



Pushing Performance



People | Power | Partnership

HARTING OPC UA Gateway for MICA User Manual

1st Edition 2020

© peraMIC GmbH & Co. KG, Espelkamp

All rights reserved, including those of the translation.

No part of this manual may be reproduced in any form (print, photocopy, microfilm or any other process), processed, duplicated or distributed by means of electronic systems without the written permission of peraMIC GmbH & Co. KG, Espelkamp.

Subject to alterations without notice.



Content

1	General	4
1.1	Release Notes for Version 3.0.1	4
2	Installation	5
3	Basic Container Operations	6
3.1	Network Settings	6
4	About OPC UA	7
5	Using the OPC UA Gateway	8
5.1	Devices	8
5.2	Reports	10
5.2.1	Adding a Subscriber	11
5.2.2	Deleting a Subscriber	11
6	Executor	12
6.1	Error messages	12
7	Working with Logs	13
8	OPC UA Gateway REST Interface	14
8.1	Quick REST Test Using a Browser	14
8.2	Using REST from Inside a Container	14

1 General

The OPC UA Gateway container lets you monitor multiple OPC UA devices and publish that data to a set of subscriber endpoints or display them on a dashboard.

1.1 Release Notes for Version 3.0.1

The OPC-UA Gateway container version 3.0.1

- Requires MICA Base Firmware 2 or higher. Firmware 9 is recommended.
- Supports only web browsers with fully ES6 support, for example the latest versions of Google Chrome or Microsoft Edge.
- Requires a license key for continuous operation. Without a license key, the container will be stopped automatically after 6 hours.

New Features

- Improved user and rest-based web interface.
- Tree walk mechanism for the device definition. A user can now browse the fields/nodes of an OPC-UA server.
- Fields and nodes now can be subscribed or polled cyclically.
- Device properties for subscription settings to device definition.
- Support for additional datatypes i.e. unsigned numbers and numerous complex types.
- Users can add all fields of a device at once using the *Add All* button.
- Namespace are now supported in the *Executor*.
- Added support to execute methods on OPC-UA nodes.
- Updated and improved subscriber mechanism.
- Integrated log component.

Bug Fixes

- Increase runtime performance
- Improve stability

2 Installation

To install the OPC UA Gateway on a MICA:



1. Log in to the MICA with admin rights.
2. Click *Install*.
3. Click *Select File* and select the installation archive.
4. Click *Execute* to start the installation.
5. The installer will display the readme file with information about the installation archive.
6. Enter the name you want to install the container as

An old container archive format has been detected. Please enter a name for the container to be installed.

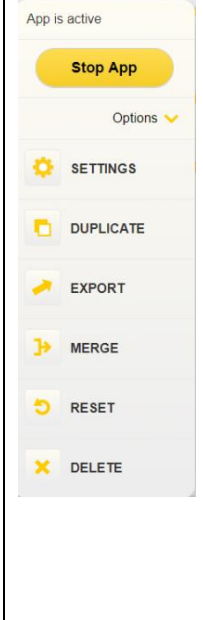
Container Name:

7. Wait until the installation is finished and click *Close*.

The installed container is initially turned off. Right click on the container and press *Start App* button to start the container.

3 Basic Container Operations

Right clicking the OPC UA Gateway container tile opens the container context menu. Press *Options* to access the following functions:

	<ul style="list-style-type: none"> • <i>Start App / Stop App</i>: Starts or stops the container. • <i>Options</i>: Expands or hides the basic container functions. • <i>Settings</i>: Shows the container information and lets you configure the IPv4 / IPv6 settings, an Additional Network Interface and the Single Sign On Mode. • <i>Duplicate</i>: Duplicates the container on your MICA. • <i>Export</i>: Exports the container to your PC or a network drive. All configurations you set will be kept. • <i>Merge</i>: Overwrites the reset point of the container with its current configuration. • <i>Reset</i>: Resets the configuration of the MICA Store Service to the last reset point (factory default if no merge was executed before). • <i>Delete</i>: Deletes the container including all its configuration and user data.
--	---

3.1 Network Settings

In the container’s context menu, click *Settings* to configure the container’s network settings and confirm by clicking the *Activate Settings* button. The container may restart to apply the new settings.

4 About OPC UA

The OPC United Architecture is a machine to machine communication protocol for industrial automation, providing a platform independent, service-oriented architecture that integrates all the functionality of the OPC Classic specifications into one extensible framework.

This multi-layered approach accomplishes the original design specifications goals of:


1. Functional equivalence: all COM OPC Classic specifications are mapped to UA
2. Platform independence: from an embedded micro-controller to cloud-based infrastructure
3. Secure: encryption, authentication, and auditing
4. Extensible: ability to add new features without affecting existing applications
5. Comprehensive information modelling: for defining complex information

Detailed information about the OPC United Architecture is available at <https://opcfoundation.org/>

5 Using the OPC UA Gateway

To open the Gateway user interface, click its container tile after the container has started and the tile has changed from gray to yellow.

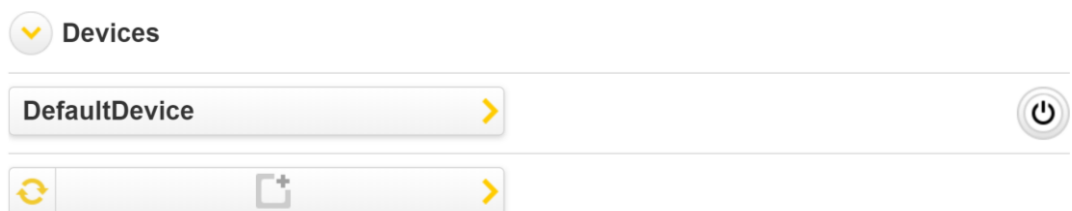


The  **Activate** button with an exclamation mark indicates that the OPC UA Gateway container is not yet licensed and running in demo mode. In this mode, the container will be stopped every 6 hours, but can be restarted manually.

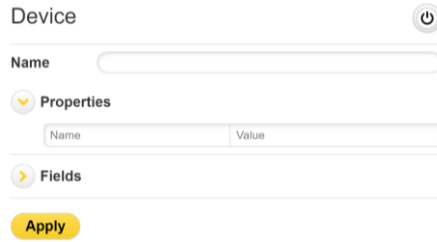
To obtain a license key, click on *Activate* and follow the steps described in the popup window.

5.1 Devices


In the section Devices you can create, edit and import OPC UA devices from which you want to collect data.



To add a new device, click the add button and configure the device in the pop up window.



Name is the displayed name of the device. The name will be also displayed in the section Reports and Executor.

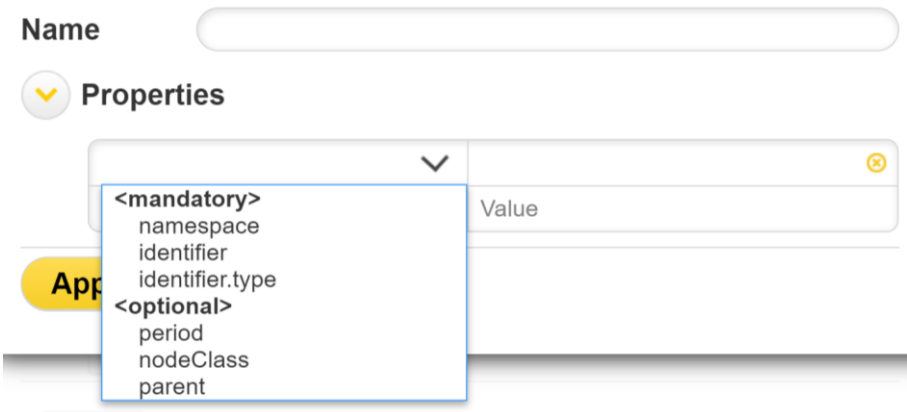
Properties lets you define a list of mandatory and optional properties. Mandatory for the connection is the host. There are several optional properties like the port or credentials for authorization. Properties can be added with a click on the Name | Value button or deleted with a click on .

Clicking on the *Field* box to open a dialog that lets you create a field for the device. new dialog.



If the OPC UA Gateway is connected to a device, you can also click the magnifying glass to receive a list of supported fields from the device.

Field



Name is the display name of the field. The name will be also displayed in the section Reports.

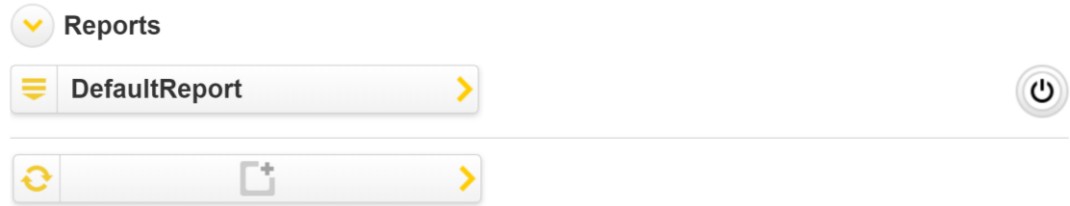
Properties is a list of mandatory and optional properties for the field. The mandatory properties are namespace, which sets the namespace index of a node in an OPC UA server, identifier, which sets the identifier for the node and identifier type, which sets the type of the nodeId. The supported types are string and numeric.


Optionally the period property can be set to define the sampling rate for the field in milliseconds.

When you have created the fields, you need and entered all mandatory properties click on the *Accept* button to add your device to the list.

5.2 Reports

The section *Reports* lets you set up reports containing machine data that will be sent to one or more *Subscription* endpoints.



To add a new report, click on the *New Report* button  and fill in the required information in the configuration window.

Report



Name

Duration


Repeat Period

Fields


Device	Field

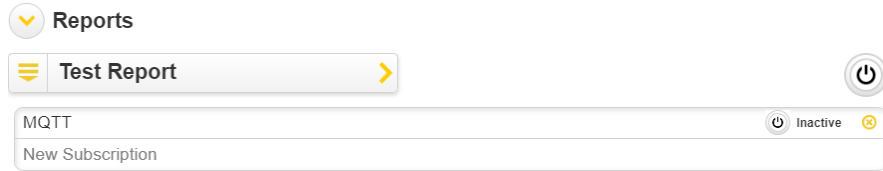
Apply

- A unique *Name* for the report.
- The *Duration* of the report in milliseconds, i.e. how long the report should run.
- The *Repeat Period* of the report in milliseconds, i.e. the time interval after which the report should start again.
- Fields is for choosing your device you added in the section *Devices* and the *Field* you want to observe. You can add new fields by clicking on the *Device* | *Field* buttons. If you want to add all fields of a device, click *Add All Fields* in the drop-down selector of the *Field* button.

You can delete an existing *Device* | *Field* pair by clicking on the  button. Click *Apply* to apply the settings and close the window.


5.2.1 Adding a Subscriber

To add a new *Subscription* endpoint, click on the small icon  on the left side of the report and then click on *New Subscription*.



A dialog will appear where a subscription can be configured.

If no subscriber was configured before, a new one can be configured by selecting *New*


Subscription 

Name

Subscriber

Topic Extension

Properties

Property Key	Property Value	
		

Apply

Subscriber ... in the *Subscriber* box.

A dialog will appear where you can add a new subscriber endpoint to publish the gathered information from the injection molding machine. The supported types of endpoints are AZURE, HTTP, HTTPS, JDBC, MQTT, MQTTS, TCP and UDP.

Subscriber

Type

Host

Port

Fragment

Username

Password


Topic Prefix

Client ID

QoS



URI

Properties

Property Key	Property Value	
		

Apply

Leave the Client ID blank to use an autogenerated ID. Otherwise, make sure to use a unique ID.

Once configured, the Subscription endpoints can be activated and deactivated with a click on  or deleted by clicking .

5.2.2 Deleting a Subscriber

You can delete a subscriber by moving your mouse over on a subscriber and clicking the *Delete* button.

6 Executor

The *Executor* is a small tool to easily read data from or write data to devices which are connected to the OPC UA Gateway. You can also use the *Execute* operation to run OPC UA commands on an OPC UA node connected to the MICA.

▼ **Executor**

Device URL

Identifier

Operation Namespace Type

Result

Data

Execute

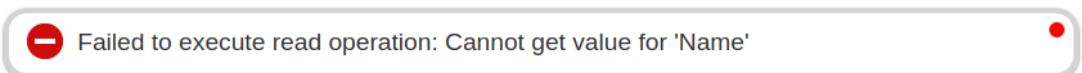
To execute an OPC UA query you must provide 5 properties:

- The *Device URL* of the device you want to connect to. You can either choose an active device from the section Devices or enter the address of an OPC UA device. If your connection requires another port than the default port 4840 you must add the port to the connection string as shown in the example.
- The *Identifier* of the node in the address space of an OPC UA device. If you have selected a device as the *Device URL* you are also able to choose from the defined fields of that device.
- The *Operation* either *READ* or *WRITE* or *EXECUTE*.
- The *Namespace* index of a node in an OPC UA server.
- The *Type* of the identifier. It can be a numeric value or a string. This property will only be visible if no field was selected as the *Identifier*.

Run the Executor by clicking the Execute button.

6.1 Error messages

When executing an OPC UA query the device may respond with error messages. The most important ones will be explained in this section. For a complete list of error messages, please check the documentation that came with your OPC UA device or the OPC UA standard. OPC UA error messages will appear on the bottom of the screen.

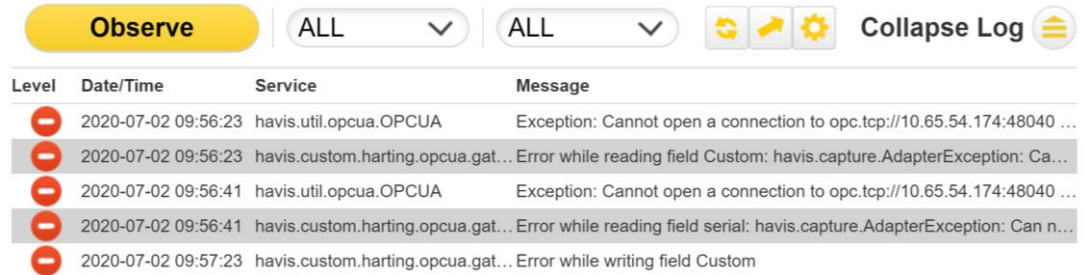


The first part is telling you what operation has failed, i.e. *Failed to execute read operation*. The second part tells you more information on what caused the problem. The following list shows the main mistakes that can lead to an error:

- Cannot get value for 'Identifier'
 - This message will appear if the OPC UA Gateway was not able to establish a connection to the given *Device URL*.
- Cannot get value for 'Identifier': INVALID PARAMETER
 - This message will appear if the OPC UA Gateway was not able to get a value for the given *Identifier*.
- Scheme seems invalid! The only supported scheme is 'tcp'
 - This message will appear if you have used an unsupported scheme in the *Device URL*.

7 Working with Logs

The OPC UA Gateway container provides full logging information. To see the log entries, click the *Expand Log* button.



The screenshot shows a log interface with the following elements:

- Observe** button (highlighted in yellow)
- Two dropdown menus set to **ALL**
- Control icons: refresh, back, and settings
- Collapse Log** button with a hamburger icon
- Log Table:**

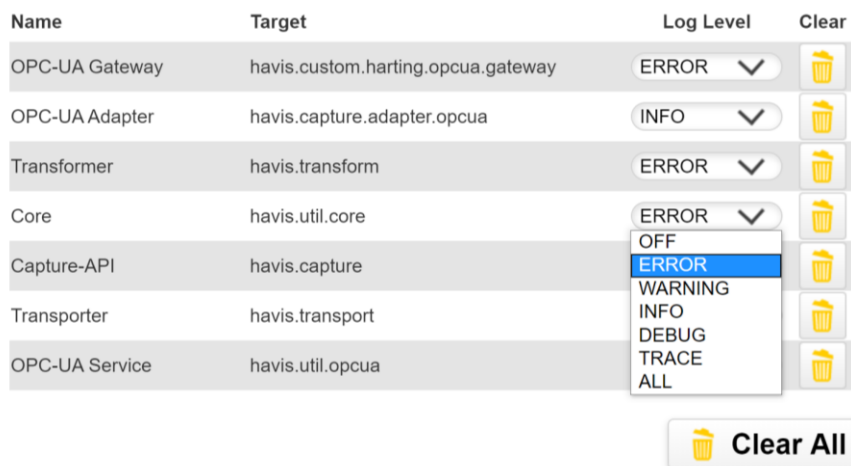
Level	Date/Time	Service	Message
	2020-07-02 09:56:23	havis.util.opcua.OPCUA	Exception: Cannot open a connection to opc.tcp://10.65.54.174:48040 ...
	2020-07-02 09:56:23	havis.custom.harting.opcua.gat...	Error while reading field Custom: havis.capture.AdapterException: Ca...
	2020-07-02 09:56:41	havis.util.opcua.OPCUA	Exception: Cannot open a connection to opc.tcp://10.65.54.174:48040 ...
	2020-07-02 09:56:41	havis.custom.harting.opcua.gat...	Error while reading field serial: havis.capture.AdapterException: Can...
	2020-07-02 09:57:23	havis.custom.harting.opcua.gat...	Error while writing field Custom

The following operations can be done in the Log part:

- Select the types of messages to be displayed
- Reload log information
- Export the log information

You can also adjust the log levels of individual components or clear their logs by clicking the cog icon in the *Log Configuration* dialog.

Log Configuration



The screenshot shows the Log Configuration dialog with the following table:

Name	Target	Log Level	Clear
OPC-UA Gateway	havis.custom.harting.opcua.gateway	ERROR	
OPC-UA Adapter	havis.capture.adapter.opcua	INFO	
Transformer	havis.transform	ERROR	
Core	havis.util.core	ERROR	
Capture-API	havis.capture	ERROR	
Transporter	havis.transport	ERROR	
OPC-UA Service	havis.util.opcua	ERROR	

A dropdown menu is open for the 'Core' component, showing the following options: OFF, ERROR (highlighted), WARNING, INFO, DEBUG, TRACE, ALL.

At the bottom right, there is a **Clear All** button with a trash icon.

8 OPC UA Gateway REST Interface

The OPC UA Gateway App provides a REST interface which let you collect information from the gateway and its connected devices without using the OPC UA protocol.

For an automatic access to the gateway it is essential to provide authorization data normally included in the request headers of the commands. In case of a newly started browser, you will be requested to enter your credentials prior to the first execution of the REST command.

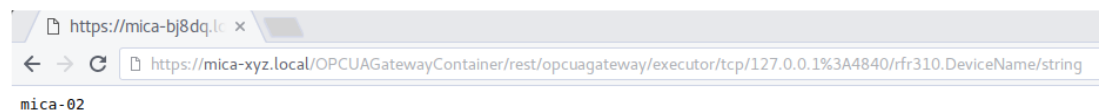
If you call the interface from a container within the same device, you can use the non secured http-protocol to save access time.

8.1 Quick REST Test Using a Browser

By means of a simple GET command you can retrieve information from the device over http. Navigate to the following URL in a browser of your choice:

```
https://[MICA_NAME].local/[CONTAINER_NAME]/rest/opcuagateway/executor/  
tcp/127.0.0.1%3A4840/rfr310.DeviceName/4/string
```

You will see something like the following



In the address contains the REST path followed by the same information you must specify in the Executor (see chapter 6).

The device address of the MICA base https://mica-xyz.local/ is followed by:

- The path to the executor section of the gateway
/[CONTAINER NAME]/rest/opcuagateway/executor/,
- The connection type tcp/
- The IP address and port 127.0.0.1%3A4840/
- The field identifier rfr310.DeviceName/
- The namespace index 4
- The identifier type string,

The operation mode (READ) is automatically selected by using the GET command.

8.2 Using REST from Inside a Container

You can also use the REST interface between two containers on the same MICA, for example using GNU WGET.

This sample script takes one parameter for the IP address, stores the result in a variable and prints it on the console.

```
#!/bin/sh  
IP=$1  
RESTPATH="rest/opcuagateway/executor/tcp/10.65.54.130/rfr310.DeviceName/  
e/4/string"  
printf "Call: %s\n" $IP  
RESULT=`wget -qO- http://admin:admin@$IP/$RESTPATH`  
printf "Result: %s\n" $RESULT
```