Hortonworks DataFlow

Installing an HDF Cluster for IBM Power Systems

(December 22, 2017)

docs.hortonworks.com

Hortonworks DataFlow: Installing an HDF Cluster for IBM Power Systems

Table of Contents

1. Installing Ambari

To install Ambari server on a single host in your cluster, complete the following steps:

1. Download the Ambari Repository [1]
2. Set Up the Ambari Server [2]
3. Start the Ambari Server [4]

1.1. Download the Ambari Repository

Ambari 2.6 Repositories

OS	Format	URL
RedHat 7	Base URL	http://public-repo-1.hortonworks.com/ambari/centos7-ppc/2.x/updates/2.6.2.2
	Repo File	https://public-repo-1.hortonworks.com/ambari/centos7-ppc/2.x/updates/2.6.2.2/ambari.repo
	Tarball md5 | asc	http://public-repo-1.hortonworks.com/ambari/centos7-ppc/2.x/updates/2.6.2.2/ambari-2.6.2.2-centos7-ppc.tar.gz

Use a command line editor to perform each instruction.

1.1.1. RHEL 7

On a server host that has Internet access, use a command line editor to perform the following steps:

1. Log in to your host as `root`.
2. Download the Ambari repository file to a directory on your installation host.

```
wget -nv http://public-repo-1.hortonworks.com/ambari/centos7-ppc/2.x/
updates/2.6.2.2/ambari.repo -O /etc/yum.repos.d/ambari.repo
```

Important

Do not modify the `ambari.repo` file name. This file is expected to be available on the Ambari Server host during Agent registration.

3. Confirm that the repository is configured by checking the repo list.

```
yum repolist
```

You should see values similar to those in the following table for the Ambari repository listing.

The values in this table are examples. Your version values depend on your installation.

repo id	repo name	status
AMBARI.2.6.2.2-1.x	Ambari 2.x	12
base	CentOS-7 - Base	6,518
extras	CentOS-7 - Extras	15

repo id	repo name	status
updates	CentOS-7 - Updates	209

4. Install the Ambari bits.

```
yum install ambari-server
```

5. Enter **y** when prompted to confirm transaction and dependency checks.

   A successful installation displays output similar to the following:

```
Installing : postgresql-libs-9.2.15-1.e17_2.ppc64le         1/4
Installing : postgresql-9.2.15-1.e17_2.ppc64le              2/4
Installing : postgresql-server-9.2.15-1.e17_2.ppc64le       3/4
Installing : ambari-server-2.6.2.2-1.ppc64le              4/4
Verifying  : postgresql-server-9.2.15-1.e17_2.ppc64le       1/4
Verifying  : ambari-server-2.6.2.2-1.ppc64le              2/4
Verifying  : postgresql-9.2.15-1.e17_2.ppc64le              3/4
Verifying  : postgresql-libs-9.2.15-1.e17_2.ppc64le         4/4

Installed:
  ambari-server.ppc64le  0:2.6.2.2-1

Dependency Installed:
 postgresql.ppc64le 0:9.2.15-1.e17_2

 Complete!
```

Note

Accept the warning about trusting the Hortonworks GPG Key. That key will be automatically downloaded and used to validate packages from Hortonworks. You will see the following message:

```
Importing GPG key 0x07513CAD: Userid: "Jenkins (HDP Builds)
<jenkin@hortonworks.com>" From :
http://s3.amazonaws.com/dev.hortonworks.com/ambari/centos7-ppc/
RPM-GPG-KEY/RPM-GPG-KEY-Jenkins
```

Note

When deploying HDP on a cluster having limited or no Internet access, you should provide access to the bits using an alternative method.

Ambari Server by default uses an embedded PostgreSQL database. When you install the Ambari Server, the PostgreSQL packages and dependencies must be available for install. These packages are typically available as part of your Operating System repositories. Please confirm you have the appropriate repositories available for the postgresql-server packages.

1.2. Set Up the Ambari Server

Before starting the Ambari Server, you **must** set up the Ambari Server. Setup configures Ambari to talk to the Ambari database, installs the JDK and allows you to customize the user account the Ambari Server daemon will run as. The

```
ambari-server setup
```

command manages the setup process.

Prerequisites

To use MySQL as the Ambari database, you must set up the mysql connector, create a user and grant user permissions before running ambari-setup.

Using Ambari with MySQL/MariaDB

Steps

1. To start the setup process, run the following command on the Ambari server host. You may also append setup options to the command.

   ```
   ambari-server setup -j $JAVA_HOME
   ```

2. Respond to the setup prompt:

   #### Setup Options

   The following table describes options frequently used for Ambari Server setup.

Option	Description
-j (or –java-home)	You must manually install the JDK on all hosts and specify the Java Home path during Ambari Server setup. If you plan to use Kerberos, you must also install the JCE on all hosts. This path must be valid on all hosts. For example: `ambari-server setup –j /usr/java/default`
–jdbc-driver	Should be the path to the JDBC driver JAR file. Use this option to specify the location of the JDBC driver JAR and to make that JAR available to Ambari Server for distribution to cluster hosts during configuration. Use this option with the –jdbc-db option to specify the database type.
–jdbc-db	Specifies the database type. Valid values are: [postgres | mysql] Use this option with the –jdbc-driver option to specify the location of the JDBC driver JAR file.
-s (or –silent)	Setup runs silently. Accepts all the default prompt values*. If you select the silent setup option, you must also include the -j (or –java-home) option.
-v (or –verbose)	Prints verbose info and warning messages to the console during Setup.
-g (or –debug)	Prints debug info to the console during Setup.

   #### Important

   *If you choose the silent setup option and do not override the JDK selection, Oracle JDK installs and you agree to the Oracle Binary Code License agreement.

   Oracle JDK is NOT supported for IBM-PPC.

3. If you have *not* temporarily disabled SELinux, you may get a warning. Accept the default **y**, and continue.

4. By default, Ambari Server runs under `root`. Accept the default **n** at the `Customize user account for ambari-server daemon` prompt, to proceed as `root`. If you want to create a different user to run the Ambari Server, or to assign a previously

created user, select **y** at the `Customize user account for ambari-server daemon` prompt, then provide a user name.

5. If you have not temporarily disabled `iptables` you may get a warning. Enter **y** to continue.

6. Select **`Custom JDK`**, you must manually install the JDK on all hosts and specify the Java Home path.

   ### Note

   Open JDK v1.8 is the only supported JDK.

7. Review the GPL license agreement when prompted. To explicitly enable Ambari to download and install LZO data compression libraries, you must answer **y**. If you enter **n**, Ambari will not automatically install LZO on any new host in the cluster. In this case, you must ensure LZO is installed and configured appropriately. Without LZO being installed and configured, data compressed with LZO will not be readable. If you do not want Ambari to automatically download and install LZO, you must confirm your choice to proceed.

8. Select **y** at `Enter advanced database configuration`.

9. In `Advanced database configuration`, enter **`Option [3] MySQL/MariaDB`**, then enter the credentials you defined for user name, password and database name.

10. At `Proceed with configuring remote database connection properties [y/n]` choose **y**.

11. Setup completes.

    ### Note

    If your host accesses the Internet through a proxy server, you must configure Ambari Server to use this proxy server.

Next Steps

Start the Ambari Server [4]

More Information

???TITLE??? [3]

Using Ambari with MySQL/MariaDB

Configuring Ambari for Non-Root

How to Set Up an Internet Proxy Server for Ambari

Configuring LZO Compression

1.3. Start the Ambari Server

- Run the following command on the Ambari Server host:

```
ambari-server start
```

- To check the Ambari Server processes:

```
ambari-server status
```

- To stop the Ambari Server:

```
ambari-server stop
```

Note

If you plan to use an existing database instance for Hive or for Oozie, you must prepare those databases **before** installing your Hadoop cluster.

On Ambari Server start, Ambari runs a database consistency check looking for issues. If any issues are found, Ambari Server **start will abort** and a message will be printed to console "DB configs consistency check failed." More details will be written to the following log file:

```
/var/log/ambari-server/ambari-server-check-database.log
```

You can force Ambari Server to start by skipping this check with the following option:

```
ambari-server start --skip-database-check
```

If you have database issues, by choosing to skip this check, **do not make any changes to your cluster topology or perform a cluster upgrade until you correct the database consistency issues**. Please contact Hortonworks Support and provide the `ambari-server-check-database.log` output for assistance.

Next Steps

Install, Configure and Deploy a Cluster

More Information

Using New and Existing Databases - Hive

Using Existing Databases-Oozie

2. Installing Databases

Schema Registry, SAM, Druid, and Superset require a relational data store to store metadata. You can use either MySQL, Postgres, or Oracle for this. This chapter describes how to install either MySQL, Postgres, and Oracle and how create a databases for SAM and Registry. If you are installing on an existing HDP cluster by using Superset, you can skip the installation instructions, because MySQL was installed with Druid. In this case, configure the databases.

Note

You should install either Postgres, Oracle or MySQL; both are not necessary. It is recommended that you use MySQL.

Warning

If you are installing Postgres, you must install Postgres 9.5 or later for SAM and Schema Registry. Ambari does not install Postgres 9.5, so you must perform a manual Postgres installation.

Installing and Configuring MySQL

- Installing MySQL [6]
- Configuring SAM and Schema Registry Metadata Stores in MySQL [7]
- Configuring Druid and Superset Metadata Stores in MySQL [8]

Installing and Configuring Postgres

- Install Postgres [8]
- Configure Postgres to Allow Remote Connections [9]
- Configure SAM and Schema Registry Metadata Stores in Postgres [10]
- Configure Druid and Superset Metadata Stores in Postgres [10]

Using an Oracle Database

- Section 2.8, “Specifying an Oracle Database to Use with SAM and Schema Registry” [11]
- Section 2.9, “Switching to an Oracle Database After Installation” [11]

2.1. Installing MySQL

About This Task

You can install MySQL 5.5 or later.

Before You Begin

On the Ambari host, install the JDBC driver for MySQL, and then add it to Ambari:

```
yum install mysql-connector-java* \
sudo ambari-server setup --jdbc-db=mysql \
--jdbc-driver=/usr/share/java/mysql-connector-java.jar
```

Steps

1. Log in to the node on which you want to install the MySQL metastore to use for SAM, Schema Registry, and Druid.

2. Install MySQL and the MySQL community server, and start the MySQL service:

   ```
   yum localinstall \
   https://dev.mysql.com/get/mysql57-community-release-el7-8.noarch.rpm

   yum install mysql-community-server

   systemctl start mysqld.service
   ```

3. Obtain a randomly generated MySQL root password:

   ```
   grep 'A temporary password is generated for root@localhost' \
   /var/log/mysqld.log |tail -1
   ```

4. Reset the MySQL root password. Enter the following command. You are prompted for the password you obtained in the previous step. MySQL then asks you to change the password.

   ```
   /usr/bin/mysql_secure_installation
   ```

2.2. Configuring SAM and Schema Registry Metadata Stores in MySQL

Steps

1. Launch the MySQL monitor:

   ```
   mysql -u root -p
   ```

2. Create the database for the Registry and SAM metastore:

   ```
   create database registry;
   create database streamline;
   ```

3. Create Schema Registry and SAM user accounts, replacing the last string with your password:

   ```
   CREATE USER 'registry'@'%' IDENTIFIED BY 'R12$%34qw';
   CREATE USER 'streamline'@'%' IDENTIFIED BY 'R12$%34qw';
   ```

4. Assign privileges to the user account:

   ```
   GRANT ALL PRIVILEGES ON registry.* TO 'registry'@'%' WITH GRANT OPTION ;
   GRANT ALL PRIVILEGES ON streamline.* TO 'streamline'@'%' WITH GRANT OPTION ;
   ```

5. Commit the operation:

```
commit;
```

2.3. Configuring Druid and Superset Metadata Stores in MySQL

About This Task

Druid and Superset require a relational data store to store metadata. To use MySQL for this, install MySQL and create a database for the Druid metastore.

Steps

1. Launch the MySQL monitor:

   ```
   mysql -u root -p
   ```

2. Create the database for the Druid and Superset metastore:

   ```
   CREATE DATABASE druid DEFAULT CHARACTER SET utf8;
   CREATE DATABASE superset DEFAULT CHARACTER SET utf8;
   ```

3. Create `druid` and `superset` user accounts, replacing the last string with your password:

   ```
   CREATE USER 'druid'@'%' IDENTIFIED BY '9oNio)ex1ndL';
   CREATE USER 'superset'@'%' IDENTIFIED BY '9oNio)ex1ndL';
   ```

4. Assign privileges to the `druid` account:

   ```
   GRANT ALL PRIVILEGES ON *.* TO 'druid'@'%' WITH GRANT OPTION;
   GRANT ALL PRIVILEGES ON *.* TO 'superset'@'%' WITH GRANT OPTION;
   ```

5. Commit the operation:

   ```
   commit;
   ```

2.4. Install Postgres

Before You Begin

If you have already installed a MySQL database, you may skip these steps.

Warning

You must install Postgres 9.5 or later for SAM and Schema Registry. Ambari does not install Postgres 9.5, so you must perform a manual Postgres installation.

Steps

1. Install RPM according to the requirements of your operating system:

   ```
   yum install https://yum.postgresql.org/9.6/redhat/rhel-7-x86_64/pgdg-
   redhat96-9.6-3.noarch.rpm
   ```

2. Install 9.5+ Postgres database:

   ```
   yum install postgresql96-server postgresql96-contrib postgresql96
   ```

3. Initialize the database:

   - For CentOS 7, use the following syntax:

     ```
     /usr/pgsql-9.6/bin/postgresql96-setup initdb
     ```

   - For CentOS 6, use the following syntax:

     ```
     sudo service postgresql initdb
     ```

4. Start Postgres.

   For example, if you are using CentOS 7, use the following syntax:

   ```
   systemctl enable postgresql-9.6.service
   systemctl start postgresql-9.6.service
   ```

5. Verify that you can log in:

   ```
   sudo su postgres
   psql
   ```

2.5. Configure Postgres to Allow Remote Connections

About This Task

It is critical that you configure Postgres to allow remote connections before you deploy a cluster. If you do not perform these steps in advance of installing your cluster, the installation fails.

Steps

1. Open /var/lib/pgsql/9.6/data/pg_hba.conf and update to the following

   ```

   # "local" is for Unix domain socket connections only
   local all all trust


   # IPv4 local connections:
   host all all 0.0.0.0/0 trust


   # IPv6 local connections:
   host all all ::/0 trust
   ```

2. Open /var/lib//pgsql/9.6/data/postgresql.conf and update to the following:

   ```
   listen_addresses = '*'
   ```

3. Restart Postgres:

4.
   ```
   systemctl stop postgresql-9.6.service
   systemctl start postgresql-9.6.service
   ```

2.6. Configure SAM and Schema Registry Metadata Stores in Postgres

About This Task

If you have already installed MySQL and configured SAM and Schema Registry metadata stores using MySQL, you do not need to configure additional metadata stores in Postgres.

Steps

1. Log in to Postgres:

   ```
   sudo su postgres
   psql
   ```

2. Create a database called `registry` with the password `registry`:

   ```
   create database registry;
   CREATE USER registry WITH PASSWORD 'registry';
   GRANT ALL PRIVILEGES ON DATABASE "registry" to registry;
   ```

3. Create a database called `streamline` with the password `streamline`:

   ```
   create database streamline;
   CREATE USER streamline WITH PASSWORD 'streamline';
   GRANT ALL PRIVILEGES ON DATABASE "streamline" to streamline;
   ```

2.7. Configure Druid and Superset Metadata Stores in Postgres

About This Task

Druid and Superset require a relational data store to store metadata. To use Postgres for this, install Postgres and create a database for the Druid metastore. If you have already created a data store using MySQL, you do not need to configure additional metadata stores in Postgres.

Steps

1. Log in to Postgres:

   ```
   sudo su postgres
   psql
   ```

2. Create a database, user, and password, each called `druid`, and assign database privileges to the user `druid`:

   ```
   create database druid;
   CREATE USER druid WITH PASSWORD 'druid';
   GRANT ALL PRIVILEGES ON DATABASE  "druid" to druid;
   ```

3. Create a database, user, and password, each called `superset`, and assign database privileges to the user `superset`:

    ```
    create database superset;
    CREATE USER superset WITH PASSWORD 'superset';
    GRANT ALL PRIVILEGES ON DATABASE "superset" to superset;
    ```

2.8. Specifying an Oracle Database to Use with SAM and Schema Registry

About This Task

You may use an Oracle database with SAM and Schema Registry. Oracle databases 12c and 11g Release 2 are supported

Prerequisites

You have an Oracle database installed and configured.

Steps

1. Register the Oracle JDBC driver jar.

    ```
    sudo ambari-server setup --jdbc-db=oracle --jdbc-driver=/usr/share/java/
    ojdbc.jar
    ```

2. From the SAM an Schema Registry configuration screen, select Oracle as the database type and provide the necessary Oracle Server JDBC credentials and connection string.

2.9. Switching to an Oracle Database After Installation

About This Task

If you want to use an Oracle database with SAM or Schema Registry after you have performed your initial HDF installation or upgrade, you can switch to an Oracle database. Oracle databases 12c and 11g Release 2 are supported

Prerequisites

You have an Oracle database installed and configured.

Steps

1. Log into Ambari Server and shut down SAM or Schema Registry.

2. From the configuration screen, select Oracle as the database type and provide Oracle credentials, the JDBC connection string and click **Save**.

3. From the command line where Ambari Server is running, register the Oracle JDBC driver jar:

```
sudo ambari-server setup --jdbc-db=oracle --jdbc-driver=/usr/share/java/
ojdbc.jar
```

4. From the host where SAM or Schema Registry are installed, copy the JDBC jar to the following location, depending on which component you are updating.

   ```
    cp ojdbc6.jar /usr/hdf/current/registry/bootstrap/lib/.
    cp ojdbc6.jar /usr/hdf/current/streamline/bootstrap/lib/.
   ```

5. From the host where SAM or Schema Registry are installed, run the following command to create the required schemas for SAM or Schema Registry.

   ```
   export JAVA_HOME=/usr/jdk64/jdk1.8.0_112 ; source /usr/hdf/current/
   streamline/conf/streamline-env.sh ; /usr/hdf/current/streamline/bootstrap/
   bootstrap-storage.sh create

   export JAVA_HOME=/usr/jdk64/jdk1.8.0_112 ; source /usr/hdf/current/registry/
   conf/registry-env.sh ; /usr/hdf/current/registry/bootstrap/bootstrap-
   storage.sh create
   ```

   Note

   You only this command run once, from a single host, to prepare the database.

6. Confirm that new tables are created in the Oracle database.

7. From Ambari, restart SAM or Schema Registry.

8. If you are specifying an Oracle database for SAM, run the following command after you have restarted SAM.

   ```
   export JAVA_HOME=/usr/jdk64/jdk1.8.0_112 ; source /usr/hdf/current/
   streamline/conf/streamline-env.sh ; /usr/hdf/current/streamline/bootstrap/
   bootstrap.sh
   ```

9. Confirm that Sam or Schema Registry are available and turn off maintenance mode.

3. Installing the HDF Management Pack

About This Task

A management pack (mpack) bundles service definitions, stack definitions, and stack add-on service definitions so they do not need to be included with the Ambari core functionality and can be updated in between major releases.

Warning

If you are installing an HDF cluster only, begin the installation with a fresh Ambari instance. Do not install the HDF management pack on a system where HDP is already installed.

Steps

1. Back up your Ambari resources folder:

   ```
   cp -r /var/lib/ambari-server/resources /var/lib/ambari-server/resources.
   backup
   ```

2. Download the Hortonworks HDF management pack. You can find the download location for your operating system in the *HDF Release Notes*.

3. Copy the bundle to `/tmp` on the node where you installed Ambari.

4. Install the management pack:

   ```
   ambari-server install-mpack \
   --mpack=/tmp/hdf-ambari-mpack-<version>.tar.gz \
   --purge
   --verbose
   ```

5. Restart the Ambari server:

   ```
   ambari-server restart
   ```

More Information

HDF Release Notes

4. Install an HDF Cluster Using Ambari

About This Task

After you start the Ambari service, you can open Ambari web in a browser and launch the Install wizard to prepare for installing an HDF cluster.

Steps

1. Navigate to http://<your.ambari.server>:8080, where <your.ambari.server> is the name of your Ambari server host.

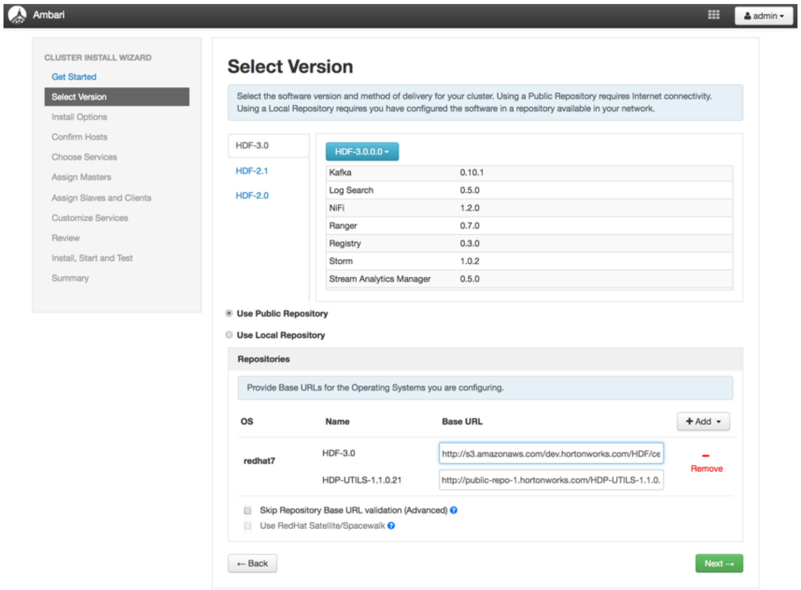
2. Log in to the Ambari server by using the default user name and password: admin and admin. You can change these credentials later.

3. In the **Ambari Welcome** page, choose **Launch Install Wizard**.

4. In the **Get Started** step, specify a name for your cluster.

5. In the **Select Version** page, remove all repositories except the one appropriate for your operating system. Change the Base URL for HDF to the base URL appropriate for your operating system. Find the HDF Base URLs in the HDF Release Notes.

   The wizard page should look similar to the following example:



Select Version

Select the software version and method of delivery for your cluster. Using a Public Repository requires Internet connectivity. Using a Local Repository requires you have configured the software in a repository available in your network.

Kafka	0.10.1
Log Search	0.5.0
NiFi	1.2.0
Ranger	0.7.0
Registry	0.3.0
Storm	1.0.2
Stream Analytics Manager	0.5.0

● Use Public Repository

○ Use Local Repository

Repositories

Provide Base URLs for the Operating Systems you are configuring.

OS	Name	Base URL
redhat7	HDF-3.0	http://s3.amazonaws.com/dev.hortonworks.com/HDF/ce
	HDP-UTILS-1.1.0.21	http://public-repo-1.hortonworks.com/HDP-UTILS-1.1.0.

□ Skip Repository Base URL validation (Advanced)

□ Use RedHat Satellite/Spacewalk

5. Configure HDF Components

You can customize your Hortonworks DataFlow (HDF) component configurations either during or after installation. During installation, you customize HDF component configurations in the **Customize Services** page of the installation wizard. After installation, you can navigate to Services > Configs in the Ambari dashboard.

5.1. Configure Schema Registry

About This Task

The main Schema Registry configuration task you have is to establish a connection between Schema Registry and the database you want to use as the metadata store.

Steps

1. In the **Customize Services** step, navigate to the **REGISTRY CONFIG** section of the **Registry** tab.

2. Select **Jar Storage Type** and then the storage type that you want to use.

   If you plan to enable HA for Schema Registry on this cluster, you must select **HDFS**.

3. If you selected **HDFS** as the **Jar Storage Type**, configure **Jar Storage HDFS URL**. This specifies the HDFS location where you want the jars to be stored. For example, `hdfs://<<NN_HOST:8020:/hdfs/registry`.

4. Configure **jar.storage** to the directory in which you want to store `.jar` files for serializers and deserializers.

5. Configure the **REGISTRY STORAGE** configurations based on the database you created to use as the Schema Registry metadata store.

6. Ensure that the registry storage connector URL has the fully qualified name of the host on which the database was installed and the connector url and default port for the database selected.

Example

```
MYSQL example:

jdbc:mysql://FQDN_MYSQL:3306/registry


Postgres Example:

jdbc:postgresql://FQDN_POSTGRES:5432/registry
```

5.2. Configure SAM

About This Task

When you configure Hortonworks Streaming Analytics Manager (SAM), you must provide information about the metadata store database, configure a connection with Schema Registry, and establish the URL for the Druid Supersets.

Steps

1. In the **Customize Services** step, navigate to the **STREAMLINE CONFIG** section of the **Streaming Analytics Manager** tab.

2. Select **Jar Storage Type**. If you plan to enable HA for SAM on this cluster, you must select **HDFS**.

3. If you selected **HDFS** as the **Jar Storage Type**, configure **Jar Storage HDFS URL**. This specifies the HDFS location where you want the jars to be stored. For example, `hdfs://<<NN_HOST:8020:/hdfs/registry`.

4. Configure **jar.storage** to the directory on which you want to store `.jar` files for custom processors.

5. Set the **streamline.dashboard.url** to the Superset URL which you can access using **Quick Links** for Druid.

6. Configure **registry.url** to the REST API Endpoint URL for the Registry.

   The format should be `http://$FQDN_REGISTRY_HOST:$REGISTRY_PORT/api/v1`, where

   - $FQDN_REGISTRY_HOST specifies the host on which you are running Schema Registry and

   - $REGISTRY_PORT specifies the Schema Registry port number, as in the following example:

     http://FQDN_REGISTRY_HOST:7788/api/v1

You can find the Schema Registry port in the **REGISTRY_CONFIG** section of the **Registry** tab.

7. Configure the **STREAMLINE STORAGE** configurations based on the database you created to use as a SAM metadata store.

8. Ensure that the registry storage connector URL has the fully qualified name of the host on which the database was installed and the connector url and default port for the database selected.

Example

```
MYSQL example:

jdbc:mysql://FQDN_MYSQL:3306/streamline


Postgres Example:

jdbc:postgresql://FQDN_POSTGRES:5432/streamline
```

5.3. Configure NiFi

About This Task

You use the **NiFi** tab in the **Customize Services** step to configure Apache NiFi. Generally, you can accept the defaults during initial installation. However, there are some settings that you must set before proceeding.

Steps

1. From **Advanced-nifi-ambari-config**, specify the **Encrypt Configuration Master Key Passwords**.

   This password is used when you generate the master key for sensitive properties encryption in the NiFi properties file when it is written to disk. It must contain at least 12 characters.

2. From **Advanced-nifi-ambari-config**, provide the **Sensitive property values encryption password**.

   This is the password used when you encrypt any sensitive property values that are configured in processors. For enhanced security, it should contain at least 10 characters.

5.4. Configure NiFi for Atlas Integration

About This Task

You can integrate NiFi with Apache Atlas to take advantage of robust dataset and application lineage support. You do this by configuring the NiFi ReportLineageToAtlas Reporting Task once you have NiFi configured and running.

Before You Begin

If NiFi is installed on an HDP cluster, you must be running HDP 2.6.4. If NiFi is installed on an HDF cluster managed by a separate Ambari instance, you must be running HDP 2.6.1 or later, and Apache Atlas 0.8.0 or later.

Steps

1. From the Global Menu located in NiFi’s upper right corner, select **Controller Services** and click the **Reporting Tasks** tab.

2. Click the **Add** (+) icon to launch the **Add Reporting Task** dialog.

3. Select ReportLineageToAtlas and click **Add**.

4. Click the **Edit** icon to launch the **Configure Reporting Task** dialog. The following Properties are required:

   - Atlas URLs – a comma-separated list of Atlas Server URLs. Once you have started reporting, you cannot modify an existing Reporting Task to add a new Atlas Server. When you need to add a new Atlas Server, you must create a new reporting task.

   - Atlas Authentication Method – Specifies how to authenticate the Reporting Task to the Atlas Server. Basic authentication is the default.

   - NiFi URL for Atlas – Specifies the NiFi cluster URL

   - NiFi Lineage Strategy – Specifies the level of granularity for your NiFi dataflow reporting to Atlas. Once you have started reporting, you should not switch between simple and complete lineage reporting strategies.

   - Provenance Record Start Position – Specifies where in the Provenance Events stream the Reporting Task should start.

   - Provenance Record Batch Size – Specifies how many records you want to send in a single batch

   - Create Atlas Configuration File – If enabled, the `atlas-application-properties` file and the `Atlas Configuration Directory` are automatically created when the Reporting Task starts.

   - Kafka Security Protocol – Specifies the protocol used to communicate with Kafka brokers to send Atlas hook notification messages. This value should match Kafka's security.protocol property value.

Result

Once you have ReportLineageToAtlas up and running, you may view dataset level lineage graphs in the Atlas UI.

Note

The default time interval for the Reporting Task to start sending data to an Atlas Server is 5 minutes so do not expect to see immediate lineage graphs.

You can change the default time interval in the Reporting Task property configuration.

More Information

For complete information, see the help included with the Reporting Task.

5.5. Configure Kafka

About This Task

You can configure Apache Kafka from the **Kafka** tab in the **Customize Services** step.

Steps

1. For your initial installation, accept the default values set by Apache Ambari.
2. If Ambari prompts you with Some configurations need your attention before you can proceed, review the list of properties and provide the required information.
3. Review the *Apache Kafka Component Guide* for information about configuring Apache Storm to meet your operational objectives.

More Information

Configuring Kafka for Production Environments

5.6. Configure Storm

About This Task

You can configure Storm from the **Storm** tab in the **Customize Services** step.

Steps

1. For your initial installation, accept the default values set by Ambari.
2. If Ambari prompts you with:

   Some configurations need your attention before you can proceed.

   Review the list of properties and provide the required information.
3. Review the *Apache Storm Component Guide* for information about configuring storm to meet your operational objectives.

More Information

Configuring Storm for Production Environments

5.7. Configure Log Search

About This Task

To ensure that you can view logs in the new SAM Log Search, you can manually review and adjust Log Search Settings for storm_worker and storm_worker_event.

Steps

1. From the left-hand navigation pane, select **Log Search | Configs**.
2. Manually set the Log Feeder Log Levels Filter for storm_worker and storm_worker_event to include **Info**, **Debug**, and **Trace**.

5.8. Deploy the Cluster Services

After you finish the wizard and deploy the cluster, some services might fail to start. If this is the case, you can start those services individually by launching them from the Ambari dashboard Services pane.

Steps

1. From Ambari's left-hand **Services** pane, click the service you want.
2. From the **Quick Links** drop-down, select the UI option.
3. Find links for the SAM UI under **Streaming Analytics Manager** and for the Schema Registry UI under **Registry.**

Result

The UI for your HDF service opens in a new window.

5.9. Access the UI for Deployed Services

About This Task

Once you have deployed your Ambari-managed cluster, you can launch the UI for any of the services from Ambari.

Steps

1. From Ambari's left-hand **Services** pane, click the service you want.
2. From the **Quick Links** drop-down, select the UI option.
3. Find links for the SAM UI under **Streaming Analytics Manager** and for the Schema Registry UI under **Registry.**

Result

The UI for your HDF service opens in a new window.

6. Configuring Schema Registry and SAM for High Availability

About This Task

You can configure Schema Registry and SAM for high availability.

Steps for Configuring SAM for HA

1. Install two or more instances of SAM on unique nodes.
2. From the **Services** pane, select **Streaming Analytics Manager** and click the **Configs** tab.
3. In the **Jar Storage Type** drop down, select **HDFS** or **Database**.

Note

If you are using a MySQL database, ensure that you make adjustments to the database configuration as well. `max_allowed_packet` must be greater than the maximum file size of any custom processor or user defined function that will be uploaded.

Steps for Configuring Schema Registry for HA

1. Install two or more instances of Schema Registry on unique nodes.
2. From the **Services** pane, select **Schema Registry** and click the **Configs** tab.
3. In the **Jar Storage Type** drop down, select **HDFS**.

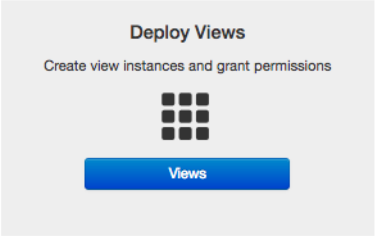
7. Install the Storm Ambari View

About This Task

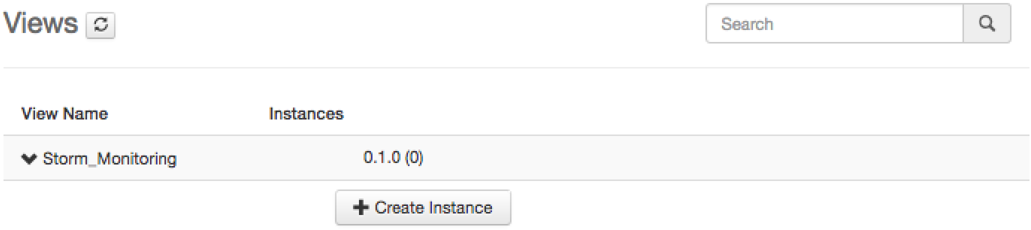
The Storm Ambari view provides you a number of different troubleshooting and debugging tools.

Steps

1. From the **admin** drop-down, select **Manage Ambari**.
2. Click the **Views** button.

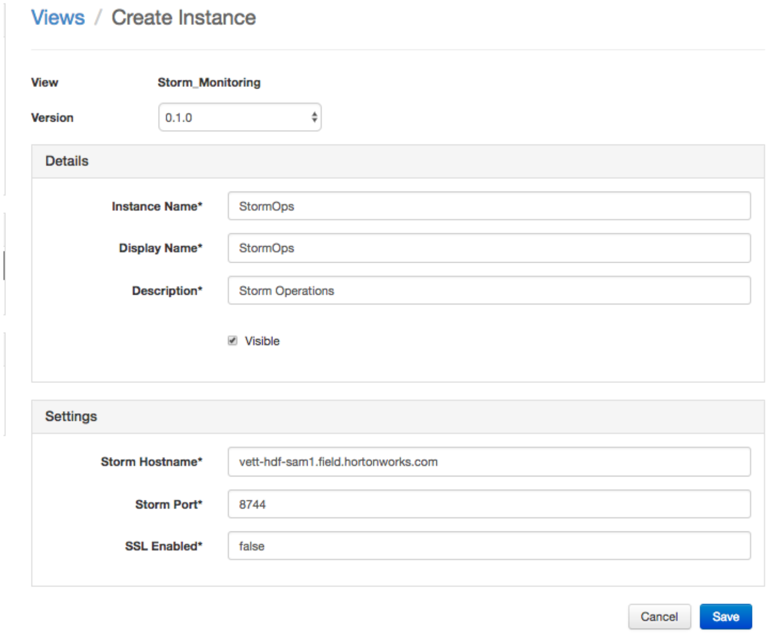


3. From the list of available Views, expand **Storm_Monitoring** and click + **Create Instance**.



View Name	Instances
Storm_Monitoring	0.1.0 (0)

4. Configure the Storm Ambari View.



Views / Create Instance

View: Storm_Monitoring

Version: 0.1.0

Details

Instance Name*: StormOps

Display Name*: StormOps

Description*: Storm Operations

☑ Visible

Settings

Storm Hostname*: vett-hdf-sam1.field.hortonworks.com

Storm Port*: 8744

SSL Enabled*: false

a. **Instance Name** and **Display Name** may not have an spaces.

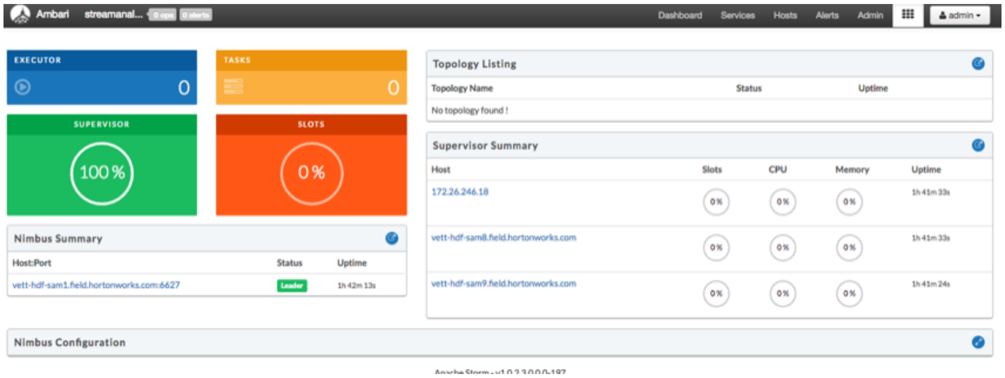
b. The **Storm Hostname** refers to the host where the Storm UI Server is deployed.

c. The Storm port is the Storm UI port server (keep it as default 8744 if you have not changed it).

d. Click **Save**.

Result

After saving it, you should see a menu item for the Storm Ambari View.



Topology Listing

Topology Name	Status	Uptime
No topology found !		

Supervisor Summary

Host	Slots	CPU	Memory	Uptime
172.26.246.18	0%	0%	0%	1h 41m 33s
vett-hdf-sam8.field.hortonworks.com	0%	0%	0%	1h 41m 33s
vett-hdf-sam9.field.hortonworks.com	0%	0%	0%	1h 41m 24s

Nimbus Summary

Host:Port	Status	Uptime
vett-hdf-sam1.field.hortonworks.com:6627	Leader	1h 42m 13s

Nimbus Configuration

Apache Storm - v1.0.2.3.0.0.0-197

8. Using a Local Repository

Local repositories are frequently used in enterprise clusters that have limited outbound internet access. In these scenarios, having packages available locally provides more governance, and better installation performance. These repositories are used heavily during installation for package distribution, as well as post-install for routine cluster operations such as service start/restart operations. The following sections describe the steps necessary to set up and use a local repository:

- Obtaining the Repositories [24]
- Set up a local repository having:
  - Setting Up a Local Repository with No Internet Access [26]
  - Setting up a Local Repository With Temporary Internet Access [28]
- Preparing The Ambari Repository Configuration File [29]

Important

Setting up and using a local repository is **optional**. After obtaining

8.1. Obtaining the Repositories

This section describes how to obtain:

- Ambari Repositories [24]
- HDP Stack Repositories [24]

8.1.1. Ambari Repositories

If you do not have Internet access, use the link appropriate for your OS family to **download a tarball** that contains the software for setting up Ambari.

If you have temporary Internet access, use the link appropriate for your OS family to **download a repository file** that contains the software for setting up Ambari.

Ambari 2.6.2 Repositories

OS	Format	URL
RedHat 7	Base URL	http://public-repo-1.hortonworks.com/ambari/centos7-ppc/2.x/updates/2.6.2.0
	Repo File	https://public-repo-1.hortonworks.com/ambari/centos7-ppc/2.x/updates/2.6.2.0/ambari.repo
	Tarball md5 | asc	http://public-repo-1.hortonworks.com/ambari/centos7-ppc/2.x/updates/2.6.2.0/ambari-2.6.2.0-centos7-ppc.tar.gz

8.1.2. HDP Stack Repositories

If you do not have Internet access, use the link appropriate for your OS family to download a tarball that contains the software for setting up the Stack.

If you have temporary Internet access, use the link appropriate for your OS family to download a repository file that contains the software for setting up the Stack.

- HDP 2.6 Repositories [25]

8.1.2.1. HDP 2.6 Repositories

OS	Version Number	Repository Name	Format	URL
RedHat 7	HDP-2.6.5.0	HDP	Version Definition File (VDF)	http://public-repo-1.hortonworks.com/HDP/centos7-ppc/2.x/updates/2.6.5.0-292.xml
			Base URL	http://public-repo-1.hortonworks.com/HDP/centos7-ppc/2.x/updates/2.6.5.0
			Repo File	http://public-repo-1.hortonworks.com/HDP/centos7-ppc/2.x/updates/2.6.5.0/hdp.repo
			Tarball md5 | asc	http://public-repo-1.hortonworks.com/HDP/centos7-ppc/2.x/updates/2.6.5.0/HDP-2.6.5.0-centos7-ppc-rpm.tar.gz
		HDP-UTILS	Base URL	http://public-repo-1.hortonworks.com/HDP-UTILS-1.1.0.22/repos/centos7-ppc
			Repo File	http://public-repo-1.hortonworks.com/HDP-UTILS-1.1.0.22/repos/centos7-ppc/hdp-utils.repo
			Tarball md5 | asc	http://public-repo-1.hortonworks.com/HDP-UTILS-1.1.0.22/repos/centos7-ppc/HDP-UTILS-1.1.0.22-centos7-ppc.tar.gz
		HDP-GPL	URL	http://public-repo-1.hortonworks.com/HDP-GPL/centos7-ppc/2.x/updates/2.6.5.0/hdp.gpl.repo
			Tarball	http://public-repo-1.hortonworks.com/HDP-GPL/centos7-ppc/2.x/updates/2.6.5.0/HDP-GPL-2.6.5.0-centos7-ppc-gpl.tar.gz

8.1.3. Setting Up a Local Repository

Based on your Internet access, choose one of the following options:

- No Internet Access

  This option involves downloading the repository tarball, moving the tarball to the selected mirror server in your cluster, and extracting to create the repository.

- Temporary Internet Access

  This option involves using your temporary Internet access to sync (using reposync) the software packages to your selected mirror server and creating the repository.

  Both options proceed in a similar, straightforward way. Setting up for each option presents some key differences, as described in the following sections:

  - Getting Started Setting Up a Local Repository [25]
  - Setting Up a Local Repository with No Internet Access [26]
  - Setting up a Local Repository With Temporary Internet Access [28]

8.1.4. Getting Started Setting Up a Local Repository

To get started setting up your local repository, complete the following prerequisites:

- Select an existing server in, or accessible to the cluster, that runs a supported operating system.
- Enable network access from all hosts in your cluster to the mirror server.
- Ensure the mirror server has a package manager installed such as yum (RHEL7).
- **Optional:** If your repository has temporary Internet access, and you are using RHEL/CentOS/Oracle Linux as your OS, install yum utilities:

  ```
  yum install yum-utils createrepo
  ```

1. Create an HTTP server.

   a. On the mirror server, install an HTTP server (such as Apache httpd) using the Apache instructions.

   b. Activate this web server.

   c. Ensure that any firewall settings allow inbound HTTP access from your cluster nodes to your mirror server.

   ### Note

   If you are using Amazon EC2, make sure that SELinux is disabled.

2. On your mirror server, create a directory for your web server.

   - For example, from a shell window, type:

     ```
     mkdir -p /var/www/html/
     ```

   - If you are using a symlink, enable the

     ```
     followsymlinks
     ```

     on your web server.

Next Steps

After you have completed the steps in this section, move on to specific set up for your repository internet access type.

More Information

httpd://apache.org/download.cgi

8.1.4.1. Setting Up a Local Repository with No Internet Access

Prerequisites

Complete the Getting Started Setting up a Local Repository procedure.

--

To finish setting up your repository, complete the following:

Steps

1. Obtain the tarball for the repository you would like to create.

2. Copy the repository tarballs to the web server directory and untar.

   a. Browse to the web server directory you created.

   ```
   cd /var/www/html/
   ```

   b. Untar the repository tarballs to the following locations: where <web.server>, <web.server.directory>, <OS>, <version>, and <latest.version> represent the name, home directory, operating system type, version, and most recent release version, respectively.

   Untar Locations for a Local Repository - No Internet Access

Repository Content	Repository Location
Ambari Repository	Untar under <web.server.directory>
HDP Stack Repositories	Create directory and untar under <web.server.directory>/hdp

3. Confirm you can browse to the newly created local repositories.

   URLs for a Local Repository - No Internet Access

Repository	Base URL
Ambari Base URL	http://<web.server>/Ambari-2.6.1.0/centos7
HDP Base URL	http://<web.server>/hdp/HDP/centos7/2.x/updates/<latest.version>
HDP-UTILS Base URL	http://<web.server>/hdp/HDP-UTILS-<version>/repos/centos7

   where <web.server> = FQDN of the web server host.

   Important

   Be sure to record these Base URLs. You will need them when installing Ambari and the cluster.

4. Optional: If you have multiple repositories configured in your environment, deploy the following plug-in on all the nodes in your cluster.

   - Install the plug-in.

     -
       ```
       yum install yum-plugin-priorities
       ```

   - Edit the `/etc/yum/pluginconf.d/priorities.conf` file to add the following:

     ```
     [main]
     ```

     ```
     enabled=1
     ```

     ```
     gpgcheck=0
     ```

More Information

Obtaining the Repositories [24]

8.1.4.2. Setting up a Local Repository With Temporary Internet Access

Prerequisites

Complete the Getting Started Setting up a Local Repository procedure.

- -

To finish setting up your repository, complete the following:

Steps

1. Put the repository configuration files for Ambari and the Stack in place on the host.
2. Confirm availability of the repositories.

   ```
   yum repolist
   ```

3. Synchronize the repository contents to your mirror server.

   - Browse to the web server directory:
   - For Ambari, create `ambari` directory and reposync.

     ```
     mkdir -p ambari/centos7
     ```

     ```
     cd ambari/centos7
     ```

     ```
     reposync -a ppc64le -r ambari-2.6.2.0
     ```

   - For HDP Stack Repositories, create `hdp` directory and reposync.

     ```
     mkdir -p hdp/centos7
     ```

     ```
     cd hdp/centos7
     ```

     ```
     reposync -a ppc64le -r HDP-2.6.5.0
     ```

     ```
     reposync -a ppc64le -r HDP-UTILS-1.1.0.22
     ```

4. Generate the repository metadata.

   - For Ambari:

     ```
     createrepo <web.server.directory>/ambari/centos7/Updates-Ambari-2.6.2.0
     ```

   - For HDP Stack Repositories:

     ```
     createrepo <web.server.directory>/hdp/centos7/HDP-<latest.version>
     ```

     ```
     createrepo <web.server.directory>/hdp/centos7/HDP-UTILS-<version>
     ```

5. Confirm that you can browse to the newly created repository.

   URLs for the New Repository

Repository	Base URL
Ambari Base URL	http://<web.server>/ambari/centos7/Updates-Ambari-2.6.2.0

Repository	Base URL
HDP Base URL	http://<web.server>/hdp/centos7/HDP-<latest.version>
HDP-UTILS Base URL	http://<web.server>/hdp/centos7/HDP-UTILS-<version>

where <web.server> = FQDN of the web server host.

Important

Be sure to record these Base URLs. You will need them when installing Ambari and the Cluster.

6. Optional. If you have multiple repositories configured in your environment, deploy the following plug-in on all the nodes in your cluster.

   - Install the plug-in.

     - `yum install yum-plugin-priorities`

   - Edit the `/etc/yum/pluginconf.d/priorities.conf` file to add the following:

     ```
     [main]

     enabled=1

     gpgcheck=0
     ```

More Information

Obtaining the Repositories [24]

8.1.5. Preparing The Ambari Repository Configuration File

1. Download the `ambari.repo` file from the public repository.

   ```
   http://public-repo-1.hortonworks.com/ambari/centos7-ppc/2.x/updates/2.6.2.0/
   ambari.repo
   ```

2. In the ambari.repo file, replace the base URL `baseurl` with the local repository URL.

   ### Note

   You can disable the GPG check by setting gpgcheck =0. Alternatively, you can keep the check enabled but replace the gpgkey with the URL to the GPG-KEY in your local repository.

   ```
   [Updates-Ambari-2.6.2.0]

   name=Ambari-2.6.2.0-Updates

   baseurl=INSERT-BASE-URL

   gpgcheck=1

   gpgkey=http://public-repo-1.hortonworks.com/ambari/centos7-ppc/RPM-GPG-KEY/
   RPM-GPG-KEY-Jenkins

   enabled=1
   ```

```
priority=1
```

Base URL for a Local Repository

Local Repository	Base URL
Built with Repository Tarball (No Internet Access)	http://<web.server>/Ambari-2.6.2.0/centos7
Built with Repository File (Temporary Internet Access)	http://<web.server>/ambari/centos7/Updates-Ambari-2.6.2.0

where <web.server> = FQDN of the web server host.

3. Place the ambari.repo file on the machine you plan to use for the Ambari Server.

   - `/etc/yum.repos.d/ambari.repo`
   - Edit the `/etc/yum/pluginconf.d/priorities.conf` file to add the following:

     ```
     [main]
     ```

     ```
     enabled=1
     ```

     ```
     gpgcheck=0
     ```

Next Steps

Proceed to *Installing Ambari Server* to install and setup Ambari Server.

More Information

Setting Up a Local Repository with No Internet Access [26]

Setting up a Local Repository With Temporary Internet Access [28]

9. Navigating the HDF Library

To navigate the Hortonworks DataFlow (HDF) documentation library, begin by deciding your current goal.

If you want to...	See this document...
Install or upgrade an HDF cluster using Apache Ambari	• Release Notes • Support Matrix • Planning Your Deployment • Ambari Upgrade • MiNiFi Java Agent Quick Start
Get started with HDF	• Getting Started with Apache NiFi • Getting Started with Stream Analytics
Use and administer HDF Flow Management capabilities	• Apache NiFi User Guide • Apache NiFi Administration Guide • Apache NiFi Developer Guide • Apache NiFi Expression Language Guide • MiNiFi Java Agent Administration Guide
Use and administer HDF Stream Analytics capabilities	• Streaming Analytics Manager User Guide • Schema Registry User Guide • Apache Storm Component Guide • Apache Kafka Component Guide