

**IBM SPSS Analytic Server**  
バージョン 3.0.1

インストールおよび構成ガイド

**IBM**

注記

本書および本書で紹介する製品をご使用になる前に、59 ページの『特記事項』に記載されている情報をお読みください。

本書は、IBM SPSS Analytic Server バージョン 3、リリース 0、モディフィケーション 1、および新しい版で明記されていない限り、以降のすべてのリリースおよびモディフィケーションに適用されます。

お客様の環境によっては、資料中の円記号がバックスラッシュと表示されたり、バックスラッシュが円記号と表示されたりする場合があります。

原典： IBM SPSS Analytic Server  
Version 3.0.1  
Installation and Configuration Guide

発行： 日本アイ・ビー・エム株式会社

担当： トランスレーション・サービス・センター

# 目次

|   |    |  |    |
|---|----|--|----|
| 第 1 章 前提条件 . . . . .                                      | 1  | Analytic Server で使用するポートの変更 . . . . .                                  | 38 |
| 第 2 章 Ambari のインストールおよび構成 . . . . .                       | 3  | 高可用性 Analytic Server . . . . .   | 38 |
| Ambari 固有の前提条件 . . . . .                                  | 3  | 小規模データ向けの JVM オプションの最適化 . . . . .                                      | 38 |
| Ambari でのインストール . . . . .                                 | 3  | マイグレーション . . . . .   | 38 |
| オフライン・インストール . . . . .                                    | 5  | Cloudera での Analytic Server のアンインストール . . . . .                        | 40 |
| 外部で管理されている MySQL 環境に対する Analytic Server のインストール . . . . . | 6  | 第 4 章 MapR のインストールおよび構成 . . . . .                                      | 41 |
| 構成 . . . . .  | 7  | MapR の概要 . . . . .   | 41 |
| セキュリティ . . . . .  | 7  | MapR での Analytic Server のインストール . . . . .                              | 41 |
| Essentials for R に対するサポートの有効化 . . . . .                   | 12 | MapR の構成 . . . . .   | 44 |
| リレーショナル・データベース・ソースの有効化 . . . . .                          | 14 | データベース・プッシュバックの有効化 . . . . .   | 44 |
| HCatalog データ・ソースの有効化 . . . . .                            | 14 | Apache Hive の有効化 . . . . .   | 45 |
| Analytic Server で使用するポートの変更 . . . . .                     | 15 | Apache HBase の有効化 . . . . .  | 46 |
| 高可用性 Analytic Server . . . . .                            | 16 | Apache Spark の有効化 . . . . .  | 47 |
| 小規模データ向けの JVM オプションの最適化 . . . . .                         | 16 | 機能フラグの有効化 . . . . .  | 47 |
| クライアント依存関係の更新 . . . . .                                   | 16 | R の有効化 . . . . .   | 47 |
| Apache Knox の構成 . . . . .                                 | 16 | LZO の有効化 . . . . .   | 48 |
| アップグレードおよびマイグレーション . . . . .                              | 21 | MapR 用の IBM SPSS Analytic Server クラスタ<br>のセットアップ . . . . .             | 49 |
| アンインストール . . . . .  | 24 | MapR のアンインストール . . . . .   | 49 |
| Essentials for R のアンインストール . . . . .                      | 25 | MapR での IBM SPSS Analytic Server のマイグレーション . . . . .                   | 49 |
| 第 3 章 Cloudera のインストールおよび構成 . . . . .                     | 27 | MapR のトラブルシューティング . . . . .  | 51 |
| Cloudera の概要 . . . . .                                    | 27 | 第 5 章 IBM SPSS Analytic Server で使用するための IBM SPSS Modeler の構成 . . . . . | 53 |
| Cloudera 固有の前提条件 . . . . .                                | 27 | Analytic Server 用の MySQL の構成 . . . . .                                 | 27 |
| Cloudera でのインストール . . . . .                               | 28 | Cloudera のインストール . . . . .   | 28 |
| Cloudera の構成 . . . . .                                    | 30 | Cloudera の構成 . . . . .   | 30 |
| セキュリティ . . . . .  | 30 | セキュリティ . . . . .   | 30 |
| Essentials for R に対するサポートの有効化 . . . . .                   | 35 | Essentials for R に対するサポートの有効化 . . . . .                                | 35 |
| リレーショナル・データベース・ソースの有効化 . . . . .                          | 36 | リレーショナル・データベース・ソースの有効化 . . . . .                                       | 36 |
| HCatalog データ・ソースの有効化 . . . . .                            | 37 | HCatalog データ・ソースの有効化 . . . . .   | 37 |
| 第 6 章 トラブルシューティング . . . . .                               | 55 | 特記事項 . . . . .   | 59 |
| 特記事項 . . . . .  | 59 | 商標 . . . . .   | 60 |
| 商標 . . . . .  | 60 |  |    |



---

## 第 1 章 前提条件

Analytic Server をインストールする前に、以下の情報を確認してください。

### システム要件

最新のシステム要件情報については、IBM Technical Support サイトの <http://publib.boulder.ibm.com/infocenter/prodguid/v1r0/clarity/softwareReqsForProduct.html> にある「Detailed system requirements」レポートを使用してください。このページでは以下を行うことができます。

1. 製品名として SPSS Analytic Server を入力して、「**Search**」をクリックします。
2. 目的のバージョンとレポートの範囲を選択して、「**Submit**」をクリックします。

### Power Systems

クラスター内のすべてのホストに IBM XLC コンパイラーおよび XLF コンパイラーがインストールされており、PATH に含まれていることを確認してください。

これらのコンパイラー用のライセンスの取得の詳細については、以下の Web サイトで確認できます。

- XL C for Linux: <http://www-03.ibm.com/software/products/en/xlcpp-linux>
- XL Fortran for Linux: <http://www-03.ibm.com/software/products/en/xlfortran-linux>

### Hive/HCatalog

NoSQL データ・ソースを使用する予定の場合、Hive および HCatalog をリモート・アクセス用に構成します。さらに、hive-site.xml に、アクティブな Thrift Hive Metastore サーバーを示す *hive.metastore.uris* プロパティが `thrift://<host_name>:<port>` の形式で設定されていることを確認します。詳しくは、使用している Hadoop ディストリビューションの資料を参照してください。

### メタデータ・リポジトリ

デフォルトでは、Analytic Server は MySQL データベースをインストールして使用します。または、既存の DB2 インストール済み環境を使用するように Analytic Server を構成することもできます。選択するデータベースのタイプにかかわらず、データベースには UTF-8 のエンコードが必要です。

### MySQL

MySQL のデフォルト文字セットはバージョンとオペレーティング・システムによって異なります。ご使用の MySQL インストール済み環境が UTF-8 に設定されているかどうかを確認するには、以下の手順を使用してください。

1. MySQL のバージョンを確認します。

```
mysql -V
```
2. MySQL のコマンド・ライン・インターフェースから以下の照会を実行して、MySQL のデフォルト文字セットを確認します。

```
mysql>show variables like 'char%';
```

文字セットが既に UTF-8 に設定されている場合、追加の変更は不要です。
3. MySQL のコマンド・ライン・インターフェースから以下の照会を実行して、MySQL のデフォルト照合を確認します。

```
mysql>show variables like 'coll%';
```

照合が既に UTF-8 に設定されている場合、追加の変更は不要です。

4. デフォルトの文字セットまたは照合が UTF-8 でない場合、文字セットを UTF-8 に変更するために `/etc/my.cnf` を編集して MySQL デーモンを再始動する方法について、MySQL の資料を参照してください。

**DB2** DB2 の構成について詳しくは、Knowledge Center ([http://www-01.ibm.com/support/knowledgecenter/SSEPGG\\_10.5.0/com.ibm.db2.luw.kc.doc/welcome.html](http://www-01.ibm.com/support/knowledgecenter/SSEPGG_10.5.0/com.ibm.db2.luw.kc.doc/welcome.html)) を参照してください。

## 高可用性クラスター

### ロード・バランサー

高可用性クラスターには、セッション・アフィニティー (スティッキー・セッションと呼ばれることもあります) をサポートするロード・バランサーが必要です。Analytic Server は、Cookie 「request-token」 でセッションを識別します。これにより、アプリケーションによって制御されるセッション・アフィニティーで使用するために、ユーザー・ログインの期間にわたってセッションが識別されます。セッション・アフィニティーがどのようにサポートされるかについて詳しくは、ご使用の特定のロード・バランサーの資料を参照してください。

---

## 第 2 章 Ambari のインストールおよび構成

---

### Ambari 固有の前提条件

一般的な前提条件に加えて、以下の情報を確認してください。

#### サービス

Analytic Server は Ambari サービスとしてインストールされます。Analytic Server をインストールする前に、HDFS、YARN、MapReduce2、Hive、および Zookeeper が Ambari サービスとして追加されていることを確認する必要があります。

#### パスワード無し SSH

root ユーザー用に Analytic Metastore ホストとクラスター内のすべてのホストの間にパスワード無し SSH をセットアップしてください。

---

### Ambari でのインストール

基本的なプロセスは、次のとおりです。Analytic Server ファイルを Ambari クラスター内のホストにインストールし、続いて Analytic Server を Ambari サービスとして追加します。さらに詳細なステップは、以下のとおりです。

1. IBM パスポート・アドバンテージ Web サイトに移動し、ご使用のスタック、スタック・バージョン、およびハードウェア・アーキテクチャーに固有の自己解凍型バイナリー・ファイルを Ambari クラスター内のホストにダウンロードします。
2. 自己解凍型バイナリー・ファイルを実行し、指示に従って (オプションで) ライセンスを表示し、ライセンスを受け入れて、オンライン・インストールまたはオフライン・インストールを選択します。

#### オンライン・インストール

Ambari サーバー・ホストおよびクラスター内のすべてのノードが <http://ibm-open-platform.ibm.com> にアクセス可能な場合は、オンライン・インストールを選択してください。

[GPFS (Spectrum Scale) のみ] ファイル [http://ibm-open-platform.ibm.com/repos/IBM-SPSS-AnalyticServer/3.0.1.0/x86\\_64/IBM-SPSS-AnalyticServer-3.0.1.0.repo](http://ibm-open-platform.ibm.com/repos/IBM-SPSS-AnalyticServer/3.0.1.0/x86_64/IBM-SPSS-AnalyticServer-3.0.1.0.repo) (x86) または <https://ibm-open-platform.ibm.com/repos/IBM-SPSS-AnalyticServer/3.0.1.0/ppc64le/IBM-SPSS-AnalyticServer-3.0.1.0.repo> (ppc64le) をダウンロードし、それを、サービスとして Analytic Server Metastore を追加した先の各ノード上のフォルダー `/etc/yum.repos.d` (RHEL、CentOS) または `/etc/zypp/repos.d` (SLES) に移動します。

#### オフライン・インストール

ご使用の Ambari サーバー・ホストがインターネットにアクセスできない場合は、オフラインを選択します。詳しくは、5 ページの『オフライン・インストール』を参照してください。

3. Ambari サーバーを再起動します。

```
ambari-server restart
```

4. Ambari サーバーにログオンし、Ambari UI を使用して Analytic Server をサービスとしてインストールします。

#### メタデータ・リポジトリ

デフォルトでは、Analytic Server は MySQL を使用して、データ・ソース、プロジェクト、およびテナントに関する情報を追跡します。インストール時に、Analytic Server と MySQL

の間の JDBC 接続で使用されるユーザー名 (**metadata.repository.user.name**) およびパスワード (**metadata.repository.password**) を指定する必要があります。インストーラーは MySQL データベースにそのユーザーを作成しますが、そのユーザーは MySQL データベースに固有であり、既存の Linux ユーザーや Hadoop ユーザーである必要はありません。

メタデータ・リポジトリを DB2 に変更するには、以下のステップを実行します。

注: インストールの完了後にメタデータのリポジトリを変更することはできません。

- a. 別のマシンに DB2 がインストールされていることを確認します。詳しくは、トピック 1 ページの『第 1 章 前提条件』の『メタデータ・リポジトリ』セクションを参照してください。
- b. Ambari の「Services」タブで、Analytic Server サービスの「Configs」タブに移動します。
- c. 「**Advanced analytics-env**」セクションを開きます。
- d. **as.database.type** の値を `mysql` から `db2` に変更します。
- e. 「**Advanced analytics-meta**」セクションを開きます。
- f. **metadata.repository.driver** の値を `com.mysql.jdbc.Driver` から `com.ibm.db2.jcc.DB2Driver` に変更します。
- g. **metadata.repository.url** の値を `jdbc:db2://{DB2_HOST}:{PORT}/{DBName}:currentSchema={SchemaName};` に変更します。ここで、
  - {DB2\_HOST} は、DB2 がインストールされているサーバーのホスト名です。
  - {PORT} は、DB2 が listen しているポートです。
  - {SchemaName} は、使用可能な、未使用のスキーマです。入力する値がわからない場合は、DB2 管理者に協力を求めてください。
- h. **metadata.repository.user.name** および **metadata.repository.password** に、有効な DB2 資格情報を入力します。
- i. 「**Save**」をクリックします。

インストール後に変更してはならない構成設定

インストール後に以下の設定は変更しないでください。変更すると Analytic Server が動作しなくなります。

- Analytic\_Server\_User
- Analytic\_Server\_UserID
- as.database.type
- metadata.repository.driver
- distrib.fs.root

5. これで、Analytic Server のインスタンスが機能するようになりました。追加の構成はオプションです。Analytic Server の構成と管理について詳しくは、7 ページの『構成』のトピックを参照してください。既存構成の新規インストール済み環境へのマイグレーションについては、21 ページの『アップグレードおよびマイグレーション』のトピックを参照してください。
6. Web ブラウザーを開き、アドレス `http://<host>:<port>/analyticserver/admin/ibm` を入力します。ここで、<host> は Analytic Server ホストのアドレスであり、<port> は Analytic Server が listen しているポートです。デフォルトではこれは 9080 です。この URL にアクセスすると、Analytic Server コンソールのログイン・ダイアログが開きます。Analytic Server 管理者としてログインします。デフォルトでは、このユーザー ID は `admin` であり、パスワードは `admin` です。

## オフライン・インストール

オフライン・インストールでは必要な RPM ファイルをダウンロードするため、<https://ibm-open-platform.ibm.com/repos/IBM-SPSS-AnalyticServer/3.0.1.0/> にアクセス可能なマシンで実行する必要があります。実行可能なバイナリー・ファイルは、使用可能な Ambari <AS\_INSTALLABLE\_HOME> ディストリビューション・ディレクトリーにあります。該当する <AS\_INSTALLABLE\_HOME> ディレクトリーの内容全体を Ambari サーバー・ホストにコピーします。

1. ローカル Yum リポジトリーを作成可能なツールをインストールします。

```
yum install createrepo
```

2. Analytic Server の RPM ファイルのリポジトリーとして機能する新規ディレクトリーを作成します。以下の例を参照してください。

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

3. このディレクトリーに、必要な Analytic Server の RPM ファイルをコピーします。必要な RPM ファイルは、ご使用のディストリビューション、バージョン、およびアーキテクチャーによって以下のように異なります。

### BigInsights 4.2 (x86\_64)

```
IBM-SPSS-AnalyticServer-ambari-2.1-BI-4.2-3.0.1.0-1.x86_64.rpm
```

```
IBM-SPSS-AnalyticServer-3.0.1.0-1.x86_64.rpm
```

### BigInsights 4.2 (PPC64LE)

```
IBM-SPSS-AnalyticServer-ambari-2.1-BI-4.2-3.0.1.0-1.ppc64le.rpm
```

```
IBM-SPSS-AnalyticServer-3.0.1.0-1.ppc64le.rpm
```

### HDP 2.4 (x86\_64)

```
IBM-SPSS-AnalyticServer-3.0.1.0-1.x86_64.rpm
```

```
IBM-SPSS-AnalyticServer-ambari-2.1-HDP-2.4-3.0.1.0-1.x86_64.rpm
```

4. ローカル・リポジトリーの定義を作成します。例えば、IBM-SPSS-AnalyticServer-3.0.1.0.repo というファイルを、/etc/yum.repos.d/ (RHEL、CentOS の場合) または /etc/zypp/repos.d/ (SLES の場合) に、以下の内容を指定して作成します。

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

5. ローカル Yum リポジトリーを作成します。以下の例を参照してください。

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

6. root ユーザーのコマンド・ウィンドウから、<AS\_INSTALLABLE\_HOME>/IBM-SPSS-AnalyticServer ディレクトリーへの cd を実行し、run ./offLineInstall.sh を実行します。スクリプトは、以前に実行されたバイナリー実行可能インストール・コマンドへの持続応答を読み取り、(rpm をインストールするための) 適切なプラットフォーム・コマンドを発行します。
7. ご使用の Ambari リポジトリー・ファイル repoinfo.xml (通常は /var/lib/ambari-server/resources/stacks/\$stackName/\$stackVersion/repos/ に配置されています) に以下の行を追加して、ローカル Yum リポジトリーを使用するように更新します。

```
<os type="host_os">
  <repo>
    <baseurl>file:///path to local repository/</baseurl>
```

```
<repid>IBM-SPSS-AnalyticServer</repid>
<reponame>IBM-SPSS-AnalyticServer-3.0.1.0</reponame>
</repo>
</os>
```

## 外部で管理されている MySQL 環境に対する Analytic Server のインストール

外部で管理されている MySQL 環境に対して Analytic Server をインストールする場合、そのプロセスは通常のインストールとは異なります。

以下のステップでは、外部で管理されている MySQL 環境に対して Analytic Server をインストールするプロセスについて説明します。

1. IBM パスポート・アドバンテージ Web サイトに移動し、ご使用のスタック、スタック・バージョン、およびハードウェア・アーキテクチャーに固有の自己解凍型バイナリー・ファイルを Ambari クラスタ内のホストにダウンロードします。
2. 自己解凍型バイナリー・ファイルを実行し、指示に従って (オプションで) ライセンスを表示し、ライセンスを受け入れます。
  - a. オンライン・オプションを選択します。
  - b. プロンプトが出されたら、「外部の MySQL データベース (External MySQL Database)」オプションを選択します。
3. `add_mysql_user.sh` スクリプトを、`/opt/AS_Installable/IBM-SPSS-AnalyticServer` から、MySQL インスタンス (AS\_MetaStore として使用される) がインストールされているノード/ホストにコピーします。例えば、`/opt/AS_InstallTools` です。
  - MySQL ノード/ホストで `add_mysql_user.sh` スクリプトを実行します。例えば、`./add_mysql_user.sh -u as_user -p spss -d aedb` です。

### Notes:

- ユーザー名およびパスワードは、Ambari 構成画面で AS\_Metastore に入力したデータベース・ユーザー名およびパスワードと一致している必要があります。
  - `add_mysql_user.sh` スクリプトは、コマンドを発行するように手動更新できます (必要な場合)。
  - 保護された (root ユーザー・アクセス) MySQL データベースに対して `add_mysql_user.sh` スクリプトを実行する場合は、`dbuserid` と `dbuserid_password` を渡すために `-r` パラメーターと `-t` パラメーターを使用します。スクリプトは `dbuserid` と `dbuserid_password` を使用して MySQL 操作を実行します。
4. Ambari サーバーを再起動します。
  5. Ambari コンソールから、通常どおり AnalyticServer サービスを追加します (ステップ 3 で入力したものと同一データベース・ユーザー名およびパスワードを入力します)。

注: 「AS\_Configuration」画面の `metadata.repository.url` 設定 (「Advanced analytics-meta」) は、MySQL データベース・ホストを指すように変更する必要があります。例えば、JDBC 設定 `mysql://{analytic_metastore_host}/aedb?createDatabaseIfNotExist=true` を `mysql://{MySQL_DB}/aedb?createDatabaseIfNotExist=true` に変更します。

---

## 構成

インストール後、オプションで Ambari UI を使用して Analytic Server を構成し、管理することができます。

注: Analytic Server ファイル・パスには以下の規則が使用されます。

- {AS\_ROOT} は、Analytic Server がデプロイされている場所を示します (例えば、/opt/IBM/SPSS/AnalyticServer/{version})。
- {AS\_SERVER\_ROOT} は、構成ファイル、ログ・ファイル、およびサーバー・ファイルのロケーションを示します (例えば、/opt/IBM/SPSS/AnalyticServer/{version}/ae\_wlpserver/usr/servers/aeserver)。
- {AS\_HOME} は、Analytic Server がルート・フォルダーとして使用する HDFS 上の場所を示します。

## セキュリティ

**security.config** パラメーターは、Analytic Server システムにプリンシパルとして追加できるユーザーとグループのレジストリーを定義します。

デフォルトでは、基本レジストリーは、単一のユーザー `admin` およびパスワード `admin` を指定して定義されています。**security.config** を編集するか、または Kerberos を構成することにより、このレジストリーを変更できます。**security.config** パラメーターは、Analytic Server サービスの「Configs」タブの「**Advanced analytics.cfg**」セクションにあります。

注: **security.config** パラメーターを編集してレジストリーを変更する場合、新規の各ユーザーをプリンシパルとして Analytic Server システムに追加する必要があります。テナント管理について詳しくは、「*IBM SPSS Analytic Server 管理者ガイド*」を参照してください。

## 基本レジストリーの変更

基本レジストリーを使用して、**security.config** パラメーター内にユーザーとグループのデータベースを定義できます。

デフォルトの基本レジストリーは以下のようになっています。

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

変更後の基本レジストリーの例を以下に示します。

```
<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>

```

securityUtility ツール ({AS\_ROOT}/ae\_wlpserver/bin にあります) を使用してパスワードをエンコードすることで、パスワードの値を難読化できます。

```

securityUtility encode changeit
    {xor}Pdc+MTg6Nis=

```

注: securityUtility ツールについて詳しくは、[http://www-01.ibm.com/support/knowledgecenter/SSD28V\\_8.5.5/com.ibm.websphere.wlp.core.doc/ae/rwlp\\_command\\_securityutil.html](http://www-01.ibm.com/support/knowledgecenter/SSD28V_8.5.5/com.ibm.websphere.wlp.core.doc/ae/rwlp_command_securityutil.html) を参照してください。

注: 基本レジストリーは、サンドボックス環境では有用ですが、実稼働環境ではお勧めしません。

## LDAP レジストリーの構成

LDAP レジストリーは、Active Directory や OpenLDAP などの外部 LDAP サーバーを使用してユーザーを認証できるようにします。

重要: LDAP ユーザーは、Ambari で Analytic Server 管理者として指定されている必要があります。

以下に、OpenLDAP の ldapRegistry の例を示します。

```

<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="(&(uid=%v)(objectClass=inetOrgPerson))"
    groupFilter="(&(cn=%v)(|(objectclass=organizationalUnit)))"
    groupMemberIdMap="posixGroup:memberUid"/>
</ldapRegistry>

```

以下の例は、Active Directory を使用した Analytic Server 認証を示しています。

```

<ldapRegistry id="Microsoft Active Directory" realm="ibm"
  host="host"
  port="389"
  baseDN="cn=users,dc=adtest,dc=mycompany,dc=com"
  bindDN="cn=administrator,cn=users,dc=adtest,dc=mycompany,dc=com"
  bindPassword="adminpassword"
  ldapType="Custom"
  <customFilters
    userFilter="(&(sAMAccountName=%v)(objectcategory=user))"
    groupFilter="(&(cn=%v)(objectcategory=group))"
    userIdMap="user:sAMAccountName"
    groupIdMap="*:cn"
    groupMemberIdMap="memberOf:member" />
</ldapRegistry>

```

注: LDAP 構成を確認するには、LDAP ビューアーのサード・パーティー・ツールを使用すると便利なおとがよくあります。

以下の例は、Active Directory を使用した WebSphere Liberty プロファイル認証を示しています。

```
<ldapRegistry id="ldap" realm="SampleLdapADRealm"
  host="ldapserverserver.mycity.mycompany.com" port="389" ignoreCase="true"
  baseDN="cn=users,dc=adtest,dc=mycity,dc=mycompany,dc=com"
  bindDN="cn=testuser,cn=users,dc=adtest,dc=mycity,dc=mycompany,dc=com"
  bindPassword="testuserpwd"
  ldapType="Microsoft Active Directory"
  sslEnabled="true"
  sslRef="LDAPSSLSettings">
  <activatedFilters
    userFilter="(&!(sAMAccountName=%v)(objectcategory=user))"
    groupFilter="(&!(cn=%v)(objectcategory=group))"
    userIdMap="user:sAMAccountName"
    groupIdMap="*:cn"
    groupMemberIdMap="memberOf:member" >
  </activatedFilters>
</ldapRegistry>

<ssl id="LDAPSSLSettings" keyStoreRef="LDAPKeyStore" trustStoreRef="LDAPTrustStore" />

<keyStore id="LDAPKeyStore" location="${server.config.dir}/LdapSSLKeyStore.jks"
  type="JKS" password="{xor}CDo9Hgw=" />

<keyStore id="LDAPTrustStore" location="${server.config.dir}/LdapSSLTrustStore.jks"
  type="JKS" password="{xor}CDo9Hgw=" />
```

#### Notes:

- Analytic Server での LDAP のサポートは WebSphere Liberty によって制御されます。詳しくは、[Configuring LDAP user registries in Liberty](#) を参照してください。
- SSL で LDAP を保護する場合は、以下の『Analytic Server から LDAP への Secure Sockets Layer (SSL) 接続の構成』セクションの指示に従ってください。

### Analytic Server から LDAP への Secure Sockets Layer (SSL) 接続の構成

1. Analytic Server マシンのそれぞれに Analytic Server ユーザーとしてログインし、SSL 証明書の共通ディレクトリーを作成します。

注: デフォルトでは、Analytic Server ユーザーは `as_user` です。Ambari コンソールの「Admin」タブの下の「**Service accounts**」を参照してください。

2. 鍵ストア・ファイルおよびトラストストア・ファイルを、すべての Analytic Server マシンの共通ディレクトリーにコピーします。また、LDAP クライアントの CA 証明書をトラストストアに追加します。以下に、手順例を示します。

```
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
```

注: JAVA\_HOME は、Analytic Server の起動に使用するのと同じ JRE です。

3. `securityUtility` ツール (`{AS_ROOT}/ae_wlpserver/bin` にあります) を使用してパスワードをエンコードすることで、パスワードの値を難読化できます。次に例を示します。

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

4. Ambari コンソールにログインし、Analytic Server の構成設定 `ssl.keystore.config` を、正しい SSL 構成設定に更新します。次に例を示します。

```
<ssl id="defaultSSLConfig" keyStoreRef="defaultKeyStore" trustStoreRef="defaultTrustStore"
  clientAuthenticationSupported="true"/>
  <keyStore id="defaultKeyStore" location="/home/as_user/security/mykey.jks" type="JKS"
    password="{xor}0zo5PiozKxYdEgwPDaWeDG1uDz4sLCg7"/>
  <keyStore id="defaultTrustStore" location="/home/as_user/security/mytrust.jks" type="JKS"
    password="{xor}PDC+MTg6NiS="/>
```

注: 鍵ファイルおよびトラストストア・ファイルについては、絶対パスを使用してください。

5. Analytic Server の構成設定 **security.config** を、正しい LDAP 構成設定に更新します。例えば、**ldapRegistry** 要素の場合、**sslEnabled** 属性を true に設定し、**sslRef** 属性を defaultSSLConfig に設定します。

## Kerberos の構成

Analytic Server は、Ambari を使用した Kerberos をサポートします。

注: IBM® SPSS® Analytic Server が Apache Knox とともに使用された場合、Kerberos シングル・サインオン (SSO) はサポートされません。

1. Analytic Server へのアクセス権を付与する予定のすべてのユーザーについて、Kerberos ユーザー・リポジトリ内にアカウントを作成します。

注: Analytic Server インストール済み環境で基本レジストリーを使用する場合、このレジストリーには、すべての Kerberos ユーザー・アカウントが、パスワードとして "-" を使用して、指定されていなければなりません。次に例を示します。

```
<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. 前のステップで、Analytic Server および Hadoop の各ノードで作成したそれぞれのユーザーについて、OS ユーザー・アカウントを作成します。
  - これらのユーザーの UID は、すべてのマシンで一致させてください。kinit コマンドを使用して各アカウントにログオンして、これをテストすることができます。
  - UID が、Yarn の「ジョブをサブミットするための最小ユーザー ID (Minimum user ID for submitting job)」設定に従っていることを確認してください。これは、container-executor.cfg 内の **min.user.id** パラメーターです。例えば、**min.user.id** が 1000 の場合、作成される各ユーザー・アカウントの UID は 1000 以上でなければなりません。
3. Analytic Server のすべてのプリンシパルについて、HDFS 上にユーザーのホーム・フォルダーを作成します。例えば、Analytic Server システムに testuser1 を追加した場合、HDFS 上に /user/testuser1 のようなホーム・フォルダーを作成し、testuser1 がこのフォルダーに対する読み取り権限と書き込み権限を持つようにします。
4. [オプション] HCatalog データ・ソースを使用する予定であり、Analytic Server が Hive Metastore とは別のマシンにインストールされている場合、HDFS で Hive クライアント名を使用する必要があります。

- a. Ambari コンソールで、HDFS サービスの「Configs」タブに移動します。
- b. **hadoop.proxyuser.hive.groups** パラメーターを編集して値 \* を設定するか、すべてのユーザーが Analytic Server へのログインを許可されているグループを指定します。
- c. **hadoop.proxyuser.hive.hosts** パラメーターを編集して値 \* を設定するか、サービスとして Hive Metastore および Analytic Server の各インスタンスがインストールされているホストのリストを指定します。
- d. HDFS サービスを再起動します。

これらのステップの実行を完了した後、Analytic Server がインストールされていると、Analytic Server がサイレントかつ自動的に Kerberos の構成を行います。

### Kerberos を使用したシングル・サインオン (SSO) 用の HAProxy の構成

1. HAProxy の資料 (<http://www.haproxy.org/#docs>) に従って HAProxy を構成して開始します。
2. HAProxy ホスト用の Kerberos プリンシパル (HTTP/<proxyHostname>@<realm>) およびキータブ・ファイルを作成します。ここで、<proxyHostname> は HAProxy ホストの完全な名前、<realm> は Kerberos レalmです。
3. キータブ・ファイルを各 Analytic Server ホストに /etc/security/keytabs/spnego\_proxy.service.keytab としてコピーします。
4. このファイルのアクセス許可を各 Analytic Server ホストで更新します。次に例を示します。

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

5. Amabari コンソールを開き、Analytic Server の「Custom analytics.cfg」セクションで以下のプロパティを更新します。
6. 構成を保存し、Amabari コンソールからすべての Analytic Server サービスを再始動します。

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

これで、ユーザーは Kerberos SSO を使用して Analytic Server にログインできるようになりました。

### Kerberos の無効化

1. Ambari コンソールで Kerberosを無効化します。
2. Analytic Server サービスを停止します。
3. カスタム analytics.cfg から、以下のパラメーターを削除します。

```
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. 「Save」をクリックし、Analytic Server サービスを再起動します。

### Analytic Server コンソールへの Secure Sockets Layer (SSL) 接続の有効化

デフォルトでは、Analytic Server は自己署名証明書を生成して Secure Socket Layer (SSL) を有効にします。自己署名証明書を受け入れることにより、セキュア・ポートを使用して Analytic Server コンソールにアクセスできるようになります。HTTPS によるアクセスの安全性をさらに強化するには、サード・パーティー・ベンダーの証明書をインストールする必要があります。

サード・パーティー・ベンダーの証明書をインストールするには、以下のステップを実行します。

1. サード・パーティー・ベンダーの鍵ストア証明書およびトラストストア証明書を、すべての Analytic Server ノードで、同じディレクトリーにコピーします。例えば、`/home/as_user/security` です。

注: Analytic Server ユーザーには、このディレクトリーの読み取りアクセス権限が必要です。

2. Ambari の「Services」タブで、Analytic Server サービスの「Configs」タブに移動します。
3. **ssl.keystore.config** パラメーターを編集します。

```
<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>"/>
```

置き換える値:

- `<KEYSTORE-LOCATION>` に、鍵ストアの絶対位置を指定します。例: `/home/as_user/security/mykey.jks`
- `<TRUSTSTORE-LOCATION>` に、トラストストアの絶対位置に指定します。例: `/home/as_user/security/mytrust.jks`
- `<TYPE>` に、証明書のタイプを指定します。例: JKS、PKCS12、その他。
- `<PASSWORD>` に、Base64 暗号化形式の暗号化パスワードを指定します。エンコードには、`securityUtility` を使用できます。例えば、以下のように指定します。`/opt/ibm/spss/analyticsserver/3.0/ae_wlpserver/bin/securityUtility encode <password>`

自己署名証明書を生成する場合は、`securityUtility` を使用できます。例えば、以下のように指定します。`/opt/ibm/spss/analyticsserver/3.0/ae_wlpserver/bin/securityUtility createSSLCertificate --server=myserver --password=mypassword --validity=365 --subject=CN=mycompany,O=myOrg,C=myCountry`

`securityUtility` およびその他の SSL 設定について詳しくは、WebSphere Liberty Profile の資料を参照してください。

4. 「Save」をクリックし、Analytic Server サービスを再起動します。

## Essentials for R に対するサポートの有効化

Analytic Server は、R モデルのスコアリング、および R スクリプトの実行をサポートしています。

R に対するサポートを構成するには、Analytic Server が正常にインストールされた後で、以下を行います。

1. IBM SPSS Modeler Essentials for R の RPM 用の自己解凍型アーカイブ (BIN) をダウンロードします。Essentials for R は、<https://www14.software.ibm.com/webapp/iwm/web/preLogin.do?source=swg-tspspp> からダウンロードできます。ご使用のスタック、スタックのバージョン、およびハードウェア・アーキテクチャーに固有のファイルを選択します。
2. 自己解凍型バイナリー・ファイルを実行し、指示に従って (オプションで) ライセンスを表示し、ライセンスを受け入れて、オンライン・インストールまたはオフライン・インストールを選択します。

## オンライン・インストール

Ambari サーバー・ホストおよびクラスター内のすべてのノードが <http://ibm-open-platform.ibm.com> にアクセス可能な場合は、オンライン・インストールを選択してください。

[GPFSS (*Spectrum Scale*) のみ] ファイル [http://ibm-open-platform.ibm.com/repos/IBM-SPSS-ModelerEssentialsR/3.0.1.0/x86\\_64/IBM-SPSS-AnalyticServer-3.0.1.0.repo](http://ibm-open-platform.ibm.com/repos/IBM-SPSS-ModelerEssentialsR/3.0.1.0/x86_64/IBM-SPSS-AnalyticServer-3.0.1.0.repo) (x86) または [http://ibm-open-platform.ibm.com/repos/IBM-SPSS-ModelerEssentialsR/3.0.1.0/x86\\_64/IBM-SPSS-AnalyticServer-3.0.1.0.repo](http://ibm-open-platform.ibm.com/repos/IBM-SPSS-ModelerEssentialsR/3.0.1.0/x86_64/IBM-SPSS-AnalyticServer-3.0.1.0.repo) (ppc64le) をダウンロードし、それを、サービスとして Analytic Server Metastore を追加した先の各ノード上のフォルダー `/etc/yum.repos.d` (RHEL、CentOS) または `/etc/zypp/repos.d` (SLES) に移動します。

## オフライン・インストール

ご使用の Ambari サーバー・ホストがインターネットにアクセスできない場合は、オフラインを選択します。オフライン・インストールでは必要な RPM ファイルをダウンロードするため、<http://ibm-open-platform.ibm.com> にアクセス可能なマシンで実行する必要があります。その後、RPM ファイルを Ambari サーバー・ホストにコピーできます。

- a. 必要な Essentials for R の RPM ファイルを Ambari サーバー・ホスト上の任意の場所にコピーします。必要な RPM ファイルは、ご使用のディストリビューション、バージョン、およびアーキテクチャーによって以下のように異なります。

### **BigInsights 4.2 (x86\_64)**

`IBM-SPSS-ModelerEssentialsR-ambari-2.1-BI-4.2-8.4.0.0-1.x86_64.rpm`

### **BigInsights 4.2 (PPC64LE)**

`IBM-SPSS-ModelerEssentialsR-ambari-2.1-BI-4.2-8.4.0.0-1.ppc64le.rpm`

### **HDP 2.4 (x86\_64)**

`IBM-SPSS-ModelerEssentialsR-ambari-2.1-HDP-2.4-8.4.0.0-1.x86_64.rpm`

- b. RPM をインストールします。例えば、以下のコマンドは Essentials for R を BigInsights 4.1 にインストールします。

```
rpm -i IBM-SPSS-ModelerEssentialsR-ambari-2.1-BI-4.2-8.4.0.0-1.x86_64.rpm
```

3. Ambari サーバーを再起動します。

```
ambari-server restart
```

4. Ambari サーバーにログオンし、Ambari コンソールを使用して SPSS Essentials for R をサービスとしてインストールします。SPSS Essentials for R は、Analytic Server および Analytic Metastore がインストールされているすべてのホストにインストールする必要があります。

注: Ambari は R をインストールする前に `gcc-c++` および `gcc-gfortran` (RHEL)、および `gcc-fortran` (SUSE) のインストールを試行します。これらのパッケージは、R の Ambari サービス定義で依存関係として宣言されています。R のインストールおよび実行場所となるサーバーが、`gcc-c++` および `gcc-[g]fortran` の RPM をダウンロードするように構成されていること、あるいは、そのサーバーに GCC コンパイラーおよび FORTRAN コンパイラーがインストールされていることを確認してください。Essentials for R のインストールが失敗する場合は、Essentials for R をインストールする前にこれらのパッケージを手動でインストールしてください。

5. Analytic Server サービスをリフレッシュします。
6. 16 ページの『クライアント依存関係の更新』の手順に従って `update_clientdeps` スクリプトを実行します。
7. SPSS Modeler Server をホストするマシンに Essentials for R をインストールすることも必要です。詳しくは、SPSS Modeler の資料を参照してください。

## リレーショナル・データベース・ソースの有効化

各 Analytic Server ホストの共有ディレクトリー内に JDBC ドライバーを配置すると、Analytic Server でリレーショナル・データベース・ソースを使用できます。デフォルトでは、このディレクトリーは /usr/share/jdbc です。

共有ディレクトリーを変更するには、以下のステップを実行します。

1. Ambari の「Services」タブで、Analytic Server サービスの「Configs」タブに移動します。
2. 「Advanced analytics.cfg」セクションを開きます。
3. **jdbc.drivers.location** で、JDBC ドライバーの共有ディレクトリーを指定します。
4. 「Save」をクリックします。
5. Analytic Server サービスを停止します。
6. 「Refresh」をクリックします。
7. Analytic Server サービスを開始します。

表 1. サポート対象データベース

| データベース                         | サポート対象バージョン         | JDBC ドライバー jar                   | ベンダー      |
|--------------------------------|---------------------|----------------------------------|-----------|
| Amazon Redshift                | 8.0.2 以降            | RedshiftJDBC41-1.1.6.1006.jar 以降 | Amazon    |
| DashDB                         | Bluemix Service     | db2jcc.jar                       | IBM       |
| DB2 for Linux、UNIX、および 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 |
| Hive                           | 1.1, 1.2            | hive-jdbc-*.jar                  | Apache    |
| 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  |

注: Analytic Server をインストールする前に Redshift データ・ソースを作成した場合、Redshift データ・ソースを使用するには以下のステップを実行する必要があります。

1. Analytic Server コンソールで Redshift データ・ソースを開きます。
2. 「Redshift」データベース・データ・ソースを選択します。
3. Redshift のサーバー・アドレスを入力します。
4. データベース名とユーザー名を入力します。パスワードは自動的に入力されます。
5. データベース表を選択します。

## HCatalog データ・ソースの有効化

Analytic Server は、Hive/HCatalog を介して複数のデータ・ソースをサポートしています。一部のソースでは、手動での構成ステップが必要です。

1. データ・ソースを有効にするために必要な JAR ファイルを収集します。詳しくは、下のセクションを参照してください。
2. これらの JAR ファイルを、各 Analytic Server ノードの {HIVE\_HOME}/auxlib ディレクトリーおよび /usr/share/hive ディレクトリーに追加します。
3. Hive Metastore サービスを再起動します。
4. Analytic Metastore サービスをリフレッシュします。
5. Analytic Server サービスの各インスタンスを再起動します。

## NoSQL データベース

Analytic Server は、ベンダーから Hive ストレージ・ハンドラーが提供されている任意の NoSQL データベースをサポートします。

Apache HBase および Apache Accumulo のサポートを有効にするために、追加のステップは必要ありません。

その他の NoSQL データベースについては、データベース・ベンダーに連絡して、該当するストレージ・ハンドラーおよび関連する jar を取得してください。

## ファイル・ベース Hive 表

Analytic Server は、組み込みまたはカスタムの Hive SerDe (serializer-deserializer) が利用可能な任意のファイル・ベース Hive 表をサポートします。

XML ファイルを処理するための Hive XML SerDe は Maven の Central Repository (<http://search.maven.org/#search%7Cga%7C1%7Cchivexmlserde>) にあります。

## Apache Spark

HCatalog 入力データ・ソースで Spark (バージョン 1.5 以降) を使用する場合は、カスタムの analytics.cfg ファイルに spark.version プロパティを手動で追加する必要があります。

1. Amabri コンソールを開き、Analytic Server の「カスタム **analytics.cfg (Custom analytics.cfg)**」セクションに以下のプロパティを追加します。
  - 「キー」: spark.version
  - 「値」: 適切な Spark バージョン番号 (例えば 1.5) を入力します。
2. 構成を保存し、Amabri コンソールからすべての Analytic Server サービスを再始動します。

## Analytic Server で使用するポートの変更

デフォルトでは、Analytic Server はポート 9080 (HTTP 用) および 9443 (HTTPS 用) を使用します。ポートの設定を変更するには、以下のステップを実行します。

1. Amabri の「Services」タブで、Analytic Server サービスの「Configs」タブに移動します。
2. 「**Advanced analytics.cfg**」セクションを開きます。
3. 使用するポートを、**http.port** (HTTP ポート) および **https.port** (HTTPS ポート) に指定します。
4. 「**Save**」をクリックします。
5. Analytic Server サービスを再始動します。

## 高可用性 Analytic Server

クラスター内の複数のノードに Analytic Server をサービスとして追加することにより、高可用性構成にすることができます。

1. Ambari コンソールで、「Hosts」タブに移動します。
2. Analytic Server をまだサービスとして実行していないホストを選択します。
3. 「Summary」タブで、「Add」をクリックし、Analytic Serverを選択します。
4. 「追加の確認 (Confirm Add)」をクリックします。

## 小規模データ向けの JVM オプションの最適化

小規模な (M3R) ジョブの実行時にご使用のシステムを最適化するために、JVM プロパティを編集できます。

Ambari コンソールで、Analytic Server サービスの「Configs」タブの Advanced analytics-jvm-options セクションを参照します。以下のパラメーターを変更して、Analytic Server (Hadoop ではなく) をホストするサーバーで実行されるジョブのヒープ・サイズを設定します。これは小規模な (M3R) ジョブを実行する場合に重要です。システムを最適化するために、これらの値を調整する必要がある場合があります。

```
-Xms512M  
-Xmx2048M
```

## クライアント依存関係の更新

このセクションでは、update\_clientdeps スクリプトを使用して Analytic Server サービスの依存関係を更新する方法を説明します。

1. Ambari サーバー・ホストに root としてログインします。
2. ディレクトリを /var/lib/ambari-server/resources/stacks/<stack-name>/<stack-version>/services/ANALYTICSERVER/package/scripts に変更します。例を示します。  

```
cd "/var/lib/ambari-server/resources/stacks/HDP/2.4/services/ANALYTICSERVER/package/scripts"
```
3. 以下の引数を指定して、update\_clientdeps スクリプトを実行します。

**-u <ambari-user>**

Ambari アカウント・ユーザー名。

**-p <ambari-password>**

Ambari アカウント・ユーザーのパスワード。

**-h <ambari-host>**

Ambari サーバーのホスト名。

**-x <ambari-port>**

Ambari が listen しているポート。

以下の例を参照してください。

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

4. 以下のコマンドを使用して Ambari サーバーを再始動します。  

```
ambari-server restart
```

## Apache Knox の構成

Apache Knox Gateway は、Apache Hadoop サービスにセキュア・アクセスの単一ポイントを提供するシステムです。このシステムにより、ユーザー (クラスター・データにアクセスし、ジョブを実行する人)

およびオペレーター (アクセスを制御し、クラスターを管理する人) の両者の Hadoop セキュリティーが簡素化されます。この Gateway は、1 つ以上の Hadoop クラスターに機能やサービスを提供するサーバー (またはサーバーのクラスター) として実行されます。

注: IBM SPSS Analytic Server が Kerberos シングル・サインオン (SSO) とともに使用された場合、Apache Knox はサポートされません。

Apache Knox Gateway は、Hadoop クラスター・トポロジーの詳細を効果的に非表示にし、エンタープライズ LDAP および Kerberos と統合されます。以下のセクションでは、Apache Knox および Analytic Server の必要な構成タスクについての情報を提供します。

重要: Analytic Server は、Knox サーバーと同じクラスター・ノード上にはインストールできません。

## 前提条件

- Analytic Server は、Knox サーバーと同じクラスター・ノード上にはインストールできません。
- Analytic Server ノードは、パスワードなしの SSH 接続で Knox サーバーに接続する必要があります。パスワードなしの SSH 接続で、Analytic Server から Knox に移動します (「**Analytic Server**」 > 「**Knox**」)。
- Analytic Server のインストールは、Knox サービスのインストール後に行う必要があります。

場合によっては、予期しない問題が発生して構成ファイルが自動的にコピーされないことがあります。このような場合は、以下の構成ファイルを手動でコピーする必要があります。

- com.ibm.spss.knox\_0.7-3.0.0.jar: このファイルのコピー元の Analytic Server の場所:

```
<Analytic_Server_Installation_Path>/ae_wlpserver/usr/servers/aeserver/apps/AE_BOOT.war/WEB-INF/lib
```

コピー先の Knox サーバー・ノード:

```
/KnoxServicePath/ext
```

例: /usr/iop/4.1.0.0/knox/ext

- rewrite.xml および service.xml: これらのファイルのコピー元の Analytic Server の場所:

```
<Analytic_Server_Installation_Path>/ae_wlpserver/usr/servers/aeserver/configuration/knox
```

コピー先の Knox サーバー・ノード:

```
/KnoxServicePath/data/services
```

例: /usr/iop/4.1.0.0/knox/data/services

## Ambari の構成

Analytic Server サービスは、Ambari ユーザー・インターフェースで構成する必要があります。

1. Ambari ユーザー・インターフェースで、「**Knox**」 > 「**Configs**」 > 「**Advanced topology**」に移動します。現在の Knox 構成設定が「**content**」ウィンドウに表示されます。
2. 以下の <service> を Knox 構成に追加します。

```
<service>
  <role>ANALYTICSERVER</role>
  <url>http://{analyticserver-host}:{analyticserver-port}/analyticserver</url>
</service>
```

{analyticserver-host} および {analyticserver-port} は、Analytic Server の対応するサーバー名およびポート番号に置き換える必要があります。

- {analyticserver-host} URL は、Ambari ユーザー・インターフェース (「**SPSS Analytic Server**」 > 「**Summary**」 > 「**Analytic Server**」) にあります。
- {analyticserver-port} 番号は、Ambari ユーザー・インターフェース (「**SPSS Analytic Server**」 > 「**Configs**」 > 「**Advanced analytics.cfg**」 > 「**http.port**」) にあります。

注: Analytic Server が複数のノードにデプロイされていて、LoadBalancer が使用される場合、{analyticserver-host} および {analyticserver-port} は LoadBalancer の URL およびポート番号に対応する必要があります。

3. Knox サービスを再起動します。

LDAP が使用される場合、Knox のデフォルトは、指定された「Demo」LDAP になります。エンタープライズ LDAP サーバー (Microsoft LDAP や OpenLDAP など) に変更できます。

## Analytic Server の構成

Analytic Server に LDAP を使用するには、Apache Knox で使用されたものと同じ LDAP サーバーを使用するように、Analytic Server を構成する必要があります。以下の Ambari 設定の <value> 項目は、対応する Knox LDAP サーバー設定を反映するように更新する必要があります。

- main.ldapRealm.userDnTemplate
- main.ldapRealm.contextFactory.url

その値は、Ambari ユーザー・インターフェース (「**Knox**」 > 「**Configs**」 > 「**Advanced topology**」) で使用可能です。以下に例を示します。

```
<param>
  <name>main.ldapRealm.userDnTemplate</name>
  <value>uid={0},ou=people,dc=hadoop,dc=apache,dc=org</value>
</param>
<param>
  <name>main.ldapRealm.contextFactory.url</name>
  <value>ldap://{knox_host_name}:33389</value>
</param>
```

Knox LDAP 設定の更新後に Knox サービスを再起動します。

重要: Analytic Server の管理者パスワードは、Knox の管理者パスワードと同じである必要があります。

## Apache Knox の構成

1. Knox サーバーで、サブディレクトリー <knox\_server>/data/service/analyticserver/3.0 を作成し、service.xml ファイルと rewrite.xml ファイルをその新規ディレクトリーにアップロードします。これらの 2 つのファイルは Analytic Server 上の <analytic\_server>/configuration/knox/analyticserver/3.0.1 にあります (例えば /opt/ibm/spss/analyticserver/3.0/ae\_wlpserver/usr/servers/aeserver/configuration/knox/analyticserver/3.0/\*.xml)。
2. <knox\_server>/bin で、スクリプト ./knoxcli.sh redeploy --cluster default を実行します。
3. com.ibm.spss.knoxservice\_0.7.0-\*.jar ファイルを <knox\_server>/ext にアップロードします。このファイルは Analytic Server 上の <analytic\_server>/apps/AE\_BOOT.war/WEB-INF/lib/com.ibm.spss.knox\_0.7-3.0.1.0.jar にあります (例えば /opt/ibm/spss/analyticserver/3.0/ae\_wlpserver/usr/servers/aeserver/apps/AE\_BOOT.war/WEB-INF/lib/com.ibm.spss.knox\_0.7-3.0.1.0.jar)。

- Ambari ユーザー・インターフェースで、「Knox」 > 「Configs」 > 「Advanced topology」を選択して、以下の要素を追加します。

```
<service>
  <role>ANALYTICSERVER</role>
  <url>http://{AS-Host}:{AS-port}/analyticserver</url>
</service>
```

- Ambari ユーザー・インターフェースで、「Knox」 > 「Configs」 > 「Advanced users-ldif」を選択して、ユーザーを追加または更新します (例えば、admin、qauser1、qauser2)。
- 「Knox」 > 「Service Actions」 > 「Start Demo LDAP」を選択して、LDAP を再起動します。
- Knox サービスを再起動します。

## Apache Knox の Hortonworks Data Platform (HDP) へのインストール

以下のステップで、Apache Knox を HDP クラスターにインストールするプロセスの概要を説明します。

- Knox ユーザーが HDP クラスター上に存在するかどうか確認します。Knox ユーザーが存在しない場合は、作成する必要があります。
- Apache Knox をダウンロードして、/home/knox の下のフォルダーに解凍します。
- HDP で、Knox ユーザーに切り替え、knox フォルダーに移動します。Knox ユーザーには、すべての knox サブフォルダーに対する permission(RWX) が必要です。
- Analytic Server の Apache Knox を構成します。詳しくは、『Apache Knox の構成』セクションを参照してください。

- {knox}/data/services の下に analyticserver/3.0.1 フォルダー階層を作成します。
- rewrite.xml ファイルと service.xml ファイルをコピーします。コピー元の Analytic Server の場所:

```
/opt/ibm/spss/analyticserver/3.0.1/ae_wlpserver/usr/servers/aeserver/configuration/knox/analyticserver/3.0.1
```

コピー先の Knox サーバー・ノード:

```
{knox}/data/services/analyticserver/3.0.1
```

- Knox \*.jar ファイルをコピーします。コピー元の Analytic Server ホスト:

```
/opt/ibm/spss/analyticserver/3.0.1/ae_wlpserver/usr/servers/aeserver/apps/AE_BOOT.war/WEB-INF/lib/com.ibm.spss.knox_0.7-*.jar
```

コピー先の Knox ext ディレクトリー:

```
{knox}/ext
```

- 以下の例のように、{knox}/conf/topologies の default.xml ファイルを更新します。

注: ファイルが存在しない場合は、作成する必要があります。

```
<topology>
  <gateway>
    <provider>
      <role>authentication</role>
      <name>ShiroProvider</name>
      <enabled>true</enabled>
      <param>
        <name>sessionTimeout</name>
        <value>30</value>
      </param>
      <param>
        <name>main.LdapRealm</name>
        <value>org.apache.hadoop.gateway.shirorealm.KnoxLdapRealm</value>
      </param>
    </provider>
  </gateway>
</topology>
```

```

    <param>
      <name>main.ldapRealm.userDnTemplate</name>
      <value>uid={0},ou=people,dc=hadoop,dc=apache,dc=org</value>
    </param>
    <param>
      <name>main.ldapRealm.contextFactory.url</name>
      <value>ldap://localhost:33389</value>
    </param>
    <param>
      <name>main.ldapRealm.contextFactory.authenticationMechanism</name>
      <value>simple</value>
    </param>
    <param>
      <name>urls./*</name>
      <value>authcBasic</value>
    </param>
  </provider>
  <provider>
    <role>identity-assertion</role>
    <name>Default</name>
    <enabled>true</enabled>
  </provider>
  <provider>
    <role>authorization</role>
    <name>AclsAuthz</name>
    <enabled>true</enabled>
  </provider>
</gateway>

<!--other service-->
<service>
  <role>ANALYTICSERVER</role>
  <!--replace the {AS-host}nas {AS-port} with real value-->
  <url>http://{AS-host}:{AS-port}/analyticsserver</url>
</service>
</topology>

```

5. {knox}/bin/knoxccli.sh を実行します。
6. {knox}/bin/ldap.sh start を実行します。

注: スクリプトでは ポート 33389 が使用されます。そのポートが現在使用中でないことを確認してください。

7. {knox}/bin/gateway.sh start を実行します。

注: スクリプトでは ポート 8443 が使用されます。そのポートが現在使用中でないことを確認してください。

8. インストールを確認します。
  - a. Knox URL の Analytic Server に対して curl コマンドを実行します。

```
curl -ikvu {username}:{password} https://{knox-host}:8443/gateway/default/analyticsserver/admin
```

#### トラブルシューティング

**問題:** Analytic Server が、インストール後に Knox で動作しません。

**解決策:** Knox を停止し、{knox}/data/deployments/\* の下にあるすべてのファイルを削除してから、Knox を再起動します。

**問題:** Knox を介して Analytic Server にログインできません。

**解決策:** {knox}/conf/users.ldif 内のユーザーを確認します。既存のユーザーを更新するか、新規の Analytic Server ユーザーを追加します。Knox ユーザーのプリンシパルおよび資格情報は Analytic Server ユーザーのものと一致する必要があります。

## Apache Knox 対応の Analytic Server の URL 構造

Knox 対応の Analytic Server ユーザー・インターフェース URL は、`https://{knox-host}:{knox-port}/gateway/default/analyticserver/admin` です。

- https プロトコル - ユーザーは証明書を受け入れて、Web ブラウザーに進む必要があります。
- `knox-host` は、Knox のホストです。
- `knox-port` は、Knox のポート番号です。
- URI は、`gateway/default/analyticserver` です。

---

## アップグレードおよびマイグレーション

Analytic Server では、既存の Analytic Server インストール済み環境から新規インストール済み環境へのデータおよび構成設定のアップグレードまたはマイグレーションが可能です。

### バージョン 3.0 から 3.0.1 へのアップグレード

Analytic Server 3.0 の既存インストール済み環境がある場合、そのインストール済み環境をインプレースでバージョン 3.0.1 にアップグレードできます。

1. Ambari コンソールで、Analytic Server サービスを停止します。
2. インストールのタイプに応じて、以下のステップに従ってください。

#### オンライン・アップグレード

- a. Ambari サーバー・ホストおよびクラスター内のすべてのノードが <http://ibm-open-platform.ibm.com> にアクセス可能であることを確認します。
- b. ファイル `IBM-SPSS-AnalyticServer-3.0.1.0.repo` を `http://ibm-open-platform.ibm.com/repos/IBM-SPSS-AnalyticServer/3.0.1.0/x86_64/IBM-SPSS-AnalyticServer-3.0.1.0.repo` (x86) または `http://ibm-open-platform.ibm.com/repos/IBM-SPSS-AnalyticServer/3.0.1.0/ppc64le/IBM-SPSS-AnalyticServer-3.0.1.0.repo` (ppc64le) からそれぞれの Analytic Server ホスト上にダウンロードし、それをフォルダー `/etc/yum.repos.d` (RHEL または CentOS) または `/etc/zypp/repos.d` (SLES) に移動します。

#### オフライン・アップグレード

- a. オフライン・アップグレードでは必要な RPM ファイルをダウンロードするため、`http://ibm-open-platform.ibm.com/repos/IBM-SPSS-AnalyticServer/3.0.1.0/` にアクセス可能なマシンで実行する必要があります。
- b. Analytic Server の RPM ファイルのリポジトリとして機能する新規ディレクトリを作成します。以下の例を参照してください。

```
mkdir /home/root/repos/IBM-SPSS-AnalyticServer/3.0.1.0/x86_64
```
- c. このディレクトリに、必要な Analytic Server の RPM ファイルをコピーします。必要な RPM ファイルは、ご使用のディストリビューション、バージョン、およびアーキテクチャーによって異なります。BigInsights 4.1 の場合、必要なファイルは以下のとおりです。

表 2. BigInsights 4.2 RPM

| BigInsights 4.2 (x86_64)                                       | BigInsights 4.2 (PPC64LE)                                       | HDP 2.4 (x86_64)  |
|--|---|---|
| IBM-SPSS-AnalyticServer-ambari-2.1-BI-4.2-3.0.1.0-1.x86_64.rpm | IBM-SPSS-AnalyticServer-ambari-2.1-BI-4.2-3.0.1.0-1.ppc64le.rpm | IBM-SPSS-AnalyticServer-ambari-2.1-HDP-2.4-3.0.1.0-1.x86_64.rpm |

表 2. BigInsights 4.2 RPM (続き)

| BigInsights 4.2 (x86_64)                     | BigInsights 4.2 (PPC64LE)                     | HDP 2.4 (x86_64)                             |
|--|---|--|
| IBM-SPSS-AnalyticServer-3.0.1.0-1.x86_64.rpm | IBM-SPSS-AnalyticServer-3.0.1.0-1.ppc64le.rpm | IBM-SPSS-AnalyticServer-3.0.1.0-1.x86_64.rpm |

- d. ローカル・リポジトリの定義を作成します。例えば、`analyticserver.repo` というファイルを、`/etc/yum.repos.d/` (RHEL、CentOS の場合) または `/etc/zypp/repos.d/` (SLES の場合) に、以下の内容を指定して作成します。

```
[IBM-SPSS-AnalyticServer]
name=IBM-SPSS-AnalyticServer-3.0.1.0
baseurl=file:///{{path to local repository}}
enabled=1
gpgcheck=0
protect=1
```

- e. ローカル Yum リポジトリを作成します。以下の例を参照してください。

```
createrepo /home/root/repos/IBM-SPSS-AnalyticServer/3.0.1.0/x86_64
```

3. ローカル・キャッシュから Ambari メタデータを消去します。例えば、RHEL または CentOS 上のキャッシュを消去するには、以下のコマンドを実行します。

```
sudo yum clean all
```

注: Analytic Server リポジトリが 2 つ表示されている場合、yum コマンドは機能しません。結果として、Analytic Server に関連する、元の \*.repo ファイルを名前変更または削除する必要があります。SLES の場合、コマンドは以下のように変わります。

```
sudo zypper refresh
```

4. 各 Analytic Server ホストで RPM をアップグレードします。例えば、RHEL または CentOS 上でアップグレードを行うには、以下のコマンドを実行します。

```
chown -R as_user:hadoop /opt/ibm/spss/analyticserver/3.0
sudo yum upgrade IBM-SPSS-AnalyticServer
```

SLES の場合、コマンドは以下のように変わります。

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

5. スタックをリフレッシュします。

## BigInsights

- Ambari コンソールで、Analytic Server サービスを開始してから停止します。
- カスタムの「**Refresh**」アクションを実行します。

## Hortonworks

Analytic Server のいずれかのノードに移動して、以下のコマンドを実行します。

```
sudo -u as_user /opt/ibm/spss/analyticserver/3.0/bin/refresh.sh
```

6. オフライン・インストールのみ。ご使用の Ambari リポジトリ・ファイル `repoinfo.xml` (通常は `/var/lib/ambari-server/resources/stacks/${stackName}/${stackVersion}/repos/` にあります) に以下の行を追加して、ローカル Yum リポジトリを使用するように更新します。

```
<os type="host_os">
  <repo>
    <baseurl>file:///{{path to local repository}}/</baseurl>
```

```
<repoId>IBM-SPSS-AnalyticServer</repoId>
<reponame>IBM-SPSS-AnalyticServer-3.0.1.0</reponame>
</repo>
</os>
```

7. Zookeeper の状態をクリアします。Zookeeper の bin ディレクトリー (例えば /usr/iop/current/zookeeper-server/bin) で、以下のコマンドを実行します。

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

8. Ambari コンソールで、Analytic Server サービスを開始します。

### Analytic Server の新規バージョンへのマイグレーション

Analytic Server 2.0 または 2.1 の既存インストール済み環境があり、3.0 を購入した場合、2.0/2.1 の構成設定を 3.0/3.0.1 のインストール済み環境にマイグレーションできます。

#### Restrictions:

- 2.0 より前のバージョンがインストールされている場合、最初にその前のバージョンから 2.0/2.1 へのマイグレーションを行ってから、次にバージョン 2.0/2.1 から 3.0/3.0.1 へのマイグレーションを行う必要があります。
- 2.0/2.1 インストール済み環境と 3.0/3.0.1 インストール済み環境は、同じ Hadoop クラスター内には共存できません。2.0/2.1 インストール済み環境と同じ Hadoop クラスターを使用するように 3.0/3.0.1 インストール済み環境を構成すると、2.0/2.1 インストール済み環境は動作しなくなります。

### 2.0/2.1 から 3.0/3.0.1 へのマイグレーション・ステップ

1. 3 ページの『Ambari でのインストール』 の手順に従って、Analytic Server の新規インストールを実行します。
2. 古いインストール済み環境から新しいインストール済み環境に Analytic のルート・ディレクトリーをコピーします。

- a. Analytic のルートの場所が不明な場合は、`hadoop -fs ls` を実行します。Analytic のルートのパスの形式は `/user/aeuser/analytic-root` です。ここで、`aeuser` は、Analytic ルートを所有しているユーザー ID です。
- b. Analytic ルートの所有権を `aeuser` から `as_user` に変更します。

```
hadoop dfs -chown -R {as_user:{group}} {path to 2.0/2.1 analytic-root}
```

注: マイグレーション後に既存の Analytic Server インストール済み環境を使用する予定の場合は、`analytic-root` ディレクトリーのコピーを HDFS 内に作成してから、そのディレクトリーのコピーで所有権を変更します。

- c. Analytic Server の新規インストール済み環境のホストに `as_user` としてログインします。`/user/as_user/analytic-root` ディレクトリーが存在する場合は削除します。
- d. 以下のコピー・スクリプトを実行します。

```
hadoop distcp http://{host of 2.0/2.1 namenode}:50070/{path to 2.0/2.1 analytic-root}
hdfs://{host of 3.0/3.0.1 namenode}/user/as_user/analytic-root
```

3. Ambari コンソールで、Analytic Server サービスを停止します。
4. Analytic Metastore サービスが実行されていることを確認します。
5. 古いインストール済み環境から構成設定を収集します。
  - a. 新規インストール済み環境の `configcollector.zip` アーカイブを、古いインストール済み環境の `{AS_ROOT}¥tools` にコピーします。

- b. コピーした configcollector.zip を解凍します。これにより、古いインストール済み環境内に新規の configcollector サブディレクトリーが作成されます。
  - c. {AS\_ROOT}¥tools¥configcollector 内の **configcollector** スクリプトを実行して、古いインストール済み環境内の構成収集ツールを実行します。その結果生成された圧縮ファイル (ZIP) を、新規インストール済み環境をホストするサーバーにコピーします。
6. **migrationtool** スクリプトを実行し、構成収集ツールによって作成された圧縮ファイルのパスを引数として渡すことで、マイグレーション・ツールを実行します。次に例を示します。
- ```
migrationtool.sh /opt/ibm/spss/analyticserver/3.0/ASConfiguration_2.1.0.0.xxx.zip
```
7. Zookeeper の状態をクリアします。Zookeeper の bin ディレクトリー (Hortonworks 上の /usr/hdp/current/zookeeper-client や BigInsights 上の /usr/iop/current/zookeeper-server など) で、以下のコマンドを実行します。
- ```
./zkCli.sh rmr /AnalyticServer
```
8. Ambari コンソールで、Analytic Server サービスを開始します。

注: 既存の Analytic Server インストール済み環境で使用するよう R を構成している場合、新規 Analytic Server インストール済み環境で R を構成するステップに従う必要があります。

---

## アンインストール

重要: Essentials for R がインストールされている場合、まず `remove_R.sh` スクリプトを実行する必要があります。Analytic Server をアンインストールする前に、Essentials for R のアンインストールに失敗すると、後から Essentials for R をアンインストールできなくなります。Analytic Server がアンインストールされると、`remove_R.sh` スクリプトは削除されます。Essentials for R のアンインストールについては、25 ページの『Essentials for R のアンインストール』を参照してください。

1. Analytic Metastore ホストで、{AS\_ROOT}/bin ディレクトリーにある `remove_as.sh` スクリプトを、以下のパラメーターを指定して実行します。
  - u** 必須。Ambari サーバー管理者のユーザー ID。
  - p** 必須。Ambari サーバー管理者のパスワード。
  - h** 必須。Ambari サーバー・ホスト名。
  - x** 必須。Ambari サーバー・ポート。
  - l** オプション。セキュア・モードを有効にします。

以下に例を示します。

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

クラスター内の Ambari ホスト `one.cluster` から Analytic Server を削除します。

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

クラスター内の Ambari ホスト `one.cluster` から Analytic Server をセキュア・モードで削除します。

注: この操作により、HDFS 上の Analytic Server フォルダーが削除されます。

注: この操作では、Analytic Server に関連付けられた DB2 スキーマは一切削除されません。スキーマを手動で削除する方法については、DB2 の資料を参照してください。

## Essentials for R のアンインストール

1. Essentials for R ホストで、{AS\_ROOT}/bin ディレクトリーにある `remove_r.sh` スクリプトを、以下のパラメーターを指定して実行します。

- u** 必須。Ambari サーバー管理者のユーザー ID。
- p** 必須。Ambari サーバー管理者のパスワード。
- h** 必須。Ambari サーバー・ホスト名。
- x** 必須。Ambari サーバー・ポート。
- l** オプション。セキュア・モードを有効にします。

以下に例を示します。

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

クラスター内の Ambari ホスト `one.cluster` から Essentials for R を削除します。

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

クラスター内の Ambari ホスト `one.cluster` から Essentials for R をセキュア・モードで削除します。

2. Ambari サーバー・サービス・ディレクトリーから R サービス・ディレクトリーを削除します。例えば、BigInsights 4.2 の場合、ESSENTIALR ディレクトリーは `/var/lib/ambari-server/resources/stacks/BigInsights/4.2/services` にあります。
3. Ambari コンソールで、Essentials for R サービスが存在していないことを確認します。



---

## 第 3 章 Cloudera のインストールおよび構成

---

### Cloudera の概要

Cloudera は、オープン・ソースの Apache Hadoop ディストリビューションです。Cloudera Distribution Including Apache Hadoop (CDH) は、当該テクノロジーのエンタープライズ・クラスのデプロイメントを対象としています。

Analytic Server は CDH プラットフォームで実行できます。CDH には、大規模なデータ・セット (主に MapReduce および HDFS) の確実かつスケーラブルな分散データ処理を実現する Hadoop の主要なコア要素が含まれています。また、セキュリティ、高可用性、およびハードウェアや他のソフトウェアとの統合を実現するその他のエンタープライズ向けのコンポーネントも含まれています。

---

### Cloudera 固有の前提条件

一般的な前提条件に加えて、以下の情報を確認してください。

#### サービス

各 Analytic Server ホストに以下のインスタンスがインストールされていることを確認してください。

- HDFS: Gateway、DataNode または NameNode
- Hive: Gateway、Hive Metastore Server または HiveServer2
- Yarn: Gateway、ResourceManager または NodeManager

以下のインスタンスは、その機能が使用される場合にのみ必要です。

- Accumulo: Gateway
- HBase: Gateway、Master または RegionServer

#### メタデータ・リポジトリ

MySQL を Analytic Server のメタデータ・リポジトリとして使用する場合は、『Analytic Server 用の MySQL の構成』の手順に従ってください。

### Analytic Server 用の MySQL の構成

Cloudera Manager で IBM SPSS Analytic Server を構成するには、MySQL サーバー・データベースをインストールして構成する必要があります。

1. MySQL データベースが格納されているノードのコマンド・ウィンドウから以下のコマンドを実行します。

```
yum install mysql-server
```

注: SuSE Linux の場合は `zypper install mysql` を使用してください。

2. 各 Cloudera クラスター・ノードのコマンド・ウィンドウから以下のコマンドを実行します。

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

注: SuSE Linux の場合は `sudo zypper install mysql-connector-java` を使用してください。

3. Analytic Server が MySQL データベースへのアクセス時に使用する Analytic Server のデータベース名、データベースのユーザー名、およびデータベースのパスワードを決定し、メモを取ります。

- 『Cloudera でのインストール』の手順に従って Analytic Server をインストールします。
- Cloudera によって管理されているいずれかのサーバーから、MySQL データベースがインストールされているノードに `/opt/cloudera/parcels/AnalyticServer/bin/add_mysql_user.sh` スクリプトをコピーします。ユーザー固有の構成に適したパラメーターを指定してそのスクリプトを実行します。以下に例を示します。

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

**Notes:** データベースがセキュア・モード (root ユーザー・パスワードが設定される) で実行される場合は `a -r <dbRootPassword>` パラメーターが必須になります。

root 以外のユーザー名を使用してデータベースがセキュア・モードで実行されている場合は `-r <dbUserPassword>` パラメーターおよび `-t <dbUserName>` パラメーターが必須になります。

- Cloudera Manager を開き、Analytic Server サービスの「構成 (Configuration)」タブに移動します。
  - 「**Analytic Server metastore driver class (jndi.aedb.driver)**」プロパティーで、`com.mysql.jdbc.Driver` を選択します。
  - Analytic Server の構成項目を指定したペインで前にメモを取った Analytic Server のデータベース名、データベースのユーザー名、およびデータベースのパスワードに一致する値を指定する必要があります。「**Analytic Server metastore repository URL (jndi.aedb.url)**」プロパティー、「**Analytic Server metastore username (jndi.aedb.username)**」プロパティー、および「**Analytic Server metastore password (jndi.aedb.password)**」プロパティーを、`add_mysql_user.sh` コマンドに指定した値と一致するように更新する必要があります。

---

## Cloudera でのインストール

以下のステップでは、Cloudera Manager で IBM SPSS Analytic Server を手動でインストールするプロセスについて説明します。

### オンライン・インストール

- Cloudera の自己解凍型 `*.bin` インストーラーをダウンロードして Cloudera Manager マスター・クラスター・ノードで実行します。ご使用条件に同意し、デフォルトの CSD インストール・ディレクトリーを維持してインストールのプロンプトに従ってください。

注: CSD ディレクトリーをデフォルトの場所から変更した場合は、別の CSD ディレクトリーを指定する必要があります。

- インストールが完了した後に Cloudera Manager を再始動します。
- Cloudera Manager インターフェース (例えば、`http://${CM_HOST}:7180/cmf/login` をデフォルトのログイン資格情報 `admin/admin` を使用して) を開き、「リモート・パーセル・リポジトリーの **URL (Remote Parcel Repository URLs)**」をリフレッシュし、URL が正しいことを確認します。以下に例を示します。

```
http://ibm-open-platform.ibm.com/repos/IBM-SPSS-AnalyticServer/3.0.1.0/cloudera/parcels/latest/
```

or

```
http://ibm-open-platform.ibm.com/repos/IBM-SPSS-AnalyticServer/3.0.1.0/cloudera/
```

注: 「パーセルの更新頻度 (**Parcel Update Frequency**)」および「リモート・パーセル・リポジトリーの **URL (Remote Parcel Repository URLs)**」は、ユーザーの固有のニーズに合わせて更新できます。

4. Cloudera Manager がパーセル・ファイルをリフレッシュした後で (「新しいパーセルの確認 (**Check for New Parcels**)」をクリックすることでパーセル・ファイルを手動でリフレッシュできます)、「**AnalyticServer**」パーセルの状況が「リモートで使用可能 (**Available Remotely**)」に設定されていることが分かります。
5. 「ダウンロード (**Download**)」 > 「配布 (**Distribute**)」 > 「アクティブ化 (**Activate**)」を選択します。「**AnalyticServer**」パーセルの状況が「配布済み、アクティブ化済み (**Distributed, Activated**)」に更新されます。
6. Analytic Server の MySQL を構成します。
7. Cloudera Manager で Analytic Server をサービスとして追加し、Analytic Server を配置する場所を決定します。以下の情報を「サービスの追加ウィザード (Add Service Wizard)」に指定する必要があります。
  - Analytic Server metastore ユーザー名
  - Analytic Server metastore パスワード

「サービスの追加ウィザード (Add Service Wizard)」には、サービス作成プロセスの各フェーズにおける全体の進行状況が表示されます。また、クラスターでサービスが正常にインストールおよび構成されたときに最終確認メッセージが表示されます。

注: Analytic Server が正常にインストールされた後に、Cloudera Manager の Analytic Server サービス・ページの「アクション (Actions)」リストで「**Analytic Server Metastore** の作成 (**Create Analytic Server Metastore**)」をクリックしないでください。metastore を作成すると、既存のメタデータ・リポジトリが上書きされます。

## オフライン・インストール

オフライン・インストールのステップは、特定のオペレーティング・システムに適したパーセル・ファイルとメタデータをユーザーが手動でダウンロードする必要がある点を除いて、オンラインのステップと同じです。

RedHat Linux では、以下のファイルが必要です。

- AnalyticServer-3.0.1.0-el6.parcel
- AnalyticServer-3.0.1.0-el6.parcel.sha
- manifest.json

または

- AnalyticServer-3.0.1.0-el7.parcel
- AnalyticServer-3.0.1.0-el7.parcel.sha

SuSE Linux では、以下のファイルが必要です。

- AnalyticServer-3.0.1.0-sles11.parcel
- AnalyticServer-3.0.1.0-sles11.parcel.sha
- manifest.json

1. Cloudera の自己解凍型 \*.bin インストーラーをダウンロードして Cloudera Manager マスター・クラスター・ノードで実行します。ご使用条件に同意し、デフォルトの CSD インストール・ディレクトリを維持してインストールのプロンプトに従ってください。

注: CSD ディレクトリーがデフォルトの場所でない場合は、別の CSD ディレクトリーを指定する必要があります。

- 必要なパーセル・ファイルとメタデータ・ファイルを、Cloudera Manager マスター・クラスター・ノード上の Cloudera repo パスにコピーします。デフォルトのパスは `/opt/cloudera/parcel-repo` です (このパスは Cloudera Manager ユーザー・インターフェースで構成可能です)。

Cloudera Manager が「**AnalyticServer**」パーセルをリフレッシュした後でそのパーセルが「ダウンロード済み (**downloaded**)」と表示されます。「新しいパーセルの確認 (**Check for New Parcels**)」をクリックすると強制的にリフレッシュできます。

- 「配布 (**Distribute**)」 > 「アクティブ化 (**Activate**)」をクリックします。

「**AnalyticServer**」パーセルが「配布済み」および「アクティブ化済み」と表示されます。

## Cloudera での Analytic Server のアップグレード

Analytic Server 3.0 の既存インストール済み環境がある場合、そのインストール済み環境をインプレースでバージョン 3.0.1 にアップグレードできます。

- Cloudera Manager で、Analytic Server サービスを停止し、削除します。
- Cloudera Manager で、Analytic Server の以前のバージョンを非アクティブにします。
- Analytic Server 3.0.1 のインストール方法については、21 ページの『アップグレードおよびマイグレーション』の『オンライン』セクションまたは『オフライン』セクションを参照してください。
- Analytic Server サービスがインストールされ、Cloudera Manager に追加された後、「**Analytic Server** バイナリーのリフレッシュ (**Refresh Analytic Server Binaries**)」を実行します。これで Analytic Server 3.0.1 が使用できるようになりました。

---

## Cloudera の構成

インストール後、オプションで Cloudera Manager を使用して Analytic Server を構成し、管理することができます。

注: Analytic Server ファイル・パスには以下の規則が使用されます。

- {AS\_ROOT} は、Analytic Server がデプロイされている場所を示します (例えば、`/opt/cloudera/parcels/AnalyticServer`)。
- {AS\_SERVER\_ROOT} は、構成ファイル、ログ・ファイル、およびサーバー・ファイルの場所を示します (例えば、`/opt/cloudera/parcels/AnalyticServer/ae_wlpserver/usr/servers/aeserver`)。
- {AS\_HOME} は、Analytic Server がルート・フォルダーとして使用する HDFS 上の場所を示します (例えば、`/user/as_user/analytic-root`)。

## セキュリティー

**security\_cfg** パラメーターは、Analytic Server システムにプリンシパルとして追加できるユーザーとグループのレジストリーを定義します。

デフォルトでは、基本レジストリーは、単一のユーザー `admin` およびパスワード `admin` を指定して定義されています。**security\_cfg** を編集するか、またはセキュリティー・プロバイダーとして Kerberos を構成することにより、このレジストリーを変更できます。**security\_cfg** パラメーターは、Analytic Server サービスの「構成 (Configuration)」タブの「**Analytic Server Advanced Configuration Snippet**」セクションにあります。

注: **security\_cfg** パラメーターを編集してレジストリーを変更する場合、新規の各ユーザーをプリンシパルとして Analytic Server システムに追加する必要があります。テナント管理について詳しくは、「IBM SPSS Analytic Server 管理者ガイド」を参照してください。

## 基本レジストリーの変更

基本レジストリーを使用して、**security\_cfg** パラメーター内にユーザーとグループのデータベースを定義できます。

デフォルトの基本レジストリーは以下のようになっています。

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

変更後の基本レジストリーの例を以下に示します。

```
<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>
```

securityUtility ツール ({AS\_ROOT}/ae\_wlpserver/bin にあります) を使用してパスワードをエンコードすることで、パスワードの値を難読化できます。

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

注: securityUtility ツールについて詳しくは、[http://www-01.ibm.com/support/knowledgecenter/SSD28V\\_8.5.5/com.ibm.websphere.wlp.core.doc/ae/rwlp\\_command\\_securityutil.html](http://www-01.ibm.com/support/knowledgecenter/SSD28V_8.5.5/com.ibm.websphere.wlp.core.doc/ae/rwlp_command_securityutil.html) を参照してください。

注: 基本レジストリーは、サンドボックス環境では有用ですが、実稼働環境ではお勧めしません。

## LDAP レジストリーの構成

LDAP レジストリーは、Active Directory や OpenLDAP などの外部 LDAP サーバーを使用してユーザーを認証できるようにします。

以下に、OpenLDAP の ldapRegistry の例を示します。

```
<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="(&(uid=%v)(objectClass=inetOrgPerson))"
  groupFilter="(&(cn=%v)(|(objectClass=organizationalUnit)))"
  groupMemberIdMap="posixGroup:memberUid"/>
</ldapRegistry>

```

追加の構成例については、テンプレート・フォルダー {AS\_ROOT}/ae\_wlpserver/templates/config を参照してください。

注: Analytic Server での LDAP のサポートは WebSphere Liberty によって制御されます。詳しくは、Configuring LDAP user registries in Liberty を参照してください。

## Analytic Server から LDAP への Secure Sockets Layer (SSL) 接続の構成

1. Analytic Server マシンのそれぞれに Analytic Server ユーザーとしてログインし、SSL 証明書の共通ディレクトリーを作成します。

注: Cloudera では、Analytic Server ユーザーは常に as\_user になり、これは変更できません。

2. 鍵ストア・ファイルおよびトラストストア・ファイルを、すべての Analytic Server マシンの共通ディレクトリーにコピーします。また、LDAP クライアントの CA 証明書をトラストストアに追加します。以下に、手順例を示します。

```

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

```

注: JAVA\_HOME は、Analytic Server の起動に使用するのと同じ JRE です。

3. securityUtility ツール ({AS\_ROOT}/ae\_wlpserver/bin にあります) を使用してパスワードをエンコードすることで、パスワードの値を難読化できます。次に例を示します。

```

securityUtility encode changeit
{xor}Pdc+MTg6Nis=

```

4. Cloudera Manager にログインし、Analytic Server の構成設定 **ssl\_cfg** を、正しい SSL 構成設定で更新します。次に例を示します。

```

<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"/>

```

注: 鍵ファイルおよびトラストストア・ファイルについては、絶対パスを使用してください。

5. Analytic Server の構成設定 **security\_cfg** を、正しい LDAP 構成設定で更新します。例えば、**ldapRegistry** 要素の場合、**sslEnabled** 属性を true に設定し、**sslRef** 属性を defaultSSLConfig に設定します。

## Kerberos の構成

Analytic Server は、Cloudera での Kerberos をサポートします。

1. Analytic Server へのアクセス権を付与する予定のすべてのユーザーについて、Kerberos ユーザー・リポジトリー内にアカウントを作成します。

注: Analytic Server インストール済み環境で基本レジストリーを使用する場合、このレジストリーには、すべての Kerberos ユーザー・アカウントが、パスワードとして "-" を使用して、指定されていなければなりません。次に例を示します。

```
<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. 前のステップで、Analytic Server および Hadoop の各ノードで作成したそれぞれのユーザーについて、OS ユーザー・アカウントを作成します。
  - これらのユーザーの UID は、すべてのマシンで一致させてください。kinit コマンドを使用して各アカウントにログオンして、これをテストすることができます。
  - UID が、Yarn の「ジョブをサブミットするための最小ユーザー ID (Minimum user ID for submitting job)」設定に従っていることを確認してください。これは、`container-executor.cfg` 内の `min.user.id` パラメーターです。例えば、`min.user.id` が 1000 の場合、作成される各ユーザー・アカウントの UID は 1000 以上でなければなりません。
3. Analytic Server のすべてのプリンシパルについて、HDFS 上にユーザーのホーム・フォルダーを作成します。例えば、Analytic Server システムに `testuser1` を追加した場合、HDFS 上に `/user/testuser1` のようなホーム・フォルダーを作成し、`testuser1` がこのフォルダーに対する読み取り権限と書き込み権限を持つようにします。
4. [オプション] HCatalog データ・ソースを使用する予定であり、Analytic Server が Hive Metastore とは別のマシンにインストールされている場合、HDFS で Hive クライアント名を使用する必要があります。
  - a. Cloudera Manager で、HDFS サービスの「構成 (Configuration)」タブに移動します。

注: 以下のパラメーターがまだ設定されていない場合、「構成 (Configuration)」タブにそれらのパラメーターが表示されない可能性があります。その場合は、検索を実行して見つけてください。
  - b. `hadoop.proxyuser.hive.groups` パラメーターを編集して値 \* を設定するか、すべてのユーザーが Analytic Server へのログインを許可されているグループを指定します。
  - c. `hadoop.proxyuser.hive.hosts` パラメーターを編集して値 \* を設定するか、サービスとして Hive Metastore および Analytic Server の各インスタンスがインストールされているホストのリストを指定します。
  - d. HDFS サービスを再起動します。

これらのステップの実行を完了した後、Analytic Server がインストールされていると、Analytic Server がサイレントかつ自動的に Kerberos の構成を行います。

## Kerberos を使用したシングル・サインオン (SSO) 用の HAProxy の構成

1. HAProxy の資料 (<http://www.haproxy.org/#docs>) に従って HAProxy を構成して開始します。

2. HAProxy ホスト用の Kerberos プリンシパル (HTTP/<proxyHostname>@<realm>) およびキータブ・ファイルを作成します。ここで、<proxyHostname> は HAProxy ホストの完全な名前、<realm> は Kerberos レalmです。
3. キータブ・ファイルを各 Analytic Server ホストに /etc/security/keytabs/spnego\_proxy.service.keytab としてコピーします。
4. このファイルのアクセス許可を各 Analytic Server ホストで更新します。次に例を示します。

```
chown root:hadoop /etc/security/keytabs/spnego_proxy.service.keytab
chmod 440 /etc/security/keytabs/spnego_proxy.service.keytab
```
5. Cloudera Manager を開き、Analytic Server の「**Analytic Server Advanced Configuration Snippet (Safety Valve) for analyticserver-conf/config.properties**」領域で以下のプロパティを追加するか、更新します。

```
web.authentication.kerberos.keytab=/etc/security/keytabs/spnego_proxy.service.keytab
web.authentication.kerberos.principal=HTTP/<proxy machine full name>@<realm>
```
6. 構成を保存し、Cloudera Manager からすべての Analytic Server サービスを再始動します。
7. Kerberos を使用するようにブラウザーを構成することをユーザーに指示します。

これで、ユーザーは Kerberos SSO を使用して Analytic Server にログインできるようになりました。

## Kerberos の無効化

1. Ambari コンソールで Kerberosを無効化します。
2. Analytic Server サービスを停止します。
3. 「**Analytic Server Advanced Configuration Snippet (Safety Valve) for analyticserver-conf/config.properties**」領域から以下のパラメーターを削除します。

```
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. 「変更の保存 (**Save Changes**)」をクリックし、Analytic Server サービスを再始動します。

## Analytic Server コンソールへの Secure Sockets Layer (SSL) 接続の有効化

デフォルトでは、Analytic Server は自己署名証明書を生成して Secure Socket Layer (SSL) を有効にします。自己署名証明書を受け入れることにより、セキュア・ポートを使用して Analytic Server コンソールにアクセスできるようになります。HTTPS によるアクセスの安全性をさらに強化するには、サード・パーティー・ベンダーの証明書をインストールする必要があります。

サード・パーティー・ベンダーの証明書をインストールするには、以下のステップを実行します。

1. サード・パーティー・ベンダーの鍵ストア証明書およびトラストストア証明書を、すべての Analytic Server ノードで、同じディレクトリーにコピーします。例えば、/home/as\_user/security です。

注: Analytic Server ユーザーには、このディレクトリーの読み取りアクセス権限が必要です。

2. Cloudera Manager で、Analytic Server サービスの「構成 (Configuration)」タブに移動します。
3. **ssl\_cfg** パラメーターを編集します。

```
<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>"/>

```

置き換える値:

- <KEYSTORE-LOCATION> に、鍵ストアの絶対位置を指定します。例: /home/as\_user/security/mykey.jks
- <TRUSTSTORE-LOCATION> に、トラストストアの絶対位置に指定します。例: /home/as\_user/security/mytrust.jks
- <TYPE> に、証明書のタイプを指定します。例: JKS、PKCS12、その他。
- <PASSWORD> に、Base64 暗号化形式の暗号化パスワードを指定します。エンコードには、securityUtility を使用できます。例: {AS\_ROOT}/ae\_wlpserver/bin/securityUtility encode <password>

自己署名証明書を生成する場合は、securityUtility を使用できます。例: {AS\_ROOT}/ae\_wlpserver/bin/securityUtility createSSLCertificate --server=myserver --password=myspassword --validity=365 --subject=CN=mycompany,O=myOrg,C=myCountry。securityUtility およびその他の SSL 設定について詳しくは、WebSphere Liberty Profile の資料を参照してください。

4. 「変更の保存 (Save Changes)」をクリックし、Analytic Server サービスを再始動します。

## Essentials for R に対するサポートの有効化

Analytic Server は、R モデルのスコアリング、および R スクリプトの実行をサポートしています。

Cloudera Manager で Analytic Server が正常にインストールされた後で Essentials for R をインストールするには、以下を行います。

1. IBM SPSS Modeler Essentials for R の RPM 用の自己解凍型アーカイブ (BIN) をダウンロードします。Essentials for R は、<https://www14.software.ibm.com/webapp/iwm/web/preLogin.do?source=swg-tspssp> からダウンロードできます。ご使用のスタック、スタックのバージョン、およびハードウェア・アーキテクチャーに固有のファイルを選択します。
2. Cloudera Manager サーバー・ホストで root ユーザーまたは sudo ユーザーとして自己解凍型アーカイブを実行します。以下のパッケージがインストール済みであるか、構成済みのリポジトリから使用可能である必要があります。
  - Red Hat Linux: gcc-gfortran、zip、gcc-c++
  - SUSE Linux: gcc-fortran、zip、gcc-c++
3. この自己解凍型インストーラーは、以下のタスクを行います。
  - a. 必要なライセンスを表示し、それに同意するようにインストーラーでプロンプトを出します。
  - b. R のソースの場所を入力するか、またはデフォルトの場所で続行するようにインストーラーでプロンプトを出します。インストールされているデフォルトの R バージョンは 3.1.0 です。別のバージョンをインストールするには、以下を行います。
    - オンライン・インストール: 必要な R バージョン・アーカイブの URL を指定します。例えば、R 2.15.3 の場合は <https://cran.r-project.org/src/base/R-2/R-2.15.3.tar.gz> です。

- オフライン・インストール: 必要な R バージョン・アーカイブをダウンロードし、Cloudera Manager サーバー・ホストにコピーします。アーカイブの名前を変更しないでください (デフォルトの名前は R-x.x.x.tar.gz です)。コピーした R アーカイブの URL を `file://<R_archive_directory>/R-x.x.x.tar.gz` のように指定します。R-2.15.3.tar.gz アーカイブをダウンロードして /root にコピーした場合、その URL は `file:///root/R-2.15.3.tar.gz` になります。

注: その他の R バージョンは <https://cran.r-project.org/src/base/> にあります。

- c. R が必要とするパッケージをインストールします。
  - d. R および Essentials for R プラグインをダウンロードしてインストールします。
  - e. パーセルおよび `parcel.sha` ファイルを作成し、それらを `/opt/cloudera/parcel-repo` にコピーします。この場所が変更されている場合は、正しい場所を入力します。
4. インストールが完了したら、Cloudera Manager で「Essentials for R」パーセルを配布およびアクティブ化します (「新しいパーセルの確認 (Check for New Parcels)」をクリックしてパーセル・リストをリフレッシュします)。
  5. Analytic Server サービスが既にインストールされている場合は、以下を行います。
    - a. サービスを停止します。
    - b. Analytic Server バイナリーをリフレッシュします。
    - c. サービスを開始して Essentials for R のインストールを完了します。
  6. Analytic Server サービスがインストールされていない場合は、そのインストールを進めます。

注: すべての Analytic Server ホストに適切なアーカイブ (zip および unzip) パッケージがインストールされている必要があります。

## リレーショナル・データベース・ソースの有効化

各 Analytic Server ホストの共有ディレクトリー内に JDBC ドライバーを配置すると、Analytic Server でリレーショナル・データベース・ソースを使用できます。デフォルトでは、このディレクトリーは `/usr/share/jdbc` です。

共有ディレクトリーを変更するには、以下のステップを実行します。

1. Cloudera Manager で、Analytic Server サービスの「構成 (Configuration)」タブに移動します。
2. `jdbc.drivers.location` で、JDBC ドライバーの共有ディレクトリーを指定します。
3. 「変更の保存 (Save Changes)」をクリックします。
4. 「アクション (Action)」ドロップダウンから「停止 (Stop)」を選択して Analytic Server サービスを停止します。
5. 「アクション (Action)」ドロップダウンから「Analytic Server バイナリーのリフレッシュ (Refresh Analytic Server Binaries)」を選択します。
6. 「アクション (Action)」ドロップダウンから「開始 (Start)」を選択して Analytic Server サービスを開始します。

表 3. サポート対象データベース

| データベース          | サポート対象バージョン     | JDBC ドライバー jar                   | ベンダー   |
|-----------------|-----------------|----------------------------------|--------|
| Amazon Redshift | 8.0.2 以降        | RedshiftJDBC41-1.1.6.1006.jar 以降 | Amazon |
| DashDB          | Bluemix Service | db2jcc.jar                       | IBM    |

表 3. サポート対象データベース (続き)

| データベース                         | サポート対象バージョン         | JDBC ドライバー jar                       | ベンダー      |
|--------------------------------|---------------------|--------------------------------------|-----------|
| DB2 for Linux、UNIX、および Windows | 10.5, 10.1, 9.7     | db2jcc.jar                           | IBM       |
| DB2 z/OS                       | 11, 10              | db2jcc.jar、<br>db2_license_cisuz.jar | IBM       |
| Greenplum                      | 5、4.2.x             | postgresql.jar                       | Greenplum |
| Hive                           | 1.1, 1.2            | hive-jdbc-*.jar                      | Apache    |
| 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、<br>terajdbc4.jar    | Teradata  |

## HCatalog データ・ソースの有効化

Analytic Server は、Hive/HCatalog を介して複数のデータ・ソースをサポートしています。一部のソースでは、手動での構成ステップが必要です。

1. データ・ソースを有効にするために必要な JAR ファイルを収集します。詳しくは、下のセクションを参照してください。
2. これらの JAR ファイルを、各 Analytic Server ノードの {HIVE\_HOME}/auxlib ディレクトリーおよび /usr/share/hive ディレクトリーに追加します。
3. Hive Metastore サービスを再起動します。
4. Analytic Server サービスの各インスタンスをすべて再起動します。

## NoSQL データベース

Analytic Server は、ベンダーから Hive ストレージ・ハンドラーが提供されている任意の NoSQL データベースをサポートします。

Apache HBase および Apache Accumulo のサポートを有効にするために、追加のステップは必要ありません。

その他の NoSQL データベースについては、データベース・ベンダーに連絡して、該当するストレージ・ハンドラーおよび関連する jar を取得してください。

## ファイル・ベース Hive 表

Analytic Server は、組み込みまたはカスタムの Hive SerDe (serializer-deserializer) が利用可能な任意のファイル・ベース Hive 表をサポートします。

XML ファイルを処理するための Hive XML SerDe は Maven の Central Repository (<http://search.maven.org/#search%7Cga%7C1%7Chivexmlserde>) にあります。

## Apache Spark

HCatalog 入力データ・ソースで Spark (バージョン 1.5 以降) を使用する場合は、`spark.version=1.5.0` プロパティを手動で追加する必要があります。

1. Cloudera Manager を開き、Analytic Server の「**Advanced Configuration Snippet (Safety Valve) for analyticserver-conf/config.properties**」領域で以下のプロパティを追加するか、更新します。

```
spark.version=1.5.0
```

2. 構成を保存し、Cloudera Manager からすべての Analytic Server サービスを再始動します。

## Analytic Server で使用するポートの変更

デフォルトでは、Analytic Server はポート 9080 (HTTP 用) および 9443 (HTTPS 用) を使用します。ポートの設定を変更するには、以下のステップを実行します。

1. Cloudera Manager で、Analytic Server サービスの「構成 (Configuration)」タブに移動します。
2. 使用する HTTP ポートおよび HTTPS ポートを、それぞれ **http.port** パラメーターおよび **https.port** パラメーターに指定します。

注: これらのパラメーターを表示するには、「フィルター (Filters)」セクションで「ポートおよびアドレス (Ports and Addresses)」カテゴリを選択する必要がある場合があります。

3. 「変更の保存 (Save Changes)」をクリックします。
4. Analytic Server サービスを再始動します。

## 高可用性 Analytic Server

クラスター内の複数のノードに Analytic Server をサービスとして追加することにより、高可用性構成にすることができます。

1. Cloudera Manager で、Analytic Server サービスの「インスタンス (Instances)」タブに移動します。
2. 「役割インスタンスの追加 (Add Role Instances)」をクリックし、Analytic Server をサービスとして追加するホストを選択します。

## 小規模データ向けの JVM オプションの最適化

小規模な (M3R) ジョブの実行時にご使用のシステムを最適化するために、JVM プロパティを編集できます。

Cloudera Manager で、Analytic Server サービスの「構成 (Configuration)」タブの「**Jvm オプション (jvm.options)**」コントロールを確認します。以下のパラメーターを変更して、Analytic Server (Hadoop ではなく) をホストするサーバーで実行されるジョブのヒープ・サイズを設定します。これは小規模な (M3R) ジョブを実行する場合に重要です。システムを最適化するために、これらの値を調整する必要があります。

```
-Xms512M  
-Xmx2048M
```

---

## マイグレーション

Analytic Server では、既存の Analytic Server インストール済み環境から新規インストール済み環境へのデータおよび構成設定のマイグレーションが可能です。

## Analytic Server の新規バージョンへのアップグレード

Analytic Server 2.0/2.1 の既存インストール済み環境がある状態で新規バージョンを購入した場合、2.0/2.1 の構成設定を新規インストール済み環境にマイグレーションできます。

制約事項: 2.0 より前のバージョンがインストールされている場合、最初にその前のバージョンから 2.0/2.1 へのマイグレーションを行ってから、次にバージョン 2.0/2.1 から新規バージョンへのマイグレーションを行う必要があります。

制約事項: 2.0/2.1 インストール済み環境と新規インストール済み環境は、同じ Hadoop クラスター内には共存できません。2.0/2.1 インストール済み環境と同じ Hadoop クラスターを使用するように新規インストール済み環境を構成すると、2.0/2.1 インストール済み環境は動作しなくなります。

## 2.1 から新規バージョンへのマイグレーション・ステップ

- 28 ページの『Cloudera でのインストール』 の手順に従って、Analytic Server の新規インストールを実行します。
- 古いインストール済み環境から新しいインストール済み環境に Analytic のルート・ディレクトリーをコピーします。

- Analytic のルートの場所が不明な場合は、`hadoop -fs ls` を実行します。Analytic のルートのパスの形式は `/user/aeuser/analytic-root` です。ここで、`aeuser` は、Analytic ルートを所有しているユーザー ID です。
- Analytic ルートの所有権を `aeuser` から `as_user` に変更します。

```
hadoop dfs -chown -R {as_user:{group}} {path to 2.1 analytic-root}
```

注: マイグレーション後に既存の Analytic Server インストール済み環境を使用する予定の場合は、`analytic-root` ディレクトリーのコピーを HDFS 内に作成してから、そのディレクトリーのコピーで所有権を変更します。

- Analytic Server の新規インストール済み環境のホストに `as_user` としてログインします。`/user/as_user/analytic-root` ディレクトリーが存在する場合は削除します。
- 以下のコピー・スクリプトを実行します。

```
hadoop distcp hftp://{host of 2.1 namenode}:50070/{path to 2.1 analytic-root}
hdfs://{host of 3.0 namenode}/user/as_user/analytic-root
```

- Cloudera Manager で、Analytic Server サービスを停止します。
- 古いインストール済み環境から構成設定を収集します。
  - 新規インストール済み環境の `configcollector.zip` アーカイブを、古いインストール済み環境の `{AS_ROOT}¥tools` にコピーします。
  - コピーした `configcollector.zip` を解凍します。これにより、古いインストール済み環境内に新規の `configcollector` サブディレクトリーが作成されます。
  - `{AS_ROOT}¥tools¥configcollector` 内の **configcollector** スクリプトを実行して、古いインストール済み環境内の構成収集ツールを実行します。その結果生成された圧縮ファイル (ZIP) を、新規インストール済み環境をホストするサーバーにコピーします。
- migrationtool** スクリプトを実行し、構成収集ツールによって作成された圧縮ファイルのパスを引数として渡すことで、マイグレーション・ツールを実行します。次に例を示します。

```
migrationtool.sh /opt/ibm/spss/analyticserver/3.0/ASConfiguration_2.1.0.0.xxx.zip
```

- Zookeeper の状態をクリアします。Zookeeper の `bin` ディレクトリー (例えば、Cloudera 上の `/opt/cloudera/parcels/CDH-5.4....../lib/zookeeper/bin`) で、以下のコマンドを実行します。

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

7. Cloudera Manager で、Analytic Server サービスを開始します。

注: 既存の Analytic Server インストール済み環境で使用するよう R を構成している場合、新規 Analytic Server インストール済み環境で R を構成するステップに従う必要があります。

---

## Cloudera での Analytic Server のアンインストール

Cloudera は、Analytic Server のサービスおよびパーセルをアンインストールするために必要なステップの大部分を自動的に処理します。

Cloudera 環境から Analytic Server をクリーンアップするには、以下のステップを実行する必要があります。

1. Analytic Server サービスを停止してから削除します。
2. Analytic Server パーセルを「非アクティブ化 (**Deactivate**)」して「ホストから削除 (**Remove From Hosts**)」します。
3. HDFS の Analytic Server ユーザー・ディレクトリーを削除します。デフォルトの場所は `/user/as_user/analytic-root` です。
4. Analytic Server が使用するデータベース (スキーマ) を削除します。

---

## 第 4 章 MapR のインストールおよび構成

---

### MapR の概要

MapR は Apache Hadoop 向けの完全なディストリビューションであり、Hadoop エコシステムから 10 以上のプロジェクトがパッケージ化され、ビッグデータの処理に幅広く対応しています。

サーバー・クラスターの外部から MapR ファイル・システムにアクセスすることはできません。結果として、IBM SPSS Analytic Server は MapR クラスター・ノードにデプロイする必要があります。このデプロイメント・シナリオでは、Analytic Server は、MapR ファイル・システムへのアクセス権を持ち、かつ Analytic Server にデプロイするために YARN にジョブをサブミットする権限を持つユーザーが (<as\_user> として) 実行する必要があります。

---

### MapR での Analytic Server のインストール

以下のステップでは、IBM SPSS Analytic Server を手動で MapR クラスターにインストールするプロセスについて詳しく説明します。

1. root ユーザーまたは sudo ユーザーで Analytic Server インストーラー (spss\_as-3.0.1.0-mapr5.1-1x86-64\_en.bin) を実行します。インストールのプロンプトに従い、ライセンスに同意し、Analytic Server をオンラインでインストールするかオフラインでインストールするかを選択します。
  - a. Analytic Server をホストするサーバーに <http://ibm-open-platform.ibm.com> へのインターネット接続がある場合はオンライン・オプションを選択します。インストーラーが自動的に Analytic Server をインストールします。
  - b. Analytic Server をホストするサーバーに <http://ibm-open-platform.ibm.com> へのインターネット接続がない場合はオフライン・オプションを選択します。その URL にアクセス可能な別のサーバーでインストーラーを実行し、Analytic Server をオフラインでインストールすることを選択します。インストーラーが自動的に RPM パッケージをダウンロードします。
2. 以下のように、Analytic Server の RPM を検索して実行します。

```
rpm -ivh IBM-SPSS-AnalyticServer-3.0.1.0-1.x86_64.rpm
```

オンラインおよびオフラインのいずれのインストール・モードでも、Analytic Server は (<as\_installation\_path> として) /opt/ibm/spss/analyticserver/3.0 にインストールされます。

3. インストール・パス内のすべてのファイルを、以下のように、Analytic Server を実行するユーザーに変更します。

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

ユーザーを <as\_user> に切り替えます。以降のすべてのステップでは <as\_user> を使用します。

4. HTTP プロパティを構成します。http\_endpoint.xml という名前のファイルをパス <as\_installation\_path>/ae\_wlpserver/usr/servers/aeserver に作成し、そのファイルに以下の行を追加します。

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

<http\_port> と <https\_port> は、HTTP プロトコルと HTTPS プロトコルを介して Analytic Server が使用するポートです。使用可能な任意のポートでそれらを置き換えます。

5. ユーザーおよびグループを追加します。security\_cfg.xml という名前のファイルをパス <as\_installation\_path>/ae\_wlpserver/usr/servers/aeserver に作成し、そのファイルに以下の行を追加します。

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

デフォルトの状態では、XML ファイルには admin ユーザーのみが含まれています。他のユーザーおよびグループを手動で <basicRegistry> 設定に追加するか、その設定を ldapRegistry に変更する必要があります。

6. メタデータ・データベースをセットアップします。Analytic Server は、DB2 データベースおよび MySQL データベースをサポートします。
- a. データベース・ユーザーを構成します。MySQL データベースを使用する場合は、MySQL シェルで以下の SQL スクリプトを実行します。

```
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. パスワードを暗号化します。データベース・ユーザーのパスワードは、Analytic Server に渡す前に暗号化する必要があります。以下のコマンドを実行します。

```
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>
```

注: このコマンドを直接 Linux シェルで実行する場合は、文字 \* を ¥\* としてエスケープする必要があります。

このコマンドの出力は、The encrypted password is '<encrypted\_db\_password>' と表示されます。暗号化されたデータベース・パスワードを記録してください。

- c. <as\_installation\_path>/ae\_wlpserver/usr/servers/aeserver/configuration/config.properties ファイルが存在する場合はそのファイルを削除し、同じ名前で新規ファイルを作成します。DB2 データベースを使用する場合は、以下のプロパティを変更します。

```
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>
```

<db\_schema\_name> スキーマが存在しない場合、ユーザー <db\_username> には、スキーマを作成するための暗黙的な権限が付与されている必要があります。MySQL データベースを使用する場合は、以下のプロパティを変更します。

```
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. MySQL データベースを使用する場合は、MySQL JDBC ドライバーをインストールする必要があります。以下のコマンドを実行します。

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

- e. 以下のコマンドを実行して、必要な表を作成します。

```
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
```

<db\_type> は、どちらのデータベースを使用するかに応じて、db2 または mysql のいずれかです。

注: MYISAM エンジンと共に MySQL を使用する場合、2 番目のコマンドで以下のエラー・メッセージが報告されますが、無視しても問題ありません。

```
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. 以下のコマンドを実行して、cf ライブラリーをアンパックします。

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

8. private\_library.xml という名前のファイルをパス <as\_installation\_path>/ae\_wlpserver/usr/servers/aeserver に作成することにより JAAS ログイン・モジュールのクラスパスを構成し、そのファイルに以下の情報を入力します。

```
<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="/analyticsserver" 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="/analyticsserver/help"/>
</server>
```

注: 上記の例は、hadoop\_simple login モジュールを構成するためのものです。MapR で他のログイン・モジュールを使用する場合は、この構成を変更する必要があります。

9. ファイル ASModules.xml がパス <as\_installation\_path>/ae\_wlpserver/usr/servers/aeserver/configuration/ に存在することを確認します。そのファイルが存在しない場合は、ファイル ASModules.xml.template (同じパス内) を ASModules.xml に名前変更します。
10. 以下のプロパティーを <as\_installation\_path>/ae\_wlpserver/usr/servers/aeserver/configuration/config.properties ファイルに追加することにより、クラスター情報を構成します。

```
ae.cluster.zookeeper.connect.string=
ae.cluster.member.name=
ae.cluster.collective.name=mapr_5.1
```

ae.cluster.zookeeper.connect.string プロパティーは、コマンド区切りの zookeeper ノード・リストです。このプロパティーは、MapR が使用する zookeeper クラスターを共有できます。

ae.cluster.member.name は、Analytic Server をホストするノードのホスト名です。

11. <as\_installation\_path>/ae\_wlpserver/usr/servers/aeserver/server.env ファイルを開き、そのファイルに以下の行を追加します。

```
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
```

<as\_installation\_path> および <java\_home> を、実際のインストール・パスおよび Java ホーム・パスで置き換えます。

12. <as\_installation\_path>/ae\_wlpserver/usr/servers/aeserver/configuration/config.properties ファイルを開き、以下の行を追加することにより、Analytic ルートを編集します。

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

<analytic\_root> は、必須の Analytic Server リモート・ファイルをホストする MapR ファイル・システムのパスです。推奨パスは /user/<as\_user>/analytic-root です。

13. <as\_installation\_path>/ae\_wlpserver/usr/servers/aeserver/configuration/config.properties ファイルを開き、以下の行を追加することにより、管理者ユーザーを設定します。

```
admin.username=admin
```

この値は、Analytic Server 管理者ユーザー名にする必要があります、かつ security\_cfg.xml ファイルで構成されたいずれかのユーザーでなければなりません。

14. 以下の行を <as\_installation\_path>/bin/hdfsUpdate.sh ファイルの行 69 に追加することにより、Analytic Server の依存関係を MapR ファイル・システムにアップロードします。

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

以下のコマンドを実行して、<analytic\_root> を作成します。

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

<as\_user> には、<analytic\_root> 親ディレクトリーに対する書き込み権限が付与されている必要があります。

15. Analytic Server を開始してから停止します。
  - a. 以下のコマンドを実行し、Analytic Server を開始します。

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

- b. 以下のコマンドを実行し、Analytic Server を停止します。

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

---

## MapR の構成

インストール後、オプションで Analytic Server MapR の機能を構成し、管理することができます。

### データベース・プッシュバックの有効化

データベース・プッシュバックとは、データベースからデータを読み取り、直接そのデータで処理を行う操作です。

IBM SPSS Analytic Server は、以下のデータベースのプッシュバックをサポートします。

- DashDB
- DB2
- DB2 for Z
- Hive

- MySQL
- Netezza
- Oracle
- PostgreSQL
- Redshift
- SQL Server
- Sybase IQ
- Teradata

データベース・プッシュバックを有効にするには、以下のステップを使用します。

1. 適切な JDBC ドライバー JAR ファイルを `<as_installation_path>/jdbc` にコピーします。
2. `<as_installation_path>/ae_wlpserver/usr/servers/aeserver/private_library.xml` ファイルを開き、ID が `maprLib` のタグ・ライブラリーを見つけ、以下の行をタグに追加します。

```
<fileset dir="<as_installation_path>/jdbc" includes="*.jar"/>
```

3. 以下のコマンドを実行します。

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

4. Analytic Server を再始動します。

## Apache Hive の有効化

Apache Hive は、データの要約、照会、および分析を提供するために Hadoop 上に構築されたデータウェアハウス・インフラストラクチャーです。

注: Hive は、metastore として MySQL を使用するように構成する必要があります。IBM SPSS Analytic Server をホストするノードに存在する `hive-site.xml` ファイルは、Hive Metastore を実行するノードのファイルと同じでなければなりません。

MapR が正常にインストールされた後で Apache Hive のサポートを有効にするには、以下を行います。

1. 以下のコマンドを実行して、Hive および hcatalog の依存関係を MapR ファイル・システムにアップロードします。

```
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>` は、41 ページの『MapR での Analytic Server のインストール』で定義した Analytic のルート・パスです。

2. `<as_installation_path>/ae_wlpserver/usr/servers/aeserver/private_library.xml` ファイルを開き、ID が `maprLib` のタグ・ライブラリーを見つけ、以下の行をタグに追加します。

```
<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. 以下のコマンドを実行して、Hive および hcatalog の構成ファイル・リンクを作成します。

```
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. Hive の `auxlib` に追加の JAR ファイルがある場合は、以下の行を `private_library.xml` ファイルに追加します。

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

前述の行を追加した後に、以下のコマンドを実行します。

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

5. Analytic Server を再始動します。

## HTTP モードでの Hive の実行

デフォルトでは、Hive はバイナリー・モード (TCP モード) で実行されます。HTTP モードで Hive を実行するには、以下の Hive 構成プロパティ (特に `hive.server2.transport.mode` プロパティ) を更新する必要があります。

注: 各プロパティについて詳しくは、[Hive Configuration Properties](#) を参照してください。

表 4. HTTP モード用の Hive プロパティ

| プロパティ名   | デフォルト値                  | 説明  |
|--|-------------------------|---|
| <code>hive.server2.transport.mode</code>                 | バイナリー                   | サーバー・トランスポート・モード。値は <code>binary</code> または <code>http</code> のいずれかです。HTTP トランスポート・モードを有効にするには、 <code>http</code> に設定します。 |
| <code>hive.server2.thrift.http.port</code>               | 10001                   | HTTP モード時のポート番号。  |
| <code>hive.server2.thrift.http.path</code>               | <code>cliservice</code> | HTTP モード時の URL エンドポイントのパス・コンポーネント。  |
| <code>hive.server2.thrift.http.min.worker.threads</code> | 5                       | HTTP モード時のサーバー・プール内のワーカー・スレッドの最小数。  |
| <code>hive.server2.thrift.http.max.worker.threads</code> | 500                     | HTTP モード時のサーバー・プール内のワーカー・スレッドの最大数。  |

注: プロパティの更新後、Hive を再起動する必要があります。

## Apache HBase の有効化

Apache HBase は、Java で記述されたオープン・ソースの非リレーショナル分散データベースです。これは Apache Software Foundation の Apache Hadoop プロジェクトの一環として開発され、HDFS (Hadoop Distributed Filesystem) 上で実行されます。

MapR が正常にインストールされた後で Apache HBase のサポートを有効にするには、以下を行います。

1. HBase の依存関係を MapR ファイル・システムにアップロードします。以下のコマンドを実行します。

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

`<as_analytic_root>` は、41 ページの『MapR での Analytic Server のインストール』で定義した Analytic のルート・パスです。

2. `<as_installation_path>/ae_wlpserver/usr/servers/aeserver/private_library.xml` ファイルを開き、ID が `maprLib` のタグ・ライブラリーを見つけ、以下の行をタグに追加します。

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

3. 以下のコマンドを実行して、HBase および `hcatalog` の構成ファイル・リンクを作成します。

```
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. IBM SPSS Analytic Server を再始動します。

## Apache Spark の有効化

Apache Spark は、バッチ処理、リアルタイム処理、および高度な分析を、柔軟にメモリー内でデータ処理するオープン・スタンダードです。

MapR が正常にインストールされた後で Apache Spark のサポートを有効にするには、以下を行います。

1. spark-assembly-1.4.1-hadoop2.5.1-mapr-1501.jar ファイルを /opt/mapr/spark/spark-1.4.1/lib から <as\_installation\_path>/ae\_wlpserver/usr/servers/aeserver/modules/spark/ にコピーします。
2. Spark の依存関係を MapR ファイル・システムにアップロードします。以下のコマンドを実行します。

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

<as\_analytic\_root> は、41 ページの『MapR での Analytic Server のインストール』で定義した Analytic のルート・パスです。

3. <as\_installation\_path>/ae\_wlpserver/usr/servers/aeserver/private\_library.xml ファイルを開き、ID が maprLib のタグ・ライブラリーを見つけ、以下の行をタグに追加します。

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

4. 以下のコマンドを実行して、Spark の構成ファイル・リンクを作成します。

```
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. 以下の行を <as\_installation\_path>/ae\_wlpserver/usr/servers/aeserver/server.env ファイルに追加します。

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

6. 以下の行を <as\_installation\_path>/ae\_wlpserver/usr/servers/aeserver/configuration/config.properties ファイルに追加します。

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

7. IBM SPSS Analytic Server を再始動します。
8. PySpark 機能を有効にするには、以下の行を yarn-env.sh ファイルに追加し、ResourceManager および NodeManager を再始動します。

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

## 機能フラグの有効化

機能フラグを使用すると、特定のアプリケーションの機能を有効にしたり無効にしたりすることができます。

MapR が正常にインストールされた後で機能フラグのサポートを有効にするには、以下を行います。

1. 以下の行を <as\_installation\_path>/ae\_wlpserver/usr/servers/aeserver/configuration/config.properties ファイルに追加します。

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

2. IBM SPSS Analytic Server を再始動します。

## R の有効化

R は、統計的な計算およびグラフィックスのための言語であり、環境です。

MapR が正常にインストールされた後で R のサポートを有効にするには、以下を行います。

注: Node Manager および IBM SPSS Analytic Server をホストするすべてのクラスター・ノードでインストーラーを実行するには、事前に以下のパッケージをインストールする必要があります。

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

1. Node Manager および Analytic Server をホストするすべてのクラスター・ノードでインストーラー `spss_er-8.3.0.0-mapr5.1-lx86_64_en.bin` を実行します。このインストーラーを実行するユーザーには、R および Analytic Server インストール・パスに対する書き込み権限が付与されている必要があります。
2. ご使用条件に同意してインストール手順に従い、必要な情報を入力してください。Analytic Server がインストール・サーバーにインストールされている場合は、プロンプトが出されたら **Yes** を選択し、`<as_installation_path>` を入力します。Analytic Server がインストール・サーバーにインストールされていない場合は、プロンプトが出されたら **No** を選択します。
3. Analytic Server がインストールされている場合、Essentials for R は自動的に Analytic Server インストール・パスにインストールされます。

- Analytic Server がインストールされていない場合、Essentials for R はパス `<installer_path>/IBM_SPSS_ModelerEssentialsR/linux` にインストールされます。
- Analytic Server を後でインストールした場合は、以下のコマンドを使用して、Analytic Server がインストールされている Analytic Server 構成パスに Essentials for R をコピーします。

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

4. パス `<as_installation_path>/ae_wlpserver/usr/servers/aeserver/configuration` の `cf.zip` ファイルを削除し、以下のコマンドを使用して新規ファイルを生成します。

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

5. 以下のコマンドを実行します。

```
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. Analytic Server を再始動します。

## LZO の有効化

LZO は、無損失データ圧縮ライブラリーであり、圧縮率よりも速度を優先します。LZO のサポートを提供するには MapR を手動で構成する必要があります。

LZO のインストールと構成の手順はこのサイト (<https://github.com/twitter/hadoop-lzo>) に記載されています。

以下のステップでは、LZO ライブラリーを MapR にインポートするためのプロセスについて詳しく説明します。

1. `hadoop-lzo-<version>.jar` ファイルを Hadoop クラスパスにコピーします。推奨パスは `/opt/mapr/hadoop/hadoop-2.7.0/share/hadoop/mapreduce/lib` です。
2. ネイティブ・ファイルの `libgplcompression.so` および `liblzo2.so.2` を `/opt/mapr/hadoop/hadoop-2.7.0/lib/native` にコピーし、以下のプロパティーを `core-site.xml` ファイルに追加します。

```
<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. <as\_installation\_path>/ae\_wlpserver/usr/servers/aeserver/server.env ファイルを開き、<lzo\_native\_path> を LD\_LIBRARY\_PATH パラメーターに追加します。 <lzo\_native\_path> は、Hadoop-LZO ネイティブ・ライブラリーが含まれるフォルダーです。

```
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. IBM SPSS Analytic Server を再始動します。

## MapR 用の IBM SPSS Analytic Server クラスターのセットアップ

MapR のサポート用に IBM SPSS Analytic Server のクラスター環境をセットアップするには、以下のステップを使用します。

1. 以下の行を <as\_installation\_path>/ae\_wlpserver/usr/servers/aeserver/configuration/config.properties ファイルに追加します。

```
enable.resume=true
```
2. このインストール・パスを他のクラスター・ノードにコピーし、config.properties ファイルの ae.cluster.member.name プロパティを正しいホスト名に変更します。
3. すべてのクラスター・ノードを開始します。

---

## MapR のアンインストール

以下のステップでは、MapR をアンインストールするプロセスについて説明します。

1. IBM SPSS Analytic Server を停止します。
2. メタデータ・データベースを削除します。
  - a. 以下のコマンドを実行します。

```
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. 以下の SQL ステートメントを実行して、データベースを除去します。

```
drop database <db_name>
```

3. 次のように、RPM パッケージをアンインストールします。

```
rpm -e IBM-SPSS-AnalyticServer-3.0.1.0-1.x86_64
```

4. 次のように、インストール・パスを削除します。

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

5. 次のように、Analytic ルートを削除します。

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

6. 次のように、zookeeper のデータを削除します。

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

---

## MapR での IBM SPSS Analytic Server のマイグレーション

IBM SPSS Analytic Server は MapR でマイグレーションできます。

MapR で IBM SPSS Analytic Server 2.0 または 2.1 からバージョン 3.0.1 へのマイグレーションを行うには、以下のステップを使用します。

1. 41 ページの『MapR での Analytic Server のインストール』のインストール手順に従って、Analytic Server 3.0.1 を MapR クラスターにインストールします。
2. analytic ルートをコピーします。

注: Analytic ルートが変更されていない場合、このステップは省略可能です。

- Analytic Server バージョン 2.0/2.1 と 3.0.1 の両方の Analytic ルートが同じ MapR クラスタにある場合は、いずれかのデータ・ノードで以下のコマンドを実行します。

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

- Analytic Server バージョン 2.0/2.1 と 3.0.1 の Analytic ルートが別々の MapR クラスタにある場合、インストールされた WEBHDFS サービスまたは NFS サービスによって処理が決まります。WEBHDFS または NFS では、Analytic ルートのデータをコピーする必要があります。これは、クラスタの外部から直接 MapR ファイル・システムにアクセスできないためです。

- a. 以前の Analytic Server 2.0/2.1 クラスタに WEBHDFS サービスが含まれている場合、新規の Analytic Server 2.1 クラスタ・ノードのいずれかで以下のコマンドを実行します。

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

- b. 新規の Analytic Server 3.0.1 クラスタに WEBHDFS サービスが含まれている場合、以前の Analytic Server 2.0/2.1 クラスタ・ノードのいずれかで以下のコマンドを実行します。

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

- c. 以前のクラスタに NFS が含まれており、かつ NFS が新規の Analytic Server 3.0.1 クラスタ・ノードのいずれかにマウントされている場合、以前の Analytic Server 2.0/2.1 クラスタ・ノードのいずれかで以下のコマンドを実行します。

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

- d. 新規のクラスタに NFS が含まれており、かつ NFS が以前の Analytic Server 2.0/2.1 クラスタ・ノードのいずれかにマウントされている場合、新規の Analytic Server 3.0.1 クラスタ・ノードのいずれかで以下のコマンドを実行します。

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

異なる MapR クラスタ間でデータのマイグレーションを行う方法については、MapR の「Data Migration」サイトを参照してください。

3. 以下のコマンドを実行して、新規の Analytic ルートの所有者とアクセス権を変更します。

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

4. Analytic Server 3.0.1 を停止します。ただし、メタデータ・データベースは引き続き実行中であることを確認してください。

5. 以前の Analytic Server 2.0/2.1 クラスタ・インストール済み環境から構成設定を収集します。

- a. 新規の Analytic Server 3.0.1 クラスタ・インストール済み環境から以前の Analytic Server 2.0/2.1 クラスタ・インストール済み環境の <old\_as\_installation\_path>/tools に configcollector.zip アーカイブをコピーします。

- b. 以前の Analytic Server 2.0/2.1 クラスタ・インストール済み環境で configcollector.zip のコンテンツを解凍します。以前の Analytic Server 2.0/2.1 クラスタ・インストール済み環境に、新規の configcollector サブディレクトリが作成されます。

- c. <old\_as\_installation\_path>/tools/configcollector から configcollector スクリプトを実行して、以前の Analytic Server 2.0/2.1 クラスタ・インストール済み環境で構成収集ツールを実行します。結果として生成される圧縮 (ZIP) ファイルを新規の Analytic Server 3.0.1 クラスタ・インストール済み環境にコピーします。

6. migrationtool スクリプトを実行し、構成収集ツールによって作成された圧縮ファイルのパスを引数として渡すことで、新規の Analytic Server 3.0.1 クラスタでマイグレーション・ツールを実行します。以下に例を示します。

```
migrationtool.sh /opt/ibm/spss/analyticserver/3.0/ASConfiguration_2.1.0.0.xxx.zip
```

7. Analytic Server 3.0.1 を始動します。

---

## MapR のトラブルシューティング

このセクションでは、MapR のインストールおよび構成の一般的な問題とその解決方法を説明します。

### hdfsUpdate.sh スクリプトに関する問題

hdfsUpdate.sh スクリプトは、新規ファイルをアップロードする前に analytic-root 内のすべてのファイルを削除するため、このスクリプトの実行は 1 回のみにする必要があります。このスクリプトを 2 回以上実行した場合、データベース・プッシュバック、Hive、HBase、および Spark の依存関係を再アップロードする必要があります。必要な依存関係を再アップロードするには、以下のコマンドを実行します。

```
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
```



---

## 第 5 章 IBM SPSS Analytic Server で使用するための IBM SPSS Modeler の構成

SPSS Modeler を Analytic Server で使用できるようにするには、SPSS Modeler Server インストール済み環境に対する更新をいくつか行う必要があります。

1. SPSS Modeler Server を構成して、Analytic Server インストール済み環境と関連付けます。
  - a. メインサーバーのインストール・ディレクトリーの config サブディレクトリーにある options.cfg ファイルを編集して、以下の行を追加または編集します。

```
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

Analytic Server でセキュア通信が構成されている場合は Y を指定して、それ以外の場合には N を指定してください。

### as\_host

Analytic Server をホストするサーバーの IP アドレス。

### as\_port

Analytic Server が listen するポート (デフォルトは 8080)。

### as\_context\_root

Analytic Server コンテキスト・ルート (デフォルトは analyticserver)。

### as\_tenant

SPSS Modeler Server インストール済み環境がメンバーになっているテナント (デフォルトのテナントは ibm)。

### as\_prompt\_for\_password

SPSS Modeler Server が、Analytic Server で使用されているユーザーおよびパスワードの認証システムと同じ認証システムを使用して構成されている場合 (例えば、Kerberos 認証を使用している場合) は N を指定します。そうでない場合は、Y を指定します。

SPSS Modeler をバッチ・モードで実行している場合、clemb コマンドの引数として `-analytic_server_username {ASusername} -analytic_server_password {ASpassword}` を追加します。

### as\_kerberos\_auth\_mode

SPSS Modeler からの Kerberos SSO を有効にする場合は Y を指定します。

### as\_kerberos\_krb5\_conf

Analytic Server で使用する Kerberos 構成ファイルへのパスを指定します (例: `¥etc¥krb5.conf`)。

### as\_kerberos\_krb5\_spn

Analytic Server Kerberos SPN を指定します (例: HTTP/ashost.mydomain.com@MYDOMAIN.COM)。

- b. SPSS Modeler Server サービスを再開します。

SSL/TLS が有効になっている Analytic Server インストール済み環境に接続するには、SPSS Modeler Server とクライアントのインストール済み環境を構成するための追加のステップがいくつかあります。

- a. `http{s}://{HOST}:{PORT}/{CONTEXT-ROOT}/admin/{TENANT}` にナビゲートして、Analytic Server コンソールにログオンします。
- b. ブラウザーから認証ファイルをダウンロードして、ファイル・システムに保存します。
- c. 認証ファイルを SPSS Modeler Server と SPSS Modeler Client の両方のインストール済み環境の JRE に追加します。更新する場所は、SPSS Modeler インストール・パスの `/jre/lib/security/cacerts` サブディレクトリーで見つかります。
  - 1) `cacerts` ファイルが読み取り専用でないことを確認します。
  - 2) Modeler に付属の `keytool` プログラムを使用します。これは、SPSS Modeler インストール・パスの `/jre/bin/keytool` サブディレクトリーにあります。

次のコマンドを実行します。

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

<as-alias> は `cacerts` ファイルの別名であることに注意してください。 `cacerts` ファイルに固有のものである限り、任意の名前を使用できます。

以下にコマンドの例を示します。

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

- d. SPSS Modeler Server および SPSS Modeler Client を再起動します。
2. [オプション] Analytic Server データ・ソースを使用してストリーム内の R モデルのスコアリングを行う予定の場合、IBM SPSS Modeler - Essentials for R をインストールします。IBM SPSS Modeler - Essentials for R は、<https://www14.software.ibm.com/webapp/iwm/web/preLogin.do?source=swg-tspssp> からダウンロードできます。

---

## 第 6 章 トラブルシューティング

このセクションでは、インストールおよび構成の一般的な問題とその解決方法を説明します。

### 一般的な問題

インストールが警告付きで成功するが、「要求を完了できません。理由: 権限が拒否されました (Permission denied)」のエラーが発生してユーザーがデータ・ソースを作成できない

**distrib.fs.root** パラメーターを Analytic Server ユーザー (デフォルトでは `as_user`) がアクセス権限を持たないディレクトリーに設定すると、エラーが発生します。Analytic Server ユーザーが **distrib.fs.root** ディレクトリーに対して読み取り、書き込み、および実行を許可されるようにしてください。

**Analytic Server** のパフォーマンスが徐々に悪くなっている

Analytic Server のパフォーマンスが期待を満たさない場合は、Knox サービスのデプロイメント・パス `<KnoxServicePath>/data/ deployments` からすべての `*.war` ファイルを削除します。

例: `/usr/iop/4.1.0.0/knox/data/deployments`。

### 特定の Hadoop ディストリビューションに関する問題

**Analytic Server** サービスに対するリフレッシュ・アクションが **Hortonworks 2.4** で無効になっている  
Hortonworks 2.4 で Analytic Server ライブラリーを手動でリフレッシュするには、以下の手順を使用します。

1. Analytic Metastore を実行しているホストに Analytic Server ユーザー (デフォルトでは `as_user`) としてログオンします。

注: このホスト名は Ambari コンソールから確認できます。

2. `{AS_ROOT}/bin` ディレクトリーにある **refresh** スクリプトを実行します。例えば、次のようにします。

```
cd /opt/ibm/spss/analyticserver/3.0.1/bin
./refresh
```

3. Ambari コンソールで Analytic Server サービスを再始動します。

外部サイトからダウンロードしたパッケージが **Cloudera Manager** のハッシュ検査で不合格になる  
ハッシュ検査エラーがパーセル・リストに表示されます。この問題は、ダウンロード・プロセスを終了させてから、`cloudera-scm-server` サービスを介して Cloudera を再起動することで、解決できます。サービスの再起動後、エラーは発生しません。

### メタデータ・リポジトリーに関する問題

**add\_mysql\_user** スクリプトの実行時に **CREATE USER** 操作が失敗する

**add\_mysql\_user** スクリプトを実行する前に、まず、追加しようとしているユーザーを mysql データベースから手動で削除する必要があります。MySQL Workbench UI または MySQL コマンドによってユーザーを削除できます。以下に例を示します。

```
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'@'%';"
```

上記のコマンドで、削除するユーザー名で \$AEEDB\_USERNAME\_VALUE を置換し、データベースがインストールされているホスト名で \$METASTORE\_HOST を置換してください。

## Spark プロセス内で実行される IBM SPSS Modeler ストリームに関する問題

SPSS Modeler ストリームが Spark プロセス内で強制的に実行されたときに、完了できません。完了できない SPSS Modeler ストリームは、Analytic Server ソース・ノード (HDFS ファイル) を使用して作成されています。このノードは Sort ノードにリンクされたうえで、別の Analytic Server データ・ソースにエクスポートされるように設定されています。ストリームの実行後、Resource Manager のユーザー・インターフェースには、新規アプリケーションが実行中であると示されますが、ストリームが完了せず、実行状態のままです。Analytic Server ログ、YARN ログ、および Spark ログに、ストリームが完了できない理由を示すメッセージはありません。

この問題は、spark.executor.memory 設定を Analytic Server 構成のカスタム analytics.cfg ファイルに追加することで解決できます。メモリー値を 4 GB に設定すると、以前に完了できなかった SPSS Modeler ストリームを (単一ノード・クラスター環境で) 2 分以内に完了させることができます。

## 高可用性クラスター

依存関係の変更が原因で Analytic Server を追加ホストに追加できない

16 ページの『クライアント依存関係の更新』の手順に従って update\_clientdeps スクリプトを実行します。

### java.net.SocketTimeoutException: Read timed out

Liberty ND のタイムアウト環境変数を以下のように変更します。

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

ここで、<milliseconds> は JMX の読み取りタイムアウトに使用する秒数です。

### java.io.IOException: CWWKX7202E: コマンド ./server start のタイムアウト値 60 (秒) が経過しました

コントローラー・サーバーの server.xml に以下を追加します。

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

### java.lang.OutOfMemoryError: Java heap space

以下の行を HA クラスターのすべてのメンバーの jvm.options に追加します。

```
-Xms512M  
-Xmx2048M
```

「分析クラスター・サービスと Zookeeper との接続が予期せず切断されました。クラスターの整合性を保つため、この JVM を終了しています。」

この状態が発生する可能性のある原因の 1 つとして、Zookeeper に書き込まれるデータの量が多すぎることがあります。Zookeeper のログに以下のような例外がある場合:

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

あるいは、Analytic Server のログに以下のようなメッセージがある場合:

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

1. Ambari コンソールで、Zookeeper サービスの「Configs」タブにナビゲートし、以下の行を env-template に追加してから、Zookeeper サービスを再始動します。

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

2. Ambari コンソールで、Analytic Server サービスの「Configs」タブに移動し、以下の行を `Advanced analytics-jvm-options` に追加してから、Analytic クラスター・サービスを再始動します。

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

`jute.maxbuffer` 設定に対して指定する数値は、例外メッセージで示されている数値よりも大きくする必要があります。

### Zookeeper のトランザクション・データが管理不能になる

`zoo.cfg` の `autopurge.purgeInterval` パラメーターを 1 に設定して、Zookeeper トランザクション・ログの自動消去を有効にします。

### Analytic クラスター・サービスが Zookeeper との接続を失う

`zoo.cfg` の `tickTime`、`initLimit`、および `syncLimit` の各パラメーターを確認して変更します。以下に例を示します。

```
# 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
```

詳細については、Zookeeper の資料 (<https://zookeeper.apache.org/doc/r3.3.3/zookeeperAdmin.html>) を参照してください。

### Analytic Server ジョブが再開されない

Analytic Server ジョブが再開されない 2 つの一般的な状態があります。

1. クラスター・メンバーの障害が原因で Analytic Server ジョブが失敗した場合、通常そのジョブは他のクラスター・メンバー上で自動的に再開されます。ジョブが再開されない場合、高可用性クラスター内に少なくとも 4 つのクラスター・メンバーが存在することを確認してください。
2. 1 つのクラスター・メンバーを静止すると、そのサーバー上のすべての Analytic Server ジョブは、通常他のクラスター・メンバー上で再開されます。確実にジョブが再開されるようにするために、`-Dcom.spss.ae.remoteclient.failover.threshold=100` を設定して、リモート・モードを使用してください。

サーバーのシャットダウン時に **Analytic Server** サーバーが時折ハングする  
手作業でサーバーを `kill` してください。



---

## 特記事項

本書は米国 IBM が提供する製品およびサービスについて作成したものです。この資料は、IBM から他の言語でも提供されている可能性があります。ただし、これを入手するには、本製品または当該言語版製品を所有している必要がある場合があります。

本書に記載の製品、サービス、または機能が日本においては提供されていない場合があります。日本で利用可能な製品、サービス、および機能については、日本 IBM の営業担当員にお尋ねください。本書で IBM 製品、プログラム、またはサービスに言及していても、その IBM 製品、プログラム、またはサービスのみが使用可能であることを意味するものではありません。これらに代えて、IBM の知的所有権を侵害することのない、機能的に同等の製品、プログラム、またはサービスを使用することができます。ただし、IBM 以外の製品とプログラムの操作またはサービスの評価および検証は、お客様の責任で行っていただきます。

IBM は、本書に記載されている内容に関して特許権 (特許出願中のものを含む) を保有している場合があります。本書の提供は、お客様にこれらの特許権について実施権を許諾することを意味するものではありません。実施権についてのお問い合わせは、書面にて下記宛先にお送りください。

〒103-8510

東京都中央区日本橋箱崎町19番21号

日本アイ・ビー・エム株式会社

法務・知的財産

知的財産権ライセンス渉外

IBM およびその直接または間接の子会社は、本書を特定物として現存するままの状態を提供し、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任を負わないものとします。国または地域によっては、法律の強行規定により、保証責任の制限が禁じられる場合、強行規定の制限を受けるものとします。

この情報には、技術的に不適切な記述や誤植を含む場合があります。本書は定期的に見直され、必要な変更は本書の次版に組み込まれます。IBM は予告なしに、随時、この文書に記載されている製品またはプログラムに対して、改良または変更を行うことがあります。

本書において IBM 以外の Web サイトに言及している場合がありますが、便宜のため記載しただけであり、決してそれらの Web サイトを推奨するものではありません。それらの Web サイトにある資料は、この IBM 製品の資料の一部ではありません。それらの Web サイトは、お客様の責任でご使用ください。

IBM は、お客様が提供するいかなる情報も、お客様に対してなら義務も負うことのない、自ら適切と信ずる方法で、使用もしくは配布することができるものとします。

本プログラムのライセンス保持者で、(i) 独自に作成したプログラムとその他のプログラム (本プログラムを含む) との間での情報交換、および (ii) 交換された情報の相互利用を可能にすることを目的として、本プログラムに関する情報を必要とする方は、下記に連絡してください。

*IBM Director of Licensing*

*IBM Corporation*

*North Castle Drive, MD-NC119*

*Armonk, NY 10504-1785*

*US*

本プログラムに関する上記の情報は、適切な使用条件の下で使用することができますが、有償の場合もあります。

本書で説明されているライセンス・プログラムまたはその他のライセンス資料は、IBM 所定のプログラム契約の契約条項、IBM プログラムのご使用条件、またはそれと同等の条項に基づいて、IBM より提供されます。

記載されている性能データとお客様事例は、例として示す目的でのみ提供されています。実際の結果は特定の構成や稼働条件によって異なります。

IBM 以外の製品に関する情報は、その製品の供給者、出版物、もしくはその他の公に利用可能なソースから入手したものです。IBM は、それらの製品のテストは行っておりません。したがって、他社製品に関する実行性、互換性、またはその他の要求については確認できません。IBM 以外の製品の性能に関する質問は、それらの製品の供給者にお願いします。

IBM の将来の方向性および指針に関する記述は、予告なく変更または撤回される場合があります。これらは目標および目的を提示するものにすぎません。

表示されている IBM の価格は IBM が小売り価格として提示しているもので、現行価格であり、通知なしに変更されるものです。卸価格は、異なる場合があります。

本書はプランニング目的としてのみ記述されています。記述内容は製品が使用可能になる前に変更になる場合があります。

本書には、日常の業務処理で用いられるデータや報告書の例が含まれています。より具体性を与えるために、それらの例には、個人、企業、ブランド、あるいは製品などの名前が含まれている場合があります。これらの名前はすべて架空のものであり、名前や住所が類似する個人や企業が実在しているとしても、それは偶然にすぎません。

著作権使用許諾:

本書には、日常の業務処理で用いられるデータや報告書の例が含まれています。より具体性を与えるために、それらの例には、個人、企業、ブランド、あるいは製品などの名前が含まれている場合があります。これらの名前はすべて架空のものであり、名前や住所が類似する個人や企業が実在しているとしても、それは偶然にすぎません。

それぞれの複製物、サンプル・プログラムのいかなる部分、またはすべての派生的創作物にも、次のように、著作権表示を入れていただく必要があります。

© (お客様の会社名) (年). このコードの一部は、IBM Corp. のサンプル・プログラムから取られています。

© Copyright IBM Corp. \_年を入れる\_. All rights reserved.

---

## 商標

IBM、IBM ロゴおよび [ibm.com](http://ibm.com) は、世界の多くの国で登録された International Business Machines Corporation の商標です。他の製品名およびサービス名等は、それぞれ IBM または各社の商標である場合があります。現時点での IBM の商標リストについては、<http://www.ibm.com/legal/copytrade.shtml> をご覧ください。

Adobe、Adobe ロゴ、PostScript、PostScript ロゴは、Adobe Systems Incorporated の米国およびその他の国における登録商標または商標です。

IT Infrastructure Library は AXELOS Limited の登録商標です。

インテル、Intel、Intel ロゴ、Intel Inside、Intel Inside ロゴ、Centrino、Intel Centrino ロゴ、Celeron、Xeon、Intel SpeedStep、Itanium、および Pentium は、Intel Corporation または子会社の米国およびその他の国における商標または登録商標です。

Linux は、Linus Torvalds の米国およびその他の国における登録商標です。

Microsoft、Windows、Windows NT および Windows ロゴは、Microsoft Corporation の米国およびその他の国における商標です。

ITIL は AXELOS Limited の登録商標です。

UNIX は The Open Group の米国およびその他の国における登録商標です。

Cell Broadband Engine は、Sony Computer Entertainment, Inc. の米国およびその他の国における商標であり、同社の許諾を受けて使用しています。

Linear Tape-Open、LTO、LTO ロゴ、Ultrium および Ultrium ロゴは、HP、IBM Corp. および Quantum の米国およびその他の国における商標です。







Printed in Japan

**日本アイ・ビー・エム株式会社**

〒103-8510 東京都中央区日本橋箱崎町19-21