Where should I run my code?

Katie McLaughlin
Developer Advocate, Google Cloud

Google Cloud

Where should I run my code?

It depends.

Thank you!

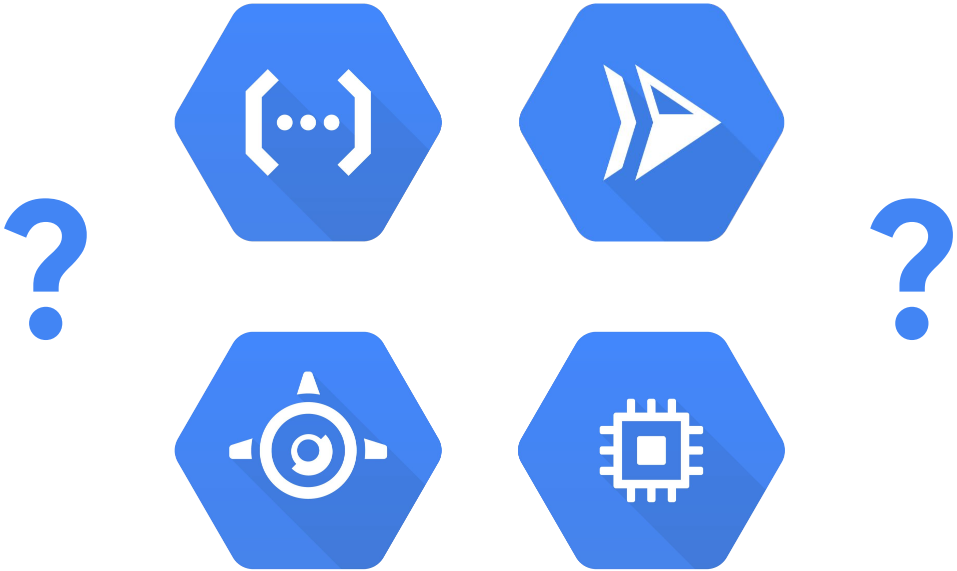
@glasnt

Google Cloud

Okay, fine.

~~It depends.~~

I'm a sysadmin.
I'm not your sysadmin.



What is the "right choice"?

The right choice
is what is right for you.

Cloud Functions

Cloud Run

App Engine

Compute Engine

Cloud Functions

Managed code execution

01

What is Cloud Functions?

A serverless environment to build and connect cloud services

Event driven - connect Cloud services

- via Cloud Pub/Sub, Cloud Storage, HTTP requests & more

Serverless

- Fully managed execution environment
- Pay only for what you use
- Autoscales with usage

What do you have to worry about?

Events

Function Definitions

What is managed for you?

Everything else.

Good fit? Constraints?

Good fit

Serverless

Using Cloud Pub/Sub, Cloud Storage

Don't want to think about runtime environment

Data transformations (ETL)

HTTP glue

Constraints

Function level granularity

Must interact via events

Specific language runtimes:

Node.js, Python, Go.

Why Cloud Functions?

Don't have to even think about servers

Pay only for what you use

Simple developer experience (deploy functions not apps)

02

App Engine

Managed Runtimes

What is App Engine?

App Engine goal: let **developers focus on app code**

No servers to manage, scale up fast, scale down to zero

No patches/updates

Versioning

Traffic splitting

App Engine Standard 2nd Generation Runtimes

Open-source, idiomatic experience

Use any extension, binary, or framework

Supported Runtimes

Python, Java, Node.js, PHP, Ruby, Go

Use Google Cloud APIs directly

What do you have to worry about?

Code

HTTP Requests

Versions

What is managed for you?

Everything else.

Good fit? Constraints?

Good fit

HTTP/S request-response

Stateless serving applications

Scaling to high traffic

Constraints

Standard

- Runtimes for Python, Java, Node, PHP, Ruby, Go
- HTTP/S only

Flexible runtimes

- Inherit Docker constraints
- Not best for very low traffic sites

Why App Engine?

Developers focus on code

Optimized for web serving workloads

Especially great for variable load

App Engine currently serves more than 100B requests per day

Cloud Run

Managed containers

03

What is Cloud Run?

Input: ***container image*** with code listening on $PORT for HTTP

SSL termination
HTTP requests to running containers
1-80 concurrent requests, configurable
Scale to zero
Pay only for actual CPU & memory used

out of beta 2 days ago!
Available in europe-west1

What do you have to worry about?

Code in a container image

Building the image

Listening on a $PORT *

What is managed for you?

Everything else.

* https://cloud.google.com/run/docs/reference/container-contract

Good fit? Constraints?

Good fit

Stateless

HTTP request-response workloads

Scale: way up, down to zero, bursty

Specific runtime requirements

Constraints

Must use containers

Decide on build process (Cloud Build, etc)

Why Cloud Run?

Serverless developer experience

Decide on operations model at deployment time

- Change with a re-deploy

Runs anything that can run in a container and listens to HTTP.

Open: run same containers on your machines

Compute Engine

Virtual Machines

04

What is Compute Engine?

Virtual machines and networking

Per Virtual Machine:

- up to 160 vCPUs
- up to 3,844 GB RAM
- up to 64 TB of disk
- optional GPUs, TPUs

Each specification independently configurable

What is Compute Engine?

Virtual machines and networking

Start quickly, ~20 seconds to user code running

Pre-built images, or create your own

- Debian, CentOS, CoreOS, Ubuntu, RHEL, SUSE
- also Windows Server Datacenter!

What do you have to worry about?

Your software

Operating system / disk images

CPU, RAM, Disk

Networking - Firewall, Load balancers, VPNs

What is managed for you?

Everything else.

@glasnt

Good fit? Constraints?

Good fit

Existing systems (lift and shift)

1:1 container:VM mapping

Specific OS / kernel required

License requirements

Running databases

Constraints

Scaling speed floor

~20s per VM, 1000s in ~60s

Decide how to handle software updates

Google Cloud

Why Compute Engine?

Consistency - ask for 10 VMs, you get 10 good VMs

Custom VMs - choose RAM and CPU, GPUs

Live disk resize - also any size disk on any machine

Billing - Sustained use discounts, preemptible VMs, per minute billing

Live migrations - your VMs stay up longer

Architecture - good fit for most existing systems

Why Compute Engine?

Google Cloud services run on it, too:

- Cloud SQL
- GKE
- Cloud Dataflow
- Cloud Dataproc
- Cloud Build
- and more...

But hold on there, Katie

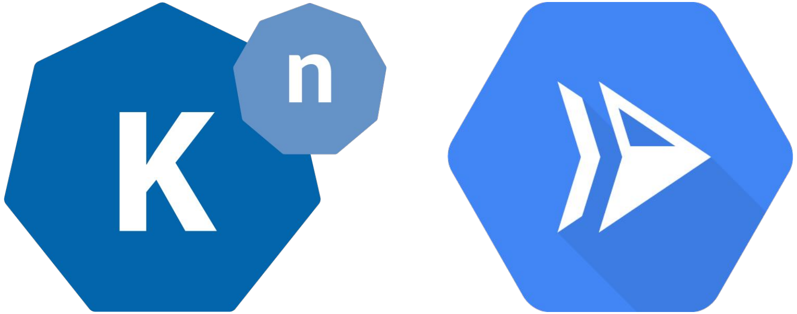
Why haven't you mentioned Kubernetes yet?

Google Cloud

Oh, but I have.

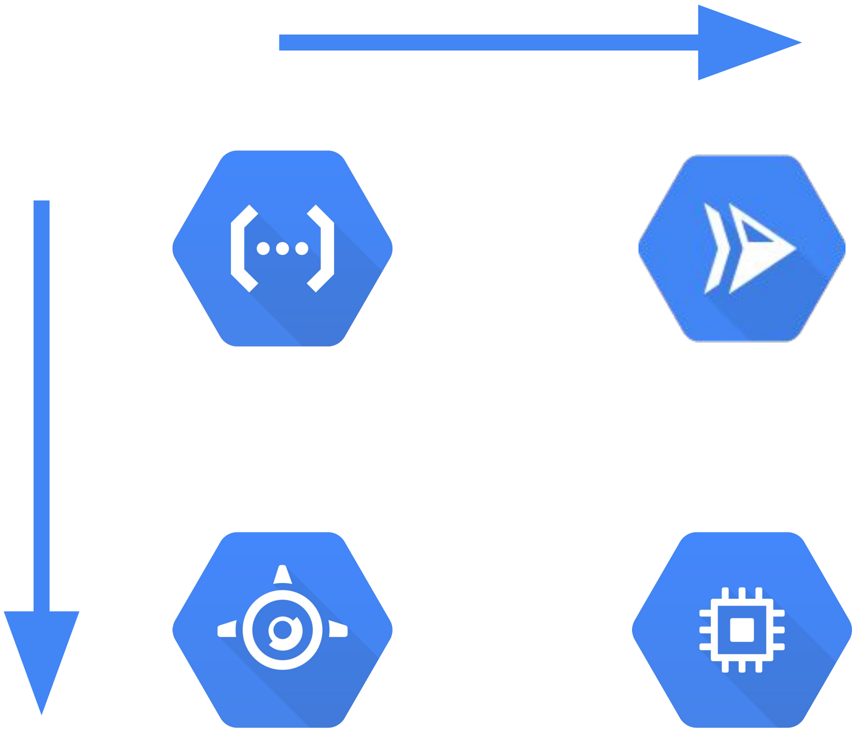
Knative

open source **building blocks**
for serverless on Kubernetes



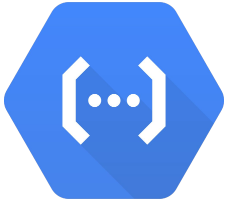
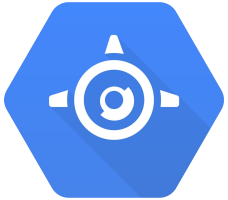
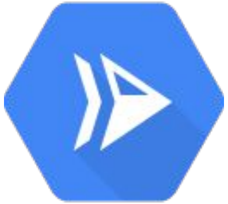
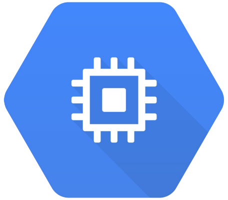
Knative directly powers **Cloud Run**

Found a good fit already?



What choice do I have?

What choice do I have?

				
Packages	Some	Any	Any	Any
Language	Some	Some	Any	Any
Operating System	None	None	Any	Any
Hardware	None	None	None	Any

Alternatively:

What do I have to worry about?

What do I have to worry about?

Packages	Worry	Worry	Worry	Worry
Language	Managed	Managed	Worry	Worry
Operating System	Managed	Managed	Worry	Worry
Hardware	Managed	Managed	Managed	Managed

The right choice is what is right for you.

Do
one thing

Do
many things

Do
anything

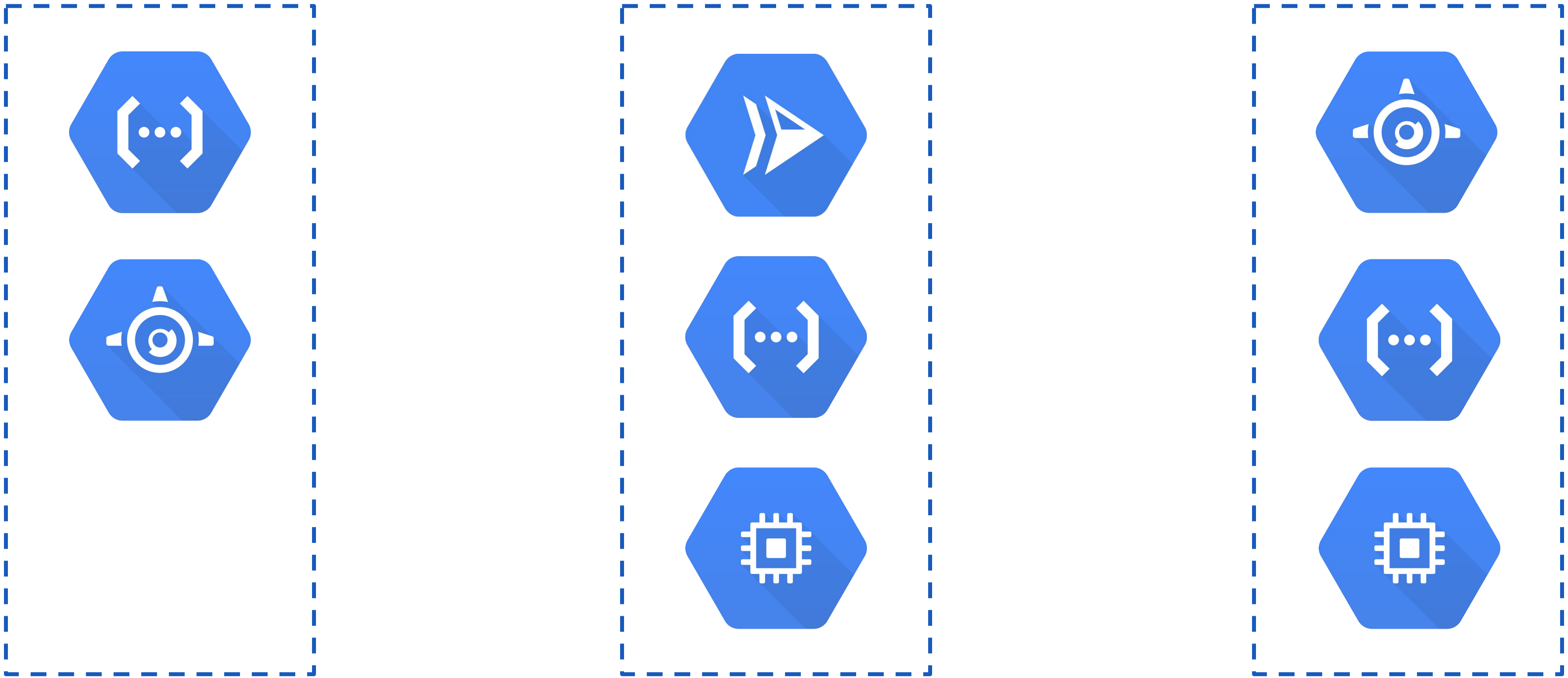
Do
everything

https://twitter.com/steren/status/1126310362494787584

The right choice *might* be more than one.

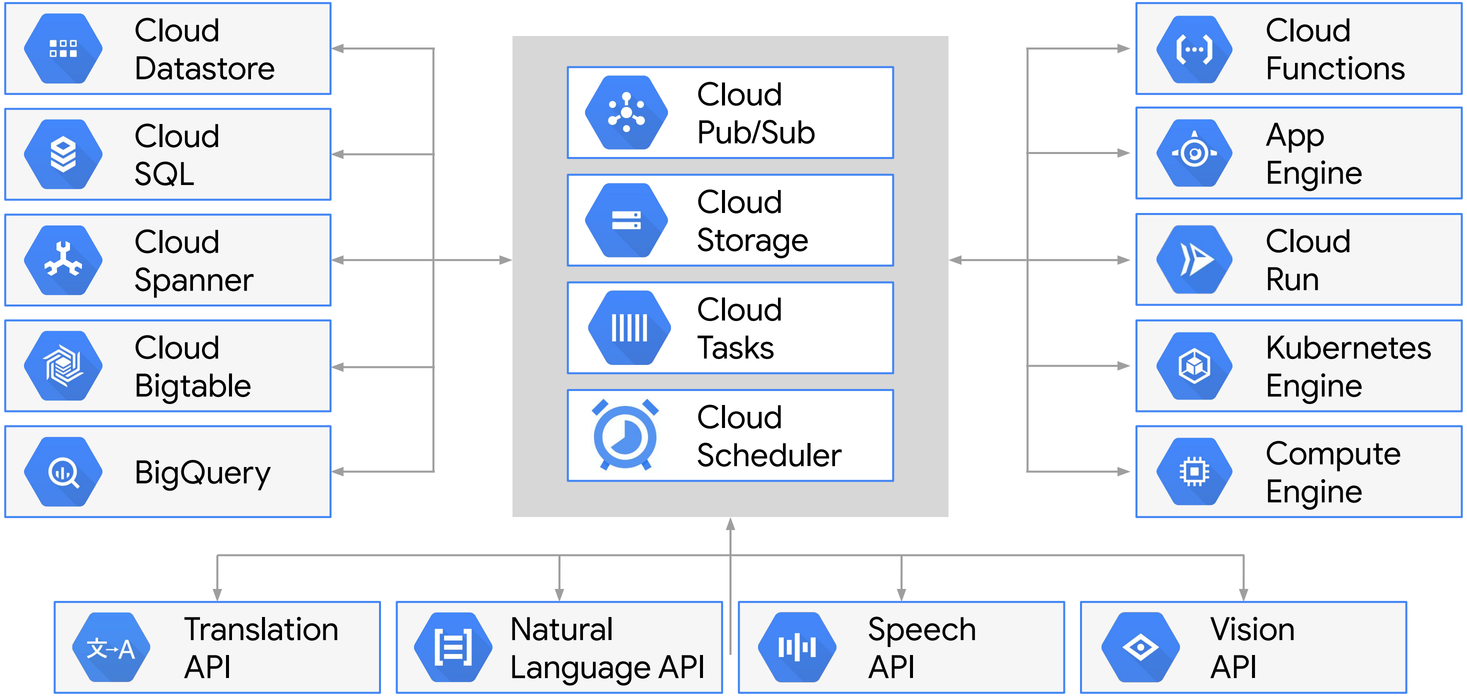
Use them together!

Most systems use multiple services together



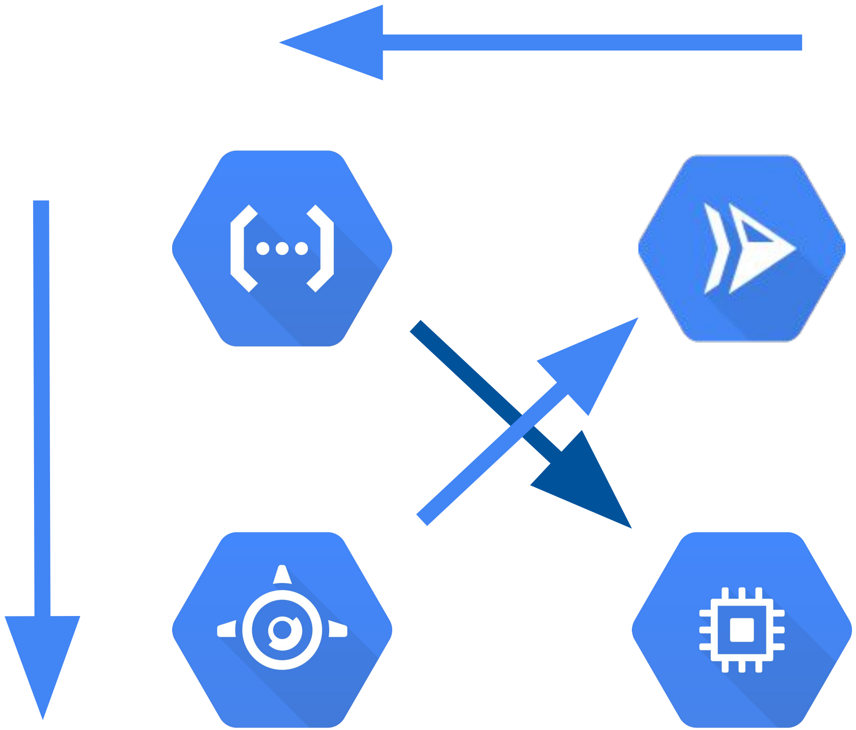
@glasnt

Complex systems could use many more component!



Google Cloud

...or move up or down abstraction levels



The right choice
is what is right for you.

Thank you!

@glasnt

Questions

Thank you!

@glasnt