

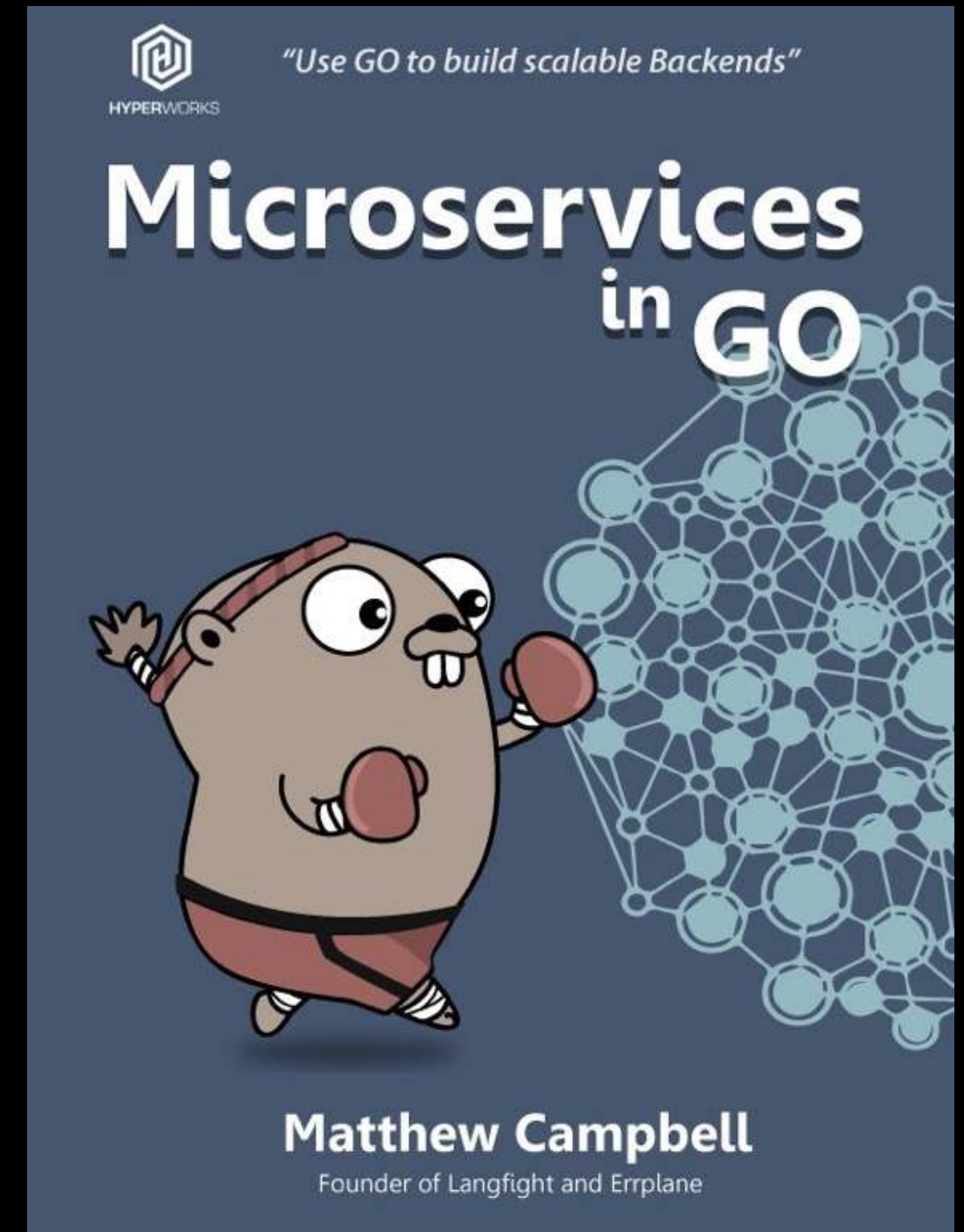
Breaking Prometheus

***Scaling prometheus
to a million machines***



About Me

- *Technical Lead Digital Ocean*
- *Microservices in GO Book*
- *Lives in Bangkok*



Dark Days

- **Graphite**
- **InfluxDB**
- **OpenTSDB (*sigh*)**

Manual Prometheus

```
scrape_configs:
- job_name: microservice1
  target_groups:
  - targets: ['server1:8083']
  - targets: ['server2:8083']
  - targets: ['server3:8083']
  - targets: ['server4:8083']
  - targets: ['server5:8083']

- job_name: otherjob
  target_groups:
  - targets: ['server3:8086']
  - targets: ['server4:8087']
  - targets: ['server5:8088']
```

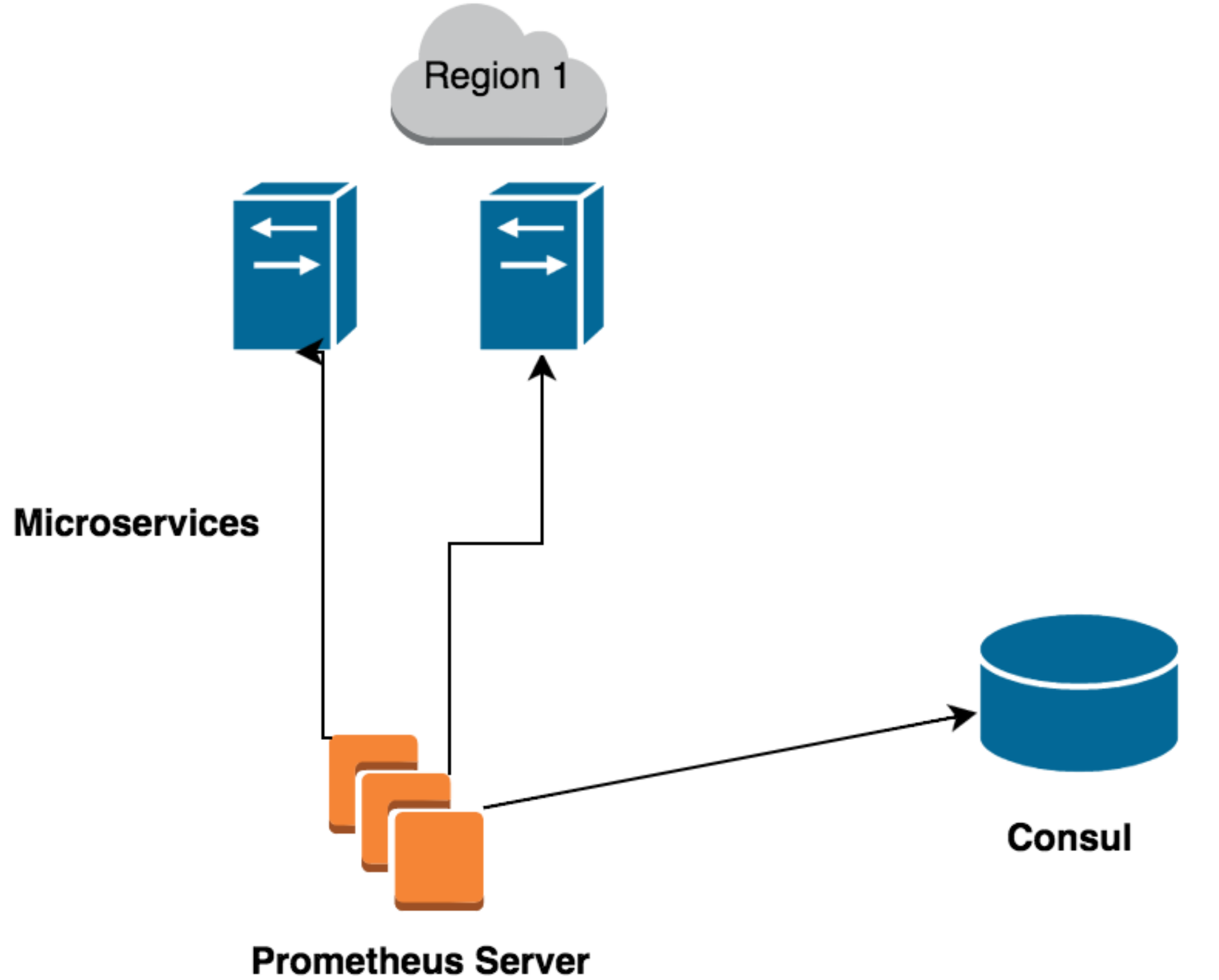
Manual Issues

- Lots of Prometheus servers
- Mismatched versions
- New machines = update config
- Missing matches



Consul + Prometheus = Peanut butter + Jelly





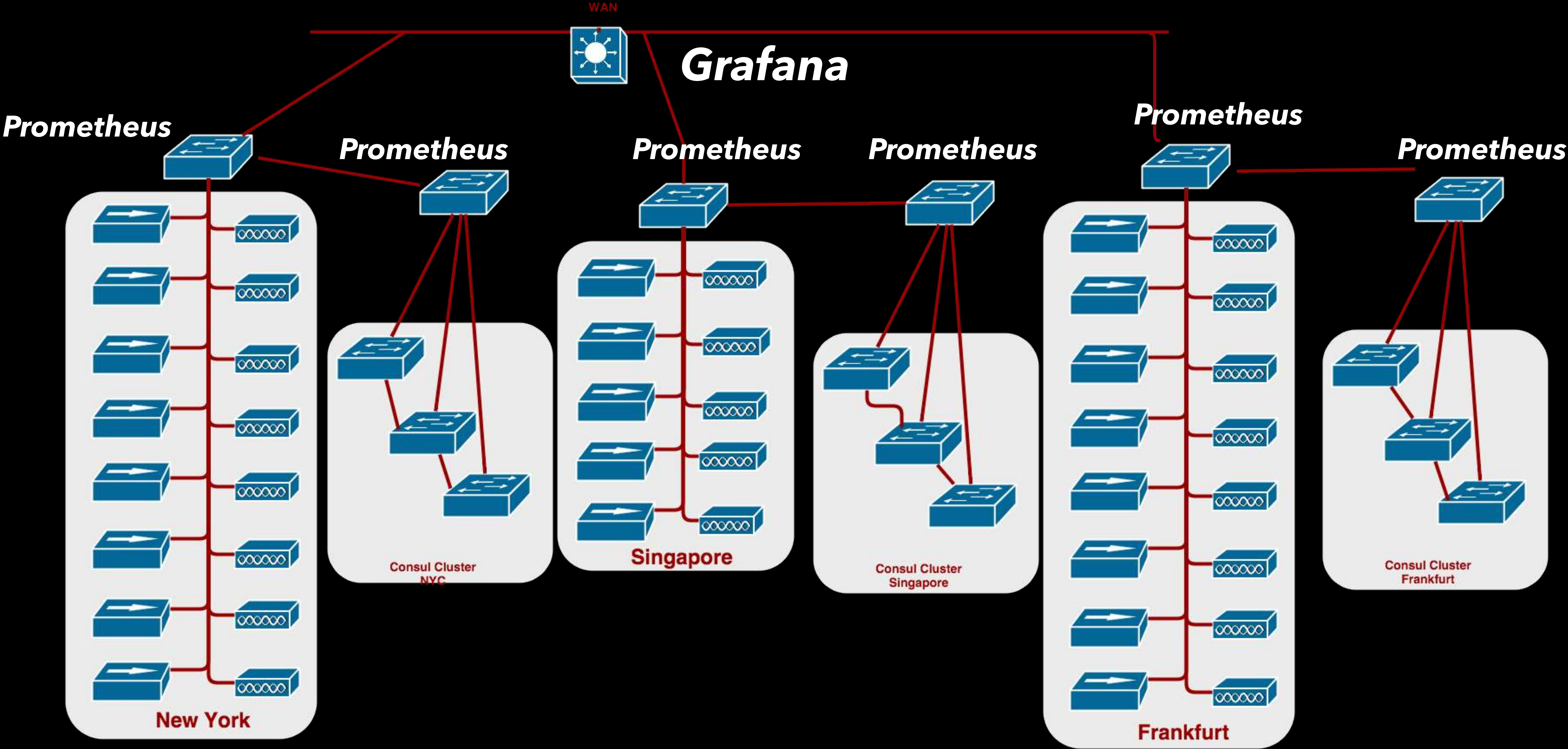
- `job_name: pandora-exporter_nbg1`
`target_groups:`
`dns_sd_configs:`
`consul_sd_configs:`
 - `server: 127.0.0.1:8500`
`datacenter: nbg1`
`services:`
 - `pandora-exporter`
- `job_name: service2-exporter_fra1`
`target_groups:`
`dns_sd_configs:`
`consul_sd_configs:`
 - `server: 127.0.0.1:8500`
`datacenter: fra1`
`services:`
 - `service2-exporter`



Stage 2:

***Datcenter wide
10,000s of nodes***

Prometheus Per Region



- ***Modify retention windows***
- ***Drop metrics from node_exporter***
- ***Larger and larger machines***

I/O Problems



Tuning options

storage.local.retention

storage.local.memory-chunks

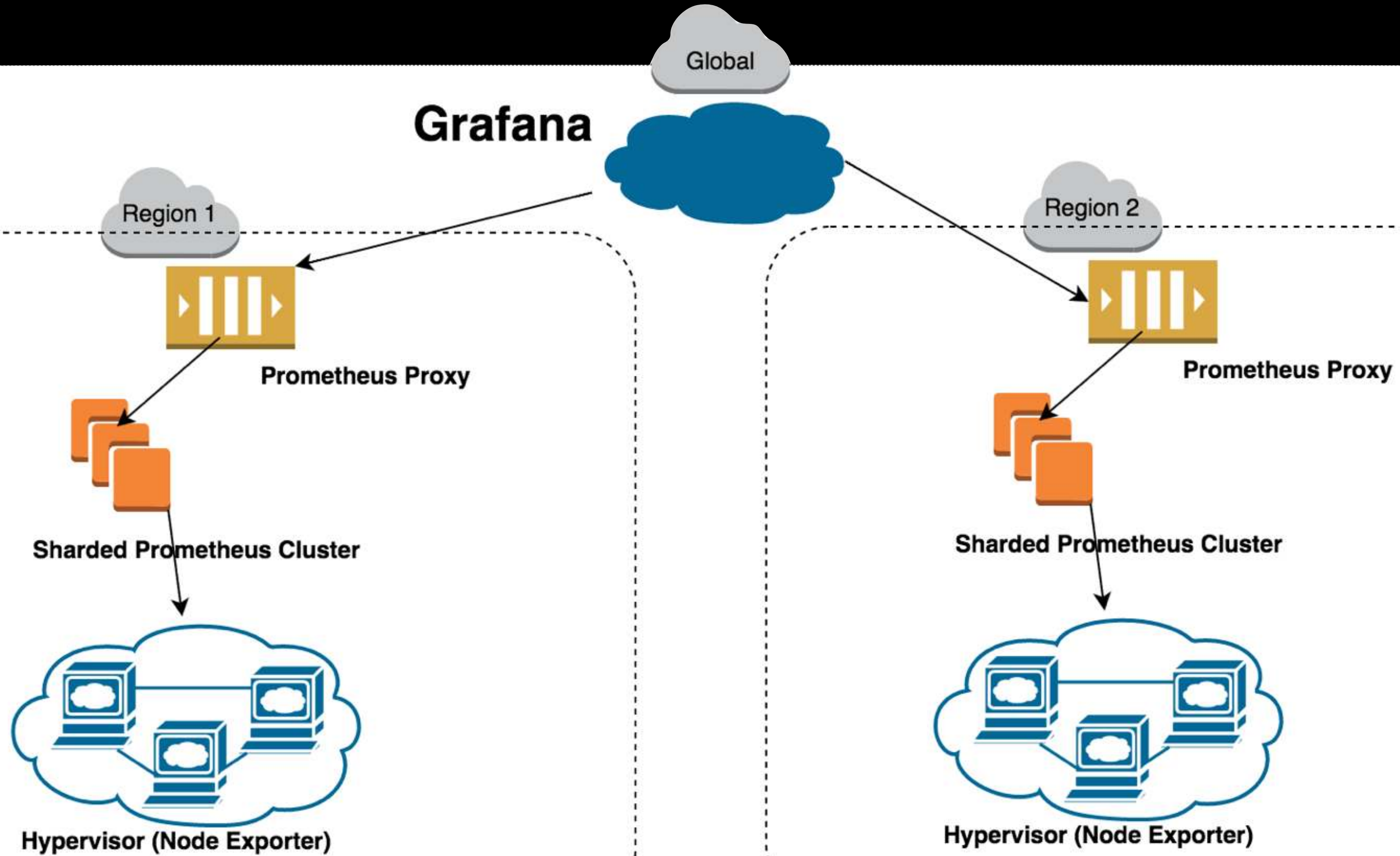
storage.local.max-chunks-to-persist

storage.local.checkpoint-interval

storage.local.checkpoint-dirty-series-limit

A close-up photograph of a shattered glass pane. The glass is broken into many small, sharp fragments, with a prominent starburst pattern of cracks radiating from a central point on the right side. A diamond-shaped mesh, likely a security screen, is overlaid on the entire image. The background is dark and out of focus.

Sharding

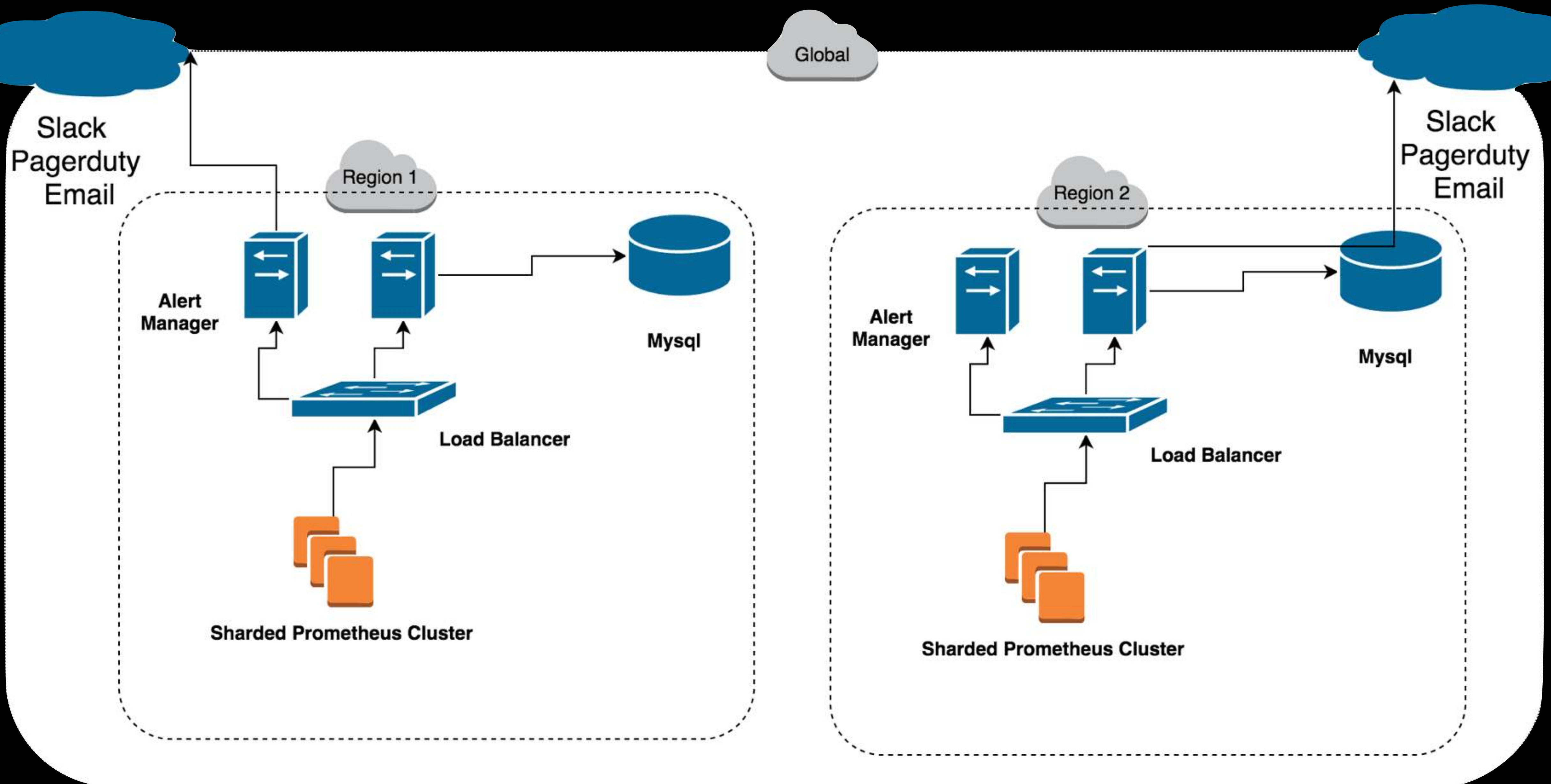


Prometheus Proxy

```
rate(node_cpu{instance="server12345.digitalocean.com:9100"}[2m])
```

Shard on red

Alert Manager



Shard Problems

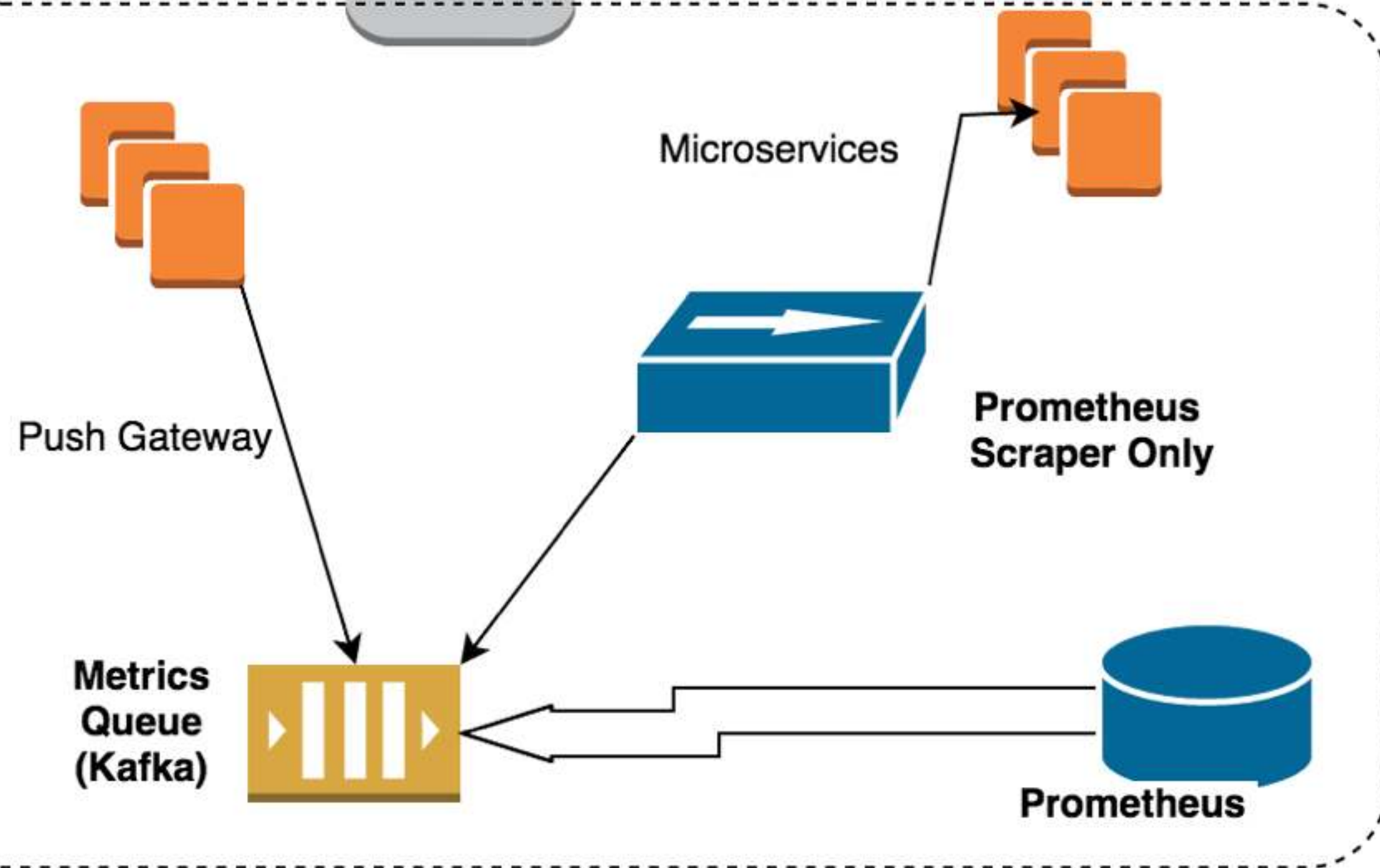
- ***Shard redistribution***
- ***Over provisioning***
- ***Data loss***
- ***Limited data windows***



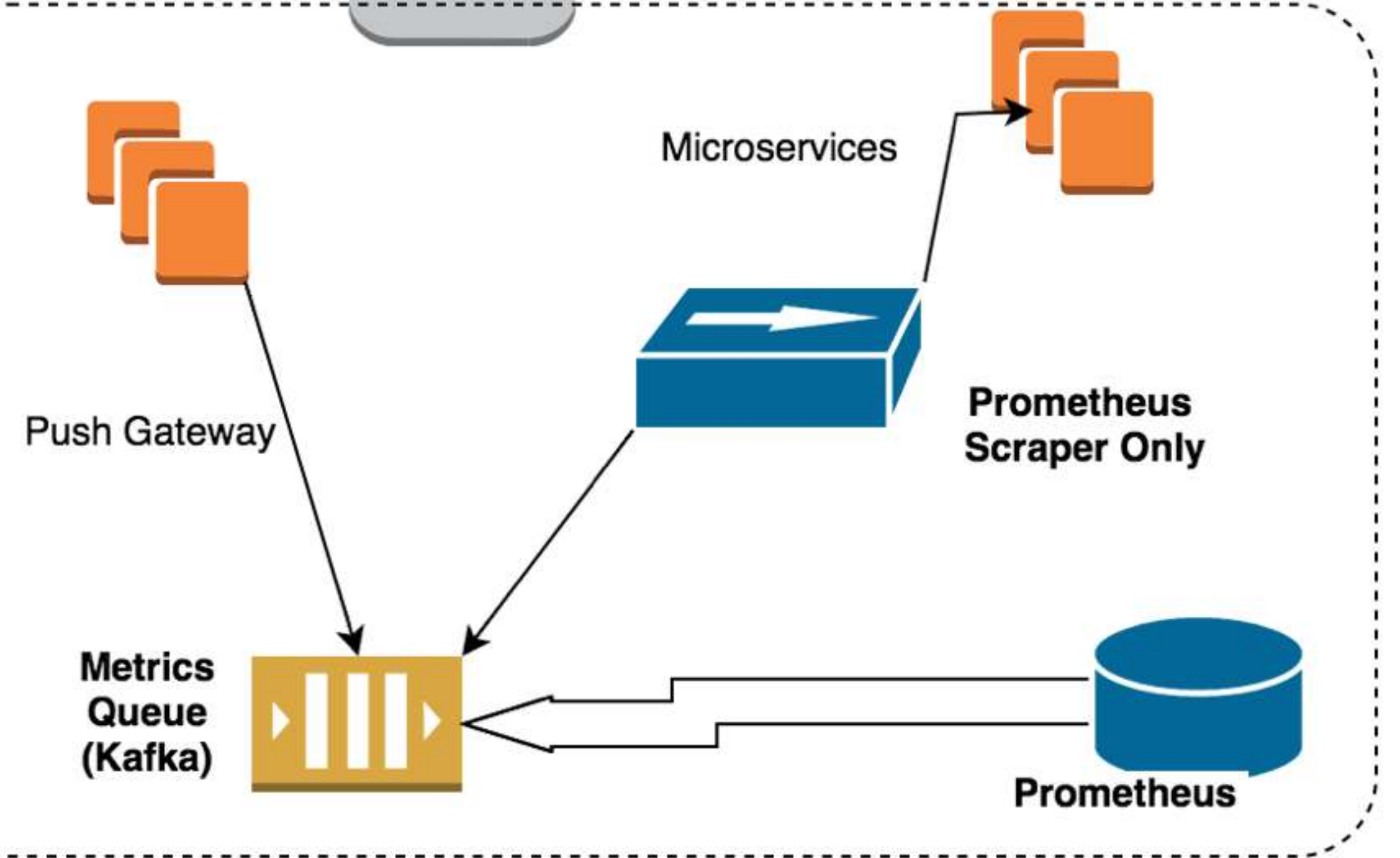
Kafka Bus

All Regions

Region 1



Region 2





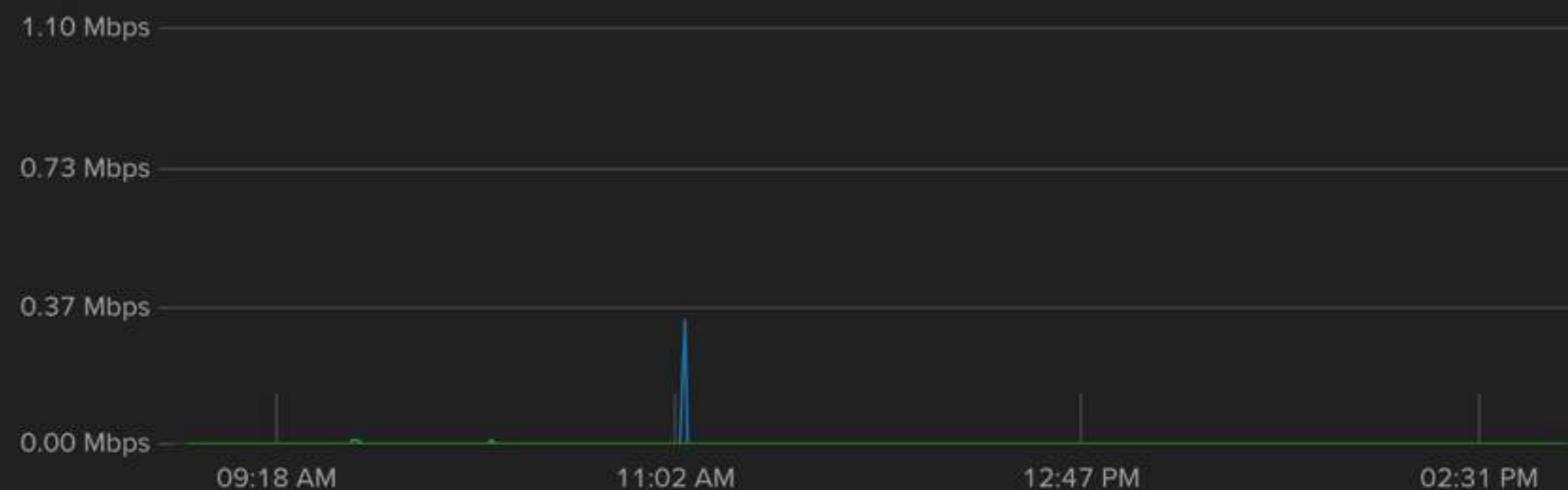
Filter by Droplet name or IP

NYC1

Last updated just now

6 hours 24 hours 7 days 30 days

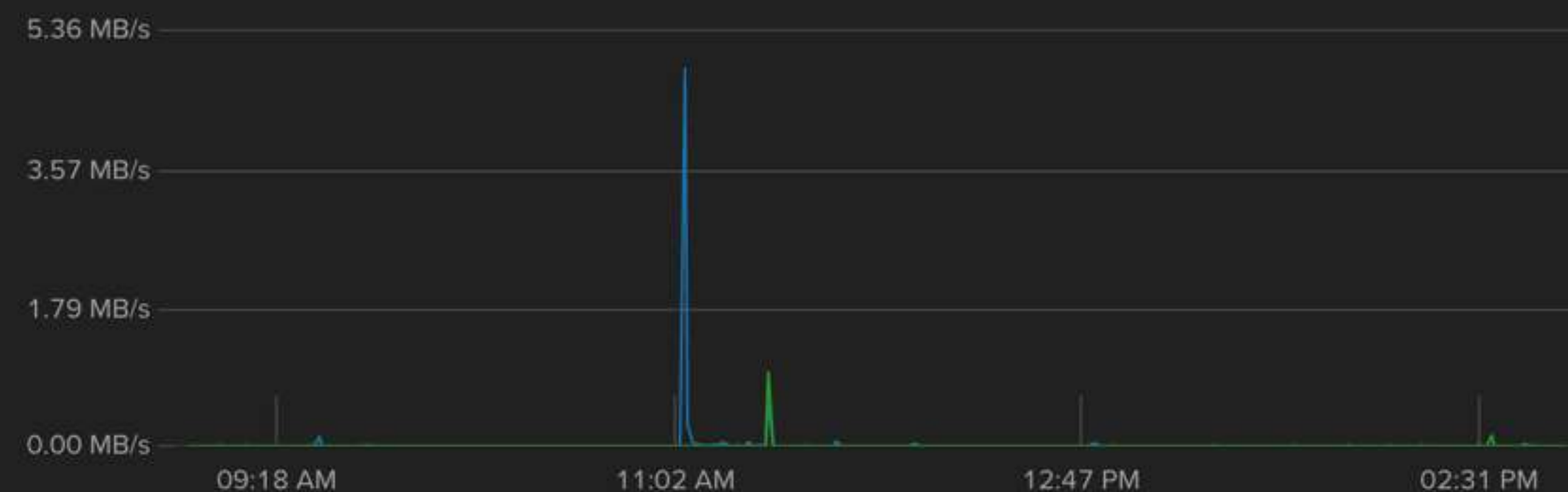
Bandwidth - Inbound



Bandwidth - Outbound



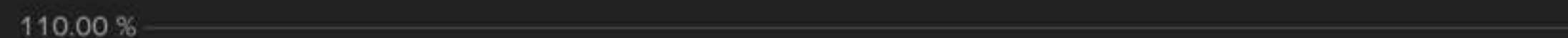
Disk - Read



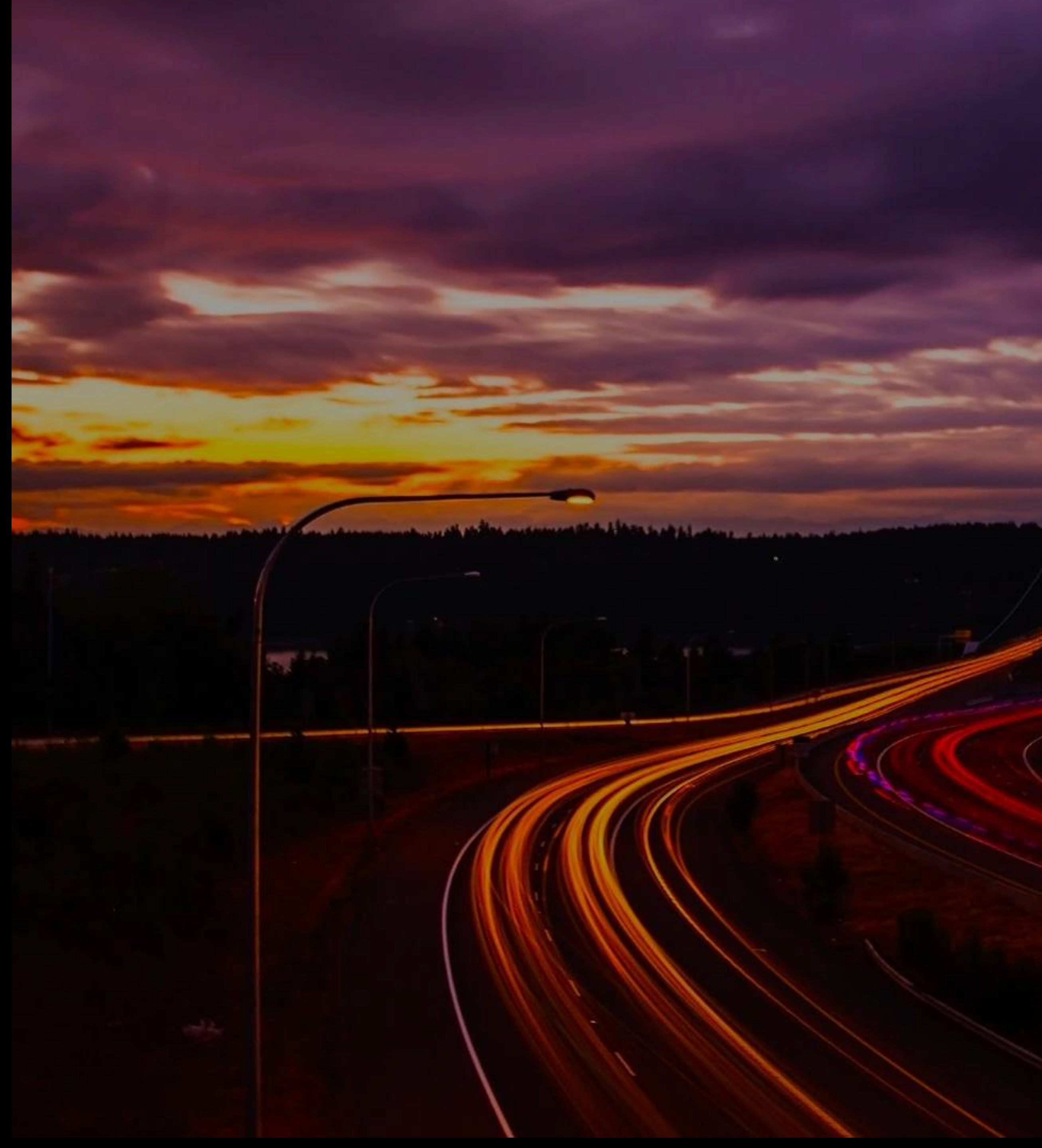
Disk - Write



CPU



Million VMs



Digital Ocean Agent

- ***Installable Metrics Agent***
- ***Authenticated Push Gateway***
- ***“Reverse node exporter”***



Query Api



- *Customer facing API*
- *GRPC / Json*
- *Authenticated per customer*
- *Prometheus queries*



Introducing Vulcan

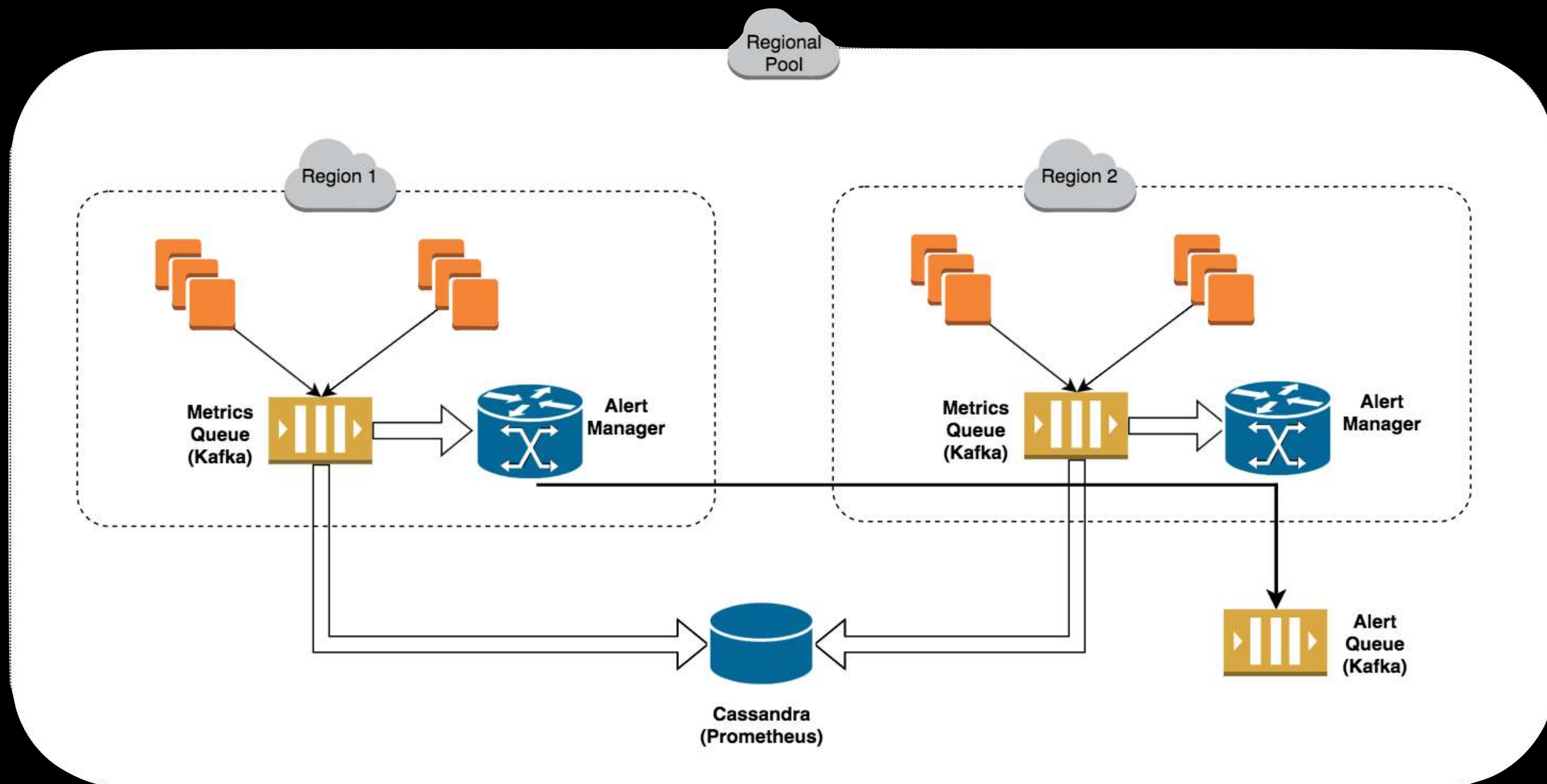
<https://github.com/digitalocean/vulcan>

Vulcan



- ***Prometheus Api***
- ***Cassandra storage***
- ***Kafka incoming***
- ***Standard Scrapers***

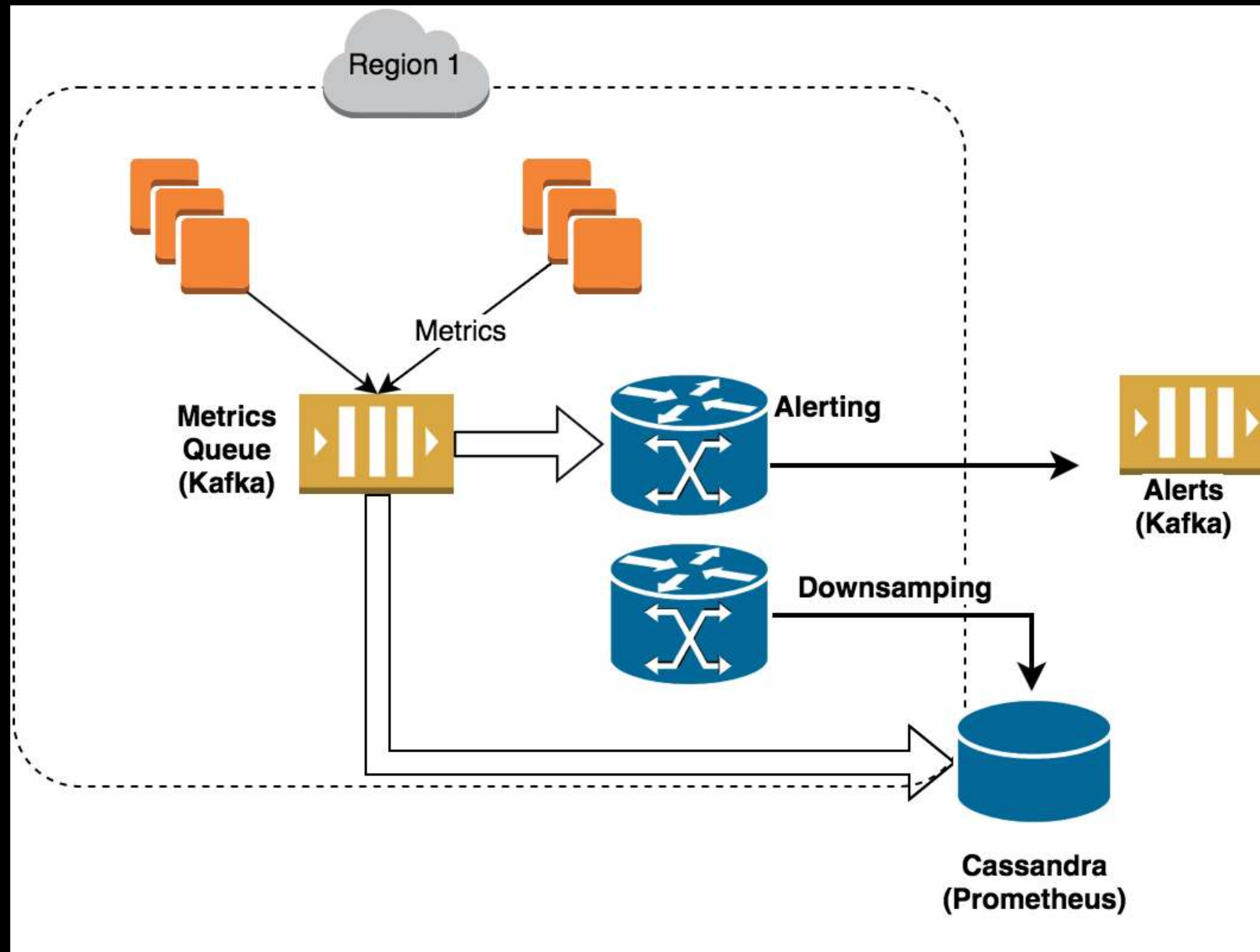
Cassandra Store



Downsampling

- ***In memory shared promethues***
- ***Driven from data in kafka***
- ***Reusable for Alerting***

Downsampling / Alerting



Future

- ***New Scrape Sources (Kafka)***
- ***Per series expiry TTLs***
- ***Plugin Storage Model (In Memory)***
- ***Alerting High Availability***

Questions?

Matthew Campbell

hyper@hyperworks.nu

@kanwisher

github.com/mattkanwisher

We're Hiring!

