# IoT quickstart

Maggie Mei
Austin Blackstone

Applications Engineer / ARM

ARM mbed Connect / Shenzhen, China
December 5th, 2016

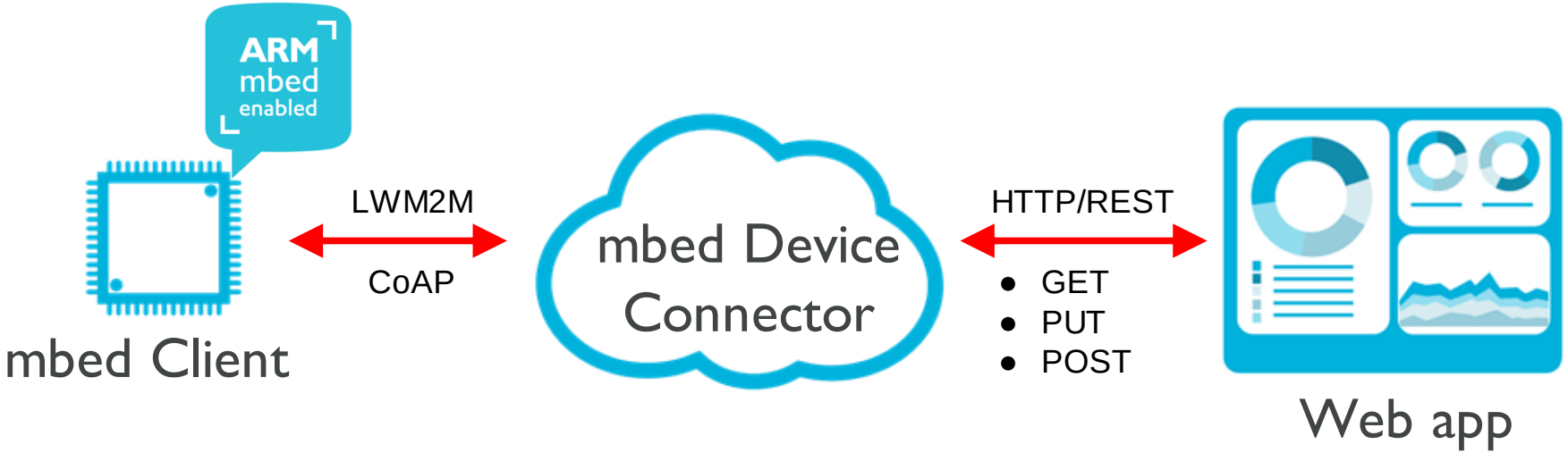
# Topics

- Overview
  - mbed URI format
  - LWM2M resources
  - REST API
- mbed Client endpoint example
- mbed Connector web app example
  - Node.js or python
- Challenges

# Prerequisites

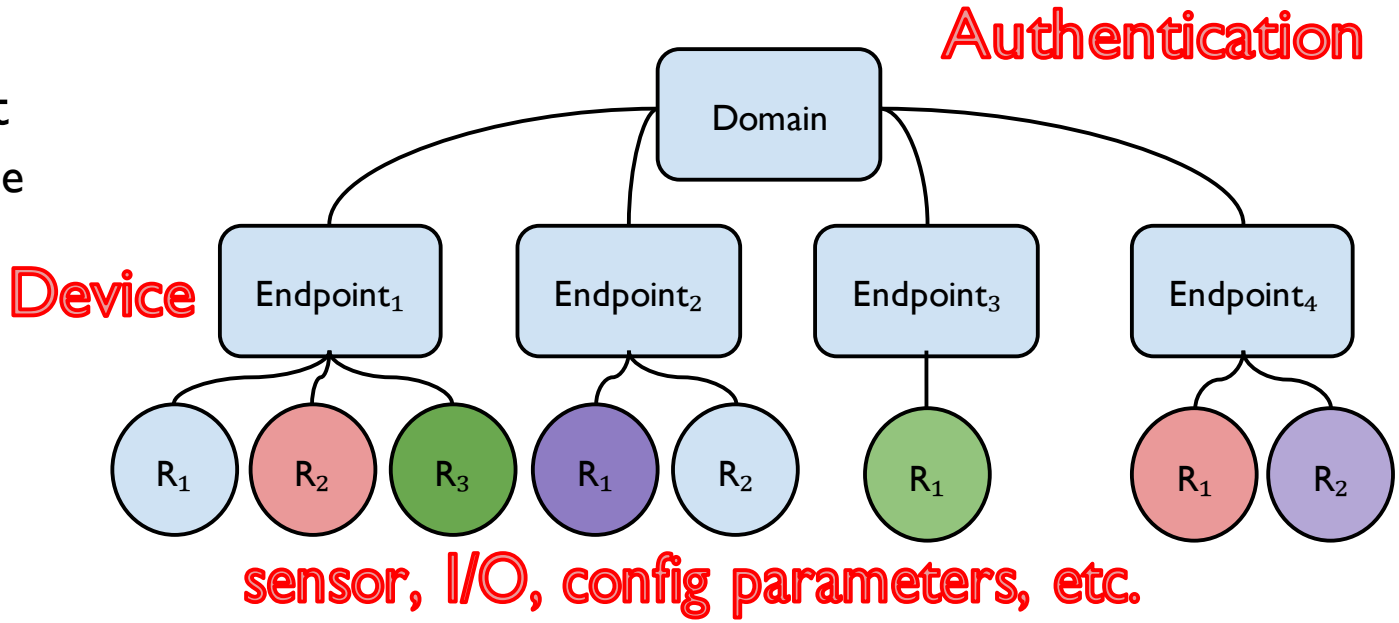
- Free mbed developer account
- Python 2.7.*
- Node.js
- Git
- mbed Windows serial driver
  - `https://developer.mbed.org/platforms/Nuvoton-NUC472/`
- Clone example code
  - `git clone https://github.com/armmbed/mbed-connector-api-python-quickstart.git`
  - `git clone https://github.com/armmbed/mbed-connector-api-node-quickstart.git`

# Using mbed Device Connector



# Data model

- Domain
  - Endpoint
    - Resource

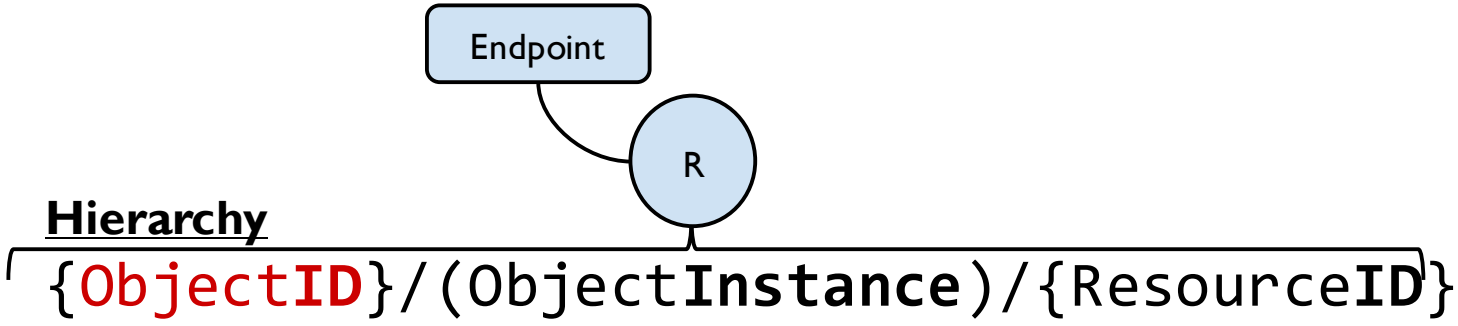



```
/endpoints/endpointName/{LWM2M Resource}
```



ARM

# Resource: LWM2M data model

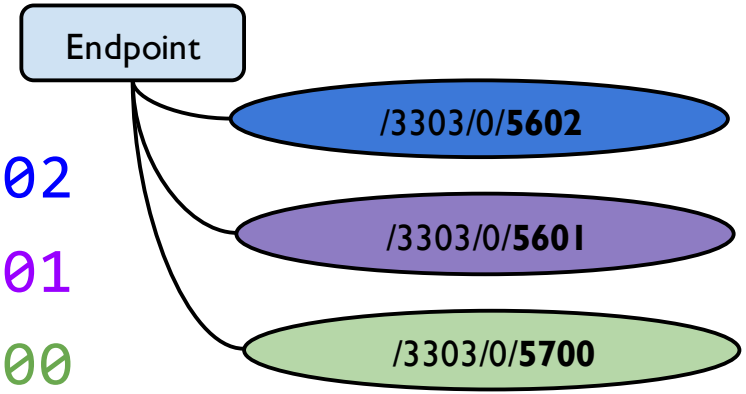



ex) temperature sensor object

/**temp**/0/maxValue /**3303**/0/5602

/**temp**/0/minValue /**3303**/0/5601

/**temp**/0/sensorValue /**3303**/0/5700






ARM

# LWM2M Spec

http://technical.openmobilealliance.org/Technical/technical-information/omna/lightweight-m2m-lwm2m-object-registry

- XML
- Detail

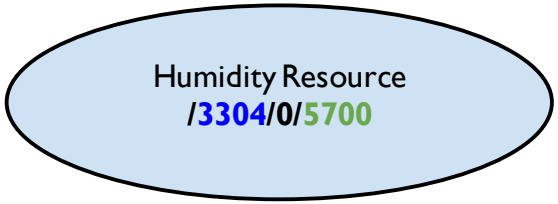
# Example: LWM2M

Example: Get temperature and humidity sensor data
(assume 1 sensor per endpoint)

Temperature ObjectID = **3303**

Humidity ObjectID = **3304**

Sensor Value *ResourceID* = **5700**



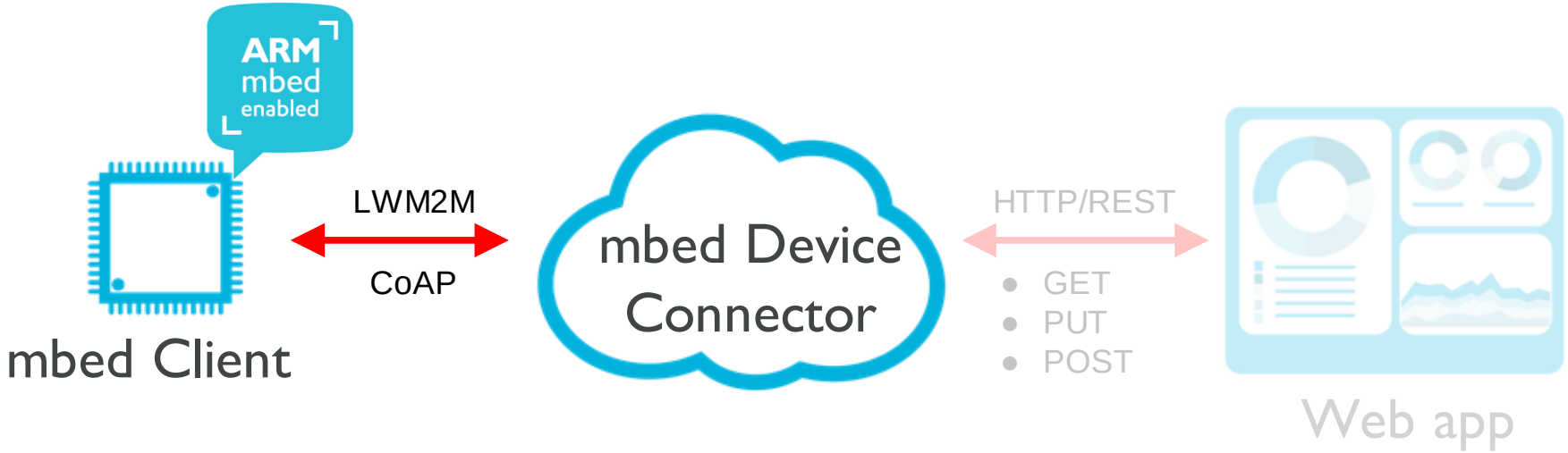

# Takeaway

Path describes data

- How you access the data tells you what the data is
- URL-based accessors
- ResourceID can have same number but different meaning, depending on object ID

# mbed Client code

**Let's build an endpoint!**



# mbed Client endpoint code

1. Plug in board to computer
2. Open mbed.html on board
3. Import code to online compiler: mbed.com/s/mc2016
4. Get security certificate from connector.mbed.com/#credentials
5. Paste into security.h
6. Compile / download code
7. Drag and drop code to board
8. Turn on board

# Serial console

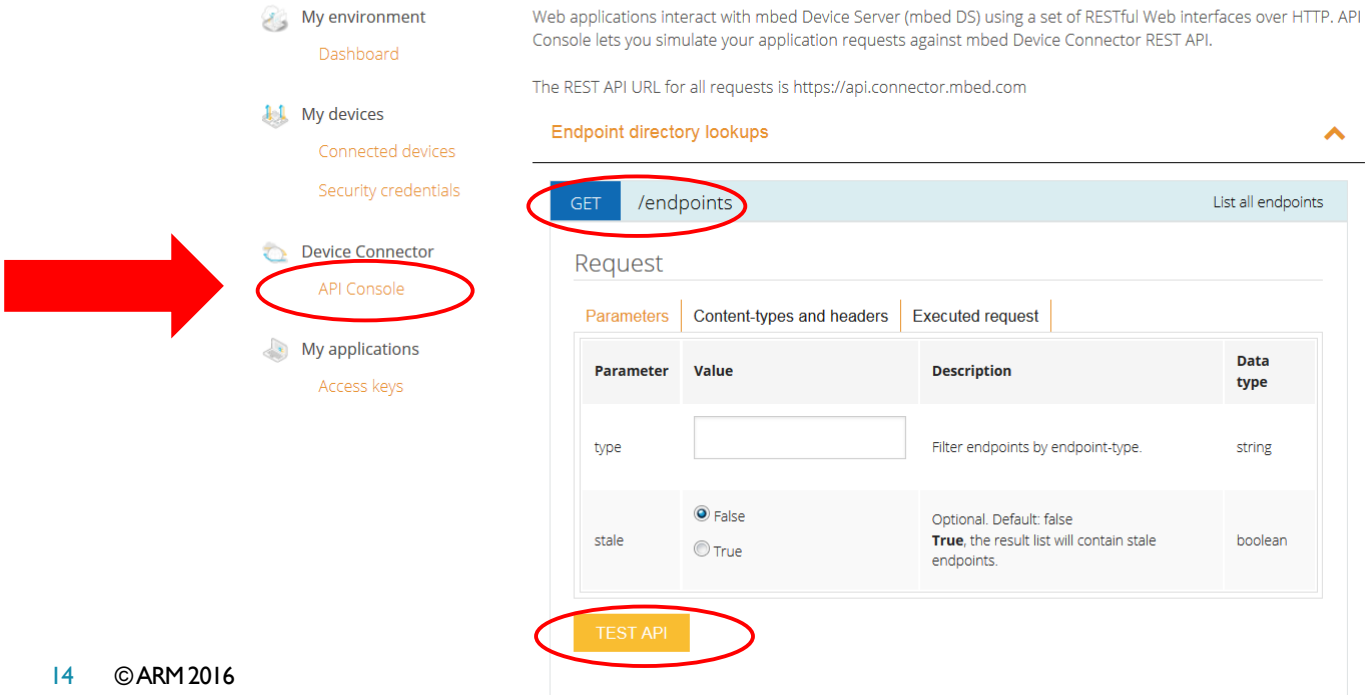
1. Open CoolTerm
2. Options -> Baudrate -> 9600 -> OK
3. Connect
4. Reset board

# API Console

## View endpoint data on connector website

# connector.mbed.com/#console

## API Console

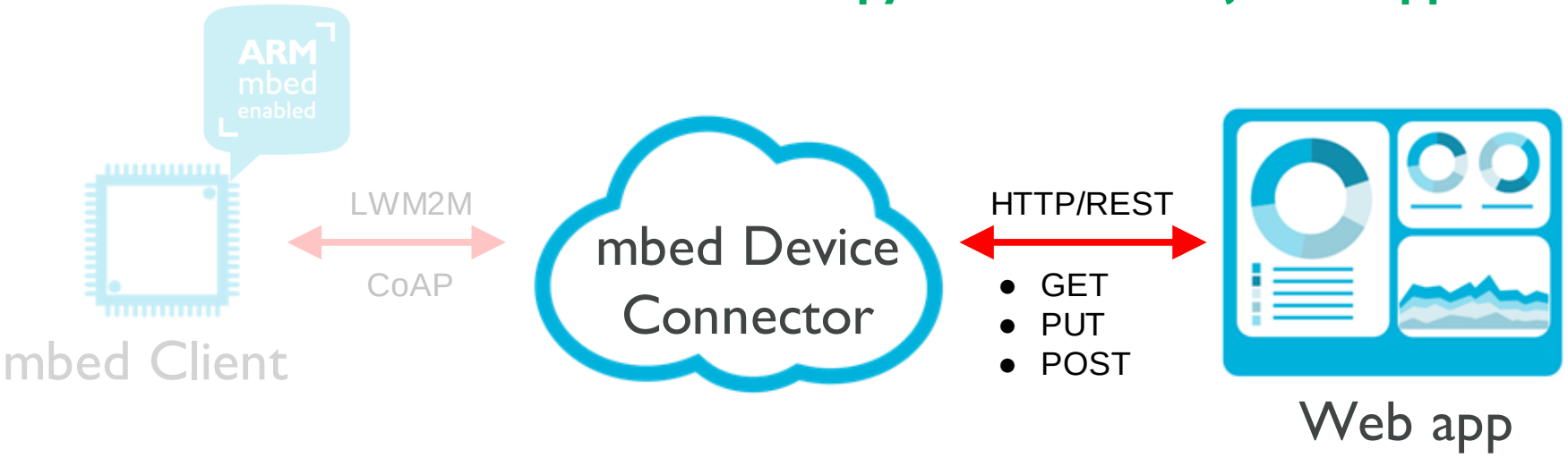


# Use API Console

- GET number of button presses '/3200/0/5501'
- POST to blink LED '/3201/0/5850'
- PUT to /3201/0/5853 change pattern of LED '1000:500:1000:500:1000:500'

# Web app

**Create python or Node.js web app**



# Node.js / Python web app

1. Open node.app in editor
2. Get access key from connector.mbed.com/#accesskeys
3. Replace 'Change Me' with access key in quotes
4. Save
5. Run command `npm install` / `pip install –r requirements.txt`
    1. If pip not work, update it with `pip install pip --upgrade`
6. Run app `node app.js` / `python app.py'
7. Open web browser to 127.0.0.1:8080
8. See the app run!

# Challenge

## Let's modify the code!

# Challenge 1 & 2

- Challenge 1 – Add a resource to endpoint
  - Add resource to change light color
    - Hint: **LED1**, **LED2**, **LED3**
  - Compile; load code to board
  - Verify in API Console

- Challenge 2 – Add resource to web app
  - Modify web app code to accept input and send value to device

# Questions?

Ask anything about mbed Client
or mbed Device Connector?