KubeCon | CloudNativeCon

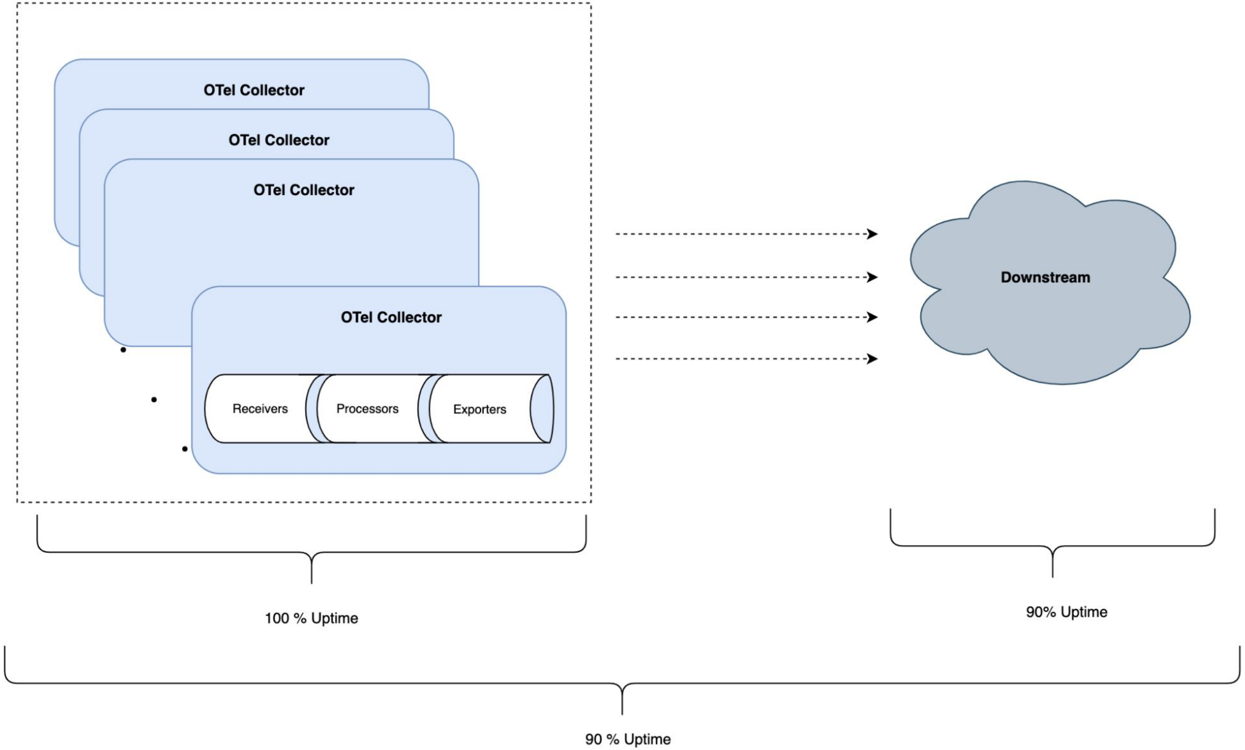
North America 2024

Minimizing Data loss within the OpenTelemetry Collector

Alex Kats, Capital One

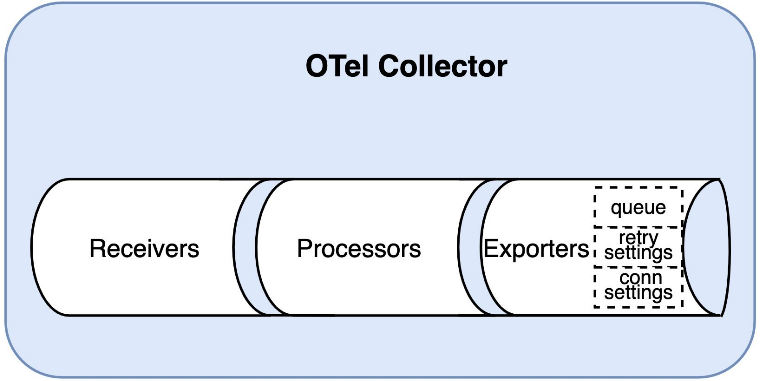
Only as Strong as the Weakest Link

Collector is only as resilient as the next target in export chain



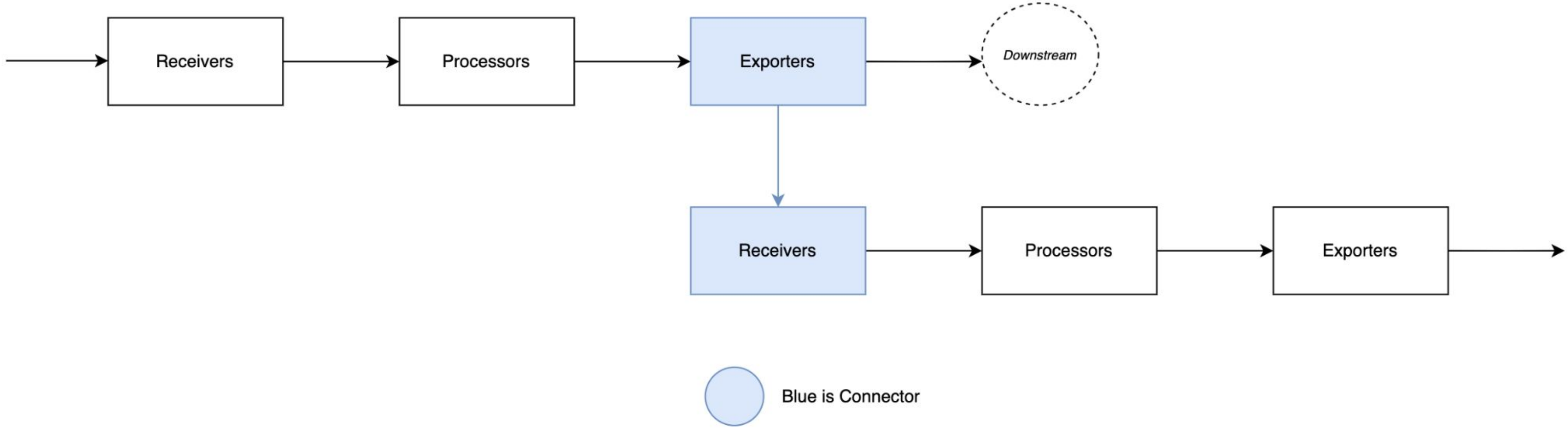
Current State of Collector Resiliency

- Data resiliency is handled by exporters
- Exporters handle async exports through queuing and retry settings
- Persistent queue option exists to persist data on file system



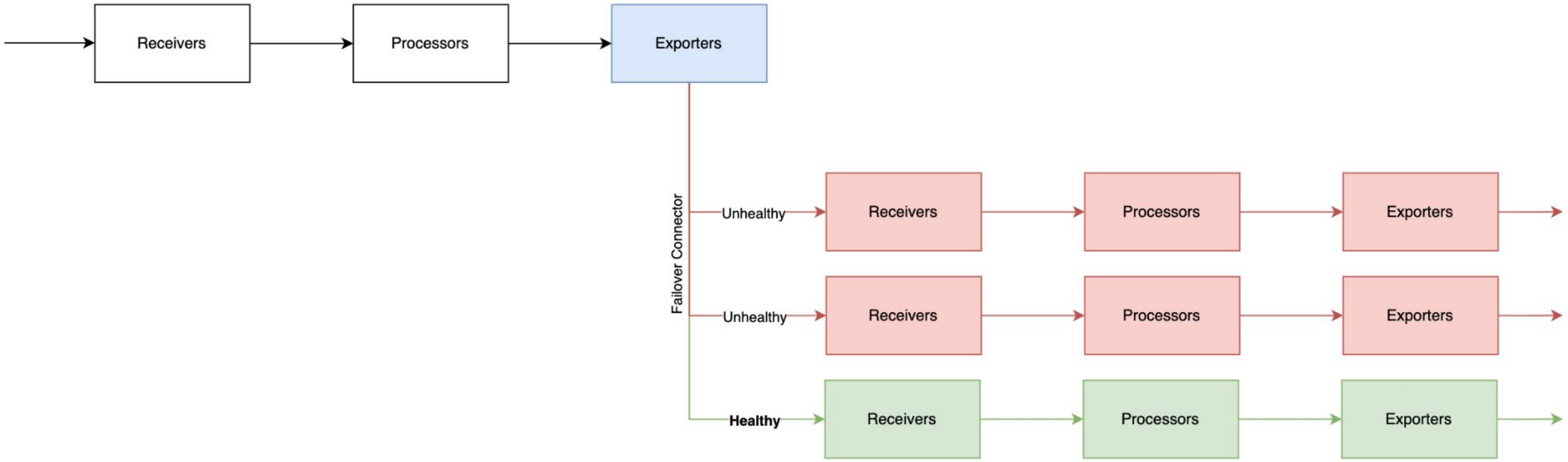
Enter Connectors

A way to export telemetry data between pipelines within the collector

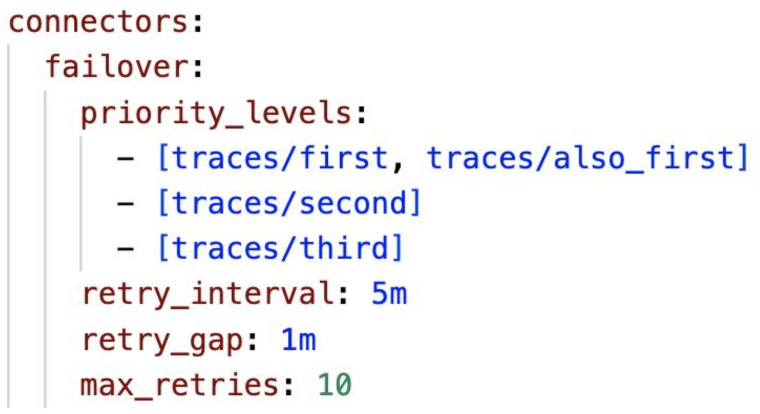


The Failover Connector

Health based routing between trace, metric, and log pipelines depending on the health of target downstream exporters



Failover Connector Configuration



```
connectors:
  failover:
    priority_levels:
      - [traces/first, traces/also_first]
      - [traces/second]
      - [traces/third]
    retry_interval: 5m
    retry_gap: 1m
    max_retries: 10
```

priority_levels: Ordered list of pipeline arrays, if one pipeline on level goes unhealthy, entire level is unhealthy

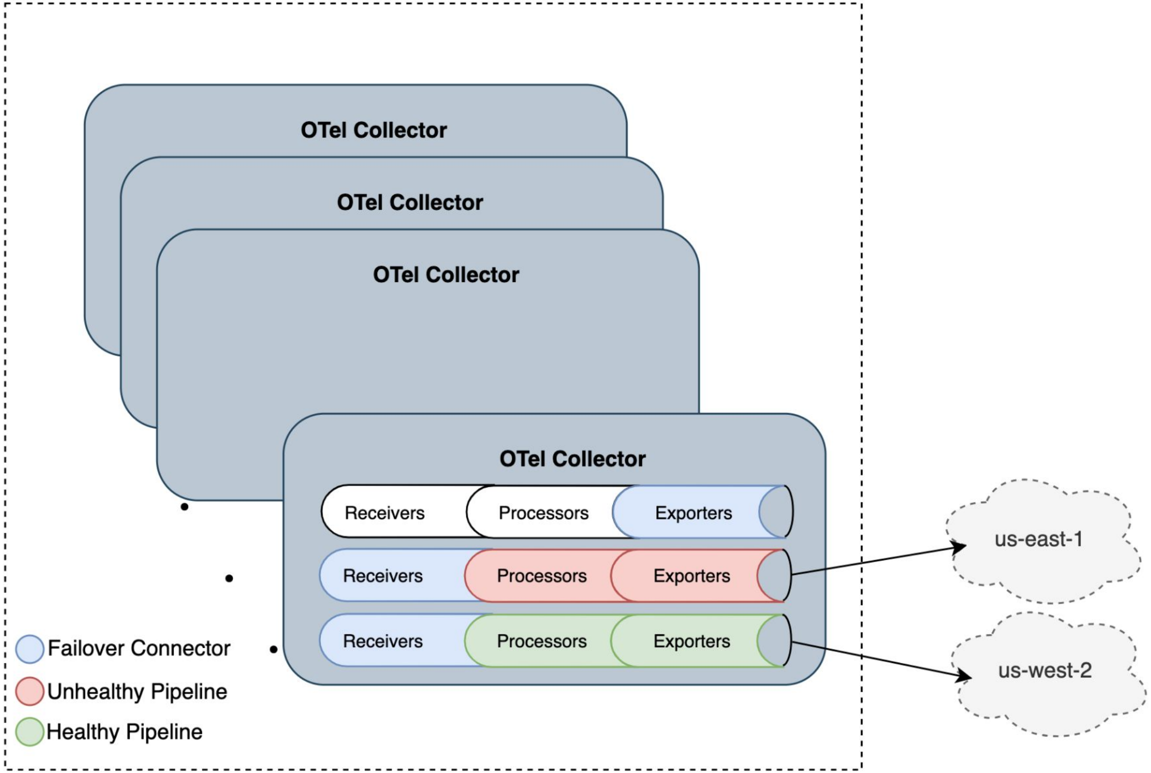
retry_interval: How often connector will go through the full priority list and try to re-establish higher priority pipelines

retry_gap: How long connector will wait between retrying two levels, if current level is three, assuming one and two still unhealthy, will wait 1m between one and two

max_retries: How many retries before a level is deemed permanently unhealthy, *can be disabled*

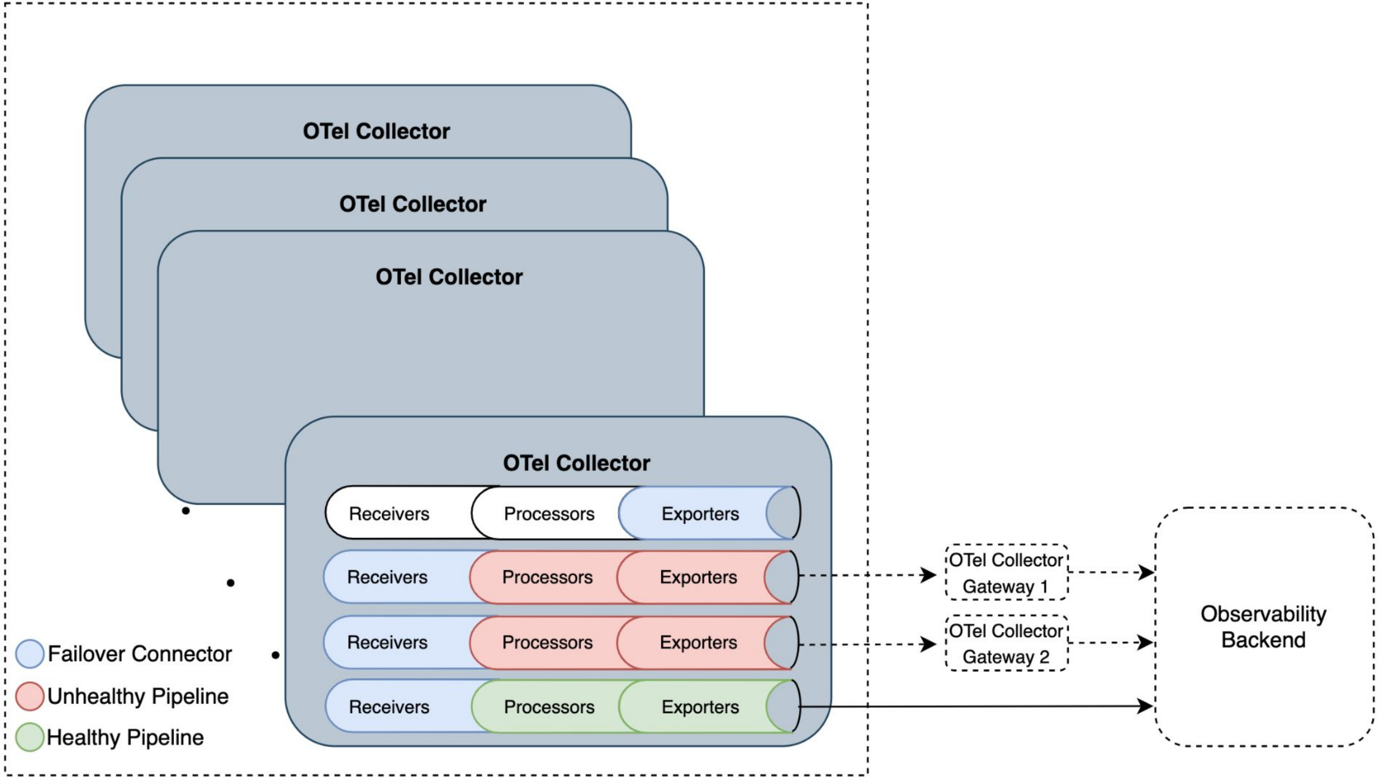
What can I do with this?

Cross Region Failover

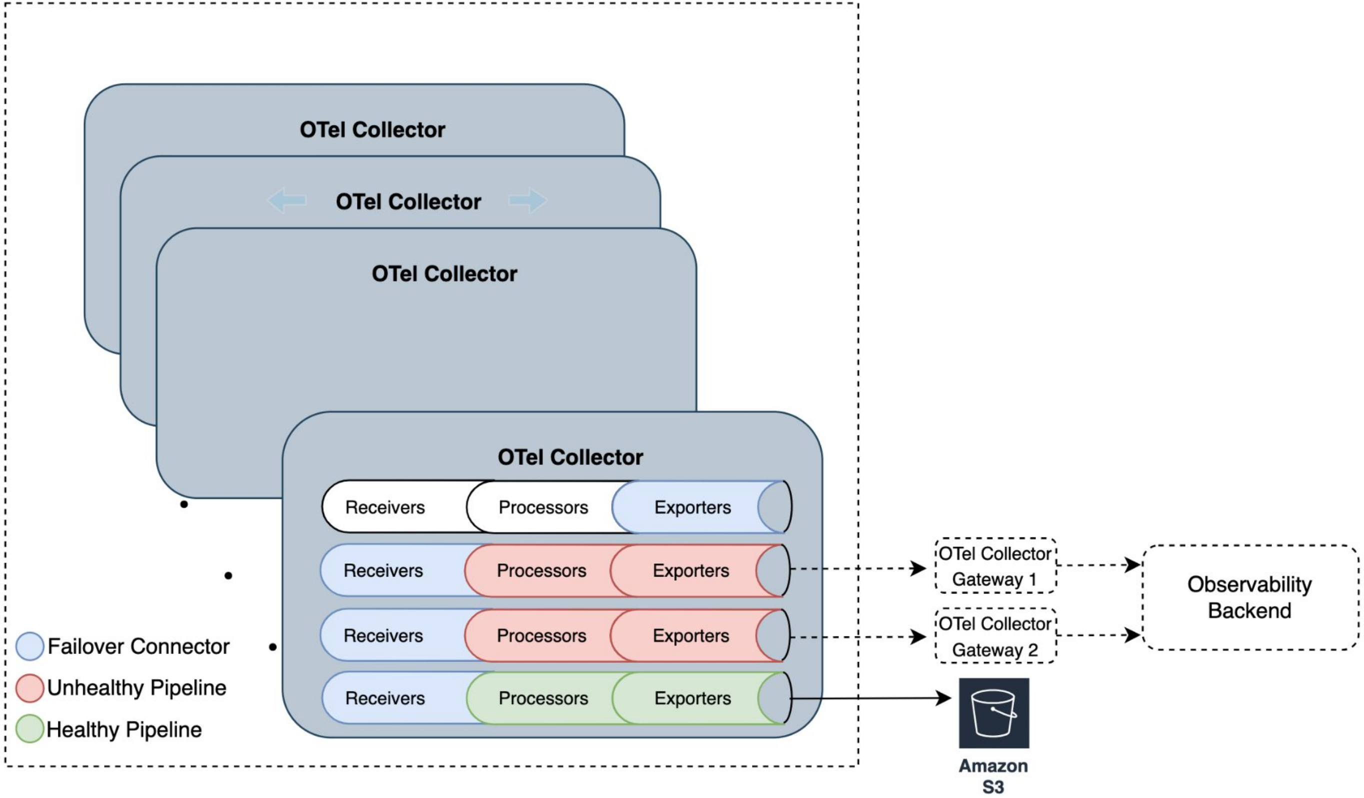


Bypass Next Target

KubeCon CloudNativeCon North America 2024



High Uptime Target



Thank you