OpenText™ Structured Data Manager

Upgrade Guide

Version : 26.1

PDF Generated on : February 18, 2026

Table of Contents

1. Upgrade Guide

OpenText™ Structured Data Manager provides powerful tools to build an archive solution that copies or moves data out of your online transaction processing database and into less expensive storage.

This guide provides information on the following areas:

- Upgrading Structured Data Manager platform installations from 6.x, or 7.x to 26.1.0 . See Supported upgrade path for detailed upgrade path to 26.1.0 .

Note

To upgrade from earlier versions and implement new integration packs, contact OpenText Enterprise Services for assistance.

Prerequisites

Prerequisites for using this product include:

- Knowledge of the operating system
- Database knowledge
- Application knowledge

Related documentation

Document Name	Description
OpenText™ Structured Data Manager API Reference Guide	Provides reference to the available programming interfaces.
OpenText™ Structured Data Manager Certification Matrix	Provides information about supported Operating Systems, databases, browsers, software integrations and other technology stacks.
OpenText™ Structured Data Manager Concepts Guide	Explains the major concepts of database archiving in general and Structured Data Manager in particular.
OpenText™ Structured Data Manager Developer’s Guide	Explains how to use the Designer component to design, build, test, and deploy your archiving projects.
OpenText™ Structured Data Manager Installation Guide	Explains how to install the product.
OpenText™ Structured Data Manager Release Notes	Lists any items of importance that were not captured in the regular documentation.
OpenText™ Structured Data Manager Runtime Guide	Explains how to use the Web Console component to run, monitor, and administer business flows that move data to and from the database.
OpenText™ Structured Data Manager Troubleshooting Guide	Explains how to diagnose and resolve errors, and provides a list of common errors and solutions.
OpenText™ Structured Data Manager Tutorial	Provides step-by-step instructions to build a sample archiving module, deploy, run, and troubleshoot errors in it.

Document Name	Description
OpenText™ Structured Data Manager Discovery Guide	Explains the purpose, how to install and use Discovery.

1.1. Upgrade Overview

In order to upgrade your Structured Data Manager installation, you must:

- Upgrade the software itself by installing Structured Data Manager release 26.1.0 with the latest patches.

  See the *OpenText™ Structured Data Manager Installation Guide*.

- Upgrade your repository, see Upgrade OpenText™ Structured Data Manager .
- Upgrade environments that require it, see Upgrade OpenText™ Structured Data Manager.
- Upgrade Discovery and Discovery service, see Upgrade Discovery

To upgrade from an earlier version of Structured Data Manager, complete the following tasks:

1. Prepare your installation for the upgrade according to the instructions in Before you begin.
2. Install the new version of Structured Data Manager according to the instructions in the *OpenText™ Structured Data Manager Installation Guide*.

   Note

   If you plan to use the scripted upgrade process, do not start the Web Console nor launch it in your browser at the end of the installation.

3. Upgrade the Structured Data Manager repository and environments in one of the following ways:
   - Upgrade from the Web Console
   - Upgrade with an upgrade script
4. Upgrade Discovery and Discovery service, in the following way
   - Upgrade Discovery
   - Upgrade Discovery Grammar
   - Upgrade Discovery Service

1.2. Upgrade Structured Data Manager

To upgrade Structured Data Manager, you need to perform the tasks in the following sections:

- Before you begin
- Install Structured Data Manager 26.1.0
- Supported upgrade path to 26.1.0
- Upgrade your previous installation
- After an upgrade

1.2.1. Before you begin

Before you perform the upgrade procedures, you must complete the following steps:

Note

- Oracle as a new repository is deprecated. It is only supported for existing installations with Oracle E-business suite modules. It is not recommended for new installation unless management of Oracle E-business suite is needed.
- SQLServer as a repository is not supported.
- For any existing installation using Oracle or SQL server as a SDM repository, migration of repository to PostgreSQL is required. For further assistance please contact Customer Support or Professional Services team.

1. For better upgrade performance, delete all unused environments.

   For more information on deleting environments, see the *OpenText™ Structured Data Manager Runtime Guide* .

   If the database for an environment you want to delete is inaccessible, see the *OpenText™ Structured Data Manager Troubleshooting Guide* for information on how to delete it.

   Caution

   Ensure that you delete only the environments that are not being used.

   Note

   Do not manually drop objects in the schemas created by Structured Data Manager.

> **Note**
>
> When the upgrade process begins, it checks for environments in an installation pending state. Those environments will not appear in the upgraded repository.

2. Ensure that you have applied the latest cumulative patches to your previous installation of SDM before upgrading it.
3. Synchronize your history schema according to the instructions in the *OpenText™ Structured Data Manager Runtime Guide*.

4. Ensure that all jobs using the previous version of Structured Data Manager have been successfully completed or canceled.

    - Jobs that failed prior to copy, delete, or mask actions can be cancelled.

    - Jobs that failed after copy, delete, or mask actions have begun must be resolved and completed. You can cancel the jobs from the Web Console.

    For more information about canceling jobs, see the *OpenText™ Structured Data Manager Runtime Guide*.

5. Uninstall any business flows that use partitioned or bulk data movement.

    > **Note**
    >
    > These data movements are not supported in SDM 7.3 and later, therefore any attempt to upgrade them will fail.

6. Ensure you have collected the following information:

    - The encryption key of the previous installation.

      When you upgrade, the encryption key used for your previous installation is used by default for the release 26.1.0 installation.

      To change the encryption key, use the password manager utility. See the *OpenText™ Structured Data Manager Runtime Guide* for more information.

    - All user names, schema names, database names, and passwords used for the previous software installation.

    - The installation directory used for the previous software installation.

    - The installation directory to be used for the 26.1.0 Structured Data Manager.

7. If you are using Oracle as your source database, perform the following steps:

    1. Stop the Web Console of your previous installation of Structured Data Manager.

    2. Connect to the Oracle repository database instance with your favorite tool (for example DBVisualizer or SQL Developer), using the repository connection credentials to connect to Oracle.

    3. Run the following query to drop the foreign key constraints on tables:

        ```
        -- Drop all table constraints:
        BEGIN
         FOR c IN
         (SELECT c.owner, c.table_name, c.constraint_name
         FROM user_constraints c, user_tables t
         WHERE c.table_name = t.table_name
         AND c.constraint_name NOT LIKE 'SYS%'
         AND c.table_name LIKE 'OBT%'
         AND c.constraint_name LIKE '%FK%')
         LOOP
         dbms_utility.exec_ddl_statement('ALTER TABLE "' || c.owner || '"."' || c.table_name || '" DROP CONSTRAINT ' || c.constraint_name);
         END LOOP;
        END;
        ```

        Note

        If you are using Oracle 12c as your source database, first upgrade your source database to Oracle 19c before upgrading to Structured Data Manager 26.1.0.

8. If your using `TrimClient.jar` file for any of your integrations, then ensure to copy the `TrimClient.jar` file from `<previous_installation_path>/obt/lib` to `<new_installation_path>/obt/lib`.

9. For any business flows that contain upload activities, regenerate them from the Designer 7.2 or above.

Business flows from older versions with an upload activity require a new parameter for schema mapping, therefore you must regenerate them in Designer before the upgrade.

10. Ensure that all Web Console users have an email address defined.
11. (*Optional*). Assess your environments to decide if you want to skip upgrading any of them during the upgrade.
12. Stop the Web Console of your existing Structured Data Manager.
13. Uninstall the Web Console service if it exist.

    > **Note**
    >
    > Post upgrade, if you want to run Web Console as a service, follow the steps in the section *Start Web Console as a Windows service* available in *OpenText™ Structured Data Manager Runtime Guide*.

14. Uninstall the Discovery service if it exist.

1.2.1.1. Home directory

Structured Data Manager stores its program data, such as configuration, log, and archive files, in a location that is separate from its program files. This home directory is determined at installation time:

> **Caution**
>
> Ensure not to delete the contents under `OBTHOME\archive` folder if you want to perform any operations (upload, reload and so on) on those archived files.

- By default, Structured Data Manager attempts to configure its home directory in the standard location for application data on the operating system. For example, on Windows, the Installer tries to use `C:\`.
- If you prefer not to use the default location as determined by Structured Data Manager, you can explicitly set the OBT_HOME environment variable prior to installation to create a custom location. See, the section *Manage the home directory* in *OpenText™ Structured Data Manager Runtime Guide* for instructions.

1.2.2. Install Structured Data Manager 26.1.0

Obtain and install Structured Data Manager 26.1.0 according to the instructions in the *OpenText™ Structured Data Manager Installation Guide*.

You must install 26.1.0 in a different location than the earlier version. That is, do not overwrite the earlier version.

Note

- The instructions in this guide assume that you are installing 26.1.0 on the same server as the earlier version.
- If you plan to use the scripted upgrade process, do not start the Web Console or launch it in your browser at the end of the installation.

1.2.3. Supported upgrade path to 26.1.0

If the repository is on versions prior to 7.2.1 through 7.5.3, then it is mandatory to upgrade the repository in the following sequence to upgrade to 26.1.0 :

```
7.2.1 -> 7.3.5 -> 7.4.9.2 -> 7.5.3 -> 7.6.7 -> 26.1.0
```

For example, if the repository is on 7.3.1 version, then you must upgrade to 7.3.5 (consider the highest version of 7.3x) first, next 7.4.9.2, next 7.5.3, next 7.6.7 and then to 26.1.0.

Similarly, if the repository is on 7.4.2 version, then you must upgrade to 7.4.9.2(consider the highest version of 7.4x) first, next 7.5.3, next 7.6.7 and then to 26.1.0.

1.2.4. Upgrade your previous installation

You can upgrade your previous installation from the Web Console or the command line:

- Upgrade from the Web Console
- Upgrade with an upgrade script
- Upgrade from an Installer

The upgrade process includes the following major steps:

1. Prepare your installation for the upgrade according to the instructions in Before you begin.
2. Import the repository from your previously installed release (7.6.7 or 25.3.0). This step copies all of the required files from the existing installation path to the new (26.1.0) installation path.

   Note

   If the repository is installed on versions prior to 7.6.7, then you must first upgrade the repository to 7.6.7 before upgrading to 26.1.0.

3. Upgrade the repository to 26.1.0.
4. Upgrade the environments associated with the repository, which includes redeploying the business flows where necessary.

Note

Prerequisites for Oracle repository and Oracle, DB2 or Sybase environments

- Before upgrading the existing repository and environment, make sure that all their respective libraries are properly configured. Refer, Configure the third-party libraries in *OpenText™ Structured Data Manager Runtime Guide*.

1.2.4.1. Upgrade from the Web Console

To upgrade the repository from the Web Console

1. If you cannot run your Web Console on the default port, 8080, then you need to change it prior to starting the Web Console.

   For more information about changing the port, see the *OpenText™ Structured Data Manager Runtime Guide*.

2. Start the Web Console for your release 26.1.0 of Structured Data Manager.

   > **Tip**
   >
   > On Windows, you can start the Web Console from the Start menu program group for Structured Data Manager. For more information on starting the Web Console, see the *OpenText™ Structured Data Manager Runtime Guide*.

3. Launch your browser, and connect to the Web Console using its URL:

   ```
   http://<hostname>:<port>/WebConsole
   ```

   where *<hostname>* is the name of the machine on which you installed Structured Data Manager, and *<port>* is the port for the Web Console.

   When starting Web Console for the first time after installing Structured Data Manager, you are prompted to either install a new repository or import an existing one.

4. Click **Import Repository from a Previous Release**.

   > **Note**
   >
   > If you do not have an existing repository from a previous release, no upgrade is necessary. See *OpenText™ Structured Data Manager Runtime Guide* for information on installing a new repository.

5. Enter the path of the previous release.

   For example:

   ```
   C:\SDM2530
   ```

6. Click **Import Repository**.

   As the import proceeds, you receive a message that the Console is restarting and eventually the Web Console login page appears.

7. Log in with administrative privileges as a user from the instance you are upgrading.

   After you log in, the Web Console opens a page prompting you to upgrade the repository.

   > **Note**
   >
   > While upgrading from 23.2.0 to 24.1.0, the prompt to upgrade the repository is not available.

8. Click **Upgrade Repository**.

   The repository upgrade proceeds.

9. Click **Continue**.

   - If environments require upgrade, the **Manage Environments** page opens.
   - If not, the normal landing page displays and you can begin operations.

   What you see next, depends on whether you have environments that need upgrading:

   - If no environments need upgrade, you can resume normal operations.
   - If some environments need upgrade, the **Manage Environments** page opens, where environments are flagged for upgrade and you can selectively upgrade them.

     Until you upgrade those environments, their use is restricted. For example, menu items such as **Launch** are disabled when an environment in need of upgrade is the active environment. Environments not in need of upgrade operate without these restrictions.

10. For those environments you wish to upgrade:

> **Note**
>
> Upgrading the environment includes an automatic attempt at redeployment of any business flows that require it.

1. Click **Upgrade** to start the upgrade process. You may initiate the upgrade of other environments at this point if you wish.
2. If the environment upgrade fails, you can recover by rerunning it.

11. Perform the steps in After an upgrade.

1.2.4.2. Upgrade with an upgrade script

The upgrade script enables you to upgrade in a batch process rather than an interactive one. This process may be more convenient in cases where you have a large number of environments.

You can run the upgrade phases individually or all at once.

To run the upgrade script

1. Navigate to the following directory:

   `<install_dir>/obt/install`

   where *<install_dir>* is the location where you installed the 26.1.0 software.

2. Run the upgrade script according to the following syntax:

Operating System	Syntax
UNIX	`./upgrade.sh -f <upgrade.properties> -I <old_installation_path> -P -p -a -i <list_of_environments> -x <list_of_environments>`
Windows	`upgrade.bat -f <upgrade_prop_file> -I <old_install_path> -P -p -a -i <list_of_environments> -x <list_of_environments>`

   where

   `-f` *`<upgrade_prop_file>`* runs with the specified upgrade properties file. Upgrade properties file.

   `-I` *`<old_install_path>`* imports the product configuration files from the specified existing installation.

   `-P` performs a repository upgrade.

`-p` reads the encryption key from `stdin`. If not specified, you will be prompted for the encryption key.

`-a` updates all environments at once.

`-i` *`<list of environments>`* upgrades the specified environments. If you use this option, it overrides `include.env` in your properties file.

> **Note**
>
> You cannot use `-i` and `-x` together in the same command.

`-x` *`<list_of_environments>`* upgrades all of the environments except the specified ones. If you use this option, it overrides `skip.env` in your properties file.

> **Note**
>
> Upgrading the environment includes an attempt at automatic redeployment of any business flows that require it.

Example

The following example runs import, repository upgrade, and environment upgrade for env1 and env2 on Windows.

```
upgrade.bat -f ..\config\upgrade.properties -I
C:/DBArchiveing630/obt -P -i env1,env2
```

This command line performs just the environment upgrade for `env1`.

```
upgrade.bat -i env1
```

This command line runs import, repository upgrade, and environment upgrade for env1 and env2 on UNIX.

```
./upgrade.sh -f ../config/upgrade.properties -I /home/obt/
DBArchiveing630/obt -P -i env1,env2
```

3. Restart the Web Console.

4. If any of the environment upgrades fail, you can recover them by running the following command:

```
upgrade.bat -f <upgrade.properties> -i <environments>
```

5. Perform the steps in After an upgrade.

1.2.4.2.1. Upgrade properties file

If you use the `-f` argument on the command line, you must create an upgrade properties file. You can include the following in the properties file:

- Repository, source or history database credentials.

  If these properties are not provided in the properties file, then you will be prompted in the console.

- List of environments to be included or excluded.

Property	Description
repository.dbadmin.username	The database administrator user name for the repository. For example, `system`.
repository.dbadmin.password	The database administrator password for the repository.
<env_name>.source.dbadmin.username	The database administrator user name of the source database for the environment. For example, `sa` for SQL Server or `system` for Oracle.
<env_name>.source.dbadmin.password	The database administrator password of the source database for the environment.
<env_name>.hist.dbadmin.username	The database administrator user name of the history database for the environment. For example, `sa` for SQL Server or `system` for Oracle.
<env_name>.hist.dbadmin.password	The database administrator password of the history database for the environment.
include.env=<env1,env2...>	A list of environments to include in the upgrade. This option is overridden by -i on the command line.
skip.env=<env1,env2,...>	A list of environments to exclude from the upgrade. See Skipping or including environments. This option is overridden by -x on the command line.

Caution

If you choose to include any passwords in the properties file, they are stored in plain text in the file.

For security purposes, omit the entire password property line from the properties file. You will then be prompted for the passwords, and they will not be stored in the properties file.

See the sample in:

```
<install_dir>\obt\config\upgrade.properties.sample
```

1.2.4.3. Upgrade from an Installer (In-place upgrade)

You can upgrade from the installer (In-place upgrade) if you have prior installation of Structured Data Manager and want to continue using the same OBTHOME used with prior version of SDM installed on your machine.

To upgrade from an installer

1. Launch the installer.
2. Choose **Upgrade** option and click **Next**.
3. Specify the folder where previous version of SDM is installed. Click **Next**.
4. Select the default location by choosing the default folder where you choose to install, or enter the location for the software to be installed. Click **Next**.
5. The home directory of the previous version of SDM installation is displayed. Click **Next**.

   > **Note**
   >
   > To copy additional jar files, see section Copy additional jar files.

6. Choose the **Web Console/Tomcat Listen Port** (default is 8080) and **Web Console/Tomcat Shutdown Port** (default is 8005). Click **Next**.
7. Select the location to create product icons. Click **Next**.
8. Review the Pre-Installation Summary. Click **Install**.

   After the software is installed, the optional PostgreSQL Install dialog opens.

9. To open the PostgreSQL Install wizard, select **Install PostgreSQL**.

   If you do not wish to install PostgreSQL, select **Don’t Install PostgreSQL**. Click **Next**.

10. The Launch Options page opens.
    1. Select the check box to start the Web Console server using the Web Console/Tomcat Listen Port, or manually start the Web Console server from the command line.

2. You can also select to launch your default browser with the default Web Console URL when you close the Installer.

11. Click **Next**.

12. Click **Done** to close the installation software.

13. Log in with administrative privileges as a user from the instance you are upgrading.

    After you log in, the Web Console opens a page prompting you to upgrade the repository.

    > **Note**
    >
    > While upgrading from 23.2.0 to 24.1.0, the prompt to upgrade the repository is not available.

14. Click **Upgrade Repository**.

    The repository upgrade proceeds.

15. Click **Continue**.

    - If environments require upgrade, the **Manage Environments** page opens.
    - If not, the normal landing page displays and you can begin operations.

    What you see next, depends on whether you have environments that need upgrading:

    - If no environments need upgrade, you can resume normal operations.
    - If some environments need upgrade, the **Manage Environments** page opens, where environments are flagged for upgrade and you can selectively upgrade them.

      Until you upgrade those environments, their use is restricted. For example, menu items such as **Launch** are disabled when an environment in need of upgrade is the active environment. Environments not in need of upgrade operate without these restrictions.

16. For those environments you wish to upgrade:

Note

Upgrading the environment includes an automatic attempt at redeployment of any business flows that require it.

1. Click **Upgrade** to start the upgrade process. You may initiate the upgrade of other environments at this point if you wish.
2. If the environment upgrade fails, you can recover by rerunning it.

1.2.5. After an upgrade

After the upgrade completes you should perform the following tasks:

1. If you are doing an In-place upgrade on Linux, you must copy all the jars in the JDBC directory from the previous installation to the current installation.

   To copy the jars, perform the following steps:

   1. Run the following commands in the command prompt:

      ```
      ...> cd <SDM Install Dir>/obt/bin
      ...> copy_jar.sh "
      <previous_installdir>/obt/lib/jdbc/*.jar" -jdbc
      ```

   2. Restart Web Console and Designer to load the copied jars.

2. If you are upgrading from SDM version prior to 7.6.5 and using DB2 as your source database, you need to enable support for rule validation in Designer for DB2.

   To enable rule validation, perform the following steps:

   1. Navigate to **<OBTHOME>\config\dbms\DB2_<*Version*>** folder.
   2. Using the text editor, open **DB2_<*Version*>.properties** file.
   3. Set the value for **SUPPORTS_VALIDATE=true**.
   4. Restart the Designer.

3. If you are upgrading from SDM version prior to 7.6.5 with Intrusive environment(s) and if any of the business flows contain IPM cartridge, then redeploy those business flows.

   > **Note**
   >
   > The unmask fails, if you redeploy the IPM business flow or cartridge after masking the data.

4. Review historical data in the Web Console to confirm that the upgrade preserved your previous metadata.

5. Confirm the location of your business flow artifacts. If you do not find them in the location you expected, check the paths in `OBTHOME/config/directories.properties`.

6. Check the logs of the upgraded environments for any business flows that failed redeployment. If any business flows did not redeploy properly, you can redeploy them manually. See the *OpenText™ Structured Data Manager Runtime Guide*.

7. Copy the required properties from `webConsole.properties` and `outerbay.properties` into the files of the same name in your new installation.

8. If OBT_JAVA_VM_OPTIONS were changed in the `<previous_installdir>/obt/bin/obt-launcher.bat | sh` file, then you must incorporate those changes in the `<new_install_dir>/obt/bin/obt-launcher.bat | sh` file.

   You must restart the Web Console for the changes to take effect.

   Note

   JDBC properties and SQL tuning properties are brought over from the earlier installation you are upgrading.

9. If you were using SSL for Content Manager, you must re-import the certificate. See the *OpenText™ Structured Data Manager Runtime Guide* for information on how to import a certificate.

10. If you installed an Asian font pack to view the PDF generated by Designer's Generate Documentation feature in your earlier version, you must update the following property in `outerbay.properties` in config under the home directory. See Home directory.

    ```
    pdf.font=STSong-Light-H
    ```

    See the comments inside of `outerbay.properties` for more information.

11. Check your Groovy scripts for as Type constructions and replace them.

    For example:

    ```
    List<Integer> prices = priceValues.collect{it as Integer}
    ```

    would change to:

    ```
    List<Integer> prices = priceValues.collect{it.toInteger()}
    ```

12. Follow the instructions in Set password security.

13. Change the paths referenced in the `log4j2.properties` file, which is located in the home directory under config. See Home directory.

    > **Note**
    >
    > log4j upgraded to log4j2. So to customize the configuration, you must manually set the properties of `log4j2.properties` file with the new syntax by referring to `log4j.properties_pre_upgrade` file.

1.2.5.1. Set password security

Starting with the 6.4 release, Structured Data Manager's security has been increased with a new password encryption algorithm for Web Console users. When you upgrade from a previous version, it is turned off by default.

> **Note**
>
> The password encryption algorithm does not apply if you are using LDAP.

To activate the password encryption algorithm, you must set the `security.passwd.advanced_encryption` property to true.

You will lose the earlier passwords when you set the `security.passwd.advanced_encryption` property to true. You must use the Lost password feature to reset the password. See Reset lost password.

To set the password property

1. Stop the Web Console.
2. From `<installerdir>/OBTHOME/config`, open the `webConsole.properties` file in a text editor.
3. Set the `security.passwd.advanced_encryption` property to `true`.
4. Save the file.
5. Start the Web Console.

1.2.5.2. Reset lost password

To reset your password, you should have specified an email address in your user account before changing the Web Console property. If you did not specify an email address before changing the Web Console property, do one of the following actions:

- Specify an email for each user.
- Specify an email for the administrator so that the administrator can restore his password and then reset passwords for all other users.

1.2.5.3. User groups

Structured Data Manager 7.2 introduces Web Console user groups. After you upgrade to version 7.2 or later, the system automatically places each existing user into an individual user group. You can reorganize these groups as needed after the upgrade.

1.2.5.4. Data Privacy and Protection Appliance Upgrade

Important

- From SDM 7.6.6 onwards, the Masking Server requires Data Privacy and Protection Appliance SimpleAPI 6.20.0.
- Make sure that you delete/uninstall any other versions of SimpleAPI and keep only SimpleAPI 6.20.0 version.

1. If you have used any of the Data Privacy and Protection Appliance (DPP Appliance) masking functions in any of the cartridges prior to SDM 7.6.5 then you will find those functions as Groovy files (names starting with **<*prefix*>** provided by you) under **<OBT_HOME>/extensions/runtime/masking**.

   Take a backup of these files and then regenerate DPP Appliance masking functions. For more information on generating masking functions, see *OpenText™ Structured Data Manager Developers Guide*.

2. The newly generated SecureData masking functions will be available under **<OBT_HOME>/extensions/runtime/masking/<*prefix*>**, where prefix is provided by you during function generation.

1.3. Upgrade Discovery

If you had installed Discovery in previous versions of SDM, then it is mandatory to perform the following steps while upgrading to SDM 26.1.0 .

Post upgrade, perform the following steps:

1. Take a backup of Webconsole folder from the path`[OBTHOME]/extensionh`.
2. Shutdown the Web Console.
3. Uninstall the Discovery service if it exist.
4. Run the following commands in the command prompt:
   - **On Windows**:

     ```
     ...> cd <SDM Install Dir>\obt\bin
     ...> DiscoveryInstaller.bat
     ```

   - **On Linux**:

     ```
     ...> cd <SDM Install Dir>/obt/bin
     ...> DiscoveryInstaller.sh
     ```

5. Start the Web Console.

Note

If you have masking functions generated from the previous version of Discovery, then regenerate those masking functions after an upgrade.

Important

After upgrading to SDM Discovery 25.1.100, Core Data Discovery & Risk Insights (Voltage Fusion) users connecting to a MySQL source via JDBC source must update the connection URL format from `jdbc:mysql://<IP>:<PORT>/schema-name` to `jdbc:mariadb://<IP>:<PORT>/schema-name`.

Post Discovery upgrade

Once you upgrade Discovery, you must upgrade the grammar set. Perform the following to upgrade:

> **Tip**
>
> Steps 1 to 3 need only be performed for the <Custom Grammar Set> file.

1. Log in to **Web Console**, and click **Discovery** tile.
2. Navigate to **Grammars** tab and click **Sets**.
3. Click **Publish** icon to publish the <Custom Grammar Set>, where <Custom Grammar Set> is the user defined grammar set.

   This generate file with the same name of the custom grammar set in the folder `<OBTHOME>\config\obtdiscovery`.
4. Go to folder `<OBTHOME>\config\obtdiscovery`.
   1. Open files <*Custom Grammar Set*> and `PII-def.xml` / `PII-def.html` / `PII-def.test` in a text editor.
      1. Change the **regex** value in the below pattern

Grammar Type	Grammar Rule	Regex
Telephone Number	Portugal Formatted	`\b(\+351)?[ -]?[2]\d{1,2}[ -]?\d{6}\b`
	Spanish Formatted	`\b(\+34)?[ -]?\d{9}\b`
	German Formatted	`\b((0\d{6,7}[ -]?\d{4})|(03\d{6}[ -]?\d{4}))\b`
Postal tt	Spain Posttt	`\b^(?!00)([0-5][0-9])[0-9]{3}$\b`
	Canadian Posttt	`\b( |^)[ABCEGHJKLMNPRSTVXY]{1,1}[0-9]{1,1}[ABCEGHJKLMNPRSTVWXYZ]{1,1}[0-9]{1,1}[ABCEGHJKLMNPRSTVWXYZ]{1,1}[0-9]{1,1}( |$)\b`
	Belgium Posttt	`\b^[1-9]{1}[0-9]{2}[0-9]{1}$\b`
	Brazil Posttt	`\b\d{5}[\-]?\d{3}\b`

Grammar Type	Grammar Rule	Regex
Identifier: Passport Number	US Passport	`^\d{9}$`
Identifier: National	German National ID	`\b(?![0-9]{9}\b)(?![A-Z]{9}\b)[A-Z0-9]{9}\b`
Street Address	Brazil Street	`\b([a-zA-ZÀ-ÿ][-a-zA-ZÀ-ÿ]*[ ,]+[0-9]{1,4}[-a-zA-ZÀ-ÿ]+\d{5}[\-]?\d{3})\b`
	Belgium Street	`\b(?:[a-zA-ZÀ-ÿ][a-zA-ZÀ-ÿ-]*[ ]?)[1-9]{1}[0-9]{2}[0-2]{1}[a-zA-ZÀ-ÿ]*\b`
	Portugal Street	`\b([a-zA-ZÀ-ÿ][-a-zA-ZÀ-ÿ ]*[ ,]+\d{2}[ ,]\d{1}[ º]?Esq\d{4}-\d{3}[a-zA-ZÀ-ÿ]+)\b`
	Spain Street	`\b([a-zA-ZÀ-ÿ][-a-zA-ZÀ-ÿ ]*[ ,]+\d{2}[ ,]\d{1}[ º]?[\D|Izq|Dcha]\d{5}[-a-zA-ZÀ-ÿ])\b`

2. Change the **PATTERN** tag for **Swift tt** under **Bank Account** type as below:

   ```
   <PATTERN name="Swift tt" regex="\b[A-Z]{6}[A-Z0-9]{2}([A-Z0-9]{3})?\b" locale="ww" enabled="0" img="world.png" groovyfile="validate" methodname="isSwifttt" />
   ```

3. Open command prompt

   - **On Windows**:

     ```
     ...> cd <SDM Install Dir>\obt\bin
     ...> Discovery.bat -a
     ```

   - **On Linux**:

     ```
     ...> cd <SDM Install Dir>/obt/bin
     ...> Discovery.sh
     ```

4. Run the command `load grammar`.

5. Open the file `field.xml` in text editor.

   1. Expand the `FIELDTYPES` tag under the attribute `name="string"`, then change the value of attribute `database` to ANY in the line `<JTYPE name="character varying" value="12" database="any"`.

      Tip

      `<JTYPE name="character varying" value="12" database="any" />` change as `<JTYPE name="character varying" value="12" database="ANY"`

   2. Under the `FIELDTYPE` tag, in the following lines add the values to it's following attributes:

Attribute Value	Line to add
string	`<JTYPE name="String" value="12" database="ANY" />`
number	`<JTYPE name="Double" value="8" database="ANY" />`
date	`<JTYPE name="TIMESTAMP(6) WITH LOCAL TIME ZONE" value="93" database="ORACLE"/>`

Upgrade Discovery Service

Post upgrade of SDM to 26.1.0 and Discovery to 26.1.100, to reinstall Discovery services, refer section *Discovery Service installation* in *OpenText™ Structured Data Manager Discovery Guide.*

1.4. Appendix

This section includes:

- Copy additional jar files

1.4.1. Copy additional jar files

During repository import, to automatically copy any additional jar files (for example, third-party-classes-1.0.jar) from SDM 7.65 to SDM later version (for example SDM 7.66)the new property **import.additional.files** is available in the **outerbay.properties** file.

By default all the jars in JDBC directory from previous installation (for example SDM 7.65) are copied to the current installation. (for example SDM 7.66)

If you want this functionality to be available in the versions prior to SDM 7.65, then manually copy the below lines and make appropriate changes.

```
## This property is used to import any additional files added by the user in
# the existing SDM installation to the new SDM installation.
# The JDBC drivers get copied during SDM version upgrade but for other files
# you need to specify in this property.
# The JDBC drivers are copied to $OBTPA_TOP/lib/jdbc, $OBTPA_TOP/lib,
# $OBTPA_TOP/ui/plugins/noneclipsedependencies and
# $OBTPA_TOP/webconsole/apache-tomcat/webapps/WebConsole/WEB-INF/lib
# Usage Example:
# import.additional.files=$OBTPA_TOP/lib/third-party-classes-1.0.jar,\
# $OBTPA_TOP/ui/plugins/noneclipsedependencies/my-util-classes.jar,\
# $OBT_HOME/config/myUtilConfig.properties,\
# $OBT_HOME/extensions/my-ext-classes.jar
# Where,
# $OBTPA_TOP is <SDM Install Folder>/obt
# $OBT_HOME is SDM home directory
# A point to note here, never use absolute path for $OBTPA_TOP or $OBT_HOME
# Also, while doing the upgrade the designer MANIFEST file is updated
# appropriately.
import.additional.files=
```

opentext™

For more info, visit https://docs.microfocus.com