IBM SPSS Analytic Server
Version 1.0.1

Administrator's Guide

Note
Before using this information and the product it supports, read the information in "Notices" on page 21.

Product Information

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

Contents

Chapter 1. Overview

IBM® SPSS® Analytic Server is a solution for big data analytics that combines IBM SPSS technology with big data systems and allows you to work with familiar IBM SPSS user interfaces to solve problems on a previously unattainable scale.

Why big data analytics matters

Data volumes collected by organizations are growing exponentially; for example, financial and retail businesses have all customer transactions for a year (or two years, or ten years), telco providers have call data records (CDR) and device sensor readings, and internet companies have the results of web crawls.

Big data analytics is needed where there exists:

- A large volume of data (terabytes, petabytes, exabytes), especially when it is a mixture of structured & unstructured data
- Rapidly changing/accumulating data

Big data analytics also assists when:

- A large number (thousands) of models are being built
- Models are frequently built/refreshed

Challenges

The same organizations that collect large volumes of data often have difficulty actually making use of it, for a variety of reasons:

- The architecture of traditional analytic products are not suited to distributed computation, and
- Existing statistical algorithms are not designed to work with big data (these algorithms expect the data to come to them, but big data is too costly to move), thus
- Performing state of the art analytics on big data requires new skills and intimate knowledge of big data systems. Very few analysts have these skills.
- In-memory solutions work for medium-size problems, but do not scale well to truly big data.

Solution

Analytic Server provides:

- A data-centric architecture that leverages big data systems, such as Hadoop Map/Reduce with data in HDFS.
- A defined interface to incorporate new statistical algorithms designed to go to the data.
- Familiar IBM SPSS user interfaces that hide the details of big data environments so that analysts can focus on analyzing the data.
- A solution that is scalable to any size problem.

Architecture

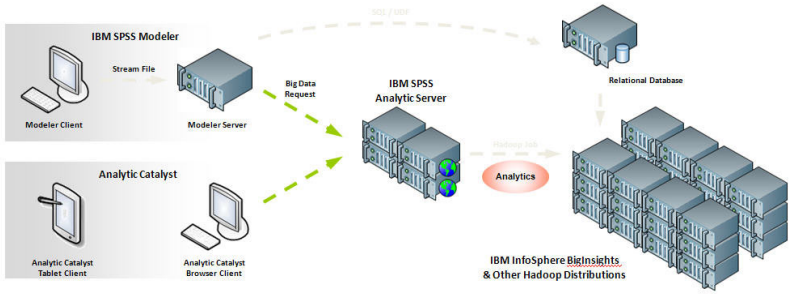

Figure 1. Architecture

The Analytic Server sits between a client application and Hadoop cloud. Assuming that the data resides in the cloud, the general outline for working with the Analytic Server is to:

1. Define Analytic Server data sources over the data in the cloud.
2. Define the analysis you want to perform in the client application. For the current release, the client applications are IBM SPSS Modeler and IBM SPSS Analytic Catalyst.
3. When you run the analysis, the client application submits an Analytic Server execution request.
4. The Analytic Server orchestrates the job to run in the Hadoop cloud and reports the results to the client application.
5. You can use the results to define further analyses, and the cycle repeats.

Chapter 2. Configuration

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 the following lines:

      ```
      as_url, http://{AS_SERVER}:{PORT}/admin/{TENANT}
      as_prompt_for_password, {Y|N}
      ```

      as_url The URL of the Analytic Server, including the IP address of the server, the port, and the tenant the SPSS Modeler server installation is a member of.

      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; otherwise, Y.

      When running SPSS Modeler in batch mode, you add `-analytic_server_username {ASusername} -analytic_server_password {ASpassword}` as arguments to the `clemb` command.

   b. Restart the SPSS Modeler server service.

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

   a. Navigate to `http://<host>:<port>/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).

Adding JDBC drivers

In order to support database data sources, you must add the JDBC drivers to the Analytic Server.

1. Stop the Analytic Server by running `{AS_ROOT}/bin/stop.sh`

2. Copy the required JDBC driver jars to {AS_ROOT}/ae_wlpserver/usr/servers/aeserver/apps/AE_BOOT.war/WEB-INF/lib
3. Update Analytic Server by running {AS_ROOT}/bin/hdfsUpdate.sh
4. Start Analytic Server by running {AS_ROOT}/bin/start.sh

Table 1. Supported databases

Database	Supported versions	JDBC driver jars	Vendor
DB2 for Linux, UNIX, and Windows	9.5, 9.7, 10.0	db2jcc.jar	IBM
DB2 z/OS	10	db2jcc.jar, db2_license_cisuz.jar	IBM
Teradata	13.1, 14	tdgssconfig.jar, terajdbc4.jar	Teradata
SQL Server	2012, 2008 R2	sqljdbc4.jar	Microsoft
Netezza	6.x, 7	nzjdbc.jar	IBM
Oracle	12g, 11g R2	ojdbc6.jar, orai18n.jar	Oracle

WebSphere Liberty configuration

WebSphere Liberty Profile is a lightweight implementation of IBM WebSphere. Analytic Server can use WebSphere application security to authenticate users. This is set up in the server.xml for the server Analytic Server is deployed. To enable application security in Liberty, the appSecurity-1.0 feature must be included in the feature manager:

```
<featureManager onError="FAIL">
...
<feature>appSecurity-1.0</feature>
...
</featureManager>
```

To enable SSL on a server, the SSL feature must be included in the server.xml file:

```
<featureManager>
     <feature>ssl-1.0</feature>
</featureManager>
```

You can find detailed information about WebSphere security at: ftp://ftp.software.ibm.com/software/webserver/appserv/library/v85/was85base_security.pdf.

Basic registry

The Basic Registry allows the administrator to define a database of users and groups within the {AS_SERVER_ROOT}/server.xml file. Passwords can be encoded to obfuscate their values with the securityUtil tool, which is located in {AS_ROOT}/ae_wlpserver/bin.

The Basic Registry is useful in a sandbox environment, but is not recommended for a production environment.

```
<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>
```

LDAP registry configuration

The LDAP Registry provides the administrator a way 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`.

Authentication using a Kerberos security provider

Before you can configure Kerberos, you must obtain the following information from your Hadoop administrator:

1. Kerberos realm; for example, `ASSSO.COM`
2. Kerberos Key Distribution Center(KDC) host name; for example, `kdc.assso.com`
3. Name node Kerberos principal; for example, `hdfs/namenode.assso.com@ASSSO.COM`.
4. MapReduce node Kerberos principal; for example, `mapred/jobtracker.assso.com@ASSSO.COM`.

Then, you must configure the `krb5.conf` file at `/etc/krb5.conf`; for example:

```
[libdefaults]
 default_realm = ASSSO.COM
 default_tkt_enctypes = rc4-hmac des-cbc-md5
 default_tgs_enctypes = rc4-hmac des-cbc-md5
 dns_lookup_realm = false
 dns_lookup_kdc = false
 ticket_lifetime = 24h
 forwardable = yes

[realms]
ASSSO.COM = {
  kdc = kdc.assso.com:88
  default_domain = assso.com
 }

[dmain_realm]
 .assso.com = ASSSO.COM
 assso.com = ASSSO.COM
```

After you create the krb5.conf file, modify the config.properties file as follows:

1. In the Analytic Server modules section:
   - Add the hdfsauth and kerberossecurityprovider modules
   - Remove the wssecurityprovider module.
2. Add the following configuration properties:

   ```
   #Kerberos authentication parameters
   hadoop.security.authentication=kerberos
   dfs.namenode.kerberos.principal=hdfs/namenode.assso.com@ASSSO.COM
   mapreduce.jobtracker.kerberos.principal=mapred/jobtracker.assso.com@ASSSO.COM
   java.security.krb5.conf=/etc/krb5.conf
   ```

   where

   hadoop.security.authentication
   Hadoop security authentication. Specify kerberos to enable Kerberos security provider.

   dfs.namenode.kerberos.principal
   Kerberos principal that is used for the keytab file, which is used start the Name node.

   mapreduce.jobtracker.kerberos.principal
   Kerberos principal that is used for the keytab file, which is used start the job tracker.

   java.security.krb5.conf
   Kerberos configuration file location.
3. Configure the Liberty LDAP user repository in server.xml. Refer to "WebSphere Liberty configuration" on page 4. All the users that are specified in the Liberty user repository must be matched with Kerberos user accounts and the same LDAP setting must be used in Kerberos server.
4. By default Analytic Server uses the .temp directory under the user's home directory as a temporary directory, but if you want to configure the temporary directory in a different location, then follow these instructions.
   a. Edit config.properties and uncomment the following configuration setting.

      ```
      #as.temp.folder=/.temp
      ```

      Change the setting as needed to the absolute path of the temporary directory. No changes are needed if the /.temp directory is used.
   b. Change permissions to this folder to allow all users access to this folder; for example, hadoop fs -chmod 777 /.temp.
5. Give read permission to the classpath and configuration folders for all Kerberos users.
   a. Open the config.properties file and note the settings for the parameters **hdfs.classpath.folder** and **component.framework.bin.path**.
   b. Set the read permissions as follows.

      For example, if **hdfs.classpath.folder**=/user/hdpadmin/classpath and **component.framework.bin.path**=/user/hdpadmin/configuration, then run

      ```
      hadoop fs -chmod -R 755 /user/hdpadmin
      hadoop fs -chmod -R 755 /user/hdpadmin/classpath
      hadoop fs -chmod -R 755 /user/hdpadmin/configuration
      ```

Note: Analytic Server does not work with HiveServer2. When running Analytic Server, the value of **hive.metastore.sasl.enabled** in the hive-site.xml file is set to false. If you want to run a command in the Hive shell on the Hadoop namenode, change the value to true and restart HCatalog, and use Hive. Once done, change the property back to false and restart HCatalog.

Enabling High Availability (HA) mode in Hadoop 2.0

Analytic Server supports running Hadoop custom modes, like High Availability (HA), by providing the cluster client API configuration into the Analytic Server configuration folder. Providing the Hadoop cluster client API configuration to Analytic Server is mandatory only if there are custom settings, like HA. These settings are required so that the Analytic Server API can balance the requests if a cluster namenode fails.

In order to configure the custom mode after a successful Analytic Server installation:

1. Obtain the cluster client configuration files (`hdfs-site.xml` and `core-sites.xml`), usually located on a namenode machine in `/etc/hadoop/conf`.
2. Copy these files into `{AS_ROOT}/ae_wlpserver/usr/servers/aeserver/configuration/hadoop-conf`.
3. If the Hadoop cluster is configured in HA mode, make sure the Analytic Server configuration property (in the `config.properties` file) `hdfs.namenode.url` is pointing to the HDFS service name and /user/Username; for example, `hdfs://nameservice1/user/hdpadmin`.
4. If you have set up HA for the job tracker service, then update the **`mapred.job.tracker`** property in `config.properties` to point to the HDFS service name.
5. Update the Hadoop file system by executing the command:
   `{AS_ROOT}/bin/hdfsUpdate.sh`
6. Start the Analytic Server by executing the command:
   `{AS_ROOT}/bin/run.sh`

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. Install R Engine on the server that hosts Analytic Server by using the following steps:

```
wget http://cran.r-project.org/src/base/R-2/R-2.15.2.tar.gz
tar -xzvf R-2.15.2.tar.gz
cd R-2.15.2
./configure --enable-R-shlib
make
```

2. Install Essentials for R on the server that hosts Analytic Server by running the installer file `install.bin`, following the instructions on the screen. Essentials for R is available for download (https://www14.software.ibm.com/webapp/iwm/web/preLogin.do?source=swg-tspssp). The installer:
   a. Updates the R Engine installation on Analytic Server; it adds the "R plug-in", and
   b. Updates the `{AS_ROOT}/ae_wlpserver/usr/servers/aeserver/configuration/ext_64/bin` directory to add a native library and a configuration file to the pasw.rstats module.
3. Deploy R Engine and R Component to Hadoop.
   a. If the Analytic Server and all the Hadoop nodes have the same version of the operating system and the same processor architecture:
      1) Create an archive for the R Engine by using the following script

```
#!/usr/bin/env bash
echo Creating R.zip...
cd /tmp
rm -r -f R
rm -f R.zip
mkdir R
export R_HOME=/home/hdpadmin/APPS/R/R-2.15.2
cp -r $R_HOME/* ./R/
cp $(ldd ./R/bin/exec/R ./R/bin/Rscript|cut -d\  -f3|grep \.so\.|sort -u) ./R/lib/
```

```
cp -P /usr/lib64/libgfortran.so.3 ./R/lib
cp /usr/lib64/libgfortran.so.3.0.0 ./R/lib
rm -r ./R/doc ./R/src ./R/include ./R/tests
zip -r R.zip R
```

2) Copy the archive `R.zip` to the `{AS_ROOT}/ae_wlpserver/usr/servers/aeserver/configuration/app_64` directory.

b. If the versions of the operating systems for the Analytic Server and Hadoop nodes are different, install the R engine and Essentials for R on each Hadoop node, and into a directory with the same path and name that the R engine is installed on the Analytic Server. If you install Essentials for R on the Hadoop node, simply specify the directory in which R is installed and skip the step of specifying the location of `../ext_64/bin`.

c. Run `{AS_ROOT}/bin/hdfsUpdate.sh` to propagate the changes to HDFS.

Note: The lines that copy Fortran libraries are dependent upon the version of Fortran installed. If, for example, the 1.0 version is installed, those lines should read as follows.

```
cp -P /usr/lib64/libgfortran.so.1 ./R/lib
cp /usr/lib64/libgfortran.so.1.0.0 ./R/lib
```

Note: Install the R Engine in a location accessible to all users, as R runs on the Hadoop cluster as a different user from the Analytic Server user.

You must also install Essentials for R on the machine that hosts SPSS Modeler Server.

Enabling HCatalog data sources

In order to configure Analytic Server for use with HCatalog databases after a successful Analytic Server installation:

1. Add a `hcataloginput@remote` entry to the list of Analytic Server modules in the `{AS_ROOT}/ae_wlpserver/usr/servers/aeserver/configuration/config.properties` file. For example:

```
ae.modules=restframework@local,\
 objectstore,\
 jndidb,\
 securityprovidermanager@local,\
 componentframework@remote,\
 ...
 hcataloginput@remote
```

2. Uncomment or add the following lines to `config.properties`

```
hive.metastore.local=false
hive.metastore.uris=thrift://hostname:portnum
```

where

hostname
The name of the machine that hosts the Thrift server

portnum
The port number that is used in the HCatalog installation script

3. Make sure that the following files are available in the `{AS_ROOT}/ae_wlpserver/usr/servers/aeserver/apps/AE_BOOT.war/WEB-INF/lib` directory and also copy these files to the HDFS directory `/user/{ae_admin}/classpath` (or the class path directory specified during installation).

   The following JAR files are for HCatalog 0.4.0 and Hive 0.9.0. You must harvest the corresponding HCatalog, Hive, and dependent JAR files as appropriate for other versions.

   - The following file can be copied from the HCatalog server installation.

     `hcatalog-0.4.0.jar`

   - The following files can be copied from the Hive server installation.

```
hive-exec-0.9.0.jar
hive-metastore-0.9.0.jar
libfb303-0.7.0.jar
slf4j-api-1.6.1.jar
slf4j-log4j12-1.6.1.jar
```

- The following files can be copied from the Hadoop 1.X installation or from http://jackson.codehaus.org/.

```
jackson-core-asl-1.8.8.jar
jackson-mapper-asl-1.8.8.jar
```

The following JAR files are for HCatalog 0.5.0.

- The following files can be copied from http://code.google.com/p/guava-libraries/wiki/Release13.

```
guava-13.0.1.jar
```

The following JAR files are for Hive 0.11.0.

- The following file can be copied from the HCatalog server installation.

```
hcatalog-core-0.11.0.1.3.0.0-107.jar
```

- The following files can be copied from the Hive server installation.

```
hive-exec-0.11.0.1.3.0.0-107.jar
hive-metastore-0.11.0.1.3.0.0-107.jar
libfb303-0.9.0.jar
slf4j-api-1.6.1.jar
slf4j-log4j12-1.6.1.jar
jackson-core-asl-1.8.8.jar
jackson-mapper-asl-1.8.8.jar
guava-11.0.2.jar
```

- Copy these JAR files from the Analytic Server installation to your Hive {HIVE_HOME}/auxlib/ directory:

```
hcatalogstoragehandler-<ver>.jar
hivexmlserde-<ver>.jar
```

4. If you plan to use Hive complex types (maps, arrays, structures) copy these JAR files to your Hive {HIVE_HOME}/auxlib/ directory:

```
hcatalog-0.4.0.jar
jackson-core-asl-1.8.8.jar
jackson-mapper-asl-1.8.8.jar
```

5. The HCatalog data sources based on the compressed files require Hadoop native libraries to be available on the Analytic Server machine. Copy the Hadoop native libraries found at {HADOOP}/lib/native/Linux-amd64-64 to a directory on the Analytic Server machine and edit the **LIB_PATH** variable in {AS_ROOT}/bin/start.sh to include the path to that directory. For example:

```
export LIB_PATH=$AE_BASE/ae_wlpserver/usr/servers/aeserver/configuration/lib_32:
                $AE_BASE/ae_wlpserver/usr/servers/aeserver/configuration/lib_64:
                <hadoop_native_libraries_directory>:
```

6. To read Hive tables that use LZO compression, the following, additional steps are necessary.
   a. Copy the hadoop-lzo-*.jar files found in your Hadoop distribution to {AS_BASE}/ae_wlpserver/usr/servers/aeserver/apps/AE_BOOT.war/WEB-INF/lib/.
   b. Add the following XML fragment to {AS_BASE}/ae_wlpserver/usr/servers/aeserver/configuration/hadoop-conf/core-site.xml.

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

```
  <property>
    <name>io.compression.codec.lzo.class</name>
    <value>com.hadoop.compression.lzo.LzoCodec</value>
  </property>
</configuration>
```

If your installation does not have {AS_ROOT}/ae_wlpserver/usr/servers/aeserver/configuration/hadoop-conf/core-site.xml, copy that file from your Hadoop distribution.

7. Run {AS_ROOT}/bin/hdfsUpdate.sh.
8. Restart Analytic Server.

Note: If your Analytic Server installation already has newer versions of these JAR files, you should not copy the older versions. For example, if the Analytic Server installer has already copied jackson-core-asl-1.8.0 from a newer Hadoop distribution, you should not copy the jackson-core-asl.1.7.3 JAR file.

XML data sources

Analytic Server provides support for XML data sources through the following steps.

1. Make the Analytic Server XML serializer/deserializer available to Hive by copying the file hivexmlserde-{version}.jar from {AS_HOME}/ae_wlpserver/usr/servers/aeserver/apps/AE_BOOT.war/WEB-INF/lib to {HIVE_HOME}/auxlib.

   This step is performed once.
2. Map the XML schema to Hive data types through the Hive Data Definition Language (DDL), according to the following rules.

```
CREATE [EXTERNAL] TABLE <table_name> (<column_specifications>)
ROW FORMAT SERDE "com.ibm.spss.hive.serde2.xml.XmlSerDe"
WITH SERDEPROPERTIES (
    ["xml.processor.class"="<xml_processor_class_name>",]
    "column.xpath.<column_name>"="<xpath_query>",
     ...
    ["xml.map.specification.<element_name>"="<map_specification>"
     ...
    ]
)
STORED AS
    INPUTFORMAT "com.ibm.spss.hive.serde2.xml.XmlInputFormat"
    OUTPUTFORMAT "org.apache.hadoop.hive.ql.io.IgnoreKeyTextOutputFormat"
[LOCATION "<data_location>"]
TBLPROPERTIES (
    "xmlinput.start"="<start_tag ",
    "xmlinput.end"="<end_tag>"
);
```

   For example, the following XML...

```
<records>
  <record customer_id="0000-JTALA">
    <demographics>
      <gender>F</gender>
      <agecat>1</agecat>
      <edcat>1</edcat>
      <jobcat>2</jobcat>
      <empcat>2</empcat>
      <retire>0</retire>
      <jobsat>1</jobsat>
      <marital>1</marital>
      <spousedcat>1</spousedcat>
      <residecat>4</residecat>
      <homeown>0</homeown>
      <hometype>2</hometype>
      <addresscat>2</addresscat>
    </demographics>
```

```
    <financial>
      <income>18</income>
      <creddebt>1.003392</creddebt>
      <othdebt>2.740608</othdebt>
      <default>0</default>
    </financial>
  </record>
</records>
```

...would be represented by the following Hive DDL.

```
CREATE TABLE xml_bank(customer_id STRING, demographics map<string,string>, financial map<string,string>)
ROW FORMAT SERDE 'com.ibm.spss.hive.serde2.xml.XmlSerDe'
WITH SERDEPROPERTIES (
    "column.xpath.customer_id"="/record/@customer_id",
    "column.xpath.demographics"="/record/demographics/*",
    "column.xpath.financial"="/record/financial/*"
)
STORED AS
    INPUTFORMAT 'com.ibm.spss.hive.serde2.xml.XmlInputFormat'
    OUTPUTFORMAT 'org.apache.hadoop.hive.ql.io.IgnoreKeyTextOutputFormat'
TBLPROPERTIES (
    "xmlinput.start"="<record customer",
    "xmlinput.end"="</record>"
);
```

See "XML to Hive Data Types Mapping" for more information.

3. Create an Analytic Server data source with HCatalog content type in the Analytic Server Console.

Limitations

- Only the XPath 1.0 specification is currently supported.
- The local part of the qualified names for the elements and attributes are used when handling Hive field names. The namespace prefixes are ignored.

XML to Hive Data Types Mapping

The data modeled in XML can be transformed to the Hive data types using the conventions documented below.

Structures

The XML element can be directly mapped to the Hive structure type so that all the attributes become the data members. The content of the element becomes an additional member of primitive or complex type.

XML data

```
<result name="ID_DATUM">03.06.2009</result>
```

Hive DDL and raw data

```
struct<name:string,result:string>
{"name":"ID_DATUM", "result":"0.3.06.2009"}
```

Arrays

The XML sequences of elements can be represented as Hive arrays of primitive or complex type. The following example shows how the user can define an array of strings using content of the XML <result> element.

XML data

```
<result>03.06.2009</result>
<result>03.06.2010</result>
<result>03.06.2011</result>
```

Hive DDL and raw data

```
result array<string>
```

```
{"result":["03.06.2009","03.06.2010",...]}
```

Maps

The XML schema does not provide native support for maps. There are three common approaches to modeling maps in XML. To accommodate the different approaches we use the following syntax:

```
"xml.map.specification.<element_name>"="<key>-><value>"
```

where

element_name
The name of the XML element to be considered as a map entry

key The map entry key XML node

value The map entry value XML node

The map specification for the given XML element should be defined under the SERDEPROPERTIES section in the Hive table creation DDL. The keys and values can be defined using the following syntax:

@attribute
The @attribute specification allows the user to use the value of the attribute as a key or value of the map.

element
The element name can be used as a key or value.

#content
The content of the element can be used as a key or value. As the map keys can only be of primitive type the complex content will be converted to string.

The approaches to representing maps in XML, and their corresponding Hive DDL and raw data, is as follows.

Element name to content
The name of the element is used as a key and the content as a value. This is one of the common techniques and is used by default when mapping XML to Hive map types. The obvious limitation with this approach is that the map key can be only of type string.

XML data

```
<entry1>value1</entry1>
<entry2>value2</entry2>
<entry3>value3</entry3>
```

Mapping, Hive DDL, and raw data

In this case you do not need to specify a mapping because the name of the element is used as a key and the content as a value by default.

```
result map<string,string>
```

```
{"result":{"entry1": "value1", "entry2": "value2", "entry3": "value3"}}
```

Attribute to Element Content
Use an attribute value as a key and the element content as a value.

XML data

```
<entry name="key1">value1</entry>
<entry name="key2">value2</entry>
<entry name="key3">value3</entry>
```

Mapping, Hive DDL, and raw data

```
"xml.map.specification.entry"="@name->#content"
```

```
result map<string,string>
```

```
{"result":{"key1": "value1", "key2": "value2", "key3": "value3"}}
```

Attribute to Attribute

XML data

```
<entry name="key1" value="value1"/>
<entry name="key2" value="value2"/>
<entry name="key3" value="value3"/>
```

Mapping, Hive DDL, and raw data

```
"xml.map.specification.entry"="@name->@value"
result map<string,string>
{"result":{"key1": "value1", "key2": "value2", "key3": "value3"}}
```

Complex Content

Complex content being used as a primitive type will be converted to a valid XML string by adding a root element called <string>. Consider the following XML:

```
<dataset>
 <value>10</value>
 <value>20</value>
 <value>30</value>
</dataset>
```

The XPath expression /dataset/* will result in a number of <value> XML nodes being returned. If the target field is of primitive type the implementation will transform the result of the query to the valid XML by adding the <string> root node.

```
<string>
 <value>10</value>
 <value>20</value>
 <value>30</value>
</string>
```

Note: The implementation will not add a root element <string> If the result of the query is a single XML element.

Text Content

The whitespace-only text content of an XML element is ignored.

NoSQL data sources

NoSQL databases provide a mechanism for storage and retrieval of data that use looser consistency models than traditional relational databases in order to achieve horizontal scaling and higher availability. Analytic Server relies on the Hive storage handlers that allow access to data stored and managed by other systems in a modular, extensible fashion.

Note: Configuring NoSQL data sources for Analytic Server will typically involve manually copying some JAR files.

1. Some JAR files need to be copied from Analytic Server server to Hive. This is needed because Analytic Server implements some interfaces required by HCatalog. Analytic Server cannot use storage handlers for Hive directly in HCatalog because HCatalog has a slightly different API.
2. Some JAR files need to be copied to Analytic Server and HDFS. This is needed because some Analytic Server functionality is executed locally on the server and some functionality is executed as map-reduce jobs on Hadoop.

Apache Accumulo

Analytic Server provides support for data sources that have underlying content in Apache Accumulo. The Apache Accumulo distributed key/value store is a data storage and retrieval system. Apache Accumulo

is based on Google's BigTable design and is built on top of Apache Hadoop, Zookeeper, and Thrift. Apache Accumulo features a few novel improvements on the BigTable design in the form of cell-based access control and a server-side programming mechanism that can modify key/value pairs at various points in the data management process.

Note: Follow the steps in "Enabling HCatalog data sources" on page 8 prior to these steps.

You can set up Analytic Server for use with Accumulo through the following steps.

1. Copy the following sets of JAR files to the Hive {HIVE_HOME}/auxlib directory.

   The following files can be found in the HCatalog installation. Prior to Hive 0.11.0, this is in {HCATALOG_HOME}/share/hcatalog; starting with Hive 0.11.0, this is in {HIVE_HOME}/hcatalog/share/hcatalog.

   ```
   hcatalog-<ver>.jar
   commons-io-<ver>.jar
   ```

   The following files can be found in the Analytic Server installation.

   ```
   accumulo-hive-storage-handler-<ver>.jar
   ```

   The following files can be found in the Apache Accumulo installation.

   ```
   accumulo-trace-<ver>.jar
   accumulo-fate-<ver>.jar
   accumulo-core-<ver>.jar
   accumulo-server-<ver>.jar
   accumulo-start-<ver>.jar
   ```

2. Copy the following JAR files to the Analytic Server {AS_ROOT}/ae_wlpserver/usr/servers/aeserver/apps/AE_BOOT.war/WEB-INF/lib directory.

   The following files can be found in the HCatalog installation.

   ```
   hcatalog-<ver>.jar
   commons-io-<ver>.jar
   ```

   The following files can be found in the Apache Hive installation.

   ```
   commons-cli-<ver>.jar
   commons-collections-<ver>.jar
   commons-configuration-<ver>.jar
   commons-logging-<ver>.jar
   ```

   The following files can be found in the Apache Zookeeper installation.

   ```
   zookeeper-<ver>.jar
   ```

   The following files can be found in your Hadoop installation.

   ```
   commons-lang-<ver>.jar
   ```

   The following files can be found in the Apache Accumulo installation.

   ```
   accumulo-trace-<ver>.jar
   accumulo-fate-<ver>.jar
   accumulo-core-<ver>.jar
   accumulo-server-<ver>.jar
   accumulo-start-<ver>.jar
   guava-<ver>.jar
   ```

3. Run {AS_ROOT}/bin/hdfsUpdate.sh to propagate the changes to the HDFS.
4. Restart the Analytic Server for the changes to take effect.

To create an external Apache Accumulo table in Hive use the following syntax:

```
set accumulo.instance.id=<instance_name>;
set accumulo.user.name=<user_name>;
set accumulo.user.pass=<user_password>;
set accumulo.zookeepers=<zookeeper_host_port>;

CREATE EXTERNAL TABLE <hive_table_name>(<table_column_specifications>)
STORED BY 'com.ibm.spss.hcatalog.AccumuloStorageHandler'
WITH SERDEPROPERTIES (
```

```
'accumulo.columns.mapping' = '<family_and_qualifier_mappings>',
'accumulo.table.name' = '<Accumulo_table_name>')
TBLPROPERTIES (
    "accumulo.instance.id"="<instance_name>",
    "accumulo.zookeepers"="<zookeeper_host_port>"
);
```

For example:

```
set accumulo.instance.id=<id>;
set accumulo.user.name=admin;
set accumulo.user.pass=test;
set accumulo.zookeepers=<host>:<port>;

CREATE EXTERNAL TABLE acc_drug1n(rowid STRING,age STRING,sex STRING,bp STRING,
    cholesterol STRING,na STRING,k STRING,drug STRING)
STORED BY 'com.ibm.spss.hcatalog.AccumuloStorageHandler'
WITH SERDEPROPERTIES (
'accumulo.columns.mapping' = 'rowID,drug|age,drug|sex,drug|bp,drug|cholesterol,
    drug|na,drug|k,drug|drug',
'accumulo.table.name' = 'drug1n')
TBLPROPERTIES (
    "accumulo.instance.id"="<id>",
    "accumulo.zookeepers"="<host>:<port>"
);
```

Note: The Accumulo user name and password for the given Accumulo table should match the user name and password of the authenticated Analytic Server user.

Apache HBase

Analytic Server provides support for data sources that have underlying content in Apache HBase. Apache HBase is an open-source, distributed, versioned, column-oriented store on top of Hadoop and HDFS.

Note: Follow the steps in "Enabling HCatalog data sources" on page 8 prior to these steps.

You can set up Analytic Server for use with Apache HBase through the following steps.

1. Copy the following JAR files to the Hive `{HIVE_HOME}/auxlib` directory.

   The following file can be found in the HCatalog installation. Prior to Hive 0.11.0, this is in `{HCATALOG_HOME}/share/hcatalog`; starting with Hive 0.11.0, this is in `{HIVE_HOME}/hcatalog/share/hcatalog`.

   ```
   hcatalog-<ver>.jar
   ```

2. Copy the following JAR files from the Hive `../lib` and `../auxlib` directories, and the HBase `../lib` directory to the Analytic Server `{AS_ROOT}/ae_wlpserver/usr/servers/aeserver/apps/AE_BOOT.war/WEB-INF/lib` directory.

   ```
   hcatalog-<ver>.jar
   zookeeper-<ver>.jar
   jersey-json-<ver>.jar
   protobuf-java-<ver>.jar
   hive-hbase-handler-<ver>.jar
   hbase-<ver>.jar
   guava-<ver>.jar
   ```

3. Run `{AS_ROOT}/bin/hdfsUpdate.sh` to propagate the changes to the HDFS.
4. Create an `hbase-conf` directory under the Analytic Server `configuration` directory and add `hbase-site.xml` from your HBase installation.
5. Restart the Analytic Server for the changes to take an effect.

To create an external HBase table in Hive use the following syntax:

```
CREATE EXTERNAL TABLE <tablename>(<table_column_specifications>)
STORED BY 'com.ibm.spss.hcatalog.HBaseStorageHandler'
WITH SERDEPROPERTIES ("hbase.columns.mapping" = "<column_mapping_spec>")
TBLPROPERTIES("hbase.table.name" = "<hbase_table_name>")
```

For example:

```
CREATE EXTERNAL TABLE hbase_drug1n(rowid STRING,age STRING,sex STRING,bp STRING,
    cholesterol STRING,na STRING,k STRING,drug STRING)
STORED BY 'com.ibm.spss.hcatalog.HBaseStorageHandler'
WITH SERDEPROPERTIES ("hbase.columns.mapping" = ":key,drug:age,drug:sex,drug:bp,
    drug:cholesterol,drug:na,drug:k,drug:drug")
TBLPROPERTIES("hbase.table.name" = "drug1n");
```

Note: For information on how to create an HBase table, see the Apache HBase Reference Guide (http://hbase.apache.org/book.html).

Note: It is a good practice to preface the database name to indicate the type of database. For example, name a database HB_drug1n to indicate an HBase database, or ACC_drug1n to indicate an Accumulo database. This will help with the selection of the HCatalog file when in the Analytic Server console.

Getting users started

Tell users to navigate to `http://<host>:<port>/admin/<tenant>`and enter their username and password to log on to the Analytic Server console.

<host>
The address of the Analytic Server host

<port>
The port that Analytic Server is listening on

<tenant>
In a multi-tenant environment, the tenant you belong to. In a single-tenant environment, the default tenant is **ibm**.

In order to access IBM SPSS Analytic Catalyst, navigate to `http://<host>:<port>/catalyst.html` and enter their username and password to log on.

Performance Tuning

Increasing server workload

In order to increase the number of Analytic Server worker processes, you must change the following parameters:

ae.pool.size
This is the number of workers. Its default value is 2. It is found in the `{AS_ROOT}/ae_wlpserver/usr/servers/aeserver/configuration/config.properties` file.

jndi.aedb.pool
This is the number of connections per worker. Its default value is 10. It is found in the `{AS_ROOT}/ae_wlpserver/usr/servers/aeserver/configuration/config.properties` file.

derby.drda.maxThreads
This is the maximum number of connections that are accepted by the Analytic Server database. It should be set to no less than **ae.pool.size*jndi.aedb.pool** + **jndi.aedb.pool**.

Problem determination

Analytic Server provides several helpful tools for problem determination.

Logging

Analytic Server creates customer log files and trace files that are helpful for diagnosing problems. With the default Liberty installation, you can find the log files in the {AS_ROOT}/ae_wlpserver/usr/servers/aeserver/logs directory. There is a separate subdirectory for each Analytic Server process.

For each process, the default logging configuration produces two log files that roll over on a daily basis.

ae.log
: This file contains the high-level summary of informational warning and error messages. Check this file first when server errors occur that cannot be resolved by using the error message that is displayed in the User interface.

ae_trace.log
: This file contains all the entries from ae.log, but adds more information that is primarily targeted to IBM support and development for debugging purposes.

Analytic Server uses Apache LOG4J as its underlying logging facility. Using LOG4J, the logging can be dynamically adjusted by editing the {AS_SERVER_ROOT}/configuration/log4j.xml configuration file. You may be asked to do this by Support to help diagnose problems, or you may want to modify this to limit the number of log files kept around. Changes to the file are detected automatically within a few seconds so the Analytic Server does not need restarted.

For more information about log4j and the configuration file, see documentation at the official Apache website at http://logging.apache.org/log4j/.

Version information

You can determine what version of Analytic Server is installed by checking the {AS_ROOT}/properties/version folder. The following files contain version information.

IBM_SPSS_Analytic_Server-*.swtag
: Contains detailed product information.

version.txt
: Version and build number for the installed product.

Log collector

When problems cannot be resolved by directly reviewing the log files, you can bundle all the logs and send them to IBM support. There is a utility that is provided to make collecting all the necessary data simpler.

Using a command shell, run the following commands

```
cd  <AS_ROOT>/tools/support/logcollector
run  >sh ./logcollector.sh
```

These commands create a compressed file under <AS_ROOT>/tools/support/logcollector. The compressed file contains all the log files and product version information.

Chapter 3. Tenant management

Tenants provide a high-level division of users, projects, and data sources. Each user accesses the system in the context of a tenant to which they are assigned. You manage tenants, and assign to tenants, in the Analytic Server console.

The view of the Tenants accordion depends upon the role of the user that is logged in to the console:

- The "super user" administrator that is set up during installation is the tenant manager. Only this user can create new tenants and edit the properties of any tenant.
- Users with the Administrator role can edit the properties of the tenant they are logged in to.
- Users with the User role cannot edit tenant properties. The Tenants accordion is hidden from them.

Administrators can access the Projects and Data sources accordions and manage any project or data source for cleanup and administration. See the *IBM SPSS Analytic Server User's Guide* for more information.

Left column

The left column displays the existing tenants under the accordion heading. Only the "super user" administrator can use these controls.

- Select a tenant to display its details in the content area and edit its properties. Typing in the search area filters the listing to display only tenants with the search string in their name.
- Click **New tenant** to create a new tenant with the name you specify in the **Add New Tenant** dialog. Names are case-sensitive, ignore leading and trailing white space, and protect against SQL injection.
- Click **Delete tenant** to remove the tenant.

Content area

The content area is divided into *Details*, *Principals* and *Projects* collapsible sections.

Details

Name An editable text field that displays the name of the tenant. Tenant names must be case-sensitive, ignore leading and trailing white space, and protect against SQL injection.

Description
An editable text field that allows you to provide explanatory text about the tenant.

URL This is the URL to give to users to log in to the tenant through the Analytic Server console, and to use to configure SPSS Modeler server. See "Configuring IBM SPSS Modeler for use with IBM SPSS Analytic Server" on page 3 for details on configuring SPSS Modeler.

Principals

Principals are users and groups that are drawn from the security provider that is set up during installation. You can add principals to a tenant as Administrators or Users.

- Typing in the text box filters on users and groups with the search string in their name. Select **Administrator** or **User** from the drop-down list to assign their role within the tenant. Click **Add participant** to add them to the list of authors.
- To remove a participant, select a user or group in the member list and click **Remove participant**.

Projects

Projects are versioned based on changes to the file and folder contents. This table lists all of the projects in the tenant, and allows an administrator to specify the maximum number of versions per tenant. Analytic Server automatically deletes the oldest committed project version when the number of versions exceeds the specified number. The values here are the same as those values on the **Automatically clean up when number of versions exceeds** text box on the Versions tab of the Projects accordion, but presented in a summary view.

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 wish 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.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

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.

Printed in USA