Procesando Big Data con Azure Data Lake

Lab 2 – Usando U-SQL Catalog

Introducción

En esta práctica, creará una base de datos Azure Data Lake para contener algunas tablas y vistas para procesar big data y reportes.

Prerrequisitos

Para completar esta práctica, usted necesita

- Un web browser
- Un Microsoft account
- Un Microsoft Azure subscription
- Un ordenador Windows, Linux, o Mac OS X

Creando un Azure Data Lake Analytics Account

En este ejercicio, creará una cuenta de Azure Data Lake Analytics y un de Azure Data Lake store asociado a esta cuenta.

Creando un Azure Data Lake Analytics Account

Antes de poder utilizar Azure Data Lake Analytics para procesar datos, debe crear una cuenta de Azure Data Lake Analytics, y asóciarla con al menos un Azure Data Lake store.

1. En un navegador web, vaya a http://portal.azure.com y, si se le solicita, inicie sesión con la Cuenta de Microsoft que está asociada con su suscripción de Azure.
2. En el portal de Microsoft Azure, en el Hub Menú, haga clic en New. Luego, en Intelligence and analytics menú, haga clic en Data Lake Analytics.
3. En New Data Lake Analytics Account, ingrese la siguiente configuración, y luego haga clic Create:
   - Name: ingrese un nombre único (¡y anótelo!)
   - Subscription: seleccione su suscripción de Azure
   - Resource Group: cree un nuevo grupo de recursos con un nombre único
   - Location: seleccione cualquier región disponible
   - Data Lake Store: cree una nueva tienda Data Lake con un nombre único (y anotelo)
   - Pin to dashboard: no seleccionado
4. En Azure Portal, vea Notifications para verificar que la implementación haya comenzado y espere a que los recursos sean creados (esto puede tomar unos minutos).

Provisionando un Source data file

En este lab, usara Azure Data Lake Analytics para procesar web server logs. Inicialmente ejecutara algunos Jobs para procesar un solo archivo

1. En el folder donde extrajo los archivos de esta práctica, expandir el iislogs folder.

2. Use un editor de texto para revisar el contenido del file 2008-01.txt, el file contiene una fila que tiene como prefijo # para indicar que esta es una fila de cabecera, y los demás registros son web server requests delimitadas por espacios.
3. En el Azure portal, vaya a la página Data Explorer del Azure Data Analytics account, debe crear un nuevo folder llamdo iislogs en el root del Azure Data Lake store.
4. Expanda el folder iislogs, luego click en Upload y carge el archivo 2008-01.txt

Creando un Azure Data Lake Database

Hemos visto que Azure Data Lake Analytics nos permite extraer y procesar datos directamente de archivos en el data lake store, podemos conseguir mejor performance y reusabilidad usando Azure Data Lake catalog para crear bases de datos para los datos

Creando una Base de datos

1. En el Azure portal, en el blade para su Azure Data Lake Analytics account, click en **New Job**
2. En el blade **New U-SQL** job, escriba **Create DB** como nombre del job en el Job Name box, luego en la ventana de código escriba lo siguiente:

```
CREATE DATABASE IF NOT EXISTS webdata;
```

3. Click **Submit Job** y observe los detalles de ejecución del job.
4. Cuando el job finalize, retorne al blade del Azure Data Lake Analytics account y haga click en **Data Explorer.**
5. En el **Data Explorer,** en el nodo **Catalog** verifique que la base de datos **webdata** está disponible además del **master** database.

Creando un Schema y una tabla

Una vez creado una base datos, podemos crear tablas

1. En el Azure portal, en el blade para su Azure Data Lake Analytics account, click en **New Job**
2. En el blade **New U-SQL** job, escriba **Create Table** como nombre del job en el Job Name box, luego en la ventana de código escriba lo siguiente:

```
USE DATABASE webdata;

CREATE SCHEMA IF NOT EXISTS iis;

CREATE TABLE iis.log

(

    date string,

    time string,

    client_ip string,

    username string,
```

```
        server_ip string,
        port int,
        method string,
        stem string,
        query string,
        status string,
        server_bytes int,
        client_bytes int,
        time_taken int?,
        user_agent string,
        referrer string,
        INDEX idx_logdate CLUSTERED (date)
    )
    DISTRIBUTED BY HASH(client_ip);
```

Este código crea un esquema llamado **iis** y una tabla llamada **log** en el **webdata** database. La tabla tiene un clustered index en la columna date, y los datos serán distribuidos en el data lake store basados en un hash de la columna **client_ip.**

3. Click en **Submit Job**, y monitoree los detalles de ejecución
4. Cuando el job haya terminado, retorne al blade del Azure Data Lake Analytics account y haga click en el **Data Explorer.**
5. En el **Data Explorer,** en el nodo Catalog expanda su account, expanda webdata database, expanda Tables, y seleccione iis.log, los detalles de la tabla incluyendo columnas e indexes se muestran aquí.

Insertando Dataos en la Tabla

La tabla que hemos creado esta inicialmente vacia, podemos usar un U-SQL job para insertar datos.

1. En el Azure portal, en el blade del Azure Data Lake Analytics account, click en la página **New Job.**
2. En el blade **New U-SQL Job**, escriba **Load Table** como nombre para el nuevo job, y en la ventana de edición de código escriba

```
USE DATABASE webdata;
@log =
    EXTRACT date string,
        time string,
```

```
            client_ip string,
            username string,
            server_ip string,
            port int,
            method string,
            stem string,
            query string,
            status string,
            server_bytes int,
            client_bytes int,
            time_taken int?,
            user_agent string,
            referrer string
        FROM "/iislogs/{*}.txt"
        USING Extractors.Text(' ', silent:true);
INSERT INTO iis.log
SELECT * FROM @log;
```

Este código lee todos los archivos con extensión .txt del **iislogs** folder en un esquema que coincide con la tabla, y luego inserta los datos extraidos en una tabla.

6. Click en **Submit Job**, y monitoree los detalles de ejecución
7. Cuando el job haya terminado, retorne al blade del Azure Data Lake Analytics account y haga click en el **Data Explorer.**
8. En el **Data Explorer,** en el nodo **Catalog** expanda su account, expanda **webdata** database, expanda **Tables**, y seleccione **iis.log**, luego click en **Query Table**
9. Modifique el default query como sigue:

```
@table = SELECT * FROM [webdata].[iis].[log]
ORDER BY date, time
FETCH FIRST 100;
OUTPUT @table
TO "/Outputs/webdata.iis.log.tsv"
USING Outputters.Tsv();
```

Este código lee la tabla y retorna los primeros 100 rows ordenados por las columnas date, time

10. Click en **Submit Job**, y monitoree los detalles de ejecución
11. Cuando el job haya terminado, click Output tab y seleccione **webdata.iis.log.tsv** para visualizar los resultados y verificar datos.

Creando y Consultando un View

Views encapsulan consultas complejas, y proveen una capa de abstracción sobre tablas. Se usan en base de datos relacionales, y son soportadas por U-SQL.

Creando un View

1. En el Azure portal, en el blade de Azure Data Analytics account, click **New Job**
2. En el blade **New U-SQL Job**, en el **Job Name** box escriba **Create View**, luego modifique el script tal como se muestra a continuación

```
USE DATABASE webdata;
CREATE VIEW iis.summary
AS
SELECT date,
     COUNT(*) AS hits,
     SUM(server_bytes) AS bytes_sent,
     SUM(client_bytes) AS bytes_received
FROM iis.log
GROUP BY date;
```

Este código crea un view llamado **summary** y recupera valores agregados diarios (daily aggregated values) de la tabla **log**

3. Click **Submit Job** y monitoree los detalles de ejecución.
4. Cuando el job haya terminado, retorne al bladde del Azure Data Lake Analytics account y click en el **Data Explorer.**

Consultando un View

1. En el **Data Explorer,** en el nodo **Catalog** expanda su account, expanda **webdata** database, expanda **Views**, y seleccione **iis.summary**, luego click en **Query View**
2. Modique el default query

```
@view = SELECT * FROM [webdata].[iis].[summary];
OUTPUT @view
TO "/Outputs/webdata.iis.summary.tsv"
ORDER BY date
USING Outputters.Tsv();
```

Este código consulta el view y retorna el resultado ordenado por la columna date.

3. Click **Submit Job** y monitoree los detalles de ejecución.
4. Cuando el job haya terminado, click **Output** tab y seleccione **webdata.iis.summary.tsv** para visualizar los resultados y verificar datos.
5. Descargue el output file y ábralo en un text editor o spreadsheet app, y note que cada fila en los datos contiene un daily summary de hits, bytes sent, and bytes received desde enero a junio.

Usando Table-Valued Functions

Creando un Table-Valued Function

Table-values functions proveen otra forma de encapsular una consulta para retornar un rowset. Adicionalmente, estas funciones pueden incluir parámetros lo que los hace mas flexibles que los views en algunos escenarios

1. En el Azure portal, en el blade de Azure Data Analytics account, click **New Job**
2. En el blade **New U-SQL Job**, en el **Job Name** box escriba **Create TVF**, luego modifique el script tal como se muestra a continuación

```
USE DATABASE webdata;

CREATE FUNCTION iis.summarizelog(@Year int, @Month int)
RETURNS @summarizedlog TABLE
(
date string,
hits long?,
bytes_sent long?,
bytes_received long?
)
AS
BEGIN
@summarizedlog =
SELECT date,
     hits,
     bytes_sent,
     bytes_received
FROM iis.summary
WHERE DateTime.Parse(date).Year == @Year
     AND DateTime.Parse(date).Month == @Month;
END;
```

   Este código define una función llamada **summarizelog** que recupera datos a partir del **summary** view para un mes y un año especificados como parametros
5. Click **Submit Job** y monitoree los detalles de ejecución.
6. Cuando el job haya terminado, retorne al bladde del Azure Data Lake Analytics account y click en el **Data Explorer.**

Consultando un Table-Valued Function

1. En el **Data Explorer,** en el nodo Catalog expanda su account, expanda **webdata** database, expanda **Table values Functions**, y verifique que **iis.summarizelog** este definido
2. En el blade **New U-SQL Job**, en el **Job Name** box escriba **Query TVF**, luego modifique el script tal como se muestra a continuación

```
USE DATABASE webdata;
@june = iis.summarizelog(2008, 6);
```

```
OUTPUT @june
TO "/Outputs/june.csv"
ORDER BY date
USING Outputters.Csv();
```

Este código invoca a la función summarizelog, especificando los parámetros para junio 2008, retorna el resultado ordenado por date.

3. Click **Submit Job** y monitoree los detalles de ejecución.
4. Cuando el job haya terminado, click **Output** tab y seleccione **june.csv** para visualizar los resultados y verificar datos.

Usando Procedures

Creando un Procedure

Procedures son una forma de encapsular tareas, como extraer datos de archivos e insertarlos en tablas.

1. En el Azure portal, en el blade de Azure Data Analytics account, click **New Job**
2. En el blade **New U-SQL Job**, en el **Job Name** box escriba **Create Procedure**, luego modifique el script tal como se muestra a continuación

```
USE DATABASE webdata;
CREATE PROCEDURE iis.LoadLog (@File string)
AS
BEGIN
@log =
     EXTRACT date string,
          time string,
          client_ip string,
          username string,
          server_ip string,
          port int,
          method string,
          stem string,
          query string,
          status string,
          server_bytes int,
          client_bytes int,
          time_taken int?,
          user_agent string,
          referrer string
     FROM @File
     USING Extractors.Text(' ', silent:true);
INSERT INTO iis.log
SELECT * FROM @log;
END;
```

Este código crea un procedure llamado **LoadLog** que carga datos a partir de archivos en un path, a la tabla **log.**

3. Click **Submit Job** y monitoree los detalles de ejecución.
4. Cuando el job haya terminado, retorne al bladde del Azure Data Lake Analytics account y click en el **Data Explorer.**

Ejecutando un Procedure

1. En el Data Explorer, dentro de Storage Account, navegie al isslogs folder que contiene los logs de enero a junio.
2. Click Upload, navegue al folder July en el directorio donde extrajo los files de la practica y suba el archivo 2008-07.txt
3. En el Data Explorer, en el nodo Catalog expanda su cuenta,expanda webdata, Procedures y seleccione iis.LoadLog, luego click en **Run Procedure**
4. En el editor modifique el codigo

```
[webdata].[iis].[LoadLog]("/iislogs/2008-07.txt");
```

Este código usa el **LoadLog** para cargar datos del archivo **2008-07.txt** en la tabla log

5. Click **Submit Job** y monitoree los detalles de ejecución.
6. Cuando el job haya terminado, retorne al bladde del Azure Data Lake Analytics account y click en el **New Job.**
7. En el New U-SQL Job blade, escriba **Get July** en el Job Name
8. En el editor escriba el siguiente codigo

```
USE DATABASE webdata;
@july = iis.summarizelog(2008, 7);
OUTPUT @july
TO "/Outputs/july.csv"
ORDER BY date
USING Outputters.Csv();
```

Este código invoca una función, especificando parámetros para Julio 2008, y retorna el output ordenado por la columna date.

9. Click **Submit Job** y monitoree los detalles de ejecución.
10. Cuando el job termine, click **Output** y seleccione july.csv para visualizar los resultados.