

# Connected Field Service IoT Central

## Guide

# Developer



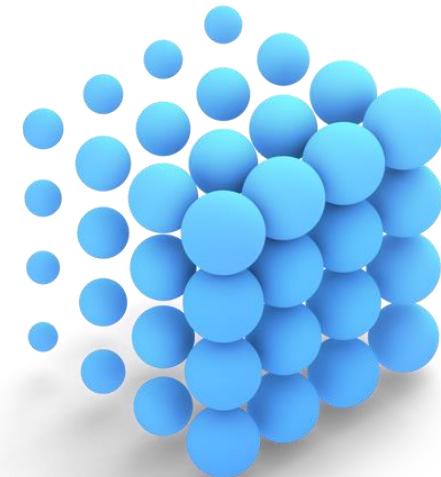
# Greg Degruy

Software Engineer and Architect

{

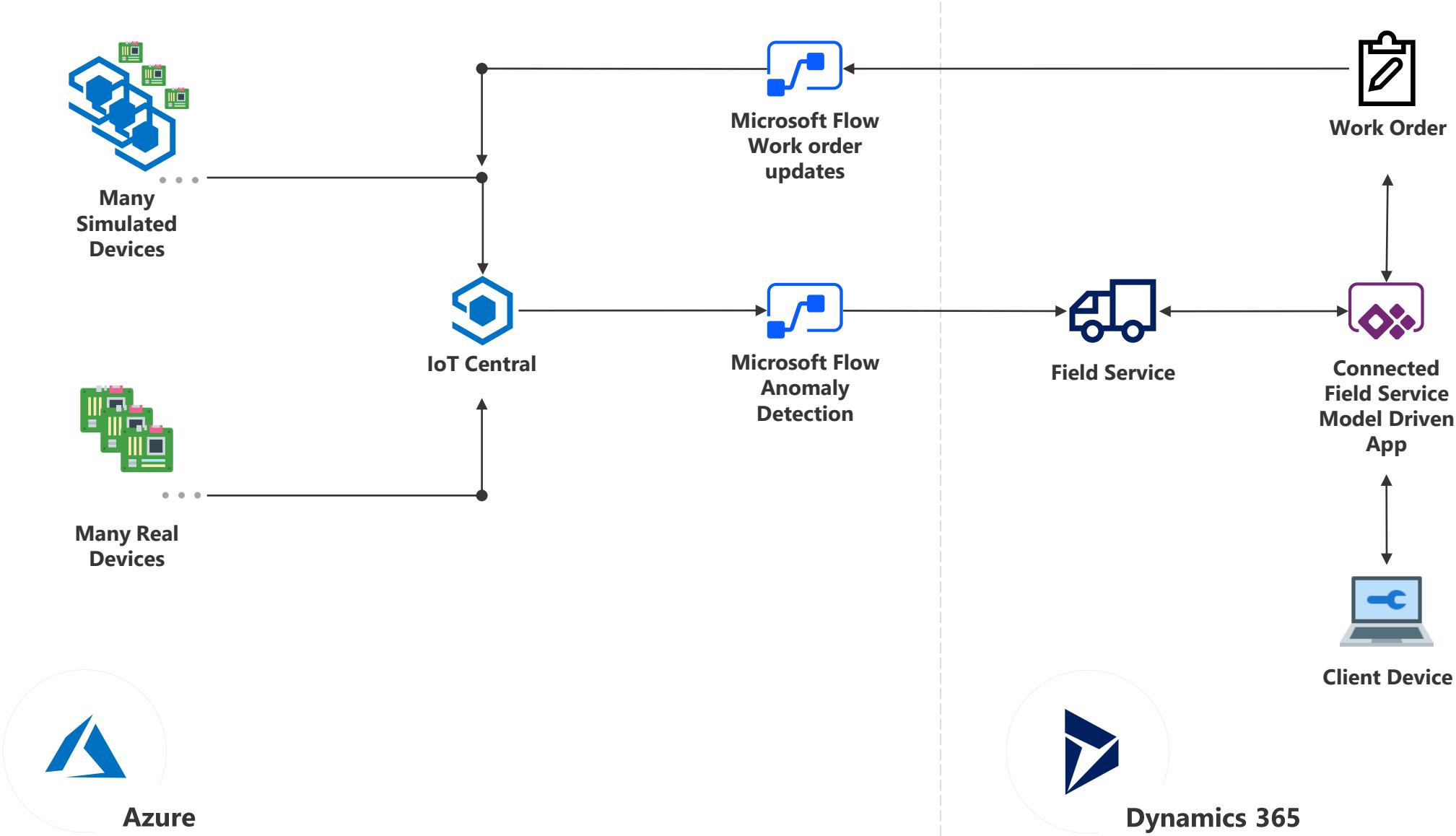
[github.com/gregdegruy](https://github.com/gregdegruy)

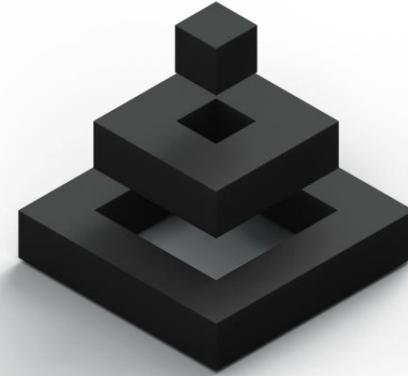
}



# Connected Field Service IoT Central { A bi-directional data integration }

CONNECTED FIELD SERVICE





# Setup IoT Central

Let's a go!

# Content

- 20 minutes
- You'll learn how to:
  - Create a device template
  - Define custom fields
  - View simulated telemetry
  - Create a Service Information model that will map to Dynamics 365



**Dynamics 365  
Field Service**



**Azure  
IoT Central**

# CFS IoT Central Home page

Browse to the Azure IoT Central Website  
<https://azure.microsoft.com/en-us/services/iot-central/>

Use Chrome, Edge or whatever you want. No favorites here.

1. Get started
2. New application. You can get back to your apps at any time from  
<https://apps.azureiotcentral.com/>



You may need to log in if you don't see your account name from the Azure provisioning section. If you skipped that section and have an Azure account already even better!

The screenshot shows two views of the Azure IoT Central interface. The top view is the landing page with the URL 'https://iotcentral.azureedge.net' in the address bar. It features the 'Azure IoT Central' logo, a sub-headline about simplicity, and a prominent blue 'Get started >' button. A large, semi-transparent mouse cursor arrow is positioned over the 'Get started' button. The bottom view is the 'Application Manager' page, which has a dark header with the 'Azure IoT Central' logo and a sub-header 'The location where you will find all the applications you create'. Below this is a large blue button with a white plus sign and the text 'New Application'. A second, smaller mouse cursor arrow points to this 'New Application' button. The bottom right corner of the screenshot shows a navigation bar with 'Portal', a user profile for 'Greg', and a search icon.

# Create Application

1. The Create Application form should have everything auto completed for you, so don't need to modify anything. Unless you want to choose Pay-As-You-Go, this will incur cost in Azure.



If you want to use Pay-As-You-Go, move onto the next slide and DO NOT do step 2 here.

2. Select Create

Create Application  
We just need a few things from you, so we can create your application

Choose payment plan

Trial   
Free trial for 7 days. No subscription required.

Pay-As-You-Go  
Price is based on the number of devices you use. Free for the first 5 devices. Subscription required. [Learn more](#)

Select an application template

Sample Contoso   
Get started with a predefined application for a connected device.

Sample Devkits  
Want to connect a Raspberry Pi or MXChip IoT DevKit? Start with this predefined app and get them connected in minutes.

Custom Application  
Start with a blank template and define your application from scratch.

Application Name \* 

URL \*   
 .azureiotcentral.com

By clicking "Create" you agree to the [Subscription Agreement](#) and [Privacy Statement](#). Provisions in the agreement with respect to pricing, cancellation fees, payment, and data retention do not apply to "Trial". "Pay-As-You-Go" requires an Azure subscription, and you acknowledge that this service is licensed to you under the terms applicable to your [Azure Subscription](#).

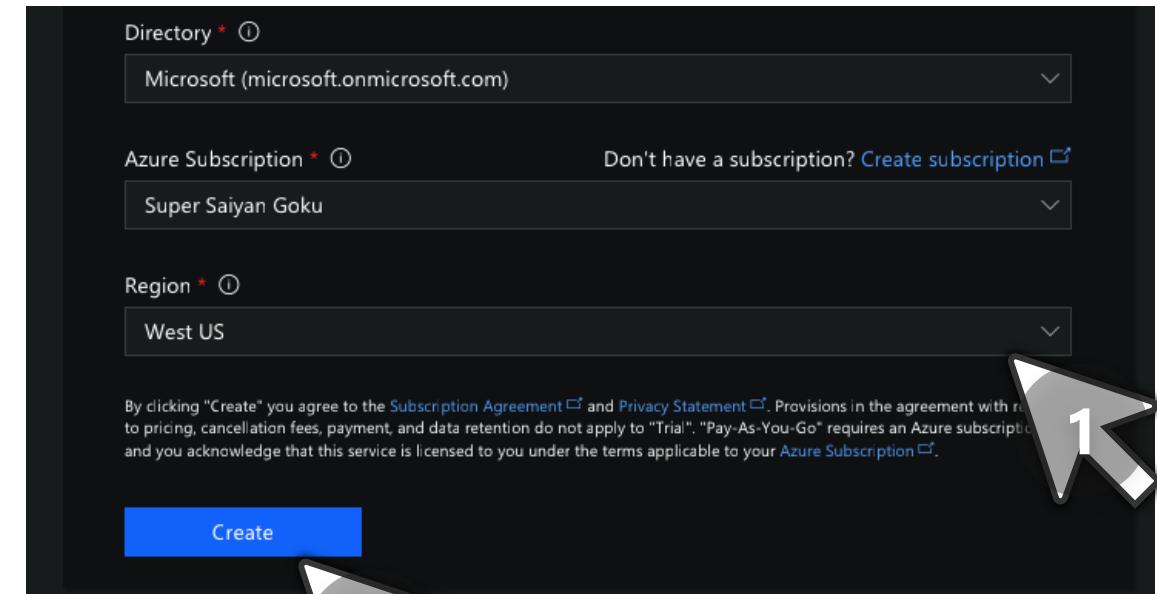


# Pay-As-You-Go

1. The Create Application for Pay-As-You-Go should have everything auto completed for you too, so no need to modify anything. Just take note of the Directory, Subscription, and Region that you want to deploy IoT Central too.
2. Select Create.



For the curious, if you visit your Azure resource groups you'll see an IOTC resource group containing your IoT Central application.



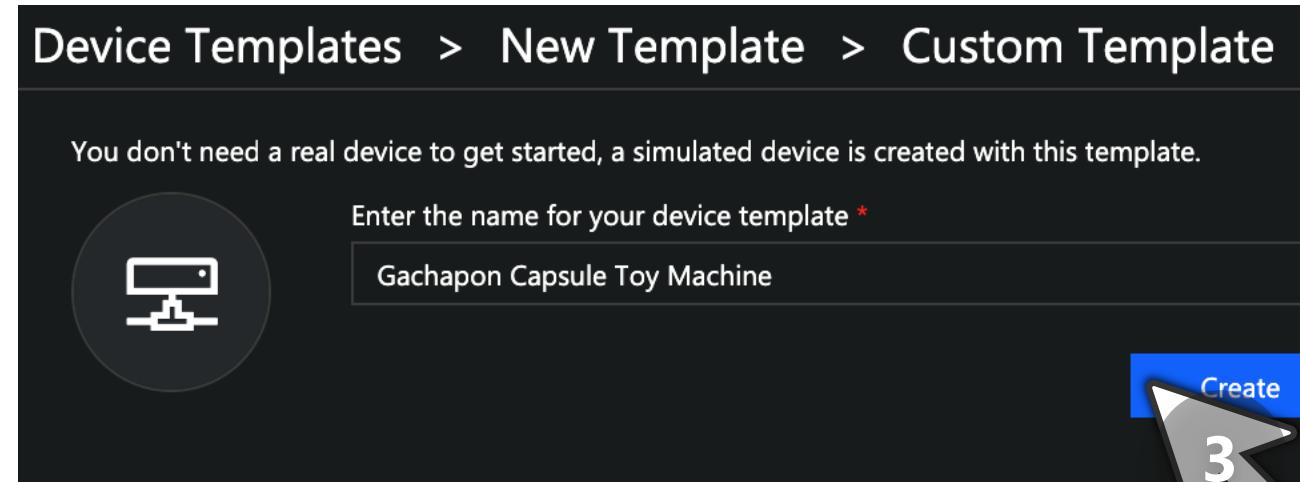
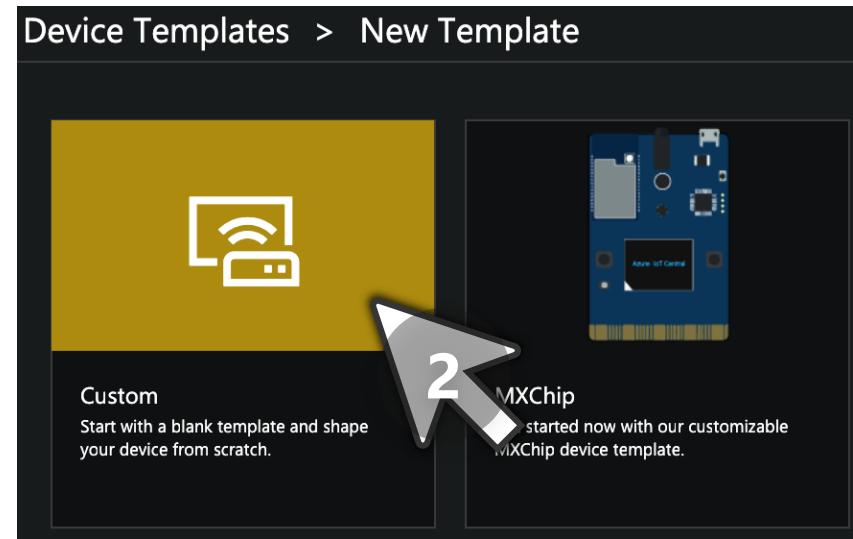
# Home Dashboard

1. You should be directed to the IoT Central Home Dashboard. All of the widgets are completely customizable. We want to add some simulated devices from our left hand menu, click it.
2. Specifically select Device Templates so we can create the simulated device.



# Device template

1. You should be directed to the Application Builder. Select Create Device Template.
2. Select Custom.
3. Give you're device template a name that best first your scenario or follow along with the Capsule Toy Machine example we're using, then click Create.



# Measurements

1. You should be directed to your simulated devices measurement page. We're going to add some telemetry, state, and event to our template before we connect this to Dynamics 365.
2. When we're done this dashboard is going to look so good you can literally eat it.



Device Template

## Gachapon Capsule Toy Machine (1.0.0)

Measurements   Settings   Properties   Commands   Rules   Dashboard

+ New Measurement

**Telemetry** ^

No measurements created

**State** ^

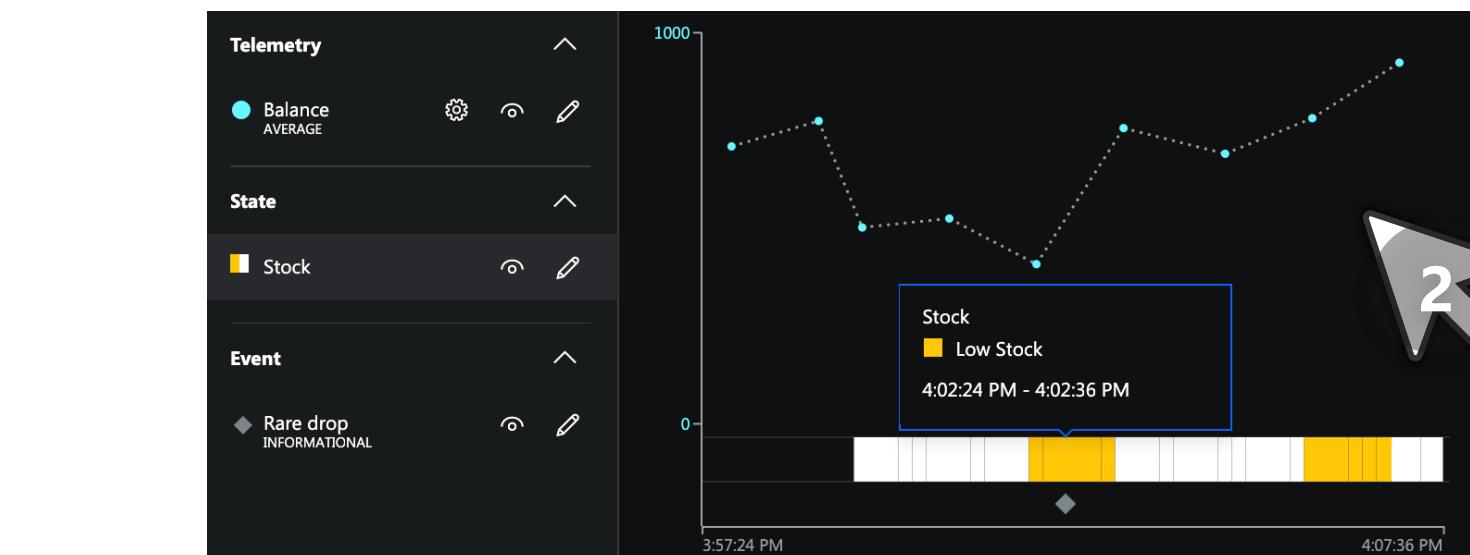
No measurements created

**Event** ^

No measurements created

1

Measurements are the telemetry, state, and event data that is sent from your device. Get started by adding a new measurement. [Learn more...](#)



# Telemetry

1. New Measurement.
2. Three measurement options will then be presented to you. Select Telemetry.

Device Template

## Gachapon Capsule Toy Mach... (1.0.0)

Measurements   Settings   Properties   Commands   Rules   Dashboard

+ New Measurement

**Telemetry**

No measurements created

**State**

No measurements created

**Event**

No measurements created

Measurements are the telemetry, state, and event data that is sent from your device. Get started by adding a new measurement. [Learn more...](#)

1

2

+ New Measurement

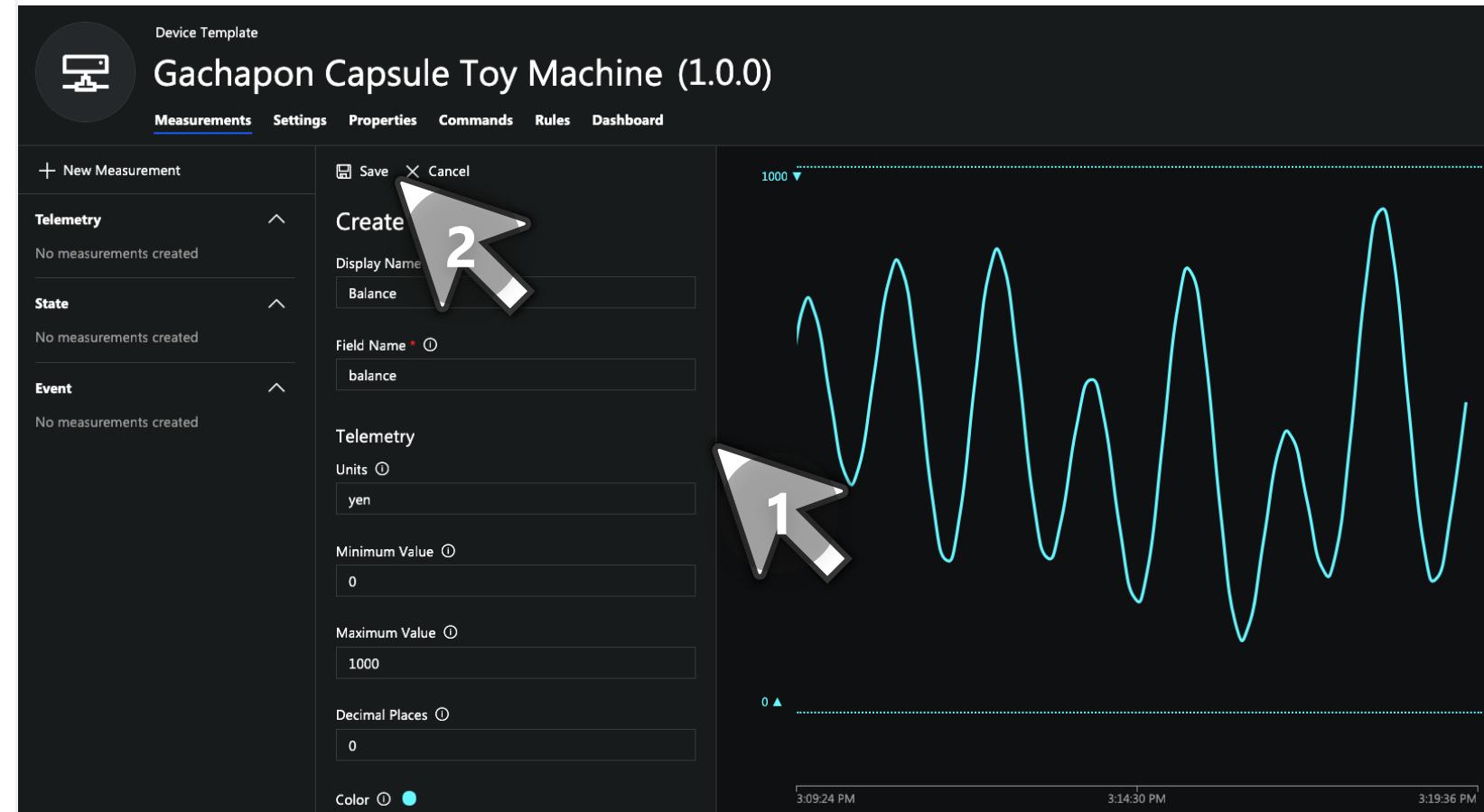
Telemetry  
Time series data from the device  
(e.g. Temperature)  


State  
Current status of the device or its components  
(e.g. Online)  


Event  
Intermittent signal from the device  
(e.g. Alarm)  


# Create Telemetry

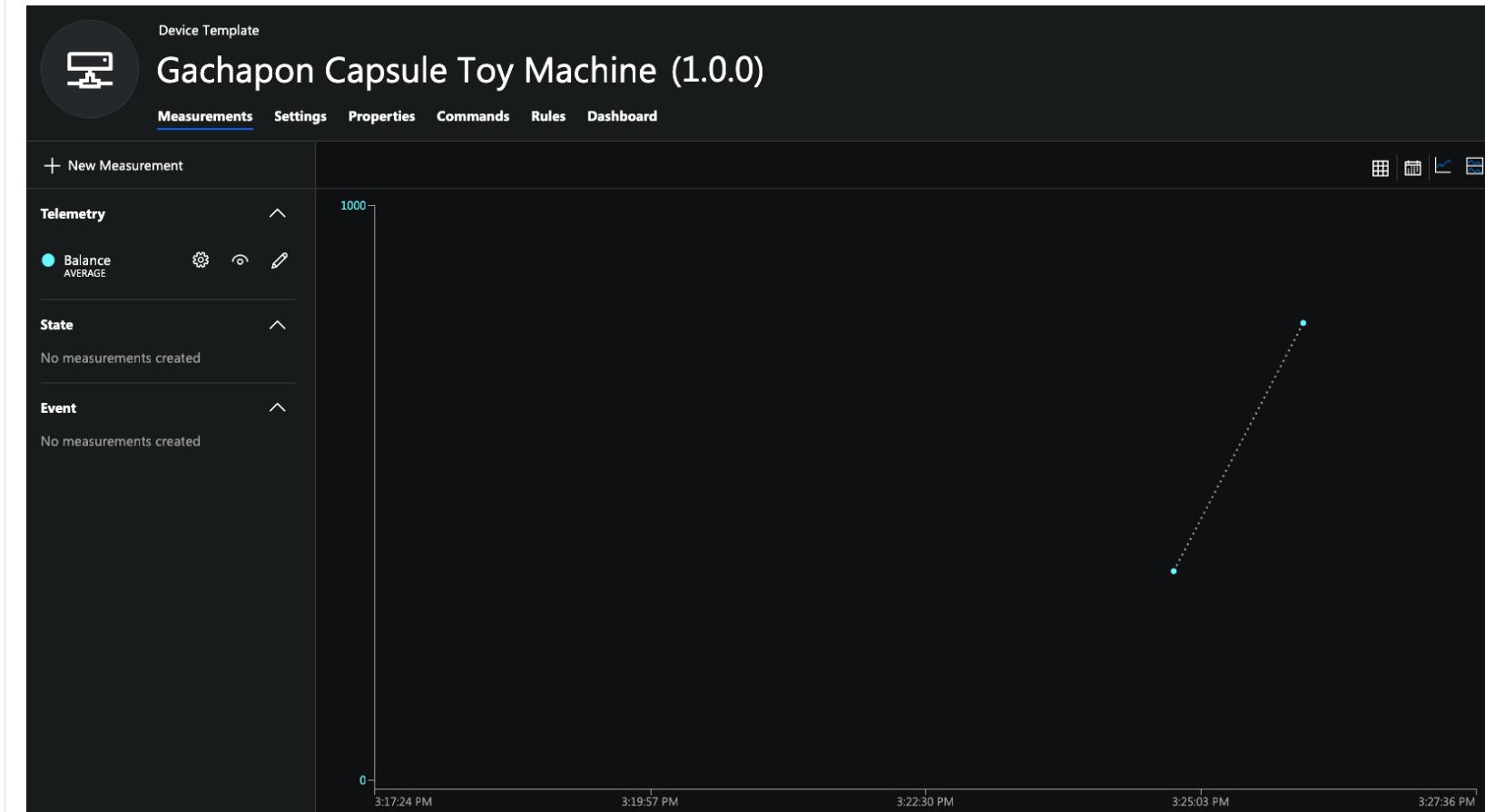
1. Add these four telemetry fields
  - Display Name Balance
  - Field Name balance
  - Units yen
  - Min 0
  - Max 1000
  - Decimal places 0
2. Select Save from the Create Telemetry blade.



# Telemetry incoming

Random system generated sample data based on our temperature telemetry range from the last slide will start coming in slowly, took under a minute for the first data point to come in my instance.

We can add some very arbitrary anomaly detection using the Event measurements too. Check out the next slide.



# New Measurement

1. New Measurement.
2. Event
3. Add the event fields
  - Display Name Rare Drop
  - Field Name raredrop
  - Default Severity Information
  - \*Note other severity options include Warning and Error.
4. Save

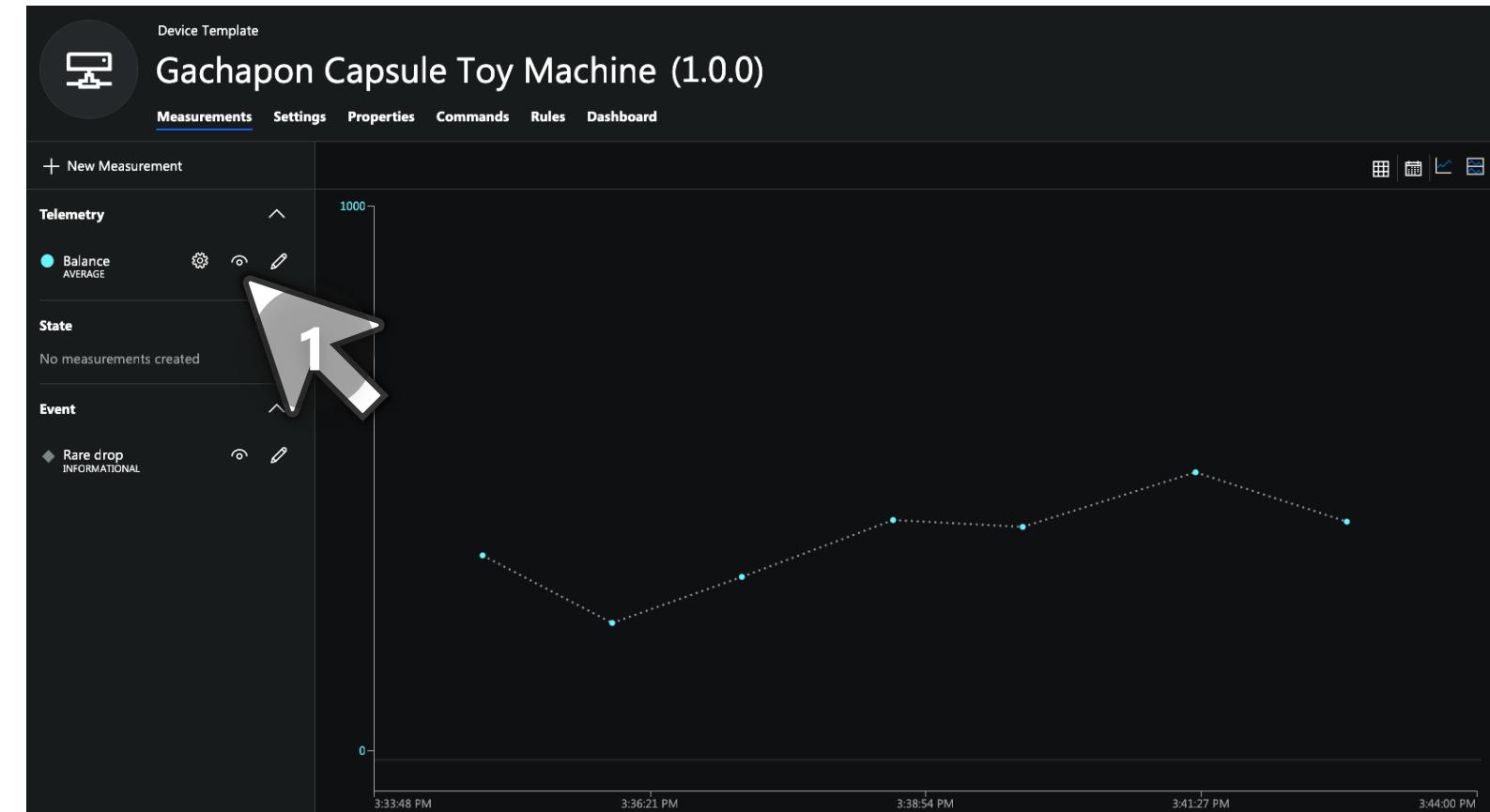
The image consists of four panels illustrating the process of creating a new measurement:

- Step 1:** A screenshot of the "New Measurement" interface. It shows three categories: "Telemetry" (selected), "State", and "Event". Each category has a description and a small visual representation. A large mouse cursor icon with the number "1" is positioned over the "Telemetry" section.
- Step 2:** A screenshot of the "Event" section. It shows a list of event types represented by diamond icons. A large mouse cursor icon with the number "2" is positioned over the second diamond icon.
- Step 3:** A screenshot of the "Create Event" dialog. It includes fields for "Display Name" (set to "Rare drop"), "Field Name" (set to "raredrop"), and "Default Severity" (set to "Information"). A large mouse cursor icon with the number "3" is positioned over the "Field Name" field.
- Step 4:** A screenshot of the "Save" button in the dialog. A large mouse cursor icon with the number "4" is positioned over the "Save" button.

# Event Measurement

You'll be navigated back to the main dashboard for your device. It will take some time for the randomly generated event date to come in... took a while for this, about 3mins to get my first error

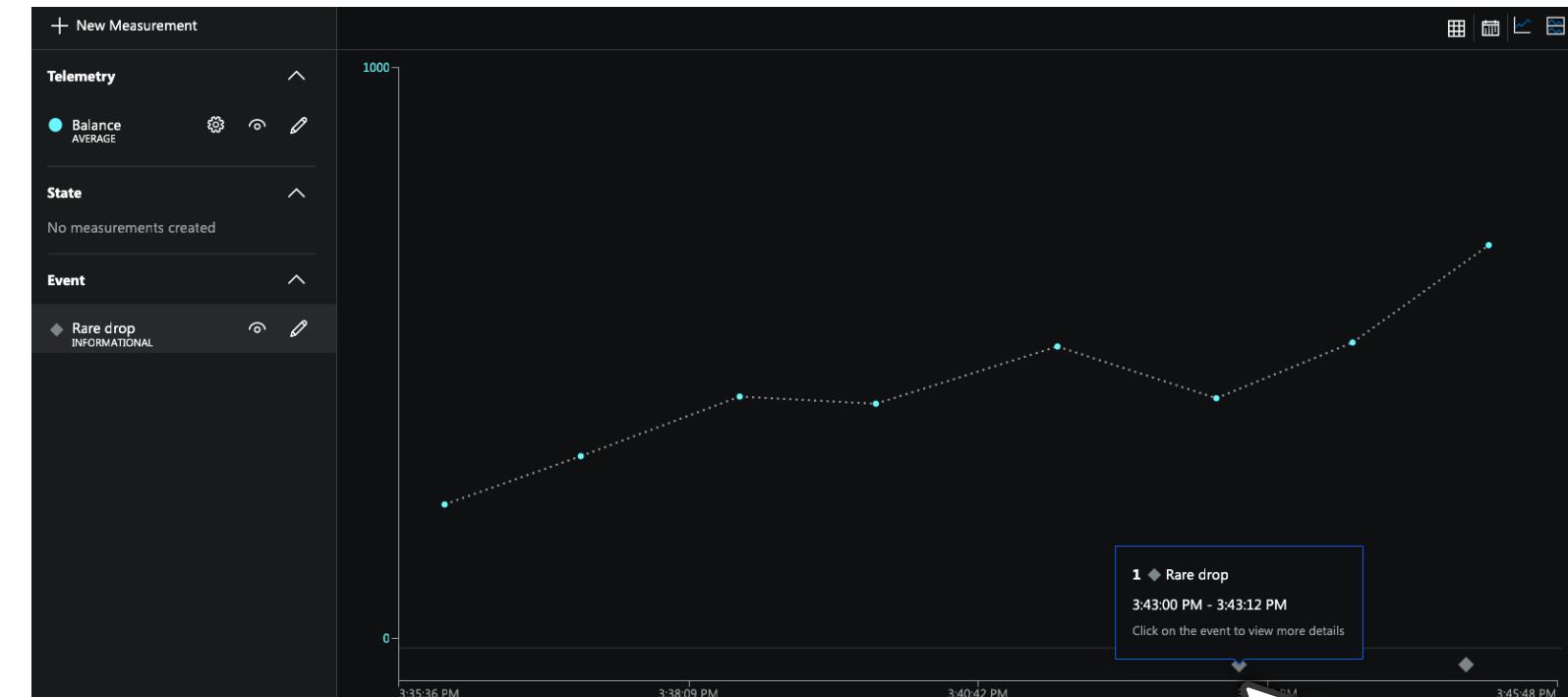
1. You can select the eyecon to hide the telemetry data and focus on the error data that's soon to come in.



# Error Data

Random sample event measurements for the anomalies should start coming in.

1. You can hover over the error diamond to see quick details or select it to open up a more detailed pop up window.



# State Measurement

State is the last measurement we'll add. In my example I need state tracking to understand my stock state.

1. New Measurement.
2. Select State.
3. Add our basic State data
  - Display Name Stock.
  - Field Name stock
4. Click + to add our first state value.
5. Add the High stock state fields
  - Value 1
  - Display label High Stock
6. Click + to add our second state value.
7. Add the Low stock state fields
  - Value 0
  - Display label Low Stock
8. Save.

The screenshots illustrate the steps to create a new measurement:

- Screenshot 1:** Shows the 'New Measurement' screen with three categories: Telemetry, State, and Event. A large mouse cursor icon with the number '1' is positioned over the 'State' category.
- Screenshot 2:** Shows the 'State' category expanded. It includes a description of 'Time series data from the device (e.g. Temperature)' with a line graph, and a sub-section for 'Event' with a description of 'Intermittent signal from the device (e.g. Alarm)' and four diamond icons. A large mouse cursor icon with the number '2' is positioned over the 'Event' section.
- Screenshot 3:** Shows the 'Create' dialog for a new measurement. The 'Display Name' field is set to 'Stock' and the 'Field Name' field is set to 'stock'. The 'Values' section contains two entries: 'High Stock' with a value of '1' and a color of black, and 'Low Stock' with a value of '0' and a color of yellow. A large mouse cursor icon with the number '3' is positioned over the 'Low Stock' entry. Another large mouse cursor icon with the number '4' is positioned over the 'High Stock' entry. A third large mouse cursor icon with the number '5' is positioned over the 'Values' section.
- Screenshot 4:** Shows the 'Save' button in the top right corner of the dialog, with a large mouse cursor icon containing the number '6' positioned directly above it.

# Template complete

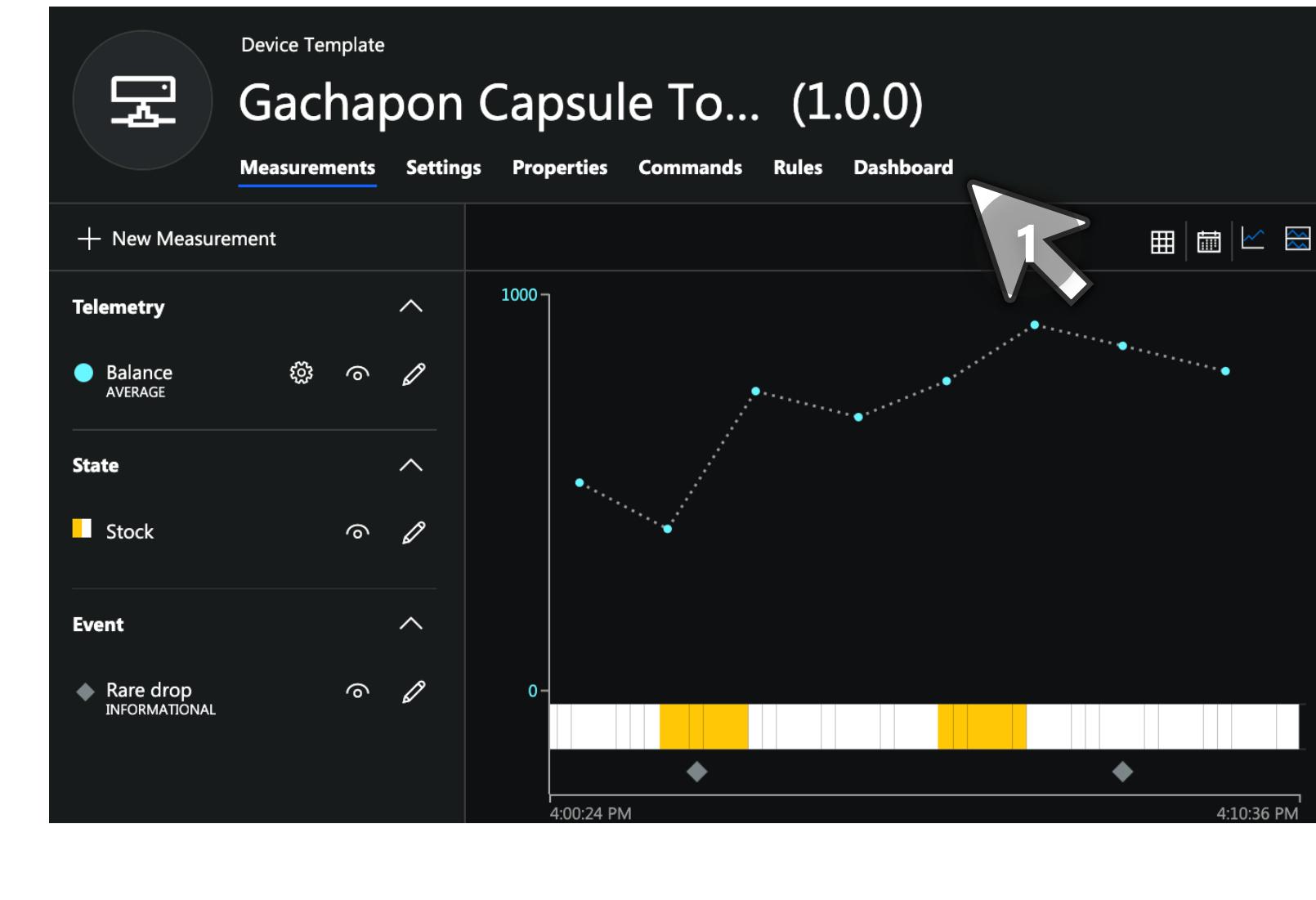
State should start flowing in! The state will randomly swap between normal and low power states for our air toy machine.

Initial build setup complete!



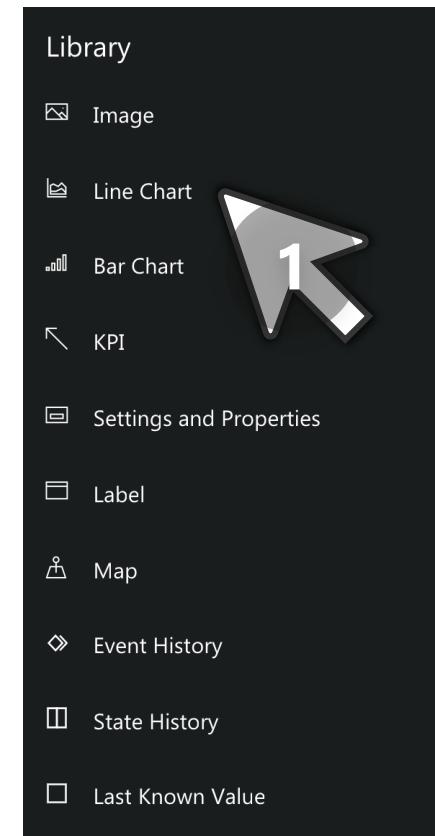
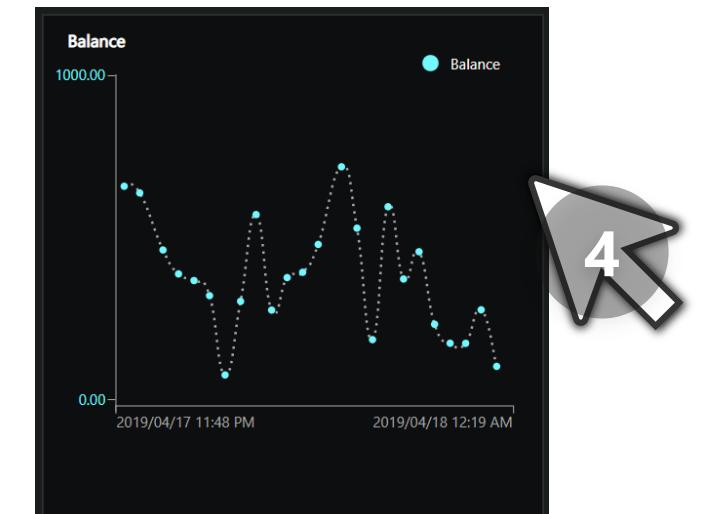
Now we can customize the operator's view of your application. When you make a change to the application as a builder, you can preview the operator's view in the Microsoft Azure IoT Central application.

1. Select Dashboard



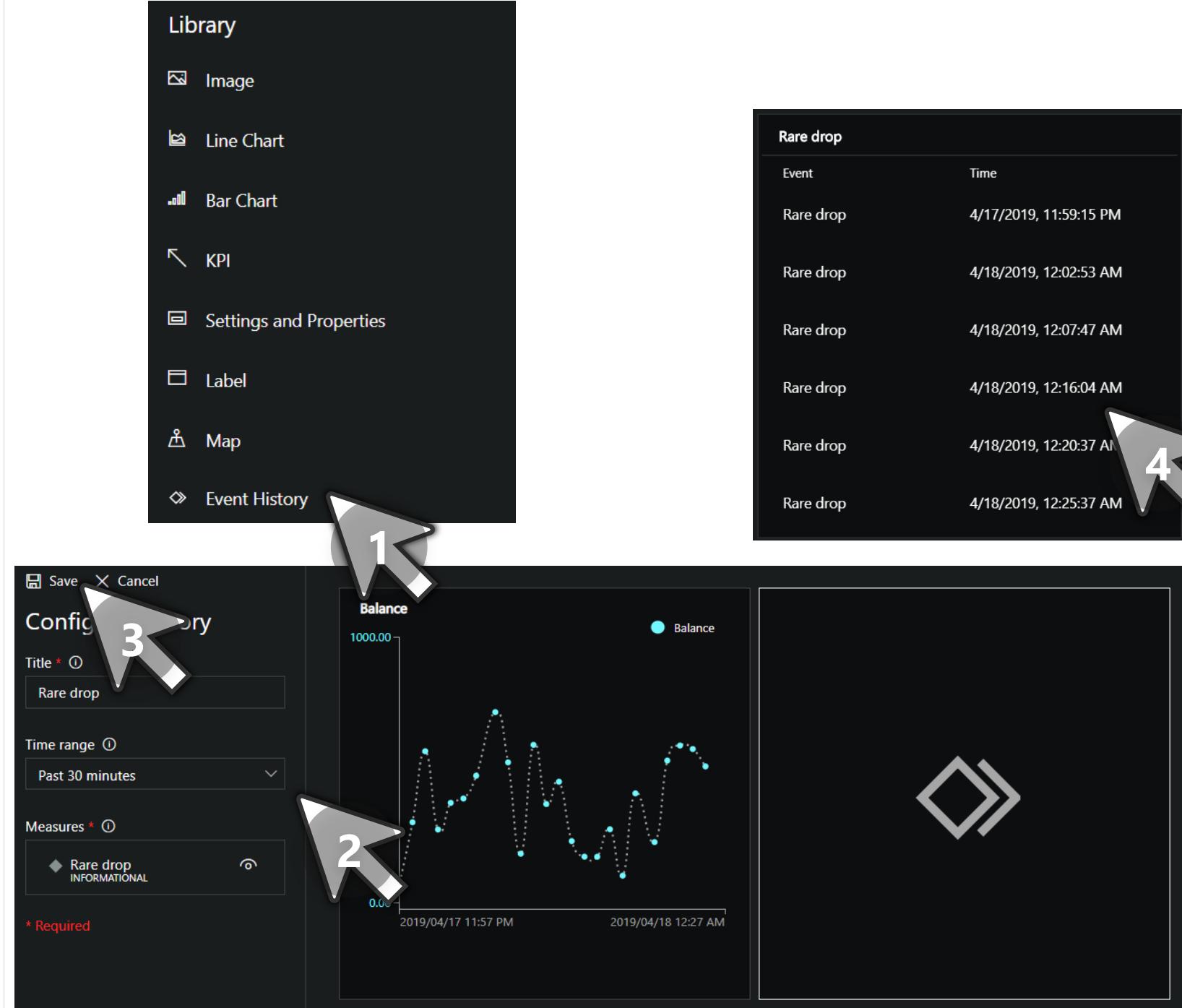
# Operator view

1. Select Line Cart
2. Completed the configure chart form with similar information to this:
  - Title Balance
  - Time Range Past 30 minutes
  - Measures Balance AVERAGE  
(choose visibility  icon in the measurement box)
3. Save
4. Once saved you save you'll soon see your temperature line chart on your dashboard

A screenshot of a configuration dialog box titled 'Configure Chart'. It shows a preview of a line chart with the title 'Balance'. The dialog includes fields for 'Title' (set to 'Balance'), 'Show Legend' (on), 'Show X-axis' (on), 'Show Y-axis' (on), 'Time range' (set to 'Past 30 minutes'), and 'Measures' (set to 'Balance AVERAGE'). A large grey mouse cursor with the number '3' is positioned over the 'Save' button at the top right. Another grey mouse cursor with the number '2' is positioned over the 'Configure Chart' title bar.

# Event history

1. Select Event History
2. Complete the configure history form with similar information to:
  - Title Rare drop
  - Time Range Past 30 minutes
  - Measures Rare drop INFORMATIONAL (choose the visibility  icon next to Fan Motor Error)
3. Save
4. Rare drop data should now appear on the dashboard



The image shows a user interface for configuring and viewing event history data.

**Library:**

- Image
- Line Chart
- Bar Chart
- KPI
- Settings and Properties
- Label
- Map
- Event History

A large mouse cursor arrow labeled **1** points to the "Event History" option in the library.

**Configure History:**

Form fields include:

- Title: Rare drop
- Time range: Past 30 minutes
- Measures: Rare drop INFORMATIONAL

A small note at the bottom states: \* Required.

A large mouse cursor arrow labeled **3** points to the "Save" button in the top left corner of the configuration dialog.

**Dashboard View:**

A line chart titled "Balance" displays fluctuating data over time, from 2019/04/17 11:57 PM to 2019/04/18 12:27 AM. The chart has a blue line and a blue dot labeled "Balance". A large mouse cursor arrow labeled **2** points to the chart area.

**Event History Table:**

Event	Time
Rare drop	4/17/2019, 11:59:15 PM
Rare drop	4/18/2019, 12:02:53 AM
Rare drop	4/18/2019, 12:07:47 AM
Rare drop	4/18/2019, 12:16:04 AM
Rare drop	4/18/2019, 12:20:37 AM
Rare drop	4/18/2019, 12:25:37 AM

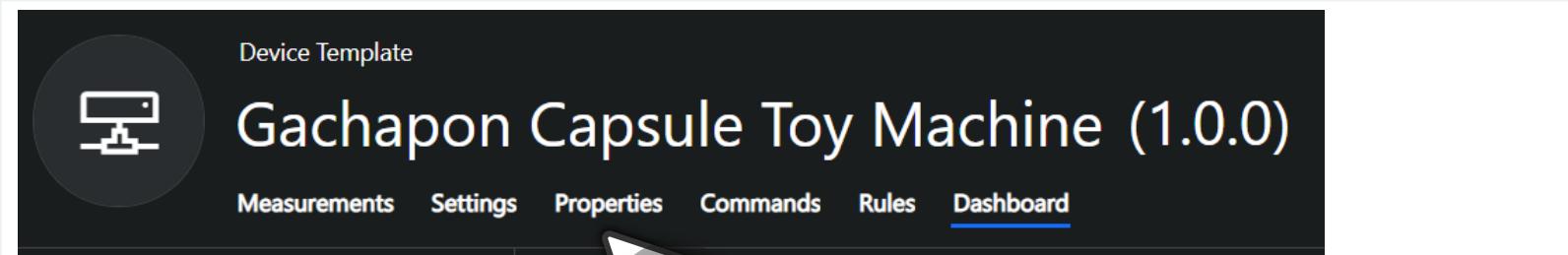
A large mouse cursor arrow labeled **4** points to the bottom right corner of the dashboard area.

# Properties

## 1. Select properties

Properties store information about your device. They can be editable properties or read-only device properties reported by the device that cannot be changed such as the device serial number and firmware version. We'll be adding these 4 properties you see here.

## 2. Select Location from the library



Device Template

## Gachapon Capsule Toy Machine (1.0.0)

Measurements Settings Properties Commands Rules Dashboard



1



Device Template

## Gachapon Capsule Toy Machine (1.0.0)

Measurements Settings Properties Commands Rules Dashboard

Library
12 Number
Text
Date
Toggle
Device Property
Label
Location

Properties are device metadata such as a customer address. [Learn more...](#)



2

# Location property

This is the first editable field we'll create.

1. Fill out the location fields
  - Display Name Location
  - Field Name location
  - Initial Value Tokyo Minato, Kanto
  - Description Device location
2. Save

The screenshot shows a dark-themed configuration dialog titled "Config Location". The dialog has a "Save" button at the top left and a "Cancel" button at the top right. The main area contains the following fields:

- Display Name \***: A text input field containing "Location". A large gray arrow points to this field with the number "2".
- Field Name \***: A text input field containing "location". A large gray arrow points to this field with the number "1".
- Initial Value**: A dropdown menu currently set to "Tokyo Minato, Kanto".
- Required**: A toggle switch set to "Off".
- Description**: A text input field containing "Device location".

A large gray arrow points from the bottom left towards the "Required" section.

# Date property

This is the second editable field we'll create.

1. Select Date
2. Fill out the Date fields
  - Display Name Last Service Date
  - Field Name lastServiceDate
  - Initial Value 4/17/2019
  - Description Last serviced
3. Save

The screenshot shows the Microsoft IoT Central setup interface. On the left, a sidebar titled 'Library' lists various field types: Number (12), Text, Date (selected), Toggle, Device Property, Label, and Location. A large mouse cursor icon with the number '1' is positioned over the 'Date' item. To the right, a configuration form is displayed for a 'Configure Date' field. The form includes:

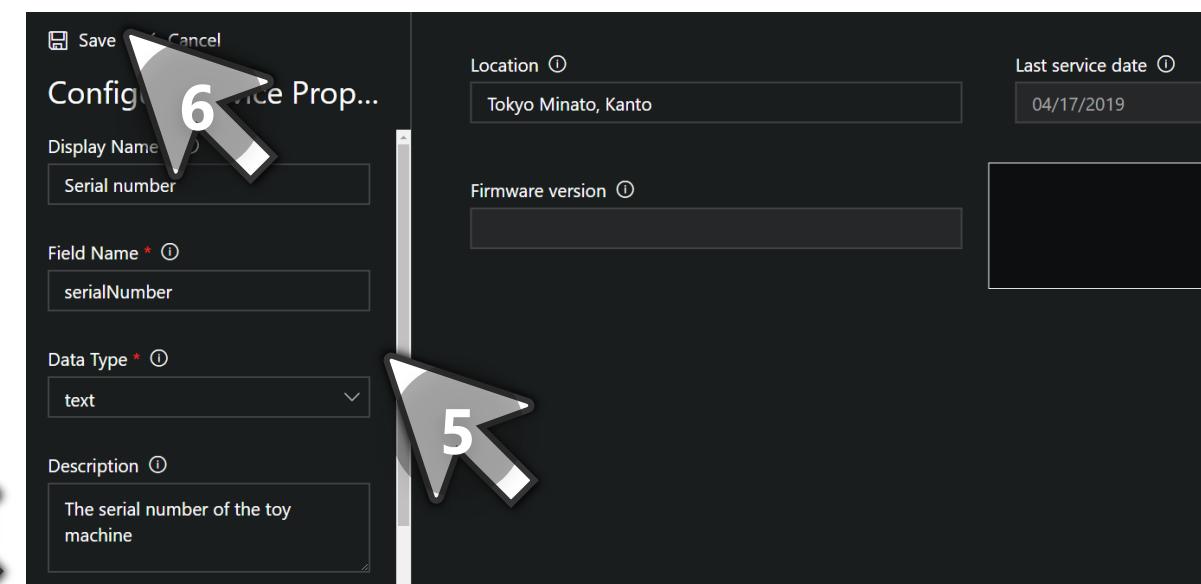
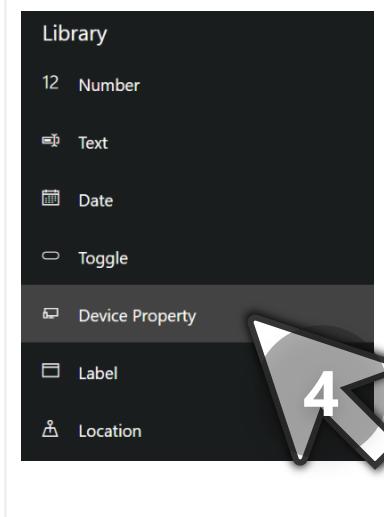
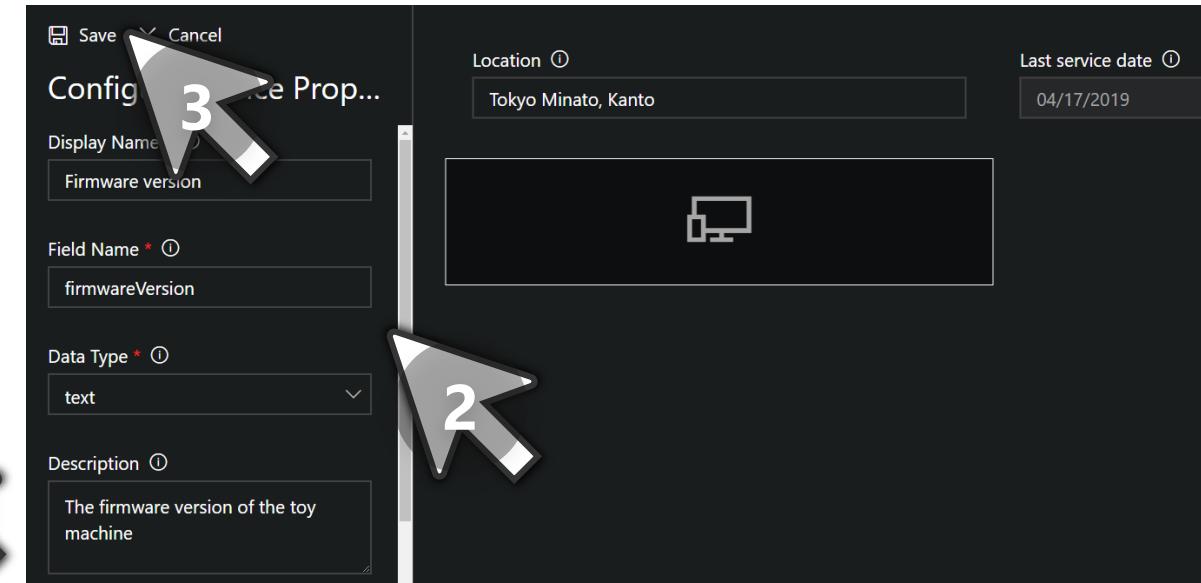
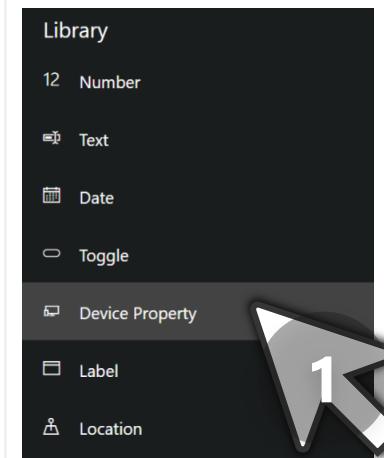
- Display Name \***: Last service date
- Field Name \***: lastServiceDate
- Show Time**: Off (switch is off)
- Initial Value \***: 04/17/2019 (with a calendar icon)
- Required**: Off (switch is off)
- Description**: Last serviced

A large mouse cursor icon with the number '3' is positioned over the 'Save' button at the top right of the form. Another large mouse cursor icon with the number '2' is positioned over the 'Initial Value' input field. The overall theme is dark with light-colored text and UI elements.

# Device properties

Create two device properties, these are both reads only

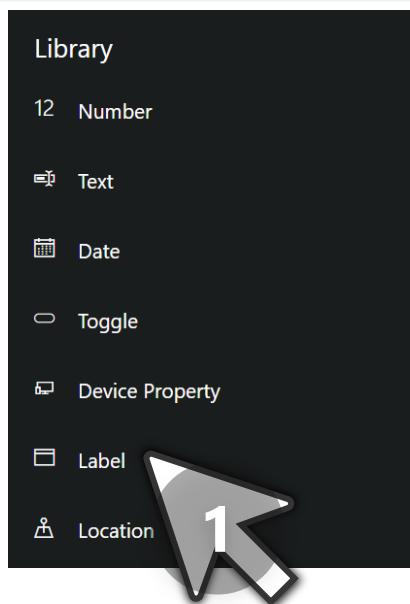
1. Select Device Property from the Library
2. Add similar Firmware version data
  - Display Name Firmware version
  - Field Name firmwareVersion
  - Data Type text
  - Description The firmware version of the toy machine
3. Save 
4. Select Device Property from the list
5. Add Serial Number data
  - Display Name Serial number
  - Field Name serialNumber
  - Data Type text
  - Description The serial number of the toy machine
6. Save 



# Label

This Service Information that we'll add along with its properties will be important in the data integration we'll create later between Dynamics 365 and Azure IoT Central.

1. Select Label
2. Complete the configure label form with
  - Text Service Information
  - Text Size large
3. Save

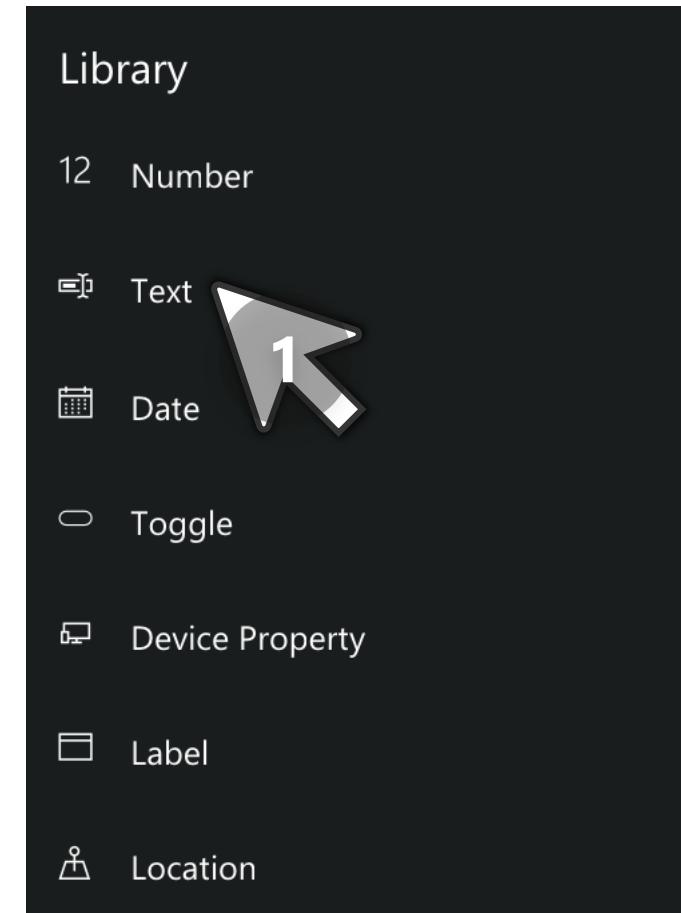
A screenshot of the 'Configure Label' configuration form. The form includes fields for 'Text' (containing 'Service Information'), 'Text Size' (set to 'large'), 'Location' (Tokyo Minato, Kanto), 'Firmware version' (Itaque consectetur tenetur.), 'Last service date' (04/17/2019), and 'Serial number' (Cum consequatur quaerat nisi qui aut est occaecati rerum). A large number '3' is overlaid on the 'Save' button, and a large number '2' is overlaid on the 'Text Size' dropdown. A small number '1' is overlaid on the 'Text' input field.

# Work order fields

Now we'll add 6 data points for Service Information that we can capture from work order information sent from Dynamics 365 to IoT Central. These are our data columns.

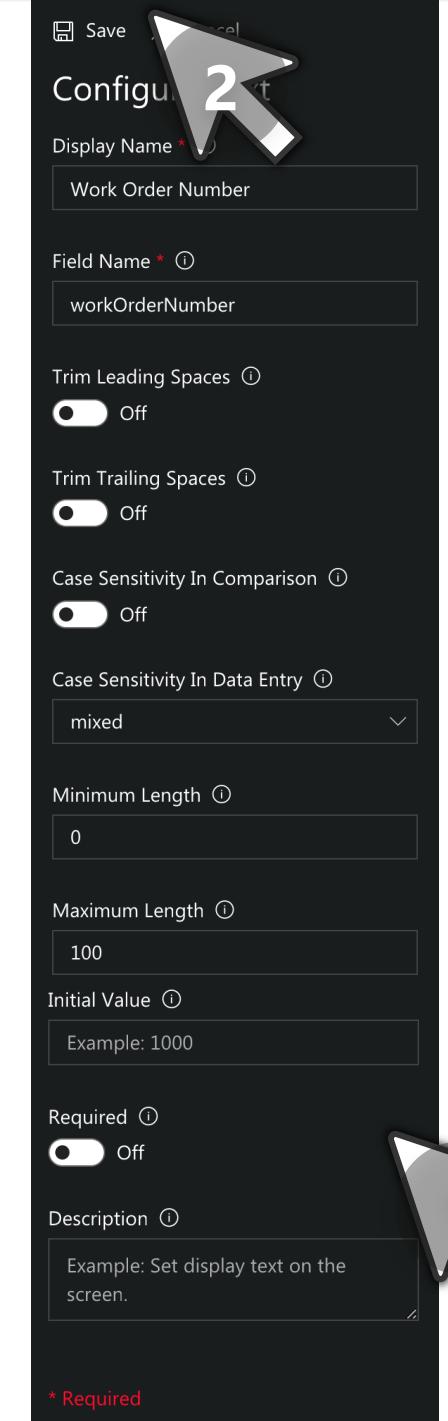
Our goal here is to ingest Work Order data <https://docs.microsoft.com/en-us/dynamics365/customer-engagement/field-service/cfs-iot-central-work-orders>

1. Select Text



# Work Order field

1. Complete the configure text form with the information I have in the entire column pointed to by arrow 1
2. Save
3. Drag the Work Order Number text under the Service Information label
4. Select Text
5. Complete the configure text form with the information I have in the entire column pointed to by arrow 5
6. Save



Configure Text

Display Name \*

Field Name \*

Trim Leading Spaces  Off

Trim Trailing Spaces  Off

Case Sensitivity In Comparison  Off

Case Sensitivity In Data Entry

Minimum Length

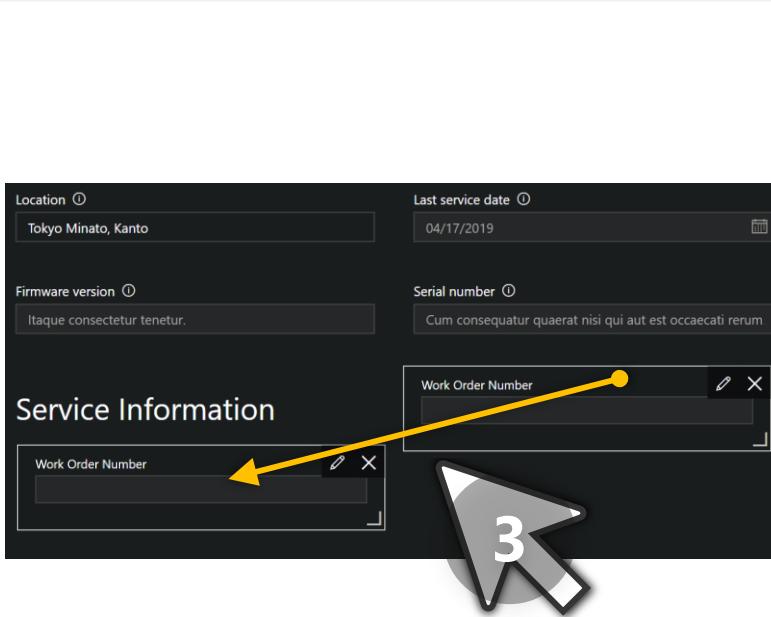
Maximum Length

Initial Value

Required  Off

Description

\* Required



Service Information

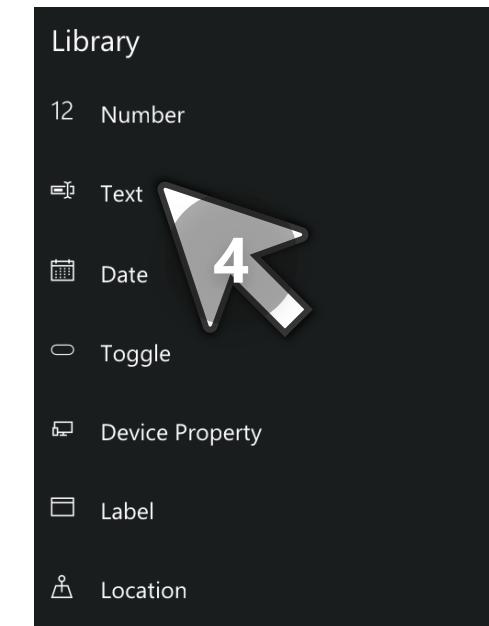
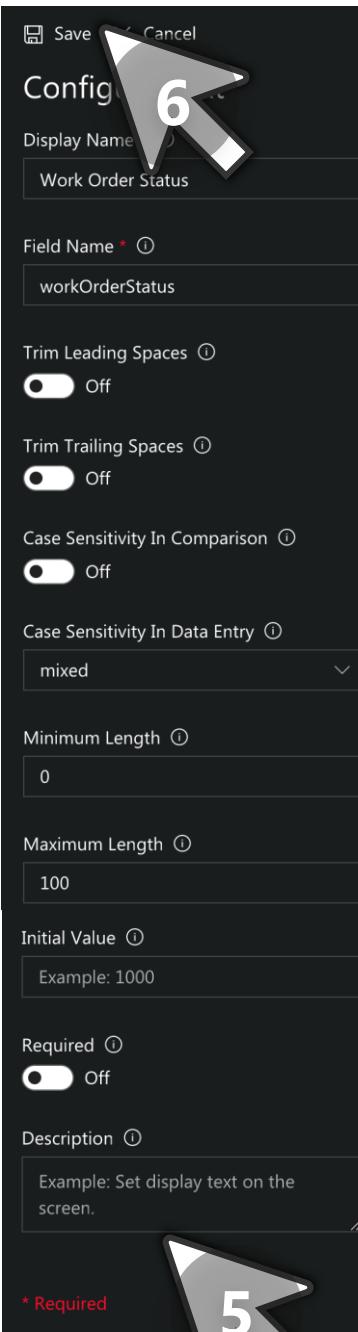
Location

Last service date

Firmware version

Serial number

Work Order Number

Configure Text

Display Name

Field Name

Trim Leading Spaces  Off

Trim Trailing Spaces  Off

Case Sensitivity In Comparison  Off

Case Sensitivity In Data Entry

Minimum Length

Maximum Length

Initial Value

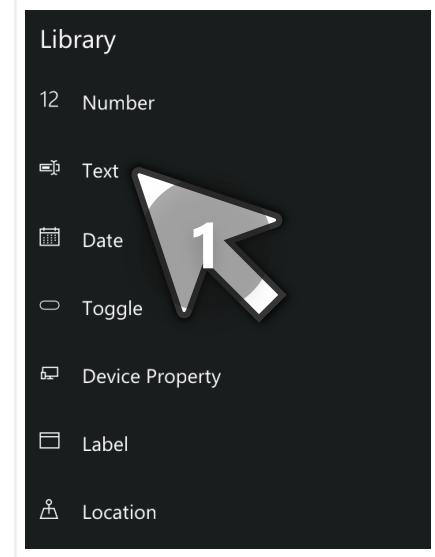
Required  Off

Description

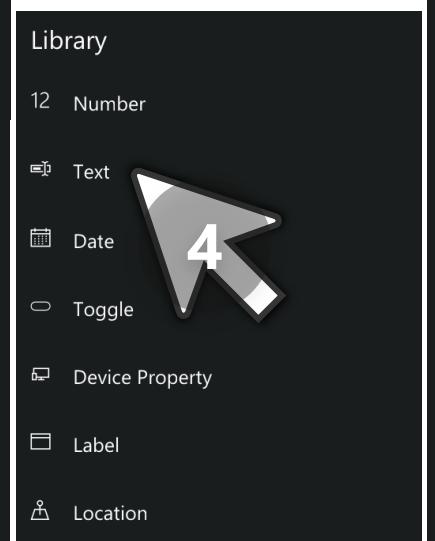
\* Required

# Incident field

1. Select text from the Library
2. Complete the configure text form with the information I have in the entire column pointed to by arrow 2
3. Save
4. Select text from the Library
5. Complete the configure text form with the information I have in the entire column pointed to by arrow 2
6. Save



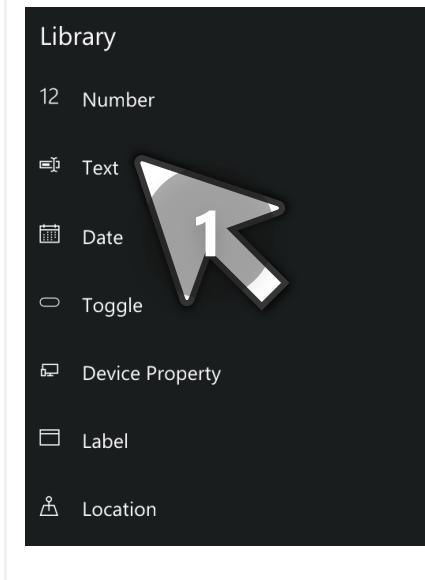
This screenshot shows the 'Configure Text' form for the first incident field. It includes fields for 'Display Name' (set to 'Incident Description'), 'Field Name' (set to 'incidentDescription'), and several configuration options like 'Trim Leading Spaces' (set to 'Off'). The 'Case Sensitivity In Data Entry' dropdown is set to 'mixed'. A large grey arrow labeled '3' points to the 'Field Name' field.



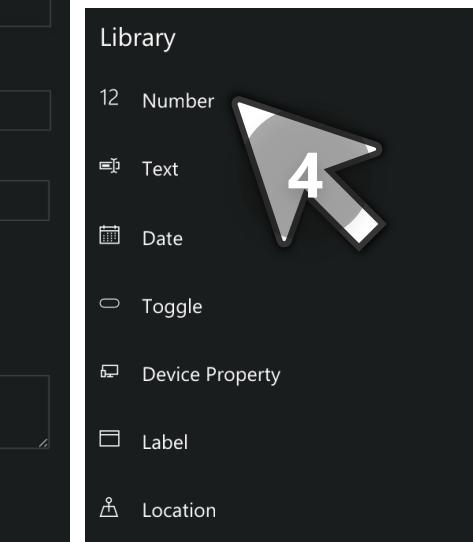
This screenshot shows the 'Configure Text' form for the second incident field. It has similar fields to the first one, including 'Display Name' (set to 'Work Order Owner Id'), 'Field Name' (set to 'workOrderOwnerId'), and configuration options. The 'Case Sensitivity In Data Entry' dropdown is set to 'mixed'. A large grey arrow labeled '5' points to the 'Field Name' field. Another arrow labeled '6' points to the 'Display Name' field.

# Service Times

1. Select Text from the Library
2. Complete the configure text form with the information I have in the entire column pointed to by arrow 2
3. Save
4. Select Number from the Library
5. Complete the configure text form with the information I have in the entire column pointed to by arrow 5
6. Save



This screenshot shows the 'Configure Text' form for 'Estimated Arrival Time'. It includes fields for 'Display Name' (set to 'Estimated Arrival Time'), 'Field Name' (set to 'estimatedArrivalTime'), and various configuration options like 'Trim Leading Spaces' (set to 'Off') and 'Case Sensitivity In Comparison' (set to 'Off'). A large white arrow labeled '3' points to the 'Display Name' field.



This screenshot shows the 'Configure Number' form for 'Estimated Service Duration'. It includes fields for 'Display Name' (set to 'Estimated Service Duration'), 'Field Name' (set to 'estimatedServiceDuration'), and various configuration options like 'Unit of Measure' (set to 'hours'), 'Number of Decimal Places' (set to '0'), and 'Minimum Value' (set to '0'). A large white arrow labeled '6' points to the 'Display Name' field.

# All service information

Lot's of data entry later and now this! We have all the important fields we need to ingest data from Dynamics 365.

Your completed Service Information in your properties page should now look like this. No we need to add these properties to our dashboard.

1. Select Dashboard

Properties Commands Rules Dashboard

Location ⓘ Tokyo Minato, Kanto

Last service date ⓘ 04/17/2019

Firmware version ⓘ Ratione consequatur sint eum voluptatem voluptatem.

Serial number ⓘ Et fuga rerum magni enim voluptatem ut.

## Service Information

Work Order Number

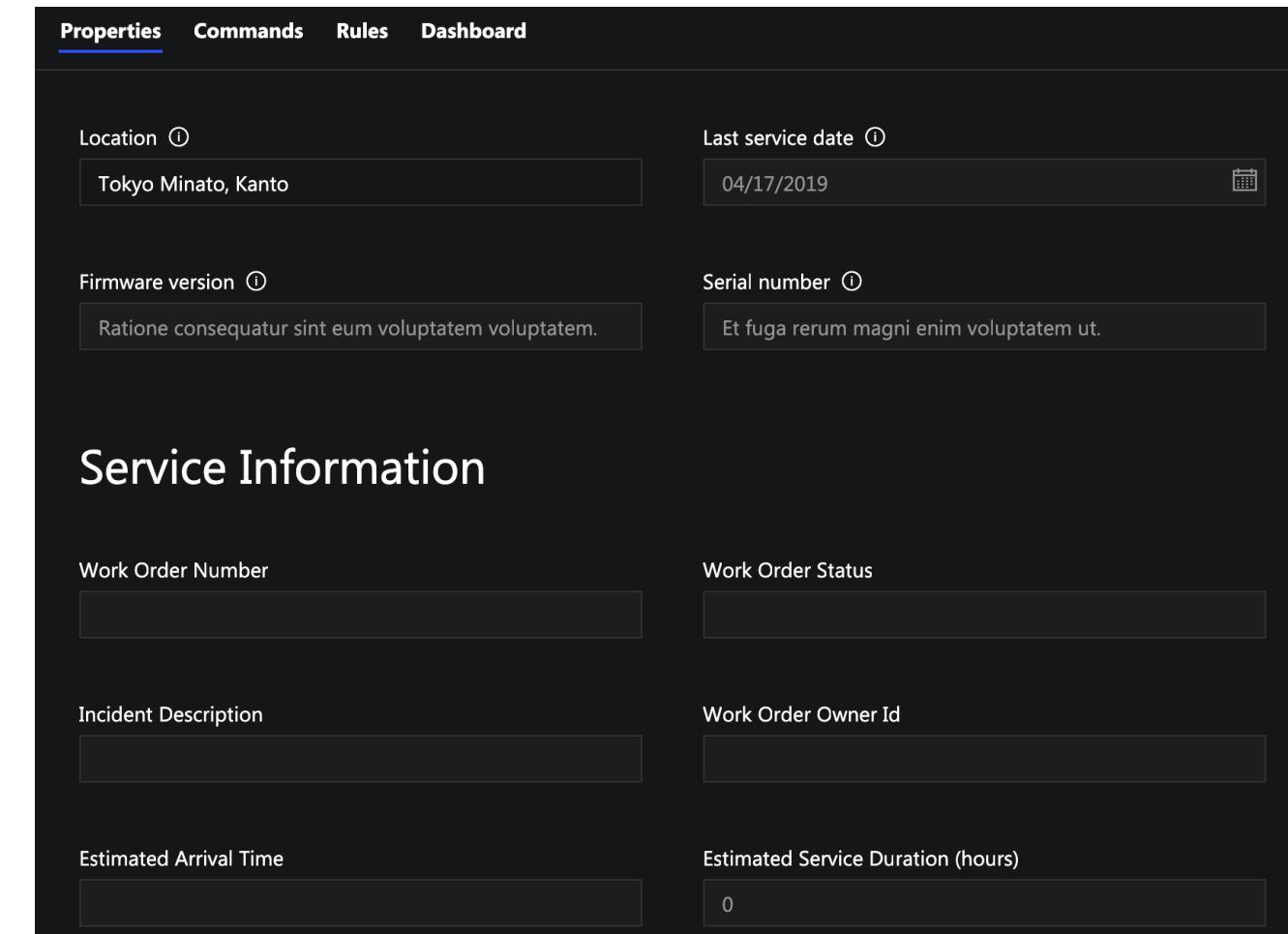
Work Order Status

Incident Description

Work Order Owner Id

Estimated Arrival Time

Estimated Service Duration (hours)

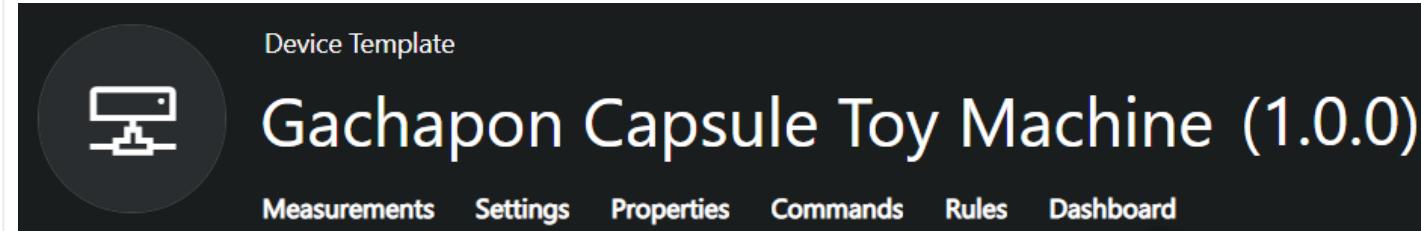


Device Template



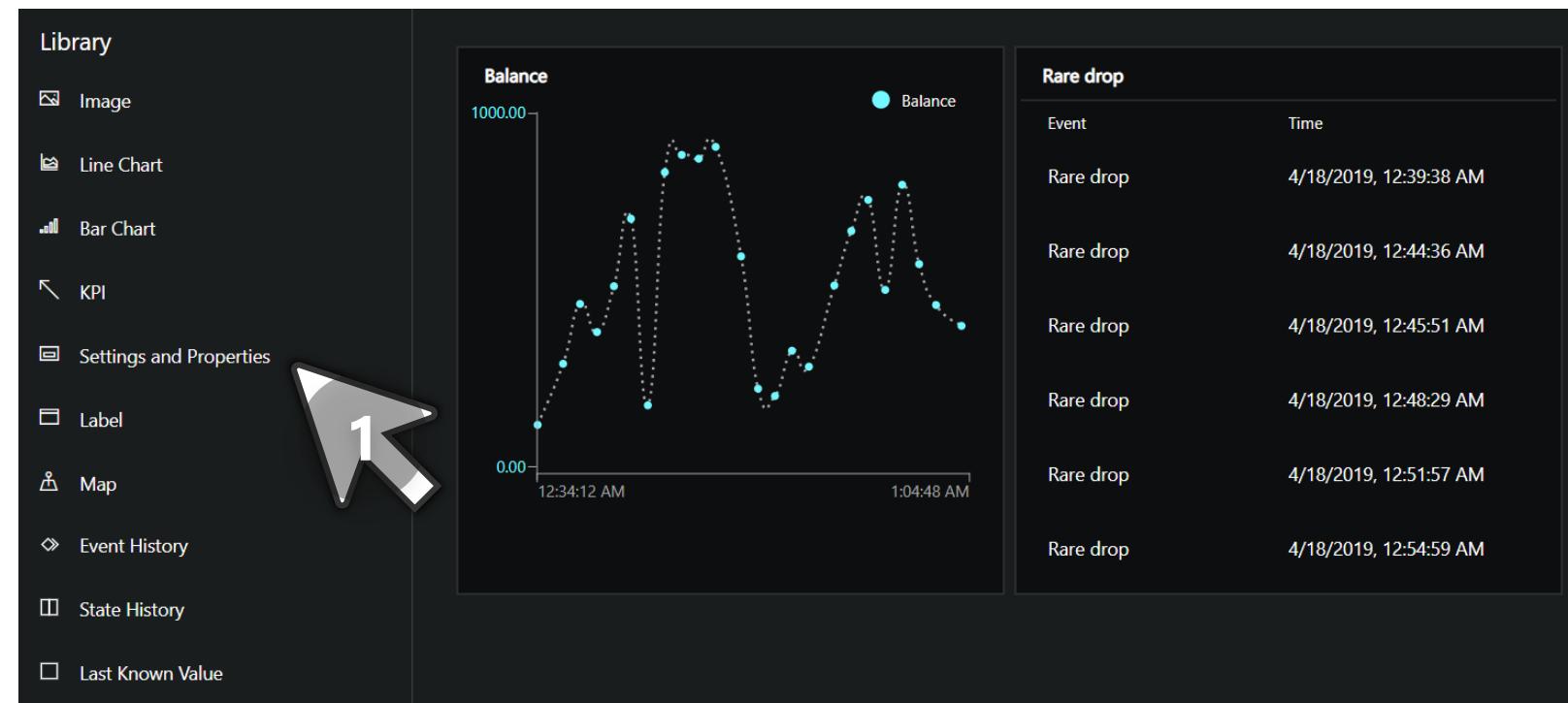
Gachapon Capsule Toy Machine (1.0.0)

Measurements Settings Properties Commands Rules Dashboard



# CFS IoT Central

## 1. Select Settings and Properties



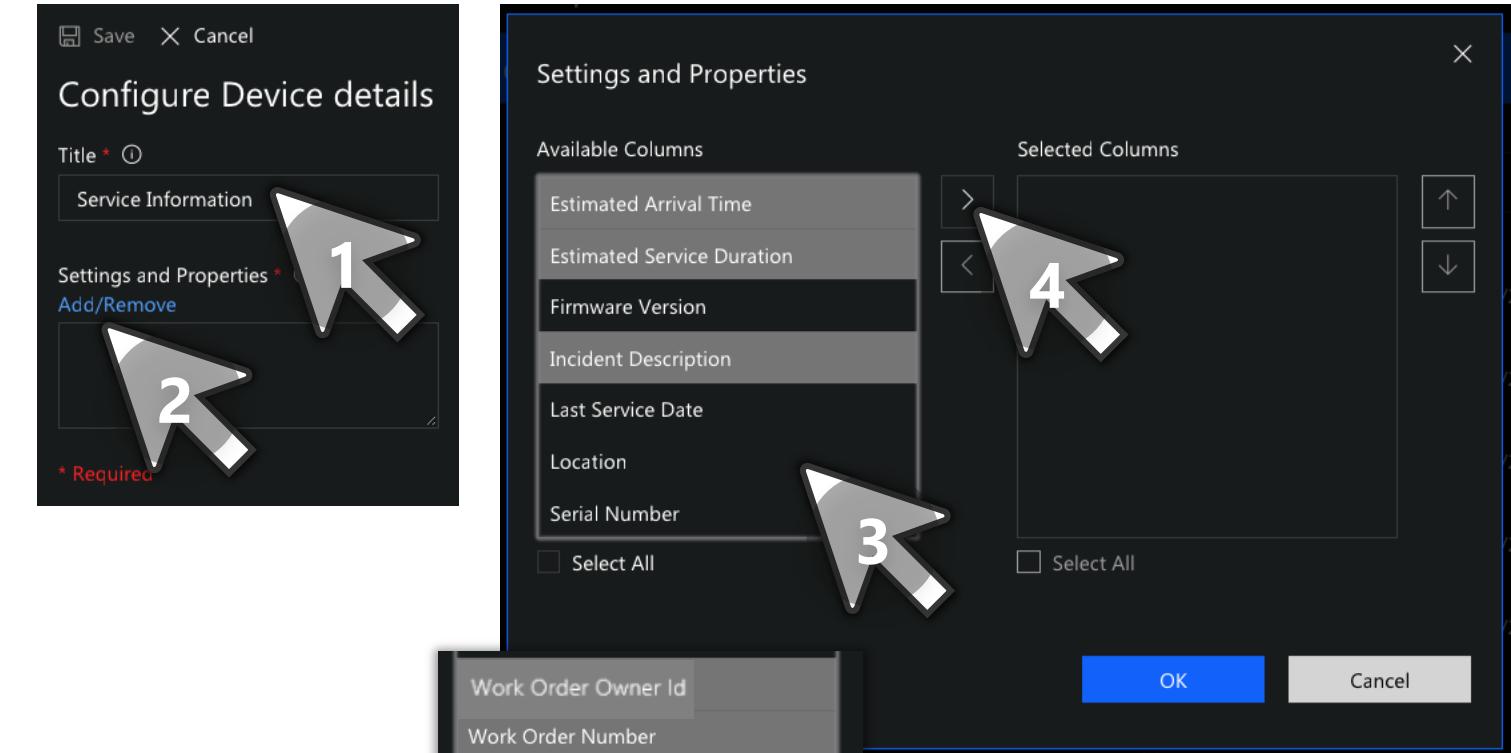
# CFS IoT Central

1. Add the Title Service Information
2. Select Add/Remove
3. This opens a window that lists all our available columns/properties that we can place on the dashboard. Select our Service Information Properties using Ctrl + Click:
  - Estimated Arrival Time
  - Estimated Service Duration
  - Incident Description
  - Technician Name
  - Work Order Number
  - Work Order Status



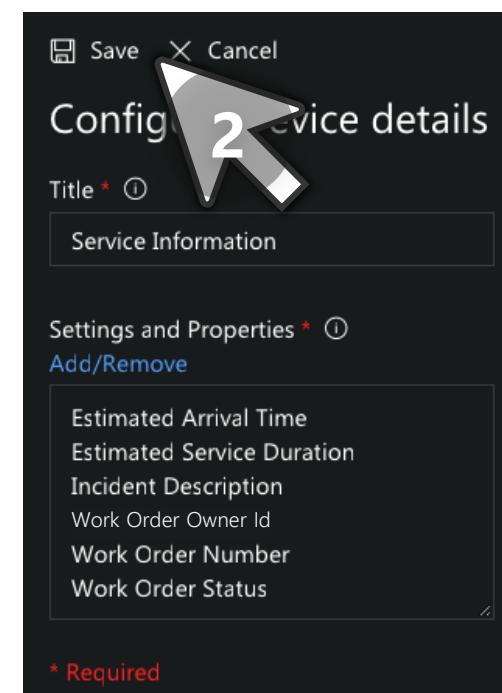
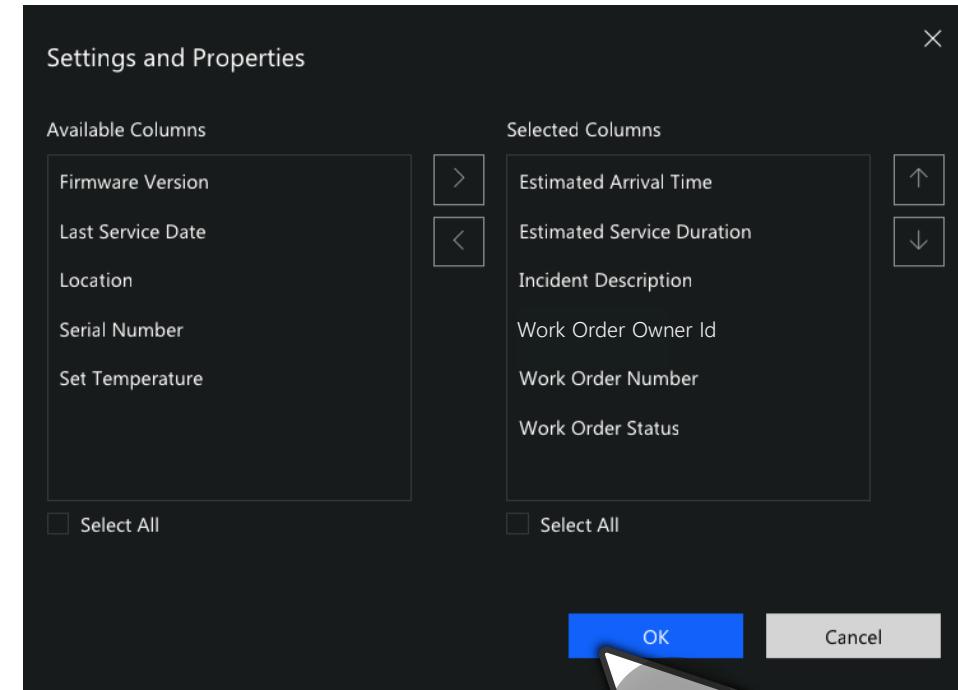
You may need to scroll down as I have to select each one

4. Select the Arrow > to move the selection to the selected columns



# CFS IoT Central

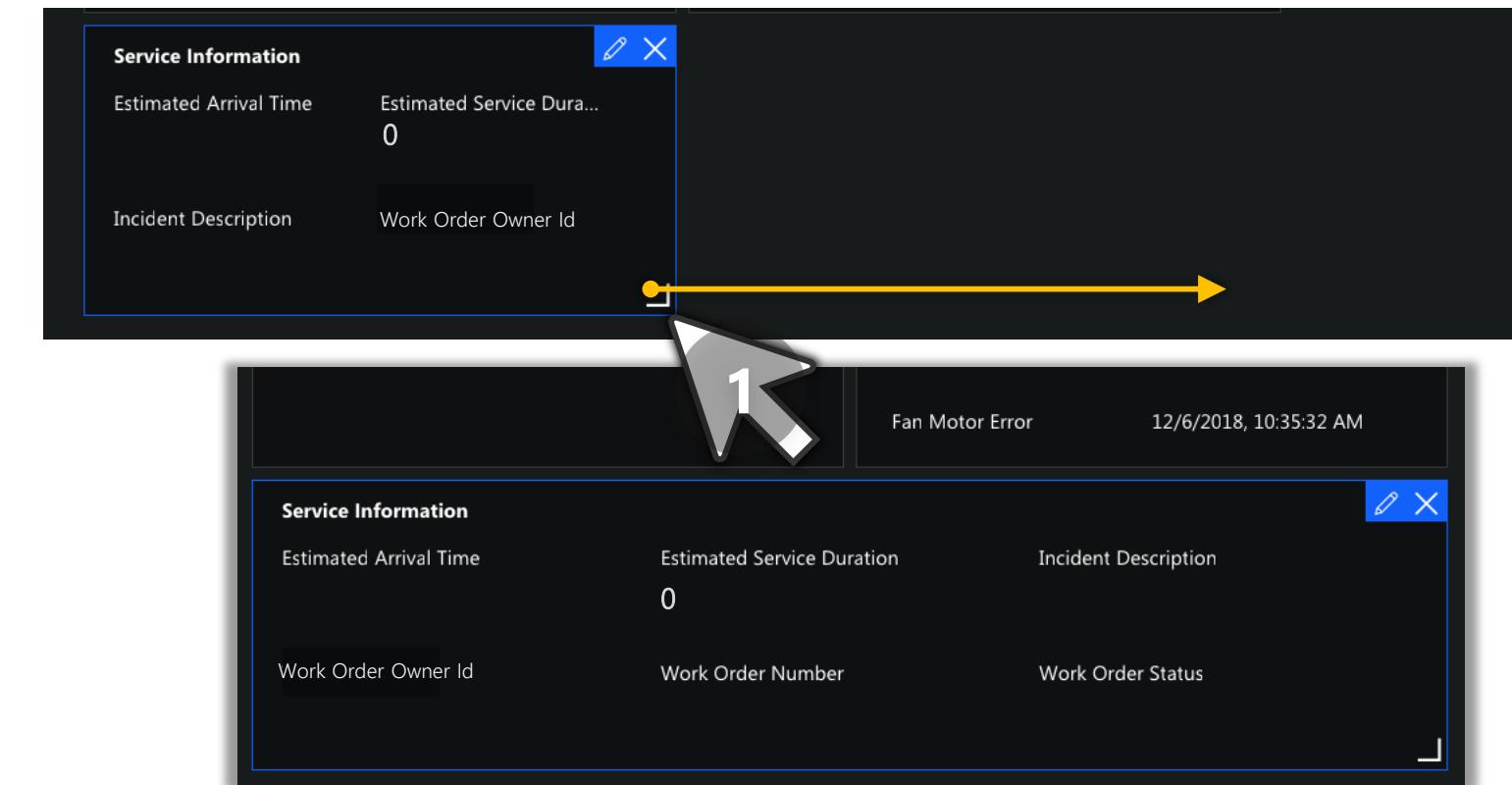
1. All of our required Service Information Columns should now be moved. Select OK
2. Save



# CFS IoT Central

Your complete Service Information properties should now be on your Dashboard

1. Drag the Service Information window to reveal all the properties



# CFS IoT Central

Dashboard complete!



In a future exercise when we prepare a Flow from Dynamics 365 to capture this Work Order Service Information data and send it in IoT Central. We'll have a component in our operator dashboard that can ingest our data and display it in real time.

Device Template

## Gachapon Capsule Toy Machine (1.0.0)

Measurements Settings Properties Commands Rules Dashboard

Library

- Image
- Line Chart
- Bar Chart
- KPI
- Settings and Properties
- Label
- Map
- Event History
- State History
- Last Known Value

**Balance**

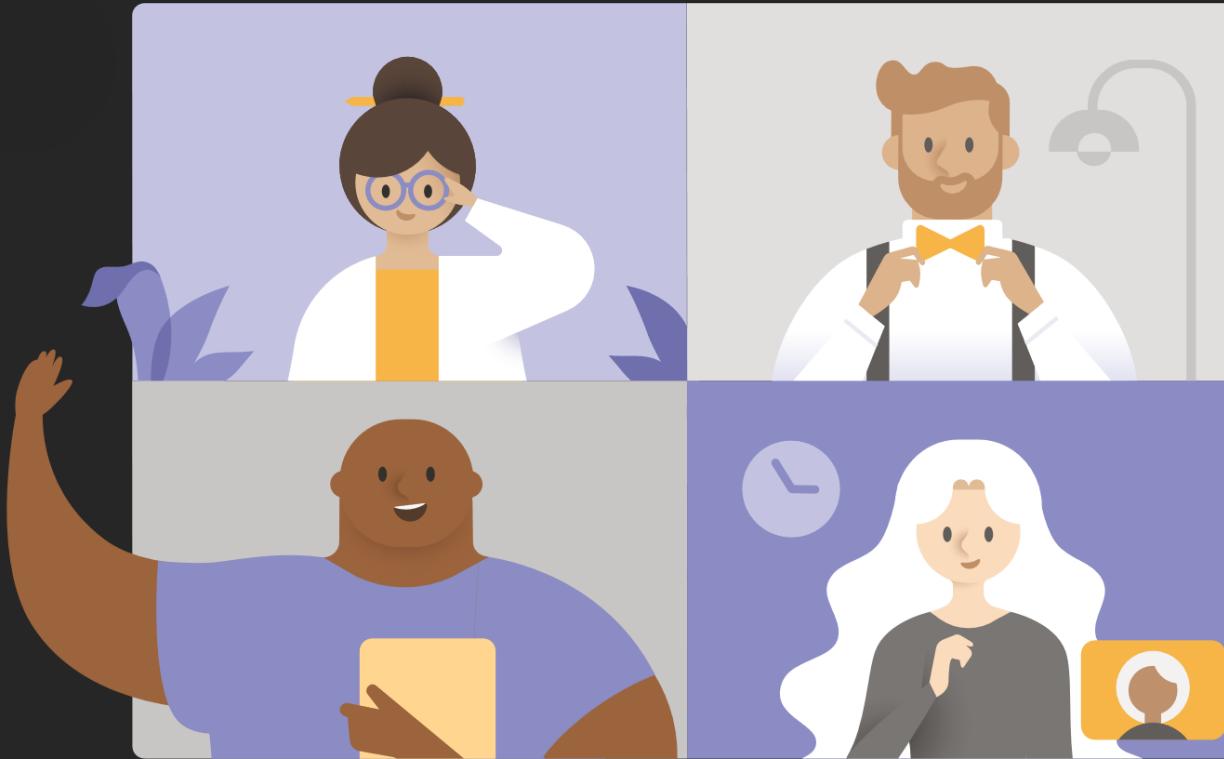
4:27:00 PM 4:57:36 PM

**Rare drop**

Event	Time
Rare drop	4/18/2019, 4:28:57 PM
Rare drop	4/18/2019, 4:35:04 PM
Rare drop	4/18/2019, 4:38:54 PM
Rare drop	4/18/2019, 4:45:03 PM
Rare drop	4/18/2019, 4:50:41 PM

**Service Information**

Estimated Arrival Time	Estimated Service Duration	Incident Description
	0	
Work Order Owner Id	Work Order Number	Work Order Status



**Thank you**