Alertmanager
on its way to high availability

Frederic Branczyk

Software Engineer at CoreOS

Prometheus/Alertmanager/Kubernetes

@brancz

What that means?

- From alert to notification
- High availability contract
- High availability implementation
- Implications on operating HA Alertmanager

Where does CoreOS fit in?

- Automating Monitoring infrastructure
- Prometheus + Kubernetes

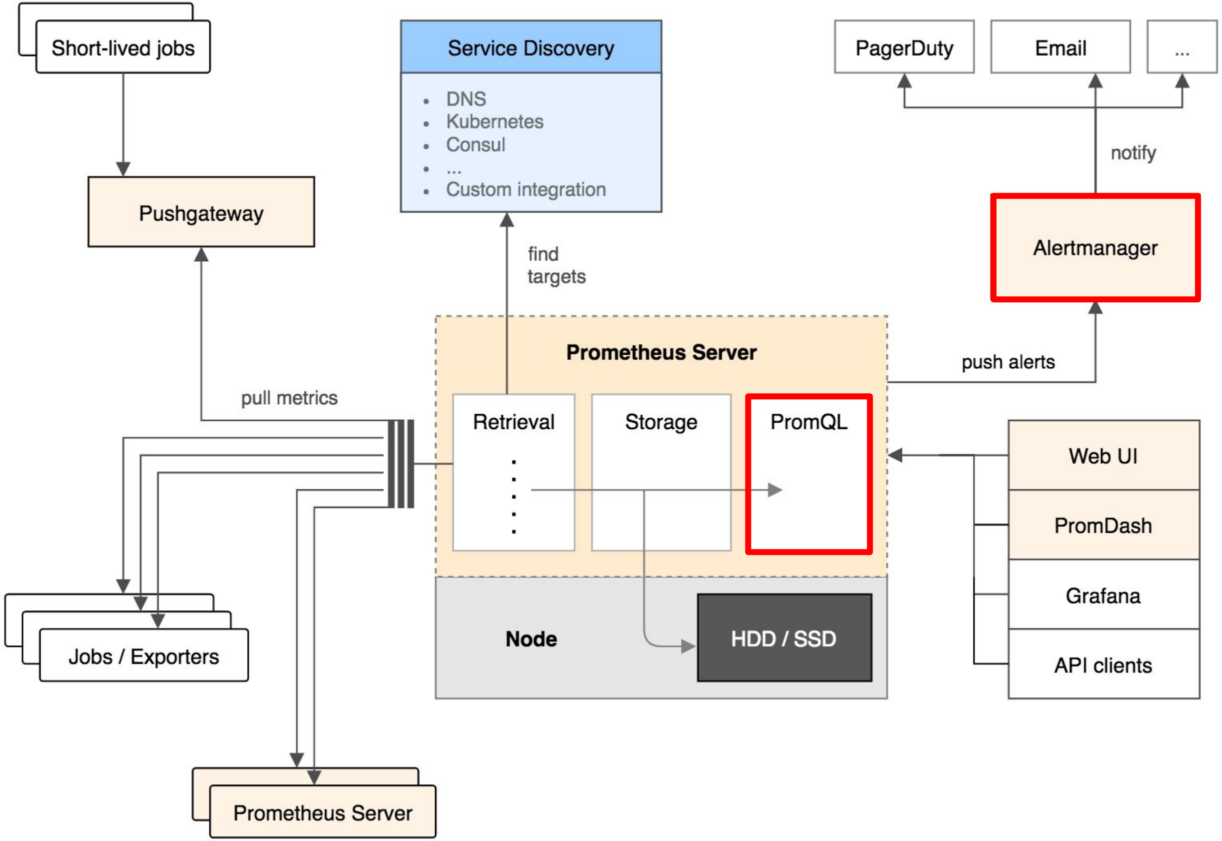
TECTONIC

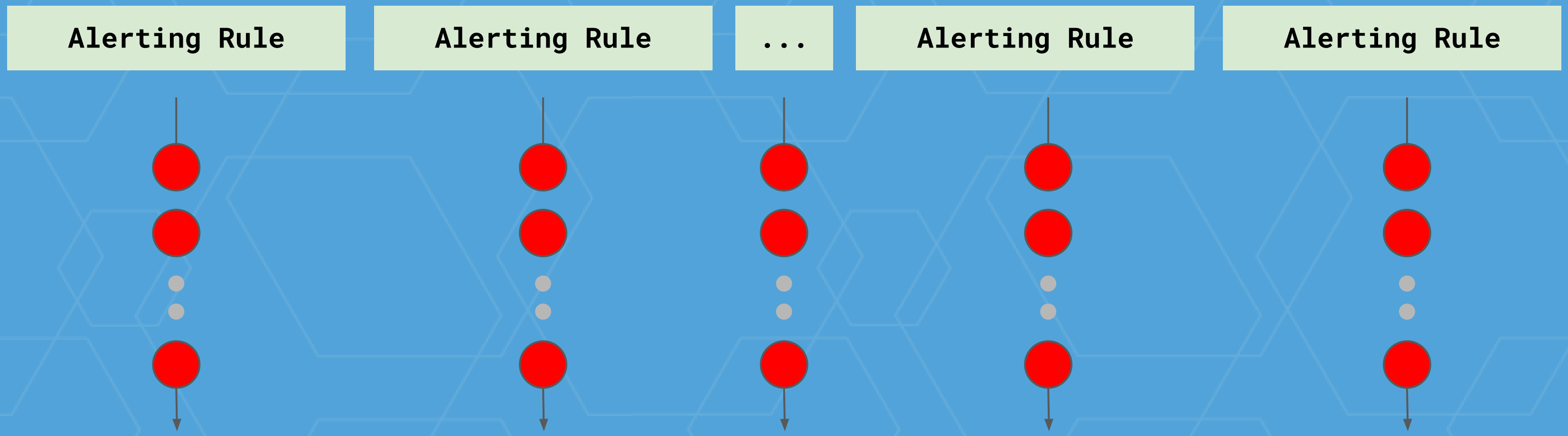
by CoreOS

Alertmanager Features

- Receives and groups alerts
- Deduplicates alerts
- Sends notifications to providers
    - Pagerduty, email, Slack, etc.
- Silencing

Prometheus & Alertmanager





Alerting Rule

Alerting Rule

...

Alerting Rule

Alerting Rule

```
04:11   hey, HighLatency,   service=”X”, zone=”eu-west”, path=/user/profile, method=GET
04:11   hey, HighLatency,   service=”X”, zone=”eu-west”, path=/user/settings, method=GET
04:11   hey, HighLatency,   service=”X”, zone=”eu-west”, path=/user/settings, method=GET
04:11   hey, HighErrorRate, service=”X”, zone=”eu-west”, path=/user/settings, method=POST
04:12   hey, HighErrorRate, service=”X”, zone=”eu-west”, path=/user/profile, method=GET
04:13   hey, HighLatency,   service=”X”, zone=”eu-west”, path=/index, method=POST
04:13   hey, CacheServerSlow, service=”X”, zone=”eu-west”, path=/user/profile, method=POST
        . . .
04:15   hey, HighErrorRate, service=”X”, zone=”eu-west”, path=/comments, method=GET
04:15   hey, HighErrorRate, service=”X”, zone=”eu-west”, path=/user/profile, method=POST
```

Grouped in one notification

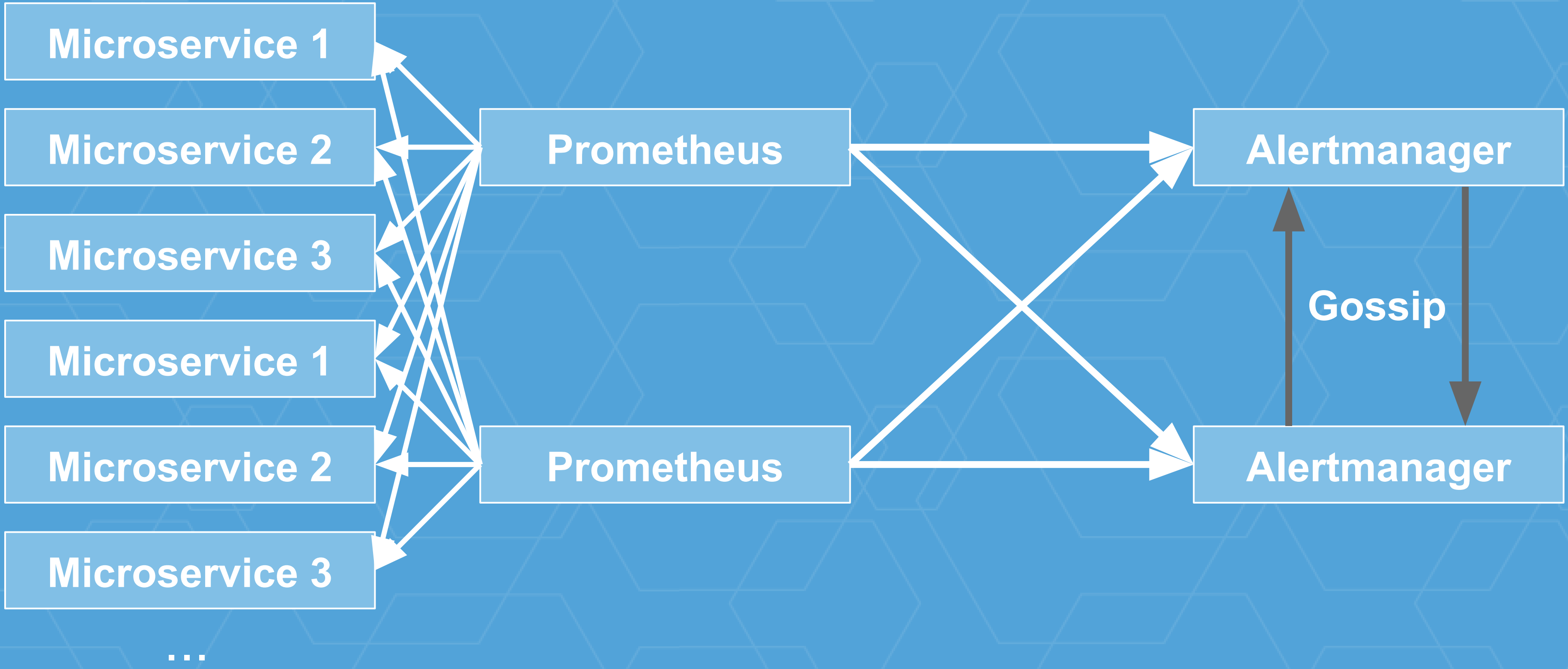
- 3 x HighLatency
- 10 x HighErrorRate
- 2 x CacheServerSlow
- (+individual Alerts)

Alertmanager

Reliably sends notifications

High Availability

Infrastructure Scaling Story



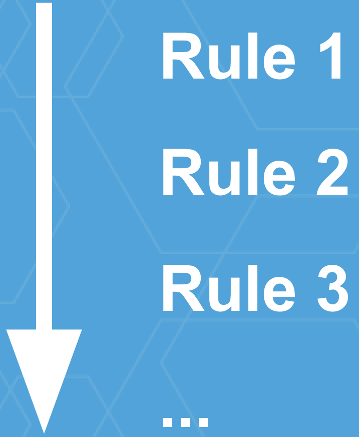
Why decoupled?

- Keep Prometheus alerting simple
- High availability of Prometheus
- No state sharing between Prometheus

Example Alerting Rule

```
ALERT NoLeader
IF etcd_has_leader == 0
FOR 10m
LABELS {
  severity = "warning"
}
ANNOTATIONS {
  summary = "etcd no leader",
  description = "etcd instance has no leader",
}
```

Alert Evaluation in Prometheus



Rule 1

Rule 2

Rule 3

...

- Evaluate Rule/Alert
- Fire alert against Alertmanager

Repeat in *rule evaluation interval*

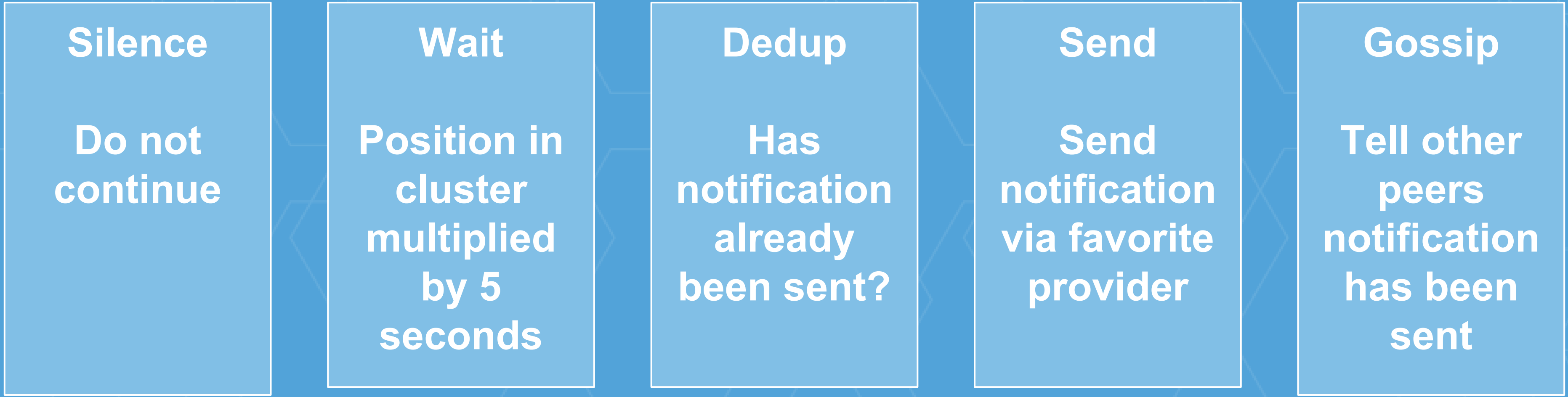
Simple configuration

```
global:
  resolve_timeout: 5m

route:
  group_by: ['job']
  group_wait: 10s
  group_interval: 10s
  repeat_interval: 1h
  receiver: 'webhook'
receivers:
- name: 'webhook'
  webhook_configs:
  - url: 'http://127.0.0.1:5001/'
```

- Resolve alerts in 5m
- Group by *job* label
- Group for 10 seconds
- Send via webhook receiver

Notification Pipeline



Silence

Do not continue

Wait

Position in cluster multiplied by 5 seconds

Dedup

Has notification already been sent?

Send

Send notification via favorite provider

Gossip

Tell other peers notification has been sent

What is gossiped?

- Yes
  - Sent notifications
  - Silences
- No
  - Received alerts

How? CRDTs!

- Conflict-free replicated data type
- Associativity (a+(b+c)=(a+b)+c)
- Commutativity (a+b=b+a)
- Idempotence (a+a=a)
- Well suited for AP systems

How? mesh by Weaveworks

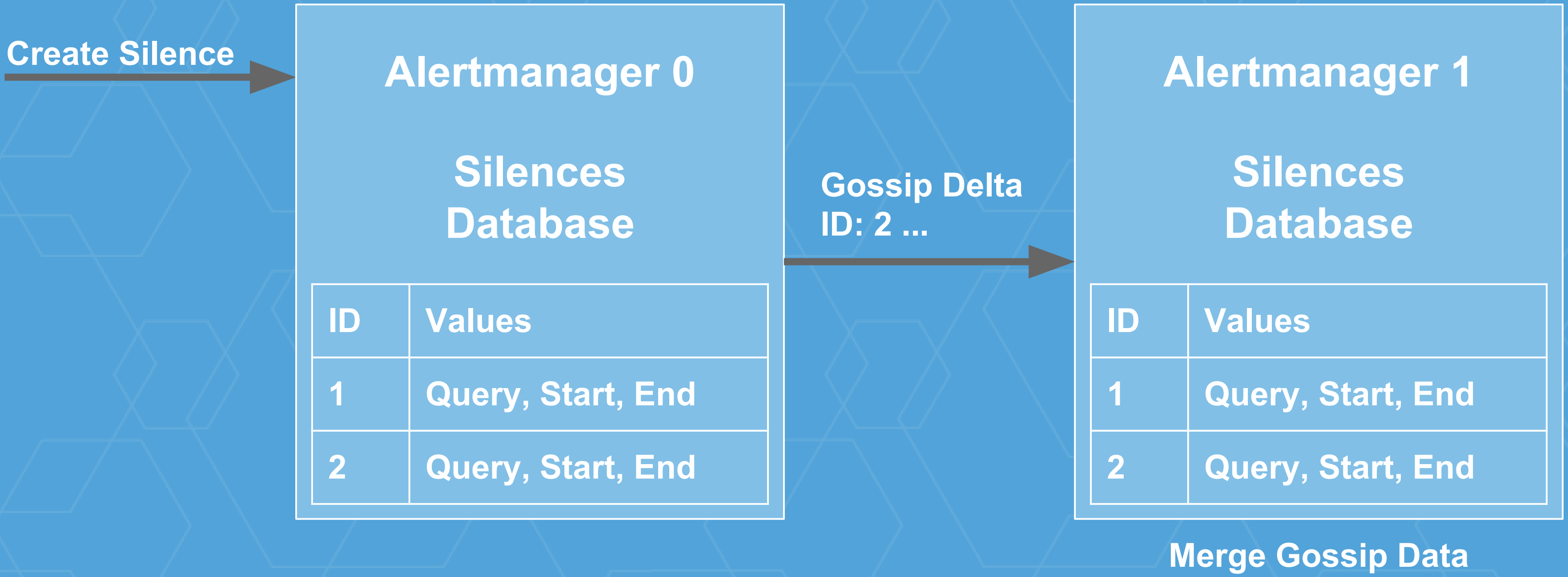
- Eventually consistent
- LWW-element-set
- Mergeable log of records
- Merges based on UID
    - On conflict latest timestamp wins

Why not etcd?

- Simple operation
  - Less moving pieces
  - Single binary
- Want: AP not CP

Silences

Create Silences



Create Silence

Alertmanager 0

Silences Database

ID	Values
1	Query, Start, End
2	Query, Start, End

Gossip Delta
ID: 2 ...

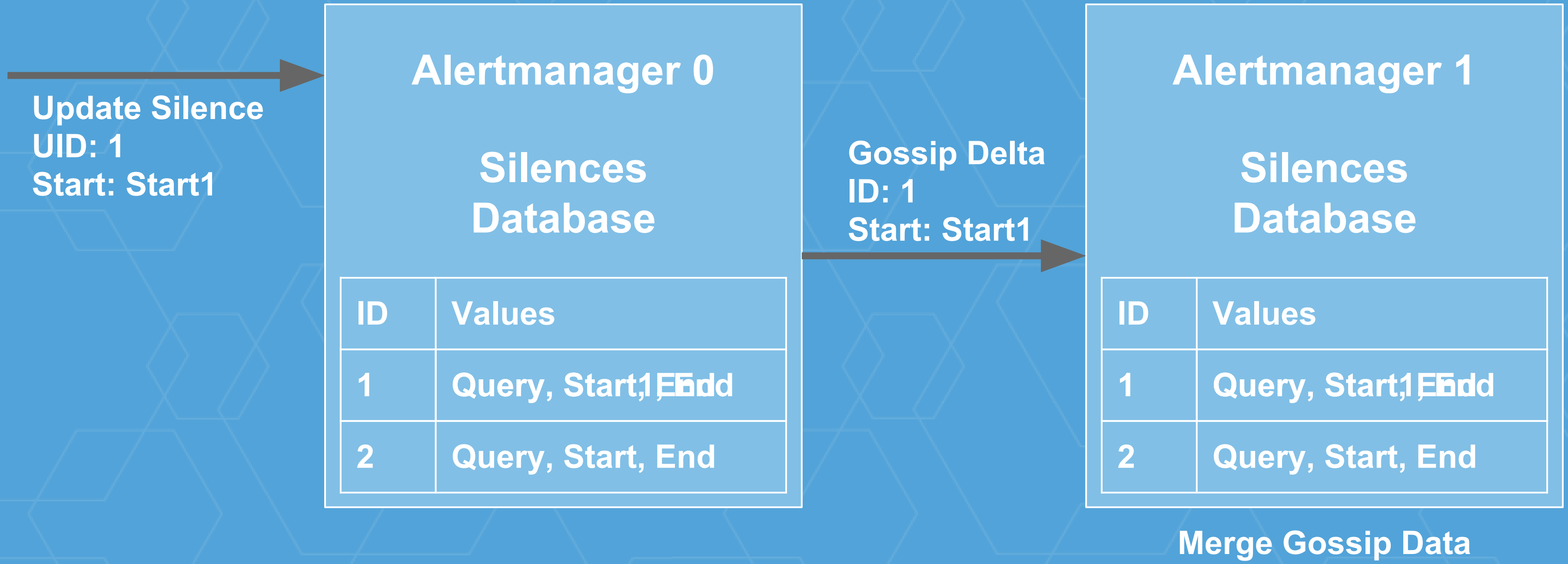
Alertmanager 1

Silences Database

ID	Values
1	Query, Start, End
2	Query, Start, End

Merge Gossip Data

Update Silences



Update Silence
UID: 1
Start: Start1

Alertmanager 0

Silences Database

ID	Values
1	Query, Start, End Query, Start1, End
2	Query, Start, End

Gossip Delta
ID: 1
Start: Start1

Alertmanager 1

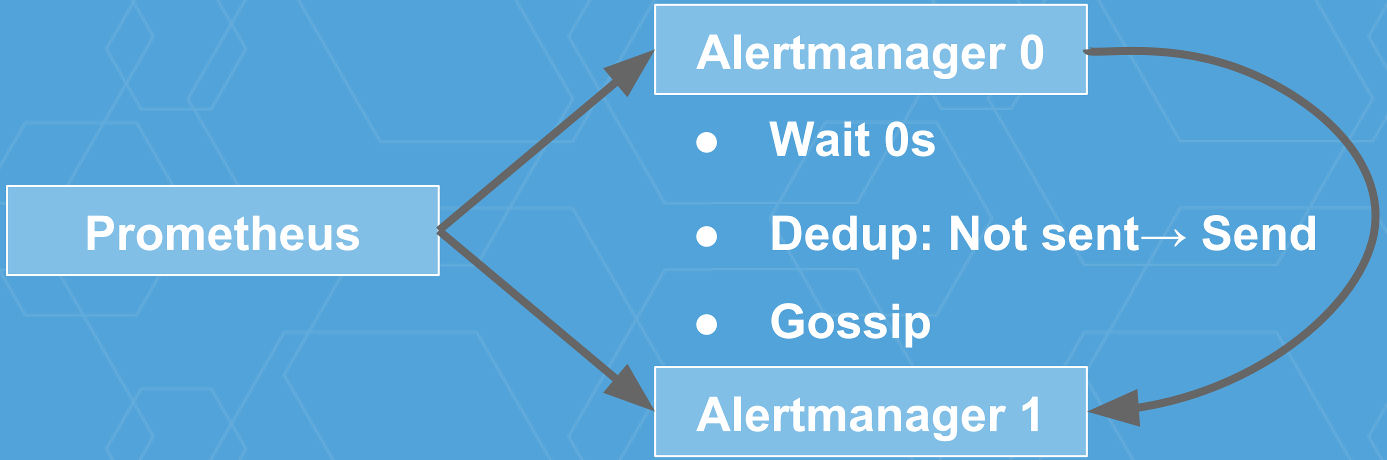
Silences Database

ID	Values
1	Query, Start, End Query, Start1, End
2	Query, Start, End

Merge Gossip Data

Notification Log

Non silenced alert example



Prometheus

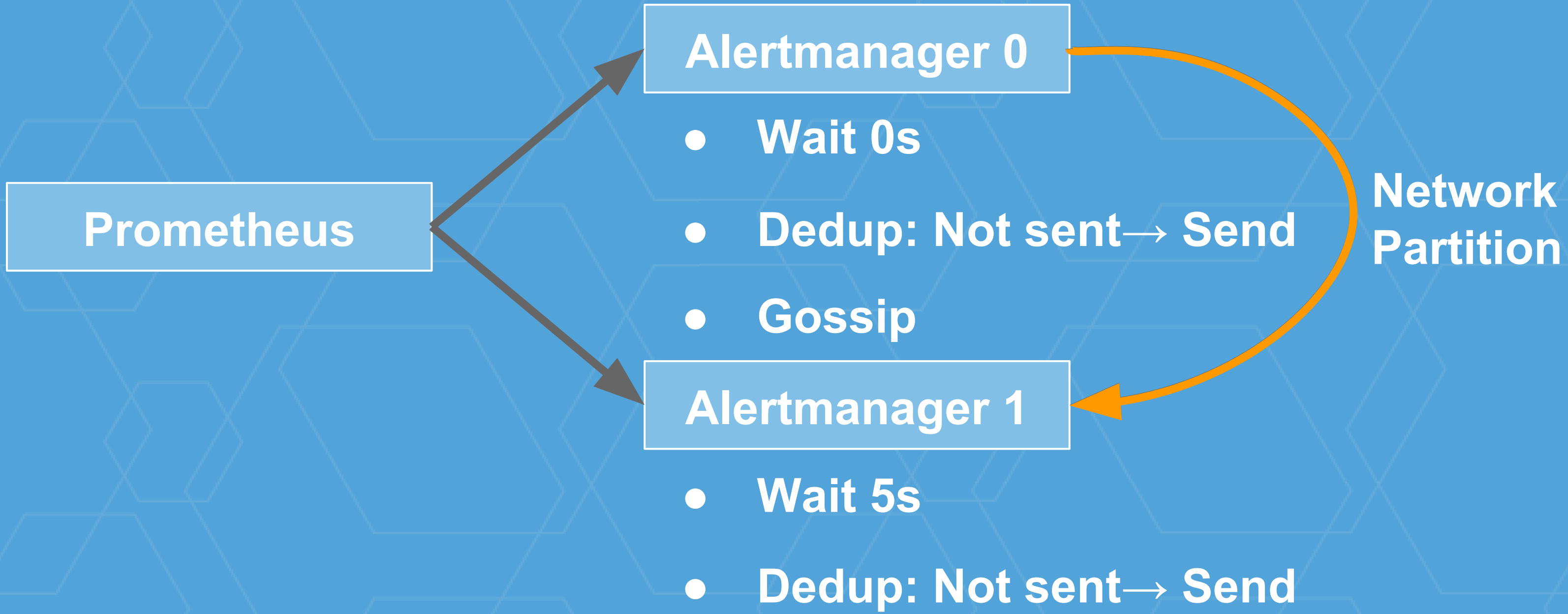
Alertmanager 0

- Wait 0s
- Dedup: Not sent→ Send
- Gossip

Alertmanager 1

- Wait 5s
- Receive Gossip Data
- Deduplicate → Do not send

Gossip Partition



Prometheus

Alertmanager 0

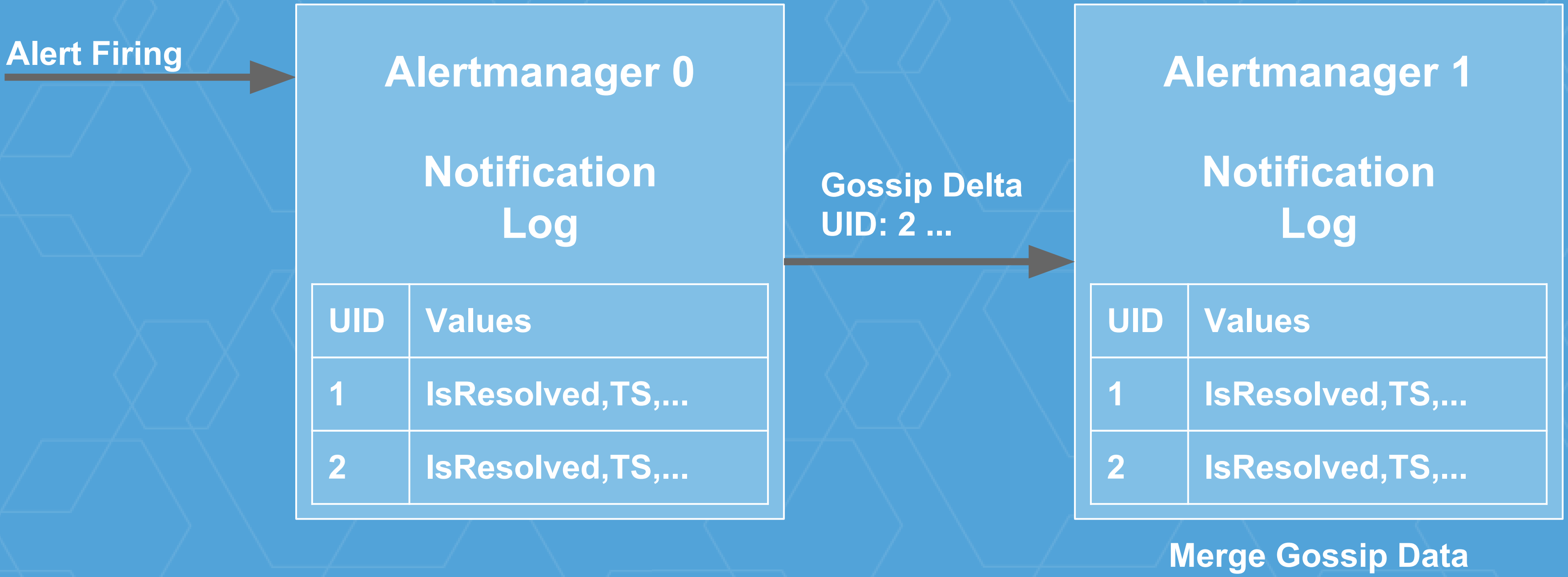
- Wait 0s
- Dedup: Not sent→ Send
- Gossip

Alertmanager 1

- Wait 5s
- Dedup: Not sent→ Send

Network Partition

Notification Log



Alert Firing

Alertmanager 0

Notification Log

UID	Values
1	IsResolved,TS,...
2	IsResolved,TS,...

Gossip Delta
UID: 2 ...

Alertmanager 1

Notification Log

UID	Values
1	IsResolved,TS,...
2	IsResolved,TS,...

Merge Gossip Data

Group Key

```
global:
  resolve_timeout: 5m

route:
  group_by: ['job']
  group_wait: 10s
  group_interval: 10s
  repeat_interval: 1h
  receiver: 'webhook'
receivers:
- name: 'webhook'
  webhook_configs:
  - url: 'http://127.0.0.1:5001/'
```

- Group at runtime
    - By Group By labels
- XOR with Route
- Concat with Receiver

DEMO!

Thanks!

QUESTIONS?

frederic.branczyk@coreos.com

GitHub: @brancz

Twitter: @fredbrancz

LONGER CHAT?

Let's talk!

#prometheus on Freenode

More events: coreos.com/community

We're hiring: coreos.com/careers

ALSO IN BERLIN!