Ewon Flexy ThingWorx Connector
Version 1.1.0
User Guide

The Ewon Flexy Connector is a software reference project from HMS Networks.

Software Change Log

Version	Release Date	Changes
1.0	12/1/2020	Initial Release
1.1	TBD	Resolved the following issues: 1. Bug 1 2. Bug 2 Updates: 1. Update 1 2. Update 2

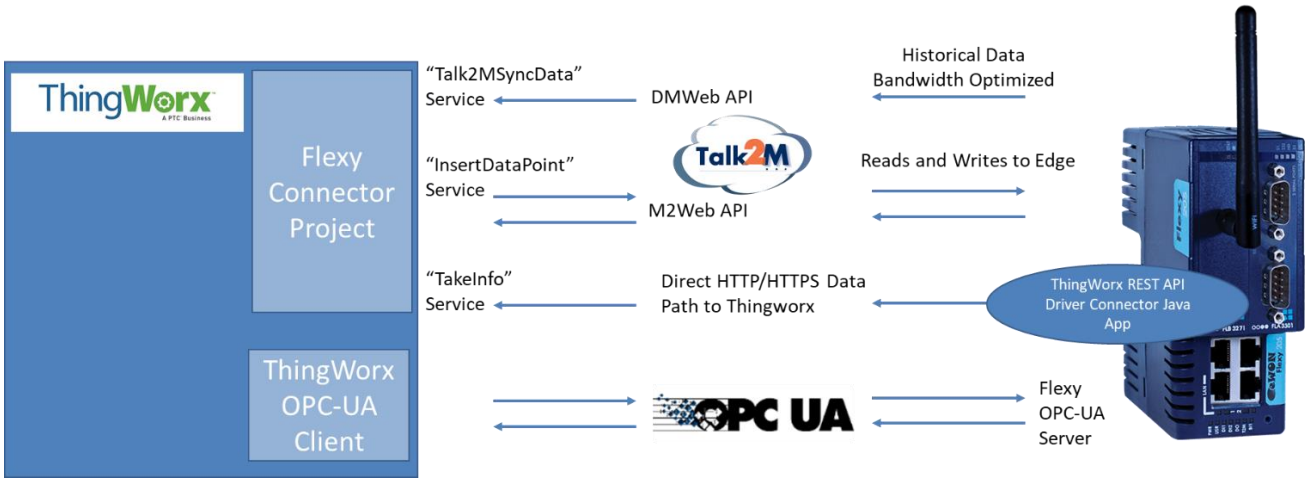
Introduction and Installation

Extensibility is a core aspect of the architecture and design of ThingWorx. Partners, third parties, and general ThingWorx users can easily add new functionality into the system, seamlessly. Extensions can be in the form of Service (function/method) Libraries, Connector Templates, Widgets, and more.
This document provides installation and usage instructions for the Ewon Flexy Connector Extension.

About the Flexy Gateway

HMS Networks is the OEM of the Ewon Flexy gateway which is a combined edge telemetry data solution and a remote VPN termination solution. The device natively connects to HMS Networks Talk2M cloud. The Talk2M cloud in combination with PC based software applications provided by HMS supports remote VPN access out of the box. Additionally, data telemetry between the Flexy gateway Talk2M is an out of the box feature. The edge gateway is configured via a web server running on the edge gateway. This configuration server can be accessed directly via the devise Ethernet ports or remotely via the Talk2M VPN connection.

The Flexy device and Talk2M support integration with third party systems. This extention is designed to enable data telemetry to ThingWorx. The solution supports three data paths. Edge to ThingWorx via Talk2M, Edge to Thingworx via the ThingWorx RestAPI, ThingWorx to Edge via Talk2m. Supported interfaces with ThingWorx are summarized in the image below.



About the EXTENSION

Ewon Flexy Connector allows you to stream telemetry data into ThingWorx directly from Ewon Flexy gateways or via the Ewon Talk2M cloud. These two historical data logging, edge to ThingWorx, data paths cannot be used simultaneously. The solution also enables communication from ThingWorx to Flexy edge gateways via the Ewon Talk2M cloud. This data path can be used simultaneously with a historical data logging data path.

Talk2M is the HMS edge device management portal. A single Talk2M account can have hundreds of associated Flexy gateways. The platform allows for device management and device configuration as well as bidirectional data telemetry via two RESTful API interfaces. M2WEB API is used for cloud to edge messaging and DMWEB API is used for edge to cloud messaging. Services included with the Ewon Flexy Connector interact with these two RESTful APIs

The direct data path allows data to be streamed directly from the Ewon Flexy to an instance of ThingWorx. In this case the Flexy device is sending payloads of historical data to the ThingWorx RESTful API interface support both HTTP and HTTPS. Those payloads are parsed in the Ewon Flexy Connector.

The Ewon Flexy Connector will create a unique thing for each Ewon Flexy. The tag data associated with the Ewon Flexy will be stored in identically named thing properties as a value stream.

The Ewon Flexy Connector Extension for ThingWorx provides customers with an easy path to capture data from remote or local PLCs and stream that data to ThingWorx. This extension is not dependent on any other extensions. HMS Networks host the Ewon Flexy ThingWorx connector as open source project on Github. Any issues or improvement requests should be logged on the Github repository. Please refer Ewon Flexy Connector Github page for additional information (https://github.com/hms-networks/flexy-thingworx-connector).

The Ewon Flexy Connector Extension requires one create a ConnectorHost "timer" thing in ThingWorx. This Thing initiates the data retrieval process and creates the EwonDevice thing for each Flexy. The ConnectorHost houses a majority of the configuration information for the Ewon Flexy Connector instance and provides services, including:

1. *InsertDataPoint*: Used by the Talk2M and direct data paths to insert a datapoint to the respective thing and thing property.

2. *MainExecution*: Used by the Talk2M data path for calling the Talk2MSyncData service within the script timeout period. This service is invoked by the ConnectorHost timer on its configured interval.

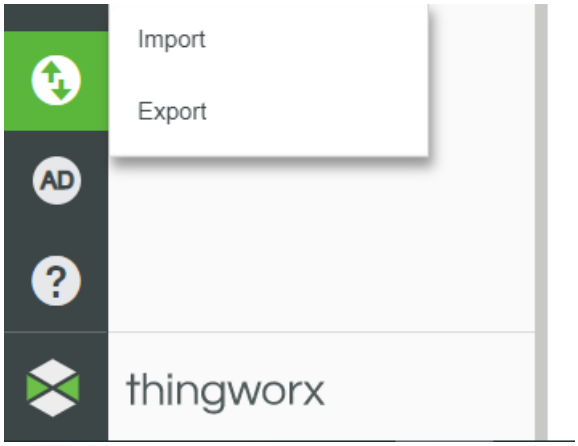
3. *ProcessTimeSinceUpdate*: Used by the Talk2M and direct data path services to track the time since the last received data update. This service is invoked by the MainExecution and TakeInfo services.

4. *TakeInfo*: Used by the direct data path to ingest telemetry messages from the Flexy Java Application component. This service is invoked by the Flexy Java Application component using the Thingworx REST API.

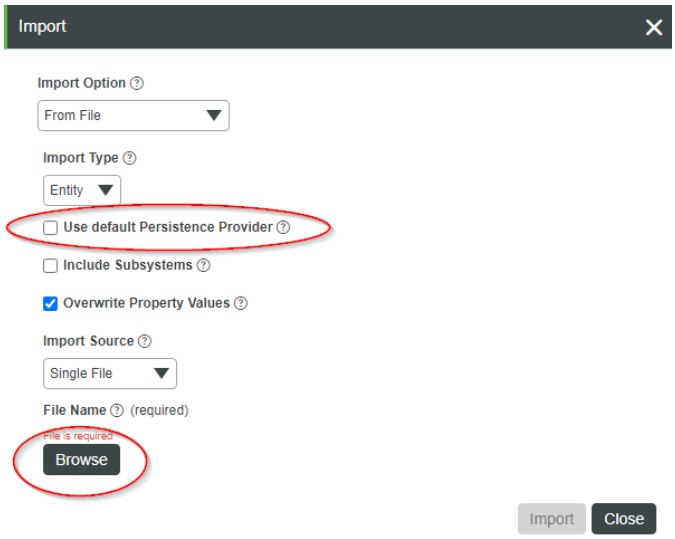
5. *Talk2MSyncData*: Used by the Talk2M data path for downloading a transaction of data points from Talk2M/DataMailbox. It requests only data it has not previously recieved using the stored value of lastTransactionId.

Installing the EXTENSION

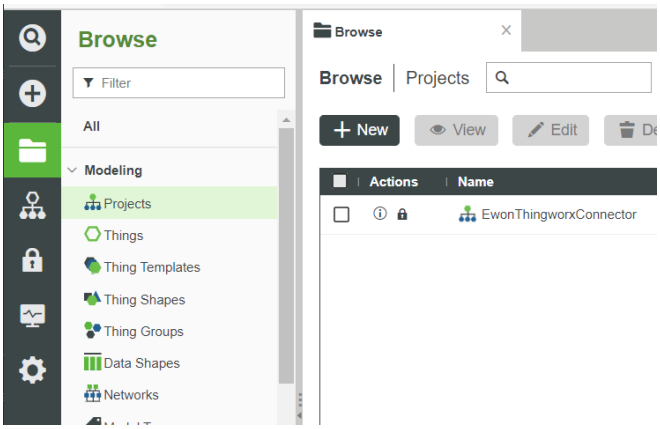
1. From a web browser, launch ThingWorx.
2. Log into ThingWorx as an administrator.
3. Go to **Import/Export > Import**.



4. Configure the Import as follows
    a. Input Option: From File
    b. Import Type: Entity
    c. Select Use Default Persistence
    d. Do not select Include Subsystems
    e. Select Overwrite Property Values
    f. Import source: Single File



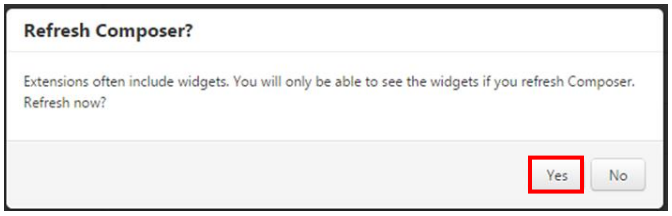
5. Click Choose File and select Entities.xml from the files included with the Ewon Flexy Connector



6. Click **Import**.

    Note: If an **Import Successful** message does not display, contact your ThingWorx System Administrator.

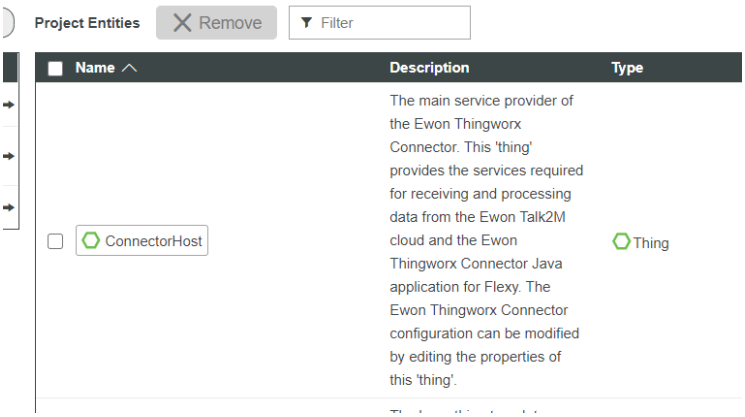
7. Click **Yes** to refresh Composer after importing the final extension.



8. Confirm that the Extension has been imported properly. Check the

Application Log for potential problems.

9. Open the ConnectorHost timer thing by clicking on it under the list of available entities, then selecting "Edit". Proceed to configuration and usage to complete the configuration process.



Configuration and Usage

Usage of the Ewon Flexy Connector Extension requires configuration of a ConnectorHost "timer" thing in ThingWorx. This thing stores a majority of the Talk2M and device access credentials and defines the rate at which data is requested from Talk2M. Testing the Ewon Flexy Connector Extension requires at least one Ewon Flexy Edge Gateway, a Talk2M Account and an Ewon Developer ID. Flexy gateways are available at a number of distributors worldwide for more info refer to this link https://www.ewon.biz/contact/find-distributor

Refer to this getting started guide to setup a Talk2M account: Ewon Quick Start Guide

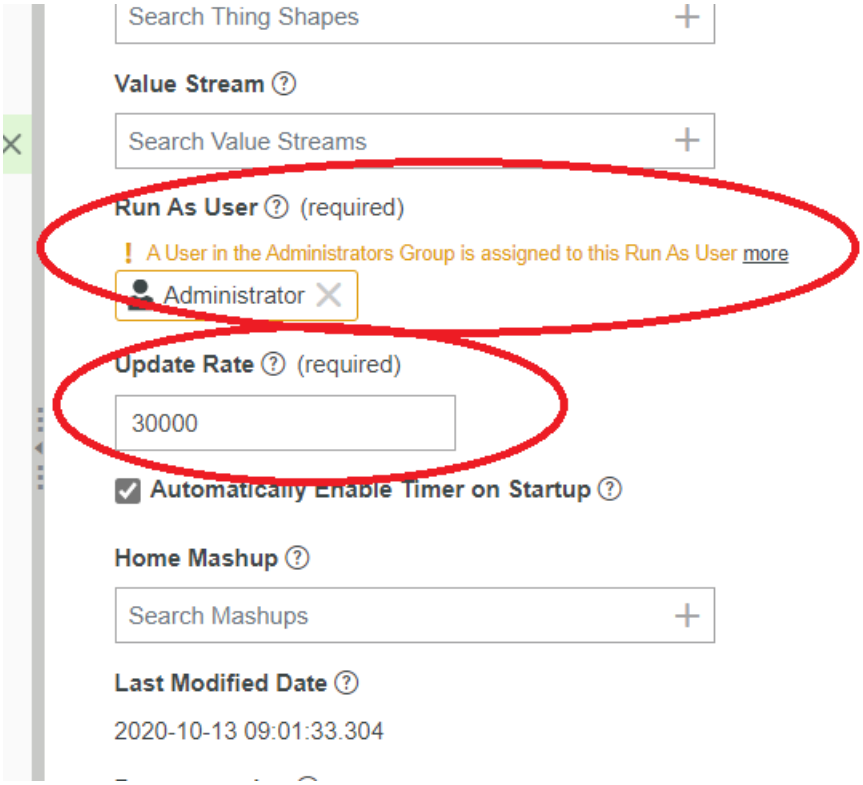
A developer id can be requested here https://developer.ewon.biz/registration

ConnectorHost Configuration

Once the ConnectorHost thing is created, it must be configured to connect to Talk2M instance. The ConnectorHost manages the interface between ThingWorx and Talk2m. This Thing is also used to disable historical logging via Talk2M and enable the direct edge to ThingWorx historical logging data path. The solution is designed to support a single Talkl2M account.

- Within the ThingWorx Composer, open up the ConnectorHost Thing
- On the General Information page, configure as follows
    - Configure "Run As User" to be something other than Administrator.
        - The selected user must posses the permissions necessary for creating and modifying things, and creating and updating thing properties.
    - Configure "Update Rate" to the interval at which the connector should run (in milliseconds).

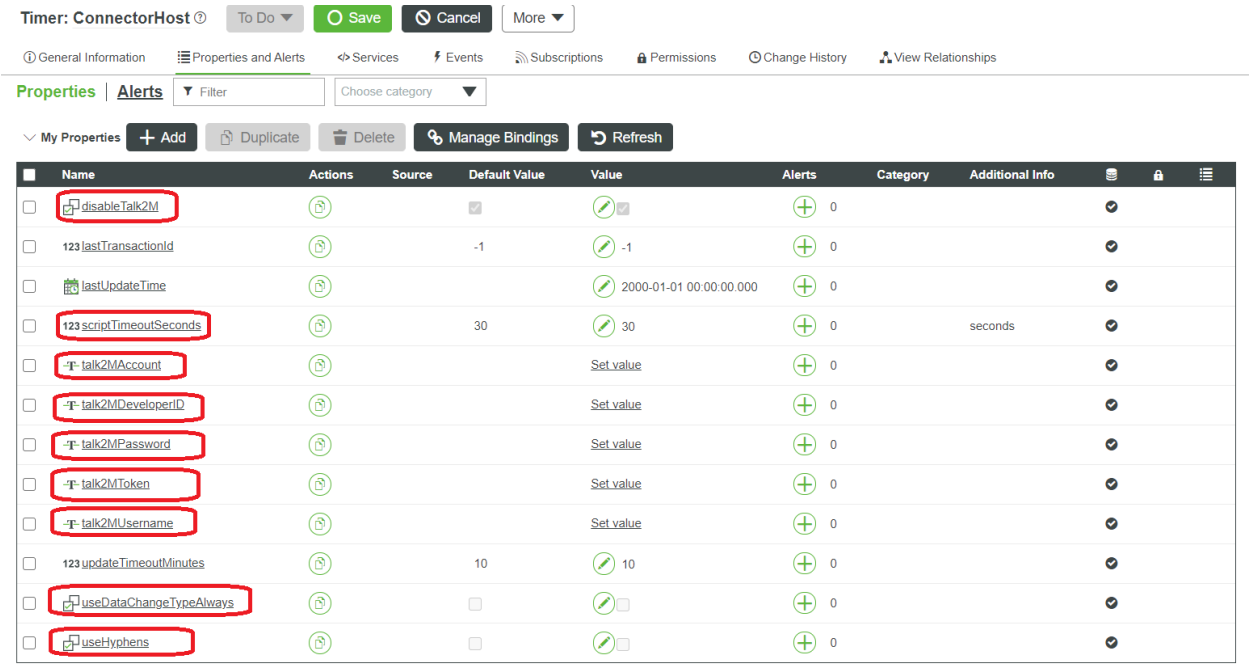
- This value is the interval at which data is downloaded from Talk2M/DataMailbox. If you are not using Talk2M functionality, this value does not need to be changed.



- On the properties and Alerts tab, configure as follows:
  - *disableTalk2M*: When set to true, this disables the Talk2M/DataMailbox functionality. For direct data path users, this option may be set to `true`, but for Talk2M/DataMailbox data path users, this option must be set to `false`.
  - *scriptTimeoutSeconds*: This should match the scriptTimeoutSeconds value in the Thingworx platform-settings.json file. If you have not modified it, the default value is 30.
  - *talk2MAccount*: The account name for login to Talk2M.
  - *talk2MDeveloperID*: A valid Talk2M developer ID.

- *talk2MPassword*: The account password for login to Talk2M.
- *talk2MToken*: An account API token for access to Talk2M.
- *talk2MUsername*: The account username for login to Talk2M.
- *useDataChangeTypeAlways*: When set to true, the DataChangeType field of newly created properties will be set to 'Always'. The default value is `false`, which will result in newly created properties using the DataChangeType of 'Value'.
- *useHyphens*: When set to true, additional organization is performed on thing properties with a prefix in the name (i.e. PREFIX-[tagname]).

Additional and more detailed information about the properties of ConnectorHost can be found by navigating to Thing Information > ConnectorHost > Properties. This view shows all properties and services associated with ConnectorHost.



GenericEwonDevice Configuration

The ConnectorHost thing will automatically leverage the GenericEwonDevice template to create EwonDevice things based on Ewon Names collected via Talk2M or the Direct connection depending on the option selected. The GenericEwonDevice template is the parent template child templates include GenericEwonDeviceTalk2M, GenericEwonDeviceDirect and GenericEwonDeviceValueStream.

If a thing of the same name already exists the connector will assume that thing was created previously and a new thing will not be created. Ewon Names must be unique across the entire ThingWorx project.

- Configuration Process Option 1: disableTalk2M == False

  Every Ewon device connected to Talk2M has a unique Ewon Name. The Ewon Name is defined when the Ewon Flexy gateway is provisioned. The Ewon Name is used to specify a single Ewon gateway when performing ThingWorx to Edge real-time reads/writes. The Ewon Name is used to create unique EwonDevice thing names.

  The Ewon Name is also used to associated telemetry data passing from the edge gateway to Talk2M and ultimately being read by the Ewon Flexy ThingWorx Connector via the Talk2MSyncData service. Received telemetry data is written to the correct EwonDevice thing properties based on the Ewon Name.

- Configuration Process Option 2: disableTalk2M == True

  Every Ewon device configured to push data directly to the ThingWorx RESTful API must have a unique thing name. The serial number of the device is used to define the thing name. The device name will be in the format *FLEXY-YYWW-SSSS-PP*, where YYWW-SSSS-PP is the Ewon Flexy's serial number. This is done when the Flexy device is enabled to push data directly to ThingWorx. Please reference the Ewon.Thingworx.Connector.Reference.Guide.pdf for information on how to configure a Flexy edge gateway to push data directly to ThingWorx.

  The Flexy Talk2M Ewon name and the thing name can be different when using the direct connection option. The user shall configure the EwonDeviceDirect property talk2MDeviceName to associate the EwonDevice thing with Talk2M. This is required to support ThingWorx to edge real-time reads and writes. Alternatively, the user can set the Ewon Name to FLEXY-YYWW-SSSS-PP when provisioning the gateway. This ensures the Talk2M Ewon Name reference and the EwonDevice thing name are identical.

- Properties
  Each Ewon gateway has a username and password which can be unique to that specific gateway. The default gateway username and password is adm/adm. The use needs to ensure the following properties match the associated Flexy gateway. When deploying large numbers of Flexy devices a defined user name and password strategy should be used decrease complexity. For example the device serial number could be the username. HMS does not recommend using the default password or the same username and password on each device.

  - *ewonDevicePassword*: The password of the referenced Ewon device. The default value is 'adm' and should be changed if your Ewon password is different. The value of this property is used by the

services in both the GenericEwonDeviceTalk2M and GenericEwonDeviceDirect thing templates.

- *ewonDeviceUsername*: The username of the referenced Ewon device. The default value is 'adm' and should be changed if your Ewon username is different. The value of this property is used by the services in both the GenericEwonDeviceTalk2M and GenericEwonDeviceDirect thing templates.

GenericEwonDevice Child Properties and Services

- EwonDeviceTalk2M Services – this service is only used when the Talk2M edge to ThingWorx data path is enabled.
  - *SendEwonOffline*: Sends the referenced Ewon device offline when using a triggered connection, such as 3G or other cellular network. More information about triggered connections can be found at https://www.ewon.biz/e-learning/library/cosy-131/remote-connection#:~:text=Triggered%20Connection%3A%20wake%20up%20%26%20put,when%20the%20user%20needs%20it.
  - *UpdateEwonTagValue*: Updates the specified tag with the specified tag value on the referenced Ewon using the Talk2M M2Web API.
  - *WakeEwonDevice*: Wakes up the referenced Ewon device when using a triggered connection, such as 3G or other cellular network. More information about triggered connections can be found at https://www.ewon.biz/e-learning/library/cosy-131/remote-connection#:~:text=Triggered%20Connection%3A%20wake%20up%20%26%20put,when%20the%20user%20needs%20it.

- EwonDeviceDirect Properties/Services – this is only used when the direct to ThingWorx data path is enabled.
  - *talk2MDeviceName*: A string property that the GenericEwonDeviceDirect services use to identify the referenced Ewon on Talk2M. Devices connected using the direct data path may have a different name in Talk2M than what appears in Thingworx, thus it must be explicitly set for direct data path things.
  - *SendEwonOffline*: Sends the referenced Ewon device offline when using a triggered connection, such as 3G or other cellular network. More information about triggered connections can be found at https://www.ewon.biz/e-learning/library/cosy-131/remote-

connection#:~:text=Triggered%20Connection%3A%20wake%20up%20%26%20put,when%20the%20user%20needs%20it.

- *UpdateEwonTagValue*: Updates the specified tag with the specified tag value on the referenced Ewon using the Talk2M M2Web API.
- *WakeEwonDevice*: Wakes up the referenced Ewon device when using a triggered connection, such as 3G or other cellular network. More information about triggered connections can be found at https://www.ewon.biz/e-learning/library/cosy-131/remote-connection#:~:text=Triggered%20Connection%3A%20wake%20up%20%26%20put,when%20the%20user%20needs%20it.

Usage

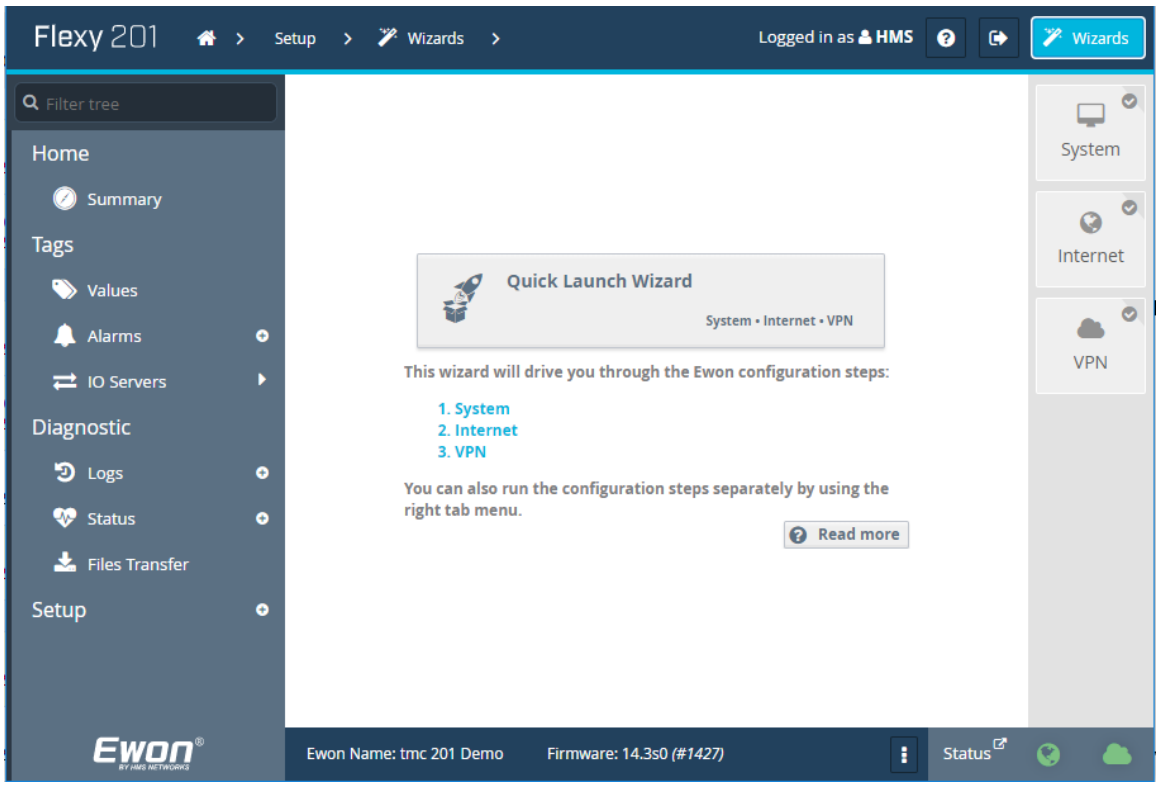
Once the system is configured each Ewon Flexy gateway will be represented by the EwonDevice thing. Each Ewon edge gateway will be configured with a set of tags defined when provision the edge gateway. Tags configured for historical logging (talk2M data path) will be streamed to ThingWorx automatically. When using the direct data-path a configuration file on the Flexy device is used to define which tags will be pushed to ThingWorx. Each tag will be written to the associated EwonDevice thing property. Property valuestreams are used to associate a timestamp with each tag written to a property. This creates a historical log of each property value. Mashup creation and use of the historical data stored in the thing is beyond the scope of this project.

The UpdateEwonValue service can be used to write a single Ewon Flexy tag. This service performs a write to a specific Ewon and specific tag. The service performs a real-time read back to verify the write was applied. The service is non-blocking. Mashup creation and use of the service is beyond the scope of this project.

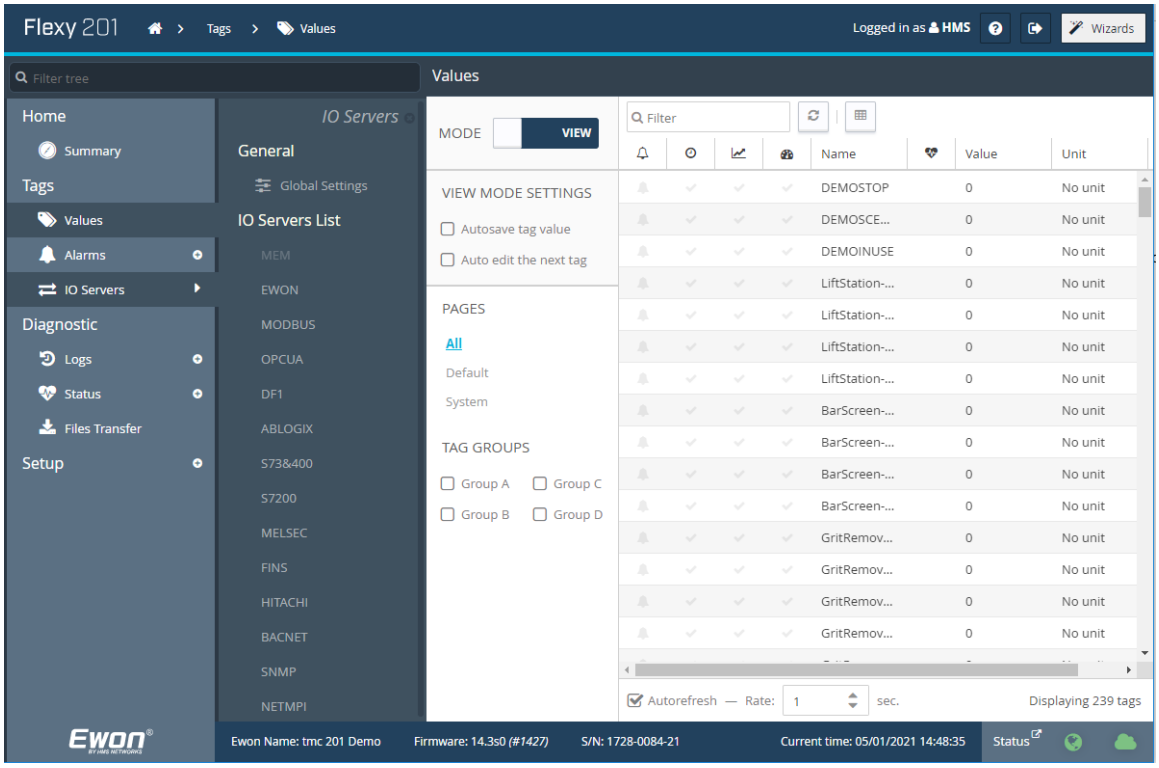
Flexy Configuration

The Flexy gateway is an application specific industrial edge gateway. The solution includes a full software stack and configuration GUI. The solution supports a Java sandbox which is used to add additional functionality to the solution. The device is not an industrial PC and does not support an open Linux interface for loading applications. The solution is designed enable connectivity with no software development.

When accessing the Flexy configuration web server the home page is as follows. The wizard configuration options on the left side of this page can be used to quickly configure the edge gateway and connect the device to Talk2M



The devices support data acquisition drivers for all PLC brands via both serial and Ethernet interfaces. The device supports data acquisition from Modbus RTU and Modbus TCP enabled sensors. The device also supports customer defined serial interfaces. Lastly the gateway supports direct inputs from both digital and analog based sensors. Data tags are configured on the Tags > Values page and I/O servers are accessed under the Tages > IO Server page.



Each tag is individually configured. The source I/O server is selected the data type, sample rate and other meta data are defined when configurating a tag. To enable historical logging of a data sample historical logging must be enabled for each tag and the logging rate must be defined.

There is no additional configuration at the Flexy required beyond the standard features to enable data telemetry via Talk2M to Thingworx.

Pushing data directly to the ThingWorx Rest API requires some additional configuration. In summary, HMS has created a custom Java App to support this interface. That Java application must be FTP'ed to the Flexy along with a script that launches the Java App. These files must be placed in the Flexy /usr directory. The files are **fley-thingworx-connector.jar** and **jvmrun**. Once these files have been FTP'ed to the Flexy it should be rebooted or power cycled. Upon reboot a ThingWorx default json configuration file will be created automatically on the Flexy in the /usr directory. This file will need to be modified to include information required to connect to a specific ThingWorx instance. Refer to the Ewon.Thingworx.Connector.Reference.Guide.pdf for additional information on the configuration process.

[Known Limitations and] Compatibility

This extension was tested for compatibility with the following ThingWorx Platform version(s) and Operating System(s):

ThingWorx Platform Version	ThingWorx 8.5.5-b103 and 9.0.0
OS	Windows Server 2019
Java	11.0.7 and 1.8.0_261
Tomcat	8.5.55 and 9.0.35
Postgres	12.3 and 12.4

Document Revision History

Revision Date	Version	Description of Change
December 1, 2020	1.0	Initial release
January 3, 2021	1.1	Added additional images and text to clarify usage