

Oracle® Big Data Discovery

Administrator's Guide

Version 1.0.0 • Revision A • March 2015

Copyright and disclaimer

Copyright © 2003, 2015, Oracle and/or its affiliates. All rights reserved.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners. UNIX is a registered trademark of The Open Group.

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.

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.

Table of Contents

Copyright and disclaimer	2
Preface	8
About this guide	8
Who should use this guide?	8
Conventions used in this document	8
Contacting Oracle Customer Support	9

Part I: Overview of Big Data Discovery Administration

Chapter 1: Introduction	11
List of administrative tasks	11
Administrative tools	12
List of Big Data Discovery logs	15
Big Data Discovery backup strategy	17
Chapter 2: Cluster Architecture	19
Cluster components	19
About a BDD cluster, nodes, and deployments	19
Diagram of a Big Data Discovery Cluster	21
Cluster of Dgraph nodes	22
Leader and follower Dgraph nodes	23
Cluster behavior	23
Load balancing and routing of requests	24
How session affinity is used	25
Startup of Dgraph nodes	25
How updates are processed	25
Role of ZooKeeper	26
How enhanced availability is achieved	26

Part II: Administering Big Data Discovery

Chapter 3: Administering Big Data Discovery with the bdd-admin Script	29
About the bdd-admin script	29
Refreshing cluster configuration	31
Configuration properties that can be modified	32
Enabling and disabling autostart	33
Starting services	34
Stopping services	34
Restarting services	35
Checking the status of services	35

Part III: Administering the Dgraph and Dgraph Gateway

Chapter 4: The Dgraph	38
About the Dgraph	38
Memory consumption by the Dgraph	39
Setting the limit of Dgraph memory consumption	40
Setting the Dgraph internal cache size	41
Managing an index merge policy	41
About an index merge policy	42
Manually forcing a merge	42
Linux ulimit settings for merges	43
Managing Dgraph core dump files	43
Appointing a new Dgraph leader node	44
About Dgraph statistics	45
Dgraph administrative operations	45
About the dgraph-admin command	45
flush	47
merge	47
stickymerge	47
stats	48
statsreset	48
logroll	48
log-status	48
log-enable	49
log-disable	49
Dgraph flags	49
Dgraph HDFS Agent flags	54
Chapter 5: The Dgraph Gateway	56
About the Dgraph Gateway	56
Dgraph Gateway configuration file	56
Starting Dgraph Gateway	58
Stopping Dgraph Gateway	58
Chapter 6: Administrating the Dgraph with the Dgraph Gateway Command Utility	60
About the Dgraph Gateway Command Utility	60
Global options for host, port, and context root	61
Allocating a bulk load port	62
Returning version information	63
Listing Dgraph nodes	63
Returning Dgraph session information	63
Warming the Dgraph cache	64

Part IV: Administering Studio

Chapter 7: Configuring Studio Settings	67
Studio settings list	67

Changing the Studio setting values	69
Chapter 8: Configuring Data Processing Settings	71
List of Data Processing Settings	71
Changing the data processing settings	74
Chapter 9: Viewing Summary Reports of Project Usage	75
About the project usage logs	75
About the System Usage page	76
Using the System Usage page	77
Chapter 10: Determining and Configuring the Locale to Use	80
Locales and their effect on the user interface	80
How Studio determines the locale to use	81
Locations where the locale may be set	81
Scenarios for selecting the locale	81
Setting the available locales	82
Selecting the default locale	82
Configuring the preferred locale for a user	84
Chapter 11: Configuring Settings for Outbound Email Notifications	86
Configuring the email server settings	86
Configuring the sender name and email address for notifications	87
Setting up the Account Created and Password Changed notifications	87
Chapter 12: Managing Projects from the Control Panel	89
Configuring the project type	89
Configuring the visibility type for a page	90
Adding and removing project members	91
Assigning project roles to project members	92
Certifying a project	93
Making a project active or inactive	93
Deleting projects	94
Part V: Controlling User Access to Studio	
Chapter 13: Configuring User-Related Settings	96
Configuring authentication settings for users	96
Configuring the password policy	97
Restricting the use of specific screen names and email addresses	98
Chapter 14: Creating and Editing Studio Users	99
About role privileges	99
Creating a new Studio user	102
Editing a Studio user	103
Deactivating, reactivating, and deleting Studio users	104

Chapter 15: Integrating with an LDAP System to Manage Users	105
About using LDAP	105
Configuring the LDAP settings and server	105
Preventing LDAP users from changing passwords in Big Data Discovery	110
Preventing encrypted LDAP passwords from being stored in Big Data Discovery	110
Assigning roles based on LDAP user groups	111
Chapter 16: Setting up Single Sign-On (SSO)	112
About using single sign-on	112
Overview of the process for configuring SSO with Oracle Access Manager	112
Configuring the reverse proxy module in OHS	113
Registering the Webgate with the Oracle Access Manager server	114
Testing the OHS URL	115
Configuring Big Data Discovery to integrate with SSO via Oracle Access Manager	116
Configuring the LDAP connection for SSO	116
Configuring the Oracle Access Manager SSO settings	117
Completing and testing the SSO integration	118

Part VI: Administering Big Data Discovery Using Enterprise Manager Cloud Control

Chapter 17: Using the Enterprise Manager for Big Data Discovery	121
Before using the Enterprise Manager	121
About the Enterprise Manager	121
The Cluster target	122
The Dgraph target	123
Information about the Dgraph HDFS Agent	125
The Studio target	125
Roles and privileges for BDD targets	127
Security credentials for BDD targets	128
Connecting to Studio over a secure port	129
Starting and stopping the Dgraph using Enterprise Manager	130
Logging for Big Data Discovery targets in Enterprise Manager	130
Configuring verbose logging for a Dgraph target	132
Dgraph Administration Operations in Enterprise Manager	133
Viewing Dgraph statistics	134
Resetting Dgraph statistics	134
Flushing the Dgraph cache	135
Merging index updates in the Dgraph	135
Rolling the Dgraph request log	136
Saving trace data for the Dgraph	136
Downloading Dgraph trace files	137

Part VII: Logging for Studio, Dgraph, and Dgraph Gateway

Chapter 18: Studio Logging	139
About logging in Studio	139
About the log4j configuration XML files	139
About the main Studio log file	140
About the metrics log file	140
Configuring the amount of metrics data to record	142
Adjusting Studio logging levels	143
Using the Performance Metrics page to monitor query performance	143
Chapter 19: Dgraph Logging	146
Dgraph request log and stdout/stderr log	146
Chapter 20: Dgraph Gateway Logging	149
Dgraph Gateway logs	149
Dgraph Gateway log entry format	151
Dgraph Gateway log entry information	153
Logging configuration	154
Customizing the HTTP access log	157

Preface

Oracle Big Data Discovery is a set of end-to-end visual analytic capabilities that leverage the power of Hadoop to transform raw data into business insight in minutes, without the need to learn complex products or rely only on highly skilled resources.

About this guide

This guide describes administration tasks associated with Oracle Big Data Discovery.

Who should use this guide?

This guide is intended for administrators who configure, monitor, and control access to Oracle Big Data Discovery.

Conventions used in this document

The following conventions are used in this document.

Typographic conventions

The following table describes the typographic conventions used in this document.

Typeface	Meaning
User Interface Elements	This formatting is used for graphical user interface elements such as pages, dialog boxes, buttons, and fields.
Code Sample	This formatting is used for sample code phrases within a paragraph.
<i>Variable</i>	This formatting is used for variable values. For variables within a code sample, the formatting is <i>Variable</i> .
File Path	This formatting is used for file names and paths.

Symbol conventions

The following table describes symbol conventions used in this document.

Symbol	Description	Example	Meaning
>	The right angle bracket, or greater-than sign, indicates menu item selections in a graphic user interface.	File > New > Project	From the File menu, choose New, then from the New submenu, choose Project.

Path variable conventions

This table describes the path variable conventions used in this document.

Path variable	Meaning
\$MW_HOME	Indicates the absolute path to your Oracle Middleware home directory, which is the root directory for your WebLogic installation.
\$DOMAIN_HOME	Indicates the absolute path to your WebLogic domain home directory. For example, if <code>bdd_domain</code> is the domain name, then the <code>\$DOMAIN_HOME</code> value is the <code>\$MW_HOME/user_projects/domains/bdd_domain</code> directory.
\$BDD_HOME	Indicates the absolute path to your Oracle Big Data Discovery home directory. For example, if <code>BDD1.0</code> is the name you specified for the Oracle Big Data Discovery installation, then the <code>\$BDD_HOME</code> value is the <code>\$MW_HOME/BDD1.0</code> directory.
\$DGRAPH_HOME	Indicates the absolute path to your Dgraph home directory. For example, the <code>\$DGRAPH_HOME</code> value might be the <code>\$BDD_HOME/dgraph</code> directory.

Contacting Oracle Customer Support

Oracle Customer Support provides registered users with important information regarding Oracle software, implementation questions, product and solution help, as well as overall news and updates from Oracle.

You can contact Oracle Customer Support through Oracle's Support portal, My Oracle Support at <https://support.oracle.com>.

Part I

Overview of Big Data Discovery Administration



Chapter 1

Introduction

This section lists administrative tasks and tools that you can use to do these tasks. It also lists all Big Data Discovery logs, and describes the backup strategy.

[List of administrative tasks](#)

[Administrative tools](#)

[List of Big Data Discovery logs](#)

[Big Data Discovery backup strategy](#)

List of administrative tasks

This topic lists top-level administrator tasks for Studio, the Dgraph, the Dgraph HDFS Agent and the Dgraph Gateway.

Section	Tasks
Overview of Big Data Discovery Administration	Learning about available administrative tools and logs used in Big Data Discovery, as well as learning about which files need to be backed up. Also, viewing the diagram of the Big Data Discovery cluster, learning about the cluster behavior, such as routing or requests, handling of data updates, and maintaining high availability.
Administering Big Data Discovery	Using the <code>bdd-admin</code> script for administering the product — starting, stopping and restarting the components, and checking the status of Big Data Discovery services.
Administering the Dgraph, the Dgraph HDFS Agent, and the Dgraph Gateway	<ul style="list-style-type: none">• Learning about the Dgraph, its memory consumption, the Dgraph internal cache, and a way to limit the Dgraph memory consumption for expensive queries.• Learning about the index merge policy for the Dgraph, and the Dgraph statistics page.• Starting and stopping the Dgraph Gateway in the WebLogic Server Administration Console.• Running the Dgraph administrative operations with the <code>bdd-admin</code> script.• Using flags for the Dgraph and for the Dgraph HDFS Agent.• Administering the Dgraph with <code>endeca-cmd</code> (known in this release as the Dgraph Gateway Command Utility).

Section	Tasks
Administering Studio	<ul style="list-style-type: none"> • Configuring framework settings. • Configuring Hadoop settings for file upload. • Managing data sources, and viewing summary reports of project usage. • Configuring the locale and email notifications. • Managing projects in the Control Panel.
Controlling User Access to Studio	<ul style="list-style-type: none"> • Configuring user-related settings in Studio. • Creating and managing users in Studio. • Integrating with an LDAP system to manage users. • Setting up Single Sign-On (SSO).
Administering Big Data Discovery using Enterprise Manager Plug-in (Enterprise Manager Cloud Control)	<ul style="list-style-type: none"> • Tasks for the Big Data Discovery targets (Cluster, Studio, Dgraph). • Running various Dgraph administrative operations, such as viewing the Dgraph statistics, or saving the Dgraph Tracing Utility data.
Logging	<ul style="list-style-type: none"> • Logging options in the <code>bdd-admin</code> script. • Logging options for Big Data Discovery targets in the Enterprise Manager plug-in. • Studio logs, their format and types, and customization options. • Dgraph Gateway logs, their format, log levels, and customization options. • Dgraph request log and stdout/sterr log.

Administrative tools

Two tools for administering Big Data Discovery exist — the Enterprise Manager plug-in, and the `bdd-admin` script. This topic introduces these administrative tools and discusses when to use each.

The Enterprise Manager plug-in for Big Data Discovery

The Enterprise Manager plug-in lets you monitor, diagnose, and manage Big Data Discovery components. The plug-in includes three targets: a target for the entire Big Data Discovery cluster, as well as targets for Studio and the Dgraph.

For information on performing administrative tasks through the plug-in, see [Using the Enterprise Manager for Big Data Discovery on page 120](#).

The `bdd-admin` script

The `bdd-admin` script lets you perform a number of administrative tasks for the Dgraph, the HDFS Agent, Studio, and the Dgraph Gateway from the command line.

For information on performing administrative tasks through the script, see [Administering Big Data Discovery with the `bdd-admin` Script on page 28](#).

Administrative tool comparison

This table illustrates which administrative tasks you can perform using the script or the Enterprise Manager plug-in. Use these guidelines:

- The `bdd-admin` script is available to you regardless of whether you are using the Enterprise Manager plug-in. Use it for all administrative tasks, or for those that you cannot do in any other way. For example, you can use it to perform administrative tasks for the Dgraph and HDFS Agent, including starting and stopping, logging, updating the configuration, and running administrative operations (such as `merge`, `ping`, `stats`, `statsreset`, `log-status`, and `logroll`).
- The Enterprise Manager plug-in is optional. The plug-in is desirable especially for monitoring and log access, thus it is recommended to be used if you have the license for it and have installed it.

This table compares these tools.

Administrative task	Enterprise Manager	<code>bdd-admin</code>	Notes
Starting and stopping Big Data Discovery (all components)	No	Yes	Enterprise Manager plug-in: You can only use it after Big Data Discovery is already up and running. Script: After you install Big Data Discovery, you use the script to start and stop the entire stack (Dgraph, Dgraph HDFS Agent, Studio, Dgraph Gateway).
Starting and stopping the Dgraph	Yes	Yes	Enterprise Manager plug-in: You can start and stop the Dgraph targets. Script: You can start and stop the Dgraph instances.
Starting and stopping Dgraph Gateway and Studio	No	Yes	Enterprise Manager plug-in: You cannot start and stop Dgraph Gateway and Studio. Script: You can use the script to start and stop both Dgraph Gateway and Studio. Note that both must be stopped and started at the same time.
Starting and stopping CDH nodes (used for Data Processing)	No	No	Enterprise Manager plug-in: You cannot start and stop CDH nodes. Script: You cannot start and stop CDH nodes.

Administrative task	Enterprise Manager	bdd-admin	Notes
Adding and removing nodes in the cluster (Dgraph nodes, Studio nodes, CDH nodes)	No	No	Enterprise Manager plug-in: You cannot add or remove nodes in the BDD deployment. Script: You cannot add or remove nodes.
Exploring Dgraph logs	Yes	Yes	You can use <code>bdd.conf</code> to specify the location of the Dgraph and HDFS Agent output files and the Dgraph Gateway log level. You can use the <code>bdd-admin</code> script to enable, disable, and check the status of extended logging features and perform a log roll for the Dgraph.
Monitoring node status	Yes	Yes	Enterprise Manager plug-in: Node status indicator lets you see if the node is up or down. Script: You can use the <code>bdd-admin</code> script to check the current status of the Dgraph, the HDFS Agent, Studio, and the Dgraph Gateway.
Creating and deleting services	No	No	Script: You cannot use the <code>bdd-admin</code> script or Enterprise Manager plug-in to create or delete services.
Enabling and disabling auto-start	No	Yes	You can use <code>bdd.conf</code> to enable the entire cluster to automatically restart after a reboot.
Refreshing configuration	No	Yes	Enterprise Manager plug-in: It does not have an option to refresh the configuration. Script: You can use the <code>bdd-admin</code> script to copy an updated version of <code>bdd.conf</code> to all servers in the BDD cluster deployment.
Running administrative operations (such as ping, merge, logroll, view and reset Dgraph statistics)	Yes	Yes	You can use administration operations for the Dgraph through the Enterprise manager plug-in and through the <code>bdd-admin</code> script.

List of Big Data Discovery logs

This topic provides a list of all the logs generated by a BDD deployment. It also has a summary of where to find logs for each BDD component, and tells you how to access logs.

This topic includes:

- [List of BDD logs on page 15](#)
- [Where to find logging information for each component on page 16](#)
- [Ways of accessing logs on page 17](#)

List of BDD logs

Log	Purpose	Default Location
WebLogic Admin Server domain log	Provides a status of the WebLogic domain for the Big Data Discovery deployment. See WebLogic Domain Log on page 150 .	\$BDD_DOMAIN/servers/AdminServer/logs/bdd_domain.log
WebLogic Admin Server server log	Contains messages from the WebLogic Admin Server subsystems. For both server logs, see WebLogic Server Log on page 150 .	\$BDD_DOMAIN/servers/AdminServer/logs/AdminServer.log
WebLogic Managed Server server log	Contains messages from the WebLogic Managed Server subsystems and applications.	\$BDD_DOMAIN/servers/<serverName>/logs/<serverName>.log
Dgraph Gateway application log	WebLogic log for the Dgraph Gateway application. See Dgraph Gateway log entry format on page 151	\$BDD_DOMAIN/servers/<serverName>/logs/<serverName>-diagnostic.log
Dgraph stdout/stderr log	Contains Dgraph startup messages, as well as warning and error messages. See Dgraph stdout/stderr log on page 147 .	\$BDD_HOME/logs/dgraph.out
Dgraph request log	Contains entries for processed Dgraph requests. See Dgraph request log on page 146 .	\$BDD_HOME/dgraph/bin/dgraph.reqlog
Dgraph tracing ebb logs	Dgraph Tracing Utility files, which are especially useful for Dgraph crashes. See Downloading Dgraph trace files on page 137 .	\$BDD_HOME/dgraph/bin/dgraph-<serverName>*.ebb
Dgraph HDFS Agent stdout/stderr log	Contains startup messages, as well as messages from operations performed by the Dgraph HDFS Agent (such as ingest operations). See the Data Processing Guide .	\$BDD_HOME/logs/dgraphHDFSAGENT.out

Log	Purpose	Default Location
Studio application log in Log4j format	Studio application log (in Log4j format). For both Studio application logs, see About the main Studio log file on page 140 .	\$BDD_DOMAIN/servers/<serverName>/logs/bdd-studio.log
Studio application log in ODL format	Studio application log (in ODL format).	\$BDD_DOMAIN/servers/<serverName>/logs/bdd-studio-odl.log
Studio metrics log in Log4j format	Studio metrics log (in Log4j format). For both Studio metrics logs, see About the metrics log file on page 140 .	\$BDD_DOMAIN/servers/<serverName>/logs/bdd-studio-metrics.log
Studio metrics log in ODL format	Studio metrics log (in ODL format).	\$BDD_DOMAIN/servers/<serverName>/logs/bdd-studio-metrics-odl.log
Data Processing logs	Contains messages resulting from Data Processing workflows. See the <i>Data Processing Guide</i> .	/opt/bdd/edp/data/edpLog*.log
CDH logs (YARN logs, Spark worker logs, ZooKeeper logs)	YARN and Spark logs from CDH processes that ran Data Processing workflows, as listed in the <i>Data Processing Guide</i> . See the Cloudera documentation for information on the ZooKeeper logs.	Available from the Cloudera Web UI for the component.

Where to find logging information for each component

This table lists how to find detailed logging information for each Big Data Discovery component:

Name of component in Big Data Discovery	Where to find logging information?
Studio	See Studio Logging on page 138 .
Data Processing	Data Processing is a component of BDD that runs on CDH nodes in the BDD deployment. For Data Processing logs, see the <i>Data Processing Guide</i> .
Dgraph Gateway (and WebLogic Server logs)	See Dgraph Gateway Logging on page 148
Dgraph	See Dgraph request log and stdout/sterr log on page 146
Dgraph HDFS Agent	The Dgraph HDFS Agent is responsible for importing and exporting Dgraph data to HDFS. For HDFS Agent logs, see the <i>Data Processing Guide</i> .

Ways of accessing logs

You can access the logs for some components of Big Data Discovery through `bdd_admin.sh dgraph-admin` and EM plug-in for BDD:

Method of accessing logs	Logging tasks
Logging options available in <code>bdd-admin.sh dgraph-admin</code>	Use the <code>bdd-admin.sh dgraph-admin</code> command for these operations on Dgraph logs: <ul style="list-style-type: none"> • logroll on page 48 • log-status on page 48 • log-enable on page 49 • log-disable on page 49
Logging options available in Enterprise Manager plug-in for Big Data Discovery	Use the Enterprise Manager plug-in to access logs for Studio, Dgraph and the BDD cluster, to configure verbose logging for the Dgraph, and to roll the Dgraph request log. <ul style="list-style-type: none"> • Logging for Big Data Discovery targets in Enterprise Manager on page 130 • Configuring verbose logging for a Dgraph target on page 132 • Rolling the Dgraph request log on page 136.

Big Data Discovery backup strategy

Oracle recommends that you back up your system to ensure the safety of your data. This topic lists the resources you should back up, as well as their locations.

Backups must be performed manually and cold. A cold backup guarantees that your project data sets, Studio database, and sample files remain in synch. This involves (at a minimum) shutting down the Dgraph and HDFS Agent to prevent them from performing an ingest during the backup procedure.

Resource	Location	Description/notes
<code>dateFormats.txt</code> <code>edp_classpath.txt</code> <code>logging.properties</code> <code>sparkContext.properties</code> <code>spark_worker_files.txt</code>	The location on HDFS defined by the <code>hdfsEdpLibPath</code> property in the <code>data_processing_CLI</code> file. By default, this is <code>/user/bdd/edp/lib</code> .	These files contain configuration settings specific to your system.

Resource	Location	Description/notes
/dataSwamp	The location on HDFS defined by the <code>edpDataDir</code> property in the <code>data_processing_CLI</code> file. By default, this is <code>/user/bdd/edp/data</code> .	This directory contains the Avro files for your sample data sets.
Data Processing log files (<code>edpLog*.log</code>)	The location on each node defined by the <code>edpJarDir</code> property in the <code>data_processing-CLI</code> file. By default, this is <code>/opt/bdd/edp/data</code> .	The Data Processing log files are located on each node that has been involved in a Data Processing job. These include the client that started the job (which could be nodes running the CLI, the Hive Table Detector, or Studio), an Oozie worker node, or a Spark worker node.
The ZooKeeper infrastructure on CDH nodes, <code>/endeca-cluster znode</code>		Refer to Cloudera's documentation for backup instructions.
HDFS and other Hadoop resources		Refer to Cloudera's documentation for backup instructions.
Studio's database		Refer to your database's documentation for backup instructions.
<code>\$MW_HOME</code>	The location defined by the <code>ORACLE_HOME</code> property in <code>bdd.conf</code> . By default, this is <code>/localdisk/Oracle/Middleware</code> .	Back up this location on the WebLogic Admin Server node and each Weblogic Managed Server node in the BDD cluster.
<code>\$DOMAIN_HOME</code>	The root directory of Studio and your WebLogic domain. By default, this is <code>\$MW_HOME/user_projects/domains/bdd_domain</code> .	Back up this location on the WebLogic Admin Server node and each Weblogic Managed Server node in the BDD cluster.
The Dgraph index	The location on the NFS defined by the <code>DGRAPH_INDEX_DIR</code> and <code>DGRAPH_INDEX_NAME</code> properties in the <code>bdd.conf</code> file.	This location contains the indexes for all of your data sets.



Chapter 2

Cluster Architecture

This section describes the architecture of a Big Data Discovery cluster.

[Cluster components](#)

[Cluster behavior](#)

Cluster components

A Big Data Discovery cluster is a deployment of Big Data Discovery on multiple machines. Such a deployment can be made up of any number of nodes — you determine the number of nodes at deployment time.

[About a BDD cluster, nodes, and deployments](#)

[Diagram of a Big Data Discovery Cluster](#)

[Cluster of Dgraph nodes](#)

[Leader and follower Dgraph nodes](#)

About a BDD cluster, nodes, and deployments

This topic provides an overview of the components in a Big Data Discovery cluster.

What is a BDD cluster?

A BDD cluster:

- Supports on-premise deployments of Big Data Discovery, both on commodity hardware and on engineered systems, such as Oracle Big Data Appliance (BDA).
- Has anywhere from three to more nodes (a minimum number of three nodes are required for the production environment, to ensure enhanced availability of query processing). For example, a production deployment can include six nodes. Each node in the cluster is known as a **BDD node**.
- Performs routing and load balancing of query requests arriving from Studio to the nodes that run the Dgraph. This assumes that there is at least one Dgraph node available to process queries arriving from Studio.

Nodes

Nodes in the BDD cluster deployment have different roles:

- They can serve as CDH cluster nodes. This is because you deploy Big Data Discovery on a set of nodes running Cloudera Distribution Including Apache Hadoop (CDH).

- They can serve as WebLogic Server nodes on which Java-based components of BDD (Studio and Dgraph Gateway) are running in the WebLogic Server.
- They can serve as Dgraph-only nodes. Together, these nodes constitute a Dgraph cluster, within the overall BDD cluster deployment. These Dgraph nodes communicate with CDH nodes and utilize ZooKeeper from the CDH installation to maintain high availability of the Dgraph processes.

For more information on nodes and their roles shown on a diagram, see [Diagram of a Big Data Discovery Cluster on page 21](#).



Note: These roles are not mutually-exclusive. For example, in demo or learning deployments, you can co-locate Dgraph instances on the same nodes that run WebLogic Server, or experiment with other configurations that have nodes serving dual roles. See the *Installation and Deployment Guide* for information on deployment scenarios and co-location.

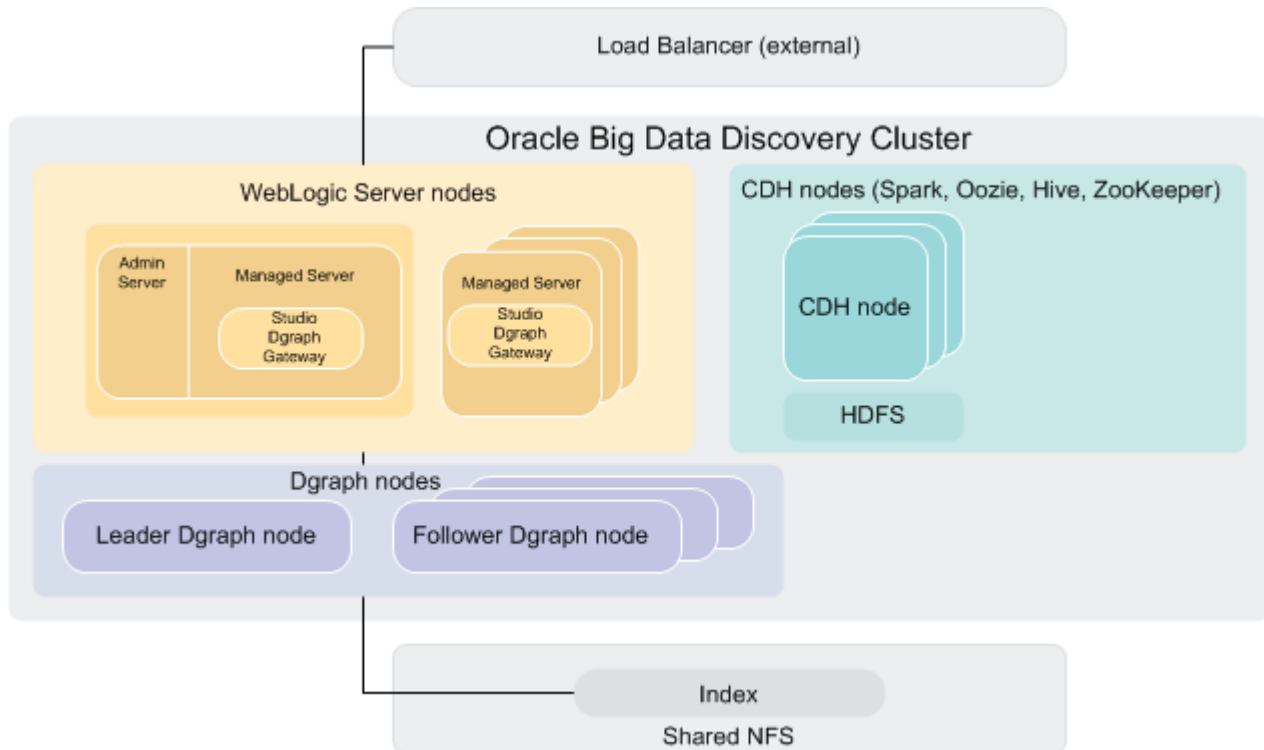
Types of BDD cluster deployments

You can choose between many ways in which to deploy BDD to utilize hardware efficiently. In the *Installation and Deployment Guide*, several recommended deployment scenarios are provided to help you efficiently deploy BDD:

- A learning, or demo deployment on one or two machines (this deployment is not intended to be turned into a production deployment).
- A production deployment on a set of six machines (three of which are running a CDH cluster only, two run WebLogic Servers with Studio and Dgraph Gateway, and one node is dedicated to running the Dgraph). The number of nodes in the production deployment can be less than six (with some software components co-located), and can be more, depending on your needs.

Diagram of a Big Data Discovery Cluster

This diagram illustrates a cluster of Big Data Discovery nodes deployed on top of an existing CDH cluster.



This diagram depicts a suggested deployment topology for production, although many configurations are possible. For information on staging and learning, demo and production-level deployment topology, see the *Installation and Deployment Guide*.

In this diagram, starting from the top, the following components of the Big Data Discovery cluster deployment are included:

- An optional external load balancer serves as the single point of entry to the Big Data Discovery cluster. All browser requests are routed through this load balancer to Studio nodes.



Note: Although it is recommended to use an external load balancer in your deployment, it is optional. For information, see [Load balancing and routing of requests on page 24](#).

- The Big Data Discovery cluster comprises three categories of nodes:
 1. Nodes that host WebLogic Server with Studio and Dgraph Gateway.
 2. CDH only nodes. These nodes do not host WebLogic Server or Dgraph instances. They run Data Processing jobs, within a Big Data Discovery deployment.
 3. Dgraph nodes. These nodes are solely dedicated to hosting Dgraph instances.
- **WebLogic Server nodes.** These nodes represent machines on which WebLogic Server is deployed that is hosting two Java applications — Studio and Dgraph Gateway. Note that WebLogic Server nodes and Dgraph nodes can be stopped and started independently of each other, although in practice both must be running in order to service requests.

- **CDH nodes.** Big Data Discovery is deployed on top of an existing CDH (or Hadoop) cluster. This diagram shows only those CDH nodes on which BDD is deployed. These CDH nodes represent a subset of the entire pre-existing CDH cluster, onto which BDD is deployed. These nodes have both CDH and BDD installation on them and share access to HDFS. Optimally, three CDH nodes are required for hosting ZooKeeper instances. ZooKeeper maintains a cluster state for all participating members of the Big Data Discovery cluster, in particular, it ensures automatic Dgraph leader node election, in case the leader Dgraph node fails.
- **Dgraph nodes.** These nodes form a Dgraph cluster that is part of the larger BDD cluster deployment. One node serves as the leader Dgraph node, and the remaining nodes are follower Dgraph nodes. All nodes in the Dgraph cluster have write access to a shared file system (NFS) on which the index is stored. Only the leader Dgraph node writes to the index located on the file system. Follower Dgraph nodes can only read from the index. The index includes internal indexes for each of the data sets in BDD.
- Enterprise Manager for Big Data Discovery is not shown on this diagram. It can be optionally used with any Big Data Discovery deployment. When used, Enterprise Manager is installed on a separate WebLogic Server. For more information, see [Using the Enterprise Manager for Big Data Discovery on page 120](#).

Cluster of Dgraph nodes

A typical BDD cluster deployment includes a set of machines that are solely dedicated to running the Dgraph. This set of machines is known as the Dgraph cluster.

A **Dgraph cluster** is a set of Dgraphs that together handle requests for data sets in Big Data Discovery. Requests arriving from Studio are routed and load-balanced between the Dgraph nodes. One of these Dgraph nodes is responsible for handling all write operations (updates, configuration changes), while the remaining Dgraphs serve as read-only. All Dgraph nodes in the cluster utilize an index residing on shared storage.

The leader and follower Dgraph nodes differ in the types of queries they can process, however, this is transparent to the end users of Big Data Discovery. The allocation of leader and follower Dgraph node roles is performed by the BDD cluster automatically.

A **Dgraph node** is a node in BDD cluster deployment that runs the Dgraph. The Dgraph is the main computational module that provides search, refinement computation, Guided Navigation, and many other features, all of which you can observe and use in Studio.

In a BDD cluster deployment, you can have only one cluster of Dgraph nodes. All nodes in BDD that run Studio and Dgraph Gateway in WebLogic Server talk to the same single cluster of Dgraph nodes. The Dgraph cluster can have any number of nodes, even though a certain number of Dgraph nodes is recommended for production environment. For more information, see the *Installation and Deployment Guide*.

Dgraph Cluster role

A Dgraph cluster is responsible for:

- **Enhanced availability of query processing by the Oracle Big Data Discovery.** In a cluster of Dgraph nodes, if one of the Dgraph nodes fails, queries continue to be processed by other Dgraph nodes.
- **Increased throughput.** At deployment time, you can add one or more Dgraph nodes to the same Dgraph cluster. This lets you spread the query load across them, without the need to increase storage requirements at the same rate.

Leader and follower Dgraph nodes

This topic introduces the terms used to describe Dgraphs — the leader and follower nodes.

Leader Dgraph node

The **leader node** is a single Dgraph node responsible for receiving and processing updates to the index and configuration. This node also does query processing, like other nodes. This node is responsible for generating information about the latest versions of the data set index, and propagating this information to the follower Dgraph nodes.

The Dgraph Gateway automatically determines which Dgraph node is the leader Dgraph node. The other Dgraph nodes are follower Dgraph nodes. Thus, each BDD cluster deployment with multiple Dgraph nodes is started with one leader Dgraph node and a number of follower Dgraph nodes.

The leader Dgraph node periodically receives full or incremental index updates. It also receives administration or configuration updates. It is the only node in the Dgraph cluster that makes updates to the index. After processing updates, the leader publishes a new version of the data and notifies all follower nodes, alerting them to start using the updated version of the index. The follower nodes acquire read-only access to an updated version of the index.

Follower Dgraph node

A **follower node** is a node in the Dgraph cluster responsible for processing queries arriving from Studio. Typically, in any Big Data Discovery cluster deployment, there is a subset of nodes serving as the Dgraph follower nodes. The follower nodes do not update the index. When the Dgraph nodes are started, the Dgraph Gateway elects a leader Dgraph node, with the other Dgraph nodes being follower nodes.

During the process of acquiring access to the recently updated index, follower nodes continue to serve queries. Each query is processed against a specific version of the index available to it at any given time.

Cluster behavior

Many scenarios of Big Data Discovery deployment clusters are possible. This section describes how the BDD cluster behaves and maintains enhanced availability in various scenarios, such as during node startup, updates to the indexes, or individual node failures.

[Load balancing and routing of requests](#)

[How session affinity is used](#)

[Startup of Dgraph nodes](#)

[How updates are processed](#)

[Role of ZooKeeper](#)

[How enhanced availability is achieved](#)

Load balancing and routing of requests

This topic discusses the load balancing and routing of requests from Studio nodes to the Dgraph nodes in Oracle Big Data Discovery.

Load balancing of requests

Depending on your deployment strategy, to the external clients, the entry point of contact with the on-premise deployment of the Big Data Discovery cluster could be either any Studio-hosting node in the cluster, or an external load balancer configured in front of Studio instances.

The Big Data Discovery cluster relies on the following two levels of load balancing of requests:

1. Load balancing of requests across the nodes hosting multiple instances of Studio. This task should be performed by an external load balancer, if you choose to use it in your deployment (an external load balancer is not included in the Big Data Discovery package).

If an external load balancer is used, it receives all requests and distributes them across all of the nodes in the Big Data Discovery cluster deployment that host the Studio application. Once a request is received from a Studio node, it is routed by BDD to the appropriate Dgraph node.

If an external load balancer is not used, external requests can be sent to any Studio node. They are then load-balanced between the nodes hosting the Dgraph.

2. Load balancing of requests across the Dgraph nodes. This task is automatically handled by the BDD cluster — the Big Data Discovery software accepts requests from its Studio and Data Processing components on any node hosting the Dgraph, and provides internal load balancing of these requests across the other Dgraph-hosting nodes in the cluster.

Routing of requests

The Big Data Discovery cluster automatically directs requests to the subset of the cluster nodes hosting the Dgraph instances.

The following statements describe the behavior of the BDD cluster for routing of requests to Dgraph nodes:

- Requests can be submitted from Studio or Data Processing components to any Dgraph Gateway in the BDD cluster, which in turn will route the request to an appropriate Dgraph node.

For example, if the request is an updating request, such as a data loading request, or a configuration update, it is routed to the leader Dgraph node in the cluster. If the request represents a non-updating (query processing) request, it is routed to the leader Dgraph node or to any of the follower Dgraph nodes. If a BDD cluster has only one node hosting the Dgraph, this node serves as the leader (with no followers).

- Non-updating requests are load-balanced using round-robin algorithm across the Dgraph nodes, for processing.
- The Big Data Discovery cluster utilizes session affinity for all requests arriving from Studio to the Dgraph, by relying on session ID in the header of each Studio request. Requests from the same session ID are always routed to the same Dgraph node in the cluster. This improves query processing performance by efficiently utilizing the Dgraph cache, and improves performance of caching entities (known in Studio as views).

How session affinity is used

When a WebLogic Server node hosting Studio and Dgraph Gateway receives a client request, it routes the request to a Dgraph node using session affinity, based on the session ID specified in the header of the request.

When end users issue queries, Studio sets session ID for the requests in the HTTP headers. Requests with the same session ID are routed to the same Dgraph node. If the BDD software cannot locate the session ID, it relies on a round-robin strategy for deciding to which Dgraph node the request should be routed.

Session affinity is enabled by default, via the `endeca-session-id-key` and `endeca-session-id-type` properties in the `EndecaServer.properties` file of the Dgraph Gateway (do not change these values):

Property	Description
<code>endeca-session-id-key=name</code>	This is the name of the object checked by the specified method. Its default value is <code>session ID</code> .
<code>endeca-session-id-type=method</code>	This is the method used for establishing session affinity. Its default value is <code>HEADER</code> .

Startup of Dgraph nodes

Once the Big Data Discovery cluster is started, it activates the Dgraph processes on a subset of the nodes that are hosting the Dgraph instances. This topic discusses the behavior of the Dgraph nodes at startup.

On startup, the following actions take place:

- Any Dgraph node is started in either a leader or follower mode and in any order. Any number of follower nodes and one leader node are started in each Big Data Discovery cluster deployment.



Note: If the BDD cluster deployment has only one node that runs the Dgraph, then this node serves as the leader. This configuration is possible but is not recommended for production environments.

- Once started, each Dgraph node registers with ZooKeeper that manages the distributed state of the Dgraph nodes. The leader node determines the current version of the index and informs ZooKeeper.
- Follower nodes do not alter the index in any way; they continue answering queries based on the version of the index to which they have access at startup, even if the leader Dgraph node is in the process of updating, merging, or deleting indexed versions on disk. Follower Dgraph nodes do not receive updating requests; They acquire access to the new index once the updates complete. For information, see the next topic.

How updates are processed

In a Dgraph cluster (it is part of the BDD cluster deployment), updates to the records in the indexes and updates to the configuration are routed to the leader Dgraph node.

The leader Dgraph node processes the update and commits it to the on-disk index. Upon completion, all follower nodes are informed that a new version of the index is available. The leader Dgraph node and all

follower Dgraph nodes can continue to use the previous version of the index to finish query processing that had started against that version.

As each Dgraph node finishes processing queries on the previous version, it releases references to it. Once the follower nodes are notified of the new version, they acquire read-only access to it and start using it.

Role of ZooKeeper

The ZooKeeper utility provides configuration and state management and distributed coordination services to Dgraph nodes of the Big Data Discovery cluster. It ensures high availability of the query processing by the Dgraph nodes in the cluster.

ZooKeeper is part of the CDH package. CDH package is assumed to be installed on all CDH nodes in the BDD cluster deployment. Even though ZooKeeper is installed on all CDH nodes in the BDD cluster, it may not be running on all of these nodes. To ensure availability of a clustered Dgraph deployment, configure an odd number (at least three) of CDH nodes to run ZooKeeper instances. This will avoid ZooKeeper being a single point of failure.

ZooKeeper has the following characteristics:

- It is a shared information repository that provides a set of distributed coordination services. It ensures synchronization, event notification, and coordination between the nodes. The communication and coordination mechanisms continue to work in the case when connections or Dgraph-hosting nodes fail.
- It provides communication between the Dgraph-hosting nodes, ensuring that if one of these nodes fails, requests are sent to other active Dgraph nodes, until the node rejoins the Dgraph cluster (this is true if more than two instances of ZooKeeper are running in the deployment).
- It provides communication between Dgraph nodes. It controls the election of the Dgraph leader node in the Dgraph cluster, in case the current leader Dgraph node fails. The newly-elected leader Dgraph node identifies the most recent version of the index, and, using ZooKeeper, informs other nodes of the current version of the index.

To summarize, in order to run, ZooKeeper requires a majority of its hosting nodes to be active. Therefore, it is recommended that ZooKeeper runs on an odd number (at least three) of the CDH nodes in the deployed Big Data Discovery cluster. You can ensure this during the installation, when running the deployment script.

How enhanced availability is achieved

This topic discusses how the BDD cluster deployment ensures enhanced availability of query-processing.



Important: The BDD cluster deployment provides enhanced availability, but does not provide high availability. This topic discusses the cluster behavior that enables enhanced availability, and notes instances where system administrators need to take action to restore services.

The following three sections discuss the BDD cluster behavior for providing enhanced availability.



Note: This topic discusses BDD deployments with more than one running instance of the Dgraph. Even though you can deploy BDD on a single node, such deployments can only serve development environments, as they do not guarantee the availability of query processing in BDD. Namely, in a BDD deployment where only one node is hosting a single Dgraph instance, a failure of the Dgraph node shuts down the Dgraph process.

Availability of WebLogic Server nodes hosting Studio

When a WebLogic Server node goes down, Studio also goes down. As long as the BDD cluster utilizes an external load balancer and consists of more than one WebLogic Server node on which Studio is started, this does not disrupt Big Data Discovery operations.

If a WebLogic Studio node hosting Studio fails, the BDD cluster (that uses an external load balancer) stops using it and relies on other Studio nodes, until you restart it.

Availability of Dgraph nodes

The ZooKeeper ensemble running on a subset of CDH nodes ensures the enhanced availability of the Dgraph cluster nodes and services:

- Failure of the leader Dgraph node. When the leader Dgraph node goes offline, the BDD cluster elects a new leader node and starts sending updates to it. During this stage, follower nodes continue maintaining a consistent view of the data and answering queries. You should manually restart this node with the `bdd-admin` script. When the node that was the leader node is restarted and joins the cluster, it becomes one of the follower nodes. It is also possible that the leader node is restarted and joins the cluster before the cluster needs to appoint a new leader node. In this case, the node continues to serve as the leader node.

If the leader node changes, the BDD cluster starts routing updating requests to the newly-elected leader Dgraph node.

- Failure of a follower Dgraph node. When one of the follower nodes goes offline, the BDD cluster starts routing requests to other available nodes. You should manually restart this node using the `bdd-admin` script. Alternatively, you can start the Dgraph through the Enterprise Manager plug-in for Big Data Discovery (if you are using it). Once the node is restarted, it rejoins the cluster, and the cluster adjusts its routing information accordingly.

Availability of ZooKeeper instances

The ZooKeeper instances themselves must be highly available. The following statements describe the requirements in detail:

- Each CDH (Cloudera Distribution for Hadoop) node in the BDD cluster deployment can be optionally configured at deployment time to host a ZooKeeper instance. To ensure availability of ZooKeeper instances, it is recommended to deploy them in a cluster of their own, known as an ensemble. At deployment time, it is recommended that a subset of the CDH nodes is configured to host ZooKeeper instances. As long as a majority of the ensemble is running, ZooKeeper services are used by the BDD cluster. Because ZooKeeper requires a majority, it is best to start an odd number of its instances — this means that ZooKeeper must be started on at least three CDH nodes in the BDD cluster. A CDH node hosting a ZooKeeper instance assumes responsibility for ensuring the ZooKeeper process uptime — it will start ZooKeeper when BDD is deployed, and will restart it should it stop running.

To summarize, although ZooKeeper can run on only one CDH node, to ensure high availability of ZooKeeper instances, ZooKeeper must run on at least three CDH nodes (or an odd number of nodes that is greater than three) in any BDD cluster. This prevents ZooKeeper itself from being a single point of failure.

- If you do not configure at least three CDH nodes to run ZooKeeper, it will be a single point of failure. Should ZooKeeper fail, access to the data sets served by BDD becomes read-only. No updates, writes, or changes of any kind are possible while ZooKeeper in the BDD cluster is down. To recover from this situation, the CDH node that was running a failed ZooKeeper must be restarted or replaced (the action required depends on the nature of the failure).

Part II

Administering Big Data Discovery



Chapter 3

Administering Big Data Discovery with the bdd-admin Script

This section describes the `bdd-admin` script and how you can use it to perform different administrative operations for the Dgraph, the Dgraph HDFS Agent, Studio, and the Dgraph Gateway.

[About the bdd-admin script](#)

[Refreshing cluster configuration](#)

[Enabling and disabling autostart](#)

[Starting services](#)

[Stopping services](#)

[Restarting services](#)

[Checking the status of services](#)

About the bdd-admin script

You can use the `bdd-admin` script to perform a number of administrative tasks for the BDD cluster from the command line.

The `bdd-admin` script must be run from the Admin Server by a user who has passwordless sudo enabled on all nodes in the cluster. The script is located in the `$BDD_HOME/BDD_manager/bin` directory.

The script has the following syntax:

```
./bdd-admin.sh <command> [component] <option>
```

The following sections describe the commands, components, and options the `bdd-admin` script supports.

Commands

The command argument determines the action the script will perform. This argument is required. The following table describes the commands the `bdd-admin` script supports.

Command	Description
<code>refresh-config</code>	Updates the configuration for all nodes in the cluster by copying a modified version of the configuration file from the Admin Server to every other node.
<code>autostart</code>	Enables and disables automatic restart for the specified services.
<code>start</code>	Starts the specified services if they are not already running.

Command	Description
stop	Stops the specified services if they are running.
restart	Restarts the specified services.
status	Returns the status of the specified services.
dgraph-admin	Performs administrative operations for the Dgraph. For more information, see Dgraph administrative operations on page 45 .
update-model	Not supported. Updates the models used by Data Enrichment modules, or reverts all changes made to the models.

Components

The component argument enables you to run the script on a specific service. This argument is optional. If you omit it, the script will run on all services.



Note: Some commands do not accept all of the following component arguments, and some don't allow you to specify a component at all.

Component	Description
--dgraph	Runs the script on the Dgraph only. Note that running some commands on the Dgraph will also have an effect on the HDFS Agent.
--agent	Runs the script on the HDFS Agent only.
--bddServer	Runs the script on Studio and the Dgraph Gateway.

Options

The option argument specifies the node(s) the script will run on. This argument is required.



Note: Some commands can only be run on all nodes and therefore don't accept the option argument.

Option	Description
--all	Runs the script on all BDD nodes in the cluster.

Option	Description
<code><hostname></code>	<p>Runs the script on the specified node. You must provide the node's fully qualified hostname; for example, <code>web009.us.example.com</code>.</p> <p>Additionally, the name you provide must match one defined in the <code><COMPONENT>_SERVERS</code> property for the service(s) the script will run on. For example, if you run the script with the <code>--dgraph</code> component argument, the hostname you provide must match a hostname listed in the <code>DGRAPH_SERVERS</code> property.</p>

bdd-admin script help

You can obtain usage information for the `bdd-admin` script and its commands by running it with the `--help` flag:

```
./bdd-admin.sh --help
```

For information on a specific command, provide the command's name after the `--help` flag:

```
./bdd-admin.sh --help refresh-config
```

Refreshing cluster configuration

You update the configuration for the entire Big Data Discovery cluster by making the desired changes to the `bdd.conf` file on the Admin Server, then running the `bdd-admin` script with the `refresh-config` command. This command uses the SCP protocol to copy `bdd.conf` from the Admin Server to all other nodes in the cluster.



Important: Only certain properties in `bdd.conf` in the `$BDD_HOME/BDD_manager/conf` directory can be updated. Modifying others can prevent your BDD cluster deployment from starting properly. For more information, see [Configuration properties that can be modified](#).

The `bdd-admin` script with the `refresh-config` command runs on the entire BDD cluster. You cannot specify a particular component or host for it to run on.

After you run the script with `refresh-config`, you must restart the BDD cluster for the changes to take effect.

To refresh the BDD cluster configuration:

1. On the Admin Server, open `bdd.conf` in the `$BDD_HOME/BDD_manager/conf` directory, in any text editor and make the desired changes.

Be sure to save the file before closing.

2. Run the following from the command line:

```
./bdd-admin.sh refresh-config <configuration_file>
```

Where `<configuration_file>` is the relative path to the edited version of `bdd.conf`.

3. Activate the new configuration by running the following from the command line:

```
./bdd-admin.sh restart --all
```

Example

The following commands copy `bdd.conf` from the Admin Server to all other nodes, then restart the cluster:

```
./bdd-admin.sh refresh-config ../bdd.conf
./bdd-admin.sh restart --all
```

Configuration properties that can be modified


Configuration properties that can be modified

You can modify some of the properties in `bdd.conf` before running the `refresh-config` command, but changing others could have negative effects on your cluster. For example, changing the value of `ADMIN_SERVER` would prevent Big Data Discovery from starting.

The table below describes the properties that can be modified. Be sure to read this information carefully before making changes. Do not make changes to any other properties in this file.

Note that the `DGRAPH_*` and `AGENT_OUT_FILE` parameters will be used when the `bdd-admin` script is run. This means that any changes will be applied for any Dgraph (and Dgraph HDFS Agent) that is subsequently restarted.

Property	Description
<code>JAVA_HOME</code>	The JDK used when starting the BDD components. If you change this value, you must also update the location used by the CLI and Studio. Note that this must be in the same location on all nodes in the cluster.
<code>DGRAPH_OUT_FILE</code>	The path to the Dgraph's stdout/stderr file.
<code>DGRAPH_INDEX_DIR</code>	The path to the Dgraph index on the NFS. This location contains the directory defined by <code>DGRAPH_INDEX_NAME</code> . You must prepare the index files on the NFS before changing the value of this property.
<code>DGRAPH_INDEX_NAME</code>	The name of the Dgraph index, which is located in the directory defined by <code>DGRAPH_INDEX_DIR</code> . You must prepare the index files on the NFS before changing the value of this property.
<code>DGRAPH_THREADS</code>	<p>The number of threads the Dgraph starts with. Oracle recommends the following:</p> <ul style="list-style-type: none"> • For machines running only the Dgraph, the number of threads should be equal to the number of CPU cores on the machine. • For machines running the Dgraph and other BDD components, the number of threads should be the number of CPU cores minus 2. For example, a machine with 4 cores should have 2 threads. <p>Be sure that the number you use is in compliance with the licensing agreement.</p>

Property	Description
DGRAPH_CACHE	The amount size of the Dgraph cache, in MB. There is no default value for this property, so you must provide one. For enhanced performance, Oracle recommends allocating at least 50% of the node's available RAM to the Dgraph cache. If you later find that queries are getting cancelled because there is not enough available memory to process them, you should increase this amount.
DGRAPH_ADDITIONAL_ARG	 Note: This property is only intended for use by Oracle Support. Defines one or more flags to start the Dgraph with. Each flag must be quoted. Note that you cannot include flags that map to properties in <code>bdd.conf</code> . For more information on Dgraph flags, see Dgraph flags .
AGENT_OUT_FILE	The path to the HDFS Agent's stdout/stderr file.

Enabling and disabling autostart

You can enable and disable autostart for the Dgraph, the HDFS Agent, Studio, and the Dgraph Gateway services by running the `bdd-admin` script with the `autostart` command. Services that have autostart enabled start automatically after their host machines are rebooted.



Note: Services are only restarted after their host machines are rebooted. Crashed services and failed nodes must be restarted manually.

Run the following command from the Admin Server:

```
./bdd-admin.sh autostart <args> [component] <option>
```

You must provide one of the following arguments:

- `on` enables autostart for the specified service(s)
- `off` disables autostart for the specified service(s)

You can specify any of the component and option arguments listed in [Components on page 30](#).

Examples

The following command enables autostart for every Dgraph, HDFS Agent, Studio, and Dgraph Gateway service in the cluster:

```
./bdd-admin.sh autostart on --all
```

The following command disables autostart for Studio and the Dgraph Gateway services on the `web009.us.example.com` node:

```
./bdd-admin.sh autostart off --bddServer web009.us.example.com
```

Starting services

You can start the Dgraph, the HDFS Agent, Studio, and the Dgraph Gateway by running the `bdd-admin` script with the `start` command.

Run the following command from the Admin Server:

```
./bdd-admin.sh start [component] <option>
```

You can specify any of the component and option arguments listed in [About the bdd-admin script on page 29](#).



Note: If you start only the Dgraph, the script starts the HDFS Agent, as well.

Examples

The following command starts every Dgraph, HDFS Agent, Studio, and Dgraph Gateway service on all nodes in the cluster:

```
./bdd-admin.sh start --all
```

The following command starts the Dgraph and HDFS Agent services on the `web009.us.example.com` node:

```
./bdd-admin.sh start --dgraph web009.us.example.com
```

Stopping services

You can stop the Dgraph, the HDFS Agent, Studio, and the Dgraph Gateway by running the `bdd-admin` script with the `stop` command.

Run the following command from the Admin Server:

```
./bdd-admin.sh stop [component] <option>
```

You can specify any of the component and option arguments listed in [About the bdd-admin script](#).



Note: If you stop only the Dgraph, the script automatically stops the HDFS Agent, as well.

When stopping the Dgraph, the script terminates all currently running processes if it does not shut down within 30 seconds. When stopping the HDFS Agent (either directly or by stopping the Dgraph), the script waits for all processes to complete before shutting it down.

Examples

The following command stops every Dgraph, HDFS Agent, Studio, and Dgraph Gateway service in the cluster:

```
./bdd-admin.sh stop --all
```

The following command stops just the Studio and Dgraph Gateway services on the `web009.us.example.com` node:

```
./bdd-admin.sh stop --bddServer web009.us.example.com
```

Restarting services

You can restart the Dgraph, the HDFS Agent, Studio, and the Dgraph Gateway by running the `bdd-admin` script with the `restart` command.

Run the following command from the Admin Server:

```
./bdd-admin.sh restart [component] <option>
```

You can specify any of the component and option arguments listed in [About the bdd-admin script](#).

Examples

The following command restarts every Dgraph, HDFS Agent, Studio, and Dgraph Gateway service in the cluster:

```
./bdd-admin.sh restart --all
```

The following command restarts just the HDFS Agent service on the `web009.us.example.com` server:

```
./bdd-admin.sh restart --agent web009.us.example.com
```

Checking the status of services

You can check the status of the Dgraph, the HDFS Agent, Studio, and the Dgraph Gateway by running the `bdd-admin` script with the `status` command.

Run the following command from the Admin Server:

```
./bdd-admin.sh status [component] <option>
```

You can specify any of the component and option arguments listed in [About the bdd-admin script](#).

When the script runs, it checks to see if the specified services are running, unresponsive, or stopped. It outputs one of the following messages for each service, depending on its status:

```
[<hostname>] <service> (pid <number>) is running...
[<hostname>] <service> (pid <number>) is not responsive...
[<hostname>] <service> is stopped.
```

Examples

The following command returns the status of every Dgraph, HDFS Agent, Studio, and Dgraph Gateway service in the cluster:

```
./bdd-admin.sh status --all
```

If all components were running, the script's output would be similar to the following:

```
[<hostname>] Dgraph (pid <number>) is running...
[<hostname>] Agent (pid <number>) is running...
[<hostname>] Studio and Endeca Server (pid <number>) is running...
```

The following command returns the status of just the Dgraph service on the `web009.us.example.com` server:

```
./bdd-admin.sh status --dgraph web009.us.example.com
```

If the Dgraph were not running, the script would output the following:

```
[web009] Dgraph is stopped.
```

Part III

Administering the Dgraph and Dgraph Gateway



Chapter 4

The Dgraph

This section describes the Dgraph, its administrative operations and flags. It also describes various Dgraph characteristics and behavior, such as memory consumption, Dgraph cache, the index merge policy, and managing the Dgraph core dump files.

[*About the Dgraph*](#)

[*Memory consumption by the Dgraph*](#)

[*Setting the limit of Dgraph memory consumption*](#)

[*Setting the Dgraph internal cache size*](#)

[*Managing an index merge policy*](#)

[*Managing Dgraph core dump files*](#)

[*Appointing a new Dgraph leader node*](#)

[*About Dgraph statistics*](#)

[*Dgraph administrative operations*](#)

[*Dgraph flags*](#)

[*Dgraph HDFS Agent flags*](#)

About the Dgraph

The Dgraph uses proprietary data structures and algorithms that allow it to provide real-time responses to queries.

The Dgraph stores the ingested data in the index with which the Dgraph was started. After the data is stored in the index, the Dgraph receives client requests from Studio, queries the index, and returns the results.

The Dgraph is stateless. This design requires that a complete query is sent to it for each request. The stateless design facilitates the addition of Dgraphs for load balancing and redundancy — any replica of a Dgraph can reply to queries independently of other replicas.

An Oracle Big Data Discovery cluster has one or more Dgraph processes that handle end-user query requests accessing the index on shared storage. One of the Dgraphs in a Big Data Discovery cluster is the leader and therefore responsible for handling all write operations (updates, configuration changes), while the remaining Dgraphs serve as read-only followers.

Dgraph Tracing Utility

The Dgraph has an internal diagnostic program, called the Dgraph Tracing Utility, that constantly keeps track of all Dgraph operations. The Tracing Utility is started automatically when the Dgraph starts and it is shut

down automatically when the Dgraph stops. That is, the Tracing Utility cannot be started or shut down manually.

The Tracing Utility stores the Dgraph target trace data it collects in *.ebb files, which are useful in analyzing Dgraph crashes. The files are intended for use by Oracle Support. The files are saved in the `$DGRAPH_HOME/bin` directory.

The trace data files are saved to disk automatically after these events:

- On normal program exit, such as stopping the Dgraph with the `bdd-admin.sh` script.
- When a crash is caught in a top-level exception handler.

You can also manually save the trace data, as described in [Saving trace data for the Dgraph on page 136](#).

Additionally, you can download the *.ebb files, as described in [Downloading Dgraph trace files on page 137](#).

Memory consumption by the Dgraph

This topic discusses the logic used by the Dgraph to control its memory consumption.

The Dgraph query performance depends on characteristics of your specific deployment — query workload and complexity, the characteristics of the loaded records, and the size of the index.

These statements describe how the Dgraph utilizes memory:

- After the installation, when the Dgraph is started it allocates considerable amounts of virtual memory on the system. This is needed for ingesting data and executing queries, including those that are complex. This is an expected behavior and is observable if you use system diagnostic tools.
- If the Dgraph is installed on a machine that is hosting other processes, other memory-intensive processes are present in the operating system and require memory. In this case, the Dgraph releases a significant portion of its physical memory quickly. Without such pressure, that is in cases when the Dgraph is the sole process on the hosting machine, the Dgraph may retain the physical memory indefinitely. This is an expected behavior.

Because of this, depending on your deployment requirements, such as the size of your deployment, it may be highly desirable to deploy the Dgraph instances on servers dedicated solely to each of the Dgraph processes (this means that these machines are not hosting any other processes, for BDD or other applications).

- By default, the memory limit that the Dgraph is allowed to use on the machine is set to 80% of the machine's available RAM. This behavior ensures that the Dgraph does not run out of memory on the machine hosting the Dgraph. In other words, with this limit in place, the Dgraph is protected from running into out-of-memory performance issues.
- In addition to the default memory consumption limit of 80% of RAM that the Dgraph uses right after it is installed and started, after the installation it is possible to set a custom limit on the amount of memory the Dgraph can consume, using the Dgraph `--memory-limit` flag. If this limit is set, then, upon the Dgraph restart, the amount of memory required by the Dgraph to process all current queries cannot exceed this custom limit.



Note: The Dgraph `--memory-limit` flag is intended for Oracle Support. For information on how to set it, see [Setting the limit of Dgraph memory consumption on page 40](#). Also, a value of 0 in for the flag means there is no limit set on the amount of memory the Dgraph can use. In this case, you should be aware that the Dgraph will use all the memory on the machine that it can allocate for its processing without any limit, and will not attempt to cancel any queries that may require the

most amount of memory. This, in turn, may lead to out-of-memory page thrashing and require manually restarting the Dgraph.

- Once the Dgraph reaches a memory consumption limit (it could be the default limit of 80% of RAM, or a custom memory limit set with `--memory-limit`), it starts to automatically cancel queries, beginning with the query that is currently consuming the most amount of memory. When the Dgraph cancels a query, it logs the amount of memory the query was using and the time it was cancelled for diagnostic purposes.
- In addition to the memory consumption limit, before you install Big Data Discovery, you can specify the Dgraph cache size, using the `DGRAPH_CACHE` property in the `bdd.conf` file located in your installation directory. The orchestration script uses this value at installation time. You can adjust the size of `DGRAPH_CACHE` later, at any point after the installation. For information, see [Setting the Dgraph internal cache size on page 41](#).
- There is one additional consideration about the Dgraph cache that is useful to keep in mind, before you decide to adjust the cache size:

While the Dgraph typically operates within the limits of its configured Dgraph cache size, it is possible for the cache to become over-subscribed for short periods of time. During such periods, the Dgraph may use up to 1.5 times more cache than it has configured. It is important to note that the Dgraph does not expect to routinely reach an increase in its configured cache usage. When the cache size reaches the 1.5 times threshold, the Dgraph starts to more aggressively evict entries that consume its cache, so that the cache memory usage can be reduced to its configured limits. This behavior is not configurable by the system administrators.

Setting the limit of Dgraph memory consumption

It is possible to specify the custom memory limit the Dgraph is allowed to use for processing. If the memory limit is changed, this overrides the default memory consumption setting in the Dgraph that is set to 80% of the machine's available RAM.



Note: Changing the limit on Dgraph memory consumption is recommended to be done by Oracle Support.

By default, the memory limit that the Dgraph is allowed to use is 80% of the machine's available RAM. This behavior ensures that the Dgraph never runs out of memory during the course of its query processing or data ingest activity.

You can override the default limit and set a custom limit on the amount of memory the Dgraph can consume in MB, using the `--memory-limit` flag. If this value is set, then the amount of memory required by the Dgraph to process all current queries can't exceed this limit.

Once the Dgraph reaches a memory consumption limit set with this flag, then, similar to how it behaves with the default memory limit of 80%, the Dgraph starts to cancel queries, beginning with the query that is consuming the most amount of memory. When the Dgraph cancels a query, it logs the amount of memory the query was using and the time it was cancelled for diagnostic purposes.

The Dgraph `--memory-limit` can be set after the installation through the `DGRAPH_ADDITIONAL_ARG` parameter in the `bdd.conf` file in the `$BDD_HOME/BDD_manager/conf` directory.

Using the `--memory-limit` flag with a value of 0 means there is no limit set on the amount of memory the Dgraph can use.

For information on all Dgraph flags, see [Dgraph flags on page 49](#).

To change the memory limit:

1. Go to `$BDD_HOME/BDD_manager/conf` directory and locate the `bdd.conf` file.
2. In the setting for `DGRAPH_ADDITIONAL_ARG`, specify the `--memory-limit` flag.
3. Save the `bdd.conf` file.
4. Run the `bdd-admin.sh refresh-config` command.

This refreshes the configuration on all the Dgraph hosting machines with the modified settings from the `bdd.conf` file. For information on how to do this, see [Refreshing cluster configuration on page 31](#).

5. Restart the Dgraph with the `bdd-admin.sh` script.

Setting the Dgraph internal cache size

The Dgraph cache size should be configured to be large enough to allow the Dgraph to operate smoothly under normal query load.

For enhanced performance, Oracle recommends allocating at least 50% of the node's available RAM to the Dgraph cache. This is a significant amount of memory that you can adjust if needed. For example, if you later find that queries are getting cancelled because there is not enough available memory to process them, you should decrease this amount.

You configure the Dgraph cache size initially by setting the `DGRAPH_CACHE` value in the `bdd.conf` file in the installation directory. The orchestration script uses this value during the BDD installation process.

After the installation, you can adjust the size of the Dgraph cache by gradually changing the `DGRAPH_CACHE` value in the `bdd.conf` file in the `$BDD_HOME/BDD_manager/conf` directory, and use the `bdd-admin refresh-config` script to update the configuration for the entire cluster. For more information, see [Refreshing cluster configuration on page 31](#).

Before you adjust the Dgraph cache, keep the following consideration in mind:

While the Dgraph typically operates within the limits of its configured Dgraph cache size, it is possible for the cache to become over-subscribed for short periods of time. During such periods, the Dgraph may use up to 1.5 times more cache than it has configured. It is important to note that the Dgraph does not expect to routinely reach an increase in its configured cache usage. When the cache size reaches the 1.5 times threshold, the Dgraph starts to more aggressively evict entries that consume its cache, so that the cache memory usage can be reduced to its configured limits.

This means that an occasional spike in Dgraph cache usage should not be the cause of alarm and that you should only consider adjusting the Dgraph cache size after observing Dgraph performance over longer periods of time.

Managing an index merge policy

An index merge policy controls how the Dgraph manages its index files. A balanced index merge policy is used by default, and, in the majority of deployments, you do not need to change it.

[About an index merge policy](#)

[Manually forcing a merge](#)

Linux ulimit settings for merges

About an index merge policy

An index merge policy determines how frequently the Dgraph merges incremental update generations in its index files.

The data layer stores the Dgraph index files as a series of internal files with versions. As a result:

- Old versions can be accessed while new versions are created.
- Old versions are garbage-collected when no longer needed.

A version of the index is stored as a sequence of generation files. A new version appends a new generation file to the sequence. Query latency depends, in part, on the number and size of generation files used to store the index files.

Generation files are combined through a process called *merging*. Merging is a background task that does not affect the Dgraph request processing, but may affect its performance. Because of this, you can set a *merge policy* that dictates the aggressiveness of the merges.

The merge policy has two settings:

- **Balanced:** This policy strikes a balance between low latency and high throughput. This is the default policy that is recommended for the majority of deployments.



Note: Under normal conditions, you do not need to change the default balanced policy.

- **Aggressive:** This policy merges index generations frequently and completely to keep query latency low at the expense of average throughput. Aggressive merge policy may help deployments where query latency is the primary concern.

In addition to setting the merge policy, you can also force a one-time index merge, without changing the overall policy that will be used in all other instances. See [Manually forcing a merge on page 42](#).

You can configure the index merge policy (or force a merge) using one of these two mechanisms:

- The Enterprise Manager plug-in (if you use it for the Big Data Discovery). For forcing a merge or setting the merge policy, see [Merging index updates in the Dgraph on page 135](#).
- Options in the `dgraph-admin` command of the `bdd-admin.sh` script. To force a merge, see [merge on page 47](#). To set the merge policy, see [merge on page 47](#).

In a clustered environment, a request to set the merge policy is automatically routed to the leader Dgraph node.

Merging is not affected by the Dgraph's memory limit.

Manually forcing a merge

Manually forcing a merge is considered a one-time option, because after the merge operation is performed (via a temporary *aggressive* change to the merge policy), the merge policy reverts to its previous setting.

Forcing an index merge is used to perform a complete merge of all generations without making a change to the default merge policy. When you issue this command, it is routed to the leader Dgraph node and the

Dgraph starts a manual merge of its index files. After the merging is performed, the merge policy reverts to its previous setting.

Forcing a merge implies starting a full merge of all generations of index files. When running this command, be aware of the following considerations:

- Memory requirements. Forcing a complete merge utilizes the server's memory. If the amount of memory reaches the amount of RAM that is available, the merge operation will continue to work, but could run substantially slower and have a higher impact on query performance.
- Disk space requirements. Forcing a merge requires provisioning two times the amount of disk space as the current size of the index files. However, if the Dgraph is performing other tasks (such as updates), the merge may require disk space that is three times the size of the index files. If not enough disk space is provisioned, it could be disruptive to force a complete merge. This consideration is especially important for running this command on the Dgraph in a production environment.

Linux ulimit settings for merges

For purposes of generation merging, it is recommended that you set the Linux option `ulimit -v` and `-m` parameters to `unlimited`.

An `unlimited` setting for the `-v` option sets no limit on the maximum amount of virtual memory available to a process, and for the `-m` option sets no limit on the maximum resident set size. Setting these options to `unlimited` can help prevent problems when the Dgraph is merging the generation files.

An example of a merge problem due to insufficient disk space and memory resources is a Dgraph error similar to the following:

```
ERROR 04/03/13 05:24:35.668 UTC (1364966675668) DGRAPH {dgraph} BackgroundMergeTask:
exception thrown: Can't parse generation file, caused by I/O Exception: While mapping file,
caused by mmap failure: Cannot allocate memory
```

In this case, the problem is caused because the Dgraph cannot allocate enough virtual memory for its merging task.

Managing Dgraph core dump files

In the rare case of a Dgraph crash, the Dgraph writes its core dump files on disk. It is recommended to use the `ulimit -c unlimited` setting for the Dgraph core dump files. Non-limited core files contain all Dgraph data that is resident in memory (RSS of the Dgraph process).

When the Dgraph runs on a very large data set, the size of its index files stored in-memory may exceed the size of the physical RAM. If such a Dgraph fails, it may need to write out potentially very large core dump files on disk. The core files are written to the directory from which the Dgraph was started.

To troubleshoot the Dgraph, it is often useful to preserve the entire set of core files written out as a result of such failures. When there is not enough disk space, only a portion of the files is written to disk until this process stops. Since the most valuable troubleshooting information is contained in the last portion of core files, to make these files meaningful for troubleshooting purposes, it is important to provision enough disk space to capture the files in their entirety.

Two situations are possible, depending on your goal:

- You can afford to provision enough disk space.

Large applications may take up the entire amount of available RAM. Because of this, the Dgraph core dump files can also grow large and take up the space equal to the size of the physical RAM on disk plus the size of the server data files in memory. To troubleshoot a Dgraph crash, provision enough disk space to capture the entire set of core files. In this case, the files are saved at the expense of potentially filling up the disk.



Note: If you are not setting `ulimit -c unlimited`, you could be seeing the Dgraph crashes that do not write any core files to disk, since on some Linux installations the default for `ulimit -c` is set to 0.

- You would like to limit the amount of disk space allotted for saving core files.

To prevent filling up the disk, you can limit the size of these files on the operating system level, with the `ulimit -c <size>` command, although this is not recommended. If you set the limit size in this way, the core files cannot be used for debugging, although their presence will confirm that the Dgraph had crashed. In this case, with large Dgraph applications, only a portion of core files is saved on disk. This may limit their usefulness for debugging purposes. To troubleshoot the crash in this case, change this setting to `ulimit -c unlimited`, and reproduce the crash while capturing the entire core file. Similarly, to enable support to troubleshoot the crash, you will need to reproduce the crash while capturing the full core file.

Appointing a new Dgraph leader node

You can use the `appointNewDgraphLeader.sh` script to appoint a new Dgraph leader.

The use case for this script is when there is a long-running ingest in progress in the Dgraph HDFS Agent, and the Dgraph goes down for some reason. Instead of waiting until a new write request comes in, the administrator can just run this script to restart the ingest on another machine. (A file is maintained in HDFS that logs the exact progress of the ingest. The newly-appointed Dgraph HDFS Agent leader reads the file and knows at what point to pick up the ingest).

For example, the Dgraph HDFS Agent on `Dgraph_A` is performing an ingest when the Dgraph crashes (which results in the ingest being suspended). When the script is run, the new leader can be `Dgraph_B`, in which case the ingest is picked up at the point when it was stopped (except that `Dgraph_B` is now performing the ingest instead of `Dgraph_A`). Because there is only one index shared among the Dgraphs, the ingest can be resumed by the new leader.

Note that if the script is run but a new leader has been appointed in the interim, then the script basically reappoints the same leader.

The syntax for running the script is:

```
./appointNewDgraphLeader.sh <dg_address>
```

where `dg_address` is the FQDN (fully-qualified domain name) and port of the Dgraph Gateway server. For example:

```
./appointNewDgraphLeader.sh web009.us.example.com:7003
```

To appoint a new Dgraph leader:

1. Navigate to the `$DGRAPH_HOME/dgraph-hdfs-agent/bin` directory.
2. Run the `appointNewDgraphLeader.sh` script with the FQDN and port of the Dgraph Gateway, as in the example above.

If a new Dgraph leader is successfully appointed, the script returns this message:

```
New Dgraph Leader appointed
```

An unsuccessful operation could return either of these messages:

```
Unable to appoint new Dgraph leader
```

```
Could not reach Dgraph gateway
```

Note that an unsuccessful attempt could be caused by an incorrect address.

About Dgraph statistics

The Dgraph statistics page provides information such as startup time, host, port, and process information, data and log paths, and so on. This information is useful to help to tune your Dgraph and useful for Oracle Support.

The statistics page information is valid as long as the Dgraph is running; it is reset upon a Dgraph restart or by resetting the statistics page.

You can view and reset the Dgraph statistics page using one of these utilities:

- Using the `bdd-admin` script: [stats on page 48](#) and [statsreset on page 48](#)
- Using the Enterprise Manager: [Viewing Dgraph statistics on page 134](#) and [Resetting Dgraph statistics on page 134](#).

Dgraph administrative operations

This section describes how to perform administrative operations for the Dgraph using the `bdd-admin` script with the `dgraph-admin` command.

[About the `dgraph-admin` command](#)

[flush](#)

[merge](#)

[stickymerge](#)

[stats](#)

[statsreset](#)

[logroll](#)

[log-status](#)

[log-enable](#)

[log-disable](#)

About the `dgraph-admin` command

You can perform a number of Dgraph administrative operations by running the `bdd-admin` script with the `dgraph-admin` command.

This command has the following syntax:

```
./bdd-admin.sh dgraph-admin <operation [args]> <hostname>
```

The operations, arguments, and hostnames the `dgraph-admin` command supports are described below.

Operations

The operation argument specifies the action the script will perform. This argument is required.

Operation	Description
<code>flush</code>	Flushes the Dgraph cache.
<code>merge</code>	Merges update generations in the index using the specified merge factor.
<code>stickymerge</code>	Changes the merge policy and forces an index merge.
<code>stats</code>	Returns the Dgraph statistics. This command is only intended for use by Oracle Support.
<code>statsreset</code>	Resets the Dgraph statistics.
<code>logroll</code>	Forces a query logroll. This reinitializes the MDEX query log and archives the old log file.
<code>log-enable</code>	Enables one or more extended logging features.
<code>log-disable</code>	Disables one or more extended logging features.
<code>log-status</code>	Returns the current settings for extended logging features.
<code>--help</code>	Returns the usage information for the <code>dgraph-admin</code> command and its operations.

Arguments

Some operations allow you to specify additional arguments, such as a list of logging features to enable. The arguments each operation supports are discussed in the following sections.

Hostnames

The hostname argument specifies the node(s) the script will run on. This argument is required.

Hostname	Description
<code>--all</code>	Runs the script on all Dgraph nodes in the cluster.
<code><hostname></code>	Runs the script on the specified node. The hostname must match one listed in the <code>DGRAPH_SERVERS</code> property in the <code>bdd.conf</code> file.

dgraph-admin command help

You can obtain usage information for the `dgraph-admin` command and its operations by running it with the `-help` flag:

```
./bdd-admin.sh dgraph-admin --help
```

For information on a specific operation, provide the operation's name before the `--help` flag:

```
./bdd-admin.sh dgraph-admin <operation> --help
```

flush

`./bdd-admin.sh dgraph-admin flush <hostname>` flushes the Dgraph cache.

The `flush` operation clears all entries from the Dgraph's cache and returns the following message:

```
[Manager] flush the Dgraph cache on <hostname>...Success
```

If you are debugging query issues, you can approximate cold-start or post-update performance by cleaning the Dgraph cache before running a request.

merge

`./bdd-admin.sh dgraph-admin merge [merge policy] <hostname>` merges update generations in the index.



Note: In a Dgraph cluster, this command is routed to and processed by the leader node.

The merge policy argument is optional. If you provide one, the script forces the merge with the specified policy. If you omit it, the script forces the merge using the current policy. The following merge policies are supported:

- `balanced`
- `aggressive`

If you specify a merge policy, the script returns the following message:

```
[Manager] Force merge the Dgraph index using <merge policy> policy on <leader node>...Success
```

If you don't specify a merge policy, the script returns the following message:

```
[Manager] Force merge the Dgraph index on <leader node>...Success
```

For more information on merges and merge policies, see [Managing an index merge policy](#).

stickymerge

`./bdd-admin.sh dgraph-admin stickymerge <merge policy> <hostname>` changes the merge policy to the specified policy and forces a merge.



Note: In a Dgraph cluster, this command is routed to and processed by the leader node.

You must specify one of the following merge policies:

- balanced
- aggressive

This command returns the following message:

```
[Manager] Change merge policy to <merge policy> and force merge the Dgraph index on <leader node>...
```

For more information on merges and merge policies, see [Managing an index merge policy](#).

stats

`./bdd-admin.sh dgraph-admin stats <hostname>` returns the Dgraph statistics page.



Note: Dgraph statistics are intended for use by Oracle Support only.

For more information on Dgraph statistics, see [About Dgraph statistics](#).

statsreset

`./bdd-admin.sh dgraph-admin statsreset <hostname>` resets the Dgraph statistics.



Note: Dgraph statistics are intended for use by Oracle Support only.

The `statsreset` operation returns the following message:

```
[Manager] Reset the Dgraph Server Statistics page on <hostname>...Success
```

logroll

`./bdd-admin.sh dgraph-admin logroll <hostname>` forces a query logroll. This reinitializes the query log and archives the old log file.

The `logroll` operation returns the following message:

```
[Manager] Force a Dgraph query log roll on <hostname>...Success
```

log-status

`./bdd-admin.sh dgraph-admin log-status <hostname>` returns a list of all possible logging variables with their statuses indicated by `true` or `false`.

You can view the results of this command in `<hostname>-dgraph-log-stats.html`, using a browser or text editor.

log-enable

`./bdd-admin.sh dgraph-admin log-enable <features> <hostname>` enables the specified extended logging features.

You must provide at least one logging feature to enable. You can use the following command to obtain the list of logging features and the current status of each:

```
./bdd-admin.sh dgraph-admin log-status
```

The `log-enable` command returns a message similar to the following:

```
[Manager] Enable the Dgraph logging <features> on <hostname>...Success
```

log-disable

`./bdd-admin.sh dgraph-admin log-disable <features> <hostname>` disables the specified extended logging features.

You must provide at least one logging feature to disable. You can use the following command to obtain the list of logging features and the current status of each:

```
./bdd-admin.sh dgraph-admin log-status
```

The `log-disable` command returns a message similar to the following:

```
[Manager] Disable the Dgraph logging <features> on <hostname>...Success
```

Dgraph flags

Dgraph flags modify the Dgraph's configuration and behavior.



Important: Dgraph flags are intended for use by Oracle Support only. They are included in this document for completeness.

You can set Dgraph flags by adding them to the `DGRAPH_ADDITIONAL_ARG` property in `bdd.conf` in `$BDD_HOME/BDD_manager/conf` directory, then using the `bdd-admin refresh-config` script to update the cluster configuration. Any flag included in this list will be set each time the Dgraph starts. For more information, see [Refreshing cluster configuration on page 31](#).






Note: Some of the Dgraph flags have the same names as HDFS Agent flags. These must have the same settings as their HDFS Agent counterparts.

Flag	Description
?	Prints the help message and exits. The help message includes usage information for each Dgraph flag.
-v	Enables verbose mode. The Dgraph will print information about each request it receives to either its stdout/stderr file (<code>dgraph.out</code>) or the file set by the <code>--out</code> flag.

Flag	Description
--backlog-timeout	<p>Specifies the maximum number of seconds that a query is allowed to spend waiting in the processing queue before the Dgraph responds with a timeout message.</p> <p>The default is 0 seconds.</p>
--bulk_load_port	<p>Sets the port on which the Dgraph listens for bulk load ingest requests. This must be the same as the port specified for the HDFS Agent --bulk_load_port flag.</p> <p>This flag maps to the DGRAPH_BULKLOAD_PORT property in bdd.conf.</p>
--cluster_identity	<p>Specifies the cluster identity of the Dgraph running on this node. The syntax is:</p> <pre data-bbox="656 779 1451 831">protocol:hostname:dgraph_port:dgraph_bulk_load_port:agent_port</pre> <p>This must be the same as the cluster identity specified for the HDFS Agent --cluster_identity flag.</p>
--coordinator	<p>Specifies the host and port that ZooKeeper is running on. The syntax is:</p> <pre data-bbox="656 1024 1451 1056"><hostname>:<port></pre> <p>This must be the same as the value specified for the HDFS Agent --coordinator flag.</p>
--coordinator_auth	<p>Obtains the ZooKeeper authentication password from stdin.</p>
--coordinator_index	<p>Specifies the index of the Dgraph cluster in the ZooKeeper ensemble. ZooKeeper uses this value to identify the Dgraph cluster. This must be the same as the value specified for the HDFS Agent --coordinator_index flag.</p> <p>This flag maps to the COORDINATOR_INDEX property in bdd.conf.</p>

Flag	Description
<code>--coordinator_session_cache</code>	<p>Specifies the name and (optionally) location of the session cache file used by the leader Dgraph. The leader uses this file to resume its last session with ZooKeeper if it exits abnormally.</p> <p>This file is created when a Dgraph is promoted to leader and deleted when the leader exists normally. If the leader exits abnormally, the file remains on disk so that the leader can resume its last session. Follower Dgraphs don't produce session cache files, and only leaders resume sessions.</p> <p>The default file is <code>\$BDD_HOME/dgraph/clustercache.token</code>. The file location should always be the same to ensure the Dgraph will be able to find it. Additionally, you should avoid modifying the contents of this file.</p>
<code>--export_port</code>	<p>Specifies the port on which the Dgraph listens for requests from the HDFS Agent.</p> <p>This should be the same as the number specified for the HDFS Agent <code>--export_port</code> flag. It should be different from the numbers specified for both the <code>--port</code> and <code>--bulk_load_port</code> flags.</p> <p>This flag maps to the <code>AGENT_EXPORT_PORT</code> property in <code>bdd.conf</code>.</p>
<code>--help</code>	<p>Prints the help message and exits. The help message includes usage information for each Dgraph flag.</p>
<code>--log</code>	<p>Specifies the path to the Dgraph request log file. The default file used is <code>dgraph.reqlog</code>.</p>
<code>--memory-limit</code>	<p>Specifies the maximum amount of memory (in MB) the Dgraph is allowed to use for processing.</p> <p>If you do not use this flag, the memory limit is by default set to 80% of the machine's available RAM.</p> <p>If you specify a limit in MB for this flag, this number is used as the memory consumption limit, for the Dgraph, instead of 80% of the machine's available RAM.</p> <p>If you specify 0 for this flag, this overrides the default of 80% and means there is no limit on the amount of memory the Dgraph can use for processing.</p> <p>For a summary of how Dgraph allocates and utilizes memory, see Memory consumption by the Dgraph on page 39.</p>
<code>--net-timeout</code>	<p>Specifies the maximum amount of time (in seconds) the Dgraph waits for the client to download data from queries across the network. The default value is 30 seconds.</p>

Flag	Description
--out	<p>Specifies a file to which the Dgraph's stdout/stderr will be remapped. If this flag is omitted, the Dgraph uses its default stdout/stderr file, <code>dgraph.out</code>.</p> <p>This file must be different from the one specified by the HDFS Agent's --out flag.</p> <p>This flag maps to the <code>DGRAPH_OUT_FILE</code> property in <code>bdd.conf</code>.</p>
--pidfile	<p>Specifies the file the Dgraph's process ID (PID) will be written to. The default filename is <code>dgraph.pid</code>.</p>
--host	<p>Specifies the name of the Dgraph's host server.</p> <p>This flag maps to the <code>DGRAPH_SERVERS</code> property in <code>bdd.conf</code>.</p>
--port	<p>Specifies the port used by the Dgraph's host server.</p> <p>This flag maps to the <code>DGRAPH_WS_PORT</code> property in <code>bdd.conf</code>.</p>
--leader	<p>Creates a read/write Dgraph leader for the index. This flag is used internally.</p>
--read-only	<p>Sets the index files to read-only. This flag is for internal use only by Oracle Support and should not be used by system administrators of Big Data Discovery 1.0.</p> <p>When this flag is set, the Dgraph can only perform read-only operations. Any operations that attempt to write to the index files are rejected and return an HTTP status code 403.</p>
--search_char_limit	<p>Specifies the maximum number of characters that a text search term can contain. The default value is 132.</p>
--search_max	<p>Specifies the maximum number of terms that a text search query can contain. The default value is 10.</p>
--snip_cutoff	<p>Specifies the maximum number of words in an attribute that the Dgraph will evaluate to identify a snippet. If a match is not found within the specified number of words, the Dgraph won't return a snippet, even if a match occurs later in the attribute value.</p> <p>The default value is 500.</p>
--snip_disable	<p>Globally disables snipping.</p>

Flag	Description
--sslcafile	 Note: This flag is not used in Oracle Big Data Discovery 1.0. Specifies the path to the SSL Certificate Authority file that the Dgraph will use to authenticate SSL communications with other components.
--sslcertfile	 Note: This flag is not used in Oracle Big Data Discovery 1.0. Specifies the path of the SSL certificate file that the Dgraph will present to clients for SSL communications.
--stat-brel	 Note: This flag is deprecated and not used in Oracle Big Data Discovery 1.0. Creates dynamic record attributes that indicate the relevance rank assigned to full-text search result records.
--syslog	Directs all output to syslog.
--threads	Specifies the number of threads the Dgraph will use to process queries and execute internal maintenance tasks. The value you provide must be a positive integer (2 or greater). The recommended number of threads for machines running only the Dgraph is the number of CPU cores the machine has. For machines co-hosting the Dgraph with other Big Data Discovery components, the recommended number of threads is the number of CPU cores the machine has minus two. This flag maps to the <code>DGRAPH_THREADS</code> property in <code>bdd.conf</code> .
--validate_data	Validates that all indexed data loads and then exits.
--version	Prints version information and then exits. The version information includes the Oracle Big Data Discovery version number and the internal Dgraph identifier.
--wildcard_max	Specifies the maximum number of terms that can match a wildcard term in a wildcard query that contains punctuation, such as <code>ab*c.def*</code> . The default is 100.

Dgraph HDFS Agent flags

This topic describes the flags used by the Dgraph HDFS Agent.

The Dgraph HDFS Agent requires several flags, which are described in the following table. Note that some flags have the same name as their Dgraph flag counterpart, and (except for `--out`) must have the same settings.

The `startDgraphHDFSAgent.sh` script can use the following flags:

Dgraph HDFS Agent flag	Description
<code>--agent_port</code>	Sets the port on which the Dgraph HDFS Agent is listening for HTTP requests. Note that there is no Dgraph version of this flag.
<code>--export_port</code>	Sets the port on which the Dgraph HDFS Agent is listening for requests from the Dgraph. This port number must be the same as specified for the Dgraph <code>--export_port</code> flag.
<code>--port</code>	Specifies the port on which the Dgraph is listening for HTTP requests. This port number must be the same as specified for the Dgraph <code>--port</code> flag.
<code>--bulk_load_port</code>	Sets the port on which the Dgraph HDFS Agent is listening for bulk load ingest requests. This port number must be the same as specified for the Dgraph <code>--bulk_load_port</code> flag.
<code>--cluster_identity</code>	Specifies the cluster identity of the Dgraph running on this node. The syntax is: <pre>protocol:hostname:dgraph_port:dgraph_bulk_load_port:agent_port</pre> This cluster identity must be the same as specified for the Dgraph <code>--cluster_identity</code> flag.
<code>--coordinator</code>	Specifies the host and port on which Zookeeper is running. The syntax is: <pre>host:port</pre> (with a semicolon separating the host name and port). This <code>host:port</code> must be the same as specified for the Dgraph <code>--coordinator</code> flag.
<code>--coordinator_index</code>	Specifies the index of the cluster in the Zookeeper ensemble. This index must be the same as specified for the Dgraph <code>--coordinator_index</code> flag.
<code>--out</code>	Specifies the file name and path of the Dgraph HDFS Agent's stdout/stderr log file. The log name must be different from that specified with the Dgraph <code>--out</code> flag.

Hadoop configuration files

The `core-site.xml` and `hdfs-site.xml` files are used to configure a Hadoop cluster, especially the one machine in the cluster that is designated as the NameNode. The NameNode contains the HDFS file system from which the Dgraph HDFS Agent will read ingest files and write export files.

At start-up, the Dgraph HDFS Agent reads in the `core-site.xml` and `hdfs-site.xml` files so it can determine the location of the NameNode.

Startup example

The following is an example of using the `startDgraphHDFSAgent.sh` to start the Dgraph HDFS Agent:

```
./startDgraphHDFSAgent.sh --agent_port 7102 --export_port 7101 --port 5555
--bulk_load_port 5556 --coordinator web04.example.com:2181 --coordinator_index cluster1
--cluster_identity http:web04.example.com:5555:5556:7102 --out /tmp/agent.log
```



Chapter 5

The Dgraph Gateway

This section describes the Dgraph Gateway role in the Big Data Discovery cluster deployment. It also discusses its configuration file, and tells you how to start and stop the Dgraph Gateway through the Administration Console of the WebLogic Server.

[About the Dgraph Gateway](#)

[Dgraph Gateway configuration file](#)

[Starting Dgraph Gateway](#)

[Stopping Dgraph Gateway](#)

About the Dgraph Gateway

Together with Studio, the Dgraph Gateway is a Java-based application that is co-hosted in the same WebLogic Server instance.

The Dgraph Gateway provides:

- Routing of requests to the Dgraph nodes in the BDD cluster
- Caching, business logic, and handling of cluster services for the Dgraph nodes.

Within the Big Data Discovery cluster deployment, you can have one or more WebLogic Server Managed nodes each of which run Studio and Dgraph Gateway. Once the Dgraph Gateway is deployed, you use the WebLogic Server's Administration Console to manage it.

Dgraph Gateway configuration file

A configuration file sets global parameters for the Dgraph Gateway, such as the default locations of files and directories.

The name of the configuration file is `EndecaServer.properties` and it is located in the `config` directory of your BDD WebLogic Server domain. For example, assuming that `obdd` is the name of your WebLogic Server domain for the Dgraph Gateway, the default location is:

```
$MW_HOME/user_projects/domains/obdd/config/EndecaServer.properties
```

The default values in the file are set when the domain is created at installation time.

Most of these parameters are used by the Dgraph Gateway application and should not be modified. If you do need to modify some of them, stop the Dgraph Gateway on the machine on which you are modifying the parameter and restart it. If you have multiple nodes in the BDD cluster deployment that run WebLogic Server hosting Studio and the Dgraph Gateway, make the changes to the `EndecaServer.properties` on all WebLogic Server machines and then restart the Dgraph Gateway instances. BDD relies on this file being the same on all WebLogic Server Managed nodes in the BDD cluster deployment.

Dgraph Gateway settings


The following configuration settings are specific to Dgraph Gateway operations:

Dgraph Gateway parameter	Description
endeca-session-id-key	Specifies name of the key used to maintain session affinity. X-Endeca-Session-ID is the default value.
endeca-session-id-type	Specifies the method used to establish session affinity. The HEADER is the default value.
endeca-ds-pin-timeout-min endeca-ds-pin-timeout-max endeca-ds-pin-timeout-default	Deprecated and not used by the Dgraph Gateway.

ZooKeeper settings

For some of its functions, the Dgraph Gateway relies on the ZooKeeper package found in the CDH installation.

The following configuration settings are specific to ZooKeeper:

ZooKeeper parameter	Description
endeca-cluster-identifier	Specifies a string identifier for a BDD cluster. This property identifies the cluster. The default is <code>cluster1</code> .
zookeeper-servers	Specifies a comma-separated list of <code>host:port</code> pairs to describe each ZooKeeper server in a ZooKeeper ensemble. The <code>host</code> represents a server running ZooKeeper. The corresponding <code>port</code> is the port on which ZooKeeper clients connect to that server. The default <code>host</code> value is the name of the Admin Server. The default <code>port</code> value is 2181. If a single server runs ZooKeeper, specify the server name, such as <code>zookeeper-servers=web009.us.example.com:2181</code> . If multiple servers run ZooKeeper, specify comma-separated <code>host:port</code> names of all servers that are part of the Zookeeper ensemble.  Note: ZooKeeper servers are those CDH nodes in the Big Data Discovery cluster deployment that run ZooKeeper instances. For example, you can have five nodes running CDH, out of which three nodes have ZooKeeper running. For a diagram of a BDD cluster deployment, see Diagram of a Big Data Discovery Cluster on page 21 .

Starting Dgraph Gateway

When you start the WebLogic Server in which the Dgraph Gateway application is deployed, it automatically starts the Dgraph Gateway.

If the application was running when WebLogic Server was shut down, the Dgraph Gateway automatically re-starts as part of the WebLogic Server start-up procedure. Additionally, you can manually start the Dgraph Gateway from the WebLogic Server Administration Console.

To start a stopped Dgraph Gateway:

1. Make sure that the Administration Server for the Big Data Discovery is running.
2. From your browser, access the Administration Server console using this syntax:

```
http://admin_server_host:admin_server_port/console
```

For example:

```
http://web007:7001/console
```

3. At the Administration Console login screen, log in with the administrator user name and password.
4. In the **Domain Structure** pane, click **Deployments**.
5. In the **Deployments** table, check the **oracle.endecaserver** Web application. Its State should be "Prepared" and its Health should be "OK".
6. In the Deployments table, click **Start>Servicing all requests** (which makes the application immediately available to all WebLogic Server clients).

You can also choose the **Servicing only administration requests** option, which makes the application available in Administration Mode only.
7. In the **Stop Application Assistant**, click **Yes**.

As a result, the Dgraph Gateway is started and its State now changes to "Active".

Stopping Dgraph Gateway

You can manually stop the Dgraph Gateway from the WebLogic Server Administration Console.

Note that it is not necessary to stop Dgraph Gateway in order to shut down WebLogic Server; in this case, WebLogic Server will stop Dgraph Gateway as part of its shut-down procedure.

To stop the Dgraph Gateway:

1. Make sure that the Administration Server is running.
2. From your browser, access the Administration Server console using this syntax:

```
http://admin_server_host:admin_server_port/console
```

For example:

```
http://web007:7001/console
```

3. At the Administration Console login screen, log in with the administrator user name and password.

4. In the Domain Structure pane, click **Deployments**.
5. In the Deployments table, check the **oracle.endecaserver** Java application. Its State should be "Active" and its Health should be "OK", as in this abbreviated example:
6. In the **Deployments** table, click **Stop**, and select one of the stop options:
 - **When work completes:** Specifies that WebLogic Server waits for the Dgraph Gateway to finish its work and for all currently connected users to disconnect.
 - **Force Stop Now:** Specifies that WebLogic Server stops the Dgraph Gateway immediately, regardless of the work that is being performed and the users that are connected.
 - **Stop, but continue servicing administration requests:** Specifies that WebLogic Server stops the Dgraph Gateway once all its work has finished, but then puts the application in Administration Mode so it can be accessed for administrative purposes.
7. In the Stop Application Assistant, click **Yes**.

As a result, the Dgraph Gateway is stopped and its State now changes to "Prepared".



Note: If the Dgraph Gateway is in a "Prepared", (that is, stopped), state when you shut down WebLogic Server, then the application is not automatically restarted when you start WebLogic Server. In this case, you must manually start Dgraph Gateway.



Chapter 6

Administering the Dgraph with the Dgraph Gateway Command Utility

This section describes the Dgraph Gateway commands (`endeca-cmd`) used for Dgraph nodes.

[About the Dgraph Gateway Command Utility](#)

[Global options for host, port, and context root](#)

[Allocating a bulk load port](#)

[Returning version information](#)

[Listing Dgraph nodes](#)

[Returning Dgraph session information](#)

[Warming the Dgraph cache](#)

About the Dgraph Gateway Command Utility

The Dgraph Gateway has a command-line interface that lists Dgraph nodes, allocates bulk load port, provides version information, and performs cache warming operations for the Dgraphs.

The Dgraph Gateway Command Utility resides by default in the `$BDD_HOME/server/endeca-cmd` directory. For example, if `BDD1.0` is the name of your BDD install directory, then the path to the directory might be:

```
/localdisk/Oracle/Middleware/BDD1.0/server/endeca-cmd
```

The directory contains a script (named `endeca-cmd`) with which you can run the commands.

The `endeca-cmd` utility requires a Java run-time environment (JRE) to run. Therefore, verify that you have included the bin directory of the installed JDK at the beginning of the `PATH` variable definition on your system. Alternatively, check that you have correctly set the `JAVA_HOME` environment variable.

Commands

The `endeca-cmd` script allows you to run the following Dgraph Gateway commands.

Option	Description
<code>allocate-bulk-load-port</code>	Returns a host name for the leader node and the port used for Bulk Load Interface.
<code>dump-session</code>	Returns session information from a Dgraph for a specified session Id.
<code>list-compute-nodes</code>	Returns a list of running Dgraph nodes in a cluster.

Option	Description
version	Lists the version of the Dgraph Gateway and the version of the Dgraph (if the Dgraphs are currently running).
warm-cache	Warms the Dgraph cache without requiring a custom warm-up script.

Syntax

The syntax for running the `endeca-cmd` script is:

```
endeca-cmd <operation> [operation options] [global options]
```

Getting online help

The `--help` option provides usage help for the Dgraph Gateway commands. The syntax for obtaining general help is:

```
endeca-cmd --help
```

The syntax for obtaining help on a specific commands is:

```
endeca-cmd <operation> --help
```

This example displays usage help for the `list-compute-nodes` commands:

```
endeca-cmd list-compute-nodes --help
```

Global options for host, port, and context root

The command utility has several global options that allow you to specify the host, port, and context root of the Dgraph Gateway.

The global options are:

- `--host`
- `--port`
- `--root`
- `--help`

Do not forget to specify global options with `endeca-cmd`. If you do not specify them, `endeca-cmd` assumes that the defaults are used for the Dgraph Gateway (such as the default port and host). For example, assume you have configured the Dgraph Gateway application in WebLogic domain to use a port that is different from the default port. In this case, in order for the `endeca-cmd` utility to find the correct port, you should list it explicitly, as one of the global options. For example, this operation returns a list of the Dgraphs available to the Dgraph Gateway running on port 9001:

```
endeca-cmd list-compute-nodes --port 9001
```

--host option

You use the `--host` option when you want to run a command on a Dgraph Gateway that is running on a remote machine. The `--host` argument can be either the full name of the remote machine or its IP address.

The following example illustrates the `--host` global option:

```
endeca-cmd list-compute-nodes --host web7.example.com
```

The command tells the Dgraph Gateway running on **web7.example.com** (and listening on its default port) to return a list of the Dgraphs compute nodes.

--port option

7001 is the default HTTP port in the WebLogic Server on which the Dgraph Gateway application is listening.



Important: HTTP is used for communication of the Dgraph Gateway with other components within Big Data Discovery. Therefore, the node hosting WebLogic Server for the Dgraph Gateway (it is the same node that hosts Studio within WebLogic Server) must be deployed behind the site's firewall.

The `--port` option is used whenever the Dgraph Gateway is not running on its default port, regardless of whether the Dgraph Gateway is running locally or on a remote machine. If you do not specify `--port`, the default port is used for the command.

The following example illustrates both the host and port global options:

```
endeca-cmd list-compute-nodes --host web7.example.com --port 7003
```

The command tells the Dgraph Gateway running on the **web7.example.com** remote machine (and listening on a non-default port 9090) to return the list of Dgraph compute nodes.

--root option

The Dgraph Gateway application uses **/endeca-server** as the default name of its context root when running in WebLogic Server. The `--root` option is used to specify this context-root name. If you do not specify `--port`, the default **/endeca-server** context root is used for the command.

Allocating a bulk load port

The `allocate-bulk-load-port` operation returns a host name for the leader node and the port used for the internally-used Bulk Load Interface.

The syntax for this command is:

```
endeca-cmd allocate-bulk-load-port [global-options]
```

This is a read-write operation — if the current leader node is available, the operation verifies the current Dgraph leader node and reports it along with the port used for Bulk Load. If the current leader node is not available, it appoints a new leader node and a new bulk load port and reports them.

To allocate a bulk load port:

1. From the command line, navigate to the `endeca-cmd` directory.
2. Run the `allocate-bulk-load-port` command.

Example

```
endeca-cmd allocate-bulk-load-port --port 7003
Bulk load host: web009.us.example.com
Bulk load port: 7019
```

Returning version information

The `version` command lists the version of the Dgraph Gateway and the version of the Dgraph nodes (if the Dgraph nodes are currently running).

The syntax for this command is:

```
endeca-cmd version [global-options]
```

To return version information:

1. From the command line, navigate to the `endeca-cmd` directory.
2. Run the `version` command.

Example

```
endeca-cmd version --port 7003
Oracle Endeca Server 1.0.0.897158
```

Listing Dgraph nodes

The `list-compute-nodes` operation returns a list of running Dgraph nodes in a cluster. This includes both leader and follower nodes. The operation does not list Dgraphs that are stopped.

The syntax for this command is:

```
endeca-cmd list-compute-nodes [global-options]
```

To list Dgraph nodes:

1. From the command line, change to the directory where `endeca-cmd` is installed.
2. Run the `list-compute-nodes` operation.

Example

```
endeca-cmd list-compute-nodes --port 7003
HTTP:web009.us.example.com:7010 Leader
```

Returning Dgraph session information

The `dump-session` operation returns session information from a Dgraph for a specified session ID. (Dgraph Gateway tracks which Dgraph instance is processing a request for a particular session.)

The syntax for this command is:

```
endeca-cmd dump-session [global-options] operation options
```

where operation options are the following:

- `--latest <argument>` is the number of most recent sessions to return.
- `--session-id <argument>` is the session ID to return.

For this command to work, you must specify one of the options and an argument for it. There is no default behavior for this command.

To return Dgraph session information:

1. From the command line, change to the directory where `endeca-cmd` is installed.
2. Run the `dump-session` operation with additional options as desired.

Examples

Example with `--latest <argument>`:

```
endeca-cmd dump-session --latest 30 --port 7003
Dump 30 of most recently queried sessions information.
Session id: faked-session-2. DGraph node: web009:7010. Time of first query: 2014-27-21 01:27:56.
Time of the last query: 2014-27-21 01:27:56. Request count: 1
Session id: faked-session-1. DGraph node: web009:7010. Time of first query: 2014-27-21 01:27:44.
Time of the last query: 2014-27-21 01:27:44. Request count: 1
Session id: faked-session-0. DGraph node: web009:7010. Time of first query: 2014-27-21 01:27:04.
Time of the last query: 2014-27-21 01:27:04. Request count: 1
```

Example with `--session-id <argument>`:

```
endeca-cmd dump-session --session-id 1234 --port 7003
Dump session information with the given session id: 1234
Session id: 1234. DGraph node: web009:7010. Time of first query: 2014-27-21 01:27:04.
Time of the last query: 2014-27-21 01:27:04. Request count: 1
```

Warming the Dgraph cache

The `warm-cache` command warms each Dgraph cache for all Dgraph instances in a cluster.

The command takes into account usage patterns of the Dgraphs and replays a set of previous queries for a specified period of time against each Dgraph. That replay warms the cache, allows a Dgraph to reuse cached results across subsequent user queries, and helps reduce the user-observable latencies in query processing.

You must explicitly issue the cache warming request. It does not run automatically. The only parameter for the command is the time limit for which the cache warming runs.

A successful invocation of `warm-cache` returns immediately with an empty response and starts the cache warming job in the background. Once the time limit is reached, the cache warming stops. If during this time you issue any other requests to the Dgraph, they take priority over cache warming.

Note that the existing cache may also contain queries that won't run after the index had changed, for example, because the records schema had changed after an update. The cache warming command ignores errors from such queries (if they are selected for replay), and proceeds to run other queries in its list. The actual queries replayed by the cache warming operation do not appear in the request log.

The syntax for this command is:

```
endeca-cmd warm-cache [--time-limit-sec <sec>] [global-options]
```


The `--time-limit-sec` parameter is optional. It specifies a time limit to replay previous queries. If you do not specify the timeout, the default value of 1800 seconds (30 minutes) is used.

To warm the Dgraph cache:

1. From the command line, navigate to the `endeca-cmd` directory.
2. Run the `warm-cache` command.

Example

```
endeca-cmd warm-cache --port 7003  
Warmed the cache on DGraph node: we009.us.example.com:7010.
```

Part IV

Administering Studio



Configuring Studio Settings

The **Studio Settings** page, on the **Control Panel**, configures many general settings for the Studio application.

[Studio settings list](#)

[Changing the Studio setting values](#)

Studio settings list

Studio settings include configuration options for timeouts, default values, and the connection to Oracle MapViewer, for the **Map** and **Thematic Map** components.

The Studio settings are:

Setting	Description
<code>df.bddSecurityManager</code>	The fully-qualified class name to use for the BDD Security Manager. Leave empty to disable the Security Manager.
<code>df.dataSourceDirectory</code>	The directory used to store keystore and certificate files for secured data.
<code>df.defaultChartColorPalette</code>	<p>The default set of colors to use to display charts in the Chart component.</p> <p>The value is a comma-separated list of between 16 and 30 hex color values.</p> <p>For reference, the default value is:</p> <pre>#54BDC1, #474C61, #FFA600, #90A9B3, #E6C400, #B64CBF, #97DA50, #FF4B37, #A1A2A0, #563864, #66E7B7, #9A1919, #F2DB76, #24D4E8, #265C49, #A2D9DB, #6D7069, #D36A02, #8398F6, #8A8B34, #E38FEB, #46B43B, #D18800, #B8C488, #B1A2CA, #28A173, #854A06, #3D55C3, #C3D600, #326C84</pre>
<code>df.defaultCurrencyList</code>	A comma-separated list of currency symbols to add to the ones currently available.
<code>df.exportBatchSize</code>	<p>When exporting a large number of records, Big Data Discovery splits the records into batches.</p> <p>This setting determines the number of records in each batch.</p> <p>The default value is 2000.</p>

Setting	Description
<code>df.helpLink</code>	Used to configure the path to the documentation for this release. Used for links to specific information in the documentation.
<code>df.mapLocation</code>	<p>The URL for the Oracle MapViewer eLocation service.</p> <p>The eLocation service is used for the text location search on the Map component, to convert the location name entered by the user to latitude and longitude.</p> <p>By default, this is the URL of the global eLocation service.</p> <p>If you are using your own internal instance, and do not have Internet access, then set this setting to "None", to indicate that the eLocation service is not available. If the setting is "None", Big Data Discovery disables the text location search.</p> <p>If this setting is not "None", and Big Data Discovery is unable to connect to the specified URL, then Big Data Discovery disables the text location search.</p> <p>Big Data Discovery then continues to check the connection each time the page is refreshed. When the service becomes available, Big Data Discovery enables the text location search.</p>
<code>df.mapTileLayer</code>	<p>The name of the MapViewer Tile Layer.</p> <p>By default, this is the name of the public instance.</p> <p>If you are using your own internal instance, then you must update this setting to use the name you assigned to the Tile Layer.</p>
<code>df.mapViewer</code>	<p>The URL of the MapViewer instance.</p> <p>By default, this is the URL of the public instance of MapViewer.</p> <p>If you are using your own internal instance of MapViewer, then you must update this setting to connect to your MapViewer instance.</p>
<code>df.maxExportRecords</code>	<p>The maximum allowable number of records that can be exported from a component.</p> <p>The default value is 1000000.</p>
<code>df.stringTruncationLimit</code>	<p>The maximum number of characters to display for a string value.</p> <p>This value may be overridden when configuring the display of a string value in an individual component.</p> <p>The default value is 10000.</p>

Setting	Description
<code>df.versionPinningTimeout</code>	<p>The time (in milliseconds) for which to pin the version of the data.</p> <p>This is used to help ensure that when users export data from a project, the same version of the data is used for the entire export.</p> <p>The default value is -1, which indicates to use the Dgraph Gateway setting. Dgraph Gateway uses a default value of 120000 milliseconds.</p>


Changing the Studio setting values

To set the values of Studio settings, you can either use the fields on the **Studio Settings** page, or add the values to `portal-ext.properties`. If you configure a setting in `portal-ext.properties`, then the field on the **Framework Settings** page is locked.

Configuring settings in `portal-ext.properties` makes it easier to migrate settings across different environments. For example, after testing the settings in a development system, you can simply copy the properties file to the production system, instead of having to reset the production settings manually from the **Control Panel**.

To change the Studio setting values:

1. To configure framework settings from the **Studio Settings** page:
 - (a) In the Studio header, click **Control Panel > Studio Settings**
 - (b) For each setting you want to update, provide a new value in the setting configuration field.

 **Note:** Take care when modifying these settings, as incorrect values can cause problems with your Studio instance.

If the setting is configured in `portal-ext.properties`, then you cannot change the setting from this page. You must set it in the file.

- (c) Click **Update Settings**.
- (d) To apply the changes, restart Big Data Discovery.
2. To add a setting to `portal-ext.properties`:
 - (a) Stop the server.
 - (b) Add the setting to `portal-ext.properties`.

In the file, the format for adding a setting is:

```
<settingname>=<value>
```

Where:

- `<settingname>` is the name of the setting from the **Framework Settings** page.
- `<value>` is the value of the setting.

For example, to set the maximum number of records to export, the entry would be:

```
df.maxExportRecords=50000
```

(c) Restart Big Data Discovery.

On the **Framework Settings** page, the setting is now read only.



Chapter 8

Configuring Data Processing Settings

In order to upload files (Excel and CSV) and perform other data processing tasks, you must configure the **Data Processing Settings** on Studio's Control Panel.

[List of Data Processing Settings](#)

[Changing the data processing settings](#)

List of Data Processing Settings

The settings listed in the table below must be set correctly in order to perform data processing tasks.

Many of the default values for these setting are populated based the values specified in `bdd.conf` during the installation process.

In general, the settings below should match the Data Processing CLI configuration properties which are contained in the script itself. Parameters that must be the same are noted as such in the table below. For information about the Data Processing CLI configuration properties, see the *Data Processing Guide*.

Hadoop Setting	Description
<code>bdd.clusterOltHome</code>	Specifies the OLT home directory in the BDD cluster. The BDD installer detects this value and populates the setting. Must match the <code>edpJarDir</code> setting in the Data Processing CLI.
<code>bdd.databaseName</code>	Specifies the name of the Hive database that stores the source data for Studio data sets.
<code>bdd.edpDataDir</code>	Specifies the directory where data processing operations (ingest and transformations) are performed. The default value is <code>/user/bdd</code> . Must match the <code>edpDataDir</code> setting in the Data Processing CLI.
<code>bdd.edpJarDir</code>	Specifies the directory that contains the contents of the <code>edp_cluster_*.zip</code> file on each worker node. Must match the <code>edpJarDir</code> setting in the Data Processing CLI.

Hadoop Setting	Description
<code>bdd.edpOozieJobsDir</code>	Specifies the HDFS directory to store Oozie job files. The BDD installer detects this value and populates the setting. Must match the <code>oozieJobsDir</code> setting in the Data Processing CLI.
<code>bdd.enableEnrichments</code>	Specifies whether to run data enrichments during the sampling phase of data processing. This setting controls the Language Detection, Term Extraction, Geocoding Address, Geocoding IP, and Reverse Geotagger modules. A value of <code>true</code> runs all the data enrichment modules and <code>false</code> does not run them. You cannot enable an individual enrichment. The default value is <code>true</code> .
<code>bdd.hadoopClusterHostname</code>	Specifies the hostname where the Hadoop services are installed (NameNode, Oozie, JobTracker, Spark, etc.). Must match the <code>oozieHost</code> , <code>hiveServerHost</code> , <code>sparkMasterHost</code> settings in the Data Processing CLI.
<code>bdd.hdfsEdpLibPath</code>	Specifies the HDFS directory to store Data Processing .jar files. The BDD installer detects this value and populates the setting. Must match the <code>hdfsEdpLibPath</code> setting in the Data Processing CLI.
<code>bdd.hiveJdbcUrlPort</code>	Specifies the port of the JDBC client for the Hive server. The Hive JDBC driver allows you to access Hive from a Java program that you write, or from a Business Intelligence or similar application that uses JDBC to communicate with database products.
<code>bdd.hiveMetastoreServerPort</code>	Specifies the port of the Metastore server. Must match the <code>hiveServerPort</code> setting in the Data Processing CLI.
<code>bdd.hueHttpPort</code>	Specifies the port number for the Hue process.
<code>bdd.javaPath</code>	Specifies the path to Java binaries within the Java installation for each host in the cluster. Java must be installed in the same location on each host. The default value for the Java binaries is in <code>/usr/java/jdk1.7.0_67/bin/java</code> . Must match the <code>oozieWorkerJavaExecPath</code> setting in the Data Processing CLI.
<code>bdd.jobTrackerPort</code>	Specifies the port of the JobTracker.

Hadoop Setting	Description
<code>bdd.kryoBufferSize</code>	Specifies the amount of buffer space allocated to Kryo. If you encounter Kryo-related exceptions, you may need to increase this value. The default value is 1024 MB.
<code>bdd.kryoMode</code>	Specifies a Boolean value to enable or disable Kryo mode. Kryo mode provides an alternative way to serialize and move data among Spark worker nodes. A value of <code>true</code> enables Kryo mode and <code>false</code> uses Java serialization. Kryo mode is generally faster for data processing but may cause exceptions in situations that are hard to anticipate. The default value is <code>false</code> .
<code>bdd.language</code>	Specifies either an ISO-639 language code (<code>EN</code> , <code>DE</code> , <code>FR</code> , and so on) or a value of <code>unknown</code> to control whether Oracle Language Technology libraries are invoked during data processing and indexing. A language code requires more processing time but gives better processing and indexing results by using OLT libraries for the specified language. If you specify <code>unknown</code> , the processing time is faster but the processing and indexing results are more generic and OLT is not invoked. The default value is <code>unknown</code> .
<code>bdd.maxRecordsToProcess</code>	<p>Specifies the maximum number of records that are processed to become the sample size of a data set in the Catalog. This is a global setting controls the sample size for all Excel and CSV files uploaded using Studio.</p> <p>For example, you if upload a file that has 5,000,000 rows, you could restrict the total number of sampled records to 1,000,000.</p> <p>The default value is 1,000,000. (This value is approximate. After data processing, the actual sample size may be slightly more or slightly less than this value.)</p>
<code>bdd.nameNodePort</code>	Specifies the port number of the HDFS NameNode. The BDD installer detects this value and populates the setting. If the HDFS NameNode host is not the same as <code>bdd.hadoopClusterHostname</code> , then specify a <code>host:port</code> value for this setting.
<code>bdd.oozieServerPort</code>	<p>Specifies the port of the Oozie server. The BDD installer detects this value and populates the setting. If the Oozie host is not the same as <code>bdd.hadoopClusterHostname</code>, then specify a <code>host:port</code> value for this setting.</p> <p>Must match the <code>ooziePort</code> setting in the Data Processing CLI.</p>

Hadoop Setting	Description
<code>bdd.sandbox</code>	Specifies the HDFS directory in which to store the avro files created when users export data from Big Data Discovery. The default value is <code>/user/bdd</code> .
<code>bdd.sparkExecutorMemorySetting</code>	<p>Specifies the amount of memory allocated to the Spark executor. The default value is 48 GB.</p> <p>This setting must be less than or equal to Spark's Total Java Heap Sizes of Worker's Executors in Bytes (<code>executor_total_max_heapsize</code>) property in Cloudera Manager. You can access this property in Cloudera Manager by selecting Clusters > Spark (Standalone), then clicking the Configuration tab. This property is in the Worker Default Group category (using the classic view).</p>
<code>bdd.sparkServerPort</code>	<p>Specifies the port of the Spark server. The BDD installer detects this value and populates the setting. If the Spark server is not the same as <code>bdd.hadoopClusterHostname</code>, then specify a <code>host:port</code> value for this setting.</p> <p>Must match the <code>sparkMasterPort</code> setting in the Data Processing CLI.</p>

Changing the data processing settings

You configure the settings on the **Data Processing Settings** page on the **Control Panel**.

To change the Hadoop setting values:

1. Log in to Studio as an administrator.
2. In the **Control Panel** menu, click **Data Processing Settings**.
3. For each setting, update the value as necessary.
4. Click **Update Settings**.

The changes are applied immediately.



Chapter 9

Viewing Summary Reports of Project Usage

Big Data Discovery provides basic reports to allow you to track project usage.

[About the project usage logs](#)

[About the System Usage page](#)

[Using the System Usage page](#)

About the project usage logs

Big Data Discovery stores project creation and usage information in its database.

When entries are added to the usage logs

Entries are added when users:

- Log in to Big Data Discovery
- Navigate to a project
- Navigate to a different page in a project
- Create a data set from the **Data Source Library**
- Create a project

When entries are deleted from the usage logs

By default, whenever you start Big Data Discovery, all entries 90 days old or older are deleted from the usage logs.

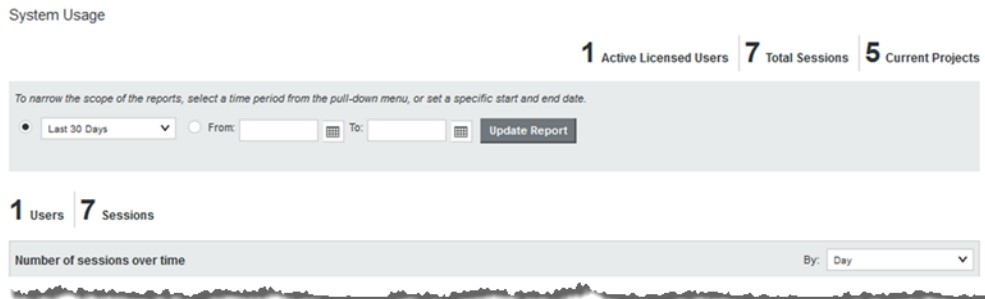
To change the age of the entries to delete, add the following setting to `portal-ext.properties`:

```
studio.startup.log.cleanup.age=entryAgeInDays
```

In addition to the age-based deletions, Big Data Discovery also deletes entries associated with data sets and projects that have been deleted.

About the System Usage page

The **System Usage** page of the **Control Panel** provides access to summary information on project usage logs.



The page is divided into the following sections:

Section	Description
Summary totals	At the top right of the page are the total number of: <ul style="list-style-type: none"> • Users in the system • Sessions that have occurred • Projects
Date range fields	Contains fields to set the range of dates for which to display report data.
Current number of users and sessions	Lists the number of users that were logged in and the number of sessions for the date range that you specify.
Number of sessions over time	Report showing the number of sessions that have been active for the date range that you specify Includes a drop-down list to set the date unit to use for the chart.
User Activity	Report that initially shows the top 10 number of sessions per user for the selected date range across all projects. You can click on any bars in this chart to drill down into the reporting data. At the top of the report are drop-down lists to select: <ul style="list-style-type: none"> • A specific user, or all users • A specific project, or all projects • Whether to display the top or bottom values (most or least sessions) • The number of values to display

Section	Description
Project Usage	<p>Report that initially shows the top 10 number of sessions per project for the selected date range across all projects. You can click on any bars in this chart to drill down into the reporting data.</p> <p>At the top of the report are drop-down lists to select:</p> <ul style="list-style-type: none"> • A specific project, or all projects • Whether to display the top or bottom values (most or least sessions) • The number of values to display
Data	<p>Contains two reports:</p> <ul style="list-style-type: none"> • The first report shows the number of times each data source in the Data Source Library was used to create a data set during the selected date range. • The report chart shows the number of projects that were created from pre-built Dgraph Gateways during the selected date range. <p>Each report includes drop-downs to select:</p> <ul style="list-style-type: none"> • Whether to display the top or bottom values (most or least uses or projects) • The number of values to display
System	<p>Contains a pie chart that shows the relative number of sessions by browser type and version for the selected date range.</p>

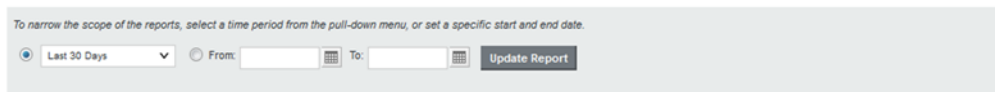
Using the System Usage page

On the **System Usage** page, you use the fields at the top to set the date range for the report data. You can also change the displayed data on individual reports.

To use the **System Usage** page:

1. To set the date range for the displayed data on all of the reports, you can either set a time frame from the current day, or a specific range of dates.

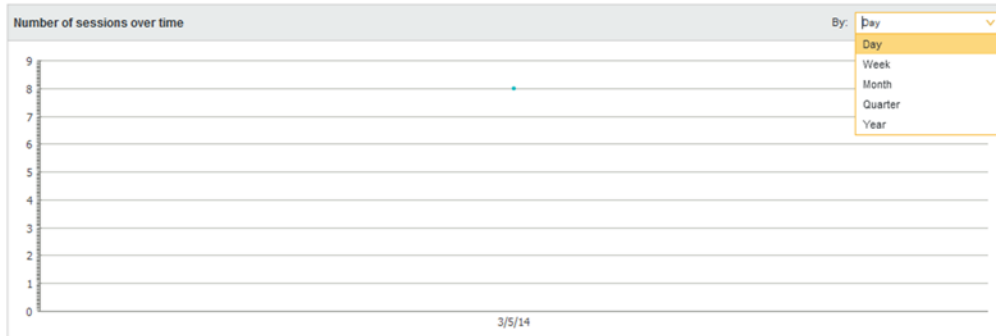
By default, the page is set to display data from the last 30 days.



- (a) To select a different time frame, from the drop-down list, select the time frame to use.
- (b) To select a specific range of dates, click the other radio button, then in the **From** and **To** date fields, provide the start and end dates.

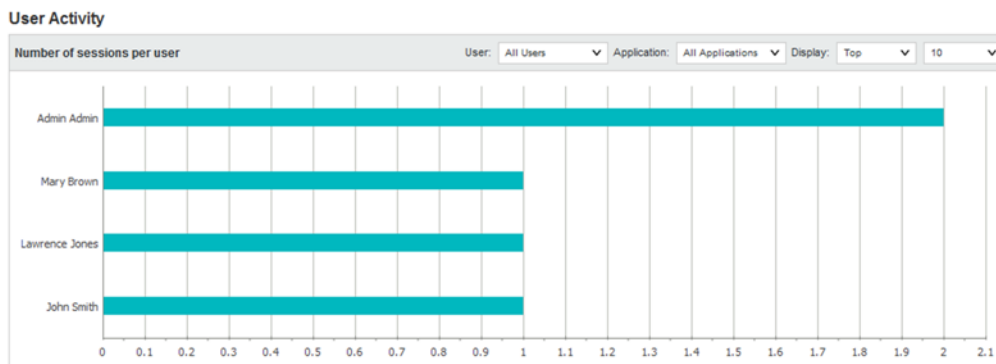
- (c) After selecting a time frame or range of dates, to update the reports to reflect the new selection, click **Update Report**.
2. For the **Number of sessions over time** report, you can control the date/time unit used to display the results.

To change the date/time unit, select the new unit from the drop-down list.



The report is updated automatically to use the new value.

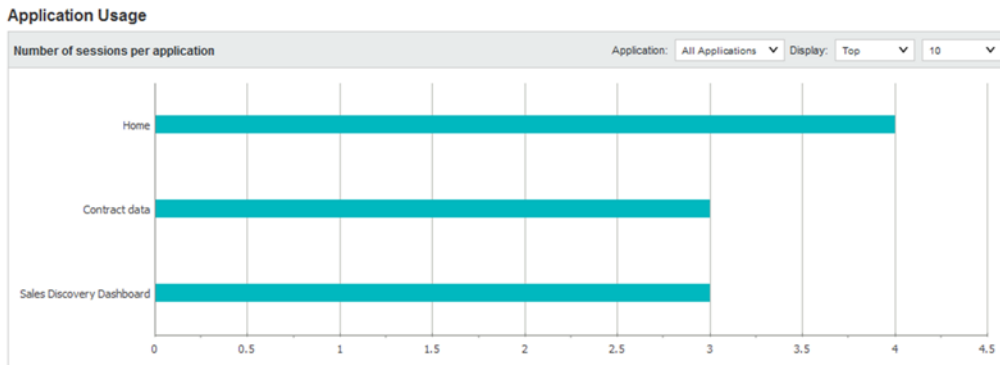
3. By default, the **User Activity** report shows the top 10 number of sessions per user for all projects during the selected time period.



You can narrow the report to show values for a specific user or project, and change the number of values displayed.

- (a) To narrow the report to a specific user, from the **User** drop-down list, select the user.
The report is updated to display the top or bottom number of sessions for projects the user has used.
- (b) To narrow the report to a specific project, from the **Project** drop-down list, select the project.
The report is updated to show the users with the top or bottom number of sessions for users.
If you select both a specific project and a specific user, the report displays a single bar showing the number of sessions for that user and project.
- (c) Use the **Display** settings to control the number of values to display, and whether to display the top or bottom values.

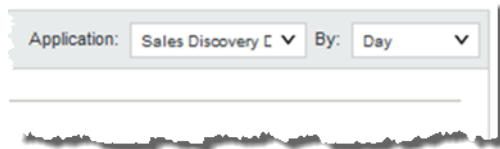
4. By default, the **Project Usage** report shows the 10 projects with the most sessions for the selected time range.



You can narrow the report to show values for a specific project, and change the number of values displayed.

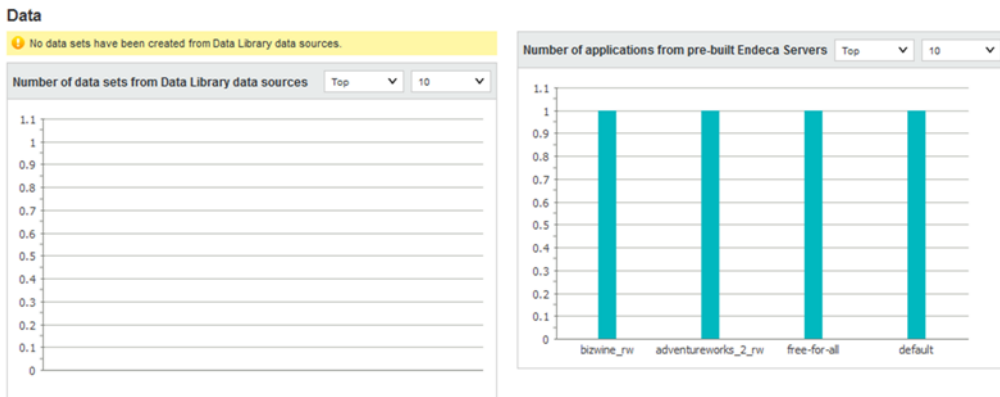
- (a) To narrow the report to a specific project, from the **Project** drop-down list, select the project. The report is changed to a line chart showing the number of sessions per day for the selected project.

A date unit drop-down list is added to allow you to select the unit to use.



For example, you can display the number of sessions per day, per week, or per month.

- (b) If you are displaying the number of sessions for all projects, use the **Display** settings to control the number of values to display, and whether to display the top or bottom values.
5. For the **Data** reports, you can control the number of values to display, and whether to display the top or bottom values.





Chapter 10

Determining and Configuring the Locale to Use

The Big Data Discovery user interface and project data can be displayed in different locales.

[Locales and their effect on the user interface](#)

[How Studio determines the locale to use](#)

[Setting the available locales](#)

[Selecting the default locale](#)

[Configuring the preferred locale for a user](#)

Locales and their effect on the user interface

The locale determines the language in which to display the user interface. It can also affect the format of displayed data values.

Big Data Discovery is configured with a default locale as well as a list of available locales.

Each user account also is configured with a preferred locale, and the user menu includes an option for users to select the locale to use.

In Big Data Discovery, when a locale is selected:

- User interface labels are displayed using the locale
- Display names of attributes are displayed in the locale.
If there is not a version for that locale, then the default locale is used.
- Data values are formatted based on the locale.

Supported locales

Big Data Discovery supports the following languages:

- Chinese - Simplified
- English - US
- Japanese
- Korean
- Portuguese - Brazilian
- Spanish

Note that this is a subset of the languages supported by the Dgraph.

How Studio determines the locale to use

When users log in, Studio needs to determine the locale to use to display the user interface and data.

[Locations where the locale may be set](#)

[Scenarios for selecting the locale](#)

Locations where the locale may be set

The locale is set in different locations.

The locale can come from:

- Cookie
- Browser locale
- Default locale
- User preferred locale, stored as part of the user account
- Locale selected using the **Change locale** option in the user menu, which is also available to users who have not yet logged in.

Scenarios for selecting the locale

The locale used depends upon the type of user, the Big Data Discovery configuration, and how the user entered Big Data Discovery.

For the scenarios listed below, Big Data Discovery determines the locale as follows:

Scenario	How the locale is determined
A new user is created	<p>The locale for a new user is initially set to Use Browser Locale, which indicates to use the current browser locale.</p> <p>This value can be changed to a specific locale.</p> <p>If the user is configured with a specific locale, then that locale is used for the user unless they explicitly select a different locale or enter with a URL that includes a supported locale.</p>
A non-logged-in user navigates to Big Data Discovery	<p>For a non-logged-in user, Big Data Discovery first tries to use the locale from the cookie.</p> <p>If there is no cookie, or the cookie is invalid, then Big Data Discovery tries to use the browser locale.</p> <p>If the current browser locale is not one of the supported locales, then the default locale is used.</p>

Scenario	How the locale is determined
A registered user logs in	<p>When a user logs in, Big Data Discovery first checks the locale configured for their user account.</p> <ul style="list-style-type: none"> If the user's locale is set to Use Browser Locale, then Big Data Discovery tries to use the locale from the cookie. <p>If there is no cookie, or if the cookie is invalid, then Big Data Discovery tries to use the browser locale.</p> <p>If the current browser locale is not a supported locale, then the default locale is used.</p> <ul style="list-style-type: none"> If the user account is configured with a locale value other than Use Browser Locale, then Big Data Discovery uses that locale, and also updates the cookie with that locale.
A non-logged-in user uses the user menu option to select a different locale	<p>When a non-logged-in user selects a locale, Big Data Discovery updates the cookie with the new locale.</p> <p>Note that this locale change is only applied locally. It is not applied to all non-logged-in users.</p>
A logged-in user uses the user menu option to select a different locale	<p>When a logged-in user selects a locale, Big Data Discovery updates both the user's account and the cookie with the selected locale.</p>

Setting the available locales

Big Data Discovery is configured with a list of available locales. This list is used to populate the drop-down list for configuring the default locale, user default locale, and the available locales displayed for the **Change locale** option.

You can customize the setting to constrain the list. Supported locales are specified in `portal.properties`.

```
locales=de_DE, en_US, es_ES, fr_FR, it_IT, ja_JP, ko_KR, pt_PT, zh_CN, zh_TW
```

To reduce this list:

- Copy this parameter into `portal-ext.properties`.
- Update the list to remove the locales that you do not want to be available.

For example, to only support English, French, and Japanese, you would update it to:

```
locales=en_US, fr_FR, ja_JP
```

Selecting the default locale

Big Data Discovery is configured with a default locale, which you can update from the **Control Panel**.

Note that if you have a clustered implementation, make sure to configure the same locale for all of the instances in the cluster.

To select the default locale:

1. On the **Control Panel** menu, in the **Platform Settings** section, click **Display Settings**.
2. On the **Display Settings** page, from the **Locale** drop-down list, select the default locale.

Display Settings

Locale

United States - English ▼

Time Zone

(UTC) Coordinated Universal Time ▼

3. Click **Save**.

Configuring the preferred locale for a user

Each user account is configured with a preferred locale. The default value for new users is **Use Browser Locale**, which indicates to use the current browser locale.

To configure the preferred locale for a user:

1. To display the setting for your own account, sign in to Studio, and in the header, select **User Options > My Account**.

My Account

* Required

User Details

Screen Name:*

Password:

Email Address:*

Retype Password:

First Name:*

Display Settings

Locale:

Middle Name:

Time Zone:

Last Name:*

Role: Administrator

▶ INHERITED ROLES

Cancel

Save

2. To display the setting for another user:
 - (a) In the Big Data Discovery header, click the control panel icon.
 - (b) In the **Control Panel** menu, select **User Settings > Users**.

(c) Locate the user and click **Actions > Edit**.

Add/Edit User

User Details

*Required

Screen Name:*	rwiggum	Email Address:*	ralph.wiggum@ssotest.com
First Name:*	Ralph	Password:*	
Middle Name:		Retype Password:*	
Last Name:*	Wiggum	Role:*	Restricted User ▼
		Locale:*	United States - English ▼

▶ INHERITED ROLES

▶ PROJECTS

Cancel Save

3. From the **Locale** drop-down list, select the preferred locale for the user.
4. Click **Save**.



Chapter 11

Configuring Settings for Outbound Email Notifications

Big Data Discovery includes settings to enable sending email notifications. Email notifications can include account notices, bookmarks, and snapshots.

Configuring the email server settings

Configuring the sender name and email address for notifications

Setting up the Account Created and Password Changed notifications

Configuring the email server settings

In order for users to be able to email bookmarks, you must configure the email server settings. The email address associated with the outbound server is used as the From address on the bookmark email message.

To configure the email server settings:

1. In the Big Data Discovery header, click the **Control Panel** icon.
2. Select **Server > Server Administration**
3. Click the **Mail** tab.
4. Fill out the fields for the incoming mail server:
 - (a) In the **Incoming POP Server** field, enter the name of the POP server to use to receive email.
 - (b) In the **Incoming Port** field, enter the port number for the POP server.
 - (c) If you are not using the SMTPS mail protocol to send the email, then the **Use a Secure Network Connection** checkbox must be unchecked.
 - (d) In the **User Name** field, type the email address to associate with the mail server.
This is the email address used as the **From:** address when end users email bookmarks.
 - (e) In the **Password** field, type the email password associated with the email address.

5. Fill out the fields for the outbound mail server:

Outgoing SMTP Server	<input type="text" value="acme.com.s7a1.pstmp.com"/>
Outgoing Port	<input type="text" value="25"/>
Use a Secure Network Connection	<input type="checkbox"/>
User Name	<input type="text" value="user_user@acme.com"/>
Password	<input type="password" value="*****"/>

- (a) In the **Outgoing SMTP Server** field, enter the name of the SMTP server to use to send the email.
 - (b) In the **Outgoing Port** field, enter the port number for the SMTP server.
 - (c) If you are not using the SMTPS mail protocol to send the email, then the **Use a Secure Network Connection** checkbox must be unchecked.
 - (d) In the **User Name** field, type the name to display for the notification sender.
This is the email address used as the From address when end users email bookmarks.
 - (e) In the **Password** field, type the email password associated with the email address.
6. Click **Save**.

Configuring the sender name and email address for notifications

From the **Email Settings** page of the **Control Panel**, you can configure the sender name and email address to display on outbound notifications.

To configure the sender name and email address:

1. On the **Control Panel** menu, click **Email Settings**.
2. On the **Settings** tab, in the **Name** field, type the name to display for the notification sender.
3. In the **Address** field, type the email address to display for the notification sender. The sender address is used as the reply-to address for most notifications. For bookmarks and snapshots, the reply-to address is the email address of the user who creates the request.
4. Click **Save**.

Setting up the Account Created and Password Changed notifications

From the **Email Settings** page of the **Control Panel**, you can configure the notifications sent when an account is created and when a user's password is changed.

These notifications only apply to users created and managed within Big Data Discovery.

The configuration includes:

- Whether to send the notification

- The subject line of the email message
- The content of the email message

To set up the Account Created and Password Changed notifications:

1. On the **Control Panel** menu, click **Email Settings**.
2. To configure the Account Created notification:
 - (a) Click the **Account Created Notification** tab.
 - (b) By default, the notification is enabled, meaning that when new users are created in Big Data Discovery, they receive the notification. To disable the notification, uncheck the **Enabled** checkbox.
 - (c) In the **Subject line** field, type the text of the email subject line.

The subject line can include any of the dynamic values listed at the bottom of the tab.

For example, to include the user's Big Data Discovery screen name in the subject line, include [\$USER_SCREENNAME\$] in the subject line.
 - (d) In the **Body** text area, type the text of the email message.

The message text can include any of the dynamic values listed at the bottom of the tab.

For example, to include the user's Big Data Discovery screen name in the message text, include [\$USER_SCREENNAME\$] in the message text.
 - (e) To save the message configuration, click **Save**.
3. To configure the Password Changed notification:
 - (a) Click the **Password Changed Notification** tab.
 - (b) By default, the notification is enabled, meaning that when new users are created in Big Data Discovery, they receive the notification. To disable the notification, uncheck the **Enabled** checkbox.
 - (c) In the **Subject line** field, type the text of the email subject line.

The subject line can include any of the dynamic values listed at the bottom of the tab.

For example, to include the user's Big Data Discovery screen name in the subject line, include [\$USER_SCREENNAME\$] in the subject line.
 - (d) In the **Body** text area, type the text of the email message.

The message text can include any of the dynamic values listed at the bottom of the tab.

For example, to include the user's Big Data Discovery screen name in the message text, include [\$USER_SCREENNAME\$] in the message text.
 - (e) To save the message configuration, click **Save**.



Managing Projects from the Control Panel

The **Control Panel** provides options for Big Data Discovery administrators to configure and remove projects.

[Configuring the project type](#)

[Configuring the visibility type for a page](#)

[Adding and removing project members](#)

[Assigning project roles to project members](#)

[Certifying a project](#)

[Making a project active or inactive](#)

[Deleting projects](#)

Configuring the project type

The project type determines whether the project is visible to users on the **Catalog**.

The project types are:

Project Type	Description
Public	The project is visible to all logged-in users, and all logged-in users can select the project in order to view public pages. Project members can also see private pages. Membership must be granted by a project administrator.
Private	The project is visible only to project members. Membership must be granted by a project administrator. Projects are by default private.

If you change the project type, then the page visibility type for all of the project pages changes to match the project type.

To change the project type for a project:

1. In the Studio header, select **Control Panel**.
2. Select **User Settings>Projects**
3. Click the **Actions** link for the project, then select **Edit**
4. From the **Type** drop-down list, select the appropriate project type.

5. Click **Save**.

Configuring the visibility type for a page

Whether users can have access to pages within a project, particularly pages for public projects they're not a member of, is based on the page visibility type.

The page visibility type works similarly to the project type. It determines whether users can view the page without logging in or being a project member.

The page visibility types are:

Page Type	Description
Public	<p>A public page is visible to all logged-in users, including users who are not members of the project.</p> <p>When non-logged-in users navigate to the URL, they are prompted to log in before they can view the page.</p>
Private	<p>A private page is only visible to logged-in users who are members of the project.</p> <p>When non-logged-in users navigate to the page URL, they are prompted to log in.</p> <p>If they log in and are not a member of the project, then they cannot view the page.</p>

By default, the page visibility type is the same as the project visibility type.

To select a different visibility type for a project page:

1. In the Studio header, select **Control Panel**.
2. Select **User Settings>Projects**
3. For the project containing the page you want to configure, click the **Actions** link, then click **Manage Pages**.
4. In the page list at the left, click the page name.

5. Click the **Page** tab for the selected page.

The screenshot shows the 'Edit Page' configuration for a page named 'Wine'. The 'Page' tab is selected, and the 'Children' tab is also visible. The configuration fields include:

- Name: Wine
- HTML Title: (empty)
- Type: Portlet
- Page Visibility Type: Private
- Hidden: (checkbox)
- Friendly URL: http://appdev-x2k8-p7.us.oracle.com:8080/oid/web/sales-discovery-dashboard /new-page
- Query String: (empty)
- Icon: (Browse... No file selected)
- Use Icon: (checkbox)
- Target: (empty)

At the bottom, there are buttons for 'Save', 'Permissions', and 'Delete'.

6. From the **Page Visibility Type** drop-down list, select the visibility type.
7. To save the change, click **Save**.

Adding and removing project members

From the **Control Panel Projects** page, Big Data Discovery administrators can add and remove members from any project.

From the **Control Panel**, to manage the membership for a project:

1. In the **Control Panel** menu, click **Projects**.
2. For the project you want to manage membership for, click the **Actions** button, then click **Assign Members**.

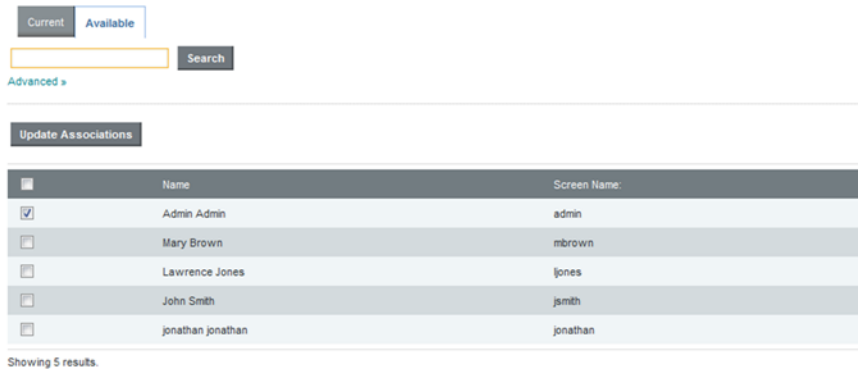
On the membership page, the **Current** tab lists the current project members.

The screenshot shows the membership management page with the 'Current' tab selected. It includes a search bar, an 'Update Associations' button, and a table listing project members.

<input checked="" type="checkbox"/>	Name	Screen Name	Application Roles	
<input checked="" type="checkbox"/>	Admin Admin	admin	Application Member, Application Administrator	Assign User Roles

Showing 1 result.

The **Available** tab lists all of the users. For current members, the checkbox is checked.



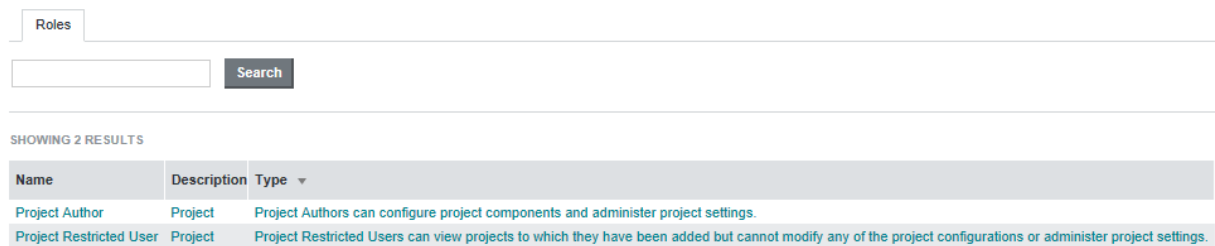
3. To add a user as a new member, on the **Available** tab, check the user's checkbox.
4. To remove a member, on either the **Current** or **Available** tab, uncheck the user's checkbox.
5. To save the membership changes, click **Update Associations**.

Assigning project roles to project members

Big Data Discovery administrators can change any project's membership to determine whether members are project authors or users.

To assign project roles to project members:

1. In the Studio header, select **Control Panel**.
2. Select **User Settings > Projects**
3. Click the **Actions** button, then click **Assign User Roles**.
4. On the **Roles** tab, click the role you want to assign.



The **Current** tab lists the users who currently have the selected role.

The **Available** tab lists all of the project members. Members who already have that role have the checkbox checked.



Current Available

Search

Advanced »

Update Associations

	Name	Screen Name
<input checked="" type="checkbox"/>	Admin Admin	admin
<input type="checkbox"/>	John Smith	jsmith

Showing 2 results.

5. On the **Available** tab, to assign the selected role to a user, check the checkbox.
6. On the **Current** or **Available** tab, to remove the role from the user, uncheck the checkbox.
7. Click **Update Associations**.

Certifying a project

Big Data Discovery administrators can certify a project.

Certifying a project can be used to indicate that the project content and functionality has been reviewed and the project is approved for use by all users who have access to it.

Note that only Big Data Discovery administrators can certify a project. Project administrators cannot change the certification status.

To certify a project:

1. In the **Control Panel** menu, select **User Settings>Projects**.
2. Click the **Actions** link for the project, then click **Edit**.
3. On the project configuration page, to certify the project, check the **Certified** checkbox.
4. Click **Save**.

Making a project active or inactive

By default, a new project is marked as active. From the **Control Panel**, Big Data Discovery administrators can control whether a project is active or inactive. Inactive projects are not displayed on the **Catalog**.

Note that this option only available to Big Data Discovery administrators.

To make a project active or inactive:

1. In the Studio header, select **Control Panel**.
2. Select **User Settings>Projects**
3. Click the **Actions** link for the project, then click **Edit**.
4. To make the project inactive, uncheck the **Active** checkbox. If the project is inactive, then to make the project active, check the **Active** checkbox.

5. Click **Save**.

Deleting projects

From the **Control Panel**, Big Data Discovery administrators can delete projects.

To delete a project:

1. In the **Control Panel** menu, select **User Settings>Projects**.
2. Click the **Actions** link for the project you want to remove.
3. Click **Delete**.

Part V

Controlling User Access to Studio



Chapter 13

Configuring User-Related Settings

You configure settings for passwords and user authentication in the Studio **Control Panel**.

[Configuring authentication settings for users](#)

[Configuring the password policy](#)

[Restricting the use of specific screen names and email addresses](#)

Configuring authentication settings for users

Each user has both an email address and a screen name. By default, users log in to Studio using their email addresses.

To configure the authentication settings for users:

1. In the Studio header, click the **Control Panel** icon.
2. Select **Platform Settings>Credentials** .
3. On the **Credentials** page, click the **Authentication** tab.

Credentials

Reserved Credentials Authentication

How do users authenticate?

By Email Address ▾

Allow users to automatically login?

Allow users to request forgotten passwords?

Configure Authentication

4. From the **How do users authenticate?** drop-down list, select the name used to log in.
To have users log in using their email address, select **By Email Address**. This is the default.
To have users log in using their screen name, select **By Screen Name**.
5. To enable the **Remember me** option on the login page, so that login information is saved when users log in, check the **Allow users to automatically login?** checkbox.
6. To enable the **Forgot Your Password?** link on the login page, so that users can request a new password if they forget it, check the **Allow users to request forgotten passwords?** checkbox.
7. Click **Save**.

Configuring the password policy

The password policy sets the requirements for creating and setting Big Data Discovery passwords.

To configure the password policy:

1. On the **Control Panel** menu, click **Password Policies**.

The **Password Policies** page is displayed.

The screenshot shows the 'Password Policies' configuration page. It is divided into three sections: 'Options', 'Syntax Checking', and 'Security'. Under 'Options', there are two checked checkboxes: 'Changeable' and 'Change Required'. Under 'Syntax Checking', there are two checked checkboxes: 'Syntax Checking Enabled' and 'Allow Dictionary Words', and a 'Minimum Length' field with the value '6'. Under 'Security', there are two unchecked checkboxes: 'History Enabled' and 'Expiration Enabled', each with a help icon.

2. Under **Options**, if the **Changeable** checkbox is checked, then the **My Account** and edit user pages include fields to change the user's password.

If the box is not checked, then user passwords cannot be changed from within Big Data Discovery. If you are using LDAP to manage users, then you should uncheck the checkbox.

3. If users can change their passwords, then the **Change Required** checkbox is displayed. To require users to change their password the first time they log in, make sure this checkbox is checked.

If the box is not checked, then they are not prompted to change the password.

4. Under **Syntax Checking** to enable syntax checking (enforcing password requirements), check the **Syntax Checking Enabled** checkbox.

If the box is not checked, then there are no restrictions on the password format.

5. If syntax checking is enabled, then:

- (a) To allow passwords to include words from the dictionary, check the **Allow Dictionary Words** checkbox.

If the box is not checked, then passwords cannot include words.

- (b) In the **Minimum Length** field, type the minimum length of a password.

6. To prevent users from using a recent previous password:
 - (a) Under **Security**, check the **History Enabled** checkbox.

Security

History Enabled ?

History Count ?

Expiration Enabled ?

Maximum Age ?

Warning Time ?

Grace Limit ?

- (b) From the **History Count** drop-down list, select the number of previous passwords to save and prevent the user from using.

For example, if you select 6, then users cannot use their last 6 passwords.
7. To have passwords expire:
 - (a) Check the **Expiration Enabled** checkbox.

You should not enable expiration if users cannot change their passwords in Big Data Discovery.
 - (b) From the **Maximum Age** drop-down list, select the amount of time before a password expires.
 - (c) From the **Warning Time** drop-down list, select the amount of time before the expiration to begin displaying warnings to the user.
 - (d) In the **Grace Limit** field, type the number of times a user can log in using an expired password.

Restricting the use of specific screen names and email addresses

If needed, you can configure lists of screen names and email addresses that should not be used for Studio users.

To restrict the user of specific screen names and email addresses:

1. In the Studio header, click the **Control Panel** icon.
2. Select **Platform Settings > Credentials** .
3. On the **Reserved Credentials** tab, in the **Screen Names** text area, type the list of screen names that cannot be used.

Put each screen name on a separate line.
4. In the **Email Addresses** text area, type the list of email addresses that cannot be used.

Put each email address on a separate line.



Chapter 14

Creating and Editing Studio Users

In Studio, roles are used to control access to general features as well as to access specific projects and data. The **Users** page on the **Control Panel** provides options for creating and editing Studio users.

[About role privileges](#)

[Creating a new Studio user](#)

[Editing a Studio user](#)

[Deactivating, reactivating, and deleting Studio users](#)

About role privileges

Each Studio user is assigned a user role. The user role determines a user's access to features within Studio.

Regular roles and project-specific roles

There are regular roles and project-specific roles for Studio users. The regular roles are Administrator, Power User, Restricted User, and User. These roles control access to Studio features in data sets, projects, and Studio administrative configuration.

In addition to regular roles, a Studio user may also have a project-specific role. The project-specific roles are Project Author and Project Restricted User. These roles control access to Studio features in data sets, projects, and Studio administrative configuration but within a specific project rather than across Studio.

An administrator can add a project-specific roles and a regular role to a user. This is the typical use case for project-specific access. A Power User or User is additionally assigned the Project Author role for greater privileges within a specified project. For example, an administrator could add a Project Author role to a User. Becoming a Project Author gives a User greater privileges within a specified project.

Inherited roles

A Studio user might have a number of assigned roles. He or she could have a regular role, a project-specific role, and belong to a user group that has a role assigned to all members of that group. In cases like this, the highest role assignment determines a user's privileges. It doesn't matter if the highest role is directly assigned or inherited from a group.

Role descriptions

The user roles are as follows:

Role	Description
<p>Administrator</p>	<p>Administrators have full access to all features in Studio.</p> <p>Administrators can:</p> <ul style="list-style-type: none"> • View all projects • Create and delete data sets and projects • Configure and manage all projects • Use all of the Control Panel features
<p>Power User</p>	<p>Power users can:</p> <ul style="list-style-type: none"> • View projects, based on the project and page type and their projects membership • Create and delete data sets and projects • Configure and manage projects for which they are an administrator • Transform data sets, if they are assigned the Project Author role • Export to HDFS and create new data sets • Edit their account information <p>Power users cannot:</p> <ul style="list-style-type: none"> • Access Control Panel features
<p>User</p>	<p>Users can:</p> <ul style="list-style-type: none"> • View projects, based on the project and page type and their project membership. • Create and delete data sets and projects • Transform data sets, if they are assigned the Project Author role • Edit their account information <p>Users cannot:</p> <ul style="list-style-type: none"> • Create new data sets or projects • Access Control Panel features • Export to HDFS

Role	Description
<p>Restricted User</p>	<p>This is the default user role for new users. It has the most restricted privileges and is essentially a read-only role. This is the default user role for new users.</p> <p>Restricted users can:</p> <ul style="list-style-type: none"> • View projects, based on the project and page type and their project membership. <p>Restricted users cannot:</p> <ul style="list-style-type: none"> • Create new data sets or projects • Access Control Panel features • Access Transform features • Export to HDFS or create new data sets • Configure and manage projects
<p>Project Author</p>	<p>Project authors are similar to administrators but have privileges for a particular project. Also, an administrator can assign the Project Author role to a Power User, Restricted User, and User. (This allows them to configure and manage projects in a way that their primary role would otherwise not allow.)</p> <p>Project authors can:</p> <ul style="list-style-type: none"> • Configure and manage projects for which they are an author • Transform data <p>Project authors cannot:</p> <ul style="list-style-type: none"> • Create new data sets • Access Control Panel features
<p>Project Restricted User</p>	<p>Project Restricted Users are similar to Users but are restricted to a particular project.</p> <p>Project Restricted Users can:</p> <ul style="list-style-type: none"> • View projects, based on the project and page type and their project membership. <p>Project restricted users cannot:</p> <ul style="list-style-type: none"> • Create new data sets or projects • Access Control Panel features • Access Transform features • Access Project Settings features • Export to HDFS

Creating a new Studio user

If you are not using LDAP, you may want to create Studio users manually.

For example, for a small development instance, you may just need a few users to develop and test projects. Or if your LDAP users for a production site are all end users, you may need a separate user account for administering the site.

To create a new Studio user:

1. In the Studio header, click the **Control Panel** icon.
2. Select **User Settings>Users** .
3. Click **Add**.
The **Details** page for the new user displays.
4. In the **Screen Name** field, type the screen name for the user.
The screen name must be unique, and cannot match the screen name of any current active or inactive user.
5. In the **Email Address** field, type the user's email address.
6. For the user's name, enter values for at least the **First Name** and **Last Name** fields.
The **Middle Name** field is optional.
7. To create the initial password for the user:
 - (a) In the **Password** field, enter the password to assign to the new user.
 - (b) In the **Retype Password** field, type the password again.
By default, the Studio password policy requires users to change their password the first time they log in.
8. From the **Locale** drop-down list, select the preferred locale for the user.
9. From the **Role** drop-down list, select the user role to assign to the user.
For details, see [About role privileges on page 99](#).

10. From the **Projects** section at the bottom of the dialog, to assign the user to projects:

Projects	Description	Project Role	Type
<input type="checkbox"/> asimoRegressionTests-11h4...		Project Restricted User	Private
<input type="checkbox"/> asimoSmokeTests-ie-14h14...		Project Restricted User	Private
<input type="checkbox"/> asimoRegressionTests-11h4...		Project Restricted User	Private
<input type="checkbox"/> asimoSmokeTests-ie-14h14...		Project Restricted User	Private
<input type="checkbox"/> alanTest		Project Restricted User	Private

- Check the checkbox next to each project you want the new user to be a member of.
 - For each project, from the **Role** drop-down list, select the project role to assign to the user.
11. Click **Save**.

The user is added to the list of users.

Editing a Studio user

The **Users** page also allows you to edit a user's account.

From the **Users** page, to edit a user:

- In the Studio header, click the **Control Panel** icon.
- Select **User Settings>Users**
- Click the **Actions** button next to the user.
- Click **Edit**.
- To change the user's password:
 - In the **Password** field, type the new password.
 - In the **Retype Password** field, re-type the new password.
- To change the user role, from the **Role** drop-down list, select the new role.
- Under **Projects**, to add a user as an project member:
 - Make sure the drop-down list is set to **Available Projects**. These are projects the user is not yet a member of.
 - Check the checkbox next to each project you want to add the user to.
 - For each project, from the **Role** drop-down list, select the project role to assign to the user.

8. Under **Projects**, to change the project role for or remove the user from a project:
 - (a) From the drop-down list, select **Assigned Projects**.

The list shows the projects the user is currently a member of.
 - (b) To change the user's project role, from the **Role** drop-down list, select the new project role.
 - (c) To remove the user from a project, uncheck the checkbox.
9. Click **Save**.

Deactivating, reactivating, and deleting Studio users

From the **Users** page of the **Control Panel**, you can make an active user inactive. You can also reactivate or delete inactive users.

Note that you cannot make your own user account inactive, and you cannot delete an active user.

From the **Users** page, to change the status of a user account:

1. To make an existing user inactive:
 - (a) In the users list, check the checkbox for the user you want to deactivate.
 - (b) Click the **Deactivate** button.

Big Data Discovery prompts you to confirm that you want to deactivate the user.

The user is then removed from the list of active users.

Note that inactive users are not removed from Big Data Discovery.
2. To reactivate or delete an inactive user:
 - (a) Click the **Advanced** link below the user search field.

Big Data Discovery displays additional user search fields.
 - (b) From the **Active** drop-down list, select **No**.

Note that if you change the **Match type** to **Any**, you must also provide search criteria in at least one of the other fields.
 - (c) Click **Search**.

The users list displays only the inactive users.
 - (d) Check the checkbox for the user you want to reactivate or delete.
 - (e) To reactivate the user, click the **Restore** button.
 - (f) To delete the user, click the **Delete** button.



Integrating with an LDAP System to Manage Users

If you have an LDAP system, you can allow users to use those credentials to log in to Big Data Discovery.

[About using LDAP](#)

[Configuring the LDAP settings and server](#)

[Preventing LDAP users from changing passwords in Big Data Discovery](#)

[Preventing encrypted LDAP passwords from being stored in Big Data Discovery](#)

[Assigning roles based on LDAP user groups](#)

About using LDAP

LDAP (Lightweight Directory Access Protocol) allows you to have users connect to Big Data Discovery using their existing LDAP user accounts, rather than creating separate user accounts from within Big Data Discovery. LDAP is also used when integrating with a single sign-on (SSO) system.

Configuring the LDAP settings and server

The LDAP settings on the **Control Panel** include whether LDAP is enabled and required for authentication, the connection to the LDAP server, and whether to support batch import or export to or from the LDAP directory. The method for processing batch imports is set in `portal-ext.properties`.

In `portal-ext.properties`, the setting `ldap.import.method` determines how to perform batch imports from LDAP. This setting is only applied if batch import is enabled. The available values for `ldap.import.method` are:

Value	Description
user	<p>Indicates to use user-based import. This is the default value.</p> <p>User-based batch import uses the import search filter configured in the User Mapping section of the LDAP tab.</p> <p>For user-first import, Big Data Discovery:</p> <ol style="list-style-type: none">1. Uses the user import search filter to run an LDAP search query.2. Imports the resulting list of users, including all of the LDAP groups the user belongs to. <p>The group import search filter is ignored.</p>

Value	Description
group	<p>Indicates to use group-based import.</p> <p>Group-based import uses the import search filter configured in the Group Mapping section of the LDAP tab.</p> <p>For group-based import, Big Data Discovery:</p> <ol style="list-style-type: none"> 1. Uses the group import search filter to run an LDAP search query. 2. Imports the resulting list of groups, including all of the users in those groups. <p>The user import search filter is ignored.</p>

The value you should use depends partly on how your LDAP system works. If your LDAP directory only provides user information, without any groups, then you have to use user-based import. If your LDAP directory only provides group information, then you have to use group-based import.

To configure the LDAP settings:

1. In the Big Data Discovery header, click the control panel icon.
2. On the **Control Panel** menu, click **Credentials**.
3. On the **Credentials** page, click the **Authentication** tab.
4. On the **Authentication** tab, click **Configure Authentication**.

The **Configure Authentication** dialog is displayed, with the **LDAP** tab selected.



5. To enable LDAP authentication, check the **Enabled** checkbox.
6. To only allow users to log in using an LDAP account, check the **Required** checkbox.
If this box is checked, then any users that you create manually in Big Data Discovery cannot log in.
To make sure that users you create manually can log in, make sure that this box is not checked.
7. To populate the LDAP server configuration fields with default values based on a specific type of provider, from the **Provider type** drop-down list, select the type of server you are using.
If you select the **Custom** option, then the fields are cleared.

8. The **Connection** settings cover the basic connection to LDAP:

▼ Connection

Base Provider URL: [?](#) Principal:

Base DN: [?](#) Credentials:

Test Connection

Field	Description
Base Provider URL	The location of your LDAP server. Make sure that the machine on which Big Data Discovery is installed can communicate with the LDAP server. If there is a firewall between the two systems, make sure that the appropriate ports are opened.
Base DN	The Base Distinguished Name for your LDAP directory. For a commercial organization, it may look something like: <code>dc=companynamehere,dc=com</code>
Principal	The user name of the administrator account for your LDAP system. This ID is used to synchronize user accounts to and from LDAP.
Credentials	The password for the administrative user.

After providing the connection information, to test the connection to the LDAP server, click the **Test Connection** button.

9. Under **User Mapping**:

▼ User Mapping

Authentication Search Filter: <input checked="" type="radio"/> Password:	First Name:	
<input type="text" value="{&(objectClass=person)(sAMA"/>	<input type="text" value="userPassword"/>	<input type="text" value="givenName"/>
Import Search Filter:	Screen Name:	Last Name:
<input type="text" value="{&(objectClass=person)(!(objec"/>	<input type="text" value="sAMAccountName"/>	<input type="text" value="sn"/>
Email Address:	Full Name:	Group:
<input type="text" value="userprincipalname"/>	<input type="text" value="cn"/>	<input type="text" value="memberOf"/>

[Test Users](#)

- (a) Use the search filter fields to configure the filters for finding and identifying users in your LDAP directory.

Field	Description
Authentication Search Filter	<p>The search criteria for user logins.</p> <p>If you do not enable batch import of LDAP users, then the first time a user tries to log in, Big Data Discovery uses this authentication search filter to search for the user in the LDAP directory.</p> <p>By default, users log in using their email address. If you have changed this setting, you must modify the search filter here.</p> <p>For example, if you changed the authentication method to use the screen name, you would modify the search filter so that it can match the entered login name:</p> <pre style="background-color: #f0f0f0; padding: 5px;">(cn=@screen_name@)</pre>
Import Search Filter	<p>The search filter to use for batch import of users.</p> <p>This filter is used if:</p> <ul style="list-style-type: none"> You enable batch import of LDAP users In <code>portal-ext.properties</code>, <code>ldap.import.method</code> is set to <code>user</code> <p>Depending on the LDAP server, there are different ways to identify the user.</p> <p>The default setting <code>(objectClass=inetOrgPerson)</code> usually is fine, but to search for only a subset of users or for users that have different object classes, you can change this.</p>

- (b) Use the remaining fields to map your LDAP attributes to the Big Data Discovery user fields.
- (c) After setting up the attribute mappings, to test the mappings, click **Test Users**.

10. Under **Group Mapping**, map your LDAP groups.

The screenshot shows a configuration panel for 'Group Mapping'. It contains four input fields arranged in a 2x2 grid:

- Import Search Filter:** (objectClass=group)
- Description:** sAMAccountName
- Group Name:** cn
- User:** member

Below these fields is a teal button labeled 'Test Groups'.

- (a) In the **Import Search Filter** field, type the filter for finding LDAP groups.

This filter is used if:

- You enable batch import of LDAP users
- In `portal-ext.properties`, `ldap.import.method` is set to `group`

- (b) Map the following group fields:

- Group Name
- Description
- User

- (c) To test the group mappings, click **Test Groups**.

The system displays a list of the groups returned by your search filter.

11. The **Options** section is used to configure importing and exporting of LDAP user data, and to select the password policy:

The screenshot shows a configuration panel for 'Options'. It contains three checkboxes:

- Import Enabled
- Export Enabled
- Use LDAP Password Policy

- (a) If the **Import Enabled** checkbox is checked, then batch import of LDAP users is enabled.

If the box is not checked, then Big Data Discovery synchronizes each user as they log in.

It is recommended that you leave this box unchecked.

If you do enable batch import, then the import process is based on the value of `ldap.import.method`.

Note also that when using batch import, you cannot filter both the imported users and imported groups at the same time. For user-based batch import mode, you cannot filter the LDAP groups to import. For group-based batch import mode, you cannot filter the LDAP users to import.

- (b) If the **Export Enabled** checkbox is checked, then any changes to the user in Big Data Discovery are exported to the LDAP system.

It is recommended that you leave this box unchecked.

- (c) To use the password policy from your LDAP system, instead of the Big Data Discovery password policy, check the **Use LDAP Password Policy** checkbox.

Preventing LDAP users from changing passwords in Big Data Discovery

When you are using LDAP, it is likely that you want user passwords to be managed outside of Big Data Discovery.

To update the Big Data Discovery password policy so that users cannot change their password in Big Data Discovery:

1. In the **Control Panel** menu, click **Password Policies**.
2. On the **Password Policies** page, to prevent users from being able to change passwords from within Big Data Discovery, uncheck the **Changeable** checkbox.

Options

Changeable
 Change Required

Syntax Checking

Syntax Checking Enabled
 Allow Dictionary Words
 Minimum Length

Security

History Enabled ?
 Expiration Enabled ?

3. To save the changes, click **Save**.

Preventing encrypted LDAP passwords from being stored in Big Data Discovery

By default, when you use LDAP for user authentication, each time a user logs in, Big Data Discovery stores a securely encrypted version of their LDAP password. For subsequent logins, Big Data Discovery can then authenticate the user even when it cannot connect to the LDAP system. For even stricter security, you can configure Big Data Discovery to prevent the passwords from being stored.

To prevent Big Data Discovery from storing the encrypted LDAP passwords:

1. Stop Big Data Discovery.
2. Add the following settings to `portal-ext.properties`:

```
ldap.password.cache.hashing=false
ldap.auth.required=true
auth.pipeline.enable.liferay.check=false
```

3. Restart Big Data Discovery.

Big Data Discovery no longer stores the encrypted LDAP passwords for authenticated users. If the LDAP system is unavailable, Big Data Discovery cannot authenticate previously authenticated users.

Assigning roles based on LDAP user groups

For LDAP integration, it is recommended that you assign roles based on your LDAP groups.

To ensure that users have the correct roles as soon as they log in, you create groups in Big Data Discovery that have the same name as your LDAP groups, but in lowercase, and assign the correct roles to each group.

To create a user group, and assign roles to that group:

1. In the Big Data Discovery header, click the control panel icon.
2. On the **Control Panel**, click **User Groups**.
3. On the **User Groups** page, to add a new group, click **Add**.
The **Add Group** dialog is displayed.
4. In the **Name** field, type the name of the group.
Make sure the name is the lowercase version of the name of a group from your LDAP system.
For example, if the LDAP group is called SystemUsers, then the user group name would be systemusers.
5. In the **Description** field, type a description of the group.
6. To assign roles to the group, from the **Role** drop-down list, select the user role to assign to the group.
The selected roles are assigned to all of the users in the group. For details on the available user roles, see [About role privileges on page 99](#).
7. Click **Save**.
The group is added to the **User Groups** list.



Setting up Single Sign-On (SSO)

You can provide user access by integrating with an SSO system.

[About using single sign-on](#)

[Overview of the process for configuring SSO with Oracle Access Manager](#)

[Configuring the reverse proxy module in OHS](#)

[Registering the Webgate with the Oracle Access Manager server](#)

[Testing the OHS URL](#)

[Configuring Big Data Discovery to integrate with SSO via Oracle Access Manager](#)

[Completing and testing the SSO integration](#)

About using single sign-on

Integrating with single sign-on (SSO) allows Studio users to be logged in to Big Data Discovery automatically once they are logged in to your SSO system.

Note that once Big Data Discovery is integrated with SSO, you cannot create and edit users from within Big Data Discovery. All users get access to Big Data Discovery using their SSO credentials. This means that you can no longer use the default administrative user provided with Big Data Discovery. You will need to make sure that there is at least one SSO user with an Administrator user role for Big Data discovery.

The officially supported method for integrating with SSO is to use Oracle Access Manager, with an Oracle HTTP Server in front of the Big Data Discovery application server. While you may be able to use another SSO tool that supports passing the user name in an HTTP header, you would have to use the documentation and support materials for that tool in order to set up the integration.

The information in this guide focuses on the details and configuration that are specific to the Big Data Discovery integration. For general information on installing Oracle Access Manager and Oracle HTTP Server, see the associated documentation for those products.

Overview of the process for configuring SSO with Oracle Access Manager

Here is an overview of the steps for using Oracle Access Manager to implement SSO in Big Data Discovery.

1. Install Oracle Access Manager 11g, if you haven't already. See the Oracle Access Manager documentation for details.
2. Install Oracle HTTP Server (OHS) 11g. See the Oracle HTTP Server documentation for details.
3. Install OHS Webgate 11g. See the Webgate documentation for details.

4. Create an instance of OHS, and confirm that it is up and running. See the OHS documentation for details.
5. Configure the reverse proxy module for the Big Data Discovery application server in Oracle HTTP Server. See [Configuring the reverse proxy module in OHS on page 113](#).
6. Install the Webgate module into the Oracle HTTP Server. See [Registering the Webgate with the Oracle Access Manager server on page 114](#).
7. In Big Data Discovery, configure the LDAP connection for your SSO implementation. See [Configuring the LDAP connection for SSO on page 116](#).
8. In Big Data Discovery, configure the Oracle Access Manager SSO settings. See [Configuring the Oracle Access Manager SSO settings on page 117](#).
9. Configure Big Data Discovery's web server settings to use the OHS server. See [Completing and testing the SSO integration on page 118](#).
10. Disable direct access to the Big Data Discovery application server, to ensure that all traffic to Big Data Discovery is routed through OHS.

Configuring the reverse proxy module in OHS

For WebLogic Server, you need to update the file `mod_wls_ohs.conf` to add the logout configuration for SSO.

Here is an example of the file with the `/bdd/oam_logout_success` section added:

```
LoadModule weblogic_module    "${ORACLE_HOME}/ohs/modules/mod_wl_ohs.so"
<IfModule weblogic_module>
    WebLogicHost hostName
    WebLogicPort portNumber
</IfModule>

<Location /bdd/oam_logout_success>
    PathTrim /bdd/oam_logout_success
    PathPrepend /bdd/c/portal
    DefaultFileName logout
    SetHandler weblogic-handler
</Location>

<Location />
    SetHandler weblogic-handler
</Location>
```

The `/bdd/oam_logout_success` Location configuration is special for Big Data Discovery. It redirects the default Webgate Logout Callback URL (`/bdd/oam_logout_success`) to an application tier logout within Big Data Discovery. With this configuration, when users sign out of SSO from another application, it is reflected in Big Data Discovery.

Registering the Webgate with the Oracle Access Manager server

After you have installed the OHS Webgate, you use the remote registration (RREG) tool to register the OHS Webgate with the OAM server.

To complete the registration:

1. Obtain the RREG tarball (`rreg.tar.gz`) from the Oracle Access Manager server.
2. Extract the file to the OHS server.
3. Modify the script `oamreg.sh`.

Correct the `OAM_REG_HOME` and `JAVA_HOME` environment variables.

`OAM_REG_HOME` should point to the extracted `rreg` directory created in the previous step.

You may not need to change `JAVA_HOME` if it's already set in your environment.

4. In the `input` directory, create an input file for the RREG tool. The file can include the list of resources secured by this Webgate.

You can omit this list if the application domain already exists.

Here is an example of an input file where the resources have not been set up for the application domain and host in Oracle Access Manager:

```
<?xml version="1.0" encoding="UTF-8"?>
<OAM11GRegRequest>
<serverAddress>http://oamserver.us.mycompany.com:7001</serverAddress>
<hostIdentifier>myserver-1234</hostIdentifier>
<agentName>myserver-1234-webgate</agentName>
<applicationDomain>Big Data Discovery</applicationDomain>
<protectedResourcesList>
  <resource>/bdd</resource>
  <resource>/bdd/.../*</resource>
</protectedResourcesList>
<publicResourcesList>
  <resource>/public/index.html</resource>
</publicResourcesList>
<excludedResourcesList>
  <resource>/excluded/index.html</resource>
</excludedResourcesList>
</OAM11GRegRequest>
```

In this example, the resources have already been set up in Oracle Access Manager:

```
<?xml version="1.0" encoding="UTF-8"?>
<OAM11GRegRequest>
<serverAddress>http://oamserver.us.mycompany.com:7001</serverAddress>
<hostIdentifier>myserver-1234</hostIdentifier>
<agentName>myserver-1234-webgate</agentName>
<applicationDomain>Big Data Discovery</applicationDomain>
</OAM11GRegRequest>
```

In the input file, the parameter values are:

Parameter Name	Description
serverAddress	The full address (<code>http://host:port</code>) of the Oracle Access Manager administrative server. The port is usually 7001.
hostIdentifier	The host identifier string for your host. If you already created a host identifier in the Oracle Access Manager console, use its name here.
agentName	A unique name for the new Webgate agent. Make sure it doesn't conflict with any existing agents in the application domain.
applicationDomain	A new or existing application domain to add this agent into. Each application domain may have multiple agents. An application domain associates multiple agents with the same authentication and authorization policies.

5. Run the tool:

```
./bin/oamreg.sh inband input/inputFileName
```

For example:

```
./bin/oamreg.sh inband input/my-webgate-input.xml
```

When the process is complete, you'll see the following message:

```
Inband registration process completed successfully! Output artifacts are created in the output folder.
```

6. Copy the generated output files from the `output` directory to the OHS instance `config` directory (under `webgate/config/`).
7. Restart the OHS instance.
8. Test your application URL via OHS.

It should forward you to the SSO login form.

Check the OAM console to confirm that the Webgate is installed and has the correct settings.

Testing the OHS URL

Before continuing to the Big Data Discovery configuration, you need to test that the OHS URL redirects correctly to Big Data Discovery.

To test the OHS URL, use it to browse to Big Data Discovery.

You should be prompted to authenticate using your SSO credentials.

Because you have not yet configured the Oracle Access Manager SSO integration in Big Data Discovery, after you complete the authentication, the Big Data Discovery login page is displayed.

Log in to Big Data Discovery using an administrator account.

Configuring Big Data Discovery to integrate with SSO via Oracle Access Manager

In Big Data Discovery, you configure the LDAP connection and Oracle Access Manager connection settings.

[Configuring the LDAP connection for SSO](#)

[Configuring the Oracle Access Manager SSO settings](#)

Configuring the LDAP connection for SSO

The SSO implementation uses LDAP to retrieve and maintain the user information. For the Oracle Access Manager SSO, you configure Big Data Discovery to use Oracle Internet Directory for LDAP.

In Big Data Discovery, to configure the LDAP connection for SSO:

1. In the **Control Panel** menu, click **Credentials**.
2. On the **Credentials** page, click **Authentication**.
3. On the **Authentication** tab, click the **Configure Authentication** button.
The **Configure Authentication** dialog is displayed, with the **LDAP** tab selected.
4. On the **LDAP** tab, check the **Enabled** checkbox. Do not check the **Required** checkbox.
5. From the **Default values** drop-down list, select **Oracle Internet Directory**.
6. Configure the LDAP connection, users, and groups as described in [Configuring the LDAP settings and server on page 105](#).
7. To save the LDAP connection information, click **Save**.
8. Configure the user roles for your user groups as described in [Assigning roles based on LDAP user groups on page 111](#).
9. Make sure that the password policy is configured to not require users to change their password. See [Preventing LDAP users from changing passwords in Big Data Discovery on page 110](#).

Configuring the Oracle Access Manager SSO settings

After you configure the LDAP connection for your SSO integration, you configure the Oracle Access Manager SSO settings.

The settings are on the **SSO** tab on the **Configure Authentication** dialog.



To configure the SSO settings:

1. In the **Control Panel** menu, click **Credentials**.
2. In the **Credentials** page, click the **Authentication** tab.
3. On the **Authentication** tab, click the **Configure Authentication** button.
4. On the **Configure Authentication** dialog, click the **SSO** tab.
5. Check the **Enabled** checkbox.
6. Check the **Import from LDAP** checkbox.
7. From the **Provider Type** drop-down list, select **Oracle Access Manager**.

Note that the only other option is **Custom**, which clears the fields. You would use the **Custom** option if you are using some other tool that passes the user name in an HTTP header. For information on setting up an SSO tool other than Oracle Access Manager, see the documentation and support materials for that tool.

8. Leave the default user header `OAM_REMOTE_USER`.

9. In the **Logout URL** field, provide the URL to navigate to when users log out. Make sure it is the same logout redirect URL you have configured for the Webgate:

For the logout URL, you can add an optional `end_url` parameter to redirect the browser to a final location after users sign out. To redirect back to Big Data Discovery, configure `end_url` to point to the OHS host and port.

For example:

```
http://oamserver.us.mycompany.com:14100/oam/server/logout?end_url=http://
/bddhost.us.company.com:7777/
```

10. To save the configuration, click **Save**.

Completing and testing the SSO integration

The final step in setting up the SSO integration is to add the OHS server host name and port to `portal-ext.properties`.

To complete and test the SSO configuration:

1. In `portal-ext.properties`:

If OHS is not using SSL, then add the following lines:

```
web.server.host=ohsHostName
web.server.http.port=ohsPortNumber
```

If OHS is using SSL, then add the following lines:

```
web.server.protocol=https
web.server.host=ohsHostName
web.server.https.port=ohsPortNumber
```

Where:

- *ohsHostName* is the fully qualified domain name (FQDN) of the server where OHS is installed. The name must be resolvable by Big Data Discovery users.

For example, you would use `webserver01.company.com`, and not `webserver01`.

You need to specify this even if OHS is on the same server as Big Data Discovery.

- *ohsPortNumber* is the port number used by OHS.

2. Restart Big Data Discovery.

Make sure to completely restart the browser to remove any cookies or sessions associated with the Big Data Discovery user login you used earlier.

3. Navigate to the Big Data Discovery URL. The Oracle Access Manager SSO form is displayed.**4. Enter your SSO authentication credentials.**

You are logged in to Big Data Discovery.

As you navigate around Big Data Discovery, make sure that the browser URL continues to point to the OHS server and port.

Part VI

Administering Big Data Discovery Using Enterprise Manager Cloud Control



Chapter 17

Using the Enterprise Manager for Big Data Discovery

This section describes how to use the Enterprise Manager plug-in for Big Data Discovery to administer Big Data Discovery components with Enterprise Manager Cloud Control.

[Before using the Enterprise Manager](#)

[About the Enterprise Manager](#)

[Starting and stopping the Dgraph using Enterprise Manager](#)

[Logging for Big Data Discovery targets in Enterprise Manager](#)

[Configuring verbose logging for a Dgraph target](#)

[Dgraph Administration Operations in Enterprise Manager](#)

Before using the Enterprise Manager

Before you can use the Enterprise Manager for Big Data Discovery, you must already have set up Enterprise Manager Cloud Control and installed and deployed the Enterprise Manager plug-in itself.

For details about these tasks, see *Enterprise Manager Plug-in for Big Data Discovery Installation Guide*.

About the Enterprise Manager

The Enterprise Manager Plug-in for Big Data Discovery extends Oracle Enterprise Manager Cloud Control to add support for monitoring, diagnosing, and managing Big Data Discovery components.

The Enterprise Manager plug-in supports three targets for Oracle Big Data Discovery components — the Cluster target, the Studio target, and the Dgraph target. The Dgraph target also includes information about the Dgraph HDFS Agent (used for importing and exporting data into Hadoop).

In addition to providing support for targets, the plug-in has several customized features, such as support for starting and stopping the Dgraph, and support for the Dgraph administrative operations.

The plug-in provides a convenient way to view and monitor logs, and also search Studio and Dgraph queries.

[The Cluster target](#)

[The Dgraph target](#)

[Information about the Dgraph HDFS Agent](#)

[The Studio target](#)

[Roles and privileges for BDD targets](#)

Security credentials for BDD targets

Connecting to Studio over a secure port

The Cluster target

A Cluster target represents an entire Big Data Discovery cluster deployment, including the Studio and Dgraph instances. The **All Targets** page provides a table that lists the Big Data Discovery clusters and corresponding status (up or down).

For example:

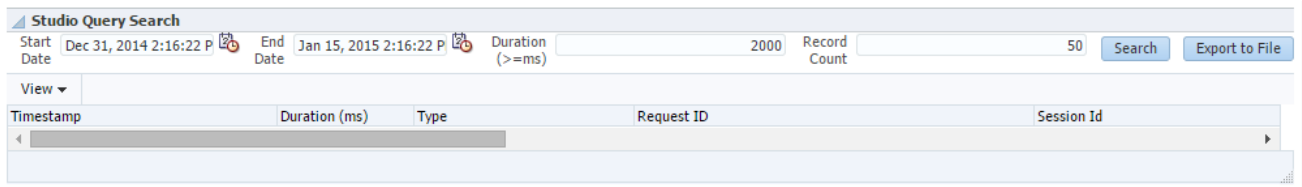
Target Name	Target Type	Target Status	Pending Activation
BigDataCluster_clusterfju	Oracle Big Data Discovery Cluster	Up	
BigDataCluster_clusterfju_1	Oracle Big Data Discovery Cluster	Up	

Clicking a **Cluster** target in the table displays that cluster's home page. On this page, you can see the status and name of each node in the cluster. In particular, on the Cluster page you can see which Dgraph is the leader node (this is useful because some of the Dgraph administrative operations, such as merging index data, can be run only from the leader node).

Some of the regions on the Cluster home page are standard to all Enterprise Manager plug-ins — such as the **Incidents** region. Other regions, such as **Studio Query Search**, are unique to the plug-in for Oracle Big Data Discovery. For example:

Name	Type	Status	Host
bigdatacluster0825.us.oracle.com_BigDataDgraph	Oracle Big Data Discovery Dgraph	Up	bigdatacluster0825.us.oracle.com
bigdatacluster0825.us.oracle.com_BigDataDgraph	Oracle Big Data Discovery Dgraph	Up	bigdatacluster0825.us.oracle.com
bigdatacluster0825.us.oracle.com_BigDataStudio	Oracle Big Data Discovery Studio	Up	bigdatacluster0825.us.oracle.com

Clicking the **Studio Query Search** region displays search features that allow you to search all Studio queries for a range of dates and identify any queries that took longer than the number of milliseconds you specify. For example:



You can also export these results to a file in Excel.

The Dgraph target

A Dgraph target lists information about a single Dgraph instance running on a host and includes information about the Dgraph HDFS Agent associated with that Dgraph. You can view resource utilization, and search Dgraph queries and export your search results to a file in Excel. You can also search and export the HDFS Agent information.

Before accessing this page, set the default preferred credentials for the Dgraph target.

The **Dgraphs** folder page provides a table that lists all Dgraph nodes and each node's status (up or down). For example:



Clicking a Dgraph target in the table displays that Dgraph's home page. This page provides a way to start and stop the Dgraph. The Dgraph manages the Dgraph HDFS Agent, so starting or stopping the Dgraph also starts or stops the Dgraph HDFS Agent. The page also provides the following regions that describe the Dgraph node:

- A **Summary** region lists the basics about installation paths, general configuration, leader/follower information, and so on. This **Summary** region also includes an area for the Dgraph HDFS Agent.
- A **Resource Utilization** region is about CPU and RAM usage for the Dgraph and Dgraph HDFS Agent on that host.
- An **Incidents and Problems** region is standard to all Enterprise Manager plug-ins. This region allows you to search, view, manage, and resolve incidents and problems impacting your environment.
- A **Dgraph Request Search** region allows you to search all Dgraph queries for a range of dates and identify any queries that took longer than the number of milliseconds you specify. You can also export your search results to a file in Excel.

For example:

Summary

Dgraph

- Status: Target Up
- Up Since: Jan 8, 2015 11:25:04 PM
- Availability (%): 100
- Version: 1.0.0.897134
- Cluster Identifier: clusterBdd
- Leader Node: TRUE
- Host: .oracle.com
- Coordinator Host: .oracle.com:2181
- Web Service Port: 7010
- Bulk Load Port: 7019
- Process Id: 24062
- Installation Directory: /localdisk/Test/Oracle/Middleware/BDD1.0/dgraph
- Request Log File Path: /localdisk/Test/Oracle/Middleware/BDD1.0/dgraph/bin/dgraph.reqlog
- Out Log File Path: /localdisk/Test/Oracle/Middleware/BDD1.0/logs/dgraph.out

HDFS Agent

- Status: Up
- Port: 7102
- Process Id: 24155
- Out Log File Path: /localdisk/Test/Oracle/Middleware/BDD1.0/logs/dgraphHDFSAgent.out

Resource Utilization

CPU

Graph showing Dgraph Process CPU (%) and HDFS Agent CPU (%) over time. Both are near 0%.

Memory

Graph showing Dgraph Process Memory (%) and HDFS Agent Memory (%) over time. Both are near 0%.

Incidents and Problems

Dgraph Request Search

Searching the Dgraph queries

Clicking the **Dgraph Request Search** region lets you search all Dgraph queries for a range of dates and identify any queries that took longer than the number of seconds you specify. For example:

Dgraph Request Search

Start Date: Oct 15, 2014 2:31:59 PM End Date: Jan 15, 2015 2:31:59 PM Duration (>ms): 50 Record Count: 50

Timestamp	Request ID	Client IP Address	Response Size (bytes)	Duration (ms)	Process Time (ms)	HTTP Response Code	HTTP URL	Target Name
Jan 09, 2015 00:01:16 AM PST	1422566033832	::ffff:10.182.74.107	672	3302.42	3300.31	200	/ws/ingest	clusterBdd-.oracle.com_Dgraph
Jan 09, 2015 00:01:13 AM PST	1421246989454	::ffff:10.182.74.107	5490	1687.72	1685.55	200	/ws/conversation	clusterBdd-.oracle.com_Dgraph
Jan 09, 2015 00:01:11 AM PST	1421547850363	::ffff:10.182.74.107	545	1074.99	1071.68	200	/ws/config	clusterBdd-.oracle.com_Dgraph
Jan 09, 2015 00:01:09 AM PST	1422733386730	::ffff:10.182.74.107	545	845.48	841.12	200	/ws/config	clusterBdd-.oracle.com_Dgraph
Jan 09, 2015 01:00:05 AM PST	1421010092136	::ffff:10.182.74.107	5488	412.9	410.97	200	/ws/conversation	clusterBdd-.oracle.com_Dgraph
Jan 09, 2015 00:01:08 AM PST	-	::ffff:10.182.74.107	561	338.51	336.71	200	/ws/admin/	clusterBdd-.oracle.com_Dgraph
Jan 09, 2015 00:01:16 AM PST	1422183739612	::ffff:10.182.74.107	672	220.26	218.32	200	/ws/ingest	clusterBdd-.oracle.com_Dgraph
Jan 15, 2015 02:19:29 PM PST	-	::ffff:10.182.74.107	604	89.35	87.11	200	/ws/admin	clusterBdd-.oracle.com_Dgraph
Jan 12, 2015 06:19:57 PM PST	-	::ffff:10.182.74.107	604	89.23	87.24	200	/ws/admin	clusterBdd-.oracle.com_Dgraph

Exporting search results from the Dgraph target page to a file

You can export the results of your search to a file in Microsoft Office Excel.

To export to file, in the Dgraph target home page, enter the values for search and click the **Search** button, then click **Export to File**.

Information about the Dgraph HDFS Agent

The Dgraph target contains a region for information about the Dgraph HDFS Agent. You can search export activities, search ingest activities, and export searched results.

Searching export activities

In addition to the basic set of metrics for HDFS Agent, the Dgraph target also provides a command, **HDFS Agent Activity**. Clicking *Dgraph_target_name*>**HDFS Agent Activity** displays a table indicating the start and end time of each operation for exporting to HDFS, and the destination where the export file was written.

For example:

The screenshot shows a table titled "HDFS Agent Export Activity". At the top, there are search filters: Start Time (Dec 31, 2014 2:41:28 PM), End Time (Jan 15, 2015 2:41:28 PM), Duration (>=ms) (2000), and Record Count (50). There are "Search" and "Export to File" buttons. Below the filters is a table with the following columns: End Time, Duration (ms), Hive Table, Destination HDFS File, and Start Time.

Searching ingest activities

In the plug-in, you can use a **Search** feature to search recent Ingest activities that have been run by the HDFS Agent.

For example, you can search by date range, to list all the ingest activities during that time period.

Exporting searched results

You can export the results of your search to a file in Microsoft Office Excel, by using **Export to File**.

The Studio target

A Studio target represents a single Studio node running on a host. The **Studios** folder page provides a table that lists all Studio nodes and each node's status.

For example:

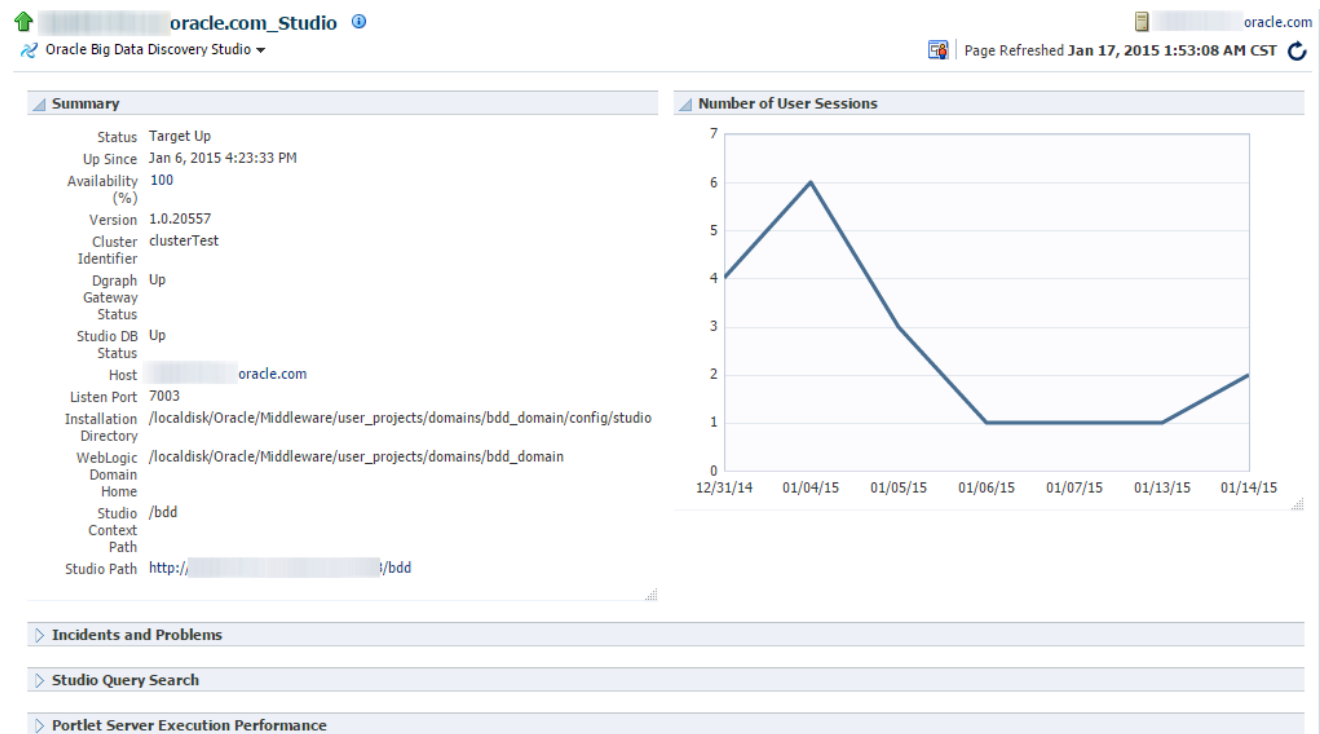
The screenshot shows the "Studios" target page. On the left is a "Target Navigation" tree with "clusterTest" expanded to show "Dgraphs" and "Studios". The main area displays a table with the following columns: Host, Name, Type, and Status. The table contains one row: Host (.oracle.com), Name (.oracle.com_Studio), Type (Oracle Big Data Discovery Studio), and Status (indicated by a green up arrow). The page is titled "Page Refreshed Jan 16, 2015 12:26:05 PM PST".

Clicking a Studio target in the table displays that Studio's home page and provides the following regions that describe the Studio node:

- A **Summary** region lists the basics about target status, availability, installation paths, server status, and so on.

- A **Number of User Sessions** region is about the total number of unique user sessions per day for the last 30 days.
- An **Incidents and Problems** region is standard to all Enterprise Manager plug-ins. This region allows you to search, view, manage, and resolve incidents and problems impacting your environment.
- A **Studio Query Search** region lets you search all Studio queries for a range of dates and identify any queries that took longer than the number of specified milliseconds. You can also export your search results to a file in Excel.
- A **Portlet Server Execution Performance** region displays running time for each portlet in Studio. (A portlet is another name for a component on a page in Studio).

For example:



Tracking performance of Studio components

Clicking the **Portlet Server Execution Performance** region displays running times for each portlet (component) in Studio.

For each portlet in Studio, the table tracks:

- Total number of queries or runs
- Total running time
- Average running time
- Maximum running time

For example:

Portlet Server Execution Performance				
Name	Count	Total Time (ms)	Avg Time (ms)	Max Time (ms)
endecaavailablerefinementsportlet	9	1928	214	725
endecabulkexportportlet	2	222	111	181
endecacatalognavigationportlet	8	788	98	264
endecacatalogresultslistportlet	6	951	158	563
endecachartportlet	4	1966	491	1201
endecadatasetsummaryportlet	3	213	71	103
endecafooterportlet	2	234	117	166
endecamultiselectqueueportlet	6	4820	803	3913
endecareultstableportlet	7	1723	246	502
endecasearchboxportlet	6	557	92	267
endecaselectedrefinementsportlet	3	640	213	406

Searching Studio queries

Clicking the **Studio Query Search** region lets you search all Studio queries for a range of dates and identify any queries that took longer than the number of specified milliseconds. For example:

Studio Query Search

Start Date: Dec 31, 2014 2:16:22 P | End Date: Jan 15, 2015 2:16:22 P | Duration (>=ms): 2000 | Record Count: 50

Buttons: Search, Export to File

Timestamp	Duration (ms)	Type	Request ID	Session Id
[Table content is obscured by a grey bar]				

Exporting search results from the Studio target page to a file

You can export the results of your search to a file in Microsoft Office Excel.

To export to file, in the Studio target home page, enter the values for search and click the **Search** button, then click **Export to File**.

Roles and privileges for BDD targets

Only users with sufficient roles and permissions can perform operations, such as Start and Stop, on the targets for Big Data Discovery.

The Enterprise Manager Cloud Control lets you use various roles and privileges, when working with targets that you are managing. For information on how to create, grant and use roles and privileges, see the *Oracle Enterprise Manager Cloud Control Security Guide*.

To make sure the BDD plug-in lets you perform the tasks you need for the Big Data Discovery, you, as the administrator of Enterprise Manager, must:

- Have a super user privilege, or

- Have both of these privileges:
 - "Operator" privilege for the host target on nodes where the Big Data Discovery product is installed. (This is because host commands are used by the Enterprise Manager, to operate on BDD targets.)
 - "Create job" privilege for the host targets on nodes where the Dgraph is installed. (This is because Enterprise Manager jobs are used to start and stop the Dgraph.)

Security credentials for BDD targets

The plug-in is configured in a way that lets it discover and work with targets that are deployed in either secure mode (SSL), or non-secure mode. The plug-in also relies on preferred credentials for two of the Big Data Discovery targets — Dgraph and Studio, and sets them automatically at discovery time to the host credential. Many features of the plug-in depend on these credentials being set.

About security credentials in the plug-in

By default, the preferred credentials in the plug-in are set for two targets — Dgraph and Studio, and they are set with a host credential. It is important to have these credentials set so that the plug-in behaves correctly.

You can always set your own credentials. To do so, go to **Setup>Security>Preferred Credentials**, and on the **Security** page, select the target, and click **Manage Preferred Credentials**. For complete details about providing preferred credentials during the target discovery process, see the topic "Setting preferred credentials for a target", http://docs.oracle.com/cd/E24628_01/timesten.121/e28645/install.htm#TTEMP604, in the *Enterprise Manager System Monitoring Plug-in for Oracle TimesTen In-Memory Database User's Guide*.

If the preferred credentials are not set

The following table list instances where the plug-in does not behave as expected, if the default credentials are not used or if you have not set your own credentials:

Target	Feature	Behavior if credential is not set
Dgraph	On the Cluster target home page, indicate which Dgraph node is the leader node.	The leader node is not identified.
Dgraph	On the Dgraph target home page, indicate the leader node and the status of the HDFS Agent.	The leader node is not identified, and the status of HDFS Agent is not listed (empty).
Dgraph	Dgraph request search.	An error is issued for the Dgraph target: Default preferred credentials are not set.
Dgraph	Dgraph administration operation	For each operation, an error is issued for the Dgraph target: Default preferred credentials are not set.
Dgraph	Start up and shut down the Dgraph	An error is issued for the Dgraph target: Default preferred credentials are not set.

Target	Feature	Behavior if credential is not set
Dgraph	View log messages for the Dgraph target.	The function is disabled without any warning.
Studio	On the Studio target home page, monitor the Dgraph Gateway status	The status of the Dgraph Gateway is empty.
Dgraph, Studio	View log messages for Cluster target.	The function is disabled without any warning.
Studio	View log messages for Studio target.	The function is disabled without any warning.

Connecting to Studio over a secure port

When Studio's outward-facing port is configured securely (with reverse proxy), the plug-in uses the same port to also connect to Studio in secure mode.

To establish a secure connection, you need to provide the plug-in with the SSL Keystore/Trust Keystore information.

Before installing Big Data Discovery, you can configure options in `bdd.conf` for a secure installation of Studio within WebLogic Server. For information, see the *Installation and Deployment Guide*. That guide also describes how you can set up Studio to use a reverse proxy, after the installation.

If Studio is not deployed securely, the Guided Discovery process of the plug-in determines this fact and does not let you provide SSL information during target discovery process in the plug-in. In such cases, to make sure that the plug-in uses the secure port for Studio in BDD 1.0, you should manually provide information about SSL for a Studio target in the plug-in by setting the monitoring credentials. For information on setting monitoring credentials within Enterprise Manager plug-in, see https://docs.oracle.com/cd/E24628_01/doc.121/e36415/sec_features.htm#sthref208 in the *Enterprise Manager Cloud Control Security Guide*.

When Studio is configured with SSL, the following requirements apply for monitoring Studio with Enterprise Manager plug-in:

- If Studio is configured in one-way SSL mode, provide only the Trust Keystore to the plug-in.
- If Studio is configured in two-way SSL mode, provide both the Keystore and Trust Keystore to the plug-in. In addition, these two requirements apply to the two-way SSL mode:
 - The password of the private key must be the same as the password of the Keystore.
 - If the Keystore contains more than one key pair, the key pair you want to use must be listed first among all the keys in the Keystore.

Starting and stopping the Dgraph using Enterprise Manager

The Dgraph home page in Enterprise Manager has **Start Up** and **Shut Down** commands to manually start and stop the Dgraph and the Dgraph HDFS Agent as necessary.



Note: The Dgraph process manages the Dgraph HDFS Agent process, so starting or stopping the Dgraph also starts or stops the HDFS Agent.

You can start up, and shut down one Dgraph instance at a time.

The start and stop commands run as jobs in Enterprise Manager that you can monitor. For example:

The screenshot shows the Enterprise Manager interface for the Dgraph target. At the top, there is a breadcrumb trail: **Oracle Big Data Discovery Dgraph** with a dropdown arrow, followed by **Start Up** (green play button) and **Shut Down** (red stop button). Below this, a job definition card is visible for **Big Data Discovery - Start Dgraph**. It includes a description: "Click Submit to start up both Dgraph and HDFS Agent." and a **Submit** button. A note below the button states: "Important: Ensure that the preferred credentials for the Dgraph have been set with Host Credentials."

Generally, the commands run quickly and return a SUCCEEDED or FAILED status.

Logging for Big Data Discovery targets in Enterprise Manager

Enterprise Manager provides standard logging controls to list, view, search, and download the log files for the Big Data Discovery targets. You can also group logging messages from each target by host, host IP address, or other parameters.

Viewing log messages

For the Cluster target, to view and search logs for all targets (Cluster, Dgraph, and Studio), select **<target name>>Logs>View Log Message** in Enterprise Manager.

For the Dgraph or Studio target, to view and search logs, use the same command for each target.

The **Log Messages** page displays the standard logging controls for any target in Enterprise Manager. For example:

The screenshot shows the **Log Messages** page. It features a search section with the following controls:

- Search Mode:** Radio buttons for **Online Logs** (selected), **Archive Logs**, and **Both**.
- Date Range:** A dropdown menu set to **Most Recent** and a text input set to **10** **Minutes**.
- Message Types:** Checkboxes for **Incident Error** (checked), **Error** (checked), **Warning**, **Notification**, **Trace**, and **Unknown** (checked).
- Search:** Radio buttons for **Selected Fields** (selected) and **All Fields**.

 Below the search controls is a **Search** button. At the bottom, there is a table with columns: **Time**, **Message Type**, **Message ID**, **Message**, **Execution Context** (with sub-columns **ECID** and **Relationship ID**), **Archive**, and **Log File**. The table currently displays the message: "(No messages matched the search criteria.)"

From this page, you can click **Target Log Files...** to view a list of the logs for the target and download the logs if desired. For example:

Log Files

Name	Directory
dgraph.reqlog	/localdisk/Oracle/Middleware/BDD1.0/dgraph/bin/
dgraph-l	.ebb /localdisk/Oracle/Middleware/BDD1.0/dgraph/bin/
dgraph.reqlog.18543.0	/localdisk/Oracle/Middleware/BDD1.0/dgraph/bin/
dgraph-	.ebb /localdisk/Oracle/Middleware/BDD1.0/dgraph/bin/
dgraph-	.ebb /localdisk/Oracle/Middleware/BDD1.0/dgraph/bin/
dgraph-	.ebb /localdisk/Oracle/Middleware/BDD1.0/dgraph/bin/
dgraph.reqlog.11083.1	/localdisk/Oracle/Middleware/BDD1.0/dgraph/bin/
dgraph-	.ebb /localdisk/Oracle/Middleware/BDD1.0/dgraph/bin/
dgraph.reqlog.11083.0	/localdisk/Oracle/Middleware/BDD1.0/dgraph/bin/
dgraph-	.ebb /localdisk/Oracle/Middleware/BDD1.0/dgraph/bin/
dgraph-	.ebb /localdisk/Oracle/Middleware/BDD1.0/dgraph/bin/
dgraph-	.ebb /localdisk/Oracle/Middleware/BDD1.0/dgraph/bin/
dgraph.out	/localdisk/Oracle/Middleware/BDD1.0/logs/
dgraphHDFSAGENT.out	/localdisk/Oracle/Middleware/BDD1.0/logs/

Grouping messages in the log viewer

For all Big Data Discovery targets, you can group log messages.

To group messages by a parameter, such as by host, in the **Log Messages** page, select a parameter from the drop-down menu. For example:

The screenshot shows the Oracle Enterprise Manager interface for 'Log Messages'. On the left is the 'Target Navigation' tree. The main area shows search filters for 'Log Messages' with options for 'Search Mode' (Online Logs, Archive Logs, Both), 'Date Range' (Time Interval, Start Date, End Date), and 'Message Types' (Incident Error, Error, Warning, Notification, Trace, Unknown). Below the filters is a table with columns for 'Time', 'Messages', 'Message ID', and 'Message'. A dropdown menu is open over the 'Messages' column, listing various grouping options. The option 'Group by Host' is highlighted in blue and circled in red.

Configuring verbose logging for a Dgraph target

The Dgraph has logging enabled by default. The Dgraph target uses additional logging variables to enable or disable *verbose* logging. You can enable verbose logging for a Dgraph node, as a whole, or for a limited set of search features that are run by the Dgraph process on a node.

The logging variables apply to a single Dgraph, and not to all Dgraph nodes in a cluster.

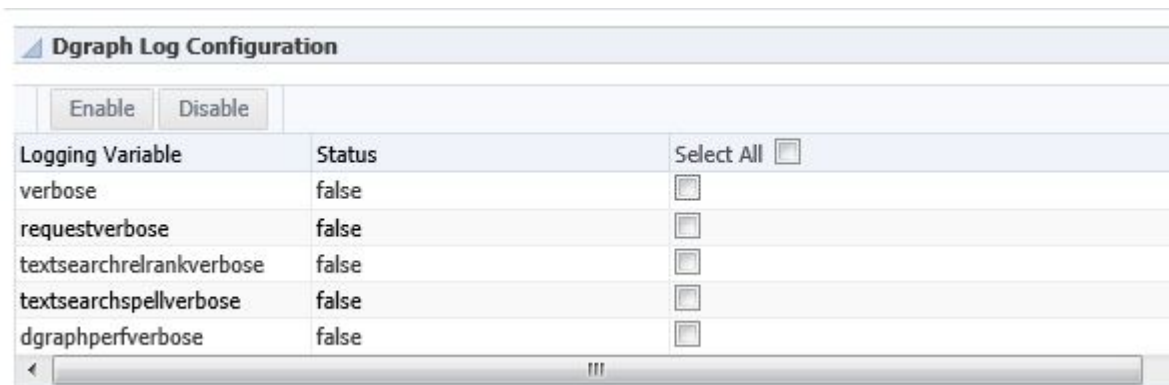
For each Dgraph target, you can set the following logging variables:

Variable	Description
<code>verbose</code>	Enables verbose logging for a Dgraph node.
<code>requestverbose</code>	Prints verbose logging about each query request to <code>stdout</code> .
<code>textsearchrelrankverbose</code>	Enables verbose logging about relevance ranking during query processing.
<code>textsearchspellverbose</code>	Enables verbose logging for spelling correction features.
<code>dgraphperfverbose</code>	Enables verbose logging for performance debugging messages during core Dgraph navigation computations.

To configure verbose logging for a Dgraph target:

1. Log in to Enterprise Manager Cloud Control.
2. Select a Dgraph target.
3. From the Dgraph target menu, select **Administration > Log Configuration**.

The following options display:



4. Select a logging variable for the Dgraph, or for the search feature as described above.

Once selected, the **Enable** and **Disable** options become available.

5. Click **Enable**.

After enabling a logging variable, its status changes to `true`.

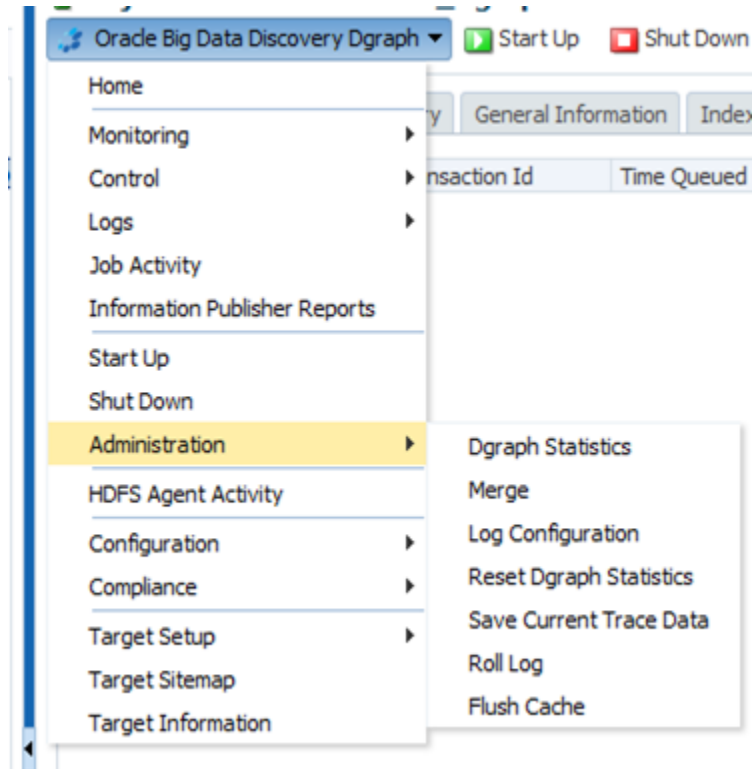
The change takes effect immediately. You do not need to restart the Dgraph.

Dgraph Administration Operations in Enterprise Manager

This section describes the **Administration** menu of options for the Dgraph target in the Enterprise Manager.

The **Administration** menu of the plug-in contains operations for viewing and resetting the statistics page for the Dgraph, as well as operations for saving and downloading Dgraph Tracing Utility data, flushing the cache, rolling the logs, and merging the index files for the Dgraph.

You can choose any operation by clicking the Dgraph target, and selecting **Administration**:



[Viewing Dgraph statistics](#)

[Resetting Dgraph statistics](#)

[Flushing the Dgraph cache](#)

[Merging index updates in the Dgraph](#)

[Rolling the Dgraph request log](#)

[Saving trace data for the Dgraph](#)

[Downloading Dgraph trace files](#)

Viewing Dgraph statistics

You can view Dgraph statistics for any Dgraph managed by Enterprise Manger Cloud Control.

To view Dgraph statistics:

1. Log in to Enterprise Manager Cloud Control.
2. Select a Dgraph target.
3. From the Dgraph target menu, select **Administration>Dgraph Statistics**.

A page with six tabs displays. For example:

The screenshot shows the 'Details' tab of the Dgraph Statistics page. It contains two main sections: 'Most Expensive Queries' and 'Hotspots'.

Most Expensive Queries

- Query1:20717602.00ms - "/ws/conversation:455"
- Query2:87209.12ms - "/ws/conversation:452"
- Query3:65509.93ms - "/ws/conversation:459"
- Query4:44102.00ms - "/ws/conversation:464"
- Query5:30102.78ms - "/ws/admin:456"
- Query6:25560.62ms - "/ws/conversation:476"
- Query7:14753.66ms - "/ws/eqd_parser:458"
- Query8:14115.50ms - "/ws/admin:521"
- Query9:13259.07ms - "/ws/conversation:443"
- Query10:9257.82ms - "/ws/conversation:477"

Hotspots

What	Num	Average	Standard Deviation	Min	Max	Total
Spell engine	0	nan ms	nan ms	nan ms	nan ms	0 ms
Page render	0	nan ms	nan ms	nan ms	nan ms	0 ms
Page render/record list	94	186.536 ms	792.18 ms	0.180176 ms	5208.11 ms	17534.4 ms
Record sort initialization	94	0.00211675 ms	0.000754736 ms	0.000976562 ms	0.00415039 ms	0.198975 ms
Query results sorting	0	nan ms	nan ms	nan ms	nan ms	0 ms
Prefetching horizontal records	127	119.244 ms	616.786 ms	0.0219727 ms	4392.98 ms	15144 ms
Heap sort	0	nan ms	nan ms	nan ms	nan ms	0 ms
Insertion sort time	0	nan ms	nan ms	nan ms	nan ms	0 ms
Preprocessing for comparison sorts	0	nan ms	nan ms	nan ms	nan ms	0 ms
Snipping	0	nan ms	nan ms	nan ms	nan ms	0 ms

Below the Hotspots table is a list of expandable sections:

- Results
- Server
- Navigation
- Record Sorting
- EVE Record Sorting
- Property Navigation
- Cache Warming
- Disk Usage
- Search
- Data Layer Performance

4. Click the tab and then region you want to examine. The statistics is intended for Oracle Support.

Resetting Dgraph statistics

You can reset Dgraph statistics for any Dgraph managed by Enterprise Manger Cloud Control plug-in.

This option runs against a single Dgraph and resets all statistics displayed on the Dgraph Statistics pages.

This option is useful if you want to view the statistics information for a single request: you reset statistics, issue a query, and inspect the updated statistics.

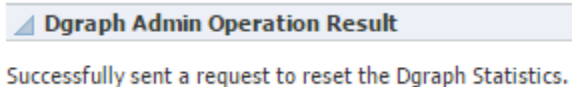
To reset the Dgraph statistics:

1. Log in to Enterprise Manager Cloud Control.
2. Select a Dgraph target.

3. From the Dgraph target menu, select **Administration>Reset Dgraph Statistics**.

A confirmation page displays.

After you confirm, the following status displays:



Flushing the Dgraph cache

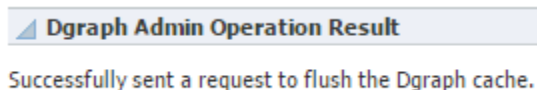
You can flush the cache of any Dgraph managed by Enterprise Manager.

This option runs against a single Dgraph. This option is useful if you are debugging query problems: you can approximate cold-start or post-update performance by clearing the Dgraph cache prior to running a request. To flush the Dgraph's cache:

1. Log in to Enterprise Manager Cloud Control.
2. Select a Dgraph target.
3. From the Dgraph target menu, select **Administration>Flush Cache**.

A confirmation page displays.

After you confirm, the following status displays:



Merging index updates in the Dgraph

You can force a merge of incremental updates into the index, or you can change the index merge policy of any Dgraph node managed by Enterprise Manger Cloud Control plug-in.

An index merge policy controls how the Dgraph manages its index files. A balanced index merge policy is used by default, and, in the majority of deployments, you do not need to change it. For information about index merging policy options and when to apply them, see [Managing an index merge policy on page 41](#).

This option runs against a single Dgraph. Additionally, the option must run against the leader node, or the operation fails. To identify the leader node, check the Cluster target page.

To merge updates to the index in the Dgraph:

1. Log in to Enterprise Manager Cloud Control.
2. Select a Dgraph target.

- From the Dgraph target menu, select **Administration>Merge**.

The following options display:

Dgraph Merge Operation

Operation Force Merge Change merge policy

Merge Policy Balanced Aggressive

Submit

- Select an operation and policy.
- Click **Submit**.

Rolling the Dgraph request log

You can roll the Dgraph request log over to a new file for any Dgraph managed by Enterprise Manger Cloud Control plug-in. This option runs against a single Dgraph.

To roll the Dgraph's request log:

- Log in to Enterprise Manager Cloud Control.
- Select a Dgraph target.
- From the Dgraph target menu, select **Administration>Roll Log**.

A confirmation page displays.

After you confirm, the following status displays:

Dgraph Admin Operation Result

Successfully rolled a log file for the Dgraph request log.

Saving trace data for the Dgraph

You can save trace-level information to a file for any Dgraph managed by Enterprise Manger Cloud Control plug-in.

The Dgraph Tracing Utility runs automatically while the Dgraph is running. It stores the Dgraph target trace data it collects in trace files (.ebb).

Saving the trace data for the Dgraph is useful when working with Oracle Support to debug and diagnose issues. This option runs against a single Dgraph. The output file is referred to as a "blackbox" file and is named `on-demand.pid.ebb`.

To save the Dgraph's trace data:

- Log in to Enterprise Manager Cloud Control.
- Select a Dgraph target.

- From the Dgraph target menu, select **Administration>Save Current Trace Data**.

A confirmation page displays.

After you confirm, the following status displays:



Unlike other types of log files, you can't view the Tracing Utility files in Enterprise Manager. Therefore, after you have saved the current trace data for the Dgraph, you may need to download it, to share with Oracle Support.

Downloading Dgraph trace files

The Dgraph Tracing Utility runs automatically while the Dgraph is running.

The Tracing Utility stores the Dgraph target trace data it collects in trace * .ebb files. The files are intended for use by Oracle Support. Unlike other types of log files, you cannot view the Tracing Utility files in Enterprise Manager. You can, however, download them from the **Log Messages** page.

To download a Tracing Utility file for the Dgraph target:

- Log in to Enterprise Manager Cloud Control and select a Dgraph target.
- From the Dgraph target menu, select **Logs>View Log Messages**.

This opens the **Log Messages** page.

- In the **Log Messages** page, click **Target Log Files...**

This opens the **Log Files** page, which lists the Dgraph target's log files. Tracing Utility files have the .ebb file extension and have a **Log Type** of **Trace**.

Log Files	
View ▾	View Log File Download
Name	
dgraph.reqlog	
dgraph.out	
dgraph-busgf1303-sigsegv-20140929-110816-212-0-pid26030.ebb	
dgraph-busgf1303-sigsegv-20140929-065301-631-0-pid24993.ebb	
dgraph-busgf1303-sigsegv-20140929-015053-059-0-pid1761.ebb	
DgraphHDFSAgent.out	
<div style="border: 1px solid #ccc; height: 15px; width: 100%;"></div>	
Rows Selected	1

- Click on a Tracing Utility file to select it, then click **Download**.

The selected Tracing Utility file is downloaded to your machine.

Part VII

Logging for Studio, Dgraph, and Dgraph Gateway



Chapter 18

Studio Logging

Studio logging helps you to monitor and troubleshoot your Studio application.

[About logging in Studio](#)

[About the log4j configuration XML files](#)

[About the main Studio log file](#)

[About the metrics log file](#)

[Configuring the amount of metrics data to record](#)

[Adjusting Studio logging levels](#)

[Using the Performance Metrics page to monitor query performance](#)

About logging in Studio

Studio uses the Apache log4j logging utility.

The Studio log files include:

- A main log file with most of the logging messages
- A second log file for performance metrics logging

The log files are generated in both the standard log4j format, and the ODL (Oracle Diagnostic Logging) format.

You can also use the **Performance Metrics** page of the **Control Panel** to view performance metrics information.

For more information about log4j, see the [Apache log4j site](#), which provides general information about and documentation for log4j.

About the log4j configuration XML files

The primary log configuration is managed in `portal-log4j.xml`, which is packed inside the portal application file `WEB-INF/lib/portal-impl.jar`.

The file is in the standard log4j XML configuration format, and allows you to:

- Create and modify appenders
- Bind appenders to loggers
- Adjust the log verbosity of different classes/packages

By default, `portal-log4j.xml` specifies a log verbosity of INFO for the following packages:

- `com.endeca`
- `com.endeca.portal.metadata`
- `com.endeca.portal.instrumentation`

It does not override any of the default log verbosity settings for other components.



Note: If you adjust the logging verbosity, it is updated for both log4j and the Java Utility Logging Implementation (JULI). Code using either of these loggers should respect this configuration.

About the main Studio log file

For Studio, the main log file (`bdd-studio.log`) contains all of the log messages.

By default the `bdd-studio.log` is stored in the WebLogic domain at `$MW_HOME/user_projects/domains/bdd_domain/<serverName>/logs` (where `serverName` is the name of the Managed Server in which Studio is installed).

The main root logger prints all messages to:

- The console, which typically is redirected to the application server's output log.
- `bdd-studio.log`, the log file in log4j format.
- `bdd-studio-odl.log`, the log file in ODL format. Also stored in `$MW_HOME/user_projects/domains/bdd_domain/logs`

The main logger does not print messages from the `com.endeca.portal.instrumentation` classes. Those messages are printed to the metrics log file.

About the metrics log file

Studio captures metrics logging, including all log entries from the `com.endeca.portal.instrumentation` classes.

The metrics log files are:

- `bdd-studio-metrics.log`, which is in log4j format.
- `bdd-studio-metrics-odl.log`, which is in ODL format.

Both metrics log files are created in the same directory as `bdd-studio.log`.

The metrics log file contains the following columns:

Column Name	Description
Total duration (msec)	The total time for this entry (End time minus Start time).
Start time (msec since epoch)	The time when this entry started. For Dgraph Gateway queries and server executions, uses the server's clock. For client executions, uses the client's clock.

Column Name	Description
End time (msec since epoch)	The time when this entry was finished. For Dgraph Gateway queries and server executions, uses the server's clock. For client executions, uses the client's clock.
Session ID	The session ID for the client.
Page ID	If client instrumentation is enabled, the number of full page refreshes or actions the user has performed. Used to help determine how long it takes to load a complete page. Some actions that do not affect the overall state of a page, such as displaying attributes on the Available Refinements panel, do not increment this counter.
Gesture ID	The full count of requests to the server.
Portlet ID	This is the ID associated with an individual instance of a component. It generally includes: <ul style="list-style-type: none"> • The type of component • A unique identifier For example, if a page includes two Chart components, the ID can be used to differentiate them.
Entry Type	The type of entry. For example: <ul style="list-style-type: none"> • PORTLET_RENDER - Server execution in response to a full refresh of a component • DISCOVERY_SERVICE_QUERY - Dgraph Gateway query • CONFIG_SERVICE_QUERY - Configuration service query • SCONFIG_SERVICE_QUERY - Semantic configuration service query • LQL_PARSER_SERVICE_QUERY - EQL parser service query • CLIENT - Client side JavaScript execution • PORTLET_RESOURCE - Server side request for resources • PORTLET_ACTION - Server side request for an action
Miscellaneous	A URL encoded JSON object containing miscellaneous information about the entry.

Configuring the amount of metrics data to record

To configure the metrics you want to include, you use a setting in `portal-ext.properties`. This setting applies to both the metrics log file and the **Performance Metrics** page.

The metrics logging can include:

- Queries by Dgraph nodes.
- Portlet server executions by component. The server side code is written in Java.
It handles configuration updates, configuration persistence, and Dgraph queries. The server-side code generates results to send back to the client-side code.
Server executions include component render, resource, and action requests.
- Component client executions for each component. The client-side code is hosted in the browser and is written in JavaScript. It issues requests to the server code, then renders the results as HTML. The client code also handles any dynamic events within the browser.

By default, only the Dgraph queries and component server executions are included.

You use the `df.performanceLogging` setting in `portal-ext.properties` to configure the metrics to include. The setting is:

```
df.performanceLogging=<metrics to include>
```

Where `<metrics to include>` is a comma-separated list of the metrics to include. The available values to include in the list are:

Value	Description
QUERY	If this value is included, then the page includes information for Dgraph queries.
PORTLET	If this value is included, then the page includes information on component server executions.
CLIENT	If this value is included, then the page includes information on component client executions.

In the default configuration, where only the Dgraph queries and component server executions are included, the value is:

```
df.performanceLogging=QUERY,PORTLET
```

To include all of the available metrics, you would add the `CLIENT` option:

```
df.performanceLogging=QUERY,PORTLET,CLIENT
```

Note that for performance reasons, this configuration is not recommended.

If you make the value empty, then the metrics log file and **Performance Metrics** page also are empty.

```
df.performanceLogging=
```

Adjusting Studio logging levels

For debugging purposes in a development environment, you can dynamically adjust logging levels for any class hierarchy.



Note: When you adjust the logging verbosity, it is updated for both `log4j` and the Java Utility Logging Implementation (JULI). Code using either of these loggers should respect this configuration.

Adjusting Studio logging levels:

1. In the Big Data Discovery header, click the Control Panel icon.
2. Choose **Server > Server Administration** .
3. Click the **Log Levels** tab.
4. On the **Update Categories** tab, locate the class hierarchy you want to modify.
5. From the logging level drop-down list, select the logging level.



Note: When you modify a class hierarchy, all classes that fall under that class hierarchy also are changed.

6. Click **Save**.

Using the Performance Metrics page to monitor query performance

The **Performance Metrics** page on the **Control Panel** displays information about component and Dgraph Gateway query performance.

It uses the same logging data that is recorded in the metrics log file.

However, unlike the metrics log file, the **Performance Metrics** page uses data stored in memory. Restarting Big Data Discovery clears the **Performance Metrics** data.

For each type of included metric, the table at the top of the page contains a collapsible section.

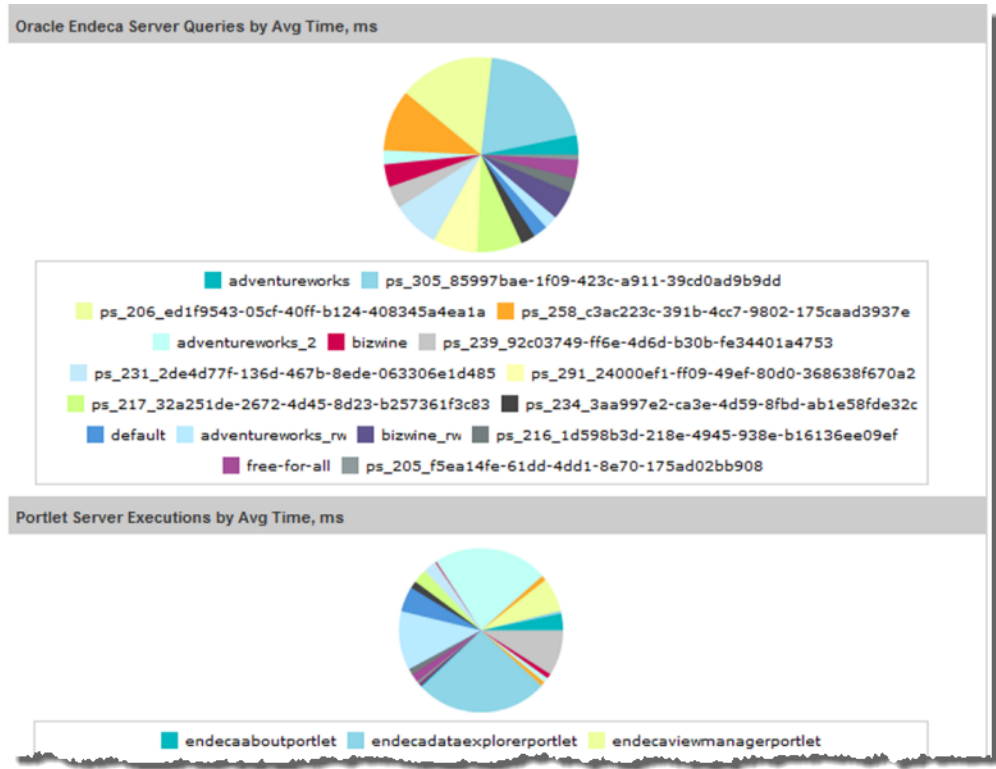
Performance Metrics

Performance Metrics				
Name ▲	Count	Total Time, ms	Avg Time, ms	Max Time, ms
▼ Oracle Endeca Server Queries				
adventureworks	28	6980	249	2603
adventureworks_2	40	7131	178	543
adventureworks_rw	928	159285	171	4840
bizwine	457	132479	289	2928
bizwine_rw	531	195181	367	4281
default	4111	734544	178	3245
free-for-all	268	63290	236	2184
ps_205_f5ea14fe-61d...	57	3814	66	649
ps_206_ed1f9543-05...	83	100603	1212	9567
ps_216_1d598b3d-21...	92	16810	182	3343
ps_217_32a251de-26...	1	574	574	574
ps_231_2de4d77f-13...	1	598	598	598
ps_234_3aa997e2-ca...	10	1860	186	1052
ps_239_92c03749-f6...	15	4264	284	1094

For each data source or component, the table tracks:

- Total number of queries or executions
- Total execution time
- Average execution time
- Maximum execution time

For each type of included metric, there is also a pie chart summarizing the average query or execution time per data source or component.



Note: Dgraph Gateway query performance does not correlate directly to a project page, as a single page often uses multiple Dgraph Gateway queries.



Chapter 19

Dgraph Logging

This section describes logging for the Dgraph.

[Dgraph request log and stdout/sterr log](#)

Dgraph request log and stdout/sterr log

Any Dgraph node creates two logs.

You can use these Dgraph logs to troubleshoot queries, or to track performance of particular queries or updates. The Dgraph logs are text files.

Collecting debugging information for the Dgraph

Before attempting to debug an issue with the Dgraph, collect the following information:

- A zip file containing the Dgraph request log and stdout/sterr log for the Dgraph instances.
- Hardware specifications and configuration.
- Description of the BDD cluster deployment. How many nodes are hosting Dgraph instances in your BDD deployment? Are Dgraph instances hosted on nodes that also host other components of BDD, or are the hosting machines dedicated to hosting only the Dgraph instances?
- The data from the Dgraph Statistics page. See [About Dgraph statistics on page 45](#).
- Description of which Dgraph instance is affected.

Dgraph request log

The Dgraph request log (also called the query log) contains one entry for each request processed. The requests are sorted by their timestamp.

The request log name and storage location is specified by the Dgraph `--log` flag. By default, the name and location of the log file is set to:

```
$BDD_HOME/dgraph/bin/dgraph.reqlog
```

The format of the Dgraph request log consists of fourteen fields, which contain the following information:

- Field 1: Timestamp (UNIX Time with milliseconds).
- Field 2: Client IP Address.
- Field 3: Outer Transaction ID, if defined.
- Field 4: Request ID.
- Field 5: Response Size (bytes).

- Field 6: Total Time (fractional milliseconds).
- Field 7: Processing Time (fractional milliseconds).
- Field 8: HTTP Response Code (0 on client disconnect).
- Field 9: - (unused).
- Field 10: Queue Status. On request arrival, the number of requests in queue (if positive) or the number of available slots at the same priority (if negative).
- Field 11: Thread ID.
- Field 12: HTTP URL (URL encoded).
- Field 13: HTTP POST Body (URL encoded; truncated to 64KBytes, by default; - if empty).
- Field 14: HTTP Headers (URL encoded).

Note that a dash (-) is entered for any field for which information is not available or pertinent.

By default, the Dgraph truncates the contents of the body for POST requests at 64K. This default setting saves disk space in the log, especially during the process of adding large numbers of records to the data domain. If you need to review the log for the full contents of the POST request body, contact Oracle support.

Dgraph stdout/stderr log

The Dgraph redirects its stdout/stderr output to the log file specified by the Dgraph `--out` flag. By default, the name and location of the file is:

```
$BDD_HOME/logs/dgraph.out
```

You can specify a new log location by changing the `DGRAPH_OUT_FILE` parameter in the `bdd.conf` file and then restarting the Dgraph.

The Dgraph stdout/stderr log includes startup messages as well as warning and error messages. You can increase the verbosity of the log via the Dgraph `-v` flag. You can also set the logging variables to toggle logging verbosity for specified features, which are described in [Configuring verbose logging for a Dgraph target on page 132](#).

Note that the Dgraph stdout/stderr log reports startup and shutdown times (and other informational messages) using the system's local time zone, with no zone label displayed, but displays warning and error messages in UTC.

Using grep on the Dgraph request log

When diagnosing performance issues, you can use `grep` with a distinctive string to find individual requests in the Dgraph request log. For example, you can use the string:

```
value%3D%22RefreshDate
```

If you have Studio, it is more useful to find the `X-Endeca-Portlet-Id` HTTP Header for the portlet sending the request, and `grep` for that. This is something like:

```
X-Endeca-Portlet-Id:
endecareultslistportlet_WAR_endecareultslistportlet_INSTANCE_5RKp_LAYOUT_11601
```

As an example, if you set:

```
PORTLET=endecareultslistportlet_WAR_endecareultslistportlet_INSTANCE_5RKp_LAYOUT_11601
```

then you can look at the times and response codes for the last ten requests from that portlet with a command such as:

```
grep $PORTLET Discovery.reqlog | tail -10 | cut -d ' ' -f 6,7,8
```

The command produces output similar to:

```
20.61 20.04 200
80.24 79.43 200
19.87 18.06 200
79.97 79.24 200
35.18 24.36 200
87.52 86.74 200
26.65 21.52 200
81.64 80.89 200
28.47 17.66 200
82.29 81.53 200
```

There are some other HTTP headers that can help tie requests together:

- `X-Endeca-Portlet-Id` — The unique ID of the portlet in the application.
- `X-Endeca-Session-Id` — The ID of the user session.
- `X-Endeca-Gesture-Id` — The ID of the end-user action (not filled in unless Studio has CLIENT logging enabled).
- `X-Endeca-Request-Id` — If multiple dgraph requests are sent for a single Dgraph Gateway request, they will all have the same `X-Endeca-Request-Id`.



Chapter 20

Dgraph Gateway Logging

This section describes the logging of the Dgraph Gateway process in the WebLogic Server domain.

[Dgraph Gateway logs](#)

[Dgraph Gateway log entry format](#)

[Dgraph Gateway log entry information](#)

[Logging configuration](#)

[Customizing the HTTP access log](#)

Dgraph Gateway logs

Dgraph Gateway uses the Apache Log4j logging utility for logging and its messages are written to WebLogic Server logs.

The BDD installation creates a WebLogic domain (**bdd_domain** is the default name), that has both an Admin Server and a Managed Server. The Admin Server is named **AdminServer** while the Managed Server has the same name as the host machine. Both the Dgraph Gateway and Studio are deployed into the Managed Server.

There are two sets of logs for the two different servers:

- Logs in the `$BDD_DOMAIN/servers/AdminServer/logs` directory are for the Admin Server.
- Logs in the `$BDD_DOMAIN/servers/<ServerName>/logs` directory are for the Managed Server.

There are three types of logs:

- WebLogic Domain Log
- WebLogic Server Log
- Application logs

Because all logs are text files, you can view their contents with a text editor. You can also view entries from the WebLogic Administration Console.

By default, these log files are located in the `$DOMAIN_HOME/servers/AdminServer/logs` directory (for the Admin Server) or one of the `$DOMAIN_HOME/servers/<serverName>/logs` directories (for a Managed Server). For example, if **bdd** is the name of your domain and **web004.us.example.com** is the name of the Managed Server, then the Managed Server logging path might be:

```
/localdisk/Oracle/Middleware/user_projects/domains/bdd/servers/web004.us.example.com/logs
```

Because all logs are text files, you can view their contents with a text editor. You can also view entries from the WebLogic Administration Console.

WebLogic Domain Log

The WebLogic domain log is generated only for the Admin Server. This domain log is intended to provide a central location from which to view the overall status of the domain.

The name of the domain log is:

```
$BDD_DOMAIN/servers/AdminServer/logs/bdd_domain.log
```

This assumes that `bdd_domain` is the name of the WebLogic domain.

The domain log is located in the `$DOMAIN_HOME/servers/AdminServer/logs` directory. For example, if `bdd` is the name of your domain, then the path of the log file might be:

```
/localdisk/Oracle/Middleware/user_projects/domains/bdd/servers/AdminServer/logs/bdd_domain.log
```

For more information on the WebLogic domain and server logs, see the "Server Log Files and Domain Log Files" topic in this page:

http://docs.oracle.com/cd/E24329_01/web.1211/e24428/logging_services.htm#WLLOG124

WebLogic Server Log

A WebLogic server log is generated for the Admin Server and for each Managed Server instance.

The default path of the Admin Server server log is:

```
$BDD_DOMAIN/servers/AdminServer/logs/AdminServer.log
```

The default path of the server log for a Managed Server is:

```
$BDD_DOMAIN/servers/<serverName>/logs/<serverName>.log
```

For example, if "web001.us.example.com" is the name of the Managed Server, then its server log is:

```
$BDD_DOMAIN/servers/web001.us.example.com/logs/web001.us.example.com.log
```

Application logs

Application logs are generated by the deployed applications. In this case, Dgraph Gateway and Studio are the applications.

For Dgraph Gateway, its application log is at:

```
$BDD_DOMAIN/servers/<serverName>/logs/<serverName>-diagnostic.log
```

For example, if "web001.us.example.com" is the name of the Managed Server, then the Dgraph Gateway application log is:

```
$BDD_DOMAIN/servers/web001.us.example.com/logs/web001.us.example.com-diagnostic.log
```

For Studio, its application log is at:

```
$BDD_DOMAIN/servers/<serverName>/logs/bdd-studio.log
```

For example, if "web001.us.example.com" is the name of the Managed Server, then its application log is:

```
$BDD_DOMAIN/servers/web001.us.example.com/logs/bdd-studio.log
```

The directory also stores other Studio metric log files, which are described in [About the metrics log file on page 140](#).

Logs to check when problems occur

For Dgraph Gateway problems, you should check the WebLogic server log for the Managed Server and the Dgraph Gateway application log:

```
$BDD_DOMAIN/servers/<serverName>/logs/<serverName>.log  
and  
$BDD_DOMAIN/servers/<serverName>/logs/<serverName>-diagnostic.log
```

For Studio issues, check the WebLogic server log for the Managed Server and the Dgraph Gateway application log:

```
$BDD_DOMAIN/servers/<serverName>/logs/<serverName>.log  
and  
$BDD_DOMAIN/servers/<serverName>/logs/bdd-studio.log
```

Dgraph Gateway log entry format

This topic describes the format of Dgraph Gateway log entries, including their message types and log levels.

The format of the Dgraph Gateway log fields are:

- Timestamp
- ComponentID
- Severity
- Message ID
- ClassName
- Host name
- Host IP
- Thread ID
- User ID
- ECID
- Message Text

The following is an example of an error message:

```
[2014-08-21T15:09:08.711+08:00] [EndecaServer] [ERROR] [OES-000091]  
[com.endeca.opmodel.ws.ControlServletContextListener] [host: YYZHU-CA] [nwaddr: 10.192.251.139]  
[tid: [ACTIVE].ExecuteThread: '24' for queue: 'weblogic.kernel.Default (self-tuning)'] [userId:  
YYZHU]  
[ecid: 0000KvrPS^C1FgUpM4^Aye1JxPgK000000,0] OES-000091: Could not find properties file:  
C:\WebLogic\Oracle\MIDDLE~1\USER_P~1\domains\BASE_D~1\config\EndecaServer.properties
```

All Dgraph Gateway log entries are prefixed with OES followed by the number and text of the message, as in this example:

```
OES-000135: Endeca Server has successfully initialized
```

The log entry fields (using the above example) and their descriptions are as follows:

Log entry field	Description	Example
Timestamp	The date and time when the message was generated. This reflects the local time zone.	[2014-08-21T15:09:08.711+08:00]
ComponentID	The ID of the component that originated the message. "EndecaServer" is hard-coded for the Dgraph Gateway.	[EndecaServer]
Severity	The type of message. Possible values are: ERROR, WARNING, NOTIFICATION, TRACE	[ERROR]
Message ID	The message ID that uniquely identifies the message within the component. The ID consists of the prefix OES (representing the component), followed by a dash, then a number.	[OES-000091]
ClassName	The Java class that prints the message entry.	[com.endeca.opmodel.ws.ControlServletContextListener]
Host name	The name of the host where the message originated.	[host: YYZHU-CA]
Host IP	The network address of the host where the message originated	[nwaddr: 10.192.251.139]
Thread ID	The ID of the thread that generated the message.	[tid: [ACTIVE].ExecuteThread: '24' for queue: 'weblogic.kernel.Default (self-tuning)']
User ID	The name of the user whose execution context generated the message.	[userId: YYZHU]
ECID	The Execution Context ID (ECID), which is a global unique identifier of the execution of a particular request in which the originating component participates.	[[ecid: 0000KvrPS^C1FgUpM4^Aye1JxPgK00000,0]
Message Text	The text of the error message.	OES-000091: Could not find properties file: ...]

Dgraph Gateway log entry information

This topic describes some of the information that is found in log entries.

For Dgraph Gateways in cluster-mode, this logged information can help you trace the life cycle of requests.

Note that all Dgraph Gateway ODL log entries are prefixed with `OES` followed by the number and text of the message, as in this example:

```
OES-000135: Endeca Server has successfully initialized
```

Logged request type and content

When a new request arrives at the server, the SOAP message in the request is analyzed. From the SOAP body, the request type of each request (such as `allocateBulkLoadPort`) is determined and logged. Complex requests (like `Conversation`) will be analyzed further, and detailed information will be logged as needed. Note that this information is logged if the log level is `DEBUG`.

For example, a `Conversation` request is sent to `Server1`. After being updated, the logs on the server might have entries such as these:

```
OES-000239: Receive request 512498665 of type 'Conversation'. This request does the
  following queries: [RecordCount, RecordList]
OES-000002: Timing event: start 512498665 ...
OES-000002: Timing event: DGraph start 512498665 ...
OES-000002: Timing event: DGraph end 512498665 ...
OES-000002: Timing event: end 512498665 ...
```

As shown in the example, when `Server1` receives a request, it will choose a node from the routing table and tunnel the request to that node. The routed request will be processed on that node. In the Dgraph request log, the request can also be tracked via the request ID in the HTTP header.

Log ingest timestamp and result

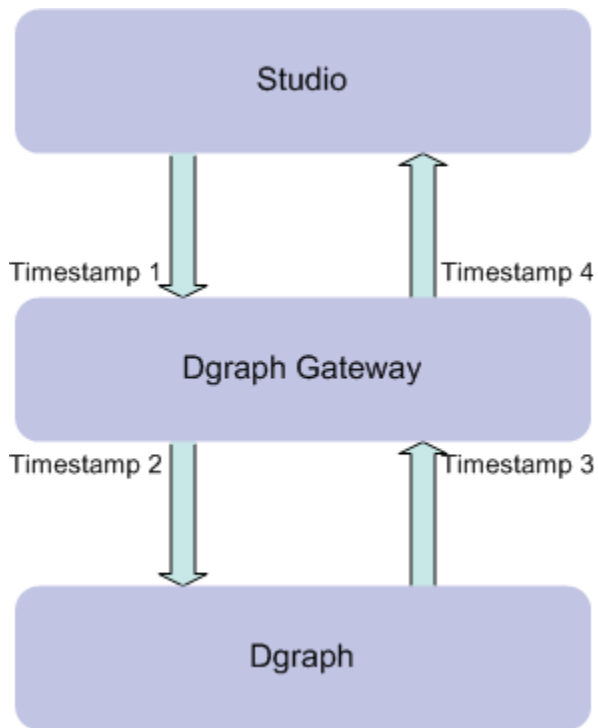
For ingest operations, a start and end timestamp is logged. At the end of the operation, the ingest results are also logged (number of added records, number of deleted records, number of updated records, number of replaced records, number of added or updated records).

Log entries would look like these examples:

```
OES-000002: Timing event: start ingest into Dgraph "http://host:7010"
OES-000002: Timing event: end ingest into Dgraph "http://
/host:7010" (1 added, 1 deleted, 0 replaced, 0 updated, 0 added or updated)
```

Total request and Dgraph processing times

Four calculated timestamps in the logs record the time points of a query as it moves from Studio to the Dgraph and back. The query path is shown in this illustration:



The four timestamps are:

1. Timestamp1: Dgraph Gateway begins to process the request from Studio
2. Timestamp2: Dgraph Gateway forwards the request to the Dgraph
3. Timestamp3: Dgraph Gateway receives the response from the Dgraph
4. Timestamp4: Dgraph Gateway finishes processing the request

To determine the total time cost of the request, the timestamp differences are calculated and logged:

- (Timestamp4 - Timestamp1) is the total request processing time in Dgraph Gateway.
- (Timestamp3 - Timestamp2) is the Dgraph processing time.

The log entries will look similar to these examples:

```
OES-000240: Total time cost(Request processing) of request 512498665 : 1717 ms
OES-000240: Total time cost(Dgraph processing) of request 512498665 : 424 ms
```

Logging configuration

Dgraph Gateway has a default Log4j configuration file that sets its logging properties.

The file is named `EndecaServerLog4j.properties` and is located in the `$DOMAIN_HOME/config` directory.

The default version of the file is as follows:

```
log4j.rootLogger=WARN, stdout, ODL

# Console Appender
log4j.appender.stdout=org.apache.log4j.ConsoleAppender
```

```

log4j.appender.stdout.layout=org.apache.log4j.PatternLayout
log4j.appender.stdout.layout.ConversionPattern=%d [%p] [%c] %L - %m%n

# ODL-format Log Appender
log4j.appender.ODL=com.endeca.util.ODLAppender
log4j.appender.ODL.MaxSize=104857600
log4j.appender.ODL.MaxSegmentSize=10485760
log4j.appender.ODL.encoding=UTF-8

# Log level per packages
log4j.logger.com.endeca=ERROR
log4j.logger.org.apache.zookeeper=WARN

```

The file defines two appenders (stdout and ODL) for the root logger and also sets log levels for two packages.

The file has the following properties:

Logging property	Description
log4j.rootLogger=WARN, stdout, ODL	The level of the root logger is defined as WARN and attaches the Console Appender (stdout) and ODL-format Log Appender (ODL) to it.
log4j.appender.stdout=org.apache.log4j.ConsoleAppender	Defines stdout as a Log4j ConsoleAppender
org.apache.log4j.PatternLayout	Sets the PatternLayout class for the stdout layout.
log4j.appender.stdout.layout.ConversionPattern	<p>Defines the log entry conversion pattern as:</p> <ul style="list-style-type: none"> • %d is the date of the logging event. • %p outputs the priority of the logging event. • %c outputs the category of the logging event. • %L outputs the line number from where the logging request was issued. • %m outputs the application-supplied message associated with the logging event while %n is the platform-dependent line separator character. <p>For other conversion characters, see: https://logging.apache.org/log4j/1.2/apidocs/org/apache/log4j/PatternLayout.html</p>
log4j.appender.ODL=com.endeca.util.ODLAppender	Defines ODL as an ODL Appender. ODL (Oracle Diagnostics Logging) is the logging format for Oracle applications.

Logging property	Description
<code>log4j.appender.ODL.MaxSize</code>	Sets the maximum amount of disk space to be used by the <code><ServerName>-diagnostic.log</code> file and the logging rollover files. The default is 104857600 (about 100 MB).
<code>log4j.appender.ODL.MaxSegmentSize</code>	Sets the maximum size (in bytes) of the log file. When the <code><ServerName>-diagnostic.log</code> file reaches this size, a rollover file is created. The default is 104857600 (about 10 MB).
<code>log4j.appender.ODL.encoding</code>	Sets character encoding the log file. The default UTF-8 value prints out UTF-8 characters in the file.
<code>log4j.logger.com.endeca</code>	Sets the default log level for the <code>Endeca</code> logger. <code>ERROR</code> is the default log level.
<code>log4j.logger.org.apache.zookeeper</code>	Sets the default log level for the <code>ZooKeeper</code> logger. <code>WARN</code> is the default log level.

For details on the FileHandler settings, see <http://docs.oracle.com/javase/7/docs/api/java/util/logging/FileHandler.html>

Logging levels

The logging level specifies the amount of information that is logged. This example shows how you can change a log level setting:

```
log4j.logger.com.endeca=INFO
```

In the example, the log level for the `Endeca` logger is set to `INFO`.

The log levels (in decreasing order of severity) are:

- `OFF` — Has the highest possible rank and is used to turn off logging.
- `FATAL` — Indicates a serious failure. In general, `FATAL` messages describe events that are of considerable importance and which will prevent normal program execution.
- `ERROR` — Indicates a serious problem that requires immediate attention from the administrator and is not caused by a bug in the product.
- `WARN` — Indicates a potential problem. In general, `WARN` messages describe events that should be reviewed by the administrator.
- `INFO` — A message level for informational messages. The `INFO` level typically indicates a major lifecycle event such as the activation or deactivation of a primary sub-component or feature.
- `DEBUG` — Debug information for events that are meaningful to administrators, such as public API entry or exit points.

These levels allow you to monitor events of interest at the appropriate granularity without being overwhelmed by messages that are not relevant. When you are initially setting up your application in a development environment, you might want to use the DEBUG level to get all messages, and change to a less verbose level in production.

Note that restarting the Dgraph Gateway is required after the log properties file has been modified.

Customizing the HTTP access log

You can customize the format of the default HTTP access log.

By default, WebLogic Server keeps a log of all HTTP transactions in a text file. The file is named `access.log` and is located in the `$DOMAIN_HOME/servers/<ServerName>/logs` directory.

The log provides true timing information from WebLogic, in terms of how long each individual Dgraph Gateway request takes. This timing information can be important in troubleshooting a slow system.

Note that this setup needs to be done on a per-server basis. That is, in a clustered environment, this has to be done for the Admin Server and for every Managed Server. This is because the clone operation (done when installing a clustered environment) does not carry over access log configuration.

The default format for the file is the common log format, but you can change it to the extended log format, which allows you to specify the type and order of information recorded about each HTTP communication. This topic describes how to add the following identifiers to the file:

- `date` — Date on which transaction completed, field has type `<date>`, as defined in the W3C specification.
- `time` — Time at which transaction completed, field has type `<time>`, as defined in the W3C specification.
- `time-taken` — Time taken for transaction to complete in seconds, field has type `<fixed>`, as defined in the W3C specification.
- `cs-method` — The request method, for example GET or POST. This field has type `<name>`, as defined in the W3C specification.
- `cs-uri` — The full requested URI. This field has type `<uri>`, as defined in the W3C specification.
- `sc-status` — Status code of the response, for example (404) indicating a "File not found" status. This field has type `<integer>`, as defined in the W3C specification.

To customize the HTTP access log:

1. Log into the Administration Server console.
2. In the Change Center of the Administration Console, click **Lock & Edit**.
3. In the left pane of the Console, expand **Environment** and select **Servers**.
4. In the Servers table, click the **AdminServer** name.
5. In the Settings for AdminServer page, select **Logging>HTTP**.
6. On the Logging > HTTP page, make sure that the **HTTP access log file enabled** checkbox is checked.
7. Click **Advanced**.
8. In the Advanced pane:
 - (a) In the Format list box, select **Extended**.

(b) In the Extended Logging Format Fields, enter this space-delimited string:

```
date time time-taken cs-method cs-uri sc-status
```

9. Click **Save**.
10. In the Change Center of the Administration Console, click **Activate Changes**.
11. Stop and then restart WebLogic Server.

The following is an example of the configured HTTP access log with several log entries:

```
#Version:      1.0
#Fields:      date time time-taken cs-method cs-uri sc-status
#Software:    WebLogic
#Start-Date:  2013-10-22 15:23:40
2013-10-22 15:27:07 0.967 POST /endeca-server/ws/cluster 200
2013-10-22 16:23:35 0.219 GET /endeca-server/ws/conversation/sh?wsdl 200
2013-10-22 16:23:35 0.0 GET /favicon.ico 404
2013-10-22 16:24:14 0.031 GET /endeca-server/ws/conversation/sh?wsdl 200
2013-10-22 16:24:14 0.031 GET /endeca-server/ws/conversation/sh?XSD
=lql_parser_types.xsd 200
```

Note that all the queries were successful (status code of 200), except for the one with the 404 status code.

Index

A

- administrative tasks, overview of 11
- aggressive merge policy 42
- allocate-bulk-load-port command 62
- autostart of BDD components, enabling 33

B

- backup strategy 17
- balanced merge policy 42
- bdd-admin script 29
 - checking service status 35
 - configuration properties that can be modified 32
 - enable/disable autostart 33
 - refresh configuration 31
 - restarting services 35
 - starting services 34
 - stopping services 34
- bdd-admin script vs Enterprise Manager plug-in 13
- Big Data Discovery cluster 21

C

- Cluster target, Enterprise Manager 122
- core dump files, Dgraph 43

D

- Dgraph
 - about 38
 - appointing new leader 44
 - checking status with bdd-admin script 35
 - crash dump files 44
 - displaying version 63
 - downloading trace files 137
 - enabling autostart with bdd-admin script 33
 - enhanced availability 27
 - Enterprise Manager target 123
 - flags 49
 - flushing cache with Enterprise Manager 135
 - logging configuration 132
 - logs 146
 - merging updates with Enterprise Manager 135
 - modifying memory consumption 40
 - rolling logs with Enterprise Manager 136
 - saving trace data with Enterprise Manager 136
 - starting with bdd-admin script 34
 - starting with the EM plug-in 130
 - startup behavior 25
 - stopping with bdd-admin script 34
 - stopping with the EM plug-in 130
 - updates 25
 - verbose logging in Enterprise Manager 132

- Dgraph administrative operations 45
 - flush 47
 - in Enterprise Manager 133
 - log-disable 49
 - log-enable 49
 - logroll 48
 - log-status 48
 - merge 47
 - stats 48
 - statsreset 48
 - stickymerge 47
- Dgraph Gateway
 - command interface, global options 61
 - logging configuration 154
 - logs 149
 - overview 56
 - properties file 56
 - starting 58
 - stopping 58
- Dgraph Gateway commands
 - allocate-bulk-load-port 62
 - dump-session 63
 - global options 61
 - host option 62
 - list-compute-nodes 63
 - port option 62
 - root option 62
 - version 63
 - warm-cache command, Dgraph Gateway 64
- Dgraph HDFS Agent
 - checking status with bdd-admin script 35
 - enabling autostart with bdd-admin script 33
 - flags 54
 - information in Enterprise Manager 125
 - starting with bdd-admin script 34
 - stopping with bdd-admin script 34
- Dgraph node 22
- Dgraph Statistics page
 - about 45
 - resetting, in Enterprise Manager 134
 - viewing in Enterprise Manager 134
- Dgraph target 123
- dump-session command, Dgraph Gateway 63

E

- email notifications
 - Account Created Notification, configuring 87
 - Password Changed Notification, configuring 87
 - sender, configuring 87
 - server, configuring 86
- enhanced availability 26
- Enterprise Manager
 - about 121

- connecting to a secure Studio target 129
- logging 130
- roles and privileges for Big Data Discovery 127
- security 128

F

- failure
 - Dgraph node 27
 - WebLogic Server node 27
 - ZooKeeper 27
- follower node 23
- forcing a merge 42
- framework settings
 - configuring 69
 - list of 67

H

- Hadoop settings
 - configuring 74
 - list of 71
- HTTP access log 157

I

- incidents and problems
 - Dgraph, in Enterprise Manager 123
 - HDFS Agent, in Enterprise Manager 123
- incremental updates, merge policy for 42

L

- LDAP integration
 - password policy, configuring 110
 - preventing passwords from being stored 110
 - roles, assigning based on groups 111
 - server connection, configuring 105
 - settings, configuring 105
- leader Dgraph node 23
- list-compute-nodes command, Dgraph Gateway 63
- locales
 - configuring available 82
 - configuring the default 82
 - configuring user preferred 84
 - effect of selection 80
 - list of supported 80
 - locations where set 81
 - scenarios for determining 81
- logging
 - Enterprise Manager 130
 - list of available logs 15
 - log4j configuration files, about 139
 - main Studio log file 140
 - metrics data, configuring 142
 - metrics log file, about 140
 - Performance Metrics page 143
 - verbosity, adjusting from the Control Panel 143
- logs

- Dgraph 146
- Dgraph Gateway 149
- WebLogic HTTP access log 157

M

- memory consumption by the Dgraph 40
- merge policy
 - changing 42
 - forcing a merge 42
 - for incremental updates 42
 - types of 42

O

- Oracle MapViewer settings in Studio 67

P

- pages visibility type 90
- password policy
 - configuring 97
 - LDAP integration, updating for 110
- passwords
 - existing user, changing for 103
 - new user, setting for 102
 - password policy, configuring 97
- Performance Metrics page 143
- preferred credentials 128
- project roles, assigning to members 92
- projects
 - certifying 93
 - deleting 94
 - existing user, changing membership 103
 - making active or inactive 93
 - members, adding 91
 - members, removing 91
 - new user, assigning membership to 103
 - project roles, assigning to members 92
 - project type, configuring 89

R

- resource utilization
 - Dgraph 123
 - HDFS Agent 123
- reverse proxy in Studio 129
- roles
 - existing user, changing 103
 - groups, assigning to for LDAP 111
 - new user, assigning 102
 - project roles, editing 99
 - user roles, editing 99
 - user roles, list of 99
- roles and privileges Enterprise Manager 127
- routing of requests to Dgraph nodes 25

S

session affinity 25

single sign-on
See SSO

SSO

about 112
LDAP connection, configuring 116
OHS URL, testing 115
Oracle Access Manager settings, configuring in Big Data Discovery 117
overview of the integration process 112
portal-ext.properties, configuring 118
reverse proxy configuration, WebLogic Server 113
Webgate, registering with Oracle Access Manager 114

Studio

checking status with bdd-admin script 35
creating users 102
Data Processing settings 71
email configuration 86
enabling autostart with bdd-admin script 33
Enterprise Manager target 125
framework settings 67
locales 80
logging 139
starting with bdd-admin script 34
stopping with bdd-admin script 34

System Usage

sections, about 76
usage logs, adding entries 75
using 77

T**Tracing Utility**

about 39
downloading trace files 137

saving trace data 136

troubleshooting

Dgraph, in Enterprise Manager 123
preferred credentials in Enterprise Manager 128

U**users**

authentication settings, configuring 96
creating 102
deactivating 104
deleting 104
editing 103
email addresses, listing restricted 98
reactivating 104
screen names, listing restricted 98

V

verbose logging for Dgraph 132
version command, Dgraph Gateway 63
visibility type, configuring for a page 90

W

warm-cache command, Dgraph Gateway 64
warming Dgraph cache 64
WebLogic logs
AdminServer 149
HTTP access log 157
WebLogic Server node failure 27

Z**ZooKeeper**

about 26
requirements 27