zenon
by COPA-DATA

Manuel de zenon

MS Azure

v.8.20

Contenu

1 Welcome to COPA-DATA help

TUTORIELS VIDÉO DE ZENON.

Des exemples concrets de configurations de projets dans zenon sont disponibles sur notre chaîne YouTube (https://www.copadata.com/tutorial_menu). Les tutoriels sont regroupés par sujet et proposent un aperçu de l'utilisation des différents modules de zenon. Les tutoriels sont disponibles en anglais.

AIDE GÉNÉRALE

Si vous ne trouvez pas certaines informations dans ce chapitre de l'aide ou si vous souhaitez nous suggérer d'intégrer un complément d'information, veuillez nous contacter par e-mail : documentation@copadata.com.

ASSISTANCE PROJET

Vous pouvez obtenir de l'aide pour tout projet en contactant par e-mail notre notre service d'assistance : support@copadata.com

LICENCES ET MODULES

Si vous vous rendez compte que vous avez besoin de licences ou de modules supplémentaires, veuillezcontacter l'équipe commerciale par e-mail : E-mail sales@copadata.com.

2 MS Azure

MS Azure is the description for the online platform on which the cloud services can be rented (Microsoft Azure Cloud). This can be used for zenon and zenon Analyzer.

MS AZURE SERVICE BUS SAVE TYPES

The connection can be established with a Queue or an Event Hub:

- MS Azure Queues:
  - **Process Gateway**: In an MS Azure Queue, data is written to the memory in the order in which it arrives. The queue is emptied again when reading and emptied according to the settings. Data always lands in a defined queue from which it can be read again.
    If there is no Internet connection when writing, no data is written. These are also not cached or entered later.
    If there is no Internet connection when reading, this is not displayed. As soon as there is a connection again, all data present in the queue is read and then the display is updated.
  - **SQL Export**: Data is written to an archive queue. The cloud-based service Archive Worker writes the data to a cloud-based SQL database. If there is no Internet connection, the data is cached locally in the Runtime folder and uploaded in the next export cycle once there is a connection.
- MS Azure Event Hub:
  An MS Azure Event Hub writes data that has been received, subdivided in up to 16 partitions. This write process is quicker than writing to a Queue. zenon can use the **Process Gateway** to write data to an MS Azure Event Hub, but cannot read from it.

USE IN ZENON

In zenon, MS Azure serves as:

- SQL evacuation location:
  zenon archives can also be evacuated to MS Azure by means of SQL export.
- Cross-location exchange of data:
  Variables can be written to MS Azure Queues using **Process Gateway**.
  They can be read in again with the zenon **AzureDrv** driver. In doing so, no extra ports need to be configured for the incoming data.
- Data storage location for third-party applications:
  The zenon **Process Gateway** can also write data to MS Azure Event Hubs. This data can thus be provided for other applications. However, it cannot be read again by zenon.

USE IN ZENON ANALYZER

In zenon Analyzer, an MS Azure SQL database can be used as a **linked server**.

3 Configuration of MS Azure

The configuration depends on the use in zenon.

MS AZURE

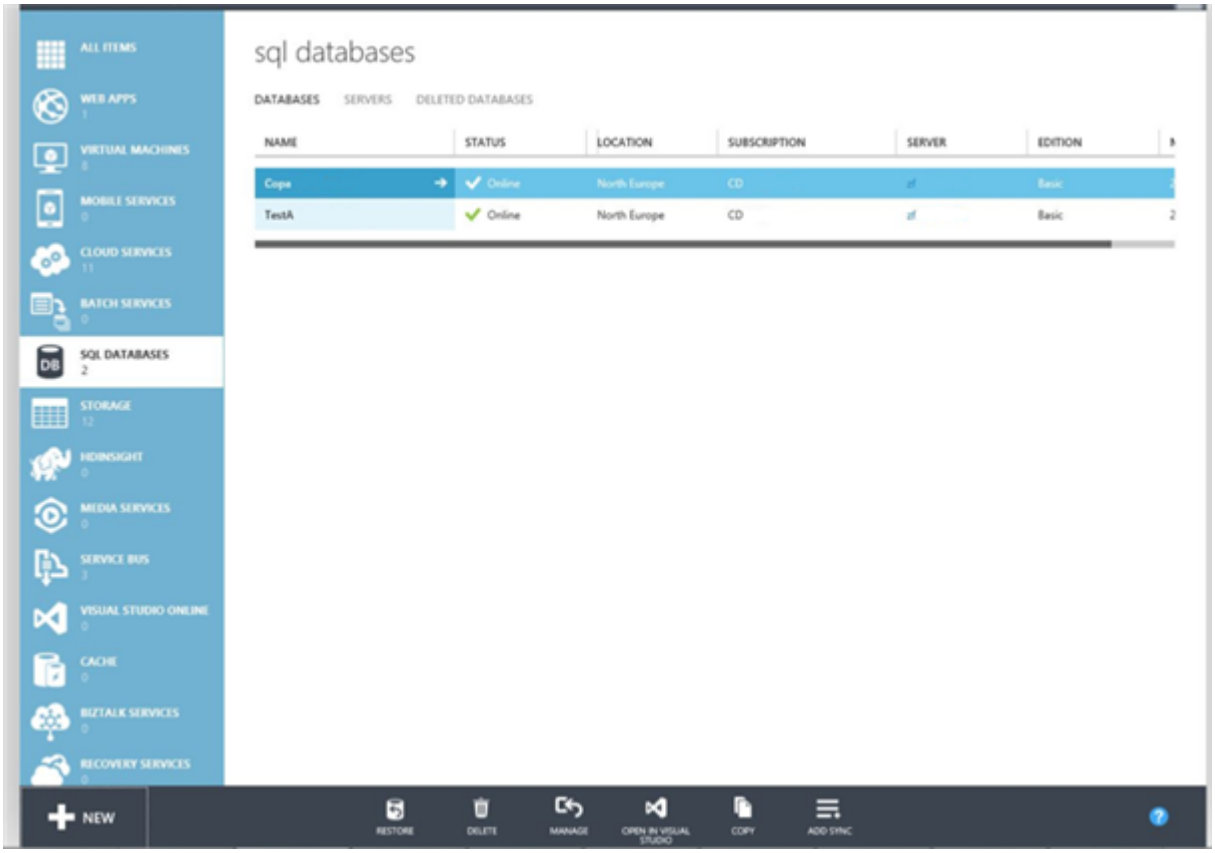
In this section, you receive information on configuration steps in MS Azure when using zenon. You can get information on the general configuration of MS Azure and databases in MS Azure from the Microsoft MS Azure help.

For use with zenon, you need the following in MS Azure, depending on how it is used:

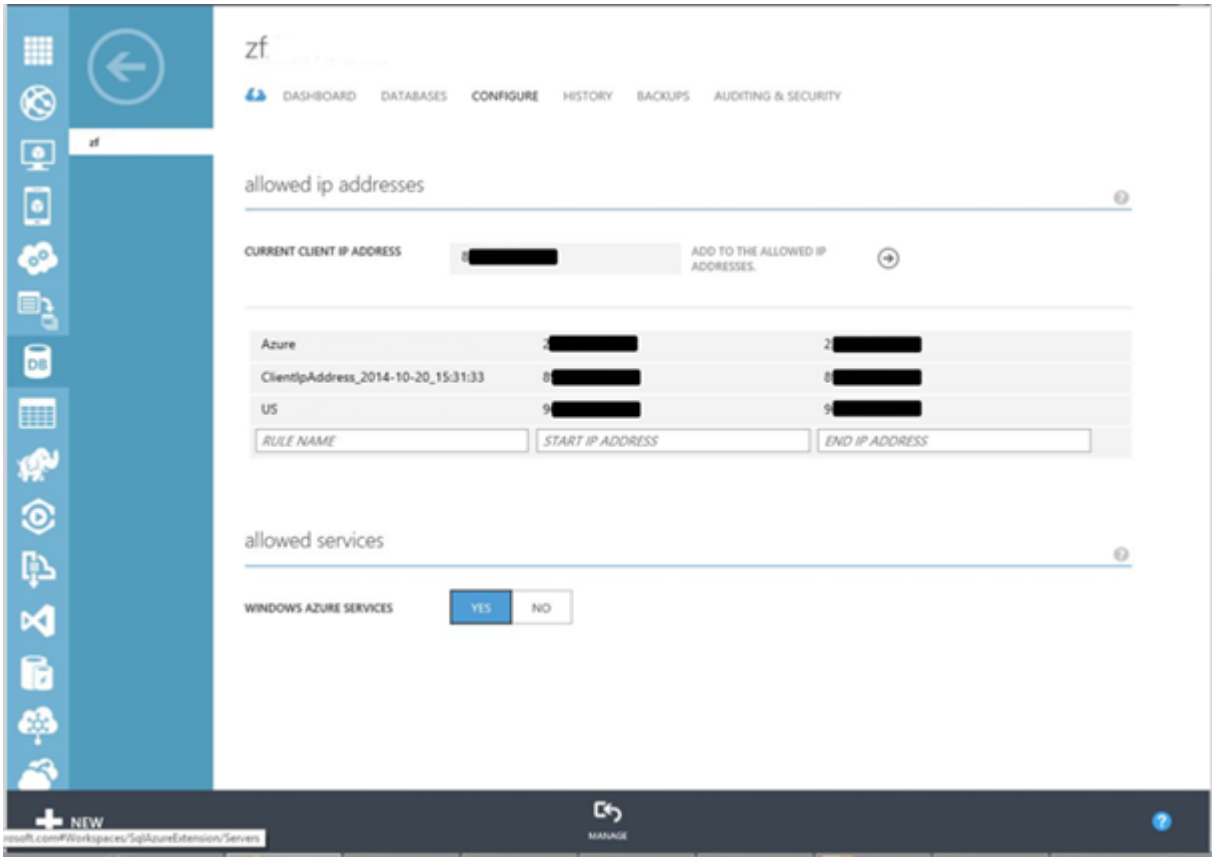
- MS Azure SQL Server
- An assigned database
- The server name and Connection String of the database
- The Namespace with
  - Queues
    or
  - Event Hubs
- The **Archive Worker** cloud service

zenon archive can be evacuated to MS Azure and read back again. Reading in can be via zenon Analyzer.
To evacuate archives and read them in again:

1. Create an SQL server and corresponding database in MS Azure, in which the data is to be saved.



2. Ensure that the Public IP of the accessing computer is included in the Allowed IP Adresses of the server.



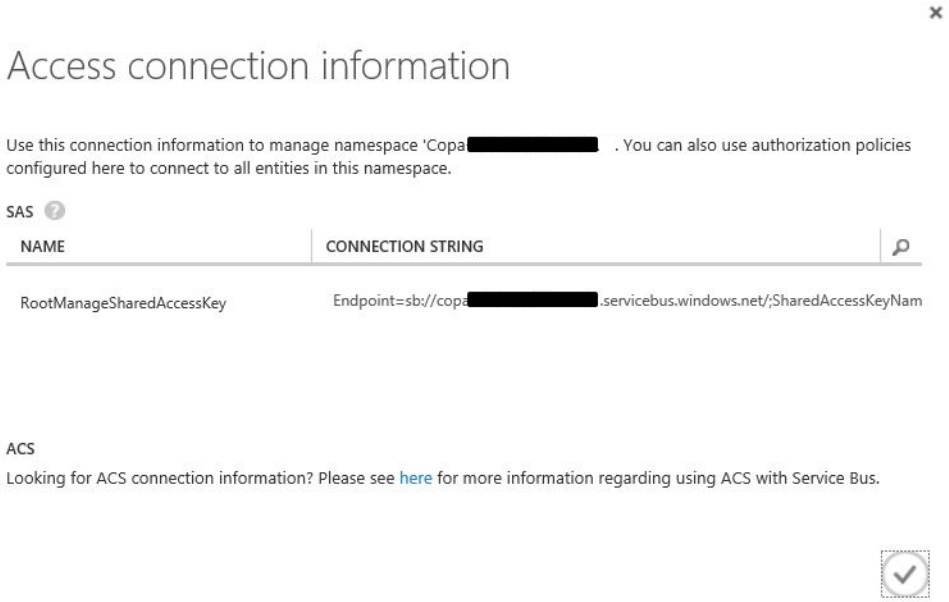
3. In the options of the newly-created database, you have the possibility of copying the **Connection String**.
   To do this, click on **View SQL Database connection strings**.

   You need this **Connection String** later for the configuration file (à la page 13) **ServiceConfiguration.Cloud.cscfg**.

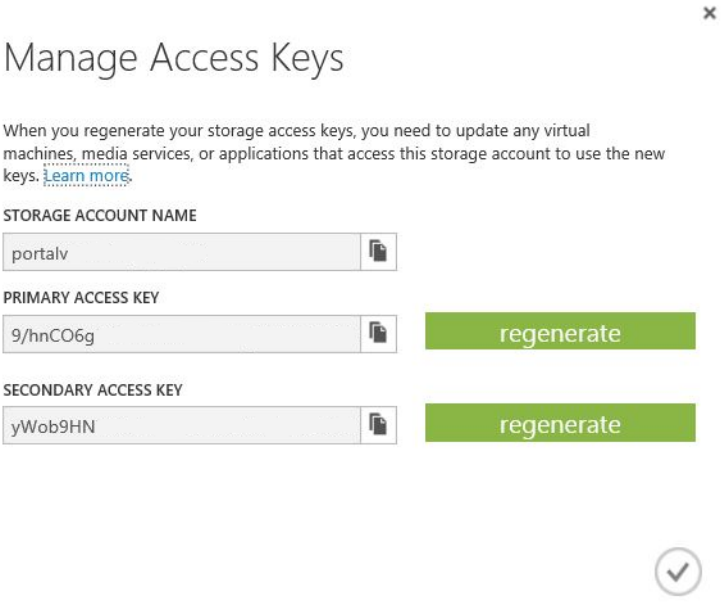
zenon needs the **ADO.NET** Connection String.

4. Create and configure a new namespace in the Service Bus menu.
   You need the *Messaging* type. This cannot be subsequently changed.
   Further Shared Access Keys can subsequently be created.
   You need this **Connection String** later for the configuration file (à la page 13) **ServiceConfiguration.Cloud.cscfg**.

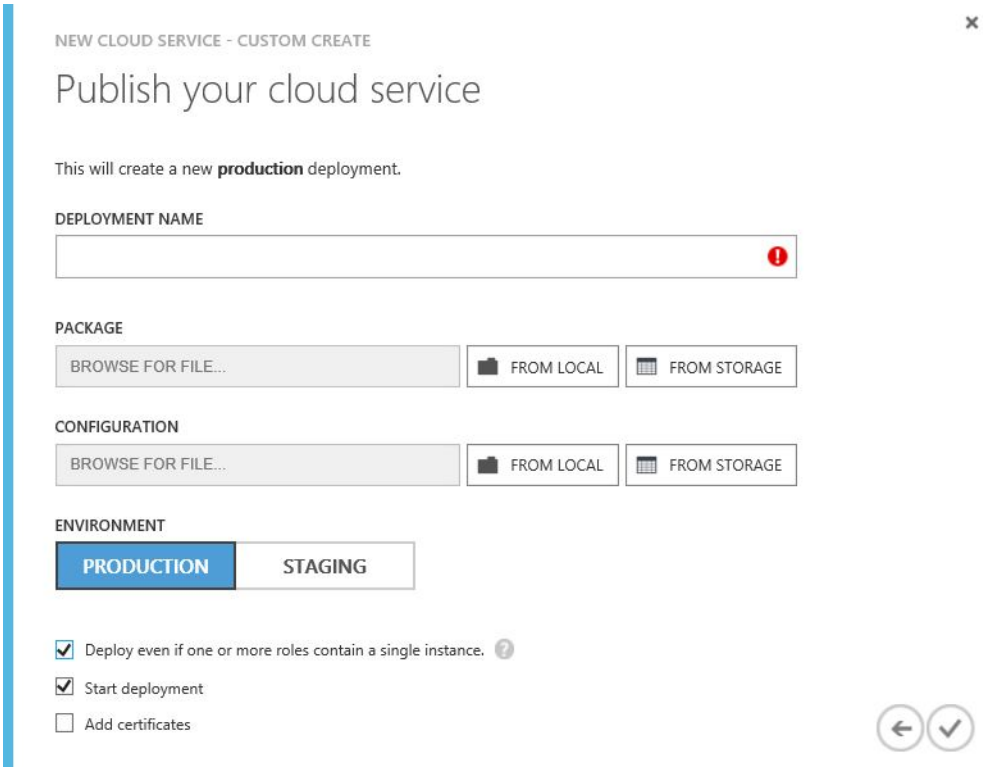
   Note: Ensure that each computer that writes to the Queue has write authorizations.



5. Configure the Access Keys.
   You need this **Storage Account Name** and the **Access Key** later for the **ServiceConfiguration.Cloud.cscfg** configuration file (à la page 13).



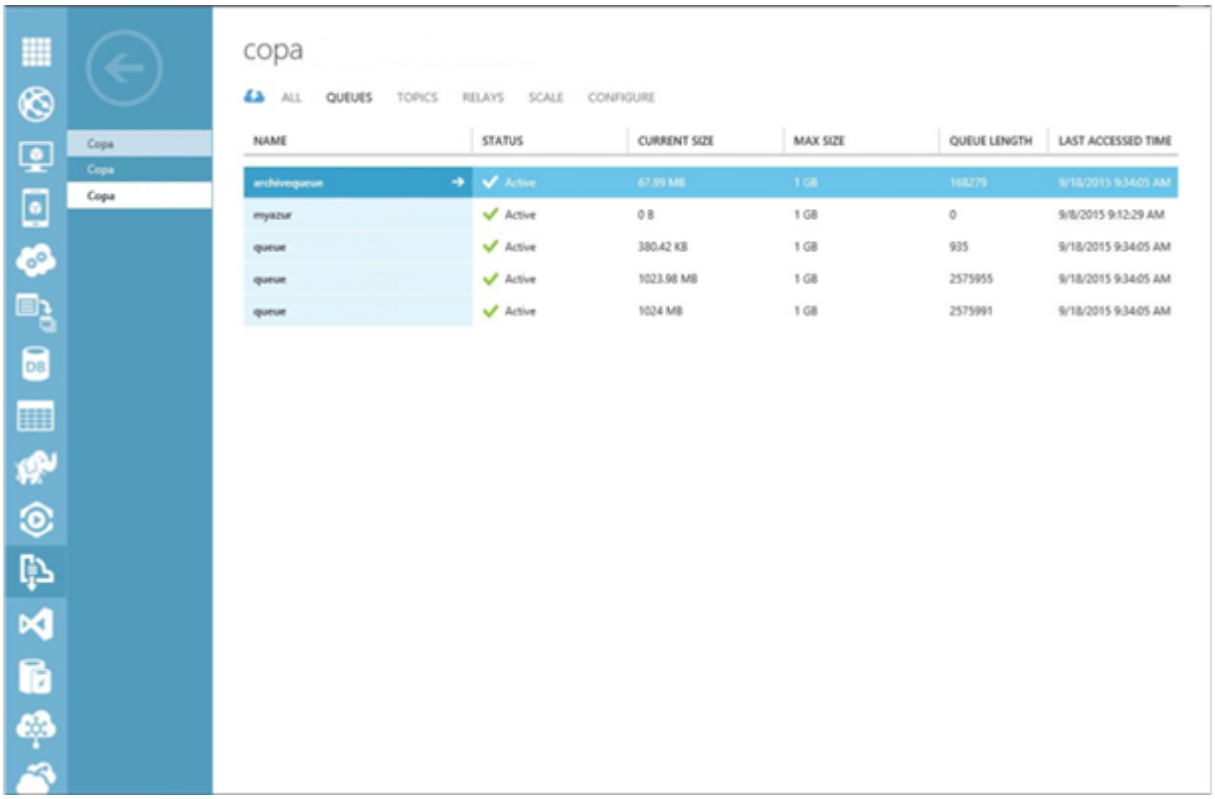
6. Create a new cloud service.
   In doing so, the Archive Worker is created.
   You can find the required files in the following folder:
   %ProgramData%\COPA-DATA\zenon8.20\CloudService.



7. Create the desired Queue or the desired Event Hub.

If a zenon archive is to be evacuated, this Queue must have the name **archivequeue**. It is automatically created with the Archive Worker.

Recommendation: To configure the zenon SQL evacuation, use the supplied zenon Package (see **evacuate SQL archives** (à la page 13) section).



4 Configuration in zenon

The configuration in zenon or zenon Analyzer depends on the task:

- Evacuate SQL archive (à la page 13): zenon archives are evacuated to MS Azure by means of SQL.
- Save data and read it in again (à la page 14): Variables are saved in Azure in a Queue or an Event Hub using zenon **Process Gateway** Data from a Queue is read back into zenon using the zenon **AzureDrv** driver.

In zenon, you generally need the following information from MS Azure for configuration:

- Connection String (server connection name)
- Name Queue or Event Hub
- Password

Note: Ensure that the necessary ports in the firewall are unlocked.
Default: *1433*

5 Evacuate SQL archives

There is a configuration package available on the zenon installation medium to configure the evacuation of archives.
Path: **%ProgramData%\COPA-DATA\zenon8.20\CloudService**

It contains the following files:

- **AzureArchiveCloudService.cspkg** (package)
- **ServiceConfiguration.Cloud.cscfg** (Configuration file)

Configuration:

- There must be a **Queue** with the name **archivequeue** in the **Namespace** on the MS Azure server.
- The following must be entered in the zenon configuration file:
  - Path to the Service Bus (**Connection String**), in which the **Archive Queue** is automatically created
  - **Connection String** to the Azure SQL database
    Attention: A password must be provided
  - ArchiveStorage and Diagnostics: **Connection String** to a Table Storage in MS Azure. Can access the same Table Storage.
- Configuration of the SQL evacuation in zenon
  - **Save** tab for **archive properties**:

    Write: Entry of the Connection Strings to the MS Azure Service Bus in the **Use MS Azure service bus for writing**.

    Read: Establish connection to the MS Azure database using the **SQL database** option.

5.1 SQL evacuation in zenon

If the **Use MS Azure Service Bus for writing** option has been activated for the SQL evacuation of an archive, all archive values in the Azure Service Bus Queue are added to the Azure service bus queue with the name **archivequeue**. This must exist in the MS Azure-Namespace of the configured connection. The MS Azure connection name is entered in the input field under the option.

CONFIGURATION

In MS Azure, there must be at least one instance of an **AzureZenonArchiveWorker** cloud service running, which receives the archive values from the queue with the name **archivequeue** and inserts these into the MS Azure SQL storage. The table format corresponds to the classical SQL evacuation of an archive.

The cloud service must be displayed manually with the **AzureArchiveCloudService.cspkg** deployment package via the MS Azure configuration user interface. You can find the package in the following folder: **%ProgramData%\COPA-DATA\zenon8.20\CloudServices**.

Settings:

- **Zenon.ArchiveServiceBus.ConnectionString**: Corresponds to the MS Azure connection name in archive configuration.
- **Zenon.ArchiveSQLServer.ConnectionString**: Denotes the name for the **MS Azure SQL storage** destination.

PROCEDURE

The archive files are read via an OLEDB connection and SQL SELECT statement. The OLEDB connection name therefore generally shows the same MS Azure SQL-Storage as in the output connection name (**Zenon.ArchiveSQLServer.ConnectionString**) in **AzureZenonArchiveWorker**.

6 Saving data and reading it in again

Data can, for example also be saved throughout locations using the MS Service Bus. To do this, it is uploaded using the zenon Process Gateway and downloaded again using the AzureDRv driver. One of the advantages is that no exceptions need to be configured in the firewall when downloading. It is also possible to provide data for third party applications.

Two settings are possible for the target in the **Namespace**:

- **Queue**: Data can be saved and called up again.
- **Event Hub**: Data is saved for third-party applications. These can no longer be read by zenon or zenon Analyzer.

6.1 Process Gateway: Saving data in MS Azure

In Runtime, the **AccessAzure** module in **Process Gateway** can be used to upload data to MS Azure.

The **Process Gateway** uses the **AccessAzure.dll** to establish a cyclical connection to MS Azure. Services in MS Azure supported by Process Gateway:

- Servicebus Queue
- Event Hub
- IoT Hub

In addition, the Process Gateway can receive data from the IoT hub via *Cloud2Device-Message*.

Data from the **Queue** can then be obtained with the **AzureDrv**driver from MS Azure and integrated into zenon processes. **IoT Hub** content can only be read by the driver if it is transferred to the **Servicebus Queue**. This can be implemented with the standard services of the MS Azure Cloud.

Data from an **Event Hub** is for third-party applications and can no longer be read by zenon.

DRIVER COMMUNICATION

To read the values from the Servicebus Queue, the **AzureDrv** driver establishes a connection to the service bus Queue with the configured name and takes all messages received from it. These messages are unpacked and the online values contained therein are allocated to the variables. The key for this is the **Adresse symbolique**.
All messages that are already in the queue when the connection is first successfully established are loaded and discarded. It is always only the current values that are displayed.

This means:

Each driver instance on each computer has its own **Queue** as an input signal.

Exemple

The Servicebus Queue for the computer **MYSERVER1** and the configured prefix **onlinedata** is called the following in MS Azure: **onlinedata_myserver1**.

Numeric *(DOUBLE)* and alphanumeric *(STRING)* values are supported. The time stamp and the system status bits are transferred to the target variable.

Informations

You can find details on configuration and formats in the **Process Gateway** manual in the **MS Azure** section.

CONFIGURATION IN THE PROCESS GATEWAY

Configuration dialog for MS Azure connection via Process Gateway.

VARIABLES

In the **Variables** group, you configure the variables whose values are saved by zenon in an MS Azure Service Bus.

Parameter	Description
Available in Scada	Displays all variables available in zenon With multi-project administration, variables from active projects can be selected.
Projects	List of all available projects. The standard project is marked with a *.
Variables	List of all variables of the selected project. List can be sorted; multiple selection is possible. **Hint:** Double clicking on the variable moves it.
Button >	Selected variables from the list of variables are moved to the **Exported into MS Azure** list.
Button <	Selected variables are removed from the **Exported into MS Azure** list.
Exported into MS Azure	List of the variables that are written to the MS Azure Service Bus by the Process Gateway. These are displayed with name (**Points**) and communication **Direction**. Double click on an entry to reverse the direction of communication. The key for the values in MS Azure is always `PROJECTNAME#VARIABLENAME.` **Points:** The name consists of: ▸ Project name ▸ **#** (as separator) ▸ Variable name. **Direction:** Shows the direction of communication: ▸ *read only* ▸ *write only* ▸ *read/write*

Parameter	Description
	Par défaut : *read only* Change the direction by double-clicking on the variable. **Note:** Only linked variables can be described.

MS AZURE CONNECTIONS

All target connections in which the current values of the selected variable are to be inserted in MS Azure are to be entered into the **MS Azure Connections** group.

In doing so, the current variable values are added to **all** existing connections at the same time.

Parameter	Description
List of MS Azure connections	Lists all connections to MS Azure configured. Each connection consists of the connection name (MS Azure Connection) and the Service Bus Queue Name. ▸ Creation of a new connection with the button **New...** ▸ A selected connection can be amended with the **Edit...** button.
MS Azure Connection	**MS Azure** connection address.
Service Bus Queue Name	Name of the queue in the MS Azure service bus.
New...	Opens dialog to configure the **MS Azure** connection.
Edit...	Opens existing connections to configure the **MS Azure** connection.
Delete	Deletes the selected **MS Azure** connection from the list.

MS AZURE SETTINGS

Parameter	Description
Integrity period	Time interval in which the current values of the selected variables are written as an image to the **MS Azure Service Bus** queue.

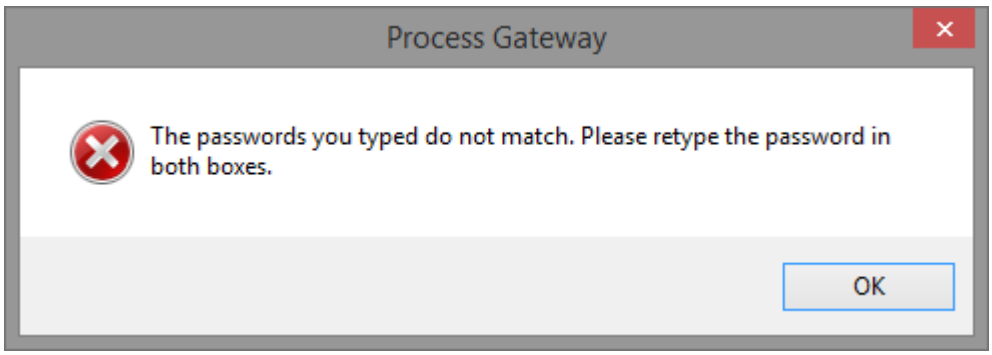
Parameter	Description
	If the value of a variable changes during this this interval, the value change is immediately transferred to the **MS Azure** queue. Par défaut : *5 s*
HTTP Proxy Domain	Address of the proxy server in the network.
HTTP Proxy User	Input field of the user name for login on the proxy server.
HTTP Proxy Password	Input field for password for login on the proxy server. **Note:** Input is shown with dots - even during entry.
Confirm HTTP Proxy Password	Input field for the confirmation of the password for login on the proxy server. **Note:** Input is shown with dots - even during entry.

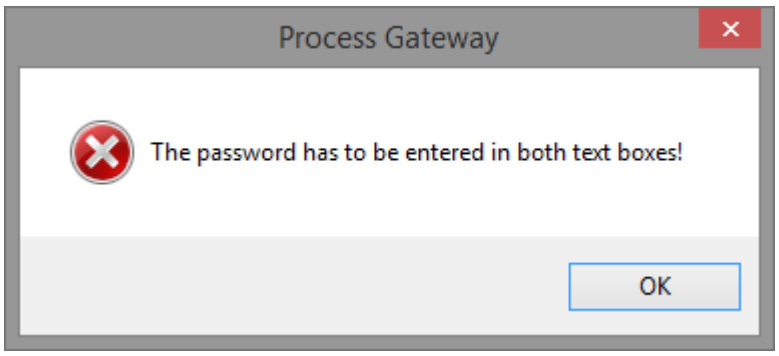
NAVIGATION

Parameter	Description
OK	Applique les paramètres et ferme la boîte de dialogue.
Cancel	Annule toutes les modifications et ferme la boîte de dialogue.

DIALOG IN THE EVENT OF INCORRECT ENTRIES

Configurations of the proxy server are validated. A corresponding warning dialog is shown in the event of an incorrect configuration.





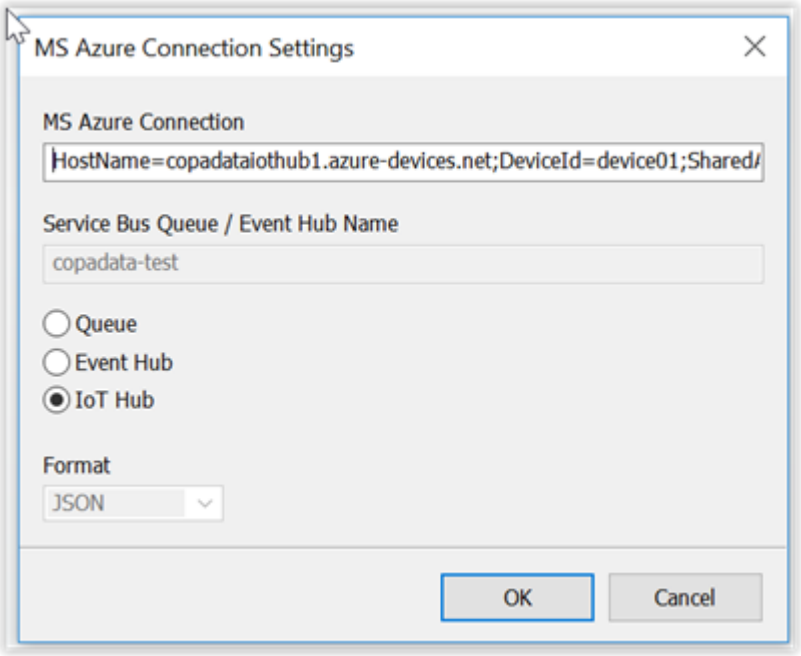
Attention

.NET 4.5 must be installed on the computer in order for the proxy settings to work.

CONFIGURATION OF CONNECTION TO QUEUE OR EVENT HUB

In the connection dialog, you stipulate whether the connection to a Queue or an Event Hub is made and how this is addressed.

Configuration dialog for the connection to **MS Azure**:



Parameter	Description
MS Azure Connection	**MS Azure** connection address. **Note:** You can read and copy this address in the Azure administration portal under **Manage Connection Strings** of the desired **Servicebus Namespace**.

Parameter	Description
Service Bus Queue / Event Hub Name	Name of the **MS Azure Service Bus** queue or the **Event Hub**. Selection by means of radio buttons. **Note:** Only characters that appear in the name of a MS Azure Service Bus Queue or in an Event Hub are permitted. Use simple, short and meaningful names. Avoid special characters, language-specific letters and blank spaces. **Service Bus Queue** The name of the **Service Bus Queue** comprises: ▸ A freely-configurable prefix ▸ An underscore (_) ▸ The NETBIOS computer name (without domain name suffix) in small lettering If the name does not yet exist in the Namespace a Queue with this name is created. Requirement: The corresponding rights are present. **Event Hub** ▸ **Message format**: Select in the drop-down list in the **Format** option. ▸ **Event Hub Name**: Name of the Event Hub in the MS Azure Service Bus The Event Hub must already be created with this name in the Namespace. **Note:** Grayed out if the **IoT Hub** option is activated as the type of communication to MS Azure.
Queue	Communication to MS Azure via *Service Bus Queue*. ▸ *Active*: The connection is established using the name of the **Service Bus Queue**.
Event Hub	Communication to MS Azure via *Event Hub*. ▸ *Active*: The connection is established using the name of the **Event Hub**. Selection of the message format using the **Format** option.

Parameter	Description
IoT Hub	Communication to MS Azure via *IoT Hub*. ▸ *Active*: Sending of actual values and the receipt of messages via *IoT Hub*. Entry of the *Device-Connection Strings* in the **MS Azure Connection** input field.
Format	Message format for connection via **Event Hub**. Select from drop-down list: ▸ *XML* ▸ *JSON* ▸ *BOND* (compact binary)
OK	Applique les paramètres et ferme la boîte de dialogue.
Cancel	Annule toutes les modifications et ferme la boîte de dialogue.

Informations

The Service Bus Queue Name can be freely configured.
This queue is automatically created in MS Azure during the first communication to MS Azure.

For each computer that calls up data from MS Azure, use the **AzureDrv** driver to create a separate MS Azure Connection.

6.2 zenon AzureDrv driver: Read data from MS Azure

The AzureDrv driver gets data from the MS Azure Service Bus for processing in zenon.

CONFIGURATION

Configuration dialog of the connection to the MS Azure service bus:

Parameter	Description
[Connection type]	Selection of the connection type via radio button. ▸ *Queue* Communication to MS Azure via *Service Bus*

Parameter	Description
	Queue. The connection is established using the name of the **Service Bus Queue**. ▸ *Event Hub* Communication to MS Azure via *Event Hub*. The connection is established using the name of the **Event Hub**. ▸ *IoT Hub* Communication to MS Azure via *IoT Hub*. Sending of actual values and reception of messages via *IoT Hub*.
MS Azure Queue Connection	Input of the name of the connection to the MS Azure Servicebus. You can read and copy these connection names in the MS Azure administration portal under **Manage Connection Strings** of the desired *Servicebus Namespace*. If the **IoT Hub** option is activated: The connection name to the MS Azure endpoint for access to read the data of an IoT Hub in the MS Azure Portal. You can read these connection names in the MS Azure Portal. Configuration in the MS Azure Portal: ▸ In the portal, go to the **All services** node. Select the **IoT Hub** option. ▸ Select an existing Iot Hub or create a new IoT Hub. At least one *IoT Device* must be configured for this IoT Hub. ▸ The **Build-in endpoints** can be found in the IoT Hub settings. ▸ Go to the **Built-in Endpoints** option to access the IoT Hub. Copy the name from the event configuration *Event Hub-compatible endpoint*. Attention: The "*;EntityPath=…*" text component is not part of the zenon driver connection string and may not be added for this option. **Example:** ▸ MS Azure Portal: *Endpoint=sb://ihsuprodamres040dednamespace.ser*

Parameter	Description
	vicebus.windows.net/;SharedAccessKeyName=iothubowner;SharedAccessKey=vbR6F7HvHGgRZ6ld0VCY1bDxo8SyMgGiHHGRmO6N4VU=;EntityPath=iothub-ehub-myiothub-1657833-791bd2d757 ▸ zenon driver: *Endpoint=sb://ihsuprodamres040dednamespace.servicebus.windows.net/;SharedAccessKeyName=iothubowner;SharedAccessKey=vbR6F7HvHGgRZ6ld0VCY1bDxo8SyMgGiHHGRmO6N4VU=*
Service Bus Queue Prefix / Event Hub Name	Input of the prefix for the name of the **Queue** or the *Event Hub name* that is to be queried in Runtime. **Note:** Only characters that appear in the name of an MS Azure-Servicebus Queue are permitted. If the **IoT Hub** option is activated: ▸ Go to the **Built-in Endpoints** option in MS Azure Portal. ▸ Copy the name from the *Event Hub-compatible name* event configuration. This also matches with the *;EntityPath=...* entry of the event configuration **Example:** *iothub-ehub-myiothub-1657833-791bd2d757*
MS Azure IoT Hub Connection	service connection string to the IoT service of MS Azure. Used for the secure login of the driver (service) to the IoT Hub endpoint. When configured correctly, values can also be written. Configuration in the MS Azure Portal: ▸ In the portal, go to the **All services** node. Select the **IoT Hub** option. ▸ Select an existing Iot Hub or create a new IoT Hub. At least one *IoT Device* must be configured for this IoT Hub. ▸ Select the configured IoT Hub which communicates with the driver. ▸ The **Shared access policies** can be found in the

Parameter	Description
	IoT Hub settings. ▸ Select the *service* entry under **POLICY**. ▸ Make sure that the **Service connect** checkbox is selected under Permissions. This option enables values to be written via the IoT Hub endpoint. ▸ Copy the configuration of the *Connection string--primary key*, for example, via copy & paste. **Example:** *HostName=myiothub.azure-devices.net;SharedAccessKeyName=service;SharedAccessKey=PdRurWJzlOqUpwnCgAe+iJzZBDeAn7O5D2GHOe/l6EI=* **Note:** only active if **IoT Hub** is selected.

FERMER LA BOÎTE DE DIALOGUE

Option	Description
OK	Applique toutes les modifications effectuées sur tous les onglets, puis ferme la boîte de dialogue.
Annuler	Annule toutes les modifications effectuées sur tous les onglets, puis ferme la boîte de dialogue.
Aide	Ouvre l'aide en ligne.

You can read further details in the **AzureDrv** manual.

7 zenon Analyzer - Linked MS Azure Server

Data from Linked SQL Servers can be edited in zenon Analyzer. These can also be located in MS Azure.

Configuration of a linked Microsoft Azure server

1. In the dialog to administer the linked server, click on the **New Microsoft Azure Server** button.
   The dialog for configuring a server is opened.
2. Assign a name for the server.
3. Enter an instance name.
   Note: You can find this on Microsoft Azure.

4. Enter the name of the database.
5. Enter the user name for access.
6. Enter the password for access.
7. Click on **OK**.

   The linked server is created and the dialog for creation is closed.
   The new linked server is then given a connection test.
   If the test is not successful, there is a query asking whether the new settings are to be retained:

   - **Yes**: Settings are retained and are displayed in the list.
   - **No**: Settings are discarded and the server is removed from the list.

   > **Informations**
   >
   > Each database on an SQL Azure instance must be added as its own linked server for technical reasons.

DIALOG CONFIGURATION

Option	Description
Linked Server name	Name of the linked server.
Microsoft SQL Azure instance name	Name of the SQL Azure instance. Can be read from Microsoft Azure. Syntax: **[any desired character sequence].[database].windows.net**
Database on Microsoft SQL Azure instance	Entry of the database name. Can be read from Microsoft Azure.
User name	Entry of the user name.
Password	Password. Is not displayed in plain text.
OK	Applique les paramètres et ferme la boîte de dialogue. A connection test is carried out afterwards. If the connection cannot be established, the option to reject the configuration is offered.
Cancel	Annule toutes les modifications et ferme la boîte de dialogue.

You can read more about the Linked Server in zenon Analyzer in the **zenon Analyzer** manual in the **Manage linked servers** section.