Put your title here

Contents

Removing deprecated components and services

Provides the steps required to remove the deprecated components and services which are not supported by Hortonworks Data Platform (HDP).

About this task

You must stop and remove all the deprecated services and components from the Ambari server.

Procedure

1. **Remove the HBASE REST SERVER component**. For each HBaseRestServer:
   a. In Ambari, select the **Hosts** link
   b. Select the host with the HBASE REST SERVER component running
   c. From the Actions drop-down menu select **Stop**
   d. From the Actions drop-down menu select **Delete** to delete the component
   e. Delete the HBASE REST SERVER component using the following REST API as follows:

```
curl -u <admin-username>:<admin-password> -H "X-Requested-By:ambari" -i -X DELETE
http://<ambari-server-host>:<port>/api/v1/clusters/<cluster-name>/services/HBASE/
components/HBASE_REST_SERVER

curl -u admin:admin -H "X-Requested-By:ambari" -i -X DELETE
http://iop1:8080/api/v1/clusters/iop/services/HBASE/components/HBASE_REST_SERVER
```

2. **Remove the TITAN, SOLR, JNBG, SYSTEMML and R4ML services in order** (if present). To remove a component, follow these steps:
   a. In Ambari, select the service (for example, TITAN) in the **Services** list.
   b. Select **Service Actions** > **Stop** to stop the service.
   c. Select **Service Actions** > **Delete** to delete the service.

   Repeat this process for all the services listed above.

What to do next

The next step in the migration process is to migrate from IOP to HDP. See **[HWX REFERENCE]** for details.

Cleaning up IBM value-add services

This topic describes how to remove IBM value-added services (such as Text Analytic) from your system as part of a migration.

About this task

These clean-up processes do not remove the Ambari server, nor do they impact any of the configurations. The scripts will remove the top-level value-add RPMs. The value-add components you might see include:

- Text Analytics (component name: TEXTANALYTICS)
- BigInsights Home (component name: WEBUIFRAMEWORK)
- BigSheets (component name: BIGSHEETS)
- Big R (component name: BIGR)
- Data service manager (component name: DATASERVERMANAGER)

Note: The cleanup process for removing BigSheets removes data on HDFS for the child workbooks. If you want to save any of the child workbook data, use the **Export Data** option from the BigSheets home page for each of the child workbooks and save the data on HDFS. For information on how to export data, see Exporting data from a BigSheets workbook.

In the Ambari UI, when you navigate to the service page for a service, the last action listed under **Service Actions** is **Delete Service**. The **Delete Service** option should NOT be used for these services, as it can leave the service in an undefined state. The proper way to remove any of these four services is to follow the steps on this page. If you already used the **Delete Service** option for one of the services, you still must follow the steps on this page for proper removal of the service. In such a scenario, you will see a message like the following during removal (using Text Analytics as an example). You must type in *y* or *Y* to finish the removal successfully.

```
Current TEXTANALYTICS status: EMPTY
Unable to detect TEXTANALYTICS in Ambari. It may have already been removed.
Would you like to continue cleanup? (Y/n) y
```

Procedure

1. Navigate to the directory that contains the clean up scripts:

   ```
   cd /usr/ibmpacks/bin/<version>
   ```

2. The value-add services include scripts to help you remove the value-add services and to clean up your environment:

Option	Description
Service cleanup	`remove_value_add_services.sh -u <AMBARI_ADMIN_USERNAME> -p <AMBARI_ADMIN_PASSWORD> -x <AMBARI_PORT> -a <STOPSERVICECOUNT> -b <REMOVESERVICECOUNT> -r -l -c <RUN_AS_USER> -f -s -q`

   Use the following parameter definitions:

   -u
   The Ambari administrator user name.

   -p
   The Ambari administrator password.

   -x
   The Ambari server port number.

   -s
   The service to remove. The following values are allowed:

   Service

   - TEXTANALYTICS
   - WEBUIFRAMEWORK - This is the BigInsights Home.
   - BIGSHEETS
   - BIGR
   - DATASERVERMANAGER

   -f
   This option is optional. The FORCE option allows you to continue removing the service, even if intermittent steps fail.

   

   Warning: This might result in an Ambari unknown service state.

   -q
   This option is optional. If removing a service, the parameter specifies to remove stack files that are associated with the service.

Warning: When running the remove service script, this option prevents a reinstallation.

-a
This option is optional. Specifies the number of attempts to stop the service.

-b
This option is optional. Specifies the number of attempts to remove the service.

-r
(optional) Removes service users.

-l
This option is optional. It enables secure https mode.

-c
This option is optional. Run as user if a non-root user is installed.

Note:

For the Text Analytics service -TEXTANALYTICS- if an existing MySQL database server was used during install, then delete the database manually from the server after running the removal script:

```
mysql> drop database tawebtoolingdb;
```

3. Complete the removal process with the following steps:
   a) Restart the Ambari service to make sure that the cache is cleared:

      Note: Some previously launched services, such as Knox, may take several minutes to refresh. Ambari may display these services in a warning state and attempts to manually launch these services may fail during the refresh.

      ```
      sudo ambari-server restart
      ```

   b) There are files that are left in `/var/lib/ambari-server/resources/stacks/BigInsights/4.2.5/services/$SERVICE/package/archive.zip`. These can remain and have no impact on future service additions.

Example

Service cleanup examples:

Normal removal

Remove the BigR service:

```
>sudo remove_value_add_services.sh
  -u admin -p admin -x 8081 BIGR
```

Removal including users

Remove the Big R service including the users:

```
>sudo remove_value_add_services.sh
  -u admin -p admin -x 8081 BIGR -r
```

Run as non-root user

Remove the Big R service as non-root user (with sudo privilege) *biadmin*:

```
>sudo remove_value_add_services.sh
  -u admin -p admin -x 8081 BIGR
  -r  -c biadmin
```

What to do next

The next step in the migration process is to remove deprecated components and services. See “Removing deprecated components and services” on page 1 for details.

Backing up metadata

Provides information on how to back up your metadata.

About this task
Provides information on how to back up:

- Data Service Manager (DSM) metadata
- Big SQL metadata using the Big SQL python script `bigsql_upgrade.py`

Procedure

1. Install the Big SQL version 5.0.1 service definition:
   a. **Obtain the IBM Big SQL package** . See #unique_4 for details.
   b. Execute the downloaded file; choose the **migration** option when asked. See #unique_5 for details.
   c. After the downloaded file has been executed, DO NOT enable the Big SQL 5.0.1 extension with the `EnableBigSQLExtension.py` script.
2. **Run the backup option of the Big SQL python script**. (For details on the script, see "bigsql_upgrade.py - Big SQL upgrade utility" on page 8.)
   a. Back up your Big SQL environment by running the *Backup* option of the `bigsql_upgrade.py` python command. The *Backup* option performs the following actions on all nodes of the cluster:
      - Backs up the Big SQL catalog, metadata and configuration information.
      - Installs the binaries for the new version.

      To perform the backup phase of the upgrade, run the `bigsql_upgrade.py` script with the *-m* option and the value `Backup`:

      ```
      python /usr/ibmpacks/scripts/5.0.1.0/upgrade/bigsql_upgrade.py -m Backup
      ```
   b. When the Backup phase is complete, the Big SQL service is no longer visible in the Ambari dashboard. However, it is operational, but not running. If needed, you can start the service from the command line and use it. In this case, the version executed is the initial Big SQL version.
3. (Optional in case the backup phase fails) Consult the master log output or the upgrade log located at `/var/ibm/bigsql/logs/upgrade.log` to identify and resolve the problem. After it is resolved, re-run the backup phase.
4. **Backup DSM metadata**. Save a copy of the following files to a backup directory, such as `/tmp`:
   - `/usr/ibmpacks/IBM-DSM/$VERSION/ibm-datasrvrmgr/Config/default_rep_db`
   - `/usr/ibmpacks/IBM-DSM/$VERSION/ibm-datasrvrmgr/Config/privileges.json`

What to do next
The next step in the migration process is to upgrade your version of Ambari. See **[HWX REFERENCE]** for details.

Migrating from IOP to HDP

This topic shows you how to migrate from IBM Open Platform (IOP) 4.2.x to Hortonworks Data Platform (HDP) 2.6.2 and upgrade Big SQL.

Before you begin

- You must upgrade Big SQL to version 5.0.1 if you want full, functional support with the most recent version of the Hadoop Data Platform (HDP). For compatibility details see the #unique_8 topic.
- To upgrade Big SQL, you must have a user with the following attributes:
  - passwordless sudo access on all nodes of the cluster, including the Ambari server itself.
  - The ability to connect passwordlessly through ssh from the Ambari server to all Big SQL nodes.

This user can be root. If the user is not root, the user name must be passed to the upgrade script with the -a option. The upgrade script must be run with root user privilege. This can be achieved by using the `sudo` command. If you have configured Ambari for non-root access (see Configuring Ambari for non-root access), use the -a option with the user name created for that purpose.

- The HDFS, Hive, and HBase services (and the services they depend on) must be running for you to perform the Big SQL upgrade. HDFS should not be running in safe mode.
- Big SQL installation requires about 2 GB of free disk space on the `/usr` partition. The upgrade process requires about 4 GB of free disk space on the root partition. This space is used temporarily by the upgrade process. It is released when the upgrade is complete.
- You must disable all high availability before performing an upgrade.
- Ambari configuration groups are not supported. The upgrade script produces a warning message if you use configuration groups. If you override the warning, you must validate that the configuration of all nodes in all configuration groups is updated as part of the upgrade. It is recommended that you remove all configuration groups before performing an upgrade.
- You must disable Yarn and Slider support for Big SQL before performing an upgrade.

For additional prerequisites for your migration process, see **[REFERENCE to HW content]**.

About this task

Provides an overview of the full process for you to migrate from IBM Open Platform (IOP) versions 4.2.x to Hortonworks Data Platform (HDP) version 2.6.2.

Procedure

1. **Back up metadata**.
   - For details on how to backup IBM BigSQL and Data Server Manager (DSM), see “Backing up metadata” on page 4.
   - For SOLR backup details, see **[HWX REFERENCE]**.
2. **Upgrade your version of Ambari**.
   For Ambari upgrade details, see **[HWX REFERENCE]**.
3. **Clean up IBM value-added services from your environment**.
   For value-add service cleanup details, see “Cleaning up IBM value-add services” on page 1.
4. **Remove deprecated components and services from your environment**.
   For component and service deletion details, see “Removing deprecated components and services” on page 1.
5. **Upgrade the stack**.
   For stack upgrade details, see **[HWX REFERENCE]**.
6. **Upgrade and finalize your Big SQL installation**.
   For upgrading and finalizing details, see “Upgrading and finalizing Big SQL” on page 5.
7. **Upgrade IBM DSM**.
   For upgrading details, see “Upgrading IBM Data Server Manager (DSM)” on page 7.

Upgrading and finalizing Big SQL

As a step in an IOP to HDP Big SQL migration, you must upgrade and finalize your version of Big SQL.

Before you begin

If you are performing this step as part of a migration, you must follow all the preceding steps described in “Migrating from IOP to HDP” on page 4.

If you are upgrading Big SQL in a non-root installation environment, you must follow the steps in #unique_12.

About this task

This topic describes how to upgrade and finalize your version of Big SQL.

Procedure

1. Ensure the umask for the root and bigsql users is 0022. Run the following commands on each node on the cluster as the root user or a user with sudo privileges to run the following commands as root:

   ```
   umaskmigrate_prep_non_root
   sudo su - root -c umask
   sudo su - bigsql -c umask
   ```

2. Disable TCP/IP connections to the Big SQL service.

   Perform the following steps on the BigSQL HeadNode as the BigSQL user:

   ```
   db2set DBCOMM=
   restart Big SQL service.
   ```

3. Stop db2 level auditing of Big SQL.

   Login to the headnode as the Big SQL user and issue the following commands to stop auditing:

   ```
   db2audit stop
   ```

   After the upgrade has been completed, you can restart auditing from the Big SQL user on the headnode by issuing the command

   ```
   db2audit start
   ```

   .

4. Use the *Upgrade* option of the `bigsql_upgrade.py` script to upgrade your version of Big SQL.

   The upgrade phase takes the following actions on all nodes of the cluster:

   - Upgrades the Big SQL catalog, metadata and configuration information to the new version of Big SQL.

   ```
   python /usr/ibmpacks/scripts/5.0.1.0/upgrade/bigsql_upgrade.py -m Upgrade
   ```

   To perform the upgrade option of the Big SQL upgrade, run the `bigsql_upgrade.py` script with the -m option and the value Upgrade. Include any additional options as documented in “bigsql_upgrade.py - Big SQL upgrade utility” on page 8.

   When the Upgrade phase is complete, the Big SQL service is not visible in the Ambari dashboard. However, the new version of BigInisghts Big SQL is operational and running. It is possible to connect applications to the BigInisghts Big SQL server to run sanity tests before proceeding with the Finalize phase of the upgrade.

   It is not possible to re-run the upgrade phase immmediately after it has completed (succesfully or not).

5. In case the previous upgrade step fails, follow these steps:

   a. Consult the script output or the upgrade log located at `/var/ibm/bigsql/logs/upgrade.log` to identify the problem.
   b. Use the *Restore* option of the `bigsql_upgrade.py` script to restore to pre-upgrade conditions:

      ```
      python /usr/ibmpacks/scripts/5.0.1.0/upgrade/bigsql_upgrade.py -m Restore
      ```

   c. Repair the issue that caused the failure.
   d. Re-run the upgrade command as shown in Step 1.

6. Enable the Big SQL service extension by running the command:

   ```
   EnableBigSQLExtension.py
   ```

   For details on this script, see #unique_13.

7. Clean up the BIGSQL and DSM service from the old stack:

Run the following commands:

```
rm -rf /var/lib/ambari-server/resources/stacks/HDP/2.4/services/BIGSQL /var/lib/
ambari-server/resources/stacks/HDP/2.5/services/BIGSQL /var/lib/ambari-server/
resources/stacks/HDP/2.6/services/BIGSQL

rm -rf /var/lib/ambari-server/resources/stacks/HDP/2.4/services/DATASERVERMANAGER /
var/lib/ambari-server/resources/stacks/HDP/2.5/services/DATASERVERMANAGER /var/lib/
ambari-server/resources/stacks/HDP/2.6/services/DATASERVERMANAGER
```

8. Use the *Finalize* option of the `bigsql_upgrade.py` script to finalize your upgrade of Big SQL.

   CAUTION: After the upgrade is finalized, the backups of the catalog, metadata and configuration information of Big SQL are cleaned up and are no longer available.

   The finalize phase takes the following actions on all nodes of the cluster:

   a. Registers the Big SQL service in the Ambari dashboard.
   b. Cleans up the binaries of the previous Big SQL version.
   c. Cleans up the backups that were created during the backup phase.

   To perform the finalize phase of the Big SQL upgrade, run the `bigsql_upgrade.py` script with the -m option and the value Finalize. Include any additional options as documented in the "bigsql_upgrade.py - Big SQL upgrade utility" on page 8.

   ```
   python /usr/ibmpacks/scripts/5.0.1.0/upgrade/bigsql_upgrade.py -m Finalize
   ```

   When the Finalize phase is complete, the backups no longer exist. The Big SQL service is visible in the Ambari dashboard. The new version of Big SQL is operational and running.

   In case of failure of the finalize phase, consult the script output or the upgrade log located at `/var/ibm/bigsql/logs/upgrade.log` to identify and resolve the problem. After it is resolved, re-run the finalize phase.

9. Re-enable your TCP/IP connections by logging on as the Big SQL user to the head node and issuing the following command:

   ```
   db2set DB2COMM=TCPIP
   restart BigSQL Service
   ```

What to do next

The next step in the migration process is to upgrade DSM. See "Upgrading IBM Data Server Manager (DSM)" on page 7 for details.

Upgrading IBM Data Server Manager (DSM)

As the final step in an IOP to HDP Big SQL migration, you must upgrade the IBM DSM.

Before you begin

If you are performing this step as part of a migration, you must follow all the preceding steps described in "Migrating from IOP to HDP" on page 4.

Note: Data Server Manager requires Knox to be installed, configured, and started. For more information, see #unique_14.

About this task

This topic describes how to upgrade your version of DSM.

Procedure

1. In the Ambari UI, re-install DSM on the cluster.

   Two important notes while re-installing DSM:

- Ignore the "Before you begin" portion of the DSM installation topic
- Click OK when asked if Ambari can install TEZ

2. Restore the backup copy of the `default_rep_db` to `/usr/ibmpacks/IBM-DSM/5.0.1.0/ibm-datasrvrmgr/Config`.
3. Restore the backup copy of `privileges.json` (if the file exists) to `/usr/ibmpacks/IBM-DSM/5.0.1.0/ibm-datasrvrmgr/Config`.
4. Ensure the ownership on the restored file `privileges.json`, the restored directory `default_rep_db`, and all of its contents match the rest of the files under the directory `/usr/ibmpacks/IBM-DSM/5.0.1.0/ibm-datasrvrmgr/Config`.
5. In Ambari, restart DSM by selecting DSM in the list of services and then selecting **Service Actions** > **Restart All**.

bigsql_upgrade.py - Big SQL upgrade utility

Allows you to Backup, Upgrade, Finalize, or Restore a Big SQL installation.

Authorization

You must run this utility as root; you must either be logged on as the root user or use the `sudo` command.

Prerequisites

The following are prerequisites for running the `bigsql_upgrade.py` command:

- To upgrade Big SQL, you must have a user with the following attributes:
  - passwordless sudo access on all nodes of the cluster, including the Ambari server itself.
  - The ability to connect passwordlessly through ssh from the Ambari server to all Big SQL nodes.

  This user can be root. If the user is not root, the user name must be passed to the upgrade script with the -a option. The upgrade script must be run with root user privilege. This can be achieved by using the `sudo` command. If you have configured Ambari for non-root access (see Configuring Ambari for non-root access), use the -a option with the user name created for that purpose.
- The HDFS, Hive, and HBase services (and the services they depend on) must be running for you to perform the Big SQL upgrade. HDFS should not be running in safe mode.
- Big SQL installation requires about 2 GB of free disk space on the `/usr` partition. The upgrade process requires about 4 GB of free disk space on the root partition. This space is used temporarily by the upgrade process. It is released when the upgrade is complete.
- You must disable all Big SQL high availability before performing an upgrade.
- Ambari configuration groups are not supported. The upgrade script produces a warning message if you use configuration groups. If you override the warning, you must validate that the configuration of all nodes in all configuration groups is updated as part of the upgrade. It is recommended that you remove all configuration groups before performing an upgrade.
- You must disable Yarn and Slider support for Big SQL before performing an upgrade.

Command parameters

-m *phase*

(Required) Specifies the upgrade phase under which the utility is to run. There are four valid modes:

Backup

Backs up the Big SQL database.

Note: The *Backup* phase removes Big SQL from Ambari.

Upgrade

Upgrades your version of Big SQL.

Finalize

Finalizes the Big SQL upgrade.

CAUTION: After the upgrade is finalized, the backups of the catalog, metadata and configuration information of Big SQL are cleaned up and are no longer available.

Restore

Restores the upgrade to the previously backed-up version. Used in the event of an upgrade failure.

Important: The restore phase is optional and must be run only in case of failure of the upgrade phase. The `bigsql_upgrade.py` script prevents you from running the restore phase if the upgrade phase is successful.

-u *ambari-admin-user*

(Optional) Specifies the Ambari admin user name. The default is admin.

-p *ambari-admin-password*

(Optional) Specifies the Ambari admin password. The default is admin.

-s *protocol*

(Optional) Specifies the Ambari server protocol (http or https) . The default is http.

-P *server-port*

(Optional) Specifies Ambari server port number. The defaults is 8080.

-k *kerberos-admin-principal*

(Optional) Specifies the Kerberos admin principal. This is required if Kerberos is enabled on the cluster.

-w *kerberos-admin-password*

(Optional) Specifies the Kerberos admin password. This is required if Kerberos is enabled on the cluster.

-a *ssh-user*

Specifies the identity used for non-root access to Ambari. The default is root.

Usage notes

The Big SQL service configuration contains the credentials *ambari-admin-user* and *ambari-admin-password* to connect Big SQL to the Ambari server. The Big SQL upgrade script `bigsql_upgrade.py` requires a user name and password to connect to Ambari. We highly recommended you use the same user name as that in the Big SQL service configuration. If you wish to use a different set of credentials, you must make sure that the ones in the Big SQL service configuration are valid (and in particular that the password is not expired).

The Big SQL administrator can upgrade a Kerberos-enabled cluster without disabling Kerberos. Make sure that the Kerberos administrator credentials are available. If your cluster is Kerberos enabled, and valid credentials are not provided during the upgrade process, the upgrade will not proceed.

Upgrade is supported with the following limitations:

- Big SQL high availability clusters. You must disable Big SQL high availability before performing an upgrade.
- Ambari configuration groups are not supported. The upgrade script produces a warning message if you use configuration groups. If you override the warning, you must validate that the configuration of all nodes in all configuration groups is updated as part of the upgrade. It is recommended that you remove all configuration groups before you run the upgrade.
- You cannot upgrade through the Ambari dashboard; it is a command-line upgrade only.
- Big SQL Yarn & Slider support. You must disable Yarn & Slider support for Big SQL before performing an upgrade.

Note: During the Big SQL migration process, BigSQL is fully removed from Ambari during most of the migration until the finalize step, discussed in "Upgrading and finalizing Big SQL" on page 5. You can not interact with BigSQL through Ambari (whether the web UI or REST API).

If you perfom an offline upgrade, make sure that the `repoinfo.xml` file is updated. If you upgrade from version 4.1.0.2 to version 4.2.2, make sure that the source and the target versions of the repo information are present in the file. Use the following example as a guide to what the file might look like:

```
<reposinfo>
<mainrepoid>IOP-4.2</mainrepoid>
<os family="redhat6">
  <repo>
```

```
    <baseurl>http://birepo-build.svl.ibm.com/repos/IOP/RHEL6/x86_64/4.2/20160401_0401</
baseurl>
    <repoid>IOP-4.2</repoid>
      <reponame>IOP</reponame>
  </repo>
  <repo>
    <baseurl>http://birepo-build.svl.ibm.com/repos/IOP-UTILS/RHEL6/x86_64/1.1</baseurl>
    <repoid>IOP-UTILS-1.1</repoid>
      <reponame>IOP-UTILS</reponame>
  </repo>
  <repo>
    <baseurl>http://birepo-build.svl.ibm.com/repos/BigInsights-Valuepacks/RHEL6/
x86_64/4.2.0.0/DEV/20160419_1025</baseurl>
    <repoid>BIGINSIGHTS-VALUEPACK-1.2.0.0</repoid>
    <reponame>BIGINSIGHTS-VALUEPACK-1.2.0.0</reponame>
   </repo>
  </os>
</reposinfo>
```

Examples

Example scenario:

For this example, assume the following criteria:

- The Ambari dashbard is `https://ambari.company.com:9443/`.
- The administrator user name is ambadmin and the password is ambpassword.
- Kerberos is not enabled.
- Ambari is set up for non-root access with an `ambari` user.
- You are logged in as the root user.

Run the 'Backup' phase of the Big SQL service upgrade process on your cluster:

```
/usr/ibmpacks/scripts/5.0.1.0/upgrade/BIGSQL/bigsql_upgrade.py
   -u ambadmin -p ambpassword -s https -P 9443 -a ssh_user -m Backup
```