

Upgrading Sunopsis to Oracle Data Integrator

The procedure below details the process for upgrading from a Sunopsis version 3.0.00.00 and later to Oracle Data Integrator. Please read the entire procedure carefully before starting the upgrade.

Note: This procedure only applies to upgrade Sunopsis v3.0 and higher.
If you want to upgrade from an Oracle Data Integrator 10gR3 (10.1.3.x) version, please refer to the section Upgrading Oracle Data Integrator. **Oracle Data Integrator 10.1.3** is the version following Sunopsis v4.1.

Important Notes

Before running the upgrade, please note the following:

- Throughout this procedure, <install_dir> refers to the existing installation directory of Sunopsis. <ODI_HOME> refers to installation directory for the new version.
- The upgrade of a master and work repository performs modifications in the repository's structure. This sequence of modifications cannot be undone and must be done in one continuous step. If an error occurs while upgrading the repository, you must restore the repository database backup before restarting the repository upgrade process.
- Oracle Data Integrator requires a Java virtual machine (JVM) 1.4. To use Web Services with Oracle Data Integrator a Java Development Kit (SDK) 1.4 is required. A JVM 1.4 is provided with the installation programs.
- Upgrading Metadata Navigator is not included in this procedure. Application servers' administrators should un-deploy then reinstall the Metadata Navigator application.

Step 1: Preparing to Upgrade

1.1: Stopping agents services

If you have started the Agent as a Windows Service, you should stop and uninstall this service before upgrading, then reinstall it afterwards. The original service configuration is stored in the backup directory.

1.2: Backup

It is essential that you perform a backup copy of the Sunopsis installation directory and repositories before starting the upgrade process.

1. Make a backup copy of the installation directory of Sunopsis (referred to hereafter as the backup directory).
2. Make a backup of the databases, libraries or schemas where the repositories are stored.

1.3: Preparing to Upgrade Profiles

Customized Profiles

Profiles are automatically reinitialized by the repositories upgrade process. If you have changed the default profiles, then you must make a copy of these modified profiles before proceeding with the upgrade. At the end of the process, you will have to reapply your changes to the profiles which will have been reset by the upgrade, using the duplicates as a reference.

If you have designed your own profiles, then you should update them at the end of the process to take into account the objects and methods added or deleted by the upgrade process. See below for a list of modified objects

Modified Objects

Objects which changed in Sunopsis v4.0, grouped by feature:

- Markers: **Markers, Marker Groups, Life Cycle States** are deprecated.
- Versioning: **Solutions, Solution Elements, Versions**. Note that versioning methods are attached to each object.

Objects which were added in Sunopsis v4.1, grouped by feature:

- Common Format Designer: **Action Groups, Actions, Action Lines, Model Folders, Diagrams**.

Objects which were added/modified in Data Integrator 10.1.3.0:

- The following object types are modified (new methods): datastore, model folders, model, variable, object, action group, action, action line, step report, version, procedure, interface, scenario, object, .
- New object types: **Scenario folder, Open Tools** (This object handles add/delete privileges on Open Tools)

Upgrading Profiles for Versions Prior to Sunopsis 3.2.03.00

If you are upgrading from a version prior to Sunopsis 3.2.03.00, make sure that you have the default profiles installed with the following **Internal ID** values (displayed on the **Version** tab of the **Profile** window):

- CONNECT: 24999
- DESIGNER: 25999
- NG DESIGNER: 28999
- METADATA ADMIN: 26999
- NG METADATA ADMIN: 29999
- TOPOLOGY ADMIN: 34999
- SECURITY ADMIN: 33999
- OPERATOR: 31999
- REPOSITORY EXPLORER: 32999
- NG REPOSITORY EXPLORER: 30999

If the profiles already have these internal ID values, then you can proceed with the upgrade. At the end of the upgrade, check that changes automatically made to the profiles to take into account the new objects and methods are appropriate for your security policy.

If at least one of your profiles has an identifier that does not match the list above:

1. Delete all existing users and profiles using Security Manager.
2. Perform the upgrade.
3. At the end of the upgrade process, re-define the security policy using the profiles imported by the upgrade process.

Step 2: Upgrade the Installed Components

The components upgrade needs to be performed on each machine where a Sunopsis module (Designer, Operator, Topology Manager, Security Manager or Agent) is installed. This upgrade can be performed either using a setup program (recommended) or manually.

Note: The setup program is typically used for upgrading the client machines and graphical modules and the manual setup is typically used to upgrade agents on servers.

Note: The setup program is available for a number of platforms. For other platforms, the manual upgrade process must be used. To retrieve the `/oracledi` directory necessary for the manual upgrade, download the Linux or Windows CD for Oracle Data Integrator from the Oracle Technology Network and use the `/oracledi` sub-directory located at the root of this CD.

Upgrade Notes

The installation program assumes that you install Oracle Data Integrator in a different directory from the old installation directory.

As a consequence:

- The connection settings (stored in `/bin/snp_login_work.xml` and `/bin/snp_login_security.xml` files) are not recovered.
- Configuration parameters stored in `/bin/snpparams.bat` (`.sh`) are not recovered.
- Demo repositories and files (in `/demo/`) as well as the old drivers files (in `/bin/drivers`) are not recovered.

You need to perform manually the steps to carry out this configuration information to the new installation folder. These operations are detailed below in the section 2.3: *Restoring the Configuration*.

2.1: Using the Setup Program to Upgrade

1. Make sure that you have made a backup of `<install_dir>`.
2. Run the appropriate setup program for your platform. For more information, see Installing Oracle Data Integrator in the Oracle Data Integrator Installation Guide. During the setup program, when prompted for the installation folder, select a folder different from the `<install_dir>` folder. This new installation folder is referred to in this procedure as `<ODI_HOME>`. Follow the instructions given by the setup program.
3. Restore the configuration as described in step 2.3, and then uninstall the old version of Sunopsis.

2.2: Upgrading Manually

1. Make sure that you have made a backup of `<install_dir>`. You will need it later.

2. If installing from an Oracle Data Integrator CD, copy the `/oracledi` directory contents of the CD into `<ODI_HOME>` directory. If installing from a downloaded archive file, decompress the Oracle Data Integrator archive file and copy the contents of the `/oracledi` directory into the `<ODI_HOME>` directory. If you transfer the files with FTP, use the BINARY transfer mode.
3. Set the following environment variable:
 - `ODI_JAVA_HOME`: This represents the installation directory of the Java Machine in use with Oracle Data Integrator. This directory should contain the java executable file. If this variable is not set, Oracle Data Integrator will use the default Java Machine.

2.3: Restoring the Configuration

You must perform the following steps for an installation made manually or using the setup program.

1. Manually copy into the installation directory any additional files saved in the backup directory during step 1, including drivers.
2. A new `odiparams` file replaces the old `snpparams`. With a text editor, restore the parameters in the new `odiparams` file in the `<ODI_HOME>/oracledi/bin` folder from the backup `snpparams` file. Once the `odiparams` file is up to date, you can remove the following environment variables if they exist as they are specified in the `odiparams` file.
 - `SNP_JAVA_HOME`
 - `SNP_HOME`
 - `SNP_ADDITIONAL_CLASSPATH`
3. Copy the following files from the backup directory into the `<ODI_HOME>/oracledi/bin` directory:
 - all files with the extension `.xml`
 - all files with the extension `.layout`

Step 3: Upgrade the Master Repository

Note: Upgrading the repositories on Oracle versions before 9i

The repository upgrade uses the CHAR semantics syntax for creating VARCHAR columns on Oracle. This syntax is not supported on Oracle version before 9i. To install a repository on such an Oracle version, edit the `TECH_Oracle.xml` file in the `/lib/scripts/xml/` directory, and replace the `VARCHAR2 (%L CHAR)` string by `VARCHAR2 (%L)`.

You must perform this step once for each master repository that you have.

To upgrade the master repository:

1. On Windows: From the **Start Menu**, select **Programs > Oracle Data Integrator > Repository Management > Master Repository Upgrade**, or run `<ODI_HOME>/oracledi/bin/mupgrade.bat`

On UNIX: Run `<ODI_HOME>/oracledi/bin/mupgrade.sh`.

The Master Repository Upgrade tool appears.

2. Select the **Login Name** corresponding to your master repository connection, or fill in the fields:
 - **Driver**: the driver used to access the technology which hosts the repository.
 - **URL**: The complete path for the data server hosting the repository.

- **User:** The id, or login, of the user who owns the tables
 - **Password:** This user's password.
 - **Technologies:** Select the technology your repository will be based on from the list.
3. Click **OK**.

The repository upgrade process will begin. You can follow its progress on the console. A message appears when the master repository has been upgraded.

Step 4: Upgrade the Work Repositories

You must perform this step once for each work repository.

To upgrade a work repository:

1. Connect to your master repository through **Topology Manager**.
2. In the **Repositories** tree view select **Work repositories**, right click on the repository you want to upgrade and choose **Upgrade**.
3. Oracle Data Integrator will now upgrade this work repository. A window appears when the work repository has been upgraded.

Repeat these steps for each work repository.

Step 5: Import the New Objects

5.1: Objects, Methods and Profiles

This Oracle Data Integrator version introduces new objects and methods, and new privileges on these objects and methods. The objects, methods are automatically added. The default profiles built into Oracle Data Integrator are automatically updated by the upgrade process to include the appropriate privileges on all new objects and methods.

5.2: Technologies

This version of Oracle Data Integrator includes new technology definitions. You must now import these new technologies.

Note the following informations:

- If you want to use the Common Format Designer feature for a given technology, you should re-import the definition for this technology. Re-importing this technology adds the Action Groups defined for the technology, and sets the **Specific Queries** required for the DDL generation.
- It is also recommended that users upgrading from a version 3.2.03.01 or older update all their technologies to benefit from the Current Date query enabling connection recovery.
- In Oracle Data Integrator 10g, most technologies have been updated with minor fixes/updates. It is recommended to updated the technologies that you use to benefit for all these changes.

New Technologies

Technologies new to Sunopsis v3.1:

- Oracle Data Integrator Engine

- XML

Technologies new to Sunopsis v3.2:

- Generic SQL
- Java Beanshell
- Javascript
- Jython
- NetRexx
- PostgreSQL
- SAP Java Connector

Technologies new to Sunopsis v4.0:

- Teradata
- Netezza
- Hyperion Essbase

Technologies new to Sunopsis v4.1:

- All technologies have been updated to support Common Format Designer.

Technologies new to Oracle Data Integrator 10.1.3:

- Axis2
- Derby
- SAS
- Salesforce.com
- Oracle BAM
- All technologies have been updated for use in Common Format Designer.

Importing Technologies

Warning! Importing a technology erases all the changes performed in the technology. It is recommended to perform copies of the technologies you have altered, and reapply the changes to the imported technologies.

1. Run **Topology Manager**. Select the **Physical Architecture** view.
2. Expand the **Technologies** node.
3. Right-click the **Technologies** node, and select **Import Technology**.
4. Select any missing or updated technologies from the `../impexp/` directory.
5. Select the **Synonymy Insert-Update** import mode.
6. Click **OK** to import the new technologies.

5.3: Languages

This Oracle Data Integrator version includes new languages. These languages are automatically updated.

Warning! Importing a language erases all the changes made to this language. It is recommended that you make a copy of the modified languages, and reapply your changes to the updated languages.

New Languages

The updated language files are in the `<ODI_HOME>/oracledi/scripts/xml/` directory.

Languages new to Sunopsis v3.1:

- SQL
- SQL_FILE

Languages new to Sunopsis v3.2:

- JYTHON
- SAP

Languages new to Sunopsis v4.0 and 4.1:

- SQL (Minor Updates in 4.1)

Languages new to Oracle Data Integrator 10.1.3:

- N/A

5.4: Generic Actions

Oracle Data Integrator version includes new and updated actions enabling DDL script generation for the Common Format Designer. These actions must be imported into the master repository.

1. Run **Topology Manager**. Select the **Generic Actions** view.
2. Right-click then select **Import Action**.
3. Select all the actions from the `<ODI_HOME>/oracledi/scripts/xml/` directory.
4. Select the **Synonymy Insert-Update** import mode.
5. Click **OK** to import the actions.

5.5: Knowledge Modules

New versions of existing knowledge modules and new knowledge modules are included in this version.

Although updating knowledge modules for existing projects is not required, it is recommended that you use the new knowledge modules for new projects. Existing scenarios compiled with old knowledge module should still work normally after the upgrade without having to re-generate them. However, one exception to this rule is the **ISO SQL Incremental Update IKM** (see below).

Before importing or updating knowledge modules, you should be aware of the following:

- New knowledge modules should be imported into projects in **Duplication** mode.
- Knowledge modules for new technologies should be imported **after** the associated technologies have been imported.
- Updating an existing knowledge module used into a project should be made using the **Import Replace** option in the context menu of the knowledge module to replace. This option automatically updates any interfaces using the selected knowledge modules. After replacing a knowledge module, the affected interfaces should be thoroughly checked and tested.
- If you have customized the default knowledge modules included in Oracle Data Integrator to suit your needs, you will have to manually implement the latest changes to these original knowledge modules. The details of these changes are given below.

Important exception: KIM ISO SQL Incremental Update

If upgrading from a version prior to Sunopsis v3.2.03.15, you must update this knowledge module. Any projects using this KM should be updated to use the "IKM SQL Incremental Update" provided in this release as described above. Scenarios that use this knowledge module should then be regenerated.

Details of Changes

In this knowledge module, the following change was made in the *Update Existing Rows* command. The column metadata was modified from:

```
$$CSV_COL_LST <%=odiRef.getColList("", "[COL_NAME]", "", "", "",  
"(UPD)")%> $$CSV_COL_LST_END
```

to:

```
$$CSV_COL_LST <%=odiRef.getColList("", "[COL_NAME]", "", "", "", "", "(UPD  
AND (NOT UK) AND (NOT TRG))")%><%=odiRef.getColList("", "[COL_NAME]", "",  
"", "", "(UK)")%> $$CSV_COL_LST_END
```

This change was made in order to comply with the new `$$CSV_COL_LST` metadata syntax implemented in Sunopsis v3.2.03.10. Any customized knowledge module using the old syntax should also be updated.

Step 6: Finalize the Upgrade

Reinstall the agent services which were stopped in step 1.1.

You can now run interfaces, packages, etc developed with previous versions of Oracle Data Integrator.

Your version of Oracle Data Integrator is now up to date!