Oracle® Fusion Middleware

Release Notes for Oracle Data Integrator

14*c* (14.1.2.0.0)

G10244-03

June 2025

Oracle Data Integrator Release Notes

These release notes contain information about the known issues associated with Oracle Data Integrator and the post-installation patches.
It includes the following sections:

- What's New in Oracle Data Integrator?

- Oracle Data Integrator 14c (14.1.2.0.0) ReadMe File

- Oracle Data Integrator Marketplace 14.1.2.0.x Issues and Workarounds

- Configuration Issues and Workarounds

- Design-Time Environment Issues and Workarounds

- Technologies and Knowledge Module Issues and Workarounds

- Post-install Patch Information for Oracle Data Integrator 14c (14.1.2.0.0)

- Documentation Accessibility

What's New in Oracle Data Integrator?

Oracle Data Integrator 14c (14.1.2.0.0) provides several new features.
To view the new features and significant product changes for Oracle Data Integrator in the Oracle Fusion Middleware 14c (14.1.2.0.0) release, see the New and Changed Features for Release 14c (14.1.2.0.0) section in the *Administering Oracle Data Integrator* guide.

Oracle Data Integrator 14c (14.1.2.0.0) ReadMe File

The ReadMe file contains information about the current release. This includes features, prerequisites, install/uninstall instructions, and the like.
A ReadMe file is included in your distribution. It is located in the top level directory of the zip. The ReadMe file includes information about this release (features, prerequisites, install/uninstall instructions). You must use the ReadMe file to install Oracle Data Integrator 14c (14.1.2.0.0). Please read the entire ReadMe file before proceeding.

Oracle Data Integrator Marketplace 14.1.2.0.x Issues and Workarounds

Use this information to understand the known issues related to Oracle Data Integrator (ODI) Marketplace 14.1.2.0.x and their workarounds.
ODI MP 14.1.2.0.x repositories must be on Oracle Database. MySQL-based ODI repositories are not supported.

Currently, options to create embedded repository is not supported in ODI MP 14.1.2.0.x Image.

This section contains information on the following issues:

- Connection to Data Server Fails if Existing ODI 12.2.1.4 MP Repository is used During Provisioning
- Data Server is not Created by Default if Oracle Database 23ai is used for Repository Creation
- Default Data Server JDBC URL has Incomplete Connection Details
- Agent does not Start Automatically if Instance is Created Using Existing Repository
- ODI Agent Fails to Update Schedule Time
- Discover ADBs Feature Fails to Fetch List of Available Autonomous Database Instances
- Oracle Object Storage Data Server Missing Post Upgrade

Connection to Data Server Fails if Existing ODI 12.2.1.4 MP Repository is used During Provisioning

When you configure ODI MP 14.1.2.0.x, if you choose to use an existing ODI MP 12.2.1.4 repository during provisioning, then test connection from the data server fails with the following exception:

```
ORA-17957: Unable to initialize the key store.
```

As a workaround,

1. Login to ODI Studio.

2. In Topology navigator click **Technologies** -> **Oracle** and select the data server.

3. Click the **JDBC** tab and delete the following from the **Properties** table:
   `oracle.net.wallet_location`

   `oracle.net.ssl_server_dn_match`

4. In the **Definition** tab, click the browse icon present beside the **Credential File** text box and browse to select the wallet for the data server from the `/u01/oracle/mwh/wallets` directory. The **Connection Details** text box appears.

5. Choose the connection URL from the **Connection Details** drop down list.

6. Enter the credentials required to open the wallet file.
   The JDBC URL and JDBC driver details are auto-populated with the credentials retrieved from the wallet file.

7. In the **JDBC** tab, validate whether the JDBC connection URL is auto-populated.

8. Click **Save** to save the Data Server details.

9. Click **Test Connection** to test the connection.

The test connection will complete successfully. [38015158]

Data Server is not Created by Default if Oracle Database 23ai is used for Repository Creation

If you use Oracle Database 23ai during repository creation, no data server is created for the available database in the compartment. Note that this issue is not seen when the instance is created using Oracle Database19c. Here data servers are created for the available database in the compartment including Oracle Database 23ai.

To fix this, you need to use the wallet downloaded during ODI MP provisioning from the `/u01/oracle/mwh/wallets` directly and manually create the data server.

To create the data server,

1. Login to ODI Studio.

2. In Topology navigator click **Technologies** -> **Oracle**.

3. Right-click and select **New Data Server**.

4. In the **Definition** tab,

   a. Enter a name of the data server.

   b. Select the **Use Credential File** checkbox to upload the connection details using a pre-configured wallet file.

   c. Click the browse icon present beside the **Credential File** text box and browse to select the wallet for the data server from the `/u01/oracle/mwh/wallets` directory. The **Connection Details** text box appears.

   d. Choose the connection URL from the **Connection Details** drop down list.

   e. Enter the credentials required to open the wallet file.
      The JDBC URL and JDBC driver details are auto-populated with the credentials retrieved from the wallet file.

5. In the **JDBC** tab, validate whether the JDBC connection URL is auto-populated.

6. Click **Save** to save the Data Server details.

7. Click **Test Connection** to test the connection.

The data server is created successfully in the ODI repository. [37994622]

Default Data Server JDBC URL has Incomplete Connection Details

When you create an ODI instance, the default data server is created automatically with pre-populated connection details. You only need to provide the username and password for the created instance to connect to the data server.

However, the default data server that is created for an ODI MP 14.1.2.0.x instance has `jdbc:oracle:thin:@null` in the JDBC URL. Note that this issue does not occur for newly created data servers.

To populate the correct JDBC URL,

1. Login to ODI Studio.

2. In Topology navigator -> **Technologies** -> **Oracle**, select the data server.

3. In the **Definition** tab, click the browse icon present beside the **Credential File** text box and browse to select the wallet for the data server from the /u01/oracle/mwh/ wallets directory.
   The **Connection Details** text box appears.

4. Choose the connection URL from the **Connection Details** drop down list.

5. Enter the credentials required to open the wallet file.
   The JDBC URL and JDBC driver details are auto-populated with the credentials retrieved from the wallet file.

6. In the **JDBC** tab, validate whether the JDBC connection URL is auto-populated.

7. Click **Save** to save the Data Server details.

8. Click **Test Connection** to test the connection.

The JDBC URL details will be saved correctly. [37993161]

Agent does not Start Automatically if Instance is Created Using Existing Repository

The Agent service fails to start automatically if you choose to use an existing ODI MP 12.2.1.4 repository during ODI MP 14.1.2.0.x provisioning. This issue does not occur if you choose to create a new repository during provisioning.

Run `manageappsodi.service` to start the Agent service.

Follow these steps:

1. Kill all jetty processes, if running, using the following commands:
   ```
   ps -ef | grep jetty

   kill -9 <process-id>
   ```

2. Log in to the provisioned ODI instance on Oracle Cloud Marketplace using SSH as opc user:
   ```
   ssh opc@<IP Address>
   ```

3. Run the `start` functionality using `systemctl`.
   ```
   sudo systemctl start manageappsodi.service
   ```

The agent starts successfully. [38014873, 37994411, 37993192]

ODI Agent Fails to Update Schedule Time

When you change the schedules for running any mappings, packages, or load plans, the ODI agent fails to update the schedule time.

To fix this,

1. Close ODI Studio.

2. Stop any Agent service that is running.

    a. Log in to the provisioned ODI instance on Oracle Cloud Marketplace using SSH as opc user:
    ```
    ssh opc@<IP Address>
    ```

    b. Run the `stop` functionality using `systemctl`.
    ```
    sudo systemctl stop manageappsodi.service
    ```

3. Under `/u01/oracle/mwh/odi/common/` create the path `odi-ff/MP`.

4. Create an `ffdefinition.config` file in `/u01/oracle/mwh/odi/common/odi-ff/MP`.

5. Add the following content to the `ffdefinition.config` file:
    ```
    #Features config

    repo-misfire-schedule-retry=false
    ```

6. Start the Agent service.

    a. Log in to the provisioned ODI instance on Oracle Cloud Marketplace using SSH as opc user:
    ```
    ssh opc@<IP Address>
    ```

    b. Run the `start` functionality using `systemctl`.
    ```
    sudo systemctl start manageappsodi.service
    ```

7. Start ODI Studio using the following command:
    Windows: `odi.exe -clean -initialize`

    UNIX: `./odi.sh -clean -initialize`

The schedule time is updated successfully. [37987790]

Discover ADBs Feature Fails to Fetch List of Available Autonomous Database Instances

When you use the **Discover ADBs** feature to display the list additional ADB Data servers in ODI Studio, it fails with a `shaded.com.oracle.oci.javasdk.javax.ws.rs.ProcessingException` error.

To fix this, you need to modify the `/u01/oracle/mwh/odi/studio/bin/odi.conf` file as follows:

After line 66, which is:

```
AddVMOption -DexternalAuthenticatorIsCaseInsensitive=false
```

Add these lines:

```
AddVMOption -Djavax.net.ssl.trustStore=$JAVA_HOME/lib/security/cacerts

AddVMOption -Djavax.net.ssl.trustStorePassword=changeit
```

Restart ODI Studio using the following command:

Windows: `odi.exe -clean -initialize`

UNIX: `./odi.sh -clean -initialize`

The Discover ADBs feature should work correctly. [37987206]

Oracle Object Storage Data Server Missing Post Upgrade

If you had configured an Oracle Object Storage technology data server along with corresponding Logical schema in ODI MP 12.2.1.4 instance and used this schema when provisioning the ODI MP 14.1.2.0.x instance, then in the provisioned ODI MP 14.1.2.0.x, the data server goes missing from the Topology Navigator.

To avoid this,

- Perform a smart export of the Object Storage Data Server objects from the ODI MP 12.2.1.4 environment.
- Provision the ODI MP 14.1.2.0.x instance.
- Import the Object Storage Data server objects to the ODI MP 14.1.2.0.x instance.

Follow these steps:

1. To perform a Smart Export of the Object Storage Data server:

    a. Log in to the ODI MP 12.2.1.4 instance.

    b. Open ODI Studio.

    c. In ODI studio, navigate to **Topology** -> **Technologies** -> **Oracle Object Storage**.

    d. From the Topology Navigator toolbar, Select **Export...**.

    e. In the Export Selection dialog, select **Smart Export**.

    f. In the **Smart Export** dialog, enter a name for the export file.

    g. Provide an **Export key**. You will use this when you import the objects in the ODI MP 14.1.2.0.x instance.

    h. Navigate to Topology tab and drag and drop the Oracle Object Storage data server into the **Selected Objects** list on the left.

    i. Drag and drop the Logical schema to the dialog box under selected section.

    j. Click **Export** to start the export process.

2. Provision the ODI MP 14.1.2.0.x instance.

3. Use SCP to download the exported file to the provisioned ODI MP 14.1.2.0.x instance.

4. Import the exported objects in the upgraded environment.

a. Log in to the ODI MP 14.1.2.0.x instance.

b. Open ODI Studio.

c. From the Topology Navigator toolbar, Select **Import**....

d. In the Import Selection dialog, select **Smart Import**.

e. In the **File Selection** field, enter the location of the Smart Export file to import.

f. Click **Next**.

g. In the **Enter Export Key** dialog provide the export key that you used for the export process.

h. Click **Finish** to start the import process.

Connect to ODI Studio -> Topology -> Technologies -> Oracle Object Storage to verify that the data server is created. [38037301]

Configuration Issues and Workarounds

Use this information to understand the known configuration issues and workarounds for Oracle Data Integrator.
This section contains information on the following issues:

- 999 is a Prohibited Master Repository ID
- Domain Assisted Schema Upgrade (DASU) Does Not Pre-populate ODI Supervisor Credentials
- Unable to Include Dependencies while Creating Version
- `CopyConfig` command cannot connect to ODI schema through External Authentication
- Upgrading commons-lang from 2.6 to commons-lang3-3.8.1.jar in SDK Script File
- Upgraded bpm-infra.jar Causes NullPointer Exception

999 is a Prohibited Master Repository ID

999 is a prohibited master repository ID and should not be used. [21083009]

Domain Assisted Schema Upgrade (DASU) Does Not Pre-populate ODI Supervisor Credentials

In the Oracle Fusion Middleware Upgrade Assistant, when the **All Schemas Used by Domain** option is selected, the Supervisor credentials for ODI are not pre-populated in the first instance as the domain does not contain them. If there are multiple ODI schemas, the Upgrade Assistant populates the user entry using the first set of credentials. [20323393]

Unable to Include Dependencies while Creating Version

When you follow the below steps:

- Enable GIT/Subversion

- Enable wallet

- Create connection to GIT/Subversion

- Add mapping to VCS

- Modify mapping

and then terminate ODI studio and start again to create a version for mapping including dependencies, you get a null pointer error.

To overcome this issue, as a workaround

- Navigate to Team -> Settings - > Edit Connection and click OK

  The wallet password dialog appears.

- Enter the wallet password and then create version with dependency.

You can successfully create version for mapping including dependencies. [25168395]

`CopyConfig` command cannot connect to ODI schema through External Authentication

The `CopyConfig` command cannot be executed on an environment configured with external authentication. It needs internal authentication to connect to the ODI schema. [27084113]

Upgrading commons-lang from 2.6 to commons-lang3-3.8.1.jar in SDK Script File

There is a change in the package structure as `commons-lang` is upgraded from `2.6` to `commons-lang3-3.8.1.jar`. Due to this change, you may get compilation errors if your SDK scripts use `org.apache.commons.lang` package. [29966240]

As a workaround, in the SDK script file, change all the references of `org.apache.commons.lang` to `org.apache.commons.lang3`.

For example :

```
import org.apache.commons.lang.ArrayUtils;
```

to

```
import org.apache.commons.lang3.ArrayUtils;
```

Upgraded bpm-infra.jar Causes NullPointer Exception

Complex file dataserver with JSON file containing value as null (without any double quotations) fails Test connection and Reverse Engineering. [30214609]

JSON Payload must not contain null as a value. As a workaround, replace:

- string with "null", "n/a" or any logical value in double quotations
- integer with the value "0" (zero)

Design-Time Environment Issues and Workarounds

Use this information to understand the known design-time environment issues and workarounds for Oracle Data Integrator.
This section contains information on the following issues:

- Preferences that are Not Used in Oracle Data Integrator Appear in ODI Studio
- Attributes are Not Copied when Duplicating a New Datastore
- Non-ASCII Characters in a Hive Table are Not Displayed Properly
- Editing Expanded Submap of Dimension or Cube component
- Performance Delay While Editing Scenarios and Load Plans

Preferences that are Not Used in Oracle Data Integrator Appear in ODI Studio

Preferences that are not used in ODI are getting picked up from the JDeveloper IDE by default and these features appear in ODI Studio > Tools > Preferences. [21656747]

Attributes are Not Copied when Duplicating a New Datastore

If you attempt to duplicate a newly created datastore with attributes without first closing the tab of the newly created datastore, the attributes are not copied.

As a workaround, save and close the newly created datastore with attributes before selecting **Duplicate Selection**. [21572433]

Non-ASCII Characters in a Hive Table are Not Displayed Properly

Non-ASCII characters in a Hive table that is based on a utf-8 encoded file are not displayed properly. As a workaround, specify `-J-Dfile.encoding=utf8` and start ODI Studio to view Non-ASCII characters in a Hive table. [19632983]

Editing Expanded Submap of Dimension or Cube component

You are not allowed to edit the expanded map of a dimension or cube component. The changes done in expanded map is not persisted and are not saved. [23110100]

Performance Delay While Editing Scenarios and Load Plans

You can observe a significant delay when you try to edit a simple Scenario or Load Plan in the Load Plans and Scenarios view from Designer or Operator UI of ODI

studio. This behavior is observed when more number of child nodes are associated to a parent node. As a workaround, avoid multiple refresh on save operation. Also limit the number of child nodes inside a folder to a maximum of 200, to avoid performance issues. [27395959]

Missing Menu Options in Topology Designer Tree After Successful REST Service Response Test

Perform the following steps in the topology designer tree:

1. Create a new data server for RESTful Service technology.

2. Create a physical schema with some available method supported by REST Service end point URL.

3. Test the response.

4. After successful REST service response test, right click the menu for physical and logical Architecture.

   New Data Server and New Logical Schema options along with some other menu items stay missing.

As a workaround, restart the ODI studio. [29792225]

Technologies and Knowledge Module Issues and Workarounds

Use this information to understand the known technologies and knowledge module issues and workarounds for Oracle Data Integrator.
This section contains information on the following issues:

- LKM Hive to File Direct Fails when Exporting to HDFS

- Log Files are Deleted Even in Case of Failure when Using the OdiOSCommand on Oozie

- Oozie Initialization Fails

- Error Displayed During Oozie Initialization

- SQOOP KMs Fail on Oozie

- Disregard Failed to set setXIncludeAware(true) for parser warnings

- Pig Does Not Provide Implicit Type Conversion

- Mapping Execution Fails in Pig

- Complex Aggregation Not Supported by Pig Latin

- Mapping Editor May Not Display All Template IKMs

- Date Comparison May Not Work as Expected if the Date is a String Datatype

- XKM SQL Distinct Limitation

- The UNION_DISTINCT Pig Operator Does Not Remove Duplicate Outputs

- [Log Level and Log File Not Displayed in the Complex File Dataserver Properties](#)
- [BinaryType Data Type Not Supported in Spark 1.1](#)
- [Hive Complex Datatypes Not Supported by LKM Spark to Hive](#)
- [Spark Execution Supports only YARN Deployment](#)
- [Spark-Cassandra: Permission Errors in YARN-client mode](#)
- [Known Datatype Issues using Spark 1.6](#)
- [Unable to Store Alias Error in Pig](#)
- [KMs Replaced During Repository Upgrade](#)
- [Erroneously Published SDK API Classes Removed from the 12*c* Javadocs](#)
- [CKM Fails with XML and Complex Files When Database is Set to External](#)
- [Flexfields Tab of KM Editor May Not Display Newly Created KMs](#)

LKM Hive to File Direct Fails when Exporting to HDFS

When executing a mapping using LKM Hive to File Direct, it fails and the following error is displayed:

```
ODI-1227: Task Unload Hive data-LKM Hive to File Direct- fails on the source
connection HIVE_DATA_SERVER
```

This is caused due to Hive bugs, HIVE-5672 and HIVE-6410, which cause the INSERT OVERWRITE statement to fail when writing to HDFS. Please note that these Hive bugs are already fixed and the issue is resolved when upgraded to a recent version of CDH and Hortonworks. [21529011]

Log Files are Deleted Even in Case of Failure when Using the OdiOSCommand on Oozie

Many KMs that use OdiOSCommand use the `OUT_FILE`/`ERR_FILE` parameters to redirect output into log files.The directory for such files is based on the KM option `TEMP_DIR`, which uses a default value of `System.getProperty("java.io.tmpdir")`. This causes ODI on Oozie to use an Oozie job temporary directory, which gets cleaned up on job completion, irrespective of whether the job was successful. This results in the log files not being available after execution.

As a workaround, when executing on Oozie, overwrite the KM option `TEMP_DIR` to a specific temporary directory. [21232650]

Oozie Initialization Fails

Oozie initialization fails and the following error is displayed:

```
java.io.IOException: E0504 : App directory <dir_name> doesn't exist OR ODI-1028:
There are issues with the Log Retriever components. No Log Retriever flow with
name <name> is running.
```

The issue occurs on pure CDH5.4.0+ pseudo/multi node clusters.

As a workaround,

1. Make sure that oozie share lib is already created using the following command:

```
oozie-setup sharelib create -fs hdfs:///user/oozie -locallib <path to local
folder [oozie-sharelib-yarn]>
```

> **Note:**
>
> Folder oozie-sharelib-yarn is local to the oozie setup. After creating the
> sharelib, you can verify the sharelib on HDFS at the location hdfs:///user/
> oozie/share/lib/lib_<timestamp>

2. Add the following properties to oozie-site.xml. These properties are needed for Oozie to obtain the hadoop configuration files to access HDFS. In the first property value, add the path after "*="

```
<property>
<name>oozie.service.HadoopAccessorService.hadoop.configurations</name>
<value>*=<replace_this_with_path_to_hadoop_configuration_folder For
Example:/etc/hadoop/conf></value>
<value>*=<replace_this_with_path_to_hadoop_configuration_folder
For Example:/etc/hadoop/conf></value>
</property>
<property>
<name>oozie.service.WorkflowAppService.system.libpath</name>
<value>hdfs:///user/oozie/share/lib</value>
</property>
```

3. Restart your oozie and hadoop services. [21410186]

Error Displayed During Oozie Initialization

There is an issue with the `OdiLogRetriever.properties` file and the
`oozie.coord.application.path` value does not get appended to it and the following
error is displayed: [21410186]

```
E0504: App directory doesn't exist
```

SQOOP KMs Fail on Oozie

KMs using SQOOP fail when executed on Oozie on a CDH version prior to 5.4.1.

As a workaround, set KM option `EXTRA_HADOOP_CONF_PROPERTIES` to `--skip-dist-
cache`. Another workaround for this issue is to upload all SQOOP jars into the HDFS
directory reported in `FileNotFoundException`. [21232570]

For example:

```
hdfs dfs -mkdir -p /opt/cloudera/parcels/CDH-5.3.0-1.cdh5.3.0.p0.30/jars
hdfs dfs -copyFromLocal
```

```
/opt/cloudera/parcels/CDH-5.3.0-1.cdh5.3.0.p0.30/jars/* /opt/cloudera/parcels/
CDH-5.3.0-1.cdh5.3.0.p0.30/jars
```

Disregard Failed to set setXIncludeAware(true) for parser warnings

When you execute Pig or Oozie workflows through ODI, you may encounter warning messages such as, Failed to set setXIncludeAware(true) for parser, regarding xml parsing failures in the ODI logs or studio console. This error occurs when JAVA xerces parser gets used for PIG execution, as the default implementation does not support XIncludeAware feature in xml parsing.

As a workaround, add xmlparserv2.jar in classpath of the Pig Dataserver. [21238180]

Pig Does Not Provide Implicit Type Conversion

When specifying constant expressions, the datatype for the constant must exactly match the attribute datatype because Pig does not provide implicit type conversion. For example, if the attribute is defined as DOUBLE, the constant expression for this attribute should be set to 999.0 instead of 999. [20808984]

Mapping Execution Fails in Pig

When a mapping is processed using Pig and there is an Aggregate component in the Pig staging area, the Having clause must be set differently from similar mappings for SQL-based technologies. [20723728]

Complex Aggregation Not Supported by Pig Latin

When using the Aggregate component in Pig staging, you cannot specify a complex expression in an aggregate function, for example, `SUM(source.col1 + source.col2)`. This kind of aggregation is called "complex aggregation" and Pig Latin does not support this. If complex expression is needed, the Expression component must be added to the mapping ahead of the Aggregate component. [20302859]

Mapping Editor May Not Display All Template IKMs

The Mapping Editor may not correctly list the imported Template IKMs for selection. To list the imported IKMs, you must change the **Target Integration** type from its default (**Control Append**) to either **Incremental Update** or **None**. [20583432]

Date Comparison May Not Work as Expected if the Date is a String Datatype

In the Spark project, if the source file uses File technology, ODI converts the Date into a string datatype. This may cause the Date comparison to fail. [20029929]

XKM SQL Distinct Limitation

When a mapping is created with Oracle as source and Oracle as target using a Distinct component and the XKM SQL Distinct is selected in the DISTINCT node, the mapping fails and the following error is displayed:

```
The physical node DISTINCT_ cannot be supported by technology Oracle on
execution unit src_UNIT of mapping Mapping New_Mapping[11] owning
folder=ODIOGG.First Folder
```

To resolve this issue, upgrade the topology information so Support Distinct Operator is set to `True`. [20234590]

The UNION_DISTINCT Pig Operator Does Not Remove Duplicate Outputs

The UNION Pig operator uses the following modifiers to specify the uniqueness characteristic: [20368827]

- unspecified – Perform a DISTINCT operation on output
- DISTINCT – Perform a DISTINCT operation on input, but not output
- ALL – Do not perform a DISTINCT operation on input or output

Log Level and Log File Not Displayed in the Complex File Dataserver Properties

When creating a Complex File dataserver, the log level (ll) and log file (lf) properties are not displayed in the Properties tab. [20377218]

BinaryType Data Type Not Supported in Spark 1.1

The Hive datatype, BinaryType, is not supported in Spark 1.1. When using LKM Hive to Spark with Spark 1.1 the following error is displayed in the Spark execution log: [20260906, 20391714]

```
HIVE ValueError: not supported type: <type 'bytearray'>
```

Hive Complex Datatypes Not Supported by LKM Spark to Hive

The following Hive complex datatypes are not supported:

- MapType
- UnionType
- ArrayType

Using these complex datatypes causes an unknown issue. [20141453, 20391743]

Spark Execution Supports only YARN Deployment

It is recommended to run Spark applications on YARN, as ODI supports only yarn-client and yarn-cluster mode executions along with a runtime check. Please switch to YARN execution, if you have been using other Spark execution modes. [24846472]

If switching to YARN execution mode is not possible or you wish to continue with unsupported Spark execution modes, the following DataServer property must be added to the Spark DataServer:

```
odi.spark.enableUnsupportedSparkModes = true
```

Also, please note that no Support Requests can be raised regarding the unsupported Spark execution modes.

Spark-Cassandra: Permission Errors in YARN-client mode

When connecting to Cassandra sources or targets using "LKM SQL to Spark" or "LKM Spark to SQL", the JDBC driver parameter or property `SchemaMap` must not be used in YARN-client mode. Due to conflicting owners in the YARN-client execution model, the WebLogic JDBC Driver for Cassandra will encounter file permission problems and displays errors messages such as:

```
py4j.protocol.Py4JJavaErrorpy4j.protocol.Py4JJavaError: An error
occurred while calling o140.jdbc.
: java.sql.SQLException: [FMWGEN][Cassandra JDBC Driver]
[Cassandra]Unable to create local database file: $$ The cause: $$
```

This error is often caused by the driver not having write access to the target directory. [24928801]

Known Datatype Issues using Spark 1.6

Due to limitations in Spark 1.6, the following Oracle datatypes cannot be handled using `LKM SQL to Spark` or `LKM Spark to SQL` [25047069] :

- Use of `FLOAT` and `REAL` will cause the following `ValueError`:

  ```
  (ValueError(u'Could not parse datatype: decimal(38,-127)',)
  ```

- Use of extended `TIMESTAMP` and `INTERVAL` datatypes such as: `TIMESTAMP WITH TIME ZONE, TIMESTAMP WITH LOCAL TIME ZONE, INTERVAL DAY TO SECOND, INTERVAL YEAR TO MONTH` will cause the following errors:

  ```
  py4j.protocol.Py4JJavaErrorpy4j.protocol.Py4JJavaError:
  An error occurred while calling o43.jdbc.:
  java.sql.SQLException: Unsupported type -101
  ```

Unable to Store Alias Error in Pig

If the mapping execution in Pig fails and the `Unable to store alias` error is displayed, the `pig.optimizer.rules.disabled` property for the Pig server should be set to `FilterLogicExpressionSimplifier`. [20520865]

KMs Replaced During Repository Upgrade

By default, all loaded KMs in the repository are replaced during repository upgrade, irrespective of whether they are modified or not. Do not upgrade the KMs during repository upgrade.

The following are the workarounds to upgrade the KMs:

- If you have SAP KMs, you must not upgrade the KMs during repository upgrade. The new SAP KMs require new ODI SAP components. Using new SAP KMs with old ODI SAP components causes any SAP mappings to fail.

  As a workaround, uncheck **Replace KMs with Mandatory Updates** when upgrading the ODI repositories. To upgrade SAP KMs, follow the upgrade instructions given in the *Application Adapters Guide for Oracle Data Integrator* for the respective ODI SAP adapter.

- If you have any custom KMs, the customizations are lost if you upgrade the KMs during repository upgrade.

  As a workaround, uncheck Replace **KMs with Mandatory Updates** when upgrading the ODI repositories. After you upgrade the repositories, manually replace only those KMs that you want to upgrade.

Erroneously Published SDK API Classes Removed from the 12*c* Javadocs

Due to a bug in Javadoc generation, 41 internal classes were erroneously published in the 12.1.2, 12.1.3, and 12.1.3.0.1 public SDK API Javadocs. These classes were intended for internal use and have been removed from the 12*c* public SDK APIs. The classes removed from the 12*c* Javadocs are listed below and if you are using any of these classes in your program, correct your program and remove their usage: [21700125]

- `AdapterException`
- `ComponentDefinitionParser`
- `ComponentRegistryHelper`
- `ExecutionUnit.GenerationType`
- `FCONamedObject`
- `FCOPropertyOwner`
- `FCORoot`
- `IMapReferenceOwner`
- `IMappingObject.SyncState`

- IModelObjectChange

- IModelObjectChange.ChangeType

- IObjectAdapterFactory

- LocationAdapterBase

- MapAttribute.ConnectionTypeInfo

- MapAttribute.ConnectionTypeSelector

- MapAttribute.DefaultConnectionTypeSelector

- MapComponent

- MapComponentOwner

- MapComponentType.uidef

- MapPhysicalDesign.ContextualComponentTreeNode

- MapPhysicalDesign.ExecutionUnitConfiguration

- MapPhysicalDesign.ExecutionUnitGraph

- MapPhysicalDesign.ExecutionUnitGraphNode

- MapPhysicalDesign.MapPhysicalDesignConfig

- MapPhysicalDesign.NodeConfiguration

- MapPhysicalDesign.PushDirection

- MapPhysicalNode.RMCStackPropertyManager

- MapRootContainer

- MappingGenericTechnology.MappingLanguage

- MappingGenericTechnology.MappingLanguageElement

- MappingGenericTechnology.MappingSubLanguage

- NamedObject

- OdiComponent

- OdiInterface.IPersistenceComparable

- PropertyOwner

- ResourceLoader

- ResourceLoader.ResourceCandidate

- ReusableMappingComponent.RMCConnectorPointDelegate

- Root

- RootIssue.TextPos

- TargetLoadOrderException

CKM Fails with XML and Complex Files When Database is Set to External

Flow control steps (CKM) fail with `ORA-00904: "NOW": invalid identifier` errors when CKM is used with XML and Complex Files. Mapping is defined to load data into a Complex File target Datastore, while the Complex File Data Server is defined to use an external database.

You get the following error message:

```
ODI-1228: Task insert PK errors-Copy of CKM SQL- fails on the target
connection COMPLEX_ROTA_OUT_ISL.
Caused By: java.sql.SQLSyntaxErrorException: ORA-00904: "NOW": invalid
identifier

at oracle.jdbc.driver.T4CTTIoer11.processError(T4CTTIoer11.java:495)
at oracle.jdbc.driver.T4CTTIoer11.processError(T4CTTIoer11.java:447)
```

The problem is due to ODI not being able to pick the right DATE function when the flow or static control is run on a XML (or Complex File) Data Server defined to use an external database.

One of the main reason behind this limitation is, the CKM code being executed on the external database technology (for example Oracle), should use DATE function specific to that technology. Instead it gets the information from the definition of the XML or Complex File technology and the resulting function does not apply to the external database technology. As a result ODI is not able to run static or flow control (CKM) on technologies such as XML and Complex Files when the Data Server is set to use an external database.

So, the workaround is to edit the `CKM Insert PK errors, Insert AK errors, Insert FK errors` and `Insert CK errors tasks' target` commands by replacing `OdiRef.getInfo("DEST_DATE_FCT")` with the date function of the used external database technology. For example — `sysdate`, if you are using Oracle external database. [28641256]

Flexfields Tab of KM Editor May Not Display Newly Created KMs

When you re-open the KM Editor and go to Flexfields tab, the newly created flexfields may not be displayed, though they are already saved. Refreshing the Tree on save when multiple editors are open may result in performance issues. To avoid performance issues, refresh the parent of the KM before you re-open it.[28561299]

Post-install Patch Information for Oracle Data Integrator 14c (14.1.2.0.0)

You can find out more information on the post-installation patches for Oracle Data Integrator 14c (14.1.2.0.0).
After installing Oracle Data Integrator 14*c* (14.1.2.0.0), perform the following steps:

1. Make a backup of your ODI repository schema.

2. Upgrade all ODI repositories associated with the installation using the Upgrade Assistant. See your Upgrade documentation for detailed upgrade instructions.

> **✎ Note:**
>
> Once the ODI repository is upgraded it cannot be reverted back even if you remove the patch. So make sure you make a proper backup of your existing ODI repository so that it can be restored if you remove this patch in the future for any reason.

3. For setting up new domains with this patch, follow the instructions in *Installing and Configuring Oracle Data Integrator*.

4. Clearing of the JDev cache is required for all installations where the ODI Client is to be launched:

   - For UNIX platforms:

     Locate `system14.1.2.0.0` in your **Home** directory and remove it.

     For example: `rm -rf $HOME/.odi/system12.2.1.0.0`

   - For Windows platforms:

     Locate `system14.1.2.0.0` in your **Home** directory and remove it.

     For example: `delete C:\Users\<username>\AppData\Roaming\odi`

5. Start ODI Studio.

6. Depending upon the installation type, start Standalone Agent or all servers (AdminServer and all Managed server(s)).

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Oracle Fusion Middleware Release Notes for Oracle Data Integrator, 14*c* (14.1.2.0.0)
G10244-03