

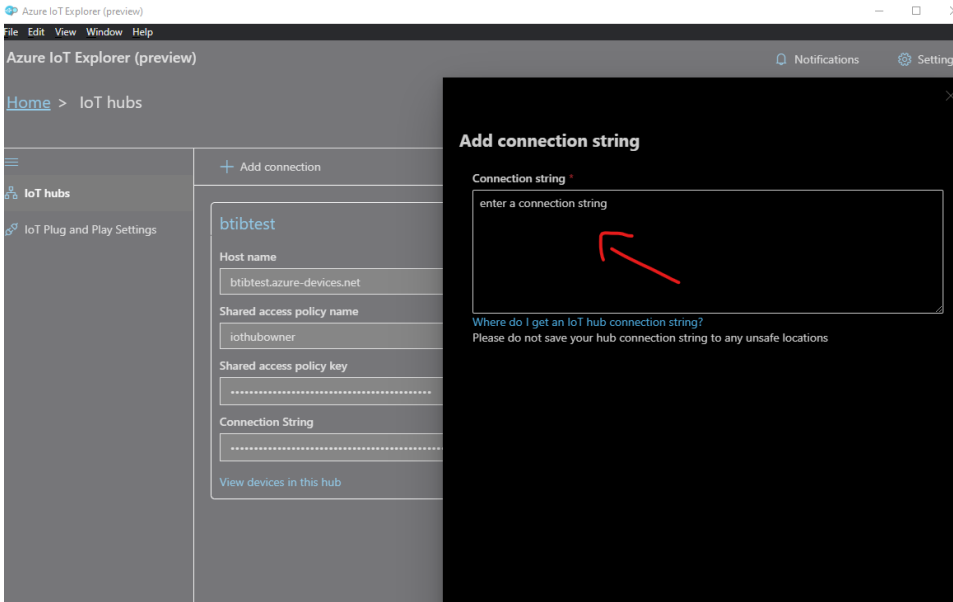
Step 3 Send messages to IoTHub from Niagara

Unfortunately Azure portal does not provide an interface to see messages directly but there is windows utility called **Azure IoT Device Explorer** that does it (see the tutorial below).

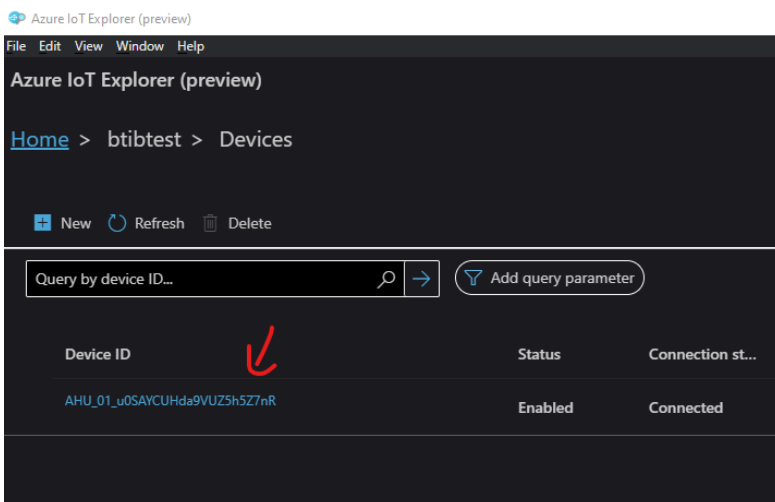
If you're a developer, you can also use Visual Studio Code and the extension dedicated to IoTHub.

Set up IoT Device Explorer

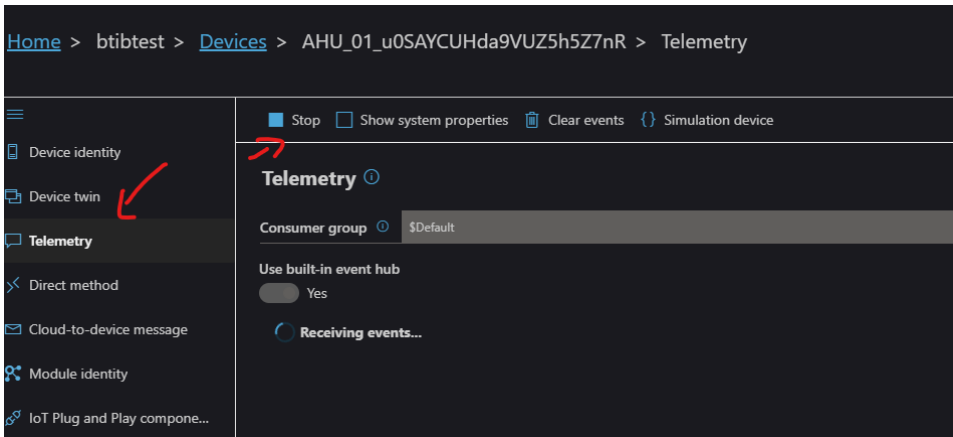
1. Download the msi from this link <https://github.com/Azure/azure-iot-explorer/releases> and install it on your Windows machine.
2. Open the utility and paste your connection string.



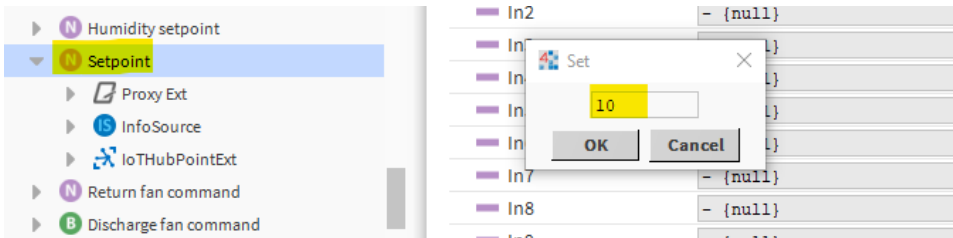
3. Open the device view..



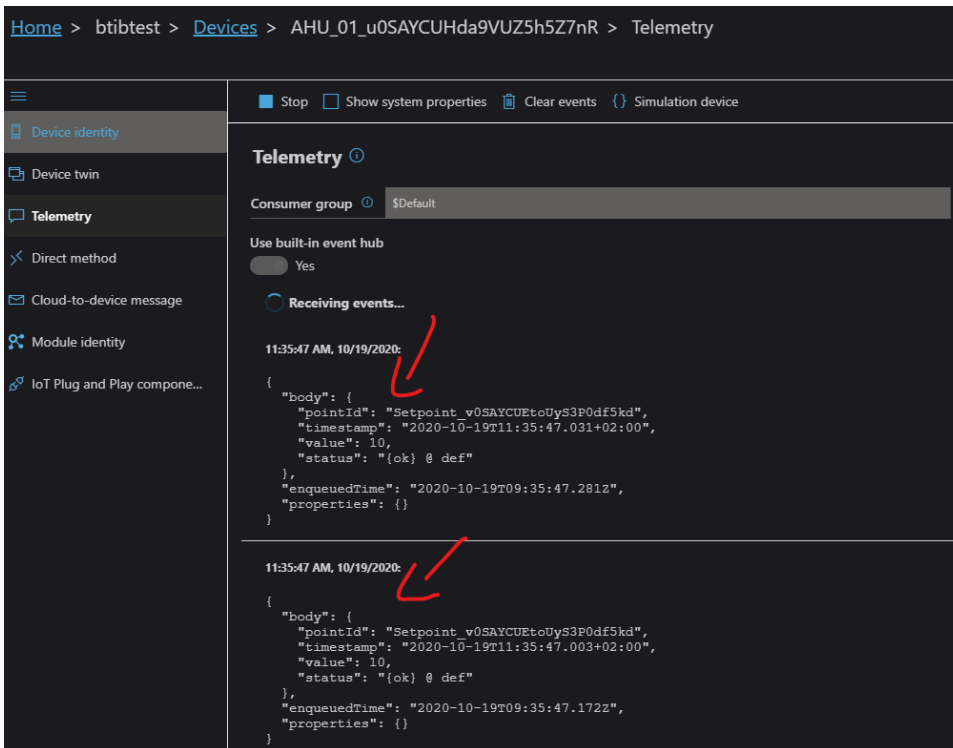
4. Go to telemetry and start listening fir messages.



5. Go to your point and trigger a change.



6. And voilà your messages is received!. note that we received 2 events (value change and status change). the body message is the same because by default the templates are the same, to change this go to the connector advanced settings.



7. No go to the reference and change a slot value.

The screenshot shows the 'My Network' tree on the left with 'Ticket1' selected. The 'Property Sheet' on the right displays the following properties:

Ticket1 (Fix Reference)	
Id	SlkjauSHBYGy7656S876AByS66
Link	https://example.com/t/SlkjauSHBYGy7656S876AByS66
name	Ticket111
IoTHubReferenceExt	Io T Hub Reference Ext

8. You should see a new message sent.

The screenshot shows the 'Telemetry' page for device 'AHU_01_u0SAYCUHda9VUZ5h5Z7nR'. The 'Consumer group' is '\$Default'. The 'Use built-in event hub' toggle is set to 'Yes'. The 'Receiving events...' status is active. A red arrow points to a received event at 11:50:20 AM, 10/19/2020:

```
{
  "body": {
    "pointId": "SlkjauSHBYGy7656S876AByS66",
    "name": "Ticket111",
    "link": "https://example.com/t/SlkjauSHBYGy7656S876AByS66",
    "id": "SlkjauSHBYGy7656S876AByS66"
  },
  "enqueuedTime": "2020-10-19T09:50:20.348z",
  "properties": {}
}
```

Next Step

Step 4 Send alarms to IoT Hub