**SCCM Administrator Dashboard**

**Summary:**

This dashboard will present the following information:

\* software updates status for the Collections

\* Client Health

\* Installed Server roles across your environment

\* Installed SCCM Server Roles in your SCCM Environment

\* OS verion and build numbers

\* Collection dashboard (credit for this [Matt Balzan](<https://techcommunity.microsoft.com/t5/core-infrastructure-and-security/sccm-collection-dashboard-report/ba-p/714828)>)

**How to install:**

First thing we need to do is get the collection(s) that you are targeting with your software update groups. Open the SCCMAdministratorDashboard.sql file in SQL Server Management Studio or your preferred SQL Editor. You will need to change the **FILENAME** path to your desired location. Change **NOV000015** to the collection(s) of your workstations. Change **NOV000016** to the collection(s) of your servers. Change **SMSDM003** to the collection(s) of your Windows Defender and other updates. Please note if you want to add more than one collection for your workstations add a comma between the two collections. An example of multiple collections is below.

**Example**:

vFCM.CollectionID in(‘NOV000015’,’NOV000029’,’NOV000190’)

Originally, I had the query limited to just the Cumulative Updates, I have commented those out and allowed everything that is applied to the collection. If you wish to filter the dashboard to certain updates, I have left the code in place so you can uncomment them out and it will work. The lines you want to change for this are just after the comment “This is where you would add the title of the update(s) if you want them filtered”. If you chose to filter on the Cumulative updates just delete the /\* and the \*/. If you want to filter on a list of updates, then you will need to copy the row with the like in it and paste it however many different updates you are looking to filter from. An example of multiple filters of software updates is below.

**Example**:

Vui.title like ‘%Cumulative Update for Windows%’ and

Vui.title like ‘%Adobe%’ and

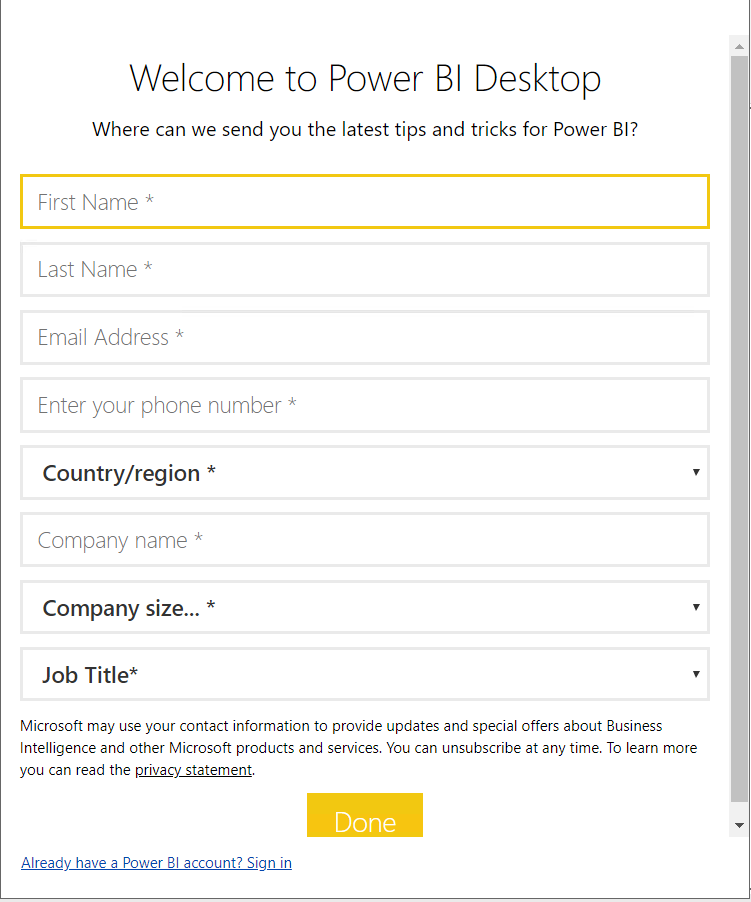
You will need to replace NOV in this SQL query to your site code. This can be done in SQL Server Management Studio or your preferred SQL Editor like we did for the FILENAME and the Collection Names.

Replace(Left(SiteSystem,CHARINDEX('\"]MSWNET:["SMS\_SITE=**NOV**"]\\',SiteSystem)-1),'["Display=\\',''),

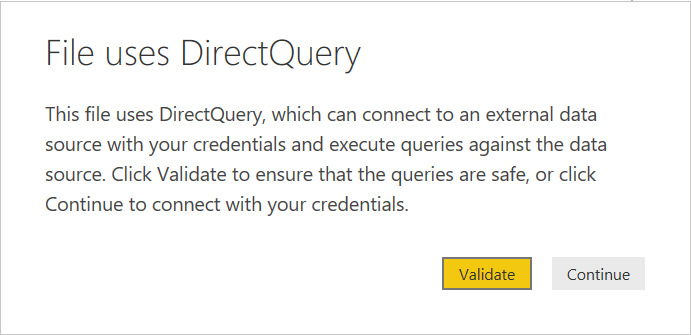
Finally, you will need to run the SCCMAdministratorDashboard.sql on the CAS/Primary database server. This script creates a database called **SCCM\_PBI\_Reporting** and then creates the tables and stored procedure needed to run the SCCMAdministratorDashboard Dashboard in Power BI so you will need admin rights to create those on the database.

**Using the dashboard**:

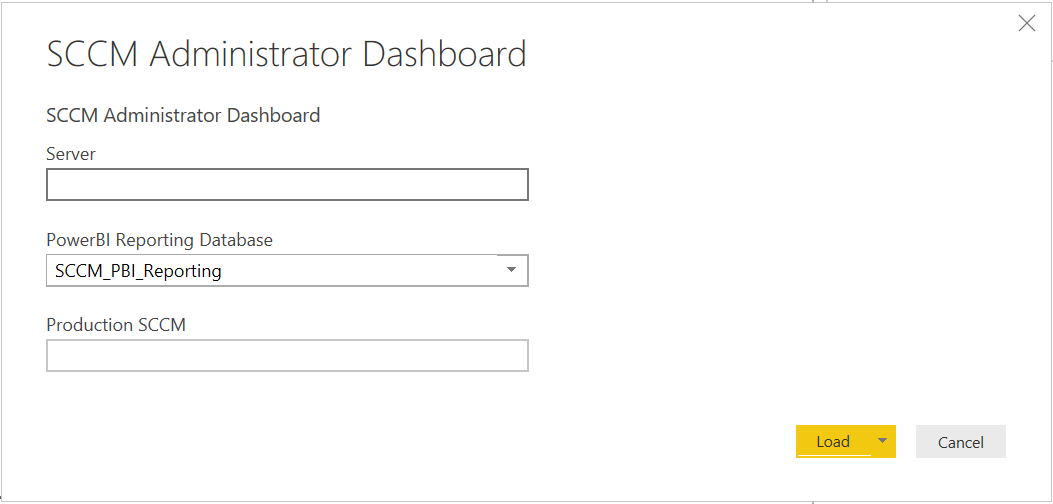
Once the script has ran successfully on, make sure you have Power BI Desktop x64 installed (Version: 2.67.5404.581 64-bit (March 2019)). Open SCCMAdministratorDashboard.pbit, the first thing that you will see is the Welcome to Power BI Desktop. Please create an account or sign in if you already have one.



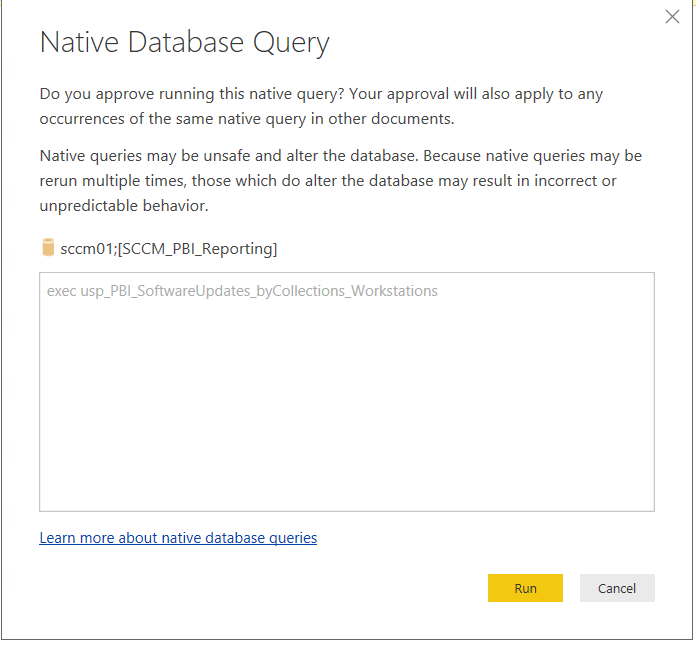
You will be prompted to validate the DirectQuery in the dashboard, these queries are the ones that pull data about the collections running in your SCCM Production environment. Click Validate if you want to check the queries or click continue to use your existing credentials.



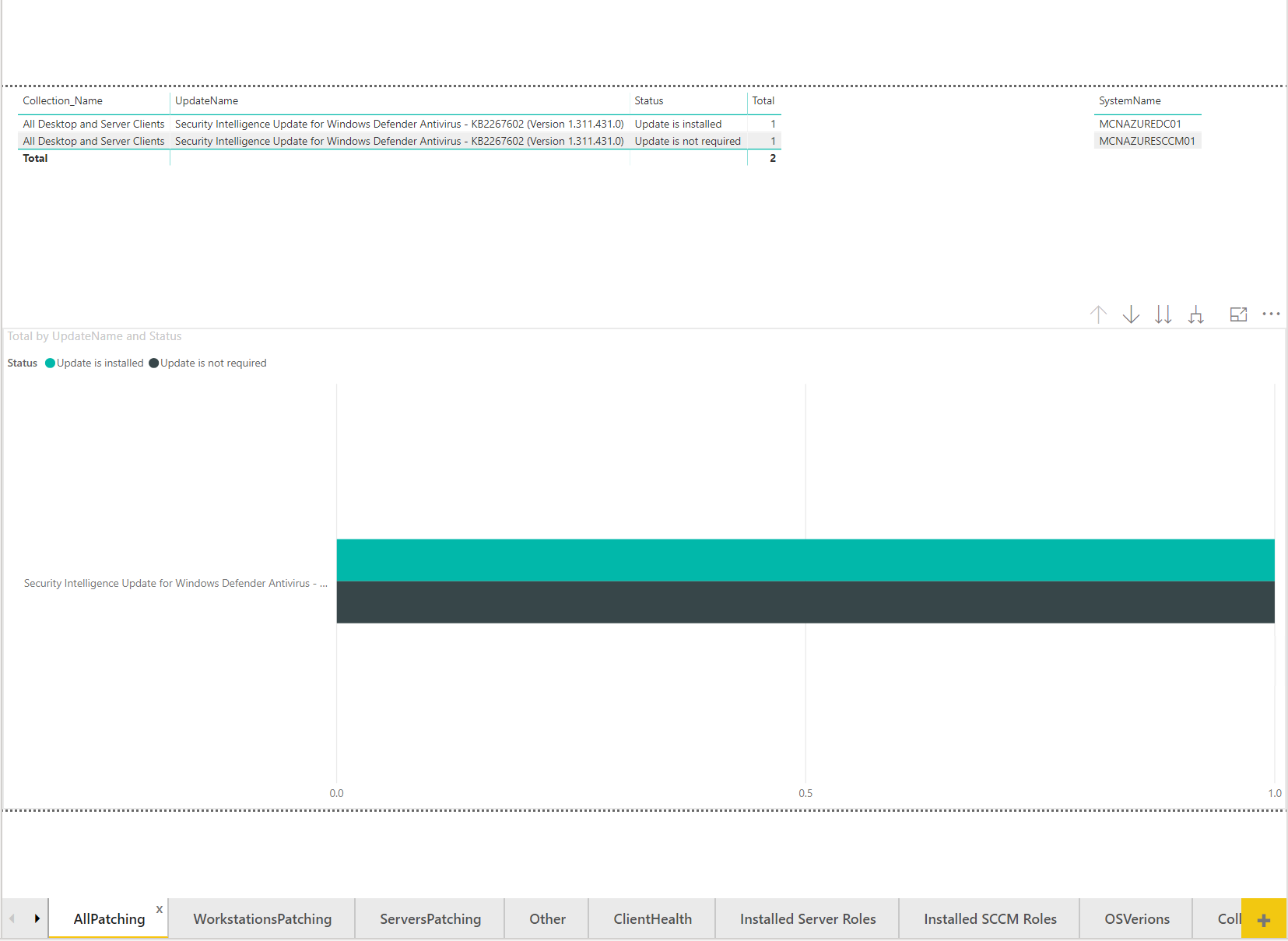
Once logged in a window will appear asking for your server, database, and Production SCCM, these are parameters for all the queries. The first parameter is your SQL Server of your CAS/Primary. The second parameter is your database of your CAS/Primary (Default is **SCCM\_PBI\_Reporting**). The last is your SCCM Production system should be CM\_???.



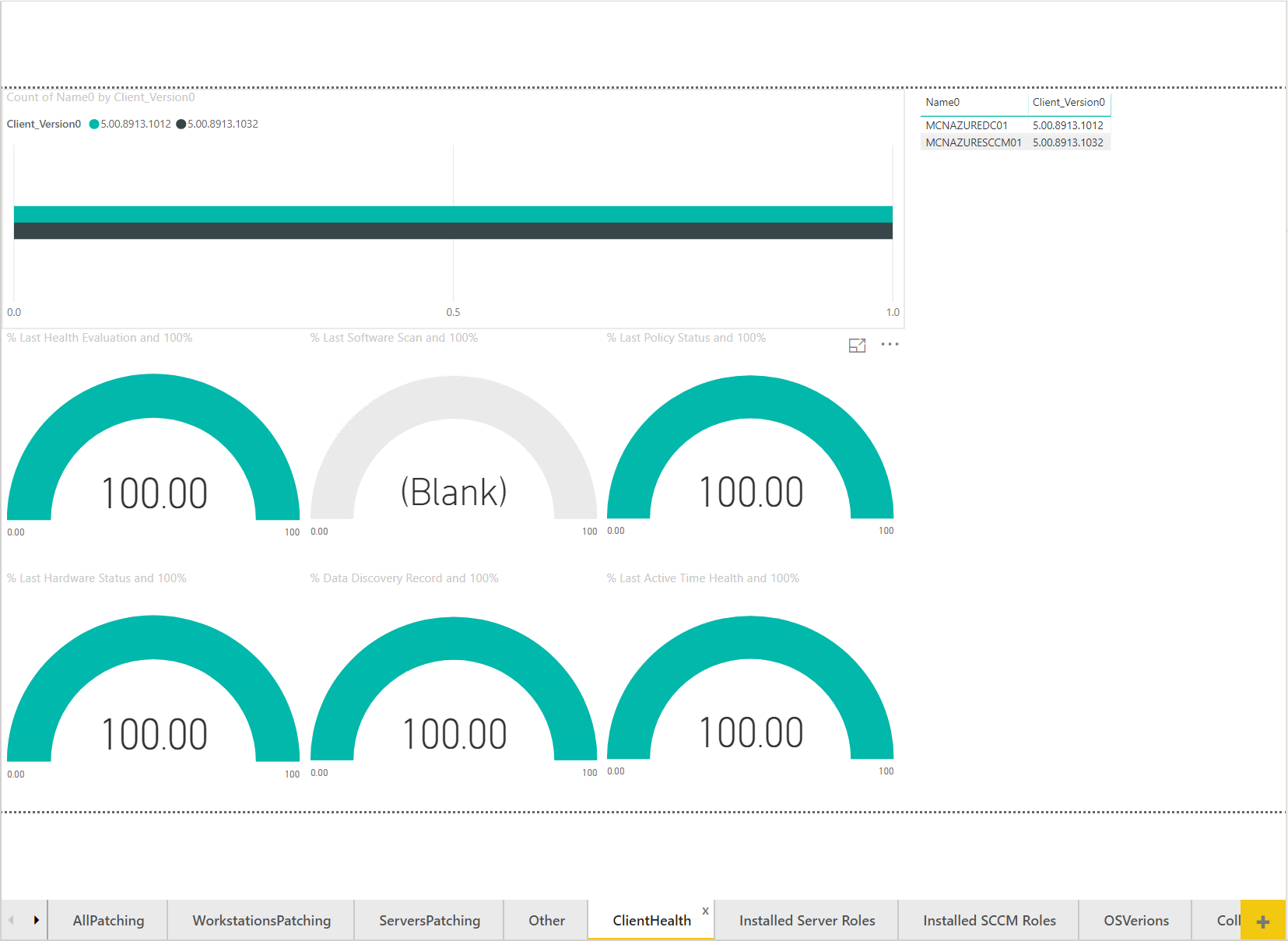
Once you click load, you may be prompted with a series message boxes stating the Native Database query needs approval to run. Click run to continue, if you do not click run the data will not load in the database. It should appear no more than 12 times (one for each query).



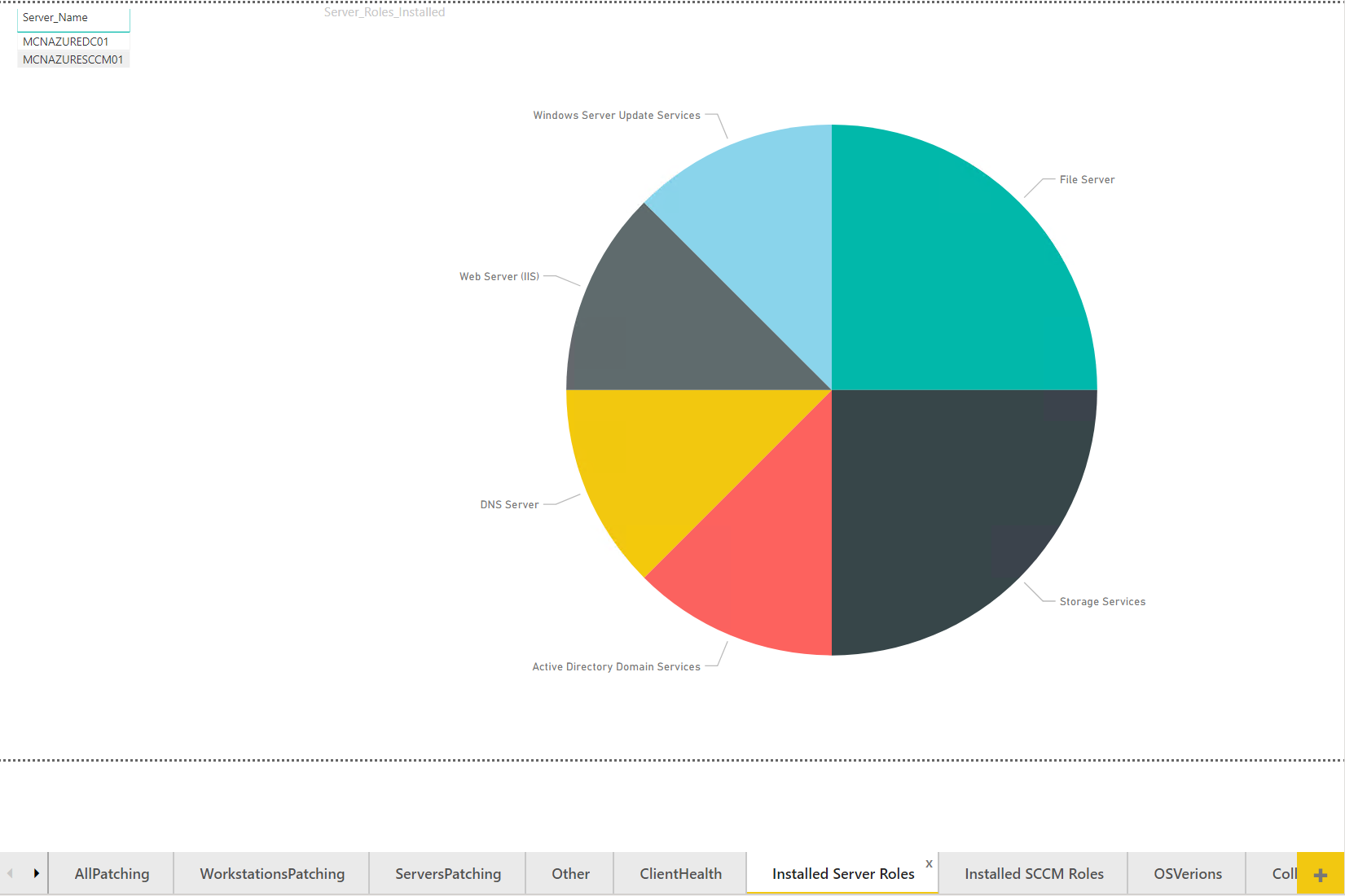
You should be able to see the dashboards if all the changes were made correctly. Your pages for “all patching”, “workstation patching”, “server patching”, and “other” should look something like this.



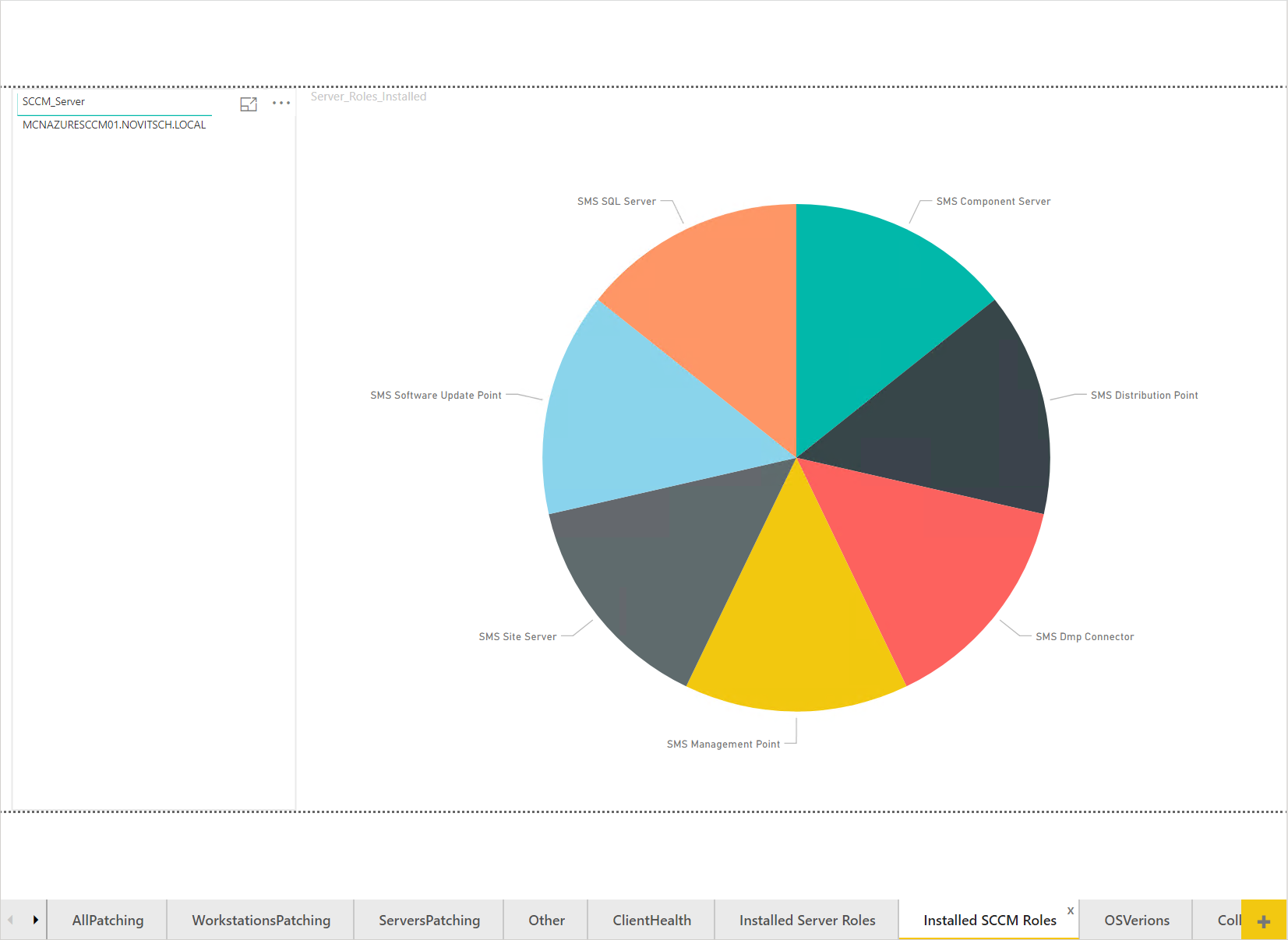
Your “ClientHealth” page should look something like this.



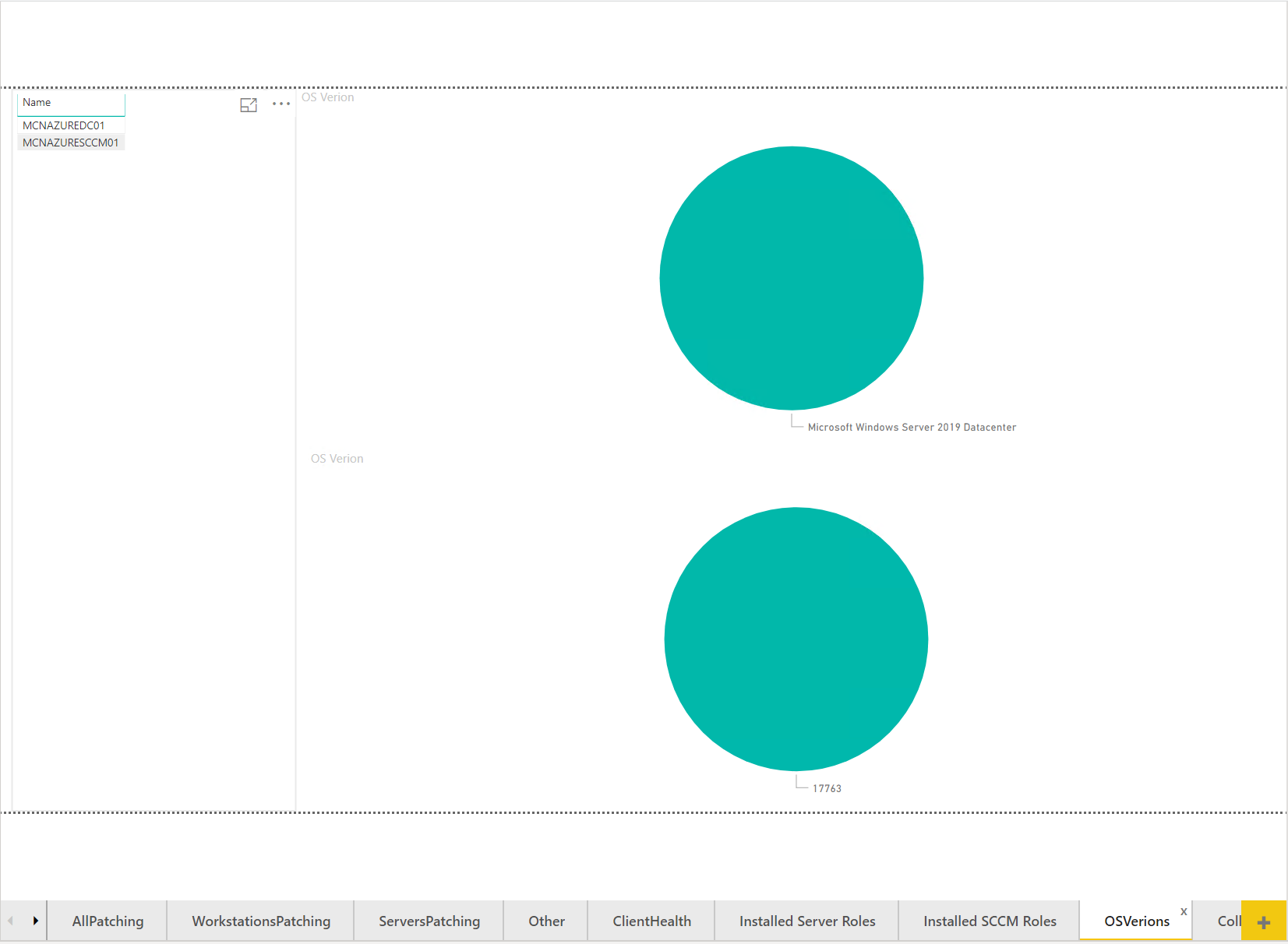
Your “Installed Server Roles” page should look something like this.



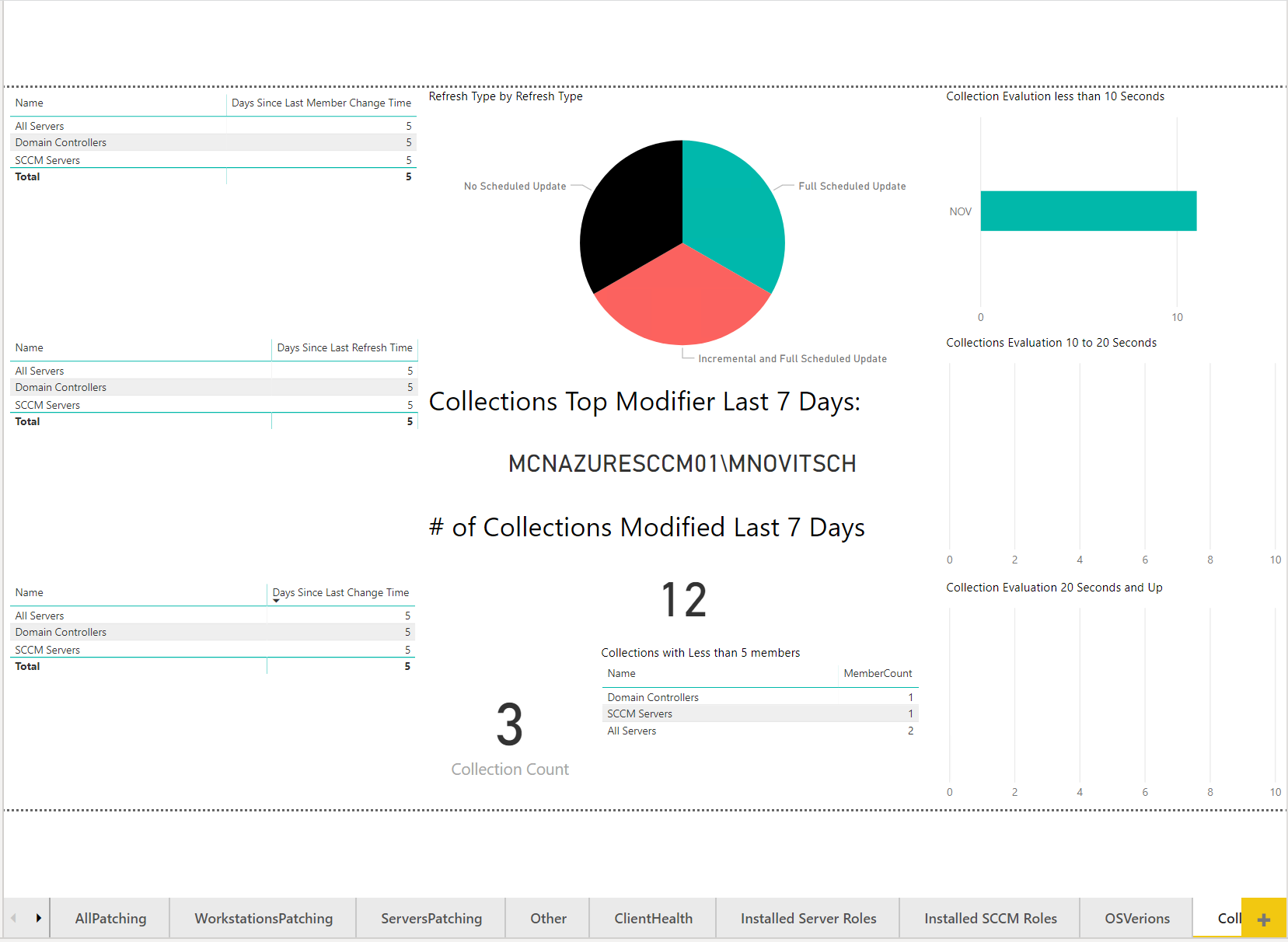
Your “Installed SCCM Roles” page should look like this.



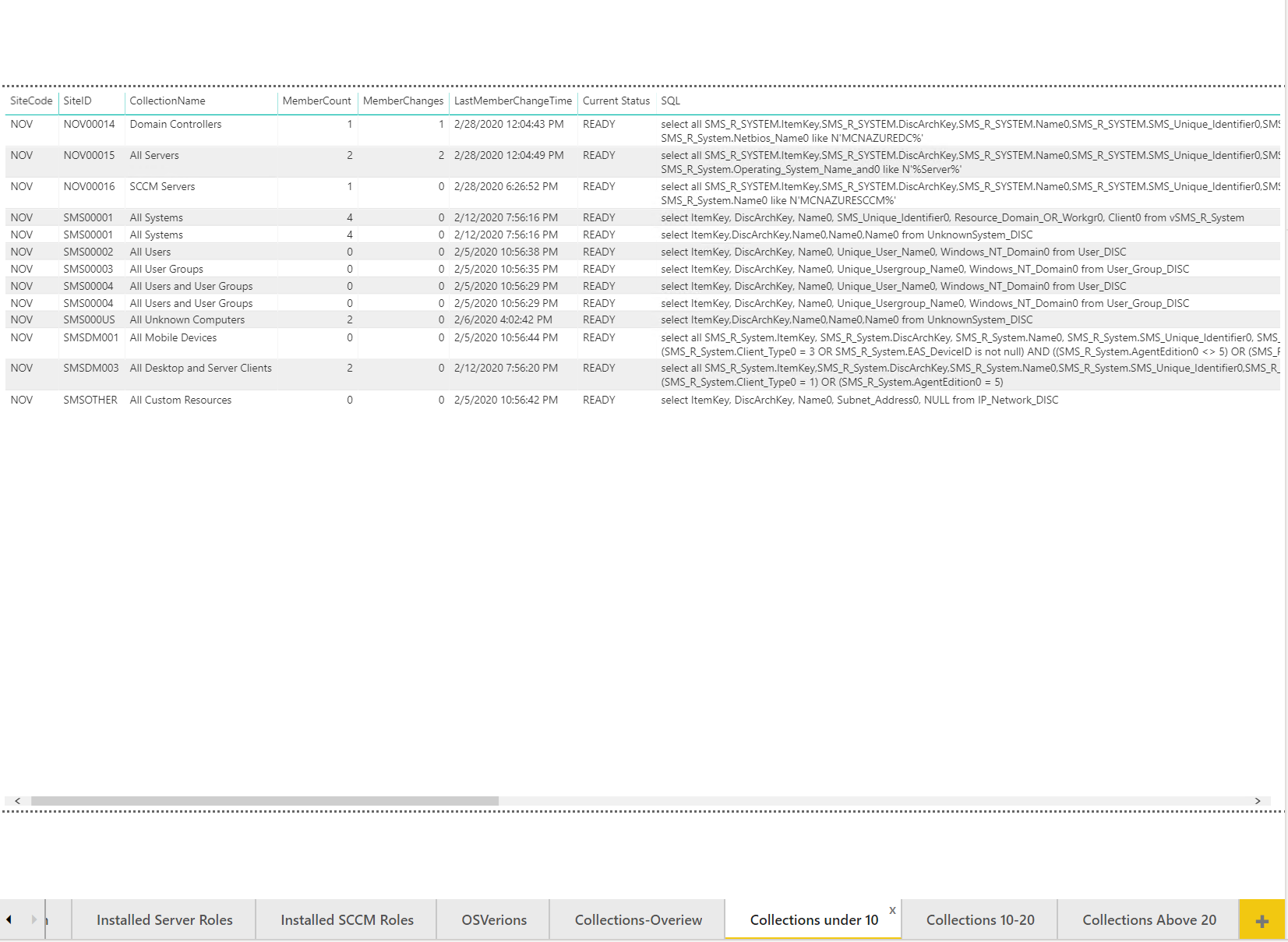
Your “OSVerions” page should look like this.



Your “Collections” page should look like this.



Your final 3 collections pages with should look like this one. First one is collections under 10 seconds, the next is between 10 and 20 seconds, and the last one is for collections running longer than 20 seconds.



Finally, please remember to save the file to your desired location so you don’t have to make the changes to the template every time. Open your file with the PBI extension next time and click the refresh button, you will be good to go then.

**How to uninstall:**

Open the SCCMAdministratorDashboard.sql file in SQL Server Management Studio or your preferred SQL Editor. Go to the bottom of the file. You will see about 10 lines that are commented out. Highlight the 10 lines and execute them. Below is what the code looks like.

--Uninstall SCCM PBI\_Reporting database

/\*

--Sets database to single user mode so it drops all other connections

USE [master]

GO

ALTER DATABASE [SCCM\_PBI\_Reporting] SET SINGLE\_USER WITH ROLLBACK IMMEDIATE

GO

--Deletes the database from SQL Server

Drop database [SCCM\_PBI\_Reporting]

\*/