

As default when GeoServer is installed it will provide a set number of Vector and Raster **Data Stores**, which allow you to connect to and publish data from certain sources, such as Shapefile, PostGIS and Image Files. The list of default Data Stores are as follows:

### New data source

Choose the type of data source you wish to configure

**Vector Data Sources**

-  Directory of spatial files (shapefiles) - Takes a directory of shapefiles and exposes it as a data store
-  PostGIS - PostGIS Database
-  PostGIS (JNDI) - PostGIS Database (JNDI)
-  Properties - Allows access to Java Property files containing Feature information
-  Shapefile - ESRI(tm) Shapefiles (\*.shp)
-  Web Feature Server (NG) - Provides access to the Features published a Web Feature Service, and the ability to perform transactions on the server (when supported / allowed).

**Raster Data Sources**

-  ArcGrid - Arc Grid Coverage Format
-  GeoTIFF - Tagged Image File Format with Geographic information
-  Gtopo30 - Gtopo30 Coverage Format
-  ImageMosaic - Image mosaicking plugin
-  WorldImage - A raster file accompanied by a spatial data file

**Other Data Sources**

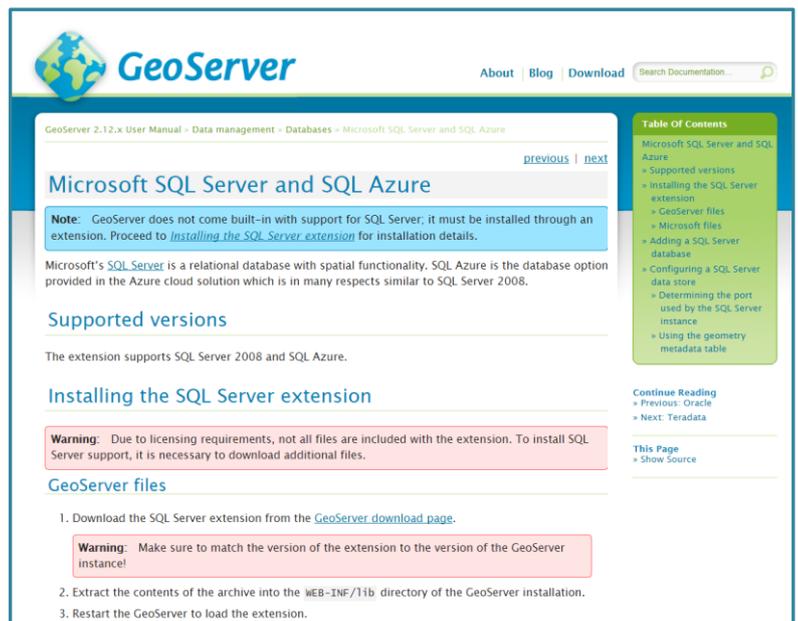
-  WMS - Cascades a remote Web Map Service

In this White Paper, we will explore how to successfully install the **SQL Server Extension** for GeoServer.

### Option A: Download the SQL Server JNDI Plugin:

Firstly, you will need to download the SQL Server Extension files for the specific instance of GeoServer. It is important to ensure that the extension version that you download matches the version number of your GeoServer – and in my case, this is **version 2.12**.

<http://docs.geoserver.org/stable/en/user/data/database/sqlserver.html>



The screenshot shows the GeoServer documentation page for Microsoft SQL Server and SQL Azure. The page includes a navigation menu, a search bar, and a table of contents. The main content area contains a note stating that GeoServer does not come built-in with support for SQL Server, and a warning about licensing requirements. The page also lists supported versions and provides instructions for installing the SQL Server extension.

**Microsoft SQL Server and SQL Azure**

**Note:** GeoServer does not come built-in with support for SQL Server, it must be installed through an extension. Proceed to [installing the SQL Server extension](#) for installation details.

Microsoft's **SQL Server** is a relational database with spatial functionality. SQL Azure is the database option provided in the Azure cloud solution which is in many respects similar to SQL Server 2008.

**Supported versions**

The extension supports SQL Server 2008 and SQL Azure.

**Installing the SQL Server extension**

**Warning:** Due to licensing requirements, not all files are included with the extension. To install SQL Server support, it is necessary to download additional files.

**GeoServer files**

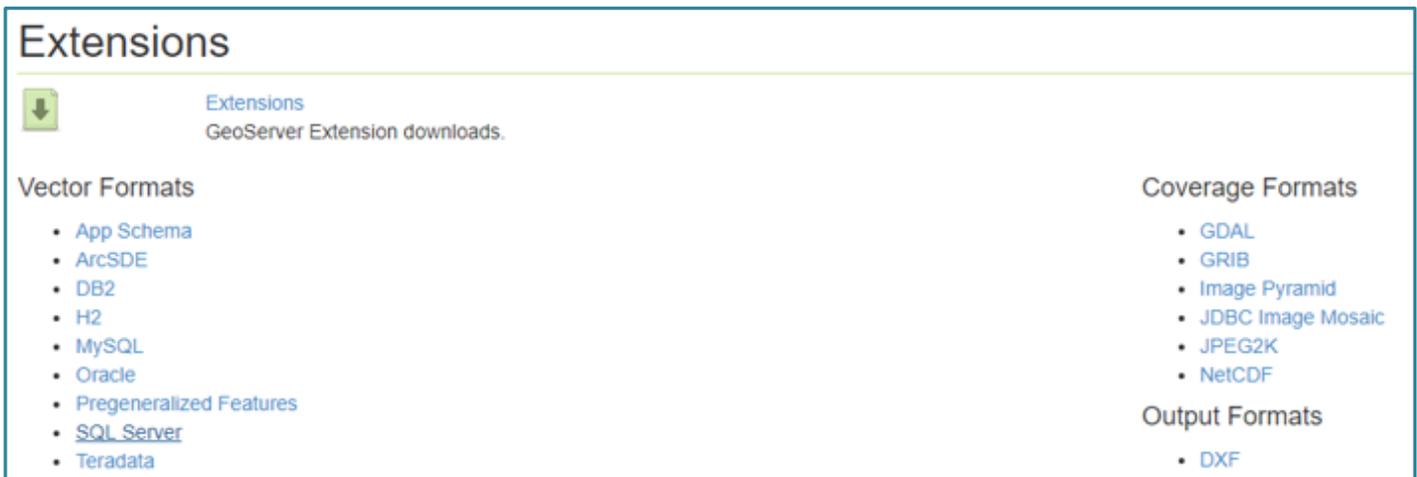
- Download the SQL Server extension from the [GeoServer download page](#).

**Warning:** Make sure to match the version of the extension to the version of the GeoServer instance!

- Extract the contents of the archive into the `WEB-INF/1ib` directory of the GeoServer installation.
- Restart the GeoServer to load the extension.

1 – Open the Download page - <http://geoserver.org/release/2.12.0/>

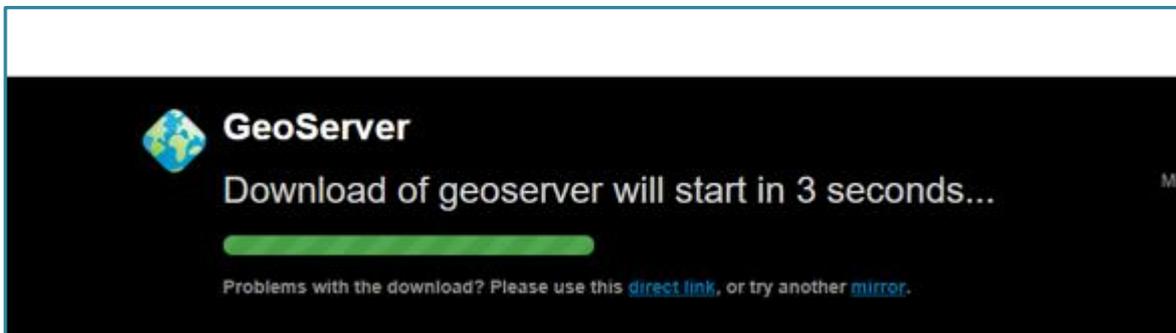
2 - And choose to download the **SQL Server Extension**:



The screenshot shows the 'Extensions' page on the GeoServer website. It features a green download icon and the text 'Extensions GeoServer Extension downloads.' Below this, there are three columns of links for different extension categories: Vector Formats, Coverage Formats, and Output Formats.

Vector Formats	Coverage Formats	Output Formats
<ul style="list-style-type: none"><li>App Schema</li><li>ArcSDE</li><li>DB2</li><li>H2</li><li>MySQL</li><li>Oracle</li><li>Pregeneralized Features</li><li><b>SQL Server</b></li><li>Teradata</li></ul>	<ul style="list-style-type: none"><li>GDAL</li><li>GRIB</li><li>Image Pyramid</li><li>JDBC Image Mosaic</li><li>JPEG2K</li><li>NetCDF</li></ul>	<ul style="list-style-type: none"><li>DXF</li></ul>

3 – A download page will appear which automatically starts the download of the extension files.



The screenshot shows a black download page for GeoServer. It features the GeoServer logo and the text 'Download of geoserver will start in 3 seconds...'. Below this is a green progress bar and a link to a direct download link.

4 – The following zip file will be downloaded to your machine.

Name	Date modified	Type	Size
 geoserver-2.12.0-sqlserver-plugin.zip	01/11/2017 14:35	WinRAR ZIP archive	41 KB

5 – Unzip the file to reveal the SQL Server .JAR file for GeoServer.

Name	Date modified	Type	Size
 geoserver-2.12.0-sqlserver-plugin.zip	01/11/2017 14:35	WinRAR ZIP archive	41 KB
 gt-jdbc-sqlserver-18.0.jar	12/10/2017 07:58	JAR File	45 KB

6 – Copy the Sql Server .JAR file to the following folder where your GeoServer instance is installed to:

**C:\Program Files (x86)\GeoServer 2.12\webapps\geoserver\WEB-INF\lib**

Local Disk (C:) > Program Files (x86) > GeoServer 2.12 > webapps > geoserver > WEB-INF > lib

Name	Date modified	Type	Size
 gt-jdbc-18.0.jar	09/10/2017 09:58	JAR File	223 KB
 gt-jdbc-postgis-18.0.jar	09/10/2017 09:58	JAR File	51 KB
 gt-jdbc-sqlserver-18.0.jar	12/10/2017 07:58	JAR File	45 KB
 gt-main-18.0.jar	09/10/2017 09:57	JAR File	1 739 KB

7 - If you now **Stop and Re-Start GeoServer**, the new SQL Server JNDI extensions have been installed and are available to use.

### New data source

Choose the type of data source you wish to configure

#### Vector Data Sources

-  Directory of spatial files (shapefiles) - Takes a directory of shapefiles and exposes it as a data store
-  GeoPackage - GeoPackage
-  Microsoft SQL Server (JNDI) - Microsoft SQL Server (JNDI)
-  Microsoft SQL Server (JTDS Driver) (JNDI) - Microsoft SQL Server (JTDS Driver) (JNDI)
-  PostGIS - PostGIS Database
-  PostGIS (JNDI) - PostGIS Database (JNDI)
-  Properties - Allows access to Java Property files containing Feature information
-  Shapefile - ESRI(tm) Shapefiles (\*.shp)
-  Web Feature Server (NG) - Provides access to the Features published a Web Feature Service, and the ability to perform transaction on the server (when supported / allowed).

#### Raster Data Sources

-  ArcGrid - ARC/INFO ASCII GRID Coverage Format
-  GeoPackage (mosaic) - GeoPackage mosaic plugin
-  GeoTIFF - Tagged Image File Format with Geographic information
-  Gtopo30 - Gtopo30 Coverage Format
-  ImageMosaic - Image mosaicking plugin
-  WorldImage - A raster file accompanied by a spatial data file

8 – Using the SQL Server JNDI extension it is possible to setup a connection to your SQL Database; however, the connection properties are stored within a Java Naming and Directory Interface (JNDI), which you may not have previously utilised. Notice below in the connection parameters the **JNDI Reference Name**:

### New Vector Data Source

Add a new vector data source

---

Microsoft SQL Server (JNDI)  
Microsoft SQL Server (JNDI)

---

**Basic Store Info**

Workspace \*

Data Source Name \*

Description

Enabled

---

**Connection Parameters**

dbtype \*

jndiReferenceName \*

schema

Note from Geoserver on JNDI: <http://docs.geoserver.org/latest/en/user/data/database/jndi.html>

Many data stores and connections in GeoServer have the option of utilizing Java Naming and Directory Interface on JNDI. JNDI allows for components in a Java system to look up other objects and data by a predefined name. A common use of JNDI is to store a JDBC data source globally in a container. This has a few benefits. First, it can lead to a much more efficient use of database resources. Database connections in Java are very resource-intensive objects, so usually they are pooled. If each component that requires a database connection is responsible for creating their own connection pool, resources will stack up fast. In addition, often those resources are under-utilized and a component may not size its connection pool accordingly. A more efficient method is to set up a global pool at the servlet container level, and have every component that requires a database connection use that.

### Option B - Download the SQL Server JDBC Drivers:

To install a more common SQL Server Data Source Connection, you can instead utilise the **JDBC drivers** for GeoServer to connect to SQL.

A **JDBC driver** is a software component enabling a Java application to interact with a database. **JDBC drivers** are analogous to **ODBC drivers**, **ADO.NET data providers**, and **OLE DB providers**. To connect with individual databases, **JDBC** (the Java Database Connectivity API) requires **drivers** for each database.

This time we will follow the steps as outlined within the Microsoft Files section –

<http://docs.geoserver.org/stable/en/user/data/database/sqlserver.html>

## Microsoft files

1. Navigate to the download page for [Microsoft JDBC Drivers for SQL Server](#).
2. Extract the contents of the archive.
3. Copy the file `sqljdbc4.jar` to the `WEB-INF/lib` directory of the GeoServer installation.
4. If GeoServer is installed on Windows, additionally copy `auth\x86\sqljdbc_auth.dll` and `xa\x86\sqljdbc_xa.dll` to `C:\Windows\System32`.

1 – Choose the link in step 1 to access the download of the required SQL Server JDBC drivers –

<https://www.microsoft.com/en-us/download/details.aspx?id=11774>

### Microsoft JDBC Driver 6.0 for SQL Server

Select Language:

English 

Download

Download the Microsoft JDBC Driver 6.0 for SQL Server, a Type 4 JDBC driver that provides database connectivity through the standard JDBC application program interfaces (APIs) available in Java Platform, Enterprise Editions.

 Details

 System Requirements

 Install Instructions

 Related Resources

2 – Choose **Download** and save the install files to your local machine.

Name	Date modified	Type	Size
license60.txt	01/11/2017 14:47	Text Document	9 KB
sqljdbc_6.0.8112.100_enu.exe	01/11/2017 14:47	Application	2,316 KB
sqljdbc_6.0.8112.100_enu.tar.gz	01/11/2017 14:47	WinRAR archive	2,144 KB

3 – Unzip the **sqljdbc\_6.0.8112.100\_enu.tar.gz** and the following folding and files are created.

Name	Date modified	Type	Size
auth	02/11/2017 10:05	File folder	
jre7	02/11/2017 10:05	File folder	
jre8	02/11/2017 10:05	File folder	
samples	02/11/2017 10:05	File folder	
xa	02/11/2017 10:05	File folder	
install.txt	17/01/2017 19:44	Text Document	2 KB
license.txt	17/01/2017 19:44	Text Document	9 KB
release.txt	17/01/2017 19:44	Text Document	9 KB

4 – From the **sqljdbc\_6.0\enu\jre8\** folder copy the file **sqljdbc4.jar** to the **WEB-INF\lib** directory of the GeoServer installation.

Name	Date modified	Type	Size
sqlite-jdbc-3.20.0.jar	23/09/2017 10:18	JAR File	6,482 KB
sqljdbc42.jar	17/01/2017 19:44	JAR File	871 KB

5 – If GeoServer is installed on Windows, additionally from the **auth\x86\** folder copy the **sqljdbc\_auth.dll** and from the **xa\x86\** folder copy the **sqljdbc\_xa.dll** to the following folder on the machine:  
**C:\Windows\System32.**

Name	Date modified	Type	Size
sqljdbc_auth.dll	17/01/2017 19:44	Application extens...	256 KB
sqljdbc_xa.dll	17/01/2017 19:44	Application extens...	166 KB

6 - If you now **Stop and Re-Start GeoServer**, the new SQL Server extension has been installed and is now available to use.

Vector Data Sources
Directory of spatial files (shapefiles) - Takes a directory of shapefiles and exposes it as a data store
GeoPackage - GeoPackage
Microsoft SQL Server - Microsoft SQL Server
Microsoft SQL Server (JNDI) - Microsoft SQL Server (JNDI)
Microsoft SQL Server (JTDS Driver) (JNDI) - Microsoft SQL Server (JTDS Driver) (JNDI)
PostGIS - PostGIS Database
PostGIS (JNDI) - PostGIS Database (JNDI)
Properties - Allows access to Java Property files containing Feature information
Shapefile - ESRI(tm) Shapefiles (*.shp)
Web Feature Server (NG) - Provides access to the Features published a Web Feature Service, and the ability to perform transac on the server (when supported / allowed).

7 – Using the SQL Server extension it is possible to setup a connection to your SQL Database; using the following parameters:

- **Host:** Name or IP Address of the Server that the SQL Database is on
- **Port:** The Port Number that your SQL Server DB is listening on – usually **1433**
- **Database:** Name of the Database within SQL that you wish to connect to – e.g. GeoStore
- **Schema:** Usually **dbo**
- **User and Password:** A User Name and password with access to login to the Database instance

Connection Parameters	
host *	<input type="text" value="laptop-ps-hp"/>
port	<input type="text" value="1433"/>
database	<input type="text" value="MTData"/>
schema	<input type="text" value="dbo"/>
user *	<input type="text" value="mapthat"/>
passwd	<input type="password" value="..."/>

## Next Steps:

To allow GeoServer to connect to your SQL Server Instance, you will need to follow several subsequent steps.

### **Allow SQL Server to accept TCP/IP connections:**

Having completed the correct SQL Server Connection parameters, when you choose to Save the Data Store the following error message may appear:

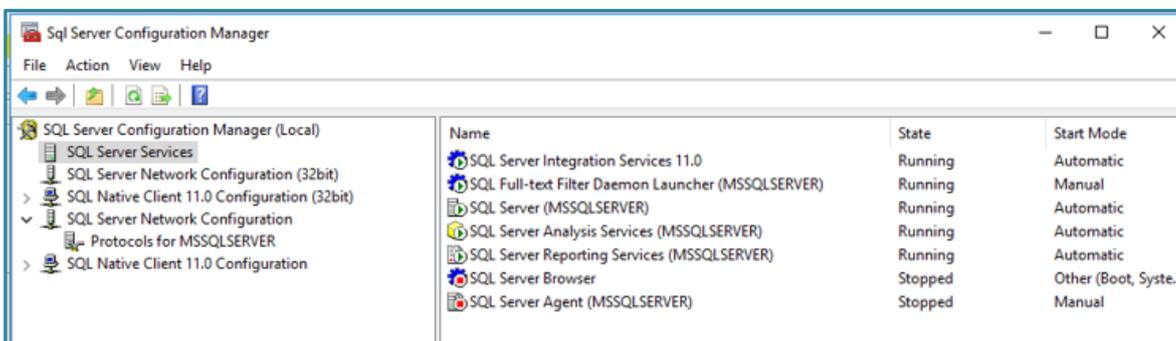
"Error creating data store, check the parameters. Error message: Unable to obtain connection: Cannot create PoolableConnectionFactory (The TCP/IP connection to the host SWD-4, port 1433 has failed. Error: connect timed out. Verify the connection properties, check that an instance of SQL Server is running on the host and accepting TCP/IP connections at the port, and that no firewall is blocking TCP connections to the port)"

This issue has a number of online fixes to enable the TCP/IP Network Protocol for SQL Server, including this post:

[https://technet.microsoft.com/en-us/library/hh231672\(v=sql.110\).aspx](https://technet.microsoft.com/en-us/library/hh231672(v=sql.110).aspx)

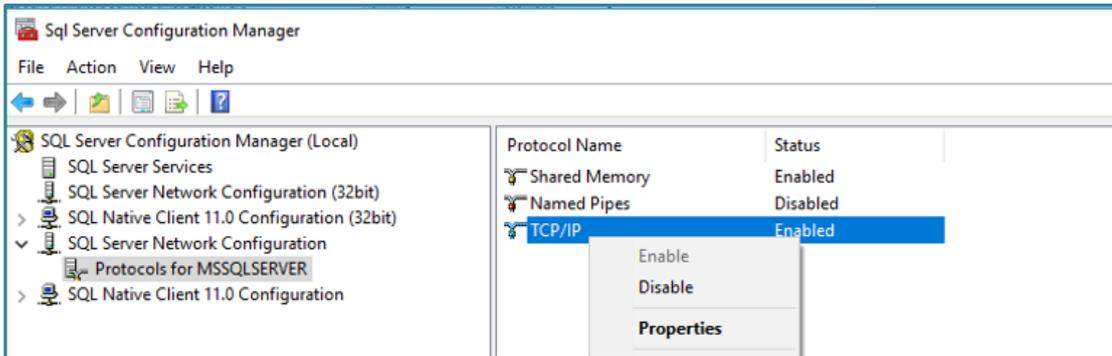
### **To enable the TCP/IP network protocol**

1. Start SQL Server Configuration Manager. Click **Start**, point to **All Programs**, and click **Microsoft SQL Server**. Click **Configuration Tools**, and then click **SQL Server Configuration Manager**.

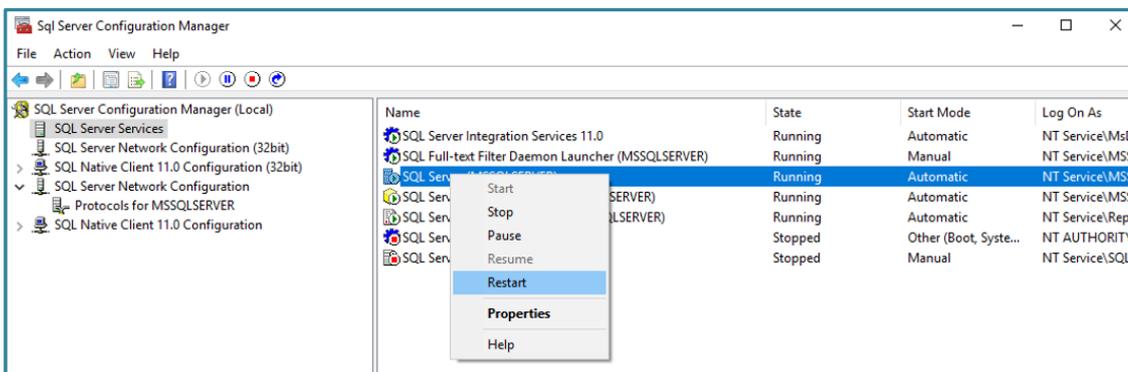


Note – To open SQL Server Config Manager on Windows 8 and up you need to search for **SQLServerManager11.msc**

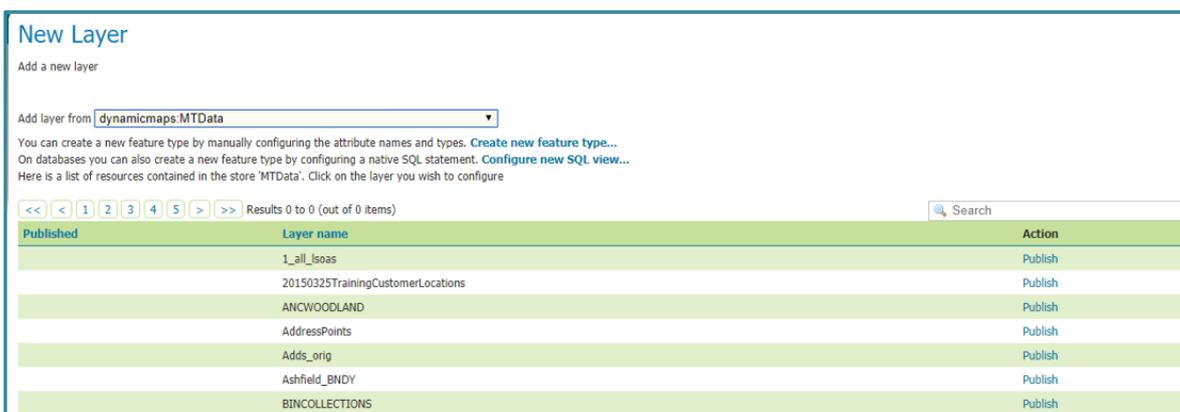
2. In SQL Server Configuration Manager, in the console pane, expand **SQL Server Network Configuration**.
3. In the console pane, click **Protocols for <instance name>**.
4. In the details pane, right-click **TCP/IP**, and then click **Enable**.



5. In the console pane, click **SQL Server Services**.
6. In the details pane, right-click **SQL Server (<instance name>)**, and then click **Restart**, to stop and restart the SQL Server service.



You should now be able to successfully Save the **New Data Store**:



And then create New WMS Layers from any of your SQL Spatial Tables:

