IBM SPSS Analytic Server
Version 2.1.0.1

Installation and Configuration Guide

IBM

Product Information

This edition applies to version 2.1.0.1, release 1, modification 0 of IBM SPSS Analytic Server and to all subsequent releases and modifications until otherwise indicated in new editions.

Contents

Chapter 1. Prerequisites

Before installing Analytic Server, review the following information.

System requirements

> For the most up-to-date system requirements information, use the Detailed system requirements reports at the IBM Technical Support site: http://publib.boulder.ibm.com/infocenter/prodguid/v1r0/clarity/softwareReqsForProduct.html. On this page:
>
> 1. Type SPSS Analytic Server as the product name and click **Search**.
> 2. Select the wanted version and scope of report, then click **Submit**.

Power systems

> Ensure that the IBM XLC and XLF compilers are installed and included in the PATH on all hosts in the cluster.
>
> You can find more information about getting a license for these compilers at the following web sites:
>
> - XL C for AIX: http://www-03.ibm.com/software/products/en/xlcaix
> - XL Fortran for AIX: http://www-03.ibm.com/software/products/en/xlfortran-aix

Hive/HCatalog

> If you plan to use NoSQL data sources, then configure Hive and HCatalog for remote access. Also ensure that `hive-site.xml` contains a *hive.metastore.uris* property in the form `thrift://<host_name>:<port>` that points to the active Thrift Hive Metastore server. Refer to your Hadoop distribution documentation for details.

Metadata repository

> By default, Analytic Server installs and uses a MySQL database. Alternatively, you can configure Analytic Server to use an existing DB2 installation. Regardless of which type of database you choose, it must have an encoding of UTF-8.

> **MySQL**
>
> > The default character set for MySQL is dependent upon the version and operating system. Use the following steps to determine whether your installation of MySQL is set to UTF-8.
> >
> > 1. Determine the version of MySQL.
> >
> >    ```
> >    mysql -V
> >    ```
> > 2. Determine the default character set for MySQL by running the following query from the MySQL command line interface.
> >
> >    ```
> >    mysql>show variables like 'char%';
> >    ```
> >
> >   If the character sets already set to UTF-8 no further changes are needed.
> > 3. Determine the default collation for MySQL by running the following query from the MySQL command line interface.
> >
> >    ```
> >    mysql>show variables like 'coll%';
> >    ```
> >
> >   If the collation is already set to UTF-8 no further changes are needed.
> > 4. If the default character set or collation is not UTF-8 refer to the MySQL documentation for details on how to edit /etc/my.cnf and restart the MySQL daemonto change the character set to UTF-8.

> **DB2** For more information on configuring DB2, see the Knowledge Center http://www-01.ibm.com/support/knowledgecenter/SSEPGG_10.5.0/com.ibm.db2.luw.kc.doc/welcome.html.

High-availability clusters

Load balancer

Your high availability cluster should have a load balancer that supports session affinity, sometimes also known as sticky sessions. Analytic Server identifies sessions with the cookie "request-token". This identifies a session for the duration of a user login for use in application-controlled session affinity. Please consult the documentation of your particular load balancer for the details of how it supports session affinity.

Chapter 2. Ambari Installation and Configuration

Ambari-specific prerequisites

In addition to the general prerequisites, review the following information.

Services
> Analytic Server is installed as an Ambari service. Prior to installing Analytic Server, you should ensure that HDFS, YARN, MapReduce2, Hive, and Zookeeper are added as Ambari services.

Password-less SSH
> Set up password-less SSH for the root user between the Analytic Metastore host and all hosts in the cluster.

Installation on Ambari

The basic process is to install the Analytic Server files on a host within the Ambari cluster, then add Analytic Server as an Ambari service. More detailed steps follow.

1. Navigate to the IBM Passport Advantage® Web Site and download the self-extracting binary file specific to your stack, stack version, and hardware architecture to a host within the Ambari cluster.

2. Execute the self-extracting binary file and follow the instructions to (optionally) view the license, accept the license, and choose online or offline installation.

   Online installation
   > Choose online installation if your Ambari server host, and all nodes in the cluster, are able to access http://ibm-open-platform.ibm.com.

   Offline installation
   > Choose offline if your Ambari server host does not have internet access. For details ee "Offline installation" on page 4.

3. Restart your Ambari server.

   ```
   ambari-server restart
   ```

4. Log on to your Ambari server and install Analytic Server as a service via the Ambari UI.

   Metadata repository

   > Analytic Server uses MySQL by default to track information about data sources, projects, and tenants. During installation you need to provide a username (`metadata.repository.user.name`) and password `metadata.repository.password` used in the JDBC connection between Analytic Server and MySQL. The installer creates the user in the MySQL database but that user is specific to the MySQL database and does not need to be an existing Linux or Hadoop user.

   > To change the metadata repository to DB2, follow these steps.

   > **Note:** You cannot change the metadata repository after installation is complete.

   > a. Ensure that DB2 is installed on another machine. For more information, see the metadata repository section of the topic Chapter 1, "Prerequisites," on page 1.
   > b. In the Ambari Services tab, navigate to the Configs tab of the Analytic Server service.
   > c. Open the **Advanced analytics-env** section.
   > d. Change the value of **as.database.type** from mysql to db2.
   > e. Open the **Advanced analytics-meta** section.
   > f. Change the value of **metadata.repository.driver** from com.mysql.jdbc.Driver to com.ibm.db2.jcc.DB2Driver.

g. Change the value of **metadata.repository.url** to `jdbc:db2://{DB2_HOST}:{PORT}/`
`{DBName}:currentSchema={SchemaName};`, where

- {DB2_HOST} is the hostname of the server where DB2 is installed
- {PORT} is the port on which DB2 is listening
- {SchemaName} is an available, unused schema.

If you are unsure of what values to enter, work with your DB2 administrator.

h. Supply valid DB2 credentials in **metadata.repository.user.name** and
metadata.repository.password.

i. Click **Save**.

Configuration settings that should not be changed after installation

Do not change the following settings after installation, or Analytic Server will fail to work.

- Analytic_Server_User
- Analytic_Server_UserID
- as.database.type
- metadata.repository.driver
- distrib.fs.root

5. You now have a functioning instance of Analytic Server. Further configuration is optional. For more information on configuring and administrating Analytic Server, see the topic: "Configuration" on page 5. For information on migrating an existing configuration to a new installation, see the topic: "Migration and upgrade" on page 13.

6. Open a web browser and enter the address `http://<host>:<port>/analyticserver/admin/ibm`, where <host> is the address of the Analytic Server host, and <port> is the port that Analytic Server is listening on. By default this is 9080. This URL opens the login dialog for the Analytic Server console. Log in as the Analytic Server administrator. By default this userid is admin and has password admin.

Offline installation

Offline installation will download the necessary RPM files, and should be run on a machine that can access http://ibm-open-platform.ibm.com. The RPM files can then be copied to the Ambari server host.

1. Install the tool that allows you to create a local Yum repository.

   `yum install createrepo`

2. Create a new directory that will serve as the repository for the Analytic Server RPM files. See the following example.

   `mkdir /home/root/repos/IBM-SPSS-AnalyticServer/x86_64`

3. Copy the necessary Analytic Server RPM files to this directory. The RPM files you need depend on your distribution, version, and architecture, shown below.

   BigInsights 4.1 (x86_64)

   `IBM-SPSS-AnalyticServer-ambari-2.1-BI-4.1-2.1.0.0-1.x86_64.rpm`

   `IBM-SPSS-AnalyticServer-2.1.0.0-1.x86_64.rpm`

   BigInsights 4.1 (PPC64LE)

   `IBM-SPSS-AnalyticServer-ambari-2.1-BI-4.1-2.1.0.0-1.ppc64le.rpm`

   `IBM-SPSS-AnalyticServer-2.1.0.0-1.ppc64le.rpm`

   HDP 2.3 (x86_64)

   `IBM-SPSS-AnalyticServer-2.1.0.0-1.x86_64.rpm`

   `IBM-SPSS-AnalyticServer-ambari-2.1-HDP-2.3-2.1.0.0-1.x86_64.rpm`

4. Create the local repository definition. For example, create a file called `analyticserver.repo` in `/etc/yum.repos.d/` (for RHEL, CentOS) or `/etc/zypp/repos.d/` (for SLES) with the following contents.

```
[IBM-SPSS-AnalyticServer]
name=IBM-SPSS-AnalyticServer-2.1.0.0
baseurl=file:///{path to local repository}
enabled=1
gpgcheck=0
protect=1
```

5. Create the local Yum repository. See the following example.

```
createrepo /home/root/repos/IBM-SPSS-AnalyticServer/x86_64
```

6. Install the Ambari metadata RPM from your local repository; this is the file of form
   `IBM-SPSS-AnalyticServer-ambari-xxx.rpm`. For example, to install the RPM for Ambari 2.1 on
   BigInsights 4.1 on RHEL or CentOS, run the following command.

```
sudo yum install IBM-SPSS-AnalyticServer-ambari-2.1-BI-4.1
```

   On SLES, the command changes as follows.

```
sudo zypper install IBM-SPSS-AnalyticServer-ambari-2.1-BI-4.1
```

7. Update your Ambari repository file `repoinfo.xml`, typically located in `/var/lib/ambari-server/`
   `resources/stacks/$stackName/$stackVersion/repos/`, to use the local Yum repository, by adding the
   following lines.

```
<os type="host_os">
   <repo>
        <baseurl>file://{path to local repository}/</baseurl>
        <repoid>IBM-SPSS-AnalyticServer</repoid>
        <reponame>IBM-SPSS-AnalyticServer-2.1.0.0</reponame>
   </repo>
</os>
```

Configuration

After installation, you can optionally configure and administer Analytic Server through the Ambari UI.

Note: The following conventions are used for Analytic Server file paths.

- {AS_ROOT} refers to the location where Analytic Server is deployed; for example,
  `/opt/IBM/SPSS/AnalyticServer/{version}`.
- {AS_SERVER_ROOT} refers to the location of the configuration, log, and server files; for example,
  `/opt/IBM/SPSS/AnalyticServer/{version}/ae_wlpserver/usr/servers/aeserver`.
- {AS_HOME} refers to the location on HDFS that is used by Analytic Server as a root folder.

Security

The **security.config** parameter defines the registry of users and groups that can be added as principals
to the Analytic Server system.

By default, a basic registry is defined with a single user, admin, with password admin. You can change the
registry by editing **security.config** or configuring Kerberos. The **security.config** parameter can be
found in the **Advanced analytics.cfg** section of the Configs tab of the Analytic Server service.

Note: If you edit the **security.config** parameter to alter the registry, you then need to add any new
users as principals to the Analytic Server system. See the *IBM SPSS Analytic Server Administrator's Guide*
for details on tenant management.

Making changes to the basic registry

The basic registry allows you to define a database of users and groups within the **security.config**
parameter.

The default basic registry looks like the following.

```
<basicRegistry id="basic" realm="ibm">
   <user name="admin" password="admin"/>
</basicRegistry>
```

An example of an altered basic registry follows.

```
<basicRegistry id="basic" realm="ibm">
  <user name="user1" password="{xor}Dz4sLG5tbGs="/>
  <user name="user2" password="Pass"/>
  <user name="user3" password="Pass"/>
  <user name="user4" password="Pass"/>
  <user name="admin" password="{xor}KzosKw=="/>
  <group name="Development">
    <member name="user1"/>
    <member name="user2"/>
  </group>
  <group name="QA">
    <member name="user3"/>
    <member name="user4"/>
  </group>
  <group name="ADMIN">
    <member name="user1"/>
    <member name="admin"/>
  </group>
</basicRegistry>
```

Passwords can be encoded to obfuscate their values with the securityUtility tool, which is in
`{AS_ROOT}/ae_wlpserver/bin`.

```
securityUtility encode changeit
        {xor}PDc+MTg6Nis=
```

Note: See http://www-01.ibm.com/support/knowledgecenter/SSD28V_8.5.5/
com.ibm.websphere.wlp.core.doc/ae/rwlp_command_securityutil.html for details of the securityUtility
tool.

Note: The basic registry is useful in a sandbox environment, but is not recommended for a production
environment.

Configure an LDAP registry
The LDAP registry allows you to authenticate users with an external LDAP server such as Active
Directory or OpenLDAP.

Here is an example of an ldapRegistry for OpenLDAP.

```
<ldapRegistry
    baseDN="ou=people,dc=aeldap,dc=org"
    ldapType="Custom"
    port="389"
    host="server"
    id="OpenLDAP"
    bindDN="cn=admin,dc=aeldap,dc=org"
    bindPassword="{xor}Dz4sLG5tbGs="
    searchTimeout="300000m"
    recursiveSearch="true">
    <customFilters
        id="customFilters"
        userFilter="(&amp;(uid=%v)(objectClass=inetOrgPerson))"
        groupFilter="(&amp;(cn=%v)(|(objectclass=organizationalUnit)))"
        groupMemberIdMap="posixGroup:memberUid"/>
</ldapRegistry>
```

For more examples of configurations, see the templates folder `{AS_ROOT}/ae_wlpserver/templates/config`.

Configure a secure socket layer (SSL) connection from Analytic Server to LDAP

1. Login to each of the Analytic Server machines as the Analytic Server user and create a common directory for SSL certificates.

   Note: By default, as_user is the Analytic Server user; see **Service accounts** under the Admin tab in the Ambari console.

2. Copy the key store and trust store files to some common directory on all Analytic Server machines. Also add the LDAP client CA certificate to the trust store. Below are some sample instructions.

   ```
   mkdir /home/as_user/security
   cd /home/as_user/security
   openssl s_client -connect <ldap-hostname>:636 -showcerts > client.cert
   $JAVA_HOME/bin/keytool -import -file ./client.cert -alias ldapCA -keystore mytrust.jks
   password : changeit
   ```

   Note: JAVA_HOME is the same JRE used for Analytic Server startup.

3. Passwords can be encoded to obfuscate their values with the securityUtility tool, which is in {AS_ROOT}/ae_wlpserver/bin. An example follows.

   ```
   securityUtility encode changeit
           {xor}PDc+MTg6Nis=
   ```

4. Login to the Ambari console and update the Analytic Server configuration setting **ssl.keystore.config** with the correct SSL configuration settings. An example follows.

   ```
   <ssl id="defaultSSLConfig" keyStoreRef="defaultKeyStore" trustStoreRef="defaultTrustStore"
        clientAuthenticationSupported="true"/>
           <keyStore id="defaultKeyStore" location="/home/as_user/security/mykey.jks" type="JKS"
                   password="{xor}Ozo5PiozKxYdEgwPDAweDG1uDz4sLCg7"/>
           <keyStore id="defaultTrustStore" location="/home/as_user/security/mytrust.jks" type="JKS"
                   password="{xor}PDc+MTg6Nis="/>
   ```

   Note: Use the absolute path for key and trust store files.

5. Update the Analytic Server configuration setting **security.config** with the correct LDAP configuration settings. For example, in the **ldapRegistry** element, set the **sslEnabled** attribute to true and the **sslRef** attribute to defaultSSLConfig.

Configuring Kerberos

Analytic Server supports Kerberos using Ambari.

1. Create accounts in the Kerberos user repository for all users you plan to give access to Analytic Server.

   Note: If the Analytic Server installation uses a basic registry, it must include the Kerberos user accounts, using "-" as the password. An example follows.

   ```
   <basicRegistry id="basic" realm="ibm">
     <user name="admin" password="-"/>
     <user name="user1" password="-"/>
     <user name="user2" password="-"/>
     <group name="group1">
        <member name="admin"/>
        <member name="user1"/>
        <member name="user2"/>
     </group>
     <group name="group2">
        <member name="admin"/>
        <member name="user1"/>
     </group>
   </basicRegistry>
   ```

2. Create an OS user account for each of the users created in the previous step on each and every Analytic Server node and Hadoop node.

- Make sure that the UID for these users matches on all machines. You can test this using the kinit command to log in to each of the accounts.
- Make sure that the UID adheres to the "Minimum user ID for submitting job" Yarn setting. This is the `min.user.id` parameter in container-executor.cfg. For example, if `min.user.id` is 1000, then each user account created must have a UID greater than or equal to 1000.

3. Create a user home folder on HDFS for all principals in Analytic Server. For example, if you add testuser1 to the Analytic Server system, then create a home folder like /user/testuser1 on HDFS and ensure that testuser1 has read and write permissions to this folder.

4. [Optional] If you plan to use HCatalog data sources and Analytic Server is installed on a different machine from the Hive metastore, you need to impersonate the Hive client on HDFS.
   a. Navigate to the Configs tab of the HDFS service in the Ambari console.
   b. Edit the `hadoop.proxyuser.hive.groups` parameter to have the value *, or a group that contains all of the users allowed to log in to Analytic Server.
   c. Edit the `hadoop.proxyuser.hive.hosts` parameter to have the value *, or the list of hosts on which the Hive metastore and every instance of Analytic Server are installed as services.
   d. Restart the HDFS service.

After these steps have been performed and Analytic Server is installed, Analytic Server silently and automatically configures Kerberos.

Configuring HAProxy for Single Sign On (SSO) using Kerberos

1. Configure and start HAProxy per the HAProxy documentation guide: http://www.haproxy.org/#docs
2. Create the Kerberos principle (HTTP/<proxyHostname>@<realm>) and keytab file for the HAProxy host, where <proxyHostname> is the full name of the HAProxy host, and <realm> is the Kerberos realm.
3. Copy the keytab file to each of the Analytic Server hosts as /etc/security/keytabs/ spnego_proxy.service.keytab
4. Update permissions to this file on each of the Analytic Server hosts. An example follows.

```
chown root:hadoop /etc/security/keytabs/spnego_proxy.service.keytab
chmod 440 /etc/security/keytabs/spnego_proxy.service.keytab
```

5. Open the Amabri console and update the following properties in the Analytic Server 'Custom analytics.cfg' section.

```
web.authentication.kerberos.keytab=/etc/security/keytabs/spnego_proxy.service.keytab
web.authentication.kerberos.principal=HTTP/<proxy machine full name>@<realm>
```

6. Save the configuration and restart all Analytic Server services from the Amabri console.

Now users are able to log in to Analytic Server using Kerberos SSO.

Disabling Kerberos

1. Disable Kerberos in the Ambari console.
2. Stop the Analytic Server service.
3. Remove the following parameters from Custom analytics.cfg.

```
default.security.provider
hdfs.keytab
hdfs.user
java.security.krb5.conf
jdbc.db.connect.method.kerberos
web.authentication.kerberos.keytab
web.authentication.kerberos.principal
```

4. Click **Save** and restart the Analytic Server service.

Enabling Secure Socket Layer (SSL) connections to the Analytic Server console

By default, Analytic Server generates self-signed certificates to enable Secure Socket Layer (SSL), so you can access the Analytic Server console through the secure port by accepting self signed certificates. In order to make HTTPS access more secure, you need to install 3rd party vendor certificates.

To install 3rd party vendor certificates, follow these steps.

1. Copy the 3rd party vendor key store and trust store certificates to the same directory in all Analytic Server nodes; for example, /home/as_user/security.

   Note: The Analytic Server User must have read access to this directory.

2. In the Ambari Services tab, navigate to the Configs tab of the Analytic Server service.

3. Edit the **ssl.keystore.config** parameter.
   ```
   <ssl id="defaultSSLConfig"
        keyStoreRef="defaultKeyStore"
        trustStoreRef="defaultTrustStore"
        clientAuthenticationSupported="true"/>
   <keyStore id="defaultKeyStore"
             location="<KEYSTORE-LOCATION>"
             type="<TYPE>"
             password="<PASSWORD>"/>
   <keyStore id="defaultTrustStore"
             location="<TRUSTSTORE-LOCATION>"
             type="<TYPE>"
             password="<PASSWORD>"/>
   ```

   Replace
   - <KEYSTORE-LOCATION> with the absolute location of the key store; for example: /home/as_user/security/mykey.jks
   - <TRUSTSTORE-LOCATION> with the absolute location of the trust store; for example: /home/as_user/security/mytrust.jks
   - <TYPE> with the type of the certificate; for example: JKS, PKCS12 etc.
   - <PASSWORD> with the encrypted password in Base64 encryption format. For encoding you can use the securityUtility; for example: /opt/ibm/spss/analyticserver/2.1/ae_wlpserver/bin/ securityUtility encode <password>

   If you want to generate a self-signed certificate, you can use securityUtility; for example: /opt/ibm/spss/analyticserver/2.1/ae_wlpserver/bin/securityUtility createSSLCertificate --server=myserver --password=mypassword --validity=365 --subject=CN=mycompany,O=myOrg,C=myCountry. For more information on securityUtility and other SSL settings, refer to the WebSphere Liberty Profile documentation.

4. Click **Save** and restart the Analytic Server service.

Enabling Support for Essentials for R

Analytic Server supports scoring R models and running R scripts.

To configure support for R after a successful Analytic Server installation:

1. Download the self-extracting archive (BIN) for IBM SPSS Modeler Essentials for R RPM. Essentials for R is available for download (https://www14.software.ibm.com/webapp/iwm/web/ preLogin.do?source=swg-tspssp). Choose the file specific to your stack, stack version, and hardware architecture.

2. Execute the self-extracting binary file and follow the instructions to (optionally) view the license, accept the license, and choose online or offline installation.

   Online installation
   Choose online installation if your Ambari server host, and all nodes in the cluster, are able to access http://ibm-open-platform.ibm.com.

Offline installation

Choose offline if your Ambari server host does not have internet access. Offline installation will download the necessary RPM files, and should be run on a machine that can access http://ibm-open-platform.ibm.com. The RPM files can then be copied to the Ambari server host.

a. Copy the necessary Essentials for R RPM files to any location on your Ambari server host. The RPM files you need depend on your distribution, version, and architecture, shown below.

BigInsights 4.1 (x86_64)
```
IBM-SPSS-ModelerEssentialsR-ambari-2.1-BI-4.1-8.2.0.0-1.x86_64.rpm
```

BigInsights 4.1 (PPC64LE)
```
IBM-SPSS-ModelerEssentialsR-ambari-2.1-BI-4.1-8.2.0.0-1.ppc64le.rpm
```

HDP 2.3 (x86_64)
```
IBM-SPSS-ModelerEssentialsR-ambari-2.1-HDP-2.3-8.2.0.0-1.x86_64.rpm
```

b. Install the RPM. For example, the following command installs Essentials for R on Hortonworks 2.3.
```
rpm -i IBM-SPSS-ModelerEssentialsR-ambari-2.1-HDP-2.3-8.2.0.0-1.x86_64.rpm
```

3. Restart your Ambari server.
```
ambari-server restart
```

4. Log on to your Ambari server and install SPSS Essentials for R as a service via the Ambari console. SPSS Essentials for R should be installed on every host where Analytic Server and the Analytic Metastore is installed.

Note: Ambari will attempt to install gcc-c++ and gcc-gfortran (RHEL) and gcc-fortran (SUSE) prior to installing R. These packages are declared as dependencies on R's Ambari service definition. Ensure that the servers where R is to be installed and executed are configured to download gcc-c++ and gcc-[g]fortran RPMs or have GCC and FORTRAN compilers installed. If the installation of Essentials for R fails, install these packages manually prior to installing Essentials for R.

5. Refresh the Analytic Server service.

6. Run the `update_clientdeps` script using the instructions in "Updating client dependencies" on page 12.

7. You must also install Essentials for R on the machine that hosts SPSS® Modeler Server. See the SPSS Modeler documentation for details.

Enabling relational database sources

Analytic Server can use relational database sources if you supply the JDBC drivers in a shared directory on each Analytic Server host. By default, this directory is `/usr/share/jdbc`.

To change the shared directory, follow these steps.

1. In the Ambari Services tab, navigate to the Configs tab of the Analytic Server service.
2. Open the **Advanced analytics.cfg** section.
3. Specify the path of the shared directory of JDBC drivers in **jdbc.drivers.location**.
4. Click **Save**.
5. Stop the Analytic Server service.
6. Click **Refresh**.
7. Start the Analytic Server service.

Table 1. Supported databases

Database	Supported versions	JDBC driver jars	Vendor
Amazon Redshift	8.0.2 or later	`RedshiftJDBC41-1.1.6.1006.jar` or later	Amazon
DB2 for Linux, UNIX, and Windows	10.5, 10.1, 9.7	`db2jcc.jar`	IBM
DB2 z/OS	11, 10	`db2jcc.jar, db2_license_cisuz.jar`	IBM
Greenplum	5, 4.2.x	postgresql.jar	Greenplum
Netezza	7, 6.x	`nzjdbc.jar`	IBM
Oracle	12c, 11g R2 (11.2)	`ojdbc6.jar, orai18n.jar`	Oracle
SQL Server	2014, 2012, 2008 R2	`sqljdbc4.jar`	Microsoft
Sybase IQ	16.x, 15.4, 15.2	`jconnect70.jar`	Sybase
Teradata	14, 14.1, 15	`tdgssconfig.jar, terajdbc4.jar`	Teradata

Note: If you created a Redshift data source prior to installing Analytic Server 2.1.0.1, you need perform the following steps in order to use the Redshift data source.

1. In the Analytic Server console, open the Redshift data source.
2. Reselect the type of database data source as Redshift.
3. Retype the Redshift server address.
4. Retype the database name and username. The password should be repopulated automatically.
5. Reselect the database table.

Enabling HCatalog data sources

Analytic Server provides support for a number of data sources through Hive/HCatalog. Some sources require manual configuration steps.

1. Collect the necessary JAR files to enable the data source. See the sections below for details.
2. Add these JAR files to the {HIVE_HOME}/auxlib directory and to the /usr/share/hive directory on each Analytic Server node.
3. Restart the Hive Metastore service.
4. Refresh the Analytic Metastore service.
5. Restart each and every instance of the Analytic Server service.

NoSQL databases

Analytic Server supports any NoSQL database for which a Hive storage handler is available from the vendor.

No additional steps are necessary to enable support for Apache HBase and Apache Accumulo.

For other NoSQL databases, contact the database vendor and obtain the storage handler and related jars.

File-based Hive tables

Analytic Server supports any file-based Hive tables for which a built-in or custom Hive SerDe (serializer-deserializer) is available.

The Hive XML SerDe for processing XML files is located in the Maven Central Repository at http://search.maven.org/#search%7Cga%7C1%7Chivexmlserde.

Changing ports used by Analytic Server

Analytic Server uses the 9080 port for HTTP and the 9443 port for HTTPS by default. To change the port settings, follow these steps.

1. In the Ambari Services tab, navigate to the Configs tab of the Analytic Server service.
2. Open the **Advanced analytics.cfg** section.
3. Specify the desired HTTP and HTTPS ports in **`http.port`** and **`https.port`**, respectively.
4. Click **Save**.
5. Restart the Analytic Server service.

High availability Analytic Server

You can make Analytic Server highly available by adding it as a service to multiple nodes in your cluster.

1. In the Ambari console, navigate to the Hosts tab.
2. Select a host that is not already running Analytic Server as a service.
3. On the Summary tab, click **Add** and select Analytic Server.
4. Click **Confirm Add**

Optimizing JVM options for small data

You can edit JVM properties in order to optimize your system when running small (M3R) jobs.

In the Ambari console, see the Advanced analytics-jvm-options section of the Configs tab in the Analytic Server service. Modifying the following parameters sets the heap size for jobs run on the server that hosts Analytic Server; that is, not Hadoop. This is important if running small (M3R) jobs, and you may need to experiment with these values to optimize your system.

```
-Xms512M
-Xmx2048M
```

Updating client dependencies

This section describes how to update the Analytic Server service's dependencies using the `update_clientdeps` script.

1. Login to Ambari server host as root.
2. Change directory to /var/lib/ambari-server/resources/stacks/<stack-name>/<stack-version>/ services/ANALYTICSERVER/package/scripts; see the following example.

   ```
   cd "/var/lib/ambari-server/resources/stacks/HDP/2.3/services/ANALYTICSERVER/package/scripts"
   ```
3. Run the `update_clientdeps` script with the following arguments.

   -u <ambari-user>
   > The Ambari account username

   -p <ambari-password>
   > The password for the Ambari account user.

   -h <ambari-host>
   > The hostname of the Ambari server.

   -x <ambari-port>
   > The port on which Ambari is listening.

   See the following example.

   ```
   ./update_clientdeps.sh -u admin -p admin -h host.domain -x 8080
   ```

4. Restart the Ambari server using the following command.

```
ambari-server restart
```

Migration and upgrade

Analytic Server allows you to migrate data and configuration settings from an existing Analytic Server installation to a new installation.

Migrate to a new version of Analytic Server

If you have an existing installation of Analytic Server 2.0 and have purchased 2.1, then you can migrate your 2.0 configuration settings to your 2.1 installation.

Restriction: If you have a version earlier than 2.0 installed, you must first migrate from the earlier version to 2.0 and then from version 2.0 to 2.1.

Restriction: Your 2.0 and 2.1 installations cannot coexist on the same Hadoop cluster. If you configure your 2.1 installation to use the same Hadoop cluster as your 2.0 installation, the 2.0 installation will no longer function.

Upgrade to a new version of Analytic Server

If you have an existing installation of Analytic Server 2.1, you can upgrade the installation in place to version 2.1.0.1.

Migration steps, 2.0 to 2.1

1. Install the new installation of Analytic Server according to the instructions in "Installation on Ambari" on page 3.
2. Copy the analytic root from your old installation to your new one.
   a. If you are unsure of the location of the analytic root, run `hadoop -fs ls`. The path to the analytic root will be of form `/user/aeuser/analytic-root`, where aeuser is the userid that owns the analytic root.
   b. Change ownership of the analytic root from aeuser to as_user

   ```
   hadoop dfs -chown -R {as_user:{group}} {path to 2.0 analytic-root}
   ```

   Note: If you plan to use the existing Analytic Server installation after migration, make a copy of the analytic-root directory in HDFS and then change the ownership on the copy of the directory.
   c. Log in to the host of the new Analytic Server installation as as_user. Delete the `/user/as_user/analytic-root` directory, if it exists.
   d. Run the following copy script.

   ```
   hadoop distcp hftp://{host of 2.0 namenode}:50070/{path to 2.0 analytic-root} hdfs://{host of 2.1 namenode}/user/as_user/analytic-root
   ```
3. In the Ambari console, stop the Analytic Server service.
4. Ensure the Analytic Metastore service is running.
5. Collect the configuration settings from the old installation.
   a. Copy the `configcollector.zip` archive in your new installation to `{AS_ROOT}\tools` in your old installation.
   b. Extract the copy of `configcollector.zip`. This creates a new `configcollector` subdirectory in your old installation.
   c. Run the configuration collector tool in your old installation by executing the **configcollector** script in `{AS_ROOT}\tools\configcollector`. Copy the resulting compressed (ZIP) file to the server that hosts your new installation.
6. Run the migration tool by executing the **migrationtool** script and passing the path of the compressed file that was created by the configuration collector as an argument. As example follows.

   ```
   migrationtool.sh /opt/ibm/spss/analyticserver/2.1/ASConfiguration_2.0.0.0.xxx.zip
   ```

7. Clear the Zookeeper state. In the Zookeeper bin directory (for example, `/usr/hdp/current/zookeeper/bin` on Hortonworks or `/usr/iop/current/zookeeper/bin` on BigInsights), run the following command.

   ```
   ./zkCli.sh rmr /AnalyticServer
   ```

8. In the Ambari console, start the Analytic Server service.

   Note: If you configured R for use with the existing Analytic Server installation, you will need to follow the steps to configure it with the new Analytic Server installation.

Upgrade steps, 2.1 to 2.1.0.1

1. In the Ambari console, stop the Analytic Server service.
2. Depending upon your type of installation, follow these steps.

   Online installation

   a. Create a copy of `IBM-SPSS-AnalyticServer-2.1.0.0.repo`
   b. Rename the copy `IBM-SPSS-AnalyticServer-2.1.0.1.repo`
   c. Edit `IBM-SPSS-AnalyticServer-2.1.0.1.repo` so that it points to
      `http://ibm-open-platform.ibm.com/repos/IBM-SPSS-AnalyticServer/2.1.0.1/x86_64`

   Offline installation

   a. Donwload and execute the self-extracting binary file.
   b. Create a new directory that will serve as the repository for the Analytic Server RPM files. See the following example.
      `mkdir /home/root/repos/IBM-SPSS-AnalyticServer/2.1.0.1/x86_64`
   c. Copy the necessary Analytic Server RPM files to this directory. The RPM files you need depend on your distribution, version, and architecture.
   d. Create the local repository definition. For example, create a file called `analyticserver.repo` in `/etc/yum.repos.d/` (for RHEL, CentOS) or `/etc/zypp/repos.d/` (for SLES) with the following contents.

      ```
      [IBM-SPSS-AnalyticServer]
      name=IBM-SPSS-AnalyticServer-2.1.0.1
      baseurl=file:///{path to local repository}
      enabled=1
      gpgcheck=0
      protect=1
      ```

   e. Create the local Yum repository. See the following example.
      `createrepo /home/root/repos/IBM-SPSS-AnalyticServer/2.1.0.1/x86_64`

3. Clean the Ambari metadata from your local cache. For example, to clean the cache on RHEL or CentOS, run the following command.

   ```
   sudo yum clean all
   ```

   On SLES, the command changes as follows.

   ```
   sudo zypper refresh
   ```

4. On each Analytic Server host, upgrade the RPMs. For example, to upgrade on RHEL or CentOS, run the following commands.

   ```
   chown -R as_user:hadoop /opt/ibm/spss/analyticserver/2.1
   sudo yum upgrade IBM-SPSS-AnalyticServer
   ```

   On SLES, the command changes as follows.

   ```
   sudo zypper up IBM-SPSS-AnalyticServer
   ```

5. Refresh the stack.

/var/lib/ambari-server/resources/stacks/$stackName/$stackVersion/repos/, to use the local Yum repository, by adding the following lines.

```
<os type="host_os">
```

BigInsights

In the Ambari console, execute the custom "Refresh" action.

Hortonworks

Navigate to one of your Analytic Server nodes and run the following command.

```
sudo -u as_user /opt/ibm/spss/analyticserver/2.1/bin/refresh.sh
```

6. **Offline installation only.** Update your Ambari repository file `repoinfo.xml`, typically located in `/var/lib/ambari-server/resources/stacks/$stackName$/$stackVersion$/repos`, to use the local Yum repository, by adding the following lines.

```
<os type="host_os">

    <repo>

        <baseurl>file://{path to local repository}/</baseurl>
        <repoid>IBM-SPSS-AnalyticServer</repoid>
        <reponame>IBM-SPSS-AnalyticServer-2.1.0.1</reponame>

    </repo>

</os>
```

7. In the Ambari console, start the Analytic Server service.

Uninstalling

1. On the Analytic Metastore host, run the `remove_as.sh` script in the {AS_ROOT}/bin directory with the following parameters.

   u   Required. The Ambari Server administrator's user ID.

   p   Required. The Ambari Server administrator's password.

   h   Required. The Ambari Server host name.

   x   Required. The Ambari Server port.

   l   Optional. Enables secure mode.

   Examples follow.

   ```
   remove_as.sh -u admin -p admin -h one.cluster -x 8081
   ```

   Removes Analytic Server from a cluster with Ambari host one.cluster.

   ```
   remove_as.sh -u admin -p admin -h one.cluster -x 8081 -l
   ```

   Removes Analytic Server from a cluster with Ambari host one.cluster, in secure mode.

Note: This operation removes the Analytic Server folder on HDFS.

Note: This operation does not remove any DB2 schemas associated with Analytic Server. Consult the DB2 documentation for information on manually removing schemas

Uninstalling Essentials for R

1. On the Essentials for R host, run the `remove_r.sh` script in the {AS_ROOT}/bin directory with the following parameters.

   u   Required. The Ambari Server administrator's user ID.

   p   Required. The Ambari Server administrator's password.

   h   Required. The Ambari Server host name.

x   Required. The Ambari Server port.

l   Optional. Enables secure mode.

Examples follow.

```
remove_r.sh -u admin -p admin -h one.cluster -x 8081
```

Removes Essentials for R from a cluster with Ambari host one.cluster.

```
remove_r.sh -u admin -p admin -h one.cluster -x 8081 -l
```

Removes Essentials for R from a cluster with Ambari host one.cluster, in secure mode.

2. Remove the R services directory from the Ambari server services directory. For example, in BigInsights 4.1, the `ESSENTIALR` directory is located in `/var/lib/ambari-server/resources/stacks/BigInsights/4.1/services`.

3. In the Ambari console, verify that the Essentials for R service no longer exists.

Chapter 3. Cloudera Installation and Configuration

Cloudera overview

Cloudera is an open source Apache Hadoop distribution. The Cloudera Distribution Including Apache Hadoop (CDH), targets enterprise-class deployments of that technology.

Analytic Server can run on the CDH platform. CDH contains the main, core elements of Hadoop that provide reliable, scalable distributed data processing of large data sets (chiefly MapReduce and HDFS), as well as other enterprise-oriented components that provide security, high availability, and integration with hardware and other software.

Cloudera-specific prerequisites

In addition to the general prerequisites, review the following information.

Services
> Ensure that the following instances are installed on each Analytic Server host.
> - Accumulo: Gateway
> - HBase: Gateway, Master or RegionServer
> - HDFS: Gateway, DataNode or NameNode
> - Hive: Gateway, Hive Metastore Server or HiveServer2
> - Yarn: Gateway, ResourceManager or NodeManager

Metadata repository
> If you plan to use MySQL as Analytic Server's metadata repository, follow the instructions for "Configuring MySQL for Analytic Server."

Configuring MySQL for Analytic Server

Configuring the IBM® SPSS Analytic Server in Cloudera Manager requires the installation and configuration of a MySQL server database.

1. Run the following command from a command window on the node on which the MySQL database is stored:

   ```
   yum install mysql-server
   ```

   Note: Use `zypper install mysql` for SuSE Linux.

2. Run the following command from a command window on each Cloudera cluster node:

   ```
   yum install mysql-connector-java
   ```

   Note: Use `zypper install mysql` for SuSE Linux.

3. Decide upon, and take note of, the Analytic Server database name, database user name, and database password that Analytic Server uses when it accesses the MySQL database.

4. Install Analytic Server according to the instructions in "Installation on Cloudera" on page 18.

5. Copy the `/opt/cloudera/parcels/AnalyticServer/bin/add_mysql_user.sh` script from one of the servers managed by Cloudera to the node where the MySQL database is installed. Run the script with parameters that are appropriate for your particular configuration. For example:

   ```
   ./add_mysql_user.sh -u <database_user_name> -p <database_password> -d
   <database_name>
   ```

   Notes: The a `-r <dbRootPassword>`) parameter is required when the database runs in secured mode (the root user password is set).

The `-r <dbUserPassword>` and `-t <dbUserName>` parameters are required when the database is running in secured mode with a user name other than `root`.

6. Open Cloudera Manager and navigate to the Configuration tab of the Analytic Server service.

   a. In the **Analytic Server metastore driver class (jndi.aedb.driver)** property, select `com.mysql.jdbc.Driver`

   b. You must specify matching values for the previously noted Analytic Server database name, database user name, and database password on the pane where the Analytic Server configuration entries are specified. The **Analytic Server metastore repository URL (jndi.aedb.url)**, **Analytic Server metastore username (jndi.aedb.username)**, and **Analytic Server metastore password (jndi.aedb.password)** properties must be updated to match the values that were provided to the `add_mysql_user.sh` command

Installation on Cloudera

The following steps explain the process of manually installing IBM SPSS Analytic Server in Cloudera Manager.

Online installation

1. Download and run the Cloudera self-extracting `*.bin` installer on the Cloudera Manager server node. Follow the installation prompts by accepting the license agreement and keeping the default CSD installation directory.

   Note: You must specify a different CSD directory if it is altered from the default location.

2. Restart Cloudera Manager after the installation is complete.

3. Open the Clouder Manager interface (for example, `http://${CM_HOST}:7180/cmf/login` with the default login credentials of `admin/admin`), refresh the **Remote Parcel Repository URLs**, and verify that the URLs are correct. For example:

   `http://ibm-open-platform.ibm.com/repos/IBM-SPSS-AnalyticServer/2.1/cloudera/parcels/latest/`

   or

   `http://ibm-open-platform.ibm.com/repos/IBM-SPSS-AnalyticServer/2.1.0.1/cloudera/`

   Note: The **Parcel Update Frequency** and **Remote Parcel Repository URLs** can be updated to suit your specific needs.

4. After Cloudera Manager refreshes the parcels (you can manually refresh the parcels by clicking **Check for New Parcels**), you will see that the **AnalyticServer** parcel status is set to **Available Remotely**.

5. Select **Download** > **Distribute** > **Activate**. The **AnalyticServer** parcel status is updated to **Distributed, Activated**.

6. Configure MySQL for Analytic Server.

7. In Cloudera Manager, add Analytic Server as a service and decide where to place the Analytic Server. You need to provide the following information in the Add Service Wizard:

   • Analytic Server metastore user name
   • Analytic Server metastore password

   The Add Service Wizard shows the overall progress during each phase of the service creation process, and provides a final confirmation message when the service is successfully installed and configured on the cluster.

Note: After successfully installing Analytic Server, do not click **Create Analytic Server Metastore** in the Actions dropdown of the Analytic Server services page in Cloudera Manager. This will overwrite the existing metadata repository.

Offline installation

The offline installation steps are the same as the online steps except you must manually download the parcels and metadata that are appropriate for your particular operating system.

RedHat Linux requires the following files:
* AnalyticServer-2.1.0.1-el6.parcel
* AnalyticServer-2.1.0.1-el6.parcel.sha
* manifest.json

SuSE Linux requires the following files:
* AnalyticServer-2.1.0.1-sles11.parcel
* AnalyticServer-2.1.0.1-sles11.parcel.sha
* manifest.json

1. Copy the files to your local Cloudera `repo` path on the Cloudera Manager master cluster node. The default path is `/opt/cloudera/parcel-repo` (the path is configurable in the Cloudera Manager user interface).

   The **AnalyticServer** parcel shows as **downloaded** after Cloudera Manager refreshes the parcel. You can click **Check for New Parcels** to force a refresh.
2. Click **Distribute** > **Activate**.

   The **AnalyticServer** parcel shows as distributed and activated.

Configuring Cloudera

After installation, you can optionally configure and administer Analytic Server through the Cloudera Manager.

Note: The following conventions are used for Analytic Server file paths.
* {AS_ROOT} refers to the location where Analytic Server is deployed; for example, `/opt/cloudera/parcels/AnalyticServer`.
* {AS_SERVER_ROOT} refers to the location of the configuration, log, and server files; for example, `/opt/cloudera/parcels/AnalyticServer/ae_wlpserver/usr/servers/aeserver`.
* {AS_HOME} refers to the location on HDFS that is used by Analytic Server as a root folder; for example, `/user/as_user/analytic-root`.

Security

The `security_cfg` parameter defines the registry of users and groups that can be added as principals to the Analytic Server system.

By default, a basic registry is defined with a single user, `admin`, with password `admin`. You can change the registry by editing `security_cfg` or configuring Kerberos as the security provider. The `security_cfg` parameter can be found in the **Analytic Server Advanced Configuration Snippet** section of the Configuration tab of the Analytic Server service.

Note: If you edit the `security_cfg` parameter to alter the registry, you then need to add any new users as principals to the Analytic Server system. See the *IBM SPSS Analytic Server Administrator's Guide* for details on tenant management.

Making changes to the basic registry
The basic registry allows you to define a database of users and groups within the `security_cfg` parameter.

The default basic registry looks like the following.

```
<basicRegistry id="basic" realm="ibm">
   <user name="admin" password="admin"/>
</basicRegistry>
```

An example of an altered basic registry follows.

```
<basicRegistry id="basic" realm="ibm">
  <user name="user1" password="{xor}Dz4sLG5tbGs="/>
  <user name="user2" password="Pass"/>
  <user name="user3" password="Pass"/>
  <user name="user4" password="Pass"/>
  <user name="admin" password="{xor}KzosKw=="/>
  <group name="Development">
    <member name="user1"/>
    <member name="user2"/>
  </group>
  <group name="QA">
    <member name="user3"/>
    <member name="user4"/>
  </group>
  <group name="ADMIN">
    <member name="user1"/>
    <member name="admin"/>
  </group>
</basicRegistry>
```

Passwords can be encoded to obfuscate their values with the securityUtility tool, which is in {AS_ROOT}/ae_wlpserver/bin.

```
securityUtility encode changeit
        {xor}PDc+MTg6Nis=
```

Note: See http://www-01.ibm.com/support/knowledgecenter/SSD28V_8.5.5/ com.ibm.websphere.wlp.core.doc/ae/rwlp_command_securityutil.html for details of the securityUtility tool.

Note: The basic registry is useful in a sandbox environment, but is not recommended for a production environment.

Configure an LDAP registry

The LDAP registry allows you to authenticate users with an external LDAP server such as Active Directory or OpenLDAP.

Here is an example of an ldapRegistry for OpenLDAP.

```
<ldapRegistry
    baseDN="ou=people,dc=aeldap,dc=org"
    ldapType="Custom"
    port="389"
    host="server"
    id="OpenLDAP"
    bindDN="cn=admin,dc=aeldap,dc=org"
    bindPassword="{xor}Dz4sLG5tbGs="
    searchTimeout="300000m"
    recursiveSearch="true">
    <customFilters
        id="customFilters"
        userFilter="(&amp;(uid=%v)(objectClass=inetOrgPerson))"
        groupFilter="(&amp;(cn=%v)(|(objectclass=organizationalUnit)))"
        groupMemberIdMap="posixGroup:memberUid"/>
</ldapRegistry>
```

For more examples of configurations, see the templates folder {AS_ROOT}/ae_wlpserver/templates/config.

Configure a secure socket layer (SSL) connection from Analytic Server to LDAP

1. Login to each of the Analytic Server machines as the Analytic Server user and create a common directory for SSL certificates.

   Note: On Cloudera, the Analytic Server user is always as_user, and this cannot be changed.

2. Copy the key store and trust store files to some common directory on all Analytic Server machines. Also add the LDAP client CA certificate to the trust store. Below are some sample instructions.

   ```
   mkdir /home/as_user/security
   cd /home/as_user/security
   openssl s_client -connect <ldap-hostname>:636 -showcerts > client.cert
   $JAVA_HOME/bin/keytool -import -file ./client.cert -alias ldapCA -keystore mytrust.jks
   password : changeit
   ```

   Note: JAVA_HOME is the same JRE used for Analytic Server startup.

3. Passwords can be encoded to obfuscate their values with the securityUtility tool, which is in `{AS_ROOT}/ae_wlpserver/bin`. An example follows.

   ```
   securityUtility encode changeit
           {xor}PDc+MTg6Nis=
   ```

4. Login to Cloudera Manager and update the Analytic Server configuration setting **ssl_cfg** with the correct SSL configuration settings. An example follows.

   ```
   <ssl id="defaultSSLConfig" keyStoreRef="defaultKeyStore" trustStoreRef="defaultTrustStore"
        clientAuthenticationSupported="true"/>
           <keyStore id="defaultKeyStore" location="/home/as_user/security/mykey.jks" type="JKS"
                   password="{xor}Ozo5PiozKxYdEgwPDAweDG1uDz4sLCg7"/>
           <keyStore id="defaultTrustStore" location="/home/as_user/security/mytrust.jks" type="JKS"
                   password="{xor}PDc+MTg6Nis="/>
   ```

   Note: Use the absolute path for key and trust store files.

5. Update the Analytic Server configuration setting **security_cfg** with the correct LDAP configuration settings. For example, in the **ldapRegistry** element, set the **sslEnabled** attribute to `true` and the **sslRef** attribute to defaultSSLConfig.

Configuring Kerberos

Analytic Server supports Kerberos in Cloudera.

1. Create accounts in the Kerberos user repository for all users you plan to give access to Analytic Server.

   Note: If the Analytic Server installation uses a basic registry, it must include the Kerberos user accounts, using "-" as the password. An example follows.

   ```
   <basicRegistry id="basic" realm="ibm">
     <user name="admin" password="-"/>
     <user name="user1" password="-"/>
     <user name="user2" password="-"/>
     <group name="group1">
        <member name="admin"/>
        <member name="user1"/>
        <member name="user2"/>
     </group>
     <group name="group2">
        <member name="admin"/>
        <member name="user1"/>
     </group>
   </basicRegistry>
   ```

2. Create an OS user account for each of the users created in the previous step on each and every Analytic Server node and Hadoop node.
   - Make sure that the UID for these users matches on all machines. You can test this using the kinit command to log in to each of the accounts.

- Make sure that the UID adheres to the "Minimum user ID for submitting job" Yarn setting. This is the `min.user.id` parameter in container-executor.cfg. For example, if `min.user.id` is 1000, then each user account created must have a UID greater than or equal to 1000.

3. Create a user home folder on HDFS for all principals in Analytic Server. For example, if you add testuser1 to the Analytic Server system, then create a home folder like `/user/testuser1` on HDFS and ensure that testuser1 has read and write permissions to this folder.

4. [Optional] If you plan to use HCatalog data sources and Analytic Server is installed on a different machine from the Hive metastore, you need to impersonate the Hive client on HDFS.

   a. Navigate to the Configuration tab of the HDFS service in Cloudera Manager.

      Note: The following parameters may not appear on the Configuration tab if they have not already been set. In this case, run a search to find them.

   b. Edit the `hadoop.proxyuser.hive.groups` parameter to have the value *, or a group that contains all of the users allowed to log in to Analytic Server.

   c. Edit the `hadoop.proxyuser.hive.hosts` parameter to have the value *, or the list of hosts on which the Hive metastore and every instance of Analytic Server are installed as services.

   d. Restart the HDFS service.

After these steps have been performed and Analytic Server is installed, Analytic Server silently and automatically configures Kerberos.

Configuring HAProxy for Single Sign On (SSO) using Kerberos

1. Configure and start HAProxy per the HAProxy documentation guide: http://www.haproxy.org/#docs

2. Create the Kerberos principle (HTTP/<proxyHostname>@<realm>) and keytab file for the HAProxy host, where <proxyHostname> is the full name of the HAProxy host, and <realm> is the Kerberos realm.

3. Copy the keytab file to each of the Analytic Server hosts as `/etc/security/keytabs/spnego_proxy.service.keytab`

4. Update permissions to this file on each of the Analytic Server hosts. An example follows.

   ```
   chown root:hadoop /etc/security/keytabs/spnego_proxy.service.keytab
   chmod 440 /etc/security/keytabs/spnego_proxy.service.keytab
   ```

5. Open Cloudera Manager and add or update the following properties in the Analytic Server **Analytic Server Advanced Configuration Snippet (Safety Valve) for analyticserver-conf/config.properties** area.

   ```
   web.authentication.kerberos.keytab=/etc/security/keytabs/spnego_proxy.service.keytab
   web.authentication.kerberos.principal=HTTP/<proxy machine full name>@<realm>
   ```

6. Save the configuration and restart all Analytic Server services from Cloudera Manager.

7. Instruct users to configure their browser to use Kerberos.

Now users are able to log in to Analytic Server using Kerberos SSO.

Disabling Kerberos

1. Disable Kerberos in the Ambari console.

2. Stop the Analytic Server service.

3. Remove the following parameters from the **Analytic Server Advanced Configuration Snippet (Safety Valve) for analyticserver-conf/config.properties** area.

   ```
   default.security.provider
   hdfs.keytab
   hdfs.user
   ```

```
java.security.krb5.conf
jdbc.db.connect.method.kerberos
web.authentication.kerberos.keytab
web.authentication.kerberos.principal
```
4. Click **Save Changes** and restart the Analytic Server service.

Enabling Secure Socket Layer (SSL) connections to the Analytic Server console

By default, Analytic Server generates self-signed certificates to enable Secure Socket Layer (SSL), so you can access the Analytic Server console through the secure port by accepting self signed certificates. In order to make HTTPS access more secure, you need to install 3rd party vendor certificates.

To install 3rd party vendor certificates, follow these steps.

1. Copy the 3rd party vendor key store and trust store certificates to the same directory in all Analytic Server nodes; for example, /home/as_user/security.

   Note: The Analytic Server User must have read access to this directory.
2. In Cloudera Manager, navigate to the Configuration tab of the Analytic Server service.
3. Edit the **ssl_cfg** parameter.
```
<ssl id="defaultSSLConfig"
     keyStoreRef="defaultKeyStore"
     trustStoreRef="defaultTrustStore"
     clientAuthenticationSupported="true"/>
<keyStore id="defaultKeyStore"
          location="<KEYSTORE-LOCATION>"
          type="<TYPE>"
          password="<PASSWORD>"/>
<keyStore id="defaultTrustStore"
          location="<TRUSTSTORE-LOCATION>"
          type="<TYPE>"
          password="<PASSWORD>"/>
```

   Replace
   - <KEYSTORE-LOCATION> with the absolute location of the key store; for example: /home/as_user/security/mykey.jks
   - <TRUSTSTORE-LOCATION> with the absolute location of the trust store; for example: /home/as_user/security/mytrust.jks
   - <TYPE> with the type of the certificate; for example: JKS, PKCS12 etc.
   - <PASSWORD> with the encrypted password in Base64 encryption format. For encoding you can use the securityUtility; for example: {AS_ROOT}/ae_wlpserver/bin/securityUtility encode <password>

   If you want to generate a self-signed certificate, you can use securityUtility; for example: {AS_ROOT}/ae_wlpserver/bin/securityUtility createSSLCertificate --server=myserver --password=mypassword --validity=365 --subject=CN=mycompany,O=myOrg,C=myCountry. For more information on securityUtility and other SSL settings, refer to the WebSphere Liberty Profile documentation.
4. Click **Save Changes** and restart the Analytic Server service.

Enabling support for Essentials for R

Analytic Server supports scoring R models and running R scripts.

To install Essentials for R after a successful Analytic Server installation in Cloudera Manager:

1. Download the self-extracting archive (BIN) for IBM SPSS Modeler Essentials for R RPM. Essentials for R is available for download (https://www14.software.ibm.com/webapp/iwm/web/preLogin.do?source=swg-tspssp). Choose the file specific to your stack, stack version, and hardware architecture.

2. Run the self-extracting archive as a root or sudo user on the Cloudera Manager server host. The following packages must be installed or available from the configured repositories:
   - Red Hat Linux: gcc-gfortran, zip, gcc-c++
   - SUSE Linux: gcc-fortran, zip, gcc-c++
3. The self-extracting installer does the following tasks:
   a. Displays the required licenses and prompts the installer to accept them.
   b. Prompts the installer to input the R source location, or continue with the default location. The default R version that is installed is 3.1.0. To install a different version:
      - Online installation: Provide the URL to the required R version archive. For example, https://cran.r-project.org/src/base/R-2/R-2.15.3.tar.gz for R 2.15.3.
      - Offline installation: Download and then copy the required R version archive to the Cloudera Manager server host. Do not rename the archive (by default, it is named `R-x.x.x.tar.gz`). Provide the URL to the copied R archive as follows: `file://<R_archive_directory>/R-x.x.x.tar.gz`. If the `R-2.15.3.tar.gz` archive was downloaded and then copied to /root, the URL is `file:///root/R-2.15.3.tar.gz`.

      Note: Other R versions can be found at https://cran.r-project.org/src/base/.
   c. Installs the packages that R requires.
   d. Downloads and installs R plus the Essentials for R plugin.
   e. Creates the parcel and `parcel.sha` file and copies them to `/opt/cloudera/parcel-repo`. Enter the correct location if the location has changed.
4. After the installation is complete, distribute and activate the **Essentials for R** parcel in Cloudera Manager (click **Check for New Parcels** to refresh the parcel list).
5. If the Analytic Server service is already installed:
   a. Stop the service.
   b. Refresh the Analytic Server binaries.
   c. Start the service to finish the Essentials for R installation.
6. If the Analytic Server service is not installed, then proceed with its installation.

Note: All Analytic Server hosts must have the appropriate archive (`zip` and `unzip`) packages installed.

Enabling relational database sources

Analytic Server can use relational database sources if you supply the JDBC drivers in a shared directory on each Analytic Server host. By default, this directory is `/usr/share/jdbc`.

To change the shared directory, follow these steps.

1. In Cloudera Manager, navigate to the Configuration tab of the Analytic Server service.
2. Specify the path of the shared directory of JDBC drivers in **jdbc.drivers.location**.
3. Click **Save Changes**.
4. Select **Stop** from the **Actions** dropdown to stop the Analytic Server service.
5. Select **Refresh Analytic Server Binaries** from the **Actions** dropdown.
6. Select **Stop** from the **Actions** dropdown to start the Analytic Server service.

Table 2. Supported databases

Database	Supported versions	JDBC driver jars	Vendor
Amazon Redshift	8.0.2 or later	`RedshiftJDBC41-1.1.6.1006.jar` or later	Amazon
DB2 for Linux, UNIX, and Windows	10.5, 10.1, 9.7	`db2jcc.jar`	IBM

Table 2. Supported databases (continued)

Database	Supported versions	JDBC driver jars	Vendor
DB2 z/OS	11, 10	`db2jcc.jar,` `db2_license_cisuz.jar`	IBM
Greenplum	5, 4.2.x	postgresql.jar	Greenplum
Netezza	7, 6.x	`nzjdbc.jar`	IBM
Oracle	12c, 11g R2 (11.2)	`ojdbc6.jar, orai18n.jar`	Oracle
SQL Server	2014, 2012, 2008 R2	`sqljdbc4.jar`	Microsoft
Sybase IQ	16.x, 15.4, 15.2	`jconnect70.jar`	Sybase
Teradata	14, 14.1, 15	`tdgssconfig.jar,` `terajdbc4.jar`	Teradata

Enabling HCatalog data sources

Analytic Server provides support for a number of data sources through Hive/HCatalog. Some sources require manual configuration steps.

1. Collect the necessary JAR files to enable the data source. See the sections below for details.
2. Add these JAR files to the {`HIVE_HOME`}/`auxlib` directory and to the `/usr/share/hive` directory on each Analytic Server node.
3. Restart the Hive Metastore service.
4. Restart each and every instance of the Analytic Server service.

NoSQL databases

Analytic Server supports any NoSQL database for which a Hive storage handler is available from the vendor.

No additional steps are necessary to enable support for Apache HBase and Apache Accumulo.

For other NoSQL databases, contact the database vendor and obtain the storage handler and related jars.

File-based Hive tables

Analytic Server supports any file-based Hive tables for which a built-in or custom Hive SerDe (serializer-deserializer) is available.

The Hive XML SerDe for processing XML files is located in the Maven Central Repository at http://search.maven.org/#search%7Cga%7C1%7Chivexmlserde.

Changing ports used by Analytic Server

Analytic Server uses the 9080 port for HTTP and the 9443 port for HTTPS by default. To change the port settings, follow these steps.

1. In Cloudera Manager, navigate to the Configuration tab of the Analytic Server service.
2. Specify the desired HTTP and HTTPS ports in the **http.port** and **https.port** parameters, respectively.

   Note: You may need to select the **Ports and Addresses** category in the Filters section in order to see these parameters.
3. Click **Save Changes**.
4. Restart the Analytic Server service.

High availability Analytic Server

You can make Analytic Server highly available by adding it as a service to multiple nodes in your cluster.

1. In Cloudera Manager, navigate to the Instances tab of the Analytic Server service.
2. Click **Add Role Instances** and select the hosts on which to add Analytic Server as a service.

Optimizing JVM options for small data

You can edit JVM properties in order to optimize your system when running small (M3R) jobs.

In Cloudera Manager, see the **Jvm Options (jvm.options)** control on the Configuration tab in the Analytic Server service. Modifying the following parameters sets the heap size for jobs run on the server that hosts Analytic Server; that is, not Hadoop. This is important if running small (M3R) jobs, and you may need to experiment with these values to optimize your system.

```
-Xms512M
-Xmx2048M
```

Migration

Analytic Server allows you to migrate data and configuration settings from an existing Analytic Server installation to a new installation.

Upgrade to a new version of Analytic Server

> If you have an existing installation of Analytic Server 2.0 and have purchased a newer version, then you can migrate your 2.0 configuration settings to your new installation.

> **Restriction:** If you have a version earlier than 2.0 installed, you must first migrate from the earlier version to 2.0 and then from version 2.0 to the newer version.

> **Restriction:** Your 2.0 and new installations cannot coexist on the same Hadoop cluster. If you configure your new installation to use the same Hadoop cluster as your 2.0 installation, the 2.0 installation will no longer function.

Migration steps, 2.0 to newer version

1. Install the new installation of Analytic Server according to the instructions in "Installation on Cloudera" on page 18.
2. Copy the analytic root from your old installation to your new one.
   a. If you are unsure of the location of the analytic root, run `hadoop -fs ls`. The path to the analytic root will be of form `/user/aeuser/analytic-root`, where aeuser is the userid that owns the analytic root.
   b. Change ownership of the analytic root from aeuser to as_user

      ```
      hadoop dfs -chown -R {as_user:{group}} {path to 2.0 analytic-root}
      ```

      Note: If you plan to use the existing Analytic Server installation after migration, make a copy of the analytic-root directory in HDFS and then change the ownership on the copy of the directory.
   c. Log in to the host of the new Analytic Server installation as as_user. Delete the `/user/as_user/analytic-root` directory, if it exists.
   d. Run the following copy script.

      ```
      hadoop distcp hftp://{host of 2.0 namenode}:50070/{path to 2.0 analytic-root} hdfs://{host of 2.1 namenode}/user/as_user/analytic-root
      ```
3. In Cloudera Manager, stop the Analytic Server service.
4. Collect the configuration settings from the old installation.
   a. Copy the `configcollector.zip` archive in your new installation to `{AS_ROOT}\tools` in your old installation.

b. Extract the copy of `configcollector.zip`. This creates a new `configcollector` subdirectory in your old installation.

c. Run the configuration collector tool in your old installation by executing the **configcollector** script in `{AS_ROOT}\tools\configcollector`. Copy the resulting compressed (ZIP) file to the server that hosts your new installation.

5. Run the migration tool by executing the **migrationtool** script and passing the path of the compressed file created by the configuration collector as an argument. An example follows.

   `migrationtool.sh /opt/ibm/spss/analyticserver/2.1/ASConfiguration_2.0.0.0.xxx.zip`

6. Clear the Zookeeper state. In the Zookeper bin directory (for example, `/opt/cloudera/parcels/CDH-5.4...../lib/zookeeper/bin` on Cloudera), run the following command.

   `./zkCli.sh rmr /AnalyticServer`

7. In Cloudera Manager, start the Analytic Server service.

   Note: If you configured R for use with the existing Analytic Server installation, you will need to follow the steps to configure it with the new Analytic Server installation.

Uninstalling Analytic Server on Cloudera

Cloudera automatically handles most of the steps that are required to uninstall the Analytic Server service and parcel.

The following steps are required to cleanup Analytic Server from the Cloudera environment:

1. Stop and delete the Analytic Server Service.
2. **Deactivate** and **Remove From Hosts** the Analytic Server parcels.
3. Delete the Analytic Server user directory in HDFS. The default location is `/user/as_user/analytic-root`.
4. Delete the database, or schema, that is used by Analytic Server.

Chapter 4. MapR Installation and Configuration

MapR Overview

MapR is a complete distribution for Apache Hadoop that packages more than a dozen projects from the Hadoop ecosystem to provide a broad set of big data capabilities.

The MapR file system cannot be accessed outside of the server cluster. As a consequence, IBM SPSS Analytic Server must be deployed in the MapR cluster nodes. In this deployment scenario, Analytic Server must be run by a user who has authority to access the MapR file system and submit jobs to yarn to deploy to Analytic Server (as `<as_user>`).

Installing Analytic Server on MapR

The following steps detail the process of manually installing IBM SPSS Analytic Server on a MapR cluster.

1. Run the Analytic Server installer (`spss_as-2.1-mapr5-lx86-64_en.bin`) with a root or sudo user. Follow the installation prompts to accept the license and choose to install Analytic Server online or offline.

    a. Select the online option when the server that hosts Analytic Server has an internet connection to `http://ibm-open-platform.ibm.com`. The installer automatically installs Analytic Server.

    b. Select the offline option when the server that hosts Analytic Server does not have an internet connection to `http://ibm-open-platform.ibm.com`. Run the installer on another server that has access the URL and choose to install Analytic Server offline. The installer automatically downloads the RPM package.

2. Find and run the RPM for Analytic Server:

    ```
    rpm -ivh IBM-SPSS-AnalyticServer-2.1.0.1-1.x86_64.rpm
    ```

    For both online and offline installation modes, Analytic Server is installed to `/opt/ibm/spss/analyticserver/2.1` (as `<as_installation_path>`).

3. Change all of the files in the installation path to the user who runs Analytic Server:

    ```
    chown -R <as_user> <as_installation_path>
    ```

    Switch the user to `<as_user>`; all of the proceeding steps use `<as_user>`.

4. Configure the HTTP property. Create a file that is named `http_endpoint.xml` in the path `<as_installation_path>/ae_wlpserver/usr/servers/aeserver` and add the following lines to the file:

    ```
    <server>
        <httpEndpoint host="*" id="defaultHttpEndpoint" httpPort="<http_port>" httpsPort="<https_port>" onError="FAIL"/>
    </server>
    ```

    `<http_port>` and `<https_port>` are the ports that are used by Analytic Server via the HTTP and HTTPS protocols. Replace them with any available ports.

5. Add users and groups. Create a file that is named `security_cfg.xml` in the path `<as_installation_path>/ae_wlpserver/usr/servers/aeserver` and add the following lines to the file:

    ```
    <server>
        <basicRegistry id="basic" realm="ibm">
            <user name="admin" password="test"/>
        </basicRegistry>
    </server>
    ```

    In the default state, the XML file contains only the admin user. You must manually add other users and groups in the `<basicRegistry>` setting, or change the setting to `ldapRegistry`.

6. Setup the metadata database. Analytic Server supports the DB2 and MySQL databases.

    a. Configure the database users. When the MySQL database is used, run the following SQL script in the MySQL shell:

```
DROP DATABASE IF EXISTS <db_name>;
CREATE DATABASE <db_name> DEFAULT CHARACTER SET utf8 DEFAULT COLLATE utf8_bin;
CREATE USER '<db_username>'@'%' IDENTIFIED BY '<db_password>';
CREATE USER '<db_username>'@'localhost' IDENTIFIED BY '<db_password>';
GRANT ALL PRIVILEGES ON *.* TO '<db_username>'@'%';
GRANT ALL PRIVILEGES ON *.* TO '<db_username>'@'localhost';
```

b. Encrypt the password. The database users' passwords must be encrypted before it can be passed to Analytic Server. Run the following command:

```
java -Duser.language=en -cp <as_installation_path>/ae_wlpserver/usr/servers/aeserver/apps/AE_BOOT.war/WEB-INF/lib/*
com.spss.ae.encryption.provider.EncryptKeystorePassword <db_password>
```

Note: When the command is run directly in a Linux shell, the character * might need to be escaped as *.

The command output reads as: The encrypted password is '<encrypted_db_password>'. Record the encrypted database password.

c. Delete the file `<as_installation_path>/ae_wlpserver/usr/servers/aeserver/configuration/config.properties`, if it exists, and create a new file with the same name. Change the following properties when the DB2 database is used:

```
jndi.aedb=jdbc/aeds
jndi.aedb.url=jdbc:db2://<db_host>:<db_port>/<db_name>:currentSchema=<db_schema_name>;
jndi.aedb.driver=com.ibm.db2.jcc.DB2Driver
jndi.aedb.username=<db_username>
jndi.aedb.password=<encrypted_db_password>
```

If the `<db_schema_name>` schema does not exist, the user `<db_username>` must have implicit permission to create the schema. Change the following properties when the MySQL database is used:

```
jndi.aedb=jdbc/aeds
jndi.aedb.url=jdbc:mysql://<db_host>:<db_port>/<db_name>?createDatabaseIfNotExist=true
jndi.aedb.driver=com.mysql.jdbc.Driver
jndi.aedb.username=<db_username>
jndi.aedb.password=<encrypted_db_password>
```

d. The MySQL JDBC driver must be installed when the MySQL database is used. Run the following command:

```
yum install mysql-connector-java
```

e. Run the following command the create the required tables:

```
cd <as_installation_path>/ae_wlpserver/usr/servers/aeserver/sql/<db_type>
java -Xmx128m -Xms128m -cp <as_installation_path>/ae_wlpserver/usr/servers/aeserver/apps/AE_BOOT.war/WEB-INF/lib/*:/usr/share/java/*
com.spss.ae.dbscript.ScriptRunner ../../configuration/config.properties schema.sql true
```

The `<db_type>` is either db2 or mysql, depending on which database is used.

Note: When MySQL with the MYISAM engine is used, the second command reports the following error messages, which can be safely ignored:

```
Error executing: set global innodb_large_prefix=ON
java.sql.SQLException: Unknown system variable 'innodb_large_prefix'
Error executing: set global innodb_file_format=BARRACUDA
java.sql.SQLException: Unknown system variable 'innodb_file_format'
Error executing: set global innodb_file_format_max=BARRACUDA
java.sql.SQLException: Unknown system variable 'innodb_file_format_max'
Error executing: set global innodb_file_per_table=TRUE
java.sql.SQLException: Variable 'innodb_file_per_table' is a read only variable
```

7. Run the following command to unpack the `cf` library.

```
cd <as_installation_path>/ae_wlpserver/usr/servers/aeserver/configuration
unzip cf.zip
```

8. Configure the JAAS login modules class path by creating a file that is named `private_library.xml` in the path`<as_installation_path>/ae_wlpserver/usr/servers/aeserver` and enter the following information in the file:

```
<server>
 <library id="maprLib">
  <fileset dir="${wlp.install.dir}/usr/servers/aeserver/apps/AE_BOOT.war/WEB-INF/lib" includes="*.jar"/>
  <fileset dir="/usr/share/java" includes="*.jar"/>
  <folder dir="/opt/mapr/hadoop/hadoop-2.7.0/etc/hadoop"/>
  <fileset dir="/opt/mapr/hadoop/hadoop-2.7.0/share/hadoop/common" includes="*.jar"/>
  <fileset dir="/opt/mapr/hadoop/hadoop-2.7.0/share/hadoop/common/lib" includes="*.jar"/>
  <fileset dir="/opt/mapr/hadoop/hadoop-2.7.0/share/hadoop/hdfs" includes="*.jar"/>
  <fileset dir="/opt/mapr/hadoop/hadoop-2.7.0/share/hadoop/hdfs/lib" includes="*.jar"/>
  <fileset dir="/opt/mapr/hadoop/hadoop-2.7.0/share/hadoop/yarn" includes="*.jar"/>
  <fileset dir="/opt/mapr/hadoop/hadoop-2.7.0/share/hadoop/yarn/lib" includes="*.jar"/>
```

```
  <fileset dir="/opt/mapr/hadoop/hadoop-2.7.0/share/hadoop/mapreduce" includes="*.jar"/>
  <fileset dir="/opt/mapr/hadoop/hadoop-2.7.0/share/hadoop/mapreduce/lib" includes="*.jar"/>
 </library>
 <jaasLoginModule id="maprLoginModule1" className="org.apache.hadoop.security.login.GenericOSLoginModule"
  controlFlag="REQUIRED" libraryRef="maprLib"></jaasLoginModule>
 <jaasLoginModule id="maprLoginModule2" className="org.apache.hadoop.security.login.HadoopLoginModule"
  controlFlag="REQUIRED" libraryRef="maprLib"></jaasLoginModule>
 <jaasLoginContextEntry id="hadoop_simple" name="hadoop_simple" loginModuleRef="maprLoginModule1,maprLoginModule2" />
 <application context-root="/analyticserver" id="AS_BOOT" location="AE_BOOT.war" name="AS_BOOT" type="war">
   <classloader commonLibraryRef="maprLib"></classloader>
 </application>
 <application id="help" location="help.war" name="help" type="war" context-root="/analyticserver/help"/>
</server>
```

Note: The previous example is for configuring the `hadoop_simple login` module. The configuration must be changed when MapR uses other login modules.

9. Verify if the file ASModules.xml exists in the path `<as_installation_path>/ae_wlpserver/usr/servers/aeserver/configuration/`. If the file does not exist, rename the file ASModules.xml.template (in the same path) to `ASModules.xml`

10. Configure the cluster information by adding the following properties in the file `<as_installation_path>/ae_wlpserver/usr/servers/aeserver/configuration/config.properties`.

    ```
    ae.cluster.zookeeper.connect.string=
    ae.cluster.member.name=
    ae.cluster.collective.name=mapr_5.0
    ```

    The `ae.cluster.zookeeper.connect.string` property is the comma-separated zookeeper node list. The property can share the zookeeper cluster that is used by MapR. `ae.cluster.member.name` is the host name of the node that hosts Analytic Server.

11. Open the file `<as_installation_path>/ae_wlpserver/usr/servers/aeserver/server.env` and add the following lines to the file:

    ```
    JAVA_HOME=<java_home>

    PATH=<as_installation_path>/ae_wlpserver/usr/servers/aeserver/configuration/linux/lib_64:<java_home>/jre/lib/amd64:/usr/sbin:
    /usr/bin:/sbin:/bin

    IBM_SPSS_AS_NATIVE_PATH=<as_installation_path>/ae_wlpserver/usr/servers/aeserver/configuration/linux/lib_64

    LD_LIBRARY_PATH=<as_installation_path>/ae_wlpserver/usr/servers/aeserver/configuration/linux/lib_64:<java_home>/jre/lib/amd64:
    /opt/mapr/hadoop/hadoop-2.7.0/lib/native
    ```

    Replace `<as_installation_path>` and `<java_home>` with the actual installation path and Java home path.

12. Edit the analytic root by opening the file `<as_installation_path>/ae_wlpserver/usr/servers/aeserver/configuration/config.properties` and adding the following line:

    ```
    distrib.fs.root=<analytic_root>
    ```

    `<analytic_root>` is a path in the MapR file system that hosts the essential Analytic Server remote files. The recommended path is /user/<as_user>/analytic-root.

13. Set the administrator user by opening the file `<as_installation_path>/ae_wlpserver/usr/servers/aeserver/configuration/config.properties` and adding the following line:

    ```
    admin.username=admin
    ```

    The value must be an Analytic Server administrator user name and must be one of the users that is configured in the `security_cfg.xml` file.

14. Upload Analytic Server dependencies to the MapR file system by adding the following line at line 69 in the file `<as_installation_path>/bin/hdfsUpdate.sh`:

    ```
    JAVA_CLASS_PATH=`hadoop classpath`:$JAVA_CLASS_PATH
    ```

    Run the following commands to create the `<analytic_root>`:

    ```
    cd <as_installation_path>/bin
    ./hdfsUpdate.sh
    ```

    `<as_user>` must have write permission to the `<analytic_root>` parent directory.

15. Start and stop Analytic Server.

    a. Run the following command to start Analytic Server:

       ```
       cd <as_installation_path>/ae_wlpserver/bin
       ./server start aeserver
       ```

b. Run the following command to stop Analytic Server:

```
cd <as_installation_path>/ae_wlpserver/bin
./server stop aeserver
```

Configuring MapR

After installation, you can optionally configure and administer Analytic Server MapR features.

Enabling database pushback

Database pushback is the practice of reading data from a database and processing directly on the data.

IBM SPSS Analytic Server supports pushback for the following databases:
- DB2
- DB2 for Z
- MySQL
- Netezza
- Oracle
- PostgreSQL
- Redshift
- SQL Server
- Sybase IQ
- Terradata

Use the following steps to enable database pushback.
1. Copy the appropriate JDBC driver JAR files to `<as_installation_path>/jdbc`.
2. Open the file `<as_installation_path>/ae_wlpserver/usr/servers/aeserver/private_library.xml`, locate the tag library with the `maprLib` ID, and add the following line in the tag:

   ```
   <fileset dir="<as_installation_path>/jdbc" includes="*.jar"/>
   ```
3. Run the following commands:

   ```
   cd <as_installation_path>/jdbc
   hadoop fs -put *.jar <as_analytic_root>/cluster1/classpath
   ```
4. Restart Analytic Server.

Enabling Apache Hive

Apache Hive is a data warehouse infrastructure that is built on top of Hadoop for providing data summarization, query, and analysis.

Note: Hive must be configured to use MySQL as a metastore. The `hive-site.xml` file that exists in the node that hosts IBM SPSS Analytic Server should be same as the file in the node that runs the Hive metastore.

To enable Apache Hive support after a successful MapR installation:
1. Upload the Hive and hcatalog dependencies to the MapR file system, by running the following commands:

   ```
   cd /opt/mapr/hive/hive-1.2/lib
   hadoop fs -put *.jar <as_analytic_root>/cluster1/classpath
   cd /opt/mapr/hive/hive-1.2/hcatalog/share/hcatalog
   hadoop fs -put *.jar <as_analytic_root>/cluster1/classpath
   ```

   `<as_analytic_root>` is the analytic root path that is defined in "Installing Analytic Server on MapR" on page 29.
2. Open the file `<as_installation_path>/ae_wlpserver/usr/servers/aeserver/private_library.xml`, locate the tag library with the ID `maprLib`, and add the following lines in the tag:

```
<fileset dir="/opt/mapr/hive/hive-1.2/lib" includes="*.jar"/>
<fileset dir="/opt/mapr/hive/hive-1.2/hcatalog/share/hcatalog" includes="*.jar"/>
```

3. Run the following commands to create Hive and hcatalog configuration file links:

```
mkdir <as_installation_path>/ae_wlpserver/usr/servers/aeserver/configuration/hive-conf
ln -s /opt/mapr/hive/hive-1.2/conf/* <as_installation_path>/ae_wlpserver/usr/servers/aeserver/configuration/hive-conf
```

4. Add the following line to the `private_library.xml` file when there are extra jar files in the auxlib for Hive:

```
<fileset dir="/opt/mapr/hive/hive-1.2/auxlib" includes="*.jar"/>
```

   Run the following commands after adding the previous line:

```
cd /opt/mapr/hive/hive-1.2/auxlib
hadoop fs -put *.jar <as_analytic_root>/cluster1/classpath
```

5. Restart Analytic Server.

Enabling Apache HBase

Apache HBase is an open source, non-relational, distributed database that is written in Java. It is developed as part of Apache Software Foundation's Apache Hadoop project and runs on top of HDFS (Hadoop Distributed Filesystem).

To enable Apache HBase support after a successful MapR installation:

1. Upload the HBase dependencies to the MapR file system and run the following commands:

```
cd /opt/mapr/hbase/hbase-0.98.12/lib
hadoop fs -put *.jar <as_analytic_root>/cluster1/classpath
```

   `<as_analytic_root>` is the analytic root path that is defined in "Installing Analytic Server on MapR" on page 29.

2. Open the file `<as_installation_path>/ae_wlpserver/usr/servers/aeserver/private_library.xml`, locate the tag library with the ID `maprLib`, and add the following line in the tag:

```
<fileset dir="/opt/mapr/hbase/hbase-0.98.12/lib" includes="*.jar"/>
```

3. Run the following commands to create HBase and hcatalog configuration file links:

```
mkdir <as_installation_path>/ae_wlpserver/usr/servers/aeserver/configuration/hbase-conf
ln -s /opt/mapr/hbase/hbase-0.98.12/conf/* <as_installation_path>/ae_wlpserver/usr/servers/aeserver/configuration/hbase-conf
```

4. Restart IBM SPSS Analytic Server.

Enabling Apache Spark

Apache Spark is an open standard for flexible in-memory data processing for batch, real-time, and advanced analytics.

To enable Apache Spark support after a successful MapR installation:

1. Copy the file `spark-assembly-1.4.1-hadoop2.5.1-mapr-1501.jar` from `/opt/mapr/spark/spark-1.4.1/lib` to `<as_installation_path>/ae_wlpserver/usr/servers/aeserver/modules/spark/`.

2. Upload the Spark dependencies to the MapR file system and run the following commands:

```
cd <as_installation_path>/ae_wlpserver/usr/servers/aeserver/modules/spark/
hadoop fs -put *.jar <as_analytic_root>/cluster1/classpath
```

   `<as_analytic_root>` is the analytic root path that is defined in "Installing Analytic Server on MapR" on page 29.

3. Open the file `<as_installation_path>/ae_wlpserver/usr/servers/aeserver/private_library.xml`, locate the tag library with the ID `maprLib`, and add the following line in the tag:

```
<fileset dir="/opt/mapr/spark/spark-1.4.1/lib" includes="spark-assembly-*.jar"/>
```

4. Run the following commands to create Spark configuration file links:

```
mkdir <as_installation_path>/ae_wlpserver/usr/servers/aeserver/configuration/spark-conf
ln -s /opt/mapr/spark/spark-1.4.1/conf/* <as_installation_path>/ae_wlpserver/usr/servers/aeserver/configuration/spark-conf
```

5. Add the following line in the file `<as_installation_path/ae_wlpserver/usr/servers/aeserver/server.env`:

```
SPARK_HOME=/opt/mapr/spark/spark-1.4.1
```

6. Add the following line in the file `<as_installation_path>/ae_wlpserver/usr/servers/aeserver/configuration/config.properties`:

```
spark.executor.extraLibraryPath=/opt/mapr/hadoop/hadoop-2.7.0/lib/native
```

7. Restart IBM SPSS Analytic Server.

8. To enable the PySpark feature, add the following line in the `yarn-env.sh` file, and then restart ResourceManagers and NodeManagers:

```
export SPARK_HOME=/opt/mapr/spark/spark-1.4.1
```

Enabling feature flags

Feature flags provide the capability of enabling and disabling specific application features.

To enable feature flag support after a successful MapR installation:

1. Add the following line in the file `<as_installation_path>/ae_wlpserver/usr/servers/aeserver/configuration/config.properties`:

```
load.feature.flags.on.msg=true
```

2. Restart IBM SPSS Analytic Server.

Enabling R

R is a language and environment for statistical computing and graphics.

To enable R support after a successful MapR installation:

Note: The following package must be installed before you can run the installer on all cluster nodes that host Node Manager and IBM SPSS Analytic Server:

```
gcc-gfortran
libgfortran
gcc-c++
```

1. Run the installer `spss_er-8.2-mapr5-lx86_64_en.bin` on all the cluster nodes that host Node Manager and Analytic Server. The user that runs the installer must have write permission to the R and Analytic Server installation paths.

2. Follow the installation instructions by accepting the license agreement and enter the required information. If Analytic Server is installed on the installation server, choose `Yes` when prompted and input `<as_installation_path>`. If Analytic Server is not installed on the installation server, choose `No` when prompted.

3. When Analytic Server is installed, Essentials for R is automatically installed in the Analytic Server installation path.

   - If Analytic Server is not installed, Essentials for R is installed to the path `<installer_path>/IBM_SPSS_ModelerEssentialsR/linux`.

   - If Analytic Server is installed later, use the following command to copy Essentials for R to the Analytic Server configuration path where Analytic Server is installed.

     ```
     cp -r <installer_path>/IBM_SPSS_ModelerEssentialsR/linux <as_installation_path>/ae_wlpserver/usr/servers/aeserver/configuration
     ```

4. Delete the `cf.zip` file in the path `<as_installation_path>/ae_wlpserver/usr/servers/aeserver/configuration` and generate a new file with the following commands:

```
cd <as_installation_path>/ae_wlpserver/usr/servers/aeserver/configuration
zip -r cf.zip linux
```

5. Run the following commands:

```
cd <as_installation_path>/ae_wlpserver/usr/servers/aeserver/configuration
hadoop fs -rm <as_analytic_root>/cluster1/configuration/cf.zip
hadoop fs -put cf.zip <as_analytic_root>/cluster1/configuration/
```

6. Restart Analytic Server.

Enabling LZO

LZO is a lossless data compression library that favors speed over compression ratio. MapR must be manually configured to provide LZO support.

The following site provides LZO installation and configuration instructions: https://github.com/twitter/hadoop-lzo.

The following steps detail the process of importing an LZO library into MapR.

1. Copy the `hadoop-lzo-<version>.jar` file to the Hadoop class path. The recommended path is `/opt/mapr/hadoop/hadoop-2.7.0/share/hadoop/mapreduce/lib`.

2. Copy the native files `libgplcompression.so` and `liblzo2.so.2` to `/opt/mapr/hadoop/hadoop-2.7.0/lib/native`, and add the following properties to the `core-site.xml` file:

   ```
   <property>
      <name>io.compression.codecs</name>
      <value>org.apache.hadoop.io.compress.GzipCodec,org.apache.hadoop.io.compress.DefaultCodec,com.hadoop.compression.lzo.
      LzoCodec,com.hadoop.compression.lzo.LzopCodec,org.apache.hadoop.io.compress.BZip2Codec</value>
   </property>
   <property>
      <name>io.compression.codec.lzo.class</name>
      <value>com.hadoop.compression.lzo.LzoCodec</value>
   </property>
   ```

3. Open the file `<as_installation_path>/ae_wlpserver/usr/servers/aeserver/server.env` and add `<lzo_native_path>` to the `LD_LIBRARY_PATH` parameter. `<lzo_native_path>` is the folder that contains the Hadoop-LZO native library.

   ```
   LD_LIBRARY_PATH=<as_installation_path>/ae_wlpserver/usr/servers/aeserver/configuration/linux/lib_64:<java_home>/jre/lib/amd64:
   /opt/mapr/hadoop/hadoop-2.7.0/lib/native:<lzo_native_path>
   ```

4. Restart IBM SPSS Analytic Server.

Setting up an IBM SPSS Analytic Server cluster for MapR

Use the following steps to setup an IBM SPSS Analytic Server cluster environment for MapR support.

1. Add the following line in the file `<as_installation_path>/ae_wlpserver/usr/servers/aeserver/configuration/config.properties`.

   ```
   enable.resume=true
   ```

2. Copy the installation path to the other cluster nodes and change the `ae.cluster.member.name` property in the `config.properties` file to the correct host name.

3. Start all of the cluster nodes.

Uninstalling MapR

The following steps explain the process of uninstalling MapR:

1. Stop IBM SPSS Analytic Server.

2. Delete the metadata database.

   a. Run the following commands:

   ```
   cd <as_installation_path>/ae_wlpserver/usr/servers/aeserver/sql/<db_type>
   java -Xmx128m -Xms128m -cp <as_installation_path>/ae_wlpserver/usr/servers/aeserver/apps/AE_BOOT.war/WEB-INF/lib/*:/usr/share/java/*
   com.spss.ae.dbscript.ScriptRunner ../../configuration/config.properties drop.sql true
   ```

   b. Run the following SQL statement to drop the database:

   ```
   drop database <db_name>
   ```

3. Uninstall the RPM package:

   ```
   rpm -e IBM-SPSS-AnalyticServer-2.1.0.1-1.x86_64
   ```

4. Delete the installation path:

   ```
   rm -r <as_installation_path>
   ```

5. Delete the analytic root:

   ```
   hadoop fs -rm -r <analytic-root>
   ```

6. Delete the zookeeper data:

```
/opt/mapr/zookeeper/zookeeper-3.4.5/bin/zkCli.sh -server <zookeeper_host>:<zookeeper_port>
rmr /AnalyticServer
```

Migrating IBM SPSS Analytic Server on MapR

IBM SPSS Analytic Server can be migrated on MapR.

Use the following steps to migrate IBM SPSS Analytic Server 2.0 to version 2.1 on MapR.

1. Install Analytic Server 2.1 on a MapR cluster by following the installation instructions in "Installing Analytic Server on MapR" on page 29.

2. Copy the analytic root.

   Note: This step can be ignored if the analytic root is not changed.

   - Run the following command on one of the data nodes if the analytic root for both Analytic Server versions 2.0 and 2.1 are on the same MapR cluster:

     ```
     hadoop fs -cp <old_analytic_root>/analytic-workspace/* <new_analytic_root>/analytic-workspace
     ```

   - The installed WEBHDFS or NFS services dictate when the analytic root for Analytic Server versions 2.0 and 2.1 are on different MapR clusters. WEBHDFS or NFS are required to copy the analytic root date because the MapR file system cannot be accessed directly outside the cluster.

     a. Run the following command on one of the new Analytic Server 2.1 cluster nodes when the old Analytic Server 2.0 cluster includes the WEBHDFS service:

        ```
        hadoop distcp webhdfs://<webhdfs_server>:<webhdfs_port>/<old_analytic_root>/analytic-workspace/*
        maprfs://<new_analytic_root>/analytic-workspace
        ```

     b. Run the following command on one of the old Analytic Server 2.0 cluster nodes when the new Analytic Server 2.1 cluster includes the WEBHDFS service:

        ```
        hadoop distcp maprfs://<old_analytic_root>/analytic-workspace/*
        webhdfs://<webhdfs_server>:<webhdfs_port>/<new_analytic_root>/analytic-workspace
        ```

     c. Run the following command one of the old Analytic Server 2.0 cluster nodes when the old cluster includes NFS, and NFS is also mounted on one of the new Analytic Server 2.1 cluster nodes:

        ```
        hadoop distcp file:///<mount_path>/<old_analytic_root>/analytic-workspace/* maprfs://<new_analytic_root>/analytic-workspace
        ```

     d. Run the following command one of the new Analytic Server 2.1 cluster nodes when the new cluster includes NFS, and NFS is also mounted on one of the old Analytic Server 2.0 cluster nodes:

        ```
        hadoop discp maprfs://<old_analytic_root>/analytic-workspace/* file:///<mount_path>/<new_analytic_root>/analytic-workspace
        ```

   Review the MapR Data Migration site for information on migrating data between different MapR clusters.

3. Run the following commands to change the new analytic root's owner and permissions:

   ```
   hadoop fs -chown -R <as_user> <analytic_root>
   hadoop fs -chmod -R 755 <>
   ```

4. Stop Analytic Server 2.1, but ensure that the metadata database is still running.

5. Collect the configuration settings from the old Analytic Server 2.0 cluster installation.

   a. Copy the configcollector.zip archive from the new Analytic Server 2.1 cluster installation to <old_as_installation_path>/tools on the old Analytic Server 2.0 cluster installation.

   b. Extract the configcollector.zip contents on the old Analytic Server 2.0 cluster installation. A new configcollector subdirectory is created in the old Analytic Server 2.0 cluster installation.

   c. Run the configuration collector tool in the old Analytic Server 2.0 cluster installation by running the configcollector script from <old_as_installation_path>/tools/configcollector. Copy the resulting compressed (ZIP) file to the new Analytic Server 2.1 cluster installation.

6. Run the migration tool on the new Analytic Server 2.1 cluster by running the migrationtool script and passing the path of the compressed file, that was created by the configuration collector, as an argument. For example:

   ```
   migrationtool.sh /opt/ibm/spss/analyticserver/2.1/ASConfiguration_2.0.0.0.xxx.zip
   ```

7. Start Analytic Server 2.1.

MapR troubleshooting

This section describes some common MapR installation and configuration issues and how you can fix them.

Issues with the hdfsUpdate.sh script

The hdfsUpdate.sh script must be run only one time because the script removes all files in the analytic-root before it uploads new files. When the script is run more than one time, you must re-upload the dependencies for database push-back, Hive, HBase, and Spark. Run the following commands to re-upload the required dependencies:

```
cd <as_installation_path>/jdbc

hadoop fs -put *.jar <as_analytic_root>/cluster1/classpath

cd /opt/mapr/hive/hive-1.2/lib
hadoop fs -put *.jar <as_analytic_root>/cluster1/classpath
cd /opt/mapr/hive/hive-1.2/hcatalog/share/hcatalog
hadoop fs -put *.jar <as_analytic_root>/cluster1/classpath

cd /opt/mapr/hbase/hbase-0.98.12/lib
hadoop fs -put *.jar <as_analytic_root>/cluster1/classpath

cd <as_installation_path>/ae_wlpserver/usr/servers/aeserver/modules/spark/

hadoop fs -put *.jar <as_analytic_root>/cluster1/classpath
```

Chapter 5. Configuring IBM SPSS Modeler for use with IBM SPSS Analytic Server

In order to enable SPSS Modeler for use with Analytic Server, you need to make some updates to the SPSS Modeler Server installation.

1. Configure SPSS Modeler Server to associate it with an Analytic Server installation.

    a. Edit the `options.cfg` file in the `config` subdirectory of the main server installation directory, and add or edit the following lines:

    ```
    as_ssl_enabled, {Y|N}
    as_host, "{AS_SERVER}"
    as_port, PORT
    as_context_root, "{CONTEXT-ROOT}"
    as_tenant, "{TENANT}"
    as_prompt_for_password, {Y|N}
    as_kerberos_auth_mode, {Y|N}
    as_kerberos_krb5_conf, {CONF-PATH}
    as_kerberos_krb5_spn, {AS-SPN}
    ```

    as_ssl_enabled
    > Specify Y if secure communication is configured on Analytic Server; otherwise, N.

    as_host
    > The IP address of the server that hosts Analytic Server.

    as_port
    > The port on which Analytic Server is listening (by default this is 8080).

    as_context_root
    > The Analytic Server context root (by default this is analyticserver).

    as_tenant
    > The tenant the SPSS Modeler Server installation is a member of (the default tenant is ibm).

    as_prompt_for_password
    > Specify N if the SPSS Modeler Server is configured with the same authentication system for users and passwords as that used on Analytic Server; for example, when using Kerberos authentication. Otherwise, specify Y.
    >
    > When running SPSS Modeler in batch mode, you add `-analytic_server_username {ASusername} -analytic_server_password {ASpassword}` as arguments to the `clemb` command.

    as_kerberos_auth_mode
    > Specify Y to enable Kerberos SSO from SPSS Modeler.

    as_kerberos_krb5_conf
    > Specify the path to the Kerberos configuration file that Analytic Server should use; for example, \etc\krb5.conf.

    as_kerberos_krb5_spn
    > Specify the Analytic Server Kerberos SPN; for example, HTTP/ ashost.mydomain.com@MYDOMAIN.COM.

    b. Restart the SPSS Modeler Server service.

    In order to connect to an Analytic Server installation that has SSL/TLS enabled, there are some further steps to configuring your SPSS Modeler Server and client installations.

    a. Navigate to `http{s}://{HOST}:{PORT}/{CONTEXT-ROOT}/admin/{TENANT}` and log on to the Analytic Server console.

b. Download the certification file from the browser and save it to your file system.

c. Add the certification file to the JRE of both your SPSS Modeler Server and SPSS Modeler Client installations. The location to update can be found under the `/jre/lib/security/cacerts` subdirectory of the SPSS Modeler installation path.

   1) Make sure the `cacerts` file is not read-only.

   2) Use the keytool program Modeler ships with – this can be found in the `/jre/bin/keytool` subdirectory of the SPSS Modeler installation path.

      Run the following command

      ```
      keytool -import -alias <as-alias> -file <cert-file> -keystore "<cacerts-file>"
      ```

      Note that <as-alias> is an alias for the `cacerts` file. You can use any name you like as long as it is unique to the `cacerts` file.

      So an example command would look like the following.

      ```
      keytool -import -alias MySSLCertAlias -file C:\Download\as.cer
              -keystore "c:\Program Files\IBM\SPSS\Modeler\{ModelerVersion}\jre\lib\security\cacerts"
      ```

d. Restart your SPSS Modeler Server and SPSS Modeler Client .

2. [optional] Install IBM SPSS Modeler - Essentials for R , if you plan to score R models in streams with Analytic Server data sources. IBM SPSS Modeler - Essentials for R is available for download (https://www14.software.ibm.com/webapp/iwm/web/preLogin.do?source=swg-tspssp).

Chapter 6. Troubleshooting

This section describes some common installation and configuration issues and how you can fix them.

General issues

Installation succeeds with warnings, but users are unable to create data sources with error "Unable to complete the request. Reason: Permission denied"

Setting the **distrib.fs.root** parameter to a directory that the Analytic Server user (by default, as_user) doesn't have access to will result in errors. Make certain that the Analytic Server user is authorized to read, write, and execute the **distrib.fs.root** directory.

Issues with specific Hadoop distributions

Refresh action for Analytic Server service is disabled on Hortonworks 2.3

To manually refresh Analytic Server libraries on Hortonworks 2.3 use the following steps.

1. Log on to the host running the Analytic Metastore as the Analytic Server user (by default as_user).

   Note: You can find this host name from the Ambari console.

2. Run the **refresh** script in the directory {AS_ROOT}/bin; for example:

   ```
   cd /opt/ibm/spss/analyticserver/2.1/bin
   ./refresh
   ```

3. Restart the Analytic Server service in the Ambari console.

Issues with the metadata repository

Operation CREATE USER fails when running the add_mysql_user script

Before running the **add_mysql_user** script, you need to first manually remove the user that you are attempting to add from the mysql database. You can remove the users via the MySQL Workbench UI or via MySQL commands. For example:

```
mysql -u root -e "DROP USER '$AEDB_USERNAME_VALUE'@'localhost';"
mysql -u root -e "DROP USER '$AEDB_USERNAME_VALUE'@'$METASTORE_HOST';"
mysql -u root -e "DROP USER '$AEDB_USERNAME_VALUE'@'%';"
```

In the above commands, replace $AEDB_USERNAME_VALUE with the username you want removed, and replace $METASTORE_HOST with the host name the database is installed on.

High availability clusters

Analytic Server cannot be added to more hosts due to changes in dependencies

Run the update_clientdeps script using the instructions in "Updating client dependencies" on

java.net.SocketTimeoutException: Read timed out

Change the Liberty ND timeout environment variable as follows:

```
export LIBERTYND_READ_TIMEOUT=<milliseconds>
```

where <milliseconds> is the number of seconds to use for the JMX read timeout.

java.io.IOException: CWWKX7202E: The timeout value 60 (seconds) for command ./server start expired

Add the following to the Controller Server server.xml

```
<!-- Increase start and stop server timeout to accommodate slow hardware -->
<serverCommands startServerTimeout="120" stopServerTimeout="120"/>
```

java.lang.OutOfMemoryError: Java heap space

Add the following lines to `jvm.options` on every member of the HA cluster.

```
-Xms512M
-Xmx2048M
```

"The Analytic Cluster Service has unexpectedly lost contact with Zookeeper, this JVM is being terminated to maintain cluster integrity."

One thing that may cause this is if the amount of data being written to Zookeeper is too large. If, in the Zookeeper logs are exceptions like:

```
java.io.IOException: Unreasonable length = 2054758
```

or in the Analytic Server logs are messages like:

```
Caused by: java.io.UTFDataFormatException: encoded string too long: 2054758 bytes
    at java.io.DataOutputStream.writeUTF(DataOutputStream.java:375)
```

1. In the Ambari console, navigate to the Zookeeper service Configs tab and add the following line to the env-template, then restart the Zookeeper service.

   ```
   export JVMFLAGS="-Xmx2048m -Djute.maxbuffer=2097152"
   ```

2. In the Ambari console, navigate to the Analytic Server service Configs tab and add the following in the Advanced analytics-jvm-options, then restart the Analytic Cluster service.

   ```
   -Djute.maxbuffer=2097152
   ```

The number to specify for the jute.maxbuffer setting should be higher than the number indicated in the exception messages.

Zookeeper transaction data becomes unmanageable

Set the **autopurge.purgeInterval** parameter in `zoo.cfg` to 1 to enable automatic purges of the Zookeeper transaction log.

Analytic cluster service loses contact with Zookeeper

Review and modify the **tickTime**, **initLimit**, and **syncLimit** parameters in `zoo.cfg`. For example:

```
# The number of milliseconds of each tick
tickTime=2000
# The number of ticks that the initial
# synchronization phase can take
initLimit=30
# The number of ticks that can pass between
# sending a request and getting an acknowledgement
syncLimit=15
```

See the Zookeeper documentation for details: https://zookeeper.apache.org/doc/r3.3.3/zookeeperAdmin.html

Analytic Server jobs do not resume

There are two common situations in which Analytic Server jobs do not resume.

1. When an Analytic Server job fails because a cluster member fails, the job is normally restarted automatically on another cluster member. If the job does not resume, check to ensure there are at least 4 cluster members in the High Availability cluster.

2. When you quiesce a cluster member, then all of the Analytic Server jobs on that server are normally resumed on another cluster member. In order to ensure that the jobs resume, set `-Dcom.spss.ae.remoteclient.failover.threshold=100` and use remote mode.

Analytic Server servers hang occasionally upon server shutdown

Kill the server manually.

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
1623-14, Shimotsuruma, Yamato-shi
Kanagawa 242-8502 Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who want to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Software Group
ATTN: Licensing
200 W. Madison St.
Chicago, IL; 60606
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

ITIL is a registered trademark, and a registered community trademark of The Minister for the Cabinet Office, and is registered in the U.S. Patent and Trademark Office.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

Linear Tape-Open, LTO, the LTO Logo, Ultrium, and the Ultrium logo are trademarks of HP, IBM Corp. and Quantum in the U.S. and other countries.

IBM ®

Printed in USA