

Control-M for Hadoop

With Control-M for Hadoop, you can simplify and automate Hadoop batch processing for faster implementation and more accurate big-data analytics. It takes the complexity out of Hadoop management, accelerating implementation and delivering more accurate results.

KEY FEATURES

- Control-M for Hadoop simplifies workflow creation and management.
- Familiar interface – Manage Hadoop workloads through a simple drag-and-drop interface that provides universal visibility and control for all enterprise workloads
- Flexible ecosystem support – Apache® Spark™ , Apache® Oozie™ , Apache® DistCP®, Apache® Impala, and other jobs can all run through Control-M for Hadoop
- Increased availability – The co-location requirement for IBM® CICS® /IBM® DB2®, CICS/ IBM® IMST™ DB, IMS TM/DB2 is no longer an issue, allowing data access calls to be redirected upon a DB2 or IMS system failure and increasing availability
- Improved reliability – Detects and prevents potential problems with forecasting features and predictive analytics; restarts can be attempted automatically without requiring human intervention

KEY BENEFITS

- Makes implementation fast and accurate by replacing manual scripting with automated workflow management and data integration, dramatically shortening development time, preventing coding errors, and reducing time-to-value
- Improves business focus instead of spending time building connectors and coordinating multiple tools.

Product Overview

Control-M for Hadoop

Supports all Apache
Hadoop 2.x Distributions

cloudera

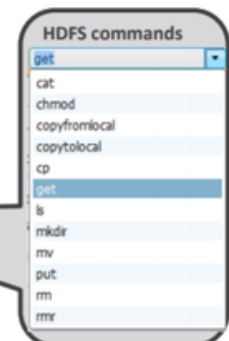
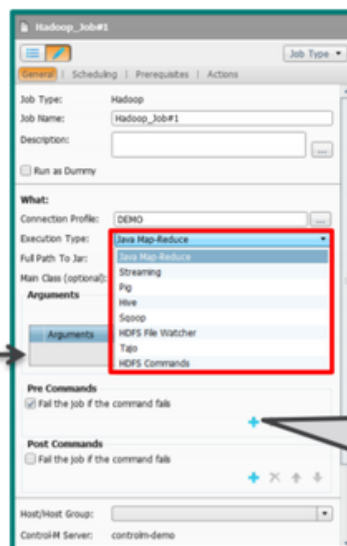
MAPR

Hortonworks

Pivotal

BigInsights

Program parameters



Implementation Steps

- Agent 64 bit: Install Control-M for Hadoop 8.0.00 + Fix Pack 1 + latest patch available
- WLA Client/Server: Install Hadoop plugin and update forms
- CCM: Configuration Management + Connection Profile Management

Kerberos Configuration

- Create Kerberos Principal for the Control-M/Agent keytab file
 - `kadmin -q "addprinc -randkey ctmagent@LOCALDOMAIN.COM"`
 - `kadmin -q "ktadd -k /home/ctmagent/ctmagent.keytab ctmagent@LOCALDOMAIN.COM"`
- Using Impersonation
 - Create Kerberos Principal and Kerberos keytab file for the target user
 - Create `.k5login` file in the target user's home directory.
 - Configure Control-M agent as a proxy user
 - Add to the `core-site.xml` file the following properties

```
<property>
  <name>hadoop.proxyuser:ctmagent.hosts</name>
  <value>*</value>
</property>
<property>
  <name>hadoop.proxyuser:ctmagent.groups</name>
  <value>*</value>
</property>
```
 - Not required in MapR distribution for Hadoop

Configuration

Control-M for Hadoop - Configuration Management

Kerberos:

CTM Agent's principal:

CTM Agent's keytab file path:

Ticket life time (minutes):

Hadoop Streaming:

Full path to streaming jar:

Update - IMPERSONATION Connection Profile

Account Name:

Hadoop General ☒ Sqoop ☐ Hive ☐ Tajo

Authentication

Run as user: (Kerberos: Use Principal)

User's Keytab file path:

☒ **Use Pig**

Pig binary path:

Job Details

Hadoop_Job#1

General | Scheduling | Prerequisites | Actions

Job Type: Hadoop

Job Name: Hadoop_Job#1

Description:

☐ Run as Dummy

What:

Connection Profile: DEMO

Execution Type: **Java Map-Reduce**

Full Path To Jar: **Java Map-Reduce**

Main Class (optional): Streaming

Arguments

☐ Arguments

Pre Commands

☒ Fail the job if the command fails

Post Commands

☐ Fail the job if the command fails

Host/Host Group:

Control-M Server: controlm-demo

HDFS commands

- get
- cat
- chmod
- copyfromlocal
- copytolocal
- cp
- mv
- mkdir
- rm
- rmdir

Kerberos account (impersonation)

Update - DEMO Connection Profile

Account Name:

Hadoop General ☒ Sqoop ☐ Hive ☐ Tajo

Authentication

Run as user: (Kerberos: Use Principal)

User's Keytab file path: