



Hardware and Software
Engineered to Work Together

Oracle Big Data Fundamentals

Student Guide – Volume 2

D86898GC10

Edition 1.0 | May 2015 | D91414

Learn more from Oracle University at oracle.com/education/

Authors

Lauran K. Serhal
Brian Pottle
Suresh Mohan

Technical Contributors and Reviewers

Marty Gubar
Melliyal Annamalai
Sharon Stephen
Jean-Pierre Dijcks
Bruce Nelson
Daniel W McClary
Josh Spiegel
Anuj Sahni
Dave Segleau
Ashwin Agarwal
Salome Clement
Donna Carver
Alex Kotopoulos
Marcos Arancibia
Mark Hornick
Charlie Berger
Ryan Stark
Swarnapriya Shridhar
Branislav Valny
Dmitry Lychagin
Mirella Tumolo
S. Matt Taylor
Lakshmi Narapareddi
Drishya Tm

Graphic Editors

Rajiv Chandrabhanu
Maheshwari Krishnamurthy

Editors

Malavika Jinka
Smita Kommini
Arijit Ghosh

Publishers

Veena Narasimhan
Michael Sebastian Almeida
Syed Ali

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Disclaimer

This document contains proprietary information and is protected by copyright and other intellectual property laws. You may copy and print this document solely for your own use in an Oracle training course. The document may not be modified or altered in any way. Except where your use constitutes "fair use" under copyright law, you may not use, share, download, upload, copy, print, display, perform, reproduce, publish, license, post, transmit, or distribute this document in whole or in part without the express authorization of Oracle.

The information contained in this document is subject to change without notice. If you find any problems in the document, please report them in writing to: Oracle University, 500 Oracle Parkway, Redwood Shores, California 94065 USA. This document is not warranted to be error-free.

Restricted Rights Notice

If this documentation is delivered to the United States Government or anyone using the documentation on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS

The U.S. Government's rights to use, modify, reproduce, release, perform, display, or disclose these training materials are restricted by the terms of the applicable Oracle license agreement and/or the applicable U.S. Government contract.

Trademark Notice

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Contents

1 Introduction

| | |
|--|------|
| Objectives | 1-2 |
| Questions About You | 1-3 |
| Course Objectives | 1-4 |
| Course Road Map: Module 1 Big Data Management System | 1-5 |
| Course Road Map: Module 2 Data Acquisition and Storage | 1-6 |
| Course Road Map: Module 3 Data Access and Processing | 1-7 |
| Course Road Map: Module 4 Data Unification and Analysis | 1-8 |
| Course Road Map: Module 5 Using and Managing Oracle Big Data Appliance | 1-9 |
| The Oracle Big Data Lite Virtual Machine (Used in this Course) Home Page | 1-10 |
| Connecting to the Practice Environment | 1-11 |
| Starting the Oracle Big Data Lite Virtual Machine (VM) Used in this Course | 1-12 |
| Starting the Oracle Big Data Lite (BDLite) Virtual Machine (VM) Used in this Course | 1-13 |
| Accessing the Getting Started Page from the Oracle BDLite VM | 1-14 |
| Accessing the Practice Files | 1-15 |
| Accessing the /home/oracle/exercises Directory | 1-16 |
| Accessing /home/oracle/movie Directory | 1-17 |
| Appendices | 1-18 |
| Oracle Big Data Appliance Documentation | 1-19 |
| Additional Resources: Oracle Big Data Tutorials on Oracle Learning Library (OLL) | 1-21 |
| Practice 1-1: Overview | 1-23 |
| Summary | 1-24 |

2 Big Data and the Oracle Information Management System

| | |
|---------------------------------------|------|
| Course Road Map | 2-2 |
| Lesson Objectives | 2-3 |
| Big Data: A Strategic IM Perspective | 2-4 |
| Big Data | 2-5 |
| Characteristics of Big Data | 2-6 |
| Importance of Big Data | 2-8 |
| Big Data Opportunities: Some Examples | 2-9 |
| Big Data Challenges | 2-10 |
| Information Management Landscape | 2-12 |

| | |
|--|------|
| Extending the Boundaries of Information Management | 2-13 |
| A Simple Functional Model for Big Data | 2-14 |
| Oracle Information Management Conceptual Architecture | 2-16 |
| IM Architecture Design Pattern: Discovery Lab | 2-18 |
| IM Architecture Design Pattern: Information Platform | 2-19 |
| IM Architecture Design Pattern: Data Application | 2-20 |
| IM Architecture Design Pattern: Information Solution | 2-21 |
| IM Architecture Design Pattern: Real-Time Events | 2-22 |
| Design Patterns to Component Usage Map | 2-23 |
| Big Data Adoption and Implementation Patterns | 2-24 |
| IM Architecture Data Approaches: Schema-on-Write vs Schema-on-Read | 2-26 |
| Course Approach: Big Data Project Phases | 2-28 |
| Goal of Oracle's IM System for Big Data | 2-29 |
| Additional Resources | 2-30 |
| Summary | 2-31 |

3 Using Oracle Big Data Lite Virtual Machine

| | |
|--|------|
| Course Road Map | 3-2 |
| Objectives | 3-3 |
| Lesson Agenda | 3-4 |
| Oracle Big Data Lite Virtual Machine: Introduction | 3-5 |
| Oracle Big Data Lite 4.0.1 VM Components | 3-6 |
| Initializing the Environment for the Oracle Big Data Lite VM | 3-7 |
| Initializing the Environment | 3-8 |
| Lesson Agenda | 3-9 |
| Oracle MoviePlex Case Study: Introduction | 3-10 |
| Introduction | 3-11 |
| Big Data Challenge | 3-12 |
| Derive Value from Big Data | 3-13 |
| Oracle MoviePlex: Goal | 3-14 |
| Oracle MoviePlex: Big Data Challenges | 3-15 |
| Oracle MoviePlex: Architecture | 3-16 |
| Oracle MoviePlex: Data Generation | 3-17 |
| Oracle MoviePlex: Data Generation Format | 3-18 |
| Oracle MoviePlex Application | 3-19 |
| Summary | 3-20 |
| Practice 3: Overview | 3-21 |

- 4 Introduction to the Big Data Ecosystem**
 - Course Road Map 4-2
 - Objectives 4-3
 - Computer Clusters 4-4
 - Distributed Computing 4-5
 - Apache Hadoop 4-6
 - Types of Analysis That Use Hadoop 4-7
 - Apache Hadoop Ecosystem 4-8
 - Apache Hadoop Core Components 4-9
 - HDFS Key Definitions 4-11
 - NameNode (NN) 4-12
 - DataNodes (DN) 4-13
 - MapReduce Framework 4-14
 - Benefits of MapReduce 4-15
 - MapReduce Job 4-16
 - Simple Word Count MapReduce: Example 4-17
 - MapReduce Versions 4-19
 - Choosing a Hadoop Distribution and Version 4-20
 - Additional Resources: Cloudera Distribution 4-21
 - Additional Resources: Apache Hadoop 4-22
 - Cloudera's Distribution Including Apache Hadoop (CDH) 4-23
 - CDH Architecture 4-24
 - CDH Components 4-25
 - CDH Architecture 4-26
 - CDH Components 4-28
 - Where to Go for More Information? 4-29
 - Summary 4-30
- 5 Introduction to the Hadoop Distributed File System (HDFS)**
 - Course Road Map 5-2
 - Objectives 5-3
 - Agenda 5-4
 - HDFS: Characteristics 5-5
 - HDFS Deployments: High Availability (HA) and Non-HA 5-7
 - HDFS Key Definitions 5-8
 - NameNode (NN) 5-9
 - Functions of the NameNode 5-10
 - Secondary NameNode (Non-HA) 5-11
 - DataNodes (DN) 5-12
 - Functions of DataNodes 5-13
 - NameNode and Secondary NameNodes 5-14

| | |
|---|------|
| Storing and Accessing Data Files in HDFS | 5-15 |
| Secondary NameNode, Checkpoint Node, and Backup Node | 5-16 |
| HDFS Architecture: HA | 5-17 |
| HDFS High Availability (HA) Using the Quorum Journal Manager (QJM) | 5-18 |
| HDFS High Availability (HA) Using the Quorum Journal Manager (QJM) Feature | 5-19 |
| Configuring an HA Cluster Hardware Resources | 5-22 |
| Enabling HDFS HA | 5-23 |
| Data Replication Rack-Awareness in HDFS | 5-25 |
| Data Replication Process | 5-26 |
| Accessing HDFS | 5-27 |
| Agenda | 5-28 |
| HDFS Commands | 5-29 |
| The File System Namespace: The HDFS FS (File System) Shell Interface | 5-30 |
| Accessing HDFS | 5-32 |
| FS Shell Commands | 5-33 |
| Basic File System Operations: Examples | 5-34 |
| Sample FS Shell Commands | 5-35 |
| Basic File System Operations: Examples | 5-36 |
| HDFS Administration Commands | 5-38 |
| Using the hdfs fsck Command: Example | 5-39 |
| HDFS Features and Benefits | 5-40 |
| Summary | 5-41 |
| Practice 5: Overview | 5-42 |

6 Acquire Data Using CLI, Fuse DFS, and Flume

| | |
|--|------|
| Course Road Map | 6-2 |
| Objectives | 6-3 |
| Reviewing the Command Line Interface (CLI) | 6-4 |
| Viewing File System Contents Using the CLI | 6-5 |
| Loading Data Using the CLI | 6-6 |
| What is Fuse DFS? | 6-7 |
| Enabling Fuse DFS on Big Data Lite | 6-8 |
| Using Fuse DFS | 6-9 |
| What is Flume? | 6-10 |
| Flume: Architecture | 6-11 |
| Flume Sources (Consume Events) | 6-12 |
| Flume Channels (Hold Events) | 6-13 |
| Flume Sinks (Deliver Events) | 6-14 |
| Flume: Data Flows | 6-15 |
| Configuring Flume | 6-16 |

Exploring a flume*.conf File 6-17

Additional Resources 6-18

Summary 6-19

Practice 6: Overview 6-20

7 Acquire and Access Data Using Oracle NoSQL Database

Course Road Map 7-2

Objectives 7-3

What is a NoSQL Database? 7-4

RDBMS Compared to NoSQL 7-5

HDFS Compared to NoSQL 7-6

Oracle NoSQL Database 7-7

Points to Consider Before Choosing NoSQL 7-8

NoSQL Key-Value Data Model 7-9

Acquiring and Accessing Data in a NoSQL DB 7-11

Primary (Parent) Table Data Model 7-12

Table Data Model: Child Tables 7-13

Creating Tables 7-14

Creating Tables: Two Options 7-15

Data Definition Language (DDL) Commands 7-16

CREATE TABLE 7-17

Accessing the CLI 7-19

Executing a DDL Command 7-20

Viewing Table Descriptions 7-21

Recommendation: Using Scripts 7-22

Loading Data Into Tables 7-23

Accessing the KVStore 7-24

Introducing the TableAPI 7-25

Write Operations: put() Methods 7-26

Writing Rows to Tables: Steps 7-27

Constructing a Handle 7-28

Creating Row Object, Adding Fields, and Writing Record 7-29

Reading Data from Tables 7-30

Read Operations: get() Methods 7-31

Retrieving Table Data: Steps 7-32

Retrieving Single a Row 7-33

Retrieving Multiple Rows 7-34

Retrieving Child Tables 7-35

Removing Data From Tables 7-36

Delete Operations: 3 TableAPIs 7-37

Deleting Row(s) From a Table: Steps 7-38

Additional Resources 7-39
Summary 7-40
Practice 7 Overview 7-41

8 Primary Administrative Tasks for Oracle NoSQL Database

Course Road Map 8-2
Objectives 8-3
Installation Planning: KVStore Analysis 8-4
InitialCapacityPlanning Spreadsheet 8-5
Planning Spreadsheet Sections 8-6
Next Topic 8-7
Configuration Requirements 8-8
Determine the Number of Shards 8-9
Determine # of Partitions and Replication Factor 8-10
Determine # of Storage Nodes 8-11
Installation and Configuration Steps 8-12
Step 1: Creating Directories 8-13
Step 2: Extracting Software 8-14
Step 3: Verifying the Installation 8-15
Step 4: Configuring Nodes (Using the makebootconfig Utility) 8-16
Using the makebootconfig Utility 8-18
Starting the Storage Node Agents 8-19
Pinging the Replication Nodes 8-20
Next Topic 8-21
Configuration and Monitoring Tools 8-22
Steps to Deploy a KVStore 8-23
Introducing Plans 8-24
States of a Plan 8-25
Starting the Configuration Tool 8-26
Configuring KVStore 8-27
Creating a Zone 8-28
Deploying Storage and Admin Nodes 8-29
Creating a Storage Pool 8-30
Joining Nodes to the Storage Pool 8-31
Creating a Topology 8-32
Deploying the KVStore 8-33
Testing the KVStore 8-34
Additional Resources 8-35
Summary 8-36

9 Introduction to MapReduce

Course Road Map 9-2
Objectives 9-3
MapReduce 9-4
MapReduce Architecture 9-5
MapReduce Version 1 (MRv1) Architecture 9-6
MapReduce Phases 9-7
MapReduce Framework 9-8
Parallel Processing with MapReduce 9-9
MapReduce Jobs 9-10
Interacting with MapReduce 9-11
MapReduce Processing 9-12
MapReduce (MRv1) Daemons 9-13
Hadoop Basic Cluster (MRv1): Example 9-14
MapReduce Application Workflow 9-15
Data Locality Optimization in Hadoop 9-17
MapReduce Mechanics: Deck of Cards Example 9-18
MapReduce Mechanics Example: Assumptions 9-19
MapReduce Mechanics: The Map Phase 9-20
MapReduce Mechanics: The Shuffle and Sort Phase 9-21
MapReduce Mechanics: The Reduce Phase 9-22
Word Count Process: Example 9-23
Submitting a MapReduce 9-24
Summary 9-25
Practice 9: Overview 9-26

10 Resource Management Using YARN

Course Road Map 10-2
Objectives 10-3
Agenda 10-4
Apache Hadoop YARN: Overview 10-5
MapReduce 2.0 (MRv2) or YARN (Yet Another Resource Negotiator)
Architecture 10-7
MapReduce 2.0 (MRv2) or YARN (Yet Another Resource Negotiator)
Daemons 10-8
Hadoop Basic Cluster YARN (MRv2): Example 10-9
YARN Versus MRv1 Architecture 10-10
YARN (MRv2) Architecture 10-11
MapReduce 2.0 (MRv2) or YARN Daemons 10-13
YARN (MRv2) Daemons 10-14
YARN: Features 10-15

| | |
|--|-------|
| Launching an Application on a YARN Cluster | 10-16 |
| MRv1 Versus MRv2 | 10-18 |
| Agenda | 10-19 |
| Job Scheduling in YARN | 10-20 |
| YARN Fair Scheduler | 10-21 |
| Cloudera Manager Resource Management Features | 10-23 |
| Static Service Pools | 10-25 |
| Working with the Fair Scheduler | 10-26 |
| Cloudera Manager Dynamic Resource Management: Example | 10-27 |
| Submitting a Job to hrpool By User lucy from the hr Group | 10-33 |
| Monitoring the Status of the Submitted MapReduce Job | 10-34 |
| Examining the marketingpool | 10-35 |
| Submitting a Job to marketingpool By User lucy from the hr Group | 10-36 |
| Monitoring the Status of the Submitted MapReduce Job | 10-37 |
| Submitting a Job to marketingpool By User bob from the marketing Group | 10-38 |
| Monitoring the Status of the Submitted MapReduce Job | 10-39 |
| Delay Scheduling | 10-40 |
| Agenda | 10-41 |
| YARN application Command | 10-42 |
| YARN application Command: Example | 10-43 |
| Monitoring an Application Using the UI | 10-45 |
| The Scheduler: BDA Example | 10-46 |
| Summary | 10-47 |
| Practice 10 | 10-48 |

11 Overview of Hive and Pig

| | |
|---|-------|
| Course Road Map | 11-2 |
| Objectives | 11-3 |
| Hive | 11-4 |
| Use Case: Storing Clickstream Data | 11-5 |
| Defining Tables over HDFS | 11-6 |
| Hive: Data Units | 11-8 |
| The Hive Metastore Database | 11-9 |
| Hive Framework | 11-10 |
| Creating a Hive Database | 11-11 |
| Data Manipulation in Hive | 11-12 |
| Data Manipulation in Hive: Nested Queries | 11-13 |
| Steps in a Hive Query | 11-14 |
| Hive-Based Applications | 11-15 |
| Hive: Limitations | 11-16 |
| Pig: Overview | 11-17 |

Pig Latin 11-18
Pig Applications 11-19
Running Pig Latin Statements 11-20
Pig Latin: Features 11-21
Working with Pig 11-22
Summary 11-23
Practice 11: Overview 11-24

12 Overview of Cloudera Impala

Course Road Map 12-2
Objectives 12-3
Hadoop: Some Data Access/Processing Options 12-4
Cloudera Impala 12-5
Cloudera Impala: Key Features 12-6
Cloudera Impala: Supported Data Formats 12-7
Cloudera Impala: Programming Interfaces 12-8
How Impala Fits Into the Hadoop Ecosystem 12-9
How Impala Works with Hive 12-10
How Impala Works with HDFS and HBase 12-11
Summary of Cloudera Impala Benefits 12-12
Impala and Hadoop: Limitations 12-13
Summary 12-14

13 Using Oracle XQuery for Hadoop

Course Road Map 13-2
Objectives 13-3
XML 13-4
Simple XML Document: Example 13-5
XML Elements 13-6
Markup Rules for Elements 13-7
XML Attributes 13-8
XML Path Language 13-9
XPath Terminology: Node Types 13-10
XPath Terminology: Family Relationships 13-11
XPath Expressions 13-12
Location Path Expression: Example 13-13
XQuery: Review 13-14
XQuery Terminology 13-15
XQuery Review: books.xml Document Example 13-16
FLWOR Expressions: Review 13-17
Oracle XQuery for Hadoop (OXH) 13-18

| | |
|--|-------|
| OXH Features | 13-19 |
| Oracle XQuery for Hadoop Data Flow | 13-20 |
| Using OXH | 13-21 |
| OXH Installation | 13-22 |
| OXH Functions | 13-23 |
| OXH Adapters | 13-24 |
| Running a Query: Syntax | 13-25 |
| OXH: Configuration Properties | 13-26 |
| XQuery Transformation and Basic Filtering: Example | 13-27 |
| Viewing the Completed Application in YARN | 13-30 |
| Calling Custom Java Functions from XQuery | 13-31 |
| Additional Resources | 13-32 |
| Summary | 13-33 |
| Practice 13: Overview | 13-34 |

14 Overview of Solr

| | |
|-----------------------------------|-------|
| Course Road Map | 14-2 |
| Objectives | 14-3 |
| Apache Solr (Cloudera Search) | 14-4 |
| Cloudera Search: Key Capabilities | 14-5 |
| Cloudera Search: Features | 14-6 |
| Cloudera Search Tasks | 14-8 |
| Indexing in Cloudera Search | 14-9 |
| Types of Indexing | 14-10 |
| The solrctl Command | 14-12 |
| SchemaXML File | 14-13 |
| Creating a Solr Collection | 14-14 |
| Using OXH with Solr | 14-15 |
| Using Solr with Hue | 14-16 |
| Summary | 14-18 |
| Practice 14: Overview | 14-19 |

15 Apache Spark

| | |
|---|------|
| Course Road Map | 15-2 |
| Objectives | 15-3 |
| Apache Spark | 15-4 |
| Introduction to Spark | 15-5 |
| Spark: Components for Distributed Execution | 15-6 |
| Resilient Distributed Dataset (RDD) | 15-7 |
| RDD Operations | 15-8 |
| Characteristics of RDD | 15-9 |

Directed Acyclic Graph Execution Engine 15-10
Scala Language: Overview 15-11
Scala Program: Word Count Example 15-12
Spark Shells 15-13
Summary 15-14
Practice 15: Overview 15-15

16 Options for Integrating Your Big Data

Course Road Map 16-2
Objectives 16-3
Unifying Data: A Typical Requirement 16-4
Introducing Data Unification Options 16-6
Data Unification: Batch Loading 16-7
Sqoop 16-8
Oracle Loader for Hadoop (OLH) 16-9
Copy to BDA 16-10
Data Unification: Batch and Dynamic Loading 16-11
Oracle SQL Connector for Hadoop 16-12
Data Unification: ETL and Synchronization 16-13
Oracle Data Integrator with Big Data Heterogeneous Integration with
Hadoop Environments 16-14
Data Unification: Dynamic Access 16-16
Oracle Big Data SQL: A New Architecture 16-17
When To Use Different Oracle Technologies? 16-18
Summary 16-19

17 Overview of Apache Sqoop

Course Road Map 17-2
Objectives 17-3
Apache Sqoop 17-4
Sqoop Components 17-5
Sqoop Features 17-6
Sqoop: Connectors 17-7
Importing Data into Hive 17-8
Sqoop: Advantages 17-9
Summary 17-10

18 Using Oracle Loader for Hadoop (OLH)

Course Road Map 18-2
Objectives 18-3
Oracle Loader for Hadoop 18-4

| | |
|--------------------------------|-------|
| Software Prerequisites | 18-5 |
| Modes of Operation | 18-6 |
| OLH: Online Database Mode | 18-7 |
| Running an OLH Job | 18-8 |
| OLH Use Cases | 18-9 |
| Load Balancing in OLH | 18-10 |
| Input Formats | 18-11 |
| OLH: Offline Database Mode | 18-12 |
| Offline Load Advantages in OLH | 18-13 |
| OLH Versus Sqoop | 18-14 |
| OLH: Performance | 18-15 |
| Summary | 18-16 |
| Practice 18: Overview | 18-17 |

19 Using Copy to BDA

| | |
|---------------------------------------|-------|
| Course Road Map | 19-2 |
| Objectives | 19-3 |
| Copy to BDA | 19-4 |
| Requirements for Using Copy to BDA | 19-5 |
| How Does Copy to BDA Work? | 19-6 |
| Copy to BDA: Functional Steps | 19-7 |
| Step 1: Identify the Target Directory | 19-8 |
| Step 2: Create an External Table | 19-9 |
| Step 3: Copy Files to HDFS | 19-10 |
| Step 4: Create a Hive External Table | 19-11 |
| Oracle to Hive Data Type Conversions | 19-12 |
| Querying the Data in Hive | 19-13 |
| Summary | 19-14 |
| Practice 19: Overview | 19-15 |

20 Using Oracle SQL Connector for HDFS

| | |
|---|-------|
| Course Road Map | 20-2 |
| Objectives | 20-3 |
| Oracle SQL Connector for HDFS | 20-4 |
| OSCH Architecture | 20-5 |
| Using OSCH: Two Simple Steps | 20-6 |
| Using OSCH: Creating External Directory | 20-7 |
| Using OSCH: Database Objects and Grants | 20-8 |
| Using OSCH: Supported Data Formats | 20-9 |
| Using OSCH: HDFS Text File Support | 20-10 |
| Using OSCH: Hive Table Support | 20-12 |

Using OSCH: Partitioned Hive Table Support 20-14
OSCH: Features 20-15
Parallelism and Performance 20-16
OSCH: Performance Tuning 20-17
OSCH: Key Benefits 20-18
Loading: Choosing a Connector 20-19
Summary 20-20
Practice 20: Overview 20-21

21 Using Oracle Data Integrator and Oracle GoldenGate with Hadoop

Course Road Map 21-2
Objectives 21-3
Oracle Data Integrator 21-4
ODI's Declarative Design 21-5
ODI Knowledge Modules (KMs) Simpler Physical Design / Shorter Implementation Time 21-6
Using ODI with Big Data Heterogeneous Integration with Hadoop Environments 21-7
Using ODI Studio 21-8
ODI Studio Components: Overview 21-9
ODI Studio: Big Data Knowledge Modules 21-10
Using OGG with Big Data 21-12
Resources 21-13
Summary 21-14
Practice 21: Overview 21-15

22 Using Oracle Big Data SQL

Course Road Map 22-2
Objectives 22-3
Barriers to Effective Big Data Adoption 22-4
Overcoming Big Data Barriers 22-5
Oracle Big Data SQL 22-6
Goal and Benefits 22-7
Using Oracle Big Data SQL 22-8
Configuring Oracle Big Data SQL 22-9
Task 1: Create System Directories on Exadata 22-10
Task 2: Deploy Configuration Files 22-11
Task 3: Create Oracle Directory Objects 22-12
Task 4: Install Required Software 22-13
Create External Tables Over HDFS Data and Query the Data 22-14
Using Access Parameters with oracle_hdfs 22-15

| | |
|--|-------|
| Create External Tables to Leverage the Hive Metastore and Query the Data | 22-16 |
| Using Access Parameters with oracle_hive | 22-17 |
| Automating External Table Creation | 22-19 |
| Applying Oracle Database Security Policies | 22-20 |
| Viewing the Results | 22-21 |
| Applying Redaction Policies to Data in Hadoop | 22-22 |
| Viewing Results from the Hive (Avro) Source | 22-23 |
| Viewing the Results from Joined RDBMS and HDFS Data | 22-24 |
| Summary | 22-25 |
| Practice 22: Overview | 22-26 |

23 Using Oracle Advanced Analytics: Oracle Data Mining and Oracle R Enterprise

| | |
|---|-------|
| Course Road Map | 23-2 |
| Objectives | 23-3 |
| Oracle Advanced Analytics (OAA) | 23-4 |
| OAA: Oracle Data Mining | 23-5 |
| What Is Data Mining? | 23-6 |
| Common Uses of Data Mining | 23-7 |
| Defining Key Data Mining Properties | 23-8 |
| Data Mining Categories | 23-10 |
| Supervised Data Mining Techniques | 23-11 |
| Supervised Data Mining Algorithms | 23-12 |
| Unsupervised Data Mining Techniques | 23-13 |
| Unsupervised Data Mining Algorithms | 23-14 |
| Oracle Data Mining: Overview | 23-15 |
| Oracle Data Miner GUI | 23-16 |
| ODM SQL Interface | 23-17 |
| Oracle Data Miner 4.1 Big Data Enhancement | 23-18 |
| Example Workflow Using JSON Query Node | 23-19 |
| ODM Resources | 23-20 |
| Practice: Overview (ODM) | 23-21 |
| OAA: Oracle R Enterprise | 23-22 |
| What Is R? | 23-23 |
| Who Uses R? | 23-24 |
| Why Do Statisticians, Data Analysts, Data Scientists Use R? | 23-25 |
| Limitations of R | 23-26 |
| Oracle's Strategy for the R Community | 23-27 |
| Oracle R Enterprise | 23-28 |
| ORE: Software Features | 23-29 |
| ORE Packages | 23-30 |
| Functions for Interacting with Oracle Database | 23-31 |

| | |
|---|-------|
| ORE: Target Environment | 23-32 |
| ORE: Data Sources | 23-33 |
| ORE and Hadoop | 23-34 |
| ORAAH: Architecture | 23-35 |
| ORAAH Package | 23-36 |
| HDFS Connectivity and Interaction | 23-37 |
| ORAAH Functions for HDFS Interaction | 23-38 |
| ORAAH Functions for Predictive Algorithms | 23-39 |
| Hadoop Connectivity and Interaction | 23-40 |
| Word Count: Example Without ORAAH | 23-41 |
| Word Count: Example with ORAAH | 23-42 |
| ORE Resources | 23-43 |
| Practice: Overview (ORE) | 23-44 |
| Summary | 23-45 |

24 Introducing Oracle Big Data Discovery

| | |
|-------------------------------------|-------|
| Course Road Map | 24-2 |
| Objectives | 24-3 |
| Oracle Big Data Discovery | 24-4 |
| Find Data | 24-5 |
| Explore Data | 24-6 |
| Transform and Enrich Data | 24-7 |
| Discover Information | 24-8 |
| Share Insights | 24-9 |
| BDD: Technical Innovation on Hadoop | 24-10 |
| Additional Resources | 24-11 |
| Summary | 24-12 |

25 Introduction to the Oracle Big Data Appliance (BDA)

| | |
|--|-------|
| Course Road Map | 25-2 |
| Objectives | 25-3 |
| Oracle Big Data Appliance | 25-4 |
| Oracle Big Data Appliance: Key Component of the Big Data Management System | 25-5 |
| Oracle-Engineered Systems for Big Data | 25-6 |
| The Available Oracle BDA Configurations | 25-7 |
| Using the Mammoth Utility | 25-8 |
| Using Oracle BDA Configuration Generation Utility | 25-10 |
| Configuring Oracle Big Data Appliance | 25-11 |
| The Generated Configuration Files | 25-13 |
| The Oracle BDA Configuration Generation Utility Pages | 25-15 |

| | |
|---|-------|
| Using Oracle BDA Configuration Generation Utility: The Customer Details Page | 25-16 |
| Using Oracle BDA Configuration Generation Utility: The Hardware Selections Page | 25-17 |
| The Oracle BDA Configuration Generation Utility: The Define Clusters Page | 25-18 |
| The Oracle BDA Configuration Generation Utility: The Cluster n Page | 25-19 |
| BDA Configurations: Full Rack | 25-20 |
| BDA Configurations: Starter Rack | 25-21 |
| BDA Configurations: In-Rack Expansion | 25-22 |
| BDA Starter Rack: Hadoop Cluster Only | 25-23 |
| BDA Starter Rack: NoSQL Cluster Only | 25-24 |
| Big Data Appliance: Horizontal Scale-Out Model | 25-25 |
| Big Data Appliance: Software Components | 25-26 |
| Oracle Big Data Appliance and YARN | 25-27 |
| Stopping the YARN Service | 25-28 |
| Critical and Noncritical Nodes in an Oracle BDA CDH Cluster | 25-29 |
| First NameNode and Second NameNode | 25-30 |
| First ResourceManager and Second ResourceManager | 25-31 |
| Hardware Failure in Oracle NoSQL | 25-32 |
| Oracle Integrated Lights Out Manager (ILOM): Overview | 25-33 |
| Oracle ILOM Users | 25-34 |
| Connecting to Oracle ILOM Using the Network | 25-35 |
| Oracle ILOM: Integrated View | 25-36 |
| Monitoring the Health of Oracle BDA: Management Utilities | 25-37 |
| Big Data Appliance: Security Implementation | 25-38 |
| Big Data Appliance: Usage Guidelines | 25-39 |
| Summary | 25-40 |
| Practice 25: Overview | 25-41 |

26 Managing Oracle BDA

| | |
|-----------------------------------|-------|
| Course Road Map | 26-2 |
| Objectives | 26-3 |
| Lesson Agenda | 26-4 |
| Mammoth Utility | 26-5 |
| Installation types | 26-6 |
| Mammoth Code: Examples | 26-7 |
| Mammoth Installation Steps | 26-8 |
| Lesson Agenda | 26-10 |
| Monitoring Oracle BDA | 26-11 |
| Oracle BDA Command-Line Interface | 26-12 |
| bdacli | 26-13 |

setup-root-ssh 26-14
Lesson Agenda 26-15
Monitor BDA with Oracle Enterprise Manager 26-16
OEM: Web and Command-Line Interfaces 26-17
OEM: Hardware Monitoring 26-18
Hadoop Cluster Monitoring 26-19
Lesson Agenda 26-20
Managing CDH Operations 26-21
Using Cloudera Manager 26-22
Monitoring Oracle BDA Status 26-23
Performing Administrative Tasks 26-24
Managing Services 26-25
Lesson Agenda 26-26
Monitoring MapReduce Jobs 26-27
Monitoring the Health of HDFS 26-28
Lesson Agenda 26-29
Cloudera Hue 26-30
Hive Query Editor (Hue) Interface 26-31
Logging in to Hue 26-32
Lesson Agenda 26-33
Starting Oracle BDA 26-34
Stopping Oracle BDA 26-35
BDA Port Assignments 26-36
Summary 26-37
Practice 26: Overview 26-38

27 Balancing MapReduce Jobs

Course Road Map 27-2
Objectives 27-3
Ideal World: Neatly Balanced MapReduce Jobs 27-4
Real World: Skewed Data and Unbalanced Jobs 27-5
Data Skew 27-6
Data Skew Can Slow Down the Entire Hadoop Job 27-7
Perfect Balance 27-8
How Does the Perfect Balance Work? 27-9
Using Perfect Balance 27-10
Application Requirements for Using Perfect Balance 27-11
Perfect Balance: Benefits 27-12
Using Job Analyzer 27-13
Getting Started with Perfect Balance 27-14
Using Job Analyzer 27-16

| | |
|--|-------|
| Environmental Setup for Perfect Balance and Job Analyzer | 27-17 |
| Running Job Analyzer as a Stand-alone Utility to Measure Data Skew in Unbalanced Jobs | 27-18 |
| Using Job Analyzer as a Stand-Alone Utility: Example with a YARN Cluster | 27-19 |
| Configuring Perfect Balance | 27-20 |
| Using Perfect Balance to Run a Balanced MapReduce Job | 27-21 |
| Running a Job Using Perfect Balance: Examples | 27-23 |
| Perfect Balance–Generated Reports | 27-25 |
| The Job Analyzer Reports: Structure of the Job Output Directory | 27-26 |
| Reading the Job Analyzer Reports | 27-27 |
| Reading the Job Analyzer Report in HDFS Using a Web Browser | 27-28 |
| Reading the Job Analyzer Report in the Local File System in a Web Browser | 27-29 |
| Looking for Skew Indicators in the Job Analyzer Reports | 27-30 |
| Job Analyzer Sample Reports | 27-31 |
| Collecting Additional Metrics with Job Analyzer | 27-32 |
| Using Data from Additional Metrics | 27-33 |
| Using Perfect Balance API | 27-34 |
| Chopping | 27-35 |
| Disabling Chopping | 27-36 |
| Troubleshooting Jobs Running with Perfect Balance | 27-37 |
| Perfect Balance Examples Available with Installation | 27-38 |
| Summary | 27-40 |
| Practice 27: Overview | 27-41 |

28 Securing Your Data

| | |
|--------------------------------------|-------|
| Course Road Map | 28-2 |
| Objectives | 28-3 |
| Security Trends | 28-4 |
| Security Levels | 28-5 |
| Outline | 28-6 |
| Relaxed Security | 28-7 |
| Authentication with Relaxed Security | 28-8 |
| Authorization | 28-9 |
| HDFS ACLs | 28-10 |
| Changing Access Privileges | 28-11 |
| Relaxed Security Summary | 28-12 |
| Challenges with Relaxed Security | 28-13 |
| BDA Secure Installation | 28-14 |
| Kerberos Key Definitions | 28-15 |
| Strong Authentication with Kerberos | 28-16 |
| Snapshot of Principals in KDC | 28-17 |

| | |
|--|-------|
| Authentication with Kerberos | 28-18 |
| User Authentication: Examples | 28-19 |
| Service Authentication and Keytabs | 28-20 |
| Review TGT Cache | 28-21 |
| Ticket Renewal | 28-22 |
| Adding a New User | 28-23 |
| Example: Adding a New User | 28-24 |
| Example: Adding a User to Hue | 28-25 |
| Authorization | 28-26 |
| Sentry Authorization Features | 28-27 |
| Sentry Configuration | 28-28 |
| Users, Groups, and Roles | 28-29 |
| Sentry Example: Overview | 28-30 |
| Example: Users, Roles, and Groups | 28-31 |
| 1. Creating Roles | 28-32 |
| 2. Assigning Roles to Groups | 28-33 |
| Show Roles for a Group | 28-34 |
| Create Databases (in Hive) | 28-35 |
| Privileges on Source Data for Tables | 28-36 |
| Granting Privileges on Source Data for Tables | 28-38 |
| Creating the Table and Loading the Data | 28-39 |
| Attempting to Query the Table Without Privileges | 28-40 |
| Grant and Revoke Access to Table | 28-41 |
| Sentry Key Configuration Tasks | 28-42 |
| Oracle Database Access to HDFS | 28-43 |
| Oracle Connection to Hadoop | 28-44 |
| Virtual Private Database Policies Restrict Data Access | 28-45 |
| Oracle Data Redaction Protects Sensitive Data | 28-46 |
| Auditing: Overview | 28-47 |
| Auditing | 28-48 |
| Oracle Audit Vault and Database Firewall | 28-49 |
| Oracle Audit Vault Reporting | 28-51 |
| User-Defined Alerts | 28-52 |
| Example: Authorized Data Access | 28-53 |
| Example: Unauthorized Data Access | 28-54 |
| Audit Vault Dashboard | 28-55 |
| Interactive Reporting | 28-56 |
| Cloudera Navigator | 28-57 |
| Cloudera Navigator Reporting | 28-58 |
| Cloudera Navigator Lineage Analysis | 28-59 |
| Encryption | 28-60 |

Network Encryption 28-61
Data at Rest Encryption 28-62
Summary 28-63

A Glossary

B Resources

15

Apache Spark

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Course Road Map

Module 1: Big Data Management System

Module 2: Data Acquisition and Storage

Module 3: Data Access and Processing

Module 4: Data Unification and Analysis

Module 5: Using and Managing Oracle Big Data Appliance

Lesson 9: Introduction to MapReduce

Lesson 10: Resource Management Using YARN

Lesson 11: Overview of Apache Hive and Apache Pig

Lesson 12: Overview of Cloudera Impala

Lesson 13: Using Oracle XQuery for Hadoop

Lesson 14: Overview of Solr

Lesson 15: Apache Spark

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

This lesson provides the overview of another large-scale data processing engine named Apache Spark and its features.

Objectives

After completing this lesson, you should be able to:

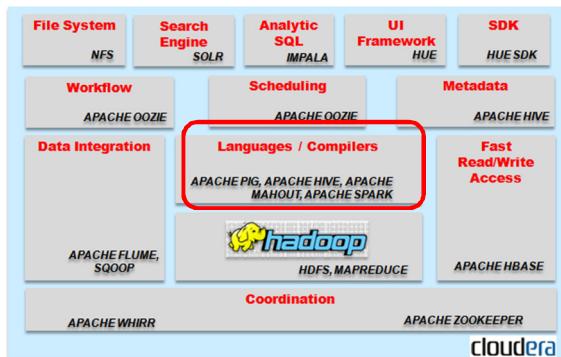
- Explain the overview of Spark
- Describe the Spark Architecture
- Explain the Resilient Distributed Dataset
- Explain the Directed Acyclic Graph



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Apache Spark

- An open source parallel data processing framework with a proven scalability up to 2000 nodes
- Complements Apache Hadoop
- Makes it easy to develop fast, unified Big Data applications combining batch, streaming, and interactive analytics on all your data



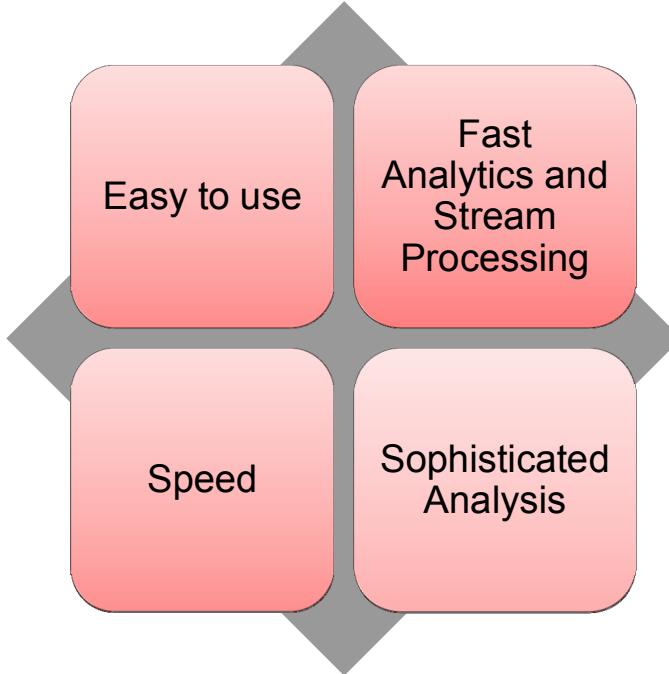
ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Apache Spark (incubating) is an open source, parallel data processing framework that complements Apache Hadoop to make it easy to develop fast, unified Big Data applications combining batch, streaming, and interactive analytics on all your data. It was originally developed in 2009 in UC Berkeley's AMPLab, and open sourced in 2010.

In subsequent years it has seen rapid adoption, used by enterprises small and large across a wide range of industries. It has quickly become one of the largest open source communities in big data, with over 200 contributors from 50+ organizations.

Introduction to Spark



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Speed

Spark is not tied to the two-stage MapReduce paradigm. Spark enables applications in Hadoop clusters to run up to 100x faster in memory, and 10x faster even when running on disk.

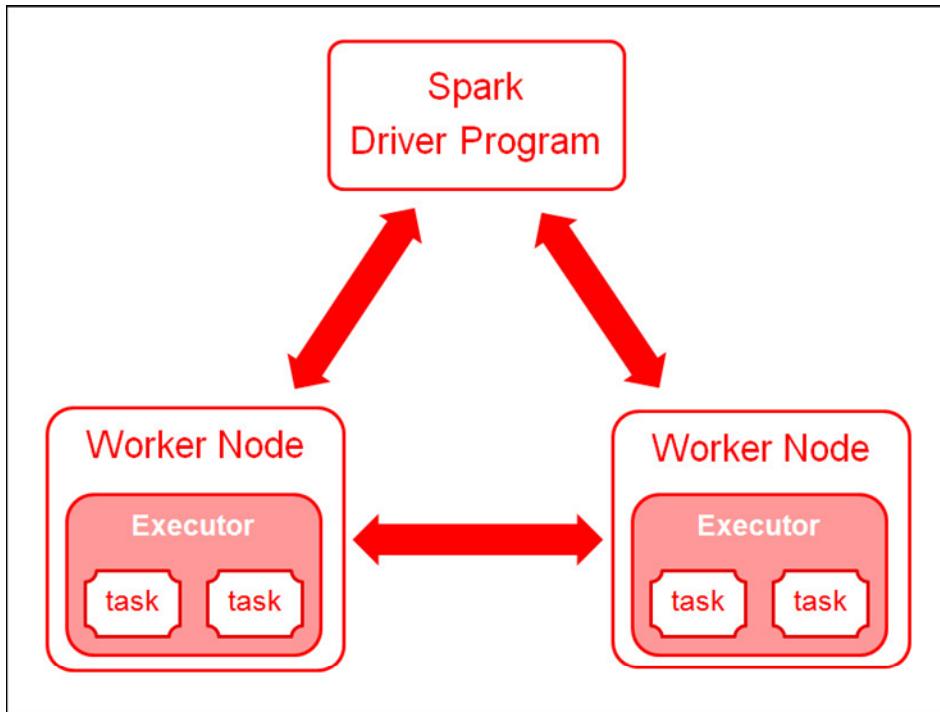
Ease of Use

Spark lets you quickly write applications in Java, Scala, or Python. It comes with a built-in set of over 80 high-level operators. You can use it interactively to query data within the shell.

Sophisticated Analytics

In addition to simple “map” and “reduce” operations, Spark supports SQL queries, streaming data, and complex analytics such as machine learning and graph algorithms out-of-the-box. Better yet, users can combine all these capabilities seamlessly in a single workflow.

Spark: Components for Distributed Execution



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Every Spark application consists of a driver program that launches various parallel operations on cluster. The driver program contains your application's main function and defines distributed data sets on the cluster and then applies operations to them. To run these operations, driver programs typically manage a number of nodes called executors.

From the architecture perspective, Apache Spark is based on two key concepts:

- Resilient Distributed Datasets (RDD)
- Directed acyclic graph (DAG) execution engine

Spark supports two types of RDDs: parallelized collections that are based on existing Scala collections and Hadoop data sets that are created from the files stored on HDFS.

Resilient Distributed Dataset (RDD)

- Resilient Distributed Dataset (RDD) is the basic level of abstraction in Spark.
- An RDD represents an immutable, partitioned collection of elements that can be operated on in parallel.
- This class contains the basic operations available on all RDDs, such as:
 - Map
 - Filter
 - Persist



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

In addition, PairRDD Functions contains operations available only on RDDs of key-value pairs, such as `groupByKey` and `join`.

RDD Operations

Transformations

- Map
- Filter
- Sample
- Union
- GroupByKey
- ReduceByKey
- Join & cache

Parallel Operations

- Reduce
- Collect
- Count
- Save
- lookupkey

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Transformations create a new data set from an existing one, and actions, which return a value to the driver program after running a computation on the data set. For example, map is a transformation that passes each data set element through a function and returns a new RDD representing the results. Alternatively, reduce is an action that aggregates all the elements of the RDD by using some function and returns the final result to the driver program (although there is also a parallel reduceByKey that returns a distributed data set).

Parallelized collections are created by calling SparkContext's parallelize method on an existing collection in your driver program (a Scala Seq). The elements of the collection are copied to form a distributed data set that can be operated on in parallel.

Characteristics of RDD

- 1 List of splits
- 2 Function for computing each split
- 3 List of dependencies on other RDDs
- 4 Partitioner for key-value RDDs
- 5 List of preferred locations



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Internally, each RDD is characterized by five main properties:

- A list of splits (partitions)
- A function for computing each split
- A list of dependencies on other RDDs
- Optionally, a Partitioner for key-value RDDs (for example, to say that the RDD is hash-partitioned)
- Optionally, a list of preferred locations to compute each split on (for example, block locations for an HDFS file)

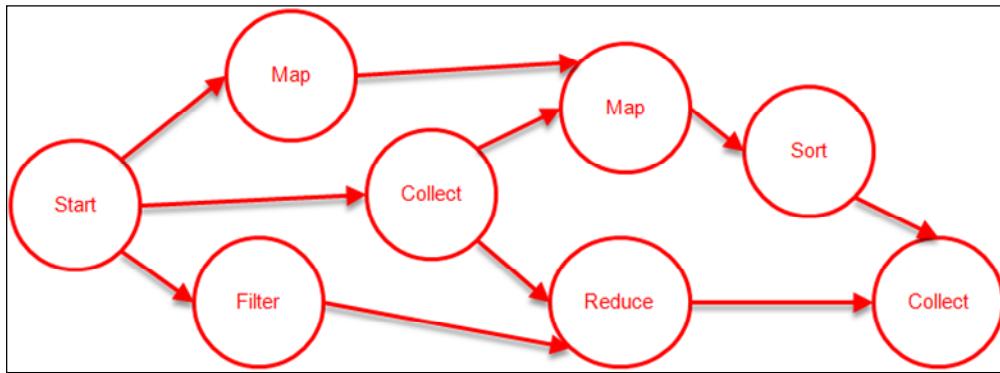
The DAG engine helps to eliminate the MapReduce multistage execution model and offers significant performance improvements.

Directed Acyclic Graph Execution Engine

Spark follows a Directed Acyclic Graph (DAG) execution. Some of the properties of DAG are:

- Directed – Propagates only in a single direction
- Acyclic – No looping or coming back

The following is a sample execution engine of DAG:



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Spark features an advanced DAG engine supporting cyclic data flow. Each Spark job creates a DAG of task stages to be performed on the cluster. Compared to MapReduce, which creates a DAG with two predefined stages—Map and Reduce, DAGs created by Spark can contain any number of stages. This allows some jobs to complete faster than they would in MapReduce, with simple jobs completing after just one stage, and more complex tasks completing in a single run of many stages, rather than having to be split into multiple jobs.

Scala Language: Overview

Scala programming language can be used for implementing Spark. Scala:

- The acronym for “Scalable Language”
- A pure-bred object-oriented language
- Runs on the JVM
- Reliable for large mission critical systems



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Scala’s approach is to develop a small set of core constructs that can be combined in flexible ways. It smoothly integrates object-oriented and functional programming. It is designed to express common programming patterns in a concise, elegant, and type-safe way.

For more information about Scala, visit: <http://www.scala-lang.org/>

Scala Program: Word Count Example

```
import org.apache.spark.SparkContext
import org.apache.spark.SparkContext._
import org.apache.spark.SparkConf
object SparkWordCount {
    def main(args: Array[String]) {
        val sc = new SparkContext(new SparkConf().setAppName("Spark
Count"))
        val threshold = args(1).toInt
        // split each document into words
        val tokenized = sc.textFile(args(0)).flatMap(_.split(" "))
        // count the occurrence of each word
        val wordCounts = tokenized.map((_, 1)).reduceByKey(_ + _)
        // filter out words with less than threshold occurrences
        val filtered = wordCounts.filter(_.value >= threshold)
        // count characters
        val charCounts = filtered.flatMap(_.value.toCharArray).map((_,
1)).reduceByKey(_ + _)
        System.out.println(charCounts.collect().mkString(", "))
    }
}
```



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The Scala program in the slide is a classic MapReduce example, which finds the word count. The sample program performs the following tasks:

- Reads an input set of text documents
- Counts the number of times each word appears
- Filters out all words that show up less than a million times
- For the remaining set, counts the number of times each letter occurs

In MapReduce, this would require two MapReduce jobs, as well as persisting the intermediate data to HDFS in between them. In contrast, in Spark, you can write a single job in about 90 percent fewer lines of code.

Spark Shells

There are two interactive Spark shells available to execute the Spark programs.

- `spark-shell` is used for Scala.
- `pyspark` is used for Python.



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Summary

In this lesson, you should have learned how to:

- Explain the overview of Spark
- Describe the Spark Architecture
- Explain the Resilient Distributed Dataset
- Explain the Directed Acyclic Graph



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Practice 15: Overview

This practice covers the following topics:

- Writing a WordCount . java program by using Scala, which is on a Hadoop Cluster
- Running the WordCount . java program and viewing the results
- Understanding the features of Spark web-based interface



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

16

Options for Integrating Your Big Data

ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Course Road Map

Module 1: Big Data Management System

Module 2: Data Acquisition and Storage

Module 3: Data Access and Processing

Module 4: Data Unification and Analysis

Module 5: Using and Managing Oracle Big Data Appliance

Lesson 16: Options for Integrating Your Big Data

Lesson 17: Overview of Apache Sqoop

Lesson 18: Using Oracle Loader for Hadoop (OLH)

Lesson 19: Using Copy to BDA

Lesson 20: Using Oracle SQL Connector for HDFS

Lesson 21: Using ODI and OGG with Hadoop

Lesson 22: Using Oracle Big Data SQL

Lesson 23: Using Advanced Analytics:
Oracle Data Mining and Oracle R Enterprise

Lesson 24: Introducing Oracle Big Data Discovery

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

This lesson begins a new module on Data Unification and Analysis. This module contains a large number of lessons, indicating the significance of this phase and the variety of technologies available for data unification and analysis.

This first lesson provides an introduction to the available data integration options, including both Hadoop projects and Oracle technologies. These integration options are covered in lessons 17 – 22. Some of the integration options also provide a degree of analytic capability.

The final two lessons in this module are devoted to purely analytic technologies.

Objectives

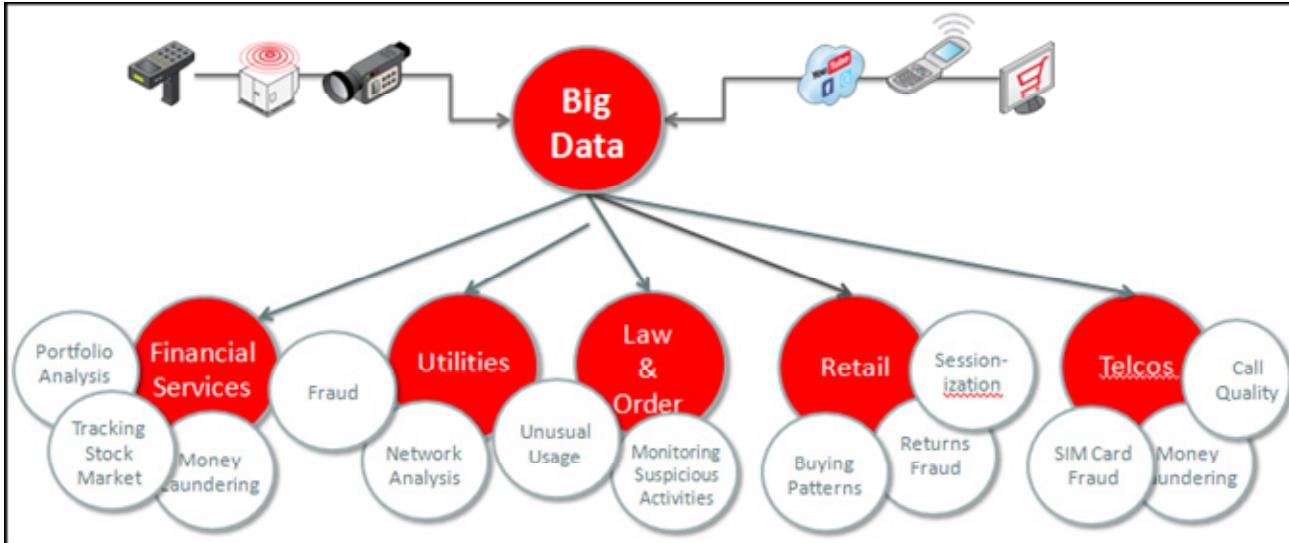
After completing this lesson, you should be able to:

- Identify the need to integrate your data
- Provide an overview of data unification technologies that are supported by the Oracle Big Data Management System



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Unifying Data: A Typical Requirement



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

There are so many use cases for big data—and they typically involve combining data in Hadoop with data that is stored in data warehouses or relational databases.

For example, consider an online retailer. All purchases are captured in their transaction systems. However, their weblogs capture the users' behavior.

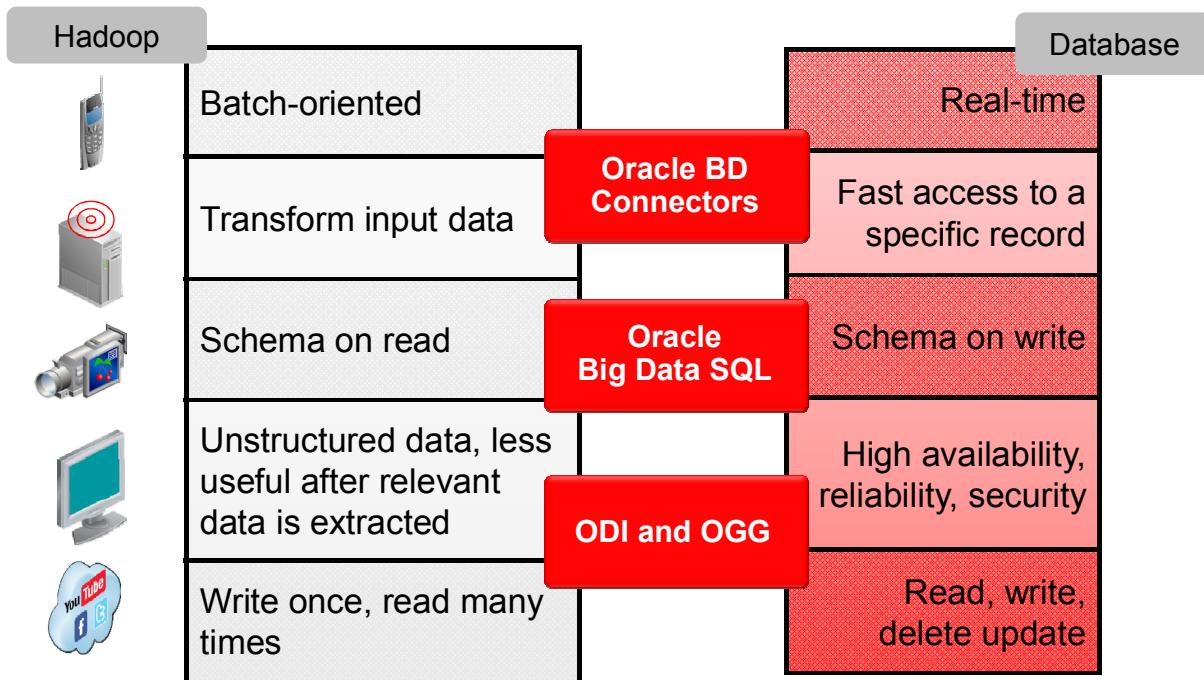
If we don't unify (integrate) the data in these separate platforms, can we answer important questions about our customers, such as these:

- What products and services are they interested in?
- What are they not looking at?
- How are they interacting with our site?
- Are they researching products and then buying elsewhere?

To answer these type of questions, you need to combine behavioral data (captured in weblogs) with actual sales and customer data (captured in the data warehouse).

In this module, you learn about a variety of options that enable you to combine and integrate data across your information management spectrum.

Integrating Data of Different Usage Patterns



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

This slide compares the general nature of the data usage between Hadoop and Oracle Database.

Given the significant usage pattern differences between the two platforms, and the need to organize both types of data in an integrated fashion, Oracle provides a range of options, including:

- Oracle Big Data Connectors
- Oracle Big Data SQL
- Oracle Data Integrator and Oracle GoldenGate

Introducing Data Unification Options

- Batch Loading
 - Apache Sqoop
 - Oracle Loader for Hadoop (Oracle Big Data Connector)
 - Copy to BDA (requires Exadata + BDA)
- Batch and Dynamic Loading
 - Oracle SQL Connector for Hadoop (Oracle Big Data Connector)
- Integration and Synchronization
 - Oracle Data Integrator for Hadoop
 - Oracle GoldenGate for Hadoop
- Dynamic Access
 - Oracle Big Data SQL (requires Exadata + BDA)



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The first part of this module – Data Unification and Analysis – focuses on the “data unification” technologies that are supported by the Oracle Big Data Management System. These are shown in the slide, and are covered in lessons 17 through 22. The second part of the module focuses on analytic technologies.

In this lesson, we introduce the data unification products and technologies that are shown here grouped by the way they function within the big data ecosystem. The categories include:

- Batch Loading
- Batch and Dynamic Loading
- Integration and Synchronization
- Dynamic Access

Note

- Two of these technologies, Copy to BDA and Big Data SQL, are only available with the combination of Oracle Exadata Database Machine and Oracle Big Data Appliance. They are high performance products that are optimized for engineered systems.
- Other products and technologies listed above are available both with engineered systems and for commodity hardware.

Data Unification: Batch Loading

- Batch Loading
 - Apache SQOOP
 - Oracle Loader for Hadoop (OLH)
 - Copy to BDA (Exadata + BDA)
- Batch and Dynamic Loading
 - Oracle SQL Connector for Hadoop
- Integration and Synchronization
 - Oracle Data Integrator for Hadoop
 - Oracle GoldenGate for Hadoop
- Dynamic Access
 - Oracle Big Data SQL



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

So first, we introduce three batch loading options for data unification:

- Apache SQOOP
- Oracle Loader for Hadoop (OLH), which is one of the Oracle Big Data Connectors
- Copy to BDA

Sqoop



Apache Sqoop is a tool that is designed for efficiently transferring bulk data between structured data stores and Apache Hadoop.

Scoop:

- Enables the data imports and exports between external data stores (including enterprise data warehouses) and Hadoop
- Parallelizes data transfer for fast performance and optimal system utilization
- Mitigates excessive loads to external systems



ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

A batch loading technique that is part of the Hadoop ecosystem is found in Apache Sqoop. Sqoop is a tool used for transferring data between an RDBMS and Hadoop. It internally generates MapReduce code to transfer data, and has a variety of benefits, as shown in the slide.

Sqoop was developed by Cloudera and then published to Apache as an incubating project.

Oracle Loader for Hadoop (OLH)

- Oracle Loader for Hadoop is an efficient and high-performance loader for fast movement of data from Hadoop into a table in Oracle Database.
- OLH tasks include:
 - Offloading expensive data processing from the database server to Hadoop
 - Working with a range of input data formats
 - Handling skew in input data to maximize performance
 - Loading using online and offline modes



ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Oracle Loader for Hadoop (OLH) enables users to use Hadoop MapReduce processing to generate optimized data sets for efficient loading and analysis in Oracle Database 12c. It also generates Oracle internal formats to load data faster and use fewer database system resources.

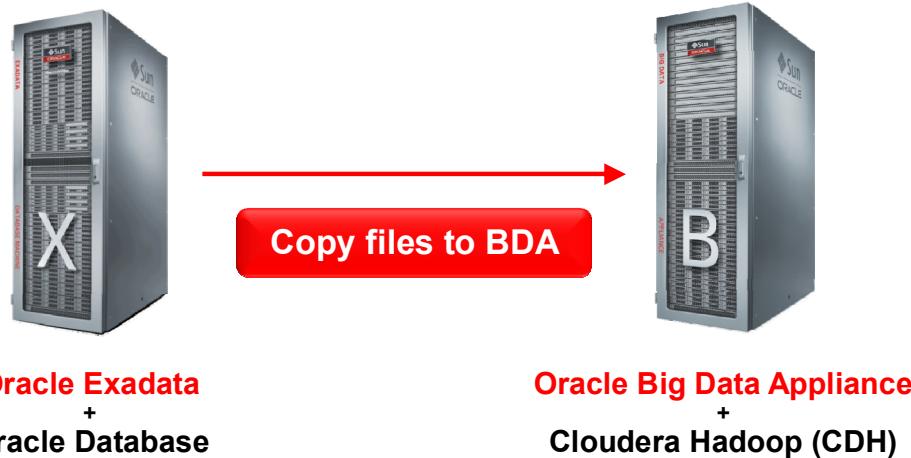
OLH is added as the last step in the MapReduce transformations as a separate map-partition-reduce step, which uses the CPUs in the Hadoop cluster to format the data into Oracle-understood formats. This results in a lower CPU load on the Oracle cluster as well as higher data load rates, because the data is already formatted for Oracle Database.

After the data is loaded, the content is permanently available in the database, providing very fast data access.

In the offline mode, Oracle Data Pump files are created on HDFS.

Copy to BDA

Copy data from Oracle Database on Exadata into CDH on BDA



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Copy to BDA enables you to copy tables from Oracle Database on the Oracle Exadata machine to Cloudera Distribution Hadoop (CDH) that is present in the Oracle Big Data Appliance.

After copying the files to the BDA, you can use Apache Hive to query the data. Hive can process the data locally without accessing Oracle Database.

As mentioned previously, Copy to BDA is optimized for the Exadata and BDA engineered systems.

Data Unification: Batch and Dynamic Loading

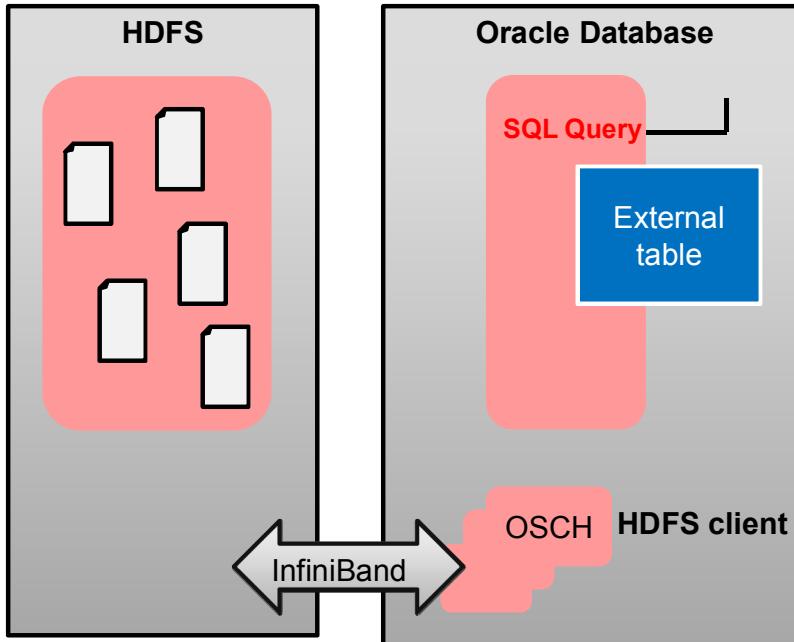
- Batch Loading
 - Apache SQOOP
 - Oracle Loader for Hadoop
 - Copy to BDA
- Batch and Dynamic
 - Oracle SQL Connector for Hadoop (OSCH)
- Integration and Synchronization
 - Oracle Data Integrator for Hadoop
 - Oracle GoldenGate for Hadoop
- Dynamic Access
 - Oracle Big Data SQL



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Next, we will introduce Oracle's batch and dynamic loading option for data unification: Oracle SQL Connector for Hadoop (OSCH). OSCH is also one of the Oracle Big Data Connectors.

Oracle SQL Connector for Hadoop



- ✓ Direct access from Oracle Database
- ✓ SQL access to Hive and HDFS
- ✓ Automated generation of external tables to access the data
- ✓ Access or load data in parallel

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

OSCH makes it possible to access data on the Hadoop cluster in HDFS and Hive from Oracle Database by using SQL. It provides a virtual table view of the HDFS files and enables parallel query access to data by using the standard Oracle Database external table mechanism.

If you need to import the HDFS data into Oracle Database, OSCH does not require a file copy or Linux Fuse. Instead, it uses the native Oracle Loader interface.

Data Unification: ETL and Synchronization

- Batch Loading
 - Apache Sqoop
 - Oracle Loader for Hadoop
 - Copy to BDA
- Batch and Dynamic
 - Oracle SQL Connector for Hadoop
- Integration and Synchronization
 - Oracle Data Integrator for Hadoop
 - Oracle GoldenGate for Hadoop
- Dynamic Access
 - Oracle Big Data SQL

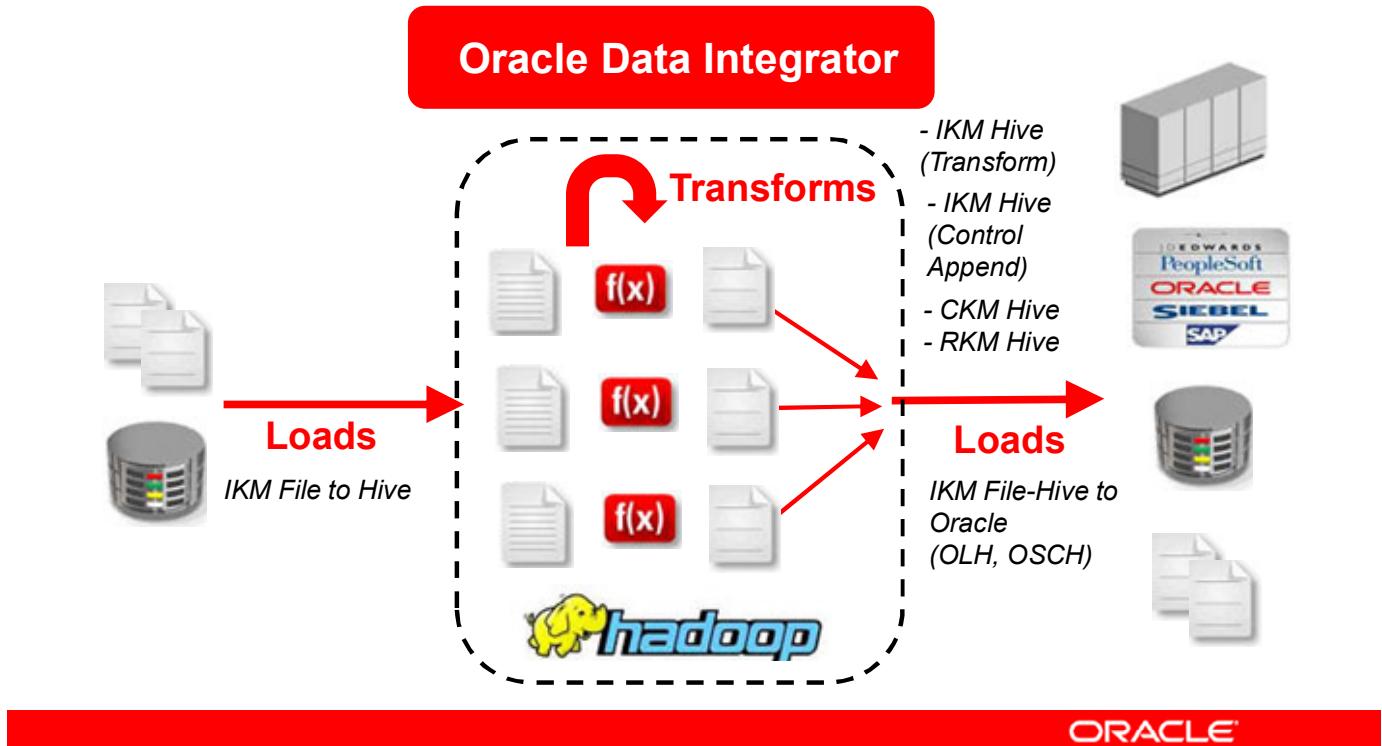


Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Now, we introduce Oracle Data Integrator and Oracle GoldenGate, which provide data integration and synchronization capabilities for the big data environment.

Oracle Data Integrator with Big Data

Heterogeneous Integration with Hadoop Environments



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Oracle Data Integrator (ODI) performs industry leading, batch process “E-LT” processing. This is a next-generation architecture from conventional ETL tools. ODI’s “E-LT” model provides greater flexibility and improved performance over the record-by-record transformation paradigm of the standard ETL process.

In addition, ODI handles both structured data sources and Hadoop/NoSQL data sources. For Big Data, ODI can take full advantage of the tools and execution engines within your Hadoop cluster:

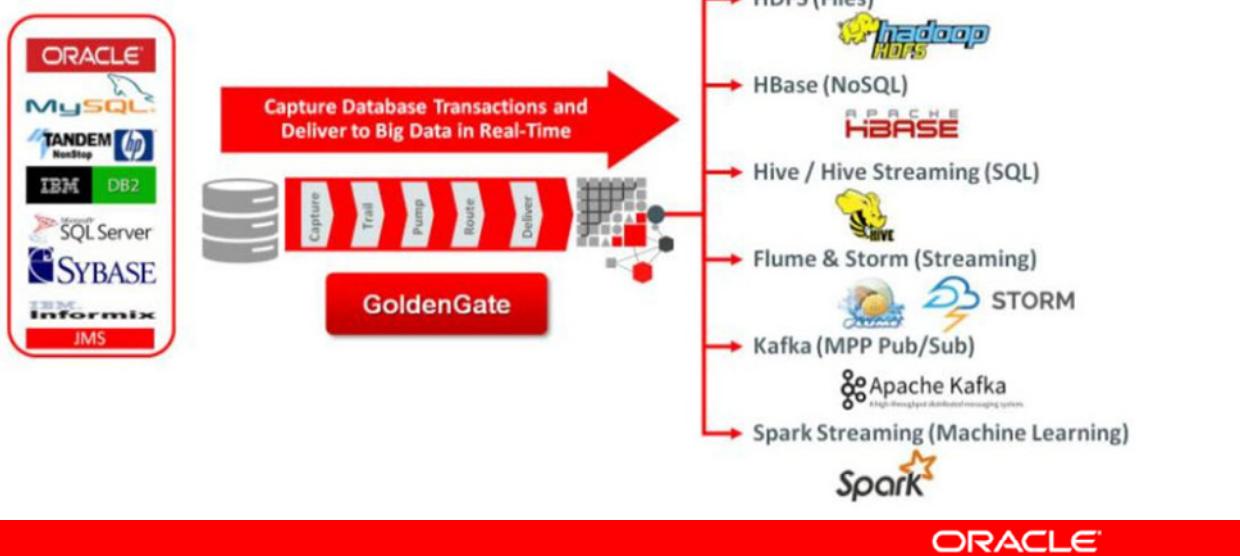
- Starting from loading data to the Hadoop cluster
- Then, transforming data natively within Hadoop, using MapReduce algorithms
- Finally, using optimized tools, such as Oracle Loader for Hadoop or the Oracle SQL Connector for Hadoop

You can mix and match these mechanisms with your existing load and transform mechanisms, for existing source and target data stores.

ODI does not force its own transformation engine on your process, everything is natively run in Hadoop MapReduce, and on your source and target databases.

Oracle GoldenGate for Big Data

- Performs real-time replication and synchronization of data
- Streamlines real-time data delivery into big data formats including Apache Hadoop, Apache HBase, Apache Hive, and Apache Flume, and others



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

ORACLE

Oracle GoldenGate is a time tested and proven product for real-time data replication and synchronization of tables or schemas. Using OGG, you can maintain two copies of the same data, with each copy being kept in sync with one another.

In a new offering, Oracle GoldenGate for Big Data, provides optimized and high performance delivery of this feature to a variety of big data formats, including:

- HDFS
- HBase
- Hive.
- Flume
- Oracle NoSQL
- Kafka
- Storm
- Spark

Data Unification: Dynamic Access

- Batch Loading
 - Apache SQOOP
 - Oracle Loader for Hadoop
 - Copy to BDA
- Batch and Dynamic
 - Oracle SQL Connector for Hadoop
- Integration and Synchronization
 - Oracle Data Integrator with Hadoop
 - Oracle GoldenGate for Hadoop
- Dynamic Access
 - Oracle Big Data SQL (Exadata + BDA)



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Finally, we introduce Oracle Big Data SQL, which enables simultaneous access to a wide variety of structured (RDBMS) and unstructured data for integrated data analysis.

Oracle Big Data SQL: A New Architecture

- Allows you to combine and analyze a variety of data formats without moving the data
- Provides powerful, high-performance SQL on Hadoop
- Enables simple data integration of Hadoop and Oracle Database
- Runs on optimized hardware



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Oracle Big Data SQL allows you to analyze data where it resides without requiring you to load the data into the database. It enables interaction with data on different tiers:

- Database in-memory—fastest—use for most frequently accessed data
- Query data in tables—also for frequently accessed data
- Query data stored in Hadoop through Oracle Database—using full Oracle SQL access

Oracle Big Data SQL, which supports external tables on Hadoop and NoSQL Database, provides simple, yet powerful data integration of Hadoop and Oracle Database, resulting in:

- A single SQL point-of-entry to access all data
- Scalable joins between Hadoop and RDBMS data

Big Data SQL is optimized to run on the Exadata and BDA engineered systems, which also incorporate:

- High-speed InfiniBand network between Hadoop and Oracle Database
- Smart Scan technology, which filters data as it streams from disk
- Storage Indexing, which ensures that only relevant data is read
- Caching, for frequently accessed data takes less time to read

When To Use Different Oracle Technologies?

| | OLH | Copy to BDA | OSCH | ODI & OGG | Big Data SQL |
|--|-----|-------------|------|-----------|--------------|
| Oracle SQL Access to data in HDFS | N | N | Y | N | Y |
| Offload processing to Hadoop | Y | Y | N | Y | Y |
| Supports Complex Source Data | Y | Y | N | Y | Y |
| Requires BDA and Exadata | N | Y | N | N | Y |

Certification Matrix:

<http://www.oracle.com/us/products/database/big-data-connectors/certifications/index.html>



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The table provides a usage matrix for deciding when to use an Oracle data unification option.

For batch processing:

- OLH is an excellent solution for loading Oracle data to Hadoop without requiring engineered systems.
- Copy to BDA, which requires BDA and Exadata, provides a more powerful and efficient batch processing capability than OLH.

For access to Hadoop data from Oracle database, OSCH provides SQL access capability. However, it does not offload any processing to Hadoop.

ODI and OGG provide industry-leading, powerful, and integrated data loading and synchronization capabilities for Oracle database and Hadoop, without requiring engineered systems. However, ODI and OGG will leverage the high-end capabilities of engineered systems.

If you use the engineered system combination of BDA and Exadata, Big Data SQL is the most flexible and powerful option. It provides the fastest, most robust way of integrating and accessing data in Hadoop, NoSQL and Oracle database, without having to move the data.

Note: For a certification matrix of the Big Data Connectors, see:

<http://www.oracle.com/us/products/database/big-data-connectors/certifications/index.html>

Summary

In this lesson, you should have learned about:

- The need to integrate your data
- Data unification technologies supported by the Oracle Big Data Management System



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

17

Overview of Apache Sqoop

ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Course Road Map

Module 1: Big Data Management System

Module 2: Data Acquisition and Storage

Module 3: Data Access and Processing

Module 4: Data Unification and Analysis

Module 5: Using and Managing Oracle Big Data Appliance

Lesson 16: Options for Integrating Your Big Data

Lesson 17: Overview of Apache Sqoop

Lesson 18: Using Oracle Loader for Hadoop (OLH)

Lesson 19: Using Copy to BDA

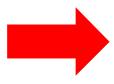
Lesson 20: Using Oracle SQL Connector for HDFS

Lesson 21: Using ODI and OGG with Hadoop

Lesson 22: Using Oracle Big Data SQL

Lesson 23: Using Advanced Analytics:
Oracle Data Mining and Oracle R Enterprise

Lesson 24: Introducing Oracle Big Data Discovery



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

This lesson shows how to transfer data between HDFS and Oracle Database by using Apache Sqoop.

Objectives

After completing this lesson, you should be able to:

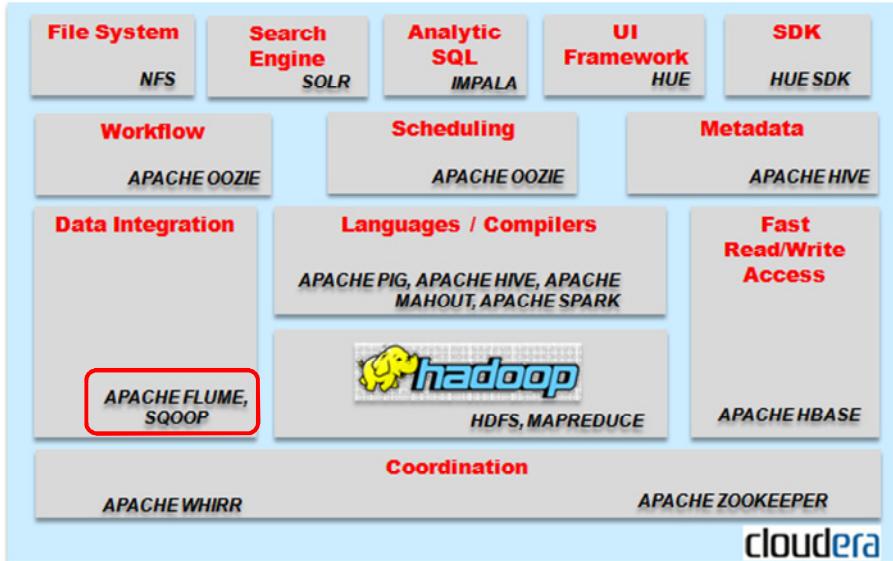
- Describe Soop
- Describe Soop features
- Describe Soop Connectors
- Describe Oracle Data Integrator integration with Soop



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Apache Sqoop

Apache Sqoop is a tool for transferring data between Apache Hadoop and relational databases.

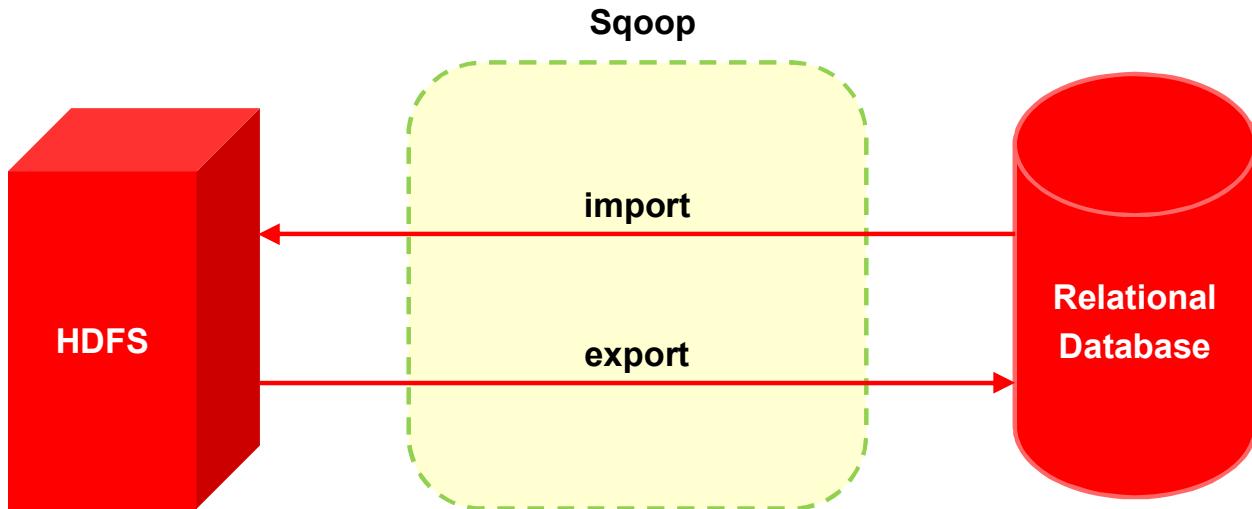


ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Apache Sqoop is a command-line tool to transfer data between Apache Hadoop and relational databases. It is an import and export utility.

Sqoop Components



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

You can use Sqoop to transfer data between HDFS and RDBMS.

Sqoop Import:

Sqoop can be used for importing data from a relational database into HDFS. The input to the import process is a database table. Sqoop reads the table row-by-row into HDFS. The output of this import process is a set of files containing a copy of the imported table. The import process is performed in parallel. For this reason, the output will be in multiple files. These files may be delimited text files (for example, with commas or tabs separating each field), or binary Avro or SequenceFiles containing serialized record data.

Sqoop Export:

Sqoop can be used for exporting data from a HDFS into relational database. Sqoop reads the set of delimited text files from HDFS in parallel, parses them into records, and inserts them as new rows in a target database table.

Sqoop Features

Apache Sqoop:

Enables data movement from a variety of relational databases to Hadoop (referred to as importing data to Hadoop)

Enables data movement from Hadoop to a variety of relational databases (referred to as exporting data from Hadoop)

Parallelizes data transfer for fast performance and optimal system utilization

Enables use of Hadoop to analyze data in a variety of relational systems

Sqoop uses MapReduce to import and export data, providing parallelism and fault tolerance



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

You can use Sqoop to import data from many relational systems and enterprise data warehouses such as MySQL, SQL Server, Netezza, and so on, and export data from Hadoop to these systems.

It can also be used to import and export data between Hadoop and Oracle Database. Note that Oracle Loader for Hadoop is a better tool for moving data to Oracle Database (exporting from Hadoop), and when using engineered systems, Copy to BDA (feature of Oracle Big Data SQL) is a better tool for moving data from Oracle Database to Hadoop (importing to Hadoop). Oracle Loader for Hadoop and Copy to BDA are designed, developed, and supported by Oracle. Oracle Loader for Hadoop works with databases and Hadoop on engineered systems and on commodity hardware, while Copy to BDA is only licensed with engineered systems.

The big advantage of Sqoop is to move data between Hadoop and a variety of other databases or data warehouses such as MySQL, SQL Server, DB2, Netezza and so on. It is relevant for use with Oracle Database when it is not possible to use Oracle Loader for Hadoop and Copy to BDA.

Sqoop: Connectors

Apache Sqoop has connectors for a variety of systems:

Oracle Connector

MySQL Connector

Netezza Connector

Teradata Connector

SQL Server Connector



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Sqoop has customized connectors for a variety of relational databases or data warehouses. They are available for download from either the Hadoop vendor, Apache, and/or the vendor of the external database.

Importing Data into Hive

Sqoop supports importing data into Hive.

- Sqoop imports data into Hive, if you have a Hive metastore associated with your HDFS cluster.
- You need to add the `--hive-import` option to your Sqoop command for importing data into Hive.

The following code snippet must be used for importing data into hive from a MySQL table:

```
sqoop import  
--connect jdbc:mysql://<ip address>\<database name>  
--username <username_for_mysql_user>  
--password <Password>  
--table <mysql_table name>  
--hive-import  
-m 1
```



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The main function of Sqoop's import tool is to upload your data into files in HDFS. If you have a Hive metastore associated with your HDFS cluster, Sqoop can also import the data into Hive by generating and executing a `CREATE TABLE` statement to define the data's layout in Hive. Importing data into Hive is as simple as adding the `--hive-import` option to your Sqoop command line.

If the Hive table already exists, you can specify the `--hive-overwrite` option to indicate that existing table in Hive must be replaced. After your data is imported into HDFS or this step is omitted, Sqoop generates a Hive script containing a `CREATE TABLE` operation defining your columns by using Hive's types, and a `LOAD DATA INPATH` statement to move the data files into Hive's warehouse directory.

Sqoop: Advantages

- Fast data transfer
- Allows easy integration of existing databases with Hadoop-based systems
- Integrated with Hadoop UI Tools and with Oracle Data Integrator



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Summary

In this lesson, you should have learned to:

- Describe Soop
- Describe Soop features
- Describe Soop Connectors
- Describe Oracle Data Integrator integration with Soop



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Using Oracle Loader for Hadoop (OLH)

18



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Course Road Map

Module 1: Big Data Management System

Module 2: Data Acquisition and Storage

Module 3: Data Access and Processing

Module 4: Data Unification and Analysis

Module 5: Using and Managing Oracle Big Data Appliance

Lesson 16: Options for Integrating Your Big Data

Lesson 17: Overview of Apache Sqoop

Lesson 18: Using Oracle Loader for Hadoop (OLH)

Lesson 19: Using Copy to BDA

Lesson 20: Using Oracle SQL Connector for HDFS

Lesson 21: Using ODI and OGG with Hadoop

Lesson 22: Using Oracle Big Data SQL

Lesson 23: Using Advanced Analytics:
Oracle Data Mining and Oracle R Enterprise

Lesson 24: Introducing Oracle Big Data Discovery



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

This lesson introduces you to Oracle Loader for Hadoop (OLH) and its features.

Objectives

After completing this lesson, you should be able to:

- Define Oracle Loader for Hadoop
- List the Oracle Loader for Hadoop installation steps
- Describe the methods of loading data from Hadoop to Oracle Database



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Oracle Loader for Hadoop

Oracle Loader for Hadoop (OLH):

- Is an efficient and high-performance loader for fast movement of data from any Hadoop cluster based on Apache Hadoop into a table in Oracle Database
- Enables you to use Hadoop MapReduce processing to create optimized data sets for efficient loading and analysis in Oracle Database
- Partitions the data and transforms it into an Oracle-ready format on the Hadoop cluster
- Sorts records by primary key before loading the data
- Can also write output files in HDFS for load, later



ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

OLH is a Java MapReduce application that balances the data across reducers to help maximize performance. It works with a range of input data formats that present the data as records with fields. It can read from sources that have the data already in a record format (such as Avro files or Apache Hive tables), or it can split the lines of a text file into fields.

You run OLH by using the Hadoop command-line utility. In the command line, you provide configuration settings with the details of the job. You typically provide these settings in a job configuration file.

Refer to the following link for additional details:

http://docs.oracle.com/cd/E55905_01/doc.40/e55819/olh.htm

Software Prerequisites

- A target database system running Oracle Database 10.2.0.5 and greater or Oracle Database 11.2.0.3 and greater or Oracle Database 12c
- The following Hadoop versions are certified:
 - Apache Hadoop 2.x
 - Cloudera's Distribution including Hadoop (CDH)
 - Hortonworks Data Platform (HDP)



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Oracle Loader for Hadoop requires the following software:

- A target database system running one of the following:
 - Oracle Database 10.2.0.5 or greater
 - Oracle Database 11g release 2 (11.2.0.3) or greater
 - Oracle Database 12c
- Any of the Hadoop:
 - Apache Hadoop 2.x (This is certified by Oracle. Oracle provides support.)
 - Cloudera's Distribution including Hadoop (CDH) 4.x or 5.x (This is certified by Oracle. Oracle provides support.)
 - Hortonworks Data Platform (HDP) 1.3 or 2.1 (This is certified by Hortonworks. Oracle provides support only for Oracle Connectors. Hortonworks directly provides support for Hadoop.)

Check the following link for further information

<http://www.oracle.com/us/products/database/big-data-connectors/certifications/index.html>

Modes of Operation

Oracle Loader for Hadoop operates in two modes:

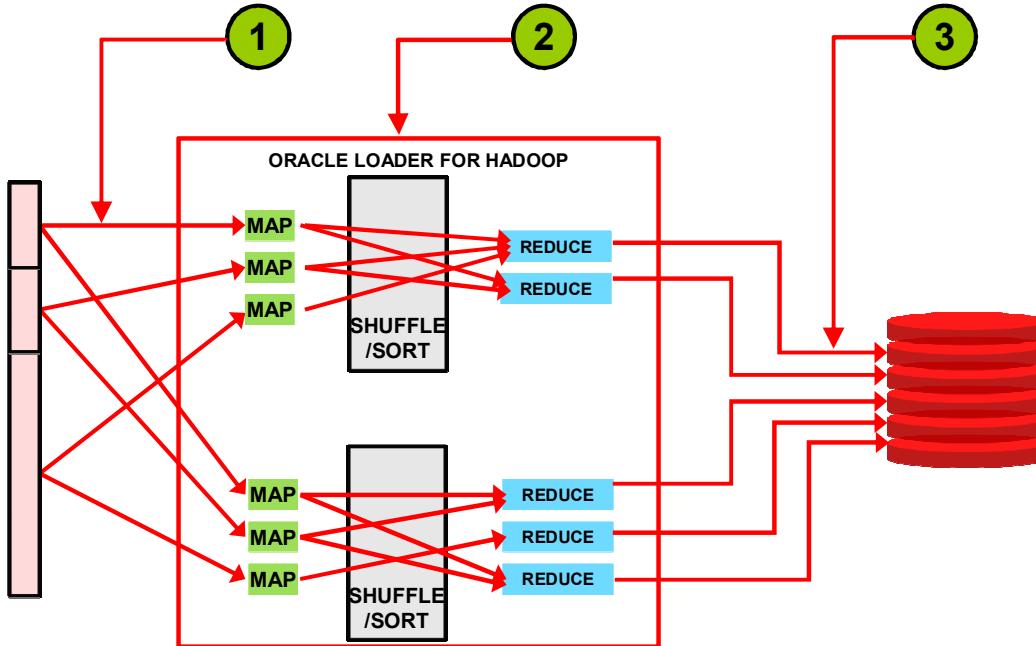
- Online Database Mode
- Offline Database Mode



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

These modes are explained in the next few slides.

OLH: Online Database Mode



ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

1. Read target table metadata from the database.
2. Perform partitioning, sorting, and data conversion.
3. Connect to the database from reducer nodes; load into database partitions in parallel.

In online database mode, data is loaded into the database by using either a JDBC output format or an OCI Direct Path output format. OCI Direct Path output format generates a high-performance direct path load of the target table. The JDBC output format performs a conventional path load.

Running an OLH Job

You can use the OraLoader utility in a Hadoop command to run a job by using OLH.

Syntax:

```
hadoop jar $OLH_HOME/jlib/oraloader.jar  
oracle.hadoop.loader.OraLoader \  
-conf job_config.xml \  
-libjars  
input_file_format1.jar[,input_file_format2.jar...]
```



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

To run an Oracle Loader for Hadoop job, you can include any generic hadoop command-line option.

- `-conf job_config.xml` identifies the job configuration file.
- `-libjars` identifies the JAR files for the input format.
- When using the Hive or Oracle NoSQL Database input formats, you must specify additional JAR files, as described later in this section.
- When using a custom input format, specify its JAR file. (Also remember to add it to `HADOOP_CLASSPATH`.)

Separate multiple file names with commas, and list each one explicitly. Wildcard characters and spaces are not allowed.

OLH Use Cases

Oracle Loader for Hadoop can be used for loading data from:

- HDFS files into Oracle database
- Hive tables into Oracle database

An XML file is used to specify the source of the data. Some of the important fields are:

- `mapreduce.inputformat.class`
- `mapreduce.outputformat.class`
- `oracle.hadoop.loader.loadermap.targetTable`



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The XML file contains the configuration parameters for the execution of OLH. You can review the file to see the parameters, including:

- `mapreduce.inputformat.class`: Specifies the input format of the input data file. For example, the input data is delimited text, so the value for this parameter will be `DelimitedTextInputFormat`.
- `mapreduce.outputformat.class`: Specifies the type of load. You specify here the value of `OCIOOutputFormat` to use the direct path online load option.
- `oracle.hadoop.loader.loaderMap.targetTable`: Specifies the name of the target table

Load Balancing in OLH

The connector generates a MapReduce partitioning scheme that assigns approximately the same amount of work to all reducers. Load balancing:

- Helps distribute the load evenly when data is not uniformly spread among the partitions
- Prevents slowdown of the load process because of unbalanced reducer loads



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Input Formats

The OLH can load with several input formats. OLH ships with the following input formats:

- Delimited Text Input Format
- Complex Text Input Formats
- Hive Table Input Format
- Avro Input Format



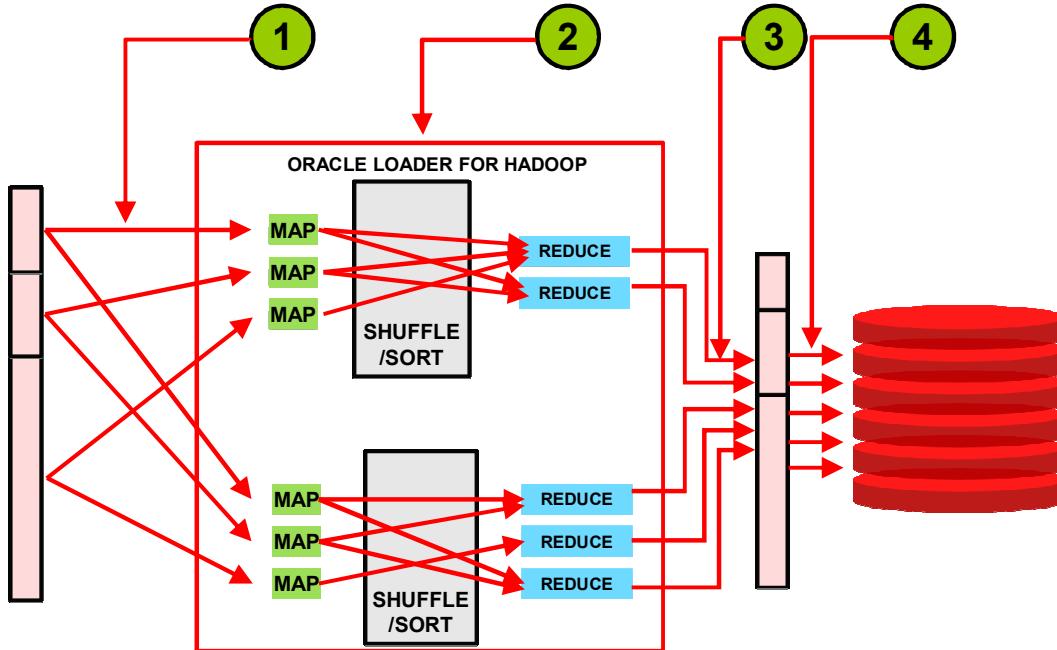
Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

An input format reads a specific type of data stored in Hadoop. Several input formats are available, which can read the data formats most commonly found in Hadoop:

- **Delimited Text Input Format:** To load data from a delimited text file
- **Complex Text Input Formats:** To load data from text files that are more complex than delimited text file. You can use regular expressions to determine which sections to load.
- **Hive Table Input Format:** To load data from a Hive table. You can load from all data sources accessible to Hive.
- **Avro Input Format:** To load data from binary Avro data files containing standard Avro-format records

Note: The input formats described in the slide are configurable. You can also write your own input format to handle other input sources.

OLH: Offline Database Mode



ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

1. Read target table metadata from the database.
2. Perform partitioning, sorting, and data conversion.
3. Write from reducer nodes to Oracle Data Pump files and copy files from HDFS to where database can access them.
4. Import into the database in parallel using external table mechanism.

In offline database mode, the reducer nodes create binary or text format output files.

Data Pump output format creates binary format files that are ready to be loaded into an Oracle database by using an external table and the ORACLE_DATAPUMP access driver.

Delimited Text output format creates text files in delimited format. These text files can be loaded into an Oracle database by using an external table and the ORACLE_LOADER access driver. The files can also be loaded by using the SQL*Loader utility.

Offline Load Advantages in OLH

OLH offline loads database server processing to Hadoop by:

- Converting the input data to final database format
- Computing the table partition for a row
- Sorting rows by primary key within a table partition
- Generating binary Data Pump files
- Balancing partition groups across reducers



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

OLH Versus Sqoop

- Sqoop is a generic loader, which can be used with the other third party database products.
- OLH is optimized for Oracle Database.
- OLH is extremely fast.
- OLH converts data into binary Oracle data types before loading. This minimizes the work of databases CPUs, so there is less impact on database applications.
- OLH automatically load balances when there is data skew, so all nodes do the same amount of work.



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

- OLH is optimized for Oracle Database.
- OLH automatically load balances when there is data skew, so all nodes are doing the same amount of work.
- OLH converts data into binary Oracle data types on Hadoop before loading. This minimizes the work database CPUs have to do, so there is less impact on database applications.
- OLH is an extremely fast loader.

<http://www.oracle.com/technetwork/database/database-technologies/bdc/hadoop-loader/connectors-hdfs-wp-1674035.pdf>

OLH: Performance

OLH is a high-performance loader that:

- Supports parallel load
- Balances load across reduce tasks
- Performs automatic mapping
- Uses a multithreaded sampler to improve performance of its MapReduce job
- Saves database CPU



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

OLH can automatically map the fields to the appropriate columns when the input data complies with the following requirements:

- All columns of the target table are loaded.
- The input data field names in the IndexedRecord input object exactly match the column names.
- All input fields that are mapped to DATE columns can be parsed by using the same Java date format.

In OLH, each sampler thread instantiates its own copy of the supplied InputFormat class. When implementing a new InputFormat, you must ensure that it is thread-safe.

Some of the benefits of OLH are:

- Shorten development time
- Minimize impact on database during load
- Easy to use
- Developed and supported by Oracle
- Works out-of-the-box with Kerberos authentication

Summary

In this lesson, you should have learned how to:

- Define Oracle Loader for Hadoop
- List the Oracle Loader for Hadoop installation steps
- Describe the methods of loading data from HDFS to Oracle Database



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Practice 18: Overview

In this practice, you will use Oracle Loader for Hadoop (OLH) to:

- Load data from HDFS files into Oracle database
- Load data from Hive tables into Oracle database



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

19

Using Copy to BDA

ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Course Road Map

Module 1: Big Data Management System

Module 2: Data Acquisition and Storage

Module 3: Data Access and Processing

Module 4: Data Unification and Analysis

Module 5: Using and Managing Oracle Big Data Appliance

Lesson 16: Options for Integrating Your Big Data

Lesson 17: Overview of Apache Sqoop

Lesson 18: Using Oracle Loader for Hadoop (OLH)

Lesson 19: Using Copy to BDA

Lesson 20: Using Oracle SQL Connector for HDFS

Lesson 21: Using ODI and OGG with Hadoop

Lesson 22: Using Oracle Big Data SQL

Lesson 23: Using Advanced Analytics:
Oracle Data Mining and Oracle R Enterprise

Lesson 24: Introducing Oracle Big Data Discovery

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

This lesson covers the last of the three batch processing data integration options in this module:
Copy to BDA.

Objectives

After completing this lesson, you should be able to:

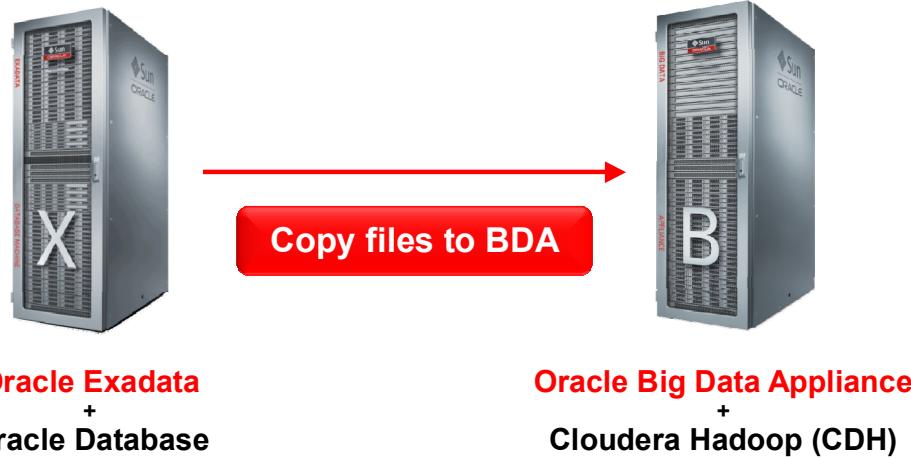
- Describe the purpose of Copy to BDA
- Use Copy to BDA



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Copy to BDA

Copy data from Oracle Database on Exadata to CDH on BDA.



ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Copy to BDA enables you to copy tables from Oracle Database on the Oracle Exadata machine to Cloudera Distribution Hadoop (CDH) that is present in the Oracle Big Data Appliance.

After copying the files to the BDA, you can use Apache Hive to query the data. Hive can process the data locally without accessing Oracle Database.

Requirements for Using Copy to BDA

License

- Copy to BDA is licensed under Oracle Big Data SQL.

Availability

- Only with Oracle Exadata Database Machine connected to Oracle Big Data Appliance



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

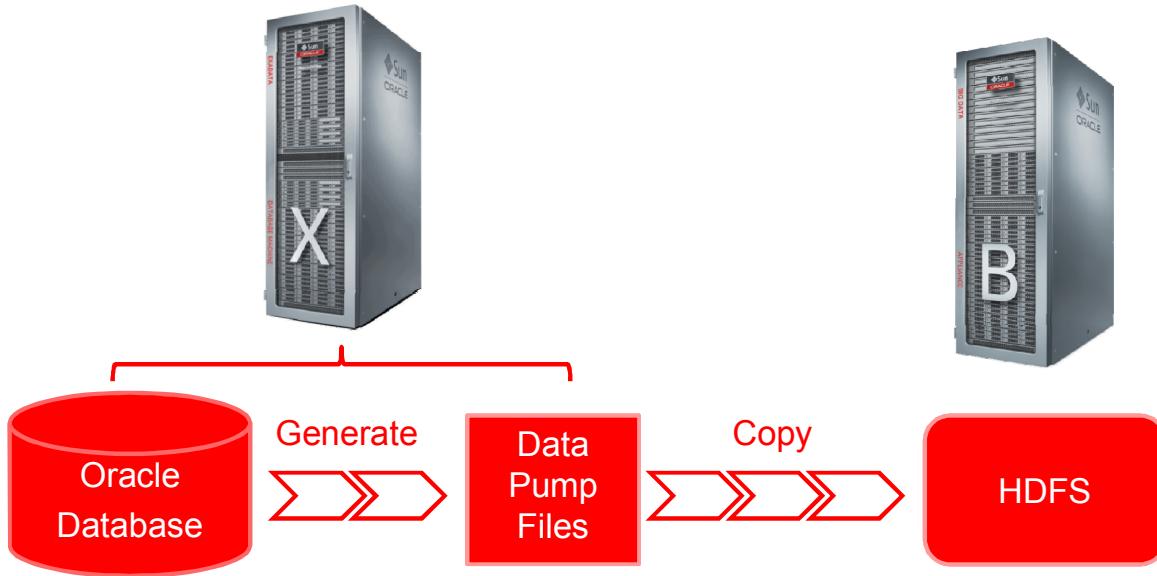
The following are the prerequisites for using Copy to BDA:

- Copy to BDA is a feature of the Oracle Big Data SQL option on the BDA. Therefore, you must have an Oracle Big Data SQL license on the BDA to use this feature.
- The feature is only available with Oracle Exadata Database Machine connected to the BDA.
- Oracle Exadata must be configured on the same InfiniBand or client network as the BDA.

Notes:

- Although not required, Oracle recommends an InfiniBand connection between Oracle Exadata Database Machine and Oracle Big Data Appliance, due to the performance benefits.
- The functionality of Big Data SQL is examined in detail in Lesson 21.

How Does Copy to BDA Work?



ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Conceptually, Copy to BDA comprises a two-stage process:

1. First, Data Pump files are generated from database tables on the Exadata machine.
2. Then, the data in the Data Pump files is copied to the HDFS on BDA.

Data Pump

Data Pump files are typically used to move data and metadata from one database to another. Copy to BDA uses this file format as an intermediary format to copy data from an Oracle database to HDFS.

To generate Data Pump format files, you create an external table from an existing Oracle table. An external table in Oracle Database is an object that identifies and describes the location of data from outside of a database. External tables use access drivers to parse and format the data.

For Copy to BDA, the `ORACLE_DATAPUMP` access driver is required to create the external table to perform the first part of the operation. The `ORACLE_DATAPUMP` access driver copies both the data and metadata from an internal Oracle table and populates the appropriate external table.

Copy to BDA: Functional Steps

1. Identify the Target Directory.
2. Create an Oracle external table.
3. Copy data to HDFS.
4. Create an external Hive table*.

* *The last step is required to provide Hive query access to the data.*



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Functionally, Copy to BDA includes four steps:

1. Identify the Target Directory on Exadata for an Oracle external table
2. Create the external table using Data Pump format
3. Copy data from the external table to the HDFS on BDA
4. Create an external Hive table over the Data Pump files on HDFS. This step enables query access to the data using Hive.

The next few slides show code examples for each task.

Step 1: Identify the Target Directory

```
SQL> CREATE DIRECTORY exportdir AS '/exportdir';
```



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

First, you must identify a target directory for the external table. This is where the Data Pump files will be stored.

Example

In the code example, a database DIRECTORY object named `exportdir` is created. This object points to the `/exportdir` directory on the Oracle Exadata Database Machine.

Note

- You must have read and write access to the target directory in Oracle Database.
- Only Oracle Database users with the `CREATE ANY DIRECTORY` system privilege can create directories.

Step 2: Create an External Table

```
CREATE TABLE export_customers
  ORGANIZATION EXTERNAL
  (
    TYPE oracle_datapump
    DEFAULT DIRECTORY exportdir
    LOCATION ('customers1.dmp', 'customers2.dmp')
  ) PARALLEL 2
AS SELECT * FROM customers;
```



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Second, create an external table by using the CREATE TABLE ... ORGANIZATION EXTERNAL statement.

Example

In the code example, an external table named `export_customers` is created (you can use any valid file system name), using the following clauses:

- **TYPE:** Specify `oracle_datapump` as the file type for the external table.
- **DEFAULT DIRECTORY:** Identifies the database directory that you created for this purpose
- **LOCATION:** Lists the name(s) of the Data Pump file(s) to be created. If more than one file is specified, separate them by commas.
- **PARALLEL:** This optional clause sets the degree of parallelism (DOP), in this case, 2. By default the DOP is 1, which is serial processing. A number larger than 1 enables parallel processing. Ideally, the number of file names should match the DOP. If no DOP is specified, the DOP is set to the number of files that are listed.
- **AS SELECT:** Use the full SELECT syntax for this clause. It is not restricted. In this example, the Oracle table to be copied is `customers`.

Result: The command generates two output files in parallel, named `customers1.dmp` and `customers2.dmp`.

Step 3: Copy Files to HDFS

```
$ cd /expdir  
$ ls *.dmp  
customers1.dmp  customers2.dmp  
$ hadoop fs -mkdir customers  
$ hadoop fs -put *.dmp /user/oracle/customers
```



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

When you license Oracle Big Data SQL on BDA, the Oracle Big Data SQL installation process installs Hadoop client files to Oracle Exadata Database Machine to facilitate Copy to BDA.

The Hadoop client installation enables you to use Hadoop commands to copy the Data Pump files to HDFS.

To copy the data pump files to HDFS, use the `hadoop fs -put` command on the BDA.

Example

In the example, the user:

- Changes to the `/expdir` directory
- Checks to ensure that the data pump files are in that directory by using the `ls` command
- Uses the `hadoop fs -mkdir` command to create an HDFS directory named `customers`
- Uses the `hadoop fs -put` command to copy the data pump files to the HDFS directory named `customers`, which is owned by the `oracle` user.

Note: The user must have write privileges on the HDFS directory to perform the last command.

Step 4: Create a Hive External Table

```
CREATE EXTERNAL TABLE customers
  ROW FORMAT SERDE 'oracle.hadoop.hive.datapump.DPSerde'
STORED AS
  INPUTFORMAT
    'oracle.hadoop.hive.datapump.DPInputFormat'
  OUTPUTFORMAT
    'org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutput
      Format'
LOCATION '/user/oracle/customers';
```



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

For external tables, Hive loads the table metadata into its own metastore, but the data remains in its original target location in the Data Pump files on the HDFS. Therefore, to provide query access to the data, you create a Hive external table over the Data Pump files.

Hive enables the definition of tables over HDFS directories by using syntax very similar to simple SQL.

Example

The preceding example uses the basic syntax of a Hive `CREATE TABLE` statement for creating a Hive external table named `customers` for use with a Data Pump format file.

This HiveQL statement creates an external table by using the Copy to BDA SerDes. The `LOCATION` clause identifies the full path to the Hadoop directory containing the Data Pump files.

Note

- Copy to BDA provides SerDes that enable Hive to deserialize the data and read the files.
- The Oracle SerDes are read only, so you cannot use them to write to the files.
- The Hive table columns automatically have the same names as the Oracle columns, which are provided by the metadata stored in the Data Pump files.
- If you drop an external Hive table (using a HiveQL `DROP TABLE` statement), then only the metadata is discarded, while the external data remains unchanged.

Oracle to Hive Data Type Conversions

| Oracle Data Type | Hive Data Type |
|-------------------|--|
| NUMBER | INT when the scale is 0 and the precision is less than 10 |
| | BIGINT when the scale is 0 and the precision is less than 19 |
| | DECIMAL when the scale is greater than 0 or the precision is greater than 19 |
| BINARY_DOUBLE | DOUBLE |
| BINARY_FLOAT | FLOAT |
| CHAR | CHAR |
| NCHAR | |
| VARCHAR2 | VARCHAR |
| NVARCHAR2 | |
| DATE | TIMESTAMP |
| TIMESTAMP | |
| TIMESTAMPTZFoot 1 | Unsupported |
| TIMESTAMPLTZ | |
| RAW | BINARY |



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The table in the slide contains the mappings between Oracle data types and Hive data types. This is used when the table is created in Hive by using the external table produced by Oracle Database.

Querying the Data in Hive

```
SELECT first_name, last_name, gender, birth, country
  FROM customers
 WHERE birth > 1985
 ORDER BY last_name LIMIT 5;

.

.

.

Opal      Aaron      M      1990      United States of America
KaKit     Abeles      M      1986      United States of America
Mitchel   Alambarati M      1987      Canada
Jade      Anderson   M      1986      United States of America
Roderica Austin      M      1986      United States of America
```



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Use HiveQL to query the Oracle data that has been copied to HDFS.

Example

As shown in the slide the HiveQL `SELECT` statement returns the same data as an equivalent SQL `SELECT` statement from Oracle Database.

Note

- If you drop an external table by using a HiveQL `DROP TABLE` statement, then only the metadata is discarded, while the external data that you copied to HDFS remains unchanged.
- External tables support data sources that are shared by multiple Hive tables.
- You use Oracle Database to update the data and generate new Data Pump files. You can then overwrite the old HDFS files with the updated files while leaving the Hive metadata intact. The same Hive table can be used to query the new data.

Summary

In this lesson, you should have learned how to:

- Describe the purpose of Copy to BDA
- Use the Copy to BDA code



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Practice 19: Overview

In this practice, you use Copy to BDA to copy files from Oracle Database to Hadoop, and then query the data in Hadoop using Hive.



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Using Oracle SQL Connector for HDFS

20

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Course Road Map

Module 1: Big Data Management System

Module 2: Data Acquisition and Storage

Module 3: Data Access and Processing

Module 4: Data Unification and Analysis

Module 5: Using and Managing Oracle Big Data Appliance

Lesson 16: Options for Integrating Your Big Data

Lesson 17: Overview of Apache Sqoop

Lesson 18: Using Oracle Loader for Hadoop (OLH)

Lesson 19: Using Copy to BDA

Lesson 20: Using Oracle SQL Connector for HDFS

Lesson 21: Using ODI and OGG with Hadoop

Lesson 22: Using Oracle Big Data SQL

Lesson 23: Using Advanced Analytics:
Oracle Data Mining and Oracle R Enterprise

Lesson 24: Introducing Oracle Big Data Discovery

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

This lesson provides the overview, operation, and the benefits of Oracle SQL Connector for HDFS (OSCH).

Objectives

After completing this lesson, you should be able to:

- Define Oracle SQL Connector for HDFS (OSCH)
- Describe the OSCH installation steps and software prerequisites
- Describe the operation and benefits of OSCH



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Oracle SQL Connector for HDFS

Oracle SQL Connector for Hadoop Distributed File System (OSCH) is a connector that enables read access to HDFS from Oracle Database by using external tables. It enables you to access:

- Data in-place in Hadoop from the database (without loading the data)
- Text data stored in HDFS files
- Data from Hive tables over text data
- Data pump files generated by Oracle Loader for Hadoop



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Refer to the following link for further details:

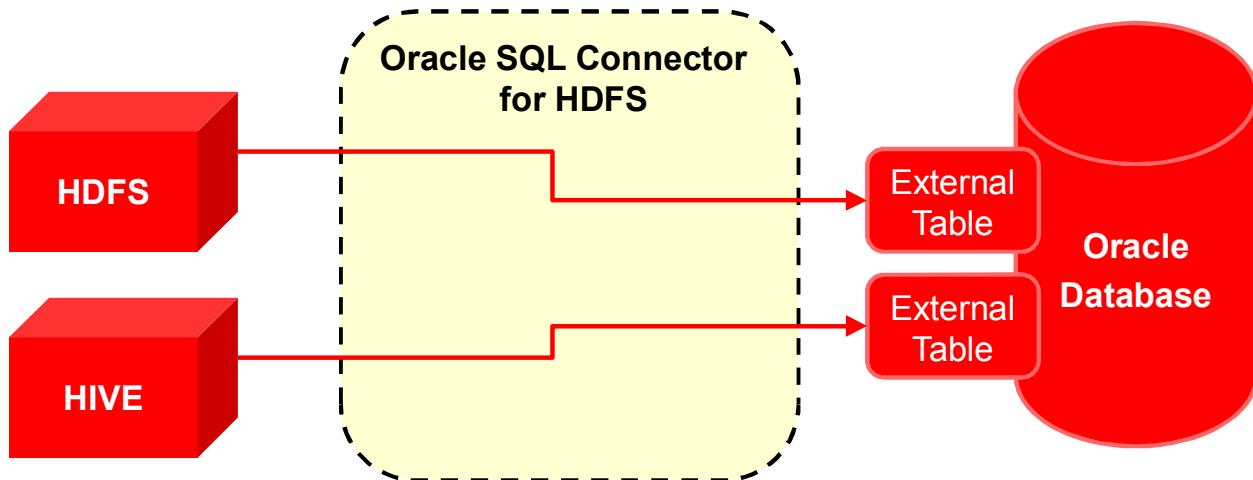
http://docs.oracle.com/cd/E55905_01/d

Check the following link for further information:

<http://www.oracle.com/us/products/database/big-data-connectors/certifications/index.html>

OSCH Architecture

Oracle SQL Connector for Hadoop Distributed File System (OSCH) provides read access to HDFS from an Oracle database.



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Oracle SQL Connector for Hadoop Distributed File System (Oracle SQL Connector for HDFS) provides read access to HDFS from an Oracle database by using external tables.

An external table is an Oracle Database object that identifies the location of data outside of the database. Oracle Database accesses the data by using the metadata provided when the external table was created. By querying the external tables, users can access data stored in HDFS as if that data were stored in tables in the database.

Using OSCH: Two Simple Steps

1. Run the OSCH utility to create external table and publish HDFS content to the external table.
2. Access and load into the database by using SQL.

```
>hadoop jar \
$OSCH_HOME/jlib/orahdfs.jar \
oracle.hadoop.hdfs.extab.ExternalTable \
-conf MyConf.xml \
-createTable
```



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

OSCH operates in two steps:

- Create an external table, and publish location of the HDFS content to the external table.
- Query the external table.

At this time, the Oracle SQL Connector preprocessor uses the information in the location files to locate the data in HDFS and stream it to the database.

You can also do joins of a table on HDFS with data in the database.

Using OSCH: Creating External Directory

Create an external table directory to store external table location files pointing to HDFS content.

```
CREATE DIRECTORY external_table_dir AS '...';
```



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Set Up Database Directory for OSCH External Table

- A directory has to be created before the user is given permission.

```
- CREATE or REPLACE DIRECTORY <external_table_dir> as '/.....';
```

Using OSCH: Database Objects and Grants

Create the user with appropriate privileges

- CONNECT / AS sysdba;
- CREATE USER hdfsuser IDENTIFIED BY password
 DEFAULT TABLESPACE hdfsdata
 QUOTA UNLIMITED ON hdfsdata;
- GRANT CREATE SESSION, CREATE TABLE, CREATE VIEW TO hdfsuser;
- GRANT EXECUTE ON sys.utl_file TO hdfsuser;
- GRANT READ, EXECUTE ON DIRECTORY osch_bin_path TO hdfsuser;
- GRANT READ, WRITE ON DIRECTORY external_table_dir TO hdfsuser;



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

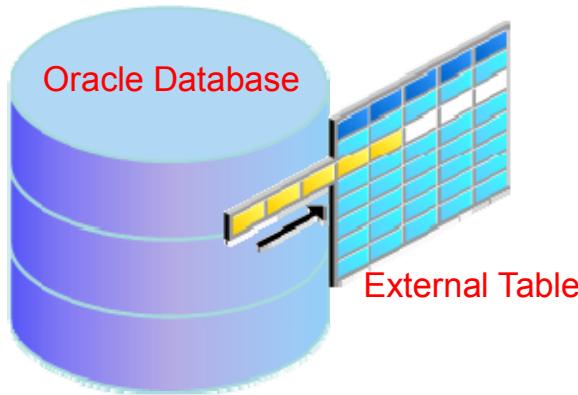
Oracle Database users require the following privileges to use OSCH:

- CREATE SESSION
- CREATE TABLE
- EXECUTE on the UTL_FILE PL/SQL package
- READ and EXECUTE on the OSCH_BIN_PATH directory created during OSCH installation

Do not grant WRITE access to anyone. Grant EXECUTE only to those who intend to use OSCH.

Using OSCH: Supported Data Formats

- Delimited text files in HDFS
- Data Pump files in HDFS
- Delimited text files in Apache Hive tables



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

OSCH can create external tables for the following data sources:

- **Data Pump files in HDFS:** OSCH queries Data Pump file headers for the information that it needs to create external table definitions for Data Pump–format files that were produced by Oracle Loader for Hadoop.
- **Hive tables:** OSCH queries the Hive metastore for the information that it needs to create external table definitions for Hive tables.
- **Delimited text files in HDFS:** OSCH uses configuration properties to retrieve the information that it needs to create external table definitions for text files stored in HDFS. All columns in the external tables are created as type VARCHAR2. All columns in the external tables are created as type VARCHAR2 by default. Data types can be specified in the configuration properties.

Using OSCH: HDFS Text File Support

OSCH facilitates Oracle database to access HDFS text files via external tables. The steps are as follows:

1. OSCH creates external table definition for the HDFS text file.
2. The external table contains the details of the HDFS text file, location, and connection information.
3. Oracle Database accesses the above-created external table.
4. The following code creates a new external table from Hive by using OSCH.

```
hadoop jar $OSCH_HOME/jlib/orahdfs.jar \
oracle.hadoop.exttab.ExternalTable \
-conf /home/oracle/movie/moviework/osch/moviefact_text.xml \
-createTable
```



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

OSCH facilitates Oracle Database to access HDFS text files via external tables. The external table definition is generated automatically from the HDFS text file definition. The data can be queried with Oracle SQL and joined with other tables in the database.

To create an external table for this purpose, you use the ExternalTable command-line tool provided with Oracle SQL Connector for HDFS. You provide ExternalTable with information about the data source in Hadoop and about your schema in an Oracle Database. You provide this information either as parameters to the ExternalTable command or in an XML file.

Oracle SQL Connector for HDFS is run with the `-createTable` option. This option generates the external table definition, creates the external table, and populates the location files in the `LOCATION` clause of the external table. After this step the external table is available for query.

Sample external table creation code.

```
CREATE TABLE "MOVIEDEMO"."MOVIE_FACT_EXT_TAB_FILE"
(
    "CUST_ID"                      VARCHAR2(4000),
    "MOVIE_ID"                      VARCHAR2(4000),
    "GENRE_ID"                      VARCHAR2(4000),
    "TIME_ID"                       VARCHAR2(4000),
    "RECOMMENDED"                  VARCHAR2(4000),
    "ACTIVITY_ID"                  VARCHAR2(4000),
    "RATING"                        VARCHAR2(4000),
    "SALES"                         VARCHAR2(4000)
)
ORGANIZATION EXTERNAL
(
    TYPE ORACLE_LOADER
    DEFAULT DIRECTORY "MOVIEWORKSHOP_DIR"
    ACCESS PARAMETERS
    (
        RECORDS DELIMITED BY 0X'0A'
        CHARACTERSET AL32UTF8
        PREPROCESSOR "OSCH_BIN_PATH":'hdfs_stream'
        FIELDS TERMINATED BY 0X'09'
        MISSING FIELD VALUES ARE NULL
        (
            "CUST_ID" CHAR(4000),
            "MOVIE_ID" CHAR(4000),
            "GENRE_ID" CHAR(4000),
            "TIME_ID" CHAR(4000),
            "RECOMMENDED" CHAR(4000),
            "ACTIVITY_ID" CHAR(4000),
            "RATING" CHAR(4000),
            "SALES" CHAR(4000)
        )
    )
    LOCATION
    (
        'osch-20150323070506-1677-1',
        'osch-20150323070506-1677-2'
    )
) PARALLEL REJECT LIMIT UNLIMITED;
```

Using OSCH: Hive Table Support

OSCH enables querying of Hive tables from Oracle database via external tables. The steps are as follows:

1. Use the utility available with OSCH to create an external table. This uses Hive metadata to get column information.
2. The location files of this external table contains the location of the Hive table data files and other details.
3. Once created, the external table can be queried by Oracle Database.

The following code creates a new external table from Hive by using OSCH:

```
hadoop jar $OSCH_HOME/jlib/orahdfs.jar \
oracle.hadoop.exttab.ExternalTable \
-conf /home/oracle/movie/moviework/osch/moviefact_hive.xml \
-createTable
```

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

OSCH facilitates Oracle Database to access Hive tables via external tables. The external table definition is generated automatically from the Hive table definition. Hive table data can be accessed by querying this external table. The data can be queried with Oracle SQL and joined with other tables in the database.

Sample external table creation code.

```
CREATE TABLE "MOVIEDEMO"."MOVIE_FACT_EXT_TAB_HIVE"
(
    "CUSTID"                      INTEGER,
    "MOVIEID"                      INTEGER,
    "GENREID"                      INTEGER,
    "TIME"                          VARCHAR2(4000),
    "RECOMMENDED"                  INTEGER,
    "ACTIVITY"                     INTEGER,
    "RATING"                       INTEGER,
    "SALES"                        NUMBER
)
ORGANIZATION EXTERNAL
(
    TYPE ORACLE_LOADER
    DEFAULT DIRECTORY "MOVIEWORKSHOP_DIR"
    ACCESS PARAMETERS
    (
        RECORDS DELIMITED BY 0X'0A'
        CHARACTERSET AL32UTF8
        PREPROCESSOR "OSCH_BIN_PATH":'hdfs_stream'
        FIELDS TERMINATED BY 0X'01'
        MISSING FIELD VALUES ARE NULL
        (
            "CUSTID" CHAR NULLIF "CUSTID"=0X'5C4E',
            "MOVIEID" CHAR NULLIF "MOVIEID"=0X'5C4E',
            "GENREID" CHAR NULLIF "GENREID"=0X'5C4E',
            "TIME" CHAR(4000) NULLIF "TIME"=0X'5C4E',
            "RECOMMENDED" CHAR NULLIF "RECOMMENDED"=0X'5C4E',
            "ACTIVITY" CHAR NULLIF "ACTIVITY"=0X'5C4E',
            "RATING" CHAR NULLIF "RATING"=0X'5C4E',
            "SALES" CHAR NULLIF "SALES"=0X'5C4E'
        )
    )
    LOCATION
    (
        'osch-20150323070852-2774-1'
    )
) PARALLEL REJECT LIMIT UNLIMITED;
```

Using OSCH: Partitioned Hive Table Support

OSCH enables querying of partitioned Hive tables from Oracle database via external tables. Querying is similar to Hive table querying. Additionally, the following components are created.

1. One external table per partition
2. A view over each external table
3. A metadata table that contains information on the external table, corresponding view, and the partition table column for that table and view



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Applications that query data in partitioned Hive tables should query the views instead of the external tables. This is because in the Hive data file storage structure, the partition column value is not part of the data. Instead, the partition column value is embedded in the directory name of the data file directory structure. When creating the views, Oracle SQL Connector for HDFS adds an additional column that contains the partition column value.

OSCH: Features

- Access and analyze data in place on HDFS via external tables.
- Query and join data on HDFS with database-resident data.
- Load into the database by using SQL (if required).
- Automatic load balancing: Data files are grouped to distribute load evenly across parallel threads.
- DML operations and indexes cannot be created on external tables.
- Data files can be text files or Oracle Data Pump files.
- Parallelism is controlled by the external table definition.



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The data on HDFS is accessed by using the external table mechanism. As a result, all the advantages and limitations of external tables are applicable. Data files can be either text files or Data Pump files created by OLH.

Parallelism is controlled by the external table definition, where PQ slaves in the database read data in parallel. For example, if you have 64 PQ slaves, 64 files are read in parallel. The number of PQ slaves is limited by the number of location files defined by the external table. HDFS data files are grouped in the location files to distribute the load evenly across the PQ slaves.

Parallelism and Performance

Performance

Performance is almost entirely determined by:

- Degree of parallelism

Parallelism

Parallelism is determined by:

- Number of data files
- Number of location files
- Degree of Parallelism

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Parallelism is controlled by the external table definition, where PQ slaves in the database read data in parallel. For example, if you have 64 PQ slaves, 64 files are read in parallel. The number of PQ slaves is limited by the number of location files defined by the external table. HDFS data files are grouped in the location files to distribute the load evenly across the PQ slaves.

- A good heuristic is

(# of data files in Hive partition)

should be equal to or a multiple of (value in oracle.hadoop.extbl.locationFileCount)

which should be equal to or a multiple of (DOP in the database)

For maximum parallelism make the

(# of data files in Hive partition) = (value in oracle.hadoop.extbl.locationFileCount) = (DOP in the database)

OSCH: Performance Tuning

Parallelism significantly improves performance of OSCH. There are three factors that determine parallelism:

1. The number of data files (when reading Hive tables or text data)
2. The number of location files (default value is 4)
3. In the database session:
 - Degree of parallelism (DOP)
 - Query hints



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Increasing the number of data files, location files, and DOP increases parallelism, but the relationship between the three is also important.

OSCH: Key Benefits

OSCH:

- Uniquely enables access to HDFS data files from Oracle Database
- Is easy to use for Oracle DBAs and Hadoop developers
- Is developed and supported by Oracle
- Has extremely fast load performance
- Performs load in parallel
- Works out-of-the-box with Kerberos authentication



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The main difference between OLH and OSCH is that OLH performs the MapReduce process and then loads the data into the database, whereas OSCH simply loads the data into partitions.

OSCH creates external tables for retrieving the content in HDFS.

You can also query the data in HDFS files from the Oracle Database, and import the data into the database whenever you want to.

Loading: Choosing a Connector

| | Oracle Loader for Hadoop | Oracle SQL Connector for HDFS |
|-------------------------|--|---|
| Use Case | Continuous or frequent load into production database | Bulk load of large volumes of data |
| Functionality | Load | Load and query in place |
| Usability | Likely to be preferred by Hadoop developers | Likely to be preferred by Oracle developers |
| Input Data Types | Load various types of input data: HBase, JSON files, Weblogs, sequence files, custom formats, etc. | Load text (HDFS files and Hive table files) Load Oracle Data Pump files that are generated by Oracle Loader for Hadoop from HBase, JSON files, Weblogs, sequence files, custom formats, and so on. |
| Performance | Spends time on Hadoop for preprocessing data | Faster load, trade-off is more, database CPU resources are used. |



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Summary

In this lesson, you should have learned how to:

- Define Oracle SQL Connector for HDFS (OSCH)
- Describe the OSCH installation steps and software prerequisites
- Describe the operation and benefits of OSCH



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Practice 20: Overview

In this practice, you will use Oracle SQL Connector for Hadoop Distributed File System (HDFS) to:

- Access data from HDFS files
- Access data from Hive tables
- Access data from partitioned Hive tables



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Using Oracle Data Integrator and Oracle GoldenGate with Hadoop

21

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Course Road Map

Module 1: Big Data Management System

Module 2: Data Acquisition and Storage

Module 3: Data Access and Processing

Module 4: Data Unification and Analysis

Module 5: Using and Managing Oracle Big Data Appliance

Lesson 16: Options for Integrating Your Big Data

Lesson 17: Overview of Apache Sqoop

Lesson 18: Using Oracle Loader for Hadoop (OLH)

Lesson 19: Using Copy to BDA

Lesson 20: Using Oracle SQL Connector for HDFS

Lesson 21: Using ODI and OGG with Hadoop

Lesson 22: Using Oracle Big Data SQL

Lesson 23: Using Advanced Analytics:
Oracle Data Mining and Oracle R Enterprise

Lesson 24: Introducing Oracle Big Data Discovery



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

This lesson provides a high-level overview of how Oracle Data Integrator and Oracle GoldenGate provide integration and synchronization capabilities for data unification of relational and hadoop data.

Objectives

After completing this lesson, you should be able to describe how:

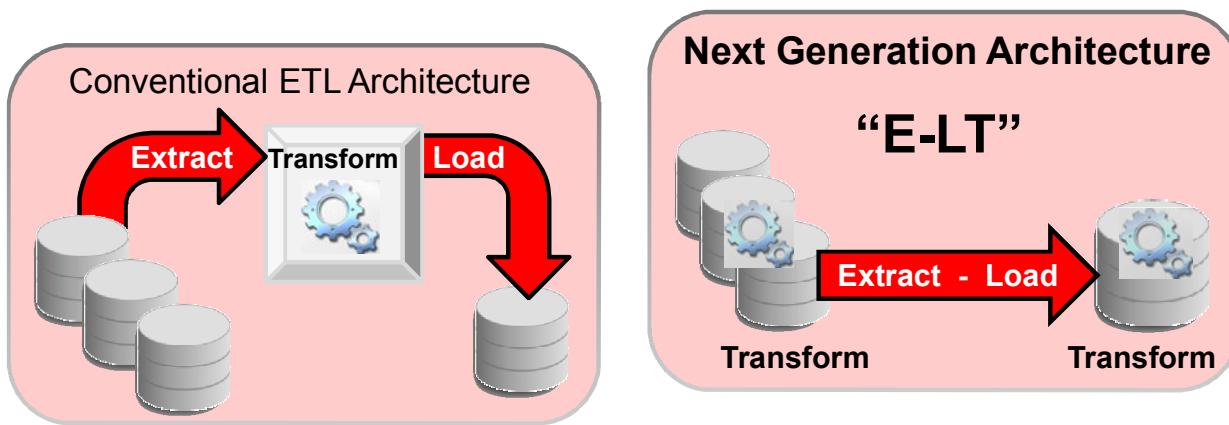
- Oracle Data Integrator interacts with Hadoop
- Oracle GoldenGate for Big Data interacts with Hadoop



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Oracle Data Integrator

- Performs industry-leading, batch process “E-LT,” including a new declarative design for integration processes
- Unified tool for both structured data and Hadoop/NoSQL
- Leverages optimized, set-based transformations
- Takes advantage of the existing hardware and software



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Oracle Data Integrator 12c incorporates a next-generation architecture from conventional ETL tools. ODI uses an “E-LT” model, which provides greater flexibility and improved performance over the record-by-record transformation paradigm of the standard ETL process.

For all supported data types, Oracle Data Integrator (ODI) performs the following:

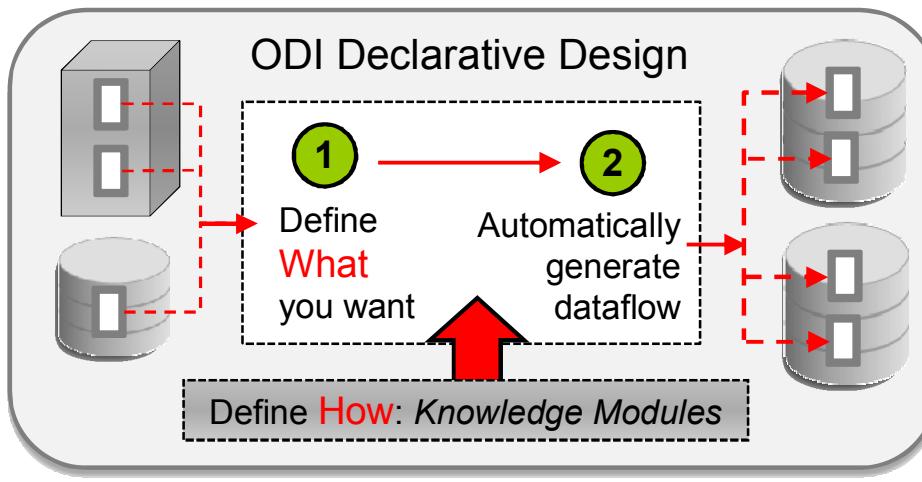
- **E-L:** Extracts data straight from the source and loads directly into the target, using optimal native loading techniques
- **T:** The transform step can be configured to happen on the source and/or the target. For example, ODI can:
 - Leverage source data transformation features, such as executing massively parallel algorithms in the Hadoop MapReduce engine to discard extraneous data.
 - Perform target table transformations where the power of the database server is harnessed. All database transformations leverage optimized set-based operations that do not have to be performed row-by-row.

In addition, ODI incorporates:

- A new declarative design approach to defining data transformation and integration processes, which supports faster and simpler development and maintenance
- A unified infrastructure to streamline data and application integration projects

ODI's Declarative Design

- Universal design model for simple to complex mappings
- Robust and reusable
- Quick to define and refactor maps
- Extensible model for any data integration mechanism



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

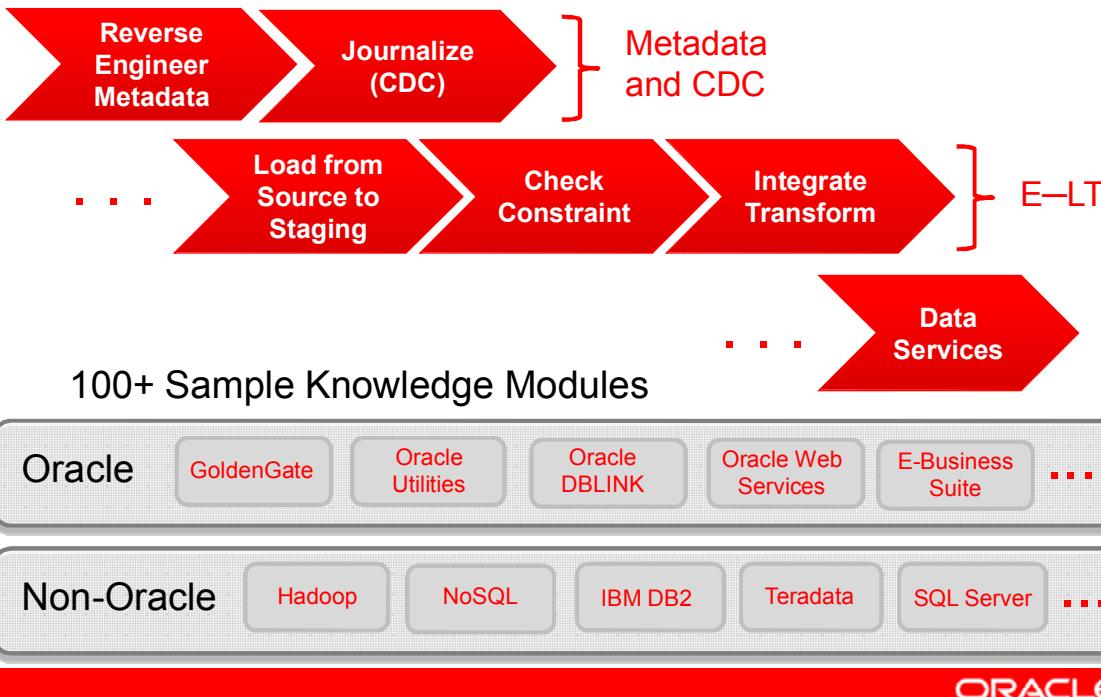
ODI's new Declarative design model enables the following:

- Easy and universal process for designing mappings, from simple to very complex. The developer just maps data from the sources to the target, and the system chooses the correct detail steps based on built-in templates called *Knowledge Modules*.
- The flow logic can be generated and reviewed. You don't have to code it yourself. This reduces the learning curve for developers and reduces implementation times.
- Robust and reusable mappings. This feature enables you to create mappings just once for any physical design.
- You can quickly define and also refactor maps, with a high degree of automation, which simplifies maintenance.

ODI Knowledge Modules (KMs)

Simpler Physical Design / Shorter Implementation Time

Pluggable Architecture:



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

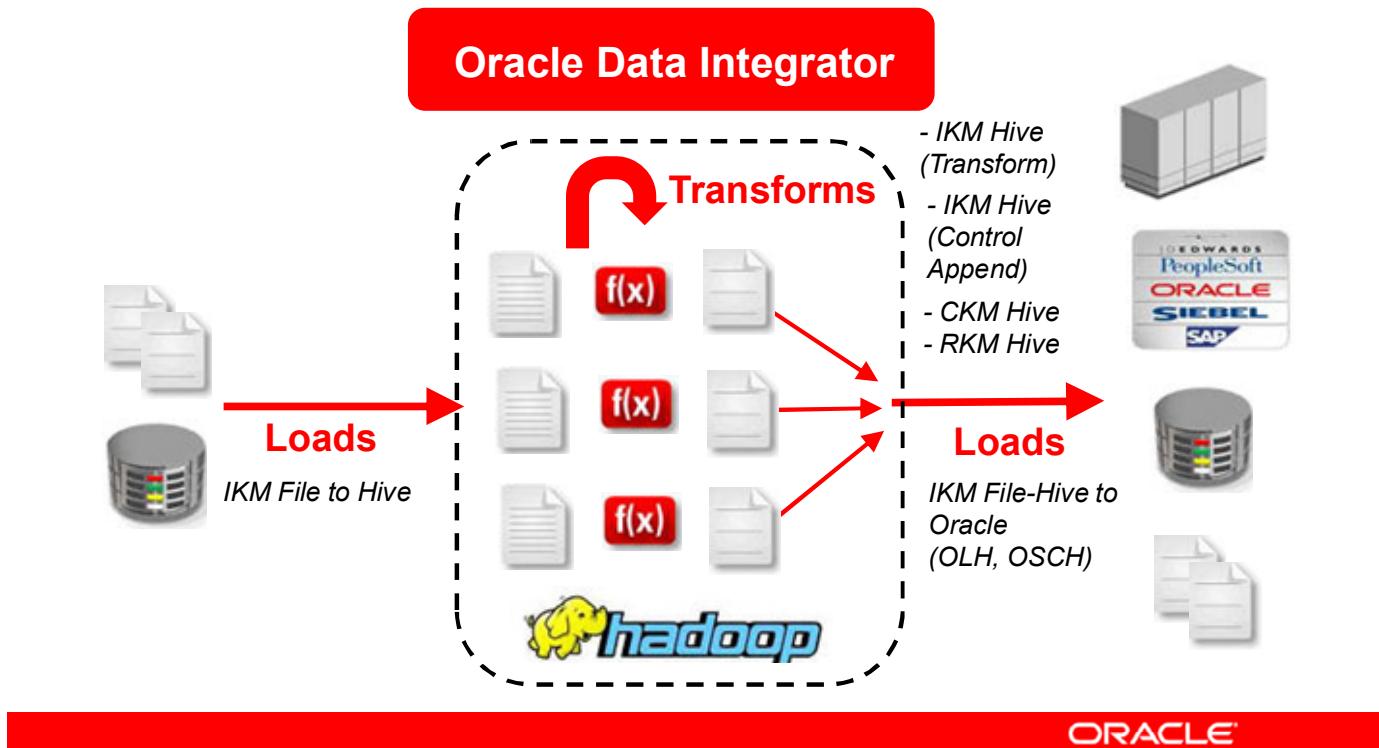
Knowledge modules are pluggable components that are at the core of the ODI architecture. ODI provides a comprehensive library of these knowledge modules, which encapsulate the templates that are used to generate the execution logic. There are six module types, which can be grouped into three categories:

- Metadata and CDC
 - Reverse modules which take metadata out of the source and target technologies to be used in the ODI metadata
 - Journalizing modules which take change data from sources, enabling ODI to apply those changes to the target.
- E-LT
 - Load modules use optimal native mechanisms to take bulk data from sources into a staging area.
 - Check modules apply business rules to the data that is in the staging area.
 - Integration modules perform transformations on data as it is loaded into the target.
- Finally, the Services modules expose the final data as web services.

ODI provides over 100 sample Out-of-the-Box knowledge modules, both for Oracle technologies and for non-Oracle technologies, including Hadoop and NoSQL.

Using ODI with Big Data

Heterogeneous Integration with Hadoop Environments



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

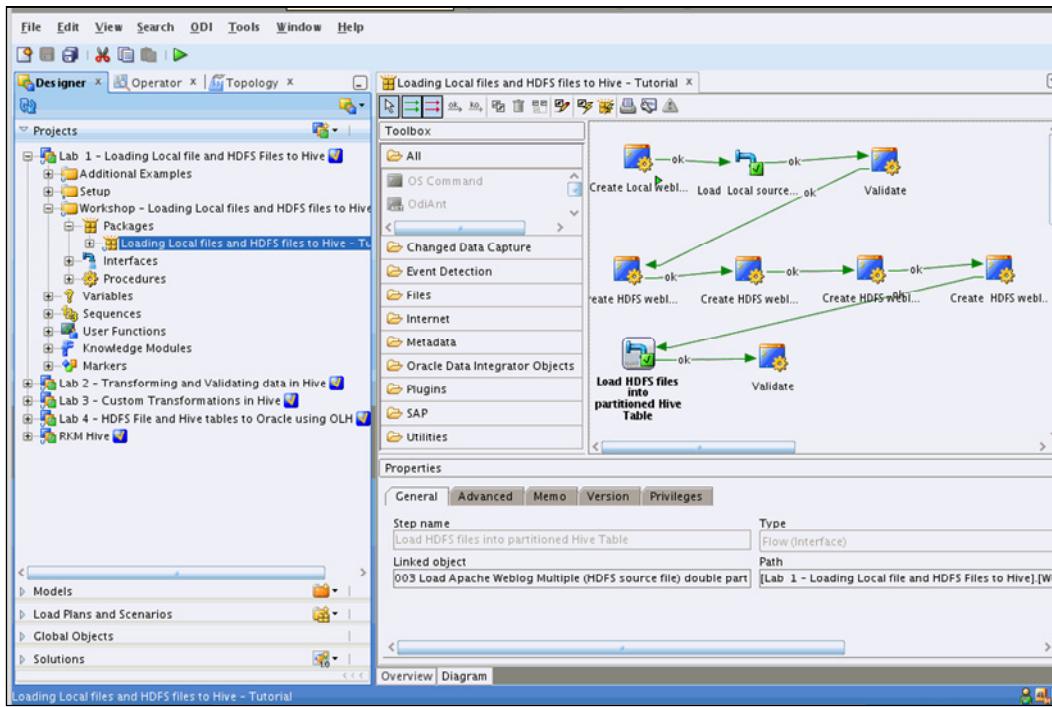
For Big Data, ODI can take full advantage of the tools and execution engines within your Hadoop cluster:

- Starting from loading data to the Hadoop cluster
- Then, transforming data natively within Hadoop, using MapReduce algorithms
- Finally, using optimized tools, such as Oracle Loader for Hadoop or the Oracle SQL Connector for Hadoop

You can mix and match these mechanisms with your existing load and transform mechanisms, for existing source and target data stores.

ODI does not force its own transformation engine on your process, everything is natively run in Hadoop MapReduce, and on your source and target databases.

Using ODI Studio



The UI has two panels: The panel on the left provides tabbed access to the navigators, and the panel on the right shows an editor.

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

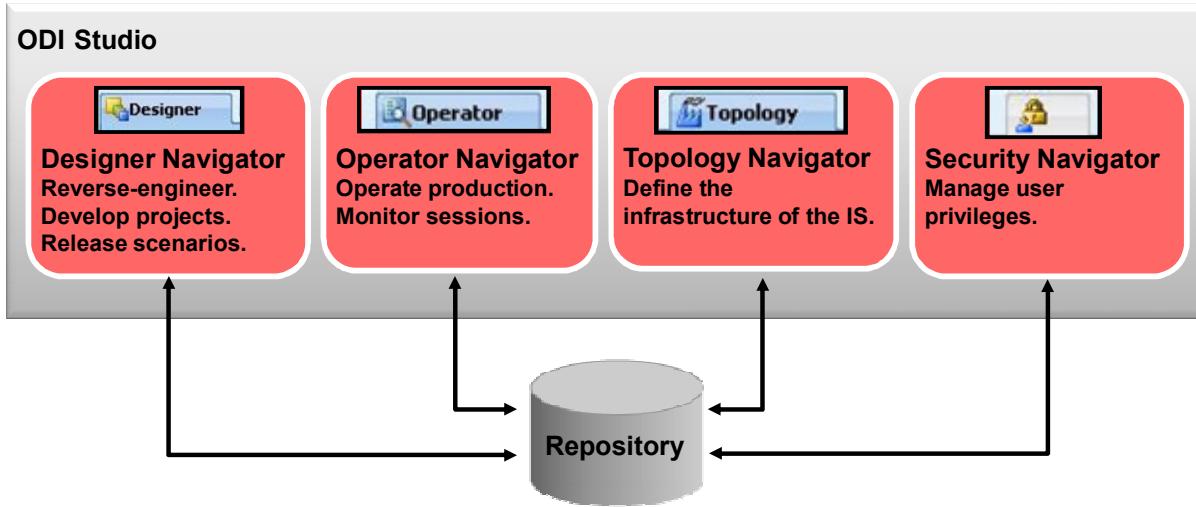
Oracle Data Integrator Studio is the GUI tool that administrators, developers, and operators use to access the repositories. This Fusion Client Platform (FCP)-based UI is used for:

- Administering the infrastructure (security and topology)
- Reverse-engineering the metadata
- Developing projects, scheduling, operating, and monitoring executions

FCP provides an efficient and flexible way to manage navigators, panels, and editors.

Business users (as well as developers, administrators, and operators) can have read access to the repository. They can also perform topology configuration and production operations through a web-based UI called Oracle Data Integrator Console. This web application can be deployed in a Java EE application server such as Oracle WebLogic.

ODI Studio Components: Overview



The Fusion Client Platform (FCP)–based UI provides an efficient and flexible way to manage navigators, panels, and editors.



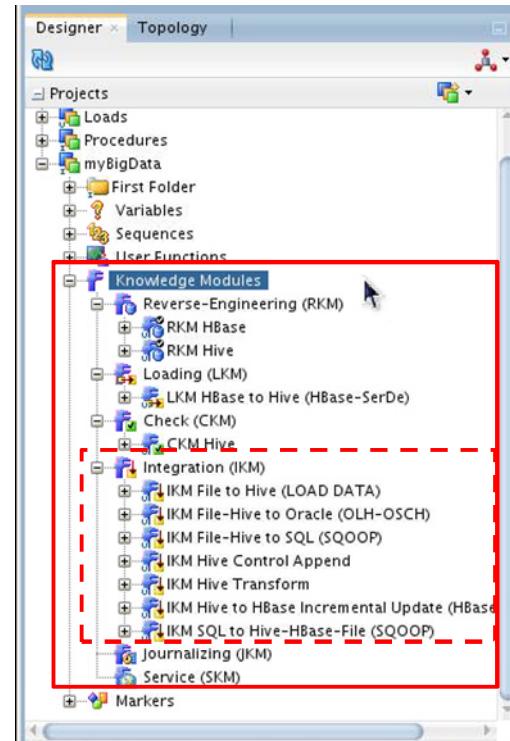
Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

ODI Studio provides four navigators for managing the different aspects and steps of an ODI integration project.

- ODI agents are runtime processes that orchestrate executions.
- ODI Console provides users web access to ODI metadata.
- ODI repositories store all of your ODI objects as databases in a relational database management system.

ODI Studio: Big Data Knowledge Modules

- Lots of KMs available
- More KMs coming



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

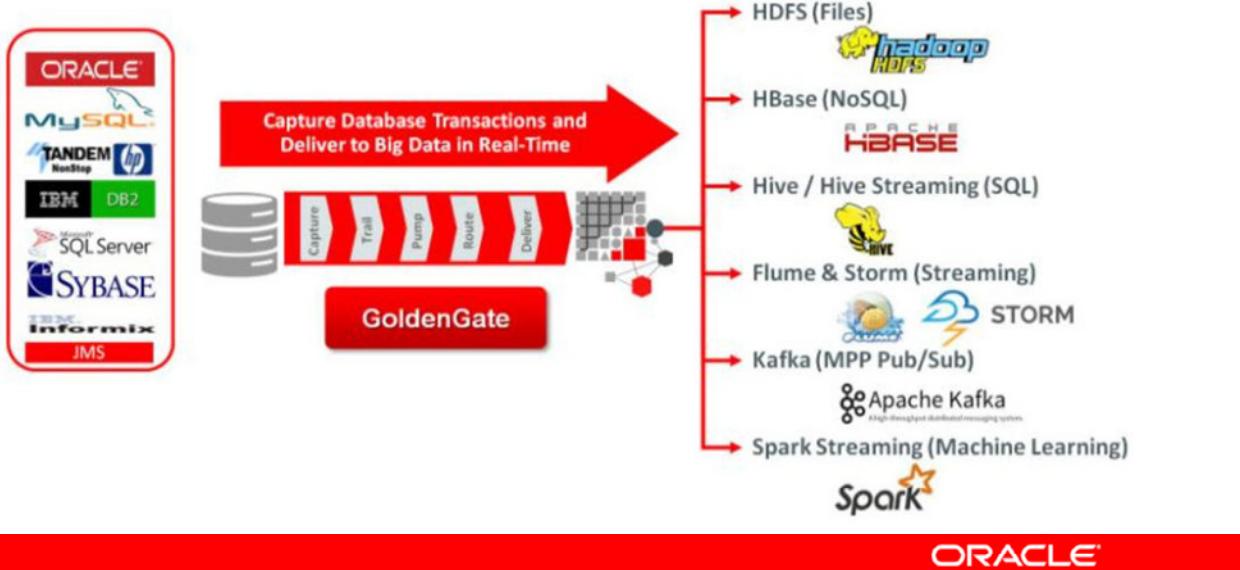
In the screenshot shown here, you can see the knowledge modules (KMs) for Hadoop listed in the Projects section under the Designer View of the Oracle Data Integrator.

Sample big data KMs include:

- RKM (Reverse Engineering)
 - RKM HBase
 - RKM Hive
- LKM (Loading): LKM HBase to Hive
- CKM (Checking): CKM Hive
- IKM (Integration)
 - IKM File to Hive
 - IKM File-Hive to Oracle
 - IKM File-Hive to SQL (Sqoop)
 - IKM Hive Control Append
 - IKM Hive Transform
 - IKM Hive to HBase Incremental Update
 - IKM SQL to Hive-HBase-File (Sqoop)

Oracle GoldenGate for Big Data

- Performs real-time replication and synchronization of data
- Streamlines real-time data delivery into big data formats including Apache Hadoop, Apache HBase, Apache Hive, and Apache Flume



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Oracle GoldenGate (OGG) is a time tested and proven product for real-time data replication and synchronization of tables or schemas. Using OGG, you can maintain two copies of the same data, with each copy being kept in sync with one another.

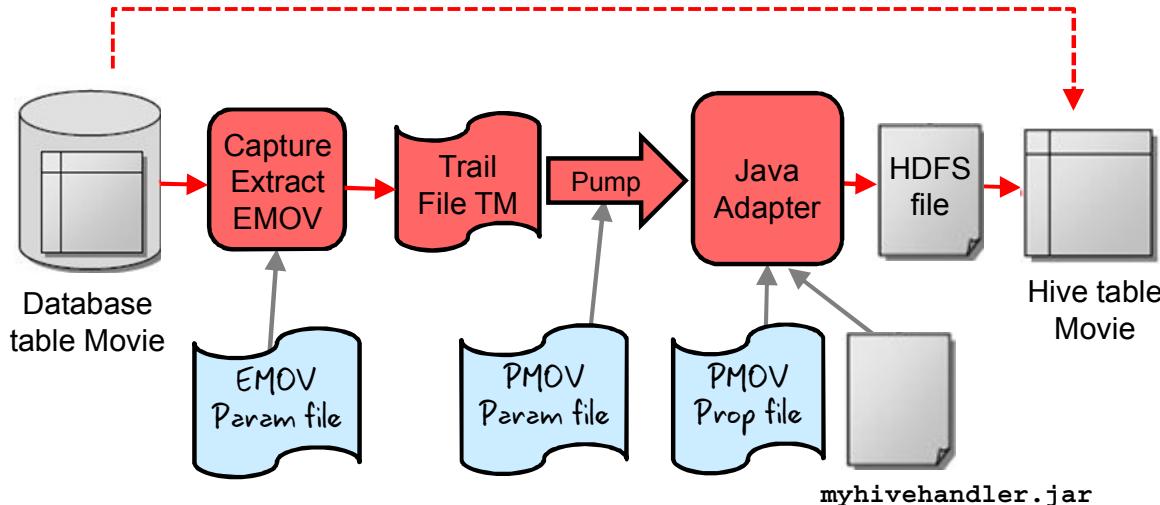
In a new offering, Oracle GoldenGate for Big Data, provides optimized and high performance delivery of this feature to the most popular big data formats by providing the following:

- OGG Adapter for Flume. Enables streaming real-time data into Apache Flume systems for fault-tolerant, highly reliable, and extensible real-time analytical applications.
- OGG Adapter for HDFS. Enables stream changed data to HDFS where the downstream applications may further process data natively on Hadoop.
- OGG Adapter for Hive. Provides real-time data to the Hive Data Store, the infrastructure provided on top of Hadoop for doing summarization and analysis.
- OGG Adapter for HBase. HBase, a non-relational database which runs on top of HDFS, adds transactional capabilities to Hadoop, allowing users to conduct updates, inserts and deletes. Using this adaptor, inserts, updates and deletes can be applied to HBase in real-time for deriving value from large data sets.

OGG for Big Data also includes Oracle GoldenGate for Java, which enables custom integration to additional big data systems, such as Oracle NoSQL, Kafka, Storm, and Spark.

Using OGG with Big Data

Replication of changes to a target system



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

For Big Data, OGG enables the capture of completed transactions from a source database, and then replicates these changes to a target system, such as HDFS. OGG serves as a noninvasive, high performance tool for capturing and applying these changes.

The diagram in the slide illustrates an OGG process that you could perform. In this scenario:

- A custom handler (Java Adaptor) is deployed as an integral part of the Oracle GoldenGate Pump process.
- The movie data extract/capture process is configured through the use of a parameters file (EMOV.prm).
- The Pump process and the Hive adapter are configured through the use of two files: a Pump parameter file (PMOV.prm) and a custom adapter properties file (PMOV.properties).
- The Pump process reads the Trail File created by the Oracle GoldenGate Capture process, and then passes the transactions to the Java adapter. Based on the configuration, the adapter then writes the transactions in the desired format (in this case an HDFS file), with the appropriate content loaded to a Hive table.

Resources

- The Oracle Big Data Learning Library

https://apexapps.oracle.com/pls/apex/f?p=44785:141:0:::141:P141_PAGE_ID,P141_SECTION_ID:27,615

- Data Integration sites on OTN:

<https://www.oracle.com/middleware/data-integration/enterprise-edition-big-data/resources.html>

<http://www.oracle.com/us/products/middleware/data-integration/goldengate/overview/index.html>

<http://www.oracle.com/technetwork/middleware/data-integrator/overview/index.html>



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Summary

In this lesson, you should have learned how:

- Oracle Data Integrator interacts with Hadoop
- Oracle GoldenGate for Big Data interacts with Hadoop



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Practice 21: Overview

In this practice, you will:

- Use Oracle Data Integrator Studio to create mappings that load data from Oracle Database into Hive tables by using a Scoop IKM.



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

22

Using Oracle Big Data SQL

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Course Road Map

Module 1: Big Data Management System

Module 2: Data Acquisition and Storage

Module 3: Data Access and Processing

Module 4: Data Unification and Analysis

Module 5: Using and Managing Oracle Big Data Appliance

Lesson 16: Options for Integrating Your Big Data

Lesson 17: Overview of Apache Sqoop

Lesson 18: Using Oracle Loader for Hadoop (OLH)

Lesson 19: Using Copy to BDA

Lesson 20: Using Oracle SQL Connector for HDFS

Lesson 21: Using ODI and OGG with Hadoop

Lesson 22: Using Oracle Big Data SQL

Lesson 23: Using Advanced Analytics:
Oracle Data Mining and Oracle R Enterprise

Lesson 24: Introducing Oracle Big Data Discovery

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

This lesson, on Oracle Big Data SQL, essentially serves as a bridge between the technologies that primarily provide data unification capabilities and those that provide analytic capabilities.

As you will learn, Big Data SQL enables dynamic, integrated access between Oracle database and Hadoop or NoSQL data for real-time analysis.

Objectives

After completing this lesson, you should be able to:

- Describe how Oracle Big Data SQL enables dynamic, integrated access between Oracle database and Hadoop or NoSQL
- Use Oracle Big Data SQL to perform integrated data analysis



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Barriers to Effective Big Data Adoption

INTEGRATION SKILLS SECURITY

| | | |
|---|---|---|
| Complexity of adding Big Data to existing IM architecture | Wide variety of technologies required to exploit Big Data | No standard for governance or enforcement |
|---|---|---|



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Why is not everyone using Big Data in an effective way today? Big Data offers incredible promise, but there are also barriers.

- **Integration:** As previously discussed in this course, the best practice approach is to integrate Big Data technologies and data into your current Information Management architecture. Your big data environment must not be an island with isolated infrastructure, processes, and analysis technologies, separated from the other important elements of your IM system. However, as you've learned integrating Big Data into your existing IM system can be a highly complex endeavor.
- **Skills:** As you have also learned, there are a wide variety of technologies and skill sets available within the Big Data ecosystem, including Hadoop, NoSQL, Pig, Hive, Spark, and others. Many Big Data implementers face a steep learning curve to leverage these technologies to deliver integrated, measureable value to the enterprise.
- **Security:** The issue of data security is an important and evolving challenge in the Big Data space. There is no true standard for governance or enforcement of data security.

Overcoming Big Data Barriers



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

This slide illustrates several elements provided by Oracle's Big Data solution that help customer overcome these barriers in the Big Data adoption process.

- Integration.
 - At the beginning of this course, you learned about the Oracle Information Management Reference Architecture and the value of integrating Big Data into your overall IM system. As mentioned, Oracle also provides Engineered Systems—integrated hardware and software solutions to address the challenges of big data—particularly with the combination of Oracle Big Data Appliance and Oracle Exadata Database Machine.
 - In fact, the last module of this course teaches you how to use and manage the Oracle Big Data Appliance.
- Skills. In this lesson, you learn to use Oracle Big Data SQL, which is an option on the BDA. Big Data SQL enables you to view and analyze data from various data stores in your big data platform, as if it were all stored together in an Oracle database.
- Security. Big Data SQL also provides access to a robust security framework for all of your data, including data inside and outside of Oracle Database.

Oracle Big Data SQL

- Provides powerful, high-performance SQL on a variety of Big Data sources
- Enables easy integration of Big Data and Oracle Database data
- Licensed as an option on the BDA
- Leverages optimized engineered systems



ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Oracle Big Data SQL, which is a licensed option on Oracle BDA, enables SQL query access to a variety of big data formats, including Apache Hive, HDFS, Oracle NoSQL Database, and Apache HBase.

Oracle Big Data SQL runs exclusively on systems with Oracle Big Data Appliance connected to Oracle Exadata Database Machine. It is optimized for this engineered platform that includes high-speed InfiniBand network between Hadoop and Exadata.

Big Data SQL provides for:

- Complete Oracle SQL query capabilities for data stored in a Hadoop cluster
- Data integration of Hadoop and Oracle Database, resulting in a single SQL point-of-entry to access all data
- Scalable joins between Hadoop and RDBMS data

Using Oracle Big Data SQL, you can execute the most complex SQL SELECT statements against data in Hadoop, either manually or by using your existing applications.

For example, users of the Oracle Advanced Analytics database option can apply their data mining models, which reside in Oracle Database, to big data that is resident on Oracle Big Data Appliance.

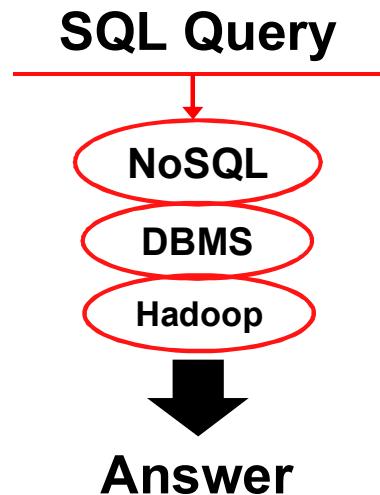
Goal and Benefits

Goal

- Provide a single SQL interface for all data

Benefits

- One engine
- Supports all data
- No stovepipes
- No federation
- Database security
- Optimized query performance



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Oracle Big Data Management System unifies the data platform by providing a common query language, management platform, and security framework across Hadoop, NoSQL, and Oracle Database.

Oracle Big Data SQL is a key component of the platform. It enables Oracle Exadata to seamlessly query data on the Oracle Big Data Appliance by using Oracle's rich SQL dialect. Data stored in Hadoop or Oracle NoSQL Database is queried in exactly the same way as all other data in Oracle Database. This means that users can begin to gain insights from these new sources by using their existing skill sets and applications.

Big Data SQL:

- Provides one engine with native SQL operators that supports query access to all of the data in your IM system
- Enables you to apply Oracle Database security features to your non-RDBMS data
- Supports optimized engineered system services, including Smart Scan technology that is found in Oracle Exadata. This technology enables Oracle Big Data Appliance to discard a huge portion of irrelevant data—up to 99 percent of the total—and return much smaller result sets to Oracle Exadata Database Machine. End users obtain the results of their queries significantly faster, as the direct result of a reduced load on Oracle Database and reduced traffic on the network.

Using Oracle Big Data SQL

- Setup:
 1. Configure Oracle Big Data SQL.
 2. Create Oracle External Tables:
 - Provides data access to various Hadoop data formats
 - Leverages the Hive metastore for data access
- Use:
 - Apply Oracle Database security over data in Hadoop
 - Use SQL across all of your data:
 - Basic SELECTs
 - Oracle Analytic SQL
 - SQL Pattern Matching
 - Whatever you need...



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

There are two required steps to set up Oracle Big Data SQL.

1. First, perform several simple configuration tasks on both the Exadata machine and BDA. These configuration tasks are covered in the next several slides.
2. Then, create Oracle External Tables over your Hadoop data to provide access to the underlying data. You can create external tables:
 - Directly over files stored in HDFS by using an access driver named `ORACLE_HDFS`. This driver uses Hive syntax to describe the data source.
 - Over Apache Hive data sources when you already have Hive tables defined for your HDFS data sources, by using an access driver named `ORACLE_HIVE`. This driver is designed to acquire the metadata directly from the Hive metadata store.

Once you have set up Big Data SQL, you can:

- Apply Oracle security policies over your Hadoop data, if needed
- Use SQL to query and analyze all of your data

Configuring Oracle Big Data SQL

1. Create the Common and Cluster Directories on the Exadata Server.
2. Deploy configuration files to the directories:
 - Create and populate the `bigdata.properties` file in the Common Directory.
 - Copy the Hadoop configuration files into the Cluster Directory.
3. Create corresponding Oracle directory objects that reference the configuration directories.
4. Install the required software.



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

There are four basic tasks required to configure Oracle Big Data SQL, including:

1. Creating configuration directories
2. Deploying configuration files
3. Creating corresponding Oracle Directory objects
4. Installing the required software

Each of these tasks is examined in the following slides.

Task 1: Create System Directories on Exadata

Common Directory

- Must be located on a cluster-wide file system
- Contains the `bigdata.properties` file

```
mkdir /u01/bigdatasql_config
```

Cluster Directory

- Must be a subdirectory of the Common directory
- Contains BDA cluster connection configuration files

```
mkdir /u01/bigdatasql_config/bigdatalite
```



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Two file system directories—the Common and Cluster directories—are required on the Exadata Server. These directories store configuration files that enable the Exadata Server to connect to the BDA. A short description of each follows.

Common directory:

- Contains a few subdirectories and one important file, named `bigdata.properties`. This file stores configuration information that is common to all BDA clusters. Specifically, it contains property-value pairs used to configure the JVM and identify a default cluster. This is described in the next slide.
- The `bigdata.properties` file must be accessible to the operating system user under which the Oracle Database runs.
- For Exadata, the Common directory must be on a clusterwide file system. It is critical that all Exadata Database nodes access the exact same configuration information.

Cluster directory:

- Contains configuration files required to connect to a specific BDA cluster. It also must be a subdirectory of the Common directory.
- In addition, the Cluster directory name is important. It is the name that you will use to identify the cluster. This is described in the next slide.

Task 2: Deploy Configuration Files

- Create `bigdata.properties` file in Common directory.

```
cd /u01/bigdatasql_config/  
cat bigdata.properties
```

```
oracle@bigdatalite:/u01/bigdatasql_config  
File Edit View Search Terminal Help  
java.libjvm.file=/usr/java/jdk1.7.0_51/jre/lib/amd64/server/libjvm.so  
java.classpath.oracle=/u01/app/oracle/product/12.1.0/dbhome_1/jlib/ora  
java.classpath.hadoop=/usr/lib/hadoop/client-0.20/*:/usr/lib/hadoop-0.  
java.classpath.hive=/usr/lib/hive/lib/*  
LD_LIBRARY_PATH=/usr/java/jdk1.7.0_51/jre/lib/lib  
bigdata.cluster.default=bigdatalite
```

- Copy Hadoop configuration files to the Cluster directory.

```
oracle@bigdatalite:/u01/bigdatasql_config/bigdatalite  
File Edit View Search Terminal Help  
[oracle@bigdatalite ~]$ cd /u01/bigdatasql_config/bigdatalite  
[oracle@bigdatalite bigdatalite]$ ls  
core-site.xml hdfs-site.xml hive-env.sh hive-site.xml mapred-site.xml
```

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

First, create the `bigdata.properties` file in the Common directory. As previously mentioned, it stores configuration information that is common to all BDA clusters. In the slide example, the `bigdata.properties` file describes the following:

- Items such as the location of the Java VM, classpaths, and the `LD_LIBRARY_PATH`. These properties not specific to a Hadoop cluster.
- The default cluster property, shown on the last line of the file:
 - In this example, the default cluster property value is `bigdatalite`.
 - The name of the cluster must match the name of the Cluster subdirectory—and it is case-sensitive!

Second, copy the required Hadoop configuration files to the Cluster directory. As previously mentioned, these configuration files enable connection to a specific BDA cluster. The files you copy to this directory will depend on the data formats that you want to access.

In our example, the configuration files enable Oracle Database to connect to HDFS and Hive.

Note

- In the practice, there is a single cluster: `bigdatalite`. Therefore, the `bigdatalite` subdirectory contains the configuration files for the `bigdatalite` cluster.
- As you will see later, the default cluster setting simplifies the definition of Oracle tables that will be accessing data in Hadoop.

Task 3: Create Oracle Directory Objects

- Oracle Directory Object for Common Directory

```
create or replace directory ORACLE_BIGDATA_CONFIG as '/u01/bigdatasql_config';
```

- Oracle Directory Object for Cluster Directory

```
create or replace directory "ORA_BIGDATA_CL_bigdatalite" as '';
```

- *Recommended Practice:* Also create the Big Data SQL Multi-threaded Agent (MTA)

```
create public database link BDSQL$_bigdatalite using 'extproc_connection_data';
create public database link BDSQL$_DEFAULT_CLUSTER using 'extproc_connection_data';
```



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

As previously shown, the required configuration files are created in, and copied to, the Common and Cluster directories on the Exadata file system. Next, you define corresponding Oracle Directory objects that reference these two configuration directories.

The two Oracle Directory objects are listed here, and have a specific naming convention:

- ORACLE_BIGDATA_CONFIG references the Common directory
- ORACLE_BIGDATA_CL_bigdatalite references the Cluster directory. The naming convention for this directory is as follows:
 - The Cluster directory object name must begin with ORACLE_BIGDATA_CL_
 - This is followed by the cluster name (bigdatalite). Recall that this name is case-sensitive, and is limited to 15 characters.
 - The cluster name must match the physical Cluster directory name in the file system.

Note: If a directory object name is in mixed, then it must be specified within double quotes when you create the object. Notice how this is the case for the cluster directory object.

Recommended Practice: In addition to the Oracle directory objects, you should also create the Big Data SQL Multi-threaded Agent (MTA). This agent bridges the metadata between Oracle Database and Hadoop. Technically, the MTA allows the external process to be multithreaded, instead of launching a JVM for every process.

Task 4: Install Required Software

- Install Oracle Big Data SQL on the BDA by using Mammoth—the BDA's installation and configuration utility.
- Install a CDH client on each Exadata Server.



ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

For detailed information on installing and configuring Oracle Big Data SQL See:

- For the Big Data Appliance:
 - https://docs.oracle.com/cd/E57371_01/doc.41/e57351/admin.htm#BIGUG76694
- To set up Exadata so it can use Big Data SQL:
 - https://docs.oracle.com/cd/E57371_01/doc.41/e57351/bigsql.htm#BIGUG76720

Create External Tables Over HDFS Data and Query the Data

- Create a table named ORDER to access data in all files stored in the /usr/cust/summary directory in HDFS:

```
CREATE TABLE ORDER (cust_num      VARCHAR2(10),
                    order_num      VARCHAR2(20),
                    order_total    NUMBER (8,2))
ORGANIZATION EXTERNAL (TYPE oracle_hdfs)
LOCATION ("hdfs:/usr/cust/summary/*");
```

- Query the HDFS data:

```
SELECT cust_num, order_num
FROM   order
WHERE  order_total >= 100;
```



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Recall from an earlier lesson that Oracle External Tables are objects that identify and describe the location of data outside of a database. With Big Data SQL, you enable query access to supported data formats on the BDA by creating external tables.

External Tables Over HDFS Data

The example in the slide shows how to create an external table over data stored in HDFS files, by using the ORACLE_HDFS access driver. Recall that this driver enables you to access many types of data that are stored in HDFS, but which do not have Hive metadata.

Because no access parameters are set in the statement, the ORACLE_HDFS access driver uses its default settings to do the following:

- Connect to the default Hadoop cluster
- Read the files as delimited text, and the fields as type STRING
- Reject any records in which the value causes a data conversion error.

Note: The statement also assumes that the number of fields, and the order of the fields, in the HDFS files match the number and order of columns in the table. Therefore, CUST_NUM must be in the first field, ORDER_NUM in the second field, and ORDER_TOTAL in the third field.

Then, you may query the HDFS immediately, as shown in the simple SELECT statement.

Using Access Parameters with oracle_hdfs

1.

```
CREATE TABLE order (cust_num      VARCHAR2(10),
                     order_num      VARCHAR2(20),
                     order_total    NUMBER (8,2),
                     order_date     DATE,
                     item_cnt       NUMBER,
                     description    VARCHAR2(100))

ORGANIZATION EXTERNAL (TYPE oracle_hdfs
  ACCESS PARAMETERS (
    com.oracle.bigdata.colmap:   {"col":"item_cnt", \
                                  "field":"order_line_item_count"}
    com.oracle.bigdata.overflow: {"action":"TRUNCATE", \
                                  "col":"DESCRIPTION"}
  )
LOCATION ("hdfs:/usr/cust/summary/*");
```
2.

```
SELECT cust_num, item_cnt, order_total, description
FROM   order
WHERE  order_date = 12-24-2014;
```

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

You can override the default behavior of the ORACLE_HDFS access driver by setting properties in the ACCESS PARAMETERS clause of the external table definition.

Example

The example:

1. Expands upon the CREATE TABLE ... ORGANIZATION EXTERNAL statement shown in the previous slide. Here, the statement:
 - Identifies three additional fields or columns: order_date, item_cnt, and description
 - Specifies two properties within the ACCESS PARAMETERS clause:
 - com.oracle.bigdata.colmap: Handles differences in column names. In this example, ORDER_LINE_ITEM_COUNT in the HDFS files matches the ITEM_CNT column in the external table.
 - com.oracle.bigdata.overflow: Truncates string data. Values longer than 100 characters for the DESCRIPTION column are truncated.
2. Shows a simple SELECT statement that leverages the new external table definition.

Note: There are a variety of other access parameter properties that may be used with the ORACLE_HDFS access driver. See *Oracle Big Data SQL Reference* for more information.

Create External Tables to Leverage the Hive Metastore and Query the Data

- Creates a table named ORDER to access Hive data:

```
CREATE TABLE ORDER (cust_num      VARCHAR2(10),
                    order_num      VARCHAR2(20),
                    description    VARCHAR2(100),
                    order_total    (NUMBER 8,2))
ORGANIZATION EXTERNAL (TYPE oracle_hive);
```

- Query the Hive data:

```
SELECT cust_num, order_num, description
FROM   order
WHERE  order_total >= 100;
```



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

You can easily create an Oracle external table for data in Apache Hive that leverages the Hive Metastore by using the ORACLE_HIVE access driver.

External Tables that Leverage the Hive Metastore

The statement example in the slide shows how to create an external table over Apache Hive data, by using the ORACLE_HIVE access driver. Because no access parameters are set in the statement, the ORACLE_HIVE access driver uses the default settings to do the following:

- Connect to the default Hadoop cluster
- Uses a Hive table named `order`. An error results if the Hive table does not have fields named `CUST_NUM`, `ORDER_NUM`, `DESCRIPTION`, and `ORDER_TOTAL`.
- Sets the value of a field to `NULL` if there is a conversion error, such as a `CUST_NUM` value longer than 10 bytes.

As before, you may immediately query the Hive data by using SQL.

Using Access Parameters with oracle_hive

1.

```
CREATE TABLE order (cust_num      VARCHAR2(10),
                     order_num      VARCHAR2(20),
                     description    VARCHAR2(100)
                     order_total    NUMBER (8,2),
                     order_date     DATE,
                     item_cnt       NUMBER)

ORGANIZATION EXTERNAL (TYPE oracle_hive
ACCESS PARAMETERS (
  com.oracle.bigdata.tablename: order_db.order_summary
  com.oracle.bigdata.colmap: {"col":"item_cnt", \
                             "field":"order_line_item_count"}
  com.oracle.bigdata.overflow: {"action":"TRUNCATE", \
                                "col":"DESCRIPTION"}
)
);
```
2.

```
SELECT cust_num, item_cnt, order_total, description
FROM   order
WHERE  order_date = 12-24-2014;
```



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

You can also override the default behavior of the ORACLE_HIVE access driver by setting properties in the ACCESS PARAMETERS clause of the external table definition.

Example

1. The example expands upon the CREATE TABLE ... ORGANIZATION EXTERNAL statement shown in the previous slide. Here, the statement:
 - Identifies two additional fields or columns: `order_date` and `item_cnt`
 - Specifies three properties within the ACCESS PARAMETERS clause:
 - `com.oracle.bigdata.tablename`: Handles differences in table names. In this example, ORACLE_HIVE looks for a Hive table named ORDER_SUMMARY in the ORDER.DB database.
 - `com.oracle.bigdata.colmap`: Handles differences in column names. In this example, the Hive ORDER_LINE_ITEM_COUNT field maps to the Oracle ITEM_CNT column.
 - `com.oracle.bigdata.overflow`: Truncates string data. Values longer than 100 characters for the DESCRIPTION column are truncated.
2. Then, a simple SELECT statement may be executed against Hive data immediately.

Accessing Hadoop and Enterprise Data Together

1. Create an external table over **Hive** data.

```
CREATE TABLE customer_address (customer_id    number(10,0),
                               street_number  char(10),
                               street_name   varchar(60),
                               city          varchar(60),
                               county        varchar(30),
                               state         char(2),
                               zip           char(10))

ORGANIZATION EXTERNAL (TYPE orecle_hive
  DEFAULT DIRECTORY DEFAULT_DIR
  ACCESS PARAMETERS (com.oracle.bigdata.cluster hadoop_c1_1)
  LOCATION ('hive://customer_address')
)
```

2. Seamlessly join **Hive** and **RDBMS** data.

```
SELECT a.customer_id, a.customer_last_name, b.county, b.state
FROM   customers a, customer_address b
WHERE  a.customer_id = b.customer_id;
```

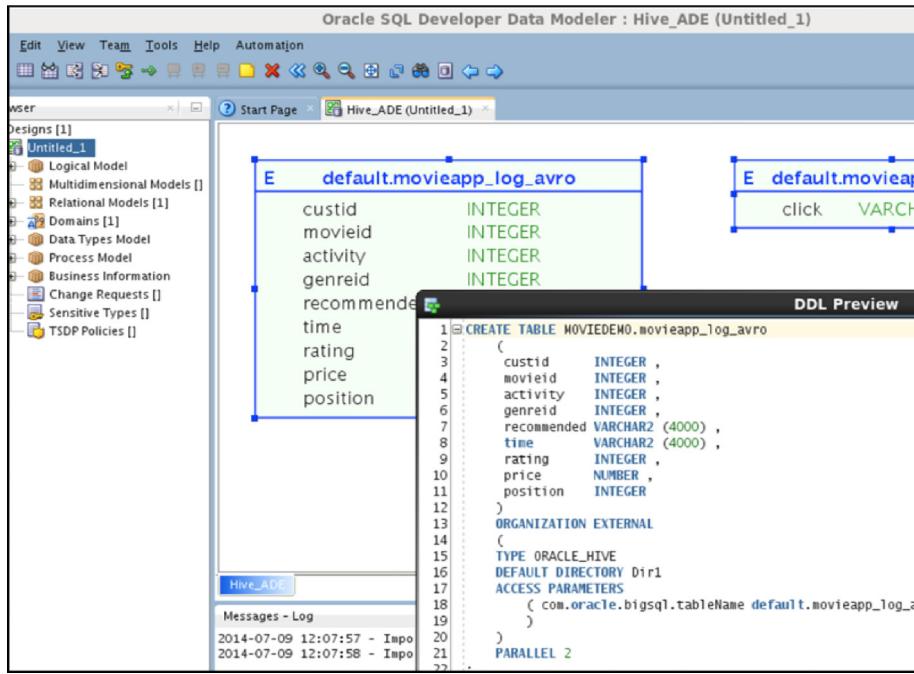


Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

This slide shows a simple example of how to join RDBMS and Hadoop data.

1. First, create an external table over the appropriate Hive table. The external table is named `customer_address` (color coded in red).
2. Then, data in Hadoop (color coded in red) may be joined with data in the `customers` table in Oracle Database (color coded in blue) via a `SELECT` statement to produce an integrated result.

Automating External Table Creation



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Oracle Data Modeler, which is part of Oracle SQL Developer, provides a feature that automates the process of creating external tables for Hadoop data sources.

Using this feature, you can:

- Generate the Oracle DDL for external table creation (as shown in the slide)
- Import Hive definitions

If you have a large number of external tables to create, this feature makes the definition process much more efficient.

Applying Oracle Database Security Policies

```
DBMS_REDACT.ADD_POLICY(
    object_schema => 'MOVIEDEMO',
    object_name => 'CUSTOMER',
    column_name => 'CUST_ID',
    policy_name => 'customer_redaction',
    function_type => DBMS_REDACT.PARTIAL,
    function_parameters => '9,1,7',
    expression => '1=1'
);

DBMS_REDACT.ALTER_POLICY(
    object_schema => 'MOVIEDEMO',
    object_name => 'CUSTOMER',
    action => DBMS_REDACT.ADD_COLUMN,
    column_name => 'LAST_NAME',
    policy_name => 'customer_redaction',
    function_type => DBMS_REDACT.PARTIAL,
    function_parameters =>
'XXXXXXXXXXXXXXXXXXXX,XXXXXXXXXXXXXXXXXXXX,* ,3,25',
    expression => '1=1'
);
```



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

In most deployments, the Oracle Database contains critical and sensitive data that must be protected. A rich set of Oracle Database security features, including strong authentication, row level access, data redaction, data masking, auditing and more, may be utilized to ensure that data remains safe. These same security policies can be leveraged when using Oracle Big Data SQL. This means that a single set of security policies can be utilized to protect all of your data.

Example

In this example, using the Moviedemo sample data, we need to protect personally identifiable information, including the customer last name and customer ID. To accomplish this task, we set up an Oracle Data Redaction policy on the customer table that obscures these two fields. This is accomplished by using the DBMS_REDACT PL/SQL package, shown in the slide.

The first PL/SQL call creates a policy called `customer_redaction`:

- It is applied to the `cust_id` column in the `moviedemo.customer` table.
- It performs a partial redaction, replacing the first seven characters with the number "9".
- The redaction policy will always apply, because the expression describing when it will apply is specified as "1=1".

The second API call updates the `customer_redaction` policy, redacting a second column in that same table. It will replace the characters 3 to 25 in the `LAST_NAME` column with an "*".

Note: The application of redaction policies does not change underlying data. Oracle Database performs the redaction at execution time, just before the data is displayed to the application user.

Viewing the Results

```
SELECT cust_id, last_name FROM customers;
```

| CUST_ID | LAST_NAME |
|---------|-----------------|
| 1 | 9999999 Re***** |
| 2 | 9999999 Su***** |
| 3 | 9999999 KJ***** |
| 4 | 9999999 Ro***** |
| 5 | 9999999 Ly***** |
| 6 | 9999999 Sr***** |
| 7 | 9999999 Sa***** |
| 8 | 9999999 Ka***** |
| 9 | 9999999 Ro***** |
| 10 | 9999999 Ug***** |



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

When you query the two columns in the `customers` table, the redaction policy produces the result shown in the slide.

Importantly, SQL executed against the redacted data remains unchanged. For example, queries can use the `cust_id` and `last_name` columns in join conditions, apply filters to them, and so on. The fact that the data is redacted is transparent to application code.

Next, you learn how to apply redaction policies to external tables that have data sourced in Hadoop.

Applying Redaction Policies to Data in Hadoop

```
BEGIN
    -- JSON file in HDFS
    DBMS_REDACT.ADD_POLICY(
        object_schema => 'MOVIEDEMO',
        object_name => 'MOVIELOG_V',
        column_name => 'CUSTID',
        policy_name => 'movielog_v_redaction',
        function_type => DBMS_REDACT.PARTIAL,
        function_parameters => '9,1,7',
        expression => '1=1'
    );

    -- Avro data from Hive
    DBMS_REDACT.ADD_POLICY(
        object_schema => 'MOVIEDEMO',
        object_name => 'MYLOGDATA',
        column_name => 'CUSTID',
        policy_name => 'mylogdata_redaction',
        function_type => DBMS_REDACT.PARTIAL,
        function_parameters => '9,1,7',
        expression => '1=1'
    );
END;
/
```



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

In the example shown here, we apply an equivalent redaction policy to two of the Oracle Big Data SQL tables in the Moviedemo data set: The redaction policies have the following effects:

- The first procedure redacts data sourced from JSON in HDFS.
- The second procedure redacts Avro data sourced from Hive.
- Both policies redact the `custid` attribute.

As a result, the `custid` column for both objects are now redacted.

Viewing Results from the Hive (Avro) Source

```
SELECT * FROM mylogdata WHERE rownum < 20;
```

| CUSTID | MOVIEID | GENREID | TIME | RECOMMENDED | ACTIVITY | RATING | PRICE |
|------------|---------|---------|--------------------------|-------------|-----------------|--------|-------|
| 1.99999999 | 0 | 0 | 2012-07-01:00:00:07 null | | 8 (null) (null) | | |
| 2.99999999 | 1948 | 9 | 2012-07-01:00:00:22 N | | 7 (null) (null) | | |
| 3.99999999 | 0 | 0 | 2012-07-01:00:00:26 null | | 9 (null) (null) | | |
| 4.99999999 | 11547 | 6 | 2012-07-01:00:00:32 Y | | 7 (null) (null) | | |
| 5.99999999 | 11547 | 6 | 2012-07-01:00:00:42 Y | | 6 (null) (null) | | |
| 6.99999999 | 0 | 0 | 2012-07-01:00:00:43 null | | 8 (null) (null) | | |
| 7.99999999 | 0 | 0 | 2012-07-01:00:00:50 null | | 9 (null) (null) | | |
| 8.99999999 | 608 | 6 | 2012-07-01:00:01:03 N | | 7 (null) (null) | | |
| 9.99999999 | 0 | 0 | 2012-07-01:00:01:07 null | | 9 (null) (null) | | |



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

As shown in the slide, the `custid` column displays a series of 9s instead of the original value.

Viewing the Results from Joined RDBMS and HDFS Data

```
SELECT j.custid, c.last_name, j.movieid, j.time
FROM   customer c, movielog_v j
WHERE  c.cust_id = j.custid;
```

| CUSTID | LAST_NAME | MOVIEID | TIME |
|--------|------------------|---------|---------------------|
| 1 | 99999999 La**** | (null) | 2012-07-01:00:00:07 |
| 2 | 99999999 Bu**** | 1948 | 2012-07-01:00:00:22 |
| 3 | 99999999 Cu***** | (null) | 2012-07-01:00:00:26 |
| 4 | 99999999 Ha***** | 11547 | 2012-07-01:00:00:32 |
| 5 | 99999999 Re**** | 11547 | 2012-07-01:00:00:42 |
| 6 | 99999999 Go***** | (null) | 2012-07-01:00:00:43 |
| 7 | 99999999 On***** | (null) | 2012-07-01:00:00:50 |
| 8 | 99999999 Re***** | 608 | 2012-07-01:00:01:03 |
| 9 | 99999999 Go***** | (null) | 2012-07-01:00:01:07 |
| 10 | 99999999 El***** | 27205 | 2012-07-01:00:01:18 |
| 11 | 99999999 Be***** | 1124 | 2012-07-01:00:01:26 |



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Now, we join the redacted HDFS data in the JSON file (color coded in red) to the `customer` table in Oracle Database (color coded in blue) by executing the `SELECT` statement shown in the slide.

Results:

- As highlighted in the example output, we used the **Sort** tool in SQL Developer to sort the output in ascending order by the `TIME` column.
- The redacted data sourced from Hadoop works seamlessly with the rest of the data in your Oracle Database.

Summary

In this lesson, you should have learned how to:

- Describe how Oracle Big Data SQL enables dynamic, integrated access between Oracle database and Hadoop or NoSQL
- Use Oracle Big Data SQL to perform integrated data analysis



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Practice 22: Overview

In this practice, you will:

- Complete the setup for a partially configured Big Data SQL environment
- Create external tables over Hadoop data and query the data
- Apply Oracle Database Security over Hadoop data
- Use Analytic SQL features on activity data in Hadoop joined with enterprise data in the data warehouse



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Using Oracle Advanced Analytics: Oracle Data Mining and Oracle R Enterprise

23

ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Course Road Map

Module 1: Big Data Management System

Module 2: Data Acquisition and Storage

Module 3: Data Access and Processing

Module 4: Data Unification and Analysis

Module 5: Using and Managing Oracle Big Data Appliance

Lesson 16: Options for Integrating Your Big Data

Lesson 17: Overview of Apache Sqoop

Lesson 18: Using Oracle Loader for Hadoop (OLH)

Lesson 19: Using Copy to BDA

Lesson 20: Using Oracle SQL Connector for HDFS

Lesson 21: Using ODI and OGG with Hadoop

Lesson 22: Using Oracle Big Data SQL

**Lesson 23: Using Advanced Analytics:
Oracle Data Mining and Oracle R Enterprise**

Lesson 24: Introducing Oracle Big Data Discovery



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

This lesson provides an overview of the two products that comprise the Oracle Advanced Analytics Option to Oracle Database 12c: Oracle Data Mining and Oracle R Enterprise.

In this lesson, you learn how to use these two technologies to perform sophisticated predictive and statistical analysis on both unstructured and structured data. In addition, the lesson provides individual hands-on practice with each technology.

Objectives

After completing this lesson, you should be able to:

- Define Oracle Advanced Analytics
- Describe the uses and benefits of Oracle Data Mining
- Describe the uses and benefits of Oracle R Enterprise

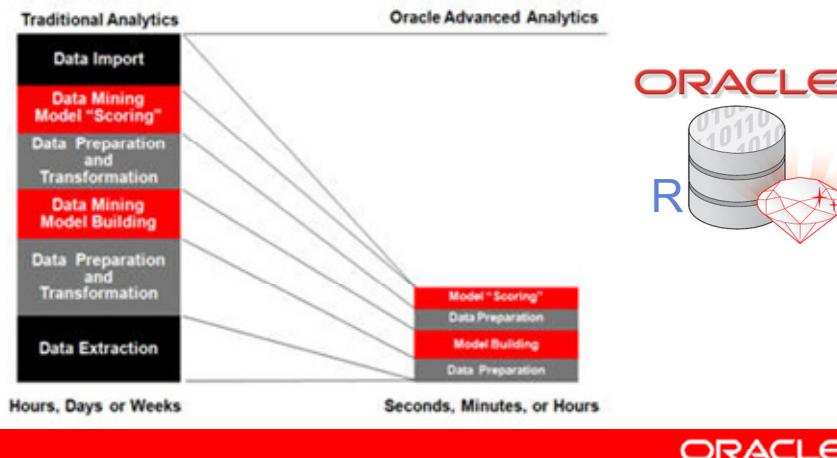


Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Oracle Advanced Analytics (OAA)

OAA = Oracle Data Mining + Oracle R Enterprise

- **Performance and Scalability.** Data management, preparation and transformations, model building and model scoring run as database processes.
- **Enterprise Predictive Analytics.** The database becomes the scalable and secure platform for delivering enterprise predictions and insights to applications.
- **Lowest Total Costs of Ownership.** No need for separate analytical servers.
- **Ease of Production Deployment**



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Oracle Advanced Analytics (OAA) is an Option to Oracle Database Enterprise Edition, which extends the database into a comprehensive advanced analytics platform for big data.

The Oracle Advanced Analytics Option is a comprehensive advanced analytics platform comprising Oracle Data Mining and Oracle R Enterprise.

- Oracle Data Mining delivers predictive analytics, data mining, and text mining.
- Oracle R Enterprise enables R programmers to leverage the power and scalability of Oracle Database, while also delivering R's world-class statistical analytics, advanced numerical computations, and interactive graphics.

With these two technologies, OAA brings powerful computations to the database, resulting in dramatic improvements in information discovery, scalability, security, and savings.

Data analysts, data scientists, statistical programmers, application developers, and DBAs can develop and automate sophisticated analytical methodologies inside the database and gain competitive advantage by leveraging the OAA Option.

OAA: Oracle Data Mining



ORACLE®

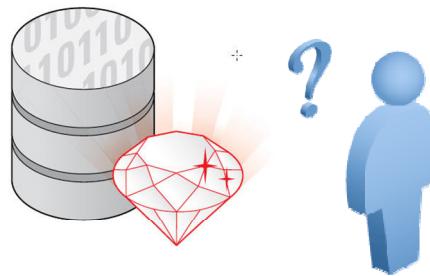
Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

First, you learn about Oracle Data Mining.

What Is Data Mining?

Definitions of the term *data mining* include the following:

- The extraction of implicit, previously unknown, and potentially useful information from data
- The process of exploration and analysis, by automatic or semiautomatic means, of large quantities of data to discover meaningful patterns and rules



ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Before we examine the Oracle Data Mining product, let us define some general data mining terms and concepts.

Data Mining goes beyond simple analysis. It can answer questions that cannot be addressed through simple query and reporting techniques.

Data mining is the practice of automatically searching large stores of data to discover patterns and trends. It uses sophisticated mathematical algorithms to segment the data and evaluate the probability of future events.

Data mining enables:

- Automatic discovery of patterns
 - Knowledge about what happened in the past
 - Characterization, segmentation, comparisons, discrimination
 - Descriptive models of patterns
- Prediction of likely outcomes, providing better decisions and forecasts
 - Knowledge about what is happening right now and in the future
 - Classification and prediction of patterns
 - Rule-and-model driven
- Creation of actionable information

Common Uses of Data Mining

- Targeting the right customer with the right offer
- Discovering hidden customer segments
- Finding most profitable selling opportunities
- Anticipating and preventing customer churn
- Exploiting the full 360-degree customer opportunity
- Detecting suspicious (anomalous) activity
- Understanding sentiments in customer conversations
- Reducing medical errors and improving health quality
- Understanding influencers in social networks
- Many more examples



ORACLE

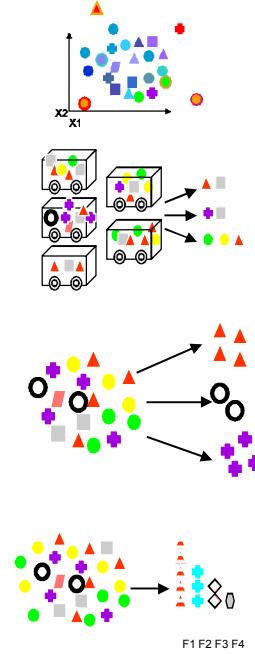
Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

There are scores of common applications for data mining in the current socio-economic environment, with the list of uses growing rapidly. Some of these include:

- **Higher Education:** Alumni donations, student acquisition and retention
- **Healthcare:** Patient procedure recommendation, patient outcome prediction, fraud detection, analysis of doctor's and nurse's notes
- **Public Sector:** Crime analysis, pattern recognition in military surveillance
- **Automotive:** Feature bundling for customer segments, supplier quality analysis
- **Chemical:** Discovery of new compounds, molecule clustering, product yield analysis
- **Utilities:** Prediction of powerline and equipment failure, product bundling, consumer fraud detection
- **Retail:** Customer segmentation, response modeling, recommendations for next likely products, profiling high-value customers
- **Life Sciences:** Drug discovery and interaction, common factors in (un)healthy patients, drug safety surveillance
- **Telecommunications:** Identifying cross-sell opportunities, customer churn, network intrusion detection

Defining Key Data Mining Properties

- Finding patterns and relationships
- Automatic discovery
- Predicting outcomes or values
- Grouping
- Creating actionable information



ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Finding Patterns and Relationships

One of the foundational properties of data mining is the discovery of patterns and/or relationships in your data. Whether you are attempting to predict a specific outcome based on certain patterns, or you want a data mining process to find unknown patterns or relationships, this concept of pattern and relationship discovery is at the core of data mining practice.

Automatic Discovery

The notion of automatic discovery refers to the execution of data mining models. Data mining is accomplished by building models, and a model uses an algorithm to act on a set of data.

Data mining models:

- Can be used to mine the data on which they are built
- Are commonly generalizable to new data. Application of a model to new data results in a process known as *scoring*.

Predicting Outcomes or Values

Many forms of data mining are predictive. For example, you might want to predict whether a customer is likely to buy a certain product (Yes or No). Or, you might want to predict how much a customer will spend at your store or on your website (an amount of money).

Depending on the model used to create predictions, certain associated metrics are generated along with the prediction. Some of these include:

- **Probability:** *How likely is this prediction to be true?*
- **Predictive confidence:** *How confident can I be of this particular prediction?*
- **Rules:** Some forms of predictive data mining generate rules, which are conditions that imply a given outcome. Rules have an associated *support*, which answers the following question: *What percentage of the population satisfies the rule?*

Note: If the problem that you are solving requires rules, you must pick one of the models that generates rules, such as Decision Tree.

Grouping

Other forms of data mining identify natural groupings in the data.

For example, a data mining model might identify the segment of the population that has the following properties in common:

- Has an income within a specified range
- Has a good driving record
- Leases a new car on a yearly basis

Creating Actionable Information

Data mining can derive actionable information from large volumes of data.

Examples:

- A town planner might use a data mining model to predict income based on demographics to develop a plan for low-income housing.
- A car leasing agency might use a model that identifies customer segments to design a promotion targeting high-value customers.

Data Mining Categories

Data mining functions fall into two categories:

- Supervised
 - Also known as *directed learning*
 - Attempts to explain the behavior of the target, or predict a value for a target, as a function of a set of independent input attributes
- Unsupervised
 - Also known as *non-directed learning*
 - Is used to find previously unknown patterns or relationships
 - No previously known result is used to guide the algorithm in building the model.



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Data mining functions fall into two categories: Supervised and Unsupervised.

Supervised data mining: Has the goal of predicting either a categorical outcome or a numeric value for an attribute called the *target* attribute

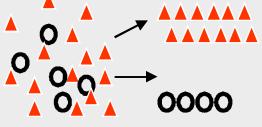
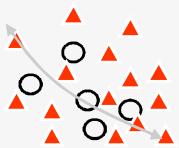
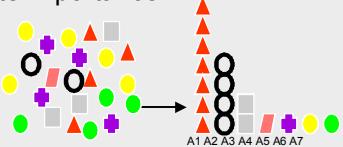
- Categorical outcome examples include:
 - Will a customer purchase product X? (Yes or No)
 - What level of revenue will a client generate over time? (Low, Medium, High)
- Numeric value examples include:
 - How much data will a smart phone customer use per month?
 - What is the size of the home loan that a prospective customer will open?

Unsupervised data mining: Has the goal of discovering relationships and patterns, rather than generating a prediction. That is, there is no target attribute.

Examples include:

- Determine distinct segments of a population and the attribute values indicating an individual's membership in a particular segment.
- Determine the five items most likely to be purchased at the same time as item X. This type of problem is usually called *market basket analysis*.
- Identify cases that differ from “normal” cases. This type of problem is also called *anomaly detection*.

Supervised Data Mining Techniques

| Problem Classification | Sample Problem |
|------------------------|--|
| Classification | <p>Based on demographic data about a set of customers, predict customer response to an affinity card program</p>  |
| Regression | <p>Based on demographic and purchasing data about a set of customers, predict how much a customer will spend</p>  |
| Attribute importance | <p>Based on customer response to an affinity card program, find the most significant predictors</p>  |

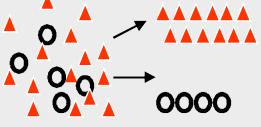
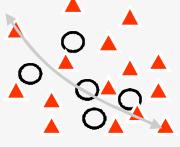
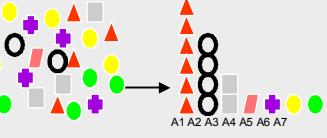


Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

There are three types of supervised modeling techniques supported by Oracle Data Mining.

- **Classification:** This technique is used to predict a categorical result for a target variable. There are two kinds:
 - **Binary classification:** The model predicts one of two target values for each case. Examples include the following:
 - Predict whether or not Product X is purchased. (Yes or No)
 - Predict whether a customer will default on a loan. (Yes or No)
 - Predict whether there will be a failure in the manufacturing process. (Yes or No)
 - **Multiclass classification:** The model predicts one of several target categories for each case. An example is predicting a level of revenue production (Low, Medium, or High).
- **Regression** (also called “continuous prediction”): The model predicts a specific target value. For example, you want to predict how much money a customer will spend. Using regression, the model predicts a specific target value for each case from among (possibly) infinitely many values.
- **Attribute importance:** This model is somewhat different than the other two (more typical) types of supervised learning techniques. It is used to find the most significant predictors of a target value. This type of model returns and ranks the attributes that are most important in predicting a target value.

Supervised Data Mining Algorithms

| Problem Classification | Algorithm | Applicability |
|---|--|--|
| Classification  | Logistic Regression (GLM) Decision Trees Naive Bayes Support Vector Machine (SVM) | Classical statistical technique Embedded app Wide/narrow data/text |
| Regression  | Multiple Regression (GLM) Support Vector Machine (SVM) | Classical statistical technique Wide/narrow data/text |
| Attribute importance  | Minimum Description Length (MDL) | Attribute reduction Identify useful data Reduce data noise |



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The chart shows a list of the algorithms that can be used to create models for particular supervised learning cases.

Unsupervised Data Mining Techniques

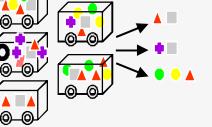
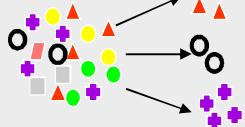
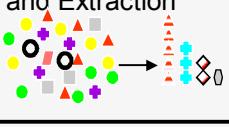
| Problem Classification | Sample Problem |
|------------------------|--|
| Anomaly detection | Based on demographic data about a set of customers, identify customer purchasing behavior that is significantly different from the norm. |
| Association rules | Find the items that tend to be purchased together and specify their relationship (market basket analysis). |
| Clustering | Segment demographic data into clusters and rank the probability that an individual will belong to a given cluster. |
| Feature extraction | Based on demographic data about a set of customers, group the attributes into general characteristics of the customers. |

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

There are four types of unsupervised modeling techniques supported by Oracle Data Mining.

- **Anomaly detection** uses one-class classification. In this approach, the model trains on data that is homogeneous. Then the model determines whether a new case is similar to the cases observed, or is somehow “abnormal” or “suspicious.” It detects abnormal cases but does not give reasons. Anomaly detection, although unsupervised, is typically used to predict whether a data point is typical among a set of cases.
- **Association rules** models are used for market basket analysis. Use this type of model to determine which cases are likely to be found together. For example, you can determine the five items most likely to be purchased at the same time as item X.
- **Clustering** models define segments, or “clusters,” of a population and then decide the likely cluster membership of each new case (although it is possible for an item to be in more than one cluster). These models use descriptive data mining techniques, but they can be applied to classify cases according to their cluster assignments. For example, you can determine distinct segments of a population and the attribute values indicating an individual’s membership in a particular segment.
- **Feature extraction** models create new attributes (features) by using combinations of the original attribute. For example, you can group the demographic attributes for a set of customers into general characteristics that describe the customer. These models can be used to produce results for personality tests or similar kinds of survey studies.

Unsupervised Data Mining Algorithms

| Problem Classification | Algorithm | Applicability |
|--|---|---|
| Anomaly Detection  | One Class SVM | Fraud detection |
| Association Rules  | Apriori | Market basket analysis |
| Clustering  | Enhanced K-Means Orthogonal Partitioning Clustering Expectation Maximization | Product grouping Hierarchical clustering Text mining Gene analysis |
| Feature Selection and Extraction  | Non-negative Matrix Factorization (NMF) Principal Component Analysis Singular Value Decomposition | Text analysis Feature reduction Survey results Pattern recognition |



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The chart shows a list of the algorithms that can be used to create models for particular unsupervised learning cases.

Oracle Data Mining: Overview

Oracle Data Mining (ODM) provides:

- Powerful in-database data mining algorithms
- SQL APIs to build applications to automate knowledge discovery
- Oracle Data Miner GUI



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Oracle Data Mining (ODM) embeds the data mining technology that was just discussed within Oracle Database. ODM algorithms operate natively on relational tables, relational views, or external tables that point non-relational data. This architecture eliminates the need to extract and transfer data into stand-alone tools or specialized analytic servers.

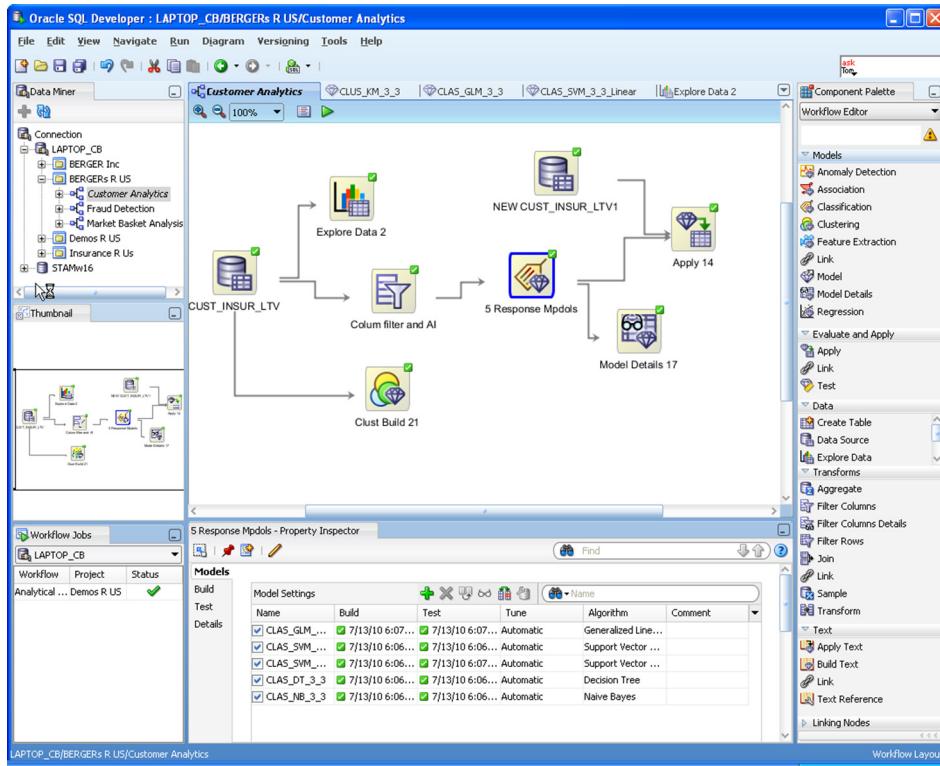
ODM's integrated architecture results in a simpler, more reliable, and more efficient data management and analysis environment.

- Data mining tasks can run asynchronously and independently of any specific user interface as part of standard database processing pipelines and applications.
- Data analysts can mine the data, build models and methodologies, and then turn those results and methodologies into full-fledged application components. The benefits of integration with Oracle Database are significant when it comes to deploying models and scoring data in a production environment.

ODM provides two interfaces to the data mining technology:

- SQL APIs that may be embedded within applications to automate knowledge discovery
- The Data Miner GUI, which enables a visual, interactive environment in building data mining models. The Data Miner also automatically generates the SQL code that is required to run the models.

Oracle Data Miner GUI



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The free Oracle Data Miner GUI is an extension to Oracle SQL Developer that enables data analysts to:

- Work directly with data inside the database, or data referenced by external tables
- Explore data graphically
- Build and evaluate multiple data mining models
- Apply Oracle Data Mining models to new data
- Deploy Oracle Data Mining predictions and insights throughout the enterprise

Oracle Data Miner workflows capture and document a user's analytical methodology. The workflows can be saved and shared with others to automate advanced analytical methodologies.

ODM SQL Interface

```

drop table CLAIMS_SET;
exec dbms_data_mining.drop_model('CLAIMSMODEL');
create table CLAIMS_SET (setting_name varchar2(30), setting_value varchar2(4000));
insert into CLAIMS_SET values ('ALGO_NAME','ALGO_SUPPORT_VECTOR_MACHINES');
insert into CLAIMS_SET values ('PREP_AUTO','ON');
commit;

begin
dbms_data_mining.create_model('CLAIMSMODEL', 'CLASSIFICATION',
'CLAIMS', 'POLICYNUMBER', null, 'CLAIMS_SET');
end;
/

-- Top 5 most suspicious fraud policy holder claims
select * from
(select POLICYNUMBER, round(prob_fraud*100,2) percent_fraud,
rank() over (order by prob_fraud desc) rnk from
(select POLICYNUMBER, prediction_probability(CLAIMSMODEL, '0' using *) prob_fraud
from CLAIMS
where PASTNUMBEROFCLAIMS in ('2to4', 'morethan4'))
where rnk <= 5
order by percent_fraud desc;

```

| POLICYNUMBER | PERCENT_FRAUD | RNK |
|--------------|---------------|-----|
| 6532 | 64.78 | 1 |
| 2749 | 64.17 | 2 |
| 3440 | 63.22 | 3 |
| 654 | 63.1 | 4 |
| 12650 | 62.36 | 5 |

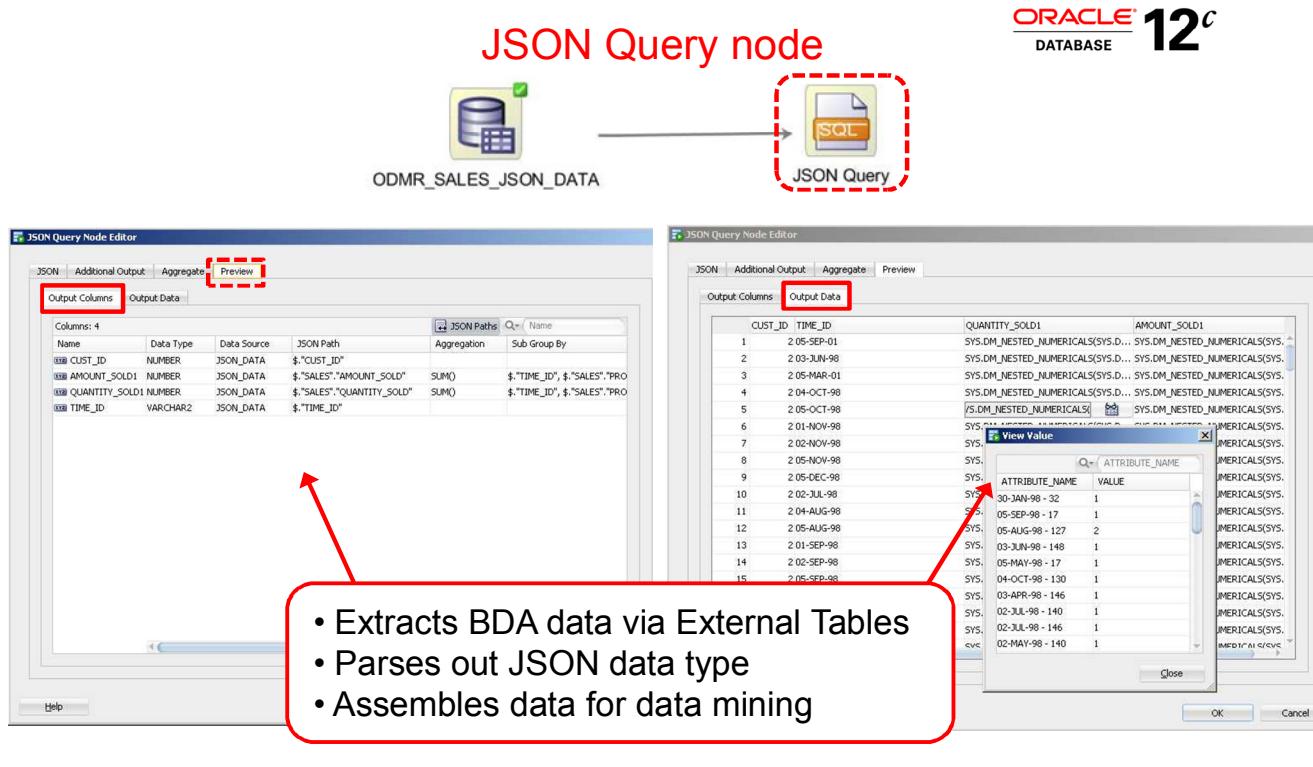
Automated Monthly "Application"! Just add:
Create
View CLAIMS2_30
As
Select * from CLAIMS2
Where mydate > SYSDATE - 30



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

As mentioned previously, the Oracle Data Miner generates SQL code for all workflow nodes, which you can access from the GUI. This feature enables you to embed Oracle Data Mining models in SQL queries and database applications.

Oracle Data Miner 4.1 Big Data Enhancement



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

JSON is a popular lightweight data structure used by Big Data. As you have seen in this course, web logs generated in the middle tier web servers are likely in JSON format. In addition, NoSQL database vendors have chosen JSON as their primary data representation. Moreover, the JSON format is widely used in the RESTful style web services responses generated by most popular social media websites like Facebook, Twitter, LinkedIn, and so on.

Oracle Database 12.1.0.2 provides the ability to store and query JSON data. To take advantage of the database JSON support, the upcoming Data Miner 4.1 (SQL Developer 4.1) added a new JSON Query node that allows data mining users to query JSON data as relational format.

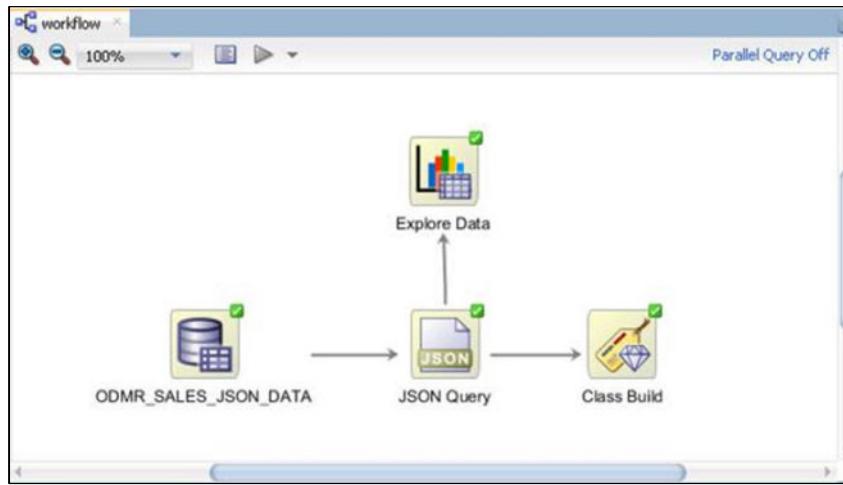
The slide shows a JSON Query node that is connected to a JSON Data Source node. By using the JSON Query Node Editor, you can view a wide range of information about the JSON data and specify options for querying the data. For example:

- The JSON tab displays attribute information about the JSON data.
- The Preview tab, shown in the slide, enables you to see both output column information (shown on the left) and output data (shown on the right).

The JSON Query node extracts BDA data via External Tables, automatically parsing out the JSON data type and assembling the data for data mining.

Note: The hands-on environment for this course uses SQL Developer 4.0.3, which does not yet include the JSON Node support.

Example Workflow Using JSON Query Node



For information on importing JSON data in a Data Miner workflow, see:

https://blogs.oracle.com/datamining/entry/how_to_import_json_data

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Here is an example of how a JSON Query node is used to project the JSON data source to relational format, so that the data can be consumed by an Explore Data node for data analysis, and by a Classification Build node for model building purposes.

To see how to construct this workflow, which imports JSON data to Oracle Data Miner for mining purposes, see the blog at:

https://blogs.oracle.com/datamining/entry/how_to_import_json_data

ODM Resources

Oracle University:

- *Oracle Data Mining Techniques* (two-day course)

Oracle Learning Library:

- Oracle By Example – A variety of hands-on tutorials
- Recorded demonstrations
- Oracle Big Data landing page
https://apex.oracle.com/pls/apex/f?p=44785:141:0:::141:P141_PAGE_ID,P141_SECTION_ID:27,615

Oracle Technology Network:

- Oracle Advanced Analytics
<http://www.oracle.com/us/products/database/options/advanced-analytics/overview/index.html>
- Oracle Data Mining
<http://www.oracle.com/technetwork/database/options/advanced-analytics/odm/overview/index.html?ssSourceSiteId=ocomen>



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Practice: Overview (ODM)

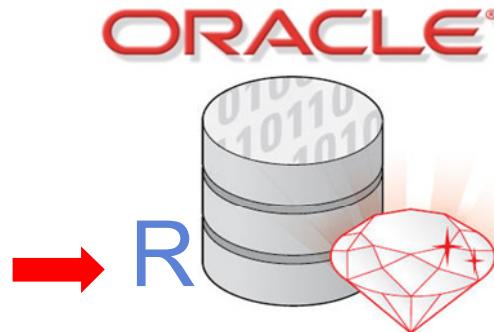
The practices for this lesson cover the use of ODM and ORE with Oracle Big Data Lite VM 4.0.1 source data.

- Practice 23-1: Using Oracle Data Miner 4.0 with Big Data Lite
- Practice 23-2: Using Oracle R Enterprise 1.4 with Big Data Lite



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

OAA: Oracle R Enterprise



ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Now, you learn about Oracle R Enterprise.

What Is R?

- R is an open-source statistical programming language and environment that provides:
 - An object-oriented programming language
 - A powerful graphical environment for visualization
 - Out-of-the-box statistical techniques
- R was created in 1994 as an alternative to SAS, SPSS, and other statistical environments.
- Its widespread use, breadth of functionality, and quality of implementation have enabled R to establish itself as a new statistical software standard.

<http://www.r-project.org/>



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

To begin, you should first understand the technology that Oracle R Enterprise is based on.

R is an open-source statistical programming language and environment that supports:

- Statistical computing and data visualization
- Data manipulations and transformations
- Sophisticated graphical displays

R was created as an alternative to statistical environments similar to SAS and SPSS.

Its functionality is comprised in a collection of R packages. An R package is a set of related functions, help files, and data files. There are currently more than 3,340 available packages.

For more information about R, see the “R Project for Statistical Computing” website at <http://www.r-project.org>.

Who Uses R?

Around 2 million R users worldwide:

- Widely taught in universities
- Many corporate analysts and data scientists

Thousands of open source packages to enhance productivity such as:

- Bioinformatics with R
- Spatial statistics with R
- Financial market analysis with R
- Linear and nonlinear modeling

The screenshot shows the CRAN Task Views website. On the left, there's a sidebar with links like 'CRAN Mirrors', 'What's new?', 'Task Views', and 'Search'. Below that are sections for 'About R', 'R Homepage', 'The R Journal', 'Software' (with links for 'R Sources', 'R Binaries', 'Packages', and 'Other'), and 'Documentation' (with links for 'Manuals', 'FAQs', and 'Contributed'). A red box highlights the top navigation links. To the right is a large list of 'CRAN Task Views' grouped by category. A red bracket groups the sidebar and the main content area.

| CRAN Task Views | |
|---|---|
| Bayesian | Bayesian Inference |
| ChemPhys | Chemometrics and Computational Physics |
| ClinicalTrials | Clinical Trial Design, Monitoring, and Analysis |
| Cluster | Cluster Analysis & Finite Mixture Models |
| Distributions | Probability Distributions |
| Econometrics | Computational Econometrics |
| Environmetrics | Analysis of Ecological and Environmental Data |
| ExperimentalDesign | Design of Experiments (DoE) & Analysis of Expt |
| Finance | Empirical Finance |
| Genetics | Statistical Genetics |
| Graphics | Graphic Displays & Dynamic Graphics & Graphi |
| gR | gRaphical Models in R |
| HighPerformanceComputing | High-Performance and Parallel Computing with R |
| MachineLearning | Machine Learning & Statistical Learning |
| MedicalImaging | Medical Image Analysis |
| Multivariate | Multivariate Statistics |
| NaturalLanguageProcessing | Natural Language Processing |
| OfficialStatistics | Official Statistics & Survey Methodology |
| Optimization | Optimization and Mathematical Programming |
| Pharmacokinetics | Analysis of Pharmacokinetic Data |
| Phylogenetics | Phylogenetics, Especially Comparative Methods |
| Psychometrics | Psychometric Models and Methods |
| ReproducibleResearch | Reproducible Research |
| Robust | Robust Statistical Methods |
| SocialSciences | Statistics for the Social Sciences |
| Spatial | Analysis of Spatial Data |
| Survival | Survival Analysis |
| TimeSeries | Time Series Analysis |

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

With over 2 million R users worldwide, R is increasingly being used as a statistical tool in the academic world. Many colleges and universities worldwide are using R today in their statistics classes. In addition, more and more corporate analysts are using R.

R benefits from around 5000 open-source packages, which can be thought of as a collection of related functions. This number grows continuously with new package submissions from the R user community.

Each package provides specialized functionality in such areas as bioinformatics and financial market analysis.

In the slide, the list on the right shows “CRAN Task Views.” CRAN stands for the Comprehensive R Archive Network, which is a network of FTP and web servers that store identical, up-to-date versions of R code and documentation.

The CRAN Task Views list areas of concentration for a set of packages. Each link contains information that is available on a wide range of topics.

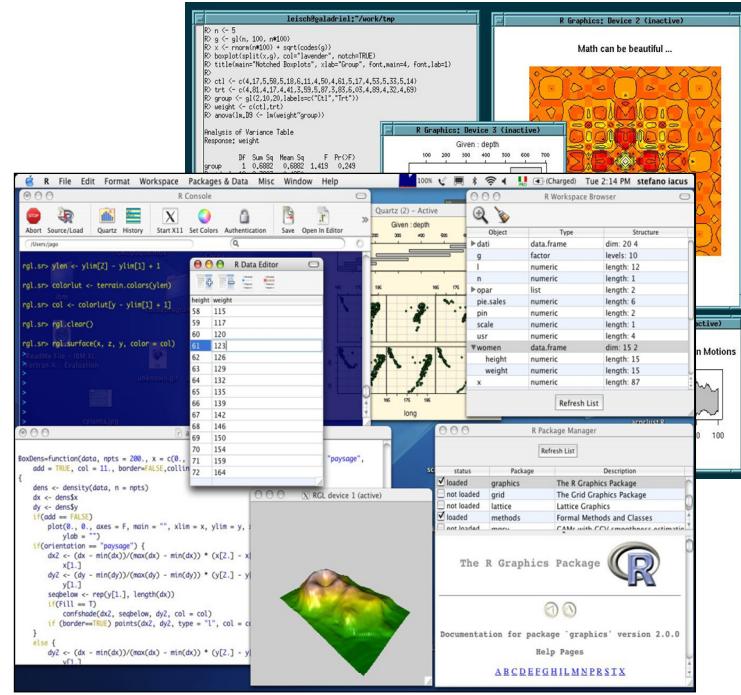
Why Do Statisticians, Data Analysts, Data Scientists Use R?

R is a statistics language that is similar to Base SAS or SPSS.

R environment:

- Powerful
- Extensible
- Graphical
- Extensive statistics
- OOTB functionality with many “knobs” but smart defaults
- Ease of installation and use
- **Free**

<http://cran.r-project.org/>



ORACLE

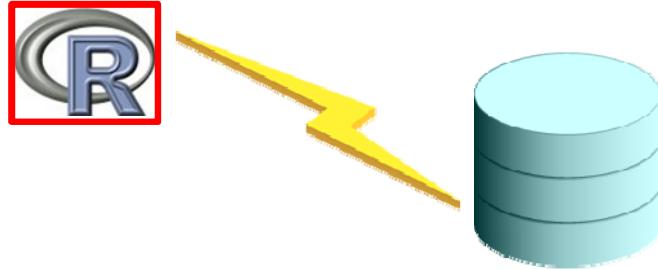
Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

So, why do statisticians, data analysts, and data scientists use R?

- As mentioned previously, R is a statistics language that is similar to SAS or SPSS.
- R is a powerful and extensible environment, with a wide range of statistics and data visualization capabilities.
 - Powerful: Users can perform data analysis and visualization with a minimal amount of R code.
 - Extensible: Users can write their own R functions and packages that can be used locally, shared within their organizations, or shared with the broader R community through CRAN.
- It is easy to install and use.
- It is free and downloadable from the R Project website.

Limitations of R

- The key disadvantage of R is that it is restricted to data sizes that fit in main memory.
 - R is a client and server bundled together as one executable, similar to Excel.
 - R's "call by value" semantics means that, as data flows into functions, a copy is made upon modifying the object.
- Oracle R Enterprise has overcome this limitation by enabling R to interact with Oracle Database and to exchange data.



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Although it is a powerful and effective statistical environment, R has limitations.

First, R was conceived as a single-user tool that was by default not multithreaded. The client and server components are bundled together as a single executable, much like Excel.

- R is limited by the memory and processing power of the machine on which it runs.
- Also, R cannot automatically leverage the CPU capacity on a user's multiprocessor laptop without special packages and programming.

Second, R suffers from another scalability limitation that is associated with RAM.

- R requires data that it operates on to be first loaded into memory.
- In addition, R's approach to passing data between function invocations results in data duplication. This "call by value" approach to parameter passing can use up memory quickly.

So inherently, R is really not designed for use with big data.

Some users have provided packages to overcome some of the memory limitations, but the users must explicitly program with these packages.

Oracle R Enterprise has overcome these limitations by enabling R programmers to leverage the power of the Oracle Database.

Oracle's Strategy for the R Community

Oracle supports R as the de facto standard statistical programming language in the following ways:

- For interaction with the database
- To enable execution of R scripts in the database, leveraging it as a high performance computing platform
- For production deployment of R scripts through SQL
- For ease of incorporating structured and image results in applications
- To provide additional native high performance analytics that execute in the database



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Oracle recognized the need to support data analysts, statisticians, and data scientists with a widely used and rapidly growing statistical programming language. Oracle chose R - recognizing it as the new de facto standard for computational statistics and advanced analytics.

Oracle supports R in the following ways:

- R as the language of interaction with the database
- R as the language in which analytics can be written and executed in the database
 - R scripts may leverage Oracle Database as a high performance computing platform
 - R scripts may be deployed through SQL for production applications
 - R script structured and image results may be incorporated in applications
- R as the language in which several native high performance analytics have been written that execute in database

Additionally, of course, you may choose to leverage any of the CRAN algorithms to execute R scripts at the database server leveraging several forms of data parallelism.

Oracle R Enterprise

Oracle R Enterprise (ORE) brings R's statistical functionality to Oracle Database by:

- Eliminating R's memory constraint by enabling R to work directly and transparently on database objects and data outside of Oracle Database
- Enabling R to run on very large data sets
- Providing enterprise production infrastructure
- Exploiting database parallelism without requiring R programming
- Enabling immediate deployment of models into production systems



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

ORE provides the familiar R environment for R programmers, while enabling them to leverage the power and scalability, whether they operate on data inside or outside of the Database. To do this, ORE overloads base R functions for data manipulation in Oracle Database, effectively eliminating R's memory constraints. No SQL knowledge is required to use ORE.

In addition, ORE provides a database-controlled, data-parallel execution framework, thereby removing the need to manage data outside the database.

ORE leverages the latest R algorithms and packages. You can use open source R, or Oracle's R Distribution with ORE. When you use Oracle R Distribution, R is an embedded component of the DBMS server.

ORE provides the following benefits:

- Scalability for Big Data analysis
- Data security
- Traceability
- Fewer moving parts
- Shorter information latency
- Reduced process complexity

ORE: Software Features

ORE consists of the following primary features:

- Transparency Layer
 - Set of packages mapping R data types to Oracle Database objects
 - Transparent SQL generation for R expressions on mapped data types
 - Enables direct interaction with database-resident data by using R
- In-Database Predictive Analytics
 - Sets of statistical functions, procedures, and algorithms that support a wide variety of predictive analytic requirements
 - Executes in Oracle Database
- Embedded R Execution with an R API and a SQL API
 - Enables database server execution of R code to facilitate embedding R in operational systems
 - Enables integration of R results (both structured and image) in database applications
 - Enables data flow parallelism, generation of rich XML output, SQL access to R, and parallel simulations capability



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

You can divide the primary features of ORE into three main categories: the Transparency Layer, In-Database Predictive Analytics, and Embedded R Execution.

The Transparency Layer is a set of packages that map R data types to Oracle Database objects.

- This feature automatically generates SQL for R expressions on mapped data types, enabling direct interaction with data in Oracle Database while using R language constructs.
- Functionally, this mapping provides access to database tables from R as a type of `data.frame`: a base R data representation with rows and columns. ORE calls this an “`ore.frame`.”
- Therefore, when you invoke an R function on an `ore.frame`, the R operation is sent to the database for execution as SQL.

The in-database predictive analytics feature provides algorithms and functions that support a variety of statistical computations.

The Embedded R execution feature, supported by both an R and a SQL interface, enables in-database embedded R execution. This feature is particularly valuable for third-party R packages, or custom functions, that do not have equivalent in-database functionality.

ORE Packages

| Package | Description |
|-------------|---|
| ORE | Top-level package for Oracle R Enterprise |
| OREbase | Corresponds to the R base package |
| OREstats | Corresponds to the R stat package |
| OREgraphics | Corresponds to the R graphics package |
| OREcommon | Common low-level functionality |
| OREeda | ORE exploratory data analysis package that contains SAS PROC-equivalent functionality |
| OREembed | Provides Embedded R Execution functionality |
| OREdm | Exposes Oracle Data Mining algorithms |
| OREmodels | ORE-provided advanced analytics algorithms |
| OREpredict | Enables scoring data in Oracle DB using R models |
| ORExml | Supports XML translation between R and Oracle Database |



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

This slide contains the list of packages that comprise Oracle R Enterprise:

- The ORE package is a top-level package that is used for the installation of the other packages.
- The Transparency Layer that was just discussed consists of the next three packages: OREbase, OREstats, and OREgraphics.
 - These packages correspond to the R base, statistics, and graphics packages, respectively.
 - In addition, these packages include overloaded functions that enable R users to access data from Oracle Database, push R data to Oracle Database, and leverage the power and scalability of Oracle Database for processing.
- The OREcommon and ORExml packages handle low-level, internal functionality between R and Oracle Database.
- The OREembed package facilitates embedding R scripts to run inside Oracle Database.
- The remaining packages enable R access to Oracle In-Database predictive analytics functionality as specified in the table.

Functions for Interacting with Oracle Database

| Function | Description |
|-------------|--|
| ore.connect | Connects to a specific schema and database Only one connection is active at a time. |
| ore.create | Creates a database table from data.frame or ore.frame |
| ore.drop | Drops a table or view in database |
| ore.push | Stores R object in database as temporary object; returns handle to object. Data frame, matrix, and vector to table; list/model/others to serialized object |
| ore.sync | Synchronizes ORE proxy objects in R with tables/views available in database, on a per-schema basis |
| ore.exists | Returns TRUE if named table or view exists in schema |

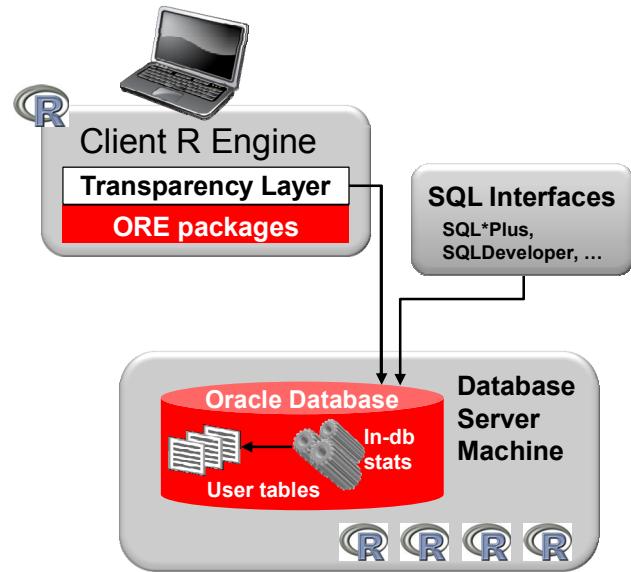


Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Note: As mentioned previously, a `data.frame` is an R object that conceptually corresponds to an Oracle table or view. By using ORE, data in an Oracle table is made available to an R session as an `ore.frame`. An `ore.frame` object is the equivalent to an R `data.frame` object.

ORE: Target Environment

- Eliminate memory constraint of client R engine.
- Execute R scripts at database server machine.
- Get maximum value from your Oracle Database.
- Enable integration and management through SQL.
- Get even better performance with Exadata.



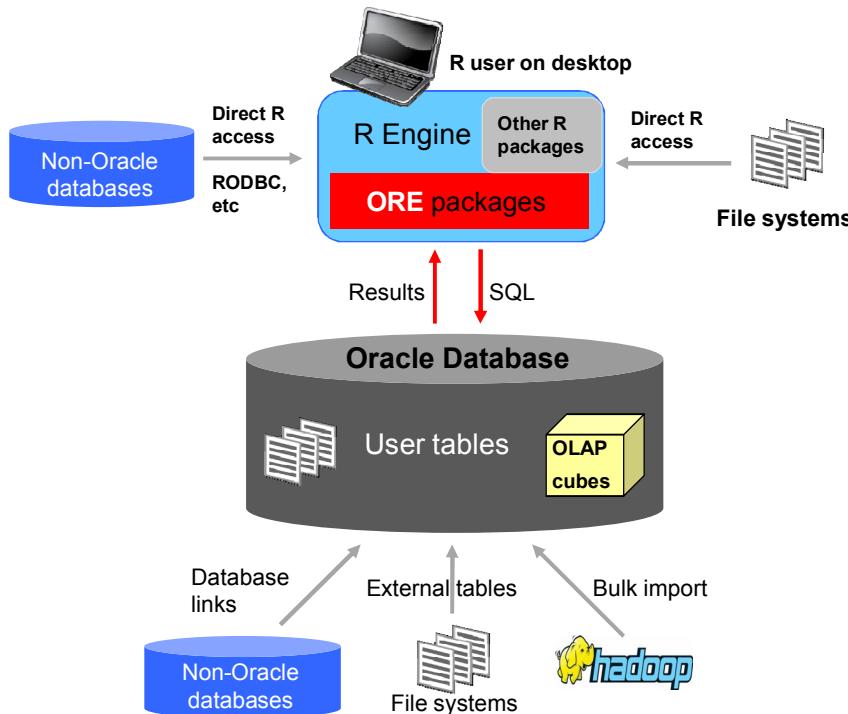
ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The ORE target environment design provides a comprehensive, database-centric environment for end-to-end analytic processes in R, with immediate deployment to production environments. It provides many benefits, including:

- Elimination of R client engine memory constraint
- Execution of R scripts through the Oracle Database server machine for scalability and performance
- Seamless integration of Oracle Database as the HPC environment for R scripts, providing data parallelism and resource management
- The ability to operationalize entire R scripts in production applications
- Scoring of R models in Oracle Database

ORE: Data Sources



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

R and ORE can receive data from many sources. In this figure, the R engine running on the user's laptop.

Through a series of R packages, R itself is able to access data stored in both files, and in databases.

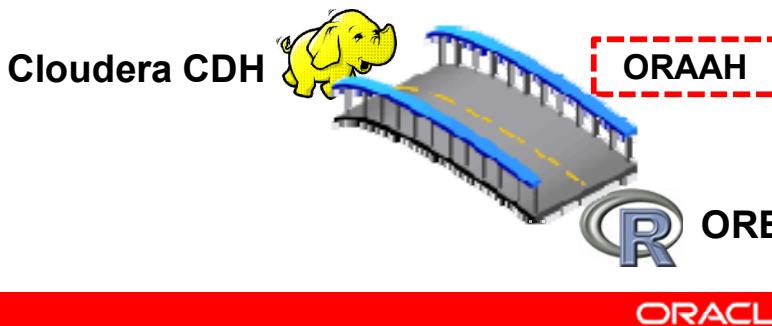
In addition, ORE provides transparent access to data stored in Oracle Database, as discussed previously.

Also, ORE has access to:

- Data in other databases, which are accessible through database links
- Data in external tables
- Of course, data in HDFS. In addition to bulk import, ORE makes it possible to access Hadoop directly, in a similar fashion to external tables, by using the HDFS connect. This means that you can join Hadoop data with database data.

ORE and Hadoop

- Oracle R Advanced Analytics for Hadoop (ORAAH) is set of packages that provides an interface between the local R environment, Oracle Database, and CDH.
- Using simple R functions, you can:
 - Sample data in HDFS
 - Copy data between Oracle Database and HDFS
 - Schedule R programs to execute as MapReduce jobs
 - Return the results to Oracle Database or to your laptop



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

ORAAH is a set of packages that provides an interface between the local R environment, Oracle Database, and Cloudera Distribution for Hadoop.

Note: While ORAAH includes some of the packages from ORE, it is not an adaptation of ORE for Hadoop.

ORAAH provides an interface to the Hadoop Cluster on Oracle Big Data Appliance, and enables the following:

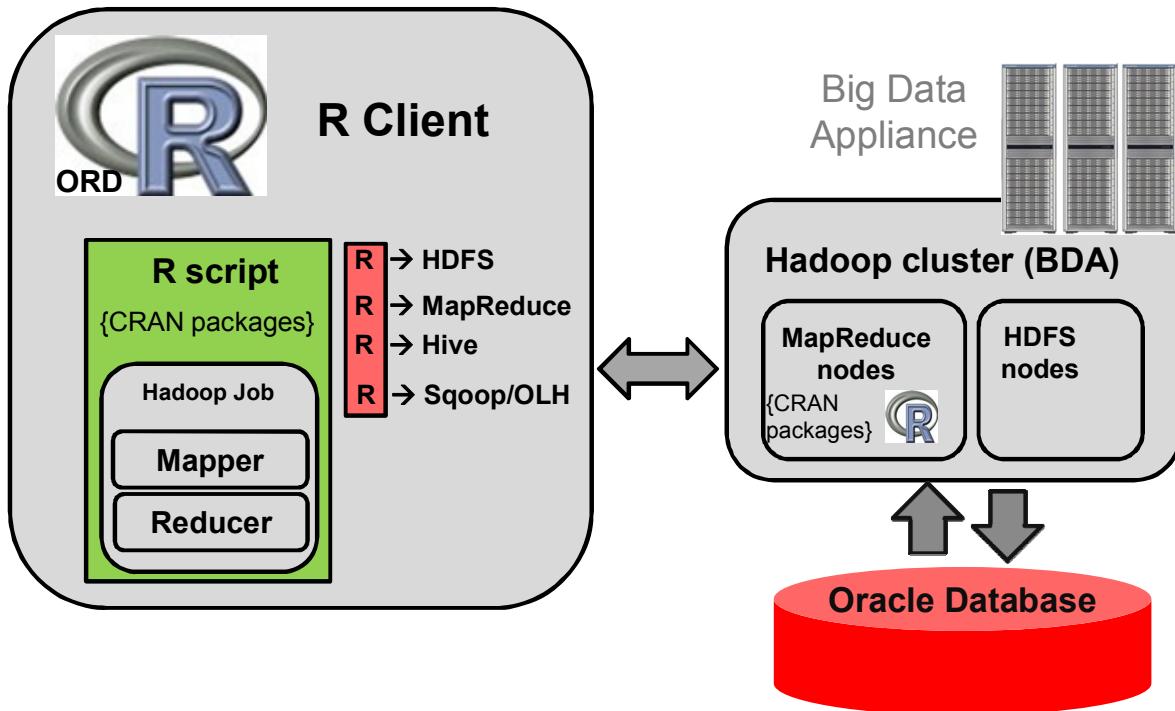
- Transparent access to the Hadoop Cluster
- MapReduce programming with R
- Data manipulation in HDFS, the database, and the file system, from R

ORAAH is used to process data from low density to higher density. The higher-density data can then be loaded into Oracle Database for further analysis by using ORE. ORAAH also facilitates interactive access to the Hadoop infrastructure from an R user's desktop as part of exploratory analysis.

Notes:

- ORAAH is one of the Oracle Big Data Connectors. It is a separate product from ORE.
- ORAAH is available only in Oracle Big Data Appliance.

ORAAH: Architecture



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

ORACLE

The architecture diagram in the slide depicts Oracle R Advanced Analytics for Hadoop as an interface between the local R environment, Oracle Database, and Cloudera CDH.

This architecture enables:

- Expanded user population that can build models on Hadoop
- Accelerated rate at which business problems are tackled
- Analytics that scale with data volumes, variables, techniques
- Transparent access to Hadoop Cluster
- The ability to:
 - Manipulate data in HDFS, database, and the file system—all from R
 - Write and execute MapReduce jobs with R
 - Leverage CRAN R packages to work on HDFS—resident data
 - Move from lab to production without requiring knowledge of Hadoop internals, Hadoop CLI, or IT infrastructure

ORAAH Package

ORAAH provides API access from a local R client to Hadoop by using the following APIs:

- `hdfs` provides an interface to HDFS.
 - `hdfs.attach`, `hdfs.download`, `hdfs.exists`,
`hdfs.get`, `hdfs.ls`, and so on
- `orch` provides an interface between the local R instance and Oracle Database for connection and algorithms
 - `orch.connect`, `orch.disconnect`, `orch.reconnect`,
`orch.which`
 - `orch.cov`, `orch.kmeans`, `orch.lm`, `orch.lmf`,
`orch.neural`, `orch.nmf`, `orch.princomp`,
`orch.sample`
- `hadoop` provides an interface to Hadoop MapReduce.
 - `hadoop.exec`, `hadoop.run`



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Refer to the *Oracle Big Data Connectors User's Guide* for a detailed description of each function.

HDFS Connectivity and Interaction

| Function | Description |
|---------------|---|
| hdfs.push | Copies data from Oracle Database to HDFS |
| hdfs.pull | Copies data from HDFS to Oracle Database |
| hdfs.upload | Copies a file from the local file system to HDFS |
| hdfs.download | Copies a file from HDFS to the local file system |
| hdfs.attach | Attaches an unstructured data file in HDFS to the ORAAH framework |
| hdfs.rm | Removes a file from HDFS |
| hdfs.get | Copies data from HDFS to a data.frame object in the local R environment |
| hdfs.put | Copies data from an R data.frame object to HDFS |
| hdfs.sample | Copies a sample of data from a Hadoop file to an R in-memory object |



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

ORAAH includes APIs that enable users to access files on HDFS and perform R calculations on the data residing in those files. You also have the option of uploading the data from the storage area, running an R script, and downloading the results on your laptop.

ORAAH Functions for HDFS Interaction

Explore files in HDFS:

- `hdfs.cd()`, `hdfs.ls()`, `hdfs.getcwd()`, `hdfs.mkdir()`
- `hdfs.mv()`, `hdfs.cp()`, `hdfs.size()`, `hdfs.sample()`

Interact with HDFS content in the ORAAH environment:

- Discover or hand-create metadata.
- Work with in-memory R objects: `hdfs.get()`, `hdfs.put()`
- Work with database objects: `hdfs.push()`, `hdfs.pull()`
- Work with local files: `hdfs.upload()`, `hdfs.download()`

Obtain ORAAH metadata descriptors:

- Discover metadata from CSV files: `hdfs.attach()`,
`hdfs.describe()`



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

In addition, ORAAH contains a variety of functions that enable an R user to interact with an HDFS system. You can:

- Explore files in HDFS, including the following activities: change directories, list files in a directory, show the current directory, create new directories, move files, copy files, determine the size of files, and sample files
- Interact with HDFS content in the ORAAH environment, including the following: discovery (or creation) of metadata and easy access to in-memory R objects, database objects, and local files
- Discover metadata from CSV files

ORAAH Functions for Predictive Algorithms

| Function | Description |
|---------------|--|
| orch.cor | Correlation matrix computation |
| orch.cov * | Performs k-means clustering on a data matrix stored as an HDFS file |
| orch.kmeans | Copies a file from the local file system to HDFS |
| orch.lm * | Fits a linear model by using TSQR factorization and parallel distribution. Computes same statistical parameters as ore.lm function. |
| orch.lmf | Fits a low rank matrix factorization model by using either the jellyfish algorithm or the Mahout ALS-WR algorithm |
| orch.neural * | Provides a neural network to model complex, nonlinear relationships between inputs and outputs, or to find patterns in the data |
| orch.nmf | Provides the main entry point to create a non-negative matrix factorization model using the jellyfish algorithm. Can work on much larger data sets than the R NMF package, because the input does not need to fit into memory. |
| orch.princomp | Principal components analysis of HDFS data |
| orch.sample | Sample HDFS data by percentage or explicit number of rows specification |

*Score data using `orch.predict`



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

ORAAH also includes functions that expose statistical algorithms through the `orch` API. These predictive algorithms are available for execution on a Hadoop Cluster.

Hadoop Connectivity and Interaction

| Function | Description |
|---------------|--|
| hadoop . exec | Starts the Hadoop engine and sends the mapper, reducer, and combiner R functions for execution. You must first load the data into HDFS. |
| hadoop . run | Starts the Hadoop engine and sends the mapper, reducer, and combiner R functions for execution. If the data is not already stored in HDFS, hadoop . run first copies the data there. |



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Finally, ORAAH contains two primary functions that provide an interface to Hadoop MapReduce.

Word Count: Example Without ORAAH

```

import java.io.IOException;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapred.MapReduceBase;
import org.apache.hadoop.mapred.Mapper;
import org.apache.hadoop.mapred.OutputCollector;
import org.apache.hadoop.mapred.Reporter;
public class WordMapper extends MapReduceBase
    implements Mapper<LongWritable, Text, Text,
    IntWritable> {
    public void map(LongWritable key, Text value,
        OutputCollector<Text, IntWritable>
    output, Reporter reporter)
        throws IOException {
        String s = value.toString();
        for (String word : s.split("\\W+")) {
            if (word.length() > 0) {
                output.collect(new Text(word), new
    IntWritable(1));
            }
        }
    }
}
Mapper

```

```

import java.io.IOException;
import java.util.Iterator;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapred.OutputCollector;
import org.apache.hadoop.mapred.MapReduceBase;
import org.apache.hadoop.mapred.Reducer;
import org.apache.hadoop.mapred.Reporter;
public class SumReducer extends MapReduceBase
implements
    Reducer<Text, IntWritable, Text,
    IntWritable> {
    public void reduce(Text key,
    Iterator<IntWritable> values,
        OutputCollector<Text, IntWritable>
    output, Reporter reporter)
        throws IOException {
        int wordCount = 0;
        while (values.hasNext()) {
            IntWritable value = values.next();
            wordCount += value.get();
        }
        output.collect(key, new
    IntWritable(wordCount));
    }
}
Reducer

```

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Without Oracle R Advanced Analytics for Hadoop, Java skills are needed to write MapReduce programs to access Hadoop data.

For example, the mapper and reducer programs shown here are needed to perform a simple word count task on text data that is stored in Hadoop.

Word Count: Example with ORAAH

```

input <- hdfs.put(corpus)
wordcount <- function (input, output = NULL, pattern = " ") {
  res <- hadoop.exec(dfs.id = input,
    mapper = function(k,v) {
      lapply( strsplit(x = v, split = pattern)[[1]],
        function(w) orch.keyval(w,1)[[1]])
    },
    reducer = function(k,vv) {
      orch.keyval(k, sum(unlist(vv)))
    },
    config = new("mapred.config",
      job.name      = "wordcount",
      map.output    = data.frame(key=0, val=''),
      reduce.output = data.frame(key='', val=0) )
  )
  res
}

```



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

With ORAAH, the same word-count task is performed with a simple R script. ORAAH functions greatly simplify access to Hadoop data by leveraging the familiar R interface.

In this example, the R script uses several common ORAAH functions to perform the task, including `hdfs.put()`, `hadoop.exec()`, and `orch.keyval()`.

The R script:

- Loads the R data into HDFS and creates a function named `wordcount` by using the input data
- Specifies and invokes the MapReduce job with the `hadoop.exec()` function
- Splits words and outputs each word in the `mapper` step
- Sums the count of each word in the `reducer` step
- Specifies the job configuration and returns the MapReduce output as the result

With ORAAH, mapper and reducer functions can be written in R, greatly simplifying access to Hadoop data.

ORE Resources

Oracle University: *Oracle R Enterprise Essentials* (2-day course)

Book: [Using R to Unlock the Value of Big Data](#), by Mark Hornick and Tom Plunkett

Blog: <https://blogs.oracle.com/R/>

Forum: <https://forums.oracle.com/forums/forum.jspa?forumID=1397>

Products: <http://oracle.com/goto/R>

- Oracle R Enterprise (ORE)
- Oracle R Distribution (Oracle's redistribution of open source R)
- ROracle (Handles connections between R and Oracle Database)
- Oracle R Advanced Analytics for Hadoop (ORAAH)

Oracle Technology Network:

<http://www.oracle.com/technetwork/database/database-technologies/r/r-enterprise/overview/index.html>



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Practice: Overview (ORE)

The practices for this lesson cover the use of ODM and ORE with Oracle Big Data Lite VM 4.0.1 source data.

- Practice 23-1: Using Oracle Data Miner 4.0 with Big Data Lite
- Practice 23-2: Using Oracle R Enterprise 1.4 with Big Data Lite



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Summary

In this lesson, you should have learned how to:

- Define Oracle Advanced Analytics
- Describe the uses and benefits of Oracle Data Mining
- Describe the uses and benefits of Oracle R Enterprise



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

24

Introducing Oracle Big Data Discovery

ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Course Road Map

Module 1: Big Data Management System

Module 2: Data Acquisition and Storage

Module 3: Data Access and Processing

Module 4: Data Unification and Analysis

Module 5: Using and Managing Oracle Big Data Appliance

Lesson 16: Options for Integrating Your Big Data

Lesson 17: Overview of Apache Sqoop

Lesson 18: Using Oracle Loader for Hadoop (OLH)

Lesson 19: Using Copy to BDA

Lesson 20: Using Oracle SQL Connector for HDFS

Lesson 21: Using ODI and OGG with Hadoop

Lesson 22: Using Oracle Big Data SQL

Lesson 23: Using Advanced Analytics:
Oracle Data Mining and Oracle R Enterprise

Lesson 24: Introducing Oracle Big Data Discovery



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

In this final lesson of Module 4, you are provided an introduction to a new Oracle application for big data: Big Data Discovery.

Big Data Discovery was not available for general release to manufacturing (RTM) at the time this course was produced. Therefore, we refer you to supporting resources at the end of this lesson.

Objectives

After completing this lesson, you should be able to describe:

- The features of Oracle Big Data Discovery
- The benefits of Oracle Big Data Discovery



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Oracle Big Data Discovery

A visual product that accesses all data in Hadoop, enabling organizations to:

- Find and explore data 
- Transform data 
- Discover information 
- Share insights 



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

ORACLE

Oracle Big Data Discovery (BDD) is a set of visual analytic capabilities, built natively on Hadoop to transform raw data into business insight.

BBD provides rapid visual access to all the data in Hadoop, enabling you to:

Find and Explore Data:

- Find relevant data quickly, through a rich interactive catalog of the raw data in Hadoop.
- Explore the data through familiar search and guided navigation.

Transform Data: Transformation and enrichment capabilities are provided natively within the product, in an intuitive and interactive visual interface, before sending the data upstream.

Discover Information:

- Ask questions of the data and get answers using familiar search and guided navigation.
- Drag to create interactive visualizations and discovery dashboards.

Share Insights:

- Save and share projects, bookmarks, and analytic snapshots with others in the organization.
- Publish blended data to HDFS for leverage in other tools.

Note: Big Data Discovery will be available as a licensed option on Big Data Appliance, or on commodity hardware.

Find Data



ORACLE® Big Data Discovery

Refine By

- USAGE
 - Created By Me
- CONTENT
 - Contains Dates
 - Filter disabled
 - true
 - false
 - Contains Locations
- METADATA
 - Project Author
 - Data Set Author
 - Project Tags
 - Data Set Tags
 - Last Modified
 - Number of Records
 - Number of Attributes
 - Between
 - 2 - 231

11 Projects 202 Data Sets + Data Set

Zagat Rating for Top 1000 Restaurants in Boston (1000 records)

Actions: Explore, Add to project, Edit Tags, Delete

Summary: 2 Views, Last Updated: 1/6/15 11:26:52 AM EST

| | | | |
|--|---|--|--|
| zhcn5 | zh_tw | zh_cn2 | Zagat Rating for Top 1000 Restaurants in Boston (1000 records) |
| Data Source: awzhcn.xlsx (60855 records) | Data Source: allanquao.xlsx (20 records) | Data Source: allanquao2.xlsx (1 records) | Data Source: boston-area-restaurants.csv (1000 records) |
| Preview New | Preview New | Preview New | Preview New |
| Zagat Rating for Top 1000 Restaurants in Boston (1000 records) Data Set Info Used in Projects (2) Related Data sets (1) + Tags To add tags, click the Tags button at left. Data sources: boston-area-restaurants.csv Created on: 1/6/15 11:26:52 AM EST by omri omri.trub | | | |
| Yelp Boston Restaurant ... | words5000 | withzhcn | WineDefaultForSnippeti... |
| Data Source: reviews-2rest-600reviews... (659 records) | Data Source: words5000.xls (4934 records) | Data Source: ccchznonly.xlsx (2 records) | Data Source: WineDefaultForSnippeti... (304 records) |
| Preview New | Preview New | Preview New | Preview New |
| wine2 | wine | Weather Stations | Weather Info of Month A... |
| (57076 records) | (57076 records) | Data Source: station.xls (4439 records) | Data Source: China-India+America+Jap... (5454 records) |
| Preview New | Preview New | Preview New | Preview New |
| Warranty Claims | Vehicles | Vehicles | Vehicles |
| Data Source: Claims.xlsx (9983 records) | Data Source: Vehicles.xls (8 records) | Data Source: Vehicles.xls (8 records) | Data Source: Vehicles.xls (8 records) |
| Preview New | Preview New | Preview New | Preview New |

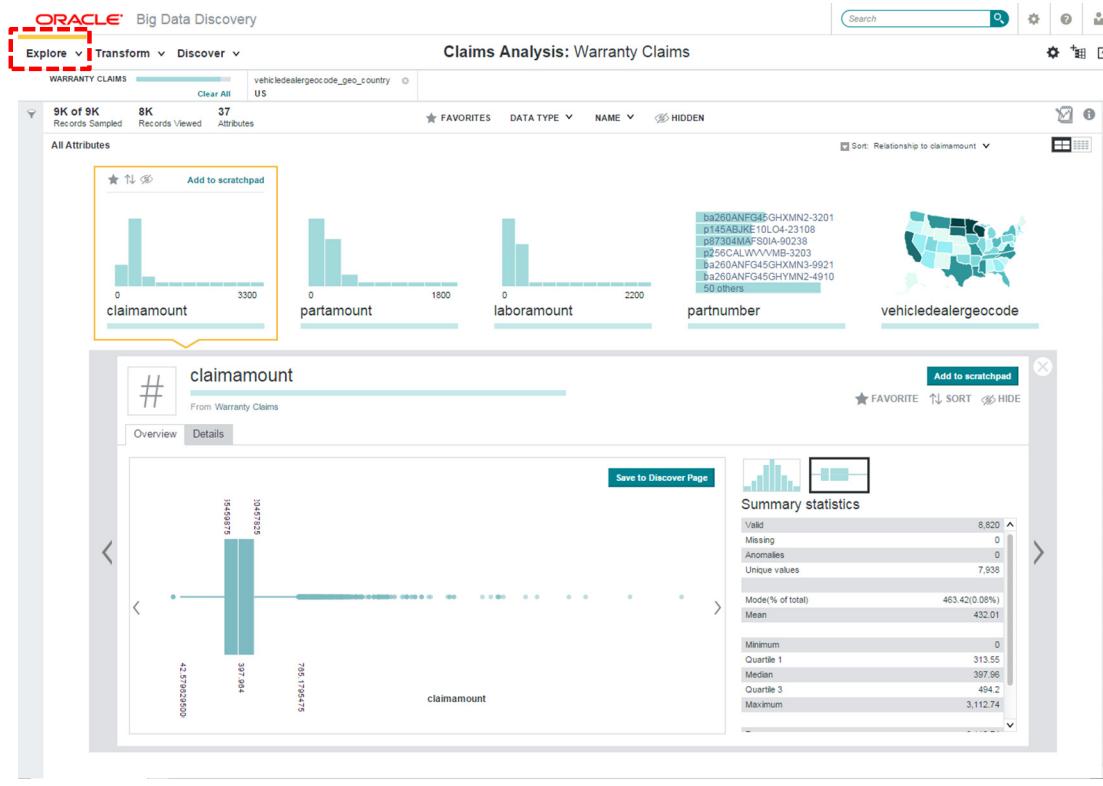
ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Using Big Data Discovery, you can easily find relevant data to start working with by using the Data Sets tab. This interface provides:

- A rich, interactive catalog that enables access to all of the data in your Hadoop cluster. You use familiar search facilities and guided navigation panes to quickly find the data that you want to investigate.
- Data set summaries, annotation from other users, and recommendations for related data sets
- A self-service facility that enables users to provision personal and enterprise data (from excel and csv files on their desktop) to Hadoop

Explore Data



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Once you have found data to work with, you need to understand it before investing in a costly analytics project.

The Big Data Discovery Explore page enables the following data exploration techniques:

- Understand the shape of any data set by:
 - Visualizing its attributes in terms of their data types
 - Sorting these attributes by information potential so the most interesting ones appear first
- Assess individual attribute statistics, understand data quality, and detect anomalies or outliers in the data.
- Use a sandbox area or scratch pad to experiment with combinations of attributes to uncover new patterns and correlations.

Transform and Enrich Data



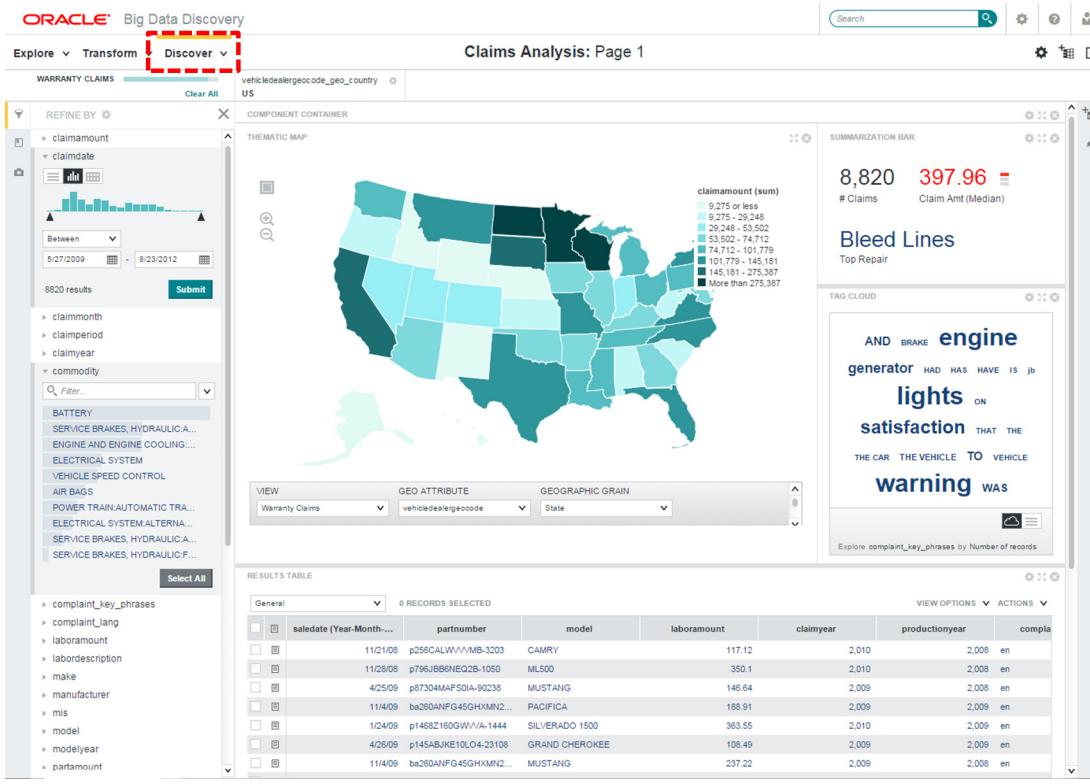
Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

ORACLE

The Big Data Discovery Transform page enables in-place transformation and enrichment of data. This interface provides:

- An intuitive, Excel-like interface for user-driven data wrangling. This feature leverages the power of Hadoop on the data in place.
 - An extensive library of common data transformations such as split, merge, or replace values, and innovative data enrichments like the ability to build out geographic hierarchies from a simple address or extract sentiment and themes from raw text
 - Preview, undo, commit, and replay actions
 - The ability to test your work on in-memory data samples before applying any transforms to the full scale data set in Hadoop.

Discover Information



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The Discover page provides a wide range of features that enable you to:

- Join or mash up different data sets for deeper analytic perspectives
- Quickly build out project pages by dragging and dropping from an extensive library of best practice discovery components, including summary bars, charts, tables, search results lists, tag clouds and so on
- Ask questions of the data by using powerful, yet familiar, keyword search and guided navigation techniques. These visually driven techniques construct and execute queries that enable you to filter through the data and discover valuable information.
- See new patterns by using interactive data visualizations. The interactive data visualizations help you to ask the next question and continue the investigation.

Share Insights



The screenshot shows the Oracle Big Data Discovery interface. A central window titled "US Claims Analysis" displays a dashboard with several charts and data visualizations. At the top of this window, there is a "Relationship between cost components by repair in US" chart. Below it, there are two line graphs showing trends over time. At the bottom of the main window, there are three smaller thumbnail previews of other analysis snapshots: "Claims Analysis_2015-01-06", "Claims Analysis_2015-01-06", and "Claims Analysis_2015-01-01". A red "Add more snapshots" button is located at the bottom right of the main window. To the left of the main window, there is a sidebar titled "SNAPSHOTS" which lists several previous analysis snapshots. At the bottom of the interface, there is a timeline from "10/1/09" to "7/1/12" with a "Max" and "Min" indicator.

ORACLE

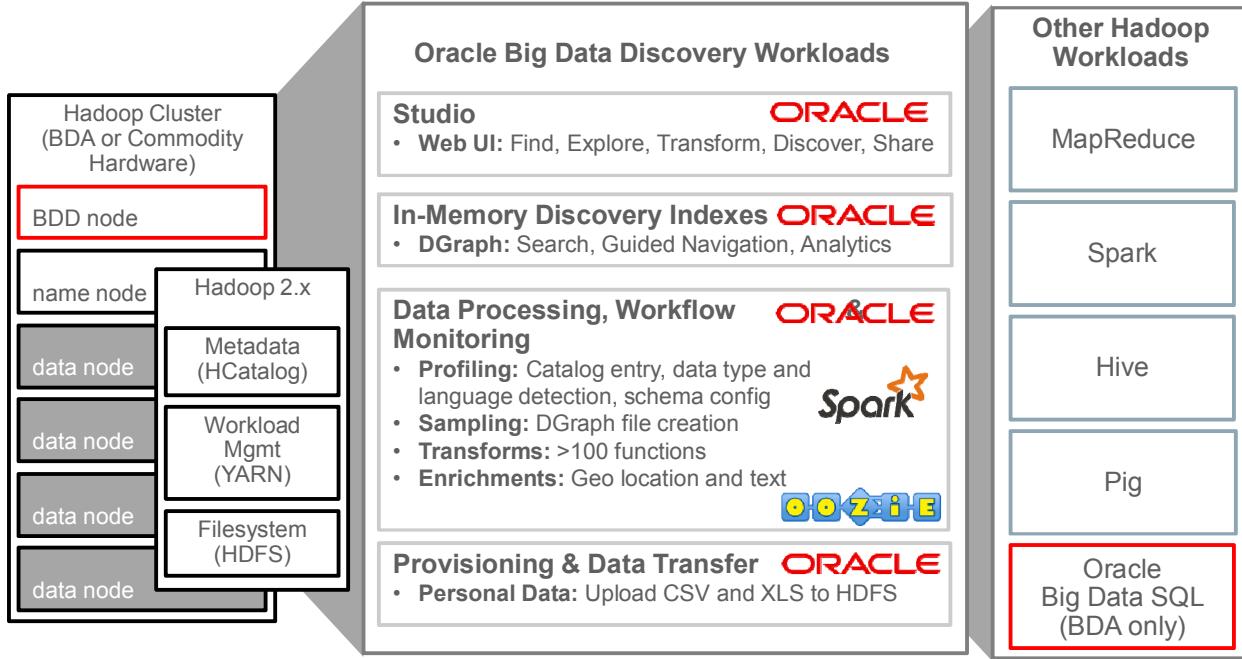
Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Finally, Big Data Discovery enables members of the Big Data team to share and leverage insights across the organization.

From a design perspective, the primary goal of Big Data Discovery is to foster collaboration. This is achieved by sharing BDD projects with others. You can even share bookmarks and “snapshots” of specific areas within your project to share. In addition, snapshots may be put into galleries to help tell Big Data stories.

Finally, you can share personal data insights by publishing your blended and enriched data sets back to HDFS. This enriched data may then be leveraged by other tools and analysis technologies that have been previously covered in this course. These include Oracle Big Data SQL, Oracle Data Mining, Oracle R Enterprise, or any other product or technology that is designed to work with data in Hadoop.

BDD: Technical Innovation on Hadoop



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

ORACLE

From a technology perspective, Oracle Big Data Discovery offers true innovation on Hadoop. BDD leverages the power of distributed storage and compute across servers or nodes to process massive amounts of information without having to move the data around. It is enabled by the following:

- Workloads run in your existing Hadoop cluster.
- A web-based user interface named Studio, which makes it easy for anyone to find, explore, transform, discover, and share data.
- The DGraph server, which allows users to operate on in-memory data sets for fast performance.
- All data processing workloads use:
 - Apache Spark to profile, sample, transform, and enrich massive amounts of information across all of the data nodes in the cluster
 - Oozie to manage the workflow between jobs
- End-users provision of personal data sets into Hadoop using self-service upload wizards
- Oracle Big Data Discovery workloads run right along side any other workloads in the cluster and will take advantage of all the administration and management components provided in the core Hadoop distribution such as YARN and HCatalog.

Additional Resources

- Documentation:
<http://docs.oracle.com/en/bigdata/index.html>
- Oracle Learning Library page:
<http://www.oracle.com/oll/bdd/index.html>
- Oracle.com:
<https://www.oracle.com/bigdata/big-data-discovery/index.html>
 - White Papers
 - Data Sheets
 - Articles
 - Videos
 - Support



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Summary

In this lesson, you should have learned to describe:

- The features of Oracle Big Data Discovery
- The benefits of Oracle Big Data Discovery



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

25

Introduction to the Oracle Big Data Appliance (BDA)

ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Course Road Map

Module 1: Big Data Management System

Module 2: Data Acquisition and Storage

Module 3: Data Access and Processing

Module 4: Data Unification and Analysis

Module 5: Using and Managing Oracle Big Data Appliance

Lesson 25: Introduction to the Oracle Big Data Appliance (BDA)

Lesson 26: Managing Oracle BDA

Lesson 27: Balancing MapReduce Jobs

Lesson 28: Securing Your Data

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

This first lesson in the final module in this course introduces the Oracle Big Data Appliance (BDA) at a high level.

Objectives

After completing this lesson, you should be able to:

- Identify Oracle Big Data Appliance (BDA)
- Identify the hardware and software components of Oracle Big Data Appliance



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Oracle Big Data Appliance

An engineered system of hardware and software that delivers:

- A complete and optimized Hadoop/NoSQL platform
- Single-vendor support for both hardware and software
- An easy-to-deploy solution
- Tight integration with Oracle Database



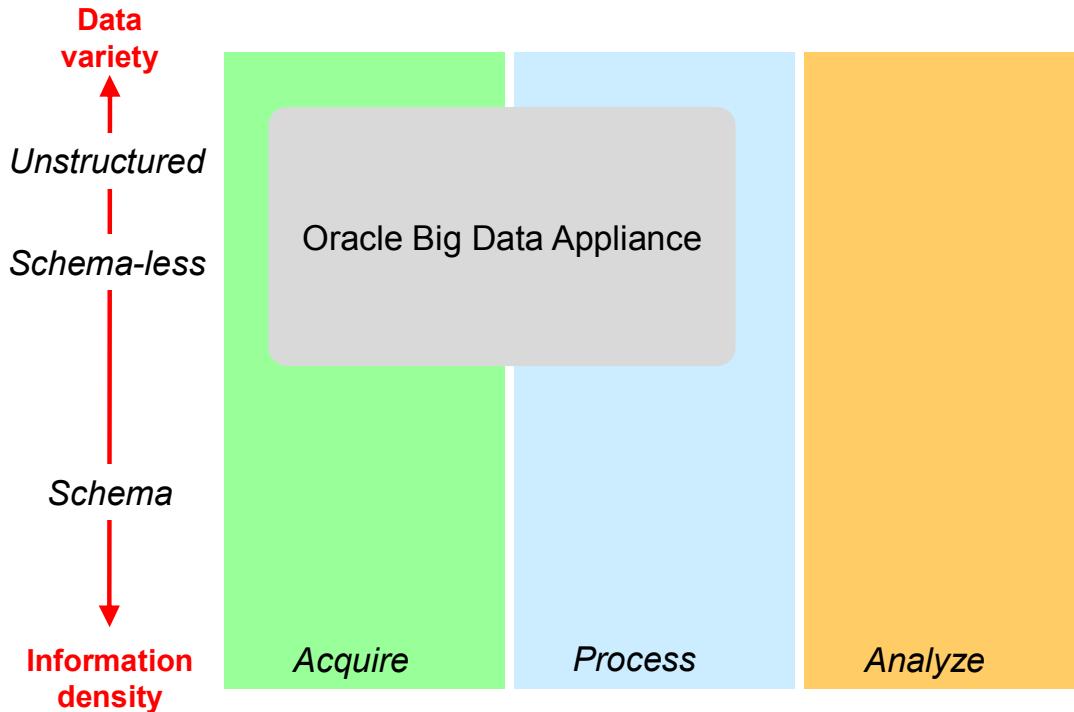
ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Oracle Big Data Appliance is optimized to capture and analyze the massive volumes of varied data generated by social media feeds, email, web logs, photographs, smart meters, sensors, and similar devices.

It is possible to connect your existing Oracle Database host to Big Data Appliance through a network to add new data sources to an existing data warehouse.

Oracle Big Data Appliance: Key Component of the Big Data Management System



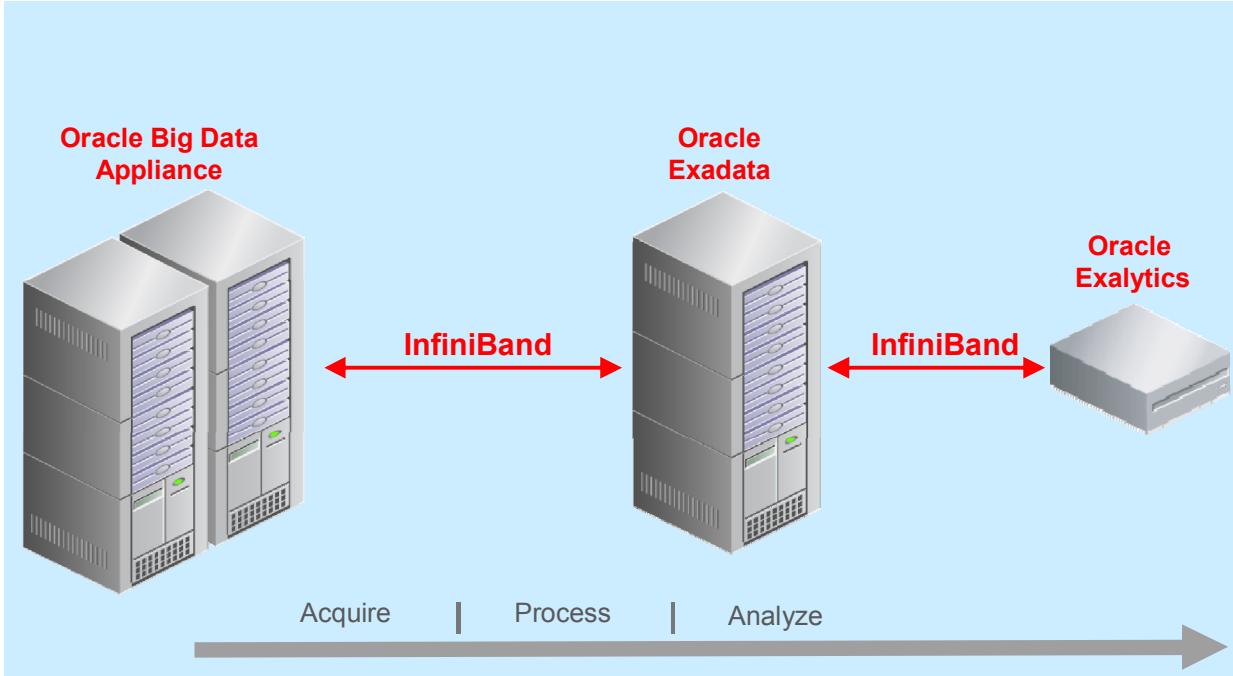
ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Oracle Big Data Appliance is a cluster of industry-standard servers that are specifically built to leverage the functionality of Cloudera's Distribution including Apache Hadoop (CDH) and Oracle NoSQL Database.

Oracle Big Data Appliance is used as the main part of the Oracle Big Data solution to acquire the unstructured data and organize it into proper form.

Oracle-Engineered Systems for Big Data



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Oracle Big Data Appliance is engineered to work with Oracle Exadata Database Machine and Oracle Exalytics In-Memory Machine to provide the most advanced analysis of all data types, with enterprise-class performance, availability, supportability, and security.

An InfiniBand network is used to establish connections.

The Available Oracle BDA Configurations

The Big Data Appliance is offered in three sizes:

- Big Data Appliance ***Starter Rack***: 6 nodes
- Big Data Appliance ***Starter Rack with one 6-nodes In-Rack Expansion Kit***: 12 nodes
- Big Data Appliance ***Full Rack***: 18 nodes (Full Rack or a Starter Rack with two In-Rack Expansion Kits)



ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Using the Mammoth Utility

Mammoth is a command-line utility for installing and configuring the Oracle BDA software. By using Mammoth, you can:

- Set up a cluster for either CDH or Oracle NoSQL Database
- Create a cluster on one or more racks
- Create multiple clusters on an Oracle BDA rack
- Extend a cluster to new servers on the same or new rack
- Update a cluster with new software
- Configure or remove optional services Service Request
- Deploy or remove the Oracle Enterprise Manager system monitoring plug-in for Oracle BDA



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

You can use the Mammoth utility to configure or remove optional services, including network encryption, disk encryption, Kerberos, Sentry, Oracle Audit Vault and Database Firewall, and Auto Service Request.

Before you install any software, you need to download the Mammoth bundle, which contains the installation files and the base image. Before you install the software, you must also use Oracle Big Data Appliance Configuration Generation Utility to generate the configuration files.

To download the Mammoth bundle, locate the download site in either My Oracle Support or Automated Release Updates (ARU).

You use the same Mammoth bundle for all procedures regardless of the rack size, and whether you are creating CDH or Oracle NoSQL Database clusters, or upgrading existing clusters.

For information about Oracle BDA Mammoth utility, see the Oracle Big Data Appliance Owner's Guide Release 4 (4.0) at:

https://docs.oracle.com/cd/E55905_01/doc.40/e55796/toc.htm

Using the Mammoth Utility

- Mammoth installs and configures the software on Oracle BDA (across all servers in the rack) by using the files generated by BDA Configuration Generation Utility.
- A cluster can be dedicated to either CDH (Hadoop) or Oracle NoSQL Database.
- Mammoth also performs the following tasks:
 - Creates the required user accounts
 - Starts the correct services
 - Sets the appropriate configuration parameters. When it is done, you could have a fully functional Hadoop or NoSQL cluster
- The `dcli` utility executes commands across a group of servers on Oracle BDA and returns the output.



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

For additional information about the DCLI utility, see the *Executing Commands Across a Cluster Using the dcli Utility*, in the *Oracle Big Data Appliance Guide* documentation reference.

Using Oracle BDA Configuration Generation Utility

- Acquires information from you, such as IP addresses and software preferences, that are required for deploying Oracle BDA.
- After guiding you through a series of pages, the utility generates a set of configuration files.
- These files help automate the deployment process and ensure that Oracle Big Data Appliance is configured to your specifications.



ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Configuring Oracle Big Data Appliance

To configure Oracle Big Data Appliance:

1. Download Oracle BDA Configuration Generation Utility, BDAConfigurator-version.zip, from OTN at:
 - <http://www.oracle.com/technetwork/database/bigdata-appliance/downloads/index.html>
Opens a new window
2. Extract the files in BDAConfigurator-version.zip.
3. Change to the BDAConfigurator-version directory.
4. Run Oracle BDA Configuration Generation Utility:

```
$ sh bdaconf.sh
```

5. On the **Welcome** page, select a configuration type.
6. Click **Import** if the button is activated, and select a previously saved configuration file.



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Note: The system must run Oracle JRE 1.6 or later.

Configuring Oracle Big Data Appliance

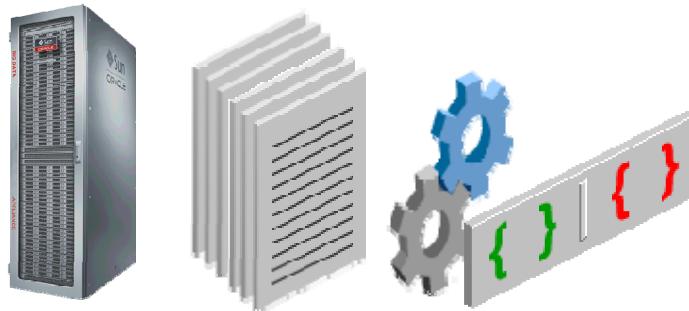
7. Follow the steps of the wizard. On the **Complete** page, click **Create Files**.
8. Validate the network configuration. See *Validating the Network Settings* in the *Oracle BDA Owner's Guide*.
9. Send the generated `bda.zip` file to your Oracle representative.



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The Generated Configuration Files

- `bda-timestamp.zip`
 - `bda-install-preview.html`
 - `bda-preinstall-checkip.sh`
 - `rack_name-network.json`
 - `cluster_name-config.json`
 - `master.xml`
 - `rack_name-networkexpansion.json`



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

`bda-timestamp.zip`

Contains a copy of the configuration files. If an Oracle customer service representative will perform the installation, then send this file to Oracle before the installation date. Otherwise, transfer the file to a USB drive for copying to Oracle Big Data Appliance.

`bda-install-preview.html`

Provides a report that lists all the details of the configuration. You can view the this report in a browser. Check it carefully to ensure that all of the settings are correct.

`bda-preinstall-checkip.sh`

Runs a series of tests to ensure the specified names and IP addresses for Oracle Big Data Appliance were added correctly to the name server, and they do not conflict with the existing network configuration.

`rack_name-network.json`

Contains the network configuration for a full rack, a starter rack, or a starter rack with one in-rack expansion kit. It contains information about all the servers, switches, and PDUs.

cluster_name-config.json

Contains all the information for a cluster, including the network configuration, port numbers, user names, and passwords. The configuration utility creates a separate parameter file for each cluster. If several clusters are being configured, then each parameter file is located in a separate subdirectory.

If an in-rack expansion kit is being configured as an addition to an existing cluster, then the configuration utility does not generate a parameter file; the Mammoth utility generates it.

master.xml

Contains all the configuration settings in XML format so that Oracle Big Data Appliance Configuration Generation Utility can read it. To alter the configuration of an Oracle Big Data Appliance deployment, you can load this file, enter the changes, and regenerate the configuration files.

This file is used only by Oracle Big Data Appliance Configuration Generation Utility. It is not used for the actual configuration of Oracle Big Data Appliance.

rack_name-networkexpansion.json

Contains the network configuration for one or two in-rack expansion kits. It contains information about all the servers, but no information about the switches and PDUs. This file is generated only when an expansion kit is being installed and configured.

The Oracle BDA Configuration Generation Utility Pages

- **Customer Details** page
- **Hardware Selection** page
- **Rack Details** page
- **Networking** page
- **Client Ethernet Network** page
- **Administration Network** page
- **InfiniBand Network** page
- **General Network Properties** page
- **Review and Edit Details** page
- **Define Clusters** page
- **Cluster** page
- **Complete** page



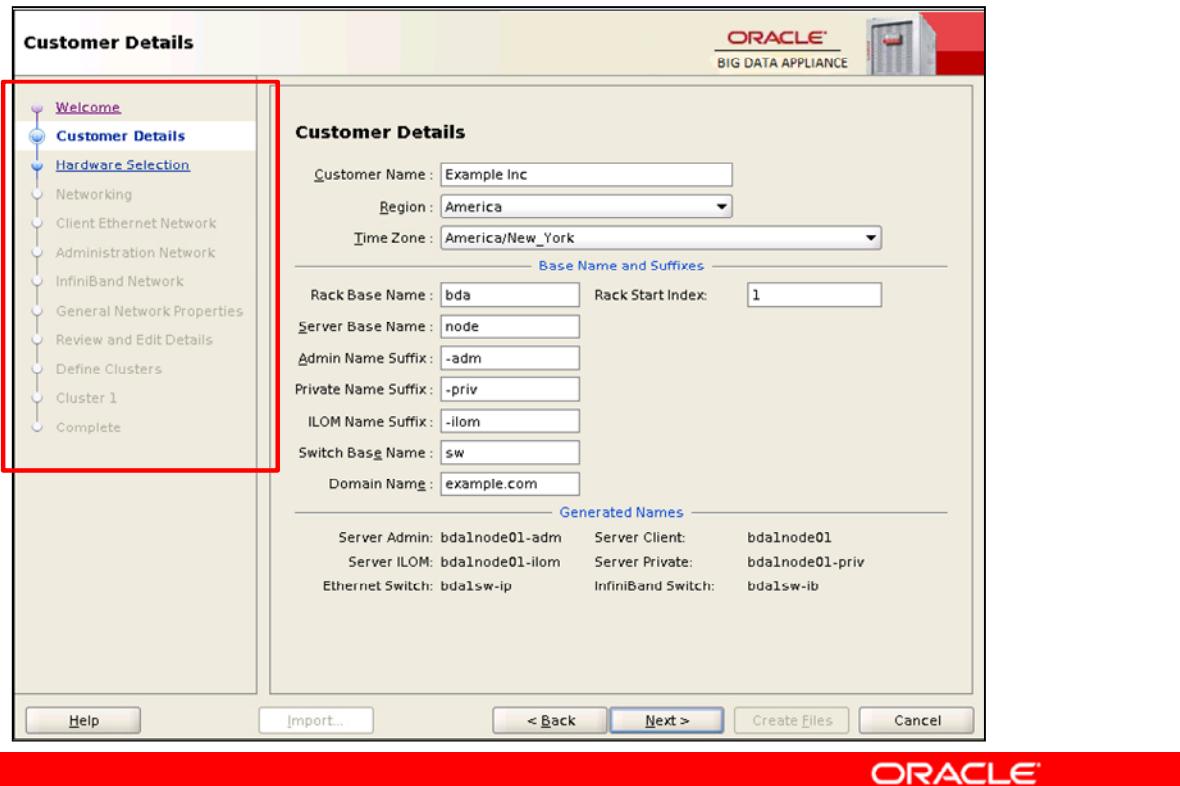
ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

For information about Oracle BDA Configuration Utility pages, see the Oracle Big Data Appliance Owner's Guide Release 4 (4.0) at:

https://docs.oracle.com/cd/E55905_01/doc.40/e55796/toc.htm

Using Oracle BDA Configuration Generation Utility: The Customer Details Page



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The screen capture in the slide shows the **Customer Details** page. The fields on this page are as follows

- **Customer Name:** The name of your enterprise. This is a required field.
- **Region:** The geographic area where Oracle Big Data Appliance will be installed
- **Time Zone:** The time zone for your installation. You must select the appropriate region before selecting the time zone.
- **Rack Base Name:** A maximum of 10 alphanumeric characters for the name of the Oracle Big Data Appliance rack
- **Rack Start Index:** A digit that uniquely identifies the rack. It is a suffix of the rack base name.
- **Server Base Name:** Base name for all servers. A two-digit suffix uniquely identifies each server. The rack name and server base name are used to generate the host names for all network interfaces: eth0, bondib0, bondeth0, and Oracle ILOM. For example, a rack base name of bda, a rack start index of 1, and a server base name of node results in host names of bdalnode01, bdalnode02, and so forth.
- **Admin Name Suffix:** Suffix to the basic host name to form the eth0 host names
- **Private Name Suffix:** Suffix to the basic host name to form the bondib0 host name

Using Oracle BDA Configuration Generation Utility: The Hardware Selections Page

- Lists the available hardware configurations
- Choose one or more racks.
- You can choose the same type of rack multiple times:
 - **Full Rack:** Contains 18 servers
 - **Starter Rack:** Contains six servers
 - **Starter Rack with In-Rack Expansion:** Contains 12 servers
 - **Single In-Rack Expansion Kit:** Contains six servers
 - **Double In-Rack Expansion Kit:** Contains 12 servers



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The Hardware Selection page identifies one or more racks that you want to deploy at the same time. The racks must be cabled together. For example, if you are deploying three full racks, then add Full Rack three times to your deployment.

Choose the option that describes the type of hardware installation you are configuring:

- One or more new Big Data Appliance racks being installed: You enter all new data for this choice.
- One or more Big Data Appliance racks being added to an existing group of Big Data Appliances: This choice activates the **Import** button, so that you can select the `BdaDeploy.json` file that was used to configure the last rack in the group.
- One or two in-rack expansion kits being added to a Big Data Appliance starter rack: This choice activates the **Import** button, so that you can select the `BdaDeploy.json` file that was last used to configure the rack (either the starter rack or one in-rack expansion kit).
- An in-process configuration using a saved `master.xml` configuration file: This choice activates the **Import** button, so that you can select the `master.xml` file, and then continue the configuration.

The Oracle BDA Configuration Generation Utility:

The Define Clusters Page

| Entity Reference | Description |
|-------------------------------------|--|
| Number of Clusters to Create | Select the number of clusters |
| Cluster Name | Enter a unique name for the cluster. The name must begin with a letter and can consist of alphanumeric characters, underscores (_), and dashes (-). |
| Cluster Type | <ul style="list-style-type: none"> CDH cluster: Installs Cloudera's Distribution including Apache Hadoop and optional software on cluster of new servers NoSQL DB cluster: Installs Oracle NoSQL Database on a cluster of new servers Adding to existing cluster: Installs the same software on the new servers as the rest of the cluster |
| Unassigned Servers | From the list on the left, select the servers for the cluster and move them to the list of assigned servers on the right |
| Assigned Servers | <i>A CDH cluster must have a minimum of six servers, and an Oracle NoSQL Database cluster must have a minimum of three servers. All clusters must be composed of multiples of three servers.</i> |



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

You use the **Define Clusters** page to identify the number of clusters to create and the servers that compose each cluster. You can configure clusters for either CDH or Oracle NoSQL Database.

You can configure multiple clusters in a single rack, or a single cluster can span multiple racks.

Each CDH cluster must have at least six servers, and each Oracle NoSQL Database cluster must have at least three servers. Thus, a starter rack supports one CDH cluster, a starter rack with one in-rack expansion supports up to two CDH clusters, and a full rack supports up to three CDH clusters.

The Oracle BDA Configuration Generation Utility: The Cluster *n* Page

Select the software to install on this cluster. The fields displayed in this field depend on the type of cluster being configured. The available choices are:

- **Adding to an Existing Cluster**
- **A New Oracle NoSQL Database Cluster**
 - Oracle NoSQL Community Edition, or
 - Oracle NoSQL Enterprise Edition
- **A New CDH Cluster**



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Select the software to install on this cluster. The fields displayed on this field depend on the type of cluster being configured. The available choices are:

- **Adding to an Existing Cluster:** You are done with the software configuration. The Mammoth utility configures the software for the new servers the same as the other servers in the cluster.
- **A New Oracle NoSQL Database Cluster:** You can use this section on the Cluster page to install either one of the following:
 - The Oracle NoSQL Database Community Edition. This is included in the license for Oracle Big Data Appliance.
 - The Oracle NoSQL Database Enterprise Edition. This requires a separate license. You must have this license to install Enterprise Edition on Oracle Big Data Appliance.
 - Oracle NoSQL Database 12c Release 1.3.0.5 and later versions support secondary zones, which are composed of nodes that function only as replicas. You can use the secondary zones on Oracle Big Data Appliance to maintain extra copies of the data for increased redundancy (default is 3) and read capacity, or to provide low latency, read access to data at a distant location.
- **A New CDH Cluster:** jkl

BDA Configurations: Full Rack

Each node in the BDA can contain a Sun Server X4-2L, Sun Server X3-2L, or Sun Fire X4270 M2 (full rack only) which is based on availability



BDA Full Rack

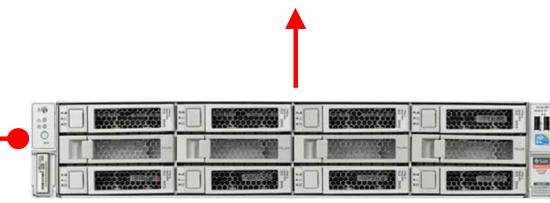
| |
|------------------------------|
| Big Data Server 18 (node 18) |
| Big Data Server 17 (node 17) |
| Big Data Server 16 (node 16) |
| Big Data Server 15 (node 15) |
| Big Data Server 14 (node 14) |
| Big Data Server 13 (node 13) |
| Big Data Server 12 (node 12) |
| Big Data Server 11 (node 11) |
| Big Data Server 10 (node 10) |
| Big Data Server 9 (node 9) |
| Big Data Server 8 (node 8) |
| Big Data Server 7 (node 7) |
| Big Data Server 6 (node 6) |
| Big Data Server 5 (node 5) |
| Big Data Server 4 (node 4) |
| Big Data Server 3 (node 3) |
| Big Data Server 2 (node 2) |
| Big Data Server 1 (node 1) |

Each Sun Server X4-2L Server contains among other components the following:

- 1 Sun Server X4-2L server base
- 2 Eight-core Intel Xeon E5-2650 v2 processors (2.6 GHz)
- 8 (8) GB DDR3 2RX4 1600 MHz DIMMs (64 GB RAM expandable up to 512 GB)
- 12 (4) TB 3.5-inch 7200 RPM drives high capacity SAS (hot swappable)

Each Sun Server X3-2L Server contains among other components the following:

- 1 Sun Server X3-2L server base
- 2 Eight-core Intel Xeon E5-2660 processors (2.2 GHz)
- 8 (8) GB DDR3 2RX4 1600 MHz DIMMs (64 GB RAM expandable up to 512 GB)
- 12 (3) TB 3.5-inch 7200 RPM drives high capacity SAS (hot swap)



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

In the slide example, only two of the three possible servers-partial components are shown, namely: Sun Server X4-2L Server and Sun Server X3-2L Server.

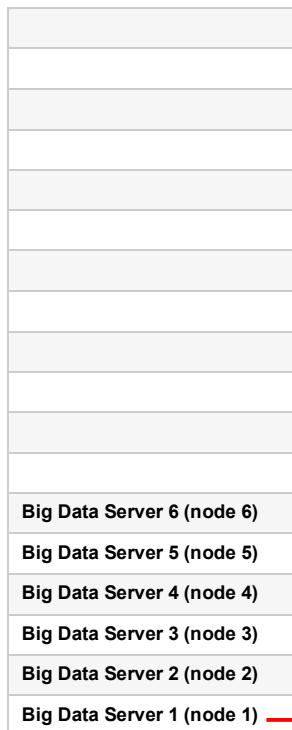
For detailed information about the complete components of the Sun Server X4-2L, Sun Server X3-2L Server, and the Sun Fire X4270 M2, see the *Oracle Big Data Appliance Owner's Guide Release 4 (4.0)* documentation reference.

BDA Configurations: Starter Rack

Each node in the BDA can contain a Sun Server X4-2L or Sun Server X3-2L which is based on availability?????



BDA Starter Rack

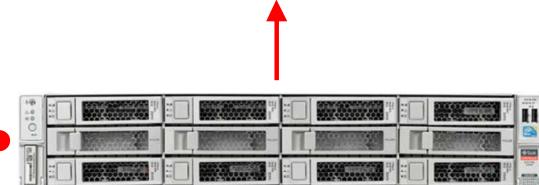


Each Sun Server X4-2L Server contains among other components the following:

- 1 Sun Server X4-2L server base
- 2 Eight-core Intel Xeon E5-2650 v2 processors (2.6 GHz)
- 8 (8) GB DDR3 2RX4 1600 MHz DIMMs (64 GB RAM expandable up to 512 GB)
- 12 (4) TB 3.5-inch 7200 RPM drives high capacity SAS (hot swappable)

Each Sun Server X3-2L Server contains among other components the following:

- 1 Sun Server X3-2L server base
- 2 Eight-core Intel Xeon E5-2660 processors (2.2 GHz)
- 8 (8) GB DDR3 2RX4 1600 MHz DIMMs (64 GB RAM expandable up to 512 GB)
- 12 (3) TB 3.5-inch 7200 RPM drives high capacity SAS (hot swap)

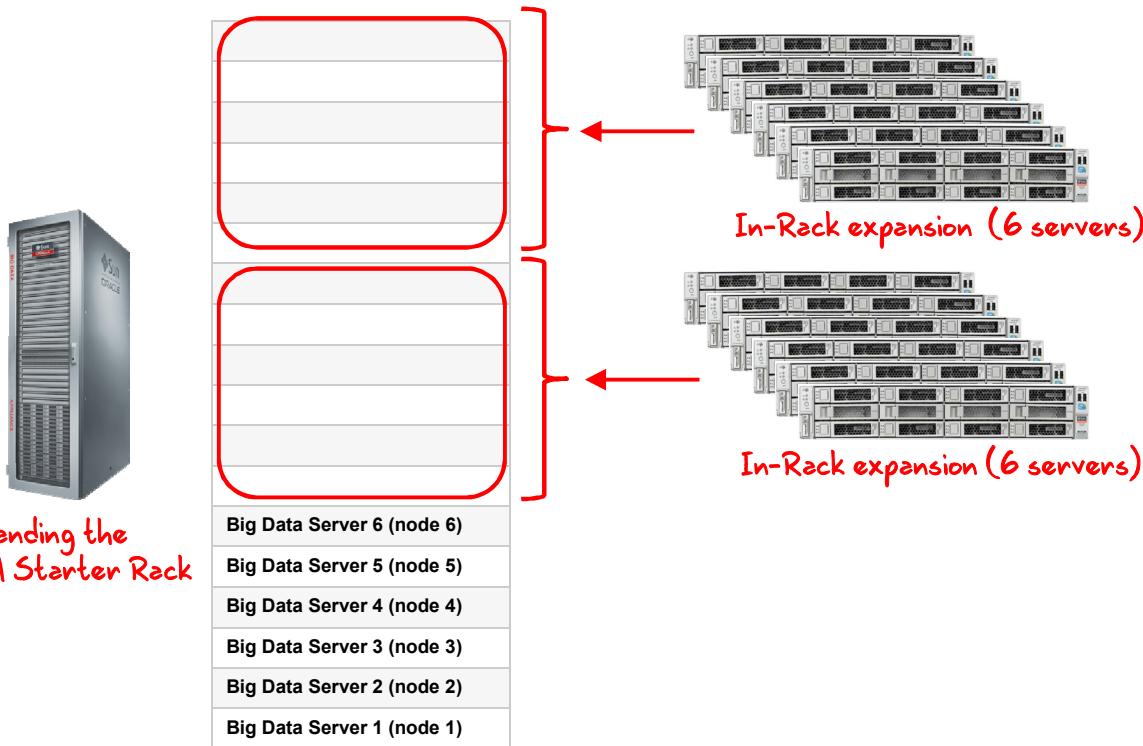


ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

An Oracle Big Data Appliance starter rack has the same hardware configuration as a full rack, except that it comes with six servers instead of 18. All switches and power supplies are included in the starter rack, and do not need to be upgraded or supplemented to support additional servers.

BDA Configurations: In-Rack Expansion



ORACLE

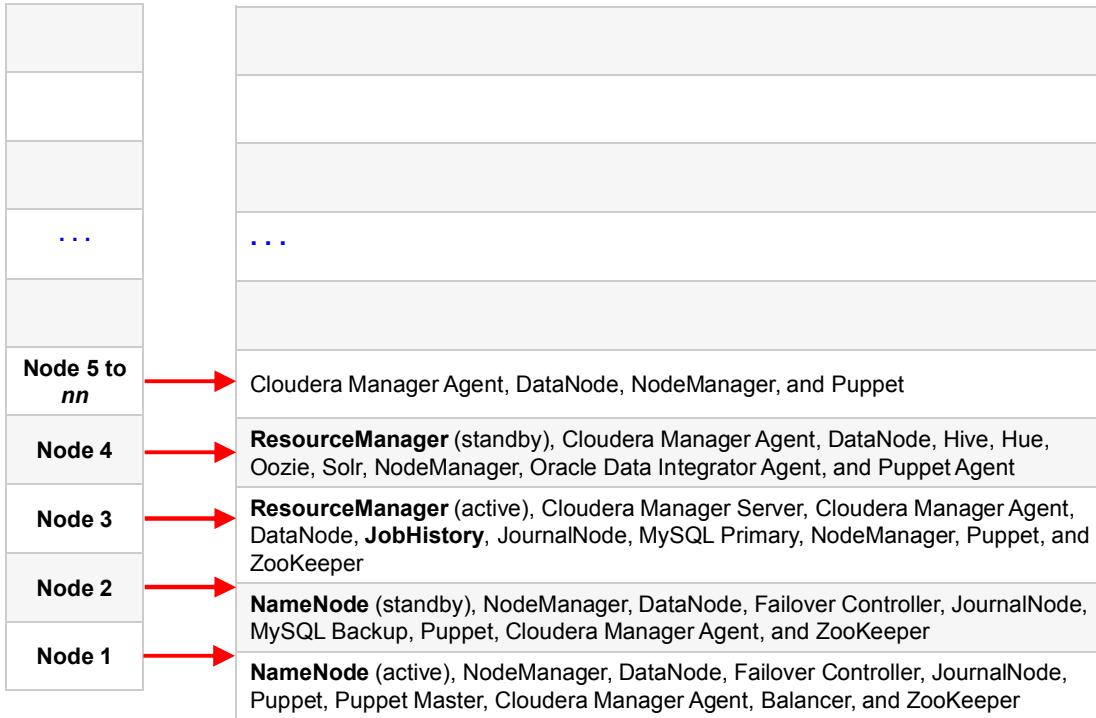
Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

An in-rack expansion kit provides six more servers and the components needed to install them. You can install one expansion kit to form a 12-server rack, or two expansion kits to form a full, 18-server rack.

If you install the second in-rack expansion kit at a later time, you must repeat the instructions provided in the *Oracle Big Data Appliance Owner's Guide Release 4 (4.0)* documentation reference. Otherwise, you can install and configure all 12 new servers together.

Note: The in-rack expansion kit contains the server model that is currently being produced and shipped. All the servers in the kit will be the same, but they might be bigger, faster, and better than the servers in your old, outdated starter rack.

BDA Starter Rack: Hadoop Cluster Only



ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Puppet

A configuration management tool for deploying and configuring software components across a cluster. The Oracle Big Data Appliance initial software installation uses Puppet.

The Puppet tool consists of these components: puppet agents, typically just called puppets; the puppet master server; a console; and a cloud provisioner.

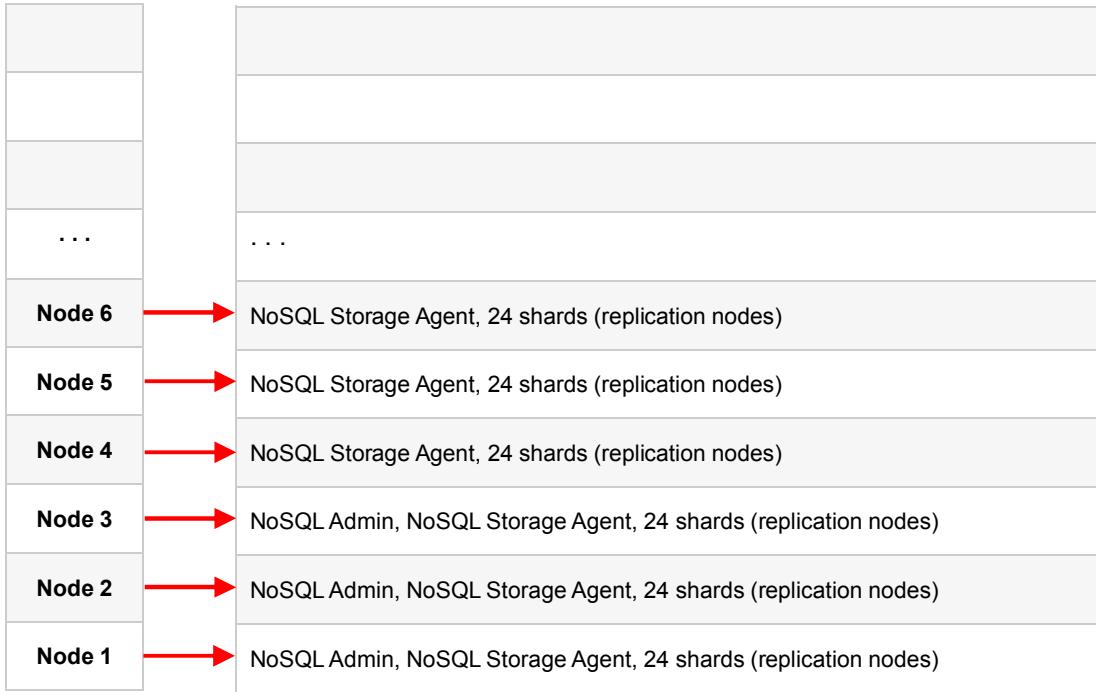
Puppet Agent

A service that primarily pulls configurations from the puppet master and applies them. Puppet agents run on every server in Oracle Big Data Appliance.

Puppet Master

A service that primarily serves configurations to puppet agents

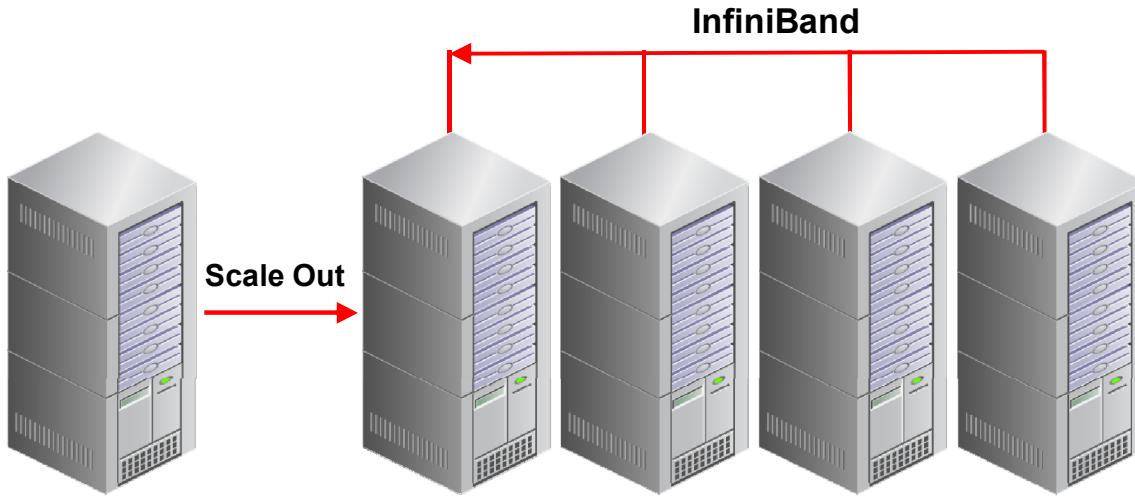
BDA Starter Rack: NoSQL Cluster Only



ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Big Data Appliance: Horizontal Scale-Out Model



ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

In-Rack Expansion delivers six server modular expansion block. Full Rack delivers optimal blend of capacity and expansion options, growing by adding rack—up to 18 racks without additional switches.

The same method can also be used to connect Exadata machines in a Big Data Appliance configuration.

You can increase the scalability of Big Data Appliance by connecting racks to each other by using a QDR InfiniBand spine switch.

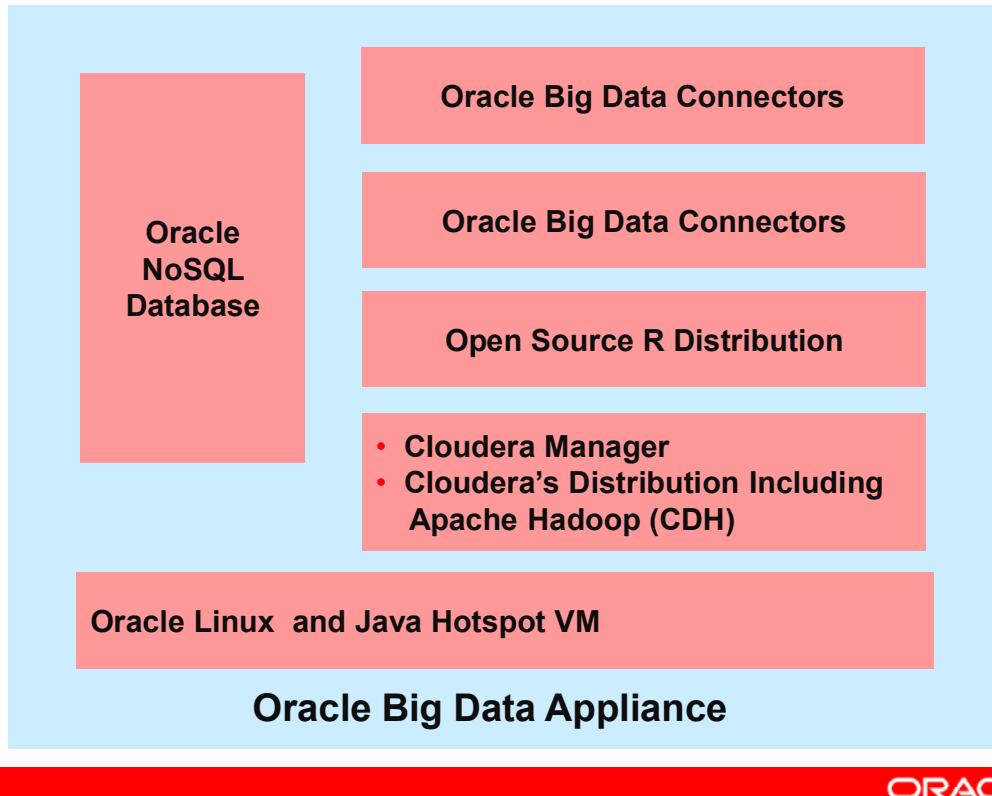
Physical Ethernet connections are created only between the site network and the gateway switches. The Big Data Appliance servers are connected only by InfiniBand to those switches.

Each server has two InfiniBand connections (one to each gateway switch) in an active backup mode; only the active InfiniBand connection is used for all InfiniBand traffic to that server. If that connection fails, it immediately fails over to the other connection.

Big Data Appliance InfiniBand Network Details

- Uses InfiniBand switches
- Runs the Subnet Manager to automatically discover the network topology
Note: Only one Subnet Manager is active at a time.
- Uses two “leaf” switches to connect individual server IB ports
- Uses one “internal spine” switch to scale out to additional racks

Big Data Appliance: Software Components



ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The Big Data Appliance foundation software includes:

- Oracle Linux
- Java Hotspot VM
- Cloudera's Distribution Including Apache Hadoop (CDH)
- Cloudera Manager
- Open-source R Distribution

The application software includes:

- Oracle NoSQL Database Community Edition
- Oracle Big Data Connectors (separately licensed software that can be preinstalled and configured on the appliance)

A brief description on the components were discussed in the previous lessons.

Each Big Data Appliance software component is described in detail in the following lessons.

Note: For the latest information about licensing, see the following page:

<http://www.oracle.com/technetwork/server-storage/engineered-systems/bigdata-appliance/overview/index.html>

Oracle Big Data Appliance and YARN

- By default, Oracle Big Data Appliance uses the YARN implementation of MapReduce.
- You can:
 - Use classic MapReduce (MR1) instead
 - Activate either the MapReduce service or the YARN service
- You cannot use both implementations in the same cluster.

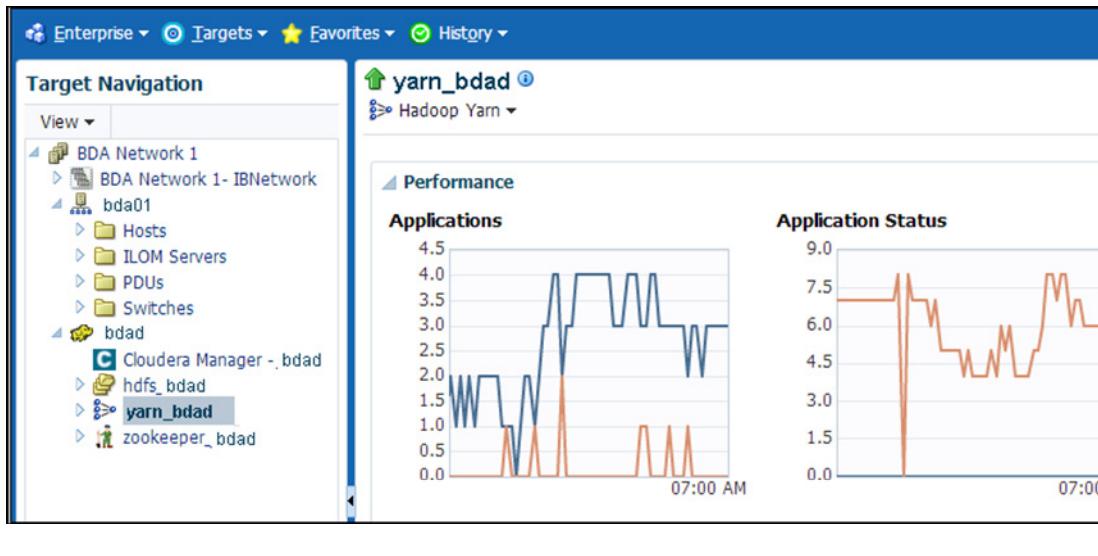


ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Stopping the YARN Service

1. Locate YARN in the list of services on the Status tab of the Home page.
2. Expand the YARN menu and click Stop.



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The screenshot shows the YARN page in Oracle Enterprise Manager.

Critical and Noncritical Nodes in an Oracle BDA CDH Cluster

| Node Name | Initial Node Position | Critical Functions |
|-------------------------|-----------------------|--|
| First NameNode | Node01 | ZooKeeper, first NameNode, NameNode failover controller, balancer, puppet master |
| Second NameNode | Node02 | ZooKeeper, second NameNode, NameNode failover controller, MySQL backup server |
| First ResourceManager | Node03 | ZooKeeper, first ResourceManager, Cloudera Manager server, MySQL primary server |
| Second Resource Manager | Node04 | Second ResourceManager, Oracle Data Integrator agent, Hue, Hive, Oozie |



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Critical nodes are required for the cluster to operate normally and to provide all services to users. In contrast, the cluster continues to operate with no loss of service when a noncritical node fails. The table in the slide shows the initial location of services for clusters that are configured on a single rack.

On single-rack clusters, the critical services are initially installed on the first four nodes of the cluster. If there is a hardware failure on one of the critical nodes, the services are moved to a noncritical server. For example, if node02 fails, you might move its critical services to node05.

In a multirack cluster, some critical services run on the first server of the second rack.

To move a critical node, you must ensure that all clients are reconfigured with the address of the new node. Alternatively, you can wait for the failed server to be repaired. You must weigh the loss of services against the inconvenience of reconfiguring the clients.

The noncritical nodes (node05 to node18) are optional. Oracle Big Data Appliance continues to operate with no loss of service if a failure occurs. The NameNode automatically replicates the lost data so that it always maintains three copies. MapReduce jobs execute on copies of the data that are stored elsewhere in the cluster.

First NameNode and Second NameNode

- First NameNode:
 - One instance initially runs on node01.
 - Second Namenode takes over if node01 fails.
- Second NameNode:
 - One instance initially runs on node02.
 - The function of the NameNode either fails over to the first NameNode (node01) or continues there without a backup if node02 fails.



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

One instance of the first NameNode initially runs on node01. If this node fails or goes offline (for example, if there is a restart), the second NameNode (node02) automatically takes over to maintain the normal activities of the cluster.

Alternatively, if the second NameNode is already active, it continues without a backup. With only one NameNode, the cluster is vulnerable to failure. The cluster has lost the redundancy needed for automatic failover.

The puppet master also runs on this node. Because the Mammoth utility uses Puppet, you cannot install or reinstall the software.

Note

- In multirack clusters, the NameNode service is installed on the first server of the *second* rack.
- One instance of the second NameNode initially runs on node02. If this node fails, the function of the NameNode either fails over to the first NameNode (node01) or continues there without a backup.
- The MySQL backup database also runs on the NameNode. MySQL Database continues to run, although there is no backup of the master database.

First ResourceManager and Second ResourceManager

First ResourceManager

1. One instance of the ResourceManager initially runs on node03.
2. The second ResourceManager (node04) automatically takes over to distribute MapReduce tasks if node03 fails or goes offline.
3. Disrupted services are Cloudera Manager and MySQL Master Database.

Second ResourceManager

1. One instance of the ResourceManager initially runs on node04.
2. If this node fails, the function of the ResourceManager either fails over to the first ResourceManager (node03) or continues there without a backup.
3. Disrupted services are Oracle Data Integrator, Hive, Hue, and Oozie



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

In the first ResourceManager, one instance of the ResourceManager initially runs on node03. If this node fails or goes offline (for example, if there is a restart), the second ResourceManager (node04) automatically takes over to distribute MapReduce tasks to specific nodes across the cluster. Alternatively, if the second ResourceManager is already active, it continues without a backup. The cluster is vulnerable to failure if there is only one ResourceManager. The cluster also loses the redundancy that is required for automatic failover.

Cloudera Manager and MySQL Master Database services are also disrupted.

In the second ResourceManager, one instance of the ResourceManager initially runs on node04. If this node fails, the function of the ResourceManager either fails over to the first ResourceManager (node03) or continues there without a backup.

Oracle Data Integrator, Hive, Hue, and Oozie services are also disrupted.

Hardware Failure in Oracle NoSQL

- In an Oracle NoSQL environment, hardware failure may occur due to the failure of:
 - Disk
 - Storage Node
 - Zone
 - Data center
- If a store is highly available, hardware failure would not bring down the store completely.



ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

In an Oracle NoSQL environment, hardware failure occurs due to the failure of:

- **Disk:** If the disk is corrupted, the replication node that uses that disk would not be able to read/write data from/to the disk.
- **Storage Node:** If the Storage Node fails, all the replication nodes present in the Storage Node would be unable to service read/write requests.
- **Zone:** If the zone fails (usually due to power outage or natural calamities), all the Storage Nodes present in the zone (including the replication nodes) would be unable to service read/write requests.
- **Data center:** If the data center fails, all the zones within the data center (including the Storage Nodes and replication nodes) would be unable to service read/write requests.

However, if the store is deployed in a way that it is made highly available, none of the above failures would bring down the store completely, which means that if the above-mentioned hardware failure occurs, the read/write requests will still be serviced.

Note: For information on recovering from hardware failures, see the *Oracle NoSQL Database, 12c Release 1 Administrator Guide*.

Oracle Integrated Lights Out Manager (ILOM): Overview

- Oracle ILOM provides preinstalled advanced service processor (SP) hardware and software to manage and monitor the Oracle BDA components.
- You can use Oracle ILOM to:
 - Learn about hardware errors and faults as they occur
 - Remotely control the power state of a server
 - View the graphical and nongraphical consoles
 - View the current status of sensors and indicators on the system
 - Determine the hardware configuration of your system
 - Receive generated alerts about system events in advance



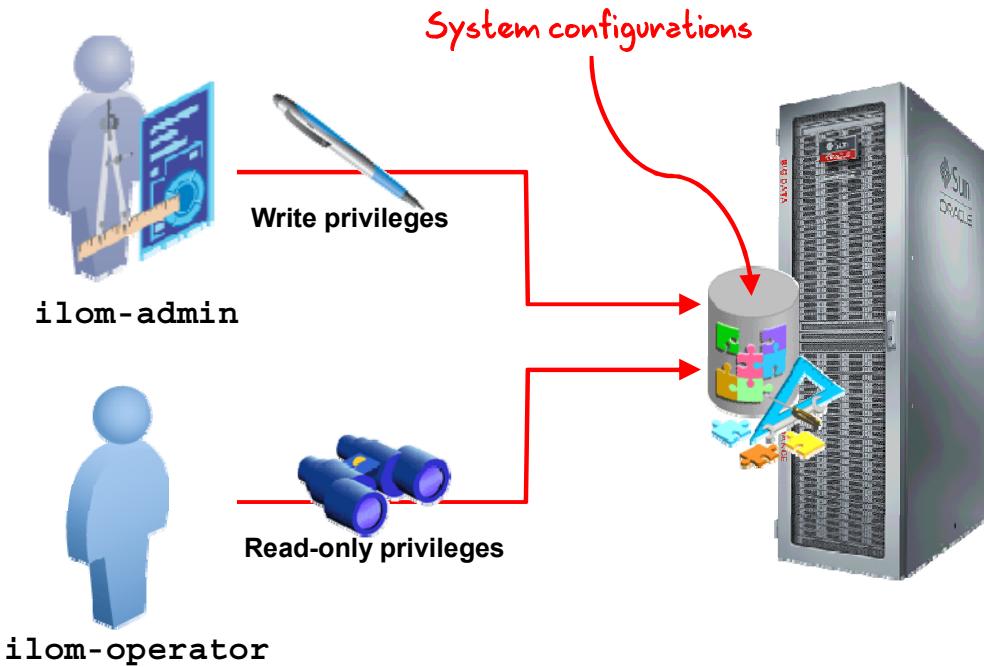
Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The Oracle ILOM service processor runs its own embedded operating system and has a dedicated Ethernet port, which together provide out-of-band management capability. In addition, you can access Oracle ILOM from the server operating system (Oracle Linux). By using Oracle ILOM, you can remotely manage Oracle Big Data Appliance as if you were using a local KVM.

For detailed information on ILOM, see the following resources:

- *Oracle Big Data Appliance Owner's Guide Release 4 (4.0)* documentation reference
- For Sun Server X4-2L and Sun Server X3-2L servers, see the *Oracle Integrated Lights Out Manager 3.1 documentation library* at http://docs.oracle.com/cd/E24707_01/index.html
- For Sun Fire X4270 M2 servers, see the *Oracle Integrated Lights Out Manager 3.0 documentation library* at <http://docs.oracle.com/cd/E19860-01/>

Oracle ILOM Users



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Connecting to Oracle ILOM Using the Network

You can access the features and functions of Oracle ILOM by using either of two supported interfaces:

- Browser-based web interface
- Command-line interface

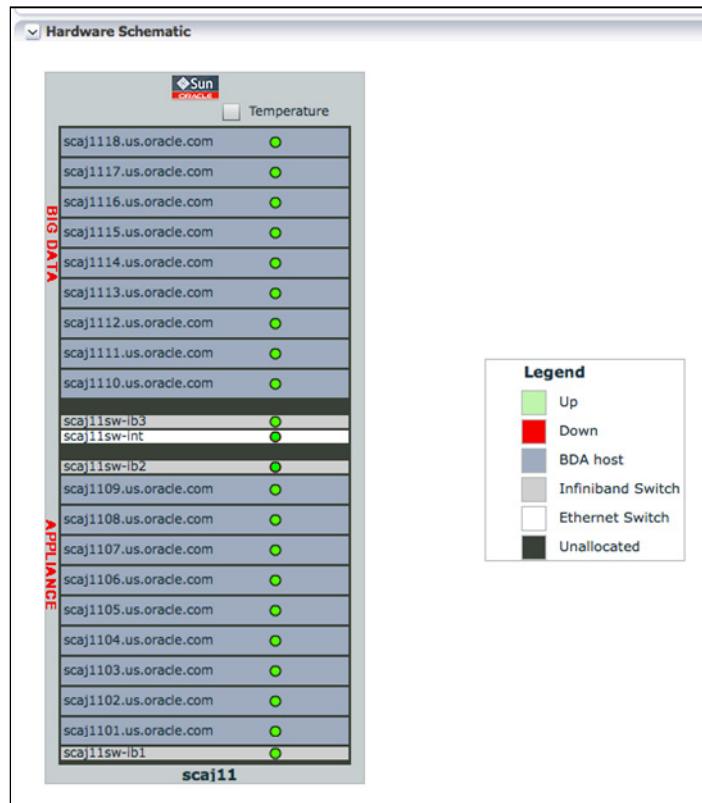


Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

You can access the features and functions of Oracle ILOM by using either of two supported interfaces:

- Browser-based web interface
- Command-line interface

Oracle ILOM: Integrated View



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

In Enterprise Manager, you can:

- Discover the components of a Big Data Appliance network and add them as managed targets
- Manage the hardware and software components that comprise a Big Data Appliance network as a single target or as individual targets
- Study collected metrics to analyze the performance of the network and each Big Data Appliance component
- Trigger alerts based on availability and system health
- Respond to warnings and incidents

Monitoring the Health of Oracle BDA: Management Utilities

- Monitoring the BDA
 - `setup-root-ssh`
 - `dcli`
 - `bdacli`
 - `mammoth`
 - Refer to the BDA User's Guide for additional utilities
- Monitor multiple clusters using Oracle Enterprise Manager
- Manage operations using Cloudera Manager
- Use Hadoop monitoring utilities



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Oracle ILOM consists of preinstalled, dedicated hardware and software that you can use to manage and monitor the servers and switches in a Big Data Appliance rack.

Important Utilities

- `setup-root-ssh`: Sets up password-less SSH for the `root` user for all the servers in a Big Data Appliance rack
- `dcli`: Executes commands across a group of servers on Big Data Appliance and returns the output
- `bdacheckib`: Checks the private InfiniBand network
- `bdachecknet`: Checks the network configuration
- `iblinkinfo`: Lists the connections in the InfiniBand network
- `listlinkup`: Identifies the active Ethernet ports
- `showvlan`: Lists the virtual local area networks (VLANs) configured on a Sun Network QDR InfiniBand Gateway Switch
- `showvnics`: Lists the virtual network interface cards (VNICS) created on a Sun Network QDR InfiniBand Gateway Switch
- `mammoth`: Used to install all end-user software onsite

For a complete list of the available utilities, see the *Oracle Big Data Appliance Owner's Guide Release 4 (4.0)* documentation reference.

Big Data Appliance: Security Implementation

- Hadoop supports the Kerberos network authentication protocol to prevent malicious impersonation.
- The present threat model of Hadoop assumes that users cannot:
 - Have root access to cluster machines
 - Have root access to shared client machines
 - Read or modify packets on the network of the cluster



ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The security features in CDH4 enable Hadoop to prevent malicious user attacks.

Kerberos is a computer network authentication protocol, which works on the basis of “tickets” to allow nodes communicating over a nonsecure network to prove their identity to one another in a secure manner. It was aimed primarily at a client/server model and it provides mutual authentication—both the user and the server verify each other's identity. Kerberos protocol messages are protected against eavesdropping and replay attacks.

You must install and configure Kerberos and set up a Kerberos Key Distribution Center and realm. You then configure CDH components to use Kerberos so that CDH can provide the following functionality:

- The CDH MasterNodes, NameNode, and JobTracker resolve the group name to avoid conflicts in group memberships.
- Map tasks run under the identity of the user who submitted the job.
- Authorization mechanisms in HDFS and MapReduce help control user access to data.

Note: Do not delete or modify the users that were created during installation because they are required for the software to operate.

Big Data Appliance: Usage Guidelines

- You cannot:
 - Customize any hardware except the administrative 48-port Cisco 4948 Ethernet switch
 - Update the firmware directly on Big Data Appliance servers. Instead, the update is done as an Oracle patch.
 - Use Cloudera Manager to move services from one server to another
- You can load additional software on Big Data Appliance servers.



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Note: For details about usage restrictions, see the following page:

<http://www.oracle.com/technetwork/server-storage/engineered-systems/bigdata-appliance/overview/index.html>

Summary

In this lesson, you should have learned how to:

- Use Oracle Big Data Appliance
- Identify the hardware and software components of Big Data Appliance



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Practice 25: Overview

In this practice, you gain a better understanding on Oracle Big Data Appliance by finding answers for questions related to working with Big Data Appliance.



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

26

Managing Oracle BDA

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Course Road Map

Module 1: Big Data Management System

Module 2: Data Acquisition and Storage

Module 3: Data Access and Processing

Module 4: Data Unification and Analysis

Module 5: Using and Managing Oracle Big Data Appliance

Lesson 25: Introduction to the Oracle Big Data Appliance (BDA)

Lesson 26: Managing Oracle BDA

Lesson 27: Balancing MapReduce Jobs

Lesson 28: Securing Your Data

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

This lesson provides the overview of monitoring and managing Oracle Big Data Appliance.

Objectives

After completing this lesson, you should be able to:

- Install the Oracle BDA software
- Identify the utilities available for monitoring Oracle BDA
- Monitor BDA by using Oracle Enterprise Manager
- Manage operations by using Cloudera Manager
- Use Hadoop monitoring utilities
- Use Cloudera Hue to interact with CDH
- Start and stop Oracle BDA



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Lesson Agenda

- Installing Oracle BDA software
- Monitoring Oracle BDA
- Monitoring BDA by using OEM
- Managing operations by using Cloudera Manager
- Using Hadoop monitoring utilities
- Using Cloudera Hue to interact with CDH
- Starting and stopping Oracle BDA



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Mammoth Utility

Mammoth is a command-line utility for installing and configuring the Oracle BDA software. By using Mammoth, you can:

- Set up a cluster for either CDH or Oracle NoSQL Database
- Create a cluster on one or more racks
- Create multiple clusters on an Oracle BDA rack
- Extend a cluster to new servers on the same or new rack
- Update a cluster with new software
- Configure or remove optional services Service Request
- Deploy or remove the Oracle Enterprise Manager system monitoring plug-in for Oracle BDA
- Apply one-off patch to your cluster



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

You can use the Mammoth utility to configure or remove optional services, including network encryption, disk encryption, Kerberos, Sentry, Oracle Audit Vault and Database Firewall, and Auto Service Request.

Before you install any software, you need to download the Mammoth bundle, which contains the installation files and the base image. Before you install the software, you must also use Oracle Big Data Appliance Configuration Generation Utility to generate the configuration files.

To download the Mammoth bundle, locate the download site in either My Oracle Support or Automated Release Updates (ARU).

You use the same Mammoth bundle for all procedures regardless of the rack size, and whether you are creating CDH or Oracle NoSQL Database clusters, or upgrading existing clusters.

For information about Oracle BDA Mammoth utility, see the Oracle Big Data Appliance Owner's Guide Release 4 (4.0) at

https://docs.oracle.com/cd/E55905_01/doc.40/e55796/mammoth.htm

Installation types

- Mammoth installs and configures the software on Oracle BDA by using the files generated by Oracle BDA Configuration Generation Utility.
- A cluster can be dedicated to either CDH (Hadoop) or Oracle NoSQL Database.



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

CDH Cluster

Mammoth installs and configures Cloudera's Distribution including Apache Hadoop. This includes all the Hadoop software and Cloudera Manager, which is the tool for administering your Hadoop cluster. If you have a license, Mammoth optionally installs and configures all components of Oracle Big Data Connectors.

NoSQL Cluster

Mammoth installs Oracle NoSQL Database. CDH and Oracle NoSQL Database do not share a cluster.

In addition to installing the software across all servers in the rack, Mammoth creates the required user accounts, starts the correct services, and sets the appropriate configuration parameters. When it is done, you have a fully functional, highly tuned, up and running Hadoop cluster.

Mammoth Code: Examples

Some of the Mammoth code examples are as follows:

- Display help for the Mammoth utility:
`./mammoth -h`
- Complete install:
`./mammoth -i bda`
- Post-install reconfig:
`./mammoth-reconfig add em`
`./mammoth-reconfig add asr`



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Some of the code examples of Mammoth are listed in the slide.

For information about Oracle BDA Mammoth utility, see the Oracle Big Data Appliance Owner's Guide Release 4 (4.0) at

https://docs.oracle.com/cd/E55905_01/doc.40/e55796/mammoth.htm

Mammoth Installation Steps

1. PreinstallChecks
2. SetupPuppet
3. PatchFactoryImage
4. CopyLicenseFiles
5. CopySoftwareSource
6. CreateLogicalVolumes
7. CreateUsers
8. SetupMountPoints
9. SetupMySQL
10. InstallHadoop
11. StartHadoopServices
12. InstallBDA Software
13. HadoopDataEncryption
14. SetupKerberos
15. SetupEMAgent
16. SetupASR
17. CleanupInstall
18. CleanupSSHroot
(Optional)



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Mammoth goes through the 18 steps mentioned in the slide while installing.

For information about Oracle BDA Mammoth utility, see the Oracle Big Data Appliance Owner's Guide Release 4 (4.0) at

https://docs.oracle.com/cd/E55905_01/doc.40/e55796/mammoth.htm

Following is a simple explanation of each of the steps.

- **Step 1 - PreinstallChecks:** Performs several tasks like validating the configuration file, setting up a secure shell, checking the network timings on all nodes, and so on
- **Step 2 - SetupPuppet:** Configures puppet agents on all nodes and starts them, and configures a puppet master on the node where the Mammoth is being run
- **Step 3 - PatchFactoryImage:** Installs the most recent Oracle Big Data Appliance image and system parameter settings

- **Step 4 - CopyLicenseFiles:** Copies third-party licenses to /opt/oss/src/OSSLicenses.pdf on every server, as required by the licensing agreements
- **Step 5 - CopySoftwareSource:** Copies third-party software source code to /opt/oss/src/ on every server, as required by the licensing agreements
- **Step 6 - CreateLogicalVolumes:** Mammoth does not create logical volumes for Oracle NoSQL Database clusters.
- **Step 7 - CreateUsers:** Creates the hdfs and mapred users, and the Hadoop group. It also creates the oracle user and the dba and oinstall groups.
- **Step 8 - SetupMountPoints:** The NameNode data is copied to multiple places to prevent a loss of this critical information should a failure occur in either the disk or the entire node where they are set up.
- **Step 9 - SetupMySQL:** Installs and configures MySQL Database. This step creates the primary database and several databases on node03 for use by Cloudera Manager. It also sets up replication of the primary database to a backup database on node02.
- **Step 10 - InstallHadoop:** Installs all packages in Cloudera's Distribution including Apache Hadoop (CDH) and Cloudera Manager. It then starts the Cloudera Manager server on node03 and configures the cluster.
- **Step 11 - StartHadoopServices:** Starts the agents on all nodes and starts all CDH services. After this step, you have a fully functional Hadoop installation.
- **Step 12 - InstallIBDASoftware (Optional):** Installs the server-side components of Oracle Big Data Connectors, if this option was selected in Oracle Big Data Appliance Configuration Generation Utility
- **Step 13 - HadoopDataEncryption:** Configures network and disk encryption
- **Step 14 - SetupKerberos:** Configures Kerberos authentication on Oracle Big Data Appliance, if this option was selected
- **Step 15 - SetupEMAgent (Optional):** Installs and configures the Oracle Enterprise Manager agents
- **Step 16 - SetupASR (Optional):** Installs and configures Auto Service Request (ASR)
- **Step 17 - CleanupInstall:** Changes the Cloudera Manager password if specified in the Installation Template. It deletes temporary files created during the installation. It copies log files.
- **Step 18 - CleanupSSHroot (Optional):** Removes password-less SSH for root that was set up in Step 1.

Lesson Agenda

- Installing Oracle BDA software
- Monitoring Oracle BDA
- Monitoring BDA by using OEM
- Managing operations by using Cloudera Manager
- Using Hadoop monitoring utilities
- Using Cloudera Hue to interact with CDH
- Starting and stopping Oracle BDA



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Monitoring Oracle BDA

There are lot of utilities available for monitoring the health of Oracle BDA. Most of these utilities are for monitoring the health of the hardware and the network.

| Utility | Description |
|--------------------|---|
| bdacli | Provides information about various configurations |
| setup-root-ssh | Establishes password-less SSH for the root user |
| bacheckhw | Checks the hardware profile of the server |
| bdachecknet | Checks whether the network configuration is working properly |
| bdadiag | Collects diagnostic information about an individual server for Oracle Support |
| bdaclustersynctime | Synchronizes the time of all servers in a cluster |



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

You can refer to the following link for details about all the utilities available.

https://docs.oracle.com/cd/E55905_01/doc.40/e55796/utilities.htm#BIGOG76779

Oracle BDA Command-Line Interface

The Oracle Big Data Appliance Command-Line Interface (`bdacli`) queries various configuration files to return information about the rack, cluster, server, and so on.

The `bdacli` utility can:

- Add and remove patches and optional services
- Can migrate critical services between critical nodes
- Add and remove servers from a cluster

Syntax:

```
bdacli action [parameters]
```



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The `bdacli` utility displays usage information if no parameters are included on the command line or the values are undefined.

The action element of the syntax can take various parameters as the following:

-help - Displays general usage information for `bdacli`, a list of actions, and a list of supported parameters for the getinfo action.

-{ add | remove } patch patch_number - Adds or removes a software patch on Oracle Big Data Appliance that matches `patch_number`. You must log in as `root` to use add or remove.

You can refer to the following link for further details about the action elements and `bdacli` at https://docs.oracle.com/cd/E55905_01/doc.40/e55796/utilities.htm#CJADBJAD

bdacli

The following commands provide information about the optional software on the cluster.

```
bdacli getinfo cluster_bdc_installed  
bdacli getinfo cluster_odi_version
```

The following command lists all switches on the current InfiBand fabric.

```
bdacli getinfo ib_switches
```

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

-getinfo - Returns the information about the system component based on the parameter passed

setup-root-ssh

The **setup-root-ssh** utility establishes password-less SSH for the root user.

Syntax:

```
setup-root-ssh [-C | -c | -g | -j] [-p]
setup-root-ssh -h
```



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The **setup-root-ssh** command takes the following parameters:

-C - Targets all servers in the cluster, using the list of servers in /opt/oracle/bda/cluster-hosts-infiniband

-c host1, host2, ... - Targets the servers specified as host1, host2, and so forth, on the command line

-g groupfile - Targets a user-defined set of servers listed in groupfile. You can enter either server names or IP addresses in the file, one per line.

-j "eth0_1ps [range]" - Specifies the range of servers in a starter rack [1-6] or a starter rack and expansion kit [1-12]. This parameter is required in the 2.2.x base image when the utility is used before network configuration.

-h - Displays Help

-p password - Specifies the root password on the command line. Oracle recommends that you omit this parameter. You will be prompted to enter the password, which the utility does not display on your screen.

Lesson Agenda

- Installing Oracle BDA software
- Monitoring Oracle BDA
- **Monitoring BDA by using OEM**
- Managing operations by using Cloudera Manager
- Using Hadoop monitoring utilities
- Using Cloudera Hue to interact with CDH
- Starting and stopping Oracle BDA

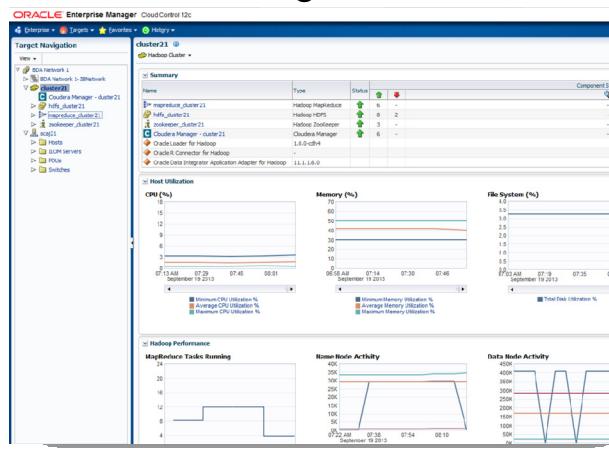


Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Monitor BDA with Oracle Enterprise Manager

Oracle Enterprise manager enables:

- Automatic discovery of BDA deployment
- Monitoring health, availability, and performance of hardware and Hadoop services
- Consolidated incident management



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

It is a single point of control for your entire stack—be it for monitoring health and availability, managing performance or configuration management, or automating lifecycle operations such as provisioning and patching.

OEM: Web and Command-Line Interfaces

1 Web interface

2 Command-line interface

% emcli
emcli>

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

ORACLE

After you log in and select a target cluster in the Oracle Enterprise Manager web interface, you can drill down into the following primary areas:

- **InfiniBand network:** Network topology and status for InfiniBand switches and ports
- **Hadoop cluster:** Software services for HDFS, MapReduce, and ZooKeeper
- **Oracle Big Data Appliance rack:** Hardware status including server hosts, Oracle Integrated Lights Out Manager (Oracle ILOM) servers, power distribution units (PDUs), and the Ethernet switch

For additional information about installing the *Oracle Enterprise Manager System Monitoring* plug-in, see *Oracle Enterprise Manager System Monitoring Plug-in Installation Guide for Oracle Big Data Appliance* for installation instructions and use cases documentation reference.

The Enterprise Manager command-line interface (EM CLI) is also installed on Oracle Big Data Appliance along with all the other software.

To start EM CLI in Interactive mode in Linux, type `emcli` at the command prompt to start an Interactive shell as shown in the second example in the slide.

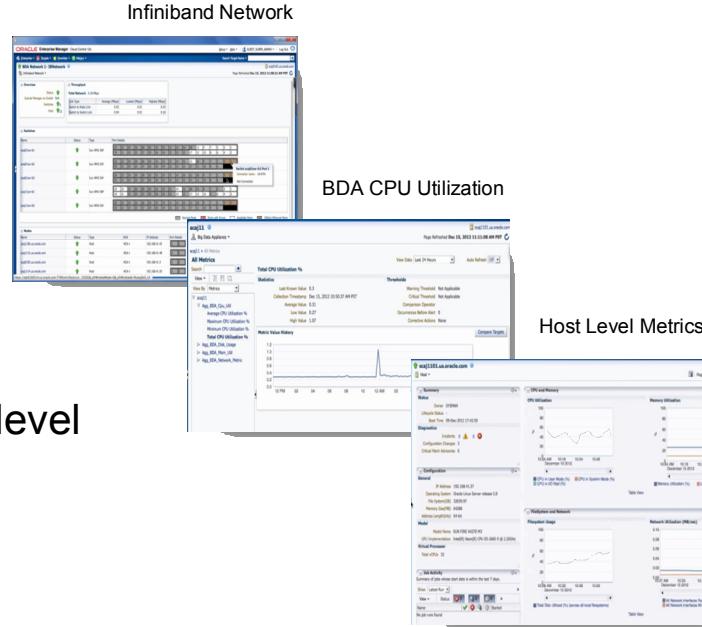
For additional information about the EM CLI, see the *Oracle Enterprise Manager Command Line Interface 12c Release 1 (12.1.0.4)* documentation reference.

OEM: Hardware Monitoring

Monitor every hardware component:

- Hosts
- ILOM Servers
- PDUs
- Switches
- Disks

View summary or detail level metrics.



ORACLE

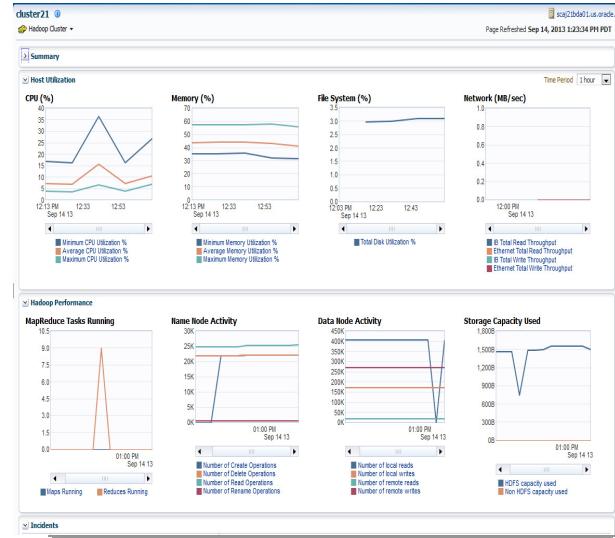
Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Hadoop Cluster Monitoring

Monitor system performance across Hadoop cluster.

Review trends in HDFS storage, MapReduce activity, and host utilization.

- In-context drill down to Cloudera Manager for detailed analysis



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Lesson Agenda

- Installing Oracle BDA software
- Monitoring Oracle BDA
- Monitoring BDA by using OEM
- Managing operations by using Cloudera Manager
- Using Hadoop monitoring utilities
- Using Cloudera Hue to interact with CDH
- Starting and stopping Oracle BDA



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Managing CDH Operations

- Cloudera Manager is installed on Oracle BDA to perform the operations of Cloudera's Distribution including Hadoop (CDH).
- You can use Cloudera Manager to perform the following administrative tasks:
 - Monitor jobs and services.
 - Start and stop services.
 - Manage security and Kerberos credentials.
 - Monitor user activity.
 - Monitor the health of the system.
 - Monitor performance metrics.
 - Track hardware use (disk, CPU, and RAM).



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Cloudera Manager provides a single administrative interface to all Oracle Big Data Appliance servers that are configured as part of the Hadoop cluster.

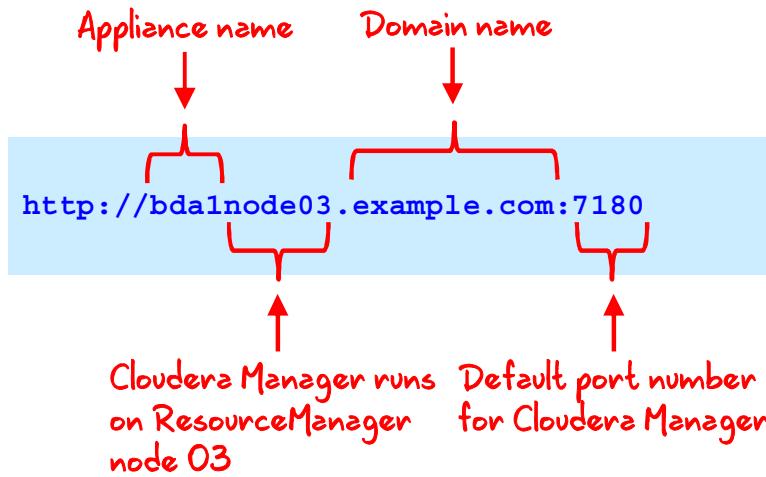
Refer to the following links for further information about the Cloudera Manager.

http://docs.oracle.com/cd/E55905_01/doc.40/e55814/admin.htm#BIGUG132

<http://www.cloudera.com/content/cloudera/en/documentation/cloudera-manager/v4-latest/Cloudera-Manager-Introduction/Cloudera-Manager-Introduction.html>

Using Cloudera Manager

- To use Cloudera Manager:
 - Open a browser and enter a URL as follows:



- Log in with a username and password for Cloudera Manager.

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

To use Cloudera Manager, enter a URL (as shown in the slide). In this example, bda1 is the name of the appliance, node03 is the name of the server, example.com is the domain, and 7180 is the default port number for Cloudera Manager.

Note: Only a user with administrative privileges can change the settings.

Monitoring Oracle BDA Status

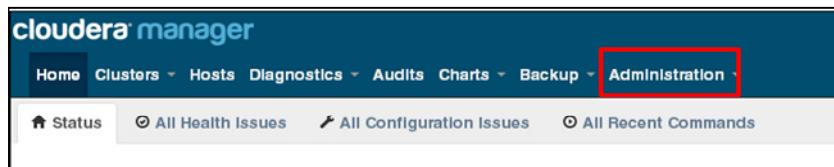
The screenshot shows the Cloudera Manager interface. At the top, there's a navigation bar with links: Home, Clusters, Hosts, Diagnostics, Audits, Charts, Backup, and Administration. The 'Home' link is highlighted with a red box. Below the navigation bar, the page title is 'Home'. It displays the status of various services: 'scaja (CDH 5.0.0, Packages)' with 1 error; 'Hosts' (green), 'Solr' (green), 'Spark' (green), 'hdfs' (green), 'hive' (orange, 1 error), 'hue' (green), 'oozie' (green), 'yarn' (green), and 'zookeeper' (green). Under 'Cloudera Management Service', it shows 'mgmt' (green). On the right, there are two charts: 'Cluster CPU' (percent) which shows usage around 10-20% over time, and 'Cluster Disk IO' (bytes / second) which shows activity between 1.9M/s and 2.9M/s. A red banner at the bottom contains the text 'Copyright © 2015, Oracle and/or its affiliates. All rights reserved.' and the 'ORACLE' logo.

In Cloudera Manager, you can choose any of the following pages from the menu bar at the top of the display:

- **Home:** Provides a graphic overview of activities and links to all services controlled by Cloudera Manager
- **Clusters:** Accesses the services on multiple clusters
- **Hosts:** Monitors the health, disk usage, load, physical memory, swap space, and other statistics for all servers in the cluster
- **Diagnostics:** Accesses events and logs. Cloudera Manager collects historical information about the systems and services. You can search for a particular phrase for a selected server, service, and time period. You can also select the minimum severity level of the logged messages included in the search: TRACE, DEBUG, INFO, WARN, ERROR, or FATAL.
- **Audits:** Displays the audit history log for a selected time range. You can filter the results by username, service, or other criteria and then download the log as a CSV file.
- **Charts:** Enables you to view metrics from the Cloudera Manager time-series data store in a variety of chart types, such as line and bar
- **Backup:** Accesses snapshot policies and scheduled replications
- **Administration:** Provides a variety of administrative options, including Settings, Alerts, Users, and Kerberos

Performing Administrative Tasks

- As a Cloudera Manager administrator, you can add users, set up Kerberos security, and change properties for monitoring the health and use of Oracle BDA.
- To access Cloudera Manager Administration:
 1. Log in to Cloudera Manager with administrative privileges
 2. Select the Administration tab



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

As a Cloudera Manager administrator, you can change various properties for monitoring the health and use of Oracle Big Data Appliance, add users, and set up Kerberos security.

Managing Services

You can use the Cloudera Manager interface to manage, configure, start, and stop the following services:

- HDFS
- Hive
- Hue
- Impala
- Oozie
- YARN
- ZooKeeper



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

You can use Cloudera Manager to change the configuration of the services listed in the slide and to stop and restart them. Before you can use additional services, you must configure them.

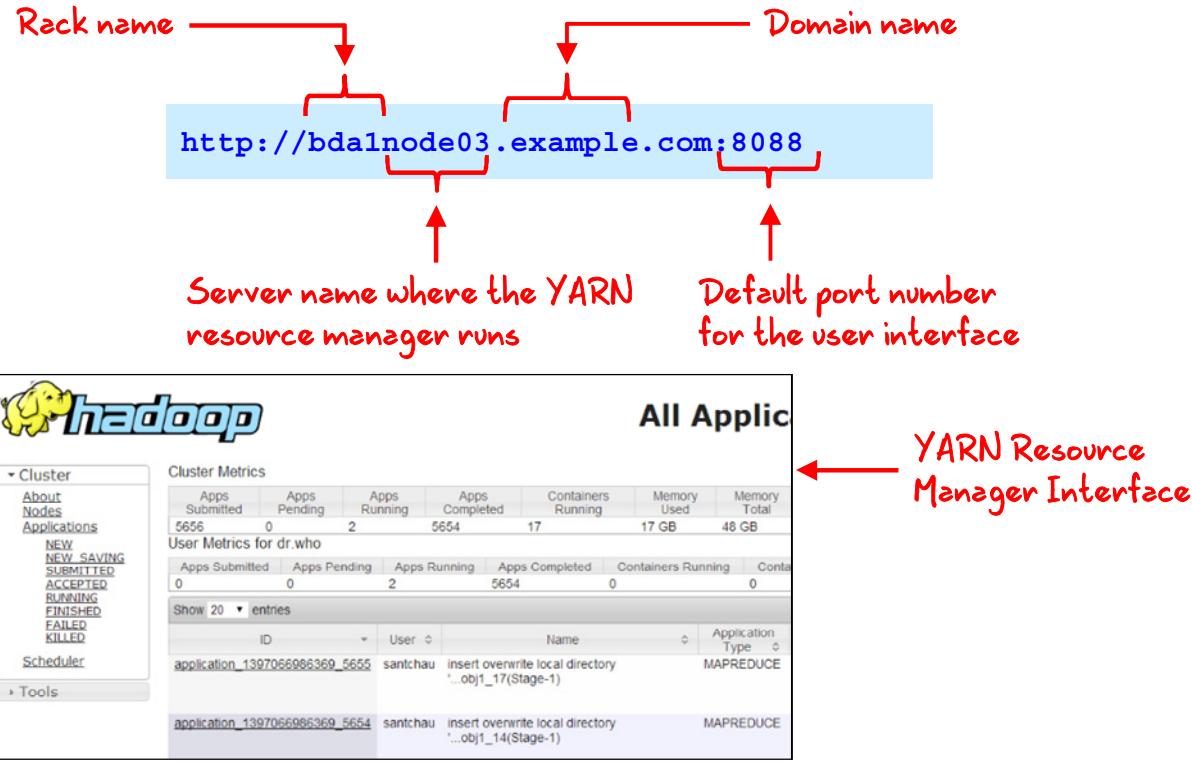
Lesson Agenda

- Installing Oracle BDA software
- Monitoring Oracle BDA
- Monitoring BDA by using OEM
- Managing operations by using Cloudera Manager
- **Using Hadoop monitoring utilities**
- Using Cloudera Hue to interact with CDH
- Starting and stopping Oracle BDA



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Monitoring MapReduce Jobs



ORACLE

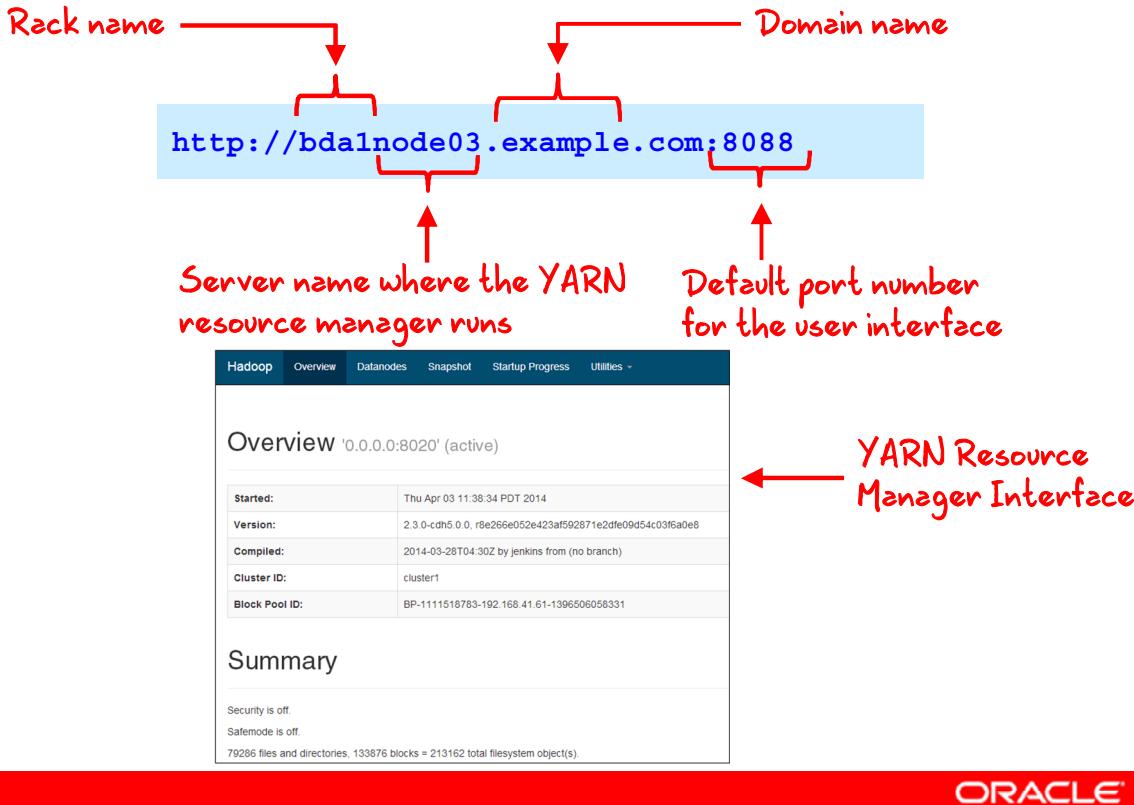
Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

You can use the native Hadoop utilities to monitor MapReduce jobs and Hadoop Distributed File Systems (HDFSs). These utilities are read-only and do not require authentication.

Cloudera Manager provides an easy way to obtain the correct URLs for these utilities. You simply expand the Web UI submenu on the YARN service page.

You can monitor MapReduce jobs by using the resource manager interface (as shown in the slide). To monitor MapReduce jobs, open your web browser and enter a URL similar to the one shown in the slide. In the slide example, bda1 is the name of the rack, node03 is the name of the server where the YARN resource manager runs, and 8088 is the default port number for the user interface.

Monitoring the Health of HDFS



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

You can monitor the health of the Hadoop file system by using the DFS health utility on the first two nodes of a cluster. To monitor HDFS, open your web browser and enter a URL similar to the one shown in the slide example. In the slide example, bda1 is the name of the rack, node01 is the name of the server where the DFS health utility runs, and 50070 is the default port number for the user interface.

Lesson Agenda

- Installing Oracle BDA software
- Monitoring Oracle BDA
- Monitoring BDA by using OEM
- Managing operations by using Cloudera Manager
- Using Hadoop monitoring utilities
- **Using Cloudera Hue to interact with CDH**
- Starting and stopping Oracle BDA



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Cloudera Hue

Cloudera Hue helps to:

- Interact with Hadoop
- Query Hive data stores
- Create, load, and delete Hive tables
- Work with HDFS files and directories
- Create, submit, and monitor MapReduce jobs
- Monitor MapReduce jobs
- Create, edit, and submit workflows by using the Oozie dashboard
- Work with Impala queries and Solr
- Manage users and groups



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Hue runs in a browser and provides an easy-to-use interface for several applications to support interaction with Hadoop and HDFS. You can use Hue to perform the tasks shown in the slide.

Hive Query Editor (Hue) Interface

The screenshot shows the Hue Query Editor interface. At the top, there's a navigation bar with links for Home, Query Editors, Data Browsers, Workflows, and Search. Below the navigation bar, the main header includes 'Hive Editor' and 'Query Editor' tabs, along with links for My Queries, Saved Queries, and History. On the left side, there's a sidebar titled 'Navigator' with a 'Settings' section and a 'DATA...' dropdown set to 'default'. A list of tables is shown under 'Table name...', including 'salaries', 'csvtab', 'airq', 'tab1', 'orehvtm...', 'cars_tab', 'cars_seq', and 'trees1'. The main content area contains a text input field with a placeholder 'Example: SELECT * FROM tablename, or press CTRL + space'. Below the input field are buttons for 'Execute', 'Save as...', 'Explain', and 'New query'. At the bottom of the main area, there's a 'Recent queries' section with tabs for 'Recent queries', 'Query', 'Log', 'Columns', 'Results', and 'Chart'. The 'Recent queries' tab is selected, showing a table with columns 'Time' and 'Query'. The message 'No data available' is displayed. A red footer bar at the bottom contains the 'ORACLE' logo and the copyright notice 'Copyright © 2015, Oracle and/or its affiliates. All rights reserved.'

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Hue allows the developers to type and execute HiveQL queries by using the query editor. It also supports Impala queries and Solr.

Logging in to Hue

1. Log in to Cloudera Manager and click the Hue service on the Home page.
 2. On the Hue page, click Hue Web UI.
 3. The following URL is an example:
- `http://bdalnode03.example.com:8888`
4. Log in with your Hue credentials.



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Oracle Big Data Appliance is not initially configured with Hue user accounts. The first user who connects to Hue can log in with any username and password. That user automatically becomes an administrator and can then create other user and administrator accounts.

Lesson Agenda

- Installing Oracle BDA software
- Monitoring Oracle BDA
- Monitoring BDA by using OEM
- Managing operations by using Cloudera Manager
- Using Hadoop monitoring utilities
- Using Cloudera Hue to interact with CDH
- Starting and stopping Oracle BDA



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Starting Oracle BDA

You start Oracle Big Data Appliance as follows:

1. Power up Oracle Big Data Appliance
2. Start the HDFS software services
3. Start Oracle Data Integrator Agent



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

To restart Oracle Big Data Appliance, you must have root access. Password-less SSH must be set up on the cluster so that you can use the `dcli` utility.

To complete each of the tasks listed in the slide, you must perform many subtasks.

For details, refer to the documentation:

http://docs.oracle.com/cd/E55905_01/doc.40/e55814/admin.htm#CACCEJID

Stopping Oracle BDA

To stop Oracle Big Data Appliance:

1. Stop all managed services
2. Stop Cloudera Manager Server
3. Stop Oracle Data Integrator Agent
4. Dismount NFS directories
5. Stop the servers
6. Stop the InfiniBand and Cisco switches



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

To stop and restart Oracle Big Data Appliance, you must have root access. Password-less SSH must be set up on the cluster so that you can use the `dcli` utility.

To complete each of the tasks listed in the slide, you must perform many subtasks.

For details, refer to the documentation:

http://docs.oracle.com/cd/E55905_01/doc.40/e55814/admin.htm#CACCEJID

BDA Port Assignments

| Port | Used By |
|--------------|--------------------------------------|
| 22 | ssh |
| 80 | yumrepos (only during installation) |
| 111 | portmap |
| 668 | rpc.statd |
| 3306 | MySQL Database |
| 5000 | Oracle NoSQL Database registration |
| 5001 | Oracle NoSQL Database administration |
| 5010 to 5020 | Oracle NoSQL Database processes |
| 6481 | xinetd (service tag) |
| 8139 | Puppet nodes |
| 8140 | Puppet parent |
| 20910 | Oracle Data Integrator agent |
| 30920 | Automated Service Monitor (ASM) |



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The table in the slide identifies the port numbers that are used by BDA software. These ports are typically available when the network is configured.

For more information about port configuration, see the following page:

http://www.cloudera.com/content/cloudera-content/cloudera-docs/CM5/latest/Cloudera-Manager-Installation-Guide/cm5ig_config_ports.html

Summary

In this lesson, you should have learned how to:

- Install the Oracle BDA software
- Identify the utilities available for monitoring Oracle BDA
- Monitor BDA by using Oracle Enterprise Manager
- Manage operations by using Cloudera Manager
- Use Hadoop monitoring utilities
- Use Cloudera Hue to interact with CDH
- Start and stop Oracle BDA



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Practice 26: Overview

In this practice, you will examine the ways to:

- Monitor the map reduce jobs
- Monitor the health of HDFS
- Use the Hive Query Editor (Hue)



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

27

Balancing MapReduce Jobs

ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Course Road Map

Module 1: Big Data Management System

Module 2: Data Acquisition and Storage

Module 3: Data Access and Processing

Module 4: Data Unification and Analysis

Module 5: Using and Managing Oracle Big Data Appliance

Lesson 25: Introduction to the Oracle Big Data Appliance (BDA)

Lesson 26: Managing Oracle BDA

Lesson 27: Balancing MapReduce Jobs

Lesson 28: Securing Your Data

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

This lesson introduces the Perfect Balance feature of Oracle BDA that you can use to balance MapReduce jobs.

Objectives

After completing this lesson, you should be able to:

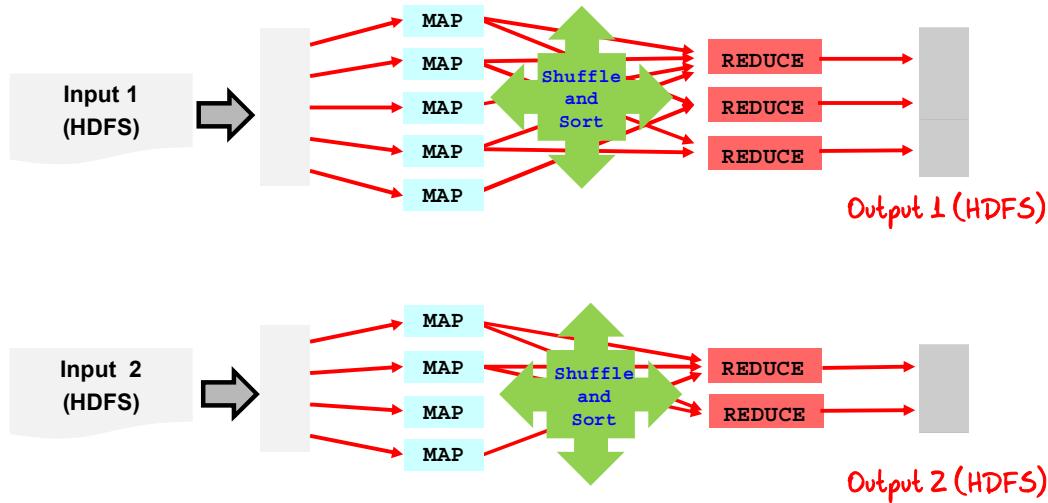
- Define the Perfect Balance feature of Oracle BDA
- Use Perfect Balance to balance MapReduce jobs
- Run Job Analyzer as a stand-alone utility or with Perfect Balance
- Identify, locate, and read generated reports
- Collect additional metrics with Job Analyzer
- Configure Perfect Balance
- Use chopping (partitioning of values)
- Troubleshoot jobs running with Perfect Balance
- Use the Perfect Balance examples



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Ideal World: Neatly Balanced MapReduce Jobs

Reduce nodes process same number of rows.



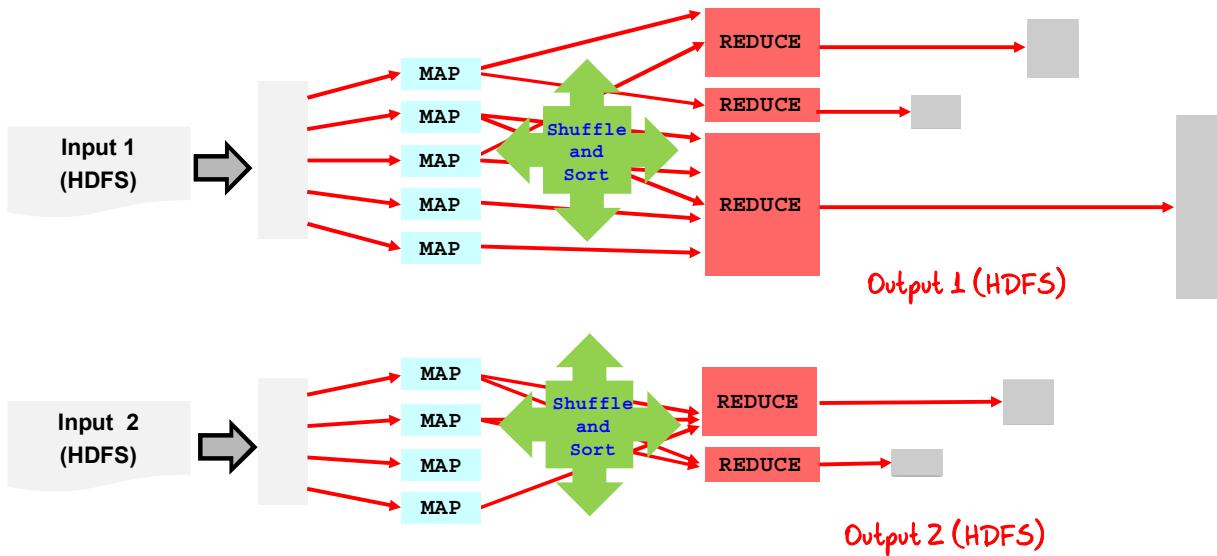
ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

In the scenario illustrated in the slide, the reducers process the same number of rows, so the work is evenly distributed across all reducer processes.

Real World: Skewed Data and Unbalanced Jobs

Reduce nodes process varying number of rows.



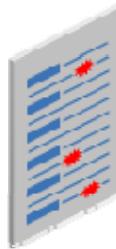
ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

In the real world, data is skewed. Some reduce processes will do more work than others, because they will process more rows than other reduce processes. These long-running reduce tasks can slow down the entire job. As shown in the slide, reduce nodes process a varying number of rows and slow down the entire job, because a job is only as fast as the slowest reducer.

Data Skew

- In a MapReduce job:
 - The map function is applied to every input record, and
 - The reduce function is applied to records output from map jobs and grouped by intermediate keys.
- Data skew is an imbalance in the loads assigned to the reduce tasks. The load is a function of:
 - The number of keys assigned to a reducer
 - The number of records and bytes in the values for each key



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Perfect Balance addresses the problem of data skew. Execution time depends on the size of data input. Data skew is an imbalance in the load that is assigned to different reduce tasks. The load is a function of:

- The number of keys assigned to a reducer
- The number of records and bytes in the values for each key

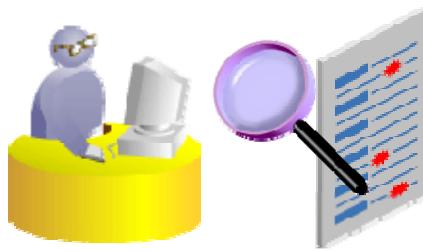
What is a key?

Every record in the data can be represented by a key. An example: Consider words sorted alphabetically in a dictionary. All words beginning with A have the key “A,” all words beginning with B have the key “B,” and so on.

Output of map jobs is organized by a key, which is the “intermediate key” referred to in the slide. Records that have the same key are sent to a reducer, and this results in skew when the data is not evenly distributed according to the keys.

Data Skew Can Slow Down the Entire Hadoop Job

| Data to be Analyzed | Skew Causes |
|---|-----------------------------------|
| Online retail activity | Peak hours, holiday season |
| Geological sensors | Seismic activities |
| Network traffic to detect threat, misuse, anomaly | Peak hours |
| Movie watching trends | Peak hours, holidays, bad weather |

The Oracle logo, consisting of the word "ORACLE" in a bold, white, sans-serif font.

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The key for records on online retail activity can be the date. Activity spikes during certain shopping seasons. Similarly, seismic activity spikes during some timestamps (corresponding to geological events), and the data volumes are much higher than at other times.

Perfect Balance

The Perfect Balance is a feature of Oracle Big Data Appliance:

- Optimizes MapReduce jobs for faster performance
- Distributes work evenly across reducers, resulting in greatly reduced elapsed times
- Works transparently
- Is a core foundational feature
 - Is the basis for improved performance of all MapReduce jobs on Oracle Big Data Appliance



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

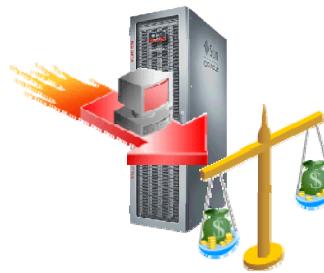
The Perfect Balance feature distributes the reducer load in a MapReduce application so that each reduce task does approximately the same amount of work. Although the default Hadoop method of distributing the reduce load is appropriate for many jobs, it does not distribute the load evenly for jobs that have significant data skew.

The total run time for a job is extended, to varying degrees, by the time that the reducer with the greatest load takes to finish the job. In jobs with a skewed load, some reducers complete the job quickly, while others take much longer. Perfect Balance can significantly reduce the total run time by distributing the load evenly, enabling all reducers to finish at about the same time.

Perfect Balance works with both MapReduce jobs on both YARN and MRv1 clusters.

How Does the Perfect Balance Work?

- Sampling
 - Determines data distribution before job execution
- Chopping
 - Sub-partitions values of a single reduce key
- Bin packing
 - Combines sub-partitions of multiple reduce keys into one reducer



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

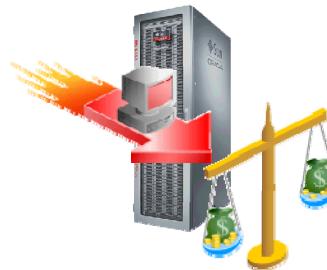
Perfect Balance distributes the load evenly across reducers by first sampling the data, optionally chopping large keys into two or more smaller keys, and using a load-aware partitioning strategy to assign keys to reduce tasks.

“Smart sampling” of Perfect Balance detects data skew before running a job. Detecting data skew is difficult; input data must be sampled to determine the data skew accurately. Smart sampling randomly selects data from many input splits, applies mapper functions, and dynamically determines when enough data has been sampled.

Using Perfect Balance

There are two ways to use this feature:

- Perfect Balance
 - Run a job without changing your application code by properly configuring Perfect Balance
 - This is the preferred method, and it is appropriate for most jobs
- Perfect Balance API
 - Add the Perfect Balance code to your application code



ORACLE

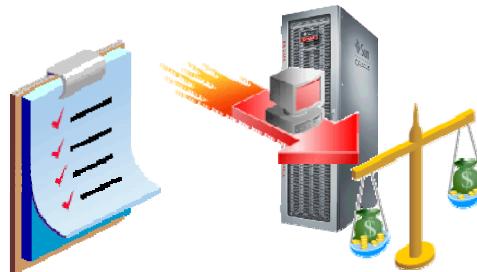
Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Perfect Balance API

Use this method when your application setup code must run before using Perfect Balance (typically application setup code is in the job driver which runs before the mappers and reducers). Otherwise, you must use the first method, which requires no change to your code.

Application Requirements for Using Perfect Balance

- To get the maximum benefit from Perfect Balance, your application must meet the following requirement:
 - The job is distributive, so that splitting a group of records associated with a reduce key does not produce incorrect results for the application
- If a job is not distributive, it can use Perfect Balance with bin packing but not chopping.



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

To balance a load, Perfect Balance subpartitions the values of large reduce keys and sends each subpartition to a different reducer. This distribution contrasts with the standard Hadoop practice of sending all values for a single reduce key to the same reducer. Your application must be able to handle output from the reducers that is not fully aggregated, so that it does not produce incorrect results. This partitioning of values is called chopping. Applications that support chopping have distributive reduce functions.

If your application is not distributive:

- You can still run Perfect Balance after you disable the chopping feature
- The job still benefits from using Perfect Balance, but the load is not as evenly balanced as with chopping

Note: BDA 4.0 release does not support combiners. Perfect Balance detects the presence of combiners and does not balance when they are present.

Perfect Balance: Benefits

- Faster completion of balanced Hadoop jobs
- Easy integration with MapReduce jobs
- Fully automated
- Low overhead
- No need to redesign MapReduce applications



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Using Job Analyzer

- You can use Job Analyzer to decide whether a job is a candidate for load balancing with Perfect Balance.
- Job Analyzer uses the output logs of a MapReduce job to generate a simple report with statistics such as:
 - Elapsed time
 - Load for each reduce task
- If the report shows that the data is skewed, then the application is a good candidate for Perfect Balance.



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

If the report shows that the data is skewed (that is, the reducers processed very different loads and the run times varied widely), then the application is a good candidate for Perfect Balance.

Getting Started with Perfect Balance

You can use Perfect Balance as follows:

1. Ensure that your application meets the requirements.
2. Log in to the server where you will submit the job.
3. Run the examples provided with Perfect Balance to become familiar with the product.
4. Set the following variables using the Bash export command:
 - BALANCER_HOME
 - HADOOP_CLASSPATH
 - HADOOP_USER_CLASSPATH_FIRST
5. Run Job Analyzer *without* the balancer and use the generated report to decide whether your job is a good candidate for using Perfect Balance (explained earlier).



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Getting Started with Perfect Balance

6. Decide which configuration properties to set, and then:
 - Create a configuration file, or
 - Specify individual settings with the `hadoop` command
7. Run the job by using Perfect Balance.
8. Use the Job Analyzer report to evaluate the effectiveness of using Perfect Balance. (explained later)
9. Modify the job configuration properties as desired before re-running the job with Perfect Balance.



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Using Job Analyzer

- Run as a stand-alone utility:
 - Job Analyzer runs against existing job output logs
 - Use this method to analyze a job that previously ran
- Run with Perfect Balance for the current job:
 - Use this method when you want to analyze the currently running job



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

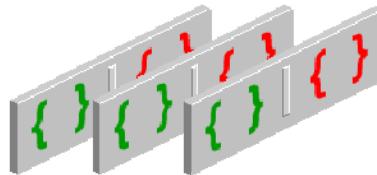
Environmental Setup for Perfect Balance and Job Analyzer

Set the following variables by using the Bash export command:

- BALANCER_HOME
- HADOOP_CLASSPATH
- HADOOP_USER_CLASSPATH_FIRST

```
BALANCER_HOME=/opt/oracle/orabalancer-2.0.0-h2
HADOOP_CLASSPATH=${BALANCER_HOME}/jlib/orabalancer-
2.0.0.jar:${BALANCER_HOME}/jlib/commons-math-
2.2.jar:$HADOOP_CLASSPATH

HADOOP_USER_CLASSPATH_FIRST=true
```



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

This is the environmental setup for running Perfect Balance or Job Analyzer.

Running Job Analyzer as a Stand-alone Utility to Measure Data Skew in Unbalanced Jobs

As a stand-alone utility, Job Analyzer provides a quick way to analyze the reduce load of a previously run job. In a YARN cluster:

1. Log in to the server where you run Job Analyzer.
2. Locate the job ID for the job to analyze.
3. Run the Job Analyzer utility:

```
hadoop jar orabalancer.jar  
    oracle.hadoop.balancer.tools.JobAnalyzer \  
-D oracle.hadoop.balancer.application_id=job_number \  
[ja_report_path]
```

4. View the Job Analyzer report in your web browser.



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The `job_number` in the syntax in the slide represents the application ID previously assigned to the job. Substitute the `job_number` with the actual job number of a previously run job such as `job_1396563311211_0947`. This is for YARN clusters. For MRv1 clusters, set `mapred.output.dir` to the output directory of the job you want to analyze.

The `ja_report_path` in the syntax in the slide represents an HDFS directory where the Job Analyzer creates its report (optional). The default directory is `<job_output_dir>/_balancer` and `<job_output_dir>` is the output directory used by the job.

Using Job Analyzer as a Stand-Alone Utility: Example with a YARN Cluster

```
hadoop jar orabalancer.jar oracle.hadoop.balancer.tools.JobAnalyzer \
-D oracle.hadoop.balancer.application_id=job_1396563311211_0947

$ sh ./runja.sh
$ hadoop fs -get jdoe_nobal_outdir/_balancer/jobanalyzer-report.html /home/jdoe
$ cd /home/jdoe
$ firefox jobanalyzer-report.html
```

The contents of the `runja.sh` script:

```
BALANCER_HOME=/opt/oracle/orabalancer-2.0.0-h2
export HADOOP_CLASSPATH=${BALANCER_HOME}/jlib/orabalancer-
2.0.0.jar:${BALANCER_HOME}/jlib/commons-math-2.2.jar:$HADOOP_CLASSPATH

export HADOOP_USER_CLASSPATH_FIRST=true

hadoop jar orabalancer.jar \
oracle.hadoop.balancer.tools.JobAnalyzer -D \
oracle.hadoop.balancer.application_id=job_1396563311211_0947
```



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The slide example runs the `runja.sh` script that sets the required variables, uses the MapReduce job logs for a job with an application ID of `job_1396563311211_0947`, and then creates the report in the default folder `_balancer` in the specified output directory: `jdoe_nobal_outdir/_balancer`. It then copies the HTML version of the report from HDFS to the `/home/jdoe` local directory and opens the report in a Firefox web browser.

If you want to run this example in YARN, replace the application ID with the application ID of the job. The application ID of the job looks like this example: `job_1396563311211_0947`. We will discuss reading a Job Analyzer report later in this lesson.

Configuring Perfect Balance

- Perfect Balance uses the standard Hadoop methods of specifying configuration properties on the command line.
- You can use the:
 - `-conf` option to identify a configuration file, or
 - `-D` option to specify individual properties
- All Perfect Balance configuration properties have default values; therefore, setting them is optional. Examples:
 - `oracle.hadoop.balancer.confidence`
 - `oracle.hadoop.balancer.enableSorting`
- Refer to **The Perfect Balance Configuration Property Reference** in the Oracle BDA Software User's Guide for a full list of properties.

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Using Perfect Balance to Run a Balanced MapReduce Job

To run a job with Perfect Balance:

1. Log in to the server where you will submit the job.
2. Set up Perfect Balance by following the steps in "Getting Started with Perfect Balance."
3. Configure the job with these Perfect Balance properties:
 - To enable balancing, set `oracle.hadoop.balancer.autoBalance` to `true`.
 - When `oracle.hadoop.balancer.autoBalance` is set to `true`, `oracle.hadoop.balancer.autoAnalyze` is automatically set to `BASIC_REPORT`. So Job Analyzer is always run when using Perfect Balance
 - To allow Job Analyzer to collect additional metrics, set `oracle.hadoop.balancer.autoAnalyze` to `REDUCER_REPORT`



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

You can provide Perfect Balance configuration properties either on the command line or in a configuration file. You can also combine Perfect Balance properties and MapReduce properties in the same configuration file. See "About Configuring Perfect Balance" in the Oracle Big Data Appliance Software User's Guide Release 4 (4.0) documentation reference.

These steps use Perfect Balance with a job by setting environmental variables and configuration properties. There are no changes to the code of your MapReduce application.

Note that setting `oracle.hadoop.balancer.autoBalance` to `true` automatically sets `oracle.hadoop.balancer.autoAnalyze` to `BASIC_REPORT`, to include running the Job Analyzer. With a setting of `REDUCER_REPORT` additional metrics are gathered.

Using Perfect Balance to Run a Balanced MapReduce Job

4. Set any additional configuration properties.
5. Run your job, using the following syntax:

```
bin/hadoop jar application_jarfile.jar ApplicationClass\  
-D application_config_property \  
-D oracle.hadoop.balancer.autoBalance=true \  
-D other_perfect_balance_config_property \  
-conf application_config_file.xml \  
-conf perfect_balance_config_file.xml
```



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Running a Job Using Perfect Balance: Examples

```
$ cat pb_balance.sh
```

Sets up Perfect Balance for a job, and then runs the job.

```
#setup perfect balance as described in Getting Started with Perfect Balance
BALANCER_HOME=/opt/oracle/orabalancer-2.0.0-h2
export HADOOP_CLASSPATH=${BALANCER_HOME}/jlib/orabalancerclient-2.0.0.jar:${BALANCER_HOME}
export HADOOP_USER_CLASSPATH_FIRST=true

# setup optional properties like java heap size and garbage collector
export HADOOP_CLIENT_OPTS="-Xmx1024M ${HADOOP_CLIENT_OPTS}"

# run the job with balancing and job analyzer enabled
hadoop jar application_jarfile.jarApplicationClass
-D application_config_property \
-D mapreduce.input.fileinputformat.inputdir=jdoe_application/input \
-D mapreduce.output.fileoutputformat.outputdir=jdoe_outdir \
-D mapreduce.job.name="autoinvoke" \
-D mapreduce.job.reduces=10 \
-D oracle.hadoop.balancer.autoBalance=true \
-D oracle.hadoop.balancer.autoAnalyze=REDUCER_REPORT \
-D oracle.hadoop.balancer.linearKeyLoad.keyWeight=93.98 \
-D oracle.hadoop.balancer.linearKeyLoad.rowWeight=0.001126 \
-D oracle.hadoop.balancer.linearKeyLoad.byteWeight=0.0 \
-conf application_config_file.xml
```



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The screen capture in the slide displays the contents of the pb_balance.sh.

Running a Job Using Perfect Balance: Examples

```
$ sh ./pb_balance.sh
14/04/14 14:59:42 INFO balancer.Balancer: Creating balancer
14/04/14 14:59:42 INFO balancer.Balancer: Starting Balancer
14/04/14 14:59:43 INFO input.FileInputFormat: Total input paths to process : 5
14/04/14 14:59:46 INFO balancer.Balancer: Balancer completed
14/04/14 14:59:47 INFO input.FileInputFormat: Total input paths to process : 5
14/04/14 14:59:47 INFO mapreduce.JobSubmitter: number of splits:5
14/04/14 14:59:47 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1397066986369_3500
14/04/14 14:59:47 INFO impl.YarnClientImpl: Submitted application application_1397066986369_3500
14/04/14 14:59:47 INFO mapreduce.Job: The url to track the job:
.
.
.
Map-Reduce Framework
Map input records=1000000
Map output records=20000000
Map output bytes=872652976
Map output materialized bytes=175650573
Input split bytes=580
Combine input records=0
Combine output records=0
Reduce input groups=106
Reduce shuffle bytes=175650573
.
.
```

...

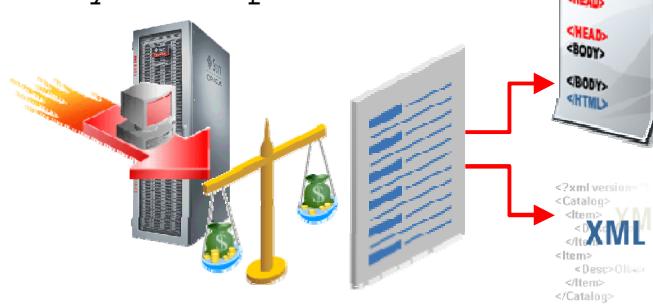


Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The screen capture displays running the `pb_balance.sh` script and a partial output.

Perfect Balance–Generated Reports

- Perfect Balance generates the Job Analyzer reports when it runs a job. The reports contain indicators about the distribution of the load in a job
 - Saved in HTML format for your use
 - Saved in XML format for Perfect Balance use
- The reports' names are:
 - jobanalyzer-report.html
 - jobanalyzer-report.xml



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The following are the reports that are generated by Perfect Balance for its own use:

Reduce key metric reports:

- Perfect Balance generates a report for each file partition, when the appropriate configuration properties are set.
- They are saved in XML for Perfect Balance use.
- They are named
 `${job_output_dir}/_balancer/ReduceKeyMetricList-attempt_jobid_taskid_task_attemptid.xml`
- Generated only when the counting reducer is used:
`oracle.hadoop.balancer.autoAnalyze=REDUCER_REPORT` when using Perfect Balance
- Or a call to the `Balancer.configureCountingReducer` method when using the API

Partition report:

- Saved in XML for Perfect Balance use only
- The generated report is only generated for balanced jobs:
 `${job_output_dir}/_balancer/orabalancer_report.xml`

Reduce key metric reports:

- Generates a report for each reduce task
- Saved in XML only for Perfect Balance use

The Job Analyzer Reports: Structure of the Job Output Directory

The reports are stored by default in the job output directory.

- `${mapreduce.output.fileoutputformat.outputdir}` in YARN
 - `${mapred.output.dir}` in MRv1

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Reading the Job Analyzer Reports

You can open the report in a browser:

- Directly in HDFS, or
- After copying it to the local file system



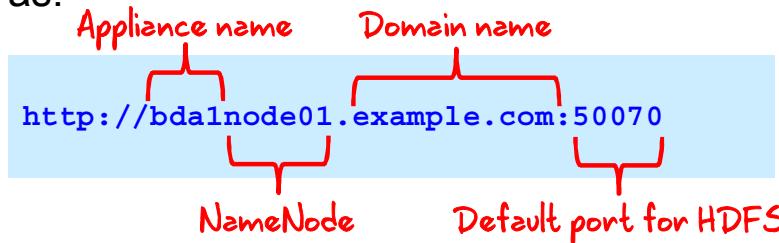
ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Reading the Job Analyzer Report in HDFS Using a Web Browser

To open a Job Analyzer report in HDFS in a browser:

1. Open the HDFS web interface on port 50070 of a NameNode node (node01 or node02) using a URL such as:



2. From the Utilities menu, choose Browse the File System.
3. Navigate to the `<job_output_dir>/_balancer` directory.



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Reading the Job Analyzer Report in the Local File System in a Web Browser

To open a Job Analyzer report in the local file system in a browser:

1. Copy the report from HDFS to the local file system:

```
$ hadoop fs -get job_output_dir/_balancer/jbanalyzer-report.html /home/jdoe
```

2. Switch to the local directory:

```
$ cd /home/jdoe
```

3. Open the file in a browser:

```
$ firefox jbanalyzer-report.html
```

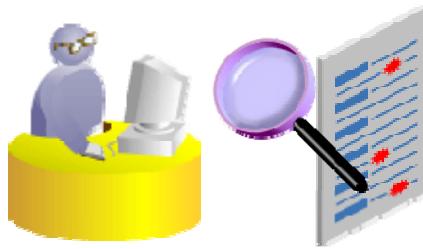


Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Looking for Skew Indicators in the Job Analyzer Reports

When inspecting the Job Analyzer report, look for indicators of skew such as:

- The execution time of some reducers is longer than others.
- Some reducers process more records or bytes than others.
- Some map output keys have more records than others.
- Some map output records have more bytes than others.



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Job Analyzer Sample Reports

| Job Information | | Time Information | |
|---------------------------------|--|-----------------------|--|
| Job Name invindx | | Map Phase 00:00:29 | |
| Job Id job_1405207264024_0122 | | Reduce Phase 00:00:19 | |
| Start Time 2014-07-22 15:03:39 | | Shuffle 00:00:04 | |
| Finish Time 2014-07-22 15:04:35 | | Merge 00:00:08 | |
| | | Reduce 00:00:15 | |
| | | Job 00:00:56 | |

| Reduce Tasks Metrics Summary | | | | | | | | |
|------------------------------|----------|----------|----------|----------|--|--|--|--|
| Task ID | Time | | | %Load | | | | |
| | Start | Finish | Elapsed | Observed | | | | |
| 0 | 15:04:15 | 15:04:27 | 00:00:11 | 12 | | | | |
| 1 | 15:04:15 | 15:04:25 | 00:00:09 | 7 | | | | |
| 2 | 15:04:15 | 15:04:25 | 00:00:09 | 8 | | | | |
| 3 | 15:04:15 | 15:04:25 | 00:00:09 | 7 | | | | |
| 4 | 15:04:15 | 15:04:27 | 00:00:11 | 10 | | | | |
| 5 | 15:04:15 | 15:04:28 | 00:00:12 | 13 | | | | |
| 6 | 15:04:16 | 15:04:24 | 00:00:07 | 4 | | | | |
| 7 | 15:04:16 | 15:04:23 | 00:00:06 | 3 | | | | |
| 8 | 15:04:16 | 15:04:35 | 00:00:18 | 29 | | | | |
| 9 | | | | | | | | |

| Reduce Tasks Metrics | | | | | | | | | | |
|----------------------|--------------|-------|--------|---------------|-----------|---------|------------|---------|-----------|----|
| Task ID | Elapsed Time | | | Input | | | | Output | | |
| | Shuffle | Merge | Reduce | Shuffle Bytes | Keys | Records | ValueBytes | Records | Bytes | |
| | count | % | count | % | count | % | count | % | count | |
| 0 | 20,987,702 | 12 | 13 | 13 | 2,300,118 | 12 | 43,702,242 | 12 | 1,698,171 | 12 |
| 1 | 15,630,571 | 9 | 12 | 12 | 1,438,876 | 7 | 27,338,644 | 7 | 1,309,838 | 9 |
| 2 | 14,682,583 | 8 | 7 | 7 | 1,503,328 | 8 | 28,563,232 | 8 | 1,220,301 | 9 |
| 3 | 14,977,081 | 9 | 10 | 10 | 1,450,603 | 7 | 27,561,457 | 7 | 1,253,630 | 9 |
| 4 | 21,175,220 | 12 | 11 | 11 | 2,073,885 | 10 | 39,403,815 | 10 | 1,768,698 | 13 |

The input records range from **3%** to **29%**, and their corresponding elapsed times range from **6** to **18** seconds. This variation indicates skew.

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The slide examples show the beginning of the analyzer report for the inverted index (invindx) example. It displays the key load coefficient recommendations, because this job ran with the appropriate configuration settings.

The task IDs are links to tables that show the analysis of specific tasks, enabling you to drill down for more details from the first, summary table.

This example uses an extremely small data set, but notice the differences between tasks 7 and 8: The input records range from **3%** to **29%**, and their corresponding elapsed times range from **6** to **18** seconds. **This variation indicates skew.**

Collecting Additional Metrics with Job Analyzer

- If you set the `oracle.hadoop.balancer.autoAnalyze` property to `REDUCER_REPORT` the Job Analyzer report includes more details—the load metrics for each key.
- The Job Analyzer report can also compare its predicted load with the actual load when `REDUCER_REPORT` is set.
 - The difference between these values measures how effective Perfect Balance was in balancing the job.



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The Job Analyzer report includes the load metrics for each key, if you set the `oracle.hadoop.balancer.autoAnalyze` property to `REDUCER_REPORT`. This additional information provides a more detailed picture of the load for each reducer, with metrics that are not available in the standard Hadoop counters.

The Job Analyzer report also compares its predicted load with the actual load. The difference between these values measures how effective Perfect Balance was in balancing the job.

Using Data from Additional Metrics

- Use the `feedbackDir` property to use Job Analyzer output from a previous run of the job while running a job with Perfect Balance.
 - `oracle.hadoop.balancer.linearKeyLoad.feedbackDir`
- Alternately, if you already know good values of the load model coefficients for your job, you can set the load model properties:
 - `oracle.hadoop.balancer.linearKeyLoad.byteWeight`
 - `oracle.hadoop.balancer.linearKeyLoad.keyWeight`
 - `oracle.hadoop.balancer.linearKeyLoad.rowWeight`



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Job Analyzer might recommend key load coefficients for the Perfect Balance key load model, based on its analysis of the job load. To use these recommended coefficients when running a job with Perfect Balance, set the

`oracle.hadoop.balancer.linearKeyLoad.feedbackDir` property to the directory containing the Job Analyzer report of a previously analyzed run of the job.

If the report contains recommended coefficients, then Perfect Balance automatically uses them. If Job Analyzer encounters an error while collecting the additional metrics, then the report does not contain the additional metrics.

Running the job with these coefficients results in a more balanced job.

Using Perfect Balance API

- The `oracle.hadoop.balancer.Balancer` class contains methods for:
 - Creating a partitioning plan
 - Saving the plan to a file
 - Running the MapReduce job using the plan
- You only need to add the code to the application's job driver Java class, not redesign the application.



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Chopping

- To balance a load, Perfect Balance might subpartition the values of a single reduce key and send each subpartition to a different reducer. This is chopping.
- You can configure how Perfect Balance chops the values by setting the `oracle.hadoop.balancer.enableSorting` configuration property:
 - Chopping by hash partitioning: Set `enableSorting=false` when sorting is not required (default)
 - Chopping by sorting: Set `enableSorting=true` to sort the values in each subpartition and order them across all subpartitions



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

You can configure how Perfect Balance chops the values by setting the [oracle.hadoop.balancer.enableSorting](#) configuration property:

- Chopping by hash partitioning: Set `enableSorting=false` when sorting is not required. This is the default chopping strategy.
- Chopping by sorting: Set `enableSorting=true` to sort the values in each subpartition and order them across all subpartitions. In any parallel sort job, each task sorts the rows within the task. The job must ensure that the values in reduce task 2 are greater than values in reduce task 1, the values in reduce task 3 are greater than the values in reduce task 2, and so on. The job generates multiple files containing data in sorted order, instead of one large file with sorted data.

For example, if a key is chopped into three subpartitions, and the subpartitions are sent to reducers 5, 8, and 9, then the values for that key in reducer 9 are greater than all values for that key in reducer 8, and the values for that key in reducer 8 are greater than all values for that key in reducer 5. When `enableSorting=true`, Perfect Balance ensures this ordering across reduce tasks.

Disabling Chopping

- To disable chopping, set:
`oracle.hadoop.balancer.keyLoad.minChopBytes=-1`
- Perfect Balance still offers performance gains by combining smaller reduce keys, called bin packing



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

If an application requires that the data is aggregated across files, then you can disable chopping by setting `oracle.hadoop.balancer.keyLoad.minChopBytes=-1`. Perfect Balance still offers performance gains by combining smaller reduce keys, called bin packing.

Troubleshooting Jobs Running with Perfect Balance

- If you get Java "out of heap space" or "GC overhead limit exceeded" errors on the client node while running the Perfect Balance sampler, then increase the client JVM heap size for the job.
- Use the `Java -Xmx` option.
- You can specify client JVM options before running the Hadoop job, by setting the `HADOOP_CLIENT_OPTS` variable:

```
$ export HADOOP_CLIENT_OPTS="-Xmx1024M $HADOOP_CLIENT_OPTS"
```



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

If you get Java "out of heap space" or "GC overhead limit exceeded" errors on the client node while running the Perfect Balance sampler, then increase the client JVM heap size for the job.

Use the `Java -Xmx` option. You can specify client JVM options before running the Hadoop job, by setting the `HADOOP_CLIENT_OPTS` variable as shown in the slide.

Setting `HADOOP_CLIENT_OPTS` changes the JVM options only on the client node. It does not change JVM options in the map and reduce tasks. See the `invindx` script for an example of setting this variable.

Setting `HADOOP_CLIENT_OPTS` is sufficient to increase the heap size for the sampler, regardless of whether `oracle.hadoop.balancer.runMode` is set to `local` or `distributed`. When `runMode=local`, the sampler runs on the client node, and `HADOOP_CLIENT_OPTS` sets the heap size on the client node.

When `runMode=distributed`, Perfect Balance automatically sets the heap size for the sampler Hadoop job based on the `-Xmx` setting you provide in `HADOOP_CLIENT_OPTS`.

Perfect Balance never changes the heap size for the map and reduce tasks of your job, only for its sampler job.

Perfect Balance Examples Available with Installation

- The Perfect Balance installation files include a full set of examples that you can run immediately.
- The InvertedIndex example:
 - Is a MapReduce application that creates an inverted index on an input set of text files
 - Maps words to the location of the words in the text files
 - The input data is included
- You can use the examples as follows:
 - Run the examples in the documentation, or
 - Use the examples as the basis for running your own jobs, after making the changes listed in the notes section



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The InvertedIndex example is also used in the Oracle Big Data Appliance Software User's Guide Release 4 (4.0) documentation reference. They use the same data set and run the same MapReduce application.

You can run the examples in the documentation, or use them as the basis for running your own jobs, after making the following changes:

- If you are modifying the examples to run your own application, add your application JAR files to `HADOOP_CLASSPATH` and `-libjars`.
- Ensure that the value of `mapreduce.input.fileinputformat.inputdir` identifies the location of your data. The `invindx/input` directory contains the sample data for the InvertedIndex example. To use this data, you must first set it up. See the "Extracting the Example Data Set" section in the Oracle Big Data Appliance Software User's Guide documentation.
- Replace `jdoe` with your Hadoop username.

- Set the `-conf` option to an existing configuration file. The `jdoe_conf_invindx.xml` file is a modification of the configuration file for the `InvertedIndex` examples. You can use the example configuration file as is or modify it. See
`/opt/oracle/orabalancer-2.0.0-`
`h2/examples/invindx/conf_mapreduce.xml` (or `conf_mapred.xml`).
- Review the configuration settings in the file and in the shell script to ensure they are appropriate for your job.
- You can run the browser from your laptop or connect to Oracle BDA by using a client that supports graphical interfaces, such as VNC.

Summary

In this lesson, you should have learned how to:

- Define the Perfect Balance feature of Oracle BDA
- Use Perfect Balance to balance MapReduce jobs
- Run Job Analyzer as a stand-alone utility or with Perfect Balance
- Identify, locate, and read the generated reports
- Collect additional metrics with Job Analyzer
- Configure Perfect Balance
- Use chopping (partitioning of values)
- Troubleshoot jobs running with Perfect Balance
- Use the Perfect Balance installation examples



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Practice 27: Overview

In this practice, you will use the Oracle BDA Perfect Balance feature and generate and view some reports.



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

28

Securing Your Data

ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Course Road Map

Module 1: Big Data Management System

Module 2: Data Acquisition and Storage

Module 3: Data Access and Processing

Module 4: Data Unification and Analysis

Module 5: Using and Managing Oracle Big Data Appliance

Lesson 25: Introduction to the Oracle Big Data Appliance (BDA)

Lesson 26: Managing Oracle BDA

Lesson 27: Balancing MapReduce Jobs

Lesson 28: Securing Your Data

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

This lesson describes how to secure Oracle BDA using Kerberos.

Objectives

After completing this lesson, you should be able to describe how to secure data on the Big Data Appliance.



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Security Trends

- Security for Hadoop is becoming more accessible and easier to implement.
- Hadoop is becoming more mainstream in the enterprise.
- Because security features are emerging, Hadoop will become a more likely target for sensitive data.



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

- Security is a calculated decision. Because security does not make systems run faster or make them easier to use, the cost of implementing any security control must balance the potential cost of *not* implementing any security procedures or practices.
- How much security is required is often a judgment made on the amount of risk an organization is willing to assume. That risk often boils down to the value of the data. If the data being housed in your BDA is a unique combination of sensor data, geological data, and periodical feeds from gold mining journals, it has very different security requirements than a cluster housing publicly available home values, traffic accident data, and citizen demographic data.

Security Levels

Level 0

- **(Relaxed security): Access controls trust users' credentials**

Level 1

- **(Bastion): Limits access and exposure to the Hadoop server**

Level 2

- **(Access, Authentication, and Trust): Adds user centralized user provisioning, security, and management**

Level 3

- **(Compliance Grade Security): Adds Role Based Access Control (RBAC) and Encryption**

ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Outline

You will explore the key capabilities that support this security spectrum, including:

- Authentication:
 - The subject (user, program, process, service) proves their identity to gain access to the system.
- Authorization:
 - Authenticated subjects are granted access to authorized resources.
- Auditing:
 - Accesses and manipulations are recorded in logs for auditing/accountability/compliance.
- Encryption (at rest and over the network):
 - It protects against man-in-the-middle attack during transit, and data breach on data at rest.



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Relaxed Security

Trust-based model:

- Any client that has the cluster connection details can access the Hadoop cluster.
- User/Group information stored in Linux accounts on the BDA's critical nodes when access controls are required.
- User/Group information also captured in Hue and Cloudera Manager (CM).



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

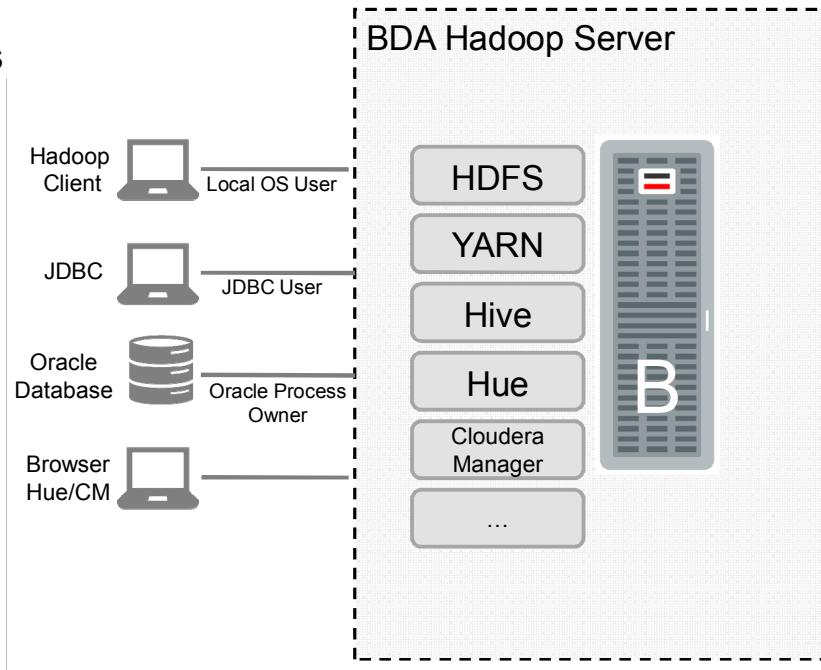
Authentication with Relaxed Security

Hadoop and JDBC Clients

- No authentication.
User identity trusted
- HiveServer2
impersonates user

Hue & CM

- Users authenticated
against application
- Hue impersonates
authenticated user



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

- **Hadoop client:** Local OS user ID is used.
- **JDBC:** Specify any user as part of the connect string.
- **Oracle Database:** The user is the owner of the Oracle process.

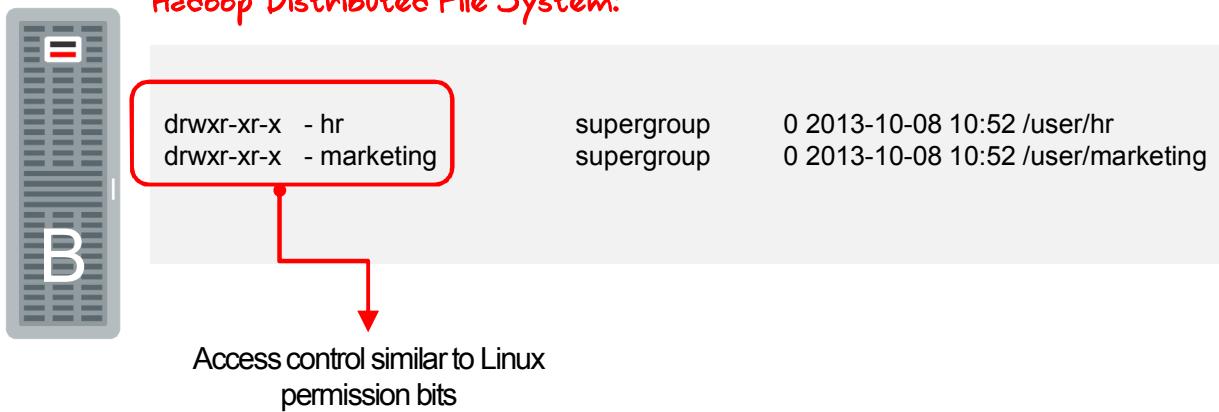
Authorization

- Group Membership
 - Group membership from Linux OS
- HDFS
 - ACL on folder/file controls file access
- Hive
 - HiveServer2 impersonates connected user
 - Folder/file ACLs used to scope data access
- Cloudera Manager
 - Authorizes access to functionality based on role



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

HDFS ACLs



ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Changing Access Privileges

Use the Hadoop CLI plus familiar Linux file system commands to change permissions on files and folders:

Review permissions for the sgtpepper.txt file:

```
$ hadoop fs -ls sgtpepper.txt
-rw-r--r-- 1 oracle oracle 34446 2014-10-30 11:29 sgtpepper.txt
```

Allow all users to read/delete the file:

```
$ hadoop fs -chmod 666 sgtpepper.txt
$ hadoop fs -ls sgtpepper.txt
-rw-rw-rw- 1 oracle oracle 34446 2014-10-30 11:29 sgtpepper.txt
```

Make lennon and thebeatles the owners of the file (must be done as superuser):

```
$ sudo -u hdfs hadoop fs -chown lennon:thebeatles/user/oracle/sgtpepper.txt
$ hadoop fs -ls sgtpepper.txt
-rw-rw-rw- 1 lennon thebeatles 34446 2014-10-30 11:29 sgtpepper.txt
```



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Relaxed Security Summary

- Trust-based model
- Not intended to prevent users from malicious behavior
- HDFS ACLs provide a level of authorization—but can be circumvented.
 - Prevents well-intentioned users from potentially incorrect action



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Challenges with Relaxed Security

Lack of strong authentication causes many problems:

- User Impersonation
 - Masquerade as someone else
- Group Impersonation
 - Masquerade as a member of a group with greater access
- Service Impersonation
 - Masquerade as a service such as becoming a fake NameNode or a fake DataNode



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

BDA Secure Installation

Mammoth automates the setup of a secure cluster as follows:

- Installs and configures Kerberos for strong authentication
- Installs and configures Sentry to manage authorization
- Configures auditing with Oracle Audit Vault
- Configures encryption



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The slide bullets span several of the levels mentioned in the “Security Levels” slide.

Kerberos Key Definitions

| Name | Description |
|-------------------------------|---|
| Kerberos | A computer network authentication protocol. It uses time-stamped tickets to allow nodes communicating over a nonsecure network to prove their identity to one another in a secure manner. |
| Principal (identities) | Identities in Kerberos are called principals. There are user and service principals. Users and services using Kerberos authentication protocol require principals to uniquely identify themselves. |
| Realm | A realm is an authentication administrative domain. User and service principals are assigned to a specific Kerberos realm. |
| Key Distribution Center (KDC) | The KDC contains: <ul style="list-style-type: none"> • The Authentication Service (AS): Issues a ticket-granting tickets (TGT) to a client that initiated a request to the AS. The client can use the TGT to access services. • The Ticket Granting Service (TGS): Validates TGTs and grants service tickets that enables an authenticated principal to use the requested service |



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

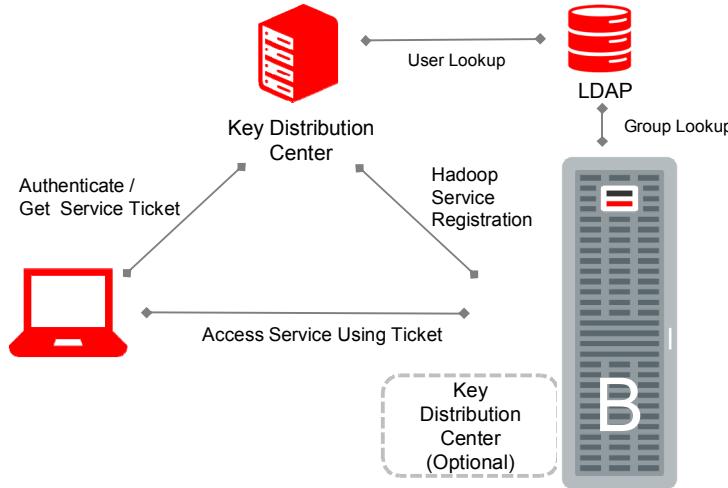
The Key Distribution Center (KDC) holds all users' and services' cryptographic keys. The KDC provides security services to entities referred to as *principals*. The KDC and each principal share a *secret key*. The KDC uses the secret key to encrypt data, sends it to the principal, and the principal uses the secret key to decrypt and process the data. There are two categories of principles:

- User principals
- Service principals

The KDC has two components:

- The Authentication Server (AS)
- The Ticket Granting Server (TGS)

Strong Authentication with Kerberos



ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Key Distribution Center (KDC): Holds all users' and services' cryptographic keys. KDC provides security services to entities known as *principals*. There are two types of principals:

- User principals: Users that are accessing machines and services
- Service principals: Services that are running on the system, such as HDFS, YARN, and so on.

Kerberos provides a secure way of ensuring the identity of users and services communicating over the network. It ensures that users are who they claim to be, and that the services, are not imposters. Instead of sending passwords over the network, encrypted, time-stamped tickets are used to gain access to services. The KDC is responsible for providing these tickets.

Snapshot of Principals in KDC

```
$ kadmin.local
kadmin.local: listprincs
```

```
kadmin.local: listprincs
HTTP/scaj51bda10.us.oracle.com@DEV.ORACLE.COM
HTTP/scaj51bda11.us.oracle.com@DEV.ORACLE.COM
HTTP/scaj51bda12.us.oracle.com@DEV.ORACLE.COM
HTTP/scaj51bda13.us.oracle.com@DEV.ORACLE.COM
K/M@DEV.ORACLE.COM
cloudera-scm/admin@DEV.ORACLE.COM
hdfs/scaj51bda10.us.oracle.com@DEV.ORACLE.COM
hdfs/scaj51bda11.us.oracle.com@DEV.ORACLE.COM
hdfs/scaj51bda12.us.oracle.com@DEV.ORACLE.COM
hdfs/scaj51bda13.us.oracle.com@DEV.ORACLE.COM
hive/scaj51bda13.us.oracle.com@DEV.ORACLE.COM
host/scaj51bda10.us.oracle.com@DEV.ORACLE.COM
host/scaj51bda11.us.oracle.com@DEV.ORACLE.COM
httpfs/scaj51bda11.us.oracle.com@DEV.ORACLE.COM
hue/scaj51bda12.us.oracle.com@DEV.ORACLE.COM
hue/scaj51bda13.us.oracle.com@DEV.ORACLE.COM
kadmin/admin@DEV.ORACLE.COM
kadmin/changepw@DEV.ORACLE.COM
kadmin/scaj51bda10.us.oracle.com@DEV.ORACLE.COM
krbtgt/DEV.ORACLE.COM@DEV.ORACLE.COM
mapred/scaj51bda12.us.oracle.com@DEV.ORACLE.COM
oozie/scaj51bda13.us.oracle.com@DEV.ORACLE.COM
oracle@DEV.ORACLE.COM
yarn/scaj51bda10.us.oracle.com@DEV.ORACLE.COM
yarn/scaj51bda11.us.oracle.com@DEV.ORACLE.COM
yarn/scaj51bda12.us.oracle.com@DEV.ORACLE.COM
yarn/scaj51bda13.us.oracle.com@DEV.ORACLE.COM
zookeeper/scaj51bda10.us.oracle.com@DEV.ORACLE.COM
zookeeper/scaj51bda11.us.oracle.com@DEV.ORACLE.COM
zookeeper/scaj51bda12.us.oracle.com@DEV.ORACLE.COM
```

- Principals automatically configured by Mammoth
- kadmin tool used to manage principals
- A principal comprises three components:
 - Primary or User
 - Instance (optional)
 - Realm

hue/scaj51bda12.us.oracle.com@DEV.ORACLE.COM
lucy@DEV.ORACLE.COM



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Setting up all of the service principals required for the Hadoop cluster has been automated by Mammoth. In the slide, you can see service principals for HDFS, Hue, YARN, and more. Note that it is not enough to simply specify the service name; the instance of the service (which includes the host where the service runs) is also part of the principal name. You can view the list of principals by using the Kerberos Admin tool. Here, we have logged into the KDC by using `kadmin.local` and then listed the principals by using the `listprincs` command.

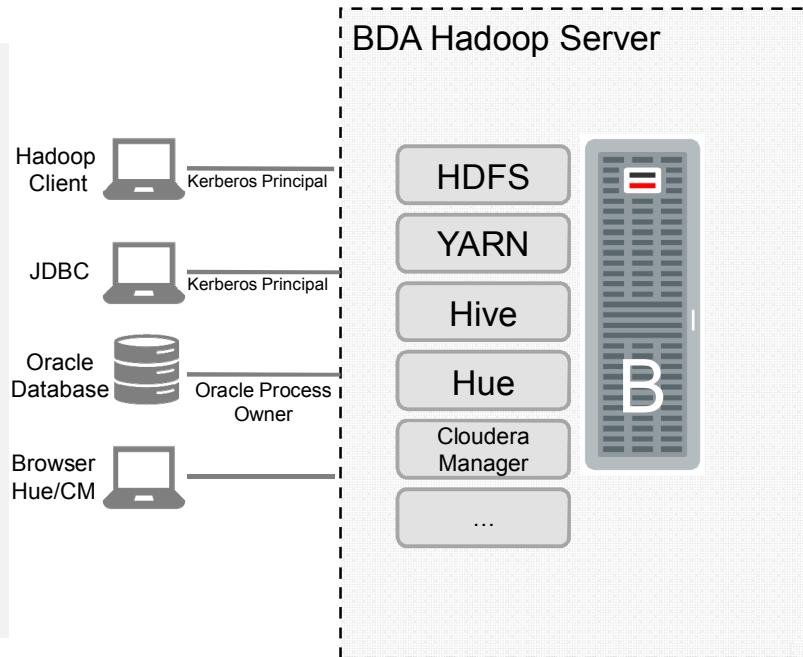
Authentication with Kerberos

Hadoop and JDBC clients

- Connected as Kerberos principal
- HiveServer2 executes operations as **hive** user*.

Hue & CM

- Users authenticated against application
- Hue impersonates authenticated user



*Requirement for Sentry—not a by-product of Kerberos authentication.

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

HiveServer2 is no longer impersonating.

User Authentication: Examples

Example 1: Attempt to access HDFS without authenticating:

```
[oracle@stcurrbdal ~]$ hadoop fs -ls  
  
15/03/05 11:39:59 WARN security.UserGroupInformation:  
PrivilegedActionException as:oracle (auth:KERBEROS)  
cause:javax.security.sasl.SaslException: GSS initiate failed [Caused by  
GSSEException: No valid credentials provided (Mechanism level:  
Failed to find any Kerberos tgt)]
```



Example 2: Authenticate and then successfully access HDFS:

```
[oracle@stcurrbdal ~]$ kinit oracle  
Password for oracle@DEV.ORACLE.COM:  
  
[oracle@stcurrbdal ~]$ hadoop fs -ls  
Found 11 items  
drwx-----  - oracle hadoop          0 2015-01-14 16:00 .Trash  
drwx-----  - oracle hadoop          0 2015-01-30 09:18 .staging  
drwxr-xr-x  - oracle hadoop          0 2015-01-08 07:45 moviework  
drwxr-xr-x  - oracle hadoop          0 2015-01-08 07:52 oggdemo  
drwxr-xr-x  - oracle hadoop          0 2015-01-15 15:41 oozie-oozi  
-rw-r--r--   3 oracle hadoop        250 2015-01-09 11:02 sgtpepper.txt
```



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Service Authentication and Keytabs

Keytabs created for each service:

- Contain principal and encrypted key
- Encrypted key derived from password

Enables a service to start without requiring a password

Limited access to keytab

Located in the
`/var/run/cloudera-scm-agent/process/<process>`
 directory.

Configurations for processes:

```
drwxr-x--x 4 hive   hive   240 Jan 18 00:57 646-hive-HIVEMETASTORE
drwxr-x--x 4 hive   hive   240 Jan 18 00:57 685-hive-HIVESERVER2
drwxr-x--x 4 hive   hive   240 Jan 18 00:57 686-hive-WEBHCAT
drwxr-x--x 4 hive   hive   240 Jan 18 01:06 684-hive-HIVEMETASTORE
drwxr-x--x 4 hive   hive   220 Jan 18 01:06 687-hive-HIVEMETASTORE
drwxr-xr-x 3 yarn   hadoop 420 Jan 18 01:08 674-yarn-NODEMANAGER
drwxr-xr-x 3 yarn   hadoop 400 Jan 18 01:08 691-yarn-NODEMANAGER
drwxr-x--x 3 yarn   hadoop 480 Jan 20 05:07 678-yarn-RESOURCEMANAGER
[root@scaj51bda13 process]# pwd
/var/run/cloudera-scm-agent/process
[root@scaj51bda13 process]# cd ..
```

Keytab and TGT cache. Cache is updated.

```
[root@: 678-yarn-RESOURCEMANAGER]# ls
capacity-scheduler.xml          log4j.properties
cloudera-monitor.properties    logs
cloudera-stack-monitor.properties mapred-site.xml
core-site.xml                  nodes_allow.txt
event-filter-rules.json         nodes_exclude.txt
fair-scheduler.xml              ssl-client.xml
hadoop-metrics2.properties     ssl-server.xml
hadoop-policy.xml              topology.map
hdfs-site.xml                  topology.py
http-auth-signature-secret    yarn.keytab
krb5cc_487                      yarn-site.xml
[root@: 678-yarn-RESOURCEMANAGER]#
```

ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Services will not prompt for passwords. You can create a Kerberos keytab file by using the `ktutil` command. You can use a keytab file, which stores passwords, and supply it to the `kinit` command with the `-t` option.

Review TGT Cache

- The TGT Cache has a starting and an expiration date.
- For services, they must be automatically refreshed in order for the service to operate.

```
[root@stcurrbdal 678-yarn-RESOURCEMANAGER]# klist krb5cc_487
Ticket cache: FILE:krb5cc_487
Default principal: yarn/stcurrbdal.us.oracle.com@DEV.ORACLE.COM

Valid starting     Expires            Service principal
01/20/15 05:07:00  01/20/15 09:55:00  krbtgt/DEV.ORACLE.COM@DEV.ORACLE.COM
                           renew until 01/20/15 09:55:00
```



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Ticket Renewal

Cloudera Manager enables administrators to specify TGT refresh policies.

The screenshot shows the Cloudera Manager Administration Settings page. A search bar at the top contains the text "lifetime". A note below the search bar states: "Settings marked will not take effect until after the Cloudera Manager Server has restarted." A "Save Changes" button is visible. The main table lists Kerberos properties:

| Category | Property | Value | Description |
|----------|---------------------------------------|----------|--|
| Kerberos | Kerberos Ticket Lifetime | 1 day(s) | Default lifetime for initial ticket requests. |
| Kerberos | Kerberos Renewable Lifetime | 7 day(s) | Default renewable lifetime for initial ticket requests. |
| Kerberos | Maximum Renewable Life for Principals | 5 day(s) | Maximum renewable lifetime for Kerberos principals generated by Cloudera Manager. This property is used only if MIT KDC is used. Set this property to zero if the KDC should provide the maximum renewable lifetime. Note: Principals with non-renewable tickets are not recommended because it can prevent Hadoop services from functioning. |

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Adding a New User

To add a new user with group membership:

- Add the user's principal to the KDC
- Add the user to each critical BDA node
 - User does not need login privileges
 - Assign user to group(s)

Note: Hue maintains its own users and follows a different process.



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Example: Adding a New User

Perform the following steps to provision two new users:

1. bob in the marketing group
2. lucy in the hr group

Add users with groups on each BDA critical node:

```
useradd -r -g marketing bob  
useradd -r -g hr lucy
```

Add users to KDC:

```
kadmin.local  
kadmin.local: addprinc bob@DEV.ORACLE.COM  
kadmin.local: addprinc lucy@DEV.ORACLE.COM
```



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Example: Adding a User to Hue

The screenshot shows the Hue User Admin interface with the 'Users' tab selected. A sub-menu 'Hue Users - Create user' is open. The process is divided into three steps: Step 1: Credentials (required), Step 2: Names and Groups, and Step 3: Advanced. Step 1 is active, showing fields for Username (bob), Password, and Password confirmation, along with a checkbox for 'Create home directory'. Navigation buttons at the bottom are 'Back', 'Next' (highlighted in blue), and 'Add user'.

Hue provides a wizard to provision a new user.

Updates Hue users/groups:

- Only impacts access through Hue
- Hue impersonates the user when accessing Hadoop services.
- It does not require updates to KDC.

ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Authorization

With strong authentication, authorization rules can be meaningful.

- HDFS ACLs prevent unauthorized file access.
- Access to Hive metadata is controlled by Sentry.



ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Without Sentry, you can query everything in Hive. This applies to Hive's Metastore.

Sentry Authorization Features

- Secure Authorization
 - Ability to control access to data and/or privileges on data for authenticated users
- Fine-grained Authorization
 - Ability to give users access to a subset of data
 - Includes access to a database, URI, table, or view
- Role-based Authorization
 - Ability to create or apply template-based privileges based on functional roles



ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Sentry Configuration

As part of the Sentry configuration, HiveServer2 impersonation must be disabled.

- All data access is executed by the Hive user.
- Changes will need to be made to the HDFS privilege model to effectively authorize access to data.
- Without changes to this model, all users accessing the Hive data would need to be part of the Hive group—rendering authorization meaningless.
- This will be covered later in this lesson.



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Users, Groups, and Roles

- User is an authenticated entity.
 - For example, LDAP user or Kerberos principal
- Group is a collection of users.
 - Users are assigned to groups.
- Sentry Role is a collection of privileges.
 - Roles are assigned to groups.



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Sentry Example: Overview

The example on the several few slides illustrates the use of Sentry to authorize access to four user segments:

| User Segment | Description |
|----------------------|--------------------------------|
| Sentry Administrator | Administers access to the data |
| HR Team | Access to the HR data only |
| Marketing Team | Access to marketing data only |
| Development Team | Access to all data |



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Example: Users, Roles, and Groups

| User | Group | Sentry Role | Capability |
|--------|-------------|-----------------|---|
| oracle | hive | | Superuser |
| sally | development | developer | Read/write data from any group |
| bob | marketing | marketing_admin | Only read/write marketing data (movie logs) |
| lucy | hr | hr_admin | Only read/write HR information (salaries) |



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

1. Creating Roles

Connect to Hive as an administrator using beeline or Hue, and then create the following roles:

1. marketing_admin
2. hr_admin
3. developer

```
CREATE ROLE marketing_admin;
CREATE ROLE hr_admin;
CREATE ROLE developer;
```



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

2. Assigning Roles to Groups

After creating the roles, assign them to the appropriate groups:

- The development group is assigned all of the roles.
- The hr group is assigned the hr_admin role.
- The marketing group is assigned the marketing_admin role.

```
GRANT ROLE developer,hr_admin,marketing_admin TO GROUP
development;

GRANT ROLE hr_admin TO GROUP hr;

GRANT ROLE marketing_admin TO GROUP marketing;
```



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Show Roles for a Group

Review roles granted to the development group:

```
SHOW ROLE GRANT GROUP development;
```

| role | grant_option | grant_time | grantor |
|-----------------|--------------|------------|---------|
| hr_admin | false | NULL | -- |
| marketing_admin | false | NULL | -- |
| developer | false | NULL | -- |



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Create Databases (in Hive)

1. The admin creates the `hr_db` and the `marketing_db` databases:

```
CREATE DATABASE marketing_db;
CREATE DATABASE hr_db;
```

2. The admin assigns:

- Full access to the `marketing_db` to the `marketing_admin` role
- Full access to the `hr_db` to the `hr_admin` role

```
GRANT ALL ON DATABASE marketing_db TO ROLE marketing_admin
WITH GRANT OPTION;
```

```
GRANT ALL ON DATABASE hr_db TO ROLE hr_admin WITH GRANT
OPTION;
```



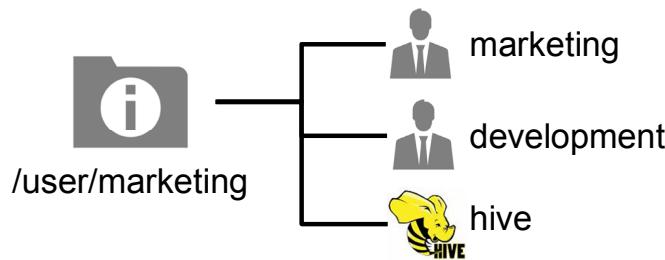
Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

The newly created roles have not yet been granted any privileges. In our example, the admin user creates databases for each group and gives the appropriate roles full access to create and grant access to objects in the respective databases. Because development group has both roles assigned, it has full access to both databases.

Privileges on Source Data for Tables

Now that each group has an assigned database, it is time to create tables in those databases.

- All files in HDFS that will be used by Hive must be accessible to Hive.
- In our example, the marketing team's source data is owned by marketing. However, Hive and the development team also need rw access to the marketing source data.



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

As part of Sentry's configuration, Hive impersonation is turned off. Therefore, the Hive superuser group must be able to access the underlying data.

You also want other users and services to be able to access that data and not just Hive.

In our example, three groups require access to the underlying data files. Simple access privileges are insufficient.

- You must be able to specify privileges for each group (which may differ).
- You must keep ACLs in sync with Sentry authorization.

Privileges on Source Data for Tables

HDFS Extended ACL support enables complex permission rules to be assigned to objects.

- `setfacl`: Sets file access control
- `getfacl`: Retrieves file access control
- + in permission bits indicates extended ACL.

The marketing group assigns privileges to the Hive group:

```
hadoop fs -setfacl -m group:hive:rwx /user/marketing
hadoop fs -getfacl /user/marketing
file: /user/marketing
owner: bob
group: marketing
user::rwx
group::r-x
group:hive:rwx
mask::rwx
other::r-x
hadoop fs -ls /user
drwxrwxr-x+ - bob      marketing ... /user/marketing
```



/user/marketing

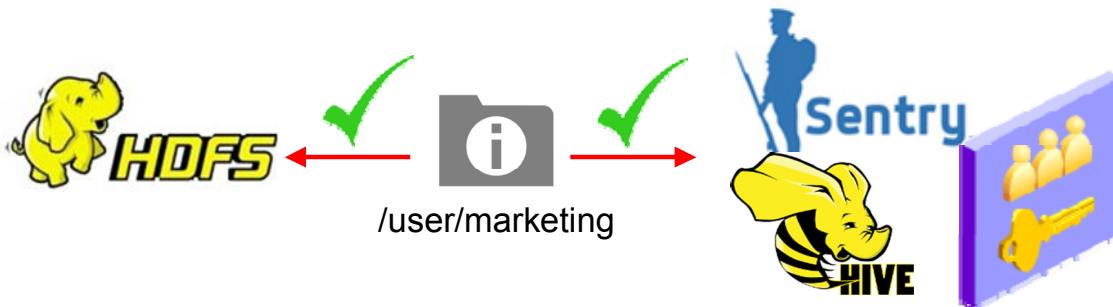
ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Granting Privileges on Source Data for Tables

- The marketing group has access to the data in HDFS.
- The marketing group must also be granted a Sentry privilege to access the files in Hive.
- The admin grants access to the URI as follows:

```
grant all on uri 'hdfs://stcurrh-ns/user/marketing/' to  
role marketing_admin;
```



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Although the marketing group owns the data in HDFS, Sentry must still authorize access to the URI.

Creating the Table and Loading the Data

1

```
CREATE TABLE movieapp_log_avro(
    custid int,
    movieid int,
    activity int,
    genreid int,
    recommended string,
    time string,
    rating int,
    price double,
    position int)
ROW FORMAT DELIMITED
STORED AS INPUTFORMAT
    'org.apache.hadoop.hive.ql.io.avro.AvroContainerInputFormat'
    OUTPUTFORMAT
    'org.apache.hadoop.hive.ql.io.avro.AvroContainerOutputFormat';

LOAD DATA INPATH '/user/marketing/movieapp_3months.avro'
OVERWRITE INTO TABLE movieapp_log_avro;
```

2

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Once the privileges are all set, create the table and load the data.

Attempting to Query the Table Without Privileges

The user in the hr group attempts to query the marketing table without having the required permissions:

```
select * from marketing_db.movieapp_log_avro limit 10;

Error while compiling statement: FAILED:
SemanticException
No valid privileges Required privileges for this query:
Server=server1->Db=marketing_db->Table=movieapp_log_avro-
>action=select;
```



ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Grant and Revoke Access to Table

- Use the GRANT statement on a table to apply SELECT, INSERT, and ALL privileges to a role.
- Use the REVOKE statement to remove privileges.

The Marketing user grants table access to HR:

```
use marketing_db;
grant select on movieapp_log_avro to role hr_admin;
```

The HR user successfully queries the marketing table.

```
use marketing_db;
select count(*) from movieapp_log_avro;

299786
```

The Marketing user subsequently revokes the table access from HR:

```
use marketing_db;
revoke select on movieapp_log_avro from role hr_admin;
```

ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

See Cloudera documentation by using the following url for the complete list of the available privileges:

http://www.cloudera.com/content/cloudera/en/documentation/core/latest/topics/cm_sg_sentry_service.html#concept_cx4_sw2_q4_unique_1

Sentry Key Configuration Tasks

There are three configuration settings that facilitate the use of Sentry:

1. Disable Hive impersonation.
2. Enable extended ACLs in HDFS.
3. Synchronize HDFS ACLs and Sentry Permissions.



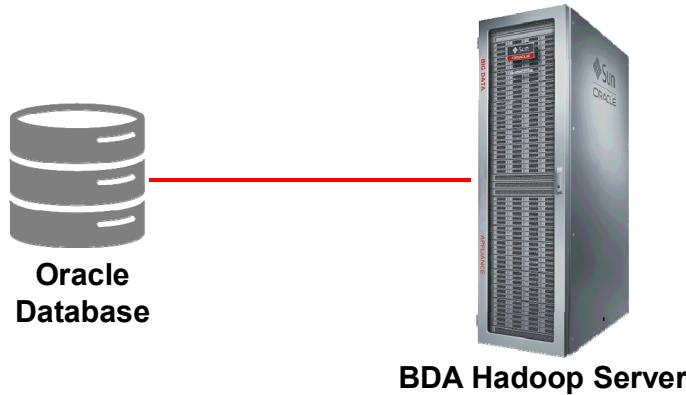
ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Oracle Database Access to HDFS

Oracle Database provides access to data in HDFS by using Oracle Big Data SQL and Oracle Big Data Connectors:

- What HDFS access privileges are required for Oracle Database?
- How can Oracle security capabilities be leveraged?

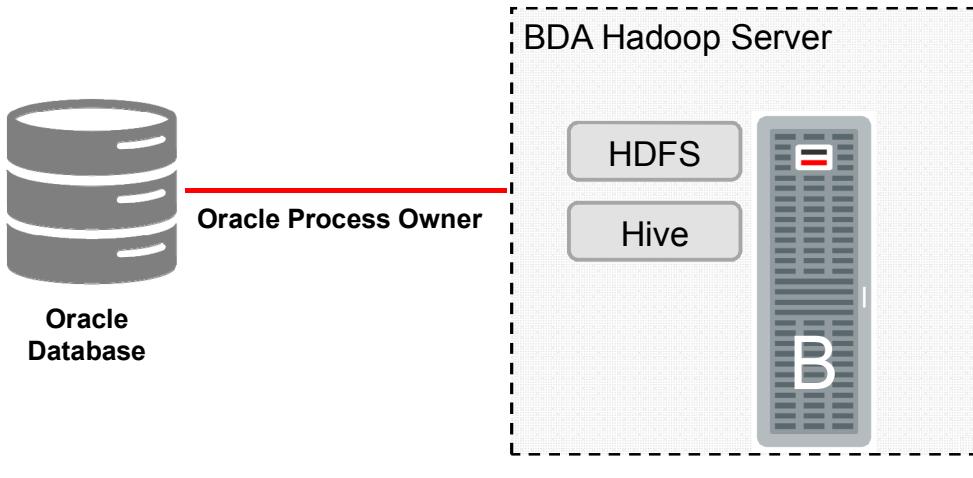


ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Oracle Connection to Hadoop

- Oracle Database connects to Hadoop as the process owner, oracle.
- oracle needs access to all of the required metadata and data.
- It is typically configured as a privileged user such as part of Hive group.



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Hadoop client: The local OS user ID is used.

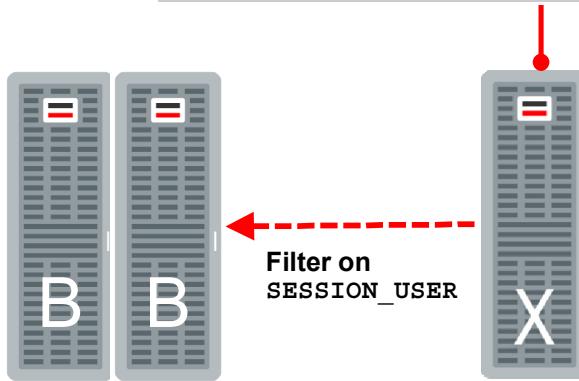
JDBC: Specify any user as part of the connect string.

Oracle Database: The user is the owner of the Oracle process.

Virtual Private Database Policies Restrict Data Access

The same VPD policies that are applied to data in Oracle Database are applied to Big Data sources.

```
SELECT * FROM my_bigdata_table  
WHERE SALES REP ID =  
SYS_CONTEXT('USERENV', 'SESSION_USER');
```



ORACLE

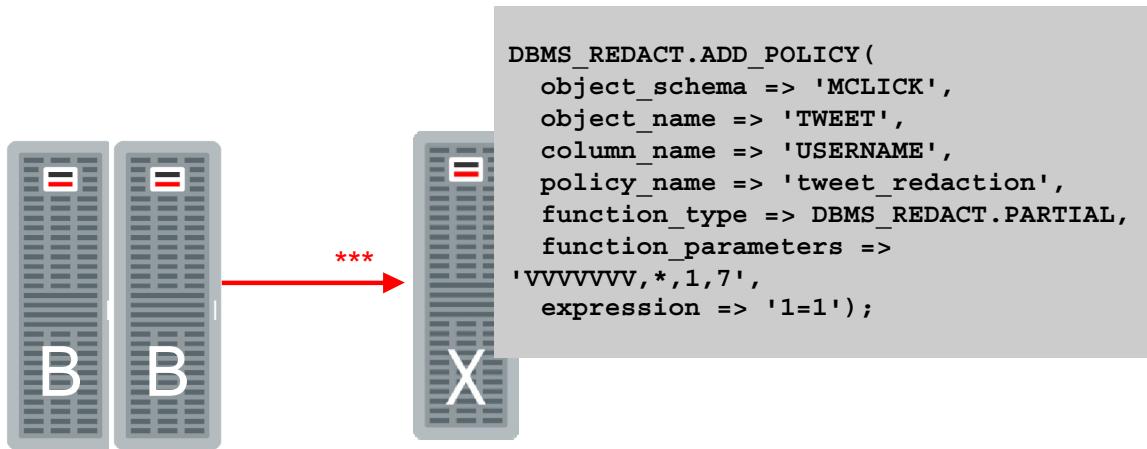
Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Oracle Virtual Private Database (VPD) enables you to create policies to restrict the data accessible to users. Essentially, Oracle Virtual Private Database adds a dynamic WHERE clause to a SQL statement that is issued against the table, view, or synonym to which an Oracle Virtual Private Database security policy was applied.

In the slide example, a policy was created that automatically adds a filter based on session information—specifically the ID of the user. The query result is automatically filtered to show rows where the SALES REP ID column equals this user ID. Note that it does not matter if the data is stored in HDFS or Oracle Database, the same policies are applied.

Oracle Data Redaction Protects Sensitive Data

Oracle Data Redaction policies mask data returned by queries, protecting personally identifiable information.



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Similar to VPD, you can leverage Oracle Data Redaction to data stored in both Oracle Database and Big Data sources. In the slide example, the USERNAME column in the TWEET table is being redacted; the first seven characters in the name are being replaced by “*”. You can control when to apply the redaction policy. Here, redaction always takes place because the expression “ $1=1$ ” is always true.

Note that redaction is applied to data at runtime—when users access the data. This is an important distinction as it allows table joins across different data stores. For example, let us say you had a CUSTOMER table that also contained the USERNAME field. Just like in the TWEET table, imagine that the USERNAME is redacted. To successfully join the USERNAME columns of these two tables, the processing uses the natural values of the columns. The USERNAME will then be encrypted as part of the query result.

Auditing: Overview

- Security requires auditing.
 - Who did what and was it allowed?
- Authentication is a prerequisite.
 - Ensure **who**
- Authorization is a prerequisite.
 - Ensure it was **allowed**



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Auditing

There are two options for auditing data on the BDA:

- Oracle Audit Vault: Integrates audit data from Hadoop with audit data from databases and operating systems
- Cloudera Navigator:
 - Provides deeper level of auditing in Hadoop
 - Does not include auditing data from other sources
 - Includes lineage analysis



cloudera navigator

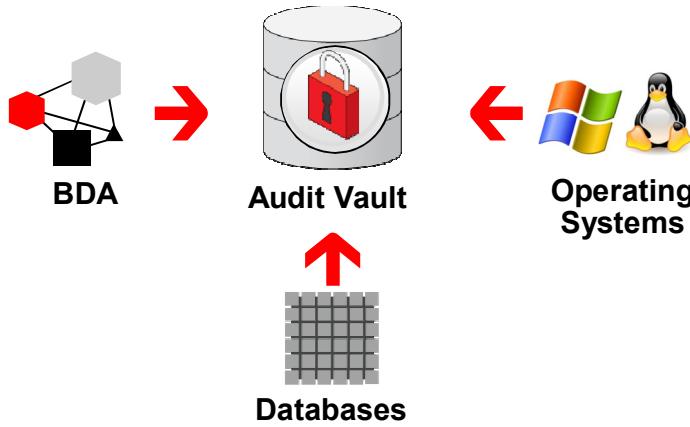
Oracle Audit Vault

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Oracle Audit Vault and Database Firewall

Provides an auditing “data warehouse,” consolidating audit data from multiple sources

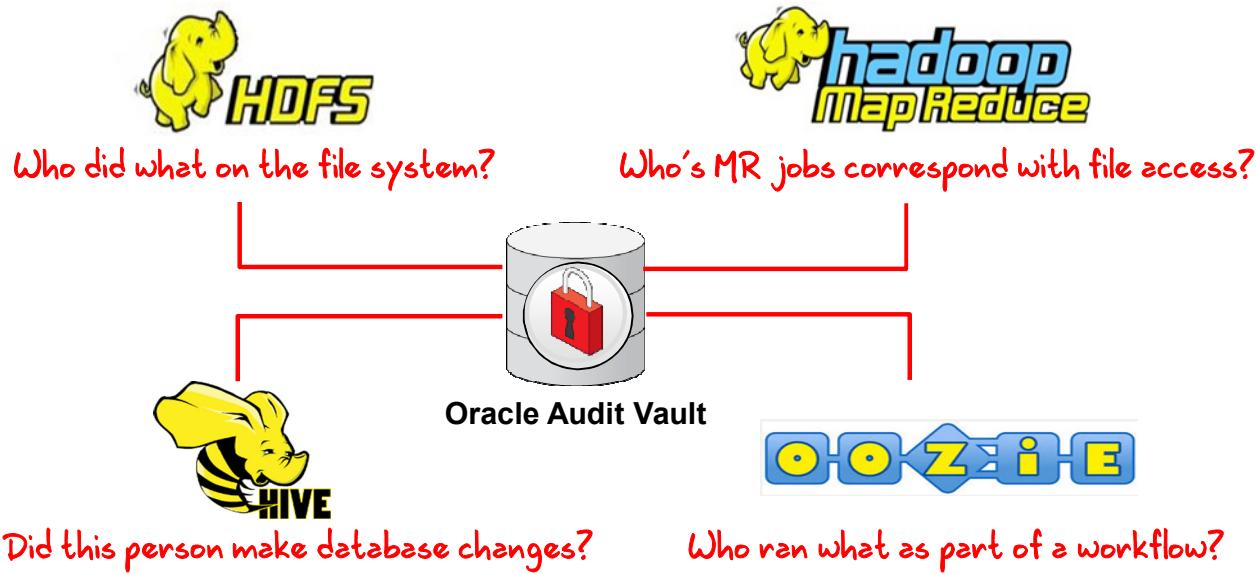


ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Oracle Audit Vault and Database Firewall

Oracle Data Vault audits the following services on the BDA:



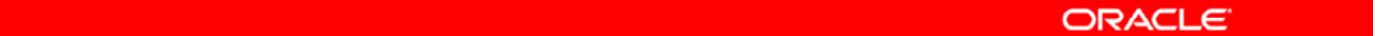
ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Oracle Audit Vault Reporting

| Built-in Reports | Activity Reports | |
|---------------------|-------------------------------------|---|
| Audit Reports | Activity Overview | Digest of all captured audit events for a specified period of time |
| Compliance Reports | Data Access | Details of audited read access to data for a specified period of time |
| Specialized Reports | Data Modification | Details of audited data modifications for a specified period of time |
| Custom Reports | Database Schema Changes | Details of audited DDL activity for a specified period of time |
| Uploaded Reports | All Activity | Details of all captured audit events for a specified period of time |
| Interactive Reports | Failed Logins | Details of audited failed user logins for a specified period of time |
| Report Workflow | User Login and Logout | Details of audited successful user logins and logouts for a specified period of time |
| Report Schedules | Entitlements Changes | Details of audited entitlement related activity for a specified period of time |
| Generated Reports | Audit Settings Changes | Details of observed user activity targeting audit settings for a specified period of time |
| Quick Links | Secured Target Startup and Shutdown | Details of observed startup and shutdown events for a specified period of time |
| Audit Trails | Alert Reports | |
| Enforcement Points | Entitlement Reports | |
| | Stored Procedure Audit Reports | |

- Flexible, interactive reporting capabilities
- User-defined alerts
- Integrated view across all audit sources

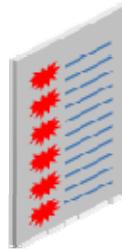
ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

User-Defined Alerts

AV Auditor defines alert conditions:

- Provides a name and description
- Identifies BDA as the Secured Target
- Triggers alert when a given IP address has two failed attempts at accessing unauthorized data



Modify Alert

| | |
|---------------------|---|
| Name * | Attempted Access w/o Authority |
| Secured Target Type | Oracle BDA Hadoop Cluster |
| Severity * | Warning |
| Threshold (times) * | 2 |
| Duration (min) * | 0 |
| Group By (Field) | CLIENT_IP |
| Status * | Enabled |
| Description | User attempted to access to a secure file without proper authorization. |
| Condition * | upper(EVENT_STATUS) = 'FAILURE' |

ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Example: Authorized Data Access

oracle user

```
oracle@bigdatalite:~$ hadoop fs -ls hr
Found 1 items
-rw-r--r-- 3 oracle.hadoop 91 2013-11-25 11:32 hr/salaries.txt
[oracle@bigdatalite ~]$ hadoop fs -get hr/salaries.txt current salaries.txt
[oracle@bigdatalite ~]$ hive
Logging initialized using configuration in jar file:/usr/lib/hive/lib/hive-0.10.0-hadoop2.4.0-hive-history-0.10.0-hadoop2.4.0.jar
hive> select * from hr.salaries;
OK
Hanks    Spielberg      10000000
Spielberg     Cameron 25000000
Cameron Oprah       1250000
Oprah    Boss        54000000
Time taken: 8.491 seconds
hive>
```

- Lists files in the HR folder
- Retrieves salary data
- Queries salary data with Hive



ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Example: Unauthorized Data Access

drevil user

```
File Edit View Search Terminal Help
[drevil@bigdatalite av]$ hadoop fs -get /user/oracle/hr/salaries.txt
get: Permission denied: user=drevil, access=READ, inode="/user/oracle/hr/salaries.txt"
[drevil@bigdatalite av]$ hadoop fs -cat /user/oracle/hr/salaries.txt
cat: Permission denied: user=drevil, access=READ, inode="/user/oracle/hr/salaries.txt"
[drevil@bigdatalite av]$ hadoop fs -cp /user/oracle/hr/salaries.txt /user
cp: Permission denied: user=drevil, access=READ, inode="/user/oracle/hr/salaries.txt"
[drevil@bigdatalite av]$ hadoop fs -tail /user/oracle/hr/salaries.txt
tail: Permission denied: user=drevil, access=READ, inode="/user/oracle/hr/salaries.txt"
[drevil@bigdatalite av]$ hive
Logging initialized using configuration in jar:file:/usr/lib/hive/lib/hive-commo
Hive history file=/tmp/drevil/hive_job_log_9d562c09-051d-4506-8034-9f7b2dd023cf
hive> desc hr.salaries;
OK
emp      string
mgr      string
salary    bigint
Time taken: 3.759 seconds
hive> select emp, salary from hr.salaries;
Total MapReduce jobs = 1
Launching Job 1 out of 1
Number of reduce tasks is set to @ since there's no reduce operator
org.apache.hadoop.security.AccessControlException: Permission denied: user=drevi
```

Fails to:

- Retrieve salary data
- Display salary data
- Query salary data with Hive



ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Audit Vault Dashboard

DrEvil Triggers Alert

The screenshot shows the Oracle Audit Vault Server interface. On the left, under 'Recently Raised Alerts', there is a single entry: 'Attempted Access w/o Authority' (highlighted with a red box). On the right, the 'Alert Details' window is open, displaying the following information:

| Name | Attempted Access w/o Authority | | | | | | | | | | | | | | | | | | | | | |
|--|--------------------------------|---------------------|-----------------------|-----------------------|------------|-----------------|--------|--------------|---|-------|---------------------|-----------------------|-----------------------|------|---------|---|-------|---------------------|-----------------------|-----------------------|------|---------|
| Alert Raised | 11/27/2013 4:07:52 PM | | | | | | | | | | | | | | | | | | | | | |
| First Event Time | 11/27/2013 4:02:12 PM | | | | | | | | | | | | | | | | | | | | | |
| Threshold (times) | 2 | | | | | | | | | | | | | | | | | | | | | |
| Duration (min) | 0 | | | | | | | | | | | | | | | | | | | | | |
| Severity | Warning | | | | | | | | | | | | | | | | | | | | | |
| Group By | CLIENT_IP | | | | | | | | | | | | | | | | | | | | | |
| Status | New | | | | | | | | | | | | | | | | | | | | | |
| Description User attempted to access to a secure file without proper authorization. | | | | | | | | | | | | | | | | | | | | | | |
| Condition upper(EVENT_STATUS) = FAILURE | | | | | | | | | | | | | | | | | | | | | | |
| Notes New Note | | | | | | | | | | | | | | | | | | | | | | |
| Event | | | | | | | | | | | | | | | | | | | | | | |
| <table border="1"> <thead> <tr> <th></th> <th>Secured Target</th> <th>User Name</th> <th>Event Time</th> <th>Collection Time</th> <th>Action</th> <th>Event Status</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>sca22</td> <td>drevil(AUTH:SIMPLE)</td> <td>11/27/2013 4:02:44 PM</td> <td>11/27/2013 4:07:50 PM</td> <td>OPEN</td> <td>FAILURE</td> </tr> <tr> <td>2</td> <td>sca22</td> <td>drevil(AUTH:SIMPLE)</td> <td>11/27/2013 4:02:12 PM</td> <td>11/27/2013 4:07:50 PM</td> <td>OPEN</td> <td>FAILURE</td> </tr> </tbody> </table> | | | Secured Target | User Name | Event Time | Collection Time | Action | Event Status | 1 | sca22 | drevil(AUTH:SIMPLE) | 11/27/2013 4:02:44 PM | 11/27/2013 4:07:50 PM | OPEN | FAILURE | 2 | sca22 | drevil(AUTH:SIMPLE) | 11/27/2013 4:02:12 PM | 11/27/2013 4:07:50 PM | OPEN | FAILURE |
| | Secured Target | User Name | Event Time | Collection Time | Action | Event Status | | | | | | | | | | | | | | | | |
| 1 | sca22 | drevil(AUTH:SIMPLE) | 11/27/2013 4:02:44 PM | 11/27/2013 4:07:50 PM | OPEN | FAILURE | | | | | | | | | | | | | | | | |
| 2 | sca22 | drevil(AUTH:SIMPLE) | 11/27/2013 4:02:12 PM | 11/27/2013 4:07:50 PM | OPEN | FAILURE | | | | | | | | | | | | | | | | |

Alert Details

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Interactive Reporting

Who is accessing salary data?

- Define custom reports
- Full range of filtering capabilities
- Custom formatting rules highlight exceptions

| Activity Overview Report | | | | | | | |
|--|------------|-----------------------------|--------------|---------------------|---------------|--------------|--|
| <input type="text"/> Go Actions ▾ | | | | | | | |
| <input checked="" type="checkbox"/> Row text contains 'salaries' <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> Event Time is in the last 24 hours <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> Secured Target Name <input checked="" type="checkbox"/> <input checked="" type="checkbox"/> Failure <input checked="" type="checkbox"/> | | | | | | | |
| Secured Target Name : scaj22 | | | | | | | |
| Event Time ▾ | Event Name | Target Object | Event Status | User Name | Client IP | Command Text | |
| 11/27/2013 4:31:55 PM | HDFS | /user/oracle/hrsalaries.txt | SUCCESS | oracle(AUTH:SIMPLE) | 10.154.167.76 | open | |
| 11/27/2013 4:31:55 PM | HDFS | /user/oracle/hrsalaries.txt | SUCCESS | oracle(AUTH:SIMPLE) | 10.154.167.76 | open | |
| 11/27/2013 4:31:18 PM | HDFS | /user/oracle/hrsalaries.txt | SUCCESS | oracle(AUTH:SIMPLE) | 10.154.167.76 | open | |
| 11/27/2013 4:09:59 PM | HDFS | /user/oracle/hrsalaries.txt | FAILURE | drevil(AUTH:SIMPLE) | 10.154.167.76 | open | |
| 11/27/2013 4:03:12 PM | HDFS | /user/oracle/hrsalaries.txt | FAILURE | drevil(AUTH:SIMPLE) | 10.154.167.76 | open | |
| 11/27/2013 4:02:59 PM | HDFS | /user/oracle/hrsalaries.txt | FAILURE | drevil(AUTH:SIMPLE) | 10.154.167.76 | open | |
| 11/27/2013 4:02:44 PM | HDFS | /user/oracle/hrsalaries.txt | FAILURE | drevil(AUTH:SIMPLE) | 10.154.167.76 | open | |
| 11/27/2013 4:02:12 PM | HDFS | /user/oracle/hrsalaries.txt | FAILURE | drevil(AUTH:SIMPLE) | 10.154.167.76 | open | |
| 11/27/2013 3:59:44 PM | HDFS | /user/oracle/hrsalaries.txt | SUCCESS | oracle(AUTH:SIMPLE) | 10.154.167.76 | open | |
| 11/27/2013 3:59:44 PM | HDFS | /user/oracle/hrsalaries.txt | SUCCESS | oracle(AUTH:SIMPLE) | 10.154.167.76 | open | |
| 11/27/2013 3:59:44 PM | HDFS | /user/oracle/hrsalaries.txt | SUCCESS | oracle(AUTH:SIMPLE) | 10.154.167.76 | open | |

ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Cloudera Navigator

Cloudera Navigator audits the following activity:

- HDFS data accessed through HDFS, Hive, HBase, and Impala services
- Hive, HBase, and Impala operations
- Hive metadata definition
- Sentry access



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Cloudera Navigator Reporting

- Captures details of Hadoop activity across services
- Easily identifies improper access attempts

The screenshot shows the Cloudera Navigator Reporting interface with the title "Recent Denied Accesses". The table lists the following data:

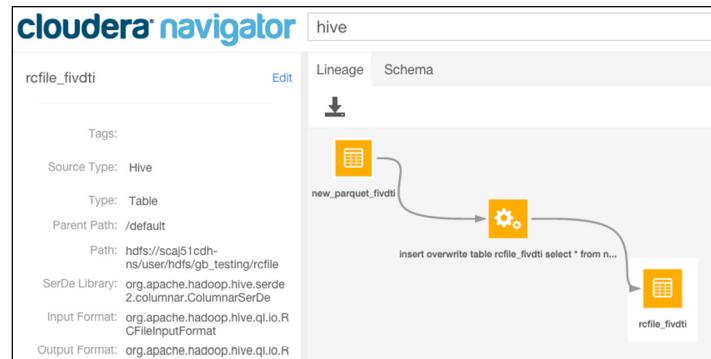
| Timestamp | Username | IP Address | Service Name | Operation | Resource |
|--------------------|--------------------|-----------------|--------------|-----------------|------------------------------------|
| Mar 8 2015 6:55 PM | lucy | 192.168.42.122 | hive | QUERY | : |
| Mar 8 2015 6:55 PM | lucy | 192.168.42.122 | hive | QUERY | marketing_db:movieapp_log_avro |
| Mar 8 2015 6:46 PM | bob@DEV.ORACLE.COM | 10.154.131.151 | hdfs | listStatus | /user/hive/warehouse/marketing_db: |
| Mar 8 2015 6:33 PM | bob | 10.154.131.151 | hive | SWITCHDATABASE | marketing_db: |
| Mar 8 2015 5:58 PM | bob | 10.154.152.117 | hive | GRANT_PRIVILEGE | : |
| Mar 8 2015 5:58 PM | bob | /192.168.42.122 | sentry | GRANT_PRIVILEGE | |
| Mar 8 2015 5:56 PM | oracle | 10.154.152.117 | hive | SWITCHDATABASE | marketing_db: |
| Mar 8 2015 5:51 PM | bob | 10.154.152.117 | hive | GRANT_PRIVILEGE | : |
| Mar 8 2015 5:51 PM | bob | /192.168.42.122 | sentry | GRANT_PRIVILEGE | |
| Mar 8 2015 3:53 PM | oracle | 192.168.42.122 | hive | SWITCHDATABASE | : |
| Mar 8 2015 3:22 PM | oracle | 10.154.152.117 | hive | CREATEDATABASE | foobar: |
| Mar 8 2015 2:58 PM | bob | 10.154.152.117 | hive | SWITCHDATABASE | marketing: |
| Mar 8 2015 2:42 PM | oracle | 10.154.152.117 | hive | SWITCHDATABASE | : |

ORACLE

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Cloudera Navigator Lineage Analysis

- Enables users to understand source of data and transformations
- Tracks lineage of:
 - HDFS files and directories
 - Hive tables and column
 - MapReduce and YARN jobs
 - Hive queries
 - Pig scripts
 - Sqoop jobs
 - Oozie workflows



ORACLE®

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Useful for consequences of purging or modifying a set of data entities.

Encryption

BDA automatically configures two types of encryption:

- Network encryption
- Data-at-rest encryption



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Network Encryption

BDA supports network encryption for key activities—preventing network sniffing between computers:

- Automatically configured by Mammoth:
 - Cloudera Manager Server communicating with Agents
 - Hadoop HDFS data transfers
 - Hadoop internal RPC communications
- Configured using Cloudera Manager:
 - Cloudera Manager web interface
 - Hadoop web UIs and web services
 - Hadoop YARN/MapReduce shuffle transfers



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Data at Rest Encryption

On-disk encryption protects data at rest. BDA offers two types of at rest data encryption:

- eCryptFS
 - All data is automatically encrypted at the OS file system level.
 - If a disk is removed from the server, the encrypted data is protected until it is installed into a server and a password is provided.
- HDFS Transparent Encryption
 - Selected HDFS folders are configured to transparently encrypt files and subdirectories.
 - Protect data from being viewed or copied outside of the Hadoop file system.



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

Summary

In this lesson, you should have learned about securing data on the Big Data Appliance.



Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

| Term | Description |
|---|--|
| Apache Flume | A distributed service for collecting and aggregating data from almost any source into a data store such as HDFS or HBase. |
| Apache Hadoop | A batch processing infrastructure that stores files and distributes work across a group of servers. Oracle Big Data Appliance uses Cloudera's Distribution including Apache Hadoop (CDH). |
| Apache HBase | An open-source, column-oriented database that provides random, read/write access to large amounts of sparse data stored in a CDH cluster. It provides fast lookup of values by key and can perform thousands of insert, update, and delete operations per second. |
| Apache Hive | An open-source data warehouse in CDH that supports data summarization, ad hoc querying, and data analysis of data stored in HDFS. It uses a SQL-like language called HiveQL. An interpreter generates MapReduce code from the HiveQL queries. By using Hive, you can avoid writing MapReduce programs in Java. |
| Apache Mahout | Apache Mahout is a machine learning library that includes core algorithms for clustering, classification, and batch-based collaborative filtering. |
| Apache Sentry | Integrates with the Hive and Impala SQL-query engines to provide fine-grained authorization to data and metadata stored in Hadoop. |
| Apache Solr | Provides an enterprise search platform that includes full-text search, faceted search, geospatial search, and hit highlighting. |
| Apache Spark | A fast engine for processing large-scale data. It supports Java, Scala, and Python applications. Because it provides primitives for in-memory cluster computing, it is particularly suited to machine-learning algorithms. It promises performance up to 100 times faster than MapReduce. |
| Apache Sqoop | A command-line tool that imports and exports data between HDFS or Hive and structured databases. The name Sqoop comes from "SQL to Hadoop." Oracle R Advanced Analytics for Hadoop uses the Sqoop executable to move data between HDFS and Oracle Database. |
| Apache Whirr | Apache Whirr is a set of libraries for running cloud services. |
| Apache YARN | An updated version of MapReduce, also called MapReduce 2. The acronym stands for Yet Another Resource Negotiator. |
| Balancer | A service that ensures that all nodes in the cluster store about the same amount of data, within a set range. Data is balanced over the nodes in the cluster, not over the disks in a node. |
| Checkpoint Node | The Checkpoint Node is a secondary NameNode that can be imported (if necessary) to the primary NameNode. It performs periodic checkpoints of the namespace and helps minimize the size of the log stored at the NameNode containing changes to the HDFS. |
| Cloudera Hue | Hadoop User Experience, a web user interface in CDH that includes several applications, including a file browser for HDFS, a job browser, an account management tool, a MapReduce job designer, and Hive wizards. Cloudera Manager runs on Hue. |
| Cloudera Impala | A massively parallel processing query engine that delivers better performance for SQL queries against data in HDFS and HBase, without moving or transforming the data. |
| Cloudera Manager | Cloudera Manager enables you to monitor, diagnose, and manage CDH services in a cluster. The Cloudera Manager agents on Oracle Big Data Appliance also provide information to Oracle Enterprise Manager, which you can use to monitor both software and hardware. |
| Cloudera Navigator | Verifies access privileges and audits access to data stored in Hadoop, including Hive metadata and HDFS data accessed through HDFS, Hive, or HBase. |
| Cloudera's Distribution including Apache Hadoop (CDH) | Cloudera's Distribution including Apache Hadoop, the version of Apache Hadoop and related components installed on Oracle Big Data Appliance. |

| Term | Description |
|--|---|
| cluster | A group of servers on a network that are configured to work together. A server is either a master node or a worker node. All servers in an Oracle Big Data Appliance rack form a cluster. Servers 1, 2, and 3 are master nodes. Servers 4 to 18 are worker nodes. |
| DataNode | A server in a CDH cluster that stores data in HDFS. A DataNode performs file system operations assigned by the NameNode. |
| edits Log File | The Namenode stores the file system modifications in an edit log file (<code>edits</code>). |
| Flume | See Apache Flume. |
| fsck | A utility to diagnose health of the file system in order to find missing files or blocks. |
| Fuse DFS | CDH 5 includes a FUSE (Filesystem in Userspace) interface into HDFS. The <code>hadoop-hdfs-fuse</code> package enables you to use your HDFS cluster as if it were a traditional file system on Linux. Is also called NFS Mount. |
| HDFS | Hadoop Distributed File System, an open-source file system designed to store extremely large data files (megabytes to petabytes) with streaming data access patterns. HDFS splits these files into data blocks and distributes the blocks across a CDH cluster. When a data set is larger than the storage capacity of a single computer, then it must be partitioned across several computers. A distributed file system can manage the storage of a data set across a network of computers. |
| Hive | See Apache Hive |
| HiveQL | A SQL-like query language used by Hive. |
| Hue (Hadoop User Experience) | See Cloudera Hue. |
| Impala | See Cloudera Impala. |
| JobTracker | A service that assigns tasks to specific nodes in the CDH cluster, preferably those nodes storing the data. MRv1 only. |
| JSON | JSON (JavaScript Object Notation) is a lightweight data-interchange format. It is easy for humans to read and write. It is easy for machines to parse and generate. JSON is a text format that is completely language independent but uses conventions that are familiar to programmers of the C-family of languages, including C, C++, C#, Java, JavaScript, Perl, Python, and many others. These properties make JSON an ideal data-interchange language. |
| Kerberos | A network authentication protocol that helps prevent malicious impersonation. Apache Hadoop is not an inherently secure system. It is protected only by network security. After a connection is established, a client has full access to the system. To counterbalance this open environment, Oracle Big Data Appliance supports Kerberos security as a software installation option. Kerberos was developed at the Massachusetts Institute of Technology (MIT). |
| LDAP (Lightweight Directory Access Protocol) | The Lightweight Directory Access Protocol is an open, vendor-neutral, industry standard application protocol for accessing and maintaining distributed directory information services over an Internet Protocol network. Wikipedia |
| Mahout | See Apache Mahout. |
| Mammoth Utility | Mammoth is a command-line utility for installing and configuring the Oracle Big Data Appliance software. |

| Term | Description |
|--|--|
| MapReduce | <p>A parallel programming model for processing data on a distributed system. Two versions of MapReduce are available, MapReduce 1 and YARN (MapReduce 2). The default version on Oracle Big Data Appliance 3.0 and later is YARN. A MapReduce program contains these functions:</p> <ul style="list-style-type: none"> Mappers: Process the records of the data set. Reducers: Merge the output from several mappers. Combiners: Optimizes the result sets from the mappers before sending them to the reducers (optional and not supported by all applications). |
| MySQL Database | See Oracle NoSQL Database. |
| NameNode | The NameNode is the most critical process because it keeps track of the location of all data. Without a healthy NameNode, the entire cluster fails. |
| NameNode | A service that maintains a directory of all files in HDFS and tracks where data is stored in the CDH cluster. |
| Node | A server in a CDH cluster. See also cluster. |
| NodeManager | A service that runs on each node and executes the tasks assigned to it by the ResourceManager. YARN only. See also ResourceManager and YARN. |
| Oozie | An open-source workflow and coordination service for managing data processing jobs in CDH. |
| Oracle Audit Vault and Database Firewall | <p>Oracle Audit Vault and Database Firewall (AVDF) secures databases and other critical components of IT infrastructure (such as operating systems) in these key ways:</p> <ol style="list-style-type: none"> 1. Provides a database firewall that can monitor activity and/or block SQL statements on the network based on a firewall policy. 2. Collects audit data, and makes it available in audit reports. 3. Provides dozens of built-in, customizable activity and compliance reports, and lets you proactively configure alerts and |
| Oracle Data Mining | A component of the Oracle Advanced Analytics Option. It provides data mining and text mining on data inside and outside (via external tables) Oracle Database. It enables predictive analysis in a graphical user environment and through the Data Mining SQL API. |
| Oracle BigDataLite 4.0 VM | Oracle Big Data Lite Virtual Machine provides an integrated environment to help you get started with the Oracle Big Data platform. Many Oracle Big Data platform components have been installed and configured - allowing you to begin using the system right away. |
| Oracle Big Data SQL | Oracle Big Data SQL, which is a licensed option on Oracle BDA, enables SQL query access to a variety of big data formats, including Apache Hive, HDFS, Oracle NoSQL Database, and Apache HBase. Using Big Data SQL, users can join Oracle Database data with big data formats to provide dynamic, integrated analysis without moving the data. The Big Data SQL option also includes the Copy to BDA feature. |
| Oracle NoSQL Database | A distributed key-value database that supports fast querying of the data, typically by key lookup. A SQL-based relational database management system. Cloudera Manager, Oracle Data Integrator, Hive, and Oozie use MySQL Database as a metadata repository on Oracle Big Data Appliance. |
| Oracle R Distribution | An Oracle-supported distribution of the R open-source language and environment for statistical analysis and graphing. |
| Oracle R Enterprise | A component of the Oracle Advanced Analytics Option. It enables R users to run R commands and scripts for statistical and graphical analyses on data stored in an Oracle database. |

| Term | Description |
|-----------------|--|
| Pig | An open-source platform for analyzing large data sets that consists of the following: Pig Latin scripting language Pig interpreter that converts Pig Latin scripts into MapReduce jobs Pig runs as a client application. |
| Puppet | A configuration management tool for deploying and configuring software components across a cluster. The Oracle Big Data Appliance initial software installation uses Puppet. The Puppet tool consists of these components: puppet agents, typically just called puppets; the puppet master server; a console; and a cloud provisioner. See also puppet agent and puppet master . |
| puppet agent | A service that primarily pulls configurations from the puppet master and applies them. Puppet agents run on every server in Oracle Big Data Appliance. |
| puppet master | A service that primarily serves configurations to the puppet agents. |
| ResourceManager | A service that assigns tasks to specific nodes in the CDH cluster, preferably those nodes storing the data. YARN only. |
| Sentry | See Apache Sentry. |
| Spark | See Apache Spark. |
| Table | In Hive, all files in a directory stored in HDFS. |
| TaskTracker | A service that runs on each node and executes the tasks assigned to it by the JobTracker service. MRv1 only. See also JobTracker . |
| Whirr | See Apache Whirr. |
| ZooKeeper | A MapReduce 1 centralized coordination service for CDH distributed processes that maintains configuration information and naming, and provides distributed synchronization and group services. |

| Web Site | URL |
|--|--|
| Apache Flume | http://flume.apache.org |
| Apache Hadoop | http://hadoop.apache.org |
| Apache Hadoop YARN | http://hadoop.apache.org/docs/r2.3.0/hadoop-yarn/hadoop-yarn-site/YARN.html |
| Apache Hbase | http://hbase.apache.org |
| Apache Hive | https://hive.apache.org |
| Apache Mahout | http://mahout.apache.org |
| Apache Oozie | http://oozie.apache.org |
| Apache Pig | http://pig.apache.org |
| Apache Solr | http://lucene.apache.org/solr/ |
| Apache Spark | https://spark.apache.org |
| Apache Spark | https://spark.apache.org |
| Apache Sqoop | http://sqoop.apache.org |
| Apache Whirr | https://whirr.apache.org |
| Apache ZooKeeper | http://zookeeper.apache.org |
| Cloudera Documentation | http://www.cloudera.com/content/support/en/documentation.html |
| Cloudera Hue | http://www.cloudera.com/content/cloudera-content/cloudera-docs/CDH4/4.2.0/Hue-2-User-Guide/hue2.html |
| Cloudera Impala | http://www.cloudera.com/content/cloudera/en/products-and-services/cdh/impala.html |
| Cloudera Manager | http://www.cloudera.com/content/cloudera/en/products-and-services/cloudera-enterprise/cloudera-manager.html |
| Cloudera Search | http://www.cloudera.com/content/cloudera/en/products-and-services/cdh/search.html |
| Document Object Model (DOM) | http://www.w3.org/DOM/ |
| fuse-dfs | http://fuse.sourceforge.net/ |
| Information Management and Big Data (Oracle White Paper) | http://www.oracle.com/ocom/groups/public/@otn/documents/webcontent/2297765.pdf https://docs.oracle.com/cd/E55905_01/doc.40/e55796/mammoth.htm |
| Mammoth | |
| Oracle Audit Vault and Database Firewall (AVDF) Administrator's Guide | http://docs.oracle.com/cd/E37100_01/doc.121/e27776/intro.htm#SIGAD40487 |
| Oracle Advanced Analytics (Oracle Data Mining and Oracle R Enterprise) | http://www.oracle.com/us/products/database/options/advanced-analytics/overview/index.html |
| Oracle Big Data | https://www.oracle.com/bigdata/index.html |
| Oracel Big Data Discovery | https://www.oracle.com/big-data/big-data-discovery/index.html |
| Oracle Big Data Appliance Administering | http://docs.oracle.com/cd/E55905_01/doc.40/e55814/admin.htm#BIGUG132 |
| Oracle Big Data Appliance Documentation | http://www.oracle.com/technetwork/database/bigdata-appliance/documentation/index.html |
| Oracle Big Data Appliance Utilities | https://docs.oracle.com/cd/E55905_01/doc.40/e55796/utilities.htm#BIGOG76779 |
| Oracle Big Data Connectors (white paper) | http://www.oracle.com/technetwork/database/database-technologies/bdc/hadoop-loader/connectors-hdfs-wp-1674035.pdf |

Oracle Big Data Documentation
Oracle Big Data Library on OLL
Oracle Big Data Lite Virtual Machine
Oracle Big Data SQL (incl Copy to BDA)
Oracle Certification Matrix
Oracle Data Integrator
Oracle GoldenGate
Oracle Documentation
Oracle Education and Training
Oracle Learning Library (OLL)
Oracle Loader for Hadoop
Oracle NoSQL Database
Oracle Technology Network
Scala
W3C Namespaces in XML
W3C XML Schema
W3C XML Schema Datatypes
W3C XML Schema Structures
W3C XML Technology
W3C XML XPath Version 1.0
XML 1.0 W3C Recommendation
XML Schema Part 0: Primer
XPath Tutorial
XQuery Tutorial
XSL Transformations (XSLT)

<http://www.oracle.com/technetwork/database/bigdata-appliance/documentation/index.html>
https://apex.oracle.com/pls/apex/f?p=44785:141:0:::P141_PAGE_ID,P141_SECTION_ID:27,617
<http://www.oracle.com/technetwork/database/bigdata-appliance/oracle-bigdatalite-2104726.html#hol>
<http://www.oracle.com/us/products/database/big-data-sql/overview/index.html>
<http://www.oracle.com/us/products/database/big-data-connectors/certifications/index.html>
<http://www.oracle.com/technetwork/middleware/data-integrator/overview/index.html>
<http://www.oracle.com/us/products/middleware/data-integration/goldengate/overview/index.html>
<http://www.oracle.com/technetwork/indexes/documentation/index.html>
<http://education.oracle.com>
<http://www.oracle.com/oll>
http://docs.oracle.com/cd/E55905_01/doc.40/e55819/olh.htm
<http://www.oracle.com/us/products/database/nosql/overview/index.html>
<http://www.oracle.com/technetwork/index.html>
<http://www.scala-lang.org/>
<http://www.w3.org/TR/xml-names/>
<http://www.w3.org/XMLSchema>
<http://www.w3.org/TR/xmlschema-2/>
<http://www.w3.org/TR/xmlschema-1/>
<http://www.w3.org/standards/xml/>
<http://www.w3.org/TR/xpath/>
<http://www.w3.org/TR/xml/>
<http://www.w3.org/TR/xmlschema-0/>
<http://www.zvon.org/xxl/XPathTutorial/General/examples.html>
<http://www.w3.org/TR/xquery/>
<http://www.w3.org/TR/xslt>