O'REILLY®

GCP Professional Cloud Architect Certification Prep

Prerequisites

- Familiarity with cloud platforms (AWS, Azure)
- Basic familiarity with the GCP

- This training focuses on breadth - not depth
- Concepts, fundamentals and applications

Introductions

I have experience with the Google Cloud Platform:

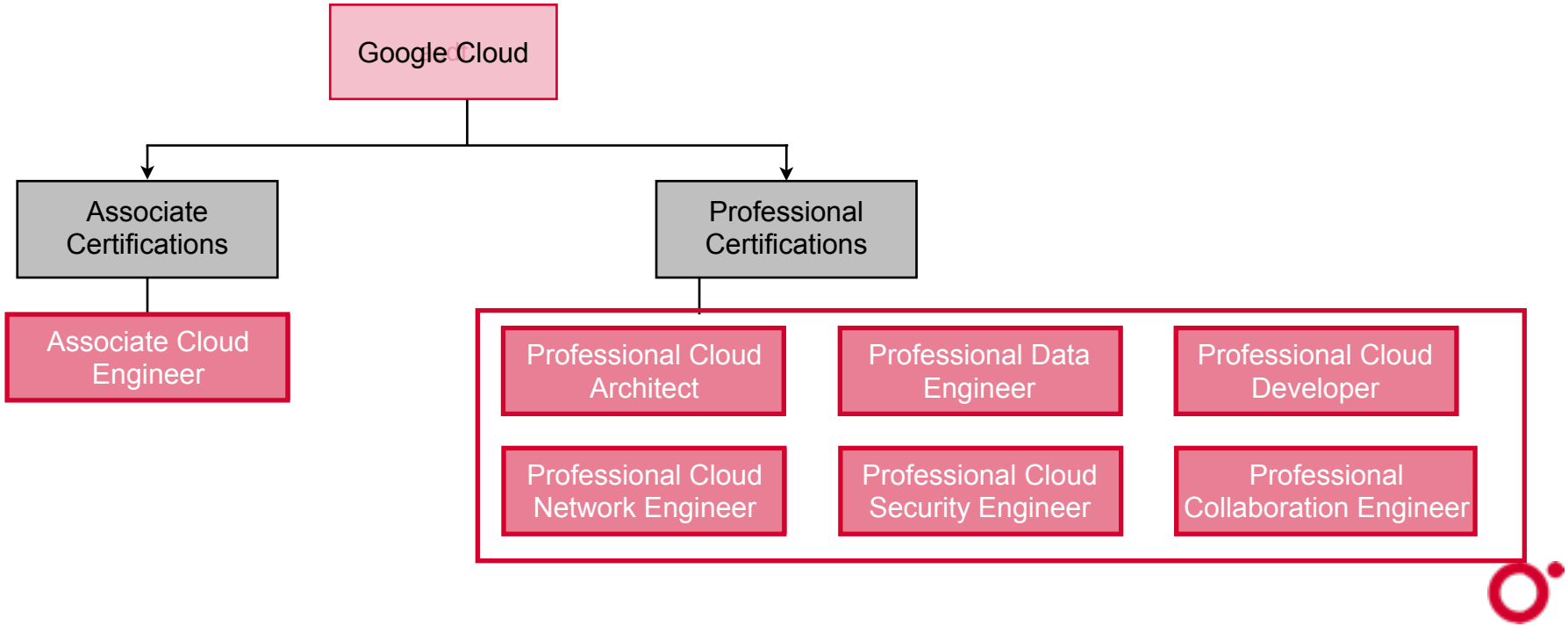
1. No experience at all
2. 0-1 years of experience
3. 2-3 years of experience
4. 3+ years of experience

Introductions

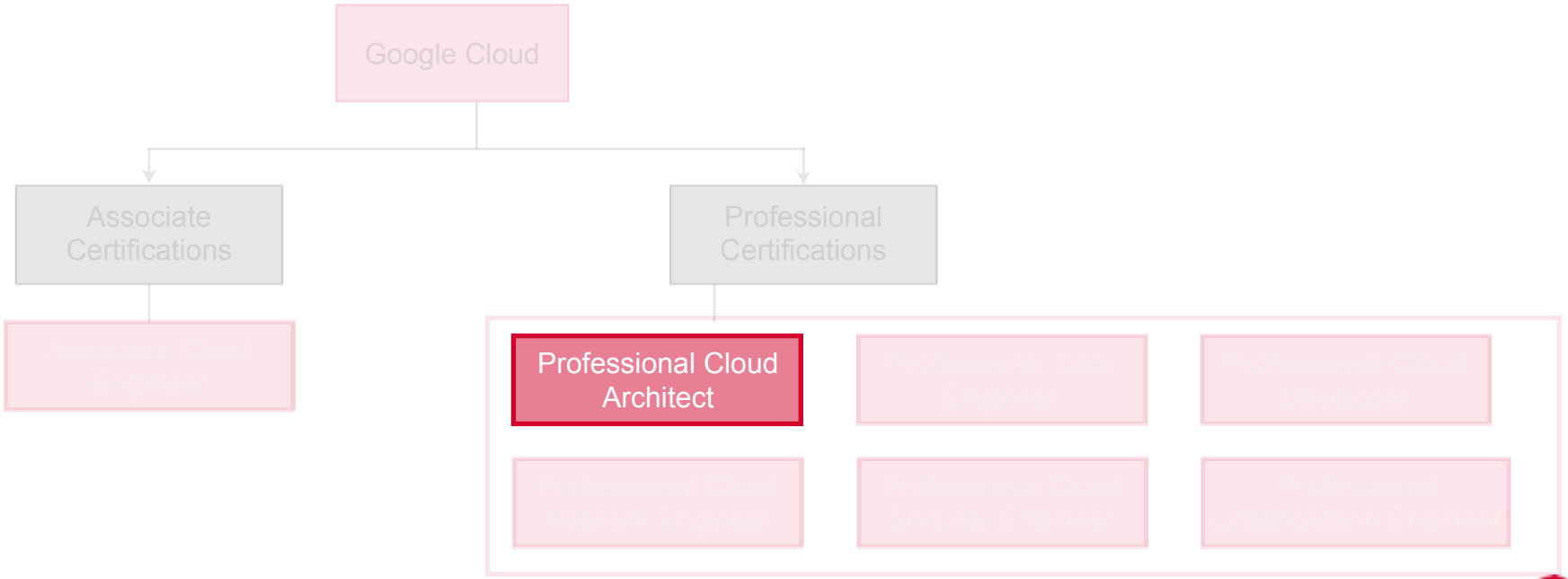
I have worked on other cloud platforms:

1. Mostly AWS
2. Mostly Azure
3. Other cloud platforms

Google Cloud Certifications



Google Cloud Certifications



Professional Cloud Architect

- Test duration: 2 hours
- Registration fee: $200 + taxes
- Languages: English, Japanese, Spanish, Portuguese
- Recommended: 3+ years GCP experience

Professional Cloud Architect

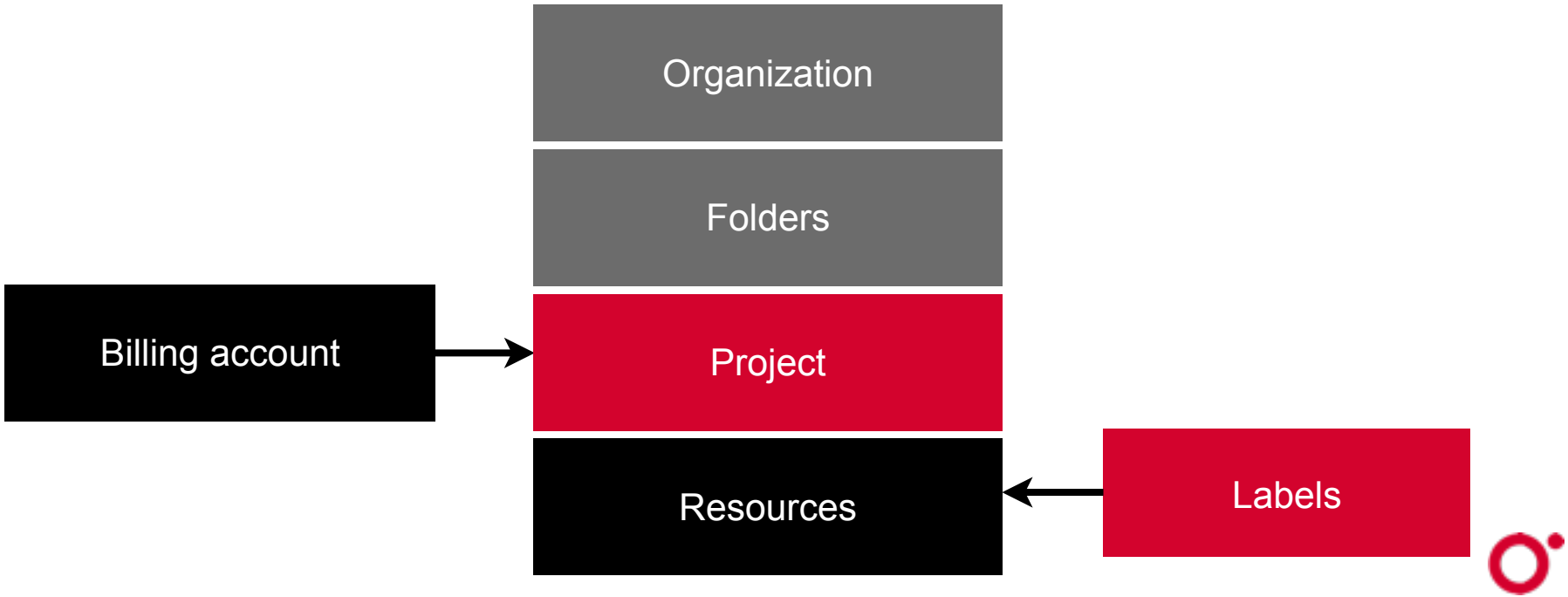
- Vast array of services for a wide variety of use cases
- A good understanding of the specialized strengths of each service
- Extensive labs for hands-on practice:
  - https://codelabs.developers.google.com/?cat=Cloud
- Case studies link here:
  - https://cloud.google.com/certification/guides/professional-cloud-architect/

O'REILLY®

Google Cloud Platform Basics

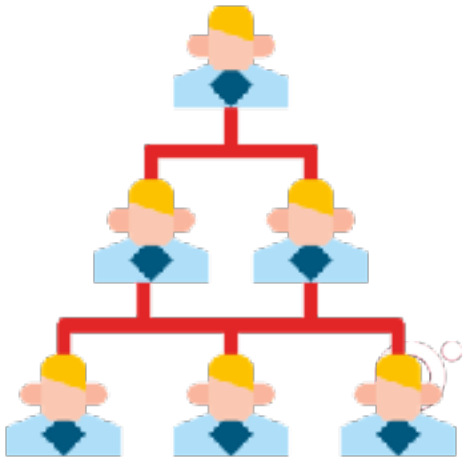
Month/Year

Resource Hierarchy Components



Organization

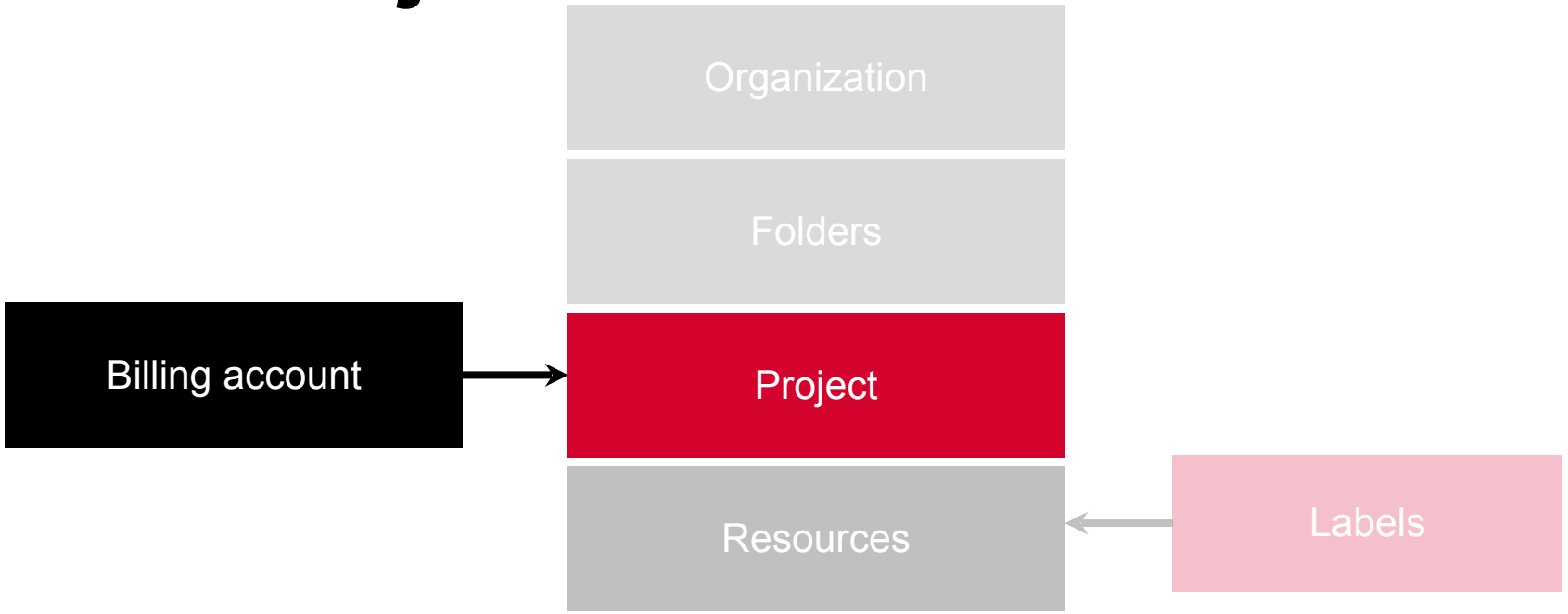
- Top of resource hierarchy - central control
- Contains projects and folders
- Identities come from G Suite or a Cloud Identity account
- IAM policies are inherited down into projects and resources
- Projects belong to the organization, not employees
- Can grant organization level roles



Folders

- Grouping mechanism within an organization
- Logical group of projects
- Can set IAM policies to administer multiple projects
- Model legal entities, departments, and teams

Billing Accounts Are Associated with Projects



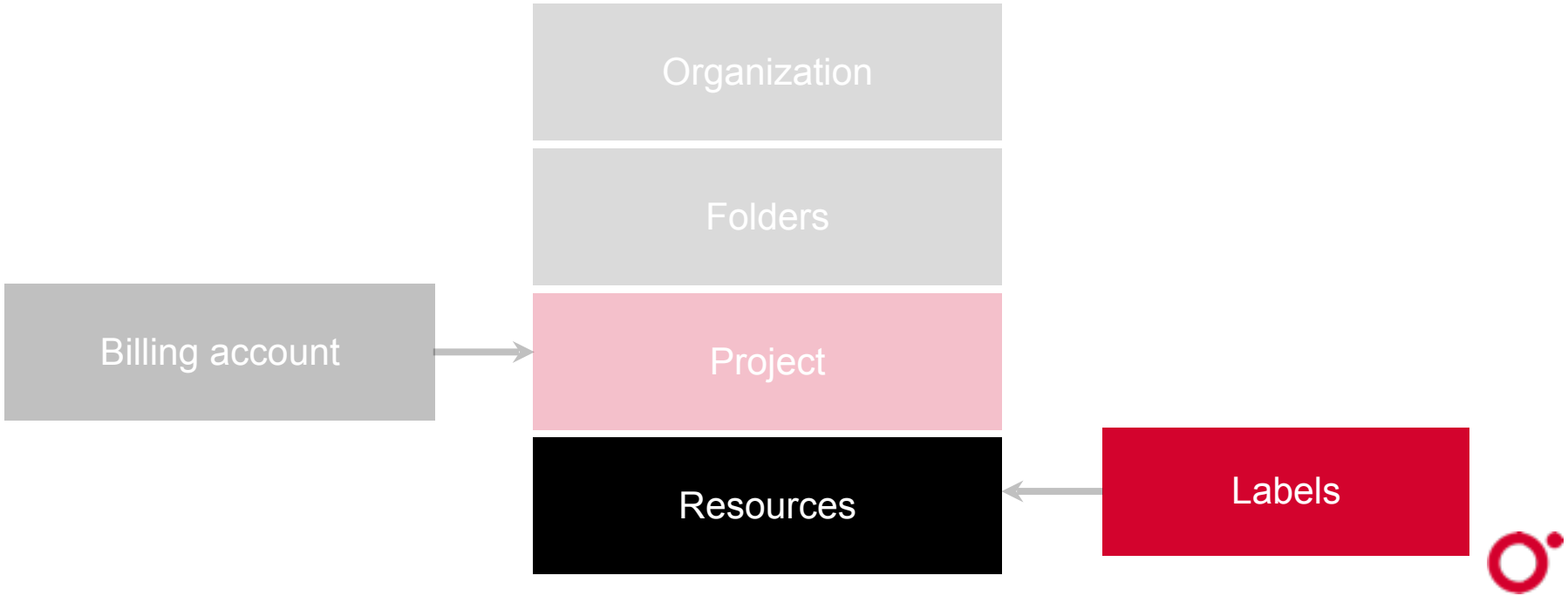
Projects

- Container for billable resources
- Some resources can be used for free
- For all others, billing account needs to be linked
- Required resource for using GCP services

Resources

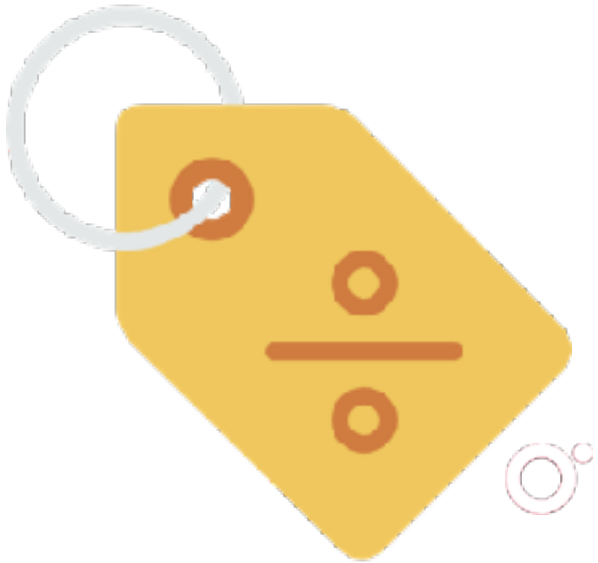
- Any component that incurs billing
- Must exist within project
- Can set resource-level IAM
- Inherits policies from organization, folder, project
- Lowest level of the hierarchy

Labels Are Applied to Resources



Labels

- Key-value pairs
- Resource metadata
- Can use to organize billing
- Can break down billing by label



Using GCP Resources

Cloud Console and Cloud Shell

Under the hood, making API calls. Cloud Shell is a great terminal utility.

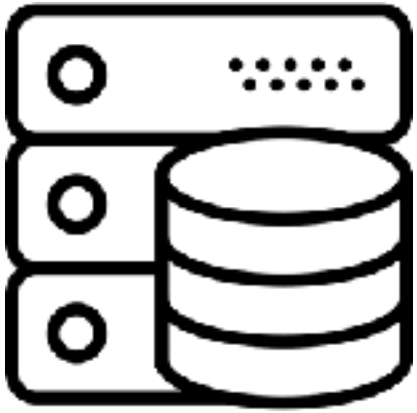
gsutil and gcloud

(bq and kubectl) Command-line tools

Client APIs

Programmatic access via HTTP calls to GCP endpoints

Choices in Computing



Compute

Where is code executed and how?

Storage

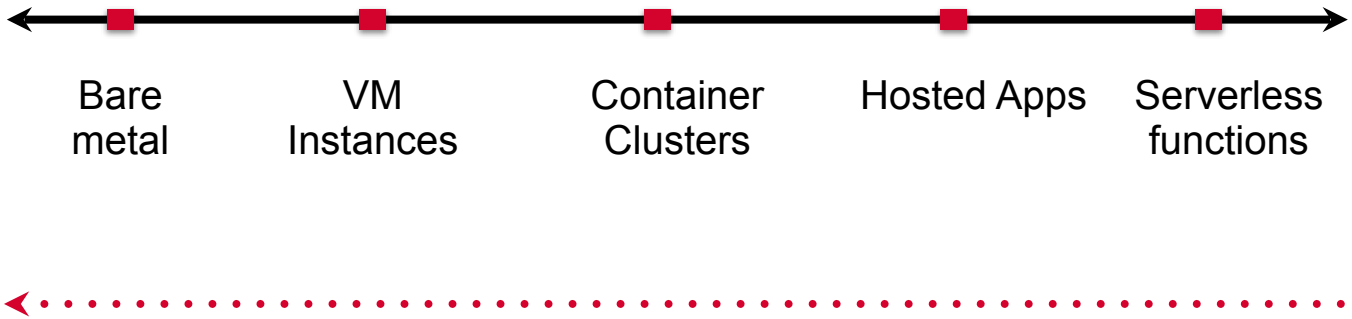
Where is data stored?

Networking, hosting, logging, are choices made after this fundamental decision

Compute Choices

Compute Choices

Compute Choices



GCP Compute Choices

Projects

Which of the following best describes a project on the GCP?

1. Logical grouping of resources based on labels
2. Root node in the resource hierarchy
3. Used to group GCP networks
4. Logical grouping for resources, associated with billing

Projects

Which of the following best describes a project on the GCP?

1. Logical grouping of resources based on labels
2. Root node in the resource hierarchy
3. Used to group GCP networks
4. **Logical grouping for resources, associated with billing**

Cloud Shell

Which of the following best describes Cloud Shell?

1. Command-line utility used to work with the GCP services
2. Ephemeral VM which offers a terminal on the browser
3. PaaS offering on the GCP for hosted applications
4. IaaS offering on the GCP

Cloud Shell

Which of the following best describes Cloud Shell?

1. Command-line utility used to work with the GCP services
2. **Ephemeral VM which offers a terminal on the browser**
3. PaaS offering on the GCP for hosted applications
4. IaaS offering on the GCP

O'REILLY®

Google Compute Engine (GCE)

Bare Metal vs. IaaS

Bare Metal

- Apps run on OS which runs on hardware
- Less portable
- CPUs
- Full burden of ops and admin

Bare Metal vs. IaaS

Bare Metal

- Apps run on OS which runs on hardware
- Less portable
- CPUs
- Full burden of ops and admin

IaaS

- Hypervisor between apps and hardware
- More portable
- vCPUs
- Much of ops burden managed by service provider

GCP Internals

Zone

Availability zone (similar to a datacenter)

Region

Set of zones with high-speed network links

Network

User-controlled IP addresses, subnets and firewalls

GCP Internals

Zone

“asia-south1-a”

Region

“asia-south1”

Network

“default”

Global, Regional and Zonal Compute Resources

- Global:
  - Static external IP addresses
  - Images and snapshots
  - Networks, firewalls, routes
- Regional
  - Subnets
  - Regional persistent disks
- Zonal
  - Instances
  - Persistent disks

Configuration Choices

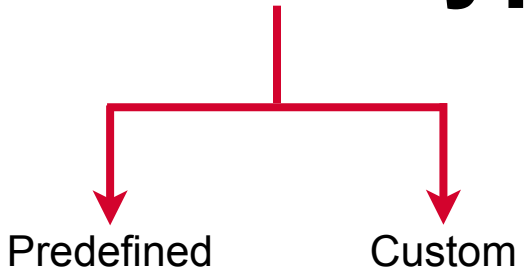
Machine Type

Memory size, virtual CPU (vCPU) count, and maximum persistent disk capability

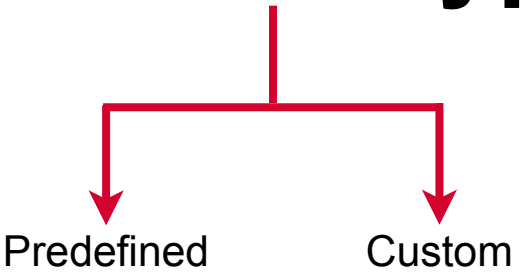
Base Image

Public (free or premium), custom, snapshots from boot disks

Machine Type

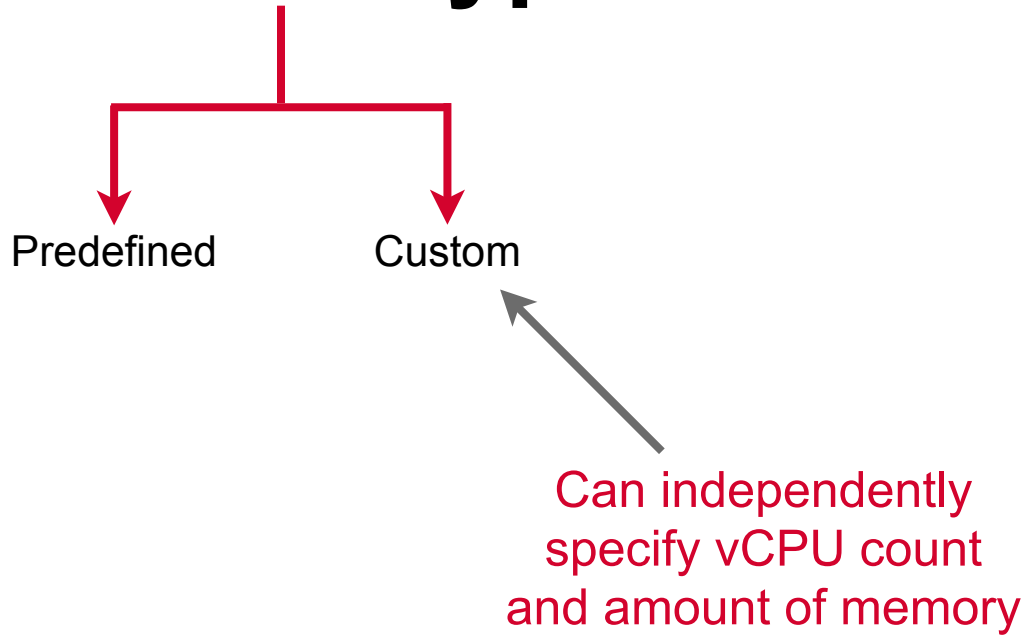


Machine Type

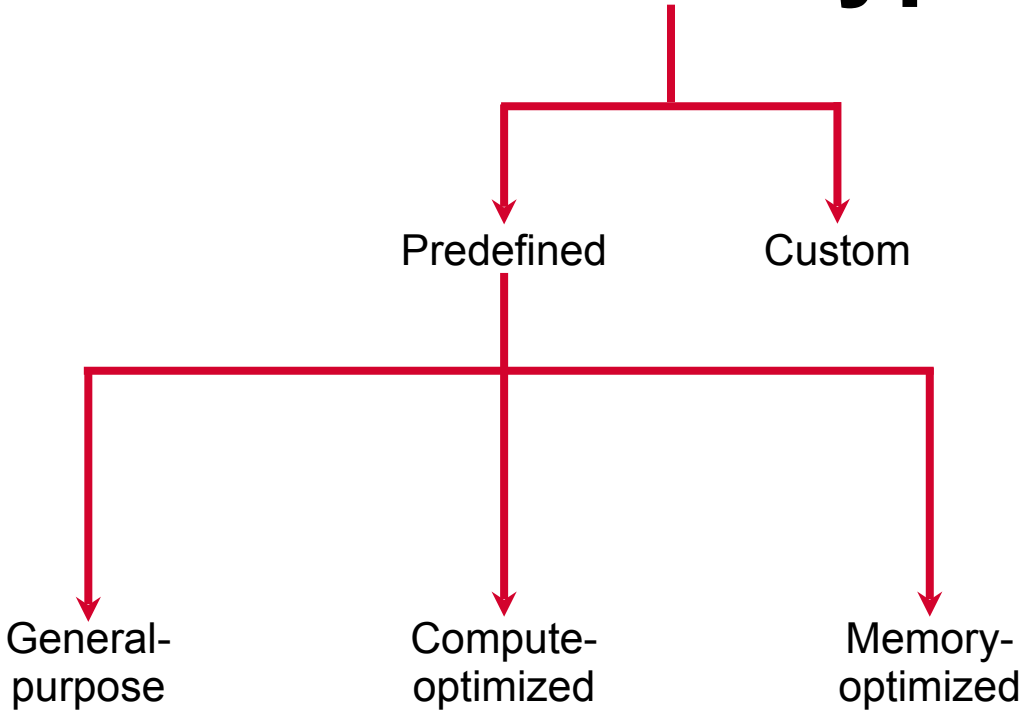


Fixed set of types with fixed ratios of memory to vCPU count

Machine Type



Machine Type



General Purpose Machines

- Day to day computing for known workloads
- Best price-performance ratio
- N1 first generation: 6.5GB of memory per vCPU
- N2 second generation: 8GB of memory per vCPU
  - More heavy duty workloads such as web serving, databases, applications use N2
- Can customize machine types
- Come in high-memory and high-cpu variants

Compute-optimized Machines

- Compute intensive workloads
- Offer the highest performance per core
- C2 machine types
- Gaming, single-threaded applications, electronic design automation
- Custom machine types not supported

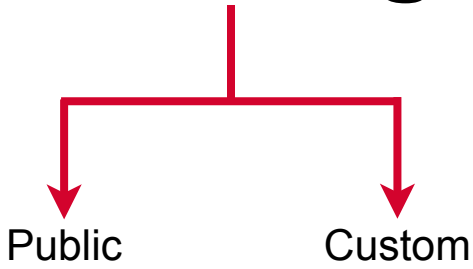
Memory-optimized Machines

- Memory-intensive workloads
- Offer the highest memory per core
- Custom machine types not supported

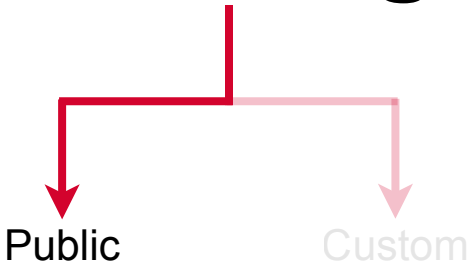
Shared-core Machines

- Cost-effective for running non-resource intensive operations
- A single vCPU run for a time period on single hardware
- Offer micro-bursting capabilities for spikes
- Instance will use additional physical CPUs during spikes

Base Images



Base Images



Provided and maintained by Google, open-source communities, and third-party vendors

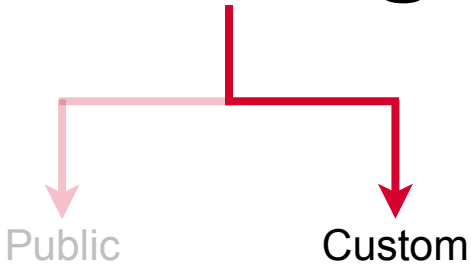
All projects have access to these images and can use them to create instances

Base Images

Linux, Windows, Container-optimized OS, SQL Server

Many images come with Shielded VM support

Base Images



Available only to your project

First, create a custom image from boot disks and other images; then, use the custom image to create an instance

Shielded VM

- Verifiable integrity of your compute instances
- Haven’t been compromised by **boot or kernel-level malware**
- **Secure Boot:** Verifies digital signature of software during boot
- **Virtual Trusted Platform Module vTPM**: Specialized computer chip to protect keys and certificates
- **Measured Boot:** Hashes boot components to verify load order and components loaded
- Integrity monitoring of VM instances

Preemptible VM Instances

An instance that you can create and run at a much lower price than normal instances. However, GCE might terminate (preempt) these instances if it requires access to those resources for other tasks.

May not always be available

Not covered by SLAs

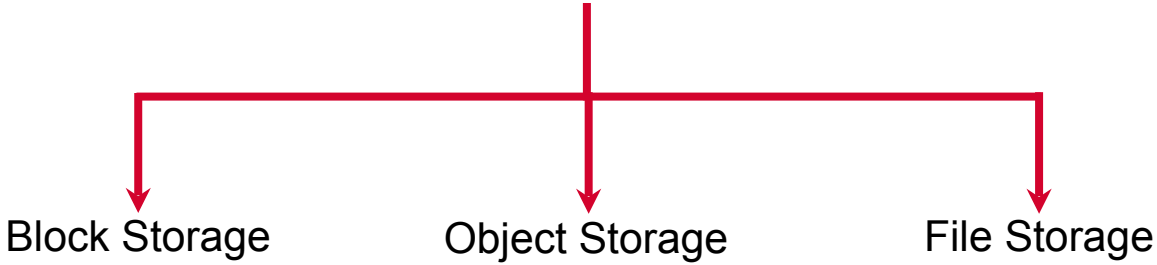
Sole-tenant Nodes

A sole-tenant node is a physical Compute Engine server that is **dedicated to hosting VM instances** only for your specific **project**

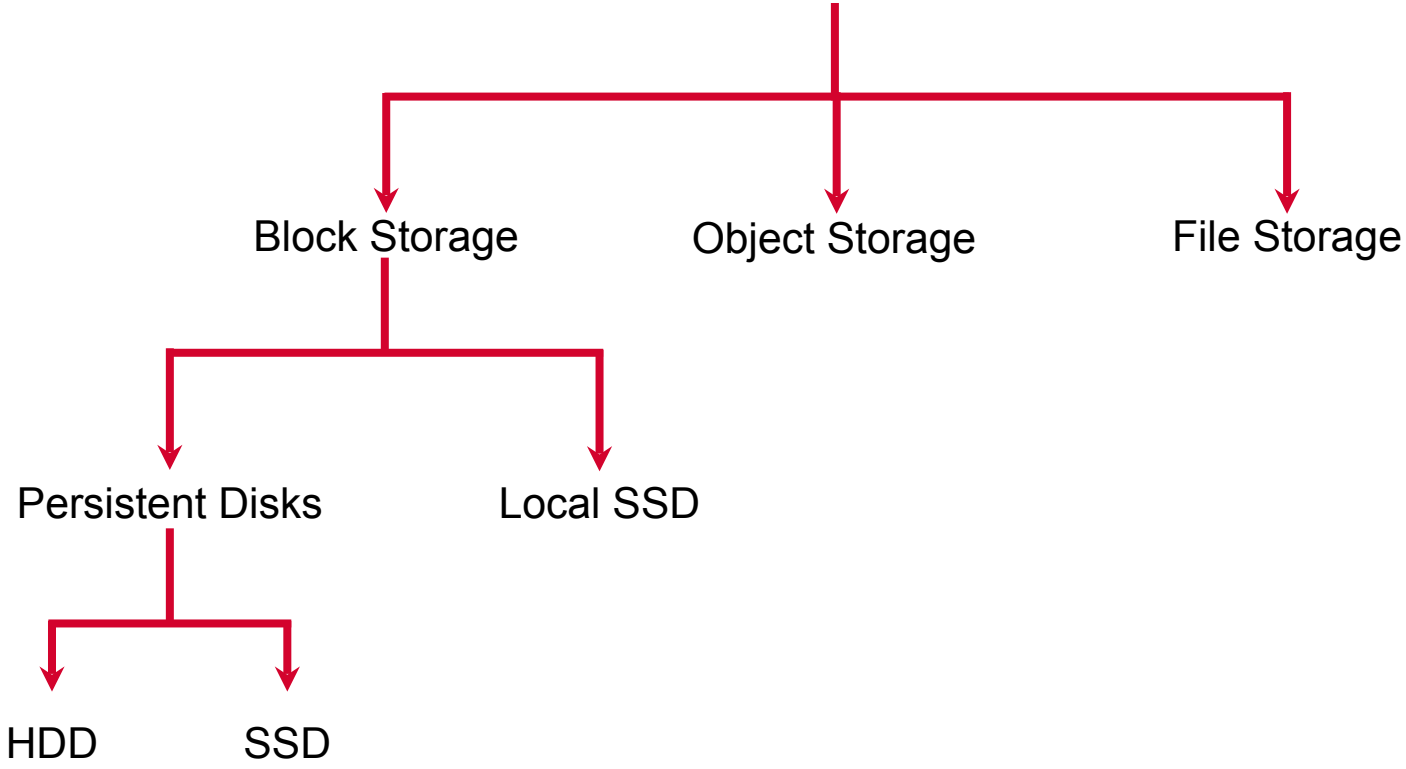
Keeps your instances physically separated from instances in other projects

Group instances together on the same hardware

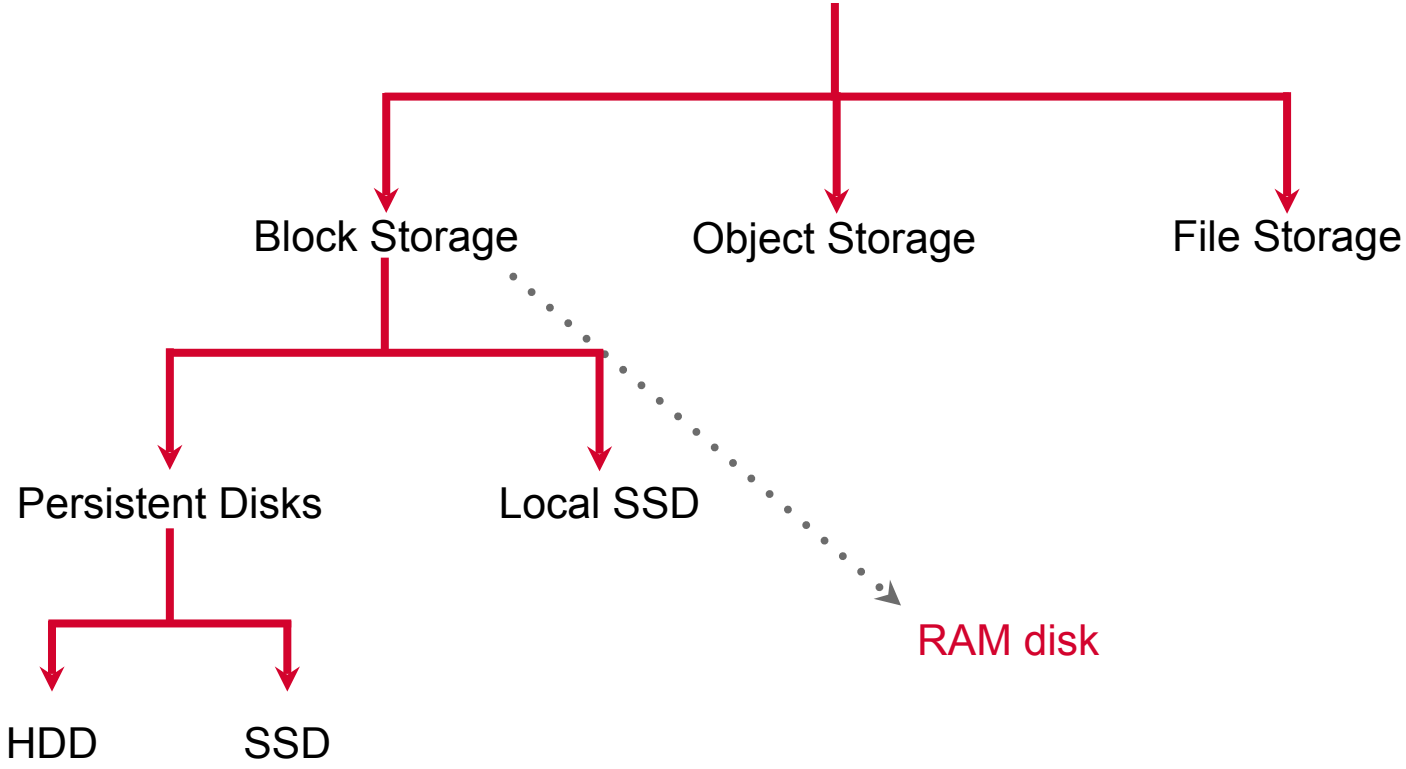
Accessing Storage from VMs



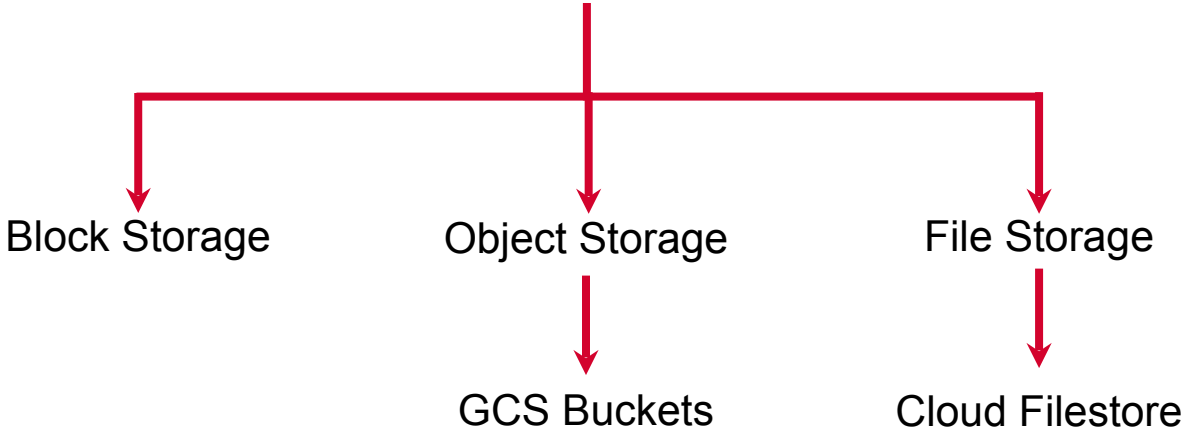
Accessing Storage from VMs



Accessing Storage from VMs



Accessing Storage from VMs



Persistent Disks vs. Buckets

Persistent Disks

- Block storage
- Max 64TB in size
- Pay what you allocate
- Tied to GCE VMs
- Zonal (or regional) access

Buckets

- Object storage
- Infinitely scalable
- Pay what you use
- Independent of GCE VMs
- Global access

Persistent Disks

- Resize on the fly
- Move across zones
- Create images and snapshots
- Encrypted at rest
  - can use custom keys

Boot Disk

- Each GCE VM needs a persistent boot disk
- This disk contains boot loader, OS etc.
- Bootable
- Durable
  - can delete VM but keep disk

Persistent Disks vs. Local SSDs

Persistent Disks

- Network-attached storage
- Data redundancy built-in
- Bootable
- Durable
- HDD or SSD
- 64TB max
- Create snapshots or images
- Relatively slow

Local SSDs

- Physically attached to instance
- No data redundancy built-in
- Not bootable
- Not durable
- SSD for better performance
- 3TB max
- Can not create snapshots or images
- Very fast, especially for random access

Labels

Key-value pairs that can be associated with any GCP resource; a lightweight way to group related resources

Network Tags

Text attributes applied to VM instances (and instance templates) as a way of applying **firewall rules and routes** to specific instances

Metadata Server

- Labels and tags are forms of metadata
- Reside outside an instance on a metadata server
- Can be programmatically queried
  - Instance itself can query without authorization

Metadata Server

- Use with startup and shutdown scripts
- Commonly used to find
  - instance host name
  - instance ID
  - startup and shutdown scripts
  - service account

Region

Which of the following best describes a region on the GCP?

1. A logical area that may be spread across countries
2. A single datacenter on the GCP
3. A geographical area with multiple datacenters
4. Physically connected hardware devices in a datacenter

Region

Which of the following best describes a region on the GCP?

1. A logical area that may be spread across countries
2. A single datacenter on the GCP
3. **A geographical area with multiple datacenters**
4. Physically connected hardware devices in a datacenter

Persistent Disks

What is the pricing mechanism for Persistent Disks?

1. If the you create a 100GB disk but you use just 5GB you pay for the entire 100GB
2. If you create a 100GB disk but you use just 5GB you pay for only 5GB
3. If you create a 100GB disk but you use just 5GB you pay for only 5GB + a little extra

Persistent Disks

What is the pricing mechanism for Persistent Disks?

1. **If the you create a 100GB disk but you use just 5GB you pay for the entire 100GB**
2. If you create a 100GB disk but you use just 5GB you pay for only 5GB
3. If you create a 100GB disk but you use just 5GB you pay for only 5GB + a little extra

Local SSD

Which of the following correctly describes a local SSD?

1. Used a as a boot disk and can be snapshotted
2. Offers lower performance as compared with Cloud Storage Buckets
3. Elastic storage which grows as you store more data in it
4. Physically attached to your VM so offers high throughput and low latency

Local SSD

Which of the following correctly describes a local SSD?

1. Used a as a boot disk and can be snapshotted
2. Offers lower performance as compared with Cloud Storage Buckets
3. Elastic storage which grows as you store more data in it
4. **Physically attached to your VM so offers high throughput and low latency**

O'REILLY®

Snapshots and Images

Image

- Binary file used to instantiate VM root disk
- Usually based off OS image
- Also contains boot loader
- Can also contain customizations
- Managed by GCP image service

Snapshot

- Binary file with exact contents of persistent disk
- “Point-in-time” snapshot
- Managed by GCP snapshot service
- Incremental backups possible too
- Used to back up data from persistent disks

Snapshots and Images

- Conceptually very similar but many differences in nitty-gritty

Images and Snapshots

Persistent Disk Images

- Create an image to use disk as basis for new instances
- Not incremental
- Relatively expensive
- Can be directly used to instantiate new instance or managed instance group
- Supports families and versioning
- Share across projects

Persistent Disk Snapshots

- Create a snapshot to backup data present in a disk
- Incremental
- Relatively cheap
- Must first be used to create a disk before instances can be created from it
- No support for families
- Specific to project

O'REILLY®

Google App Engine

GCP Compute Choices

Google Compute Engine

Google Kubernetes Engine

Google App Engine

Google Cloud Functions

Google App Engine

Web framework and platform for hosting web applications on the Google Cloud Platform

Support for Go, PHP, Java, Python, Node.js, .NET, Ruby and other languages

Google App Engine

Web framework and platform for hosting web applications on the Google Cloud Platform

Support for Go, PHP, Java, Python, Node.js, .NET, Ruby and other languages

Focus on development and code

Infrastructure and scaling taken care of by the platform

App Engine Environments

Standard Environment

Flexible Environment

App Engine Environments

Standard

- App runs in a proprietary sandbox
- Instances start up in seconds
- Code in few languages/versions only
- No other runtimes possible
- Apps cannot access Compute Engine resources
- No installation of third-party binaries

App Engine Environments

Standard

- App runs in a proprietary sandbox
- Instances start up in seconds
- Code in few languages/versions only
- No other runtimes possible
- Apps cannot access Compute Engine resources
- No installation of third-party binaries

Flexible

- Runs in Docker container on GCE VM
- Instance start up in minutes
- Code in far more languages/ versions
- Custom runtimes possible
- Apps can access Compute Engine resources, some OS packages
- Can install and access third-party binaries

App Engine Environments

Standard

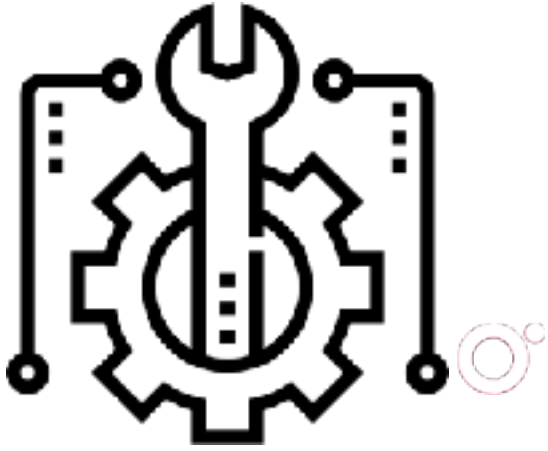
- Apps that experience traffic spikes
- Usually stateless HTTP web apps

Flexible

- Apps that experience consistent traffic
- General purpose apps

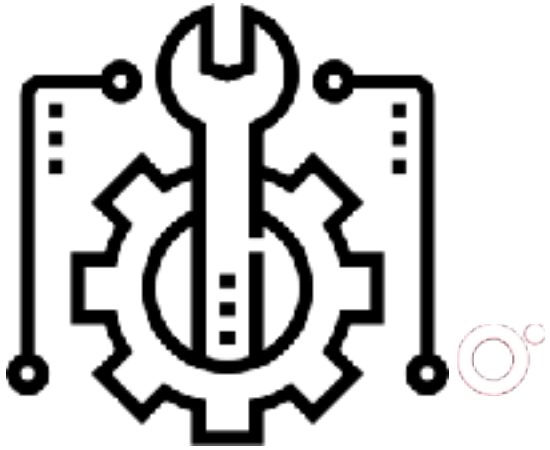
App Engine Standard Runtimes

- Python 2.7, Python 3.7
- Java 8, Java 7
- Node.js 8 (beta), 10 (beta)
- PHP 5.5, 7.2 (beta)
- Go 1.9, 1.11 (beta)
- More on the way



App Engine Flexible Runtimes

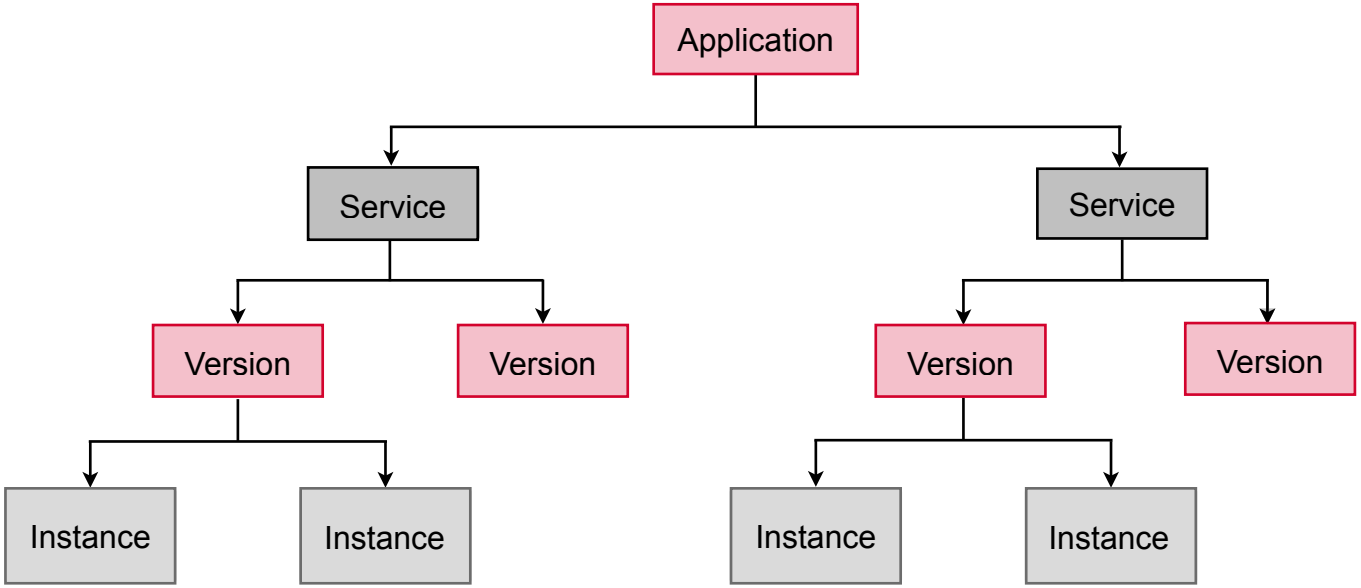
- Python 2.7, Python 3.6
- Java 8
- Node.js
- Go 1.9, 1.10, 1.11
- Ruby
- PHP 5.6, 7.0, 7.1, 7.2
- .NET
- Custom runtimes



App Engine App

Single regional application resource consisting of hierarchy of services, versions and instances

Components of an Application



O'REILLY®

Google Cloud Functions

Month/Year

GCP Compute Choices

Cloud Functions

Event-driven serverless compute platform

Event-driven Serverless Compute

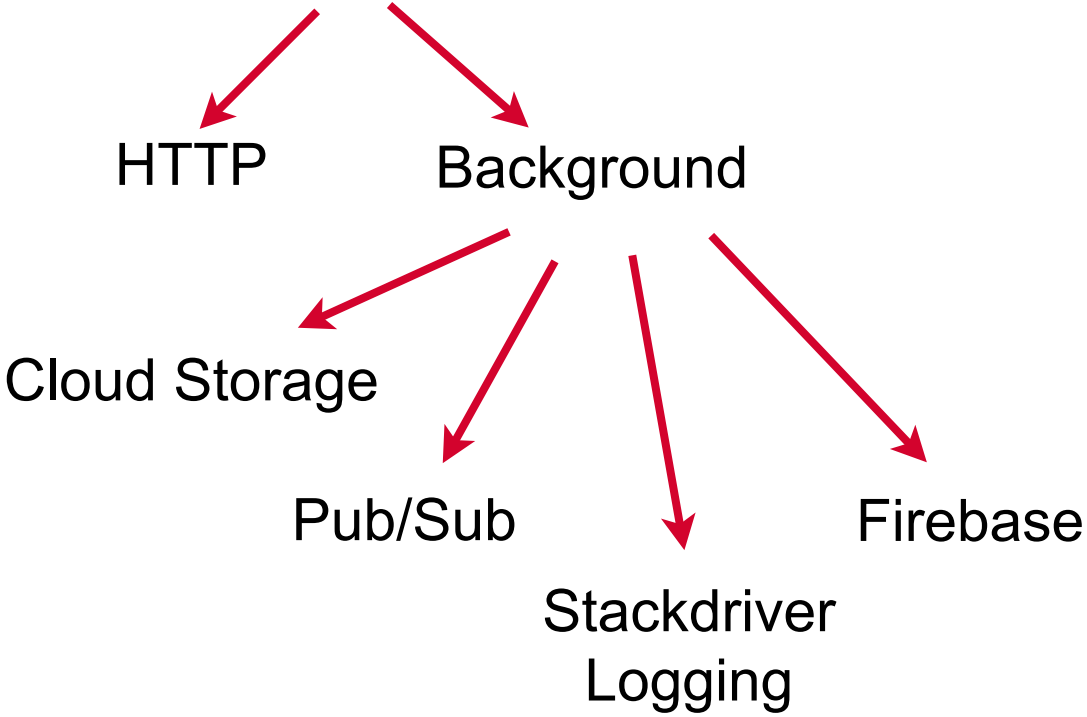
Event occurs

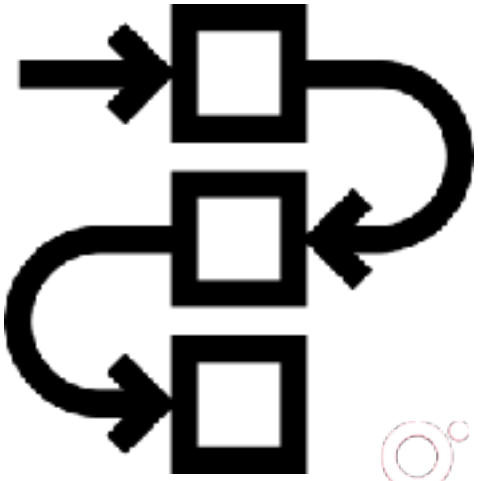
Platform triggers execution

Cloud Function code runs

Invokes other GCP services

Types of Events



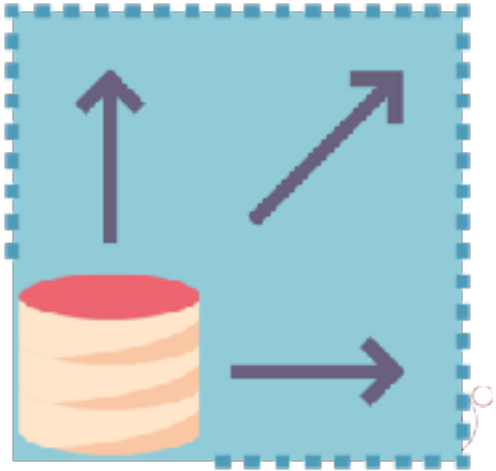


Execution Environments

- Limited runtimes
- Go
- Python
  - Python 3.7.1
  - Flask to handle requests
- JavaScript
  - Node.js 8 default
  - Node.js 10 beta

Concurrency and Scale

- Spin up function instances based on current load
- Functions receive event parameters from platform
- Functions do not share memory or variables
- An instance processes a single request
- Functions should be stateless



Cloud Functions

- Simplest compute option - focus on writing code
- Event-driven
- Deployment very simple
- Software and infrastructure fully managed by Google
- Pay only while code runs

Snapshots

Which of the following is a characteristic of a snapshot?

1. Snapshots are bound to a project and cannot be shared
2. Snapshots can be incremental used to back up data
3. Snapshots are more heavyweight than images
4. Snapshots are exclusively used to create VM instances

Snapshots

Which of the following is a characteristic of a snapshot?

1. Snapshots are bound to a project and cannot be shared
2. **Snapshots can be incremental used to back up data**
3. Snapshots are more heavyweight than images
4. Snapshots are exclusively used to create VM instances

AppEngine

Which of the following is true about the standard environment on AppEngine?

1. Can be used with custom runtimes
2. Runs in a proprietary sandbox on the GCP
3. Runs within a Docker container
4. Takes a couple of minutes to startup

AppEngine

Which of the following is true about the standard environment on AppEngine?

1. Can be used with custom runtimes
2. **Runs in a proprietary sandbox on the GCP**
3. Runs within a Docker container
4. Takes a couple of minutes to startup

AppEngine

Which of the following is true about the flexible environment on AppEngine?

1. Cannot install third party libraries
2. Runs in a proprietary sandbox on the GCP
3. Runs within a Docker container
4. Takes only a few seconds to startup

AppEngine

Which of the following is true about the flexible environment on AppEngine?

1. Cannot install third party libraries
2. Runs in a proprietary sandbox on the GCP
3. **Runs within a Docker containe**r
4. Takes only a few seconds to startup

Compute

Which of the compute options is great for stateless computation which reacts to external events?

1. Cloud Functions
2. AppEngine
3. Container clusters
4. Apps running on VMs

Compute

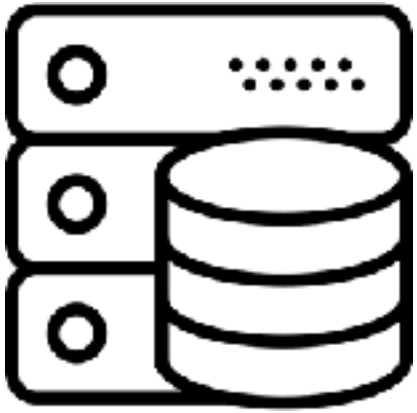
Which of the compute options is great for stateless computation which reacts to external events?

1. **Cloud Functions**
2. AppEngine
3. Container clusters
4. Apps running on VMs

O'REILLY®

Session 3: Storage

Choices in Computing



Compute

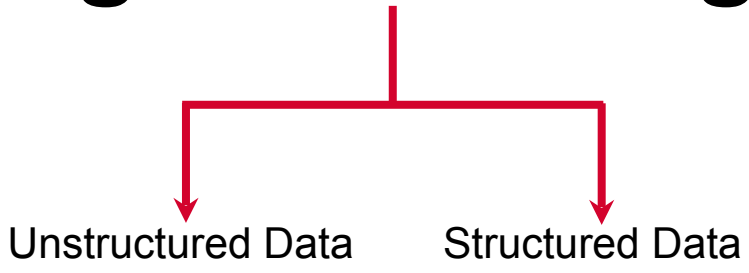
Where is code executed and how?

Storage

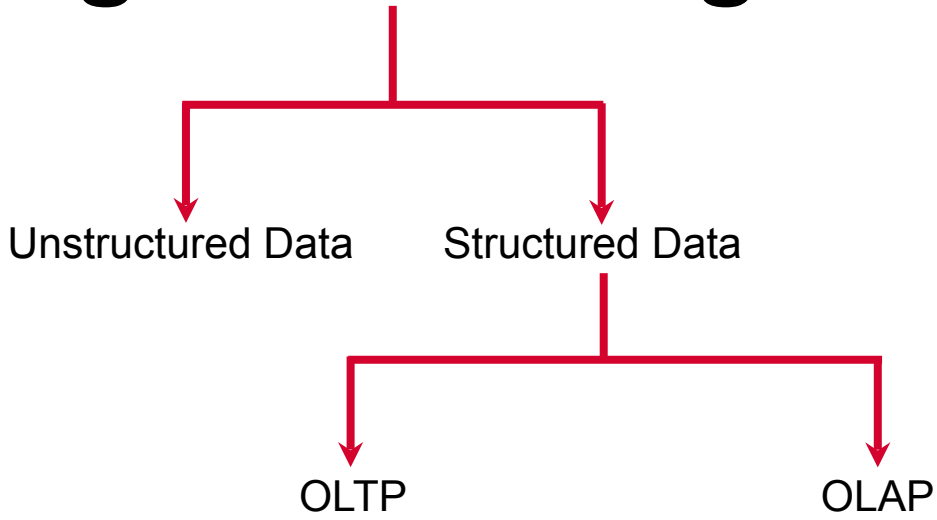
Where is data stored?

Networking, hosting, logging, are choices made after this fundamental decision

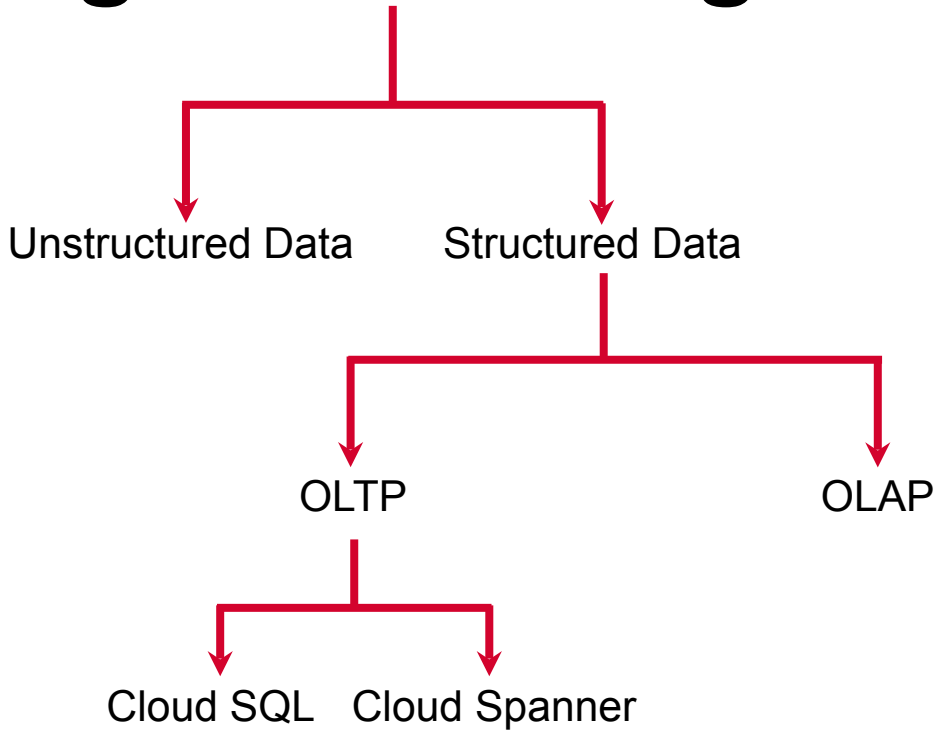
Storage Technologies



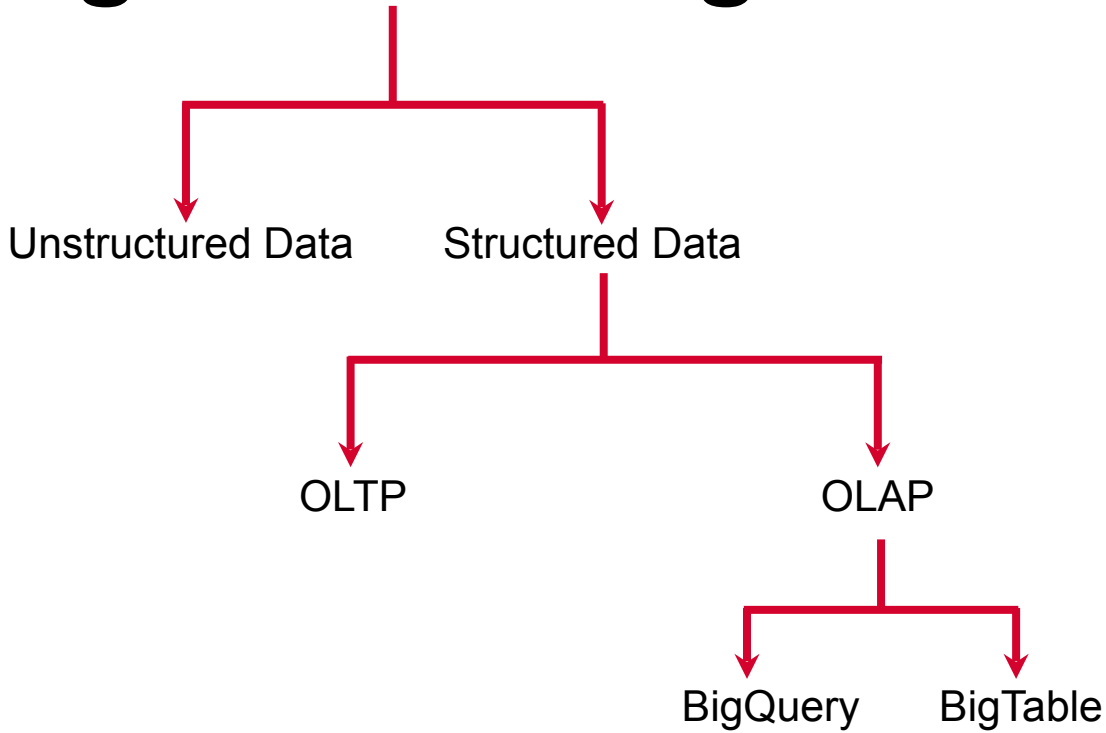
Storage Technologies



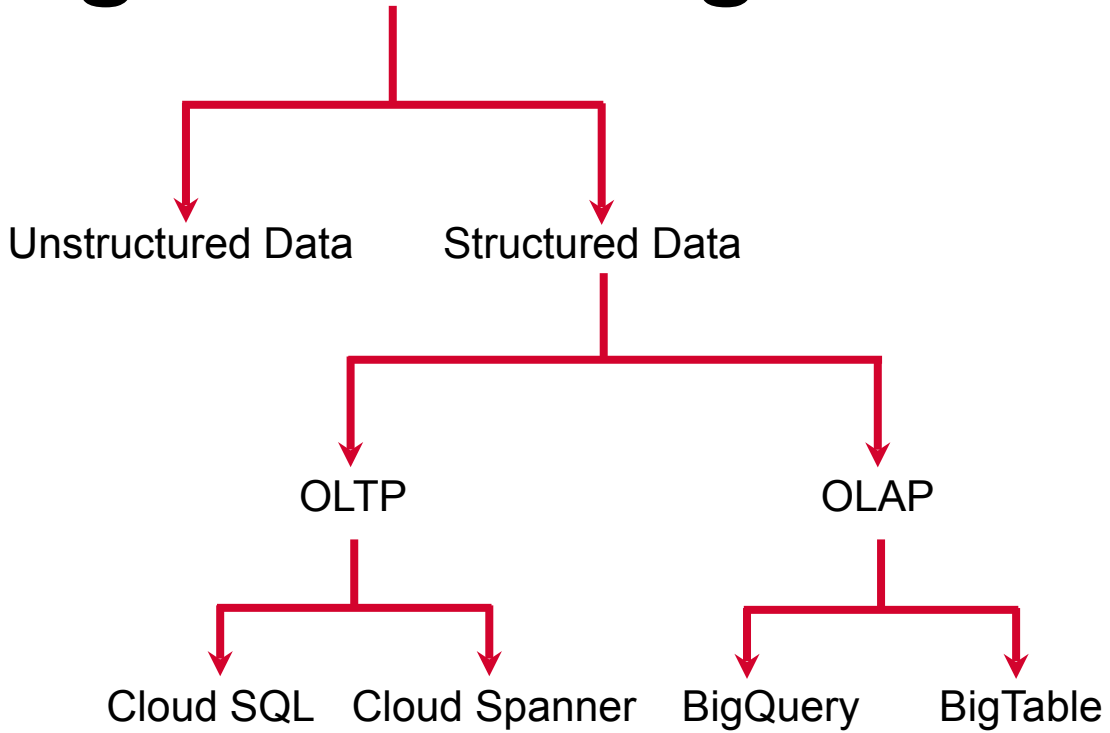
Storage Technologies



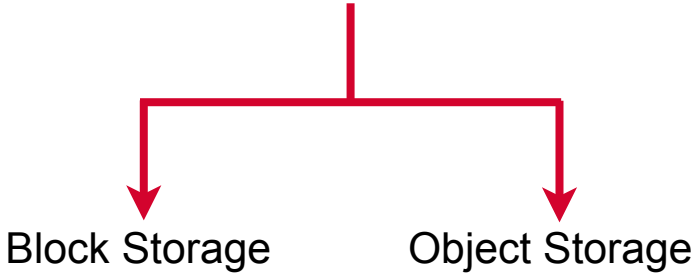
Storage Technologies



Storage Technologies



Unstructured Data



Unstructured Data

Physically addressable storage accessed from compute

Unstructured Data

Logically addressable storage accessed from compute or by human users

Persistent Disks vs. Buckets

Persistent Disks

- Block storage
- Max 64TB in size
- Pay what you allocate
- Tied to GCE VMs
- Zonal (or regional) access

Buckets

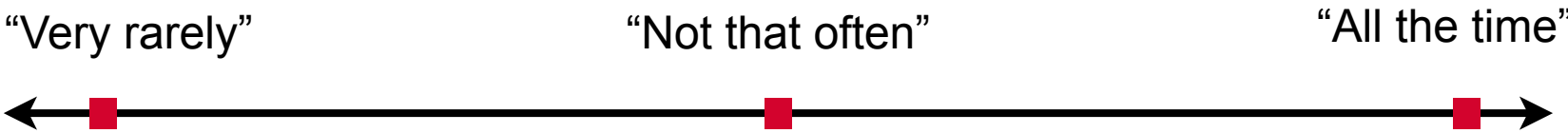
- Object storage
- Infinitely scalable
- Pay what you use
- Independent of GCE VMs
- Global access

Working with GCS

- Cloud console
- gsutil command line utility
  - different from gcloud
- Client libraries
  - programmatic access

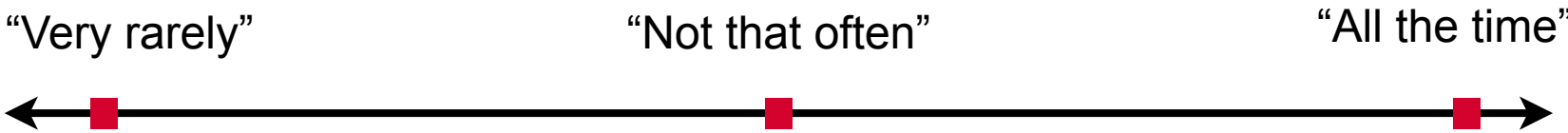
GCS Storage Classes

How often is a data item accessed?



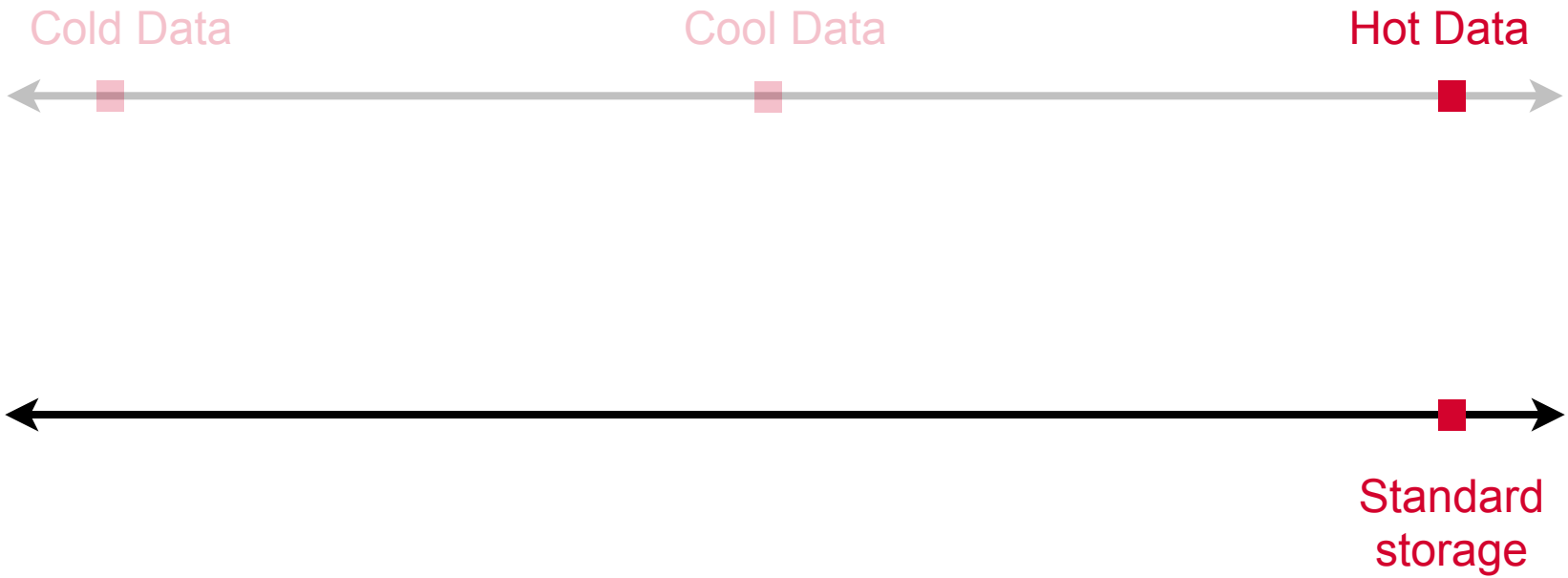
GCS Storage Classes

How often is a data item accessed?

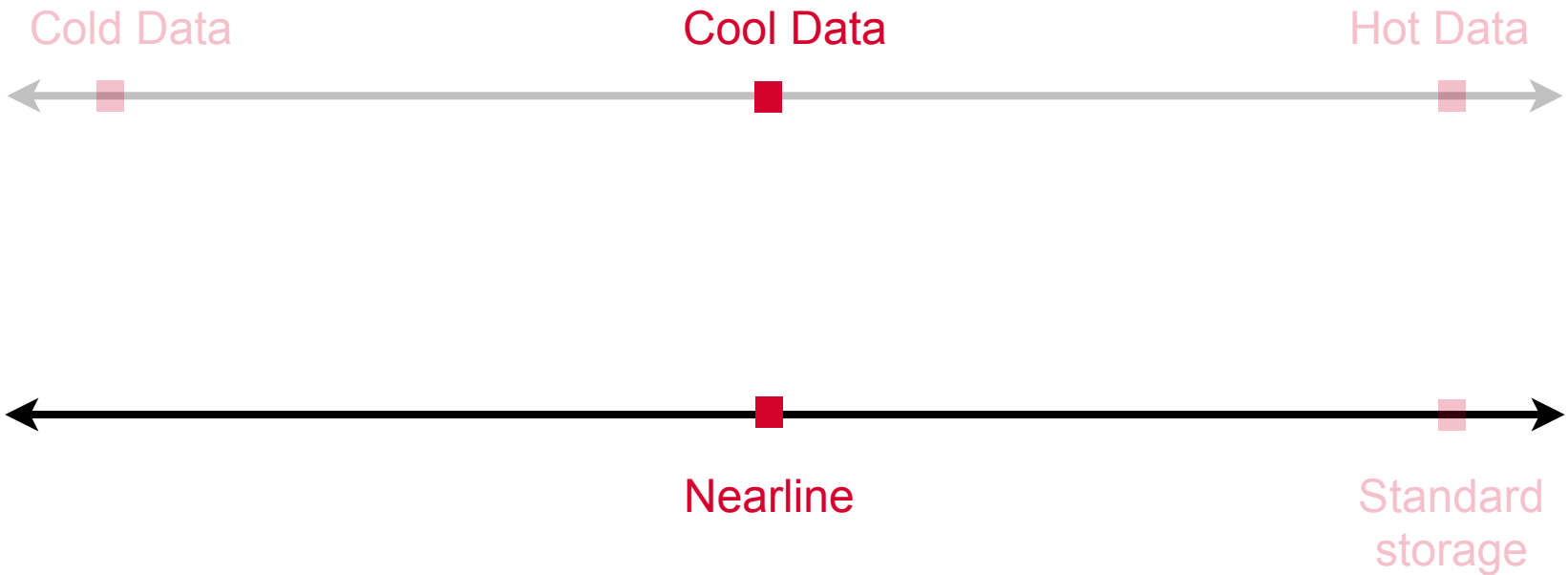


Cold Data	Cool Data	Hot Data
Less than once a year	Once a month	Many times a month

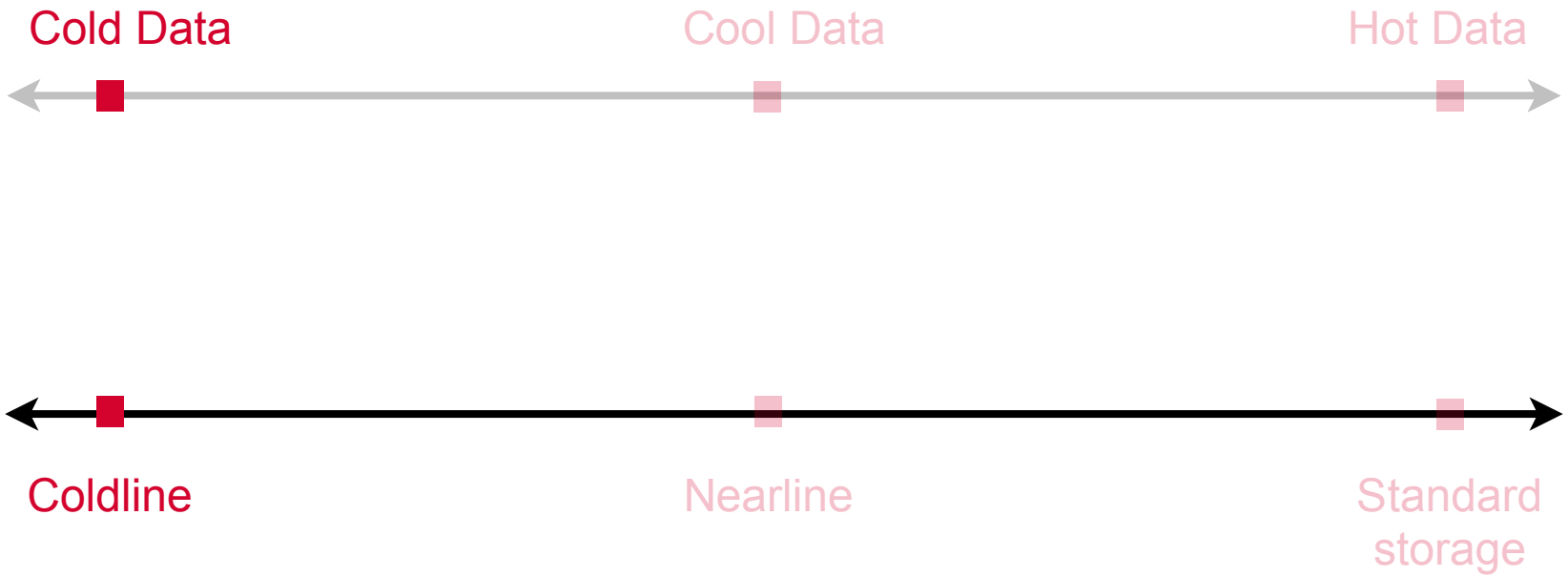
GCS Storage Classes



GCS Storage Classes



GCS Storage Classes



All Storage Classes

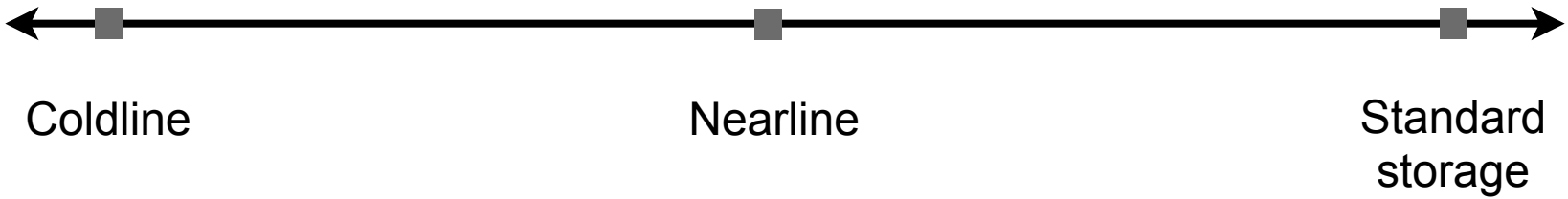
Where is the data
item accessed from?

All Storage Classes

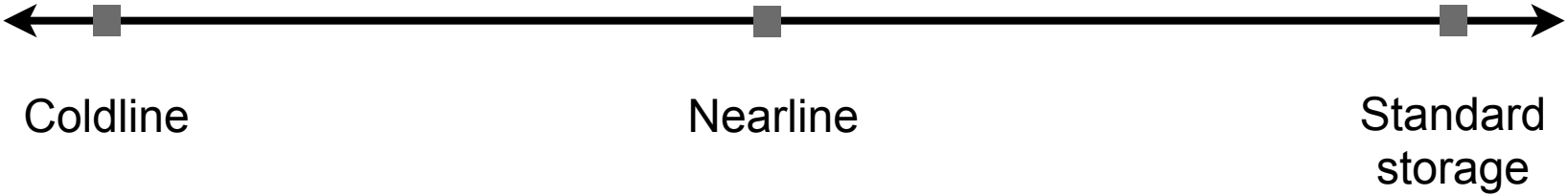
Where is the data item accessed from?

“A specific region”	“Geographically separate locations”	“Accessed from anywhere in the world”
Region	Dual-region	Multi-region

GCS Storage Classes



GCS Storage Classes



GCS Storage Classes

New storage class which is even colder than Coldline

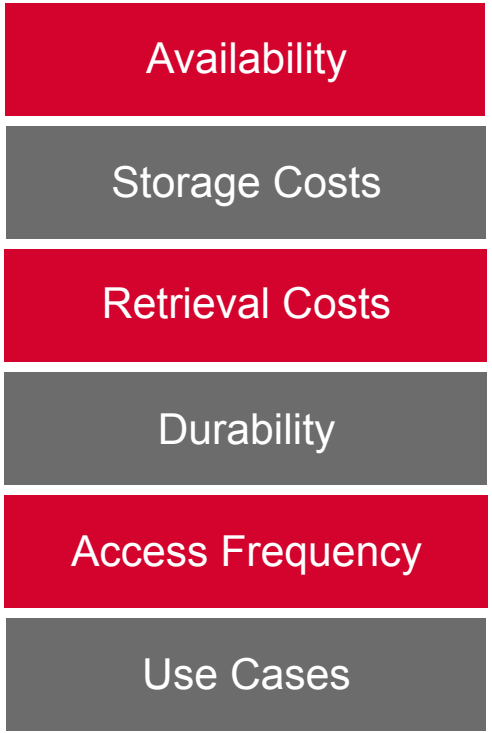
GCS Storage Classes

Lowest cost, highly durable service for data archiving

GCS Storage Classes

365 day minimum storage, no availability SLA, higher data access costs than other storage classes

Coldline and Archive has about the same speed of access as other storage classes (different from AWS Glacier and S3)



Different storage classes represent different trade-offs

Several parameters along which to compare

Availability

Storage Costs

Retrieval Costs

Durability

Access Frequency

Use Cases

Storage Class	Availability
Standard storage (dual and multi-regional)	99.95%
Standard storage (regional)	99.9%
Nearline (regional)	99.0%
Coldline (regional)	99.0%

Dual-region and multi-region buckets are tied to multi-regional locations: US, EU and Asia

Helps adhere to data storage regulations in the US and EU

Availability

Storage Costs

Retrieval Costs

Durability

Access Frequency

Use Cases

Storage Class	Storage Cost (cents/GB/month)
Standard (multi-region)	2.6
Nearline (multi-region)	1.0
Coldline (multi-region)	0.7

Availability

Storage Costs

Retrieval Costs

Durability

Access Frequency

Use Cases

Storage Class	Retrieval Cost (cents/GB)
Standard	**None**
Nearline	1.0
Coldline	5.0

Availability

Storage Costs

Retrieval Costs

Durability

Access Frequency

Use Cases

Storage Class	Minimum Commitment
Standard	**None**
Nearline	30 days*
Coldline	90 days*

*Early deletion will incur charges

Availability

Storage Costs

Retrieval Costs

Durability

Access Frequency

Use Cases

Storage Class	Durability
Standard	99.999999999%
Nearline	99.999999999%
Coldline	99.999999999%

“11 nines”

Availability

Storage Costs

Retrieval Costs

Durability

Access Frequency

Use Cases

Storage Class	Access Frequency
Standard	Daily
Nearline	Monthly or less
Coldline	Monthly or less

Availability

Storage Costs

Retrieval Costs

Durability

Access Frequency

Use Cases

Storage Class	Access Frequency
Standard storage (dual and multi-regional)	Serving websites, interactive workloads, mobile and gaming applications
Standard storage (regional)	Access from Compute Engine VMs or Dataproc cluster
Nearline	Data backup, disaster recovery, archival storage
Coldline	Legal or regulatory needs; also disaster recovery where recovery time is important

GCS for Object Storage

File Storage

- Hierarchical structure
- Support for nesting and directories
- File-level locks

- File and directory headers

Object Storage

- Flat, non-nested structure
- Nested structure merely simulated
- No distributed lock - last write wins
- Unstructured series of bytes

Object Storage Class

- Every bucket has an associated storage class
- Every object also has an associated storage class
- On creation, object inherits storage class of bucket
- The storage class of an object can be changed separately from the storage class of a bucket

Object Versioning

- Needs to be enabled for bucket
- Once enabled, bucket creates archived versions of each object
- Whenever live object is overwritten or deleted
- Version with unique generation number is created
- Each copy charged separately

Object Lifecycle Management

- Can automatically specify changes to object storage class
  - “Change from regional to nearline after 30 days”
  - “Delete all data created before 1/8/2018”
  - “Delete all but 2 most recent versions”

Encryption

- Encrypted even at rest
- Default: Google generates keys
- Can use CSEK
  - *C*ustomer *S*upplied *E*ncryption *K*ey

Cloud IAM

- Identity and Access Management
- Used for all GCP resources
- Role-based Access Control (RBAC)
- Preferred to ACL-based access control

Access Control Lists

- IAM is preferred method for restricting control
- GCS is the only service where ACLs can be used too

Access Control Lists

- Each entry in an ACL includes
  - Permission: What action can be performed
  - Scope: Who can perform the specified action

Restricting Access

- Cloud IAM
  - project-level
  - bucket-level
- Access Control Lists
  - for individual objects
  - e.g. PII (*P*ersonally *I*dentifiable *I*nformation)

Signed URLs

- Time-limited, signed URL
- Provides access without further authentication
  - Valet key
- Specific operations can be specified
  - GET, PUT, DELETE (not POST)

Object Change Notifications

- Can respond to changes in specific object
  - Trigger web hook
- Use in combination with Pub/Sub
  - GCP’s reliable messaging middleware

Storage Class

Which of the following is true for coldline storage?

1. Low cost of storage, high cost of retrieval
2. Low cost of storage, low cost of retrieval
3. High cost of storage, low cost of retrieval
4. High cost of storage, high cost of retrieval

Storage Class

Which of the following is true for coldline storage?

1. **Low cost of storage, high cost of retrieval**
2. Low cost of storage, low cost of retrieval
3. High cost of storage, low cost of retrieval
4. High cost of storage, high cost of retrieval

GCS Buckets

When might you choose to use ACLs to control access on GCP buckets?

1. To allow access to all authenticated users
2. To mitigate DDoS attacks on static content in buckets
3. Role-based access controls for buckets as a whole
4. Access control for individual objects with sensitive information

GCS Buckets

When might you choose to use ACLs to control access on GCP buckets?

1. To allow access to all authenticated users
2. To mitigate DDoS attacks on static content in buckets
3. Role-based access controls for buckets as a whole
4. **Access control for individual objects with sensitive information**

Storage Use Cases

Use Case	Appropriate GCP Service	Non-GCP Equivalents
Block storage	Persistent disks or local SSDs	AWS EBS, Azure Disk
Object/blob storage	Cloud Storage (GCS) buckets	AWS S3, Azure Blob Storage
Relational data - small, regional payloads	Cloud SQL	AWS RDS, Azure SQL Database
Relational data - large, global payloads	Cloud Spanner	
HTML/XML documents with NoSQL access	Firestore	AWS DynamoDB, Azure Cosmos DB
Large, naturally ordered data with NoSQL access	BigTable	
Analytics and complex queries with SQL access	BigQuery	AWS Redshift, Azure Data Warehouse

Cloud SQL

Cloud SQL is the fully-managed MySQL and PostgreSQL database service on the Google Cloud Platform

SQL Server currently available in beta

Transactional support, ACID support

Easiest migration path for on-prem RDBMS

High availability using failover replicas in different zones

Cloud Storage vs. Cloud SQL

Cloud Storage Buckets

- Unstructured (object) data
- gsutil or web console access
- Regional or global
- Scales to any data size
- No support for ACID properties
- Relatively cheap

Cloud SQL

- Relational data (rows and columns)
- SQL access
- Regional only
- Max 10TB in size
- ACID properties for transactions
- Relatively expensive

Google Cloud Spanner

A global, horizontally scaling, strongly consistent relational database service built on proprietary technology

Scales horizontally by adding nodes

ACID support at scale

Relatively expensive and Google proprietary

Cloud SQL vs. Cloud Spanner

Cloud SQL

- Relational (ACID support)
- Regional only
- Scales vertically
- Smaller, lower IOPS
- MySQL and PostgreSQL
- Relatively cheap

Cloud Spanner

- Relational (ACID support)
- Regional or global
- Scales horizontally (add nodes)
- Scales to any size, IOPS
- Google proprietary
- Relatively expensive

Cloud Firestore

Flexible, scalable, NoSQL database for keeping data in sync across client apps.

Mobile and web server development as a part of GCP's Firebase platform

Realtime listeners and offline support

GCP vs. Firebase

GCP

- Makes Google's infrastructure publicly available as services
- Main users are server-side and backend developers
- Services focus on leveraging Google's core infrastructure
- Networking, storage, machine learning, traffic management, scaling

GCP vs. Firebase

GCP

- Makes Google's infrastructure publicly available as services
- Main users are server-side and backend developers
- Services focus on leveraging Google's core infrastructure
- Networking, storage, machine learning, traffic management, scaling

Firebase

- Build mobile and web applications quickly
- Mainly used by client-side application developers
- Services to build applications, engage and grow users
- Realtime database, crashlytics, performance management, messaging

BigQuery is a Data Warehouse that is hard to tell apart from an RDBMS

BigQuery Features

- Serverless: No cluster, no provisioning
- Structured data with fields
- Can ingest streaming data at scale
- Autoscaling
- Automatic high availability
- Simple SQL queries

Cloud SQL vs. BigQuery

Cloud SQL

- OLTP
- SQL RDBMS
- Max 10TB
- Instance as server
- MySQL and PostgreSQL
- Relatively more expensive

BigQuery

- OLAP
- SQL data warehouse
- Scales to PB
- Serverless autoscaling
- Google proprietary
- Relatively cheap

Redis

Very popular in-memory key-value NoSQL database

Cloud Memorystore

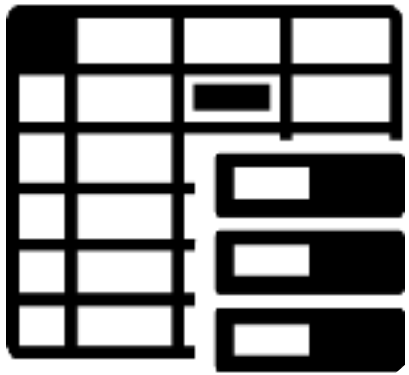
Google managed service for Redis that offers scaling, high availability and a convenient migration path

Google Cloud Bigtable

NoSQL database technology ideal for very large, sparse datasets with sequential ordering in key column; provides very fast writes as well as reads

Choose Bigtable For

- Time series data: Naturally ordered
- Internet of Things data: Constant stream of writes
- Financial data: Often efficiently represented as time series data
- Large datasets > 1 TB with each row < 10 MB



Storage

You have about 5TB of data on your on-premises MySQL database, you want to lift and shift this to the GCP. Which storage technology would you use?

1. Cloud SQL
2. Cloud Spanner
3. BigQuery
4. Cloud Memorystore

Storage

You have about 5TB of data on your on-premises MySQL database, you want to lift and shift this to the GCP. Which storage technology would you use?

1. **Cloud SQL**
2. Cloud Spanner
3. BigQuery
4. Cloud Memorystore

Storage

You have a financial application where transaction support is critical and your clients are distributed globally. Which GCP technology would you use?

1. Cloud SQL
2. Cloud Spanner
3. BigQuery
4. Cloud Memorystore

Storage

You have a financial application where transaction support is critical and your clients are distributed globally. Which GCP technology would you use?

1. Cloud SQL
2. **Cloud Spanner**
3. BigQuery
4. Cloud Memorystore

Storage

You are building a chat application within your product and you want your users to get realtime message updates. Which GCP technology would you choose?

1. Cloud Firestore
2. Cloud Spanner
3. BigQuery
4. Cloud Bigtable

Storage

You are building a chat application within your product and you want your users to get realtime message updates. Which GCP technology would you choose?

1. **Cloud Firestore**
2. Cloud Spanner
3. BigQuery
4. Cloud Bigtable

Storage

You have a realtime stock market application that stores stores stock prices at every tick. You want extremely low latency access to price data at 5 minute intervals. What GCP technology would you choose?

1. Cloud Firestore
2. Cloud Spanner
3. BigQuery
4. Cloud Bigtable

Storage

You have a realtime stock market application that stores stores stock prices at every tick. You want extremely low latency access to price data at 5 minute intervals. What GCP technology would you choose?

1. Cloud Firestore
2. Cloud Spanner
3. BigQuery
4. **Cloud Bigtable**

O'REILLY®

Session 4: Networking

Networking Requirements

Objective

- Resources within a project need to communicate
- Resources on GCP need to communicate with outside world
- Traffic sent to an IP address needs to reach that address
- Platform users need to be able to restrict traffic flows

GCP Solution

- Internal IP addresses
- External IP addresses
- Routes
- Firewall rules

IP addresses, routes and firewall rules all exist inside a GCP resource called a VPC Network

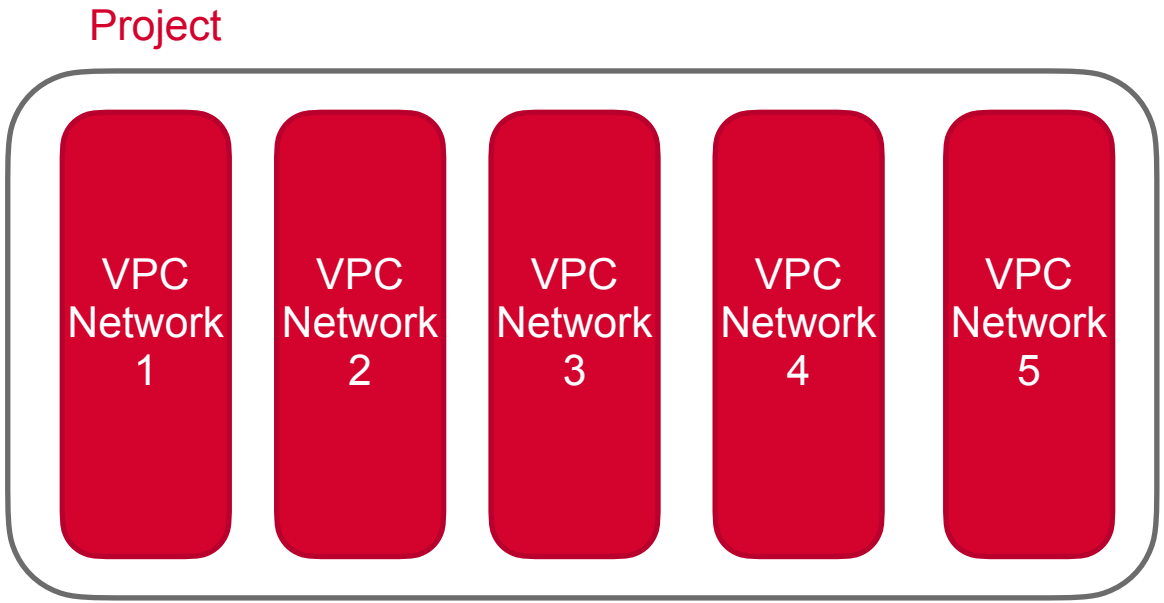
Google Virtual Private Cloud

A VPC network, often just called a network, is a global, private, isolated virtual network partition that provides managed network functionality on the GCP

Google Virtual Private Cloud

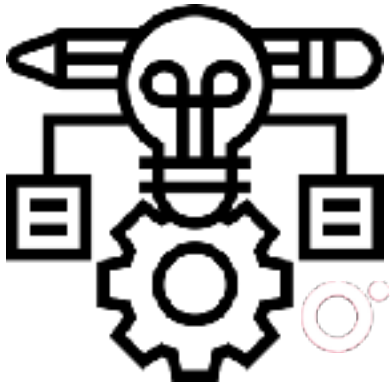
A VPC network, often just called a network, is a global, private, isolated virtual network partition that provides managed network functionality on the GCP

Multiple VPCs in a Project

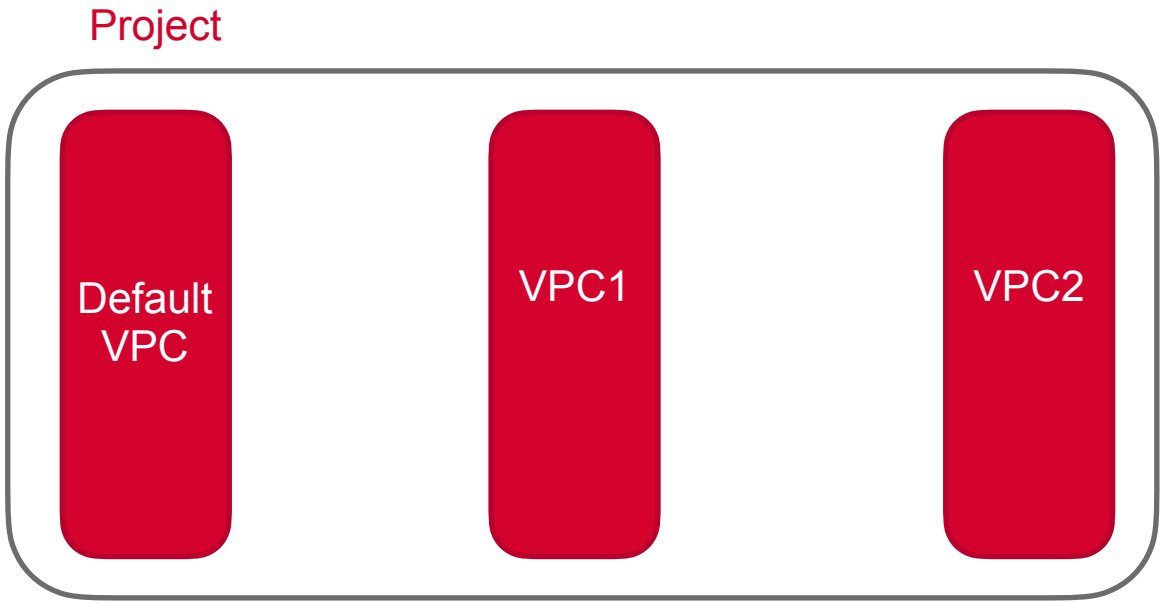


Projects and VPCs

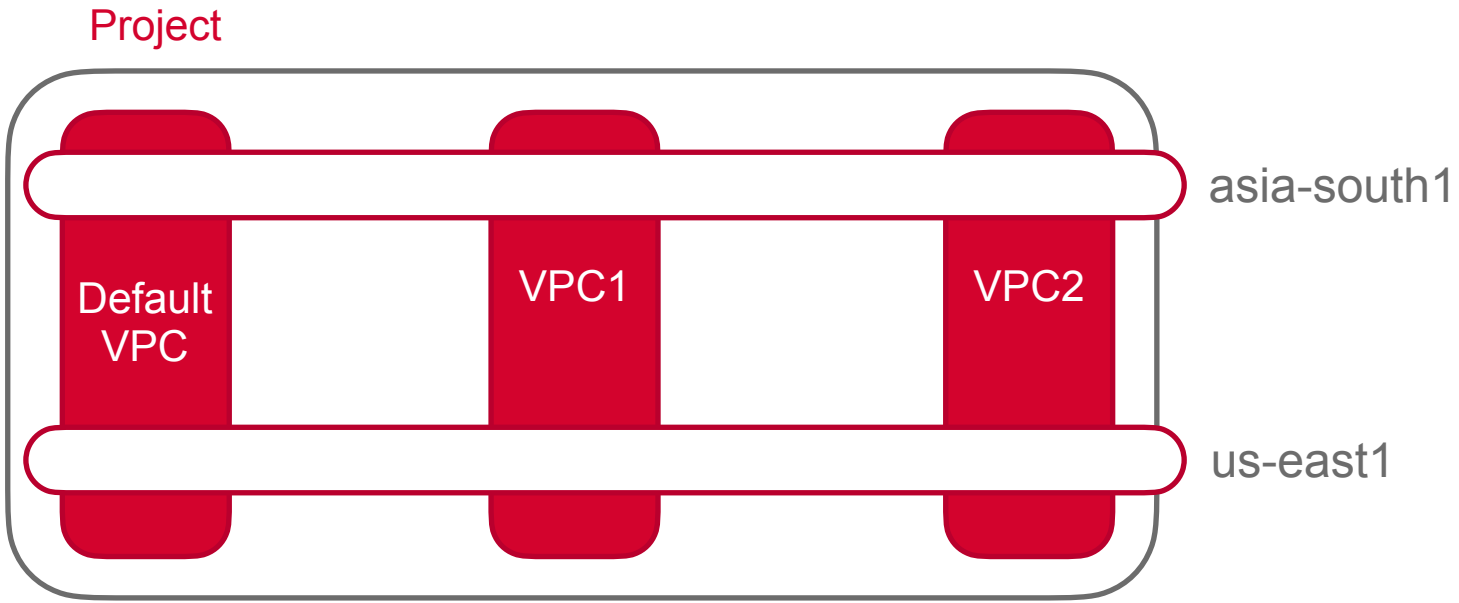
- VPCs are global resources on the GCP
- Each VPC must exist inside a project
- Default VPC pre-created in each project
- Can add additional VPCs
    - Auto Mode
    - Custom Mode



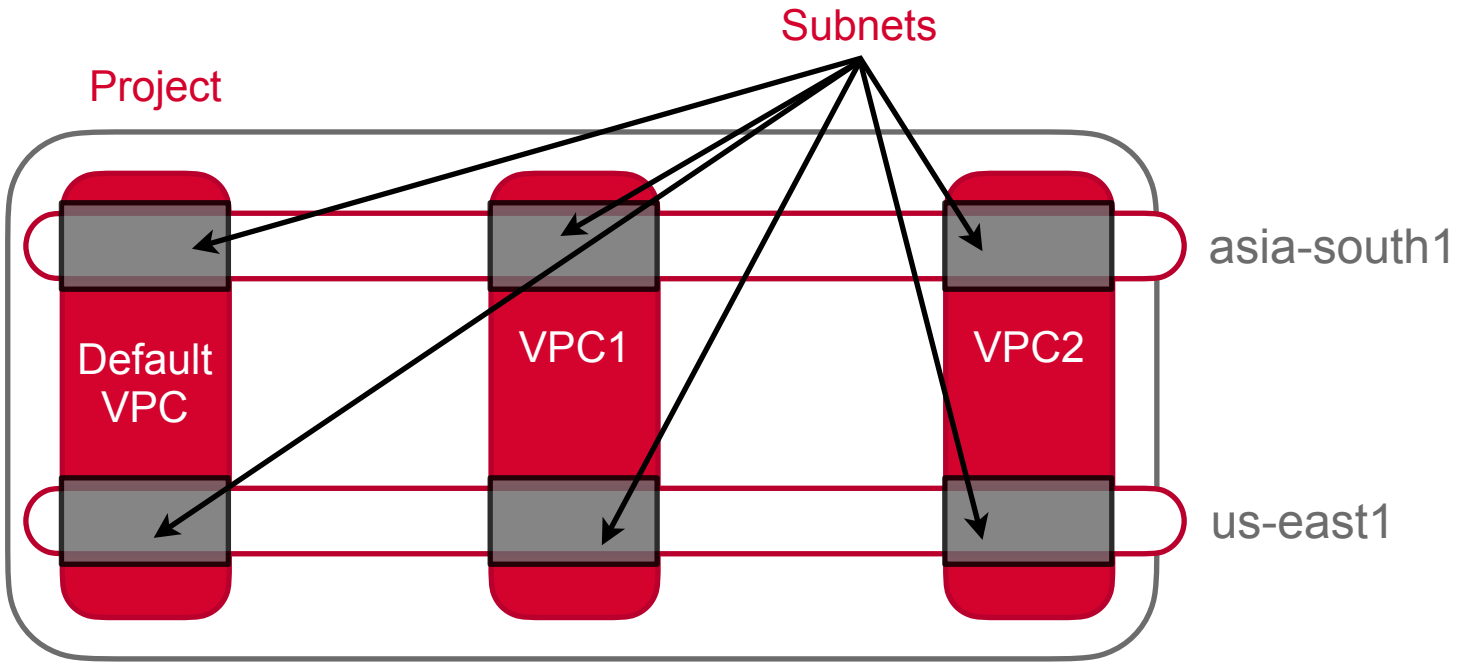
VPCs Are Global



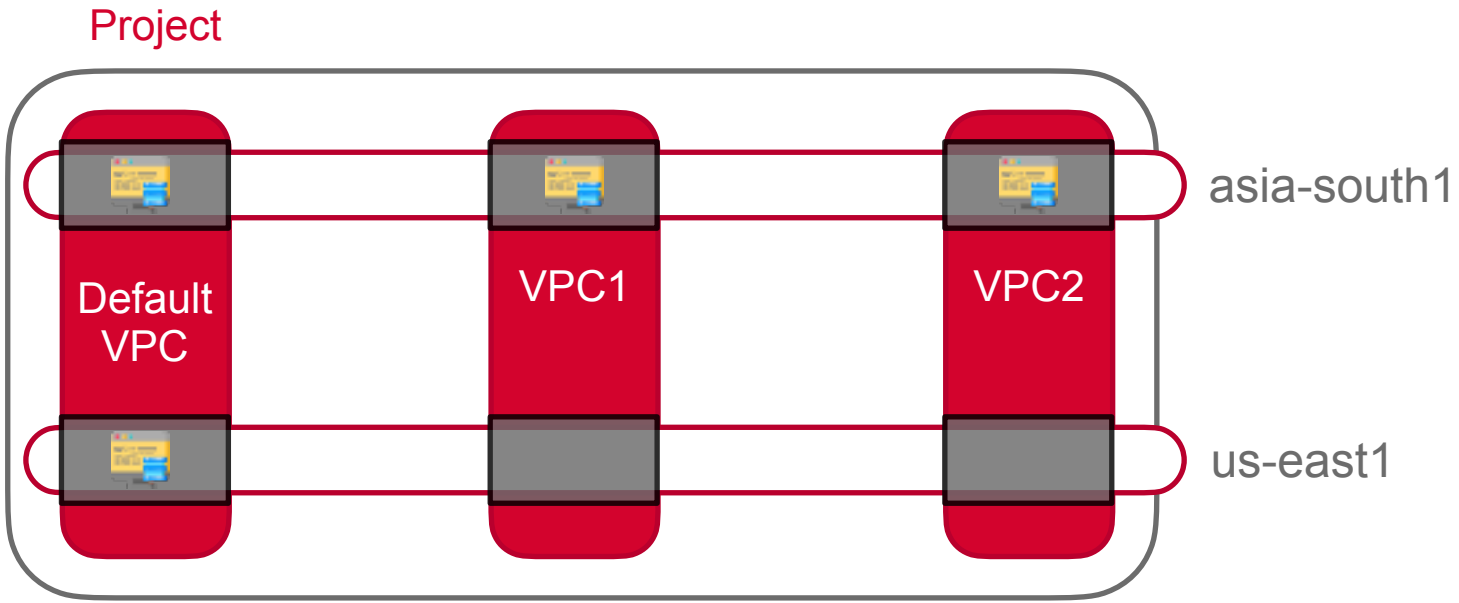
VPCs Are Global



Subnets in Each Region

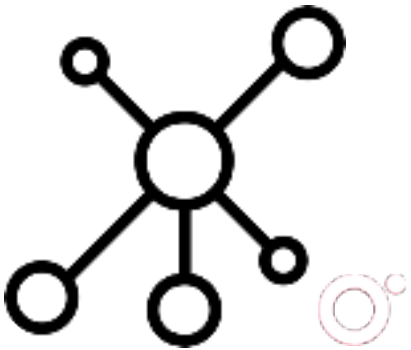


Resources Provisioned on Subnets



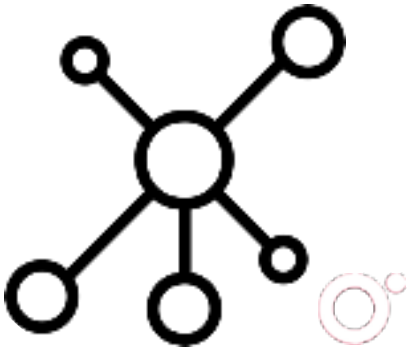
Subnets

- IP range partitions within global VPCs
- VPCs have no IP ranges
- Subnets are regional - can span zones inside a region
- Network has to have at least one subnet before you can use it

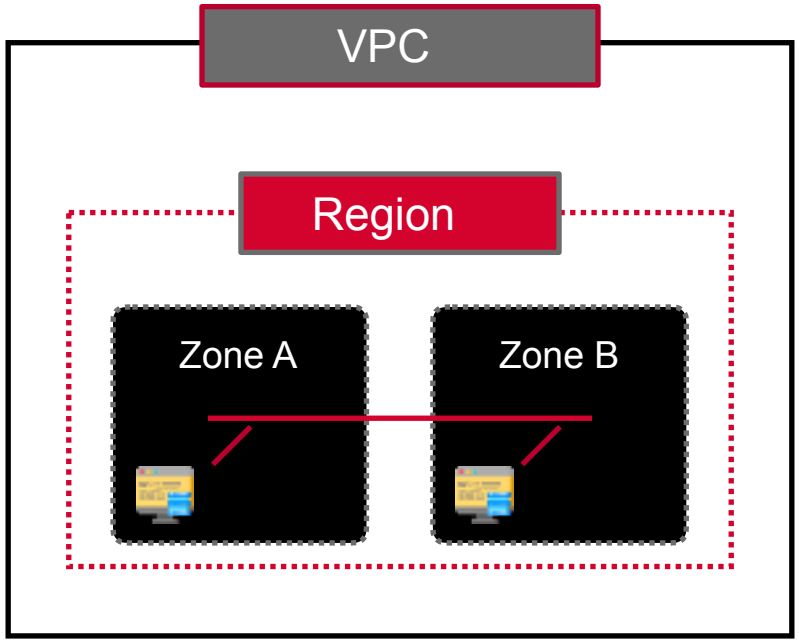


Subnets

- Auto Mode VPCs have pre-created subnets
  - One in each GCP region
- Custom Mode VPCs start with no subnets
  - Full control over which regions have subnets
  - Can create multiple subnets in a region

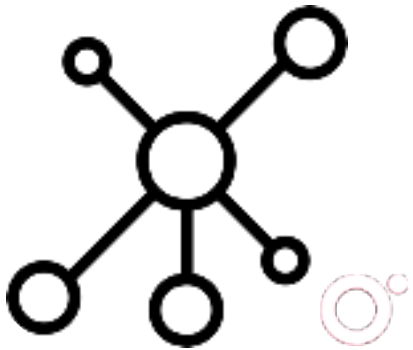


Subnets Span Zones

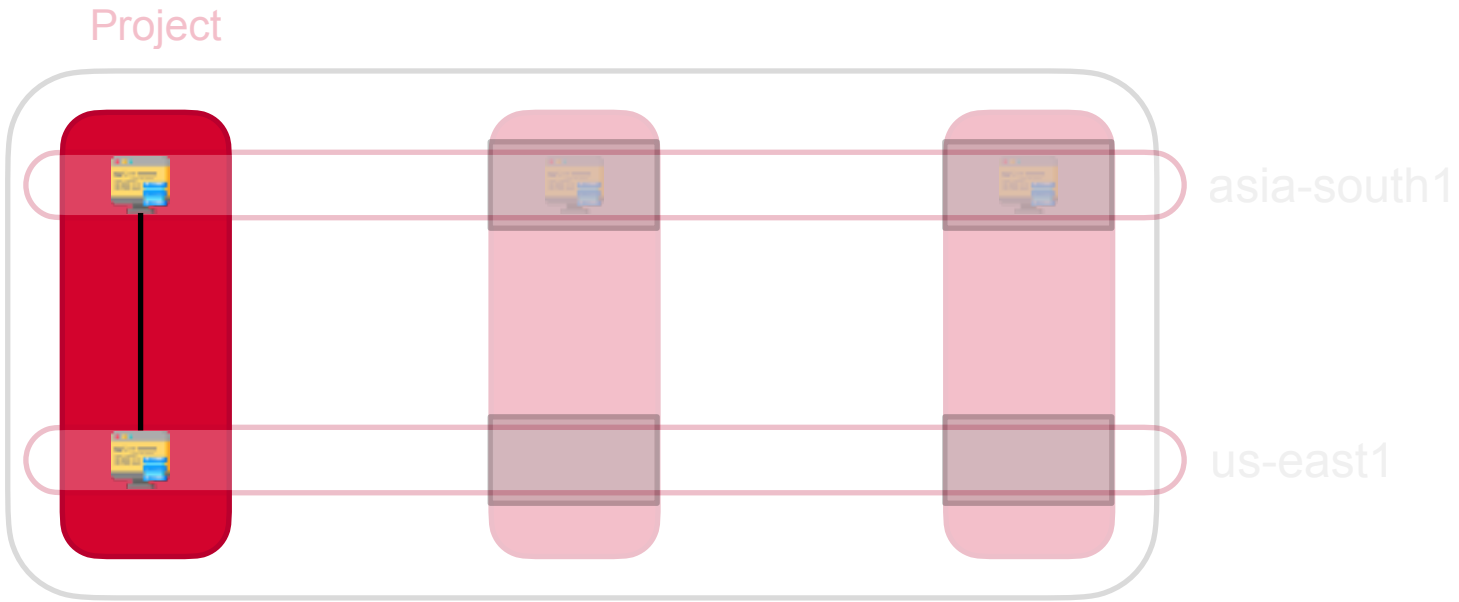


Subnets and IP Ranges

- Each subnet must have primary address range
- Valid RFC 1918 CIDR block
- Subnet ranges in same network cannot overlap
- Subnet ranges in different networks can overlap

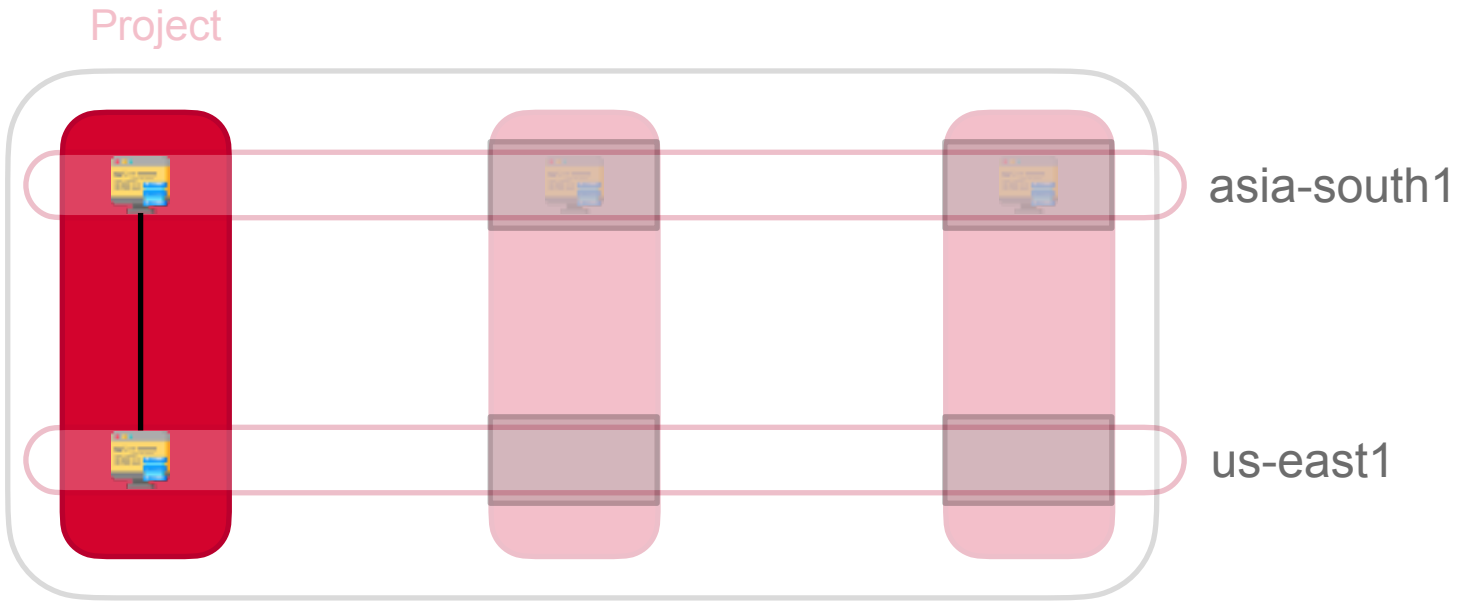


Communication on VPCs



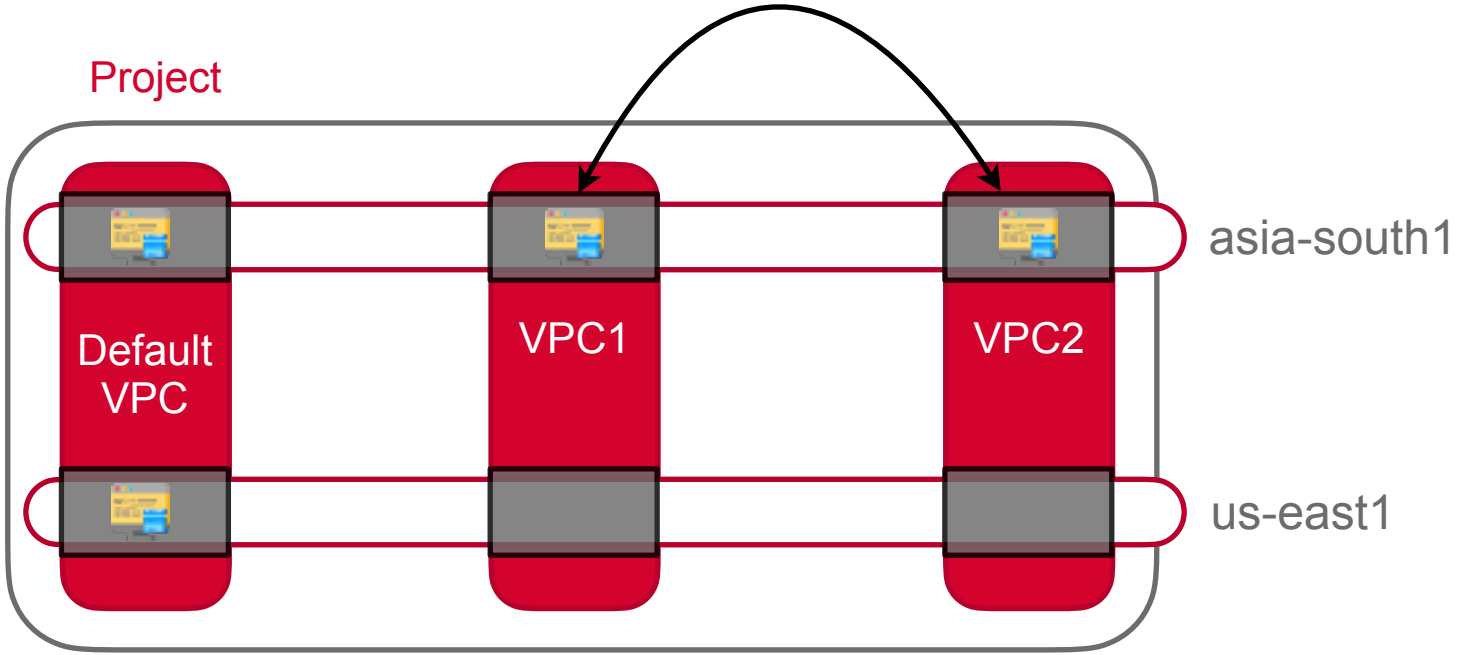
Resources within a VPC communicate using private IP addresses

Communication on VPCs



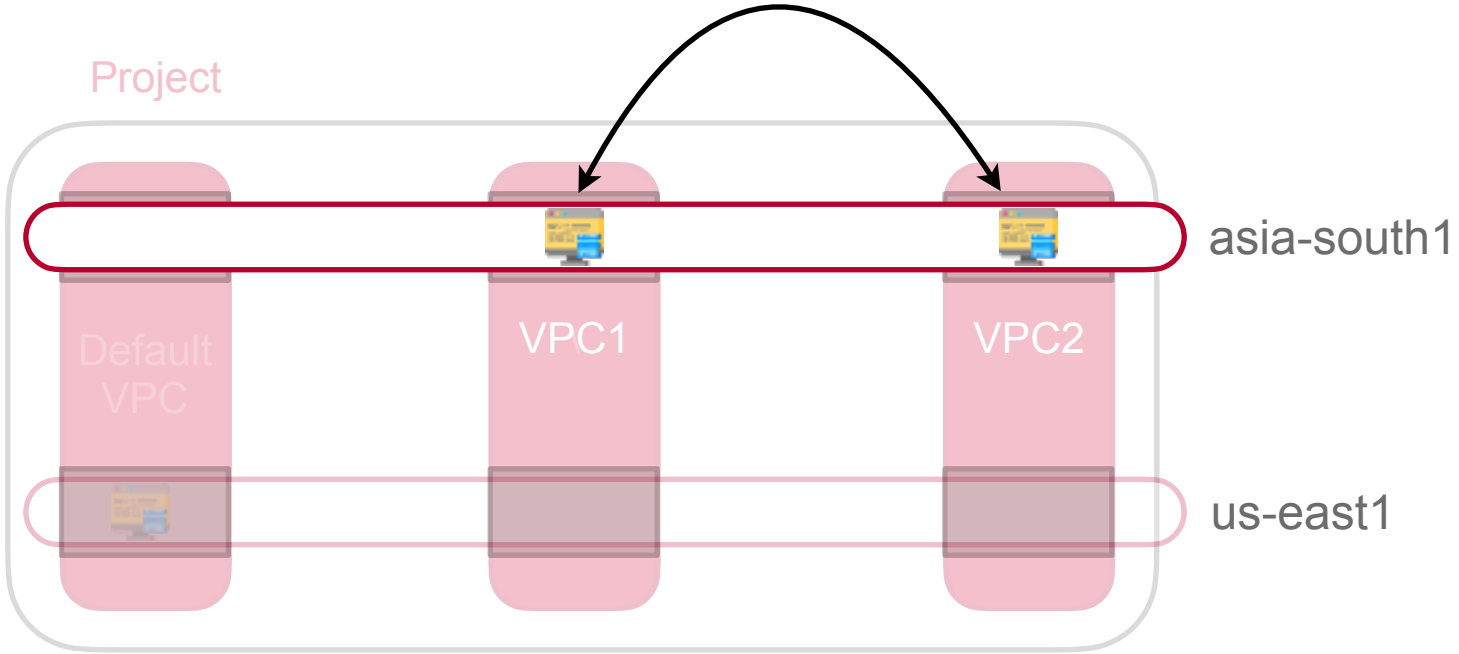
Wherever they are located in the world - irrespective of physical location

Communication on VPCs



Resources on different VPCs communicate over the internet using external IPs

Communication on VPCs



Even though they are in the same region - they may even be in the same zone on the same physical hardware

Default VPC

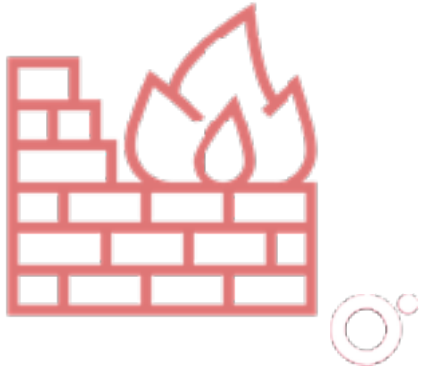
- Pre-created on every project
- Includes subnet for each GCP region
- New subnets added when new regions are created
- Resources created here by default

Default VPC

- Includes routes for all resources
- All VMs on the default VPC can talk to each other
- Default gateway to internet
- Includes several firewall rules

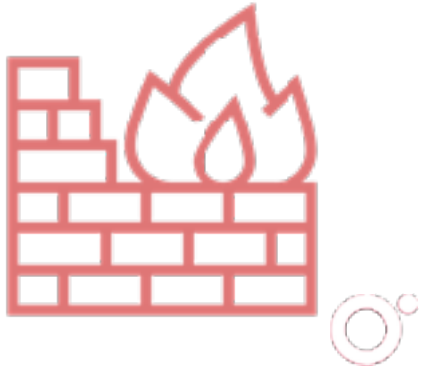
Firewall Rules

- Every VPC is a distributed firewall
- Firewall rules defined in VPC
- Are applied on per-instance basis
- Can also regulate internal traffic



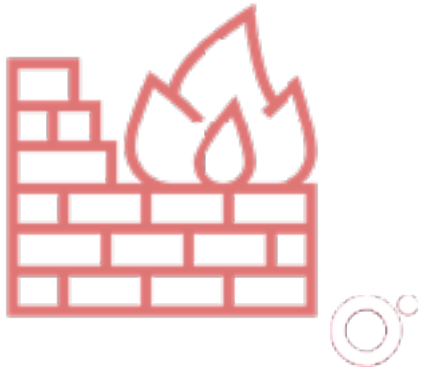
Firewall Rules

- Every VPC has two permanent rules
  - Implied allow egress
  - Implied deny ingress
- Can be overridden by more specific rules
- In addition, default VPC has several rules



Additional Rules in Default VPC

- default-allow-internal
- default-allow-ssh
- default-allow-rdp
- default-allow-icmp



Networking

Which of the following is true for GCP subnets?

1. They are zonal resources
2. They are global resources
3. Every resource has to be provisioned on a subnet
4. They are physical network partitions

Networking

Which of the following is true for GCP subnets?

1. They are zonal resources
2. They are global resources
3. **Every resource has to be provisioned on a subnet**
4. They are physical network partitions

Networking

How do GCP resources in the same region but on different VPCs communicate with each other?

1. Using private IP addresses
2. Using external IP addresses
3. They cannot communicate with each other
4. Using hostnames

Networking

How do GCP resources in the same region but on different VPCs communicate with each other?

1. Using private IP addresses
2. **Using external IP addresses**
3. They cannot communicate with each other
4. Using hostnames

Networking

Which of the following statements is true for the default VPC?

1. They cannot be manually configured once set up
2. They allow external clients to send traffic to all resources by default
3. They come with no firewall rules configured
4. Subnets in new GCP regions are automatically added

Networking

Which of the following statements is true for the default VPC?

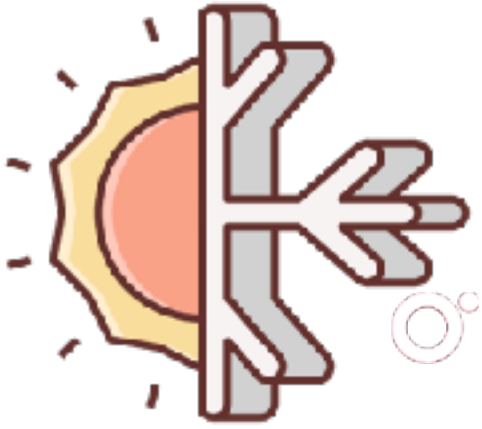
1. They cannot be manually configured once set up
2. They allow external clients to send traffic to all resources by default
3. They come with no firewall rules configured
4. **Subnets in new GCP regions are automatically added**

VPCs on the Google Cloud

Auto Mode	Custom Mode
Subnets automatically created in each region, default firewall rules	Manually create subnets in regions, no defaults preconfigured

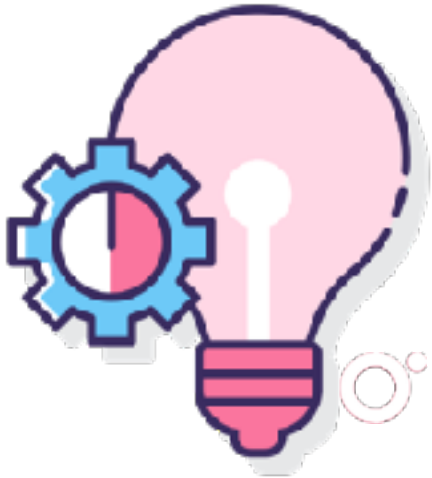
Changing VPC Mode

- Auto -> Custom: Possible
- Custom -> Auto: Not possible



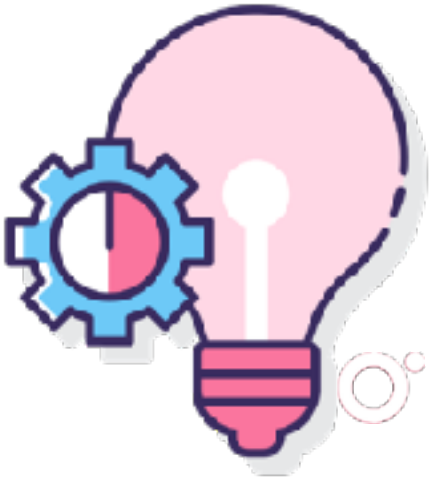
Choosing Auto Mode

- Easy to use, GCP does all the work
- Automatically defined ranges for all regions
- Pre-defined IP ranges



Choosing Custom Mode

- More control over network configuration
- No need for subnets in each region
- Predefined IP ranges might clash with peer network
- Preferably use custom networks with
    - VPC peering
    - Cloud VPN



O'REILLY®

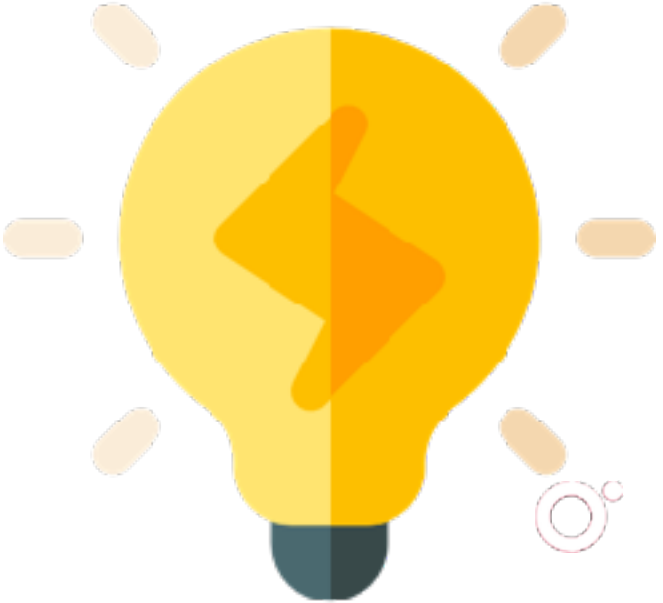
Firewall Rules

Firewall Rules

Restrict and regulate network traffic flows in a VPC

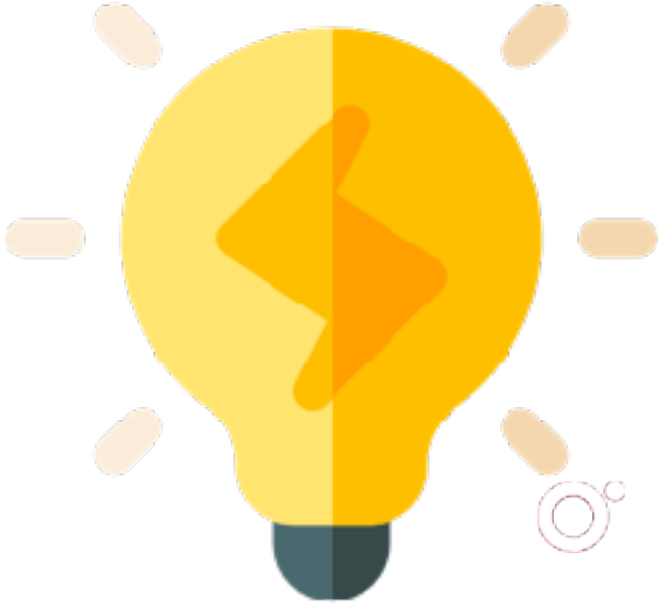
Firewall Rules

- Every VPC has two permanent rules
  - Implied allow egress
  - Implied deny ingress
- Can be overridden by more specific rules



Firewall Rules

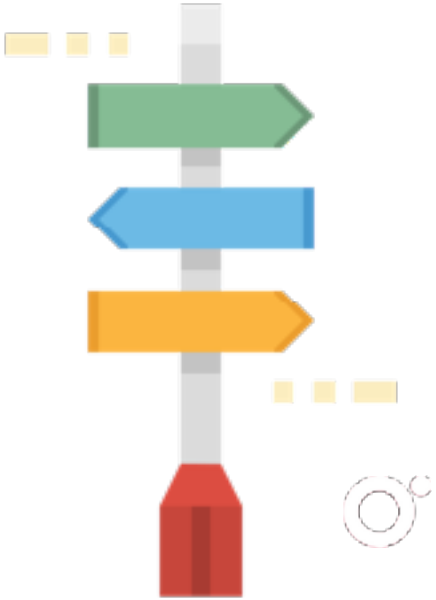
- Every firewall rule has several components
  - Priority (0 highest, 65535 lowest)
  - Direction (ingress/egress)
  - Action (allow/deny)
  - Target
  - Source or destination
  - Protocol and port
  - Enforcement status (enabled/disabled)



Direction and Action

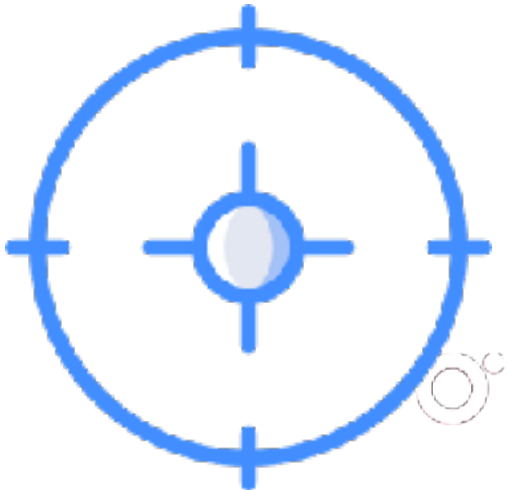
- Direction always defined from perspective of target
  - Ingress: Traffic coming into target from some source
  - Egress: Traffic sent out by target to some destination

- Action to be taken when match found
  - Allow: Permit connection
  - Deny: Block connection
- Rule can only specify one action



Target

- Three possible specifications
  - All instances in network
  - Instances by target tag
  - Instances by target service account



Source or Destination Filter

- Can specify exactly one (not both)
- For ingress rules: specify source
- For egress rules: specify destination

Source and Destination

Sources

- Any IP (0.0.0.0/0)
- Source IP ranges
- Source tags
- Source service accounts
- Some combinations

Destinations

- Any IP (0.0.0.0/0)
- Destination IP ranges

Protocol and Port

- If both omitted - rule applies to all traffic
- Protocol can be name or decimal number
- If port omitted, applies to all ports
- Can specify combinations
  - tcp:80
  - tcp:20-22
  - tcp:80; tcp:443

O'REILLY®

Connecting Networks

Month/Year

Shared VPC

- Share VPC across projects on GCP
- One VPC shared across projects
- Projects must be in the same organization
- Host project, guest resources
- Shared VPC admin to administer the shared VPC

VPC Peering

- Two or more VPCs shared across projects
- Projects need not be in the same organization
- Allows resources on different VPC networks to communicate using private IP addresses
- Reduced latency, higher security and lower cost as compared with using external IPs

Shared VPCs vs. Network Peering

Shared VPCs

- Only within same organization
- One VPC used across projects
- Host and service projects are not peers
- Only single level of sharing possible

Network Peering

- Across organization boundaries
- Multiple VPCs share resources
- Connected VPCs are peers
- Multiple levels of peering possible

Interconnecting Networks

GCP-to-GCP

VPC Network Peering

Enterprise connectivity

Peering and interconnect options

Interconnecting Networks

GCP-to-GCP

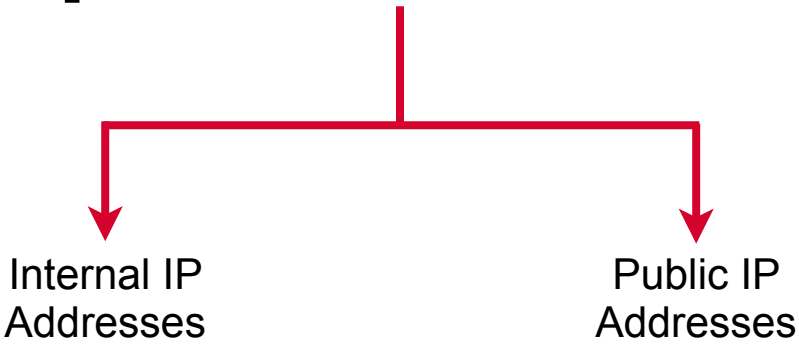
VPC Network Peering

Enterprise connectivity

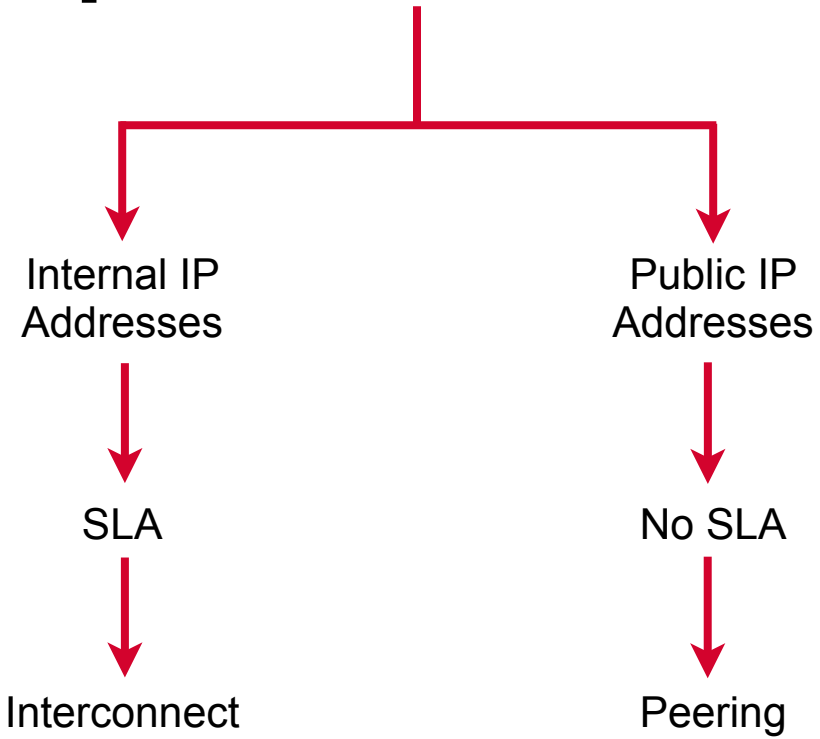
Peering and interconnect options

Connect a cloud network with an on-premise network using private or public IP addresses

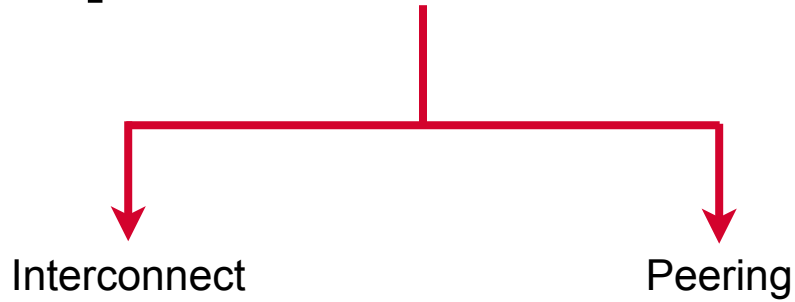
Enterprise Connectivity



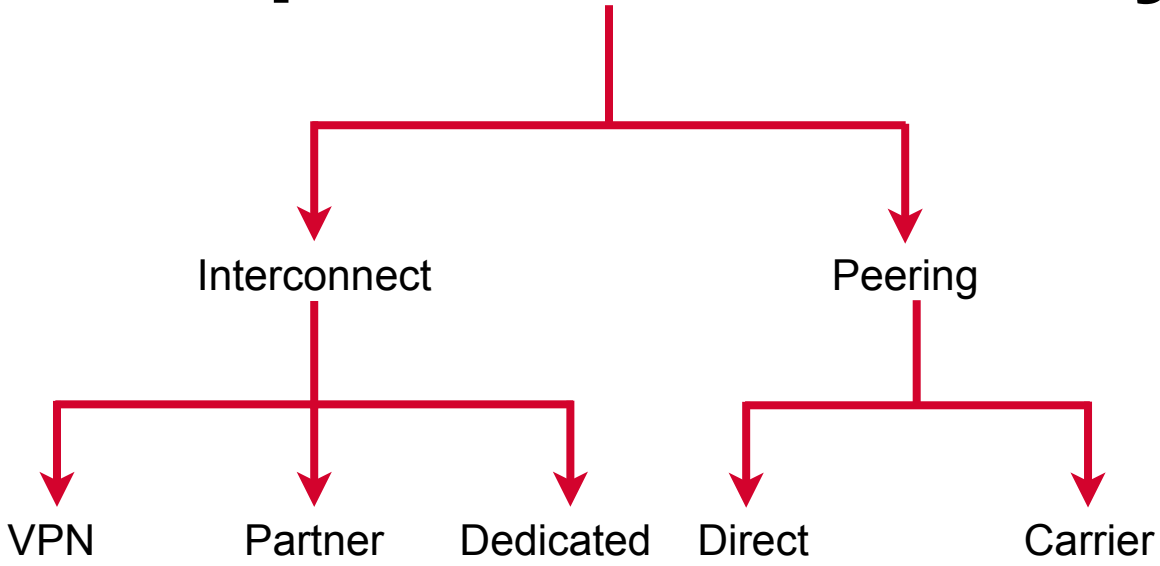
Enterprise Connectivity



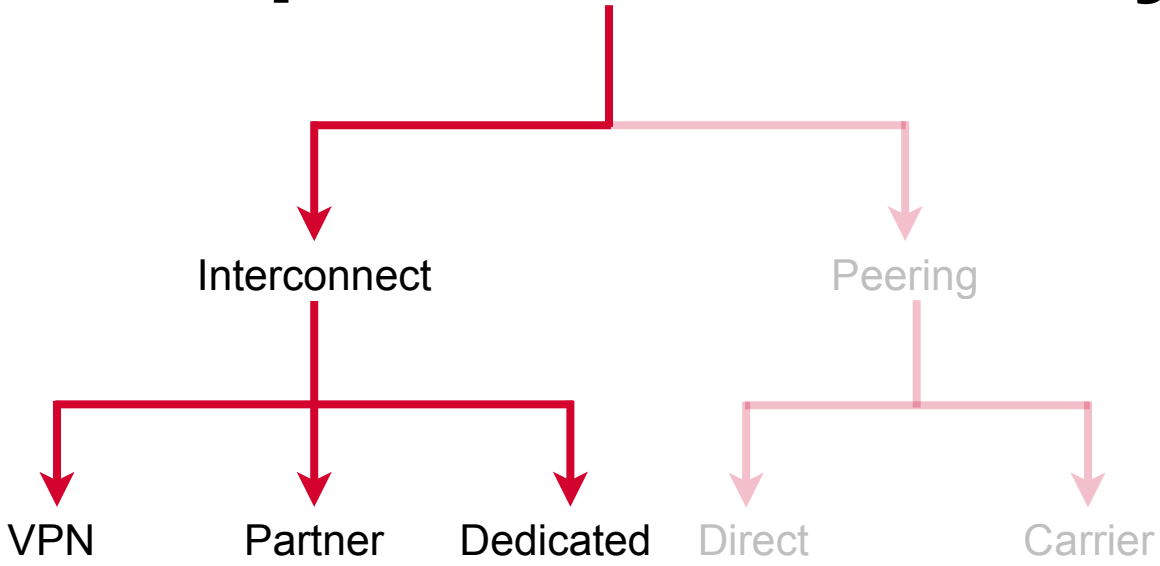
Enterprise Connectivity



Enterprise Connectivity



Enterprise Connectivity



Internal IP addresses in
RFC 1918 address space
With SLA

VPN Tunnel

Configuration Property	Choice
Connection	**Encrypted tunnel to VPC networks through the public Internet**
Access Type	Internal IP addresses in RFC 1918 address space
Capacity	1.5-3 Gbps for each tunnel
Other Considerations	**Requires a VPN device on your on-premises network**

Elements of VPN

- Two VPN gateways
- One for cloud network, another for on-prem network0

- Traffic encrypted at one gateway
- Decrypted at other gateway
- Keys need to be exchanged

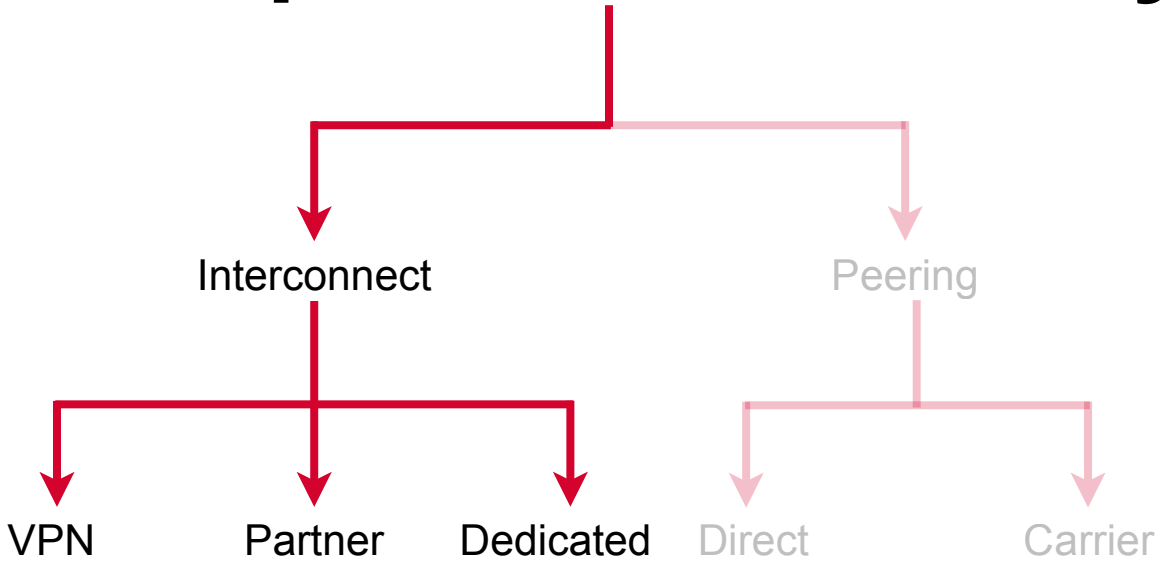
Cloud VPN

Mechanism for secure connection between on-premise and GCP VPC; secure tunnel using two VPN gateways, one at each end

Cloud Router

Fully distributed and managed GCP service (not a physical device) that dynamically exchanges routes between GCP and on-premise networks using BGP (Border Gateway Protocol)

Enterprise Connectivity



Internal IP addresses in
RFC 1918 address space
With SLA

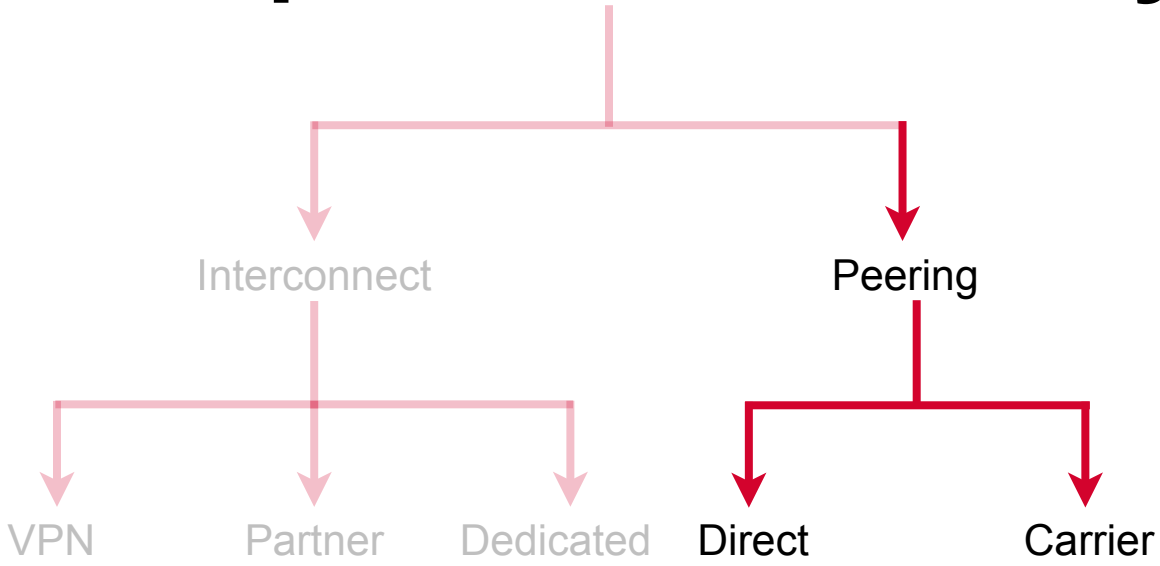
Dedicated Interconnect

Configuration Property	Choice
Connection	Dedicated, direct connection to VPC networks
Access Type	Internal IP addresses in RFC 1918 address space
Capacity	10 Gbps for each link
Other Considerations	**Must have connection in a Google supported colocation facility, either directly or through a carrier**

Partner Interconnect

Configuration Property	Choice
Connection	Dedicated Bandwidth, connection to VPC network through a service provider
Access Type	Internal IP addresses in RFC 1918 address space
Capacity	50Mbps - 10Gbps per connection
Other Considerations	**Service providers might have specific restrictions or requirements**

Enterprise Connectivity



Public IP addresses

No SLA

Direct Peering

Configuration Property	Choice
Connection	Dedicated, direct connection to Google's network
Access Type	Public IP addresses
Capacity	10 Gbps for each link
Other Considerations	**Must have connection in a Google supported colocation facility, either directly or through a carrier**

Carrier Peering

Configuration Property	Choice
Connection	Peering through service provider to Google's public network
Access Type	Public IP addresses
Capacity	Varies based on partner offering
Other Considerations	**Requirements vary by partner**

Networking

Let’s say you have two instances, called VM1 and VM2, in the same VPC network. VM1 has a firewall rule permitting incoming ICMP traffic but the firewall rule allowing instances on the same network to communicate with each other has been deleted.

How can we use VM2 to communicate with VM1?

1.By pinging VM1’s internal IP addresses
2.By pinging VM1’s external IP addresses
3.VM1 and VM2 cannot communicate with each other

Networking

Let’s say you have two instances, called VM1 and VM2, in the same VPC network. VM1 has a firewall rule permitting incoming ICMP traffic but the firewall rule allowing instances on the same network to communicate with each other has been deleted.

How can we use VM2 to communicate with VM1?

1.By pinging VM1’s internal IP addresses
2.By pinging VM1’s external IP addresses
3.VM1 and VM2 cannot communicate with each other

Networking

Which of the following is a difference between using Shared VPC and Peering to interconnect networks in different GCP projects?

1. Shared VPC can span projects in multiple organizations but Peering cannot
2. Shared VPC cannot span projects in multiple organizations but Peering can
3. Shared VPC offers lower latency as compared with Peering
4. Shared VPCs allow communication using internal IPs but with Peering you use external IPs

Networking

Which of the following is a difference between using Shared VPC and Peering to interconnect networks in different GCP projects?

1. Shared VPC can span projects in multiple organizations but Peering cannot
2. **Shared VPC cannot span projects in multiple organizations but Peering can**
3. Shared VPC offers lower latency as compared with Peering
4. Shared VPCs allow communication using internal IPs but with Peering you use external IPs

Networking

Among the following interconnect options in the GCP, which one requires your on premise network to physically meet Google's network in a colocation facility?

1. VPN Tunnel
2. Carrier Peering
3. Dedicated Interconnect
4. Partner Interconnect

Networking

Among the following interconnect options in the GCP, which one requires your on premise network to physically meet Google's network in a colocation facility?

1. VPN Tunnel
2. Carrier Peering
3. **Dedicated Interconnect**
4. Partner Interconnect

O'REILLY®

Session 5: IAM and Security

O'REILLY®

Identity and Access Management

Cloud IAM

Manage identity and access control by defining *who* (identity) has *what access* (role) for *which resource*.

Cloud IAM

Permission to access a resource is not granted *directly* to the end user. Instead, permissions are grouped into *roles*, and roles are granted to authenticated *members*.

Cloud IAM

Permission to access a resource is not granted *directly* to the end user. Instead, permissions are grouped into *roles*, and roles are granted to authenticated *members*.

- Member: GCP identity - user, group, service account
- Role: Collection of permissions
- Policy: Binding members to a role

Role-based Access Control

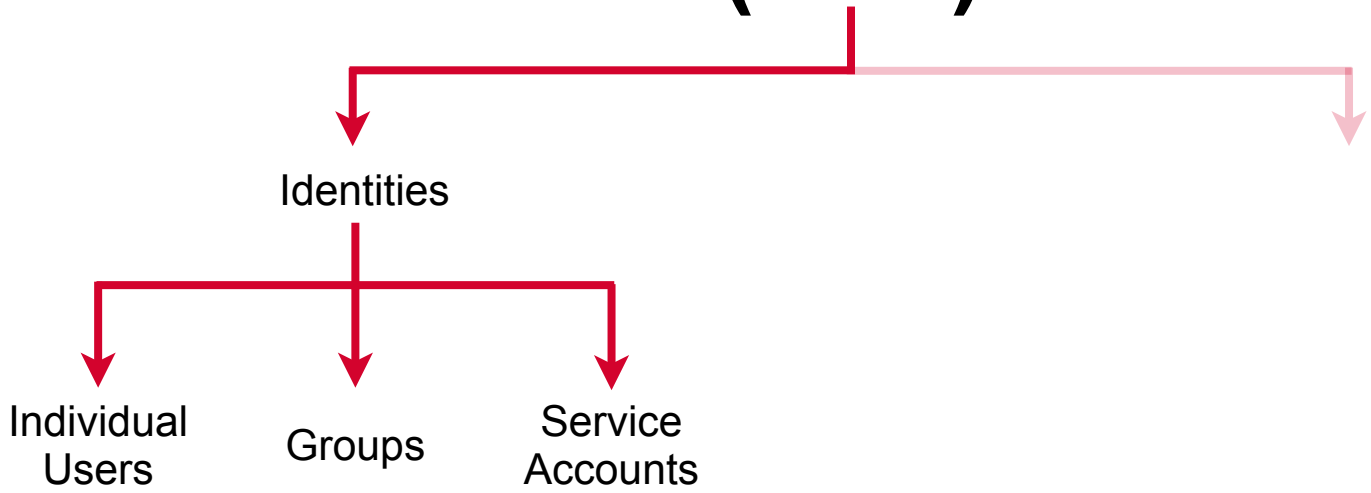
Identity

Permissions

Resource

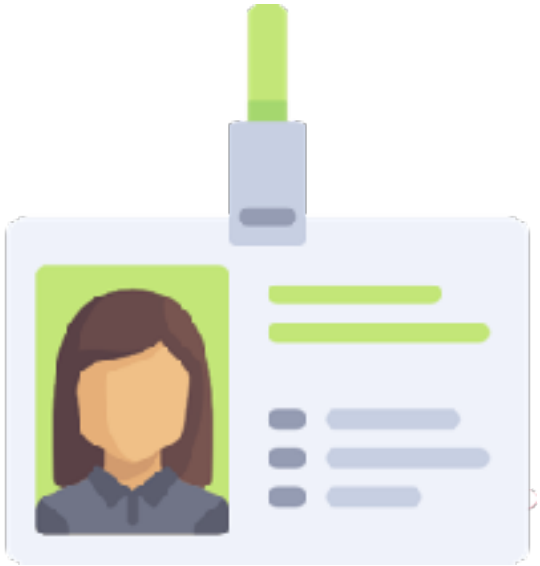
Identity and Access Management (IAM)

Identity and Access Management (IAM)



GCP Identities

- Member types:
  - Google accounts
  - Service accounts
  - Google groups
  - G Suite domains
  - Cloud Identity domains



Google account

A Google account represents a developer, an administrator, or any other person who interacts with GCP.

Service account

A service account is an account that belongs to your application instead of to an individual end user.

Google Group

A Google Group is a named collection of Google accounts and service accounts. Every group has a unique email address that is associated with the group.

G Suite domain

A G Suite domain represents a virtual group of all the Google accounts that have been created in an organization’s G Suite account.

G Suite domains represent your organization's Internet domain name.

Cloud Identity domain

A Cloud Identity domain is like a G Suite domain because it represents a virtual group of all Google accounts in an organization.

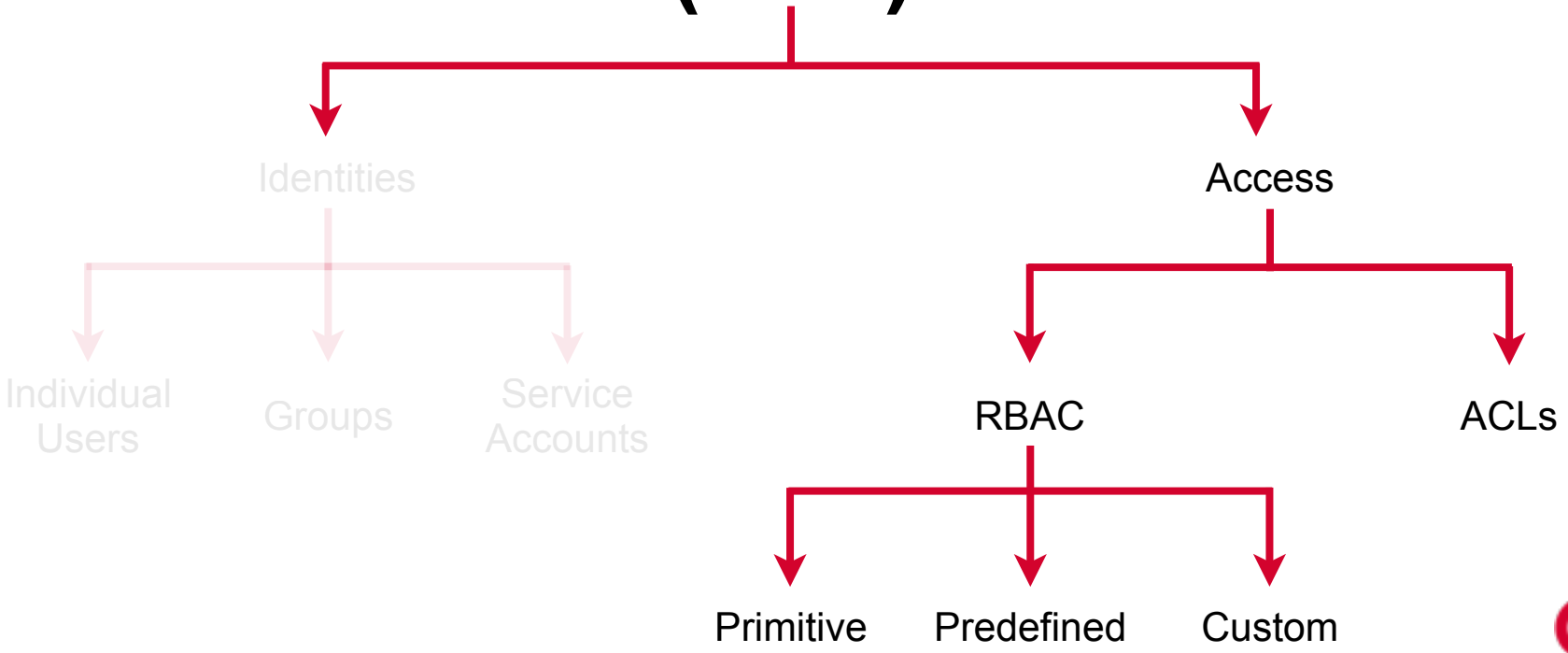
However, Cloud Identity domain users don't have access to G Suite applications and features.

Service account

A service account is an account that belongs to your application instead of to an individual end user.

- Service account is both an identity and a resource
- Can have IAM policies attached to it to determine who can use the service account

Identity and Access Management (IAM)



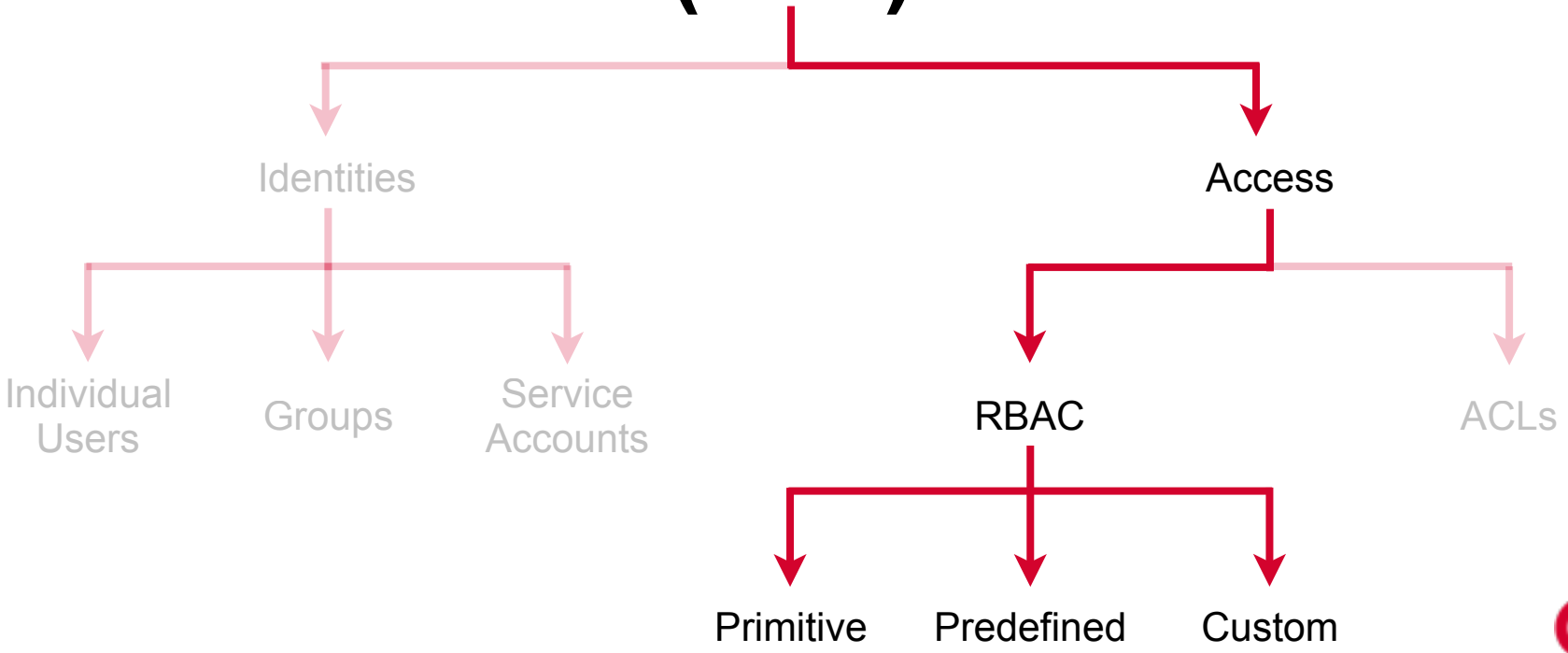
Primitive Roles

Three concentric roles that existed prior to the introduction of Cloud IAM: Owner, Editor, and Viewer of any resource.

Primitive Roles

Role Name	Role Title	Permissions
roles/viewer	Viewer	Permissions for read-only actions that do not affect state, such as viewing existing resources or data
roles/editor	Editor	All viewer permissions, plus permissions for actions that modify state, such as changing existing resources
roles/owner	Owner	All editor permissions and permissions for the following actions: • Manage roles, permissions for a project and all resources in project • Set up billing for a project

Identity and Access Management (IAM)



Predefined Roles

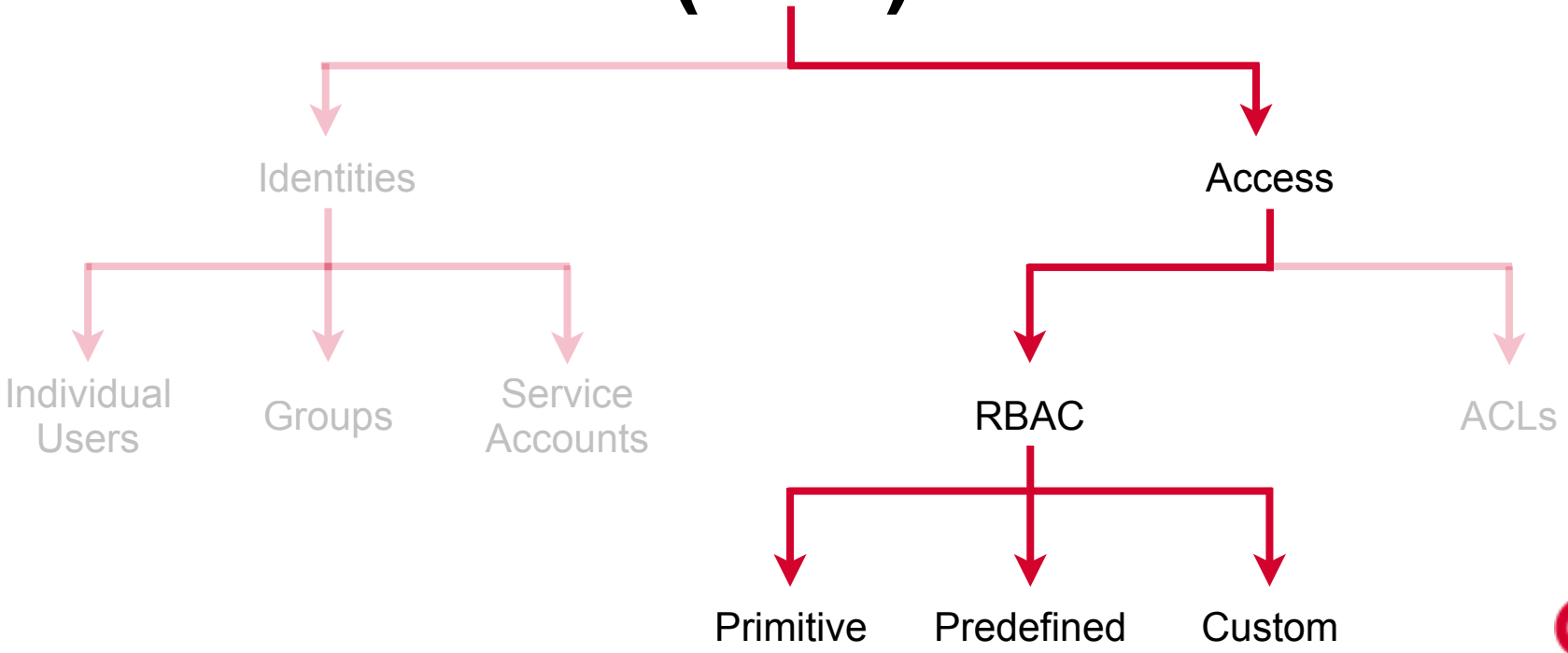
- Project Roles
- App Engine Roles
- BigQuery Roles
- Cloud Bigtable Roles
- Cloud Billing Roles

Predefined Roles

roles/bigquery.dataViewer

bigquery.datasets.get
bigquery.datasets.getIamPolicy
bigquery.models.getData
bigquery.models.getMetadata
bigquery.models.list
bigquery.routines.get
bigquery.routines.list
bigquery.tables.export
bigquery.tables.get
bigquery.tables.getData
bigquery.tables.list
resourcemanager.projects.get
resourcemanager.projects.list

Identity and Access Management (IAM)



Custom Roles

User-defined roles that bundle one or more supported permissions to meet your specific needs.

Not maintained by Google; when new permissions, features, or services are added to GCP, your custom roles will not be updated automatically.

O'REILLY®

Security

BeyondCorp

Google’s implementation of the zero-trust security model

Shifts access control from network perimeter to individual users and devices

Allows employees, contractors and users to work from any location without using VPN

Cloud Armor

Security policies and IP allow and deny lists that work with HTTP(S) load balancing on the GCP

- Works with HTTP(S) load balancer
- Provides built-in defense against DDoS
- Used by Google Search, Gmail, YouTube

Cloud Security Scanner

Cross-site scripting	Flash injection	Mixed content
Clear-text password	Invalid headers	Outdated libraries

Identifies security vulnerabilities in App Engine and Compute Engine web applications

Data Loss Prevention

A strategy for making sure that end users do not send sensitive or critical information outside the corporate network

- Classification of sensitive content whether text or images
- Redaction removes sensitive matches
- De-identification removes identifying features from data

All data stored on the Google Cloud is always encrypted at rest

Three Options for Encryption

Encryption by default - completely managed by the GCP

Customer-managed Encryption Keys (CMEK) reside on cloud

Customer-supplied Encryption Keys (CSEK) reside on-premise

Three Options for Encryption

More control, more ops

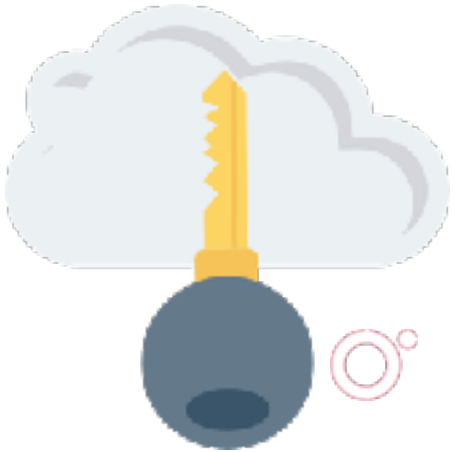
Simpler, less control

Encryption by Default

- Simplest, with the least administrative overhead
- Automatically encrypted when written
- Keys and encryption managed by Google
- Using the same keystore as Google’s production services
- Most data on the GCP protected this way

CMEK Using Cloud Key Management Solution

- Keys stored in the cloud used directly by services
- Create, manage, rotate, destroy keys easily
- Used for application layer encryption in all GCP products



CSEK

- Keys on premises, used to encrypt cloud services
- Google keeps key in memory, does not write out to disk
- Provide key as a part of API calls

CSEK

- Support available for Cloud Storage and Compute Engine
- Compliance or sensitivity issues require own key managed on premises

Google Cloud KMS

Cloud-hosted key management service for generating, using, rotating and destroying cryptographic keys. Easiest way to implement CMEK on GCP.

Two Categories of Keys

Symmetric

Both encryption and decryption are performed using the same key

Asymmetric

Have a public/private: key pair one for encryption, one for decryption

Three Purposes of Keys

Symmetric encryption

Encrypt and decrypt messages using the same key

Asymmetric signing

Private key to encrypt text, public key to decrypt text

Asymmetric encryption

Public key to encrypt text, private key to decrypt text

IAM

A developer writes an application that invokes various GCP services. Following best practices the application should get its permissions from:

1. The project editor
2. The project owner
3. The developer's identity
4. A service account

IAM

A developer writes an application that invokes various GCP services. Following best practices the application should get its permissions from:

1. The project editor
2. The project owner
3. The developer's identity
4. **A service account**

IAM

When new permissions are created, the following entity will not automatically be updated with any additional appropriate permissions:

1. Custom roles
2. Primitive roles
3. Project owner
4. Predefined roles

IAM

When new permissions are created, the following entity will not automatically be updated with any additional appropriate permissions:

1. **Custom roles**
2. Primitive roles
3. Project owner
4. Predefined roles

IAM

Which of the following GCP resources support Access Control Lists (ACLs) in addition to Role-based Access Control?

1.Service accounts
2.Cloud Storage buckets
3.BigQuery
4.GCE VMs

IAM

Which of the following GCP resources support Access Control Lists (ACLs) in addition to Role-based Access Control?

1.Service accounts
2.Cloud Storage buckets
3.BigQuery
4.GCE VMs

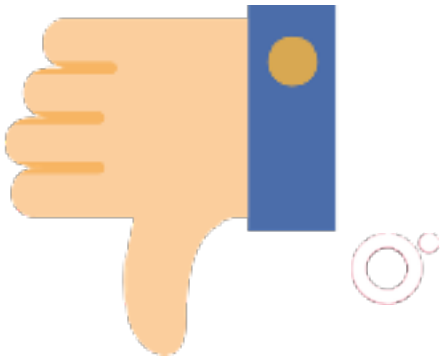
O'REILLY®

Session 6: Containers

GCP Compute Choices

Drawbacks of VMs

- Contain guest OS
  - Introduces platform dependency
  - Bloats image size to GB (apps far smaller)
- Heavyweight
  - Slow to boot up
- Not trivial to migrate
  - VM migration tools needed



Container

A container image is a lightweight, stand-alone, executable package of a piece of software that includes everything needed to run it; code, runtime, system tools, system libraries, settings

Container

- Contains applications
- And all of the application’s dependencies
- Platform independent
- Runs on layer of abstraction
- Docker Runtime (for Docker containers)

Attractions of Containers

- No guest OS
  - Platform independent
  - Considerably smaller than VM images
- Lightweight
  - Small and fast
  - Quick to start
  - Speeds up autoscaling
- Hybrid, multi-cloud
  - Hybrid: Work on-premise and on cloud
  - Multi-cloud: Not tied to any specific cloud platform

Standalone Container Limitations

- No autohealing
  - Crashed containers won’t restart automatically
  - Need higher level orchestration
- No scaling or autoscaling
  - Overloaded containers don’t spawn more automatically
  - Need higher level orchestration
- No load balancing
  - Containers can’t share load automatically
  - Need higher level orchestration
- No isolation
  - Crashing containers can take each other down
  - Need sandbox to separate them

Kubernetes

Orchestration technology for containers - convert isolated containers running on different hardware into a cluster

Kubernetes is fast emerging as middle-ground between IaaS and PaaS in a hybrid, multi-cloud world

IaaS vs. PaaS

Infrastructure-as-a-Service

- Heavy operational burden
- Migration is hard

Platform-as-a-Service

- Provider lock-in
- Migration is very hard

Compute Choices

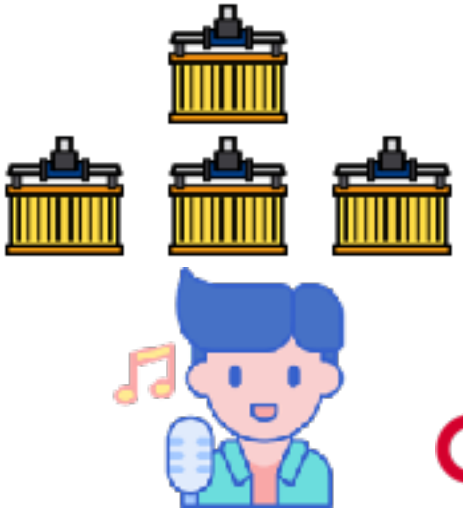
IAAS

PAAS

Containers

Container Clusters

Kubernetes



Kubernetes as Orchestrator

- Fault-tolerance
- Autohealing
- Isolation
- Scaling
- Autoscaling
- Load balancing

All of these are possible in a Kubernetes cluster using higher level abstractions

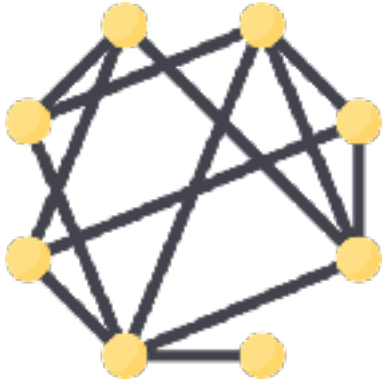
Google Kubernetes Engine (GKE)

- Service for working with Kubernetes clusters on GCP
- Runs Kubernetes on GCE VM instances
- Many more abstractions and a lot more support than using plain Kubernetes on-premises

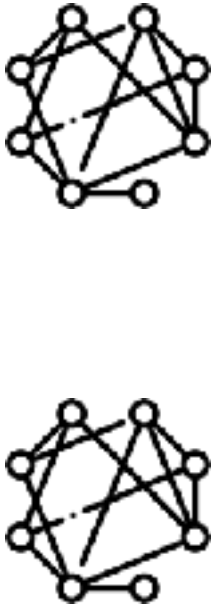
GKE On-Prem

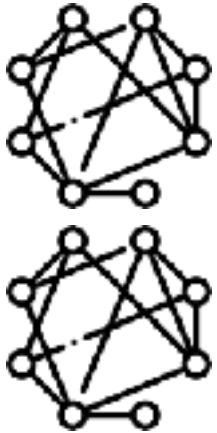
Part of the Anthos platform for hybrid clouds. Provides tools, integrations and access to help unify access and treat on-prem clusters as though they run on the cloud.

Kubernetes Clusters



Master node



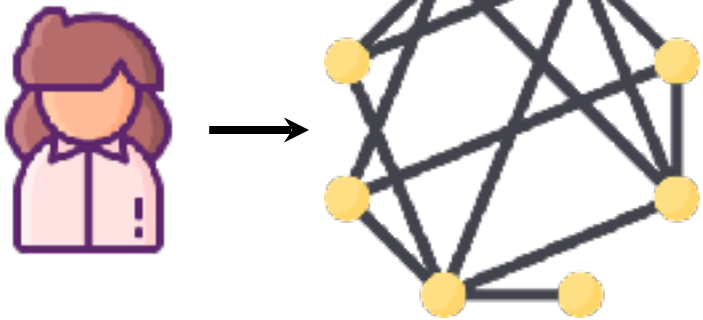


Worker nodes

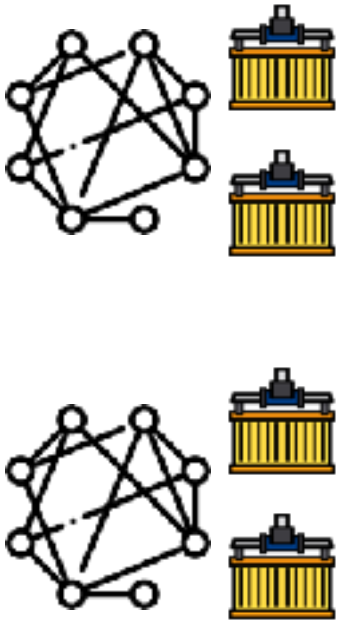
Master

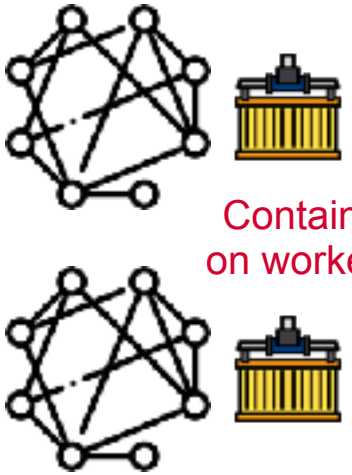
- One or more nodes designated master node
- Unified endpoint for your cluster
- Managed by GKE, not visible directly to user
- Multi-master for high-availability
- Kubernetes Control Plane directed from here

Kubernetes Clusters



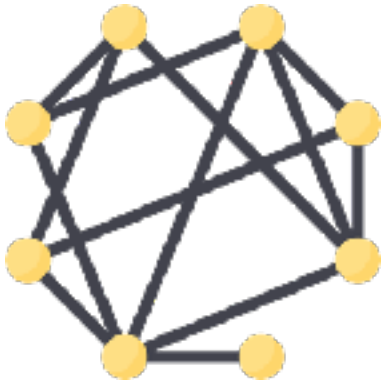
Users interact with the master node





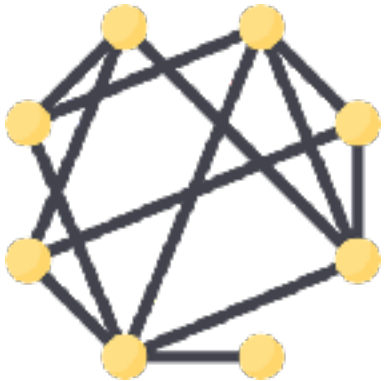
Containers run on worker nodes

Nodes



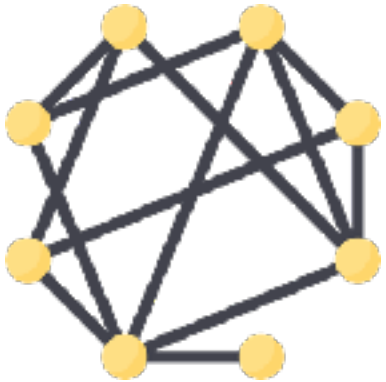
Nodes are on-premises or cloud VMs on which containers are run

Nodes

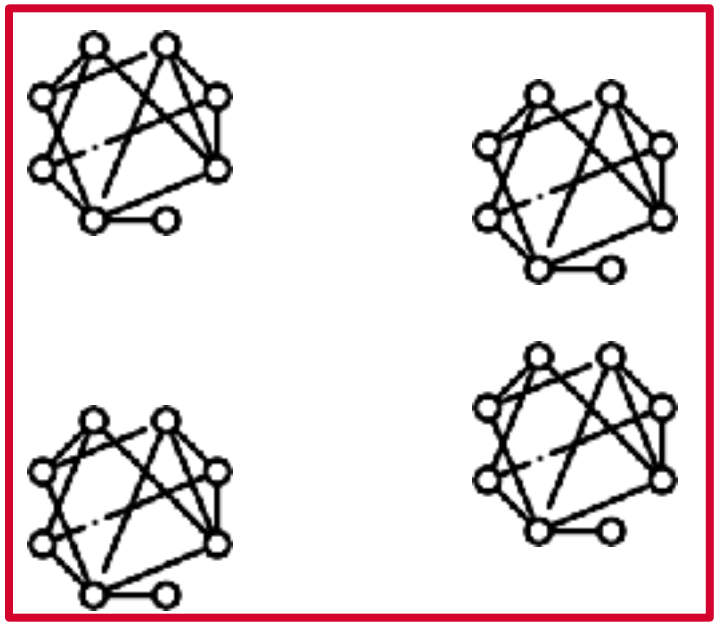


Run the services needed to host Docker containers - communicate with the master

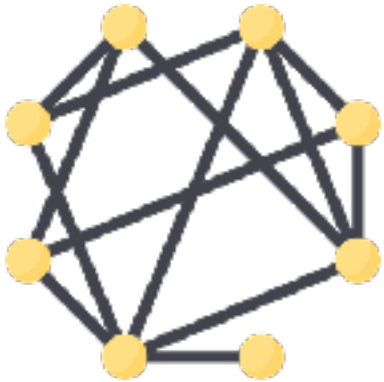
Node Pools



A subset of node instances which have the same configuration are called node pools



Node Images



Special operating system images are available on the Google Cloud to run on Kubernetes nodes

Kubernetes does not interact directly with containers

Instead it uses a number of higher-level entities referred to as **objects**

Different objects responsible for applications with **different characteristics**

YAML Specification Files for Objects

Current State

The current state of the object

Desired State

The end state of the object

YAML Specification Files for Objects

Controllers in the Kubernetes cluster run reconciliation loops to get the actual state to match the desired state

Using the Google Kubernetes Engine almost completely eliminates the need to explicitly configure YAML files

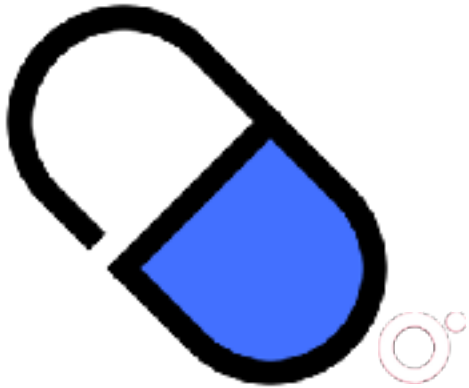
Simply use the web console or the gcloud command line utility

How Does Kubernetes Make this Possible?

- Fault-tolerance
- Autohealing
- Isolation
- Scaling
- Autoscaling
- Load balancing

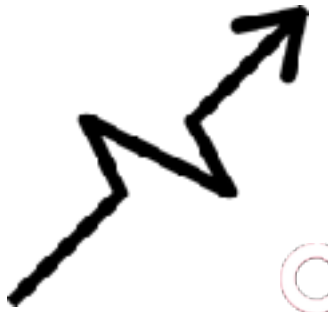
Pods on Kubernetes Nodes

- Smallest and most basic deployable object in Kubernetes
- Can not run a container without enclosing pod
- Pods provide isolation between containers
- Pods act as sandbox for enclosed containers
- Multi-container pods are possible
  - tightly-coupled
  - not usually recommended



Higher-level Abstractions

- ReplicaSet
  - Scaling and healing
- Deployment
  - Versioning and rollback
- Service
  - Static (non-ephemeral) IP addresses
  - Stable networking
- Persistent volumes
  - Non-ephemeral storage

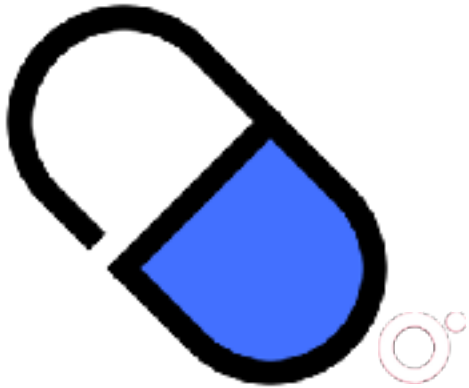


The ReplicaSet Object

Multiple identical pods which are replicas of each other

ReplicaSet

- If pod crashes, ReplicaSet will start a new one
- Key to fault-tolerance, healing, and scaling
- All pods are replicas of each other



The Deployment Object

Adds on deployment and rollback functionality

Deployment Objects

- Easy to push out new version of container
- Triggers creation of new ReplicaSet and new containers
- Pods in old ReplicaSet gradually reduced to zero
- Every change to a Deployment object creates a new revision
- Trivial to rollback to previous revision
- Offers versioning support

Ephemeral IP Addresses

- Containers expose ports in pod specifications
- Pod IP addresses are ephemeral
- Where should clients send requests?

Service Objects

Provide stable IP addresses for external connections and load balancing

Service Objects

- Provides stable (non-ephemeral) IP address
- Connects to set of back-end pods
- Set of pods changes dynamically
- Basic load balancing too

Storage with Containers

- On disk files within a container
  - Only accessible to the container itself
  - Ephemeral: is lost when the container stops or crashes
- Volume abstractions
  - A directory accessible to all containers in a pod
  - Have the same lifetime as the enclosing pod

For durable storage use persistent volumes

The volume is preserved even when the pod is removed and can be handed off to another pod

Workloads on Kubernetes

To deploy and manage containerized applications on the GKE the Kubernetes system creates controller objects i.e. higher level abstractions

Kubernetes abstractions also allow managing different kinds of workloads

Workloads on Kubernetes

- Stateless applications
  - Does not preserve state, saves no data to persistent disk
  - Deployed using the Deployment object
- Stateful applications
  - State is saved or persisted, uses persistent volumes
  - Deployed using the StatefulSet object
- Batch jobs
  - Finite, independent, parallel jobs
  - Deployed using the Job object
- Daemons
  - Ongoing, background tasks, run without intervention
  - Deployed using a DaemonSet

Containers

Which of the following statements regarding standalone containers are true?

1. They can automatically heal themselves
2. They can spawn new container to handle additional load
3. Higher level abstractions are needed for container clusters
4. Containers only contain your application code

Containers

Which of the following statements regarding standalone containers are true?

1. They can automatically heal themselves
2. They can spawn new container to handle additional load
3. **Higher level abstractions are needed for container clusters**
4. Containers only contain your application code

Containers

Which one refers to the machines that constitute the cluster and run the various Kubernetes services?

1.Containers
2.Nodes
3.Pods
4.ReplicaSets

Containers

Which one refers to the machines that constitute the cluster and run the various Kubernetes services?

1.Containers
2.Nodes
3.Pods
4.ReplicaSets

Containers

Which of the following abstractions offer autohealing and autoscaling in containers?

1.Pods
2.Nodes
3.Service
4.ReplicaSets

Containers

Which of the following abstractions offer autohealing and autoscaling in containers?

1.Pods
2.Nodes
3.Service
4.ReplicaSets

Containers

Which of the following abstractions offer versioning support and rollback?

1.Deployment
2.Pods
3.Service
4.ReplicaSets

Containers

Which of the following abstractions offer versioning support and rollback?

1.Deployment
2.Pods
3.Service
4.ReplicaSets

O'REILLY®

Session 7: Load Balancing

Compute

User Traffic

On an ordinary day

Backend Service

Can be serviced using a certain number of instances

What about special sale days, Single's Day or Big Billion Days?

Scalable Compute

User Traffic

Incoming requests from users during sale day

Backend Service

Group of instances to service those requests needs to scale

Managed Instance Groups

User Traffic

Incoming requests from users during sale day

Backend Service

Managed Instance Group

Managed Instance Groups are a horizontally scaled IaaS offering with **autohealing** and **autoscaling**

Managed Instance Group

Group of identical GCE VM instances, created from the same instance template that are managed by the platform

Scalable Compute with MIGs

User Traffic

Incoming requests from users during sale day

Backend Service

Managed Instance Group to serve those incoming requests

Need to Answer These Questions

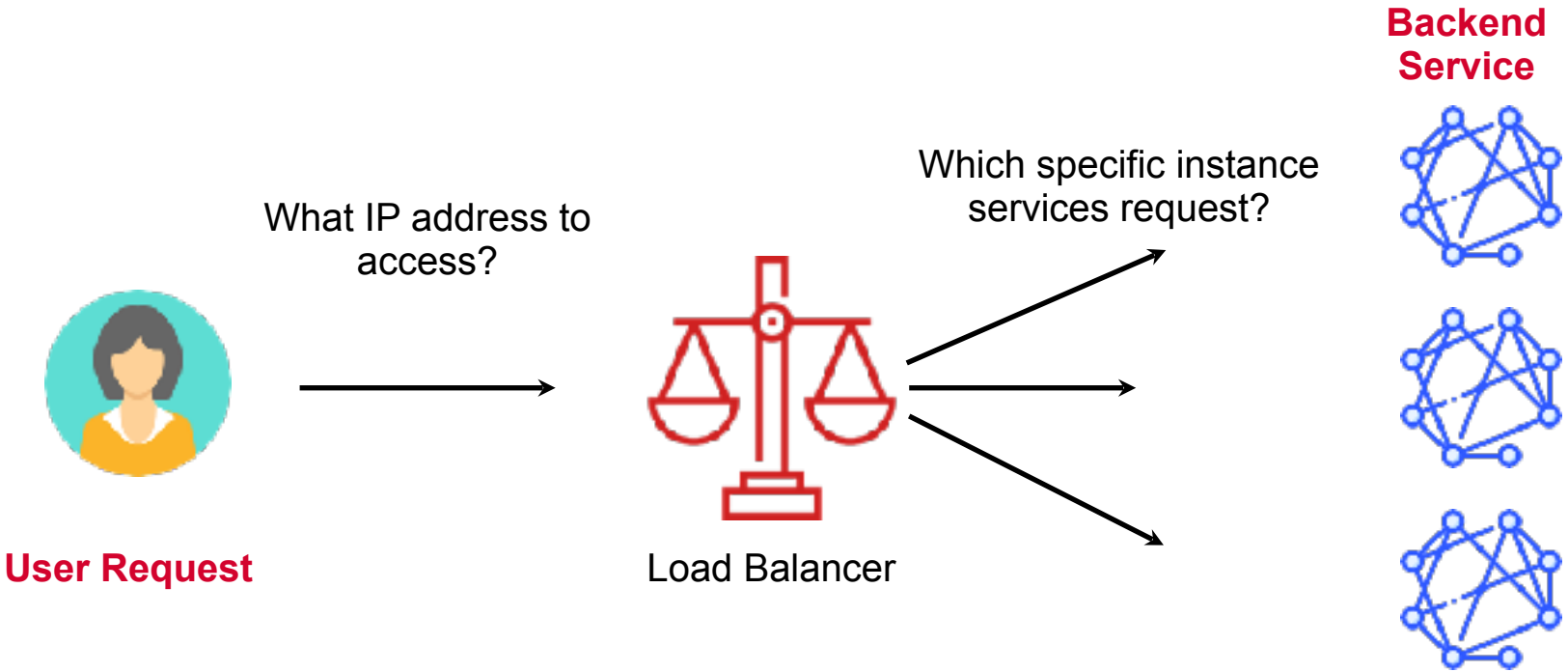
What IP Address?

Need a stable IP address to send traffic to - not ephemeral

Which specific instance?

Individual VMs may be terminated, restarted, overloaded

Load Balancers



Load Balancers

- Complex service
- Many moving parts
- Basic idea
  - Stable front-end IP
  - Forwarding rules to funnel traffic
  - Connect to backend service
  - Distribute load intelligently
  - Health checks to avoid unhealthy instances

Load balancers **distribute** traffic to resources close to users and meet **high-availability** requirements

Load Balancers on the GCP

- Fully managed, software-defined, redundant and highly available
- Supports > 1 million queries per second with high performance and low latency
- Autoscaling to meet increased traffic
- Route traffic to closest VM

Load balancers on the GCP can also work with **unmanaged instance groups** which offer **no** autoscaling and autohealing properties

Global Load Balancing

Use when your users and instances are globally distributed,
Provides IPv4 and IPv6 termination

Regional Load Balancing

Use when instances and users are concentrated in one region and only IPv4 termination is needed

External Load Balancing

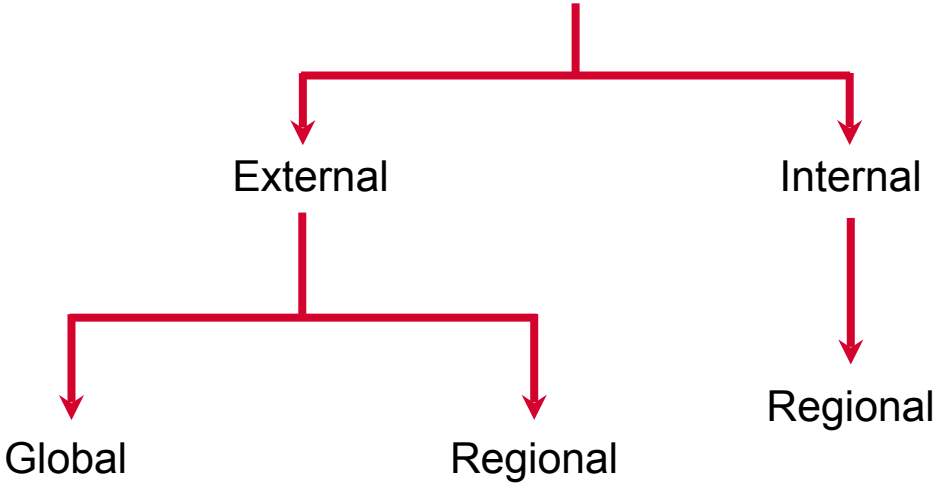
Distributes traffic from the internet to a GCP network

Internal Load Balancing

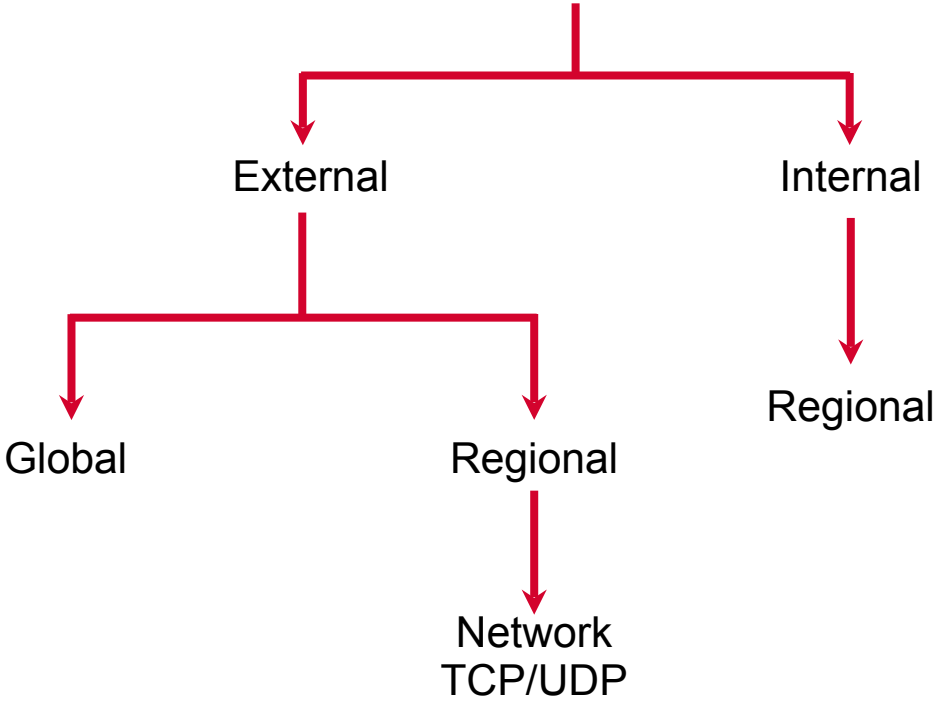
Distributes traffic only within a GCP network

Load Balancing

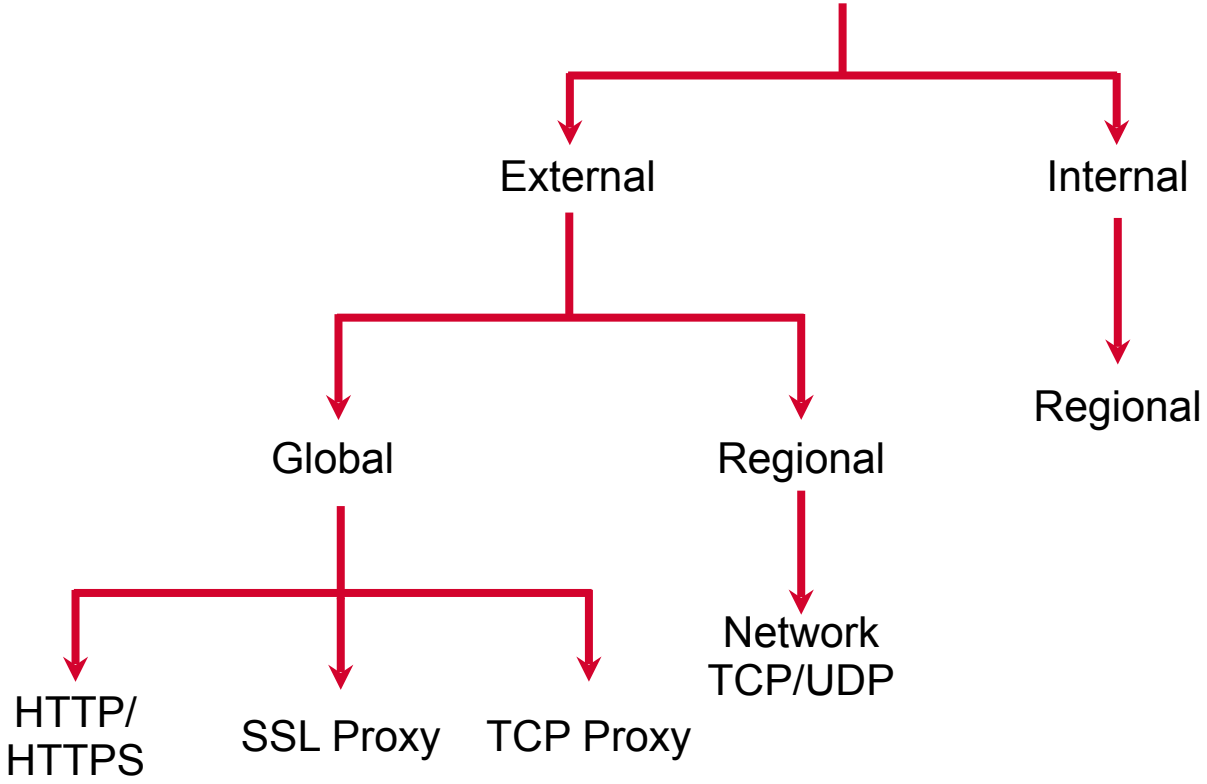
Load Balancing



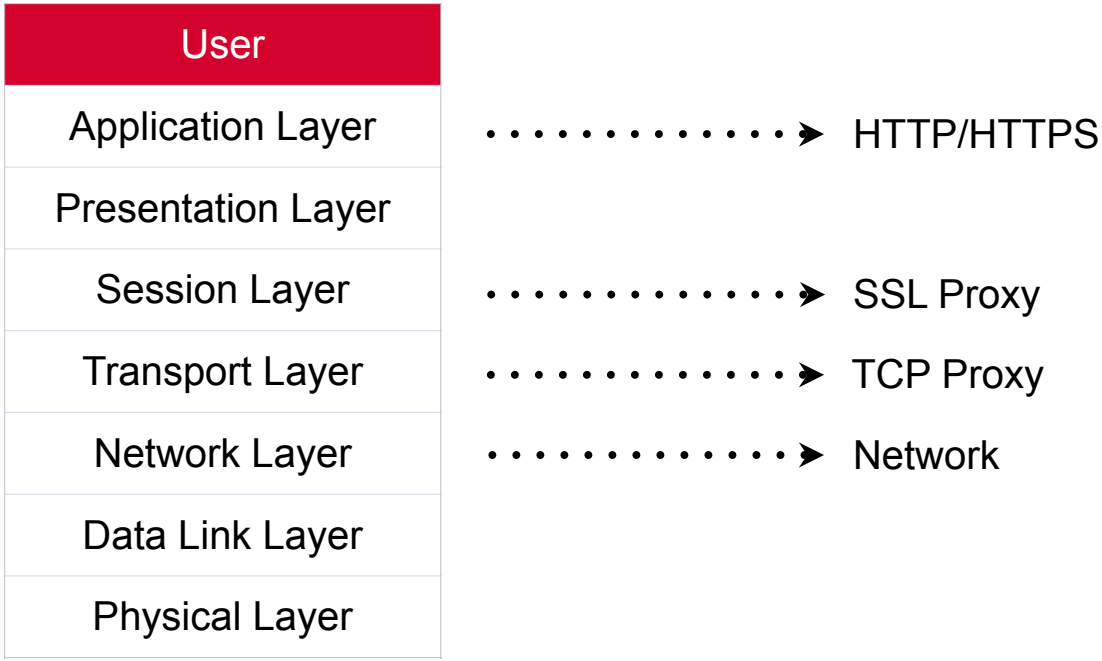
Load Balancing



Load Balancing



OSI Network Stack



Choose the load balancer at the highest layer in the OSI stack

HTTP(S) Load Balancing

- Distributes HTTP(S) traffic among groups of instances based on:
  - Proximity to the user
  - Requested URL
  - Or both.

SSL Proxy Load Balancing

- Use only for non-HTTP(S) SSL traffic
- For HTTP(S), just use HTTP(S) load balancing
- SSL connections are terminated at the global layer
- Then proxied to the closest available instance group

TCP Proxy Load Balancing

- Allows you to use a single IP address for all users around the world
- TCP connection terminated at the load balancing layer and proxied to closest instance group
- Automatically routes traffic to the instances that are closest to the user
- Better security, TCP vulnerabilities patched at the load balancer

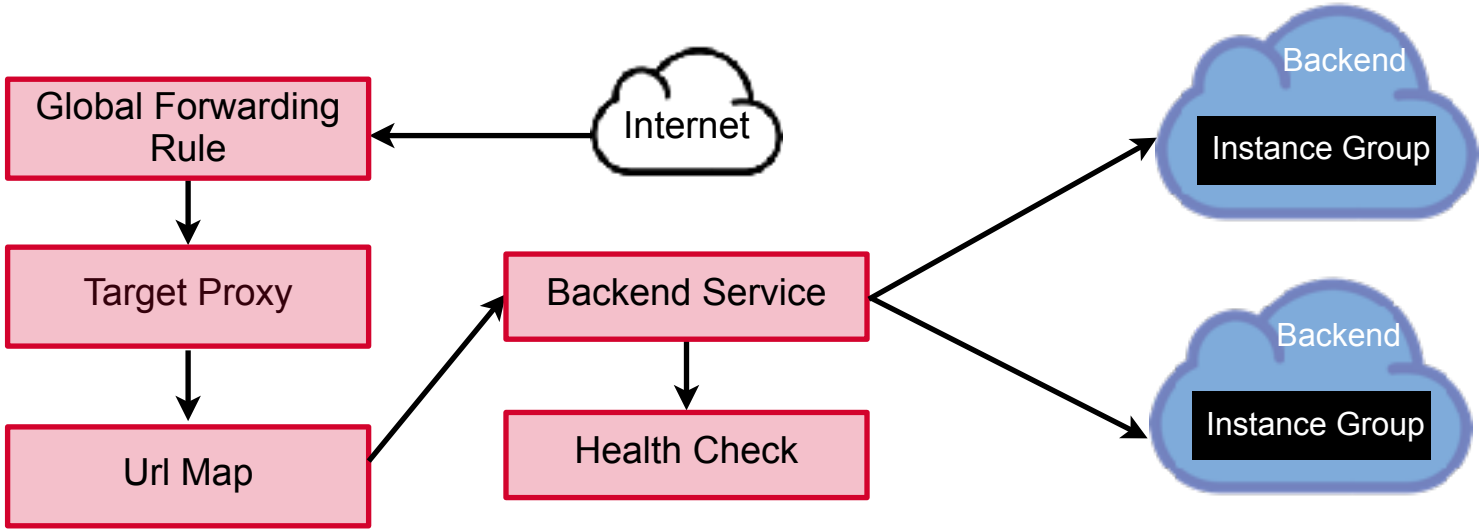
Network Load Balancing

- Based on incoming IP protocol data, such as address, port, and protocol type
- Pass-through, regional load balancer - does not proxy connections from clients
- Use it to load balance UDP traffic, and TCP and SSL traffic
- Load balances traffic on ports that are not supported by the SSL proxy and TCP proxy load balancers

Internal Load Balancing

- Private load balancing IP address that only your VPC instances can access
- VPC traffic stays internal - less latency, more security
- No public IP address needed
- Internal HTTP(S) and TCP/UDP load balancing
- Useful to balance requests from your frontend to your backend instances

HTTP(S) Load Balancing

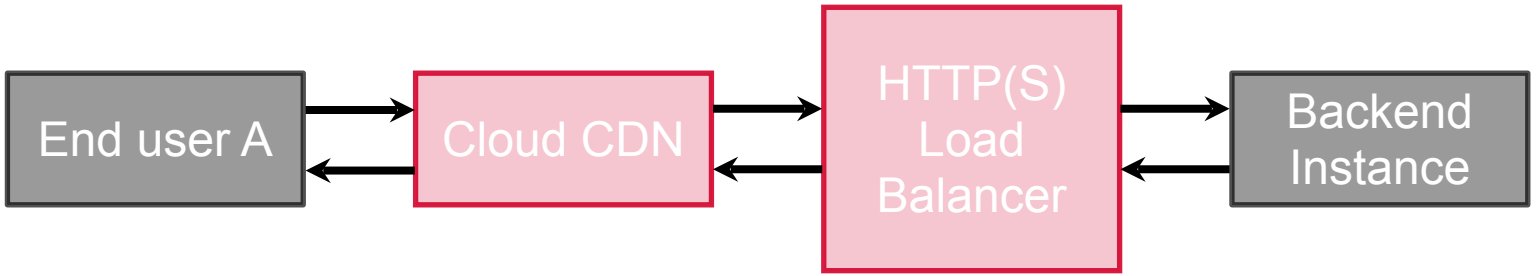


A global, external load balancing service offered on the GCP

Cloud CDN

Works with HTTP(S) load balancing to deliver content to users from numerous worldwide caches located close to users and at the edge of Google’s network

Cache Content Using Cloud CDN



The Cloud CDN will try and deliver content from the cache if content is present in the cache

Content will be cached on cache misses

Load Balancers

What sort of load balancer would you use for UDP traffic?

1.TCP
2.Internal
3.Network
4.HTTP(S)

Load Balancers

What sort of load balancer would you use for UDP traffic?

1.TCP
2.Internal
3.Network
4.HTTP(S)

Load Balancers

Which of the following load balancers would you use for distributing traffic from frontend instances to backend instances when both are on the GCP?

1.TCP
2.Internal
3.Network
4.HTTP(S)

Load Balancers

Which of the following load balancers would you use for distributing traffic from frontend instances to backend instances when both are on the GCP?

1.TCP
2.Internal
3.Network
4.HTTP(S)

Load Balancers

If your traffic was such that you had the choice to use any of these load balancers, which one would you choose?

1.TCP
2.SSL
3.Network
4.HTTP(S)

Load Balancers

If your traffic was such that you had the choice to use any of these load balancers, which one would you choose?

1.TCP
2.SSL
3.Network
4.HTTP(S)

Load Balancers

Which of the following best describes Cloud CDN?

1. Service that allows using GCS buckets behind a load balancer
2. Service that caches static content close to users
3. Service that improves network latency for a load balancer
4. Service that specifies forwarding rules for a load balancer

Load Balancers

Which of the following best describes Cloud CDN?

1.Service that allows using GCS buckets behind a load balancer
2.Service that caches static content close to users
3.Service that improves network latency for a load balancer
4.Service that specifies forwarding rules for a load balancer

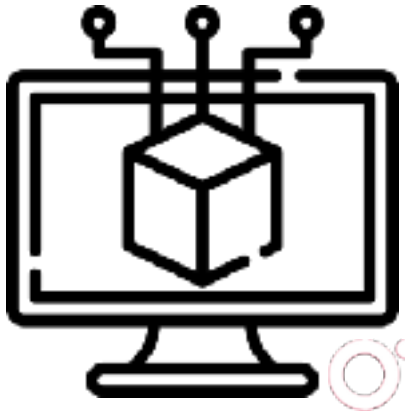
O'REILLY®

Session 8a: Managed Instance Groups

Month/Year

Cloud VM Instances

- The easiest compute option to begin with
- “Lift-and-shift” migration from on-premise data center
- However, two significant drawbacks - no autoscaling and no autohealing



Managed Instance Groups are a horizontally scaled IaaS offering with **autohealing** and **autoscaling**

Managed Instance Group

Group of identical GCE VM instances, created from the same instance template that are managed by the platform

Managed Instance Group

Group of identical GCE VM instances, created from the same instance template that are managed by the platform

Instances have the exact same configuration

Managed Instance Group

Group of identical GCE VM instances, created from the same instance template that are managed by the platform

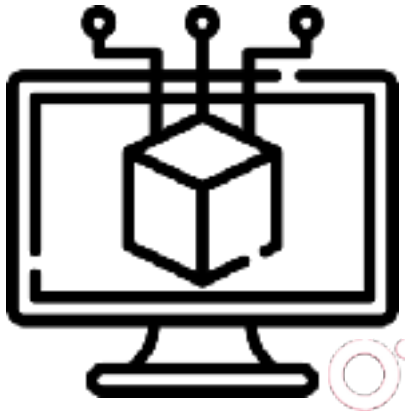
The configuration is specified in an instance template

Instance Template

A specification of machine type, boot disk (or container image), zone, labels and other instance properties that can be used to instantiate either individual VM instances or a Managed Instance Group

Features of MIGs

- Autoscaling policies
- Load balancing
- Identification and recreation of unhealthy instances
- Rolling updates

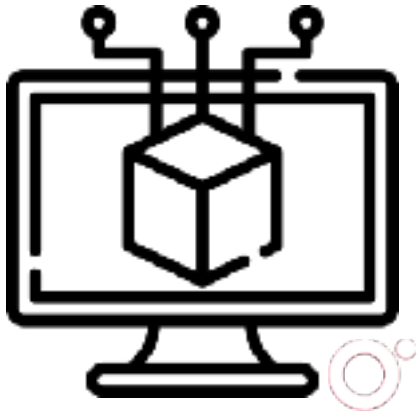


Unmanaged Instance Group

Dissimilar VM instances that are arbitrarily grouped together after-the-fact, usually for load balancing

Unmanaged Instance Groups

- Do not support
  - Autoscaling
  - Rolling updates
- Do support
  - Load balancing (primary use case)



Health Checks

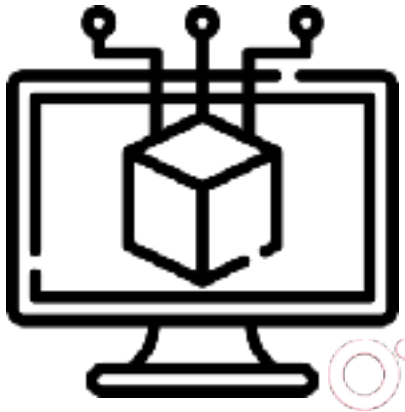
Health Check

Instance Template

Managed Instance Group

Health Checks

- If instances unhealthy, do not respond within time period
- Replace instance with new one



Autoscaling Policies

CPU Utilization

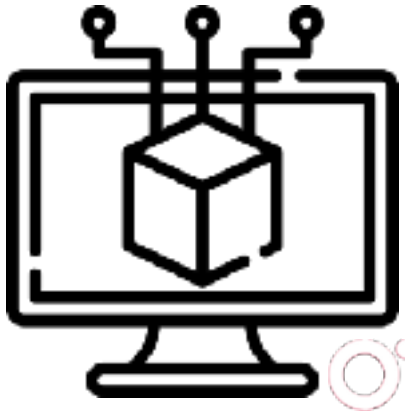
Requests/ Second

Instance Template

Managed Instance Group

Autoscaling Policies

- Check whether policy is being satisfied
- If more instances needed, add instances
- If fewer instances needed, remove instances



Session 8b: StackDriver, Deployment Manager, Apigee, Dataproc, Pubsub

Month/Year

Google Stackdriver

Suite of ops services providing monitoring, logging, debugging, error reporting, tracing, alerting and profiling. Integrates with several third-party tools

Stackdriver Suite

Monitoring	Logging	Trace
Error Reporting	Debugging	Profiling

Cloud Deployment Manager

Infrastructure deployment service that automates the creation and management of Google Cloud Platform resources

Cloud Deployment Manager

- Infrastructure-as-Code (IAC)
- Declarative format (YAML) for provisioning infrastructure
- Configuration as code
- Repeatable and scalable deployments

Apigee

A company that built a full lifecycle API management platform acquired by Google in 2016.

Apigee Edge

The API management platform built by Apigee, which allows developers to build and manage API proxies.

API Proxy

A program that sits in front of your API and proxies incoming user requests to the API and provides various value-added features.

Consuming APIs Directly

Client Apps

Mobile

PoS Devices

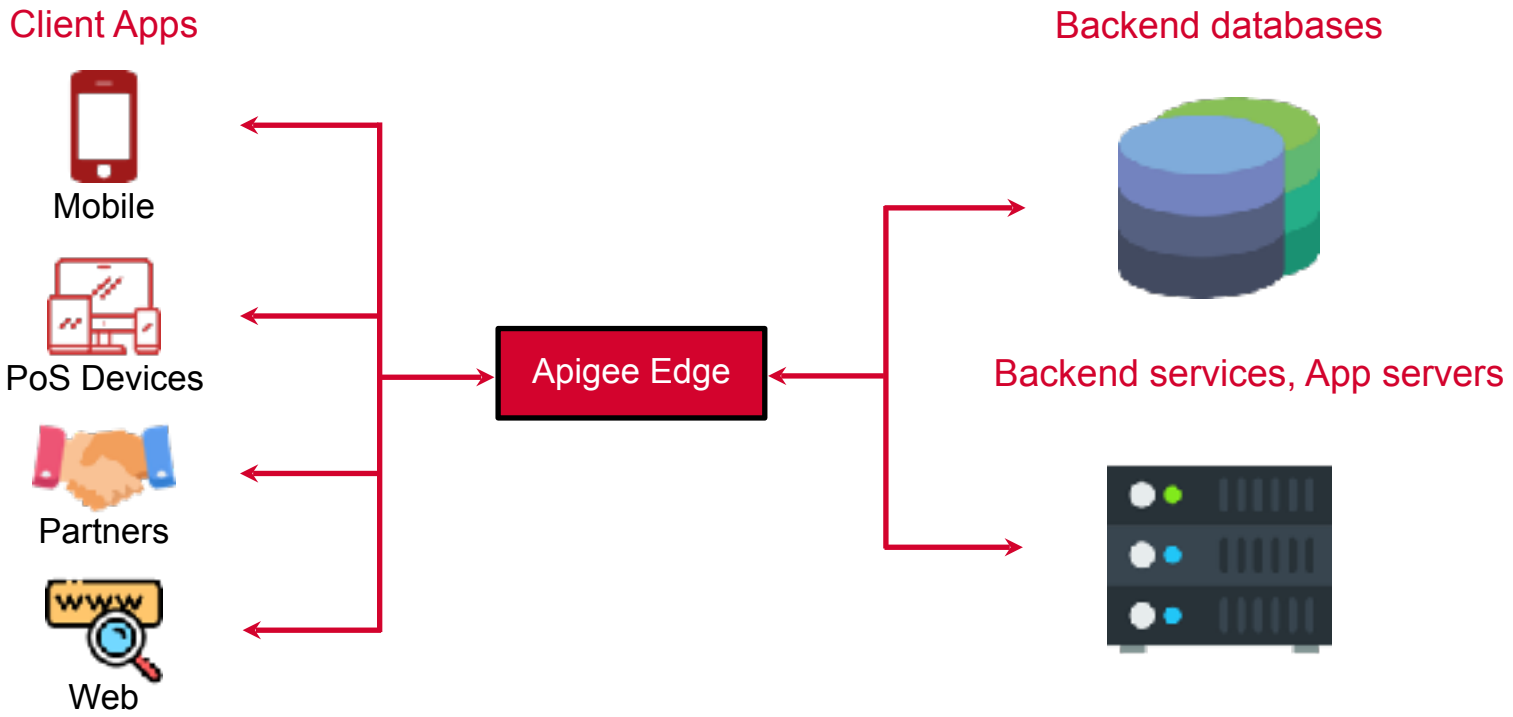
Partners

Web

Backend databases

Backend services, App servers

Apigee Edge



Decouple apps from APIs, control access to APIs

Apigee Use Cases

- Build API proxies easily
- Secure API calls
- Secure data
- Manage and throttle traffic
- Monetize smartly
- Set and enforce policies

Hadoop

- HDFS for storage
- MapReduce for compute on local machines
- YARN for co-ordination

Cloud Dataproc i.e. Managed Hadoop on the GCP

- Google Cloud Storage for storage
- MapReduce for compute on GCE VMs
- Dataproc service = Dataproc + YARN for co-ordination

Dataproc clusters should be created on the fly for **compute** - don’t use them for storage

Store data in Cloud Storage buckets which is cheap and does not need VMs provisioned and running

Hadoop vs. Dataproc

Hadoop

- Clusters always provisioned and running
- HDFS runs on cluster node
- Store data on cluster nodes
- Cluster is stateful

Dataproc

- Create clusters on the fly for compute requirements
- HDFS runs on persistent disks
- Store data in Cloud Storage
- Cluster is stateless

Pub/Sub

- Many-to-many asynchronous messages
- Decouples senders and receivers
- Publishers publish messages to a topic
- Subscribers listen or subscribe to topics
- Reliable, scalable delivery

O'REILLY®

Session 8c: Anthos

Month/Year

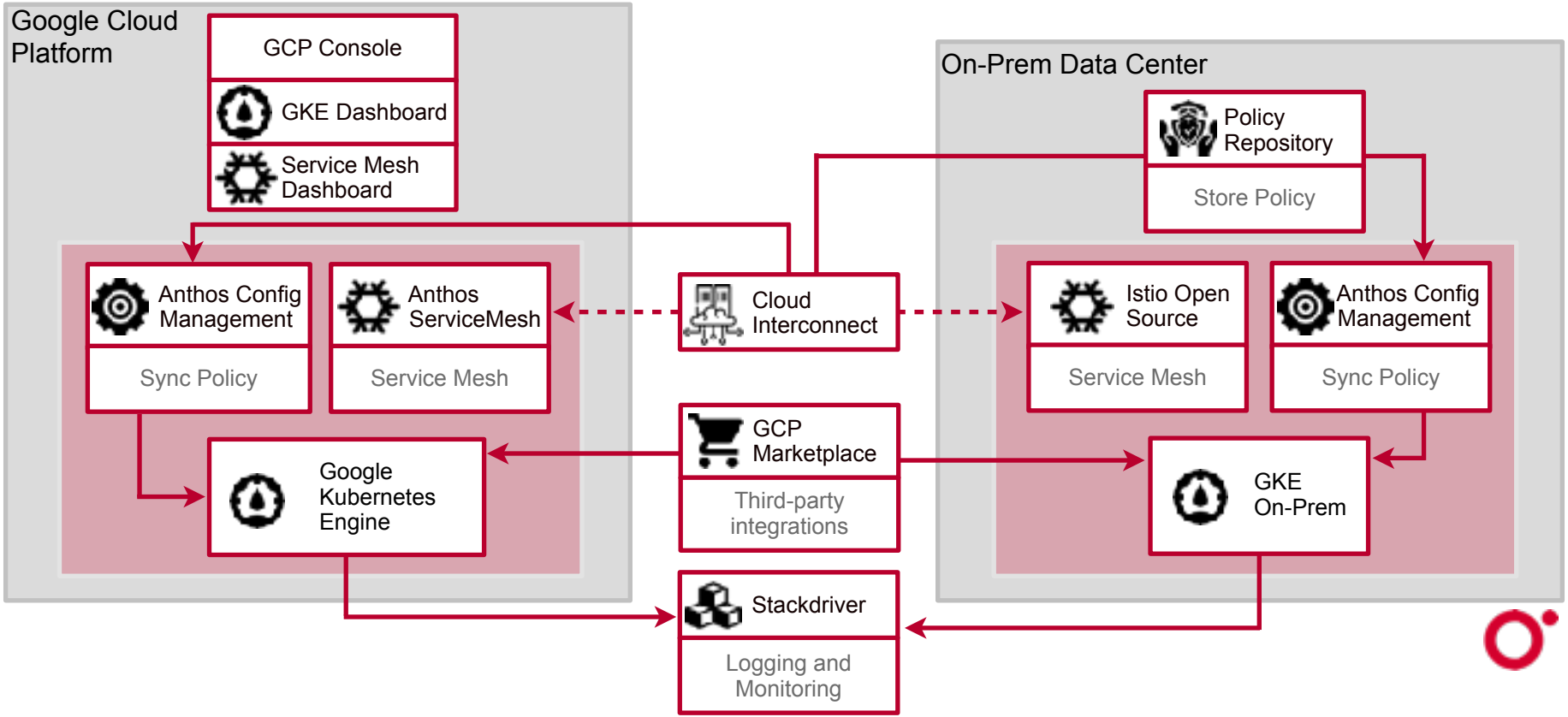
Anthos

A single open application platform to manage and run your applications across on-premises and cloud environments

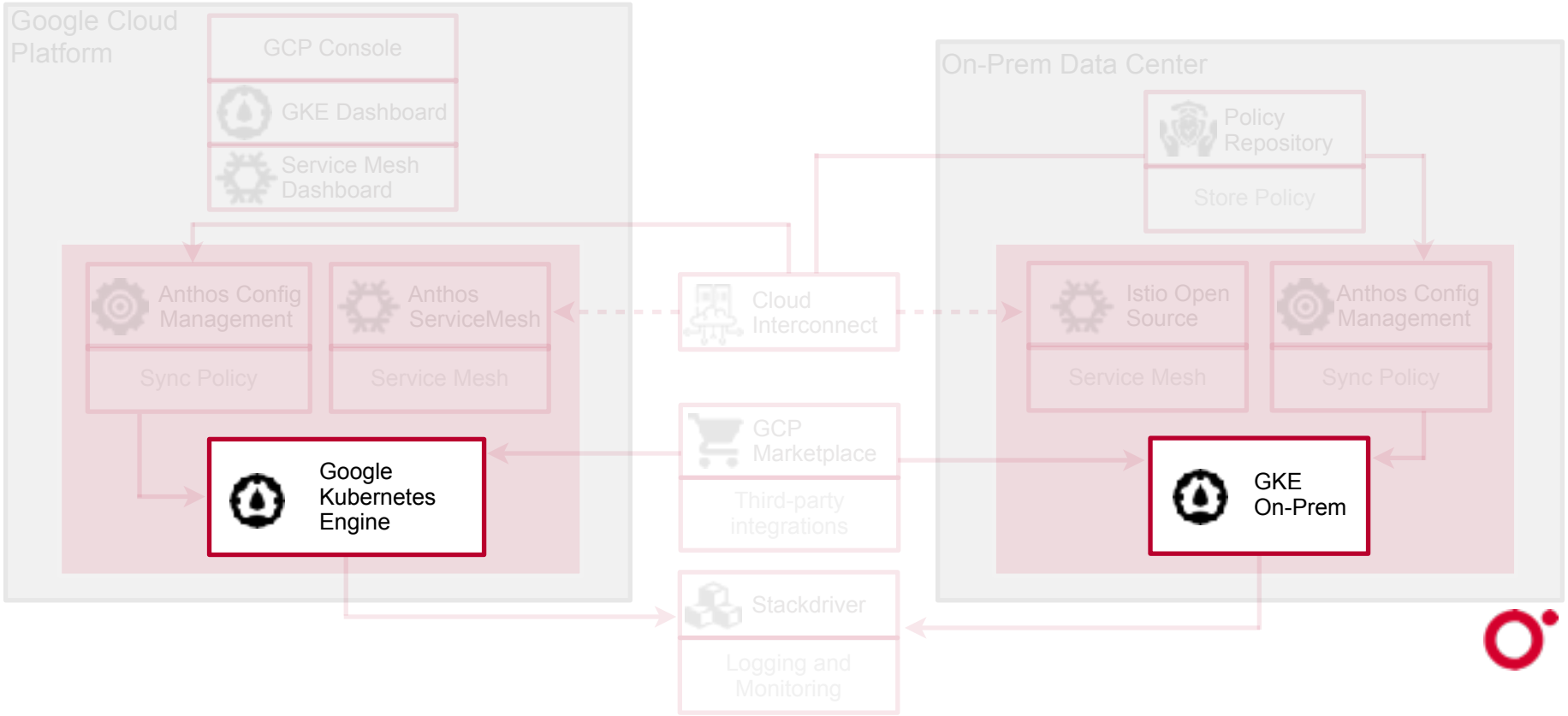
Anthos

Modernize applications, migrate workloads, apply policies and security at scale with a consistent experience across on-premises and cloud

Anthos Components



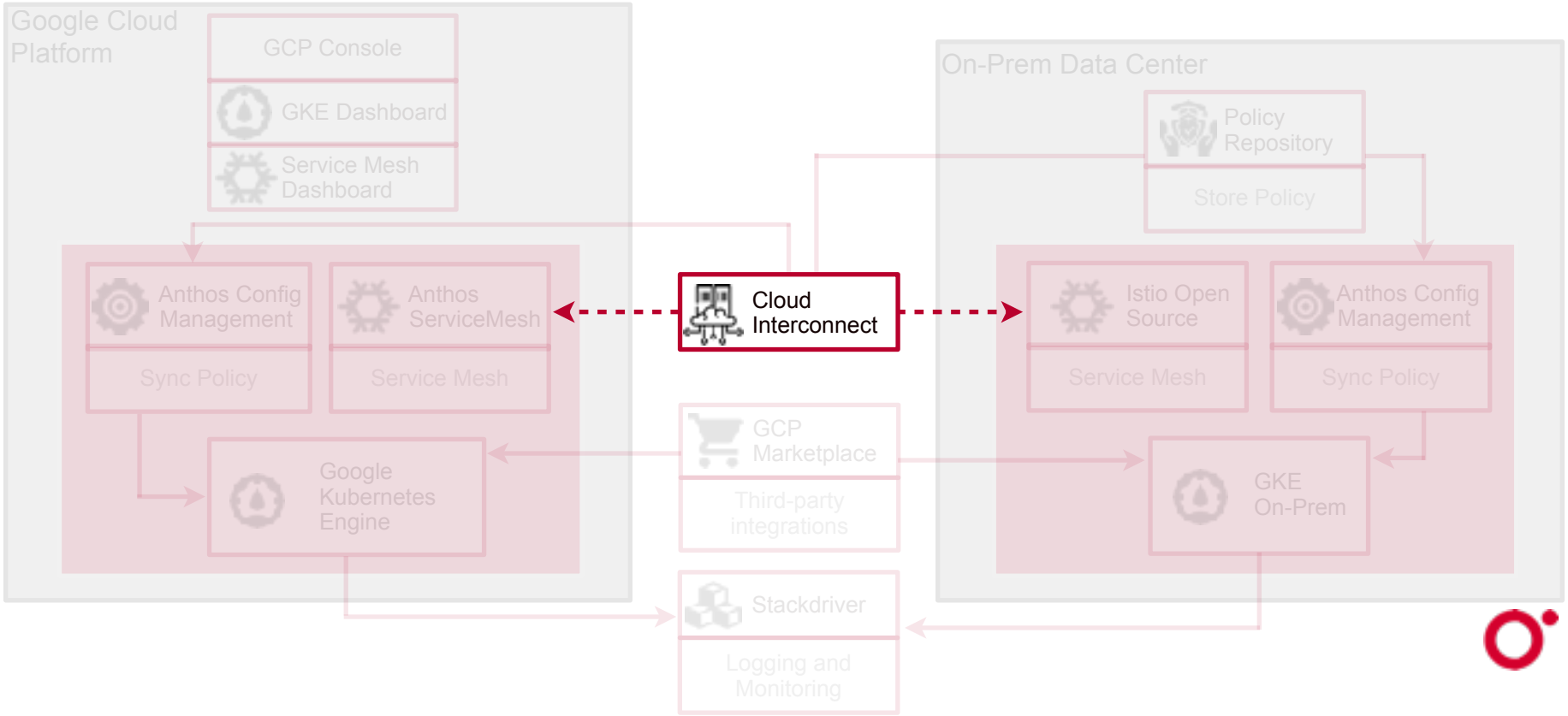
Anthos Components



Computing Environment

- Google Kubernetes Engine (GKE) and GKE On-Prem to manage installations
- Common orchestration layer no matter where your clusters and applications are located
- Manages application deployment, configuration, upgrade and scaling

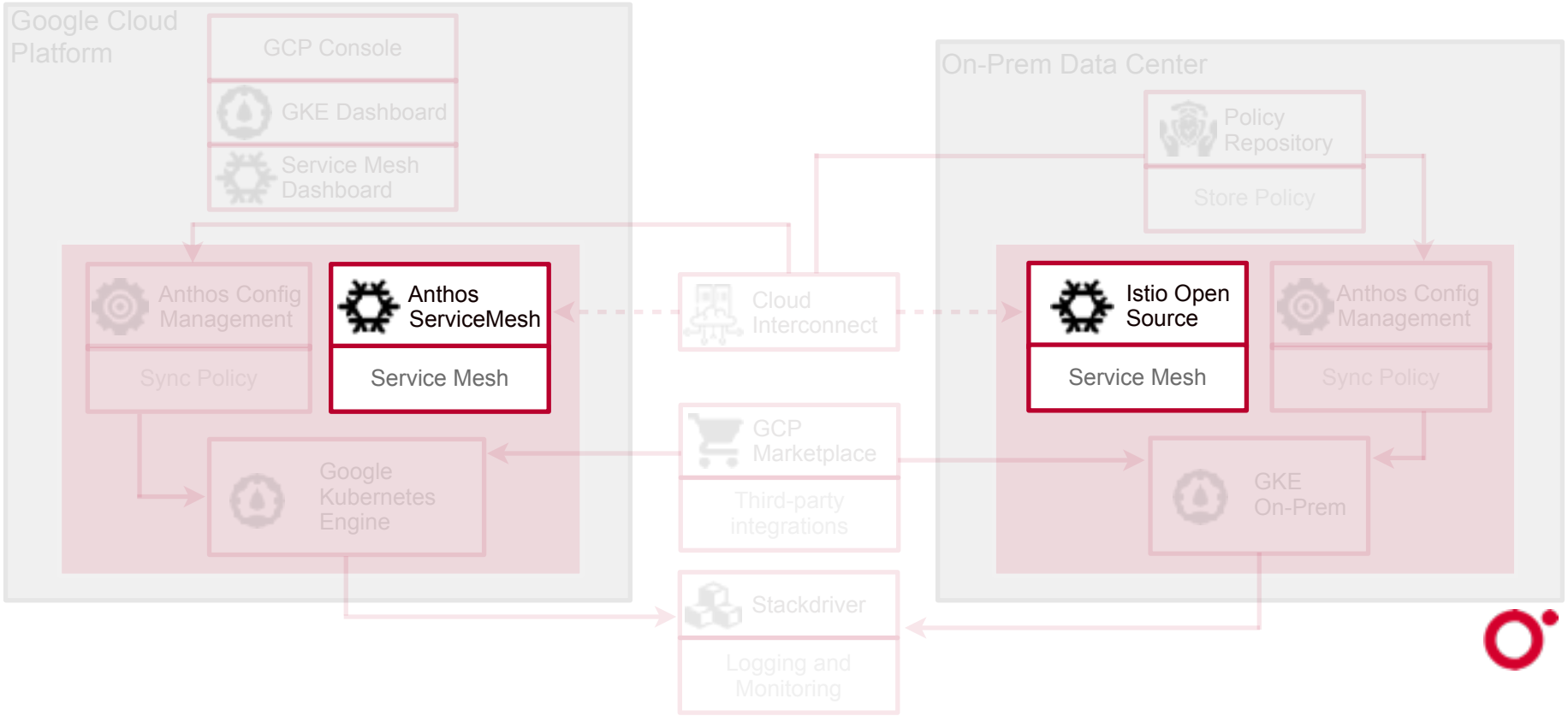
Anthos Components



Networking Environment

- Interconnect GCP and on-premises networks
- VPN tunnels using Cloud VPN on the GCP
- Dedicated and Partner interconnects for lower latency and high throughput

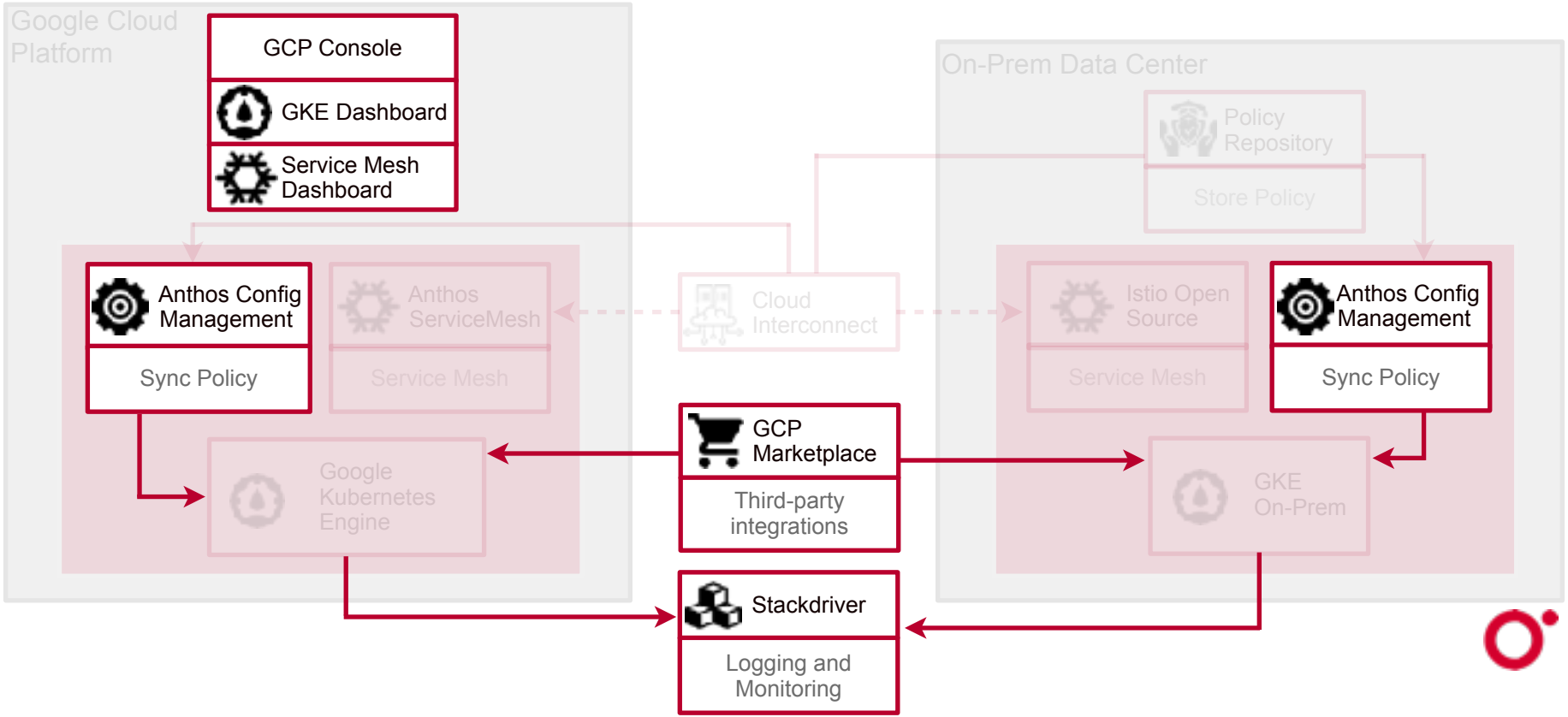
Anthos Components



Microservices Architecture

- Microservices architecture involve many services communicating over the network
- Service mesh model using the open-source implementation Istio
- Manages network inconsistencies by abstracting communication into a separate container in the same pod as the application
- Anthos Service Mesh to manage Istio + additional features
- Communication between services

Anthos Components



Other Components

- Centralized configuration management using configuration as code
- Consolidated logging and monitoring using Stackdriver
- Unified user interface for GCP and on-prem
- Third-party Kubernetes applications from the GCP marketplace e.g. storage solutions CI/CD tools