Delivering Arduino Data to Thingworx

Using MQTT over a Network

By
Bill Reichardt (bill.reichardt@thingworx.com)

ThingWorx™
A PTC® Business

Overview

* Arduino is a low resource platform - Not suitable for any ThingWorx SDK
* MQTT protocol was designed for low resource platforms
* ThingWorx provides a free MQTT extension

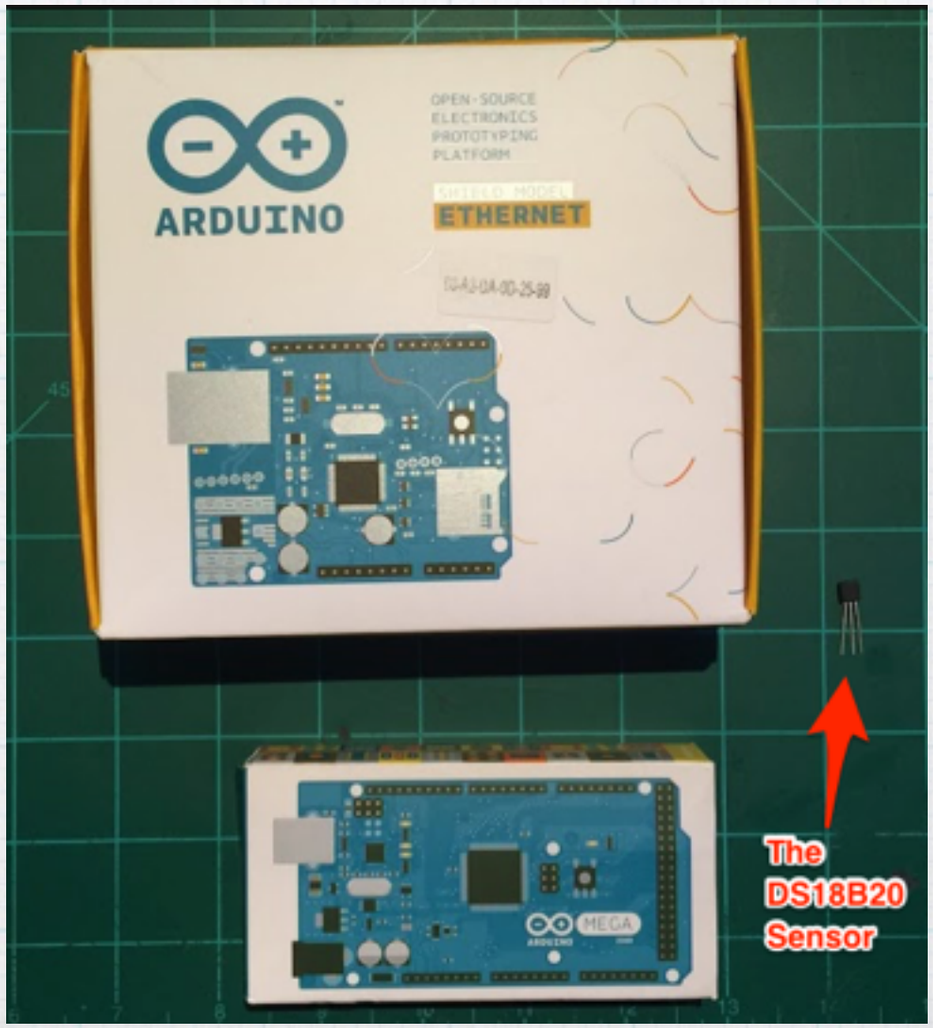
MQTT Protocol

TCP/IP

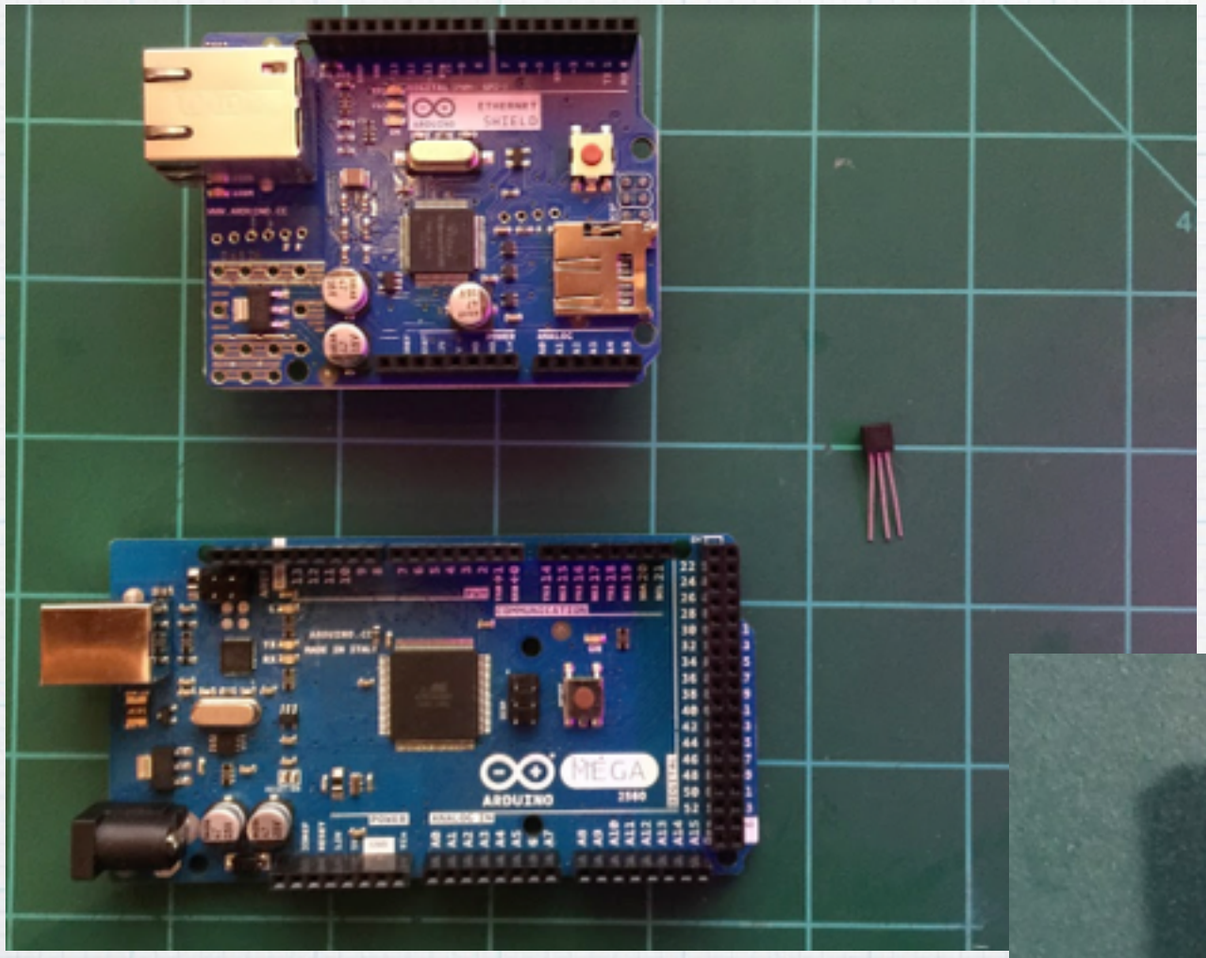
MQTT Protocol TCP/IP

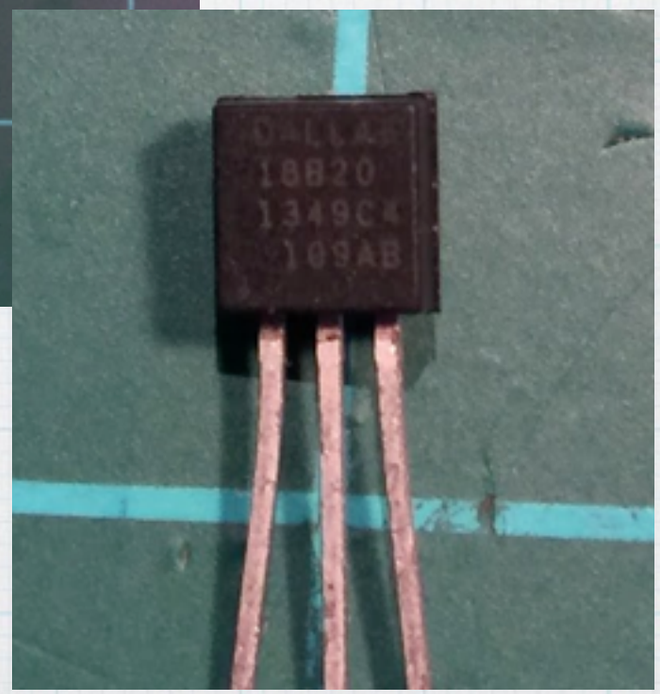
The Hardware

* 1 Arduino, I used the Mega2560
* 1 Arduino Ethernet Shield
* 1 Dallas Systems DS18B20 “One Wire” Temperature sensor

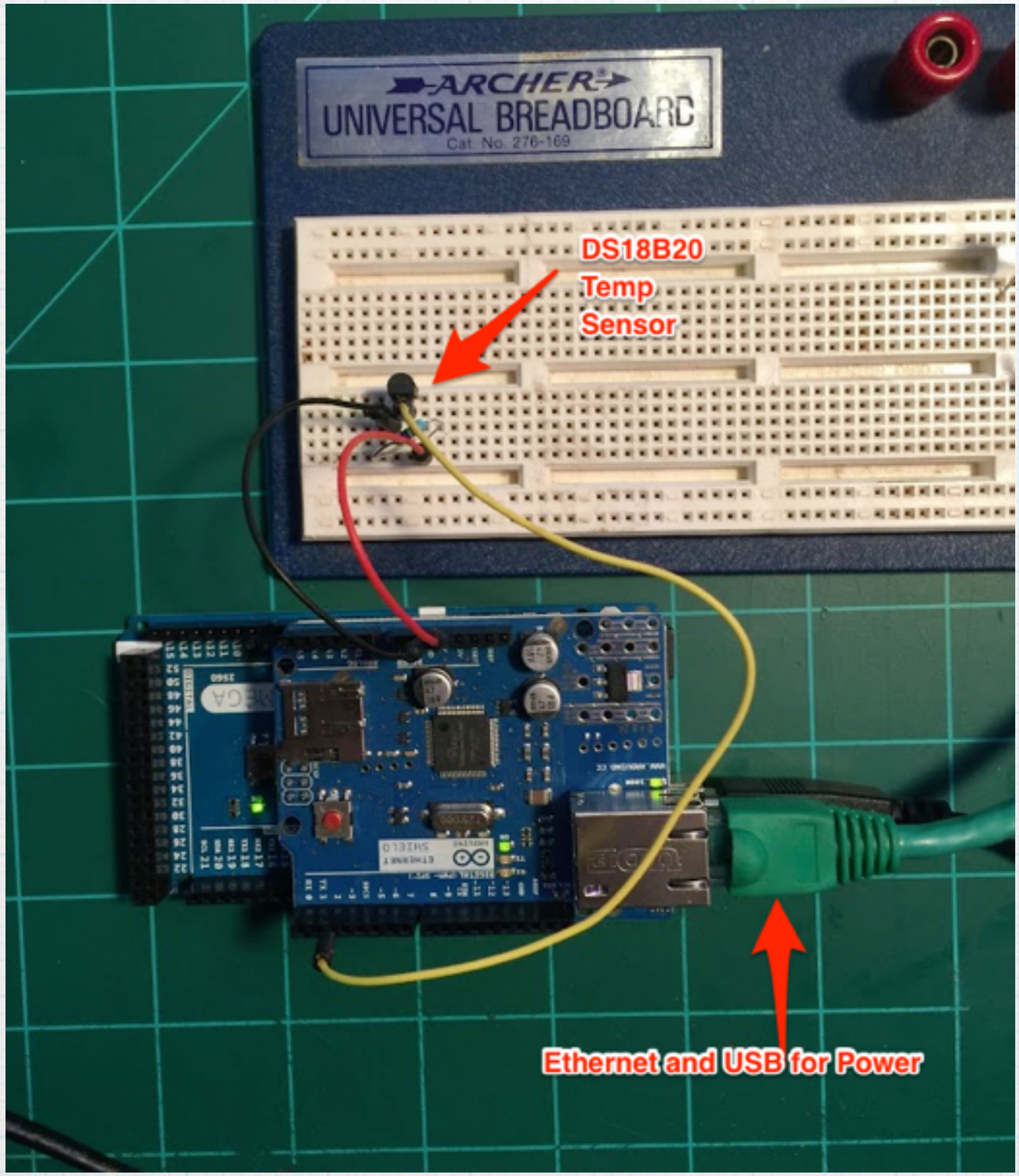


Hardware Out of the Box

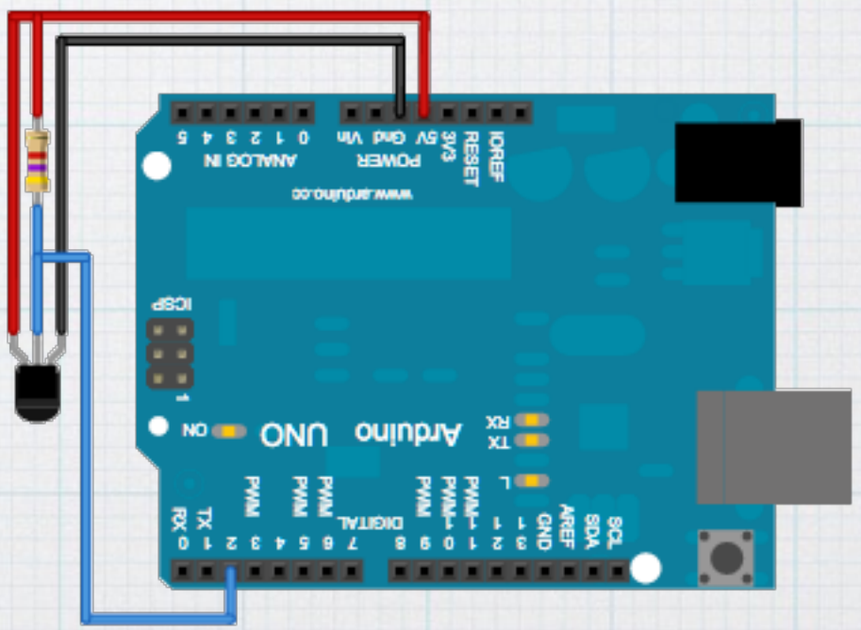




Assembled

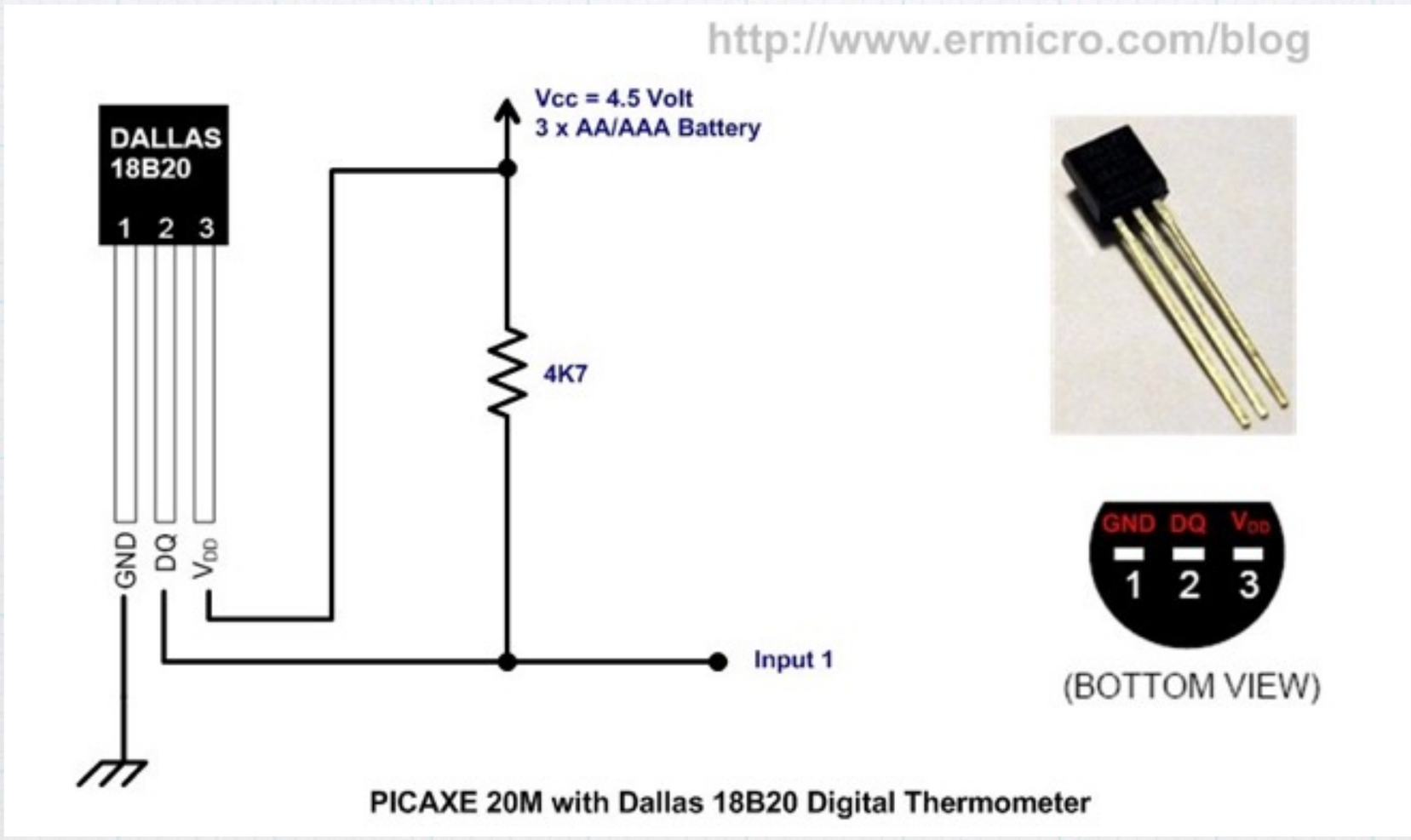


* 4.7 KΩ Resistor is required (see next slide)
* The Sensor is powered by the Arduino via USB
* Multiple Sensors can share the same pins and resistor



Warning!

* Do not reverse +5V and GND on the 18B20!!!
* The sensor will overheat and burn out in seconds
* Buy more than one



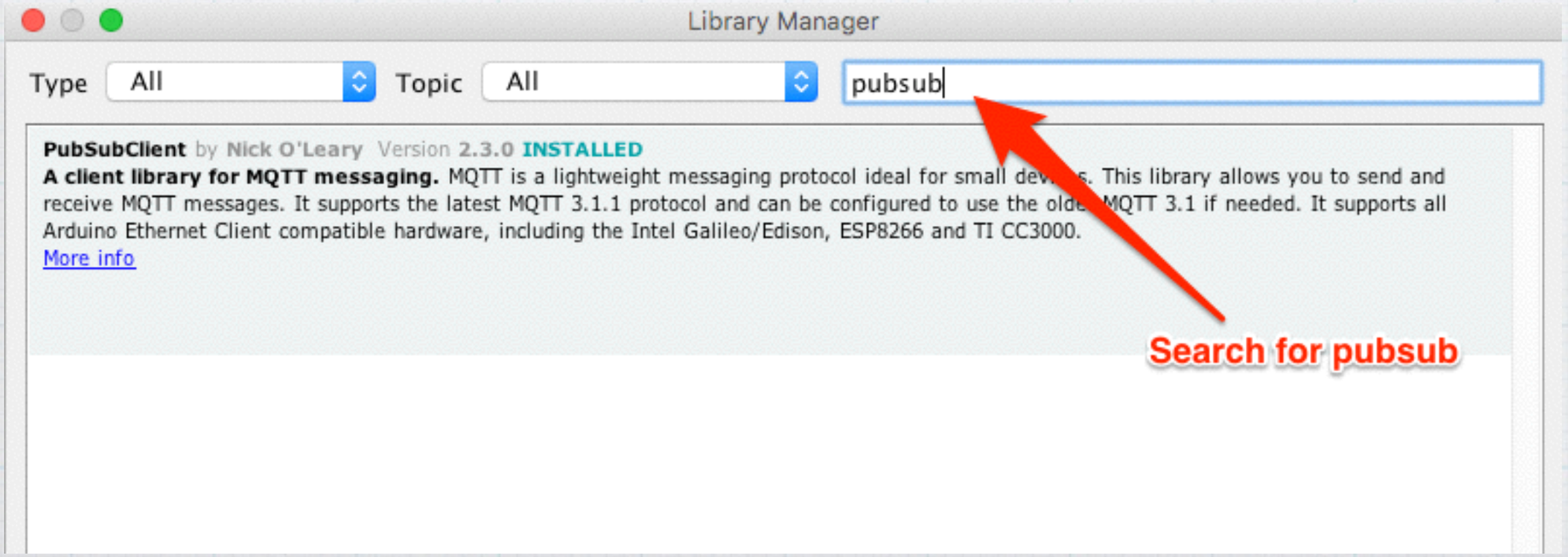
More detailed description of this hardware can be found at http://www.hacktronics.com/Tutorials/arduino-1-wire-tutorial.html

Create an MQTT Server 

* An MQTT server stores property updates from devices
* ThingWorx can apply these updates to Things on the server
* **Mosquitto MQTT server**
  * http://mosquitto.org/
* OSX Install http://www.xappsoftware.com/wordpress/2014/10/30/install-mosquitto-on-mac-os-x/
* Windows comes with an installer
* You can usually install it on the same system that ThingWorx is on.

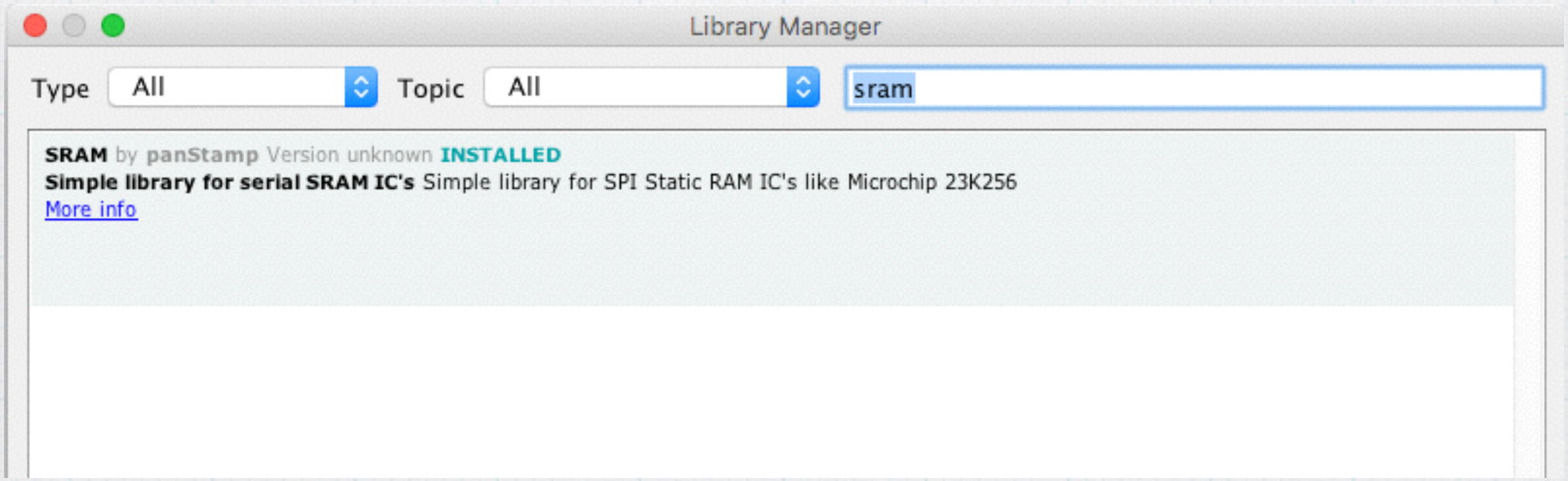
Install pubsubclient

* You will need the **pubsubclient** library installed in your Arduino IDE
* Use the Sketch>Include Library>Manage Libraries.. Menu to install it



Install sram

* You will need the **sram** library
* It is required by pubsub
* You may have to restart before you can use your installed libraries



Test Out Your Connection

* In your Arduino IDE menu go to File>Sketchbook>libraries>pubsubclient>mqtt_stream
* This should open the mqtt_stream.ino example
* Make the changes shown here and upload it to your Arduino
* Script creates a topic called **outTopic**

Testing Your Server

* Use the Mosquitto command line client to subscribe to your topic
* Run:
  * **mosquitto_sub -h 127.0.0.1 -t outTopic**
* -h is the server, 127.0.0.1 assumes you are running this command on the same host
* If everything is working, you should see “Hello World” in appear in your console when you turn your Arduino on.

Connecting the Sensor

* There should be a **DS18x20_MQTT.ino** sketch accompanying this slide deck
* You may have to install the **OneWire** library to compile it
* Make the same IP address and Mac changes you made to the MQTT_Stream.ino example
* Note the topic changes, outTopic has been replaced with /Thingworx/DS18Thing/F & C to report temp readings

```
    else if (cfg == 0x20) raw = raw & ~3; // 10 bit res, 187.5 ms
    else if (cfg == 0x40) raw = raw & ~1; // 11 bit res, 375 ms
    //// default is 12 bit resolution, 750 ms conversion time
  }
  celsius = (float)raw / 16.0;
  fahrenheit = celsius * 1.8 + 32.0;
  Serial.print("  Temperature = ");
  Serial.print(celsius);
  Serial.print(" Celsius, ");
  Serial.print(fahrenheit);
  Serial.println(" Fahrenheit");
  char tempString[100];
  String(fahrenheit).toCharArray(tempString, sizeof(tempString));
  client.publish("/Thingworx/DS18Thing/F", tempString);
  String(celsius).toCharArray(tempString, sizeof(tempString));
  client.publish("/Thingworx/DS18Thing/C", tempString);
  client.loop();
}
```

These topics map to Thingworx properties

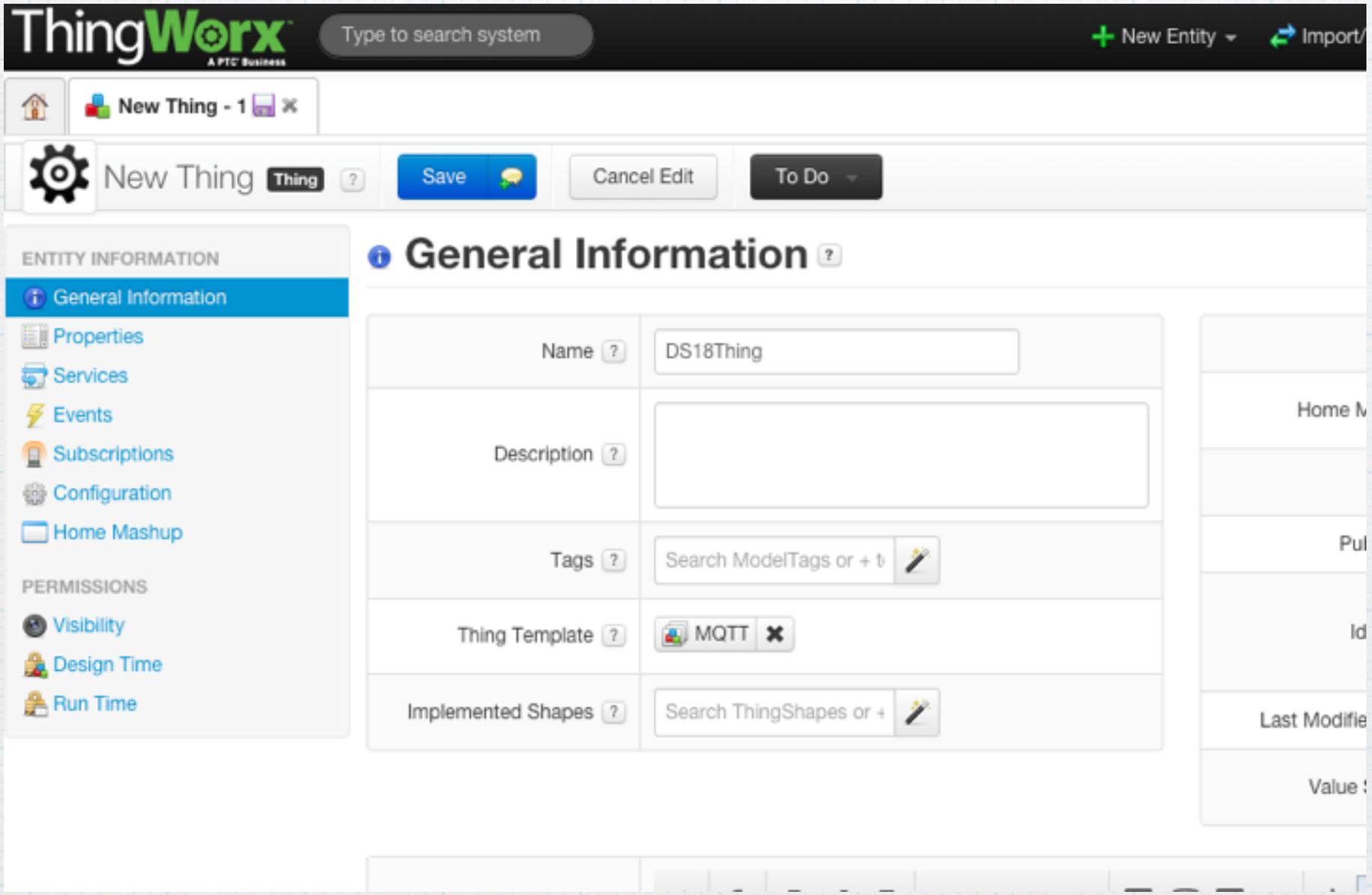
Installing the MQTT Extension

* Download and install the MQTT extension from the ThingWorx Marketplace at:
  * http://marketplace.thingworx.com/downloads/mqtt/
* When done you will now have these things…

☐		MQTTSubscriber	MQTT subscriber interface	ThingTemplate
☐		MQTTConnection	MQTT connection interface	ThingTemplate
☐		MQTT	MQTT interface	ThingTemplate

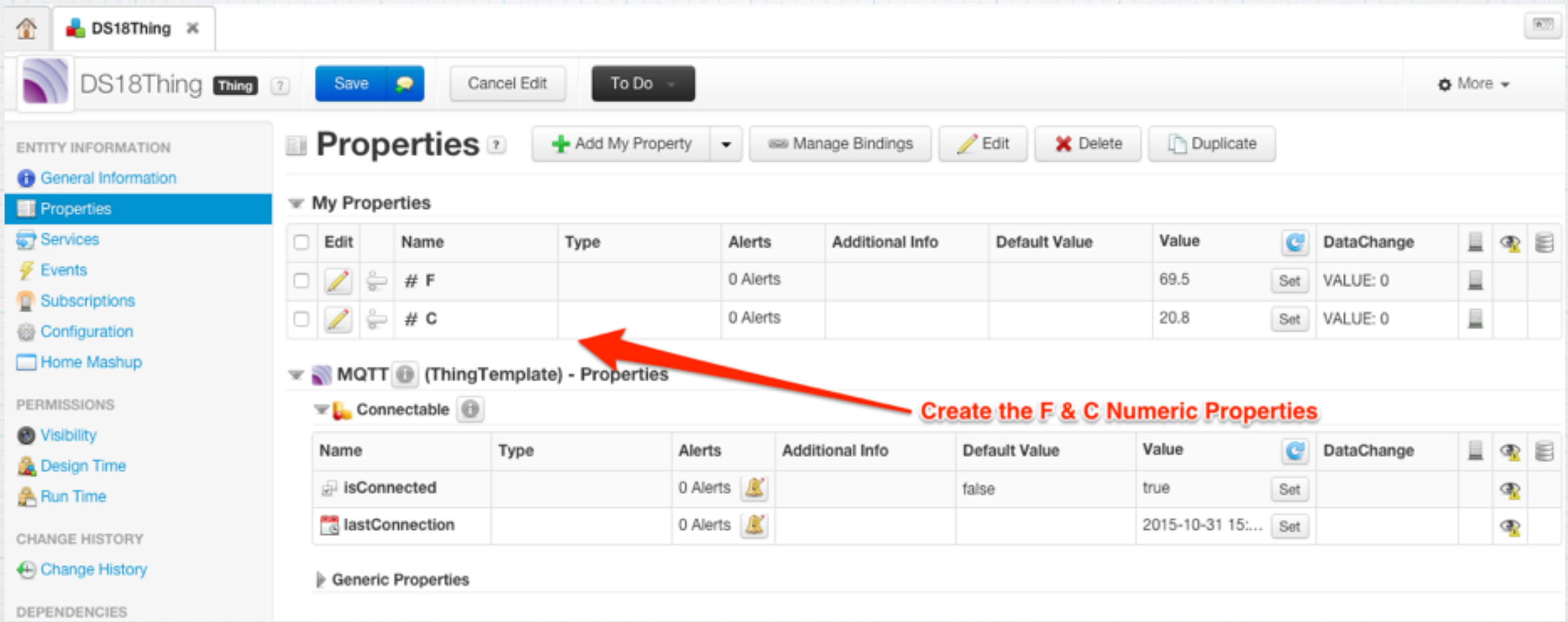
Creating a Thing for Your Data

* Create A new Thing using the MQTT ThingTemplate called DS18Thing



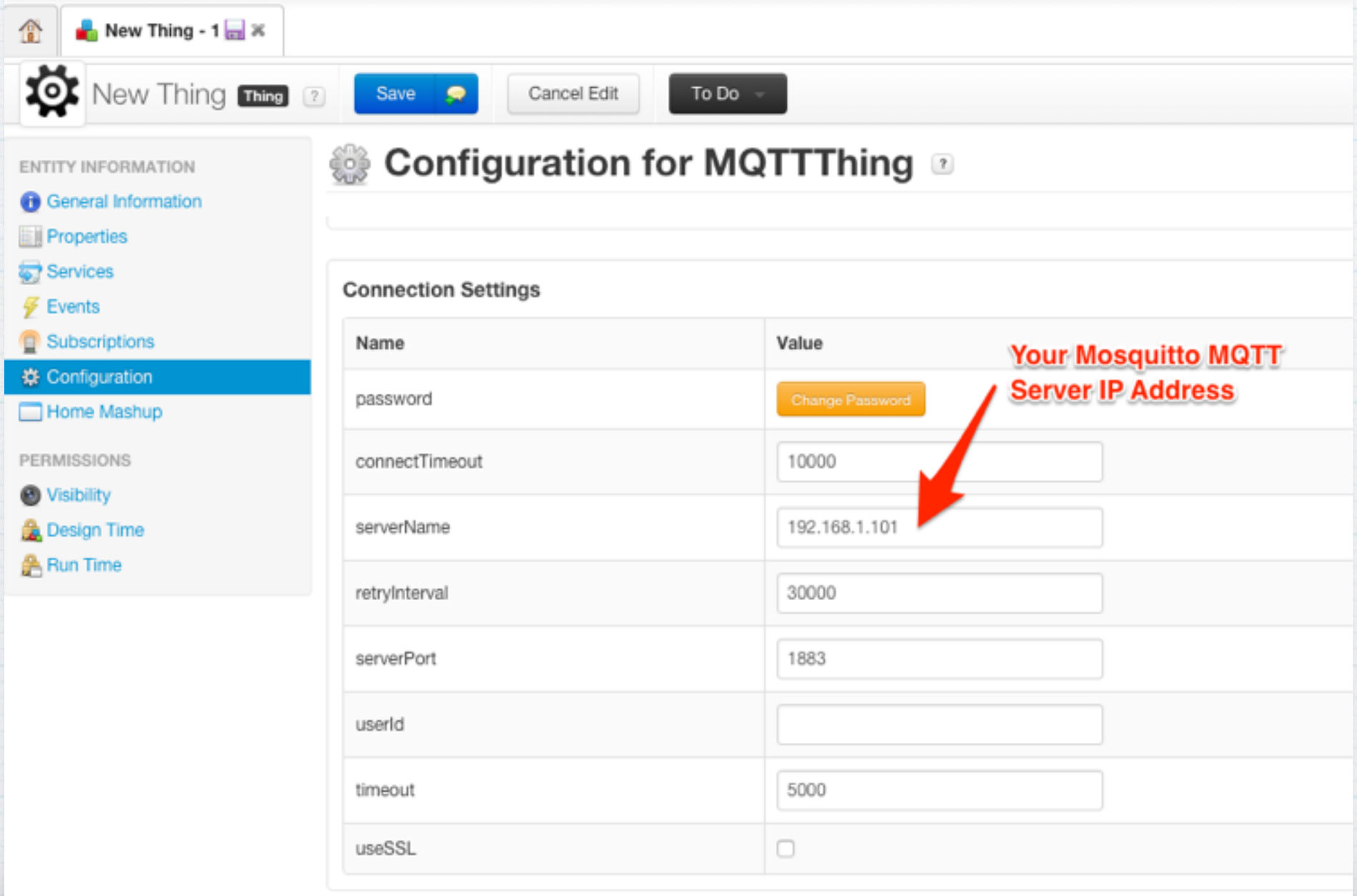
Give it Properties

* Since we are sending F & C temperature values, Give your thing an F & C number property.



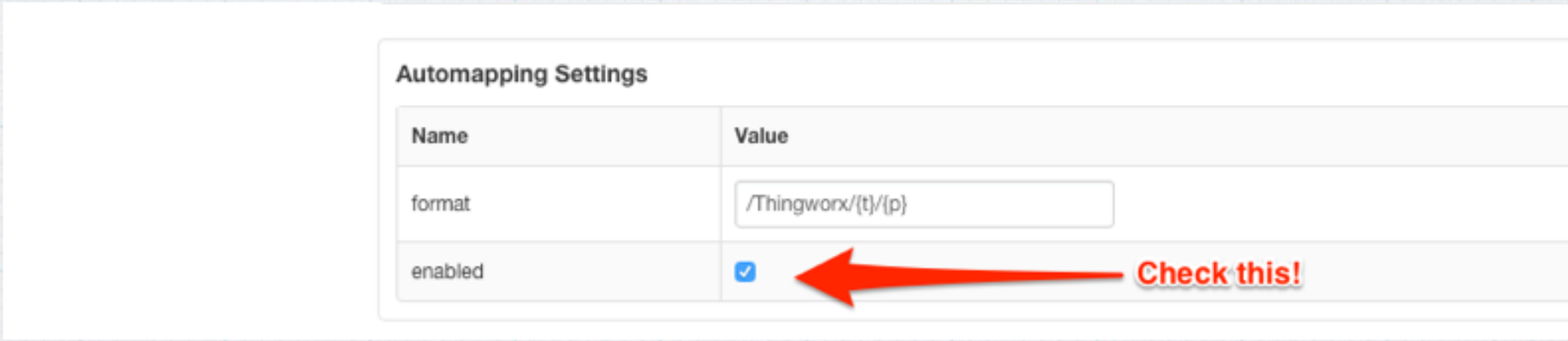
Configure Your Thing

* The configuration tab is where you enter your server settings.
* Just enter your server ip address.

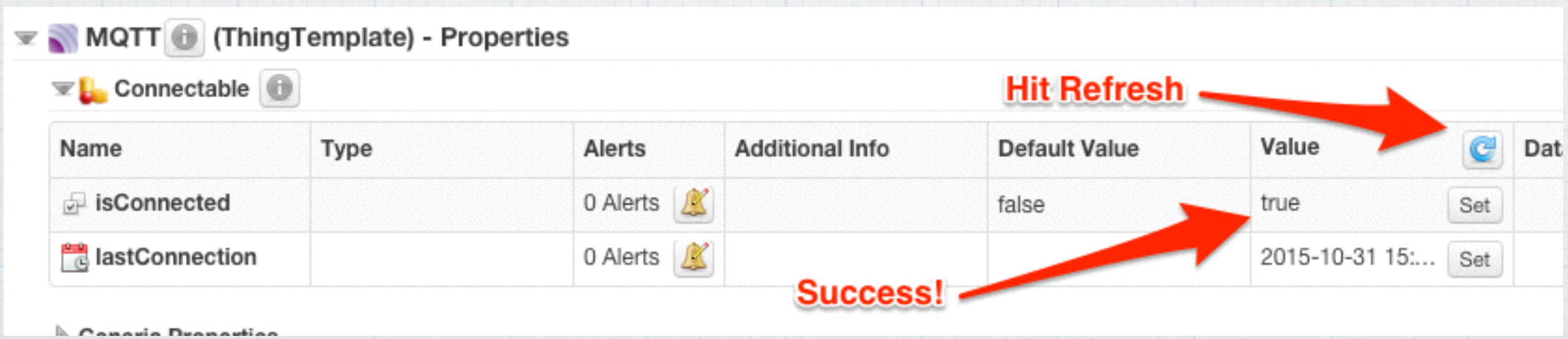


Automating Makes it Easy

* The Default Automapping settings work
* The two topics we are using are
  * Fahrenheit Temp - /Thingworx/DS18Thing/F
  * Celcius Temp - /Thingworx/DS18Thing/C
  * Generally /Thingworx/<Thingname>/<propname>
* Make sure you check enabled!

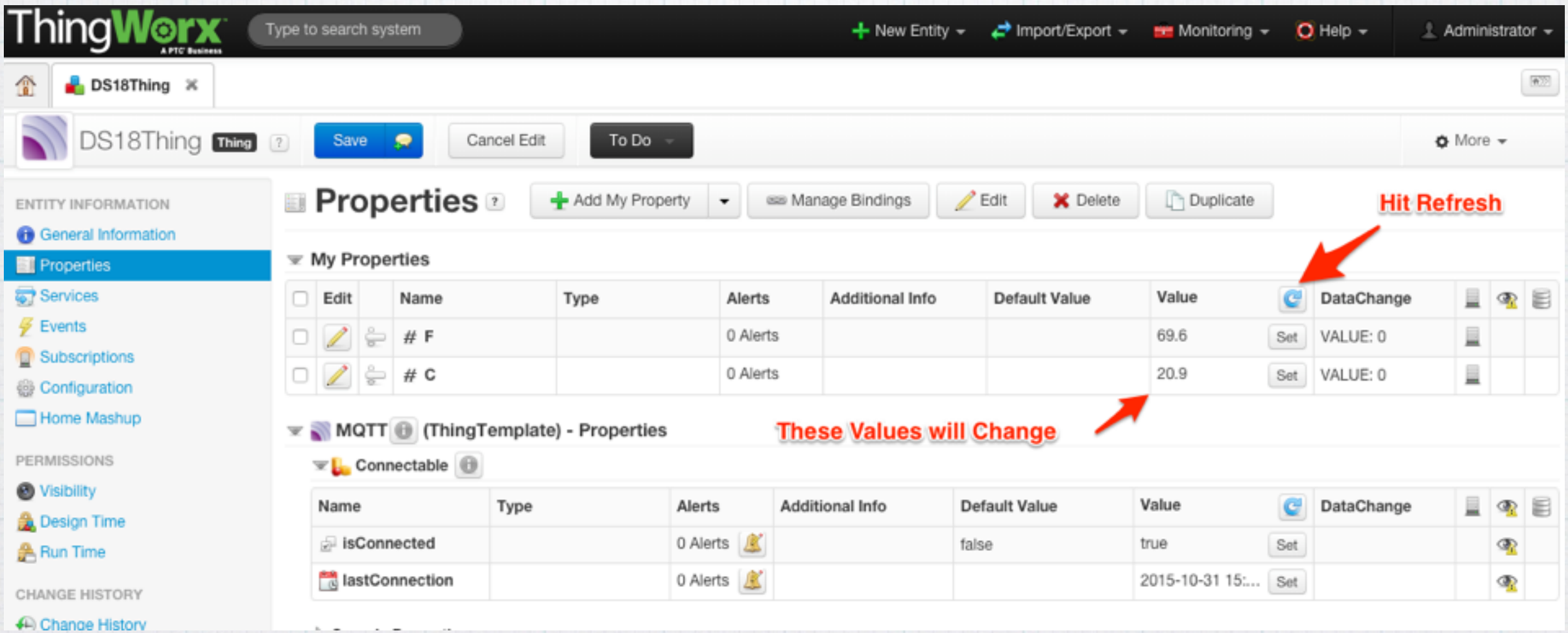


* Save this thing. If all is well you will see



Watch Your Data Roll In

* Once your connection is established, touch the sensor
* Hitting the refresh button will allow you to observe the temperature changes



Conclusion

* Multiple DS18 Temperature sensors can share the same, single in on your Arduino so you have created a platform to take multiple temperature Readings
* The MQTT server can store readings and deliver them when the ThingWorx server is visible on your network
* MQTT protocol is higher performance and easier to create than a REST web service connection
* I want to credit a great topic in the ThingWorx community for getting me started on this project.
  * https://community.thingworx.com/message/5063#5063