

Oracle® Big Data Appliance

Software User's Guide

Release 1 (1.0)

E25961-04

June 2012

Oracle Big Data Appliance Software User's Guide, Release 1 (1.0)

E25961-04

Copyright © 2012, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

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

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Cloudera, Cloudera CDH, and Cloudera Manager are registered and unregistered trademarks of Cloudera, Inc.

Contents

Preface	v
Audience	v
Documentation Accessibility	v
Related Documents	v
Conventions	vi
1 Introducing Oracle Big Data Appliance	
What Is Big Data?	1-1
High Variety	1-1
High Complexity	1-2
High Volume	1-2
High Velocity	1-2
The Oracle Big Data Solution	1-2
Software for Big Data	1-3
Software Component Overview	1-4
Acquiring Data For Analysis	1-4
Hadoop Distributed File System	1-5
Oracle NoSQL Database	1-5
Organizing Big Data	1-6
MapReduce	1-6
Oracle R Support for Big Data	1-6
Oracle Big Data Connectors	1-7
Analyzing and Visualizing Big Data	1-9
2 Administering Oracle Big Data Appliance	
Managing CDH Operations	2-1
Monitoring the Status of Oracle Big Data Appliance	2-2
Performing Administrative Tasks	2-2
Collecting Diagnostic Information	2-3
Using Hadoop Monitoring Utilities	2-4
Monitoring the JobTracker	2-4
Monitoring the TaskTracker	2-5
Providing Remote Client Access to CDH	2-6
Installing CDH on the Client System	2-6
Configuring CDH	2-7

Managing User Accounts	2-8
Software Layout	2-9
Software Components	2-9
Logical Disk Layout	2-10
Software Services	2-11
Parent Services.....	2-11
Child Services	2-12
Software Services Distribution	2-12
Effects of Hardware on Software Availability	2-13
Node01: Critical for All Services	2-14
Node02 to Node03: Required for Some Services	2-14
Node04 to Node18: Optional for All Services	2-14
Security on Oracle Big Data Appliance	2-15
CDH Security	2-15
Port Numbers Used on Oracle Big Data Appliance.....	2-15
Security of Software Components	2-16

Index

Preface

Oracle Big Data Appliance Software User's Guide describes how to manage and use the installed software.

Audience

This guide is intended for users of Oracle Big Data Appliance including:

- Application developers
- Data analysts
- Data scientists
- Database administrators
- Systems administrators

Oracle Big Data Appliance Software User's Guide introduces the terminology and concepts necessary to discuss Oracle Big Data Appliance. However, you must acquire the necessary information about administering Hadoop clusters and writing MapReduce programs from other sources.

Documentation Accessibility

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

Access to Oracle Support

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

Related Documents

For more information, see the following manuals:

- *Oracle Big Data Appliance Owner's Guide*
- *Oracle Big Data Connectors User's Guide*

Conventions

The following text conventions are used in this document:

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

Introducing Oracle Big Data Appliance

This chapter presents an overview of Oracle Big Data Appliance and describes the software installed on the system. This chapter contains the following topics:

- [What Is Big Data?](#)
- [The Oracle Big Data Solution](#)
- [Software for Big Data](#)
- [Acquiring Data For Analysis](#)
- [Organizing Big Data](#)
- [Analyzing and Visualizing Big Data](#)

What Is Big Data?

Using transactional data as the source of business intelligence has been commonplace for many years. As digital technology and the World Wide Web spread into every aspect of modern life, other sources of data can make important contributions to business decision making. Many businesses are looking to these new data sources. They are finding opportunities in analyzing data that until just recently was simply thrown away.

Big data is characterized by:

- [High Variety](#)
- [High Complexity](#)
- [High Volume](#)
- [High Velocity](#)

These characteristics pinpoint the challenges in deriving value from big data, and the differences between big data and traditional data sources, which primarily provide highly structured, transactional data.

High Variety

Big data is derived from a variety of sources, such as:

- **Equipment sensors:** Medical, manufacturing, transportation, and other machine sensor transmissions
- **Machine generated:** Call detail records, web logs, smart meter readings, Global Positioning System (GPS) transmissions, and trading systems records

- Social media: Data streams from social media sites like Facebook and blogging sites like Twitter

Analysts can mine this data repeatedly as they devise new ways of extracting meaningful insights. What may seem irrelevant today may prove to be highly pertinent to your business tomorrow.

Challenge: Flexible systems to handle this high variety.

High Complexity

As the variety of data types increases, the complexity of the system increases. The complexity of data types also increases in big data because of its low structure.

Challenge: Finding solutions that apply across a broad range of data types.

High Volume

Social media can generate terabytes of daily data. Equipment sensors and other machines may generate that much data in less than an hour.

Even traditional data sources for data warehouses, such as customer profiles from CRM systems, transactional ERP data, store transactions, and general ledger data, have increased ten fold in volume over the past decade.

Challenge: Scalability and ease in growing the system.

High Velocity

Huge numbers of sensors, web logs, and other machine sources generate data continuously and at a much higher speed than traditional sources, such as individuals entering orders into a transactional database.

Challenge: Handling the data at high speed without stressing the structured systems.

The Oracle Big Data Solution

Oracle Big Data Appliance is an engineered system comprising both hardware and software components. The hardware is optimized to run the enhanced big data software components.

Oracle Big Data Appliance delivers:

- Complete and optimized solution for big data
- Single-vendor support for both hardware and software
- Easy-to-deploy solution
- Tight integration with Oracle Database and Oracle Exadata Database Machine

Oracle provides a big data platform that captures, organizes, and supports deep analytics on extremely large, complex data streams flowing into your enterprise from a large number of data sources. You can choose the best storage and processing location for your data depending on its structure, workload characteristics, and end-user requirements.

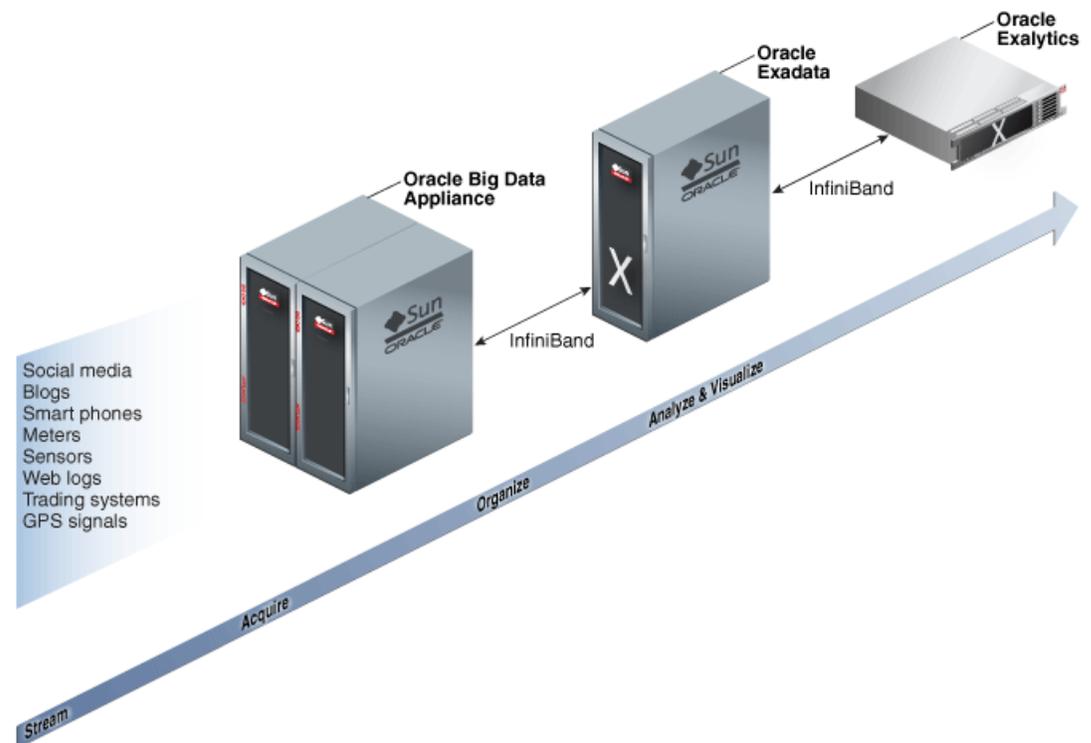
Oracle Database enables all data to be accessed and analyzed by a large user community using identical methods. By adding Oracle Big Data Appliance in front of Oracle Database, you can bring new sources of information to an existing data warehouse. Oracle Big Data Appliance is the platform for acquiring and organizing

big data so the relevant portions with true business value can be analyzed in Oracle Database.

For maximum speed and efficiency, Oracle Big Data Appliance can be connected to Oracle Exadata Database Machine running Oracle Database. Oracle Exadata Database Machine provides outstanding performance in hosting data warehouses and transaction processing databases. Moreover, Oracle Exadata Database Machine can be connected to Oracle Exalytics In-Memory Machine for the best performance of business intelligence and planning applications. Connections between these engineered systems are highly parallel using InfiniBand, which enables high-speed data transfer for batch or query workloads.

Figure 1–1 shows the relationships among these engineered systems.

Figure 1–1 Oracle Engineered Systems for Big Data



Software for Big Data

Oracle Linux operating system and Cloudera's Distribution including Apache Hadoop (CDH) underlie all other software components installed on Oracle Big Data Appliance. CDH is an integrated stack of components that have been tested and packaged to work together.

CDH has a batch processing infrastructure that can store files and distribute work across a set of computers. Data is computed on the same computer where it is stored. In a single Oracle Big Data Appliance Rack, CDH distributes the files and workload across 18 servers, which compose a cluster. Each server in the cluster is a node.

The software framework consists of these primary components:

- File system: The Hadoop Distributed File System (HDFS) is a highly scalable file system that stores large files across multiple servers. It achieves reliability by

replicating data across multiple servers without RAID technology. It runs on top of the Linux file system on Oracle Big Data Appliance.

- MapReduce engine: The MapReduce engine provides a platform for the massively parallel execution of algorithms written in Java.
- Administrative framework: Cloudera Manager is a comprehensive administrative tool for CDH

CDH is written in Java, and Java is the language for applications development. However, several CDH utilities and other software available on Oracle Big Data Appliance provide graphical and web-based interfaces for ease of use.

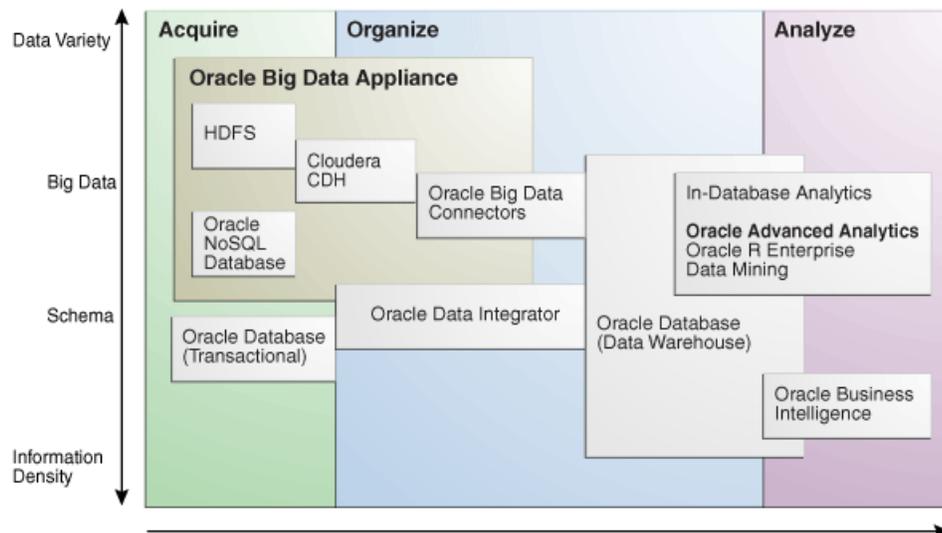
Software Component Overview

The major software components perform three basic tasks:

- Acquire
- Organize
- Analyze and Visualize

The best tool for each task depends on the density of the information and the degree of structure. [Figure 1-2](#) shows the relationships among the tools and identifies the tasks they perform.

Figure 1-2 Oracle Big Data Appliance Software Overview



Acquiring Data For Analysis

Oracle Big Data Appliance provides these facilities for capturing and storing big data:

- Hadoop Distributed File System (HDFS)
- Oracle NoSQL Database

Databases used for Online Transactional Processing (OLTP) are the traditional data sources for data warehouses. The Oracle solution enables you to analyze traditional data stores with big data in the same Oracle data warehouse. Relational data continues to be an important source of business intelligence, while running on separate hardware from Oracle Big Data Appliance Rack.

Hadoop Distributed File System

Cloudera's Distribution including Apache Hadoop (CDH) on Oracle Big Data Appliance uses the Hadoop Distributed File System (HDFS). HDFS stores extremely large files containing record-oriented data. It splits large data files into chunks of 64 MB, and replicates the chunk across three different nodes in the cluster. The size of the chunks and the number of replications are configurable.

Chunking enables HDFS to store files that are larger than the physical storage of one server. It also allows the data to be processed in parallel across multiple machines with multiple processors, all working on data that is stored locally. Replication assures the high availability of the data: if a server fails, the other servers automatically take over its work load.

HDFS is typically used to store all the various types of big data.

See Also:

- For conceptual information about Hadoop technologies, refer to this third-party publication:

Hadoop: The Definitive Guide, Second Edition by Tom White.
Published by O'Reilly Media Inc., 2010. (ISBN: 978-1449389734)

- For documentation about Cloudera's Distribution including Apache Hadoop, see the Cloudera library at

<http://oracle.cloudera.com/>

Oracle NoSQL Database

Oracle NoSQL Database is a distributed key-value database, built on the proven storage technology of Berkeley DB Java Edition. Whereas HDFS stores unstructured data in very large files, Oracle NoSQL Database indexes the data and supports transactions. But unlike Oracle Database, which stores highly structured data, Oracle NoSQL Database has relaxed consistency rules, no schema structure, and only modest support for joins, particularly across storage nodes.

NoSQL databases, or "Not Only SQL" databases, have developed over the past decade specifically for storing big data. However, they vary widely in implementation. Oracle NoSQL Database has these characteristics:

- Uses a system-defined, consistent hash index for data distribution
- Supports high availability through replication
- Provides single record, single operation transactions with relaxed consistency guarantees
- Provides a Java API

Oracle NoSQL Database is designed to provide highly reliable, scalable, predictable, and available data storage. The key-value pairs are stored in shards or partitions (that is, subsets of data) based on a primary key. Data on each shard is replicated across multiple storage nodes to ensure high availability. Oracle NoSQL Database supports fast querying of the data, typically by key lookup.

An intelligent driver links the NoSQL database with client applications and provides access to the requested key-value on the storage node with the lowest latency.

Oracle NoSQL Database includes hashing and balancing algorithms to ensure proper data distribution and optimal load balancing, replication management components to

handle storage node failure and recovery, and an easy-to-use administrative interface to monitor the state of the database.

Oracle NoSQL Database is typically used to store customer profiles and similar data for identifying and analyzing big data. For example, you might log in to a website and see advertisements based on your stored customer profile (a record in Oracle NoSQL Database) and your recent activity on the site (web logs currently streaming into HDFS).

Oracle NoSQL Database is an optional component of Oracle Big Data Appliance. It is always installed, but may not be activated during installation of the software.

See Also:

- *Oracle NoSQL Database Getting Started Guide* at <http://docs.oracle.com/cd/NOSQL/html/index.html>
- *Oracle Big Data Appliance Licensing Information*

Organizing Big Data

Oracle Big Data Appliance provides several ways of organizing, transforming, and reducing big data for analysis:

- [MapReduce](#)
- [Oracle R Support for Big Data](#)
- [Oracle Big Data Connectors](#)

MapReduce

The MapReduce engine provides a platform for the massively parallel execution of algorithms written in Java. MapReduce uses a parallel programming model for processing data on a distributed system. It can process vast amounts of data quickly and can scale linearly. It is particularly effective as a mechanism for batch processing of unstructured and semi-structured data. MapReduce abstracts lower level operations into computations over a set of keys and values.

Although big data is often described as unstructured, incoming data always has some structure. However, it does not have a fixed, predefined structure when written to HDFS. Instead, MapReduce creates the desired structure as it reads the data for a particular job. The same data can have many different structures imposed by different MapReduce jobs.

A simplified description of a MapReduce job is the successive alternation of two phases, the Map phase and the Reduce phase. Each Map phase applies a transform function over each record in the input data to produce a set of records expressed as key-value pairs. The output from the Map phase is input to the Reduce phase. In the Reduce phase the Map output records are sorted into key-value sets so that all records in a set have the same key value. A reducer function is applied to all the records in a set and a set of output records are produced as key-value pairs. The Map phase is logically run in parallel over each record while the Reduce phase is run in parallel over all key values.

Oracle R Support for Big Data

R is an open source language and environment for statistical analysis and graphing, providing linear and nonlinear modeling, standard statistical methods, time-series

analysis, classification, clustering, and graphical data displays. Thousands of open-source packages are available in the Comprehensive R Archive Network (CRAN) for a spectrum of applications, such as bioinformatics, spatial statistics, and financial and marketing analysis. The popularity of R has increased as its functionality matured to rival that of costly proprietary statistical packages

Analysts typically use R on a PC, which limits the amount of data and the processing power available for analysis. Oracle eliminates this restriction by extending the R platform to directly leverage Oracle Database and Oracle Big Data Appliance for a fully scalable solution. Analysts continue to work on their PCs using the familiar R user interface while manipulating huge amounts of data stored in an Oracle database or in HDFS using massively parallel processing.

For example, an analyst might have a hypothesis about shopping behavior, which can be tested using web logs. The analyst writes an R program on his or her PC and runs it on a sample of data stored in Oracle Big Data Appliance. If the sample results are promising, then the analyst can run the program on the entire web log and, optionally, store the results in Oracle Database for further analysis.

While testing and developing the R program, the analyst works interactively and gets the results in real time. If the particular type of analysis warrants it, the program can be scheduled to run overnight in batch mode as part of the routine maintenance of the database.

The standard R distribution is installed on all nodes of Oracle Big Data Appliance, enabling R programs to run as MapReduce jobs on vast amounts of data. Users can transfer existing R scripts and packages from their PCs to use on Oracle Big Data Appliance.

Oracle R Connector for Hadoop provides R users high performance, native access to HDFS and the MapReduce programming framework. The R Connector is included in the Oracle Big Data Connectors. See "[Oracle R Connector for Hadoop](#)" on page 1-9.

These components make Oracle Database and the Hadoop computational infrastructure available to statistical users without requiring them to learn the native programming languages of either one.

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

Oracle Big Data Connectors

Oracle Big Data Connectors facilitate data access between data stored in CDH and Oracle Database. They are licensed separately from Oracle Big Data Appliance.

These are the connectors

- [Oracle Direct Connector for Hadoop Distributed File System](#)
- [Oracle Loader for Hadoop](#)
- [Oracle Data Integrator Application Adapter for Hadoop](#)
- [Oracle R Connector for Hadoop](#)

See Also: *Oracle Big Data Connectors User's Guide*

Oracle Direct Connector for Hadoop Distributed File System

Oracle Direct Connector for Hadoop Distributed File System (Oracle Direct Connector) provides read access to HDFS from an Oracle database 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. External tables are often used to stage data to be transformed during a database load.

These are a few ways that you can use Oracle Direct Connector:

- Access any data stored in HDFS files
- Access CSV files generated by Oracle Loader for Hadoop
- Load data extracted and transformed by Oracle Data Integrator

Oracle Loader for Hadoop

Oracle Loader for Hadoop is an efficient and high performance loader for fast movement of data from CDH into a table in Oracle Database. Oracle Loader for Hadoop partitions the data and transforms it into an Oracle-ready format on CDH. It optionally sorts records by primary key before loading the data or creating output files.

You can use Oracle Loader for Hadoop as either a Java program or a command-line utility. The load runs as a MapReduce job on the CDH cluster.

Oracle Loader for Hadoop also reads from and writes to Oracle Data Pump files.

Oracle Data Integrator Application Adapter for Hadoop

Oracle Data Integrator (ODI) extracts, transforms, and loads data into Oracle Database from a wide range of sources.

In Oracle Data Integrator, a knowledge module (KM) is a code template dedicated to a specific task in the data integration process. You use ODI Studio to load, select, and configure the KMs for your particular application. More than 150 KMs are available to help you acquire data from a wide range of third-party databases and other data repositories. You only need to load a few KMs for any particular job.

Oracle Data Integrator Application Adapter for Hadoop contains the KMs specifically for use with big data. [Table 1–1](#) describes these knowledge modules. They stage the data in Hive, a data warehouse built on Hadoop, for the best performance.

Table 1–1 Oracle Data Integrator Application Adapter for Hadoop Knowledge Modules

KM	Description
IKM File to Hive (Load Data)	Loads data from local and HDFS files into Hive tables. It provides options for better performance through Hive partitioning and fewer data movements.
IKM Hive Control Append	Integrates data into a Hive target table in truncate/ insert (append) mode. Data can be controlled (validated). Invalid data is isolated in the error Table and can be recycled.
IKM Hive Transform	Integrates data into a Hive target table after the data has been transformed by a customized script such as Perl or Python.
IKM File-Hive to Oracle (OLH)	Integrates data from an HDFS file or Hive source into an Oracle Database target using Oracle Loader for Hadoop.
CKM Hive	Validates data against constraints.
RKM Hive	Reverse engineers Hive tables.

See Also: *Oracle Fusion Middleware Application Adapters Guide for Oracle Data Integrator Version 1.1.1.6 or later*

Oracle R Connector for Hadoop

Oracle R Connector for Hadoop is an R package that provides an interface between the local R environment, Oracle Database, and Hadoop on Oracle Big Data Appliance. Using simple R functions, you can sample data in HDFS, copy data between Oracle Database and HDFS, and schedule R programs to execute as MapReduce jobs. You can return the results to Oracle Database or your laptop.

Analyzing and Visualizing Big Data

After big data is transformed and loaded in Oracle Database, you can use the full panoply of Oracle business intelligence solutions and decision support products to further analyze and visualize all your data.

See Also:

- Oracle Business Intelligence Solutions website at
<http://www.oracle.com/us/solutions/ent-performance-bi/business-intelligence/index.html>
- Data Warehousing and Business Intelligence in the Oracle Database Documentation Library at
http://www.oracle.com/pls/db112/portal.portal_db?selected=6&frame=

Administering Oracle Big Data Appliance

This chapter provides information about the software and services installed on Oracle Big Data Appliance. It contains these topics:

- [Managing CDH Operations](#)
- [Providing Remote Client Access to CDH](#)
- [Managing User Accounts](#)
- [Software Layout](#)
- [Software Services](#)
- [Effects of Hardware on Software Availability](#)
- [Security on Oracle Big Data Appliance](#)

Managing CDH Operations

Cloudera Manager is installed on Oracle Big Data Appliance to help you with Cloudera's Distribution including Apache Hadoop (CDH) operations. Cloudera Manager provides a single administrative interface to all Oracle Big Data Appliance servers configured as part of the Hadoop cluster.

Cloudera Manager simplifies the performance of these 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)

Cloudera Manager runs on node02 and is available on port 7180.

To use Cloudera Manager:

1. Open a browser and enter a URL like the following:

```
http://bda1node02.example.com:7180
```

In this example, `bda1` is the name of the appliance, `node02` is the name of the server, `example.com` is the domain, and `7180` is the default port number for Cloudera Manager.

- Log in with a user name and password for Cloudera Manager. Only a user with administrative privileges can change the settings. Other Cloudera Manager users can view the status of Oracle Big Data Appliance.

See Also: *Cloudera Manager User Guide* at <http://oracle.cloudera.com/> or click **Help** on the Cloudera Manager Help menu.

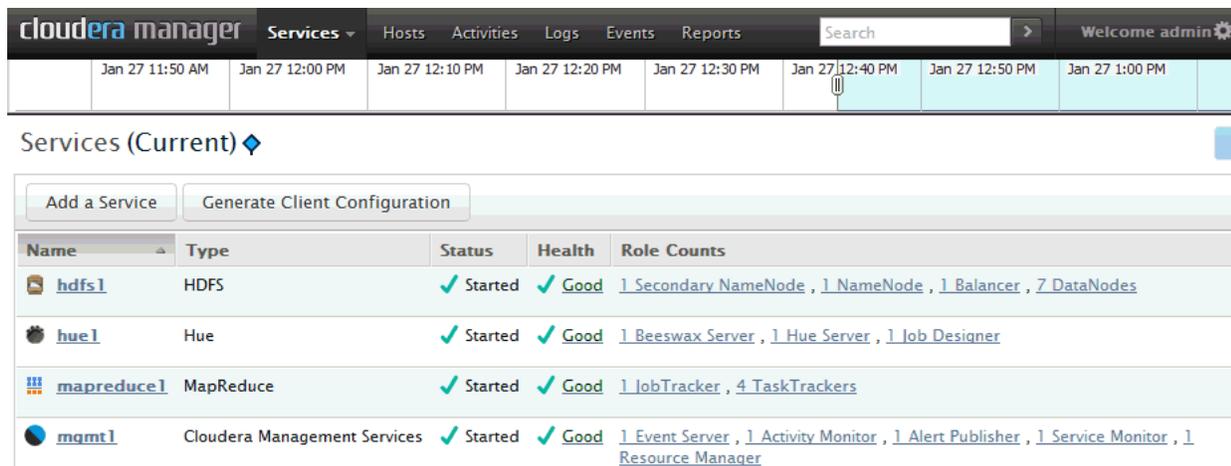
Monitoring the Status of Oracle Big Data Appliance

In Cloudera Manager, you can choose the page from the Navigation Bar across the top of the display:

- Services:** Monitors the status and health of services running on Oracle Big Data Appliance. Click the name of a service to drill down to additional information.
- Hosts:** Monitors the health, disk usage, load, physical memory, swap space, and so forth of all servers.
- Activities:** Monitors all MapReduce jobs running in the selected time period.
- Logs:** 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.
- Events:** Records a change in state and other noteworthy occurrences. You can search for one or more keywords for a selected server, service, and time period. You can also select the event types: Audit Event, Activity Event, Health Check, or Log Message.
- Reports:** Generates reports on demand for disk and MapReduce use.

Figure 2-1 shows the opening display of Cloudera Manager, which is the Services page.

Figure 2-1 Cloudera Manager Services Page



Performing Administrative Tasks

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.

To access Cloudera Manager Administration:

1. Log in to Cloudera Manager with administrative privileges.
2. Click **Welcome admin** at the top right of the page.

Collecting Diagnostic Information

If you need help from Oracle Support to troubleshoot CDH issues, then you should first collect diagnostic information using Cloudera Manager.

To collect diagnostic information about CDH:

1. Log in to Cloudera Manager with administrative privileges.
2. From the Help menu, click **Send Diagnostic Data**.
3. Verify that **Send Diagnostic Data to Cloudera Automatically** is not selected. Keep the other default settings.
4. Click **Collect Host Statistics Globally**.
5. Wait while all statistics are collected on all nodes.
6. Click **Download Result Data** and save the ZIP file with the default name. It identifies your CDH license.
7. Go to My Oracle Support at <http://support.oracle.com>.
8. Open a Service Request (SR) if you have not already done so.
9. Upload the ZIP file into the SR. If the file is too large, then upload it to <ftp.oracle.com>, as described in the next procedure.

To upload the diagnostics to ftp.oracle.com:

1. Open an FTP client and connect to <ftp.oracle.com>.
You can use an FTP client such as WinSCP4 to upload the ZIP file. See [Example 2-1](#) if you are using a command-line FTP client.
2. Log in as user `anonymous` and leave the password blank.
3. In the `bda/incoming` directory, create a directory using the SR number for the name, in the format `SRnumber`. The resulting directory structure looks like this:

```
bda
  incoming
    SRnumber
```

4. Set the binary option to prevent corruption of binary data.
5. Upload the diagnostics ZIP file to the `bin` directory.
6. Update the SR with the full path and file name.

[Example 2-1](#) shows the commands to upload the diagnostics using the Windows FTP command interface.

Example 2-1 Uploading Diagnostics Using Windows FTP

```
ftp> open ftp.oracle.com
Connected to bigip-ftp.oracle.com.
220-*****
220-Oracle FTP Server
.
```

```
220-*****
220
220 User (bigip-ftp.oracle.com:(none)): anonymous
231 Please specify the password.
231 Password:
230 Login successful.
ftp> cd bda/incoming
250 Directory successfully changed.
ftp> mkdir SR12345
257 "/bda/incoming/SR12345" created
ftp> cd SR12345
250 Directory successfully changed.
ftp> bin
200 Switching to Binary mode.
ftp> put D:\Downloads\3609df...c1.default.20122505-15-27.host-statistics.zip
200 PORT command successful. Consider using PASV.
150 Ok to send data.
226 File receive OK.
ftp: 706755 bytes sent in 1.97Seconds 358.58Kbytes/sec.
```

Using Hadoop Monitoring Utilities

Users can monitor MapReduce jobs without providing a Cloudera Manager user name and password.

Monitoring the JobTracker

Hadoop Map/Reduce Administration monitors the JobTracker, which runs on port 50030 of node03 on Oracle Big Data Appliance.

To monitor the JobTracker:

- Open a browser and enter a URL like the following:

```
http://bda1node03.example.com:50030
```

In this example, bda1 is the name of the appliance, node03 is the name of the server, and 50030 is the default port number for Hadoop Map/Reduce Administration.

Figure 2–2 shows part of a Hadoop Map/Reduce Administration display.

Figure 2–2 Hadoop Map/Reduce Administration

bda1node03 Hadoop Map/Reduce Administration

State: RUNNING
 Started: Mon Jan 23 18:26:30 PST 2012
 Version: 0.20.2-cdh3u2, 95a824e4005b2a94fe1c11f1ef9db4c672ba43cb
 Compiled: Thu Oct 13 21:51:41 PDT 2011 by root from Unknown
 Identifier: 201201231826

Cluster Summary (Heap Size is 517.12 MB/910.25 MB)

Running Map Tasks	Running Reduce Tasks	Total Submissions	Nodes	Occupied Map Slots	Occupied Reduce Slots	Reserved Map Slots	Reserved Reduce Slots	Map Task Capacity	Reduce Task Capacity	Avg. Tasks/Node
0	0	10	14	0	0	0	0	168	168	24.00

Scheduling Information

Queue Name	State	Scheduling Information
default	running	N/A

Filter (Jobid, Priority, User, Name)
 Example: 'users:smith 3200' will filter by 'smith' only in the user field and '3200' in all fields

Running Jobs

Completed Jobs

Jobid	Priority	User	Name	Map % Complete	Map Total	Maps Completed	Reduce % Complete	Reduce Total
job_201201231826_0006	NORMAL	oracle	DataGenCSV	100.00%	128	128	100.00%	0
job_201201231826_0007	NORMAL	oracle	OraLoader_fivdti_i7_hash128_dtext_dl_5000000000_128_tasks128	100.00%	384	384	100.00%	128
job_201201231826_0008	NORMAL	oracle	OraLoader_fivdti_i7_hash128_dtext_dl_5000000000_128_tasks128	100.00%	3837	3837	100.00%	128
job_201201231826_0009	NORMAL	oracle	OraLoader_fivdti_i7_hash128_dtext_dl_5000000000_128_tasks128	100.00%	3837	3837	100.00%	128
job_201201231826_0010	NORMAL	oracle	OraLoader_fivdti_i7_hash128_dtext_dl_5000000000_128_tasks128	100.00%	3837	3837	100.00%	128

Monitoring the TaskTracker

The Task Tracker Status interface is available on port 50060 of node04 -node18 on Oracle Big Data Appliance.

To monitor the TaskTracker:

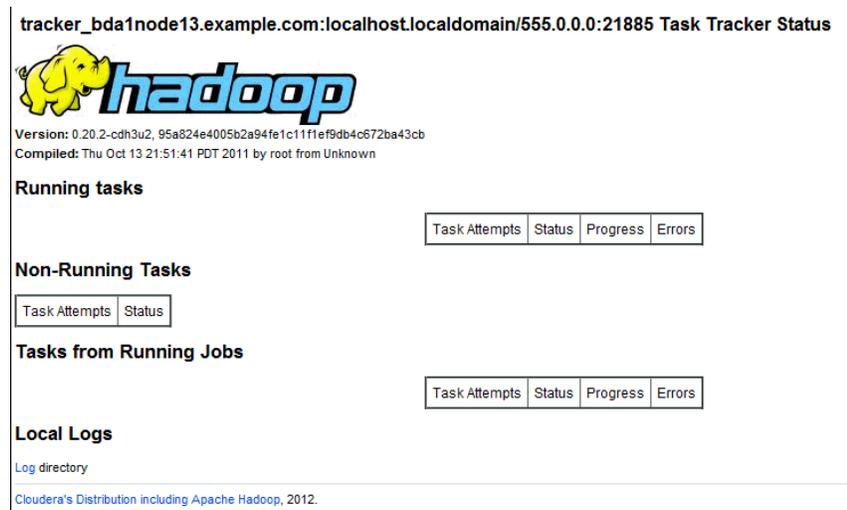
- Open a browser and enter a URL like the following:

`http://bda1node13.example.com:50060`

In this example, bda1 is the name of the rack, node13 is the name of the server, and 50060 is the default port number for Task Tracker Status.

Figure 2–3 shows the TaskTracker.

Figure 2–3 Task Tracker Status



Providing Remote Client Access to CDH

Oracle Big Data Appliance supports full local access to all commands and utilities in Cloudera's Distribution including Apache Hadoop (CDH).

You can use a browser on any computer on the same network as Oracle Big Data Appliance to access Cloudera Manager, Hadoop Map/Reduce Administration, Hadoop Task Tracker UI, and other browser-based Hadoop tools.

To issue Hadoop commands remotely, however, you must connect from a system configured as a CDH client. This chapter explains how to set up a computer so you can access HDFS and submit MapReduce jobs on Oracle Big Data Appliance.

To follow these procedures, you must have these access privileges:

- Root access to the client system.
- Read access to the `/opt/hadoop/client` directory on an Oracle Big Data Appliance server.
- Login access to Cloudera Manager.

If you do not have these access privileges, then contact your system administrator for help.

Installing CDH on the Client System

The system that you use to access Oracle Big Data Appliance must run Oracle Linux 5 or a compatible Linux distribution, that is, one that permits installation of Oracle Linux 5 RPMs. You must install the same version of CDH that Oracle Big Data Appliance runs, or CDH3u4 or later.

To install the CDH client software:

1. Log in to the Linux system as `root` and change to the `/tmp` directory.

```
cd /tmp
```

2. Perform a secure copy of the Hadoop client RPM to the `/tmp` directory:

```
scp username@bda_node_name:/opt/hadoop/client/*.rpm .
```

Or, to use `sftp` instead of `scp`:

- a. Open a secure connection to any server in Oracle Big Data Appliance:

```
sftp username@bda_node_name
```

- b. Copy the RPM file:

```
get /opt/hadoop/client/*.rpm
```

- c. Close the SFTP connection:

```
quit
```

3. Ensure that no Hadoop client currently exists on your system:

```
rpm -qa | grep hadoop
```

If you see just the prompt, then no Hadoop client is installed, and you can continue with the next step.

If the command returns a value, then remove that version:

```
rpm -e hadoop-version
```

4. Install the new CDH client:

```
rpm -ihv hadoop-version
```

[Example 2-2](#) illustrates the previous steps. It uses `scp` to copy `hadoop-0.20-0.20.2+923.202-1.noarch.rpm` from `bda1node09`, removes an older version of Hadoop, and installs the new version.

Example 2-2 Installing the CDH Client Software

```
[root@client]$ cd /tmp
[root@client]$ scp username@bda1node09.example.com:/opt/hadoop/client/*.rpm .
username@bda1node09.example.com's password:
hadoop-0.20-0.20.2+923.202-1.noarch.rpm 100% 30MB 10.0MB/s 00:03
[root@client]$ rpm -qa | grep hadoop
hadoop-0.20-0.20.2+923.194-1
[root@client]$ rpm -e hadoop-0.20-0.20.2+923.194-1
[root@client]$ rpm -ihv hadoop-0.20-0.20.2+923.202-1.noarch.rpm
warning: hadoop-0.20-0.20.2+923.202-1.noarch.rpm: Header V4 DSA signature: NOKEY,
key ID e8f86acd
Preparing...          ##### [100%]
 1:hadoop-0.20       ##### [100%]
```

Configuring CDH

After installing CDH, you must configure it for use with Oracle Big Data Appliance.

To configure the Hadoop client:

1. Open a browser on your client system and connect to Cloudera Manager. It runs on `node02` and listens on port 7180, as shown in this example:

```
http://bda1node02.example.com:7180
```

2. Log in as `admin`.
3. Cloudera Manager opens on the Services tab. Click the **Generate Client Configuration** button.

4. On the Command Details page (shown in [Figure 2-4](#)), click **Download Result Data** to download `global-clientconfig.zip`.
5. Unzip `global-clientconfig.zip` into the `/tmp` directory on the client system. It creates a `hadoop-conf` directory containing these files:

```
core-site.xml
hadoop-env.sh
hdfs-site.xml
log4j.properties
mapred-site.xml
README.txt
ssl-client.xml.example
```

6. Open `hadoop-env.sh` in a text editor and change `JAVA_HOME` to the correct location on your system:

```
export JAVA_HOME=full_directory_path
```

7. Delete the hash mark (`#`) to uncomment the line, then save the file.
8. Copy the configuration files to the Hadoop conf directory:

```
cd /tmp/hadoop-conf
cp * /usr/lib/hadoop/conf/
```

9. Validate the installation by changing to the `mapred` user and submitting a MapReduce job, such as the one shown here:

```
su mapred
hadoop jar /usr/lib/hadoop/hadoop-examples.jar pi 10 1000000
```

[Figure 2-4](#) shows the download page for the client configuration.

Figure 2-4 Cloudera Manager Command Details: GenerateClient Page

The screenshot shows the Cloudera Manager interface. At the top, there's a navigation bar with 'Services', 'Hosts', 'Activities', 'Logs', 'Events', and 'Reports'. A search box and 'Welcome admin' are also visible. The main content area is titled 'Command Details: GenerateClient'. It displays the following information:

- Command:** GenerateClient
- Started at:** Mon, 26 Mar 2012 06:12 PM
- Progress:** Finished (indicated by a progress bar and a 'Finished' label)
- Completed at:** Mon, 26 Mar 2012 06:12 PM
- Result:** Successfully created global client configuration

Below this information is a blue button labeled 'Download Result Data'. Underneath is a section titled 'Child Commands' which contains a table:

Command	Context	Started at	Status
GenerateClient	Service mapreduce1	Mon, 26 Mar 2012 06:12 PM	Finished at Mon, 26 Mar 2012 06:12 PM

Below the table, it says 'Successfully created client configuration' and 'Showing items 1--1. [Next](#)'.

Managing User Accounts

Every open source package installed on Oracle Big Data Appliance creates one or more users and groups. Most of these users do not have login privileges, shells, or home directories. They are used by daemons and are not intended as an interface for

individual users. For example, Hadoop operates as the `hdfs` user, MapReduce operates as `mapred`, and Hive operates as `hive`. [Table 2–1](#) identifies the operating system users and groups that are created automatically during installation of Oracle Big Data Appliance Software for use by CDH components and other software packages.

You can use the `oracle` identity to run Hadoop and Hive jobs immediately after the Oracle Big Data Appliance software is installed. This user account has login privileges, a shell, and a home directory. Oracle NoSQL Database and Oracle Data Integrator run as the `oracle` user. Its primary group is `oinstall`.

Note: Do not delete or modify the users created during installation, because they are required for the software to operate.

When creating additional user accounts, define them as follows:

- To run MapReduce jobs, users must be in the `hadoop` group.
- To create and modify tables in Hive, users must be in the `hive` group.

Table 2–1 *Operating System Users and Groups*

User Name	Group	Used By	Login Rights
flume	flume	Flume parent and nodes	No
hbase	hbase	HBase processes	No
hdfs	hadoop	NameNode, DataNode	No
hive	hive	Hive metastore and server processes	No
hue	hue	Hue processes	No
mapred	hadoop	JobTracker, TaskTracker, Hive Thrift daemon	Yes
mysql	mysql	MySQL Server	Yes
oozie	oozie	Oozie server	No
oracle	dba, oinstall	Oracle NoSQL Database, Oracle DBA, Oracle Loader for Hadoop, Oracle Data Integrator	Yes
puppet	puppet	Puppet parent (puppet nodes run as root)	No
sqoop	sqoop	Sqoop metastore	No
svctag	--	Auto Service Request	No
zookeeper	zookeeper	Zookeeper processes	No

Software Layout

The following sections identify the software installed on Oracle Big Data Appliance and where it runs in the rack. Some components operate with Oracle Database 11.2.0.2 and later releases.

Software Components

These software components are installed on all 18 servers in Oracle Big Data Appliance Rack. Oracle Linux, required drivers, firmware, and hardware verification utilities are factory installed. All other software is installed on site using the Mammoth Utility.

Note: You do not need to install software on Oracle Big Data Appliance. Doing so may result in a loss of warranty and support. See the *Oracle Big Data Appliance Owner's Guide*.

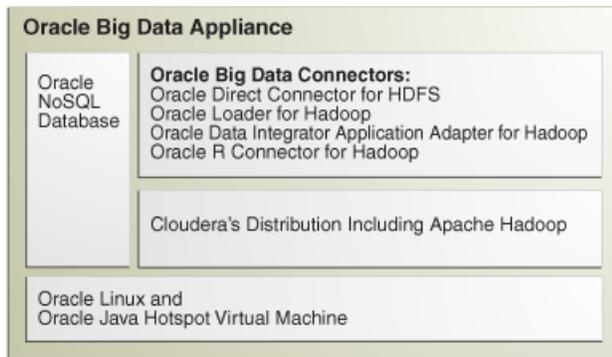
Installed software:

- Oracle Linux 5.6
- Java HotSpot Virtual Machine 6 Update 29
- Cloudera's Distribution including Apache Hadoop Release 3 Update 3 (CDH)
- Cloudera Manager 3.7
- Oracle Loader for Hadoop 1.1
- Oracle NoSQL Database Community Edition 11g Release 1.2.123
- Oracle Data Integrator Agent 1.1.1.6
- Oracle R Connector for Hadoop 1.0
- R distribution 2.13.2
- Oracle Direct Connector for Hadoop Distributed File System 1.0
- Oracle Instant Client 11.2.0.3
- MySQL Database SE 5.5.18

See Also: *Oracle Big Data Appliance Owner's Guide* for information about the Mammoth Utility.

Figure 2–5 shows the relationships among the major components.

Figure 2–5 Major Software Components of Oracle Big Data Appliance



Logical Disk Layout

Each server has 12 disks. The critical information is stored on disks 1 and 2.

Table 2–2 describes how the disks are partitioned.

Table 2–2 Logical Disk Layout

Disk	Description
1 to 2	150 GB mirrored, physical and logical partition with the Linux operating system, all installed software, NameNode data, and MySQL data, for a total of four copies 2.8 TB HDFS data partition
3	Single Oracle NoSQL Database partition, if activated during software installation; otherwise, a single HDFS data partition
4 to 12	Single HDFS data partition

Software Services

This section identifies the services, where they run, and their default status. Services that are always on are required for normal operation. Services that you can switch on and off are optional.

You can use Cloudera Manager view the services.

To view the services:

1. In Cloudera Manager, click the **Services** tab at the top of the page to display the Services page.
2. Click the name of a service to see its detail pages. The service opens on the Status page.
3. Click the link to the page you want to view: Status, Instances, Commands, Configuration, or Audits.

Parent Services

Table 2–3 describes the parent services and those that run without child services.

Table 2–3 Parent Services

Service	Role	Description	Default Status
hbase	--	HBase database	OFF
hdfs1	NameNode	Tracks all files stored in the cluster.	Always ON
hdfs1	Secondary NameNode	Tracks information for the NameNode	Always ON
hdfs1	Balancer	Periodically issues the balancer command; although the balancer service is enabled, it does not run all the time	Always ON
hive	--	Hive data warehouse for Hadoop	Always ON
hue1	Hue Server	GUI for HDFS, MapReduce, and Hive, with shells for Pig, Flume, and HBase	Always ON
mapreduce1	JobTracker	Used by MapReduce	Always ON
mgmt1	all	Cloudera Manager	Always ON
MySQL	--	MySQL Master Database	ON

Table 2–3 (Cont.) Parent Services

Service	Role	Description	Default Status
ODI Agent	--	Oracle Data Integrator agent, installed on same node as MySQL Database	ON
oozie	--	Workflow and coordination service for Hadoop	OFF
ZooKeeper	--	ZooKeeper coordination service	OFF

Child Services

Table 2–4 describes the child services.

Table 2–4 Child Services

Service	Role	Description	Default Status
HBase Region Server	--	Hosts data and processes requests for HBase	OFF
hdfs1	DataNode	Stores data in HDFS	Always ON
mapreduce1	TaskTracker	Accepts tasks from the JobTracker	Always ON
NoSQL DB Storage Node	--	Supports Oracle NoSQL Database	ON
nosqldb	--	Supports a web console or command-line interface for administering Oracle NoSQL Database	ON

Software Services Distribution

All services are installed on all servers, but individual services run only on designated servers.

Service Locations

Table 2–5 identifies the nodes where the services run. Services cannot be run on different nodes in this release, so do not attempt to change this configuration.

Table 2–5 Software Service Locations

Service	Node
Balancer	Node01
Beeswax Server	Node03
Cloudera Manager Agents	All nodes
Cloudera Manager SCM Server	Node02
Datanode	All nodes
Hive Server	Node03
Hue Server	Node03
JobTracker	Node03
MySQL Backup	Node02
MySQL Primary Server	Node03
NameNode	Node01
Oracle Data Integrator Agent ¹	Node03

Table 2–5 (Cont.) Software Service Locations

Service	Node
Oracle NoSQL Database Administration ¹	Node02
Oracle NoSQL Database Server Processes ¹	All nodes
Puppet Agents	All nodes
Puppet Master	Node01
Secondary NameNode	Node02
TaskTracker	Node04 to Node18

¹ Started only if requested in the *Oracle Big Data Appliance Configuration Worksheets*

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. This vulnerability is intrinsic to Apache Hadoop (v0.20.2 and earlier).

Oracle protects against catastrophic failure by maintaining four copies of the NameNode logs:

- Node01: Working copy of the NameNode snapshot and update logs is stored in `/opt/hadoop/dfs/` and is automatically mirrored in a local Linux partition.
- Node02: Backup copy of the logs is stored in `/opt/sharedir/` and is also automatically mirrored in a local Linux partition.

A fifth backup outside of Oracle Big Data Appliance can be configured during the software installation.

Note: The Secondary NameNode is not a backup of the primary NameNode and does not provide failover. The Secondary NameNode performs memory-intensive functions for the primary NameNode.

Unconfigured Software

The following tools are installed but not configured. Before using them, you must configure them for your use.

- Flume
- Mahout
- Oozie
- Sqoop
- Whirr

See Also: *CDH3 Installation and Configuration Guide* for configuration procedures at

<http://oracle.cloudera.com>

Effects of Hardware on Software Availability

The effects of a server failure vary depending on the server's function within the CDH cluster. Sun Fire servers are more robust than commodity hardware, so you should

experience fewer hardware failures. This section highlights the most important services that run on the various servers. For a full list, see "[Service Locations](#)" on page 2-12.

Node01: Critical for All Services

Node01 is critically important because it is where the NameNode runs. If this server fails, the effect is downtime for the entire cluster, because the NameNode keeps track of the data locations. However, there are always four copies of the NameNode metadata on Oracle Big Data Appliance, plus an optional NFS backup.

The current state and update logs are written to these locations:

- **Node01:** `/opt/hadoop/dfs/` on Disk 1 is the working copy with a local, operating system, mirrored partition on Disk 2 providing a second copy.
- **Node04:** `/opt/shareddir/` on Disk 1 is the third copy, which is also duplicated on a mirrored partition on Disk 2.

Node02 to Node03: Required for Some Services

The cluster continues to function after a loss of node2 or node03, but with a loss of some services that might be critical to your operation. The disruptions are in these areas:

Node02:

- **Cloudera Manager:** This tool provides central management for the entire CDH cluster. Without this tool, you can still monitor activities using the utilities described in "[Using Hadoop Monitoring Utilities](#)" on page 2-4.
- **Oracle NoSQL Database:** This database is an optional component of Oracle Big Data Appliance, so the extent of the disruption depends on whether you are using it and how critical it is to your applications.

Node03:

- **Oracle Data Integrator:** This service supports Oracle Data Integrator Application Adapter for Hadoop. You cannot use this connector when node03 is down.
- **MySQL Master Database:** Cloudera Manager, Oracle Data Integrator, Hive, and Oozie use MySQL Database. The data is replicated automatically, but you cannot access it when the master database server, which runs on node03, is down.
- **JobTracker:** Assigns MapReduce tasks to specific nodes in the CDH cluster.

Node04 to Node18: Optional for All Services

Node04 to node18 are optional in that Oracle Big Data Appliance continues to operate with no loss of service if a failure occurs. The NameNode automatically replicates the lost data to maintain three copies at all times. MapReduce jobs execute on copies of the data stored elsewhere in the cluster. The only loss is in computational power, because there are fewer servers on which to distribute the work.

Node04 stores two duplicate copies of the critical NameNode data, but a loss of this backup does not affect operation of the NameNode.

Security on Oracle Big Data Appliance

This section identifies security vulnerabilities and discusses the precautions you can take to prevent unauthorized use of the software and data on Oracle Big Data Appliance. It consists of these subsections:

- [CDH Security](#)
- [Port Numbers Used on Oracle Big Data Appliance](#)
- [Security of Software Components](#)

CDH Security

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.

Cloudera's Distribution including Apache Hadoop (CDH) supports Kerberos network authentication protocol to prevent malicious impersonation. You must install and configure Kerberos and set up a Kerberos Key Distribution Center and realm. Then you configure various components of CDH to use Kerberos.

CDH provides these securities when configured to use Kerberos:

- The CDH master nodes, NameNode, and JobTracker resolve the group name so that users cannot manipulate their 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.

See Also: <http://oracle.cloudera.com> for these manuals:

- *CDH3 Security Guide*
- *Configuring Hadoop Security with Cloudera Manager*
- *Configuring TLS Security for Cloudera Manager*

Port Numbers Used on Oracle Big Data Appliance

[Table 2–6](#) identifies the port numbers that may be used in addition to those used by CDH. For the full list of CDH port numbers, go to the Cloudera website at

<http://ccp.cloudera.com/display/CDHDOC/Configuring+Ports+for+CDH3>

To view the ports used on a particular server:

1. In Cloudera Manager, click the **Hosts** tab at the top of the page to display the Hosts page.
2. In the Name column, click a server link to see its detail page.
3. Scroll down to the Ports section.

See Also: The Cloudera website for CDH port numbers:

- *Hadoop Default Ports Quick Reference*
<http://www.cloudera.com/blog/2009/08/hadoop-default-ports-quick-reference/>
- *Configuring Ports for CDH3* at
<https://ccp.cloudera.com/display/CDHDOC/Configuring+Ports+for+CDH3>

Table 2–6 Oracle Big Data Appliance Port Numbers

Service	Port
Automated Service Monitor (ASM)	30920
MySQL Database	3306
Oracle Data Integrator Agent	20910
Oracle NoSQL Database administration	5001
Oracle NoSQL Database processes	5010 to 5020
Oracle NoSQL Database registration	5000
Port map	111
Puppet master service	8140
Puppet node service	8139
rpc.statd	668
ssh	22
xinetd (service tag)	6481

Security of Software Components

Following are configuration details about the software components and any special security precautions they require.

Puppet

The puppet node service (`puppetd`) runs continuously as root on all servers. It listens on port 8139 for "kick" requests, which trigger it to request updates from the puppet master. It does not receive updates on this port.

The puppet master service (`puppetmasterd`) runs continuously as the puppet user on the first server of the primary Oracle Big Data Appliance rack. It listens on port 8140 for requests to push updates to puppet nodes.

The puppet nodes generate and send certificates to the puppet master to register initially during installation of the software. For updates to the software, the puppet master signals ("kicks") the puppet nodes, which then request all configuration changes from the puppet master node that they are registered with.

The puppet master sends updates only to puppet nodes that have known, valid certificates. Puppet nodes only accept updates from the puppet master host name they initially registered with. Because Oracle Big Data Appliance uses an internal network for communication within the rack, the puppet master host name resolves using `/etc/hosts` to an internal, private IP address.

Glossary

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.

Cloudera's Distribution including Apache Hadoop (CDH)

See [CDH](#).

CDH

The version of Apache Hadoop and related components installed on Oracle Big Data Appliance.

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 Oracle Big Data Appliance Rack form a cluster. Servers 1, 2, and 3 are master nodes. Servers 4 to 18 are worker nodes.

See [Hadoop](#).

DataNode

A server in a CDH cluster that stores data in HDFS. A DataNode performs file system operations assigned by the NameNode.

See also [HDFS](#); [NameNode](#).

Flume

A distributed service in CDH for collecting and aggregating data from almost any source into a data store like HDFS or HBase.

See also [HBase](#); [HDFS](#).

JobTracker

A service that assigns MapReduce tasks to specific nodes in the CDH cluster, preferably those nodes storing the data.

See also [Hadoop](#); [MapReduce](#).

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).

Hadoop Distributed File System (HDFS)

See [HDFS](#).

Hadoop User Experience (HUE)

See [HUE](#).

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.

See also [cluster](#).

HDFS

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.

See also [cluster](#).

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.

See also [Hive Thrift](#); [HiveQL](#); [MapReduce](#)

Hive Thrift

An RPC interface for remote access to CDH for Hive queries.

See also [Hive](#).

HiveQL

See also [Hive](#).

HotSpot

HotSpot is a Java Virtual Machine (JVM) that is maintained and distributed by Oracle. It automatically optimizes code that is executed frequently, leading to high performance. HotSpot is the standard JVM for the other components of the Oracle Big Data Appliance stack.

HUE

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.

See also [HDFS](#); [Hive](#).

Java HotSpot Virtual Machine

See [HotSpot](#).

MapReduce

A method of distributing work across a cluster used by the MapReduce engine.

A programming model that enables the MapReduce engine to distribute the work across the cluster. MapReduce programs can run massively in parallel in CDH.

A MapReduce program contains these tasks:

- Mappers: Process the records of the data set.
- Reducers: Merge the output from several mappers.
- Combiners: Concatenate the results from all reducers into a result set.

MySQL Database

A SQL-based relational database management system. On Oracle Big Data Appliance, Cloudera Manager, Oracle Data Integrator, Hive, and Oozie use MySQL Database as a metadata repository.

NameNode

A service that maintains a directory of all files in HDFS and tracks where data is stored in the CDH cluster.

See also [HDFS](#).

node

A server in a CDH cluster.

See [cluster](#).

Oozie

An open-source workflow and coordination service for managing data processing jobs in CDH.

Oracle Database Instant Client

A small-footprint client that enables Oracle applications to run without a standard Oracle client.

Oracle Linux

An open-source operating system. Oracle Linux 5.6 is the same version used by Exalogic 1.1. It features the Oracle Unbreakable Enterprise Kernel.

Oracle Wallet Manager

An application for managing the security credentials stored in Oracle wallets. A wallet is a password-protected container that stores authentication and signing credentials.

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.

See also [MapReduce](#).

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 three components: puppet agents, typically just called puppets; the puppet master server; a console; and a cloud provisioner.

See also [puppet agent](#); [puppet master](#).

puppet agent

Primarily pull configurations from the puppet master and apply them. Puppet agents run on every server in Oracle Big Data Appliance.

puppet master

Primarily serve configurations to the puppet agents.

See also [Puppet](#); [puppet agent](#).

R

An open-source language and environment for statistical analysis and graphing.

Oracle Auto Service Request for Sun Systems

A software tool that monitors the health of the hardware and automatically generates a service request if it detects a problem. This tool is a feature of an Oracle Warranty.

See also [Oracle Automated Service Monitor \(OASM\)](#).

Oracle Automated Service Monitor (OASM)

A service for monitoring the health of Oracle Sun hardware systems. Formerly named Sun Automatic Service Monitor (SASM).

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 Connector for Hadoop uses the Sqoop executable to move data between HDFS and Oracle Database.

table

In Hive, all files in a directory stored in HDFS.

See also [HDFS](#).

ZooKeeper

A centralized coordination service for CDH distributed processes that maintains configuration information and naming, and provides distributed synchronization and group services.

Index

A

application adapters, 1-8
ASM, 2-16
Automated Service Monitor, 2-16

B

Balancer, 2-11, 2-12
Beeswax server, 2-12
Berkeley DB, 1-5
big data description, 1-1
business intelligence, 1-3

C

chunking files, 1-5
Cloudera Manager, 2-10, 2-11, 2-14
 accessing administrative tools, 2-2
 connecting to, 2-1
 description, 2-1
 UI overview, 2-2
Cloudera Manager agents, 2-12
Cloudera Manager SCM server, 2-12
Cloudera's Distribution including Apache
 Hadoop, 1-5, 2-10
Cloudera's Distribution including Apache
 Hadoop, 1-3
CSV files, 1-8

D

Data Pump files, 1-8
data replication, 1-5
DataNode, 2-12
diagnostics, creating, 2-3
disk layout, 2-10
disk mirroring, 2-11
disk partitioning, 2-11

E

engineered systems, 1-3
Exadata Database Machine, 1-3
Exalytics In-Memory Machine, 1-3
external tables, 1-8

F

Flume, 2-11, 2-13
ftp.oracle.com, uploading to, 2-3

G

groups, 2-8
groups, operating system, 2-9

H

Hadoop Distributed File System, 1-3, 1-5
hadoop group, 2-8
Hadoop Map/Reduce Administration, 2-4
Hadoop version, 1-3
HBase, 2-11
HDFS, 1-3, 1-5, 2-11
HDFS data files, 1-8
Hive, 2-11
hive group, 2-8
Hive server, 2-12
Hotspot Java Virtual Machine, 2-10
Hue, 2-11
Hue server, 2-12

J

JobTracker, 2-4, 2-11, 2-12, 2-14
JobTracker, opening, 2-4

K

key-value database, 1-5
knowledge modules, 1-8

L

Linux, 2-10
loading data, 1-8

M

Mahout, 2-13
MapReduce, 1-4, 1-6, 2-11
MySQL backup, 2-12
MySQL Database, 2-10, 2-11, 2-14, 2-16

MySQL primary server, 2-12

N

NameNode, 2-11, 2-12, 2-13, 2-14

NameNode, Secondary, 2-13

NameNode, secondary, 2-13

NoSQL database, 1-5

O

oinstall group, 2-8

Oozie, 2-13

operating system users, 2-9

Oracle Data Integrator, 1-8, 2-10, 2-12, 2-14

Oracle Data Integrator Agent, 2-12

Oracle Data Integrator agent, 2-16

Oracle Data Pump files, 1-8

Oracle Direct Connector, 2-10

Oracle Direct Connector for Hadoop Distributed File System, 1-7

Oracle Exadata Database Machine, 1-3

Oracle Exalytics In-Memory Machine, 1-3

Oracle Instant Client, 2-10

Oracle Linux, 1-3, 2-10

Oracle Loader for Hadoop, 1-8, 2-10

Oracle NoSQL Database, 1-5, 2-10, 2-12, 2-14

Oracle NoSQL Database administration, 2-13, 2-16

Oracle NoSQL Database processes, 2-16

Oracle NoSQL Database registration, 2-16

Oracle NoSQL Database server processes, 2-13

Oracle R Connector for Hadoop, 1-9, 2-10

Oracle Support, creating a service request, 2-3

oracle user, 2-8

P

partitioning, 2-10

Pig, 2-11

planning applications, 1-3

port map, 2-16

port numbers, 2-15

puppet agents, 2-13

puppet master, 2-13

puppet master service, 2-16

puppet node service, 2-16

R

R Connector, 1-9

R distribution, 2-10

R language support, 1-6

remote access, 2-6

remote client access, 2-6

replicating data, 1-5

rpc.statd, 2-16

S

Secondary NameNode, 2-13

secondary NameNode, 2-13

security, 2-15

service request, creating for CDH, 2-3

services, 2-11

software components, 2-9

software framework, 1-3

Sqoop, 2-13

ssh, 2-16

T

tables, 1-8

Task Tracker Status interface, 2-5

TaskTracker, 2-12, 2-13

opening, 2-5

U

user groups, 2-8

users, operating system, 2-9

W

Whirr, 2-13

X

xinetd, 2-16

Z

ZooKeeper, 2-12