Oracle® Data Integrator

Tools Reference Guide

12*c* (12.1.3.0.1)

E51097-03

March 2015

ORACLE®

Oracle Data Integrator Tools Reference Guide, 12c (12.1.3.0.1)

E51097-03

Copyright © 2010, 2015,

 Oracle and/or its affiliates. All rights reserved.

Contents

Preface

This guide describes how to use and develop Open Tools using Oracle Data Integrator to design integration scenarios

This preface contains the following topics:

- Audience
- Documentation Accessibility
- Related Documents
- Conventions

Audience

This document is intended for Oracle Data Integrator application developers who will use Open Tools to design integration scenarios.

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.

Related Documents

For more information, see the following documents in *Oracle Data Integrator Library*.

- Release Notes for Oracle Data Integrator
- Understanding Oracle Data Integrator
- Administering Oracle Data Integrator
- Developing Integration Projects with Oracle Data Integrator
- Installing and Configuring Oracle Data Integrator

- Upgrading Oracle Data Integrator

- Application Adapters Guide for Oracle Data Integrator

- Developing Knowledge Modules with Oracle Data Integrator

- Connectivity and Knowledge Modules Guide for Oracle Data Integrator

- Migrating From Oracle Warehouse Builder to Oracle Data Integrator

- Oracle Data Integrator Tool Reference

- Data Services Java API Reference for Oracle Data Integrator

- Getting Started with SAP ABAP BW Adapter for Oracle Data Integrator

- Java API Reference for Oracle Data Integrator

- Getting Started with SAP ABAP ERP Adapter for Oracle Data Integrator

- *Oracle Data Integrator 12c Online Help*, which is available in ODI Studio through the JDeveloper Help Center when you press **F1** or from the main menu by selecting **Help**, and then **Search** or **Table of Contents**.

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1

Using Oracle Data Integrator Open Tools

This appendix provides a reference of Oracle Data Integrator (ODI) tools. It describes how to use Open Tools to develop new scenarios in Oracle Data Integrator.

This appendix includes the following sections:

- Using Oracle Data Integrator Tools
- Using Open Tools
- Developing Open Tools

Using Oracle Data Integrator Tools

Oracle Data Integrator tools (also called Oracle Data Integrator commands) are commands provided for performing specific tasks at runtime. These tasks can be as simple as waiting for a certain time or producing a sound, or as sophisticated as executing Ant scripts or reading email from a server.

Oracle Data Integrator tools are used in Packages, Procedure Commands, Knowledge Modules Commands, or directly from a command line.

> **Note:** Previous versions of Oracle Data Integrator supported calling built-in tools from Jython or Java scripts using their internal Java classes (such as `SnpsSendMail` and `SendMail`). This approach is no longer supported.

> **Note:** Carriage returns in commands are not permitted.

Using a Tool in a Package

Adding and using an Oracle Data Integrator tool in a Package is described in "Adding Oracle Data Integrator Tool Steps" in *Developing Integration Projects with Oracle Data Integrator*.

You can sequence the tool steps within the package and organize them according to their success and failure. For more information about sequencing, see "Arranging the Steps Layout" and "Defining the Sequence of Steps" in *Developing Integration Projects with Oracle Data Integrator*.

You can use variable values, sequences, or Oracle Data Integrator substitution method calls directly in tool parameters.

Using a Tool in a Knowledge Module or Procedure Command

Using an Oracle Data Integrator tool in a Knowledge Module or Procedure is described in "Working with Procedures" in *Developing Integration Projects with Oracle Data Integrator*.

You can use variable values, sequences, Oracle Data Integrator substitution method calls, or the results from a SELECT statement directly in tool parameters.

Using a Tool From a Command Line

Command line scripts for Oracle Data Integrator tools are run from the *DOMAIN_HOME*/bin directory. To run a tool from a command line, you must first create an ODI Physical Agent instance in the ODI Topology and configure an ODI Standalone Agent instance in a Domain. For more information about performing these tasks, see *Installing and Configuring Oracle Data Integrator*.

When you run a tool from a command line, you must specify the -INSTANCE parameter, where <agent_name> is the name of the physical agent you configured (for example, OracleDIAgent1).

To use an Oracle Data Integrator tool from a command line:

1. Launch the command shell for your environment (Windows or UNIX).

2. Navigate to the *DOMAIN_HOME*/bin directory.

3. Launch the startcmd.cmd (Windows) or startcmd.sh (UNIX) command and run an Oracle Data Integrator tool with the following syntax:

```
startcmd.<cmd|sh> -INSTANCE=<agent_name> <command_name> [<command_parameters>]*
```

Command names and command parameters are case-sensitive.

Important Notes

Note the following:

- On Windows platforms, command arguments that contain equal (=) signs or spaces must be surrounded with double quotation marks. This differs from the UNIX command call. For example:

```
startcmd.cmd OdiSleep "-INSTANCE=OracleDIAgent1" "-DELAY=5000"
./startcmd.sh OdiSleep -INSTANCE=OracleDIAgent1 -DELAY=5000
```

- The following tools do not support direct invocation through a command line:

  - OdiRetrieveJournalData

  - OdiRefreshJournalCount

Using Open Tools

The Open Tools feature provides an extensible platform for developing custom third-party tools that you can use in Packages and Procedures. As with the standard tools delivered with Oracle Data Integrator, Open Tools can interact with the operating system and manipulate data.

Open Tools are written in Java. Writing your own Open Tools is described in "Developing Open Tools" on page 1-4.

Open Tools are delivered as a Java package (.zip or .jar) that contains several files:

- A compiled Java `.class` file
- Other resources, such as icon files

Installing and Declaring an Open Tool

Before you can use an Open Tool, you must install and add it.

Installing an Open Tool

To install an Open Tool, you must add the Open Tool JAR into the classpath or the component using the tool.

Open Tool JARs must be added to the *DOMAIN_HOME*/`lib` directory. Drivers are added to the same location.

To deploy an Open Tool JAR with a Java EE agent, generate a server template for this agent. The Open Tool displays in the **Libraries and Drivers** list in the Template Generation Wizard. See "Creating a Server Template for the Java EE Agent" in *Administering Oracle Data Integrator* for more information.

> **Note:** This operation must be performed for each Oracle Data Integrator Studio from which the tool is being used, and for each agent that will run sessions using this tool.

Declaring a New Open Tool

This operation declares an Open Tool in a master repository and enables the tool to display in Oracle Data Integrator Studio.

To declare an Open Tool a JAR must be added in `<ide.user.dir>/oracledi/userlib`

To declare a new tool:

1. In Oracle Data Integrator Studio, select the **ODI** menu and then select **Add Remove/Open Tools**. The **Add Open Tools** dialog displays.

2. Enter the name of the class in the **Open Tool Class Name** field.

or:

1. Click **Find in the ClassPath**, then browse to the name of the Open Tool's Java class. To search for the class by name, enter part of the name in the field at the top.

2. Click **OK**.

   Note that all classes currently available to Oracle Data Integrator are displayed, including those that are not Open Tools. You must know the name of your class in order to add it.

3. Click **Add Open Tool**.

4. Select the line containing your Open Tool.

   - If the tool was correctly found on the classpath, the supplied icons, and the tool's syntax, description, provider, and version number are displayed.

   - If the tool was not found, an error message is displayed. Change the classpath, or move the Open Tool to the correct directory.

   > **Note:** This operation to declare a new Open Tool must be performed only once for a given master repository.

> **Note:** An Open Tool name cannot start with `Snp` or `Odi`. An Open Tool with a name that starts with these strings is ignored.

Using Open Tools in a Package or Procedure

You can use Open Tools in a Package or Procedure, similar to the tools provided with Oracle Data Integrator.

Developing Open Tools

An Open Tool is a Java package that contains a compiled Java class that implements the interface `oracle.odi.sdk.opentools.IOpenTool`. For a complete description of classes and methods, see the *Oracle Data Integrator Open Tools Java API Reference* (JavaDoc).

An Open Tool package typically should also contain two icons, which are used to represent the Open Tool in the Oracle Data Integrator graphical interface.

Classes

The following table lists and describes Open Tool classes and interfaces.

Class or Interface	Description
IOpenTool	Interface that every Open Tool must implement.
OpenToolAbstract	Abstraction of the interface with some helper methods. Preferably extend this class rather than implementing the interface directly.
IOpenToolParameter	Interface that parameters used by Open Tools must implement. In most cases, `OpenToolParameter` should be used rather than implementing this interface.
OpenToolParameter	Complete implementation of `IOpenToolParameter`. Each `OpenToolParameter` holds one parameter.
OpenToolsExecutionException	Exception class that should be thrown if necessary by Open Tool methods.
SimpleOpenToolExample	A simple example of an Open Tool, which can be used as a starting point.

Developing a New Open Tool

The following steps describe the development of a basic Open Tool, SimpleMessageBox. The source code for this class is available in the `demo/plugins/src` directory.

1. Define the syntax. In this example, the Open Tool is called as follows:

   ```
   SimpleMessageBox "-TEXT=<text message>" "-TITLE=<window title>"
   ```

2. Create 16x16 and 32x32 icons (usually in `.gif` format).

3. Create and implement the class. See "Implementing the Class" on page 1-5.

4. Compile the class and create a package with the two icon files.

5. Install and declare the Open Tool as described in "Installing and Declaring an Open Tool" on page 1-3.

Implementing the Class

Implementing the class consists of the following steps:

1. Declaration
2. Importing Packages
3. Defining the Parameters
4. Implementing Informational Functions
5. Execution

Declaration Before you declare the class, you must name the package.

Naming the Package

Put the class in a package named appropriately. The package name is used to identify the Open Tool when installing it.

```
package com.myCompany.OpenTools;
```

Declaring the Class

There are two basic approaches to developing an Open Tool:

- Extend an existing class that you want to convert into an Open Tool. In this case, simply implement the interface `IOpenTool` directly on the existing class.

- Develop a new class. In this case, it is easiest to extend the abstract class `OpenToolAbstract`. This abstract class also contains additional helper methods for working with parameters.

  ```
  public class SimpleMessageBox extends OpenToolAbstract {
  ```

Importing Packages Almost every Open Tool must import the following Open Tool SDK packages:

```
import oracle.odi.sdk.opentools.IOpenTool; /* All Open Tool classes need these
three classes */

import oracle.odi.sdk.opentools.IOpenToolParameter;

import oracle.odi.sdk.opentools.OpenToolExecutionException;

import oracle.odi.sdk.opentools.OpenToolAbstract; /* The abstract extended for the
Open Tool */

import oracle.odi.sdk.opentools.OpenToolParameter; /* The class used for
parameters */
```

In this particular example, a package to create the message box is also needed:

```
import javax.swing.JOptionPane; /* Needed for the message box used in this example
*/
```

Defining the Parameters Add a property to store the `OpenToolParameter` objects. This is used to both define them for the syntax, and to retrieve the values of the parameters from the eventual user. It is easiest to define the parameters of the Open Tool with a static array as follows. This array should be private, as it will be accessed through an accessor function.

```
private static final IOpenToolParameter[] mParameters = new IOpenToolParameter[]
```

```
{
    new OpenToolParameter("-TEXT", "Message text", "Text to show in the messagebox
(Mandatory).", true),
    new OpenToolParameter("-TITLE", "Messagebox title", "Title of the
messagebox.", false)
};
```

The four parameters passed to the `OpenToolParameter()` constructor are as follows:

1. The code of the parameter, including the initial hyphen. This code must correspond to the syntax returned by `getSyntax()`.

2. The user-friendly name, which is used if the user is using the graphical interface to set parameters.

3. A descriptive help text.

4. Whether the parameter is mandatory. This is an indication to the user.

> **Note:** Oracle Data Integrator does not enforce the mandatory flag on parameters. Your class must be able to handle any combination of parameters being provided.

You must implement the accessor function `getParameters()` to retrieve the parameters:

```
public IOpenToolParameter[] getParameters()
{
    return mParameters;
}
```

Implementing Informational Functions Implement functions to return information about your Open Tool: `getDescription()`, `getVersion()`, `getProvider()`.

```
public String getDescription() { return "This Open Tool displays a message box
when executed."; }
public String getVersion() { return "v1.0"; }
public String getProvider() { return "My Company, Inc."; }
```

The `getSyntax()` function determines the name of the Open Tool as it is displayed in the Oracle Data Integrator graphical interface, and also the initial values of the parameter. Make sure the names of the parameters here match the names of the parameters returned by `getParameters()`.

```
public String getSyntax()
{
    return "SimpleMessageBox \"-TEXT=<text message>\" \"-TITLE=<window
title>\"";
}
```

The `getIcon()` method should then return paths to two appropriately sized images. It should look something like this:

```
public String getIcon(int pIconType)
{
    switch (pIconType)
    {
        case IOpenTool.SMALL_ICON:
        return "/com/myCompany/OpenTools/images/SimpleMessageBox_16.gif";
        case IOpenTool.BIG_ICON:
```

```
                    return "/com/myCompany/OpenTools/images/SimpleMessageBox_32.gif";
                    default:
                    return "";
            }
}
```

Execution Finally, the `execute()` method, which carries out the functionality provided by the Open Tool. In this case, a message box is shown. If you are extending the `OpenToolAbstract` class, use the `getParameterValue()` method to easily retrieve the values of parameters, as they are set at runtime.

> **Note:** You must catch all exceptions and only raise an
> `OpenToolExecutionException`.

```
public void execute() throws OpenToolExecutionException
{
    try
    {
    if (getParameterValue("-TITLE") == null ||
getParameterValue("-TITLE").equals("")) /* title was not filled in by user */
    {
            JOptionPane.showMessageDialog(null, (String)
getParameterValue("-TEXT"), (String) "Message", JOptionPane.INFORMATION_MESSAGE);
        } else
        {
            JOptionPane.showMessageDialog(null, (String)
getParameterValue("-TEXT"),
                       (String) getParameterValue("-TITLE"),
                       JOptionPane.INFORMATION_MESSAGE);
        }
    }
    /* Traps any exception and throw them as OpenToolExecutionException */
    catch (IllegalArgumentException e)
    {
        throw new OpenToolExecutionException(e);
    }
}
```

Open Tools at Runtime

In general, your Open Tool class is instantiated only very briefly, and is used in the following ways.

Installation

When the user chooses to install an Open Tool, Oracle Data Integrator instantiates the class and calls the methods `getDescription()`, `getProvider()`, `getIcon()`, and `getVersion()` to retrieve information about the class.

Use in a Package

When the Open Tool is used in a package, the class is instantiated briefly to call the methods `getDescription()`, `getProvider()`, `getIcon()`, and `getVersion()`. Additionally, `getSyntax()` is called to retrieve the code name of the Open Tool and its default arguments. The method `getParameters()` is called to display the list of arguments to the user.

Execution

Each time the Open Tool is executed in a package or procedure, the class is instantiated again; it has no persistence after its execution. The `execute()` method is called just once.

> **Tip:** See also Using Open Tools and Open Tools SDK documentation (JavaDoc).

2

Oracle Data Integrator Tools

This chapter lists all the Oracle Data Integrator Tools by category and describes its commands and parameters.

This section lists Oracle Data Integrator tools by category.

Oracle Data Integrator Tools by Category

This section lists Oracle Data Integrator tools by category.

Metadata

- OdiReverseGetMetaData
- OdiReverseManageShortcut
- OdiReverseResetTable
- OdiReverseSetMetaData

Oracle Data Integrator Objects

- OdiDeleteScen
- OdiExportAllScen
- OdiExportEnvironmentInformation
- OdiExportLog
- OdiExportMaster
- OdiExportObject
- OdiExportScen
- OdiExportWork
- OdiGenerateAllScen
- OdiImportObject
- OdiImportScen

Utilities

- OdiAnt
- OdiBeep

- OdiEnterpriseDataQuality
- OdiKillAgent
- OdiOSCommand
- OdiPingAgent
- OdiPurgeLog
- OdiReinitializeSeq
- OdiRemoveTemporaryObjects
- OdiRetrieveHadoopLog
- OdiStartLoadPlan
- OdiStartOwbJob
- OdiStartScen
- OdiUpdateAgentSchedule

Internet Related Tasks

- OdiFtp
- OdiFtpGet
- OdiFtpPut
- OdiInvokeWebService
- OdiReadMail
- OdiScpGet
- OdiScpPut
- OdiSftp
- OdiSftpGet
- OdiSftpPut
- OdiSendMail

Files

- OdiFileAppend
- OdiFileCopy
- OdiFileDelete
- OdiFileMove
- OdiFileWait
- OdiMkDir
- OdiOutFile
- OdiSqlUnload
- OdiUnZip
- OdiZip

SAP

- OdiSAPALEClient and OdiSAPALEClient3
- OdiSAPALEServer and OdiSAPALEServer3

XML

- OdiXMLConcat
- OdiXMLSplit

Event Detection

- OdiFileWait
- OdiReadMail
- OdiSleep
- OdiWaitForChildSession
- OdiWaitForData
- OdiWaitForLoadPlans
- OdiWaitForLogData
- OdiWaitForTable

Changed Data Capture

- OdiManageOggProcess
- OdiRefreshJournalCount
- OdiRetrieveJournalData
- OdiWaitForData
- OdiWaitForLogData
- OdiWaitForTable

Alphabetical List of Oracle Data Integrator Tools

This section lists Oracle Data Integrator tools in alphabetical order.

- OdiAnt
- OdiBeep
- OdiDeleteScen
- OdiEnterpriseDataQuality
- OdiExportAllScen
- OdiExportEnvironmentInformation
- OdiExportLog
- OdiExportMaster
- OdiExportObject
- OdiExportScen

- OdiExportWork
- OdiFileAppend
- OdiFileCopy
- OdiFileDelete
- OdiFileMove
- OdiFileWait
- OdiFtp
- OdiFtpGet
- OdiFtpPut
- OdiGenerateAllScen
- OdiImportObject
- OdiImportScen
- OdiInvokeWebService
- OdiKillAgent
- OdiManageOggProcess
- OdiMkDir
- OdiOSCommand
- OdiOutFile
- OdiPingAgent
- OdiPurgeLog
- OdiReadMail
- OdiRefreshJournalCount
- OdiReinitializeSeq
- OdiRemoveTemporaryObjects
- OdiRetrieveHadoopLog
- OdiRetrieveJournalData
- OdiReverseGetMetaData
- OdiReverseManageShortcut
- OdiReverseResetTable
- OdiReverseSetMetaData
- OdiSAPALEClient and OdiSAPALEClient3
- OdiSAPALEServer and OdiSAPALEServer3
- OdiScpGet
- OdiScpPut
- OdiSendMail
- OdiSftp
- OdiSftpGet

- OdiSftpPut
- OdiSleep
- OdiSqlUnload
- OdiStartLoadPlan
- OdiStartOwbJob
- OdiStartScen
- OdiUnZip
- OdiUpdateAgentSchedule
- OdiWaitForChildSession
- OdiWaitForData
- OdiWaitForLoadPlans
- OdiWaitForLogData
- OdiWaitForTable
- OdiXMLConcat
- OdiXMLSplit
- OdiZip

OdiAnt

Use this command to execute an Ant buildfile.

For more details and examples of Ant buildfiles, refer to the online documentation: http://jakarta.apache.org/ant/manual/index.html

Usage

```
OdiAnt -BUILDFILE=<file> -LOGFILE=<file> [-TARGET=<target>]
[-D<property name>=<property value>]* [-PROJECTHELP] [-HELP]
[-VERSION] [-QUIET] [-VERBOSE] [-DEBUG] [-EMACS]
[-LOGGER=<classname>] [-LISTENER=<classname>] [-FIND=<file>]
```

Parameters

Parameters	Mandatory	Description
-BUILDFILE=<file>	Yes	Ant buildfile. XML file containing the Ant commands.
-LOGFILE=<file>	Yes	Use given file for logging.
-TARGET=<target>	No	Target of the build process.
-D<property name>=<property value>	No	List of properties with their values.
-PROJECTHELP	No	Displays the help on the project.
-HELP	No	Displays Ant help.
-VERSION	No	Displays Ant version.
-QUIET	No	Run in nonverbose mode.
-VERBOSE	No	Run in verbose mode.
-DEBUG	No	Prints debug information.
-EMACS	No	Displays the logging information without adornments.
-LOGGER=<classname>	No	Java class performing the logging.
-LISTENER=<classname>	No	Adds a class instance as a listener.
-FIND=<file>	No	Looks for the Ant buildfile from the root of the file system and uses it.

Examples

Download the *.html files from the directory /download/public using FTP from ftp.mycompany.com to the directory C:\temp.

Step 1: Generate the Ant buildfile:

```
OdiOutFile -FILE=c:\temp\ant_cmd.xml
<?xml version="1.0"?>
<project name="myproject" default="ftp" basedir="/">
    <target name="ftp">
        <ftp action="get" remotedir="/download/public"
        server="ftp.mycompany.com" userid="anonymous"
        password="me@mycompany.com">
            <fileset dir="c:\temp">
         <include name="**/*.html"/>
```

```
                    </fileset>
        </ftp>
    </target>
</project>
```

Step 2: Run the Ant buildfile:

```
OdiAnt -BUILDFILE=c:\temp\ant_cmd.xml -LOGFILE=c:\temp\ant_cmd.log
```

OdiBeep

Use this command to play a default beep or sound file on the machine hosting the agent.

The following file formats are supported by default:

- WAV

- AIF

- AU

> **Note:** To play other file formats, you must add the appropriate JavaSound Service Provider Interface (JavaSound SPI) to the application classpath.

Usage

```
OdiBeep [-FILE=<sound_file>]
```

Parameters

Parameters	Mandatory	Description
-FILE	No	Path and file name of sound file to play. If not specified, the default beep sound for the machine is used.

Examples

```
OdiBeep -FILE=c:\wav\alert.wav
```

OdiDeleteScen

Use this command to delete a given scenario version.

Usage

```
OdiDeleteScen -SCEN_NAME=<name> -SCEN_VERSION=<version>
```

Parameters

Parameters	Mandatory	Description
-SCEN_NAME=<name>	Yes	Name of the scenario to delete.
-SCEN_VERSION=<version>	Yes	Version of the scenario to delete.

Examples

Delete the DWH scenario in version 001.

```
OdiDeleteScen -SCEN_NAME=DWH -SCEN_VERSION=001
```

OdiEnterpriseDataQuality

Use this command to invoke an Oracle Enterprise Data Quality (Datanomic) job.

> **Note:** The OdiEnterpriseDataQuality tool supports Oracle Enterprise Data Quality version 8.1.6 and later.

Usage

```
OdiEnterpriseDataQuality "-JOB_NAME=<EDQ job name>"
"-PROJECT_NAME=<EDQ project name>" "-HOST=<EDQ server host name>"
"-PORT=<EDQ server JMX port>" "-USER=<EDQ user>"
"-PASSWORD=<EDQ user's password>" "-SYNCHRONOUS=<yes|no>" "-DOMAIN=-<EDQ_DOMAIN>"
```

Parameters

Parameters	Mandatory	Description
-JOB_NAME=<EDQ job name>	Yes	Name of the Enterprise Data Quality job.
-PROJECT_NAME=<EDQ project name>	Yes	Name of the Enterprise Data Quality project.
-HOST=<EDQ server host name>	Yes	Host name of the Enterprise Data Quality server. Example: localhost
-PORT=<EDQ server JMX port>	Yes	JMX port of the Enterprise Data Quality server. Example: 9005
-USER=<EDQ user>	Yes	User name of the Enterprise Data Quality server user. Example: dnadmin
-PASSWORD=<EDQ user's password>	Yes	Password of the Enterprise Data Quality user.
-SYNCHRONOUS=<yes|no>	No	If set to Yes (default), the tool waits for the quality process to complete before returning, with possible error code. If set to No, the tool ends immediately with success and does not wait for the quality process to complete.
-DOMAIN=<EDQ_DOMAIN>	No	Name of the MBean domain. The default value is dndirector.

Examples

Execute the Enterprise Data Quality job CLEANSE_CUSTOMERS located in the project CUSTOMERS.

```
EnterpriseDataQuality "-JOB_NAME=CLEANSE_CUSTOMERS" "-PROJECT_NAME=CUSTOMERS"
"-HOST=machine.oracle.com" "-PORT=9005" "-USER=odi" "-PASSWORD=odi"
"DOMAIN=dndirector"
```

OdiExportAllScen

Use this command to export a group of scenarios from the connected repository.

The export files are named SCEN_<scenario name><scenario version>.xml. This command reproduces the behavior of the export feature available in Designer Navigator and Operator Navigator.

Usage

```
OdiExportAllScen -TODIR=<directory> [-FORCE_OVERWRITE=<yes|no>]
[-FROM_PROJECT=<project_id>] [-FROM_FOLDER=<folder_id>]
[-FROM_PACKAGE=<package_id>] [-RECURSIVE_EXPORT=<yes|no>]
[-XML_VERSION=<1.0>] [-XML_CHARSET=<charset>]
[-JAVA_CHARSET=<charset>] [-EXPORT_KEY=<key>] [-EXPORT_MAPPING=<yes|no>]
[-EXPORT_PACK=<yes|no>] [-EXPORT_POP=<yes|no>]
[-EXPORT_TRT=<yes|no>] [-EXPORT_VAR=<yes|no>]
[EXPORT_WITHOUT_CIPHER_DATA=<yes|no>]
```

Parameters

Parameters	Mandatory	Description
-TODIR=<directory>	Yes	Directory into which the export files are created.
-FORCE_OVERWRITE=<yes|no>	No	If set to Yes, existing export files are overwritten without warning. The default value is No.
-FROM_PROJECT=<project_id>	No	ID of the project containing the scenarios to export. This value is the **Global ID** that displays in the **Version** tab of the project window in Studio. If this parameter is not set, scenarios from all projects are taken into account for the export.
-FROM_FOLDER=<folder_id>	No	ID of the folder containing the scenarios to export. This value is the **Global ID** that displays in the **Version** tab of the folder window in Studio. If this parameter is not set, scenarios from all folders are taken into account for the export.
-FROM_PACKAGE=<package_id>	No	ID of the source package of the scenarios to export. This value is the **Global ID** that displays in the **Version** tab of the package window in Studio. If this parameter is not set, scenarios from all components are taken into account for the export.
-RECURSIVE_EXPORT=<yes|no>	No	If set to Yes (default), all child objects (schedules) are exported with the scenarios.
-XML_VERSION=<1.0>	No	Sets the XML version shown in the XML header. The default value is 1.0.
-XML_CHARSET=<charset>	No	Encoding specified in the XML export file in the tag <?xml version="1.0" encoding="ISO-8859-1"?>. The default value is ISO-8859-1. For the list of supported encodings, see: http://java.sun.com/j2se/1.4.2/docs/guide/intl/encoding.doc.html

Parameters	Mandatory	Description
-JAVA_CHARSET=<charset>	No	Target file encoding. The default value is ISO8859_1. For the list of supported encodings, see: http://java.sun.com/j2se/1.4.2/docs/guide/intl/encoding.doc.html
-EXPORT_KEY=<key>	No[1]	Specifies a cryptographic private key used to encrypt sensitive cipher data. You must specify this key again when importing the exported object in order to import the cipher data.
-EXPORT_MAPPING=<yes|no>	No	Indicates if the mapping scenarios should be exported. The default value is No.
-EXPORT_PACK=<yes|no>	No	Indicates if the scenarios attached to packages should be exported. The default value is Yes.
-EXPORT_POP=<yes|no>	No	Indicates if the scenarios attached to mappings should be exported. The default value is No.
-EXPORT_TRT=<yes|no>	No	Indicates if the scenarios attached to procedures should be exported. The default value is No.
-EXPORT_VAR=<yes|no>	No	Indicates if the scenarios attached to variables should be exported. The default value is No.
-EXPORT_WITHOUT_CIPHER_DATA=<yes|no>	No[2]	When set to Yes, specifies that sensitive (cipher) values should be set to null in the object when it is exported. When set to No or when this parameter is omitted, you must include the -EXPORT_KEY parameter and specify a valid key. The default value is No.

[1] If the -EXPORT_KEY parameter is not specified, the -EXPORT_WITHOUT_CIPHER_DATA parameter must be specified, and must be set to Yes.

[2] If -EXPORT_WITHOUT_CIPHER_DATA is not specified, or if it is specified and set to No, you must specify the -EXPORT_KEY parameter with a valid key value.

Examples

Export all scenarios from the DW01 project of **Global ID** 2edb524d-eb17-42ea-8aff-399ea9b13bf3 into the /temp/ directory, with all dependent objects, using the key examplekey1 to encrypt sensitive data.

```
OdiExportAllScen -FROM_PROJECT=2edb524d-eb17-42ea-8aff-399ea9b13bf3 -TODIR=/temp/
-RECURSIVE_EXPORT=yes -EXPORT_KEY=examplekey1
```

OdiExportEnvironmentInformation

Use this command to export the details of the technical environment into a comma separated (.csv) file into the directory of your choice. This information is required for maintenance or support purposes.

Usage

```
OdiExportEnvironmentInformation -TODIR=<toDir> -FILE_NAME=<FileName>
[-CHARSET=<charset>] [-SNP_INFO_REC_CODE=<row_code>]
[-MASTER_REC_CODE=<row_code>] [-WORK_REC_CODE=<row_code>]
[-AGENT_REC_CODE=<row_code>] [-TECHNO_REC_CODE=<row_code>]
[-RECORD_SEPARATOR_HEXA=<rec_sep>] [-FIELD_SEPARATOR_HEXA=<field_sep]
[-TEXT_SEPARATOR=<text_sep>]
```

Parameter

Parameters	Mandatory	Description
-TODIR=<toDir>	Yes	Target directory for the export.
-FILE_NAME=<FileName>	Yes	Name of the CSV export file. The default value is snps_tech_inf.csv.
-CHARSET=<charset>	No	Character set of the export file.
-SNP_INFO_REC_CODE=<row_code>	No	Code used to identify rows that describe the current version of Oracle Data Integrator and the current user. This code is used in the first field of the record. The default value is SUNOPSIS.
-MASTER_REC_CODE=<row_code>	No	Code for rows containing information about the master repository. The default value is MASTER.
-WORK_REC_CODE=<row_code>	No	Code for rows containing information about the work repository. The default value is WORK.
-AGENT_REC_CODE=<row_code>	No	Code for rows containing information about the various agents that are running. The default value is AGENT.
-TECHNO_REC_CODE=<row_code>	No	Code for rows containing information about the data servers, their versions, and so on. The default value is TECHNO.
-RECORD_SEPARATOR_HEXA=<rec_sep>	No	One or several characters in hexadecimal code separating lines (or records) in the file. The default value is O0D0A.
-FIELD_SEPARATOR_HEXA=<field_sep>	No	One or several characters in hexadecimal code separating the fields in a record. The default value is 2C.
-TEXT_SEPARATOR=<text_sep>	No	Character in hexadecimal code delimiting a STRING field. The default value is 22.

Examples

Export the details of the technical environment into the `/temp/snps_tech_inf.csv` export file.

```
OdiExportEnvironmentInformation "-TODIR=/temp/"
"-FILE_NAME=snps_tech_inf.csv" "-CHARSET=ISO8859_1"
"-SNP_INFO_REC_CODE=SUNOPSIS" "-MASTER_REC_CODE=MASTER"
"-WORK_REC_CODE=WORK" "-AGENT_REC_CODE=AGENT"
"-TECHNO_REC_CODE=TECHNO" "-RECORD_SEPARATOR_HEXA=0D0A"
"-FIELD_SEPARATOR_HEXA=2C" "-TEXT_SEPARATOR_HEXA=22"
```

OdiExportLog

Use this command to export the execution log into a ZIP export file.

Usage

```
OdiExportLog -TODIR=<toDir> [-EXPORT_TYPE=<logsToExport>] [-EXPORT_KEY=<key>]
[-ZIPFILE_NAME=<zipFileName>] [-XML_CHARSET=<charset>]
[-JAVA_CHARSET=<charset>] [-FROMDATE=<from_date>] [-TODATE=<to_date>]
[-AGENT=<agent>] [-CONTEXT=<context>] [-STATUS=<status>] [-USER_FILTER=<user>]
[-NAME=<sessionOrLoadPlanName>] [EXPORT_WITHOUT_CIPHER_DATA=<yes|no>]
```

Parameters

Parameters	Mandatory	Description
-EXPORT_TYPE=<logsToExport>	No	Export the log of: ■ `LOAD_PLAN_RUN`: All Load Plan runs that match the export criteria are exported, including all sessions launched by the Load Plan runs along the child session's hierarchy. ■ `SESSION`: All session logs that match the export filter criteria are exported. All Load Plan sessions will be excluded when exporting the session logs. ■ `ALL`: All Load Plan runs and session logs that match the filter criteria are exported.
-EXPORT_KEY=<key>	No[1]	Specifies a cryptographic private key used to encrypt sensitive cipher data. You must specify this key again when importing the exported object in order to import the cipher data.
-TODIR=<toDir>	Yes	Target directory for the export.
-ZIPFILE_NAME=<zipFileName>	No	Name of the compressed file.
-XML_CHARSET=<charset>	No	XML version specified in the export file. Parameter `xml version` in the XML file header. `<?xml version="1.0" encoding="ISO-8859-1"?>`. The default value is `ISO-8859-1`. For the list of supported encodings, see: http://java.sun.com/j2se/1.4.2/docs/guide/intl/encoding.doc.html
-JAVA_CHARSET=<charset>	No	Result file Java character encoding. The default value is `ISO8859_1`. For the list of supported encodings, see: http://java.sun.com/j2se/1.4.2/docs/guide/intl/encoding.doc.html
-FROMDATE=<from_date>	No	Beginning date for the export, using the format yyyy/MM/dd hh:mm:ss. All sessions from this date are exported.
-TODATE=<to_date>	No	End date for the export, using the format yyyy/MM/dd hh:mm:ss. All sessions to this date are exported.

Parameters	Mandatory	Description
`-AGENT=<agent>`	No	Exports only sessions executed by the agent `<agent>`.
`-CONTEXT=<context>`	No	Exports only sessions executed in the context code `<context>`.
`-STATUS=<status>`	No	Exports only sessions in the specified state. Possible states are Done, Error, Queued, Running, Waiting, and Warning.
`-USER_FILTER=<user>`	No	Exports only sessions launched by `<user>`.
`-NAME=<sessionOrLoadPlanName>`	No	Name of the session or Load Plan to be exported.
`-EXPORT_WITHOUT_CIPHER_ DATA=<yes|no>`	No[2]	When set to Yes, specifies that sensitive (cipher) values should be set to null in the object when it is exported. When set to No or when this parameter is omitted, you must include the `-EXPORT_KEY` parameter and specify a valid key. The default value is No.

[1] If the `-EXPORT_KEY` parameter is not specified, the `-EXPORT_WITHOUT_CIPHER_DATA` parameter must be specified, and must be set to Yes.

[2] If `-EXPORT_WITHOUT_CIPHER_DATA` is not specified, or if it is specified and set to No, you must specify the `-EXPORT_KEY` parameter with a valid key value.

Examples

Export and compress the log into the `/temp/log2.zip` export file.

```
OdiExportLog "-EXPORT_TYPE=ALL" "-EXPORT_KEY=examplekey1"
"-TODIR=/temp/" "-ZIPFILE_NAME=log2.zip" "-XML_CHARSET=ISO-8859-1"
"-JAVA_CHARSET=ISO8859_1"
```

OdiExportMaster

Use this command to export the master repository to a directory or ZIP file. The versions and/or solutions stored in the master repository are optionally exported.

Usage

```
OdiExportMaster -TODIR=<toDir> [-ZIPFILE_NAME=<zipFileName>]
[-EXPORT_KEY=<key>] [-EXPORT_SOLUTIONS=<yes|no>] [-EXPORT_VERSIONS=<yes|no>]
[-XML_CHARSET=<charset>] [-JAVA_CHARSET=<charset>]
[EXPORT_WITHOUT_CIPHER_DATA=<yes|no>]
```

Parameters

Parameters	Mandatory	Description
-TODIR=<toDir>	Yes	Target directory for the export.
-ZIPFILE_NAME=<zipFileName>	No	Name of the compressed file.
-EXPORT_KEY=<key>	No[1]	Specifies a cryptographic private key used to encrypt sensitive cipher data. You must specify this key again when importing the exported object in order to import the cipher data.
-EXPORT_SOLUTIONS=<yes|no>	No	Exports all solutions that are stored in the repository. The default value is No.
-EXPORT_VERSIONS=<yes|no>	No	Exports all versions of objects that are stored in the repository. The default value is No.
-XML_CHARSET=<charset>	No	XML version specified in the export file. Parameter xml version in the XML file header. `<?xml version="1.0" encoding="ISO-8859-1"?>`. The default value is ISO-8859-1. For the list of supported encodings, see: http://java.sun.com/j2se/1.4.2/docs/guide/intl/encoding.doc.html
-JAVA_CHARSET=<charset>	No	Result file Java character encoding. The default value is ISO8859_1. For the list of supported encodings, see: http://java.sun.com/j2se/1.4.2/docs/guide/intl/encoding.doc.html
-EXPORT_WITHOUT_CIPHER_DATA=<yes|no>	No[2]	When set to Yes, specifies that sensitive (cipher) values should be set to null in the object when it is exported. When set to No or when this parameter is omitted, you must include the -EXPORT_KEY parameter and specify a valid key. The default value is No.

[1] If the -EXPORT_KEY parameter is not specified, the -EXPORT_WITHOUT_CIPHER_DATA parameter must be specified, and must be set to Yes.

[2] If -EXPORT_WITHOUT_CIPHER_DATA is not specified, or if it is specified and set to No, you must specify the -EXPORT_KEY parameter with a valid key value.

Examples

Export and compress the master repository into the export.zip file located in the /temp/ directory.

```
OdiExportMaster "-TODIR=/temp/" "-ZIPFILE_NAME=export.zip"
"-EXPORT_KEY=examplekey1" "-XML_CHARSET=ISO-8859-1"
"-JAVA_CHARSET=ISO8859_1" "-EXPORT_VERSIONS=YES"
```

OdiExportObject

Use this command to export an object from the current repository. This command reproduces the behavior of the export feature available in the user interface.

Usage

```
OdiExportObject -CLASS_NAME=<class_name> -I_OBJECT=<object_id>
[-EXPORT_KEY=<key>] [-EXPORT_DIR=<directory>]
[-EXPORT_NAME=<export_name>|-FILE_NAME=<file_name>] [-FORCE_OVERWRITE=<yes|no>]
[-RECURSIVE_EXPORT=<yes|no>] [-XML_VERSION=<1.0>] [-XML_CHARSET=<charset>]
[-JAVA_CHARSET=<charset>] [EXPORT_WITHOUT_CIPHER_DATA=<yes|no>]
```

Parameters

Parameters	Mandatory	Description
-CLASS_NAME=<class_name>	Yes	Class of the object to export (see the following list of classes).
-I_OBJECT=<object_id>	Yes	Object identifier. This value is the **Global ID** that displays in the **Version** tab of the object edit window.
-EXPORT_KEY=<key>	No[1]	Specifies a cryptographic private key used to encrypt sensitive cipher data. You must specify this key again when importing the exported object in order to import the cipher data.
-FILE_NAME=<file_name>	No	Export file name. Absolute path or relative path from EXPORT_DIR.
		This file name may or may not comply with the Oracle Data Integrator standard export file prefix and suffix. To comply with these standards, use the -EXPORT_NAME parameter instead. This parameter cannot be used if -EXPORT_NAME is set.
-EXPORT_DIR=<directory>	No	Directory where the object will be exported. The export file created in this directory is named based on the -FILE_NAME and -EXPORT_NAME parameters.
		If -FILE_NAME or -EXPORT_NAME are not specified, the export file is automatically named <object_prefix>_<object_name>.xml. For example, a project named Datawarehouse would be exported to PRJ_Datawarehouse.xml.
-EXPORT_NAME=<export_name>	No	Export name. Use this parameter to generate an export file named <object_prefix>_<export_name>.xml. This parameter cannot be used with -FILE_NAME.
-FORCE_OVERWRITE=<yes|no>	No	If set to Yes, an existing export file with the same name is forcibly overwritten. The default value is No.
-RECURSIVE_EXPORT=<yes|no>	No	If set to Yes (default), all child objects are exported with the current object. For example, if exporting a project, all folders, KMs, and so on in this project are exported into the project export file.
-XML_VERSION=<1.0>	No	Sets the XML version that appears in the XML header. The default value is 1.0.

Parameters	Mandatory	Description
-XML_CHARSET=<charset>	No	Encoding specified in the XML file, in the tag `<?xml version="1.0" encoding="ISO-8859-1"?>`. The default value is `ISO-8859-1`. For the list of supported encodings, see: http://java.sun.com/j2se/1.4.2/docs/guide/intl/encoding.doc.html
-JAVA_CHARSET=<charset>	No	Target file encoding. The default value is `ISO8859_1`. For the list of supported encodings, see: http://java.sun.com/j2se/1.4.2/docs/guide/intl/encoding.doc.html
-EXPORT_WITHOUT_CIPHER_ DATA=<yes|no>	No[2]	When set to Yes, specifies that sensitive (cipher) values should be set to null in the object when it is exported. When set to No or when this parameter is omitted, you must include the -EXPORT_KEY parameter and specify a valid key. The default value is No.

[1] If the -EXPORT_KEY parameter is not specified, the -EXPORT_WITHOUT_CIPHER_DATA parameter must be specified, and must be set to Yes.

[2] If -EXPORT_WITHOUT_CIPHER_DATA is not specified, or if it is specified and set to No, you must specify the -EXPORT_KEY parameter with a valid key value.

List of Classes

Object	Class Name
Column	SnpCol
Condition/Filter	SnpCond
Context	SnpContext
Data Server	SnpConnect
Datastore	SnpTable
Folder	SnpFolder
Interface	SnpPop
Language	SnpLang
Loadplan	SnpLoadPlan
Mapping	SnpMapping
Model	SnpModel
Package	SnpPackage
Physical Schema	SnpPschema
Procedure or KM	SnpTrt
Procedure or KM Option	SnpUserExit
Project	SnpProject
Reference	SnpJoin
Reusable Mapping	SnpMapping
Scenario	SnpScen
Sequence	SnpSequence
Step	SnpStep

Object	Class Name
Sub-Model	SnpSubModel
Technology	SnpTechno
User Functions	SnpUfunc
Variable	SnpVar
Version of an Object	SnpVer

Examples

Export the `DW01` project of **Global ID** `2edb524d-eb17-42ea-8aff-399ea9b13bf3` into the `/temp/dw1.xml` export file, with all dependent objects.

```
OdiExportObject -CLASS_NAME=SnpProject
-I_OBJECT=2edb524d-eb17-42ea-8aff-399ea9b13bf3 -EXPORT_KEY=examplekey1
-FILE_NAME=/temp/dw1.xml -FORCE_OVERWRITE=yes
-RECURSIVE_EXPORT=yes
```

OdiExportScen

Use this command to export a scenario from the current work repository.

Usage

```
OdiExportScen -SCEN_NAME=<scenario_name> -SCEN_VERSION=<scenario_version>
[-EXPORT_KEY=<key>] [-EXPORT_DIR=<directory>]
[-FILE_NAME=<file_name>|EXPORT_NAME=<export_name>] [-FORCE_OVERWRITE=<yes|no>]
[-RECURSIVE_EXPORT=<yes|no>] [-XML_VERSION=<1.0>] [-XML_CHARSET=<encoding>]
[-JAVA_CHARSET=<encoding>] [EXPORT_WITHOUT_CIPHER_DATA=<yes|no>]
```

Parameters

Parameters	Mandatory	Description
-SCEN_NAME=<scenario_name>	Yes	Name of the scenario to be exported.
-SCEN_VERSION=<scenario_version>	Yes	Version of the scenario to be exported.
-EXPORT_KEY=<key>	No[1]	Specifies a cryptographic private key used to encrypt sensitive cipher data. You must specify this key again when importing the exported object in order to import the cipher data.
-FILE_NAME=<file_name>	Yes	Export file name. Absolute path or relative path from -EXPORT_DIR. This file name may or not comply with the Oracle Data Integrator standard export file prefix and suffix for scenarios. To comply with these standards, use the -EXPORT_NAME parameter instead. This parameter cannot be used if -EXPORT_NAME is set.
-EXPORT_DIR=<directory>	No	Directory where the scenario will be exported. The export file created in this directory is named based on the -FILE_NAME and -EXPORT_NAME parameters. If -FILE_NAME or -EXPORT_NAME are not specified, the export file is automatically named SCEN_<scenario_name><scenario_version>.xml.
-EXPORT_NAME=<export_name>	No	Export name. Use this parameter to generate an export file named SCEN_<export_name>.xml. This parameter cannot be used with -FILE_NAME.
-FORCE_OVERWRITE=<yes|no>	No	If set to Yes, overwrites the export file if it already exists. The default value is No.
-RECURSIVE_EXPORT=<yes|no>	No	Forces the export of the objects under the scenario. The default value is Yes.
-XML_VERSION=<1.0>	No	Version specified in the generated XML file, in the tag <?xml version="1.0" encoding="ISO-8859-1"?>. The default value is 1.0.

Parameters	Mandatory	Description
-XML_CHARSET=<encoding>	No	Encoding specified in the XML file, in the tag `<?xml version="1.0" encoding="ISO-8859-1"?>`. The default value is `ISO-8859-1`. For the list of supported encodings, see: http://java.sun.com/j2se/1.4.2/docs/guide/intl/encoding.doc.html
-JAVA_CHARSET=<encoding>	No	Target file encoding. The default value is `ISO8859_1`. For the list of supported encodings, see: http://java.sun.com/j2se/1.4.2/docs/guide/intl/encoding.doc.html
-EXPORT_WITHOUT_CIPHER_DATA=<yes|no>	No[2]	When set to Yes, specifies that sensitive (cipher) values should be set to null in the object when it is exported. When set to No or when this parameter is omitted, you must include the -EXPORT_KEY parameter and specify a valid key. The default value is No.

[1] If the -EXPORT_KEY parameter is not specified, the -EXPORT_WITHOUT_CIPHER_DATA parameter must be specified, and must be set to Yes.

[2] If -EXPORT_WITHOUT_CIPHER_DATA is not specified, or if it is specified and set to No, you must specify the -EXPORT_KEY parameter with a valid key value.

Examples

Export the LOAD_DWH scenario in version 1 into the /temp/load_dwh.xml export file, with all dependent objects.

```
OdiExportScen -SCEN_NAME=LOAD_DWH -SCEN_VERSION=1 -EXPORT_KEY=examplekey1
-FILE_NAME=/temp/load_dwh.xml -RECURSIVE_EXPORT=yes
```

OdiExportWork

Use this command to export the work repository to a directory or ZIP export file.

Usage

```
OdiExportWork -TODIR=<directory> [-ZIPFILE_NAME=<zipFileName>]
[-EXPORT_KEY=<key>] [-XML_CHARSET=<charset>] [-JAVA_CHARSET=<charset>]
[EXPORT_WITHOUT_CIPHER_DATA=<yes|no>]
```

Parameters

Parameters	Mandatory	Description
-TODIR=<directory>	Yes	Target directory for the export.
-ZIPFILE_NAME=<zipFileName>	No	Name of the compressed file.
-EXPORT_KEY=<key>	No[1]	Specifies a cryptographic private key used to encrypt sensitive cipher data. You must specify this key again when importing the exported object in order to import the cipher data.
-XML_CHARSET=<charset>	No	XML version specified in the export file. Parameter xml version in the XML file header. <?xml version="1.0" encoding="ISO-8859-1"?>. The default value is ISO-8859-1. For the list of supported encodings, see: http://java.sun.com/j2se/1.4.2/docs/guide/intl/encoding.doc.html
-JAVA_CHARSET=<charset>	No	Result file Java character encoding. The default value is ISO8859_1. For the list of supported encodings, see: http://java.sun.com/j2se/1.4.2/docs/guide/intl/encoding.doc.html
-EXPORT_WITHOUT_CIPHER_DATA=<yes|no>	No[2]	When set to Yes, specifies that sensitive (cipher) values should be set to null in the object when it is exported. When set to No or when this parameter is omitted, you must include the -EXPORT_KEY parameter and specify a valid key. The default value is No.

[1] If the -EXPORT_KEY parameter is not specified, the -EXPORT_WITHOUT_CIPHER_DATA parameter must be specified, and must be set to Yes.

[2] If -EXPORT_WITHOUT_CIPHER_DATA is not specified, or if it is specified and set to No, you must specify the -EXPORT_KEY parameter with a valid key value.

Examples

Export and compress the work repository into the /temp/workexport.zip export file.

```
OdiExportWork "-TODIR=/temp/" "-ZIPFILE_NAME=workexport.zip"
"-EXPORT_KEY=examplekey1"
```

OdiFileAppend

Use this command to concatenate a set of files into a single file.

Usage

```
OdiFileAppend -FILE=<file> -TOFILE=<target_file> [-OVERWRITE=<yes|no>]
[-CASESENS=<yes|no>] [-HEADER=<n>] [-KEEP_FIRST_HEADER=<yes|no>]
```

Parameters

Parameters	Mandatory	Description
-FILE=<file>	Yes	Full path of the files to concatenate. Use * to specify generic characters.
		Examples:
		/var/tmp/*.log (all files with the log extension in the folder /var/tmp)
		arch_*.lst (all files starting with arch_ and with the extension lst)
-TOFILE=<target_file>	Yes	Target file.
-OVERWRITE=<yes|no>	No	Indicates if the target file must be overwritten if it already exists. The default value is No.
-CASESENS=<yes|no>	No	Indicates if file search is case-sensitive. By default, Oracle Data Integrator searches files in uppercase (set to No).
-HEADER=<n>	No	Number of header lines to be removed from the source files before concatenation. By default, no lines are removed.
		When the -HEADER parameter is omitted, the concatenation does not require file edition, and therefore runs faster.
-KEEP_FIRST_HEADER=<yes|no>	No	Keep the header lines of the first file during the concatenation. The default value is Yes.

Examples

Concatenate the files *.log of the folder /var/tmp into the file /home/all_files.log.

```
OdiFileAppend -FILE=/var/tmp/*.log -TOFILE=/home/all_files.log
```

OdiFileCopy

Use this command to copy files or folders.

Usage

```
OdiFileCopy -DIR=<directory> -TODIR=<target_directory> [-OVERWRITE=<yes|no>]
[-RECURSE=<yes|no>] [-CASESENS=<yes|no>]

OdiFileCopy -FILE=<file> -TOFILE=<target_file>|-TODIR=<target_directory>
[-OVERWRITE=<yes|no>] [-RECURSE=<yes|no>] [-CASESENS=<yes|no>]
```

Parameters

Parameters	Mandatory	Description
-DIR=<directory>	Yes if -FILE is omitted	Directory (or folder) to copy.
-FILE=<file>	Yes if -DIR is omitted	The full path of the files to copy. Use * to specify the generic character.
		Examples:
		/var/tmp/*.log (all files with the log extension in folder /var/tmp)
		arch_*.lst (all files starting with arch_ and with the extension lst)
-TODIR=<target_directory>	Yes if -DIR is specified	Target directory for the copy.
		If a directory is copied (-DIR), this parameter indicates the name of the copied directory.
		If one or several files are copied (-FILE), this parameter indicates the destination directory.
-TOFILE=<target_file>	Yes if -TODIR is omitted	Destination file(s). This parameter cannot be used with parameter -DIR.
		This parameter contains:
		■ The name of the destination file if only one file is copied (no generic character).
		■ The mask of the new name of the destination files if several files are copied.
		Note that -TODIR and -TOFILE are exclusive parameters. If both are specified, only -TODIR is taken into account, and -TOFILE is ignored.
-TGT_LSCHEMA=<target_file>	No	The file located on a data server, based on the Logical Schema value. For example, the LSCHEMA may point to a Hadoop Data Server and the tool will access the file from that data server if the file needs to be accessed from HDFS

Parameters	Mandatory	Description
-OVERWRITE=<yes|no>	No	Indicates if the files of the folder are overwritten if they already exist. The default value is No.
-RECURSE=<yes|no>	No	Indicates if files are copied recursively when the directory contains other directories. The value No indicates that only the files within the directory are copied, not the subdirectories. The default value is Yes.
-CASESENS=<yes|no>	No	Indicates if file search is case-sensitive. By default, Oracle Data Integrator searches files in uppercase (set to No).

Examples

Copy the file hosts from the directory /etc to the directory /home.

```
OdiFileCopy -FILE=/etc/hosts -TOFILE=/home/hosts
```
Copy all *.csv files from the directory /etc to the directory /home and overwrite.

```
OdiFileCopy -FILE=/etc/*.csv -TODIR=/home -OVERWRITE=yes
```

OdiFileDelete

Use this command to delete files or directories.

The most common uses of this tool are described in the following table where:

- x means is supplied
- o means is omitted

-DIR	-FILE	-RECURSE	Behavior
x	x	x	Every file with the name or with a name matching the mask specified in -FILE is deleted from -DIR and from all of its subdirectories.
x	o	x	The subdirectories from -FILE are deleted.
x	x	o	Every file with the name or with a name matching the mask specified in -FILE is deleted from -DIR.
x	o	o	The -DIR is deleted.

Usage

```
OdiFileDelete -DIR=<directory> -FILE=<file> [-RECURSE=<yes|no>]
[-CASESENS=<yes|no>] [-NOFILE_ERROR=<yes|no>] [-FROMDATE=<from_date>]
[-TODATE=<to_date>]
```

Parameters

Parameters	Mandatory	Description
-DIR=<directory>	Yes if -FILE is omitted	If -FILE is omitted, specifies the name of the directory (folder) to delete.
		If -FILE is supplied, specifies the path where files should be deleted from.
-FILE=<file>	Yes if -DIR is omitted	Name or mask of file(s) to delete. If -DIR is not specified, provide the full path. Use * to specify wildcard characters.
		Examples:
		/var/tmp/*.log (all files with the log extension of the directory /var/tmp)
		/arch_*.lst (all files starting with arch_ and with the extension lst)
-RECURSE=<yes|no>	No	If -FILE is omitted, the -RECURSE parameter has no effect: all subdirectories are implicitly deleted.
		If -FILE is supplied, the -RECURSE parameter specifies if the files should be deleted from this directory and from all of its subdirectories.
		The default value is Yes.
-CASESENS=<yes|no>	No	Specifies that Oracle Data Integrator should distinguish uppercase and lowercase when matching file names. The default value is No.

Parameters	Mandatory	Description
-NOFILE_ERROR=<yes|no>	Yes	Indicates that an error should be generated if the specified directory or files are not found. The default value is Yes.
-FROMDATE=<from_date>	No	All files with a modification date later than this date are deleted. Use the format yyyy/MM/dd hh:mm:ss.
		The -FROM_DATE is not inclusive.
		If -FROMDATE is omitted, all files with a modification date earlier than the -TODATE date are deleted.
		If both -FROMDATE and -TODATE are omitted, all files matching the -FILE parameter value are deleted.
-TODATE=<to_date>	No	All files with a modification date earlier than this date are deleted. Use the format yyyy/MM/dd hh:mm:ss.
		The TO_DATE is not inclusive.
		If -TODATE is omitted, all files with a modification date later than the -FROMDATE date are deleted.
		If both -FROMDATE and -TODATE parameters are omitted, all files matching the -FILE parameter value are deleted.

> **Note:** You cannot delete a file and a directory at the same time by combining the -DIR and -FILE parameters. To achieve that, you must make two calls to OdiFileDelete.

Examples

Delete the file my_data.dat from the directory c:\data\input, generating an error if the file or directory is missing.

```
OdiFileDelete -FILE=c:\data\input\my_data.dat -NOFILE_ERROR=yes
```
Delete all .txt files from the bin directory, but not .TXT files.

```
OdiFileDelete "-FILE=c:\Program Files\odi\bin\*.txt" -CASESENS=yes
```
This statement has the same effect:

```
OdiFileDelete "-DIR=c:\Program Files\odi\bin" "-FILE=*.txt" -CASESENS=yes
```
Delete the directory /bin/usr/nothingToDoHere.

```
OdiFileDelete "-DIR=/bin/usr/nothingToDoHere"
```
Delete all files under the C:\temp directory whose modification time is between 10/01/2008 00:00:00 and 10/31/2008 22:59:00, where 10/01/2008 and 10/31/2008 are not inclusive.

```
OdiFileDelete -DIR=C:\temp -FILE=* -NOFILE_ERROR=NO -FROMDATE=FROMDATE=10/01/2008
00:00:00 -TODATE=10/31/2008 22:59:00
```
Delete all files under the C:\temp directory whose modification time is earlier than 10/31/2008 17:00:00.

```
OdiFileDelete -DIR=C:\temp -FILE=* -NOFILE_ERROR=YES -TODATE=10/31/2008 17:00:00
```
Delete all files under the C:\temp directory whose modification time is later than 10/01/2008 08:00:00.

```
OdiFileDelete -DIR=C:\temp -FILE=* -NOFILE_ERROR=NO -FROMDATE=10/01/2008 08:00:00
```

OdiFileMove

Use this command to move or rename files or a directory into files or a directory.

Usage

```
OdiFileMove -FILE=<file> -TODIR=<target_directory> -TOFILE=<target_file>
[-OVERWRITE=<yes|no>] [-RECURSE=<yes|no>] [-CASESENS=<yes|no>]

OdiFileMove -DIR=<directory> -TODIR=<target_directory> [-OVERWRITE=<yes|no>]
[-RECURSE=<yes|no>] [-CASESENS=<yes|no>]
```

Parameters

Parameters	Mandatory	Description
-DIR=<directory>	Yes if -FILE is omitted	Directory (or folder) to move or rename.
-FILE=<file>	Yes if -DIR is omitted	Full path of the file(s) to move or rename. Use * for generic characters.
		Examples:
		/var/tmp/*.log (all files with the log extension in the directory /var/tmp)
		arch_*.lst (all files starting with arch_ and with the extension lst)
-TODIR=<target_directory>	Yes if -DIR is specified	Target directory of the move.
		If a directory is moved (-DIR), this parameter indicates the new name of the directory.
		If a file or several files are moved (-FILE), this parameter indicates the target directory.
-TOFILE=<target_file>	Yes if -TODIR is omitted	Target file(s). This parameter cannot be used with parameter -DIR.
		This parameter is:
		■ The new name of the target file if one single file is moved (no generic character).
		■ The mask of the new file names if several files are moved.
-OVERWRITE=<yes|no>	No	Indicates if the files or directory are overwritten if they exist. The default value is No.
-RECURSE=<yes|no>	No	Indicates if files are moved recursively when the directory contains other directories. The value No indicates that only files contained in the directory to move (not the subdirectories) are moved. The default value is Yes.
-CASESENS=<yes|no>	No	Indicates if file search is case-sensitive. By default, Oracle Data Integrator searches files in uppercase (set to No).

Examples

Rename the `hosts` file to `hosts.old`.

```
OdiFileMove -FILE=/etc/hosts -TOFILE=/etc/hosts.old
```
Move the file `hosts` from the directory `/etc` to the directory `/home/odi`.

```
OdiFileMove -FILE=/etc/hosts -TOFILE=/home/odi/hosts
```
Move all files `*.csv` from directory `/etc` to directory `/home/odi` with overwrite.

```
OdiFileMove -FILE=/etc/*.csv -TODIR=/home/odi -OVERWRITE=yes
```
Move all `*.csv` files from directory `/etc` to directory `/home/odi` and change their extension to `.txt`.

```
OdiFileMove -FILE=/etc/*.csv -TOFILE=/home/odi/*.txt -OVERWRITE=yes
```
Rename the directory `C:\odi` to `C:\odi_is_wonderful`.

```
OdiFileMove -DIR=C:\odi -TODIR=C:\odi_is_wonderful
```
Move the directory `C:\odi` and its subfolders into the directory `C:\Program Files\odi`.

```
OdiFileMove -DIR=C:\odi "-TODIR=C:\Program Files\odi" -RECURSE=yes
```

OdiFileWait

Use this command to manage file events. This command regularly scans a directory and waits for a number of files matching a mask to appear, until a given timeout is reached. When the specified files are found, an action on these files is triggered.

Usage

```
OdiFileWait -DIR=<directory> -PATTERN=<pattern>
[-ACTION=<DELETE|COPY|MOVE|APPEND|ZIP|NONE>] [-TODIR=<target_directory>]
[-TOFILE=<target_file>] [-OVERWRITE=<yes|no>] [-CASESENS=<yes|no>]
[-FILECOUNT=<n>] [-TIMEOUT=<n>] [-POLLINT=<n>] [-HEADER=<n>]
[-KEEP_FIRST_HEADER=<yes|no>] [-NOFILE_ERROR=<yes|no>]
```

Parameters

Parameters	Mandatory	Description
-ACTION= `<DELETE|COPY|MOVE|APPEND|ZIP|NONE>`	No	Action taken on the files found: DELETE: Delete the files found. COPY: Copy the files found into the directory -TODIR. MOVE: Move or rename the files found into folder -TODIR by naming them as specified by -TOFILE. APPEND: Concatenates all files found and creates a result file -TOFILE. Source files are deleted. ZIP: Compress the files found and store them in ZIP file -TOFILE. NONE (default): No action is performed.
-DIR=<directory>	Yes	Directory (or folder) to scan.
-PATTERN=<pattern>	Yes	Mask of file names to scan. Use * to specify the generic characters. Examples: *.log (all files with the log extension) arch_*.lst (all files starting with arch_ and with the extension lst)
-TODIR=<target_directory>	No	Target directory of the action. When the action is: COPY: Directory where the files are copied. MOVE: Directory where the files are moved.

Parameters	Mandatory	Description
`-TOFILE=<target_file>`	No	Destination file(s). When the action is:
		`MOVE`: Renaming mask of the moved files.
		`APPEND`: Name of the file resulting from the concatenation.
		`ZIP`: Name of the resulting ZIP file.
		`COPY`: Renaming mask of the copied files.
		Renaming rules:
		■ Any alphanumeric character is replaced in the original file name with the alphanumeric characters specified for `<target_file>`.
		■ `?` at `-TOFILE` leaves origin symbol on this position.
		■ `*` at `-TOFILE` means all remaining symbols from origin file name.
`-OVERWRITE=<yes|no>`	No	Indicates if the destination file(s) will be overwritten if they exist. The default value is No.
		Note that if this option is used with `APPEND`, the target file will only contain the contents of the latest file processed.
`-CASESENS=<yes|no>`	No	Indicates if file search is case-sensitive. By default, Oracle Data Integrator searches files in uppercase (set to No).
`-FILECOUNT=<n>`	No	Maximum number of files to wait for (the default value is 0). If this number is reached, the command ends.
		The value 0 indicates that Oracle Data Integrator waits for all files until the timeout is reached.
		If this parameter is 0 and the timeout is also 0, this parameter is then forced implicitly to 1.
`-TIMEOUT=<n>`	No	Maximum waiting time in milliseconds (the default value is 0).
		If this delay is reached, the command yields control to the following command and uses its value `-FILECOUNT`.
		The value 0 is used to specify an infinite waiting time (wait until the maximum number of messages to read as specified in the parameter `-FILECOUNT`).
`-POLLINT=<n>`	No	Interval in milliseconds to search for new files. The default value is 1000 (1 second), which means that Oracle Data Integrator looks for new messages every second. Files written during the OdiFileWait are taken into account only after being closed (file size unchanged) during this interval.

Parameters	Mandatory	Description
-HEADER=<n>	No	This parameter is valid only for the APPEND action.
		Number of header lines to suppress from the files before concatenation. The default value is 0 (no processing).
-KEEP_FIRST_HEADER=<yes|no>	No	This parameter is valid only for the APPEND action.
		Keeps the header lines of the first file during the concatenation. The default value is Yes.
-NOFILE_ERROR=<yes|no>	No	Indicates the behavior if no file is found.
		The default value is No, which means that no error is generated if no file is found.

Examples

Wait indefinitely for file flag.txt in directory c:\events and proceed when this file is detected.

```
OdiFileWait -ACTION=NONE -DIR=c:\events -PATTERN=flag.txt -FILECOUNT=1
-TIMEOUT=0 -POLLINT=1000
```

Wait indefinitely for file flag.txt in directory c:\events and suppress this file when it is detected.

```
OdiFileWait -ACTION=DELETE -DIR=c:\events -PATTERN=flag.txt -FILECOUNT=1
-TIMEOUT=0 -POLLINT=1000
```

Wait for the sales files *.dat for 5 minutes and scan every second in directory c:\sales_in, then concatenate into file sales.dat in directory C:\sales_ok. Keep the header of the first file.

```
OdiFileWait -ACTION=APPEND -DIR=c:\sales_in -PATTERN=*.dat
TOFILE=c:\sales_ok\sales.dat -FILECOUNT=0 -TIMEOUT=350000 -POLLINT=1000
-HEADER=1 -KEEP_FIRST_HEADER=yes -OVERWRITE=yes
```

Wait for the sales files *.dat for 5 minutes every second in directory c:\sales_in, then copy these files into directory C:\sales_ok. Do not overwrite.

```
OdiFileWait -ACTION=COPY -DIR=c:\sales_in -PATTERN=*.dat -TODIR=c:\sales_ok
-FILECOUNT=0 -TIMEOUT=350000 -POLLINT=1000 -OVERWRITE=no
```

Wait for the sales files *.dat for 5 minutes every second in directory c:\sales_in and then archive these files into a ZIP file.

```
OdiFileWait -ACTION=ZIP -DIR=c:\sales_in -PATTERN=*.dat
-TOFILE=c:\sales_ok\sales.zip -FILECOUNT=0 -TIMEOUT=350000
-POLLINT=1000 -OVERWRITE=yes
```

Wait for the sales files *.dat for 5 minutes every second into directory c:\sales_in, then move these files into directory C:\sales_ok. Do not overwrite. Append .bak to the file names.

```
OdiFileWait -ACTION=MOVE -DIR=c:\sales_in -PATTERN=*.dat
-TODIR=c:\sales_ok -TOFILE=*.bak -FILECOUNT=0 -TIMEOUT=350000
-POLLINT=1000 -OVERWRITE=no
```

OdiFtp

Use this command to use the FTP protocol to connect to a remote system and to perform standard FTP commands on the remote system. Trace from the script is recorded against the Execution Details of the task representing the OdiFtp step in Operator Navigator.

Usage

```
OdiFtp -HOST=<ftp server host name> -USER=<ftp user>
[-PASSWORD=<ftp user password>] -REMOTE_DIR=<remote dir on ftp host>
-LOCAL_DIR=<local dir> [-PASSIVE_MODE=<yes|no>] [-TIMEOUT=<time in seconds>]
[-STOP_ON_FTP_ERROR=<yes|no>] -COMMAND=<command>
```

Parameters

Parameters	Mandatory	Description
-HOST=<ftp server host name>	Yes	Host name of the FTP server.
-USER=<ftp user>	Yes	User on the FTP server.
-PASSWORD=<ftp user password>	No	Password of the FTP user.
-REMOTE_DIR=<remote dir on ftp host>	Yes	Directory path on the remote FTP host.
-LOCAL_DIR=<local dir>	Yes	Directory path on the local machine.
-PASSIVE_MODE=<yes|no>	No	If set to No, the FTP session uses Active Mode. The default value is Yes, which means the session runs in passive mode.
-TIMEOUT=<time in seconds>	No	Time in seconds after which the socket connection times out.
-STOP_ON_FTP_ERROR=<yes|no>	No	If set to Yes (default), the step stops when an FTP error occurs instead of running to completion.
-COMMAND=<command>	Yes	Raw FTP command to execute. For a multiline command, pass the whole command as raw text after the OdiFtp line without the -COMMAND parameter.
		Supported commands:
		APPE, CDUP, CWD, DELE, LIST, MKD, NLST, PWD, QUIT, RETR, RMD, RNFR, RNTO, SIZE, STOR

Examples

Execute a script on a remote host that makes a directory, changes directory into the directory, puts a file into the directory, and checks its size. The script appends another file, checks the new size, and then renames the file to dailyData.csv. The -STOP_ON_FTP_ERROR parameter is set to No so that the script continues even if the directory exists.

```
OdiFtp -HOST=machine.oracle.com -USER=odiftpuser -PASSWORD=<password>
-LOCAL_DIR=/tmp -REMOTE_DIR=c:\temp -PASSIVE_MODE=YES -STOP_ON_FTP_ERROR=No
MKD dataDir
CWD dataDir
STOR customers.csv
```

```
SIZE customers.csv
APPE new_customers.csv customers.csv
SIZE customers.csv
RNFR customers.csv
RNTO dailyData.csv
```

OdiFtpGet

Use this command to download a file from an FTP server.

Usage

```
OdiFtpGet -HOST=<ftp server host name> -USER=<ftp user>
[PASSWORD=<ftp user password>] -REMOTE_DIR=<remote dir on ftp host>
[-REMOTE_FILE=<file name under the -REMOTE_DIR>] -LOCAL_DIR=<local dir>
[-LOCAL_FILE=<file name under the -LOCAL_DIR>] [-PASSIVE_MODE=<yes|no>]
[-TIMEOUT=<time in seconds>]
```

> **Note:** If a Local or Remote file name needs to have % as part of its name, %25 needs to be passed instead of just %.
>
> %25 will resolve automatically to %.
>
> For example, if file name needs to be temp%result, it should be passed as REMOTE_FILE=temp%25result or -LOCAL_FILE=temp%25result.

Parameters

Parameters	Mandatory	Description
-HOST=<host name of the ftp server>	Yes	Host name of the FTP server.
-USER=<host name of the ftp user>	Yes	User on the FTP server.
-PASSWORD=<password of the ftp user>	No	Password of the FTP user.
-REMOTE_DIR=<dir on the ftp host>	Yes	Directory path on the remote FTP host.
-REMOTE_FILE=<file name under -REMOTE DIR>	No	File name under the directory specified in the -REMOTE_DIR argument. If this argument is missing, the file is copied with the -LOCAL_FILE file name. If the -LOCAL_FILE argument is also missing, the -LOCAL_DIR is copied recursively to the -REMOTE_DIR.
-LOCAL_DIR=<local dir path>	Yes	Directory path on the local machine.
-LOCAL_FILE=<local file>	No	File name under the directory specified in the -LOCAL_DIR argument. If this argument is missing, all files and directories under the -LOCAL_DIR are copied recursively to the -REMOTE_DIR. To filter the files to be copied, use * to specify the generic characters. Examples: *.log (all files with the log extension) - arch_*.lst (all files starting with arch_ and with the extension lst)
-PASSIVE_MODE=<yes|no>]	No	If set to No, the FTP session uses Active Mode. The default value is Yes, which means the session runs in passive mode.

Parameters	Mandatory	Description
-TIMEOUT=<time in seconds>	No	The time in seconds after which the socket connection times out.
-TGT_LSCHEMA=<target_file>	No	The file located on a data server resolved based on the Logical Schema value. For example, the LSCHEMA may point to a Hadoop Data Server and the tool will access the file from that data server.

Examples

Copy the remote directory /test_copy555 on the FTP server recursively to the local directory C:\temp\test_copy.

```
OdiFtpGet -HOST=machine.oracle.com -USER=test_ftp -PASSWORD=<password> -LOCAL_
DIR=C:\temp\test_copy -REMOTE_DIR=/test_copy555
```
Copy all files matching the Sales*.txt pattern under the remote directory / on the FTP server to the local directory C:\temp\ using Active Mode for the FTP connection.

```
OdiFtpGet -HOST=machine.oracle.com -USER=test_ftp -PASSWORD=<password> -LOCAL_
DIR=C:\temp -LOCAL_FILE=Sales*.txt -REMOTE_DIR=/ -PASSIVE_MODE=NO
```

OdiFtpPut

Use this command to upload a local file to an FTP server.

Usage

```
OdiFtpPut -HOST=<ftp server host name> -USER=<ftp user>
[PASSWORD=<ftp user password>] -REMOTE_DIR=<remote dir on ftp host>
[-REMOTE_FILE=<file name under the -REMOTE_DIR>] -LOCAL_DIR=<local dir>
[-LOCAL_FILE=<file name under the -LOCAL_DIR>] [-PASSIVE_MODE=<yes|no>]
[-TIMEOUT=<time in seconds>]
```

> **Note:** If a Local or Remote file name needs to have % as part of its name, %25 needs to be passed instead of just %.
>
> %25 will resolve automatically to %.
>
> For example, if file name needs to be temp%result, it should be passed as REMOTE_FILE=temp%25result or -LOCAL_FILE=temp%25result.

Parameters

Parameters	Mandatory	Description
-HOST=<host name of the ftp server>	Yes	Host name of the FTP server.
-USER=<host name of the ftp user>	Yes	User on the FTP server.
-PASSWORD=<password of the ftp user>	No	Password of the FTP user.
-REMOTE_DIR=<dir on the ftp host>	Yes	Directory path on the remote FTP host.
-REMOTE_FILE=<file name under -REMOTE DIR>	No	File name under the directory specified in the -REMOTE_DIR argument. If this argument is missing, the file is copied with the -LOCAL_FILE file name. If the -LOCAL_FILE argument is also missing, the -LOCAL_DIR is copied recursively to the -REMOTE_DIR.
-LOCAL_DIR=<local dir path>	Yes	Directory path on the local machine.
-LOCAL_FILE=<local file>	No	File name under the directory specified in the -LOCAL_DIR argument. If this argument is missing, all files and directories under the -LOCAL_DIR are copied recursively to the -REMOTE_DIR. To filter the files to be copied, use * to specify the generic characters. Examples: ■ *.log (all files with the log extension) ■ arch_*.lst (all files starting with arch_ and with the extension lst)
-PASSIVE_MODE=<yes|no>	No	If set to No, the FTP session uses Active Mode. The default value is Yes, which means the session runs in passive mode.

Parameters	Mandatory	Description
-TIMEOUT=<time in seconds>	No	The time in seconds after which the socket connection times out.

> **Note:** For OdiFtp execution to be successful, you must have LIST privilege in the user's home directory.

Examples

Copy the local directory `C:\temp\test_copy` recursively to the remote directory `/test_copy555` on the FTP server.

```
OdiFtpPut -HOST=machine.oracle.com -USER=test_ftp -PASSWORD=<password>
 -LOCAL_DIR=C:\temp\test_copy -REMOTE_DIR=/test_copy555"
```

Copy all files matching the `Sales*.txt` pattern under the local directory `C:\temp\` to the remote directory `/` on the FTP server.

```
OdiFtpPut -HOST=machine.oracle.com -USER=test_ftp -PASSWORD=<password> -LOCAL_
DIR=C:\temp -LOCAL_FILE=Sales*.txt -REMOTE_DIR=/
```

Copy the `Sales1.txt` file under the local directory `C:\temp\` to the remote directory `/` on the FTP server as a `Sample1.txt` file.

```
OdiFtpPut -HOST=machine.oracle.com -USER=test_ftp -PASSWORD=<password> -LOCAL_
DIR=C:\temp -LOCAL_FILE=Sales1.txt -REMOTE_DIR=/Sample1.txt
```

OdiGenerateAllScen

Use this command to generate a set of scenarios from design-time components (Packages, Mappings, Procedures, or Variables) contained in a folder or project, filtered by markers.

Usage

```
OdiGenerateAllScen -PROJECT=<project_id> [-FOLDER=<folder_id>]
[-MODE=<REPLACE|CREATE>] [-GRPMARKER=<marker_group_code>
[-MARKER=<marker_code>] [-MATERIALIZED=<yes|no>]
[-GENERATE_MAP=<yes|no>] [-GENERATE_PACK=<yes|no>]
[-GENERATE_POP=<yes|no>] [-GENERATE_TRT=<yes|no>]
[-GENERATE_VAR=<yes|no>]
```

Parameters

Parameters	Mandatory	Description
-PROJECT=<project_id>	Yes	ID of the Project containing the components to generate scenarios for.
-FOLDER=<folder_id>	No	ID of the Folder containing the components to generate scenarios for.
-MODE=<REPLACE|CREATE>	No	Scenario generation mode: ■ REPLACE (default): Causes the last scenario generated for the component to be replaced by the new one generated, with no change of name or version. Any schedules linked to this scenario are deleted. If no scenario exists, a new one is generated. ■ CREATE: Creates a new scenario with the same name as the latest scenario generated for the component, with the version number automatically incremented (if the latest version is an integer) or set to the current date (if the latest version is not an integer). If no scenario has been created for the component, a first version of the scenario is automatically created. New scenarios are named after the component according to the *Scenario Naming Convention* user parameter.
-GRPMARKER=<marker_group_code>	No	Group containing the marker used to filter the components for which scenarios must be generated. When -GRPMARKER and -MARKER are specified, scenarios will be (re-)generated only for components flagged with the marker identified by the marker code and the marker group code.
-MARKER=<marker_code>	No	Marker used to filter the components for which scenarios must be generated. When -GRPMARKER and -MARKER are specified, scenarios will be (re-)generated only for components flagged with the marker identified by the marker code and the marker group code.

Parameters	Mandatory	Description
-MATERIALIZED=<yes|no>	No	Specifies whether scenarios should be generated as if all underlying objects are materialized. The default value is No.
-GENERATE_MAP=<yes|no>	No	Specifies whether scenarios should be generated from the mapping. The default value is No.
-GENERATE_PACK=<yes|no>	No	Specifies whether scenarios attached to packages should be (re-)generated. The default value is Yes.
-GENERATE_POP=<yes|no>	No	Specifies whether scenarios attached to mappings should be (re-)generated. The default value is No.
-GENERATE_TRT=<yes|no>	No	Specifies whether scenarios attached to procedures should be (re-)generated. The default value is No.
-GENERATE_VAR=<yes|no>	No	Specifies whether scenarios attached to variables should be (re-)generated. The default value is No.

Examples

Generate all scenarios in the project whose ID is 1003 for the current repository.

```
OdiGenerateAllScen -PROJECT=1003
```

OdiImportObject

Use this command to import the contents of an export file into a repository. This command reproduces the behavior of the import feature available from the user interface.

Use caution when using this tool. It may work incorrectly when importing objects that depend on objects that do not exist in the repository. It is recommended that you use this API for importing high-level objects (projects, models, and so on).

> **WARNING:** The import type and the order in which objects are imported into a repository should be carefully specified. Refer to the chapter Exporting and Importing in *Developing Integration Projects with Oracle Data Integrator* for more information on import.

Usage

```
OdiImportObject -FILE_NAME=<FileName> [-WORK_REP_NAME=<workRepositoryName>]
-IMPORT_MODE=<DUPLICATION|SYNONYM_INSERT|SYNONYM_UPDATE|SYNONYM_INSERT_UPDATE>]
[-IMPORT_SCHEDULE=<yes|no>] [-EXPORT_KEY=<key>] [-UPGRADE_KEY=<upgradeKey>]
[IMPORT_WITHOUT_CIPHER_DATA=<yes|no>]
```

Parameters

Parameters	Mandatory	Description
-FILE_NAME=<FileName>	Yes	Name of the XML export file to import.
-WORK_REP_NAME=<workRepositoryName>	No	Name of the work repository into which the object must be imported. This work repository must be defined in the connected master repository. If this parameter is not specified, the object is imported into the current master or work repository.
-IMPORT_MODE=<DUPLICATION|SYNONYM_INSERT|SYNONYM_UPDATE|SYNONYM_INSERT_UPDATE>	Yes	Import mode for the object. The default value is DUPLICATION. For more information about import types, see Import Modes in *Developing Integration Projects with Oracle Data Integrator*.
-IMPORT_SCHEDULE=<yes|no>	No	If the selected file is a scenario export, imports the schedules contained in the scenario export file. The default value is No.
-EXPORT_KEY=<key>	No[1]	Specifies a cryptographic private key used to encrypt sensitive cipher data. You must specify this key when importing the exported object in order to import the cipher data.
-UPGRADE_KEY=<upgradeKey>	No	Upgrade key to import repository objects from earlier versions of Oracle Data Integrator (pre-12*c*).

Parameters	Mandatory	Description
-IMPORT_WITHOUT_CIPHER_DATA=<yes|no>	No[2]	When set to Yes, specifies that sensitive (cipher) values should be set to null in the object when it is imported. When set to No or when this parameter is omitted, you must include the -EXPORT_KEY parameter and specify a valid key. The default value is No.

[1] If the -EXPORT_KEY parameter is not specified, the -IMPORT_WITHOUT_CIPHER_DATA parameter must be specified, and must be set to Yes.

[2] If -EXPORT_WITHOUT_CIPHER_DATA is not specified, or if it is specified and set to No, you must specify the -EXPORT_KEY parameter with a valid key value.

Examples

Import the /temp/DW01.xml export file (a project) into the WORKREP work repository using DUPLICATION mode.

```
OdiImportObject -FILE_NAME=/temp/DW01.xml -WORK_REP_NAME=WORKREP
-IMPORT_MODE=DUPLICATION -EXPORT_KEY=examplekey1
```

OdiImportScen

Use this command to import a scenario into the current work repository from an export file.

Usage

```
OdiImportScen -FILE_NAME=<FileName>
[-IMPORT_MODE=<DUPLICATION|SYNONYM_INSERT|SYNONYM_UPDATE|SYNONYM_INSERT_UPDATE>]
[-EXPORT_KEY=<key>] [-IMPORT_SCHEDULE=<yes|no>] [-FOLDER=<parentFolderGlobalId>]
[-UPGRADE_KEY=<upgradeKey>] [IMPORT_WITHOUT_CIPHER_DATA=<yes|no>]
```

Parameters

Parameters	Mandatory	Description
-FILE_NAME=<FileName>	Yes	Name of the export file.
-IMPORT_MODE=<DUPLICATION|SYNONYM_ INSERT|SYNONYM_UPDATE|SYNONYM_INSERT_UPDATE>	No	Import mode of the scenario. The default value is DUPLICATION. For more information about import types, see Import Modes in *Developing Integration Projects with Oracle Data Integrator*.
-EXPORT_KEY=<key>	No[1]	Specifies a cryptographic private key used to encrypt sensitive cipher data. You must specify this key when importing the exported object in order to import the cipher data.
-IMPORT_SCHEDULE=<yes|no>	No	Imports the schedules contained in the scenario export file. The default value is No.
-FOLDER=<parentFolderGlobalId>	No	Global ID of the parent scenario folder.
-UPGRADE_KEY=<upgradeKey>	No	Upgrade key to import repository objects from earlier versions of Oracle Data Integrator (pre-12*c*).
-IMPORT_WITHOUT_CIPHER_DATA=<yes|no>	No[2]	When set to Yes, specifies that sensitive (cipher) values should be set to null in the object when it is imported. When set to No or when this parameter is omitted, you must include the -EXPORT_KEY parameter and specify a valid key. The default value is No.

[1] If the -EXPORT_KEY parameter is not specified, the -IMPORT_WITHOUT_CIPHER_DATA parameter must be specified, and must be set to Yes.

[2] If -IMPORT_WITHOUT_CIPHER_DATA is not specified, or if it is specified and set to No, you must specify the -EXPORT_KEY parameter with a valid key value.

Examples

Import the /temp/load_dwh.xml export file (a scenario) into the current work repository using DUPLICATION mode.

```
OdiImportScen -FILE_NAME=/temp/load_dwh.xml -IMPORT_MODE=DUPLICATION
-EXPORT_KEY=examplekey1
```

OdiInvokeWebService

> **Note:** This tool replaces the OdiExecuteWebService tool.

Use this command to invoke a web service over HTTP/HTTPS and write the response to an XML file.

This tool invokes a specific *operation* on a *port* of a web service whose description file (WSDL) *URL* is provided.

If this operation requires a web service request, it is provided either in a request file, or directly written out in the tool call (<XML Request>). This request file can have two different formats (XML, which corresponds to the XML body only, or SOAP, which corresponds to the full-formed SOAP envelope including a SOAP header and body) specified in the -RESPONSE_FILE_FORMAT parameter. The response of the web service request is written to an XML file that can be processed afterwards in Oracle Data Integrator. If the web service operation is one-way and does not return any response, no response file is generated.

> **Note:** This tool cannot be executed in a command line with startcmd.

Usage

```
OdiInvokeWebService -URL=<url> -PORT=<port> -OPERATION=<operation>
[<XML Request>] [-REQUEST_FILE=<xml_request_file>]
[-RESPONSE_MODE=<NO_FILE|NEW_FILE|FILE_APPEND>]
[-RESPONSE_FILE=<xml_response_file>] [-RESPONSE_XML_ENCODING=<charset>]
[-RESPONSE_FILE_CHARSET=<charset>] [-RESPONSE_FILE_FORMAT=<XML|SOAP>]
[-HTTP_USER=<user>]
[-HTTP_PASS=<password>] [-TIMEOUT=<timeout>]
```

Parameters

Parameters	Mandatory	Description
-URL=<url>	Yes	URL of the Web Service Description File (WSDL) file describing the web service.
-PORT_TYPE=<port_type>	Yes	Name of the WSDL port type to invoke.
-OPERATION=<operation>	Yes	Name of the web service operation to invoke.
<XML Request>	No	Request message in SOAP (Simple Object Access Protocol) format. This message should be provided on the line immediately following the OdiInvokeWebService call.
		The request can alternately be passed through a file whose location is provided with the -REQUEST_FILE parameter.

Parameters	Mandatory	Description
-REQUEST_FILE=<xml_request_file>	No	Location of the XML file containing the request message in SOAP format.
		The request can alternately be directly written out in the tool call (<xmlRequest>).
-RESPONSE_MODE=<NO_FILE|NEW_FILE|FILE_APPEND>	No	Generation mode for the response file. This parameter takes the following values:
		▪ NO_FILE (default): No response file is generated.
		▪ NEW_FILE: A new response file is generated. If the file already exists, it is overwritten.
		▪ FILE_APPEND: The response is appended to the file. If the file does not exist, it is created.
-RESPONSE_FILE=<file>	Depends	The name of the result file to write. Mandatory if -RESPONSE_MODE is NEW_FILE or APPEND.
-RESPONSE_FILE_CHARSET=<charset>	Depends	Response file character encoding. See the following table. Mandatory if -RESPONSE_MODE is NEW_FILE or APPEND.
-RESPONSE_XML_ENCODING=<charset>	Depends	Character encoding that will be indicated in the XML declaration header of the response file. See the following table. Mandatory if -RESPONSE_MODE is not NO_FILE.
-RESPONSE_FILE_FORMAT=<XML|SOAP>	No	Format of the request and response file.
		▪ If XML is selected (default), the request is processed as a SOAP body. The tool adds a default SOAP header and envelope content to this body before sending the request. The response is stripped from its SOAP envelope and headers and only the response's body is written to the response file.
		▪ If SOAP is selected, the request is processed as a full-formed SOAP envelope and is sent as is. The response is also written to the response file with no processing.
-HTTP_USER=<user>	No	User account authenticating on the HTTP server.
-HTTP_PASS=<password>	No	Password of the HTTP user.
		Note: When using an ODI variable as the password, the variable content must be encrypted using the encode script.
-TIMEOUT=<timeout>	No	The web service request waits for a reply for this amount of time before considering that the server will not provide a response and an error is produced. The default value is 15 seconds.

The following table lists the most common XML/Java character encoding schemes. For a more complete list, see:

http://java.sun.com/j2se/1.4.2/docs/guide/intl/encoding.doc.html

XML Charset	Java Charset
US-ASCII	ASCII
UTF-8	UTF8
UTF-16	UTF-16
ISO-8859-1	ISO8859_1

Examples

The following web service call returns the capital city for a given country (the ISO country code is sent in the request). Note that the request and response format, as well as the port and operations available, are defined in the WSDL passed in the URL parameter.

```
OdiInvokeWebService -
-URL=http://www.oorsprong.org/websamples.countryinfo/CountryInfoService.wso
?WSDL -PORT_TYPE=CountryInfoServiceSoapType -OPERATION=CapitalCity
-RESPONSE_MODE=NEW_FILE -RESPONSE_XML_ENCODING=ISO-8859-1
"-RESPONSE_FILE=/temp/result.xml" -RESPONSE_FILE_CHARSET=ISO8859_1 -RESPONSE_FILE_
FORMAT=XML
<CapitalCityRequest>
<sCountryISOCode>US</sCountryISOCode>
</CapitalCityRequest>
```

The generated /temp/result.xml file contains the following:

```
<CapitalCityResponse>
<m:CapitalCityResponse>
<m:CapitalCityResult>Washington</m:CapitalCityResult>
</m:CapitalCityResponse>
</CapitalCityResponse>
```

Packages

Oracle Data Integrator provides a special graphical interface for calling OdiInvokeWebService in packages. See the chapter Using Web Services in *Developing Integration Projects with Oracle Data Integrator* for more information.

OdiKillAgent

Use this command to stop a standalone agent.

Java EE Agents deployed in an application server cannot be stopped using this tool and must be stopped using the application server utilities.

Usage

```
OdiKillAgent (-PORT=<TCP/IP Port>|-NAME=<physical_agent_name>)
[-IMMEDIATE=<yes|no>] [-MAX_WAIT=<timeout>]
```

Parameter

Parameters	Mandatory	Description
-PORT=<TCP/IP Port>	No	If this parameter is specified, the agent running on the local machine with the specified port is stopped.
-NAME=<physical_agent_name>	Yes	If this parameter is specified, the physical agent whose name is provided is stopped. This agent may be a local or remote agent. It must be declared in the master repository.
-IMMEDIATE=<yes|no>	No	If this parameter is set to Yes, the agent is stopped without waiting for its running sessions to complete. If this parameter is set to No, the agent is stopped after its running sessions reach completion or after the -MAX_WAIT timeout is reached. The default value is No.
-MAX_WAIT=<timeout>	No	This parameter can be used when -IMMEDIATE is set to No. The parameter defines a timeout in milliseconds after which the agent is stopped regardless of the running sessions. The default value is 0, which means no timeout and the agent is stopped after its running sessions complete.

Examples

Stop the ODI_AGT_001 physical agent immediately.

```
OdiKillAgent -NAME=ODI_AGT_001 -IMMEDIATE=yes
```

OdiManageOggProcess

Use this command to start and stop Oracle GoldenGate processes.

The `-NB_PROCESS` parameter specifies the number of processes on which to perform the operation and applies only to Oracle GoldenGate Delivery processes.

If `-NB_PROCESS` is not specified, the name of the physical process is derived from the logical process. For example, if logical schema `R1_LS` maps to physical process `R1`, an Oracle GoldenGate process named `R1` is started or stopped.

If `-NB_PROCESS` is specified with a positive value, sequence numbers are appended to the process and all processes are started or stopped with the new name. For example, if the value is set to `3`, and logical schema `R2_LS` maps to physical process `R2`, processes `R21`, `R22` and `R23` are started or stopped.

If Start Journal is used to start the CDC (Changed Data Capture) process with Oracle GoldenGate JKMs (Journalizing Knowledge Modules), Oracle Data Integrator generates the Oracle GoldenGate Delivery process with the additional sequence number in the process name. For example, if Delivery process `RP` is used for the Start Journal action, Start Journal generates an Oracle GoldenGate Delivery process named `RP1`. To stop and start the process using the OdiManageOggProcess tool, set `-NB_PROCESS` to `1`. The maximum value of `-NB_PROCESS` is the value of the `-NB_APPLY_PROCESS` parameter of the JKM within the model.

Usage

```
OdiManageOggProcess -OPERATION=<start|stop>
-PROCESS_LSCHEMA=<OGG logical schema> [-NB_PROCESS=<number of processes>]
```

Parameters

Parameters	Mandatory	Description
`-OPERATION=<start|stop>`	Yes	Operation to perform on the process.
`-PROCESS_LSCHEMA=<OGG logical schema>`	Yes	Logical schema of the process.
`-NB_PROCESS=<number of processes>`	No	Number of processes on which to perform the operation.

Examples

Start Oracle GoldenGate process `R1`, which maps to logical schema `R1_LS`.

```
OdiManageOggProcess "-OPERATION=START" "-PROCESS_LSCHEMA=R1_LS
```

OdiMkDir

Use this command to create a directory structure.

If the parent directory does not exist, this command recursively creates the parent directories.

Usage

```
OdiMkDir -DIR=<directory>
```

Parameters

Parameters	Mandatory	Description
-DIR=<directory>	Yes	Directory (or folder) to create.
-TO_HDFS=<yes|no>	No	Indicates if the target is HDFS

Examples

Create the directory odi in C:\temp. If C:\temp does not exist, it is created.

```
OdiMkDir "-DIR=C:\temp\odi"
```

OdiOSCommand

Use this command to invoke an operating system command shell to carry out a command, and redirect the output result to files.

The following operating systems are supported:

- Windows operating systems, using `cmd`

- POSIX-compliant operating systems, using `sh`

The following operating systems are not supported:

- Mac OS

Usage

```
OdiOSCommand [-OUT_FILE=<stdout_file>] [-ERR_FILE=<stderr_file>]
[-FILE_APPEND=<yes|no>] [-WORKING_DIR=<workingdir>] [-SYNCHRONOUS=<yes|no>]
[CR/LF <command> | -COMMAND=<command>]
```

Parameters

Parameters	Mandatory	Description
-COMMAND=<command>	Yes	Command to execute. For a multiline command, pass the whole command as raw text after the OdiOSCommand line without the -COMMAND parameter.
-OUT_FILE=<stdout_file>	No	Absolute name of the file to redirect standard output to.
-ERR_FILE=<stderr_file>	No	Absolute name of the file to redirect standard error to.
-FILE_APPEND=<yes|no>	No	Whether to append to the output files, rather than overwriting them. The default value is Yes.
-WORKING_DIR=<workingdir>	No	Directory in which the command is executed.
-SYNCHRONOUS=<yes|no>	No	If set to Yes (default), the session waits for the command to terminate. If set to No, the session continues immediately with error code 0. The default is synchronous mode.
-CAPTURE_OUT_STREAM=[ON_ERROR[,]][ALL|NONE|[NSTART][,NEND]]	No	Use to capture some of the content that is written to the output stream and display in the Task Execution details in Operator. If set to **ON_ERROR**, the content will only be captured if the task fails. If set to **ALL** or **NONE**, either all or none of the output stream will be capture. Use **NSTART** and **NEND** to specify the number of lines to be captured (from the start and end).

Parameters	Mandatory	Description
-CAPTURE_ERR_STREAM=[ON_ERROR[,]][ALL|NONE|[NSTART][,NEND]]	No	Use to capture some of the content that is written to the error stream and display in the Task Error Message in Operator. If set to **ON_ERROR**, the content will only be captured if the task fails. If set to **ALL** or **NONE**, either all or none of the output stream will be capture. Use **NSTART** and **NEND** to specify the number of lines to be captured (from the start and end).

Examples

Execute the file `c:\work\load.bat` (on a Windows machine) and append the output streams to files.

```
OdiOSCommand "-OUT_FILE=c:\work\load-out.txt"
"-ERR_FILE=c:\work\load-err.txt" "-FILE_APPEND=YES"
"-WORKING_DIR=c:\work" c:\work\load.bat
```

OdiOutFile

Use this command to write or append content to a text file.

Usasge

```
OdiOutFile -FILE=<file_name> [-APPEND] [-CHARSET_ENCODING=<encoding>]
[-XROW_SEP=<hexadecimal_line_break>] [CR/LF <text> | -TEXT=<text>]
```

Parameters

Parameters	Mandatory	Description
-FILE=<file_name>	Yes	Target file. Its path may be absolute or relative to the execution agent location.
-APPEND	No	Indicates whether <text> must be appended at the end of the file. If this parameter is not specified, the file is overwritten if it exists.
-CHARSET_ENCODING=<encoding>	No	Target file encoding. The default value is ISO-8859-1. For the list of supported encodings, see: http://java.sun.com/j2se/1.4.2/docs/guide/intl/encoding.doc.html
-XROW_SEP=<hexadecimal_line_ break>	No	Hexadecimal code of the character used as a line separator (line break). The default value is 0A (UNIX line break). For a Windows line break, the value is 0D0A.
CR/LF <text> or -TEXT=<text>	No	Text to write in the file. This text can be typed on the line following the OdiOutFile command (a carriage return - CR/LF - indicates the beginning of the text), or can be defined with the -TEXT parameter. The -TEXT parameter should be used when calling this Oracle Data Integrator command from an OS command line. The text can contain variables or substitution methods.
-TO_HDFS=<yes|no>	No	Indicates if the output file is created in HDFS
-TGT_LSCHEMA	No	Indicates if the file is located on a data server resolved based on the Logical Schema value.

Examples

Generate the file /var/tmp/my_file.txt on the UNIX system of the agent that executed it.

```
OdiOutFile -FILE=/var/tmp/my_file.txt
Welcome to Oracle Data Integrator
This file has been overwritten by <%=odiRef.getSession("SESS_NAME")%>
```

Add the entry PLUTON into the file hosts of the Windows system of the agent that executed it.

```
OdiOutFile -FILE=C:\winnt\system32\drivers\etc\hosts -APPEND
195.10.10.6 PLUTON pluton
```

OdiPingAgent

Use this command to perform a test on a given agent. If the agent is not started, this command raises an error.

Usage

```
OdiPingAgent -AGENT_NAME=<physical_agent_name>
```

Parameters

Parameters	Mandatory	Description
-AGENT_NAME=<physical_agent_name>	Yes	Name of the physical agent to test.

Examples

Test the physical agent AGENT_SOLARIS_DEV.

```
OdiPingAgent -AGENT_NAME=AGENT_SOLARIS_DEV
```

OdiPurgeLog

Use this command to purge the execution logs.

The OdiPurgeLog tool purges all session logs and/or Load Plan runs that match the filter criteria.

The -PURGE_TYPE parameter defines the objects to purge:

- Select SESSION to purge all session logs matching the criteria. Child sessions and grandchild sessions are purged if the parent session matches the criteria. Note that sessions launched by a Load Plan execution, including the child sessions, are *not* purged.

- Select LOAD_PLAN_RUN to purge all load plan logs matching the criteria. Note that all sessions launched from the Load Plan run are purged even if the sessions attached to the Load Plan runs themselves do not match the criteria.

- Select ALL to purge both session logs and Load Plan runs matching the criteria.

The -COUNT parameter defines the number of sessions and/or Load Plan runs (after filter) to preserve in the log. The -ARCHIVE parameter enables automatic archiving of the purged sessions and/or Load Plan runs.

> **Note:** Load Plans and sessions in running, waiting, or queued status are not purged.

Usage

```
OdiPurgeLog
[-PURGE_TYPE=<SESSION|LOAD_PLAN_RUN|ALL>]
[-COUNT=<session_number>] [-FROMDATE=<from_date>] [TODATE=<to_date>]
[-CONTEXT_CODE=<context_code>] [-USER_NAME=<user_name>]
[-AGENT_NAME=<agent_name>] [-PURGE_REPORTS=<Yes|No>] [-STATUS=<D|E|M>]
[-NAME=<session_or_load_plan_name>] [-ARCHIVE=<Yes|No>] [-TODIR=<directory>]
[-ZIPFILE_NAME=<zipfile_name>] [-XML_CHARSET=<charset>] [-JAVA_CHARSET=<charset>]
[-REMOVE_TEMPORARY_OBJECTS=<yes|no>]
```

Parameters

Parameters	Mandatory	Description
-PURGE_TYPE=<SESSION|LOAD_PLAN_RUN|ALL>	No	Purges only session logs, Load Plan logs, or both. The default is session.
-COUNT=<session_number>	No	Retains the most recent count number of sessions and/or Load Plan runs that match the specified filter criteria and purges the rest. If this parameter is not specified or equals 0, purges all sessions and/or Load Plan runs that match the filter criteria.
-FROMDATE=<from_date>	No	Starting date for the purge, using the format yyyy/MM/dd hh:mm:ss.
		If -FROMDATE is omitted, the purge is done starting with the oldest session and/or Load Plan run.

Parameters	Mandatory	Description
-TODATE=<to_date>	No	Ending date for the purge, using the format yyyy/MM/dd hh:mm:ss.
		If -TODATE is omitted, the purge is done up to the most recent session and/or Load Plan run.
-CONTEXT_CODE=<context_code>	No	Purges only sessions and/or Load Plan runs executed in <context_code>.
		If -CONTEXT_CODE is omitted, the purge is done on all contexts.
-USER_NAME=<user_name>	No	Purges only sessions and/or Load Plan runs launched by <user_name>.
-AGENT_NAME=<agent_name>	No	Purges only sessions and/or Load Plan runs executed by <agent_name>.
-PURGE_REPORTS=<0|1>	No	If set to 1, scenario reports (appearing under the execution node of each scenario) are also purged.
-STATUS=<D|E|M>	No	Purges only the sessions and/or Load Plan runs with the specified state:
		■ D: Done
		■ E: Error
		■ M: Warning
		If this parameter is not specified, sessions and/or Load Plan runs in all of these states are purged.
-NAME=<session_or_load_plan_name>	No	Session name or Load Plan name.
-ARCHIVE=<Yes|No>	No	If set to Yes, exports the sessions and/or Load Plan runs before they are purged.
-TODIR=<directory>	No	Target directory for the export. This parameter is required if -ARCHIVE is set to Yes.
-ZIPFILE_NAME=<zipfile_name>	No	Name of the compressed file.
		Target directory for the export. This parameter is required if -ARCHIVE is set to Yes.
-XML_CHARSET=<charset>	No	XML encoding of the export files. The default value is ISO-8859-1. For the list of supported encodings, see:
		http://java.sun.com/j2se/1.4.2/docs/guide/intl/encoding.doc.html
-JAVA_CHARSET=<charset>	No	Export file encoding. The default value is ISO8859_1. For the list of supported encodings, see:
		http://java.sun.com/j2se/1.4.2/docs/guide/intl/encoding.doc.html
-REMOVE_TEMPORARY_ OBJECTS=<yes|no>	No	If set to Yes (default), cleanup tasks are performed before sessions are purged so that any temporary objects are removed.

Examples

Purge all sessions executed between 2001/03/25 00:00:00 and 2001/08/31 21:59:00.

```
OdiPurgeLog "-FROMDATE=2001/03/25 00:00:00" "-TODATE=2001/08/31 21:59:00"
```
Purge all Load Plan runs that were executed in the GLOBAL context by the Internal agent and that are in Error status.

```
OdiPurgeLog "-PURGE_TYPE=LOAD_PLAN_RUN" "-CONTEXT_CODE=GLOBAL"
"-AGENT_NAME=Internal" "-STATUS=E"
```

OdiReadMail

Use this command to read emails and attachments from a POP or IMAP account.

This command connects the mail server -MAILHOST using the connection parameters specified by -USER and -PASS. The execution agent reads messages from the mailbox until -MAX_MSG messages are received or the maximum waiting time specified by -TIMEOUT is reached. The extracted messages must match the filters such as those specified by the parameters -SUBJECT and -SENDER. When a message satisfies these criteria, its content and its attachments are extracted in a directory specified by the parameter -FOLDER. If the parameter -KEEP is set to No, the retrieved message is suppressed from the mailbox.

Usage

```
OdiReadMail -MAILHOST=<mail_host> -USER=<mail_user>
-PASS=<mail_user_password> -FOLDER=<folder_path>
[-PROTOCOL=<pop3|imap>] [-FOLDER_OPT=<none|sender|subject>]
[-KEEP=<no|yes>] [-EXTRACT_MSG=<yes|no>] [-EXTRACT_ATT=<yes|no>]
[-MSG_PRF=<my_prefix>] [-ATT_PRF=<my_prefix>] [-USE_UCASE=<no|yes>]
[-NOMAIL_ERROR=<no|yes>] [-TIMEOUT=<timeout>] [-POLLINT=<pollint>]
[-MAX_MSG=<max_msg>] [-SUBJECT=<subject_filter>] [-SENDER=<sender_filter>]
[-TO=<to_filter>] [-CC=<cc_filter>]
```

Parameters

Parameters	Mandatory	Description
-MAILHOST=<mail_host>	Yes	IP address of the POP or IMAP mail server.
-USER=<mail_user>	Yes	Valid mail server account.
-PASS=<mail_user_password>	Yes	Password of the mail server account.
-FOLDER=<folder_path>	Yes	Full path of the storage folder for attachments and messages.
-PROTOCOL=<pop3|imap>	No	Type of mail accessed (POP3 or IMAP). The default is POP3.
-FOLDER_OPT=<none|sender|subject>	No	Allows the creation of a subdirectory in the directory -FOLDER according to the following parameters: ■ none (default): No action. ■ sender: A subdirectory is created with the external name of the sender. ■ subject: A subdirectory is created with the subject of the message. For the sender and subject folder options, the spaces and nonalphanumeric characters (such as @) are replaced by underscores in the generated folder's name.

Parameters	Mandatory	Description
-KEEP=<no|yes>	No	If set to Yes, keep the messages that match the filters in the mailbox after reading them.
		If set to No (default), delete the messages that match the filters of the mailbox after reading them.
-EXTRACT_MSG=<yes|no>	No	If set to Yes (default), extract the body of the message into a file.
		If set to No, do not extract the body of the message into a file.
-EXTRACT_ATT=<yes|no>	No	If set to Yes (default), extract the attachments into files.
		If set to No, do not extract attachments.
-MSG_PRF=<my_prefix>	No	Prefix of the file that contains the body of the message. The default is MSG.
-ATT_PRF=<my_prefix>	No	Prefix of the files that contain the attachments. The original file names are kept.
-USE_UCASE=<no|yes>	No	If set to Yes, force the file names to uppercase.
		If set to No (default), keep the original letter case.
-NOMAIL_ERROR=<no|yes>	No	If set to Yes, generate an error if no mail matches the specified criteria.
		If set to No (default), do not generate an error when no mail corresponds to the specified criteria.
-TIMEOUT=<timeout>	No	Maximum waiting time in milliseconds. If this waiting time is reached, the command ends.
		The default value is 0, which means an infinite waiting time (as long as needed for the maximum number of messages specified with -MAX_MSG to be received).
-POLLINT=<pollint>	No	Searching interval in milliseconds to scan for new messages. The default value is 1000 (1 second).
-MAX_MSG=<max_msg>	No	Maximum number of messages to extract. If this number is reached, the command ends. The default value is 1.
-SUBJECT=<subject_filter>	No	Parameter used to filter the messages according to their subjects.
-SENDER=<sender_filter>	No	Parameter used to filter messages according to their sender.
-TO=<to_filter>	No	Parameter used to filter messages according to their addresses. This option can be repeated to create multiple filters.
-CC=<cc_filter>	No	Parameter used to filter messages according to their addresses in copy. This option can be repeated to create multiple filters.

Examples

Automatic reception of the mails of support with attachments detached in the folder `C:\support` on the system of the agent. Wait for all messages with a maximum waiting time of 10 seconds.

```
OdiReadMail -MAILHOST=mail.mymail.com -USER=myaccount -PASS=mypass
-KEEP=no -FOLDER=c:\support -TIMEOUT=0 -MAX_MSG=0
-SENDER=support@mycompany.com -EXTRACT_MSG=yes -MSG_PRF=TXT
-EXTRACT_ATT=yes
```

Wait indefinitely for 10 messages and check for new messages every minute.

```
OdiReadMail -MAILHOST=mail.mymail.com -USER=myaccount -PASS=mypass
-KEEP=no -FOLDER=c:\support -TIMEOUT=0 -MAX_MSG=10 -POLLINT=60000
-SENDER=support@mycompany.com -EXTRACT_MSG=yes -MSG_PRF=TXT
-EXTRACT_ATT=yes
```

OdiRefreshJournalCount

Use this command to refresh for a given journalizing subscriber the number of rows to consume for the given table list or CDC set. This refresh is performed on a logical schema and a given context, and may be limited.

> **Note:** This command is suitable for journalized tables in simple or consistent mode and cannot be executed in a command line with `startcmd`.

Usage

```
OdiRefreshJournalCount -LSCHEMA=<logical_schema>
-SUBSCRIBER_NAME=<subscriber_name>
(-TABLE_NAME=<table_name> | -CDC_SET_NAME=<cdc set name>)
[-CONTEXT=<context>] [-MAX_JRN_DATE=<to_date>]
```

Parameters

Parameters	Mandatory	Description
`-LSCHEMA=<logical_schema>`	Yes	Logical schema containing the journalized tables.
`-TABLE_NAME=<table_name>`	Yes for working with simple CDC	Journalized table name, mask, or list to check. This parameter accepts three formats: ■ Table Name ■ Table Name Mask: This mask selects the tables to poll. The mask is specified using the SQL LIKE syntax: the `%` symbol replaces an unspecified number of characters and the `_` symbol acts as a wildcard. ■ Table Names List: List of table names separated by commas. Masks as defined above are **not** allowed. Note that this option works only for tables in a model journalized in simple mode. This parameter cannot be used with `-CDC_SET_NAME`. It is mandatory if `-CDC_SET_NAME` is not set.
`-CDC_SET_NAME=<cdcSetName>`	Yes for working with consistent set CDC	Name of the CDC set to check. Note that this option works only for tables in a model journalized in consistent mode. This parameter cannot be used with `-TABLE_NAME`. It is mandatory if `-TABLE_NAME` is not set.
`-SUBSCRIBER_NAME=<subscriber_name>`	Yes	Name of the subscriber for which the count is refreshed.
`-CONTEXT=<context>`	No	Context in which the logical schema will be resolved. If no context is specified, the execution context is used.
`-MAX_JRN_DATE=<to_date>`	No	Date (and time) until which the journalizing events are taken into account.

Examples

Refresh for the CUSTOMERS table in the SALES_APPLICATION schema the count of modifications recorded for the SALES_SYNC subscriber. This datastore is journalized in simple mode.

```
OdiRefreshJournalCount -LSCHEMA=SALES_APPLICATION
-TABLE_NAME=CUSTOMERS -SUBSCRIBER_NAME=SALES_SYNC
```

Refresh for all tables from the SALES CDC set in the SALES_APPLICATION schema the count of modifications recorded for the SALES_SYNC subscriber. These datastores are journalized with consistent set CDC.

```
OdiRefreshJournalCount -LSCHEMA=SALES_APPLICATION
-SUBSCRIBER_NAME=SALES_SYNC -CDC_SET_NAME=SALES
```

OdiReinitializeSeq

Use this command to reinitialize an Oracle Data Integrator sequence.

Usage

```
OdiReinitializeSeq -SEQ_NAME=<sequence_name> -CONTEXT=<context>
-STD_POS=<position>
```

Parameters

Parameters	Mandatory	Description
-SEQ_NAME=<sequence_name>	Yes	Name of the sequence to reinitialize. It must be prefixed with GLOBAL. for a global sequence, or by <project code>. for a project sequence.
-CONTEXT=<context>	Yes	Context in which the sequence must be reinitialized.
-STD_POS=<position>	Yes	Position to which the sequence must be reinitialized.

Examples

Reset the global sequence SEQ_I to 0 for the GLOBAL context.

```
OdiReinitializeSeq -SEQ_NAME=GLOBAL.SEQ_I -CONTEXT=GLOBAL
-STD_POS=0
```

OdiRemoveTemporaryObjects

Use this command to remove temporary objects that could remain between executions. This is performed by executing the cleanup tasks for the sessions identified by the parameters specified in the tool parameters.

Usage

```
OdiRemoveTemporaryObjects [-COUNT=<session_number>] [-FROMDATE=<from_date>]
[-TODATE=<to_date>] [-CONTEXT_CODE=<context_code>]
[-AGENT_NAME=<agent_name>] [-USER_NAME=<user_name>]
[-NAME=<session_name>] [-ERRORS_ALLOWED=<number_of_errors_allowed>]
```

Parameter

Parameters	Mandatory	Description
-COUNT=<session_number>	No	Number of sessions for which to skip cleanup. The most recent number of sessions (<session_number>) is kept and the rest are cleaned up.
-FROMDATE=<from_date>	No	Start date for the cleanup, using the format yyyy/MM/dd hh:mm:ss. All sessions started after this date are cleaned up. If -FROMDATE is omitted, the cleanup starts with the oldest session.
-TODATE=<to_date>	No	End date for the cleanup, using the format yyyy/MM/dd hh:mm:ss. All sessions started before this date are cleaned up. If -TODATE is omitted, the cleanup starts with the most recent session.
-CONTEXT_CODE=<context_code>	No	Cleans up only those sessions executed in this context (<context_code>). If -CONTEXT_CODE is omitted, cleanup is performed on all contexts.
-AGENT_NAME=<agent_name>	No	Cleans up only those sessions executed by this agent (<agent_name>).
-USER_NAME=<user_name>	No	Cleans up only those sessions launched by this user (<user_name>).
-NAME=<session_name>	No	Session name.
-ERRORS_ALLOWED=<number_of_errors_allowed>	No	Number of errors allowed before the step ends with OK. If set to 0, the step ends with OK regardless of the number of errors encountered during the cleanup phase.

Examples

Remove the temporary objects by performing the cleanup tasks of all sessions executed between 2013/03/25 00:00:00 and 2013/08/31 21:59:00.

```
OdiRemoveTemporaryObjects "-FROMDATE=2013/03/25 00:00:00" "-TODATE=2013/08/31
21:59:00"
```

Remove the temporary objects by performing the cleanup tasks of all sessions executed in the GLOBAL context by the Internal agent.

```
OdiRemoveTemporaryObjects "-CONTEXT_CODE=GLOBAL" "-AGENT_NAME=Internal"
```

OdiRetrieveJournalData

Use this command to retrieve the journalized events for a given journalizing subscriber, a given table list or CDC set. The retrieval is performed specifically for the technology containing the tables. This retrieval is performed on a logical schema and a given context.

> **Note:** This tool works for tables journalized using simple or consistent set modes and cannot be executed in a command line with `startcmd`.

Usage

```
OdiRetrieveJournalData -LSCHEMA=<logical_schema>
-SUBSCRIBER_NAME=<subscriber_name>
(-TABLE_NAME=<table_name> | -CDC_SET_NAME=<cdc_set_name>)
[-CONTEXT=<context>] [-MAX_JRN_DATE=<to_date>]
```

Parameters

Parameters	Mandatory	Description
-LSCHEMA=<logical_schema>	Yes	Logical schema containing the journalized tables.
-TABLE_NAME=<table_name>	No	Journalized table name, mask, or list to check. This parameter accepts three formats:
		■ Table Name
		■ Table Name Mask: This mask selects the tables to poll. The mask is specified using the SQL LIKE syntax: the % symbol replaces an unspecified number of characters and the _ symbol acts as a wildcard.
		■ Table Names List: List of table names separated by commas. Masks as defined above are **not** allowed.
		Note that this option works only for tables in a model journalized in simple mode.
		This parameter cannot be used with -CDC_SET_NAME. It is mandatory if -CDC_SET_NAME is not set.
-CDC_SET_NAME=<cdc_set_name>	No	Name of the CDC set to update.
		Note that this option works only for tables in a model journalized in consistent mode.
		This parameter cannot be used with -TABLE_NAME. It is mandatory if -TABLE_NAME is not set.
-SUBSCRIBER_NAME=<subscriber_name>	Yes	Name of the subscriber for which the data is retrieved.
-CONTEXT=<context>	No	Context in which the logical schema will be resolved. If no context is specified, the execution context is used.
-MAX_JRN_DATE=<to_date>	No	Date (and time) until which the journalizing events are taken into account.

Examples

Retrieve for the CUSTOMERS table in the SALES_APPLICATION schema the journalizing events for the SALES_SYNC subscriber.

```
OdiRetrieveJournalData -LSCHEMA=SALES_APPLICATION
-TABLE_NAME=CUSTOMERS -SUBSCRIBER_NAME=SALES_SYNC
```

OdiReverseGetMetaData

Use this command to reverse-engineer metadata for the given model in the reverse tables using the JDBC driver capabilities. This command is typically preceded by OdiReverseResetTable and followed by OdiReverseSetMetaData.

> **Notes:**
>
> - This command uses the same technique as the standard reverse-engineering, and depends on the capabilities of the JDBC driver used.
>
> - The use of this command is restricted to DEVELOPMENT type Repositories because the metadata is not available on EXECUTION type Repositories.

Usage

```
OdiReverseGetMetaData -MODEL=<model_id>
```

Parameters

Parameters	Mandatory	Description
-MODEL=<model_id>	Yes	Model to reverse-engineer.

Examples

Reverse the RKM's current model.

```
OdiReverseGetMetaData -MODEL=<%=odiRef.getModel("ID")%>
```

OdiReverseManageShortcut

Use this command to define how to handle shortcuts when they are reverse-engineered in a model.

Usage

```
OdiReverseManageShortcut "-MODEL=<model_id>" "-MODE=MATERIALIZING_MODE"
```

Parameters

Parameters	Mandatory	Description
-MODEL=<model_id>	Yes	Global identifier of the model to be reversed.
-MODE=ALWAYS_MATERIALIZE|ALWAYS_SKIP|PROMPT	Yes	This parameter is supported only when a package or scenario is run in ODI Studio.
		This parameter accepts the following values:
		■ ALWAYS_MATERIALIZE: Conflicted shortcuts are always materialized and datastores are reversed (default).
		■ ALWAYS_SKIP: Conflicted shortcuts are always skipped and not reversed.
		■ PROMPT: The **Shortcut Conflict Detected** dialog is displayed. You can define how to handle conflicted shortcuts. Select **Materialize**, to materialize and reverse-engineer the conflicted datastore shortcut. Leave **Materialize** unselected, to skip the conflicted shortcuts. Unselected datastores are not reversed and the shortcut remains.

Examples

Reverse model 44fa5543-a378-4442-ac64-3dabab65ef98 in ALWAYS_MATERIALIZE mode.

```
OdiReverseManageShortcut -MODEL=44fa5543-a378-4442-ac64-3dabab65ef98 -MODE=ALWAYS_
MATERIALIZE
```

OdiReverseResetTable

Use this command to reset the content of reverse tables for a given model. This command is typically used at the beginning of a customized reverse-engineering process.

Usage

```
OdiReverseResetTable -MODEL=<model_id>
```

Parameters

Parameters	Mandatory	Description
-MODEL=<model_id>	Yes	Global identifier of the model to be reversed.

Examples

```
OdiReverseResetTable -MODEL=44fa5543-a378-4442-ac64-3dabab65ef98
```

OdiReverseSetMetaData

Use this command to integrate metadata from the reverse tables into the Repository for a given data model.

Usage

```
OdiReverseSetMetaData -MODEL=<model_id> [-USE_TABLE_NAME_FOR_UPDATE=<true|false>]
```

Parameters

Parameters	Mandatory	Description
-MODEL=<model_id>	Yes	Global identifier of the model to be reversed.
-USE_TABLE_NAME_FOR_ UPDATE=<true|false>	No	■ If true, the TABLE_NAME is used as an update key on the target tables. ■ If false (default), the RES_ NAME is used as the update key on the target tables.

Example

Reverse model 125880, using the TABLE_NAME as an update key on the target tables.

```
OdiReverseSetMetaData -MODEL=44fa5543-a378-4442-ac64-3dabab65ef98 -USE_TABLE_NAME_
FOR_UPDATE=true
```

OdiSAPALEClient and OdiSAPALEClient3

Use this command to generate SAP Internal Documents (IDoc) from XML source files and transfer these IDocs using ALE (Application Link Enabling) to a remote tRFC server (SAP R/3 server).

> **Note:** The OdiSAPALEClient tool supports SAP Java Connector 2.x. To use the SAP Java Connectors 3.x, use the OdiSAPALEClient3 tool.

Usage

```
OdiSAPALEClient -USER=<sap_logon> -ENCODED_PASSWORD=<password>
-GATEWAYHOST=<gateway_host> -SYSTEMNR=<system_number>
-MESSAGESERVERHOST=<message_server> -R3NAME=<system_name>
-APPLICATIONSERVERSGROUP=<group_name>
[-DIR=<directory>] [-FILE=<file>] [-CASESENS=<yes|no>]
[-MOVEDIR=<target_directory>] [-DELETE=<yes|no>] [-POOL_KEY=<pool_key>]
[-LANGUAGE=<language>] [-CLIENT=<client>] [-MAX_CONNECTIONS=<n>]
[-TRACE=<no|yes>]
```

Usage for OdiSAPALEClient3

```
OdiSAPALEClient3 -USER=<sap_logon> -ENCODED_PASSWORD=<password>
-GATEWAYHOST=<gateway_host> -SYSTEMNR=<system_number>
-MESSAGESERVERHOST=<message_server> -R3NAME=<system_name>
-APPLICATIONSERVERSGROUP=<group_name>
[-DIR=<directory>] [-FILE=<file>] [-CASESENS=<yes|no>]
[-MOVEDIR=<target_directory>] [-DELETE=<yes|no>] [-POOL_KEY=<pool_key>]
[-LANGUAGE=<language>] [-CLIENT=<client>] [-MAX_CONNECTIONS=<n>]
[-TRACE=<no|yes>]
```

Parameter

Parameters	Mandatory	Description
-USER=<sap_logon>	Yes	SAP logon. This user may be a system user.
-PASSWORD=<password>	Deprecated	SAP logon password. This command is deprecated. Use -ENCODED_PASSWORD instead.
-ENCODED_PASSWORD=<password>	Yes	SAP logon password, encrypted. The OS command encode <password> can be used to encrypt this password.
-GATEWAYHOST=<gateway_host>	No	Gateway host, mandatory if -MESSAGESERVERHOST is not specified.
-SYSTEMNR=<system_number>	No	SAP system number, mandatory if -GATEWAYHOST is used. The SAP system number enables the SAP load balancing feature.
-MESSAGESERVERHOST=<message_server>	No	Message server host name, mandatory if -GATEWAYHOST is not specified. If -GATEWAYHOST and -MESSAGESERVERHOST are both specified, -MESSAGESERVERHOST is used.

Parameters	Mandatory	Description
-R3NAME=<system_name>	No	Name of the SAP system (r3name), mandatory if -MESSAGESERVERHOST is used.
-APPLICATIONSERVERSGROUP=<group_name>	No	Application servers group name, mandatory if -MESSAGESERVERHOST is used.
-DIR=<directory>	No	XML source file directory. This parameter is taken into account if -FILE is not specified. At least one of the -DIR or -FILE parameters must be specified.
-FILE=<file>	No	Name of the source XML file. If this parameter is omitted, all files in -DIR are processed. At least one of the -DIR or -FILE parameters must be specified.
-CASESENS=<yes|no>	No	Indicates if the source file names are case-sensitive. The default value is No.
-MOVEDIR=<target_directory>	No	If this parameter is specified, the source files are moved to this directory after being processed.
-DELETE=<yes|no>	No	Deletes the source files after their processing. The default value is Yes.
-POOL_KEY=<pool_key>	No	Name of the connection pool. The default value is ODI.
-LANGUAGE=<language>	No	Language code used for error messages. The default value is EN.
-CLIENT=<client>	No	Client identifier. The default value is 001.
-MAX_CONNECTIONS=<n>	No	Maximum number of connections in the pool. The default value is 3.
-TRACE=<no|yes>	No	The generated IDoc files are archived in the source file directory. If the source files are moved (-MOVEDIR parameter), the generated IDocs are also moved. The default value is No.

Examples

Process all files in the /sap directory and send them as IDocs to the SAP server. The original XML and generated files are stored in the /log directory after processing.

```
OdiSAPALEClient -USER=ODI -ENCODED_PASSWORD=xxx -SYSTEMNR=002
-GATEWAYHOST=GW001 -DIR=/sap -MOVEDIR=/log -TRACE=yes
```

OdiSAPALEServer and OdiSAPALEServer3

Use this command to start a tRFC listener to receive SAP IDocs transferred using ALE (Application Link Enabling). This listener transforms incoming IDocs into XML files in a given directory.

> **Note:** The OdiSAPALEServer tool supports SAP Java Connector 2.x. To use the SAP Java Connectors 3.x, use the OdiSAPALEServer3 tool.

Usage

```
OdiSAPALEServer -USER=<sap_logon> -ENCODED_PASSWORD=<password>
-GATEWAYHOST=<gateway_host> -SYSTEMNR=<system_number>
-GATEWAYNAME=<gateway_name> -PROGRAMID=<program_id> -DIR=<target_directory>
[-TIMEOUT=<n>] [-POOL_KEY=<pool_key>] [-LANGUAGE=<Language>]
[-CLIENT=<client>] [-MAX_CONNECTIONS=<n>]
[-INTERREQUESTTIMEOUT=<n>] [-MAXREQUEST=<n>] [-TRACE=<no|yes>]
```

Usage of OdiSAPALEServer3

```
OdiSAPALEServer3 -USER=<sap_logon> -ENCODED_PASSWORD=<password>
-GATEWAYHOST=<gateway_host> -SYSTEMNR=<system_number>
-GATEWAYNAME=<gateway_name> -PROGRAMID=<program_id> -DIR=<target_directory>
[-TIMEOUT=<n>] [-POOL_KEY=<pool_key>] [-LANGUAGE=<Language>]
[-CLIENT=<client>] [-MAX_CONNECTIONS=<n>]
[-INTERREQUESTTIMEOUT=<n>] [-MAXREQUEST=<n>] [-TRACE=<no|yes>]
```

Parameters

Parameters	Mandatory	Description
-USER=<UserName>	Yes	SAP logon. This user may be a system user.
-ENCODED_PASSWORD=<password>	Yes	SAP logon password, encrypted. The system command `encode <password>` can be used to encrypt this password.
-GATEWAYHOST=<gateway_host>	Yes	Gateway host.
-SYSTEMNR=<system_number>	Yes	SAP system number.
-GATEWAYNAME=<gateway_name>	Yes	Gateway name.
-PROGRAMID=<program_id>	Yes	The program ID. External name used by the tRFC server.
-DIR=<target_directory>	Yes	Directory in which the target XML files are stored. These files are named `<IDOC Number>.xml`, and are located in subdirectories named after the IDoc type. The default is `./FromSAP`.
-POOL_KEY=<pool_key>	Yes	Name of the connection pool. The default value is `ODI`.
-LANG=<language>	Yes	Language code used for error messages. The default value is `EN`.
-CLIENT=<client>	Yes	SAP client identifier. The default value is `001`.

Parameters	Mandatory	Description
-TIMEOUT=<n>	No	Life span in milliseconds for the server. At the end of this period the server stops automatically. If this timeout is set to 0, the server life span is infinite. The default value is 0.
-MAX_CONNECTIONS=<n>	Yes	Maximum number of connections allowed for the pool of connections. The default value is 3.
-INTERREQUESTTIMEOUT=<n>	No	If no IDOC is received during an interval of n milliseconds, the listener stops. If this timeout is set to 0, the timeout is infinite. The default value is 0.
-MAXREQUEST=<n>	No	Maximum number of requests after which the listener stops. If this parameter is set to 0, the server expects an infinite number of requests. The default value is 0.
		Note: If -TIMEOUT, -INTERREQUESTTIMEOUT, and -MAXREQUEST are set to 0 or left empty, then -MAXREQUEST automatically takes the value 1.
-TRACE=<no|yes>	No	Activate the debug trace. The default value is No.
	No	Must match the RFC destination in SAP. Verify that the Unicode setting in SAP transaction SM59 matches this parameter.
		Note: Applies to OdiSAPALEServer3 only.

Examples

Wait for 2 IDoc files and generate the target XML files in the /temp directory.

```
OdiSAPALEServer -POOL_KEY=ODI -MAX_CONNECTIONS=3 -CLIENT=001
-USER=ODI -ENCODED_PASSWORD=xxx -LANGUAGE=EN
-GATEWAYHOST=SAP001 -SYSTEMNR=002 -GATEWAYNAME=GW001
-PROGRAMID=ODI01 -DIR=/tmp -MAXREQUEST=2
```

OdiScpGet

Use this command to download a file from an SSH server.

Usage

```
OdiScpGet -HOST=<ssh server host name> -USER=<ssh user>
[-PASSWORD=<ssh user password>] -REMOTE_DIR=<remote dir on ssh host>
[-REMOTE_FILE=<file name under the REMOTE_DIR>] -LOCAL_DIR=<local dir>
[-LOCAL_FILE=<file name under the LOCAL_DIR>]
[-TIMEOUT=<time in seconds>]
[-IDENTITY_FILE=<full path to the private key file of the user>]
[-KNOWNHOSTS_FILE=<full path to known hosts file>] [COMPRESSION=<yes|no>]
[-STRICT_HOSTKEY_CHECKING=<yes|no>] [-PROXY_HOST=<proxy server host name>]
[-PROXY_PORT=<proxy server port>] [-PROXY_TYPE=<HTTP|SOCKS5>]
```

Parameters

Parameters	Mandatory	Description
-HOST=<ssh server host name>	Yes	Host name of the SSH server.
-USER=<ssh user>	Yes	User on the SSH server.
-PASSWORD=<ssh user password>	No	The password of the SSH user or the passphrase of the password-protected identity file. If the -IDENTITY_FILE argument is provided, this value is used as the passphrase for the password-protected private key file. If public key authentication fails, it falls back to the normal user password authentication.
-REMOTE_DIR=<dir on remote SSH>	Yes	Directory path on the remote SSH host.
-REMOTE_FILE=<file name under -REMOTE DIR>	No	File name under the directory specified in the -REMOTE_DIR argument. Note that all subdirectories matching the remote file name will also be transferred to the local folder. If this argument is missing, the file is copied with the -LOCAL_FILE file name. If -LOCAL_FILE is also missing, the -LOCAL_DIR is copied recursively to the -REMOTE_DIR.
-LOCAL_DIR=<local dir path>	Yes	Directory path on the local machine.

Parameters	Mandatory	Description
-LOCAL_FILE=<local file>	No	File name under the directory specified in the -LOCAL_DIR argument. If this argument is missing, all files and directories under -LOCAL_DIR are copied recursively to the -REMOTE_DIR. To filter the files to be copied use * to specify the generic characters. Examples: ■ *.log (all files with the log extension) ■ arch_*.lst (all files starting with arch_ and with the extension lst)
-IDENTITY_FILE=<full path to the private key file of the user>	No	Private key file of the local user. If this argument is specified, public key authentication is performed. The -PASSWORD argument is used as the password for the password-protected private key file. If authentication fails, it falls back to normal user password authentication.
-KNOWNHOSTS_FILE=<full path to the known hosts file on the local machine>	No	Full path to the known hosts file on the local machine. The known hosts file contains the host keys of all remote machines the user trusts. If this argument is missing, the <user home dir>/.ssh/known_hosts file is used as the known hosts file if it exists.
-COMPRESSION=<yes|no>	No	If set to Yes, data compression is used. The default value is No.
-STRICT_HOSTKEY_CHECKING=<yes|no>	No	If set to Yes (default), strict host key checking is performed and authentication fails if the remote SSH host key is not present in the known hosts file specified in -KNOWNHOSTS_FILE.
-PROXY_HOST=<proxy server host name>	No	Host name of the proxy server to be used for the connection.
-PROXY_PORT=<proxy server port>	No	Port number of the proxy server.
-PROXY_TYPE=<HTTP|SOCKS5>	No	Type of proxy server you are connecting to, HTTP or SOCKS5.
-TIMEOUT=<time in seconds>	No	Time in seconds after which the socket connection times out.

Examples

Copy the remote directory /test_copy555 on the SSH server recursively to the local directory C:\temp\test_copy.

```
OdiScpGet -HOST=machine.oracle.com -USER=test_ftp -PASSWORD=<password>
 -LOCAL_DIR=C:\temp\test_copy -REMOTE_DIR=/test_copy555
```

Copy all files matching the Sales*.txt pattern under the remote directory / on the SSH server to the local directory C:\temp\.

```
OdiScpGet -HOST=machine.oracle.com -USER=test_ftp -PASSWORD=<password>
 -LOCAL_DIR=C:\temp -REMOTE_FILE=Sales*.txt -REMOTE_DIR=/
```

Copy the `Sales1.txt` file under the remote directory `/` on the SSH server to the local directory `C:\temp\` as a `Sample1.txt` file.

```
OdiScpGet -HOST=machine.oracle.com -USER=test_ftp -PASSWORD=<password>
 -REMOTE_DIR=/ REMOTE_FILE=Sales1.txt -LOCAL_DIR=C:\temp
-LOCAL_FILE=Sample1.txt
```

Copy the `Sales1.txt` file under the remote directory `/` on the SSH server to the local directory `C:\temp\` as a `Sample1.txt` file. Public key authentication is performed by providing the path to the identity file and the path to the known hosts file.

```
OdiScpGet -HOST=machine.oracle.com -USER=test_ftp -PASSWORD=<password>
-REMOTE_DIR=/ -REMOTE_FILE=Sales1.txt -LOCAL_DIR=C:\temp
-LOCAL_FILE=Sample1.txt -IDENTITY_FILE=C:\Documents and Settings\username\.ssh\id_
dsa -KNOWNHOSTS_FILE=C:\Documents and Settings\username\.ssh\known_hosts
```

Copy the `Sales1.txt` file under the remote directory `/` on the SSH server to the local directory `C:\temp\` as a `Sample1.txt` file. Public key authentication is performed by providing the path to the identity file. All hosts are trusted by passing the No value to the `-STRICT_HOSTKEY_CHECKING` parameter.

```
OdiScpGet -HOST=machine.oracle.com -USER=test_ftp -PASSWORD=<password>
-REMOTE_DIR=/ -REMOTE_FILE=Sales1.txt -LOCAL_DIR=C:\temp -LOCAL_FILE=Sample1.txt
-IDENTITY_FILE=C:\Documents and Settings\username\.ssh\id_dsa
-STRICT_HOSTKEY_CHECKING=NO
```

OdiScpPut

Use this command to upload a file to an SSH server.

Usage

```
OdiScpPut -HOST=<SSH server host name> -USER=<SSH user>
[-PASSWORD=<SSH user password>] -LOCAL_DIR=<local dir>
[-LOCAL_FILE=<file name under the LOCAL_DIR>] -REMOTE_DIR=<remote dir on ssh host>
[-REMOTE_FILE=<file name under the REMOTE_DIR>]
[-TIMEOUT=<time in seconds>]
[-IDENTITY_FILE=<full path to the private key file of the user>]
[-KNOWNHOSTS_FILE=<full path to known hosts file>] [-COMPRESSION=<yes|no>]
[-STRICT_HOSTKEY_CHECKING=<yes|no>] [<-PROXY_HOST=<proxy server host name>]
[-PROXY_PORT=<proxy server port>] [-PROXY_TYPE=<HTTP|SOCKS5>]
```

Parameters

Parameters	Mandatory	Description
-HOST=<host name of the SSH server>	Yes	Host name of the SSH server.
-USER=<host name of the SSH user>	Yes	User on the SSH server.
-PASSWORD=<password of the SSH user>	No	Password of the SSH user or the passphrase of the password-protected identity file. If the -IDENTITY_FILE argument is provided, this value is used as the passphrase for the password-protected private key file. If public key authentication fails, it falls back to the normal user password authentication.
-REMOTE_DIR=<dir on remote SSH	Yes	Directory path on the remote SSH host.
-REMOTE_FILE=<file name under -REMOTE DIR>	No	File name under the directory specified in the -REMOTE_DIR argument. If this argument is missing, the file is copied with the -LOCAL_FILE file name. If the -LOCAL_FILE argument is also missing, the -LOCAL_DIR is copied recursively to the -REMOTE_DIR.
-LOCAL_DIR=<local dir path>	Yes	Directory path on the local machine.
-LOCAL_FILE=<local file>	No	File name under the directory specified in the -LOCAL_DIR argument. If this argument is missing, all files and directories under the -LOCAL_DIR are copied recursively to the -REMOTE_DIR. To filter the files to be copied use * to specify the generic characters. Examples: ■ *.log (all files with the log extension) ■ arch_*.lst (all files starting with arch_ and with the extension lst)

Parameters	Mandatory	Description
-IDENTITY_FILE=<full path to the private key file of the user>	No	Private key file of the local user. If this argument is specified, public key authentication is performed. The -PASSWORD argument is used as the password for the password-protected private key file. If authentication fails, it falls back to normal user password authentication.
-KNOWNHOSTS_FILE=<full path to the known hosts file on the local machine>	No	Full path to the known hosts file on the local machine. The known hosts file contains the host keys of all remote machines the user trusts. If this argument is missing, the <user home dir>/.ssh/known_hosts file is used as the known hosts file if it exists.
-COMPRESSION=<yes|no>	No	If set to Yes, data compression is used. The default value is No.
-STRICT_HOSTKEY_CHECKING=<yes|no>	No	If set to Yes (default), strict host key checking is performed and authentication fails if the remote SSH host key is not present in the known hosts file specified in -KNOWNHOSTS_FILE.
-PROXY_HOST=<proxy server host name>	No	Host name of the proxy server to be used for the connection.
-PROXY_PORT=<proxy server port>	No	Port number of the proxy server.
-PROXY_TYPE=<HTTP|SOCKS5>	No	Type of proxy server you are connecting to, HTTP or SOCKS5.
-TIMEOUT=<timeout value>	No	Time in seconds after which the socket connection times out.

Examples

Copy the local directory `C:\temp\test_copy` recursively to the remote directory `/test_copy555` on the SSH server.

```
OdiScpPut -HOST=machine.oracle.com -USER=test_ftp -PASSWORD=<password> -LOCAL_
DIR=C:\temp\test_copy -REMOTE_DIR=/test_copy555
```
Copy all files matching the `Sales*.txt` pattern under the local directory `C:\temp\` to the remote directory `/` on the SSH server.

```
OdiScpPut -HOST=machine.oracle.com -USER=test_ftp -PASSWORD=<password> -LOCAL_
DIR=C:\temp -LOCAL_FILE=Sales*.txt -REMOTE_DIR=/
```
Copy the `Sales1.txt` file under the local directory `C:\temp\` to the remote directory `/` on the SSH server as a `Sample1.txt` file.

```
OdiScpPut -HOST=machine.oracle.com -USER=test_ftp -PASSWORD=<password>
-LOCAL_DIR=C:\temp -LOCAL_FILE=Sales1.txt -REMOTE_DIR=/ -REMOTE_FILE=Sample1.txt
```
Copy the `Sales1.txt` file under the local directory `C:\temp\` to the remote directory `/` on the SSH server as a `Sample1.txt` file. Public key authentication is performed by providing the path to the identity file and the path to the known hosts file.

```
OdiScpPut -HOST=machine.oracle.com -USER=test_ftp
-PASSWORD=<password> -LOCAL_DIR=C:\temp -LOCAL_FILE=Sales1.txt
-REMOTE_DIR=/ -REMOTE_FILE=Sample1.txt
-IDENTITY_FILE=C:\Documents and Settings\username\.ssh\id_dsa
-KNOWNHOSTS_FILE=C:\Documents and Settings\username\.ssh\known_hosts
```

Copy the `Sales1.txt` file under the local directory `C:\temp\` to the remote directory `/` on the SSH server as a `Sample1.txt` file. Public key authentication is performed by providing the path to the identity file. All hosts are trusted by passing the No value to the `-STRICT_HOSTKEY_CHECKING` parameter.

```
OdiScpPut -HOST=machine.oracle.com -USER=test_ftp
-PASSWORD=<password> -LOCAL_DIR=C:\temp -LOCAL_FILE=Sales1.txt
-REMOTE_DIR=/ -REMOTE_FILE=Sample1.txt
-IDENTITY_FILE=C:\Documents and Settings\username\.ssh\id_dsa
-STRICT_HOSTKEY_CHECKING=NO
```

OdiSendMail

Use this command to send an email to an SMTP server.

Usage

```
OdiSendMail -MAILHOST=<mail_host> -FROM=<from_user> -TO=<address_list>
[-CC=<address_list>] [-BCC=<address_list>] [-SUBJECT=<subject>]
[-ATTACH=<file_path>]* [-MSGBODY=<message_body> | CR/LF<message_body>]
```

Parameters

Parameters	Mandatory	Description
-MAILHOST=<mail_host>	Yes	IP address of the SMTP server.
-FROM=<from_user>	Yes	Address of the sender of the message.
		Example: support@mycompany.com
		To send the external name of the sender, the following notation can be used:
		"-FROM=Support center <support@mycompany.com>"
-TO=<address_list>	Yes	List of email addresses of the recipients, separated by commas.
		Example:
		"-TO=sales@mycompany.com, support@mycompany.com"
-CC=<address_list>	No	List of e-mail addresses of the CC-ed recipients, separated by commas.
		Example:
		"-CC=info@mycompany.com"
-BCC=<address_list>	No	List of email-addresses of the BCC-ed recipients, separated by commas.
		Example:
		"-BCC=manager@mycompany.com"
-SUBJECT=<subject>	No	Object (subject) of the message.
-ATTACH=<file_path>	No	Path of the file to join to the message, relative to the execution agent. To join several files, repeat -ATTACH.
		Example: Attach the files .profile and .cshrc to the mail:
		-ATTACH=/home/usr/.profile -ATTACH=/home/usr/.cshrc
CR/LF <message_body> or -MSGBODY=<message_body>	No	Message body (text). This text can be typed on the line following the OdiSendMail command (a carriage return - CR/LF - indicates the beginning of the mail body), or can be defined with the -MSGBODY parameter. The -MSGBODY parameter should be used when calling this Oracle Data Integrator command from an OS command line.

Parameters	Mandatory	Description
PORT	No	The Port number of the mail server. Default is @ default port used by javax.mail.
PROTOCOL	No	E-mail protocol. It can be SMTP or POP3. Default is SMTP.
AUTH	No	The authentication to be used. The values are YES or NO. Default is NO.
AUTHMECHANISMS	No	The authentication mechanism. The values are PLAIN, LOGIN or DIGEST-MD5.
USER	No	User for authentication. Only if authentication is used.
PASS	No	Password for authentication. Only if authentication is used.

Examples

```
OdiSendMail -MAILHOST=mail.mymail.com "-FROM=Application Oracle Data
Integrator<odi@mymail.com>" -TO=admin@mymail.com "-SUBJECT=Execution OK"
-ATTACH=C:\log\job.log -ATTACH=C:\log\job.bad
Hello Administrator !
Your process finished successfully. Attached are your files.
Have a nice day!
Oracle Data Integrator.
```

OdiSftp

Use this command to connect to an SSH server with an enabled SFTP subsystem and perform standard FTP commands on the remote system. Trace from the script is recorded against the Execution Details of the task representing the OdiSftp step in Operator Navigator.

Usage

```
OdiSftp -HOST=<ssh server host name> -USER=<ssh user>
[-PASSWORD=<ssh user password>] -LOCAL_DIR=<local dir>
-REMOTE_DIR=<remote dir on ssh host> [-PASSIVE_MODE=<yes|no>]
[-TIMEOUT=<time in seconds>] [-IDENTITY_FILE=<full path to private key file of
user>] [-KNOWNHOSTS_FILE=<full path to known hosts file on local machine>]
[-COMPRESSION=<yes|no>] [-STRICT_HOSTKEY_CHECKING=<yes|no>]
[-PROXY_HOST=<proxy server host name>] [-PROXY_PORT=<proxy server port>]
 [-PROXY_TYPE=<HTTP|SOCKS5>] [STOP_ON_FTP_ERROR=<yes|no>]
-COMMAND=<command>
```

Parameters

Parameters	Mandatory	Description
-HOST=<ssh server host name>	Yes	Host name of the SSH server.
-USER=<ssh user>	Yes	User on the SSH server.
-PASSWORD=<ssh user password>	No	Password of the SSH user.
-LOCAL_DIR=<local dir>	Yes	Directory path on the local machine.
-REMOTE_DIR=<remote dir on ssh host>	Yes	Directory path on the remote SSH host.
-TIMEOUT=<time in seconds>	No	Time in seconds after which the socket connection times out.
-IDENTITY_FILE=<full path to private key file of user>	No	Private key file of the local user. If specified, public key authentication is performed. The -PASSWORD argument is used as the password for the password-protected private key file. If authentication fails, normal user password authentication is performed.
-KNOWNHOSTS_FILE=<full path to known hosts file on local machine>	No	Full path to the known hosts file on the local machine. The known hosts file contains host keys for all remote machines trusted by the user. If this argument is missing, the <user home dir>/.ssh/known_hosts file is used as the known hosts file if it exists.
-COMPRESSION=<yes|no>	No	If set to Yes, data compression is used. The default value is No.
-STRICT_HOSTKEY_CHECKING=<yes|no>	No	If set to Yes (default), strict host key checking is performed and authentication fails if the remote SSH host key is not present in the known hosts file specified in -KNOWNHOSTS_FILE.

Parameters	Mandatory	Description
-PROXY_HOST=<proxy server host name>	No	Host name of the proxy server to be used for the connection.
-PROXY_PORT=<proxy server port>	No	Port number of the proxy server.
-PROXY_TYPE<HTTP|SOCKS5>	No	Type of proxy server you are connecting to, HTTP or SOCKS5.
STOP_ON_FTP_ERROR=<yes|no>	No	If set to Yes (default), the step stops with an Error status if an error occurs rather than running to completion.
-COMMAND=<command>	Yes	Raw FTP command to execute. For a multiline command, pass the whole command as raw text after the OdiSftp line without the -COMMAND parameter. Supported commands: APPE, CDUP, CWD, DELE, LIST, MKD, NLST, PWD, QUIT, RETR, RMD, RNFR, RNTO, SIZE, STOR

Examples

Execute a script on a remote host that changes directory into a directory, deletes a file from the directory, changes directory into the parent directory, and removes the directory.

```
OdiSftp -HOST=machine.oracle.com -USER=odiftpuser -PASSWORD=<password>
-LOCAL_DIR=/tmp -REMOTE_DIR=/tmp -STOP_ON_FTP_ERROR=No
CWD /tmp/ftpToolDir1
DELE ftpToolFile
CDUP
RMD ftpToolDir1
```

OdiSftpGet

Use this command to download a file from an SSH server with an enabled SFTP subsystem.

Usage

```
OdiSftpGet -HOST=<ssh server host name> -USER=<ssh user>
[-PASSWORD=<ssh user password>] -REMOTE_DIR=<remote dir on ssh host>
[-REMOTE_FILE=<file name under REMOTE_DIR>] -LOCAL_DIR=<local dir>
[-LOCAL_FILE=<file name under LOCAL_DIR>]
[-TIMEOUT=<time in seconds>]
[-IDENTITY_FILE=<full path to private key file of user>]
[-KNOWNHOSTS_FILE=<full path to known hosts file on local machine>]
[-COMPRESSION=<yes|no>] [-STRICT_HOSTKEY_CHECKING=<yes|no>]
[-PROXY_HOST=<proxy server host name>] [-PROXY_PORT=<proxy server port>]
[-PROXY_TYPE=<HTTP|SOCKS5>]
```

> **Note:** If a Local or Remote file name needs to have % as part of its name, %25 needs to be passed instead of just %.
>
> %25 will resolve automatically to %.
>
> For example, if file name needs to be temp%result, it should be passed as REMOTE_FILE=temp%25result or -LOCAL_FILE=temp%25result.

Parameters

Parameters	Mandatory	Description
-HOST=<ssh server host name>	Yes	Host name of the SSH server.
		You can add the port number to the host name by prefixing it with a colon (:). For example: machine.oracle.com:25
		If no port is specified, port 22 is used by default.
-USER=<ssh user>	Yes	User on the SSH server.
-PASSWORD=<ssh user password>	No	Password of the SSH user.
-REMOTE_DIR=<remote dir on ssh host>	Yes	Directory path on the remote SSH host.
-REMOTE_FILE=<file name under -REMOTE DIR>	No	File name under the directory specified in the -REMOTE_DIR argument. If this argument is missing, the file is copied with the -LOCAL_FILE file name. If the -LOCAL_FILE argument is also missing, the -LOCAL_DIR is copied recursively to the -REMOTE_DIR.
-LOCAL_DIR=<local dir>	Yes	Directory path on the local machine.

Parameters	Mandatory	Description
-LOCAL_FILE=<file name under LOCAL_DIR>	No	File name under the directory specified in the -LOCAL_DIR argument. If this argument is missing, all files and directories under the -LOCAL_DIR are copied recursively to the -REMOTE_DIR.
		To filter the files to be copied, use * to specify the generic characters.
		Examples:
		■ *.log (all files with the log extension)
		■ arch_*.lst (all files starting with arch_ and with the extension lst)
-IDENTITY_FILE=<full path to private key file of user>	No	Private key file of the local user. If this argument is specified, public key authentication is performed. The -PASSWORD argument is used as the password for the password-protected private key file. If authentication fails, it falls back to normal user password authentication.
-KNOWNHOSTS_FILE=<full path to known hosts file on local machine>	No	The full path to the known hosts file on the local machine. The known hosts file contains the host keys of all remote machines the user trusts. If this argument is missing, the <user home dir>/.ssh/known_hosts file is used as the known hosts file if it exists.
-COMPRESSION=<yes|no>	No	If set to Yes, data compression is used. The default value is No.
-STRICT_HOSTKEY_CHECKING=<yes|no>	No	If set to Yes (default), strict host key checking is performed and authentication fails if the remote SSH host key is not present in the known hosts file specified in -KNOWNHOSTS_FILE.
-PROXY_HOST=<proxy server host name>	No	Host name of the proxy server to be used for the connection.
-PROXY_PORT=<proxy server port>	No	Port number of the proxy server.
-PROXY_TYPE=<HTTP|SOCKS5>	No	Type of proxy server you are connecting to, HTTP or SOCKS5.
-TIMEOUT=<time in seconds>	No	Time in seconds after which the socket connection times out.

Examples

Copy the remote directory /test_copy555 on the SSH server recursively to the local directory C:\temp\test_copy.

```
OdiSftpGet -HOST=machine.oracle.com -USER=test_ftp -PASSWORD=<password> -LOCAL_
DIR=C:\temp\test_copy -REMOTE_DIR=/test_copy555
```
Copy all files matching the Sales*.txt pattern under the remote directory / on the SSH server to the local directory C:\temp\.

```
OdiSftpGet -HOST=machine.oracle.com -USER=test_ftp -PASSWORD=<password> -LOCAL_
DIR=C:\temp -REMOTE_FILE=Sales*.txt -REMOTE_DIR=/
```

Copy the `Sales1.txt` file under the remote directory / on the SSH server to the local directory `C:\temp\` as a `Sample1.txt` file.

```
OdiSftpGet -HOST=machine.oracle.com -USER=test_ftp -PASSWORD=<password> -REMOTE_
DIR=/ -LOCAL_FILE=Sales1.txt -LOCAL_DIR=C:\temp -LOCAL_FILE=Sample1.txt
```
Copy the `Sales1.txt` file under the remote directory / on the SSH server to the local directory `C:\temp\` as a `Sample1.txt` file. Public key authentication is performed by providing the path to the identity file and the path to the known hosts file.

```
OdiSftpGet -HOST=machine.oracle.com -USER=test_ftp -PASSWORD=<password>
-REMOTE_DIR=/ -REMOTE_FILE=Sales1.txt -LOCAL_DIR=C:\temp -LOCAL_FILE=Sample1.txt
-IDENTITY_FILE=C:\Documents and Settings\username\.ssh\id_dsa
-KNOWNHOSTS_FILE=C:\Documents and Settings\username\.ssh\known_hosts
```
Copy the `Sales1.txt` file under the remote directory / on the SSH server to the local directory `C:\temp\` as a `Sample1.txt` file. Public key authentication is performed by providing the path to the identity file. All hosts are trusted by passing the No value to the `-STRICT_HOSTKEY_CHECKING` parameter.

```
OdiSftpGet -HOST=dev3 -USER=test_ftp -PASSWORD=<password>
-REMOTE_DIR=/ -REMOTE_FILE=Sales1.txt -LOCAL_DIR=C:\temp -LOCAL_FILE=Sample1.txt
-IDENTITY_FILE=C:\Documents and Settings\username\.ssh\id_dsa
-STRICT_HOSTKEY_CHECKING=NO
```

OdiSftpPut

Use this command to upload a file to an SSH server with the SFTP subsystem enabled.

Usage

```
OdiSftpPut -HOST=<ssh server host name> -USER=<ssh user>
[-PASSWORD=<ssh user password>] -LOCAL_DIR=<local dir>
[-LOCAL_FILE=<file name under LOCAL_DIR>] -REMOTE_DIR=<remote dir on ssh host>
[-REMOTE_FILE=<file name under REMOTE_DIR>]
[-TIMEOUT=<time in seconds>]
[-IDENTITY_FILE=<full path to private key file of user>]
[-KNOWNHOSTS_FILE=<full path to known hosts file on local machine>]
[-COMPRESSION=<yes|no>] [-STRICT_HOSTKEY_CHECKING=<yes|no>]
[-PROXY_HOST=<proxy server host name>] [-PROXY_PORT=<proxy server port>]
[-PROXY_TYPE=<HTTP|SOCKS5>]
```

> **Note:** If a Local or Remote file name needs to have % as part of its name, %25 needs to be passed instead of just %.
>
> %25 will resolve automatically to %.
>
> For example, if file name needs to be `temp%result`, it should be passed as `REMOTE_FILE=temp%25result` or `-LOCAL_FILE=temp%25result`.

Parameter

Parameters	Mandatory	Description
-HOST=<ssh server host name>	Yes	Host name of the SSH server.
		You can add the port number to the host name by prefixing it with a colon (:). For example: `machine.oracle.com:25`
		If no port is specified, port 22 is used by default.
-USER=<ssh user>	Yes	User on the SSH server.
-PASSWORD=<ssh user password>	No	Password of the SSH user or the passphrase of the password-protected identity file. If the -IDENTITY_FILE argument is provided, this value is used as the passphrase for the password-protected private key file. If public key authentication fails, it falls back to normal user password authentication.
-REMOTE_DIR=<remote dir on ssh host	Yes	Directory path on the remote SSH host.
-REMOTE_FILE=<file name under -REMOTE DIR>	No	File name under the directory specified in the -REMOTE_DIR argument. If this argument is missing, the file is copied with the -LOCAL_FILE file name. If the -LOCAL_FILE argument is also missing, the -LOCAL_DIR is copied recursively to the -REMOTE_DIR.
-LOCAL_DIR=<local dir>	Yes	Directory path on the local machine.

Parameters	Mandatory	Description
-LOCAL_FILE=<file name under LOCAL_DIR>	No	File name under the directory specified in the -LOCAL_DIR argument. If this argument is missing, all files and directories under the -LOCAL_DIR is copied recursively to the -REMOTE_DIR.
		To filter the files to be copied, use * to specify the generic characters.
		Examples:
		■ *.log (all files with the log extension)
		■ arch_*.lst (all files starting with arch_ and with the extension lst)
-IDENTITY_FILE=<full path to private key file of user>	No	Private key file of the local user. If this argument is specified, public key authentication is performed. The -PASSWORD argument is used as the password for the password-protected private key file. If authentication fails, it falls back to normal user password authentication.
-KNOWNHOSTS_FILE=<full path to known hosts file on local machine>	No	Full path to the known hosts file on the local machine. The known hosts file contains the host keys of all remote machines the user trusts. If this argument is missing, the <user home dir>/.ssh/known_hosts file is used as the known hosts file if it exists.
-COMPRESSION=<yes|no>	No	If set to Yes, data compression is used. The default value is No.
-STRICT_HOSTKEY_CHECKING=<yes|no>	No	If set to Yes (default), strict host key checking is performed and authentication fails if the remote SSH host key is not present in the known hosts file specified in -KNOWNHOSTS_FILE.
-PROXY_HOST=<proxy server host name>	No	Host name of the proxy server to be used for the connection.
-PROXY_PORT=<proxy server port>	No	Port number of the proxy server.
-PROXY_TYPE=<HTTP|SOCKS5>	No	Type of proxy server you are connecting to, HTTP or SOCKS5.
-TIMEOUT=<time in seconds>	No	Time in seconds after which the socket connection times out.

Examples

Copy the local directory C:\temp\test_copy recursively to the remote directory /test_copy555 on the SSH server.

```
OdiSftpPut -HOST=machine.oracle.com -USER=test_ftp -PASSWORD=<password> -LOCAL_
DIR=C:\temp\test_copy -REMOTE_DIR=/test_copy555
```
Copy all files matching the Sales*.txt pattern under the local directory C:\temp\ to the remote directory / on the SSH server.

```
OdiSftpPut -HOST=machine.oracle.com -USER=test_ftp -PASSWORD=<password> -LOCAL_
DIR=C:\temp -LOCAL_FILE=Sales*.txt -REMOTE_DIR=/
```

Copy the `Sales1.txt` file under the local directory `C:\temp\` to the remote directory `/` on the SSH server as a `Sample1.txt` file.

```
OdiSftpPut -HOST=machine.oracle.com -USER=test_ftp -PASSWORD=<password> -LOCAL_
DIR=C:\temp -LOCAL_FILE=Sales1.txt -REMOTE_DIR=/Sample1.txt
```

Copy the `Sales1.txt` file under the local directory `C:\temp\` to the remote directory `/` on the SSH server as a `Sample1.txt` file. Public key authentication is performed by providing the path to the identity file and the path to the known hosts file.

```
OdiSftpPut -HOST=machine.oracle.com -USER=test_ftp -PASSWORD=<password>
-LOCAL_DIR=C:\temp -LOCAL_FILE=Sales1.txt -REMOTE_DIR=/Sample1.txt
-IDENTITY_FILE=C:\Documents and Settings\username\.ssh\id_dsa
-KNOWNHOSTS_FILE=C:\Documents and Settings\username\.ssh\known_hosts
```

Copy the `Sales1.txt` file under the local directory `C:\temp\` to the remote directory `/` on the SSH server as a `Sample1.txt` file. Public key authentication is performed by providing the path to the identity file. All hosts are trusted by passing the No value to the `-STRICT_HOSTKEY_CHECKING` parameter.

```
OdiSftpPut -HOST=machine.oracle.com -USER=test_ftp -PASSWORD=<password>
-LOCAL_DIR=C:\temp -LOCAL_FILE=Sales1.txt -REMOTE_DIR=/Sample1.txt
-IDENTITY_FILE=C:\Documents and Settings\username\.ssh\id_dsa
-STRICT_HOSTKEY_CHECKING=NO
```

OdiSleep

Use this command to wait for `<delay>` milliseconds.

Usage

```
OdiSleep -DELAY=<delay>
```

Parameters

Parameters	Mandatory	Description
-DELAY=<delay>	Yes	Number of milliseconds to wait

Examples

```
OdiSleep -DELAY=5000
```

OdiSqlUnload

Use this command to write the result of a SQL query to a file.

This command executes the SQL query `<sql_query>` on the data server whose connection parameters are provided by `<driver>`, `<url>`, `<user>`, and `<encoded_pass>`. The resulting resultset is written to `<file_name>`.

Usage

```
OdiSqlUnload -FILE=<file_name> -DRIVER=<driver> -URL=<url> -USER=<user>
-PASS=<password> [-FILE_FORMAT=<file_format>] [-FIELD_SEP=<field_sep> |
-XFIELD_SEP=<field_sep>] [-ROW_SEP=<row_sep> | -XROW_SEP=<row_sep>]
[-DATE_FORMAT=<date_format>] [-CHARSET_ENCODING=<encoding>]
[-XML_CHARSET_ENCODING=<encoding>] [-FETCH_SIZE=<array_fetch_size>]
( CR/LF <sql_query> | -QUERY=<sql_query> | -QUERY_FILE=<sql_query_file> )
```

Parameters

Parameters	Mandatory	Description
-FILE=<file_name>	Yes	Full path to the output file, relative to the execution agent.
-DRIVER=<driver>	Yes	Name of the JDBC driver used to connect to the data server.
-URL=<url>	Yes	JDBC URL to the data server.
-USER=<user>	Yes	Login of the user on the data server that will be used to run the SQL query.
-PASS=<password>	Yes	Encrypted password for the login to the data server. This password can be encrypted with the system command encode <clear_text_ password>.
		Note that agent(.bat or .sh) is located in the /bin subdirectory of your Oracle Data Integrator installation directory.

Parameters	Mandatory	Description

-FILE_FORMAT=<file_format>	No	Specifies the file format with one of the following three values:
		
		■ `fixed`: Fixed size recording
		
		■ `variable`: Variable size recording
		
		■ `xml`: XML file
		
		If `<file_format>` is not specified, the format defaults to `variable`.
		
		If `<file_format>` is `xml`, the XML nodes generated have the following structure:
		
		`<TABLE>`
		
		`<ROW>`
		
		`<column_name>![CDATA[VALUE]]</column_name>`
		
		`<column_name>![CDATA[VALUE]]</column_name>`
		
		...
		
		`</ROW>`
		
		....
		
		`</TABLE>`

-FIELD_SEP=<field_sep>	No	Field separator character in ASCII format if `-FILE_FORMAT=variable`. The default `<field_sep>` is a tab character.
-XFIELD_SEP=<field_sep>	No	Field separator character in hexadecimal format if `-FILE_FORMAT=variable`. The default `<field_sep>` is a tab character.
-ROW_SEP=<row_sep>	No	Record separator character in ASCII format. The default `<row_sep>` is a Windows carriage return. For instance, the following values can be used: ■ UNIX: `-ROW_SEP=\n` ■ Windows: `-ROW_SEP=\r\n`
-XROW_SEP=<row_sep>	No	Record separator character in hexadecimal format. Example: `0A`.
-DATE_FORMAT=<date_format>	No	Output format used for date datatypes. This date format is specified using the Java date and time format patterns. For a list of these patterns, see: http://java.sun.com/j2se/1.4.2/docs/api/java/text/SimpleDateFormat.html.
-CHARSET_ENCODING=<encoding>	No	Target file encoding. The default value is `ISO-8859-1`. For the list of supported encodings, see: http://java.sun.com/j2se/1.4.2/docs/guide/intl/encoding.doc.html
-XML_CHARSET_ENCODING=<encoding>	No	Encoding specified in the XML file, in the tag `<?xml version="1.0" encoding="ISO-8859-1"?>`. The default value is `ISO-8859-1`. For the list of supported encodings, see: http://java.sun.com/j2se/1.4.2/docs/guide/intl/encoding.doc.html
-FETCH_SIZE=<array_fetch_size>	No	Number of rows (records read) requested by Oracle Data Integrator in each communication with the data server.

Parameters	Mandatory	Description
`-CR/LF=<sql_query>` | `-QUERY=<sql_query>` | `-QUERY_ FILE=<sql_query_file>`	Yes	SQL query to execute on the data server. The query must be a `SELECT` statement or a call to a stored procedure returning a valid recordset. This query can be entered on the line following the OdiSqlUnload command (a carriage return - `CR/LF` - indicates the beginning of the query). The query can be provided within the `-QUERY` parameter, or stored in a file specified with the `-QUERY_FILE` parameter. The `-QUERY` or `-QUERY_ FILE` parameters must be used when calling this command from an OS command line.

Examples

Generate the file `C:\temp\clients.csv` separated by `;` containing the result of the query on the `Customers` table.

```
OdiSqlUnload -FILE=C:\temp\clients.csv -DRIVER=sun.jdbc.odbc.JdbcOdbcDriver
-URL=jdbc:odbc:NORTHWIND_ODBC -USER=sa
-PASS=NFNEKKNGGJHAHBHDHEHJDBGBGFDGGH -FIELD_SEP=;
"-DATE_FORMAT=dd/MM/yyyy hh:mm:ss"

select cust_id, cust_name, cust_creation_date from Northwind.dbo.Customers
```

OdiRetrieveHadoopLog

Use this command to retrieve log information from executions in an Oozie execution agent.

Usage

```
OdiRetrieveHadoopLog [-SESSION_LIST=<session-ids>] -POLLINT=<poll>
-TIMEOUT=<timeout>
```

Parameters

Parameters	Mandatory	Description
-SESSION_LIST=<session-ids>	No	A comma separated list of sessions IDs to be retrieved. If blank, all Oozie sessions currently running will be retrieved.
-POLLINT=<poll>	No	The length of time between each the instance that the log data is retrieved. Can be in secs (s), mins (m), hours (h), days (d) or years (y). If zero, the log data will be retrieved once and then the tool will end.
-TIMEOUT=<timeout>	No	The maximum period of time that the tool will execute for. Can be in secs(s), mins(m), hours(h), days(d) or years(h). If zero, the log will be polled an retrieved according to the poll interval and will end when no sessions are candidates for retrieval

Examples

Perform a one time retrieval of the Hadoop Log for the current session if it is being executed in an Oozie execution engine.

```
OdiRetrieveHadopLog -SESSION_LIST=<?=odiRef.getSession("SESS_NO")?>
```

OdiStartLoadPlan

Use this command to start a Load Plan.

The -SYNC parameter starts a load plan in synchronous or asynchronous mode. In synchronous mode, the tool ends with the same status as the completed load plan run.

Usage

```
OdiStartLoadPlan -LOAD_PLAN_NAME=<load_plan_name> [-LOG_LEVEL=<log_level>]
[-CONTEXT=<context_code>] [-AGENT_URL=<agent_url>]
[-AGENT_CODE=<logical_agent_code>] [-ODI_USER=<ODI User>]
[-ODI_PASS=<ODI Password>] [-KEYWORDS=<Keywords>]
[-<PROJECT_CODE>.<VARIABLE>=<var_value> ...] [-SYNC=<yes|no>] [-POLLINT=<msec>]
```

Parameters

Parameters	Mandatory	Description
-LOAD_PLAN_NAME=<load_plan_name>	Yes	Name of the load plan to start.
-LOG_LEVEL=<log_level>	No	Level of logging information to retain. All sessions with a defined log level lower than or equal to this value are kept in the session log when the session completes. However, if object execution ends abnormally, all tasks are kept, regardless of this setting.
		Note that log level 6 has the same behavior as log level 5, but with the addition of variable and sequence tracking. See Tracking Variables and Sequences in *Developing Integration Projects with Oracle Data Integrator* for more information.
[-CONTEXT=<context_code>]	Yes	Code of the execution context. If this parameter is omitted, the load plan starts in the execution context of the calling session, if any.
[-AGENT_URL=<agent_url>]	No	URL of the remote agent that starts the load plan.
[-AGENT_CODE=<logical_agent_code>]	No	Code of the logical agent responsible for starting this load plan. If this parameter and -AGENT_URL are omitted, the current agent starts this load plan. This parameter is ignored if -AGENT_URL is specified.
[-ODI_USER=<ODI user>]	No	Oracle Data Integrator user to be used to start the load plan. The privileges of this user are used. If this parameter is omitted, the load plan is started with privileges of the user launching the parent session.
[-ODI_PASS=<ODI Password>]	No	Password of the Oracle Data Integrator user. This password must be encoded. This parameter is required if -ODI_USER is specified.

Parameters	Mandatory	Description
-KEYWORDS=<keywords>	No	Comma-separated list of keywords attached to this load plan. These keywords make load plan execution identification easier.
-<VARIABLE>=<value>	No	List of project or global variables whose value is set as the default for the execution of the load plan. Project variables should be named <project_code>.<variable_name> and global variables should be named GLOBAL.<variable_name>. This list is of the form -<variable>=<value>.
-SYNC=<yes|no>	No	Specifies whether the load plan should be executed synchronously or asynchronously.
		If set to Yes (synchronous mode), the load plan is started and runs to completion with a status of Done or Error before control is returned.
		If set to No (asynchronous mode), the load plan is started and control is returned before the load plan runs to completion. The default value is No.
-POLLINT=<msec>	No	The time in milliseconds to wait between polling the load plan run status for completion state. The -SYNC parameter must be set to Yes. The default value is 1000 (1 second). The value must be greater than 0.

Examples

Start load plan LOAD_DWH in the GLOBAL context on the same agent.

```
OdiStartLoadPlan -LOAD_PLAN_NAME=LOAD_DWH -CONTEXT=GLOBAL
```

OdiStartOwbJob

Use this command to execute Oracle Warehouse Builder (OWB) objects from within Oracle Data Integrator and to retrieve the execution audit data into Oracle Data Integrator.

This command uses an Oracle Warehouse Builder runtime repository data server that can be created in Topology Navigator. This data server must connect as an Oracle Warehouse Builder user who can access an Oracle Warehouse Builder workspace. The physical schemas under this data server represent the Oracle Warehouse Builder workspaces that this user can access. For information about the Oracle Data Integrator topology, see Setting Up a Topology in *Administering Oracle Data Integrator*

Usage

```
OdiStartOwbJob -WORKSPACE=<logical_owb_repository> -LOCATION=<owb_location>
-OBJECT_NAME=<owb_object> -OBJECT_TYPE=<owb_object_type>
[-EXEC_PARAMS=<exec_params>] [-CONTEXT=<context_code>] [-LOG_LEVEL=<log_level>]
[-SYNC_MODE=<1|2>] [-POLLINT=<n>] [-SESSION_NAME=<session_name>]
[-KEYWORDS=<keywords>] [<OWB parameters>]
```

Parameters

Parameters	Mandatory	Description
-WORKSPACE=<logical_owb_repository>	Yes	Logical schema of the OWB Runtime Repository technology. This resolves to a physical schema that represents the Oracle Warehouse Builder workspace that contains the Oracle Warehouse Builder object to be executed. The Oracle Warehouse Builder workspace was chosen when you added a Physical Schema under the OWB Runtime Repository DataServer in Topology Navigator. The context for this mapping can also be specified using the -CONTEXT parameter.
-LOCATION=<owb_location>	Yes	Name of the Oracle Warehouse Builder location that contains the Oracle Warehouse Builder object to be executed. This location must exist in the physical workspace that resolves from -WORKSPACE.
-OBJECT_NAME=<owb_object>	Yes	Name of the Oracle Warehouse Builder object. This object must exist in -LOCATION.
-OBJECT_TYPE=<owb_object_type>	Yes	Type of Oracle Warehouse Builder object, for example: PLSQLMAP, PROCESSFLOW, SQLLOADERCONTROLFILE, MAPPING, DATAAUDITOR, ABAPFILE
-EXEC_PARAMS=<exec_params>	No	Custom and/or system parameters for the Oracle Warehouse Builder execution.

Parameters	Mandatory	Description
-CONTEXT=<context_code>	No	Execution context of the Oracle Warehouse Builder object. This is the context in which the logical workspace will be resolved. Studio editors use this value or the Default Context. Execution uses this value or the Parent Session context.
-LOG_LEVEL=<log_level>	No	Log level (0-5). The default value is 5, which means that maximum details are captured in the log.
-SYNC_MODE=<1|2>	No	Synchronization mode of the Oracle Warehouse Builder job:
		1 - Synchronous (default). Execution of the session waits until the Oracle Warehouse Builder job terminates.
		2 - Asynchronous. Execution of the session continues without waiting for the Oracle Warehouse Builder job to terminate.
-POLLINT=<n>	No	The period of time in milliseconds to wait between each transfer of Oracle Warehouse Builder audit data to Oracle Data Integrator log tables. The default value is 0, which means that audit data is transferred at the end of the execution.
-SESSION_NAME=<session_name>	No	Name of the Oracle Warehouse Builder session as it appears in the log.
-KEYWORDS=<keywords>	No	Comma-separated list of keywords attached to the session.
<OWB parameters>	No	List of values for the Oracle Warehouse Builder parameters relevant to the object. This list is of the form -PARAM_NAME=value. Oracle Warehouse Builder system parameters should be prefixed by OWB_SYSTEM, for example, OWB_SYSTEM.AUDIT_LEVEL.

Examples

Execute the Oracle Warehouse Builder process flow LOAD_USERS that has been deployed to the Oracle Workflow DEV_OWF.

```
OdiStartOwbJob -WORKSPACE=OWB_WS1 -CONTEXT=QA
-LOCATION=DEV_OWF -OBJECT_NAME=LOAD_USERS -OBJECT_TYPE=PROCESSFLOW
```

Execute the Oracle Warehouse Builder PL/SQL map STAGE_USERS that has been deployed to the database location DEV_STAGE. Poll and transfer the Oracle Warehouse Builder audit data every 5 seconds. Pass the input parameter AGE_LIMIT whose value is obtained from an Oracle Data Integrator variable, and specify an Oracle Warehouse Builder system parameter relevant to a PL/SQL map.

```
OdiStartOwbJob -WORKSPACE=OWB_WS1 -CONTEXT=QA
-LOCATION=DEV_STAGE -OBJECT_NAME=STAGE_USERS -OBJECT_TYPE=PLSQLMAP
-POLLINT=5000 -OWB_SYSTEM.MAX_NO_OF_ERRORS=25 -AGE_LIMIT=#VAR_MINAGE
```

OdiStartScen

Use this command to start a scenario.

The optional parameter -AGENT_CODE is used to dedicate this scenario to another agent other than the current agent.

The parameter -SYNC_MODE starts a scenario in synchronous or asynchronous mode.

> **Note:** The scenario that is started should be present in the repository into which the command is launched. If you go to production with a scenario, make sure to also take all scenarios called by your scenario using this command. The Solutions can help you group scenarios for this purpose.

Usage

```
OdiStartScen -SCEN_NAME=<scenario> -SCEN_VERSION=<version>
[-CONTEXT=<context>] [-ODI_USER=<odi user> -ODI_PASS=<odi password>]
[-SESSION_NAME=<session_name>] [-LOG_LEVEL=<log_level>]
[-AGENT_CODE=<logical_agent_name>] [-SYNC_MODE=<1|2>]
[-KEYWORDS=<keywords>] [-<VARIABLE>=<value>]*
```

Parameters

Parameters	Mandatory	Description
-SCEN_NAME=<scenario>	Yes	Name of the scenario to start.
-SCEN_VERSION=<version>	Yes	Version of the scenario to start. If the version specified is -1, the last version of the scenario is executed.
-CONTEXT=<context>	No	Code of the execution context. If this parameter is omitted, the scenario is executed in the execution context of the calling session.
-ODI_USER=<odi user>	No	Oracle Data Integrator user to be used to run the scenario. The privileges of this user are used. If this parameter is omitted, the scenario is executed with privileges of the user launching the parent session.
-ODI_PASS=<odi password>	No	Password of the Oracle Data Integrator user. This password should be encoded. This parameter is required if the user is specified.
-SESSION_NAME=<session_name>	No	Name of the session that will appear in the execution log.
-LOG_LEVEL=<log_level>	No	Trace level (0 .. 5) to keep in the execution log. The default value is 5.
-AGENT_CODE=<logical_agent_name>	No	Name of the logical agent responsible for executing this scenario. If this parameter is omitted, the current agent executes this scenario.

Parameters	Mandatory	Description
-SYNC_MODE=<1|2>	No	Synchronization mode of the scenario: 1 - Synchronous mode (default). The execution of the calling session is blocked until the scenario finishes its execution. 2 - Asynchronous mode. The execution of the calling session continues independently from the return of the called scenario.
-KEYWORDS=<keywords>	No	Comma-separated list of keywords attached to this session. These keywords make session identification easier.
-<VARIABLE>=<value>	No	List of variables whose value is set for the execution of the scenario. This list is of the form PROJECT.VARIABLE=value or GLOBAL.VARIABLE=value.

Examples

Start the scenario LOAD_DWH in version 2 in the production context (synchronous mode).

```
OdiStartScen -SCEN_NAME=LOAD_DWH -SCEN_VERSION=2
-CONTEXT=CTX_PRODUCTION
```
Start the scenario LOAD_DWH in version 2 in the current context in asynchronous mode on the agent UNIX Agent while passing the values of the variables START_DATE (local) and COMPANY_CODE (global).

```
OdiStartScen -SCEN_NAME=LOAD_DWH -SCEN_VERSION=2 -SYNC_MODE=2
"-AGENT_CODE=UNIX Agent" -MY_PROJECT.START_DATE=10-APR-2002
-GLOBAL.COMPANY_CODE=SP4356
```

OdiUnZip

Use this command to extract an archive file to a directory.

Usage

```
OdiUnZip -FILE=<file> -TODIR=<target_directory> [-OVERWRITE=<yes|no>]
[-ENCODING=<file_name_encoding>]
```

Parameters

Parameters	Mandatory	Description
-FILE=<file>	Yes	Full path to the ZIP file to extract.
-TODIR=<target_file>	Yes	Destination directory or folder.
-OVERWRITE=<yes|no>	No	Indicates if the files that already exist in the target directory must be overwritten. The default value is No.
-ENCODING=<file_name_encoding>	No	Character encoding used for file names inside the archive file. For a list of possible values, see:
		http://java.sun.com/j2se/1.4.2/docs/guide/intl/encoding.doc.html
		Defaults to the platform's default character encoding.

Examples

Extract the file `archive_001.zip` from directory `C:\archive\` into directory `C:\TEMP`.

```
OdiUnZip "-FILE=C:\archive\archive_001.zip" -TODIR=C:\TEMP\
```

OdiUpdateAgentSchedule

Use this command to force an agent to recalculate its schedule of tasks.

Usage

```
OdiUpdateAgentSchedule -AGENT_NAME=<physical_agent_name>
```

Parameters

Parameters	Mandatory	Description
-AGENT_NAME=<physical_agent_name>	Yes	Name of the physical agent to update.

Examples

Cause the physical agent agt_s1 to update its schedule.

```
OdiUpdateAgentSchedule -AGENT_NAME=agt_s1
```

OdiWaitForChildSession

Use this command to wait for the child session (started using the OdiStartScen tool) of the current session to complete.

This command checks every `<polling_interval>` to determine if the sessions launched from `<parent_sess_number>` are finished. If all child sessions (possibly filtered by their name and keywords) are finished (status of Done, Warning, or Error), this command terminates.

Usage

```
OdiWaitForChildSession [-PARENT_SESS_NO=<parent_sess_number>]
[-POLL_INT=<polling_interval>]
[-SESSION_NAME_FILTER=<session_name_filter>]
[-SESSION_KEYWORDS=<session_keywords>]
[-MAX_CHILD_ERROR=ALL|<error_number>]
```

Parameters

Parameters	Mandatory	Description
-PARENT_SESS_NO=<parent_sess_number>	No	ID of the parent session. If this parameter is not specified, the current session ID is used.
-POLL_INT=<polling_interval>	No	Interval in seconds between each sequence of termination tests for the child sessions. The default value is 1.
-SESSION_NAME_FILTER=<session_name_filter>	No	Only child sessions whose names match this filter are tested. This filter can be a SQL LIKE-formatted pattern.
-SESSION_KEYWORDS=<session_keywords>	No	Only child sessions for which ALL keywords have a match in this comma-separated list are tested. Each element of the list can be a SQL LIKE-formatted pattern.
-MAX_CHILD_ERROR= ALL|<error_number>	No	This parameter enables OdiWaitForChildSession to terminate in error if a number of child sessions have terminated in error: ■ ALL: Error if all child sessions terminate in error. ■ <error_number>: Error if <error_number> or more child sessions terminate in error. If this parameter is equal to 0, negative, or not specified, OdiWaitForChildSession never terminates in an error status, regardless of the number of failing child sessions.

Examples

Wait and poll every 5 seconds for all child sessions of the current session with a name filter of LOAD% and keywords MANDATORY and CRITICAL to finish.

```
OdiWaitForChildSession -PARENT_SESS_NO=<%=odiRef.getSession("SESS_NO")%>
-POLL_INT=5 -SESSION_NAME_FILTER=LOAD%
-SESSION_KEYWORDS=MANDATORY,CRITICAL
```

OdiWaitForData

Use this command to wait for a number of rows in a table or set of tables. This can also be applied to a number of objects containing data, such as views.

The OdiWaitForData command tests that a table, or a set of tables, has been populated with a number of records. This test is repeated at regular intervals (-POLLINT) until one of the following conditions is met: the desired number of rows for one of the tables has been detected (-UNIT_ROWCOUNT), the desired, cumulated number of rows for all of the tables has been detected (-GLOBAL_ROWCOUNT), or a timeout (-TIMEOUT) has been reached.

Filters may be applied to the set of counted rows. They are specified by an explicit SQL where clause (-SQLFILTER) and/or the -RESUME_KEY_xxx parameters to determine field-value-operator clause. These two methods are cumulative (AND).

The row count may be considered either in absolute terms (with respect to the total number of rows in the table) or in differential terms (the difference between a stored reference value and the current row count value).

When dealing with multiple tables:

- The -SQLFILTER and -RESUME_KEY_xxx parameters apply to **ALL** tables concerned.

- The -UNIT_ROWCOUNT parameter determines the row count to be expected for each table. The -GLOBAL_ROWCOUNT parameter determines the SUM of the row count number cumulated over the set of tables. When only one table is concerned, the -UNIT_ROWCOUNT and -GLOBAL_ROWCOUNT parameters are equivalent.

Usage

```
OdiWaitForData -LSCHEMA=<logical_schema> -TABLE_NAME=<table_name>
[-OBJECT_TYPE=<list of object types>] [-CONTEXT=<context>]
[-RESUME_KEY_VARIABLE=<resumeKeyVariable>
-RESUME_KEY_COL=<resumeKeyCol>
[-RESUME_KEY_OPERATOR=<resumeKeyOperator>]|-SQLFILTER=<SQLFilter>]
[-TIMEOUT=<timeout>] [-POLLINT=<pollInt>]
[-GLOBAL_ROWCOUNT=<globalRowCount>]
[-UNIT_ROWCOUNT=<unitRowCount>] [-TIMEOUT_WITH_ROWS_OK=<yes|no>]
[-INCREMENT_DETECTION=<no|yes> [-INCREMENT_MODE=<M|P|I>]
[-INCREMENT_SEQUENCE_NAME=<incrementSequenceName>]]
```

Parameters

Parameters	Mandatory	Description
-LSCHEMA=<logical_schema>	Yes	Logical schema containing the tables.

Parameters	Mandatory	Description
-TABLE_NAME=<table_name>	Yes	Table name, mask, or list of table names to check. This parameter accepts three formats: ■ Table Name ■ Table Name Mask: This mask selects the tables to poll. The mask is specified using the SQL LIKE syntax: the % symbol replaces an unspecified number of characters and the _ symbol is a single character wildcard. ■ Table Names List: Comma-separated list of table names. Masks as defined above are allowed.
-OBJECT_TYPE=<list of object types>	No	Type of objects to check. By default, only tables are checked. To take into account other objects, specify a comma-separated list of object types. Supported object types are: ■ T: Table ■ V: View
-CONTEXT=<context>	No	Context in which the logical schema will be resolved. If no context is specified, the execution context is used.
-SQLFILTER=<SQLFilter>	No	Explicit SQL filter to be applied to the table(s). This statement must be valid for the technology containing the checked tables. Note that this statement must not include the WHERE keyword.
-RESUME_KEY_VARIABLE=<resumeKeyVariable> -RESUME_KEY_COL=<resumeKeyCol> [-RESUME_KEY_OPERATOR=<resumeKeyOperator>]	No	The RESUME_KEY_xxx parameters enable filtering of the set of counted rows in the polled tables. ■ <key_column>: Name of a column in the checked table. ■ <operator>: Valid comparison operator for the technology containing the checked tables. If this parameter is omitted, the value > is used by default. ■ <variable_name>: Variable name whose value has been previously set. The variable name must be prefixed with : (bind) or # (substitution). The variable scope should be explicitly stated in the Oracle Data Integrator syntax; GLOBAL.<variable name> for global variables or <project code>.<variable name> for project variables.
-TIMEOUT=<timeout>	No	Maximum period of time in milliseconds over which data is polled. If this value is equal to 0, the timeout is infinite. The default value is 0.

Parameters	Mandatory	Description
-POLLINT=<pollInt>	No	The period of time in milliseconds to wait between data polls. The default value is 1000.
-UNIT_ROWCOUNT=<unitRowCount>	No	Number of rows expected in a polled table to terminate the command. The default value is 1.
-GLOBAL_ROWCOUNT=<globalRowCount>	No	Total number of rows expected cumulatively, over the set of tables, to terminate the command. If not specified, the default value 1 is used.
-INCREMENT_DETECTION=<no|yes>	No	Defines the mode in which the command considers row count: either in absolute terms (with respect to the total number of rows in the table) or in differential terms (the difference between a stored reference value and the current row count value).

Under the last row's description:

- If set to Yes, the row count is performed in differential mode. The number of additional rows in the table is compared to a stored **reference value**. The reference value depends on the -INCREMENT_MODE parameter.

- If set to No, the count is performed in absolute row count mode.

The default value is No.

Parameters	Mandatory	Description
-INCREMENT_MODE=<M|P|I>	No	This parameter specifies the persistence mode of the **reference value** between successive OdiWaitForData calls.
		Possible values are:
		■ M: Memory. The **reference value** is nonpersistent. When OdiWaitForData is called, the **reference value** takes a value equal to the number of rows in the polled table. When OdiWaitForData ends the value is lost. A following call in this mode sets a new reference value.
		■ P: Persistent. The **reference value** is persistent. It is read from the **increment sequence** when OdiWaitForData starts and saved in the increment sequence when OdiWaitForData ends. If the **increment sequence** is not set (at initial call time) the current table row count is used.
		■ I: Initial. The **reference value** is initialized and is persistent. When OdiWaitForData starts, the **reference value** takes a value equal to the number of rows in the polled table. When OdiWaitForData ends, it is saved in the increment sequence as for the persistent mode.
		The default value is M.
		Note that using the Persistent or Initial modes is not supported when a mask or list of tables is polled.
-INCREMENT_SEQUENCE_ NAME=<incrementSequenceName>	No	This parameter specifies the name of an automatically allocated storage space used for reference value persistence. This increment sequence is stored in the Repository. If this name is not specified, it takes the name of the table.
		Note that this Increment Sequence is not an Oracle Data Integrator Sequence and cannot be used as such outside a call to OdiWaitForData.
-TIMEOUT_WITH_ROWS_OK=<yes|no>	No	If this value is set to Yes, at least one row was detected, and the timeout occurs before the expected number of rows has been inserted, the API exits with a return code of 0. Otherwise, it signals an error. The default value is Yes.

Examples

Wait for the DE1P1 table in the ORA_WAITFORDATA schema to contain 200 records matching the filter.

```
OdiWaitForData -LSCHEMA=ORA_WAITFORDATA -TABLE_NAME=DE1P1
-GLOBAL_ROWCOUNT=200 "-SQLFILTER=DATMAJ >
```

```
to_date('#MAX_DE1_DATMAJ_ORACLE_CHAR', 'DD/MM/YYYY HH24:MI:SS')"
```
Wait for a maximum of 4 hours for new data to appear in either the CITY_SRC or the
CITY_TRG table in the logical schema SQLSRV_SALES.

```
OdiWaitForData -LSCHEMA=SQLSRV_SALES -TABLE_NAME=CITY%
-TIMEOUT=14400000 -INCREMENT_DETECTION=yes
```

OdiWaitForLoadPlans

Use this command to wait for load plan runs to complete.

Usage

```
OdiWaitForLoadPlans [-PARENT_SESS_NO=<parent_sess_guid>]
[-LP_NAME_FILTER=<load_plan_name_filter>] [-LP_KEYWORDS=<load_plan_keywords>]
[-MAX_LP_ERROR=ALL|<number_of_lp_errors>] [-POLLINT=<polling_interval_msec>]
```

Parameters

Parameters	Mandatory	Description
-PARENT_SESS_NO=<parent_sess_guid>	No	Global ID of the parent session that started the load plan. If this parameter is not specified, the global ID of the current session is used.
-LP_NAME_FILTER=<load_plan_name_filter>	No	Only load plan runs whose name matches this filter are tested for completion status. This filter can be a SQL LIKE-formatted pattern.
-LP_KEYWORDS=<load_plan_keywords>	No	Only load plan runs whose keywords contain all entries in this comma-separated list are tested for completion status. Each element in the list can be a SQL LIKE-formatted pattern.
-MAX_LP_ERROR=ALL|<number_of_lp_errors>	No	OdiWaitForLoadPlans terminates in error if a number of load plan runs are in Error status: ■ ALL: Error if all load plan runs complete in Error status. ■ <number_of_lp_errors>: Error if the number of load plan runs in Error status is at or above this value <number_of_lp_errors> when all load plan runs are complete. If this parameter is not specified or its value is less than 1, OdiWaitForLoadPlans never terminates in error, regardless of the number of load plan runs in Error status.
-POLLINT=<polling_interval_msec>	No	The time in milliseconds to wait between polling load plan run status for completion state. The default value is 1000 (1 second). The value must be greater than 0.

Examples

Wait and poll every 5 seconds for all load plan runs started by the current session with a name filter of POPULATE% and keywords MANDATORY and CRITICAL to finish in a Done or Error status. If 2 or more load plan runs are in Error status when execution is complete for all selected load plan runs, OdiWaitForLoadPlans ends in error.

```
OdiWaitForLoadPlans -PARENT_SESS_NO=<%=odiRef.getSession("SESS_GUID")%>
-LP_NAME_FILTER=POPULATE% -LP_KEYWORDS=MANDATORY,CRITICAL
-POLLINT=5000 -MAX_LP_ERROR=2
```

OdiWaitForLogData

Use this command to wait for a number of modifications to occur on a journalized table or a list of journalized tables.

The OdiWaitForLogData command determines whether rows have been modified on a table or a group of tables. These changes are detected using the Oracle Data Integrator changed data capture (CDC) in simple mode (using the -TABLE_NAME parameter) or in consistent mode (using the -CDC_SET_NAME parameter). The test is repeated every -POLLINT milliseconds until one of the following conditions is met: the desired number of row modifications for one of the tables has been detected (-UNIT_ROWCOUNT), the desired cumulative number of row modifications for all of the tables has been detected (-GLOBAL_ROWCOUNT), or a timeout (-TIMEOUT) has been reached.

> **Note:** This command takes into account all journalized operations (inserts, updates and deletes).
>
> The command is suitable for journalized tables only in simple or consistent mode.

Usage

```
OdiWaitForLogData -LSCHEMA=<logical_schema>  -SUBSCRIBER_NAME=<subscriber_name>
(-TABLE_NAME=<table_name> | -CDC_SET_NAME=<cdcSetName>)
[-CONTEXT=<context>] [-TIMEOUT=<timeout>] [-POLLINT=<pollInt>]
[-GLOBAL_ROWCOUNT=<globalRowCount>]
[-UNIT_ROWCOUNT=<unitRowCount> [-OPTIMIZED_WAIT=<yes|no|AUTO>]
[-TIMEOUT_WITH_ROWS_OK=<yes|no>]
```

Parameters

Parameters	Mandatory	Description
-CONTEXT=<context>	No	Context in which the logical schema will be resolved. If no context is specified, the execution context is used.
-GLOBAL_ROWCOUNT=<globalRowCount>	No	Total number of changes expected in the tables or the CDC set to end the command. The default value is 1.
-LSCHEMA=<logical_schema>	Yes	Logical schema containing the journalized tables.

Parameters	Mandatory	Description
-OPTIMIZED_WAIT=<yes|no|AUTO>	No	Method used to access the journals.
		■ yes: Optimized method. This method works for later versions of journalizing. It runs faster than the nonoptimized mode.
		■ no: Nonoptimized method. A count is performed on the journalizing table. This method is of lower performance but compatible with earlier versions of the journalizing feature.
		■ AUTO: If more than one table is checked, the optimized method is used. Otherwise, the nonoptimized method is used.
		The default value is AUTO.
-POLLINT=<pollInt>	No	The period of time in milliseconds to wait between polls. The default value is 2000.
-SUBSCRIBER_NAME=<subscriber_name>	Yes	Name of the subscriber used to get the journalizing information.
-TABLE_NAME=<table_name>	Yes	Journalized table name, mask, or list to check. This parameter accepts three formats:
		■ Table Name
		■ Table Name Mask: This mask selects the tables to poll. The mask is specified using the SQL LIKE syntax: the % symbol replaces an unspecified number of characters and the _ symbol acts as a wildcard.
		■ Table Names List: List of table names separated by commas. Masks as defined above are **not** allowed.
		Note that this option works only for tables in a model journalized in simple mode.
		This parameter cannot be used with -CDC_SET_NAME. It is mandatory if -CDC_SET_NAME. is not set.
-CDC_SET_NAME=<cdcSetName>	Yes	Name of the CDC set to check. This CDC set name is the fully qualified model code, typically PHYSICAL_SCHEMA_NAME.MODEL_CODE.
		It can be obtained in the current context using a substitution method API call, as shown below: <%=odiRef.getObjectName("L", "model_code", "logical_schema", "D")%>.
		Note that this option works only for tables in a model journalized in consistent mode.
		This parameter cannot be used with -TABLE_NAME. It is mandatory if -TABLE_NAME is not set.
-TIMEOUT=<timeout>	No	Maximum period of time in milliseconds over which changes are polled. If this value is equal to 0, the timeout is infinite. The default value is 0.

Parameters	Mandatory	Description
-TIMEOUT_WITH_ROWS_OK=<yes|no>	No	If this parameter is set to Yes, at least one row was detected, and the timeout occurs before the predefined number of rows has been polled, the API exits with a return code of 0. Otherwise, it signals an error. The default value is Yes.
-UNIT_ROWCOUNT=<unitRowCount>	No	Number of changes expected in one of the polled tables to end the command. The default value is 1.
		Note that -UNIT_ROWCOUNT is not taken into account with -CDC_SET_NAME.

Examples

Wait for the CUSTOMERS table in the SALES_APPLICATION schema to have 200 row modifications recorded for the SALES_SYNC subscriber.

```
OdiWaitForLogData -LSCHEMA=SALES_APPLICATION
-TABLE_NAME=CUSTOMERS -GLOBAL_ROWCOUNT=200
-SUBSCRIBER_NAME=SALES_SYNC
```

OdiWaitForTable

Use this command to wait for a table to be created and populated with a predefined number of rows.

The OdiWaitForTable command regularly tests whether the specified table has been created and has been populated with a number of records. The test is repeated every -POLLINT milliseconds until the table exists and contains the desired number of rows (-GLOBAL_ROWCOUNT), or until a timeout (-TIMEOUT) is reached.

Usage

```
OdiWaitForTable -CONTEXT=<context> -LSCHEMA=<logical_schema>
-TABLE_NAME=<table_name> [-TIMEOUT=<timeout>] [-POLLINT=<pollInt>]
[-GLOBAL_ROWCOUNT=<globalRowCount>] [-TIMEOUT_WITH_ROWS_OK=<yes|no>]
```

Parameters

Parameters	Mandatory	Description
-CONTEXT=<context>	No	Context in which the logical schema will be resolved. If no context is specified, the execution context is used.
-GLOBAL_ROWCOUNT=<globalRowCount>	No	Total number of rows expected in the table to terminate the command. The default value is 1. If not specified, the command finishes when a new row is inserted into the table.
-LSCHEMA=<logical_schema>	Yes	Logical schema in which the table is searched for.
-POLLINT=<pollInt>	No	Period of time in milliseconds to wait between each test. The default value is 1000.
-TABLE_NAME=<table_name>	Yes	Name of table to search for.
-TIMEOUT=<timeout>	No	Maximum time in milliseconds the table is searched for. If this value is equal to 0, the timeout is infinite. The default value is 0.
-TIMEOUT_WITH_ROWS_OK=<yes|no>	No	If this parameter is set to Yes, at least one row is detected, and the timeout occurs before the expected number of records is detected, the API exits with a return code of 0. Otherwise, it signals an error. The default value is Yes.

Examples

Wait for the DE1P1 table in the ORA_WAITFORDATA schema to exist, and to contain at least 1 record.

```
OdiWaitForTable -LSCHEMA=ORA_WAITFORDATA -TABLE_NAME=DE1P1
-GLOBAL_ROWCOUNT=1
```

OdiXMLConcat

Use this command to concatenate elements from multiple XML files into a single file.

This tool extracts all instances of a given element from a set of source XML files and concatenates them into one target XML file. The tool parses and generates well formed XML. It does not modify or generate a DTD for the generated files. A reference to an existing DTD can be specified in the -HEADER parameter or preserved from the original files using -KEEP_XML_PROLOGUE.

> **Note:** XML namespaces are not supported by this tool. Provide the local part of the element name (without the namespace or prefix value) in the -ELEMENT_NAME parameter.

Usage

```
OdiXMLConcat -FILE=<file_filter> -TOFILE=<target_file>
-XML_ELEMENT=<element_name> [-CHARSET_ENCODING=<encoding>]
[-IF_FILE_EXISTS=<overwrite|skip|error>]
[-KEEP_XML_PROLOGUE=<all|xml|doctype|none>] [-HEADER=<header>]
[-FOOTER=<footer>]
```

Parameters

Parameters	Mandatory	Description
-FILE=<file_filter>	Yes	Filter for the source XML files. This filter uses standard file wildcards (?,*). It includes both file names and directory names. Source files can be taken from the same folder or from different folders.
		The following file filters are valid:
		■ /tmp/files_*/customer.xml
		■ /tmp/files_*/*.*
		■ /tmp/files_??/customer.xml
		■ /tmp/files/customer_*.xml
		■ /tmp/files/customer_??.xml
-TOFILE=<target_file>	Yes	Target file into which the elements are concatenated.
-XML_ELEMENT=<element_name>	Yes	Local name of the XML element (without enclosing <> characters, prefix, or namespace information) to be extracted with its content and child elements from the source files.
		Note that this element detection is not recursive. If a given instance of <element_name> contains other instances of <element_name>, only the element of higher level is taken into account and child elements are only extracted as a part of the top element's content.

Parameters	Mandatory	Description
-CHARSET_ENCODING=<encoding>	No	Target files encoding. The default value is ISO-8859-1. For the list of supported encodings, see: http://java.sun.com/j2se/1.4.2/docs/guide/intl/encoding.doc.html
-IF_FILE_ EXISTS=<overwrite|skip|error>	No	Define behavior when the target file exists. ■ overwrite: Overwrite the target file if it exists. ■ skip: Do nothing for this file. ■ error: Raise an error.
-KEEP_XML_ PROLOGUE=<all|xml|doctype|none>	No	Copies the source file XML prologue in the target file. Depending on this parameter's value, the following parts of the XML prologue are preserved: ■ all: Copies all of the prologue (XML and document type declaration). ■ xml: Copies only the XML declaration <?xml...?> and not the document type declaration. ■ doctype: Copies only the document type declaration and not the XML declaration. ■ none: Does not copy the prologue from the source file. **Note**: If all or part of the prologue is not preserved, it should be specified in the -HEADER parameter.
-HEADER=<header>	No	String that is appended after the prologue (if any) in each target file. You can use this parameter to create a customized XML prologue or root element.
-FOOTER=<footer>	No	String that is appended at the end of each target file. You can use this parameter to close a root element added in the header.

Examples

Concatenate the content of the IDOC elements in the files ord1.xml, ord2.xml, and so on in the ord_i subfolder into the file MDSLS.TXT.XML, with the root element <WMMBID02> added to the target.

```
OdiXMLConcat "-FILE=./ord_i/ord*.xml" "-TOFILE=./MDSLS.TXT.XML" -XML_ELEMENT=IDOC
"-CHARSET_ENCODING=UTF-8" -IF_FILE_EXISTS=overwrite -KEEP_XML_PROLOGUE=xml
"-HEADER=<WMMBID02>" "-FOOTER=</WMMBID02>"
OdiXMLConcat "-FILE=./o?d_*/ord*.xml" "-TOFILE=./MDSLS.TXT.XML" -XML_ELEMENT=IDOC
"-CHARSET_ENCODING=UTF-8" -IF_FILE_EXISTS=overwrite -KEEP_XML_PROLOGUE=none
"-HEADER=<?xml version=\"1.0\" encoding=\"UTF-8\"?>\n<WMMBID02>"
"-FOOTER=</WMMBID02>"
```

Concatenate the EDI elements of the files ord1.xml, ord2.xml, and so on in the ord_i subfolder into the file MDSLS2.XML. This file will have the new root element EDI_BATCH above all <EDI> elements.

```
OdiXMLConcat "-FILE=./o?d_?/ord*.xml" "-TOFILE=./MDSLS2.XML" -XML_ELEMENT=EDI
"-CHARSET_ENCODING=UTF-8" -IF_FILE_EXISTS=overwrite -KEEP_XML_PROLOGUE=xml
"-HEADER= <EDI_BATCH>" "-FOOTER=</EDI_BATCH>"
```

OdiXMLSplit

This tool extracts all instances of a given element stored in a source XML file and splits it over several target XML files. This tool parses and generates well formed XML. It does not modify or generate a DTD for the generated files. A reference to an existing DTD can be specified in the -HEADER parameter or preserved from the original files using -KEEP_XML_PROLOGUE.

> **Note:** XML namespaces are not supported by this tool. Provide the local part of the element name (without the namespace or prefix value) in the -ELEMENT_NAME parameter.

Usage

```
OdiXMLSplit -FILE=<file> -TOFILE=<file_pattern> -XML_ELEMENT=<element_name>
[-CHARSET_ENCODING=<encoding>] [-IF_FILE_EXISTS=<overwrite|skip|error>]
[-KEEP_XML_PROLOGUE=<all|xml|doctype|none>] [-HEADER=<header>]
[-FOOTER=<footer>]
```

Parameters

Parameters	Mandatory	Description
-FILE=<file>	Yes	Source XML file to split.

Parameters	Mandatory	Description
-TOFILE=<file_pattern>	Yes	File pattern for the target files. Each file is named after a pattern containing a mask representing a generated number sequence or the value of an attribute of the XML element used to perform the split:
		■ **Number Sequence Mask**: Use the * (star) value to indicate the place of the file number value. For example, if the <file_ pattern> is equal to target_*.xml, the files created are named target_1.xml, target_2.xml, and so on.
		■ **Attribute Value Mask**: Between square brackets, specify the name of the attribute of <element_name> whose value should be pushed to create the file name. For example, customer_[CUSTID].xml creates files named customer_041.xml, customer_123.xml, and so on, depending on the value of the attribute CUSTID of the element used to split. Note that if a value repeats over several successive elements, target files may be overwritten according to the value of the -OVERWRITE parameter.
		Note that the pattern can be used for creating different files within a directory or files in different directories. The following patterns are valid:
		■ /tmp/files_*/customer.xml
		■ /tmp/files_[CUSTID]/customer.xml
		■ /tmp/files/customer_*.xml
		■ /tmp/files/customer_[CUSTID].xml
-XML_ELEMENT=<element_name>	Yes	Local name of the XML element (without enclosing <> characters, prefix, or namespace information) to be extracted with its content and child elements from the source files.
		Note that this element detection is not recursive. If a given instance of <element_name> contains other instances of <element_name>, only the element of higher level is taken into account and child elements are only extracted as a part of the top element's content.
-CHARSET_ENCODING=<encoding>	No	Target files encoding. The default value is ISO-8859-1. For the list of supported encodings, see: http://java.sun.com/j2se/1.4.2/docs/guide/intl/encoding.doc.html
-IF_FILE_ EXISTS=<overwrite|skip|error>	No	Define behavior when the target file exists.
		■ overwrite: Overwrite the target file if it exists.
		■ skip: Do nothing for this file.
		■ error: Raise an error.

Parameters	Mandatory	Description
-KEEP_XML_ PROLOGUE=<all|xml|doctype|none>	No	Copies the source file XML prologue in the target file. Depending on this parameter's value, the following parts of the XML prologue are preserved: ■ all: Copies all of the prologue (XML and document type declaration). ■ xml: Copies only the XML declaration <?xml...?> and not the document type declaration. ■ doctype: Copies only the document type declaration and not the XML declaration. ■ none: Does not copy the prologue from the source file. **Note**: If all or part of the prologue is not preserved, it should be specified in the -HEADER parameter.
-HEADER=<header>	No	String that is appended after the prologue (if any) in each target file. You can use this parameter to create a customized XML prologue or root element.
-FOOTER=<footer>	No	String that is appended at the end of each target file. You can use this parameter to close a root element added in the header.

Examples

Split the file MDSLS.TXT.XML into several files. The files ord1.xml, ord2.xml, and so on are created and contain each instance of the IDOC element contained in the source file.

```
OdiXMLSplit "-FILE=./MDSLS.TXT.XML" "-TOFILE=./ord_i/ord*.xml" -XML_ELEMENT=IDOC
"-CHARSET_ENCODING=UTF-8" -IF_FILE_EXISTS=overwrite -KEEP_XML_PROLOGUE=xml
"-HEADER= <WMMBID02>" "-FOOTER= </WMMBID02>"
```
Split the file MDSLS.TXT.XML the same way as in the previous example except name the files using the value of the BEGIN attribute of the IDOC element that is being split. The XML prologue is not preserved in this example but entirely generated in the header.

```
OdiXMLSplit "-FILE= ./MDSLS.TXT.XML" "-TOFILE=./ord_i/ord[BEGIN].xml"
-XML_ELEMENT=IDOC "-CHARSET_ENCODING=UTF-8" -IF_FILE_EXISTS=overwrite -KEEP_XML
PROLOGUE=none "-HEADER= <?xml version=\"1.0\" encoding=\"UTF-8\"?>\n<WMMBID02>"
"-FOOTER=</WMMBID02>"
```

OdiZip

Use this command to create a ZIP file from a directory or several files.

Usage

```
OdiZip -DIR=<directory> -FILE=<file> -TOFILE=<target_file> [-OVERWRITE=<yes|no>]
[-RECURSE=<yes|no>] [-CASESENS=<yes|no>]
[-ENCODING=<file_name_encoding>]
```

Parameters

Parameters	Mandatory	Description
-DIR=<directory>	Yes if -FILE is omitted	Base directory (or folder) that will be the future root in the ZIP file to generate. If only -DIR and not -FILE is specified, all files under this directory are archived.
-FILE=<file>	Yes if -DIR is omitted	Path from the base directory of the file(s) to archive. If only -FILE and not -DIR is specified, the default directory is the current work directory if the -FILE path is relative.
		Use * to specify the generic characters.
		Examples:
		/var/tmp/*.log (all files with the log extension of the directory /var/tmp)
		arch_*.lst (all files starting with arch_ and with the extension lst)
-TOFILE=<target_file>	Yes	Target ZIP file.
-OVERWRITE=<yes|no>	No	Indicates whether the target ZIP file must be overwritten (Yes) or simply updated if it already exists (No). By default, the ZIP file is updated if it already exists.
-RECURSE=<yes|no>	No	Indicates if the archiving is recursive in the case of a directory that contains other directories. The value No indicates that only the files contained in the directory to copy (without the subfolders) are archived.
-CASESENS=<yes|no>	No	Indicates if file search is case-sensitive. By default, Oracle Data Integrator searches files in uppercase (set to No).
-ENCODING=<file_name_encoding>	No	Character encoding to use for file names inside the archive file.
		For the list of supported encodings, see:
		http://java.sun.com/j2se/1.4.2/docs/guide/intl/encoding.doc.html
		This defaults to the platform's default character encoding.

Examples

Create an archive of the directory C:\Program files\odi.

```
OdiZip "-DIR=C:\Program Files\odi" -FILE=*.* -TOFILE=C:\TEMP\odi_archive.zip
```

Create an archive of the directory `C:\Program files\odi` while preserving the `odi` directory in the archive.

```
OdiZip "-DIR=C:\Program Files" -FILE=odi\*.* -TOFILE=C:\TEMP\odi_archive.zip
```