Capacity planning and cost optimization

Architecting with Google Cloud Platform: Design and Process

Google Cloud

Last modified 2018-08-08

“Cost Optimization is built-in to many GCP services.”

https://pixabay.com/en/coins-calculator-budget-1015125/

Agenda

Capacity planning

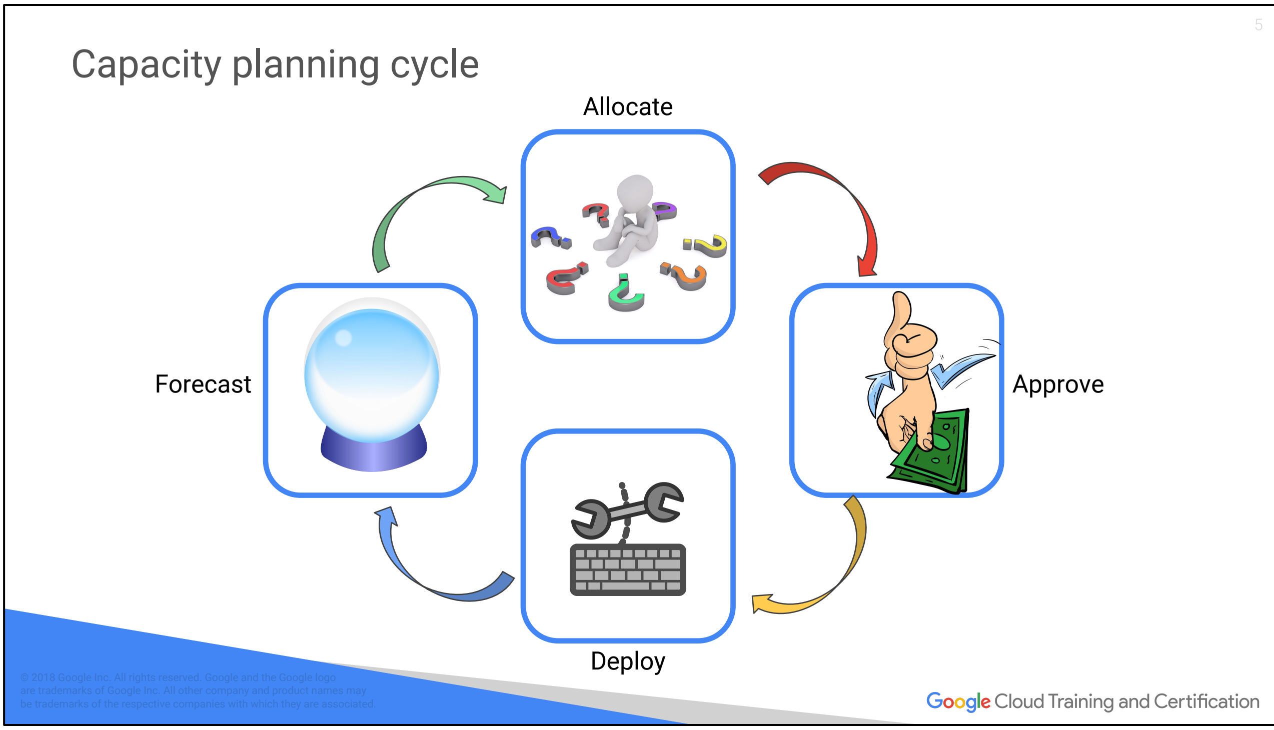
Cost management

Photo service: Cost and capacity

Design challenge #6: Dimensioning

No GCP lab in this module

Capacity planning



The standard capacity planning cycle. It starts with forecasting. Each step in the cycle is discussed in the sections that follow.

https://pixabay.com/en/astrology-ball-crystal-156576/
https://pixabay.com/en/water-drop-crystal-ball-sphere-38389/
https://pixabay.com/en/money-financial-business-cach-1425581/
https://pixabay.com/en/question-mark-consider-think-2318030/
https://pixabay.com/en/keyboard-fix-configure-spanner-27183/

Forecast

Monitor growth

Predict future demand

Plan for feature launches

Forecasting is iterative:

1. What was the prediction last time?
2. Compare with actuals. High or low?
3. Account for error in the prediction model.
4. Make a prediction for next time.

Forecasting converges on a practical value by learning from errors.

What other values should you include in your forecasting estimate?

Predict future demand (monitoring growth, planned feature launches)
Iterative process of comparing previous forecasts against actuals.

iterative method: An initial guess converges towards a practical value by incorporating the error resulting from the difference between the previous prediction and the actual result.
https://en.wikipedia.org/wiki/Iterative_method

Forecasting estimation

Testing beats tradition

Don't mistake launch demand for stable demand, or you might over-provision.

Remember to include N+2 servers or you will risk overload failure.

Add headroom to deal with non-linearity of demand.

Add overhead to instance estimates.

Google Cloud Training and Certification

Testing beats tradition.
Traditional values provided as estimates have hidden assumptions. For example, if someone says "Use 1000 QPS for that. We always use 1000 QPS." It might be a good estimate, it might not. You just don't know. And if it's not a good estimate, your recommendations could be wrong.

Don't mistake launch demand for stable demand.
Remember to include N+2 or you risk overload failure
Add headroom for non-linearity of demand
Allow for overhead
Consider related/dependent capacities: Disk capacity changes with VM/CPU capacity, Network capacity changes with # of cores.
Don't assume symmetries. For example, an internal IP and an external IP offer different and asymmetric performance.

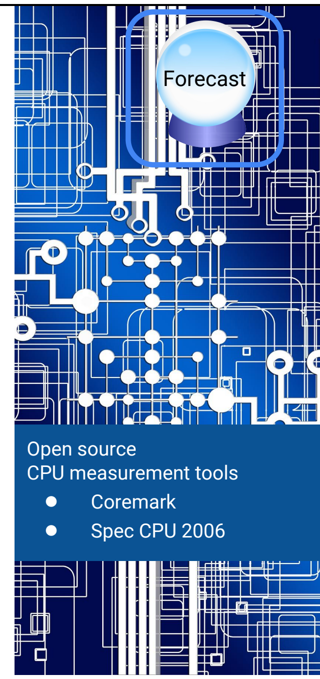
Instance overhead estimation

Of the 100% capacity of a VM, some will be consumed by overhead.

If you can't test, 30% overhead is a cautious estimate

Load testing can better estimate the overhead

- **OS image** choice can make a difference (up to 20%)
  - Test several images
- **Hardware architecture** (specific to zone) can make a difference for some workloads
- **OS firewalls** turn off if GCP firewalls are sufficient
- On Windows, move **Pagefile** to Local SSD if affordable



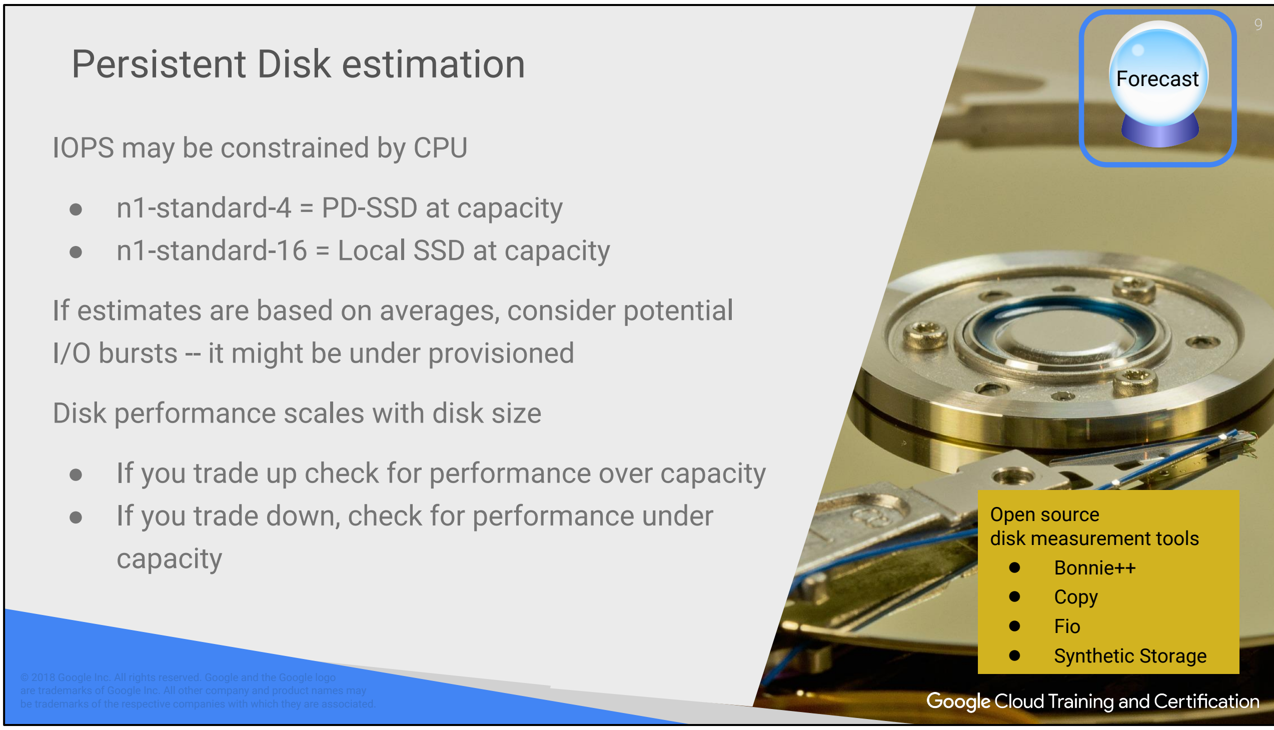
Google Cloud Training and Certification

This page lists the hardware architectures in the different zones. Sandy Bridge, Haswell, Broadwell, IvyBridge, Skylake, and so forth.
https://cloud.google.com/compute/docs/regions-zones/regions-zones
There are a lot of benchmark comparisons online. It is suggested that you test your application and workload in different zones to see what difference the hardware in the zone might make.

Open source CPU measurement tools

- Coremark
- Spec CPU 2006

https://pixabay.com/en/board-circuits-trace-control-center-1709192/

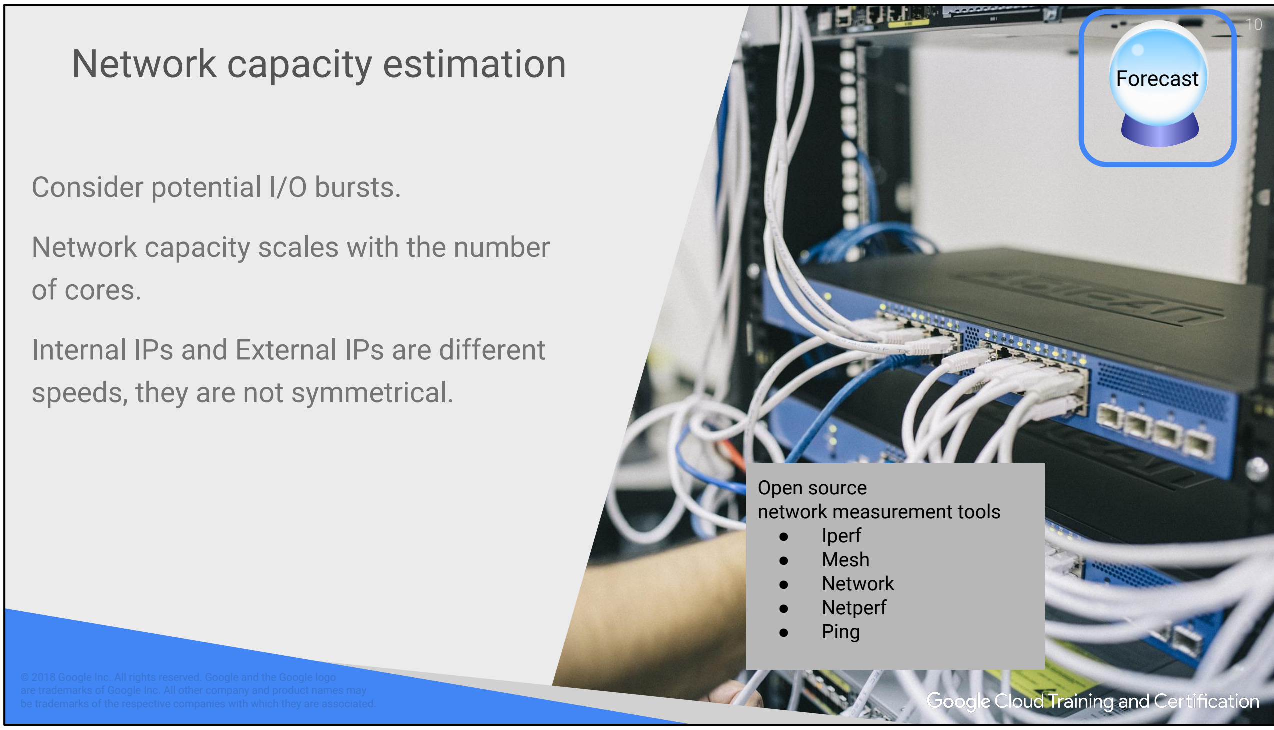


Consider potential I/O bursts. If you have planned on IOPS based on an average, and the actual disk usage is bursty, the disk could be under provisioned for dealing with the bursts.
Persistent disk performance scales with the size of the disk. So if you trade up to a larger disk in your design, revisit the performance to avoid over capacity, and if you trade disk size down, check for under capacity.
Potential IOPS may be constrained by CPU. An n1-standard-4 can drive a PD-SSD at capacity, and an n1-standard-16 can drive a Local SSD at capacity.

Open source disk measurement tools

- Bonnie++
- Copy
- Fio
- Synthetic Storage

https://pixabay.com/en/hard-drive-hdd-macro-disk-computer-463922/



In general, Internal IPs are faster than External IPs.

Open source network measurement tools

- Iperf

Mesh
Network
Netperf
Ping

Consider potential I/O bursts. If you have planned on IOPS based on an average, and the actual traffic is bursty, it could be under provisioned for dealing with the bursts.

Network capacity scales with the number of cores. So if you change out the number of cores in the VMs in your design, revisit the network capacity to make sure the design does not over- or under- provision.

Internal IPs and External IPs are different speeds, they are not symmetrical. Even VM-to-VM in the same zone over External IPs can be ~1Gbps vs ~8.5Gbps for Internal IPs.

https://pixabay.com/en/engineer-fiber-fieldengineer-2558705/

Workload estimation

Throughput depends on

- Types of requests or operations
- Requests that change state
- Request size (payload)
- Whether system uses sharding, pipelining or batching

https://pixabay.com/en/engineer-fiber-fieldengineer-2558705/

Perfkit Benchmarker

PerfKit Benchmarker is an open source benchmarking tool used to measure and compare cloud offerings.

Originated by Google.

It is a wrapper around (currently) **28** open performance benchmarking tools.

Many GCP services are designed to achieve efficiency at scale. So single functional tests might not perform as well as larger scale tests.

http://googlecloudplatform.github.io/PerfKitBenchmarker/
https://github.com/GoogleCloudPlatform/PerfKitBenchmarker

https://pixabay.com/en/horses-racing-jockey-sport-rider-380415/

Allocate

Forecasting tells you how much capacity is required.

Allocation tells you how much resource is necessary to provide that much capacity.

The allocation ratio (sometimes called the resource-to-capacity ratio):

```
resource : capacity
```

How much resource yields how much capacity?

- Estimate the capacity required based on forecasting
- Validate the estimates with load testing
- Calculate the resources required based on the allocation ratio

Google Cloud Training and Certification

QPS - Queries Per Second

- A common metric used for capacity
- Queries do not consume the same amount of resource under all conditions
- Resources required may change over time
- QPS often includes assumptions about static resources

Allocation example

Forecasting

- Service is growing.
- In the coming year, demand for storage throughput capacity will increase 30%.
- Current capacity is 252,000 qps. (252,000 x 130% = 327,600)
- Service requires data storage throughput capacity of 327,600 qps

Allocation

- Bigtable standard says max of 10,000 qps per Bigtable node
- Allocation ratio: 10,000:1
- Resource required for 327,0600 qps = 33 Bigtable nodes

Have you considered alternatives besides increasing resources?

Identify improvements:

- Caching?
- Tuning?
- Better algorithms?
- Alternate services?

Example. If the service expects 20% growth in the coming year, and you can make it 20% more efficient without additional resources, then it might not need to grow additional resource to handle the increased capacity.

Approve

If you have been using SLOs and SLIs, you'll have the numbers to support the recommendation.

Have you considered alternatives besides increasing existing resources?

Capacity planning for launch

Deploy

Test first

Work through issues before serving user traffic

- Identify bottlenecks

Slow, staged, iterative

- Dark launches
- Use invitations to stage launch

Use invitations to slow or stage a launch.
https://pixabay.com/en/dark-night-smoke-spaceship-launch-2578134/

PROCESS

Have a balanced approach to the launch process

Easy to use	Customizable
Catches obvious errors, handles simple cases	Detailed, consistent, reproducible for hard issues
Can be used for a large number of simple launches	Can be used for a small number of complex launches
Defaults for common launch cases	Adaptable to different kinds of launches and circumstances

Launch includes initial deployment, but also any change or upgrade to the system.

https://pixabay.com/en/weight-scale-equal-arm-balance-scale-2402966/

Cost management

Optimizing VM cost

VM Dimensioning: standard, high-CPU, high-mem, GPU, and custom

Sustained use discounts

- Machine-type discounts
- Inferred instance type discount

Committed use discounts

- Compute Engine and Kubernetes Engine VMs
- **Not:** App Engine / Flex, DataProc, Dataflow, or Cloud SQL VMs

Preemptible VMs

Google Cloud Training and Certification

https://cloud.google.com/compute/docs/sustained-use-discounts
https://cloud.google.com/compute/docs/instances/signing-up-committed-use-discounts
https://cloud.google.com/compute/docs/instances/preemptible

Compute Pricing: https://cloud.google.com/compute/pricing
Preemptible Instance Pricing:
https://cloud.google.com/compute/pricing#predefined_machine_types

Optimizing disk cost

Determine how much space you need

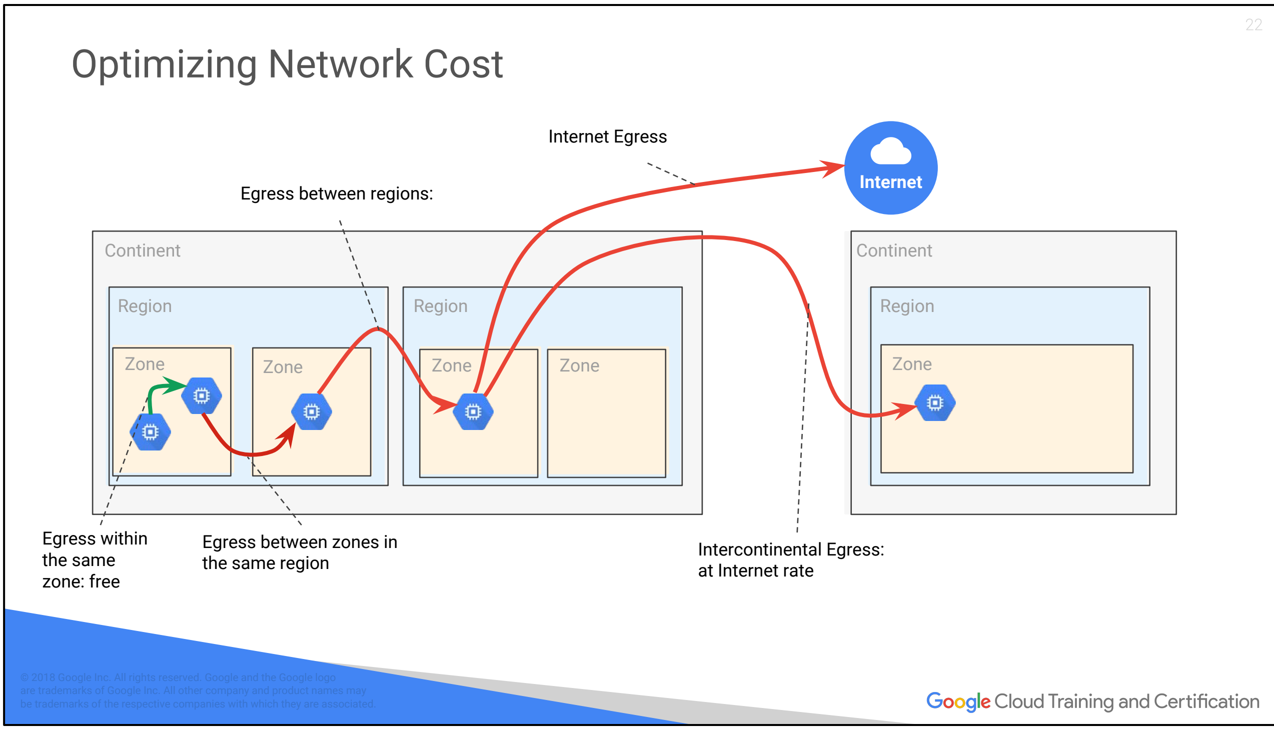
Determine what performance characteristics your applications require

- I/O Pattern: small reads and writes or large reads and writes
- Configure your instances to optimize storage performance

Monthly charges	**Standard PD**	**SSD PD**
10 GB	$0.40	$1.70
1 TB	$40	$170
16 TB	$655.36	$5,570.56

Google Cloud Training and Certification

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



Ingress is free. Networking costs are similar per GCP product but are billed per product. So you need to view the pricing documentation per product for the details. Example: Cloud Storage has standard egress costs. But there are also separate charges for data migration and for Cloud Storage operations.

Example: Egress between regions: $0.01 per Gb. Egress to Internet: 1 TB to world destinations, $0.12.

https://cloud.google.com/bigtable/pricing#network
https://cloud.google.com/storage/pricing

VM to VM in the same Zone

Egress throughput cap

- Each core is subject to a 2 Gbits/second (Gbps) cap
- Each additional core increases the cap by 2 Gbits/second
- Maximum is 16 Gbits/second
- Instances of 0.5 vCPU or less get 1 Gbits/second cap

These theoretical caps are irrespective of actual network condition

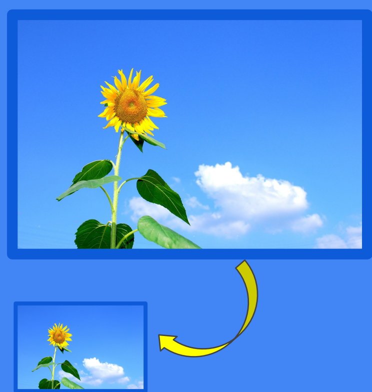
To measure the actual egress network performance from a VM through the VPC, Google provides a tool: **PerfKitBenchmarker**

- https://github.com/GoogleCloudPlatform/PerfKitBenchmarker

Google Cloud Training and Certification

https://cloud.google.com/compute/docs/vpc/advanced-vpc#measurenetworkthroughput
https://github.com/GoogleCloudPlatform/PerfKitBenchmarker

Cost and capacity



Could the photo application offer the same service for less money?

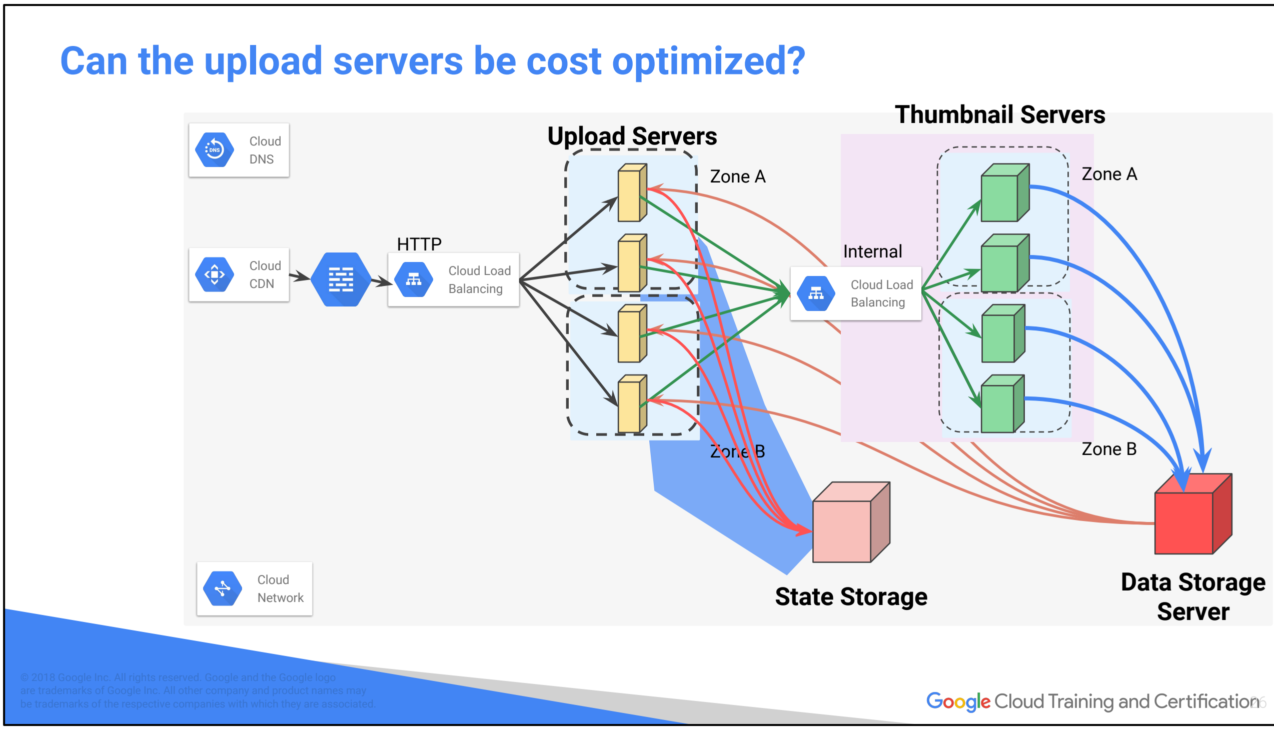
https://pixabay.com/en/summer-sunflower-flowers-sky-cloud-368224/

Budget time again...

The reports from last year are in. The forecasts are done. And the recommendation is ready to go for approval. However, there is one thing you need to do before submitting the recommendation.

Could the service save money by moving from the current upload instances, which are n1-standard-1 to n1-standard-4, 4-core instances?

Be careful. The number of cores also changes the network throughput and disk characteristics.



Rightsize servers
Autoscale servers in instance groups
Consider provisioning a optimum number of servers as committed use
Measure and rightsize the disks to match the work and performance on the servers
Move backups to archival storage
Make state storage local to the servers

Non-abstract cost optimization problem

VMs are subject to a max egress caps of 2 Gbps / core from VM-to-VM in the same zone over internal IPs. Each core increase the egress cap by 2 Gbps. There is a theoretical max of 16 Gbps per VM.

- The current upload VMs are n1-standard-1 machine types, with one vCPU running Debian 9 OS.
- Monitoring indicates a peak pool size of 576 machines in the managed instance groups for the front end servers.
- PerfkitBenchmark tests indicate current performance of n1-standard-1 is about 1.4 Gbps and performance of n1-standard-4 is about 6.1 Gbps

Your challenge:

Make a recommendation to move to n1-standard-4 machines or not.

Google Cloud Training and Certification

https://cloud.google.com/compute/docs/networks-and-firewalls
https://cloud.google.com/compute/docs/vpc/advanced-vpc#measurenetworkthroughput

Updated section added to the documentation:
https://cloud.google.com/compute/docs/networks-and-firewalls#ingress_and_egress_throughput_caps

Bottom line:

- Yes, the effective ingress capacity to a VM is approximately 10 Gbps.
- This is an engineering artifact, not a product feature.
- It could change in the future.
- Customers should use other DDoS protection techniques (GCLB, CDN, Cloud Armor).

Move to n1-standard-4 or stick with n1-standard-1?

Peak pool size is 576 instances
Perfkit Benchmark says n1-standard-1's are getting 1.4 Gbps
Peak throughput is 576 x 1.4 Gbps = **806.4** Gbps peak throughput

Perfkit Benchmark says n1-standard-4's are getting 6.1 Gbps
806.4 Gbps / 6.1 Gbps = **132** instances

Go to the pricing calculator:
https://cloud.google.com/products/calculator/

576 Web servers n1-standard-1 = Total Estimated Cost: **$13,980.96** per 1 month
132 Web servers n1-standard-4 = Total Estimated Cost: **$12,815.88** per 1 month

Cost optimization savings: **$1,165.08**

YOUR TURN

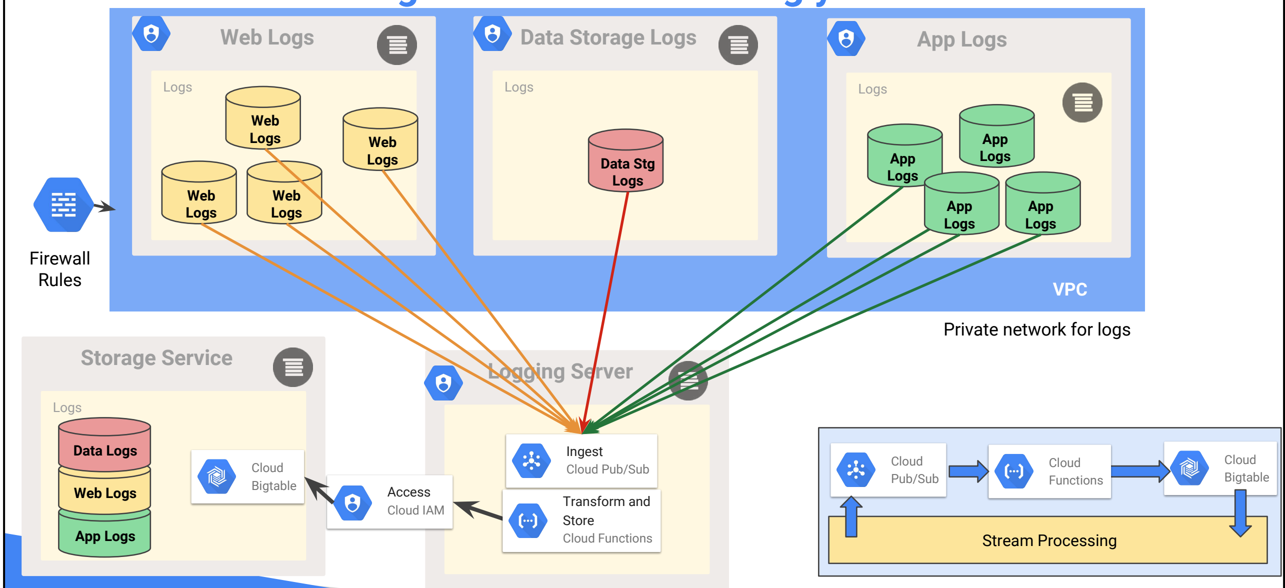
Design challenge #6

Dimensioning

Google Cloud Training and Certification

https://pixabay.com/en/the-strategy-win-champion-1080527/

What if the service grows x2 in the coming year?



Bigtable

Current solution for joined logs:

- 22 x Bigtable nodes
- SSD disk technology

Growth in the photo thumbnail service is expected to double in the coming year.

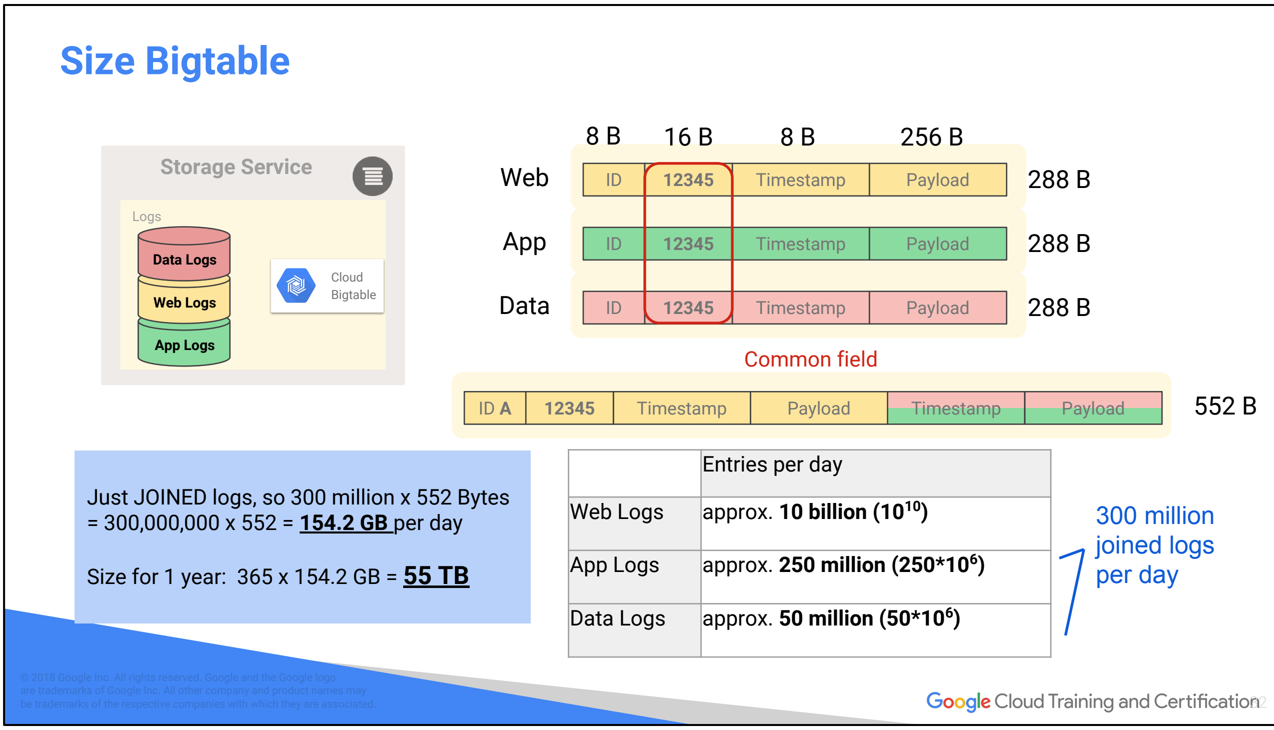
CALCULATE:

- Queries per Second (QPS)
- Throughput in MB/s
- Total data stored in one year (x 2)

Bigtable

- Each node will deliver up to 10,000 QPS and 10 MB/s of throughput.

The number of Big Table nodes is explained on the next slide.



Just JOINED logs, so 300 million x 552 Bytes = 300,000,000 x 552 = 154.2 GB

300000000*552/1024^3 = 154.2 TB x 365 = 54.97 TB (~ 55 TB per year)

Big Table storage capacity per node:

- SSD = 2.5 TB per node
- HDD = 8.0 TB per node

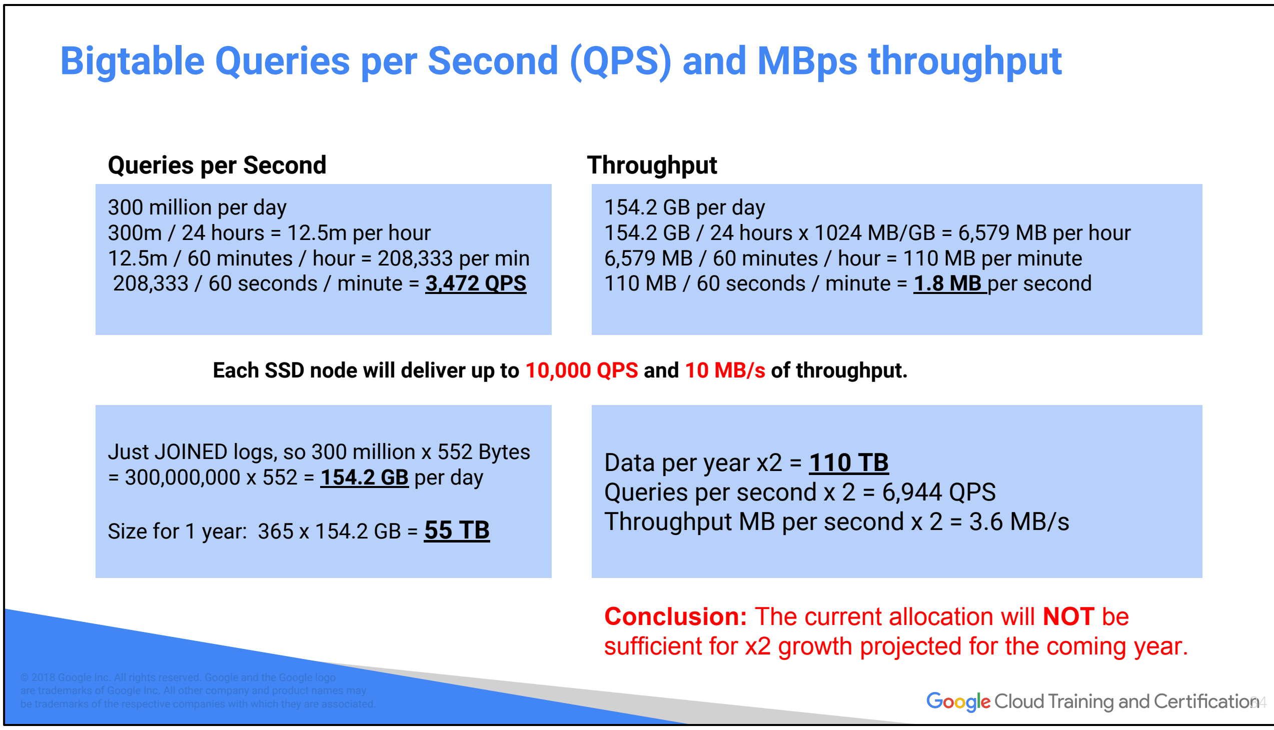
55 TB per year / 2.5 TB per node = 22 BigTable SSD nodes
55 TB per year / 8.0 TB per node = 7 BigTable HDD nodes

Take a few minutes to design your solution

Problem: The application is expected to double in size in the coming year. The dimensioning question to be answered is whether the logging service can stand the growth. Given the parameters provided on the next slides, should the company budget to purchase more Bigtable nodes in the coming year, or will the current allocation be sufficient for the period?

Calculate the dimensions x2 to advise management.

There are multiple designs possible depending on your assumptions. Your solution may be better than the one shown. The point of this exercise is to "think about the design" to develop your architecting skills.



Throughput:
((154.2*1024)/24)/60/60 = 1.8MBps

Given 55 TB of yearly data, they require 22 Big Table SSD nodes today.
If they continued to use Big Table SSD nodes, they will need 44 nodes to handle the 2x growth due to the capacity.

Big Table read operations per node:

- SSD is 10,000 QPS @ 6ms latency for reads and writes
- HDD is 10,000 QPS @ 50ms latency for writes
- HDD is only 500 QPS @ 200ms latency for reads

Big Table storage capacity per node:

- SSD = 2.5 TB per node
- HDD = 8.0 TB per node

Technically, the solution could operate with 8 Big Table HDD nodes.
8 HDD Nodes x 500 QPS each = 4,000 QPS (Only 3,472 QPS is required by the solution)
8 HDD Nodes x 8.0 TB per node = 64 TB capacity (only 59.4 TB is required by the solution)

They would need 15 Big Table HDD nodes for the 2x solution.
15 HDD Nodes x 500 QPS each = 7,500 QPS (Only 6,944 QPS is required by the solution)
15 HDD Nodes x 8.0 TB per node = 120.0 TB capacity (only 118.8 TB is required by the solution)

Pricing

Given what you know about the dimensions of the logging service, would it be less expensive to switch from hard disks (HDD) to solid state disks (SSD) on the Bigtable nodes?

What is the cost using HDD instead of SSD?

- SSD provides 6ms latency at 99th percentile.
- HDD provides 200ms for reads and 50ms for writes at 99th percentile.

Use the products calculator to make a recommendation.

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

Google Cloud Training and Certification