Informatica

Informatica® Big Data Management
10.1.1 HotFix 1

Installation and Upgrade Guide

Informatica Big Data Management Installation and Upgrade Guide
10.1.1 HotFix 1
October 2017

© Copyright Informatica LLC 2014, 2018

This software and documentation are provided only under a separate license agreement containing restrictions on use and disclosure. No part of this document may be reproduced or transmitted in any form, by any means (electronic, photocopying, recording or otherwise) without prior consent of Informatica LLC.

Informatica, the Informatica logo, PowerExchange, and Big Data Management are trademarks or registered trademarks of Informatica LLC in the United States and many jurisdictions throughout the world. A current list of Informatica trademarks is available on the web at https://www.informatica.com/trademarks.html. Other company and product names may be trade names or trademarks of their respective owners.

Portions of this software and/or documentation are subject to copyright held by third parties, including without limitation: Copyright DataDirect Technologies. All rights reserved. Copyright © Sun Microsystems. All rights reserved. Copyright © RSA Security Inc. All Rights Reserved. Copyright © Ordinal Technology Corp. All rights reserved. Copyright © Aandacht c.v. All rights reserved. Copyright Genivia, Inc. All rights reserved. Copyright Isomorphic Software. All rights reserved. Copyright © Meta Integration Technology, Inc. All rights reserved. Copyright © Intalio. All rights reserved. Copyright © Oracle. All rights reserved. Copyright © Adobe Systems Incorporated. All rights reserved. Copyright © DataArt, Inc. All rights reserved. Copyright © ComponentSource. All rights reserved. Copyright © Microsoft Corporation. All rights reserved. Copyright © Rogue Wave Software, Inc. All rights reserved. Copyright © Teradata Corporation. All rights reserved. Copyright © Yahoo! Inc. All rights reserved. Copyright © Glyph & Cog, LLC. All rights reserved. Copyright © Thinkmap, Inc. All rights reserved. Copyright © Clearpace Software Limited. All rights reserved. Copyright © Information Builders, Inc. All rights reserved. Copyright © OSS Nokalva, Inc. All rights reserved. Copyright Edifecs, Inc. All rights reserved. Copyright Cleo Communications, Inc. All rights reserved. Copyright © International Organization for Standardization 1986. All rights reserved. Copyright © ej-technologies GmbH. All rights reserved. Copyright © Jaspersoft Corporation. All rights reserved. Copyright © International Business Machines Corporation. All rights reserved. Copyright © yWorks GmbH. All rights reserved. Copyright © Lucent Technologies. All rights reserved. Copyright © University of Toronto. All rights reserved. Copyright © Daniel Veillard. All rights reserved. Copyright © Unicode, Inc. Copyright IBM Corp. All rights reserved. Copyright © MicroQuill Software Publishing, Inc. All rights reserved. Copyright © PassMark Software Pty Ltd. All rights reserved. Copyright © LogiXML, Inc. All rights reserved. Copyright © 2003-2010 Lorenzi Davide, All rights reserved. Copyright © Red Hat, Inc. All rights reserved. Copyright © The Board of Trustees of the Leland Stanford Junior University. All rights reserved. Copyright © EMC Corporation. All rights reserved. Copyright © Flexera Software. All rights reserved. Copyright © Jinfonet Software. All rights reserved. Copyright © Apple Inc. All rights reserved. Copyright © Telerik Inc. All rights reserved. Copyright © BEA Systems. All rights reserved. Copyright © PDFlib GmbH. All rights reserved. Copyright © Orientation in Objects GmbH. All rights reserved. Copyright © Tanuki Software, Ltd. All rights reserved. Copyright © Ricebridge. All rights reserved. Copyright © Sencha, Inc. All rights reserved. Copyright © Scalable Systems, Inc. All rights reserved. Copyright © jQWidgets. All rights reserved. Copyright © Tableau Software, Inc. All rights reserved. Copyright© MaxMind, Inc. All Rights Reserved. Copyright © TMate Software s.r.o. All rights reserved. Copyright © MapR Technologies Inc. All rights reserved. Copyright © Amazon Corporate LLC. All rights reserved. Copyright © Highsoft. All rights reserved. Copyright © Python Software Foundation. All rights reserved. Copyright © BeOpen.com. All rights reserved. Copyright © CNRI. All rights reserved.

This product includes software developed by the Apache Software Foundation (http://www.apache.org/), and/or other software which is licensed under various versions of the Apache License (the "License"). You may obtain a copy of these Licenses at http://www.apache.org/licenses/. Unless required by applicable law or agreed to in writing, software distributed under these Licenses is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the Licenses for the specific language governing permissions and limitations under the Licenses.

This product includes software which was developed by Mozilla (http://www.mozilla.org/), software copyright The JBoss Group, LLC, all rights reserved; software copyright © 1999-2006 by Bruno Lowagie and Paulo Soares and other software which is licensed under various versions of the GNU Lesser General Public License Agreement, which may be found at http:// www.gnu.org/licenses/lgpl.html. The materials are provided free of charge by Informatica, "as-is", without warranty of any kind, either express or implied, including but not limited to the implied warranties of merchantability and fitness for a particular purpose.

The product includes ACE(TM) and TAO(TM) software copyrighted by Douglas C. Schmidt and his research group at Washington University, University of California, Irvine, and Vanderbilt University, Copyright (©) 1993-2006, all rights reserved.

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (copyright The OpenSSL Project. All Rights Reserved) and redistribution of this software is subject to terms available at http://www.openssl.org and http://www.openssl.org/source/license.html.

This product includes Curl software which is Copyright 1996-2013, Daniel Stenberg, <daniel@haxx.se>. All Rights Reserved. Permissions and limitations regarding this software are subject to terms available at http://curl.haxx.se/docs/copyright.html. Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

The product includes software copyright 2001-2005 (©) MetaStuff, Ltd. All Rights Reserved. Permissions and limitations regarding this software are subject to terms available at http://www.dom4j.org/ license.html.

The product includes software copyright © 2004-2007, The Dojo Foundation. All Rights Reserved. Permissions and limitations regarding this software are subject to terms available at http://dojotoolkit.org/license.

This product includes ICU software which is copyright International Business Machines Corporation and others. All rights reserved. Permissions and limitations regarding this software are subject to terms available at http://source.icu-project.org/repos/icu/icu/trunk/license.html.

This product includes software copyright © 1996-2006 Per Bothner. All rights reserved. Your right to use such materials is set forth in the license which may be found at http:// www.gnu.org/software/ kawa/Software-License.html.

This product includes OSSP UUID software which is Copyright © 2002 Ralf S. Engelschall, Copyright © 2002 The OSSP Project Copyright © 2002 Cable & Wireless Deutschland. Permissions and limitations regarding this software are subject to terms available at http://www.opensource.org/licenses/mit-license.php.

This product includes software developed by Boost (http://www.boost.org/) or under the Boost software license. Permissions and limitations regarding this software are subject to terms available at http:/ /www.boost.org/LICENSE_1_0.txt.

This product includes software copyright © 1997-2007 University of Cambridge. Permissions and limitations regarding this software are subject to terms available at http:// www.pcre.org/license.txt.

This product includes software copyright © 2007 The Eclipse Foundation. All Rights Reserved. Permissions and limitations regarding this software are subject to terms available at http:// www.eclipse.org/org/documents/epl-v10.php and at http://www.eclipse.org/org/documents/edl-v10.php.

This product includes software licensed under the terms at http://www.tcl.tk/software/tcltk/license.html, http://www.bosrup.com/web/overlib/?License, http:// www.stlport.org/doc/ license.html, http://asm.ow2.org/license.html, http://www.cryptix.org/LICENSE.TXT, http://hsqldb.org/web/hsqlLicense.html, http:// httpunit.sourceforge.net/doc/ license.html, http://jung.sourceforge.net/license.txt , http://www.gzip.org/zlib/zlib_license.html, http://www.openldap.org/software/ release/license.html, http://www.libssh2.org, http://slf4j.org/license.html, http://www.sente.ch/software/OpenSourceLicense.html, http://fusesource.com/downloads/ license-agreements/fuse-message-broker-v-5-3- license-agreement; http://antlr.org/license.html; http://aopalliance.sourceforge.net/; http://www.bouncycastle.org/ licence.html; http://www.jgraph.com/jgraphdownload.html; http://www.jcraft.com/jsch/LICENSE.txt; http://jotm.objectweb.org/bsd_license.html; . http://www.w3.org/ Consortium/Legal/2002/copyright-software-20021231; http://www.slf4j.org/license.html; http://nanoxml.sourceforge.net/orig/copyright.html; http://www.json.org/ license.html; http://forge.ow2.org/projects/javaservice/, http://www.postgresql.org/about/licence.html, http://www.sqlite.org/copyright.html, http://www.tcl.tk/ software/tcltk/license.html, http://www.jaxen.org/faq.html, http://www.jdom.org/docs/faq.html, http://www.slf4j.org/license.html; http://www.iodbc.org/dataspace/ iodbc/wiki/iODBC/License; http://www.keplerproject.org/md5/license.html; http://www.toedter.com/en/jcalendar/license.html; http://www.edankert.com/bounce/ index.html; http://www.net-snmp.org/about/license.html; http://www.openmdx.org/#FAQ; http://www.php.net/license/3_01.txt; http://srp.stanford.edu/license.txt;

http://www.schneier.com/blowfish.html; http://www.jmock.org/license.html; http://xsom.java.net; http://benalman.com/about/license/; https://github.com/CreateJS/EaselJS/blob/master/src/easeljs/display/Bitmap.js; http://www.h2database.com/html/license.html#summary; http://jsoncpp.sourceforge.net/LICENSE; http://jdbc.postgresql.org/license.html; http://protobuf.googlecode.com/svn/trunk/src/google/protobuf/descriptor.proto; https://github.com/rantav/hector/blob/master/LICENSE; http://web.mit.edu/Kerberos/krb5-current/doc/mitK5license.html; http://jibx.sourceforge.net/jibx-license.html; https://github.com/lyokato/libgeohash/blob/master/LICENSE; https://github.com/hjiang/jsonxx/blob/master/LICENSE; https://code.google.com/p/lz4/; https://github.com/jedisct1/libsodium/blob/master/LICENSE; http://one-jar.sourceforge.net/index.php?page=documents&file=license; https://github.com/EsotericSoftware/kryo/blob/master/license.txt; http://www.scala-lang.org/license.html; https://github.com/tinkerpop/blueprints/blob/master/LICENSE.txt; http://gee.cs.oswego.edu/dl/classes/EDU/oswego/cs/dl/util/concurrent/intro.html; https://aws.amazon.com/asl/; https://github.com/twbs/bootstrap/blob/master/LICENSE; https://sourceforge.net/p/xmlunit/code/HEAD/tree/trunk/LICENSE.txt; https://github.com/documentcloud/underscore-contrib/blob/master/LICENSE, and https://github.com/apache/hbase/blob/master/LICENSE.txt.

This product includes software licensed under the Academic Free License (http://www.opensource.org/licenses/afl-3.0.php), the Common Development and Distribution License (http://www.opensource.org/licenses/cddl1.php) the Common Public License (http://www.opensource.org/licenses/cpl1.0.php), the Sun Binary Code License Agreement Supplemental License Terms, the BSD License (http:// www.opensource.org/licenses/bsd-license.php), the new BSD License (http://opensource.org/licenses/BSD-3-Clause), the MIT License (http://www.opensource.org/licenses/mit-license.php), the Artistic License (http://www.opensource.org/licenses/artistic-license-1.0) and the Initial Developer's Public License Version 1.0 (http://www.firebirdsql.org/en/initial-developer-s-public-license-version-1-0/).

This product includes software copyright © 2003-2006 Joe Walnes, 2006-2007 XStream Committers. All rights reserved. Permissions and limitations regarding this software are subject to terms available at http://xstream.codehaus.org/license.html. This product includes software developed by the Indiana University Extreme! Lab. For further information please visit http://www.extreme.indiana.edu/.

This product includes software Copyright (c) 2013 Frank Balluffi and Markus Moeller. All rights reserved. Permissions and limitations regarding this software are subject to terms of the MIT license.

See patents at https://www.informatica.com/legal/patents.html.

DISCLAIMER: Informatica LLC provides this documentation "as is" without warranty of any kind, either express or implied, including, but not limited to, the implied warranties of noninfringement, merchantability, or use for a particular purpose. Informatica LLC does not warrant that this software or documentation is error free. The information provided in this software or documentation may include technical inaccuracies or typographical errors. The information in this software and documentation is subject to change at any time without notice.

NOTICES

This Informatica product (the "Software") includes certain drivers (the "DataDirect Drivers") from DataDirect Technologies, an operating company of Progress Software Corporation ("DataDirect") which are subject to the following terms and conditions:

1. THE DATADIRECT DRIVERS ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT.
2. IN NO EVENT WILL DATADIRECT OR ITS THIRD PARTY SUPPLIERS BE LIABLE TO THE END-USER CUSTOMER FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, CONSEQUENTIAL OR OTHER DAMAGES ARISING OUT OF THE USE OF THE ODBC DRIVERS, WHETHER OR NOT INFORMED OF THE POSSIBILITIES OF DAMAGES IN ADVANCE. THESE LIMITATIONS APPLY TO ALL CAUSES OF ACTION, INCLUDING, WITHOUT LIMITATION, BREACH OF CONTRACT, BREACH OF WARRANTY, NEGLIGENCE, STRICT LIABILITY, MISREPRESENTATION AND OTHER TORTS.

The information in this documentation is subject to change without notice. If you find any problems in this documentation, please report them to us in writing at Informatica LLC 2100 Seaport Blvd. Redwood City, CA 94063.

Informatica products are warranted according to the terms and conditions of the agreements under which they are provided. INFORMATICA PROVIDES THE INFORMATION IN THIS DOCUMENT "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING WITHOUT ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND ANY WARRANTY OR CONDITION OF NON-INFRINGEMENT.

Publication Date: 2018-06-09

Table of Contents

Preface. 12

Informatica Resources. 12

- Informatica Network. 12
- Informatica Knowledge Base. 12
- Informatica Documentation. 12
- Informatica Product Availability Matrixes. 12
- Informatica Velocity. 13
- Informatica Marketplace. 13
- Informatica Global Customer Support. 13

Part I: Pre-Installation. 14

Chapter 1: Introduction to Big Data Management 15

Installation and Upgrade Overview. 15

Big Data Management Component Architecture. 16

- Hadoop Environment. 16
- Clients and Tools. 17
- Application Services. 17
- Repositories. 17

Integration with Informatica Products. 18

- PowerExchange Adapters Connectivity. 18
- Data Lineage. 18
- Data Lakes. 19
- Address Validation. 20
- Data Discovery. 20
- Data Replication. 20
- Data Relationship Management. 20
- Data Streaming. 21

Chapter 2: Before You Install or Upgrade. 22

Before You Install or Upgrade Overview. 22

Read the Release Notes. 23

Uninstall Big Data Management. 23

Verify System Requirements. 23

- Verify Product Installations. 23
- Verify the Hadoop Distribution and Installation Package. 24
- Verify Installer Requirements. 24
- Verify Port Requirements. 25
- Verify Hortonworks HDP Requirements. 25
- Verify Azure HDInsight Requirements. 26

Create a Data Integration Service User for Hive . . . 26
Prepare the Hadoop Cluster for the Blaze Engine. . . . 26
Create a Blaze User Account. . . . 27
Create Blaze Engine Directories and Grant Permissions. . . . 27
Grant Permissions on the Hive Source Database. . . . 28
Create a Staging Directory on HDFS . . . 28
Record Configuration Information. . . . 28
Gather Domain Information. . . . 29
Gather Connection Information. . . . 29
Gather Information for Configuration Through SSH. . . . 30

Part II: Install and Configure for Amazon EMR. . . . 32

Chapter 3: Installation for Amazon EMR. . . . 33

Installation for Amazon EMR Overview. . . . 33
Download the Distribution Package. . . . 33
Install on a Single Node. . . . 34
Install on a Cluster Using SCP. . . . 34
Install on a Cluster Using NFS. . . . 35
Install on a Cluster Using the HadoopDataNodes File. . . . 36
Create a Cluster and Install Big Data Management. . . . 36
Create the Bootstrap Script. . . . 37
Run the Cluster Creation Wizard. . . . 38

Chapter 4: Configuration for Amazon EMR. . . . 39

Configure the Data Integration Service. . . . 39
Download the JDBC Drivers for Sqoop Connectivity. . . . 40
Copy .jar Files for Hive Access. . . . 40
Update Files on the Domain Environment for Amazon EMR. . . . 41
Update hadoopEnv.properties. . . . 41
Update hive-site.xml. . . . 43
Update yarn-site.xml. . . . 43
Update Files on the Hadoop Environment for Amazon EMR. . . . 44
Update core-site.xml. . . . 44
Update yarn-site.xml . . . 45
Set S3 Bucket Access Policies. . . . 46
Developer Tool Tasks for Amazon EMR. . . . 46
Create Connections. . . . 46
Update Files on the Developer Tool. . . . 47
Complete Upgrade Tasks. . . . 47
Update the hadoopEnv.properties File. . . . 48
Complete Connection Upgrade. . . . 48
Update Connection Properties. . . . 49

Replace the Connections. 49

Chapter 5: Uninstall for Amazon EMR. 50

Before You Uninstall. 50
Uninstall Big Data Management. 50

Part III: Install and Configure for Azure HDInsight. 52

Chapter 6: Installation for Azure HDInsight. 53

Installation for Azure HDInsight Overview. 53
Download the Distribution Package. 53
Install on a Single Node 54
Install on a Cluster Using SCP 54
Install on a Cluster Using NFS. 54
Install on a Cluster Using the HadoopDataNodes File. 55

Chapter 7: Hadoop Configuration Manager for Azure HDInsight. 56

Hadoop Configuration Manager Overview. 56
Start the Configuration Manager in Console Mode. 56
Complete the Configuration through Apache Ambari. 57
Complete the Configuration through SSH. 59
Run the Configuration Manager in Silent Mode. 61
Configure the Properties File. 61
Run the Configuration Manager. 65

Chapter 8: Manual Configuration Tasks for Azure HDInsight. 66

Update Files on the Hadoop Environment for Azure HDInsight. 66
Update core-site.xml. 66
Update yarn-site.xml. 67
Import Files for Data Decryption. 68
Update Files on the Domain Environment for Azure HDInsight. 69
Update hadoopEnv.properties. 69
Update hive-site.xml. 71
Create a Softlink for the core-site.xml, hdfs-site.xml and mapred-site.xml Files. 71
Update Files on the Developer Tool. 72
Download the JDBC Drivers for Sqoop Connectivity. 73
Complete Upgrade Tasks. 73
Update the hadoopEnv.properties File. 73
Replace the Connections. 74
Update Connection Properties. 74

Chapter 9: Uninstall for Azure HDInsight. 75

Before You Uninstall. 75

Uninstall Big Data Management. 75

Part IV: Install and Configure for Cloudera CDH. 77

Chapter 10: Installation for Cloudera CDH. 78

Download the Distribution Package. 78

Install Using Cloudera Manager. 78

Chapter 11: Hadoop Configuration Manager for Cloudera CDH. 80

Hadoop Configuration Manager Overview. 80

Start the Configuration Manager in Console Mode. 80

Complete the Configuration through Cloudera Manager. 81

Complete the Configuration through SSH. 83

Run the Configuration Manager in Silent Mode. 85

Configure the Properties File. 85

Run the Configuration Manager. 89

Chapter 12: Manual Configuration Tasks for Cloudera CDH. 91

Update Files on the Hadoop Environment for Cloudera CDH. 91

Update core-site.xml. 91

Update mapred-site.xml. 92

Update yarn-site.xml 93

Update Files on the Domain Environment for Cloudera CDH. 94

Update hadoopEnv.properties. 95

Update hive-site.xml. 96

Update yarn-site.xml 97

Creating and Configuring Security Certificates and Truststore Files. 98

Update Files on the Developer Tool. 99

Download the JDBC Drivers for Sqoop Connectivity. 100

Complete Upgrade Tasks. 101

Update the hadoopEnv.properties File. 101

Complete Connection Upgrade. 102

Update Connection Properties. 102

Replace the Connections. 102

Chapter 13: Uninstall for Cloudera CDH. 104

Uninstall Big Data Management. 104

Part V: Install and Configure for Hortonworks. 105

Chapter 14: Installation for Hortonworks HDP. 106

Installation for Hortonworks HDP Overview. 106

Download the Distribution Package. 106

Install in an Ambari Stack. 107
Troubleshooting Installation in an Ambari Stack. 110

Chapter 15: Hadoop Configuration Manager for Hortonworks HDP. 111

Hadoop Configuration Manager Overview. 111
Start the Configuration Manager in Console Mode. 111
Complete the Configuration through Apache Ambari. 112
Complete the Configuration through SSH. 114
Run the Configuration Manager in Silent Mode. 116
Configure the Properties File. 116
Run the Configuration Manager. 120

Chapter 16: Manual Configuration Tasks for Hortonworks HDP. 121

Download the JDBC Drivers for Sqoop Connectivity. 121
Update Files on the Hadoop Environment for Hortonworks HDP. 122
Update core-site.xml 122
Update hive-site.xml 123
Update yarn-site.xml. 123
Update Files on the Domain Environment for Hortonworks HDP. 125
Update hadoopEnv.properties. 125
Update hive-site.xml. 126
Update yarn-site.xml 127
Creating and Configuring Security Certificates and Truststore Files. 128
Update Files on the Developer Tool. 129
Complete Upgrade Tasks. 130
Update the hadoopEnv.properties File. 131
Complete Connection Upgrade. 131
Update Connection Properties. 132
Replace the Connections. 132

Chapter 17: Uninstall for Hortonworks HDP. 133

Uninstall Big Data Management. 133

Part VI: Install and Configure for IBM BigInsights. 135

Chapter 18: Installation for IBM BigInsights. 136

Installation for IBM BigInsights Overview. 136
Download the Distribution Package. 136
Install on a Single Node. 137
Install on a Cluster Using SCP. 137
Install on a Cluster Using NFS. 138
Install on a Cluster Using the HadoopDataNodes File. 139

Chapter 19: Hadoop Configuration Manager for IBM BigInsights. 140
Hadoop Configuration Manager Overview. 140
Start the Configuration Manager in Console Mode 140
Complete the Configuration through Apache Ambari. 141
Complete the Configuration through SSH. 143
Run the Configuration Manager in Silent Mode. 145
Configure the Properties File. 145
Run the Configuration Manager. 149

Chapter 20: Manual Configuration Tasks for IBM BigInsights. 150
Download the JDBC Drivers for Sqoop Connectivity. 150
Update Files on the Hadoop Environment for IBM BigInsights. 150
Update core-site.xml. 151
Update yarn-site.xml. 151
Update Files on the Domain Environment for IBM BigInsights. 152
Update hadoopEnv.properties. 153
Update hive-site.xml. 154
Update yarn-site.xml 155
Update Files on the Developer Tool. 155
Complete Upgrade Tasks. 156
Update the hadoopEnv.properties File. 156
Complete Connection Upgrade. 157
Update Connection Properties. 157
Replace the Connections. 158

Chapter 21: Uninstall for IBM BigInsights. 159
Before You Uninstall. 159
Uninstall Big Data Management. 159

Part VII: Install and Configure for MapR. 161

Chapter 22: Installation for MapR. 162
Installation for MapR Overview. 162
Download the Distribution Package. 162
Install on a Single Node. 163
Install on a Cluster Using SCP. 163
Install on a Cluster Using NFS. 164
Install on a Cluster Using the HadoopDataNodes File. 165

Chapter 23: Hadoop Configuration Manager for MapR. 166
Hadoop Configuration Manager Overview. 166
Run the Configuration Manager in Console Mode. 166

Chapter 24: Manual Configuration Tasks for MapR. 169

Download the JDBC Drivers for Sqoop Connectivity. 169
Update Configuration Files on the Domain Environment. 169
Update hadoopEnv.properties. 170
Update hive-site.xml. 171
Update yarn-site.xml 172
Update Files on the Hadoop Environment for MapR. 173
Update core-site.xml. 173
Update hive-site.xml 174
Update yarn-site.xml 174
Generate MapR Tickets. 175
Create the Developer Tool User. 176
Generate Tickets. 176
Configure Informatica Application Services. 176
Developer Tool Tasks. 179
Configure Files on the Developer Tool. 180
Complete Upgrade Tasks. 181
Update the hadoopEnv.properties File. 181
Complete Connection Upgrade. 182
Update Connection Properties. 182
Replace the Connections. 182

Chapter 25: Uninstall for MapR. 184

Before You Uninstall. 184
Uninstall Big Data Management. 184

Appendix A: Connection Property Reference. 186

Connection Property Reference Overview. 186
Hadoop Connection Properties. 187
HBase Connection Properties. 196
HDFS Connection Properties. 197
Hive Connection Properties. 198
JDBC Connection Properties. 205

Appendix B: Port Reference 208

Informatica Domain Port Reference. 208
Application Service Port Reference. 209
Analyst Service Port Reference. 209
Content Management Service Port Reference. 209
Data Integration Service Port Reference. 209
Metadata Manager Service Port Reference. 210
Hadoop Distribution Port Reference. 210

IBM BigInsights Port Reference. 210
Cloudera Port Reference. 211
HDInsight Port Reference. 212
Hortonworks Port Reference. 213
MapR Port Reference. 214
Hadoop Cluster Port Reference. 214
Blaze Engine Port Reference 215
Developer Tool Port Reference. 215

Index. 216

Preface

The *Informatica Big Data Management™ Installation and Upgrade Guide* is written for the system administrator who is responsible for installing or upgrading Informatica Big Data Management. Readers of this guide need to have knowledge of the Hadoop environment and distributions, operating systems, and the database engines in their environment.

Informatica Resources

Informatica Network

Informatica Network hosts Informatica Global Customer Support, the Informatica Knowledge Base, and other product resources. To access Informatica Network, visit https://network.informatica.com.

As a member, you can:

- Access all of your Informatica resources in one place.
- Search the Knowledge Base for product resources, including documentation, FAQs, and best practices.
- View product availability information.
- Review your support cases.
- Find your local Informatica User Group Network and collaborate with your peers.

Informatica Knowledge Base

Use the Informatica Knowledge Base to search Informatica Network for product resources such as documentation, how-to articles, best practices, and PAMs.

To access the Knowledge Base, visit https://kb.informatica.com. If you have questions, comments, or ideas about the Knowledge Base, contact the Informatica Knowledge Base team at KB_Feedback@informatica.com.

Informatica Documentation

To get the latest documentation for your product, browse the Informatica Knowledge Base at https://kb.informatica.com/_layouts/ProductDocumentation/Page/ProductDocumentSearch.aspx.

If you have questions, comments, or ideas about this documentation, contact the Informatica Documentation team through email at infa_documentation@informatica.com.

Informatica Product Availability Matrixes

Product Availability Matrixes (PAMs) indicate the versions of operating systems, databases, and other types of data sources and targets that a product release supports. If you are an Informatica Network member, you can access PAMs at https://network.informatica.com/community/informatica-network/product-availability-matrices.

Informatica Velocity

Informatica Velocity is a collection of tips and best practices developed by Informatica Professional Services. Developed from the real-world experience of hundreds of data management projects, Informatica Velocity represents the collective knowledge of our consultants who have worked with organizations from around the world to plan, develop, deploy, and maintain successful data management solutions.

If you are an Informatica Network member, you can access Informatica Velocity resources at http://velocity.informatica.com.

If you have questions, comments, or ideas about Informatica Velocity, contact Informatica Professional Services at ips@informatica.com.

Informatica Marketplace

The Informatica Marketplace is a forum where you can find solutions that augment, extend, or enhance your Informatica implementations. By leveraging any of the hundreds of solutions from Informatica developers and partners, you can improve your productivity and speed up time to implementation on your projects. You can access Informatica Marketplace at https://marketplace.informatica.com.

Informatica Global Customer Support

You can contact a Global Support Center by telephone or through Online Support on Informatica Network.

To find your local Informatica Global Customer Support telephone number, visit the Informatica website at the following link:
http://www.informatica.com/us/services-and-training/support-services/global-support-centers.

If you are an Informatica Network member, you can use Online Support at http://network.informatica.com.

Part I: Pre-Installation

This part contains the following chapters:

- Introduction to Big Data Management , 15
- Before You Install or Upgrade, 22

CHAPTER 1

Introduction to Big Data Management

This chapter includes the following topics:

- Installation and Upgrade Overview, 15
- Big Data Management Component Architecture, 16
- Integration with Informatica Products, 18

Installation and Upgrade Overview

The Big Data Management installer includes components that allow communication between the Informatica domain and the Hadoop environment. You can install Big Data Management in a single node environment or in a cluster environment.

If you upgraded the Informatica platform or applied the hotfix, you need to uninstall Big Data Management from the cluster and perform a new installation. You also perform the installation configuration tasks in addition to some upgrade tasks.

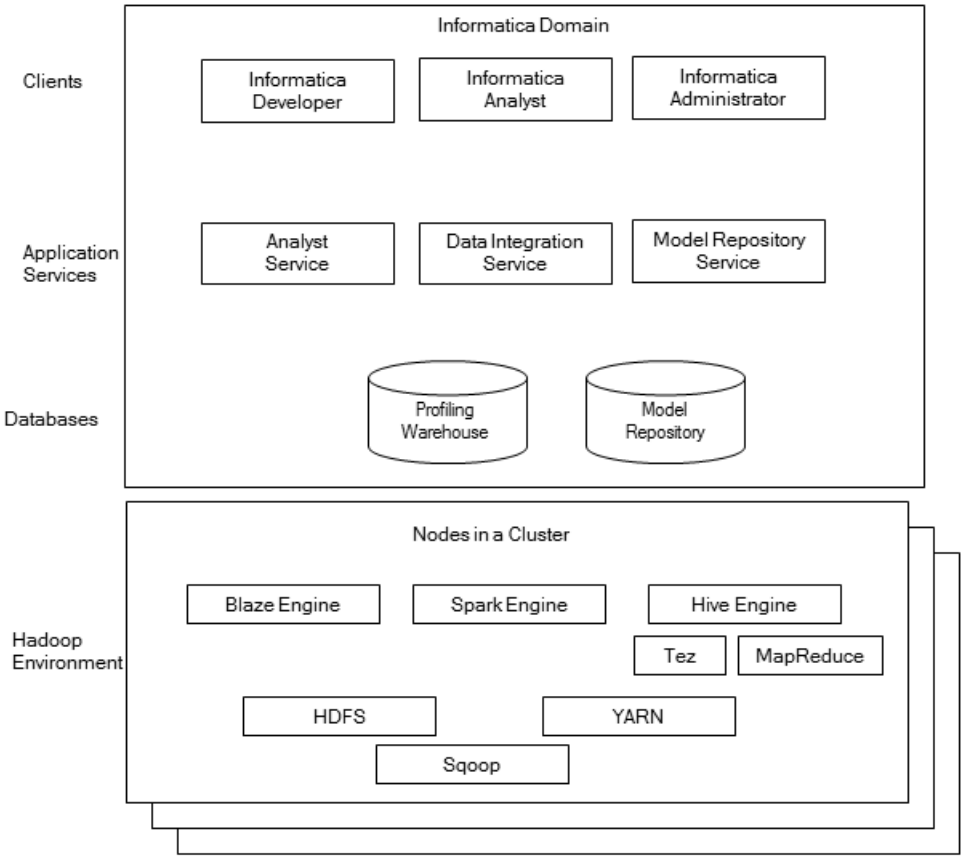
To install or upgrade Big Data Management, complete the following high-level tasks:

1. **Complete prerequisite tasks**. Prerequisite tasks include verifying minimum system requirements, creating users and directories, and installing third-party products. If you upgraded the Informatica platform or applied the hotfix, you must also uninstall Big Data Management from the cluster.
2. **Run the installer**. Download a package associated with the Hadoop distribution, and run the installer in the Hadoop environment.
3. **Configure the environments**. Run the Informatica Hadoop Configuration Manager (the configuration manager) on the machine that hosts the Data Integration Service to establish communication between the Informatica and Hadoop environments. After you run the configuration manager, you perform configuration tasks in both the domain and the Hadoop environments. If you upgraded the Informatica platform or applied the hotfix, you need to perform additional configuration tasks.

Big Data Management Component Architecture

The Big Data Management components include client tools, application services, repositories, and third-party tools that Big Data Management uses for a big data project. The specific components involved depend on the task you perform.

The following image shows the components of Big Data Management:



Hadoop Environment

Big Data Management connects to Hadoop clusters that are distributed by third parties. Hadoop is an open-source software framework that enables distributed processing of large data sets across clusters of machines. You might also need to use third-party software clients to set up and manage your Hadoop cluster.

Big Data Management can connect to Hadoop as a data source and push job processing to the Hadoop cluster. It can also connect to HDFS, which enables high performance access to files across the cluster. It can connect to Hive, which is a data warehouse that connects to HDFS and uses SQL-like queries to run MapReduce jobs on Hadoop, or YARN, which can manage Hadoop clusters more efficiently. It can also connect to NoSQL databases such as HBase, which is a database comprising key-value pairs on Hadoop that performs operations in real-time.

The Data Integration Service pushes mapping and profiling jobs to the Blaze, Spark, or Hive engine in the Hadoop environment.

Clients and Tools

Based on your product license, you can use multiple Informatica tools and clients to manage big data projects.

Use the following tools to manage big data projects:

Informatica Administrator

Monitor the status of profile, mapping, and MDM Big Data Relationship Management jobs on the Monitoring tab of the Administrator tool. The Monitoring tab of the Administrator tool is called the Monitoring tool. You can also design a Vibe Data Stream workflow in the Administrator tool.

Informatica Analyst

Create and run profiles on big data sources, and create mapping specifications to collaborate on projects and define business logic that populates a big data target with data.

Informatica Developer

Create and run profiles against big data sources, and run mappings and workflows on the Hadoop cluster from the Developer tool.

Application Services

Big Data Management uses application services in the Informatica domain to process data.

Big Data Management uses the following application services:

Analyst Service

The Analyst Service runs the Analyst tool in the Informatica domain. The Analyst Service manages the connections between service components and the users that have access to the Analyst tool.

Data Integration Service

The Data Integration Service can process mappings in the native environment or push the mapping for processing to the Hadoop cluster in the Hadoop environment. The Data Integration Service also retrieves metadata from the Model repository when you run a Developer tool mapping or workflow. The Analyst tool and Developer tool connect to the Data Integration Service to run profile jobs and store profile results in the profiling warehouse.

Model Repository Service

The Model Repository Service manages the Model repository. The Model Repository Service connects to the Model repository when you run a mapping, mapping specification, profile, or workflow.

Repositories

Big Data Management uses repositories and other databases to store data related to connections, source metadata, data domains, data profiling, data masking, and data lineage. Big Data Management uses application services in the Informatica domain to access data in repositories.

Big Data Management uses the following databases:

Model repository

The Model repository stores profiles, data domains, mapping, and workflows that you manage in the Developer tool. The Model repository also stores profiles, data domains, and mapping specifications that you manage in the Analyst tool.

Profiling warehouse

The Data Integration Service runs profiles and stores profile results in the profiling warehouse.

Integration with Informatica Products

To expand functionality and to process data more efficiently, you can use Big Data Management in conjunction with other Informatica products.

Big Data Management integrates with the following Informatica products:

- PowerExchange adapters. Connect to data sources through adapters.
- Enterprise Information Catalog. Perform data lineage analysis for big data sources and targets.
- Intelligent Data Lake. Discover raw data and publish it in a lake as a Hive table.
- Data Quality. Perform address validation and data discovery.
- Data Replication. Replicate change data to a Hadoop Distributed File System (HDFS).
- Data Transformation. Process complex file sources from the Hadoop environment.
- Intelligent Streaming. Stream data as messages, and process it as it becomes available.
- Vibe Data Stream. Collect and ingest data in real time to a Kafka queue.

PowerExchange Adapters Connectivity

Based on your business needs, install and configure Informatica adapters. Use Big Data Management with Informatica adapters for access to sources and targets.

You can use the following Informatica adapters in conjunction with Big Data Management:

- PowerExchange for Amazon Redshift
- PowerExchange for Amazon S3
- PowerExchange for Greenplum
- PowerExchange for HBase
- PowerExchange for HDFS
- PowerExchange for Hive
- PowerExchange for MapR-DB
- PowerExchange for Microsoft Azure Blob Storage
- PowerExchange for Microsoft Azure SQL Data Warehouse
- PowerExchange for Netezza
- PowerExchange for Teradata Parallel Transporter API

For more information, see the PowerExchange adapter documentation.

Data Lineage

Perform data lineage analysis in Enterprise Information Catalog for big data sources and targets.

Use Live Data Map to create a Cloudera Navigator resource to extract metadata for big data sources and targets and perform data lineage analysis on the metadata. Cloudera Navigator is a data management tool

for the Hadoop platform that enables users to track data access for entities and manage metadata about the entities in a Hadoop cluster.

You can create one Cloudera Navigator resource for each Hadoop cluster that is managed by Cloudera Manager. Live Data Map extracts metadata about entities from the cluster based on the entity type.

Live Data Map extracts metadata for the following entity types:

- HDFS files and directories
- Hive tables, query templates, and executions
- Oozie job templates and executions
- Pig tables, scripts, and script executions
- YARN job templates and executions

Note: Live Data Map does not extract metadata for MapReduce job templates or executions.

For more information, see the *Live Data Map Administrator Guide*.

Data Lakes

A data lake is a shared repository of raw and enterprise data from a variety of sources. It is often built over a distributed Hadoop cluster, which provides an economical and scalable persistence and compute layer.

Intelligent Data Lake is a collaborative self-service big data discovery and preparation solution for data analysts and data scientists. It enables analysts to rapidly discover and turn raw data into insight and allows IT to ensure quality, visibility, and governance. With Intelligent Data Lake, analysts to spend more time on analysis and less time on finding and preparing data.

Hadoop makes it possible to store large volumes of structured and unstructured data from various enterprise systems within and outside the organization. Data in the lake can include raw and refined data, master data and transactional data, log files, and machine data.

Intelligent Data Lake provides the following benefits:

- Data analysts can use semantic search and smart recommendations to find and explore trusted data assets within the data lake and outside the data lake.
- Data analysts can transform, cleanse, and enrich data in the data lake using an Excel-like spreadsheet interface.
- Data analysts can publish data and share knowledge with the rest of the community and analyze the data using their choice of BI or analytic tools.
- IT and governance staff can monitor user activity related to data usage in the lake.
- IT can track data lineage to verify that data is coming from the right sources and going to the right targets.
- IT can enforce appropriate security and governance on the data lake.
- IT can operationalize the work done by data analysts into a data delivery process that can be repeated and scheduled.

Address Validation

If you have a Data Quality product license, you can push a mapping that contains data quality transformations to a Hadoop cluster. Data quality transformations can use reference data to verify that data values are accurate and correctly formatted.

When you apply a pushdown operation to a mapping that contains data quality transformations, the operation can copy the reference data that the mapping uses. The pushdown operation copies reference table data and content set data to the Hadoop cluster. After the mapping runs, the cluster deletes the reference data that the pushdown operation copied with the mapping.

Note: The pushdown operation does not copy address validation reference data. If you push a mapping that performs address validation, you must install the address validation reference data files on each DataNode that runs the mapping. The cluster does not delete the address validation reference data files after the address validation mapping runs.

Address validation mappings validate and enhance the accuracy of postal address records. You can buy address reference data files from Informatica on a subscription basis. You can download the current address reference data files from Informatica at any time during the subscription period.

For more information, see the *Informatica Reference Data Guide*.

Data Discovery

Data discovery is the process of discovering the metadata of source systems that include content, structure, patterns, and data domains. Content refers to data values, frequencies, and data types. Structure includes candidate keys, primary keys, foreign keys, and functional dependencies. The data discovery process offers advanced profiling capabilities.

Run a profile to evaluate the data structure and to verify that data columns contain the types of information you expect. You can drill down on data rows in profiled data. If the profile results reveal problems in the data, you can apply rules to fix the result set. You can create scorecards to track and measure data quality before and after you apply the rules.

You can push column profiles and the data domain discovery process to the Hadoop cluster.

For more information, see the *Informatica Data Discovery Guide*.

Data Replication

You can replicate large amounts of transactional data between heterogeneous databases and platforms with Data Replication. You might replicate data to distribute or migrate the data across your environment.

You can configure Data Replication to write transactional changes to flat files instead of to a target database. Data Replication can generate these flat files on a file system or Hadoop Distributed File System (HDFS).

For more information, see the *Informatica Data Replication User Guide*.

Data Relationship Management

You can manage big data relationships by integrating data from different sources and indexing and linking the data in a Hadoop environment. Use Big Data Management to integrate data from different sources. Then use MDM Big Data Relationship Manager to index and link the data in a Hadoop environment.

MDM Big Data Relationship Manager indexes and links the data based on the indexing and matching rules. You can configure rules based on which to link the input records. MDM Big Data Relationship Manager uses the rules to match the input records and then group all the matched records. MDM Big Data Relationship

Manager links all the matched records and creates a cluster for each group of the matched records. You can load the indexed and matched record into a repository.

For more information, see the *MDM Big Data Relationship Management User Guide*.

Data Streaming

You can subscribe to sources that stream data and process data as it becomes available. Streaming sources stream data as messages. Use Informatica Intelligent Streaming mappings to collect the streamed data, build the business logic for the data, and push the logic to a Spark engine for processing.

Intelligent Streaming is built on Informatica Big Data Management and extends the platform to provide streaming capabilities. Intelligent Streaming uses Spark Streaming to process streamed data. It uses YARN to manage the resources on a Spark cluster more efficiently and uses third-parties distributions to connect to and push job processing to a Hadoop environment.

You can create streaming mappings to stream machine, device, and social media data. You can stream data from sources such as JMS providers and Apache Kafka brokers. A streaming mapping receives data from unbounded data sources. An unbounded data source is one where data continuously flows in and there is no definite boundary. Sources stream data as events. The Spark engine receives the input data streams and divides the data into micro batches. The Spark engine processes the data and publishes data in batches.

You can also use Informatica Vibe Data Stream to collect and ingest data in real time, for example, data from sensors and machine logs to a Kafka queue. Intelligent Streaming can process the data.

For more information, see the*Informatica Intelligent Streaming User Guide* and the *Informatica Vibe Data Stream for Machine Data User Guide*.

CHAPTER 2

Before You Install or Upgrade

This chapter includes the following topics:

- Before You Install or Upgrade Overview, 22
- Read the Release Notes, 23
- Uninstall Big Data Management, 23
- Verify System Requirements, 23
- Create a Data Integration Service User for Hive , 26
- Prepare the Hadoop Cluster for the Blaze Engine, 26
- Create a Staging Directory on HDFS , 28
- Record Configuration Information, 28

Before You Install or Upgrade Overview

The pre-install, install, and post-install tasks that you perform depend on your business requirements and Hadoop environment. Before you begin pre-install tasks, you need to understand the information required and the tasks that you need to perform during the installation and configuration process.

Some prerequisite tasks are required for all installations and upgrades. Some tasks depend on the following environmental factors:

- Hadoop distribution and installation package
- Single node or cluster environment
- Configuration through web interface or SSH
- Connections required
- Sqoop connectivity
- Run-time engines

Tip: The Informatica documentation team might update this guide with additions and clarifications after the general release. For the latest guide, go to the following link: https://network.informatica.com/onlinehelp/bdm/1011HF1/en/index.htm.

Read the Release Notes

Read the Release Notes for updates to the installation and upgrade process. You can also find information about known and fixed limitations for the release.

Uninstall Big Data Management

If you upgraded the Informatica domain or applied the hotfix, you must uninstall and reinstall Big Data Management from the cluster.

For information about uninstalling Big Data Management from Azure HDInsight, see Chapter 9, “Uninstall for Azure HDInsight” on page 75.

For information about uninstalling Big Data Management from Cloudera CDH, see Chapter 13, “Uninstall for Cloudera CDH” on page 104.

For information about uninstalling Big Data Management from Hortonworks HDP, see Chapter 17, “Uninstall for Hortonworks HDP” on page 133.

For information about uninstalling Big Data Management from IBM BigInsights, see Chapter 21, “Uninstall for IBM BigInsights” on page 159.

For information about uninstalling Big Data Management from MapR, see Chapter 25, “Uninstall for MapR” on page 184.

Verify System Requirements

Distribution: Amazon EMR, Azure HDInsight, Cloudera CDH, Hortonworks HDP, IBM BigInsights, MapR

Verify that your environment meets the minimum system requirements for the installation process, disk space requirements, port availability, and third-party software.

For more information about product requirements and supported platforms, see the Product Availability Matrix on Informatica Network:
https://network.informatica.com/community/informatica-network/product-availability-matrices/overview

Verify Product Installations

Distribution: Amazon EMR, Azure HDInsight, Cloudera CDH, Hortonworks HDP, IBM BigInsights, MapR

Before you install Big Data Management, verify that Informatica and third-party products are installed.

You must install the following products:

Informatica domain and clients

Verify that the Informatica domain and the Developer tool are 10.1.1 HotFix 1. The Informatica domain must have a Model Repository Service and a Data Integration Service.

Hadoop File System and MapReduce

Verify that Hadoop is installed with Hadoop File System (HDFS) and MapReduce on each node. Install Hadoop in a single node environment or in a cluster. The Hadoop installation must include a Hive data

warehouse with a non-embedded database for the Hive metastore. For more information, see the Apache website: http://hadoop.apache.org.

Database client software

Install the database client software to perform database read and write operations in native mode. Informatica requires the client software to run MapReduce jobs. For example, install the Oracle client to connect to an Oracle database.

Verify the Hadoop Distribution and Installation Package

Distribution: Amazon EMR, Azure HDInsight, Cloudera CDH, Hortonworks HDP, IBM BigInsights, MapR

Informatica distributes the installation package to the Hadoop cluster through a download package. Identify the distribution that you use and the required download package.

The following table lists the distribution versions and download packages for Big Data Management:

Distribution	Version	Package
Amazon EMR	5.0, 5.4 **Note:** Big Data Management supports Amazon EMR 5.4. To enable support for Amazon EMR 5.4, apply EBF-9585. When you apply the EBF, you disable support for Amazon EMR 5.0.	Red Hat Package Manager (RPM)
Azure HDInsight	3.5	Debian
Cloudera CDH	5.8, 5.9, 5.10, 5.11	Cloudera Parcel
Hortonworks HDP	2.3, 2.4, 2.5, 2.6	Ambari stack
IBM BigInsights	4.2	Red Hat Package Manager (RPM)
MapR	5.2 (MEP 1.0)	Red Hat Package Manager (RPM)

Verify Installer Requirements

Distribution: Amazon EMR, Azure HDInsight, Cloudera CDH, Hortonworks HDP, IBM BigInsights, MapR

Verify the following installer requirements before performing an installation:

- Verify that the user who performs the installation can run `sudo` commands or has user root privileges.
- Verify that the temporary folder on the local node has at least 2 GB of disk space.
- If you plan to install on a cluster, verify the connections to the Hadoop cluster nodes.
- If you plan to install with an RPM or Debian package, configure a Secure Shell (SSH) connection between the machine that runs the Big Data Management installation and the nodes in the Hadoop cluster. Configure passwordless SSH for the root user.
- If you are upgrading Big Data Management, verify that the destination directory for Informatica binary files is empty. Files that remain from previous installations can cause conflicts that lead to mapping run failures.

Verify Port Requirements

Distribution: Amazon EMR, Azure HDInsight, Cloudera CDH, Hortonworks HDP, IBM BigInsights, MapR

Open a range of ports to enable the Informatica domain to communicate with the Hadoop cluster and the distribution engine. If the Hadoop cluster is behind a firewall, work with your network administrator to open the range of ports that a distribution engine uses.

The following table lists the ports to open:

Port	Description
8020	NameNode RPC. Required for Amazon EMR and Azure HDInsight.
8032	ResourceManager. Required for Amazon EMR and Azure HDInsight
8080	NameNode API. Required for Amazon EMR and Azure HDInsight
8088	Debugging port. Optional for all distributions.
9080	Blaze monitoring console. Required for all distributions.
9083	Hive metastore. Required for Amazon EMR and Azure HDInsight
12300 to 12600	Port range for the Blaze distribution engine. Required for all distributions.
19888	Debugging port. Optional for all distributions.
50070	Debugging port. Optional for all distributions.

Verify Hortonworks HDP Requirements

Distribution: Hortonworks HDP

Before you install Big Data Management in the Ambari stack, verify the following prerequisites:

- The cluster has an Ambari stack using Ambari version 2.4.0.1 or later.
- The name node has at least two client cluster nodes where Big Data Management can be deployed.
- Verify symbolic links in the site-packages on the cluster name node and on each client node. Verify links in the following file: `/usr/lib/python<version>/site-packages`:
  The following table lists the links and the path on each node:

Link	Path on Each Client Node	Path on the NameNode
resource_management	`/usr/lib/ambari-agent/lib/resource_management`	`/usr/lib/ambari-server/lib/resource_management`
ambari_commons	`/usr/lib/ambari-agent/lib/ambari_commons`	`/usr/lib/ambari-server/lib/ambari_commons`

Link	Path on Each Client Node	Path on the NameNode
ambari_jina2	/usr/lib/ambari-agent/lib/ambari_jinja2	/usr/lib/ambari-server/lib/ambari_jinja2
ambari_simplejson	/usr/lib/ambari-agent/lib/ambari_simplejson	/usr/lib/ambari-server/lib/ambari_simplejson

Verify Azure HDInsight Requirements

Distribution: Azure HDInsight

To ensure that Informatica can access the HDInsight cluster, edit the `/etc/hosts` file on the machine that hosts the Data Integration Service to add the following information:

- Enter the IP address, DNS name, and DNS short name for each data node on the cluster. Use `headnodehost` to identify the host as the cluster headnode host.
  For example:
  ```
  10.75.169.19 hn0-rndhdi.grg2yxlb0aouniiuvfp3bet13d.ix.internal.cloudapp.net
  headnodehost
  ```
- If the HDInsight cluster is integrated with ADLS storage, you also need to enter the IP addresses and DNS names for the hosts listed in the cluster property fs.azure.datalake.token.provider.service.urls.
  For example:
  ```
  1.2.3.67  gw1-ltsa.1320suh5npyudotcgaz0izgnhe.gx.internal.cloudapp.net
  1.2.3.68  gw0-ltsa.1320suh5npyudotcgaz0izgnhe.gx.internal.cloudapp.net
  ```
  Note: To get the IP addresses, run a telnet command from the cluster host using each host name found in the fs.azure.datalake.token.provider.service.urls property.

Create a Data Integration Service User for Hive

Distribution: Amazon EMR, Azure HDInsight, Cloudera CDH, Hortonworks HDP, IBM BigInsights, MapR

To access the Hive warehouse and logs, create a Data Integration Service user on every node in the cluster. Verify that the Data Integration Service user has read and write permissions on the following directories:

- Log directory. The logs are in the following location: `opt/<distribution>/hive/hive-<version>/logs`.
- Warehouse directory. You can find the location of the warehouse directory in the following location: `/opt/<distribution>/hive/hive-<version>/conf/hive-site.xml`. For example, `/user/hive/warehouse`.

Note: If the MapR distribution uses Ticket or Kerberos authentication ensure that the gid of the Data Integration Service user matches the gid of the MapR user.

Prepare the Hadoop Cluster for the Blaze Engine

Distribution: Amazon EMR, Azure HDInsight, Cloudera CDH, Hortonworks HDP, IBM BigInsights, MapR

To run mappings on the Blaze engine, you need to create a Blaze engine user and log directories. You also need to grant permissions to the user.

Create a Blaze User Account

Distribution: Amazon EMR, Azure HDInsight, Cloudera CDH, Hortonworks HDP, IBM BigInsights, MapR

On all nodes in the Hadoop cluster, create an operating system user account for the Blaze engine user. For example, run the following command:

```
useradd blaze
```

Create Blaze Engine Directories and Grant Permissions

Distribution: Amazon EMR, Azure HDInsight, Cloudera CDH, Hortonworks HDP, IBM BigInsights, MapR

Create the following directories on the Hadoop cluster:

Home directory for the blaze user account

Create a home directory for the blaze user account and make it the owner. For example, run the following commands:

```
hdfs hadoop fs -mkdir /user/blaze
hdfs hadoop fs -chown blaze:blaze /user/blaze
```

Local services log directory

Create a local services log directory on all nodes in the cluster, and grant permissions to the blaze user account and all mapping impersonation users. For example, run the following commands:

```
mkdir -p /opt/Informatica/blazeLogs
chmod 777 /opt/Informatica/blazeLogs
```

The hadoopEnv.properties file on the domain contains an entry for this directory. The file uses an environment variable, $HADOOP_NODE_INFA_HOME, that gets set to the Big Data Management installation directory. The default log services directory for Cloudera CDH is `/opt/cloudera/parcels/INFORMATICA`. The default directory for other distributions is `/opt/Informatica`.

Note: The `infagrid.node.local.root.log.dir` property in hadoopEnv.properties is set to `/opt/Informatica/blazeLogs`. You must update the hadoopEnv.properties file if you use a directory other than `/opt/Informatica/blazeLogs`.

Aggregated HDFS log directory

Create a log directory on HDFS to contain aggregated logs for local services, and grant permissions to the blaze user account and all mapping impersonation users. For example, run the following commands:

```
hadoop fs -mkdir -p /var/log/hadoop-yarn/apps/informatica
hadoop fs -chmod 777 /var/log/hadoop-yarn/apps/informatica
```

The hadoopEnv.properties file on the domain contains an entry for this directory.

Note: The `infacal.hadoop.logs.directory` property in hadoopEnv.properties is set to `/var/log/hadoop-yarn/apps/informatica`. You must update the hadoopEnv.properties file if you use a directory other than `/var/log/hadoop-yarn/apps/informatica`.

HDFS temporary working directory

Create a working directory on HDFS for the Blaze engine, and grant permissions to the blaze user account and all mapping impersonation users. For example, run the following commands:

```
hadoop fs -mkdir -p /blaze/workdir
hadoop fs -chmod 777 /blaze/workdir
```

When you configure connection properties through the Hadoop Configuration Manager or through the Developer tool, you need to provide the path to this directory. Alternatively, you can create this directory when you create the connection.

Grant Permissions on the Hive Source Database

Distribution: Amazon EMR, Azure HDInsight, Cloudera CDH, Hortonworks HDP, IBM BigInsights, MapR

Grant the Blaze user account CREATE TABLE permission on the Hive source database. The CREATE TABLE permission is required in the following situations:

- The Hive source table uses SQL standard-based authorization.
- A mapping contains a Lookup transformation with an SQL override.

Create a Staging Directory on HDFS

Distribution: Amazon EMR, Azure HDInsight, Cloudera CDH, Hortonworks HDP, IBM BigInsights, MapR

If the cluster uses Hive, you must either grant the anonymous user the Execute permission on the staging directory or create another staging directory on HDFS.

Perform one of the following tasks:

Grant permission on the default staging directory.

Grant the anonymous user the Execute permission on the default staging directory. If you cannot grant the anonymous user the Execute permission on this directory, then you must enter a valid user name for the user in the Hive connection when you run the configuration manager.

Create a staging directory on HDFS.

To create the staging directory, run the following commands from the command line of the machine that runs the Hadoop cluster:

```
hadoop fs -mkdir /staging
```

```
hadoop fs -chmod -R 0777 /staging
```

Add the staging directory to the `yarn.app.mapreduce.am.staging-dir` property in the following files:

- mapred-site.xml on the Hadoop cluster. You can find the file in the following location: `/etc/hadoop/conf`
- hive-site.xml on the machine where the Data Integration Service runs. You can find the file in the following location: `<Informatica installation directory>/services/shared/hadoop/cloudera_<version>/conf`

For example,

```
<property>
   <name>yarn.app.mapreduce.am.staging-dir</name>
   <value>/staging</value>
</property>
```

Record Configuration Information

Distribution: Azure HDInsight, Cloudera CDH, Hortonworks HDP, IBM BigInsights, MapR

When you run the Hadoop Configuration Manager, you need to know information about the domain and Hadoop environments. Gather the required information so you can easily perform the configuration.

Gather Domain Information

Distribution: Azure HDInsight, Cloudera CDH, Hortonworks HDP, IBM BigInsights, MapR

When you run the configuration manager, you must provide information about the Informatica domain. Gather domain information before you run the configuration manager.

The following table describes the information that you need to know about the Informatica domain:

Domain Information	Description
Domain user name	The domain user name.
Domain password	The password of the domain user.
Data Integration Service name	The name of the Data Integration Service.
Informatica home directory on Hadoop	The directory where Big Data Management is installed in the Hadoop environment. Get this information after you perform the installation.
Hadoop Kerberos service principal name	Required if the Hadoop cluster uses Kerberos authentication.
Hadoop Kerberos keytab location	Required if the Hadoop cluster uses Kerberos authentication. The location of the keytab file on the Data Integration Service machine.

Gather Connection Information

Distribution: Azure HDInsight, Cloudera CDH, Hortonworks HDP, IBM BigInsights, MapR

When you complete configuration on the domain, you can choose to create connections for Hadoop, Hive, HDFS, and HBase.

The following table describes the information that you need to know to create connections:

Connection Information	Description
Hive connection name	Required for the Hive connection. The name is not case sensitive and must be unique within the domain. The name cannot exceed 128 characters, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Hive user name	Required for the Hive connection. User name of the user that the Data Integration Service impersonates to run mappings on a Hadoop cluster. The user name depends on the JDBC connection string that you specify in the Metadata Connection String or Data Access Connection String for the native environment. If the Hadoop cluster runs Hortonworks HDP, you must provide a user name. If you use Tez to run mappings, you must provide the user account for the Data Integration Service. If you do not use Tez to run mappings, you can use an impersonation user account. If the Hadoop cluster uses Kerberos authentication, the principal name for the JDBC connection string and the user name must be the same. Otherwise, the user name depends on the behavior of the JDBC driver. With Hive JDBC driver, you can specify a user name in many ways and the user name can become a part of the JDBC URL. If the Hadoop cluster does not use Kerberos authentication, the user name depends on the behavior of the JDBC driver.

Connection Information	Description
HDFS connection name	Required for the HDFS connection. The name is not case sensitive and must be unique within the domain. The name cannot exceed 128 characters, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
HDFS user name	Required for the HDFS connection. User name to access HDFS.
Hadoop connection name	Required for the Hadoop connection. The name is not case sensitive and must be unique within the domain. The name cannot exceed 128 characters, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Impersonation user name	Required for the Hadoop connection if the cluster uses Kerberos authentication. User name of the user that the Data Integration Service impersonates to run mappings on a Hadoop cluster. If the Hadoop cluster uses Kerberos authentication, the principal name for the JDBC connection string and the user name must be the same.
Blaze work directory on HDFS	Required for all connection types. The HDFS file path of the directory that the Blaze engine uses to store temporary files.
Blaze user name	Required for all connection types. The Blaze user account on the cluster.
Spark staging directory on HDFS	Required for all connection types. The HDFS file path of the directory that the Spark engine uses to store temporary files for running jobs. The YARN user, Spark engine user, and mapping impersonation user must have write permission on this directory.
Spark event log directory	Optional for all connection types. The HDFS file path of the directory that the Spark engine uses to log events. The Data Integration Service accesses the Spark event log directory to retrieve final source and target statistics when a mapping completes. These statistics appear on the Summary Statistics tab and the Detailed Statistics tab of the Monitoring tool. If you do not configure the Spark event log directory, the statistics might be incomplete in the Monitoring tool.
Spark execution parameters list	Optional for all connection types.
HBase connection name	Required for HBase connections. The name is not case sensitive and must be unique within the domain. The name cannot exceed 128 characters, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
ZooKeeper hosts	Required for HBase connections. Name of the machine that hosts the ZooKeeper server.

Gather Information for Configuration Through SSH

Distribution: Azure HDInsight, Cloudera CDH, Hortonworks HDP, IBM BigInsights, MapR

When you run the configuration manager you can choose the tool that you use to configure the domain connectivity. You can perform the configuration through a web interface or through SSH. Complete this section if you plan to configure the domain information through SSH.

Note: If you did not use the web user interface to create the cluster, you cannot use the web user interface to run the configuration utility.

Gather the following information that is required to perform configuration through SSH:

Host name of the NameNode or Resource Manager, based on the distribution

Host name of JobTracker

Host name of the Hive client

Location of hdfs-site.xml

Location of core-site.xml

Location of mapred-site.xml

Location of yarn-site.xml

Location of hive-site.xml

Part II: Install and Configure for Amazon EMR

This part contains the following chapters:

- Installation for Amazon EMR, 33
- Configuration for Amazon EMR, 39
- Uninstall for Amazon EMR, 50

CHAPTER 3

Installation for Amazon EMR

This chapter includes the following topics:

- Installation for Amazon EMR Overview, 33
- Download the Distribution Package, 33
- Install on a Cluster Using SCP, 34
- Install on a Cluster Using NFS, 35
- Install on a Cluster Using the HadoopDataNodes File, 36
- Create a Cluster and Install Big Data Management, 36

Installation for Amazon EMR Overview

You can install Big Data Management for Amazon EMR in different environments.

Install Big Data Management on Amazon EMR in one of the following ways:

- Install on a single node.
- Install on a cluster from the primary NameNode using SCP protocol.
- Install on a cluster from the primary NameNode using NFS protocol.
- Install on a cluster using the HadoopDataNodes file.
- Create a cluster on Amazon EMR and then install on the cluster.

Download the Distribution Package

The tar.gz file that you download includes an RPM package and the binary files required to run the installation.

1. Create a temporary directory in a shared location on a local disk.
2. Download the following file: `InformaticaHadoop-<version>.<platform>-x64.tar.gz`
3. Extract the file to the machine from where you want to distribute the package and run the Big Data Management installation.

Install on a Single Node

You can install Big Data Management in a single node environment.

1. Log in to the machine as root user or as a user with sudo privileges.
2. Run one of the following commands to start the installation:

   ```
   bash InformaticaHadoopInstall.sh
   sh InformaticaHadoopInstall.sh
   ./InformaticaHadoopInstall.sh
   ```

3. Press **y** to accept the Big Data Management terms of agreement.
4. Press **Enter**.
5. Press **1** to install Big Data Management in a single node environment.
6. Press **Enter**.
7. Type the absolute path for the Big Data Management installation directory and press **Enter**.

   Start the path with a slash. The directory names in the path must not contain spaces or the following special characters: { } ! @ # $ % ^ & * ( ) : ; | ' ` < > , ? + [ ] \

   If you type a directory path that does not exist, the installer creates the entire directory path on the node during the installation. Default is `/opt`.
8. Press **Enter**.

   The installer creates the following directory and populates all of the file systems with the contents of the RPM package: `/<Big Data Management installation directory>/Informatica`

You can view the `informatica-hadoop-install.<date time stamp>.log` installation log file to get more information about the tasks performed by the installer.

Install on a Cluster Using SCP

You can install Big Data Management in a cluster environment from the primary namenode using the SCP protocol.

1. Log in to the primary NameNode as root user or as a user with sudo privileges.
2. Run one of the following commands to start the installation:

   ```
   bash InformaticaHadoopInstall.sh
   sh InformaticaHadoopInstall.sh
   ./InformaticaHadoopInstall.sh
   ```

3. Press **y** to accept the Big Data Management terms of agreement.
4. Press **Enter**.
5. Press **2** to install Big Data Management in a cluster environment.
6. Press **Enter**.
7. Type the absolute path for the Big Data Management installation directory.

   Start the path with a slash. The directory names in the path must not contain spaces or the following special characters: { } ! @ # $ % ^ & * ( ) : ; | ' ` < > , ? + [ ] \

   If you type a directory path that does not exist, the installer creates the entire directory path on each of the nodes during the installation. Default is `/opt/Informatica`.
8. Press **Enter**.

9. Press **1** to install Big Data Management from the primary name node.
10. Press **Enter**.
11. Type the absolute path for the Hadoop installation directory. Start the path with a slash.
12. Press **Enter**.
13. Type **y**.
14. Press **Enter**.

    The installer retrieves a list of DataNodes from the `$HADOOP_HOME/conf/slaves` file. On each DataNode, the installer creates the following directory and populates all of the file systems with the contents of the RPM package: `/<Big Data Management installation directory>/Informatica`

You can view the `informatica-hadoop-install.<date time stamp>.log` installation log file to get more information about the tasks performed by the installer.

Install on a Cluster Using NFS

You can install Big Data Management in a cluster environment from the primary NameNode using the NFS protocol.

1. Log in to the primary NameNode as root user or as a user with sudo privileges.
2. Run one of the following commands to start the installation:

   ```
   bash InformaticaHadoopInstall.sh
   sh InformaticaHadoopInstall.sh
   ./InformaticaHadoopInstall.sh
   ```
3. Press **y** to accept the Big Data Management terms of agreement.
4. Press **Enter**.
5. Press **2** to install Big Data Management in a cluster environment.
6. Press **Enter**.
7. Type the absolute path for the Big Data Management installation directory.

   Start the path with a slash. The directory names in the path must not contain spaces or the following special characters: { } ! @ # $ % ^ & * ( ) : ; | ' ` < > , ? + [ ] \

   If you type a directory path that does not exist, the installer creates the entire directory path on each of the nodes during the installation. Default is `/opt`.
8. Press **Enter**.
9. Press **1** to install Big Data Management from the primary name node.
10. Press **Enter**.
11. Type the absolute path for the Hadoop installation directory. Start the path with a slash.
12. Press **Enter**.
13. Type **n**.
14. Press **Enter**.
15. Type **y**.

16. Press **Enter**.

    The installer retrieves a list of DataNodes from the `$HADOOP_HOME/conf/slaves` file. On each DataNode, the installer creates the following directory and populates all of the file systems with the contents of the RPM package: `/<Big Data Management installation directory>/Informatica`

You can view the `informatica-hadoop-install.<date time stamp>.log` installation log file to get more information about the tasks performed by the installer.

Install on a Cluster Using the HadoopDataNodes File

You can install Big Data Management in a cluster environment using the HadoopDataNodes file.

1. Log in to the machine as the root user.
2. Update the `HadoopDataNodes` file that was extracted with the download package. Add an entry for each IP address or machine host name in the cluster.
3. Run one of the following commands to start the installation:

   ```
   bash InformaticaHadoopInstall.sh
   sh InformaticaHadoopInstall.sh
   ./InformaticaHadoopInstall.sh
   ```
4. Press **y** to accept the Big Data Management terms of agreement.
5. Press **Enter**.
6. Press **2** to install Big Data Management in a cluster environment.
7. Press **Enter**.
8. Type the absolute path for the Big Data Management installation directory and press **Enter**. Start the path with a slash. Default is `/opt`.
9. Press **Enter**.
10. Press **2** to install Big Data Management using the `HadoopDataNodes` file.
11. Press **Enter**.

    The installer creates the following directory on each node that appears in the `HadoopDataNodes` file:`/<Big Data Management installation directory>/Informatica`. The installer then populates all of the file systems with the contents of the RPM package.

You can view the `informatica-hadoop-install.<date time stamp>.log` installation log file to get more information about the tasks performed by the installer.

Create a Cluster and Install Big Data Management

You can run a cluster creation wizard and install Big Data Management on the created cluster.

You upload the RPM package to an S3 bucket, and prepare and upload a bootstrap script. Use the cluster creation wizard to create an Amazon EMR cluster. The cluster creation wizard uses values in the script to download the RPM package from the Amazon S3 bucket and extract the package. Then the wizard creates a cluster and installs Big Data Management.

Perform the following steps to create a cluster on Amazon EMR and install Big Data Management:

1. Create a bootstrap script that the cluster creation wizard uses to install Big Data Management.
2. Upload the Big Data Management RPM package to an S3 bucket.
3. Upload the EBF-9585 Hadoop installer `EBF9585_HadoopEBF_EBFInstaller.tar` to the same S3 bucket.
4. Run the cluster creation wizard to create the Amazon EMR cluster and run the script.

Create the Bootstrap Script

Create a bootstrap script that the cluster creation wizard uses to download the RPM package from the Amazon S3 bucket and extract the package.

1. Copy the following text to a text editor:

```
#!/bin/bash

echo s3 location of RPM
export S3_LOCATION_RPM=s3://<s3-bucket-name>

echo Temp location to extract the RPM
export TEMP_DIR=/tmp/<TEMP-DIR-TO-EXTRACT-RPM>

echo Default location to install Informatica RPM
#make sure that INFA_RPM_INSTALL_HOME will have enough space to install the
Informatica RPM
export INFA_RPM_INSTALL_HOME=/opt/

echo Extracting the prefix part from the rpm file name
echo The rpm installer name would be  informatica_1011HF1_hadoop_linux-x64.tar.gz
export INFA_RPM_FILE_PREFIX=informatica_1011HF1_hadoop_linux-x64
export INFA_RPM_FOLDER=InformaticaHadoop-10.1.101-1.55
export INFA_RPM_EBF_PREFIX=EBF9585_HadoopEBF_EBFInstaller
export INFA_RPM_EBF_FOLDER=EBF9585_HadoopEBF

echo S3_LOCATION_RPM = $S3_LOCATION_RPM
echo TEMP_DIR = $TEMP_DIR
echo INFA_RPM_INSTALL_HOME = $INFA_RPM_INSTALL_HOME
echo INFA_RPM_FILE_PREFIX = $INFA_RPM_FILE_PREFIX
echo INFA_RPM_EBF_PREFIX = $INFA_RPM_EBF_PREFIX

echo  Installing the RPM:
echo "Creating temporary folder for rpm extraction"
sudo mkdir -p $TEMP_DIR
cd $TEMP_DIR/
echo "current directory =" $(pwd)

echo Getting RPM installer
echo Copying the rpm installer $S3_LOCATION_RPM/$INFA_RPM_FILE_PREFIX.tar.gz to $
(pwd)
sudo aws s3 cp $S3_LOCATION_RPM/$INFA_RPM_FILE_PREFIX.tar.gz .
sudo tar -zxvf $INFA_RPM_FILE_PREFIX.tar.gz
cd $INFA_RPM_FOLDER
echo Installing RPM to $INFA_RPM_INSTALL_HOME

sudo rpm -ivh --replacefiles --replacepkgs InformaticaHadoop-10.1.101-1.x86_64.rpm --
prefix=$INFA_RPM_INSTALL_HOME

echo Contents of $INFA_RPM_INSTALL_HOME
echo $(ls $INFA_RPM_INSTALL_HOME)

echo chmod
cd $INFA_RPM_INSTALL_HOME
sudo mkdir -p Informatica/blazeLogs
sudo chmod 766 -R Informatica/blazeLogs
echo removing temporary folder
```

```
sudo rm -rf $TEMP_DIR/

# Insert commands to perform additional tasks.
# For example, the section below applies an EBF.

echo Applying the EBF
echo Creating temporary folder
sudo mkdir -p $TEMP_DIR
cd $TEMP_DIR/
echo Getting EBF installer
sudo aws s3 cp $S3_LOCATION_RPM/"$INFA_RPM_EBF_PREFIX".tar .
echo Applying EBF installer to $INFA_RPM_INSTALL_HOME
sudo tar -xvf "$INFA_RPM_EBF_PREFIX".tar
cd $TEMP_DIR/$INFA_RPM_EBF_FOLDER
sudo chmod +x InformaticaHadoopEBFInstall.sh
sudo -S ./InformaticaHadoopEBFInstall.sh <<< $1$'\nYes\n1\n'

cd $INFA_RPM_INSTALL_HOME
echo Removing temporary folder
sudo rm -rf $TEMP_DIR/

echo done
```

2. Edit the bootstrap script to supply values for the following variables:

   <s3-bucket-name>

   Name of the Amazon S3 bucket that contains the RPM .tar file.

   <TEMP-DIR-TO-EXTRACT-RPM>

   Temporary directory location to extract the RPM package to.

   <build_number>

   RC build number.

3. Optionally add additional tasks, such as installing an EBF or patch release, at the following point in the script:

   ```
   # You can insert additional tasks at this point in the script.
   ```

4. Save the script file with the suffix `.bash` in the file name.
5. Upload the edited script file to the S3 bucket where you want to create the cluster.

Run the Cluster Creation Wizard

Before you run the cluster creation wizard, upload the RPM package .tar file to the S3 bucket where you want to create the cluster. You supply this location during the cluster creation steps.

1. Launch the cluster creation wizard.
2. On the **Edit software settings (optional)** page, select **Enter configuration**.
3. In the text pane, paste the following set of properties and values to configure the cluster for the Blaze run-time engine:

   ```
   classification=yarn-site,properties=[yarn.scheduler.minimum-allocation-
   mb=256,yarn.nodemanager.resource.memory-mb=14000,yarn.nodemanager.resource.cpu-
   vcores=15,yarn.scheduler.maximum-allocation-mb=8192,yarn.nodemanager.vmem-check-
   enabled=false, yarn.nodemanager.vmem-pmem-ratio=12]
   ```

   Note: The values specified in the sample above are the minimum values required.

4. On the **General Cluster Settings** page, provide the S3 location of the bootstrap script.
5. Click **Create Cluster**.

The cluster creation wizard uses values in the bootstrap script to download the RPM package from the Amazon S3 bucket and extract the package. Then the wizard creates a cluster, where it installs Big Data Management.

CHAPTER 4

Configuration for Amazon EMR

This chapter includes the following topics:

- Configure the Data Integration Service, 39
- Download the JDBC Drivers for Sqoop Connectivity, 40
- Copy .jar Files for Hive Access, 40
- Update Files on the Domain Environment for Amazon EMR, 41
- Update Files on the Hadoop Environment for Amazon EMR, 44
- Set S3 Bucket Access Policies, 46
- Developer Tool Tasks for Amazon EMR, 46
- Complete Upgrade Tasks, 47

Configure the Data Integration Service

Configure Data Integration Service properties for the Hadoop environment.

You can configure the Data Integration Service in the Administrator tool.

The following table describes the Hadoop properties for the Data Integration Service:

Property	Description
Informatica Home Directory on Hadoop	The Big Data Management home directory on every data node created by the Hadoop RPM install. Default is `/opt/Informatica`.
Hadoop Distribution Directory	The directory containing a collection of Hive and Hadoop JARS on the cluster from the RPM Install locations. The directory contains the minimum set of JARS required to process Informatica mappings in a Hadoop environment. Type `/<Big Data Management installation directory>/Informatica/services/shared/hadoop/<Hadoop distribution name>_<version number>`. For example: `/opt/Informatica/services/shared/hadoop/amazon_emr_5.0.0`
Data Integration Service Hadoop Distribution Directory	The Hadoop distribution directory on the Data Integration Service node. Type `/<Informatica installation directory>/Informatica/services/shared/hadoop/<Hadoop distribution name>_<version number>`. For example: `../../services/shared/hadoop/amazon_emr_5.0.0` **Note:** The contents of the Data Integration Service Hadoop distribution directory must be identical to Hadoop distribution directory on the data nodes.

Download the JDBC Drivers for Sqoop Connectivity

To configure Sqoop connectivity for relational databases, you must download JDBC driver jar files.

1. Download any Type 4 JDBC driver that the database vendor recommends for Sqoop connectivity.
2. Copy the jar files to the following directory on the machine where the Data Integration Service runs:
   `<Informatica installation directory>\externaljdbcjars`

At run time, the Data Integration Service copies the jar files to the Hadoop distribution cache so that the jar files are accessible to all nodes in the cluster.

Note: The DataDirect JDBC drivers that Informatica ships are not licensed for Sqoop connectivity.

Copy .jar Files for Hive Access

Copy .jar files to enable Hive access.

To access Hive tables, you need to copy .jar files from the master node to the Hadoop distribution directories on the cluster and on the Data Integration Service nodes.

Find the following .jar files on the master node in the Hadoop cluster:

- emrfs-hadoop-assembly-2.9.0.jar
- s3-dist-cp.jar

- hadoop-common-2.7.2-amzn-3.jar

Copy the .jar files to the following locations:

- The Hadoop distribution directory on the Hadoop cluster: `<RPM installation directory>/services/shared/hadoop/amazon_emr_<version number>/lib`
  For example, `/opt/Informatica/services/shared/hadoop/amazon_emr_5.4.0/lib`
- The Data Integration Service Hadoop distribution directory: `/<Informatica installation directory>/services/shared/hadoop/amazon_emr_<version number>/lib`

Update Files on the Domain Environment for Amazon EMR

You can configure additional functionality through the configuration files on the domain.

When the Informatica installer creates nodes on the domain, it creates directories and files for Big Data Management. When you configure Big Data Management on the domain, you need to edit configuration files to enable functionality.

The following table describes the installation directories and the default paths:

Directory	Description
Data Integration Service Hadoop distribution directory	The Hadoop distribution directory on the Data Integration Service node. It corresponds to the distribution on the Hadoop environment. You can view the directory path in the Data Integration Service properties of the Administrator tool. Default is `<Informatica installation directory>/Informatica/services/shared/hadoop/<Hadoop distribution name>_<version>`.
Configuration file directories	Configuration files for Big Data Management are stored in the following directories: - `<Informatica installation directory>/Informatica/services/shared/hadoop/<Hadoop distribution name>_<version>/InfaConf`. Contains the hadoopEnv.properties file. - `<Informatica installation directory>/Informatica/services/shared/hadoop/<Hadoop distribution name>_<version>/conf`. Contains *-site files.

Update hadoopEnv.properties

Update the hadoopEnv.properties file on the domain environment to configure functionality such as Sqoop connectivity and the Spark run-time engine.

Open hadoopEnv.properties on the domain environment and back it up before you configure it. You can find the hadoopEnv.properties file in the following location:

```
<Informatica installation directory>/services/shared/hadoop/<distribution name>_<version number>/infaConf
```

Configure Sqoop Connectivity

Configure the following property for Sqoop connectivity:

infapdo.env.entry.hadoop_node_jdk_home

Configure the HADOOP_NODE_JDK_HOME to represent the JDK version that the cluster nodes use. You must use JDK version 1.7 or later.

Configure the property as follows:

```
infapdo.env.entry.hadoop_node_jdk_home=HADOOP_NODE_JDK_HOME=<cluster JDK home>/
jdk<version>
```

For example,

```
infapdo.env.entry.hadoop_node_jdk_home=HADOOP_NODE_JDK_HOME=/usr/java/default
```

Configure Performance for the Spark Engine

Configure the following performance properties for the Spark engine:

spark.dynamicAllocation.enabled

Performance configuration to run mappings on the Spark engine. Enables dynamic resource allocation. Required when you enable the external shuffle service.

Set the value to TRUE.

spark.shuffle.service.enabled

Performance configuration to run mappings on the Spark engine. Enables the external shuffle service. Required when you enable dynamic resource allocation.

Set the value to TRUE.

spark.scheduler.maxRegisteredResourcesWaitingTime

Performance configuration to run mappings on the Spark engine. The number of milliseconds to wait for resources to register before scheduling a task. Reduce this from the default value of 30000 to reduce delays before starting the Spark job execution.

Set the value to 15000.

spark.scheduler.minRegisteredResourcesRatio

Performance configuration to run mappings on the Spark engine. The minimum ratio of registered resources to acquire before task scheduling begins. Reduce this from the default value of 0.8 to reduce any delay before starting the Spark job execution.

Set the value to .5.

spark.executor.instances

Performance configuration to run mappings on the Spark engine. If you enable dynamic resource allocation for the Spark engine, Informatica recommends that you convert this property to a comment. For example, `#spark.executor.instances=100`

By default, Amazon AWS enables the Spark shuffle service. You need to enable dynamic allocation for the Data Integration Service, but not for Amazon EMR. For information about the Amazon EMR configuration, see the Amazon EMR documentation.

Configure Environment Variables

You can optionally add third-party environment variables and extend the existing PATH environment variable in the hadoopEnv.properties file. The following text shows sample entries to configure environment variables:

```
infapdo.env.entry.oracle_home=ORACLE_HOME=/databases/oracle
infapdo.env.entry.db2_home=DB2_HOME=/databases/db2
infapdo.env.entry.db2instance=DB2INSTANCE=OCA_DB2INSTANCE
infapdo.env.entry.db2codepage=DB2CODEPAGE="1208"
infapdo.env.entry.odbchome=ODBCHOME=$HADOOP_NODE_INFA_HOME/ODBC7.1
infapdo.env.entry.home=HOME=/opt/thirdparty
infapdo.env.entry.gphome_loaders=GPHOME_LOADERS=/databases/greenplum
```

```
infapdo.env.entry.pythonpath=PYTHONPATH=$GPHOME_LOADERS/bin/ext
infapdo.env.entry.nz_home=NZ_HOME=/databases/netezza
infapdo.env.entry.ld_library_path=LD_LIBRARY_PATH=$HADOOP_NODE_INFA_HOME/services/
shared/bin:$HADOOP_NODE_INFA_HOME/DataTransformation/bin:$HADOOP_NODE_HADOOP_DIST/lib/
native:$HADOOP_NODE_INFA_HOME/ODBC7.1/lib:$HADOOP_NODE_INFA_HOME/jre/lib/
amd64:$HADOOP_NODE_INFA_HOME/jre/lib/amd64/server:$HADOOP_NODE_INFA_HOME/java/jre/lib/
amd64:$HADOOP_NODE_INFA_HOME/java/jre/lib/amd64/server:/databases/oracle/lib:/
databases/db2/lib64:$LD_LIBRARY_PATH
infapdo.env.entry.path=PATH=$HADOOP_NODE_HADOOP_DIST/scripts:$HADOOP_NODE_INFA_HOME/
services/shared/bin:$HADOOP_NODE_INFA_HOME/jre/bin:$HADOOP_NODE_INFA_HOME/java/jre/bin:
$HADOOP_NODE_INFA_HOME/ODBC7.1/bin:/databases/oracle/bin:/databases/db2/bin:$PATH
#teradata
infapdo.env.entry.twb_root=TWB_ROOT=/databases/teradata/tbuild
infapdo.env.entry.manpath=MANPATH=/databases/teradata/odbc_64:/databases/teradata/odbc_64
infapdo.env.entry.nlspath=NLSPATH=/databases/teradata/odbc_64/msg/%N:/databases/
teradata/msg/%N
infapdo.env.entry.pwd=PWD=/databases/teradata/odbc_64/samples/C
```

Update hive-site.xml

Update the hive-site.xml file on the domain environment to set properties for dynamic partitioning.

Configure Dynamic Partitioning

To use Hive dynamic partitioned tables, configure the following properties:

hive.exec.dynamic.partition

Enables dynamic partitioned tables. Set this value to `TRUE`.

exec.dynamic.partition.mode

Allows all partitions to be dynamic. Set this value to `nonstrict`.

Update yarn-site.xml

Update the yarn-site.xml file on the domain environment. You can find the yarn-site.xml file in the following location: `<Informatica installation directory>/services/shared/hadoop/<distribution name>_<version number>/conf`

Update Host Name

Replace all instances of `HOSTNAME` in the yarn-site.xml file with the master host name from the Hadoop environment. You can find the master host name in the yarn-site.xml file that is in the Hadoop environment. The file is in the following location on the Hadoop environment: `/etc/hadoop/conf` .

Configure Access to Hive Tables in Amazon S3 Buckets

Configure the AWS access key to run mappings with sources and targets on Hive tables in Amazon S3 buckets.

Use an s3 URI to connection to Amazon S3 buckets on Amazon EMR. For more information about using an s3 URI, see the Amazon note at the following URL: https://aws.amazon.com/premiumsupport/knowledge-center/emr-file-system-s3/.

Note: To use a Hive table as a target on Amazon S3, grant write permission to the bucket through bucket policies, or add these properties to the configuration file. You must add these properties to the core-site.xml file on the Hadoop environment and to the yarn-site.xml file on the domain environment.

Configure the following properties:

fs.s3.awsAccessKeyId

The ID for the Blaze and Spark engines to connect to the Amazon S3 file system. For example,

```
<property>
    <name>fs.s3.awsAccessKeyId</name>
    <value>[Your Access Key]</value>
 </property>
```

fs.s3.awsSecretAccessKey

The password for the Blaze and Spark engines to connect to the Amazon S3 file system. For example,

```
<property>
    <name>fs.s3.awsSecretAccessKey</name>
    <value>[Your Access Id]</value>
</property>
```

When you use a server side encryption protocol for Hive buckets, enable the server side encryption property to enable access to encrypted Hive buckets.

Configure the following property:
fs.s3.enableServerSideEncryption

Set to TRUE to enable server side encryption. For example,

```
<property>
    <name>fs.s3.enableServerSideEncryption</name>
    <value>TRUE</value>
 </property>
```

Update Files on the Hadoop Environment for Amazon EMR

You can configure additional functionality through the configuration files on the Hadoop environment.

When you update the configuration files, update the files on every node in the cluster. You can find the configuration files in the `/etc/hadoop` directory of the Hadoop distribution. Changes to the files take effect after you restart the Hadoop cluster and services. Ensure that all dependent services start when you restart the cluster.

Update core-site.xml

Update the core-site.xml file in the Hadoop environment to enable access to Hive tables in Amazon S3 buckets.

Configure Access to Hive Tables in Amazon S3 Buckets

Configure the AWS access key to run mappings with sources and targets on Hive tables in Amazon S3 buckets.

Use an s3 URI to connection to Amazon S3 buckets on Amazon EMR. For more information about using an s3 URI, see the Amazon note at the following URL:
https://aws.amazon.com/premiumsupport/knowledge-center/emr-file-system-s3/.

Note: To use a Hive table as a target on Amazon S3, grant write permission to the bucket through bucket policies, or add these properties to the configuration file. You must add these properties to the core-site.xml file on the Hadoop environment and to the yarn-site.xml file on the domain environment.

Configure the following properties:

fs.s3.awsAccessKeyId

The ID for the Blaze and Spark engines to connect to the Amazon S3 file system. For example,

```
<property>
    <name>fs.s3.awsAccessKeyId</name>
    <value>[Your Access Key]</value>
 </property>
```

fs.s3.awsSecretAccessKey

The password for the Blaze and Spark engines to connect to the Amazon S3 file system. For example,

```
<property>
    <name>fs.s3.awsSecretAccessKey</name>
    <value>[Your Access Id]</value>
</property>
```

When you use a server side encryption protocol for Hive buckets, enable the server side encryption property to enable access to encrypted Hive buckets.

Configure the following property:

fs.s3.enableServerSideEncryption

Set to TRUE to enable server side encryption. For example,

```
<property>
    <name>fs.s3.enableServerSideEncryption</name>
    <value>TRUE</value>
 </property>
```

Update yarn-site.xml

Update the yarn-site.xml file on the Hadoop environment to configure functionality such as dynamic resource allocation and virtual memory limits.

Allocate Cluster Resources for the Blaze Engine

Update the following properties to verify that the cluster allocates sufficient memory and resources for the Blaze engine:

yarn.nodemanager.resource.memory-mb

The maximum RAM available for each container. Set the maximum memory on the cluster to increase resource memory available to the Blaze engine. Configure this property on the following nodes:

- Run-time nodes. Set to at least 10 GB.
- Management node. Set to at least 13 GB.

yarn.nodemanager.resource.cpu-vcores

The number of virtual cores for each container. The number might correspond to the number of physical cores, but you can increase the value to allow for more processing. Configure this property on the following nodes:

- Run-time nodes. Set to at 6.
- Management node. Set to 9.

yarn.scheduler.minimum-allocation-mb

The minimum RAM available for each container. Set the minimum memory to allow the VM to spawn sufficient containers. Configure this property to be no less than 4 GB of the maximum memory on the run-time and management nodes.

Configure Virtual Memory Limits for the Blaze Engine

Configure the following property to remove virtual memory limits for every node in the Hadoop cluster:
yarn.nodemanager.vmem-check-enabled

Determines virtual memory limits. Set the value to false to disable virtual memory limits. For example,

```
<property>
    <name>yarn.nodemanager.vmem-check-enabled</name>
    <value>false</value>
    <description>Enforces virtual memory limits for containers.</description>
</property>
```

Set S3 Bucket Access Policies

To run mappings using a Hadoop connection and the Spark run-time engine, set S3 bucket access policies to allow any user read and write access.

S3 bucket access policies allow you to control user access to S3 buckets and the actions that users can perform.

1. In the Amazon AWS interface, select the bucket that you want to set policies for.
2. Click **Add bucket policy**. Or, if the bucket already has an access policy, click **Edit bucket policy**.
3. Type the bucket policy to grant read and write access to all users. Then click **Save**.

For example,

```
{
"Version": "<date>",
"Id": "Allow",
"Statement": [
{ "Sid": "<Statement ID>", "Effect": "Allow", "Principal": "*", "Action": "s3:*",
"Resource": [ "arn:aws:s3:::<bucket name>/*", "arn:aws:s3:::<bucket name>" ] }
]
}
```

In the example, Principal is set to `"*"` to grant access to the bucket to all users.

Developer Tool Tasks for Amazon EMR

Perform tasks on the machine that hosts the Developer tool. The tasks that you complete enable the Developer tool and the Data Integration Service to connect to the Hadoop environment.

Create Connections

Define a Hadoop connection to run a mapping in the Hadoop environment.

1. Log in to the Developer tool.
2. Click **Window** > **Preferences**.
3. Select **Informatica** > **Connections**.
4. Expand the domain in the Available Connections list.

5. Select the **Cluster** connection type from the list and click **Add**.
   The **New Cluster Connection** dialog box appears.
6. Enter the general properties for the connection.
7. Click **Next**.
8. Enter the hadoop cluster properties and the common properties for the Hadoop connection.
9. Click **Next**.
10. Enter the Hive pushdown configuration properties and the Hive configuration.
11. Click **Next**.
12. Enter properties for the Blaze engine.
13. Enter properties for the Spark engine.
14. Click **Finish**.

For information about connection properties, see Appendix A, “Connection Property Reference” on page 186.

Update Files on the Developer Tool

Update developerCore.ini on the machine that hosts the Developer tool.

developerCore.ini

Edit developerCore.ini to enable communication between the Developer tool and the Hadoop cluster. You can find developerCore.ini in the following directory: `<Informatica installation directory>\clients\DeveloperClient`

Add the following property:

```
-DINFA_HADOOP_DIST_DIR=hadoop\<distribution><version>
```

The change takes effect when you restart the Developer tool.

Complete Upgrade Tasks

If you upgraded the Informatica platform or applied the hotfix, you need to perform some additional tasks within the Informatica domain.

Based on the version that you upgraded from, perform the following tasks:

Update the hadoopEnv.properties file.

Applies to all upgrades. The 10.1.1 HotFix 1 hadoopEnv.properties file contains additional properties. You need to manually update it to include customized configuration from previous versions.

Complete connection upgrades.

Applies to upgrades from 9.6.1 or any 9.6.1 hotfix. You need to generate Hadoop connections from the Hive connection and update parameter files.

Update connection properties.

Applies to any upgrade. If you did not create connections when you ran the Hadoop Configuration Manager, you must update values for connection properties that changed.

Replace connections.

Applies to any upgrade. If you created connections when you ran the Hadoop Configuration Manager, you need to replace connections in mappings with the new connections.

Update the hadoopEnv.properties File

When you run the Informatica upgrade or apply the hotfix on a machine that hosts the Data Integration Service, the installer creates a new hadoopEnv.properties file and backs up the existing configuration file.

You need to edit the version 10.1.1 HotFix 1 hadoopEnv.properties file to include any manual configuration that you performed in the previous version for the corresponding distribution.

If you applied the hotfix, you can find the backup hadoopEnv.properties file in the following location:

```
<Informatica installation directory>/BACK_FROM_HF1/services/shared/hadoop/<Hadoop
distribution name>_<version>/infaConf
```

If you upgraded from a previous version, you can find the backup hadoopEnv.properties file in the following location:

```
<Previous Informatica installation directory>/services/shared/hadoop/<Hadoop distribution
name>_<version>/infaConf
```

You can find the 10.1.1 HotFix 1 hadoopEnv.properties file in the following location:

```
<Informatica installation directory>/services/shared/hadoop/<Hadoop distribution
name>_<version>/infaConf
```

Complete Connection Upgrade

Effective in version 10.0, Big Data Management requires a Hadoop connection to run mappings on the Hadoop cluster. If you upgraded from 9.6.1 or any 9.6.1 hotfix, you must generate Hadoop connections from Hive connections that are enabled to run mappings in the Hadoop environment.

The upgrade process generates a connection name for the Hadoop connection and replaces the connection name in the mappings. It does not create the physical connection object. When the upgrade is complete, you must run a command to generate the connection. Generate Hadoop connections from Hive connections if the following conditions are true:

- You upgraded from 9.6.1 or any 9.6.1 hotfix.
- The Hive connections are configured to run mappings in the Hadoop environment.

Complete the following tasks to upgrade connections:

1. Run `infacmd isp generateHadoopConnectionFromHiveConnection` to generate a Hadoop connection from a Hive connection that is configured to run in the Hadoop environment.
   The command renames the connection as follows: "Autogen_<Hive connection name>." If the renamed connection exceeds the 128 character limit, the command fails.
2. If the command fails, complete the following tasks:
   a. Rename the connection to meet the character limit and run the command again.
   b. Run `infacmd dis replaceMappingHadoopRuntimeConnections` to replace connections associated with mappings that are deployed in applications.
   c. Run `infacmd mrs replaceMappingHadoopRuntimeConnections` to replace connections associated with mappings that you run from the Developer tool.

3. If the Hive connection was parameterized, you must update the connection names in the parameter file. Verify that the Hive sources, Hive targets, and the Hive engine parameters are updated with the correct connection name.
4. If any properties changed in the cluster, such as host names, URIs, or port numbers, you must update the properties in the connections.

For information about the infacmd commands, see the *Informatica Command Reference*.

Update Connection Properties

If you upgraded from version 10.0 or later, and you did not create connections when you ran the Hadoop Configuration Manager, you need to verify connection properties.

Review the Hadoop, Hive, HDFS, and HBase connections that you created in a previous release to update the values for connection properties. For example, if you added nodes to the cluster or if you updated the distribution version, you might need to verify host names, URIs, or port numbers for some of the properties.

Replace the Connections

If you created connections when you ran the Hadoop Configuration Manager, you need to replace connections in mappings with the new connections.

The method that you use to replace connections in mappings depends on the type of connection.

Hadoop connection

If you created a Hadoop connection when you ran the Hadoop Configuration Manager, run the following commands:

- `infacmd dis replaceMappingHadoopRuntimeConnections`. Replaces connections associated with mappings that are deployed in applications.
- `infacmd mrs replaceMappingHadoopRuntimeConnections`. Replaces connections associated with mappings that you run from the Developer tool.

For information about the infacmd commands, see the *Informatica Command Reference*.

Hive, HDFS, or HBase connection

If you created a Hive, HDFS, or HBase connection when you ran the Hadoop Configuration Manager, you must replace the connections manually.

CHAPTER 5

Uninstall for Amazon EMR

This chapter includes the following topics:

- Before You Uninstall, 50
- Uninstall Big Data Management, 50

Before You Uninstall

Verify prerequisites before you uninstall Big Data Management.

1. Verify that the Big Data Management administrator can run `sudo` commands.
2. If you are uninstalling Big Data Management in a cluster environment, configure the root user to use a passwordless Secure Shell (SSH) connection between the machine where you want to run the Big Data Management uninstall and all of the nodes where Big Data Management is installed.
3. If you are uninstalling Big Data Management in a cluster environment using the `HadoopDataNodes` file, verify that the `HadoopDataNodes` file contains the IP addresses or machine host names of each of the nodes in the Hadoop cluster from which you want to uninstall Big Data Management. The `HadoopDataNodes` file is located on the node from where you want to launch the Big Data Management installation. You must add one IP address or machine host name of the nodes in the Hadoop cluster for each line in the file.

Uninstall Big Data Management

You can uninstall Big Data Management from a single node or cluster environment.

1. Log in to the machine as root user. The machine you log in to depends on the Big Data Management environment and uninstallation method.
   - To uninstall in a single node environment, log in to the machine on which Big Data Management is installed.
   - To uninstall in a cluster environment using the HADOOP_HOME environment variable, log in to the primary name node.
   - To uninstall in a cluster environment using the `HadoopDataNodes` file, log in to any node.

2. Run the following command to start the uninstallation in console mode:

```
bash InformaticaHadoopInstall.sh
sh InformaticaHadoopInstall.sh
./InformaticaHadoopInstall.sh
```

3. Press **y** to accept the Big Data Management terms of agreement.
4. Press **Enter**.
5. Select **3** to uninstall Big Data Management.
6. Press **Enter**.
7. Select the uninstallation option, depending on the Big Data Management environment:
   - Select **1** to uninstall Big Data Management from a single node environment.
   - Select **2** to uninstall Big Data Management from a cluster environment.
8. Press **Enter**.
9. If you are uninstalling Big Data Management in a cluster environment, select the uninstallation option, depending on the uninstallation method:
   - Select **1** to uninstall Big Data Management from the primary name node.
   - Select **2** to uninstall Big Data Management using the `HadoopDataNodes` file.
10. Press **Enter**.
11. If you are uninstalling Big Data Management from a cluster environment from the primary name node, type the absolute path for the Hadoop installation directory. Start the path with a slash.

The uninstaller deletes all of the Big Data Management binary files from the following directory: `/<Big Data Management installation directory>/Informatica`
In a cluster environment, the uninstaller delete the binary files from all of the nodes within the Hadoop cluster.

Part III: Install and Configure for Azure HDInsight

This part contains the following chapters:

- Installation for Azure HDInsight, 53
- Hadoop Configuration Manager for Azure HDInsight, 56
- Manual Configuration Tasks for Azure HDInsight, 66
- Uninstall for Azure HDInsight, 75

CHAPTER 6

Installation for Azure HDInsight

This chapter includes the following topics:

- Installation for Azure HDInsight Overview, 53
- Download the Distribution Package, 53
- Install on a Single Node , 54
- Install on a Cluster Using SCP , 54
- Install on a Cluster Using NFS, 54
- Install on a Cluster Using the HadoopDataNodes File, 55

Installation for Azure HDInsight Overview

You can install Big Data Management for Azure HDInsight in different environments.

If you are installing Big Data Management for the first time or are upgrading, you need to install the Big Data Management binaries on the Hadoop environment.

Install Big Data Management on Azure HDInsight in one of the following ways:

- Install on a single node.
- Install on a cluster from the primary name node using SCP protocol.
- Install on a cluster from the primary name node using NFS protocol.
- Install on a cluster using the HadoopDataNodes file.

Download the Distribution Package

To install Big Data Management on Ubuntu Hadoop distributions on Azure HDInsight, download the tar.gz file that includes a Debian package and the binary files that you need.

1. Create a temporary directory in a shared location on a local disk.
2. Download the following file: `InformaticaHadoop-<version>-Deb.tar.gz`
3. Extract the file to the machine from where you want to distribute the Debian package and run the Big Data Management installation.

Install on a Single Node

You can install Big Data Management in a single node environment.

1. Log in to the machine as root user or as a user with sudo privileges.
2. Run one of the following commands to start the installation:

```
bash InformaticaHadoopInstall.sh
sh InformaticaHadoopInstall.sh
./InformaticaHadoopInstall.sh
```

3. Press **y** to accept the Big Data Management terms of agreement.
4. Press **Enter**.
5. Press **1** to install Big Data Management in a single node environment.
6. Press **Enter**.

To get more information about the tasks performed by the installer, you can view the `informatica-hadoop-install.<DateTimeStamp>.log` installation log file.

Install on a Cluster Using SCP

You can install Big Data Management in a cluster environment from the primary namenode using the SCP protocol.

1. Log in to the primary NameNode as root user or as a user with root privileges.
2. Run one of the following commands to start the installation:

```
bash InformaticaHadoopInstall.sh
sh InformaticaHadoopInstall.sh
./InformaticaHadoopInstall.sh
```

3. Press **y** to accept the Big Data Management terms of agreement.
4. Press **Enter**.
5. Press **2** to install Big Data Management in a cluster environment.
6. Press **Enter**.
7. Press **1** to install Big Data Management from the primary namenode.
8. Press **Enter**.

The installer installs Big Data Management in the HDInsight Hadoop cluster. The SCP utility copies the product binaries to every node on the cluster in the following directory: `/opt/Informatica`.

You can view the `informatica-hadoop-install.<DateTimeStamp>.log` installation log file to get more information about the tasks performed by the installer.

Install on a Cluster Using NFS

You can install Big Data Management in a cluster environment from the primary NameNode using NFS.

1. Log in to the primary NameNode as root user or as a user with sudo privileges.

2. Run one of the following commands to start the installation:

```
bash InformaticaHadoopInstall.sh
sh InformaticaHadoopInstall.sh
./InformaticaHadoopInstall.sh
```

3. Press **y** to accept the Big Data Management terms of agreement.
4. Press **Enter**.
5. Press **2** to install Big Data Management in a cluster environment.
6. Press **Enter**.
7. Press **1** to install Big Data Management from the primary NameNode.
8. Press **Enter**.

You can view the `informatica-hadoop-install.<DateTimeStamp>.log` installation log file to get more information about the tasks performed by the installer.

Install on a Cluster Using the HadoopDataNodes File

You can install Big Data Management in a cluster environment using the HadoopDataNodes file.

1. Log in to the machine as the root user.
2. Update the `HadoopDataNodes` file that was extracted with the download package. Add an entry for each IP address or machine host name in the cluster.
3. Run one of the following commands to start the installation:

```
bash InformaticaHadoopInstall.sh
sh InformaticaHadoopInstall.sh
./InformaticaHadoopInstall.sh
```

4. Press **y** to accept the Big Data Management terms of agreement.
5. Press **Enter**.
6. Press **2** to install Big Data Management in a cluster environment.
7. Press **Enter**.

CHAPTER 7

Hadoop Configuration Manager for Azure HDInsight

This chapter includes the following topics:

- Hadoop Configuration Manager Overview, 56
- Start the Configuration Manager in Console Mode, 56
- Run the Configuration Manager in Silent Mode, 61

Hadoop Configuration Manager Overview

After you install Big Data Management on the Hadoop environment, you configure Big Data Management on the Informatica domain.

Configure Big Data Management using the Hadoop Configuration Manager (configuration manager). The configuration manager can create connections between the Hadoop environment and the Data Integration Service, updates Data Integration Service properties, and it updates configuration files on the Data Integration Service machine.

You can run the configuration manager in console mode or silent mode.

After you run the configuration manager, you must perform some additional configuration tasks.

Start the Configuration Manager in Console Mode

Run the Hadoop Configuration Manager in console mode to configure the Informatica domain for Big Data Management.

1. On the machine where the Data Integration Service runs, open the command line.
2. Go to the following directory: `<Informatica installation directory>/tools/BDMUtil`
3. Run `BDMConfig.sh`.
4. Choose option 5 to configure for Azure HDInsight.
5. In the **Distribution Folder Selection** section, choose the directory of the Azure HDInsight distribution that you want to configure.

Note: The distribution might be stored in a directory that uses a different distribution version number.

6. In the **Connection Type** section, select the option to access files on the Hadoop cluster.

Option	Description
1	Apache Ambari. Select this option to use the Ambari REST API to access files on the Hadoop cluster. Informatica recommends that you use this option.
2	Secure Shell (SSH). Select this option to use SSH to access files on the Hadoop cluster. This option requires SSH connections from the Informatica domain to the machines that host the NameNode, YARN ResourceManager, and Hive client. If you select this option, Informatica recommends that you use an SSH connection without a password or have sshpass or Expect installed.

Complete the configuration based on the option that you choose.

Complete the Configuration through Apache Ambari

To complete the configuration, you can choose to update Data Integration Service properties and create connection objects.

You might want to create connections if you are installing for the first time or if you are upgrading and the changes to the Hadoop environment are significant, such as a different distribution. If you are upgrading and you create connections that you want to use in existing mappings, you need to update mappings to use the new connections after you upgrade.

You might want to skip creating connections if the changes to the Hadoop environment are minimal, such as a change in distribution version. If you are upgrading and you choose not to create connections, you can continue to use the connections in existing mappings. However, you need update any changed properties after you upgrade. Property changes might include host names, URIs, or port numbers.

1. In the **Ambari Administration Information** section, enter the connection information to connect to the Ambari Manager.
   a. Enter the Ambari Manager host name or IP address and port.
   b. Enter the Ambari user ID.
   c. Enter the password for the user ID.
   d. Enter the port for Ambari Manager.
   e. Select whether to use MapReduce or Tez as the execution engine type.
      - 1 - MapReduce
      - 2 - Tez

   The Hadoop Configuration Manager retrieves the required information from the Hadoop cluster.
2. In the **Hadoop Configuration Manager Output** section, select whether you want to update Data Integration Service properties.

   Select from the following options:

Option	Description
1	No. Select this option to update Data Integration Service properties later.
2	Yes. Select this option to update Data Integration Service properties now.

3. Select whether you want to restart the Data Integration Service.

   Select from the following options:

Option	Description
1	No. Select this option if you do not want the configuration manager to restart the Data Integration Service.
2	Yes. Select this option if you want the configuration manager to restart the Data Integration Service when the configuration is complete.

4. Select whether you want to create connections for Big Data Management.

   Select from the following options:

Option	Description
1	No. Select this option if you do not want to create connections. If you are upgrading and you choose not to create connections, you can continue to use the connections in existing mappings. However, you need update any changed properties after you upgrade. Property changes might include host names, URIs, or port numbers.
2	Yes. Select this option if you want to create connections. If you are upgrading and you create connections that you want to use in existing mappings, you need to update mappings to use the new connections after you upgrade.

5. In the **Create Connections** section, select the connection type to create Big Data Management connections:

Option	Description
1. Hive	Create a Hive connection to access Hive as a source or target.
2. HDFS	Create an HDFS connection to read data from or write data to the HDFS file system on a Hadoop cluster.
3. Hadoop	Create a Hadoop connection to run mappings in the Hadoop environment.
4. HBase	Create an HBase connection to access HBase.
5. Select all	Create all four types of connection.

6. In the **Connection Details** section, provide the connection properties.

   Based on the type of connection that you choose to create, the Hadoop Configuration Manager requires different properties.

   The Hadoop Configuration Manager creates the connections.

7. The Hadoop Configuration Manager reports a summary of its operations, including whether connection creation succeeded and the location of log files.

The Hadoop Configuration Manager creates the following file in the `<Informatica installation directory>/tools/BDMUtil` directory:

ClusterConfig.properties.<timestamp>

Contains details about the properties fetched from the Hadoop cluster, including cluster node names, to provide templates for connection creation commands. To use these connection creation templates to

create connections to the Hadoop cluster, edit the connection name, domain username and password in the generated commands.

Complete the Configuration through SSH

When you complete configuration through SSH, you must provide host names and Hadoop configuration file locations.

Configuration through SSH requires SSH connections from the Informatica domain to the machines that host the NameNode, YARN ResourceManager, and Hive client. Informatica recommends that you use an SSH connection without a password or have sshpass or Expect installed. If you do not use one of these methods, you must enter the password each time the utility downloads a file from the Hadoop cluster.

1. Enter the name node host name.
2. Enter the SSH user ID.
3. Enter the password for the SSH user ID, or press enter if you use an SSH connection without a password.
4. Enter the location for the `hdfs-site.xml` file on the Hadoop cluster.
5. Enter the location for the `core-site.xml` file on the Hadoop cluster.

   The Hadoop Configuration Manger connects to the name node and downloads the following files: `hdfs-site.xml` and `core-site.xml`.
6. Enter the Yarn resource manager host name.

   Note: Yarn resource manager was formerly known as JobTracker.
7. Enter the SSH user ID.
8. Enter the password for the SSH user ID, or press enter if you use an SSH connection without a password.
9. Enter the directory for the `mapred-site.xml` file on the Hadoop cluster.
10. Enter the directory for the `yarn-site.xml` file on the Hadoop cluster.

    The utility connects to the JobTracker and downloads the following files: `mapred-site.xml` and `yarn-site.xml`.
11. Enter the Hive client host name.
12. Enter the SSH user ID.
13. Enter the password for the SSH user ID, or press enter if you use an SSH connection without a password.
14. Enter the directory for the `hive-site.xml` file on the Hadoop cluster.

    The configuration manager connects to the Hive client and downloads the following file: `hive-site.xml`.
15. Optionally, You can choose to configure the HBase server.

    Select the following options:

Option	Description
1	No. Select this option if you do not want to configure the HBase server.
2	Yes. Select this option to configure the HBase server.

If you select **Yes**, enter the following information to configure the HBase server:

a. Enter the HBase server host name.

b. Enter the SSH user ID.

c. Enter the password for the SSH user ID, or press enter if you use an SSH connection without a password.

d. Enter the directory for the `hbase-site.xml` file on the Hadoop cluster.

16. In the **Create Connections** section, select the connection type to create Big Data Management connections:

Option	Description
1. Hive	Create a Hive connection to access Hive as a source or target.
2. HDFS	Create an HDFS connection to read data from or write data to the HDFS file system on a Hadoop cluster.
3. Hadoop	Create a Hadoop connection to run mappings in the Hadoop environment.
4. HBase	Create an HBase connection to access HBase.
5. Select all	Create all four types of connection.

Press the number that corresponds to your choice.

17. The **Domain Information** section displays the domain name and the node name. Enter the following additional information about Informatica domain.

a. Enter the domain user name.

b. Enter the domain password.

c. Enter the Data Integration Service name.

d. If the Hadoop cluster uses Kerberos authentication, enter the following information:

- Hadoop Kerberos service principal name
- Hadoop Kerberos keytab location. Location of the keytab on the Data Integration Service machine.

Note: After you enter the Data Integration Service name, the utility tests the domain connection, and then recycles the Data Integration Service.

18. In the **Connection Details** section, provide the connection properties.

Based on the type of connection you choose to create, the utility requires different properties.

The Hadoop Configuration Manager creates the connections.

19. The Hadoop Configuration Manager reports a summary of its operations, including whether connection creation succeeded and the location of log files.

The Hadoop Configuration Manager creates the following file in the `<Informatica installation directory>/tools/BDMUtil` directory:

ClusterConfig.properties.<timestamp>

Contains details about the properties fetched from the Hadoop cluster, including cluster node names, to provide templates for connection creation commands. To use these connection creation templates to create connections to the Hadoop cluster, edit the connection name, domain username and password in the generated commands.

Run the Configuration Manager in Silent Mode

To configure the Big Data Management without user interaction in silent mode, run the Hadoop Configuration Manager in silent mode. Use the `SilentInput.properties` file to specify the configuration options. The configuration manager reads the file to determine the configuration options.

To configure the domain for Big Data Management in silent mode, perform the following tasks:

1. Configure the properties in the `SilentInput.properties` file.
2. Run the configuration manager with the `SilentInput.properties` file.

Configure the Properties File

The `SilentInput.properties` file includes the properties that are required by the configuration manger. Customize the properties file to specify the options for your configuration and then run the configuration manager in silent mode.

You might want to create connections if you are installing for the first time or if you are upgrading and the changes to the Hadoop environment are significant, such as a different distribution. If you are upgrading and you create connections that you want to use in existing mappings, you need to update mappings to use the new connections after you upgrade.

You might want to skip creating connections if the changes to the Hadoop environment are minimal, such as a change in distribution version. If you are upgrading and you choose not to create connections, you can continue to use the connections in existing mappings. However, you need update any changed properties after you upgrade. Property changes might include host names, URIs, or port numbers.

1. Find the sample `SilentInput.properties` file in the following location: `<Informatica installation directory>/tools/BDMUtil`
2. Create a backup copy of the `SilentInput.properties` file.
3. Use a text editor to open the file and modify the values of the configuration properties.
   The following table describes the configuration properties that you can modify:

Property Name	Description
CLOUDERA_SELECTION	Indicates whether to configure the Informatica domain for the Cloudera CDH cluster. To configure the domain for the Cloudera CDH cluster, set the value of this property to 1.
HORTONWORKS_SELECTION	Indicates whether to configure the Informatica domain for the Hortonworks cluster. To configure the domain for the Hortonworks cluster, set the value of this property to 1.
BIG_INSIGHT	Indicates whether to configure the Informatica domain for the BigInsights cluster. To configure the domain for the BigInsights cluster, set the value of this property to 1.
HD_INSIGHT	Indicates whether to configure the Informatica domain for the HDInsight cluster. To configure the domain for the HDInsight cluster, set the value of this property to 1.

Property Name	Description
DIST_FOLDER_NAME	Indicates the Hadoop distribution folder name present in INFA_HOME/services/shared/hadoop.
INSTALL_TYPE	Indicates how to access files on the Hadoop cluster. Set the value depending on the Hadoop distribution. To access files on the Cloudera CDH cluster using the Cloudera Manager, set the value of this property to 0. To access files on the Hortonworks HDP, IBM BigInsights, or the Azure HDInsight cluster using Apache Ambari, set the value of this property to 3.
CLOUDERA_HOSTNAME	Required if you set INSTALL_TYPE=0. Indicates the host name of the Cloudera Manager.
CLOUDERA_USER_NAME	Required if you set INSTALL_TYPE=0. Indicates the user id for the Cloudera Manager.
CLOUDERA_USER_PASSWD	Required if you set INSTALL_TYPE=0. Indicates the password for the Cloudera Manager.
CLOUDERA_PORT	Required if you set INSTALL_TYPE=0. Indicates the port of the Cloudera Manager.
CLOUDERA_SINGLECLUSTER_NAME	Required if you set INSTALL_TYPE=0. Indicates the name of the Cloudera CDH cluster for which you want to configure Big Data Management. If the Cloudera Manager manages a single cluster, you do not have to set this property. Retain this property as a comment or remove the property from the file. If the Cloudera Manager manages more than one cluster, uncomment and set the property to the name of the cluster for which you want to configure Big Data Management. Note: In a multi-cluster environment, the configuration fails if the property is commented or if the property has an invalid cluster name as the value.
AMBARI_HOSTNAME	Required if you set INSTALL_TYPE =3. Indicates the password for Apache Ambari.
AMBARI_USER_NAME	Required if you set INSTALL_TYPE=3. Indicates the port number for Apache Ambari.
AMBARI_USER_PASSWD	Required if you set INSTALL_TYPE=3. Indicates the password for Apache Ambari.
AMBARI_PORT	Required if you set INSTALL_TYPE=3. Indicates the port number for Apache Ambari.
TEZ_EXECUTION_MODE	Required if you set INSTALL_TYPE=3. Indicates whether to enable Tez for the Hive engine. To enable Tez on the Hortonworks HDP or Azure HDInsight cluster, set the value of this property to 1. Otherwise, set the value to 0.

Property Name	Description
CREATE_CONNECTION	Indicates whether to update the Data Integration Service and create connections for Big Data Management. To update the Data Integration Service and create connections, set the value of this property to 1 and define the connection properties. If you are upgrading and you choose not to create connections, you can continue to use the connections in existing mappings. However, you need update any changed properties after you upgrade. Property changes might include host names, URIs, or port numbers. If you are upgrading and you create connections that you want to use in existing mappings, you need to update mappings to use the new connections after you upgrade. To exit the configuration manager, set the value to 0.
UPDATE_DIS	Required if you set CREATE_CONNECTION=1. Indicates whether to update the Data Integration Service properties. To update the Data Integration Service, set this property to 1. Otherwise, set the value to 0.
CREATE_BDM_CONNECTIONS	Required if you set CREATE_CONNECTION=1. Set this property to 1 to create Hadoop, HDFS, HBase, and Hive connections. Otherwise, the configuration manager will not create connections.
DOMAIN_USER	Required if you set CREATE_CONNECTION=1. Indicates the user name for the domain administrator.
DOMAIN_PSSWD	Required if you set CREATE_CONNECTION=1. Indicates the password for the domain administrator.
DIS_SERVICE_NAME	Required if you set CREATE_CONNECTION=1 and UPDATE_DIS=1. Indicates the name of the Data Integration Service.
CLUSTER_INSTALLATION_DIR	Required if you set CREATE_CONNECTION=1 and UPDATE_DIS=1. Indicates the Informatica home directory on every node on the Hadoop cluster that is created during the Big Data Management installation. Default is /opt/Informatica.
KERBEROS_PRINCIPAL_NAME	Required if you set CREATE_CONNECTION=1 and UPDATE_DIS=1 and if the Hadoop cluster uses Kerberos authentication. Indicates the Hadoop Kerberos service principal name.
KERBEROS_KEYTAB	Required if you set CREATE_CONNECTION=1 and UPDATE_DIS=1 and if the Hadoop cluster uses Kerberos authentication. Indicates the Hadoop Kerberos keytab location.
HIVE_CONNECTION	Required if you set CREATE_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Set this property to 1 to create a Hive connection. Otherwise, set the value to 0.

Property Name	Description
HDFS_CONNECTION	Required if you set CREATE_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Set this property to 1 to create a HDFS connection. Otherwise, set the value to 0.
HADOOP_CONNECTION	Required if you set CREATE_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Set this property to 1 to create a Hadoop connection. You must create a Hadoop connection to run the mappings in the Hadoop environment.
HBASE_CONNECTION	Required if you set CREATE_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Set this property to 1 to create a HBASE connection. Otherwise, set the value to 0.
SELECT_ALL_CONNECTION	Required if you set CREATE_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Set this property to 1 to create all listed connections. Also set HIVE_CONNECTION=1, HDFS_CONNECTION=1, HADOOP_CONNECTION=1, and HBASE_CONNECTION=1 .
SAMPLE_HIVE_CONNECTION	Required if CREATE_CONNECTION=1, HIVE_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Indicates connection name for Hive.
HIVE_EXECUTION_MODE	Required if you set CREATE_CONNECTION=1, HIVE_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Indicates the execution mode for Hive. Set this property to either Local or Remote.
HIVE_USER_NAME	Required if you set CREATE_CONNECTION=1, HIVE_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Indicates the user name for Hive. You must set the execution mode to Remote.
SAMPLE_HDFS_CONNECTION	Required if you set CREATE_CONNECTION=1, HDFS_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Indicates the connection name for HDFS.
HDFS_USER_NAME	Required if you set CREATE_CONNECTION=1, HDFS_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Indicates the user name to access HDFS.
SAMPLE_HADOOP_CONNECTION	Required if you set CREATE_CONNECTION=1 , HADOOP_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Indicates the connection name for Hadoop.
BLAZE_WORKING_DIR	Required if you set CREATE_CONNECTION=1 , HADOOP_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Indicates the existing Blaze work directory on HDFS.

Property Name	Description
BLAZE_USER	Required if you set CREATE_CONNECTION=1, HADOOP_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Indicates the user name for Blaze.
SPARK_HDFS_STAGING_DIR	Required if you set CREATE_CONNECTION=1 , HADOOP_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Indicates the existing Spark HDFS staging directory.
SPARK_EVENTLOG_DIR	Required if you set CREATE_CONNECTION=1, HADOOP_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Optional: Indicates the Spark event log directory
SPARK_PARAMETER_LIST	Required if you set CREATE_CONNECTION=1, HADOOP_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Indicates the Spark execution parameters list.
IMPERSONATION_USER	Required if you set CREATE_CONNECTION=1, HADOOP_CONNECTION=1, CREATE_BDM_CONNECTIONS=1, and if the Hadoop cluster uses Kerberos authentication.
SAMPLE_HBASE_CONNECTION	Required if you set CREATE_CONNECTION=1 , HBASE_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Indicates the connection name for HBase.
ZOOKEEPER_HOSTS	Required if you set CREATE_CONNECTION=1 , HBASE_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Indicates the ZooKeeper host.

Run the Configuration Manager

After you configure the properties file, open a command prompt to start the silent configuration.

1. Go to the following directory: `<Informatica installation directory>/tools/BDMUtil`
2. Verify that the directory contains the `SilentInput.properties` file that you edited.
3. Run the `BDMSilentConfig.sh` file.

The installer runs in the background. The process can take a while. The silent configuration is complete when the following log files are created in the `<Informatica installation directory>/tools/BDMUtil` directory:

- Informatica_Big_Data_Edition_Configuration_Utility.log.<timestamp>
- ClusterConfig.properties.<timestamp>

The silent configuration fails if you incorrectly configure the properties file or if the installation directory is not accessible. The `silenterror.log` file is created in the home directory of the UNIX user that runs the configuration manager in silent mode. Correct the errors and then run the silent configuration again.

CHAPTER 8

Manual Configuration Tasks for Azure HDInsight

This chapter includes the following topics:

- Update Files on the Hadoop Environment for Azure HDInsight, 66
- Import Files for Data Decryption, 68
- Update Files on the Domain Environment for Azure HDInsight, 69
- Create a Softlink for the core-site.xml, hdfs-site.xml and mapred-site.xml Files, 71
- Update Files on the Developer Tool, 72
- Download the JDBC Drivers for Sqoop Connectivity, 73
- Complete Upgrade Tasks, 73

Update Files on the Hadoop Environment for Azure HDInsight

You can configure additional functionality through the configuration files on the Hadoop environment.

When you update the configuration files, update the files on every node in the cluster. You can find the configuration files in the `/etc/hadoop` directory of the Hadoop distribution. Changes to the files take effect after you restart the Hadoop cluster and services. Ensure that all dependent services start when you restart the cluster.

Update core-site.xml

Update core-site.xml on the Hadoop environment to configure functionality such as user impersonation for Sqoop mappings.

Configure Impersonation for Sqoop

To run Sqoop mappings on clusters that do not use Kerberos authentication, you must create a proxy user for the yarn user who will impersonate other users.

Configure the following properties, and then restart Hadoop services and the cluster:

hadoop.proxyuser.yarn.groups

Allows impersonation from any group.

```
<property>
    <name>hadoop.proxyuser.yarn.groups</name>
    <value><Name_of_the_impersonation_user></value>
    <description>Allows impersonation from any group.</description>
</property>
```

hadoop.proxyuser.yarn.hosts

Allows impersonation from any host.

```
<property>
    <name>hadoop.proxyuser.yarn.hosts</name>
    <value>*</value>
    <description>Allows impersonation from any host.</description>
</property>
```

Update yarn-site.xml

Update the yarn-site.xml file on the Hadoop environment to configure functionality such as virtual memory limits and dynamic resource allocation for the Spark engine.

Configure Dynamic Resource Allocation for the Spark Engine

Configure dynamic resource allocation for mappings to run on the Spark engine. Update the following properties on each node where YARN node manager is running:

yarn.nodemanager.aux-services

The auxiliary service name. Add spark_shuffle to the current entry. For example, if the current value is "mapreduce_shuffle," update it to be "mapreduce_shuffle,spark_shuffle."

```
<property>
    <name>yarn.nodemanager.aux-services</name>
    <value>mapreduce_shuffle,spark_shuffle</value>
</property>
```

yarn.nodemanager.aux-services.spark_shuffle.class

The auxiliary service class to use. Set to org.apache.spark.network.yarn.YarnShuffleService. For example,

```
<property>
    <name>yarn.nodemanager.aux-services.spark_shuffle.class</name>
    <value>org.apache.spark.network.yarn.YarnShuffleService</value>
</property>
```

yarn.application.classpath

The classpath for YARN applications. Verify that the yarn classpath includes the path to the Spark shuffle.jar file packaged with the distribution. If the .jar file is not in any Hadoop distribution directory, update the path to use the .jar file in the Apache Spark distribution directory. For example,

```
<property>
    <name>yarn.application.classpath</name>
    <value>/opt/Informatica/services/shared/spark/lib_spark_2.0.1_hadoop_2.6.0/spark-
networkshuffle_2.11-2.0.1.jar</value>
</property>
```

Configure Virtual Memory Limits for the Blaze Engine

Configure the following property to remove virtual memory limits for every node in the Hadoop cluster:

yarn.nodemanager.vmem-check-enabled

Determines virtual memory limits. Set the value to false to disable virtual memory limits. For example,

```
<property>
    <name>yarn.nodemanager.vmem-check-enabled</name>
    <value>false</value>
    <description>Enforces virtual memory limits for containers.</description>
</property>
```

Allocate Cluster Resources for the Blaze Engine

Update the following properties to verify that the cluster allocates sufficient memory and resources for the Blaze engine:

yarn.nodemanager.resource.memory-mb

The maximum RAM available for each container. Set the maximum memory on the cluster to increase resource memory available to the Blaze engine. Configure this property on the following nodes:

- Run-time nodes. Set to at least 10 GB.
- Management node. Set to at least 13 GB.

yarn.nodemanager.resource.cpu-vcores

The number of virtual cores for each container. The number might correspond to the number of physical cores, but you can increase the value to allow for more processing. Configure this property on the following nodes:

- Run-time nodes. Set to at 6.
- Management node. Set to 9.

yarn.scheduler.minimum-allocation-mb

The minimum RAM available for each container. Set the minimum memory to allow the VM to spawn sufficient containers. Configure this property to be no less than 4 GB of the maximum memory on the run-time and management nodes.

Remove the XML Parser Path for the Spark Engine

When you complete configuration through SSH, remove the `/usr/hdp/current/hadoop-hdfs-client/lib/*` from the `yarn.application.classpath` property:

The following example shows the updated `yarn.application.classpath` property:

```
<property>
<name>yarn.application.classpath</name>
<value>$HADOOP_CONF_DIR,/usr/hdp/current/hadoop-client/,
/usr/hdp/current/hadoop-client/lib/,/usr/hdp/current/hadoop-hdfs-client/,
/usr/hdp/current/hadoop-yarn-client/,
/usr/hdp/current/hadoop-yarn-client/lib/*</value>
</property>
```

Import Files for Data Decryption

When you run a mapping on the Azure HDInsight cluster that has encrypted data, you must configure files to decrypt data on the Azure HDInsight cluster.

The Data Integration Service host machine requires the following files from the Hadoop cluster to decrypt data:

- key_decryption_cert.prv

- decrypt.sh

Perform the following steps to import the files:

1. Locate these files on the cluster head node.
2. Copy the files from the Hadoop cluster to the machine that runs the Data Integration Service. The directory structure must be the same as that of the Hadoop cluster.
3. Verify that permissions on the directory are 775.

Update Files on the Domain Environment for Azure HDInsight

You can configure additional functionality through the configuration files on the domain.

When the Informatica installer creates nodes on the domain, it creates directories and files for Big Data Management. When you configure Big Data Management on the domain, you need to edit configuration files to enable functionality.

The following table describes the installation directories and the default paths:

Directory	Description
Data Integration Service Hadoop distribution directory	The Hadoop distribution directory on the Data Integration Service node. It corresponds to the distribution on the Hadoop environment. You can view the directory path in the Data Integration Service properties of the Administrator tool. Default is `<Informatica installation directory>/Informatica/services/shared/hadoop/<Hadoop distribution name>_<version>`.
Configuration file directories	Configuration files for Big Data Management are stored in the following directories: - `<Informatica installation directory>/Informatica/services/shared/hadoop/<Hadoop distribution name>_<version>/InfaConf`. Contains the hadoopEnv.properties file. - `<Informatica installation directory>/Informatica/services/shared/hadoop/<Hadoop distribution name>_<version>/conf`. Contains *-site files.

Update hadoopEnv.properties

Update the hadoopEnv.properties file to configure functionality such as Sqoop connectivity and the Spark run-time engine.

Open hadoopEnv.properties and back it up before you configure it. You can find the hadoopEnv.properties file in the following location:`<informatica installation directory>/services/shared/hadoop/<Hadoop distribution name>_<version number>/infaConf`

Configure Sqoop Connectivity

Configure the following property for Sqoop connectivity:
infapdo.env.entry.hadoop_node_jdk_home

Configure the HADOOP_NODE_JDK_HOME to represent the JDK version that the cluster nodes use. You must use JDK version 1.7 or later.

Configure the property as follows:

```
infapdo.env.entry.hadoop_node_jdk_home=HADOOP_NODE_JDK_HOME=<cluster JDK home>/
jdk<version>
```

For example,

```
infapdo.env.entry.hadoop_node_jdk_home=HADOOP_NODE_JDK_HOME=/usr/java/default
```

Configure Performance for the Spark Engine

Configure the following performance properties for the Spark engine:

spark.dynamicAllocation.enabled

Performance configuration to run mappings on the Spark engine. Enables dynamic resource allocation. Required when you enable the external shuffle service.

Set the value to TRUE.

spark.shuffle.service.enabled

Performance configuration to run mappings on the Spark engine. Enables the external shuffle service. Required when you enable dynamic resource allocation.

Set the value to TRUE.

spark.scheduler.maxRegisteredResourcesWaitingTime

Performance configuration to run mappings on the Spark engine. The number of milliseconds to wait for resources to register before scheduling a task. Reduce this from the default value of 30000 to reduce delays before starting the Spark job execution.

Set the value to 15000.

spark.scheduler.minRegisteredResourcesRatio

Performance configuration to run mappings on the Spark engine. The minimum ratio of registered resources to acquire before task scheduling begins. Reduce this from the default value of 0.8 to reduce any delay before starting the Spark job execution.

Set the value to .5.

spark.executor.instances

Performance configuration to run mappings on the Spark engine. If you enable dynamic resource allocation for the Spark engine, Informatica recommends that you convert this property to a comment. For example, `#spark.executor.instances=100`

Configure Environment Variables

You can optionally add third-party environment variables and extend the existing PATH environment variable in the hadoopEnv.properties file. The following text shows sample entries to configure environment variables:

```
infapdo.env.entry.oracle_home=ORACLE_HOME=/databases/oracle
infapdo.env.entry.db2_home=DB2_HOME=/databases/db2
infapdo.env.entry.db2instance=DB2INSTANCE=OCA_DB2INSTANCE
infapdo.env.entry.db2codepage=DB2CODEPAGE="1208"
infapdo.env.entry.odbchome=ODBCHOME=$HADOOP_NODE_INFA_HOME/ODBC7.1
infapdo.env.entry.home=HOME=/opt/thirdparty
infapdo.env.entry.gphome_loaders=GPHOME_LOADERS=/databases/greenplum
infapdo.env.entry.pythonpath=PYTHONPATH=$GPHOME_LOADERS/bin/ext
infapdo.env.entry.nz_home=NZ_HOME=/databases/netezza
infapdo.env.entry.ld_library_path=LD_LIBRARY_PATH=$HADOOP_NODE_INFA_HOME/services/
shared/bin:$HADOOP_NODE_INFA_HOME/DataTransformation/bin:$HADOOP_NODE_HADOOP_DIST/lib/
native:$HADOOP_NODE_INFA_HOME/ODBC7.1/lib:$HADOOP_NODE_INFA_HOME/jre/lib/
amd64:$HADOOP_NODE_INFA_HOME/jre/lib/amd64/server:$HADOOP_NODE_INFA_HOME/java/jre/lib/
amd64:$HADOOP_NODE_INFA_HOME/java/jre/lib/amd64/server:/databases/oracle/lib:/
databases/db2/lib64:$LD_LIBRARY_PATH
infapdo.env.entry.path=PATH=$HADOOP_NODE_HADOOP_DIST/scripts:$HADOOP_NODE_INFA_HOME/
```

```
services/shared/bin:$HADOOP_NODE_INFA_HOME/jre/bin:$HADOOP_NODE_INFA_HOME/java/jre/bin:
$HADOOP_NODE_INFA_HOME/ODBC7.1/bin:/databases/oracle/bin:/databases/db2/bin:$PATH
#teradata
infapdo.env.entry.twb_root=TWB_ROOT=/databases/teradata/tbuild
infapdo.env.entry.manpath=MANPATH=/databases/teradata/odbc_64:/databases/teradata/odbc_64
infapdo.env.entry.nlspath=NLSPATH=/databases/teradata/odbc_64/msg/%N:/databases/
teradata/msg/%N
infapdo.env.entry.pwd=PWD=/databases/teradata/odbc_64/samples/C
```

Configure mapreduce classpath

Configure the following property to include the softlinks for the core-site.xml, hdfs-site.xml, and mapred-site.xml files on all the data nodes:

INFA_MAPRED_CLASSPATH

Update the property by adding the directory where the softlink exists.

Update hive-site.xml

Update the hive-site.xml file on the domain environment to set properties for dynamic partitioning.

Configure Dynamic Partitioning

To use Hive dynamic partitioned tables, configure the following properties:

hive.exec.dynamic.partition

Enables dynamic partitioned tables. Set this value to `TRUE`.

exec.dynamic.partition.mode

Allows all partitions to be dynamic. Set this value to `nonstrict`.

Create a Softlink for the core-site.xml, hdfs-site.xml and mapred-site.xml Files

Create softlinks for the core-site.xml, hdfs-site.xml, and mapred-site.xml files on all the data nodes, and update the `hadoopEnv.properties` file on the Data Integration Service machine.

1. If you configured passwordless SSH for the root user on the cluster, use the sample script to create softlinks for the files on all the data nodes. Perform the following steps:
   a. Log in to the name node host as root user.
   b. Run the script as follows:
      `./createSoftLinks [File that contains a list of data nodes] [Directory on which you want to create a softlink]`

      For example, `./createSoftLinks /etc/hadoop/conf/slaves /opt/Informatica/conf`
2. If you have not configured passwordless SSH for the root user on the cluster, perform the following steps:
   a. On all the data nodes, create a directory. For example, `/opt/Informatica/conf`
   b. Create softlinks for the core-site.xml, hdfs-site.xml, and mapred-site.xml files from `/etc/hadoop/conf` to this directory.
3. In the `hadoopEnv.properties` file, update the `INFA_MAPRED_CLASSPATH` property by adding the directory where the softlink exists.

Sample Script

```
#!/bin/bash
#This script accepts two arguments. 1. File with list of data nodes 2. Directory on
which you want to create soft link.
Directory will be created if it doesn't exist
#Invoking the script: ./createSoftLinks.sh /etc/hadoop/conf/slaves [Directory Structure]
#The script assumes that password less SSH for root user is enabled from the node
running the script to all other nodes
datanodehosts=$1
conffiledir=$2
if [ "$1" == "" ] || [ "$2" == "" ]
then
  echo "Please provide arguments : format : ./createSoftLinks.sh 'file having list of
data nodes' 'required directory on all the data nodes"
        exit
fi

for node in $(cat $datanodehosts)
    do
        echo $node
        if ssh "$node" mkdir -p $2; then
            echo "successfully connected and created $node"
        else
            echo "ERROR: failure creating /"$2 " folder on node $node."
            echo "Aborting script..."
            exit 1
        fi
        if ssh "$node" ln -sf /etc/hadoop/conf/hdfs-site.xml $2; then
            echo "successfully created a soft link for hdfs-site.xml to data node $node"
        else
            echo "ERROR: failed to creat soft link for hdfs-site.xml"
            echo "Aborting script..."
            exit 1
        fi

        if ssh "$node" ln -sf /etc/hadoop/conf/core-site.xml $2; then
            echo "successfully created a soft link for core-site.xml to data node $node"
        else
            echo "ERROR: failed to creat soft link for core-site.xml"
            echo "Aborting script..."
            exit 1
        fi
        if ssh "$node" ln -sf /etc/hadoop/conf/mapred-site.xml $2; then
            echo "successfully created a soft link for mapred-site.xml to data node
$node"
        else
            echo "ERROR: failed to creat soft link for mapred-site.xml"
            echo "Aborting script..."
            exit 1
        fi
    done
```

Update Files on the Developer Tool

Update developerCore.ini on the Developer tool machine.

developerCore.ini

Edit developerCore.ini to enable communication between the Developer tool and the Hadoop cluster. You can find developerCore.ini in the following directory: `<Informatica installation directory>\clients\DeveloperClient`

Add the following property:

```
-DINFA_HADOOP_DIST_DIR=hadoop\<distribution><version>
```

The change takes effect when you restart the Developer tool.

Download the JDBC Drivers for Sqoop Connectivity

To configure Sqoop connectivity for relational databases, you must download JDBC driver jar files.

1. Download any Type 4 JDBC driver that the database vendor recommends for Sqoop connectivity.
2. Copy the jar files to the following directory on the machine where the Data Integration Service runs:
   `<Informatica installation directory>\externaljdbcjars`

At run time, the Data Integration Service copies the jar files to the Hadoop distribution cache so that the jar files are accessible to all nodes in the cluster.

Note: The DataDirect JDBC drivers that Informatica ships are not licensed for Sqoop connectivity.

Complete Upgrade Tasks

If you upgraded the Informatica platform or applied the hotfix, you need to perform some additional tasks within the Informatica domain.

Perform the following tasks to complete the upgrade to version 10.1.1 HotFix 1.

Update the hadoopEnv.properties file.

The 10.1.1 HotFix 1 hadoopEnv.properties file contains additional properties. You need to manually update it to include customized configuration from previous versions.

Update connection properties.

If you did not create connections when you ran the Hadoop Configuration Manager, you must update values for connection properties that changed.

Replace connections.

If you created connections when you ran the Hadoop Configuration Manager, you need to replace connections in mappings with the new connections.

Update the hadoopEnv.properties File

When you run the Informatica upgrade or apply the hotfix on a machine that hosts the Data Integration Service, the installer creates a new hadoopEnv.properties file and backs up the existing configuration file.

You need to edit the version 10.1.1 HotFix 1 hadoopEnv.properties file to include any manual configuration that you performed in the previous version for the corresponding distribution.

If you applied the hotfix, you can find the backup hadoopEnv.properties file in the following location:

```
<Informatica installation directory>/BACK_FROM_HF1/services/shared/hadoop/<Hadoop
distribution name>_<version>/infaConf
```

If you upgraded from a previous version, you can find the backup hadoopEnv.properties file in the following location:

```
<Previous Informatica installation directory>/services/shared/hadoop/<Hadoop distribution name>_<version>/infaConf
```

You can find the 10.1.1 HotFix 1 hadoopEnv.properties file in the following location:

```
<Informatica installation directory>/services/shared/hadoop/<Hadoop distribution name>_<version>/infaConf
```

Replace the Connections

If you created connections when you ran the Hadoop Configuration Manager, you need to replace connections in mappings with the new connections.

The method that you use to replace connections in mappings depends on the type of connection.

Hadoop connection

If you created a Hadoop connection when you ran the Hadoop Configuration Manager, run the following commands:

- `infacmd dis replaceMappingHadoopRuntimeConnections`. Replaces connections associated with mappings that are deployed in applications.
- `infacmd mrs replaceMappingHadoopRuntimeConnections`. Replaces connections associated with mappings that you run from the Developer tool.

For information about the infacmd commands, see the *Informatica Command Reference*.

Hive, HDFS, or HBase connection

If you created a Hive, HDFS, or HBase connection when you ran the Hadoop Configuration Manager, you must replace the connections manually.

Update Connection Properties

If you upgraded from version 10.0 or later, and you did not create connections when you ran the Hadoop Configuration Manager, you need to verify connection properties.

Review the Hadoop, Hive, HDFS, and HBase connections that you created in a previous release to update the values for connection properties. For example, if you added nodes to the cluster or if you updated the distribution version, you might need to verify host names, URIs, or port numbers for some of the properties.

CHAPTER 9

Uninstall for Azure HDInsight

This chapter includes the following topics:

- Before You Uninstall, 75
- Uninstall Big Data Management, 75

Before You Uninstall

Verify prerequisites before you uninstall Big Data Management.

1. Verify that the Big Data Management administrator can run `sudo` commands.
2. If you are uninstalling Big Data Management in a cluster environment, configure the root user to use a passwordless Secure Shell (SSH) connection between the machine where you want to run the Big Data Management uninstall and all of the nodes where Big Data Management is installed.
3. If you are uninstalling Big Data Management in a cluster environment using the `HadoopDataNodes` file, verify that the `HadoopDataNodes` file contains the IP addresses or machine host names of each of the nodes in the Hadoop cluster from which you want to uninstall Big Data Management. The `HadoopDataNodes` file is located on the node from where you want to launch the Big Data Management installation. You must add one IP address or machine host name of the nodes in the Hadoop cluster for each line in the file.

Uninstall Big Data Management

You can uninstall Big Data Management from a single node or cluster environment.

1. Log in to the machine as root user. The machine you log in to depends on the Big Data Management environment and uninstallation method.
   - To uninstall in a single node environment, log in to the machine on which Big Data Management is installed.
   - To uninstall in a cluster environment using the HADOOP_HOME environment variable, log in to the primary name node.
   - To uninstall in a cluster environment using the `HadoopDataNodes` file, log in to any node.

2. Run the following command to start the uninstallation in console mode:

```
bash InformaticaHadoopInstall.sh
sh InformaticaHadoopInstall.sh
./InformaticaHadoopInstall.sh
```

3. Press **y** to accept the Big Data Management terms of agreement.
4. Press **Enter**.
5. Select **3** to uninstall Big Data Management.
6. Press **Enter**.
7. Select the uninstallation option, depending on the Big Data Management environment:
   - Select **1** to uninstall Big Data Management from a single node environment.
   - Select **2** to uninstall Big Data Management from a cluster environment.
8. Press **Enter**.
9. If you are uninstalling Big Data Management in a cluster environment, select the uninstallation option, depending on the uninstallation method:
   - Select **1** to uninstall Big Data Management from the primary name node.
   - Select **2** to uninstall Big Data Management using the `HadoopDataNodes` file.
10. Press **Enter**.
11. If you are uninstalling Big Data Management from a cluster environment from the primary name node, type the absolute path for the Hadoop installation directory. Start the path with a slash.

The uninstaller deletes all of the Big Data Management binary files from the following directory: `/<Big Data Management installation directory>/Informatica`
In a cluster environment, the uninstaller delete the binary files from all of the nodes within the Hadoop cluster.

Part IV: Install and Configure for Cloudera CDH

This part contains the following chapters:

- Installation for Cloudera CDH, 78
- Hadoop Configuration Manager for Cloudera CDH, 80
- Manual Configuration Tasks for Cloudera CDH, 91
- Uninstall for Cloudera CDH, 104

CHAPTER 10

Installation for Cloudera CDH

This chapter includes the following topics:

- Download the Distribution Package, 78
- Install Using Cloudera Manager, 78

Download the Distribution Package

If you are installing Big Data Management for the first time or are upgrading, you need to install the Big Data Management binaries on the Hadoop environment.

Download the parcel.tar file that includes a Cloudera Parcel package and the binary files that you need to run the Big Data Management installation.

1. Create a temporary folder on a local disk.
2. Download the following file: `INFORMATICA-<version>-informatica-<version>.parcel.tar`
3. Extract `manifest.json` and the parcels from the .tar file.

Install Using Cloudera Manager

Install Big Data Management on a Cloudera CDH cluster using Cloudera Manager.

1. Log in to Cloudera Manager.
2. To verify the location of your Local Parcel Repository, click **Administration** > **Settings** > **Parcels**.
3. Create an .sha file with the parcel name and hash listed in `manifest.json` that corresponds with the Hadoop cluster.
   For example, use the following parcel name for Hadoop cluster nodes that run Red Hat Enterprise Linux 6.4 64-bit:

   ```
   INFORMATICA-<version>informatica-<version>-el6.parcel
   ```

   Use the following hash listed for Red Hat Enterprise Linux 6.4 64-bit:
   8e904e949a11c4c16eb737f02ce4e36ffc03854f

   To create an .sha file, run the following command:

   ```
   echo <hash> > <ParcelName> .sha
   ```

For example, run the following command:

```
echo "8e904e949a11c4c16eb737f02ce4e36ffc03854f" >
INFORMATICA-10.1.1-1.informatica10.1.1.1.p0.1203-el6.parcel.sha
```

4. Use FTP to transfer the parcel and the .sha file to the Local Parcel Repository.
5. Check for new parcels with Cloudera Manager by clicking **Hosts** > **Parcels**.
6. Distribute the Big Data Management parcels.
7. Activate the Big Data Management parcels.

CHAPTER 11

Hadoop Configuration Manager for Cloudera CDH

This chapter includes the following topics:

- Hadoop Configuration Manager Overview, 80
- Start the Configuration Manager in Console Mode, 80
- Run the Configuration Manager in Silent Mode, 85

Hadoop Configuration Manager Overview

After you install Big Data Management on the Hadoop environment, you configure Big Data Management on the Informatica domain.

Configure Big Data Management using the Hadoop Configuration Manager (configuration manager). The configuration manager can create connections between the Hadoop environment and the Data Integration Service, updates Data Integration Service properties, and it updates configuration files on the Data Integration Service machine.

You can run the configuration manager in console mode or silent mode.

After you run the configuration manager, you must perform some additional configuration tasks.

Start the Configuration Manager in Console Mode

Run the Hadoop Configuration Manager in console mode to configure the Informatica domain for Big Data Management.

1. On the machine where the Data Integration Service runs, open the command line.
2. Go to the following directory: `<Informatica installation directory>/tools/BDMUtil`
3. Run `BDMConfig.sh`.
4. Choose option 1 to configure for Cloudera CDH.
5. In the **Distribution Folder Selection** section, choose the directory of the Cloudera CDH distribution that you want to configure.

Note: The distribution might be stored in a directory that uses a different distribution version number.

6. In the **Connection Type** section, select the option to access files on the Hadoop cluster.

Option	Description
1	Cloudera Manager. Select this option to use the Cloudera Manager API to access files on the Hadoop cluster. Informatica recommends using this option.
2	Secure Shell (SSH). Select this option to use SSH to access files on the Hadoop cluster. This option requires SSH connections from the Informatica domain to the machines that host the NameNode, YARN ResourceManager, and Hive client. If you select this option, Informatica recommends that you use an SSH connection without a password or have sshpass or Expect installed.

Complete the configuration based on the option that you choose.

Complete the Configuration through Cloudera Manager

To complete the configuration, you can choose to update Data Integration Service properties and create connection objects.

You might want to create connections if you are installing for the first time or if you are upgrading and the changes to the Hadoop environment are significant, such as a different distribution. If you are upgrading and you create connections that you want to use in existing mappings, you need to update mappings to use the new connections after you upgrade.

You might want to skip creating connections if the changes to the Hadoop environment are minimal, such as a change in distribution version. If you are upgrading and you choose not to create connections, you can continue to use the connections in existing mappings. However, you need update any changed properties after you upgrade. Property changes might include host names, URIs, or port numbers.

1. In the **Cloudera Manager Admin Information** section, enter the following connection information to connect to the Cloudera Manager Admin Tool:
   - Cloudera Manager host
   - Cloudera user ID
   - Password for the user ID
   - Port number for Cloudera Manager

   The Hadoop Configuration Manager gets the required information from the Hadoop cluster.
2. If Cloudera Manager has multiple clusters, use the **Cloudera Cluster Name Selection** to select the Cloudera cluster that you want to configure:

Option	Description
1	cluster
2	cluster2

3. In the **Hadoop Configuration Manager Output**, select whether you want to update Data Integration Service properties.

Select from the following options:

Option	Description
1	No. Select this option to update Data Integration Service properties later.
2	Yes. Select this option to update Data Integration Service properties now. Default is 2.

4. Select whether you want to restart the Data Integration Service.

   Select from the following options:

Option	Description
1	No. Select this option if you do not want the configuration manager to restart the Data Integration Service.
2	Yes. Select this option if you want the configuration manager to restart the Data Integration Service when the configuration is complete. Default is 2.

5. Select whether you want to create connections for Big Data Management.

   Select from the following options:

Option	Description
1	No. Select this option if you do not want to create connections. If you are upgrading and you choose not to create connections, you can continue to use the connections in existing mappings. However, you need update any changed properties after you upgrade. Property changes might include host names, URIs, or port numbers.
2	Yes. Select this option if you want to create connections. If you are upgrading and you create connections that you want to use in existing mappings, you need to update mappings to use the new connections after you upgrade. Default is 2.

6. In the **Create Connections** section, select the connection type to create Big Data Management connections:

Option	Description
1. Hive	Create a Hive connection to access Hive as a source or target.
2. HDFS	Create an HDFS connection to read data from or write data to the HDFS file system on a Hadoop cluster.
3. Hadoop	Create a Hadoop connection to run mappings in the Hadoop environment.
4. HBase	Create an HBase connection to access HBase.
5. Select all	Create all four types of connection.

7. In the **Domain Information** section, enter the following information about Informatica domain.
   - The domain user name
   - The domain password

- The Data Integration Service name
- The Informatica home directory on Hadoop
- Hadoop Kerberos service principal name, if the Hadoop cluster uses Kerberos authentication
- Location of the Kerberos keytab on the Data Integration Service machine, if the Hadoop cluster uses Kerberos authentication

After you enter the details, the Hadoop Configuration Manager updates the properties and recycles the Data Integration Service.

8. In the **Connection Details** section, provide the connection properties.

   Based on the type of connection you choose to create, the Hadoop Configuration Manager requires different properties.

   The Hadoop Configuration Manager creates the connections.

9. The Hadoop Configuration Manager reports a summary of its operations, including whether connection creation succeeded and the location of log files.

The Hadoop Configuration Manager creates the following file in the `<Informatica installation directory>/tools/BDMUtil` directory:

ClusterConfig.properties.<timestamp>

Contains details about the properties fetched from the Hadoop cluster, including cluster node names, to provide templates for connection creation commands. To use these connection creation templates to create connections to the Hadoop cluster, edit the connection name, domain username and password in the generated commands.

Complete the Configuration through SSH

When you complete configuration through SSH, you must provide host names and Hadoop configuration file locations.

Configuration through SSH requires SSH connections from the Informatica domain to the machines that host the NameNode, YARN ResourceManager, and Hive client. Informatica recommends that you use an SSH connection without a password or have sshpass or Expect installed. If you do not use one of these methods, you must enter the password each time the utility downloads a file from the Hadoop cluster.

1. Enter the name node host name.
2. Enter the SSH user ID.
3. Enter the password for the SSH user ID, or press enter if you use an SSH connection without a password.
4. Enter the location for the `hdfs-site.xml` file on the Hadoop cluster.
5. Enter the location for the `core-site.xml` file on the Hadoop cluster.

   The Hadoop Configuration Manger connects to the name node and downloads the following files: `hdfs-site.xml` and `core-site.xml`.

6. Enter the Yarn resource manager host name.

   Note: Yarn resource manager was formerly known as JobTracker.

7. Enter the SSH user ID.
8. Enter the password for the SSH user ID, or press enter if you use an SSH connection without a password.
9. Enter the directory for the `mapred-site.xml` file on the Hadoop cluster.
10. Enter the directory for the `yarn-site.xml` file on the Hadoop cluster.

    The utility connects to the JobTracker and downloads the following files: `mapred-site.xml` and `yarn-site.xml`.

11. Enter the Hive client host name.
12. Enter the SSH user ID.
13. Enter the password for the SSH user ID, or press enter if you use an SSH connection without a password.
14. Enter the directory for the `hive-site.xml` file on the Hadoop cluster.

    The configuration manager connects to the Hive client and downloads the following file: `hive-site.xml`.
15. Optionally, You can choose to configure the HBase server.

    Select the following options:

Option	Description
1	No. Select this option if you do not want to configure the HBase server.
2	Yes. Select this option to configure the HBase server.

    If you select **Yes**, enter the following information to configure the HBase server:

    a. Enter the HBase server host name.

    b. Enter the SSH user ID.

    c. Enter the password for the SSH user ID, or press enter if you use an SSH connection without a password.

    d. Enter the directory for the `hbase-site.xml` file on the Hadoop cluster.
16. In the **Create Connections** section, select the connection type to create Big Data Management connections:

Option	Description
1. Hive	Create a Hive connection to access Hive as a source or target.
2. HDFS	Create an HDFS connection to read data from or write data to the HDFS file system on a Hadoop cluster.
3. Hadoop	Create a Hadoop connection to run mappings in the Hadoop environment.
4. HBase	Create an HBase connection to access HBase.
5. Select all	Create all four types of connection.

    Press the number that corresponds to your choice.
17. The **Domain Information** section displays the domain name and the node name. Enter the following additional information about Informatica domain.

    a. Enter the domain user name.

    b. Enter the domain password.

    c. Enter the Data Integration Service name.

    d. If the Hadoop cluster uses Kerberos authentication, enter the following information:

    - Hadoop Kerberos service principal name

- Hadoop Kerberos keytab location. Location of the keytab on the Data Integration Service machine.

Note: After you enter the Data Integration Service name, the utility tests the domain connection, and then recycles the Data Integration Service.

18. In the **Connection Details** section, provide the connection properties.

    Based on the type of connection you choose to create, the utility requires different properties.

    The Hadoop Configuration Manager creates the connections.

19. The Hadoop Configuration Manager reports a summary of its operations, including whether connection creation succeeded and the location of log files.

The Hadoop Configuration Manager creates the following file in the `<Informatica installation directory>/tools/BDMUtil` directory:

ClusterConfig.properties.<timestamp>

Contains details about the properties fetched from the Hadoop cluster, including cluster node names, to provide templates for connection creation commands. To use these connection creation templates to create connections to the Hadoop cluster, edit the connection name, domain username and password in the generated commands.

Run the Configuration Manager in Silent Mode

To configure the Big Data Management without user interaction in silent mode, run the Hadoop Configuration Manager in silent mode. Use the `SilentInput.properties` file to specify the configuration options. The configuration manager reads the file to determine the configuration options.

To configure the domain for Big Data Management in silent mode, perform the following tasks:

1. Configure the properties in the `SilentInput.properties` file.
2. Run the configuration manager with the `SilentInput.properties` file.

Configure the Properties File

The `SilentInput.properties` file includes the properties that are required by the configuration manger. Customize the properties file to specify the options for your configuration and then run the configuration manager in silent mode.

You might want to create connections if you are installing for the first time or if you are upgrading and the changes to the Hadoop environment are significant, such as a different distribution. If you are upgrading and you create connections that you want to use in existing mappings, you need to update mappings to use the new connections after you upgrade.

You might want to skip creating connections if the changes to the Hadoop environment are minimal, such as a change in distribution version. If you are upgrading and you choose not to create connections, you can continue to use the connections in existing mappings. However, you need update any changed properties after you upgrade. Property changes might include host names, URIs, or port numbers.

1. Find the sample `SilentInput.properties` file in the following location: `<Informatica installation directory>/tools/BDMUtil`
2. Create a backup copy of the `SilentInput.properties` file.

3. Use a text editor to open the file and modify the values of the configuration properties.
   The following table describes the configuration properties that you can modify:

Property Name	Description
CLOUDERA_SELECTION	Indicates whether to configure the Informatica domain for the Cloudera CDH cluster. To configure the domain for the Cloudera CDH cluster, set the value of this property to 1.
HORTONWORKS_SELECTION	Indicates whether to configure the Informatica domain for the Hortonworks cluster. To configure the domain for the Hortonworks cluster, set the value of this property to 1.
BIG_INSIGHT	Indicates whether to configure the Informatica domain for the BigInsights cluster. To configure the domain for the BigInsights cluster, set the value of this property to 1.
HD_INSIGHT	Indicates whether to configure the Informatica domain for the HDInsight cluster. To configure the domain for the HDInsight cluster, set the value of this property to 1.
DIST_FOLDER_NAME	Indicates the Hadoop distribution folder name present in INFA_HOME/services/shared/hadoop.
INSTALL_TYPE	Indicates how to access files on the Hadoop cluster. Set the value depending on the Hadoop distribution. To access files on the Cloudera CDH cluster using the Cloudera Manager, set the value of this property to 0. To access files on the Hortonworks HDP, IBM BigInsights, or the Azure HDInsight cluster using Apache Ambari, set the value of this property to 3.
CLOUDERA_HOSTNAME	Required if you set INSTALL_TYPE=0. Indicates the host name of the Cloudera Manager.
CLOUDERA_USER_NAME	Required if you set INSTALL_TYPE=0. Indicates the user id for the Cloudera Manager.
CLOUDERA_USER_PASSWD	Required if you set INSTALL_TYPE=0. Indicates the password for the Cloudera Manager.
CLOUDERA_PORT	Required if you set INSTALL_TYPE=0. Indicates the port of the Cloudera Manager.
CLOUDERA_SINGLECLUSTER_NAME	Required if you set INSTALL_TYPE=0. Indicates the name of the Cloudera CDH cluster for which you want to configure Big Data Management. If the Cloudera Manager manages a single cluster, you do not have to set this property. Retain this property as a comment or remove the property from the file. If the Cloudera Manager manages more than one cluster, uncomment and set the property to the name of the cluster for which you want to configure Big Data Management. Note: In a multi-cluster environment, the configuration fails if the property is commented or if the property has an invalid cluster name as the value.

Property Name	Description
AMBARI_HOSTNAME	Required if you set INSTALL_TYPE =3. Indicates the password for Apache Ambari.
AMBARI_USER_NAME	Required if you set INSTALL_TYPE=3. Indicates the port number for Apache Ambari.
AMBARI_USER_PASSWD	Required if you set INSTALL_TYPE=3. Indicates the password for Apache Ambari.
AMBARI_PORT	Required if you set INSTALL_TYPE=3. Indicates the port number for Apache Ambari.
TEZ_EXECUTION_MODE	Required if you set INSTALL_TYPE=3. Indicates whether to enable Tez for the Hive engine. To enable Tez on the Hortonworks HDP or Azure HDInsight cluster, set the value of this property to 1. Otherwise, set the value to 0.
CREATE_CONNECTION	Indicates whether to update the Data Integration Service and create connections for Big Data Management. To update the Data Integration Service and create connections, set the value of this property to 1 and define the connection properties. If you are upgrading and you choose not to create connections, you can continue to use the connections in existing mappings. However, you need update any changed properties after you upgrade. Property changes might include host names, URIs, or port numbers. If you are upgrading and you create connections that you want to use in existing mappings, you need to update mappings to use the new connections after you upgrade. To exit the configuration manager, set the value to 0.
UPDATE_DIS	Required if you set CREATE_CONNECTION=1. Indicates whether to update the Data Integration Service properties. To update the Data Integration Service, set this property to 1. Otherwise, set the value to 0.
CREATE_BDM_CONNECTIONS	Required if you set CREATE_CONNECTION=1. Set this property to 1 to create Hadoop, HDFS, HBase, and Hive connections. Otherwise, the configuration manager will not create connections.
DOMAIN_USER	Required if you set CREATE_CONNECTION=1. Indicates the user name for the domain administrator.
DOMAIN_PSSWD	Required if you set CREATE_CONNECTION=1. Indicates the password for the domain administrator.
DIS_SERVICE_NAME	Required if you set CREATE_CONNECTION=1 and UPDATE_DIS=1. Indicates the name of the Data Integration Service.

Property Name	Description
CLUSTER_INSTALLATION_DIR	Required if you set CREATE_CONNECTION=1 and UPDATE_DIS=1. Indicates the Informatica home directory on every node on the Hadoop cluster that is created during the Big Data Management installation. Default is /opt/Informatica.
KERBEROS_PRINCIPAL_NAME	Required if you set CREATE_CONNECTION=1 and UPDATE_DIS=1 and if the Hadoop cluster uses Kerberos authentication. Indicates the Hadoop Kerberos service principal name.
KERBEROS_KEYTAB	Required if you set CREATE_CONNECTION=1 and UPDATE_DIS=1 and if the Hadoop cluster uses Kerberos authentication. Indicates the Hadoop Kerberos keytab location.
HIVE_CONNECTION	Required if you set CREATE_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Set this property to 1 to create a Hive connection. Otherwise, set the value to 0.
HDFS_CONNECTION	Required if you set CREATE_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Set this property to 1 to create a HDFS connection. Otherwise, set the value to 0.
HADOOP_CONNECTION	Required if you set CREATE_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Set this property to 1 to create a Hadoop connection. You must create a Hadoop connection to run the mappings in the Hadoop environment.
HBASE_CONNECTION	Required if you set CREATE_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Set this property to 1 to create a HBASE connection. Otherwise, set the value to 0.
SELECT_ALL_CONNECTION	Required if you set CREATE_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Set this property to 1 to create all listed connections. Also set HIVE_CONNECTION=1, HDFS_CONNECTION=1, HADOOP_CONNECTION=1, and HBASE_CONNECTION=1 .
SAMPLE_HIVE_CONNECTION	Required if CREATE_CONNECTION=1, HIVE_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Indicates connection name for Hive.
HIVE_EXECUTION_MODE	Required if you set CREATE_CONNECTION=1, HIVE_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Indicates the execution mode for Hive. Set this property to either Local or Remote.
HIVE_USER_NAME	Required if you set CREATE_CONNECTION=1, HIVE_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Indicates the user name for Hive. You must set the execution mode to Remote.

Property Name	Description
SAMPLE_HDFS_CONNECTION	Required if you set CREATE_CONNECTION=1, HDFS_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Indicates the connection name for HDFS.
HDFS_USER_NAME	Required if you set CREATE_CONNECTION=1, HDFS_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Indicates the user name to access HDFS.
SAMPLE_HADOOP_CONNECTION	Required if you set CREATE_CONNECTION=1 , HADOOP_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Indicates the connection name for Hadoop.
BLAZE_WORKING_DIR	Required if you set CREATE_CONNECTION=1 , HADOOP_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Indicates the existing Blaze work directory on HDFS.
BLAZE_USER	Required if you set CREATE_CONNECTION=1, HADOOP_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Indicates the user name for Blaze.
SPARK_HDFS_STAGING_DIR	Required if you set CREATE_CONNECTION=1 , HADOOP_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Indicates the existing Spark HDFS staging directory.
SPARK_EVENTLOG_DIR	Required if you set CREATE_CONNECTION=1, HADOOP_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Optional: Indicates the Spark event log directory
SPARK_PARAMETER_LIST	Required if you set CREATE_CONNECTION=1, HADOOP_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Indicates the Spark execution parameters list.
IMPERSONATION_USER	Required if you set CREATE_CONNECTION=1, HADOOP_CONNECTION=1, CREATE_BDM_CONNECTIONS=1, and if the Hadoop cluster uses Kerberos authentication.
SAMPLE_HBASE_CONNECTION	Required if you set CREATE_CONNECTION=1 , HBASE_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Indicates the connection name for HBase.
ZOOKEEPER_HOSTS	Required if you set CREATE_CONNECTION=1 , HBASE_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Indicates the ZooKeeper host.

Run the Configuration Manager

After you configure the properties file, open a command prompt to start the silent configuration.

1. Go to the following directory: `<Informatica installation directory>/tools/BDMUtil`
2. Verify that the directory contains the `SilentInput.properties` file that you edited.
3. Run the `BDMSilentConfig.sh` file.

The installer runs in the background. The process can take a while. The silent configuration is complete when the following log files are created in the `<Informatica installation directory>/tools/BDMUtil` directory:

- Informatica_Big_Data_Edition_Configuration_Utility.log.<timestamp>
- ClusterConfig.properties.<timestamp>

The silent configuration fails if you incorrectly configure the properties file or if the installation directory is not accessible. The `silenterror.log` file is created in the home directory of the UNIX user that runs the configuration manager in silent mode. Correct the errors and then run the silent configuration again.

CHAPTER 12

Manual Configuration Tasks for Cloudera CDH

This chapter includes the following topics:

- Update Files on the Hadoop Environment for Cloudera CDH, 91
- Update Files on the Domain Environment for Cloudera CDH, 94
- Creating and Configuring Security Certificates and Truststore Files, 98
- Update Files on the Developer Tool, 99
- Download the JDBC Drivers for Sqoop Connectivity, 100
- Complete Upgrade Tasks, 101

Update Files on the Hadoop Environment for Cloudera CDH

You can configure additional functionality through the configuration files the in Hadoop environment.

When you use Cloudera Manager to update properties, updates are propagated to each node in the cluster. After you make updates, you can restart the Hadoop cluster and services through Cloudera Manager.

Update core-site.xml

Update the core-site.xml file on the Hadoop environment to enable access to Hive tables on the cluster.

Configure Access to Hive Tables in Amazon S3 Buckets

Configure the AWS access key to run mappings with sources and targets on Hive tables in Amazon S3 buckets.

Note: To use a Hive table as a target on Amazon S3, grant write permission to the bucket through bucket policies, or add these properties to the configuration file. You must add these properties to the core-site.xml file on the Hadoop environment and to the yarn-site.xml file on the domain environment.

Configure the following properties:

fs.s3a.access.key

The ID for the Blaze and Spark engines to connect to the Amazon S3a file system. For example,

```
<property>
    <name>fs.s3a.access.key</name>
    <value>[Your Access Key]</value>
 </property>
```

fs.s3a.secret.key

The password for the Blaze and Spark engines to connect to the Amazon S3a file system.

```
<property>
    <name>fs.s3a.secret.key</name>
    <value>[Your Access Id]</value>
</property>
```

When you use a server side encryption protocol for Hive buckets, configure properties to enable access to encrypted Hive buckets.

Configure the following properties:

fs.s3.enableServerSideEncryption

Set to TRUE to enable server side encryption. For example,

```
<property>
    <name>fs.s3.enableServerSideEncryption</name>
    <value>TRUE</value>
 </property>
```

fs.s3a.server-side-encryption-algorithm

The encryption algorithm that you use to encrypt Hive buckets. For example,

```
<property>
    <name>fs.s3a.server-side-encryption-algorithm</name>
    <value>AES256</value>
 </property>
```

Update mapred-site.xml

Update the mapred-site.xml file on the Hadoop environment to configure functionality such as Sqoop connectivity.

Configure Sqoop for Clusters

To enable Sqoop connectivity for Cloudera clusters, configure MapReduce properties in the yarn configuration in Cloudera Manager. Update the **NodeManager Advanced Configuration Snippet (Safety Valve) for mapred-site.xml** property, redeploy the client configurations, and restart Hadoop services and the cluster.

Configure the following properties:

mapreduce.application.classpath

Classpaths for MapReduce applications.

```
<property>
    <name>mapreduce.application.classpath</name>
    <value>$HADOOP_MAPRED_HOME/*,$HADOOP_MAPRED_HOME/lib/*,$MR2_CLASSPATH,
$CDH_MR2_HOME</value>
    <description>Classpaths for MapReduce applications</description>
</property>
```

mapreduce.jobhistory.intermediate-done-dir

Directory where the map-reduce jobs write history files.

```
<property>
    <name>mapreduce.jobhistory.intermediate-done-dir</name>
    <value><Directory where the map-reduce jobs write history files></value>
    <description>Directory where the map-reduce jobs write history files</
description>
</property>
```

mapreduce.jobhistory.address

Location of the MapReduce JobHistory Server. The default port is 10020.

```
<property>
    <name>mapreduce.jobhistory.address</name>
    <value><host name>:port</value>
    <description>MapReduce JobHistory Server IPC host:port</description>
</property>
```

Update yarn-site.xml

Update the yarn-site.xml file on the Hadoop environment to configure functionality such as dynamic resource allocation and virtual memory limits.

Allocate Cluster Resources for the Blaze Engine

Update the following properties to verify that the cluster allocates sufficient memory and resources for the Blaze engine:

yarn.nodemanager.resource.memory-mb

The maximum RAM available for each container. Set the maximum memory on the cluster to increase resource memory available to the Blaze engine. Configure this property on the following nodes:

- Run-time nodes. Set to at least 10 GB.
- Management node. Set to at least 13 GB.

yarn.nodemanager.resource.cpu-vcores

The number of virtual cores for each container. The number might correspond to the number of physical cores, but you can increase the value to allow for more processing. Configure this property on the following nodes:

- Run-time nodes. Set to at 6.
- Management node. Set to 9.

yarn.scheduler.minimum-allocation-mb

The minimum RAM available for each container. Set the minimum memory to allow the VM to spawn sufficient containers. Configure this property to be no less than 4 GB of the maximum memory on the run-time and management nodes.

Configure Virtual Memory Limits for the Blaze Engine

Configure the following property to remove virtual memory limits for every node in the Hadoop cluster:

yarn.nodemanager.vmem-check-enabled

Determines virtual memory limits. Set the value to false to disable virtual memory limits. For example,

```
<property>
    <name>yarn.nodemanager.vmem-check-enabled</name>
    <value>false</value>
    <description>Enforces virtual memory limits for containers.</description>
</property>
```

Configure Dynamic Resource Allocation for the Spark Engine

Configure dynamic resource allocation for mappings to run on the Spark engine. Update the following properties on each node where YARN node manager is running:

yarn.nodemanager.aux-services

The auxiliary service name. Add spark_shuffle to the current entry. For example, if the current value is "mapreduce_shuffle," update it to be "mapreduce_shuffle,spark_shuffle."

```
<property>
    <name>yarn.nodemanager.aux-services</name>
    <value>mapreduce_shuffle,spark_shuffle</value>
</property>
```

yarn.nodemanager.aux-services.spark_shuffle.class

The auxiliary service class to use. Set to org.apache.spark.network.yarn.YarnShuffleService. For example,

```
<property>
    <name>yarn.nodemanager.aux-services.spark_shuffle.class</name>
    <value>org.apache.spark.network.yarn.YarnShuffleService</value>
</property>
```

yarn.application.classpath

The classpath for YARN applications. Verify that the yarn classpath includes the path to the Spark shuffle.jar file packaged with the distribution. If the .jar file is not in any Hadoop distribution directory, update the path to use the .jar file in the Apache Spark distribution directory. For example,

```
<property>
    <name>yarn.application.classpath</name>
    <value>/opt/Informatica/services/shared/spark/lib_spark_2.0.1_hadoop_2.6.0/spark-
networkshuffle_2.11-2.0.1.jar</value>
</property>
```

Update Files on the Domain Environment for Cloudera CDH

You can configure additional functionality through the configuration files on the domain.

When the Informatica installer creates nodes on the domain, it creates directories and files for Big Data Management. When you configure Big Data Management on the domain, you need to edit configuration files to enable functionality.

The following table describes the installation directories and the default paths:

Directory	Description
Data Integration Service Hadoop distribution directory	The Hadoop distribution directory on the Data Integration Service node. It corresponds to the distribution on the Hadoop environment. You can view the directory path in the Data Integration Service properties of the Administrator tool. Default is `<Informatica installation directory>/Informatica/services/shared/hadoop/<Hadoop distribution name>_<version>`.
Configuration file directories	Configuration files for Big Data Management are stored in the following directories: - `<Informatica installation directory>/Informatica/services/shared/hadoop/<Hadoop distribution name>_<version>/InfaConf`. Contains the hadoopEnv.properties file. - `<Informatica installation directory>/Informatica/services/shared/hadoop/<Hadoop distribution name>_<version>/conf`. Contains *-site files.

Update hadoopEnv.properties

Update the hadoopEnv.properties file to configure functionality such as Sqoop connectivity and the Spark run-time engine.

Open hadoopEnv.properties and back it up before you configure it. You can find the hadoopEnv.properties file in the following location: `<Informatica installation directory>/services/shared/hadoop/<Hadoop distribution name>_<version number>/infaConf`

Configure Sqoop Connectivity

Configure the following property for Sqoop connectivity:

infapdo.env.entry.hadoop_node_jdk_home

Configure the HADOOP_NODE_JDK_HOME to represent the JDK version that the cluster nodes use. You must use JDK version 1.7 or later.

Configure the property as follows:

```
infapdo.env.entry.hadoop_node_jdk_home=HADOOP_NODE_JDK_HOME=<cluster JDK home>/jdk<version>
```

For example,

```
infapdo.env.entry.hadoop_node_jdk_home=HADOOP_NODE_JDK_HOME=/usr/java/default
```

Configure Performance for the Spark Engine

Configure the following performance properties for the Spark engine:

spark.dynamicAllocation.enabled

Performance configuration to run mappings on the Spark engine. Enables dynamic resource allocation. Required when you enable the external shuffle service.

Set the value to TRUE.

spark.shuffle.service.enabled

Performance configuration to run mappings on the Spark engine. Enables the external shuffle service. Required when you enable dynamic resource allocation.

Set the value to TRUE.

spark.scheduler.maxRegisteredResourcesWaitingTime

Performance configuration to run mappings on the Spark engine. The number of milliseconds to wait for resources to register before scheduling a task. Reduce this from the default value of 30000 to reduce delays before starting the Spark job execution.

Set the value to 15000.

spark.scheduler.minRegisteredResourcesRatio

Performance configuration to run mappings on the Spark engine. The minimum ratio of registered resources to acquire before task scheduling begins. Reduce this from the default value of 0.8 to reduce any delay before starting the Spark job execution.

Set the value to .5.

spark.executor.instances

Performance configuration to run mappings on the Spark engine. If you enable dynamic resource allocation for the Spark engine, Informatica recommends that you convert this property to a comment. For example, `#spark.executor.instances=100`

Configure Environment Variables

You can optionally add third-party environment variables and extend the existing PATH environment variable in the hadoopEnv.properties file. The following text shows sample entries to configure environment variables:

```
infapdo.env.entry.oracle_home=ORACLE_HOME=/databases/oracle
infapdo.env.entry.db2_home=DB2_HOME=/databases/db2
infapdo.env.entry.db2instance=DB2INSTANCE=OCA_DB2INSTANCE
infapdo.env.entry.db2codepage=DB2CODEPAGE="1208"
infapdo.env.entry.odbchome=ODBCHOME=$HADOOP_NODE_INFA_HOME/ODBC7.1
infapdo.env.entry.home=HOME=/opt/thirdparty
infapdo.env.entry.gphome_loaders=GPHOME_LOADERS=/databases/greenplum
infapdo.env.entry.pythonpath=PYTHONPATH=$GPHOME_LOADERS/bin/ext
infapdo.env.entry.nz_home=NZ_HOME=/databases/netezza
infapdo.env.entry.ld_library_path=LD_LIBRARY_PATH=$HADOOP_NODE_INFA_HOME/services/
shared/bin:$HADOOP_NODE_INFA_HOME/DataTransformation/bin:$HADOOP_NODE_HADOOP_DIST/lib/
native:$HADOOP_NODE_INFA_HOME/ODBC7.1/lib:$HADOOP_NODE_INFA_HOME/jre/lib/
amd64:$HADOOP_NODE_INFA_HOME/jre/lib/amd64/server:$HADOOP_NODE_INFA_HOME/java/jre/lib/
amd64:$HADOOP_NODE_INFA_HOME/java/jre/lib/amd64/server:/databases/oracle/lib:/
databases/db2/lib64:$LD_LIBRARY_PATH
infapdo.env.entry.path=PATH=$HADOOP_NODE_HADOOP_DIST/scripts:$HADOOP_NODE_INFA_HOME/
services/shared/bin:$HADOOP_NODE_INFA_HOME/jre/bin:$HADOOP_NODE_INFA_HOME/java/jre/bin:
$HADOOP_NODE_INFA_HOME/ODBC7.1/bin:/databases/oracle/bin:/databases/db2/bin:$PATH
#teradata
infapdo.env.entry.twb_root=TWB_ROOT=/databases/teradata/tbuild
infapdo.env.entry.manpath=MANPATH=/databases/teradata/odbc_64:/databases/teradata/odbc_64
infapdo.env.entry.nlspath=NLSPATH=/databases/teradata/odbc_64/msg/%N:/databases/
teradata/msg/%N
infapdo.env.entry.pwd=PWD=/databases/teradata/odbc_64/samples/C
```

Update hive-site.xml

Update the hive-site.xml file on the domain environment to set properties for dynamic partitioning.

Configure Dynamic Partitioning

To use Hive dynamic partitioned tables, configure the following properties:

hive.exec.dynamic.partition

Enables dynamic partitioned tables. Set this value to `TRUE`.

exec.dynamic.partition.mode

Allows all partitions to be dynamic. Set this value to `nonstrict`.

Update yarn-site.xml

Update the yarn-site.xml file on the domain to enable access to Hive tables on the cluster.

Configure Access to Hive Tables in Amazon S3 Buckets

Configure the AWS access key to run mappings with sources and targets on Hive tables in Amazon S3 buckets.

Note: To use a Hive table as a target on Amazon S3, grant write permission to the bucket through bucket policies, or add these properties to the configuration file. You must add these properties to the core-site.xml file on the Hadoop environment and to the yarn-site.xml file on the domain environment.

Configure the following properties:

fs.s3a.access.key

The ID for the Blaze and Spark engines to connect to the Amazon S3a file system. For example,

```
<property>
    <name>fs.s3a.access.key</name>
    <value>[Your Access Key]</value>
 </property>
```

fs.s3a.secret.key

The password for the Blaze and Spark engines to connect to the Amazon S3a file system.

```
<property>
    <name>fs.s3a.secret.key</name>
    <value>[Your Access Id]</value>
</property>
```

When you use a server side encryption protocol for Hive buckets, configure properties to enable access to encrypted Hive buckets.

Configure the following properties:

fs.s3.enableServerSideEncryption

Set to TRUE to enable server side encryption. For example,

```
<property>
    <name>fs.s3.enableServerSideEncryption</name>
    <value>TRUE</value>
 </property>
```

fs.s3a.server-side-encryption-algorithm

The encryption algorithm that you use to encrypt Hive buckets. For example,

```
<property>
    <name>fs.s3a.server-side-encryption-algorithm</name>
    <value>AES256</value>
 </property>
```

Creating and Configuring Security Certificates and Truststore Files

When you use custom, special, or self-signed security certificates to secure the Hadoop cluster, Informatica clients that connect to the cluster require these certificates to be present in the client machine truststore.

Import Security Certificates

To connect to the Hadoop cluster to develop a mapping, the Developer tool requires security certificate aliases on the machine that hosts the Developer tool. To run a mapping, the machine that hosts the Data Integration Service requires these same certificate alias files.

Perform the following steps from the Developer tool host machine and from the Data Integration Service host machine:

1. Run the following command to export the certificates from the cluster:

   ```
   keytool -export -alias <alias name> -keystore <custom.truststore file location> -
   file <exported certificate file location> -storepass <password>
   ```

   For example:

   ```
   <java home>/jre/bin/keytool -export -alias <alias name> -keystore ~/
   custom.truststore -file ~/exported.cer
   ```

   The command produces a certificate file.

2. Choose to import security certificates to an SSL-enabled domain or a domain that is not SSL-enabled.
   - If the domain is SSL-enabled, then import the certificate alias file to the following locations:
     - The following path on the Developer tool machine: `<Developer Tool home>\clients\shared\security\infa_truststore.jks`
     - The following path on the machine that hosts the Data Integration Service: `<Informatica installation home>/services/shared/security/infa_truststore.jks`
   - If the Informatica domain is not SSL-enabled, import the security certificate alias file to the following locations:
     - The following path on the Developer tool machine: `<Developer Tool home>\clients\java\jre\lib\security\cacerts`
     - The following path on the machine that hosts the Data Integration Service: `<Informatica installation home>/java/jre/lib/security/cacerts`

Configure the Data Integration Service to Use Truststore File Paths

To enable the Data Integration Service to access truststore files on the Hadoop cluster, perform the following steps to configure Data Integration Service properties with truststore file paths:

1. Get the location of truststore files from the cluster manager administration web page.
   If you do not have access to the cluster manager, ask the cluster administrator for the path to the truststore files for the resources you want to access.
2. To enable access to Hive sources in Native mode, copy the truststore files to the corresponding location in the Hadoop distribution directory of the Informatica domain machine.
   For example, if the truststore file is located at `/etc/security/serverKeys/all.jks` on the cluster, copy the file to the same location in the Hadoop distribution directory on the domain machine: `/etc/security/serverKeys/all.jks`. Create the directory if it does not exist.
3. In the Administrator tool, select the Data Integration Service in the **Domain Navigator**. Click the **Properties** tab to display Data Integration Service properties.

4. Click the **Edit** icon for Custom Properties.
   The **Edit Custom Properties** dialog box appears.
5. Enter the following name for the custom property: JVMOption.
   If a JVMOption custom property exists, then increment the name with an integer like JVMOption1.
6. In the **Value** pane for the property, type the following value:

   ```
   -Djavax.net.ssl.trustStore=<path to the truststore file on the cluster>
   ```

   For example:

   ```
   -Djavax.net.ssl.trustStore=/etc/security/serverKeys/all.jks
   ```
7. Click **OK**.
8. To access additional resources that use truststore files in different locations, repeat steps 4-7. Increment the custom property name with an integer.
   The following image shows custom properties that allow the Data Integration Service to access truststore files in two different locations:

   Note: The Data Integration service converts the name of custom properties to add the "ExecutionContextOptions." prefix when you recycle the service.
9. Recycle the Data Integration Service.
10. If you have ever run a mapping on the Blaze engine, you must stop the Grid Manager application on the Data Integration Service host machine.
    a. Run the following command to list existing YARN applications:

       ```
       yarn application -list
       ```
    b. In the list of YARN applications, identify the application ID for the Blaze Grid Manager.
    c. Run the following command to stop the Grid Manager application:

       ```
       yarn application -kill <application ID>
       ```

Now you can run the mapping from the Developer tool. The Data Integration System imports the truststore certificates at run time.

Update Files on the Developer Tool

Update configuration files to enable communication and high availability between the Developer tool and the cluster.

Before you perform configuration on the Developer tool, you need to copy some configuration files to the machine that hosts the Developer tool.

1. Find hdfs-site.xml and yarn-site.xml on the name node of the cluster.
2. Copy the files to the following location on the machine that hosts the Developer tool: `<Informatica installation directory>/clients/DeveloperClient/Hadoop/<Hadoop distribution><version>/conf`

yarn-site.xml

Configure the following property in yarn-site.xml:

yarn.application.classpath

Sets the classpath for YARN applications and enables the Developer tool to read from and write to a highly available cluster.

To identify the classpath, run the `yarn classpath` command. The following sample text shows a sample classpath entry:

```
<property>
<name>yarn.application.classpath</name>
<value>
/etc/hadoop/conf:/opt/cloudera/parcels/CDH-5.4.0-1.cdh5.4.0.p0.27/lib/hadoop/
libexec/../../hadoop/lib/*:/opt/cloudera/parcels/CDH-5.4.0-1.cdh5.4.0.p0.27/lib/
hadoop/libexec/../../hadoop/.//*:/opt/cloudera/parcels/
CDH-5.4.0-1.cdh5.4.0.p0.27/lib/hadoop/libexec/../../hadoop-hdfs/./:/opt/cloudera/
parcels/CDH-5.4.0-1.cdh5.4.0.p0.27/lib/hadoop/libexec/../../hadoop-hdfs/lib/*:/opt/
cloudera/parcels/CDH-5.4.0-1.cdh5.4.0.p0.27/lib/hadoop/libexec/../../hadoophdfs/.//
*:/
opt/cloudera/parcels/CDH-5.4.0-1.cdh5.4.0.p0.27/lib/hadoop/libexec/../../
hadoop-yarn/lib/*:/opt/cloudera/parcels/CDH-5.4.0-1.cdh5.4.0.p0.27/lib/hadoop/
libexec/../../hadoop-yarn/.//*:/opt/cloudera/parcels/CDH/lib/hadoopmapreduce/
lib/*:/opt/cloudera/parcels/CDH/lib/hadoop-mapreduce/.//*:/opt/cloudera/
parcels/CDH-5.4.0-1.cdh5.4.0.p0.27/lib/hadoop/libexec/../../hadoop-yarn/.//*:/opt/
cloudera/parcels/CDH-5.4.0-1.cdh5.4.0.p0.27/lib/hadoop/libexec/../../hadoopyarn/
lib/*
</value>
</property>
```

developerCore.ini

Edit developerCore.ini to enable communication between the Developer tool and the Hadoop cluster. You can find developerCore.ini in the following directory: `<Informatica installation directory>\clients\DeveloperClient`

Add the following property:

```
-DINFA_HADOOP_DIST_DIR=hadoop\<distribution><version>
```

The change takes effect when you restart the Developer tool.

Download the JDBC Drivers for Sqoop Connectivity

To configure Sqoop connectivity for relational databases, you must download the relevant JDBC driver jar files and copy the jar files to the node where the Data Integration Service runs.

You can use any Type 4 JDBC driver that the database vendor recommends for Sqoop connectivity. Download additional files if you use Cloudera Connector Powered by Teradata.

1. Download the JDBC driver jar files for the database that you want to connect to.
2. If you use Cloudera Connector Powered by Teradata, download the following files:
   - The package named `sqoop-connector-teradata-<version>.tar.gz`. You can download this package from the following URL: http://www.cloudera.com/downloads.html
   - The files named `terajdbc4.jar` and `tdgssconfig.jar` . You can download the files from the following URL: http://downloads.teradata.com/download/connectivity/jdbc-driver
3. Copy the jar files to the following directory on the machine where the Data Integration Service runs:

   `<Informatica installation directory>\externaljdbcjars`

At run time, the Data Integration Service copies the jar files to the Hadoop distribution cache so that the jar files are accessible to all nodes in the cluster.

Note: The JDBC drivers that Informatica ships are not licensed for Sqoop connectivity.

Complete Upgrade Tasks

If you upgraded the Informatica platform or applied the hotfix, you need to perform some additional tasks within the Informatica domain.

Based on the version that you upgraded from, perform the following tasks:

Update the hadoopEnv.properties file.

Applies to all upgrades. The 10.1.1 HotFix 1 hadoopEnv.properties file contains additional properties. You need to manually update it to include customized configuration from previous versions.

Complete connection upgrades.

Applies to upgrades from 9.6.1 or any 9.6.1 hotfix. You need to generate Hadoop connections from the Hive connection and update parameter files.

Update connection properties.

Applies to any upgrade. If you did not create connections when you ran the Hadoop Configuration Manager, you must update values for connection properties that changed.

Replace connections.

Applies to any upgrade. If you created connections when you ran the Hadoop Configuration Manager, you need to replace connections in mappings with the new connections.

Update the hadoopEnv.properties File

When you run the Informatica upgrade or apply the hotfix on a machine that hosts the Data Integration Service, the installer creates a new hadoopEnv.properties file and backs up the existing configuration file.

You need to edit the version 10.1.1 HotFix 1 hadoopEnv.properties file to include any manual configuration that you performed in the previous version for the corresponding distribution.

If you applied the hotfix, you can find the backup hadoopEnv.properties file in the following location:

```
<Informatica installation directory>/BACK_FROM_HF1/services/shared/hadoop/<Hadoop
distribution name>_<version>/infaConf
```

If you upgraded from a previous version, you can find the backup hadoopEnv.properties file in the following location:

```
<Previous Informatica installation directory>/services/shared/hadoop/<Hadoop distribution
name>_<version>/infaConf
```

You can find the 10.1.1 HotFix 1 hadoopEnv.properties file in the following location:

```
<Informatica installation directory>/services/shared/hadoop/<Hadoop distribution
name>_<version>/infaConf
```

Complete Connection Upgrade

Effective in version 10.0, Big Data Management requires a Hadoop connection to run mappings on the Hadoop cluster. If you upgraded from 9.6.1 or any 9.6.1 hotfix, you must generate Hadoop connections from Hive connections that are enabled to run mappings in the Hadoop environment.

The upgrade process generates a connection name for the Hadoop connection and replaces the connection name in the mappings. It does not create the physical connection object. When the upgrade is complete, you must run a command to generate the connection. Generate Hadoop connections from Hive connections if the following conditions are true:

- You upgraded from 9.6.1 or any 9.6.1 hotfix.
- The Hive connections are configured to run mappings in the Hadoop environment.

Complete the following tasks to upgrade connections:

1. Run `infacmd isp generateHadoopConnectionFromHiveConnection` to generate a Hadoop connection from a Hive connection that is configured to run in the Hadoop environment.
   The command renames the connection as follows: "Autogen_<Hive connection name>." If the renamed connection exceeds the 128 character limit, the command fails.
2. If the command fails, complete the following tasks:
   a. Rename the connection to meet the character limit and run the command again.
   b. Run `infacmd dis replaceMappingHadoopRuntimeConnections` to replace connections associated with mappings that are deployed in applications.
   c. Run `infacmd mrs replaceMappingHadoopRuntimeConnections` to replace connections associated with mappings that you run from the Developer tool.
3. If the Hive connection was parameterized, you must update the connection names in the parameter file. Verify that the Hive sources, Hive targets, and the Hive engine parameters are updated with the correct connection name.
4. If any properties changed in the cluster, such as host names, URIs, or port numbers, you must update the properties in the connections.

For information about the infacmd commands, see the *Informatica Command Reference*.

Update Connection Properties

If you upgraded from version 10.0 or later, and you did not create connections when you ran the Hadoop Configuration Manager, you need to verify connection properties.

Review the Hadoop, Hive, HDFS, and HBase connections that you created in a previous release to update the values for connection properties. For example, if you added nodes to the cluster or if you updated the distribution version, you might need to verify host names, URIs, or port numbers for some of the properties.

Replace the Connections

If you created connections when you ran the Hadoop Configuration Manager, you need to replace connections in mappings with the new connections.

The method that you use to replace connections in mappings depends on the type of connection.

Hadoop connection

If you created a Hadoop connection when you ran the Hadoop Configuration Manager, run the following commands:

- `infacmd dis replaceMappingHadoopRuntimeConnections`. Replaces connections associated with mappings that are deployed in applications.
- `infacmd mrs replaceMappingHadoopRuntimeConnections`. Replaces connections associated with mappings that you run from the Developer tool.

For information about the infacmd commands, see the *Informatica Command Reference*.

Hive, HDFS, or HBase connection

If you created a Hive, HDFS, or HBase connection when you ran the Hadoop Configuration Manager, you must replace the connections manually.

CHAPTER 13

Uninstall for Cloudera CDH

This chapter includes the following topic:

- Uninstall Big Data Management, 104

Uninstall Big Data Management

Uninstall Big Data Management on Cloudera from the Cloudera Manager.

1. In Cloudera Manager, browse to **Hosts** > **Parcels** > **Informatica**.
2. Select **Deactivate**.

   Cloudera Manager stops the Informatica Big Data Management instance.
3. Select **Remove**.

   The cluster uninstalls Informatica Big Data Management.

Part V: Install and Configure for Hortonworks

This part contains the following chapters:

- Installation for Hortonworks HDP, 106
- Hadoop Configuration Manager for Hortonworks HDP, 111
- Manual Configuration Tasks for Hortonworks HDP, 121
- Uninstall for Hortonworks HDP, 133

CHAPTER 14

Installation for Hortonworks HDP

This chapter includes the following topics:

- Installation for Hortonworks HDP Overview, 106
- Download the Distribution Package, 106
- Install in an Ambari Stack, 107
- Troubleshooting Installation in an Ambari Stack, 110

Installation for Hortonworks HDP Overview

You can install Big Data Management to an Ambari stack on Hortonworks HDP.

If you are installing Big Data Management for the first time or are upgrading, you need to install the Big Data Management binaries on the Hadoop environment.

An Ambari stack is a cluster, containing a set of services from Apache and other vendors, that you administer and monitor using the Ambari management console. When you install Big Data Management to an Ambari stack, the stack integrates it with the other services in the stack and with client nodes that are registered to the stack.

Download the Distribution Package

The tar.gz file that you download includes the binaries required to run the installation.

1. Download the Big Data Management archive file `InformaticaAmbariService-<version>.<platform>-x64.tar.gz` to a location on the head node machine, and uncompress the archive.

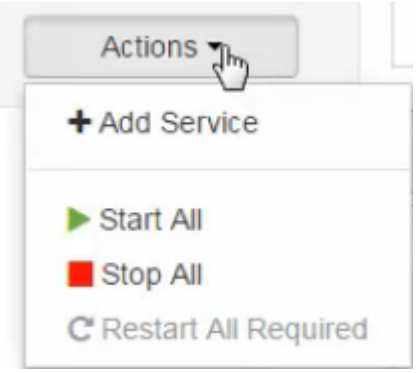
   The archive file contains a folder named INFORMATICABDM. This folder contains Big Data Management binaries. It also contains configuration files that Ambari uses to deploy the software.
2. Copy the INFORMATICABDM folder to the following location on the name node: `/var/lib/ambari-server/resources/stacks/<Hadoop_distribution>/<version>/services/`
3. Restart the Ambari server.

   When Ambari restarts, it registers the presence of the Big Data Management package and makes it available to install as a service in the stack.

Install in an Ambari Stack

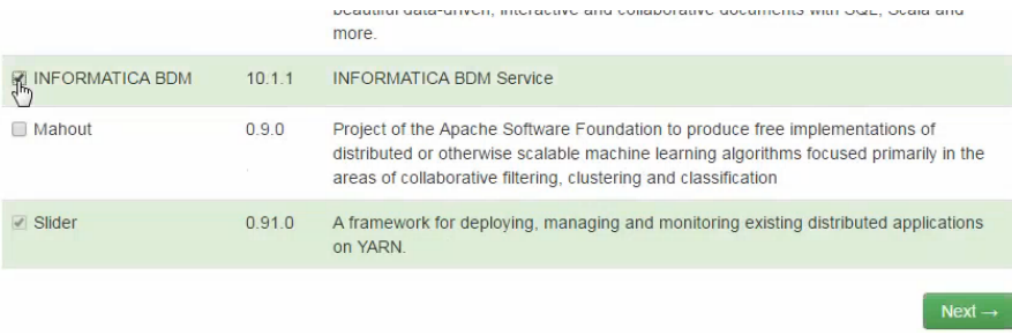
1. In the Ambari management console, click the **Admin** tab and select **Stacks and Versions**.
2. Click the **Actions** button and select **Add Service**.

   The following image shows the **Add Service** menu item:

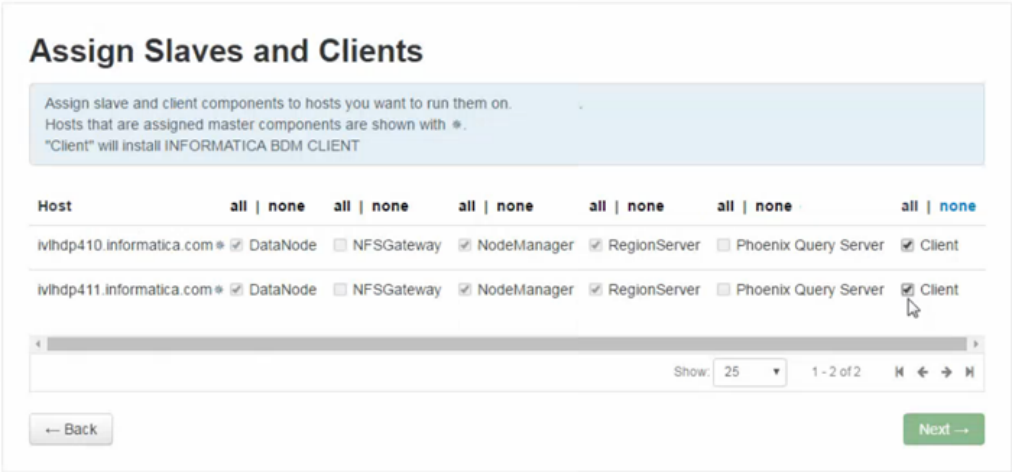
   The **Add Service Wizard** opens.
3. In the list of available services, select **INFORMATICABDM** and click **Next**.

   The following image shows the Big Data Management service in the list of available services:

   The **Assign Slaves and Clients** screen opens.
4. In the **Assign Slaves and Clients** screen, select **Client** for each node where you want to install Big Data Management.

   The following image shows how to designate nodes as clients:

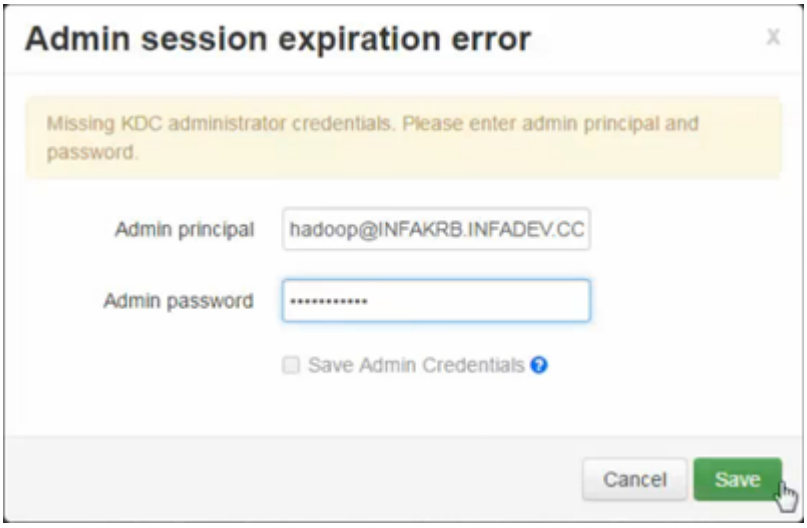


5. In the **Customize Services** screen, click **Next**.
6. A **Configurations** screen might appear with warning messages about configuration and environment details. You can ignore these warnings and click **Proceed Anyway**.

   The **Configure Identities** screen displays default configuration properties.
7. Click **Next**.

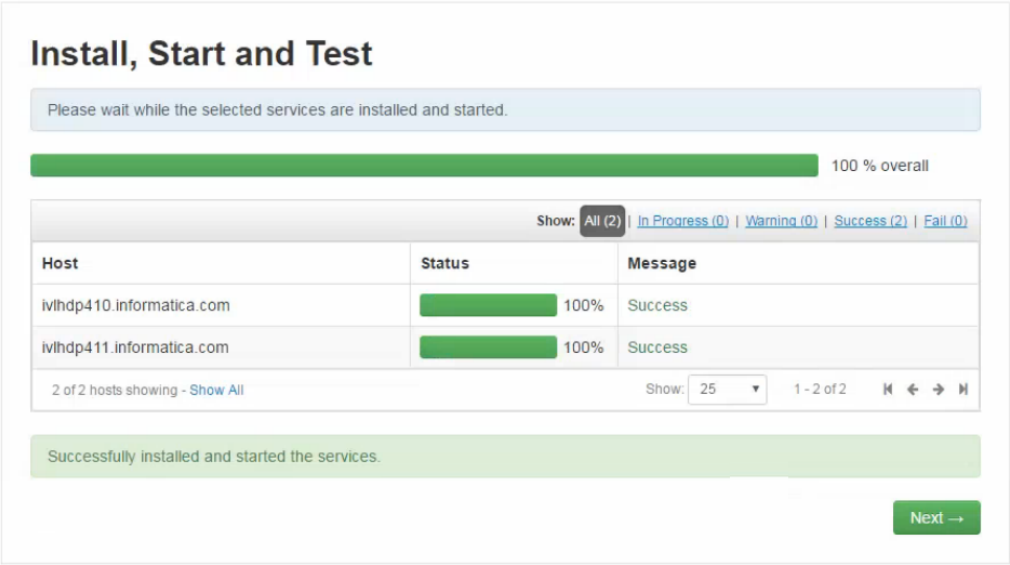
   The **Review** screen displays a summary of the client installation to perform.
8. Click **Deploy**.
9. If the cluster uses Kerberos authentication, the **Admin session expiration error** dialog box appears. Enter Administrator principal authentication credentials and click **Save**.

   The following image shows the dialog box populated with authentication credentials:



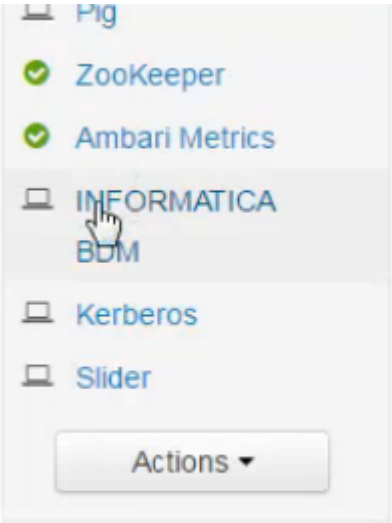
10. Monitor client installation in the **Install, Start and Test** screen.

    The following image shows the **Install, Start and Test** screen when installation is complete:



11. Click **Complete**.

    Big Data Management is installed on cluster nodes.
    The following image shows the list of stack services after you install Big Data Management:



Big Data Management is installed as an Ambari stack service.

Troubleshooting Installation in an Ambari Stack

If the **Install, Start and Test** screen indicates an error, you can review the install log for more information.

1. Click on the text of the message in the Message area of the **Install, Start and Test** screen.
   A dialog box opens that displays the series of installation tasks that Ambari performed or attempted to perform.

   The following image shows the dialog box:

2. Select **INFORMATICA BDM CLIENT Install** to view the install log.

The install log opens in a separate window.

CHAPTER 15

Hadoop Configuration Manager for Hortonworks HDP

This chapter includes the following topics:

- Hadoop Configuration Manager Overview, 111
- Start the Configuration Manager in Console Mode, 111
- Run the Configuration Manager in Silent Mode, 116

Hadoop Configuration Manager Overview

After you install Big Data Management on the Hadoop environment, you configure Big Data Management on the Informatica domain.

Configure Big Data Management using the Hadoop Configuration Manager (configuration manager). The configuration manager can create connections between the Hadoop environment and the Data Integration Service, updates Data Integration Service properties, and it updates configuration files on the Data Integration Service machine.

You can run the configuration manager in console mode or silent mode.

After you run the configuration manager, you must perform some additional configuration tasks.

Start the Configuration Manager in Console Mode

Run the Hadoop Configuration Manager in console mode to configure the Informatica domain for Big Data Management.

1. On the machine where the Data Integration Service runs, open the command line.
2. Go to the following directory: `<Informatica installation directory>/tools/BDMUtil`
3. Run `BDMConfig.sh`.
4. Choose option 2 to configure for Hortonworks HDP.
5. In the **Distribution Folder Selection** section, choose the directory of the Hortonworks HDP distribution that you want to configure.

Note: The distribution might be stored in a directory that uses a different distribution version number.

6. In the **Connection Type** section, select the option to access files on the Hadoop cluster.

Option	Description
1	Apache Ambari. Select this option to use the Ambari REST API to access files on the Hadoop cluster. Informatica recommends that you use this option.
2	Secure Shell (SSH). Select this option to use SSH to access files on the Hadoop cluster. This option requires SSH connections from the Informatica domain to the machines that host the NameNode, YARN ResourceManager, and Hive client. If you select this option, Informatica recommends that you use an SSH connection without a password or have sshpass or Expect installed.

Complete the configuration based on the option that you choose.

Complete the Configuration through Apache Ambari

To complete the configuration, you can choose to update Data Integration Service properties and create connection objects.

You might want to create connections if you are installing for the first time or if you are upgrading and the changes to the Hadoop environment are significant, such as a different distribution. If you are upgrading and you create connections that you want to use in existing mappings, you need to update mappings to use the new connections after you upgrade.

You might want to skip creating connections if the changes to the Hadoop environment are minimal, such as a change in distribution version. If you are upgrading and you choose not to create connections, you can continue to use the connections in existing mappings. However, you need update any changed properties after you upgrade. Property changes might include host names, URIs, or port numbers.

1. In the **Ambari Administration Information** section, enter the connection information to connect to the Ambari Manager.
   a. Enter the Ambari Manager host name or IP address and port.
   b. Enter the Ambari user ID.
   c. Enter the password for the user ID.
   d. Enter the port for Ambari Manager.
   e. Select whether to use MapReduce or Tez as the execution engine type.
      - 1 - MapReduce
      - 2 - Tez

   The Hadoop Configuration Manager retrieves the required information from the Hadoop cluster.
2. In the **Hadoop Configuration Manager Output** section, select whether you want to update Data Integration Service properties.

   Select from the following options:

Option	Description
1	No. Select this option to update Data Integration Service properties later.
2	Yes. Select this option to update Data Integration Service properties now.

3. Select whether you want to restart the Data Integration Service.

   Select from the following options:

Option	Description
1	No. Select this option if you do not want the configuration manager to restart the Data Integration Service.
2	Yes. Select this option if you want the configuration manager to restart the Data Integration Service when the configuration is complete.

4. Select whether you want to create connections for Big Data Management.

   Select from the following options:

Option	Description
1	No. Select this option if you do not want to create connections. If you are upgrading and you choose not to create connections, you can continue to use the connections in existing mappings. However, you need update any changed properties after you upgrade. Property changes might include host names, URIs, or port numbers.
2	Yes. Select this option if you want to create connections. If you are upgrading and you create connections that you want to use in existing mappings, you need to update mappings to use the new connections after you upgrade.

5. In the **Create Connections** section, select the connection type to create Big Data Management connections:

Option	Description
1. Hive	Create a Hive connection to access Hive as a source or target.
2. HDFS	Create an HDFS connection to read data from or write data to the HDFS file system on a Hadoop cluster.
3. Hadoop	Create a Hadoop connection to run mappings in the Hadoop environment.
4. HBase	Create an HBase connection to access HBase.
5. Select all	Create all four types of connection.

6. In the **Connection Details** section, provide the connection properties.

   Based on the type of connection that you choose to create, the Hadoop Configuration Manager requires different properties.

   The Hadoop Configuration Manager creates the connections.

7. The Hadoop Configuration Manager reports a summary of its operations, including whether connection creation succeeded and the location of log files.

The Hadoop Configuration Manager creates the following file in the `<Informatica installation directory>/tools/BDMUtil` directory:

ClusterConfig.properties.<timestamp>

Contains details about the properties fetched from the Hadoop cluster, including cluster node names, to provide templates for connection creation commands. To use these connection creation templates to

create connections to the Hadoop cluster, edit the connection name, domain username and password in the generated commands.

Complete the Configuration through SSH

When you complete configuration through SSH, you must provide host names and Hadoop configuration file locations.

Configuration through SSH requires SSH connections from the Informatica domain to the machines that host the NameNode, YARN ResourceManager, and Hive client. Informatica recommends that you use an SSH connection without a password or have sshpass or Expect installed. If you do not use one of these methods, you must enter the password each time the utility downloads a file from the Hadoop cluster.

1. Enter the name node host name.
2. Enter the SSH user ID.
3. Enter the password for the SSH user ID, or press enter if you use an SSH connection without a password.
4. Enter the location for the `hdfs-site.xml` file on the Hadoop cluster.
5. Enter the location for the `core-site.xml` file on the Hadoop cluster.

   The Hadoop Configuration Manger connects to the name node and downloads the following files: `hdfs-site.xml` and `core-site.xml`.
6. Enter the Yarn resource manager host name.

   Note: Yarn resource manager was formerly known as JobTracker.
7. Enter the SSH user ID.
8. Enter the password for the SSH user ID, or press enter if you use an SSH connection without a password.
9. Enter the directory for the `mapred-site.xml` file on the Hadoop cluster.
10. Enter the directory for the `yarn-site.xml` file on the Hadoop cluster.

    The utility connects to the JobTracker and downloads the following files: `mapred-site.xml` and `yarn-site.xml`.
11. Enter the Hive client host name.
12. Enter the SSH user ID.
13. Enter the password for the SSH user ID, or press enter if you use an SSH connection without a password.
14. Enter the directory for the `hive-site.xml` file on the Hadoop cluster.

    The configuration manager connects to the Hive client and downloads the following file: `hive-site.xml`.
15. Optionally, You can choose to configure the HBase server.

    Select the following options:

Option	Description
1	No. Select this option if you do not want to configure the HBase server.
2	Yes. Select this option to configure the HBase server.

    If you select **Yes**, enter the following information to configure the HBase server:

    a. Enter the HBase server host name.

    b. Enter the SSH user ID.

c. Enter the password for the SSH user ID, or press enter if you use an SSH connection without a password.

d. Enter the directory for the `hbase-site.xml` file on the Hadoop cluster.

16. In the **Create Connections** section, select the connection type to create Big Data Management connections:

Option	Description
1. Hive	Create a Hive connection to access Hive as a source or target.
2. HDFS	Create an HDFS connection to read data from or write data to the HDFS file system on a Hadoop cluster.
3. Hadoop	Create a Hadoop connection to run mappings in the Hadoop environment.
4. HBase	Create an HBase connection to access HBase.
5. Select all	Create all four types of connection.

Press the number that corresponds to your choice.

17. The **Domain Information** section displays the domain name and the node name. Enter the following additional information about Informatica domain.

a. Enter the domain user name.

b. Enter the domain password.

c. Enter the Data Integration Service name.

d. If the Hadoop cluster uses Kerberos authentication, enter the following information:

- Hadoop Kerberos service principal name
- Hadoop Kerberos keytab location. Location of the keytab on the Data Integration Service machine.

Note: After you enter the Data Integration Service name, the utility tests the domain connection, and then recycles the Data Integration Service.

18. In the **Connection Details** section, provide the connection properties.

Based on the type of connection you choose to create, the utility requires different properties.

The Hadoop Configuration Manager creates the connections.

19. The Hadoop Configuration Manager reports a summary of its operations, including whether connection creation succeeded and the location of log files.

The Hadoop Configuration Manager creates the following file in the `<Informatica installation directory>/tools/BDMUtil` directory:

ClusterConfig.properties.<timestamp>

Contains details about the properties fetched from the Hadoop cluster, including cluster node names, to provide templates for connection creation commands. To use these connection creation templates to create connections to the Hadoop cluster, edit the connection name, domain username and password in the generated commands.

Run the Configuration Manager in Silent Mode

To configure the Big Data Management without user interaction in silent mode, run the Hadoop Configuration Manager in silent mode. Use the `SilentInput.properties` file to specify the configuration options. The configuration manager reads the file to determine the configuration options.

To configure the domain for Big Data Management in silent mode, perform the following tasks:

1. Configure the properties in the `SilentInput.properties` file.
2. Run the configuration manager with the `SilentInput.properties` file.

Configure the Properties File

The `SilentInput.properties` file includes the properties that are required by the configuration manger. Customize the properties file to specify the options for your configuration and then run the configuration manager in silent mode.

You might want to create connections if you are installing for the first time or if you are upgrading and the changes to the Hadoop environment are significant, such as a different distribution. If you are upgrading and you create connections that you want to use in existing mappings, you need to update mappings to use the new connections after you upgrade.

You might want to skip creating connections if the changes to the Hadoop environment are minimal, such as a change in distribution version. If you are upgrading and you choose not to create connections, you can continue to use the connections in existing mappings. However, you need update any changed properties after you upgrade. Property changes might include host names, URIs, or port numbers.

1. Find the sample `SilentInput.properties` file in the following location: `<Informatica installation directory>/tools/BDMUtil`
2. Create a backup copy of the `SilentInput.properties` file.
3. Use a text editor to open the file and modify the values of the configuration properties.
   The following table describes the configuration properties that you can modify:

Property Name	Description
CLOUDERA_SELECTION	Indicates whether to configure the Informatica domain for the Cloudera CDH cluster. To configure the domain for the Cloudera CDH cluster, set the value of this property to 1.
HORTONWORKS_SELECTION	Indicates whether to configure the Informatica domain for the Hortonworks cluster. To configure the domain for the Hortonworks cluster, set the value of this property to 1.
BIG_INSIGHT	Indicates whether to configure the Informatica domain for the BigInsights cluster. To configure the domain for the BigInsights cluster, set the value of this property to 1.
HD_INSIGHT	Indicates whether to configure the Informatica domain for the HDInsight cluster. To configure the domain for the HDInsight cluster, set the value of this property to 1.

Property Name	Description
DIST_FOLDER_NAME	Indicates the Hadoop distribution folder name present in INFA_HOME/services/shared/hadoop.
INSTALL_TYPE	Indicates how to access files on the Hadoop cluster. Set the value depending on the Hadoop distribution. To access files on the Cloudera CDH cluster using the Cloudera Manager, set the value of this property to 0. To access files on the Hortonworks HDP, IBM BigInsights, or the Azure HDInsight cluster using Apache Ambari, set the value of this property to 3.
CLOUDERA_HOSTNAME	Required if you set INSTALL_TYPE=0. Indicates the host name of the Cloudera Manager.
CLOUDERA_USER_NAME	Required if you set INSTALL_TYPE=0. Indicates the user id for the Cloudera Manager.
CLOUDERA_USER_PASSWD	Required if you set INSTALL_TYPE=0. Indicates the password for the Cloudera Manager.
CLOUDERA_PORT	Required if you set INSTALL_TYPE=0. Indicates the port of the Cloudera Manager.
CLOUDERA_SINGLECLUSTER_NAME	Required if you set INSTALL_TYPE=0. Indicates the name of the Cloudera CDH cluster for which you want to configure Big Data Management. If the Cloudera Manager manages a single cluster, you do not have to set this property. Retain this property as a comment or remove the property from the file. If the Cloudera Manager manages more than one cluster, uncomment and set the property to the name of the cluster for which you want to configure Big Data Management. Note: In a multi-cluster environment, the configuration fails if the property is commented or if the property has an invalid cluster name as the value.
AMBARI_HOSTNAME	Required if you set INSTALL_TYPE =3. Indicates the password for Apache Ambari.
AMBARI_USER_NAME	Required if you set INSTALL_TYPE=3. Indicates the port number for Apache Ambari.
AMBARI_USER_PASSWD	Required if you set INSTALL_TYPE=3. Indicates the password for Apache Ambari.
AMBARI_PORT	Required if you set INSTALL_TYPE=3. Indicates the port number for Apache Ambari.
TEZ_EXECUTION_MODE	Required if you set INSTALL_TYPE=3. Indicates whether to enable Tez for the Hive engine. To enable Tez on the Hortonworks HDP or Azure HDInsight cluster, set the value of this property to 1. Otherwise, set the value to 0.

Property Name	Description
CREATE_CONNECTION	Indicates whether to update the Data Integration Service and create connections for Big Data Management. To update the Data Integration Service and create connections, set the value of this property to 1 and define the connection properties. If you are upgrading and you choose not to create connections, you can continue to use the connections in existing mappings. However, you need update any changed properties after you upgrade. Property changes might include host names, URIs, or port numbers. If you are upgrading and you create connections that you want to use in existing mappings, you need to update mappings to use the new connections after you upgrade. To exit the configuration manager, set the value to 0.
UPDATE_DIS	Required if you set CREATE_CONNECTION=1. Indicates whether to update the Data Integration Service properties. To update the Data Integration Service, set this property to 1. Otherwise, set the value to 0.
CREATE_BDM_CONNECTIONS	Required if you set CREATE_CONNECTION=1. Set this property to 1 to create Hadoop, HDFS, HBase, and Hive connections. Otherwise, the configuration manager will not create connections.
DOMAIN_USER	Required if you set CREATE_CONNECTION=1. Indicates the user name for the domain administrator.
DOMAIN_PSSWD	Required if you set CREATE_CONNECTION=1. Indicates the password for the domain administrator.
DIS_SERVICE_NAME	Required if you set CREATE_CONNECTION=1 and UPDATE_DIS=1. Indicates the name of the Data Integration Service.
CLUSTER_INSTALLATION_DIR	Required if you set CREATE_CONNECTION=1 and UPDATE_DIS=1. Indicates the Informatica home directory on every node on the Hadoop cluster that is created during the Big Data Management installation. Default is /opt/Informatica.
KERBEROS_PRINCIPAL_NAME	Required if you set CREATE_CONNECTION=1 and UPDATE_DIS=1 and if the Hadoop cluster uses Kerberos authentication. Indicates the Hadoop Kerberos service principal name.
KERBEROS_KEYTAB	Required if you set CREATE_CONNECTION=1 and UPDATE_DIS=1 and if the Hadoop cluster uses Kerberos authentication. Indicates the Hadoop Kerberos keytab location.
HIVE_CONNECTION	Required if you set CREATE_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Set this property to 1 to create a Hive connection. Otherwise, set the value to 0.

Property Name	Description
HDFS_CONNECTION	Required if you set CREATE_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Set this property to 1 to create a HDFS connection. Otherwise, set the value to 0.
HADOOP_CONNECTION	Required if you set CREATE_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Set this property to 1 to create a Hadoop connection. You must create a Hadoop connection to run the mappings in the Hadoop environment.
HBASE_CONNECTION	Required if you set CREATE_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Set this property to 1 to create a HBASE connection. Otherwise, set the value to 0.
SELECT_ALL_CONNECTION	Required if you set CREATE_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Set this property to 1 to create all listed connections. Also set HIVE_CONNECTION=1, HDFS_CONNECTION=1, HADOOP_CONNECTION=1, and HBASE_CONNECTION=1 .
SAMPLE_HIVE_CONNECTION	Required if CREATE_CONNECTION=1, HIVE_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Indicates connection name for Hive.
HIVE_EXECUTION_MODE	Required if you set CREATE_CONNECTION=1, HIVE_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Indicates the execution mode for Hive. Set this property to either Local or Remote.
HIVE_USER_NAME	Required if you set CREATE_CONNECTION=1, HIVE_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Indicates the user name for Hive. You must set the execution mode to Remote.
SAMPLE_HDFS_CONNECTION	Required if you set CREATE_CONNECTION=1, HDFS_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Indicates the connection name for HDFS.
HDFS_USER_NAME	Required if you set CREATE_CONNECTION=1, HDFS_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Indicates the user name to access HDFS.
SAMPLE_HADOOP_CONNECTION	Required if you set CREATE_CONNECTION=1 , HADOOP_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Indicates the connection name for Hadoop.
BLAZE_WORKING_DIR	Required if you set CREATE_CONNECTION=1 , HADOOP_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Indicates the existing Blaze work directory on HDFS.

Property Name	Description
BLAZE_USER	Required if you set CREATE_CONNECTION=1, HADOOP_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Indicates the user name for Blaze.
SPARK_HDFS_STAGING_DIR	Required if you set CREATE_CONNECTION=1 , HADOOP_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Indicates the existing Spark HDFS staging directory.
SPARK_EVENTLOG_DIR	Required if you set CREATE_CONNECTION=1, HADOOP_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Optional: Indicates the Spark event log directory
SPARK_PARAMETER_LIST	Required if you set CREATE_CONNECTION=1, HADOOP_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Indicates the Spark execution parameters list.
IMPERSONATION_USER	Required if you set CREATE_CONNECTION=1, HADOOP_CONNECTION=1, CREATE_BDM_CONNECTIONS=1, and if the Hadoop cluster uses Kerberos authentication.
SAMPLE_HBASE_CONNECTION	Required if you set CREATE_CONNECTION=1 , HBASE_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Indicates the connection name for HBase.
ZOOKEEPER_HOSTS	Required if you set CREATE_CONNECTION=1 , HBASE_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Indicates the ZooKeeper host.

Run the Configuration Manager

After you configure the properties file, open a command prompt to start the silent configuration.

1. Go to the following directory: `<Informatica installation directory>/tools/BDMUtil`
2. Verify that the directory contains the `SilentInput.properties` file that you edited.
3. Run the `BDMSilentConfig.sh` file.

The installer runs in the background. The process can take a while. The silent configuration is complete when the following log files are created in the `<Informatica installation directory>/tools/BDMUtil` directory:

- Informatica_Big_Data_Edition_Configuration_Utility.log.<timestamp>
- ClusterConfig.properties.<timestamp>

The silent configuration fails if you incorrectly configure the properties file or if the installation directory is not accessible. The `silenterror.log` file is created in the home directory of the UNIX user that runs the configuration manager in silent mode. Correct the errors and then run the silent configuration again.

CHAPTER 16

Manual Configuration Tasks for Hortonworks HDP

This chapter includes the following topics:

- Download the JDBC Drivers for Sqoop Connectivity, 121
- Update Files on the Hadoop Environment for Hortonworks HDP, 122
- Update Files on the Domain Environment for Hortonworks HDP, 125
- Creating and Configuring Security Certificates and Truststore Files, 128
- Update Files on the Developer Tool, 129
- Complete Upgrade Tasks, 130

Download the JDBC Drivers for Sqoop Connectivity

To configure Sqoop connectivity for relational databases, you must download the relevant JDBC driver jar files and copy the jar files to the node where the Data Integration Service runs.

You can use any Type 4 JDBC driver that the database vendor recommends for Sqoop connectivity. Download additional files if you use Hortonworks Connector for Teradata.

1. Download the JDBC driver jar files for the database that you want to connect to.
2. If you use Hortonworks Connector for Teradata, download additional files:
   - Download the Hortonworks Connector for Teradata package from the following URL: http://hortonworks.com/downloads/#addons
   - Download the `avro-mapred-1.7.4-hadoop2.jar` file from the following URL: https://archive.apache.org/dist/avro/avro-1.7.4/java/
3. Copy the jar files to the following directory on the machine where the Data Integration Service runs:

   ```
   <Informatica installation directory>\externaljdbcjars
   ```

At run time, the Data Integration Service copies the jar files to the Hadoop distribution cache so that the jar files are accessible to all nodes in the cluster.

Note: The JDBC drivers that Informatica ships are not licensed for Sqoop connectivity.

Update Files on the Hadoop Environment for Hortonworks HDP

You can configure additional functionality through the configuration files the Hadoop environment.

When you use Apache Ambari to update properties, updates are propagated to each node in the cluster. After you make updates, you can restart the Hadoop cluster and services through Apache Ambari.

Update core-site.xml

Update the core-site.xml file on the Hadoop environment to enable access to Hive tables on the cluster and configure user impersonation for Sqoop mappings.

Configure Access to Hive Tables in Amazon S3 Buckets

Configure the AWS access key to run mappings with sources and targets on Hive tables in Amazon S3 buckets.

Note: To use a Hive table as a target on Amazon S3, grant write permission to the bucket through bucket policies, or add these properties to the configuration file. You must add these properties to the core-site.xml file on the Hadoop environment and to the yarn-site.xml file on the domain environment.

Configure the following properties:

fs.s3n.awsAccessKeyId

The ID for the Blaze and Spark engines to connect to the Amazon S3n file system. For example,

```
<property>
    <name>fs.s3n.awsAccessKeyId</name>
    <value>[Your Access Key]</value>
 </property>
```

fs.s3n.awsSecretAccessKey

The password for the Blaze and Spark engines to connect to the Amazon S3n file system.

```
<property>
    <name>fs.s3n.awsSecretAccessKey</name>
    <value>[Your Access Id]</value>
</property>
```

When you use a server side encryption protocol for Hive buckets, configure properties to enable access to encrypted Hive buckets.

Configure the following properties:

fs.s3.enableServerSideEncryption

Set to TRUE to enable server side encryption. For example,

```
<property>
    <name>fs.s3.enableServerSideEncryption</name>
    <value>TRUE</value>
 </property>
```

fs.s3n.server-side-encryption-algorithm

The encryption algorithm that you use to encrypt Hive buckets. For example,

```
<property>
    <name>fs.s3n.server-side-encryption-algorithm</name>
    <value>AES256</value>
 </property>
```

Configure Impersonation for Sqoop

To run Sqoop mappings on clusters that do not use Kerberos authentication, you must create a proxy user for the yarn user who will impersonate other users.

Configure the following properties, and then restart Hadoop services and the cluster:

hadoop.proxyuser.yarn.groups

Allows impersonation from any group.

```
<property>
    <name>hadoop.proxyuser.yarn.groups</name>
    <value><Name_of_the_impersonation_user></value>
    <description>Allows impersonation from any group.</description>
</property>
```

hadoop.proxyuser.yarn.hosts

Allows impersonation from any host.

```
<property>
    <name>hadoop.proxyuser.yarn.hosts</name>
    <value>*</value>
    <description>Allows impersonation from any host.</description>
</property>
```

Update hive-site.xml

Update hive-site.xml on the Hadoop environment to enable mappings to read or write to Hive tables on the cluster.

When the cluster uses Kerberos authentication and the mapping is configured to use the Blaze or Spark run-time engines, configure the following property in `hive-site.xml` on the cluster:

hive.server2.enable.doAs

The authentication that the server is set to use. Default is FALSE.

To enable access to HiveServer2, set the property to TRUE. When you set the property to TRUE, you enable the Blaze and Spark engines to access the data in the Hive tables as the DIS user, or as the impersonation user when impersonation is enabled.

When you set the property to FALSE, you enable HiveServer2 to run MapReduce jobs as the Hive user.

Update yarn-site.xml

Update the yarn-site.xml file on the Hadoop environment to configure functionality such as dynamic resource allocation and virtual memory limits.

Configure Dynamic Resource Allocation for the Spark Engine

Configure dynamic resource allocation for mappings to run on the Spark engine. Update the following properties on each node where YARN node manager is running:

yarn.nodemanager.aux-services

The auxiliary service name. Add spark_shuffle to the current entry. For example, if the current value is "mapreduce_shuffle," update it to be "mapreduce_shuffle,spark_shuffle."

```
<property>
    <name>yarn.nodemanager.aux-services</name>
    <value>mapreduce_shuffle,spark_shuffle</value>
</property>
```

yarn.nodemanager.aux-services.spark_shuffle.class

The auxiliary service class to use. Set to org.apache.spark.network.yarn.YarnShuffleService. For example,

```
<property>
    <name>yarn.nodemanager.aux-services.spark_shuffle.class</name>
    <value>org.apache.spark.network.yarn.YarnShuffleService</value>
</property>
```

yarn.application.classpath

The classpath for YARN applications. Verify that the yarn classpath includes the path to the Spark shuffle.jar file packaged with the distribution. If the .jar file is not in any Hadoop distribution directory, update the path to use the .jar file in the Apache Spark distribution directory. For example,

```
<property>
    <name>yarn.application.classpath</name>
    <value>/opt/Informatica/services/shared/spark/lib_spark_2.0.1_hadoop_2.6.0/spark-
networkshuffle_2.11-2.0.1.jar</value>
</property>
```

Allocate Cluster Resources for the Blaze Engine

Update the following properties to verify that the cluster allocates sufficient memory and resources for the Blaze engine:

yarn.nodemanager.resource.memory-mb

The maximum RAM available for each container. Set the maximum memory on the cluster to increase resource memory available to the Blaze engine. Configure this property on the following nodes:

- Run-time nodes. Set to at least 10 GB.
- Management node. Set to at least 13 GB.

yarn.nodemanager.resource.cpu-vcores

The number of virtual cores for each container. The number might correspond to the number of physical cores, but you can increase the value to allow for more processing. Configure this property on the following nodes:

- Run-time nodes. Set to at 6.
- Management node. Set to 9.

yarn.scheduler.minimum-allocation-mb

The minimum RAM available for each container. Set the minimum memory to allow the VM to spawn sufficient containers. Configure this property to be no less than 4 GB of the maximum memory on the run-time and management nodes.

Configure Virtual Memory Limits for the Blaze Engine

Configure the following property to remove virtual memory limits for every node in the Hadoop cluster:

yarn.nodemanager.vmem-check-enabled

Determines virtual memory limits. Set the value to false to disable virtual memory limits. For example,

```
<property>
    <name>yarn.nodemanager.vmem-check-enabled</name>
    <value>false</value>
    <description>Enforces virtual memory limits for containers.</description>
</property>
```

Update Files on the Domain Environment for Hortonworks HDP

You can configure additional functionality through the configuration files on the domain.

When the Informatica installer creates nodes on the domain, it creates directories and files for Big Data Management. When you configure Big Data Management on the domain, you need to edit configuration files to enable functionality.

The following table describes the installation directories and the default paths:

Directory	Description
Data Integration Service Hadoop distribution directory	The Hadoop distribution directory on the Data Integration Service node. It corresponds to the distribution on the Hadoop environment. You can view the directory path in the Data Integration Service properties of the Administrator tool. Default is `<Informatica installation directory>/Informatica/services/shared/hadoop/<Hadoop distribution name>_<version>`.
Configuration file directories	Configuration files for Big Data Management are stored in the following directories: - `<Informatica installation directory>/Informatica/services/shared/hadoop/<Hadoop distribution name>_<version>/InfaConf`. Contains the hadoopEnv.properties file. - `<Informatica installation directory>/Informatica/services/shared/hadoop/<Hadoop distribution name>_<version>/conf`. Contains *-site files.

Update hadoopEnv.properties

Update the hadoopEnv.properties file to configure functionality such as Sqoop connectivity and the Spark run-time engine.

Open hadoopEnv.properties and back it up before you configure it. You can find the hadoopEnv.properties file in the following location: `<Informatica installation directory>/services/shared/hadoop/<Hadoop distribution name>_<version number>/infaConf`

Configure Performance for the Spark Engine

Configure the following performance properties for the Spark engine:

spark.dynamicAllocation.enabled

Performance configuration to run mappings on the Spark engine. Enables dynamic resource allocation. Required when you enable the external shuffle service.

Set the value to TRUE.

spark.shuffle.service.enabled

Performance configuration to run mappings on the Spark engine. Enables the external shuffle service. Required when you enable dynamic resource allocation.

Set the value to TRUE.

spark.scheduler.maxRegisteredResourcesWaitingTime

Performance configuration to run mappings on the Spark engine. The number of milliseconds to wait for resources to register before scheduling a task. Reduce this from the default value of 30000 to reduce delays before starting the Spark job execution.

Set the value to 15000.

spark.scheduler.minRegisteredResourcesRatio

Performance configuration to run mappings on the Spark engine. The minimum ratio of registered resources to acquire before task scheduling begins. Reduce this from the default value of 0.8 to reduce any delay before starting the Spark job execution.

Set the value to .5.

spark.executor.instances

Performance configuration to run mappings on the Spark engine. If you enable dynamic resource allocation for the Spark engine, Informatica recommends that you convert this property to a comment. For example, `#spark.executor.instances=100`

Configure Sqoop Connectivity

Configure the following property for Sqoop connectivity:
infapdo.env.entry.hadoop_node_jdk_home

Configure the HADOOP_NODE_JDK_HOME to represent the JDK version that the cluster nodes use. You must use JDK version 1.7 or later.

Configure the property as follows:

```
infapdo.env.entry.hadoop_node_jdk_home=HADOOP_NODE_JDK_HOME=<cluster JDK home>/
jdk<version>
```

For example,

```
infapdo.env.entry.hadoop_node_jdk_home=HADOOP_NODE_JDK_HOME=/usr/java/default
```

Configure Environment Variables

You can optionally add third-party environment variables and extend the existing PATH environment variable in the hadoopEnv.properties file. The following text shows sample entries to configure environment variables:

```
infapdo.env.entry.oracle_home=ORACLE_HOME=/databases/oracle
infapdo.env.entry.db2_home=DB2_HOME=/databases/db2
infapdo.env.entry.db2instance=DB2INSTANCE=OCA_DB2INSTANCE
infapdo.env.entry.db2codepage=DB2CODEPAGE="1208"
infapdo.env.entry.odbchome=ODBCHOME=$HADOOP_NODE_INFA_HOME/ODBC7.1
infapdo.env.entry.home=HOME=/opt/thirdparty
infapdo.env.entry.gphome_loaders=GPHOME_LOADERS=/databases/greenplum
infapdo.env.entry.pythonpath=PYTHONPATH=$GPHOME_LOADERS/bin/ext
infapdo.env.entry.nz_home=NZ_HOME=/databases/netezza
infapdo.env.entry.ld_library_path=LD_LIBRARY_PATH=$HADOOP_NODE_INFA_HOME/services/
shared/bin:$HADOOP_NODE_INFA_HOME/DataTransformation/bin:$HADOOP_NODE_HADOOP_DIST/lib/
native:$HADOOP_NODE_INFA_HOME/ODBC7.1/lib:$HADOOP_NODE_INFA_HOME/jre/lib/
amd64:$HADOOP_NODE_INFA_HOME/jre/lib/amd64/server:$HADOOP_NODE_INFA_HOME/java/jre/lib/
amd64:$HADOOP_NODE_INFA_HOME/java/jre/lib/amd64/server:/databases/oracle/lib:/
databases/db2/lib64:$LD_LIBRARY_PATH
infapdo.env.entry.path=PATH=$HADOOP_NODE_HADOOP_DIST/scripts:$HADOOP_NODE_INFA_HOME/
services/shared/bin:$HADOOP_NODE_INFA_HOME/jre/bin:$HADOOP_NODE_INFA_HOME/java/jre/bin:
$HADOOP_NODE_INFA_HOME/ODBC7.1/bin:/databases/oracle/bin:/databases/db2/bin:$PATH
#teradata
infapdo.env.entry.twb_root=TWB_ROOT=/databases/teradata/tbuild
infapdo.env.entry.manpath=MANPATH=/databases/teradata/odbc_64:/databases/teradata/odbc_64
infapdo.env.entry.nlspath=NLSPATH=/databases/teradata/odbc_64/msg/%N:/databases/
teradata/msg/%N
infapdo.env.entry.pwd=PWD=/databases/teradata/odbc_64/samples/C
```

Update hive-site.xml

Update the hive-site.xml file on the domain environment to set properties for dynamic partitioning.

Configure Dynamic Partitioning

To use Hive dynamic partitioned tables, configure the following properties:

hive.exec.dynamic.partition

Enables dynamic partitioned tables. Set this value to `TRUE`.

exec.dynamic.partition.mode

Allows all partitions to be dynamic. Set this value to `nonstrict`.

Update yarn-site.xml

Update the yarn-site.xml file on the domain to configure functionality such as access to Hive buckets.

Configure Access to Hive Tables in Amazon S3 Buckets

Configure the AWS access key to run mappings with sources and targets on Hive tables in Amazon S3 buckets.

Note: To use a Hive table as a target on Amazon S3, grant write permission to the bucket through bucket policies, or add these properties to the configuration file. You must add these properties to the core-site.xml file on the Hadoop environment and to the yarn-site.xml file on the domain environment.

Configure the following properties:

fs.s3n.awsAccessKeyId

The ID for the Blaze and Spark engines to connect to the Amazon S3n file system. For example,

```
<property>
    <name>fs.s3n.awsAccessKeyId</name>
    <value>[Your Access Key]</value>
 </property>
```

fs.s3n.awsSecretAccessKey

The password for the Blaze and Spark engines to connect to the Amazon S3n file system.

```
<property>
    <name>fs.s3n.awsSecretAccessKey</name>
    <value>[Your Access Id]</value>
</property>
```

When you use a server side encryption protocol for Hive buckets, configure properties to enable access to encrypted Hive buckets.

Configure the following properties:

fs.s3.enableServerSideEncryption

Set to TRUE to enable server side encryption. For example,

```
<property>
    <name>fs.s3.enableServerSideEncryption</name>
    <value>TRUE</value>
 </property>
```

fs.s3n.server-side-encryption-algorithm

The encryption algorithm that you use to encrypt Hive buckets. For example,

```
<property>
    <name>fs.s3n.server-side-encryption-algorithm</name>
    <value>AES256</value>
 </property>
```

Creating and Configuring Security Certificates and Truststore Files

When you use custom, special, or self-signed security certificates to secure the Hadoop cluster, Informatica clients that connect to the cluster require these certificates to be present in the client machine truststore.

Import Security Certificates

To connect to the Hadoop cluster to develop a mapping, the Developer tool requires security certificate aliases on the machine that hosts the Developer tool. To run a mapping, the machine that hosts the Data Integration Service requires these same certificate alias files.

Perform the following steps from the Developer tool host machine and from the Data Integration Service host machine:

1. Run the following command to export the certificates from the cluster:

   ```
   keytool -export -alias <alias name> -keystore <custom.truststore file location> -
   file <exported certificate file location> -storepass <password>
   ```

   For example:

   ```
   <java home>/jre/bin/keytool -export -alias <alias name> -keystore ~/
   custom.truststore -file ~/exported.cer
   ```

   The command produces a certificate file.

2. Choose to import security certificates to an SSL-enabled domain or a domain that is not SSL-enabled.
   - If the domain is SSL-enabled, then import the certificate alias file to the following locations:
     - The following path on the Developer tool machine: `<Developer Tool home>\clients\shared \security\infa_truststore.jks`
     - The following path on the machine that hosts the Data Integration Service: `<Informatica installation home>/services/shared/security/infa_truststore.jks`
   - If the Informatica domain is not SSL-enabled, import the security certificate alias file to the following locations:
     - The following path on the Developer tool machine: `<Developer Tool home>\clients\java\jre\lib \security\cacerts`
     - The following path on the machine that hosts the Data Integration Service: `<Informatica installation home>/java/jre/lib/security/cacerts`

Configure the Data Integration Service to Use Truststore File Paths

To enable the Data Integration Service to access truststore files on the Hadoop cluster, perform the following steps to configure Data Integration Service properties with truststore file paths:

1. Get the location of truststore files from the cluster manager administration web page.
   If you do not have access to the cluster manager, ask the cluster administrator for the path to the truststore files for the resources you want to access.
2. To enable access to Hive sources in Native mode, copy the truststore files to the corresponding location in the Hadoop distribution directory of the Informatica domain machine.
   For example, if the truststore file is located at `/etc/security/serverKeys/all.jks` on the cluster, copy the file to the same location in the Hadoop distribution directory on the domain machine: `/etc/ security/serverKeys/all.jks`. Create the directory if it does not exist.
3. In the Administrator tool, select the Data Integration Service in the **Domain Navigator**. Click the **Properties** tab to display Data Integration Service properties.

4. Click the **Edit** icon for Custom Properties.
   The **Edit Custom Properties** dialog box appears.
5. Enter the following name for the custom property: JVMOption.
   If a JVMOption custom property exists, then increment the name with an integer like JVMOption1.
6. In the **Value** pane for the property, type the following value:

   ```
   -Djavax.net.ssl.trustStore=<path to the truststore file on the cluster>
   ```

   For example:

   ```
   -Djavax.net.ssl.trustStore=/etc/security/serverKeys/all.jks
   ```
7. Click **OK**.
8. To access additional resources that use truststore files in different locations, repeat steps 4-7. Increment the custom property name with an integer.
   The following image shows custom properties that allow the Data Integration Service to access truststore files in two different locations:

   Note: The Data Integration service converts the name of custom properties to add the "ExecutionContextOptions." prefix when you recycle the service.
9. Recycle the Data Integration Service.
10. If you have ever run a mapping on the Blaze engine, you must stop the Grid Manager application on the Data Integration Service host machine.
    a. Run the following command to list existing YARN applications:

       ```
       yarn application -list
       ```
    b. In the list of YARN applications, identify the application ID for the Blaze Grid Manager.
    c. Run the following command to stop the Grid Manager application:

       ```
       yarn application -kill <application ID>
       ```

Now you can run the mapping from the Developer tool. The Data Integration System imports the truststore certificates at run time.

Update Files on the Developer Tool

Update configuration files to enable communication and high availability between the Developer tool and the cluster.

Before you perform configuration on the Developer tool, you need to copy some configuration files to the machine that hosts the Developer tool.

1. Find hdfs-site.xml and yarn-site.xml on the name node of the cluster.
2. Copy the files to the following location on the machine that hosts the Developer tool: `<Informatica installation directory>/clients/DeveloperClient/Hadoop/<Hadoop distribution><version>/conf`

yarn-site.xml

Configure the following property in yarn-site.xml:

yarn.application.classpath

Sets the classpath for YARN applications and enables the Developer tool to read from and write to a highly available cluster.

To identify the classpath, run the `yarn classpath` command. The following sample text shows a sample classpath entry:

```
<property>
<name>yarn.application.classpath</name>
<value>
/etc/hadoop/conf:/opt/cloudera/parcels/CDH-5.4.0-1.cdh5.4.0.p0.27/lib/hadoop/
libexec/../../hadoop/lib/*:/opt/cloudera/parcels/CDH-5.4.0-1.cdh5.4.0.p0.27/lib/
hadoop/libexec/../../hadoop/.//*:/opt/cloudera/parcels/
CDH-5.4.0-1.cdh5.4.0.p0.27/lib/hadoop/libexec/../../hadoop-hdfs/./:/opt/cloudera/
parcels/CDH-5.4.0-1.cdh5.4.0.p0.27/lib/hadoop/libexec/../../hadoop-hdfs/lib/*:/opt/
cloudera/parcels/CDH-5.4.0-1.cdh5.4.0.p0.27/lib/hadoop/libexec/../../hadoophdfs/.//
*:/
opt/cloudera/parcels/CDH-5.4.0-1.cdh5.4.0.p0.27/lib/hadoop/libexec/../../
hadoop-yarn/lib/*:/opt/cloudera/parcels/CDH-5.4.0-1.cdh5.4.0.p0.27/lib/hadoop/
libexec/../../hadoop-yarn/.//*:/opt/cloudera/parcels/CDH/lib/hadoopmapreduce/
lib/*:/opt/cloudera/parcels/CDH/lib/hadoop-mapreduce/.//*:/opt/cloudera/
parcels/CDH-5.4.0-1.cdh5.4.0.p0.27/lib/hadoop/libexec/../../hadoop-yarn/.//*:/opt/
cloudera/parcels/CDH-5.4.0-1.cdh5.4.0.p0.27/lib/hadoop/libexec/../../hadoopyarn/
lib/*
</value>
</property>
```

developerCore.ini

Edit developerCore.ini to enable communication between the Developer tool and the Hadoop cluster. You can find developerCore.ini in the following directory: `<Informatica installation directory>\clients\DeveloperClient`

Add the following property:

```
-DINFA_HADOOP_DIST_DIR=hadoop\<distribution><version>
```

The change takes effect when you restart the Developer tool.

Complete Upgrade Tasks

If you upgraded the Informatica platform or applied the hotfix, you need to perform some additional tasks within the Informatica domain.

Based on the version that you upgraded from, perform the following tasks:

Update the hadoopEnv.properties file.

Applies to all upgrades. The 10.1.1 HotFix 1 hadoopEnv.properties file contains additional properties. You need to manually update it to include customized configuration from previous versions.

Complete connection upgrades.

Applies to upgrades from 9.6.1 or any 9.6.1 hotfix. You need to generate Hadoop connections from the Hive connection and update parameter files.

Update connection properties.

Applies to any upgrade. If you did not create connections when you ran the Hadoop Configuration Manager, you must update values for connection properties that changed.

Replace connections.

Applies to any upgrade. If you created connections when you ran the Hadoop Configuration Manager, you need to replace connections in mappings with the new connections.

Update the hadoopEnv.properties File

When you run the Informatica upgrade or apply the hotfix on a machine that hosts the Data Integration Service, the installer creates a new hadoopEnv.properties file and backs up the existing configuration file.

You need to edit the version 10.1.1 HotFix 1 hadoopEnv.properties file to include any manual configuration that you performed in the previous version for the corresponding distribution.

If you applied the hotfix, you can find the backup hadoopEnv.properties file in the following location:

```
<Informatica installation directory>/BACK_FROM_HF1/services/shared/hadoop/<Hadoop
distribution name>_<version>/infaConf
```

If you upgraded from a previous version, you can find the backup hadoopEnv.properties file in the following location:

```
<Previous Informatica installation directory>/services/shared/hadoop/<Hadoop distribution
name>_<version>/infaConf
```

You can find the 10.1.1 HotFix 1 hadoopEnv.properties file in the following location:

```
<Informatica installation directory>/services/shared/hadoop/<Hadoop distribution
name>_<version>/infaConf
```

Complete Connection Upgrade

Effective in version 10.0, Big Data Management requires a Hadoop connection to run mappings on the Hadoop cluster. If you upgraded from 9.6.1 or any 9.6.1 hotfix, you must generate Hadoop connections from Hive connections that are enabled to run mappings in the Hadoop environment.

The upgrade process generates a connection name for the Hadoop connection and replaces the connection name in the mappings. It does not create the physical connection object. When the upgrade is complete, you must run a command to generate the connection. Generate Hadoop connections from Hive connections if the following conditions are true:

- You upgraded from 9.6.1 or any 9.6.1 hotfix.
- The Hive connections are configured to run mappings in the Hadoop environment.

Complete the following tasks to upgrade connections:

1. Run `infacmd isp generateHadoopConnectionFromHiveConnection` to generate a Hadoop connection from a Hive connection that is configured to run in the Hadoop environment.
   The command renames the connection as follows: "Autogen_<Hive connection name>." If the renamed connection exceeds the 128 character limit, the command fails.
2. If the command fails, complete the following tasks:
   a. Rename the connection to meet the character limit and run the command again.

b. Run `infacmd dis replaceMappingHadoopRuntimeConnections` to replace connections associated with mappings that are deployed in applications.

c. Run `infacmd mrs replaceMappingHadoopRuntimeConnections` to replace connections associated with mappings that you run from the Developer tool.

3. If the Hive connection was parameterized, you must update the connection names in the parameter file. Verify that the Hive sources, Hive targets, and the Hive engine parameters are updated with the correct connection name.
4. The generated Hadoop connection uses the MRV2 as the engine type. If you use Tez to run mappings, you must update the engine type.
5. If any properties changed in the cluster, such as host names, URIs, or port numbers, you must update the properties in the connections.

For information about the infacmd commands, see the *Informatica Command Reference*.

Update Connection Properties

If you upgraded from version 10.0 or later, and you did not create connections when you ran the Hadoop Configuration Manager, you need to verify connection properties.

Review the Hadoop, Hive, HDFS, and HBase connections that you created in a previous release to update the values for connection properties. For example, if you added nodes to the cluster or if you updated the distribution version, you might need to verify host names, URIs, or port numbers for some of the properties.

Replace the Connections

If you created connections when you ran the Hadoop Configuration Manager, you need to replace connections in mappings with the new connections.

The method that you use to replace connections in mappings depends on the type of connection.

Hadoop connection

If you created a Hadoop connection when you ran the Hadoop Configuration Manager, run the following commands:

- `infacmd dis replaceMappingHadoopRuntimeConnections`. Replaces connections associated with mappings that are deployed in applications.
- `infacmd mrs replaceMappingHadoopRuntimeConnections`. Replaces connections associated with mappings that you run from the Developer tool.

For information about the infacmd commands, see the *Informatica Command Reference*.

Hive, HDFS, or HBase connection

If you created a Hive, HDFS, or HBase connection when you ran the Hadoop Configuration Manager, you must replace the connections manually.

CHAPTER 17

Uninstall for Hortonworks HDP

This chapter includes the following topic:

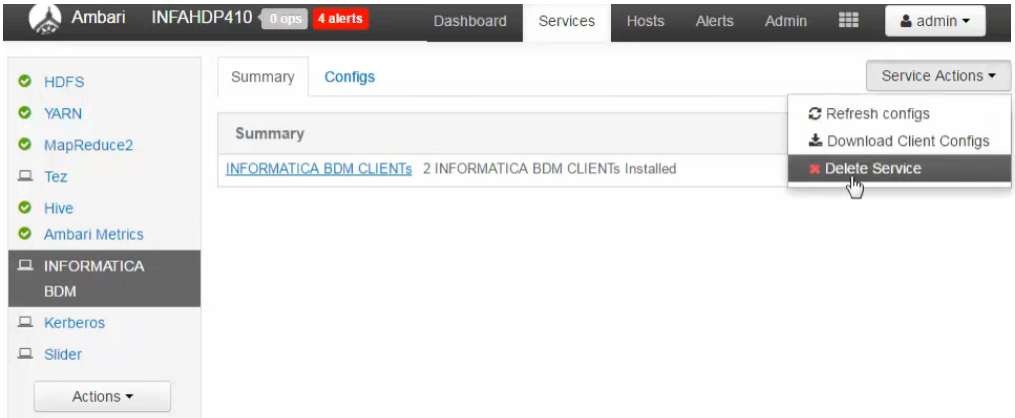
- Uninstall Big Data Management, 133

Uninstall Big Data Management

To uninstall the stack deployment of Big Data Management, you use the Ambari configuration manager to stop and deregister the Big Data Management service, and then perform manual removal of Informatica files from the cluster.

1. In the Ambari configuration manager, select **INFORMATICA BDM** from the list of services.
2. Click the **Service Actions** dropdown menu and select **Delete Service**.

   The following image shows the option in the **Service Actions** dropdown menu:



3. To confirm that you want to delete Informatica Big Data Management, perform the following steps:
   a. In the **Delete Service** dialog box, click **Delete**.
   b. In the Confirm Delete dialog box, type `delete` and then click **Delete**.
   c. When the deletion process is complete, click **OK**.

   Ambari stops the Big Data Management service and deletes it from the listing of available services. To fully delete Big Data Management from the cluster, continue with the next steps.
4. In a command window, delete the `INFORMATICABDM` folder from the following directory on the name node of the cluster: `/var/lib/ambari-server/resources/stacks/<Hadoop distribution>/<Hadoop version>/services/`

5. Delete the `INFORMATICABDM` folder from the following location on all cluster nodes where it was installed: `/var/lib/ambari-agent/cache/stacks/<Hadoop distribution>/<Hadoop version>/services`
6. Perform the following steps to remove RPM binary files:
   a. Run the following command to determine the name of the RPM binary archive:
      ```
      rpm -qa |grep Informatica
      ```
   b. Run the following command to remove RPM binary files:
      ```
      rpm -ev <output_from_above_command>
      ```
      For example:
      ```
      rpm -ev InformaticaHadoop-10.1.1-1.x86_64
      ```
7. Repeat the previous step to remove RPM binary files from each cluster node.
8. Delete the following directory, if it exists, from the name node and each client node: `/opt/Informatica/`.
9. Repeat the last step on each cluster node where Big Data Management was installed.
10. On the name node, restart the Ambari server.

Part VI: Install and Configure for IBM BigInsights

This part contains the following chapters:

- Installation for IBM BigInsights, 136
- Hadoop Configuration Manager for IBM BigInsights, 140
- Manual Configuration Tasks for IBM BigInsights, 150
- Uninstall for IBM BigInsights, 159

CHAPTER 18

Installation for IBM BigInsights

This chapter includes the following topics:

- Installation for IBM BigInsights Overview, 136
- Download the Distribution Package, 136
- Install on a Single Node, 137
- Install on a Cluster Using SCP, 137
- Install on a Cluster Using NFS, 138
- Install on a Cluster Using the HadoopDataNodes File, 139

Installation for IBM BigInsights Overview

You can install Big Data Management for IBM BigInsights in different environments.

If you are installing Big Data Management for the first time or are upgrading, you need to install the Big Data Management binaries on the Hadoop environment.

Install Big Data Management on IBM BigInsights in one of the following ways:

- Install on a single node.
- Install on a cluster from the primary name node using SCP protocol.
- Install on a cluster from the primary name node using NFS protocol.
- Install on a cluster using the HadoopDataNodes file.

Download the Distribution Package

The tar.gz file that you download includes an RPM package and the binary files required to run the installation.

1. Create a temporary directory in a shared location on a local disk.
2. Download the following file: `InformaticaHadoop-<version>.<platform>-x64.tar.gz`
3. Extract the file to the machine from where you want to distribute the package and run the Big Data Management installation.

Install on a Single Node

You can install Big Data Management in a single node environment.

1. Log in to the machine as root user or as a user with sudo privileges.
2. Run one of the following commands to start the installation:

   ```
   bash InformaticaHadoopInstall.sh
   sh InformaticaHadoopInstall.sh
   ./InformaticaHadoopInstall.sh
   ```

3. Press **y** to accept the Big Data Management terms of agreement.
4. Press **Enter**.
5. Press **1** to install Big Data Management in a single node environment.
6. Press **Enter**.
7. Type the absolute path for the Big Data Management installation directory and press **Enter**.

   Start the path with a slash. The directory names in the path must not contain spaces or the following special characters: { } ! @ # $ % ^ & * ( ) : ; | ' ` < > , ? + [ ] \

   If you type a directory path that does not exist, the installer creates the entire directory path on the node during the installation. Default is `/opt`.

8. Press **Enter**.

   The installer creates the following directory and populates all of the file systems with the contents of the RPM package: `/<Big Data Management installation directory>/Informatica`

You can view the `informatica-hadoop-install.<date time stamp>.log` installation log file to get more information about the tasks performed by the installer.

Install on a Cluster Using SCP

You can install Big Data Management in a cluster environment from the primary namenode using the SCP protocol.

1. Log in to the primary NameNode as root user or as a user with sudo privileges.
2. Run one of the following commands to start the installation:

   ```
   bash InformaticaHadoopInstall.sh
   sh InformaticaHadoopInstall.sh
   ./InformaticaHadoopInstall.sh
   ```

3. Press **y** to accept the Big Data Management terms of agreement.
4. Press **Enter**.
5. Press **2** to install Big Data Management in a cluster environment.
6. Press **Enter**.
7. Type the absolute path for the Big Data Management installation directory.

   Start the path with a slash. The directory names in the path must not contain spaces or the following special characters: { } ! @ # $ % ^ & * ( ) : ; | ' ` < > , ? + [ ] \

   If you type a directory path that does not exist, the installer creates the entire directory path on each of the nodes during the installation. Default is `/opt/Informatica`.

8. Press **Enter**.
9. Press **1** to install Big Data Management from the primary name node.
10. Press **Enter**.
11. Type the absolute path for the Hadoop installation directory. Start the path with a slash.
12. Press **Enter**.
13. Type **y**.
14. Press **Enter**.

    The installer retrieves a list of DataNodes from the `$HADOOP_HOME/conf/slaves` file. On each DataNode, the installer creates the following directory and populates all of the file systems with the contents of the RPM package: `/<Big Data Management installation directory>/Informatica`

You can view the `informatica-hadoop-install.<date time stamp>.log` installation log file to get more information about the tasks performed by the installer.

Install on a Cluster Using NFS

You can install Big Data Management in a cluster environment from the primary NameNode using the NFS protocol.

1. Log in to the primary NameNode as root user or as a user with sudo privileges.
2. Run one of the following commands to start the installation:

```
bash InformaticaHadoopInstall.sh
sh InformaticaHadoopInstall.sh
./InformaticaHadoopInstall.sh
```

3. Press **y** to accept the Big Data Management terms of agreement.
4. Press **Enter**.
5. Press **2** to install Big Data Management in a cluster environment.
6. Press **Enter**.
7. Type the absolute path for the Big Data Management installation directory.

   Start the path with a slash. The directory names in the path must not contain spaces or the following special characters: { } ! @ # $ % ^ & * ( ) : ; | ' ` < > , ? + [ ] \

   If you type a directory path that does not exist, the installer creates the entire directory path on each of the nodes during the installation. Default is `/opt`.
8. Press **Enter**.
9. Press **1** to install Big Data Management from the primary name node.
10. Press **Enter**.
11. Type the absolute path for the Hadoop installation directory. Start the path with a slash.
12. Press **Enter**.
13. Type **n**.
14. Press **Enter**.
15. Type **y**.

16. Press **Enter**.

    The installer retrieves a list of DataNodes from the `$HADOOP_HOME/conf/slaves` file. On each DataNode, the installer creates the following directory and populates all of the file systems with the contents of the RPM package: `/<Big Data Management installation directory>/Informatica`

You can view the `informatica-hadoop-install.<date time stamp>.log` installation log file to get more information about the tasks performed by the installer.

Install on a Cluster Using the HadoopDataNodes File

You can install Big Data Management in a cluster environment using the HadoopDataNodes file.

1. Log in to the machine as the root user.
2. Update the `HadoopDataNodes` file that was extracted with the download package. Add an entry for each IP address or machine host name in the cluster.
3. Run one of the following commands to start the installation:

   ```
   bash InformaticaHadoopInstall.sh
   sh InformaticaHadoopInstall.sh
   ./InformaticaHadoopInstall.sh
   ```

4. Press **y** to accept the Big Data Management terms of agreement.
5. Press **Enter**.
6. Press **2** to install Big Data Management in a cluster environment.
7. Press **Enter**.
8. Type the absolute path for the Big Data Management installation directory and press **Enter**. Start the path with a slash. Default is `/opt`.
9. Press **Enter**.
10. Press **2** to install Big Data Management using the `HadoopDataNodes` file.
11. Press **Enter**.

    The installer creates the following directory on each node that appears in the `HadoopDataNodes` file:`/<Big Data Management installation directory>/Informatica`. The installer then populates all of the file systems with the contents of the RPM package.

You can view the `informatica-hadoop-install.<date time stamp>.log` installation log file to get more information about the tasks performed by the installer.

CHAPTER 19

Hadoop Configuration Manager for IBM BigInsights

This chapter includes the following topics:

- Hadoop Configuration Manager Overview, 140
- Start the Configuration Manager in Console Mode , 140
- Run the Configuration Manager in Silent Mode, 145

Hadoop Configuration Manager Overview

After you install Big Data Management on the Hadoop environment, you configure Big Data Management on the Informatica domain.

Configure Big Data Management using the Hadoop Configuration Manager (configuration manager). The configuration manager can create connections between the Hadoop environment and the Data Integration Service, updates Data Integration Service properties, and it updates configuration files on the Data Integration Service machine.

You can run the configuration manager in console mode or silent mode.

After you run the configuration manager, you must perform some additional configuration tasks.

Start the Configuration Manager in Console Mode

Run the Hadoop Configuration Manager in console mode to configure the Informatica domain for Big Data Management.

1. On the machine where the Data Integration Service runs, open the command line.
2. Go to the following directory: `<Informatica installation directory>/tools/BDMUtil`
3. Run `BDMConfig.sh`.
4. Choose option 4 to configure for IBM BigInsights.
5. In the **Distribution Folder Selection** section, choose the directory of the IBM BigInsights distribution that you want to configure.

Note: The distribution might be stored in a directory that uses a different distribution version number.

6. In the **Connection Type** section, select the option to access files on the Hadoop cluster.

Option	Description
1	Apache Ambari. Select this option to use the Ambari REST API to access files on the Hadoop cluster. Informatica recommends that you use this option.
2	Secure Shell (SSH). Select this option to use SSH to access files on the Hadoop cluster. This option requires SSH connections from the Informatica domain to the machines that host the NameNode, YARN ResourceManager, and Hive client. If you select this option, Informatica recommends that you use an SSH connection without a password or have sshpass or Expect installed.

Complete the configuration based on the option that you choose.

Complete the Configuration through Apache Ambari

To complete the configuration, you can choose to update Data Integration Service properties and create connection objects.

You might want to create connections if you are installing for the first time or if you are upgrading and the changes to the Hadoop environment are significant, such as a different distribution. If you are upgrading and you create connections that you want to use in existing mappings, you need to update mappings to use the new connections after you upgrade.

You might want to skip creating connections if the changes to the Hadoop environment are minimal, such as a change in distribution version. If you are upgrading and you choose not to create connections, you can continue to use the connections in existing mappings. However, you need update any changed properties after you upgrade. Property changes might include host names, URIs, or port numbers.

1. In the **Ambari Administration Information** section, enter the connection information to connect to the Ambari Manager.
   a. Enter the Ambari Manager host name or IP address and port.
   b. Enter the Ambari user ID.
   c. Enter the password for the user ID.
   d. Enter the port for Ambari Manager.
   e. Select whether to use MapReduce or Tez as the execution engine type.
      - 1 - MapReduce
      - 2 - Tez

   The Hadoop Configuration Manager retrieves the required information from the Hadoop cluster.
2. In the **Hadoop Configuration Manager Output** section, select whether you want to update Data Integration Service properties.

   Select from the following options:

Option	Description
1	No. Select this option to update Data Integration Service properties later.
2	Yes. Select this option to update Data Integration Service properties now.

3. Select whether you want to restart the Data Integration Service.

   Select from the following options:

Option	Description
1	No. Select this option if you do not want the configuration manager to restart the Data Integration Service.
2	Yes. Select this option if you want the configuration manager to restart the Data Integration Service when the configuration is complete.

4. Select whether you want to create connections for Big Data Management.

   Select from the following options:

Option	Description
1	No. Select this option if you do not want to create connections. If you are upgrading and you choose not to create connections, you can continue to use the connections in existing mappings. However, you need update any changed properties after you upgrade. Property changes might include host names, URIs, or port numbers.
2	Yes. Select this option if you want to create connections. If you are upgrading and you create connections that you want to use in existing mappings, you need to update mappings to use the new connections after you upgrade.

5. In the **Create Connections** section, select the connection type to create Big Data Management connections:

Option	Description
1. Hive	Create a Hive connection to access Hive as a source or target.
2. HDFS	Create an HDFS connection to read data from or write data to the HDFS file system on a Hadoop cluster.
3. Hadoop	Create a Hadoop connection to run mappings in the Hadoop environment.
4. HBase	Create an HBase connection to access HBase.
5. Select all	Create all four types of connection.

6. In the **Connection Details** section, provide the connection properties.

   Based on the type of connection that you choose to create, the Hadoop Configuration Manager requires different properties.

   The Hadoop Configuration Manager creates the connections.

7. The Hadoop Configuration Manager reports a summary of its operations, including whether connection creation succeeded and the location of log files.

The Hadoop Configuration Manager creates the following file in the `<Informatica installation directory>/tools/BDMUtil` directory:

ClusterConfig.properties.<timestamp>

Contains details about the properties fetched from the Hadoop cluster, including cluster node names, to provide templates for connection creation commands. To use these connection creation templates to

create connections to the Hadoop cluster, edit the connection name, domain username and password in the generated commands.

Complete the Configuration through SSH

When you complete configuration through SSH, you must provide host names and Hadoop configuration file locations.

Configuration through SSH requires SSH connections from the Informatica domain to the machines that host the NameNode, YARN ResourceManager, and Hive client. Informatica recommends that you use an SSH connection without a password or have sshpass or Expect installed. If you do not use one of these methods, you must enter the password each time the utility downloads a file from the Hadoop cluster.

1. Enter the name node host name.
2. Enter the SSH user ID.
3. Enter the password for the SSH user ID, or press enter if you use an SSH connection without a password.
4. Enter the location for the `hdfs-site.xml` file on the Hadoop cluster.
5. Enter the location for the `core-site.xml` file on the Hadoop cluster.

   The Hadoop Configuration Manger connects to the name node and downloads the following files: `hdfs-site.xml` and `core-site.xml`.
6. Enter the Yarn resource manager host name.

   Note: Yarn resource manager was formerly known as JobTracker.
7. Enter the SSH user ID.
8. Enter the password for the SSH user ID, or press enter if you use an SSH connection without a password.
9. Enter the directory for the `mapred-site.xml` file on the Hadoop cluster.
10. Enter the directory for the `yarn-site.xml` file on the Hadoop cluster.

    The utility connects to the JobTracker and downloads the following files: `mapred-site.xml` and `yarn-site.xml`.
11. Enter the Hive client host name.
12. Enter the SSH user ID.
13. Enter the password for the SSH user ID, or press enter if you use an SSH connection without a password.
14. Enter the directory for the `hive-site.xml` file on the Hadoop cluster.

    The configuration manager connects to the Hive client and downloads the following file: `hive-site.xml`.
15. Optionally, You can choose to configure the HBase server.

    Select the following options:

Option	Description
1	No. Select this option if you do not want to configure the HBase server.
2	Yes. Select this option to configure the HBase server.

If you select **Yes**, enter the following information to configure the HBase server:

a. Enter the HBase server host name.

b. Enter the SSH user ID.

c. Enter the password for the SSH user ID, or press enter if you use an SSH connection without a password.

d. Enter the directory for the `hbase-site.xml` file on the Hadoop cluster.

16. In the **Create Connections** section, select the connection type to create Big Data Management connections:

Option	Description
1. Hive	Create a Hive connection to access Hive as a source or target.
2. HDFS	Create an HDFS connection to read data from or write data to the HDFS file system on a Hadoop cluster.
3. Hadoop	Create a Hadoop connection to run mappings in the Hadoop environment.
4. HBase	Create an HBase connection to access HBase.
5. Select all	Create all four types of connection.

Press the number that corresponds to your choice.

17. The **Domain Information** section displays the domain name and the node name. Enter the following additional information about Informatica domain.

a. Enter the domain user name.

b. Enter the domain password.

c. Enter the Data Integration Service name.

d. If the Hadoop cluster uses Kerberos authentication, enter the following information:

- Hadoop Kerberos service principal name
- Hadoop Kerberos keytab location. Location of the keytab on the Data Integration Service machine.

Note: After you enter the Data Integration Service name, the utility tests the domain connection, and then recycles the Data Integration Service.

18. In the **Connection Details** section, provide the connection properties.

Based on the type of connection you choose to create, the utility requires different properties.

The Hadoop Configuration Manager creates the connections.

19. The Hadoop Configuration Manager reports a summary of its operations, including whether connection creation succeeded and the location of log files.

The Hadoop Configuration Manager creates the following file in the `<Informatica installation directory>/tools/BDMUtil` directory:

ClusterConfig.properties.<timestamp>

Contains details about the properties fetched from the Hadoop cluster, including cluster node names, to provide templates for connection creation commands. To use these connection creation templates to create connections to the Hadoop cluster, edit the connection name, domain username and password in the generated commands.

Run the Configuration Manager in Silent Mode

To configure the Big Data Management without user interaction in silent mode, run the Hadoop Configuration Manager in silent mode. Use the `SilentInput.properties` file to specify the configuration options. The configuration manager reads the file to determine the configuration options.

To configure the domain for Big Data Management in silent mode, perform the following tasks:

1. Configure the properties in the `SilentInput.properties` file.
2. Run the configuration manager with the `SilentInput.properties` file.

Configure the Properties File

The `SilentInput.properties` file includes the properties that are required by the configuration manger. Customize the properties file to specify the options for your configuration and then run the configuration manager in silent mode.

You might want to create connections if you are installing for the first time or if you are upgrading and the changes to the Hadoop environment are significant, such as a different distribution. If you are upgrading and you create connections that you want to use in existing mappings, you need to update mappings to use the new connections after you upgrade.

You might want to skip creating connections if the changes to the Hadoop environment are minimal, such as a change in distribution version. If you are upgrading and you choose not to create connections, you can continue to use the connections in existing mappings. However, you need update any changed properties after you upgrade. Property changes might include host names, URIs, or port numbers.

1. Find the sample `SilentInput.properties` file in the following location: `<Informatica installation directory>/tools/BDMUtil`
2. Create a backup copy of the `SilentInput.properties` file.
3. Use a text editor to open the file and modify the values of the configuration properties.
   The following table describes the configuration properties that you can modify:

Property Name	Description
CLOUDERA_SELECTION	Indicates whether to configure the Informatica domain for the Cloudera CDH cluster. To configure the domain for the Cloudera CDH cluster, set the value of this property to 1.
HORTONWORKS_SELECTION	Indicates whether to configure the Informatica domain for the Hortonworks cluster. To configure the domain for the Hortonworks cluster, set the value of this property to 1.
BIG_INSIGHT	Indicates whether to configure the Informatica domain for the BigInsights cluster. To configure the domain for the BigInsights cluster, set the value of this property to 1.
HD_INSIGHT	Indicates whether to configure the Informatica domain for the HDInsight cluster. To configure the domain for the HDInsight cluster, set the value of this property to 1.

Property Name	Description
DIST_FOLDER_NAME	Indicates the Hadoop distribution folder name present in INFA_HOME/services/shared/hadoop.
INSTALL_TYPE	Indicates how to access files on the Hadoop cluster. Set the value depending on the Hadoop distribution. To access files on the Cloudera CDH cluster using the Cloudera Manager, set the value of this property to 0. To access files on the Hortonworks HDP, IBM BigInsights, or the Azure HDInsight cluster using Apache Ambari, set the value of this property to 3.
CLOUDERA_HOSTNAME	Required if you set INSTALL_TYPE=0. Indicates the host name of the Cloudera Manager.
CLOUDERA_USER_NAME	Required if you set INSTALL_TYPE=0. Indicates the user id for the Cloudera Manager.
CLOUDERA_USER_PASSWD	Required if you set INSTALL_TYPE=0. Indicates the password for the Cloudera Manager.
CLOUDERA_PORT	Required if you set INSTALL_TYPE=0. Indicates the port of the Cloudera Manager.
CLOUDERA_SINGLECLUSTER_NAME	Required if you set INSTALL_TYPE=0. Indicates the name of the Cloudera CDH cluster for which you want to configure Big Data Management. If the Cloudera Manager manages a single cluster, you do not have to set this property. Retain this property as a comment or remove the property from the file. If the Cloudera Manager manages more than one cluster, uncomment and set the property to the name of the cluster for which you want to configure Big Data Management. Note: In a multi-cluster environment, the configuration fails if the property is commented or if the property has an invalid cluster name as the value.
AMBARI_HOSTNAME	Required if you set INSTALL_TYPE =3. Indicates the password for Apache Ambari.
AMBARI_USER_NAME	Required if you set INSTALL_TYPE=3. Indicates the port number for Apache Ambari.
AMBARI_USER_PASSWD	Required if you set INSTALL_TYPE=3. Indicates the password for Apache Ambari.
AMBARI_PORT	Required if you set INSTALL_TYPE=3. Indicates the port number for Apache Ambari.
TEZ_EXECUTION_MODE	Required if you set INSTALL_TYPE=3. Indicates whether to enable Tez for the Hive engine. To enable Tez on the Hortonworks HDP or Azure HDInsight cluster, set the value of this property to 1. Otherwise, set the value to 0.

Property Name	Description
CREATE_CONNECTION	Indicates whether to update the Data Integration Service and create connections for Big Data Management. To update the Data Integration Service and create connections, set the value of this property to 1 and define the connection properties. If you are upgrading and you choose not to create connections, you can continue to use the connections in existing mappings. However, you need update any changed properties after you upgrade. Property changes might include host names, URIs, or port numbers. If you are upgrading and you create connections that you want to use in existing mappings, you need to update mappings to use the new connections after you upgrade. To exit the configuration manager, set the value to 0.
UPDATE_DIS	Required if you set CREATE_CONNECTION=1. Indicates whether to update the Data Integration Service properties. To update the Data Integration Service, set this property to 1. Otherwise, set the value to 0.
CREATE_BDM_CONNECTIONS	Required if you set CREATE_CONNECTION=1. Set this property to 1 to create Hadoop, HDFS, HBase, and Hive connections. Otherwise, the configuration manager will not create connections.
DOMAIN_USER	Required if you set CREATE_CONNECTION=1. Indicates the user name for the domain administrator.
DOMAIN_PSSWD	Required if you set CREATE_CONNECTION=1. Indicates the password for the domain administrator.
DIS_SERVICE_NAME	Required if you set CREATE_CONNECTION=1 and UPDATE_DIS=1. Indicates the name of the Data Integration Service.
CLUSTER_INSTALLATION_DIR	Required if you set CREATE_CONNECTION=1 and UPDATE_DIS=1. Indicates the Informatica home directory on every node on the Hadoop cluster that is created during the Big Data Management installation. Default is /opt/Informatica.
KERBEROS_PRINCIPAL_NAME	Required if you set CREATE_CONNECTION=1 and UPDATE_DIS=1 and if the Hadoop cluster uses Kerberos authentication. Indicates the Hadoop Kerberos service principal name.
KERBEROS_KEYTAB	Required if you set CREATE_CONNECTION=1 and UPDATE_DIS=1 and if the Hadoop cluster uses Kerberos authentication. Indicates the Hadoop Kerberos keytab location.
HIVE_CONNECTION	Required if you set CREATE_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Set this property to 1 to create a Hive connection. Otherwise, set the value to 0.

Property Name	Description
HDFS_CONNECTION	Required if you set CREATE_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Set this property to 1 to create a HDFS connection. Otherwise, set the value to 0.
HADOOP_CONNECTION	Required if you set CREATE_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Set this property to 1 to create a Hadoop connection. You must create a Hadoop connection to run the mappings in the Hadoop environment.
HBASE_CONNECTION	Required if you set CREATE_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Set this property to 1 to create a HBASE connection. Otherwise, set the value to 0.
SELECT_ALL_CONNECTION	Required if you set CREATE_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Set this property to 1 to create all listed connections. Also set HIVE_CONNECTION=1, HDFS_CONNECTION=1, HADOOP_CONNECTION=1, and HBASE_CONNECTION=1 .
SAMPLE_HIVE_CONNECTION	Required if CREATE_CONNECTION=1, HIVE_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Indicates connection name for Hive.
HIVE_EXECUTION_MODE	Required if you set CREATE_CONNECTION=1, HIVE_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Indicates the execution mode for Hive. Set this property to either Local or Remote.
HIVE_USER_NAME	Required if you set CREATE_CONNECTION=1, HIVE_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Indicates the user name for Hive. You must set the execution mode to Remote.
SAMPLE_HDFS_CONNECTION	Required if you set CREATE_CONNECTION=1, HDFS_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Indicates the connection name for HDFS.
HDFS_USER_NAME	Required if you set CREATE_CONNECTION=1, HDFS_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Indicates the user name to access HDFS.
SAMPLE_HADOOP_CONNECTION	Required if you set CREATE_CONNECTION=1 , HADOOP_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Indicates the connection name for Hadoop.
BLAZE_WORKING_DIR	Required if you set CREATE_CONNECTION=1 , HADOOP_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Indicates the existing Blaze work directory on HDFS.

Property Name	Description
BLAZE_USER	Required if you set CREATE_CONNECTION=1, HADOOP_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Indicates the user name for Blaze.
SPARK_HDFS_STAGING_DIR	Required if you set CREATE_CONNECTION=1 , HADOOP_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Indicates the existing Spark HDFS staging directory.
SPARK_EVENTLOG_DIR	Required if you set CREATE_CONNECTION=1, HADOOP_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Optional: Indicates the Spark event log directory
SPARK_PARAMETER_LIST	Required if you set CREATE_CONNECTION=1, HADOOP_CONNECTION=1, and CREATE_BDM_CONNECTIONS=1. Indicates the Spark execution parameters list.
IMPERSONATION_USER	Required if you set CREATE_CONNECTION=1, HADOOP_CONNECTION=1, CREATE_BDM_CONNECTIONS=1, and if the Hadoop cluster uses Kerberos authentication.
SAMPLE_HBASE_CONNECTION	Required if you set CREATE_CONNECTION=1 , HBASE_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Indicates the connection name for HBase.
ZOOKEEPER_HOSTS	Required if you set CREATE_CONNECTION=1 , HBASE_CONNECTION=1 and CREATE_BDM_CONNECTIONS=1. Indicates the ZooKeeper host.

Run the Configuration Manager

After you configure the properties file, open a command prompt to start the silent configuration.

1. Go to the following directory: `<Informatica installation directory>/tools/BDMUtil`
2. Verify that the directory contains the `SilentInput.properties` file that you edited.
3. Run the `BDMSilentConfig.sh` file.

The installer runs in the background. The process can take a while. The silent configuration is complete when the following log files are created in the `<Informatica installation directory>/tools/BDMUtil` directory:

- Informatica_Big_Data_Edition_Configuration_Utility.log.<timestamp>
- ClusterConfig.properties.<timestamp>

The silent configuration fails if you incorrectly configure the properties file or if the installation directory is not accessible. The `silenterror.log` file is created in the home directory of the UNIX user that runs the configuration manager in silent mode. Correct the errors and then run the silent configuration again.

CHAPTER 20

Manual Configuration Tasks for IBM BigInsights

This chapter includes the following topics:

- Download the JDBC Drivers for Sqoop Connectivity, 150
- Update Files on the Hadoop Environment for IBM BigInsights, 150
- Update Files on the Domain Environment for IBM BigInsights, 152
- Update Files on the Developer Tool, 155
- Complete Upgrade Tasks, 156

Download the JDBC Drivers for Sqoop Connectivity

To configure Sqoop connectivity for relational databases, you must download JDBC driver jar files.

1. Download any Type 4 JDBC driver that the database vendor recommends for Sqoop connectivity.
2. Copy the jar files to the following directory on the machine where the Data Integration Service runs:
   `<Informatica installation directory>\externaljdbcjars`

At run time, the Data Integration Service copies the jar files to the Hadoop distribution cache so that the jar files are accessible to all nodes in the cluster.

Note: The DataDirect JDBC drivers that Informatica ships are not licensed for Sqoop connectivity.

Update Files on the Hadoop Environment for IBM BigInsights

You can configure additional functionality through the configuration files the Hadoop environment.

When you update the configuration files, update the files on every node in the cluster. You can find the configuration files in the `/etc/hadoop` directory of the Hadoop distribution. Changes to the files take effect after you restart the Hadoop cluster and services. Ensure that all dependent services start when you restart the cluster.

Update core-site.xml

Update core-site.xml on the Hadoop environment to configure functionality such as user impersonation for Sqoop mappings.

Configure Impersonation for Sqoop

To run Sqoop mappings on clusters that do not use Kerberos authentication, you must create a proxy user for the yarn user who will impersonate other users.

Configure the following properties, and then restart Hadoop services and the cluster:

hadoop.proxyuser.yarn.groups

Allows impersonation from any group.

```
<property>
    <name>hadoop.proxyuser.yarn.groups</name>
    <value><Name_of_the_impersonation_user></value>
    <description>Allows impersonation from any group.</description>
</property>
```

hadoop.proxyuser.yarn.hosts

Allows impersonation from any host.

```
<property>
    <name>hadoop.proxyuser.yarn.hosts</name>
    <value>*</value>
    <description>Allows impersonation from any host.</description>
</property>
```

Update yarn-site.xml

Update the yarn-site.xml file on the Hadoop environment to configure functionality such as virtual memory limits and dynamic resource allocation for the Spark engine.

Configure Dynamic Resource Allocation for the Spark Engine

Configure dynamic resource allocation for mappings to run on the Spark engine. Update the following properties on each node where YARN node manager is running:

yarn.nodemanager.aux-services

The auxiliary service name. Add spark_shuffle to the current entry. For example, if the current value is "mapreduce_shuffle," update it to be "mapreduce_shuffle,spark_shuffle."

```
<property>
    <name>yarn.nodemanager.aux-services</name>
    <value>mapreduce_shuffle,spark_shuffle</value>
</property>
```

yarn.nodemanager.aux-services.spark_shuffle.class

The auxiliary service class to use. Set to org.apache.spark.network.yarn.YarnShuffleService. For example,

```
<property>
    <name>yarn.nodemanager.aux-services.spark_shuffle.class</name>
    <value>org.apache.spark.network.yarn.YarnShuffleService</value>
</property>
```

yarn.application.classpath

The classpath for YARN applications. Verify that the yarn classpath includes the path to the Spark shuffle.jar file packaged with the distribution. If the .jar file is not in any Hadoop distribution directory, update the path to use the .jar file in the Apache Spark distribution directory. For example,

```
<property>
    <name>yarn.application.classpath</name>
    <value>/opt/Informatica/services/shared/spark/lib_spark_2.0.1_hadoop_2.6.0/spark-
networkshuffle_2.11-2.0.1.jar</value>
</property>
```

Configure Virtual Memory Limits for the Blaze Engine

Configure the following property to remove virtual memory limits for every node in the Hadoop cluster:

yarn.nodemanager.vmem-check-enabled

Determines virtual memory limits. Set the value to false to disable virtual memory limits. For example,

```
<property>
    <name>yarn.nodemanager.vmem-check-enabled</name>
    <value>false</value>
    <description>Enforces virtual memory limits for containers.</description>
</property>
```

Allocate Cluster Resources for the Blaze Engine

Update the following properties to verify that the cluster allocates sufficient memory and resources for the Blaze engine:

yarn.nodemanager.resource.memory-mb

The maximum RAM available for each container. Set the maximum memory on the cluster to increase resource memory available to the Blaze engine. Configure this property on the following nodes:

- Run-time nodes. Set to at least 10 GB.
- Management node. Set to at least 13 GB.

yarn.nodemanager.resource.cpu-vcores

The number of virtual cores for each container. The number might correspond to the number of physical cores, but you can increase the value to allow for more processing. Configure this property on the following nodes:

- Run-time nodes. Set to at 6.
- Management node. Set to 9.

yarn.scheduler.minimum-allocation-mb

The minimum RAM available for each container. Set the minimum memory to allow the VM to spawn sufficient containers. Configure this property to be no less than 4 GB of the maximum memory on the run-time and management nodes.

Update Files on the Domain Environment for IBM BigInsights

You can configure additional functionality through the configuration files on the domain.

When the Informatica installer creates nodes on the domain, it creates directories and files for Big Data Management. When you configure Big Data Management on the domain, you need to edit configuration files to enable functionality.

The following table describes the installation directories and the default paths:

Directory	Description
Data Integration Service Hadoop distribution directory	The Hadoop distribution directory on the Data Integration Service node. It corresponds to the distribution on the Hadoop environment. You can view the directory path in the Data Integration Service properties of the Administrator tool. Default is `<Informatica installation directory>/Informatica/services/shared/hadoop/<Hadoop distribution name>_<version>`.
Configuration file directories	Configuration files for Big Data Management are stored in the following directories: - `<Informatica installation directory>/Informatica/services/shared/hadoop/<Hadoop distribution name>_<version>/InfaConf`. Contains the hadoopEnv.properties file. - `<Informatica installation directory>/Informatica/services/shared/hadoop/<Hadoop distribution name>_<version>/conf`. Contains *-site files.

Update hadoopEnv.properties

Update the hadoopEnv.properties file to configure functionality such as Sqoop connectivity and the Spark run-time engine.

Open hadoopEnv.properties and back it up before you configure it. You can find the hadoopEnv.properties file in the following location: `<Informatica installation directory>/services/shared/hadoop/<Hadoop distribution name>_<version number>/infaConf`

Configure Performance for the Spark Engine

Configure the following performance properties for the Spark engine:

spark.dynamicAllocation.enabled

Performance configuration to run mappings on the Spark engine. Enables dynamic resource allocation. Required when you enable the external shuffle service.

Set the value to TRUE.

spark.shuffle.service.enabled

Performance configuration to run mappings on the Spark engine. Enables the external shuffle service. Required when you enable dynamic resource allocation.

Set the value to TRUE.

spark.scheduler.maxRegisteredResourcesWaitingTime

Performance configuration to run mappings on the Spark engine. The number of milliseconds to wait for resources to register before scheduling a task. Reduce this from the default value of 30000 to reduce delays before starting the Spark job execution.

Set the value to 15000.

spark.scheduler.minRegisteredResourcesRatio

Performance configuration to run mappings on the Spark engine. The minimum ratio of registered resources to acquire before task scheduling begins. Reduce this from the default value of 0.8 to reduce any delay before starting the Spark job execution.

Set the value to .5.

spark.executor.instances

Performance configuration to run mappings on the Spark engine. If you enable dynamic resource allocation for the Spark engine, Informatica recommends that you convert this property to a comment. For example, `#spark.executor.instances=100`

Configure Sqoop Connectivity

Configure the following property for Sqoop connectivity:

infapdo.env.entry.hadoop_node_jdk_home

Configure the HADOOP_NODE_JDK_HOME to represent the JDK version that the cluster nodes use. You must use JDK version 1.7 or later.

Configure the property as follows:

```
infapdo.env.entry.hadoop_node_jdk_home=HADOOP_NODE_JDK_HOME=<cluster JDK home>/
jdk<version>
```

For example,

```
infapdo.env.entry.hadoop_node_jdk_home=HADOOP_NODE_JDK_HOME=/usr/java/default
```

Configure Environment Variables

You can optionally add third-party environment variables and extend the existing PATH environment variable in the hadoopEnv.properties file. The following text shows sample entries to configure environment variables:

```
infapdo.env.entry.oracle_home=ORACLE_HOME=/databases/oracle
infapdo.env.entry.db2_home=DB2_HOME=/databases/db2
infapdo.env.entry.db2instance=DB2INSTANCE=OCA_DB2INSTANCE
infapdo.env.entry.db2codepage=DB2CODEPAGE="1208"
infapdo.env.entry.odbchome=ODBCHOME=$HADOOP_NODE_INFA_HOME/ODBC7.1
infapdo.env.entry.home=HOME=/opt/thirdparty
infapdo.env.entry.gphome_loaders=GPHOME_LOADERS=/databases/greenplum
infapdo.env.entry.pythonpath=PYTHONPATH=$GPHOME_LOADERS/bin/ext
infapdo.env.entry.nz_home=NZ_HOME=/databases/netezza
infapdo.env.entry.ld_library_path=LD_LIBRARY_PATH=$HADOOP_NODE_INFA_HOME/services/
shared/bin:$HADOOP_NODE_INFA_HOME/DataTransformation/bin:$HADOOP_NODE_HADOOP_DIST/lib/
native:$HADOOP_NODE_INFA_HOME/ODBC7.1/lib:$HADOOP_NODE_INFA_HOME/jre/lib/
amd64:$HADOOP_NODE_INFA_HOME/jre/lib/amd64/server:$HADOOP_NODE_INFA_HOME/java/jre/lib/
amd64:$HADOOP_NODE_INFA_HOME/java/jre/lib/amd64/server:/databases/oracle/lib:/
databases/db2/lib64:$LD_LIBRARY_PATH
infapdo.env.entry.path=PATH=$HADOOP_NODE_HADOOP_DIST/scripts:$HADOOP_NODE_INFA_HOME/
services/shared/bin:$HADOOP_NODE_INFA_HOME/jre/bin:$HADOOP_NODE_INFA_HOME/java/jre/bin:
$HADOOP_NODE_INFA_HOME/ODBC7.1/bin:/databases/oracle/bin:/databases/db2/bin:$PATH
#teradata
infapdo.env.entry.twb_root=TWB_ROOT=/databases/teradata/tbuild
infapdo.env.entry.manpath=MANPATH=/databases/teradata/odbc_64:/databases/teradata/odbc_64
infapdo.env.entry.nlspath=NLSPATH=/databases/teradata/odbc_64/msg/%N:/databases/
teradata/msg/%N
infapdo.env.entry.pwd=PWD=/databases/teradata/odbc_64/samples/C
```

Update hive-site.xml

Update the hive-site.xml file on the domain environment to set properties for dynamic partitioning.

Configure Dynamic Partitioning

To use Hive dynamic partitioned tables, configure the following properties:

hive.exec.dynamic.partition

Enables dynamic partitioned tables. Set this value to `TRUE`.

exec.dynamic.partition.mode

Allows all partitions to be dynamic. Set this value to `nonstrict`.

Update yarn-site.xml

Update the yarn-site.xml file on the domain to access Hive tables in Amazon S3 buckets.

Configure Access to Hive Tables in Amazon S3 Buckets

Configure the AWS access key to run mappings with sources and targets on Hive tables in Amazon S3 buckets.

Note: To use a Hive table as a target on Amazon S3, grant write permission to the bucket through bucket policies, or add these properties to the configuration file. You must add these properties to the core-site.xml file on the Hadoop environment and to the yarn-site.xml file on the domain environment.

Configure the following properties:

fs.s3a.access.key

The ID for the Blaze and Spark engines to connect to the Amazon S3a file system. For example,

```
<property>
    <name>fs.s3a.access.key</name>
    <value>[Your Access Key]</value>
 </property>
```

fs.s3a.secret.key

The password for the Blaze and Spark engines to connect to the Amazon S3a file system.

```
<property>
    <name>fs.s3a.secret.key</name>
    <value>[Your Access Id]</value>
</property>
```

When you use a server side encryption protocol for Hive buckets, configure properties to enable access to encrypted Hive buckets.

Configure the following properties:

fs.s3.enableServerSideEncryption

Set to TRUE to enable server side encryption. For example,

```
<property>
    <name>fs.s3.enableServerSideEncryption</name>
    <value>TRUE</value>
 </property>
```

fs.s3a.server-side-encryption-algorithm

The encryption algorithm that you use to encrypt Hive buckets. For example,

```
<property>
    <name>fs.s3a.server-side-encryption-algorithm</name>
    <value>AES256</value>
 </property>
```

Update Files on the Developer Tool

Update developerCore.ini on the machine that hosts the Developer tool.

Before you perform configuration on the Developer tool, you need to copy some configuration files to the machine that hosts the Developer tool.

1. Find hdfs-site.xml and core-site.xml on the name node of the cluster.

2. Copy the files to the following location on the machine that hosts the Developer tool:`<Informatica installation directory>/clients/DeveloperClient/Hadoop/<Hadoop distribution><version>/conf`

developerCore.ini

Edit developerCore.ini to enable communication between the Developer tool and the Hadoop cluster. You can find developerCore.ini in the following directory: `<Informatica installation directory>\clients\DeveloperClient`

Add the following property:

```
-DINFA_HADOOP_DIST_DIR=hadoop\<distribution><version>
```

The change takes effect when you restart the Developer tool.

Complete Upgrade Tasks

If you upgraded the Informatica platform or applied the hotfix, you need to perform some additional tasks within the Informatica domain.

Based on the version that you upgraded from, perform the following tasks:

Update the hadoopEnv.properties file.

Applies to all upgrades. The 10.1.1 HotFix 1 hadoopEnv.properties file contains additional properties. You need to manually update it to include customized configuration from previous versions.

Complete connection upgrades.

Applies to upgrades from 9.6.1 or any 9.6.1 hotfix. You need to generate Hadoop connections from the Hive connection and update parameter files.

Update connection properties.

Applies to any upgrade. If you did not create connections when you ran the Hadoop Configuration Manager, you must update values for connection properties that changed.

Replace connections.

Applies to any upgrade. If you created connections when you ran the Hadoop Configuration Manager, you need to replace connections in mappings with the new connections.

Update the hadoopEnv.properties File

When you run the Informatica upgrade or apply the hotfix on a machine that hosts the Data Integration Service, the installer creates a new hadoopEnv.properties file and backs up the existing configuration file.

You need to edit the version 10.1.1 HotFix 1 hadoopEnv.properties file to include any manual configuration that you performed in the previous version for the corresponding distribution.

If you applied the hotfix, you can find the backup hadoopEnv.properties file in the following location:

```
<Informatica installation directory>/BACK_FROM_HF1/services/shared/hadoop/<Hadoop distribution name>_<version>/infaConf
```

If you upgraded from a previous version, you can find the backup hadoopEnv.properties file in the following location:

```
<Previous Informatica installation directory>/services/shared/hadoop/<Hadoop distribution name>_<version>/infaConf
```

You can find the 10.1.1 HotFix 1 hadoopEnv.properties file in the following location:

```
<Informatica installation directory>/services/shared/hadoop/<Hadoop distribution name>_<version>/infaConf
```

Complete Connection Upgrade

Effective in version 10.0, Big Data Management requires a Hadoop connection to run mappings on the Hadoop cluster. If you upgraded from 9.6.1 or any 9.6.1 hotfix, you must generate Hadoop connections from Hive connections that are enabled to run mappings in the Hadoop environment.

The upgrade process generates a connection name for the Hadoop connection and replaces the connection name in the mappings. It does not create the physical connection object. When the upgrade is complete, you must run a command to generate the connection. Generate Hadoop connections from Hive connections if the following conditions are true:

- You upgraded from 9.6.1 or any 9.6.1 hotfix.
- The Hive connections are configured to run mappings in the Hadoop environment.

Complete the following tasks to upgrade connections:

1. Run `infacmd isp generateHadoopConnectionFromHiveConnection` to generate a Hadoop connection from a Hive connection that is configured to run in the Hadoop environment.
   The command renames the connection as follows: "Autogen_<Hive connection name>." If the renamed connection exceeds the 128 character limit, the command fails.
2. If the command fails, complete the following tasks:
   a. Rename the connection to meet the character limit and run the command again.
   b. Run `infacmd dis replaceMappingHadoopRuntimeConnections` to replace connections associated with mappings that are deployed in applications.
   c. Run `infacmd mrs replaceMappingHadoopRuntimeConnections` to replace connections associated with mappings that you run from the Developer tool.
3. If the Hive connection was parameterized, you must update the connection names in the parameter file. Verify that the Hive sources, Hive targets, and the Hive engine parameters are updated with the correct connection name.
4. If any properties changed in the cluster, such as host names, URIs, or port numbers, you must update the properties in the connections.

For information about the infacmd commands, see the *Informatica Command Reference*.

Update Connection Properties

If you upgraded from version 10.0 or later, and you did not create connections when you ran the Hadoop Configuration Manager, you need to verify connection properties.

Review the Hadoop, Hive, HDFS, and HBase connections that you created in a previous release to update the values for connection properties. For example, if you added nodes to the cluster or if you updated the distribution version, you might need to verify host names, URIs, or port numbers for some of the properties.

Replace the Connections

If you created connections when you ran the Hadoop Configuration Manager, you need to replace connections in mappings with the new connections.

The method that you use to replace connections in mappings depends on the type of connection.

Hadoop connection

If you created a Hadoop connection when you ran the Hadoop Configuration Manager, run the following commands:

- `infacmd dis replaceMappingHadoopRuntimeConnections`. Replaces connections associated with mappings that are deployed in applications.
- `infacmd mrs replaceMappingHadoopRuntimeConnections`. Replaces connections associated with mappings that you run from the Developer tool.

For information about the infacmd commands, see the *Informatica Command Reference*.

Hive, HDFS, or HBase connection

If you created a Hive, HDFS, or HBase connection when you ran the Hadoop Configuration Manager, you must replace the connections manually.

CHAPTER 21

Uninstall for IBM BigInsights

This chapter includes the following topics:

- Before You Uninstall, 159
- Uninstall Big Data Management, 159

Before You Uninstall

Verify prerequisites before you uninstall Big Data Management.

1. Verify that the Big Data Management administrator can run `sudo` commands.
2. If you are uninstalling Big Data Management in a cluster environment, configure the root user to use a passwordless Secure Shell (SSH) connection between the machine where you want to run the Big Data Management uninstall and all of the nodes where Big Data Management is installed.
3. If you are uninstalling Big Data Management in a cluster environment using the `HadoopDataNodes` file, verify that the `HadoopDataNodes` file contains the IP addresses or machine host names of each of the nodes in the Hadoop cluster from which you want to uninstall Big Data Management. The `HadoopDataNodes` file is located on the node from where you want to launch the Big Data Management installation. You must add one IP address or machine host name of the nodes in the Hadoop cluster for each line in the file.

Uninstall Big Data Management

You can uninstall Big Data Management from a single node or cluster environment.

1. Log in to the machine as root user. The machine you log in to depends on the Big Data Management environment and uninstallation method.
   - To uninstall in a single node environment, log in to the machine on which Big Data Management is installed.
   - To uninstall in a cluster environment using the HADOOP_HOME environment variable, log in to the primary name node.
   - To uninstall in a cluster environment using the `HadoopDataNodes` file, log in to any node.

2. Run the following command to start the uninstallation in console mode:

```
bash InformaticaHadoopInstall.sh
sh InformaticaHadoopInstall.sh
./InformaticaHadoopInstall.sh
```

3. Press **y** to accept the Big Data Management terms of agreement.
4. Press **Enter**.
5. Select **3** to uninstall Big Data Management.
6. Press **Enter**.
7. Select the uninstallation option, depending on the Big Data Management environment:
   - Select **1** to uninstall Big Data Management from a single node environment.
   - Select **2** to uninstall Big Data Management from a cluster environment.
8. Press **Enter**.
9. If you are uninstalling Big Data Management in a cluster environment, select the uninstallation option, depending on the uninstallation method:
   - Select **1** to uninstall Big Data Management from the primary name node.
   - Select **2** to uninstall Big Data Management using the `HadoopDataNodes` file.
10. Press **Enter**.
11. If you are uninstalling Big Data Management from a cluster environment from the primary name node, type the absolute path for the Hadoop installation directory. Start the path with a slash.

The uninstaller deletes all of the Big Data Management binary files from the following directory: `/<Big Data Management installation directory>/Informatica`
In a cluster environment, the uninstaller delete the binary files from all of the nodes within the Hadoop cluster.

Part VII: Install and Configure for MapR

This part contains the following chapters:

- Installation for MapR, 162
- Hadoop Configuration Manager for MapR, 166
- Manual Configuration Tasks for MapR, 169
- Uninstall for MapR, 184

CHAPTER 22

Installation for MapR

This chapter includes the following topics:

- Installation for MapR Overview, 162
- Download the Distribution Package, 162
- Install on a Single Node, 163
- Install on a Cluster Using SCP, 163
- Install on a Cluster Using NFS, 164
- Install on a Cluster Using the HadoopDataNodes File, 165

Installation for MapR Overview

You can install Big Data Management for MapR in different environments.

If you are installing Big Data Management for the first time or are upgrading, you need to install the Big Data Management binaries on the Hadoop environment.

Install Big Data Management on MapR in one of the following ways:

- Install on a single node.
- Install on a cluster from the primary name node using SCP protocol.
- Install on a cluster from the primary name node using NFS protocol.
- Install on a cluster from a non-name node machine.
- Install on a cluster using the HadoopDataNodes file.

Download the Distribution Package

The tar.gz file that you download includes an RPM package and the binary files required to run the installation.

1. Create a temporary directory in a shared location on a local disk.
2. Download the following file: `InformaticaHadoop-<version>.<platform>-x64.tar.gz`
3. Extract the file to the machine from where you want to distribute the package and run the Big Data Management installation.

Install on a Single Node

You can install Big Data Management in a single node environment.

1. Log in to the machine as root user or as a user with sudo privileges.
2. Run one of the following commands to start the installation:

   ```
   bash InformaticaHadoopInstall.sh
   sh InformaticaHadoopInstall.sh
   ./InformaticaHadoopInstall.sh
   ```

3. Press **y** to accept the Big Data Management terms of agreement.
4. Press **Enter**.
5. Press **1** to install Big Data Management in a single node environment.
6. Press **Enter**.
7. Type the absolute path for the Big Data Management installation directory and press **Enter**.

   Start the path with a slash. The directory names in the path must not contain spaces or the following special characters: { } ! @ # $ % ^ & * ( ) : ; | ' ` < > , ? + [ ] \

   If you type a directory path that does not exist, the installer creates the entire directory path on the node during the installation. Default is `/opt`.

8. Press **Enter**.

   The installer creates the following directory and populates all of the file systems with the contents of the RPM package: `/<Big Data Management installation directory>/Informatica`

You can view the `informatica-hadoop-install.<date time stamp>.log` installation log file to get more information about the tasks performed by the installer.

Install on a Cluster Using SCP

You can install Big Data Management in a cluster environment from the primary namenode using the SCP protocol.

1. Log in to the primary NameNode as root user or as a user with sudo privileges.
2. Run one of the following commands to start the installation:

   ```
   bash InformaticaHadoopInstall.sh
   sh InformaticaHadoopInstall.sh
   ./InformaticaHadoopInstall.sh
   ```

3. Press **y** to accept the Big Data Management terms of agreement.
4. Press **Enter**.
5. Press **2** to install Big Data Management in a cluster environment.
6. Press **Enter**.
7. Type the absolute path for the Big Data Management installation directory.

   Start the path with a slash. The directory names in the path must not contain spaces or the following special characters: { } ! @ # $ % ^ & * ( ) : ; | ' ` < > , ? + [ ] \

   If you type a directory path that does not exist, the installer creates the entire directory path on each of the nodes during the installation. Default is `/opt/Informatica`.

8. Press **Enter**.
9. Press **1** to install Big Data Management from the primary name node.
10. Press **Enter**.
11. Type the absolute path for the Hadoop installation directory. Start the path with a slash.
12. Press **Enter**.
13. Type **y**.
14. Press **Enter**.

    The installer retrieves a list of DataNodes from the `$HADOOP_HOME/conf/slaves` file. On each DataNode, the installer creates the following directory and populates all of the file systems with the contents of the RPM package: `/<Big Data Management installation directory>/Informatica`

You can view the `informatica-hadoop-install.<date time stamp>.log` installation log file to get more information about the tasks performed by the installer.

Install on a Cluster Using NFS

You can install Big Data Management in a cluster environment from the primary NameNode using the NFS protocol.

1. Log in to the primary NameNode as root user or as a user with sudo privileges.
2. Run one of the following commands to start the installation:

   ```
   bash InformaticaHadoopInstall.sh
   sh InformaticaHadoopInstall.sh
   ./InformaticaHadoopInstall.sh
   ```

3. Press **y** to accept the Big Data Management terms of agreement.
4. Press **Enter**.
5. Press **2** to install Big Data Management in a cluster environment.
6. Press **Enter**.
7. Type the absolute path for the Big Data Management installation directory.

   Start the path with a slash. The directory names in the path must not contain spaces or the following special characters: { } ! @ # $ % ^ & * ( ) : ; | ' ` < > , ? + [ ] \

   If you type a directory path that does not exist, the installer creates the entire directory path on each of the nodes during the installation. Default is `/opt`.
8. Press **Enter**.
9. Press **1** to install Big Data Management from the primary name node.
10. Press **Enter**.
11. Type the absolute path for the Hadoop installation directory. Start the path with a slash.
12. Press **Enter**.
13. Type **n**.
14. Press **Enter**.
15. Type **y**.

16. Press **Enter**.

    The installer retrieves a list of DataNodes from the `$HADOOP_HOME/conf/slaves` file. On each DataNode, the installer creates the following directory and populates all of the file systems with the contents of the RPM package: `/<Big Data Management installation directory>/Informatica`

You can view the `informatica-hadoop-install.<date time stamp>.log` installation log file to get more information about the tasks performed by the installer.

Install on a Cluster Using the HadoopDataNodes File

You can install Big Data Management in a cluster environment using the HadoopDataNodes file.

1. Log in to the machine as the root user.
2. Update the `HadoopDataNodes` file that was extracted with the download package. Add an entry for each IP address or machine host name in the cluster.
3. Run one of the following commands to start the installation:

   ```
   bash InformaticaHadoopInstall.sh
   sh InformaticaHadoopInstall.sh
   ./InformaticaHadoopInstall.sh
   ```
4. Press **y** to accept the Big Data Management terms of agreement.
5. Press **Enter**.
6. Press **2** to install Big Data Management in a cluster environment.
7. Press **Enter**.
8. Type the absolute path for the Big Data Management installation directory and press **Enter**. Start the path with a slash. Default is `/opt`.
9. Press **Enter**.
10. Press **2** to install Big Data Management using the `HadoopDataNodes` file.
11. Press **Enter**.

    The installer creates the following directory on each node that appears in the `HadoopDataNodes` file:`/<Big Data Management installation directory>/Informatica`. The installer then populates all of the file systems with the contents of the RPM package.

You can view the `informatica-hadoop-install.<date time stamp>.log` installation log file to get more information about the tasks performed by the installer.

CHAPTER 23

Hadoop Configuration Manager for MapR

This chapter includes the following topics:

- Hadoop Configuration Manager Overview, 166
- Run the Configuration Manager in Console Mode, 166

Hadoop Configuration Manager Overview

After you install Big Data Management on the Hadoop environment, you configure Big Data Management on the Informatica domain.

Configure Big Data Management using the Hadoop Configuration Manager (configuration manager). The configuration manager creates connections between the Hadoop environment and the Data Integration Service and it updates Data Integration Service properties.

After you run the configuration manager, you must perform some additional configuration tasks.

Run the Configuration Manager in Console Mode

Run the Hadoop Configuration Manager in console mode to configure the Informatica domain for Big Data Management.

1. On the machine where the Data Integration Service runs, open the command line.
2. Go to the following directory: `<Informatica installation directory>/tools/BDMUtil`
3. Run `BDMConfig.sh`.
4. Choose option 3 to configure for MapR.
5. In the **Distribution Folder Selection** section, choose the directory of the MapR distribution that you want to configure.

   Note: The distribution might be stored in a directory that uses a different distribution version number.
6. In the **Connection Type** section, select 1 to configure MapR through SSH.

Configuration through SSH requires SSH connections from the Informatica domain to the machines that host the NameNode, YARN ResourceManager, and Hive client. Informatica recommends that you use an SSH connection without a password or have sshpass or Expect installed. If you do not use one of these methods, you must enter the password each time the utility downloads a file from the Hadoop cluster.

7. For the NameNode host name field, enter the MapR Resource Manager host name.
8. Enter the SSH user ID.
9. Enter the password for the SSH user ID, or press enter if you use an SSH connection without a password.
10. Enter the location for the `core-site.xml` file on the Hadoop cluster.
11. Enter the location for the `mapr-clusters.conf` file on the Hadoop cluster.

    The Hadoop Configuration Manger connects to the name node and downloads the following files: `core-site.xml` and `mapr-clusters.conf`.
12. Enter the YARN Resource Manager host name.

    Note: YARN Resource Manager was formerly known as JobTracker.
13. Enter the SSH user ID.
14. Enter the password for the SSH user ID, or press enter if you use an SSH connection without a password.
15. Enter the directory for the `mapred-site.xml` file on the Hadoop cluster.
16. Enter the directory for the `yarn-site.xml` file on the Hadoop cluster.

    The utility connects to the JobTracker and downloads the following files: `mapred-site.xml` and `yarn-site.xml`.
17. Enter the Hive client host name.
18. Enter the SSH user ID.
19. Enter the password for the SSH user ID, or press enter if you use an SSH connection without a password.
20. Enter the directory for the `hive-site.xml` file on the Hadoop cluster.

    The configuration manager connects to the Hive client and downloads the following file: `hive-site.xml`.
21. Optionally, You can choose to configure the HBase server.

    Select the following options:

Option	Description
1	No. Select this option if you do not want to configure the HBase server.
2	Yes. Select this option to configure the HBase server.

If you select **Yes**, enter the following information to configure the HBase server:

a. Enter the HBase server host name.
b. Enter the SSH user ID.
c. Enter the password for the SSH user ID, or press enter if you use an SSH connection without a password.
d. Enter the directory for the `hbase-site.xml` file on the Hadoop cluster.

22. In the **Create Connections** section, select the connection type to create Big Data Management connections:

Option	Description
1. Hive	Create a Hive connection to access Hive as a source or target.
2. HDFS	Create an HDFS connection to read data from or write data to the HDFS file system on a Hadoop cluster.
3. Hadoop	Create a Hadoop connection to run mappings in the Hadoop environment.
4. HBase	Create an HBase connection to access HBase.
5. Select all	Create all four types of connection.

Press the number that corresponds to your choice.

23. The **Domain Information** section displays the domain name and the node name. Enter the following additional information about Informatica domain.
    a. Enter the domain user name.
    b. Enter the domain password.
    c. Enter the Data Integration Service name.
    d. If the Hadoop cluster uses Kerberos authentication, enter the following information:
        - Hadoop Kerberos service principal name
        - Hadoop Kerberos keytab location. Location of the keytab on the Data Integration Service machine.

    Note: After you enter the Data Integration Service name, the utility tests the domain connection, and then recycles the Data Integration Service.

24. In the **Connection Details** section, provide the connection properties.

    Based on the type of connection you choose to create, the utility requires different properties.

    The Hadoop Configuration Manager creates the connections.

25. The Hadoop Configuration Manager reports a summary of its operations, including whether connection creation succeeded and the location of log files.

The Hadoop Configuration Manager creates the following file in the `<Informatica installation directory>/tools/BDMUtil` directory:

ClusterConfig.properties.<timestamp>

Contains details about the properties fetched from the Hadoop cluster, including cluster node names, to provide templates for connection creation commands. To use these connection creation templates to create connections to the Hadoop cluster, edit the connection name, domain username and password in the generated commands.

CHAPTER 24

Manual Configuration Tasks for MapR

This chapter includes the following topics:

- Download the JDBC Drivers for Sqoop Connectivity, 169
- Update Configuration Files on the Domain Environment, 169
- Update Files on the Hadoop Environment for MapR, 173
- Generate MapR Tickets, 175
- Developer Tool Tasks, 179
- Complete Upgrade Tasks, 181

Download the JDBC Drivers for Sqoop Connectivity

To configure Sqoop connectivity for relational databases, you must download JDBC driver jar files.

1. Download any Type 4 JDBC driver that the database vendor recommends for Sqoop connectivity.
2. Copy the jar files to the following directory on the machine where the Data Integration Service runs:
   `<Informatica installation directory>\externaljdbcjars`

At run time, the Data Integration Service copies the jar files to the Hadoop distribution cache so that the jar files are accessible to all nodes in the cluster.

Note: The DataDirect JDBC drivers that Informatica ships are not licensed for Sqoop connectivity.

Update Configuration Files on the Domain Environment

You can configure additional functionality through the configuration files on the domain.

When the Informatica installer creates nodes on the domain, it creates directories and files for Big Data Management. When you configure Big Data Management on the domain, you need to edit configuration files to enable functionality.

The following table describes the installation directories and the default paths:

Directory	Description
Data Integration Service Hadoop distribution directory	The Hadoop distribution directory on the Data Integration Service node. It corresponds to the distribution on the Hadoop environment. You can view the directory path in the Data Integration Service properties of the Administrator tool. Default is `<Informatica installation directory>/Informatica/services/shared/hadoop/<Hadoop distribution name>_<version>`.
Configuration file directories	Configuration files for Big Data Management are stored in the following directories: - `<Informatica installation directory>/Informatica/services/shared/hadoop/<Hadoop distribution name>_<version>/InfaConf`. Contains the hadoopEnv.properties file. - `<Informatica installation directory>/Informatica/services/shared/hadoop/<Hadoop distribution name>_<version>/conf`. Contains *-site files.

Update hadoopEnv.properties

Update the hadoopEnv.properties file to configure functionality such as Sqoop connectivity and the Spark run-time engine.

Open hadoopEnv.properties and back it up before you configure it. You can find the hadoopEnv.properties file in the following location:`<Data Integration Service Hadoop distribution directory>/infaConf`

Configure Performance for the Spark Engine

Configure the following performance properties for the Spark engine:

spark.dynamicAllocation.enabled

Performance configuration to run mappings on the Spark engine. Enables dynamic resource allocation. Required when you enable the external shuffle service.

Set the value to TRUE.

spark.shuffle.service.enabled

Performance configuration to run mappings on the Spark engine. Enables the external shuffle service. Required when you enable dynamic resource allocation.

Set the value to TRUE.

spark.scheduler.maxRegisteredResourcesWaitingTime

Performance configuration to run mappings on the Spark engine. The number of milliseconds to wait for resources to register before scheduling a task. Reduce this from the default value of 30000 to reduce delays before starting the Spark job execution.

Set the value to 15000.

spark.scheduler.minRegisteredResourcesRatio

Performance configuration to run mappings on the Spark engine. The minimum ratio of registered resources to acquire before task scheduling begins. Reduce this from the default value of 0.8 to reduce any delay before starting the Spark job execution.

Set the value to .5.

spark.executor.instances

Performance configuration to run mappings on the Spark engine. If you enable dynamic resource allocation for the Spark engine, Informatica recommends that you convert this property to a comment. For example, `#spark.executor.instances=100`

Configure Sqoop Connectivity

Configure the following property for Sqoop connectivity:

infapdo.env.entry.hadoop_node_jdk_home

Configure the HADOOP_NODE_JDK_HOME to represent the JDK version that the cluster nodes use. You must use JDK version 1.7 or later.

Configure the property as follows:

```
infapdo.env.entry.hadoop_node_jdk_home=HADOOP_NODE_JDK_HOME=<cluster JDK home>/
jdk<version>
```

For example,

```
infapdo.env.entry.hadoop_node_jdk_home=HADOOP_NODE_JDK_HOME=/usr/java/default
```

Configure Environment Variables

You can optionally add third-party environment variables and extend the existing PATH environment variable in the hadoopEnv.properties file. The following text shows sample entries to configure environment variables:

```
infapdo.env.entry.oracle_home=ORACLE_HOME=/databases/oracle
infapdo.env.entry.db2_home=DB2_HOME=/databases/db2
infapdo.env.entry.db2instance=DB2INSTANCE=OCA_DB2INSTANCE
infapdo.env.entry.db2codepage=DB2CODEPAGE="1208"
infapdo.env.entry.odbchome=ODBCHOME=$HADOOP_NODE_INFA_HOME/ODBC7.1
infapdo.env.entry.home=HOME=/opt/thirdparty
infapdo.env.entry.gphome_loaders=GPHOME_LOADERS=/databases/greenplum
infapdo.env.entry.pythonpath=PYTHONPATH=$GPHOME_LOADERS/bin/ext
infapdo.env.entry.nz_home=NZ_HOME=/databases/netezza
infapdo.env.entry.ld_library_path=LD_LIBRARY_PATH=$HADOOP_NODE_INFA_HOME/services/
shared/bin:$HADOOP_NODE_INFA_HOME/DataTransformation/bin:$HADOOP_NODE_HADOOP_DIST/lib/
native:$HADOOP_NODE_INFA_HOME/ODBC7.1/lib:$HADOOP_NODE_INFA_HOME/jre/lib/
amd64:$HADOOP_NODE_INFA_HOME/jre/lib/amd64/server:$HADOOP_NODE_INFA_HOME/java/jre/lib/
amd64:$HADOOP_NODE_INFA_HOME/java/jre/lib/amd64/server:/databases/oracle/lib:/
databases/db2/lib64:$LD_LIBRARY_PATH
infapdo.env.entry.path=PATH=$HADOOP_NODE_HADOOP_DIST/scripts:$HADOOP_NODE_INFA_HOME/
services/shared/bin:$HADOOP_NODE_INFA_HOME/jre/bin:$HADOOP_NODE_INFA_HOME/java/jre/bin:
$HADOOP_NODE_INFA_HOME/ODBC7.1/bin:/databases/oracle/bin:/databases/db2/bin:$PATH
#teradata
infapdo.env.entry.twb_root=TWB_ROOT=/databases/teradata/tbuild
infapdo.env.entry.manpath=MANPATH=/databases/teradata/odbc_64:/databases/teradata/odbc_64
infapdo.env.entry.nlspath=NLSPATH=/databases/teradata/odbc_64/msg/%N:/databases/
teradata/msg/%N
infapdo.env.entry.pwd=PWD=/databases/teradata/odbc_64/samples/C
```

Note: If you plan to push down a mapping that includes a Consolidation transformation or a Match transformation, update the Java virtual memory value in the hadoopEnv.properties file.

The hadoopEnv.properties file has the following default Java virtual memory value:

```
infapdo.java.opts=-Xmx512M
```

To run a mapping that includes a Consolidation transformation or a Match transformation, increase the Xmx value to at least 700M.

Update hive-site.xml

Update the hive-site.xml file to set properties for dynamic partitioning.

Configure Dynamic Partitioning

To use Hive dynamic partitioned tables, configure the following properties:

hive.exec.dynamic.partition

Enables dynamic partitioned tables. Set this value to `TRUE`.

exec.dynamic.partition.mode

Allows all partitions to be dynamic. Set this value to `nonstrict`.

Update yarn-site.xml

Update the yarn-site.xml file on the domain to enable access to Hive tables in Amazon S3 buckets.

Configure Access to Hive Tables in Amazon S3 Buckets

Configure the AWS access key to run mappings with sources and targets on Hive tables in Amazon S3 buckets.

Note: To use a Hive table as a target on Amazon S3, grant write permission to the bucket through bucket policies, or add these properties to the configuration file. You must add these properties to the core-site.xml file on the Hadoop environment and to the yarn-site.xml file on the domain environment.

Configure the following properties:

fs.s3a.access.key

The ID for the Blaze and Spark engines to connect to the Amazon S3a file system. For example,

```
<property>
    <name>fs.s3a.access.key</name>
    <value>[Your Access Key]</value>
 </property>
```

fs.s3a.secret.key

The password for the Blaze and Spark engines to connect to the Amazon S3a file system.

```
<property>
    <name>fs.s3a.secret.key</name>
    <value>[Your Access Id]</value>
</property>
```

When you use a server side encryption protocol for Hive buckets, configure properties to enable access to encrypted Hive buckets.

Configure the following properties:

fs.s3.enableServerSideEncryption

Set to TRUE to enable server side encryption. For example,

```
<property>
    <name>fs.s3.enableServerSideEncryption</name>
    <value>TRUE</value>
 </property>
```

fs.s3a.server-side-encryption-algorithm

The encryption algorithm that you use to encrypt Hive buckets. For example,

```
<property>
    <name>fs.s3a.server-side-encryption-algorithm</name>
    <value>AES256</value>
 </property>
```

Update Files on the Hadoop Environment for MapR

You can configure additional functionality through the configuration files the Hadoop environment.

When you update the configuration files, update the files on every node in the cluster. You can find the configuration files in the `/etc/hadoop` directory of the Hadoop distribution. Changes to the files take effect after you restart the Hadoop cluster and services. Ensure that all dependent services start when you restart the cluster.

Update core-site.xml

Update the core-site.xml file on the Hadoop environment to enable access to Hive tables on the cluster and configure user impersonation for Sqoop mappings.

Configure Access to Hive Tables in Amazon S3 Buckets

Configure the AWS access key to run mappings with sources and targets on Hive tables in Amazon S3 buckets.

Note: To use a Hive table as a target on Amazon S3, grant write permission to the bucket through bucket policies, or add these properties to the configuration file. You must add these properties to the core-site.xml file on the Hadoop environment and to the yarn-site.xml file on the domain environment.

Configure the following properties:

fs.s3a.access.key

The ID for the Blaze and Spark engines to connect to the Amazon S3a file system. For example,

```
<property>
    <name>fs.s3a.access.key</name>
    <value>[Your Access Key]</value>
 </property>
```

fs.s3a.secret.key

The password for the Blaze and Spark engines to connect to the Amazon S3a file system.

```
<property>
    <name>fs.s3a.secret.key</name>
    <value>[Your Access Id]</value>
</property>
```

When you use a server side encryption protocol for Hive buckets, configure properties to enable access to encrypted Hive buckets.

Configure the following properties:

fs.s3.enableServerSideEncryption

Set to TRUE to enable server side encryption. For example,

```
<property>
    <name>fs.s3.enableServerSideEncryption</name>
    <value>TRUE</value>
 </property>
```

fs.s3a.server-side-encryption-algorithm

The encryption algorithm that you use to encrypt Hive buckets. For example,

```
<property>
    <name>fs.s3a.server-side-encryption-algorithm</name>
    <value>AES256</value>
 </property>
```

Configure Impersonation for Sqoop

To run Sqoop mappings on non-secure MapR clusters, you must create a proxy user for the yarn user who will impersonate other users.

Configure the following properties, and then restart Hadoop services and the cluster:

hadoop.proxyuser.yarn.groups

Allows impersonation from any group.

```
<property>
    <name>hadoop.proxyuser.yarn.groups</name>
    <value><Name_of_the_impersonation_user></value>
    <description>Allows impersonation from any group.</description>
</property>
```

hadoop.proxyuser.yarn.hosts

Allows impersonation from any host.

```
<property>
    <name>hadoop.proxyuser.yarn.hosts</name>
    <value>*</value>
    <description>Allows impersonation from any host.</description>
</property>
```

Update hive-site.xml

Update hive-site.xml on the Hadoop environment to enable mappings to read or write to Hive tables on the cluster.

When the cluster uses Kerberos authentication and the mapping is configured to use the Blaze or Spark run-time engines, configure the following property in `hive-site.xml` on the cluster:

hive.server2.enable.doAs

The authentication that the server is set to use. Default is FALSE.

To enable access to HiveServer2, set the property to TRUE. When you set the property to TRUE, you enable the Blaze and Spark engines to access the data in the Hive tables as the DIS user, or as the impersonation user when impersonation is enabled.

When you set the property to FALSE, you enable HiveServer2 to run MapReduce jobs as the Hive user.

Update yarn-site.xml

Update the yarn-site.xml file on the MapR cluster to configure functionality such as resource allocation and virtual memory limits.

Allocate Cluster Resources for the Blaze Engine

Update the following properties to verify that the cluster allocates sufficient memory and resources for the Blaze engine:

yarn.nodemanager.resource.memory-mb

The maximum RAM available for each container. Set the maximum memory on the cluster to increase resource memory available to the Blaze engine. Configure this property on the following nodes:

- Run-time nodes. Set to at least 10 GB.
- Management node. Set to at least 13 GB.

yarn.nodemanager.resource.cpu-vcores

The number of virtual cores for each container. The number might correspond to the number of physical cores, but you can increase the value to allow for more processing. Configure this property on the following nodes:

- Run-time nodes. Set to at 6.
- Management node. Set to 9.

yarn.scheduler.minimum-allocation-mb

The minimum RAM available for each container. Set the minimum memory to allow the VM to spawn sufficient containers. Configure this property to be no less than 4 GB of the maximum memory on the run-time and management nodes.

Configure Virtual Memory Limits for the Blaze Engine

Configure the following property to remove virtual memory limits for every node in the Hadoop cluster:

yarn.nodemanager.vmem-check-enabled

Determines virtual memory limits. Set the value to false to disable virtual memory limits. For example,

```
<property>
    <name>yarn.nodemanager.vmem-check-enabled</name>
    <value>false</value>
    <description>Enforces virtual memory limits for containers.</description>
</property>
```

Configure Dynamic Resource Allocation for the Spark Engine

Configure dynamic resource allocation for mappings to run on the Spark engine. Update the following properties on each node where YARN node manager is running:

yarn.nodemanager.aux-services

The auxiliary service name. Add spark_shuffle to the current entry. For example, if the current value is "mapreduce_shuffle," update it to be "mapreduce_shuffle,spark_shuffle."

```
<property>
    <name>yarn.nodemanager.aux-services</name>
    <value>mapreduce_shuffle,spark_shuffle</value>
</property>
```

yarn.nodemanager.aux-services.spark_shuffle.class

The auxiliary service class to use. Set to org.apache.spark.network.yarn.YarnShuffleService. For example,

```
<property>
    <name>yarn.nodemanager.aux-services.spark_shuffle.class</name>
    <value>org.apache.spark.network.yarn.YarnShuffleService</value>
</property>
```

Generate MapR Tickets

To run mappings on a MapR cluster that uses Kerberos or MapR Ticket authentication with information in Hive tables, generate a MapR ticket for the following users:

Data Integration Service User

The Data Integration Service user requires an account on the MapR cluster and a MapR ticket on the Data Integration Service machine.

When the MapR cluster uses both Kerberos and Ticket authentication, you generate a ticket for the Data Integration Service user for each authentication system.

Developer Tool User

The Developer tool user requires an account on the MapR cluster and a MapR ticket on the machine where the Developer tool is installed.

After you generate and save MapR tickets, you perform additional steps to configure application services to communicate with the MapR cluster.

Create the Developer Tool User

To enable the Developer tool to communicate with the MapR cluster, create and configure an account for the Developer tool user on every node in the cluster.

Generate Tickets

After you create a MapR user account for the Data Integration Service user and the Developer tool user, generate a MapR ticket for each user and save it to a local directory, depending on the user requirements that are listed below.

To generate a MapR ticket, refer to MapR documentation.

Data Integration Service User Ticket

Generate a MapR ticket for the Data Integration Service user. Name the ticket file using the following naming convention:

```
maprticket_<user name>
```

Save the ticket file in the `/tmp` directory of the machine that runs the Data Integration Service.

When the MapR cluster is configured to enable a user to use Kerberos authentication and MapR Ticket authentication, you generate a MapR ticketfile for the user for each authentication mode. Save one ticketfile in /tmp. Save the other ticketfile in any directory on the Data Integration Service machine, and provide the location as the value for the MAPR_TICKETFILE_LOCATION property in the Data Integration Service Process properties.

Developer Tool User Ticket

Generate a MapR ticket for the Developer tool user. Name the ticket file using the following naming convention:

```
maprticket_<user name>
```

Save the ticket file in the `%TEMP%` directory of the machine the runs the Developer tool.

Configure Informatica Application Services

Configure properties on the following Informatica application services:

Data Integration Service

When the MapR cluster is secured with MapR Kerberos authentication, edit Data Integration Service properties to enable communication between the Informatica domain and the cluster.

Analyst Service

If you use the Analyst tool to profile data in Hive data objects, configure properties on the Analyst Service to enable communication between the Analyst tool and the cluster, including testing of the Hive connection.

Configure the Data Integration Service

When the MapR cluster is secured with MapR Kerberos authentication, edit Data Integration Service properties to enable communication between the Informatica domain and the cluster.

Data Integration Service Properties

In the Administrator tool Domain Navigator, select the Data Integration Service to configure, then select the **Properties** tab.

In the **Custom Properties** area, define the following property value:

Property	Value
ExecutionContextOptions.JVMOption1	`-Dmapr.library.flatclass`

Data Integration Service Process Properties

In the Administrator tool Domain Navigator, select the Data Integration Service to configure, and then select the **Processes** tab.

In the **Custom Properties** area, define the following properties and values:

Property	Value
ExecutionContextOptions.JVMOption	`-Djava.security.krb5.conf=<Informatica installation directory>/services/shared/security/krb5.conf`
ExecutionContextOptions.JVMOption1	`-Dmapr.library.flatclass`
ExecutionContextOptions.JVMOption2	`-Dhadoop.login=<MAPR_ECOSYSTEM_LOGIN_OPTS> -Dhttps.protocols=TLSv1.2` where <MAPR_ECOSYSTEM_LOGIN_OPTS> is the value of the MAPR_ECOSYSTEM_LOGIN_OPTS property in the file `/opt/mapr/conf/env.sh`. For example, `-Dhadoop.login=hybrid`
ExecutionContextOptions.JVMOption7	`-Dhttps.protocols=TLSv1.2`

In the **Environment Variables** area, configure the following property to define the Kerberos authentication protocol:

Property	Value
JAVA_OPTS	`-Dhadoop.login=<MAPR_ECOSYSTEM_LOGIN_OPTS> -Dhttps.protocols=TLSv1.2` where <MAPR_ECOSYSTEM_LOGIN_OPTS> is the value of the MAPR_ECOSYSTEM_LOGIN_OPTS property in the file `/opt/mapr/conf/env.sh`.
MAPR_HOME	Hadoop distribution directory location on the machine that runs the Data Integration Service. For example, `<Informatica installation directory>/services/shared/hadoop/mapr_5.2.0`
MAPR_TICKETFILE_LOCATION	Optional. Directory where an additional MapR Ticket file is stored on the machine that runs the Data Integration Service. When the MapR cluster is configured to enable a user to use Kerberos authentication and MapR Ticket authentication, generate a MapR ticketfile for the user for each authentication mode. Save one ticketfile in `/tmp`. Save the other ticketfile in any directory on the Data Integration Service machine, and provide the location as the value for this property. For example, for a user id 1234, save a MapR ticketfile named like `maprticket_1234` in `/tmp`, and save another MapR ticketfile named like `maprticket_1234` in the MAPR_TICKETFILE_LOCATION. **Note:** The ticketfiles can have the same or different names. You must generate the MapR ticketfiles separately and save one to the MAPR_TICKETFILE_LOCATION.

Changes take effect when you restart the Data Integration Service.

Configure the Analyst Service

If you use the Analyst tool to profile data in Hive data objects, configure properties on the Analyst Service to enable communication between the Analyst tool and the cluster, including testing of the Hive connection.

In the Administrator tool Domain Navigator, select the Analyst Service to configure, then select the **Processes** tab.

In the **Advanced Properties** area, define the following property value:

Property	Value
ExecutionContextOptions.JVMOption1	`-Dmapr.library.flatclass`

In the **Environment Variables** area, configure the following property to define the Kerberos authentication protocol:

Property	Value
JAVA_OPTS	`-Dhadoop.login=hybrid -Dhttps.protocols=TLSv1.2`
MAPR_HOME	Hadoop distribution directory location on the machine that runs the Data Integration Service. For example, `<Informatica installation directory>/services/shared/hadoop/mapr_5.2.0`
MAPR_TICKETFILE_LOCATION	Directory where the MapR Ticket file is stored on the machine that runs the Analyst Service. For example, `/export/home/username1/Keytabs_and_krb5conf/Tickets/project1/maprticket_30103`
LD_LIBRARY_PATH	The location of Hadoop libraries. For example, `<Informatica installation directory>/java/jre/lib:<Informatica installation directory>/services/shared/bin:<Informatica installation directory>/server/bin:<Informatica installation directory>/services/shared/hadoop/<MapR location>/lib/native/Linux-amd64-64`

Changes take effect when you restart the Analyst Service.

Test the Hive Connection

After you configure users for MapR Ticket or Kerberos authentication on MapR clusters, you can test the Hive connection.

To test the Hive connection, or perform a metadata fetch task, use the following format for the connection string if the cluster is Kerberos-enabled:

```
jdbc:hive2://<hostname>:10000/default;principal=<SPN>
```

For example,

```
jdbc:hive2://myServer2:10000/default;principal=mapr/myServer2@clustername
```

Note: When the mapping performs a metadata fetch of a complex file object, the user whose maprticket is present at %TEMP% on the Windows machine must have read permission on the HDFS directory to list the files inside it and perform the import action. The metadata fetch operation ignores privileges of the user who is listed in the HDFS connection definition.

Developer Tool Tasks

Perform tasks on the machine that hosts the Developer tool. Update configuration files to enable communication with the Hadoop cluster and to enable high availability between the Developer tool and the

cluster. Update connection properties to enable high availability between the Data Integration Service and the cluster.

Before you perform configuration tasks for the Developer tool, copy configuration files to the machine that hosts the Developer tool.

1. Find the following files on the Resource Manager node of the cluster:
   - `hive-site.xml`
   - `mapr-clusters.conf`
   - `yarn-site.xml`
2. Copy the files to the following location on the machine that hosts the Developer tool: `<Informatica installation directory>/clients/DeveloperClient/Hadoop/<Hadoop distribution><version>/conf`

Configure Files on the Developer Tool

Configure files on the Developer tool client machine to create, edit and run mappings on the MapR cluster.

run.bat

Edit the `run.bat` file to enable Developer tool launch settings for MapR.

You can find `run.bat` in the following directory on the machine where the Developer tool runs: `<Informatica installation directory>\<version_number>\clients\DeveloperClient`

Edit `run.bat` to include the MAPR_HOME environment variable and the -clean settings. For example, include the following lines:

```
<Informatica installation directory>\clients\DeveloperClient\hadoop\mapr_5.2.0
developerCore.exe -clean
```

developerCore.ini

Edit the `developerCore.ini` file to enable communication between the Developer tool and the Hadoop cluster.

You can find developerCore.ini in the following directory: `<Informatica installation directory>\clients\DeveloperClient`

Configure the following properties:

-Dmapr.library.flatclass -Djava.library.path

Path to the Java library on the machine that hosts the Developer tool.

Use the following value: `hadoop\mapr_5.2.0\lib\native\Win64;bin;..\DT\bin`

-DINFA_HADOOP_DIST_DIR

Path to the Hadoop distribution directory on the cluster. Use the following value: `hadoop\<distribution>_<version>`

For example, `-DINFA_HADOOP_DIST_DIR=hadoop\mapr_5.2.0`

hive-site.xml

Edit the `hive-site.xml` file. You can find `hive-site.xml` in the following directory: `<Informatica installation directory>\clients\DeveloperClient\hadoop\mapr_<version_number>\conf\`

Add the following property to hive-site.xml to enable SASL:

```
<property>
 <name>hive.metastore.sasl.enabled</name>
<value>true</value>
</property>
```

Complete Upgrade Tasks

If you upgraded the Informatica platform or applied the hotfix, you need to perform some additional tasks within the Informatica domain.

Based on the version that you upgraded from, perform the following tasks:

Update the hadoopEnv.properties file.

Applies to all upgrades. The 10.1.1 HotFix 1 hadoopEnv.properties file contains additional properties. You need to manually update it to include customized configuration from previous versions.

Complete connection upgrades.

Applies to upgrades from 9.6.1 or any 9.6.1 hotfix. You need to generate Hadoop connections from the Hive connection and update parameter files.

Update connection properties.

Applies to any upgrade. If you did not create connections when you ran the Hadoop Configuration Manager, you must update values for connection properties that changed.

Replace connections.

Applies to any upgrade. If you created connections when you ran the Hadoop Configuration Manager, you need to replace connections in mappings with the new connections.

Update the hadoopEnv.properties File

When you run the Informatica upgrade or apply the hotfix on a machine that hosts the Data Integration Service, the installer creates a new hadoopEnv.properties file and backs up the existing configuration file.

You need to edit the version 10.1.1 HotFix 1 hadoopEnv.properties file to include any manual configuration that you performed in the previous version for the corresponding distribution.

If you applied the hotfix, you can find the backup hadoopEnv.properties file in the following location:

```
<Informatica installation directory>/BACK_FROM_HF1/services/shared/hadoop/<Hadoop
distribution name>_<version>/infaConf
```

If you upgraded from a previous version, you can find the backup hadoopEnv.properties file in the following location:

```
<Previous Informatica installation directory>/services/shared/hadoop/<Hadoop distribution
name>_<version>/infaConf
```

You can find the 10.1.1 HotFix 1 hadoopEnv.properties file in the following location:

```
<Informatica installation directory>/services/shared/hadoop/<Hadoop distribution
name>_<version>/infaConf
```

Complete Connection Upgrade

Effective in version 10.0, Big Data Management requires a Hadoop connection to run mappings on the Hadoop cluster. If you upgraded from 9.6.1 or any 9.6.1 hotfix, you must generate Hadoop connections from Hive connections that are enabled to run mappings in the Hadoop environment.

The upgrade process generates a connection name for the Hadoop connection and replaces the connection name in the mappings. It does not create the physical connection object. When the upgrade is complete, you must run a command to generate the connection. Generate Hadoop connections from Hive connections if the following conditions are true:

- You upgraded from 9.6.1 or any 9.6.1 hotfix.
- The Hive connections are configured to run mappings in the Hadoop environment.

Complete the following tasks to upgrade connections:

1. Run `infacmd isp generateHadoopConnectionFromHiveConnection` to generate a Hadoop connection from a Hive connection that is configured to run in the Hadoop environment.
   The command renames the connection as follows: "Autogen_<Hive connection name>." If the renamed connection exceeds the 128 character limit, the command fails.
2. If the command fails, complete the following tasks:
   a. Rename the connection to meet the character limit and run the command again.
   b. Run `infacmd dis replaceMappingHadoopRuntimeConnections` to replace connections associated with mappings that are deployed in applications.
   c. Run `infacmd mrs replaceMappingHadoopRuntimeConnections` to replace connections associated with mappings that you run from the Developer tool.
3. If the Hive connection was parameterized, you must update the connection names in the parameter file. Verify that the Hive sources, Hive targets, and the Hive engine parameters are updated with the correct connection name.
4. If any properties changed in the cluster, such as host names, URIs, or port numbers, you must update the properties in the connections.

For information about the infacmd commands, see the *Informatica Command Reference*.

Update Connection Properties

If you upgraded from version 10.0 or later, and you did not create connections when you ran the Hadoop Configuration Manager, you need to verify connection properties.

Review the Hadoop, Hive, HDFS, and HBase connections that you created in a previous release to update the values for connection properties. For example, if you added nodes to the cluster or if you updated the distribution version, you might need to verify host names, URIs, or port numbers for some of the properties.

Replace the Connections

If you created connections when you ran the Hadoop Configuration Manager, you need to replace connections in mappings with the new connections.

The method that you use to replace connections in mappings depends on the type of connection.

Hadoop connection

If you created a Hadoop connection when you ran the Hadoop Configuration Manager, run the following commands:

- `infacmd dis replaceMappingHadoopRuntimeConnections`. Replaces connections associated with mappings that are deployed in applications.
- `infacmd mrs replaceMappingHadoopRuntimeConnections`. Replaces connections associated with mappings that you run from the Developer tool.

For information about the infacmd commands, see the *Informatica Command Reference*.

Hive, HDFS, or HBase connection

If you created a Hive, HDFS, or HBase connection when you ran the Hadoop Configuration Manager, you must replace the connections manually.

CHAPTER 25

Uninstall for MapR

This chapter includes the following topics:

- Before You Uninstall, 184
- Uninstall Big Data Management, 184

Before You Uninstall

Verify prerequisites before you uninstall Big Data Management.

1. Verify that the Big Data Management administrator can run `sudo` commands.
2. If you are uninstalling Big Data Management in a cluster environment, configure the root user to use a passwordless Secure Shell (SSH) connection between the machine where you want to run the Big Data Management uninstall and all of the nodes where Big Data Management is installed.
3. If you are uninstalling Big Data Management in a cluster environment using the `HadoopDataNodes` file, verify that the `HadoopDataNodes` file contains the IP addresses or machine host names of each of the nodes in the Hadoop cluster from which you want to uninstall Big Data Management. The `HadoopDataNodes` file is located on the node from where you want to launch the Big Data Management installation. You must add one IP address or machine host name of the nodes in the Hadoop cluster for each line in the file.

Uninstall Big Data Management

You can uninstall Big Data Management from a single node or cluster environment.

1. Log in to the machine as root user. The machine you log in to depends on the Big Data Management environment and uninstallation method.
   - To uninstall in a single node environment, log in to the machine on which Big Data Management is installed.
   - To uninstall in a cluster environment using the HADOOP_HOME environment variable, log in to the primary name node.
   - To uninstall in a cluster environment using the `HadoopDataNodes` file, log in to any node.

2. Run the following command to start the uninstallation in console mode:

```
bash InformaticaHadoopInstall.sh
sh InformaticaHadoopInstall.sh
./InformaticaHadoopInstall.sh
```

3. Press **y** to accept the Big Data Management terms of agreement.
4. Press **Enter**.
5. Select **3** to uninstall Big Data Management.
6. Press **Enter**.
7. Select the uninstallation option, depending on the Big Data Management environment:
   - Select **1** to uninstall Big Data Management from a single node environment.
   - Select **2** to uninstall Big Data Management from a cluster environment.
8. Press **Enter**.
9. If you are uninstalling Big Data Management in a cluster environment, select the uninstallation option, depending on the uninstallation method:
   - Select **1** to uninstall Big Data Management from the primary name node.
   - Select **2** to uninstall Big Data Management using the `HadoopDataNodes` file.
10. Press **Enter**.
11. If you are uninstalling Big Data Management from a cluster environment from the primary name node, type the absolute path for the Hadoop installation directory. Start the path with a slash.

The uninstaller deletes all of the Big Data Management binary files from the following directory: `/<Big Data Management installation directory>/Informatica`
In a cluster environment, the uninstaller delete the binary files from all of the nodes within the Hadoop cluster.

APPENDIX A

Connection Property Reference

This appendix includes the following topics:

- Connection Property Reference Overview, 186
- Hadoop Connection Properties, 187
- HBase Connection Properties, 196
- HDFS Connection Properties, 197
- Hive Connection Properties, 198
- JDBC Connection Properties, 205

Connection Property Reference Overview

Define a Hadoop connection to run a mapping in the Hadoop environment. Depending on the sources and targets, define connections to access data in HBase, HDFS, Hive, or relational databases. You can create the connections using the Developer tool, Administrator tool, and infacmd.

You can create the following types of connections:

Hadoop connection

Create a Hadoop connection to run mappings in the Hadoop environment. If you select the mapping validation environment or the execution environment as Hadoop, select the Hadoop connection. Before you run mappings in the Hadoop environment, review the information in this guide about rules and guidelines for mappings that you can run in the Hadoop environment.

HBase connection

Create an HBase connection to access HBase. The HBase connection is a NoSQL connection.

HDFS connection

Create an HDFS connection to read data from or write data to the HDFS file system on a Hadoop cluster.

Hive connection

Create a Hive connection to access Hive as a source or target. You can access Hive as a source if the mapping is enabled for the native or Hadoop environment. You can access Hive as a target if the mapping runs on the Blaze or Hive engine.

JDBC connection

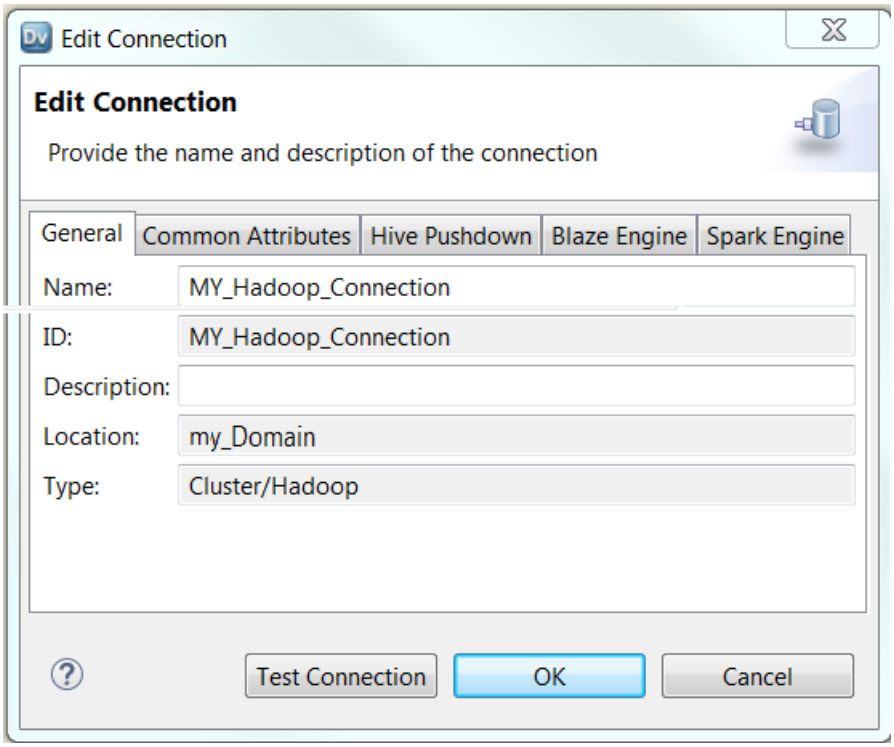
Create a JDBC connection and configure Sqoop properties in the connection to import and export relational data through Sqoop.

Note: For information about creating connections to other sources or targets such as social media web sites or Teradata, see the respective PowerExchange adapter user guide for information.

Hadoop Connection Properties

Use the Hadoop connection to configure mappings to run on a Hadoop cluster. A Hadoop connection is a cluster type connection. You can create and manage a Hadoop connection in the Administrator tool or the Developer tool. You can use infacmd to create a Hadoop connection. Hadoop connection properties are case sensitive unless otherwise noted.

The following image shows the Hadoop connection properties:



General Properties

The following table describes the general connection properties for the Hadoop connection:

Property	Description
Name	The name of the connection. The name is not case sensitive and must be unique within the domain. You can change this property after you create the connection. The name cannot exceed 128 characters, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
ID	String that the Data Integration Service uses to identify the connection. The ID is not case sensitive. It must be 255 characters or less and must be unique in the domain. You cannot change this property after you create the connection. Default value is the connection name.

Property	Description
Description	The description of the connection. Enter a string that you can use to identify the connection. The description cannot exceed 4,000 characters.
Location	The domain where you want to create the connection. Select the domain name.
Type	The connection type. Select Hadoop.

Common Attributes - Hadoop Cluster Properties

The following table describes the connection properties that you configure for the Hadoop cluster:

Property	Description

Resource Manager Address	The service within Hadoop that submits requests for resources or spawns YARN applications.
	Use the following format:
	`<hostname>:<port>`
	Where
	- <hostname> is the host name or IP address of the Yarn resource manager.
	- <port> is the port on which the Yarn resource manager listens for remote procedure calls (RPC).
	For example, enter: `myhostame:8032`
	You can also get the Resource Manager Address property from yarn-site.xml located in the following directory on the Hadoop cluster: `/etc/hadoop/conf/`
	The Resource Manager Address appears as the following property in yarn-site.xml:
	`<property>`
	`<name>yarn.resourcemanager.address</name>`
	`<value>hostname:port</value>`
	`<description>The address of the applications manager interface in the Resource Manager.</description>`
	`</property>`
	Optionally, if the `yarn.resourcemanager.address` property is not configured in yarn-site.xml, you can find the host name from the `yarn.resourcemanager.hostname` or `yarn.resourcemanager.scheduler.address` properties in yarn-site.xml. You can then configure the Resource Manager Address in the Hadoop connection with the following value: `hostname:8032`

Default File System URI	The URI to access the default Hadoop Distributed File System.
	Use the following connection URI:
	`hdfs://<node name>:<port>`
	Where
	- <node name> is the host name or IP address of the NameNode.
	- <port> is the port on which the NameNode listens for remote procedure calls (RPC).
	For example, enter: `hdfs://myhostname:8020/`
	You can also get the Default File System URI property from core-site.xml located in the following directory on the Hadoop cluster: `/etc/hadoop/conf/`
	Use the value from the `fs.defaultFS` property found in core-site.xml.
	For example, use the following value:
	`<property>`
	`<name>fs.defaultFS</name>`
	`<value>hdfs://localhost:8020</value>`
	`</property>`
	If the Hadoop cluster runs MapR, use the following URI to access the MapR File system: `maprfs:///`.
	The Azure HDInsight File System default file system can be Windows Azure Storage Blob (WASB) or Azure Data Lake Store (ADLS).
	If the cluster uses WASB storage, use the following string to specify the URI:
	`wasb://<container_name>@<account_name>.blob.core.windows.net/<path>`
	where:
	- `<container_name>` identifies a specific Azure Storage Blob container.
	Note: `<container_name>` is optional.
	- `<account_name>` identifies the Azure Storage Blob object.

Property	Description

	Example:
	`wasb://infabdmoffering1storage.blob.core.windows.net/infabdmoffering1cluster/mr-history`
	If the cluster uses ADLS storage, use the following format to specify the URI: `adl://home`
	The following is the `fs.defaultFS` property as it appears in `hdfs-site.xml`:
	`<property>fs.defaultFS</property>`
	`<value>adl://home</value>`

Common Attributes - Common Properties

The following table describes the common connection properties that you configure for the Hadoop connection:

Property	Description
Impersonation User Name	User name of the user that the Data Integration Service impersonates to run mappings on a Hadoop cluster. If the Hadoop cluster uses Kerberos authentication, the principal name for the JDBC connection string and the user name must be the same. **Note:** You must use user impersonation for the Hadoop connection if the Hadoop cluster uses Kerberos authentication. If the Hadoop cluster does not use Kerberos authentication, the user name depends on the behavior of the JDBC driver. If you do not specify a user name, the Hadoop cluster authenticates jobs based on the operating system profile user name of the machine that runs the Data Integration Service.
Temporary Table Compression Codec	Hadoop compression library for a compression codec class name.
Codec Class Name	Codec class name that enables data compression and improves performance on temporary staging tables.
Hadoop Connection Custom Properties	Custom properties that are unique to the Hadoop connection. You can specify multiple properties. Use the following format: `<property1>=<value>` Where - <property1> is a Blaze, Hive, or Hadoop property. - <value> is the value of the Hive or Hadoop property. To specify multiple properties use `&:` as the property separator. Use custom properties only at the request of Informatica Global Customer Support.

Hive Pushdown - Hive Pushdown Configuration

The following table describes the connection properties that you configure to push mapping logic to the Hadoop cluster:

Property	Description
Environment SQL	SQL commands to set the Hadoop environment. The Data Integration Service executes the environment SQL at the beginning of each Hive script generated in a Hive execution plan. The following rules and guidelines apply to the usage of environment SQL: - Use the environment SQL to specify Hive queries. - Use the environment SQL to set the classpath for Hive user-defined functions and then use environment SQL or PreSQL to specify the Hive user-defined functions. You cannot use PreSQL in the data object properties to specify the classpath. The path must be the fully qualified path to the JAR files used for user-defined functions. Set the parameter hive.aux.jars.path with all the entries in infapdo.aux.jars.path and the path to the JAR files for user-defined functions. - You can use environment SQL to define Hadoop or Hive parameters that you want to use in the PreSQL commands or in custom queries. - If you use multiple values for the environment SQL, ensure that there is no space between the values. The following sample text shows two values that can be used for the Environment SQL property: `set hive.execution.engine='tez';set hive.exec.dynamic.partition.mode='nonstrict';`
Database Name	Namespace for tables. Use the name `default` for tables that do not have a specified database name.

Hive Warehouse Directory on HDFS	The absolute HDFS file path of the default database for the warehouse that is local to the cluster. For example, the following file path specifies a local warehouse: `/user/hive/warehouse`
	For Cloudera CDH, if the Metastore Execution Mode is remote, then the file path must match the file path specified by the Hive Metastore Service on the Hadoop cluster.
	You can get the value for the Hive Warehouse Directory on HDFS from the `hive.metastore.warehouse.dir` property in hive-site.xml located in the following directory on the Hadoop cluster: `/etc/hadoop/conf/`
	For example, use the following value:
	`<property>`
	`<name>hive.metastore.warehouse.dir</name>`
	`<value>/usr/hive/warehouse </value>`
	`<description>location of the warehouse directory</description>`
	`</property>`
	For MapR, `hive-site.xml` is located in the following directecory: `/opt/mapr/hive/<hive version>/conf`.

Hive Pushdown - Hive Configuration

You can use the values for Hive configuration properties from hive-site.xml or mapred-site.xml located in the following directory on the Hadoop cluster: `/etc/hadoop/conf/`.

The following table describes the connection properties that you configure for the Hive engine:

Property	Description

Metastore Execution Mode	Controls whether to connect to a remote metastore or a local metastore. By default, local is selected. For a local metastore, you must specify the Metastore Database URI, Metastore Database Driver, Username, and Password. For a remote metastore, you must specify only the Remote Metastore URI.
	You can get the value for the Metastore Execution Mode from hive-site.xml. The Metastore Execution Mode appears as the following property in hive-site.xml:
	`<property>`
	`<name>hive.metastore.local</name>`
	`<value>true</true>`
	`</property>`
	Note: The `hive.metastore.local` property is deprecated in hive-site.xml for Hive server versions 0.9 and above. If the `hive.metastore.local` property does not exist but the `hive.metastore.uris` property exists, and you know that the Hive server has started, you can set the connection to a remote metastore.

Metastore Database URI	The JDBC connection URI used to access the data store in a local metastore setup. Use the following connection URI:
	`jdbc:<datastore type>://<node name>:<port>/<database name>`
	where
	- <node name> is the host name or IP address of the data store.
	- <data store type> is the type of the data store.
	- <port> is the port on which the data store listens for remote procedure calls (RPC).
	- <database name> is the name of the database.
	For example, the following URI specifies a local metastore that uses MySQL as a data store:
	`jdbc:mysql://hostname23:3306/metastore`
	You can get the value for the Metastore Database URI from hive-site.xml. The Metastore Database URI appears as the following property in hive-site.xml:
	`<property>`
	`  <name>javax.jdo.option.ConnectionURL</name>`
	`  <value>jdbc:mysql://MYHOST/metastore</value>`
	`</property>`

Metastore Database Driver	Driver class name for the JDBC data store. For example, the following class name specifies a MySQL driver:
	`com.mysql.jdbc.Driver`
	You can get the value for the Metastore Database Driver from hive-site.xml. The Metastore Database Driver appears as the following property in hive-site.xml:
	`<property>`
	`  <name>javax.jdo.option.ConnectionDriverName</name>`
	`  <value>com.mysql.jdbc.Driver</value>`
	`</property>`

Metastore Database User Name	The metastore database user name.
	You can get the value for the Metastore Database User Name from hive-site.xml. The Metastore Database User Name appears as the following property in hive-site.xml:
	`<property>`
	`  <name>javax.jdo.option.ConnectionUserName</name>`
	`  <value>hiveuser</value>`
	`</property>`

Property	Description

Metastore Database Password	The password for the metastore user name.
	You can get the value for the Metastore Database Password from hive-site.xml. The Metastore Database Password appears as the following property in hive-site.xml:
	`<property>`
	`  <name>javax.jdo.option.ConnectionPassword</name>`
	`  <value>password</value>`
	`</property>`

Remote Metastore URI	The metastore URI used to access metadata in a remote metastore setup. For a remote metastore, you must specify the Thrift server details.
	Use the following connection URI:
	`thrift://<hostname>:<port>`
	Where
	- <hostname> is name or IP address of the Thrift metastore server.
	- <port> is the port on which the Thrift server is listening.
	For example, enter: `thrift://myhostname:9083/`
	You can get the value for the Remote Metastore URI from hive-site.xml. The Remote Metastore URI appears as the following property in hive-site.xml:
	`<property>`
	`  <name>hive.metastore.uris</name>`
	`  <value>thrift://<n.n.n.n>:9083</value>`
	`<description> IP address or fully-qualified domain name and port of the metastore host</description>`
	`</property>`

Property	Description

Engine Type	The engine that the Hadoop environment uses to run a mapping on the Hadoop cluster. Select a value from the drop down list.
	For example select: `MRv2`
	To set the engine type in the Hadoop connection, you must get the value for the `mapreduce.framework.name` property from mapred-site.xml located in the following directory on the Hadoop cluster: `/etc/hadoop/conf/`
	If the value for `mapreduce.framework.name` is `classic`, select `mrv1` as the engine type in the Hadoop connection.
	If the value for `mapreduce.framework.name` is `yarn`, you can select the `mrv2` or `tez` as the engine type in the Hadoop connection. Do not select Tez if Tez is not configured for the Hadoop cluster.
	You can also set the value for the engine type in hive-site.xml. The engine type appears as the following property in hive-site.xml:
	`<property>`
	`<name>hive.execution.engine</name>`
	`<value>tez</value>`
	`<description>Chooses execution engine. Options are: mr (MapReduce, default) or tez (Hadoop 2 only)</description>`
	`</property>`

Job Monitoring URL	The URL for the MapReduce JobHistory server. You can use the URL for the JobTracker URI if you use MapReduce version 1.
	Use the following format:
	`<hostname>:<port>`
	Where
	- <hostname> is the host name or IP address of the JobHistory server.
	- <port> is the port on which the JobHistory server listens for remote procedure calls (RPC).
	For example, enter: `myhostname:8021`
	You can get the value for the Job Monitoring URL from mapred-site.xml. The Job Monitoring URL appears as the following property in mapred-site.xml:
	`<property>`
	`<name>mapred.job.tracker</name>`
	`<value>myhostname:8021 </value>`
	`<description>The host and port that the MapReduce job tracker runs at.</description>`
	`</property>`

Blaze Engine

The following table describes the connection properties that you configure for the Blaze engine:

Property	Description
Temporary Working Directory on HDFS	The HDFS file path of the directory that the Blaze engine uses to store temporary files. Verify that the directory exists. The YARN user, Blaze engine user, and mapping impersonation user must have write permission on this directory. For example, enter: `/blaze/workdir`
Blaze Service User Name	The operating system profile user name for the Blaze engine.
Minimum Port	The minimum value for the port number range for the Blaze engine. For example, enter: `12300`

Property	Description
Maximum Port	The maximum value for the port number range for the Blaze engine. For example, enter: `12600`
Yarn Queue Name	The YARN scheduler queue name used by the Blaze engine that specifies available resources on a cluster. The name is case sensitive.
Blaze Service Custom Properties	Custom properties that are unique to the Blaze engine. You can specify multiple properties. Use the following format: `<property1>=<value>` Where - <property1> is a Blaze engine optimization property. - <value> is the value of the Blaze engine optimization property. To enter multiple properties, separate each name-value pair with the following text: `&:`. Use custom properties only at the request of Informatica Global Customer Support.

Spark Engine

The following table describes the connection properties that you configure for the Spark engine:

Property	Description
Spark HDFS Staging Directory	The HDFS file path of the directory that the Spark engine uses to store temporary files for running jobs. The YARN user, Spark engine user, and mapping impersonation user must have write permission on this directory.
Spark Event Log Directory	Optional. The HDFS file path of the directory that the Spark engine uses to log events. The Data Integration Service accesses the Spark event log directory to retrieve final source and target statistics when a mapping completes. These statistics appear on the Summary Statistics tab and the Detailed Statistics tab of the Monitoring tool. If you do not configure the Spark event log directory, the statistics might be incomplete in the Monitoring tool.
Spark Execution Parameters	An optional list of configuration parameters to apply to the Spark engine. You can change the default Spark configuration properties values, such as `spark.executor.memory` or `spark.driver.cores`. Use the following format: `<property1>=<value>` - <property1> is a Spark configuration property. - <value> is the value of the property. For example, you can configure a YARN scheduler queue name that specifies available resources on a cluster: `spark.yarn.queue=TestQ` To enter multiple properties, separate each name-value pair with the following text: `&:`

HBase Connection Properties

Use an HBase connection to access HBase. The HBase connection is a NoSQL connection. You can create and manage an HBase connection in the Administrator tool or the Developer tool. HBase connection properties are case sensitive unless otherwise noted.

The following table describes HBase connection properties:

Property	Description
Name	The name of the connection. The name is not case sensitive and must be unique within the domain. You can change this property after you create the connection. The name cannot exceed 128 characters, contain spaces, or contain the following special characters: `~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /`
ID	String that the Data Integration Service uses to identify the connection. The ID is not case sensitive. It must be 255 characters or less and must be unique in the domain. You cannot change this property after you create the connection. Default value is the connection name.
Description	The description of the connection. The description cannot exceed 4,000 characters.
Location	The domain where you want to create the connection.
Type	The connection type. Select HBase.
ZooKeeper Host(s)	Name of the machine that hosts the ZooKeeper server.
ZooKeeper Port	Port number of the machine that hosts the ZooKeeper server. Use the value specified for `hbase.zookeeper.property.clientPort` in hbase-site.xml. You can find `hbase-site.xml` on the Namenode machine in the following directory: `/opt/HDinsight/hbase/hbase-0.98.7/conf`
Enable Kerberos Connection	Enables the Informatica domain to communicate with the HBase master server or region server that uses Kerberos authentication.

Property	Description
HBase Master Principal	Service Principal Name (SPN) of the HBase master server. Enables the ZooKeeper server to communicate with an HBase master server that uses Kerberos authentication. Enter a string in the following format: `hbase/<domain.name>@<YOUR-REALM>` Where: - domain.name is the domain name of the machine that hosts the HBase master server. - YOUR-REALM is the Kerberos realm.
HBase Region Server Principal	Service Principal Name (SPN) of the HBase region server. Enables the ZooKeeper server to communicate with an HBase region server that uses Kerberos authentication. Enter a string in the following format: `hbase_rs/<domain.name>@<YOUR-REALM>` Where: - domain.name is the domain name of the machine that hosts the HBase master server. - YOUR-REALM is the Kerberos realm.

HDFS Connection Properties

Use a Hadoop File System (HDFS) connection to access data in the Hadoop cluster. The HDFS connection is a file system type connection. You can create and manage an HDFS connection in the Administrator tool, Analyst tool, or the Developer tool. HDFS connection properties are case sensitive unless otherwise noted.

Note: The order of the connection properties might vary depending on the tool where you view them.

The following table describes HDFS connection properties:

Property	Description
Name	Name of the connection. The name is not case sensitive and must be unique within the domain. The name cannot exceed 128 characters, contain spaces, or contain the following special characters: `~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /`
ID	String that the Data Integration Service uses to identify the connection. The ID is not case sensitive. It must be 255 characters or less and must be unique in the domain. You cannot change this property after you create the connection. Default value is the connection name.
Description	The description of the connection. The description cannot exceed 765 characters.
Location	The domain where you want to create the connection. Not valid for the Analyst tool.
Type	The connection type. Default is Hadoop File System.

Property	Description
User Name	User name to access HDFS.

NameNode URI	The URI to access the Hadoop Distributed File System.
	Use the following connection URI:
	`hdfs://<node name>:<port>`
	Where
	- <node name> is the host name or IP address of the NameNode.
	- <port> is the port on which the NameNode listens for remote procedure calls (RPC).
	For example, enter: `hdfs://myhostname:8020/`
	You can also get the Default File System URI property from core-site.xml located in the following directory on the Hadoop cluster: `/etc/hadoop/conf/`
	Use the value from the `fs.defaultFS` property found in core-site.xml.
	For example, use the following value:
	`<property>`
	`<name>fs.defaultFS</name>`
	`<value>hdfs://localhost:8020</value>`
	`</property>`
	The Azure HDInsight File System default file system can be Windows Azure Storage Blob (WASB) or Azure Data Lake Store (ADLS).
	If the cluster uses WASB storage, use the following string to specify the URI:
	`wasb://<container_name>@<account_name>.blob.core.windows.net/<path>`
	where:
	- `<container_name>` identifies a specific Azure Storage Blob container.
	Note: `<container_name>` is optional.
	- `<account_name>` identifies the Azure Storage Blob object.
	Example:
	`wasb://infabdmoffering1storage.blob.core.windows.net/infabdmoffering1cluster/mr-history`
	If the cluster uses ADLS storage, use the following format to specify the URI: `adl://home`
	The following is the `fs.defaultFS` property as it appears in `hdfs-site.xml`:
	`<property>fs.defaultFS</property>`
	`<value>adl://home</value>`

Hive Connection Properties

Use the Hive connection to access Hive data. A Hive connection is a database type connection. You can create and manage a Hive connection in the Administrator tool, Analyst tool, or the Developer tool. Hive connection properties are case sensitive unless otherwise noted.

Note: The order of the connection properties might vary depending on the tool where you view them.

The following table describes Hive connection properties:

Property	Description
Name	The name of the connection. The name is not case sensitive and must be unique within the domain. You can change this property after you create the connection. The name cannot exceed 128 characters, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
ID	String that the Data Integration Service uses to identify the connection. The ID is not case sensitive. It must be 255 characters or less and must be unique in the domain. You cannot change this property after you create the connection. Default value is the connection name.
Description	The description of the connection. The description cannot exceed 4000 characters.
Location	The domain where you want to create the connection. Not valid for the Analyst tool.
Type	The connection type. Select Hive.
Connection Modes	Hive connection mode. Select at least one of the following options: - Access Hive as a source or target. Select this option if you want to use Hive as a source or a target. - Use Hive to run mappings in Hadoop cluster. Select this option if you want to use the Hive driver to run mappings in the Hadoop cluster.

Property	Description
User Name	User name of the user that the Data Integration Service impersonates to run mappings on a Hadoop cluster. The user name depends on the JDBC connection string that you specify in the Metadata Connection String or Data Access Connection String for the native environment. If the Hadoop cluster runs Hortonworks HDP, you must provide a user name. If you use Tez to run mappings, you must provide the user account for the Data Integration Service. If you do not use Tez to run mappings, you can use an impersonation user account. If the Hadoop cluster uses Kerberos authentication, the principal name for the JDBC connection string and the user name must be the same. Otherwise, the user name depends on the behavior of the JDBC driver. With Hive JDBC driver, you can specify a user name in many ways and the user name can become a part of the JDBC URL. If the Hadoop cluster does not use Kerberos authentication, the user name depends on the behavior of the JDBC driver. If you do not specify a user name, the Hadoop cluster authenticates jobs based on the following criteria: - The Hadoop cluster does not use Kerberos authentication. It authenticates jobs based on the operating system profile user name of the machine that runs the Data Integration Service. - The Hadoop cluster uses Kerberos authentication. It authenticates jobs based on the SPN of the Data Integration Service.
Common Attributes to Both the Modes: Environment SQL	SQL commands to set the Hadoop environment. In native environment type, the Data Integration Service executes the environment SQL each time it creates a connection to a Hive metastore. If you use the Hive connection to run profiles in the Hadoop cluster, the Data Integration Service executes the environment SQL at the beginning of each Hive session. The following rules and guidelines apply to the usage of environment SQL in both connection modes: - Use the environment SQL to specify Hive queries. - Use the environment SQL to set the classpath for Hive user-defined functions and then use environment SQL or PreSQL to specify the Hive user-defined functions. You cannot use PreSQL in the data object properties to specify the classpath. The path must be the fully qualified path to the JAR files used for user-defined functions. Set the parameter hive.aux.jars.path with all the entries in infapdo.aux.jars.path and the path to the JAR files for user-defined functions. - You can use environment SQL to define Hadoop or Hive parameters that you want to use in the PreSQL commands or in custom queries. - If you use multiple values for the Environment SQL property, ensure that there is no space between the values. The following sample text shows two values that can be used for the Environment SQL: `set hive.execution.engine='tez';set hive.exec.dynamic.partition.mode='nonstrict';` If you use the Hive connection to run profiles in the Hadoop cluster, the Data Integration service executes only the environment SQL of the Hive connection. If the Hive sources and targets are on different clusters, the Data Integration Service does not execute the different environment SQL commands for the connections of the Hive source or target.

Properties to Access Hive as Source or Target

The following table describes the connection properties that you configure to access Hive as a source or target:

Property	Description
Metadata Connection String	The JDBC connection URI used to access the metadata from the Hadoop server. You can use PowerExchange for Hive to communicate with a HiveServer service or HiveServer2 service. To connect to HiveServer2, specify the connection string in the following format: `jdbc:hive2://<hostname>:<port>/<db>;transportMode=<mode>` Where - <hostname> is name or IP address of the machine on which HiveServer2 runs. - <port> is the port number on which HiveServer2 listens. - <db> is the database to which you want to connect. If you do not provide the database name, the Data Integration Service uses the default database details. - <mode> is the value of the hive.server2.transport.mode property in the Hive tab of the Ambari tool.
Bypass Hive JDBC Server	JDBC driver mode. Select the check box to use the embedded JDBC driver mode. To use the JDBC embedded mode, perform the following tasks: - Verify that Hive client and Informatica services are installed on the same machine. - Configure the Hive connection properties to run mappings in the Hadoop cluster. If you choose the non-embedded mode, you must configure the Data Access Connection String. Informatica recommends that you use the JDBC embedded mode.
Observe Fine Grained SQL Authorization	When you select the option to observe fine-grained SQL authentication in a Hive source, the mapping observes row and column-level restrictions on data access. If you do not select the option, the Blaze run-time engine ignores the restrictions, and results include restricted data.
Data Access Connection String	The JDBC connection URI used to access data from the Hadoop server. You can use PowerExchange for Hive to communicate with a HiveServer service or HiveServer2 service. To connect to HiveServer2, specify the connection string in the following format: `jdbc:hive2://<hostname>:<port>/<db>;transportMode=<mode>` Where - <hostname> is name or IP address of the machine on which HiveServer2 runs. - <port> is the port number on which HiveServer2 listens. - <db> is the database to which you want to connect. If you do not provide the database name, the Data Integration Service uses the default database details. - <mode> is the value of the hive.server2.transport.mode property in the Hive tab of the Ambari tool.

Properties to Run Mappings in Hadoop Cluster

The following table describes the Hive connection properties that you configure when you want to use the Hive connection to run Informatica mappings in the Hadoop cluster:

Property	Description
Database Name	Namespace for tables. Use the name `default` for tables that do not have a specified database name.

Default FS URI	The URI to access the Hadoop Distributed File System.
	Use the connection URI that matches the storage type. The storage type is configured for the cluster in the fs.defaultFS property.
	If the cluster uses HDFS storage, use the following string to specify the URI:
	`hdfs://<cluster_name>`
	Example:
	`hdfs://my-cluster`
	If the cluster is enabled with Namenode High Availability, you can use the value of the dfs.nameservices property from the hdfs-site.xml file for the Default FS URI property.
	Example:
	Where the following is the dfs.nameservices property as it appears in hdfs-site.xml:
	`<property>`
	`  <name>dfs.nameservices</name>`
	`  <value>infaqaha</value>`
	`  </property>`
	The value to use for Default FS URI is:
	`hdfs://infaqaha`
	The Azure HDInsight File System default file system can be Windows Azure Storage Blob (WASB) or Azure Data Lake Store (ADLS).
	If the cluster uses WASB storage, use the following string to specify the URI:
	`wasb://<container_name>@<account_name>.blob.core.windows.net/<path>`
	where:
	- `<container_name>` identifies a specific Azure Storage Blob container.
	Note: `<container_name>` is optional.
	- `<account_name>` identifies the Azure Storage Blob object.
	Example:
	`wasb://infabdmoffering1storage.blob.core.windows.net/infabdmoffering1cluster/mr-history`
	If the cluster uses ADLS storage, use the following format to specify the URI:
	`adl://home`
	The following is the `fs.defaultFS` property as it appears in `hdfs-site.xml`:
	`<property>fs.defaultFS</property>`
	`<value>adl://home</value>`

Property	Description

Yarn Resource Manager URI	The service within Hadoop that submits the MapReduce tasks to specific nodes in the cluster.
	For HDInsight 3.3 with YARN, use the following format:
	`<hostname>:<port>`
	Where
	- <hostname> is the host name or IP address of the JobTracker or Yarn resource manager.
	- <port> is the port on which the JobTracker or Yarn resource manager listens for remote procedure calls (RPC).
	Use the value specified by `yarn.resourcemanager.address` in `yarn-site.xml`. You can find `yarn-site.xml` in the following directory on the NameNode: `/etc/hive/<version>/0/`.
	For HDInsight 3.3 with MapReduce 2, use the following URI:
	`hdfs://host:port`

JobTracker/Yarn Resource Manager URI	The service within Hadoop that submits the MapReduce tasks to specific nodes in the cluster. Use the following format: `<hostname>:<port>` Where - <hostname> is the host name or IP address of the JobTracker or Yarn resource manager. - <port> is the port on which the JobTracker or Yarn resource manager listens for remote procedure calls (RPC). If the cluster uses MapR with YARN, use the value specified in the `yarn.resourcemanager.address` property in yarn-site.xml. You can find `yarn-site.xml` in the following directory on the NameNode of the cluster: `/opt/mapr/hadoop/hadoop-2.5.1/etc/hadoop`. MapR with MapReduce 1 supports a highly available JobTracker. If you are using MapR distribution, define the JobTracker URI in the following format: `maprfs:///`
Hive Warehouse Directory on HDFS	The absolute HDFS file path of the default database for the warehouse that is local to the cluster. For example, the following file path specifies a local warehouse: `/user/hive/warehouse` If the Metastore Execution Mode is remote, then the file path must match the file path specified by the Hive Metastore Service on the hadoop cluster. Use the value specified for the `hive.metastore.warehouse.dir` property in `hive-site.xml`. You can find `yarn-site.xml` in the following directory on the node that runs HiveServer2: `/etc/hive/<version>/0/`.

Property	Description

Advanced Hive/Hadoop Properties	Configures or overrides Hive or Hadoop cluster properties in hive-site.xml on the machine on which the Data Integration Service runs. You can specify multiple properties.
	Use the following format:
	`<property1>=<value>`
	Where
	- <property1> is a Hive or Hadoop property in hive-site.xml.
	- <value> is the value of the Hive or Hadoop property.
	To specify multiple properties use `&:` as the property separator.
	The maximum length for the format is 1 MB.
	If you enter a required property for a Hive connection, it overrides the property that you configure in the Advanced Hive/Hadoop Properties.
	The Data Integration Service adds or sets these properties for each map-reduce job. You can verify these properties in the JobConf of each mapper and reducer job. Access the JobConf of each job from the Jobtracker URL under each map-reduce job.
	The Data Integration Service writes messages for these properties to the Data Integration Service logs. The Data Integration Service must have the log tracing level set to log each row or have the log tracing level set to verbose initialization tracing.
	For example, specify the following properties to control and limit the number of reducers to run a mapping job:
	`mapred.reduce.tasks=2&:hive.exec.reducers.max=10`

Temporary Table Compression Codec	Hadoop compression library for a compression codec class name.
Codec Class Name	Codec class name that enables data compression and improves performance on temporary staging tables.
Metastore Execution Mode	Controls whether to connect to a remote metastore or a local metastore. By default, local is selected. For a local metastore, you must specify the Metastore Database URI, Driver, Username, and Password. For a remote metastore, you must specify only the Remote Metastore URI.

Metastore Database URI	The JDBC connection URI used to access the data store in a local metastore setup. Use the following connection URI:
	`jdbc:<datastore type>://<node name>:<port>/<database name>`
	where
	- <node name> is the host name or IP address of the data store.
	- <data store type> is the type of the data store.
	- <port> is the port on which the data store listens for remote procedure calls (RPC).
	- <database name> is the name of the database.
	For example, the following URI specifies a local metastore that uses MySQL as a data store:
	`jdbc:mysql://hostname23:3306/metastore`
	Use the value specified for the `javax.jdo.option.ConnectionURL` property in `hive-site.xml`. You can find `hive-site.xml` in the following directory on the node that runs HiveServer2: `/etc/hive/<version>/0/hive-site.xml`.

Property	Description
Metastore Database Driver	Driver class name for the JDBC data store. For example, the following class name specifies a MySQL driver: Use the value specified for the `javax.jdo.option.ConnectionDriverName` property in `hive-site.xml`. You can find `hive-site.xml` in the following directory on the node that runs HiveServer2: `/etc/hive/<version>/0/hive-site.xml`.
Metastore Database Username	The metastore database user name. Use the value specified for the `javax.jdo.option.ConnectionUserName` property in `hive-site.xml`. You can find `hive-site.xml` in the following directory on the node that runs HiveServer2: `/etc/hive/<version>/0/hive-site.xml`.
Metastore Database Password	Required if the Metastore Execution Mode is set to local. The password for the metastore user name. Use the value specified for the `javax.jdo.option.ConnectionPassword` property in `hive-site.xml`. You can find `hive-site.xml` in the following directory on the node that runs HiveServer2: `/etc/hive/<version>/0/hive-site.xml`.
Remote Metastore URI	The metastore URI used to access metadata in a remote metastore setup. For a remote metastore, you must specify the Thrift server details. Use the following connection URI: `thrift://<hostname>:<port>` Where - <hostname> is name or IP address of the Thrift metastore server. - <port> is the port on which the Thrift server is listening. Use the value specified for the `hive.metastore.uris` property in `hive-site.xml`. You can find `hive-site.xml` in the following directory on the node that runs HiveServer2: `/etc/hive/<version>/0/hive-site.xml`.
Hive Connection String	The JDBC connection URI used to access the metadata from the Hadoop server. You can use PowerExchange for Hive to communicate with a HiveServer service or HiveServer2 service. To connect to HiveServer2, specify the connection string in the following format: `jdbc:hive2://<hostname>:<port>/<db>;transportMode=<mode>` Where - <hostname> is name or IP address of the machine on which HiveServer2 runs. - <port> is the port number on which HiveServer2 listens. - <db> is the database to which you want to connect. If you do not provide the database name, the Data Integration Service uses the default database details. - <mode> is the value of the hive.server2.transport.mode property in the Hive tab of the Ambari tool.

JDBC Connection Properties

You can use a JDBC connection to access tables in a database. You can create and manage a JDBC connection in the Administrator tool, the Developer tool, or the Analyst tool.

Note: The order of the connection properties might vary depending on the tool where you view them.

The following table describes JDBC connection properties:

Property	Description
Database Type	The database type.
Name	Name of the connection. The name is not case sensitive and must be unique within the domain. The name cannot exceed 128 characters, contain spaces, or contain the following special characters: `~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /`
ID	String that the Data Integration Service uses to identify the connection. The ID is not case sensitive. It must be 255 characters or less and must be unique in the domain. You cannot change this property after you create the connection. Default value is the connection name.
Description	The description of the connection. The description cannot exceed 765 characters.
User Name	The database user name.
Password	The password for the database user name.

JDBC Driver Class Name	Name of the JDBC driver class.
	The following list provides the driver class name that you can enter for the applicable database type:
	- DataDirect JDBC driver class name for Oracle:
	`com.informatica.jdbc.oracle.OracleDriver`
	- DataDirect JDBC driver class name for IBM DB2:
	`com.informatica.jdbc.db2.DB2Driver`
	- DataDirect JDBC driver class name for Microsoft SQL Server:
	`com.informatica.jdbc.sqlserver.SQLServerDriver`
	- DataDirect JDBC driver class name for Sybase ASE:
	`com.informatica.jdbc.sybase.SybaseDriver`
	- DataDirect JDBC driver class name for Informix:
	`com.informatica.jdbc.informix.InformixDriver`
	- DataDirect JDBC driver class name for MySQL:
	`com.informatica.jdbc.mysql.MySQLDriver`
	For more information about which driver class to use with specific databases, see the vendor documentation.

Connection String	Connection string to connect to the database. Use the following connection string: `jdbc:<subprotocol>:<subname>`
Environment SQL	Optional. Enter SQL commands to set the database environment when you connect to the database. The Data Integration Service executes the connection environment SQL each time it connects to the database. **Note:** If you enable Sqoop, Sqoop ignores this property.
Transaction SQL	Optional. Enter SQL commands to set the database environment when you connect to the database. The Data Integration Service executes the transaction environment SQL at the beginning of each transaction. **Note:** If you enable Sqoop, Sqoop ignores this property.
SQL Identifier Character	Type of character that the database uses to enclose delimited identifiers in SQL queries. The available characters depend on the database type. Select (None) if the database uses regular identifiers. When the Data Integration Service generates SQL queries, the service does not place delimited characters around any identifiers. Select a character if the database uses delimited identifiers. When the Data Integration Service generates SQL queries, the service encloses delimited identifiers within this character. **Note:** If you enable Sqoop, Sqoop ignores this property.

Property	Description
Support Mixed-case Identifiers	Enable if the database uses case-sensitive identifiers. When enabled, the Data Integration Service encloses all identifiers within the character selected for the **SQL Identifier Character** property. When the **SQL Identifier Character** property is set to none, the **Support Mixed-case Identifiers** property is disabled. **Note:** If you enable Sqoop, Sqoop honors this property when you generate and execute a DDL script to create or replace a target at run time. In all other scenarios, Sqoop ignores this property.
Use Sqoop Connector	Enables Sqoop connectivity for the data object that uses the JDBC connection. The Data Integration Service runs the mapping in the Hadoop run-time environment through Sqoop. You can configure Sqoop connectivity for relational data objects, customized data objects, and logical data objects that are based on a JDBC-compliant database. Select **Sqoop v1.x** to enable Sqoop connectivity. Default is **None**.

Sqoop Arguments	Enter the arguments that Sqoop must use to connect to the database. Separate multiple arguments with a space.
	To read data from or write data to Teradata through Teradata Connector for Hadoop (TDCH) specialized connectors for Sqoop, define the TDCH connection factory class in the Sqoop arguments. The connection factory class varies based on the TDCH Sqoop Connector that you want to use.
	- To use the Cloudera Connector Powered by Teradata, configure the following Sqoop argument:
	`–Dsqoop.connection.factories=com.cloudera.connector.teradata.TeradataManagerFactory`
	- To use the Hortonworks Connector for Teradata (powered by the Teradata Connector for Hadoop), configure the following Sqoop argument:
	`–Dsqoop.connection.factories=org.apache.sqoop.teradata.TeradataManagerFactory`
	If you do not enter Sqoop arguments, the Data Integration Service constructs the Sqoop command based on the JDBC connection properties.
	On the Hive engine, to run a column profile on a relational data object that uses Sqoop, set the Sqoop argument m to 1. Use the following syntax:
	`-m 1`

APPENDIX B

Port Reference

This appendix includes the following topics:

- Informatica Domain Port Reference, 208
- Application Service Port Reference, 209
- Hadoop Distribution Port Reference, 210
- Hadoop Cluster Port Reference, 214

Informatica Domain Port Reference

Informatica domain services have unique ports.

Port	Description
Service Manager port	Port number used by the Service Manager on the node. The Service Manager listens for incoming connection requests on this port. Client applications use this port to communicate with the services in the domain. The Informatica command line programs use this port to communicate to the domain. This is also the port for the SQL data service JDBC/ODBC driver. Default is 6006.
Service Manager Shutdown port	Port number that controls server shutdown for the domain Service Manager. The Service Manager listens for shutdown commands on this port. Default is 6007.
Informatica Administrator port	Port number used by Informatica Administrator. Default is 6008.
Informatica Administrator HTTPS port	No default port. Enter the required port number when you create the service. Setting this port to 0 disables an HTTPS connection to the Administrator tool.
Informatica Administrator shutdown port	Port number that controls server shutdown for Informatica Administrator. Informatica Administrator listens for shutdown commands on this port. Default is 6009.
Minimum port number	Lowest port number in the range of dynamic port numbers that can be assigned to the application service processes that run on this node. Default is 6014.
Maximum port number	Highest port number in the range of dynamic port numbers that can be assigned to the application service processes that run on this node. Default is 6114.

Application Service Port Reference

Some of the Informatica application services require unique ports.

Analyst Service Port Reference

The following table lists the default ports associated with the Analyst Service:

Port Name	Default Port Number
Analyst Service (HTTP)	8085
Analyst Service (HTTPS)	No default port.
Analyst Service (Staging database)	No default port.

Content Management Service Port Reference

The following table lists the default ports associated with the Content Management Service:

Port Name	Default Port Number
Content Management Service (HTTP)	8105
Content Management Service (HTTPS)	No default port.

Data Integration Service Port Reference

The following table lists the default ports associated with the Data Integration Service:

Type	Default Port
Data Integration Service (HTTP proxy)	8080
Data Integration Service (HTTP)	8095
Data Integration Service (HTTPS)	No default port. Enter the required port number when you create the service.
Profiling Warehouse database	No default port. Enter the database port number.
Human Task database	No default port. Enter the database port number.

Metadata Manager Service Port Reference

The following table lists the default ports associated with the Metadata Manager Service:

Port Name	Default Port Number
Metadata Manager Service (HTTP)	10250
Metadata Manager Service (HTTPS)	No default port.

Hadoop Distribution Port Reference

A Hadoop distribution requires some unique port numbers.

IBM BigInsights Port Reference

The following table lists the default ports associated with IBM BigInsights:

Hadoop Component	Port
HBase master	60000
HBase master web	60010
HBase region server	60020
HDFS read/write	50010, 50020
Hive metastore	9933
HiveServer	10000
JobTracker	9001
JobTracker web manager	50030
MapReduce Application Master	50100-50200
MapReduce JobHistory server	10020
MapReduce JobHistory server webapp	19888
MySQL	3306
NameNode	9000
ResourceManager	8032
ResourceManager webapp	8088

Hadoop Component	Port
ResourceTracker	8031
Scheduler address	8030
Shuffle HTTP	13562
TaskTracker web management	50060
ZooKeeper	2181

Cloudera Port Reference

The following table lists the ports associated with Cloudera:

Hadoop Component	Port
Cloudera Manager	7180
HBase master	60000
HBase master web	60010
HBase region server	60020
HDFS read/write	50010, 50020
Hive metastore	9083
HiveServer	10000
JobTracker	8021
JobTracker web manager	50030
MapReduce Application Master	50100-50200
MapReduce JobHistory server	10020
MapReduce JobHistory server webapp	19888
MySQL	3306
NameNode	8020
ResourceManager	8050
ResourceManager webapp	8088
ResourceTracker	8031
Scheduler address	8030

Hadoop Component	Port
Shuffle HTTP	13562
TaskTracker web management	50060
ZooKeeper	2181

HDInsight Port Reference

The following table lists the ports associated with HDInsight:

Hadoop Component	Port
HBase master	60000
HBase master web	60010
HBase region server	60020
HDFS read/write	50010, 50020
Hive metastore	9933
HiveServer	10000
JobTracker	9001
JobTracker web manager	50030
MapReduce Application Master	50100-50200
MapReduce JobHistory server	10020
MapReduce JobHistory server webapp	19888
MySQL	3306
NameNode	9000
ResourceManager	8032
ResourceManager webapp	8088
ResourceTracker	8031
Scheduler address	8030
Shuffle HTTP	13562
TaskTracker web management	50060
ZooKeeper	2181

Hortonworks Port Reference

The following table lists the ports associated with Hortonworks:

Hadoop Component	Port
HBase master	60000
HBase master web	60010
HBase region server	60020
HDFS read/write	50010, 50020
Hive metastore	9933
HiveServer	10000
JobTracker	8021
JobTracker web manager	50030
MapReduce Application Master	50100-50200
MapReduce JobHistory server	10020
MapReduce JobHistory server webapp	19888
MySQL	3306
NameNode	8020
ResourceManager	8032
ResourceManager webapp	8088
ResourceTracker	8031
Scheduler address	8030
Shuffle HTTP	13562
TaskTracker web management	50060
ZooKeeper	2181

MapR Port Reference

The following table lists the ports associated with MapR:

Hadoop Component	Port	Notes
CLDB	7222	
CLDB JMX monitor	7220	
CLDB web port	7221	
HBase master	16000	
HBase master web	16010	
HBase region server	16020	
Hive metastore	9083	
HiveServer	10000	
JobTracker	9001	
JobTracker web manager	50030	
MapReduce Application Master	50100-50200	
MapReduce JobHistory server	10020	YARN only
MapReduce JobHistory server web app	19888	YARN only
MySQL	3306	Required if you use MySQL.
ResourceManager	8032	YARN only
ResourceManager web app	8088	YARN only
Secure Resource Manager web app (HTTPS)	8090	
ResourceTracker	8031	YARN only
Scheduler address	8030	YARN only
Shuffle HTTP	13562	
TaskTracker web management	50060	
ZooKeeper	5181	

Hadoop Cluster Port Reference

A Hadoop cluster requires some unique port numbers.

Blaze Engine Port Reference

The Blaze engine uses ports for the Blaze Job Monitor and the Blaze services.

Blaze Job Monitor

The following table lists the ports for the Blaze Job Monitor:

Port Name	Default Port Number
HTTP	9080
JSF	9090

Blaze Services

The following table lists the ports for Blaze services:

Port Name	Default Port Number
Minimum and Maximum Port Range for Hadoop Connection	12300 - 12600

Developer Tool Port Reference

The following table lists the ports for the Developer tool:

Port Name	Default Port Number
CLDB	7222 - MapR only
HBase master	60000
HBase region server	60200
HiveServer2 thrift	10000
NameNode RPC	8020 - except MapR
ZooKeeper client	2181, 5181

INDEX

A

Amazon EMR
- cluster creation wizard 38
- create connections 46
- Data Integration Service configuration 39
- Developer tool tasks 46
- Hive access 40, 46
- installation overview 33
- configuration files on the domain 41
- configuration files on the Hadoop environment 44
- S3 bucket access policy 46

architecture
- Big Data Management 16
- Hadoop environment 16

Azure HDInsight
- Developer tool tasks 72
- distribution package 53
- install on a single node 54
- install using NFS 54
- installation overview 53
- prerequisites 26
- running configuration manager 56
- configuration files on the domain 69
- configuration files on the Hadoop environment 66

B

big data
- application services 17
- data lineage 18
- repositories 17

Big Data Management
- integration with Informatica products 18
- cluster installation 54
- Data Quality 20

Blaze engine
- create a user account 26
- port requirements 25
- connection properties 187
- directories to create 26

bootstrap script
- creating for Amazon EMR cluster 37

C

Cloudera CDH
- distribution package 78
- hadoopEnv.properties configuration 95
- installation 78
- running configuration manager 80
- configuration files on the domain 94
- configuration files on the Hadoop environment 91
- yarn-site.xml configuration 93

cluster creation wizard
- create the bootstrap script 37

cluster installation
- primary NameNode 34, 54, 137, 163
- using the HadoopDataNodes file 55

component architecture
- clients and tools 17

configuration
- Hadoop Configuration Manager 57, 112, 141
- through SSH 59, 83, 114, 143

configuration files
- Developer tool configuration 99, 129, 180

connections
- create in Amazon EMR 46
- properties 187
- HBase 186
- HDFS 186
- Hive 186
- JDBC 186

core-site.xml
- configuration for Cloudera 91
- configuration on Hadoop HDP 66, 122, 151
- configuration for Amazon EMR 44
- configuration for MapR 172, 173

D

Data Discovery
- description 20

Data Integration Service
- configuration for Amazon EMR 39
- configuration for MapR 177, 178

data lakes
- description 19

Data Quality
- reference data 20

Data Replication
- description 20

data streaming
- description 21

Developer tool
- configuration 99, 129, 180

developerCore.ini
- Azure HDInsight 72
- updating 47, 155
- MapR 180

distribution package
- download 33, 136, 162
- download for Azure HDInsight 53
- download for Cloudera CDH 78
- download for Hortonworks HDP 106

domain configuration
- Hadoop Configuration Manager 57, 81, 112, 141

dynamic partitioning
- configuration 43, 71, 96, 126, 154, 171

G

Grid Manager 98, 128

H

Hadoop 186
Hadoop Configuration Manager
  Cloudera configuration 81
  overview 56, 80, 111, 140, 166
  properties for silent mode 61, 85, 116, 145
  running for Azure HDInsight 56
  running for Cloudera CDH 80
  running for Hortonworks HDP 111
  running for IBM BigInsights 140
  running for MapR 166
  running in silent mode 65, 89, 120, 149
  silent mode 61, 85, 116, 145
Hadoop distributions
  supported 24
hadoopEnv.properties
  configure 69
  configuring on Cloudera CDH 95
  configuring on Hortonworks HDP 125
  configuring on IBM BigInsights 153
  configuring on MapR 170
  configure on Amazon EMR 41
HBase connections
  properties 196
HDFS connections
  properties 197
HDFS staging directory 28
high availability
  configuration on Developer tool 99, 129, 180
Hive access
  for Amazon EMR 40, 46
  Amazon EMR 44
  Amazon S3 buckets 172
  MapR 173
Hive buckets
  access 91
Hive connections
  properties 198
Hive logs
  user permissions 26
Hive pushdown
  connection properties 187
Hive warehouse directory
  user permissions 26
hive-site.xml
  configuration 43, 71, 96, 126, 154, 171
  configuration on Hadoop HDP 123, 174
  configuration for MapR 123, 174
Hortonworks HDP
  hadoopEnv.properties 125
  install troubleshooting 110
  installation overview 106
  prerequisites 25
  running configuration manager 111
  configuration files on the domain 125
  configuration files on the Hadoop environment 122

I

IBM BigInsights
  hadoopEnv.properties 153
IBM BigInsights *(continued)*
  installation overview 136
  running configuration manager 140
  configuration files on the domain 152
  configuration files on the Hadoop environment 150
Informatica adapters
  integration with Big Data Management 18
installation
  for Hortonworks 107
  in an Ambari Stack 107
  on a single node 34, 54, 137, 163
  on Cloudera CDH 78
  cluster creation wizard 38
  on a cluster using SCP 34, 137, 163
  on a single node 34, 54, 137, 163
  overview 15
  process 15
  using NFS protocol 35, 138, 164
  using the HadoopDataNodes file 36, 139, 165
  with cluster creation wizard on Amazon EMR 36
installation requirements
  prerequisites 24

J

JDBC
  Sqoop connectivity 40, 73, 150, 169
JDBC connections
  properties 205
  Sqoop configuration 205

M

MapR
  Developer tool tasks 180
  hadoopEnv.properties 170
  installation overview 162
  running configuration manager 166
  configuration files on the domain 169
  configuration files on the Hadoop environment 173
  Data Integration Service configuration 177, 178
  Spark engine 170
  tickets 175
mapr-cluster.conf 180
mapred-site.xml
  configuration on Cloudera 92
MDM Big Data Relationship Management
  description 20

P

permissions
  Blaze engine user 26
  Hive warehouse directory 26
ports
  Amazon EMR requirements 25
  Azure HDInsight requirements 25
  Blaze engine requirements 25
prerequisites
  Azure HDInsight 26
  create directories for the Blaze engine 26
  gather connection information 29
  gather domain information 29
  gather information to configure through SSH 30
  Hortonworks HDP 25

prerequisites *(continued)*
overview 22
record install information 28
verify Hadoop distribution 24
verify system requirements 23
installer 24
verify product installations 23
primary NameNode
SCP protocol 34, 54, 137, 163
product installations
prerequisites 23
properties file
silent configuration 61, 85, 116, 145

R

resource allocation
Amazon EMR 45

S

security certificates 98, 128
silent mode
Hadoop Configuration Manager 61, 85, 116, 145
Spark
dynamic resource allocation 97
Spark deploy mode
Hadoop connection properties 187
Spark engine
configuring on MapR 170
hadoopEnv.properties configuration 41, 69
connection properties 187
hadoopEnv.properties configuration 41, 69
Spark engine performance
configuration 67, 151
Spark Event Log directory
Hadoop connection properties 187
Spark execution parameters
Hadoop connection properties 187
Spark HDFS staging directory
Hadoop connection properties 187
Sqoop
hadoopEnv.properties 69, 95, 125, 153, 170
JDBC drivers 40, 73, 121, 150, 169
mapred-site.xml configuration 92
user impersonation on Hadoop HDP 66, 122, 151
hadoopEnv.properties 69, 95, 125, 153, 170
hadoopEnv.properties configuration 41
Sqoop *(continued)*
JDBC drivers 40, 73, 121, 150, 169
Sqoop configuration
copying JDBC driver jar files 100
SSH
Hadoop Configuration Manager 59, 83, 114, 143
SSL 98, 128
staging directory
HDFS 28
system requirements
prerequisites 23

T

truststore files 98, 128

U

uninstall
Ambari stack 133
from Cloudera 104
from Hortonworks 133
prerequisites 50, 75, 159, 184
steps 50, 75, 159, 184
user accounts
MapR 175
user permissions
Hive logs 26

V

virtual memory limits
yarn-site.xml 97, 123, 174

Y

yarn-site.xml
configuration 67, 97, 151
configuration for Amazon EMR 43, 45
configuration for Cloudera CDH 93
configuration for Hortonworks HDP 123
configuration for IBM BigInsights 155
configuration for MapR 174
configuration for Amazon EMR 43, 45