Google Cloud Platform

Virtual Machines

v 1.0

Instance, VM instance, and VM (virtual machine) are used interchangeably in this class.
Many of the command line utilities refer to *instances*. VM is simpler.

Agenda

1. Google Compute Engine (GCE)
2. Lab #1
3. Compute options (vCPU and Memory)
4. Images
5. Disk Options
6. Lab #2
7. Common GCE actions
8. Review

GCP Compute & Processing Options

	Compute Engine	Container Engine	App Engine Standard	App Engine Flexible	Cloud Functions
Language support	Any	Any	Java 7 Python 2.7 Go PHP	Java 8 Python 2.7/3.5 Go Node.js Ruby Custom Runtimes	Triggers
Usage model	IaaS	IaaS PaaS	PaaS	PaaS	Microservices Architecture
Scaling	Server Autoscaling	Cluster	Autoscaling managed servers		Serverless
Primary use case	General Workloads	Container Workloads	Scalable web applications Mobile backend applications		Lightweight Event Actions

SLA for GCE: https://cloud.google.com/compute/sla

Google Compute Engine

- Infrastructure as a Service (IaaS)
- Predefined or Custom machine types
  - vCPU's (cores) and Memory (RAM)
  - Persistent disks - HDD, SSD, and Local SSD
  - Networking
  - Linux or Windows

Google Compute Engine

Google Compute Engine lets you create and run virtual machines on Google infrastructure. Compute Engine offers scale, performance, and value that allows you to easily launch large compute clusters on Google's infrastructure. There are no upfront investments and you can run thousands of virtual CPUs on a system that has been designed to be fast, and to offer strong consistency of performance.

An instance is a virtual machine (VM) hosted on Google's infrastructure. You can create an instance by using the Google Cloud Platform Console or the gcloud command-line tool. A Compute Engine instances can run Linux and Windows Server images provided by Google, or any customized versions of these images. You can also build and run images of other operating systems. You can choose the machine properties of your instances, such as the number of virtual CPUs and the amount of memory, by using a set of predefined machine types or by creating your own custom machine types.

GCE Features

- Instance metadata
- Startup scripts

Machine rightsizing
- Recommendation engine for optimum machine size
- Stackdriver statistics
- New recommendation 24 hrs after VM create or resize

Global load balancing:
- Multiple regions for availability

Availability policies:
- Live migrate
- Auto restart

- Per-minute billing
- Sustained use discounts

Preemptible:
- Up to 80% discount
- No SLA

If the VM availability policy is set to the default, live migrate, during regular system maintenance your VM will be migrated to different hardware so there is no downtime due to maintenance activities. Auto restart refers to what behavior the VM should take after a hardware failure or a system event. If marked auto restart, the system will try to launch a replacement VM. Auto restart does not restart the VM if it was terminated due to a user event, such as shutting down and terminating the VM.

Recommendation is designed for minimizing expense by optimizing machine type
You may have other goals:

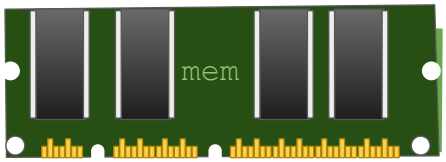
- Run overcapacity, for example, to handle spike in traffic
- Run overcapacity for resilliency
- Run undercapacity for guaranteed utilization, queue up work
- Run undercapacity to cap spend

https://cloud.google.com/compute/docs/instances/viewing-sizing-recommendations-for-instances

Compute

- High CPU, high memory, standard and shared-core machine types
  - Network throughput scales at 2Gb per vCPU
  - Max throughput of 16 Gb or 8 vCPU
- A vCPU is equal to 1 hyperthreaded core
- 2 vCPU is equal to 1 physical core

TAS



TAS

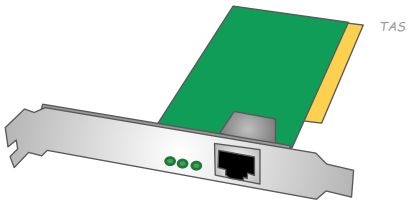
3 Machine types

- Higher proportion of Memory to CPU
- Higher proportion of CPU to memory
- A blend of both in our standard configuration

Software defined networking allows you to scale network and disk I/O performance. 2 Gb of throughput per vCPU up to 16 Gb. This also shares throughput with disk throughput.

Networking

- Robust networking features
  - Default, custom networks
  - Inbound/Outbound Firewall rules
    - IP based
    - Instance / group tags
  - Regional HTTP(s) load balancing
  - Network load balancing
    - Does not require pre-warming
  - Global and multi-regional Subnetworks



Your networks connect your instances to each other and to the Internet. You can segment your networks, use firewall rules to restrict access to instances, and create static routes to forward traffic to specific destinations.

Scale your applications on Google Compute Engine from zero to full-throttle with Google Cloud Load Balancing, with no pre-warming needed. Distribute your load-balanced compute resources in single or multiple regions, close to your users and to meet your high availability requirements. Cloud Load Balancing can put your resources behind a single anycast IP and scale your resources up or down with intelligent autoscaling. Cloud Load Balancing comes in a variety of flavors and is integrated with Google Cloud CDN for optimal application and content delivery.

Subnetworks segments your Cloud network IP space into subnetworks. Subnetwork prefixes can be automatically allocated, or you can create a custom topology. For more information on subnetworks, see: https://cloud.google.com/compute/docs/subnetworks.

Storage

- Persistent disks
  - Standard, SSD, or local SSD
  - Standard and SSD PD's scale in performance for each GB of space allocated
  - https://cloud.google.com/compute/docs/disks/performance
- Resize disks, migrate instances with no downtime

- Software defined networking allows standard and SSD PD to scale per GB allocated up to their maximum allowed I/O defined at https://cloud.google.com/compute/docs/disks/performance
- Local SSD is only allocated in fixed pre-defined disk sizes
- Live migration of disk only applies to Standard and SSD PD. Local SSD is considered ephemeral and should not be relied upon to be available during a live migration

Pricing

- Per-minute billing, sustained use discounts
  - 10 minute minimum
- Preemptible instances
  - Live at most 24 hours
  - Can be pre-empted with a 30 second notification via API
  - Up to 80% discount
- Custom machine types
  - Customize amount of memory and CPU
- Recommendation Engine
  - Notifies you of under utilized instances

For more info on preemptible instances, go to:
https://cloud.google.com/compute/docs/instances/preemptible.
For more information on using custom machine types, see:
https://cloud.google.com/compute/docs/instances/creating-instance-with-custom-machine-type.

VM charges and discounts

- All machines are billed for a minimum of 10 minutes
- Per-minute charge, rounded up to nearest minute
- Lower price for *preemptible* instances
- Scaled discounts for sustained use

Predefined-type sustained use discount

% of month usage	Percent of base rate
0%-25%	100%
25%-50%	80%
50%-75%	60%
75%-100%	40%

Inferred instances

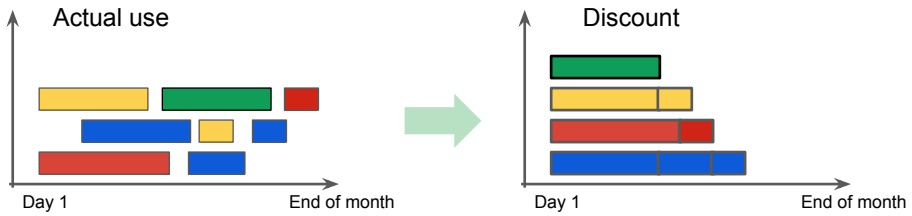
Custom-type sustained use discount

% of usage	Percent of base rate
First 25%	100%
Next 25%	80%
Last 25%	60%

Combined resources

All machines are charged for 10 minutes at boot time. This is the minimum charge for a VM. After that, per-minute pricing begins. And for any part of a minute, the charge is rounded up to the nearest minute.
Discounts are complicated. Predefined machine types are discounted based on the percent of monthly use. And custom-type is discounted on a percent of total use. *Inferred instances* means that for billing purposes, the same type of machine used in the same zone will be combined into a single charge so that you get the most discount -- as if it were one machine in use the whole time. And *combined resources* means that memory and vCPU of the same type are combined so that you get the discount on the greatest resource consumption in custom-types.

Inferred instances discount



Usage of VMs of the same machine type in the same zone are combined as if they were one machine to give you the best discount

Combined resources means that memory and vCPU of the same type are combined so that you get the discount on the greatest resource consumption in custom-types.

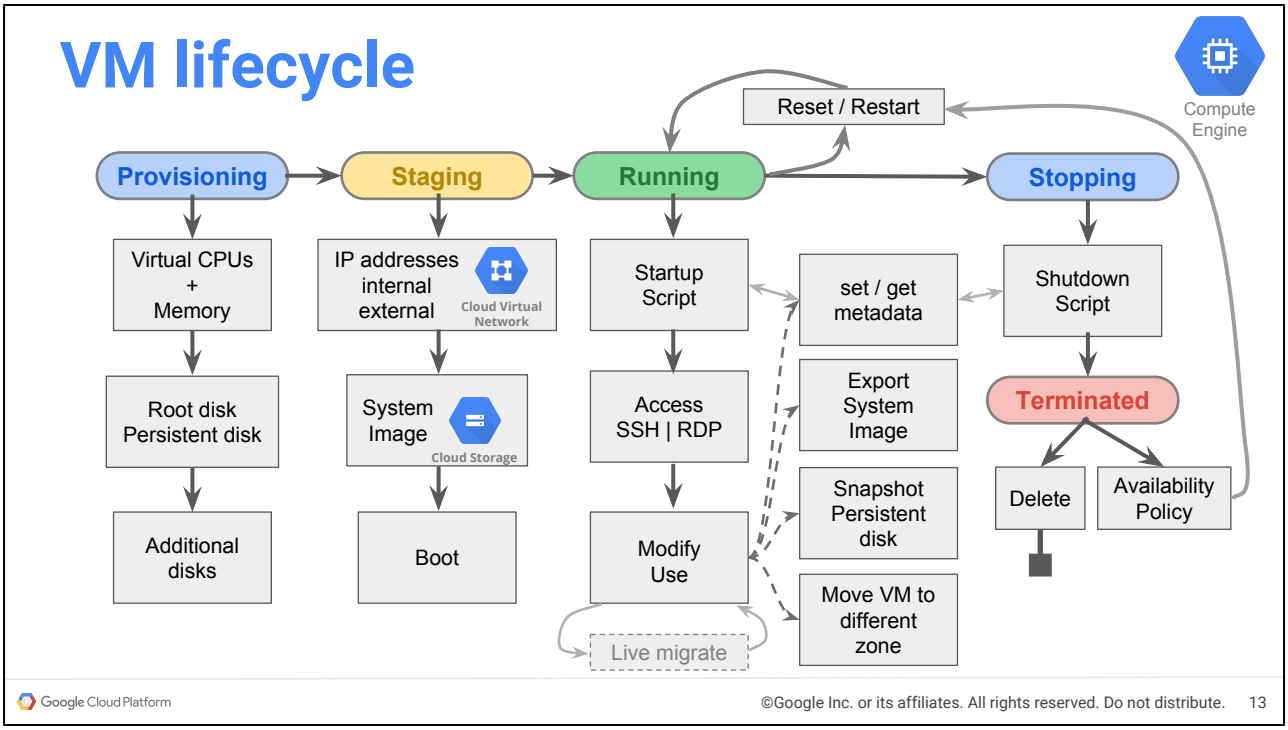
A new discount called Committed Use Discounts was announced in March of 2017. Customer can received up to 57% discount in exchange for a one or three year commitment paid monthly with no up-front costs.
https://cloud.google.com/compute/docs/instances/signing-up-committed-use-discounts

VM access

- Linux: SSH
  - SSH from console, SSH from CloudShell via Cloud SDK
  - SSH from computer, 3rd Party client + generate key pair
  - Requires Firewall rule to allow tcp:22
- Windows: RDP
  - Requires setting the Windows password
  - Requires Firewall rule to allow tcp:3389
  - RDP clients
    - Chrome extension, 3rd party apps, MS Windows RDP client
    - Linux freerdp, remmina
  - Powershell terminal

Owners/Editors can use console, CloudShell, or SDK gcloud compute. Usera that are not members can use 3rd party SSH.
In addition to SSH, you can protect access to instances with external IP addresses using port forwarding and SOCKS proxies. For more information, see: https://cloud.google.com/solutions/connecting-securely#https-and-ssl

Some RDP clients are incompatible because they don't support Network Level Authentication (NLA). Example: rdesktop
Microsoft RDP client for OSX 10.9 or earlier is incompatible with. Upgrade to Microsoft RDP client for Mac version 8.0.14.



Create a new VM from console, CloudShell gcloud command, or API.

The rest of this module covers the subjects on this slide.

- PROVISIONING - Resources being reserved for the instance
- STAGING - Resources acquired, instance being prepared
- RUNNING - Instance is booting up or running
- STOPPING - Instance being stopped due to failure or shut down
  - Temporary status; instance will move to TERMINATED
  - When documentation says "STOPPED" it means "TERMINATED" state.
- TERMINATED - Instance shut down or encountered failure (through API or inside the guest)
- Check instance state in gcloud using:
  - `gcloud compute instances describe <instance>`

Changing VM state from running

	methods	Shutdown Script time	state
`reset`	console, `gcloud`, API, OS	no	remains running
`restart`	console, `gcloud`, API, OS	~90 sec	stop->running
`reboot`	OS: `sudo reboot`	~90 sec	stop->running
`stop`	console, `gcloud`, API	~90 sec	terminated
`shutdown`	OS: `sudo shutdown`	~90 sec	terminated
`delete`	console, `gcloud`, API	~90 sec	N/A
preemption	*automatic*	~30 sec	N/A

"ACPI Power Off"

Stopping an instance moves it into a Terminated state. So "stopped" and "terminated" are the same state.
Stopping sends and Advanced Configuration Power Interface (ACPI) "ACPI Power Off" command.
Script normally has about 90 seconds to run, preempted VMs have 30 seconds.
Running and time allowed for shutdown script is not guaranteed.
When a VM is stopped, it loses it's ephemeral external IPs and the contents of RAM memory.
Terminated VMs still exist and can be restarted until they are deleted.

Availability policy - automatic changes

Called "scheduling options" in SDK/API

- *Automatic restart*
  - Automatic VM restart due to crash or maintenance event
    - Not preemption or a user-initiated terminate
- *On host maintenance*
  - Determines whether host is live migrated or terminated due to a maintenance event. Live migration is the default
- Live migration
  - During maintenance event, VM is migrated to different hardware without interruption
  - metadata indicates occurrence of live migration

Automatic restart due to maintenance event, hardware failure, or software failure.

Stopped (Terminated) VM

- No charge for stopped VM
  - Still charged for attached disks and IPs
- Actions
  - Change the machine type
  - Add or remove attached disks, change auto-delete settings
  - Modify instance tags
  - Modify custom VM or project-wide metadata
  - Remove or set a new static IP
  - Modify VM availability policy
  - ***Can't*** change the image of a stopped VM

Note: You don't *have* to stop a VM to make many of these changes. However, these are the actions that you *can* make to a stopped VM.

Can't change the Image of a stopped VM
You can create a new VM from the stopped disk, even if the original VM was deleted
Machine type can increase or decrease in size up, however the disk must be big enough to hold the image..

Agenda

1. Google Compute Engine (GCE)
2. Lab #1
3. Compute options (vCPU and Memory)
4. Images
5. Disk Options
6. Lab #2
7. Common GCE actions
8. Review

Lab #1 Creating Virtual Machines

In this lab you will explore the Virtual Machine instance options and create several VMs with different characteristics.

03-1 Creating Virtual Machines

Agenda

1. Google Compute Engine (GCE)
2. Lab #1
3. Compute options (vCPU and Memory)
4. Images
5. Disk Options
6. Lab #2
7. Common GCE actions
8. Review

Machine types

n1-standard-*vCPUs*					n1-highmem-*vCPUs*		n1-highcpu-*vCPUs*	
vCPUs	Mem GB	# Disks / Beta	Max Storage		High Mem		High CPU vCPUs	High CPU Mem
1	3.75 GB	16 / 32	64 TB					
2	7.5 GB	16 / 64	64 TB		13 GB		2	1.80
4	15 GB	16 / 64	64 TB		26 GB		4	3.60
8	30 GB	16 / 128	64 TB		52 GB		8	7.20
16	60 GB	16 / 128	64 TB		104 GB		16	14.4 GB
32	120 GB	16 / 128	64 TB		208 GB		32	28.8 GB
64	240	16/ 128	64 TB					

1	0.6 GB	4 / 16	3 TB	**f1-micro**
1	1.7 GB	4 / 16	3 TB	**g1-small**

shared core micro-bursting

Google Cloud Platform

https://cloud.google.com/compute/docs/machine-types

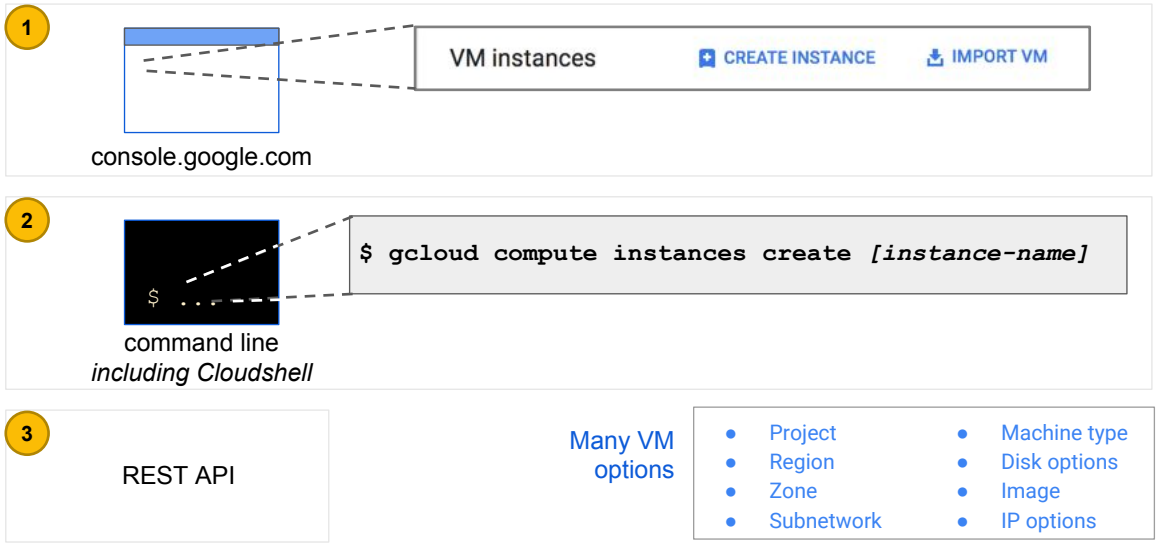
High-CPU doesn't mean more CPU power. It means less memory for the same CPU power.

Micro-bursting: The shared CPU can occasionally use more than its specified share of the CPU for a brief period.

Some zones do not offer 32-core or 64-core VMs. (Currently: us-central1-a and europe-west1-b. *Check documentation for latest information*.)

GPUs are now in Beta in limited locations. NVIDIA® Tesla® K80 GPUs.
https://cloud.google.com/compute/docs/gpus/

Creating a VM



Console does more validation and checking for you. Note that the VM choices are constrained by the network structure. There are inherent networking dependencies. Subnetwork determines DHCP internal IP address on the VM, and subnetwork is homed in one region but may cross multiple zones in that region. The console interface analyzes GCP and presents you with only pull-down selection options that are *possible* to provision and stage. When you use the `gcloud compute` command or the API you are expected to know which options are possible and which are not. If you submit the request with impossible options, the command and API can't check them first. So they will instead attempt to start the VM and the command may result in a "fail" after a few minutes.

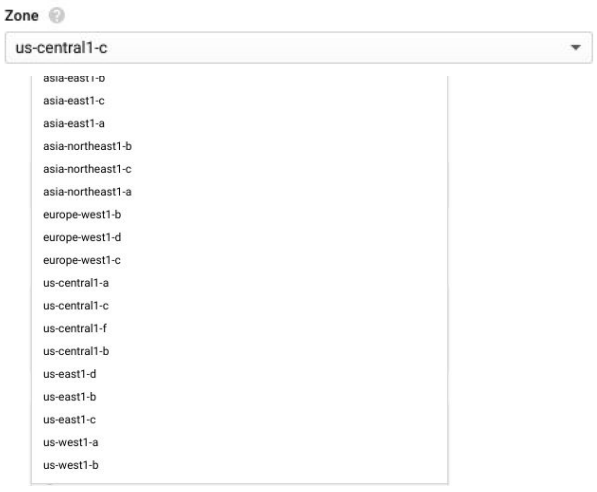
REST API:
https://cloud.google.com/compute/docs/api/how-tos/api-requests-responses

Choosing region and zone

Different zones support different CPU architectures

- Broadwell = 64 CPU max
- Sandy Bridge = 16 CPU max
- Ivy Bridge = 32 CPU max
- Haswell = 64 CPU max

If you create an instance in zone **asia-northeast1-a** it will use a Broadwell processor.



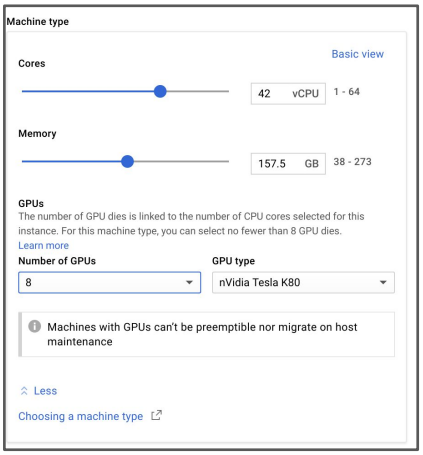
https://cloud.google.com/compute/docs/regions-zones/regions-zones#available

https://cloud.google.com/compute/docs/regions-zones/regions-zones#available
Distributing multiple VMs across multiple zones is recommended for higher availability.

Examples:
us-west1-a / us-west1-b
(Broadwell processors)
us-central-1-a
(Sandy Bridge processors)
Limits the number of CPUs

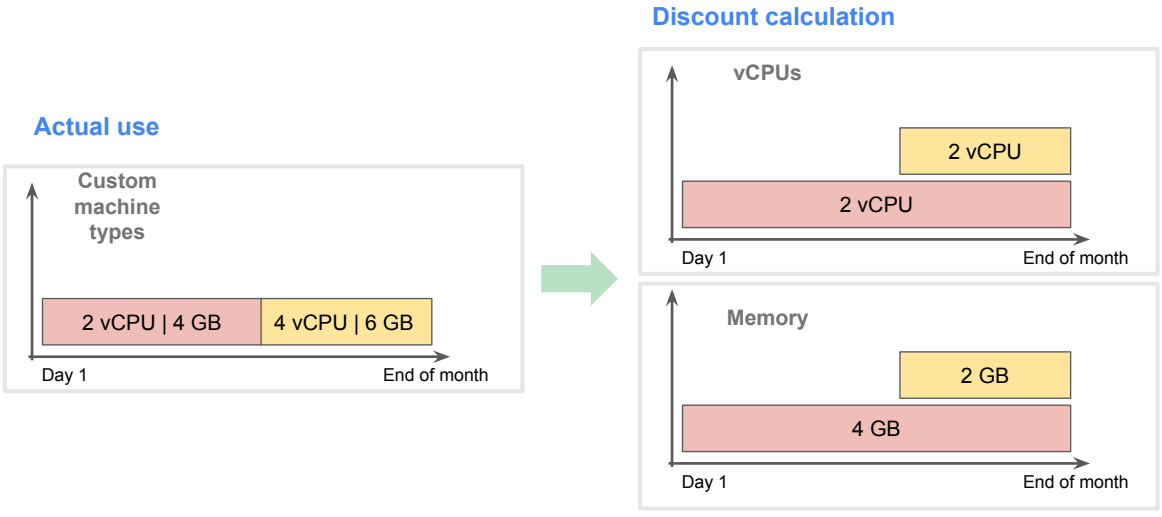
Creating custom machine types

- Number of vCPUs per instance
  - Only 1 vCPU or an even number of vCPUs
- Memory
  - 0.9 GB to 6.5 GB per vCPU
- When to select custom:
  - Requirements fit between the predefined types
  - Need more memory or more CPU
  - Need GPUs
- Customize the amount of memory and CPU for your machine
- Get recommendations for a predefined match
  - Custom VM will generally be more expensive than an identical predefined VM



Charges based on number of vCPUs and memory hours consumed. Sustained use discounts are calculated differently.
Currently, SUSE OS is not supported in custom types.
Multiple network interfaces (NICs) on a single instance is currently not supported.

Custom machine discounts



Accumulating inferred instances by common machine type in a zone isn't possible with custom machine types.
Instead, usage is broken down by resource (vCPU, memory) and the total accumulated resource usage is discounted.

Preemptible

- Lower price for interruptible service (up to 80%)
- VM might be terminated at any time
  - No charge if terminated in the first 10 minutes
  - 24 hours max
  - 30-second terminate warning, but not guaranteed
    - *Time for a shutdown script*
- No live migrate, no auto restart
- You *can* request that CPU quota for a region to be split between regular and preemption
  - Default is preemptible VMs count against region CPU quota

Regular instances usually have a 90 second shutdown notice. Preemptible instances have a 30 second shutdown notice.
Pricing is not bid-based as with other cloud providers. This is intentional to provide you with more reliable cost estimation.
Preemption is based on most recently launched first (not longest running, as with other cloud providers). Avoids preempting too many instances from a single customer.
In one case, preemption rate averages were measured between 5% and 15% over a 7-day project.
Split quotas to avoid running-out of VMs for mission critical work by batch background work scaling up in preemptible instances.

Agenda

1. Google Compute Engine (GCE)
2. Lab #1
3. Compute options (vCPU and Memory)
4. Images
5. Disk Options
6. Lab #2
7. Common GCE actions
8. Review

What's in an image?

- Contents of an image
  - Boot loader
  - Operating system
  - File system structure
  - Software
  - Customizations
- Image storage
  - A tar and gzip'd file
  - In a private area of GCS managed by the image service

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

Images

- Public base images
  - Google, 3rd party vendors, and community; Premium images (p)
  - Linux
    - CentOS, CoreOS, Debian, RHEL(p), SUSE,(p) Ubuntu, openSUSE & FreeBSD
  - Windows
    - Windows Server 2016(p), 2012-r2(p), 2008-r2(p)
    - SQL Server pre-installed on Windows
- Private and custom Images
  - Create new image from VM - pre-configured and installed SW
  - Import from on-prem, workstation, or another cloud
  - Management features: image sharing, image family, deprecation

Premium images (p) incur additional per-minute charges. Premium image prices vary with the machine type. Prices are global and do not vary by region or zone.
https://cloud.google.com/compute/pricing#premiumimages
openSUSE and FreeBSD are the community-supported OS's

Virtualbox or AWS
Image can be stored in Cloud Storage

Image family: points to most recent image so scripts and templates don't need to reference a specific version

Image management best practices:
https://cloud.google.com/solutions/image-management-best-practices

Google Compute Engine uses operating system images to create the root persistent disks for your instances. You specify an image when you create an instance. Images contain a boot loader, an operating system, and a root file system.

For more information on importing Linux images, see:
https://cloud.google.com/compute/docs/images/import-existing-image.

Agenda

1. Google Compute Engine (GCE)
2. Lab #1
3. Compute options (vCPU and Memory)
4. Images
5. Disk Options
6. Lab #2
7. Common GCE actions
8. Review

Boot disk

- VM comes with a single root persistent disk
- Image is loaded onto root disk during boot process
  - Bootable - you can attach to a VM and boot from it
  - Snapshots - incremental backups
  - Durable -- can survive VM terminate
- Some SW is installed and OS is configured by GCE
- Can survive VM deletion if "Delete boot disk when instance is deleted" is unchecked

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

The image name contains the version of the OS. Generally, the latest patches were applied prior to creating the image.
In most cases the image has some level of "auto update" enabled, at least for security patches. Check the documentation of the particular OS.
This behavior may or may not meet your requirements. In some cases, stability is preferred over security, or applying security patches is under change management to minimize risks.

Persistent disks

- Network storage appearing as a block device
  - Attached to a VM through the network interface
  - Durable storage -- *can* survive VM terminate
  - Bounded to zone -- cannot be moved between zones
  - Bootable - you can attach to a VM and boot from it
  - Snapshots - incremental backups
- Features
  - HDD (magnetic) or SSD (faster, solid-state) options
  - Disk resizing -- even running and attached!
  - Can be attached in read-only mode to multiple VMs
  - Checksums built-in; Automatic encryption -- your key optional
  - Performance: https://cloud.google.com/compute/docs/disks/performance

- Persistent disk: use SSD instead of HDD for high-rate random IOPS
- Persistent disks can be attached to multiple instances in read-only mode for data sharing
- Disk resizing -- you can make them bigger, but never smaller

- Economical for storing large amounts of local data
- Performs best with sustained read/write of large files
- Virtual machines can save up I/O capability and burst for an I/O spike well above the average
- Allows smaller disks to be used in cases where typical I/O is low but periodic bursting is required over the maximum allowed sustained I/O
- Performance: https://cloud.google.com/compute/docs/disks/performance

Local SSD disks

- Physically attached to VM - *not available on shared core*
  - More IOPS, lower latency, higher throughput, than Persistent Disk
  - SCSI or NVMe interface
    - NVMe requires an NVMe-enabled image
    - NVMe is faster than SCSI
- 375 GB disk up to eight, total of 3 TB
- Up to 680K read IOPS and up to 360K write IOPS
- Data survives a restart, but not a VM stop or terminate
  - VM-specific, cannot be reattached to a different VM
- *Can* use your own encryption keys

- Limited to 4 partitions total of 1.5 TB in us-central1-a and europe-west1-b due to hardware differences in those zones.
- Local SSDs are not available to shared core machines.

RAM disk

- `tmpfs`
- Faster than Local disk, slower than memory
  - Use when your application expects a file system structure and cannot directly store its data in memory
  - Fast scratch disk, or fast cache
- Very volatile - erase on stop or restart
- May need a larger machine-type if RAM was sized for the application
- Consider using a Persistent disk to backup RAM disk data

https://cloud.google.com/compute/docs/disks/mount-ram-disks
https://www.kernel.org/doc/Documentation/filesystems/tmpfs.txt

Summary of disk options

	Persistent disk HDD	Persistent disk SSD	Local SSD disk	RAM disk
Data redundancy	Yes	Yes	No	No
Encryption at rest	Yes	Yes	Yes	N/A
Snapshotting	Yes	Yes	No	No
Bootable	Yes	Yes	No	Not
Use case	General, bulk file storage	Very random IOPS	High IOPS + low latency	low latency + risk of data loss

HDD = Standard Hard Disk Drive
SSD = Solid State Drive

GCS versus Persistent disk
Both persistent disks and GCS buckets are *network storage that is accessible to the VM*. Notice that the persistent disk is created via the GCE API -- it's part of GCE not part of Google Cloud Storage (GCS). You use `gcloud` for disks, not `gsutil`. They are ***not*** using the same hardware.

An open source method exists to mount a GCS bucket as part of the VM file system. It is recommended that you avoid this technique. If you need GCS storage use the appropriate tools or go through the API (covered in the Storage module). GCS is an object store and some of the basic assumptions contradict the behavior of a file system. For example, an object store provides no serialization lock, so if multiple writers are writing to a file at once, the last one "wins". At several critical points the simulation of a file system on top of an object store breaks down and can produce random errors that you would not expect of a file system.

Maximum disks

Number of cores	Disk number limit
Shared core	16 disks
1 core	32 disks
2-4 cores	64 disks
8 or more cores	128 disks

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

Persistent disk management differences

Cloud Persistent Disk

- Single file system is best
- Resize (grow) disks
- Resize filesystem
- Built-in redundancy
  Built-in snapshot service
- Automatic encryption prior to write
  - use your keys

Computer Hardware Disk

- Partitioning
- Repartition disk
- Reformat
- Redundant disk arrays
- Subvolume management and snapshots
- Encrypt files before write to disk

There are many differences between a physical hard disk in a computer and a Persistent disk which is essentially a virtual networked device.

You CAN perform all of the hardware disk actions listed on the right. They just won't have the benefit you anticipate, and might be a lot of work for something that already comes standard with Persistent Disks.

A single file system gives the best performance on Persistent disk. If you need more space, IOPS, or throughput, you can resize a persistent disk, and resize the filesystem that it contains. To do something similar on a computer disk, you would partition, and then when you need more space you might re-partition and reformat the partitions. You don't need to build redundant disk arrays to get durability, the system already has built-in redundancy. You don't need to implement striping for performance improvement -- that's done automatically and transparently.
You don't need to use something like subvolume management for snapshotting, there is an automatic built-in snapshotting service. Finally, you don't have to encrypt files before writing them to disk, because all files are encrypted before they are transmitted to the Persistent Disk.

Persistent disk snapshots

- Snapshot is not available for local SSD
- Creates an *incremental* backup to GCS
  - Not visible in *your* buckets, it's managed by the snapshot service
  - Consider cron jobs for periodic incremental backup
- Snapshots can be restored to a new persistent disk
  - New disk can be in another region or zone in the same project
  - Basis of VM migration - "moving" a VM to a new zone
    - Snapshot doesn't backup VM metadata, tags, etc
  - Cannot restore to other disk types
- Don't use for database migration across zones

https://cloud.google.com/compute/docs/disks/create-snapshots

Snapshots are the size of the existing disk. They cannot be restored to a smaller disk.
So "shrinking" a disk would be an OS copy process of data from one attached disk to a second smaller attached disk. Not through GCE services.

Multiple copies of each snapshot are redundantly stored across multiple locations with automatic checksums to ensure the integrity of your data.
You cannot share snapshots across projects.

Some people use the term "differential" (ie from "diff") backup, thinking about the differences that have occurred to the disk since the last backup. However, in storage discussions it is common to use "incremental" backup, because each backup file itself is incremental within the lineage, usually date-time-stamped. Both terms are technically correct.

Databases have existing replication technologies that are better suited. When you snapshot a database, you have to write-lock the database, which affects (violates) the SLA. Behind the scenes, you are using compression and decompression on highly structure and already compressed data, so when you restore, it can take hou

Snapshot disk preparation

- Boot disk, halt the system
  - `sudo shutdown -h now`
- Additional disk, unmount the file system
  - `sudo umount </mount/point>`

  If unmount isn't possible
  - Stop applications from writing to the persistent disk
  - Complete pending writes and flush cache
    - `sudo sync`
  - Suspend writing to the disk device
    - `sudo fsfreeze -f </mount/point>`

On Windows, you can enable Volume Shadow Service (VSS) (Beta), which enables a disk to be backed up without having to be shut down.
https://cloud.google.com/compute/docs/instances/windows/creating-windows-persistent-disk-snapshot?hl=en_US&_ga=1.38057187.1394348339.1478206226

Moving a VM to a new zone

- Manual process:
  - Snapshot all persistent disks on the source VM
  - Create new persistent disks in destination zone restored from snapshots
  - Promote ephemeral external IP to static external IP
  - Create new VM in the destination zone and attach new persistent disks
  - Assign static IP to new VM, demote to ephemeral
  - Update references to VM
  - Delete the snapshots, original disks, and original VM
- Automated process:
  - `gcloud compute instances move`
  - Update references to VM, not automatic

Don't use on a VM with a local SSD. The local SSD data cannot be backed up and will just be discarded.
Persistent disks must be attached to only the VM you are going to move, not to multiple VMs.
Sufficient quota must exist for all the resources copied during duplication, or the process will fail.
Why would you move a VM to a different zone? To support availability policy. Also, if a zone is deprecated you can use this process to preserve your application.

Agenda

1. Google Compute Engine (GCE)
2. Lab #1
3. Compute options (vCPU and Memory)
4. Images
5. Disk Options
6. Lab #2
7. Common GCE actions
8. Review

Lab #2 - Working with Virtual Machines

In this lab you will customize a VM and configure it to as a production application server.

Customizations include:

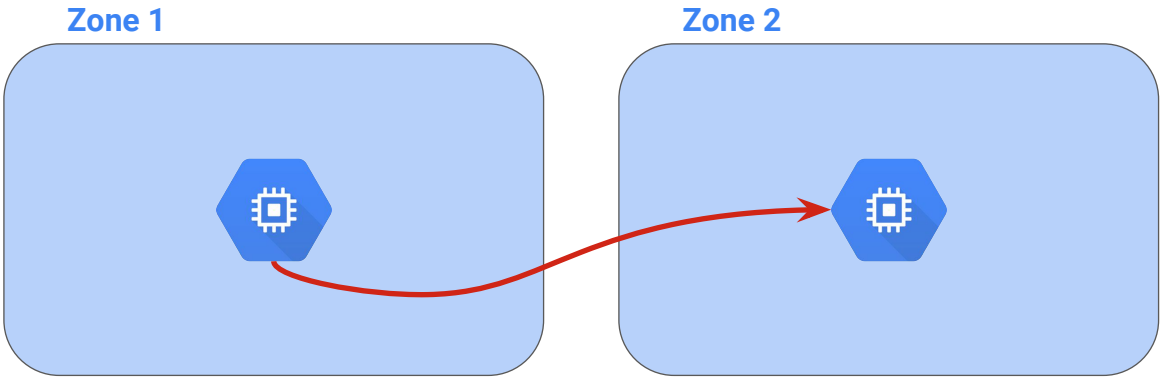
- Disk
- Networking
- Tags
- Backups
- Maintenance procedures and automation

03-2 Lab Working with Virtual Machines

Agenda

1. Google Compute Engine (GCE)
2. Lab #1
3. Compute options (vCPU and Memory)
4. Images
5. Disk Options
6. Lab #2
7. Common GCE actions
8. Review

Moving an Instance to a new zone




```
gcloud compute instances move
```

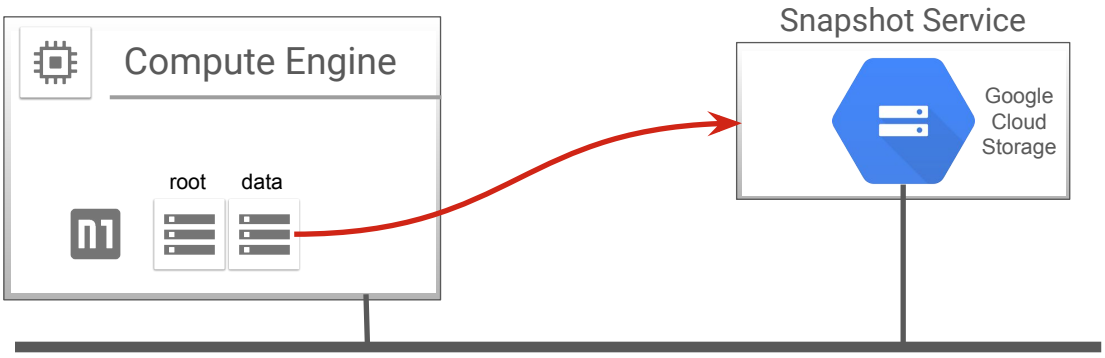

- Move using gcloud command-line utility:
  - `gcloud compute instances move`
- Use API `moveInstance` method
  - Make a POST request to the `moveInstance` method with a request body that contains the `targetInstance` and the `destinationZone`
- Use disk snapshots (manual procedure)

Requirements

- Quotas to:
  - Create new snapshots
  - Promote any ephemeral external IP addresses
  - Create the new instance and disks in the destination zone
- Persistent disks attached to the instance you want to move cannot be attached to more than one instance
- Source instance cannot contain a local SSD
- Must update any existing references you have to the original resource (not done automatically)

For more information on the requirements for moving instances between zones, see: https://cloud.google.com/compute/docs/instances/moving-instance-across-zones#requirements.

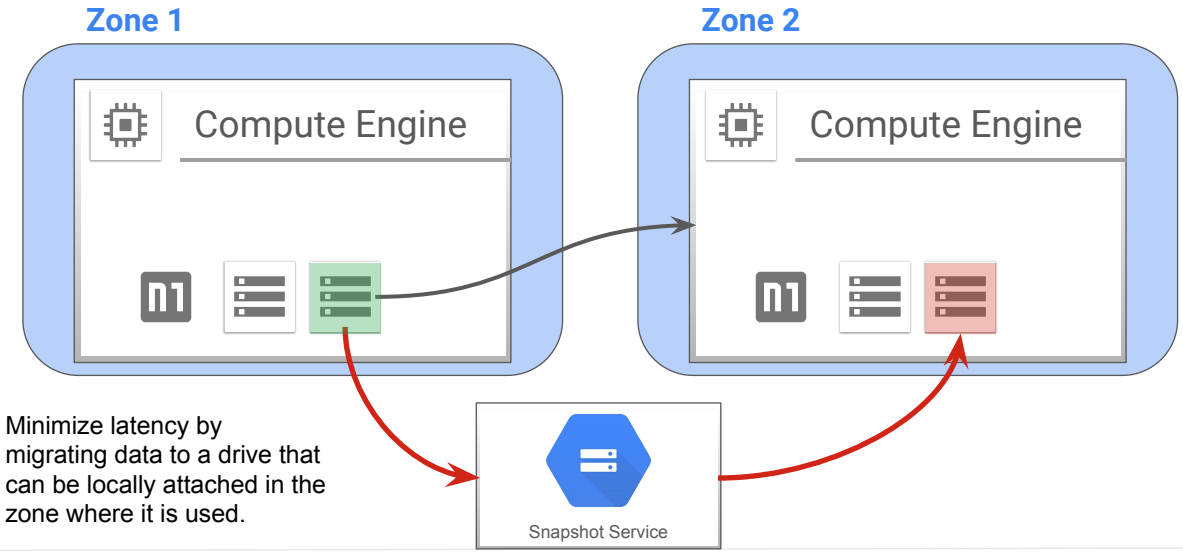
Snapshot: disk backup



Backup critical data into a durable storage solution to meet application availability and recovery requirements

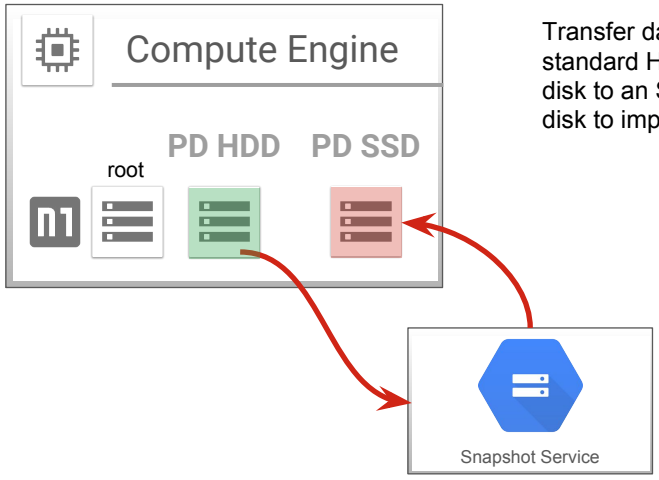
Google Cloud Storage (GCS) is covered in the next part of this class.

Snapshot: migrate data between zones



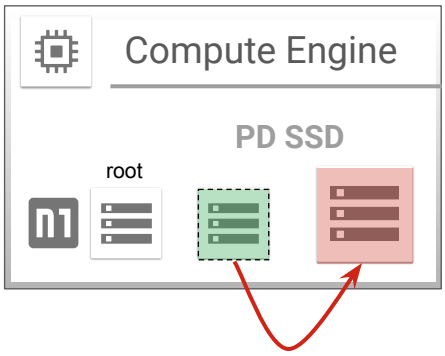
Minimize latency by migrating data to a drive that can be locally attached in the zone where it is used.

Snapshot: transfer to different disk type



Transfer data from a standard HDD Persistent disk to an SSD Persistent disk to improve performance

Resize persistent disk



- Improve I/O performance by increasing storage capacity of a persistent disk
- You can edit a persistent disk, increase it's size and increase its IOPS capacity
- A disk can be resized even when it is attached to a VM and while it is running
- You don't need to use a snapshot to accomplish this
- *You can grow disks, but never shrink them*

	Sustained Random IOPS Limit (Read or Write)	Sustained throughput limit (mb/s)
10 GB SSD	300	4.8
100 GB SSD	3000	48
500 GB SSD	15000	240

Note: A snapshot cannot be restored to a smaller disk. So not only can you not shrink an existing disk, you can't restore to a disk to shrink it. If you needed to do something like that you'd need to use OS copy tools. So, it is better to be conservative with the size of the disk and to later expand it as necessary.

Agenda

1. Google Compute Engine (GCE)
2. Lab #1
3. Compute options (vCPU and Memory)
4. Images
5. Disk Options
6. Lab #2
7. Common GCE actions
8. Review


cloud.google.com

Create and boot an instance

Select a public image

```
gcloud compute images list
```

Create the instance (automatically uses the latest version)

```
gcloud compute instances create [INSTANCE_NAME]--image-family [IMAGE_FAMILY] --image-project [IMAGE_PROJECT]
```

Create an instance from a SPECIFIC image (version)

```
gcloud compute instances create [INSTANCE_NAME]--image [IMAGE_ID] --image-project [IMAGE_PROJECT]
```

Create an instance from a CUSTOM image

```
gcloud compute instances create [EXAMPLE_INSTANCE]--image [IMAGE_NAME]
```

Create an instance from a SHARED image

```
gcloud compute instances create [INSTANCE_NAME]--image [IMAGE] --image-project [IMAGE_PROJECT]
```

Create an instance from a SNAPSHOT (backup image)

Step 1: Create the persistent disk from the snapshot

```
gcloud compute disks create [DISK_NAME] --source-snapshot [SNAPSHOT_NAME]
```

Step 2: Start the instance by booting from the persistent disk

```
gcloud compute instances create [INSTANCE_NAME] --disk name=[DISK_NAME],boot=yes
```

Preemptible instance add:

```
--preemptible
```

Create and attach a persistent disk

Create the disk
```
gcloud compute disks create [DISK_NAME] --size [DISK_SIZE] --type [DISK_TYPE]
```

Attach the disk to the instance
```
gcloud compute instances attach-disk [INSTANCE_NAME] --disk [DISK_NAME]
```

Identify disk
```
ls /dev/disk/by-id
```

Format the disk (in this case with ext4)
```
sudo mkfs.ext4 -F -E lazy_itable_init=0,lazy_journal_init=0,discard /dev/disk/by-id/google-[DISK_NAME]
```

Create a mount point
```
sudo mkdir -p /mnt/disks/[MNT_DIR]
```

Mount the disk
```
sudo mount -o discard,defaults /dev/disk/by-id/google-[DISK_NAME] /mnt/disks/[MNT_DIR]
```

Configure access
```
sudo chmod a+w /mnt/disks/[MNT_DIR]
```


Just reviewing the process and commands that you will be performing in the lab.