

Serverless 101

The What, Why and How of Serverless in the Real World





Aaron Williams

**Co-Founder, CEO
Funcatron**

 @_arw_

 /williamsaaron

 /in/aaronwilliams/

 aaron@funcatron.com



Follow Along or Follow-up:

 [/williamsaaron/serverless101-YUL](https://github.com/williamsaaron/serverless101-YUL)



Agenda

1. What the heck is “Serverless?”
2. Why is it gaining momentum? (Why should I care?)
3. How is it being used? (What are the categories of good use cases?)
4. How can I use it?
5. DEMO
6. What’s next?
7. Q&A



1. What the heck is “Serverless?”



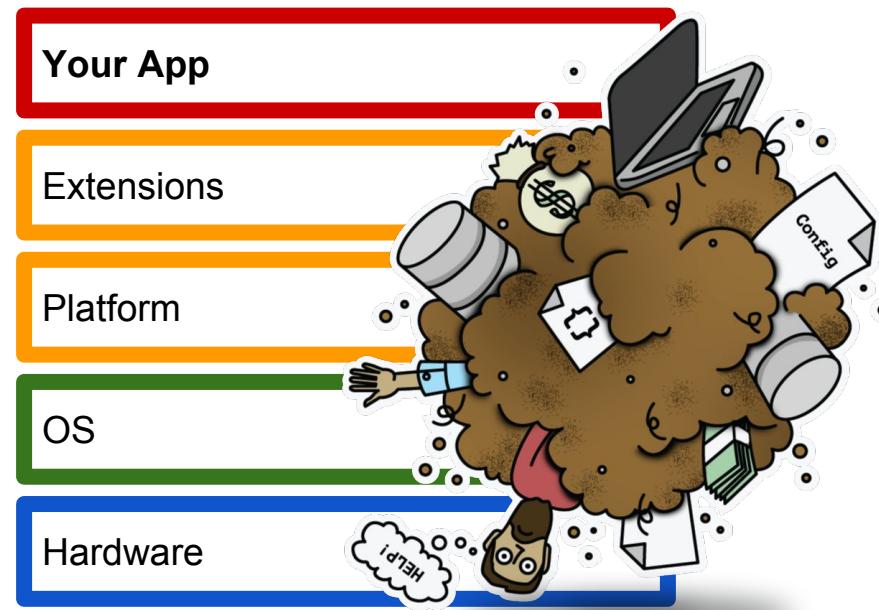


M A G I C ?!

Serverless History

It comes down to what you have to care about.

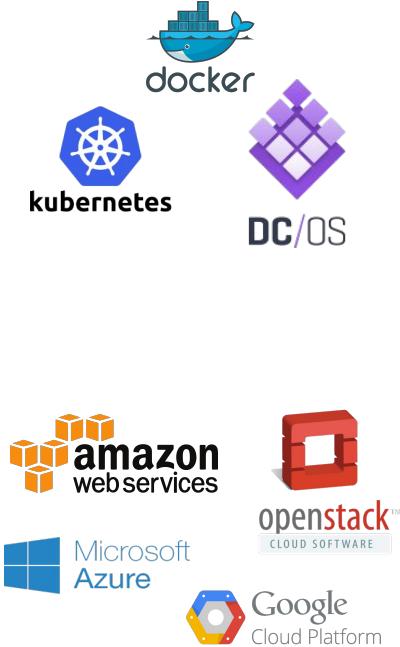
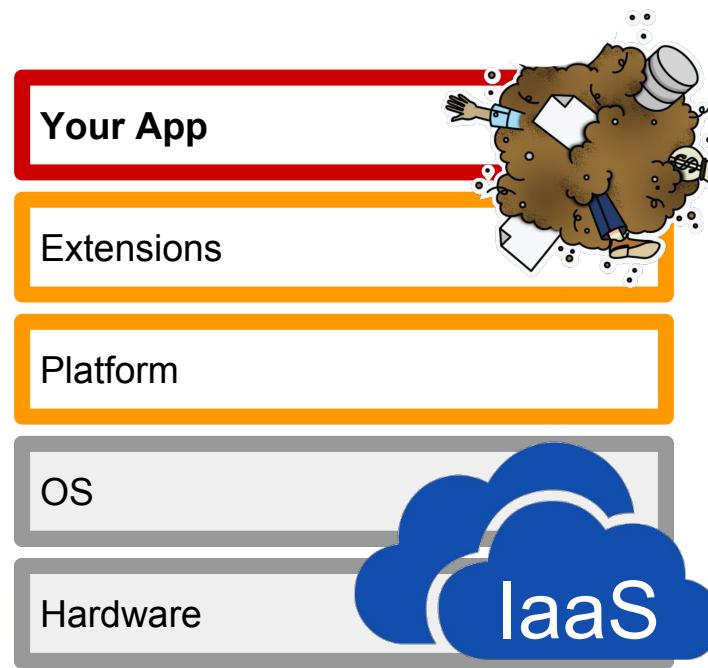
ERA #1 Pre-Cloud (the *DARK AGES*)



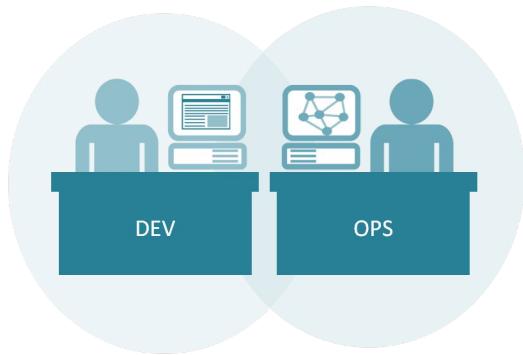
Serverless History

It comes down to what you have to care about.

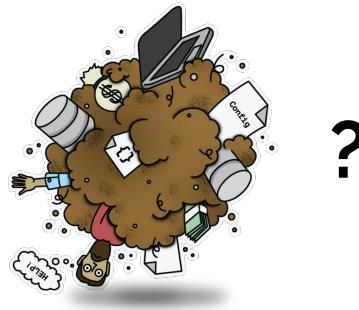
ERA #2
Cloud
+
Containers



A Quick Sidebar on “DevOps”



OR



We've traded more power for more responsibility (by which I mean scripting ... ugh)

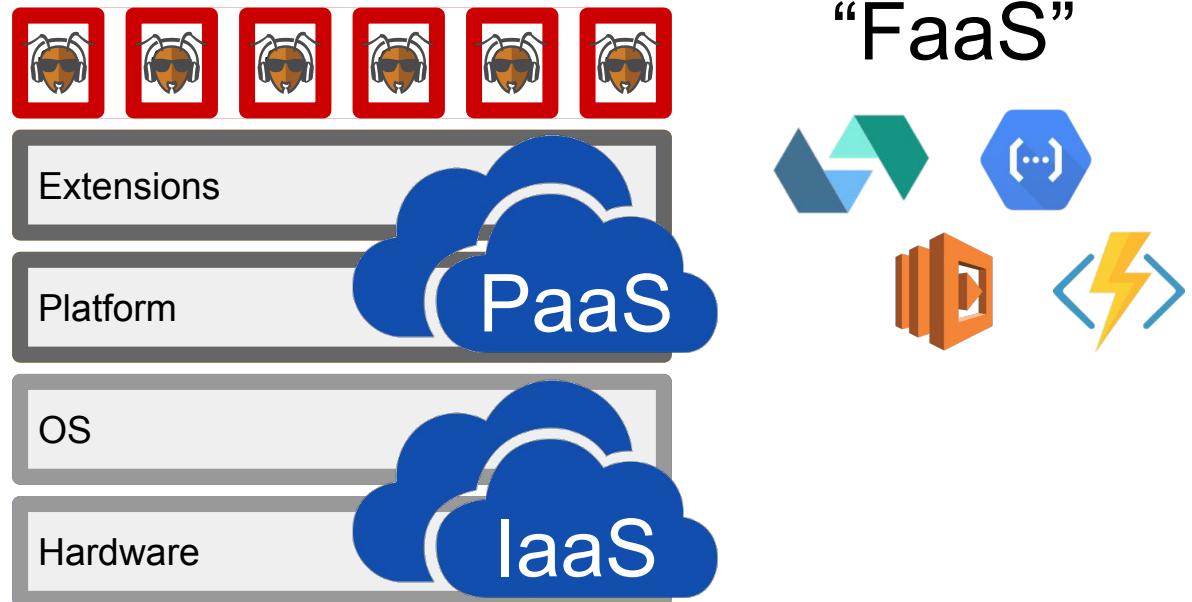
Developers and managers are starting to feel the pain of this at scale

Devs don't want to learn ops; they just want better interfaces to be better devs

Serverless History

It comes down to what you have to care about.

ERA #3
Serverless



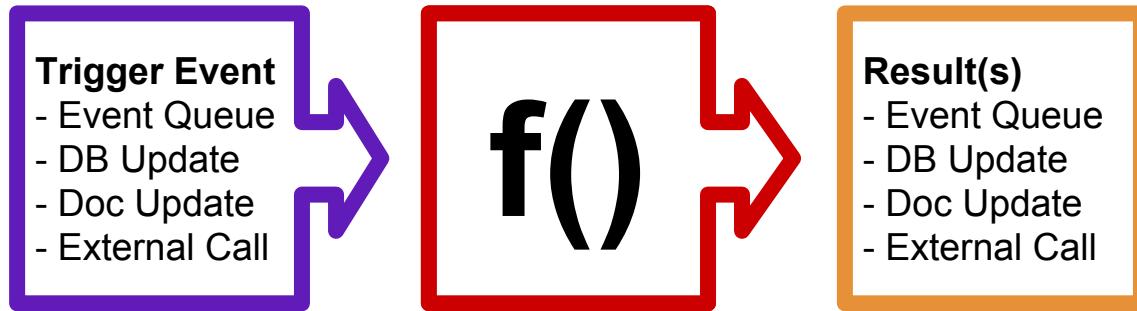
ANTZ!

Ants are:

- Excellent communicators and hard workers
- Abundant (the total weight of ants on earth equals the total weight of humans)
- Almost all female



Anatomy of a Function



Think events! The message queue is becoming the source of truth

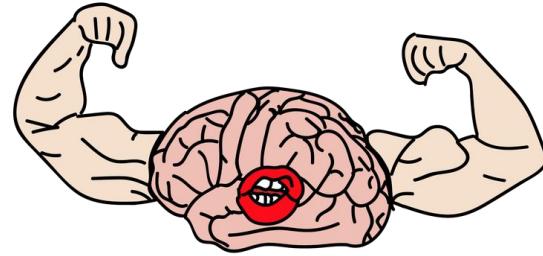
Functions are deployed independently, as each dev cycle requires

They're also run and scaled independently, as the event queue demands

Serverless is a Philosophy

What is **Serverless**?

1. Microservices architecture
2. Event-driven (“stateless”)
3. Agile teams development



Functions-as-a-Service (FaaS) is an offering, and every cloud has one

“**Serviceful**” is probably a better term ... but the “Serverless” ship has sailed



Serverlessconf

[Home](#)[Videos](#)[Playlists](#)[Channels](#)[About](#)

Serverlessconf Austin 2017 Talks

Serverlessconf Austin Talks



11:22

[Serverlessconf Austin '17
Introduction - Peter Sharski &](#)



34:23

[Serverlessconf Austin '17
Keynote - Austen Collins](#)



54:03

[Serverlessconf Austin '17
Keynote - John Gossman](#)



32:49

[The Story of a Serverless Startup
- Sam Kroonenburg](#)



[Serverless at Nordstrom - Rob Gruhl
by Serverlessconf](#)



[Serverlessconf Austin '17 Keynote - John Gossman
by Serverlessconf](#)

<https://www.youtube.com/channel/UCqIcVgk8SkUmve4Kw4xSlgw>

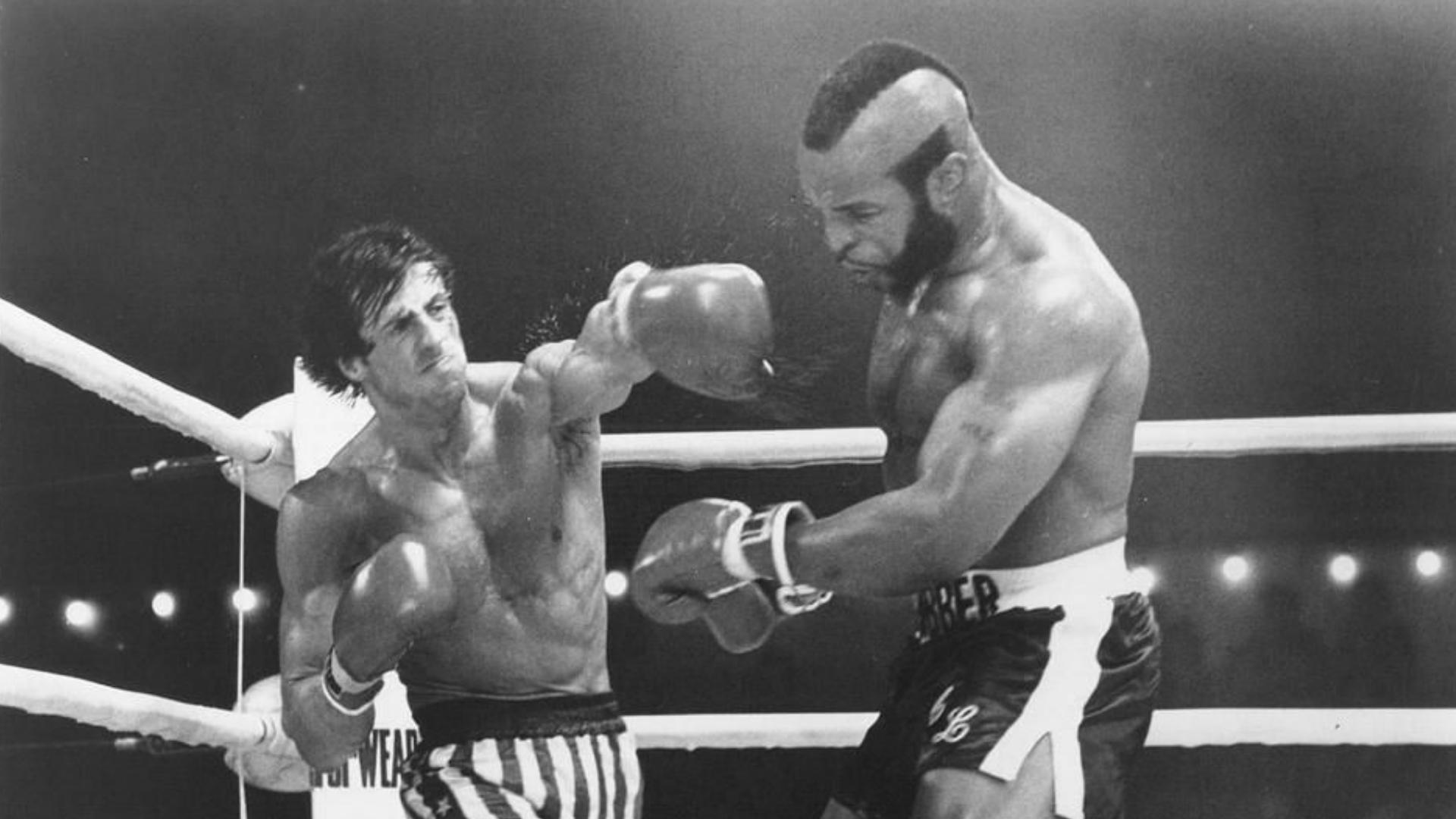
2. Why is it gaining momentum?



has
"Every business ~~will~~
become a software
business, build applications,
use advanced analytics and
provide SaaS services."

- Smart CEO Guy



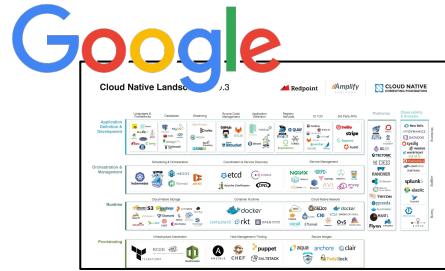


WEAR

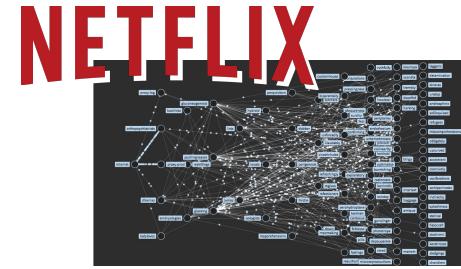
Business Trends



Hardware
Abstraction



Software
Automation



Microservice
Architecture

“Cloud Native”



37%

of IT decision makers have already
adopted serverless technologies

from Sept 2016

451 Research: Voice of the Enterprise



The Cost of Serverless

The principle: you're only billed for function execution (consumed vs. provisioned)

What it means:

- No unused capacity (datacenters build for 95% of peak usage; are lucky to reach 25% mean utilization)
- No scale-fails (the 5%)!
- Easier to optimize costs when they are tied directly to the activity of your app
- BUT: it is not automatically cheaper



SHOWDOWN!



<https://headmelted.com/serverless-showdown-4a771ca561d2>



Serverless Cost Calculator

Peter Sbarski and the A Cloud Guru Team.

Grab our book [Serverless Architectures on AWS](#).



© Peter Sbarski ([A Cloud Guru](#)) | Design: [HTML5 UP](#)

Serverless Cost Calculator (beta)

Calculating cost for AWS Lambda, Azure Functions, Google Cloud Functions, and IBM OpenWhisk

100000000

Number of Executions

90

Estimated Execution Time (ms)

640MB

Memory Size

True False

Include Free-Tier

True False

HTTP Requests

Vendor	Request Cost	Compute Cost	Total
AWS Lambda	\$20.00	\$104.19	\$124.19
Azure Functions	\$20.00	\$100.00	\$120.00
Google Cloud Functions	\$40.00	\$115.63	\$155.63
IBM OpenWhisk	\$0.00	\$106.25	\$106.25

<http://serverlesscalc.com/>

3. How is it being used?



Top Use Cases

Functions that are event-driven

file updates and notifications; bots; IoT

Functions that need to scale

payroll-related tasks; ETL / data pipeline tasks

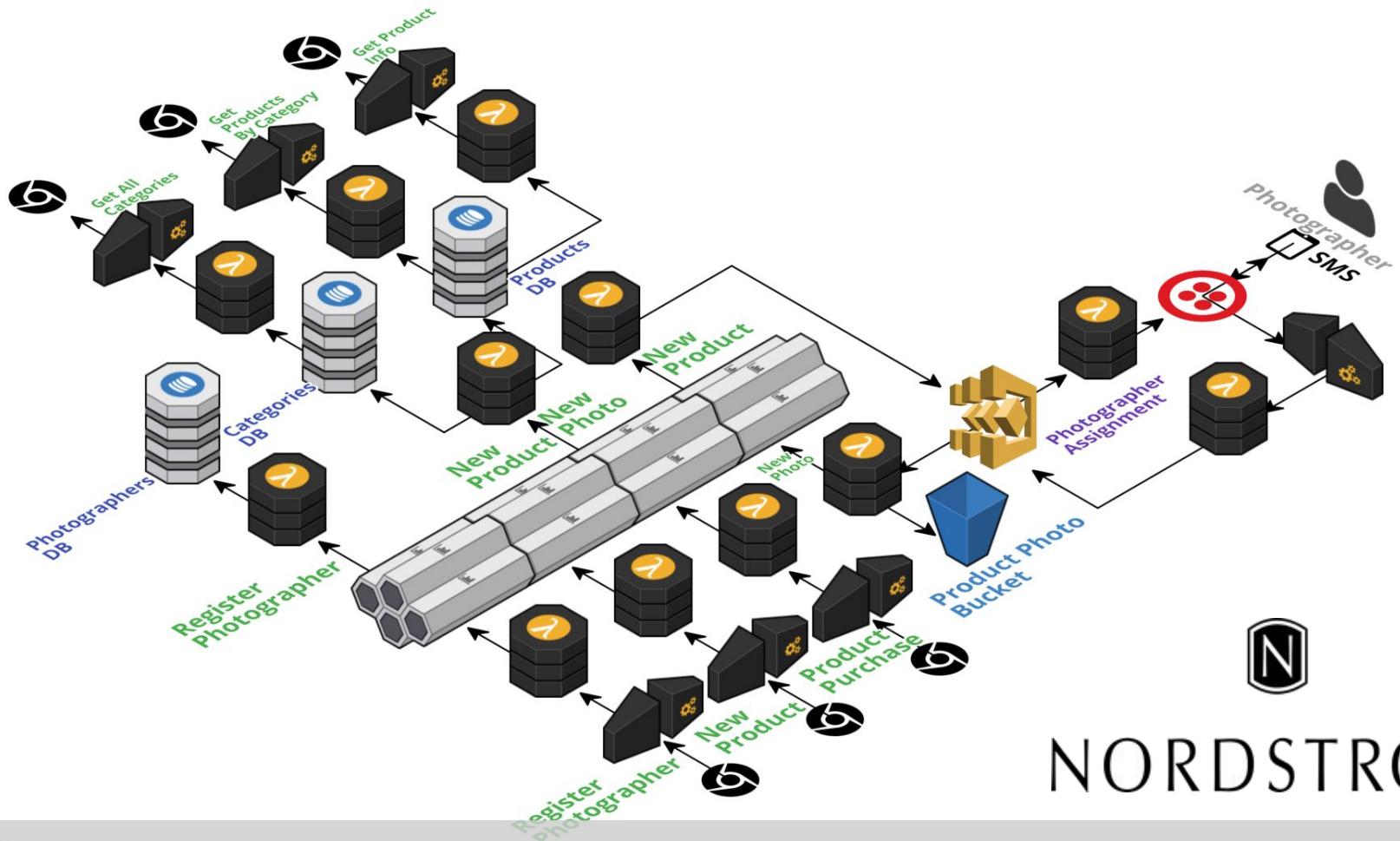
Functions that you want abstracted from another service

image manipulations; credit checks

Functions that are not sensitive to latency (so, not whole web sites, generally)

backups; generating reports; other housekeeping





NORDSTROM



4. How can I use it?



The Dreaded Mudball



The Cloud Native Transformation



Low-risk Digital (Cloud Native) Transformation



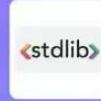
Security Services



Cloud Providers / Serverless Platforms



Serverless Libraries



Databases



Opensource Serverless Frameworks



Monitoring Services



Serverless Frameworks

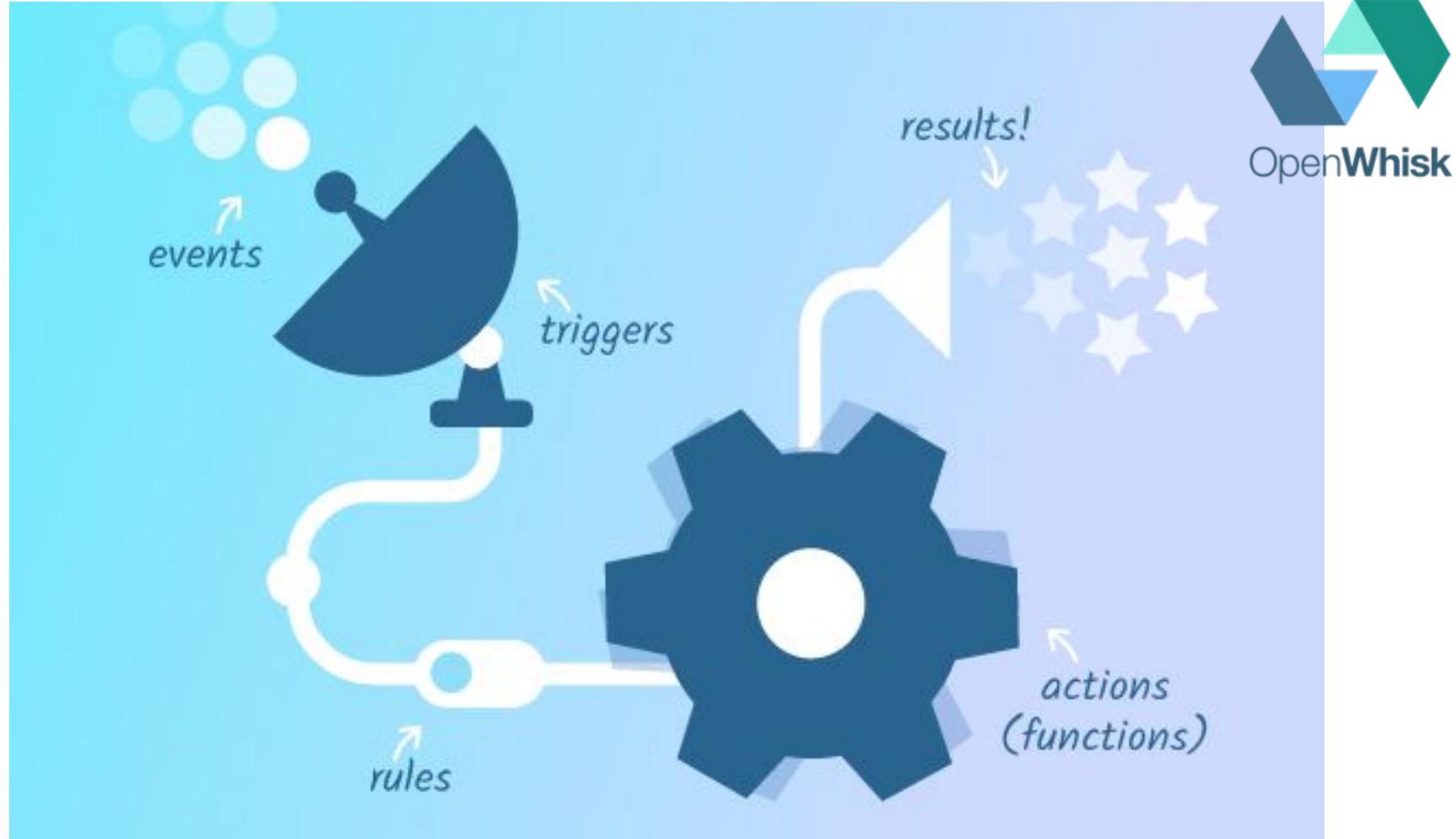


Lever OS ΔAPEX ZAPPIA MULTICOLOUR EFFE KAPPA ZAPIER



5. DEMO TIME!





Setup

Create a BlueMix account : <https://console.ng.bluemix.net/registration/>

Download the OpenWhisk CLI: <https://console.ng.bluemix.net/openwhisk/>

** Make sure you set your namespace and auth key.

Clone the sample code: <https://github.com/williamsaaron/serverless101-YUL>

First, let's write some Hello World code just to familiarize ourselves with the CLI ...

A Weather Bot for Slack

Take aways:

- Getting started requires no config (zero ops)
- Almost all of the code we write will be specific for our app
- All clouds offer a free tier



Weather Bot Flow



Two scripts, one sequence, and some simple Slack config

Let's do it!

Quick Recap

I know ... **bots are lame**, but they're a good 101-level intro. Review the use cases for better, more complete examples.

We did ~0% ops, and ~100% of our code was specific to our app.

OpenWhisk is not unique, Lambda from AWS, Cloud Functions from Google and Azure Functions from Microsoft all have the same basic interface (CLI and GUI)

(If you have suggestions to improve the demo, things you'd like to see more of, please reach out!)



6. What's next?



Some Predictions

Latency will become a non-issue

Cost savings will be the primary driver through 2018,
but developer agility will be the long term value

Team work will be a limiting factor

Habs win the cup in 2018?!



7. Any questions?





THANK YOU!

Aaron Williams

 @_arw_

 /williamsaaron

 /in/aaronwilliams/

 aaron@funcatron.com

