Compute Engine & Load Balancing FOR ARCHITECTS

Compute Engine & Load Balancing for Architects

It is not sufficient to get things working. We want more!

- Build Resiliency
- Increase Availability
- Increase Scalability
- Improve Performance
- Improve Security
- Lower Costs
- and .....

Professional Cloud Architect vs Associate Cloud Engineer

- **Associate Cloud Engineer**
  - Focused on tasks that Cloud Engineers perform in day to day job!
- **Professional Cloud Architect**
  - Understand business and technical requirements
  - Design cloud solutions that meet your functional and non-functional needs
- **GCP Services** are the same:
  - BUT **your perspective** should be different
  - With **Professional Cloud Architect**
    - You need to know the services
    - AND learn to build highly resilient, highly available, scalable, secure, performant solutions that have low cost!
  - Sounds Complex??
    - (Don't worry) We will understand each of these as we go further
      - Availability, Scalability, Resilience etc..

Google Cloud

What is Availability?

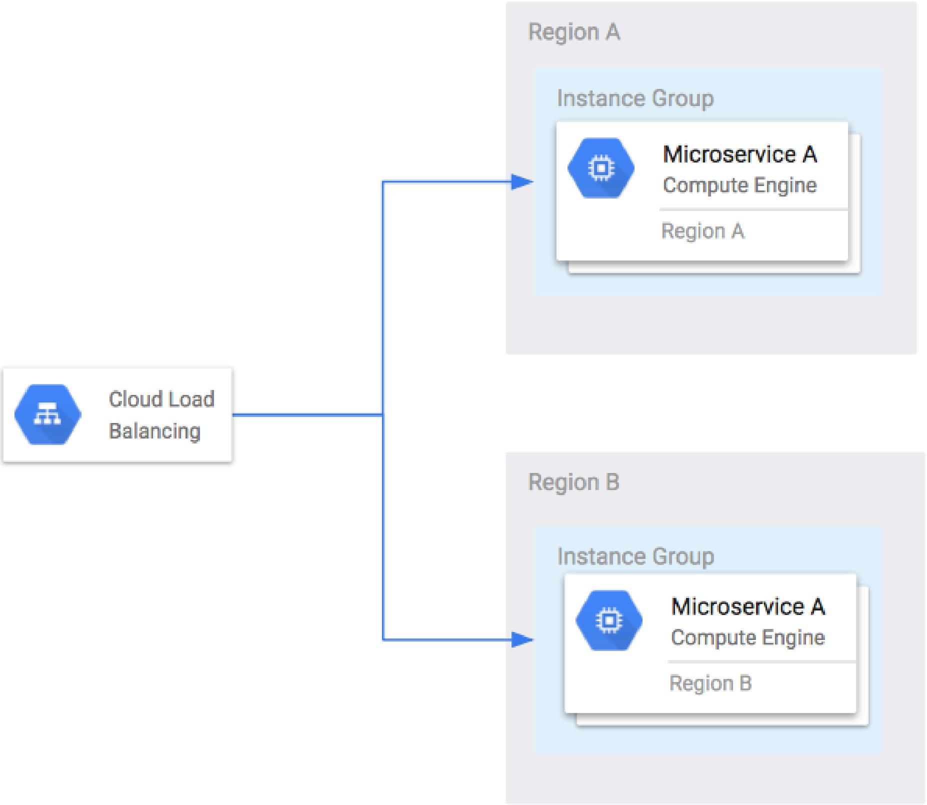
- Are the applications available **when the users need them**?
- **Percentage of time** an application provides the operations expected of it
- **Example**: 99.99% availability. Also called four 9's availability

Availability Table

Availability	Downtime (in a month)	Comment
99.95%	22 minutes	
99.99% (four 9's)	4 and 1/2 minutes	Most online apps aim for 99.99% (four 9's)
99.999% (five 9's)	26 seconds	Achieving 5 9's availability is tough

High Availability for Compute Engine & Load Balancing

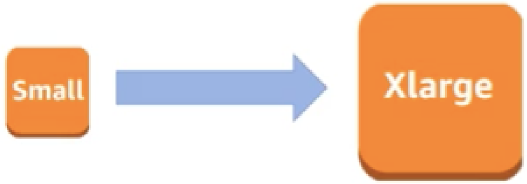
- **Highly Available Architecture:**
  - Multiple Regional Instance Groups for each Microservice
  - Distribute Load using a Global HTTPS Load Balancing
  - Configure Health Checks for Instance Group and Load Balancing
  - Enable Live Migration for VM instances
- **Advantages**:
  - Instances distributed across regions
    - Even if a region is down, your app is available
  - Global Load Balancing is highly available
  - Health checks ensure auto healing



What is Scalability?

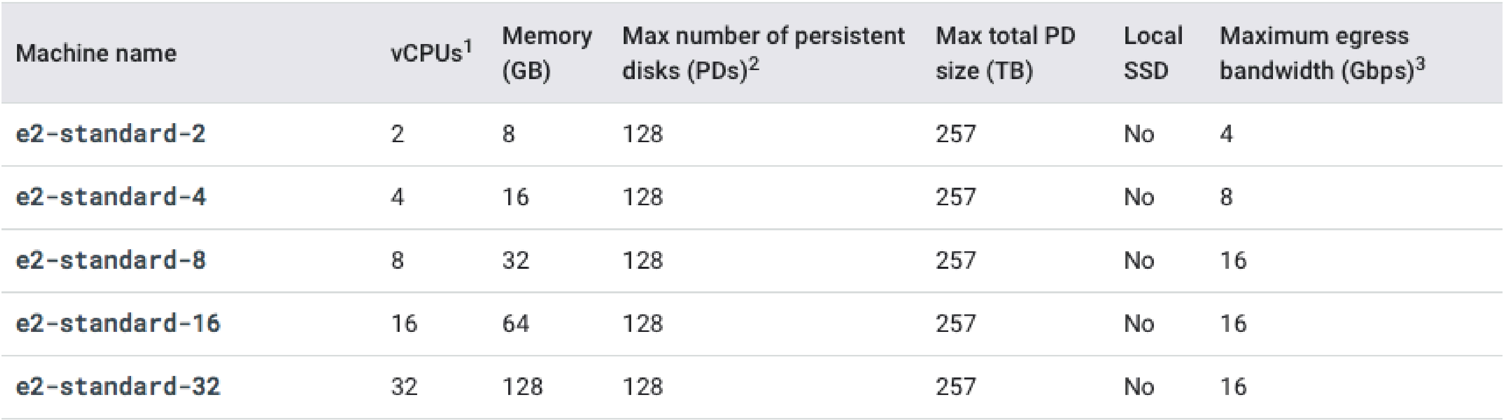
- A system is handling 1000 transactions per second. Load is expected to increase 10 times in the next month
  - Can we handle a **growth in users, traffic, or data size** without any drop in performance?
  - Does ability to serve more growth increase **proportionally** with resources?
- Ability to **adapt** to changes in demand (users, data)
- What are the options that can be considered?
  - Deploy to a bigger instance with bigger CPU and more memory
  - Increase the number of application instances and setup a load balancer
  - And a lot more.

What is Vertical Scaling?



- Deploying application/database to **bigger instance**:
  - A larger hard drive
  - A faster CPU
  - More RAM, CPU, I/O, or networking capabilities
- There are limits to vertical scaling

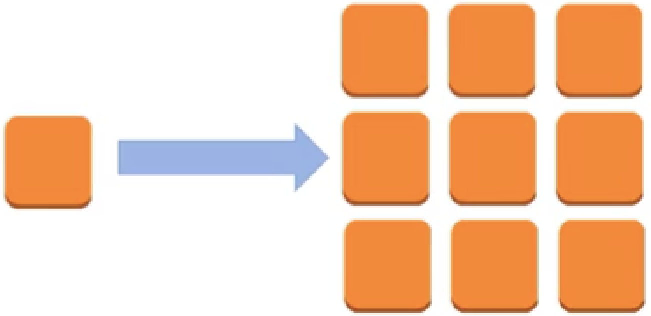
Vertical Scaling for GCE VMs



Machine name	vCPUs[1]	Memory (GB)	Max number of persistent disks (PDs)[2]	Max total PD size (TB)	Local SSD	Maximum egress bandwidth (Gbps)[3]
e2-standard-2	2	8	128	257	No	4
e2-standard-4	4	16	128	257	No	8
e2-standard-8	8	32	128	257	No	16
e2-standard-16	16	64	128	257	No	16
e2-standard-32	32	128	128	257	No	16

- Increasing **VM machine size:**
  - *e2-standard-2* to *e2-standard-4* or
  - *e2-standard-16* to *e2-standard-32* or
  - ...

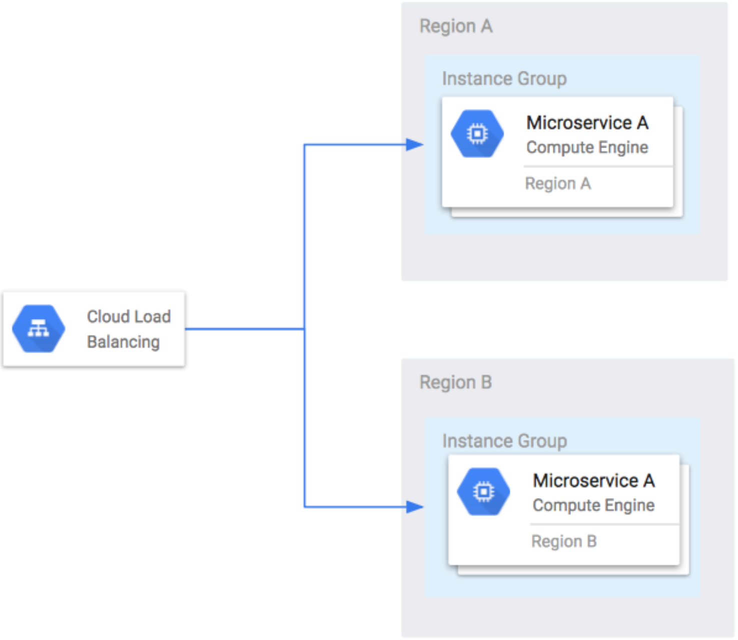
What is Horizontal Scaling?



- Deploying multiple instances of application/database
- (Typically but not always) Horizontal Scaling is preferred to Vertical Scaling:
  - Vertical scaling has limits
  - Vertical scaling can be expensive
  - Horizontal scaling increases availability
- (BUT) Horizontal Scaling needs additional infrastructure:
  - Load Balancers etc.

Horizontal Scaling for GCE VMs

- Distribute VM instances
  - in a single zone
  - in multiple zones in single region
  - in multiple zones across multiple regions
- **Auto scale:** Managed Instance Group (s)
- **Distribute load** : Load Balancing



Compute Engine : Live Migration & Availability Policy

- How do you keep your VM instances running when a host system needs to be updated (a software or a hardware update needs to be performed)?
- Live Migration
  - Your running instance is migrated to another host in the same zone
  - Does NOT change any attributes or properties of the VM
  - SUPPORTED for instances with local SSDs
  - NOT SUPPORTED for GPUs and preemptible instances
- Important Configuration - **Availability Policy:**
  - **On host maintenance**: What should happen during periodic infrastructure maintenance?
    - Migrate (default): Migrate VM instance to other hardware
    - Terminate: Stop the VM instance
  - **Automatic restart** - Restart VM instances if they are terminated due to non-user-initiated reasons (maintenance event, hardware failure etc.)

Compute Engine Features: GPUs

- How do you accelerate math intensive and graphics-intensive workloads for AI/ML etc?
- Add a **GPU** to your virtual machine:
  - High performance for math intensive and graphics-intensive workloads
  - Higher Cost
  - (REMEMBER) Use **images with GPU libraries** (Deep Learning) installed
    - OTHERWISE, GPU will not be used
  - **GPU restrictions**:
    - **NOT supported on all machine types** (For example, not supported on shared-core or memory-optimized machine types)
    - **On host maintenance** can only have the value "Terminate VM instance"
- Recommended **Availability policy** for GPUs
  - Automatic restart - on

GPU

Compute Engine & Load Balancing for Architects

Security

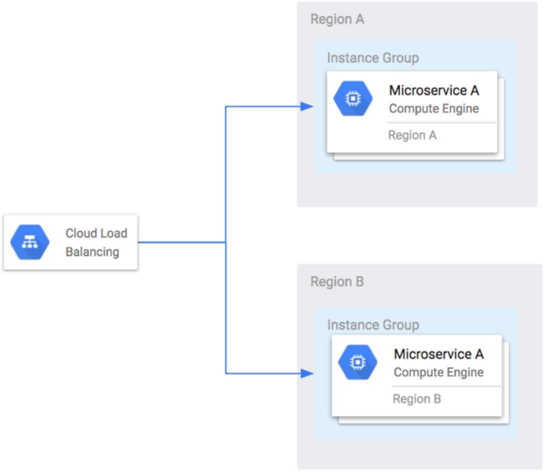
- Use **Firewall Rules** to restrict traffic
- Use **Internal IP Addresses** as much as possible
- Use **Sole-tenant nodes** when you have regulatory needs
- Create a hardened **custom image** to launch your VMs

Performance

- Choose right **Machine Family** for your workload
- Use GPUs and TPUs to increase performance
  - Use GPUs to accelerate machine learning and data processing workloads
  - Use TPUs for massive matrix operations performed in your machine learning workloads
- Prefer creating a hardened **custom image** to installing software at startup

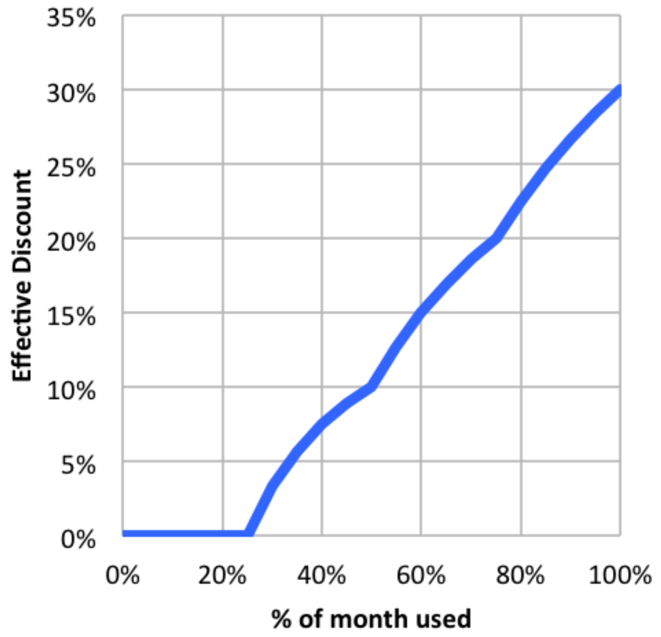
Resiliency for Compute Engine & Load Balancing

- **Resiliency** - "Ability of system to provide acceptable behavior even when one or more parts of the system fail"
- Build Resilient Architectures
  - Run VMs in MIG behind global load balancing
- Have the right data available
  - Use Cloud Monitoring for monitoring
  - Install logging agent to send logs to Cloud Logging
- Be prepared for the unexpected (and changes)
  - Enable Live Migration and Automatic restart when available
  - Configure the right **health checks**
  - **(Disaster recovery)** Upto date image copied to multiple regions
- We will talk about resiliency as we go further!



Sustained use discounts

- **Automatic discounts** for running VM instances for significant portion of the billing month
  - Example: If you use N1, N2 machine types for more than 25% of a month, you get a 20% to 50% discount on every incremental minute.
  - Discount increases with usage (graph)
  - No action required on your part!
- **Applicable** for instances created by **Google Kubernetes Engine** and **Compute Engine**
- **RESTRICTION**: Does NOT apply on certain machine types (example: E2 and A2)
- **RESTRICTION**: Does NOT apply to VMs created by App Engine flexible and Dataflow



Source: *https://cloud.google.com*

Committed use discounts

- For workloads with **predictable resource** needs
- **Commit** for 1 year or 3 years
- **Up to 70% discount** based on machine type and GPUs
- **Applicable** for instances created by **Google Kubernetes Engine** and **Compute Engine**
- **RESTRICTION**: Does NOT apply to VMs created by App Engine flexible and Dataflow

Compute Engine

Preemptible VM

Compute Engine

- **Short-lived cheaper** (upto 80%) compute instances
  - Can be stopped by GCP any time (preempted) within 24 hours
  - Instances get 30 second warning (to save anything they want to save)
- **Use Preempt VM's** if:
  - Your applications are **fault tolerant**
  - You are very **cost sensitive**
  - Your workload is **NOT immediate**
  - Example: Non immediate batch processing jobs
- **RESTRICTIONS:**
  - NOT always available
  - NO SLA and CANNOT be migrated to regular VMs
  - NO Automatic Restarts
  - Free Tier credits not applicable

Spot VMs

- **Spot VMs**: Latest version of preemptible VMs
- **Key Difference:** Does not have a maximum runtime
  - Compared to traditional preemptible VMs which have a maximum runtime of 24 hours
- **Other features similar to traditional preemptible VMs**
  - May be reclaimed at any time with 30-second notice
  - NOT always available
  - Dynamic Pricing: 60 - 91% discount compared to on-demand VMs
  - Free Tier credits not applicable

Compute Engine

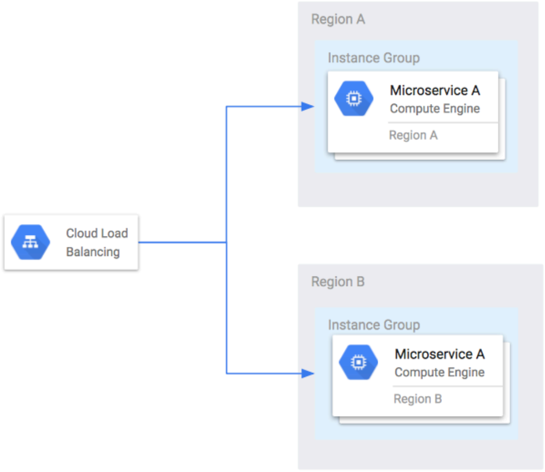
Google Compute Engine - Billing

Compute Engine

- You are **billed by the second** (after a minimum of 1 minute)
- You are NOT billed for compute when a compute instance is stopped
  - However, you will be billed for any storage attached with it!
- (RECOMMENDATION) **Always create Budget alerts** and make use of Budget exports to stay on top of billing!
- What are the ways you can save money?
  - Choose the right machine type and image for your workload
  - Be aware of the discounts available:
    - Sustained use discounts
    - Committed use discounts
    - Discounts for preemptible VM instances

Compute Engine & Load Balancing - Cost Efficiency

- Use Auto Scaling
  - Have optimal **number and type** of VM instances running
- Understand Sustained use discounts
- Make use of Committed use discounts for predictable long term workloads
- Use Preemptible VMs for non critical fault tolerant workloads


Region A
Instance Group
Microservice A
Compute Engine
Region A
Cloud Load Balancing
Region B
Instance Group
Microservice A
Compute Engine
Region B