Processing Big Data with Hadoop in Azure HDInsight

Lab 2A – Hive Fundamentals

Overview

In this lab, you will process data in web server log files by creating Hive tables, populating them with data, and querying them.

What You'll Need

To complete the labs, you will need the following:

- A web browser
- A Microsoft account
- A Microsoft Azure subscription
- A Microsoft Windows computer with the following software installed:
  - Microsoft Azure PowerShell
  - Microsoft Visual Studio with the Azure SDK
  - Microsoft Power BI Desktop
- The lab files for this course

Note: To set up the required environment for the lab, follow the instructions in the **Setup** document for this course. Specifically, you must have signed up for an Azure subscription, installed and configured Azure PowerShell, imported the publisher settings for your Azure subscription into PowerShell, installed Visual Studio and the Azure SDK, and installed Microsoft Power BI Desktop.

When working with cloud services, transient network errors can occasionally cause scripts to fail. If a script fails, and you believe that you have entered all of the required variables correctly; wait a few minutes and run the script again.

Preparing for Data Processing

Before you can use Hive to analyze the log files, you must provision an HDInsight cluster and upload the source data to Azure storage.

Provision an Azure Storage Account and HDInsight Cluster

Note: If you already have an HDInsight cluster and associated storage account, you can skip this task.

1. In a web browser, navigate to http://azure.microsoft.com. Then click **Portal**, and if prompted, sign in using the Microsoft account that is associated with your Azure subscription.
2. In the Microsoft Azure portal, view the **HDInsight** page and verify that there are no existing HDInsight clusters in your subscription.
3. Click **NEW** (at the bottom of the page) and then click **CUSTOM CREATE**. Then use the New HDInsight Cluster wizard to create a new cluster with the following settings. Click the arrows to navigate through all of the wizard pages:
    - **Cluster Name**: *Enter a unique name (and make a note of it!)*
    - **Cluster Type**: Hadoop
    - **Operating System**: Windows Server 20012 R2 Datacenter
    - **HDInsight Version**: 3.2 (HDP 2.2, Hadoop 2.6)
    - **Data Nodes**: 2
    - **Region**: *Select any available region*
    - **Head Node Size**: A3 (4 cores, 7 GB memory)
    - **Data Node Size**: A3 (4 cores, 7 GB memory)
    - **HTTP User Name**: *Enter a user name of your choice (and make a note of it!)*
    - **HTTP Password**: *Enter and confirm a strong password (and make a note of it!)*
    - **Enable the remote desktop for cluster:** Selected
    - **RDP User Name:** *Enter another user name of your choice (and make a note of it!)*
    - **RDP Password:** *Enter and confirm a strong password (and make a note of it!)*
    - **Expires on**: Select tomorrow's date
    - **Enter the Hive/Oozie Metastore**: Unselected
    - **Storage Account**: Create New Storage
    - **Account Name**: *Enter a unique name for your storage account (and make a note of it!)*
    - **Default Container**: *Enter a unique name for your container (and make a note of it!)*
    - **Additional Storage Accounts**: 0
    - **Additional scripts to customize the cluster**: *None*
4. Wait for the cluster to be provisioned and the status to change to **Running** (this can take a while.)

View Source Data

1. In the C:\HDILabs\Lab02A\iislogs folder, open any of the text files in Notepad. Each file in this folder is an Internet Information Services (IIS) log file from a web server, and there is a file for each day between January 1st 2008 and July 31st 2008. When you have viewed the information in the file, close Notepad.
2. View the contents of the C:\HDILabs\Lab02A\iislogs_gz folder. This contains the same log files compressed using the gzip compression algorithm, significantly reducing the size of the files.

Upload the Log Files to Azure Storage

1. In the C:\HDILabs\Lab02A folder, rename **Upload Files.txt** to **Upload Files.ps1** (you may need to modify the *View* options for the folder to see file extensions). Then right-click **Upload Files.ps1** and click **Edit** to open the script file in the PowerShell ISE.
2. Change the values assigned to the **$clusterName**, **$storageAccountName**, and **$containerName** variables to match the configuration of your HDInsight cluster.
3. Examine the rest of the script, noting that it uploads all of the files in the **iislogs_gz** folder to your Azure storage container, storing them in **/data/logs**.
4. Save the script. Then on the toolbar, click **Run Script**.
5. Observe the information displayed in the PowerShell command line pane as the script uploads each file to Azure storage. This can take a few minutes.
6. Close the PowerShell ISE.

Creating, Loading, and Querying Hive Tables

Now that you have provisioned an HDInsight cluster and uploaded the source data, you can create Hive tables and use them to process the data.

Create a Hive Table for the Raw Log Data

1. In a web browser, navigate to http://azure.microsoft.com. Then click **Portal**, and if prompted, sign in using the Microsoft account that is associated with your Azure subscription.
2. In the Azure portal, on the **HDInsight** page, select your HDInsight cluster and click **Query Console**. Then log into the query console using the HTTP user name and password for your cluster.
3. In the HDInsight query console, view the **Hive Editor** page.
4. In the **Query Name** box, type **Create Raw Table**. Then replace the default **Select** statement with the following HiveQL query (you can copy and paste this from **Create Raw Table.txt** in the C:\HDILabs\Lab02A folder):

```
CREATE TABLE rawlog
(log_date STRING,
 log_time STRING,
 c_ip STRING,
 cs_username STRING,
 s_ip STRING,
 s_port STRING,
 cs_method STRING,
 cs_uri_stem STRING,
 cs_uri_query STRING,
 sc_status STRING,
 sc_bytes INT,
 cs_bytes INT,
 time_taken INT,
 cs_user_agent STRING,
 cs_referrer STRING)
ROW FORMAT DELIMITED FIELDS TERMINATED BY ' ';
```

5. Click **Submit** and wait for the query job to complete (the status of the query in the **Job Session** area will change to *Completed* when the job is finished).

Load the Source Data into the Raw Log Table

1. In the Hive Editor, change the value in the **Query Name** box to **Load Raw Data**. Then replace the **CREATE TABLE** code you entered in the previous task with the following code (you can copy and paste this from **Load Raw Data.txt** in the C:\HDILabs\Lab02A folder):

```
LOAD DATA INPATH '/data/logs' INTO TABLE rawlog;
```

2. Click **Submit** and wait for the query job to complete.

Query the Raw Log Table

1. In the Hive Editor, change the value in the **Query Name** box to **Query Raw Data**. Then replace the **LOAD DATA** code you entered in the previous task with the following code (you can copy and paste this from **Query Raw Data.txt** in the C:\HDILabs\Lab02A folder):

```
SELECT * FROM rawlog LIMIT 100;
```

2. Click **Submit** and wait for the query job to complete. It may take a few minutes.
3. When the job has completed, click its name in the **Query Name** column of the **Job Session** table. This opens a new tab containing the output generated by the job.

4. View the output, noting that the query retrieved the first 100 rows from the table in tab-delimited text format. If you want to, you can download the output as a text file and open it in Notepad. Note that the output includes rows containing comments from the source document (the first column value for these rows is prefixed with a **#** character).
5. Close the web browser.

Clean the Log Data

1. Start Visual Studio and create a new project from the **Hive Application** template in the **HDInsight** category.
2. If the Server Explorer pane is not visible, on the **View** menu, click **Server Explorer**.
3. In Server Explorer, click **Connect to Microsoft Azure Subscription**, and then sign in using the Microsoft account associated with your Azure subscription. If the **Get Started with Azure** page opens, close it.
4. In the **Script.hql** page, enter the following HiveQL code (you can copy and paste this from **Clean Log Data.txt** in the C:\HDILabs\Lab02A folder):

```
CREATE EXTERNAL TABLE cleanlog
(log_date DATE,
 log_time STRING,
 c_ip STRING,
 cs_username STRING,
 s_ip STRING,
 s_port STRING,
 cs_method STRING,
 cs_uri_stem STRING,
 cs_uri_query STRING,
 sc_status STRING,
 sc_bytes INT,
 cs_bytes INT,
 time_taken INT,
 cs_user_agent STRING,
 cs_referrer STRING)
ROW FORMAT DELIMITED FIELDS TERMINATED BY ' '
STORED AS TEXTFILE LOCATION '/data/cleanlog';

INSERT INTO TABLE cleanlog
SELECT *
FROM rawlog
WHERE SUBSTR(log_date, 1, 1) <> '#';
```

5. On the toolbar in the **Script.hql** pane, in the drop-down list, select your HDInsight cluster. Then click **Submit**.
6. In the **Job Summary** pane, view the job status. Wait a few minutes and then click **Refresh**, and keep doing this until the status changes from **Running** to **Completed** (this may take a while – in a later module you'll learn how to improve the performance of Hive queries).
7. In Server Explorer, expand **Azure**, expand **HDInsight**, expand your HDInsight cluster, expand your Azure storage account, and double-click the default container for your HDInsight cluster.
8. In the container page, double-click the **data** folder and the **cleanlog** folder, and note that a file containing the data for the **cleanlog** table has been created in this location.
9. In Server Explorer, right-click your HDInsight server and click **Write a Hive Query**. Then in the new query page, enter the following HiveQL code (you can copy and paste this from **Query Clean Log.txt** in the C:\HDILabs\Lab02A folder):

```
SELECT * FROM cleanlog LIMIT 100;
```

10. On the toolbar, in the HDInsight cluster drop-down list, ensure that your HDInsight cluster is still selected. Then click **Submit**.
11. In the **Job Summary** pane, view the job status. Wait a few minutes and then click **Refresh**, and keep doing this until the status changes from **Running** to **Completed**. Then, at the bottom of the **Job Summary** pane, click the **Job Output** link and view the results returned by the query, noting that the clean table does not include any rows prefixed with a **#** character.
12. Close Visual Studio without saving any files.

Create a Summary Table

1. In the C:\HDILabs\Lab02A folder, rename **Hive Query.txt** to **Hive Query.ps1** (you may need to modify the *View* options for the folder to see file extensions). Then right-click **Hive Query.ps1** and click **Edit** to open the script file in the PowerShell ISE.
2. Change the value assigned to the **$clusterName** variable to match the name of your HDInsight cluster.
3. Examine the rest of the script, noting that it performs the following tasks:
    a. Uses the **New-AzureHDInsightHiveJobDefinition** cmdlet to create a Hive job that drops and recreates a table named **webactivity** and loads it with the results of a query that retrieves grouped and aggregated data from the **cleanlog** table.
    b. Uses the **Invoke-Hive** cmdlet to query the **webactivity** table.
4. Save the script. Then on the toolbar, click **Run Script**.
   Note: If you installed Azure PowerShell after August 14th 2015, you may see the following error. You can ignore this.
   Get-AzureHDInsightJobOutput : Could not load file or assembly 'Microsoft.WindowsAzure.Storage, Version=3.0.3.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35' or one of its dependencies. The system cannot find the file specified.
5. Observe the information displayed in the PowerShell command line pane as the script runs.
6. View the results of the query, which consist of the date, count of page hits, total bytes received, and total bytes sent for every day in the log files.
7. Close the PowerShell ISE.

Querying Hive using ODBC

You can use the Hive ODBC provider to query Hive tables from ODBC clients, such as Microsoft Excel and Power BI Desktop.

Install the Hive ODBC Provider

1. Browse to http://www.microsoft.com/en-us/download/details.aspx?id=40886.
2. Download the installer that matches the 32-bit or 64-bit version of Windows installed on your computer (HiveODBC32.msi or HiveODBC64.msi).
3. Run the installer to install the Hive ODBC driver.
4. On the Start screen, view all apps. Then open the 32-bit or 64-bit ODBC Administrator tool (depending on the version of Windows you have installed) and add a System DSN based on the Microsoft Hive ODBC Driver with the following settings.
    - **Data Source Name**: Hive
    - **Description**: Hive tables in HDInsight
    - **Host**: The fully qualified DNS name of your HDInsight cluster (for example, *hd123456.azurehdinsight.net*)
    - **Port**: 443
    - **Database**: Default
    - **Hive Server Type**: Hive Server 2
    - **Authentication Mechanism**: Windows Azure HDInsight Service
    - **User Name**: The HTTP user name for your cluster
    - **Password**: The password for your HTTP user name

Query Hive from Power BI Desktop

1. Start Power BI Desktop and close the welcome page if it opens.
   Note: Power BI Desktop is the released version of the Power BI Designer preview tool used in the demonstrations for this course. The tool has been renamed and updated, and looks cosmetically different from the preview version; but still provides the same functionality as shown in the demonstrations.
2. On the **Home** tab, in the **Get Data** list, click **More**.
3. In the **Get Data** page, click **Other**. Then select **ODBC** and click **Connect**.
4. In the **From ODBC** page, enter the following details, and then click **OK**.
    - **Connection string**: DSN=Hive;
    - **SQL statement**: SELECT * FROM webactivity ORDER BY log_date;
5. If you are prompted for credentials, on the **Database** page, enter the following details, and click **Connect**.
    - **Username**: The HTTP user name for your cluster
    - **Password**: The password for your HTTP user name
6. Wait for the data preview to be displayed (this may take a minute or so), and then click **Load**.
7. In the **Fields** pane, expand the **Query1** table, and select the **page_hits** and **log_date** fields. This creates a column chart.
8. Resize the chart so that it fills the report.
9. Review the chart. Then close Power BI Desktop without saving your changes.

Cleaning Up

Now that you have finished this lab, you can delete the HDInsight cluster and storage account.

Note: If you are proceeding straight to the next lab, omit this task and use the same cluster in the next lab. Otherwise, follow the steps below to delete your cluster and storage account.

Delete the HDInsight Cluster

If you no longer need the HDInsight cluster used in this lab, you should delete it to avoid incurring unnecessary costs (or using credits in a free trial subscription).

1. In the Azure portal, click the **HDInsight** tab.
2. Select the row containing your HDInsight cluster, and then at the bottom of the page, click **Delete**. When prompted to confirm the deletion, click **Yes**.
3. Wait for your cluster to be deleted, and then click the **Storage** tab, and if necessary refresh the browser to view the storage account that was created with your cluster.
4. Select the row containing the storage account, and then at the bottom of the page, click **Delete**. When prompted to confirm the deletion, enter the storage account name and click **OK**.
5. Close the browser.