Oracle Data Integrator for Hyperion 101: An Introduction to the Basics

Brian Williams
Vertical Edge Consulting Group

Introduction

Oracle Data Integrator (ODI) is an enterprise class extract, transform, and load product that can be used for extensive data integrations and transformations across many different technologies in an organization. The complexity of the possible usage of ODI can be overwhelming for someone unfamiliar with advanced ETL tools. A Hyperion administrator with a background geared more towards the functional side of the system would typically be more concerned with answering the question "How can I use ODI with Hyperion?" For the purposes of this paper, we will focus on just the basics regarding how it can be used to build processes which update Hyperion applications.

Now that Hyperion Application Link (HAL) is near the end of its supported life, many organizations are in need of a replacement product for Hyperion integrations. For these users as well as new Hyperion implementations, there are a handful of options available. One of these products is ODI. The ODI Studio consists of three modules – Topology, Designer, and Operator. This first step in understanding ODI is to learn how each of these is used as well as other key terms and their meanings.

Topology

The ODI topology is the physical and logical representation of the architecture and components of the data sources and targets that you want to use in your integrations. A Hyperion Planning application and a text file or SQL table that contains the data to be loaded into the application would be represented in your topology configuration. The topology consists of Physical Architecture and Logical Architecture. Physical Architecture defines the different elements of the information system that it represents. Each type of data store – relational database, file format, or application software – is represented in the list of technologies in ODI physical architecture. Physical Architecture configuration in ODI contains connection information for the data store (server name, login ID and password, file directories, etc.). Physical Architecture also contains the physical components the store structure data, such as tables in a SQL database or dimensions in a Hyperion Planning application.

Logical Architecture allows you to identify a group of similar physical schemas as one single logical schema. In other words, datastores that are structurally identical but located in different physical locations can be grouped together with one logical schema. Every logical schema must be connected to a physical schema in ODI topology, and vice versa. Logical schemas are associated with corresponding physical schemas through what is called a Context in ODI topology.

Knowledge Modules

Knowledge Modules (KM) are code templates that are dedicated to an individual task in the data integration process. These are somewhat similar to adapters that integrate different types of technologies. A KM is reused across several interfaces that integrate the same types of technologies, for example SQL to SQL, or File to SQL integrations. An interface is simply the defined mappings of your data source to data target which processes your data integration. There are a few different types of knowledge modules that are used for different types of tasks, but there are three types in particular that apply to basic Hyperion integration – Reverse Engineering Knowledge Modules (RKM), Integration Knowledge Modules (IKM), and Loading Knowledge Modules (LKM).

Reverse Engineering Knowledge Module (RKM) -
RKM's are used to retrieve metadata, or data structures, from a datastore into ODI. SQL tables and Hyperion Planning dimensions are pulled into ODI used an RKM.

COLLABORATE 13 Copyright ©2013 by Brian Williams

Loading Knowledge Module (LKM) -
LKM's are used to load data from a source into a staging area before it is integrated into the target datastore. In ODI interfaces, a source or target datastore can also be used as the staging area.

Integration Knowledge Module (IKM) -
IKM's are used to integrate data from the staging area to the target datastore.
Let's use an example of a flat file containing a dimension hierarchy to be loaded into a Hyperion Planning application to explain how the KM's are utilized. The data columns and types of data in the flat file as well as the Hyperion Planning dimensionality would be retrieved into an ODI model using an RKM before you begin building your interface. During development of your interface, you would define an LKM to load the data from your source flat file into your defined staging area. From there, you would define an IKM to integrate the data from that staging area into your Hyperion Planning application, which is your target datastore. The same concepts described in the Hyperion Planning example can also be applied to Essbase and HFM applications.

ODI Designer

The ODI Designer is where you define all of the data models to be used in your integrations as well as the projects which contain all of your interfaces. It is also where all of the Knowledge Modules reside. In short, it is where all of the development and maintenance work on the integration processes is done. It contains a section for Models, which is where you data sources and targets have been defined and reverse engineered with RKM's, and a section for Projects, which is a way to organize your integration processes into folders.

ODI interfaces are developed and stored inside of projects in the Designer. The Mapping tab of the interface is where you map the columns from the data source into the columns of your data target. You can also apply filters and straightforward transformations between source and target columns in interfaces. Using our flat file to Planning example, you would drag the source flat file and target Hyperion Planning applications into the interface from the Models panel and then map the "Account" column in the flat file to the "Account" column in the Planning application if you were building an interface to update the Account dimension. The Flow tab of the interface is where you define the LKM and IKM to be used.

ODI Operator

The ODI Operator is the process monitoring tool within the ODI Studio. It is used by employees who monitor production and develop processes to check the execution of processes and debug issues. Inside of the Operator, there is a list of Sessions which contains all of the processes that have been executed and their status (success, failure, or in progress). Each session can be expanded to its individual steps to pinpoint where a process failed or simply to review execution statistics such as run time and number of updates. If a process failed, its session in the Operator will contain an error message to assist with troubleshooting.

Hyperion Planning Dimension Build Steps

The following is a high level list of steps that need to be followed when building an ODI interface to update a Hyperion Planning dimension, followed by an example of a source flat file and the resulting Hyperion Planning dimension updated using ODI.

- ☐ Topology Manager
  - ☐ Add Physical Architecture for Data Source and Target
    - ■ Add Data Server
    - ■ Create Physical Schema
  - ☐ Add Logical Architecture
  - ☐ Associate Physical and Logical Architecture with Context
- ☐ Designer

 Copyright ©2013 by Brian Williams

- Add Project and Project Folder(s)
- Import Knowledge Modules Into Project or Global Objects
- Insert Model Folder
- Insert Model
- For File type Models, add Datastore under Model
- Reverse Engineer Model or Datastore
- Set up Variables if Needed (Project Specific or Global Objects)
- Build and Execute Interface, Process, or Package

- Operator
  - Monitor Process Status and Troubleshoot

Using the concepts and steps discussed in this document, the flat file below was used to build the resulting Entity dimension in a Hyperion Planning application.

```
Department      Parent  Alias: Default  Storage Agg     REVPLAN COPPLAN
DRAL    Entity  DRAL - DIR-SUMMARY OF ALL DIRECTORS COMBINED    Store   +       1       0
D0000   DRAL    D0000 - UNSPECIFIED     Store   +       1       0
DRUN    DRAL    DRUN - CORPORATE (CORP) Store   +       1       0
D1000   DRUN    D1000 - CORP BOARD/TRUSTEES 1000        Store   +       1       0
D1001   DRUN    D1001 - CORP EVP/CEO    Store   +       1       0
D6100   DRUN    D6100 - CORP PUBLIC RELATIONS   Store   +       1       0
MG12    DRUN    MG12 - MANAGER-SUMMARY OF PUBLIC RELATIONS      Store   +       1       0
DRAD    DRAL    DRAD - REGULATORY & GOVERNMENT RELATIONS (R&G)  Store   +       1       0
D3000   DRAD    D3000 - R&G ADMIN & MNGT        Store   +       1       0
D3001   DRAD    D3001 - R&G COMPLIANCE  Store   +       1       0
D3515   DRAD    D3515 - R&G RISK MANAGEMENT     Store   +       1       0
D5600   DRAD    D5600 - R&G AUDIT SERVICES      Store   +       1       0
D7104   DRAD    D7104 - R&G KEY ACCOUNTS        Store   +       1       0
MG11    DRAD    MG11 - MANAGER-SUMMARY OF AUDIT SERVICES        Store   +       1       0
DRCE    DRAL    DRCE - CUSTOMER CARE (CCO)      Store   +       1       0
D4200   DRCE    D4200 - CCO ADMIN & MNGT        Store   +       1       0
D4240   DRCE    D4240 - CCO OP CUSTOMER PAYMENT CENTERS Store   +       1       0
D4261   DRCE    D4261 - CCO OP CONTACT CENTER   Store   +       1       0
D7101   DRCE    D7101 - CCO MES ENERGY SERVICES Store   +       1       0
D7102   DRCE    D7102 - CCO MES MARKETING       Store   +       1       0
D4251   DRCE    D4251 - CCO FS RESOURCE SCHEDULING      Store   +       1       0
D4252   DRCE    D4252 - CCO FS TECHNICIANS      Store   +       1       0
DRCO    DRAL    DRCO - FINANCE & ACCOUNTING (F&A)       Store   +       1       0
D1500   DRCO    D1500 - F&A CORPORATE COSTS     Store   +       1       0
D1501   DRCO    D1501 - F&A CORPORATE COSTS     Store   +       1       0
D5000   DRCO    D5000 - F&A ADMIN & MNGT        Store   +       1       0
D4253   DRCO    D4253 - F&A CREDIT & COLLECTIONS         Store   +       1       0
MG02    DRCO    MG02 - F&A FINANCIAL ACCOUNTING (FA)    Store   +       1       0
D5300   MG02    D5300 - F&A FA ADMIN & MNGT     Store   +       1       0
D5201   MG02    D5201 - F&A FA GEN ACCT & BUDGET        Store   +       1       0
D5211   MG02    D5211 - F&A FA PLANT ACCOUNTING Store   +       1       0
D5312   MG02    D5312 - F&A FA ACCOUNTING       Store   +       1       0
D5311   DRCO    D5311 - F&A CUSTOMER ACCOUNTING Store   +       1       0
MGB2    DRCO    MGB2 - F&A FLEET & MATERIALS MANAGEMENT (F&MM)  Store   +       1       0
D3631   MGB2    D3631 - F&A PURCHASING  Store   +       1       0
D3632   MGB2    D3632 - F&A INVENTORY/INVESTMENT RECOVERY        Store   +       1       0
```

Name (81 Total)	Alias (Default)	Data Storage	Security	Attributes	Plan Type
Entity		Never Share			1,2
DRAL	DRAL - DIR-SUMMARY OF ALL DIRECTORS COMBINED	Store			1
D0000	D0000 - UNSPECIFIED	Store			1
DRUN	DRUN - CORPORATE (CORP)	Store			1
D1000	D1000 - CORP BOARD/TRUSTEES 1000	Store			1
D1001	D1001 - CORP EVP/CEO	Store			1
D6100	D6100 - CORP PUBLIC RELATIONS	Store			1
MG12	MG12 - MANAGER-SUMMARY OF PUBLIC RELATIONS	Store			1
DRAD	DRAD - REGULATORY & GOVERNMENT RELATIONS (R&G)	Store			1
DRCE	DRCE - CUSTOMER CARE (CCO)	Store			1
DRCO	DRCO - FINANCE & ACCOUNTING (F&A)	Store			1
DRIT	DRIT - INFORMATION TECHNOLOGY (IT)	Store			1
DRHR	DRHR - HUMAN RESOURCES & FACILITIES SERVICES (HR&F)	Store			1
DROP	DROP - ELECTRIC OPERATIONS (E&O)	Store			1

 Copyright ©2013 by Brian Williams