openHAB Alexa

Past, Present and Future

Dan Cunningham
@digitaldan

Agenda

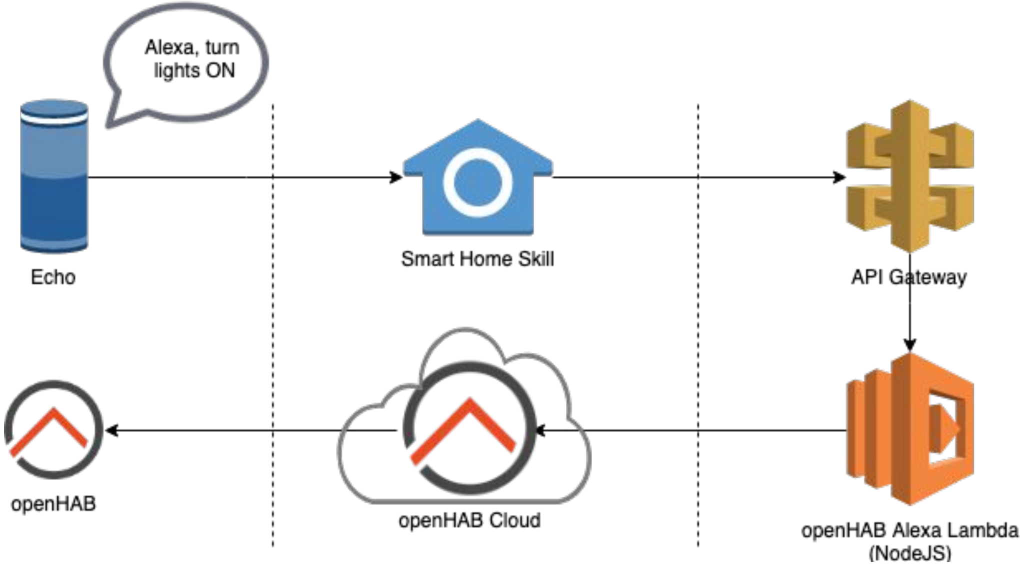
- A little history
- How Alexa and openHAB work together
- Past - Alexa v2 skill
- Present - Alexa v3 skill
- Future and beyond

A little history

- Amazon Echo was released in November 2014
- Alexa private home automation program was launched on April 8, 2015.
- Alexa public smart home API beta program Aug 2015
- openHAB Skill development started April 18 2016,
- openHAB Skill Official Release Feb 21 2017

amazon echo
Volume ring
Reflex port
Enhances the woofer's output for deeper sounds without distortion
2.5 inch woofer
Delivers deep bass response
2.0 inch tweeter
Crisply hits the high notes
9.25 in
3.27 in
Light ring
Action button
7-microphone array
Microphone off button
Power adapter
Echo also includes a remote with a built-in microphone and music playback and volume controls.

How it works

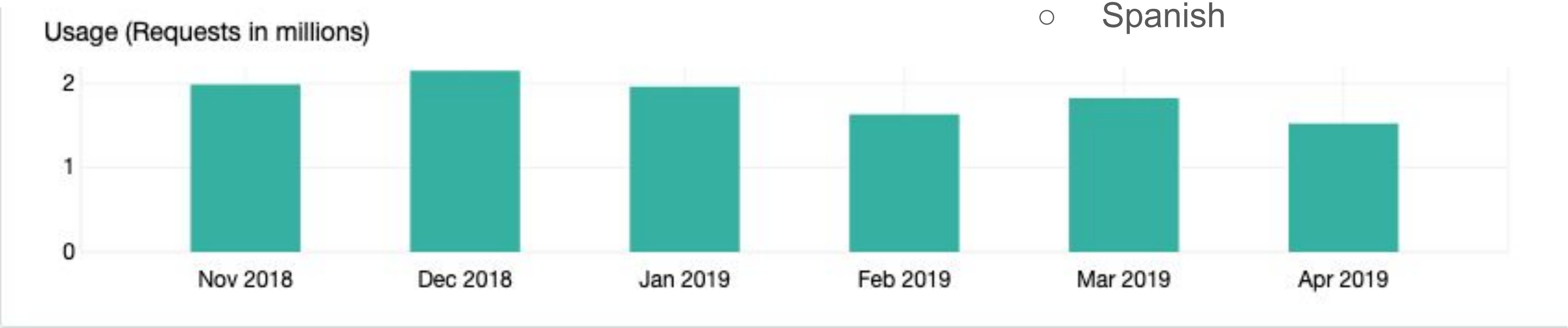


- Alexa request is routed to the openHAB Alexa Smart Home skill
- The Skill presents a AWS lambda function behind a AWS API gateway
- The lambda makes REST calls to openHAB Cloud to be proxied
- A REST endpoint (/rest/items) is called on the user’s openHAB.

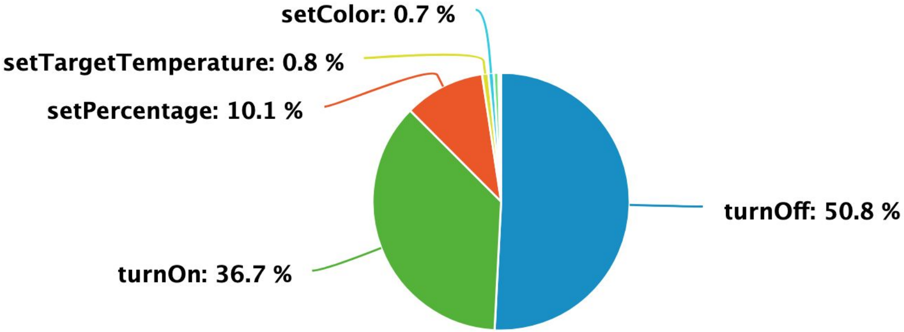
Usage

- 6000+ users
- 757,815 voice commands last month
- 1,848,443 avg requests per month

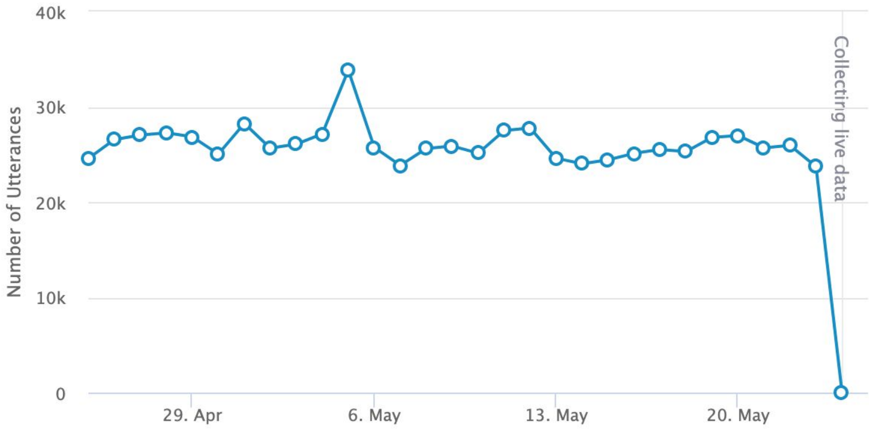
- Regions and Languages
  - English (AU)
  - English (UK)
  - French (CA)
  - English (US)
  - English (CA)
  - English (IN)
  - Spanish (MX)
  - Italian
  - German
  - French (FR)
  - Spanish

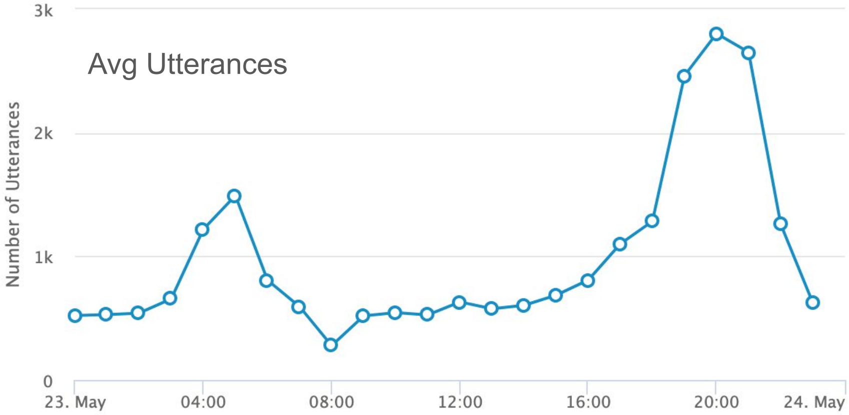


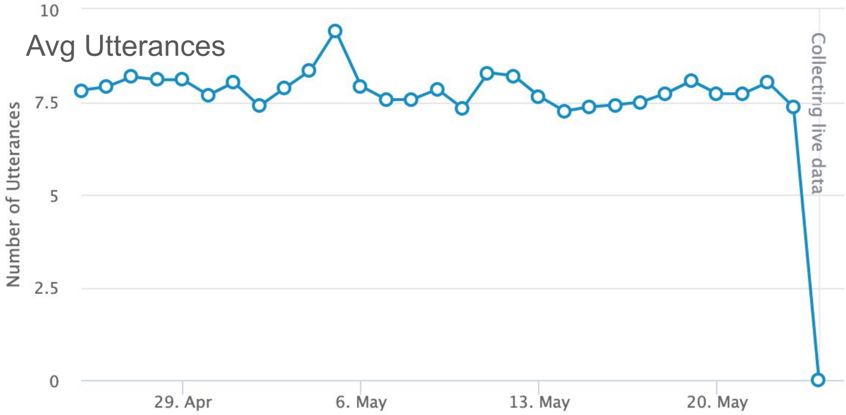
Usage



Total Utterances







Past (ish): openHAB + Alexa v2

- Based on the first public release (v2) of the Alexa Smart Home API
- Smart Home Skill != Basic Alexa Skill
    - Does not require a application word
    - Different review process
- Official skill supports the basic v2 set of features
    - On / OFF
    - Percentage
    - Brighten / Dim
    - Lock (but not unlock)
    - Temperature (Set / Increase / Decrease)

openHAB -> V2 mapping

- **Items** are mapped to **Appliances**
- Appliances have “**actions**’
- Actions are map to a item tag
  - Lighting, Switchable,CurrentTemperature, homekit:HeatingCoolingMode, TargetTemperature
- Simple logic, less than 1000 lines of NodeJS code.

```
Dimmer Kitchen_Light "Kitchen Light"
<light> (gKitchen) ["Lighting"]
{channel="..."}
```

```
{
  "applianceId": "Kitchen Light",
  "manufacturerName": "openHAB",
  "modelName": "Dimmer",
  "version": "2",
  "friendlyName": "Kitchen Light",
  "friendlyDescription": "",
  "isReachable": true,
  "actions": [
    "turnOn",
    "turnOff",
    "setPercentage",
    "incrementPercentage",
    "decrementPercentage"
  ],
  "additionalApplianceDetails": {
  }
}
```

Present: openHAB + Alexa v3

- Version 3 (v3) of the Smart Home API introduces a much richer set functionality
  - **Switchable***
  - **Lighting***
  - **ColorTemperature***
  - **Lock***
  - **CurrentTemperature***
  - **TargetTemperature***
  - LowerTemperature
  - UpperTemperature
  - HeatingCoolingMode
  - Activity
  - Scene
  - EntertainmentChannel
  - EntertainmentInput
  - EqualizerBass
  - EqualizerMidrange
  - EqualizerTreble
  - EqualizerMode
  - MediaPlayer
  - SpeakerMute
  - SpeakerVolume
  - ContactSensor
  - MotionSensor
  - SecurityAlarmMode
  - BurglaryAlarm
  - FireAlarm
  - CarbonMonoxideAlarm
  - WaterAlarm
  - ModeComponent
  - RangeComponent
  - ToggleComponent

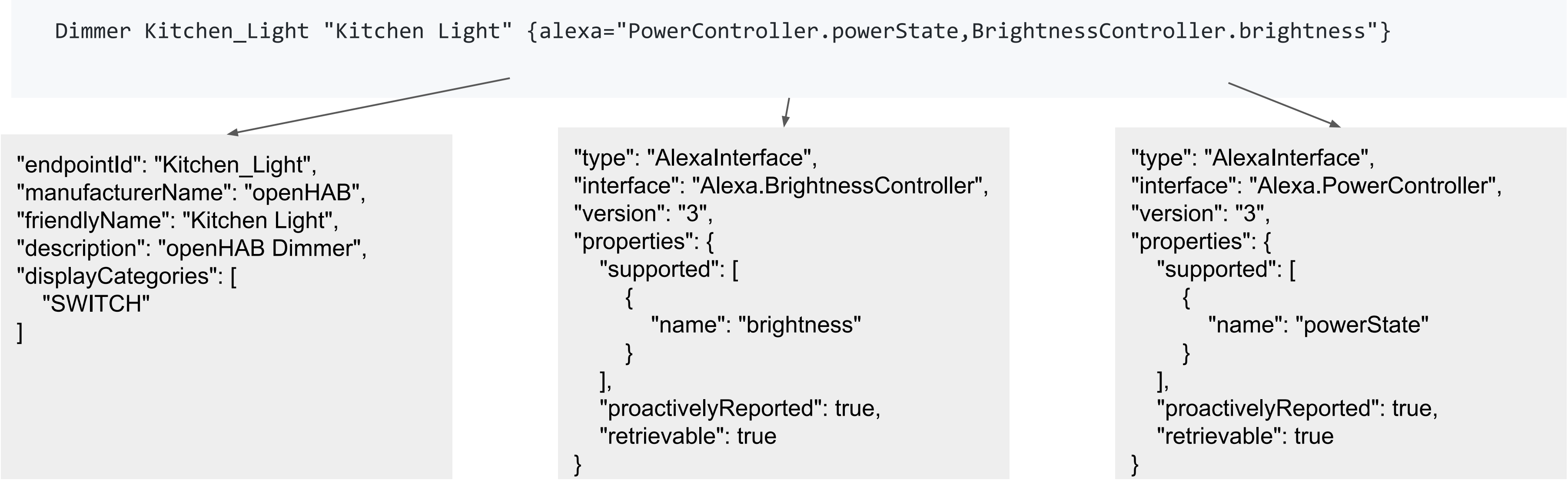
*** Denotes v2 support**

Present: openHAB + Alexa v3

- openHAB Alexa v3 code merges into master **this weekend!**
- Official Skill Submission will happen **next week**.
- Complete rewrite of existing code base
- Uses Modern ES6 Javascript
- Very high percentage of unit testing
- Automated build and deployments
- Modular design
- **HUGE** Thank you to Jeremy (aka Github @jsetton) who has taken the lead on all of the above plus a TON of the new functionality. (Seriously Amazing)

openHAB -> V3 mapping

- Item **metadata** is used, not tags
- Items are mapped to **Endpoints**
- Endpoints contain **Capability Interfaces**
- Capabilities and **Properties** describe an endpoint’s functionality



```
Dimmer Kitchen_Light "Kitchen Light" {alexa="PowerController.powerState,BrightnessController.brightness"}
```

```
"endpointId": "Kitchen_Light",
"manufacturerName": "openHAB",
"friendlyName": "Kitchen Light",
"description": "openHAB Dimmer",
"displayCategories": [
    "SWITCH"
]
```

```
"type": "AlexaInterface",
"interface": "Alexa.BrightnessController",
"version": "3",
"properties": {
    "supported": [
        {
            "name": "brightness"
        }
    ],
    "proactivelyReported": true,
    "retrievable": true
}
```

```
"type": "AlexaInterface",
"interface": "Alexa.PowerController",
"version": "3",
"properties": {
    "supported": [
        {
            "name": "powerState"
        }
    ],
    "proactivelyReported": true,
    "retrievable": true
}
```

Group Items

- Groups map **many** openHAB **items** to a single Alexa **endpoint**

Thermostat:

```
Group  Thermostat    "Bedroom"                                {alexa="Endpoint.Thermostat"}
Number Temperature   "Temperature [%.0f F]"    (Thermostat)   {alexa="TemperatureSensor.temperature"}
Number HeatSetpoint  "Heat Setpoint [%.0f F]"  (Thermostat)   {alexa="ThermostatController.upperSetpoint"}
Number CoolSetpoint  "Cool Setpoint [%.0f F]"  (Thermostat)   {alexa="ThermostatController.lowerSetpoint"}
Number Mode          "Mode [%s]"               (Thermostat)   {alexa="ThermostatController.thermostatMode"}
```

Stereo:

```
Group Stereo    "Stereo"            {alexa="Endpoint.Speaker"}
Number Volume   "Volume"  (Stereo)  {alexa="Speaker.volume"}
Switch Mute     "Mute"    (Stereo)  {alexa="Speaker.muted"}
Switch Power    "Power"   (Stereo)  {alexa="PowerController.powerState"}
String Input    "Input"   (Stereo)  {alexa="InputController.input" [supportedInputs="HDMI1,TV"]}
String Channel  "Channel" (Stereo)  {alexa="ChannelController.channel"}
Player Player   "Player"  (Stereo)  {alexa="PlaybackController.playbackState"}
Number Bass     "Bass"    (Stereo)  {alexa="EqualizerController.bands:bass" [range="-10:10"]}
Number Midrange "Mid"     (Stereo)  {alexa="EqualizerController.bands:midrange" [range="-10:10"]}
Number Treble   "Treble"  (Stereo)  {alexa="EqualizerController.bands:treble" [range="-10:10"]}
String Mode     "Mode"    (Stereo)  {alexa="EqualizerController.modes" [supportedModes="MOVIE,MUSIC,TV"]}
```

Group Items

Alarm Panel:

```
Group  SecuritySystem      "Security System"                  {alexa="Endpoint.SecurityPanel"}
String AlarmMode           "Alarm Mode"      (SecuritySystem) {alexa="SecurityPanelController.armState"
[supportedArmStates="DISARMED,ARMED_STAY,ARMED_AWAY"]}
Switch BurglaryAlarm       "Burglary"        (SecuritySystem) {alexa="SecurityPanelController.burglaryAlarm"}
Switch FireAlarm           "Fire"            (SecuritySystem) {alexa="SecurityPanelController.fireAlarm"}
Switch CarbonMonoxideAlarm "Carbon Monoxide" (SecuritySystem) {alexa="SecurityPanelController.carbonMonoxideAlarm"}
Switch WaterAlarm          "Water"           (SecuritySystem) {alexa="SecurityPanelController.waterAlarm"}
```

Washing Machine:

```
Group Washer       "Washer"                {alexa="Endpoint.Other"}
String Cycle       "Cycle"        (Washer) {alexa="ModeController.mode"
[supportedModes="Normal=Normal:Cottons,Delicate=@Value.Delicate:Knites",friendlyNames="Wash Cycle,Wash Setting",ordered=false]}
Number Temperature "Temperature" (Washer) {alexa="ModeController.mode"
[supportedModes="0=Cold:Cool,1=Warm,2=Hot",friendlyNames="Wash Temperature,@Setting.WaterTemperature",ordered=true]}
  Switch Power       "Power"       (Washer) {alexa="ToggleController.toggleState" [friendlyNames="@DeviceName.Washer"]}
```

Metadata Labels

- Alexa metadata is powerful and configurable, but also verbose
- Item **metadata labels** translates to a set of **capabilities** and can be used as a convenience to using the longer metadata format configuration
- 2 Examples (but there labels for every interface type)

```
Dimmer LightDimmer "Light Dimmer" {alexa="Lighting"}

Shorthand for

Dimmer LightDimmer "Light Dimmer" {alexa="PowerController.powerState,BrightnessController.brightness" [category="LIGHT"]}
```

```
Rollershutter ShutterSwitch "Shutter Switch" {alexa="Switchable"}

Shorthand for

Rollershutter ShutterSwitch "Shutter Switch" {alexa="PowerController.powerState,PercentageController.percentage" [category="SWITCH"]}
```

Other Features

- Item State Formating
- UoM support
- Item Sensors
- Item State Presentation
- Display Capabilities
- V2 backwards compatibility

Future: openHAB Alexa Binding + Alexa v3

- Original design was to use “out of the box” API’s that openHAB already provided.
- Resisted suggestions to using a binding (yes, i’m stubborn)
- But…. there are good reasons for V3
  - Change Reports
  - Asynchronous Response
  - Proactive Discovery
  - Too expensive for openHAB Cloud or Lambda
- Other benefits
  - Moves intensive logic to user’s computer
  - Reduces amount of REST calls needed per request
  - Simplifies logging and debugging issues

Thanks

- https://github.com/openhab/openhab-alexa
- https://github.com/openhab/openhab-alexa/blob/master/USAGE.md