Google Cloud Platform Cloud Architect Essentials

Overview of Google Cloud Platform (GCP)

Benefits of Cloud Computing (in general)

- “Pay-as-you-go” basis, use only what you need
- Convert capital expenses into operating expenses
- Focus on rapid innovation
- Productivity enhanced due to no software installed
- “Vertically integrated” stacks enhance functionality, performance, reliability, and security

What is Google Cloud Platform (GCP)

- Google's suite of cloud computing services
  - Runs on the same infrastructure and network as Google
- Google's worldwide collection of data centers
- Hosts IaaS, PaaS, and SaaS use cases
- At a basic level, it hosts and manages your computer infrastructure so you don't have to
  - Does it at 'Google-scale'

GCP at a glance

How does Google Cloud Platform stand out?

- Access the same infrastructure Google uses for their own services
- Customer friendly pricing (https://cloud.google.com/pricing)
  - Compute Engine instances billed per minute, not hour
  - **Automatic** sustained use discounts – no upfront commitment required
- Private Global Fiber Network
  - Blazing fast
  - Data moving between data centers never leaves the private network
- Live migration of Virtual Machines
- Better performance
  - Website benchmarks showed as much as 50% improvement in loading speed
  - Instantly scalable. VM's can auto-scale without shutting machine down.
  - Access to GPU's for high end scientific computing and machine learning

How does Google Cloud Platform stand out?

- Industry leading security
  - Over 500 full time security professionals
  - All data encrypted in transit and at rest
  - Expansive audit compliance list
    - SSAE16, ISO 27017, ISO 27018, PCI, and HIPAA
- Access to innovative resources not available anywhere else
  - Big Data, Machine Learning
  - API's (Video Intelligence, Maps)
  - "Build what's next"

Google Cloud Platform Cloud Architect Essentials

Datacenter Infrastructure

Infrastructure breakdown

- Data Centers
  - Regions and Zones
- Backbone
  - High speed private fiber network
- Points of Presence
  - connect Google's network to the rest of the internet
- Edge Nodes
  - Closest to end users worldwide

Datacenters - Worldwide presence

https://cloud.google.com/about/locations/

Cloud Regions and Zones

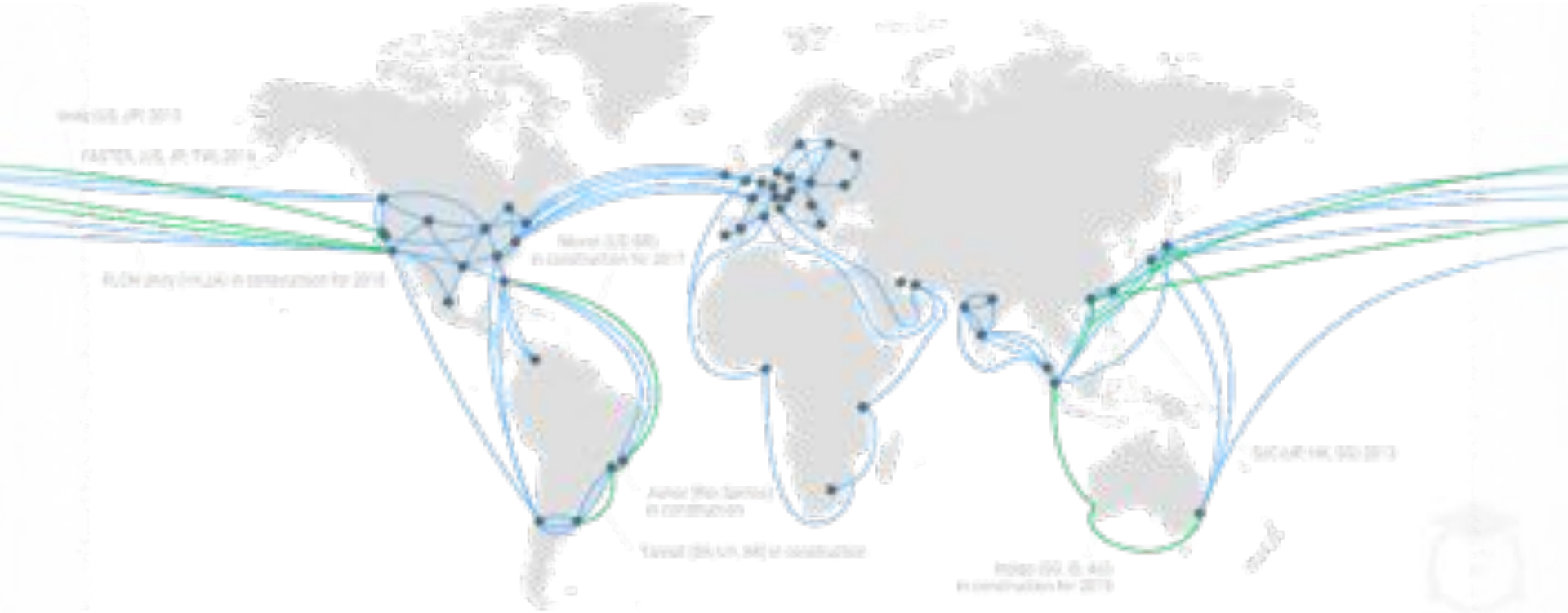
- Nine regions split into 27 zones
- Regions
  - Regions are specific geographical locations where you can run your resources
  - Collections of zones
  - Regional resources are available to resources in any zone in the region
  - Frequently expanding
- Zones
  - Isolated physical locations within a region
  - Zonal resources are only available in that zone
  - Machines in different zones have no single point of failure

- For example: an effective disaster recovery plan would have assets deployed across multiple zones, or even different regions.

Commitment to environmental responsibility

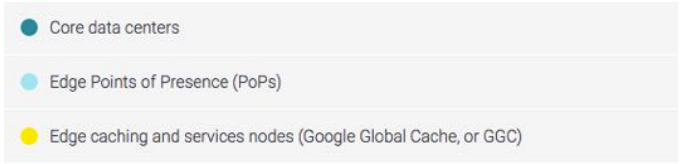
- 100% renewable energy
- Extremely efficient infrastructure
  - Optimized to run within Google's environment
  - Carbon neutral since 2007
- Hosting company resources on GCP is more environmentally friendly than hosting your own server resources
- Learn more at https://www.google.com/about/datacenters/

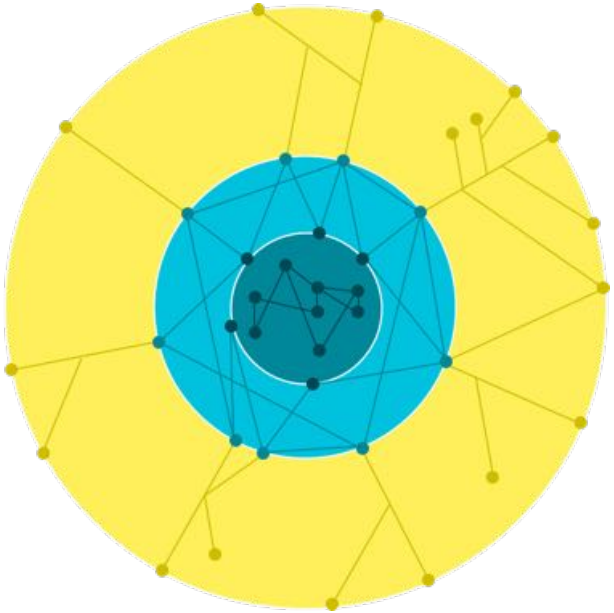
Backbone – global fiber network



Network infrastructure from data centers to end users

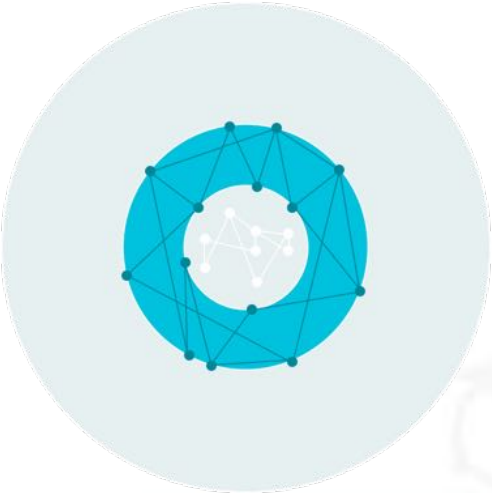
- Three elements:
  - Core data centers
  - Edge Points of Presence (PoPs)
  - Edge nodes





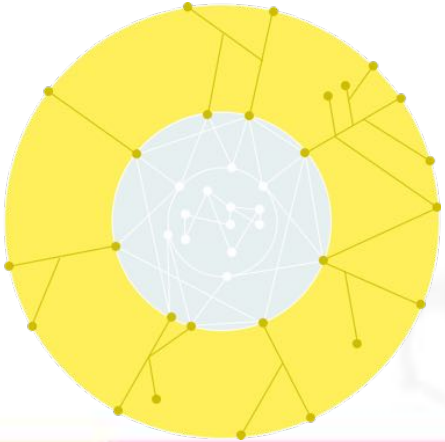
Points of Presence

- 90+ locations worldwide
- Brings Google traffic closer to users worldwide, thereby reducing their costs and providing users with a better experience.
- Connects to the private meshed network backbone that connects Edge PoPs to data centers, and bridges to the public Internet



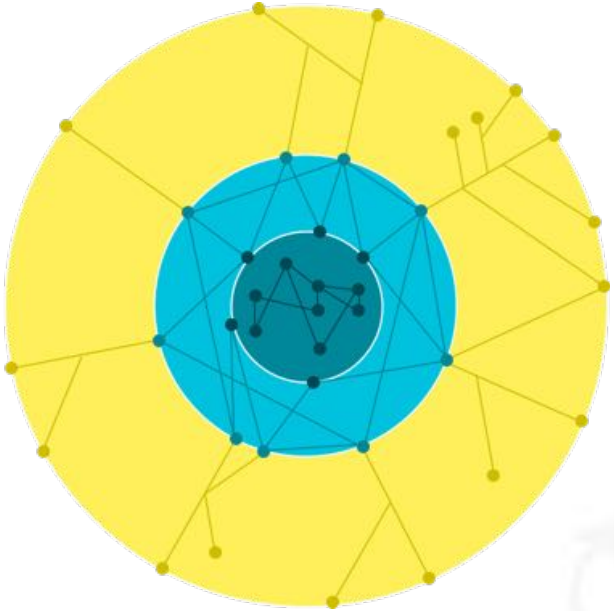
Edge nodes

- Tier of Google’s infrastructure closest to end users
- Internet service providers (ISP’s) deploy Google-supplied servers inside their own network
- “Static content that is very popular with the local host's user base, including YouTube and Google Play, is temporarily cached on edge nodes. Google's traffic management systems direct user requests to an edge node that will provide the best experience.”
- Pulling popular content from edge cache is substantially faster than pulling everything from data centers



End result

- Fast, redundant, worldwide presence that provides fast and reliable access to your resources no matter where in the world you are
- All of this while being 100% carbon neutral



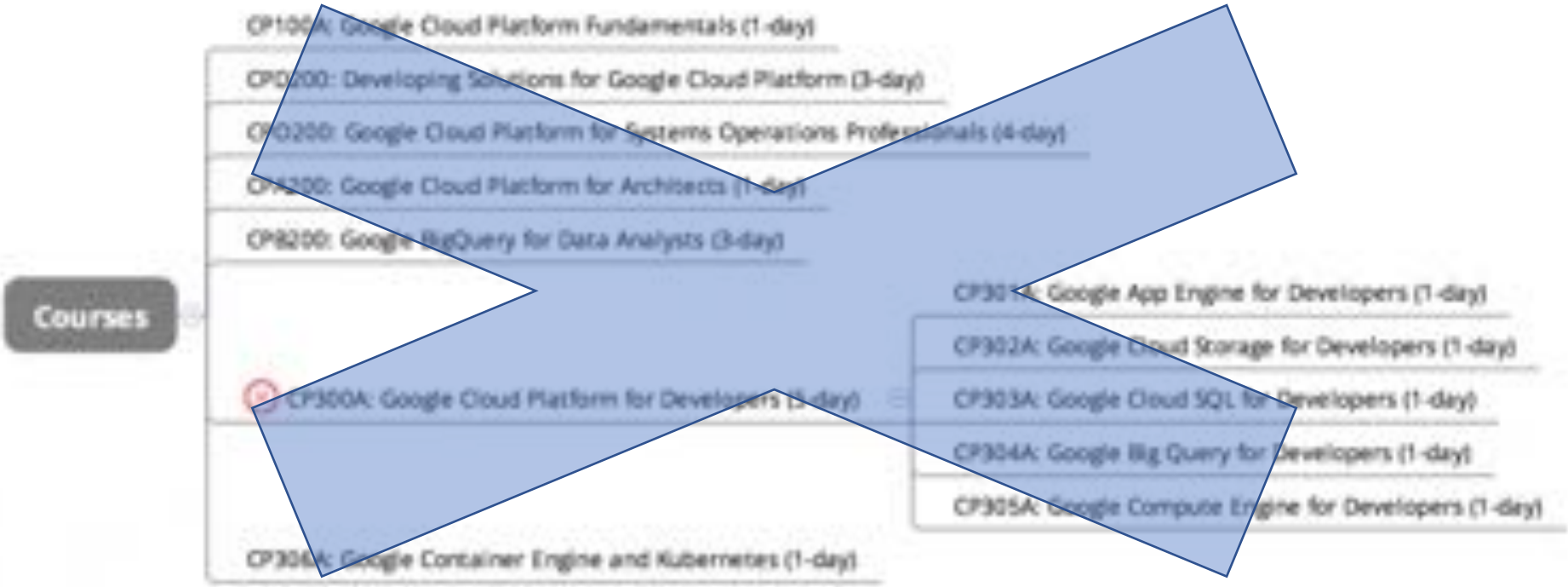
Google Cloud Platform Cloud Architect Essentials

Certification Overview

How are Google Cloud certifications composed?

- Very fluid and rapidly changing environment
- GCP substantially changed their certification tests Dec 2016


Courses
CP100A: Google Cloud Platform Fundamentals (1-day)
CP0200: Developing Solutions for Google Cloud Platform (3-day)
CP0200: Google Cloud Platform for Systems Operations Professionals (4-day)
CPA200: Google Cloud Platform for Architects (1-day)
CP8200: Google BigQuery for Data Analysts (3-day)
CP300A: Google Cloud Platform for Developers (5-day)
CP301A: Google App Engine for Developers (1-day)
CP302A: Google Cloud Storage for Developers (1-day)
CP303A: Google Cloud SQL for Developers (1-day)
CP304A: Google Big Query for Developers (1-day)
CP305A: Google Compute Engine for Developers (1-day)
CP306A: Google Container Engine and Kubernetes (1-day)


Courses
CP100A: Google Cloud Platform Fundamentals (1-day)
CP200: Developing Solutions for Google Cloud Platform (3-day)
CP300: Google Cloud Platform for Systems Operations Professionals (4-day)
CP8200: Google BigQuery for Data Analysts (3-day)
CP300A: Google Cloud Platform for Developers (3-day)
CP301A: Google App Engine for Developers (1-day)
CP302A: Google Cloud Storage for Developers (1-day)
CP303A: Google Cloud SQL for Developers (1-day)
CP304A: Google Big Query for Developers (1-day)
CP305A: Google Compute Engine for Developers (1-day)
CP306A: Google Container Engine and Kubernetes (1-day)

Current GCP Certifications

- Google Certified Professional
  - Cloud Architect
  - Data Engineer
- Google Certified Associate
  - G Suite Administrator

GCP Cloud Architect certification compared to AWS

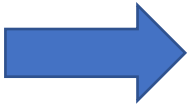
Architecting

Architecting exams validate technical knowledge for solutions architects, solution design engineers, and anyone who designs applications and systems on AWS.

AWS Certified Solutions Architect – Associate

This exam tests your technical expertise in designing and deploying scalable, highly available, and fault tolerant systems on AWS. This is for anyone with one or more years of hands-on experience designing distributed applications and systems on the AWS platform. Learn More >>

Register Now



AWS Certified Solutions Architect – Professional

This exam tests your advanced technical skills and experience in designing distributed applications and systems on the AWS platform. This is for anyone that has completed an AWS Certified Solutions Architect - Associate with two or more years of hands-on experience designing and deploying cloud architecture on AWS. Learn More >>

Register Now

‘Old’ tests rolled into non-tested training tracks

- https://cloud.google.com/training
- Multiple non-tested training tracks along with tested certifications
- This course will follow the Google Cloud Platform Fundamentals: Core Infrastructure objectives

‘Cramming’ for exam made difficult by design

- “There are no formal requirements to take the Google Cloud Architect Certification exam. However, please note that hands-on experience performing the job tasks in the guide and using Google Cloud Platform technology is the best preparation for the exam. Google's certifications are intended to identify people who have demonstrated skills required for a job role. That's why we do not offer exam prep materials, as the exams are not intended to assess someone's mastery of training.”

Preparing to pass the exam

- This course will the the foundation to build upon for more advanced concepts.
- **Learn by doing** - Best way to prepare for the Cloud Architect exam is to *use* GCP and become intimately familiar with all of it's services, and know what service serves which purpose.
- Our courses will stress working hands on, and will give exercises to use it to achieve business objectives.
- We will provide the tools to master both the hands on and high level concepts necessary to pass the exam as we ramp up our courses in skill level.


Linux Academy
Google Cloud Platform Cloud Architect Essentials
Free Trial and Always Free Tier

Free Trial and Always Free Tier

- https://cloud.google.com/free/
  - Free trial + Always Free
  - Always free tier does not cut into free trial credit

Free Trial

- $300 credit for 12 months
- Trial ends when either credit is used up, or 12 months pass, whichever comes first
- Requires credit card for trial, but does not auto-charge when trial runs out
- When upgrading to full account, free credit/time still remains and will be auto-charged after credit runs out

- Trial limitations (not exhaustive list)
  - Compute Engine
    - No more than 8 cores running at once (across all instances)
    - No more than 100GB solid state disk (SSD) space
    - 2TB persistent standard disk space

Always Free Tier

- Try GCP products for free up to usage limits
- Used for testing/experimenting
- Only for US regions (does not apply to App Engine)
- Limits apply to all projects across all regions per account
  - Example: 5GB Cloud Storage can be used in one region, or split across 2 regions
- Usage beyond Always Free limitations will be charged to free trial credit or billed if trial is expired
- Always Free tier never expires
- For example:
  - Compute Engine
    - 1 f1-micro instance per month (US regions only)
    - 30 GB-months HDD, 5 GB-months snapshot
- More info for usage limits: https://cloud.google.com/free/docs/always-free-usage-limits

Signing up for GCP account

- Requires Google account
- Can use non-Gmail email to associate with Google account

Let’s get started!

Google Cloud Platform Cloud Architect Essentials

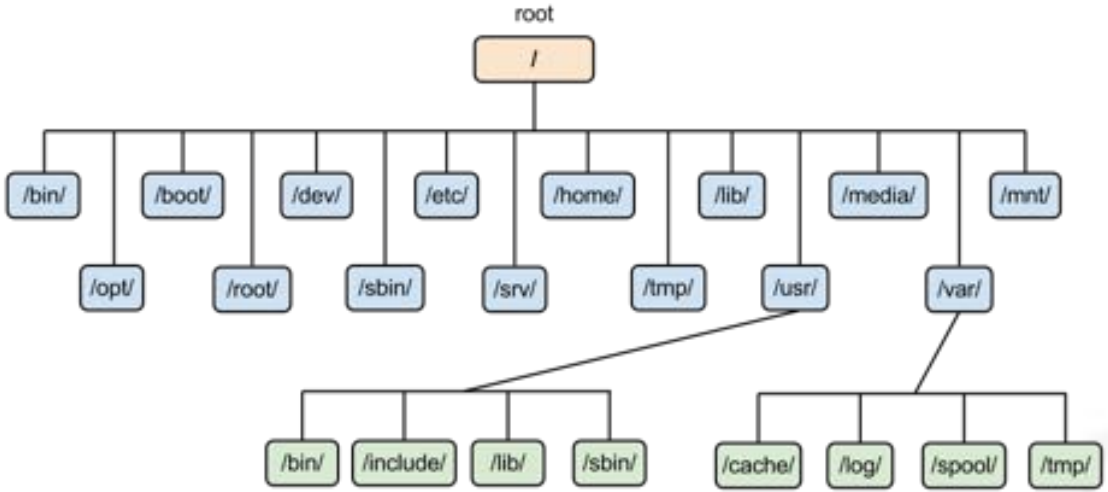
Organization and Projects

For the rest of this course….

- Overview of a concept (slides)
- Hands on with that concept
- Your turn to do the same (homework)

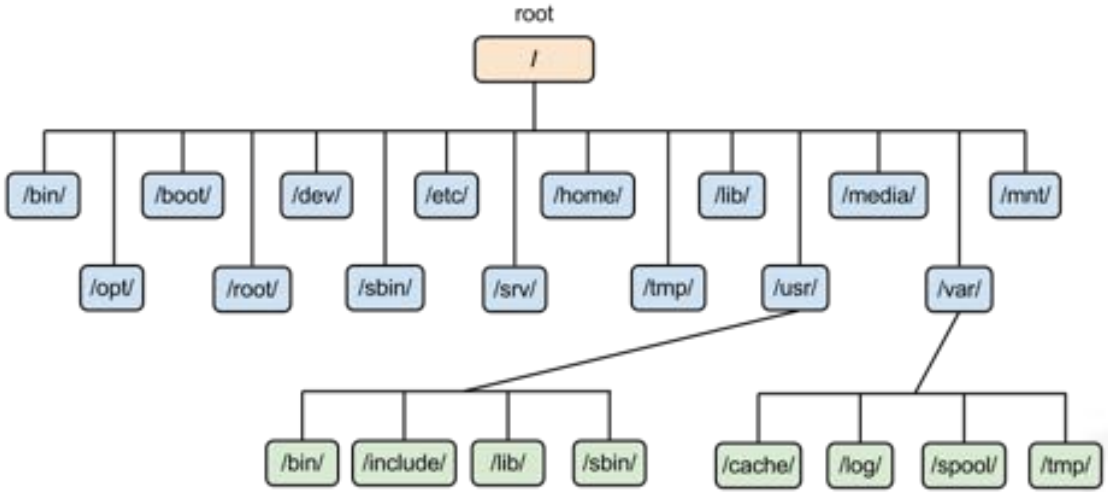
GCP Organization hierarchy

- Like any operating system or complex tech resource, some form of organization and access control is a necessity
- GCP's implementation is called **Cloud Resource Hierarchy**



Purpose of the Cloud Resource Hierarchy

- Provide a hierarchy of ownership
  - Identity and Access Management (IAM)
- Provide attach points and inheritance for access control and organization policies.



Hierarchy Overview

- Organization (not applicable to individual accounts)
- Projects
- Resources

What are Projects?

- **Core** organizational component of GCP
- Control access to resources
  - Who has access to what?
- Creating, enabling, and using all Cloud Platform services
  - Per project basis
  - Permissions
  - Billing
  - API's

Three identifying attributes

- Project Name – aka “Friendly Name”
- Project ID
  - Also known as Application ID
- Project Number
  - Used in various places for identifying resources that belong to specific projects. For example, service account access and names

Your Homework

- Access Project Manager
- Create a project
  - Choose a friendly name
  - Watch how Project ID is automatically created from friendly name
- Look up Project info after creation
- Delete your Project
- Create new Project – “Bookshelf Application”
  - Do NOT delete this Project!

Google Cloud Platform Cloud Architect Essentials

Identity and Access Management (IAM)

What we are covering

- Discuss the importance of resource access and management
- Overview of GCP's implementation of Identity and Access Management (IAS)
  - Members, Roles, Resources
  - What are Members?
  - Primitive and Predefined Roles
  - IAM Policy Hierarchy
  - Hands on demo

For any organization:

Why is this important?

- Important to provide granular access to resources
- Prevent unwanted access to other resources
- Adopt the security principle of least privilege

How does this relate to GCP?

Cloud Identity and Access Management (Cloud IAM)

- Members (who) are granted permissions and roles (what) to GCP services (resource) using the principle of **least privilege**

What is a member (the 'who')?

- Can be either a person or a service account
- People
  - Google account
  - Google group – e.g. DevTeam@mycompany.com
  - G Suite Domain
  - Cloud Identity domain (Beta)
    - Organization domain that is not a Google domain/account
- Service account
  - Application access

What is a Service Account?

- Special type of Google account that belongs to your application, not an end user
- Identity for carrying out server-to-server interactions in a project
  - Example: Local server backup application writing data to Cloud Storage
- Identified with an email address:
  - <project_number>@developer.gserviceaccount.com
    <project_id>@developer.gserviceaccount.com

What is a Role (the 'what')?

- Collection of permissions to give access to a given resource
- Permissions represented in form <service>.<resource>.<verb>. E.g.: compute.instances.delete

Permissions vs. roles

- Users are not directly assigned permissions, but are assigned roles which contain a collection of permissions

Primitive vs Predefined (Curated) Roles

Primitive

Historically available GCP roles before Cloud IAM was implemented.

- Applied to Project level
- Broad roles
- Viewer
  - Read only actions that preserve state (i.e. cannot make changes)
- Editor
  - Same as above + can modify state
- Owner
  - Same as above + manage access to project and all project resources
  - Set up project billing

When to choose Primitive Roles?

- When the Cloud Platform service does not provide a predefined role
- Only need broader permissions for a project
- Allow a member to modify permissions for a project
- When you work in a small team where the team members don’t need granular permissions.

Predefined (or Curated) Roles

- Much more granular access, prevent unwanted access to other resources
- Granted at resource level
- Example: App Engine Admin – Full access to only App Engine resources
- Multiple predefined roles can be given to individual users

- All current Predefined Roles - https://cloud.google.com/iam/docs/understanding-roles#predefined_roles

IAM Policy

- Collection of statements that define who has what type of access
- Full list of roles granted to a member to a resource

Policy Hierarchy

- Resource access is organized hierarchically, from the Organization down to the Resources
- Organization → Project → Resources – parent/child format
- Each child has exactly one parent
- Children inherit Parent roles
- Parent policies overrule restrictive child policies

Roles - Putting it all together

- Roles are a collection of individual permissions
  - Roles are assigned to members, not permissions directly
- Roles include Primitive (broad) and Predefined (granular)
- IAM Policy is a full list of roles granted to members on a resource
- IAM Policies are hierarchally defined, with parent overruling child policy

Resources

- At lowest level, fundamental components of GCP
  - Compute Engine VM's
  - Google Cloud Storage Buckets
- Includes Organization, Projects, GCP Services and resources inside

Putting ALL of it together – IAM….

- …grants members… (users, groups, organizations, service accounts)
- …various roles… – primitive (broad) and predefined (granular)
- …in a hierarchal format… - parent overrules child
- …to GCP Resources – all resources into the GCP Hierarchy

Examples

- joe@gmail.com is granted Owner role to Project ‘My First Project’
- my-first-project-172354@developer.gserviceaccount.com granted App Engine Service Admin role to App Engine

Google Cloud Platform Cloud Architect Essentials

Interacting with Google Cloud Platform

What we will cover

- Introduce the three methods of interacting with GCP
  - Cloud console (web interface)
  - Google Cloud SDK (Software Development Kit)
  - RESTful API
- Following lessons will be hands on demos of the above

Cloud Console

- Centralized access to all your Google Cloud Platform projects.
- Access to the Google Cloud Shell
- A customizable project dashboard, with an overview of Google Cloud resources, billing, and a filterable activity listing
- Easy access to all Google Cloud Platform APIs, with a dashboard specific to each API, and access to manage your resources
- Links to Google Cloud Platform starting points, news, and documentation
- Essentially a ‘one stop shop’ for all that GCP has to offer
- Full tour next lesson

Google Cloud SDK

- SDK = Software Development Kit
- Command line interface (CLI) tools for managing resources and applications on GCP
- Includes gcloud, gsutil, and bq CLI tools
  - gcloud – many common GCP tasks
  - gsutil – access Cloud Storage
  - bq – work with data in BigQuery
- Can be locally installed as Docker image or ran from Cloud Shell
  - Install from https://cloud.google.com/sdk

Cloud Shell

- Interactive web based shell environment for GCP, accessed from the web console
- Easy to manage resources without having to install the Google Cloud SDK locally, and comes with all necessary tools built in. Always available when you need it.
- Includes:
  - A temporary Compute Engine virtual machine instance
  - Command-line access to the instance from a web browser
  - 5 GB of persistent disk storage
  - Pre-installed Google Cloud SDK and other tools
  - Language support for Java, Go, Python, Node.js, PHP and Ruby
  - Web preview functionality
  - Built-in authorization for access to Cloud Platform Console projects and resources

Restful API's (Application Program Interface)

- Per Google: "Intended for software developers"
- Programmatic access to GCP resources
  - Typically use JSON as an interchange format
  - Use OAuth 2.0 for authentication and authorization
- Enabled via GCP Console
- Most API's have daily quota's that can be raised upon request
  - Plan ahead for capacity
- Can experiment with APIs Explorer

API Explorer

- Interactive tool to easily try Google APIs using a browser
  - Browse available APIs and versions.
  - View methods, parameters, and inline documentation.
  - Execute requests for any method and see responses in real time.
  - Make authenticated and authorized API calls with ease.

- Additionally, other client libraries are able to interact with GCP APIs
  - Go, Java, Node.js, Python, Ruby, PHP, and C#

DEMO TIME

Google Cloud Platform Cloud Architect Essentials

Cloud Shell/gcloud commands

What we will cover

- Review Cloud Shell
- gcloud commands overview
  - Where to find command reference
- Hands on demo

Cloud Shell

- Pre-configured Google SDK Linux instance
- Automatic authentication based on GCP Console login
- Accessible from any web browser within Cloud Console
- All client libraries for web applications pre-installed
  - Includes built in code editor
- 5GB persistent storage
- Preview web apps before deployment
- Nothing to install!

Limitations

- 1 hour time out for inactivity
  - Machine will terminate/self-delete
  - $HOME directory contents will be preserved for new session
- Direct interactive use only
  - Not for running high computational/network workloads
  - If in violation, session can be terminated without notice
- For long periods of inactivity, home disk may be recycled (with advance notice via email)
  - If need longer inactive period, consider either local installed SDK or use Google Cloud Storage for long term storage

gcloud commands

- For Cloud Shell and local installed Google Cloud SDK
- Manage Google Cloud Platform resources and developer workflow
- (Typical) command format:
  - gcloud [GROUP] [GROUP] [COMMAND]—arguments

- Examples:
  - gcloud compute instances create instance-1 --zone us-central1-a
  - gcloud config set project my-unique-project-id

- Full gcloud command list at https://cloud.google.com/sdk/gcloud/reference/

Hands on demo

Google Cloud Platform Cloud Architect Essentials

Compute Options

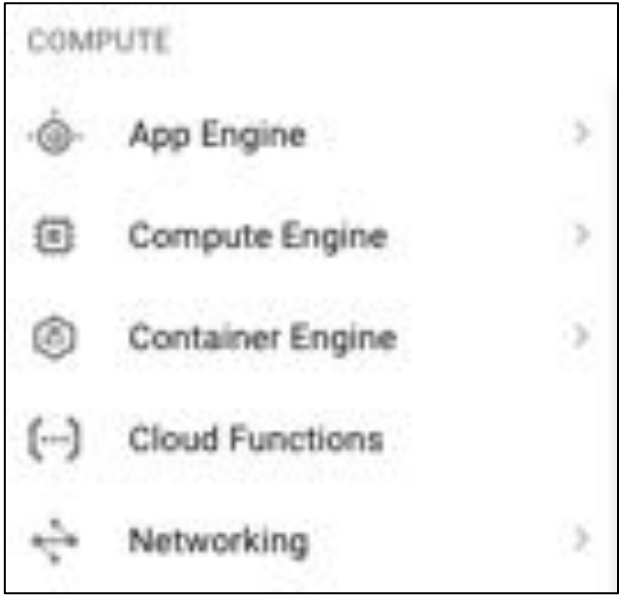
Choosing Correct Compute Option

Multiple Compute Options for Hosting Applications

Compute Option = method for hosting applications on GCP

- Google Compute Engine
- Google App Engine
- Google Container Engine
- Google Cloud Functions (Beta)

- Each option can take advantage of the rest of GCP services
  - Storage
  - Networking
  - Big Data
  - Security



Role of the Google Cloud Architect

Given a set of business and technical requirements, know how to choose and implement the right tool for the task

Which one should we pick?

- Choose the best option for your requirements
- Each option has their pros and cons
- Sliding scale of flexibility vs. managed
- Can combine options

Compute Options Overview – Compute Engine

- Infrastructure as a Service (IaaS)
- Virtual machines – referred to as '**instances**'
- Virtual version of physical PC
  - CPU/GPU
  - Memory
  - Disk space
  - OS
  - Firewall controls
  - Network connection/management (VPN, load balancing)
- Offers complete control and most flexibility, but also comes with most operational overhead.
- Great fit for organizations moving existing servers into the cloud and are used to managing VM's

Compute Options Overview – Container Engine

- Deploy containerized applications
  - De-couples application components from OS
  - Run app in multiple environments, regardless of OS
- Powered by Kubernetes – open source container orchestration system
- “Manage applications, not machines”

Compute Options Overview – App Engine

- Platform as a Service (PaaS)
- Developers can focus on writing code, while Google handles the rest
- Build scalable web applications and mobile backends
- **Managed** service
  - Never touch the underlying infrastructure
  - Deployment, maintenance, and scalability handled
  - Reduces operational overhead
- Standard and Flexible environments
  - Standard supports Python, Java, PHP, Go
  - Flexible natively supports Java 8 / Servlet 3.1 / Jetty 9, Python 2.7 and Python 3.5, Node.js, Ruby, PHP, .NET core, and Go, plus any other runtime if using a custom Docker image

Cloud Functions (BETA)

- Serverless environment for building and connecting cloud services
- Event-driven
  - Function executes as a ‘trigger’ in response to a cloud based event
  - Simple, single purpose functions
- Written in Javascript and execute in Node.js
- Example: A file is uploaded to Cloud Storage (event), function executes in response to event (trigger)
- Easier and less expensive than provisioning a server to watch for events

Comparisons – What do you want to think about?

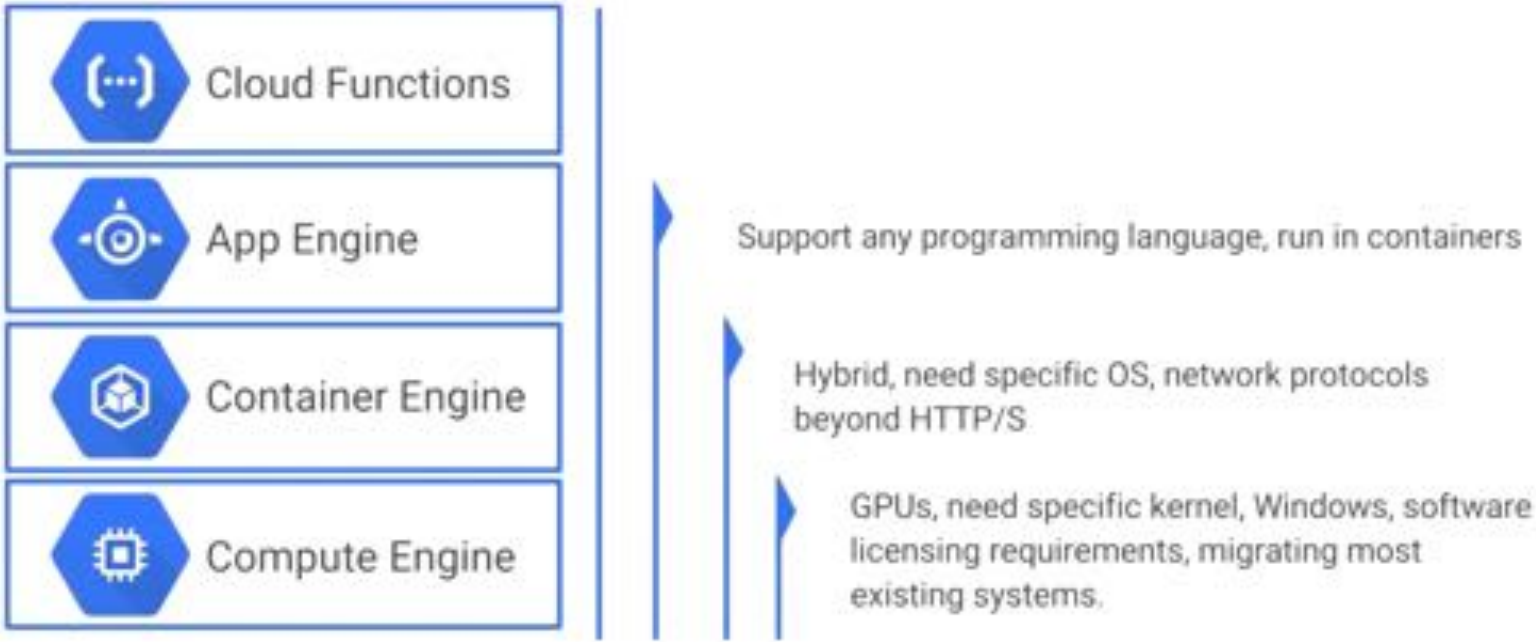
Events
Function definitions

Code
HTTP Requests

Applications, not computers or containers
What programs? How are they connected? State?

Your software, Operating system / disk images
CPU, RAM, Disk
Networking: Firewall rules, Load balancers, VPNs

Comparisons – Technical Requirements



To summarize:

Google Cloud Platform Cloud Architect Essentials

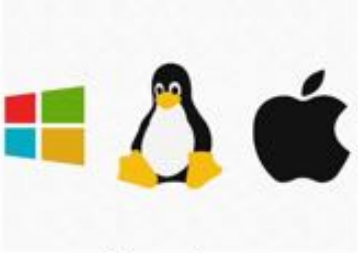
What is Infrastructure as a Service (IaaS)?

What is Infrastructure as a Service?

- One of three cloud computing models, the other two being Platform as a Service (PaaS) and Software as a Service (SaaS)
- Often referred to as the core layer of cloud computing. Behind the scenes, PaaS and SaaS are running on an IaaS layer.
- It is a form of cloud computing that provides virtualized computing resources over the Internet
- Deals with Virtual Machines and (in some cases) virtual networks

That's great…what is 'infrastructure'?

Traditional infrastructure (PC/server) components



Operating System

CPU

Memory (RAM)

Hard Drive

Network Access

Firewall

Virtual Machine defined

- Virtual version of a physical PC/server. Just as a physical computer has a CPU, memory, hard drive, an operating system, network access, and firewall to manage network traffic, so does a virtual machine.
- Like a physical machine, it is up to the customer to manage the above aspects, as well as perform typical OS maintenance such as updates.
- However, Virtual Machines have many advantages compared to traditional physical infrastructure.

Google Cloud Platform Cloud Architect Essentials

Compute Engine Overview

What is Google Compute Engine?

Compute Engine is **IaaS** running **virtual machines (instances)**

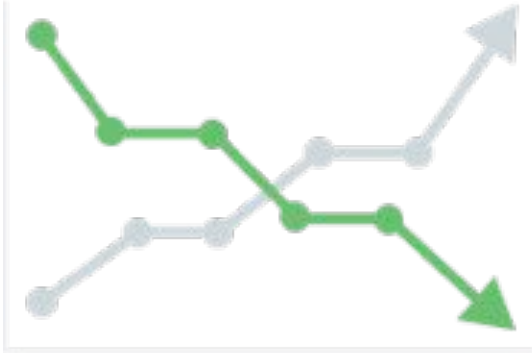
- Run high scale workloads on Google's infrastructure
- Create Linux and Windows instances
- Robust network features
  - Custom networks
  - Firewall rules
  - Regional HTTP(s) load balancing
  - Network load balancing
  - Subnetworks

What makes Google Compute Engine unique?

Industry Leading Price & Performance

- Compute Engine VMs boot quickly and are consistently high performance.
- Offers **local solid state drive (SSD) performance**.
- Positioned as the higher performance option
  - “Google Compute Engine ranked #1 in price-performance” (https://lp.google-mkto.com/rs/248-TPC-286/images/Cloud-Spectator-Best-Hyperscale-Cloud-Providers.pdf)

What makes Google Compute Engine unique?



Low Cost + Automatic Discounts

- Google bills in **minute-level increments** (with a 10-minute minimum charge)
- **Automatic** sustained use discounts for long-running workloads
    - No up-front commitment required

High speed, secure, private network

- Global private fiber network
- Connections between data centers on private high speed connections
- Same network infrastructure used by Google

What makes Google Compute Engine unique?

Extremely Flexible

- Create custom images
- Low cost Pre-emptible VM's for batch workloads
- Custom machine types (CPU/memory specs)
- Resize disks with no downtime
- Create metadata and startup scripts

Google Cloud Platform Cloud Architect Essentials

Basic Instance Management

For the rest of this section

- Micro-lessons
- Cover the ‘basics’
- We will learn:
  - Create, manage, and delete instances
  - Configure ‘basic’ instance options before and after creation
  - Connect to Linux and Windows instance
  - Attaching and resizing disks
  - Create custom image
  - Snapshots
  - Some gcloud command line reference
  - Preview advanced concepts (e.g. instance groups)

Our goal:

- Become comfortable creating, managing, and connecting to Linux and Windows instances

Google Cloud Platform Cloud Architect Essentials

Cloud Launcher

What is Cloud Launcher?

- “The fastest way to get started on GCP”
- Quickly deploy functional software packages that run on GCP
- Search for different solutions
  - In some cases, several variations of the same solution
- Manage and view info with Deployment Manager
- Most packages are free minus normal usage fees, however some are billed through a third party at an additional cost

Google Cloud Platform Cloud Architect Essentials

Networking Overview

High level overview of:

- Virtual Private Cloud (VPC) networks
- External IP addressing
- Firewall Rules
- Routes
- Load Balancing
- Cloud DNS
- VPN/Cloud Routers
- Cloud Content Delivery Network (CDN)

Virtual Private Cloud (VPC) Networks

- Virtual version of the traditional physical network
- A VPC network provides connectivity for your Compute Engine instances, Container Engine containers, and App Engine Flex services
- All GCP projects start with a ‘default’ network
- Resources in a VPC have an internal (private) IP address that is drawn from its subnet
- All other Networking services are tied to a VPC

External IP addressing

- Compute Engine instances have 1 or more network interfaces, each with an IP address
  - Can be internal or external (or both)
- External IP's are either ephemeral or static
  - Ephemeral by default, and change when instance is restarted
- Static IP's can be reserved and attached to an instance

Firewall rules

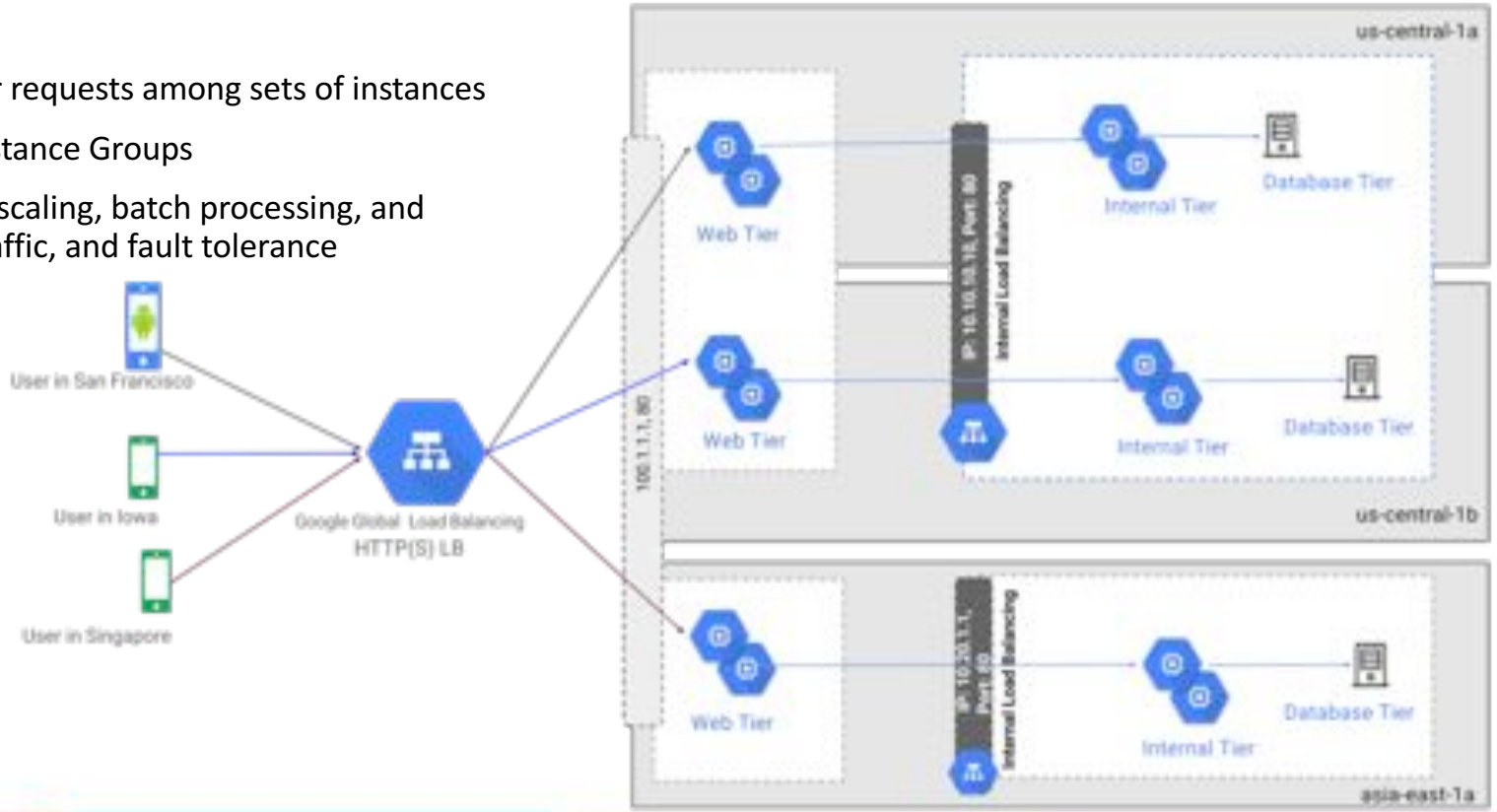
- Every VPC network has a managed firewall
- Manages both inbound (ingress) and outbound (egress) traffic
- Can be applied to entire VPC network or individual instances

Routes

- ‘Roadmap’ of how to get from point A to point B
- Mapping of an IP range to a destination
- Tell a VPC network what route to take to send packets to an IP address
- Can either use default routing table, or create custom rules

Load Balancing

- Distribute user requests among sets of instances
- Works with Instance Groups
- Used for auto-scaling, batch processing, and distributing traffic, and fault tolerance



Cloud DNS (Domain Name System)

- DNS translates a computer domain names (like google.com) into IP addresses
- Google Cloud DNS is a high-performance, resilient, global Domain Name System (DNS) service that publishes your domain names to the global DNS in a cost-effective way.
- Create managed zones
    - Add, edit, and delete DNS records

Virtual Private Network (VPN) and Cloud Routers

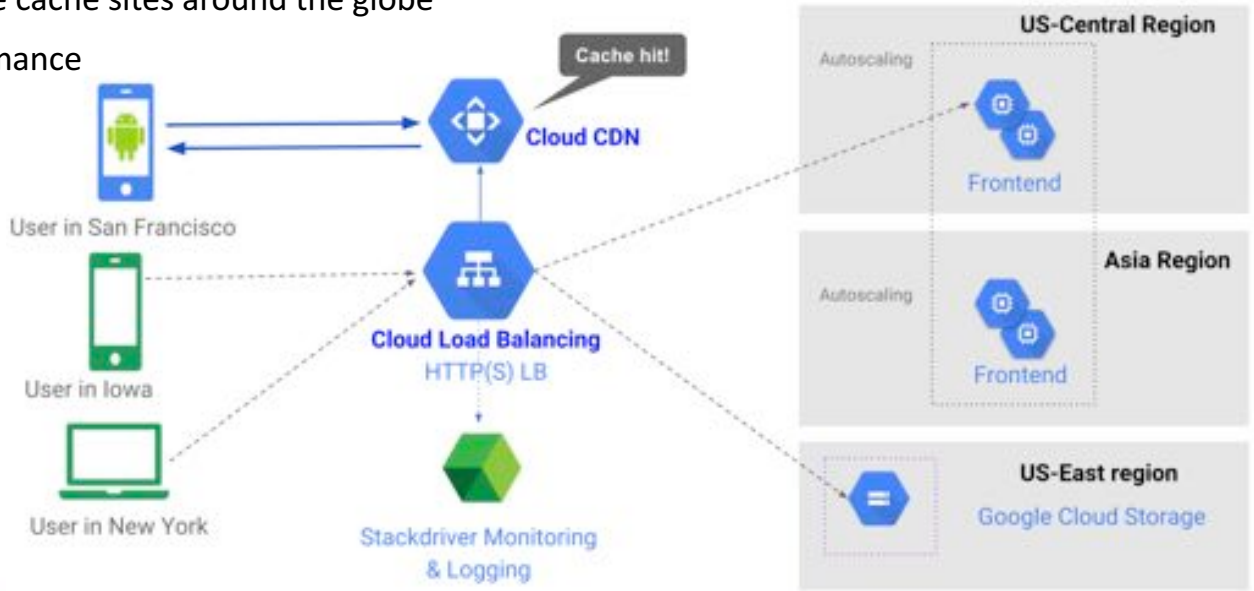
- Securely connect on premise network location to GCP through an IPsec encrypted connection
  - Private, internal connection over the public Internet
  - Supports gateway to gateway connections (i.e. VPN router to GCP VPN network), not ‘road warrior’ client VPN

Cloud Router

- Supports dynamic routing for Google VPN
- Managed service that handles routing for you

Cloud Content Delivery Network (CDN)

- Simply defined, CDN places your online content closer to your users for faster response times
- Works with load balancing
- Content is ‘cached’ in 80+ edge cache sites around the globe
- Closer content = faster performance



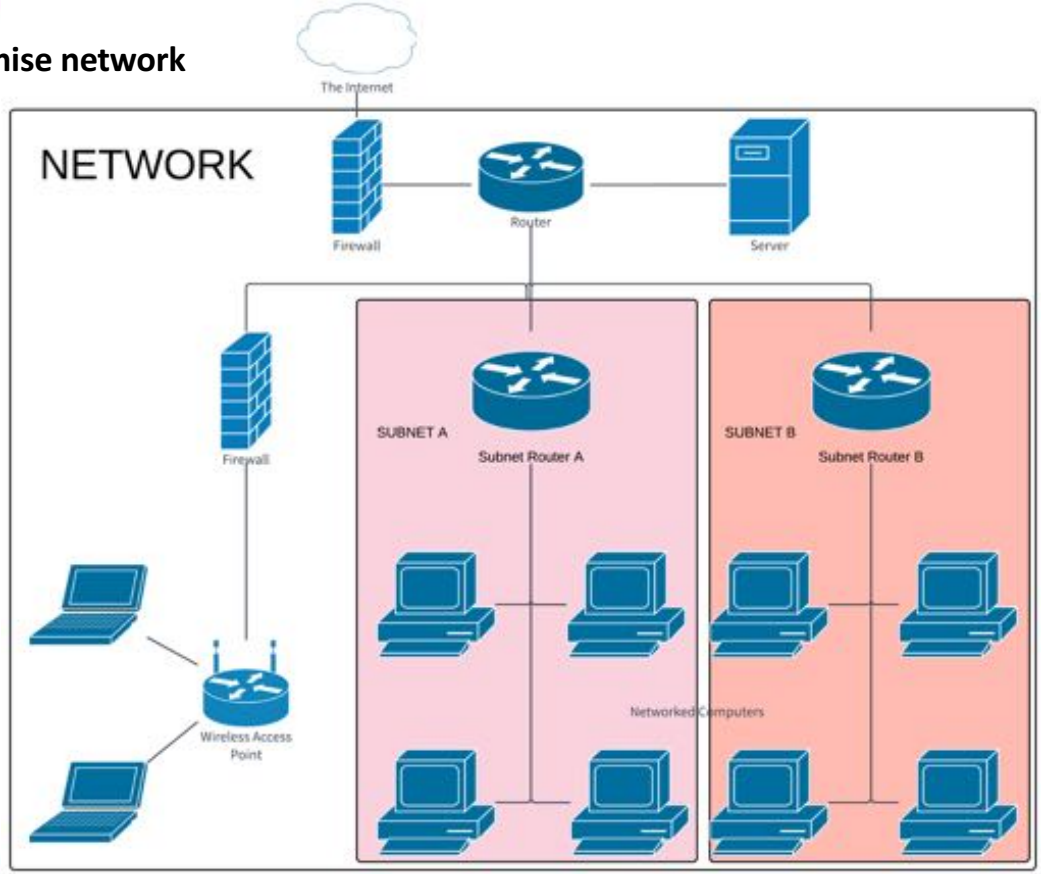
Google Cloud Platform Cloud Architect Essentials

Virtual Private Cloud (VPC)

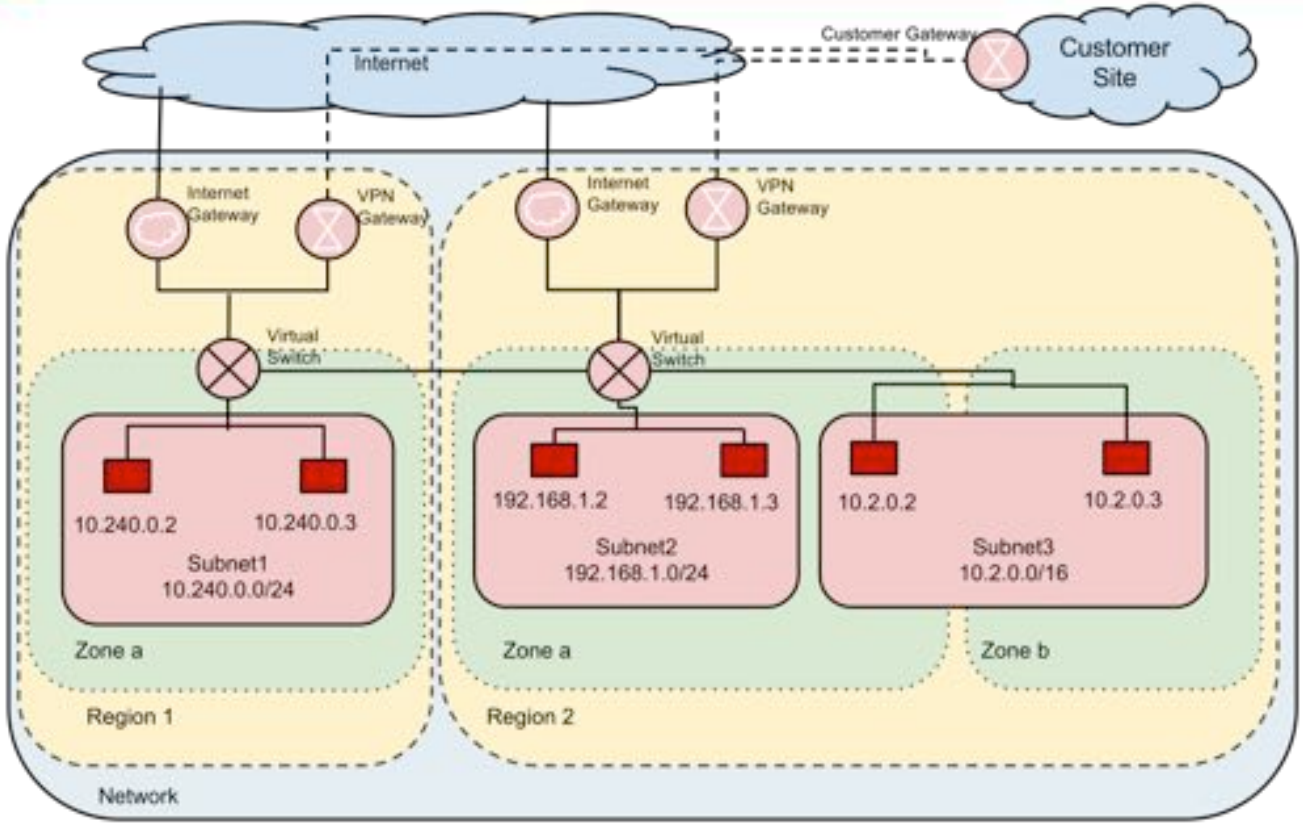
What is VPC?

- A VPC network is a virtual version of the traditional physical networks
- Exist within and between physical GCP data centers
  - Network and subnets and span regions and zones
- All GCP networking resources are assigned to a VPC network

Traditional on-premise network



VPC network


Internet
Customer Gateway
Customer Site
Internet Gateway
VPN Gateway
Internet Gateway
VPN Gateway
Virtual Switch
Virtual Switch
10.240.0.2
10.240.0.3
Subnet1
10.240.0.0/24
Zone a
Region 1
192.168.1.2
192.168.1.3
Subnet2
192.168.1.0/24
Zone a
10.2.0.2
10.2.0.3
Subnet3
10.2.0.0/16
Zone b
Region 2
Network

Google Cloud Platform Cloud Architect Essentials

IP Addressing and Firewall Rules

Instance IP addressing

- All instances come with a private (internal) IP based on their subnet
- Optionally can have a public IP address (defaults to enabled)
  - Ephemeral – not reserved, and changes when instance stopped
  - Static (reserved) – reserve own IP address to permanently attach to a resource
  - Instances can only have one external address
  - Unassigned static IP addresses cost $.01 per hour

Firewall rules

- Protect resources from unapproved network connections
- Both inbound (ingress) and outbound (egress) are managed
- Allow or deny
- Based on IP addresses, ports, and protocols
- Can be applied to an instance, or the entire network

Google Cloud Platform Cloud Architect Essentials

Operations and Management Tools

Google Stackdriver

- Provides powerful monitoring, logging, and diagnostics for cloud operations
- Natively monitor GCP, Amazon Web Services (AWS), or a hybrid of both environments
  - Combine metrics, logs, and metadata from both platforms into a single viewing environment
- Robust partner ecosystem to make working
  - PagerDuty, BMC, Splunk, and others

Stackdriver components

- Stackdriver Monitoring
  - Full-stack monitoring for Google Cloud Platform and Amazon Web Services.
- Stackdriver Logging
  - Real-time log management and analysis.
- Stackdriver Error Reporting
  - Identify and understand your application errors.
- Stackdriver Debugger
  - Investigate your code's behavior in production.
- Stackdriver Trace
  - Find performance bottlenecks in production.

Google Cloud Deployment Manager

Infrastructure Management Service

- Specify resources needed for your application in a declarative format using yaml
  - YAML = YAML Ain't Markup Language
  - Human-readable structured data format
  - Lists each of the resources in your deployment
- Repeatable deployment process
- Declarative language
- Focus on the application
- Template-Driven

Google Cloud Source Repositories

- Git repository hosted on GCP
- Built in Source Code Editor
- Integrates with Stackdriver debugger
- Connect to GitHub or Bitbucket

Google Cloud Platform Cloud Architect Essentials

App Engine and PaaS Overview

As we go forward in this course…

- More command line usage
- Application development focus

What is Platform as a Service (PaaS)?

- Definition: PaaS is a category of cloud computing that provides a platform and environment to allow developers to build applications and services over the internet without having to worry about allocating and managing infrastructure.
- Popular with software and web developers
- Key distinctions vs. IaaS:
  - Focused on application development
  - Managed infrastructure
  - Pay per use vs. pay per allocation

What is Google App Engine?

- GCP's tool to build modern web and mobile applications on an open cloud platform
- Fully managed application platform
- Supports your own language runtimes, frameworks, and third party libraries
  - 'out of the box' environment with default configurations, App Engine supports Node.js, Java, Ruby, C#, Go, Python, PHP, and .Net
  - Support for custom Docker images/runtimes
  - Portable – no vendor lock in

App Engine

Simply defined:

You bring your code, Google handles the rest

What is 'the rest'?

- With traditional physical infrastructure, you need to manage:
  - Firewalls
  - Denial of service attacks
  - Viruses
  - Patches
  - Network configurations
  - Failover
  - Load balancing
- Capacity planning (scaling)
- OS patches and upgrades (in particular security related)
- Hardware upgrades or fixes
- Certification levels
- Most security issues
- Routing
- IP addressing

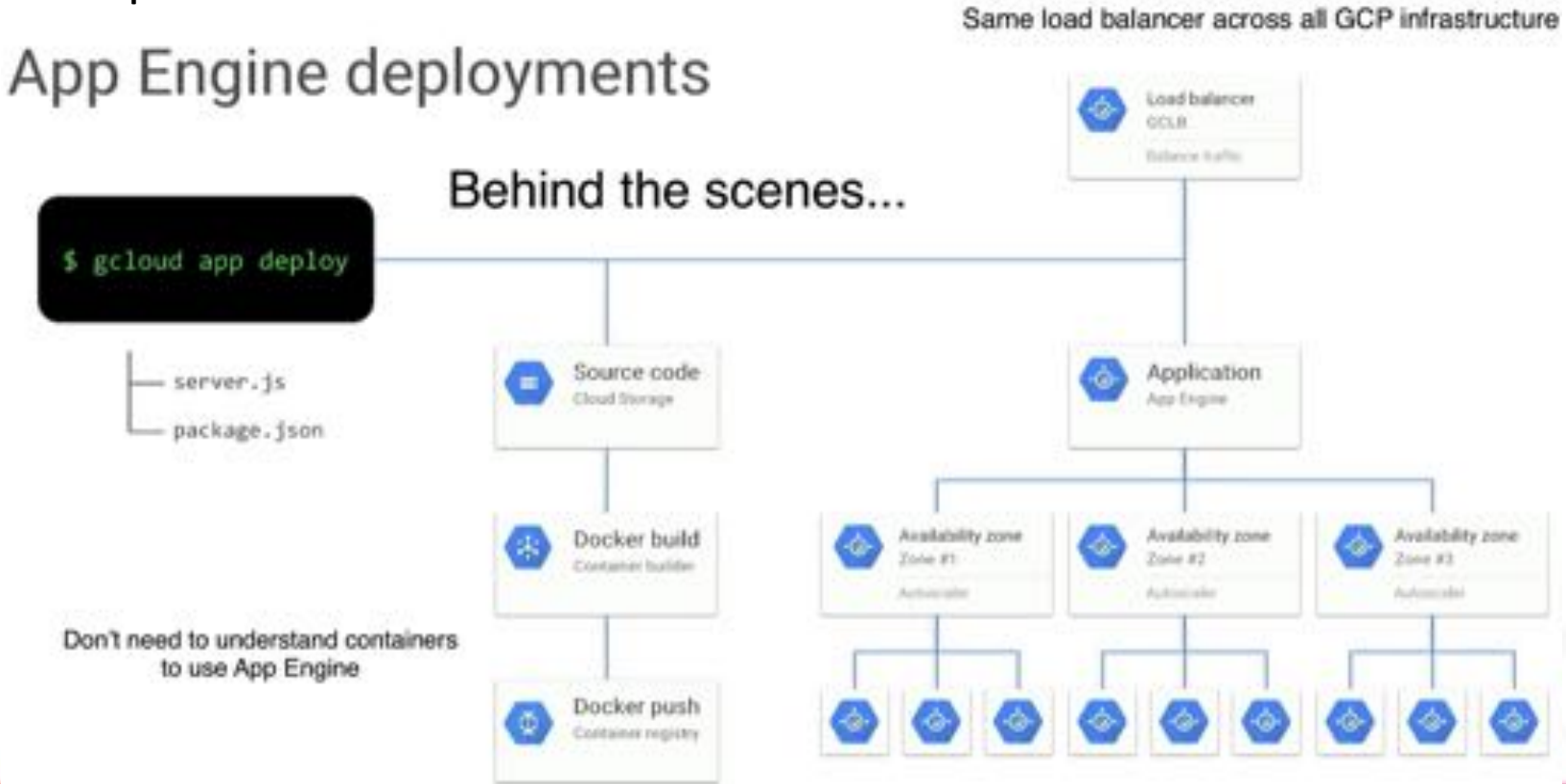
Manage all of this **in addition to** focusing on your code

- High amount of administrative overhead
- IaaS solutions (like Compute Engine) help with some of the list, but App Engine handles **all** of it for you

Why is this important?

- We put a lot of work into our code
- It really stinks when a machine, or machine configuration sinks our efforts
- Defer machine management to someone else who has been doing it for years
- FOCUS ON CODE

Visual representation



Compute Engine vs. App Engine

App Engine Standard and Flexible environments

Standard Environment

- Managed runtime for specific versions of Java, Python, PHP, and Go
- Rapid, automatic scaling to meet spikes in demand
  - Go from zero to ‘Google-scale’, fast
- Runs in a secure, **sandboxed** environment
  - Within secure environment that is independent of the hardware, operating system, or physical location of the server
- More restrictions than Flexible environment
  - E.g. cannot write to local file system or modify runtime
- Charged on instance hours (how often it’s used) – machine type not calculated

Flexible Environment

- Based on Compute Engine
- Automatically scales application up and down based on load (load balancing)
- Native support for Java 8 / Servlet 3.1 / Jetty 9, Python 2.7 and Python 3.5, Node.js, Ruby, PHP, .NET core, and Go
  - Or provide own runtime by supplying own Docker image
- Customize infrastructure: custom libraries, use SSH, or use own Docker containers
  - Specify CPU/memory configuration, and Flexible environment will provision for you
- Charged by CPU, memory, and disk usage
- Compared to Compute Engine:
  - Instances are automatically updated/patched
  - Health check handled for you
  - Region placement handled for you for optimal performance
  - Root SSH disabled by default (but can be enabled)

Choose Standard or Flexible?

Standard

- More constraints
- Python 2.7, Java 7/8, PHP 5.5, Go 1.6
- Faster scale up time (seconds)
- Intend to run for free or very low cost
    - Only pay for what you need when you need it (if no traffic, no instances in use)

Flexible

- Less constraints, more customization, but more management
- Python, Java, Node.js, PHP, Go, Ruby, or .NET
    - Or any other in own Docker container
- Slower scale up time (minutes)
    - Consistent traffic, gradual scale up/down
- Load OS-dependent packages

App Engine locations

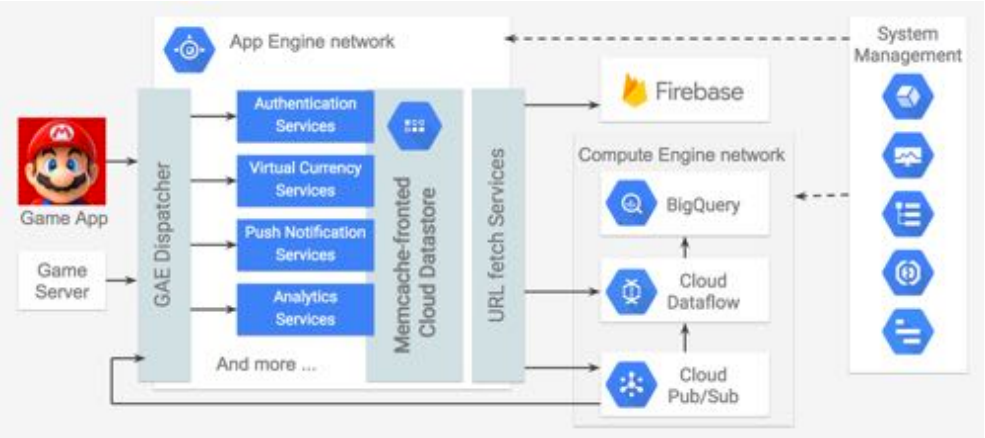
- App Engine is *regional*
  - Redundantly available across all the zones within that region

- App Engine is available in the following regions:
  - us-central1
  - us-east1
  - us-east4
  - europe-west1
  - europe-west2
  - asia-northeast1
  - australia-southeast1

Who uses App Engine?

Nintendo and DeNA – Super Mario Run

- Simultaneous worldwide launch
  - Launch in 150 different countries at the same time
  - Any downtime would be unacceptable
  - Needed to handle millions of users from day one



- App Engine is able to scale up and down with volume within milliseconds

Google Cloud Platform Cloud Architect Essentials

App Engine Hands On

Before we begin…

- Greater focus on Linux command line and app development concepts

In this lesson…

- Deploy a sample Python application called Bookshelf to the App Engine standard runtime environment
- Test the Bookshelf application and inspect data saved to Cloud Datastore
- App deployment is entirely via command line, no web console option
- Reference: https://codelabs.developers.google.com/codelabs/cp100-app-engine

Cloud Datastore (preview)

- Database designed for application backends
- NoSQL – store for billions of rows
- Auto-scaling and fully managed database solution

Steps to App deployment

- Step 1 – ‘git’ the code
- Step 2 – review the code
- Step 3 – install requirements
- Step 4 – deploy (gcloud app deploy)

Step 1 – ‘git’ the code

- git commands
- Create default GCP Source Repository
- Link the console GCP Source Repositories with Cloud Shell
- Pull the public example code into our Cloud Shell instance
- Push code into GCP Source Repository

- Note: if you want to ‘cheat’ and don’t care about GCP Source Repositories, simply ‘clone’ instead.

‘git’ the code (cont)

- From web console: Enter bookshelf project
  - Tools → Source Repositories → CREATE REPOSITORY → name it ‘default’
- From Cloud Shell:

mkdir cp100	Create directory we’re placing code into
cd cp100	Enter directory we just made
gcloud source repos clone default	Clone our (empty) Source Repository from web console step – also links it
cd default	Enter default directory in cp100 folder
git pull https://github.com/GoogleCloudPlatformTraining/cp100-bookshelf	Clone (or ‘pull’) copy of code from Google’s github page into our Cloud Shell default folder
git push origin master	‘Push’ our Cloud Shell copy to Source Repository

Step 2 – review the code

- Back in Source Repositories
- Go to default repository – no longer empty! → Go to app-engine
- app.yaml – tells App Engine (via 'gloud app deploy') what to do
  - Required and optional configuration data used to deploy and manage app
- main.py – imports bookshelf code and loads configuration data
  - (also contains code if running locally)
- config.py – manage configuration data
  - In this case, specifies Google Datastore for our back end
- crud.py – coordinate create, read, update, delete (or c.r.u.d.) operations

Step 3 - Install requirements

- From app-engine directory, open requirements.txt
  - Required libraries and dependencies to local cloned repository (i.e. Cloud Shell) before we can deploy
- Run command to install install required Python packages
  - pip install -r requirements.txt -t lib

Step 4 – Deploy!

- From Cloud Shell:
  - Make sure we're in the app-engine directory
  - gcloud app deploy
  - Confirm information is correct

Google Cloud Platform Cloud Architect Essentials

Cloud Endpoints

What is Cloud Endpoints?

- Create, deploy, protect, monitor, analyze, and serve your APIs
- Uses the same infrastructure Google uses for its own APIs

What are API's?

- **Application Programming Interface** (**API**)
- Subroutine definitions, protocols, and tools for building applications
- Standardized interface for developers to build applications
  - i.e. standard, reusable building blocks
  - Only exposes necessary information
  - Securely provide connection components to your app

- Example: Google Drive API
  - Developer can integrate Drive access in their application
  - Developer doesn't need your Google login for app to access Drive

So what does Cloud Endpoints do?

- Build your own API on App Engine Standard
- Expose your API using RESTful interface
- Oauth 2.0 authorization
- Supports Python and Java
- Comes with same benefits of App Engine
    - Scaling, DOS protection, and high availability

Google Cloud Platform Cloud Architect Essentials

Storage Options Overview

Storage Options

- “Where am I going to put my data?”
- “What is my data’s format?”

Different storage types need different storage - breakdown

Is your data structured or unstructured?

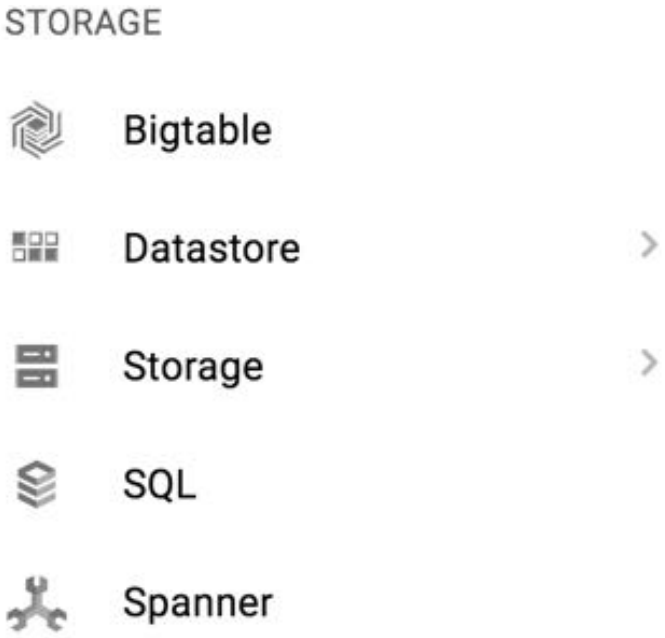
- Structured – database
- Unstructured – Google Cloud Storage

Running analytics?

- Yes = Bigtable
- No = the rest

Is your database relational or non-relational?

- Relational = SQL
- Non-relational = NoSQL



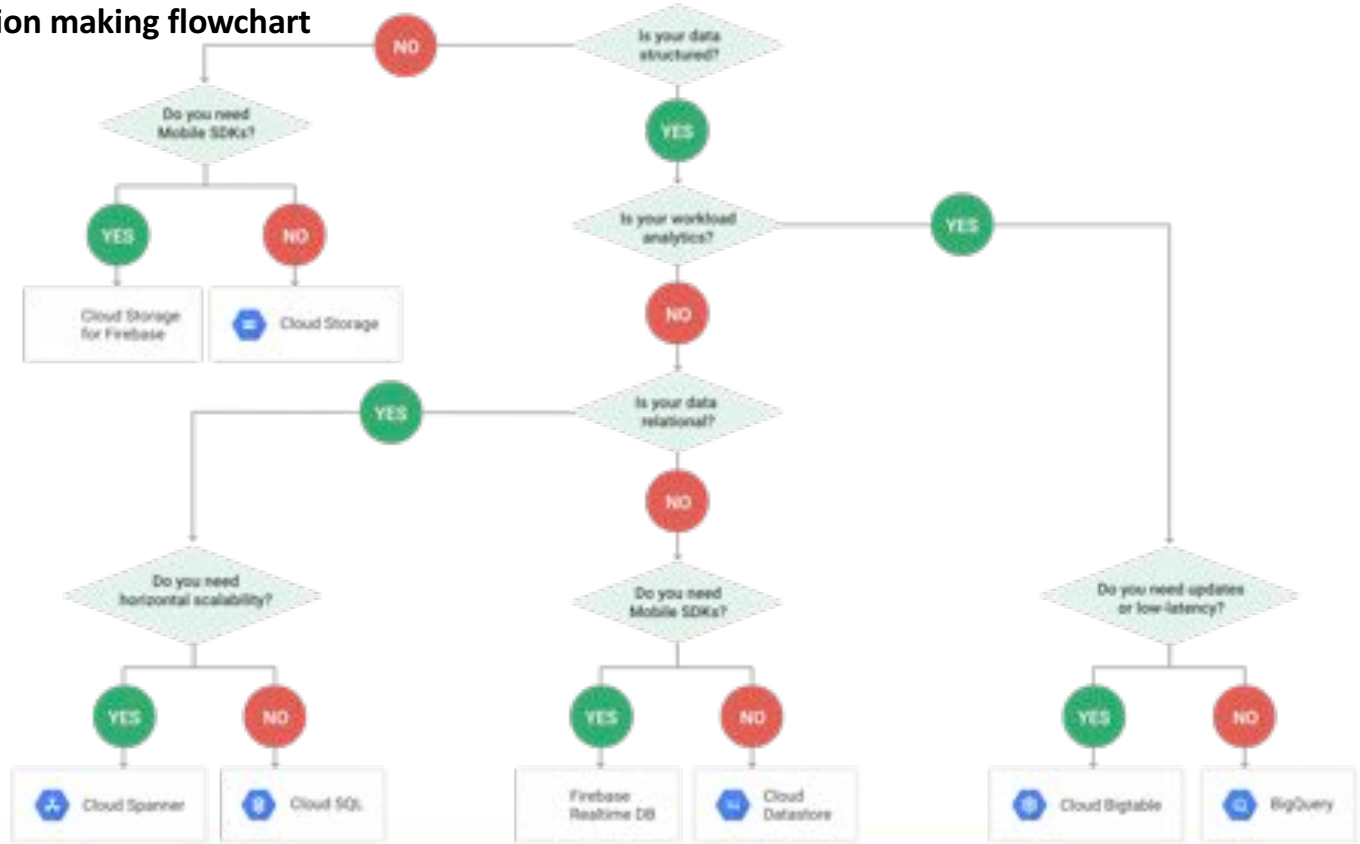
What is SQL/NoSQL?

- SQL = Server Query Language
- SQL is known for consistency, not as much for scalability/flexibility
- NoSQL known for scalability/flexibility, but not as consistent as SQL
- Both are blurring lines toward each other
- SQL uses data tables/rigid design (think Excel/spreadsheets)
- NoSQL table-less, easier to manage
- SQL based on ACID properties (Atomicity, Consistency, Isolation, Durability) to guarantee validity

The database breakdown

- SQL = SQL
- NoSQL = Datastore, Bigtable
- Spanner = new category

Decision making flowchart



Google Cloud Platform Cloud Architect Essentials

Database Options

Database Offerings

Cloud SQL = SQL

Datastore = NoSQL (but with some SQL aspects)

Bigtable = NoSQL

Spanner = new category

Google Cloud SQL

- Create instance/region/size
- Hosted MySQL service
  - Not similar to My SQL, it IS MySQL
- Low maintenance instance
  - Instanced, but OS management/updated handled for you
- Vertical scaling (read/write), horizontal scale (read)
- Limited scalability

Google Cloud Datastore

- Born as App Engine repository
- Scale and flexibility
- No provisioning resources, true NoOps
  - Fully managed
- Scale from 0 to terabytes of data
- Cost efficient
- Supports ACID transactions

Google Cloud Bigtable

- Managed NoSQL analytics for massive amounts of data
  - Terabytes to Petabytes
- Hosted version of Google's own internal Bigtable technology
  - Gmail and Google Analytics
- HBase was born from Bigtable
- High volume writes
- Latency in millisecond range
- Pricier than Datastore, costs whether using it or not

Google Cloud Spanner

- New offering
- Relational database (like SQL)
- ….but with much better scalability
  - Horizontally scales
- Billed as the best of both worlds

	CLOUD SPANNER	TRADITIONAL RELATIONAL	TRADITIONAL NON-RELATIONAL
Schema	✓ **Yes**	✓ Yes	✗ No
SQL	✓ **Yes**	✓ Yes	✗ No
Consistency	✓ **Strong**	✓ Strong	✗ Eventual
Availability	✓ **High**	✗ Failover	✓ High
Scalability	✓ **Horizontal**	✗ Vertical	✓ Horizontal
Replication	✓ **Automatic**	◌ Configurable	◌ Configurable

Common use cases

- Cloud SQL – web frameworks, content management systems, eCommerce
- Datastore – user profiles, product catalogs, game states
- Bigtable – high throughput analytics, Internet of things (IoT), ad tech
- Spanner – scale + consistency, financial services, global supply chain

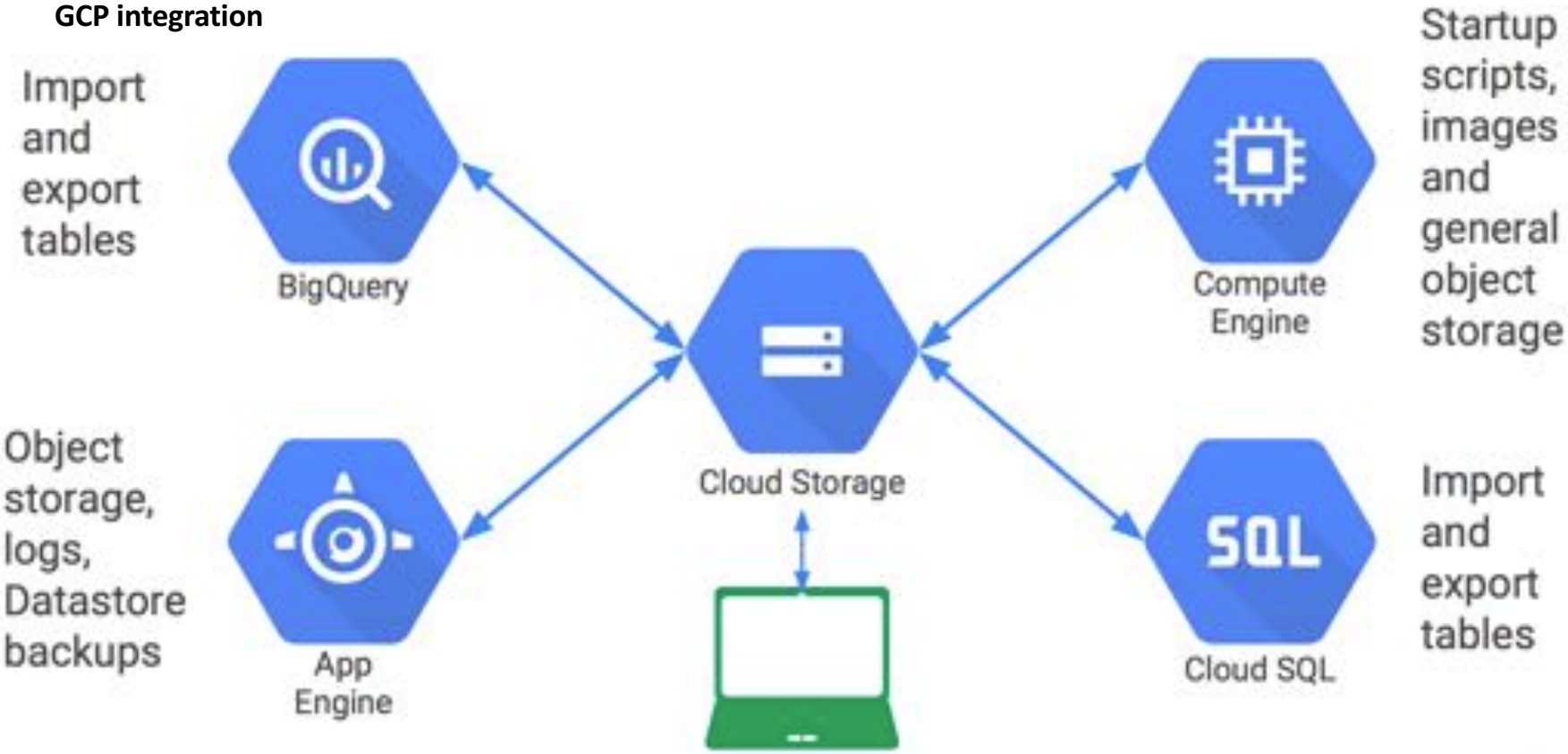
Google Cloud Platform Cloud Architect Essentials

Cloud Storage

What is unstructured data?

- Data not contained in a database or organized in a pre-defined manner
- Examples: pictures, videos, objects, documents, text files, multimedia, file backups
- Sometimes referred to as ‘BLOB’ storage (binary large object)

GCP integration



Google Cloud Storage

- Unified Object Storage
  - Unified offering across the availability spectrum for many applications
  - e.g. live analytics streaming to backup solutions
- Extremely price competitive
- Pay per use, not pay per allocation. i.e. no capacity management
  - High scalability
- Multiple storage classes based on storage needs (access by same API)
- **Not** a file system, but can be set up like one via 3rd party tools
- Data encrypted in transit and at rest

Cloud Storage Terminology and Organization

- Bucket – basic container that holds your data
  - Cannot nest buckets
  - Recommended to have fewer buckets/more stuff per bucket (performance)
  - Bucket name must be unique to all of GCP
- Objects – individual pieces of data (i.e. files)
  - Can be up to 5TB for a single object
- Data Opacity
  - Unlike databases, Cloud Storage has no knowledge of structure/order
  - Regardless of internal structure

Storage Classes

- Multi-Regional, Regional, Nearline, Coldline
- **All** classes have same throughput, low latency, and high durability
- Differences in availability, storage durations, and pricing for storage **and** access
    - Availability range between 99.95% and 99%

Storage Classes

- Multi-Regional - 99.95% SLA
  - Geo-redundant
  - Data frequently accessed worldwide, website content, streaming videos, games
- Regional - 99.9% SLA
  - Limited to geographical region
  - Frequent access in same region (e.g. data analytics in DataProc)
- Nearline – 99% SLA
  - Very low cost per GB stored, data retrieval cost, 30 day minimum duration
  - Infrequent access (e.g. regular data backups)
- Coldline – 99% SLA
  - Even lower cost per GB stored, even higher data retrieval cost, 90 day minimum duration
  - Cold data (e.g. disaster recovery, archived data)

Storage Classes

	ACCESS FREQUENCY	AT REST PRICING	RETRIEVAL PRICING	SLA
Multi-Regional	Frequent, Cross-regional	$0.026 per GB/month	FREE	99.95%
Regional	Frequent, Single-region	$0.02 per GB/month	FREE	99.9%
Nearline	Less than once per month	$0.01 per GB/month	$0.01 per GB	99.0%
Coldline	Less than once per year	$0.007 per GB/month	$0.05 per GB	99.0%

Google Cloud Platform Cloud Architect Essentials

What are Containers and Kubernetes?

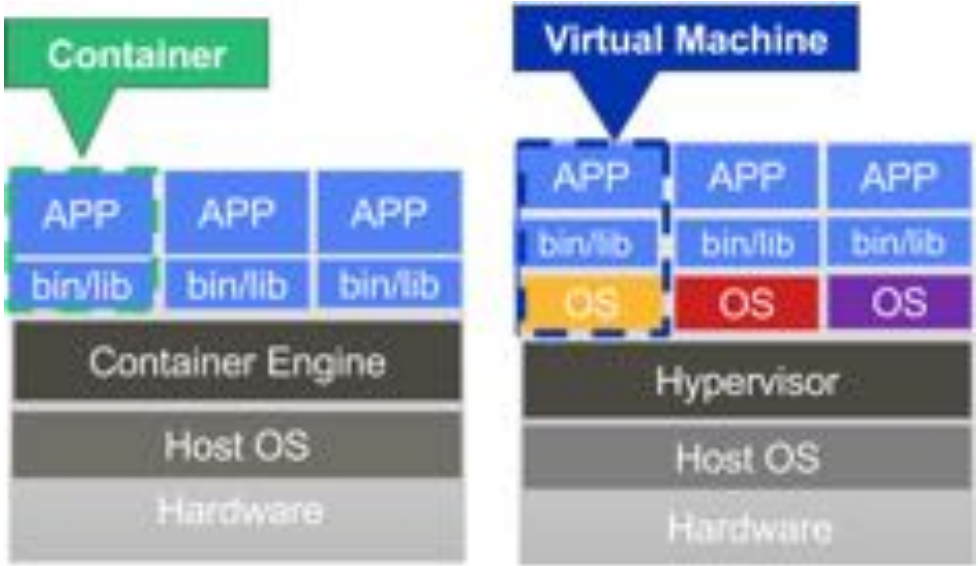
What are Containers?

- Method of OS virtualization
- Contains entire runtime environment, including application code, configurations, and dependencies, libraries and other binaries - and packages them into self-contained building blocks
  - De-couples application from host OS
- Reliably run across different environments
  - Development, test, staging, and productions environments
    - No more “it works on my machine…”
  - Physical PC’s to VM’s
  - Previously, different software environments could break programs

Like a shipping container

How are containers different from standard VM's (IaaS)?

- A VM contains an entire operating system packaged with the application
- A container only runs OS kernel. It is much more lightweight and uses less resources



Other container advantages

- Faster start time since no host OS to boot
- Smaller size
- Broken down into easier to manage modules
    - Updating application only required updating the needed module

Docker – popular container implementation

- For this course – Docker and containers are synonymous

What is Kubernetes?

- Greek for ‘helmsman’
  - Piloting a cargo ship
- Open source container manager
- Created by Google for internal use, now open source for anyone
- Automates deployment, scaling, and management
- Container management engine for Google Container Engine

kubernetes

Benefits of Kubernetes

- Deploy your applications quickly and predictably
- Scale your applications on the fly
- Roll out new features seamlessly
- Limit hardware usage to required resources only
- Run applications across different environment and cloud providers

Necessary terminology (not exhaustive)

- **Master:** Controls Kubernetes nodes.
- **Node:** Machines (instances) that perform tasks. Controlled by the Kubernetes master
- **Pod:** Group of 1 or more containers in a node
  - Share an IP address, hostname, and other resources. Abstracts network and storage away from the container, resulting in easy movement
- **Replication controller:** Ensures specified number of pod replicas are running at any one time across nodes
- **kubectl:** CLI tool for Kubernetes.

Google Cloud Platform Cloud Architect Essentials

Google Container Engine (GKE)

What is Google Container Engine (GKE)?

- Fully managed environment for deploying containerized applications
  - Uses Compute Engine resources
- Google has been running containers for over 15 years
- As a managed service – details handled for you
  - Set CPU, memory, and storage requirements, and GKE will do the rest
- Self healing – resulting in high availability and reliability
- Autoscaling – scale up and down based on demand
- Powered by Kubernetes – the open source container orchestrator that Google invented
  - No vendor lock-in
- Custom OS – Container-Optimized OS
  - Docker container runtime and all Kubernetes components

Scale of managed services

When to choose Container Engine over App Engine?

- Hybrid or multi-cloud development
- Use of protocols beyond HTTP/S
- Need multi-container solution – need orchestration
- Want to use Kubernetes

When to choose Compute Engine over Container Engine (for containers)?

- Need GPU's
- Non-Kubernetes container solution
- Migrating existing on premise solutions to cloud
- Custom OS or kernel needs

You can run Kubernetes using Compute Engine…but it's much harder

- Kubernetes the Hard Way
- https://github.com/kelseyhightower/kubernetes-the-hard-way

Cloud Infrastructure Provisioning

Setting up a CA and TLS Cert Generation

Setting up TLS Client Bootstrap and RBAC Authentication

Bootstrapping a H/A etcd cluster

Bootstrapping a H/A Kubernetes Control Plane

Bootstrapping Kubernetes Workers

Configuring the Kubernetes Client - Remote Access

Managing the Container Network Routes

Deploying the Cluster DNS Add-on

…. or you can just have all that done for you in a few clicks with Container Engine

Google Container Engine Organization/Components

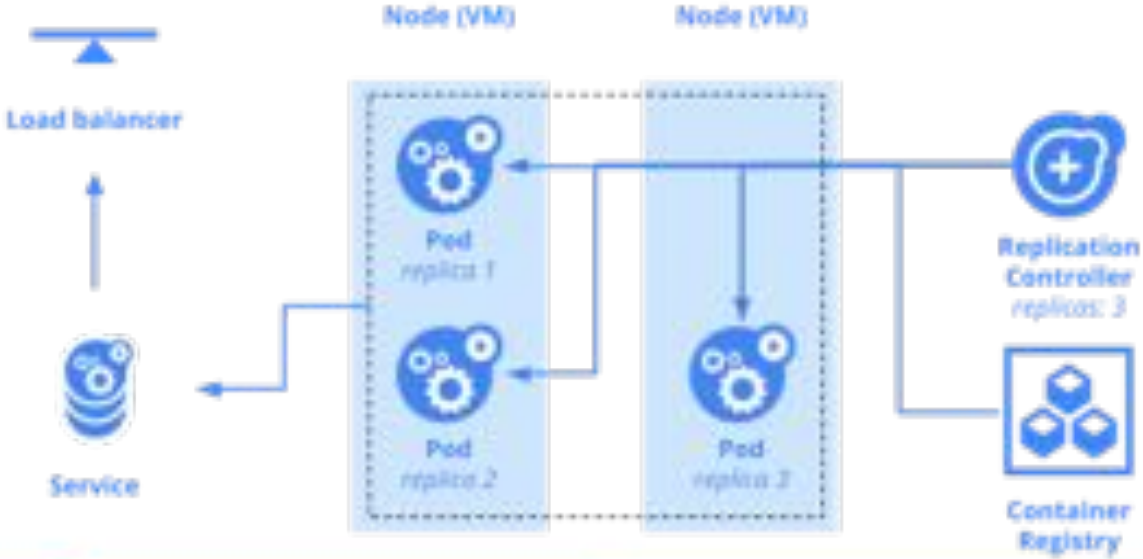
- Container cluster
  - Group of Compute Engine instances running Kubernetes
  - Contains 1 or more node instances and managed Kubernetes master endpoint
  - Central foundation of GKE - Nodes, pods, services, and replication controllers all run within cluster
- Kubernetes master
  - Manages the cluster, single endpoint
- Pods
  - Group of one or more containers
  - Share storage and configuration data among containers
  - Pods can contain multiple containers, and multiple pods can exist on each node
  - Pods have a short lifespan, and can be deleted and recreated as necessary
- Nodes
  - Individual Compute Engine instances
  - Run services to support Docker containers
  - Each node contains one or more pod

Google Container Engine Organization/Components

- Replication controller
  - Ensures requested number of pod replicas are always available
  - Automatically adds or removes pods as needed
- Services
  - Defines a logical set of pods across nodes and a way to access them using single IP address and port number
  - Services create an abstraction layer that decouples frontend clients from backend pods
- Container Registry
  - Not part of the Container Engine, but a separate service
  - Secure, private Docker image storage for deployments
  - Push container images to registry for deployment to GKE, GCE, or your own hardware

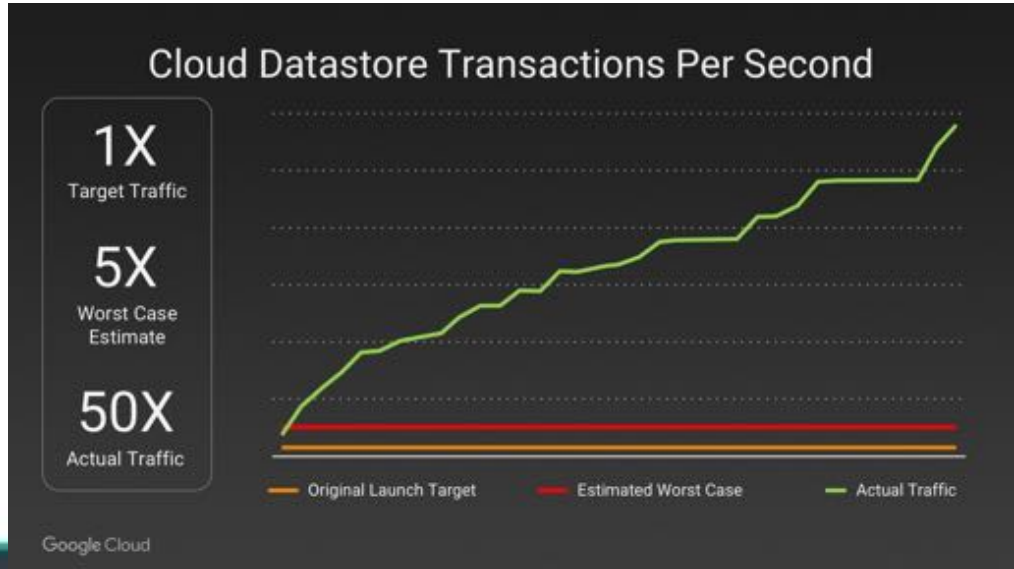
Putting it all together…

- GKE uses container cluster to manage all other components with Kubernetes master
- Container images are grouped into pods (shared data)
- Pods are replicated across nodes (instances that do the work)
- Replication controller both kills and duplicates pods across nodes as needed
- Services give a single point of access, without worrying what pod is where
- Container Registry stores container images for easy deployment



Container Engine case study: Niantic - Pokemon Go

- Explosive growth on launch day – over 50x expected amount
  - 10x worse than initial worse case estimates
- Powered by GKE
  - Deployed at planetary scale
- Upgraded Kubernetes version w/ no downtime
  - Like changing plane engine mid-flight
- Upgrade Load Balancer also w/ no downtime
- Largest Kubernetes deployment ever
- GKE running tens of thousands of cores
- Google private network reduced overall latency



Google Cloud Platform Cloud Architect Essentials

What is Big Data?

Lesson plan

- What is big data?
- Where does big data come from?
- Why is big data important?
- Why is the cloud the best platform for big data?
- How does GCP do big data well?

What is big data?

- Extremely large data sets that may be analyzed computationally to reveal patterns, trends, and associations, especially relating to human behavior and interactions
- Too expensive to store, manage, and analyze using traditional database systems.
  - Not cost-inefficient
  - Difficulty storing unstructured and “high-velocity” (real-time) data
  - Hard to support very large (petabyte-scale) data volumes
- New approaches and systems needed
  - Example: Apache Hadoop and NoSQL databases

Where does big data come from?

- In the past, most data came from structured formats
- Today, large amount of data is unstructured and massive size
- ‘Internet of Things’ (IoT) is major source
- Regulated industry compliance

Why is big data important?

- Business value
  - Trait of successful organizations across every industry
  - For some industries, it’s literally a matter of survival
- More data = more value
- Close ties with machine learning
  - More data can ‘train’ big data models to get even more value

Why is the cloud the best platform for big data?

- Difficult to manage on premise
  - High space
  - High costs
  - Very complex administration
- Managed big data services reduce cost and complexities of hosting your own system
  - Experiment with different products/systems without the up front cost

How does GCP do big data well?

- Big data analytics at Google scale
  - Able to store and process the sheer volume of data
  - Big data is in their DNA
- Serverless, managed infrastructure
  - All backend infrastructure handled for you, including autoscaling
- Fast action on petabyte sized datasets
- Both batch and stream processing
- Spark and Hadoop in the cloud
- Industry leading Machine Learning capabilities

Google Cloud Platform Cloud Architect Essentials

GCP Big Data Services

GCP Big Data Suite

- Google BigQuery
- Google Cloud Dataflow
- Google Cloud Dataproc
- Google Cloud Datalab
- Google Cloud Dataprep
- Google Cloud Pub/Sub
- Observe flow of data between services

Google BigQuery

- Enterprise data warehouse
  - Fully managed – no resources to allocate or deploy
- Stores and queries massive datasets
- SQL syntax for queries
- Processing power of Google's infrastructure
- Real-time analysis - use data as soon as the data enters the system

Google Cloud Dataflow

- Fully managed data processing service
  - No resource provisioning – on demand and nearly limitless capacity
  - Data pipeline to other services
- Batch and stream processing
  - Batch = process large volume of data at once (e.g. all data from past week)
  - Stream = continuous input and output of data
    - Fast turn-around of data
- Open source – no vendor lock in
- Tight integration with rest of GCP

Google Cloud Dataproc

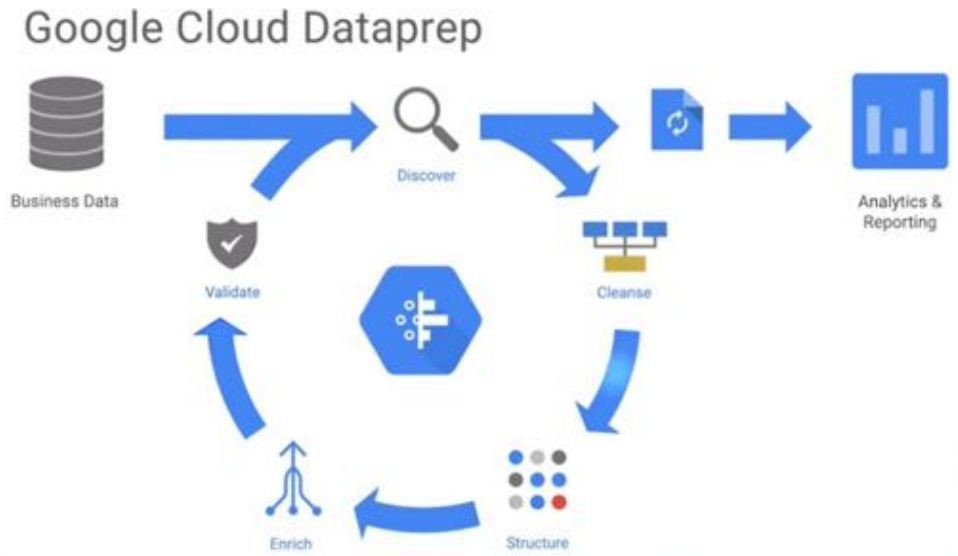
- Fully-managed cloud service for running Apache Spark and Apache Hadoop clusters
- Scalable clusters - quickly create and resize clusters as needed
- Allocated resources are billed by the minute
  - Pre-emptible instances for batch processing recommended
- Open source and integrated
- Ideal for:
  - Migrate Hadoop jobs to the cloud
  - Analyze data stored in Cloud Storage
  - Use Spark to perform data mining and analysis
  - Use Spark Machine Learning Libraries to run classification algorithms

Google Cloud Datalab

- Interactive tool for data exploration, analysis, visualization and machine learning
- Open source – built on Jupyter (formerly iPython)
- Integrates with other GCP services
- Supports machine learning models based on TensorFlow

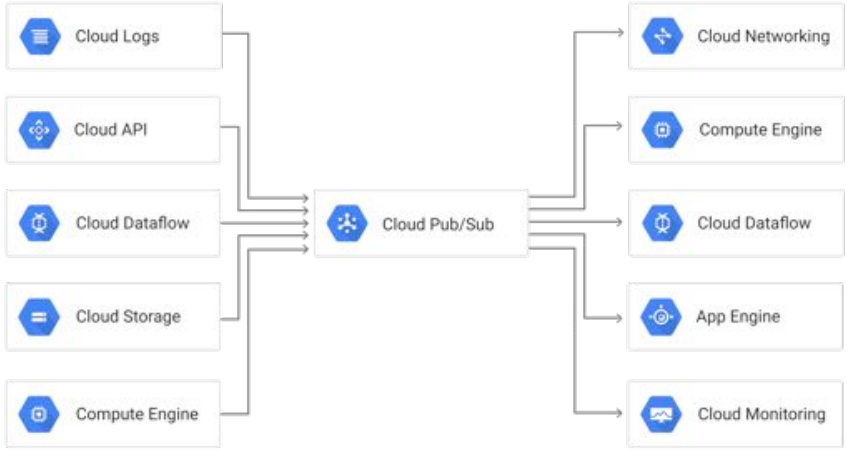
Google Cloud Dataprep

- Visually explore, clean, and prepare data for analysis
- Fully managed, serverless, no operational overhead
  - Prepare data with a few clicks, no code needed
- Runs on top of Dataflow

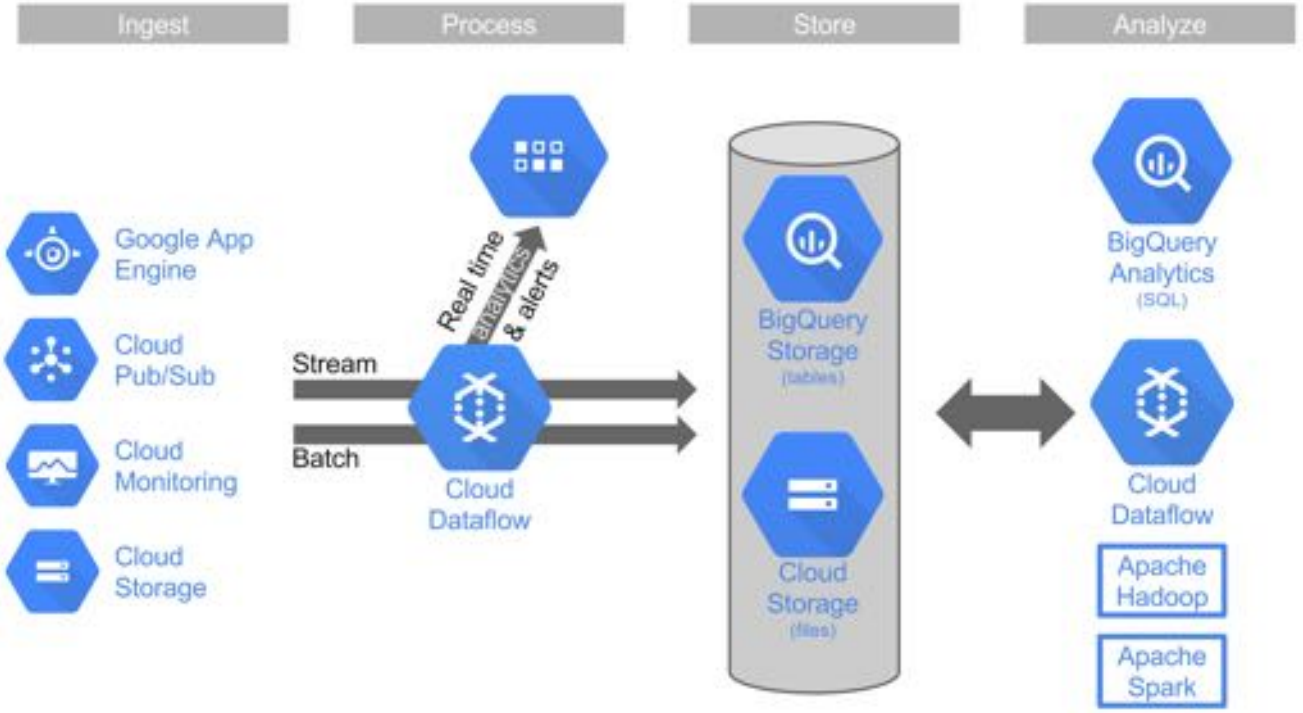


Google Cloud Pub/Sub

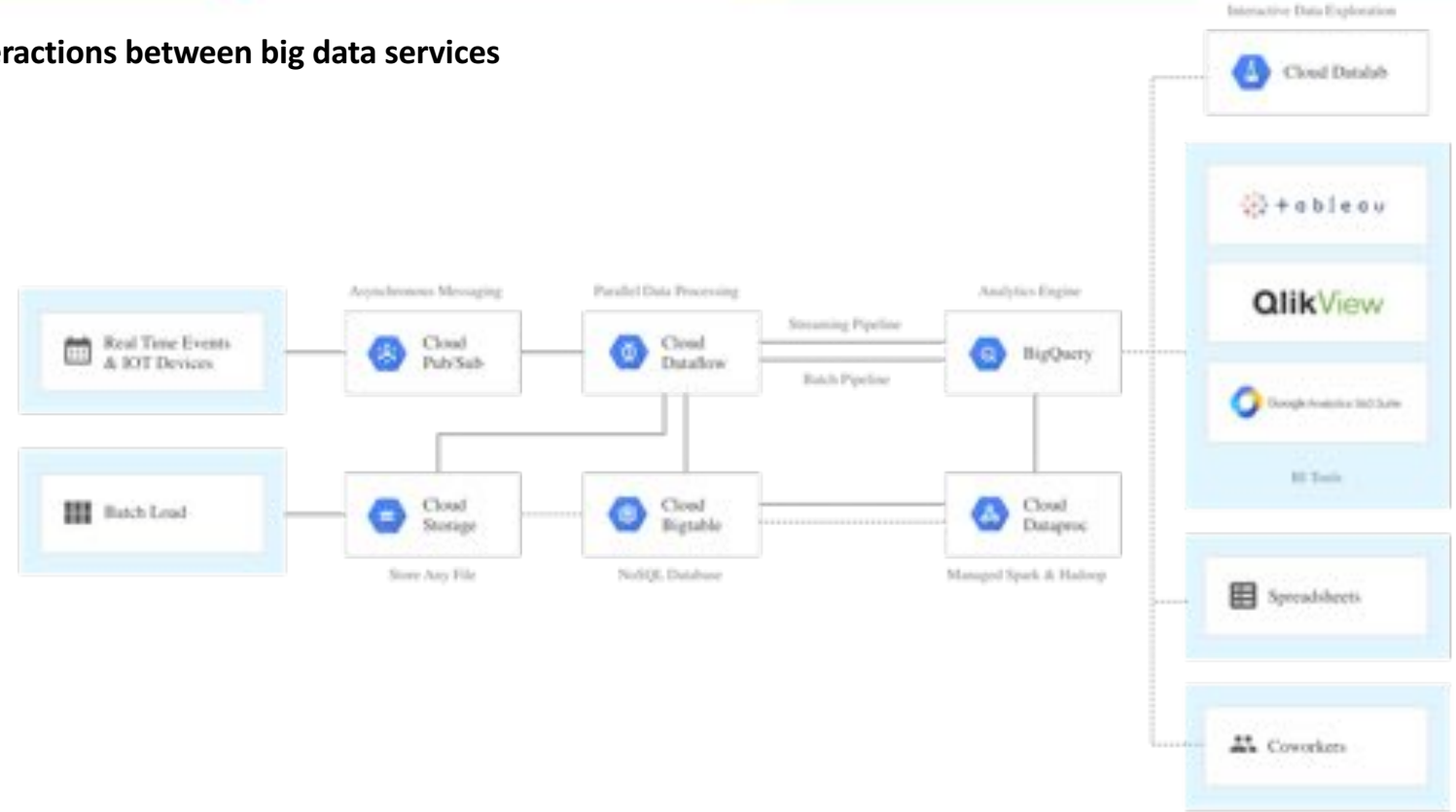
- Messaging middleware
- Apps publish and subscribe to ’topics’
- Send and receive messages between messages in real time
  - Ideal for stream processing
- Connecting service between other GCP services
- Decouples senders and receivers
  - Dependable communication between independently written apps
  - Many to many messaging (asynchronous)
- Example: Data streaming from various devices to cloud services



Big data lifecycle



Interactions between big data services



Google Cloud Platform Cloud Architect Essentials

Google Cloud Machine Learning Platform

In this lesson:

- What is Machine Learning?
- Why is it important?
- Google Cloud Platform and Machine Learning
  - Pre-built and custom models
- Fun hands on demos

What is Machine Learning?

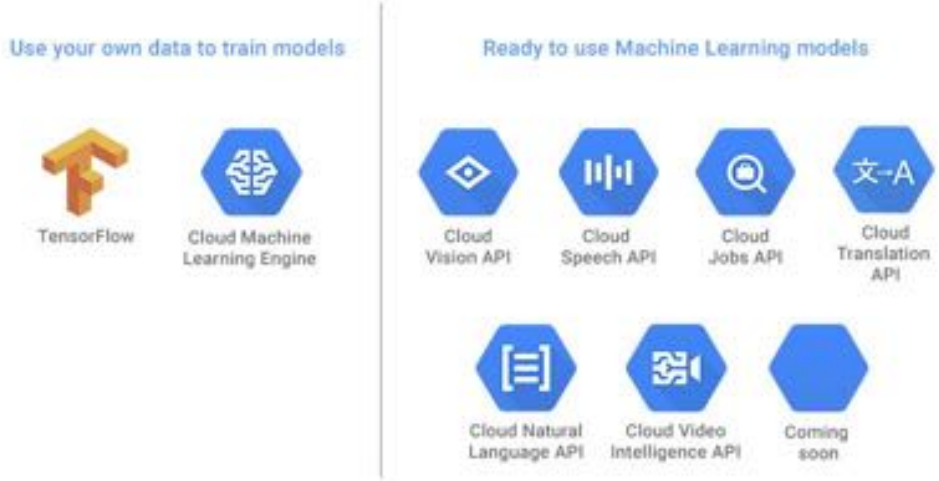
- Machines and applications can learn and adapt new
  - Without having to be explicitly programmed to do so.
  - When exposed to new data, a ML app can learn from it without external input
- Simple definition: creating applications that can **see, hear** and **understand** the world around them.
  - Intelligent applications
- Big Data and Machine Learning go hand in hand

Why is Machine Learning important?

- Next frontier in technology innovation
  - All of the major technology companies are making it a focus
- Practical applications include:
  - Predictive protection against cyber-crime, using big data models
  - Automatic identification of images (Google Photos)
  - Real time language translation
    - Text, audio, and even images
  - Voice to text dictation and vice versa
  - Smartphone personal assistants - Siri, Google Assistant, Alexa
    - Automated notifications of when to leave for an appointment
  - Automates tasks in analyzing big data
  - In the future - self driving cars

Google Cloud and Machine Learning

- Google is a clear leader in machine learning
- Most popular consumer applications use machine learning to make them unique
  - examples: Google Photos, Google Assistant, Google Translate
  - Same capabilities are available for anyone on GCP



GCP Machine Learning Organization

- Built on TensorFlow - Google's open source tool to build and run neural network models
  - Design, build, and train deep learning models
  - Previously used internally at Google for their own Deep Learning data flows
  - Works on a large variety of platforms
  - Same framework Google uses for their own ML products

Machine Learning Engine

- Create your own machine learning service
- Build machine learning models that work on any type of data, of any size.
- Managed service - no resource allocation necessary
- Ideal for custom predictive analytics
- Create own model with TensorFlow framework
  - Open source
- Tightly integrates with Cloud Storage, BigQuery, and other Big Data resources
- Supports thousands of users and TB’s of data

ML Engine Use Cases:

- Data security – detect minute variations in file signatures to identify malware
- Financial trading – ML algorithms conduct high speed trades using large datasets
- Health care – understand risk factors and spot patterns in medical info
- Marketing – personalized ads based on your interests
- Fraud detection – PayPal using ML to fight money laundering, find anomalies in transactions
- Smart cars – both self-driving cars and smaller goals

Pre-trained models offered by GCP – Cloud Vision API

- Image recognition without manually tagging each one
- Classify image into thousands of categories (boat, lion, car, etc)
- Detect inappropriate content
- Find topical web items such as celebrities, logos, and events
- Detect and extract text for OCR - automatic language detection

Cloud Natural Language API

- Reveal the structure and meaning of text
- Extract information about people, places, events
  - Mentioned in text documents, news articles or blog posts.
- Understand written sentiment about your product on social media or parse intent from customer conversations happening in a call center or a messaging app.

Cloud Translate API

- Language translation and detection
- Convert language from one source to target translation
- Detect unknown languages

Cloud Speech API

- Speech recognition
- Convert audio to text and vice versa
- Accuracy improves over time
- Over 110 languages supported
- Context aware recognition – learn context based on when/where request made

Cloud Video Intelligence (Beta)

- Video analysis
- Makes videos searchable by content
- Automatically label objects and people in videos
- Detect scene transitions and search within a scene

Hands on demo