



ICONICS – Using the Azure Cloud Connector



APPLICATION NOTE

June 2017

Description: Guide to using the Azure Cloud Connector.

General Requirement: Valid account for Azure, including Cloud Service, SQL Azure and Azure Storage.

Introduction

Cloud Connector is a FrameWorX Server version for Windows Azure, Microsoft's cloud computing platform and infrastructure (created by Microsoft for building, deploying and managing applications and services through a global network of Microsoft-managed datacenters).

The functionality of Cloud Connector is very similar to FrameWorX Server running in DMZ in forwarding mode.

There are two major differences between Cloud Connector and the standard FrameWorX Server:

- Instead of connecting to other FrameWorX Servers, it allows other FrameWorX Servers to connect to itself. This reverses the direction in which the connection is created and solves the "inbound port 80" issue. The on premises firewall does not have to be open for inbound communication because the connection is initiated by the computers behind the firewall (thus, it goes outbound).
- The functionality of Cloud Connector is limited, partly because the features have not been fully implemented and partly because the features are not yet available on Azure (i.e., there are no OPC servers on the cloud. They are always on premises.). The requests that Cloud Connector cannot handle may be forwarded to a selected FrameWorX Server on premises.

Cloud Connector provides a connection between client computers and FrameWorX Servers on premises. These FrameWorX Servers can then expose any data available in GENESIS64.

Both clients and FrameWorX Servers can run behind firewalls (i.e. no firewall setup is needed).

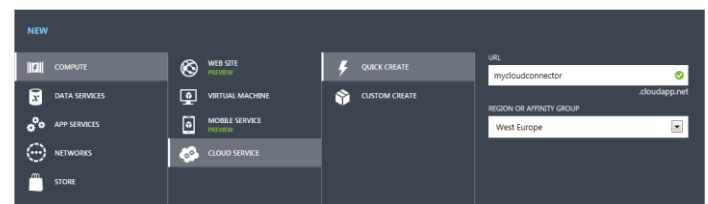
Azure Setup

The following set up assumes the user has an account on Azure. It describes the steps to install Cloud Connector on the Azure account.

NOTE: ICONICS is not responsible for Azure accounts. You need to connect Microsoft for detailed information and pricing.

Create a Cloud Service

1. In Windows Azure Management Portal, click NEW (in the bottom left corner), then select COMPUTE → CLOUD SERVICE → QUICK CREATE. Specify a URL for the service, select a region and hit OK.



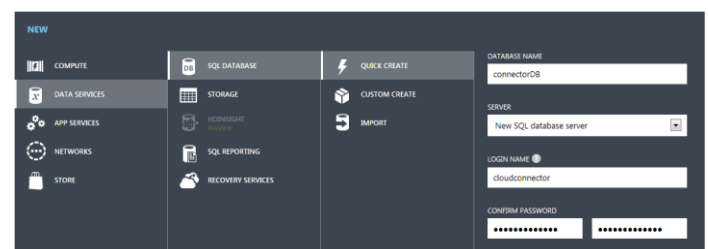
2. The new cloud service will appear in the list of services:

cloud services

NAME	SERVICE STATUS	PRODUCTION	STAGING	SUBSCRIPTION	LOCATION	URL	
anyglass	✓ Created	-	-	Subscription-1	North Europe	http://anyglass.cloudapp.net	
ico	✓ Created	-	-	Subscription-1	North Centr...	http://ico.cloudapp.net	
icoEurope	✓ Created	✓ Running	-	Subscription-1	North Europe	http://icoEurope.cloudapp.net	
mycloudconnector	✓ Created	-	-	Subscription-1	West Europe	http://mycloudconnector.cloudapp.net	

Create a SQL Azure Database

1. You have to create a new SQL Azure database. It will store the configuration for the Cloud Connector. Click NEW, then select DATA SERVICES → SQL DATABASE → QUICK CREATE. Specify a database name, a preferred server, then the login name and password.



2. The QUICK CREATE option creates the SQL server in North Central US region. To change the region, use the CUSTOM CREATE option and specify the region.



ICONICS – Using the Azure Cloud Connector



APPLICATION NOTE

June 2017

NEW SQL DATABASE - CUSTOM CREATE

SQL database server settings

LOGIN NAME
cloudconnector

LOGIN PASSWORD

CONFIRM PASSWORD

REGION
West Europe

☒ ALLOW WINDOWS AZURE SERVICES TO ACCESS THE SERVER.

- The new SQL database will appear in the list of databases.

sql databases

NAME	STATUS	LOCATION	SUBSCRIPTION	SERVER	EDITION	MAX SIZE	
Genesis	Online	West Europe	Subscription-1	ontrcs30mb	Web	1 GB	
connectorDB	Online	West Europe	Subscription-1	px5bmhb4ql	Web	1 GB	

Create an Azure Storage

- The Azure storage will be used for Cloud Connector to exchange data with the FrameWorX Servers on premises. Create the storage with NEW → DATA SERVICES → STORAGE → QUICK CREATE. Select a name for the storage and its location.

NEW

COMPUTE

DATA SERVICES

APP SERVICES

NETWORKS

STORE

SQL DATABASE

STORAGE

SQL REPORTING

RECOVERY SERVICES

QUICK CREATE

URL
cloudconnectorstorage

LOCATION/AFFINITY GROUP
West Europe

☒ Enable Geo-Replication

- The new storage will appear in the list of storages.

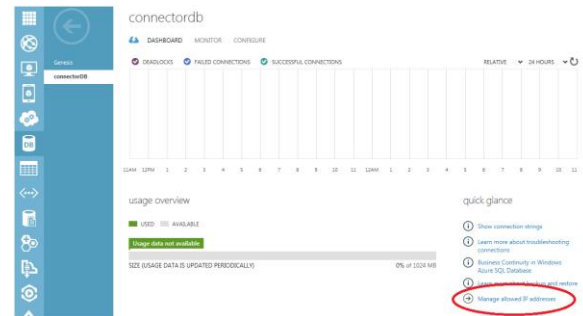
storage

NAME	STATUS	LOCATION	SUBSCRIPTION	
icloudstorage	Online	West Europe	Subscription-1	
cloudconnectorstorage	Online	West Europe	Subscription-1	

Create Tables in the Configuration Database

- Your newly created SQL Azure database is a SQL Server with limited functionality. You may connect to it from your computer with SQL Server Management Studio. However, the connection from the outside of Azure is disabled by default. To allow connection from your computer, select SQL DATABASES in Azure Portal, then select your database

(connectorDB). Click on DASHBOARD and Manage allowed IP addresses.



- Add your client IP address to the list of allowed IP addresses. You may want to add more IP addresses if you plan to configure the database from other computers.

allowed ip addresses

CURRENT CLIENT IP ADDRESS

ADD TO THE ALLOWED IP ADDRESSES

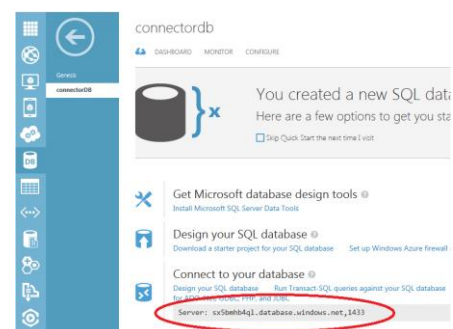
ClientIPAddress_2013-04-17_111745

RULE NAME

START IP ADDRESS

END IP ADDRESS

- Click SAVE to store the configuration.
- Now you are allowed to connect to your SQL Azure from your computer. Find the name of the SQL server (select SQL DATABASES → then select your database (connector DB)).

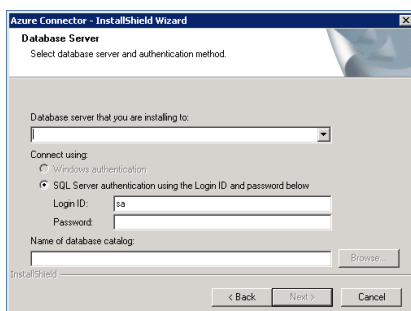


- Now that you have set the server name, you should be able to connect to the SQL Azure server with the Cloud Connector installation to create the tables needed for the Cloud Connector.



Running the Azure Connector Setup

1. After creating your database in Azure, the next step would be to run the Azure Connector Installation, which will create the databases in Azure.
2. Start the installation and go to the screen that asks what database server you are connecting to.



3. Make sure that the database server is the same as the in the previous section, and enter in the database name and the SQL username and password. Then choose your database catalog, by clicking on the browse button. Then click next to continue the installation.
4. Once this is finished installing, your tables should be setup in your Azure cloud instance. A way to check this would be to connect to the Azure SQL server with SQL Server Management studio. You can connect to this by entering in your server name in the SQL Management Studio Window and confirming that ICONICS related tables were created in your database.

Creating a Certificate for the Cloud Connector Service

Before uploading your package to Azure, you would need to edit the .CSCFG file with an appropriate thumbprint. For this application note, you may use a self-signed certificate, but when using this with a real application you should get a certificate from a certification authority.

NOTE: According to Microsoft, the certification authorities do not issue certificates for domains (*.cloudapp.net), so you would have to use a custom domain to redirect to the URL of your Azure Service. To find out more information go to the following link:
<https://azure.microsoft.com/en-us/documentation/articles/cloud-services-configure-ssl-certificate/>

1. Open an instance of an elevated Command Prompt by right-clicking on Command Prompt and selecting, "Run as Administrator." Run the following command to create a self-

signed certificate (and changing the URL of your Azure Service accordingly):

```
C:\Program Files\ICONICS\GENESIS64\InstCert makecert
-n "CN=icocloudconnector.cloudapp.net" -sr
LocalMachine -ss My -r -sky exchange -pe -sk
MarkContainerName
```

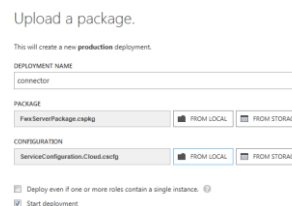
2. This command will generate a certificate in the Local Machine\Personal Store. To export this, use the Microsoft Management Console (also called MMC) to access the new certificate. (In MMC, select Add Snap-In from File menu; choose Certificates and hit OK; then, under Local Computer → Personal, select Certificates and locate your newly generated certificate to export.)
3. Export the certificate with a private key to a .PFX file by right clicking on the certificate and selecting All Tasks → Export.
4. In the wizard, select "Yes, export the private key".
5. Use the CERTIFICATES menu in Azure to upload the certificate that has been exported into a .PFX file.

Upload the Cloud Connector Service

1. Before uploading your package, go to C:\Program Files\ICONICS\AzureConnector\FwxServer or where you installed the Azure Cloud Connector setup, and edit the .CSCFG file. The section we are changing is the <Certificates> section. Look for the section as shown in the image below. Update the thumbprint to be the same as the one uploaded for your cloud connector to Azure.

```
<Setting name="TracerXCategoryName" value="CloudConnector" />
<Setting name="TracerXLoggingLevel" value="1000" />
<Setting name="ForwardUnhandledRequestsTo" value="" />
</ConfigurationSettings>
<Certificates>
  <Certificate name="CloudConnectorCertificate" thumbprint="DEE89C9EB6D087F081301F9D865189D1EE18C646" thumbprintIntalg="1" />
</Certificates>
</Role>
</ServiceConfiguration>
```

2. After changing the .CSCFG file to be the same, then you would upload your package to Azure:
3. In the Azure Portal, select CLOUD SERVICES and then your new service (mycloudconnector). Select DASHBOARD and UPLOAD A NEW PRODUCTION DEPLOYMENT. You will be prompted to upload an installation package.





4. Select the service package file (.CSPKG) on your computer and the service configuration file (.CSCFG).
5. The installation package will create the Cloud Connector service in your Azure account.

Configure the Cloud Connector Service

1. Once the Cloud Connector service has been created, you have to connect it with the SQL Azure database and the Azure Storage you created earlier. In Azure, with the Cloud Connector service selected, click on CONFIGURE.
2. Scroll a bit down to see the settings of fwxserverwebrole.
3. Update the DatabaseConnectionString and StorageConnectionString settings so that they refer to the database and storage created earlier, using the following steps:
4. To find the database connection string, select SQL DATABASES → connectorDB → View SQL Database connection string. A new window will pop up. Copy the first connection string for ADO.NET.

Connection Strings

ADO.NET:

```
Server=tcp:sx5bmhb4ql.database.windows.net,1433;Database=connectorDB;User ID=cloudconnector@sx5bmhb4ql;Password={your_password_here};Trusted_Connection=False;Encrypt=True;Connection Timeout=30;
```

ODBC:

```
Driver={SQL Server Native Client 10.0};Server=tcp:sx5bmhb4ql.database.windows.net,1433;Database=connectorDB;Uid=cloudconnector@sx5bmhb4ql;Pwd=
```

5. Paste the connection string into the DatabaseConnectionString setting, being sure to type in the password to the database instead of {your_password_here}.
6. To get the storage connection string, select STORAGE – cloudconnectorstorage and click MANAGE ACCESS KEYS on the bottom of the window. Copy the STORAGE ACCOUNT NAME and PRIMARY ACCESS KEY. Paste them into the StorageConnectionString setting, so you get something like this:

DefaultEndpointsProtocol=http;AccountName=cloudconnectorstorage;AccountKey=dFYljiK/yshGJSj64KLjdsGp3bgjKLj87423HsdjsjlOuu93jHJrJQIGIC4SA==

7. Click SAVE.
8. After the Cloud Connector service restarts, it will be ready to use.

Verify that the Cloud Connector Service Works

1. Type the URL of your Cloud Connector service (e.g., <http://mycloudconnector.cloudapp.net/>) into a Web browser. You should see this:

mycloudconnector.cloudapp.net - /

```
4/17/2013 9:46 AM <dir> bin
4/16/2013 6:15 PM 126 DebugService.svc
4/16/2013 6:15 PM 133 FwXMonitorService.svc
4/16/2013 6:15 PM 124 FwXServer.svc
4/16/2013 6:15 PM 727 Packages.config
4/16/2013 6:15 PM 8863 Web.config
```

2. Click FwXServer.svc. You should get this:

FwXCloudService Service

You have created a service.

To test this service, you will need to create a client and use it to call the service. You can do this using

```
svcutil.exe http://mycloudconnector.cloudapp.net/FwXServer.svc?wsdl
```

You can also access the service description as a single file:

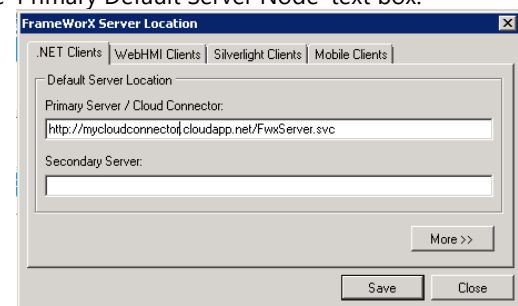
```
http://mycloudconnector.cloudapp.net/FwXServer.svc?singleWsdl
```

3. You may get another message, but so long as it refers to the service being created or active, you can proceed.
4. Now you are ready to set up the GENESIS64 client and servers to take advantage of Cloud Connector.

Client Setup

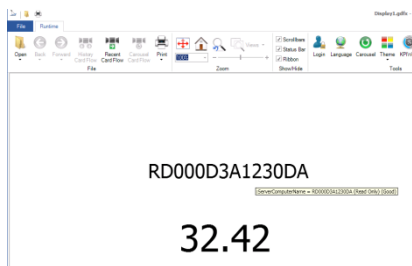
Set up the client computer to use Cloud Connector as its default FrameWorX Server:

1. Open the Default Server Location dialog (either Workbench → Tools → FrameWorX Server Location, or open Start Menu → All Programs → ICONICS → Tools → FrameWorX Server Location)
2. Type the URL of your Cloud Connector service (<http://mycloudconnector.cloudapp.net/FwXServer.svc>) into the 'Primary Default Server Node' text box.





3. You may leave the protocol and port number settings as they are. The client app will get all that information from the URL.
4. Click OK or Save.
5. If Cloud Connector is running, you should be able to run the client applications, but you will not get any data from FrameWorX Servers. Also, you may get control and simulation data from the Cloud Connector.
6. You should still see the same data browser information, but if the Cloud Connector does not have any information for that provider, then it will not show anything there
7. And, you should be able to get control and simulated data from Cloud Connector (points :ServerComputerName and svrsim:sine):



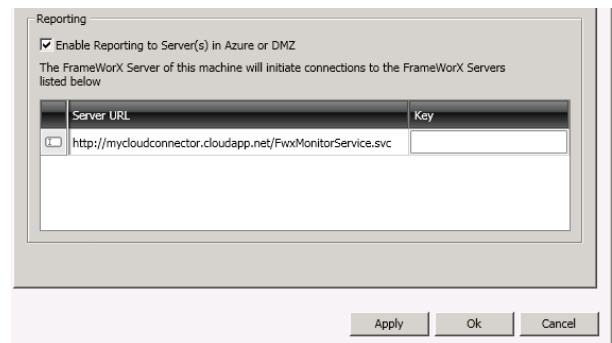
8. Note that 'ServerComputerName' can be browsed under Home → Control Information. Make sure these points work before proceeding.
9. If they do not work, check the TraceWorX log for GraphWorX64.

Setup FrameWorX Server on Premises

The following article explains how to setup FrameWorX Server on premises. FrameWorX Server will connect to the Cloud Connector

and send data to it. Cloud Connector will make the data available to its clients.

10. Open Platform Services configuration in Workbench (Tools → Platform Services) or open Start Menu → All Programs → ICONICS → Tools → Platform Services).
11. Select 'Enable Reporting to Server(s) in Azure or DMZ' on the Advanced tab.
12. Type the URL of the FwxMonitorService (http://mycloudconnector.cloudapp.net/FwxMonitorService.svc) into the 'New Server' text box and click '+' to add the server to the list of servers.



13. Click 'Ok'
14. Restart FrameWorX Server.
15. Repeat this for each FrameWorX Server you want to expose via Cloud Connector.
16. When your FrameWorX Server connects to Cloud Connector, you will be able to browse it on the client computer under the FrameWorX network
17. Now the FrameWorX Server on premises is available to clients connected to Cloud Connector.