Consultant Sensors Howto

www.controlplus.biz

2009-06-28 Version 1

Sensors

This document is a guide to creating Sensors in ConsultantX software from Control Plus.

www.controlplus.biz

You will learn :-

- How to create a Standard sensor.
- How to create a Raw Data sensor.
- How to create a Communication Fail sensor.
- How to create a Status sensor.
- How to create a Derived sensor.
- How to create a Pinging sensor.
- How to create a Ping Time sensor.
- How to create a Ping Average sensor.

Before you begin.

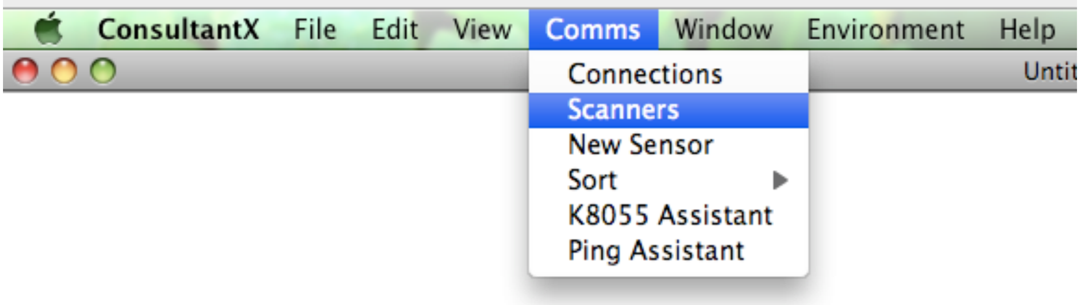
Before you begin creating sensors you must first create a connection to a device and then a scanner. Please refer to the how to documents for the particular type of device you are using.

What is a sensor.

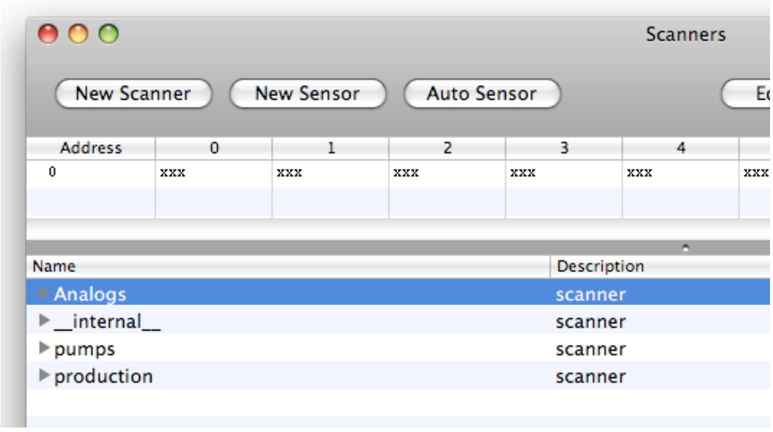
A scanner is an object that gathers blocks of data from a device such as a PLC. This block of data could be 100's of bytes in length. A sensor is an object that defines a single point within this block of data. This single point could be an analog value in a 32 bit floating point format or it could be a single bit within a single byte that indicates the on/off status of something.

Creating a new standard sensor.

Select *Scanners* from the *Comms* Menu.



Select the scanner that will own this sensor by clicking on its name in the list of scanners. In this example we have a scanner called *Analogs.*



Address	0	1	2	3	4	
0	xxx	xxx	xxx	xxx	xxx	xxx

Name	Description
▶ Analogs	scanner
▶ __internal__	scanner
▶ pumps	scanner
▶ production	scanner

Click the *New Sensor* button in the Scanners window. In this example we are going to create a standard sensor that accesses data element 1. Data element 1 is actually the 2nd element in the scanners data table. As with most computer applications the 1st element is element 0. This sensor is the amps from a conveyor motor and the expected range of the sensor is 0 to 500 amps. The Min and Max values are very important as the value this sensor displays or records will be clipped to this Min and Max values. For example if a sensor had a Max value of 100 entered but the data point in the PLC normally operated above 100 the sensor would only ever display 100.

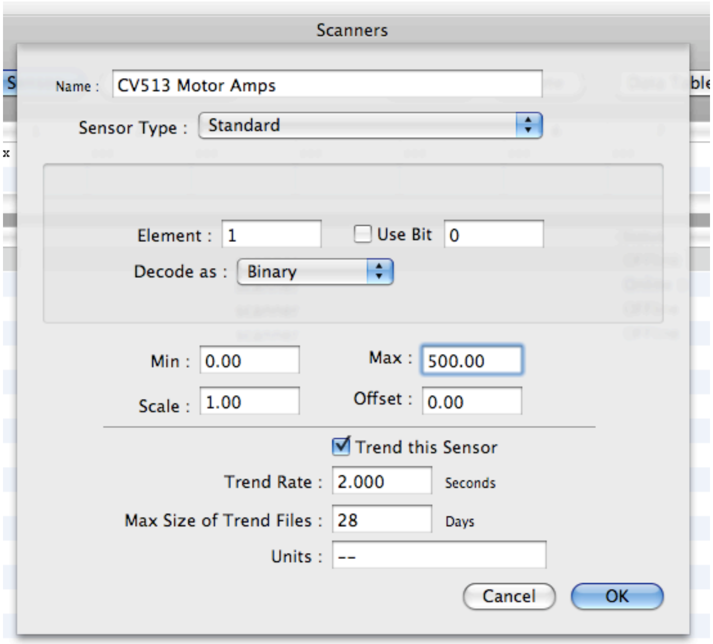
Scale and Offset

The output value of the sensor is calculated by applying the following formula.

Sensor output = ActualValue scale + offset.*

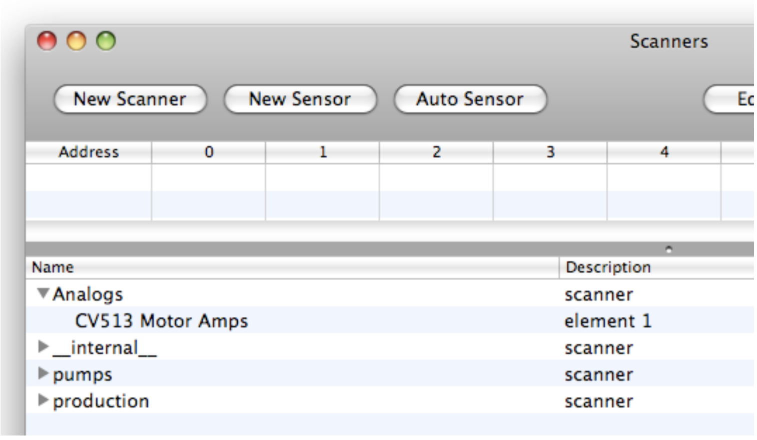
This allows for a quick way to scale and offset values read in from the PLC. In this example we only need the default values. Trending allows the value of the sensor to be saved to disk for later analysis or trending. If you want to view what this sensor has done over time you must tick the *Trend this Sensor* box. In this example we want to record the value of this sensor to disk every 2 seconds.

NOTE, trending can very quickly fill your hard drive if lots of sensors are set to 1 second or below. ConsultantX can handle this activity however it is recommended to install a second hard drive just for trend files.



Name :	CV513 Motor Amps
Sensor Type :	Standard
Element :	1
☐ Use Bit	0
Decode as :	Binary
Min :	0.00
Max :	500.00
Scale :	1.00
Offset :	0.00
☑ Trend this Sensor	
Trend Rate :	2.000 Seconds
Max Size of Trend Files :	28 Days
Units :	--

Your new sensor is now created and ready for use. To see it in the scanners window click the little triangle next to the scanner name to reveal all the sensors in a particular scanner.



Name	Description
▼Analogs	scanner
CV513 Motor Amps	element 1
▶__internal__	scanner
▶pumps	scanner
▶production	scanner

Creating a Raw Data Sensor

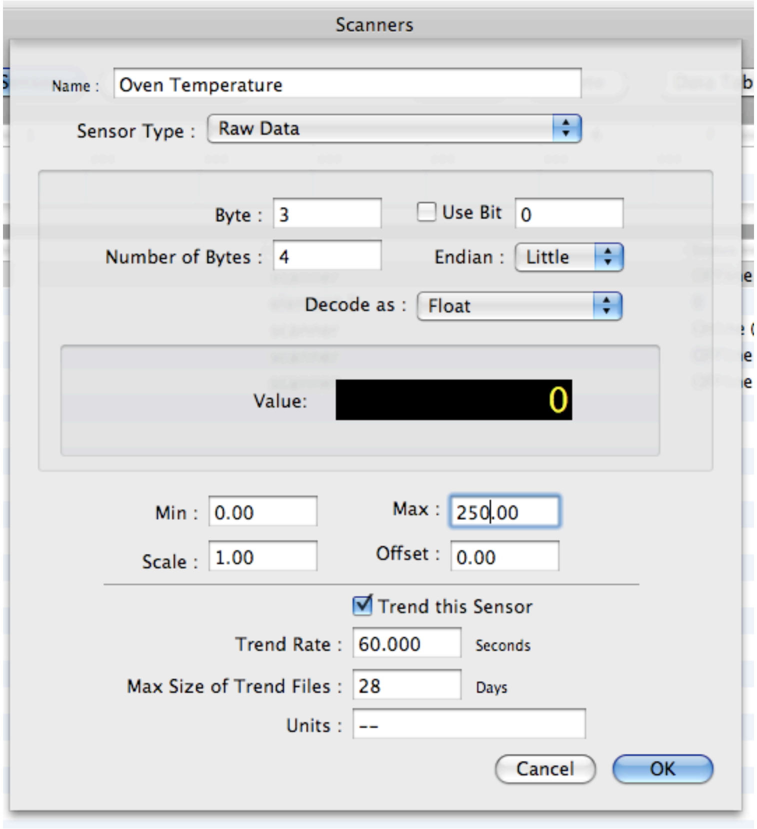
A raw data sensor access the actual bytes returned from the PLC or device in the underlying communications packet. Some communication scanners only return raw data. Ethernet/IP scanners are an example because the underlying data can be of a complex nature where there is a combination of data types that is unknown to ConsultantX.

Select the scanner in the scanner window that is to own this sensor and then click the *New Sensor* button.

Select *Raw Data* from the *Sensor Type* menu.

In this example we are creating a sensor that accesses 4 bytes starting a byte 3. The 4 bytes are to be interpreted as a float point number and are arranged in little endian format.

To help you make sure you have accessed the data correctly the value of the sensor is display in the large black box in the middle of the dialog box. This allows you to play around with the various settings until you get the number you are look for.



Scanners

Name : Oven Temperature

Sensor Type : Raw Data

Byte : 3 ☐ Use Bit 0

Number of Bytes : 4 Endian : Little

Decode as : Float

Value: 0

Min : 0.00 Max : 250.00

Scale : 1.00 Offset : 0.00

☑ Trend this Sensor

Trend Rate : 60.000 Seconds

Max Size of Trend Files : 28 Days

Units : --

Min, Max, Scale, Offset and trending are all the same for this type of sensor as for the standard sensor above.

Derived Sensors

Derived sensors are internal calculations that allow you to combine sensors and constants in complex mathematical expressions. In this example we are going to create a new sensor that is the addition of the Oven Temperature sensor created above with the internal or built in sensor *secondsOfMinute*. This sensor returns the current number of seconds in the current minute, a value that changes from 0 to 59.

The syntax of derived sensors is very simple, sensors that are already defined can be used by including the scanner name and the sensor name in square brakets.

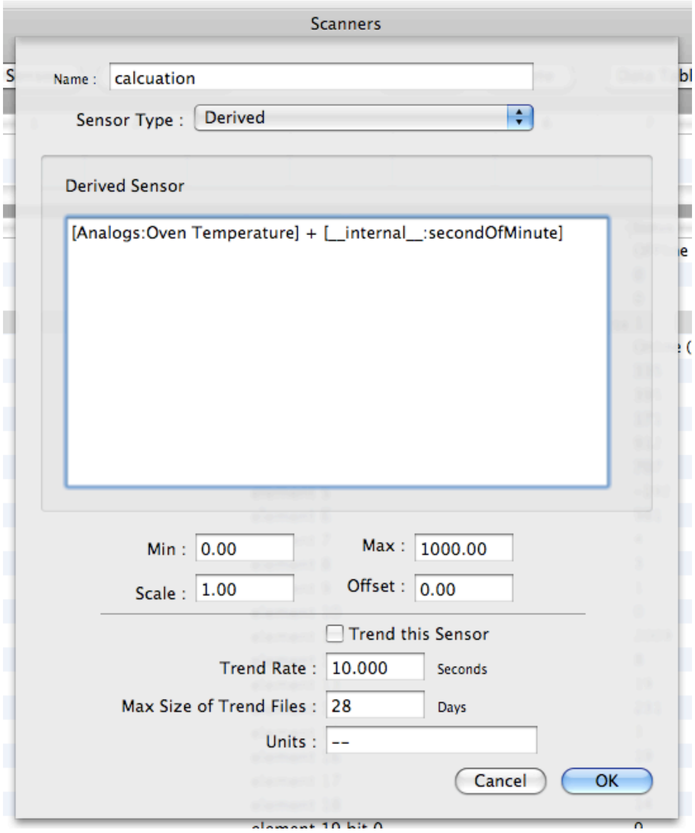
[scanner:sensor]

just make sure you spell the scanner and sensor name exactly the same and it is case sensitive.

The calculation you want this sensor to return is just the same as writing out formula.

Operators that are supported are:-

+	addition
-	subtraction
*	multiplication
/	division
^	exponent (to the power of)
>	greater than
<	less than
>=	greater then or equal to
<=	less than or equal to
!	logical NOT
!=	NOT equal
&&	Logical AND
&	Bitwise AND
| |	Logical OR
|	Bitwise OR
==	Equality
%	modulus



Scanners

Name : calcuation

Sensor Type : Derived

Derived Sensor

[Analogs:Oven Temperature] + [__internal__:secondOfMinute]

Min : 0.00 Max : 1000.00

Scale : 1.00 Offset : 0.00

□ Trend this Sensor

Trend Rate : 10.000 Seconds

Max Size of Trend Files : 28 Days

Units : --

Communications Fail Sensor

A communications fail sensor allows you to access, display and trend if the scanner is communicating with the device.

Status Sensor

A status sensor allows you to access, display and trend the error codes from the underlying scanner and communications protocol. When communications is healthy the status will be 0. When communication is not healthy the status sensor will give you an indication of what is wrong. The codes that are returned are from the underlying protocol.

Pinging

If the underlying communications protocol is based on TCP/IP networking you can access, display and trend 3 special sensors associated with ping.

Pinging sensors return 1 if a ping was successfully returned and 0 if not.

Ping Time sensors return the time in seconds for the last ping. This sensor is very useful to monitor the performance of your network.

Ping average sensors return the average of the last 10 ping times.

Howto's

To learn more about ConsultantX refer to the other ConsultantX howto's