Internet of Python

IOT programming
with python
and serverless computing

@Sonal Raj

HydPy Meetup | Hyderabad | February 2019

Hi, I'm Sonal Raj...

.. I do a lot of open source work

.. graph databases, real time processing enthusiast

.. grad in CS, post-grad in IT and half an MBA

.. Python and I share 10+ years of friendship

.. Pursue research in education technology of late

.. Part time author and blogger

... and here is where you can find me

github.com/sonal-raj

twitter.com/@sonalraj

www.sonalraj.com

THE SCOPE

Internet Of Things

Serverless Computing

Python Integration

Create Applications

Ideas

… Enable things to connect with back-end systems that support your ability to act on the data …. Build a central nervous system that connects data insights to organizational (or individual) response….

IBM on IOT

Senses

Proximity to a surface

Can listen to what you say

See things in front and behind it.

Transfer Info over a wireless network

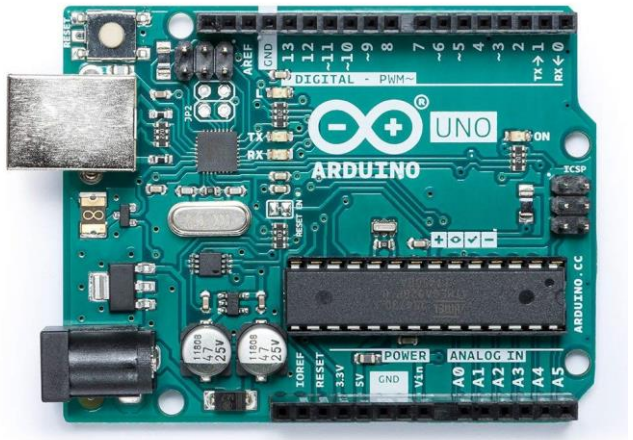
Your location

Your state of motion and direction

Orientation of the device

Ambience or light in the room

Raspberry Pi 3B+



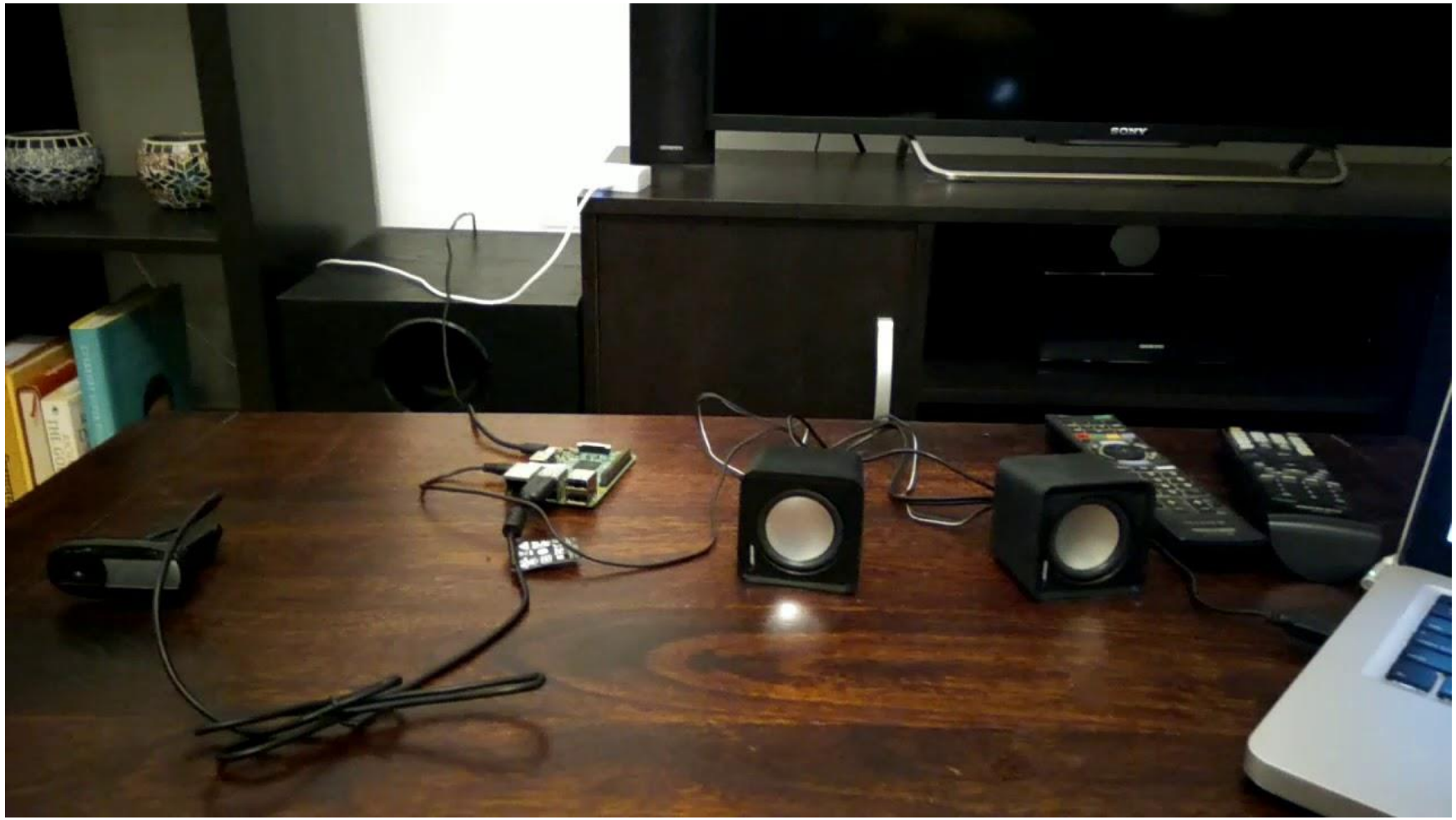
Arduino

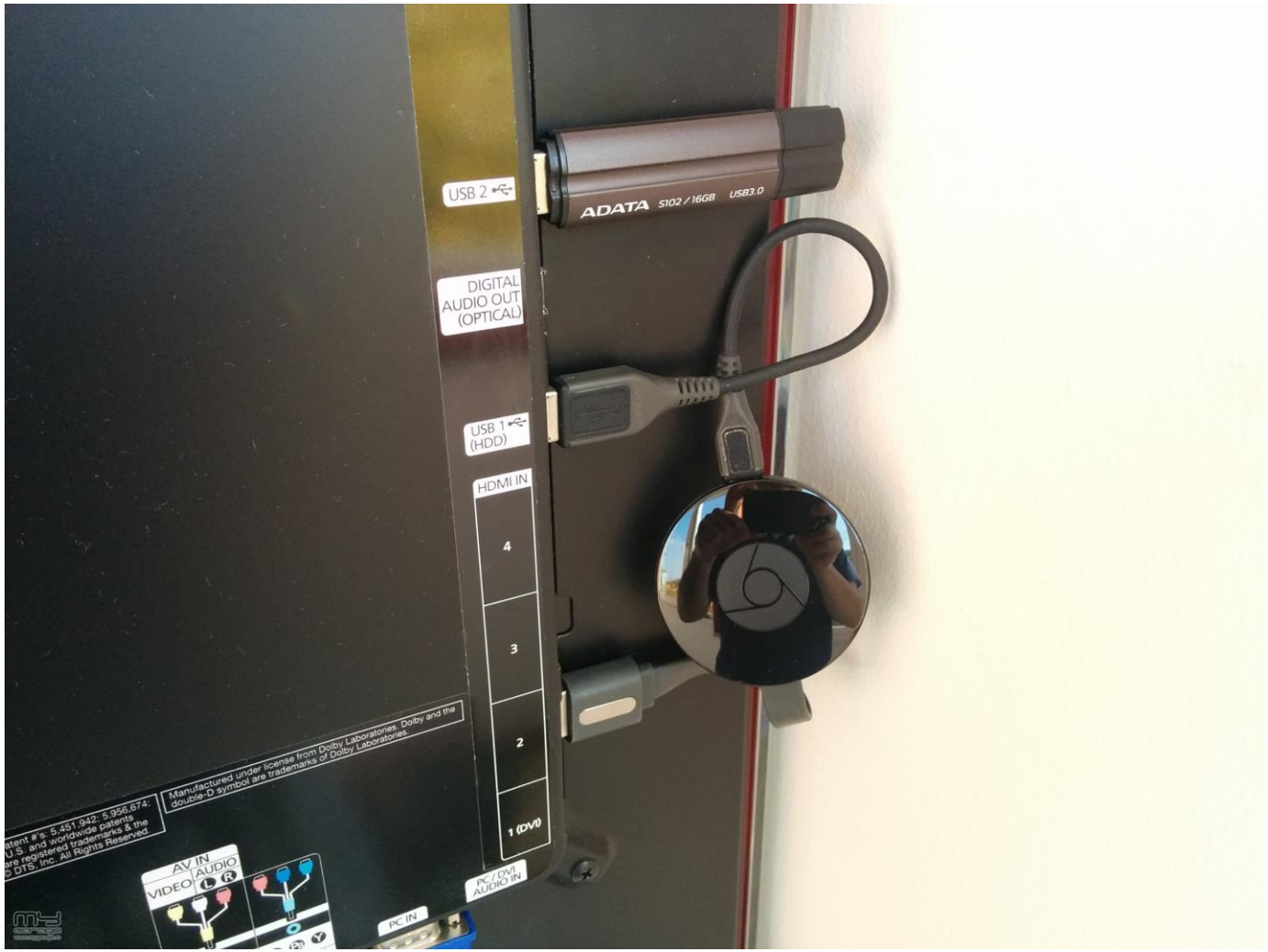
PLC / PAC

How does a basic IOT system really look like?


PHILIPS
PHILIPS
hue
PHILIPS


PHILIPS


SONY


USB 2
ADATA S102 / 16GB USB3.0
DIGITAL AUDIO OUT (OPTICAL)
USB 1 (HDD)
HDMI IN
4
3
2
1 (DVI)
Manufactured under license from Dolby Laboratories. Dolby and the double-D symbol are trademarks of Dolby Laboratories.
AV IN
VIDEO
AUDIO
L
R
PC IN
PC / DVI AUDIO IN


nest

Internet Enabled

Support for Digital Assistants

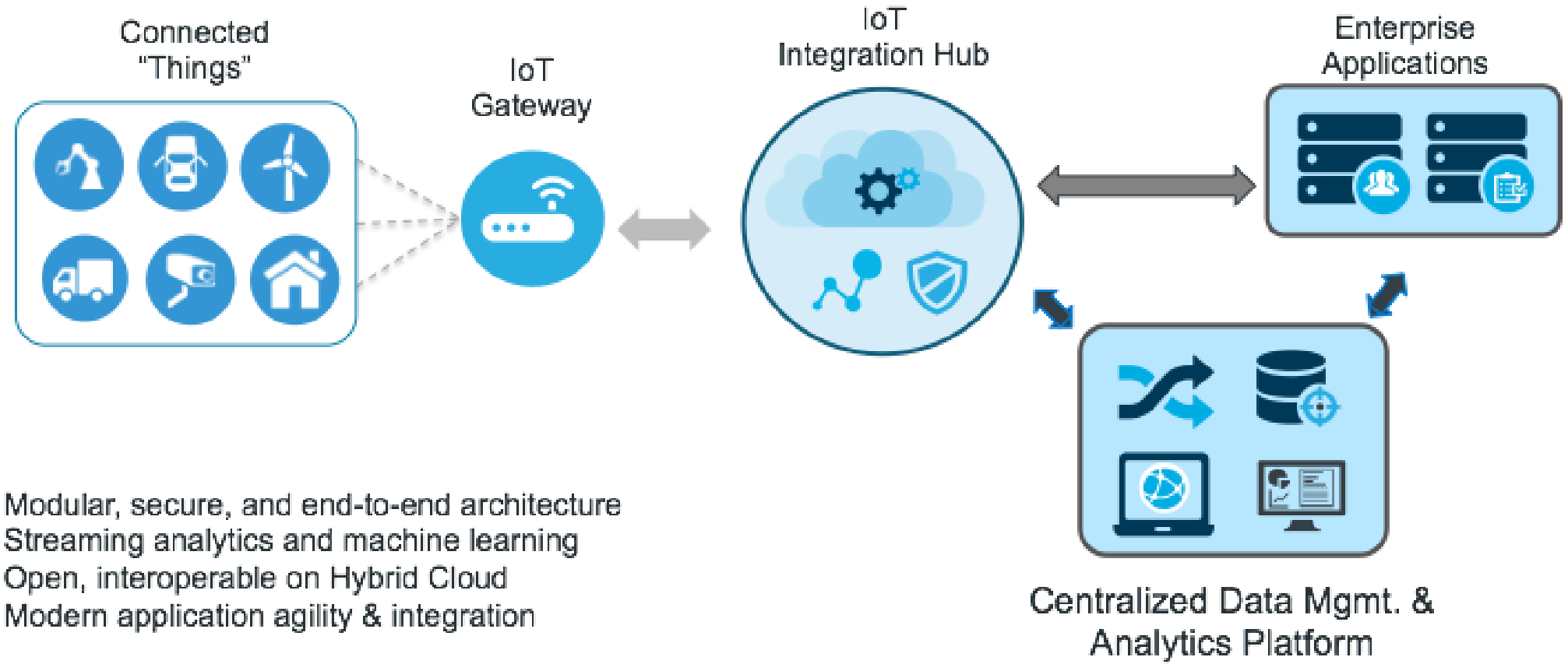
Smart phone Controlled

Programmable APIs

Google Drive

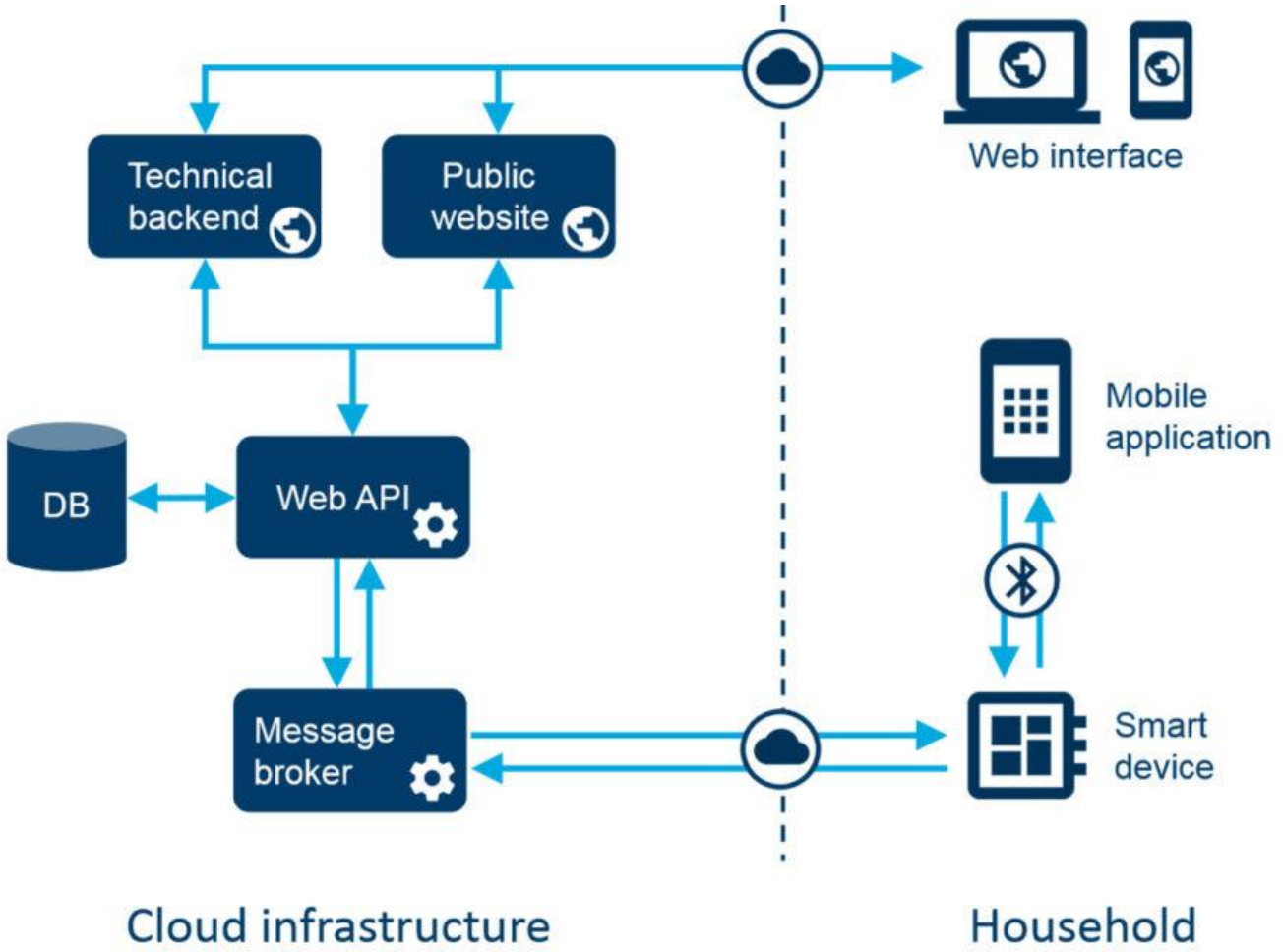
amazon alexa

Architecture



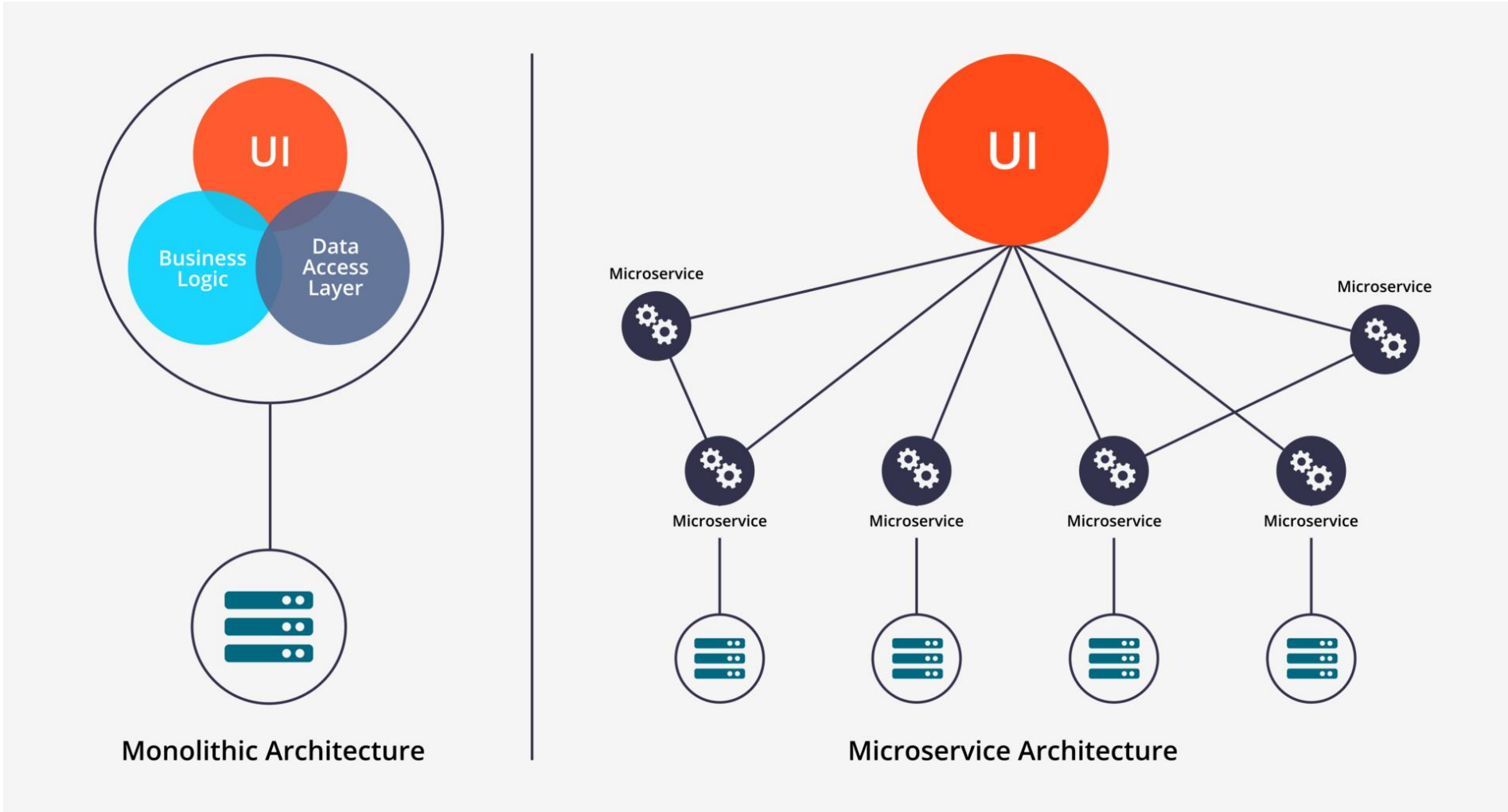
- Modular, secure, and end-to-end architecture
- Streaming analytics and machine learning
- Open, interoperable on Hybrid Cloud
- Modern application agility & integration

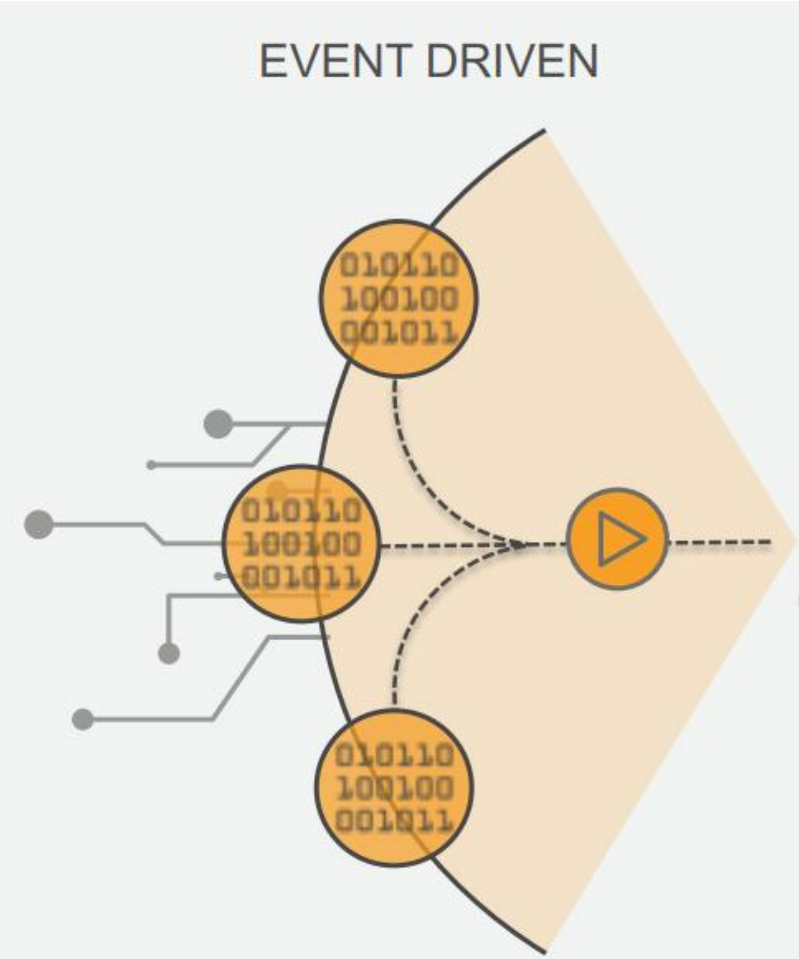
Smart Homes


Technical backend
Public website
Web interface
DB
Web API
Mobile application
Message broker
Smart device
Cloud infrastructure
Household

Smart Cities

Serverless Computing


UI
Business Logic
Data Access Layer
Monolithic Architecture
UI
Microservice
Microservice
Microservice
Microservice
Microservice
Microservice Architecture



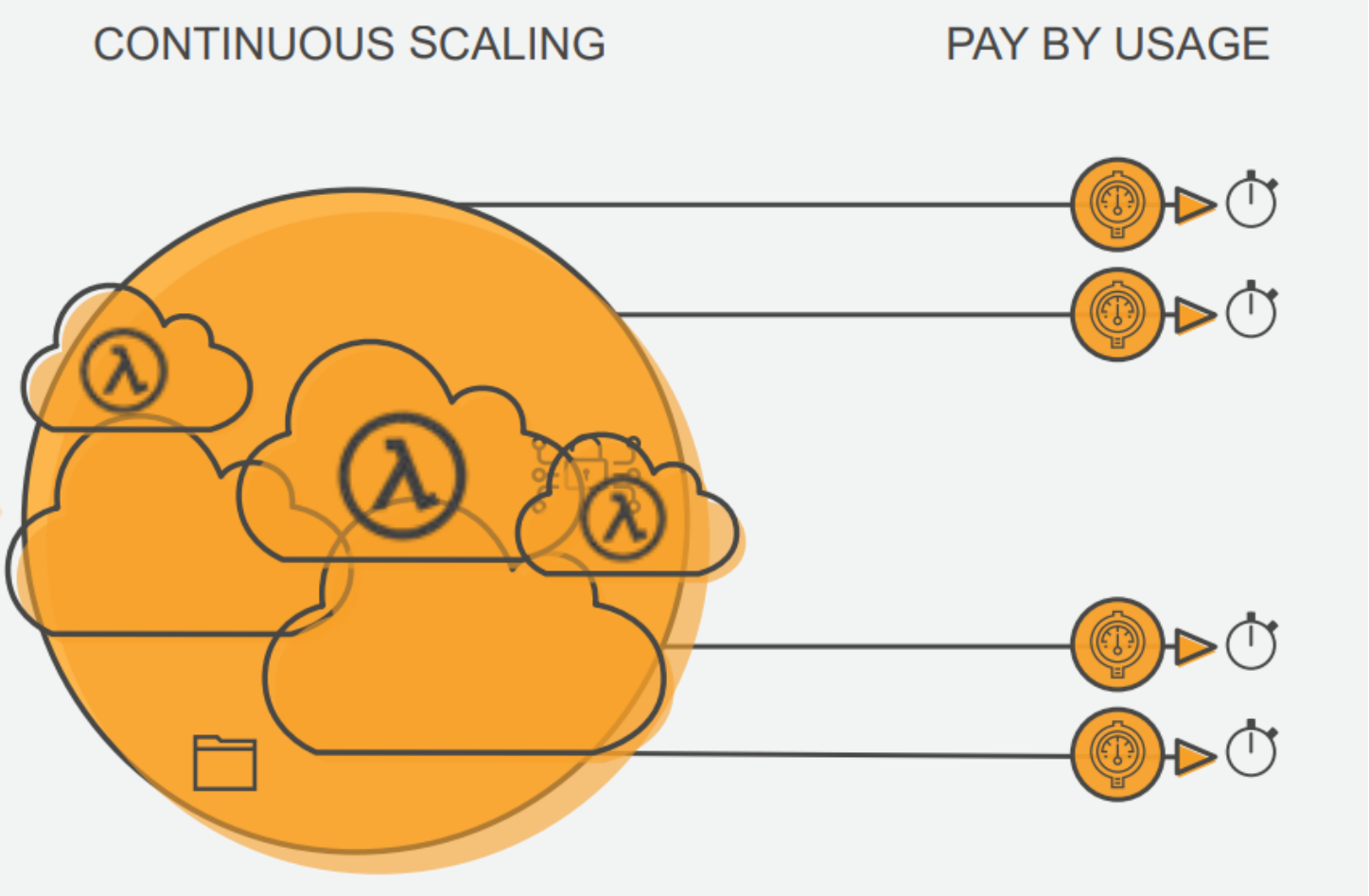


Image Source: Amazon Web Services

Event Triggers

Image Source: Amazon Web Services

So, why Serverless?

No provisioning or management of infrastructure

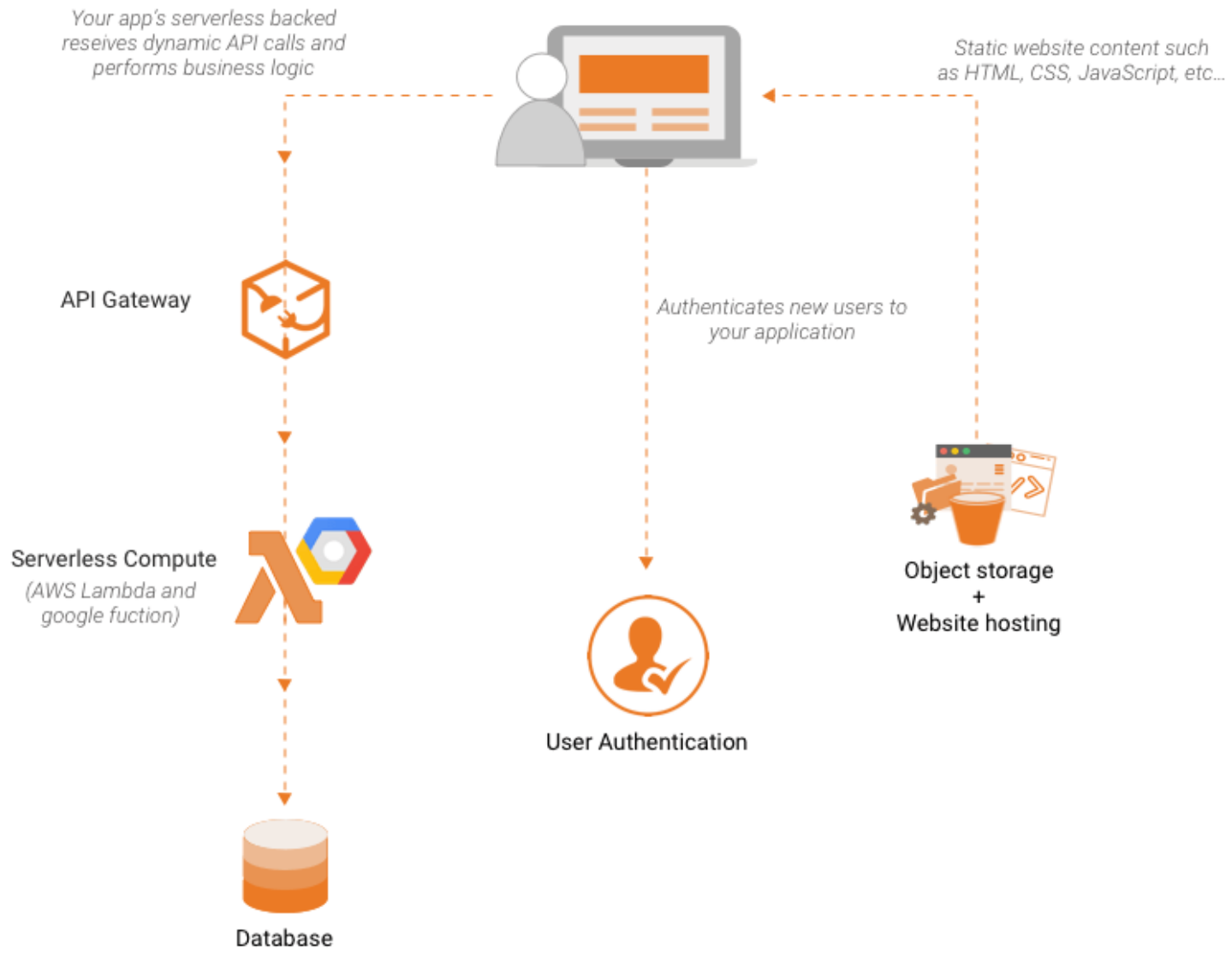
Automatic Scaling

Pay per use / No Idle Capacity

Automatic Implied Availability and Scalability

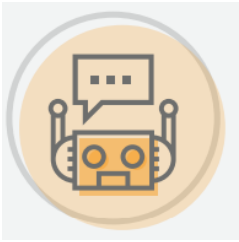
Serverless Providers

The how it works?



Applications

Static Websites, Web Apps, Flask Packages

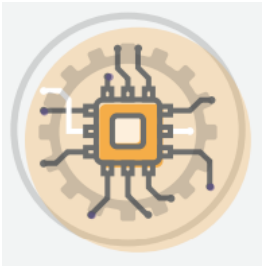


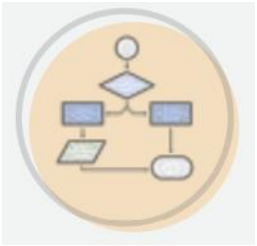
Creating logic for chatbots

Backend Apps and Services, Mobile and IOT

Voice enabled apps and skills in Alexa assistants

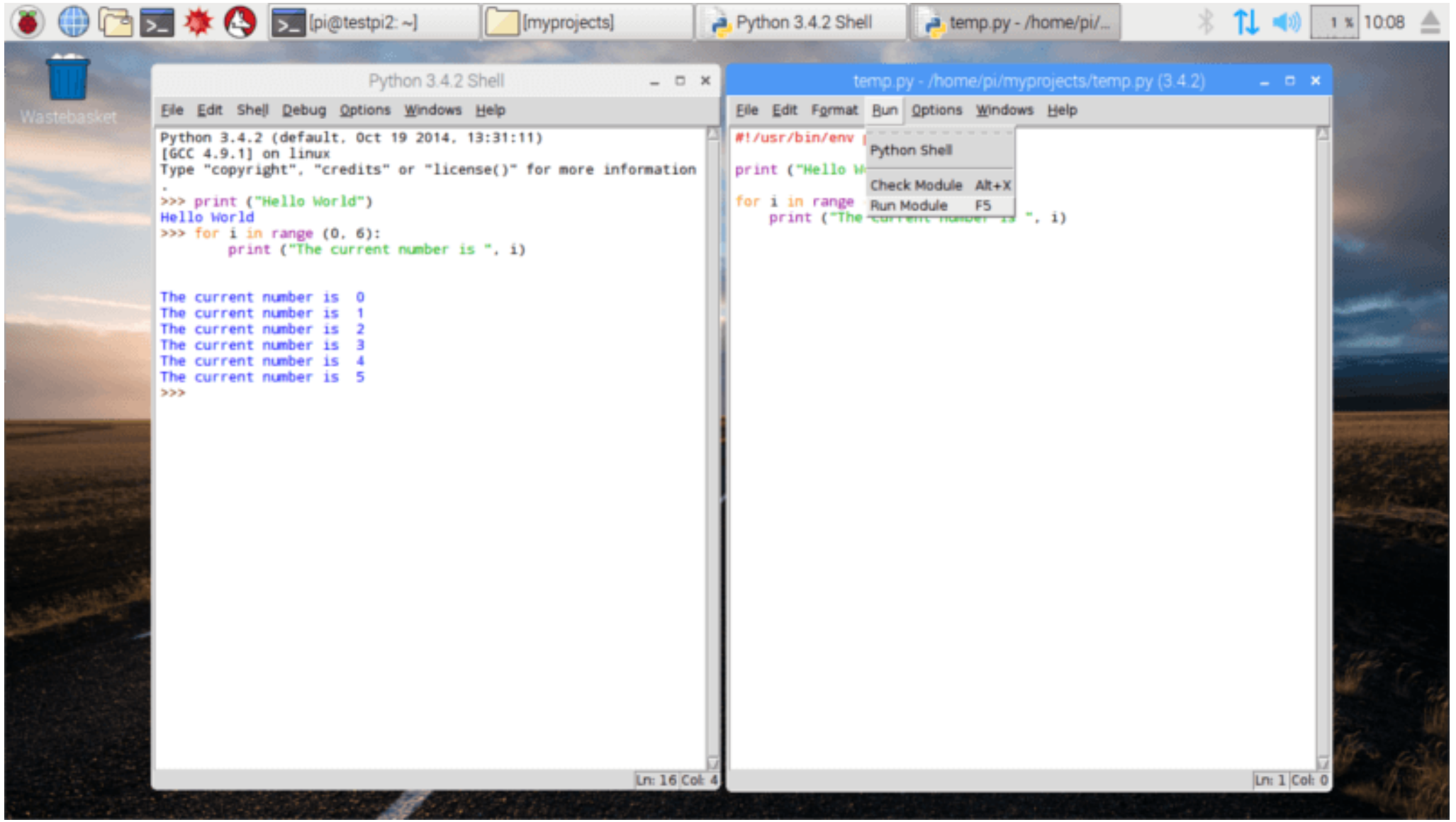
Data Processing in Real Time, Batch or MapReduce





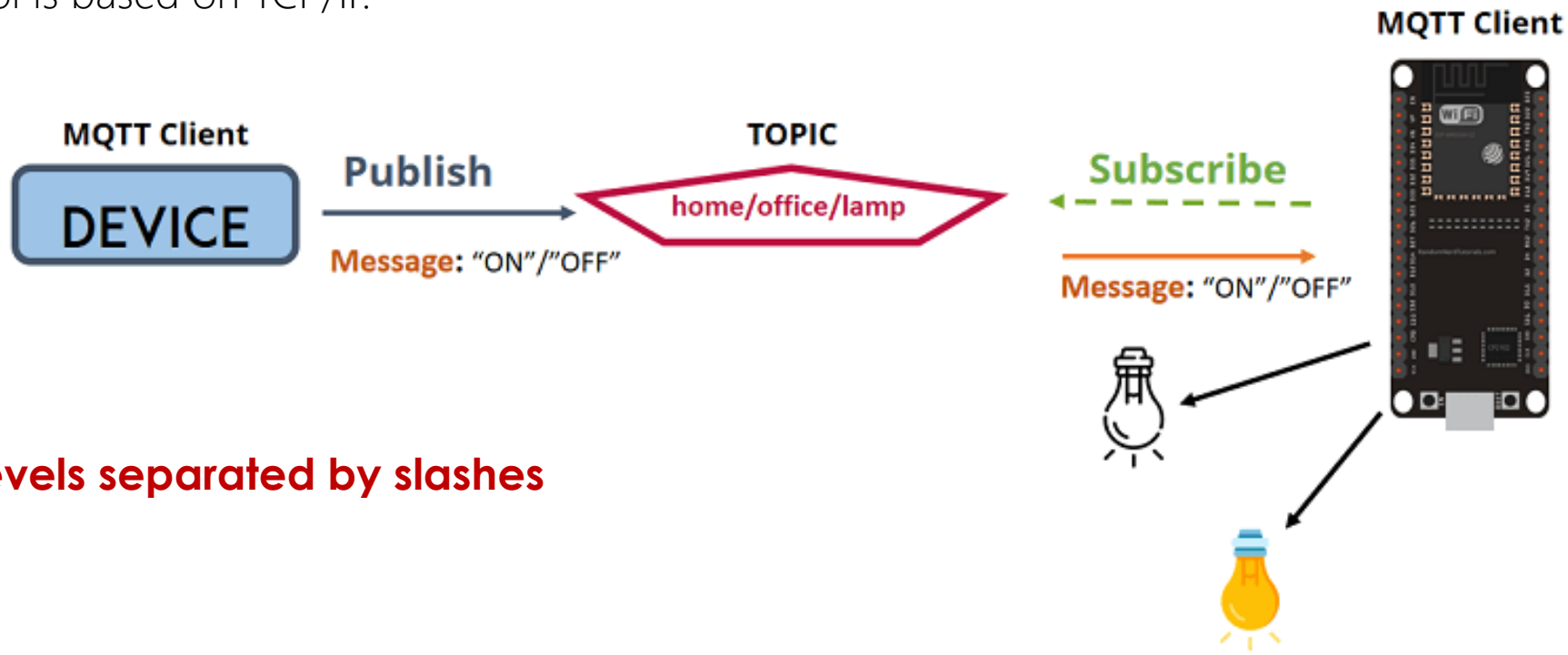
Automation of Services and Infrastructure management

What can we do with Python?


[pi@testpi2: ~]
[myprojects]
Python 3.4.2 Shell
temp.py - /home/pi/...
10:08
Wastebasket
Python 3.4.2 Shell
File Edit Shell Debug Options Windows Help
Python 3.4.2 (default, Oct 19 2014, 13:31:11)
[GCC 4.9.1] on linux
Type "copyright", "credits" or "license()" for more information.
>>> print ("Hello World")
Hello World
>>> for i in range (0, 6):
print ("The current number is ", i)
The current number is 0
The current number is 1
The current number is 2
The current number is 3
The current number is 4
The current number is 5
>>>
Ln: 16 Col: 4
temp.py - /home/pi/myprojects/temp.py (3.4.2)
File Edit Format Run Options Windows Help
Python Shell
Check Module Alt+X
Run Module F5
#!/usr/bin/env
print ("Hello
for i in range
print ("The
", i)
Ln: 1 Col: 0

MESSAGE QUEUEING
TELEMETRY TRANSPORT

- Machine to Machine connectivity protocol
- Lightweight Publish / Subscribe
- Small Footprint and low bandwidth
- The MQTT protocol is based on TCP/IP.



Topics - series of levels separated by slashes

```
house/alarm/status
house/alarm/zone

house/camera/capture
house/camera/newpicture
```

```
house/camera/#
house/camera/+
```

Messages

```
FFFFF11000 # Float device identifier
2-1235489698.jpg
```

Quality of Service

```
0 – best effort
1 – deliver at least once
2 – exactly once delivery
```

Security

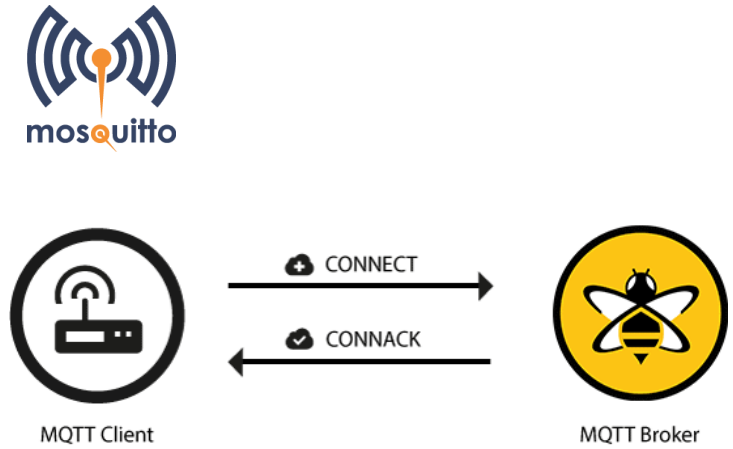
```
Userid / deviceid / passwords
TLS / Certificate
```

Clients

```
Paho-mqtt (https://pypi.org/project/paho-mqtt/)
```

Brokers

```
Mosquitto (Python)
Mosca (NodeJS)
```

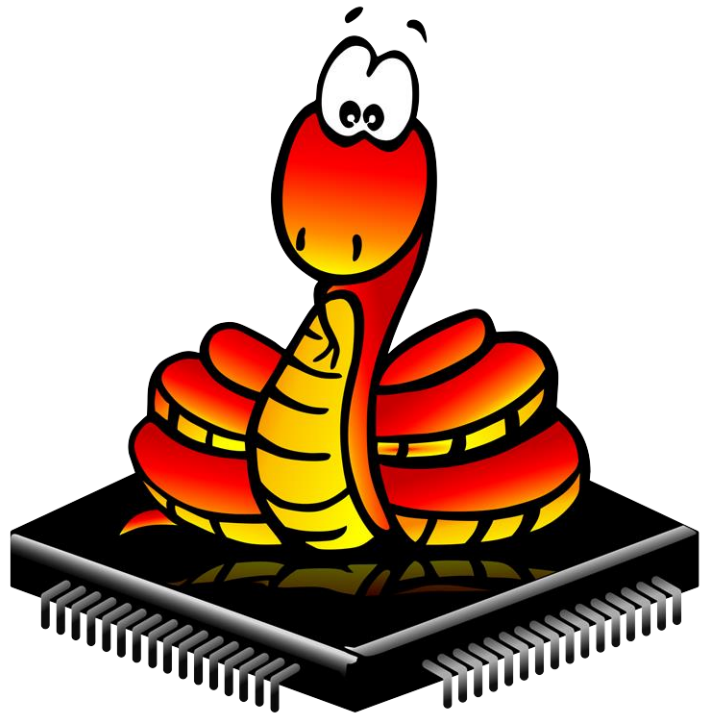


IOT Services

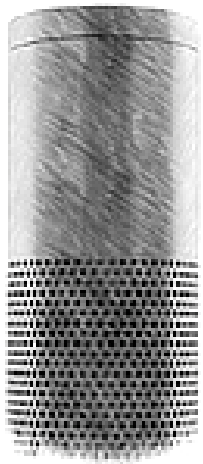
```
IBM bluemix
AWS-IOT
AZURE-IOT
```

libmraa - Low Level Skeleton Library for Communication on GNU/Linux platforms

- Easy Portable Interface for working with sensors and actuators
- Low level Interface
- Access and manipulate the basic I/O capabilities of a platform
- Translation layer on top of the Linux GPIO
- Platform Agnostic APIs
- Written in C, but has binding for many languages including Python!



Micropython



Flask-Ask

Rapid Alexa Skills Kit Development
for Amazon Echo Devices

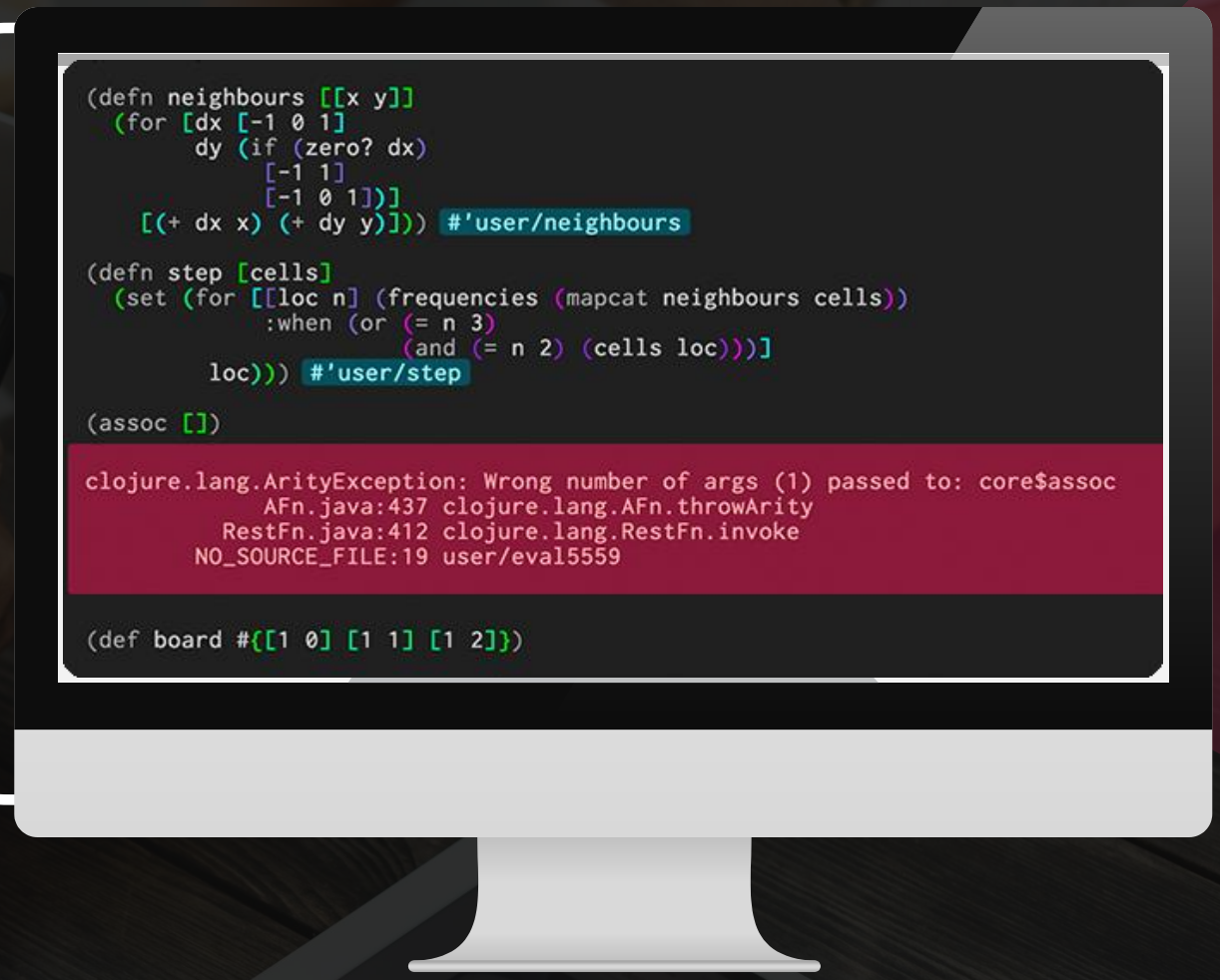
A Flask extension that has wrappers for the common constructs for the alexa skills API.

- Has decorators to map Alexa requests and intent slots to view functions
- Helps construct ask and tell responses, re-prompts and cards
- Makes session management easy
- Allows for the separation of code and speech through Jinja templates
- Verifies Alexa request signatures

Go through some smart home application code

Setup a serverless API using AWS Lambda functions

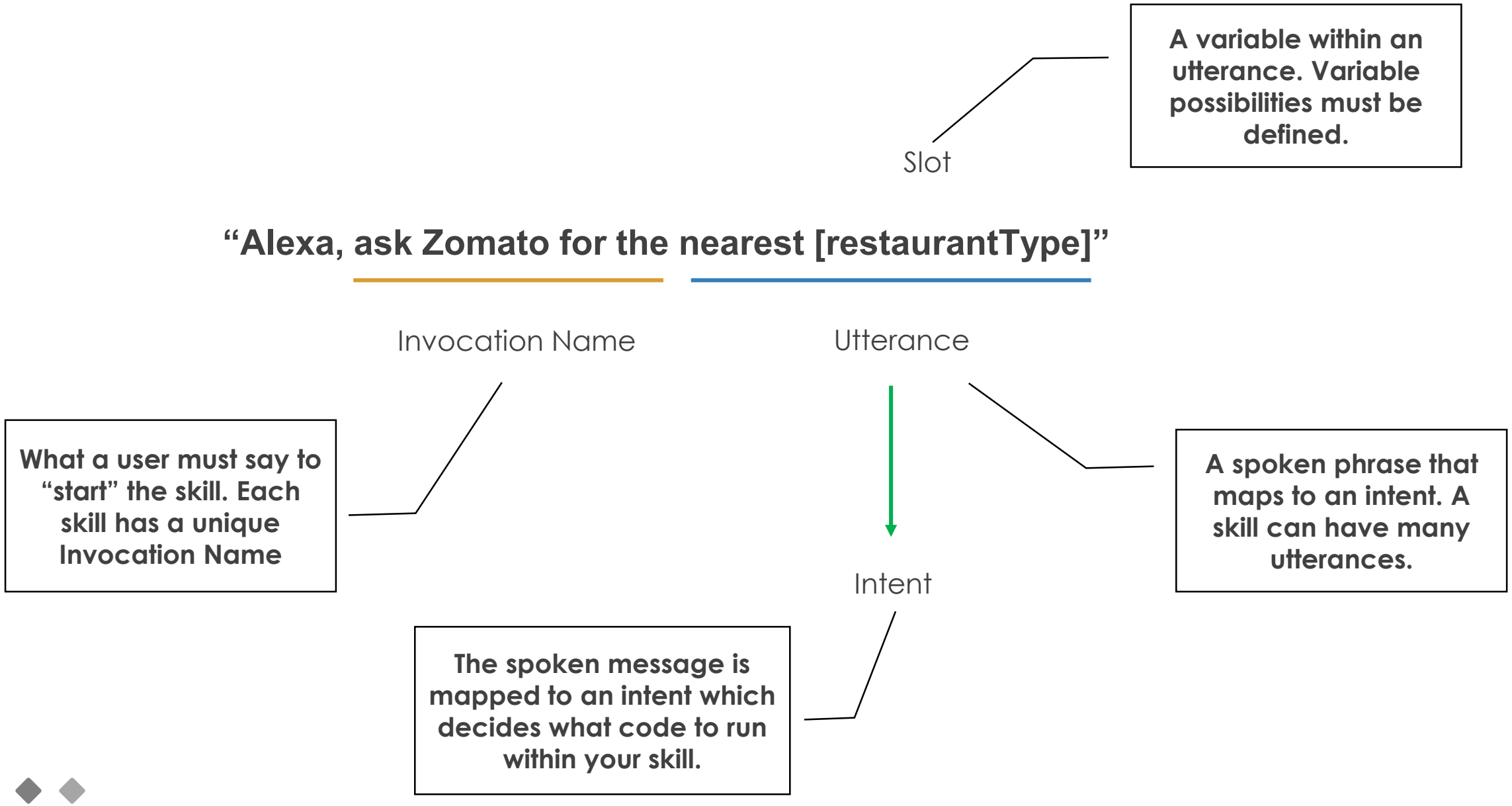
Creating an echo skill for voice control

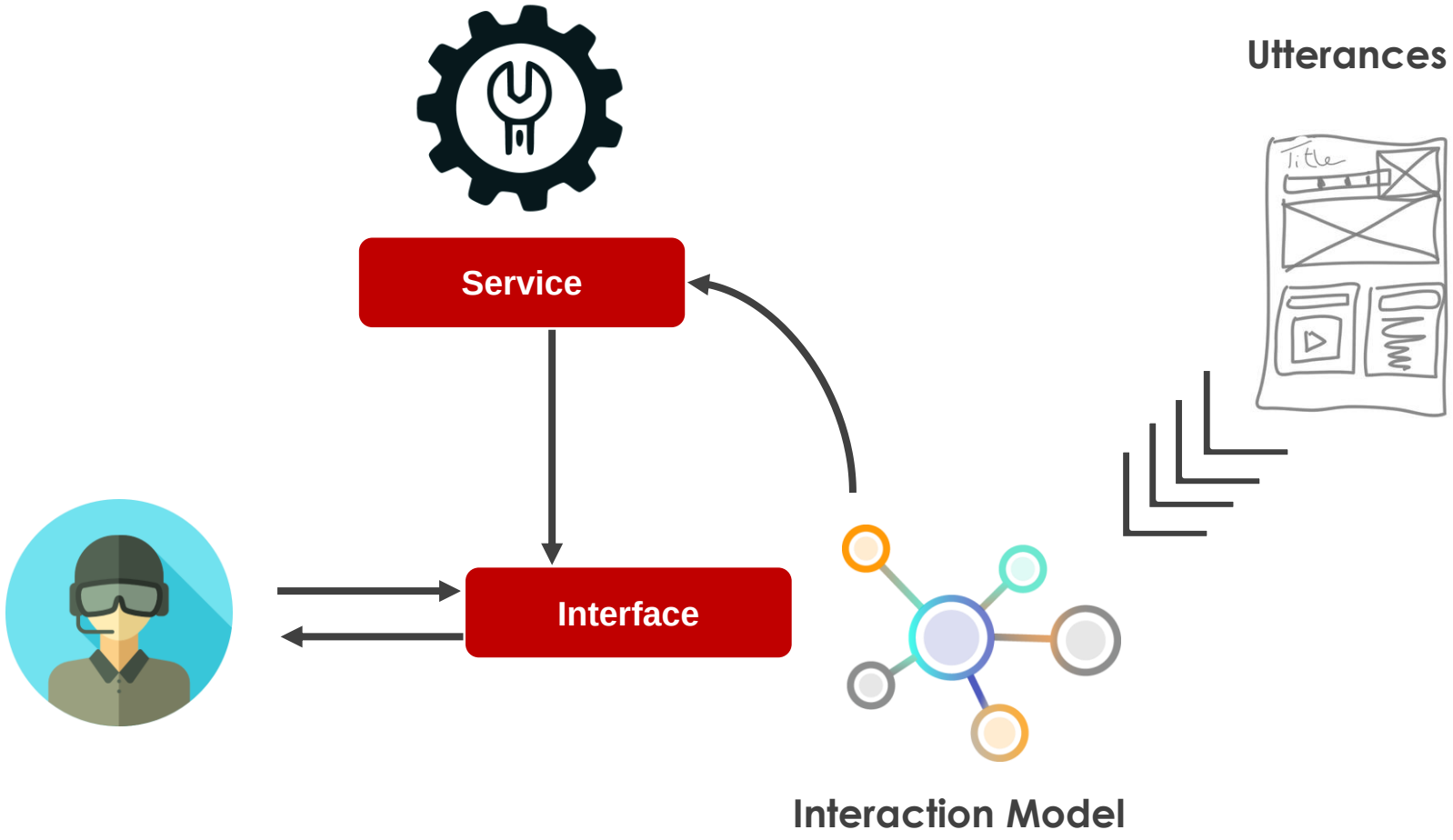


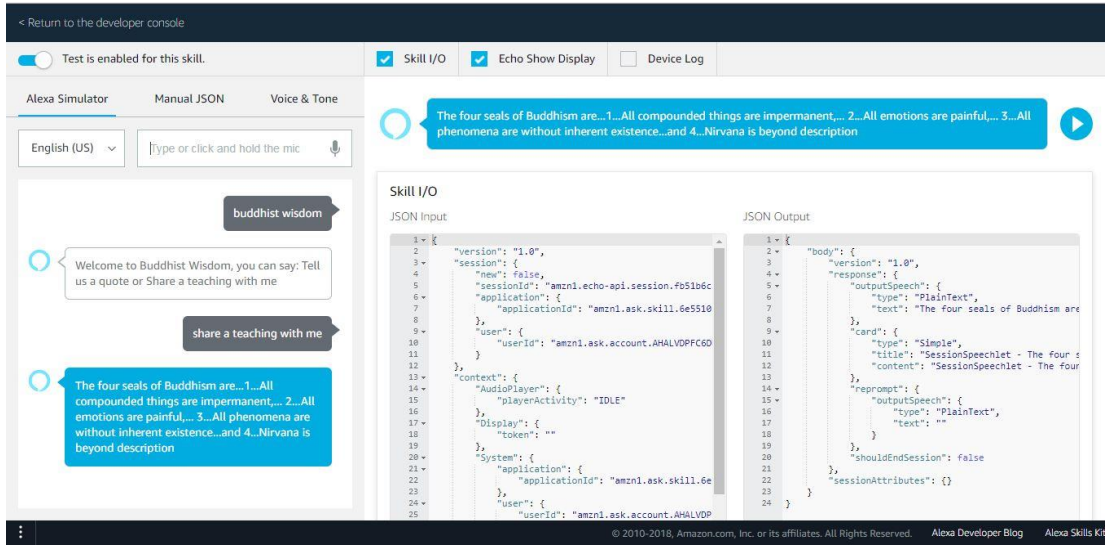
Smart Home Applications Code Demo

Setting up and Configuring Serverless Lambda Functions on AWS Demo

Create an echo assistant skill and deploy demo

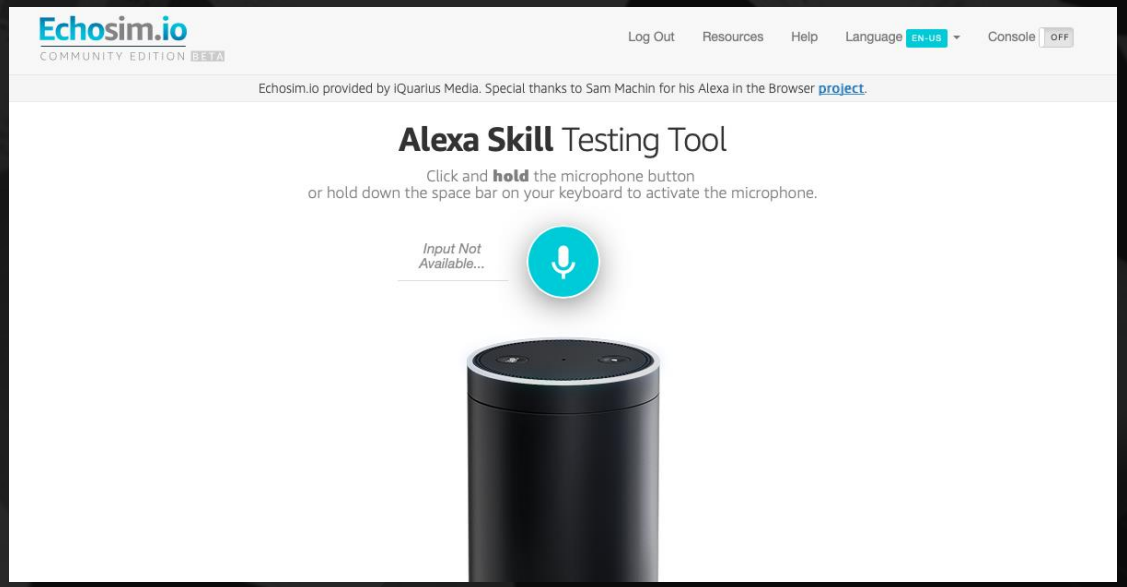

A variable within an utterance. Variable possibilities must be defined.
Slot
"Alexa, ask Zomato for the nearest [restaurantType]"
Invocation Name
Utterance
What a user must say to "start" the skill. Each skill has a unique Invocation Name
A spoken phrase that maps to an intent. A skill can have many utterances.
Intent
The spoken message is mapped to an intent which decides what code to run within your skill.


Utterances
Service
Interface
Interaction Model



Simulator

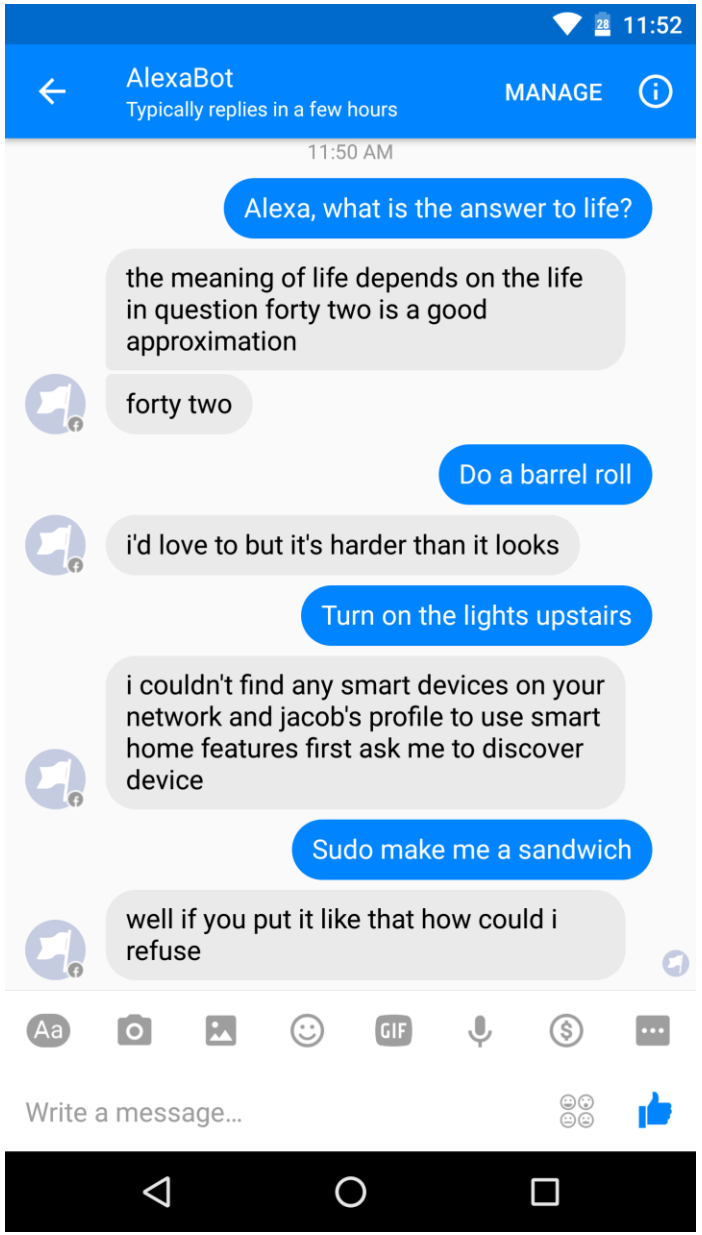
Echosim.io



What's **Next**?

AlexaBot

https://github.com/jacobajit/AlexaBot



Google Assistant on Your Amazon Echo

https://goo.gl/UEzRGo

Using data from fitness trackers . . .

- Continuous HR data – detect spikes, stroke patterns, anxiety
- Calorie estimation from step count, and alert when you need to eat.
- Alerting triggers based on thresholds
- Create APIs to connect other devices like glucometers, and Blood Pressure monitors.
- Publish data to a central store, with web visualization support.

The Future

Interactive Social Networks

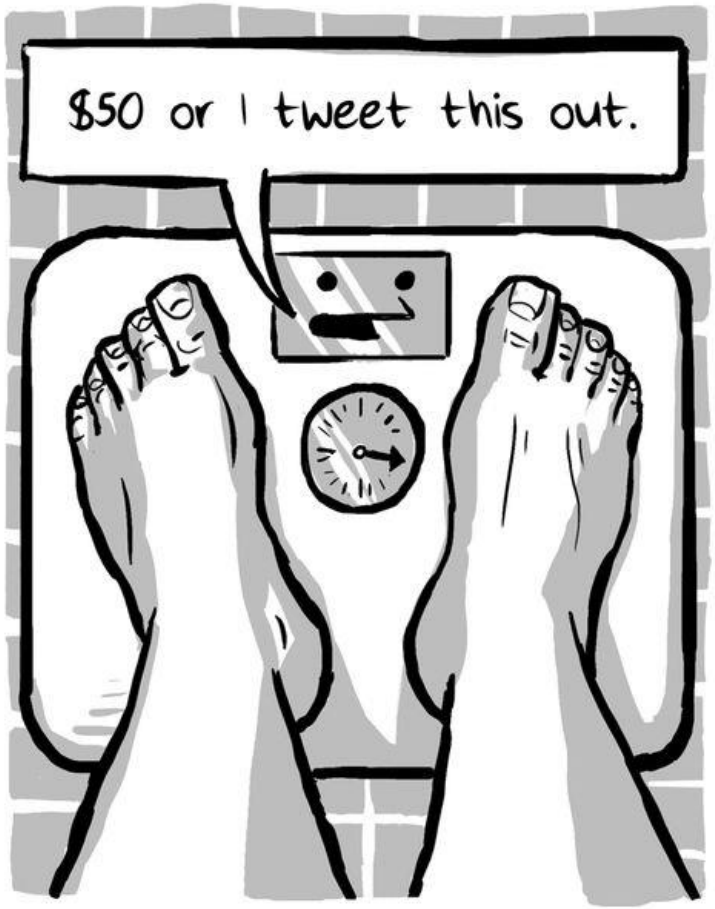
Interactive Idea boards

Voice Gaming

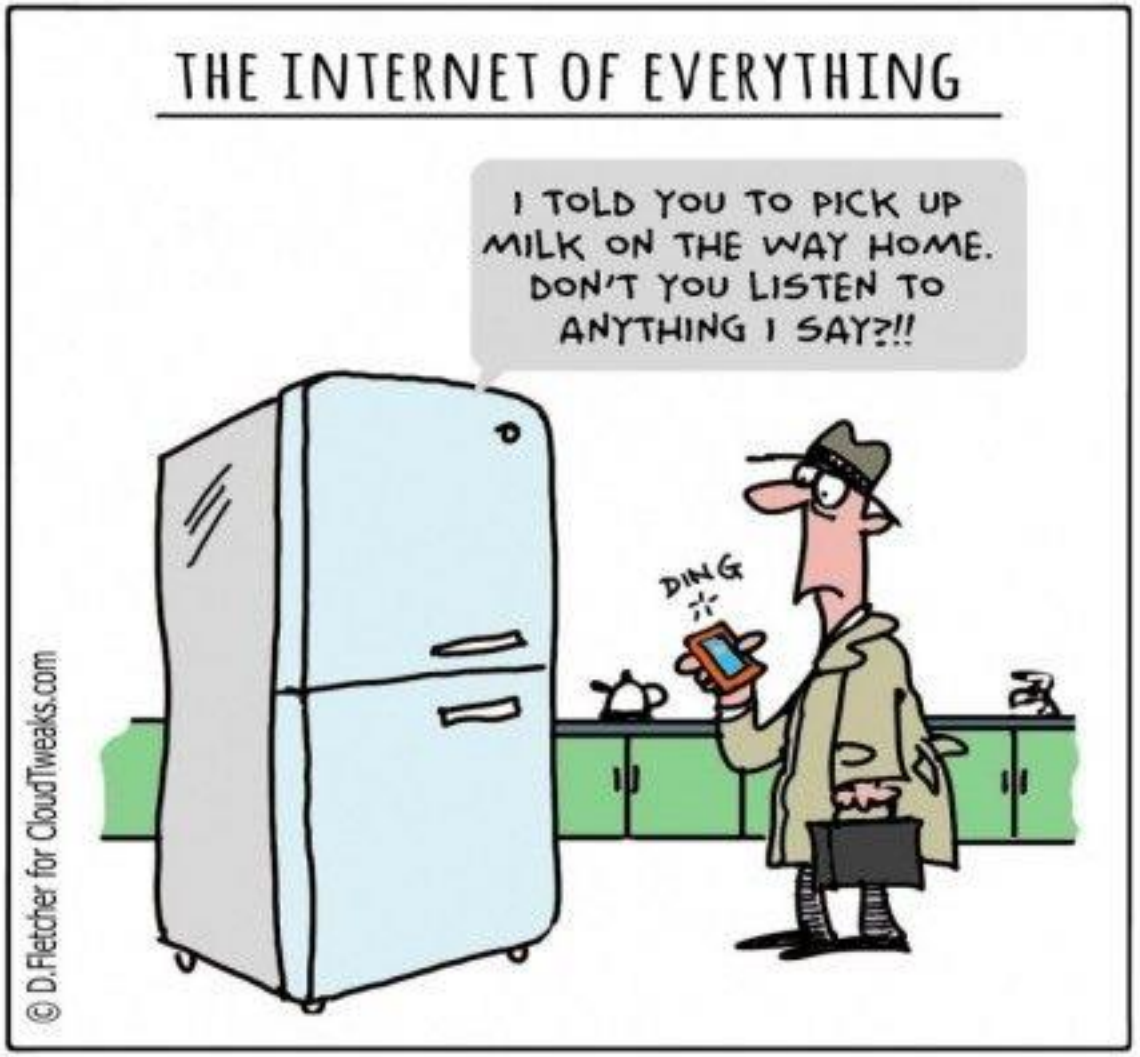
Crisis Alerting

Office Integration – Meeting room bookings, etc.

Always aware state!! (Creepy!!)



Smart Problems!



Headed here….


Starting another side-project, are we?
Yeah... Well no actually, I've nearly finished the other one so I just wanted to test out this other thing

THANKS!!