**Setting Up Personal Cluster**

**Index**

1. **Prerequisites**
2. **Configuring Windows**
3. **Installing and Configuring SQL server**
4. **Installing Visual Studios**
5. **Initializing the inContact databases**

**Prerequisites**

To reduce some confusion, note a few things:

1. There is a **InContact helpdesk** that can help you with any problem you are stuck with. Even if they cannot directly help you with you issues, they will create tickets on your behalf and inform you of it so that you can move forward. Hence, when in doubt, contact the helpdesk.

email: [helpdesk@niceincontact.com](mailto:helpdesk@niceincontact.com)

Tel: +1 888 821 0062

1. There is a Global Service Desk(**GSD**) team. This team will help you with problems with your computer, installing software, networking issues and admin rights issues.
2. Depending on your particular problem, you can create tickets on the **nsc(Nice Shared Service Center** [**https://nicensc.service-now.com/**](https://nicensc.service-now.com/)**) portal**. Try your best to choose the right section while creating a ticket. If you are having doubts as to which section a ticket should be created, it is best to simply ask your manager for clarification on who you can contact for support.
3. **Get your VPN access for nicevpn.niceincontact.com domain**

-For getting your VPN access, please contact the GSD team. You can either send them an email at [nicensc@service-now.com](mailto:nicensc@service-now.com) or create a ticket on the nsc portal. Ask them to give you VPN access and add you to the "NICVPN-NICE USERS ACCESS" group to provide Incontact VPN access

1. **Get your personal UCN(UCN is just the old name of NICE) account**

-For This, send an email to the helpdesk and request them to give you an UCN account. Most likely, the helpdesk will create a ticket on the niceincontact.mysalesforce.com portal for this, which is why access to it is needed (see step 4)

1. **Get MSDN subscription**

-Send an email to the helpdesk stating that you need a MSDN subscription. CC your manager on this email. In case you already have it and it is expired, then simply ask them to renew it.

1. **Get access to niceincontact.my.salesforce.com**

Contact the CXone Salesforce team for this

1. **Get access to the following folders:**

* **\\corpfs01**
* **\\dra-dfs\DFS-Root\MyACD\DB\**
* **\\corpfs02\Latest\**

For this, send an email to inContact helpdesk. Specifically mention all of the folders

1. **Download and Install Virtual Clone Drive**

https://virtual-clonedrive.en.softonic.com/

**Note: Any other image mounting software is also ok**

**Configuring Windows**

For a lot of these steps, you will need to get in touch with the tech support team. They will need to take control of your PC to add the \_rdbuild user and install the IIS server. Before contacting them, you can setup the following things yourself.

1. **Change the “UseWUServer”**

* Go to Windows Start->Registry Editor and run the “Registry Editor” as administrator. In the “Registry Editor”, navigate to Computer\HKEY\_LOCAL\_MACHINE\SOFTWARE\Policies\Microsoft\Windows\WindowsUpdate\AU
* Double click on the “UseWUServer” and change the “value data” field to “0” (without the double quotes)

1. **Restart the “Services”**

* Go to Windows Start->Services and run it as administrator
* Scroll down to “Windows Update”, right click on it and click on it and click on “start”. If it has already started, then click on “restart”.

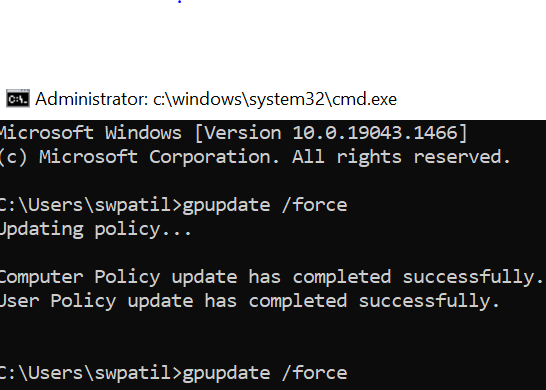
1. **Install the .NET Framework**

* Go to Windows Start, type “appwiz.cpl” and run this as an administrator. The “Programs and Features” window should automatically open.
* Select “Turn Windows features on or off” on the left side
* Select the “.NET Framework 3.5(includes .NET 2.0 and 3.0” option and click on ok. During the installation, choose the option “Let Windows Update download the files for you”(let this process run, it may take a while).
* Restart your computer

1. **Contact the tech support team to add the \_rdbuild user to the “Remote Desktop Users” and also install the IIS server**

* Please note that when the tech support person opens the “Remote Desktop Properties” and clicks on the “Add” button, that they select “Entire Directory” after hitting the “Locations” button and then in the textbox that appears below “Enter the object names to select”, they enter “ucn\\_rdbuild” and then click on the “Check Names” button.
* Ask the tech support to to add your host name to the appropriate group for adding the DNS Suffixes - to get your hostname, go to Start->cmd and run that as administrator
* Type “hostname” and give this hostname to tech support. After they have added it to the group, run the following command – “gpupdate /force”

Note: You might need to restart your pc if the above command fails. If it fails the update the policy, then restart the pc and try again. Upon successful completion, you will see the following message in the command prompt:



* While installing the IIS, make sure you follow the instructions on [acd-personal-cluster-setup/1\_ConfiguringWindows.md at master · nice-cxone/acd-personal-cluster-setup (github.com)](https://github.com/nice-cxone/acd-personal-cluster-setup/blob/master/setupGuide/1_ConfiguringWindows.md) under section 1.8(1.8.1,1.8.2)

1. **Planning installation path**

The typical development computer is equipped with two hard drives, one smaller solid state drive as the boot drive (C: drive), and one disk drive which is larger (D: or E: drive).

You should seriously consider creating some of these folders on your secondary drive. The Proj folder alone will be over 90 gigabytes in size once it is fully populated. Also, most of our production machines have the web sites, the services, and the data over on the secondary drive because the log files and other things get written to the C:</span> drive and may create a disk space shortage if the C:</span> drive already has a lot of stuff on it.

Decide where you will put these directories. The document will use  to refer to this location throughout.

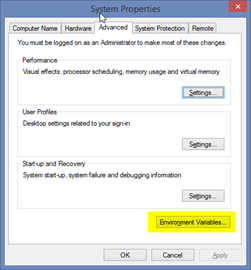
You will need to create the following directory structure on your hard drive:

* \Uploads
* \Proj
  + \Proj\LogFiles
  + \Proj\InstallLogs
  + \Proj\InDataDBInstaller

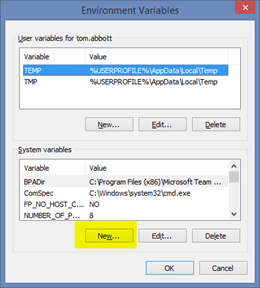
Once you have set up the Proj folder, you will need to map an environment variable LOGPATH to point to the LogFiles folder under the Proj folder (\Proj\LogFiles).

## Setting the LOGPATH environment variable

Add an environment variable called "LOGPATH" that points to your C:\Proj\LogFiles.  To do this, go to the system properties dialog. In Windows 7 and above, you can get here by holding down the Windows key and pressing the Pause/Break key. In Windows 8 and above, this will take you to System. Click Advanced System Settings on the left and Environmental Variables on the bottom right of the window that appears.

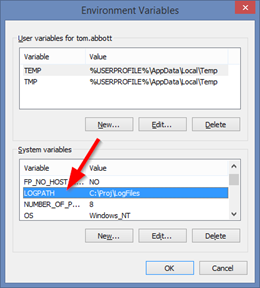


In the lower section where it says System variables, click the New… button.



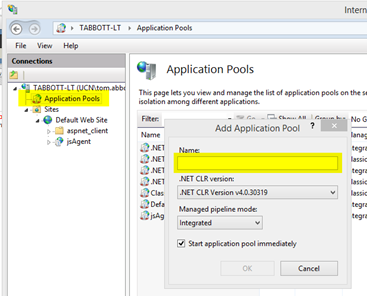
Now add the environment variable (**make sure to use \Proj\LogFiles**):

You should see it in the list now:



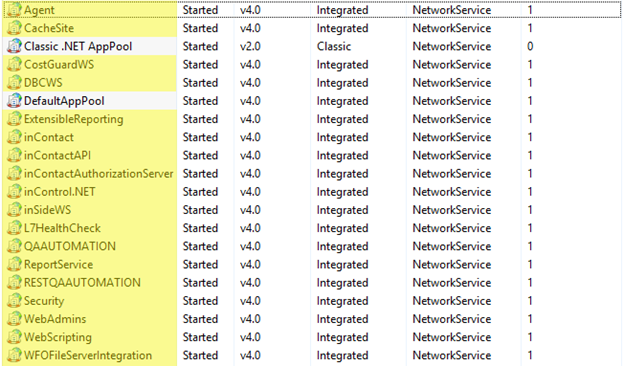
1. **Create App Pools used by inContact**

* Open the IIS manager in administrator mode
* In the left pane, right click on Application Pools, and from the context menu select Add Application Pool.



* Add the following application pools.  A lot of these you will never use, but you’ll need them anyway, just so WebAdmins doesn’t bomb when you do a full push.
  + - Agent
    - CacheSite
    - CostGuardWS
    - DBCWS
    - ExtensibleReporting
    - inContact
    - inContactAPI
    - inContactAuthorizationServer
    - inControl.NET
    - inSideWS
    - L7HealthCheck
    - ReportService
    - Security
    - WFOFileServerIntegration
    - QAAUTOMATION
    - RESTQAAUTOMATION
    - WebAdmins
    - WebScripting
* Check the App pools

You should now have a list of app pools like so:

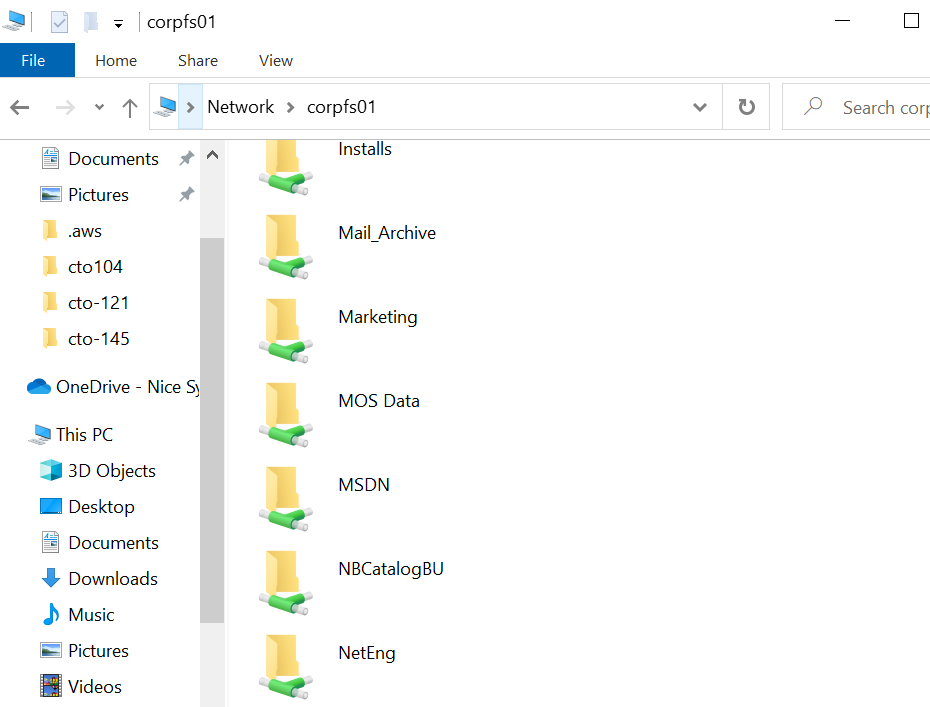


1. **Configure Windows firewall ports**

* Open the command prompt in administrator mode and run the following commands:
* netsh advfirewall firewall add rule name="Co-op Service" dir=in action=allow protocol=TCP localport=9303,8897
* netsh advfirewall firewall add rule name="File Server" dir=in action=allow protocol=TCP localport=9301,8899
* netsh advfirewall firewall add rule name="Media Server" dir=in action=allow protocol=TCP localport=9303,8915,9001,9300,5060,7930
* netsh advfirewall firewall add rule name="SQL Server (TCP)" dir=in action=allow protocol=TCP localport=1434,1433
* netsh advfirewall firewall add rule name="SQL Server (UDP)" dir=in action=allow protocol=UDP localport=1434
* netsh advfirewall firewall add rule name="VC" dir=in action=allow protocol=TCP localport=9300,9310,8898
* netsh advfirewall firewall add rule name="IIS Health Checks" dir=in action=allow protocol=TCP localport=80
* netsh advfirewall firewall add rule name="Co-op Service" dir=out action=allow protocol=TCP localport=9303,8897
* netsh advfirewall firewall add rule name="File Server" dir=out action=allow protocol=TCP localport=9301,8899
* netsh advfirewall firewall add rule name="Media Server" dir=out action=allow protocol=TCP localport=9303,8915,5060,7930
* netsh advfirewall firewall add rule name="SQL Server (TCP)" dir=out action=allow protocol=TCP localport=1434,1433
* netsh advfirewall firewall add rule name="SQL Server (UDP)" dir=out action=allow protocol=UDP localport=1434
* netsh advfirewall firewall add rule name="VC" dir=out action=allow protocol=TCP localport=9300,9310,8898
* netsh advfirewall firewall add rule name="IIS Health Checks" dir=out action=allow protocol=TCP localport=80

if the above commands do not work, please follow section 1.9.2 on [acd-personal-cluster-setup/1\_ConfiguringWindows.md at master · nice-cxone/acd-personal-cluster-setup (github.com)](https://github.com/nice-cxone/acd-personal-cluster-setup/blob/master/setupGuide/1_ConfiguringWindows.md) to manually configure the Windows firewall ports

**Installing and configuring SQL Server**



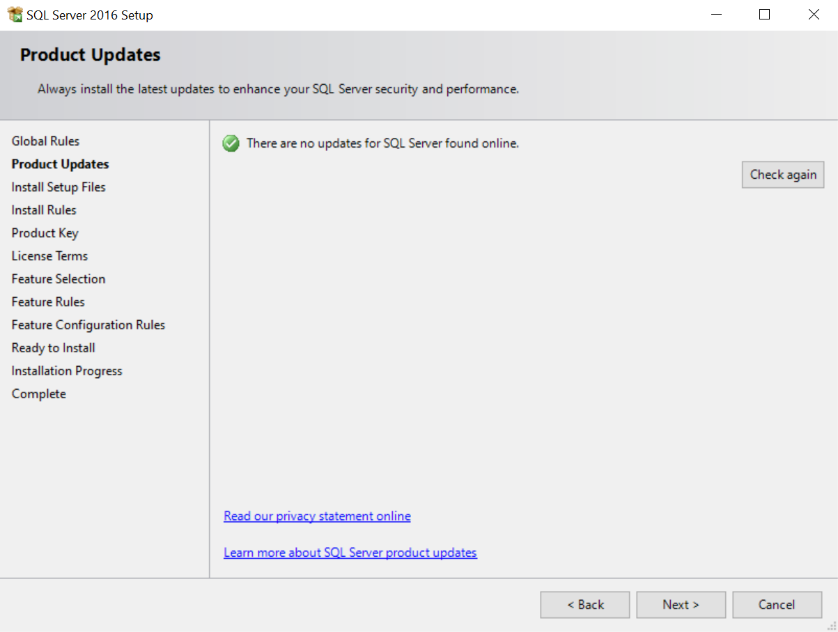
Assuming you have access to the corpfs1 folder, make Sure you enter [\\corpfs1](file:///\\corpfs1) in the address bar and not just \corpfs1(see the double slash)

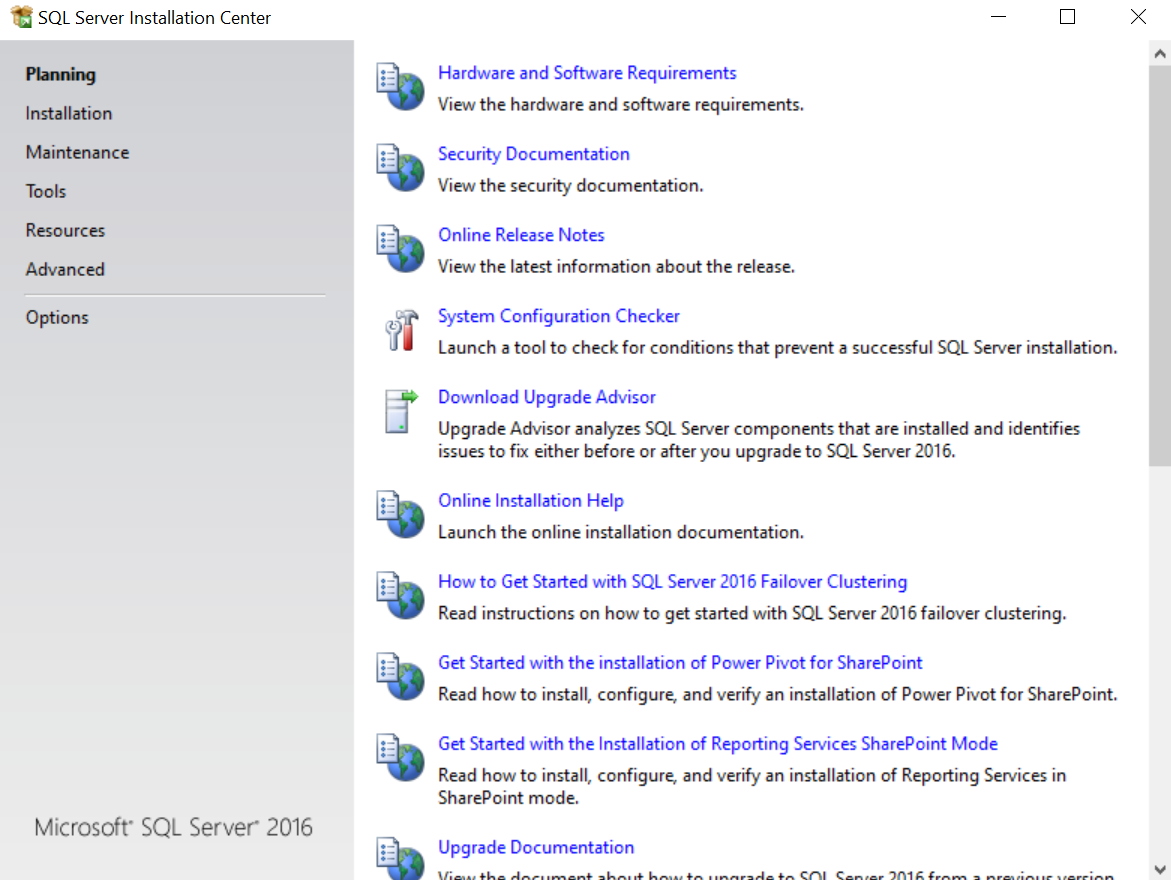
map this location to a local drive.

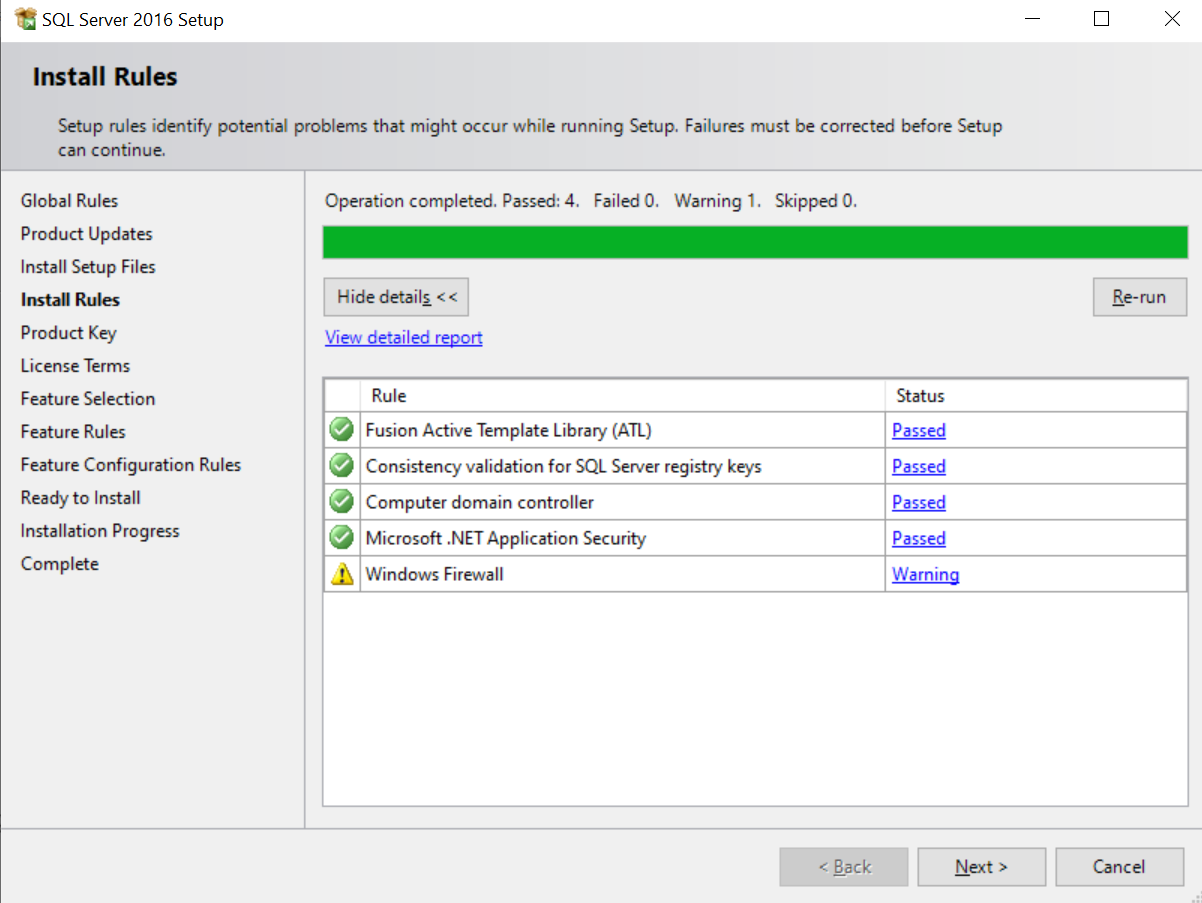
1. Open Windows file explorer
2. Go to \corpfs01
3. Right-click on the MSDN folder, and from the context menu select Map Network Drive.  A dialog will appear with a number of options.
   1. Select a drive letter (use Z:)
   2. Check the Reconnect at sign-in checkbox.
   3. Click the Finish button.

Download

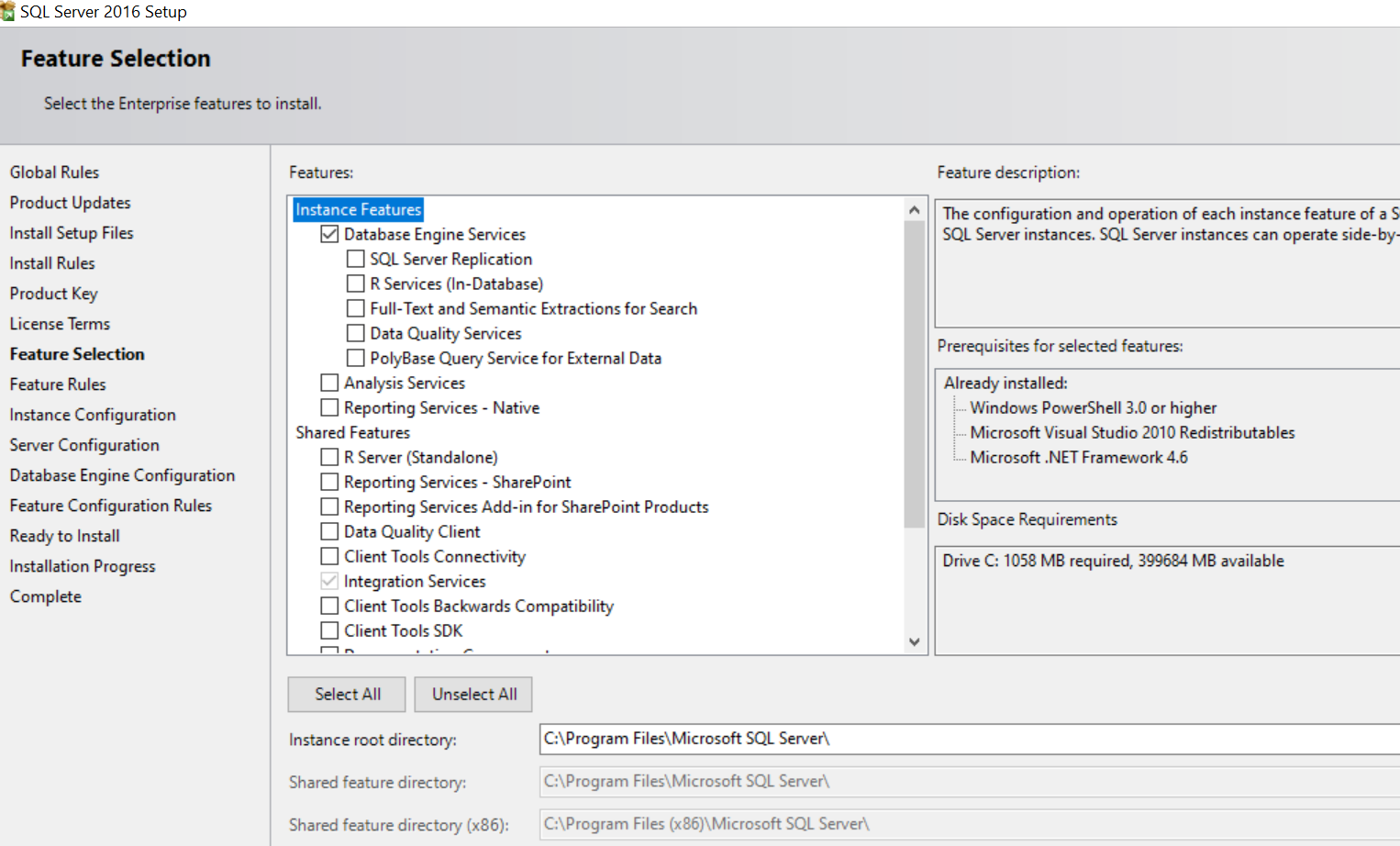
[\\corpfs01\MSDN\SQL Server\2016\Enterprise SP3\ enu\_sql\_server\_2016\_enterprise\_edition\_with\_service\_pack\_3\_\_x64\_dvd\_3db53b8](file:///\\corpfs01\MSDN\SQL%20Server\2016\Enterprise%20SP3\%20enu_sql_server_2016_enterprise_edition_with_service_pack_3__x64_dvd_3db53b8)b

Mount this image using the Virtual CloneDrive, access it, and run the setup as an admin. You should see the following window then. 

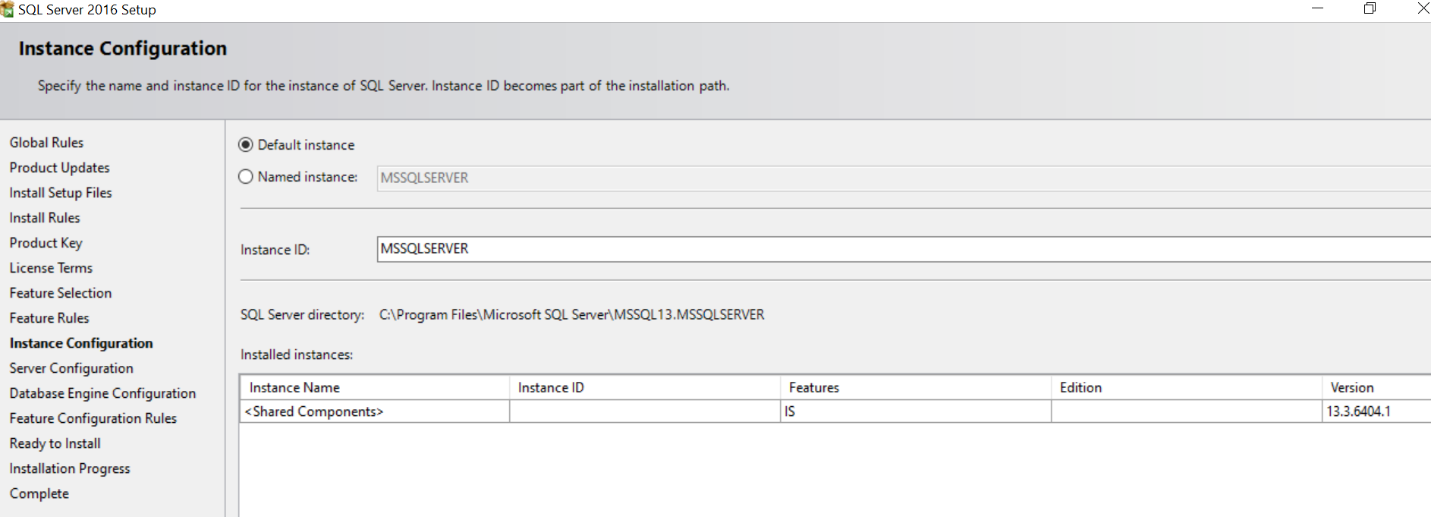
Click on installation in the left pane



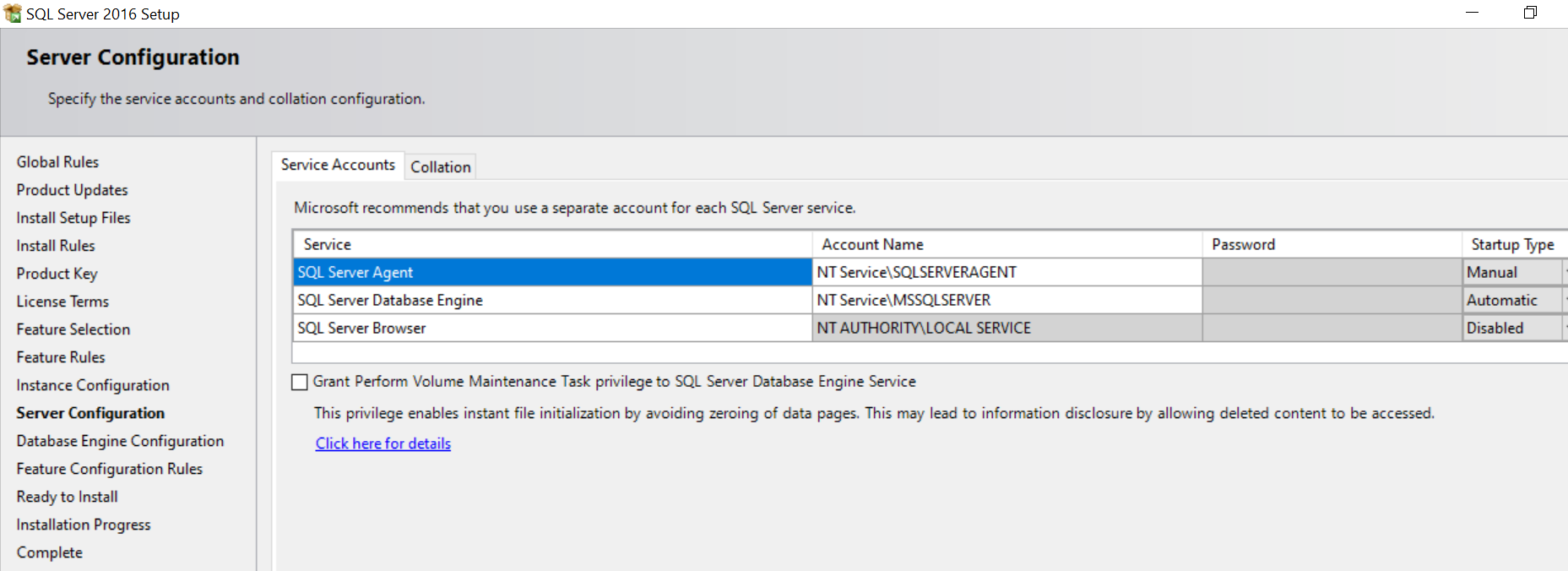
Click Next



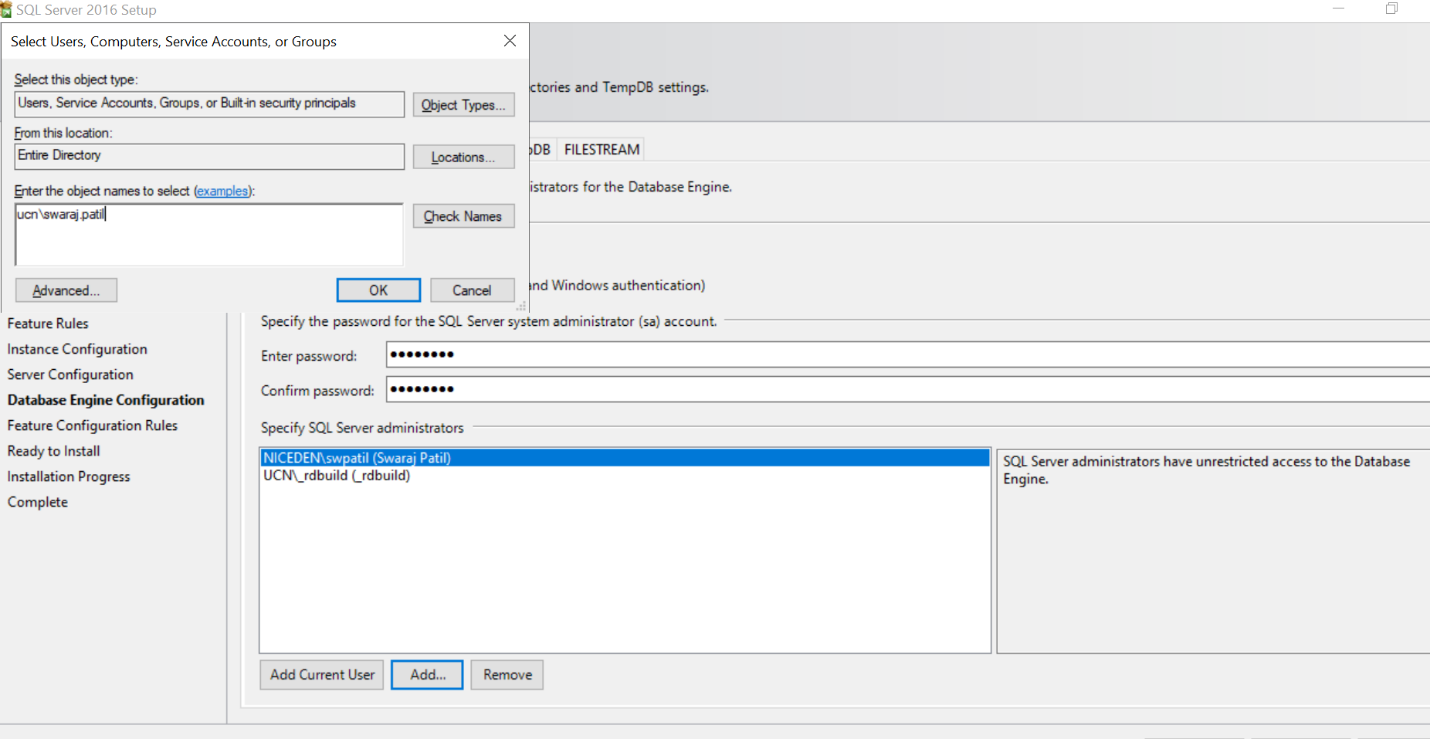
Select Database Engine Service, scroll down and also select Integration Service and continue



Select Default Instance and continue

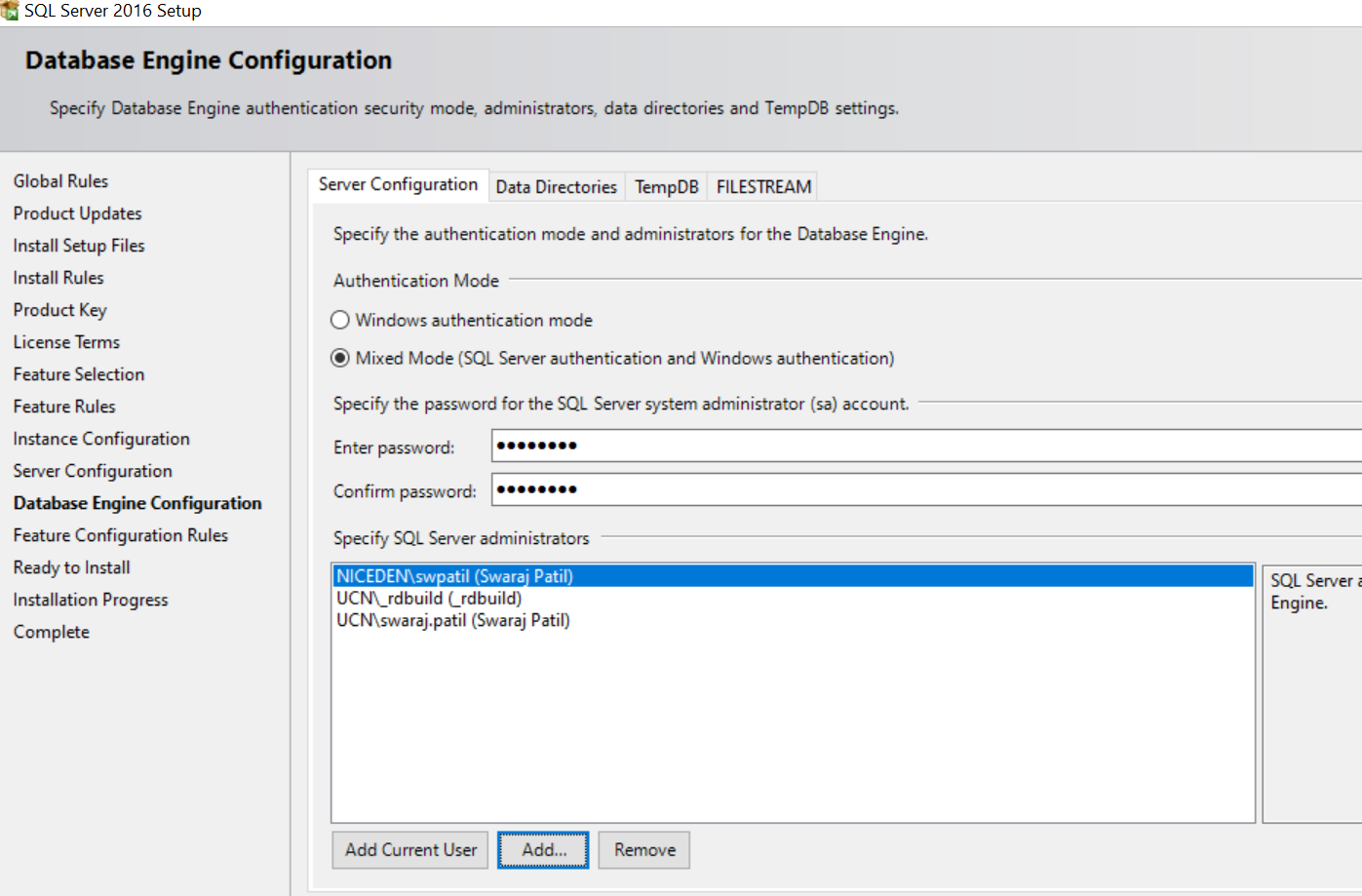


Keep everything default here and continue

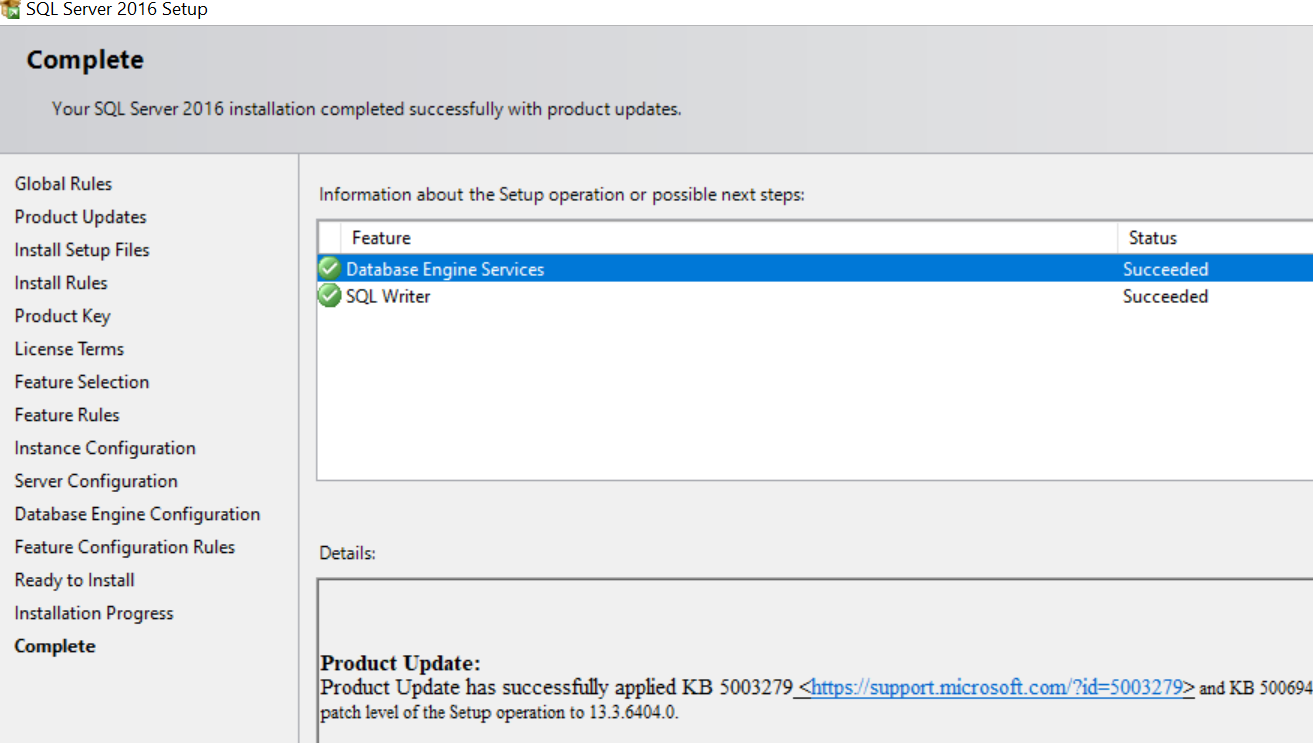


Click the “Add Current User” button

Also, click add and add the UCN accounts



Continue and finish the installation



## Install SQL Server Management studio 2014(SSMS) from:

[\\corpfs01\MSDN\SQL Server\2014 Management Studio\en\_sql\_server\_2014\_management\_studio\_x86\_exe\_3941421.exe](file:///\\corpfs01\MSDN\SQL%20Server\2014%20Management%20Studio\en_sql_server_2014_management_studio_x86_exe_3941421.exe)

If your download is taking too much time, you can download the file from MSDN.

This setup app is fairly automated. Just click through it and babysit it until it’s finished.

**Installing Visual Studios**

* Install Visual Studios 2019
* Follow the instructions on [acd-personal-cluster-setup/3\_InstallingVisualStudio.md at CTO-148 · nice-cxone/acd-personal-cluster-setup (github.com)](https://github.com/nice-cxone/acd-personal-cluster-setup/blob/CTO-148/setupGuide/3_InstallingVisualStudio.md) for doing the right settings on visual studios
* **Use the fully qualified name of the domain which is corptfsapp01.ucn.net – the github doc has it incorrectly mentioned as just corptfsapp01**

**Initializing the InContact Database**

Follow the instructions on this page: [acd-personal-cluster-setup/4\_InitializingTheInContactDatabases.md at master · nice-cxone/acd-personal-cluster-setup (github.com)](https://github.com/nice-cxone/acd-personal-cluster-setup/blob/master/setupGuide/4_InitializingTheInContactDatabases.md)

**Note**:

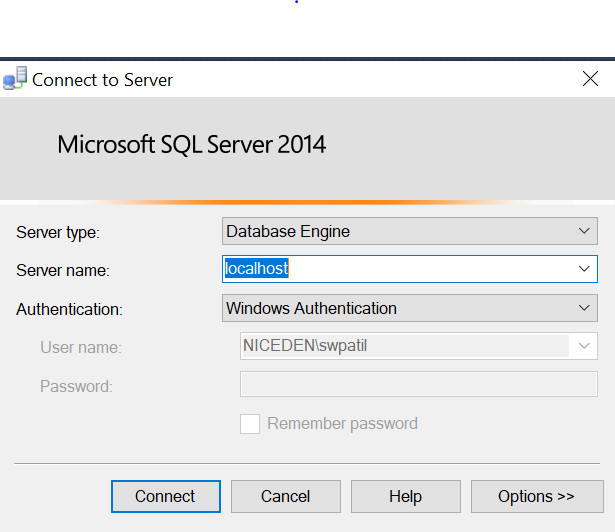
* Instructions for section 5.3 are for visual studios
* In section 5.4, use “localhost”(without the quotes) in the data source section. For the security tab, depending on where you are, you might need to add the NICEDEN\username account instead of the UCN\username account. Try with both in case you get a Authentication error.
* There are constant changes to the database schema (no surprises here). You will probably run into missing tables and entries while deploying your personal cluster. You can look in the troubleshooting section for an example script to update the tables but will probably need to talk to someone to add the latest schema changes to the script.

**5.1 Create the inContact databases**

Your cluster uses six databases:

* inData: contains all the tables used by our platform.
* inCode: contains all the stored procedures used to access those tables.
* FileServer: a database used for keeping track of files that customers store on our servers.
* MyGlobal: a database used to share certain information globally across all clusters in production.
* inDialerData: A special database to keep track of outbound dialer configuration.
* Cumulus: A database for media server.

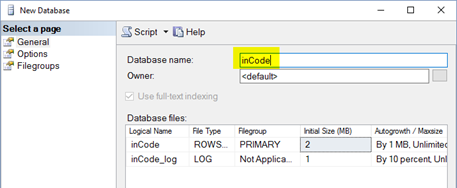
You will have to create each of these databases manually.  Start by opening SSMS (SQL Server Management Studio).

1. Connect to your computer  
   

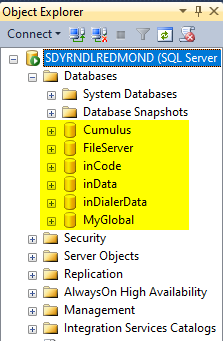
Put your Server name as localhost and use your UCN username

1. In **Object Explorer**, right click where it says **Databases**
2. From the context menu, select **New Database...**. A dialog will appear.
   1. Where it says **Database name**, type inData.
   2. Click **OK**. You shouldn’t have to modify any other settings on this page.
   3. Repeat the same steps to create the following databases: inCode, FileServer, MyGlobal, inDialerData, and Cumulus

Here’s a screenshot.



When you’re done you should have six new databases:



**5.2 Seeding static tables in the FileServer database**

After creating the databases, create a new query in the FileServer Database and run the following query:

Click to expand...

**5.3 Setting up the MyGlobal database**

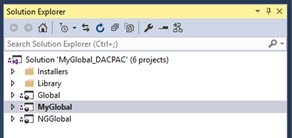
Usually we use **WebAdmins** to deploy a database, but MyGlobal only needs to be deployed *ONE TIME*, so this step is manual.

**Note**: If you haven’t downloaded the **MAIN** branch yet, you’ll want to do that now.

1. Checkout the latest changes from **MAIN**.
2. Navigate to the following directory: $/inContact/Source Code/MAIN/Database/MyGlobal\_DACPAC
3. Double-click on **MyGlobal\_DACPAC.sln** to open it.

Make sure **Solution Explorer** is open. If it is not, then you can open it from the main menu by selecting *View* > *Solution Explorer*.

There are three database projects in this solution (among others). They are **Global**, **MyGlobal**, and **NGGlobal**.

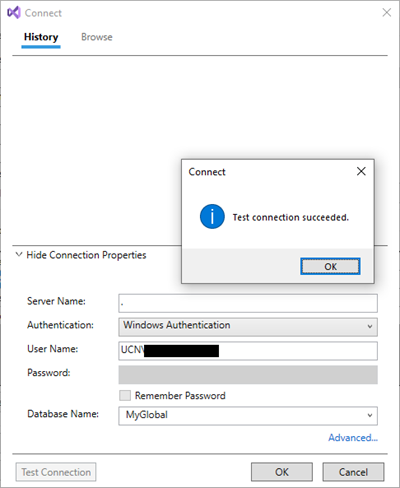


**Note**: You will only need to deploy **MyGlobal**. The other databases are only used in a fully-replicated environment, and are not applicable for a personal cluster.

1. In **Solution Explorer**, right-click on the **MyGlobal** project
2. From the pop-up menu select **Publish...**.

**Note:** A publish dialog will eventually appear (this may take a few seconds...).

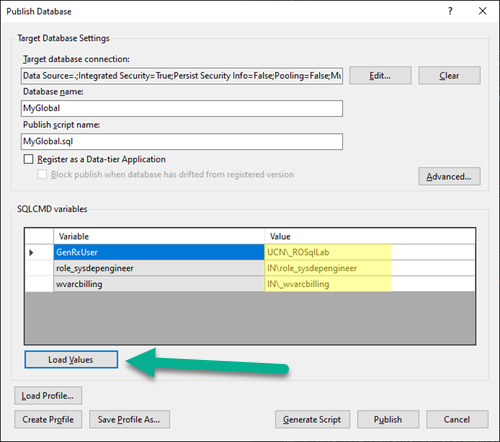
1. Where it says *Target database connection*, click the **Edit...** button. A connection dialog will appear.
2. Click on the expandable label that says **Show Connection Properties**. The connection properties will slide up and become visible.
3. Set *Server Name* to the [name of your computer] (or localhost or . or (local)).
4. Set *Authentication* to **Windows Authentication**.
5. Set *Database Name* to **MyGlobal**.
6. Click the **Test Connection** button!



1. Click **OK**.

Back in the publish dialog...

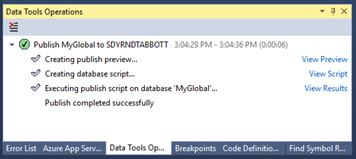
1. There is a section titled *SQLCMD variables*. Click the **Load Values** button.
   1. This will load default values.



1. Click the **Publish** button.

And away we go...

VS will show a window titled **Data Tools Operations**. It will crank away, and when you’re done it will show a message that says: **Publish completed successfully**.



Once completed, refresh the **Object Explorer** and expand the MyGlobal database. You should see that the database is populated with tables and stored procedures (under *Programmability/Stored Procedures*).

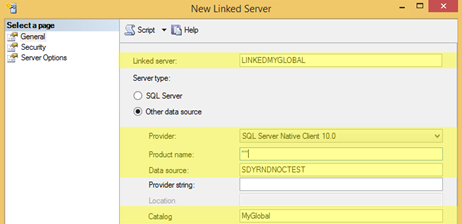
**5.4 Setting up LinkedMyGlobal**

Next we need to create a "*Linked Server*" back to your new MyGlobal database.

1. Go to the **Object Explorer** pane on the left and expand *Server Objects*.
2. Right-click on *Linked Servers* and select **New Linked Server**.
3. Follow the three screenshots below.

**General tab**

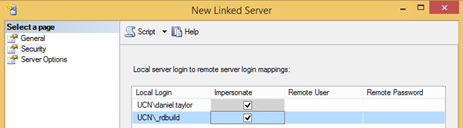
1. For *Linked server*, type LINKEDMYGLOBAL.
2. For *Data source*, enter [your machine name].
3. For *Catalog*, type MyGlobal.
4. For *Provider*, select SQL Server Native Client 11.0.
5. For *Product name*, type "".



**Security tab**

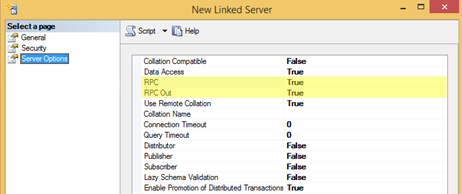
Add the following entries under **Local Login**:

* UCN\[your user name] (*Impersonate* = checked)
* UCN\\_rdbuild (*Impersonate* = checked)



**Server Options tab**

Ensure that both **RPC** and **RPC Out** are set to True



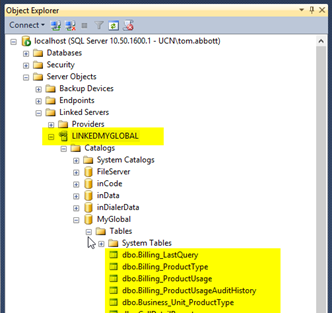
Press **OK** after configuring *General*, *Security* and *Server Options* tabs as show in the screenshots above.

**Validate it!**

When you’ve finished, go to **Object Explorer**, and expand down through the following:

*Server Objects* > *Linked Servers* > *LINKEDMYGLOBAL* > *Catalogs* > *MyGlobal* > *Tables*.

You should see a bunch of tables like the following...

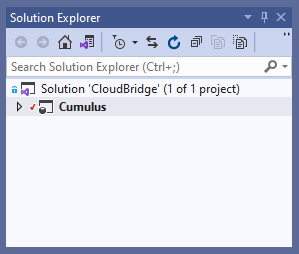


**5.5 Setting up Cumulus**

Back in **Visual Studio**, open the solution for **Cloudbridge**. You will find it at:

$/inContact/Source Code/MAIN/Database/CloudBridge

Double-click on the **Cloudbridge.sln** to open it. From the Solution Explorer you will see the Cumulus database project.



In the Solution Explorer, right-click on the **Cumulus** project, and from the pop-up menu select **Build**.

**Note**: *If you get an error "****This statement is not recognized in this context"****from API\_GetEvolveConfiguration.sql, you can get around this by removing the 5 lines of code at the top of the file starting with the IF block.*

Then click **Publish** button. Visual Studio will build the solution after which a publish dialog will appear.

1. Where it says **Target database connection**, click the **Edit...** button. A connection dialog will appear.
2. Click the expandable label that says **Show Connection Properties**.
   1. Set **Server Name** to [name of your computer] (or localhost or . or (local))
   2. Set **Authentication** to Windows Authentication.
   3. Set  **Database Name** to Cumulus.
   4. Click the **Test Connection** button, and make sure you get a success message.
   5. Click **OK**.
3. (Load values are not needed for the Cumulus database.)