Cloud Computing
Cloud Applications

CIS437

Erik Fredericks // frederer@gvsu.edu

A question

What makes a cloud application different from a normal application?

And further

Are there different types of cloud applications?

Limitations of "local" apps

...remember the advantages of cloud computing?

- Fault tolerance
- Scalability
- Networking
- Federation
- Computing at scale
- ...

First, some things to consider before we dip into *aaS

APIs
Availability zones
Fault tolerance
Migration
Monitoring
Federation
Elasticity
Architectures

i.e., important things to consider in your cloud app

APIs

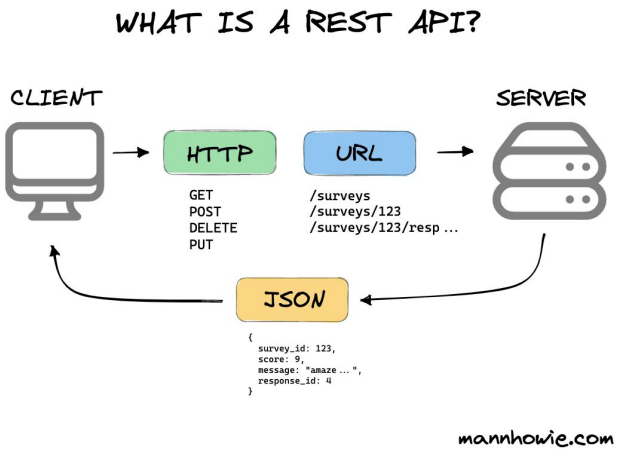
What is an API (other than something we'll talk about more when we go over serverless/microservices)?

A method for accessing services programmatically

- Could be a human or a program!

Why important to cloud?

- Ability to write code to automate *everything* in the cloud!




```
$ aws ec2 run-instances \
    --image-id ami-1a2b3c4d \
    --count 1 \
    --instance-type c3.large \
    --key-name MyKeyPair \
    --security-groups MySecurityGroup
```

(a)

```
$ openstack server create --flavor 1 --image 397e713c-b95b-4186-ad46-6126863ea0a9 \
  --security-group default --key-name KeyPair01 --user-data cloudinit.file \
  myCirrosServer
```

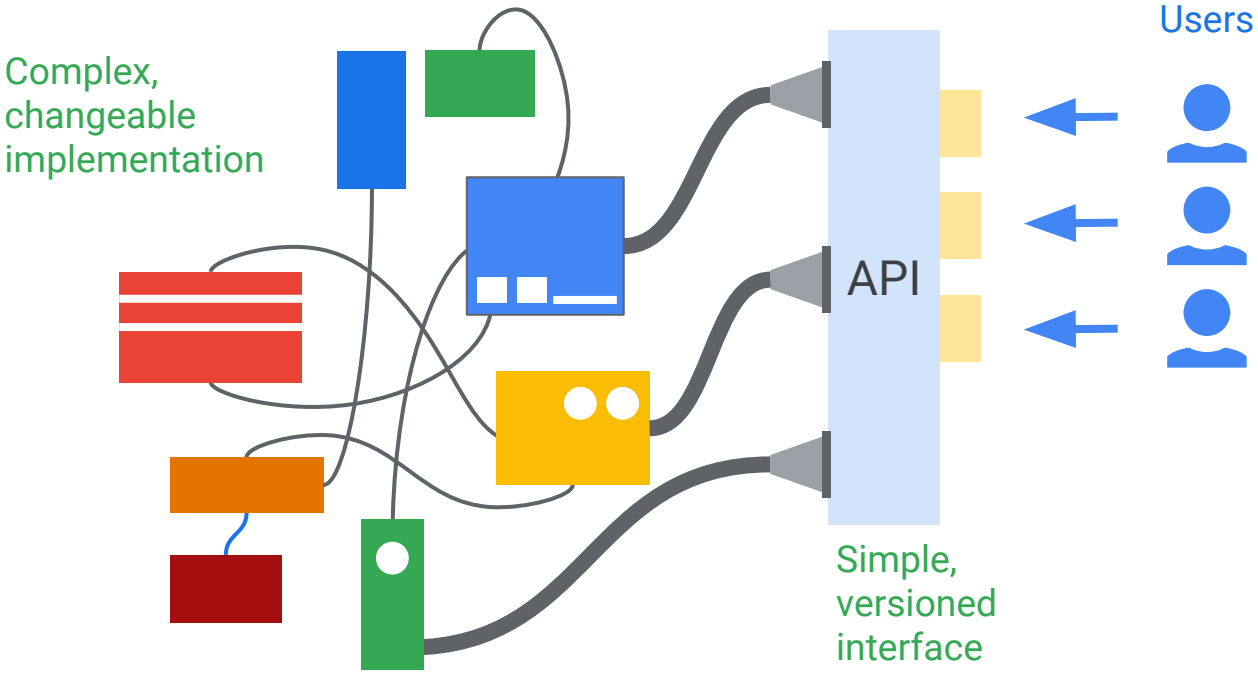
(b)

```
$ gcloud compute instances create "my-new-instance" \
      --zone="us-west1-b" \
      --image-family="tf-latest-cu92" \
      --image-project=deeplearning-platform-release \
      --maintenance-policy=TERMINATE \
      --accelerator="type=nvidia-tesla-v100,count=8" \
      --machine-type="n1-standard-8" \
      --boot-disk-size=120GB \
      --metadata="install-nvidia-driver=True"
```

(c)

Figure 3.1: How to run a new VM by using the command-line client software provided by (a) AWS, (b) OpenStack and (c) GCP respectively.

APIs hide the details and enforce contracts



Google Cloud

Availability zones

or, where does your provider have service?

Compute/serve near your client
- Faster delivery / quality of service

Regions can have multiple zones as well!

https://cloud.google.com/about/locations#lightbox-regions-map
https://aws.amazon.com/about-aws/global-infrastructure/?p=ngi&loc=1

Fault tolerance

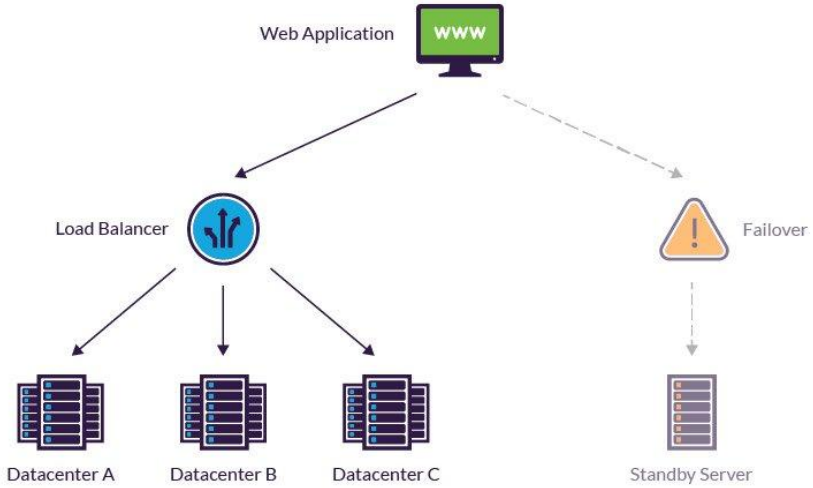
There are a *lot* of resources available

- What do we do if one fails?
  - e.g., a virtual machine crashes, or a zone goes down

Services need to be configured to *fail nicely*

- E.g., a mirror picks up the slack (failover)
  - Another instance is automatically spun up
  - etc.

Could be provided automatically by the provider or configured by the designer



Migration

May need to move from one service to another

- E.g., one VM needs to be moved to another
  - Because of availability, new services, etc.

Hot (live) migration

- Services still "on" while migration happens

Cold migration

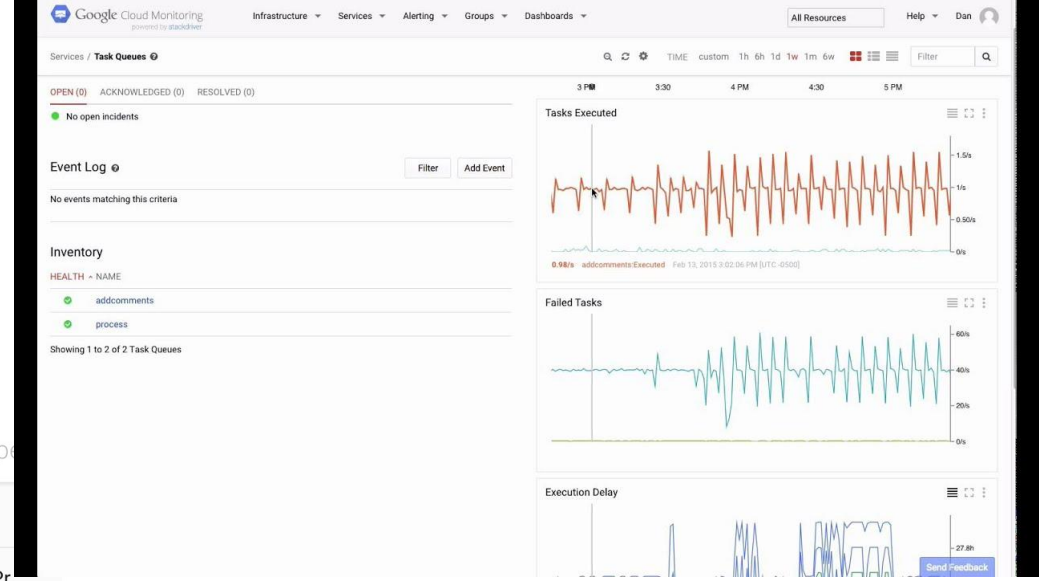
- Services "off" while migration happens

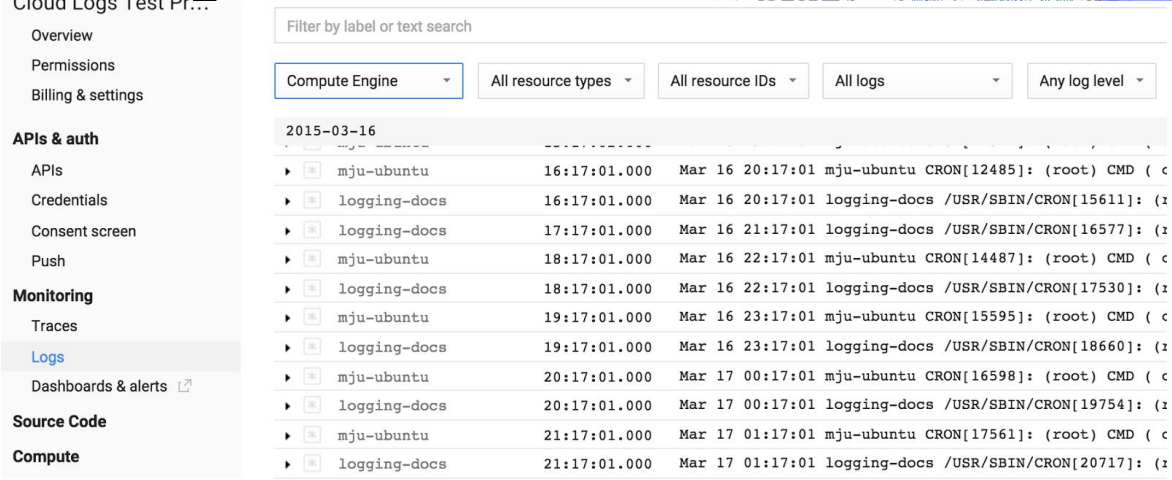
Why one over the other?

Monitoring

KEEPIN' AN EYE ON THINGS

- Why on earth would we want to monitor our services?
- What can we do if things *go wrong?*





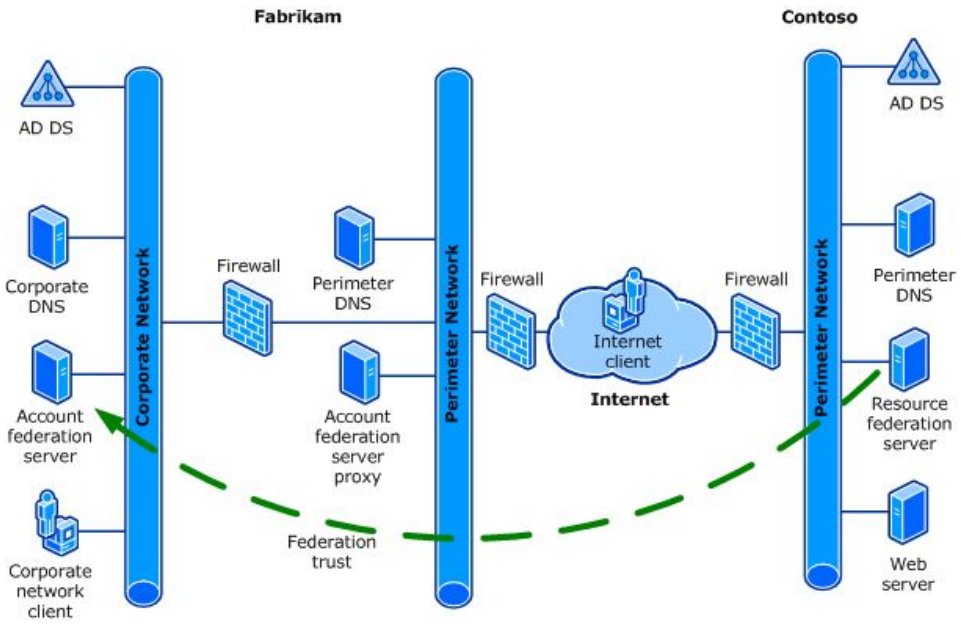
Federation

Interoperability of different providers

Think:
- Microsoft Active Directory
- Mastodon / Lemmy

Is this ... something that most cloud providers allow?

What do you think "lock-in" means?



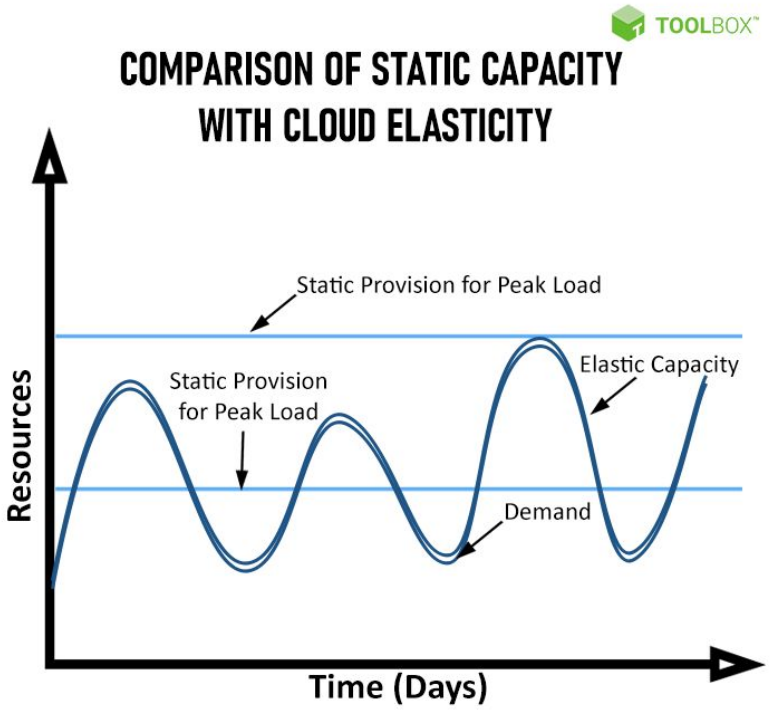
Elasticity

This is what gives us scalability

- Detect if workloads are too high and scale up/down as needed
- Relies on monitoring and automated triggering!

Think of your programs you've made so far though...

- How in the world could you automatically scale them up/down *at run time?*



Architecture

"Common"
(from the book)

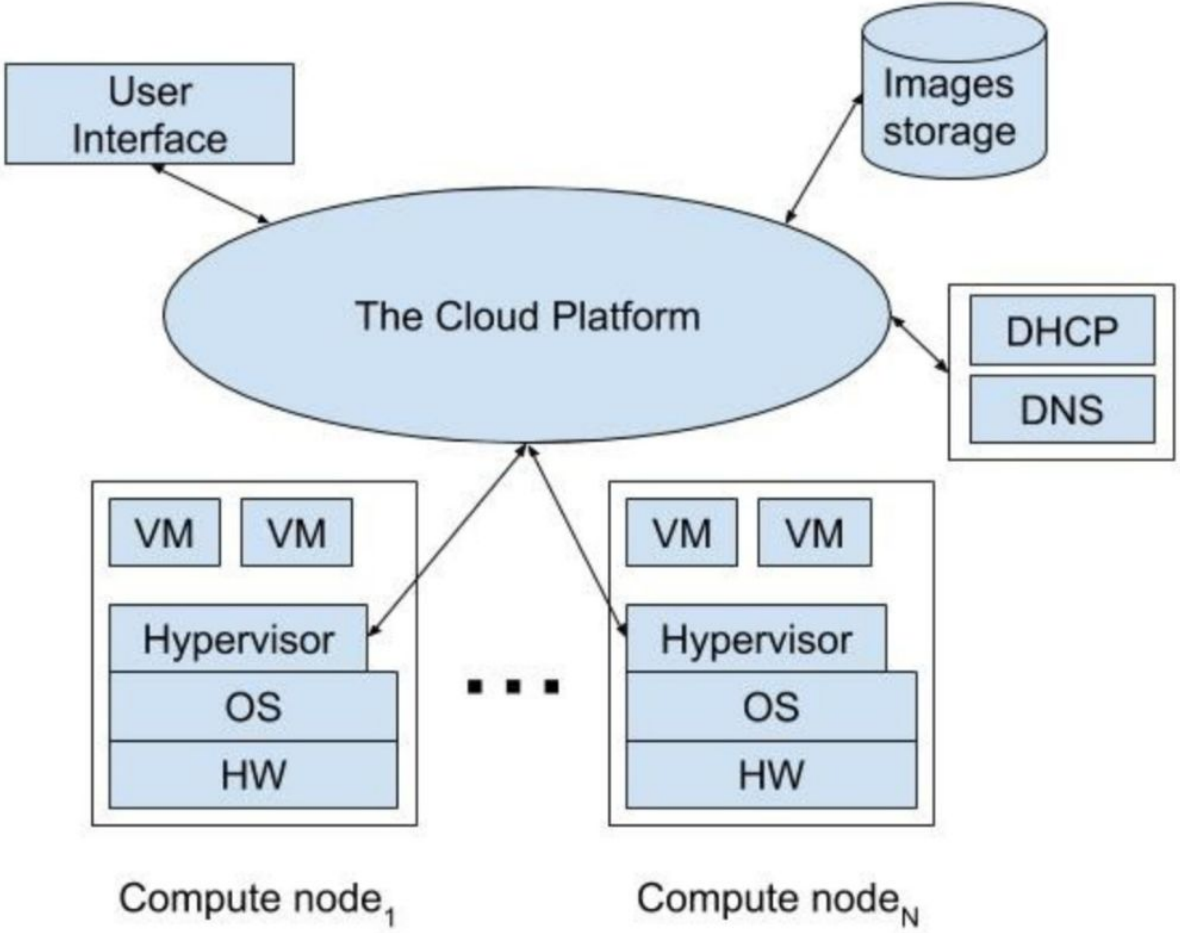


Figure 3.2: The common cloud platform architecture.

Many different types, but typically **three tiers** of apps

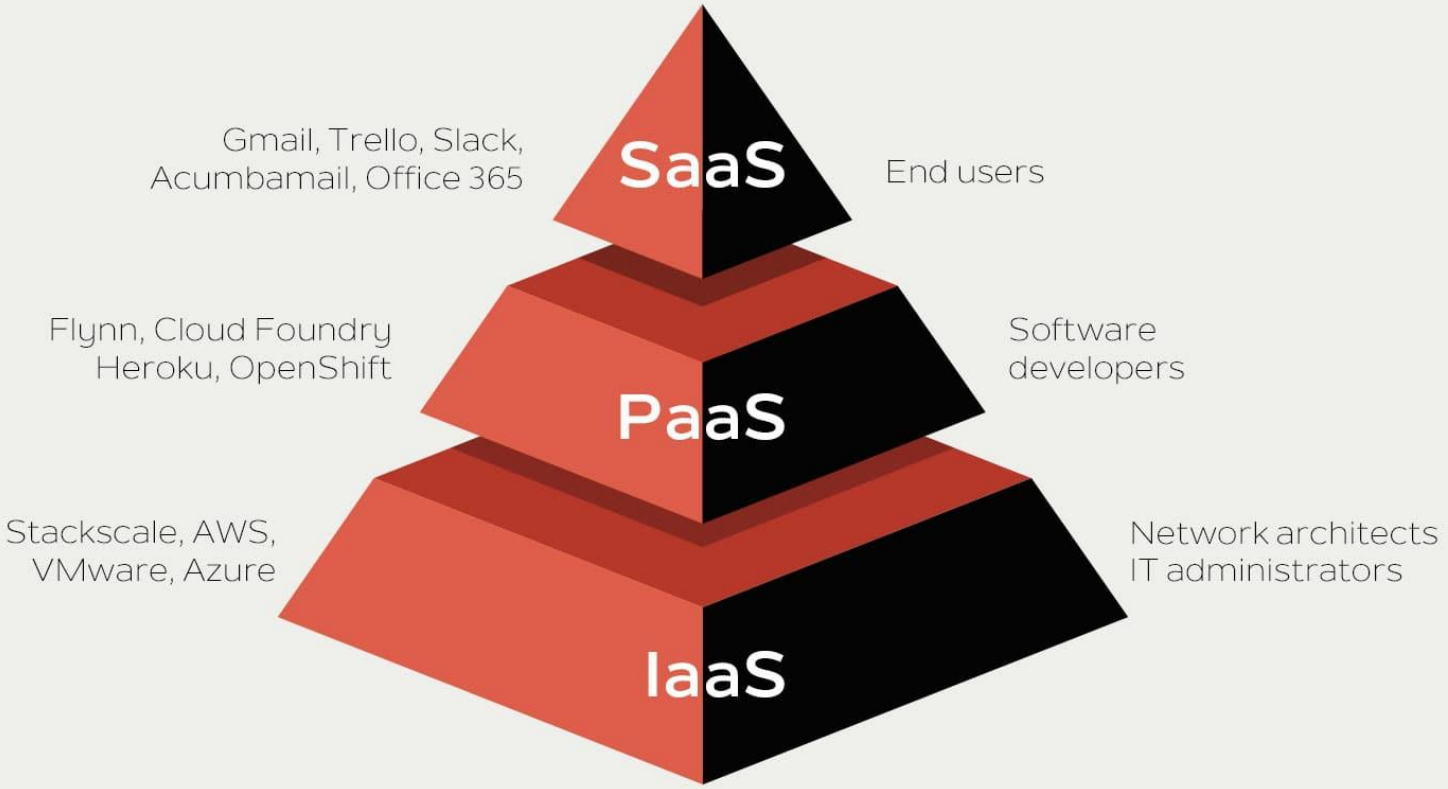
* as a service

- Infrastructure as a service (IaaS)
- Platform as a service (PaaS)
- Software as a service (SaaS)

Interestingly, there will typically be some cloud product that aligns with each of these

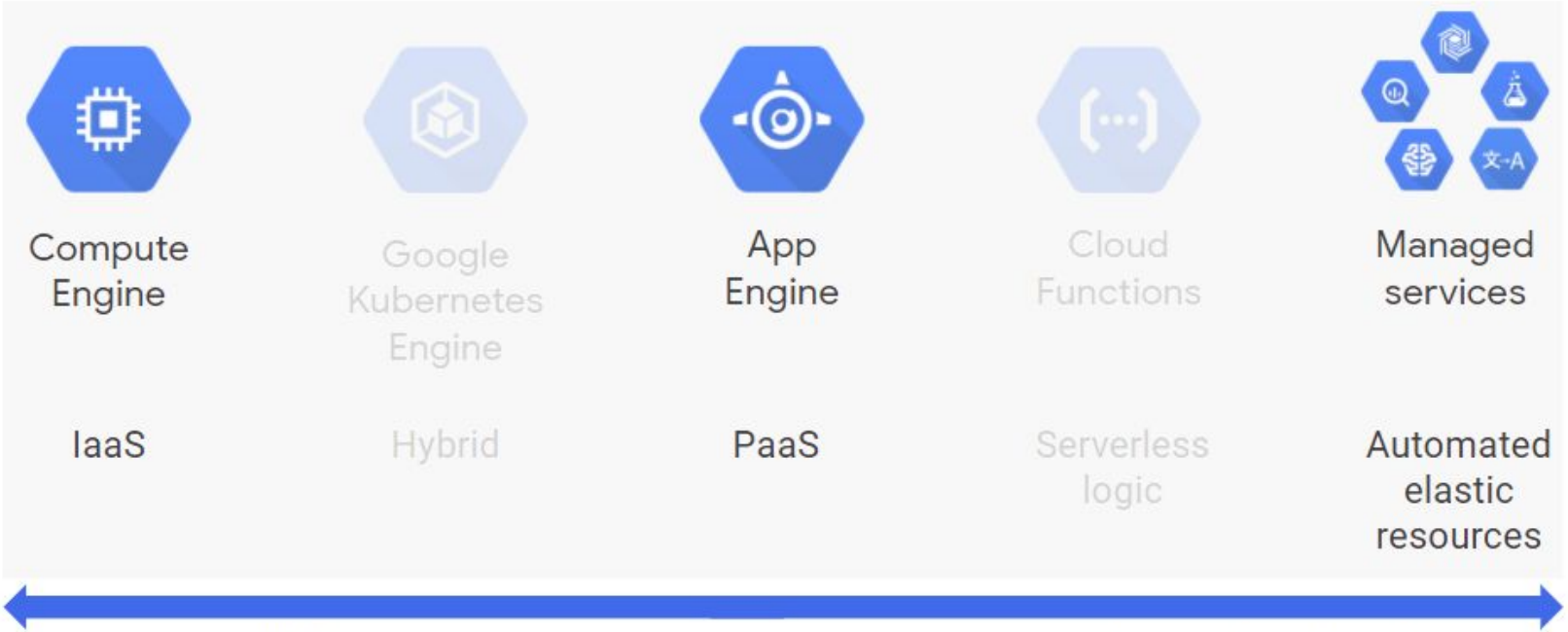
- Or some combination

Cloud service models



Stackscale

Google Cloud provides a variety of choices

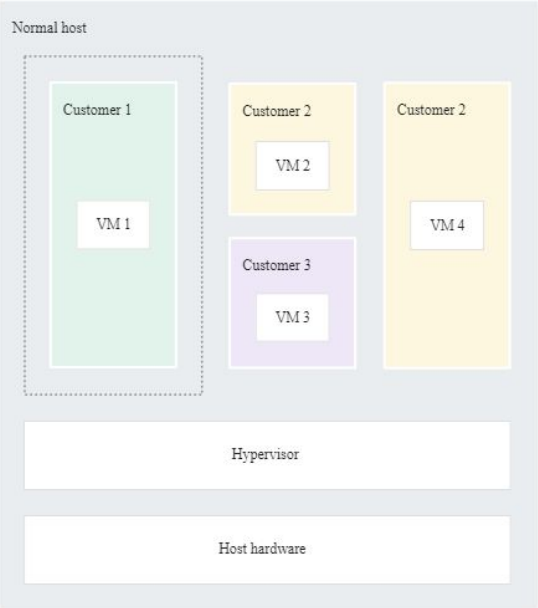


Google Cloud

IaaS

Typically **virtual machines**

- Meaning, the infrastructure is virtualized
  - Don't need a server "on-prem"



IaaS

Things to consider for the VM:
- Machine type/specs
- Operating system
- Who has access
- Fresh install vs. template

Other considerations:
- Firewalls (internal and external!)
- Scaling (elasticity)
- Location
- Price!
    - Use vs. storage

What are some sample use cases?

Autoscaling Google Compute Engine

https://cloud.google.com/compute/docs/autoscaler

Uses "managed instance groups"

- Auto add/delete instance from groups as needed

Requires:

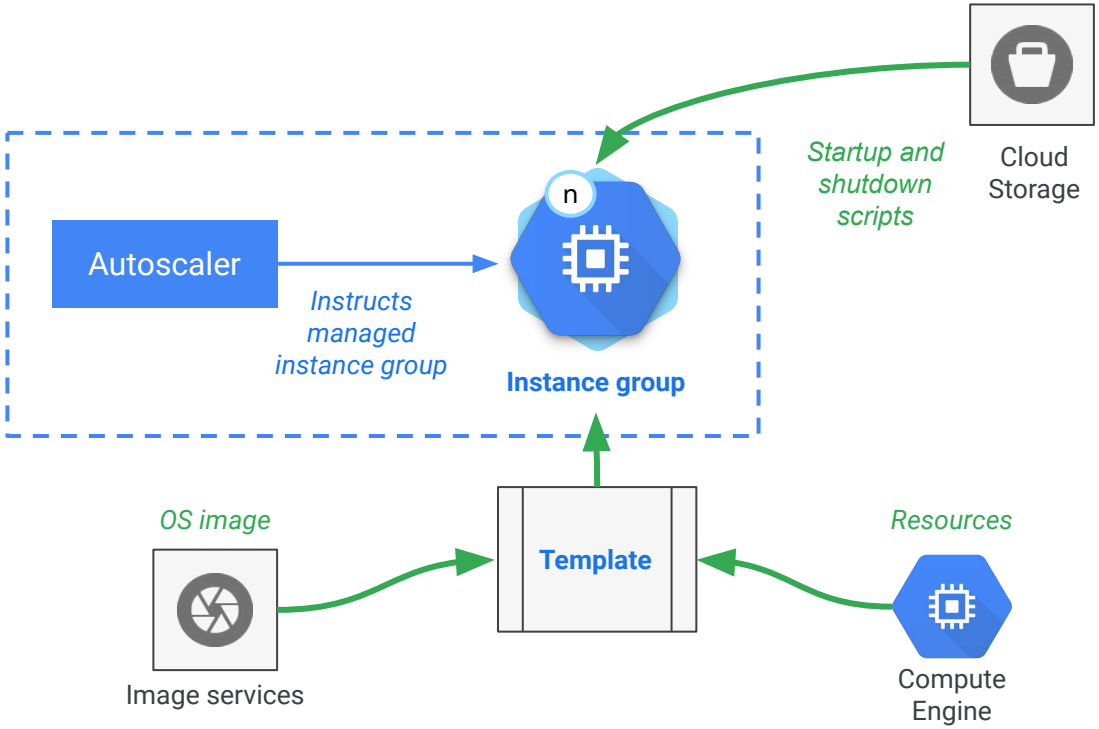
- Autoscaling service agent
- A policy (metrics to look for)
  - CPU usage
  - Load balancing
  - Etc.

Create a managed instance group (MIG)

A managed instance group (MIG) is a group of virtual machine (VM) instances that you control as a single entity. MIGs support features such as autohealing, autoscaling, load balancing, multiple zone coverage, and stateful workloads.

Quickstart tutorial in VM creation

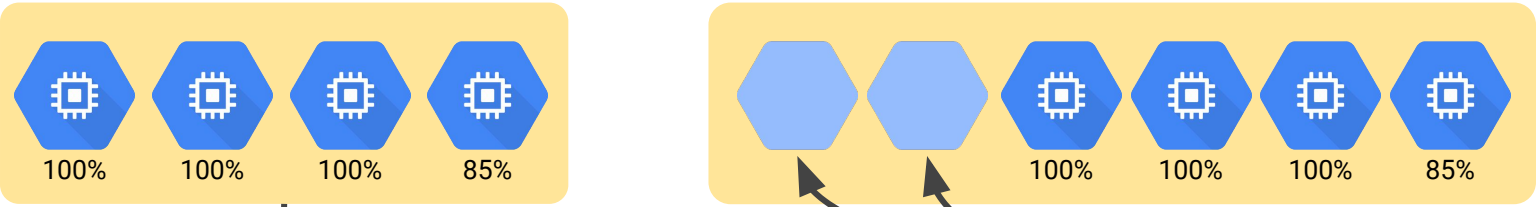
How autoscaling works



Google Cloud

Scale-out policy decision

managed instance group



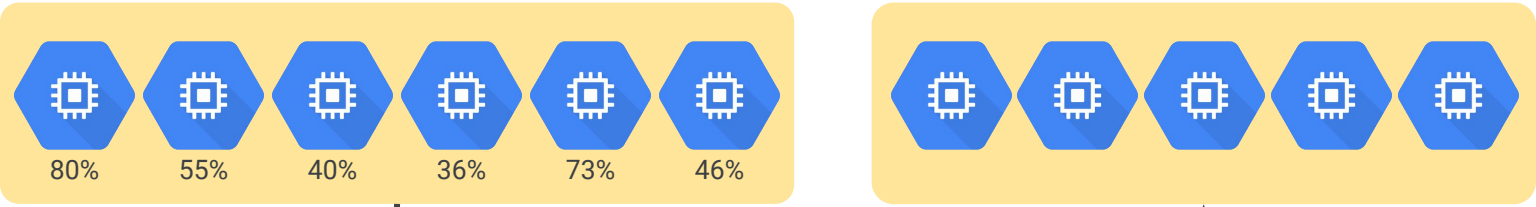
Target CPU utilization = 0.75

1. Actual utilization = 385/400 = 96.25%
2. If +1 then 385/500 = 77% (Still above target 75%)
3. If +2 then 385/600 = 64.16%

Google Cloud

Scale-in policy decision

managed instance group



Target CPU utilization = 0.75

1. Actual utilization = 330/600 = 55%
2. If -1 then 330/500 = 66% (Still below target)
3. If -2 then 330/400 = 82.5%

Google Cloud

Autoscale demo

https://codelabs.developers.google.com/codelabs/hpc-slurm-on-gcp#0

~~(Interestingly, Slurm is what runs on GVSU's HPC environment)~~

- ~~And many others...~~

~~Notes:~~

- ~~Create new project~~
- ~~Pull this instead: https://github.com/GoogleCloudPlatform/slurm-gcp~~
- ~~Look in here for the tfvars:~~
  ~~slurm-gcp/terraform/slurm_cluster/examples/slurm_cluster/simple_cloud~~

aaaaaaAAAAAAAAAAAAAAAAAAAHHHH

Well Slurm is at least fun let's try THAT out

https://cloud.google.com/cluster-toolkit/docs/quickstarts/slurm-cluster

```
srun -N 3 hostname
```

This command creates three compute nodes for your HPC cluster. This may take a minute while Slurm auto-scales to create the three nodes.

When the job finishes you should see an output similar to:

```
$ srun -N 3 hostname
    hpcslurm-debug-ghpc-0
    hpcslurm-debug-ghpc-1
    hpcslurm-debug-ghpc-2
```

Proof in case it broke again

VM instances

Filter Enter property name or value

Status	Name	Zone	Recc
✓	hpcslurm-controller	us-central1-a	
	hpcslurm-debugnodeset-0	us-central1-a	
	hpcslurm-debugnodeset-1	us-central1-a	
	hpcslurm-debugnodeset-2	us-central1-a	
✓	hpcslurm-slurm-login-001	us-central1-a	

Mozilla Firefox

https://ssh.cloud.google.com/v2/ssh/projects/glowing-cargo-434302-r8/zones/us-central1-a/instances/hpcslur

Firefox prevented this site from opening a pop-up window. Preferences

SSH-in-browser UPLOAD FILE DOWNLOAD FILE

```
Creating directory '/home/erik_fredericks_gmail_com'.
[erik_fredericks_gmail_com@hpcslurm-slurm-login-001 ~]$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
compute      up   infinite     20  idle~ hpcslurm-computenodeset-[0-19]
debug*       up   infinite      4  idle~ hpcslurm-debugnodeset-[0-3]
h3           up   infinite     20  idle~ hpcslurm-h3nodeset-[0-19]
[erik_fredericks_gmail_com@hpcslurm-slurm-login-001 ~]$ srun -N 3 hostname
```

PaaS

Before we didn't worry about the bare-metal hardware

- Now we don't worry about that **plus** the operating system/environment!
- i.e., a *serverless* approach

i.e., we just want to deploy a Python app and don't care about the server itself

Google Cloud	- App Engine / Cloud Functions
AWS	- App Runner / Lambda
Azure	- Azure PaaS / Azure Functions

(there are more - just these are common)

PaaS Demo!

This one's pretty straightforward - basic website with App Engine

https://cloud.google.com/appengine/docs/standard/hosting-a-static-website

Buuut,

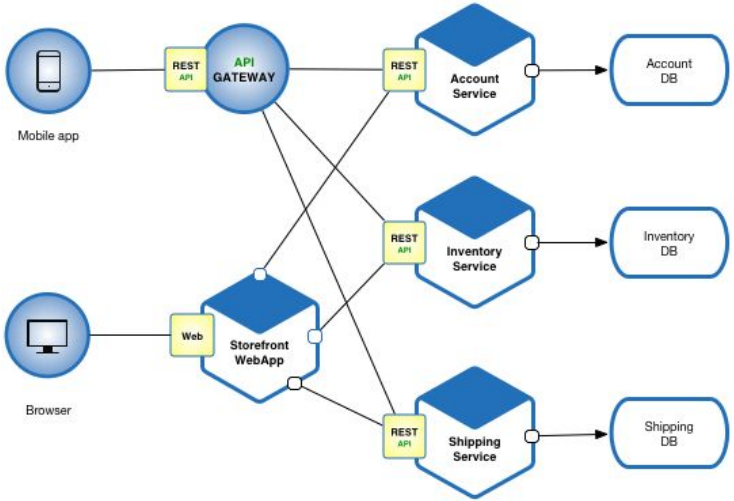
- Gets us to YAML (get ready for pain and suffering)

Functions as a Service (FaaS)

Less common in marketing materials, but here:

- Functions deployed remotely without a care for the app *as a whole*
- i.e., I am calling a function as an API
  - Part of the serverless paradigm, so still PaaS

- Lambda / Cloud Functions / Azure Functions also apply here



What are some sample use cases?

We'll be going into greater detail on these later

SaaS

Don't care about the hardware

- Or the operating system
- Or the installation!

Think: Google Docs, Office 365, etc.

And now … MaaS

AI and machine learning products

Try Gemini 1.5 models, the latest and most advanced multimodal models in Vertex AI. See what you can build with up to a 2M token context window, starting as low as $0.0001.

Try it in console | Contact sales

Summarize large documents with generative AI

Deploy a preconfigured solution that uses generative AI to quickly extract text and summarize large documents.

Deploy an AI/ML image processing pipeline

Launch a preconfigured, interactive solution that uses pre-trained machine learning models to analyze images and generate image annotations.

Create a chat app using retrieval-augmented generation (RAG)

Deploy a preconfigured solution with a chat-based experience that provides questions and answers based on embeddings stored as vectors.

Ok, so we have the various *aaS

Things **you** need to take away from this:

1) If you have a need, there's probably a solution out there
2) Know **what** technology to use for **which** problem

FOR EXAMPLE - WHAT TECH CAN YOU USE?

1) I need you to build a website for our company
2) I need you to create a backend service so our users can purchase products from the online store
3) I need you to create a chatbot because [expletive deleted] every website needs a chatbot now

Let's take a look at IaaS for a moment

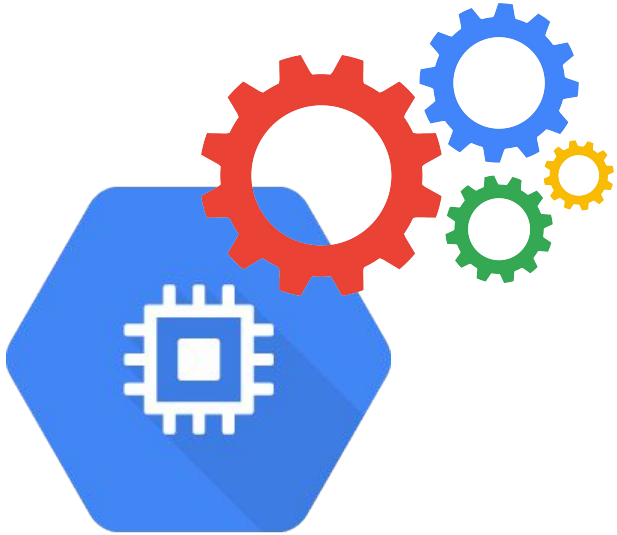
Where Compute Engine fits within Google Cloud

Compute Engine	App Engine	Cloud Functions	Google Kubernetes Engine
IaaS	PaaS	Serverless logic	Hybrid
Virtual machines with industry-leading price/performance	A flexible, zero ops platform for building highly available apps	A lightweight fully managed serverless execution environment for building and connecting cloud services	Cluster manager and orchestration engine built on Google's container experience

Google Cloud

Compute Engine is an infrastructure-centric solution

- Type of IaaS
- Scalable, high-performance VMs
- Run any computing workload
- Predefined or custom machine types
- Windows or Linux
- No upfront investment required



Google Cloud

Create VMs that are right for your workloads

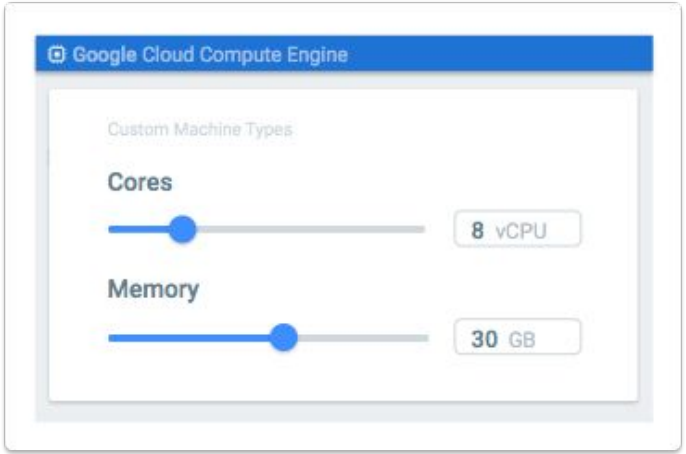
Machine type options to consider:

- Higher proportion of memory to CPU
- Higher proportion of CPU to memory
- Blend of both

Select from predefined VM configurations:

- General-purpose
- Memory-optimized
- Compute-optimized

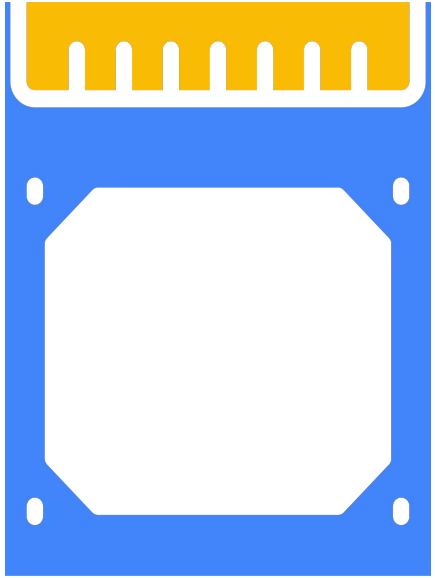
Customize your own configuration



Google Cloud

Building virtual disks

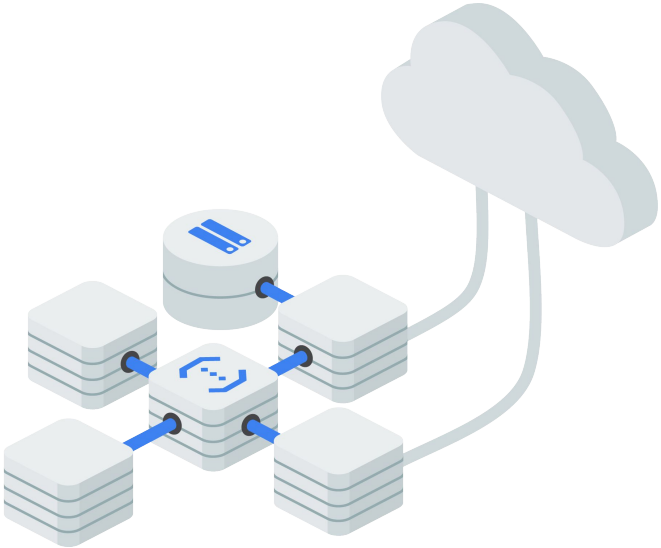
- Network storage can be attached to VMs as persistent disks (PDs).
- PDs are cost-effective, durable, and offer good performance.
- Local SSDs provide higher performance with lower latency, but exist only for the lifetime of a specific instance.
- Standard PD throughput performance and IOPS increases linearly with the size of the disk until it reaches set per-instance limits.
- SSD PD IOPS performance depends on the number of vCPUs in the instance in addition to disk size.



Google Cloud

Networks connect Compute Engine instances to each other and to the internet

- Inbound/outbound firewall rules
- Create static routes
- Scale and distribute applications using Cloud Load Balancing
- Global and multi-regional subnetwork



Google Cloud

Compute Engine pricing

Google Cloud Platform Pricing Calculator

Prices are up to date. Last update: 15-August-2019

COMPUTE ENGINE | APP ENGINE | KUBERNETES ENGINE | CLOUD RUN | CLOUD STORAGE | NETWORKING | BIGQUERY | BIGQUERY ML

Search for a product you are interested in.

Instances

Number of instances *
5

What are these instances for?
Databases

Operating System / Software
Free: Debian, CentOS, CoreOS, Ubuntu, or other User Provided OS

Machine Class
Preemptible

Machine Family
General purpose

Machine Generation
Second (latest)

Machine type
n2-standard-2 (vCPUs: 2, RAM: 8GB)

Estimate

Compute Engine

5 x Databases

3,650 total hours per month

VM class: preemptible

Instance type: n2-standard-2

Region: Iowa

Total available local SSD space 1x375 GB

Estimated Component Cost: USD 175.77 per 1 month

Total Estimated Cost: USD 175.77 per 1 month

Estimate Currency
USD - US Dollars

EMAIL ESTIMATE | SAVE ESTIMATE

https://cloud.google.com/products/calculator/

Google Cloud

and now PaaS

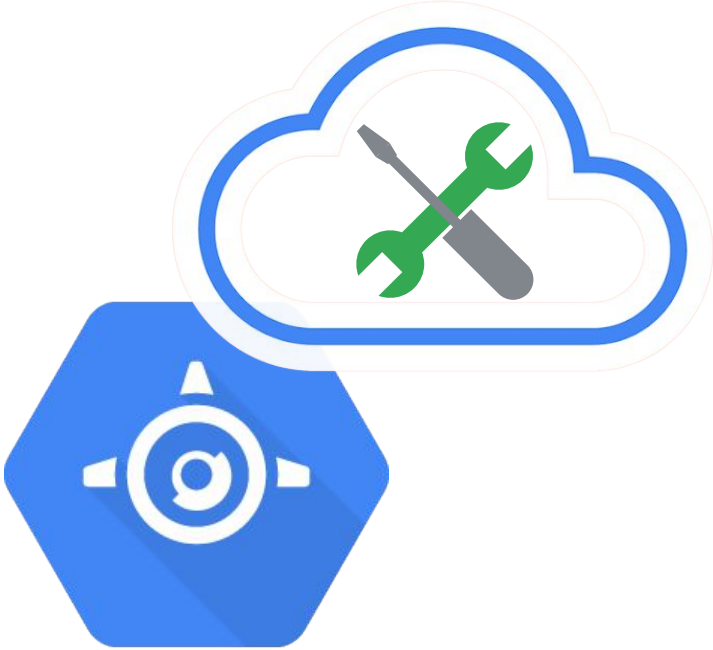
Where App Engine fits within Google Cloud

Compute Engine	App Engine	Cloud Functions	Google Kubernetes Engine
IaaS	PaaS	Serverless logic	Hybrid
Virtual machines with industry-leading price/performance	A flexible, zero ops platform for building highly available apps	A lightweight fully managed serverless execution environment for building and connecting cloud services	Cluster manager and orchestration engine built on Google's container experience

Google Cloud

App Engine is a platform-centric solution

- Type of PaaS
- No need to buy, build, or operate hardware/infrastructure
- No managing servers or configuring deployments
- Focus on app development instead of operations
- Use a range of languages and tools
- Automatic scaling



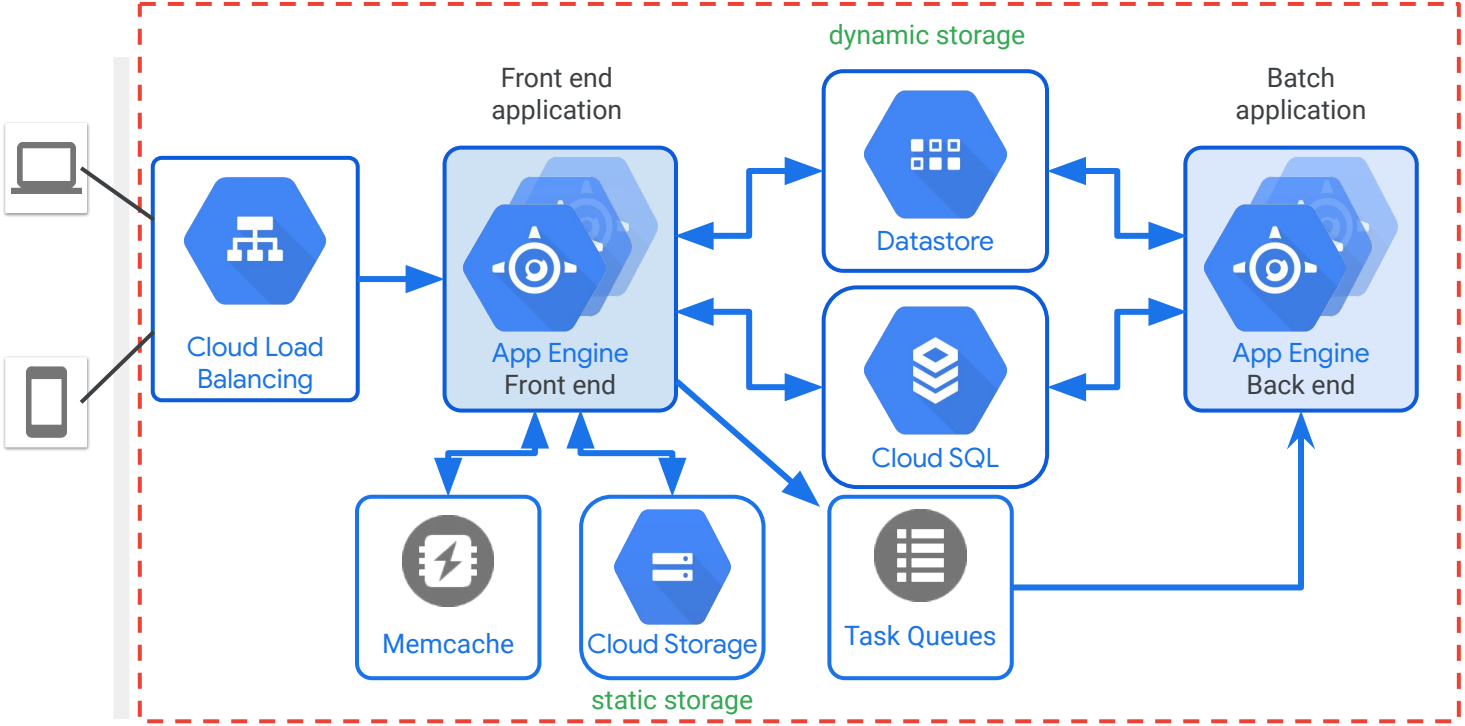
Google Cloud

App Engine offers two different environments

Standard environment	Flexible environment
• Fully-managed	• Docker container support
• Scale to zero	• VMs exposed
• Specific versions of supported languages	• Any language in your container
• Changes/configuration limited	• More options for infrastructure customization and configuration for performance

Google Cloud

An App Engine architecture example



Google Cloud

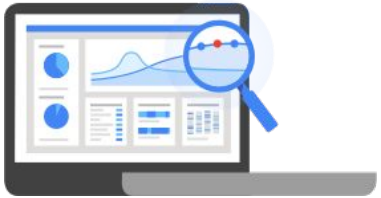
App Engine addresses the key needs of developers

Multiple storage options

Automatic scaling

Load balancing

App versioning



Monitoring and logging

Security

Google Cloud

SaaS

https://youtu.be/WJmP4Lqopps

- (Important bit? What tools/tech to use to build the "thing" you want)

https://cloud.google.com/saas

- Tends to be 'install trusted plugin from Marketplace'

Another demo!

~~https://developers.google.com/codelabs/gcp-marketplace-saas#0~~
<appears broken>

Let's do a PHP app! Different than what we've done so far
https://cloud.google.com/appengine/docs/standard/php-gen2/building-app