USU Analytics
Administration Manual

December 2023
Product version 10.2.0

Table Of Contents

1. Introduction

This manual describes how to install the solution and how to administrate, configure and deploy content to it.

2. Installation

2.1. Prerequisites

This manual describes how to install the USU Analytics solution. The following sections summarize and outline issues that should be considered and understood prior to installation or performing changes to an established environment.

The core USU Analytics solution requires a suitable server layout. Usually this is done in three-tier architecture by separating the functional parts:

- Backend tier (data integration and data storage):
  - Database Management System
- Middle tier (business analytics server including webserver to provide frontend functionality):
  - USU Analytics BA Server
  - USU Analytics Toolkit
  - USU Analytics Job Scheduler
  - USU Analytics User Console
  - USU Analytics Analyzer
  - USU Analytics Dashboard Designer
- Frontend tier (design tools)
  - Report Designer

Based on the specific user and performance requirements, the architectural layout may differ from the one described above.

2.1.1. Check - Supported Platforms

The USU Analytics suite provides the core services. Depending on specific user and performance requirements, the standard installation can be deployed on a single server or multiple servers.

To run the solution properly, the hardware and software requirements should meet those described in the System Requirements document. It's possible to use less capable machines, but in most scenarios, this will result in an undesirable level of performance.

Even though the server components of USU Analytics can run in a 32-bit environment, we do not recommend nor support it. To run in a pure 64-bit environment, it must be ensured that a 64-bit operating system is installed. This includes ensuring that database management system and Java are 64-bit.

2.1.2. Check - Basic Skills and Knowledge

To install the software, the responsible person must be familiar with technical concepts and system administration operations pertaining to network services, including modifying the firewall to open specific ports, and adding services to the system startup and shutdown scripts. It is also required to have knowledge of using the operating system command line interface and/or graphical system administration tools. In addition, the installer should be familiar with the following software components:

- Database management systems and SQL administration clients
- Application server administration (Apache Tomcat 9.0.x (installed with standard installation))
- Operating systems administration (Windows Server 2019 or 2022 or modern Linux distributions)
- Oracle Java 11 JDK or OpenJDK 11 64-bit setup
- Web Browser
- XML/text editor

2.1.3. Check - Access Rights

The installation process requires having the ability to install software, open firewall ports, start and stop system services on the machine you are installing on and to reboot the machine.

The most important Access Rights are related to:

- OS user account under which the software is installed and run.
- In case of preinstalled database system: Permissions to create users and databases are required.
- In case of using LDAP: It may be required to create additional technical users and roles. Refer to the administration manuals for details.
- In case of virtualization: It may be required to attach and detach additional storage devices (USB, NAS, etc.) to upload required sources.
- In case of scheduled reports to be send via e-mail: SMTP server and account are required. For more details refer to the administration manuals.

2.1.4. Check – Binaries and Licenses

The installation requires several software packages and files.

- Install package of the latest version of USU Analytics
- Licenses for your USU Analytics version
- Suitable JDBC Driver (see System Requirements for supported drivers)
- Installed database management system (see System Requirements for supported DBMS)

On each server/client:

- **Proper installed** distribution of the Java **version 11 64-bit or OpenJDK 11 64-bit** (see System Requirements document) **(mandatory)**
- **Proper configured environment variable JAVA_HOME (mandatory)**
- **Proper installed tar archive executable (mandatory)**
- PDF Reader (optional)
- XML/TXT Editor (optional)

Usually, the installation is performed by USU staff which will provide the binaries in advance or at the day of installation. If, however, you want to install the software on your own, the suitable binaries can be provided upon request.

2.1.5. Check – Firewall Rules

Make sure that your firewall is configured server for the external application (e.g. USU Service Management or USU License Management) can:

- reach database of USU Analytics
- reach database of your source system
- be reached from network to allow enduser access

In case you use a reverse proxy or load balancer some rules have to be set up.

For details see Setup a Load Balancer (Proxy) (page 34).

2.1.6. Check - Database Configuration

For MySQL and MariaDB please make sure that the following parameter is set to 1:

```
lower_case_table_names = 1
```

If the parameter does not have value 1, the system cannot function.

Further information on this and other recommended parameter settings can be found in the chapters Recommended MySQL server settings (page 46) and Recommended MariaDB server settings (page 47).

For Oracle it is necessary, that the DB users `usu_si_ea`, `usu_si_dm` and `usu_si_md` have the grant to execute the DBMS_CRYPTO package. The installer automatically creates SQL scripts that contain these GRANTs. The package belongs to the SYS user, so the GRANTs must also be assigned by the SYS user. Please note that the names of the database users `usu_si_ea`, `usu_si_dm` and `usu_si_md` may differ in your environment.

```
-- SYS is the owner of DBMS_CRYPTO, so it must be executed/granted as such
GRANT EXECUTE ON DBMS_CRYPTO TO usu_si_ea;
GRANT EXECUTE ON DBMS_CRYPTO TO usu_si_dm;
GRANT EXECUTE ON DBMS_CRYPTO TO usu_si_m
```

2.1.7. Security Integration

The authentication mechanism of USU Analytics is based on the core services that come with the Spring Security pluggable authentication framework.

By default, the solution is configured to make use of a JDBC-based data access object that is tied to a Jackrabbit database. Users and roles are configured through the USU Analytics User Console.

USU Analytics, however, can be reconfigured to let the server use existing security tables in a different database, or to authenticate through your existing LDAP (including Active Directory) server or Central Authentication Service (CAS). As security is based on a framework, it is possible to create a custom data access object, or completely remove all authentication functionality.

For more details about LDAP integration, please, refer to the administration manual.

For more information about the integration of other security technologies, please, contact the USU Support Center.

2.1.8. Document your Installation

It is recommended to document your setup. You can use the template ***USU_Analytics_Installation_Protocol.docx*** we ship with our installation.

2.2. Structural Outline

USU Analytics solution is partially built out of the third-party software components such as Java Runtime Environment, database management system and application server. All these components are usually installed during standard installation process. However, if you want to install components separately, the requirements described in the system requirements document must be met. In such case, the related environment variables may have to be enhanced.

The following subsections outline and describe the most important directories of the solution and their structure, including the referential installation paths, which are not required to be necessarily followed, however are recommended. Please, consult further installation requirements with USU Support Center in case the different, than recommended, paths are going to be used.

2.2.1. Microsoft Windows Based Operating System

Path	Reference	Description
`d:\usu`	USU_HOME	Home directory of USU products
`d:\usu\vma`	VMA_HOME	UA home directory
`d:\usu\vma\server\pentaho-server`	VMA_BASERVER	USU Analytics BA Server
`d:\usu\vma\server\tomcat_toolkit`	TOMCAT_INSTALL_DIR	(Optional) dedicated Tomcat for USU Analytics Toolkit
`%VMA_BASERVER%\tomcat`		Home directory of Apache Tomcat on which BA Server is running
`%VMA_BASERVER%\tomcat\webapps\console`		USU Analytics User Console web application
`%VMA_BASERVER%\tomcat\logs`		Apache Tomcat log directory and USU Analytics Toolkit log folder
`%VMA_BASERVER%\logs`		BA Server log directory
`%VMA_BASERVER%\pentaho-solutions\system`		BA Server plugins and configuration files

2.2.2. UNIX Based Operating System

Path	Reference	Description
`/opt/usu`	USU_HOME	Home directory of USU products
`/opt/usu/si`	VMA_HOME	UA home directory
`/opt/usu/si/server/pentaho-server`	VMA_BASERVER	USU Analytics BA Server
`/opt/usu/tomcat_toolkit`	TOMCAT_INSTALL_DIR	(Optional) dedicated Tomcat for USU Analytics Toolkit
`%VMA_BASERVER%/tomcat`		Home directory of Apache Tomcat on which BA Server is running
`%VMA_BASERVER%/tomcat/webapps/console`		USU Analytics User Console web application
`%VMA_BASERVER%/tomcat/logs`		Apache Tomcat log directory and USU Analytics Toolkit log folder
`%VMA_BASERVER%/logs`		BA Server log directory

Path	Reference	Description
%VMA_BASERVER%/pentaho-solutions/system		BA Server plugins and configuration files

2.3. Installation of USU Analytics

We describe the installation on a Windows based OS. Where there are differences and discrepancies to installations on UNIX based OS, we draw attention to them by a comment.

2.3.1. Install BA Server

1. 

   Execute for UA Basic and UA Full installation.

   Some steps are not required for installation for USU Analytics for Software Asset Management .

2. Read the prerequisites chapter of this document, if not already done.
3. **If you are installing on a UNIX based OS**

   Create new user account (if BA server is not going to be installed and run under the existing one), e.g. usu_si_ba, and add it to a user group. E.g.:

   ```
   useradd –g <user_group_id> -m usu_si_ba
   ```

4. **If you are installing on a UNIX based OS**

   Create target installation path for %USU_HOME% if it does not exist and enhance the access rights according to the new created user, e.g.:

   ```
   mkdir –p <USU_HOME>
   chown –R usu_si_ba <USU_HOME>
   ```

5. Unzip the USU Analytics release package <UA RELEASE PACKAGE> and make it available on the UA application server.

   If you are installing on a UNIX based OS

   Check the release package <UA RELEASE PACKAGE>/platform/installer/usu_analytics-full_installation-<version number>.jar and enhance the access rights according to the previous steps. To avoid possible issues with access rights we recommend uploading the installation jar to %USU_HOME% directory and run the installation process from that directory, e.g.:

   ```
   chown –R usu_si_ba:<user_group_name> %USU_HOME%/ usu_analytics-full_installation-<version number>.jar chmod 755 usu_analytics-full_installation-<version number>.jar
   ```

6. Unzip the USU Analytics licenses and make them available on the UA application server.

   If you are installing on a UNIX based OS

   To avoid possible issues with access rights we recommend uploading the files to %USU_HOME% directory.

7. Get the appropriate JDBC-Driver for your USU Analytics and your source system database. Make them available on the UA application server.

   If you are installing on a UNIX based OS

   To avoid possible issues with access rights we recommend uploading the files to %USU_HOME% directory.

8.

We depend on the correct configuration of environment variable for java. Please check if JAVA_HOME is configured correctly and points to an Oracle Java 11 JDK or Open JDK 11. (i.e. `JAVA_HOME=c:\Program Files\Java\jdk11.xxx\`)

Also, be sure to check for the correct value in environment variable PATH. There you should have the same value as in JAVA_HOME followed by \bin. (i.e. `PATH=c:\Program Files\Java\jdk11.xxx\bin;...`)

If not installed by a user with administrator rights, add a suitable user with enough access rights.

If you are installing on a UNIX based OS

Recommended way to install and prepare java (under root user account) is this:

```
# Download java JDK!
wget -c --header "Cookie: oraclelicense=accept-securebackup-cookie" http://
download.oracle.com/otn-pub/java/jdk/xxxx/.../jdk-xxxx-linux-x64.tar.gz
# Create folder
mkdir /opt/java/jdkxxxx/
# Install java
tar zxvf jdk-xxxx-linux-x64.tar.gz -C /opt/java/jdkxxxx/
```

9. **If you are installing on a UNIX based OS**

   Set required system properties.

```
export PATH=/opt/java/jdkxxxx/bin:$PATH
export JAVA_HOME=/opt/java/jdkxxxx
```

10. Open folder `UA RELEASE PACKAGE>\platform\installer` and run `usu_analytics-full_installation-<version number>.jar` with administrative permissions and follow the instructions.

```
java -jar usu_analytics-full_installation-<version number>.jar
```

11. The install wizard opens.
    Read the text and click the **Next** button to continue
12. Select Full Installation.
    The update is described in the upgrade manual.
    Click the **Next** button to continue.
13. Select the installation path.
    - Standard on Windows: `D:\usu\analytics`
    - Standard on Linux/Unix: `/opt/usu/analytics`

    Answer the security message.
    Click the **Next** button to continue.

> By default, on Windows Server OS the system is run as a service under the Local Service account. This user has usually limited rights and has not sufficient permissions on the installation folder.
>
> The Local Service user must be added to the installation folder's access rights list so that it can access it. Also, the Local Service account must be granted the Modify permission for this folder.
>
> THIS CHANGE HAS TO BE DONE BEFORE THE SERVICE IS STARTED FOR THE FIRST TIME.

14. User for startup defines the user under whom USU Analytics will run on the system.
    - On Windows: Use 'Local Service'. For a technical user refer to the installation manual after the installation is completed for manual steps.
    - On Linux/Unix: Specify a user for the startup script.

    Click the **Next** button to continue.
15.
16. Select your default theme.

> The default theme defines the login screen. For example, if you have installed the system with the default theme USU Blue, the system always displays the login screen for the USU Blue theme.
>
> A user can choose a theme he likes, but scheduled reports will always use the default theme. For example, a user chooses USU Yellow in the front end, but USU Blue is the default. If the user schedules an analysis report to be sent via the Job Scheduler, the default theme will be used.
>
> The default theme can only be changed on the application server. To do this open the file **.\server\pentaho-server\pentaho-solutions\system\pentaho.xml** and change the value of the tag **default-theme**. This change requires a restart of the service to be activated.

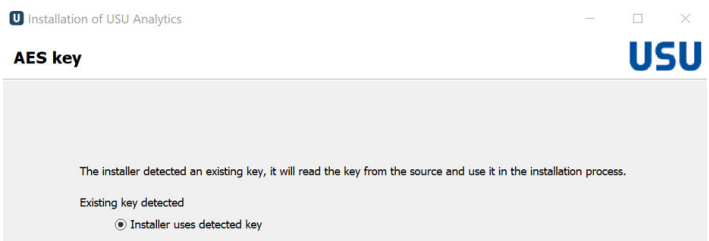
    Click the **Next** button to continue.
17. Define an AES key.

    To use AES encryption, define a master key. You either can let the system generate a valid key or you can provide a valid 256-bit AES key.

    We recommend letting the system generate the key for you.

    In case of an installation in a clustered environment, all nodes need to use the same master key. In that case provide the same valid 256-bit AES key manually in all node installations.

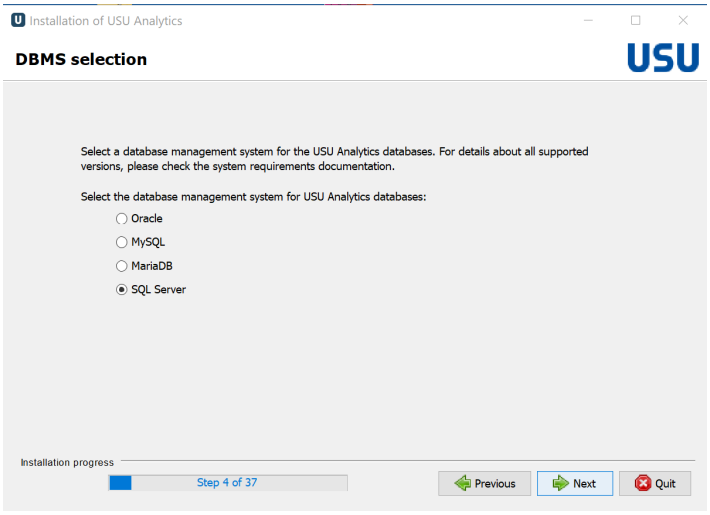
    When you install on a machine where you had installed USU Analytics the installer detects the master key and will reuse it.



Click the **Next** button to continue.

18. Select a DBMS for the USU Analytics databases.

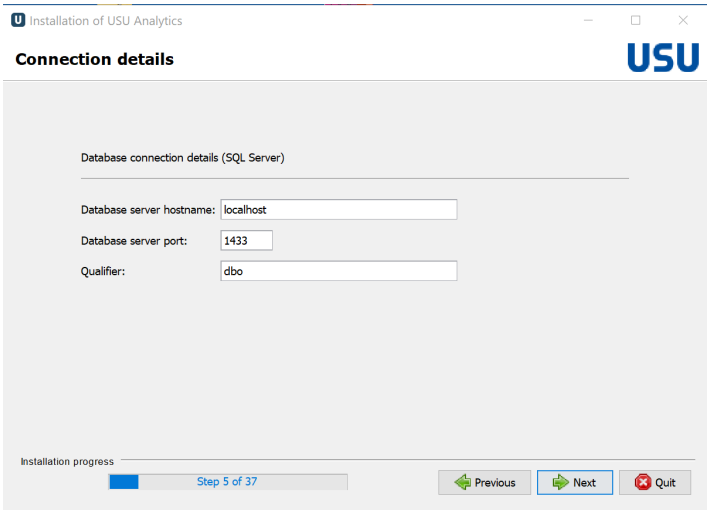
> For USU Analytics for Software Asset Management use MySQL.



Please check the supported platforms document before you proceed.

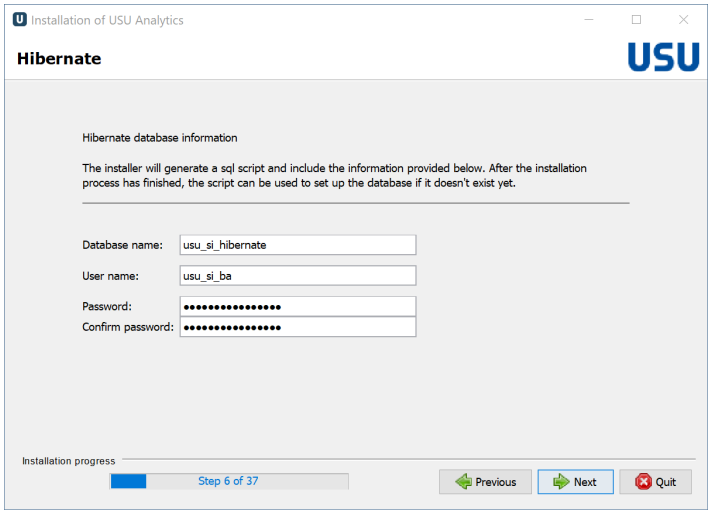
Click the **Next** button to continue.

19. Please enter the connection parameters to USU Analytics DBMS. The required parameters vary depending on the DBMS you have selected.



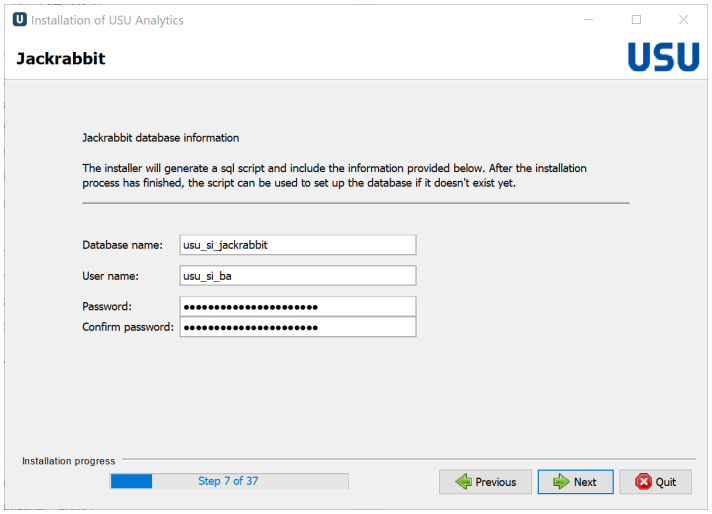
Click the **Next** button to continue.

20. Provide the database name and credentials for the USU Analytics Hibernate database.



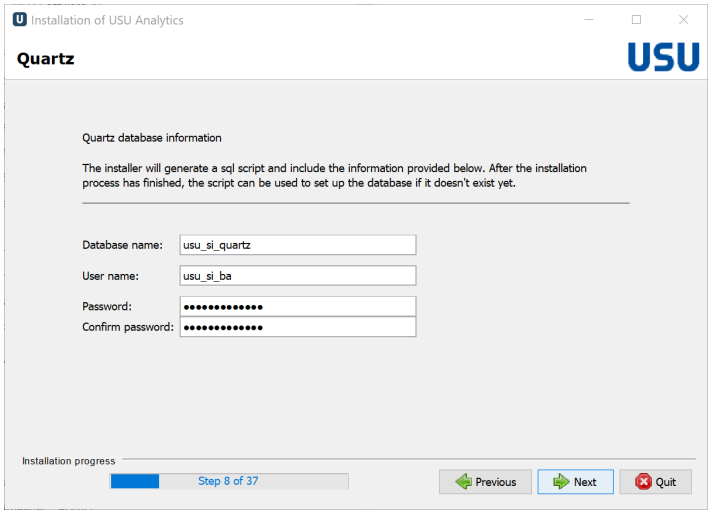
Click the **Next** button to continue.

21. Provide the database name and credentials for the USU Analytics Jackrabbit database.



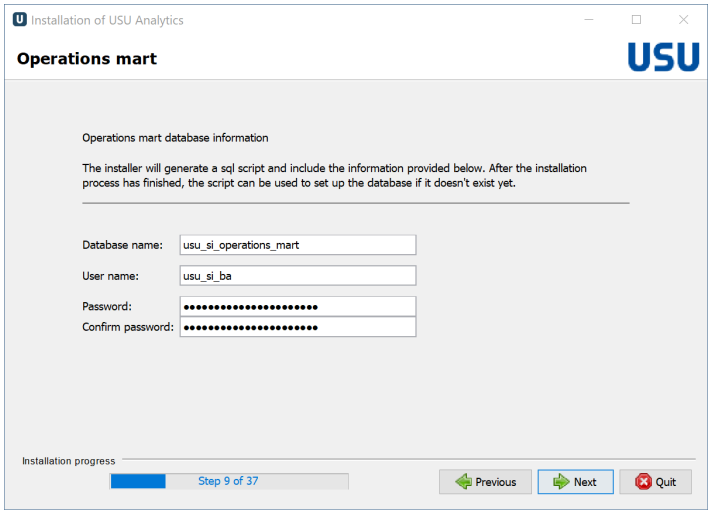
Click the **Next** button to continue.

22. Provide the database name and credentials for the USU Analytics Quartz database.



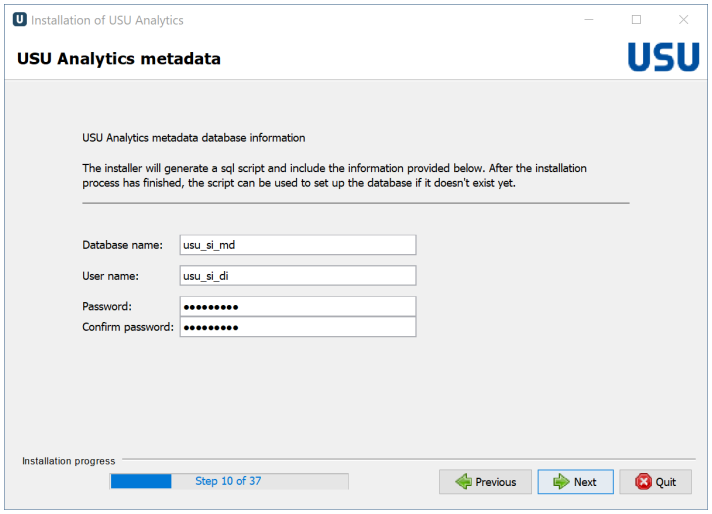
Click the **Next** button to continue.

23. Provide the database name and credentials for the USU Analytics Operations Mart database.



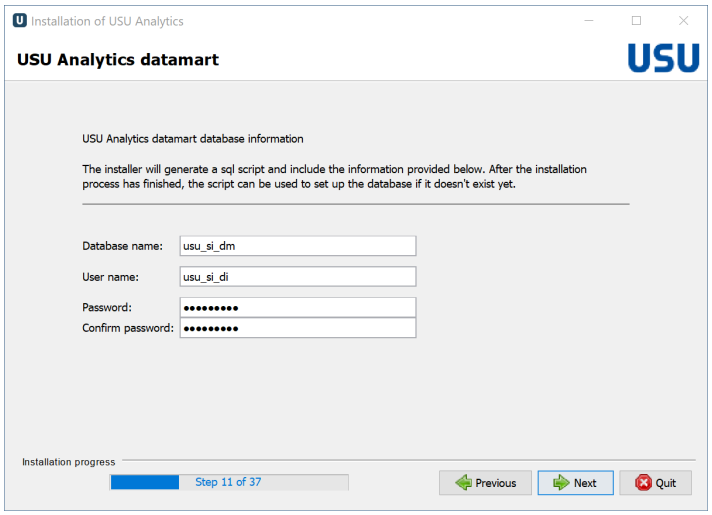
Click the **Next** button to continue.

24. Provide the database name and credentials for the USU Analytics Metadata database.



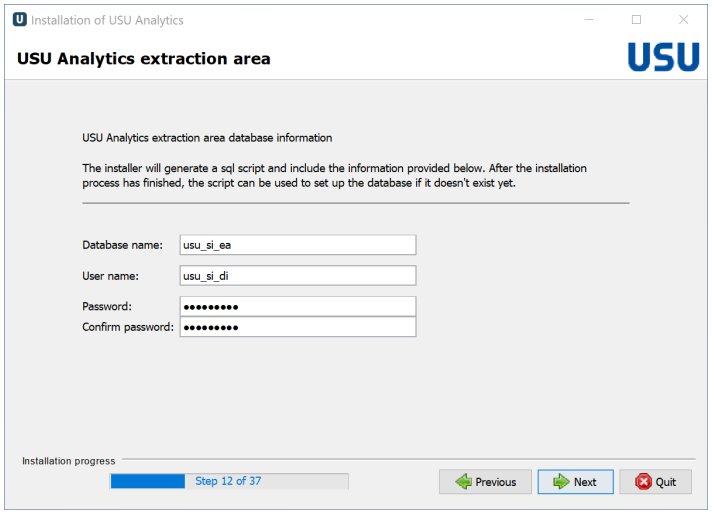
Click the **Next** button to continue.

25. Provide the database name and credentials for the USU Analytics Datamart database.



Click the **Next** button to continue.

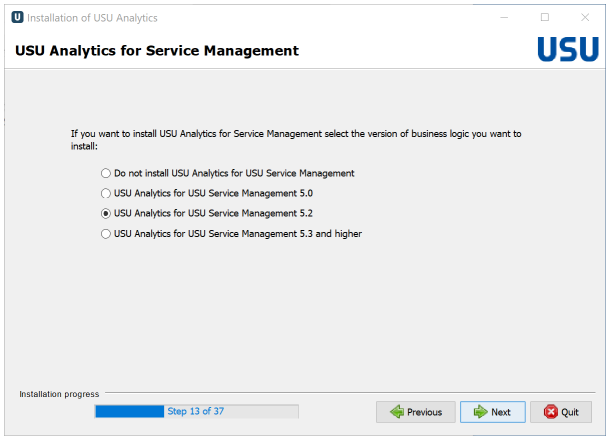
26. Provide the database name and credentials for the USU Analytics Extraction Area database.



Click the **Next** button to continue.

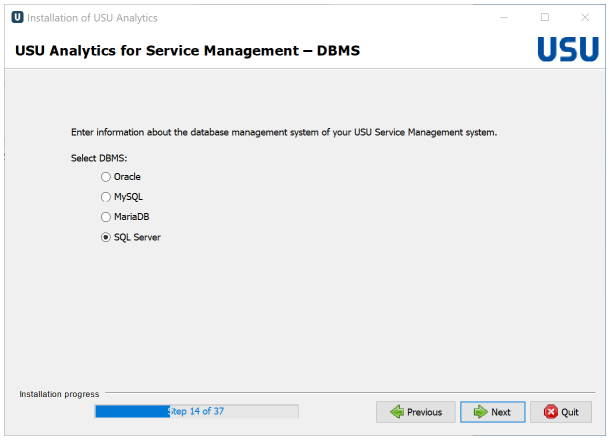
27. **USU Analytics for USU Service Management**

    If you want to install USU Analytics for USU Service Management , select the version of business logic you want to install.



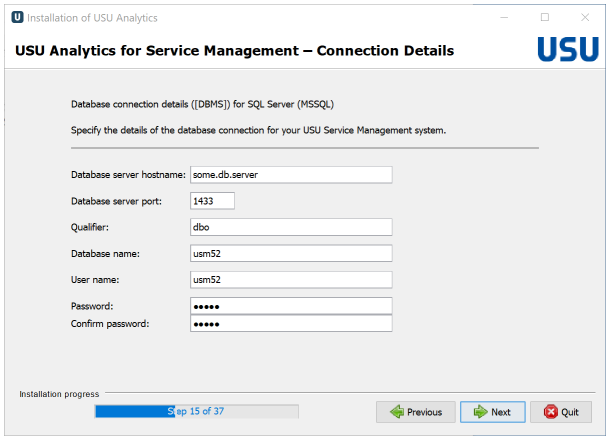
Click the **Next** button to continue.

If you have chosen to install USU Analytics for USU Service Management you are asked to select the DBMS of your Service Management source system.



Click the **Next** button to continue.

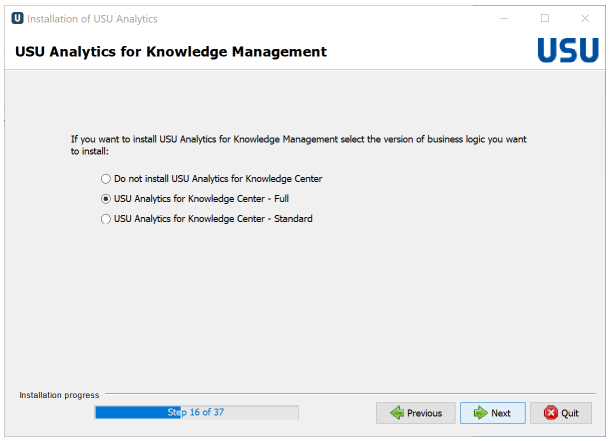
Now you enter the connection parameter and credentials to the USU Service Management source system database. The required parameters vary depending on the DBMS you use.



Click the **Next** button to continue.

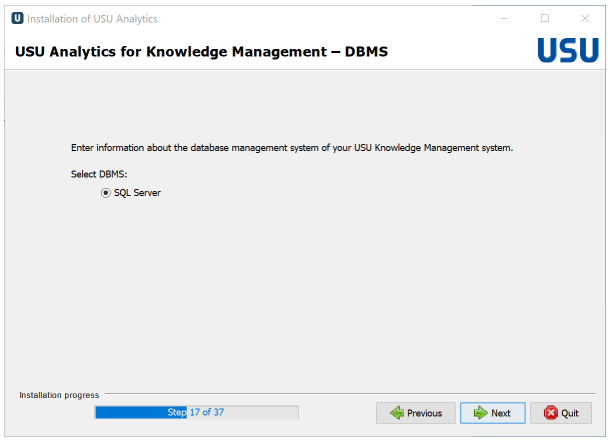
28. **USU Analytics for USU Knowledge Management**

    If you want to install USU Analytics for USU Knowledge Management select the version of business logic you want to install.



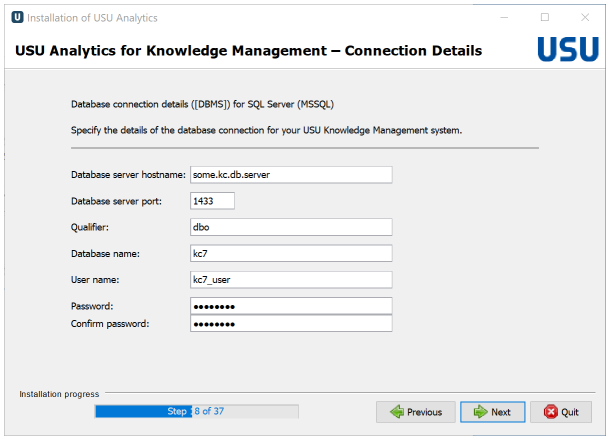
Click the **Next** button to continue.

If you have chosen to install USU Analytics for USU Knowledge Management you are asked to select the DBMS of your Knowledge Management source system.



Click the **Next** button to continue.

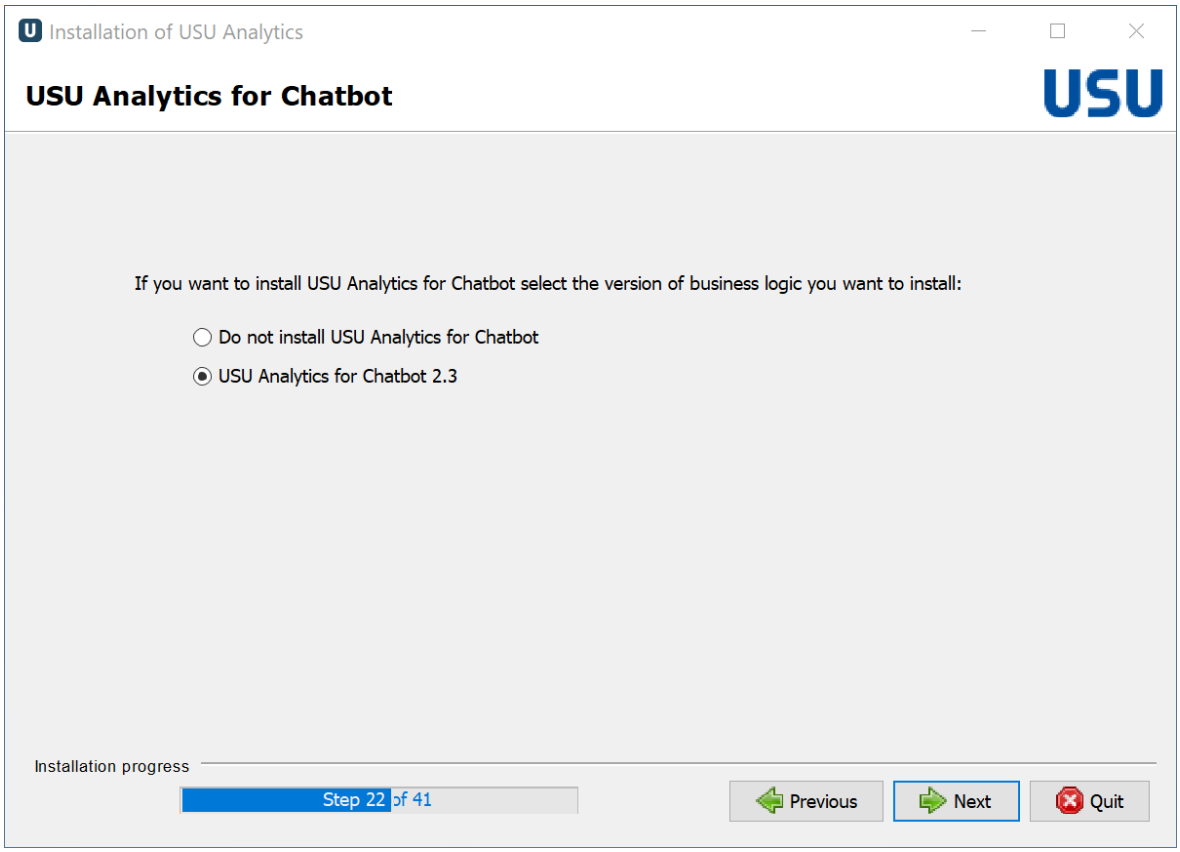
Now you enter the connection parameter and credentials to the USU Knowledge Management source system database. The required parameters vary depending on the DBMS you use.



Click the **Next** button to continue.

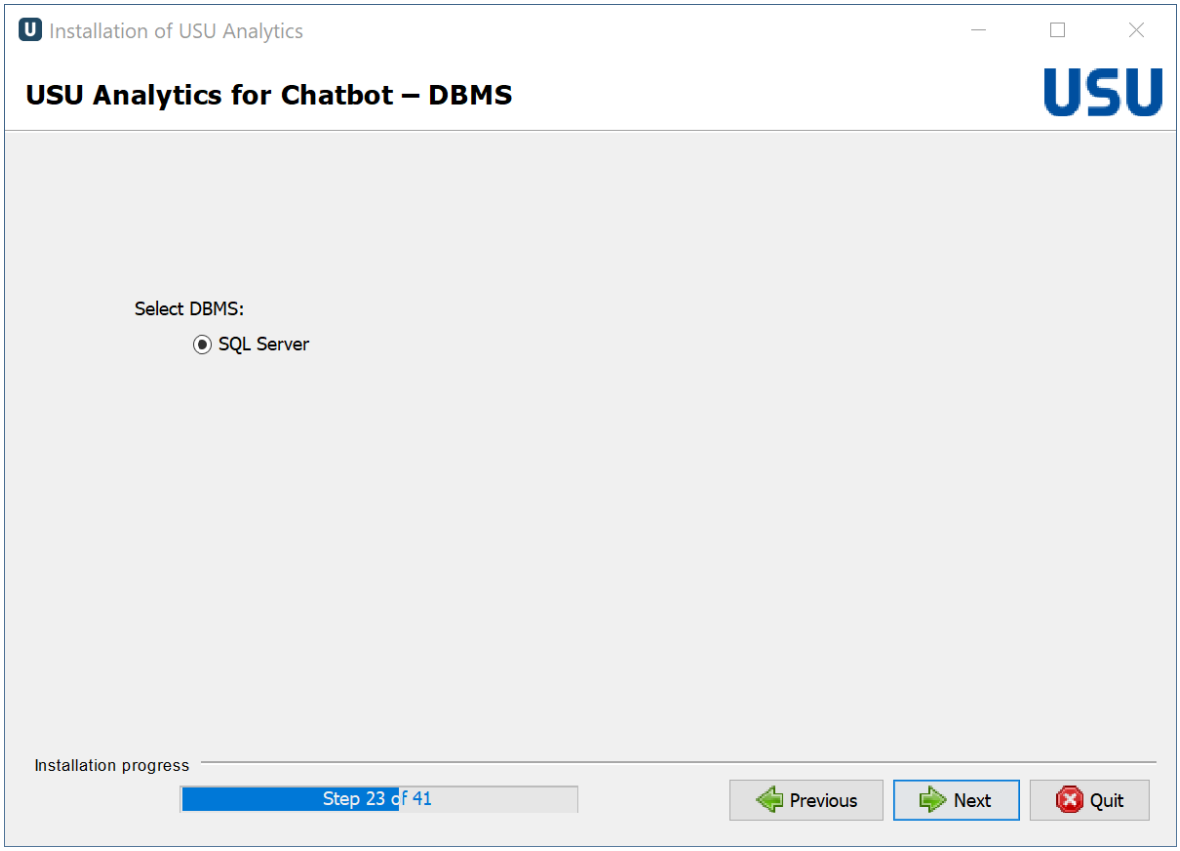
29. **USU Analytics for Chatbot**

    If you want to install USU Analytics for Chatbot select the verion of business logic you want to install.



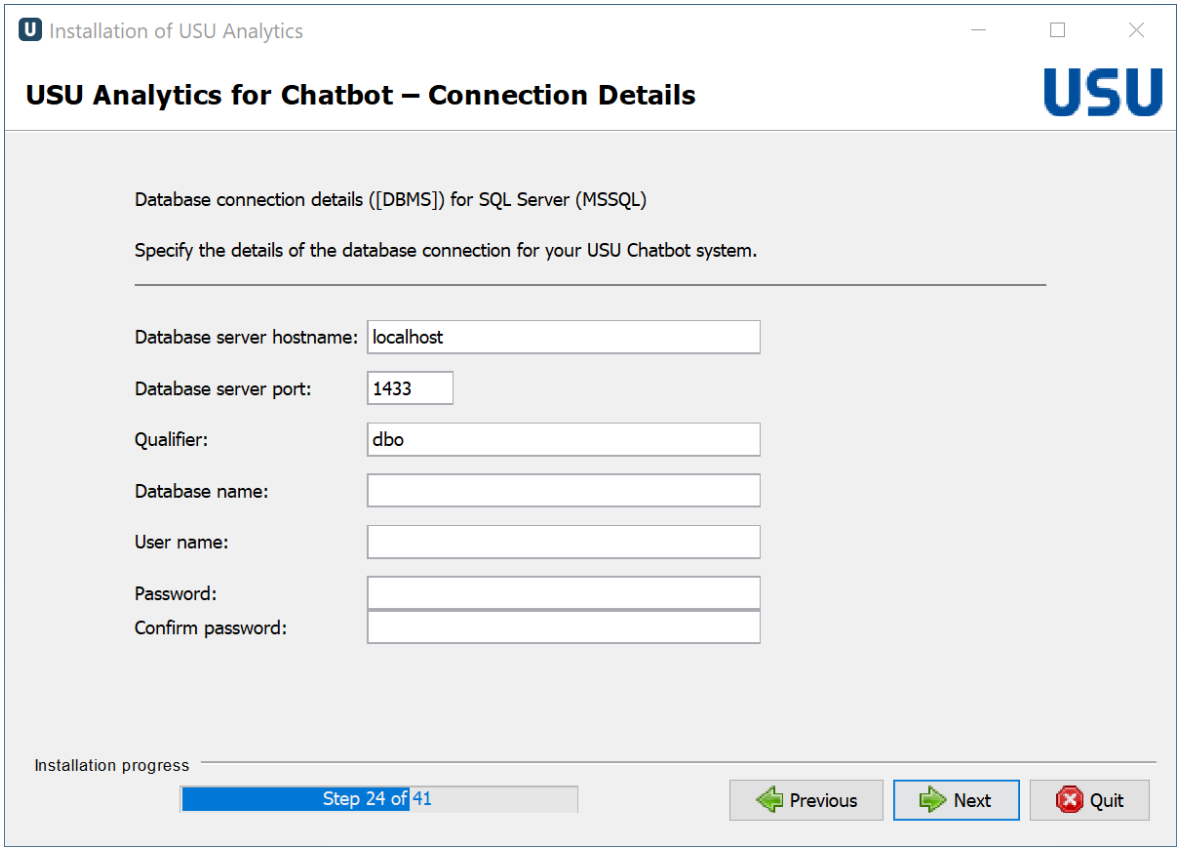
Click the **Next** button to continue.

Select the DBMS.



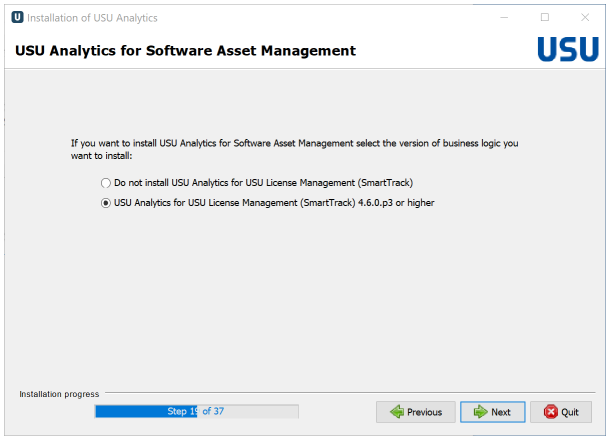
Click the **Next** button to continue.

Now you enter the connection parameter and credentials to the Chatbot source system database. The required parameters vary depending on the DBMS you use.



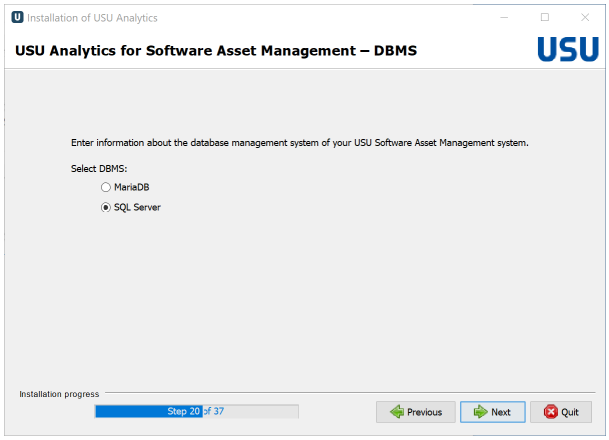
30. **USU Analytics for USU Software Asset Management**

    If you want to install USU Analytics for USU Software Asset Management select the version of business logic you want to install.



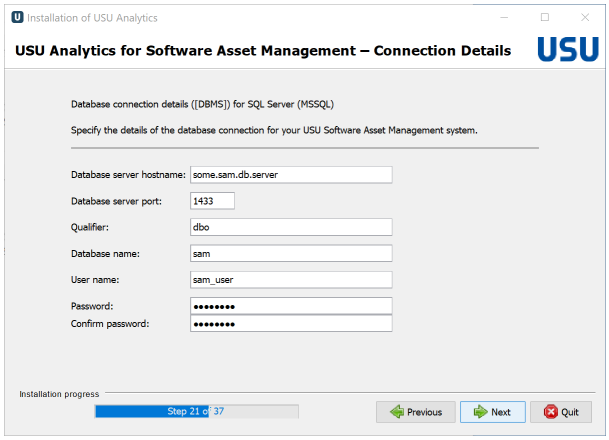
    Click the **Next** button to continue.

    If you have chosen to install USU Analytics for USU Software Asset Management you are asked to select the DBMS of your Software Asset Management source system.



Click the **Next** button to continue.

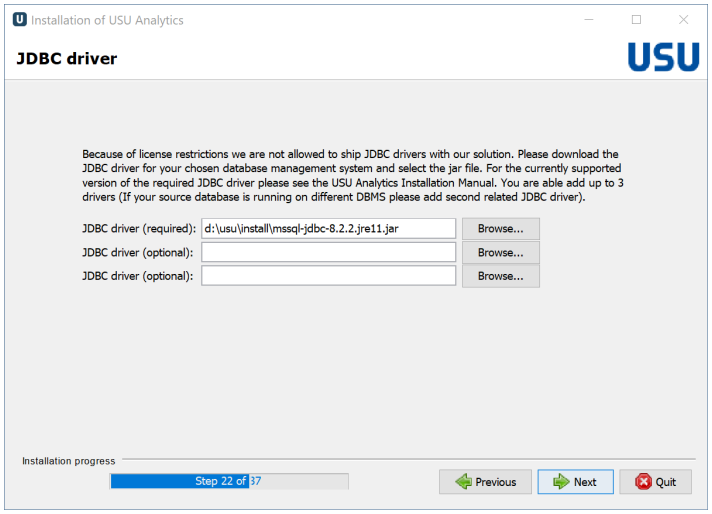
Now you enter the connection parameter and credentials to the USU Software Asset Management source system database. The required parameters vary depending on the DBMS you use.



Click the **Next** button to continue.

31. Browse for the JDBC driver jar files you want to use. For example, upload the Oracle and the MS SQL Server driver, when your source system runs on Oracle and our USU Analytics database will be installed on MS SQL Server.

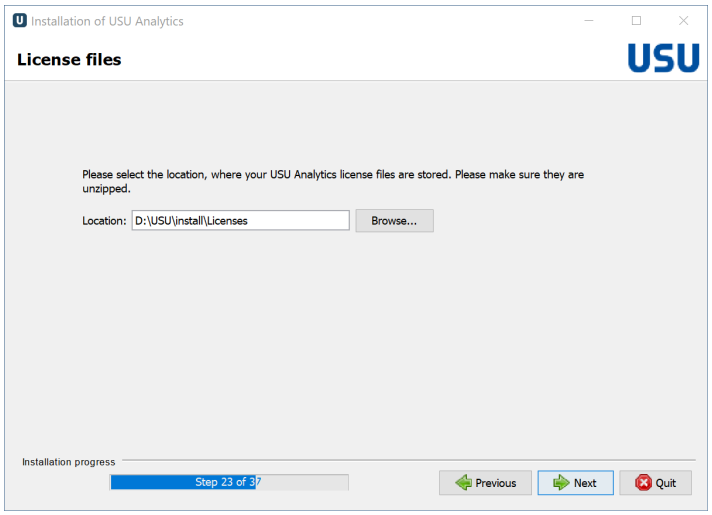
> Oracle is not supported for USU Analytics for Software Asset Management .



System repository database	Source system database	Upload supported JDBC driver for
Oracle	Oracle	Oracle
Oracle	MariaDB	Oracle
Oracle	MS SQL Server	Oracle, MS SQL Server
MS SQL Server	Oracle	MS SQL Server, Oracle
MS SQL Server	MariaDB	MS SQL Server
MS SQL Server	MS SQL Server	MS SQL Server
MySQL	Oracle	Oracle, MySQL
MySQL	MariaDB	MySQL
MySQL	MS SQL Server	MS SQL Server, MySQL
MariaDB	Oracle	Oracle, MySQL
MariaDB	MariaDB	MySQL
MariaDB	MS SQL Server	MS SQL Server, MySQL
MariaDB CS	Oracle	Oracle, MySQL
MariaDB CS	MariaDB	MySQL
MariaDB CS	MS SQL Server	MS SQL Server, MySQL

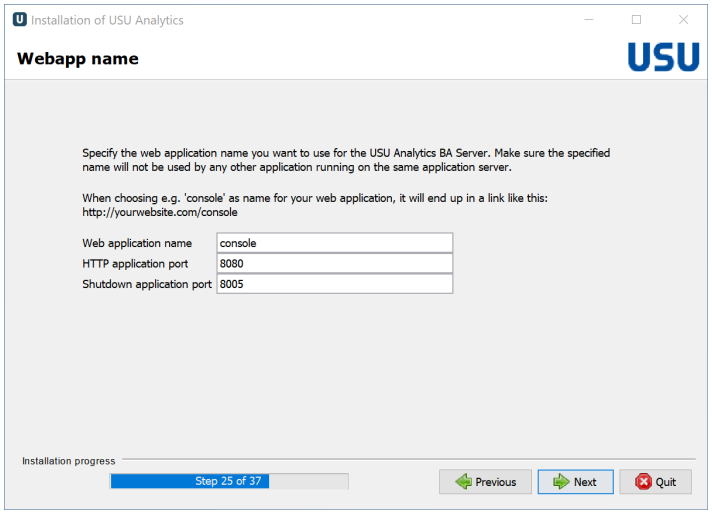
Click the **Next** button to continue.

32. Please select the location, where your USU Analytics license files are located. Make sure they are not zipped.



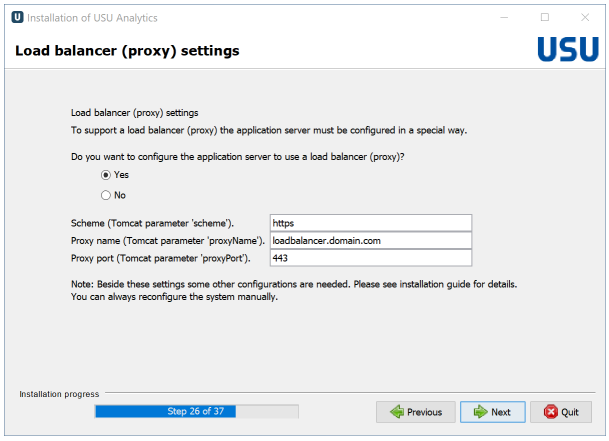
Click the **Next** button to continue.

33. Specify the web application name and ports you want to use for the USU Analytics BA server. Make sure the specified name and ports are not used by any other application running on the same application server. We recommend using the standard name console.



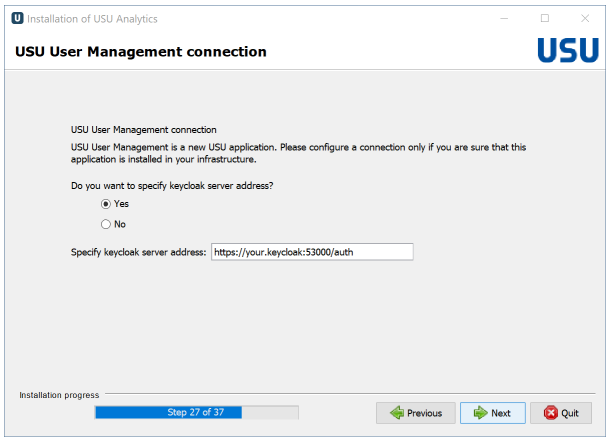
Click the **Next** button to continue.

34. Configure load balancer (proxy) settings if needed. Beside these settings some other configurations are needed. For more information see Setup a Load Balancer (Proxy) (page 34)



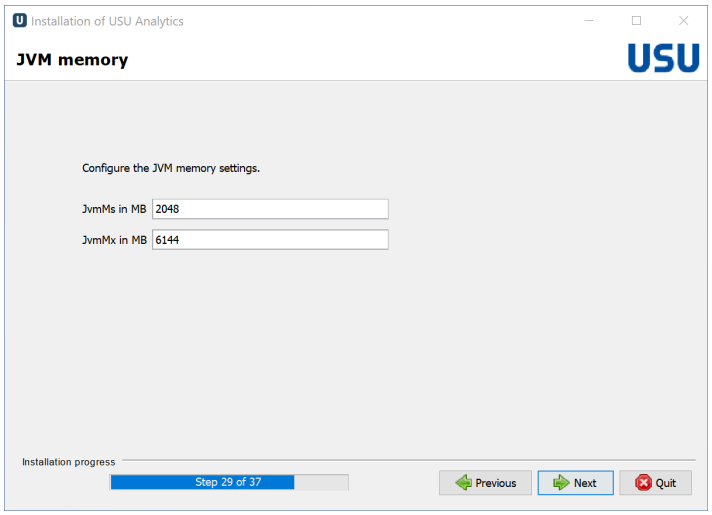
35. Configure the USU User Management connection

    USU User Management is a new USU application. Configure a connection only, if you are sure that this application is installed in your infrastructure.



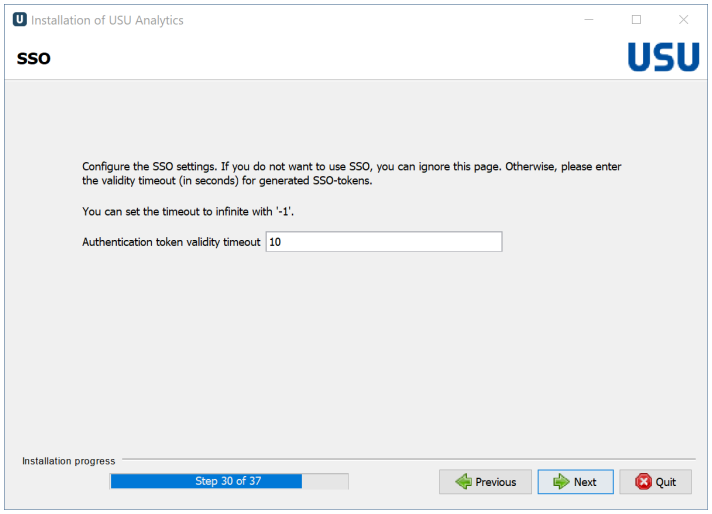
Click the **Next** button to continue.

36. Configure the Java Virtual Machine memory settings.
    Set them according to your server infrastructure.



    Click the **Next** button to continue.

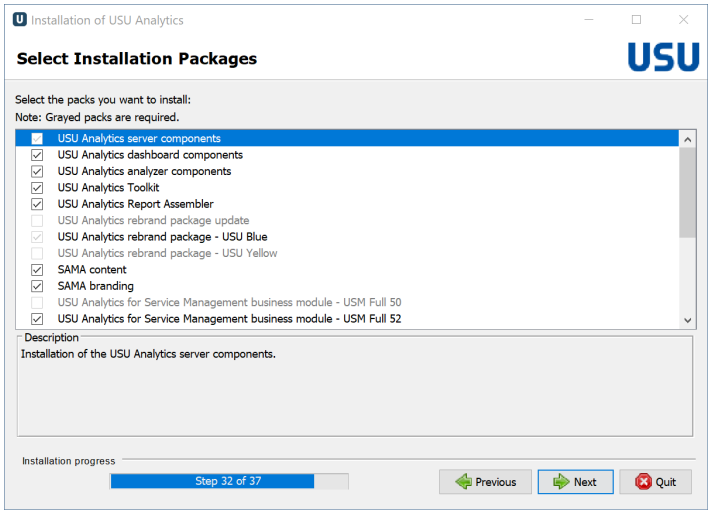
37. Configure the SSO settings.
    If you don't want to use SSO, you can ignore this page.
    Otherwise enter the validity timeout for SSO-tokens in seconds.



    Click the **Next** button to continue.

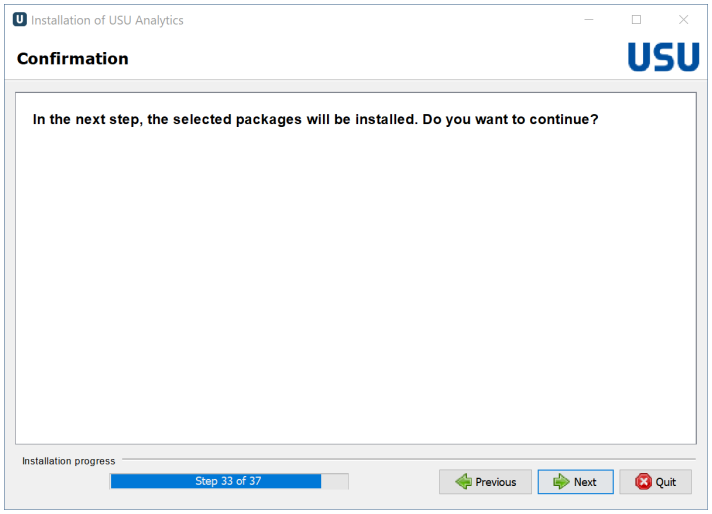
38. Select the security provider.
    Memory is recommended for SSO with USU Service Management or USU License Management.
    Provide password for superuser admin and confirm the password.
    Click the **Next** button to continue.

39. Based on your previous selection, the packages to be installed are compiled.
    Optional packages can be deselected or added here.



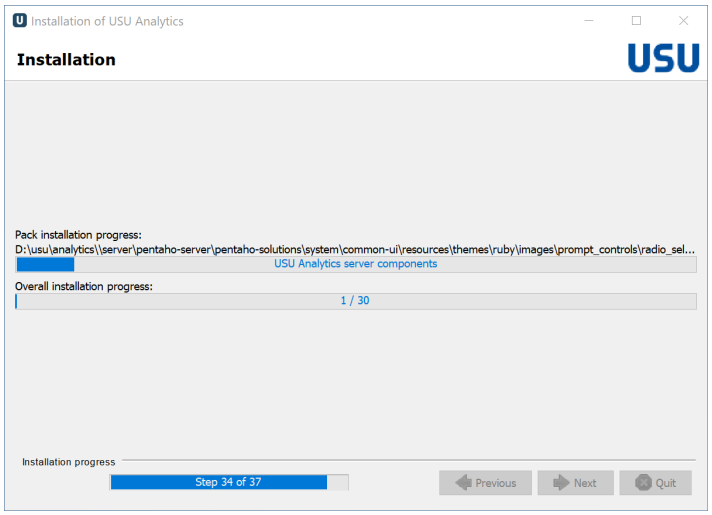
Click the **Next** button to continue.

40. Click the **Next** button to continue.

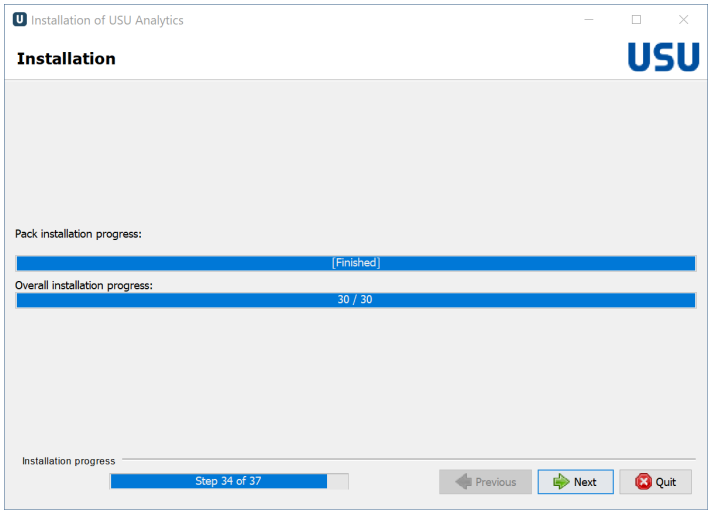


The installation process will start now.

41. Installation is in progress. This may take a few minutes.

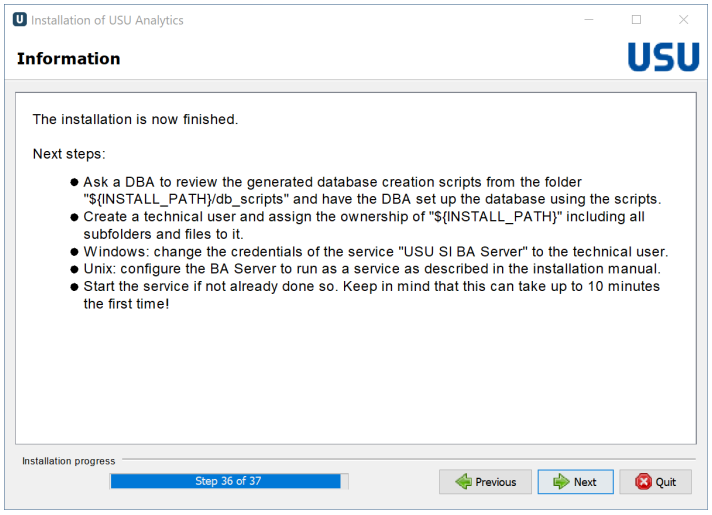


42. When you can see the **Next** button, the installer has finished the copy routine and prepared the service.

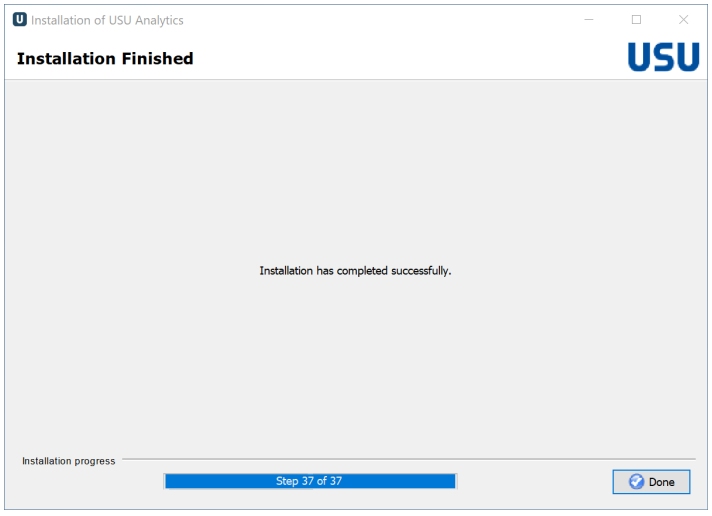


Click the **Next** button to continue.

43. Click the **Next** button to continue.



44. Click the **Done** button to finish the installer.



45. **If you are installing on a UNIX based OS**

Execute this step if you want to install and run as a Service with Systemd script

You can find an example of a startup script, which can be used to start BA Server as a service under newer Linux distributions, here: `%USU_HOME%/analytics/scripts/linux_startup/systemd/vm-analytics.service`

It's an example of deploying startup script on Linux distributions has "systemd" functionality and configuring BA Server to start as a service.

Note that outlined example expects the *usu_si_ba* user account to exist and once implemented, the BA Server will start under this user account. In case a different user account is supposed to be used for starting BA Server, the example needs to be adjusted.

To deploy the startup script, run the following sets of commands under root user account (if required, change the path where the customization package is deployed):

```
cp %USU_HOME%/analytics/scripts/linux_startup/systemd/vm-analytics.service /etc/systemd/
system
cd /etc/systemd/system
chown root:root vm-analytics.service
chmod 0750 vm-analytics.service
systemctl daemon-reload
systemctl enable vm-analytics
```

It is possible to use following commands:

```
systemctl start vm-analytics
systemctl stop vm-analytics
systemctl restart vm-analytics
```

46. **If you are installing on a UNIX based OS**

Execute this step if you want to install and run as a Service with Init.d script

You can find an example of a startup script, which can be used to start BA Server as a service under Linux, here: `%USU_HOME%/analytics/scripts/linux_startup/init.d/vm-analytics`

It's an example of deploying startup script on SUSE Linux Enterprise Server and configuring BA Server to start as a service. For different Linux distribution, the script and startup sequence are required to be enhanced accordingly.

Note that outlined example expects the *usu_si_ba* user account to exist and once implemented, the BA Server will start under this user account. In case a different user account is supposed to be used for starting BA Server, the example needs to be adjusted.

To deploy the startup script, run the following sets of commands under root user account (if required, change the path where the customization package is deployed):

```
cp %USU_HOME%/analytics/scripts/linux_startup/init.d/vm-analytics /etc/init.d/
cd /etc/init.d/
chown root:root vm-analytics
chmod 0750 vm-analytics
chkconfig --add vm-analytics
```

It is possible to use following commands:

```
service start vm-analytics
service stop vm-analytics
service restart vm-analytics
```

2.3.2. Prepare Databases for USU Analytics

The installer generates SQL-scripts for the creation of all databases USU Analytics needs and saves them in folder `%VMA_HOME%/db_scripts`. The scripts reflect the DBMS that was chosen as well as any database-names and credentials specified during the installation.

It is recommended to have a DBA review the scripts before execution to make sure everything is in order and complies with company standards.

The DB scripts can also be provided before installation. Passwords and database names must then be changed manually and entered consistently in the installer.

Check and make sure that the database uses UTF encoding (for example: MySQL needs to use “utf8_general_ci“)

Execute for UA Basic and UA Full installation.

1. Open `folder %VMA_HOME%/db_scripts`.
2. Execute following scripts on the database server that has been provided or have a DBA execute the scripts for you in case you don’t have enough permissions.
   - `01-create_db_schemas_for_ba_server.sql`
   - `02-create_db_schemas_for_md_dm_ea.sql`
3. Depending on your database system the following users/schemas/databases should be available afterwards:
   - Users: `usu_si_ba, usu_si_di`
   - Databases: `usu_si_jackrabbit, usu_si_quartz, usu_si_hibernate, usu_si_operations_mart, usu_si_md, usu_si_ea and usu_si_dm`

2.3.3. Update USU Service Management database

The installer generates SQL-scripts of views to be deployed in USU Service Management. USU Analytics needs these views in USU Service Management to access USU Service Management data directly for RTCs (UA Basic).

Execute for UA Basic and UA Full installation.

1. Open folder `%VMA_HOME%/init_import/business_content/modules/vm_itil_basic_<VERSION>/sql/<DBMS>`.

   Choose the folder for the DBMS your USU Service Management is using.
2. Execute all scripts on the USU Service Management database server or have a DBA execute the scripts for you in case you don’t have enough permissions.

   The scripts will replace existing database views or create new ones. The views will match the Mondrian schemas we deployed before together with the installation of USU Analytics .

2.3.4. Start the BA server

By default, on Windows Server OS the system is run as a service under the Local Service account. This user has usually limited rights and has not sufficient permissions on the installation folder.

The Local Service user must be added to the installation folder's access rights list so that it can access it. Also, the Local Service account must be granted the Modify permission for this folder.

THIS CHANGE HAS TO BE DONE BEFORE THE SERVICE IS STARTED FOR THE FIRST TIME.

Execute for UA Basic and UA Full installation.

1. Start BA Server by starting the service `USU SI BA Server`.

It can take several minutes before the BA-Server is up and running despite the service status switching to “running” almost immediately.

2. You can check the initialization progress in logs.
   The logs you find in folder `%VMA_BASERVER%\tomcat\logs`.

2.3.5. Configure Authentication

Execute for Basic and Full installation.

Depending on the authentication method different configuration steps are necessary. See the administration manual for provided possibilities.

Recommended setup for USU Analytics is to use credentials.[1] The following steps describe how to **Configure Authentication.**

1. Log in to user console as administrator.
2. In user console Home screen click on **Toolkit**. The Toolkit application opens.
3. Click Maintenance / System Maintenance / **System Configuration**.
   Navigate to section **Expert Parameters**.
   Set checkbox **SM authentication** to true.
4. Open Administration and configure the rights of the groups in Administration / User & Roles / **Manage Roles**.

	ANALYTICS ADMIN	ANALYTICS POWERUSER	ANALYTICS READER
Administer Security	☑	☐	☐
Schedule Content	☑	☑	☐
Read Content	☑	☑	☑

[1]The recommended setup for USU Analytics for Software Asset Management is to use Single-Sign-On

	ANALYTICS ADMIN	ANALYTICS POWERUSER	ANALYTICS READER
Publish Content	☑	☑	☐
Create Content	☑	☑	☐
Execute	☑	☑	☐
Manage Data Sources	☑	☐	☐

These settings are only recommendations. You can set the authorizations according to your requirements.

We are sorry, but for technical reasons it is not possible to set these defaults during the installation routine.

5. Log on to and assign one of the groups listed above to the corresponding users.

2.3.6. Smoke Test

It is recommended to test that:

Test	Expected result
Login to user console is working	Access the USU Analytics User Console. The following credential should be available, and you should use it for the test: admin/<password entered during installation> (Administrator). The URL of the User Console is `http://localhost:<port>/<webappname>/Login` Replace `<port>` by the port you have chosen during installation – the default is `8080`. Replace `<webappname>` by the name chosen during installation. The default is `console`. Expected result is that you are logged in and you see the user console Home screen.
Data source usu_si_dm is configured correctly	In user console Home screen click on **Manage Data Sources**. A modal form opens. Scroll down until you find the data source **usu_si_dm**. Mark this line and select the **Edit...** option from the gear wheel menu. In the new form click on **Test** button. Test result should be **Successful**.
Data source vm is configured correctly	In user console Home screen click on **Manage Data Sources**. A modal form opens. Scroll down until you find the data source **vm**. Mark this line and select the **Edit...** option from the gear wheel menu. In the new form click on **Test** button. Test result should be **Successful**.
Data source to source system is configured correctly	In user console Home screen click on **Toolkit**. The Toolkit application opens. Click Maintenance / System Maintenance / **Data Source Configuration**. After the new form opened, click the **Config Connection** button of the Data Sources Configuration you want to test, e.g. USU Service Management Click **Test connection** button – It will change color to green, if the test was successful. Otherwise an error message will be displayed.
USU Service Management Authentication is working	Log in with a user, you configured in chapter Configure Authentication (page 32). Expected result is that you are logged in and you see the user console Home screen.

Test	Expected result
A custom directory can be created in the Public folder on Browse Files and that the access rights can be changed on that directory (only possible with Admin user)	Please refer to the user manual for details.

2.4. Data Load

A data load is only necessary, if you are using UA Full version.

Even for the UA full version, it is not absolutely necessary to start a data load at this point, but this step is recommended in order to test the data load functionality in principle.

2.4.1. Start install workflow

1. Execute for UA Full installation.
2. Open the Toolkit from the home screen.
3. Open the Data Load Organizer (DLO).
4. Start the system **install** workflow with the **play** button.

2.4.2. Start Data Load

1. Execute for Full installation.
2. Open the Toolkit from the home screen.
3. Start the relevant Data Load workflow, eg. USU Service Management complete load with the **play** button.

2.4.3. Smoke Test

It is recommended to test that:

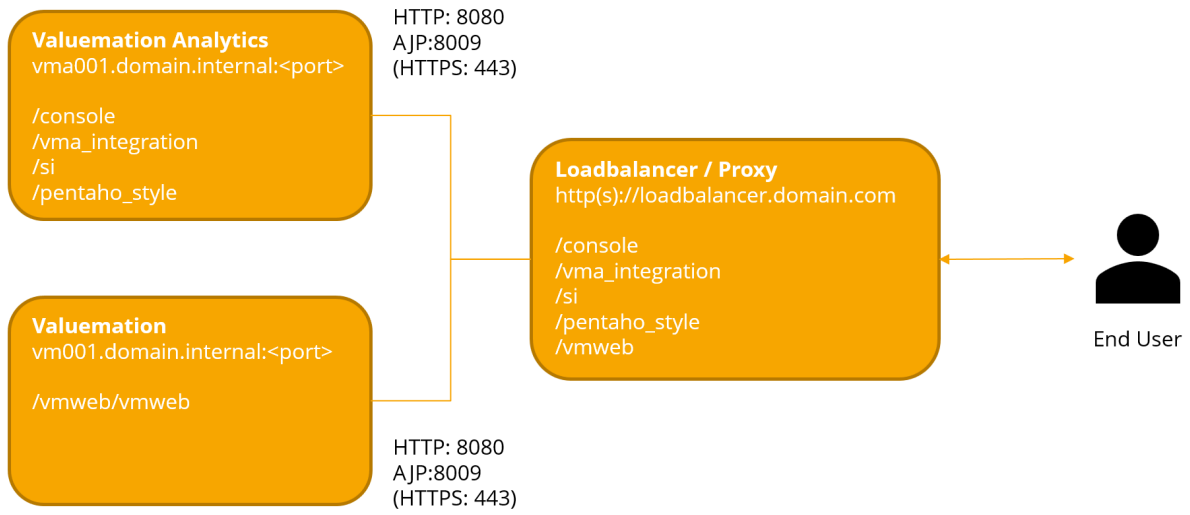
Test	Expected result
An analyzer report can be created, saved and scheduled to be executed	Please refer to the user manual for details. Please note, that analysis you create at this point could be empty depending on the data in source system or USU Analytics database.
A dashboard can be opened	Please refer to the user manual for details. Please note, that dashboards you open at this point could be empty depending on the data in source system or USU Analytics database.

2.5. Setup a Load Balancer (Proxy)

This is not required for USU Analytics for Software Asset Management installations.

A load balancer (proxy) is not necessary if you plan to run the system as a non-integrated application. But in customer environment it is sometimes wanted to access UA user console via a load balancer (proxy).

> ⚠ However, if you want to integrate UA and USU Service Management, a load balancer (proxy) is mandatory. UA and USU Service Management must be accessible via the same URL.



2.5.1. Load Balancer (Proxy) Configuration

Setup a reverse proxy or load balancer.

These rules must be applied. They are just an example for an Apache Proxy using HTTP, living on the same Linux server as the UA application server - Ports and server names must be changed.

```
ProxyPass /console http://0.0.0.0:8080/console
ProxyPassReverse /console http://0.0.0.0:8080/console

ProxyPass /pentaho-style http://0.0.0.0:8080/pentaho-style
ProxyPassReverse /pentaho-style http://0.0.0.0:8080/pentaho-style

ProxyPass /si http://0.0.0.0:8080/si
ProxyPassReverse /si http://0.0.0.0:8080/si

ProxyPass /vma_integration http://0.0.0.0:8080/vma_integration
ProxyPassReverse /vma_integration http://0.0.0.0:8080/vma_integration
```

If you are using a Load Balancer configure it correspondingly.

2.5.2. Changes/Checks in USU Service Management

This is not required for USU Analytics for Software Asset Management installations.

1. Configure the following USU Service Management main parameters according to your authentication method:
   - `Analytics:si_url: url` to USU Analytics server depending on authentication method
     - SM Authentication (recommended for usage with USU Analytics Basic Version): `/si/launch_vm.jsp?loc=`
     - SSO: `/si/launch_sso.jsp?loc=`

   Path of main parameter must be **ValuemationAnalytics**. In previous USU Service Management Versions it was **ServiceIntelligence**, which does no longer work.

2. Check global parameter vmwebroot. USU Service Management and USU Analytics must run on the same URL. In this case it is the URL of the load balancer (proxy).

2.5.3. Changes/Checks in USU Analytics

If you have done the configuration during the installation process, the following manual steps are not necessary.

To support a load balancer (proxy) the UA server must be configured in a special way.

To do this follow these steps:

1. Stop BA Server
2. Open file `%VMA_BASERVER%\tomcat\conf\server.xml`
3. Change the following information on your connector from:

```
<Connector URIEncoding="UTF-8" port="8080" protocol="HTTP/1.1"
connectionTimeout="20000"
redirectPort="8443" />
```

to

```
<Connector URIEncoding="UTF-8" port="8080" protocol="HTTP/1.1"
scheme="https" proxyName="loadbalancer.domain.com" proxyPort="443"
connectionTimeout="20000"
redirectPort="8443" />
```

Make sure to change the proxyName URL to match the one of your load balancers. E.g. if the URL you want to use to approach UA console is https://loadbalancer.domain.com/console your proxyName is loadbalancer.domain.com.

Depending on your environment; if you have configured to run via AJP you may have to change the protocol as follows:

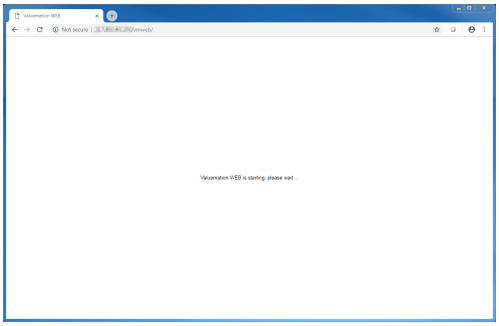
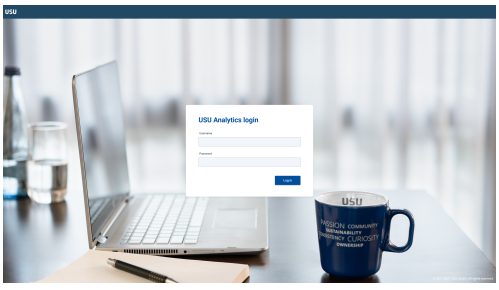
```
<Connector URIEncoding="UTF-8" port="8009" protocol="org.apache.coyote.ajp.AjpProtocol"
redirectPort="443"
scheme="https" secure="true"
proxyName="loadbalancer.domain.com" proxyPort="443"
connectionTimeout="20000" />
```

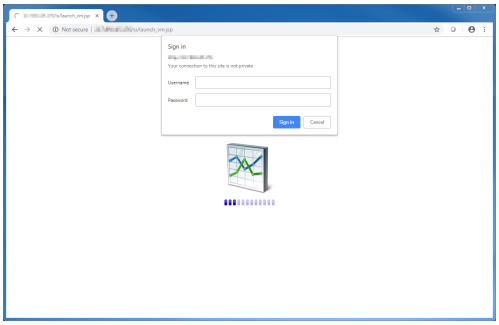
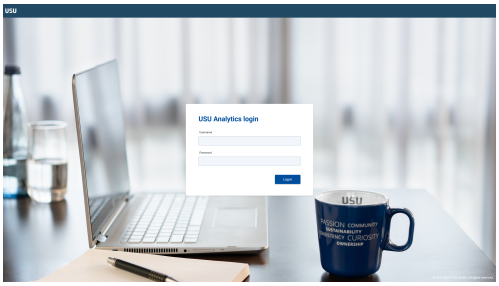
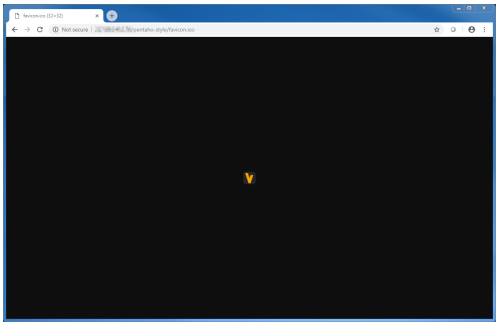
4. Delete the content of these folders:
   - %VMA_BASERVER%\tomcat\temp
   - %VMA_BASERVER%\tomcat\work
5. Restart your BA Server.

2.5.4. Smoke Test

This is not required for USU Analytics for Software Asset Management installations.

To check the configuration of the load balancer (proxy) and the UA server, you should perform the following tests:

Test	Expected result
Test access to USU Service Management using the load balancer (proxy) URL: <LOAD BALANCER URL>/vmweb	The login screen opens in browser. 
Test access to UA user console using the load balancer (proxy) URL: <LOAD BALANCER URL>/console	The login screen opens in browser. 

Test	Expected result
Test access to UA SI web application using the load balancer (proxy) URL: <LOAD BALANCER URL>/si/launch_vm.jsp	You see a spinner and a sign in modal form. 
Test access to UA SI web application using the load balancer (proxy) URL: <LOAD BALANCER URL>/vma_integration/launch.jsp	The login screen opens in browser. 
Test access to UA pentaho-style web application using the load balancer (proxy) URL: <LOAD BALANCER URL>/pentaho-style/favicon.ico	The USU Service Management icon opens in browser. 
Open USU Service Management. Choose a UA InfoObject and open it.	The InfoObject opens in USU Service Management and the dashboard or analysis is displayed. Please note, that analysis or dashboards you open at this point could be empty depending on the data in source system or database.

2.6. System Configuration

Execute for UA Basic and UA Full installation.

Depending on your needs you can change the variables in System Configuration. To do this open Toolkit / System Configuration. Check administration manual for provided possibilities.

2.7. Time for coffee!

Congratulations! The installation and initial setup of USU Analytics is now complete, and you should have a running system. Data will be visible in the system after the data was loaded for the first time. This depends on your configured schedule.

Further configuration beyond the scope of this manual can now be performed as needed. More information about available options can be found in the administration manual or requested from the USU Analytics support team.

2.8. Connection to other USU solutions

2.8.1. IA44 Connecting USU Service Management to USU Analytics (IA44)

Follow these steps to integrate USU Service Management and USU Analytics and to use the USU Service Management DB as data source.

1. Execute for UA Basic and UA Full installation.
2. Define a user for the USU Service Management database, which should be used in USU Analytics to read data from USU Service Management.

   This user needs:
   - Read rights to the information schema of USU Service Management database
   - Read rights to all tables of USU Service Management database
   - Write rights to table `AMT_VMA_CONTENT`
3. Schedule regular update of ticket report table (AMS_TICKET_REPORT) in USU Service Management.

   See USU Service Management hints how to activate regular update of ticket report table in your USU Service Management documentation.
4. Configure the following USU Service Management main parameters according to your authentication method if you are using USU Service Management 5.0 or lower:

   `Analytics:si_url: url` to USU Analytics server depending on authentication method
   - USU Service Management Authentication: /si/launch_vm.jsp?loc= (recommended for usage with USU Analytics Basic Version)
   - For USU Service Management 5.1 and higher and the new USU Service Management InfoObjects this is not needed.

     `SSO: /si/launch_sso.jsp?loc=`

   > In previous USU Service Management versions the path of main parameter was **ServiceIntelligence**. The system will search first for **ValuemationAnalytics** and then for **ServiceIntelligence**.

5. > If you are using a USU Service Management version prior to USU Service Management 5.2 please follow these steps. From USU Service Management 5.2 on the database objects created in the following steps are part of the standard deployment.

   Open `Valuemation_<Version>_yetty.zip/scripts/ADDON/Dashboards/VMA/` and execute the following scripts in the USU Service Management database:
   - Following SQL files in folder <database_version>/, i.e.
     - `<database_version>/si-000-dim_datetime.sql`
     - `<database_version>/si-000-unassigned.sql`

- `001-dim_date_data_de.sql`
- `001-dim_date_data_en.sql`
- `001-dim_date_data_fr.sql`
- `001-dim_date_data_it.sql`
- `001-dim_time_data.sql`

2.8.2. IA49 Connecting USU License Management to USU Analytics (IA49)

Follow these steps to integrate USU License Management and USU Analytics and to use the USU License Management DB as data source.

1	Create SOAP user for (page 40) A SOAP user in is required to connect to .
2	Create DB user for (page 41) This user will be used by SAMA to read data from LIMA
3	Configure connection in configuration file (page 41)
4	Configure connection in (page 41)
5	Configure Job Scheduler in (page 41)
6	Configure Xtract in (page 42)
7	Initialize database in (page 42)

Please adjust the firewall ports to permit communication between SOAP and database

- http: Port 80
- https: Port 443
- SQL DB: Port 1433
- MariaDB: Port 3306

Create SOAP user for

1. Check if a SOAP interface user is existing in . Open **Administration > Configuration > Interface Users** in . If no SOAP interface user exists, create it.

   The creation of a SOAP interface user is described in chapter "Prerequisites to use SOAP / Web-Services" of the administration manual

2. For USU License Management with Windows Authentication, open the IIS authentication settings and set:

   Windows authentication = disabled

   Anonymous authentication = enabled

> Required because a local account instead of a domain login is used for soap authentication

3. Assign the write permissions **USU Analytics: Build interface** to the SOAP interface user.

Create DB user for

Create a read-only user for the USU License Management database to use in USU Analytics for data access from USU License Management.

Configure connection in configuration file

1. Open %VMA_BASERVER%/rebuild-st-interfaces/application.yml with a text editor.
2. Enter the SOAP Url of the system that will be the datasource.

> The WSDL address for it is .../htdocs/soap/dyn_server.php?wsdl attached to the URL of the used application instance.

3. Enter the username and password of the interface user you created in Create SOAP user for (page 40).
4. Save your settings.

Configure connection in

1. Open **Toolkit > Maintenance > Data Source Configuration**.
2. Find the data source "Aspera - " and click the **pencil button** to edit the data source.
3. Enter the username and password of the interface user you created in Create SOAP user for (page 40).
4. Enter the Hostname.
5. For MSSQL enter the Qualifier, which is provided with the DB user. This step is optional for MariaDB or MySQL.
6. For Oracle enter the SID provided by the Oracle admin and Service Name

Configure Job Scheduler in

1. Open **Toolkit > Job-Scheduler**.
2. Find the job "prep ST interfaces" and click the **pencil button** to edit the job.
3. Expand the section **Command editor**.
4. Edit first command line to you actual installation path.
5. Optionally edit second command line. If you don't need the truststore, remove `-Djavax.net.ssl.trustStore=/etc/ssl/aspera-truststore.jks`
6. Expand the section **Trigger**.
7. Set a time for the job to start.

> The Job "SAMI standard load" is required to run before "prep ST interfaces"

Configure Xtract in

1. Open **Toolkit > Xtract**.
2. Choose source **USU License Management - Import Quality [iq]**

> The source refresh can take several minutes

Initialize database in

1. Open **Toolkit > Maintenance > System Configuration**.
2. Expand the section **Expert Parameters**.
3. Enable the option **Staging mode** and set **Staging mode database type** to **COLUMNSTORE**.
4. Save your settings.
5. Go back to **Toolkit** and go to section **Data Load**.
6. Click **Create DM**.
7. Click **ReCreate DateTime dimensions**.
8. You are done. You can wait for the jobs to be executed or you can initialize the jobs manually in the **Job-Scheduler**.

TEST THE CONNECTION

Perform the following test to check if the connection was established correctly.

Test	Expected result
An analyzer report can be created, saved and scheduled to be executed	Please refer to the user manual for details. Please note, that analysis you create at this point could be empty depending on the data in source system or USU Analytics database.
A dashboard can be opened	Please refer to the user manual for details. Please note, that dashboards you open at this point could be empty depending on the data in source system or USU Analytics database.

2.9. Uninstall USU Analytics

2.9.1. Microsoft Windows Based Operating System

It is possible to uninstall the USU Analytics application by executing the script `uninstall.bat` in *%VMA_HOME%* folder. The script needs to be executed as administrator.

2.9.2. UNIX Based Operating System

For Linux instances, it is enough to remove the *%VMA_HOME%* folder.

2.10. Appendix

2.10.1. Manual steps during installation on Windows Server 2012R2 and 2016

Since version 5.5 it is mandatory to have utility tar available in operating system in order to install the product.

On Microsoft Windows Server 2019 and on Linux distributions there is tar usually shipped together with operating system.

On Microsoft Windows Server 2012R2 and 2016 it might be needed to apply following steps to install tar:

1. Install Microsoft Visual C++ 2015 Redistributable Update from **https://www.microsoft.com/en-us/download/details.aspx?id=52685**
2. Deploy tar.exe from release package folder platform\tar\ to C:\Windows\

2.10.2. Unattended Installation

USU Analyticscan be installed unattended (automatic installation with predefined values):

1. We deliver a defaults file with the release package. You find it next to the installer jar. We recommend using this file.
2. If you want to generate a defaults file by yourself, you can do it like this:

```
java -jar usu_analytics-full_installation-<version number>.jar -options-template
usu_analytics-full_installation-<version number>.defaults
```

When creating the default installation file, you may be asked for values for which no default value is provided, e.g. like this:

```
Validation failed:
No path was specified. Please enter a valid path.

Press 1 to redisplay, 2 to quit
```

Answer this question with 1 and press enter.

```
JDBC driver


Because of license restrictions we are not allowed to ship JDBC drivers with our solution.
Please download the JDBC driv er for your chosen database management system and select the
jar file. For the currently supported version of the required JDBC driver please see the
Analytics/SAMI: id_TITLE Installation Manual.
You are able add up to 3 drivers (If your source database is running on different DBMS
please add second related JDBC driver).

JDBC driver (required):[]
```

Provide a valid path and file name for a jdbc driver and press enter. The next two you can skip by pressing enter. Answer the next two questions with 1 and enter.

```
License files


Please select the location, where your Analytics/SAMI: id_TITLE license files are stored.
Please make sure they are unzipped.
```

```
Location:[]
```

Provide a valid path and press enter.

```
Press 1 to continue, 2 to quit, 3 to redisplay
```

Answer this question with 1 and press enter.

3. Edit the defaults file. Fill required variables.
4. Encrypt passwords

   For unattended installation it is needed to encrypt passwords in the defaults file. We provide you with a tool to do a bulk encrypt.

   Copy this tool to your installation folder: <UA RELEASE PACKAGE>\platform\tools\encryption_tool-1.0.jar

   Be sure you have set the parameter aes_key_override with a valid AES 256 bit key

```
# Password Security
aes_key_override=UsRcIGQ/n7FJJYdEuo050rqZuuVG/1om
```

> If you don't use a valid key, your installation will fail and the system will not be usable.

   Start the tool with following parameters:

```
java -jar encryption_tool-1.0.jar -e
-f "<path to the defaults file with plain text passwords>\usu_analytics-full_installation--
<version number>.defaults"
-o "<path to the defaults file with encrypted passwords>\usu_analytics-full_installation--
<version number>.defaults"
```

5. Execute automatic installation:

```
java -jar usu_analytics-full_installation-<version number>.jar -defaults-file "<path to
the defaults file with encrypted passwords>\usu_analytics-full_installation--<version num-
ber>.defaults"valuemation_analytics-full_installation-<version number>.defaults –auto
```

   If defaults file has same name and is placed in same directory, it doesn't have to be specified.

```
java -jar usu_analytics-full_installation-<version number>.jar -auto
```

   If used without -auto, the according interactive mode is used (either GUI or console), and panels will show user input fields prefilled with the default values found in the file.

> This can be used only for variables. Steps with optional packages use default selection.

More information can be found here https://izpack.atlassian.net/wiki/spaces/IZPACK/pages/42270722/Mixed+Installation+Mode+Using+Variable+Defaults.

2.10.3. Change Java Memory Settings

2.10.3.1. Microsoft Windows Based Operating System

This section outlines how to adjust maximum amount of memory to be available for the application server (Tomcat) under which BA Server is running.

1. Stop BA Server (service `USU SI BA Server`)
2. Go into the `%VMA_BASERVER%\tomcat\bin` directory and rename the `ususibaserverw.exe` file to match the BA Server service name `ususibaserverw.exe > ususibaserverw_changeit.exe`
3. After you have renamed the file, open it by double-clicking on it or right-click and select Run as Administrator. This depends on your user permission settings. The Properties Window will open.
4. Select the Java tab
5. Based on available memory, set the Maximum memory pool parameter to 1024, 2048, 4096, 8192 etc. and click on OK
6. Rename the `ususibaserverw_changeit.exe` file back to `ususibaserverw.exe`
7. Start BA Server

Your Tomcat server now has increased maximum memory limits. You can adjust the JvmMx number, which is a parameter that specifies the maximum limit, to a higher number if you prefer. However, if the Java virtual machine refuses to start with increased limits, then you will have to add more RAM to your system, stop some memory-intensive services, or reduce the maximum memory limit to a lower number. This problem occurs when there is not enough contiguous memory available to assign to the JVM.

We also recommend to add `-XX:+UseConcMarkSweepGC -XX:+CMSIncrementalMode`. This is a garbage collection algorithm that allows to do most of the garbage collection work in the background without stopping application threads while working.

2.10.3.2. UNIX Based Operating System

This section outlines how to adjust maximum amount of memory to be available for the application server (Tomcat) under which BA Server is running and how to allow a garbage collection for running application.

1. Stop BA Server
2. Based on available memory, edit startup script `%VMA_BASERVER%/start-pentaho.sh`, and change value next to the `–Xms` and `–Xmx` parameter respectively to 1024m, 2048m, 4096m, 8192m etc. respectively, e.g.: `export CATALINA_OPTS="-Xms2048m –Xmx8192m -XX:MaxPermSize=256m`
3. We also recommend to add `-XX:+UseConcMarkSweepGC -XX:+CMSIncrementalMode`. This is a garbage collection algorithm that allows to do most of the garbage collection work in the background without stopping application threads while working.

   For Linux can look like e.g.:

   ```
   CATALINA_OPTS="-Xms2048m -Xmx8192m -XX:MaxPermSize=256m
   -XX:+UseConcMarkSweepGC -XX:+CMSIncrementalMode -Dsun.rmi.dgc.client.gcInterval=3600000
   -Dsun.rmi.dgc.server.gcInterval=3600000 -Dfile.encoding=utf8 -DDI_HOME=\"$DI_HOME\""
   ```

4. Start BA Server

Your Tomcat server now has increased maximum memory limits. You can adjust the JvmMx number, which is a parameter that specifies the maximum limit, to a higher number if you prefer.

However, if the Java virtual machine refuses to start with increased limits, then you will have to add more RAM to your system, stop some memory-intensive services, or reduce the maximum memory limit to a lower number. This problem occurs when there is not enough contiguous memory available to assign to the JVM.

2.10.4. Installation of Database Management System

The database management system serves all parts of the solution that require having a relational database in place.

These parts are:

- USU Analytics BA Server (BA Server Database Repository) to store
- BI Artefacts (analyzer reports, dashboards, etc.)
- User Access Management
- Scheduling
- USU Analytics Toolkit and data storage
- Metadata (MD)
- Extraction Area (EA)
- Data Mart (DM)

2.10.4.1. Recommended MySQL server settings

In one of the following files:

- `%SI_DBMS_SERVER%\my.ini` ([mysqld] section) for Windows based OS
- `/etc/my.cnf` (*[mysqld]* section) for UNIX based OS

insert/edit the following parameters:

```
innodb_buffer_pool_size = 2G
innodb_file_per_table = 1
innodb_log_buffer_size = 256M
lower_case_table_names = 1
max_connections = 500
query_cache_size = 0
query_cache_type = 0
```

For MySQL 8 this setting is required:

```
log_bin_trust_function_creators = 1
```

The most important is the `innodb_buffer_pool_size` parameter. The more the better, but it should not exceed 70-80% of server's RAM.

The other (default) MySQL settings should be kept as are unless there's a specific need for an adjustment. Any change can greatly affect the performance of the data integration and should be discussed separately.

If you encounter problems with encoding you can try solving this by adding following parameters to your my.cnf:

```
character-set-server = utf8
collation-server = utf8_unicode_ci

skip-character-set-client-handshake
init_connect = 'SET character_set_client = utf8'
init_connect = 'SET character_set_results = utf8'
init_connect = 'SET collation_connection = utf8_unicode_ci'
```

Generally said, the better hardware is used for the data integration and MySQL server, the better performance. This is mainly related to the size of RAM, to the HDD/SSD disk's speed and to the performance of a CPU and the number of cores it has.

Running both the data integration and the database instance on the same server oftentimes improves the performance, because any network device (physical or virtual) between the data integration server and the database will likely slow down the whole process. In some cases, where there's a misconfigured network device or a device with a lot of traffic outside the data-integration-to-DB communication, there can be a difference in dozens of percent.

2.10.4.2. Recommended MariaDB server settings

In one of the following files:

- `%SI_DBMS_SERVER%\my.ini` ([mysqld] section) for Windows based OS
- `/etc/my.cnf` ([mysqld] section) for UNIX based OS

insert/edit the following parameters:

```
innodb_buffer_pool_size = 2G
innodb_file_per_table = 1
innodb_log_buffer_size = 256M
lower_case_table_names = 1
max_connections = 500
query_cache_size = 0
query_cache_type = 0
```

The most important is the `innodb_buffer_pool_size` parameter. The more the better, but it should not exceed 70-80% of server's RAM.

The other (default) MariaDB CS settings should be kept as are unless there's a specific need for an adjustment. Any change can greatly affect the performance of the data integration and should be discussed separately.

If you encounter problems with encoding you can try solving this by adding following parameters to your my.cnf:

```
character-set-server = utf8
collation-server = utf8_unicode_ci

skip-character-set-client-handshake
init_connect = 'SET character_set_client = utf8'
init_connect = 'SET character_set_results = utf8'
init_connect = 'SET collation_connection = utf8_unicode_ci'
```

Generally said, the better hardware is used for the data integration and MariaDB server, the better performance. This is mainly related to the size of RAM, to the HDD/SSD disk's speed and to the performance of a CPU and the number of cores it has.

Running both the data integration and the database instance on the same server oftentimes improves the performance, because any network device (physical or virtual) between the data integration server and the database will likely slow down the whole process. In some cases, where there's a misconfigured network device or a device with a lot of traffic outside the data-integration-to-DB communication, there can be a difference in dozens of percent.

2.10.4.3. Recommended MariaDB CS settings

In one of the following files:

- `%SI_DBMS_SERVER%\my.ini` ([mysqld] section) for Windows based OS
- `/etc/my.cnf` ([mysqld] section) for UNIX based OS

insert/edit the following parameters:

```
innodb_buffer_pool_size = 2G
innodb_file_per_table = 1
innodb_log_buffer_size = 256M
lower_case_table_names = 1
max_connections = 500
query_cache_size = 0
query_cache_type = 0
sort_buffer_size = 128M
```

The most important is the `innodb_buffer_pool_size` parameter. The more the better, but it should not exceed 70-80% of server's RAM.

The other (default) MariaDB CS settings should be kept as are unless there's a specific need for an adjustment. Any change can greatly affect the performance of the data integration and should be discussed separately.

If you encounter problems with encoding you can try solving this by adding following parameters to your my.cnf:

```
character-set-server = utf8
collation-server = utf8_unicode_ci

skip-character-set-client-handshake
init_connect = 'SET character_set_client = utf8'
init_connect = 'SET character_set_results = utf8'
init_connect = 'SET collation_connection = utf8_unicode_ci'
```

Generally said, the better hardware is used for the data integration and MariaDB server, the better performance. This is mainly related to the size of RAM, to the HDD/SSD disk's speed and to the performance of a CPU and the number of cores it has.

Running both the data integration and the database instance on the same server oftentimes improves the performance, because any network device (physical or virtual) between the data integration server and the database will likely slow down the whole process. In some cases, where there's a misconfigured network device or a device with a lot of traffic outside the data-integration-to-DB communication, there can be a difference in dozens of percent.

2.10.5. USU Analytics Toolkit Standalone Installation

It is possible to setup a separate tomcat server to deploy the USU Analytics Toolkit. In this case the following steps are necessary based on the OS.

2.10.5.1. Prerequisites

Java Development Kit (JDK) version 8 must be installed on the server. The latest version of JDK 8 can be obtained here: http://www.oracle.com/technetwork/java/javase/downloads/index.html

2.10.5.2. Microsoft Windows Based Operating System

1. Get binary distribution of Tomcat from here: https://tomcat.apache.org/download-80.cgi
2. Select Core Binary distribution for 64-bit Windows.zip and download it

3. Unpack binary distribution of Tomcat on server to folder *%TOMCAT_INSTALL_DIR%*

2.10.5.2.1. Configure Java Virtual Machine

If JAVA_HOME already points to the installed JDK, this subchapter can be skipped.

A specific JAVA_HOME configuration will be defined for this Tomcat installation.

- Create file *%TOMCAT_INSTALL_DIR%*/bin/setenv.bat
- Edit file setenv.bat and add the following line where JAVA_HOME_DIR is the location of the JDK-installation, i.e. C:\Program Files (x86)\Java\jdk1.8.xxxx\:

```
Set JAVA_HOME=<JAVA_HOME_DIR>
```

2.10.5.2.2. Install as Service

Switch to folder *%TOMCAT_INSTALL_DIR%/bin* and execute the following command to create a service called USU Toolkit

service.bat install USUToolkit

2.10.5.3. UNIX Based Operating System

2.10.5.3.1. Deployment

- Get binary distribution of Tomcat 8 from here: https://tomcat.apache.org/download-80.cgi
- Select Core Binary distribution tar.gz and download it
- Unpack binary distribution of Tomcat on server to folder *%TOMCAT_INSTALL_DIR%*

2.10.5.3.2. Permissions

1. Give permissions to user usu_si_ba:

```
sudo chgrp -R <user_group_name> %TOMCAT_INSTALL_DIR%/conf
sudo chmod g+rwx %TOMCAT_INSTALL_DIR%/conf
sudo chmod g+r %TOMCAT_INSTALL_DIR%/conf/*
```

2. Make user usu_si_ba owner of work, temp and log directories:

```
sudo chown -R usu_si_ba %TOMCAT_INSTALL_DIR%/work/ %TOMCAT_INSTALL_DIR%/temp/
%TOMCAT_INSTALL_DIR%/logs/
```

2.10.5.3.3. Configure Java Virtual Machine

If JAVA_HOME already points to the installed JDK, this subchapter can be skipped.

A specific JAVA_HOME configuration will be defined for this Tomcat installation.

- Create file *%TOMCAT_INSTALL_DIR%/bin/*setenv.sh and make it executable

```
cat > %TOMCAT_INSTALL_DIR%/bin/setenv.sh
chmod +x %TOMCAT_INSTALL_DIR%/bin/setenv.sh
```

Edit file setenv.sh and add the following line where JAVA_HOME_DIR is the location of the JDK-installation, i.e. /usr/java/jdk1.8.0_111/:

```
JAVA_HOME=<JAVA_HOME_DIR>
```

2.10.5.3.4. Install init script for UNIX based OS

To start tomcat automatically on startup a startup script needs to be created.

The following templates can be used.

init.d script

1. Create file /etc/init.d/tomcat8_toolkit
2. Enter the following content and replace the variables:

```
#!/bin/bash

### BEGIN INIT INFO
# Provides:tomcat8_toolkit
# Required-Start:  $network
# Required-Stop:   $network
# Default-Start:   2 3 4 5
# Default-Stop:    0 1 6
# Short-Description: Start/Stop tomcat8_toolkit server
### END INIT INFO

PATH=/sbin:/bin:/usr/sbin:/usr/bin

start() {
 sh %TOMCAT_INSTALL_DIR%/bin/startup.sh
}

stop() {
 sh %TOMCAT_INSTALL_DIR%/bin/shutdown.sh
}

case $1 in
  start|stop) $1;;
  restart) stop; start;;
  *) echo "Run as $0 <start|stop|restart>"; exit 1;;
es
```

3. Change permissions and set symlink

```
chmod 755 /etc/init.d/tomcat8_toolkit
update-rc.d tomcat8_toolkit defaults
```

The server now starts automatically on startup and can be controlled with the following commands:

```
service tomcat8_toolkit <stop|start|restart>
```

systemd script

1. Create file /etc/systemd/system/tomcat8_toolkit.service
2. Enter the following content and replace the variables:

```
############################################################
#
# sample of forking service - USU USU Analytics Toolkit - systemd service file
#
# /etc/systemd/system/tomcat8_toolkit.service
#
[Unit]
Description=USU USU Analytics Toolkit
#After=mysql.target
[Install]
WantedBy=multi-user.target
Alias=tomcat8_toolkit.service
[Service]
Type=forking
User=usu_si_ba
Group=usu_si_ba
```

```
# Start main service
ExecStart=%TOMCAT_INSTALL_DIR%/bin/startup.sh
# Stop main service
ExecStop=%TOMCAT_INSTALL_DIR%/bin/shutdown.sh
Restart=no
PrivateTmp=false
#########################################################
```

3. Change permissions and activate the service

```
cd /etc/systemd/system
chown root:root tomcat8_toolkit.service
chmod 0750 tomcat8_toolkit.service
systemctl daemon-reload
systemctl enable tomcat8_toolkit
```

The server now starts automatically on startup and can be controlled with the following commands:

```
systemctl start tomcat8_toolkit
systemctl stop tomcat8_toolkit
```

2.10.5.4. General configuration

2.10.5.4.1. Deploy sitoolkit.war

Copy `sitoolkit.war` from installation package to *`%TOMCAT_INSTALL_DIR%/webapps`*

By default, USU Analytics Toolkit is automatically deployed on startup of tomcat or when tomcat is running.

2.10.5.4.2. Configure Tomcat Security

To prevent unauthorized users to access USU Analytics Toolkit it is required to create a Tomcat user with a password and restrict access to USU Analytics Toolkit only to his user.

`tomcat/conf/tomcat-users.xml`

1. Create role SI_TOOLKIT_USER
2. Assign it to a user

2.10.5.4.3. Change Port

To change the default port of Apache Tomcat, edit the following file:

```
%TOMCAT_INSTALL_DIR%/conf/server.xml
```

After the configuration of server port, the Apache Tomcat server needs to be restarted.

The following ports can be configured:

```
Line 22: <Server port="8005" shutdown="SHUTDOWN">
```

Usage of port: The TCP/IP port number on which this server waits for a shutdown command. Set to -1 to disable the shutdown port.

Disabling the shutdown port works well when Tomcat is started using Apache Commons Daemon (running as a service on Windows or with jsvc on un*xes). It cannot be used when running Tomcat with the standard shell scripts though, as it will prevent shutdown.bat|.sh and catalina.bat|.sh from stopping it gracefully.

```
Line 69: <Connector port="8080" protocol="HTTP/1.1"
```

Usage of port: The TCP port number on which this Connector will create a server socket and await incoming connections. Your operating system will allow only one server application to listen to a port number on a IP address. If the special value of 0 (zero) is used, then Tomcat will select a free port at random to use for this connector. This is typically only useful in embedded and testing applications.

```
Line 71: redirectPort="8443"
```

Usage of redirectPort: If this Connector is supporting non-SSL requests, and a request is received for which a matching <security-constraint> requires SSL transport, Catalina will automatically redirect the request to the port number specified here.

```
Line 71: <Connector port="8009" protocol="AJP/1.3" redirectPort="8443" />
```

3. BA Server Administration

3.1. User Authentication

By default, the BA Server establishes roles, users, and initializes a basic configuration for the default built-in security data access object. You will almost certainly want to customize the roles and delete or modify the default users and add your own; at most, you will want to use your own LDAP authentication mechanism with the BA Server. This section explains these tasks in detail.

3.1.1. Local based authentication

The default security data access object is a custom Hibernate-based user/password DAO that reads and writes usernames, passwords, and roles to a relational database via Hibernate object-relational mapping. You do not have to do anything to initialize the default data access object; it is enabled automatically. However, you will almost certainly need to establish roles and users to match your organizational structure. Instructions for creating and modifying roles and users are in the *Authorization* section below.

The BA Server regulates user- and role-level access to the resources available in the solution repository of the User Console. The resources can be URLs and BI artefacts such as reports, analysis views, dashboards, etc. This section explains how to modify these access controls according to your preferences.

3.1.1.1. Add or Delete Users

1. Login to the User Console as an admin-user.
2. From User Console Home menu, click **Administration**, then select **Users & Roles** from the left.
3. Click on **Manage Users** tab.
4. To **add** a user:
   a. Click the plus sign (**+**) next to **Users**: The **New User** dialog box appears.
   b. Enter the new user's **User Name, Password, Password Confirmation** (the same password typed in a second time).
   c. Click **OK**.
5. To **delete** a user:
   - Choose (highlight) the user and click the x sign (**X**) next to **Users**: and confirm the deletion.

3.1.1.2. Add or Delete Roles and Assign Users to Roles

1. Login to the User Console as an admin-user.
2. From User Console Home menu, click **Administration**, then select **Users & Roles** from the left.
3. To **add** a role:
   a. Click on **Manage Roles** tab.
   b. Click the plus sign (**+**) next to **Roles**. The **New Role** dialog box appears
   c. Enter the role's **Role Name**.
   d. Click **OK**.

4. To **delete** a role:
   - Click the x sign (**X**) next to **Roles** and confirm the deletion.
5. To **assign** users to roles:
   a. Under **Roles**, select the role that you want to associate with a user or users.
   b. In the **Users Available** list, click to highlight the user or users that you want to associate with the selected role.
   c. Click the right arrow to move the selected users to the **Users Selected** list. You can click the double-right arrow to move all users from the **Users Available** to the **Users Selected** list.
   d. You can remove users from the **Users Selected** list by highlighting that user and clicking on the left arrow. The user moves from the **Users Selected** to **Users Available** list, and no longer has the permissions associated with the highlighted role.

The users that appear in the **Users Selected** list are now tied to the highlighted role and have all of the permissions associated with that role.

3.1.2. LDAP based authentication

You must have a working directory server with an established configuration including two LDAP roles **Admin** and **Authenticated** created before continuing. If roles are not possible to be created, please, contact our consulting services for further reference.

It is possible to change to different users, please see the security_guide.pdf for Pentaho for detailed instructions.

Follow the below instructions to switch the BA Server's default security data access authentication backend to LDAP.

1. From User Console Home menu, click **Administration**, then select **Authentication** from the left. The **Authentication** interface appears. **Local - Use basic Pentaho Authentication** is selected by default.
2. Choose the **External - Use LDAP / Active Directory server** radio button. The **LDAP Server Connection** fields populate with a default URL, user name, and password.
3. Change the **Server URL**, **User Name**, and **Password** as needed.
4. Click **Test Server Connection** to verify the connection to your LDAP server and to complete the set up.
5. Click the node to select the **Pentaho System Administrator** user and role to match your LDAP configuration.
6. Click **OK**.
7. Choose your **LDAP Provider** from the drop-down menu.
8. Configure the LDAP connection as explained in LDAP Properties.
9. Click Test.

The Admin user is required for all system-related operations, including the creation of user folders. The Administrator Role is required for mapping a third-party admin role to the Pentaho admin role (Administrator). This is required for all ABS functionality to work properly.

3.1.3. USU Service Management authentication

Users which have a user account in USU Service Management can authenticate without additional synchronization. Roles are assigned in the USU Service Management user management and used in USU Analytics.

Three groups are defined for analytics. We defined the rights of the groups as follows:

- ANALYTICS ADMIN: User, who should be able administrate the application
- ANALYTICS POWERUSER: User is allowed to create and schedule reports, but not adjust the loading process or administrate roles
- ANALYTICS READER: User can only read data and open existing reports

You are not limited to these groups – you are free to define your own setup. Also, you can define the rights differently.

These steps are needed to enable the functionality:

1. Open USU Analytics Toolkit and start configuration:
   a. Configure proper SD database connection to the application in Data Source Configuration
   b. Open System Configuration and ensure that parameter **Expert Parameters / SM authentication** is set to true (checked).
   c. Open System Configuration and define in parameter **Toolkit / SM roles** the groups, which should have access to the application. Provide a comma separated list, e.g.: ANALYTICS ADMIN, ANALYTICS POWERUSER, ANALYTICS READER.
   d. Set parameter provider=memory in file `server\pentaho-server\pentaho-solutions\system\security.properties`
   e. Add hardcoded users to your configuration in file `server\pentaho-server\pentaho-solutions\system\applicationContext-spring-security-memory.xml`
   f. Configure the groups you have added to parameter VM_ROLES_AUTHENTICATED in file`server\pentaho-server\pentaho-solutions\system\applicationContext-pentaho-security-memory.xml`
2. Restart BA server.
3. Log in to User Console as administrator and configure the rights of the groups in Administration (User & Roles \ Manage Roles).

The application sometimes uses domain information. If you log in to the application using domain information, please use <USERNAME>@<DOMAIN> in User Name field in login screen.

3.1.4. USU Service Management authentication with custom SQL

By default, USU Analytics provides the possibility to setup authentication against USU Service Management and roles.

This behavior can be enhanced if special rules should be checked, i.e.: users are only allowed to access USU Analytics if special conditions are met or roles need to be "generated" from USU

Service Management data, i.e. for Multi-Tenency or for special roles based on organization units or business partners.

In these cases the following configurations need to be done:

1. How to activate USU Service Management Authentication is explained in chapter above.
2. Open USU Analytics with administrative permissions.
3. Open Toolkit/Data Source Configuration.
4. Open data source in edit mode. For this example we will use a1.
5. Configure SQL statements for dynamic queries.
6. Provide SQL statement used to obtain users from the source system and import them, e.g. for USU Service Management :

```
SELECT s3u.DOMAIN          AS domain,
       s3u.CRYPTEDPASSWORD AS cryptedpassword,
       s3u.HASH            AS hash
  FROM AMT_S3USER s3u
  WHERE s3u.ACTIVE = 'Y'
    AND s3u.USERID =?
```

7. Provide SQL used to obtain roles from the source system and import them, e.g. for USU Service Management :

```
SELECT s3g.GROUPNAME AS role
  FROM AMT_S3GROUP s3g
```

8. Provide SQL statement used to obtain all users from the source system, e.g. for USU Service Management :

```
SELECT s3u.DOMAIN AS domain,
       s3u.USERID AS userid
  FROM AMT_S3USER s3u
  WHERE s3u.ACTIVE = 'Y'
```

9. Provide SQL statement used to obtain the relationship between users and roles from the source system and import them, e.g. for USU Service Management :

```
SELECT s3u.DOMAIN    AS domain,
       s3g.GROUPNAME AS groupname
  FROM AMT_S3USER       s3u
  JOIN AMT_USERGROUPMAP ugm ON s3u.S3USER_ID = ugm.S3USER_ID
  JOIN AMT_S3GROUP      s3g ON ugm.S3GROUP_ID = s3g.S3GROUP_ID
  WHERE s3u.USERID =?
```

10. Open Toolkit/System configuration. Check SM-Authentication with user defined SQL
11. Restart Analytics server
12. Verify result with login

3.1.5. USU License Management authentication

Users which have a user account in USU License Management can authenticate without additional synchronization. Roles are assigned in the USU License Management user management and used in USU Analytics.

The following steps are needed to enable the functionality:

1. Open USU Analytics with administrative permissions.
   - Configure a working data source for USU License Management.

2. Open the following configuration file and set the parameter **provider** to **memory**.
   `server\pentaho-server\pentaho-solutions\system\security.properties`
3. Open the following configuration file and specify a secure password for the **admin** user.
   `server\pentaho-server\pentaho-solutions\system\applicationContext-spring-security-memory.xml`
4. Restart USU Analytics for the changes to take effect.
5. Open USU Analytics as “admin” using the password specified in Step 3.
   - Go to **Home > Administration > Users > Roles > Manage Roles** and configure the permissions for the three default roles: **ANALYTICS ADMIN**, **ANALYTICS POWERUSER** and **ANALYTICS READER**.
6. Open USU License Management with administrative permissions.
   a. Add the following configuration variables:
      i. `security_auth_token_cipher`: set its value to **rijndael-128**
      ii. `security_auth_token_key`: set its value to a secure key of your own choice.

      > The key must be 16 characters long.

   b. Open the configuration for the application USU Analytics and insert the USU Analytics URL which should be similar to:

      ```
      http://localhost:8080/vma_integration/launch_st.jsp
      ```

   c. Create the application roles **ANALYTICS ADMIN**, **ANALYTICS POWERUSER** and **ANALYTICS READER** and grant the permission **USU Analytics** to those roles.
   d. Assign the roles to USU License Management users as needed.
7. Test the USU License Management authentication via the menu item **Applications > USU License Management**.

Optional: specifying custom roles

You are not limited to using the three roles that are supplied by default for use with USU License Management authentication. You are free to define your own setup by adding or removing roles in the following USU Analytics configuration file.

```
server\pentaho-server\pentaho-solutions\system\applicationContext-pentaho-security-memory.xml
```

Restart **USU Analytics** for your changes to take effect. Go to **Home > Administration > Users > Roles > Manage Roles** and configure the permissions for the newly added roles. Go to **Toolkit > System Configuration > Toolkit** and configure the permissions for the newly added roles. Any changes made here should also be reflected in USU License Management as described in Step 6.c and Step 6.d.

Optional: specifying hardcoded users

In addition to users that are supplied by USU License Management, additional hardcoded users can be specified. This can be helpful for a variety of reasons.

- When USU License Management authentication is (temporarily) unavailable for some reason, a hardcoded user can serve as a backup.

- Users that need access to USU Analytics but do not have access to USU License Management can't use USU License Management authentication. Specifying such users in a static way circumvents this issue.

Static users can be specified in the following configuration file.

```
server\pentaho-server\pentaho-solutions\system\applicationContext-spring-security-memory.xml
```

The username is defined as the key and a comma-separated list of values contain the password followed by any number of roles that will be assigned to the user as shown in the following example.

```
<prop key="admin">password,Administrator,Authenticated</prop>
```

Choose secure passwords for any users defined here.

3.1.6. SSO Based on HTTP Header

There are multiple ways to configure SSO for the application. As an example, we will describe the configuration of Single Sign On mechanism based on SAML 2. This is a type of SSO, which is based on HTTP header information.

In the example it is expected, that Single Sign On authentication will be handled by Apache HTTPD server plugin mod_auth_mellon and our application will receive information about the authenticated user in HTTP header parameter VMA_USER.

For our example we need these prerequisites:

- SM Authentication is configured and working
- USU Analytics roles and groups are configured and working
- Apache HTTPD 2.4 is available and following plugins are available: mod_auth_mellon, mod_proxy_ajp, mod_proxy
- The plugin mod_auth_mellon should be configured like described on following blog: **http://mkchendil.blogspot.cz/2015/02/apache-and-openam-saml-federation.html**.
- Add parameter RequestHeader set VMA_USER %{MELLON_UID}s in site configuration

These steps are needed to enable the functionality:

- Log in to the application with administrative rights and open Toolkit
- Open System Configuration\User Console Parameters and provide the field SSO HTTP header with the name of the parameter, which holdes the name of the authenticated user. In our example this is VMA_USER.
- Log out and restart the service.
- You should now be logged in automatically

3.1.7. Debugging SSO

3.1.7.1. Enable Logs for SSO

1. Go to file \server\pentaho-server\tomcat\webapps\console\WEB-INF\classes\log4j.xml
2. Make backup of the file log4j.xml before you edit it.
3. Add this code to the end of the file:

```
<appender name="USU_SSO" class="org.apache.log4j.RollingFileAppender">
    <param name="File" value="${catalina.base}/logs/usu_sso.log"/>
    <param name="Append" value="true"/>
    <param name="MaxFileSize" value="500KB"/>
    <param name="MaxBackupIndex" value="1"/>
    <layout class="org.apache.log4j.PatternLayout">
        <param name="ConversionPattern" value="%d %-5p [%c] %m%n"/>
    </layout>
</appender>

<category name="de.usu">
    <priority value="DEBUG"/>
    <appender-ref ref="USU_SSO"/>
</category>
```

4. Restart VMA server
5. Go to file \server\pentaho-server\tomcat\logs\usu_sso.log
6. When you finish troubleshooting it is recommended to revert log4j.xml to old state and restart the server to disable debugging messages.

3.1.7.2. Use request_diagnostics.jsp for Debugging SSO

In our newer versions we deliver a possibility to retrieve detailed information from the SSO request. If request_diagnostics.jsp is not available on your installation, please ask our support team.

To access the page:

- Log in to the application. Without login, the test will not work.
- Access the page with a link similar to this: **https://your.vma.url.com/console/request_diagnostics.jsp**

Check the HTTP header information and especially the value of the parameter VMA_USER.

- The value of parameter must be the same user name and format as you used, when you logged in when just the USU Service Management Authentication was configured.
- Is the correct domain delivered?
- Is domain information in correct place? VMA_USER@demo.domain.com or demo.domain.com@VMA_USER ?

3.1.8. Other Possibilities to Implement SSO

- **SSO based on encrypted token** Configuration based on encrypted token can be used to integrate the application with other applications like USU Service Management or USU License Management. It can be enabled by modifying file server\pentaho-server\pentaho-solutions\system\applicationContext-spring-security-sso.xml and setting attribute SIWebSSOUserHTTPHeaderName to value ENCRYPTED_TOKEN. Configuration of authentication token is in file applicationContext-security-encrypted_token_security.properties. Authentication token is a 16-character long

key used to encrypt tokens. The tool can automatically create users, who use SSO log in with token. In order to enable this functionality, it is required to modify file server\pentaho-server\pentaho-solutions\system\applicationContext-spring-security-sso.xml and setting parameter “security” to value “memory”. By enabling this it will not be possible to maintain users in User Console any more. Users will have to be maintained in file server\pentaho-server\pentaho-solutions\system\applicationContext-spring-security-memory.xml. It is possible that there will be additional configuration needed. In this case, please contact USU Consulting.

- **SSO based on remote user** The application can be configured to use remote user provided by servlet container in request (method request.getRemoteUser()). This configuration can be done by reconfiguring file server\pentaho-server\pentaho-solutions\system\applicationContext-spring-security-sso.xml and setting parameter SIWebSSOUserHTTPHeaderName to value "REMOTE_USER”.All users must be created manually when this type of SSO is enabled. It is possible that there will be additional configuration needed. In this case, please contact USU Consulting.
- **SSO based on user principal** The tool can be configured to use session principal provided by servlet container in request (method request.getUserPrincipal()). This configuration can be done by reconfiguring file server\pentaho-server\pentaho-solutions\system\applicationContext-spring-security-sso.xml and setting parameter SIWebSSOUserHTTPHeaderName to value “USER_PRINCIPAL”. If session principal contains symbol ‘\’ (for example when it is domain name and user name), it will be changed to symbol ‘@’. For example user principal ‘USU\Joe.Doe’ will be changed to ‘USU@Joe.Doe’. Then it is required to have all users in format DOMAIN@Joe.Doe.All users must be created manually when this type of SSO is enabled. This type of SSO can be used with frameworks like Waffle. It is possible that there will be additional configuration needed. In this case, please contact USU Consulting.

3.2. Role Permission

1. Login to the User Console as an admin-user.
2. From User Console Home menu, click **Administration**, then select **Users & Roles** from the left.
3. Click on **Manage Roles** tab.
4. Click on the role in **Roles**. To the right you can see the five permissions a role can have:
5. Administer Security

   The default Administrator role automatically conveys all operation permissions to users assigned to that role, even if the check box next to it is cleared.
   - Gives access to the Administration perspective of the User Console.
   - Allows access to and the ability to manage all content in the Browse perspective.
   - Allows the ability to view and work with all user schedules in the Schedules perspective.
   - Gives the ability to create server block out times in the Schedules perspective.
6. Read Content (users in this role may read files
   - Gives the user the ability to view content in the Browse perspective.
   - Gives the user the ability to view content through the File > Open dialog.
7. Publish Content
   - Allows the user to schedule reports and content.
   - Gives the user the ability to view, edit, or delete their own schedules using the Schedules perspective.

8. Schedule Content
   - Allows client tools to store reports or data models in the Pentaho repository.
   - This includes tools such as Report Designer, Agile BI, Schema Workbench, and Metadata Editor.
9. Create Content
   - Allows the user to save a report to the repository.
   - Gives the user the ability to see the data sources that are used to create reports or dashboards.

3.3. Access Control List

1. Login to User Console as an admin-user.
2. From the User Console Home menu, click on **Browse Files** then select the folder or file you want to change the access rights for.
3. On the right side click on **More** menu and then choose **Share**.

3.4. License Installation

3.4.1. Via User Console

1. Login to User Console as an admin-user.
2. From User Console Home menu, click on **Administration** then select **Licenses** from the left.
3. Add license files by clicking on the **+**-sign on the right side above the license table.

3.4.2. Via Command Line

You can also manager License Keys from the command line.

Although the Administration page of the User Console is the quickest, easiest, and most comprehensive way to manage the BA Server, you may be in environments where it is difficult or impossible to deploy or use the console.

To ensure that the BA Server and DI Server use the same location to store and retrieve licenses, you must create a PENTAHO_INSTALLED_LICENSE_PATH system variable for your user account. If you do not set this variable, the User Console and the command line license installation script will store license data in a place that the DI Server and BA Server will not look.

3.4.2.1. Set the License Path Variable on Linux

1. Edit your **/etc/environment** file with a text editor. If you're using Solaris, you will have to set this environment variable through whatever means are available to you.
2. Add this line in a convenient place (changing the path as explained above, if necessary):
   ```
   export
   PENTAHO_INSTALLED_LICENSE_PATH=<path_for_licenses>/.installedLicenses.xml
   ```
3. You must log out and log back into the operating system for the change to take effect.
4. Verify that the variable is properly set:
   ```
   env | grep PENTAHO_INSTALLED_LICENSE_PATH
   ```
5. The PENTAHO_INSTALLED_LICENSE_PATH variable is now set.

3.4.2.2. Set the License Path Variable on Windows

- In **Windows Server**, right-click on **Computer**, then select **Properties** from context menu, and then click **Advanced System Settings**. The **System Properties** window will come up.
- In the System Properties window, click the **Advanced** tab, then click **Environment Variables**.
- In the System Variable section, click New.
- A popup dialog will ask for a variable name and value. Type **PENTAHO_INSTALLED_LICENSE_PATH** into the name field, and the directory you intend to install licenses to plus **.installedLicenses.xml** in the value field, then click **OK**. C:/licenses/.installedLicenses.xml
- In the parent window, click **Apply Changes**.
- You must restart your computer for the change to take effect.
- Verify that the variable is properly set:

```
echo %PENTAHO_INSTALLED_LICENSE_PATH%
```

- The PENTAHO_INSTALLED_LICENSE_PATH variable is now set.

If you run Tomcat automatically as a Windows service, you must also configure it to run on the Windows user account where the **.installedLicenses.xml** file is located.

3.4.2.3. Install or Update License Keys from the Command Line

1. Download the .lic file you want to install.
2. Navigate to the /license-installer/ directory.
3. Copy your .lic files to the /license-installer/ directory.
4. Run the license installation script.
5. **Linux:**

   Run *install_license.sh* with the install switch and the location and name of your .lic file as a parameter. You can specify multiple .lic files separated by spaces. Be sure to use backslashes to escape any spaces in the path or file name:`./install_license.sh install BA\ Platform\ Enterprise\ Edition.lic`
6. **Windows:**

   Run install_license.bat with the install switch and the location and name of your license file as a parameter:`install_license.bat install "C:/Users/dvader/Downloads/BA Platform Enterprise Edition.lic"`

3.5. Email Configuration for Pentaho

1. Login to User Console as an admin-user.
2. From User Console Home menu, click on *Toolkit* then click *System Configuration*.
3. Provide values for 'Mail Server (Server Settings)':
4. SMTP Host, Port, User Name, Password
5. Protocol (e.g. SMTPS or SMTP)

6. If wanted enter an email address in **All email sent from server will come from this address.**
7. If needed choose **Use Start TLS** or **Use SSL**
8. Press button Test Email Configuration.
9. If OK, press button Save.

3.6. Email Configuration Job-Scheduler

1. Login to User Console as an admin-user.
2. From User Console Home menu, click on *Toolkit* then click *System Configuration*.
3. Open section *Mail Server Settings*.
4. Provide values for 'Mail Server (Server Settings)':
5. SMTP Host, Port, User Name, Password
6. Protocol (e.g. SMTPS or SMTP)
7. If wanted enter an email address in **All email sent from server will come from this address.**
8. If needed choose **Use Start TLS** or **Use SSL**
9. If OK, press button Save.

3.7. Deletion Schedule

It is possible to delete the report files which were generated by a schedule with a specific age or older to be able to clear space.

1. Login to User Console as an admin-user.
2. From User Console Home menu, click on **Administration** and select **Settings** from the left. The **Settings** interface appears.
3. To manually delete generated files now, confirm the age of the files to be deleted, and then press **Delete Now**. The generated files older than the specified age are deleted.
4. To schedule regular deletion of generated files, press **Schedule Deletion**. The **Schedule Deletion** window appears.
5. Enter an age for the generated files in the field specifying number of days.
6. Choose a time interval for the schedule from the **Recurrence** list.
7. Choose a **Start Time** and a **Recurrence Pattern**, as well as **Start** and **End Dates** for the scheduled deletion.
8. Click **OK** to save.

The **Deletion Schedule** is created and the generated files older than the specified age are deleted during the scheduled times.

3.8. Change Default Theme

The default theme defines the login screen. For example, if you have installed the system with the default theme USU Blue, the system always displays the login screen for the USU Blue theme.

A user can choose a theme he likes, but scheduled reports will always use the default theme. For example, a user chooses USU Yellow in the frontend, but USU Blue is the default. If the user schedules an analysis report to be sent via the Job Scheduler, the default theme will be used.

The default theme can only be changed on the application server. To do this open the file **.\server\pentaho-server\pentaho-solutions\system\pentaho.xml** and change the value of the tag **default-theme**. This change requires a restart of the service to be activated.

3.9. Restart BA Server

Standard installation of the BA Server can be restarted by

- On Windows by restarting service “USU SI BA Server”
- On Linux by execute script:
  `/opt/usu/analytics/server/pentaho-server/restart-pentaho.sh`
- or following command if BA server is configured to start at system startup:
  init.d `script – /etc/init.d/vm-analytics restart`*systemd* `script – systemctl restart vm-analytics`

3.10. Deployment of Content

3.10.1. Add new database connection (JDBC)

1. Login to the User Console as an administrator user.
2. Add new JDBC connection through **Manage Data Sources/ New Connection...** button.
3. “usu_si_ra” name is recommended (RA = Reporting and Analysis)

3.10.2. Add/update Analysis Data Source (Mondrian Schema)

1. Login to the User Console as an administrator user.
2. Backup existing analysis data sources (for details see Backup section).
3. Add new Analysis data source through **Manage Data Sources/ Import Analysis** ... button
4. Browse for particular Mondrian schema and import it (replace if already existing)
5. Link the schema with *usu_si_ra* database connection created in previous step

Add the Localization Parameters

Follow these steps if importing localized Mondrian schema only. They are required to be done with every update of Mondrian schema.

1. Login to the User Console as an administrator user.
2. Edit existing analysis data source through **Manage Data Sources/ Edit...** button.
3. Check the „Manually enter data source parameter values“ option and add the following parameters:

Name	Value
DynamicSchemaProcessor	de.usu.si.mondrian.dsp.VMASchemaProcessor
UseContentChecksum	true

Use this for Versions lower than 4.5. It will use the localization properties files in the BA server: mondrian.i18n.LocalizingDynamicSchemaProcessor

3.10.3. Download and Upload Custom Artifacts

You can download and upload artifacts, like analyzer reports and custom dashboards. With this functionality it is possible to transfer and deploy customized content from one system to another.

Uploading and downloading from the repository is done with the User Console. The ability to upload and download assumes that you have already created a data source, that data content exists to be pushed, and defines permissions for the repository.

For uploading, any starting location can be selected. Permission settings are inherited through the folder structure if the destination location has existing permission settings. It is advisable to keep existing security settings as defaults for the upload. Uploading more than one file or folder requires the files to be zipped before beginning the upload.

For downloading, you can select the destination location for the downloaded file or folder. The download process always creates a .zip file that includes a manifest file along with the downloaded content. The manifest file contains the collection of permissions settings for the downloaded files and folders and is found in the root directory of the .zip file.

3.10.3.1. Download a Folder

Downloading folders can be done through the User Console.

1. From the User Console **Home**, click **Browse Files**. The **Browse Files** page appears.
2. From the **Browse** pane on the left, browse to the location of the folder to be downloaded.
3. With the folder highlighted, click **Download** in the **More** menu.
4. Choose **Save File** in the window that appears, and click **OK**.

The folder is saved as a .zip file with the manifest located in the top level of the file.

3.10.3.2. Download a File

Downloading files can be done through the User Console.

1. From the User Console **Home**, click **Browse Files**. The **Browse Files** page appears.
2. Browse to the location of the file by clicking through the folders in the **Browse** pane on the left. The **Files** pane in the center populates with a list of reports.
3. Click to select the file in the **Files** pane and choose **Download** from **More** menu.
4. Choose **Save File** in the window that appears, and click **OK**.

The file is saved as a .zip file with the manifest located in the top level of the file.

3.10.3.3. Upload Folders and Files

The User Console can be used to upload files and folders you downloaded as described in Download and Upload Custom Artifacts (page 65) to the repository. A quick word about manifests: when a user downloads content out of the repository; a manifest file is created within the zip file which contains the permission and ownership information about the file.

For Retain permission on upload file, the file permission contained in the uploaded zip (exportManifest.xml) will be the permission applied the repository. If the file doesn't have an entry in the exportManifest.xml for the permission, then it will use the default permission, which is inherited.

For Set Owner based on uploaded file, the owner found in the uploaded zip (exportManifest.xml) will be the owner of the file in the repository. If the file doesn't have an entry in the exportManifest.xml for the Owner, then it will set the Owner to the user who is uploading the zip.

These steps walk you through uploading files and folders you downloaded like described in Download and Upload Custom Artifacts (page 65) to the repository with the User Console.

1. From the User Console **Home**, click **Browse Files**. The **Browse Files** page appears.
2. From the **Browse** pane on the left, click to choose the destination folder for the upload.
3. With the destination folder highlighted, click **Upload** in **More** menu. The **Upload** dialog box appears.
4. Browse to the file or folder to be uploaded by clicking **Browse**.
5. Click **OK** to begin upload using the default settings.
6. Choose preferences for the upload.
7. Choose Replace the Existing File or Do Not Upload from the first menu.
8. Choose File Permissions from the second menu. The choices are Do Not Change Permissions or Retain Permissions on the Uploaded File.
9. If you selected Retain Permissions on the Uploaded File, choose File Ownership by selecting Do Not Change Owner or Set Owner Based on Uploaded File from the third menu.
10. Choose None, Short, or Verbose from the Logging menu.
11. Click **OK**.

The upload runs, and the files or folders are uploaded to the repository. If the upload fails, an error log window opens with specific information.

3.11. Manual password encryption

The system supports encryption of database passwords located in configuration files and MD database. Passwords are encrypted by default during installation process. For this type of password encryption, it is required to have a master password – key (AES 256 bit key), which is used by symmetric cryptography to encrypt all database passwords.

This master password is stored in the file system. For windows installation it is saved by default in folder `Windows\ServiceProfiles\LocalService\.usu\.analytics_encryption_key`.

When you install on a Linux based OS you find the file in the home folder of the user which is running the application, e.g., `/home/<my analytics user>/.usu/.analytics_encryption_key`.

If you want to change the location of the key file you can specify the path to this file in `Windows\ServiceProfiles\LocalService\.kettle\kettle.properties`. On Linux, in the corresponding directory of the user running the application.

Because the encryption of the passwords is done during installation, there should be no need to encrypt passwords manually. If you have to, e.g., due to password policies, we deliver a tool to do this manually. You find the tool in the release package: `<UA RELEASE PACKAGE>\platform\tools\encryption_tool-1.0.jar`.

The tool has a set of parameters, which are described in the help function of the tool:

`Run java -jar encryption_tool-1.0.jar -h`

```
Usage: encryption-tool [-eghV] [-f=<FILE>] [-k=<secret key|file:FILE>]
                       [-o=<output>] [-p=<password>]
Encrypt and Decrypt password for Analytics installation.
  -e, --encrypt           Passwords will be encrypted (Default)
  -f, --file=<FILE>       Defaults file used by analytics installer
  -g, --generate          Generate new secret key. Printed in console output.
  -h, --help              Show this help message and exit.
  -k, --key=<secret key|file:FILE>
                          Secret key (256-bit) or path to keyfile (file:
                            path/to/file)
  -o, --output=<output>   File output path
  -p, --password=<password>
                          Password for encrypt
  -V, --version           Print version information and exit.
```

The database connections to the MD, DM and EA databases are maintained in this file as JNDI datasources: `server\pentaho-server\tomcat\conf\server.xml`.

JNDI data sources used for analyses and Mondrian are in `server\pentaho-server\tomcat\webapps\console\META-INF\context.xml`. MD and DM database connection are maintained there as ResourceLink. So, the credentials are maintained only once in server.xml.

To encrypt passwords, you must always use the correct master-key.

Be very careful when performing this function manually. The use of an incorrect key or a copying error will cause the installation to fail.

3.12. Backup

3.12.1. Backup Before Deployment of Content

3.12.1.1. BA Server

3.12.1.1.1. Analysis Data Source (Mondrian Schemas)

1. Login to the User Console as an administrator user.
2. Export existing analysis data sources through Manage Data Sources dialog in User Console to the backup location.

3.12.1.1.2. Data Repository

The BA Server should be stopped before preceding the backup of its database repository. The following databases should be backed up:

- usu_si_jackrabbit
- usu_si_hibernate
- usu_si_quartz
- usu_si_operations_mart

3.12.1.2. Data Integration

The following databases should be backed up before every deployment and execution of new ETL jobs (if the default DB names are used):

- usu_si_md
- usu_si_ea
- usu_si_dm

Jobs must not be running during the backup procedure!

3.12.2. Regular Backup

3.12.2.1. BA Server

The BA Server should be stopped before preceding the backup. The backup procedures described in this section should be streamlined to be done at the same time to prevent inconsistency of *pentaho-solutions* and database repositories.

3.12.2.1.1. Web Application and Solution Repository

The following directories should be backed up regularly.

- %SI_BASERVER%/tomcat/webapps/console
- %SI_BASERVER%/pentaho-solutions

3.12.2.1.2. Database Repository

The following databases should be backed up regularly:

- usu_si_jackrabbit
- usu_si_hibernate
- usu_si_quartz
- usu_si_operations_mart

3.12.2.2. Data Integration

The following databases should be backed up before every deployment and execution of new ETL jobs (if the default DB names are used):

- usu_si_md
- usu_si_ea
- usu_si_dm

Jobs must not be running during the backup procedure!

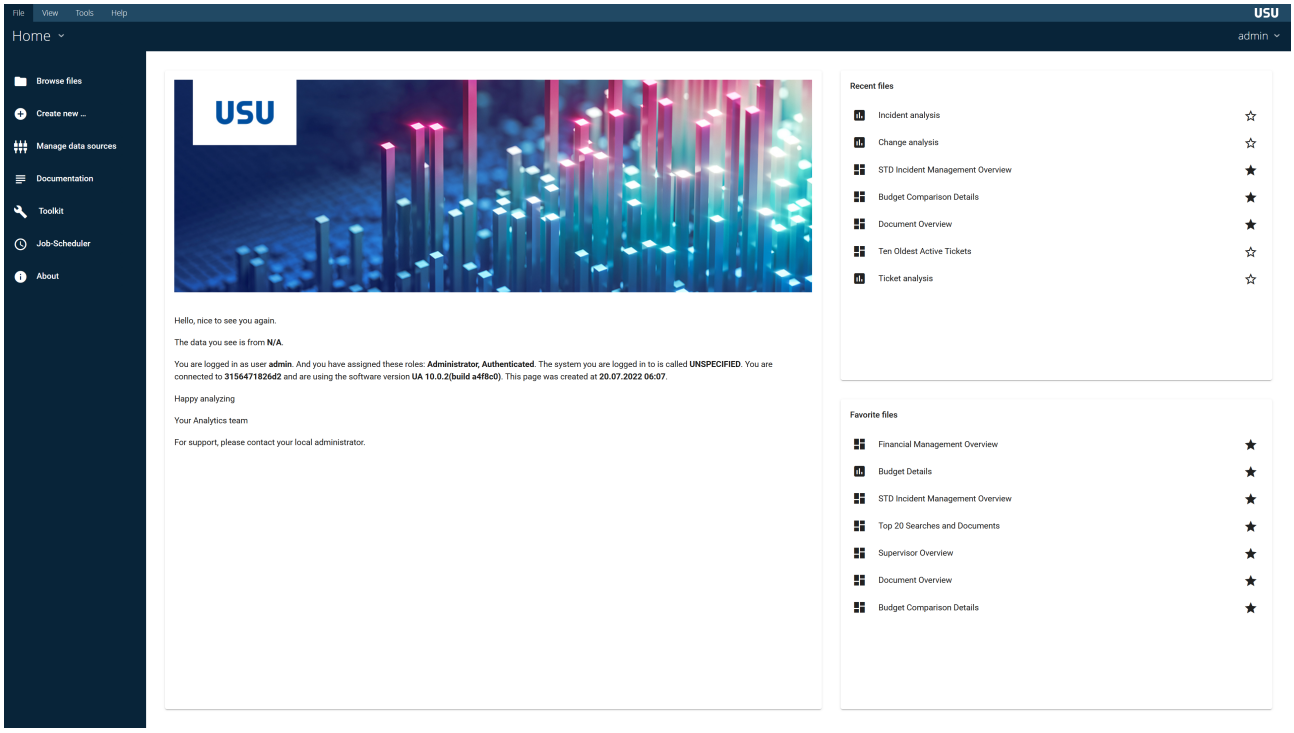
3.13. Monitoring

The monitoring of the solution is highly dependent on the installation scenario.

For more details refer to the *Installation Protocol* document created during the installation.

For data integration monitoring please have a look at chapter Monitor (page 120).

4. Home



When a user logs in to the system the Home screen opens. The screen is organized in so called cards.

On the left side you find a card with buttons. With the appropriate rights you can:

- Browse files
- Create new objects like analyzer report, dashboards and report assembler definitions
- Manage your data sources
- View documentation
- Open Toolkit
- Open Job-Scheduler
- Open About information

On the right you see the **Recent files** and **Favorite files** cards.

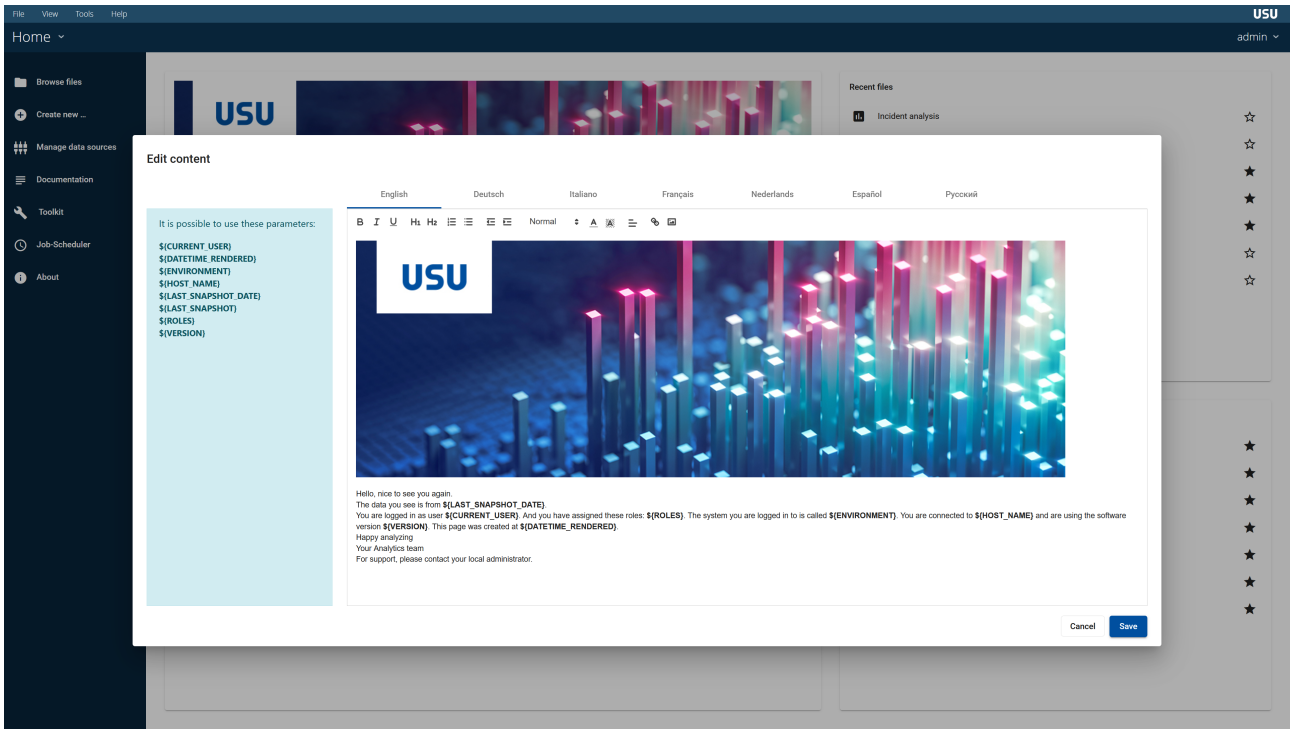
- In **Recent files** you have the objects you have recently opened
- In the **Favorite files** you see the objects you defined as your personal favorites

The card in the middle shows customizable content. For users with administrator rights we provide an editor for the content of this card. Texts can be stored there for all supported languages. To be able to display current system information to the end user, the text can be provided with parameters that are set when the page is created.

We provide these parameters:

- **VMA_CURRENT_USER**: Current logged in user

- **VMA_DATETIME_RENDERED**: Date and time, when this page was created
- **VMA_ENVIRONMENT**: Name of the environment like stored in the system configuration
- **VMA_HOST_NAME**: Name of the host you are connected
- **VMA_LAST_SNAPSHOT**: Internal name of the actual snapshot
- **VMA_LAST_SNAPSHOT_DATE**: Date and time, when the actual snapshot was loaded
- **VMA_ROLES**: Comma separated list of all roles assigned to current logged in user
- **VMA_VERSION**: Software version



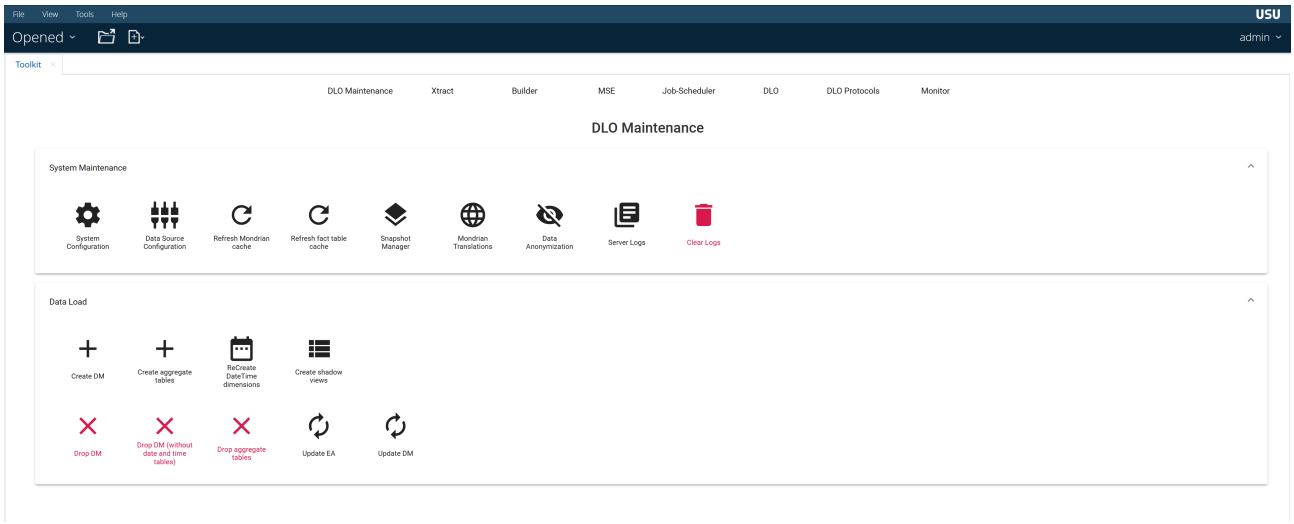
To change the content in the middle card, follow these steps:

1. Log in as a user with administrative rights.
2. Move the mouse over the card in the middle of the home screen. An Edit icon appears in the upper right corner of the card. Click on it and an editor opens.
3. Change the text to your needs. You can use build in parameters, which are replaced during creation of the page. You find a list of possible parameters in the blue area on the left side of the editor screen.
4. Save your changes

5. Toolkit

5.1. General

The Toolkit is a collection of tools to configure, maintain and customize your data integration. It also provides you with the possibility to start data integration and maintenance processes from the application. It is integrated into User Console. Only users with assigned Administrator role have access to this module.



To access it:

1. Log in to User Console as a user with Administrator role assigned.
2. Click **Tools** and then select **USU Toolkit** from the menu.
3. The Toolkit **Home** screen opens.
4. From here you can access all tools by clicking the tiles or by using the navigation bar in the upper part of the application.

The tools you can use are briefly:

- **Maintenance**: The main entry point to Toolkit. Start data integration and maintenance processes from application.
- **Xtract**: Define what you want to extract from your source systems.
- **Builder**: Define your analytical database structure and the data integration process.
- **MSE** (Mondrain Schema Editor): Add, edit or delete your Mondrian schemas.
- **Job-Scheduler**: Open the Job-Scheduler application from here.
- **Monitor**: See what's going on - what data integration jobs are executed and their status.

5.1.1. Basic Skills and Knowledge

To use Toolkit properly the responsible user must be familiar with some technical terms and concepts:

- Basic Business Intelligence knowledge is particularly advantageous. Terms and concepts like star schema, fact, dimension, degenerated dimension, measure and ETL should not be loanwords for you and are assumed for further comprehension.
- You should be able to understand and write SQL select statements.
- A good knowledge about the data structure of your source systems is beneficial.

5.1.2. Setup Toolkit

The following steps describe how to setup a standard Toolkit. Information how to implement the steps you find in the mentioned chapters. We assume that you have performed a complete installation as described in *Installation Manual*.

Steps to setup a standard Toolkit:

1. Check preferences (see chapter Before You Get Started (page 73))
2. Change system configuration parameters (see chapter Change System Configuration Parameters (page 73))
3. Change data source configuration (see chapter Change Data Source Configuration parameter (page 73))
4. Import Xtract definitions (see chapter Import Xtract Definitions (page 74))
5. Import builder definitions (see chapter Import Builder Definitions (page 74))
6. Execute jobs (see chapter Execute Jobs from Maintenance Screen)
7. Extract SD to EA
8. Create shadow views
9. Create DM
10. Fill date and time dimensions
11. Load DM
12. And many more...
13. Schedule execution (see chapter Schedule Execution)

5.1.2.1. Before You Get Started

Please make sure that Toolkit binaries are installed on a BA server (for details see *Installation Manual*)

You need access to User Console as a user with Administrator role assigned. Ask your local administrator for permission.

5.1.2.2. Change System Configuration Parameters

Change parameters to connect to User Console in System Configuration. They are e.g. needed to refresh Mondrian cache.

For more information about System Configuration, please see chapter System Configuration (page 75).

5.1.2.3. Change Data Source Configuration parameter

Change parameters to connect to the source database using the Data Source Configuration tool.

For more information about Data Source Configuration, please see chapter Export Old Configuration.

5.1.2.4. Import Xtract Definitions

The system ships with a set of standard Xtract definitions. You find it in your installation package.

```
<installation package>\si_toolkit\business_modules\<business module>\si-xtract.xml
```

Please login to Toolkit, open Xtract module and import the file. Please find a detailed description how to do this in chapter Import Xtract Definition (page 93).

5.1.2.5. Import Builder Definitions

The system ships with a set of standard Builder definitions. Because Builder is dependent on DBMS specific SQL commands, we deliver a ZIP archive per supported DBMS system and business module.

You find the ZIP archive in your installation package.

- **MS SQL Server:** `<installation package>\si_toolkit\business_modules\<business module>\si-builder\mssql\si-builder.zip`
- **MySQL:** `<installation package>\si_toolkit\business_modules\<business module>\si-builder\mysql\si-builder.zip`
- **Oracle:** `<installation package>\si_toolkit\business_modules\<business module>\si-builder\oracle\si-builder.zip`

Please login to Toolkit, open Builder module and import the file for the DBMS and business module you are using. Please find a detailed description how to do this in chapter Import Builder Definition (page 113).

5.1.3. Basic Functionality

In Toolkit we provide you with some basic functionality.

To make it easier to find entries in a grid a full text search is implemented.

- Type in the field **Quick search...** the string you are searching.
- The system will search for the very string as substring in all fields of the grid.
  Placeholders like *, _ or % are not supported.
- The search is executed immediately after you type in.
- The search is not case sensitive.

You can sort the columns of a grid by clicking in the header. An arrow indicates how the column is sorted.

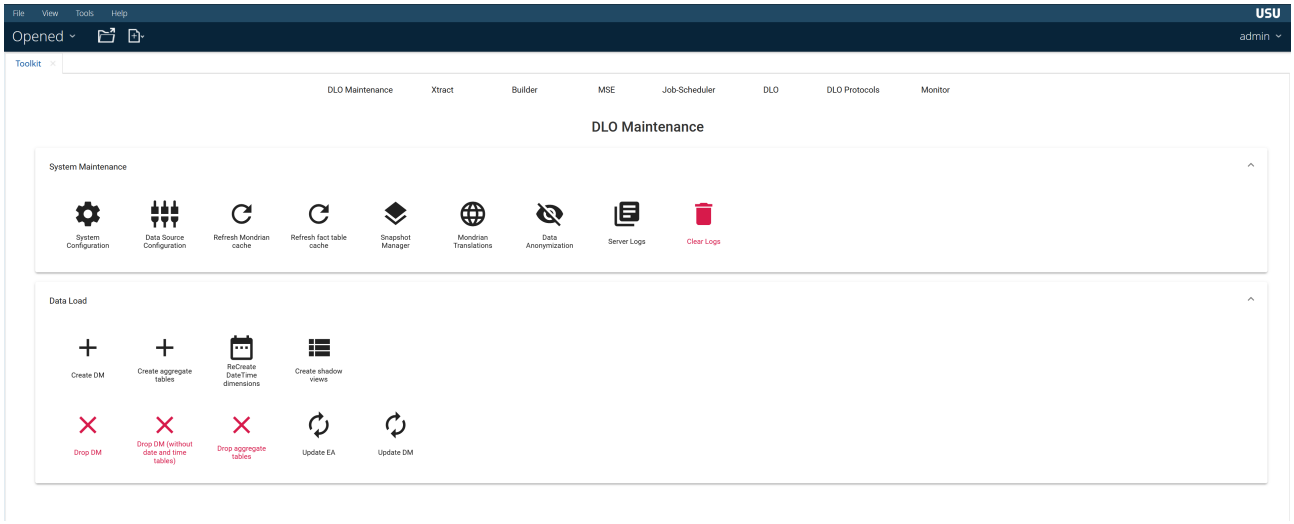
- **Arrow up**: Ascending sort order.
- **Arrow down**: Descending sort order.
- **No arrow**: Standard sort order.

You can sort multiple columns by pressing the Shift key while clicking in the column header.

System messages like **general information**, **feedback**, **warnings** or **errors** are viewed in an extra window or they are part of a modal window.

- **General Information**: Information about application, process and general usage information. Shown in blue.
- **Feedback**: Information about successful results. Shown in green.
- **Warnings**: Shown in orange.
- **Errors**: Shown in red.

5.2. Maintenance



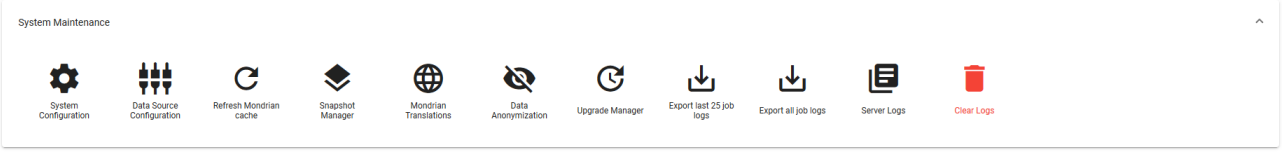
5.2.1. Overview

The Maintenance screen provides you with the ability to start global jobs on application level and maintain the system.

The Maintenance screen is organized in the following sections:

- System Maintenance
- Monitor
- Data Load

5.2.2. System Maintenance



5.2.2.1. System Configuration

On the System Configuration page, a system administrator can set various system properties. These parameters control the general behavior of the application.

General Parameters	
Transaction key	Transaction Key to be used for the next SD to EA extraction (possible values: auto"/"<customer_key>_<yyyyMMdd>_<HHmmss>")." Default value: auto

General Parameters	
Number of threads used	Maximum number of parallel sub jobs running (number of threads). Default value: 4
Load these domains into EA	List of domains to consider when processing data from SD to EA (e.g. “INM, ASM” or “ALL” for all domains). Default value: ALL
Load these domains into DM	List of domains to consider when processing data from EA to DM (e.g. INM, ASM or ALL for all domains). Default value: ALL
Value for unassigned dimension members	Default value for unassigned dimension members. Default value: -unassigned-
Value for unknown (NULL) values	Default value for unknown (NULL) values. Default value: -unknown-
Start date for calculating the date dimension	Start date for calculating the Date dimension. Default value: 1995-01-01
Number of years to be created in the date dimension	Number of years to be created in the Date dimension. Default value: 30
Language	If you use the ${SI_LANGUAGE} variable in an SQL statement, it is replaced with this value at runtime.
Default language for Mondrian schemas	Abbreviation of the language (according to RFC 3066) used as the default language for Mondrian schemas. The default language is used for users who use unsupported languages.
Gather statistics in EA	Set the value to true if the Gather Statistics function is to be executed on the EA database after each load from SD to EA.
Create indexes in EA	Specifies whether indexes are to be created during the CREATE EA job. It has no influence on the UPDATE EA job (indexes are always created during this job).
Gather statistics in DM	Set the value to true if the Gather Statistics function is to be executed on the DM database after each load from EA to DM.

Fact Table Load Parameters	
Initial snapshot date	Date and time of the first snapshot in a fact table (format: "yyyy-MM-dd HH:mm:ss").
Time granularity	Time granularity of a fact table (allowed values: minute/hour/day/month/year).
Time multiplier	Multiplier of the time-based granularity of a fact table (allowed values: positive integer).
Number of snapshots	Maximum number of time-based snapshots to be held in a fact table ("moving window").
Snapshot enumeration	Comma-separated list of snapshot time stamps to use in addition to the generated values.

For more details about these parameters please see chapter Description of Fact Table Related Parameters (page 184).

Toolkit	
Roles with access to Toolkit	Comma-separated list of user roles that have access to the Toolkit.
Roles with editor rights	Comma-separated list of user roles that can edit reports and dashboards.
Roles with admin access to Toolkit	Comma-separated list of user roles that have administrator access to the Toolkit.
Roles with admin access to Job-Scheduler	Comma-separated list of user roles that have administrator access to the Job-Scheduler.
Roles with access to Job-Scheduler	Comma-separated list of user roles that have access to the Job-Scheduler.
USU Service Management roles	Comma-separated list of USU Service Management groups that have access.
Environment	The value can be used to distinguish between DEV, TEST or PROD environments, for example. The information is displayed at the bottom of the Maintenance screen in toolkit. If the value is empty, UNSPECIFIED is displayed.

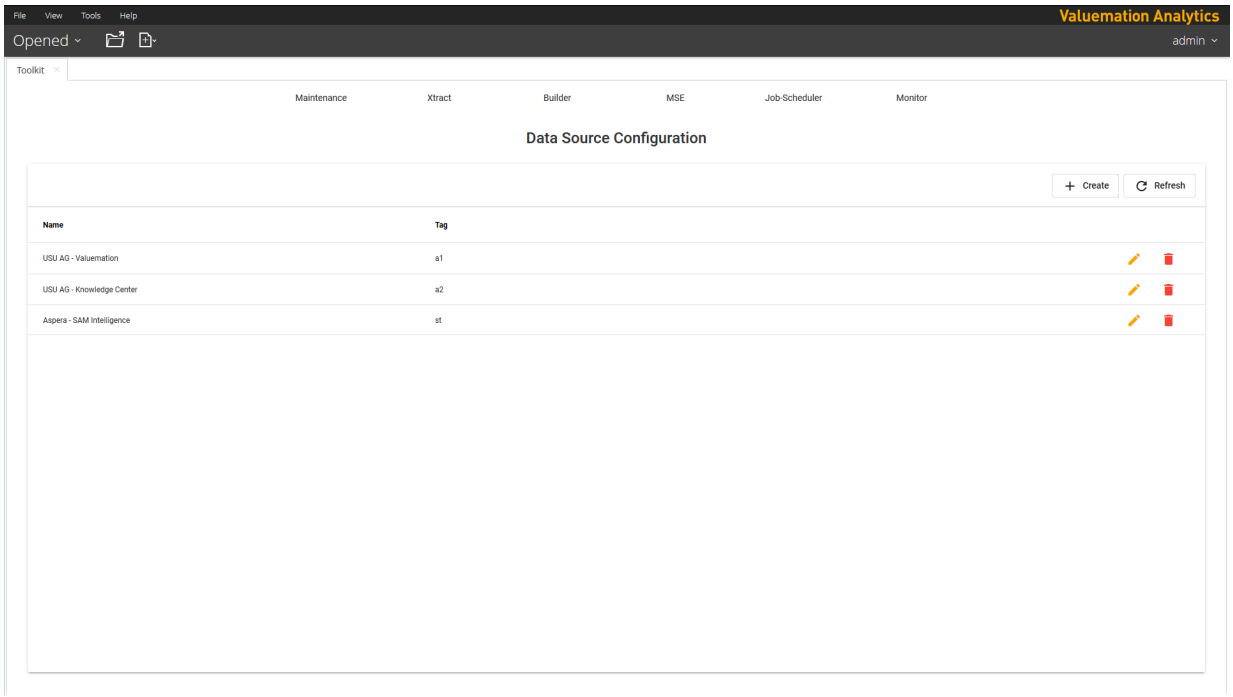
User Console Parameters	
User console URL	User Console (server) URL.
User console user	User Console (server) credential.
User console user password	User Console password.
Refresh Mondrian schema	Name of Mondrian schema, which should be refreshed with job Refresh Mondrian Schema. If not specified, all Mondrian schemas are refreshed.
SSO HTTP header	Name of HTTP header with username or type of SSO used in User Console.
SSO timeout	Timeout applied when validating key used by encrypted key SSO.
SSO anonymous user	Name of user for anonymous SSO.

Expert Parameters	
SQL statement (authorization policies)	SQL statement which is used to load authorization policies. SQL statement must have columns role, schema, cube, hierarchy and member.
Delete generated protocol records older than ... days	Number of days. JobAudit and GlobalJob that are older than this number of days are deleted. Zero or empty field means that all records will be deleted.
Email address (report job failed)	An email address to which an email is sent when a report job fails.
Email address (ETL job failed)	An email address to which an e-mail is sent when a scheduler job fails.
Staging mode database type	Database type used for staging area tables. If parameter is empty and staging area is enabled, the database for staging area will be the same as for data mart.
Staging mode	Set to true to enable staging mode.
USU Service Management authentication	Enable/disable USU Service Management authentication.
Export old configuration	Extract old parameters and insert them into data sources.

Mail Server Settings	
Host Name (SMTP)	Host name (SMTP) of you mail server
Port	Port, default is 25
Server Type / Protocol	Select SMPT or SMTPS
Email "From" Address	Email adress, which should be shown as from address.
Use Authentication	Use authentication (user/password below)
User Name	User Name for the email account you want to use
Password	Password for the email account you want to use
Use SSL	Use SSL
Use Start TLS	Use Start TLS

BETA Functionality	
Changing entries	We place here triggers for BETA functionalities. If you check the checkbox the functionality is activated.

5.2.2.2. Data Source Configuration



This tool is used to create, edit, configure and delete data sources.

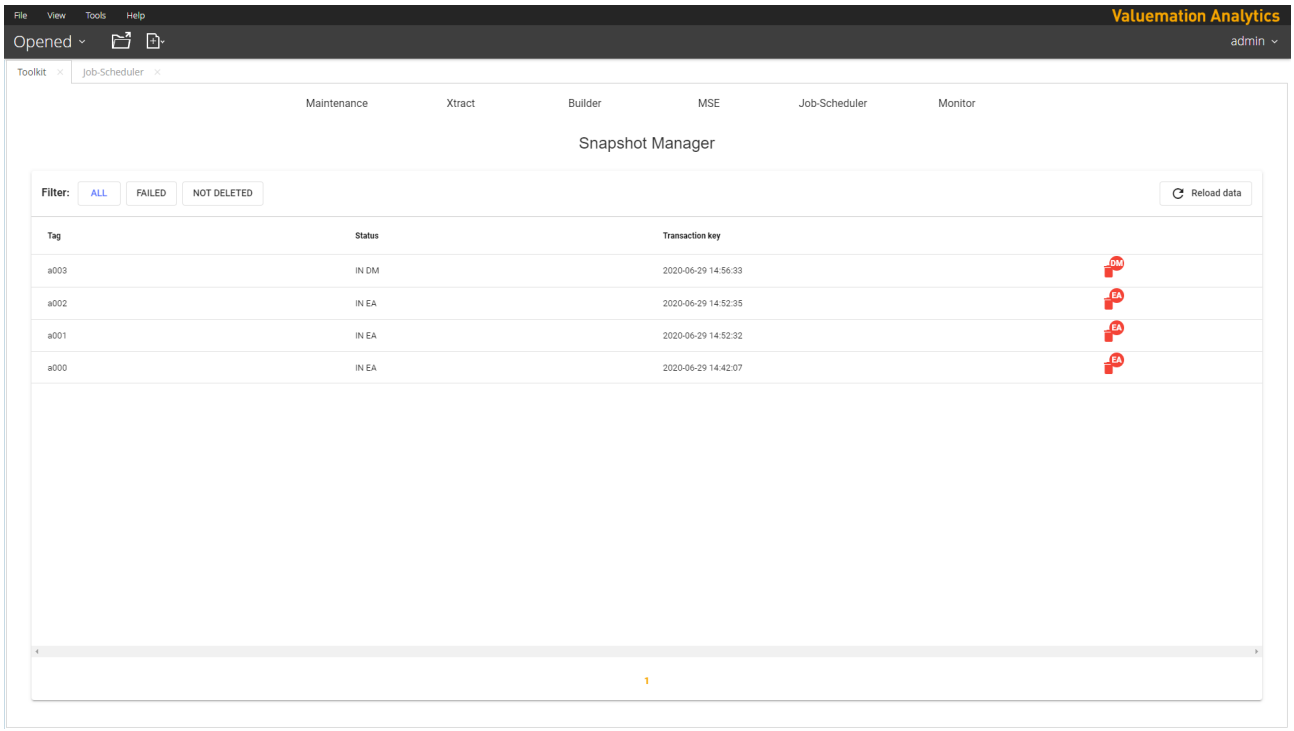
Feature	Description
Create	If you click the **Create** button, a modal form opens. You can add the name and the tag of the data source and the database connection to the data source. Specify the database type, credentials, server name and port. You can also provide the connection data as JDBC string. With the **Test connection** button, you can test the connectivity to the server, user and password you provided in this data source.
Edit	If you click the **Edit** button, a modal form opens. You can edit the name of the data source and the database connection to the data source. Specify the database type, credentials, server name and port. You can also provide the connection data as JDBC string. With the **Test connection** button, you can test the connectivity to the server, user and password you provided in this data source.
Delete	To delete a data source, click the **Delete** button. Answer the security question with Yes and the data source will be deleted. ⚠ Together with the data source the Xtract definition associated with data source is deleted. Deletion of a data source and with it the Xtract definition is irreversible.

5.2.2.3. Refresh Mondrian cache

After data is loaded to data mart database you should refresh Mondrian cache to make sure the end user sees the actual data.

When you click this button one job is scheduled. The job connects to the BA server with the user specified in System Configuration parameters in section User Console Parameters (User console URL, User console user, User console user password) and refreshes the cache for the defined Mondrian schema.

5.2.2.4. Snapshot Manager



Snapshot Manager gives you an overview about the snapshots you have stored in the system and their status. The tool enables you to change the transaction key and the status of a snapshot or delete it completely.

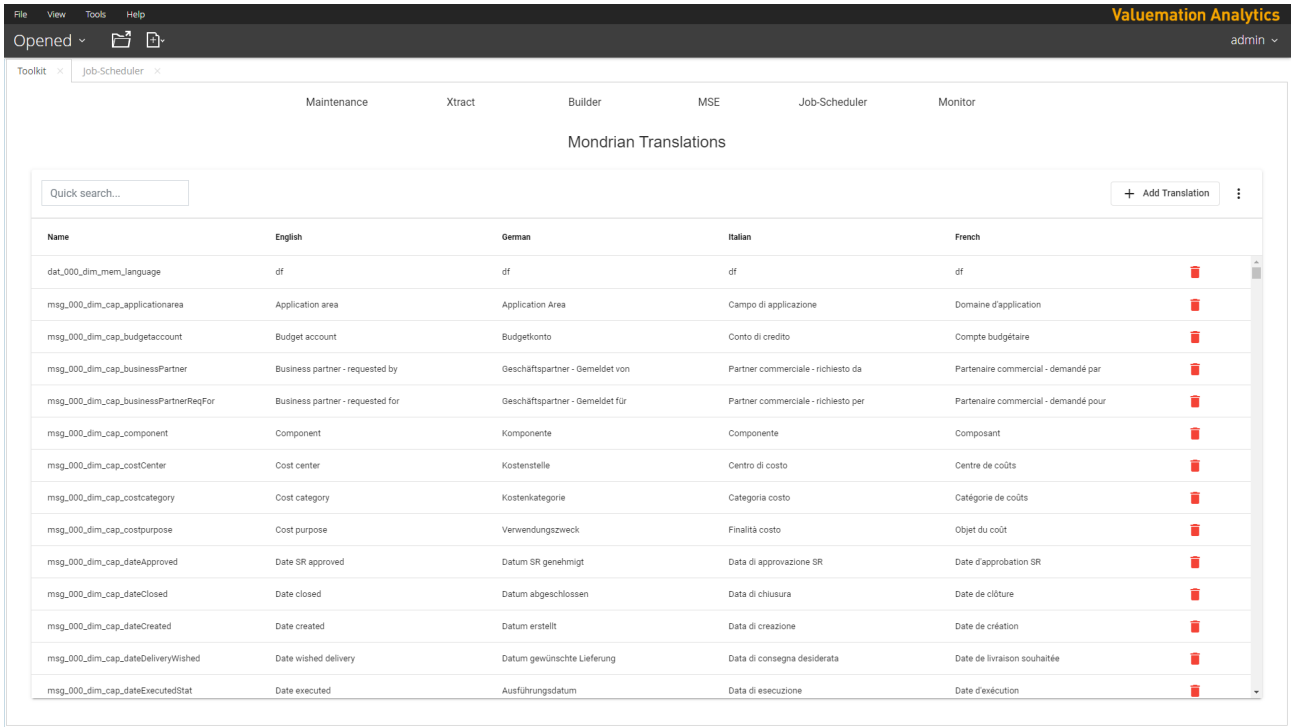
Feature	Description
Filter	We added the possibility to filter snapshots by status. Just click the button and the filter is applied.
Change status	In some corner cases, it could be necessary to change the status of a snapshot. To do this click in field **Status** and chose a status from the list.
Change transaction key	In some corner cases, it could be necessary to change the transaction key of a snapshot. To do this click in field **Transaction key** and chose a value from the date picker.
Drop in EA	When you click this button, a global job is scheduled, that drops the database tables associated with the snapshot from EA.
Drop in DM	When you click this button, a global job is scheduled, that deletes this snapshot from DM.

Changing of the status or transaction key has direct impact on the load process. It may happen that something will not be loaded.

If you click Drop in EA the deletion of the extraction area database tables is scheduled immediately. It is not possible to cancel this process.

All tables of this snapshot will be dropped.

5.2.2.5. Mondrian Translations



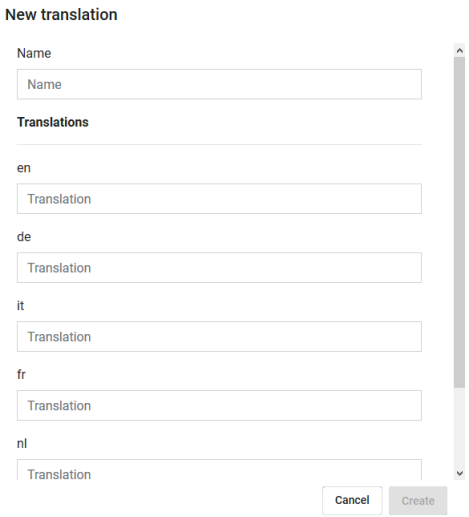
Localization of Mondrian was done using localization properties files, which are stored on the BA Server. To change those files, you had to access the server infrastructure, change the files and restart the BA service. Only after restart the changes took effect.

The new module Mondrian Translations gives you the possibility to maintain your translations using an integrated user interface. You can add manually, but also by extracting from a Mondrian schema in Mondrian Schema Editor application.

Default translation language for Mondrian cubes can be specified in System Configuration in parameter **General Parameters / Default language for Mondrian schemas**. The Mondrian cube is translated with default language when a user uses a language, which is not supported (e.g. user uses browser with Russian language which is not supported as a standard).

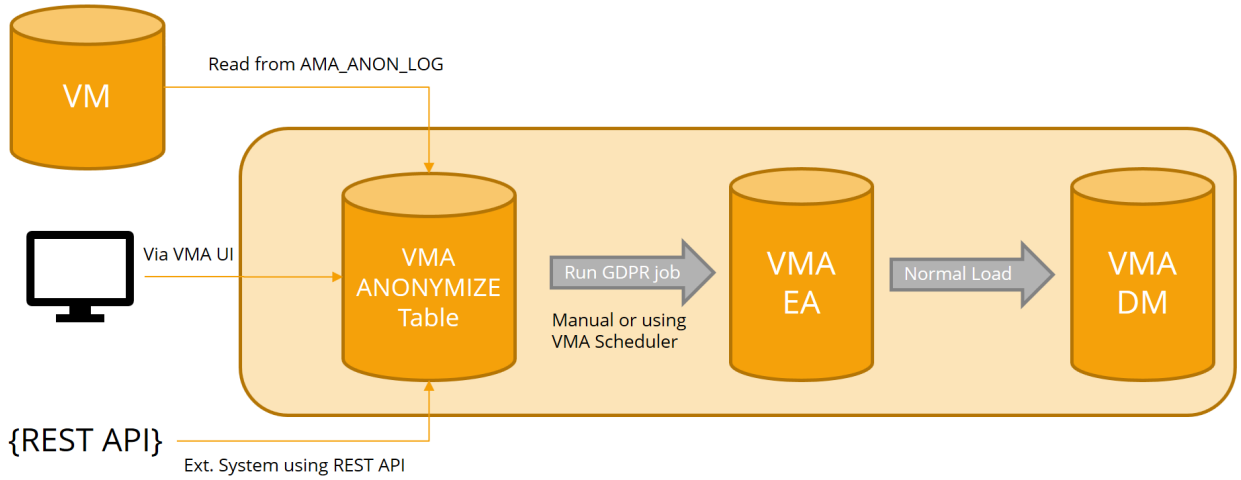
It is also possible to import and export the keys and translations in XML format.

Localizations from Mondrian Translations are automatically applied in Mondrian schemas which use the parameter DynamicSchemaProcessor=de.usu.si.mondrian.dsp.VMASchemaProcessor. No service restart is needed. You just refresh the Mondrian cache, and the values are changed.

Feature	Description
Add Translation	If you click the button Add Translation a modal form opens.  You can add the key (name) and the translations for the supported languages.
Delete translation	To delete a translation, click the red cross you find at the end of each line in the grid. Answer the security question with Yes and the record is deleted.
Show only empty values	Open the **Options** menu. When you check **Show only empty values** you see only records in the grid, which have not translated values.
Import translations from properties files	Choose **Import translations from properties files** from **Options** menu. A modal form opens, where you can select the ZIP archive with properties files you want to import, thus all relevant properties files must be put in the ZIP archive. Click **Import** and the system will import the values. ⚠ Even if it is possible to import the properties files during an upgrade, this only makes sense for an initial import. Do not use this functionality in your normal work with the system. You could overwrite your correct translations. Old versions cannot be restored.
Import translations	Choose **Import translations** from **Options** menu. A modal form opens, where you can select the XML file you want to use. You must use an XML file which was exported by Export translations functionality. Select the desired mode to be used in case of merge conflicts. • Overwrite existing data: Values from import file will be used. Data in database will be overwritten. • Delete deprecated entries: Values, which are not in the import file are deleted from database. Click **Import** and the system will scan the XML file for keys and add the new values to the grid.
Export translations	Choose **Export translations** from **Options** menu. The system asks you, where to save the translation XML file. After the selection of the destination folder the export starts.
en, de, it, fr, es, nl	To choose the languages you want to see in the grid open Options menu and check or uncheck a language from the list.

5.2.2.6. EU-GDPR Module – Data Anonymization

The new EU-GDPR module supports you in the targeted anonymization of your data stored in the analytical database.

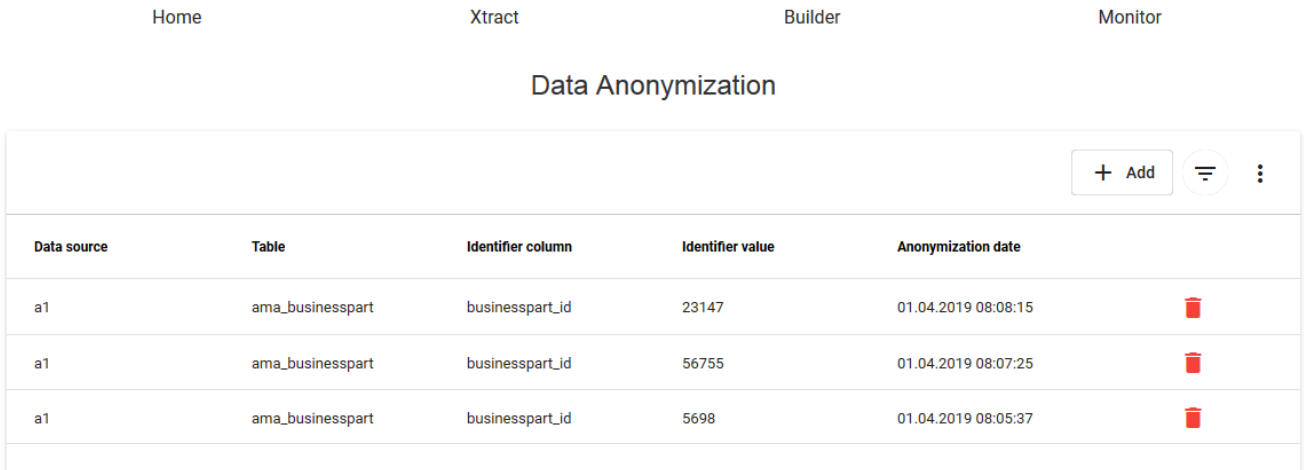


You use the user interface to define which data records and which data you want to make anonymous. The system supports you with lists of possible tables and fields and with data-type-dependent random values for anonymization. Anonymization of the complete database of your snapshots is performed manually or time-controlled.

If you have already imported data into USU Service Management , you can import the anonymization instructions from there. You can also apply these definitions manually or time-controlled to the complete dataset of your snapshots.

Here is an example how to handle GDPR module in Toolkit. The GDPR module is a part of the Toolkit. You can find it on the Maintenance screen in the *System Maintenance* panel as *Data Anonymization* button.

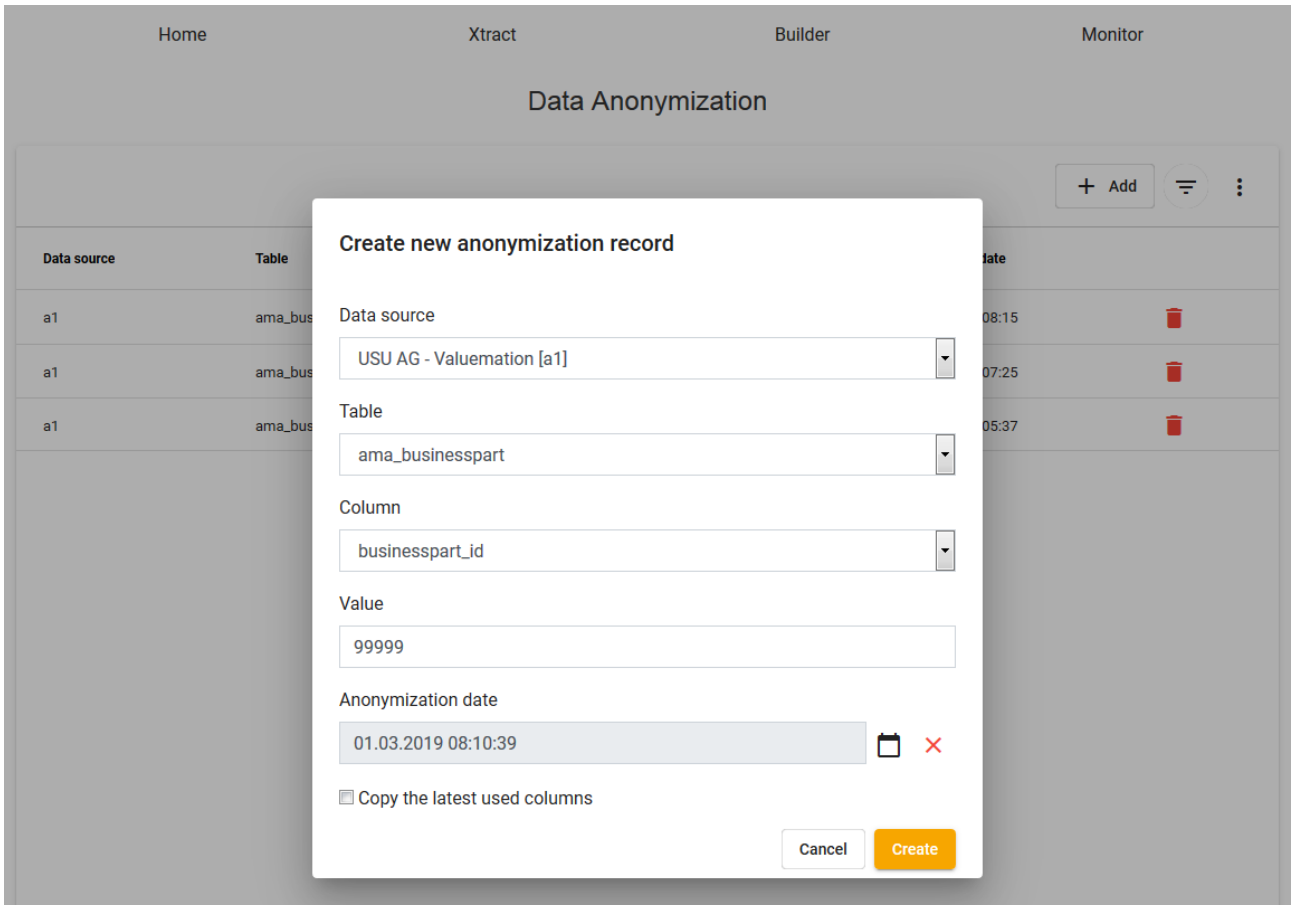
On main screen is an overview of the *GDPR Anonymization* data records.



Button	Description
Filter	For filtering the content. The settings will be explained in example below.

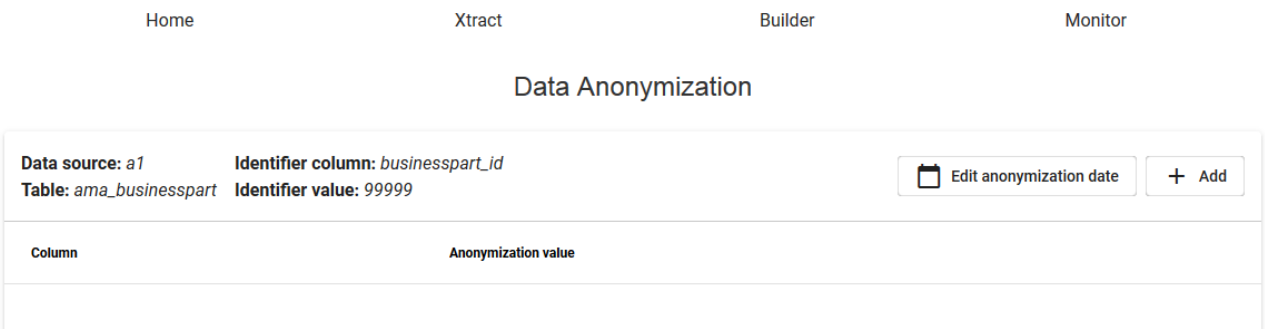
Button	Description
Add	Add a new Anonymization record to the record list. Clicking on the button you get this Create new anonymization record form displayed on the next figure.
Options	You get other options how to operate the anonymization process. Each option will be explained below.

Create new anonymization record form after the Add button click.



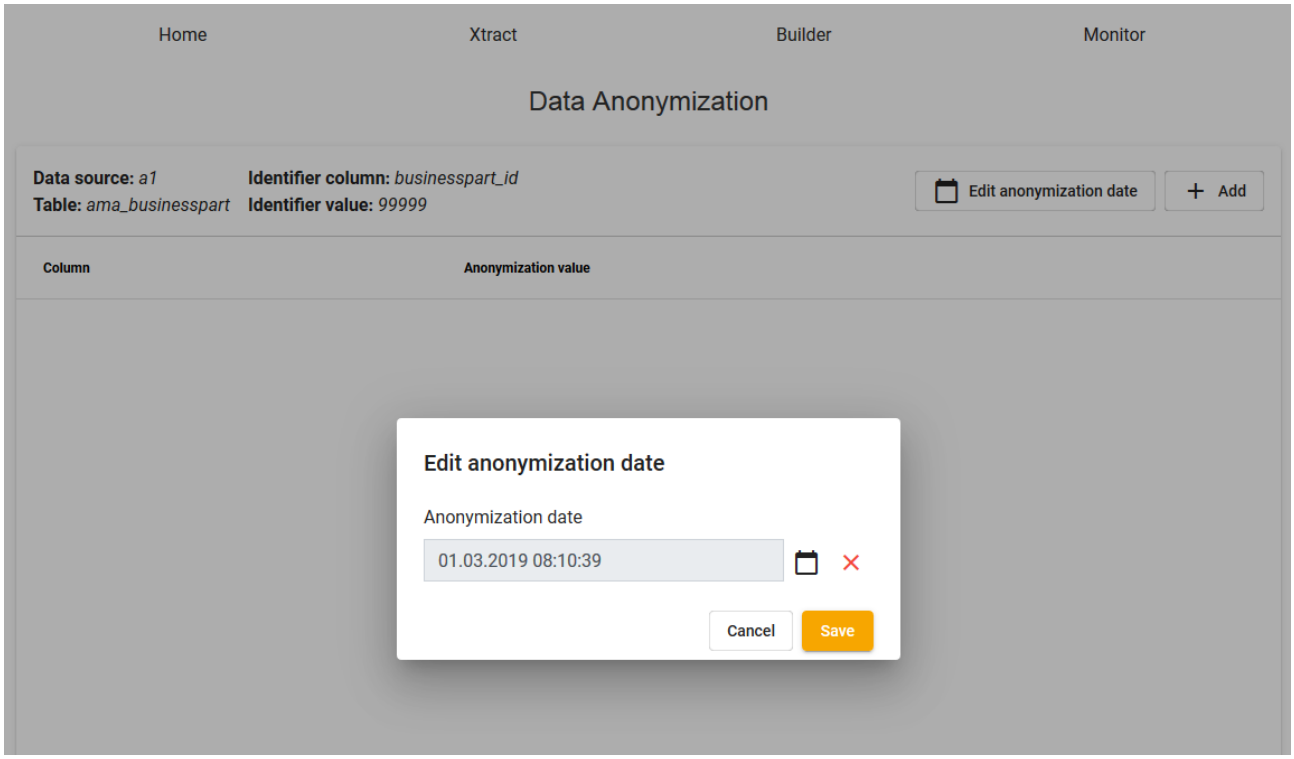
Field/button	Description
Data source	The data source is one of Toolkit data sources (a1, a2, etc.).
Table	According chosen Data source you get the table list and can select the table from the list.
Column	In the field Column it is needed to select primary key column of the table.
Value	It is needed to specify in attribute Value primary key identifier of entity which will be anonymized.
Anonymization date	You can specify the anonymization date. This value defines the date of load till when the data must be anonymized. If you keep anonymization date empty, Data Anonymization module will anonymize all currently loaded snapshots and all snapshots which will be loaded in future.
Copy the latest used columns	This checkbox is not used in this example, so we will explain the point of the feature below for specific example.

Clicking on Create button you get empty anonymization record column overview.

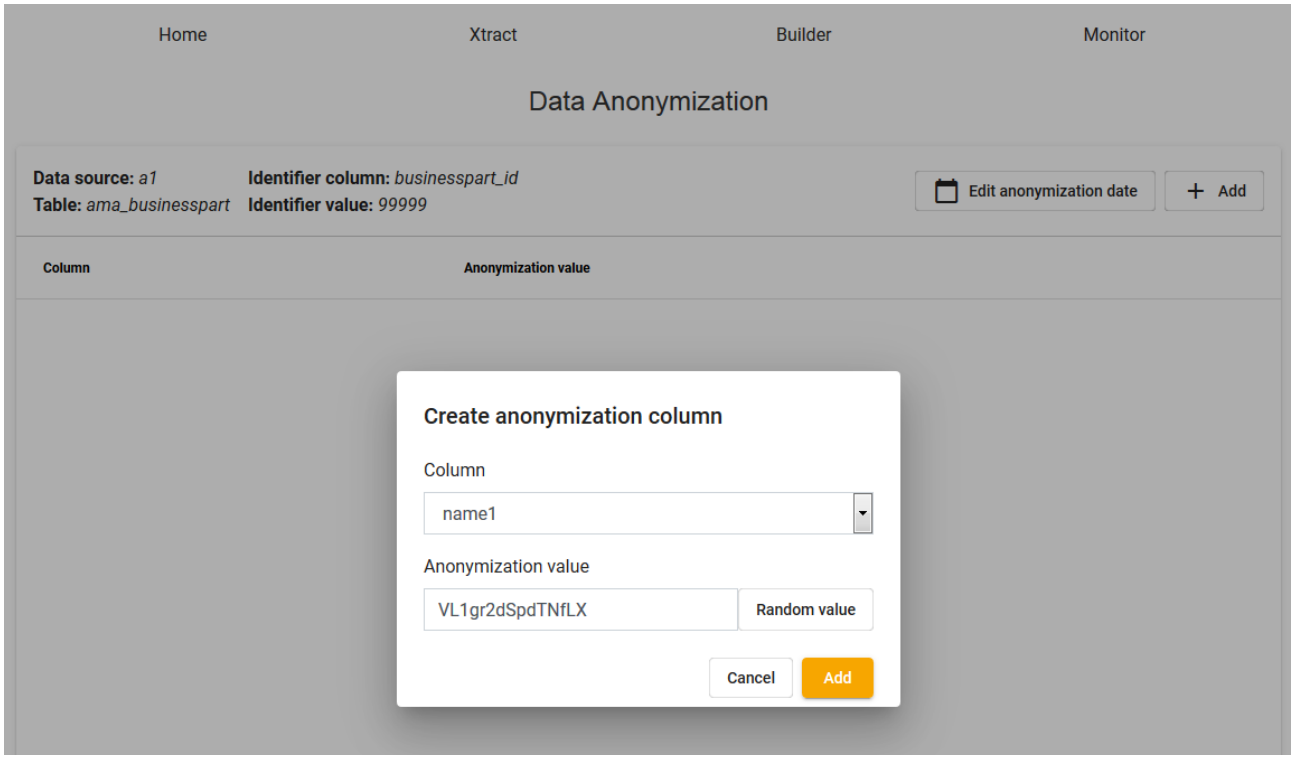


Button	Description
Edit anonymization date	If you want to change the anonymization date, click on this date button. It is possible to do it for each anonymization table record separately. The figure below is showing the form.
Add	Here you can specify column and values in the table which will be anonymized. Clicking on the Add button you get a form (displayed below) where you can select the table column and enter new value which will be used instead the current value in the column. It is also possible to let the system generate the new value automatically. You can select more columns for the table record sequentially. It depends on the table and the Xtract configuration in Toolkit.

Edit anonymization date form.

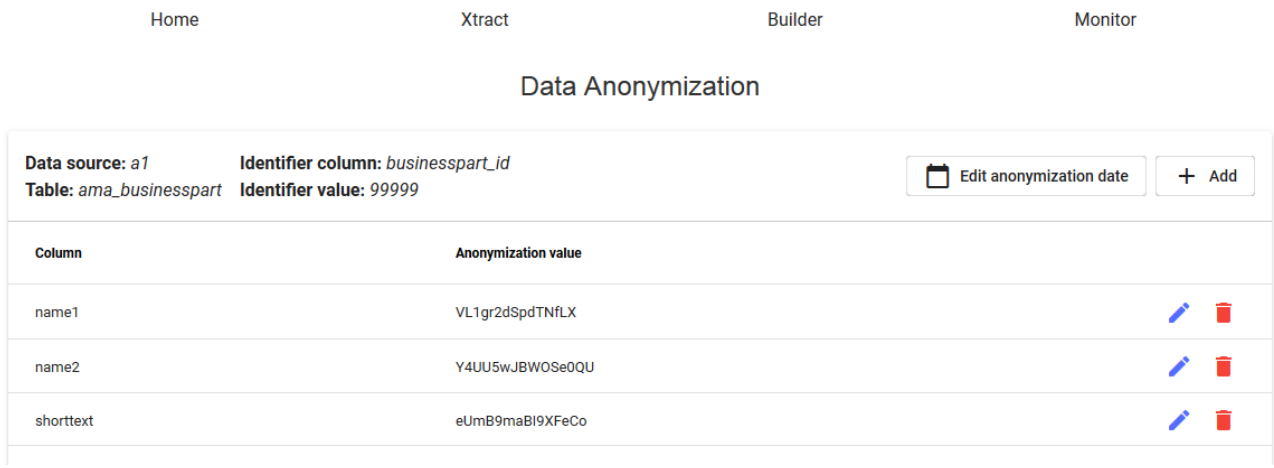


The form after Add button click.



Field/button	Description
Column	The select box offers columns for specific table. These columns values can be anonymized.
Anonymization Value	The input field is intended for directly writing. You can write your own value, or let the system generate a unique value.
Random value	The button intended to generate the value by the system.

After filling and confirming the form by clicking on *Add* button you get the record in the record column overview. In *Actions* column you can see the possibilities you have when you need to handle the record. For returning to the table records list (previous overview) you can go by *Back* button.



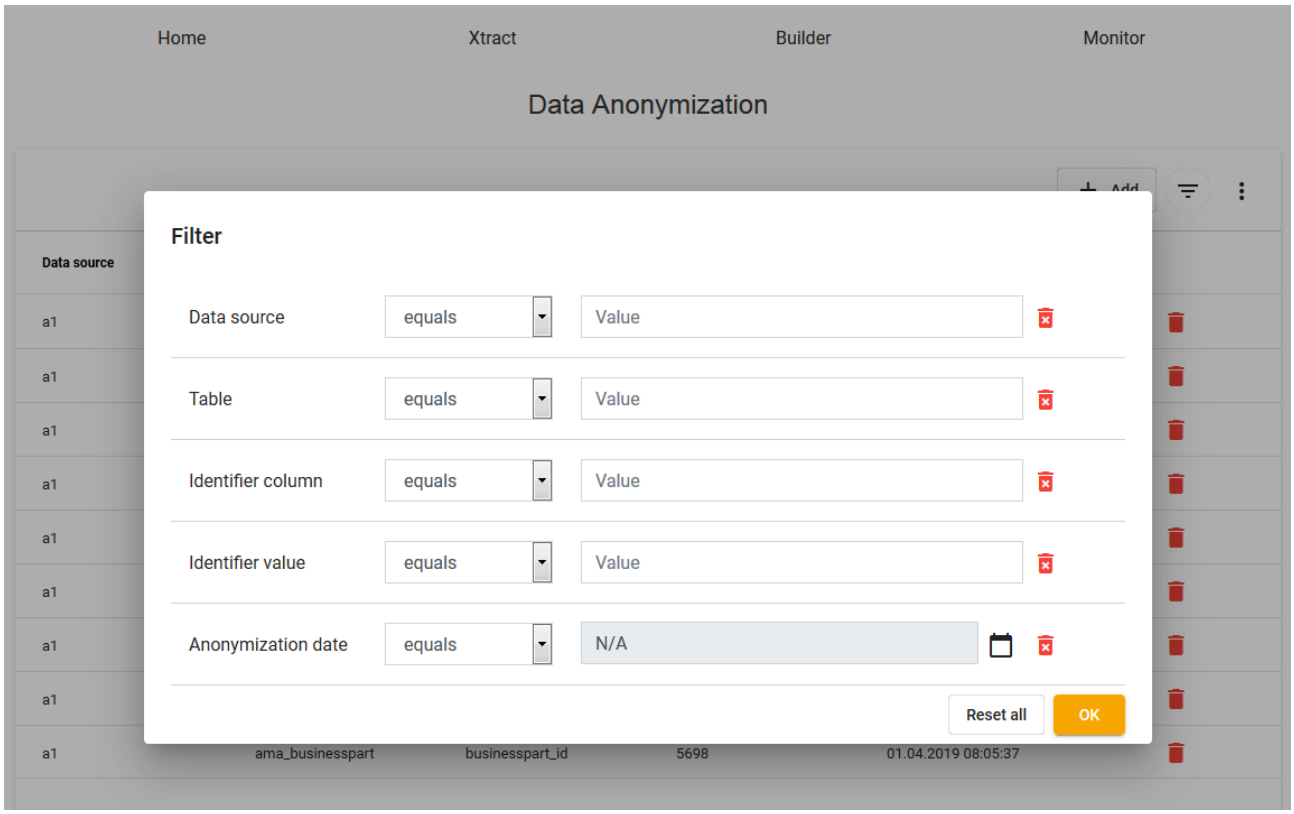
Now let's go to create next record for the same table, but for another data record. We show you how you can manage the anonymization records easier and faster. Let's fill the *Create new anonymization record* form for new record (with another *Value* identifier), but for last used table.

Notice, that now we have selected *Copy the latest used columns* checkbox. This option will offer the last used combination of columns for the specified table.

In the example, we set the anonymization instructions for the client, city and street columns last time for the table. Because we checked *Copy the latest used columns,* you get column list for the record and you save time, because there is no need to select these columns again.

All what you need now is just change the values for the column (now are empty) with *Edit* button.

But, let's go back to *GDPR Anonymization* records overview. You can use the *Filter* button to restrict the amount of displayed records. This example displays the combination of all filter condition.



If you have set the Filter, you are noticed about this fact with *active* flag displayed on the Filter button. When you click on the button and reset the filter, you set the filter off.

The Option button is covering next possibilities for anonymization functionalities.

Option	Functionality description
Refresh	The Refresh option retrieves the up-to-date records list.
Synchronize with VM	The Synchronize with VM is the option to synchronize anonymization instruction with the application. All records which were anonymized in USU Service Management will be automatically copied to Data Anonymization module. Values entered by user manually are not being deleted or overwritten after running the synchronization with USU Service Management . This functionality is available in USU Analytics from version 5.1.
Run anonymization	The Run anonymization option generates the Toolkit job which starts the anonymization process under all existing snapshots in EA managed by Toolkit.

5.2.2.7. Server Logs

With this tool you can download the tomcat server logs from application. All files located in folder */vma/server/pentaho-server/tomcat/logs* are displayed in a table and you can download the files.

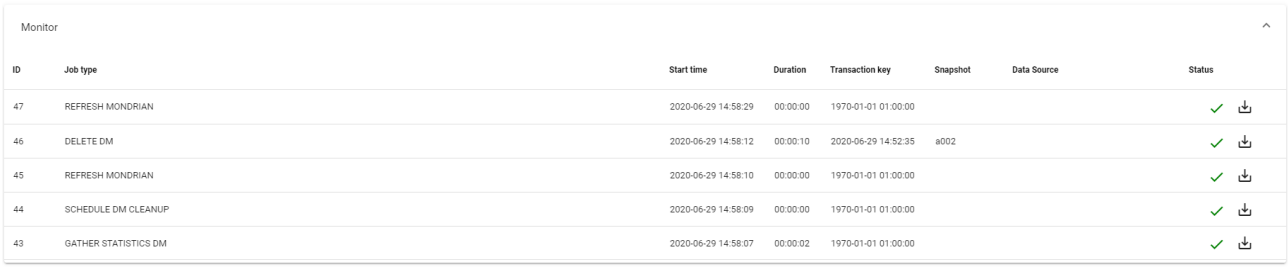
With the *Quick search* you can search for a specific file. By clicking on the column headers you can sort by this column.

5.2.2.8. Clear Logs

Log and monitor tables could grow very fast. So, we added the possibility to clear log and monitor by using the Maintenance user interface.

In System Configuration we added a new parameter called **Delete generated audit records older than ... days**. If you set this to e.g. 30 and run the clean task, all protocol and monitor records older than 30 days will be deleted. 0 (zero) deletes all records.

5.2.3. Monitor

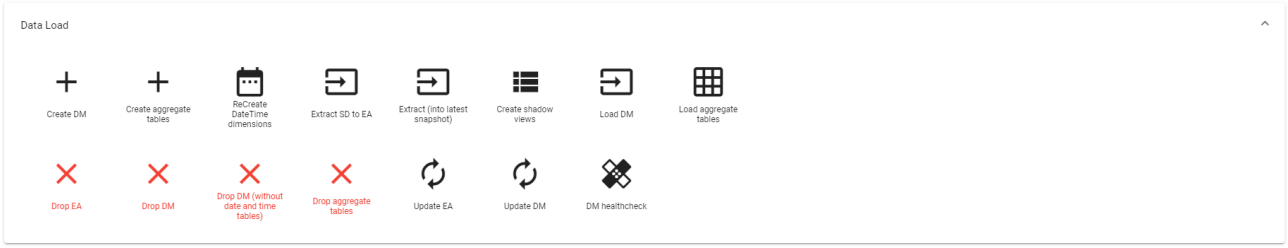


In Monitor section in Maintenance module basic information about global jobs is listed for the latest five global jobs:

- Job ID: Internal system generates ID of the job
- Job type: Name of the job
- Start time: Date and time when execution started
- End time: Date and time, when the execution ended
- Transaction key
- Snapshot tag:
- Status: Execution status of the global job
- If you click on this icon you can download the log information for this global job

The grid is refreshed automatically every second.

5.2.4. Data Load



5.2.4.1. Create DM

When you click this button one job is scheduled. The job uses the chosen data source, connects to data mart database and creates database structure like defined in Builder.

5.2.4.2. Create aggregate tables

When you click this button one job is scheduled. The job uses the chosen data source, connects to data mart database and creates database structure for aggregate tables like defined in Builder.

5.2.4.3. ReCreate DateTime Dimensions

After a clean installation the date and time dimension tables, dim_date and dim_time, do not exist.

When you click this button two jobs are scheduled. The jobs create the tables on data mart database and insert data into the date and time dimension tables.

For the time dimension the system will always create a full set of records – one record per second.

For the date dimension, you must set two System Configuration parameters in section General Parameters:

- Start date for calculating the date dimension
- Number of years to be created in the date dimension

5.2.4.4. Create shadow views

When you click this button a job is scheduled, which creates views in extraction area database for all tables extracted. We call them shadow views. You need them e.g. to test your SQL statements in the Builder forms.

The views have the same name as the table in source database with data source tag as prefix, e.g. if the data source tag for the source you are importing from is defined as a1 and the table from source database is ama_service, the shadow view created is a1_ama_service. The views have the same structure as the underlying tables and are bound to the latest snapshot available in the system when the creation was executed.

You must execute this call after you imported data to extraction area database and before you can use the SQL test function in Builder forms.

When Oracle 12.1 and lower is used as EA database, it is not possible to create shadow views for tables with longer names than 30 characters. This is caused by Oracle internal limitations.

5.2.4.5. Drop DM

When you click this button a job is scheduled, which deletes all tables in data mart database created by Toolkit.

If you click this button the deletion of the data mart database tables is scheduled immediately. It is not possible to cancel this process.

All tables will be deleted.

You can recreate data mart layer and load data from your extraction area– however this can be very time consuming and last several hours.

5.2.4.6. Drop DM (without date and time table)

When you click this button a job is scheduled, which deletes all tables in data mart database created by Toolkit except the date and time dimension tables.

If you click this button the deletion of the data mart database tables is scheduled immediately. It is not possible to cancel this process.

All tables except the date and time dimension tables will be deleted.

You can recreate data mart layer and load data from your extraction area– however this can be very time consuming and last several hours.

5.2.4.7. Drop aggregate tables

When you click this button a job is scheduled, which deletes all aggregate tables in data mart database created by Toolkit.

If you click this button the deletion of the data mart database tables is scheduled immediately. It is not possible to cancel this process.

All aggregate tables will be deleted.

You can recreate these tables and load data from your extraction area– however this can be very time consuming.

5.2.4.8. Update EA

Old snapshots in EA can have a different structure than the actual snapshot. Therefore, they could be incompatible with new snapshots in EA. This this could to errors during DM load.

The tool adds new tables, columns and indexes to existing snapshots in EA.

Update EA job will create new tables, add new empty columns and add indexes to existing EA snapshots. It will not delete tables and remove column or indexes.

5.2.4.9. Update DM

Update DM is a job similar to Update EA job. It detects changes in existing DM data model in database and DM builder definitions and then generates ALTER statements which update database model to current DM Builder definition. The advantage of this job is the fact that it is not needed to run DROP DM, CREATE DM and LOAD DM after every change in DM Builder. The most common changes in data model in DM are supported.

We support following changes in the data model:

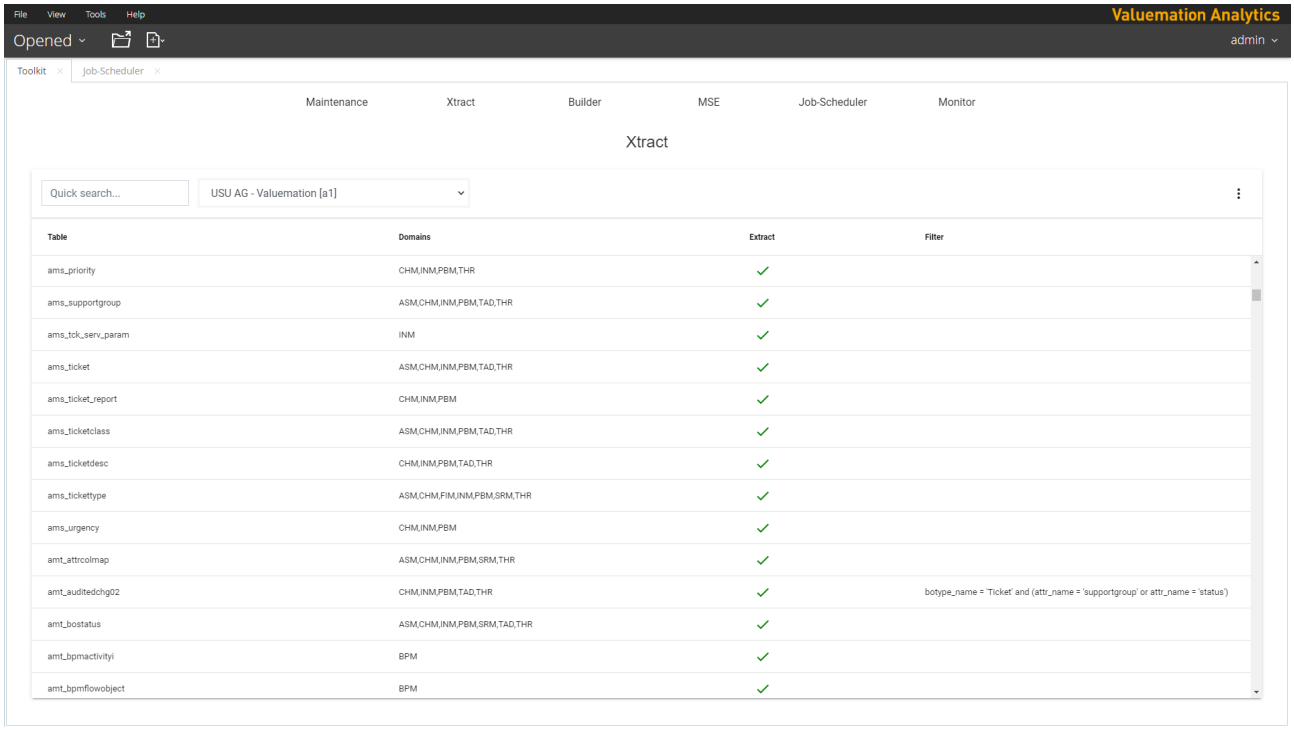
- Table is added
- Column is added
- Column length changed (longer)

We explicitly do not support:

- Table is deleted
- Column is deleted
- Column length changed (shorter)
- Data type of a column changed

If you add new tables or columns, they will be empty until the next loading. The values remain empty for old snapshots. If you want to fill in the data in an old snapshot and the data was previously available in the extract, you must follow the procedure: DROP DM, CREATE DM and LOAD DM.

5.3. Xtract



5.3.1. Overview

With the Xtract tool you define which data from your source system is imported into extraction area.

After you choose the source, the Xtract module loads the respective Xtract definition. It shows all tables in the so-called **Table Extraction Overview**.

You can find here following information:

- Table: All tables from source database
- Domains: Domains, which need this table
- Extract:
- Shows a red x, when the table is not extracted
- Shows a green checkmark, when the complete table is extracted
- Shows a green checkmark in a box, when the table is extracted partially
- Filter: the filter that is applied to the table

When you double-click a table, you enter the **Column Extraction Overview**.

Here you see:

- The columns of the table you just double-clicked
- The type and the length of the columns
- If the column is extracted
- Red x means it is not extracted
- Green checkmark means it is extracted
- If a column is indexed
- Red x means it is not indexed
- Green checkmark means it is indexed

You see only columns which have a datatype which can be imported to the application. Datatypes like RAW, BIN, LOB, CLOB or TEXT are not supported and therefore columns with this kind of datatype are not shown in Column Extraction Overview.

Every time you choose a source in Xtract the application connects to the database and checks the database information schema for changes. This could take a view minute.

5.3.2. Extract a New Table

To extract a new table from an existing source please follows these steps:

1. Open Xtract application.
2. Choose an existing source.
3. Search for the table you want to extract. You can use the full text search for that.

4. Click in column Extract.
5. You should see a green checkmark. On the next load from source database to Extraction Area the table will be extracted.

5.3.3. Extract a Table Partially

Most of the time it is not necessary to extract the complete table. To extract a table partially please follows these steps:

1. Open Xtract application.
2. Choose an existing source.
3. Search for the table you want to extract partially. You can use the full text search for that.
4. Click in column Extract, if it is not already extracted.
5. Double-click the table. The column extraction overview opens.
6. Uncheck the columns you don't want to import.
7. Check the columns you want to import.
8. Next time, when data is loaded from source database to Extraction Area only the checked columns will be extracted.

Excluding columns from an extract process, which had been imported before may lead to errors in the data load of the data mart.

5.3.4. Add or Remove an Index

Xtract is not creating the indexes the imported tables have in the source database.

To add an index to a table please follows these steps:

1. Open Xtract application.
2. Choose an existing source.
3. Search for the table you want to add an index. You can use the full text search for that.
4. Double-click the table. The column extraction overview opens.
5. Check the icon in column Index, if you want to use an index on this column.
6. Uncheck the icon in column Index, if you don't want to use an index on this column.
7. Next time, when data is loaded from source database to Extraction Area only the columns which have a checkmark are indexed.

The creation of composite indexes is not supported.

5.3.5. Add a Filter to Table Extraction

In some cases, you only want to load a part of your data to the application, e.g. only tickets, which were created after 01.01.2010. To do this, you can add a filter clause to a table. Basically, it is the where clause in a SQL select statement.

To add a filter to a table please follows these steps:

1. Prepare your filter SQL statement in a SQL client. Test it on you source database.
2. Open Xtract application.
3. Choose an existing source.
4. Search for the table you want to add the filter. You can use the full text search for that.
5. Click in column Filter and add your statement.
6. On the next load from source database to Extraction Area the data for this table will by filtered.

5.3.6. Load With Third Party Extractor

It is possible to use third party extractor and extract from data sources like CSV, REST API or Webservice etc. To use this functionality, it is needed to create an extractor, which does extraction from source system and insert data into relational database. Afterwards it is possible to copy data into EA layer with Xtract.

This extractor can be executed during SD→EA extraction. To configure this setup, it is needed to configure the parameter **External loader (command)** in the Data Source Configuration and provide path to third party extractor. Then the extractor will be executed automatically during SD→EA load.

5.3.7. The Options Menu

5.3.7.1. Reset Sorting

The sorting of the grid can be changed. If you click **Reset sorting** in **Options** menu the sorting of all columns will be reset to the standard sorting.

5.3.7.2. Reload Data

If you click **Reload data** in **Options** menu, the grid is reloaded from database.

5.3.7.3. Refresh Source

If you click **Refresh source** in **Options** menu, the system connects to source database specified in the Data Source Configuration and reads the present information form database information schema. Database metadata is updated, and you see new or changed tables and columns.

5.3.7.4. Import Xtract Definition

To exchange Xtract definitions between systems it is possible to import definitions.

To do so please follow these steps:

1. Click **Import Xtract definition** in **Options** menu, a new dialog opens.
2. Choose a definition file to import.
3. Select the desired mode to be used in case of merge conflict between an existing and an imported definition.

- Overwrite existing data: Values from import file will be used. Data in database will be overwritten.
- Delete deprecated entries: Information, which is not in the import file is deleted from database.

4. Click **Import** button and start the import of the chosen import file.

When you start the import without selecting how to solve merge conflicts the system will import without overwriting an existing definition. New information will be added.

5.3.7.5. Export Xtract Definition

To exchange Xtract definitions between systems it is possible to export them. The export file is formatted in XML.

To export one or more Xtract definitions please follow these steps:

1. Click **Export Xtract definition** in **Options** menu, a new dialog opens.
2. Choose the Xtract definition you want to export. You can use the CTRL and SHIFT keys to select the desired combination of definitions.
3. Click **Export** button.
4. The system asks you, where to save the file. After the selection of the destination folder the export starts.

5.3.8. Limitations

5.3.8.1. Indexes of MySQL 5.6

It is not possible to create index on column which length is more than 767. This behavior cannot be fixed. The problem is caused by source database (SD). If real length of column in SD table has more than 767 bytes and index in Xtract is applied, the error occurs.

However, the problem occurs only on MySQL 5.6 database. We are recommending upgrade to MySQL version 5.7.

Second option to solve this problem:

It is possible to remove indexes from Xtract definition from following tables (this option slows extract to EA):

Table: 'amt_bostatus' - Columns: 'statusvalue'

Table: 'amt_auditedchg02' - Columns: 'attr_name', 'botype_name'

Table: 'ams_ticketclass' - Columns: 'ticketclass'

Table: 'ams_ticket' - Columns: 'status'

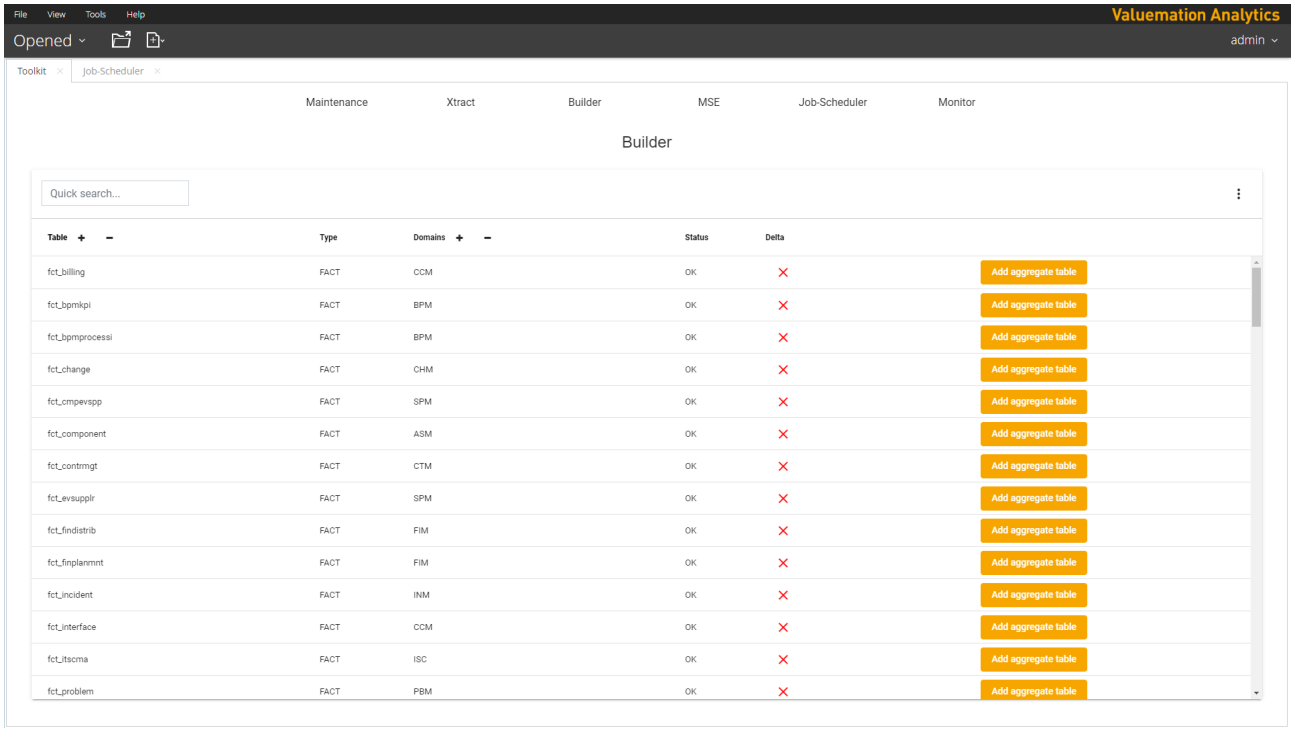
5.3.8.2. Extracting from MSSQL to Oracle

Unsolvable issue with strings which have exactly 4000 characters.

On MSSQL it is possible to store exactly 4000 characters into VARCHAR (4000) column. On Oracle it is different, because the maximum for VARCHAR2 column is 4000 bytes. One character

can have from 1 to 4 bytes. In the end it is possible to store into VARCHAR2(4000 CHAR) from 1000 CHAR to 4000 CHAR. It depends on character's length. It should be considered, that into column VARCHAR2(4000 CHAR) it is possible to store in extreme case only 1000 characters, but according to Oracle documentation it is true.

5.4. Builder



5.4.1. Overview

With the Builder application, you build the metadata model of your data mart and define the data lineage. Basically, a data mart in consists of fact, dimension and bridge tables. The tool enables you to implement new objects as well as changing existing. It supports you with semi-automatic SQL statement generation (intellisense) and various semantical and logical check mechanisms.

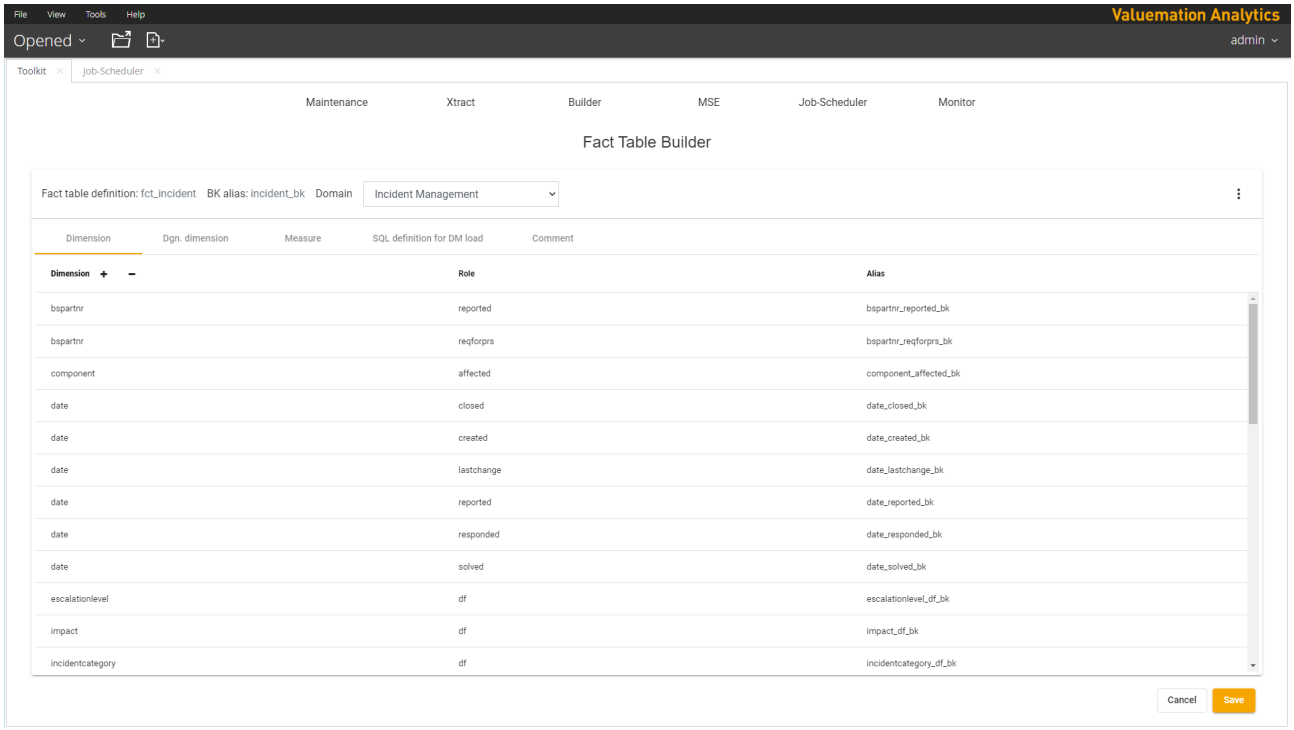
The metadata of the model is stored in MD database and can be exported in XML format. The exported file could be used to transport your implementations from one system to another. Or you can save the file in a source control tool (e.g. GIT or SVN) to track your changes.

On the Builder overview page, you see:

- All defined table objects
- Their type (FACT, DIMENSION or BRIDGE)
- The associated domains, e.g. INM for Incident Management
- And the status of the implementation
- EMPTY: The table doesn't have any columns etc. defined.
- MISSING SQL: No SQL statement is provided.
- OK: The table object is well defined.
- See the aggregate tables for fact tables

With this application, you define the metadata and the data linage of your data mart. To create the physical data structure, you must execute **Create DM** in Maintenance module.

5.4.2. The Fact Builder Form



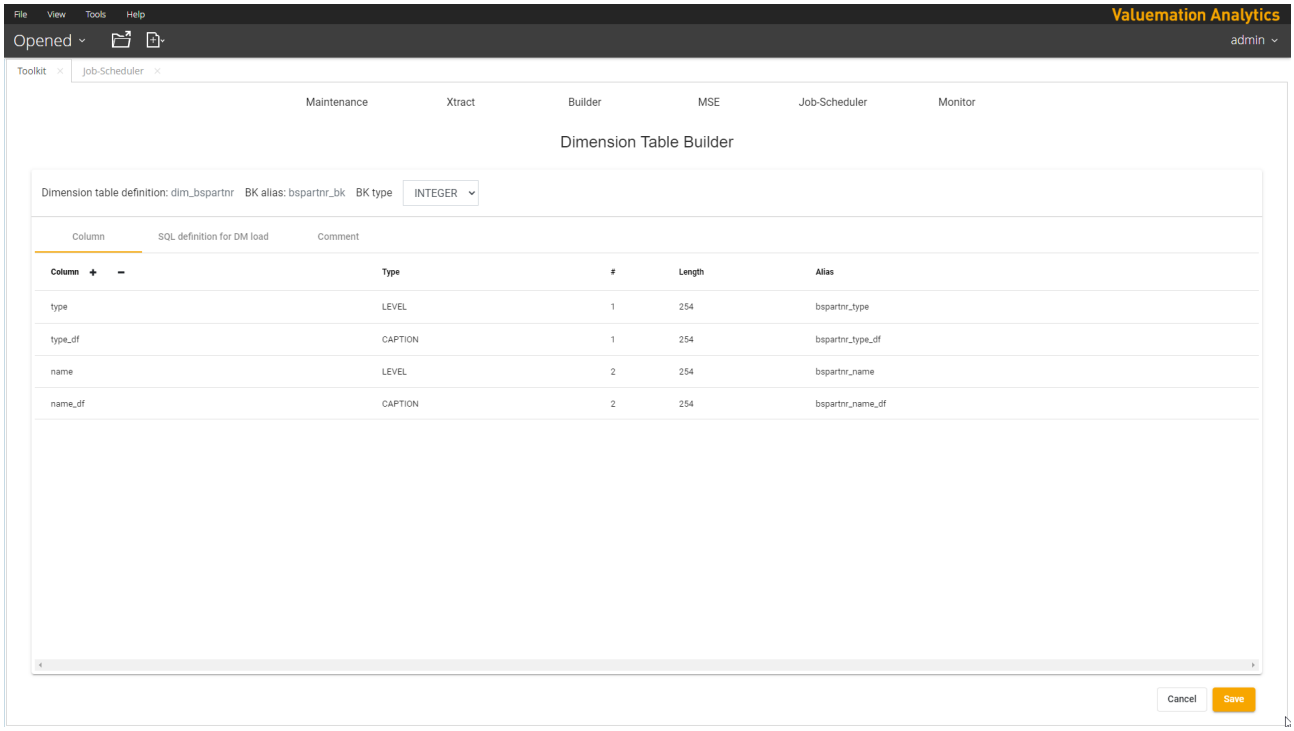
With this form, you implement and change the definition of a data mart fact table. You also specify how the data is loaded from EA to DM.

Object	Description
Fact table definition: <name>	<name> is the fact table name this definition belongs to.
BK alias	Auto generated field name for the business key alias in SQL definition for DM load.
Domain	Every fact table belongs to a domain.
Dimension tab	In the **Dimension** grid, you define the relations between your fact and existing dimensions. With + (plus) button you add a new relation, with – (minus) button you delete a relation. In the **Dimension** column, you see the name of an existing dimension. In column Role, you set the role the dimension is having in this relation. Alias is the auto generated field name for the business key alias in SQL definition for DM load. The date dimension is a good example how the role definition works. In the source system two different time stamps exist: The date, when a ticket was created and the date when the ticket was closed. In we have a dimension table with thousands of members – one record for one day. To be able to use the same database table you assign the same dimension table date but give it two different roles (closed and created).

Object	Description
Dgn. Dimension tab	In the **Dgn. dimension** grid you define degenerated dimensions for your fact. With + (plus) button you add a column, with – (minus) button you delete column. In the **Dgn. Dimension** column you set the name of the degenerated dimension you want to use. Alias is the auto generated field name for the business key alias in SQL definition for DM load. You also define here if the system generates a database index for this column.
Measure tab	In the **Measure** grid, you define measures for your fact. With + (plus) button you add a column, with – (minus) button you delete column. In the **Measure** column, you set the name of the measure you want to use. Alias is the auto generated field name for the measure in SQL definition for DM load.
SQL definition for DM load tab	This SQL statement is used to populate the fact table when you start the DM load. It is executed against the relevant EA snapshot and the result is written in DM table. This is how you define the data lineage. You can write the SQL by yourself, but it is recommended to use the function **Generate SQL statement** from right menu. This function generates a template SQL statement, which you must fill with reasonable information. Use the buttons on the right to preview the result, format your code or generate SQL statement. For more details about SQL editor see chapter The SQL Editor in Fact, Dimension and Bridge Builder (page 102). When you click save, the SQL is checked against the metadata. It is not possible to save a fact definition with a faulty or not complete SQL statement.
Comment tab	You can add your notes and comments here.

When you use Oracle database functions with spaces are not allowed i.e. **date '2017-01-01'**, must be replaced by **to_date('2017-01-01','YYYY-MM-DD')**.

5.4.3. The Dimension Builder Form

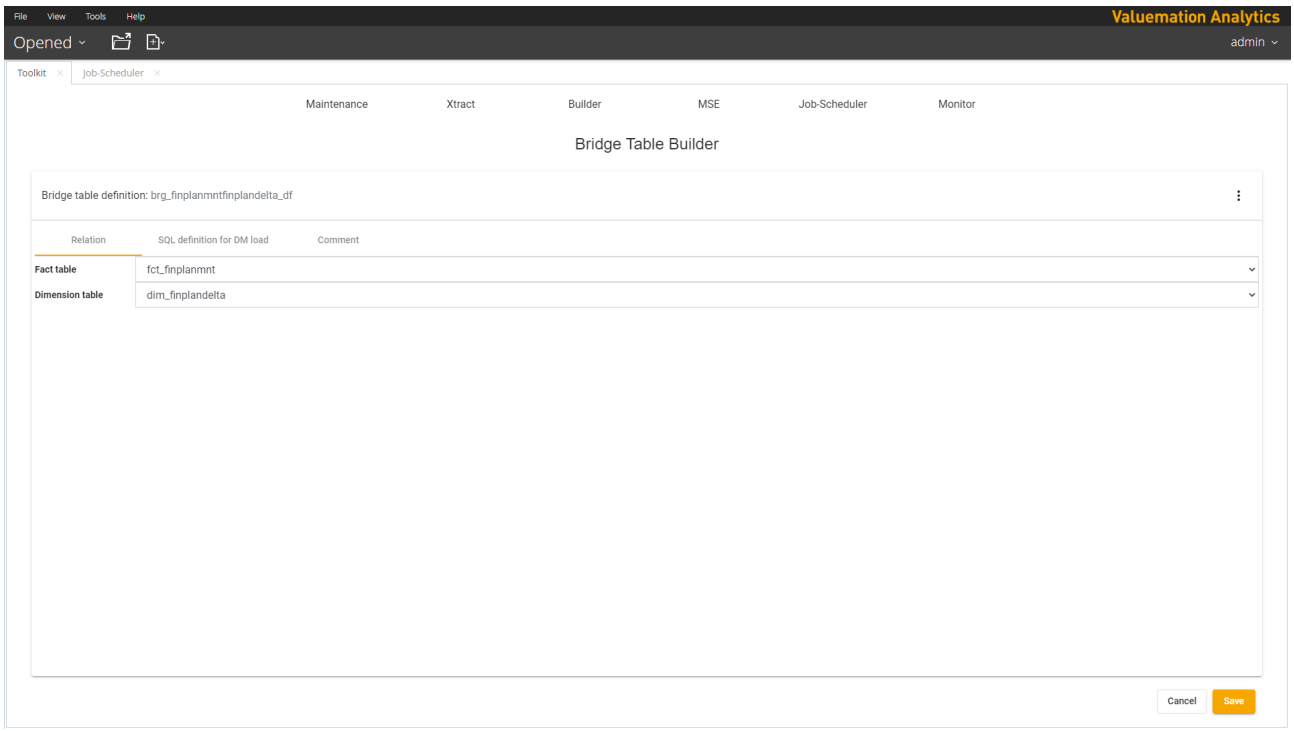


With this form, you implement and change the definition of a data mart dimension table. You also specify how the data is loaded from EA to DM.

Object	Description
Dimension table definition: <name>	<name> is the dimension table name this definition belongs to.
BK alias	Auto generated field name for the business key alias in SQL definition for DM load.
Column tab	In the **Column** grid, you define the hierarchy and levels of your dimension. With + (plus) button you add a new column, with – (minus) button you delete a column. In the **Column** column, you set the name of a column in the dimension table. In column Type you define if this is a level or a cation. With # you define the hierarchical level. Length sets the length of the fields on database level. And Alias is the auto generated field name for the business key alias in SQL definition for DM load.
SQL definition for DM load tab	This SQL statement is used to populate the dimension table when you start the DM load. It is executed against the relevant EA snapshot and the result is written in DM table. This is how you define the data lineage. You can write the SQL by yourself, but it is recommended to use the function **Generate SQL statement** from right menu. This function generates a template SQL statement, which you must fill with reasonable information. Use the buttons on the right to preview the result, format your code or generate SQL statement. For more details about SQL editor see chapter The SQL Editor in Fact, Dimension and Bridge Builder (page 102). When you click save, the SQL is checked against the metadata model. It is not possible to save a dimension definition with a faulty SQL statement.
Comment tab	You can add your notes and comments here.

When you use Oracle database functions with spaces are not allowed i.e. **date '2017-01-01'**, must be replaced by **to_date('2017-01-01','YYYY-MM-DD')**.

5.4.4. The Bridge Builder Form

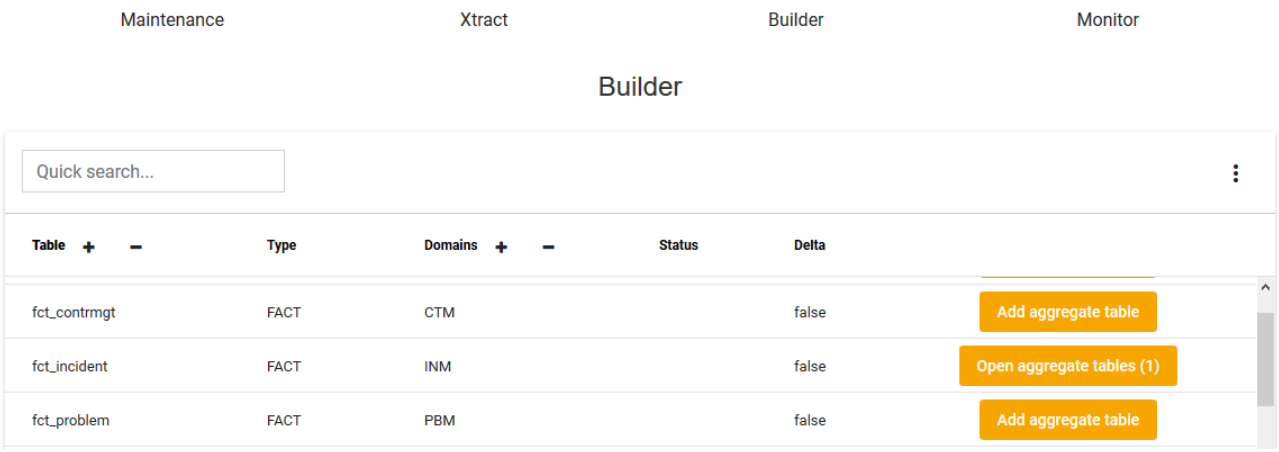


With this form, you implement and change the definition of a data mart bridge table. You also specify how the data is loaded from EA to DM.

Object	Description
Bridge table definition: <name>	<name> is the bridge table name this definition belongs to.
Relation tab	Fact table: Fact table of the bridge table relation. Dimension table: Dimension table of the bridge table relation.
SQL definition for DM load tab	This SQL statement is used to populate the bridge table when you start the DM load. It is executed against the relevant EA snapshot and the result is written in DM table. This is how you define the data lineage. You can write the SQL by yourself, but it is recommended to use the function **Generate SQL statement** from right menu. This function generates a template SQL statement, which you must fill with reasonable information. Use the buttons on the right to preview the result, format your code or generate SQL statement. For more details about SQL editor see chapter The SQL Editor in Fact, Dimension and Bridge Builder (page 102). When you click save, the SQL is checked against the metadata model. It is not possible to save a bridge definition with a faulty SQL statement.
Comment tab	You can add your notes and comments here.

When you use Oracle database functions with spaces are not allowed i.e. **date '2017-01-01'**, must be replaced by **to_date('2017-01-01','YYYY-MM-DD')**.

5.4.5. The Aggregate Table Builder



Aggregate tables could be created for each fact table. Link *Aggregate Tables* opens new page **Aggregate Table Builder**.

In this builder, it is possible to create aggregate tables, which help to optimize performance of Analyzer Report.

Object	Description
Dimension alias tab	**[+]** To add dimensions of fact table belonging to aggregate table **[-]** To delete chosen dimensions
Measure tab	To add measures of fact table belonging to aggregate table**[+]** Select type of measure [BOOLEAN, FLOAT, INTEGER] Select type of aggregator [AVG, COUNT, MAX, MIN, SUM] **[-]** To delete chosen measures
Name and alias tab	**[+]** To create new aggregate table, it is needed to define a name It is possible to copy definition of existing aggregate table **[-]** To delete created aggregate table
	If all definitions are done it is needed to confirm by to click **Save.**
	To create new aggregate table, it is needed to define a name**[+]** It is possible to copy definition of existing aggregate table **[-]** To delete created aggregate table

For further information about aggregate tables look at this page: http://mondrian.pentaho.com/documentation/aggregate_tables.php.

5.4.5.1. Examples of aggregation tables

To use prepared examples of aggregate tables you need to upload following file via Upgrade Manager.

```
<installation package>\si_toolkit\platform\agg_table_examples\si-builder\<your dbms>.zip
```

Upgrade Manager → upload si-builder file → Auto resolve → missing in origin → next → next → Apply decision.

More about upgrade manager can be found in chapter Upgrade Manager.

Afterward you need to load the aggregate table via toolkit more in chapters Create aggregate tables (page 88) and Load aggregate tables.

Then upload the Mondrian schema.

<installation package>\si_toolkit\platform\agg_table_examples\mondrian\100-dm-incident_management-agg*.xml

Manage Data Source → edit Incident Management → upload the Mondrian with agg. tables.

5.4.6. Common Fact, Dimension and Bridge Builder functions

5.4.6.1. Generate SQL statement

Generate SQL statement will use the metadata model of your fact, dimension or bridge (left side of the form) and build a template for the SQL you must fill with reasonable information. If the SQL field is empty the function will create a template like this:

```
SELECT
    NULL AS service_bk,
    NULL AS service_name_df,
    NULL AS service_name
FROM
    DUAL d
```

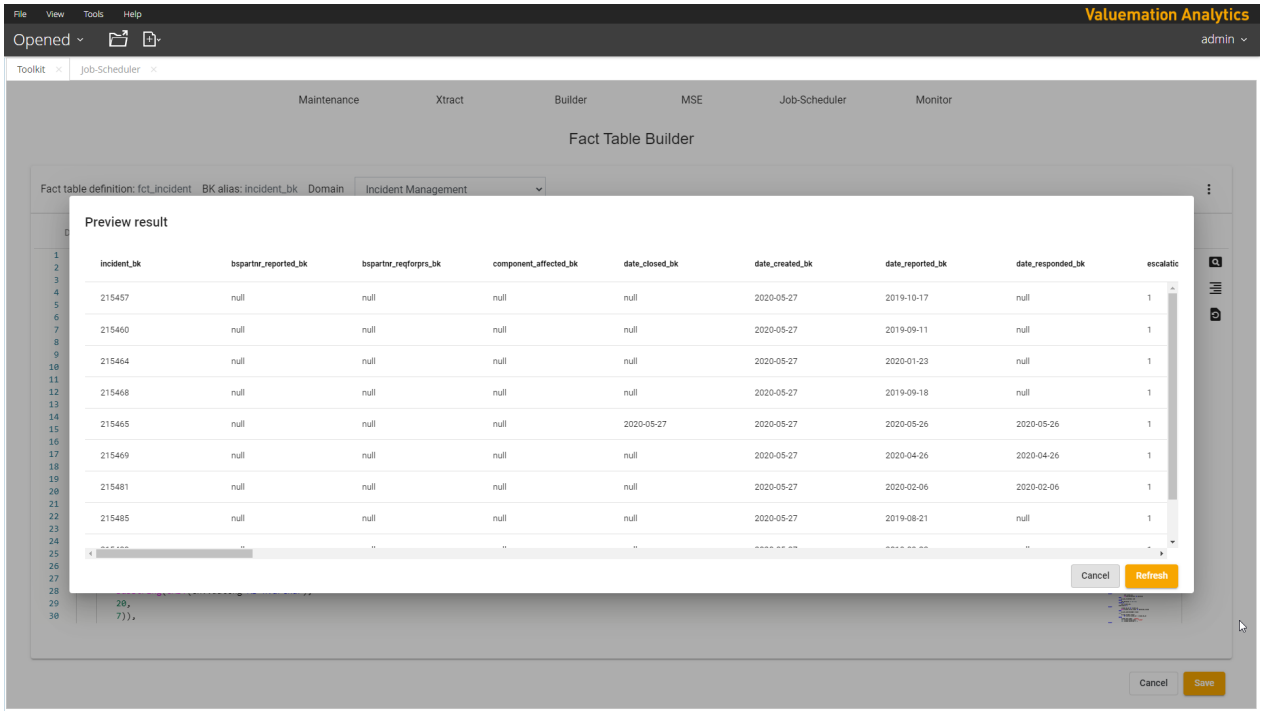
You must add the right fields and columns from your source database to fill the data mart tables.

If you have added a e.g. a measure in an existing fact definition and click **Generate SQL statement,** the system will recognize the missing part in the SQL and add a template line.

```
…
    CASE
        WHEN tre.firstlev_solution = 'Y' THEN 1
        ELSE 0
    END AS incident_isfirstlevsolved,
    tdc.duration * 60 AS incident_ttspendinseconds,
    tre.sol_dur_min * 60 AS incident_ttsolveinseconds,
    NULL AS incident_mynewmeasure
FROM
    a1_ams_ticket tck
…
```

5.4.6.2. Preview Result

You can run the SQL statement defined in field **SQL definition for DM load** against the extraction area (shadow views). The system will return to you the first ten rows of the result set.



Before you can use this functionality, you must create shadow views. Please see chapter Load aggregate tables.

5.4.6.3. Reset Sorting

The sorting of the grids can be changed. If you click **Reset sorting** in **Options** menu the sorting of all columns in all grids will be reset to the standard sorting.

5.4.6.4. Reload Data

If you click **Reload data** in **Options** menu, the form is reloaded from database.

5.4.7. The SQL Editor in Fact, Dimension and Bridge Builder

In Fact, Dimension and Bridge Builder we integrated a simple SQL editor. To make working with it more comfortable, we implemented these functionalities:

- Syntax highlighting, row numbers, search and replace
- Auto completion for ANSI SQL commands, extracted tables and columns
- Format SQL
- Generate SQL Statement
- Expand SQL editor window

Syntax highlighting, row numbers and auto completion are features, which are integrated into the editor window itself:

Feature	Description
Syntax highlighting	The SQL commands and build in functions are highlighted to make the text more readable.

Feature	Description
Row numbers	Editor shows row number of code line.
Auto completion (intellisense)	If you start typing, the editor gives you hints and opens an auto completion form. The tool tries to auto complete the text and uses for this a list of ANSI SQL commands, extracted tables and columns.
Search and replace	If the cursor is placed in the SQL editor field, you can start a search by typing CTRL-f or start a replace dialog by typing CTRL-h.
Minimap (code outline)	A Minimap (code outline) gives you a high-level overview of your SQL statement, which is useful for quick navigation and code understanding. A SQL Minimap is shown on the right side of the editor. You can click or drag the shaded area to quickly jump to different sections of your code.

To access the other features, we provide you with buttons:

Feature	Description
Preview result	 The preview form opens.
Format SQL	 This feature is formatting the SQL code for you, without checking the syntax against the meta data.
Generate SQL Statement	 You can write the SQL by yourself, but it is recommended to use the function Generate SQL statement. This function generates a template SQL statement, which you must fill with reasonable information.

5.4.8. How To

This is a collection of common use cases. It covers most of the functions necessary to create and maintain Mondrian schemas.

Let's assume you have the Incident Management domain installed and you want to do customizing of this domain. The usual things you want to do are:

- Add a new measure to your fact definition, because you would like to us it in your analysis.
- Add an already existing dimension to your fact definition.
- Add new dimensions based on standard date or time dimension.
- Add a degenerated dimension to your fact definition.
- Change the source of a measure in your fact definition. This could be necessary after you customized your source system.
- Change the source of a degenerated dimension in your fact definition.
- Build a new dimension and add it to your fact definition, because you would like to us it in your analysis.
- Build a new bridge table and add it to your fact definition.
- Build a new fact table and use it in your analysis.
- Build a new domain and assign Builder objet to it.

- Do a cleanup and delete not needed Builder objects like facts, dimensions, bridges and domains.

5.4.8.1. Add Dimension to Fact

Let's assume the dimension dim_sla is not used in fact fct_incident, but it already exists in our data mart.

If you want to add an existing dimension to an existing fact, please follow these steps:

1. Make sure:
   - That the dimension you want to add is implemented. If not, please add it to Builder.
   - That the source of the dimension relation you want to add is extracted.

   The dimension we use in our example is dim_sla. We checked in Builder and it exists.

   The relation to the SLA is saved in ams_ticket.tck_serv_param_id. We checked the import and it is extracted.
2. Open Builder and open fact definition in Fact Builder. In our example fct_incident.
3. Click the **+** (plus) in column Dimension. The **New dimension** dialog opens.

   Select an existing Dimension from dropdown list. In our example, we choose sla.

   Give the dimension relation a Role. It should be descriptive. In our example, we give the relation the role requested, because the SLAs are dealing with this topic.

   The Alias is generated automatically. The result is in our case sla_requested_bk.

   Click **Create** button. The new dimension is added to Dimension grid.
4. Choose **Generate SQL statement** from **Options** menu. The system will recognize the missing part in the SQL and add a template line. You could also do this by yourself, but we strongly recommend using this option.

   In our example, this line was added to the SQL statement: NULL AS sla_requested_bk.
5. Modify the SQL statement and add expression for dimension relation.

   In our example, we created this expression: tck.tck_serv_param_id AS sla_requested_bk.

   tck is the table alias for ams_ticket table in this context.
6. You can test your result with the Preview result function in Options menu.
7. Save your changes. Now the metadata of your data mart was changed.

Changes in the metadata of your data mart are not reflected in the database structure automatically. The following manual procedure is needed, if you want to use your changes. Otherwise the next data load will fail.

Execute **Drop DM** in Maintenance module

Execute **Create DM** in Maintenance module

Execute **ReCreate DateTime dimensions** in Maintenance module (this may take some minutes, you don't need to do this, if you are in development process)

Execute **Load DM** in Maintenance module

5.4.8.2. How to Use Date and Time Dimension in Fact Table

By default, date and time dimensions are automatically created by Toolkit when Create DM job is executed. They are empty, and it is needed to call job **ReCreate DateTime dimensions** to insert data into them. When job Drop DM is called, also date and time dimensions are dropped.

To use date and time dimension in fact table it is needed to do the following:

1. Open Builder and open fact definition in Fact Builder. In our example fct_incident.
2. Click the **+** (plus) in column Dimension. The **New dimension** dialog opens.

   Select dimension "date" or "time" and define the Role of this dimension. It should be descriptive.

   Click **Create** button. The new dimension is added to Dimension grid.
3. Choose **Generate SQL statement** from **Options** menu. The system will recognize the missing part in the SQL and add a template line.
4. Modify the SQL statement and add an expression for the date or time dimension foreign key:
   - For date dimension it is expected, that the expression will return a format like YYYY-MM-DD (e.g. 2015-01-01),
   - For time dimension the format must be HH:MM:SS (e.g. 14:04:04)

   For detailed examples please see the Incident Management implementation.
5. Save your changes. Now the metadata of your data mart was changed.

Changes in the metadata of your data mart are not reflected in the database structure automatically. The following manual procedure is needed, if you want to use your changes. Otherwise the next data load will fail.

Execute **Drop DM** in Maintenance module

Execute **Create DM** in Maintenance module

Execute **ReCreate DateTime dimensions** in Maintenance module (this may take some minutes, you don’t need to do this, if you are in development process)

Execute **Load DM** in Maintenance module

5.4.8.3. Add Degenerated Dimension to Fact

If you want to add a new degenerated dimension to an existing fact, please follow these steps:

1. Make sure the source of the degenerated dimension you want to add is extracted. If not, please add it to Xtract definition.

   The field we use in our example is ams_ticket_report.callmedia. We checked the import and it is extracted.
2. Open Builder and open fact definition in Fact Builder. In our example fct_incident.
3. Click + (plus) in Dgn. dimension grid. The **New degenerated dimension** dialog opens.

   Add the name of the new degenerated dimension. In our example incidentsourcechreported.

   The Alias is generated automatically. The result is in our case incidentsourcechreported_bk.

   With checkbox Index you define if a database index is generated for this degenerated dimension or not.

Click **Create** button. The new degenerated dimension is added to Dgn. dimension grid.

4. Choose **Generate SQL statement** from **Options** menu. The system will recognize the missing part in the SQL and add a template line. You could also do this by yourself, but we strongly recommend using this option.

   In our example, this line was added to the SQL statement: NULL AS incidentsourcechreported_bk.

5. Modify the SQL statement and add expression for the degenerated dimension.

   In our example, we created this expression: tck.callmedia AS incidentsourcechreported_bk.

   - tck is the table alias for ams_ticket table in this context.

6. You can test your result with the Preview result function in Options menu.
7. Save your changes. Now the metadata of your data mart was changed.

Changes in the metadata of your data mart are not reflected in the database structure automatically. The following manual procedure is needed, if you want to use your changes. Otherwise the next data load will fail.

Execute **Drop DM** in Maintenance module

Execute **Create DM** in Maintenance module

Execute **ReCreate DateTime dimensions** in Maintenance module (this may take some minutes, you don't need to do this, if you are in development process)

Execute **Load DM** in Maintenance module

5.4.8.4. Add Measure to Fact

In our analysis, we are interested in the solution time in days. In the current implementation, this measure is missing.

If you want to add a new measure to an existing fact, please follow these steps:

1. Make sure the source of the measure you want to add is extracted. If not, please add it to Xtract definition.

   The field we use in our example is ams_ticket_report.sol_dur_min. We checked the import and it is extracted.

2. Open Builder and open fact definition in Fact Builder. In our example fct_incident.
3. Click + (plus) in Measure grid. The **New measure** dialog opens.

   Add the name of the new measure. In our example ttsolveindays.

   Choose a Type for the new measure. We will use FLOAT in our example.

   The Alias is generated automatically. The result is in our case incident_ttsolveindays.

   Click **Create** button. The new measure is added to Measure grid.

4. Choose **Generate SQL statement** from **Options** menu. The system will recognize the missing part in the SQL and add a template line. You could also do this by yourself, but we strongly recommend using this option.

   In our example, this line was added to the SQL statement: NULL AS incident_ttsolveindays.

5. Modify the SQL statement and add expression for measure calculation. You can use e.g. database functions and sub-selects.

In our example, we created this expression: tre.sol_dur_min / 1440 AS incident_ttsolveindays

- tre is the table alias for ams_ticket_report
- sol_dur_min is the field we need for our calculation
- 1440 is 60*24 (the minutes of a day)

6. You can test your result with the Preview result function in Options menu.
7. Save your changes. Now the metadata of your data mart was changed.

Changes in the metadata of your data mart are not reflected in the database structure automatically. The following manual procedure is needed, if you want to use your changes. Otherwise the next data load will fail.

Execute **Drop DM** in Maintenance module

Execute **Create DM** in Maintenance module

Execute **ReCreate DateTime dimensions** in Maintenance module (this may take some minutes, you don't need to do this, if you are in development process)

Execute **Load DM** in Maintenance module

5.4.8.5. Change Source of Measure in Fact

Sometimes it is necessary to change the source of a measure in an existing fact, e.g. if you have customized your source system. To change the source of a measure in an existing fact, please follow these steps:

1. Make sure the new source of the measure you want to change is extracted. If not, please add it to Xtract definition.

   The field we use in our example is ams_ticket_report.x_sol_dur_min. We checked the import and it is extracted.
2. Open Builder and open fact definition in Fact Builder. In our example fct_incident.
3. Modify the SQL statement and add expression for measure calculation. You can use e.g. database functions and sub-selects.

   In our former example, we created this: tre.sol_dur_min / 1440 AS incident_ttsolveindays

   Because the source was changed we can easily replace sol_dur_min with x_sol_dur_min.
4. You can test your result with the Preview result function in Options menu.
5. Save your changes. Now the metadata of your data mart was changed.

From next DM load on this new definition is used to fill the data mart.

If the value from the new field hasn't existed in former EA snapshots and you rebuild your data mart from scratch, the measure values are empty for the old snapshots.

5.4.8.6. Change Source of Degenerated Dimension in Fact

Sometimes it is necessary to change the source of a degenerated dimension in an existing fact, e.g. if you have customized your source system. To change the source of a degenerated in an existing fact, please follow these steps:

1. Make sure the new source of the degenerated dimension you want to change is extracted. If not, please add it to Xtract definition.

   The field we use in our example is ams_ticket_report.x_callmedia. We checked the import and it is extracted.
2. Open Builder and open fact definition in Fact Builder. In our example fct_incident.
3. Modify the SQL statement and add expression for the degenerated dimension.

   In our former example, we created this: tck.callmedia AS incidentsourcechreported_bk.

   Because the source was changed we can easily replace callmedia with x_callmedia.
4. You can test your result with the Preview result function in Options menu.
5. Save your changes. Now the metadata of your data mart was changed.

From next DM load on this new definition is used to fill the data mart.

If the values form the new field hasn't existed in former EA snapshots and you rebuild your data mart from scratch, the measure values are empty for the old snapshots.

5.4.8.7. Build a Dimension Table

To create a new dimension table, please follow these steps:

1. Make sure the source of the new dimension you want to add is extracted. If not, please add it to Xtract definition.

   The dimension we use in our example is dim_services. The dimension will be sourced by the table ama_services. We checked the import and it is extracted.
2. Open Builder application.
3. Click the **+** (plus) in column Table. The **New table** dialog opens.

   Choose **DIMENSION** as type. Fill in the **name** of the new dimension table. In our example, we use services.

   Click **Create** button. The new table is added to Builder overview grid with status EMPTY.
4. Double-click on the table name and the Dimension Builder form opens.
5. Choose a BK type
   - If the source system delivers a unique primary key with data type int use INTEGER.
   - If the source system delivers a unique primary key with data type varchar use STRING and choose the correct length of the field
   - If the source system delivers no unique primary key use STRING and check Hash. will generate a 32 character hash code.

   In our example, we choose INTEGER.
6. Add columns.
   1. Click + (plus) in column header Column.
   2. The New column dialog opens.

3. Fill in the name of the new column. Please follow naming conventions.
4. Select a type
   - LEVEL: This column will be in Mondrian as <column> attribute
   - CAPTION: This column will be in Mondrian as <captionColumn> attribute
   - PARENT_CHILD: This column will be used in Mondrian for parent-child hierarchy definition
5. Fill in Order of the new column. In this context Order has the same meaning as Hierarchy Level.
6. Fill in Length of the new column. Get the correct value from your source system.
7. Click Create button.
8. Repeat these steps for each hierarchy level you want to design.

7. Choose **Generate SQL statement** from **Options** menu. The system will create a SQL template. You could also do this by yourself, but we strongly recommend using this option.

   In our example, the template looks like this:

```
SELECT
    NULL AS service_bk,
    NULL AS service_name_df,
    NULL AS service_name
FROM
    DUAL d
```

8. Modify the SQL statement and add the expressions to fill the dimension.

   In our example, the result could look like this:

```
SELECT
    asr.services_id AS service_bk,
    asr.servicename AS service_name,
    asr.servicename AS service_name_df
FROM
    a1_ama_services asr
```

9. Save your changes. Now the metadata of your data mart was changed, and you can start using the dimension in your facts.

Changes in the metadata of your data mart are not reflected in the database structure automatically. The following manual procedure is needed, if you want to use your changes. Otherwise the next data load will fail.

Execute **Drop DM** in Maintenance module

Execute **Create DM** in Maintenance module

Execute **ReCreate DateTime dimensions** in Maintenance module (this may take some minutes, you don't need to do this, if you are in development process)

Execute **Load DM** in Maintenance module

5.4.8.8. Build a Bridge Table

In some circumstances, it is necessary to create a 1:n relation between a fact and a dimension table – a so called bridge table.

To create a new bridge table, please follow these steps:

1. Make sure the source of the new bridge table you want to add is extracted. If not, please add it to Xtract definition.
2. Open Builder application.
3. Click the **+** (plus) in column Table. The **New table** dialog opens.
   Choose **BRIDGE** as type. Fill in the **name** of the new dimension table.
   Give the bridge table a Role.
   Choose a Fact table from dropdown list.
   Choose a Dimension table from dropdown list.
   Click **Create** button. The new table is added to Builder overview grid with status MISSING SQL.
4. Double-click on the table name and the Bridge Builder form opens.
5. Choose **Generate SQL statement** from **Options** menu. The system will create a SQL template. You could also do this by yourself, but we strongly recommend using this option.
6. Modify the SQL statement and add the expressions to fill the dimension.
7. Save your changes. Now the metadata of your data mart was changed, and you can start using the bridge dimension in your facts.

Changes in the metadata of your data mart are not reflected in the database structure automatically. The following manual procedure is needed, if you want to use your changes. Otherwise the next data load will fail.

Execute Drop DM in Maintenance module

Execute Create DM in Maintenance module

Execute ReCreate DateTime dimensions in Maintenance module (this may take some minutes, you don't need to do this, if you are in development process)

Execute Load DM in Maintenance module

5.4.8.9. Build a Fact Table

To create a new fact table, please follow these steps:

1. Make sure the source of the new fact you want to add is extracted. If not, please add it to Xtract definition.#
2. Open Builder application.
3. Click the **+** (plus) in column Table. The **New table** dialog opens.
   Choose **FACT** as type. Fill in the **name** of the new fact table.
   Choose a Domain from dropdown list.
   Click **Create** button. The new table is added to Builder overview grid with status EMPTY.
4. Double-click on the table name and the Bridge Builder form opens.
5. Add columns for Dimension relations, Dgn. Dimensions and Measures.
   - Click + (plus) in the respective column header.
   - The New dialog opens.

- Fill in the needed information.
- Click Create button.
- Repeat these steps for each hierarchy level you want to design.

For a complex example please see Incident Management implementation.

6. Choose **Generate SQL statement** from **Options** menu. The system will create a SQL template. You could also do this by yourself, but we strongly recommend using this option.
7. Modify the SQL statement and add the expressions to fill the fact table.
   For complex examples please see Incident Management implementation.
8. Save your changes. Now the metadata of your data mart was changed.

Changes in the metadata of your data mart are not reflected in the database structure automatically. The following manual procedure is needed, if you want to use your changes. Otherwise the next data load will fail.

Execute Drop DM in Maintenance module

Execute Create DM in Maintenance module

Execute ReCreate DateTime dimensions in Maintenance module (this may take some minutes, you don't need to do this, if you are in development process)

Execute Load DM in Maintenance module

5.4.8.10. Use Transaction Key Information in Builder Select

It is possible to use parameter `${SI_CURRENT_TRANSACTION_DATETIME}` in all selects in DM Builder. This parameter is replaced during load with string which contains snapshot transaction key.

The format is yyyy-MM-dd HH:mm:ss (e.g. 2017-03-02 08:08:58).

You can use it for time related measure calculations, e.g.

```
CASE
WHEN TO_DATE('${SI_CURRENT_TRANSACTION_DATETIME}', 'YYYY-MM-DD HH24:MI:SS') <
CAST(doc.c_creatordate AS DATE)
THEN NULL
ELSE (TO_DATE('${SI_CURRENT_TRANSACTION_DATETIME}', 'YYYY-MM-DD HH24:MI:SS') -
CAST(doc.c_creatordate AS DATE)) * 24
END
```

5.4.8.11. Delete an Existing Table Object

To delete an existing table object please follows these steps:

1. Open Builder application.
2. Click the **-** (minus) in column Table. The **Delete table** dialog opens.
   Choose the table you want to delete.
   Click **Delete** button. Now the chosen table is deleted from VMAMDR.

Deletion of tables is irreversible. Please make sure you choose the correct table. The physical table is not deleted form DM database immediately. It exists if you do not drop and rebuild your data mart database.

5.4.8.12. Create a New Domain

To create a new domain object please follow these steps:

1. Open Builder application.
2. Click the **+** (plus) in column Domain. The **New domain** dialog opens.
   Fill in the name of the new domain. Fill in a three-letter abbreviation for the new domain, like INM for Incident Management.
   Click **Create** button.
3. The new domain is now created and can be used.

5.4.8.13. Delete an Existing Domain

To delete an existing domain please follows these steps:

1. Open Builder application.
2. Click the **-** (minus) in column Domain. The **Delete domain** dialog opens.
   Choose the domain you want to delete from the list.
   Click **Delete** button. Now the chosen domain is deleted from VMAMDR.

Deletion of domains is irreversible. Please make sure you choose the correct domain. You can delete only domains which are not used in any table object definition.

5.4.8.14. Mondrian Schema Changes

All changes you do in DM using Toolkit must be reflected in Mondrian schema. You can use the Mondrian Schema Editor application to do this.

Even though we offer the Mondrian Schema Editor, an application for editing a Mondrian schema, the adaptation is mostly manual work. We offer trainings on this topic. Please contact us for more information.

5.4.8.15. Use Your Own Database Functions

For some use cases, it is easier to use custom database functions. It makes the fact table SQL easier to understand and sometimes problems are much easier solved by using them. A user can define a database function and save the SQL CREATE statement in a file in this folder:

\si\server\pentaho-server\tomcat\webapps\sitoolkit_backend\WEB-INF\classes\db\function

After each SD to EA extraction the functions are recreated according to the files saved in this folder. You can use this custom database function in Builder.

From version 4.4 we ship a custom database function VMA_UNISTR() in our standard.

The file name must start with VMA_. Other files are ignored. It is a good habit to use the file ending .sql, e.g. the function we ship with our standard is saved in a file named VMA_UNISTR.sql.

5.4.9. The Options Menu

5.4.9.1. Reset Sorting

The sorting of the grid can be changed. If you click **Reset sorting** in **Options** menu the sorting of all columns will be reset to the standard sorting.

5.4.9.2. Reload Data

If you click **Reload data** in **Options** menu, the grid is reloaded from database.

5.4.9.3. Import Builder Definition

To exchange Builder definitions between systems it is possible to import definitions. To do so please follow these steps:

1. Click **Import Builder definition** in **Options** menu, a new dialog opens.
2. Choose a definition file to import.
3. Click **Import** button and start the import of the chosen import file.

The objects are **not** merged but overwritten! If you import a new zip archive and you have not made a backup of your implementation, it is lost and cannot be restored.

5.4.9.4. Export Builder Definition

To exchange Builder definitions between systems it is possible to export them. The system will export all Builder objects to XML formatted files and save them as ZIP archive.

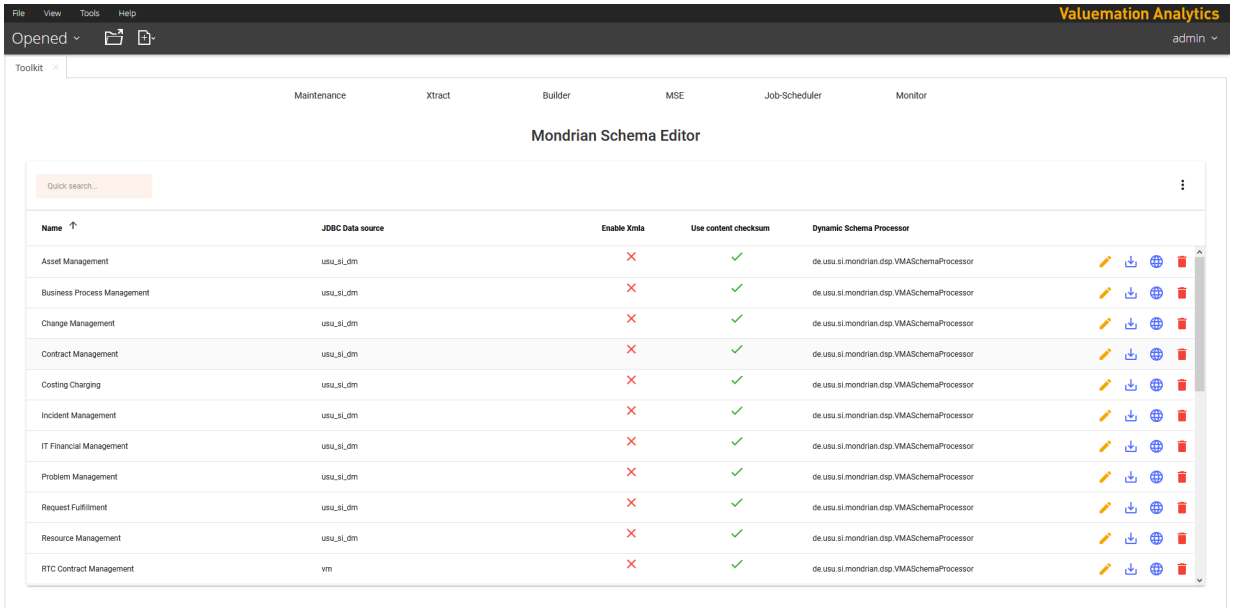
To export the Builder definitions please follow these steps:

1. Click **Export Builder definition** in **Options** menu, a new dialog opens.
2. Click **Export** button.
3. The system asks you, where to save the file. After the selection of the destination folder the export starts.

In addition to the Builder object XML files a file called About.xml is saved in the ZIP archive. There you find export information about version build, database type, hostname, database and user.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<About>
<version>SI 4.0</version>
<build>52b833</build>
<databaseType>MYSQL</databaseType>
<hostName>172.16.40.3</hostName>
<databaseName>j_t201_vmd_1_mys_76_md</databaseName>
<user>usu_devel</user>
</About>
```

5.5. Mondrian Schema Editor



5.5.1. Overview

With Builder application you defined a multidimensional data model, which is generated in your database by executing the job Create DM in Toolkit. With this physical multidimensional data model in place, you must create a logical model that maps to it. A Mondrian schema is essentially an XML file that performs this mapping, thereby defining a multidimensional database structure. You can create Mondrian schemas using the **Mondrian Schema Editor** application.

In a very basic scenario, you will create a Mondrian schema with one cube that consists of a single fact table and a few dimensions, each with a single hierarchy consisting of a handful of levels. More complex schemas may involve multiple virtual cubes, and instead of mapping directly to the single fact table at the centre of a star schema, they might map to views or inline tables instead.

The Mondrian schemas can be exported in XML format. The exported file could be used to transport your implementations from one system to another. Or you can save the file in a source control tool (e.g. GIT or SVN) to track your changes.

On the Mondrian Schema Editor overview page, you can:

- See all your Mondrian Schemas stored in the system
- Their JDBC data source and dynamic schema processor
- The values of parameter UseContentChecksum and EnableXmla
- Search for Mondrian Schemas by using the quick search

From the more menu you can:

- Create a new Monrian schema
- Import a Mondrian schema
- Download all mondrian schemas at once

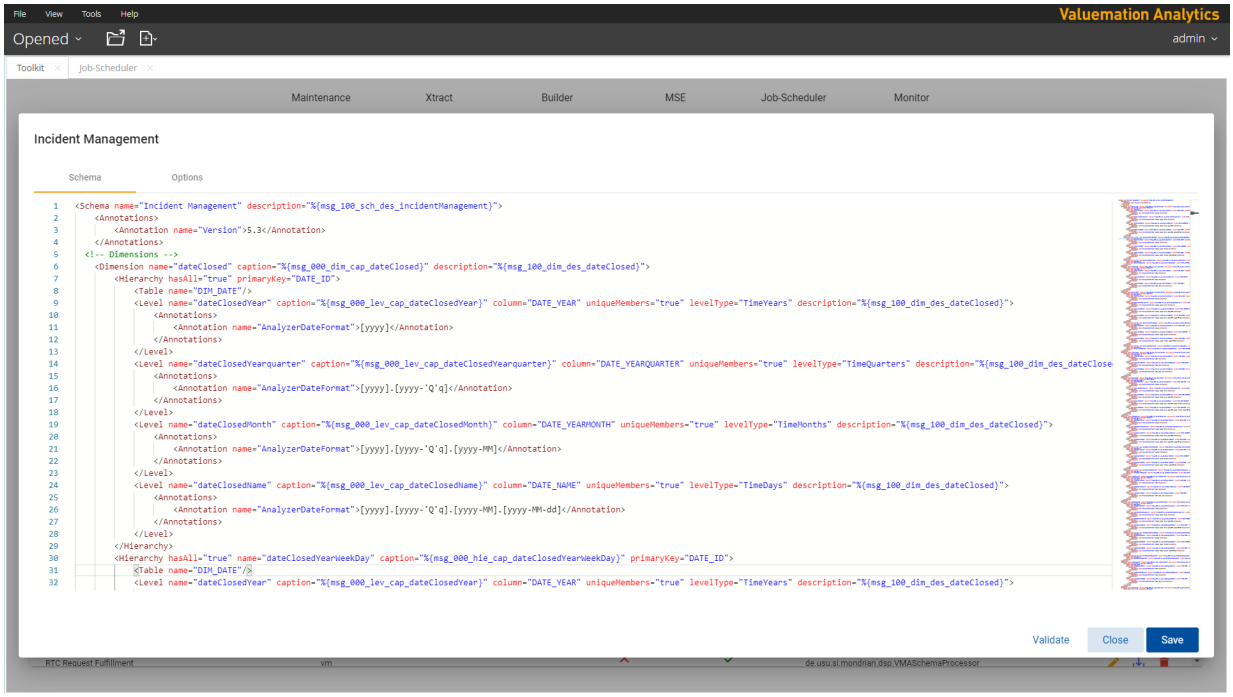
With a double-click you open the editor form for a schema.

5.5.2. The Mondrian Schema Details Form

With this form, you edit a Mondrian Schema and maintain parameters.

Object	Description
Schema tab	On the Schema tab you see the Mondrian schema in an XML editor. Here you can edit your Mondrian Schema. The name of the Mondrian schema **cannot** be changed.
Options tab	On this tab you set the parameters, the Mondrian schema needs to work: • JDBC Data source: Select from a list of defined JDBC data sources. • Dynamic Schema Processor: Define the Dynamic Schema processor, which is used for translating the labels of the schema and the security integration. • Enable Xmla: This must be set to false (unchecked). We have the possibility to change this value for compatibility reasons. • Use content checksum: This must be set to true (checked). We have the possibility to change this value for compatibility reasons.
Validate button	Press this button to validate your Mondrian schema. In case of a failed validation the findings are shown on the right of the screen.
Close button	Press this button to close the form.
Save button	Press this button to save. Before saving, the schema is automatically validated. When validation failed, a message pops up and you can decide, whether to save the invalid file or not.

5.5.3. The XML Editor



In Mondrian Schema Editor we integrated a simple XML editor. To make working with it more comfortable, we implemented these functionalities:

- Syntax highlighting, row numbers, search and replace
- Auto completion (intellisense) for XML tags, extracted tables and columns
- Format XML
- Minimap (code outline)

Syntax highlighting, row numbers, auto completion and minimap are features, which are integrated into the editor window itself:

Feature	Description
Syntax highlighting	The XML tags are highlighted to make the text more readable.
Row numbers	Editor shows row number of code line.
Auto completion (intellisense)	If you start typing, the editor gives you hints and opens an auto completion form. The tool tries to auto complete the text and uses for this a list of XML tags, extracted tables and columns.
Search and replace	If the cursor is placed in the XML editor field, you can start a search by typing CTRL-f or start a replace dialog by typing CTRL-h.
Minimap (code outline)	A Minimap (code outline) gives you a high-level overview of your XML document, which is useful for quick navigation and code understanding. An XML Minimap is shown on the right side of the editor. You can click or drag the shaded area to quickly jump to different sections of your code.

5.5.4. How To

This is a collection of common use cases. It covers most of the functions necessary to create and maintain Mondrian schemas.

5.5.4.1. Create new Mondrian Schema

To create a new Mondrian Schema, please follow these steps:

1. Open the User Console **Home** screen. Go to Toolkit and open **Mondrian Schema Editor**.
2. Open the more menu and choose **Create new Mondrian Schema**. A new form opens.
3. Fill in the Name field and set the other fields with the correct values.
   - JDBC Data source: Select from a list of defined JDBC data sources.
   - Dynamic Schema Processor: Define the Dynamic Schema processor, which is used for translating the labels of the schema and the security integration.
   - Enable Xmla: This must be set to false (unchecked). We have the possibility to change this value for compatibility reasons.
   - Use content checksum: This must be set to true (checked). We have the possibility to change this value for compatibility reasons.
4. Click **Create** to go on.
5. The edit Mondrian Schema form opens automatically, and you can start implementing your schema.
6. Click **Save** to save your changes.

5.5.4.2. Edit a Mondrian Schema

To edit a Mondrian Schema, please follow these steps:

1. Open the User Console **Home** screen. Go to Toolkit and open **Mondrian Schema Editor**.
2. Search for the Mondrian Schema you want to edit. Click on the edit icon on the right of the grid or simply double-click on the line.

   The **Mondrian Schema Editor** details form opens.
3. Even though we offer the Mondrian Schema Editor, the adaptation is mostly manual work. We offer trainings on this topic. Please contact us for more information.

   Detailed information about this topic is available here: https://mondrian.pentaho.com/documentation/schema.php
4. Click **Save** to save your changes.

5.5.4.3. Validate your Mondrian Schema

To detect errors in you Mondrian Schema, please follow these steps:

1. Open the User Console **Home** screen. Go to Toolkit and open **Mondrian Schema Editor**.
2. Search for the Mondrian Schema you want to validate. Click on the edit icon on the right of the grid or simply double-click on the line.

   The **Mondrian Schema Editor** details form opens.
3. Click the validate button. In case of a failed validation the findings are shown on the right of the screen.

   Change you Mondrian schema XML and validate again.
4. Click **Save** to save your changes.

5.5.4.4. Localization and Internationalization of Schemas

The Mondrian Schemas we deliver as standard are prepared for localization. Labels and descriptions of dimensions and measures are parametrized so that the Dynamic Schema Processor (DSP) can replace the values according the chosen language.

As an example, we use dimension service from Incident Management cube:

```
<Dimension name="service" caption="%{msg_000_dim_cap_service}" description="%
{msg_100_dim_des_service}">
        <Hierarchy hasAll="true" primaryKey="SERVICE_ID">
```

```
            <Table name="DIM_SERVICE"/>
            <Level name="serviceName" caption="%{msg_000_lev_cap_serviceName}"
column="SERVICE_NAME"
                   captionColumn="SERVICE_NAME_DF" ordinalColumn="SERVICE_NAME_DF"
                   description="%{msg_100_dim_des_service}"/>
        </Hierarchy>
</Dimension>
```

You can see, that the attributes caption and description are defined in a special way. The Dynamic Schema Processor searches for this formatting %{YOUR_VARIABLE} and replaces it during runtime with the value it finds in the Mondrian Schema translations for the relevant language. If it doesn't find a value for the relevant language; it uses the fall-back language (English). And in case it finds nothing, it shows the name of the variable.

So, the normal workflow is:

1. Open the User Console **Home** screen. Go to Toolkit and open **Mondrian Schema Editor**.
2. Search for the Mondrian Schema you want to edit. Click on the edit icon on the right of the grid or simply double-click on the line.

   The **Mondrian Schema Editor** details form opens.
3. Edit the Mondrian Schema, e.g. add or change dimensions or measures.
4. Replace the values in caption and description with variables (format is %{YOUR_VARIABLE}).
5. Save your changes.
6. Search for the Mondrian Schema from which you want to extract the translation variables from. Click on the Extract translation variables icon on the right of the grid.
7. Open Mondrian Translations application and fill in the needed translations. You can find not translated values by checking the option Show only empty values.
8. Open Analyzer Report. Use the changed Mondrian Schema. The implemented values should be translated. If not, please refresh Mondrian cache before debugging.

   A service restart is not needed.

5.5.4.5. Download Mondrian Schema

To download a Mondrian Schema, please follow these steps:

1. Open the User Console **Home** screen. Go to Toolkit and open **Mondrian Schema Editor**.
2. Search for the Mondrian Schema you want to download. Click on the download icon on the right of the grid.
3. The system asks you, where to save the export file. After the selection of the destination folder the export starts.

5.5.4.6. Import Mondrian Schema

To import a Mondrian Schema, please follow these steps:

1. Open the User Console **Home** screen. Go to Toolkit and open **Mondrian Schema Editor**.
2. Open the more menu and choose **Import Mondrian schema**. A new form opens.
3. Browse for the Mondrian Schema XML file in your file structure and set the other fields with the correct values.
   - JDBC Data source: Select from a list of defined JDBC data sources.
   - Dynamic Schema Processor: Define the Dynamic Schema processor, which is used for translating the labels of the schema and the security integration.
   - Enable Xmla: This must be set to false (unchecked). We have the possibility to change this value for compatibility reasons.

- Use content checksum: This must be set to true (checked). We have the possibility to change this value for compatibility reasons.

4. Click **Import** to go on.

   If a schema with the same name already exist in your repository, the system asks you for permission to overwrite this schema. Answer the question with **Yes** to proceed.

   The schema will now be imported. A message will inform you whether the import was successful or not.

5.5.4.7. Download All

To download all Mondrian Schemas in one step, please follow these steps:

1. Open the User Console **Home** screen. Go to Toolkit and open **Mondrian Schema Editor**.
2. Open the more menu and choose **Download all**.
3. The system asks you, where to save the export file. After the selection of the destination folder the export starts.

5.5.4.8. Delete Mondrian Schema

To delete a Mondrian Schema, please follow these steps:

1. Open the User Console **Home** screen. Go to Toolkit and open **Mondrian Schema Editor**.
2. Search for the Mondrian Schema you want to delete. Click on the delete icon on the right of the grid.
3. The system asks you for permission to delete the Mondrian schema. Answer the question with **Yes** to proceed.
4. The Mondrian schema is deleted permanently.

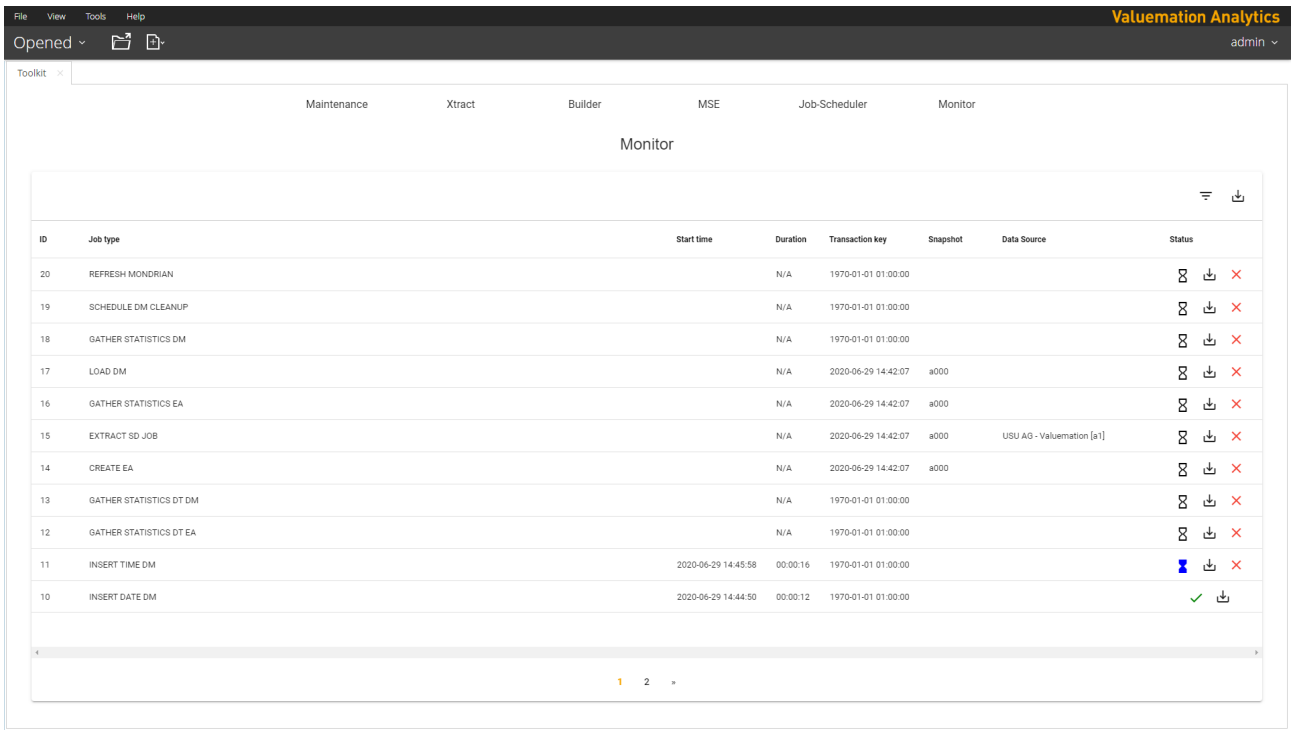
Deletion of Mondrian Schemas is irreversible. There is no roll back possible. Please make sure you choose the correct Mondrian Schema.

Any content associated with this Mondrian Schema (data source), such as analyses, dashboards and reports, will no longer work.

5.6. DLO and DLO protocols

See chapter Data Load Organizer (page 122).

5.7. Monitor



5.7.1. Overview

The jobs in Toolkit are organized in a queue. With Job-Scheduler or manually from Toolkit you simply add global jobs to the queue. They are executed in the order they were created.

In Monitor module, basic information about global jobs is listed:

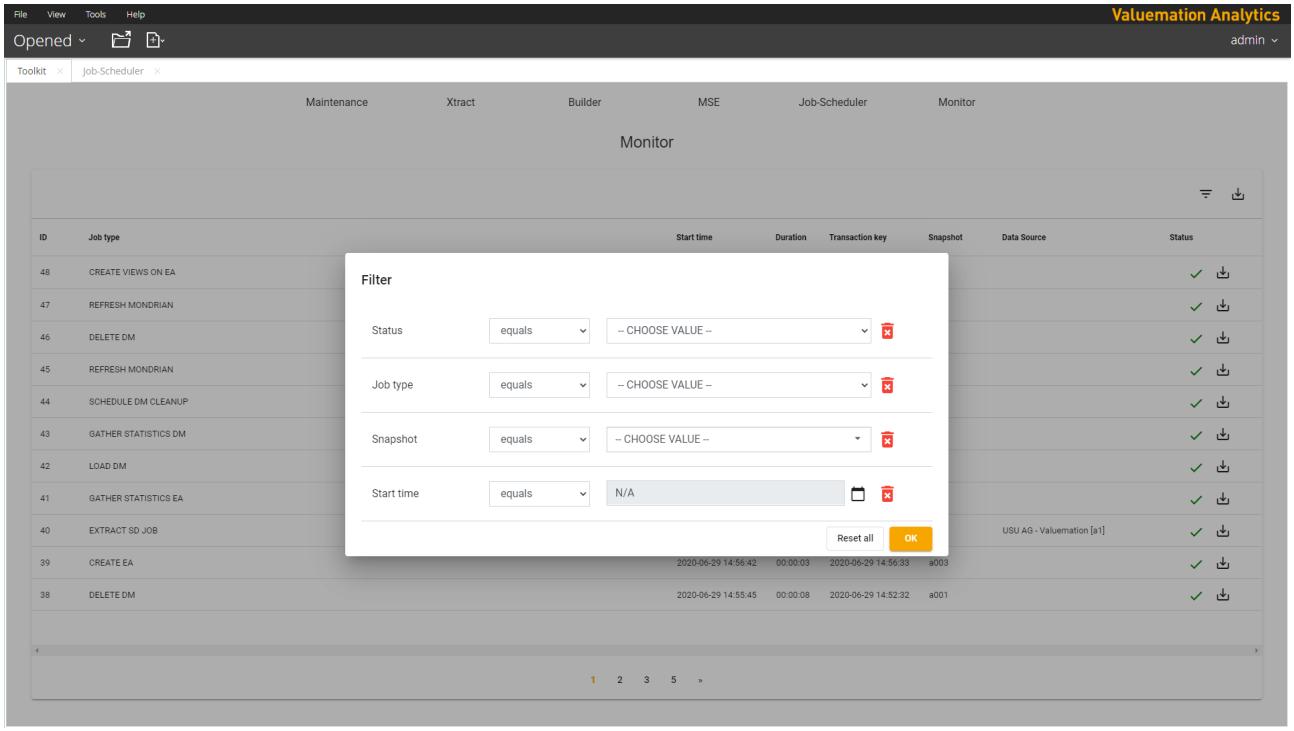
- Job ID: Internal system generates ID of the job
- Job type: Name of the job
- Start time: Date and time when execution started
- End time: Date and time, when the execution ended
- Transaction key
- Snapshot tag:
- Status: Execution status of the global job

- If you click on this icon  you can download the log information for this global job
- If a job shows an X button in the Status column you can cancel this job. The job is canceled immediatelly – there is not security question.

The grid is refreshed automatically every second.

In Monitor, we provide you information about the last 25 global jobs. If you need information about all global jobs, sub jobs, more detailed status information, warnings and error messages, you can export the queue to an XML file with the **Export jobs** option. You could use the file for debugging or send it to support.usu.de in case of an error.

5.7.2. Filter



Click the button **Filter** to open the filter form to add, change and reset the filter for the Monitor grid. If a filter is active, a little green active sign is displayed on the Filter button.

5.7.3. The Options Menu

5.7.3.1. Reload Data

If you click **Reload data** in **Options** menu, the grid is reloaded from database.

5.7.3.2. Export Jobs

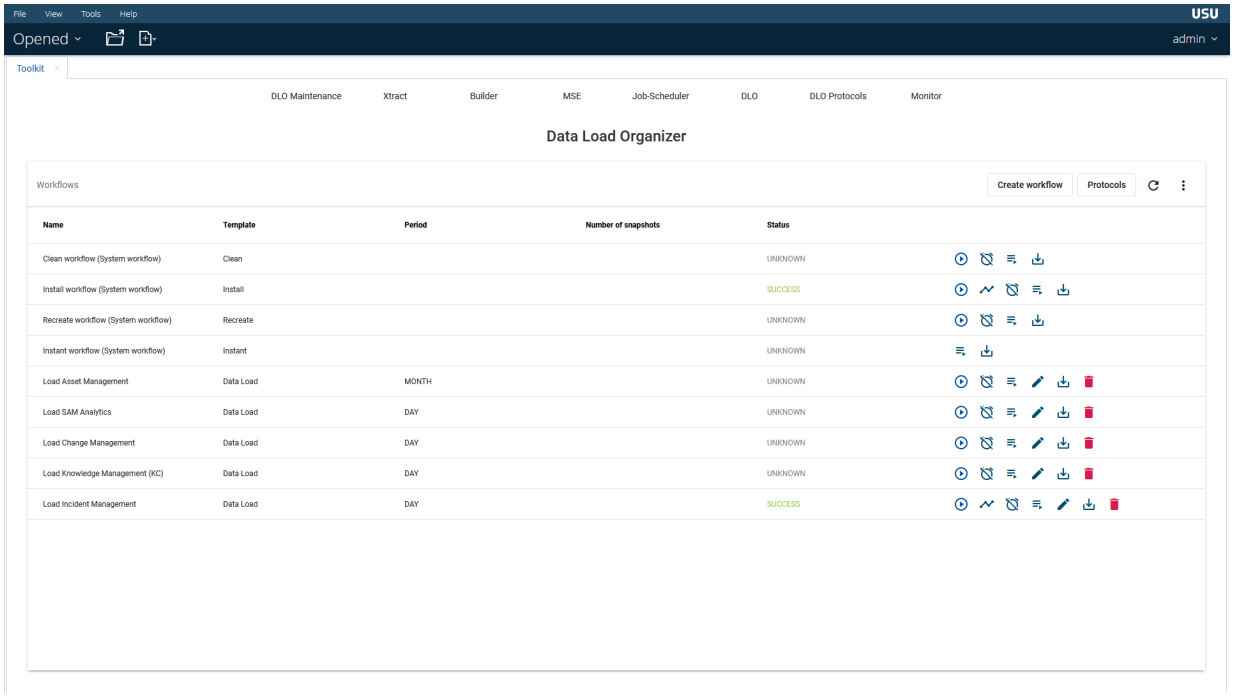
With this option you can export the detailed job information. In Monitor grid the global jobs are listed only. In the export file, every global job including its sub jobs is listed. Here you find also more detailed status information, warnings and error messages. You can use it for debugging reasons and send it to support.usu.de in case of an error. The export file is formatted in XML.

To export job information please follow these steps:

1. Click **Export jobs** in **Options** menu, a new dialog opens.
2. Click **Export last 25 global jobs** or **Export all jobs** button.
3. The system asks you, where to save the monitoring export file. After the selection of the destination folder the export starts.

Export of complex jobs with multiple SQL sub jobs can take several minutes.

6. Data Load Organizer



6.1. Overview

Data Load Organizer (DLO) is an application for automatic scheduling and execution of your data loads. The module can be called from the toolkit. It uses the information from Xtract, Builder and Data Source Management to build the extraction area and fill the data mart.

The main functionalities are:

- Organize your data load schedules within User Console.
- You can schedule data loads from multiple data sources.
- Have different granularities for your fact tables, e.g. load some on daily basis, some every 2 weeks.
- You can create, edit, activate, deactivate and delete your data load workflows.
- Define your retention policy rules in a granular way.
- DLO workflow definitions can be exported and imported.
- All runs are logged in so called protocols.
- You can use filters to search in DLO protocols.

6.1.1. DLO vs Job-Scheduler ETL job

You must not run Data Load Organizer and Job-Scheduler ETL loads in parallel. This will lead to data corruption and malfunctioning of the system.

DLO workflows and Job-Scheduler ETL jobs are basically doing the same. They load data from a source system into the USU Analytics database.

Job-Scheduler has some weaknesses:

- It is not easily possible to implement data loads for different cubes with different granularity, e.g., the incident data daily and asset data once per month.
- It is not possible to load data from different data sources and define a different number of snapshots per data source, e.g., 1 snapshot for USU Service Management and 50 snapshots for USU License Management.
- It is not possible to mix different data load types, e.g., use Standard Mode, Staging Mode and CDC/Delta Load in one installation.

With DLO we overcome these problems.

In the medium term, the ETL load jobs of the job scheduler are set to deprecated. In the long term, this feature will no longer be officially supported.

We deliver a separate documentation for DLO in this release. In the next releases this will be integrated in the administration manual.

6.1.2. Migration from Job-Scheduler ETL jobs to DLO

6.1.2.1. Migration to DLO if data in DM and EA can be truncated

Steps needed to upgrade from old loader:

1. Make a complete backup of your system including application server and all databases
2. In Scheduler deactivate old ETL job used for data load
3. Execute action
    1. Drop DM (without date and time tables)

    After applying this step it will not be possible for users to generate reports and use Analyzer Report until migration procedure is finished

    2. Drop EA
    3. Create DM
4. Activate DLO in Toolkit -> System Configuration -> Toolkit -> Enable DLO
5. Open DLO workflow with name Standard load <NAME OF PRODUCT> in edit mode
    1. Set trigger of DLO workflow to same value as trigger in old ETL job
    4. Set retention policy to the same values as Fact Table Load Parameters (can be found in Toolkit → Global Config)
    5. Activate this workflow and save it
    6. Execute the workflow to reload data
6. Wait until DLO workflow from step 4 finishes and verify that data are loaded in DM by opening Analyzer Report
7. System is now set up to execute loads automatically with DLO and migration is finished

6.1.2.2. Migration to DLO if data in DM and EA cannot be truncated

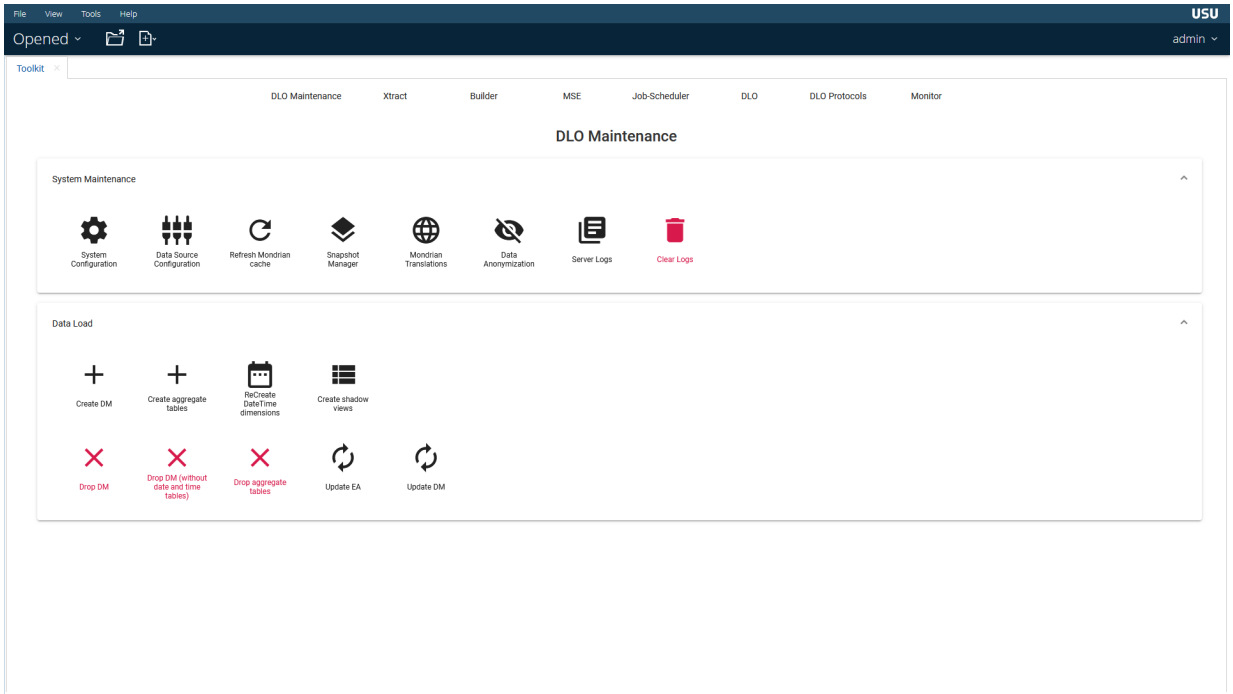
Steps needed to upgrade from old loader to DLO:

1. Make a complete backup of your system including application server and all databases
2. Create a backup of MD, EA and DM database
3. In Scheduler bdeactivate old ETL job used for data load
4. Execute following action on DM database - This action will assign new Data Load workflow with name “Standard load <PRODUCT_NAME” to old existing snapshots.

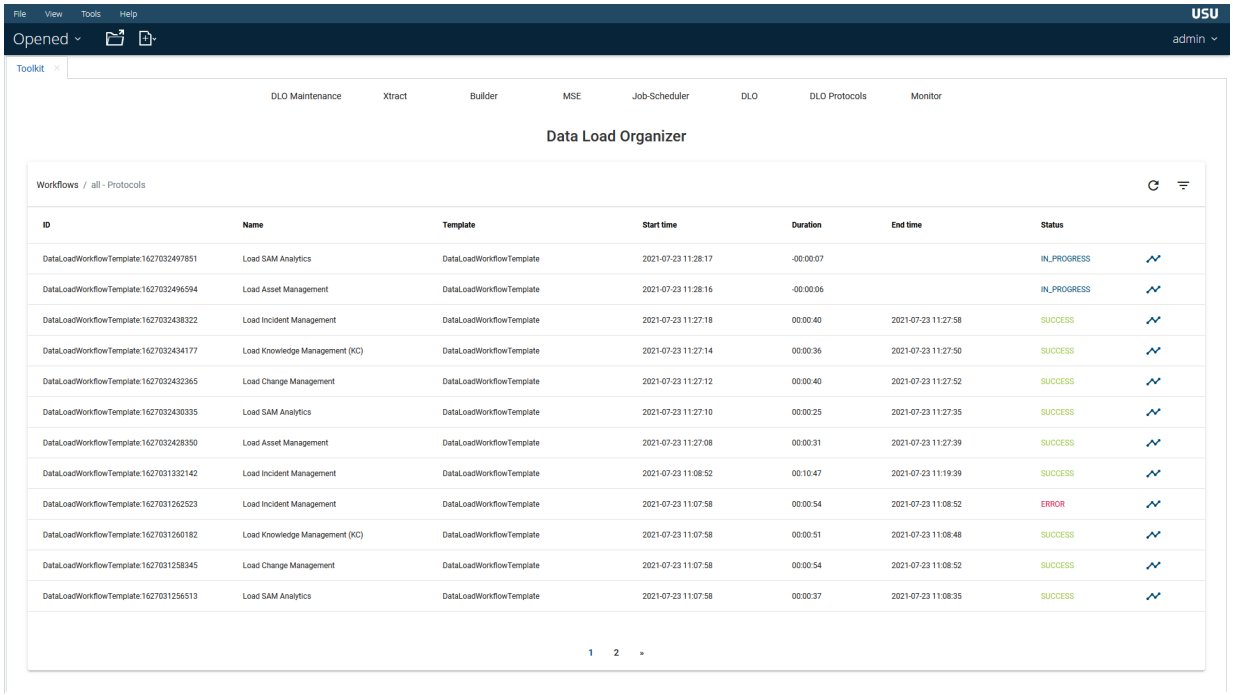
```
UPDATE sil_snapshot
SET
    dlo_configuration_id = (SELECT
            MIN(configuration_id)
        FROM
            vma_md.sio_configuration
        WHERE
            template = 'DATA_LOAD'
        ORDER BY configuration_id)
WHERE
    status IN (3 , 4);
```

5. Activate DLO in **Toolkit > System Configuration > Toolkit > Enable DLO**
6. Open DLO workflow with name Standard load <NAME OF PRODUCT> in edit mode
7. Set trigger of DLO workflow to same value as trigger in old ETL job
8. Set retention policy to the same values as Fact Table Load Parameters (can be found in Toolkit → Global Config)
9. Activate this workflow and save it
10. Execute the workflow to reload data
11. Wait until DLO workflow from step 4 finishes and verify that data are loaded in DM by opening Analyzer Report
12. System is now set up to execute loads automatically with DLO and migration is finished

6.2. DLO Maintenance



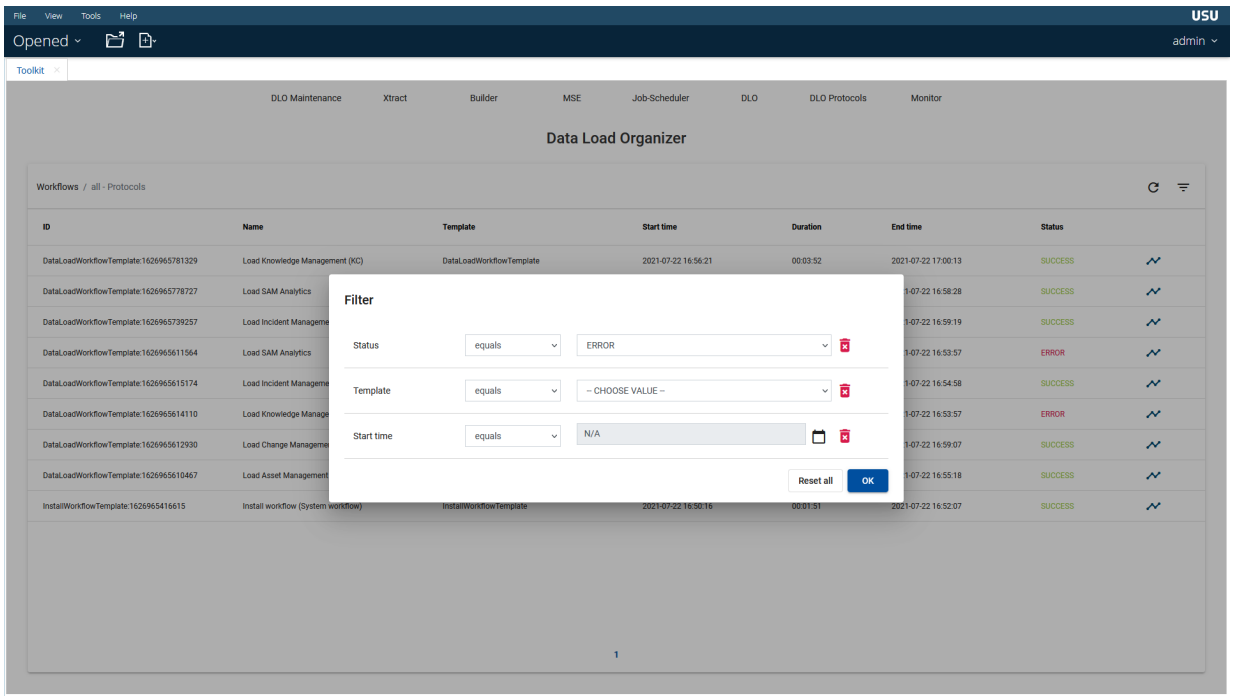
6.3. DLO Protocols



The DLO Protocols form displays all protocols of all workflows executed after the latest clear workflow job.

- You see information about name, used template, start and end time and status of the workflow execution.
- The table is ordered by start time.
- By clicking on the chart icon, you can open a detailed protocol.
- To refresh the displayed information click the refresh button.

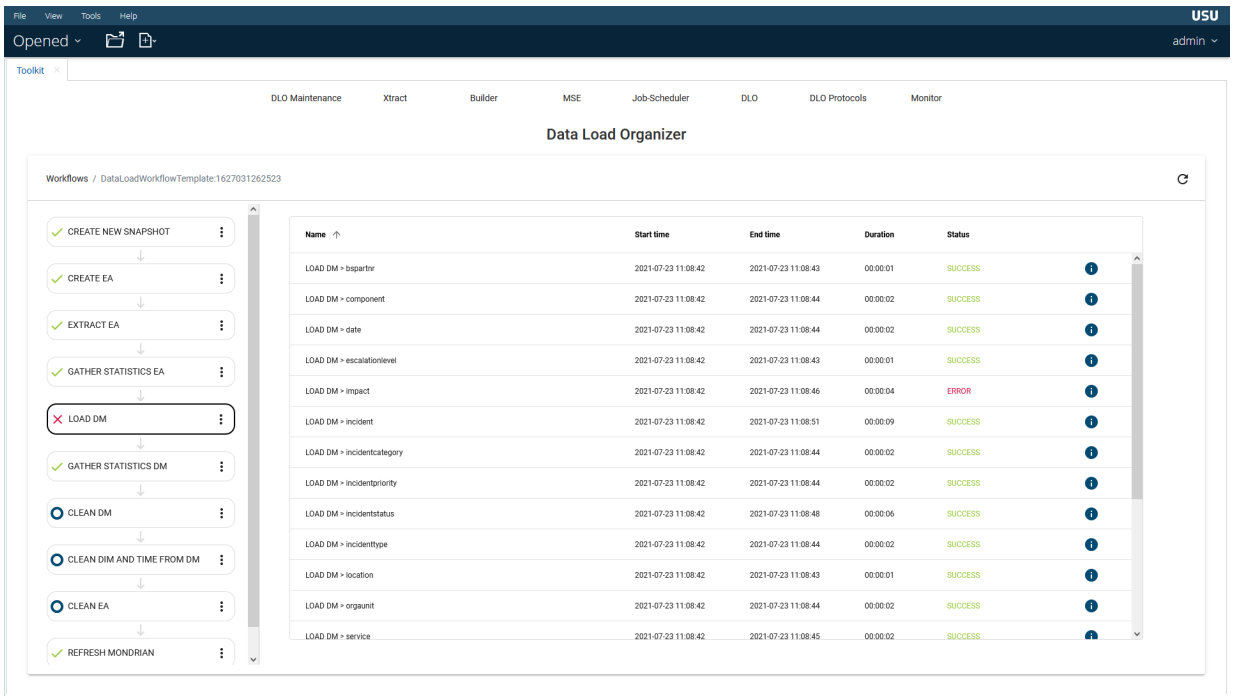
6.3.1. Use filters to search in DLO protocols



To filter protocols, follow these steps:

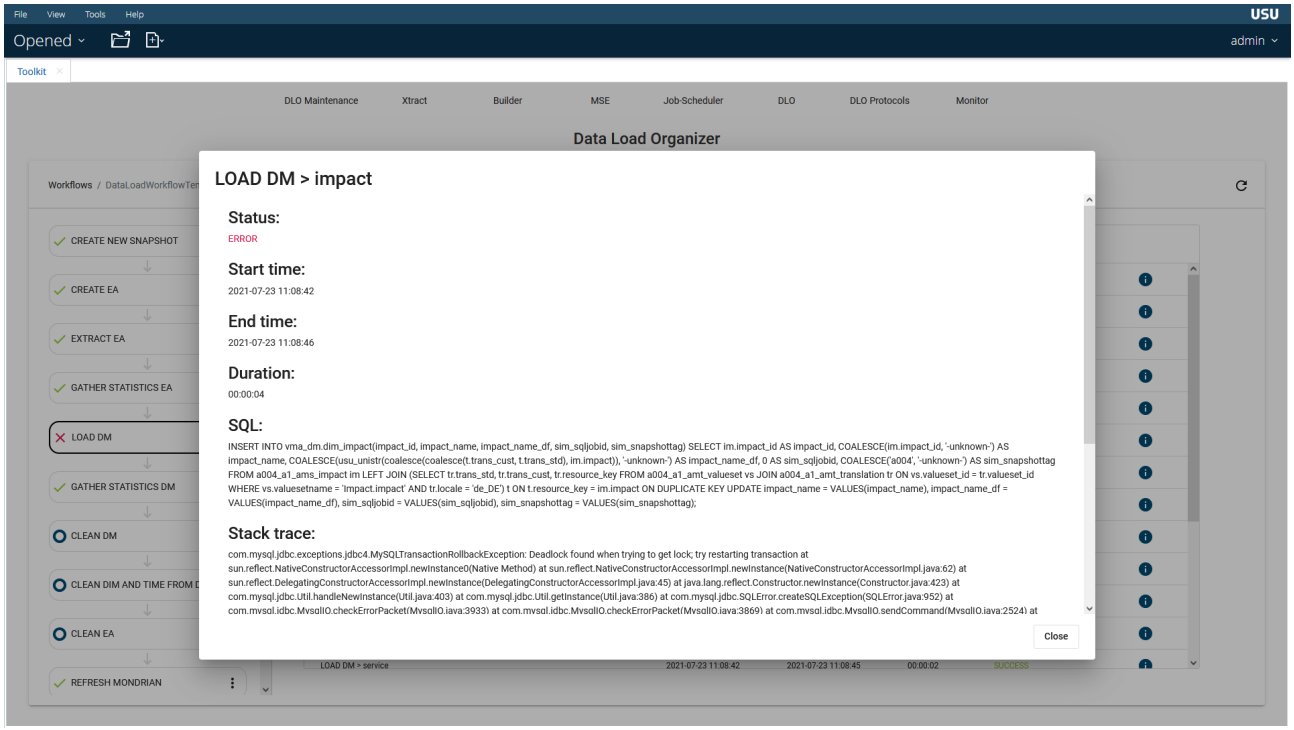
1. Open DLO application.
2. Open Protocols.
3. Click the **filter icon**, a new dialog opens.
4. You can set filter criteria and values for Status, Template and Start time.
5. Click **Ok** and the filtering is activated. An active filter is indicated by the **orange color** of the filter icon.
6. To disable filters, click on the **filter icon** and use the red **Delete** button behind a filter criterion to disable that particular filter, or use the **Reset all** button to disable all filters. Click **Ok** to activate your changes.

6.3.2. Workflow protocol details



To access protocol details of a workflow execution, follow these steps:

1. Open DLO application.
2. Open Protocols.
3. If needed use filters.
4. By clicking on the chart icon, you can open a detailed protocol.
5. On the **left** you see a **list of executed jobs** within the workflow. The icon indicates the overall status of the job (success, error, skipped, in progress)
6. If you click on one of the jobs, the **right** part of the form shows **sub jobs associated** to the main job.
   a. The status column indicates the status of the sub job (success, error, skipped, in progress). This information is used to calculate the overall status of the main job.
   b. By default, the view is ordered by name column. Click on another header to change the order.
   c. If you click on the i icon you get even more detailed information.



6.4. Workflow types

The system provides two types of workflow: System and Load workflows.

System workflows are used to execute jobs necessary for system maintenance. They are maintained by USU developers and can not be changed by users. System workflow cannot be exported.

Load workflows can be defined by user and exported to deploy and backup.

6.4.1. System Workflows

The system provides following system workflows: Clean. Install, Recreate and Instant workflow

6.4.1.1. Install workflow

The install workflow installs system like defined in Builder definitions. When at least one workflow uses staging mode, the system also prepares data objects for staging mode.

6.4.1.2. Clean workflow

The Clean workflow deletes all data from all tables in DM, including the date and time dimension data. Then it drops all EA tables and makes a cleanup of snapshot maintenance tables.

6.4.1.3. Recreate workflow

The recreate workflow:

- Drops, creates and inserts data in date/time tables in EA
- Drops, creates and inserts data in date/time tables in DM
- Drops and creates DM
- Gathers statistics

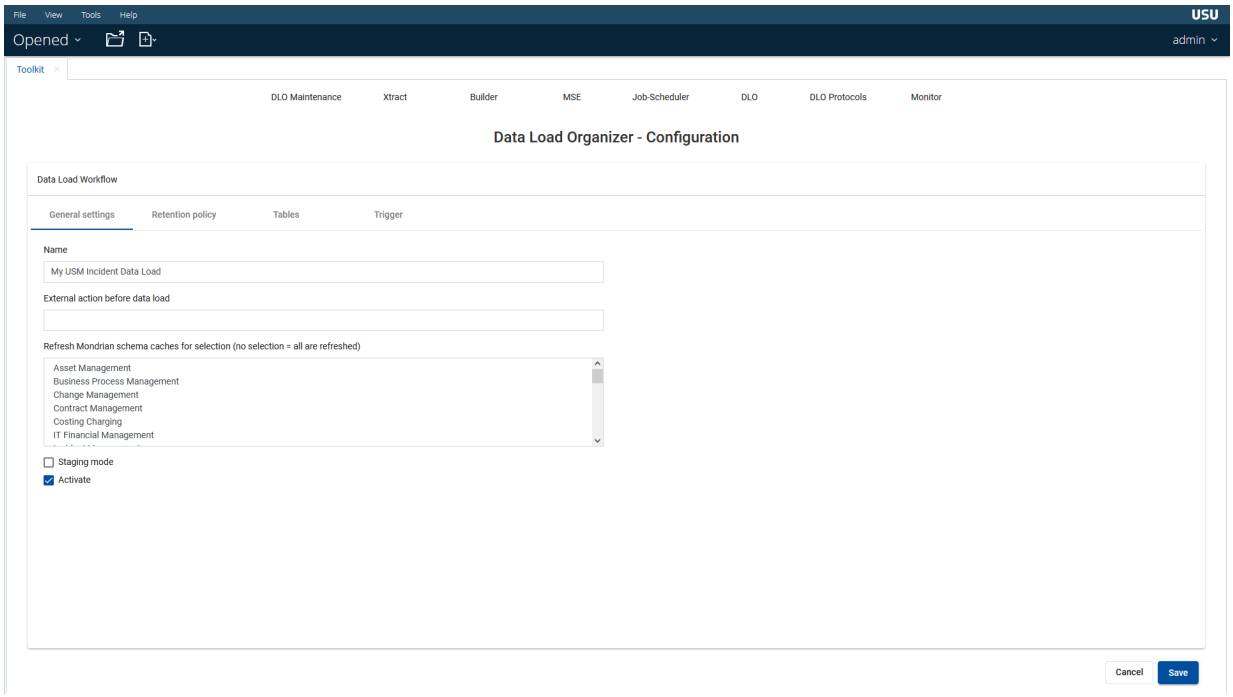
6.4.1.4. Instant workflow

Instant workflows are technical workflows used by the application, e.g., to start the execution of actions triggered on the DLO Maintenance screen. You can view details in DLO protocols form.

6.4.2. Load Workflows

In this version we deliver four types of data load workflows: Data Load, CDC, CBD and Columnstore Load. Please make yourself familiar with these data load modes.

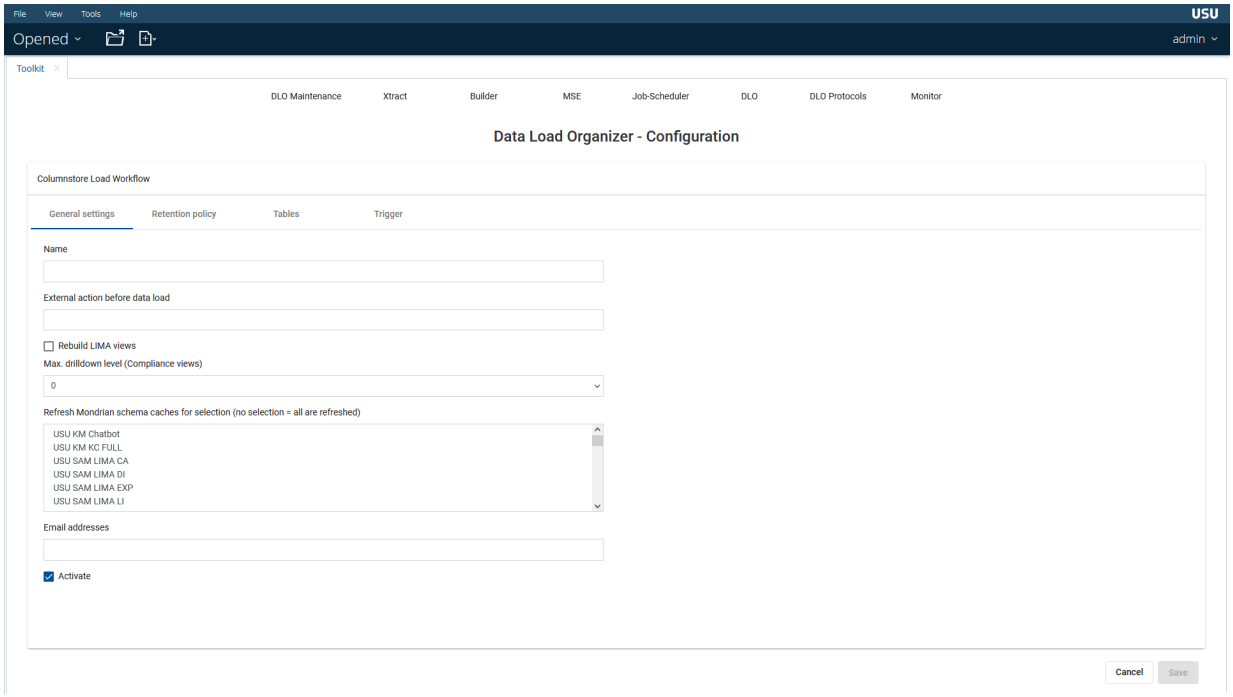
6.4.2.1. Data Load



Object	Description
Name	User given name of the workflow
External action before data load	Some application needs to prepare data before USU Analytics can import it, e.g., USU SAM Analytics starts data exports using a bat or sh script. Specify the place the script is stored on the server. It will be executed during the run of the workflow.
Refresh Mondrian schema caches for selection (no selection = all are refreshed)	Select the Mondrian schema cache you want to refresh after data is loaded. If you have no value selected all Mondrian schema caches are refreshed.
Staging mode	Check this to use staging mode for this data load. If not checked, the system used standard mode. See administration manual for details about standard and staging mode.
Active	Only active workflows are started automatically using the trigger information.
Retention policy tab	On **tab Retention policy** you setup your retention policy for this load. For details about retention policies see Retention policy rules (page 137).
Tables tab	On **tab Tables** you choose the tables you want to load in this workflow. You select the fact tables. The system computes the relevant extractions and dimension tables.

Object	Description
Trigger tab	On **tab Trigger** you define the start time and time interval of your workflow execution. Use Minute, Hourly, Daily, Weekly, Monthly or Yearly section to define the most common time intervals. To enter a custom time interval, choose Advanced and consult Quartz Cron Attributes (page 162) to learn the Quartz Cron syntax in administration manual.

6.4.2.2. Columnstore Load



This load mechanism is optimized for MariaDB ColumnStore database. It utilizes MariaDB ColumnStore database specific functions and executables and will fail in other DBMS.

Object	Description
Name	User given name of the workflow
External action before data load	Some application needs to prepare data before USU Analytics can import it, e.g., USU Analytics for Software Asset Management starts data exports using a bat or sh script. Specify the place the script is stored on the server. It will be executed during the run of the workflow.
Rebuild LIMA views	*When this is checked, the Software Asset Management API is called directly from Analytics backend and the data is prepared.* *To use this functionality, you have to provide the* • *LIMA SOAP URL,* • *LIMA interface user login,* • *LIMA interface user password* *in the data source connection definition to LIMA (st).*
Max. drilldown level (Compliance views)	*Compliance views drill-down level allows the known options: 0,1,2.* *This is only used, when Enable LIMA refresh view is checked.*

Object	Description
Refresh Mondrian schema caches for selection (no selection = all are refreshed)	Select the Mondrian schema cache you want to refresh after data is loaded. If you have no value selected all Mondrian schema caches are refreshed.
Email address	In case of an error this mail address will be notified.
Active	Only active workflows are started automatically using the trigger information.
Retention policy tab	On **tab Retention policy** you setup your retention policy for this load. For details about retention policies see **Retention policy rules**.
Tables tab	On **tab Tables** you choose the tables you want to load in this workflow. You select the fact tables. The system computes the relevant extractions and dimension tables.
Trigger tab	On **tab Trigger** you define the start time and time interval of your workflow execution. Use Minute, Hourly, Daily, Weekly, Monthly or Yearly section to define the most common time intervals. To enter a custom time interval, choose Advanced and consult **Quartz Cron Attributes** to learn the Quartz Cron syntax in administration manual.

6.4.2.2.1. Requirements

Basic Skills and Knowledge

To use Columnstore load properly the responsible user must be familiar with some technical terms and concepts:

- Knowledge about the specifics of a MariaDB ColumnStore database system.
- Basic Business Intelligence knowledge is particularly advantageous. Terms and concepts like fact, dimension, degenerated dimension, measure are assumed.

Software components and versions

To use Columnstore load in USU Analytics it is needed to have following software components:

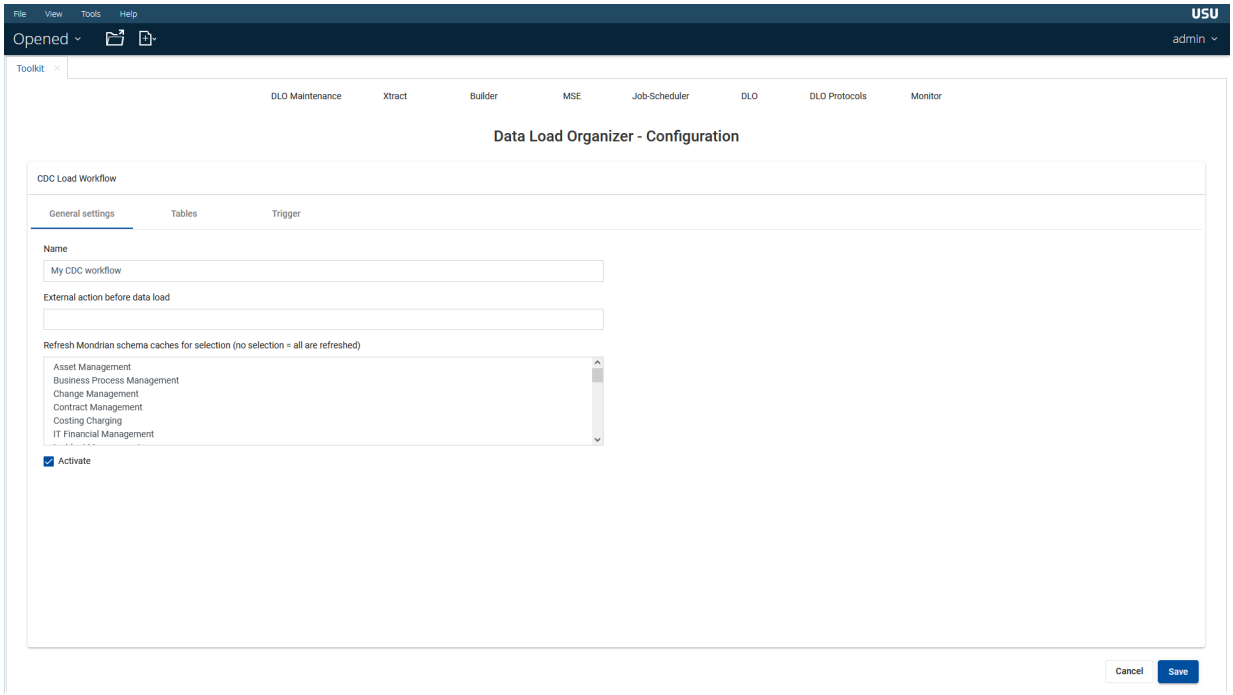
- USU Analytics 10.1.2 or higher
- This load mechanism is optimized for MariaDB ColumnStore database. It utilizes MariaDB ColumnStore database specific functions and executables and will fail in other DBMS.

  So, running USU Analytics on a MariaDB ColumnStore database is a must.

USU Analytics configuration

- In the system configuration, the value COLUMNSTORE must be selected for the expert parameter "Staging mode database engine for MariaDB".
- The staging mode **must not** be activated for any load workflow.

6.4.2.3. CDC Load



Object	Description
Name	User given name of the workflow
External action before data load	Some application needs to prepare data before USU Analytics can import it, e.g., USU SAM Analytics starts data exports using a bat or sh script. Specify the place the script is stored on the server. It will be executed during the run of the workflow.
Refresh Mondrian schema caches for selection (no selection = all are refreshed)	Select the Mondrian schema cache you want to refresh after data is loaded. If you have no value selected all Mondrian schema caches are refreshed.
Active	Only active workflows are started using the trigger information.
Retention policy tab	On **tab Retention policy** you setup your retention policy for this load. For details about retention policies see Retention policy rules (page 137).
Tables tab	On **tab Tables** you choose the tables you want to load in this workflow. You select the fact tables. The systems computes the relevant extractions and dimension tables.
Trigger tab	On **tab Trigger** you define the start time and time interval of your workflow execution. Use Minute, Hourly, Daily, Weekly, Monthly or Yearly section to define the most common time intervals. To enter a custom time interval, choose Advanced and consult Quartz Cron Attributes (page 162) to learn the Quartz Cron syntax in administration manual.

6.4.2.4. CBD Load

The Catalog Based Datasource module (CBD) allows technically skilled users to configure data sources in USU Analytics for Software Asset Management based on USU Service Management catalogs:

- Attribute definition and filtering is done in the USU Service Management catalog
- CBD is integrated into the DLO and thus into the standard loading process
- Mondrian schema is generated automatically

- No need for SQL or XML know-how

6.4.2.4.1. Requirements

Basic Skills and Knowledge

To use USU Analytics for Software Asset Management CBD properly the responsible user must be familiar with some technical terms and concepts:

- Ability to create new catalogs or change existing in USU Service Management.
- Knowledge about the object structure of USU Service Management is beneficial.
- Basic Business Intelligence knowledge is particularly advantageous. Terms and concepts like fact, dimension, degenerated dimension, measure are assumed.

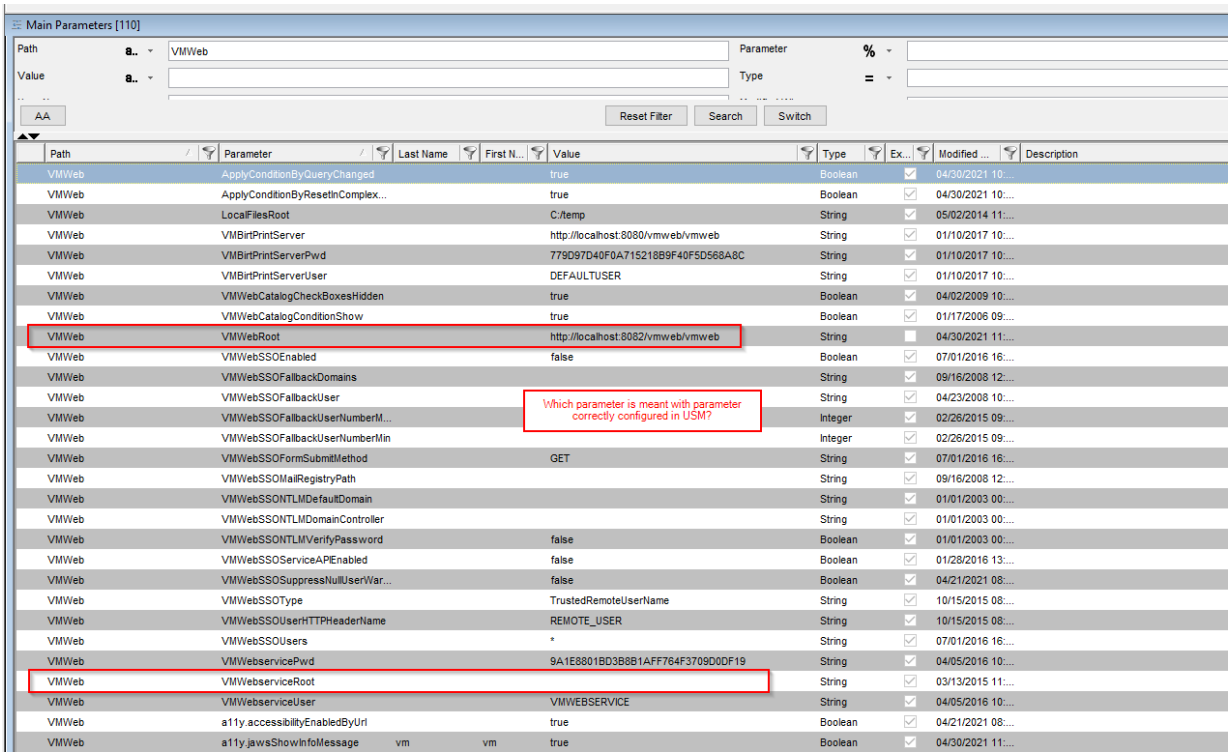
Software components and versions

To use CBD in USU Analytics it is needed to have following software components:

- USU Service Management
- VM51HF30 – without support of historical records
- VM52HF18 – without support of historical records
- VM52HF24 – with support of historical records

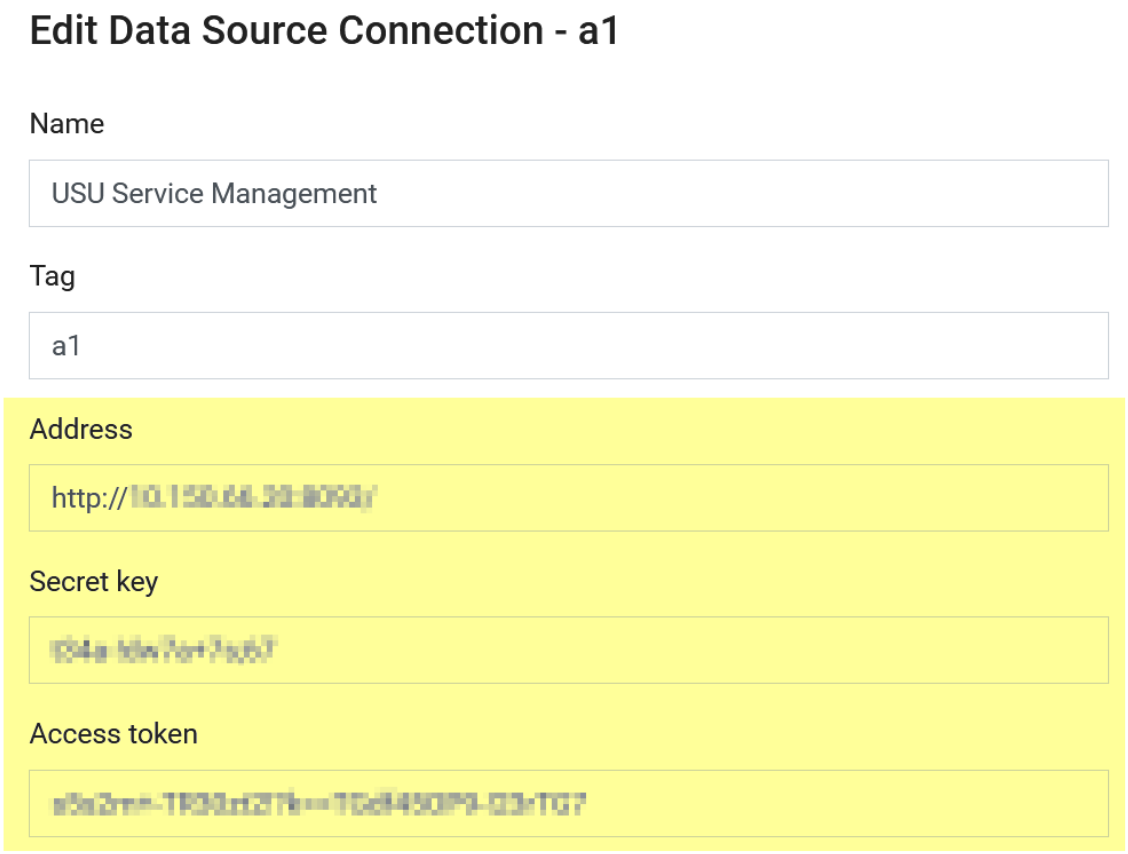
USM configuration

- Main parameter VMWeb VMWebRoot needs to be configured properly. Value of this parameter represents URL under which users' access USU Service Management application server.
- Main parameter VMWeb VMWebserviceRoot needs to be configured properly. Value of this parameter represents URL under which applications access USU Service Management webservices.



USU Analytics configuration

- USU Analytics application server needs to be able to communicate via HTTP/HTTPS protocol with USU Service Management application server and firewalls between these two servers need to be opened.
- Data source A1 (USU Service Management) needs to be configured properly.
- Address: USU Service Management URL e.g., https://usm-app.com/ (must end with /)
- Secret key: Optional - USU Service Management main parameter Webservice secretKey. This parameter must have 16 characters.
- Access token: Optional - USU Service Management Web service client access token in USM Self Service Manager



6.4.2.4.2. CBD form

You define a CBD data load workflow in DLO with CBD definition form.

Object	Description
General setting	
Name	User given name of the workflow. Changing the name when editing can cause duplicate Mondrian schemas.
Email addresses	Notification about load will be send to this address or multiple addresses split by symbol semicolon (;).
Activate	Only active workflows are started automatically using the trigger information.
Automatic Mondrian schema generation	When this is checked, the system generates a Mondrian schema as analytical data source automatically during the data load process.
Datasource configuration	

Object	Description
USU Service Management User	
Catalog name	USU Service Management catalog name (not translated technical)
Object type	USU Service Management object type
Date dimension tab	Add here the attributes you want to define as date dimension. The system filters the attributes for you – you see only attributes in the drop down, which have the data type datetime.
Dgn. dimension tab	Add here the attributes you want to define as dimension. You can choose from all attributes of the catalog, except those with numeric and date types.
Measure tab	Add here the attributes you want to define as measures and how the values of this measures should be aggregated. The system filters the attributes for you – you see only attributes with data types Int, Boolean and Number. You can add attributes from catalog multiple times with different aggregators.
Trigger tab	On tab Trigger you define the start time and time interval of your workflow execution.

6.4.2.4.3. How to

Create new CBD data load workflow

Prerequisites:

- You prepared a USU Service Management catalog to fit your needs
- You have all attributes you need
- The data is filtered
- USU Analytics has access to the USM server
- The user you want to use to read the catalog has access to it

To create a new CBD data load workflow, follow these steps:

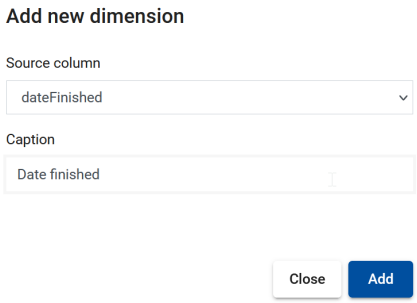
1. Open Data Load Organizer application.
2. Click on **Create workflow** button and choose **CBD**.
3. A new form called **CBD** opens on tab **General settings**.
4. Fill in the field **Name**. It is a user given name and you can change it later.

   Be aware, that imports identify workflows by name.

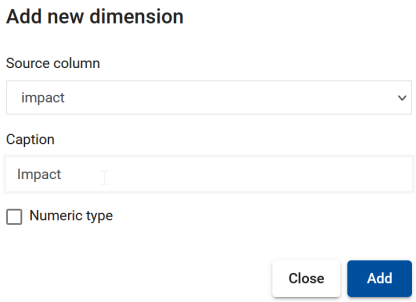
5. Fill in the field **Email addresses**. When errors occur, an email is sent with the xml output of the workflow to the maintained email addresses.
6. The field **Activate** is unchecked by default. Only active workflows are started automatically using the trigger information. Leave it uncheck it when you don't want the workflow started, when the schedule is triggered, e.g., when the workflow is in development state.
7. The field **Automatic Mondrian schema generation** is checked by default. This means, the system generates a Mondrian schema as analytical data source automatically during the data load process. We recommend letting it checked, unless you want to rewrite the Mondrian schema manually.
8. Change to tab **Datasource configuration**.

9. Provide value for **USM User**, which is the USU Service Management User under which context the catalog is opened.
10. Provide value for **Catalog name**, which is the name of the USU Service Management catalog you want to read data from.
11. Specify the corresponding USU Service Management object type in field **Object type**.
12. Select a Data Source – Usually this is a1 – USU Service Management.
13. Change to tab **Date dimension**.
14. To add a new date dimension, click **+** and a modal form opens. Select a **Source column**. The system filters the attributes for you – you see only attributes in the drop down, which have the data type datetime. Provide a caption in the **Caption** field. Click **Add** to add the date dimension



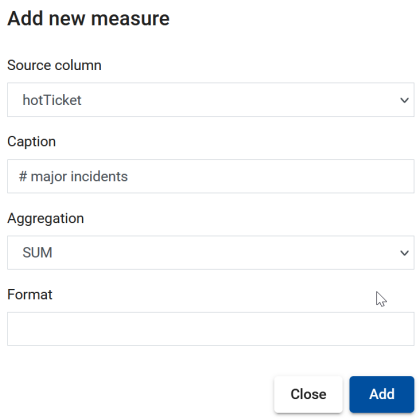
Repeat this until you have all date dimensions defined.

15. Change to tab **Dgn. dimension**.
16. To add a new (degenerated) dimension, click + and a modal form opens. Select a **Source column**. Provide a caption in the **Caption** field. Check checkbox Index, when you want to index this column. Click **Add** to add the dimension.



Repeat this until you have all dimensions defined. The Numeric type would allow you to create a dimension based on a numeric field. This would enable special filtering in Analytics.

17. Change to tab **Measures**.
18. To add a new measure, click + and a modal form opens. Select a **Source column**. Provide a caption in the **Caption** field. Choose from the list of possible functions in field **Aggregation**. Possible values are: `AVG, COUNT, SUM, MIN, MAX, DISTINCT_COUNT`. IN field Format you can add a format definition to your measure.. Click **Add** to add the measure.



Repeat this until you have all measures defined.

19. Change to tab **Trigger**.
20. Here you define the start time and time interval of your workflow execution. Use Minute, Hourly, Daily, Weekly, Monthly or Yearly section to define the most common time intervals. To enter a custom time interval, choose Advanced and consult Quartz Cron Attributes to learn the Quartz Cron syntax in administration manual.
21. Click **Save**.
22. When you now click the **Play** button the CBD data load workflow is triggered immediately. Data is loaded and when finished, the Mondrian schema and the analytical data source is generated automatically (when you have checked this option).
23. You can now open a new Analyzer Report and choose the data source and start analysing.

Maintain CBD workflows

This works in the same way as for the other DLO workflow types.

6.5. Retention policy rules

With retention policy rules you define the snapshots you want to keep in your database (DM and EA). Retention policy rules are defined per workflow.

In this chapter we explain in examples how rules could be configured and what result these rules would have in your database.

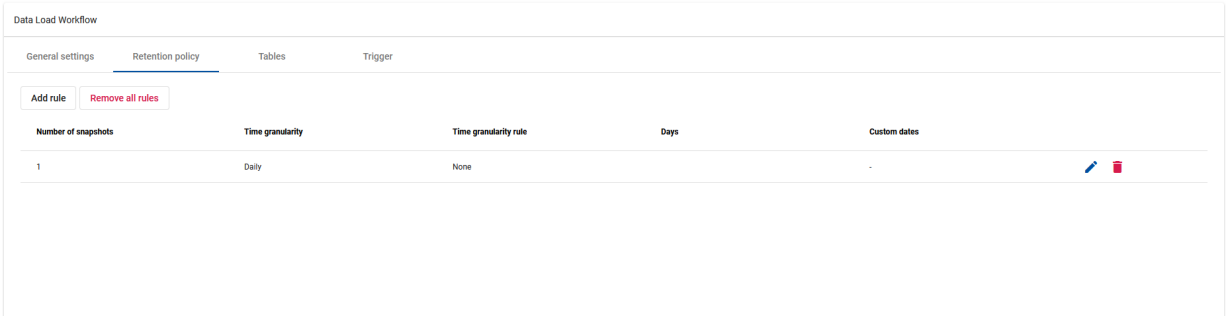
In standard business logic implementation for USU Service Management the Mondrian Schemas expect only one snapshot in DM fact table.

When you configure the system to load multiple snapshots in USU Service Management based fact tables you have to change the Mondrian schemas. If you don't do this your analysis results will be wrong.

On the other hand, USU Analytics for Software Asset Management business logic expects data to be loaded using multiple snapshot configuration.

Means: The setup of your retention policy rules must reflect the implementation in Mondrian schema and the use case.

6.5.1. Use case: Single snapshot

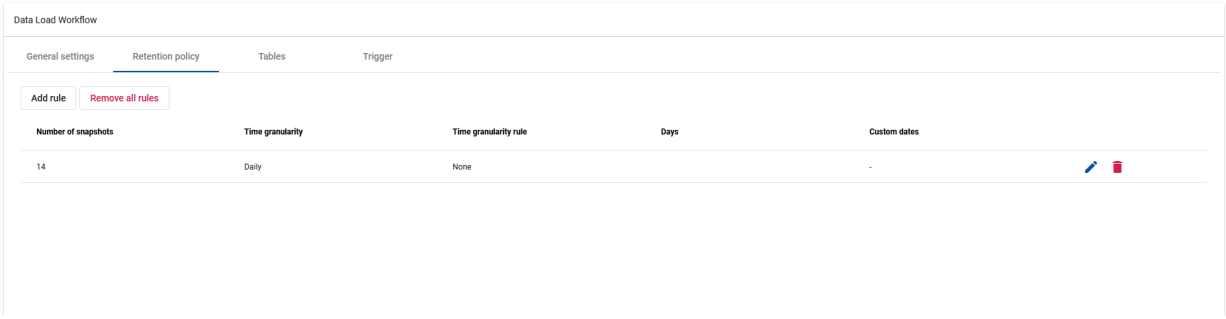


This is the most basic setup and also the default if no retention policy rule is defined.

You define a daily load in Trigger section and a retention policy rule with 1 snapshot and daily time granularity.

When retention policy rules are evaluated, the system deletes all old snapshots, except the latest one.

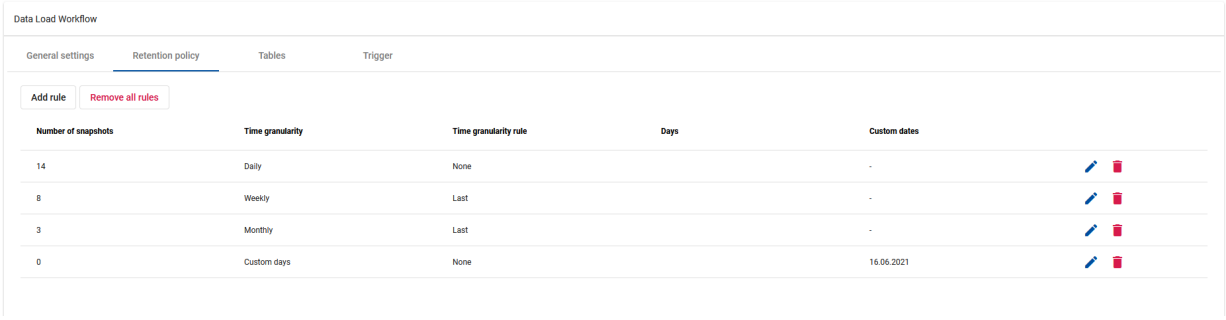
6.5.2. Use case: Multi snapshot



You define a daily load in Trigger section and a retention policy rule with 14 snapshots and daily time granularity.

When retention policy rules are evaluated, the system deletes all snapshots older then 14 days. Basically, the system creates a moving window of 14 snapshots.

6.5.3. Use case: Multi snapshot and combined rules



You defined a daily load in Trigger section and following retention policy rules:

1. **14** snapshots and **Daily** time granularity
2. **8** snapshots and **Weekly** time granularity and **Last** as time granularity rule
3. **3** snapsots and **Monthly** time granularity and **Last** as time granularity rule
4. **Custom** day **18.06.2021** (because this is a date you want to keep because of reasons)

When retention policy rules are evaluated, and we combine these rules we get this result. On the right you see the snapshots which are in database on 27.07.2021.

Day of load	Day of week	a)	b)	c)	d)	Result (Snapshot in DB)
26.07.2021	Monday	X				X
25.07.2021	Sunday	X	X			X
24.07.2021	Saturday	X				X
23.07.2021	Friday	X				X
22.07.2021	Thursday	X				X
21.07.2021	Wednesday	X				X
20.07.2021	Tuesday	X				X
19.07.2021	Monday	X				X
18.07.2021	Sunday	X	X			X
17.07.2021	Saturday	X				X
16.07.2021	Friday	X				X
15.07.2021	Thursday	X				X
14.07.2021	Wednesday	X				X
13.07.2021	Tuesday	X				X
12.07.2021	Monday					
11.07.2021	Sunday		X			X
10.07.2021	Saturday					
09.07.2021	Friday					
08.07.2021	Thursday					
07.07.2021	Wednesday					
06.07.2021	Tuesday					
05.07.2021	Monday					
04.07.2021	Sunday		X			X
03.07.2021	Saturday					
02.07.2021	Friday					
01.07.2021	Thursday					
30.06.2021	Wednesday			X		X
29.06.2021	Tuesday					
28.06.2021	Monday					
27.06.2021	Sunday		X			X
26.06.2021	Saturday					
25.06.2021	Friday					
24.06.2021	Thursday					
23.06.2021	Wednesday					
22.06.2021	Tuesday					
21.06.2021	Monday					
20.06.2021	Sunday		X			X
19.06.2021	Saturday					
18.06.2021	Friday				X	X
17.06.2021	Thursday					
16.06.2021	Wednesday					

Day of load	Day of week	a)	b)	c)	d)	Result (Snapshot in DB)
15.06.2021	Tuesday					
14.06.2021	Monday					
13.06.2021	Sunday		X			X
12.06.2021	Saturday					
11.06.2021	Friday					
10.06.2021	Thursday					
09.06.2021	Wednesday					
08.06.2021	Tuesday					
07.06.2021	Monday					

6.6. How To

6.6.1. Install database objects

When you use USU Analytics for the first time you need to create some database objects. Follow these steps to do this:

1. Open Data Load Organizer application.
2. Click **Play** in **Install workflow (System workflow)**.
3. The system now creates needed database objects and inserts date and time dimension record.
4. Open workflow protocol by clicking on chart icon. You can now see all actions executed by this workflow.

For details about **Install workflow (System workflow)** see Install workflow (page 128).

6.6.2. Create a new data load workflow

To create a new data load workflow, follow these steps:

1. Open Data Load Organizer application.
2. Click on **Create workflow** button and choose **Data Load** or **CDC Load**, based on you use case.
3. A new form opens.
4. Fill in the field **Name**. It is a user given name and you can change it later.

Be aware, that imports identify workflows by name.

5. If you want to execute a script before data load, please provide the full qualified path and file name in the field **External action before data load**.
6. In field **Refresh Mondrian schema caches for selection (no selection = all are refreshed)** you can specify which Mondrian schema should be refreshed after data load was done.
7. Check the field **Staging mode** if you want to use this data load mode. Otherwise, the standard mode is used. See administration manual for details about standard and staging mode.

8. The field **Active** is checked by default. Only active workflows are started automatically using the trigger information. Uncheck it when you don't want the workflow started, when the schedule is triggered, e.g., when the workflow is in development state.
9. Change to tab **Retention policy. Here** you can add your rules. For details see Retention policy rules (page 137).
10. Change to tab **Tables**. Here you choose the fact tables you want to load in this workflow. You only select the fact tables. The system computes the relevant extractions and dimension tables.

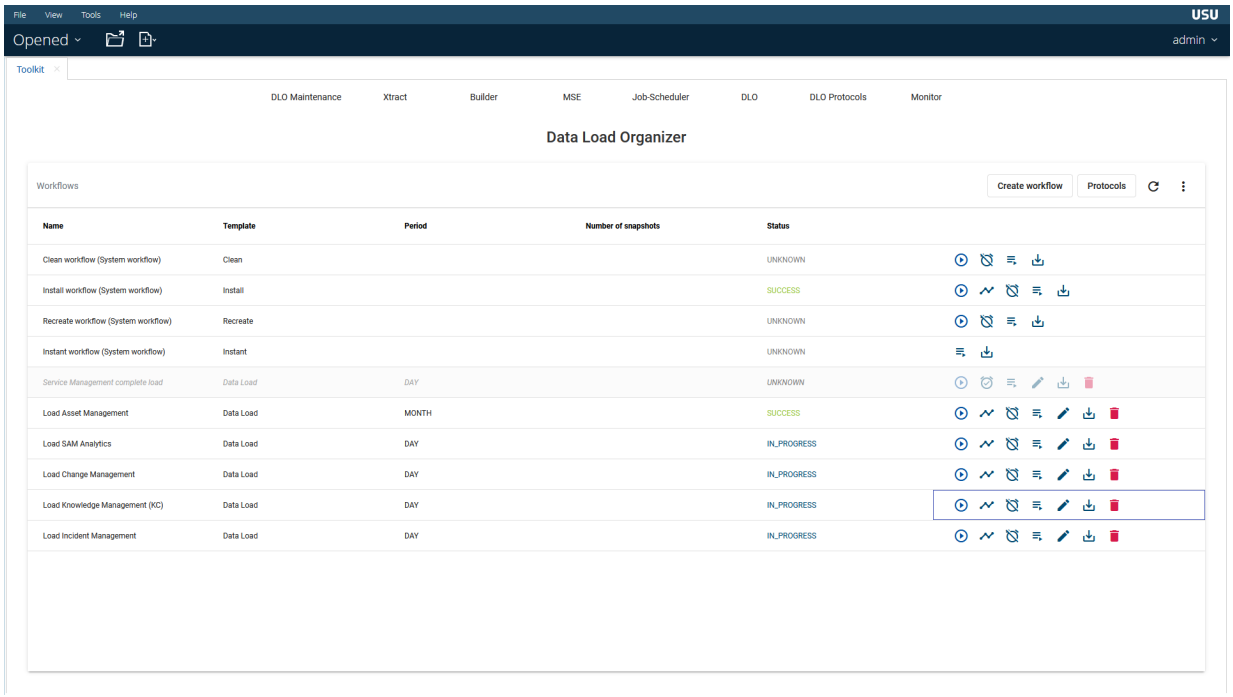
    You can add tables using the buttons

    a. Button **Add all** tables is adding all tables of the selected Data Source

    b. Button **Add tables** opens a new form where you can select from the tables of the selected data source. You can add one or more tables by holding the Ctrl key on your keyboard.

    c. Button **Clear tables** deletes all tables from the workflow definition.
11. Change to tab **Trigger**. Here you define the start time and time interval of your workflow execution. Use Minute, Hourly, Daily, Weekly, Monthly or Yearly section to define the most common time intervals. To enter a custom time interval, choose Advanced and consult Quartz Cron Attributes (page 162) to learn the Quartz Cron syntax in administration manual.

6.6.3. Maintain DLO workflows in DLO overview



You can maintain existing Scheduler jobs in the Scheduler overview screen. In the action's column, you find up to seven buttons. They give you the possibility to:

- Run immediately
- Display latest run
- Activate /deactivate Scheduler jobs
- Access protocols of a workflow
- Edit a workflow

- Export a workflow
- Delete a workflow

Some buttons are not available for some types of workflows. Also, some actions are not available based on the status of a workflow.

6.6.3.1. Run immediately

To run a workflow immediately click on the play button in actions column. The workflow will be executed directly after you pressed the button.

6.6.3.2. Display latest run

To access the protocol of latest run of a workflow click on the chart button in actions column. A new form opens, where you can see the protocol of the latest run.

6.6.3.3. Activate/deactivate a workflow

To activate or deactivate a workflow click on the clock button in actions column. Deactivated Scheduler jobs are greyed out and the trigger is ignored.

6.6.3.4. Access the protocols of a workflow

To access all protocols of a workflow click on the lines button in actions column. A new form opens with a list of all protocols for this workflow. Click on the chart button in the row of the protocol you want to display, and a new form opens, where you can see the details of the protocol you just have chosen.

6.6.3.5. Edit a workflow

When you click the fourth button in actions column the edit window opens. It is much the same interface as you use when you create a new workflow. For details, please see there.

6.6.3.6. Export a workflow

When you click the download button in actions column the download process starts. The system asks you, where to save the export file. After the selection of the destination folder the export starts. Some browsers have a default download directory. In this case you are not asked for a destination folder and the export starts immediately.

6.6.3.7. Delete a workflow

When you click the red waste basket button in actions column the deletion process starts. A security question is asked. If you do not deny, the workflow is deleted. You cannot undo this action.

6.6.4. Export all DLO workflows

To export all DLO workflows, follow these steps:

1. Open Data Load Organizer application.
2. Click **Export all workflows** in **more** menu, a new dialog opens.
3. The system asks you, where to save the export file. After the selection of the destination folder the export starts. Some browsers have a default download directory. In this case you are not asked for a destination folder and the export starts immediately.
4. All workflows are exported in one zip archive, except system in instant workflows.

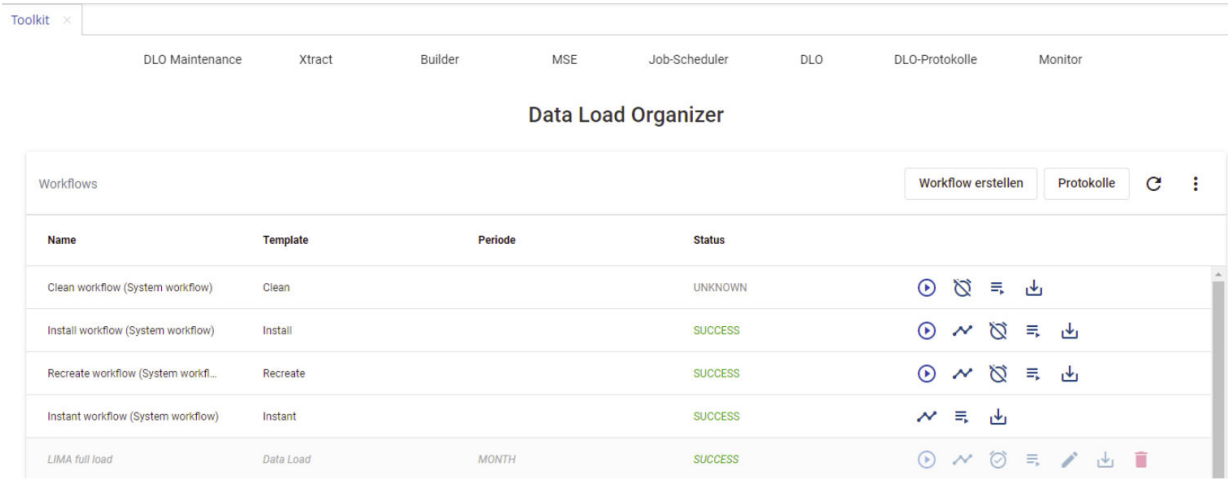
6.6.5. Import DLO workflows

To import DLO workflows, follow these steps:

1. Open Data Load Organizer application.
2. Click **Import** in **more** menu, a new dialog opens.
3. Choose a file to import. You can either choose on specific workflow definition file or you can use a zip archive to import multiple workflows a once.
4. Select the desired mode to be used in case of merge conflict between an existing and an imported workflow. If you check **Overwrite workflow**, existing workflows are overwritten.
5. Click **Import** button and start the import of the chosen import file.

6.6.6. Import USU License Management interfaces and data

To import data from USU License Management, execute the workflow **LIMA full load**. Execute the job to build the USU License Management interfaces and import the data from USU License Management.



6.6.7. Create, edit, or delete retention policy rules

To create, edit, or delete retention policy rules, follow these steps:

1. Open Data Load Organizer application.
2. Click edit button for the workflow you want maintain retention policy rules.
3. Here you can:
   a. **Add** a new rule by clicking button **Add rule**.
   b. **Edit a rule** by clicking the **pencil icon**.
   c. **Delete** a rule by clicking on the **red waste basket icon**.

For more details about retention policy rules see Retention policy rules (page 137).

Be sure your data load Trigger is matching the retention policy rule you define. Not matching definitions can lead to unwanted results and in worst case to data loss.

6.6.8. Recreate the system

When you want to recreate your system follow these steps:

1. Open Data Load Organizer application.
6. Click **Play** in **Recreate workflow (System workflow)**.
7. The system now recreates needed database objects and inserts date and time dimension record.
8. Open workflow protocol by clicking on chart icon. You can now see all actions executed by this workflow.

For details about **Recreate workflow (System workflow)** see Recreate workflow (page 128).

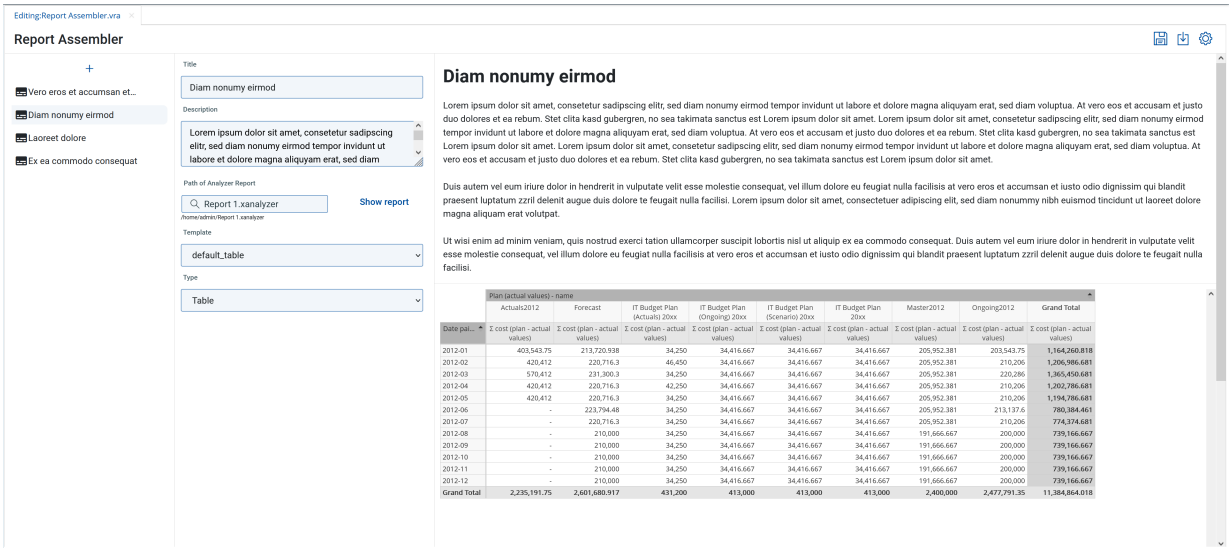
6.6.9. Clean the system

To clean the system, follow these steps:

1. Open Data Load Organizer application.
9. Click **Play** in **Clean workflow (System workflow)**.
10. The system now deletes data in DM and drops tables in EA.
11. Open workflow protocol by clicking on chart icon. You can now see all actions executed by this workflow.

For details about **Clean workflow (System workflow)** see Data Load (page 129).

7. Report Assembler



7.1. Overview

The Report Assembler is an application to combine multiple Analyzer Report to one set and create a PDF file out of it.

The main functionalities are:

- Assemble different Analyzer Reports to a set and create a PDF.
- Print chart and/or table information.
- Customize your report templates to your corporate identity.
- Enrich Analyzer Report information with texts.
- Schedule the creation and send the file via Job-Scheduler module.
- Use parameters to filter in Job-Scheduler and use parameters in texts.

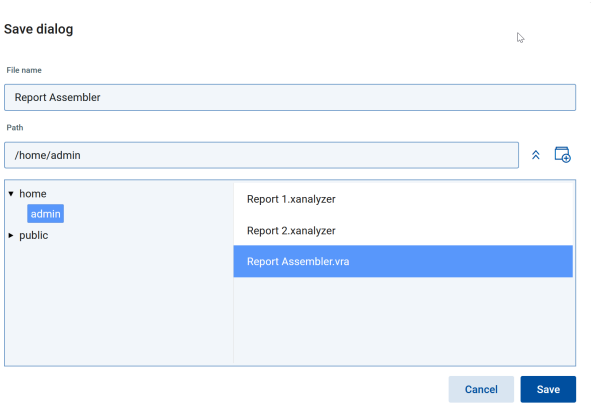
This feature is an add-on with additional license fee. Please get in contact with your responsible sales person to get more information.

7.2. The Report Assembler Application

The Report Assembler has three main areas:

- On the left side you see the panel with an overview of your chosen Analyzer Reports
- one in the middle to configure information for the chosen sub report
- and on the right, you see the representation of the selected sub report.

In the following table we describe the main functionalities you can use in Report Assembler.

Object	Description
Save	When you click **Save**, a new dialog opens.  Select the path to store the Report Assembler Definition in and fill in the file name. Click the **Save** button. The file will be saved.
Settings	When you click the **Settings** button, the **Settings** dialog opens.  **Template:** In the template section you choose the Report Assembler template you want to use. We ship our product with two examples: Landscape and Portrait. You can add new Report Assembler templates or customize the existing (see chapter How To (page 103)). **Header:** Here you define the report header information. You can use build-in parameters. For details see chapter Build-in Parameter (page 146). **Footer:** Here you define the report footer information. You can use build-in parameters. For details see chapter Build-in Parameter (page 146).
Generate report	Click the button **Generate report** and the report will be created instantly. For each Analyzer Report we defined a fixed time out of 5 minutes.

7.3. Parameter

7.3.1. Build-in Parameter

In Report Assembler definitions you can use build-in parameter, which are replaced on runtime of the PDF generation. You can use them in various areas of the application.

Implemented build-in Parameter are:

Parameter	Description
${CURRENT_TIMESTAMP}	Date and time information. Example: 2018-08-08 18:10:02
${VMA_CURRENT_SERVER_URL}	Server URL Example: http://localhost:8080/console
${CURRENT_DATE}	Date information Example: 2018-08-08
${VMA_CURRENT_USER}	User, which is executing the generation. Example: admin

7.3.2. Analyzer Report Parameter

In Report Assembler definitions you can use parameter you defined an Analyzer Report, which are replaced on runtime of the scheduled PDF generation. The parameter you can use in the Report Assembler context are shown in the blue area in settings and in the tile edit dialog.

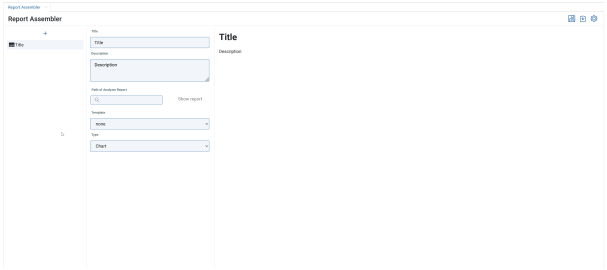
A detailed description on how to add and maintain parameters in Analyzer Reports is available in the user manual.

7.4. How To

7.4.1. Create a New Report Assembler Definition

To create a new Report Assembler Definition, please follow these steps:

1. Open the User Console **Home** screen.
2. Click **Create New** and then click Report Assembler. The Report Assembler application opens, and an empty tile is displayed.



3. Fill in the fields in the middle panel

   Main settings for the selected tile are:

   Title: Here you define the title of the selected tile. You can use build-in parameters.

   Description: Here you define the description of the selected tile. You can use build-in parameters.

   Path of Analyzer Report: In this field you can select the Analyzer Report you want to use in this tile. Click the field and a search field appears. You can search and select the Analyzer Report you want to use here. *Show report* opens the Analyzer Report in a new tab.

   Type: An Analyzer Report can have two representations, as chart or as table. Select what you want to see.

   Template : Chose the table template you want to use. This influences the way a table will be rendered.

> Not all chart types are supported. We support only those chart types, which are possible to export in the standard export possibilities in Analyzer Report.

Click **Save** in the **Edit tile** dialog and the tile will be updated.

4. Click **Save** to save the Report Assembler definition. Choose the **Path** and add the **File name**.
   Your definition is now saved in the server repository.

7.4.2. Edit an Existing Report Assembler Definition

To edit an existing Report Assembler Definition, please follow these steps:

1. Open the User Console **Home** screen.
2. Click Browse Files and select the Report Assembler definition you want to edit.
   Choose **Edit** from **More** menu.
3. The Report Assembler definition opens in edit mode.
4. You now can edit the settings of the definition.
5. To add a new tile to the definition, click on the **+** (plus) icon you see in the Report Assembler menu.
   An empty tile will be added below the active tile, and you can edit this tile like described in the former chapter.
6. To clone a tile, activate the tile you want to clone and click the clone icon . The copy is added below the active tile.
7. To move a tile, click the tile an keep the left mouse button pressed. Drag the tile where you want it and let it drop.
8. To remove a tile, activate the tile you want to delete and click the trash icon.

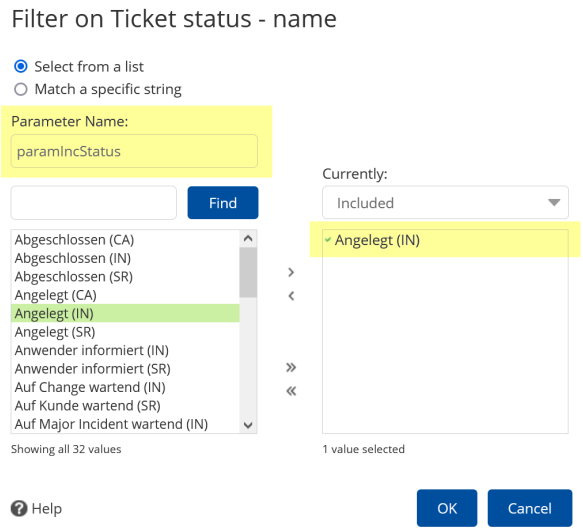
> The tile is removed, and it is not possible to revert this change.

9. Save your changes by clicking on the **Save** button.

7.4.3. Use Parameters

To use parameters in Report Assembler, please follow these steps:

1. Create a filter in the Analyzer Report you want to have parametrized in Report Assembler.



In this example we created a filter for the dimension **Incident status – name**. The value we selected in filter is **Created (IN)**. The Name of the parameter we set to **paramIncStatus**.

2. Now we open or create a Report Assembler definition and add the Analyzer Report with the filter we created before to the definition.
3. Select the tile with the new Analyzer Report. Open the edit dialog and add the parameter to the description, e.g. like this:

   This is the parameter we defined in the Analyzer Report: ${paramIncStatus}

   Click Save.

> In browser view, the parameters are not replaced and the chart shows the values using the value we selected in the filter in Analyzer Report.

4. Save the Report Assembler definition.
5. Open the Scheduler application.
6. Create a new Scheduler job of type Report.

   In Actions select the Report Assembler definition we created in the steps before.
7. Add the parameter to the Action.

> To define the parameter value you must use the business key / ID of the value you want to filter by. You can get this value if you open the Analyzer Report, open the filter and move the mouse over the filtered value. In this case the value is **IN_CLD**.

8. Save the job and test it.
9. After the report was rendered you can see that the parameter was replaced in the text and the data was filtered according the parameter value.

7.4.4. Schedule an Existing Report Assembler Definition

To schedule an existing Report Assembler Definition, please follow the steps in chapter Create a New REPORT Scheduler Job (page 156) and Use Parameters (page 148).

7.4.5. Report Assembler Template

A Report Assembler template is a Report Designer report especially designed for the usage in the Report Assembler application. We ship two examples: Landscape and Portrait.

You can create your own templates or change the standard templates.

Please be aware that adding or changing a Report Assembler template is as administrational task.

You need to access the server on file level, and you must be familiar with the Report Designer tool. To create or change a template you need the Report Designer tool we ship with version 4.7 or higher.

For support please contact the support team.

7.5. Report Assembler Templates

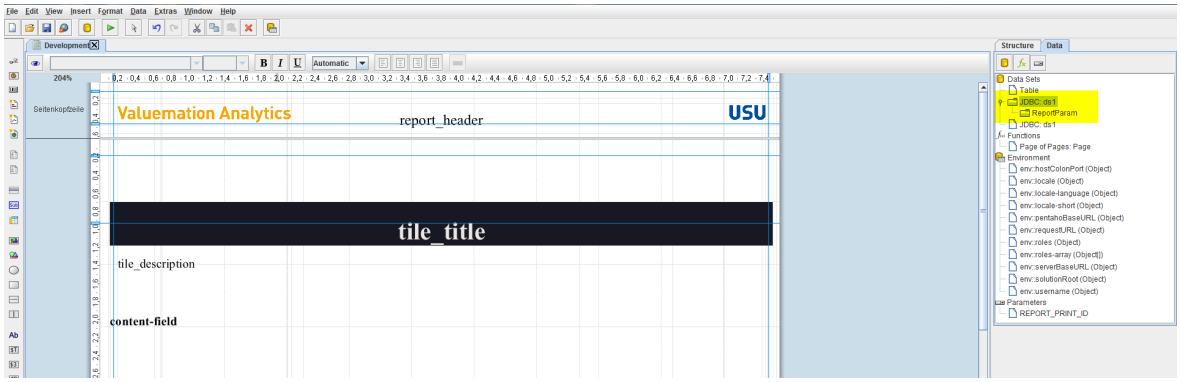
The Report Assembler is an application to combine multiple Analyzer Report to one set and create a PDF file out of it. The following chapters are mainly about how to create new or change existing Report Assembler templates.

7.5.1. Prerequisites

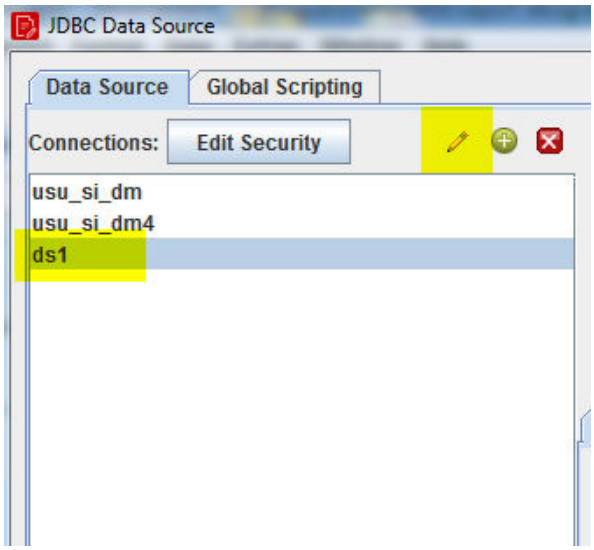
- You must be familiar with PRD development and customizing.
- To customize report templates, you need Report Designer version 8 (own USU build). We ship this version since version 4.7 in our installation package.
- The Report Assembler templates (prpt files) are not backward compatible. Opening with older Report Designer version than Report Designer version 8 (own USU build) will fail.
- Add JDBC driver for the DBMS System you are using to your Report Designer installation
- Database must be reachable from your Report Designer installation.
- You find the Report Assembler templates (prpt files) in Browse Files dialog here: /public/Administration/Report Assembler/PRPT template
- Download these files. Make them available from your Report Designer installation.

7.5.2. Prepare the template for customizing

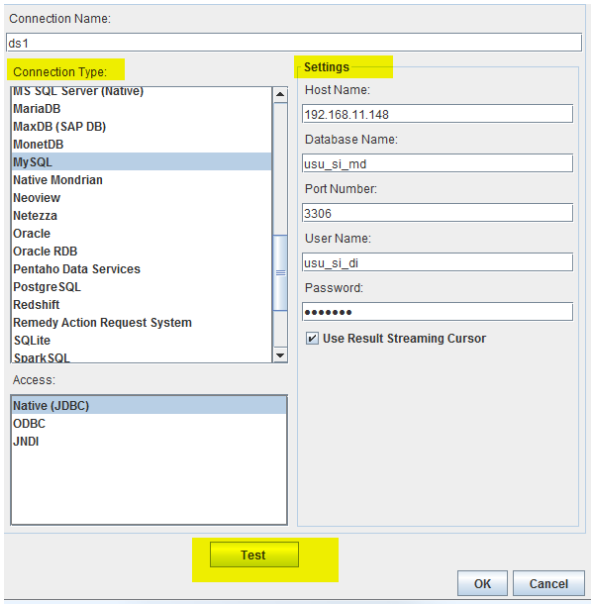
1. Open Report Assembler template (prpt file) in Report Designer version 8 (own USU build).
2. Open Data tab and double-click ds1 DB connection.



3. Edit ds1 database connection.



4. Set the correct connection type (Oracle, MySQL, MS SQL) and add the correct settings.



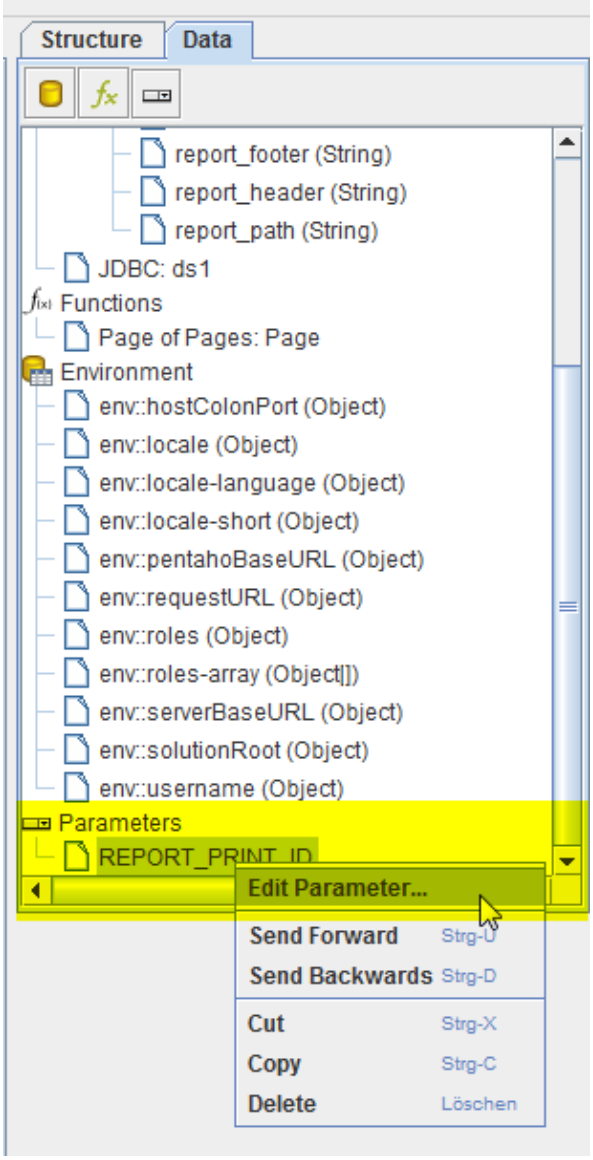
Test the connection.

Click OK.

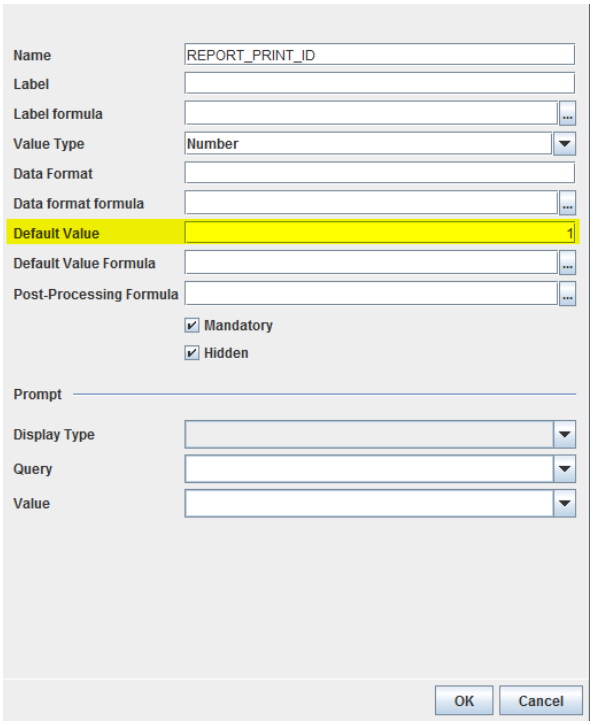
5. Open the sub report.

Open the DB connection and do the same as described above.

6. Open a DB client and execute this select on MD database.
   `SELECT * FROM usu_si_md.sir_report`
   In this table all Report Assembler executions are stored.
   Get an ID of a Report Assembler execution to test with.
7. Go back to main report and edit parameter REPORT_PRINT_ID.



8. Replace the value in the field Default value with the ID you just selected from database.



9. Save the report.
   It is best practice to use a new file name.
   The template will be listed with this name in Report Assembler application.
10. Now you can change the Report Assembler to your needs and test it offline.

7.5.3. Publish New or Changed Report Assembler Templates

- To publish new or changed table templates upload the new or changed ftl file in Browse Files to /public/Administration/Report Assembler/FTL template
- If you open the tile you will see your new template or your changed template will be used.
- You don't have to change the DB connection before you upload the prpt file. The DB connection will be replaced on the fly during runtime.
- To publish new or changed templates no service restart is necessary.
- To perform this action you need to be administrator.

7.5.4. Table Templates

Report Assembler can print data as chart and/or table. Tables could vary from some small short tables to very wide and long ones. So it is needed to change the appearance of a table.

To do this we use a templating engine called FreeMaker. In the FreeMarker template, you can use all the features offered by this technology. The link to the FreeMarker documentation https://freemarker.apache.org/docs/index.html . The limitations are on the HTML and CSS, which are supported by Report Designer. For a correct result it is therefore recommended to use HTML 3.2 and CSS1.

You can also create new templates and use them in your reports. The templates are stored in Browse Files: /public/Administration/Report Assembler/FTL template.

This template is connected to only one object and that is Table. The Table object contains the following variables that can be used in the FreeMarker template. All variables are case-sensitive.

```
Table
    |_rows (List) -> contains only a collection of cells
        |_cells (List) -> cell contains the name of the cell type from the cell type list
            |_cellType (CellType)
            |   |_COLUMN_HEADER_NAME    -> upper part of the table header indicating column
name labels
            |   |_COLUMN_HEADER_VALUE   -> header column name value
            |   |_COLUMN_HEADER_SUM     -> column label in the table header section,
indicating the sum of all values.
            |   |_COLUMN_SUM            -> column with the sum of values in the rows.
            |   |_ROW_HEADER_NAME       -> upper part of the table header indicating row name
labels
            |   |_ROW_HEADER_VALUE      -> row name value
            |   |_ROW_SUM_BOTTOM        -> last row for with sume values for each column
            |   |_MEASURE_HEADER        -> name for mearsure. Always the last top row of the
table header section
            |   |_DATA                  -> cell with table data
            |   |_EMPTY                 -> marker for empty value without any content
            |_value -> The concrete value of the cell. The cell value type is unknown. And it
is always displayed as string.
```

We deliver a default ftl definition with the product. Use this as an example for your customizing. We recommend you to not change the default file, but upload your changes in a file with a different name.

7.5.5. Publish New or Changed Table Templates

- To publish new or changed table templates upload the new or changed ftl file in Browse Files to /public/Administration/Report Assembler/FTL template
- If you open the tile you will see your new template or your changed template will be used.
- To publish new or changed templates no service restart is necessary.
- To perform this action you need to be administrator.

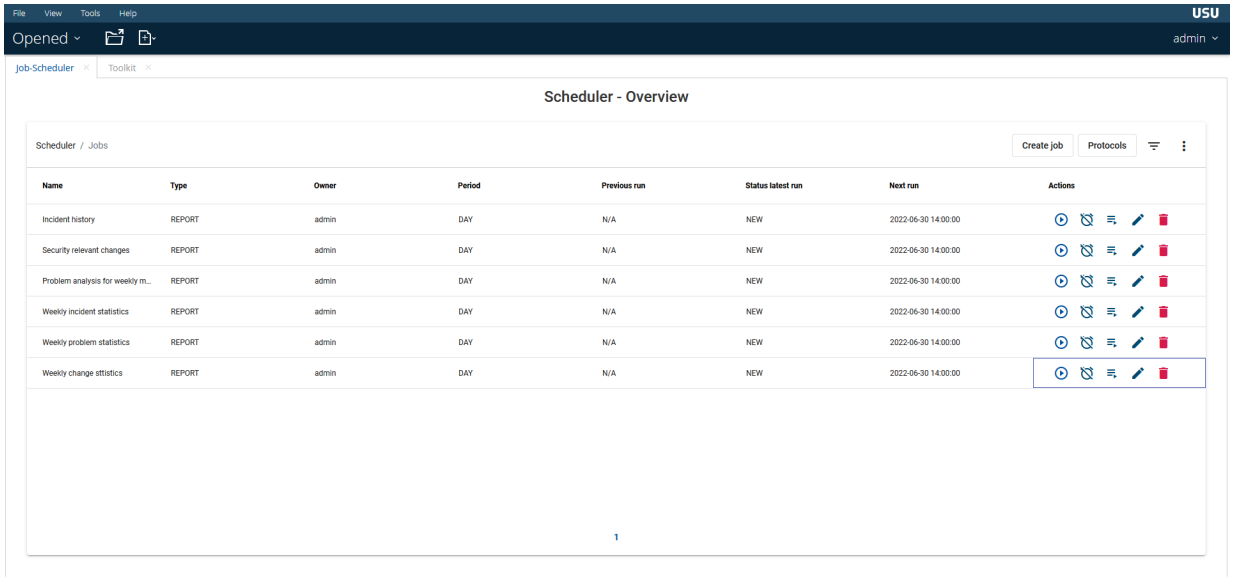
7.5.6. Good to Know

- In principle, you can use all the options provided by the Report Designer.
- The relevant tables you find in MD database. The tables start with sir_.
- The report table information is stored as HTML in the sir_tile.table_html database field. To ensure that the contents of the report are displayed correctly, the Rich-Text-Type of the file must be set to text/html.
- You should run a Report Assembler definition minimum one time to have a sample record in database.

- RA PNGs are not deleted
- Talky error messages
- REST API will have documentation

- Pictures are rendered during runtime and are saved temporary on the server. If you don’t run the system in DEV mode, the charts will not be viewed in Report Designer.
- Column is_chart indicates, if analyzer report is printed as chart or tables. Value is true or false and could be directly used in PRD functions.
- We defined a hardcoded time out of 5 minutes per sheet. E.g. if you have two Analyzer Reports in you RA definition the timeout will be 10 minutes.

8. Job-Scheduler



8.1. Overview

The Job-Scheduler is an application to schedule and render reports and publish them and execute maintenance jobs such as Refresh Mondrian automatically. The module can be accessed from **Home** screen or from **Toolkit**.

The main functionalities are:

- Organize your data load schedules within User Console – so there is no need to use Windows Task Scheduler or CRON any more.
- You can create, edit, activate, deactivate, and delete your scheduler jobs.
- You can define dependencies between scheduler jobs e.g., run job B only, if job A finished successfully.
- All runs are logged.
- Send a email after the job has finished. You can decide if you want to send out this message after every run or only in case of an error. Log information could be attached.
- Scheduler job definitions can be exported and imported.
- You can use filters to search for specific Scheduler jobs.
- You can use filters to search in Scheduler protocols.

You can choose from three types of schedules:

- **ETL**: With ETL you create scheduled jobs related to data integration and system maintenance.

> This functionality is deprecated and replaced by the new DLO module (page 122).

- **REPORT**: With REPORT you create scheduled jobs related to report creation and publishing.

- **SYSTEM**: With SYSTEM you create scheduled jobs which can execute scripts on OS level (like .bat or .sh) or predefined actions.

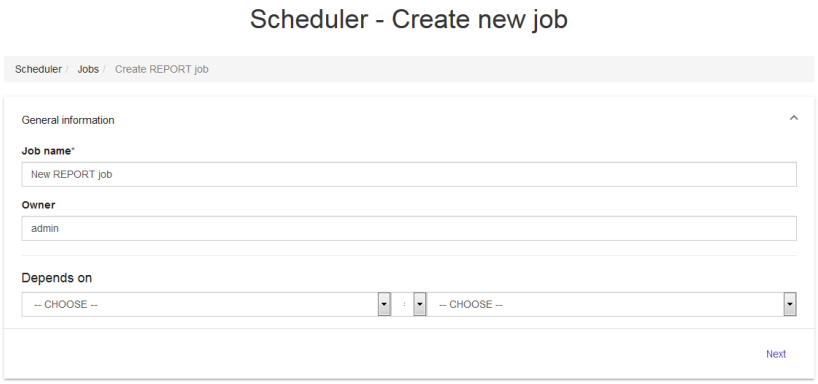
8.2. How To

8.2.1. Create a New REPORT Scheduler Job

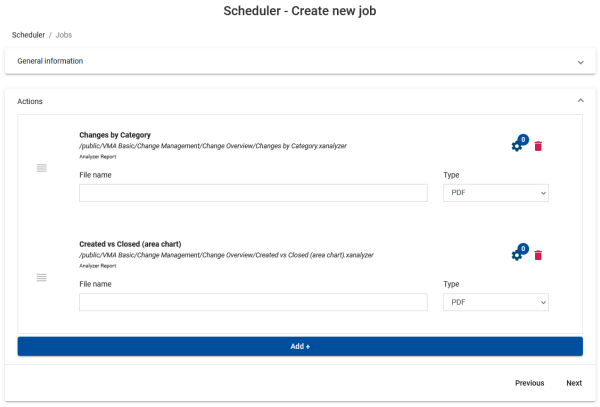
To create a new Scheduler job of type REPORT, please follow these steps:

1. Open Scheduler application.
2. Click the **Create job** button and choose REPORT. The **Scheduler – Create new job** dialog opens.
3. Fill in the **Job name**. The job name is unique per user.

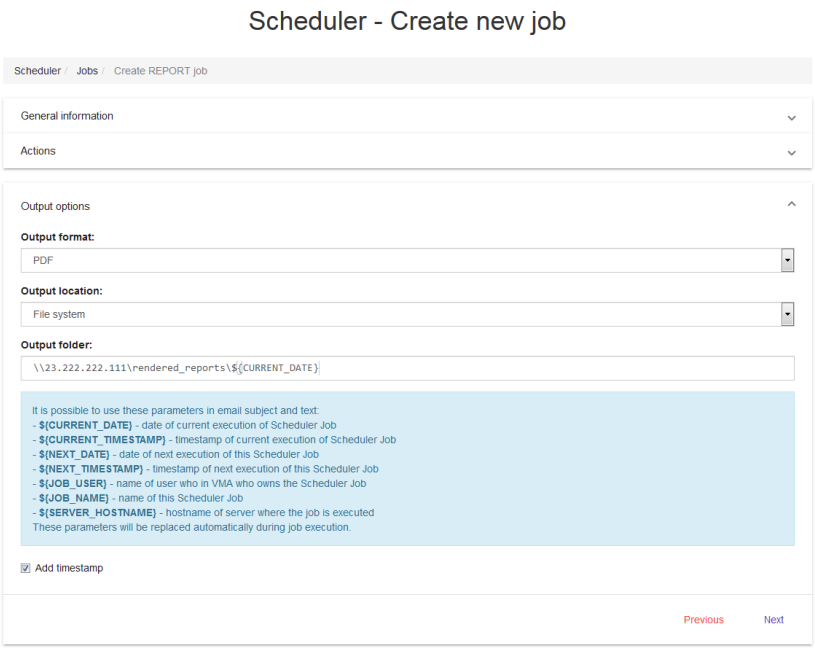
   If your new job depends on the result of another job, you define this dependency in **Depends on**.



   Click **Next** to get to the **Actions** section.
4. Add actions you want to run in this Scheduler job. Click the Add + button, select one or more of the offered reports. You can sort the selected reports by dragging them around or delete a report by clicking on the waste basket icon. Define the name and output format per report.



   Click **Next** to get to the **Output options** section.
5. In the Output options section, you define the output format of the reports, output location and the output folder.



Choose an **Output location** and an **Output folder**, if you want to save a copy of the rendered reports. File system will save the copies to some storage device. Repository will save the copies in the database. When specified folder doesn't exist, it will be created. It is possible to use in path also variables such as ${CURRENT_DATE}, ${JOB_NAME} etc. For example, path can be specified as /public/reports/${CURRENT_DATE}/. Then there will be automatically created new folders every day in format /public/reports/2018-12-20/.

When checking **Add timestamp**, a timestamp information will be added to the file name of the rendered report files.

Click **Next** to get to the **Trigger** section.

6. In the Trigger section you define the start time and time interval of you schedule.

   Use Minute, Hourly, Daily, Weekly, Monthly or Yearly section to define the most common time intervals.

   To enter a custom time interval, choose *Advanced* and consult chapter *Quartz Cron Attributes* to learn the Quartz Cron syntax.

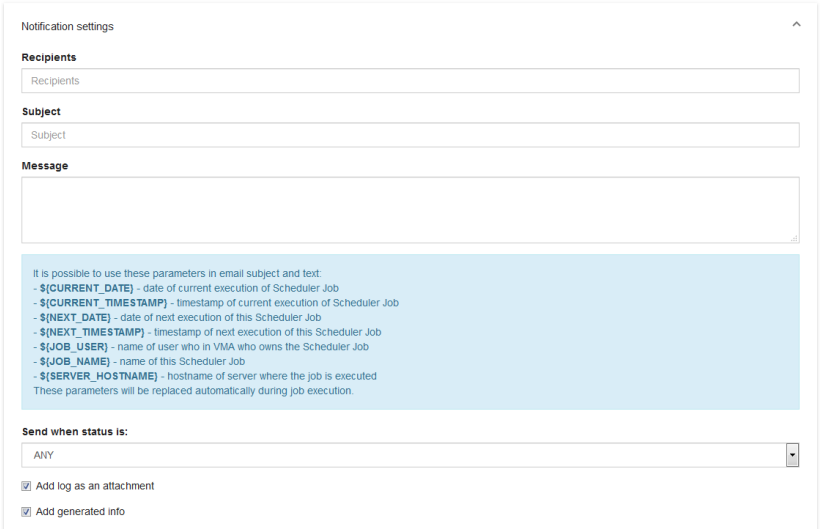
   Click **Next** to get to the **Notification settings** section.

7. Information in **Notification settings** section is optional. When a Scheduler job is finished you can define if an automatic message is sent.

   Fill in a comma separated list of email addresses as **recipients**, a **subject** and a **message** text.

   With **Send when status is** you can define when a message is sent, e.g. only in case of a failure.

   With **Add log as an attachment** and **Add generated info** you can add log information to you email.



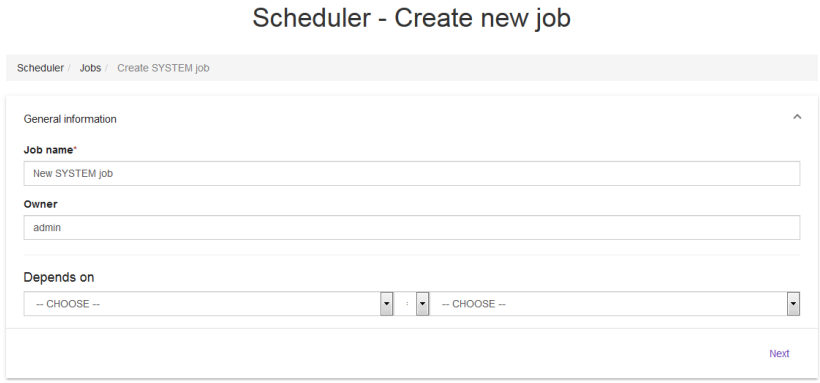
8. With the **Previous** button, you can always go back to other sections.
9. Save your new Scheduler job.

8.2.2. Create a New SYSTEM Scheduler Job

To create a new Scheduler job of type SYSTEM, please follow these steps:

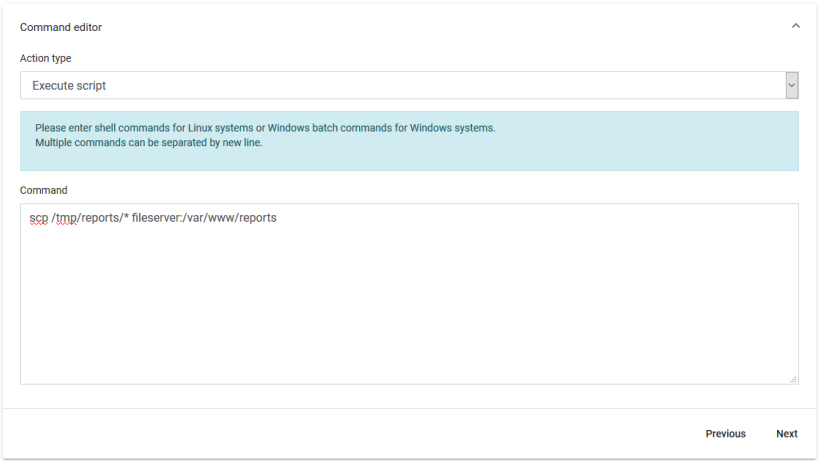
1. Open Scheduler application.
2. Click the **Create job** button and choose SYSTEM. The **Scheduler – Create new job** dialog opens.
3. Fill in the **Job name**. The job name is unique per user.

   If your new job depends on the result of another job, you define this dependency in **Depends on**.



   Click **Next** to get to the **Command editor** section.
4. In *Command editor* section you can choose, whether a self-created script is executed or a prepared action.

   Execute script: Choose action type *Execute script* and add your OS level commands to the command field. The commands will be executed as .bat or .sh script on OS level. Please use the commands respective to the operating system the BA server is running on. The example below copies files from one file server to another. You can also use this functionality to upload files to Microsoft SharePoint or other systems.



Execute prepared action: Choose action type *Execute prepared action* and and select the action.

Click **Next** to get to the **Trigger** section.

5. In the Trigger section you define the start time and time interval of you schedule.

   Use Minute, Hourly, Daily, Weekly, Monthly or Yearly section to define the most common time intervals.

   To enter a custom time interval, choose *Advanced* and consult chapter *Quartz Cron Attributes* to learn the Quartz Cron syntax.

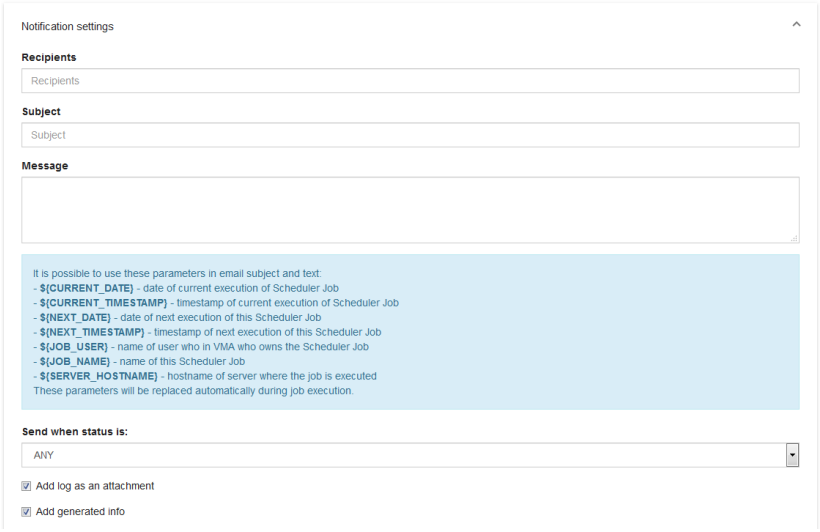
   Click **Next** to get to the **Notification settings** section.

6. Information in **Notification settings** section is optional. When a Scheduler job is finished you can define if an automatic message is sent.

   Fill in a comma separated list of email addresses as **recipients**, a **subject** and a **message** text.

   With **Send when status is** you can define when a message is sent, e.g. only in case of a failure.

   With **Add log as an attachment** and **Add generated info** you can add log information to you email.



7. With the **Previous** button, you can always go back to other sections.
8. Save your new Scheduler job.

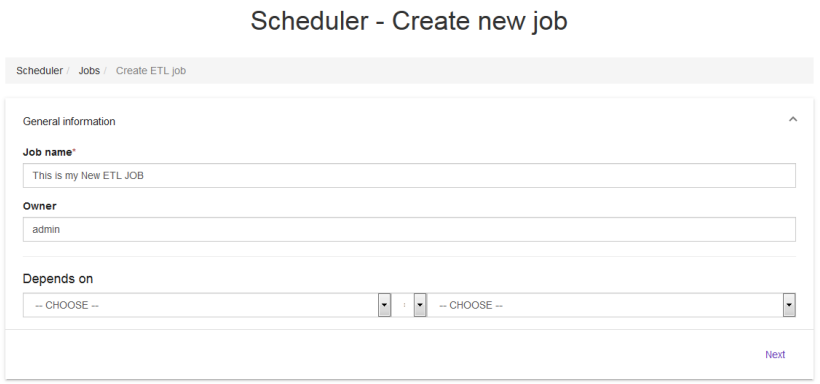
8.2.3. Create a New ETL Scheduler Job

This functionality is deprecated and replaced by the new DLO module (page 122).

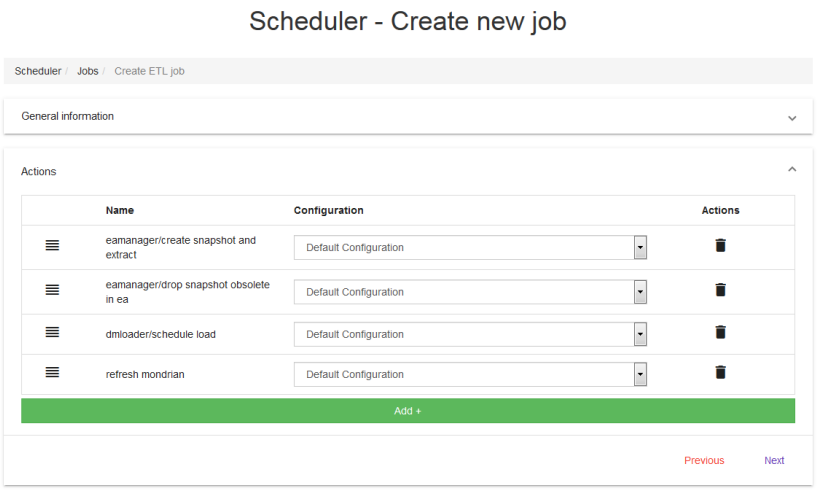
To create a new Scheduler job of type ETL, please follow these steps:

1. Open Scheduler application.
2. Click the **Create job** button and choose ETL. The **Scheduler – Create new job** dialog opens.
3. Fill in the **Job name**. The job name is unique per user.

   If your new job depends on the result of another job, you define this dependency in **Depends on**.



   Click **Next** to get to the **Actions** section.
4. Add actions you want to run in this Scheduler job. Click the **Add +** button, select one or more of the offered actions. You can sort the selected action by dragging them around or delete an action by clicking on the waste basket icon.



Click **Next** to get to the **Trigger** section.

5. In the Trigger section you define the start time and time interval of you schedule.

   Use Minute, Hourly, Daily, Weekly, Monthly or Yearly section to define the most common time intervals.

   To enter a custom time interval, choose *Advanced* and consult chapter *Quartz Cron Attributes* to learn the Quartz Cron syntax.

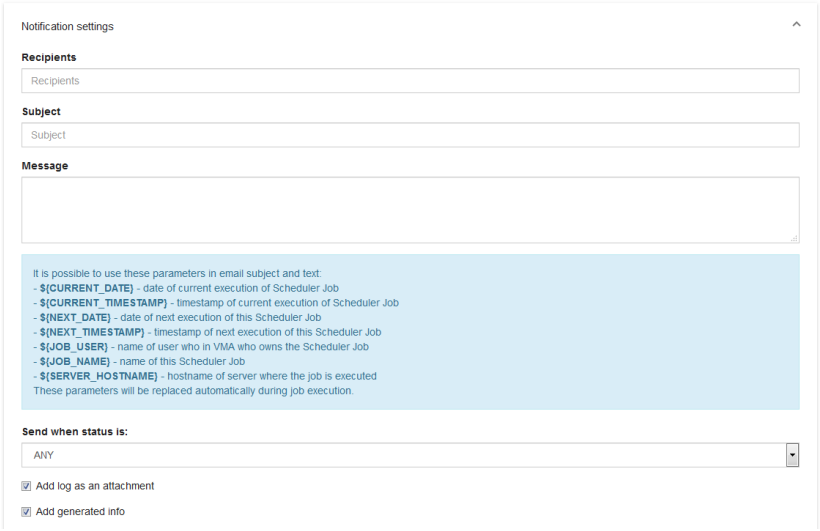
   Click **Next** to get to the **Notification settings** section.

6. Information in **Notification settings** section is optional. When a Scheduler job is finished you can define if an automatic message is sent.

   Fill in a comma separated list of email addresses as **recipients**, a **subject** and a **message** text.

   With **Send when status is** you can define when a message is sent, e.g. only in case of a failure.

   With **Add log as an attachment** and **Add generated info** you can add log information to you email.



7. With the **Previous** button, you can always go back to other sections.
8. Save your new Scheduler job.

8.2.4. Quartz Cron Attributes

The Quartz cron engine supports a seven-attribute time declaration with many possible values. The number format is the same for every expression, even if the values are different -- it must be listed as seconds, minutes, hours, day of month, month, day of week, then the year. A space separates each attribute.

These are the possible values for each attribute: 0 to 59 for seconds and minutes, 0 to 23 for hours, 1 to 31 for days, 1 to 12 for months, 1 to 7 for day of week, and a four-digit year. Alternatively, you can use three-letter values for the day of week (MON, TUE, WED, THU, FRI, SAT, SUN), and three-letter values for the month (JAN, FEB, MAR, APR, MAY, JUN, JUL, AUG, SEP, OCT, NOV, DEC).

An asterisk (*) indicates *all values*, so an asterisk in the minute field would mean that the report runs once every minute. You can specify a range of values with the - (dash) operator, and you can specify multiple individual values with a comma. If you need to excuse a value in the day of month and day of week field from a cron job, you can use the question mark (?) character to indicate that this value doesn't matter. If you need to split values, you can do so with the slash (/) character -- this operator literally means "every," so */15 would mean "Every 15." In the day of month field, you can use the # character to indicate a certain instance of a day of the month, for instance the second Friday of the month would be 6#2. Lastly, you can use a capital L in the day of month and day of week field to indicate "Last," as in the last day of the week. A capital C in either of these fields means "Calendar," and combined with a number means that the report should execute the interval indicated by the C number according to the loaded calendar. A capital W in the day of month attribute means "Weekday," which only encompasses Monday through Friday. Most of these values can be combined to create unusual cron schedules.

Attribute	Conditionals and Operators
Seconds	, - * /
Minutes	, - * /
Hours	, - * /
Day of month	, - * ? / L W C
Month	, - * /
Day of week	, - * ? / L C #
Year	, - * /

Here is how you would execute a report at 10:15 AM on every last Friday of every month during the years 2008, 2009, 2010, 2011, 2012, and 2013.

```
0 15 10 ? * 6L 2008-2013
```

You can find more details including examples here:

http://www.quartz-scheduler.org/documentation/quartz-2.3.0/tutorials/crontrigger.html

A nice page, which helps to write and test your Quartz Cron expressions, is this:

https://www.freeformatter.com/cron-expression-generator-quartz.html#

8.2.5. Specify Report Names

If you don’t specify specifically a name for the rendered report, the file name of the definition will be used.

To change the name per rendered report, please follow these steps:

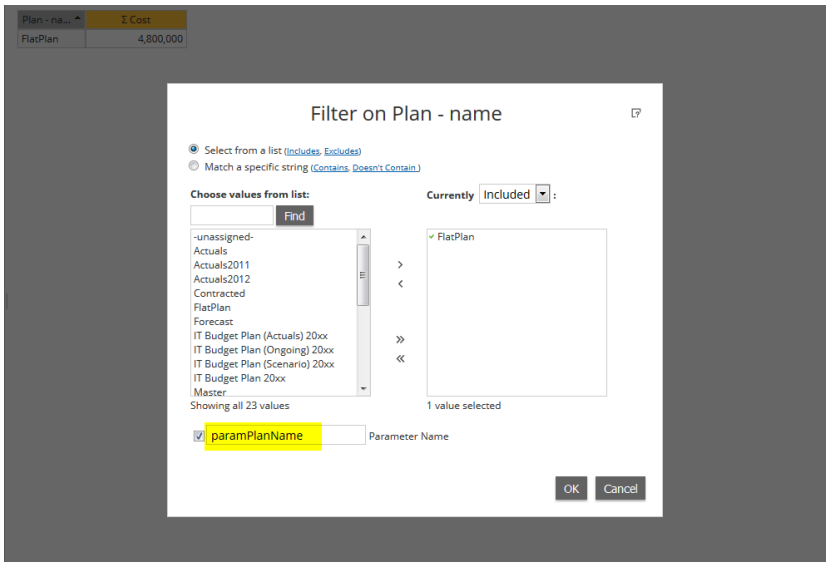
1. Open Scheduler application.
2. Open an existing schedule or create a new one.
3. Open Actions section.

   Fill in the **File name**. It is possible to use variables such as ${CURRENT_DATE}, ${JOB_NAME} etc. For example, file name can be specified as My_Report_Name_${CURRENT_DATE}/. This will result in a file with the name My_Report_Name_2018-12-20.
4. Do this for all reports you want to give names, which are different to the report file name.
5. Save the Scheduler definition.

8.2.6. Use Parameters with Analyzer Report

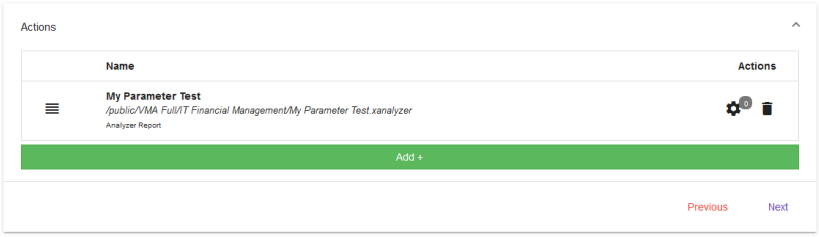
To use Analyzer report with parameters in Scheduler, please follow these steps:

1. Create a filter in the Analyzer Report you want to have parametrized in Scheduler.



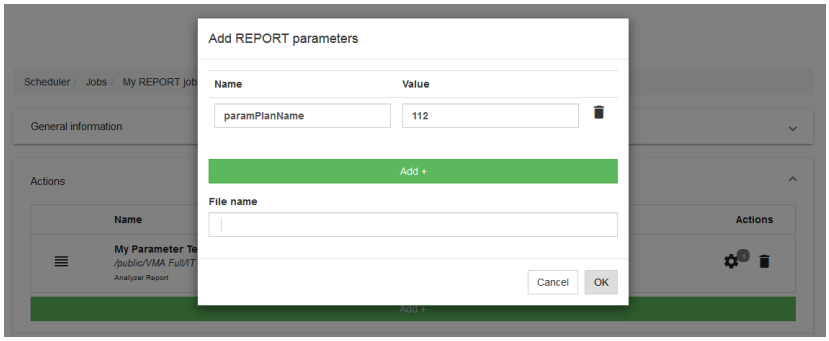
   In this example we created a filter for the dimension **Plan – name**. The value we selected in filter is **FlatPlan**. The Name of the parameter we set to **paramPlanName**.
2. Open the Scheduler application.
3. Create a new Scheduler job of type Report.

   In Actions select the Analyzer Report definition we created in the steps before.

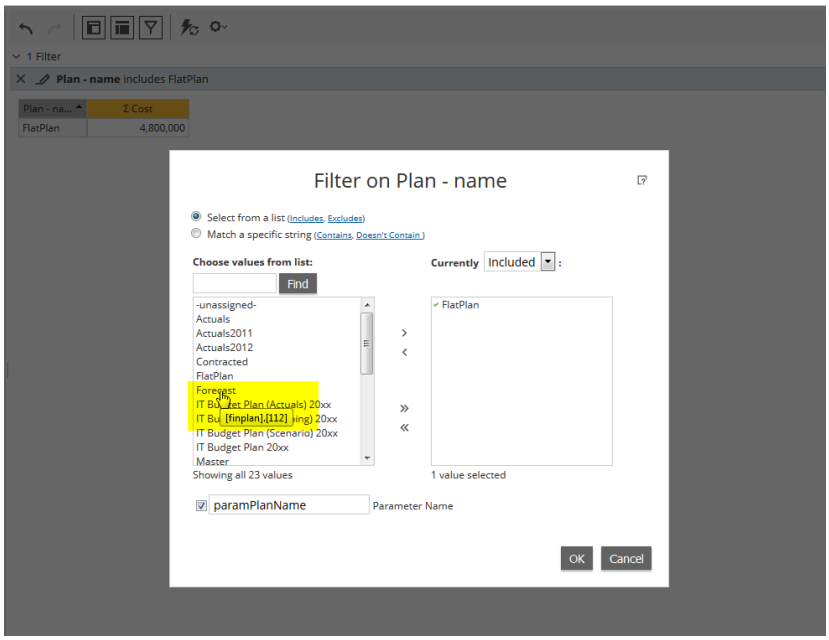


4. Click on the gear wheel icon in the Actions column. A new dialog opens.

   Add the parameter to the Action.



To define the parameter value you must use the business key / ID of the value you want to filter by. You can get this value if you open the Analyzer Report, open the filter and move the mouse over the filtered value. In this case the value is **112**.



5. Save the job and test it.
6. After the report was rendered you can see that the parameter was replaced, and the data was filtered according the parameter value.

8.2.7. Use Parameters with Report Designer Report

To use Report Designer report parameters in Scheduler, please follow these steps:

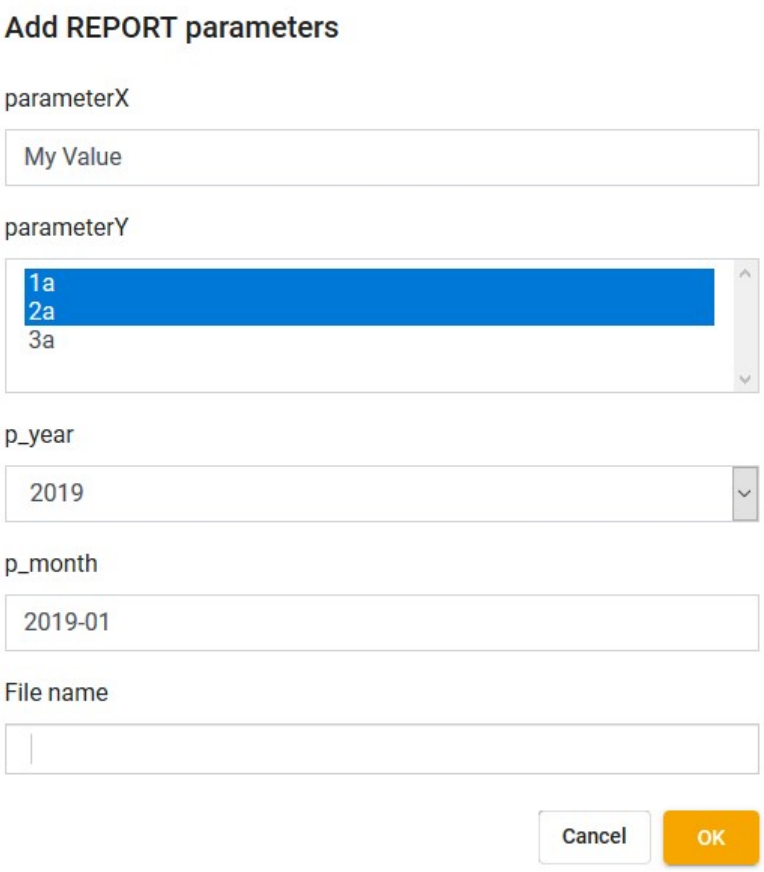
1. Create parameters in your Report Designer report.

   Upload the Report Designer report to your application server using Browse files.

   Make sure the Report Resigner report works in this environment and the prompts and parameter are functioning properly.
2. Open the Scheduler application.
3. Create a new Scheduler job of type Report.

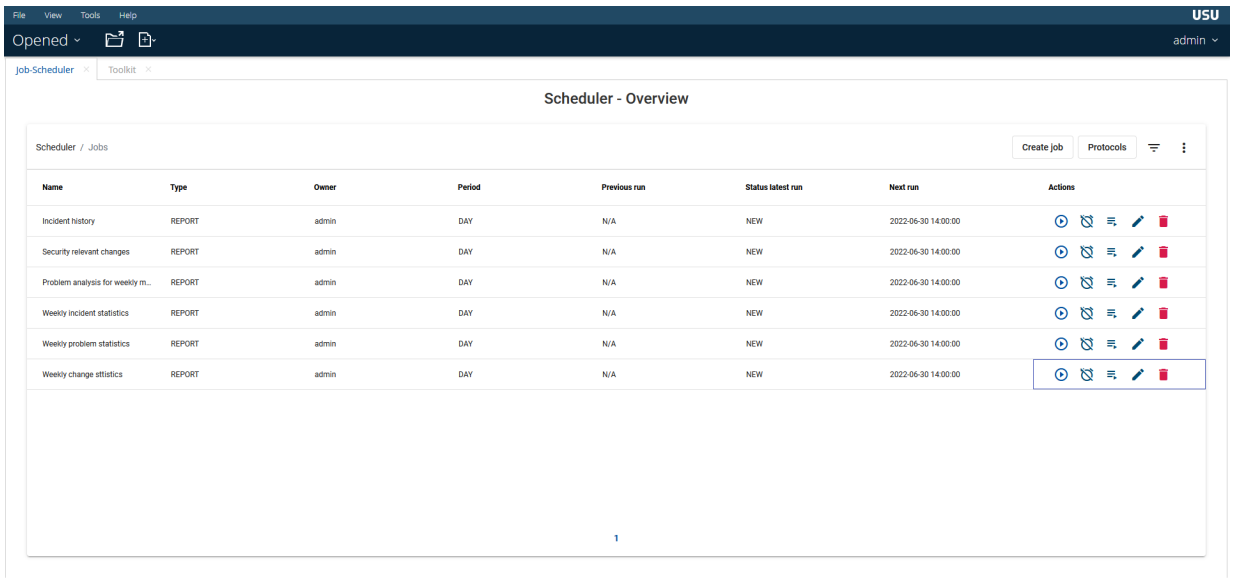
   In Actions select the Report Designer report we uploaded in the steps before.

4. Click on the gear wheel icon in the Actions column. A new dialog opens. It shows you all parameters defined in the Report Designer report. For some types of parameters, it is possible to select values from a list.



5. Save the job and test it.
6. After the report was rendered you can see that the parameter was replaced, and the data was filtered according the parameter value.

8.2.8. Maintain Scheduler Jobs in Scheduler Overview



You can maintain existing Scheduler jobs in the Scheduler overview screen. In the action's column, you find four buttons. They give you the possibility to:

- Activate /deactivate Scheduler jobs
- Run immediately
- See the log/protocol of a Scheduler Job
- Edit Scheduler jobs
- Delete Scheduler jobs
- Use filters to search for specific Scheduler jobs

8.2.8.1. Run immediately

To run a schedule immediately click on the first button in actions column. The job will be executed directly after you pressed the button.

8.2.8.2. Activate / Deactivate Scheduler Jobs

To activate or deactivate click on the second button in actions column. Deactivated Scheduler jobs are greyed out and the trigger is ignored.

8.2.8.3. See the Log/Protocol of a Scheduler Job

When you click the third button in actions column the run history of this Scheduler job opens.

8.2.8.4. Edit Scheduler Jobs

When you click the fourth button in actions column the edit window opens. It is much the same interface as you use when you create a new Scheduler job. For details please see there.

8.2.8.5. Delete Scheduler Jobs

When you click the fifth button in actions column the deletion process starts. A security question is asked. If you not deny, the Scheduler job is deleted.

8.2.9. Export Scheduler Jobs

1. Open Scheduler application.
2. Click **Export** in **Options** menu, a new dialog opens.
3. The system asks you, where to save the export file. After the selection of the destination folder the export starts.

   Some browsers have a default download directory. In this case you are not asked for a destination folder and the export starts immediately.

8.2.10. Import Scheduler Jobs

1. Open Scheduler application.
2. Click **Import** in **Options** menu, a new dialog opens.
3. Choose a definition file to import.
4. Select the desired mode to be used in case of merge conflict between an existing and an imported definition.
   - Overwrite existing data:
   - If not checked, jobs with same name are not imported.
   - If checked, jobs with same name are overwritten and autit is kept.
   - Overwrite ownership: As admin you are allowed to change this setting.

- Click **Import** button and start the import of the chosen import file.

8.2.11. Use Filters to Search for Specific Scheduler Jobs

1. Open Scheduler application.
2. Click the **filter icon**, a new dialog opens.
3. You can set filter criteria and values for Name, Owner, Type, Status and Status latest run.
4. In **Name** and **Owner**, you can search with text.
   In **Type** you can choose ETL, REPORT or SYSTEM.
   In **Status** you search for DISABLED or ACTIVE jobs.
   In **Status last run** you have these values to filter by:
   - FINISHED: Last run was finished successfully
   - FAILED: Last run failed during execution
   - RUNNING: Still running
   - NEW: Waiting in queue for next run
   - SKIPPED: Undefined termination – usually the job was canceled manually by user
5. Click **Ok** and the filtering is activated.
   An active filter is indicated by the **blue color** of the filter icon.
6. To disable filters, click on the **filter icon** and use the red **Delete** button behind a filter criterion to disable that particular filter, or use the **Reset all** button to disable all filters. Click **Ok** to activate your changes.

8.2.12. Use Filters to Search in Scheduler Protocols

1. Open Scheduler application.
   Open protocols.
2. Click the **filter icon**, a new dialog opens.
3. You can set filter criteria and values for Status, Email status and Start time.
4. In **Status** you have these values to filter by:
   - FINISHED: Last run was finished successfully
   - FAILED: Last run failed during execution
   - RUNNING: Still running
   - NEW: Waiting in queue for next run
   - SKIPPED: Undefined termination – usually the job was canceled manually by user

   In **Email status** you have these values to filter by:
   - INITIALIZATION: The process of sending an email is started
   - DISABLED: Not possible to process due to missing values (e.g. recipient)
   - TRYING TO SEND: Sending email in progress
   - FAILED MAIL SERVICE IS NOT CONFIGURED: Failed due to wrong configuration
   - FAILED MAIL ENGINE: Failed while sending mail
   - SENT: Mail, was send successfully

   When filtering by start time it is also possible to define a range.
5. Click **Ok** and the filtering is activated.
   An active filter is indicated by the **blue color** of the filter icon.

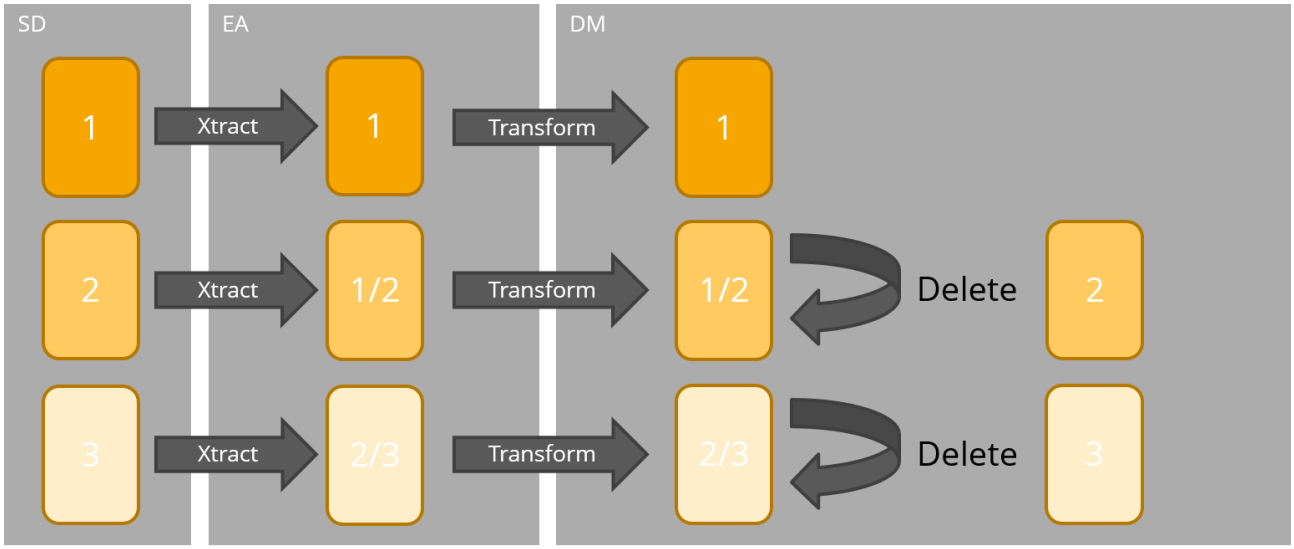
6. To disable filters, click on the **filter icon** and use the red **Delete** button behind a filter criterion to disable that particular filter, or use the **Reset all** button to disable all filters. Click **Ok** to activate your changes.

9. Data Load Modes

9.1. Standard Mode

9.1.1. Standard Mode Single Snapshot

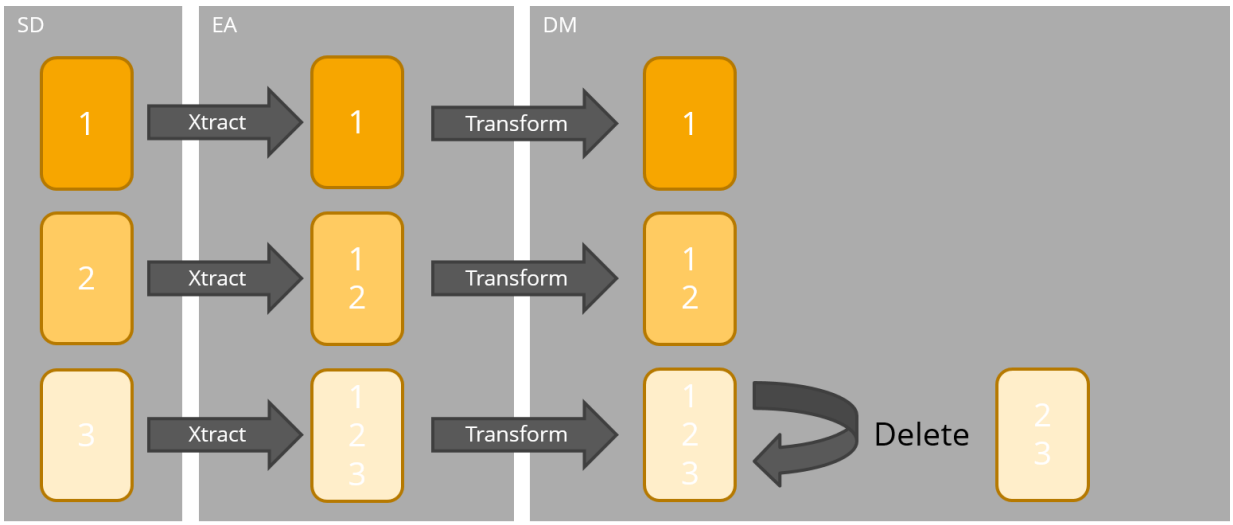
After installation the system operates in this mode.



When you use the standard mode with single snapshot configured, the data load process works like this:

Load	Description
1	The data is loaded from SD (source database) to EA (extraction area). Then the data is transformed by using the Builder definitions and inserted into DM (data mart).
2	New data is loaded from SD to EA. Now we have the data from load 1 and load 2 in EA. The data of load 2 is now transformed by using the Builder definitions and inserted into DM. After that, data from load 1 is deleted from DM.
3	New data is loaded from SD to EA. Now we have the data from load 2 and load 3 in EA. The system was configured to delete obsolete snapshots. The data of load 3 is now transformed by using the Builder definitions and inserted into DM. After that, data from load 2 is deleted from DM.
X	And so on...

9.1.2. Standard Mode Multi Snapshot



When you use the standard mode with multi snapshot configured (keep two days snapshots), the data load process works like this:

Load	Description
1	The data is loaded from SD (source database) to EA (extraction area). Then the data is transformed by using the Builder definitions and inserted into DM (data mart).
2	New data is loaded from SD to EA. Now we have the data from load 1 and load 2 in EA. The data of load 2 is now transformed by using the Builder definitions and inserted into DM. Because we have configured to keep two snapshots, we have load 1 and load 2 in DM. *You can access the loads by using Date loaded dimension in Analyzer.*
3	New data is loaded from SD to EA. Now we have the data from load 1, load 2 and load 3 in EA. The data of load 3 is now transformed by using the Builder definitions and inserted into DM. After that, data from load 1 is deleted from DM. Because we have configured to keep two snapshots we have load 2 and load 3 in DM. You can access the loads be using Date loaded dimension in Analyzer.
X	And so on...

9.1.3. Pros and Cons

Pros:

- This mode is preconfigured and easy to use. No changes are needed.

Cons:

- When errors occur during the DM load, usually the end users are affected. During the rebuild of DM no user can us the Analyzer or Dashboards.
- Inserting in and deleting from heavily indexed tables can result in long runtimes of SQL statements. The more data that is loaded, the more expensive it becomes to continuously update the indices in terms of execution times. In extreme cases, this can lead to timeouts and the termination of the load jobs.

9.2. Staging Mode

You configure the staging mode in System Configuration. The behavior is controlled by following System Configuration parameter:

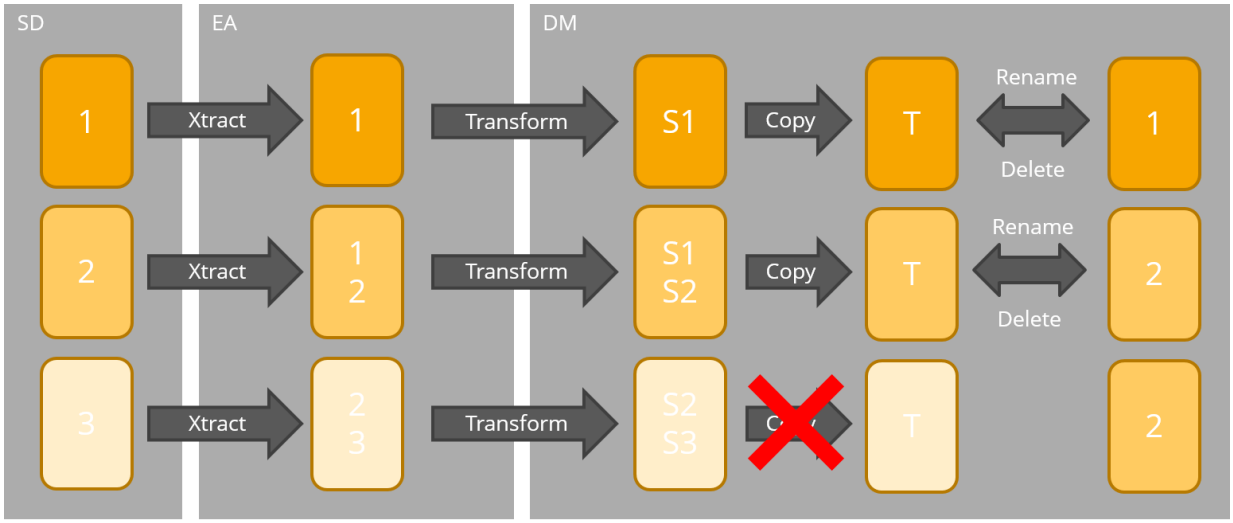
- **Expert Parameters / Staging mode**: When set to true (checked), the system uses the Staging mode
- **Expert Parameters / Staging mode database type**: This parameter is only used, when **Expert Parameters / Staging mode** is set to true.Values:
  - For Oracle, MS SQL Server: Empty
  - For MySQL: INNODB
  - For MariaDB ColumnStore: INNODB or COLUMNSTORE

Before you run a load with staging mode enabled you have to perform these tasks:

- Drop DM
- Enable staging mode in System Configuration
- Recreate time and date dimension
- Create DM
- Start LOAD DM job

Please apply the described procedure otherwise LOAD DM job fails.

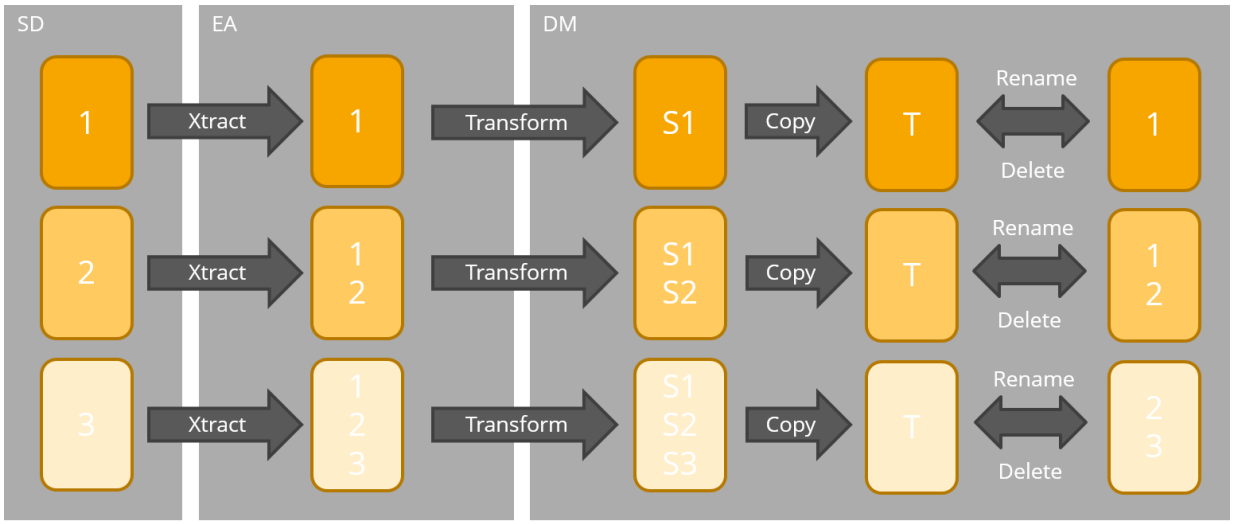
9.2.1. Staging Mode Single Snapshot



When you use the staging mode with single snapshot configured, the data load process works like this:

Load	Description
1	The data is loaded from SD (source database) to EA (extraction area). Then the data is transformed by using the Builder definitions. In DM (data mart) we create a staging structure. The tables have a **_s_** in the name, e.g. fct**_s_**incident. If the creation of the staging structure has finished without errors, we copy this structure to a temporary structure. The resulting tables have a **_t_** in the name, e.g. fct**_t_**incident. In this step the data mart tables are renamed and dropped. E.g. we are renaming fct_incident to fct**_d_**incident and then fct**_t_**incident to fct_incident. After that, all tables renamed to **_d_** are dropped.
2	The same happens for load 2
3	On load 3 an error happens during load into the temporary table structure. The load will be canceled, and an error will be displayed in the Monitor, but the end user can use the data loaded with load 2.

9.2.2. Staging Mode Multi Snapshot



When you use the staging mode with multi snapshot configured (keep two days snapshots), the data load process works like this:

Load	Description
1	The data is loaded from SD (source database) to EA (extraction area). Then the data is transformed by using the Builder definitions. In DM (data mart) we create a staging structure. The tables have a **_s_** in the name, e.g. fct**_s_**incident. If the creation of the staging structure has finished without errors, we copy this structure to a temporary structure. The resulting tables have a **_t_** in the name, e.g. fct**_t_**incident. In this step the data mart tables are renamed and dropped. E.g. we are renaming fct_incident to fct**_d_**incident and then fct**_t_**incident to fct_incident. After that, all tables renamed to **_d_** are dropped.
2	The system is loading load 2 into EA and inserting data to DM. The process we described in load 1 is repeated. Because we configured the system to have two days snapshot in DM we will have load 1 and load 2 in DM.
3	The system is loading load 3 into EA and inserting data to DM. The process we described in load 1 is repeated. Because we configured the system to have two days snapshot in DM we will have load 2 and load 3 in DM.
X	And so on...

9.2.3. Pros and Cons

Pros:

- The advantage of this mode is that you can fill your DM database without disturbing the end users. If errors occur during the load process, the users are only affected to the extent that he or she analyzes for data that is not current.
- We described this for MariaDB usage, but you can also set up this mode for all our supported database systems. If you are using MariaDB, you will especially benefit from the use of MariaDB ColumnStore technology. Internal performance tests showed, that MariaDB ColumnStore is up to 100 times faster than relational database systems with the same data volume, running on comparable infrastructure.
- But also, when you use Oracle, MS SQL Server or MySQL you can have a significant performance increase. Inserting in and deleting from heavily indexed tables can result in long runtimes of SQL statements – caused, among other things, by the constant updating of the indexes.

Cons:

- The downside of the Staging mode is, that you will need more space in your DM database. In the most pessimistic case it will need temporarily three times more, then you had in the old DM.

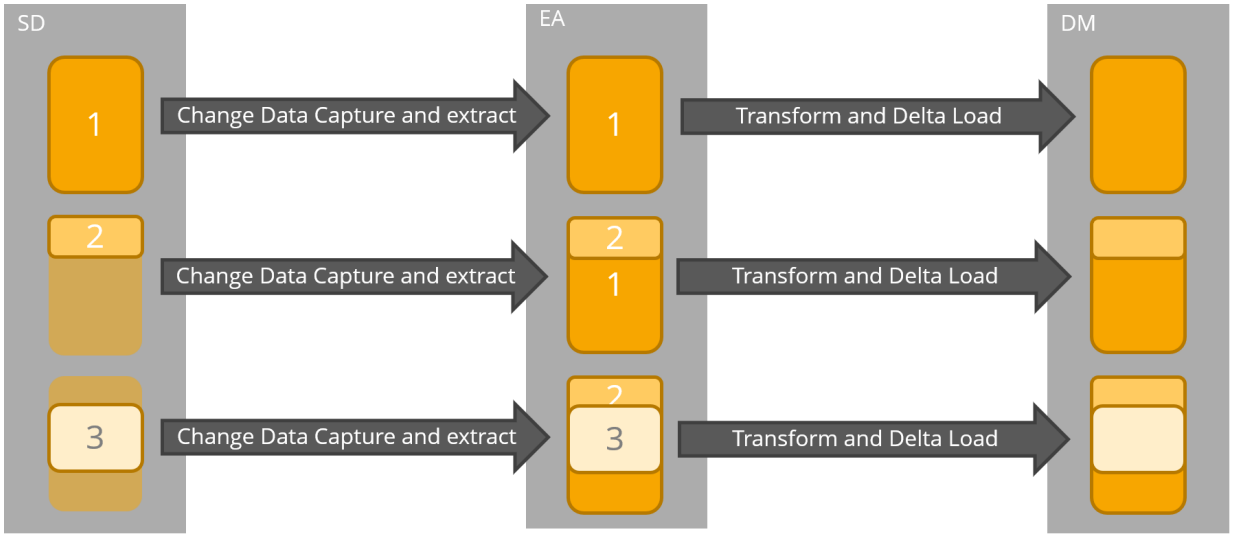
9.3. Change Data Capture (CDC) and Delta Load

You configure the Change Data Capture (CDC) mode in System Configuration. The behavior is controlled by following System Configuration parameter:

- **Expert Parameters / CDC extraction**: When enabled, CDC process is used for extration from source database.
- **Expert Parameters / Delta load**: When enabled, the delta load process is used to update the data mart.

If you want to this CDC/Delta Load functionality, please enable always both parameters.

9.3.1. How it Works



Load	Description
1	On day 1 we start loading the data. The system extracts what it finds in SD and processes the data down to the data mart.
2	On day 2 the system extracts only the changes - records that have changed or been added. Only the delta is processed further.
3	On day 3 the system extracts only the changes again. And also, this time only the delta is processed further.

9.3.2. Pros and Cons

Pros:

- As only the changed data is detected and loaded the whole process is very efficient and ressource saving.
- Faster load intervalles are possible.

Cons:

- Only single snapshot mode is possible.

10. Using DBMS Specific Indexes

We added the possibility to use DBMS specific index types: The Bitmap index on Oracle and the ColumnStore index on MS SQL Server.

In our tests the usage of this indexes improved performance of the data load to DM and the response times of the analyses and dashboards. In our experiments, the loading time was halved, and the analysis times significantly shortened.

MS SQL Server (Mondrian execution time)		Oracle (Mondrian execution time)	
Standard (ms)	**Clustered Column Store index (ms)**	**Standard (ms)**	**BITMAP index (ms)**
1719	689	1491	1177
2064	719	1483	772
1620	933	1492	787
1989	612	1599	779
2037	651	1486	773
1447	780	1499	781

You activate the usage by checking the option **Advanced Indexes on DM layer** in System Configuration in the **BETA Functionality** section. The steps you need to execute to use this feature depend on the data load mode you are using.

1. Check the option Advanced Indexes on DM layer
2. Start job Drop DM
3. Start job Create DM
4. Start job Load DM
5. After that the system will use the advanced indexes on DM layer

1. Check the option Advanced Indexes on DM layer
2. During next DM load the system recreates DM table as part of the Staging mode procedure itself
3. After that the system will use the advanced indexes on DM layer

Bitmap indexes in Oracle and ColumnStore indexes in MS SQL Server are not standard functionalities. Please check if the version and license you are using allows a usage.

11. Multitenancy DSP

We deliver a DynamicSchemaProcessor (DSP) called de.usu.si.mondrian.dsp.VMASchemaProcessor, which allows you to use additional parameters in Mondrian. Together with this we have the attribute Tenant REGEX in System Parameters. This REGEX expression is applied on the roles of the logged in user to extract the tenant roles.

The new parameters are:

- **%{VMA_USERNAME}**: User name of the currently logged in user
- **%{VMA_ROLES}:** All roles of the currently logged in user as commaseparated list (e.g.: 'Administrator', 'tenant_role_01', 'role_01', 'role_02', 'tenant_role_01'...)
- **%{VMA_TENANT}**: First role of the currently logged in user, for which the Tenent REGEX applies (e.g.: 'tenant_role_01')
- **%{VMA_TENANTS}**: All roles of the currently logged in user, for which the Tenent REGEX applies, as commaseparated list (e.g.: 'tenant_role_01', 'tenant_role_01')

If you want to use the parameters, you must modify the existing DynamicSchemaProcessor in the Manage Data Source dialog for the corresponding Mondrian schema. The variables can then be used as in the example below.

```
<Dimension name="systemService" caption="%{msg_210_dim_cap_systemService}" description="%
{msg_210_dim_des_systemService}">
    <Hierarchy hasAll="true" primaryKey="SERVICE_ID">
        <View alias="DIM_SERVICE"/>
            <SQL dialect="generic">
                <![CDATA[
                    SELECT SERVICE_ID,
                    SERVICE_NAME,
                    SERVICE_NAME_DF,
                    TENANT
                    FROM DIM_SERVICE
                    WHERE TENANT IN (%{VMA_TENANTS})
                ]]>
            </SQL>
        </View>
        <Level name="systemServiceName" caption="…" column="SERVICE_NAME"
aptionColumn="SERVICE_NAME_DF" ordinalColumn="SERVICE_NAME_DF" description="…"/>
    </Hierarchy>
</Dimension>
```

12. Security Module

12.1. Overview

USU Analytics Security is a plugin for USU Analytics BA Server which allows creating authorization rules for Mondrian OLAP cubes and storing them in database. It is expected that USU Analytics Security will be used by third party applications by using REST API.

Authorization rules defined in USU Analytics Security are stored in database SIMDR and injected dynamically into Mondrian OLAP cubes on runtime.

With USU Analytics Security it is possible to allow or deny access to particular:

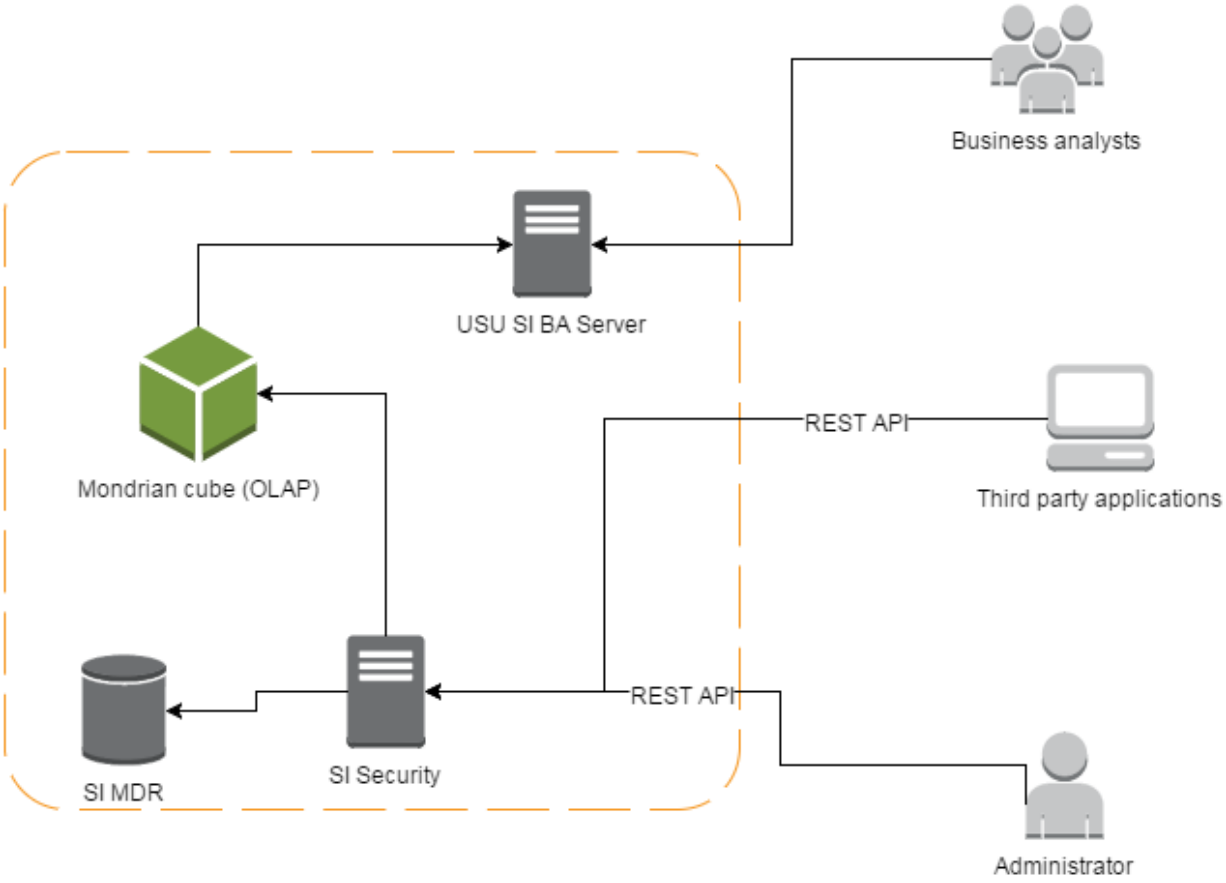
- Mondrian schema
- Mondrian cube
- Dimension
- Member of dimension
- Measure

Authorization rules are always defined for a specific user role.

12.2. Components

USU Analytics Security consists of following software components:

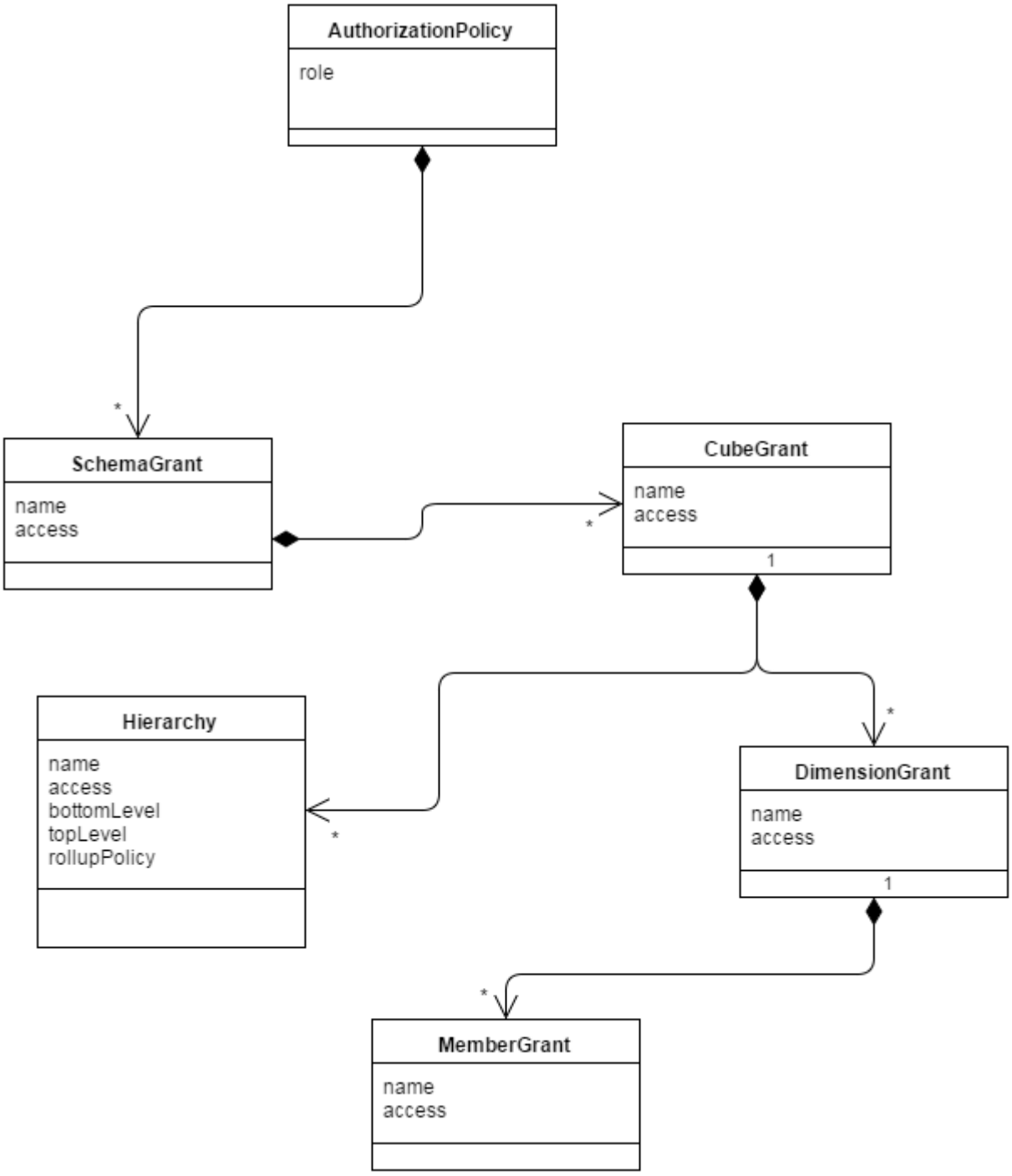
- USU Analytics Security web application – A separate web application, which provides REST API to third party applications and administrators of USU Analytics . It also enrich Mondrian cube used by defined authorization rules.
- ANALYTICS MDR – Database schema which stores information about authorization rules.
- Mondrian cube – Definition of OLAP cube used by OLAP engine Mondrian.



Architecture overview

12.3. Data model

USU Analytics Security data model can be described by following class diagram.



USU Analytics Security data model

AuthorizationPolicy is an object which holds authorization rules for one user role.

SchemaGrant defines the default access for objects in a schema. The access attribute can be "all" or "none"; this access can be overridden for specific objects. In this case, because access="none", a user would only be able to browse the "Sales" cube, because it is explicitly granted.

CubeGrant defines the access to a particular cube. As for <SchemaGrant>, the access attribute can be "all", "custom" or "none", and can be overridden for specific sub-objects in the cube.

DimensionGrant defines access to a dimension. The access attribute can be "all", "custom" or "none". An access level of "all" means that all the child hierarchies of the dimension will get inherited access. An access level of "custom" means that the role does not get an inherent access to the child hierarchies, unless the role is explicitly granted using a <HierarchyGrant> element. Please find more detailed information about grant ALL_DIMENSION behavior here: http://mondrian.pentaho.com/head/documentation/xml_schema.php#SchemaGrant

HierarchyGrant defines access to a hierarchy. The access attribute can be "all", meaning all members are visible; "none", meaning the hierarchy's very existence is hidden from the user; and "custom". With custom access, you can use the topLevel attribute to define the top level which is visible (preventing users from seeing too much of the 'big picture', such as viewing revenues rolled up to the Store Country level); or use the bottomLevel attribute to define the bottom level which is visible (here, preventing users from invading looking at individual customers' details); or control which sets of members the user can see, by defining nested <MemberGrant> elements.

12.3.1. Limitation of Model

You can only define a <MemberGrant> element if its enclosing <HierarchyGrant> has access="custom". Member grants give (or remove) access to a given member, and all of its children. Here are the rules:

- Members inherit access from their parents. If you deny access to California, you won't be able to see San Francisco.
- Grants are order-dependent. If you grant access to USA, then deny access to Oregon, and then you won't be able to see Oregon, or Portland. But if you were to deny access to Oregon, then grant access to USA, you can effectively see everything.
- A member is visible if any of its children are visible. Suppose you deny access to USA, then grant access to California. You will be able to see USA, and California, but none of the other states. The totals against USA will still reflect all states, however. If the parent HierarchyGrant specifies a top level, only the parents equal or below this level will be visible. Similarly, if a bottom level is specified, only the children above or equal to the level are visible.
- Member grants don't override the hierarchy grant's top- and bottom-levels. If you set topLevel="[Store].[Store State]", and grant access to California, you won't be able to see USA. Member grants do not override the topLevel and bottomLevel attributes. You can grant or deny access to a member of any level, but the top and bottom constraints have precedence on the explicit member grants.

For further description of authorization in Mondrian please read official documentation available at http://mondrian.pentaho.com/documentation/schema.php#Access_control.

12.4. REST API

Authorization policies in USU Analytics Security can be created and maintained by using REST API. All REST calls have to be authenticated over HTTP authentication (see chapter REST API Authentication (page 181)).

HTTP header of calls with HTTP method POST, PUT and DELETE must contain following parameters:

```
Content-Type: application/json
```

12.4.1. REST API Authentication

Authentication of access to REST API is done by using container managed security embedded in Tomcat. Tomcat users who can access REST API must have role Valuemation Analytics_TOOLKIT_USER.

Tomcat authentication can be configured in configuration file <Valuemation Analytics/HOME>server/biserver-ee/tomcat/conf/tomcat-users.xml. Following example of configuration in tomcat-users.xml contains defines one user with username si_security_user1, who can access USU Analytics Security REST API.

```
<?xml version='1.0' encoding='utf-8'?>

<tomcat-users>
<role rolename="Valuemation Analytics_TOOLKIT_USER"/>
<user username="si_toolkit_user1" password="CHANGE_ME" roles="Valuemation
Analytics_TOOLKIT_USER"/>
</tomcat-users>
```

More details about contained managed security embedded in Tomcat can be found in Tomcat documentation (https://tomcat.apache.org/tomcat-6.0-doc/realm-howto.html).

12.4.2. Reading Authorization Policies

12.4.2.1. Reading of All Authorization Policies

HTTP Method: GET

URL: `http://<TOMCAT_DOMAIN_NAME>:<TOMCAT_PORT>/sitoolkit_backend/api/authorization_ policies`

Result: Returns all authorization policies stored in USU Analytics Security. Result is in JSON format.

Return code: 200 - OK

12.4.2.2. Reading of One Authorization Policy

HTTP Method: GET

URL: `http://<TOMCAT_DOMAIN_NAME>:<TOMCAT_PORT>/sitoolkit_backend/api/authorization_policies/<AUTHORIZATION_POLICY_ID>`

Required parameters:

- AUTHORIZATION_POLICY_ID **–** Authorization policy identifier

Result: One authorization policy. Result is in JSON format.

Return code: 200 - OK

12.4.3. Create Authorization Policy

HTTP Method: POST

URL: `http://<TOMCAT_DOMAIN_NAME>:<TOMCAT_PORT>/sitoolkit_backend/api/authorization_policies`

Request body: array of policies in JSON

Result: Created authorization policy. Result is in JSON format.

Return code: 201 - OK

12.4.4. Import Authorization Policies

This method deletes all existing authorization policies and creates new authorization policies provided in body of this REST call.

HTTP Method: POST

URL: `http://<TOMCAT_DOMAIN_NAME>:<TOMCAT_PORT>/sitoolkit_backend/api/authorization_policies`

Request body: array of policies in JSON

Return code: 200 - OK

12.4.5. Delete Authorization Policy

HTTP Method: DELETE

URL: `http://<TOMCAT_DOMAIN_NAME>:<TOMCAT_PORT>/sitoolkit_backend/api/authorization_policies/<AUTHORIZAATION_POLICY_ID>`

Return code: 200 – OK

12.4.6. Delete All Authorization Policies

HTTP Method: DELETE

URL: `http://<TOMCAT_DOMAIN_NAME>:<TOMCAT_PORT>/sitoolkit_backend/api/authorization_policies`

Return code: 200 - OK

12.4.7. Examples

12.4.7.1. Create New Member Grant

HTTP Method: POST

URL: `http://<TOMCAT_DOMAIN_NAME>:<TOMCAT_PORT>/sitoolkit_backend/api/authorization_policies`

Request body:

```
[
  {
    "schemaGrants": [
      {
        "name": "Analytics monitoring",
        "access": "NONE",
        "cubeGrants": [
          {
            "name": "Analytics monitoring",
            "access": "ALL",
            "dimensionGrants": [],
```

```
            "hierarchyGrants": [
              {
                "name": "[user]",
                "access": "CUSTOM",
                "bottomLevel": null,
                "topLevel": null,
                "rollupPolicy": "partial",
                "memberGrants": [
                  {
                    "name": "[user].[suzy]",
                    "access": "ALL"
                  },
                  {
                    "name": "[user].[admin]",
                    "access": "NONE"
                  }
                ]
```

12.4.8. Error Handling

In case of error REST API returns a JSON object which contains description of error. HTTP response contains also error code 500 (internal server error).

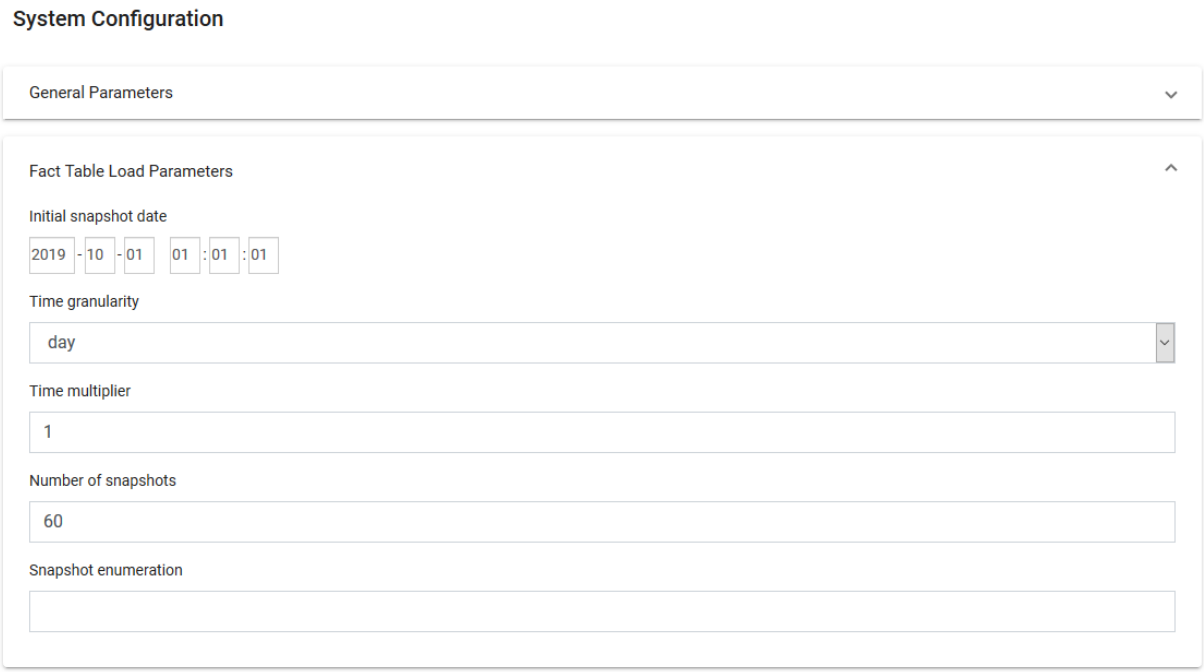
```
{
  "errors": [
    {
      "name": "org.springframework.dao.DataIntegrityViolationException",
      "message": "could not execute statement; SQL
[n/a]; constraint [UK_puo8teeu64ho91hssfrrn55v0]; nested exception is
org.hibernate.exception.ConstraintViolationException: could not execute statement",
      "details": "org.hibernate.exception.ConstraintViolationException: could not execute
state-ment"
    }
  ]
}
```

Example of error message in JSON format

13. Appendix

13.1. Description of Fact Table Related Parameters

To control the behavior of the fact table loading process and the time granularity of the fact table, the following parameters are used:



The meaning of the parameters and how their settings affect the loading process will be explained on the next examples.

Let's assume some facts:

- The structure of all database layers has been already created
- Parameters are set like shown in the screenshot
- The ETL process is scheduled with the same periodicity as the time granularity of the snapshot defined above (1 day).

In order to identify all data records loaded within one ETL execution, the transaction key is used (TK). TK is a timestamp and is by default generated at the time, when the ETL process starts (specifically, when the "Load data from SD→EA" job starts).

In respect to **Initial snapshot date** parameter value and the TK value of particular ETL execution, there are several scenarios, including the corner ones, described. They will help to understand the basic concept:

13.1.1. Scenario 1

The 1st TK equals to **Initial snapshot date**, that means 2014-01-01 23:50:00

To ensure that the transaction key equals to **Initial snapshot date** it has to be set externally.

This is ideal scenario in which the 1st snapshot will be loaded into the DM with the first execution of the ETL (Load data from DW→DM), will be referenced by 2014-01-01 date of the “Date loaded” dimension and will rightly reflect the data loaded on 2014-01-01 23:50:00.

13.1.2. Scenario 2

The 1st TK is higher than **Initial snapshot date**, let it be “2014-01-01 23:55:00”

- 1st execution (2014-01-01 23:55:00): 1st snapshot referenced by 2014-01-01 reflecting data from 2014-01-01 23:55:00 will be loaded to DM
- 2nd execution (2014-01-02 23:55:00): 2nd snapshot referenced by 2014-01-02 reflecting data from 2014-01-02 23:55:00 will be loaded to DM

13.1.3. Scenario 3

Important! This scenario is not recommended to be applied and is listed only to explain the behavior of DM loading in as many corner cases as possible.

The 1st TK is higher than **Initial snapshot date**, let it be “2014-01-02 00:05:00”

- 1st execution (2014-01-02 00:05:00): 1st snapshot referenced by 2014-01-01 reflecting data from 2014-01-02 00:05:00
- 2nd execution (2014-01-03 00:05:00): 2nd snapshot referenced by 2014-01-02 reflecting the same data from 2014-01-02 00:05:00
- 3rd execution (2014-01-04 00:05:00): 3rd snapshot referenced by 2014-01-03 reflecting data from 2014-01-03 00:05:00

13.1.4. Scenario 4

The 1st TK is higher than **Initial snapshot date**, let it be “2014-01-02 23:55:00”

- 1st execution (2014-01-02 23:55:00): in such scenario two snapshots will be loaded to DM:
  - 1st snapshot referenced by 2014-01-01 reflecting data from 2014-01-02 23:55:00
  - 2nd snapshot referenced by 2014-01-02 reflecting data from 2014-01-02 23:55:00
- 2nd execution (2014-01-03 23:55:00): 3rd snapshot referenced by 2014-01-03 reflecting data from 2014-01-02 23:55:00
- 3rd execution (2014-01-03 23:55:00): 4th snapshot referenced by 2014-01-04 reflecting data from 2014-01-03 23:55:00

13.1.5. Scenario 5

Important! This scenario is not recommended to be applied and is listed only to explain the behavior of DM loading in as many corner cases as possible.

The 1st TK is lower than **Initial snapshot date**, let it be “2014-01-01 23:45:00”

- 1st execution (2014-01-01 23:45:00): no snapshot will be loaded (due to the possibility that “newer” data between 23:45:00 and 23:50:00 may still arrive)
- 2nd execution (2014-01-02 23:45:00): 1st snapshot referenced by 2014-01-01 reflecting data from 2014-01-01 23:45:00
- 3rd execution (2014-01-03 23:45:00): 2nd snapshot referenced by 2014-01-02 reflecting data from 2014-01-02 23:45:00

13.1.6. Scenario 6

The 1st TK is lower than **Initial snapshot date**, let it be “2013-12-31 23:55:00”

- 1st execution (2013-12-31 23:55:00): no snapshot will be loaded
- 2nd execution (2014-01-01 23:55:00): 1st snapshot referenced by 2014-01-01 reflecting data from 2013-12-31 23:55:00
- 3rd execution (2014-01-02 23:55:00): 2nd snapshot referenced by 2014-01-02 reflecting data from 2014-01-01 23:55:00

13.1.7. Scenario 7

The 1st TK is lower than **Initial snapshot date**, let it be “2013-12-31 23:45:00”

- 1st execution (2013-12-31 23:45:00): no snapshot will be loaded
- 2nd execution (2014-01-01 23:45:00): no snapshot will be loaded
- 3rd execution (2014-01-02 23:45:00): 1st snapshot referenced by 2014-01-01 reflecting data from 2014-01-01 23:45:00

13.2. REST API

With the Toolkit REST API, you can, e.g.:

- Create database layers for EA and DM
- Start an extraction task
- Load data in data mart
- Monitor data integration processes

Documentation can be provided on request.

13.3. Recommended MariaDB CS settings

In one of the following files:

- `%SI_DBMS_SERVER%\my.ini` ([mysqld] section) for Windows based OS
- `/etc/my.cnf` ([mysqld] section) for UNIX based OS

insert/edit the following parameters:

```
innodb_buffer_pool_size = 2G
innodb_file_per_table = 1
innodb_log_buffer_size = 256M
```

```
lower_case_table_names = 1
max_connections = 500
query_cache_size = 0
query_cache_type = 0
sort_buffer_size = 128M
```

The most important is the `innodb_buffer_pool_size` parameter. The more the better, but it should not exceed 70-80% of server's RAM.

The other (default) MariaDB CS settings should be kept as are unless there's a specific need for an adjustment. Any change can greatly affect the performance of the data integration and should be discussed separately.

If you encounter problems with encoding you can try solving this by adding following parameters to your my.cnf:

```
character-set-server = utf8
collation-server = utf8_unicode_ci

skip-character-set-client-handshake
init_connect = 'SET character_set_client = utf8'
init_connect = 'SET character_set_results = utf8'
init_connect = 'SET collation_connection = utf8_unicode_ci'
```

Generally said, the better hardware is used for the data integration and MariaDB server, the better performance. This is mainly related to the size of RAM, to the HDD/SSD disk's speed and to the performance of a CPU and the number of cores it has.

Running both the data integration and the database instance on the same server oftentimes improves the performance, because any network device (physical or virtual) between the data integration server and the database will likely slow down the whole process. In some cases, where there's a misconfigured network device or a device with a lot of traffic outside the data-integration-to-DB communication, there can be a difference in dozens of percent.

13.4. Naming Conventions

13.4.1. General

- Table names, columns, and aliases use only lowercase ASCII letters (a-z), digits (0-9) and an underscore (_).
- Underscore has a special meaning and is only used as a separator of semantic parts of a name.
- Possible semantic parts of a name are: prefix, root name, role or custom name and suffix (not all parts have to be present, though).
- Table name or table column length does not exceed 30 characters (further restrictions apply to specific tables).
- All tables and columns in a SQL select statement have an alias.
- First data source in a SQL select statement's FROM clause is a table (not an inner select statement).
- Other DBMS (specific) constraints apply (e.g. name cannot start with a digit).

13.4.2. Tables

- Fact tables start with a fct_ prefix.

- Dimension tables start with a dim_ prefix.
- Bridge tables start with a brg_ prefix.
- Prefix is followed by a root name of a table with no further underscores for fact and dimension tables, and with an underscore followed by a role for a bridge table.
- All tables have a technical sim_sqljobid column (this column is not a part of the related source SQL select statement, though).

13.4.3. Columns

- Only primary and foreign key columns end with an _id suffix.
- Only Business Key columns end with a _bk suffix.

13.4.4. Fact Select Statement

- All columns are a fact Business Key, a measure, or a (degenerated) dimension Business Key.
- First column is a fact table Business Key.
- Fact table Business Key comprises of a fact table root name and a _bk suffix.
- Measures comprise of a fact table root name, followed by an underscore and a custom measure name.
- Degenerated dimension columns comprise of a dimension root name and a _bk suffix.
- Dimension Business Keys comprise of a dimension table root name, followed by an underscore, dimension role and a _bk suffix.

13.4.5. Fact Table

- All columns are a primary key, a Business Key, a measure, a degenerated dimension column, a dimension foreign key or a sim_sqljobid column.
- First column is a primary key and comprises of a fact table root name and an _id suffix (this column is not a part of the related source SQL select statement, though).
- Second column is a Business Key and comprises of a fact table root name and a _bk suffix.
- Measures comprise of a fact table root name, followed by an underscore and a custom measure name.
- Degenerated dimension columns comprise of a dimension root name and a _bk suffix.
- Dimension foreign keys comprise of a dimension table root name, followed by an underscore, dimension role and an _id suffix.
- Last column is a sim_sqljobid column.

13.4.6. Dimension Select Statement

- All columns start with a root name of a dimension table, followed by an underscore.
- First column is a dimension table Business Key.

13.4.7. Dimension Table

- All columns are a primary key, a level column, a caption column or a sim_sqljobid column.
- First column is a primary key and comprises of a dimension table root name and an _id suffix.

- Subsequent columns are level columns and caption columns in an order defined by their ordinal position, and ultimately form a hierarchy of the dimension.
- Level columns and caption columns comprise of a dimension table root name, followed by an underscore and a custom level or caption name.
- Last column is a sim_sqljobid column.

13.4.8. Bridge Select Statement

- Select statement has exactly two columns.
- First column is a fact table Business Key.
- Second column is a dimension table Business Key.

13.4.9. Bridge Table

- Root name is a concatenation of a fact table root name, immediately followed by a dimension table root name.
- Full name is a concatenation of a brg_ prefix, followed by a root name, an underscore and a bridge table role name.
- Has exactly four columns.
- First column is a primary key and comprises of a bridge table root name and an _id suffix.
- Second column is a fact table foreign key and comprises of a fact table root name and an _id suffix.
- Third column is a dimension table foreign key and comprises of a dimension table root name and an _id suffix.
- Last column is a sim_sqljobid column.

14. Glossary

Value	Description
DM	Data mart database layer
EA	Extraction area database layer
MD	Metadata database layer
SD	Source database; represents a part or whole database of source system, e.g. USU Service Management