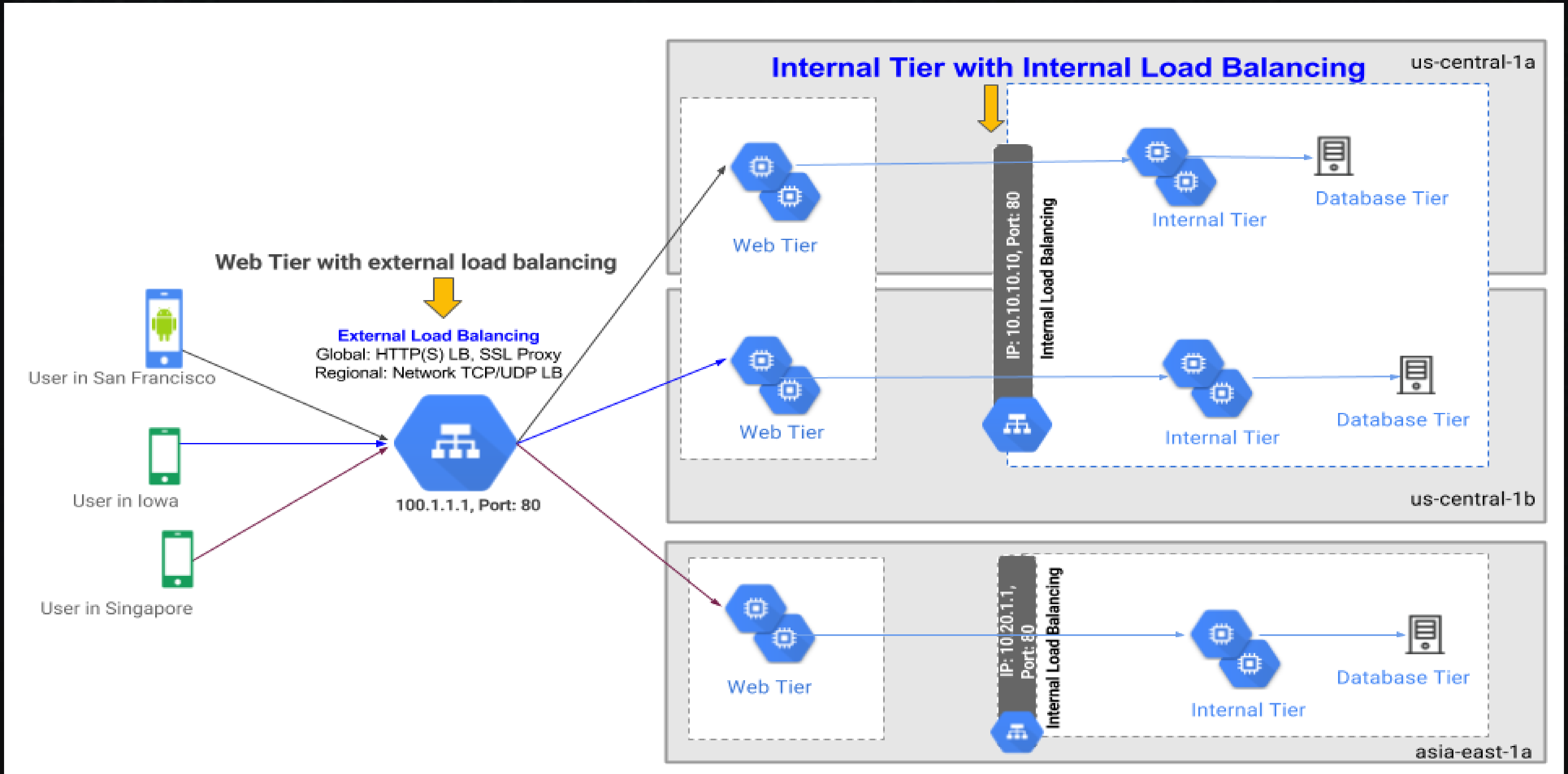
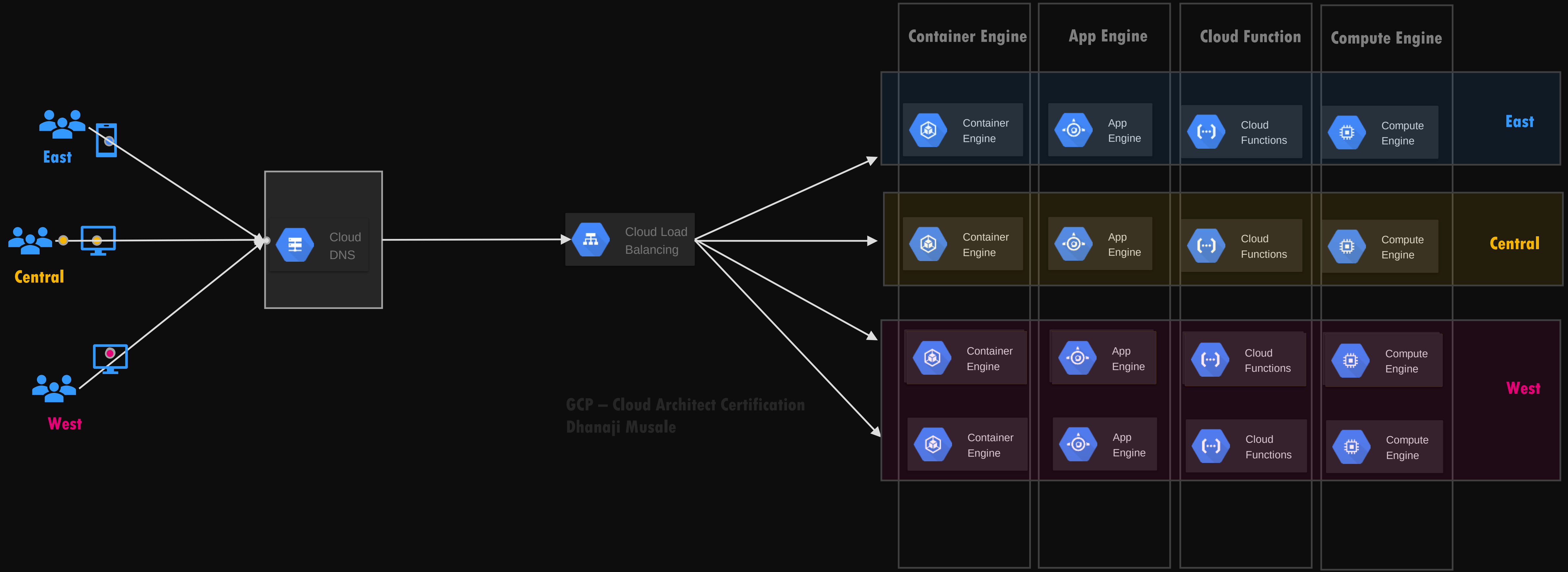
GCP Networking

GCE – Load Balancing , Auto Scaling & High Availability

Google Cloud Platform

Load Balancing, High Availability and Auto Scaling





GCP – Cloud Architect Certification
Dhanaji Musale

Load Balancing Auto Scaling & HA

Google Compute Engine – Terms used

Instance Template

Instance Groups

GCE : Instance Templates

Instance templates define the **machine type, image, zone**, and other instance properties for the instances in a managed instance group

Create an instance template once and can reuse it for multiple groups and configuration

Instance template is a global resource that is not bound to a zone or a region, but it can have zonal resources which makes templates zonal.

GCE : Instance Groups

Instance groups are group of virtual machine (VM) instances so that you don't have to individually control each instance in your project

There are two types of Instance Groups

- Managed Instance Groups
  - Zonal Managed
  - Reginal Managed
- Unmanaged Instance Groups

Compute Engine

Load Balancing

High Availability

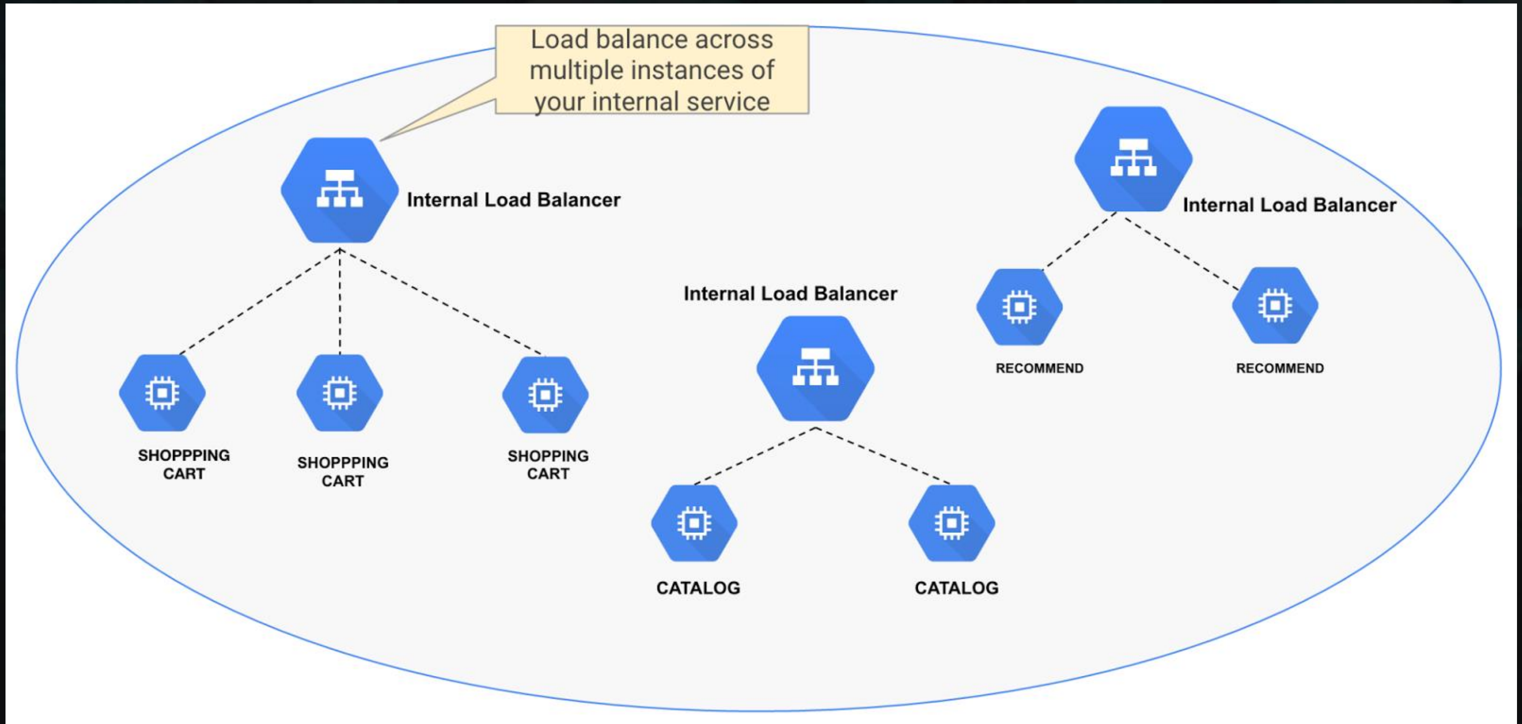
Auto Scaling

Load Balancing

Google Compute Engine

GCE : Load balancing

GCP offers server-side load balancing so you can distribute incoming traffic across multiple virtual machine instances.



Google Cloud Platform Application composed of ILB-scaled services

GCE : Load balancing

Load balancing provides the following benefits:

- ✓ Scale your application
- ✓ Support heavy traffic
- ✓ Detect and automatically remove unhealthy virtual machine instances. Instances that become healthy again are automatically re-added. Health checking configuration used to identify healthy instances.
- ✓ Route traffic to the closest virtual machine

GCE : Types of Load Balancing

Global external load balancing

- HTTP(S) load balancing
- SSL Proxy load balancing
- TCP Proxy load balancing

Global

Regional external load balancing

- Network load balancing

Regional internal load balancing

- Internal load balancing

Regional

- HTTP Load Balancer
- SSL Load Balancer
- TCP Proxy Load Balancer

Global External Load Balancer

Google Compute Service

HTTP Load Balancer

Global External Load Balancer

Global Ext. LB : HTTP(S) Load Balancing

Google Cloud Platform (GCP) HTTP(S) load balancing provides global load balancing for HTTP(S) requests destined for your instances.

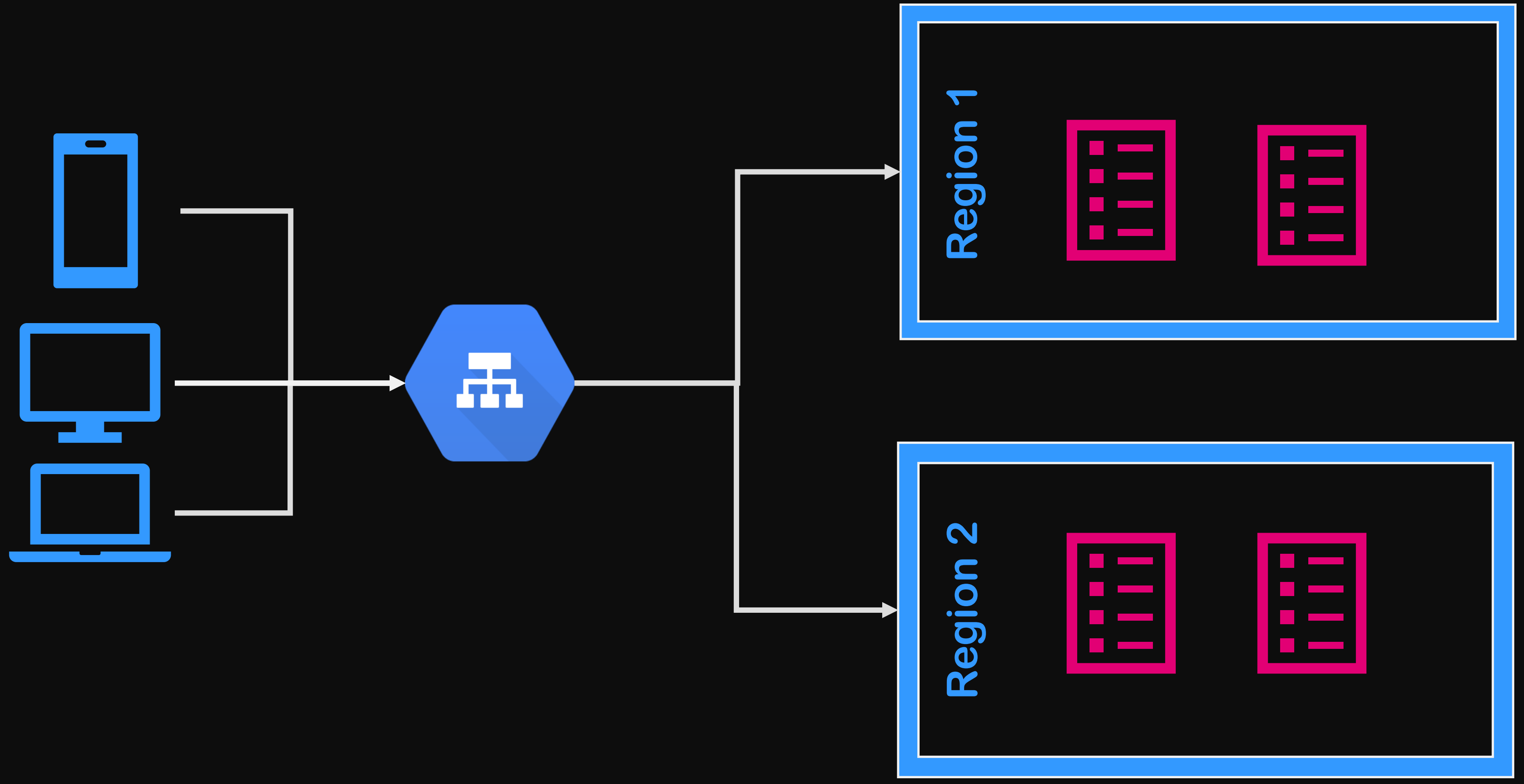
You can configure URL rules that route some URLs to one set of instances and route other URLs to other instances.

Requests are always routed to the instance group that is closest to the user, provided that group has enough capacity and is appropriate for the request.

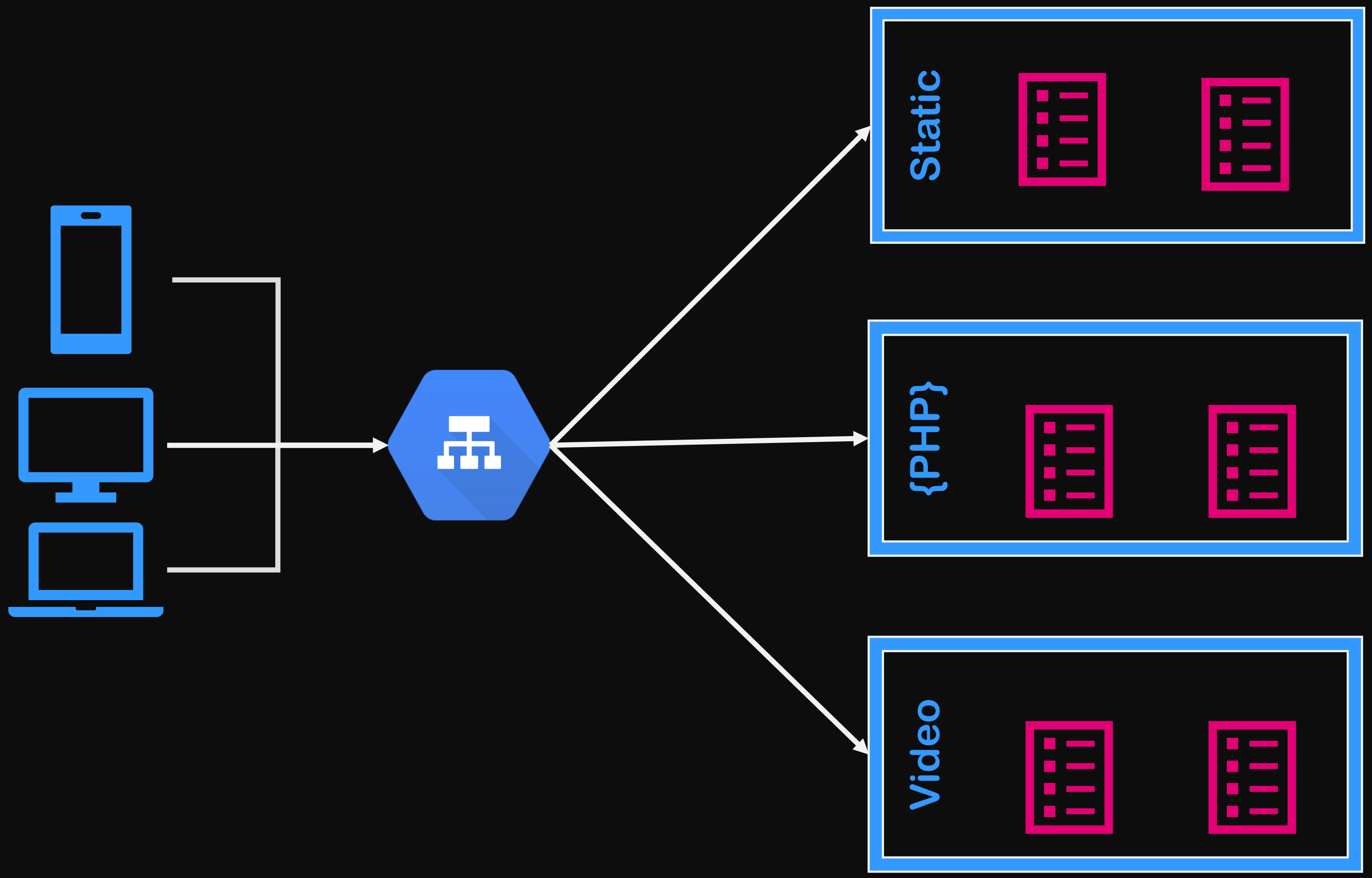
If the closest group does not have enough capacity, the request is sent to the next closest group that does have capacity.

HTTP(S) load balancing uses instance groups to organize instances. Make sure you are familiar with instance groups before you use load balancing.

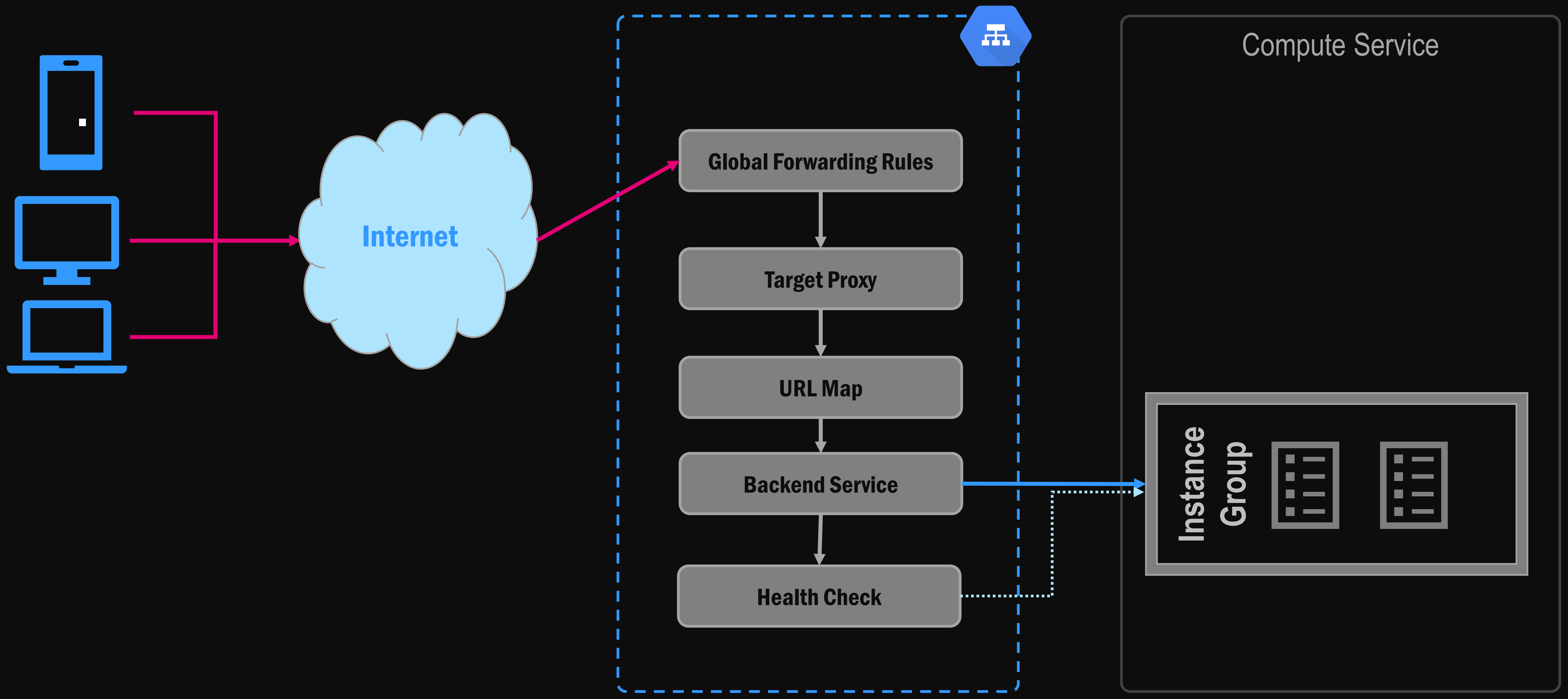
HTTP(S) LB : Cross Region Load Balancer



HTTP(S) LB : Cross Region Load Balancer



HTTP(S) LB Basics



HTTP(s) LB Basics : Load distribution algorithm

HTTP(S) load balancing provides two methods of determining instance load (**param : balancingMode**)

- Requests per second (RPS)
- CPU utilization

HTTP(s) LB Basics : Session Affinity

Session affinity sends all request from the same client to the same virtual machine instance as long as the instance stays healthy and has capacity.

GCP HTTP(S) Load Balancing offers two types of session affinity:

- client IP affinity
- generated cookie affinity

HTTP(s) LB Basics : WebSocket Proxy Support

- The HTTP(S) load balancer has native support for the WebSocket protocol.
- Backends that use WebSocket to communicate with clients can use the HTTP(S) load balancer as a front end, for scale and availability.
- The load balancer does not need any additional configuration
- The default timeout for a connection is 30 seconds regardless of whether the connection is in use or not.
- If your service requires longer-lived connections, increase the timeout value (timeoutSec in the API) for the backend service

HTTP(s) LB Basics : Interfaces

Your HTTP(S) load balancing service can be configured and updated through the following interfaces:

The gcloud command-line tool

```
gcloud compute http-health-checks create --help
```

The Google Cloud Platform Console

The REST API

HTTP(s) LB Basics : Timeouts and retries

HTTP(S) load balancing has a default response timeout of 30 seconds

HTTP(S) load balancing has a TCP session timeout of 10 minutes (600 seconds) by default

HTTP(S) load balancing retries failed GET requests, but does not retry failed POST requests.

HTTP(s) LB Basics : Illegal request handling

The load balancer blocks the following for HTTP/1.1 compliance:

- It cannot parse the first line of the request.
- A header is missing the : delimiter.
- Headers or the first line contain invalid characters.
- The content length is not a valid number, or there are multiple content length headers.
- There are multiple transfer encoding keys, or there are unrecognized transfer encoding values.
- There's a non-chunked body and no content length specified.
- Body chunks are unparseable.

HTTP(s) LB Basics : Illegal request handling

The load balancer also blocks the request if any of the following are true:

- The combination of request URL and headers is longer than about 15KB.
- The request method does not allow a body, but the request has one.
- The request contains an upgrade header.
- The HTTP version is unknown.

HTTP(s) LB Basics : Logging

Each HTTP(S) request is logged temporarily via Stackdriver Logging.

If you have been accepted into the Alpha testing phase, logging is automatic and does not need to be enabled.

HTTP(S) LB Basics : Notes and Restrictions

HTTP(S) load balancing supports the HTTP/1.1 100 Continue response.

- If your load balanced instances are running a public operating system image supplied by Compute Engine, then firewall rules in the operating system will be configured automatically to allow load balanced traffic.
- If you are using a custom image, you have to configure the operating system firewall manually. This is separate from the GCP firewall rule that must be created as part of configuring an HTTP(S) load balancer.
- Load balancing does not keep instances in sync.
- The HTTP(S) load balancer does not support sending an HTTP DELETE with a body to the load balancer.

SSL Load Balancer

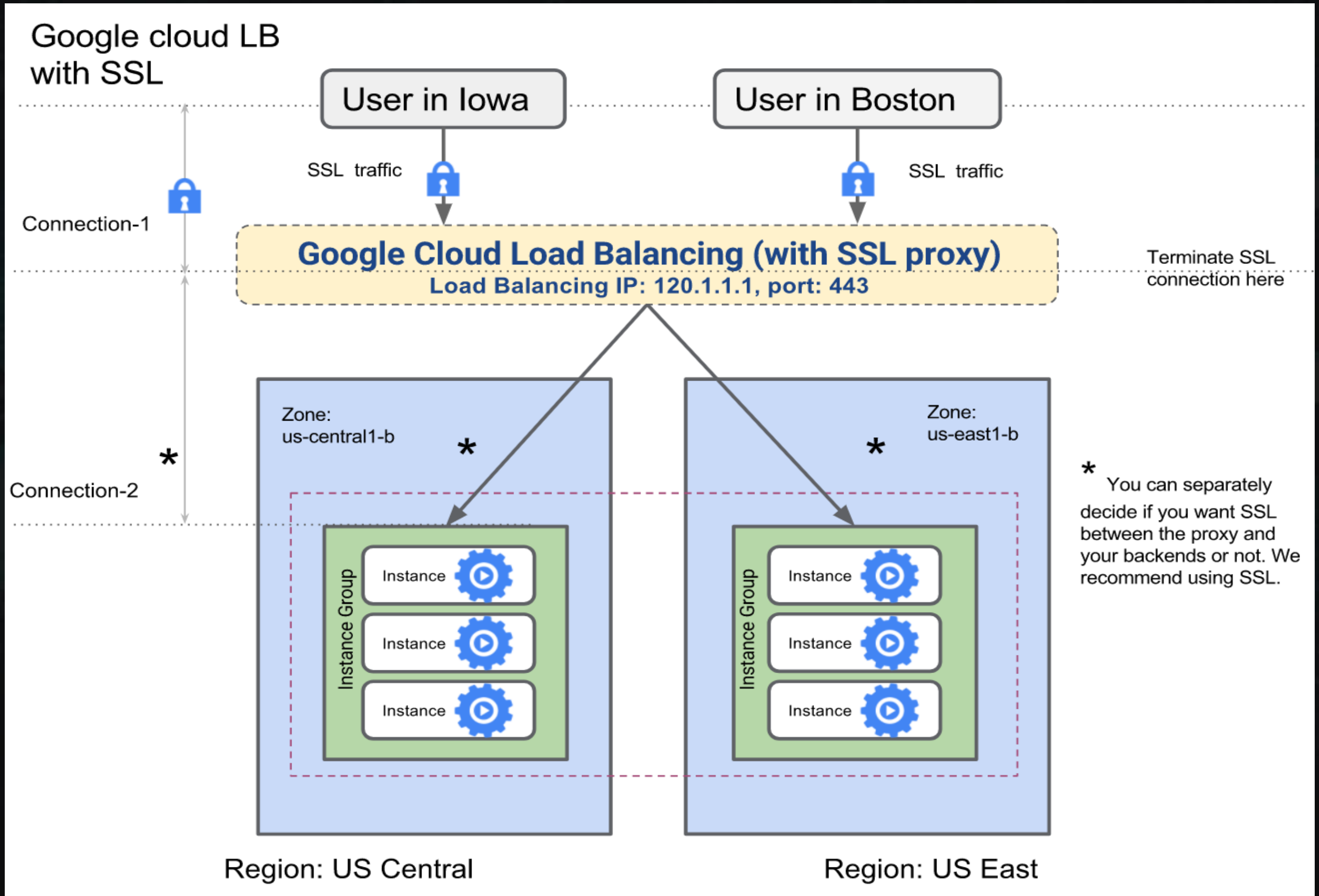
Global External Load Balancer

Global Ext. LB : SSL Load Balancing

Cloud SSL proxy is intended for non-HTTP(S) traffic. For HTTP(S) traffic, HTTP(S) load balancing is recommended instead.

SSL load balancer proxy at the global load balancing layer, SSL connections are terminated at the global layer then proxied balances connections across instance - > to the closest available instance via SSL or TCP

Global Ext. LB : SSL Load Balancing



* You can separately decide if you want SSL between the proxy and your backends or not. We recommend using SSL.

Global Ext. LB : SSL Load Balancing

Benefits

- Intelligent routing
- Better utilization of the virtual machine instances
- Certificate management
- Security patching
- SSL proxy supports the following ports: 25, 43, 110, 143, 195, 443, 465, 587, 700, 993, 995

Notes:

- While choosing to send the traffic over unencrypted TCP between the global load balancing layer and instances enables you to offload SSL processing from your instances, it also comes with reduced security between your global load balancing layer and instances and is therefore not recommended.

Global Ext. LB : SSL Load Balancing

Components

- Health Checking
- Backend Service
- SSL certificate and key
- Global forwarding rule
- .....

Note : Rejects invalid HTTP requests or responses

Negotiates HTTP/2 and SPDY/3.1

Spreads the request load more evenly among instances

HTTPS load balances each request separately, whereas SSL proxy sends all bytes from the same SSL or TCP connection to the same instance.

TCP Load Balancer

Global External Load Balancer

Global Ext. LB : TCP Load Balancing

arries almost same properties as SSL Proxy load Balancing

Please check the document if you want to check this in detail.

https://cloud.google.com/compute/docs/load-balancing/tcp-ssl/tcp-proxy

Regional external load balancing

- Network load balancing

Regional internal load balancing

- Internal load balancing

Regional Load Balancer

Google Compute Service

Internal Load Balancer

Regional Load Balancer

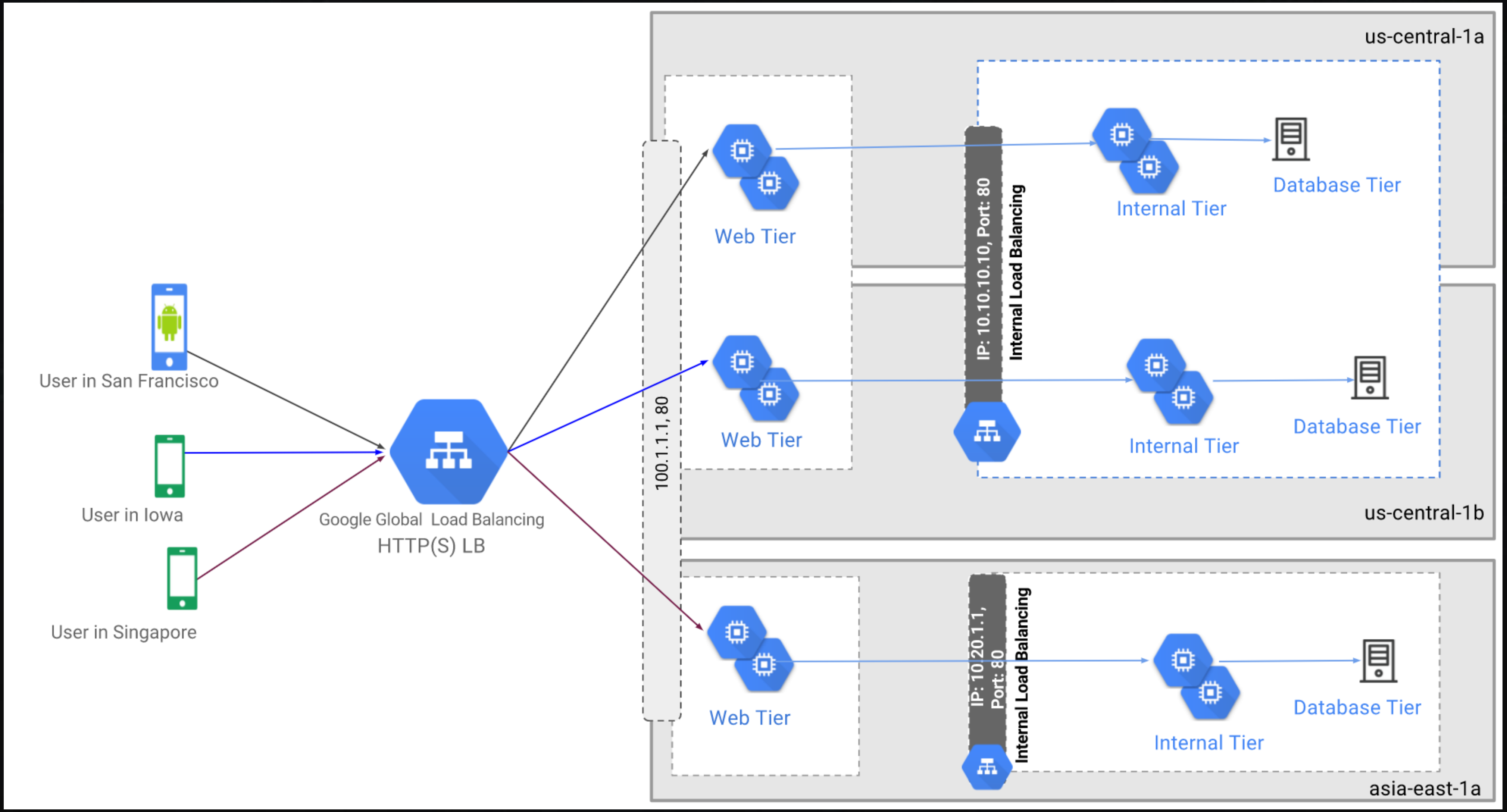
Regional LB: Internal Load Balancing

Internal load balancing enables you to run and scale your services behind a private load balancing IP address which is accessible only to instances internal to your Virtual Private Cloud (VPC).

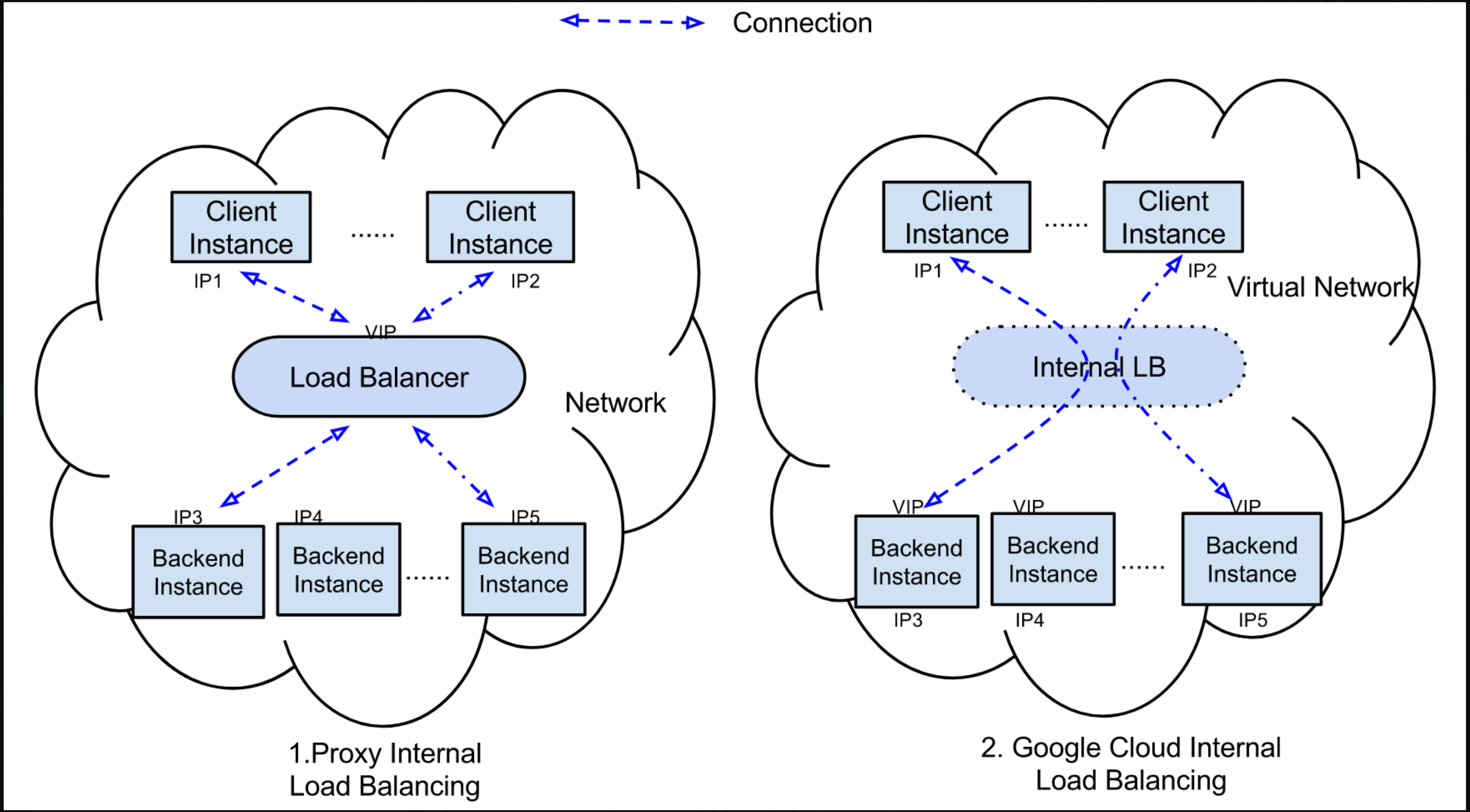
Your internal client requests stay internal to your VPC network and region, likely resulting in lowered latency since all your load-balanced traffic will stay within Google’s network. Overall, your configuration becomes simpler.

Internal load balancing works with auto mode VPC networks, custom mode VPC networks, and legacy networks. Internal load balancing can also be implemented with regional managed instance groups. This allows you to autoscale across a region, making your service immune to zonal failures

Regional LB: Internal Load Balancing



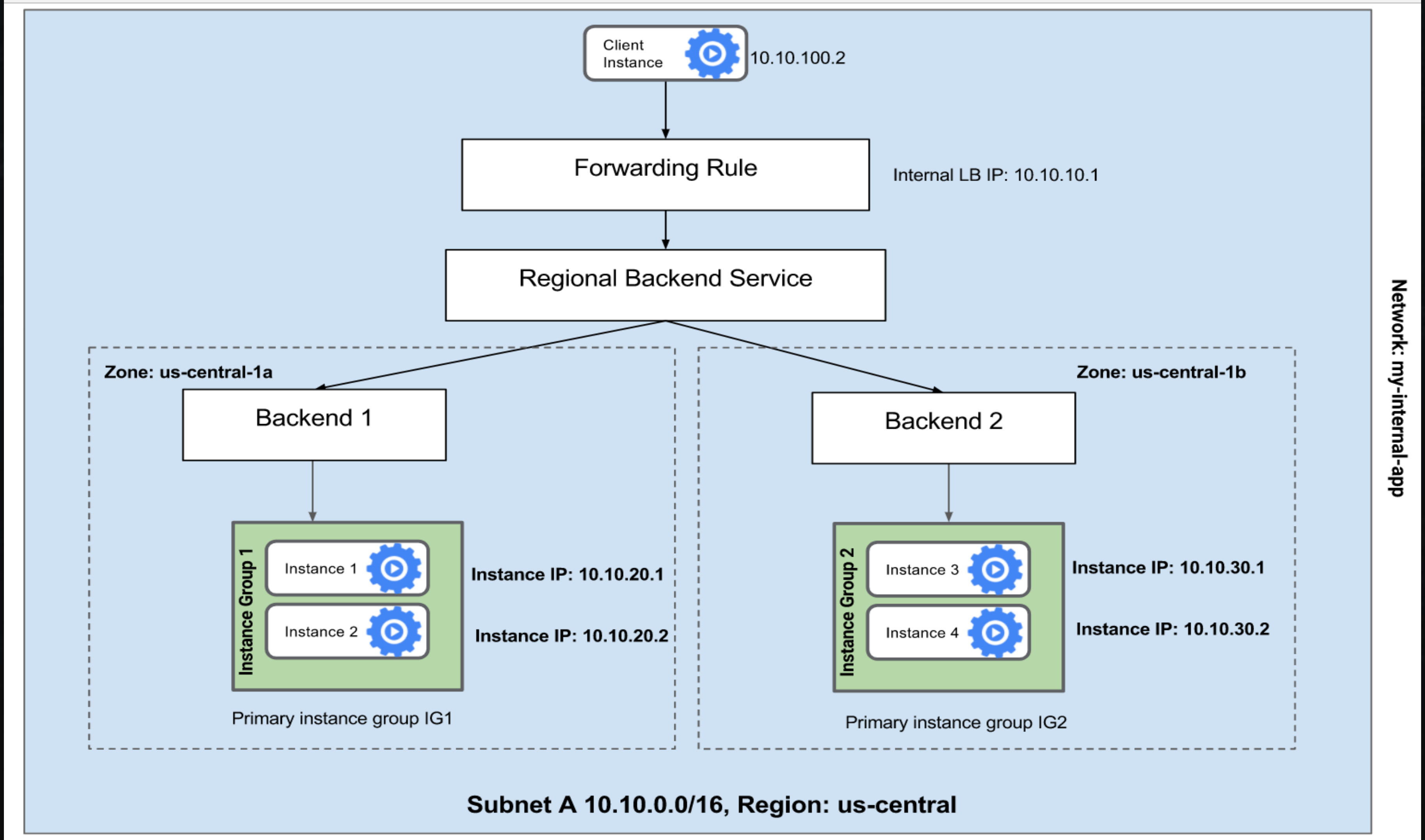
Regional LB: Architecture



1.Proxy Internal Load Balancing

2. Google Cloud Internal Load Balancing

Regional LB: Architecture



Internal LB : Selection Algorithm

By Default internal LB use 5-tuple hash

- client source IP
- client port
- destination IP (the load balancing IP)
- destination port
- protocol (either TCP or UDP).

If you want to control how the traffic directs to backend – You can use following options for Session Affinity

- Hash based on 3-tuple (Client IP, Dest IP, Protocol)
- Hash based on 2-tuple (Client IP, Dest IP)

Internal LB : Health Checking

Internal load balancing supports four types of health checks:

- TCP health checks
- SSL (TLS) health checks
- HTTP health checks
- HTTPS health checks

Internal LB : Other Considerations

Restrictions

- Client instances and backend instances have to be on Google Cloud Platform. Sending traffic from clients on-premises (outside of GCP) to the Internal load balancer in GCP is not supported in this release.
- The Internal load balancer IP cannot be the next-hop IP of a manually configured route.
- You cannot send traffic through a VPN tunnel to your load balancer IP.

Limits

- A maximum number of 50 Internal load balancer forwarding rules is allowed per network.
- A maximum of 250 backend is allowed per Internal load balancer forwarding rule

External (Regional) Load Balancer

Regional Load Balancer

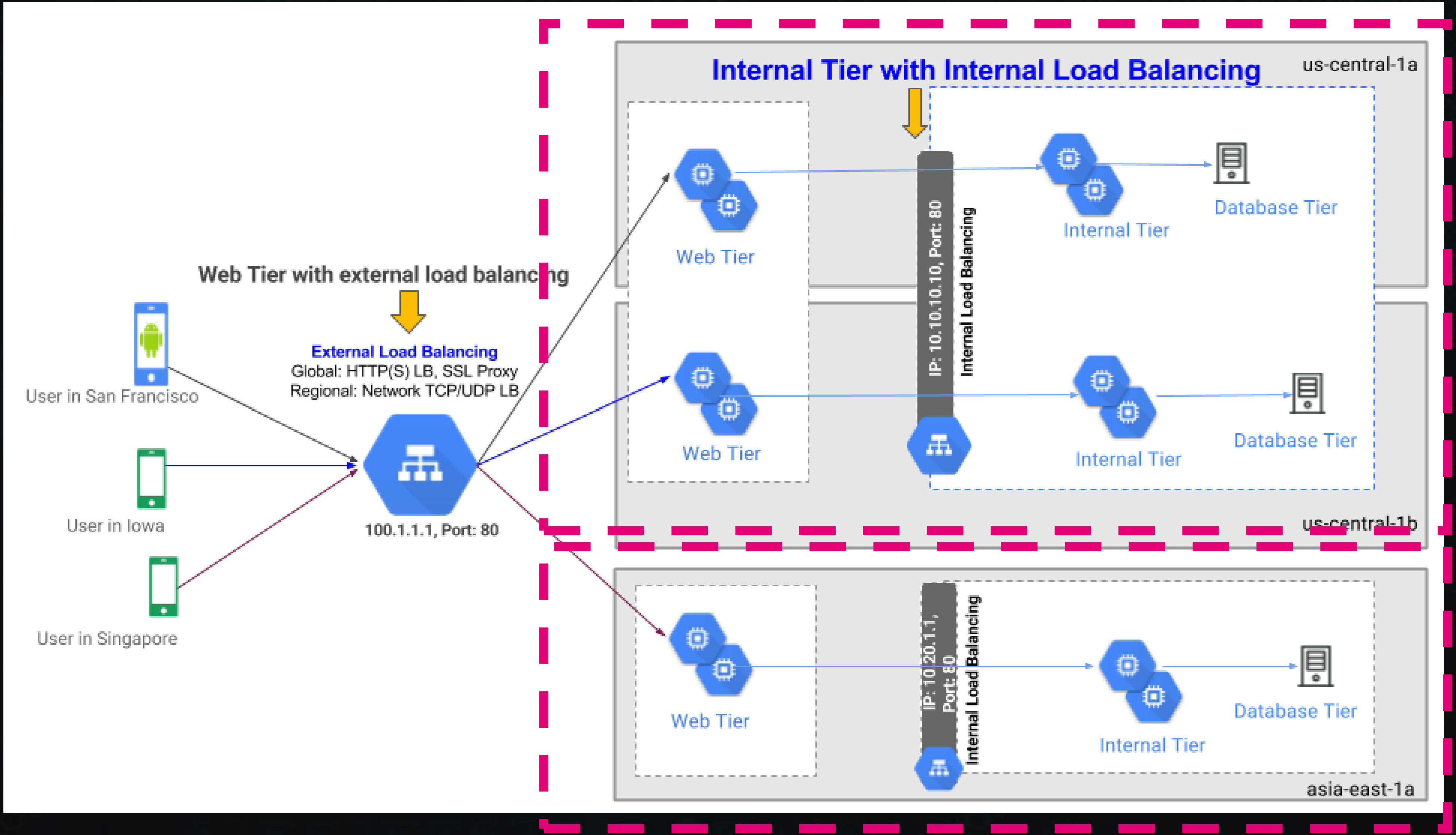
Regional External Load Balancer : Network LB

Network load balancing allows you to balance load of your systems based on incoming IP protocol data, such as address, port, and protocol type.

Network load balancing uses forwarding rules that point to target pools, which list the instances available for load balancing and define which type of health check that should be performed on these instances

Network load balancer is a pass-through load balancer. It does not proxy connections from clients.

Regional External Load Balancer : Network LB



Network LB : Consideration

Load Distribution Algorithm

Target pools

Session affinity

Health Checking

Firewall rules and Network load balancing

Load balancer Pricing – All Load balancer

All Regions	
Hourly service charge	$0.025 (5 rules included) $0.010 per additional rule
Per GB of data processed	$0.008

Load balancer - Connection Draining

- You can enable connection draining on backend services to ensure minimal interruption to your users when an instance is removed from an instance group, either manually or by an autoscaler.

- To enable connection draining, you set a timeout duration during which the backend service preserves existing sessions being handled by endpoints on an instance that will be removed.

- The backend service preserves these sessions until the timeout duration has elapsed, allowing user sessions to gracefully terminate, but preventing new connections to the instance

- After the timeout duration is reached, the instance is terminated and all remaining connections are forcibly closed. You can set a timeout duration between 1 to 3600 seconds

Auto Scaling

Google Compute Engine

GCE : Auto Scaling

- Compute Engine offers autoscaling to automatically add or remove virtual machines from an instance group based on increases or decreases in load.
- This allows your applications to gracefully handle increases in traffic and reduces cost when the need for resources is lower.
- You just define the autoscaling policy and the autoscaler performs automatic scaling based on the measured load.

GCE : Auto Scaling Policies

- CPU utilization
- Load balancing serving capacity
- Stackdriver Monitoring metrics
- Google Cloud Pub/Sub queuing workload (Alpha)

GCE : Auto Scaling Specifications

- Autoscaling only works with managed instance groups. Unmanaged instance groups are not supported.
- Do not use Compute Engine autoscaling with managed instance groups that are owned by Google Container Engine. For Google Container Engine groups, use Cluster Autoscaling instead.

Look for the gke prefix in the managed instance group name.

For example, gke-test-1-3-default-pool-eadji9ah.

Google Compute Service

Load Balancer & Auto Scaling

Demo Next ..

Google Cloud Platform