

Cortana Analytics – HDInsight Sensor Data Lab with Power BI

Contents

[Lab Overview 1](#_Toc447033952)

[Prerequisites 1](#_Toc447033953)

[Sensor Data Loaded into Windows Azure Storage Blob 2](#_Toc447033954)

[Creating Hive Tables to Query the Sensor Data in the Windows Azure Blob Storage 2](#_Toc447033955)

[Creating Hive Queries over Sensor Data 3](#_Toc447033956)

[Complete Query 3](#_Toc447033957)

[Loading Data into Excel 4](#_Toc447033958)

[Visualizing Data 8](#_Toc447033959)

[Summary 14](#_Toc447033960)

# Lab Overview

Many personal and commercial devices now contain sensors, which collect information from the physical world. For example, most phones have a GPS, fitness devices track how many steps you've taken, and thermostats can monitor the temperature of a building.

In this tutorial, you'll learn how HDInsight can be used to process historical data produced by heating, ventilation, and air conditioning (HVAC) systems to identify systems that are not able to reliably maintain a set temperature. You will learn how to:

* Refine and enrich temperature data from buildings in several countries
* Analyze the data to determine which buildings have problems maintaining comfortable temperatures (actual recorded temperature vs. temperature the thermostat was set to)
* Infer reliability of HVAC systems used in the buildings
* Visualize the data in Microsoft Power BI Desktop

# Prerequisites

There are a few things you need prior to coming to this lab:

* Load the files *building.csv* and *HVAC.csv* into your HDInsight Storage, noting the location of those files.
* To export the analyzed data to Power BI Desktop, you must meet the following requirements:
* You must have Microsoft Power BI Desktop installed.
* You must have Microsoft Hive ODBC Driver to import data from Hive into Power BI Desktop. Use the 64-bit version for Microsoft Power BI Desktop: <http://www.microsoft.com/en-us/download/details.aspx?id=40886>

# Sensor Data Loaded into Windows Azure Storage Blob

The files at the URLs below contain the temperature that the HVAC system was set to, as well as the actual recorded temperature. The files also contain building metadata, such as the location and HVAC system information. We have loaded this data into the Windows Azure Storage Blob for this cluster.

The data stored in Windows Azure Storage Blob can be accessed using the **File Browser** tab at the top of this page. The data for this sample can be found under the **Resources** path for this class. Change the paths

*wasb://bwoodyhdi@bwoodysa.blob.core.windows.net/HdiSamples/SensorSampleData/hvac/* and *wasb://bwoodyhdi@bwoodysa.blob.core.windows.net/HdiSamples/SensorSampleData/hvac/* to the location where you have loaded your files.

Sensor data from 20 large buildings in the world- wasb://bwoodyhdi@bwoodysa.blob.core.windows.net/HdiSamples/SensorSampleData/hvac/

Metadata about the 20 buildings - wasb://bwoodyhdi@bwoodysa.blob.core.windows.net/HdiSamples/SensorSampleData/building/

# Creating Hive Tables to Query the Sensor Data in the Windows Azure Blob Storage

The following Hive statements create external tables that allow Hive to query data stored in Azure Blob Storage. External tables preserve the data in the original file format, while allowing Hive to perform queries against the data within the file. In this case, the data is stored in the files as comma separated values (CSV).

The Hive statements below create two new tables, named **hvac** and **building**, by describing the fields within the files, the delimiter (comma) between fields, and the location of the file in Azure Blob Storage. This will allow you to create Hive queries over your data. *(Don't run this, just read it)*

## Create HVAC Table:

DROP TABLE IF EXISTS hvac;

--create the hvac table on comma-separated sensor data

CREATE EXTERNAL TABLE hvac(hvacdate STRING, time STRING, targettemp BIGINT,

actualtemp BIGINT,

system BIGINT,

systemage BIGINT,

buildingid BIGINT)

ROW FORMAT DELIMITED FIELDS TERMINATED BY ','

STORED AS TEXTFILE LOCATION

'wasb://bwoodyhdi@bwoodysa.blob.core.windows.net/HdiSamples/SensorSampleData/hvac/';

## Create building table:

DROP TABLE IF EXISTS building;

--create the building table on comma-separated building data

CREATE EXTERNAL TABLE building(buildingid BIGINT, buildingmgr STRING,

buildingage BIGINT,

hvacproduct STRING,

country STRING)

ROW FORMAT DELIMITED FIELDS TERMINATED BY ','

STORED AS TEXTFILE LOCATION 'wasb://bwoodyhdi@bwoodysa.blob.core.windows.net/HdiSamples/SensorSampleData/building/';

# Creating Hive Queries over Sensor Data

The following Hive queries (which you can run in Ambari) selects temperatures from your HVAC data, looking for temperature variations (see the query below). Specifically, the difference between the target temperature the thermostat was set to and the recorded temperature. If the difference is greater than 5, the **temp\_diff** column will be set to 'HOT',or 'COLD' and **extremetemp** will be set to 1; otherwise, **temp\_diff** will be set to ‘NORMAL’ and **extremetemp** will be set to 0.

The queries will write the results into two new tables: **hvac\_temperatures** and **hvac\_building** (see the CREATE TABLE statements below). The **hvac\_building** table will contain building information such as the manager, building age, and the HVAC system for buildings, and will also be used to look up temperature data for the building through the JOIN with the **hvac\_temperatures** table. *(Don't run this, just read it)*

## Create hvac\_temperature table:

DROP TABLE IF EXISTS hvac\_temperatures;

--create the hvac\_temperatures table by selecting from the hvac table

CREATE TABLE hvac\_temperatures AS

SELECT \*, targettemp - actualtemp AS temp\_diff,

IF((targettemp - actualtemp) > 5, 'COLD',

IF((targettemp - actualtemp) < -5, 'HOT', 'NORMAL')) AS temprange,

IF((targettemp - actualtemp) > 5, '1', IF((targettemp - actualtemp) < -5, '1', 0)) AS extremetemp

FROM hvac;

## Create hvac\_building table:

DROP TABLE IF EXISTS hvac\_building;

--create the hvac\_building table by joining the building table and the hvac\_temperatures table

CREATE TABLE hvac\_building AS

SELECT h.\*, b.country, b.hvacproduct, b.buildingage, b.buildingmgr

FROM building b JOIN hvac\_temperatures h ON b.buildingid = h.buildingid;

# Complete Query

This is a combination of the above scripts - **Submit** this to your cluster in Ambari to create the Hive tables and execute the queries that analyze the sensor data stored in Windows Azure Blob Storage. After submitting the job, check that it runs and continue to **Loading Data into Excel**.

DROP TABLE IF EXISTS hvac;

--create the hvac table on comma-separated sensor data

CREATE EXTERNAL TABLE hvac(hvacdate STRING, time STRING, targettemp BIGINT,

actualtemp BIGINT,

system BIGINT,

systemage BIGINT,

buildingid BIGINT)

ROW FORMAT DELIMITED FIELDS TERMINATED BY ','

STORED AS TEXTFILE LOCATION 'wasb://bwoodyhdi@bwoodysa.blob.core.windows.net/HdiSamples/SensorSampleData/hvac/';

DROP TABLE IF EXISTS building;

--create the building table on comma-separated building data

CREATE EXTERNAL TABLE building(buildingid BIGINT, buildingmgr STRING,

buildingage BIGINT,

hvacproduct STRING,

country STRING)

ROW FORMAT DELIMITED FIELDS TERMINATED BY ','

STORED AS TEXTFILE LOCATION 'wasb://bwoodyhdi@bwoodysa.blob.core.windows.net/HdiSamples/SensorSampleData/building/';

DROP TABLE IF EXISTS hvac\_temperatures;

--create the hvac\_temperatures table by selecting from the hvac table

CREATE TABLE hvac\_temperatures AS

SELECT \*, targettemp - actualtemp AS temp\_diff,

IF((targettemp - actualtemp) > 5, 'COLD',

IF((targettemp - actualtemp) < -5, 'HOT', 'NORMAL')) AS temprange,

IF((targettemp - actualtemp) > 5, '1', IF((targettemp - actualtemp) < -5, '1', 0)) AS extremetemp

FROM hvac;

DROP TABLE IF EXISTS hvac\_building;

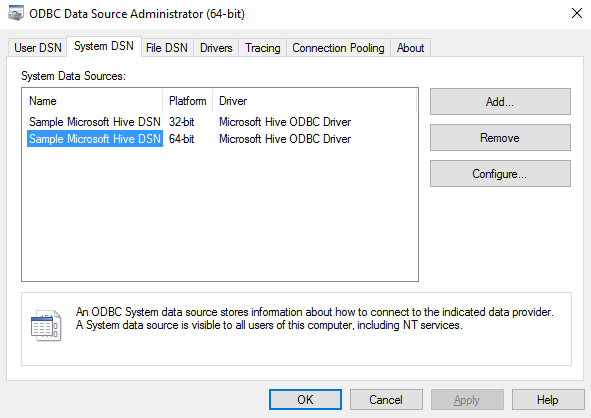
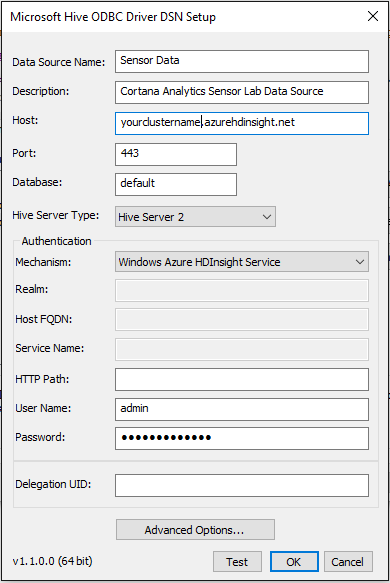
--create the hvac\_building table by joining the building table and the hvac\_temperatures table

CREATE TABLE hvac\_building AS

SELECT h.\*, b.country, b.hvacproduct, b.buildingage, b.buildingmgr

FROM building b JOIN hvac\_temperatures h ON b.buildingid = h.buildingid;

# Create Hive ODBC Data Source

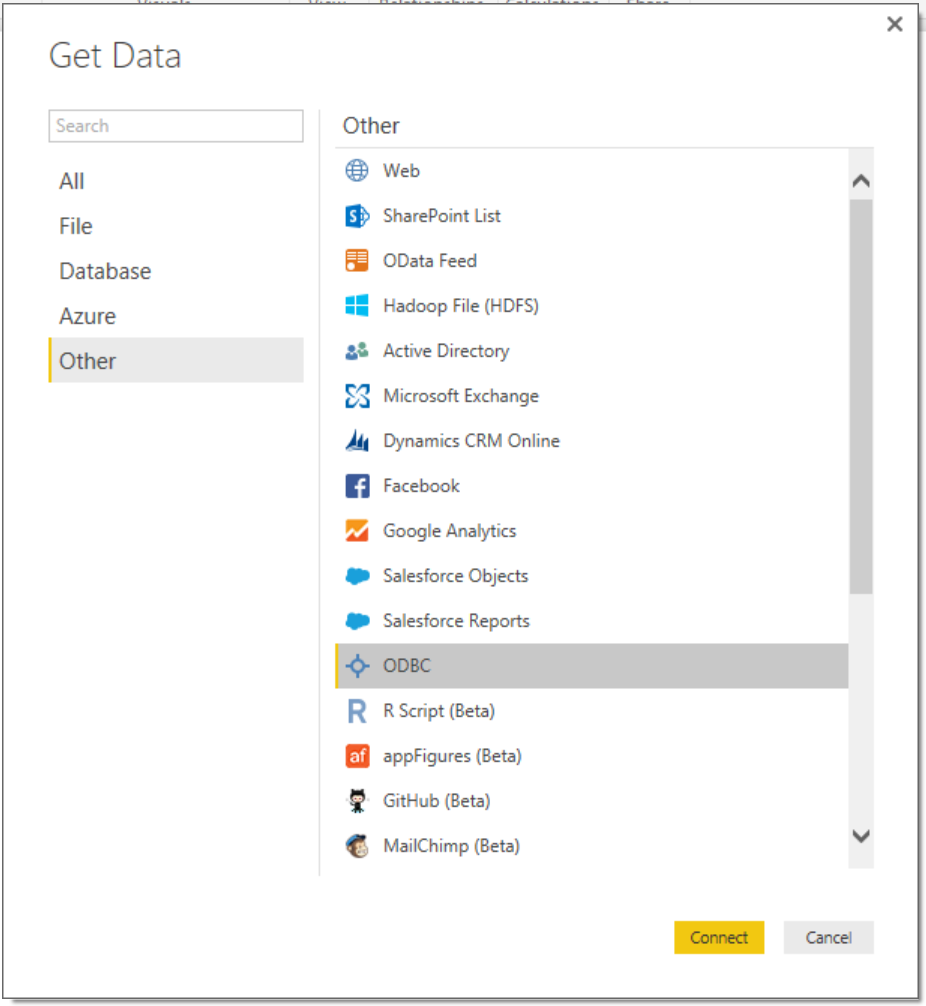
1. From the command line type: odbcad32.exe
2. Navigate to the System DSN tab
3. Click Add  
   
4. Select Microsoft Hive ODBC Driver, Click Finish
5. Enter the details  
   
6. Make sure it works by verifying with the Test button
7. Click OK

# Loading Data into Power BI Desktop

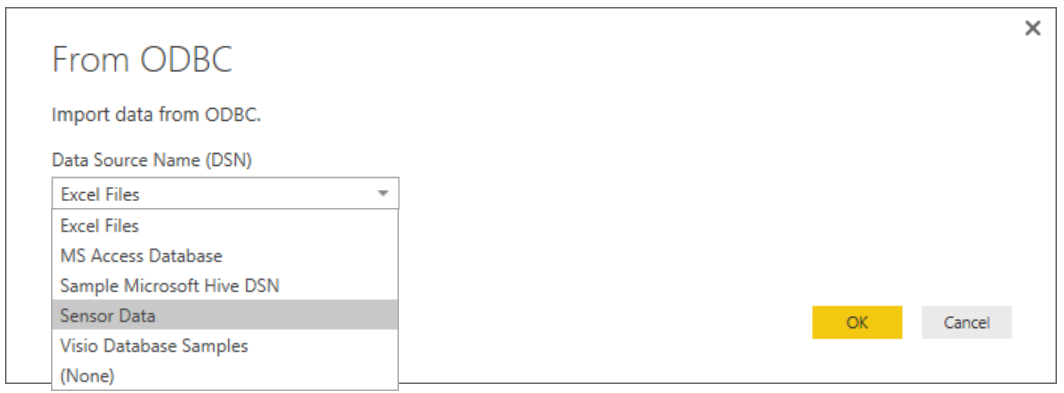
Once the job has successfully completed, you can use the [Microsoft Hive ODBC Driver](http://www.microsoft.com/en-us/download/details.aspx?id=40886) to import data from Hive into Power BI Desktop. Once you have installed the driver and setup the Data Source, use the following steps to connect to the table.

1. Open Power BI Desktop.

2. Click **Get Data,** and select Other -> ODBC



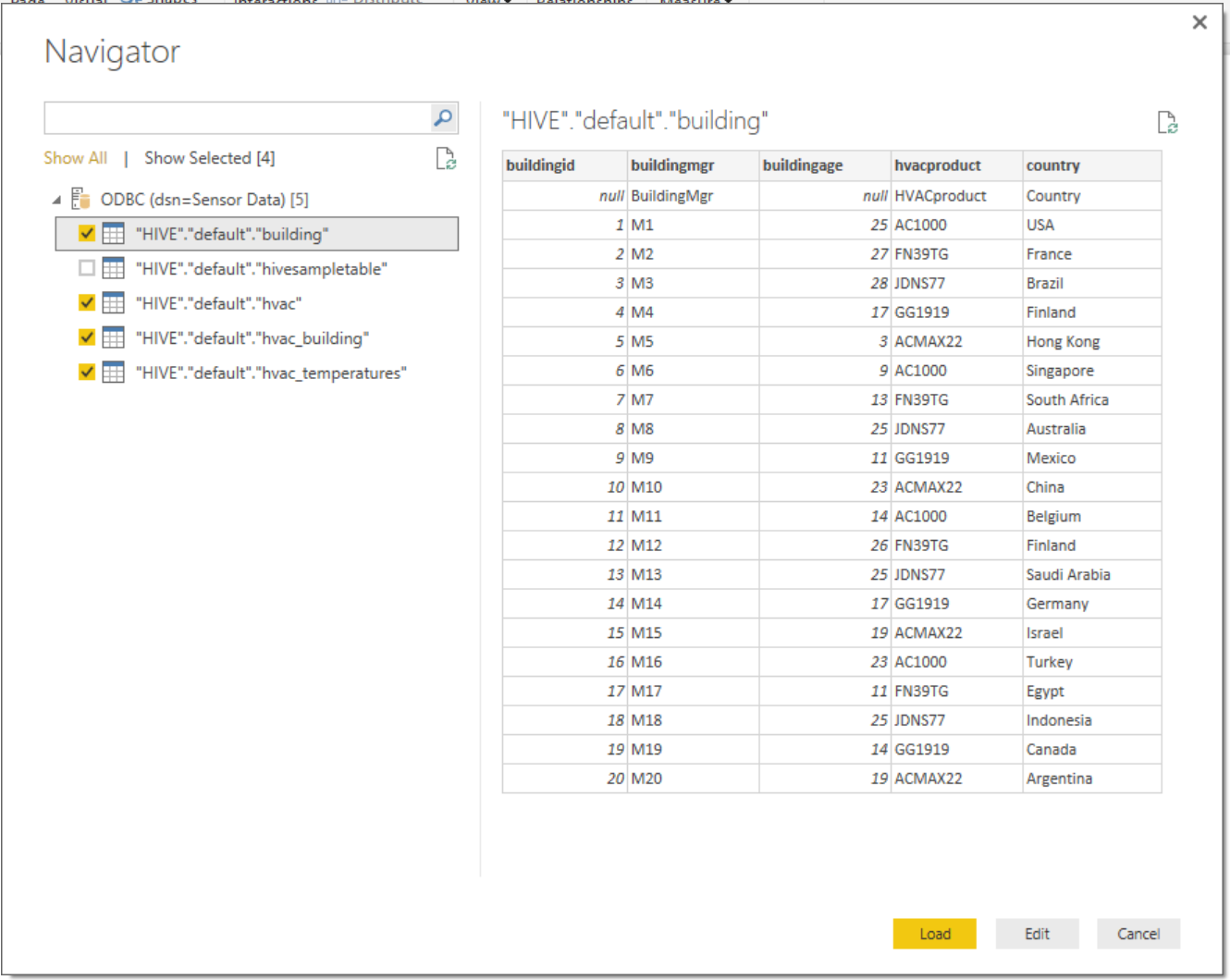
3. Click **Connect** and select the Data Source created in the previous section



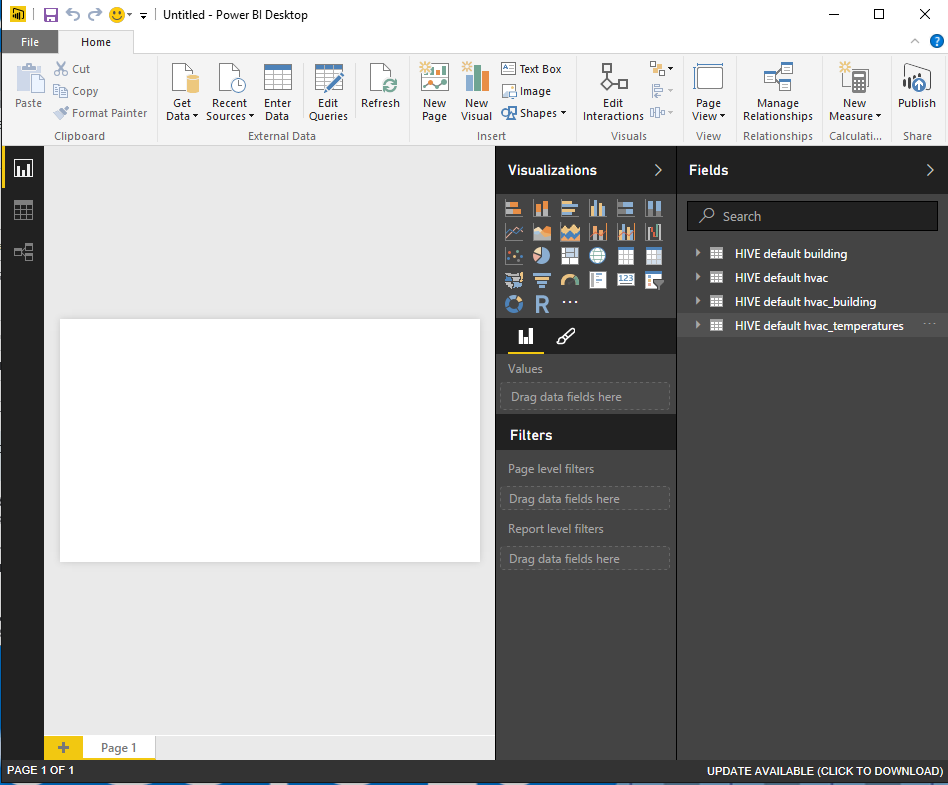
4. Click OK

5. In the Navigator, select

* building
* hvac
* hvac\_building
* hvac\_temperatures



6. Click **Load** to import the data

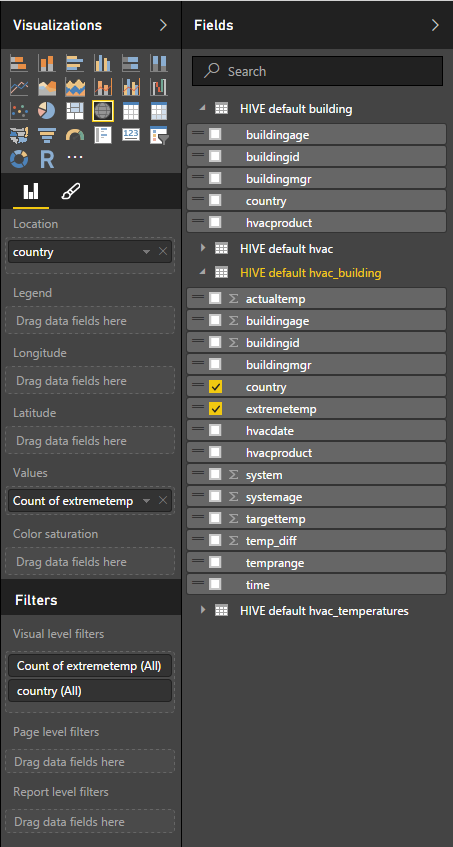
7. Then the 4 tables should show up in Power BI Desktop   


# Visualizing Data

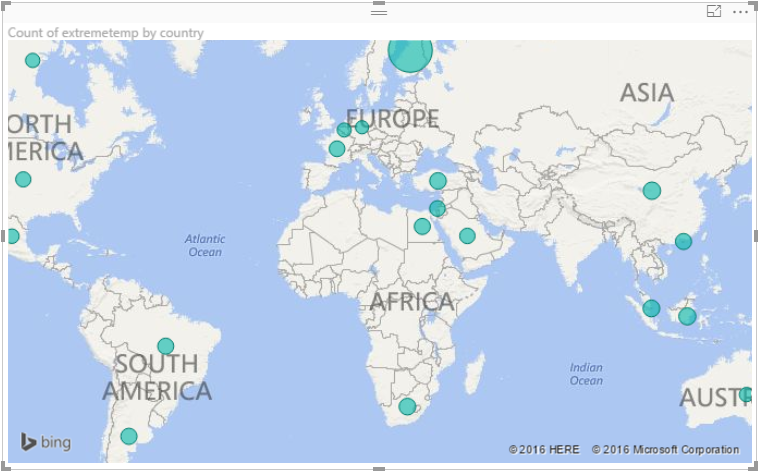
Now that the data has been imported into Power BI Desktop, you will use it to visually explore the data.

1. From the report view, drag the **country** column from the **hvac\_building** table to the blank canvas

2. Then drag **extremetemp**  from the **hvac\_building** table into the values field for the chart



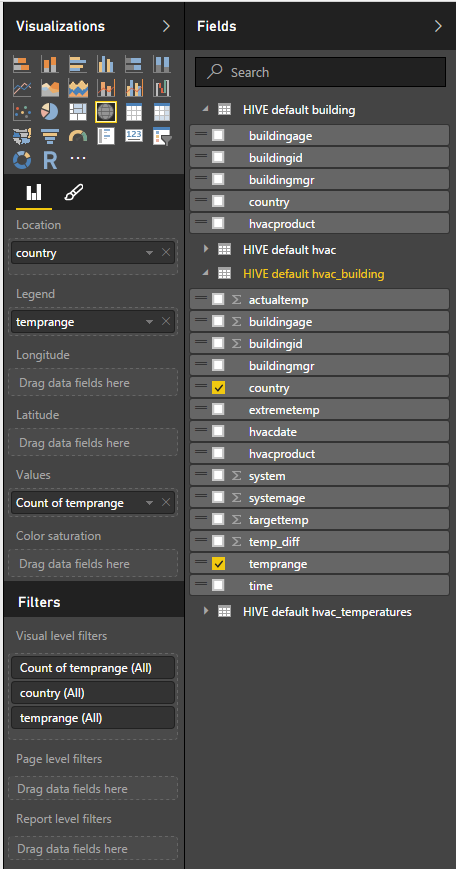
3. Make sure the **Map** visualization is selected and the result should be similar to the figure below



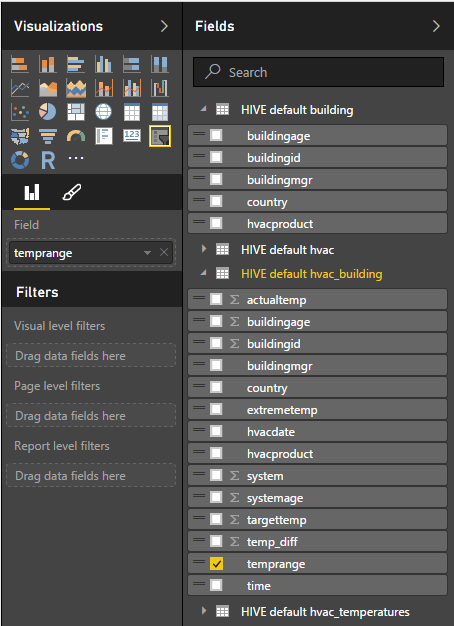
Each circle displayed on the map represents how many times the recorded temperature was greater than 5 degrees higher or lower than the target temperature. The larger the circle, the more instances of extreme temperature were recorded at this location.

To filter the data further, so you can see sites where the recorded temperature was hotter or colder than the target temperatures, perform the following steps.

1. In Power BI Desktop, remove **extremetemp** from the values field, and then drag **temprange** to **Values** and **Legend**



2. Click to the right of the map, and add a Slicer visualization. Drag the **temprange** for the field of the slicer.

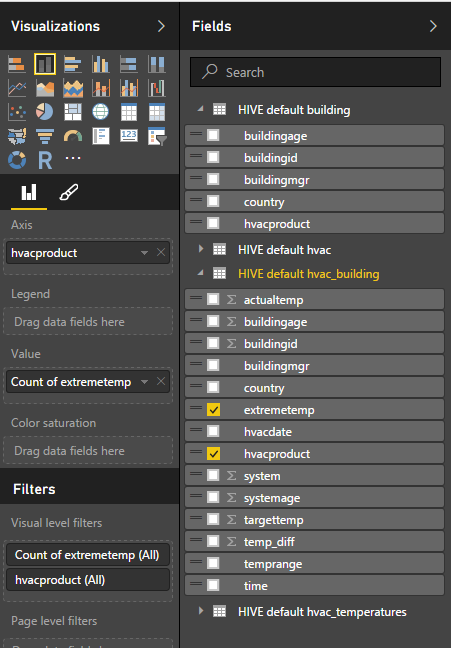


3. Now the final visualization should look similar to below



While visualizing the temperature data on a map allows you to more easily see which sites have a problem maintaining the target temperature, it does not give insight into the underlying cause. Perform the following steps to use building information to determine if the HVAC system may be source of the problem.

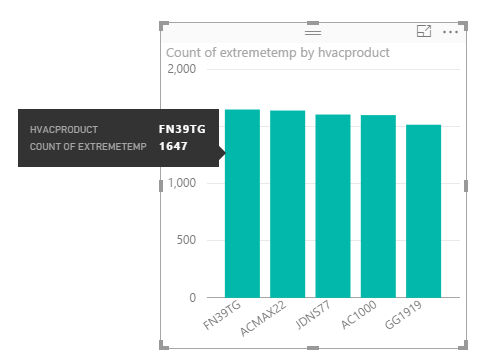
1. Drag hvacproduct to the canvas, select **Stacked Column Chart,** and drag **extremetemp** to the Value



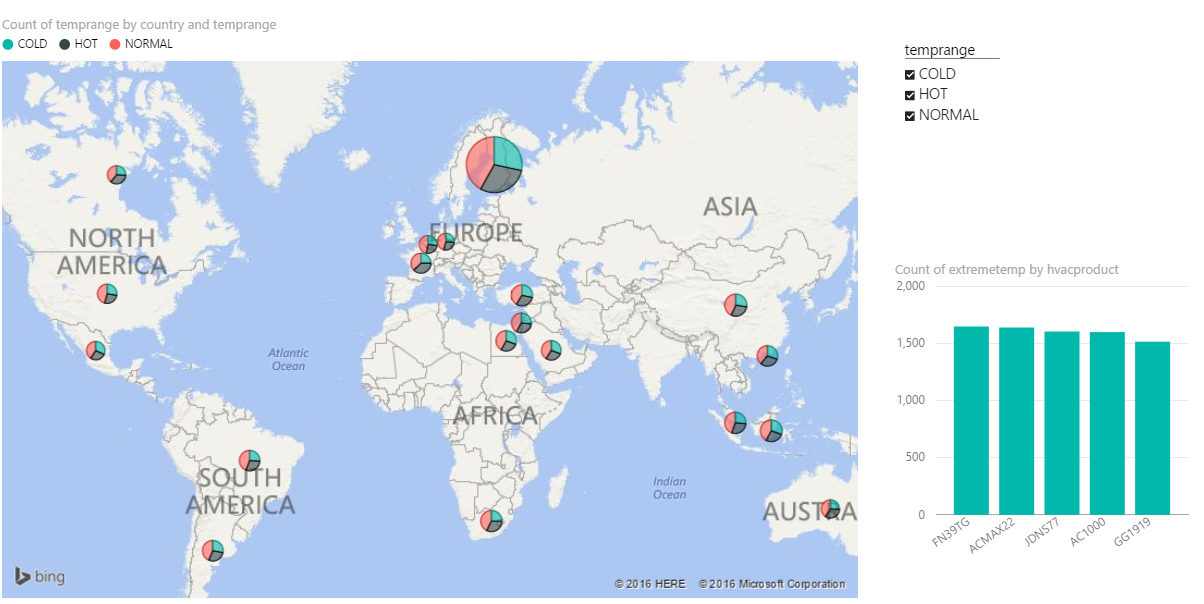
2. Once the chart is displayed, select the “…” in the upper-right corner and sort by **count of extremetemp**



3. From the chart, you can see that the FN39TG has a higher instance of extreme temperatures than other units.



The overall Report



# Summary

In this tutorial you have successfully executed an HDInsight Hive job to analyze building HVAC sensor data. To learn more, please visit the rest of our sample jobs. If you have feedback on this, or other samples, please use the **Help + Feedback** link above.

Use the following links to continue learning about how you can use Hive and Excel with HDInsight.

* [Using Hive with Hadoop in HDInsight](http://azure.microsoft.com/en-us/documentation/articles/hdinsight-use-hive/)
* [Analyze Twitter data with Hadoop in HDInsight](http://azure.microsoft.com/en-us/documentation/articles/hdinsight-analyze-twitter-data/)