Plug and Play Bootcamp
Hands on lab

Script – track 1: Device builder

1. Based upon a provided template, by leveraging the VS Code PnP plugin, author
   - A couple of interfaces
   - Your capability model integrating those interfaces
2. Register your capability model against the portal
3. Create the skeleton of your device agent
4. Complete the code in order to get the expected behavior
5. Deploy the PnP device agent
6. Create a new IoT Central App
7. Create a new device by importing the capability model
8. Build your dashboard
9. Simulate then connect your physic device
10. Check data flows into IoT Central
11. Change a property and verify the change has been applied
12. Execute a command and validate it has been processed.

Script – track 2: Solution developer, IoT Central

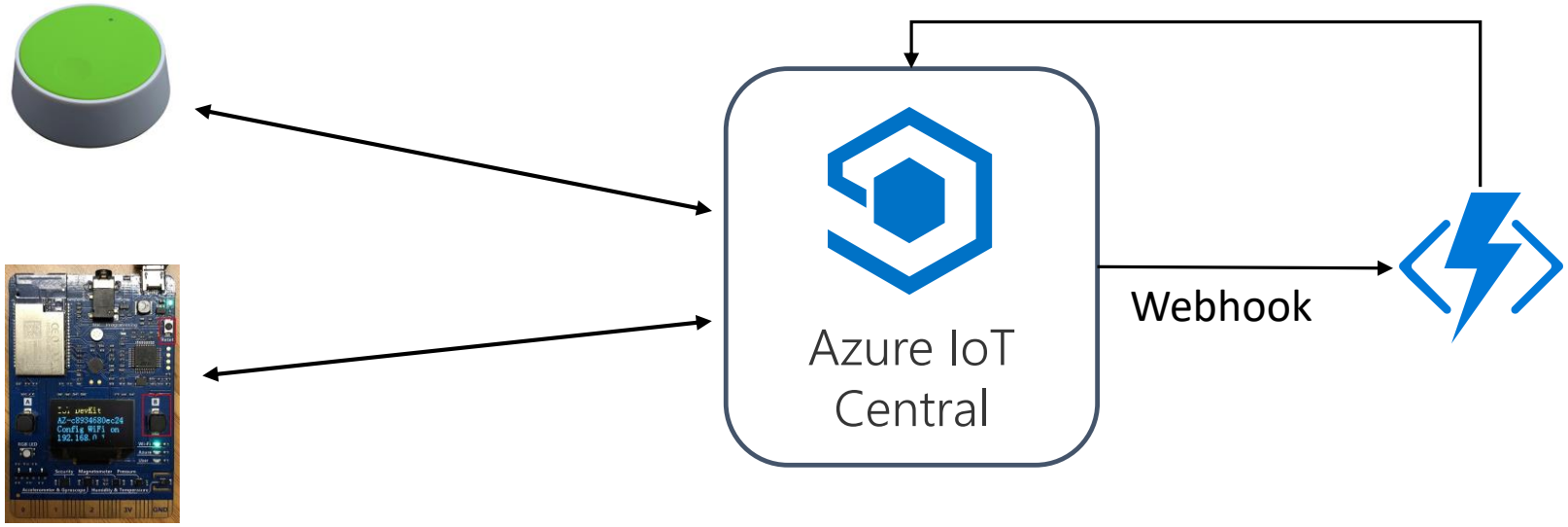
1. Go to the Azure IoT catalog https://catalog.azureiotsolutions.com/
2. Click on "all devices" then filter by IoT Plug and Play, pre-certified device
3. Select a plug and play device and have a look at its capability model/interfaces
4. Create a new IoT central app
5. Select the device from the catalog by creating a new device template leveraging the Mxchip one
6. Have a look at the capability model leveraging the IoT Central interface
7. Add a new **simulated** device in your app
8. Build your dashboard

Script – track 2: Solution developer, IoT Central

1. Add a new **real** device (simulated option off) to the solution using the MXChip board device template
2. Add a new dashboard page and customize it
3. Verify that command and properties can be changed, telemetry flows to IoT Central

https://github.com/Azure-Samples/mxchip-iot-devkit-pnp

Implementation



https://seeedjp.github.io/ReButton/
https://docs.microsoft.com/fr-fr/azure/iot-central/core/howto-trigger-azure-functions
https://docs.microsoft.com/en-us/azure/iot-central/preview/howto-create-webhooks

Script – track 2: Solution developer, IoT Hub

1. Using Azure IoT explorer, provision a new device
2. Using the C# (Node JS) service SDK, build one/multiple console applications
   - Getting the interesting information about Model and interface for a specific device
   - Displaying the received telemetry
   - Modifying a properties
   - Executing a command
3. Run each application and verify the outcomes.

https://github.com/Azure-Samples/azure-iot-samples-csharp/tree/master/digitaltwin/Samples/service