Preliminary

Connectors

Contents

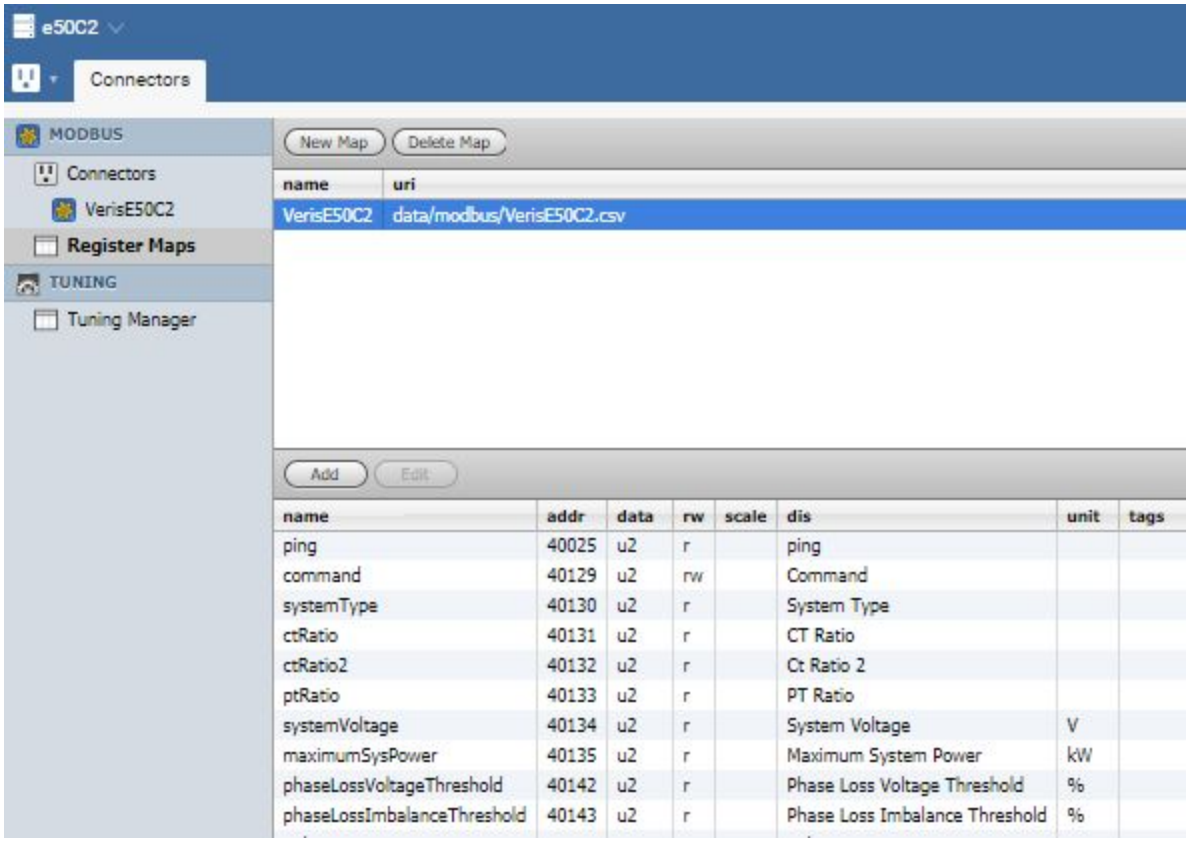
Introduction

This is an introduction to the Connectors app of Tipify™. We will cover how to access it through the graphical interface, and use it to debug connectors and connected points. We will not cover connector creation since that will vary by connector type. Reference information is also provided for further study.

Overview

The Connectors app provides the ability to manage your Connectors. You can perform many tasks with Builder, but the Connectors app provides additional diagnostic capabilities. The sidebar is grouped by connector type which contains an item called "Connectors" used to view and manage the type's connectors. You will also see an item for each connector configured which lets you view and manage the points.

If you are using Modbus, there is also a simple tool to manage your register map files.



Conn Remote Status

Previously connector point status did not take into account the remote system's status. We've enhanced the connector framework to handle this case. Now `curStatus` will report status problems in the remote system via the new strings:

- `remoteDown`
- `remoteFault`
- `remoteDisabled`
- `remoteUnknown`

The following connectors now support this feature: BACnet, Haystack, and oBIX.

Remember that if you are using local history collection, that values are only logged when curStatus is "ok". Previously the connector might have reported "ok" when in fact the remote point was in an error condition.

Conn Tuning

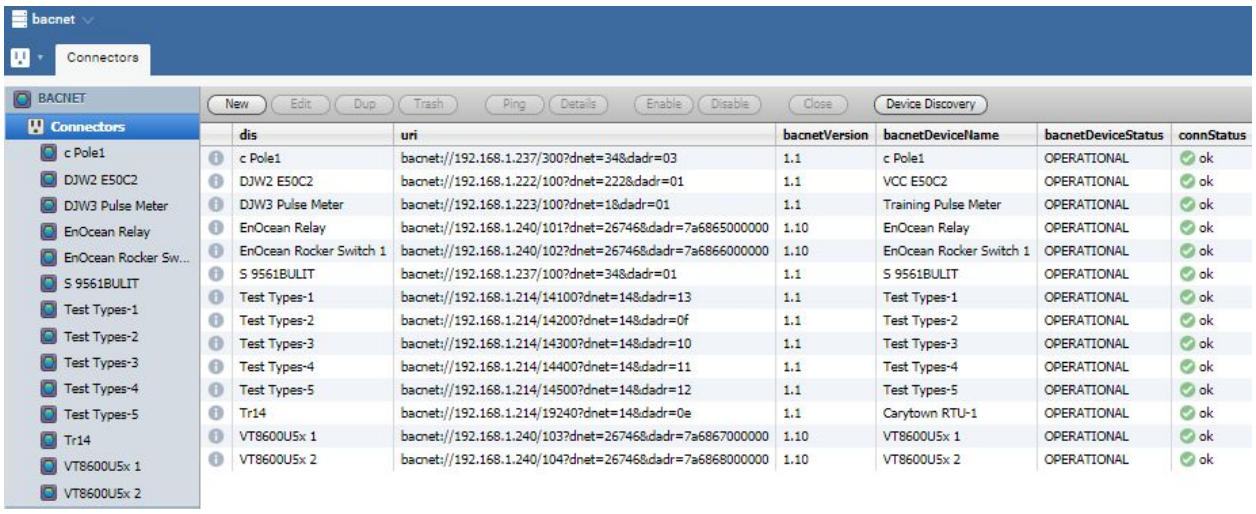
There is a new subsystem within the connector framework to support tuning of connector points. You can now create records with the `connTuning` tag. Points are assigned a tuning configuration via the`connTuningRef` tag applied to the point, connector, or ext itself.

Three new features are included in the first version of tuning:

- staleTime: transitions a point's status from "ok" to "stale" after a configurable period of time elapses after the last successful read. This lets applications and users know that the point data is not fresh.
- writeMinTime: throttles the frequency of writes to the remote device.
- writeMaxTime: configures periodic rewrites to the remote device.

Connectors

The Connectors under each type in the tree, allow you to see all the connectors of that type.



dis	uri	bacnetVersion	bacnetDeviceName	bacnetDeviceStatus	connStatus
c Pole1	bacnet://192.168.1.237/300?dnet=348&dadr=03	1.1	c Pole1	OPERATIONAL	ok
DJW2 E50C2	bacnet://192.168.1.222/100?dnet=228&dadr=01	1.1	VCC E50C2	OPERATIONAL	ok
DJW3 Pulse Meter	bacnet://192.168.1.223/100?dnet=18&dadr=01	1.1	Training Pulse Meter	OPERATIONAL	ok
EnOcean Relay	bacnet://192.168.1.240/101?dnet=267468&dadr=7a6865000000	1.10	EnOcean Relay	OPERATIONAL	ok
EnOcean Rocker Switch 1	bacnet://192.168.1.240/102?dnet=267468&dadr=7a6866000000	1.10	EnOcean Rocker Switch 1	OPERATIONAL	ok
S 9561BULIT	bacnet://192.168.1.237/100?dnet=348&dadr=01	1.1	S 9561BULIT	OPERATIONAL	ok
Test Types-1	bacnet://192.168.1.214/14100?dnet=148&dadr=13	1.1	Test Types-1	OPERATIONAL	ok
Test Types-2	bacnet://192.168.1.214/14200?dnet=148&dadr=0f	1.1	Test Types-2	OPERATIONAL	ok
Test Types-3	bacnet://192.168.1.214/14300?dnet=148&dadr=10	1.1	Test Types-3	OPERATIONAL	ok
Test Types-4	bacnet://192.168.1.214/14400?dnet=148&dadr=11	1.1	Test Types-4	OPERATIONAL	ok
Test Types-5	bacnet://192.168.1.214/14500?dnet=148&dadr=12	1.1	Test Types-5	OPERATIONAL	ok
Tr14	bacnet://192.168.1.214/19240?dnet=148&dadr=0e	1.1	Carytown RTU-1	OPERATIONAL	ok
VT8600U5x 1	bacnet://192.168.1.240/103?dnet=267468&dadr=7a6867000000	1.10	VT8600U5x 1	OPERATIONAL	ok
VT8600U5x 2	bacnet://192.168.1.240/104?dnet=267468&dadr=7a6868000000	1.10	VT8600U5x 2	OPERATIONAL	ok

The columns provide information on each of the connectors. They will vary based on connector type. Some common properties include:

dis – Display name of Connector

connStatus – Connector Status - ConnStatus enumeration

- disabled
- down
- fault
- ok
- unknown

connState – Connector State - Current connection state of a connector as one of the predefined strings:

- closed: connection is closed
- closing: a connection is currently being closed
- open: connection is open
- opening: a connection is currently being opened See connExt

connErr – Connector Error - Error message associated when `connStatus` indicates an error condition. See connExt

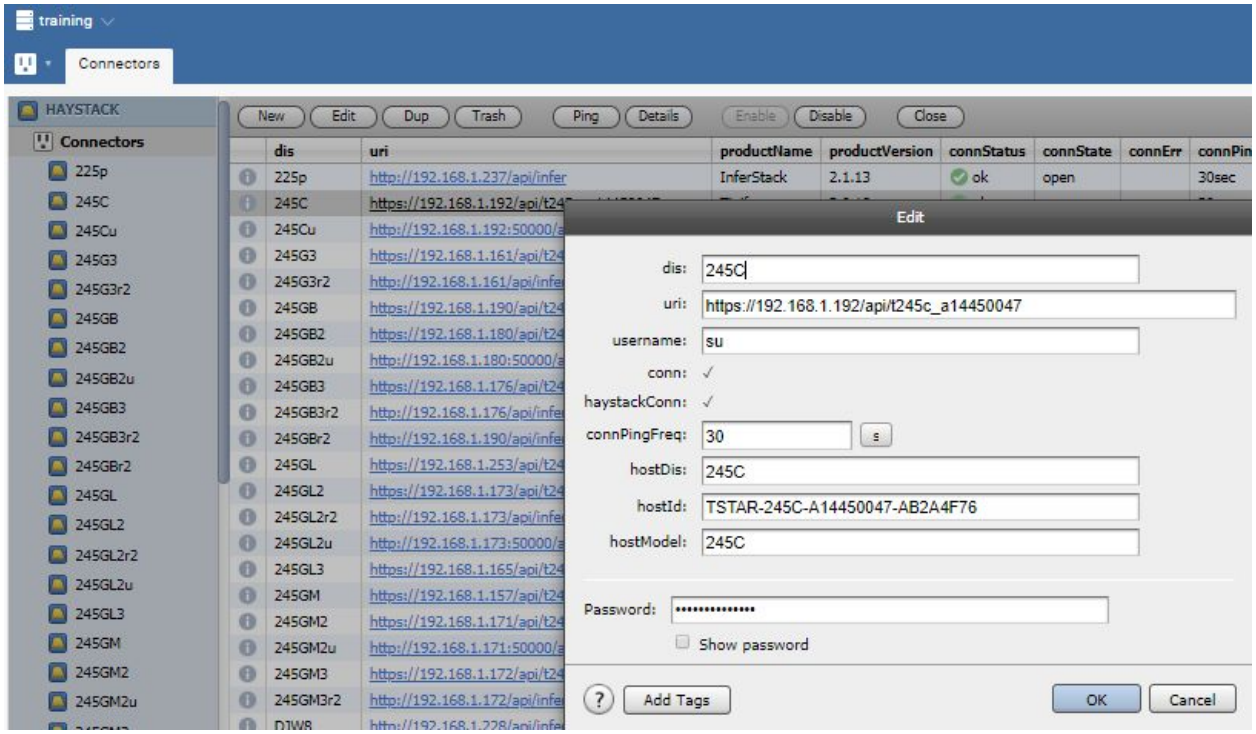
The buttons across the top allow you to manage your connectors.

New

This allows you to create a new connector of the type selected.

Edit

This allows you to edit the selected connector.



Dup

This allows you to duplicate the selected connector.

Trash

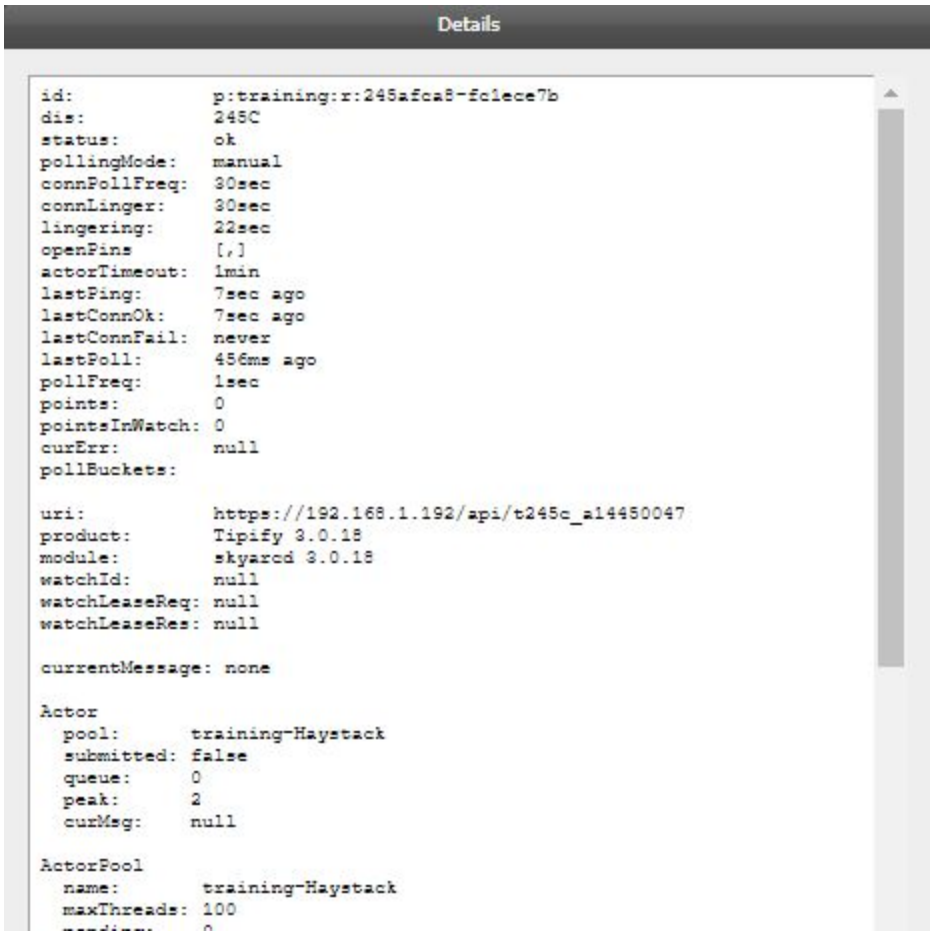
This allows you to delete the selected connector (move it to the Trash).

Ping

This allows you to ping the selected connector.

Details

This allows you to view the details of the selected connector.



Enable

This allows you to enable the selected connector.

Disable

This allows you to disable the selected connector.

Close

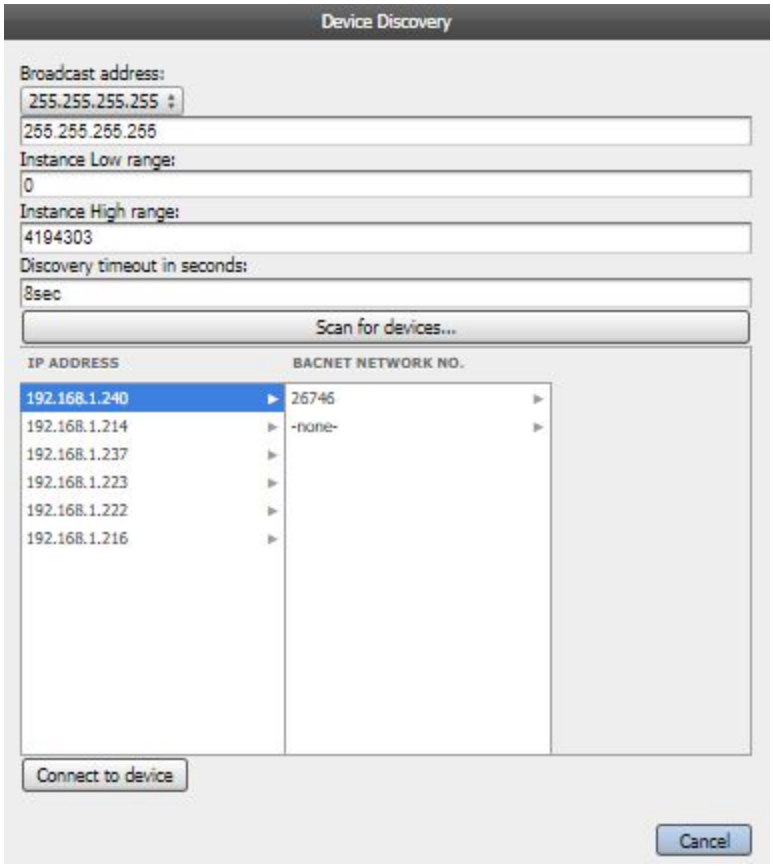
This allows you to close the selected connector.

Device Discovery (if supported)

This allows you to discover devices when supported by the selected connector (like BACnet or Haystack).

BACnet Discovery

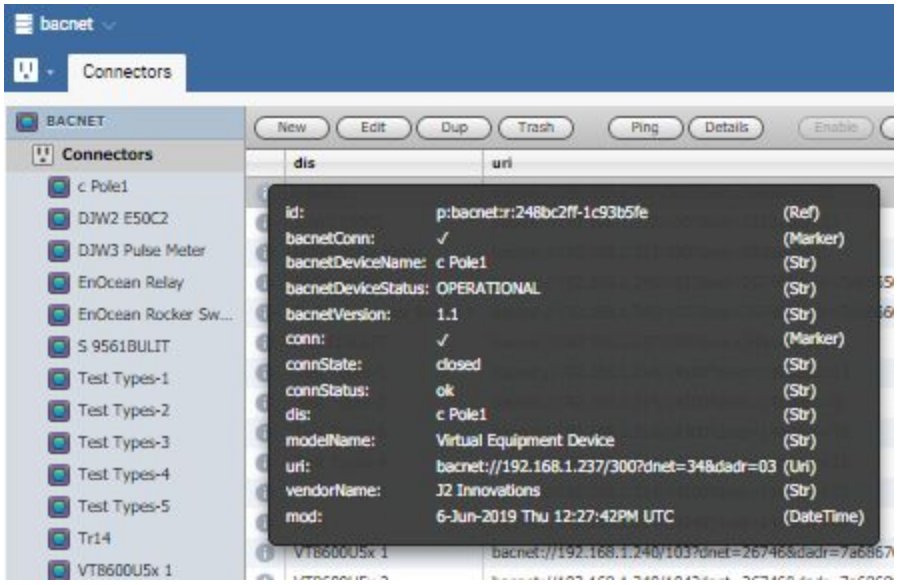
Select BACnet Connectors and press Device discovery. Press Scan for devices.



Select a device and press Connect to device. You will get a response like:

Record Information

Record Information displays all the properties of the connector. Select a connector and press the logo.



Points

Pressing a connector allows you to review Points associated with a connector.

The columns provide information on each of the points. They will vary based on connector type. Some common properties include:

dis – Display name of Point

cur – Marker tag which indicates the point has a real-time, current value accessible via the `curVal`tag. Typically this tag means that the point is mapped to a connector and when subscribed by a watch will ensure that the latest value is synchronized from the remote system by the connector. But it may also be used for real-time values generated by internal application such a control program. See pointExt

write – This displays these properties if writeable:

writable `Marker` Annotates a point as a writable output with 16-level priority array. See pointExt

writeErr `Str` Error message associated when `writeStatus` indicates an error condition. See pointExt

writeLevel `Number` Current priority level for `writeVal` as number between 1 and 17. The value 17 indicates the relinquish default value. See pointExt

writeStatus `Str` Current status of a writable point output:

unbound: writable point is not bound to a connector

ok: last write was successful

down: connectivity/networking problem - see `writeErr`

fault: configuration error - see `writeErr`

disabled: manual disable of the point or parent connector

unknown: we don't know anything (typically boot state) See pointExt

writeVal `Obj` Current desired value to write to output. See pointExt

his – This displays these properties if history enabled:

hisCollectCov `Obj` Enables COV history collection of the `curVal`. The tag value must be Marker to collection on any change. Or if the point is Numeric, then the value may be a Number which indicates the tolerance a point must change before logged. See pointExt.

hisCollectInterval `Obj` Enables interval history collection of the `curVal`. The value must be a duration Number with a "min" or "hr" unit. If minutely then the value must be evenly divisible into a 60min hour. If interval is hourly then the value must be evenly divisible into a 24hr day. SeepointExt.

hisErr `Str` Error message associated when `hisStatus` indicates an error condition. See pointExt

hisStatus `Str` Current status of a point history sync as one of the predefined strings:

ok: last sync attempt was successful

fault: a configuration problem - see `hisErr`

down: a communication or network problem - see `hisErr`

disabled: manual disable of the point or parent connector

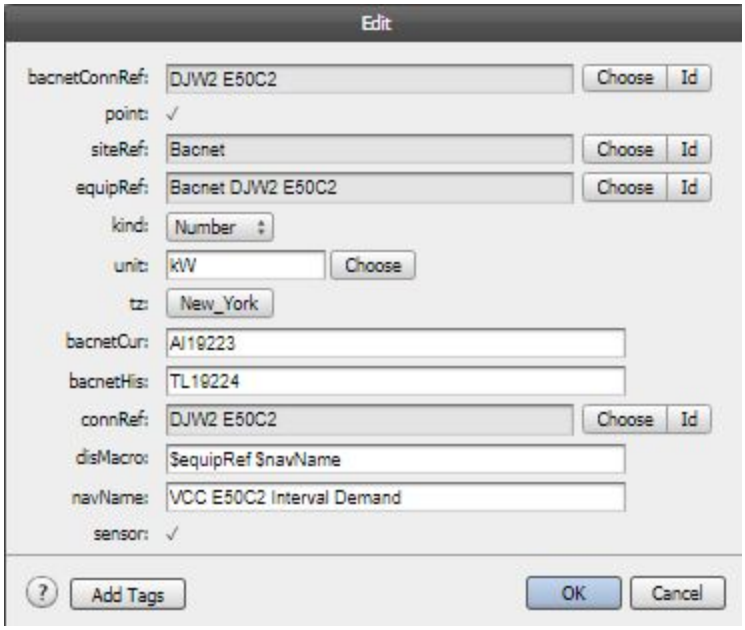
unknown: we don't know anything (usually boot state)

pending: sync has been scheduled and is waiting to running

syncing: sync is currently running See pointExt

Edit

This allows you to edit the selected point.

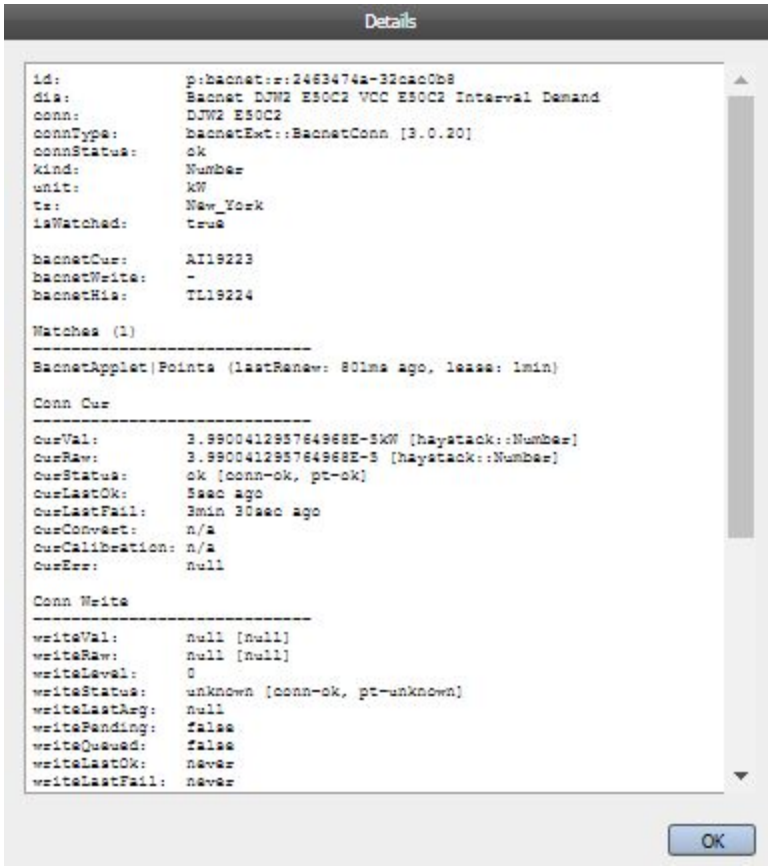


Trash

This allows you to delete the selected point (move it to the Trash).

Details

This allows you to view the details of the selected point including Watches, Conn Cur, Conn Write, Conn Tuning and His Collect.



Watches

This displays the watches for the point.

Conn Cur

This displays the current values for the point.

Conn Write

This shows write values for the point.

Conn His

This displays history collection information for the point.

Conn Tuning

This displays polling for the point.

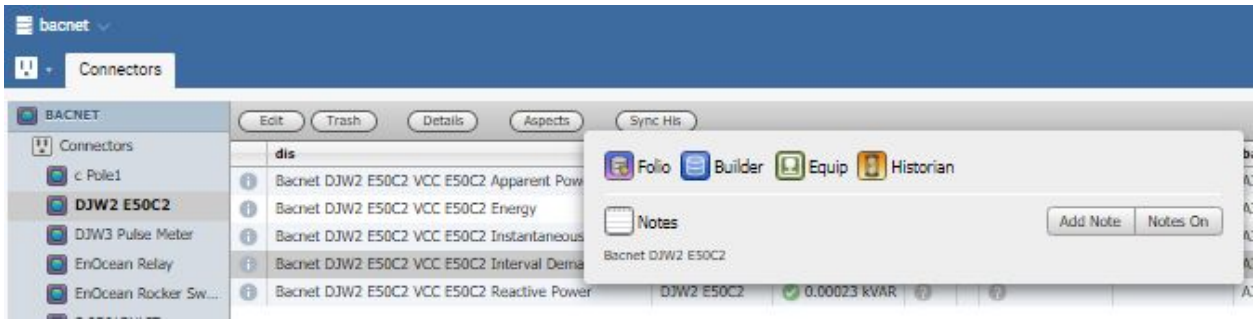
Writable

This displays the priority array for the point.

```
Writable
============================
Level    Val           Who
-----    ----------    --------
1        null
2        null
3        null
4        null
5        null
6        null
7        null
8        null
9        null
10       null
11       null
12       null
13       null
14       true           Control Task: oscillate @ VCC WEMULTI-IO
15       null
16       null
17       null
```

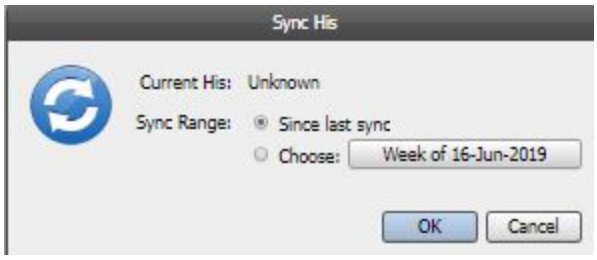
Aspects

Aspects allows you to access other apps for the selected point or add notes to the equipment.



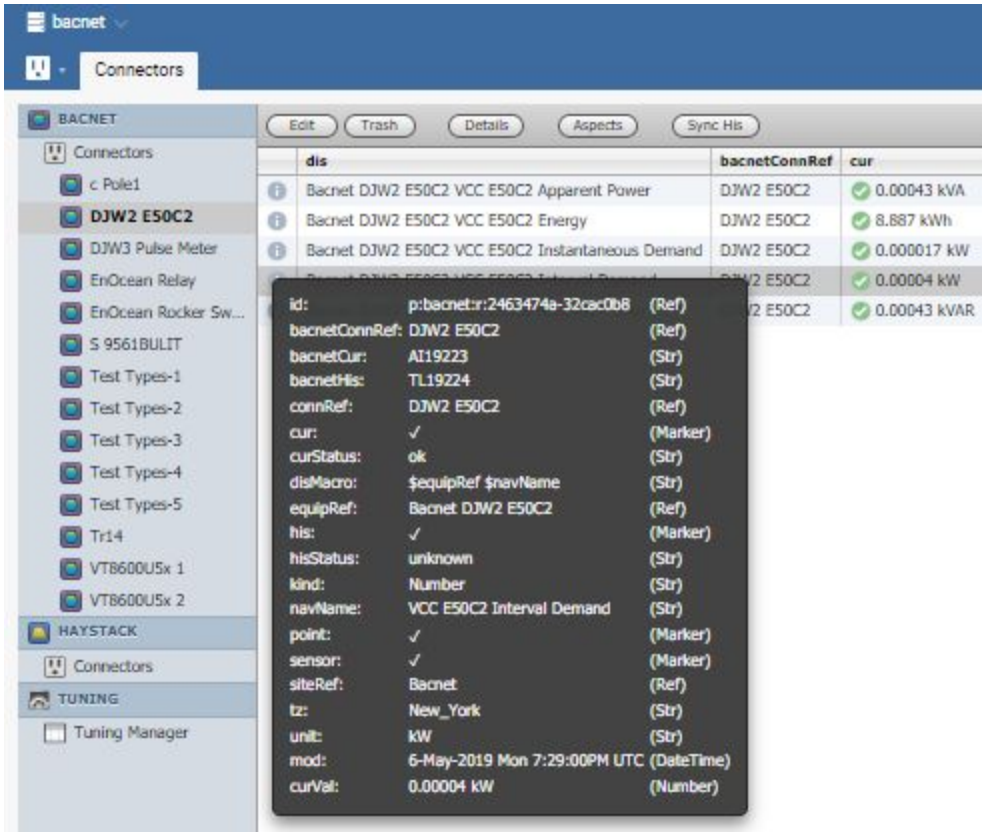
Sync His

This allows you to synchronize the history if possible.



Record Information

Record Information displays all the properties of the point. Select a point and press the logo.



Actions

Actions display all the actions that can be performed on the point. Select a point and press the logo. These include the appropriate commands for the point like these

Emergency Set

Emergency Auto

Manual Set

Manual Auto

Set Default

Set Null

Manual On

Manual Off

Manual Auto

Emergency Active

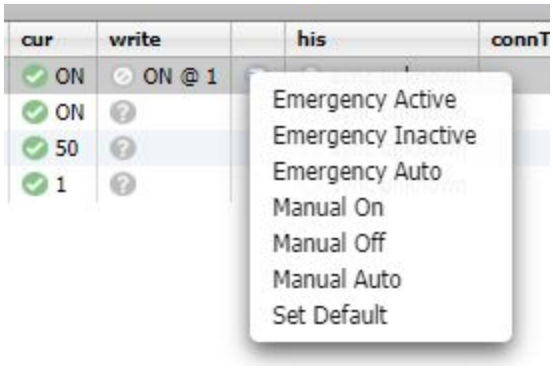
Emergency Inactive

Emergency Auto

Manual Active

Manual Inactive

Manual Auto

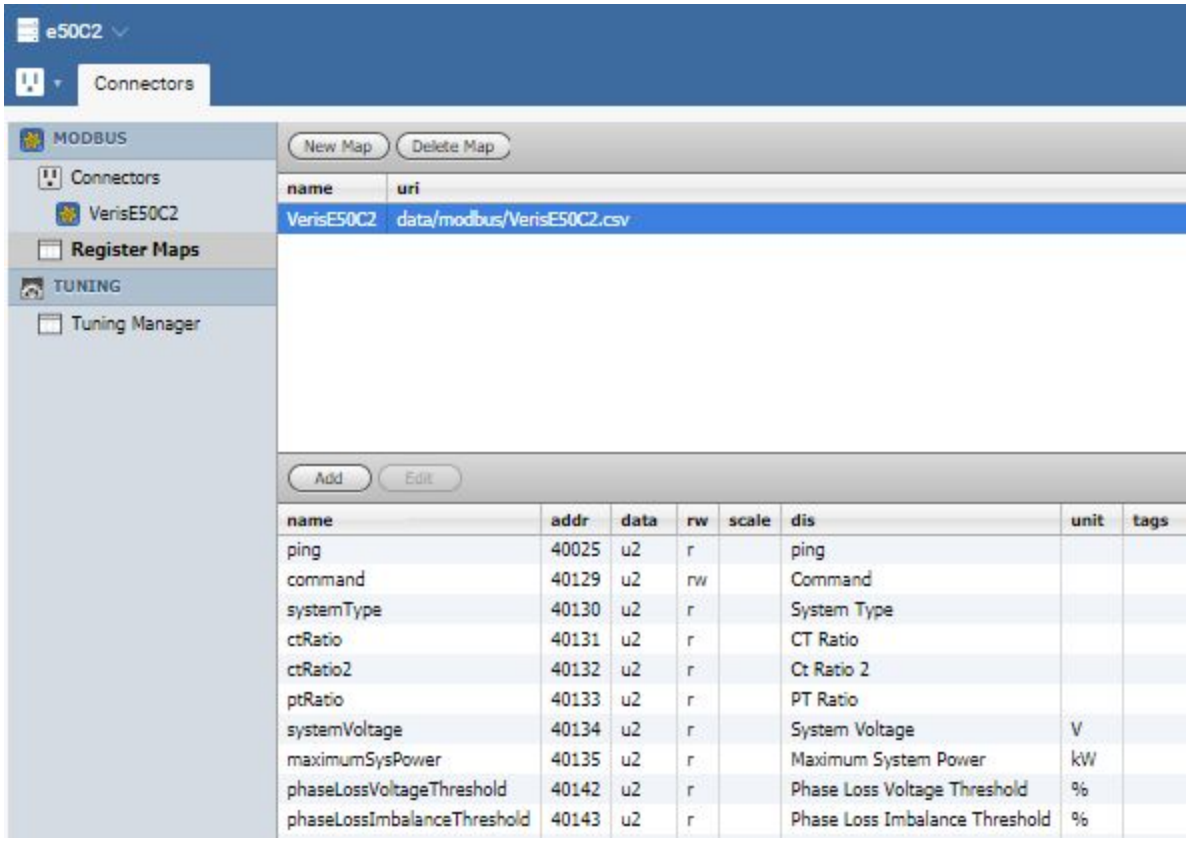


Make Points Commandable

In order to add commandable points, add cmd, writable and actions string tags to point.

Modbus Register Maps

If you are using Modbus, there is also a simple new tool to manage your register map files. Register Maps provides access to your Modbus register maps. You can add new ones or edit existing ones.



Register Maps allow you to define the registers in your Modbus device for easy interaction as points. The columns include: name, addr, data, rw, scale, dis, unit and tags. See https://skyfoundry.com/doc/ext-modbus/doc#regMap for more information.

New Map

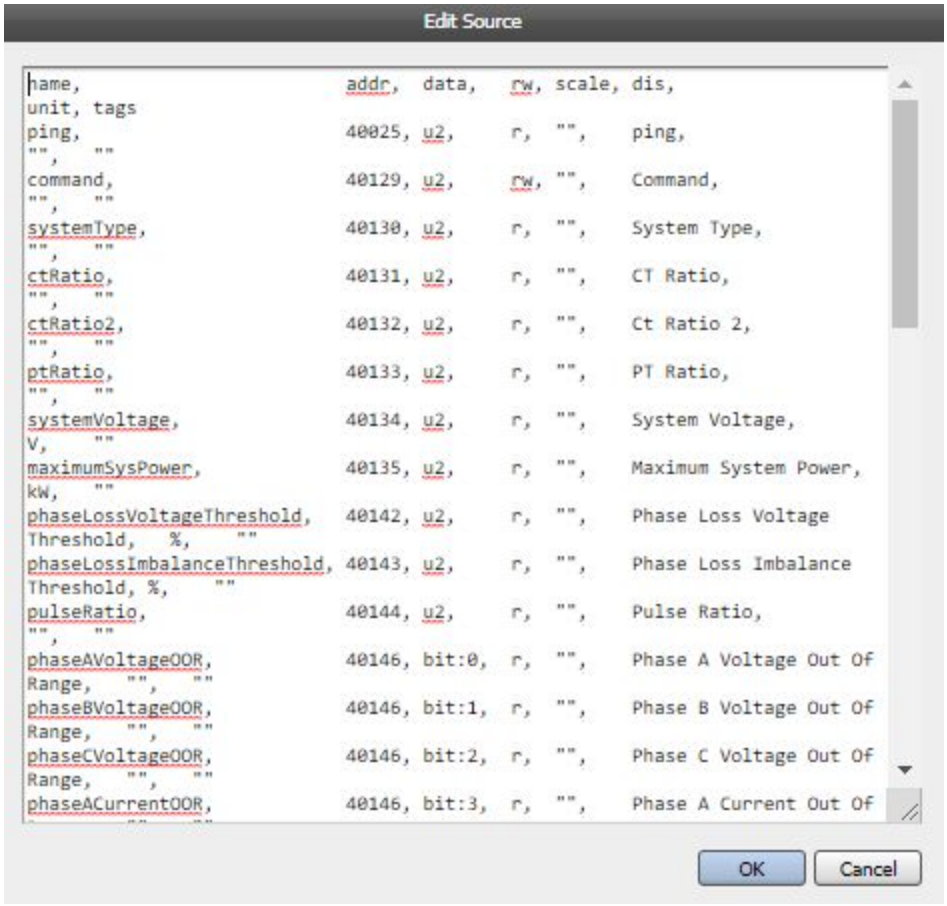
Press New Map to create a new Register Map.

Name it and press OK.

Add or Edit Source

You can Add records one at a time or use Edit Source to paste a template containing these columns:

```
name, addr, data, rw, scale, dis, unit, tags
```



You can also select an existing entry in a register map and Add or Edit it.

Name

The name of the register must start with a small letter.

Addr

This is the register address.

Data

This is the data type.

Rw

This is whether the register can be written or is read only.

Scale

This allows a scale factor to be applied efficiently to the value of the register.

Dis
This is the display name for the point

Unit
All the current units are listed here (which is included in your setup under /etc/sys/units.txt):

https://bitbucket.org/fantom/fan-1.0/src/tip/etc/sys/units.txt?at=default

Tags
This allows you to add tags to your registers.

Provisioning
The Provisioning provides tools for centrally managing groups of InferStack™ devices. This tool provides support for:

- Monitoring connection status
- Batch upgrading devices
- Batch patching devices

Once you enable the Provisioning extension in Settings, you will see Provisioning under Haystack. See Provisioning for more information.

Tuning Manager
The Tuning Manager provides the ability to create, edit and delete connTuning records to tune connector points. Points are assigned to a tuning configuration via the `connTuningRef` tag. This tag is searched in the following order:

1. Point rec itself
2. Point's connector rec
3. Connector's `ext` rec
4. Fallback to default tuning

Note: all the tuning timers rely on periodic background processing which is only accurate to within 5 to 10sec.

pollTime
The `pollTime` tag specifies a duration Number which is the frequency used to poll a point for `curVal`. This tag is only used for connectors which support polling for their current value. Connectors which use a COV subscription model will ignore this value. If unspecified the default is 10sec.

You can then add connTuningRef where needed as in points shown here.

See https://skyfoundry.com/doc/ext-conn/doc#tuning for more information on tuning.

Monitoring Connectors

You can add connPingFreq to any connector to monitor it regardless of connected points.

connPingFreq is a Duration used to configure the auto-ping feature on a given connector. When this tag is configured the connector will automatically attempt a ping based on the configured frequency. For connectors which might not have watched points this ensures periodic checks of the connectivity status. If this tag is not defined on a connector then the feature is disabled.

For more information see https://skyfoundry.com/doc/ext-conn/tags#connPingFreq

Reference

There are a number of references available on your device or server. In the Doc app, scroll down to Extensions and select from these:

conn – Connector Extension Framework

equip – Equipment Extension

his – Historian Extension

point – Point Extension

The Standard Rules for Naming can be found in docInferStack™ Folio under naming (https://skyfoundry.com/doc/docSkySpark/Folio#naming).

Online searchable docs are available at https://skyfoundry.com/doc/.