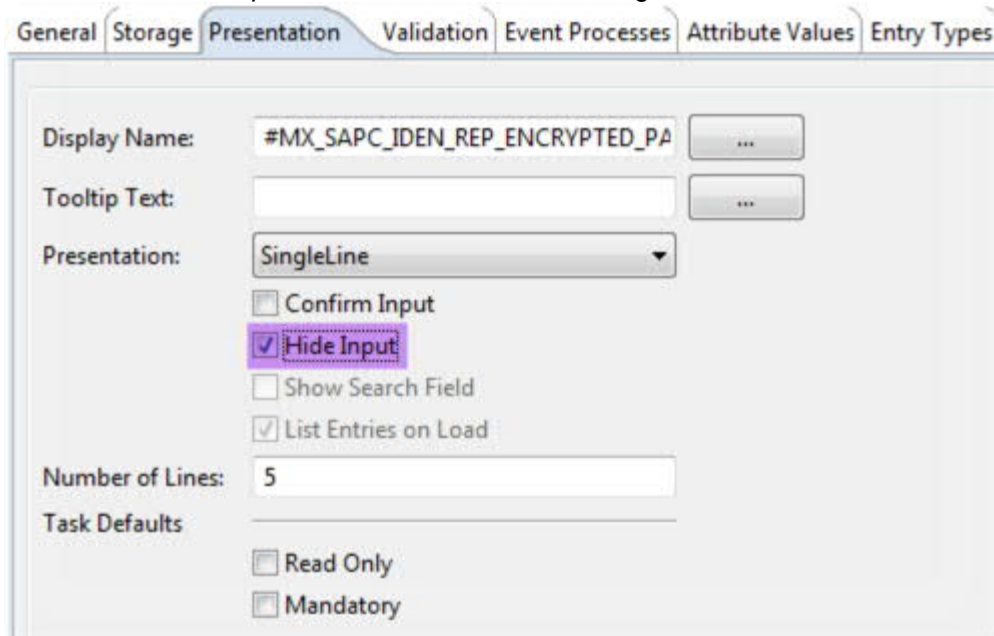
### 5.3.5 Post Load Configuration Steps

#### Manual Configuration of System Specific Attributes

The following steps are required to be performed manually:

Changing of attribute `SAPC_IDEN_REP_ENCRYPTED_PASSWORD_<repository>`:

- Navigate to your Identity Store → *Identity Store Schema* → *Attributes* and open attribute `SAPC_IDEN_REP_ENCRYPTED_PASSWORD_<repository>`.
- Choose the *Presentation* tab.
- Select the *Hide input* checkbox and save the changes.



General Storage **Presentation** Validation Event Processes Attribute Values Entry Types

Display Name: #MX\_SAPC\_IDEN\_REP\_ENCRYPTED\_PA ...

Tooltip Text: ...

Presentation: SingleLine

☐ Confirm Input

☒ **Hide Input**

☐ Show Search Field

☒ List Entries on Load

Number of Lines: 5

Task Defaults

☐ Read Only

☐ Mandatory

**Figure 19 - Hide Encrypted Password**

The following jobs should be scheduled to run on a frequent basis:

- *SAPC AS Java (Database) – Update Job*

It is mandatory to schedule those jobs for every repository. Follow the instructions:

1. Open the *IDM Administration and Monitoring Interface* on `http(s)://<host>:<port>/idm/admin`.
2. Go to tab *System Configuration*.
3. In the left hand pane select *Repositories* and select the repository to schedule the job for from the upper table.
4. In the lower section of the page select the tabs *Jobs* and chose the job to be scheduled.
5. Select a scheduling rule from the *Schedule Rule* drop down menu that is set to *On Demand* by default.
6. Save the setting and repeat it for other jobs and repositories.



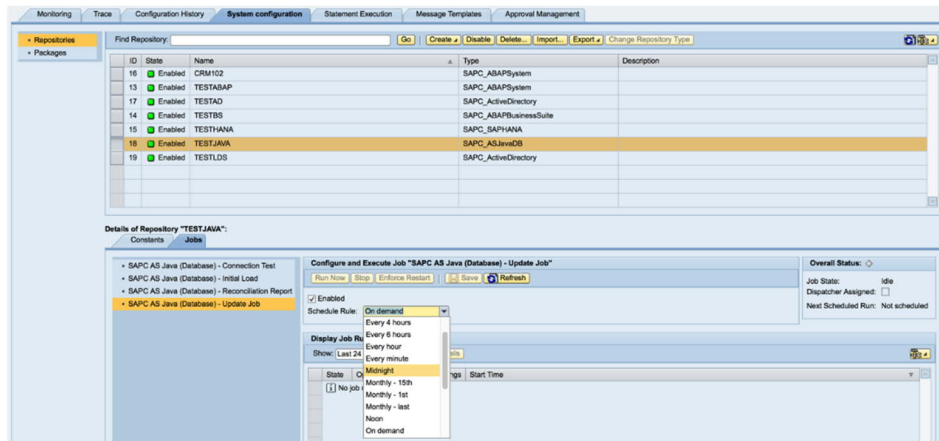


Figure 20 - Scheduling of Repository Job

## 5.4 Connecting an SAP HANA System

### 5.4.1 General Information

The connectivity to an SAP HANA system is based on the standard SAP Provisioning Framework containing specific enhancements. For the general usage of the SAP Provisioning Framework, you can refer to the standard documentation that describes the architecture, technical overview, and detailed configuration of SAP Provisioning Framework here:

[http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/93/831fcc0ff044c78a0ad718ac9472be/content.htm?frameset=/en/f9/0d3de3dc0f47c6bbcc17c4e83f27f7/frameset.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/93/831fcc0ff044c78a0ad718ac9472be/content.htm?frameset=/en/f9/0d3de3dc0f47c6bbcc17c4e83f27f7/frameset.htm)

The subsequent sections in this guide contain detailed information about settings specific to this solution package. When going through these steps, the reader is assumed to be familiar with the standard SAP Provisioning Framework.

Please make yourself familiar with the restrictions that apply to all connectors and to SAP HANA connector only.

Restrictions that Apply to All Connectors:

[http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/e5/4b384d2cc14813891fb687bbe3d587/content.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/e5/4b384d2cc14813891fb687bbe3d587/content.htm)

Restrictions for SAP HANA System Connectors:

[http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/d1/ce4aa1928d4942b901dc2ba27261a7/content.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/d1/ce4aa1928d4942b901dc2ba27261a7/content.htm)

### 5.4.2 Preparation Steps

To be able to connect to the SAP HANA system a specific database driver is needed to be installed on the machine(s) where the SAP IDM dispatcher(s) is/are running.

#### Procedure

Adding JDBC Driver in Identity Management Dispatcher Classpath:

1. Download the latest available version of *ngdbc.jar* file for HANA JDBC driver. It is delivered as part of the SAP HANA client at SAP Service Marketplace.
2. Copy it to the system where the dispatcher(s) is/are running, for example in the Identity Center Java subfolder.



3. Navigate to the folder "...\\Identity Center\\Service-Scripts" and open the `<dispatcher_name>.prop` file(s) for all dispatchers.
4. In the section `DSECLASSPATH` add the path to HANA JDBC driver `ngdbc.jar` file. For example:  

```
DSECLASSPATH=%DSE_HOME%/Java/DSE.jar;%DSE_HOME%/Java;%DSE_HOME%/Java/sapjco.jar;%DSE_HOME%/Java/sapjco3.jar;%DSE_HOME%/Java/ngdbc.jar
```
5. Restart your dispatcher(s).

To connect to an SAP HANA system, you need a technical user that is used to establish the connection. This technical user should only have the necessary authorizations in the back-end system as required for your use case.



It is recommended to use HANA roles for provisioning of user assignments instead of assigning HANA privileges directly.



*HANA Catalog Roles* are also called *HANA Runtime Roles* in the SAP IDM documentation. *HANA Repository Roles* are also called *HANA Designtime Roles* in the SAP IDM documentation. The main difference for provisioning of user assignments between those objects in SAP IDM is the value of attribute `MX_HANA_ROLE_TYPE` (taken from `CREATOR` column in SAP HANA):

- **\_SYS\_REPO:** HANA Designtime Role; grantor during provisioning of the assignment done by the SAP IDM connector is `_SYS_REPO`
- **Any other value:** HANA Catalog Role; grantor during provisioning of the assignment is the connection user entered in the SAP HANA repository in SAP IDM



If you need to provision assignments to HANA System Privileges you have to enhance the permissions of the technical connection user. This user needs to be assigned to each HANA System Privilege that should be used and must have the flag "*Grantable to other users and roles*" enabled.

More details can be found in this chapter of the official SAP IDM 8.0 documentation:

[http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/e7/5973d32b174ed7a2107dd5c680d1fb/content.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/e7/5973d32b174ed7a2107dd5c680d1fb/content.htm)

## Procedure

Creating the service user:

1. Open the SAP HANA Studio with a user who has the privileges for the SQL command below.
2. Run the following SQL commands (example):

```
CREATE USER <name of technical user> PASSWORD <password>;
ALTER USER <name of technical user> DISABLE PASSWORD LIFETIME;
```

```

GRANT MONITORING TO <name of technical user>;
GRANT ROLE ADMIN TO <name of technical user>;
GRANT USER ADMIN TO <name of technical user>;
GRANT EXECUTE ON GRANT_ACTIVATED_ROLE TO <name of technical user>;
GRANT EXECUTE ON REVOKE_ACTIVATED_ROLE TO <name of technical user>;
GRANT EXECUTE ON GRANT_APPLICATION_PRIVILEGE TO <name of technical user>;
GRANT EXECUTE ON REVOKE_APPLICATION_PRIVILEGE TO <name of technical user>;

```

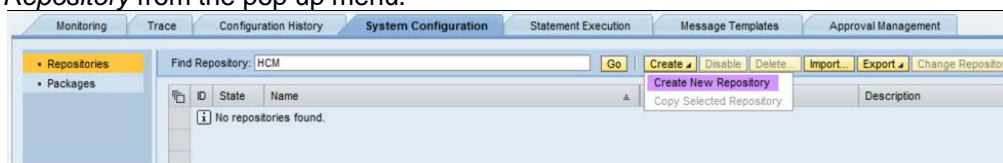
### 5.4.3 Creating the SAP HANA Repository

Before connecting an SAP HANA system to SAP Identity Management, the notes and remarks on the official help page regarding creation of repositories should be considered. This information can be found here:

[http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/9d/a17e94cf624066a5958322c07139a6/content.htm?frameset=/en/e1/312e41e4a34f09a43924c49ca30ea6/frameset.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/9d/a17e94cf624066a5958322c07139a6/content.htm?frameset=/en/e1/312e41e4a34f09a43924c49ca30ea6/frameset.htm)

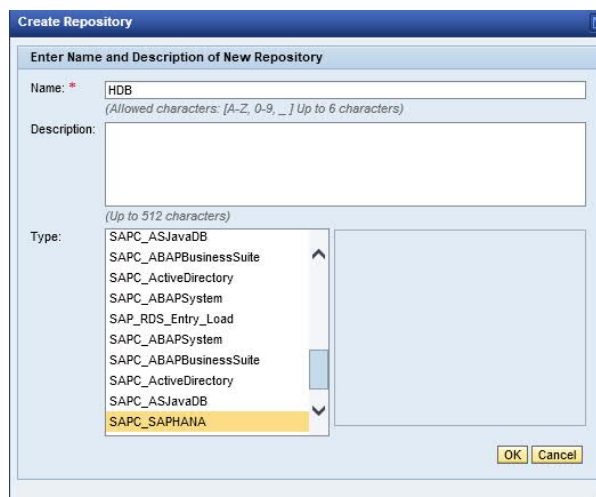
#### Procedure

1. In the *Administration and Monitoring* interface for Identity Management ([http\(s\)://<host>:<port>/idm/admin](http(s)://<host>:<port>/idm/admin)) open the tab *System Configuration*.
2. Make sure *Repositories* is selected in the left hand pane of the screen.
3. Click the *Create* button in the second level navigation and chose *Create New Repository* from the pop up menu.



**Figure 21 - Create SAP HANA Repository**

4. Select *SAPC\_SAPHANA* from the selection menu in the popup window. Enter a name (length limitation of 6 characters when using Oracle as database) and a description for the repository to be created.



**Figure 22 - Define SAP HANA Repository**

5. Click *OK*.
6. Select the repository created by you in the upper table and set the connection information in the table in the bottom of the page. Necessary information that are required to be maintained are described here (sections about repository constants): [http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/04/7f9e4916174a25b59e694d00555760/content.htm?frameset=/en/f5/b72224836b41d19a70d99adf498cac/frameset.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/04/7f9e4916174a25b59e694d00555760/content.htm?frameset=/en/f5/b72224836b41d19a70d99adf498cac/frameset.htm)



Please make sure to enter the user name (in SAP IDM configuration) to be used for the connection to the SAP HANA system in the same syntax (such as lower and uppercase) as created in the SAP HANA system.

7. Maintain additional (RDS specific) information as described in the table below:

Name	Current Value	Description
MX_REQ_PRIV_NOM ASTER_TASK	SAPC No Master Process	Process that will be called if the account privilege is missing on an identity. The default process will assign the missing account privilege automatically and thus trigger the creation of the user after an assignment has been added. Remove the process if that is not wanted.
SAPC_REP_PRODUC TIVE	<input checked="" type="checkbox"/>	Enable if repository should be active in the system (for instance to be displayed in forms: <i>Reset Password, Enable/Disable Identity</i> )
SAPC_REP_DISPLAY NAME		Display name of the repository, used in user interfaces for end users.
SAPC_LOAD_FILTER _*		Those constants can be used to filter objects to be read. If for example only Z_* SAP HANA roles should be read a filter value for SAPC_LOAD_FILTER_HANAROLE would be "where ROLE_NAME like 'Z_%'" Can also be used like blacklist, if for examples test users shall not be loaded the filter value for SAPC_LOAD_FILTER_USERS would be "where USER_NAME not like 'TMP_%' or USER_NAME not like 'Z_TEST%'" It is also possible to filter SAP HANA application and system privileges by using SAPC_LOAD_FILTER_APPPRIV or SAPC_LOAD_FILTER_SYSTEMPRIV.
SAPC_LOAD_SKIP_C HANGES_FROM_BE	<input checked="" type="checkbox"/>	This flag is used to control whether the update load job is loading changes on user from the connected systems (false). If it is true (default), only roles, profiles and additional user information will be read from the SAP AS Java system. If it is false, also changes on attributes and role or profile assignments will by synchronized back into SAP IDM.

#### 5.4.4 Repository Jobs for SAP HANA

In the following section the repository type jobs are described in the order that they are supposed to be executed.

- *SAPC HANA - Connection Test*: This job will check the connection to the system to be connected.
- *SAPC HANA – Initial Load*: This job synchronizes the information between the system to be connected and SAP IDM. Additionally it will create system specific attributes and objects.
- *SAPC HANA – Update Job*: This job will read updated information from the SAP HANA system to SAP IDM. This can be roles, system privileges and application privileges but also changes on the identities. Using the repository constant *SAPC\_LOAD\_SKIP\_CHANGES\_FROM\_BE* it can be defined whether changes on the identities should be synchronized back from the connected SAP HANA system to SAP IDM.

- *SAPC HANA – Reconciliation Report*: This job compares the data in SAP IDM and the given backend system and creates an HTML based report showing the differences.
- *SAPC HANA – Reset Delta*: This job resets the SAP IDM internal delta key information for the given backend system.



The handling of the filters by using the repository constants `SAPC_LOAD_FILTER_*` in jobs *SAPC HANA – Initial Load* and *SAPC HANA – Update Job* is different to the same jobs for SAP AS ABAP, SAP AS JAVA and Microsoft ADS. The HANA jobs consider the filters already during the reading of all data from the backend system (as part of the SOURCE SQL statements of the jobs).



Please be aware of the fact that certain HANA user parameters are treated as special user parameters and synchronized with the respective SAP IDM attributes directly (such as E-Mail being synchronized with `MX_MAIL_PRIMARY`). All other HANA user parameters are synchronized to the global and / or system specific attributes for HANA user parameters (attributes `MX_USER_PARAMS` and `SAPC_IDEN_REP_HANA_PARAMETERS_<repname>`).

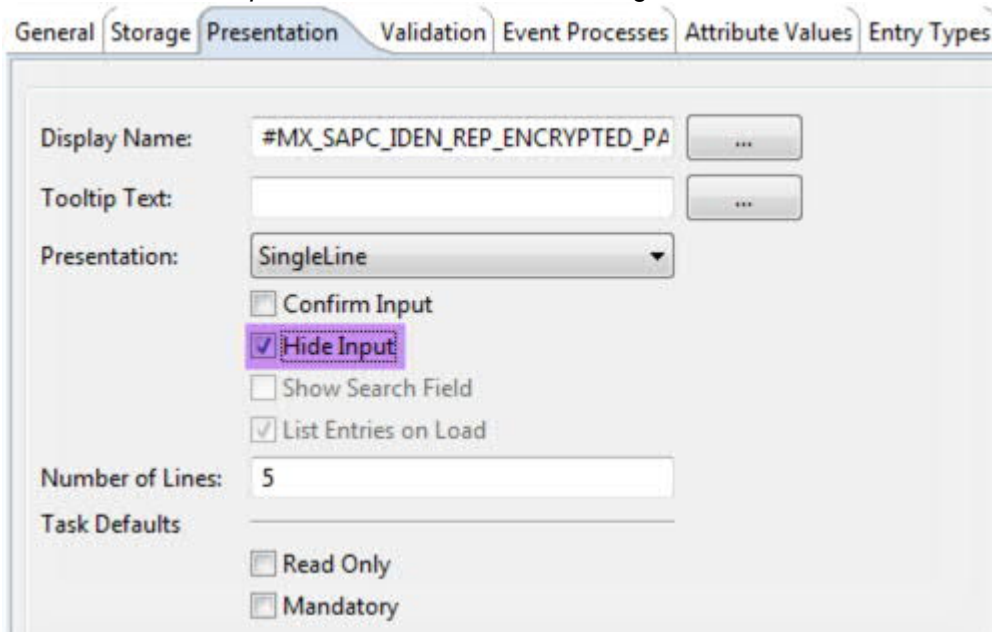
## 5.4.5 Post Load Configuration Steps

### Manual Configuration of System Specific Attributes

The following steps are required to be performed manually:

Changing of attribute `SAPC_IDEN_REP_ENCRYPTED_PASSWORD_<repository>`:

- Navigate to your Identity Store → *Identity Store Schema* → *Attributes* and open attribute `SAPC_IDEN_REP_ENCRYPTED_PASSWORD_<repository>`.
- Choose the *Presentation* tab.
- Select the *Hide input* checkbox and save the changes.



The screenshot shows the configuration window for the attribute `SAPC_IDEN_REP_ENCRYPTED_PASSWORD_<repository>`. The **Presentation** tab is selected. The **Display Name** is `#MX_SAPC_IDEN_REP_ENCRYPTED_PA`. The **Presentation** dropdown is set to **SingleLine**. The **Hide Input** checkbox is checked and highlighted with a red box. Other options include **Confirm Input**, **Show Search Field**, **List Entries on Load**, **Number of Lines** (set to 5), and **Task Defaults** (Read Only and Mandatory checkboxes).

Figure 23 - Hide Encrypted Password

The following jobs should be scheduled to run on a frequent basis:

- *SAPC HANA– Update Job*: Depending on what changes are to be synchronized it is recommended to run the job every time there is changes in authorization objects expected (for example after role transport). If changes made to identities on the HANA system itself need to be synchronized to SAP IDM, it is recommended to run the job more frequent (for example every night).

It is mandatory to schedule those jobs for every repository. Follow the instructions:

1. Open the *IDM Administration and Monitoring Interface* on `http(s)://<host>:<port>/idm/admin`.
2. Go to tab *System Configuration*.
3. In the left hand pane select *Repositories* and select the repository to schedule the job for from the upper table.
4. In the lower section of the page select the tabs *Jobs* and chose the job to be scheduled.
5. Select a scheduling rule from the *Schedule Rule* drop down menu that is set to *On Demand* by default.
6. Save the setting and repeat it for other jobs and repositories.

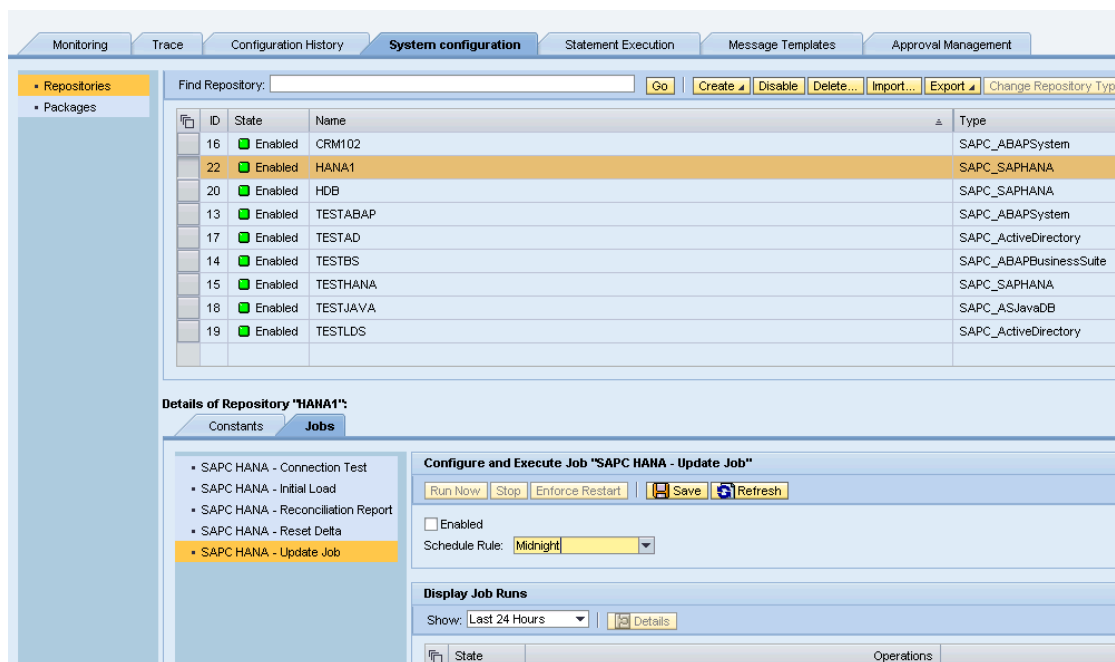


Figure 24 - Scheduling of Repository Job

## 5.5 Connecting an SAP SuccessFactors System

### 5.5.1 General Information

The connectivity to a SuccessFactors system is based on the standard SAP Provisioning Framework containing specific enhancements. For the general usage of the SAP Provisioning Framework, you can refer to the standard documentation that describes the architecture, technical overview, and detailed configuration of SAP Provisioning Framework here:

[http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/bc/7c98740824425494df38ec8a428e97/content.htm?frameset=/en/93/831fcc0ff044c78a0ad718ac9472be/frameset.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/bc/7c98740824425494df38ec8a428e97/content.htm?frameset=/en/93/831fcc0ff044c78a0ad718ac9472be/frameset.htm)

The subsequent sections in this guide contain detailed information about settings specific to this solution package. When going through these steps, the reader is assumed to be familiar with the standard SAP Provisioning Framework.

Please make yourself familiar with the restrictions that apply to all connectors and to SuccessFactors connector only.

Restrictions that Apply to All Connectors:

[http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/e5/4b384d2cc14813891fb687bbe3d587/content.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/e5/4b384d2cc14813891fb687bbe3d587/content.htm)

Restrictions for SuccessFactors System Connectors:

[http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/e4/0b642f525947c79bd5c3bb3b40750b/content.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/e4/0b642f525947c79bd5c3bb3b40750b/content.htm)

The whole configuration for the SuccessFactors connector is based in the standard SAP IDM functionality and documentation. Therefore no additional documentation is provided within this document.

## 5.6 Connecting Microsoft Active Directory / LDS

### 5.6.1 General Information

The connectivity to the Microsoft Active Directory / LDS system is based on the standard SAP Provisioning Framework containing specific enhancements. For the general usage of the SAP Provisioning Framework, you can refer to the standard documentation that describes the architecture, technical overview, and detailed configuration of SAP Provisioning Framework here:

[http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/84/40dc9e770340ca870b5a0986ecb25b/content.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/84/40dc9e770340ca870b5a0986ecb25b/content.htm)

The subsequent sections in this guide contain detailed information about settings specific to this solution package. When going through these steps, the reader is assumed to be familiar with the standard SAP Provisioning Framework.

Please make yourself familiar with the restrictions that apply to all connectors and to LDAP connector only.

Restrictions that Apply to All Connectors:

[http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/e5/4b384d2cc14813891fb687bbe3d587/content.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/e5/4b384d2cc14813891fb687bbe3d587/content.htm)

Restrictions for LDAP Connectors:

[http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/d2/9616b7ce014d249aff3d99f8edfedf/content.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/d2/9616b7ce014d249aff3d99f8edfedf/content.htm)

Usage of GUID to provision data:

- The SAP IDM standard product is using the dn (distinguishedName) of a given object (user or group) to uniquely identify it in the Active Directory or LDS
  - This has one major disadvantage: if the object is moved within the Active Directory or LDS it cannot be identified and provisioned by SAP IDM anymore
- The IDM RDS package connector for Active Directory and LDS supports the usage of the GUID (which remains the same for the whole lifetime of an object)
- Example:
  - dn: cn=IDM\_TEST02,OU=Users,DC=sectest,DC=mycompany,DC=com
  - GUID: 91390352-3e4b-4d2c-a263-743484e60709
    - For provisioning based on the LDAP protocol the GUID has to be put within brackets (<91390352-3e4b-4d2c-a263-743484e60709>)



## 5.6.2 Preparation Steps

Before you can connect a Microsoft Active Directory / LDS system to SAP Identity Management, you need a service user with administrative access to the Active Directory / LDS that is used to establish the connection.

## 5.6.3 Creating the AD / LDS Repository

Before connecting a Microsoft Active Directory / LDS system to SAP Identity Management, the notes and remarks on the official help page regarding creation of repositories should be considered. This information can be found here:

[http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/9d/a17e94cf624066a5958322c07139a6/content.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/9d/a17e94cf624066a5958322c07139a6/content.htm)

## Procedure

1. In the *Administration and Monitoring* interface for Identity Management ([http\(s\)://<host>:<port>/idm/admin](http(s)://<host>:<port>/idm/admin)) open the tab *System Configuration*.
2. Make sure *Repositories* is selected in the left hand pane of the screen.
3. Click the *Create* button in the second level navigation and chose *Create New Repository* from the pop up menu.

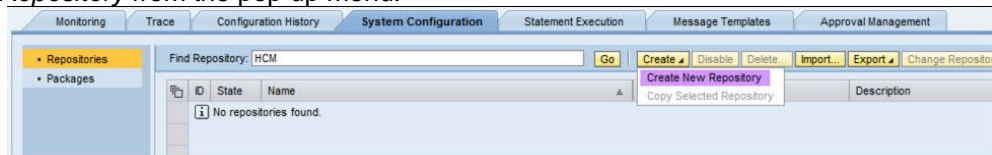


Figure 25 - Create Microsoft Active Directory Repository

4. Select *SAPC\_ActiveDirectory* from the selection menu in the popup window. Enter a name and a description for the repository to be created.

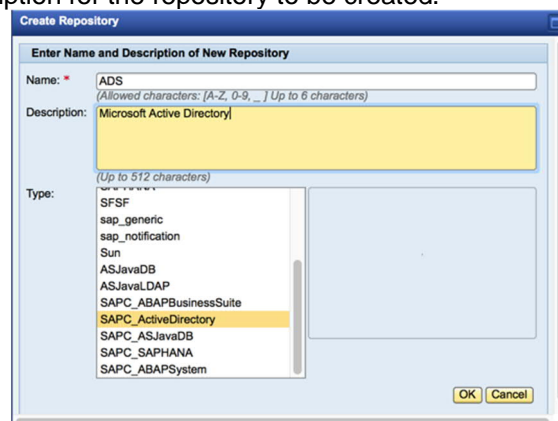



Figure 26 - Define Microsoft Active Directory Repository

5. Click *OK*.
6. Select the repository created by you in the upper table and set the connection information in the table in the bottom of the page. Necessary information that is required to be maintained is described here (sections about repository constants): [http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/d8/032660e11849978a2ba2675b5dd719/content.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/d8/032660e11849978a2ba2675b5dd719/content.htm).
7. Maintain additional (RDS specific) information as described in the table below:

Name	Current Value	Description
MX_REQ_PRIV_N OMASTER_TASK	SAPC No Master Process	Process that will be called if the account privilege is missing on an identity. The default process will assign the missing account privilege automatically and thus trigger the creation of the user after an



		assignment has been added. Remove the process if that is not wanted.
SAPC_REP_PRODUCTIVE	<input checked="" type="checkbox"/>	Enable if repository should be active in the system (for instance to be displayed in forms: <i>Reset Password, Enable/Disable Identity</i> )
SAPC_REP_DISPLAYNAME		Display name of the repository, used in user interfaces for end users.
SAPC_LOAD_FILTER_*		<p>Those constants can be used to filter objects to be read. If for example only Z* groups should be read a filter value for SAPC_LOAD_FILTER_GROUP would be "where cn like 'Z%'"</p> <p>Can also be used like blacklist, if for examples TMP* users shall not be loaded the filter value for SAPC_LOAD_FILTER_USER would be "where userid not like 'TMP%'"</p>  <p>Depending on your database type and the way you loaded your data into SAP IDM please be aware of lowercase / uppercase during filtering and adapt your filter accordingly.</p>
SAPC_LOAD_SKIP_CHANGES_FROM_BE	TRUE	This flag is used to control whether the update load job is loading changes on user from the connected systems (false). If it is true (default), only roles, profiles and additional user information will be read from the SAP AS ABAP system. If it is false, also changes on attributes and role or profile assignments will be synchronized back into SAP IDM.
SAPC_CN_NAMING_CONVENTION	%MX_FIRSTNAME% %MX_LASTNAME%	This constant determines the naming convention of the user CN. Possible values could be: 1. %MSKEYVALUE%; 2. %MX_FIRSTNAME% %MX_LASTNAME%; 3. %MX_LASTNAME%, %MX_FIRSTNAME%.
SAPC_REPOSITORY_SUB_TYPE	AD	Determine the repository sub-type. AD for Microsoft Active Directory, LDS for Microsoft Lightweight Directory Services.

### 5.6.4 Repository Jobs for AD / LDS

In the following section the repository type jobs are described in the order that they are supposed to be executed.

- *SAPC LDAP (ADS) – Connection Test*: This job will check the connection to the system to be connected.
- *SAPC LDAP (ADS) – Initial Load*: This job synchronizes the information between the system to be connected and SAP IDM. Additionally, it will create system specific attributes and objects. It is using the repository constant *SAPC\_REPOSITORY\_SUB\_TYPE* to distinguish between Microsoft Active Directory (AD) and Lightweight Directory Services (LDS). Default value is Microsoft Active Directory (AD).
- *SAPC LDAP (ADS) – Update Job*: This job will read updated information from the Microsoft Active Directory / LDS system to SAP IDM. This can be groups but also changes on the identities. Using the repository constant *SAPC\_LOAD\_SKIP\_CHANGES\_FROM\_BE* it can be defined whether changes on the identities should be synchronized back from the connected system to SAP IDM or not.

- **SAPC LDAP (ADS) – Reconciliation Report:** This job compares the data between SAP IDM and the given backend system and creates an HTML based report showing the differences.
- **SAPC LDAP (ADS) – Reset Delta:** This job resets the SAP IDM internal delta key information for the given backend system. It might be required after job failures occurred (for further information see chapter 5.1.3 - [UpdateLoadJob](#)).

It is recommended to execute the jobs from the administration interface ([http\(s\)://<host>:<port>/idm/admin](http(s)://<host>:<port>/idm/admin)) as described in the following:

1. Select the tab *System Configuration*.
2. Assure the item *Repositories* is selected in the left pane.
3. Select the repository the job is to be executed for from the table in the upper middle section.
4. Select the tab *Jobs* from the lower middle section.
5. Select the job you want to execute and click on the button Run now.
6. Check the job log after the job has been executed. Use the *Details* button to see the messages in the log after selecting it from the table in the lower middle section.

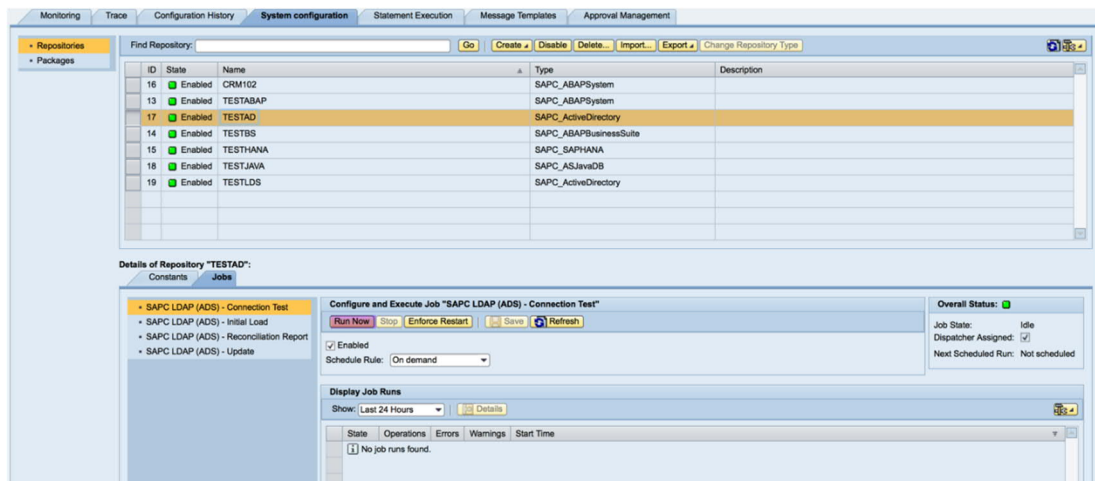


Figure 27 - Execute Repository Job in Admin UI

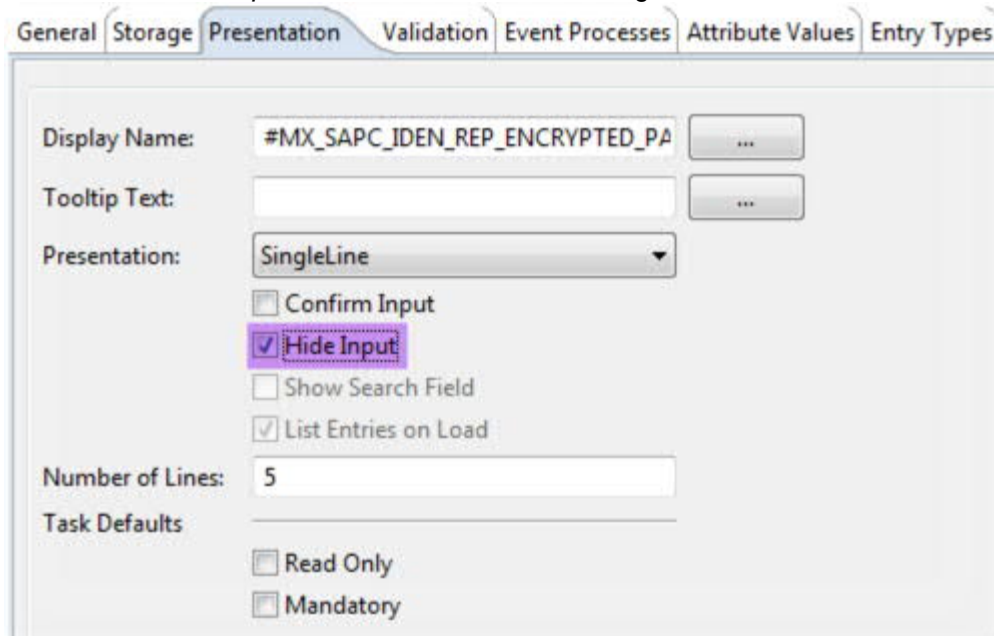
## 5.6.5 Post Load Configuration Steps

### Manual Configuration of System Specific Attributes

The following steps are required to be performed manually:

Changing of attribute `SAPC_IDEN_REP_ENCRYPTED_PASSWORD_<repository>`:

- Navigate to your Identity Store → *Identity Store Schema* → *Attributes* and open attribute `SAPC_IDEN_REP_ENCRYPTED_PASSWORD_<repository>`.
- Choose the *Presentation* tab.
- Select the *Hide input* checkbox and save the changes.



The screenshot shows the configuration interface for an attribute in the SAP Identity Store. The 'Presentation' tab is selected. The attribute name is '#MX\_SAPC\_IDEN\_REP\_ENCRYPTED\_PA'. The 'Presentation' dropdown is set to 'SingleLine'. The 'Hide Input' checkbox is checked and highlighted with a red box. Other options include 'Confirm Input', 'Show Search Field', 'List Entries on Load', 'Number of Lines' (set to 5), and 'Task Defaults' (Read Only and Mandatory checkboxes).

**Figure 28 - Hide Encrypted Password**

The following jobs should be scheduled to run on a frequent basis:

- *SAPC LDAP (ADS) – Update Job*: Depending on what changes are to be synchronized it is recommended to run the job every time there is changes to groups. If changes made to identities on the LDAP system itself need to be synchronized to SAP IDM, it is recommended to run the job more frequent (for example every night).

It is mandatory to schedule those jobs for every repository. Follow the instructions:

1. Open the *IDM Administration and Monitoring Interface* on `http(s)://<host>:<port>/idm/admin`.
2. Go to tab *System Configuration*.
3. In the left hand pane select *Repositories* and select the repository to schedule the job for from the upper table.
4. In the lower section of the page select the tabs *Jobs* and chose the job to be scheduled.
5. Select a scheduling rule from the *Schedule Rule* drop down menu that is set to *On Demand* by default.
6. Save the setting and repeat it for other jobs and repositories.

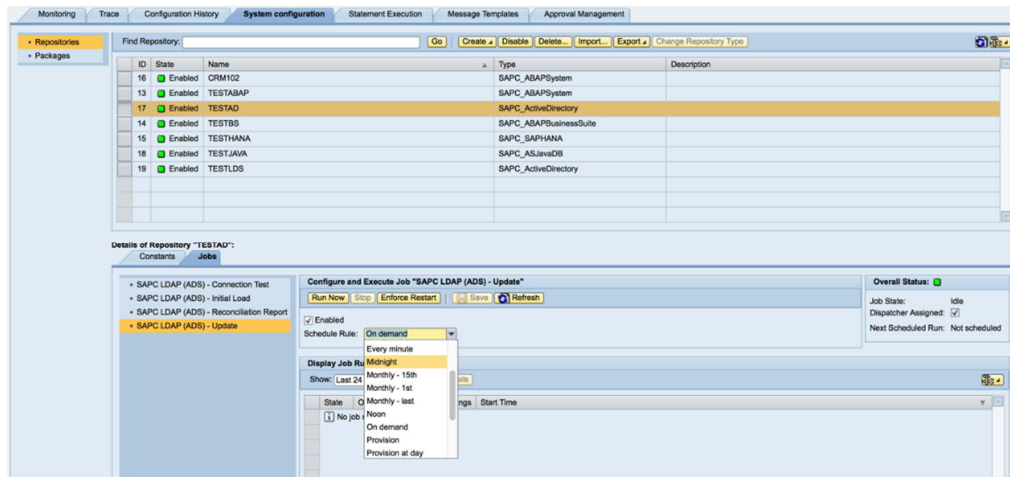


Figure 29 - Scheduling of Repository Job

## 5.7 Post System Connection Steps

### 5.7.1 UI Forms for System Specific Attributes

System specific attributes are created by the initial load jobs for the different repository types. Thus, system specific attributes only exist after a system got connected to IdM and the Initial Load Job has been executed successfully. In order to make system specific attributes visible and allow modifications to those that can be changed, a manual configuration of the according UI forms is required. The RDS configuration package *com.sap.rds.idm.forms.systemspecific* contains templates for those forms that should be used.



It is necessary to check out that package in order to modify the configuration of it. Thus it is recommended to import the package *com.sap.rds.idm.forms.systemspecific* as a new package in the folder where the packages owned by the customer are stored.

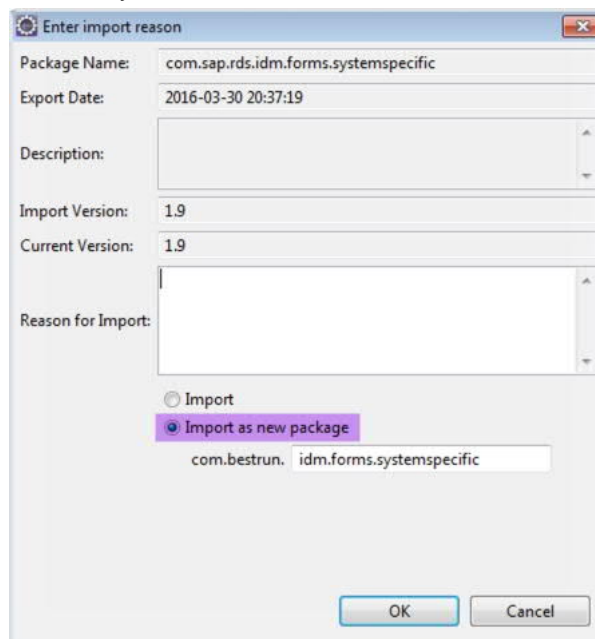


Figure 30 - Import as new package



In most cases the connected systems to an SAP IDM system vary between the different environments; the same systems should usually not be connected to multiple SAP IDM systems. This means that the system specific attributes are different between the environments. Therefore it is usually not useful to transport the system specific forms but required to maintain them on every environment directly as a customer package (by importing the template on every SAP IDM system directly).

The next configuration steps are necessary to enable the system-specific Web enabled tasks for the new repository. These configurations steps must be executed for each repository.

1. Open the form *SAPC Display Identity - System Specific Data* of the imported package *com.sap.rds.idm.forms.systemspecific* (the name might be different in the system if the package was imported as a new custom package as recommended above).
2. Add the system specific attributes to the UI form as shown in the example screenshot below.
3. Move the attributes up towards the section for system specific attributes of the attribute group, as shown in the example screenshot below and tag the *Read Only* flag.



A value help function of Eclipse Studio allows to find the right attributes. After typing *sapc\_iden\_rep\_* the keyboard combination Ctrl+Space will provide a list of attributes to use.

SAPC Display Identity - System Specific Data ⓘ

General Result Handling **Attributes** Access Control Presentation Documentation

Entry Type: 7/MX\_PERSON

Form Attributes:

Attribute	List	Mandatory	Read Only	Default Value/Caption
--Tab--				#MX_SAPC_TAB_GENERAL_DN
--Section--				
--Column--				#MX_SAPC_COLUMN_HEADER_DATA_DN
MSKEYVALUE	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
DISPLAYNAME	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
MX_SALUTATION	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
MX_FIRSTNAME	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
MX_LASTNAME	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
--Column--				#MX_SAPC_COLUMN_AUDIT_DATA_DN
SAPC_IDEN_CREATED_BY	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
SAPC_IDEN_LAST_CHANGED_BY	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
SAPC_IDEN_LAST_CHANGED_AT	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
--Section--				
--Line--				
SAPC_TEMP_REQUESTED_BY	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	%ADMINMSKEY%
--Tab--				#MX_SAPC_TAB_VALIDITY_DN
--Section--				
--Column--				#MX_SAPC_COLUMN_GLOBAL_DATA_DN
--Section--				
MX_VALIDFROM	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
--Column--				
MX_VALIDTO	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
--Section--				
--Line--				
--Section--				
--Column--				#MX_SAPC_COLUMN_SPECIFIC_DATA_DN
--Section--				
--Column--				
--Label--				#MX_SAPC_LABEL_VALID_FROM_DN
SAPC_IDEN_REP_VALIDFROM_ABAP1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
SAPC_IDEN_REP_VALIDFROM_ASJAVA	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
SAPC_IDEN_REP_VALIDFROM_HANA1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
--Column--				
--Label--				#MX_SAPC_LABEL_VALID_TO_DN
SAPC_IDEN_REP_VALIDTO_ABAP1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
SAPC_IDEN_REP_VALIDTO_ASJAVA	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
SAPC_IDEN_REP_VALIDTO_HANA1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
--Tab--				#MX_SAPC_TAB_USER_GROUP_START_MENU_DN
--Section--				
--Column--				#MX_SAPC_COLUMN_GLOBAL_DATA_DN
MX_ADMIN_UNIT	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
--Column--				
MX_START_MENU	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
--Section--				
--Line--				
--Section--				
--Column--				#MX_SAPC_COLUMN_SPECIFIC_DATA_DN
--Section--				
--Column--				
--Label--				#MX_SAPC_LABEL_ADMIN_UNIT_DN
SAPC_IDEN_REP_ADMIN_UNIT_ABAP1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
--Column--				
--Label--				#MX_SAPC_LABEL_START_MENU_DN
SAPC_IDEN_REP_START_MENU_ABAP1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
--Tab--				#MX_SAPC_TAB_ACCOUNT_DN
--Section--				
--Column--				#MX_SAPC_COLUMN_GLOBAL_DATA_DN
MX_DISABLED	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
MX_PASSWORD_DISABLED	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
--Section--				
--Line--				
--Section--				
--Column--				#MX_SAPC_COLUMN_SPECIFIC_DATA_DN
--Section--				
--Column--				
--Label--				#MX_SAPC_LABEL_ACCOUNT_DISABLE_DN
SAPC_IDEN_REP_DISABLED_ABAP1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
SAPC_IDEN_REP_DISABLED_ASJAVA	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
SAPC_IDEN_REP_DISABLED_HANA1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
--Column--				
--Label--				#MX_SAPC_LABEL_PASSWORD_DISABLED_DN
SAPC_IDEN_REP_PASSWORD_DISABLED_ABAP1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
SAPC_IDEN_REP_PASSWORD_DISABLED_ASJAVA	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
SAPC_IDEN_REP_PASSWORD_DISABLED_HANA1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	



--Column--					
--Label--					#MX_SAPC_LABEL_LAST_LOGON_DN
SAPC_IDEN_REP_LASTLOGON_DATETIME_ABAP1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
--Column--					
--Label--					#MX_SAPC_LABEL_LOCKED_WRONG_LOGON_DN
SAPC_IDEN_REP_LOCKED_WRONG_LOGON_ABAP1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
--Tab--					#MX_SAPC_TAB_LICENSE_DATA_DN
--Section--					
--Column--					#MX_SAPC_COLUMN_GLOBAL_DATA_DN
--Section--					
--Column--					
SAPC_IDEN_LICENSE_TYPE	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
--Column--					
SAPC_IDEN_LICENSE_SPEC_VERS	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
--Section--					
--Column--					#MX_SAPC_COLUMN_SPECIFIC_DATA_DN
--Section--					
--Column--					
--Label--					#MX_SAPC_LABEL_LICENSE_TYPE_DN
SAPC_IDEN_REP_LICENSE_TYPE_ABAP1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
--Column--					
--Label--					#MX_SAPC_LABEL_LICENSE_COUNTRY_SURCHARGE...
SAPC_IDEN_REP_LICENSE_COUNTRY_SURCHARGE_AB...	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
--Column--					
--Label--					#MX_SAPC_LABEL_LICENSE_SPECIAL_VERSION_DN
SAPC_IDEN_REP_LICENSE_SPEC_VERS_ABAP1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
--Tab--					#MX_SAPC_TAB_CHANGE_HISTORY_DN
--Section--					
--Column--					#MX_SAPC_COLUMN_SPECIFIC_DATA_DN
--Section--					
--Column--					
--Label--					#MX_SAPC_LABEL_CREATION_DATE_DN
SAPC_IDEN_REP_CREATION_DATE_ABAP1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
--Column--					
--Label--					#MX_SAPC_LABEL_LAST_MOD_DATE_DN
SAPC_IDEN_REP_LASTMODDATE_ABAP1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
--Column--					
--Label--					#MX_SAPC_LABEL_LAST_MOD_TIME_DN
SAPC_IDEN_REP_LASTMODTIME_ABAP1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
--Column--					
--Label--					#MX_SAPC_LABEL_LAST_MODIFIER_DN
SAPC_IDEN_REP_LASTMODIFIER_ABAP1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
--Tab--					#MX_SAPC_TAB_USER_CATEGORY_DN
--Section--					
--Column--					#MX_SAPC_COLUMN_GLOBAL_DATA_DN
MX_USER_CATEGORY	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
--Section--					
--Column--					#MX_SAPC_COLUMN_SPECIFIC_DATA_DN
--Section--					
--Column--					
--Label--					#MX_SAPC_LABEL_USER_CATEGORY_DN
SAPC_IDEN_REP_USER_CATEGORY_ABAP1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
--Tab--					#MX_SAPC_TAB_USER_PARAMETERS_DN
--Section--					
--Column--					#MX_SAPC_COLUMN_GLOBAL_DATA_DN
--Section--					
MX_PARAMETER	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
--Line--					
--Label--					#MX_SAPC_HANA_PARAMS_FORMAT
MX_USER_PARAMS	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
--Section--					
--Line--					
--Section--					
--Column--					#MX_SAPC_COLUMN_SPECIFIC_DATA_DN
--Section--					
--Column--					
--Label--					#MX_SAPC_LABEL_PARAMETER_DN
SAPC_IDEN_REP_PARAMETER_ABAP1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
--Column--					
--Label--					#MX_SAPC_LABEL_USER_PARAMETERS_DN
SAPC_IDEN_REP_HANA_PARAMETERS_HANA1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
--Tab--					#MX_SAPC_TAB_PRINTERSETTINGS_DN
--Section--					
--Column--					#MX_SAPC_COLUMN_GLOBAL_DATA_DN
MX_PRINTERSETTINGS_SPLD	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
MX_PRINTERSETTINGS_SPDB	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
MX_PRINTERSETTINGS_SPDA	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
MX_PRINTERSETTINGS_SPLG	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
--Section--					
--Line--					
--Section--					
--Column--					#MX_SAPC_COLUMN_SPECIFIC_DATA_DN
--Section--					
--Column--					
--Label--					#MX_MX_PRINTERSETTINGS_SPLD_DN
SAPC_IDEN_REP_PRINTERSETTINGS_SPLD_ABAP1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		



Repeat the same steps for the form *SAPC Change Identity – System Specific Data*. But for this form the *Read Only* flag needs to be un-tagged for those system specific attributes that are supposed to be maintainable by user administrators.

#### Reading value help data for system-specific attributes from an SAP AS ABAP back-end system:

- Certain system-specific attributes are using value help data in the Web UI so that the user can select from a list of allowed, predefined values. This value help data needs to be filled initially. If the job *SAPC AS ABAP - Read System Specific AttrValueHelp* has not been executed yet, please do so as described in [RepositoryJobsABAP](#).

#### Configuring a set of system-specific attributes to use value help data:

- Some system-specific attributes should be set to use only allowed values from the value help table.
  1. Navigate to your *Identity Store* → *Identity Store Schema* → *Attributes*.
  2. For the attributes below, go to the *Attribute values* tab and configure the use of value help data.
  3. Select the *Value help* radio button and enter **mxl\_AttrValueHelp** in the *Table name* field for all attributes.
  4. Be sure to always select the *Language dependent* flag.

Attribute Name	Values ID
SAPC_IDEN_REP_USER_CATEGORY_<rep>	SAPC_IDEN_REP_USER_CATEGORY_<rep>
SAPC_IDEN_REP_ADMIN_UNIT_<rep>	SAPC_IDEN_REP_USER_CATEGORY_<rep>
SAPC_IDEN_REP_LICENSE_TYPE_<rep>	SAPC_IDEN_REP_LICENSE_TYPE_<rep>



The following attributes are already part of the job *SAPC AS ABAP - Read System Specific AttrValueHelp*, but caused by technical limitations, they cannot be loaded automatically by default. Nevertheless, it is possible to set up custom value help data for these attributes and use the value help with your custom value help data. The job *SAPC – Read AttrValueHelp* of the package *com.sap.rds.idm.system.administration* can be used to upload a set of data for those (see [UploadAttributeValueHelpData](#)). The attributes are:

- *SAPC\_IDEN\_REP\_START\_MENU\_* %\$rep.\$NAME%
- *SAPC\_IDEN\_REP\_PARAMETER\_* %\$rep.\$NAME%

## 5.8 Removing a Backend System

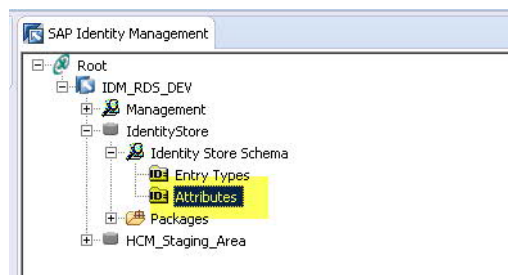
If you need to remove a backend system from the SAP IDM configuration you have to perform the following steps. Some of the deletion steps will be executed by a job; other steps have to be executed manually. Please make sure you execute the steps in the described order.

### Procedure

1. In the Administration and Monitoring interface for Identity Management ([http\(s\)://<host>:<port>/idm/admin](http(s)://<host>:<port>/idm/admin)) open the tab *System Configuration*.
2. Make sure *Packages* is selected in the left hand pane of the screen.

3. Enter `com.sap.rds.idm.system.administration` in the filter field and click on **Go**.
4. In the lower section of the page select the tab *Constants*.
5. Go to constant `SAPC_REMOVE_REPOSITORY_NAME` and enter the repository name that you want to remove.
6. Now select the tab *Jobs*.
7. Select job *SAPC - Removal of Repository* from the table shown in the left panel of the lower section, check that the dispatcher settings for the job are fine and run the job by clicking **Run Now** from the information section.
8. Afterwards perform the following steps manually:
  - a. Delete System Specific Attributes

- Go to your Identity Store Schema within Eclipse
- Click on *Attributes*



**Figure 31 – Attributes of Identity Store Schema**

- Filter for your specific name of the backend system (example CRM102\_01)

Attribute List/IdentityStore				
CRM102_01				
Name	Add Attribute Process	Modify Attribute Process	Delete Attribute Process	Description
ACCOUNTCRM102_01				Account name in repository CRM102_01
SAPC_IDEN_REP_ADMIN_UNIT_CRM102_01				System specific MX_PERSON attribute storing the user gr...
SAPC_IDEN_REP_CREATED_BY_CRM102_01				Created by user in repository CRM102_01
SAPC_IDEN_REP_CREATION_DATE_CRM102_01				Date when the user has been created in repository CRM1...
SAPC_IDEN_REP_DISABLED_CRM102_01				System specific MX_PERSON attribute that stores if a use...
SAPC_IDEN_REP_ENCRYPTED_PASSWORD_CRM10...				System specific MX_PERSON attribute storing the system ...
SAPC_IDEN_REP_LASTLOGON_DATETIME_CRM102...				Date and time when the user performed the last local log...
SAPC_IDEN_REP_LASTMODDATE_CRM102_01				System specific MX_PERSON attribute storing the date of...
SAPC_IDEN_REP_LASTMODIFIER_CRM102_01				System specific MX_PERSON attribute storing the user th...
SAPC_IDEN_REP_LASTMODTIME_CRM102_01				System specific MX_PERSON attribute storing the time of ...
SAPC_IDEN_REP_LICENSE_COUNTRY_SURCHARGE...				System specific MX_PERSON attribute storing the license i...
SAPC_IDEN_REP_LICENSE_SPEC_VERS_CRM102_01				System specific MX_PERSON attribute storing the license i...
SAPC_IDEN_REP_LICENSE_TYPE_CRM102_01				System specific MX_PERSON attribute storing the license i...
SAPC_IDEN_REP_LOCKED_WRONG_LOGON_CRM1...				System specific MX_PERSON attribute storing system spe...
SAPC_IDEN_REP_PARAMETER_CRM102_01				User parameters for repository CRM102_01
SAPC_IDEN_REP_PASSWORD_DISABLED_CRM102_01				System specific MX_PERSON attribute storing the passwo...
SAPC_IDEN_REP_PRINTERSETTINGS_SPLD_CRM10...				Users default printer for repository CRM102_01
SAPC_IDEN_REP_START_MENU_CRM102_01				System specific MX_PERSON attribute storing the start m...
SAPC_IDEN_REP_USER_CATEGORY_CRM102_01				System specific MX_PERSON attribute storing the user ca...
SAPC_IDEN_REP_VALIDFROM_CRM102_01				System specific MX_PERSON attribute storing valid from d...
SAPC_IDEN_REP_VALIDTO_CRM102_01				System specific MX_PERSON attribute storing valid to dat...

**Figure 32 – Filtering of Attributes in Identity Store Schema**

- Delete the respective attributes manually
- b. Delete System Specific Database Tables
    - Delete the system specific database tables manually on your database; this usually requires involvement of your DBA (database administrator)

- An overview about the respective system specific database tables is provided in the table below.

Backend System Type	Table Name	Comments
SAP AS ABAP	sapc_{\$REP.\$NAME%}_AGR_1252 sapc_{\$REP.\$NAME%}_AGR_AGRS sapc_{\$REP.\$NAME%}_AGR_DEFINE sapc_{\$REP.\$NAME%}_company sapc_{\$REP.\$NAME%}_profile sapc_{\$REP.\$NAME%}_profileAssign sapc_{\$REP.\$NAME%}_role sapc_{\$REP.\$NAME%}_roleAssign sapc_{\$REP.\$NAME%}_TSAD2 sapc_{\$REP.\$NAME%}_TSAD2T sapc_{\$REP.\$NAME%}_TSAD3 sapc_{\$REP.\$NAME%}_TSAD3T sapc_{\$REP.\$NAME%}_TSAD4 sapc_{\$REP.\$NAME%}_TSAD5 sapc_{\$REP.\$NAME%}_TSAD5T sapc_{\$REP.\$NAME%}_TSP03 sapc_{\$REP.\$NAME%}_TUTYPNOW sapc_{\$REP.\$NAME%}_user sapc_{\$REP.\$NAME%}_userGroups sapc_{\$REP.\$NAME%}_userParameter sapc_{\$REP.\$NAME%}_usgrp sapc_{\$REP.\$NAME%}_usgrpt sapc_{\$REP.\$NAME%}_USHIST sapc_{\$REP.\$NAME%}_USR02 sapc_recon_{\$REP.\$NAME%}_*	
SAP AS JAVA	sapc_{\$REP.\$NAME%}_user sapc_{\$REP.\$NAME%}_role sapc_{\$REP.\$NAME%}_group sapc_{\$REP.\$NAME%}_roleAssign sapc_{\$REP.\$NAME%}_groupAssign sapc_{\$REP.\$NAME%}_check sapc_recon_{\$REP.\$NAME%}_*	
SAP HANA	sapc_{\$REP.\$NAME%}_check sapc_{\$REP.\$NAME%}_features sapc_{\$REP.\$NAME%}_users sapc_{\$REP.\$NAME%}_roles sapc_{\$REP.\$NAME%}_applprivs sapc_{\$REP.\$NAME%}_systemprivileges sapc_{\$REP.\$NAME%}_usersroles sapc_{\$REP.\$NAME%}_userssystemprivs sapc_{\$REP.\$NAME%}_usersappprivs sapc_{\$REP.\$NAME%}_usersParams sapc_{\$REP.\$NAME%}_samlproviders	

	sapc_%\$REP.\$NAME%_usersSaml sapc_%\$REP.\$NAME%_usersX509 sapc_recon_%\$REP.\$NAME%_*	
MS Active Directory / LDS	sapc_%\$REP.\$NAME%_check sapc_%\$REP.\$NAME%_entryParents sapc_%\$REP.\$NAME%_group sapc_%\$REP.\$NAME%_groupAssign sapc_%\$REP.\$NAME%_user sapc_%\$REP.\$NAME%_check sapc_%\$REP.\$NAME%_entryParents sapc_%\$REP.\$NAME%_group sapc_%\$REP.\$NAME%_groupAssign sapc_%\$REP.\$NAME%_user sapc_recon_%\$REP.\$NAME%_*	

## 6 Approval Configuration

Details about the handling of approval workflows of the SAP IDM product can be found at:

[http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/aa/8d53c32c224b0e8fb0d8ad028cf629/frameset.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/aa/8d53c32c224b0e8fb0d8ad028cf629/frameset.htm)

### 6.1 Approval Workflows of IDM RDS Package

The following default approval workflows are available as part of the IDM RDS Package (contained in package *com.sap.rds.idm.approval*):

- 1 step – Manager Approval
- 2 step – Manager and Role Approver Approval
- 2 step – Manager and Role Member Approval
- 3 step – Manager and Role Approver and Role Member Approval

The role approver scenario is typically used for cases where you have responsible persons configured (owner of the role or privilege in SAP IDM) that should approve the assignments.

The role member scenario can be used in a flexible way for use cases such as:

- All members of the security or compliance department
- Certain special members of the IT department

The role member scenario is limited to one dedicated role that has to be configured in package constant *SAPC\_APPROVAL\_ROLE*.

Each approval step has a default escalation route defined.

The IDM RDS approval workflows are using the standard IDM mail templates for approval related E-Mails.

### 6.2 Required Master Data for Approval Workflows

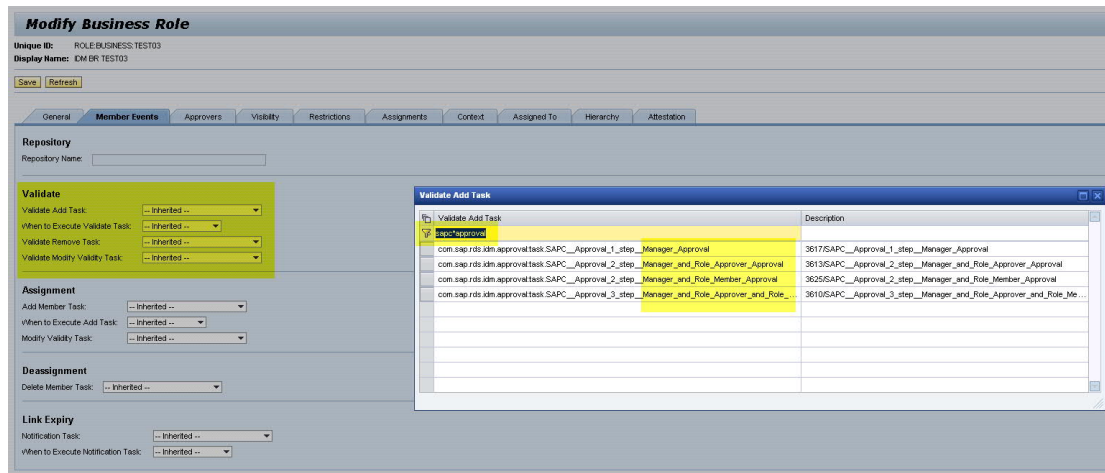
In the table below you can find an overview about the required SAP IDM master data for each approval step:

Approval Step	Master Data
Manager Approval	All users that are maintained as manager (attribute <i>MX_MANAGER</i> ) of the respective user
Role Approver Approval	All users that are configured as approvers (attribute <i>MX_APPROVERS</i> ) of the respective privilege or role
Role Member Approval	All user members assigned to the role which is configured in package constant <i>SAPC_APPROVAL_ROLE</i>
Escalation Approval for each above step	All user members assigned to role <i>SAPC_IDM_ESCALATION_APPROVERS</i>

### 6.3 Enable Approval Workflow

To enable the approval workflow for a given privilege or business role you have to perform the following steps:

- Start the form *Modify Privilege Details* or *Modify Business Role*
- Go to tab Member Events
- Depending on your use case set the validate tasks according to your requirements
  - To find the RDS workflow tasks you can search for “approval”
  - Validate Add Task
  - Validate Remove Task
  - Validate Modify Validity Task



**Modify Business Role**

Unique ID: ROLE-BUSINESS-TEST03  
Display Name: KM BR TEST03

Save Refresh

General Member Events Approvers Visibility Restrictions Assignments Context Assigned To Hierarchy Attestation

Repository  
Repository Name:

**Validate**

Validate Add Task: -- Inherited --  
When to Execute Validate Task: -- Inherited --  
Validate Remove Task: -- Inherited --  
Validate Modify Validity Task: -- Inherited --

**Assignment**

Add Member Task: -- Inherited --  
When to Execute Add Task: -- Inherited --  
Modify Validity Task: -- Inherited --

**Deassignment**

Delete Member Task: -- Inherited --

**Link Expiry**

Notification Task: -- Inherited --  
When to Execute Notification Task: -- Inherited --

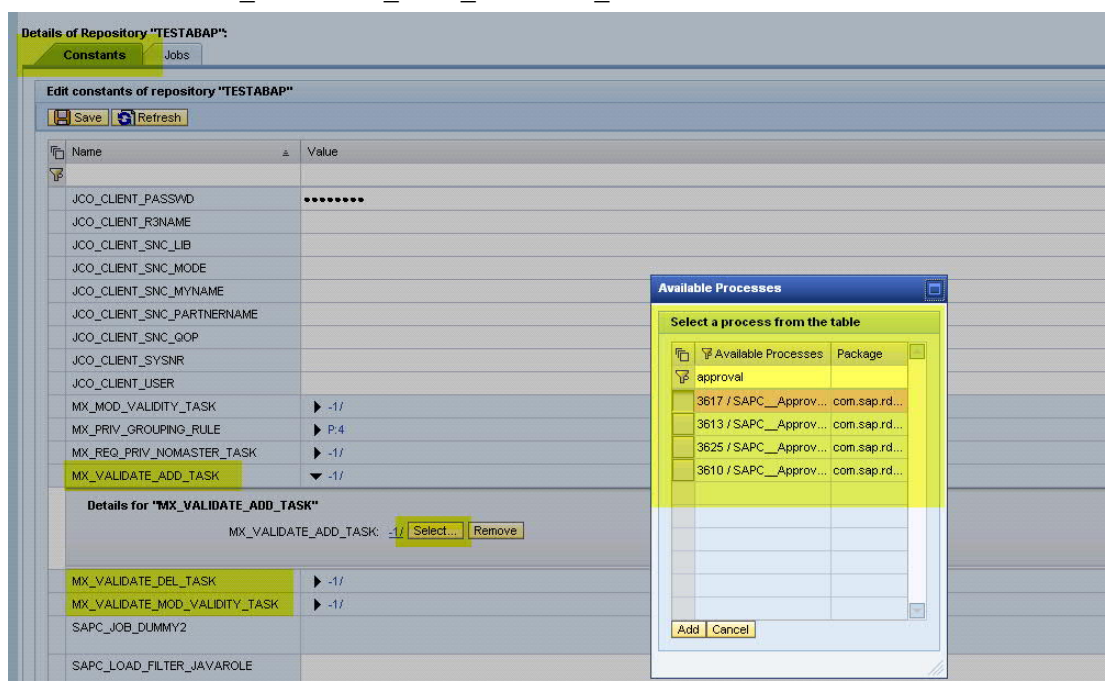
**Validate Add Task**

Task	Description
com.sap.rds.idm.approvaltask.SAPC__Approval_1_step_Manager_Approval	3617/SAPC__Approval_1_step_Manager_Approval
com.sap.rds.idm.approvaltask.SAPC__Approval_2_step_Manager_and_Role_Approver_Approval	3613/SAPC__Approval_2_step_Manager_and_Role_Approver_Approval
com.sap.rds.idm.approvaltask.SAPC__Approval_2_step_Manager_and_Role_Member_Approval	3625/SAPC__Approval_2_step_Manager_and_Role_Member_Approval
com.sap.rds.idm.approvaltask.SAPC__Approval_3_step_Manager_and_Role_Approver_and_Role_Member_Approval	3610/SAPC__Approval_3_step_Manager_and_Role_Approver_and_Role_Member_Approval

- By saving your changes the respective privilege or business role is enabled for the workflow

You can also activate the workflow on repository level so that it is inherited to all privileges of this repository. To do so please follow the steps:

- In the Administration and Monitoring interface for Identity Management ([http\(s\)://<host>:<port>/idm/admin](http(s)://<host>:<port>/idm/admin)) open the tab *System Configuration*.
- Make sure *Repositories* is selected in the left hand pane of the screen.
- Select your repository
- Make sure that the tab *Constants* is enabled and maintain the following repository constants based on your requirements:
  - To find the correct process you can search for "approval"
  - MX\_VALIDATE\_ADD\_TASK
  - MX\_VALIDATE\_DEL\_TASK
  - MX\_VALIDATE\_MOD\_VALIDITY\_TASK



**Details of Repository "TESTABAP"**

Constants Jobs

Edit constants of repository "TESTABAP"

Save Refresh

Name	Value
JCO_CLIENT_PASSWD	*****
JCO_CLIENT_R3NAME	
JCO_CLIENT_SNC_LIB	
JCO_CLIENT_SNC_MODE	
JCO_CLIENT_SNC_MYNAME	
JCO_CLIENT_SNC_PARTNERNAME	
JCO_CLIENT_SNC_QOP	
JCO_CLIENT_SYSNR	
JCO_CLIENT_USER	
MX_MOD_VALIDITY_TASK	-1/
MX_PRIV_GROUPING_RULE	P.4
MX_REQ_PRIV_NOMASTER_TASK	-1/
MX_VALIDATE_ADD_TASK	-1/

**Details for "MX\_VALIDATE\_ADD\_TASK"**

MX\_VALIDATE\_ADD\_TASK: -1/ Select... Remove

MX\_VALIDATE\_DEL\_TASK: -1/

MX\_VALIDATE\_MOD\_VALIDITY\_TASK: -1/

SAPC\_JOB\_DUMMY2

SAPC\_LOAD\_FILTER\_JAVAROLE

**Available Processes**

Select a process from the table

Available Processes	Package
approval	
3617 / SAPC__Approv...	com.sap.rd...
3613 / SAPC__Approv...	com.sap.rd...
3625 / SAPC__Approv...	com.sap.rd...
3610 / SAPC__Approv...	com.sap.rd...

Add Cancel

More complex settings can be done based on custom jobs depending on the special requirements.



## 6.4 Configuration of Approval Steps

If needed you can configure the respective approval steps based on your requirements. Configurable options are based on the standard IDM approval task settings. More details can be found in the standard documentation:

[http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/20/01b506b8a64188adcbaa8b8087264d/frameset.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/20/01b506b8a64188adcbaa8b8087264d/frameset.htm)

[http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/e8/9a447e443348f4a8a46191c26753ff/content.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/e8/9a447e443348f4a8a46191c26753ff/content.htm)

## 7 Further Configuration Details

### 7.1 Script `sapc_localStandardScriptsContainer`

Mainly caused by product limitations in the early version(s) of SAP IDM 8.0 it was necessary to collect certain SAP standard scripts in one container script to enable the IDM RDS delivery.

In the below table you can find an overview about the scripts contained in this container script.

Script	Package / Connector	Comments
<code>sap_abap_checkAcademicTitleValue</code>	ABAP	
<code>sap_abap_checkAdminValue</code>	ABAP	
<code>sap_abap_checkGroupValue</code>	ABAP	
<code>sap_abap_checkNull</code>	ABAP	
<code>sap_abap_checkPrefixValue</code>	ABAP	
<code>sap_abap_checkPrinterValue</code>	ABAP	
<code>sap_abap_checkSalutationValue</code>	ABAP	
<code>sap_abap_checkSupplementValue</code>	ABAP	
<code>sap_abap_skipExistingValueHelpEntry</code>	ABAP	
<code>sap_setRequiredPrivilege</code>	ABAP	
<code>sap_getRepositoryType</code>	ABAP	
<code>sap_getPrivilegeType</code>	ABAP	
<code>sap_abap_generateSelectValidityStatement</code>	ABAP	
<code>sap_getAccountPrivilegesWithDelimiter</code>	ABAP	
<code>sap_bs_stopMsSqlPass</code>	BusinessSuite	
<code>sap_bs_stopOraclePass</code>	BusinessSuite	
<code>sap_asj_getLocaleLanguage</code>	AS Java	
<code>sap_asj_getLocaleVariant</code>	AS Java	
<code>sap_asj_checkSPMLValidDate</code>	AS Java	
<code>sap_ad_checkTable</code>	AD	
<code>sap_ad_cleanMemberOf</code>	AD	
<code>sap_removeMemberOfNotPartOfStartingPointGroups</code>	AD	
<code>sap_findPrimaryDeltaObject</code>	General	
<code>sap_findSecondaryDeltaObject</code>	General	
<code>sap_core_getDatabasePrefix</code>	General	
<code>sap_noop</code>	General	

In case the respective SAP standard script is changed and will be updated by a patch it is necessary to check and update the same script within `sapc_localStandardScriptsContainer` as well.

### 7.2 Configuration of System Specific Attributes

By default delivery the IDM RDS package is configured to have system specific attributes that depend on values and attribute events of global attributes.

This means once the global attribute value is changed in SAP IDM the respective system specific attributes are updated as well.

The list of supported attributes is configured in package constant `SAPC_SYSTEM_SPECIFIC_ATTRIBUTES` of package `com.sap.rds.idm.core`. The default value of this constant is:

- `MX_VALIDFROM;MX_VALIDTO;MX_ENCRYPTED_PASSWORD;MX_PASSWORD_DISABLED`

The list of supported repositories for the eventing of the respective global attributes is configured in package constant `SAPC_REPOSITORY_TYPES_SUPPORTING_SYSTEM_SPECIFIC_ATTRIBUTES` of package `com.sap.rds.idm.core`. The default value of this constant is:

- `ABAP;JAVA;LDAP;SAP_IN_MEMORY_DB`

By changing the respective values you can limit the eventing for certain attributes and / or repositories.

## 7.3 Presentation Types for Attributes

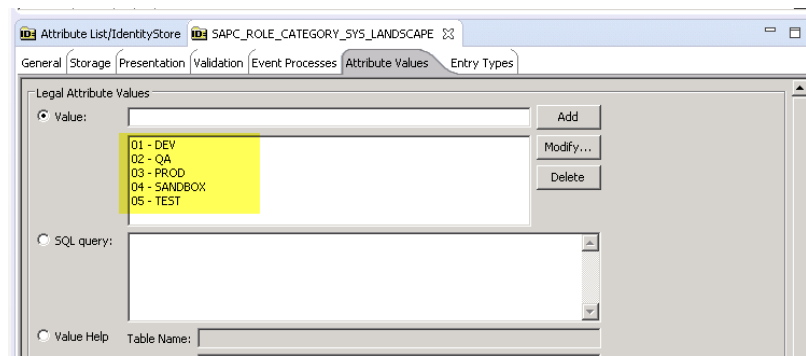
If you want to change the presentation type of an attribute automatically by the system (as part of job or task) you can use the stored procedure `mxi_schema_attr_ns`. This procedure supports the following presentation types to be set.

ID	Name
1	Single Line
2	MultiLine
3	SingleSelect
4	MultiSelect
5	Boolean
8	Referral
9	File
10	Lookup
12	Radio Button
13	Date
15	Mail
16	ObjectValueHelp
17	AutoContexts

## 7.4 Attribute Value Help for Attribute `SAPC_ROLE_CATEGORY_SYS_LANDSCAPE`

The attribute `SAPC_ROLE_CATEGORY_SYS_LANDSCAPE` is used in the forms `SAPC Create Business Role` and `SAPC Modify Business Role` and allows you to map a business role to one or several landscapes (such as DEV, QA, PROD).

If needed you have to manually adapt the list of legal attribute values for this attribute in the SAP IDM schema.



## 8 Mass Administration Jobs

In comparison to previous RDS versions, the approach of mass administration jobs has been changed.

The major change is, that the jobs are not executed from the administration UI (*/idm/admin*) anymore, but from the manage UI (*/idm*) as a request (entry type *SAPC Request*). The user creating the request in the UI is stored inside the request as well as the uploaded file. This adds transparency to the performed operations, as the information who made the upload is stored not only on the request object but also on the changed objects itself.

Furthermore, the execution status and further status information is stored on the request objects (described in detail in the sections below). This assures that a job cannot be executed in parallel which prevents unpredictable and unwanted side effects and allows the user to troubleshoot job execution errors directly from the request object.

### 8.1 The Request Object

The following information are stored inside the request object:

- Requestor (who has started the mass operation). This information is also used to store user information on the changed objects
- Upload / Download File of the mass operation, allowing auditing of mass operations
- Job state and state information (who started the request, when was the job started and when did it finish)
- Information about the job (ID/GUID)

#### 8.1.1 Attributes **SAPC\_REQ\_STATE & STATE\_INFO**

In order to make the state and important messages visible in the UI, the new attributes **SAPC\_REQ\_STATE** and **SAPC\_REQ\_STATE\_INFO** are holding informations about the execution status, which are:

- PREPROCESSING (the state when the process is preparing the job execution, before the job is actually running)
- RUNNING
- IDLE
- ERROR

While the status information will show messages like who started the request when, when did the job execution actually start and finish as well as error messages if the job run into errors (Set by new pass in the jobs *Set Job State Finished*).

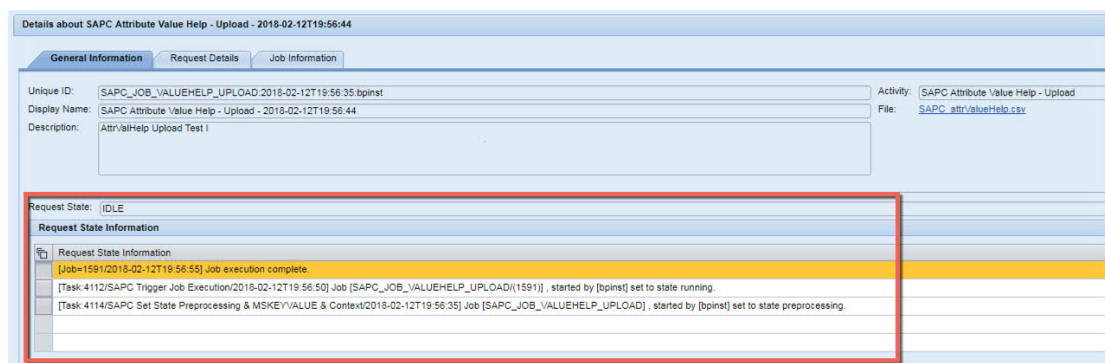
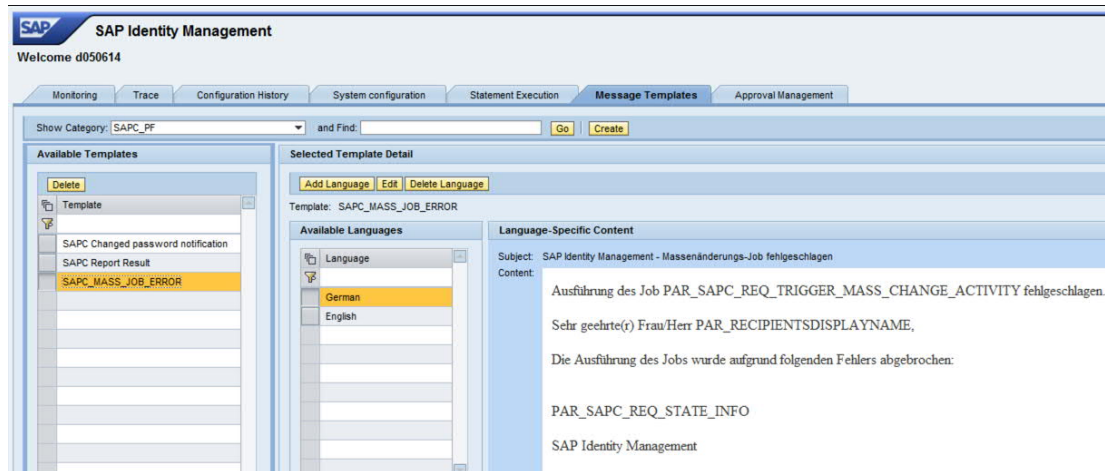


Figure 33 - Mass Administration Request State

#### 8.1.2 Error Mail Template **SAPC\_MASS\_JOB\_ERROR**

Since the user does not get direct feedback about whether a job is running or not when creating a request object in UI, the only way to notify a user in case the job could not be

executed because it is running already is to notify the user by mail. The according mail template created for that case is `SAPC_MASS_JOB_ERROR`.



### 8.1.3 The Process / Workflow

1. Users are creating a request object in UI
2. Process will check
  - a. Job to be existent
  - b. Job to be not in state running
3. If check was negative, an error mail will be send, if not the request will be set to state preprocessing before it finally will execute the job (`uRunJob()`).
4. The request state will be set to **RUNNING**.
5. A new pass *Set Job State Finished* has been added to the end of every job and will set the job state at the end of a job execution. If for example the job runs into an error, the job will determine the according request object by the job GUID and store the error messages on that request object, to make it visible in UI.

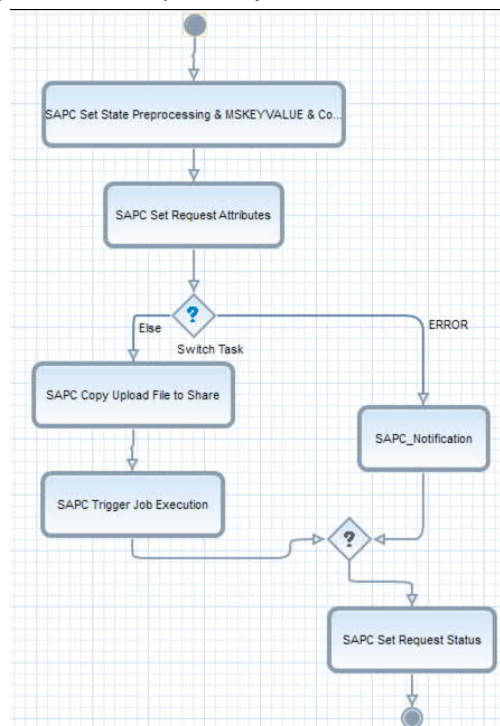


Figure 34 - Mass Administration Process Flow

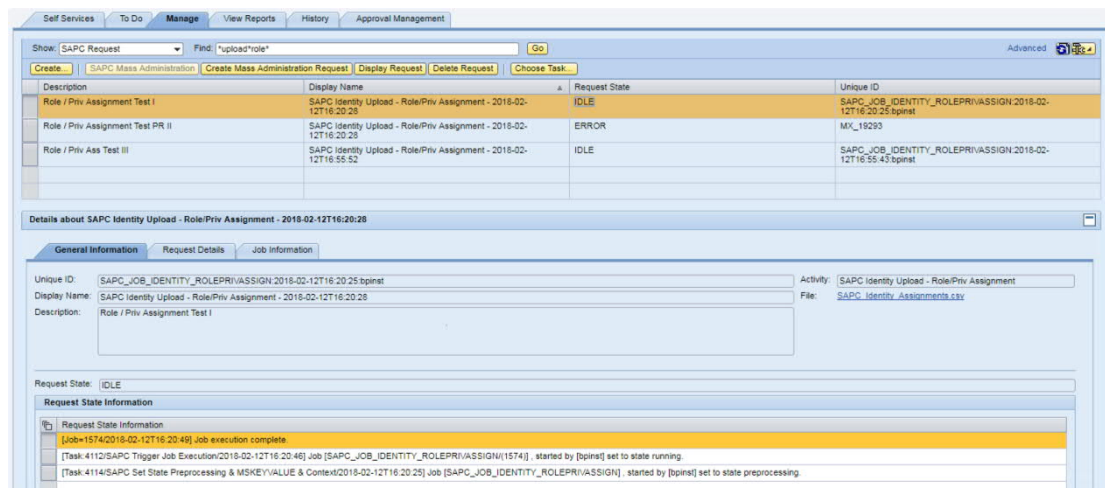


Figure 35 - The Mass Administration Request Object

## 8.2 Setting Up a Custom Mass Administration Job

Customers can utilize the mass administration logic of IdM Business Extensions for upload jobs as well as for download jobs. The following sequence shows the procedure with an example job called Upload Identities:

1. Creation of the Job inside the customer package.
  - a. Add following scripts to the job (in addition to the scripts required by the jobs logic)

Source Package	Script Name	
com.sap.rds.idm.core	sapc_core_script_functions	
com.sap.rds.idm.core	sapc_core_getNolock	
com.sap.rds.idm.core	sapc_getJobVar	
com.sap.rds.idm.mass.administration	sapc_setInitialJobConstants	
com.sap.rds.idm.mass.administration	sapc_getRequestMSKEY	
com.sap.rds.idm.mass.administration	sapc_setJobState	
com.sap.rds.idm.mass.administration	sapc_setFileAttributeContentForRequest	*
com.sap.rds.idm.core	sapc_getMskeyvalueFromMSKEY	*

\* only required for download jobs (jobs where data from IDM DB is downloaded)

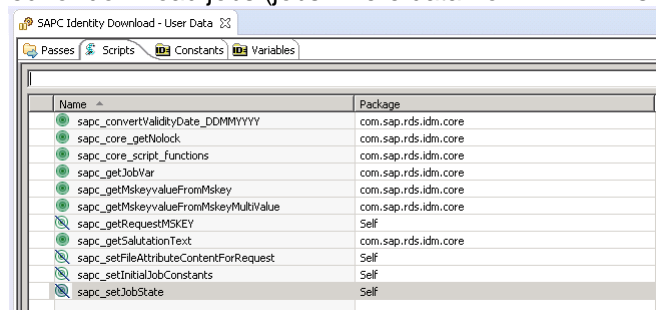
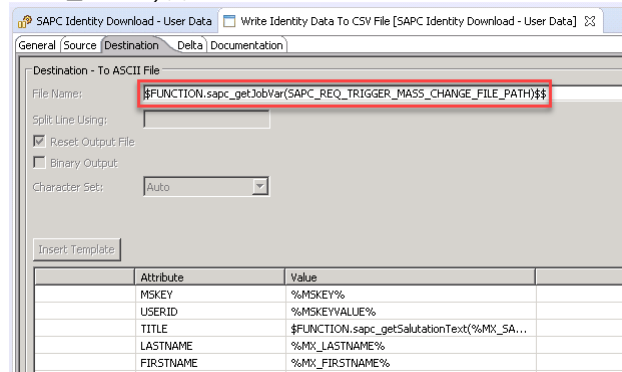


Figure 36 - Scripts for Custom Mass Administration Job

- b. The file name will be taken from the request object. In the pass where you read the file has to be of value  
`$FUNCTION.sapc_getJobVar(SAPC_REQ_TRIGGER_MASS_CHANGE_FI`

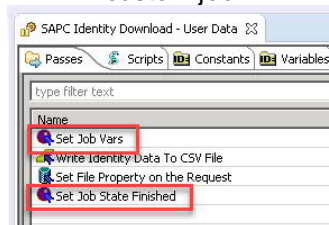


`LE_PATH)$$` as shown in the screenshot below:



**Figure 37 - File Name for Custom Mass Administration Job**

- c. Copy the first and the last pass from one of the existing mass administration jobs from the IdM Business Extensions Package into the same position of the custom job:

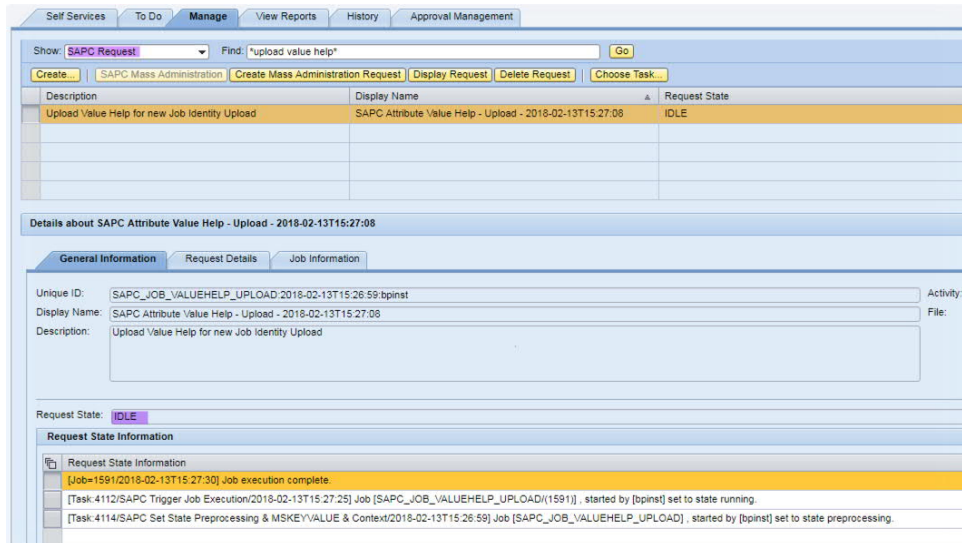


**Figure 38 - Pass Set Job State of Custom Mass Administration Job**

2. Upload a new entry for the value help of attribute *Activity* of entry type *SAPC Request*.
  - a. Prepare a file for the upload of the new value.  
The first line is :  
`ValId;ValKey;ValLocale;ValText`  
The second line is:  
`SAPC_REQ_TRIGGER_MASS_CHANGE_ACTIVITY;<GUID of custom Job>;<DisplayName of custom Job>`
  - b. Perform an upload of the file prepared in the previous step, using the *SAPC Request* as shown in the screenshot below

**Figure 39 - Upload Value Help for Custom Mass Administration Job**

- c. Check the request object, for the upload executed in step b and wait until it is in state IDLE.



The screenshot shows the SAP Self Services interface. The top navigation bar includes 'Self Services', 'To Do', 'Manage', 'View Reports', 'History', and 'Approval Management'. The 'Manage' tab is active. Below the navigation bar, there is a search bar with 'Show: SAPC Request' and a 'Find' field containing 'upload value help'. A 'Go' button is next to the search bar. Below the search bar, there are buttons for 'Create', 'SAPC Mass Administration', 'Create Mass Administration Request', 'Display Request', 'Delete Request', and 'Choose Task...'. The main table displays a list of requests. The first row is highlighted in orange and shows the following details:

Description	Display Name	Request State
Upload Value Help for new Job Identity Upload	SAPC Attribute Value Help - Upload - 2018-02-13T15:27:08	IDLE

Below the table, there is a section titled 'Details about SAPC Attribute Value Help - Upload - 2018-02-13T15:27:08'. This section contains tabs for 'General Information', 'Request Details', and 'Job Information'. The 'General Information' tab is active. It displays the following fields:

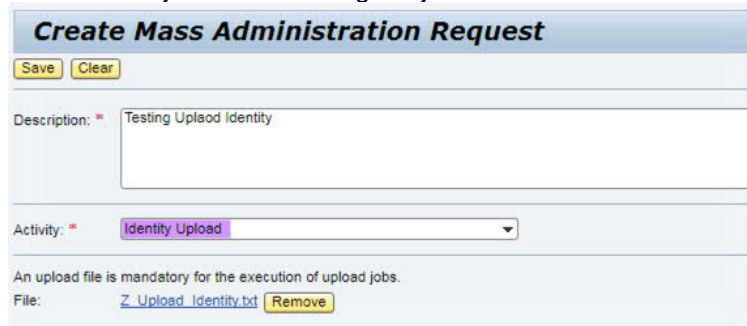
- Unique ID: SAPC\_JOB\_VALUEHELP\_UPLOAD:2018-02-13T15:26:59:bpinst
- Display Name: SAPC Attribute Value Help - Upload - 2018-02-13T15:27:08
- Description: Upload Value Help for new Job Identity Upload

Below these fields, there is a 'Request State' field showing 'IDLE'. Underneath, there is a 'Request State Information' section with a list of tasks:

- [Job=1591/2018-02-13T15:27:30] Job execution complete.
- [Task=4112/SAPC Trigger Job Execution/2018-02-13T15:27:25] Job [SAPC\_JOB\_VALUEHELP\_UPLOAD/(1591)] , started by [bpinst] set to state running.
- [Task=4114/SAPC Set State Preprocessing & MSKEYVALUE & Context/2018-02-13T15:26:59] Job [SAPC\_JOB\_VALUEHELP\_UPLOAD] , started by [bpinst] set to state preprocessing.

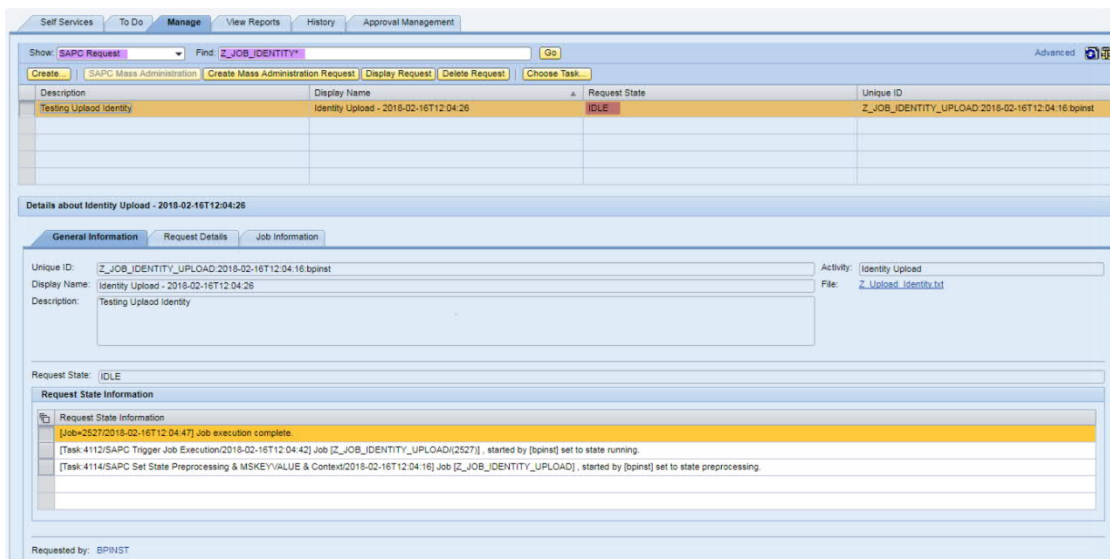
Figure 40 - The Request of Creation of a Custom Mass Administration Job

3. The created mass administration activity can be tested by creating a SAPC Request Object and selecting the job created above.



The screenshot shows the 'Create Mass Administration Request' form. It has a title bar with 'Save' and 'Clear' buttons. Below the title bar, there is a 'Description' field with the text 'Testing Upload Identity'. Below the description field, there is an 'Activity' dropdown menu with 'Identity Upload' selected. Below the activity dropdown, there is a text box stating 'An upload file is mandatory for the execution of upload jobs.' Below this text box, there is a 'File' field with the text 'Z\_Upload\_Identity.txt' and a 'Remove' button.

Figure 41 - Test Custom Mass Administration Job



The screenshot shows the SAP Self Services interface. The top navigation bar includes 'Self Services', 'To Do', 'Manage', 'View Reports', 'History', and 'Approval Management'. The 'Manage' tab is active. Below the navigation bar, there is a search bar with 'Show: SAPC Request' and a 'Find' field containing 'Z\_JOB\_IDENTITY'. A 'Go' button is next to the search bar. Below the search bar, there are buttons for 'Create', 'SAPC Mass Administration', 'Create Mass Administration Request', 'Display Request', 'Delete Request', and 'Choose Task...'. The main table displays a list of requests. The first row is highlighted in orange and shows the following details:

Description	Display Name	Request State	Unique ID
Testing Upload Identity	Identity Upload - 2018-02-16T12:04:26	IDLE	Z_JOB_IDENTITY_UPLOAD:2018-02-16T12:04:16:bpinst

Below the table, there is a section titled 'Details about Identity Upload - 2018-02-16T12:04:26'. This section contains tabs for 'General Information', 'Request Details', and 'Job Information'. The 'General Information' tab is active. It displays the following fields:

- Unique ID: Z\_JOB\_IDENTITY\_UPLOAD:2018-02-16T12:04:16:bpinst
- Display Name: Identity Upload - 2018-02-16T12:04:26
- Description: Testing Upload Identity

Below these fields, there is a 'Request State' field showing 'IDLE'. Underneath, there is a 'Request State Information' section with a list of tasks:

- [Job=2527/2018-02-16T12:04:47] Job execution complete.
- [Task=4112/SAPC Trigger Job Execution/2018-02-16T12:04:42] Job [Z\_JOB\_IDENTITY\_UPLOAD/(2527)] , started by [bpinst] set to state running.
- [Task=4114/SAPC Set State Preprocessing & MSKEYVALUE & Context/2018-02-16T12:04:16] Job [Z\_JOB\_IDENTITY\_UPLOAD] , started by [bpinst] set to state preprocessing.

At the bottom of the page, there is a 'Requested by:' field showing 'BPINST'.

Figure 42 - Check Result of Custom Mass Administration Job

## 9 Transport

Details about the transport of SAP IDM configurations from the DEVELOPMENT system to the CONSOLIDATION system and finally to the PRODUCTION system can be found in the standard documentation:

[http://help.sap.com/saphelp\\_nwidmic\\_80/helpdata/en/6a/4a929e683b496fbadd0307d8a6286b/content.htm?frameset=/en/d2/322e5c6b784e8ca315a8d7cbc1bc5d/frameset.htm](http://help.sap.com/saphelp_nwidmic_80/helpdata/en/6a/4a929e683b496fbadd0307d8a6286b/content.htm?frameset=/en/d2/322e5c6b784e8ca315a8d7cbc1bc5d/frameset.htm)

### 9.1 Considerations Regarding System Specific Attributes

The IDM RDS package provides functionality for system specific attribute values. Those attribute values are stored in specific attributes (for each connected backend system) that are typically different between the different SAP IDM landscapes or systems.

When you want to export your Identity Store schema and import it into another SAP IDM system you have to consider the system specific attributes. If those attributes differ between the SAP IDM systems you should perform a manual cleanup / correction of the SAP IDM export file(s) and remove all the attributes that should not be part of the export / import process. The main reason for this manual interaction is that currently it is not possible to limit the export to certain attributes only.

It is recommended to import the package *com.sap.rds.idm.forms.systemspecific* directly into each SAP IDM system. This package is usually not transported but directly imported. After the import you perform a manual configuration of the required attributes per each landscape.

## 10 Appendix

### 10.1 Overview of System Specific Attributes

#### 10.1.1 System Specific Attributes for SAP AS ABAP

Attribute	Description
ACCOUNT%\$rep\$NAME%	Account Attribute
SAPC_IDEN_REP_ENCRYPTED_PASSWORD_%\$rep\$NAME%	Encrypted Password
SAPC_IDEN_REP_PASSWORD_DISABLED_%\$rep\$NAME%	Password Disabled Status
SAPC_IDEN_REP_VALIDFROM_%\$rep\$NAME%	Valid From
SAPC_IDEN_REP_VALIDTO_%\$rep\$NAME%	Valid To
SAPC_IDEN_REP_DISABLED_%\$rep\$NAME%	Disabled Status
SAPC_IDEN_REP_LOCKED_WRONG_LOGON_%\$rep\$NAME%	Locked by Wrong Logon Attempts
SAPC_IDEN_REP_ADMIN_UNIT_%\$rep\$NAME%	User Group
SAPC_IDEN_REP_LASTMODDATE_%\$rep\$NAME%	Last Modification Date
SAPC_IDEN_REP_LASTMODTIME_%\$rep\$NAME%	Last Modification Time
SAPC_IDEN_REP_LASTMODIFIER_%\$rep\$NAME%	Last Modifier
SAPC_IDEN_REP_LASTLOGON_DATETIME_%\$rep\$NAME%	Last Logon Date & Time
SAPC_IDEN_REP_CREATED_BY_%\$rep\$NAME%	Created By
SAPC_IDEN_REP_CREATION_DATE_%\$rep\$NAME%	Creation Date
SAPC_IDEN_REP_LICENSE_TYPE_%\$rep\$NAME%	License Type
SAPC_IDEN_REP_LICENSE_COUNTRY_SURCHARGE_%\$rep\$NAME%	License Country Surcharge
SAPC_IDEN_REP_LICENSE_SPEC_VERS_%\$rep\$NAME%	License Special Version
SAPC_IDEN_REP_LICENSE_SAPSYSTEM_%\$rep\$NAME%	License Chargeable user in SAP System
SAPC_IDEN_REP_LICENSE_CLIENT_%\$rep\$NAME%	License Client
SAPC_IDEN_REP_LICENSE_NAME_%\$rep\$NAME%	License Name
SAPC_IDEN_REP_START_MENU_%\$rep\$NAME%	Start Menu
SAPC_IDEN_REP_USER_CATEGORY_%\$rep\$NAME%	User Group (multi-value)
SAPC_IDEN_REP_PARAMETER_%\$rep\$NAME%	User Parameter
SAPC_IDEN_REP_PRINTERSETTINGS_SPLD_%\$rep\$NAME%	Printer Settings

#### 10.1.2 System Specific Attributes for SAP AS Java

Attribute	Description
ACCOUNT%\$rep\$NAME%	Account Attribute
SAPC_IDEN_REP_ENCRYPTED_PASSWORD_%\$rep\$NAME%	Encrypted Password
SAPC_IDEN_REP_PASSWORD_DISABLED_%\$rep\$NAME%	Password Disabled Status
SAPC_IDEN_REP_VALIDFROM_%\$rep\$NAME%	Valid From
SAPC_IDEN_REP_VALIDTO_%\$rep\$NAME%	Valid To
SAPC_IDEN_REP_DISABLED_%\$rep\$NAME%	Disabled Status

### 10.1.3 System Specific Attributes for SAP HANA

Attribute	Description
ACCOUNT%\$rep\$NAME%	Account Attribute
SAPC_IDEN_REP_ENCRYPTED_PASSWORD_%\$rep\$NAME%	Encrypted Password
SAPC_IDEN_REP_PASSWORD_DISABLED_%\$rep\$NAME%	Password Disabled Status
SAPC_IDEN_REP_VALIDFROM_%\$rep\$NAME%	Valid From
SAPC_IDEN_REP_VALIDTO_%\$rep\$NAME%	Valid To
SAPC_IDEN_REP_DISABLED_%\$rep\$NAME%	Disabled Status
SAPC_IDEN_REP_HANA_PARAMETERS_%\$rep.\$NAME%	HANA Parameters

### 10.1.4 System Specific Attributes for AD / LDS

Attribute	Description
ACCOUNT%\$rep\$NAME%	Account Attribute
SAPC_IDEN_REP_ENCRYPTED_PASSWORD_%\$rep\$NAME%	Encrypted Password
SAPC_IDEN_REP_PASSWORD_NOTREQUIRED_%\$rep\$NAME%	Password Disabled Status
SAPC_IDEN_REP_VALIDFROM_%\$rep\$NAME%	Valid From
SAPC_IDEN_REP_VALIDTO_%\$rep\$NAME%	Valid To
SAPC_IDEN_REP_DISABLED_%\$rep\$NAME%	Disabled Status