An Oracle White Paper
February 2014

Oracle Data Integrator 12c Architecture Overview

Introduction

Oracle Data Integrator (ODI) 12c is built on several components all working together around a centralized metadata repository. These components – graphical modules, runtime agents and web-based interfaces – in conjunction with other advanced features make ODI a lightweight, state of the art data integration platform. This technical brief describes the Oracle Data Integrator 12c architecture in detail.

Architecture Overview

The Oracle Data Integrator architecture is organized around a modular repository, which is accessed in client-server mode by components such as the **ODI Studio** and execution **Agents** that are written entirely in Java. The architecture also incorporates web-based components such as the **Management Pack for Oracle Data Integrator** which integrates with the **Oracle Enterprise Manager Cloud Control** infrastructure or the **ODI Console** which enables users to access metadata information through a Web interface.

ODI Studio

ODI Studio provides four graphical Navigators for managing ODI artifacts: **Designer**, **Operator**, **Topology** and **Security**. ODI Studio offers an easy to use graphical user interface and can be installed on various platforms such as Microsoft Windows or Linux.

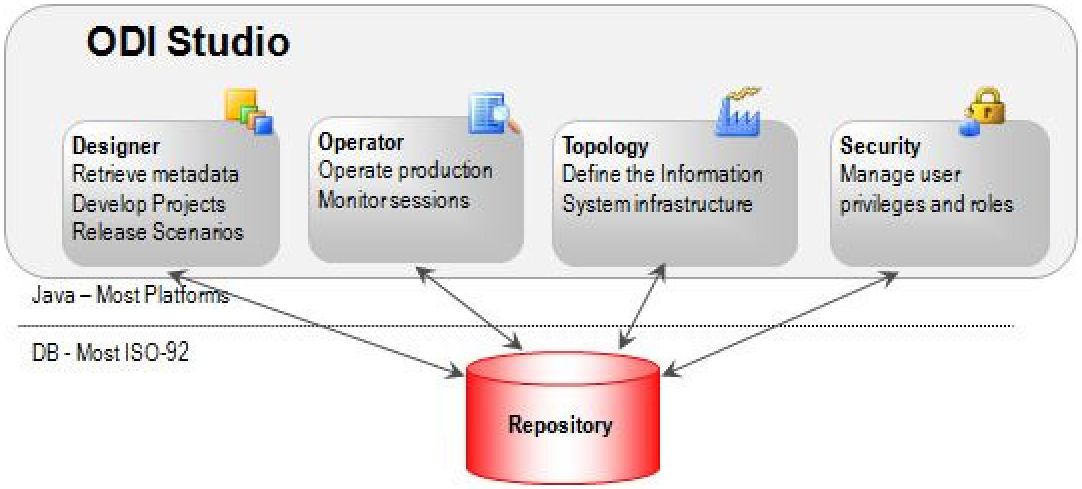

Figure 1 - ODI Studio Navigators connect to the repository

ODI Studio Navigators are as follows:

- **Designer** defines declarative rules for data transformation and data integrity. All project development takes place in this module; this is where database and application metadata are imported and defined. The Designer module uses metadata and rules to generate data integration scenarios or load plans for production. This is the core module for developers and metadata administrators.

- **Operator** manages and monitors data integration processes in production. It is designed for operators and shows execution logs with error counts, the number of rows processed, execution statistics, the actual code that is executed, and so on. At design time, developers can also use the Operator module for troubleshooting purposes.

- **Topology** defines the physical and logical architecture of the infrastructure. The infrastructure or projects administrators register servers, database schemas and catalogs, and agents in the master repository through this module.

- **Security** manages user profiles, roles and their privileges. Security can also assign access authorization to objects and features. Security administrators generally use this module.

All modules store their information in the centralized repository.

Runtime Components: The Agents

At runtime, the **Agent** coordinates the execution of the ODI sessions. It retrieves the code stored in the ODI repository, connects to the various source and target systems and orchestrates the overall data integration process. There are three types of Agents in Oracle Data Integrator 12c:

- **Standalone Agents** can be installed on the source or target systems and require a Java Virtual Machine.

- **Colocated Standalone Agents** can be installed on the source or target systems as well. They can be managed using Oracle Enterprise Manager and must be configured with an Oracle WebLogic domain. Colocated Standalone Agents can run on a separate machine from the Oracle WebLogic Administration Server.

- **Java Enterprise Edition (Java EE) Agents** are deployed on Oracle WebLogic Server and can benefit from the application server layer features such as clustering for High Availability requirements. Java EE Agents can be managed using Oracle Enterprise Manager.

With the Extract-Load Transform (E-LT) architecture, the Agent rarely performs any transformation. It simply retrieves code from the ODI repository and then requests database servers, operating systems, or scripting engines to execute that code. When the execution is completed, the Agent updates the execution logs in the repository and then reports error messages and execution statistics. Users can review the execution logs from the **Operator** Navigator or from **Oracle Enterprise Manager**.

It is important to understand that although the Agent can act as a transformation engine, it is rarely used for that purpose. Agents are installed at tactical locations in the information system to coordinate the integration processes and leverage existing systems. They are multithreaded, load-balanced, lightweight components in this distributed integration architecture.

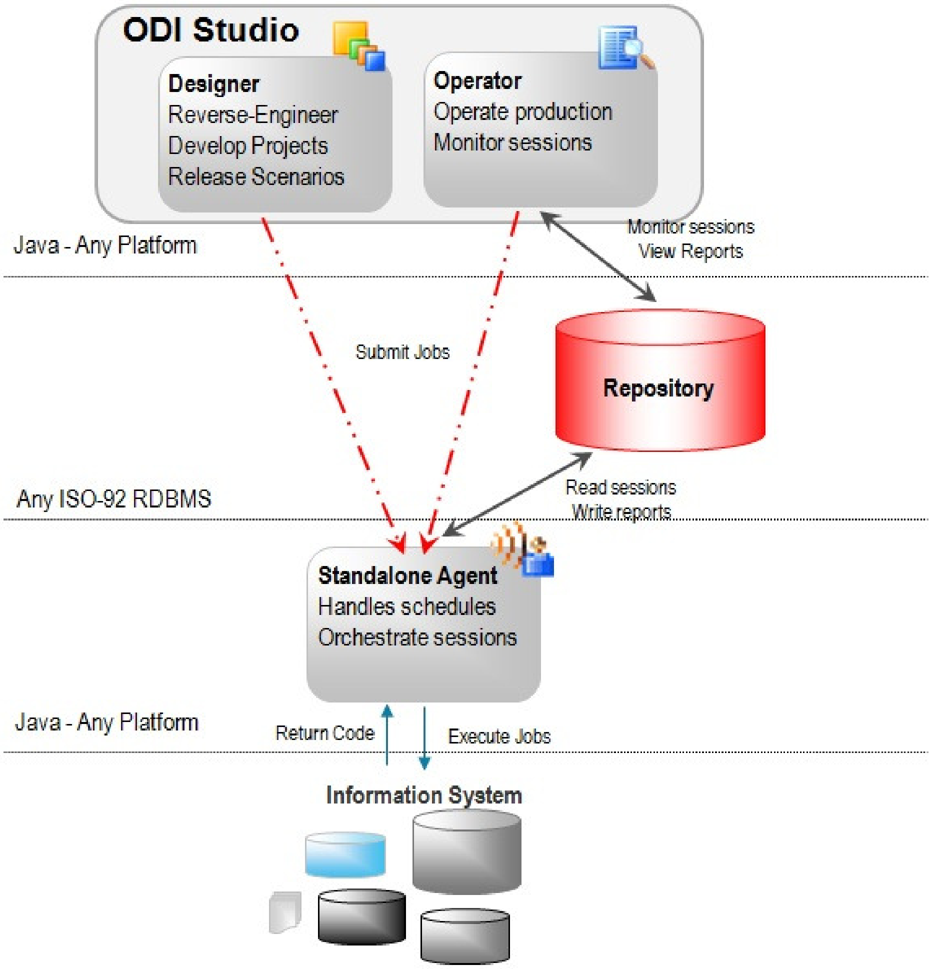

Figure 2 - Standalone Runtime Components

Repositories

The **Repository** consists of one or more **Master Repositories** and typically several **Work Repositories**. These repositories are sets of tables stored in relational database management systems such as Oracle, Microsoft SQL Server, IBM DB2 or MySQL. All objects that the ODI modules configure, develop, or use are stored in one of these repositories, and are accessed in client-server mode by the various components of the architecture.

The **Master Repository** contains the security information (user profiles, roles and privileges), the topology information (definitions of technologies and servers), and the source code for all versions of all ODI objects ever versioned. The information contained in the Master Repository is maintained with Topology and Security Navigators in ODI Studio as well as with ODI Console.

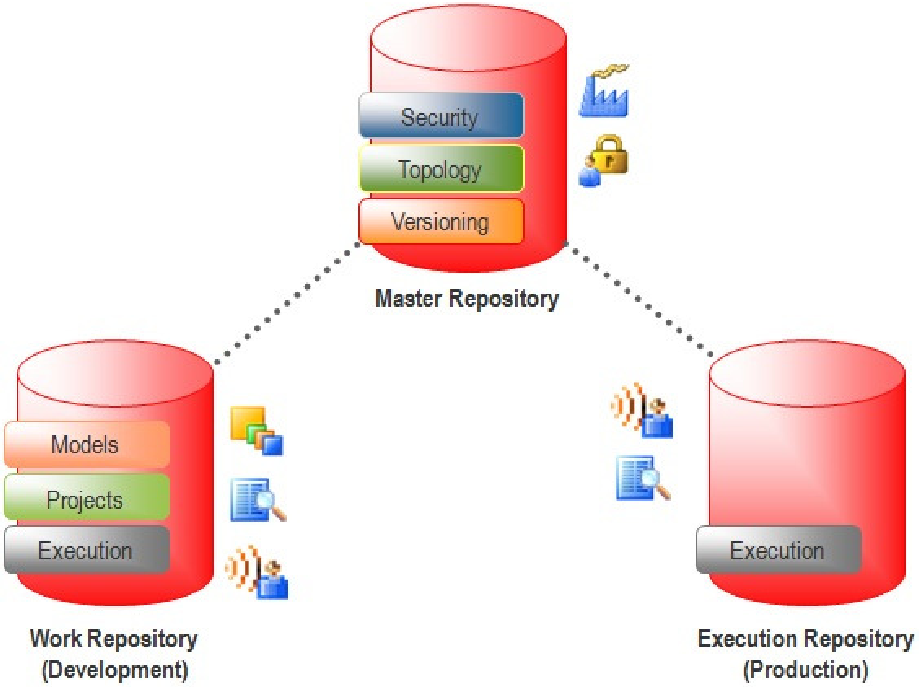

Figure 3 - Master repository and Work repositories

Project objects are stored in a **Work Repository**. Several Work Repositories can coexist in the same installation. This is useful for maintaining separate environments or to reflect a particular versioning lifecycle—for example, development, quality assurance, and production environments.

A work repository stores information for

- **Models** (i.e. metadata)—including datastores, columns, data integrity constraints, cross references, data lineage and impact analysis
- **Projects**—including mappings, packages, procedures, folders, knowledge modules, and variables
- **Runtime information**—including scenarios, load plans, scheduling information, and execution logs

Users manage the content of a work repository with the Designer and Operator Navigators in ODI Studio. The Agent at runtime also accesses work repositories. When a Work Repository is used only to store execution information (typically for production purposes), it is called an execution repository. An execution repository can be accessed at runtime with the Operator Navigator, the Agents and by Oracle Enterprise Manager. It is important to remember that each Work Repository is always attached to one and only one master repository.

Management Pack for Oracle Data Integrator

The **Management Pack for Oracle Data Integrator** leverages **Oracle Enterprise Manager Cloud Control** best-in-class application performance management, service level management and configuration management capabilities to provide a centralized management solution for Oracle Data Integrator.

The Management Pack for ODI allows administrators to monitor and manage all of their ODI components along with other Oracle products from a single administration console. In addition, it provides operators and developers end-to-end execution monitoring and drill-down capabilities into Oracle databases activity. End users can also leverage powerful alerting and notification capabilities to manage their Service Level Agreements (SLA).

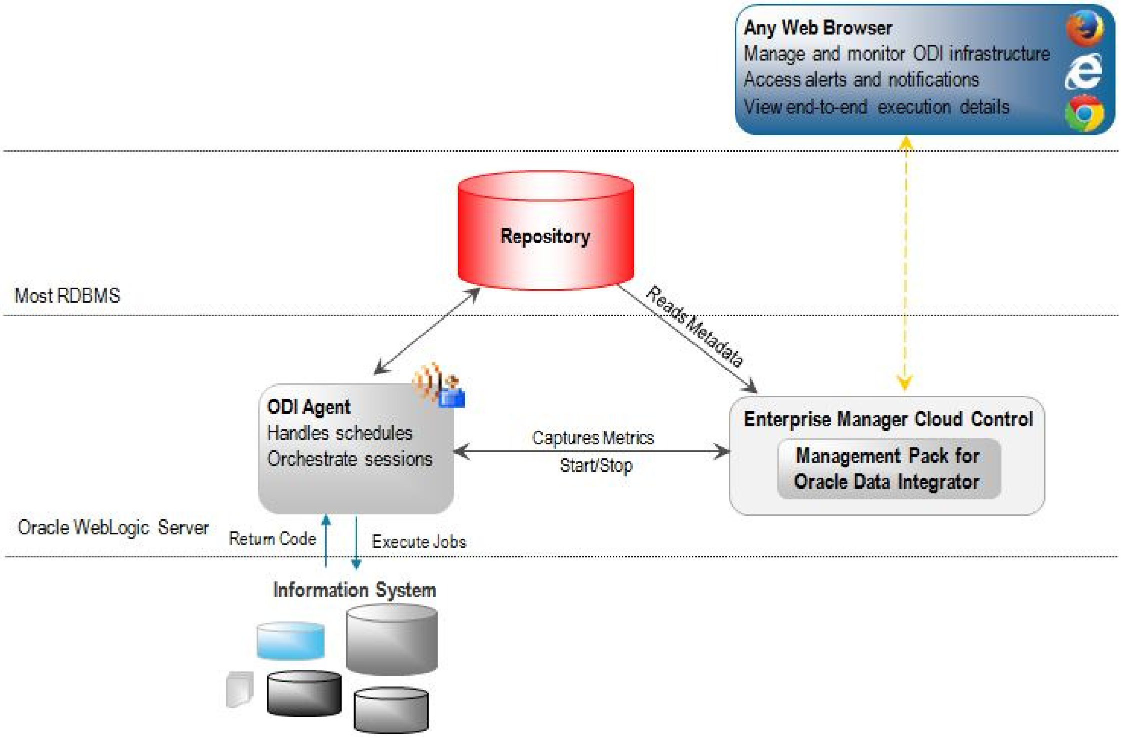

Figure 4 - Management Pack for Oracle Data Integrator

ODI Console

ODI Console is a Java EE application that can be deployed in a WebLogic domain and provides administrators with the ability to view and edit Topology objects such as Data Servers, Physical and Logical Schemas as well as to manage their repositories. In addition ODI Console allows users to browse Design-Time objects, including projects or models. Through its comprehensive Web interface, users can see flow maps and data lineage diagrams, trace the source of all data, and even drill down to the field level to understand the transformations used to build the data.

Other Features

Oracle Data Integrator also includes the following components and features:

- **Knowledge Modules** make it possible to quickly and easily integrate technologies, databases, and applications. They exist for a large range of platforms, including Oracle, Exadata, Big Data, Oracle Applications, Oracle GoldenGate, Teradata, Netezza, SAP ERP, SAP BW and XML.

- The **Changed Data Capture** (CDC) framework tracks changes in source systems and reduces the volume of processed data by extracting only the changed data. ODI can use native database features to capture changes or **Oracle GoldenGate**.

- The integration with **Oracle Business Intelligence Enterprise Edition** provides business users with report-to-source data lineage.

- Advanced **Web Services** capabilities offer best-in-class integration with Service-Oriented Architecture (SOA) environments

- The **Load Balancing** feature enables large volumes of data to be processed by automatically balancing the workload between several Agents.

- Advanced **Version Management** provides an interface to manage, safeguard, and replicate revisions of units of work, even in the largest development environments.

Conclusion

Oracle Data Integrator 12c is a lightweight, state-of-the-art data integration platform. With its superior performance and flexible architecture, Oracle Data Integrator can be used in various types of projects such as Data Warehousing, SOA, Business Intelligence or Application Integration.

Oracle is committed to developing practices and products that help protect the environment

ODI Architecture Overview
February 2014
Author: Julien Testut

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
oracle.com

SOFTWARE. HARDWARE. COMPLETE.