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
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

- setColor: 0.7%
- setTargetTemperature: 0.8%
- setPercentage: 10.1%
- turnOn: 36.7%
- turnOff: 50.8%

Total Utterances

Avg Utterances

Number of Utterances

Avg Utterances

Number of 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 mapped 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="..."}
```
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

```json
"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
```

```
"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
```

```
"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
```

```
"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
```

```
"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:**

<table>
<thead>
<tr>
<th>Group</th>
<th>Item</th>
<th>Type</th>
<th>Alexa</th>
</tr>
</thead>
<tbody>
<tr>
<td>Thermostat</td>
<td>&quot;Bedroom&quot;</td>
<td>(Thermostat)</td>
<td>&quot;Endpoint.Thermostat&quot;</td>
</tr>
<tr>
<td>Temperature</td>
<td>&quot;Temperature [%.0f F]&quot;</td>
<td>(Thermostat)</td>
<td>&quot;TemperatureSensor.temperature&quot;</td>
</tr>
<tr>
<td>HeatSetpoint</td>
<td>&quot;Heat Setpoint [%.0f F]&quot;</td>
<td>(Thermostat)</td>
<td>&quot;ThermostatController.upperSetpoint&quot;</td>
</tr>
<tr>
<td>CoolSetpoint</td>
<td>&quot;Cool Setpoint [%.0f F]&quot;</td>
<td>(Thermostat)</td>
<td>&quot;ThermostatController.lowerSetpoint&quot;</td>
</tr>
<tr>
<td>Mode</td>
<td>&quot;Mode [%s]&quot;</td>
<td>(Thermostat)</td>
<td>&quot;ThermostatController.thermostatMode&quot;</td>
</tr>
</tbody>
</table>

**Stereo:**

<table>
<thead>
<tr>
<th>Group</th>
<th>Item</th>
<th>Type</th>
<th>Alexa</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stereo</td>
<td>&quot;Stereo&quot;</td>
<td>(Stereo)</td>
<td>&quot;Endpoint.Speaker&quot;</td>
</tr>
<tr>
<td>Volume</td>
<td>&quot;Volume&quot;</td>
<td>(Stereo)</td>
<td>&quot;Speaker.volume&quot;</td>
</tr>
<tr>
<td>Mute</td>
<td>&quot;Mute&quot;</td>
<td>(Stereo)</td>
<td>&quot;Speaker.muted&quot;</td>
</tr>
<tr>
<td>Power</td>
<td>&quot;Power&quot;</td>
<td>(Stereo)</td>
<td>&quot;PowerController.powerState&quot;</td>
</tr>
<tr>
<td>Input</td>
<td>&quot;Input&quot;</td>
<td>(Stereo)</td>
<td>&quot;InputController.input&quot; [supportedInputs=&quot;HDMI1,TV&quot;]</td>
</tr>
<tr>
<td>Channel</td>
<td>&quot;Channel&quot;</td>
<td>(Stereo)</td>
<td>&quot;ChannelController.channel&quot;</td>
</tr>
<tr>
<td>Player</td>
<td>&quot;Player&quot;</td>
<td>(Stereo)</td>
<td>&quot;PlaybackController.playbackState&quot;</td>
</tr>
<tr>
<td>Bass</td>
<td>&quot;Bass&quot;</td>
<td>(Stereo)</td>
<td>&quot;EqualizerController.bands:bass&quot; [range=&quot;-10:10&quot;]</td>
</tr>
<tr>
<td>Midrange</td>
<td>&quot;Mid&quot;</td>
<td>(Stereo)</td>
<td>&quot;EqualizerController.bands:midrange&quot; [range=&quot;-10:10&quot;]</td>
</tr>
<tr>
<td>Treble</td>
<td>&quot;Treble&quot;</td>
<td>(Stereo)</td>
<td>&quot;EqualizerController.bands:treble&quot; [range=&quot;-10:10&quot;]</td>
</tr>
<tr>
<td>Mode</td>
<td>&quot;Mode&quot;</td>
<td>(Stereo)</td>
<td>&quot;EqualizerController.modes&quot; [supportedModes=&quot;MOVIE,MUSIC,TV&quot;]</td>
</tr>
</tbody>
</table>
Alarm Panel:

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 Formatting
- 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