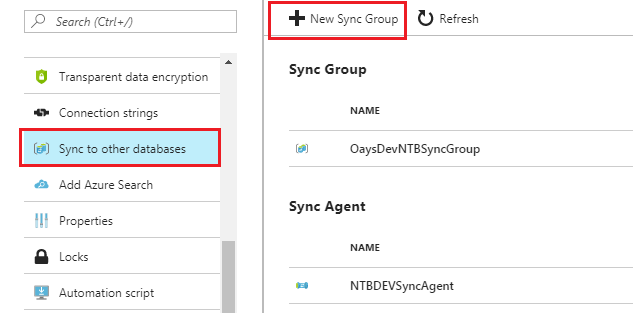
**Set up SQL Data Sync**

|  |
| --- |
| Data Sync is not appropriate for the following scenarios:+   * Disaster Recovery * Read Scale * ETL (OLTP to OLAP) * Migration from on-premises SQL Server to Azure SQL Database |
| When to Use Data Sync |
| Data Sync is useful in cases where data needs to be kept up to date across several Azure SQL Databases or SQL Server databases. Here are the main use cases for Data Sync:   * Hybrid Data Synchronization: With Data Sync, you can keep data synchronized between your on-premises databases and Azure SQL Databases to enable hybrid applications. This capability may appeal to customers who are considering moving to the cloud and would like to put some of their application in Azure. * Distributed Applications: In many cases, it's beneficial to separate different workloads across different databases. For example, if you have a large production database, but you also need to run a reporting or analytics workload on this data, it's helpful to have a second database for this additional workload. This approach minimizes the performance impact on your production workload. You can use Data Sync to keep these two databases synchronized. * Globally Distributed Applications: Many businesses span several regions and even several countries. To minimize network latency, it's best to have your data in a region close to you. With Data Sync, you can easily keep databases in regions around the world synchronized. |

## Step 1 - Create sync group

1. In your browser, navigate to the Azure portal.
2. In the portal, locate your SQL databases from your Dashboard or from the SQL Databases icon on the toolbar.
3. On the **SQL databases** page, select the existing SQL database that you want to use as the hub database for Data Sync. The SQL database page opens.
4. On the SQL database page for the selected database, select **Sync to other databases**. The Data Sync page opens. (see following figure)



### Create a new Sync Group

1. On the Data Sync page, select **New Sync Group**. The **New sync group** page opens with Step 1, **Create sync group**, highlighted. The **Create Data Sync Group** page also opens.
2. On the **Create Data Sync Group** page, do the following things:
3. In the **Sync Group Name** field, enter a name for the new sync group.
4. In the **Sync Metadata Database** section, choose whether to create a new database (recommended) or to use an existing database.

Note: \* (You better to create new Meta database. You can't change the Sync Metadata Database or its name without dropping it.)

If you chose **New database**, select **Create new database.** The **SQL Database**page opens. On the **SQL Database** page, name and configure the new database. Then select **OK**.

If you chose **Use existing database**, select the database from the list.

1. In the **Automatic Sync** section, first select **On** or **Off**.

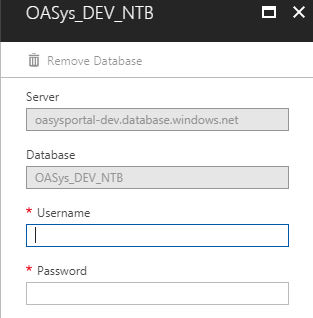
If you chose **On**, in the **Sync Frequency** section, enter a number and select Seconds, Minutes, Hours, or Days.

1. In the **Conflict Resolution** section, select "Hub wins" or "Member wins."
2. Select **OK** and wait for the new sync group to be created and deployed.

## Step 2 - Add sync members

After the new sync group is created and deployed, Step 2, **Add sync members**, is highlighted in the **New sync group** page.+

In the **Hub Database** section, enter the existing credentials for the SQL Database server on which the hub database is located. Don't enter new credentials in this section.

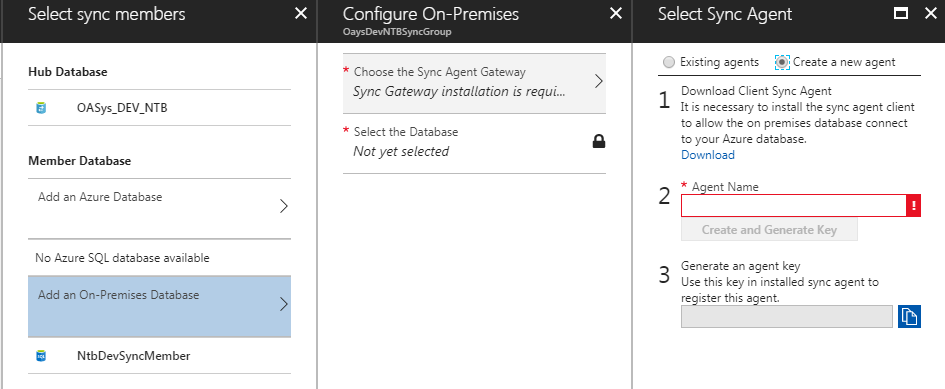


### Add an on-premises SQL Server database

In the **Member Database** section, optionally add an on-premises SQL Server to the sync group by selecting **Add an On-Premises Database**. The **Configure On-Premises** page opens.

On the **Configure On-Premises** page, do the following things:

1. Select **Choose the Sync Agent Gateway**. The **Select Sync Agent** page opens. (as following figure shows)



On the **Choose the Sync Agent Gateway** page, choose whether to use an existing agent or create a new agent.

If you chose **Existing agents**, select the existing agent from the list.

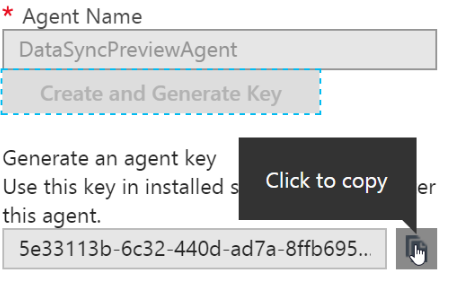
If you chose **Create a new agent**, do the following things:

1. Download the client sync agent software from the link provided and install it on the computer where the SQL Server is located.

Important

You have to open outbound TCP port 1433 in the firewall to let the client agent communicate with the server.

1. Enter a name for the agent.
2. Select **Create and Generate Key**.
3. Copy the agent key to the clipboard.



e. Select **OK** to close the **Select Sync Agent** page.

**Download and Install client sync agent on local machine**

|  |
| --- |
| Prerequisites   * SQL Server 2012 CLR Types - x86 version for US-EN (1033) locale   + Download the [SQLSysClrTypes.msi](http://go.microsoft.com/fwlink/?LinkID=239643&clcid=0x409) for SQL Server 2012 CLR Types. * SQL Server 2012 Shared Management Objects - x86 version for US-EN (1033) locale   + Download  the [SharedManagementObjects.msi](http://go.microsoft.com/fwlink/?LinkID=239658&clcid=0x409) for SQL Server Shared Management Objects. * .Net 4.5 Framework   + Download for .Net 4.5 Framework can be found [here](https://www.microsoft.com/en-us/download/details.aspx?id=30653). |

Download page: <https://www.microsoft.com/en-us/download/details.aspx?id=27693>

Please read system requirements and download and install other necessary .msi files on your local machine.

Then you can install Microsoft SQL Data Sync Agent.

After installing open “Microsoft SQL Data Sync Agent 2.0” and give your local pc credential to start service.

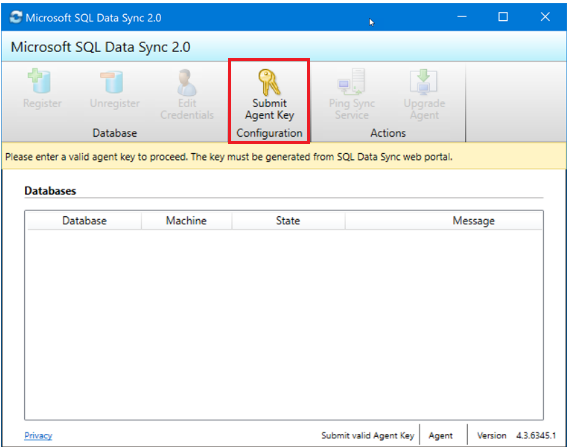
# Note : If you got error while connecting, Please refer “*SQL Azure Data Sync - Troubleshooting Guide”*

# [*https://social.technet.microsoft.com/wiki/contents/articles/2655.sql-azure-data-sync-troubleshooting-guide.aspx*](https://social.technet.microsoft.com/wiki/contents/articles/2655.sql-azure-data-sync-troubleshooting-guide.aspx)

Most common permission error:

# 

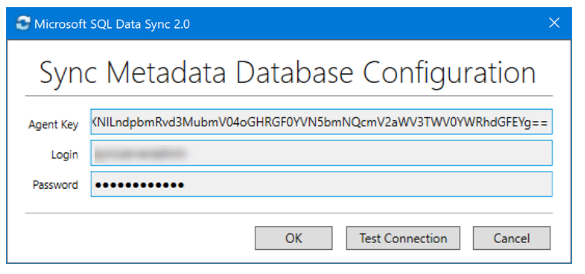
After successfully login you can see something like this.



f. On the SQL Server computer, locate and run the Client Sync Agent app. (as above figure)

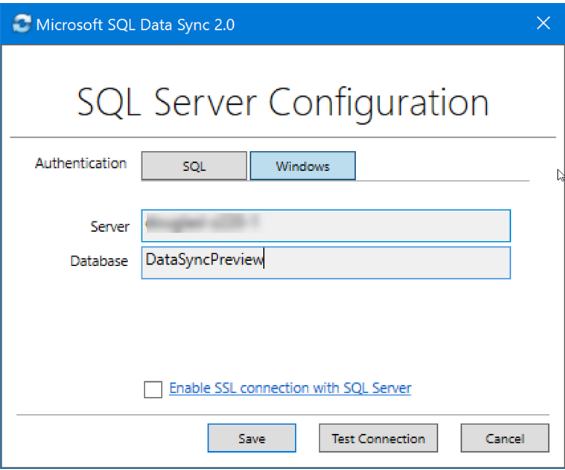
g. In the sync agent app, select **Submit Agent Key**. The **Sync Metadata Database Configuration** dialog box opens.

h. In the **Sync Metadata Database Configuration** dialog box, paste in the agent key copied from the Azure portal. Also provide the existing credentials for the Azure SQL Database server on which the metadata database is located. (If you created a new metadata database, this database is on the same server as the hub database.) Select **OK** and wait for the configuration to finish.

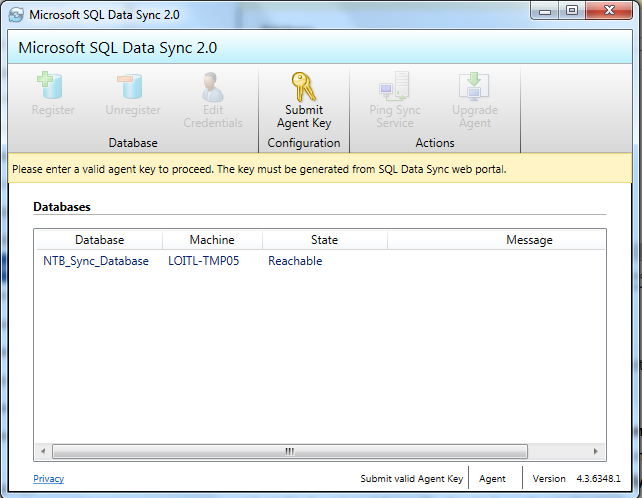


**Note:** If you get a firewall error at this point, you have to create a firewall rule on Azure to allow incoming traffic from the SQL Server computer. You can create the rule manually in the portal.

I. In the Client Sync Agent app, click **Register** to register a SQL Server database with the agent. The **SQL Server Configuration** dialog box opens.



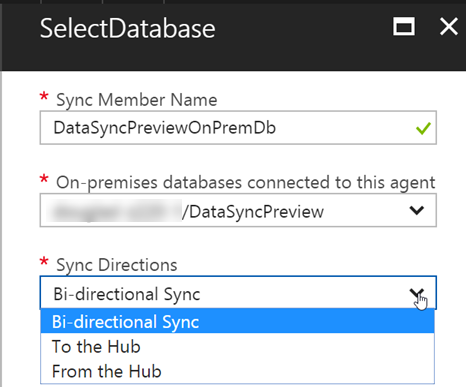
j. In the **SQL Server Configuration** dialog box, choose whether to connect by using SQL Server authentication or Windows authentication. If you chose SQL Server authentication, enter the existing credentials. Provide the SQL Server name and the name of the database that you want to sync. Select **Test connection** to test your settings. Then select **Save**. The registered database appears in the list.



k. You can now close the Client Sync Agent app.

l. In the portal, on the **Configure On-Premises** page, select **Select the Database.**The **Select Database** page opens.

m. On the **Select Database** page, in the **Sync Member Name** field, provide a name for the new sync member. This name is distinct from the name of the database itself. Select the database from the list. In the **Sync Directions** field, select Bi-directional Sync, To the Hub, or From the Hub.

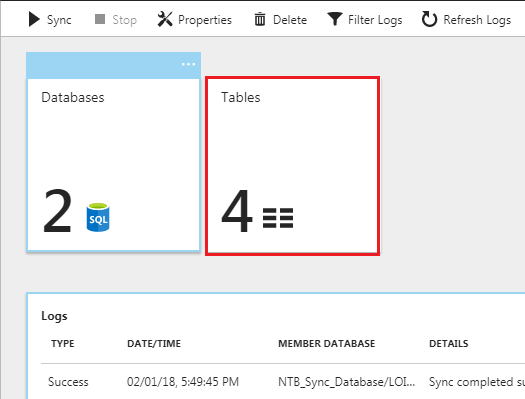


n. Select **OK** to close the **Select Database** page. Then select **OK** to close the **Configure On-Premises** page and wait for the new sync member to be created and deployed. Finally, click **OK** to close the **Select sync members** page.

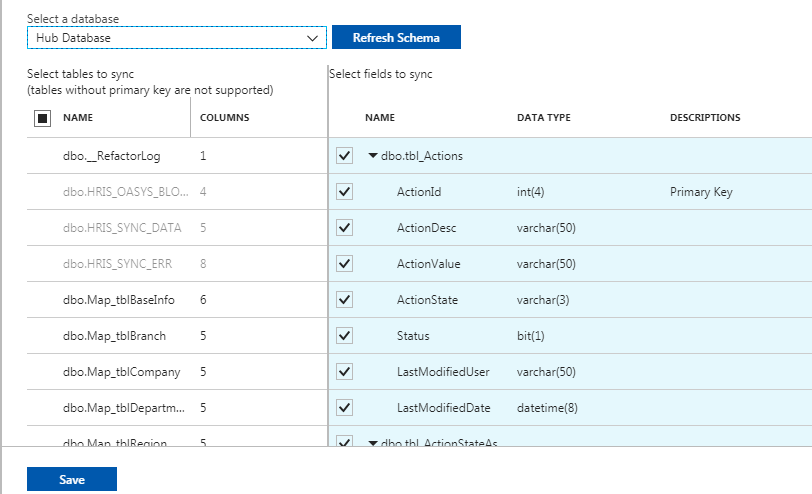
**Note :** To connect to SQL Data Sync and the local agent, add your user name (local machine sql server login) to the role DataSync\_Executor. Data Sync creates this role on the SQL Server instance (you can see “Meta database” server roles).

## Step 3 - Configure sync group

After the new sync group members are created and deployed, Step 3, **Configure sync group**, is highlighted in the **New sync group** page.



1. On the **Tables** page, select a database from the list of sync group members, and then select **Refresh schema**.
2. From the list of available tables, select the tables that you want to sync.
3. By default, all columns in the table are selected. If you don't want to sync all the columns, disable the checkbox for the columns that you don't want to sync. Be sure to leave the primary key column selected.



1. Finally, select **Save**.

## Requirements and limitations

### General requirements

* Each table must have a primary key. Don't change the value of the primary key in any row. If you have to change a primary key value, delete the row and recreate it with the new primary key value.
* Snapshot isolation must be enabled. For more info, see [Snapshot Isolation in SQL Server](https://docs.microsoft.com/dotnet/framework/data/adonet/sql/snapshot-isolation-in-sql-server). (<https://docs.microsoft.com/en-us/dotnet/framework/data/adonet/sql/snapshot-isolation-in-sql-server>)

To Enable Snapshot Isolation

|  |
| --- |
| ALTER DATABASE <databaseName>  SET ALLOW\_SNAPSHOT\_ISOLATION ON |

### General limitations

* A table cannot have an identity column that is not the primary key.
* The names of objects (databases, tables, and columns) cannot contain the printable characters period (.), left square bracket ([), or right square bracket (]).
* Azure Active Directory authentication is not supported.

#### Unsupported data types

* FileStream
* SQL/CLR UDT
* XMLSchemaCollection (XML supported)
* Cursor, Timestamp, Hierarchyid

Limitations on service and database dimensions

|  |  |
| --- | --- |
| ****Dimensions**** | ****Limit**** |
| Maximum number of sync groups any database can belong to. | 5 |
| Maximum number of endpoints in a single sync group | 30 |
| Maximum number of on-premises endpoints in a single sync group. | 5 |
| Database, table, schema, and column names | 50 characters per name |
| Tables in a sync group | 500 |
| Columns in a table in a sync group | 1000 |
| Data row size on a table | 24mb |
| Minimum sync interval | 5min |
| On-premise Database of SQL Server 2005 SP2 or higher  Note: SQL Azure Data Sync works best with SQL Server 2008 R2 and later including SQL Server 2012 as support for SQL Azure was added to the SQL Server Management Studio since SQL Server 2008 R2.  The local agent can be installed on any machine that meets the hardware and OS requirements whether or not it has an instance of SQL Server installed. **SQL AZURE DATA SYNC SERVICE, SQL EXPRESS AND SQL EXPRESS 2012 LOCALDB** https://jtabadero.wordpress.com/2012/02/17/sql-azure-data-sync-service-sql-express-and-sql-express-2012-localdb/ |  |