



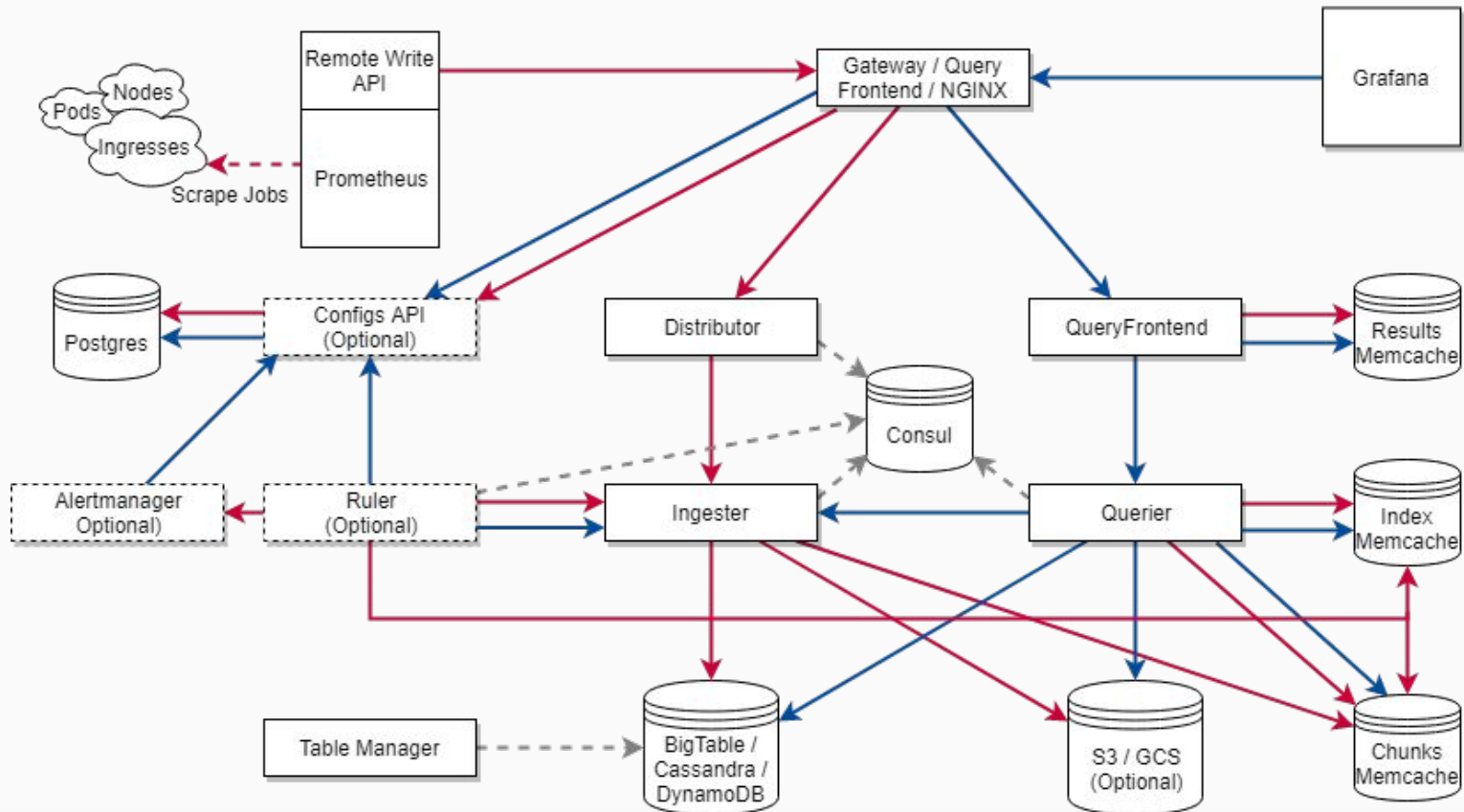
Configuring Cortex for max performance

Goutham Veeramachaneni

 @putadent

```
Limit how long back data can be queried
-store.max-query-length duration
Limit to length of chunk store queries, 0 to disable.
-store.min-chunk-age duration
Minimum time between chunk update and being saved to the store.
-store.query-chunk-limit int
Maximum number of chunks that can be fetched in a single query. (default 2000000)
-table-manager.retention-deletes-enabled
If true, enables retention deletes of DB tables
-table-manager.retention-period duration
Tables older than this retention period are deleted. Note: This setting is destructive to data!(default: 0, v
-table-manager.throughput-updates-disabled
If true, disable all changes to DB capacity
-target value
target module (default All) (default all)
-validation.create-grace-period duration
Duration which table will be created/deleted before/after it's needed; we won't accept sample from before th
-validation.enforce-metric-name
Enforce every sample has a metric name. (default true)
-validation.max-label-names-per-series int
Maximum number of label names per series. (default 30)
-validation.max-length-label-name int
Maximum length accepted for label names (default 1024)
-validation.max-length-label-value int
Maximum length accepted for label value. This setting also applies to the metric name (default 2048)
-validation.reject-old-samples
Reject old samples.
-validation.reject-old-samples.max-age duration
Maximum accepted sample age before rejecting. (default 336h0m0s)
→ cortex git:(master) x ./cortex --help 2>&1 | wc -l
1005
```

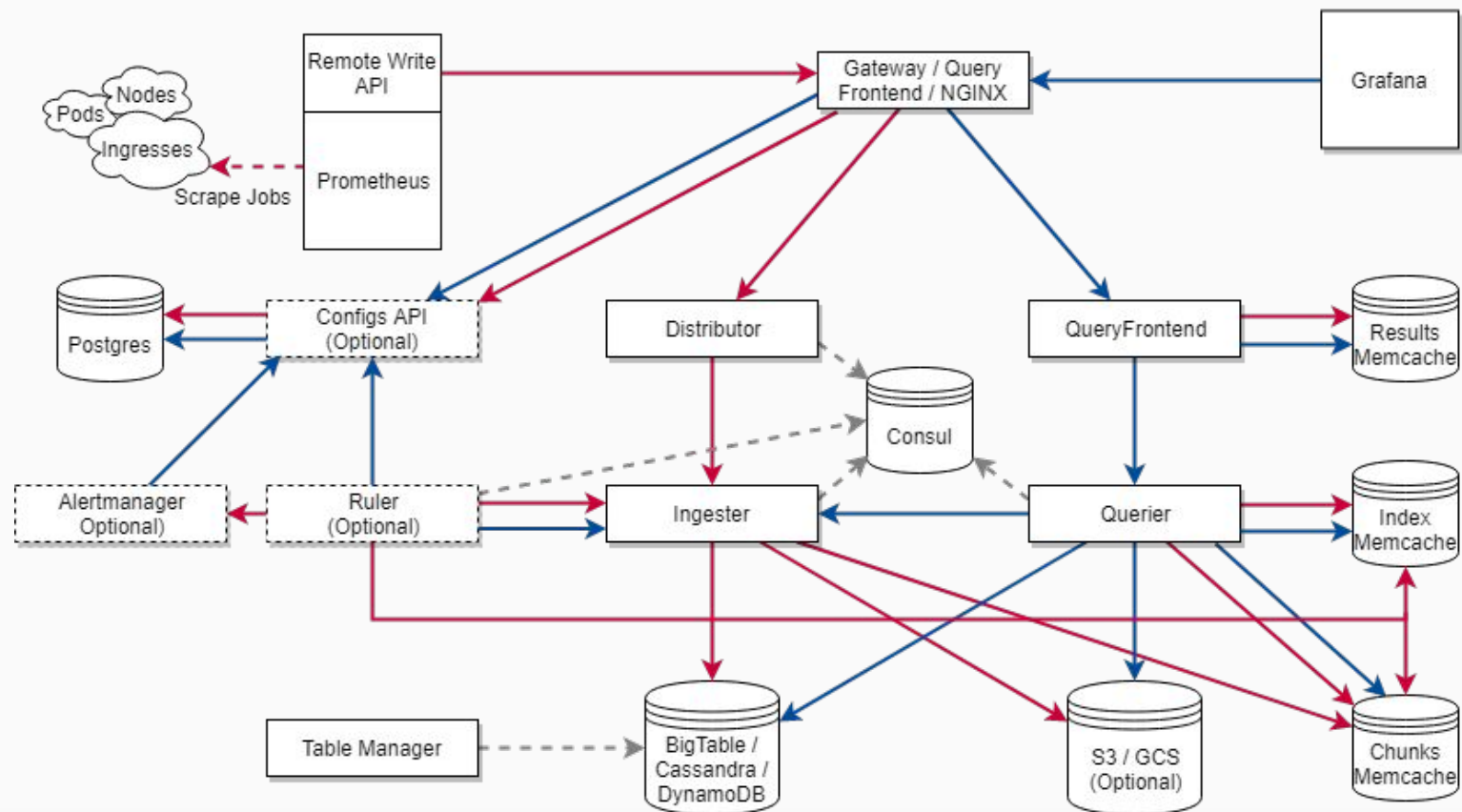
- Write Path
- Query path
- - - → Control requests



The Write Path

The easy path ;)

- Write Path
- Query path
- - - → Control requests



Headlines

Samples / s

142K reqps

Active Series

2.178 Mil

QPS

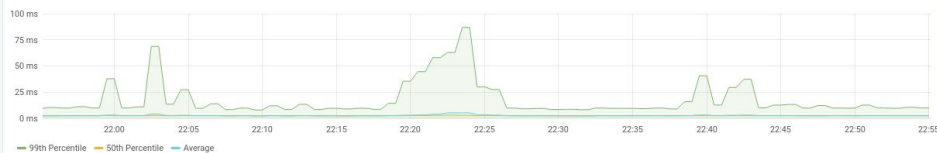
2.492K reqps

Gateway

QPS



Latency



Distributor

QPS



Latency



Etcd (HA Dedupe)

QPS



Latency



▼ Workload-based scaling

Workload-based scaling ▼					
Cluster ▲	Deployment	Namespace	Current Replicas	Required Replicas, by ingestion rate	Required Replicas, by active series
us-central1	ingester	dev	12	7	7
us-central1	memcached	dev	-	-	4

▼ Resource-based scaling

Resource-based scaling ▼					
Cluster	Deployment	Namespace	Current Replicas ▼	Required Replicas, by CPU usage	Required Replicas, by Memory usage
us-central1	ingester	dev	12	30	21
us-central1	distributor	dev	8	22	8 Bil
us-central1	querier	dev	6	0	1
us-central1	cortex-gw	dev	6	3	3
us-central1	memcached	dev	4	1	4
us-central1	memcached-index-writes	dev	3	0	2
us-central1	memcached-index-queries	dev	3	0	2
us-central1	memcached-frontend	dev	3	0	0



```
/etc/prometheus/alerts.rules > cortex-provisioning
```

CortexProvisioningMemcachedTooSmall (0 active)

CortexProvisioningTooManyActiveSeries (0 active)

CortexProvisioningTooManyWrites (0 active)

CortexProvisioningTooMuchMemory (0 active)



CortexReadErrorBudgetBurn (0 active)

```
alert: CortexReadErrorBudgetBurn
expr: ((100
  * namespace_job:cortex_gateway_read_slo_errors_per_request:ratio_rate1h > 0.5
  * 14.4) and (100 * namespace_job:cortex_gateway_read_slo_errors_per_request:ratio_rate5m
  > 0.5 * 14.4))
for: 2m
labels:
  period: 1h
  severity: critical
annotations:
  description: '{{ $value | printf `%.2f` }}% of {{ $labels.job }}'s read requests
    in the last 1h are failing or too slow to meet the SLO.'
  runbook_url: https://github.com/kubernetes-monitoring/kubernetes-mixin/tree/master/runbook.md#alert-name-cortexreaderrorbudgetburn
  summary: Cortex burns its read error budget too fast.
```

CortexWriteErrorBudgetBurn (0 active)

CortexWriteErrorBudgetBurn (0 active)

CortexWriteErrorBudgetBurn (0 active)

CortexWriteErrorBudgetBurn (0 active)

I need some 

<https://github.com/grafana/cortex-jsonnet>

Our write outage!

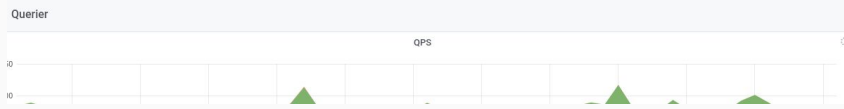
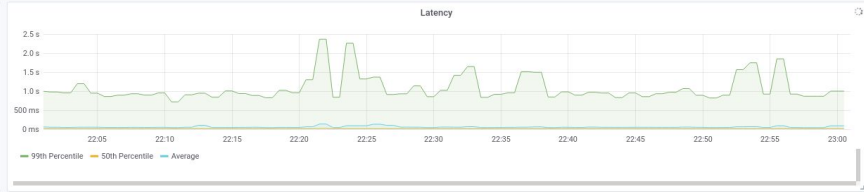
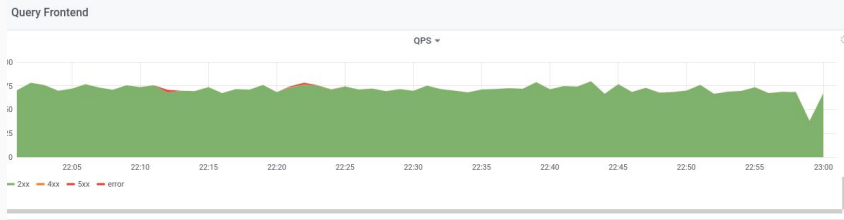
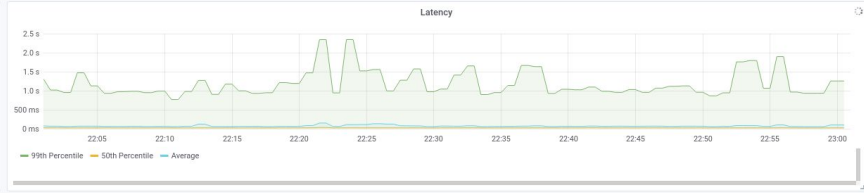
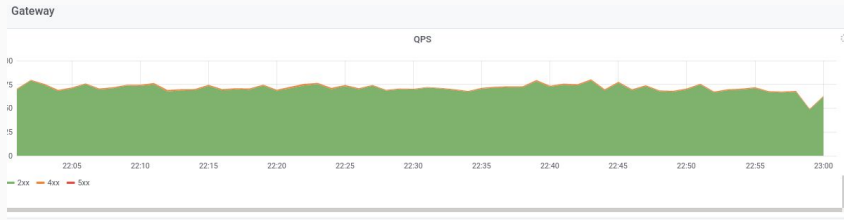
- etcd borked!



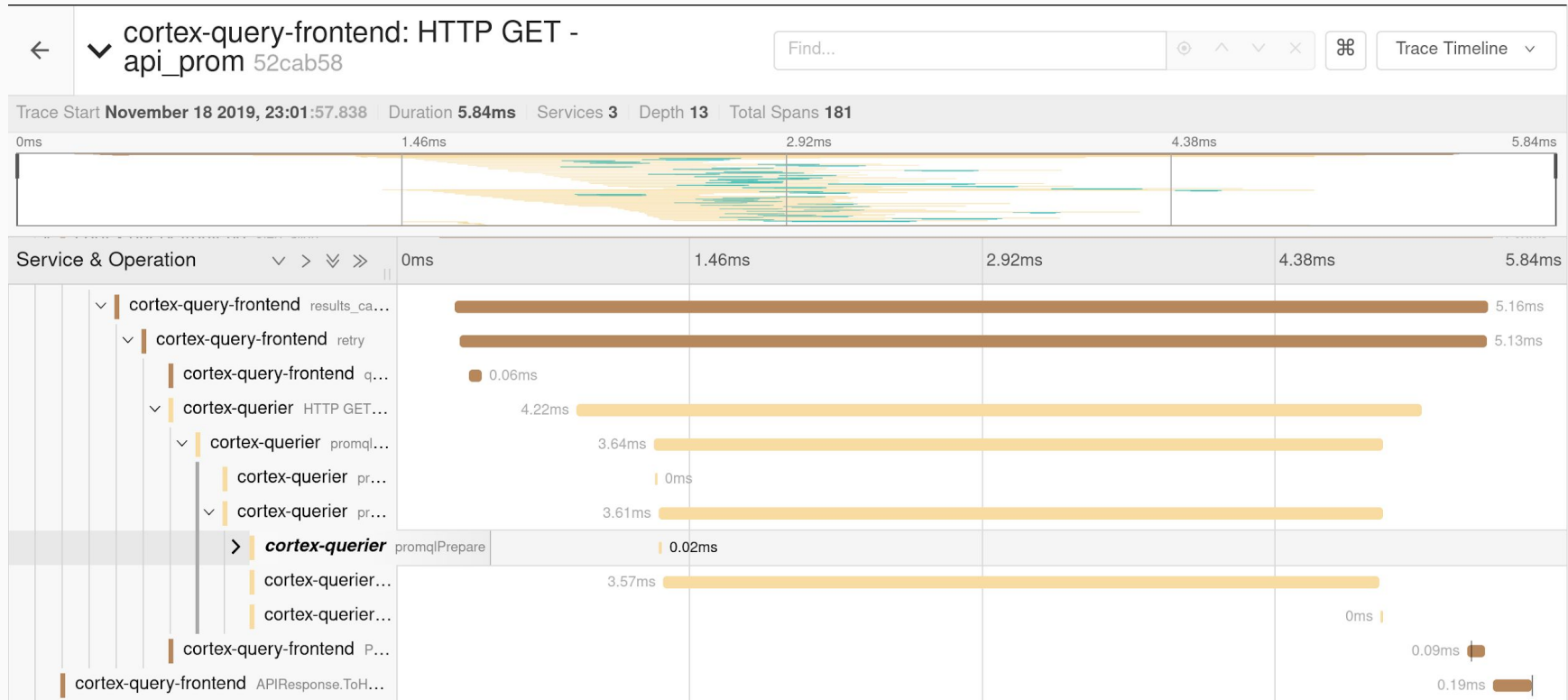
The Read Path

Now we're talking!

Step -1: Install the mixin



Step 0: Install Jaeger



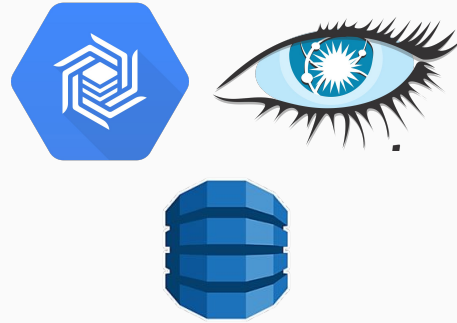
```
jaeger_mixin::
  if $_config.jaeger_agent_host == null
  then {}
  else
    container.withEnvMixin([
      container.envType.new('JAEGER_AGENT_HOST', $_config.jaeger_agent_host),
      container.envType.new('JAEGER_TAGS', 'namespace=%s,cluster=%s' % [$_config.namespace, $_config.cluster]),
      container.envType.new('JAEGER_SAMPLER_MANAGER_HOST_PORT', 'http://%s:5778/sampling' % $_config.jaeger_agent_host),
    ]),
```

```
querier_container::
  container.new('querier', $_images.querier) +
  container.withPorts($_util.defaultPorts) +
  container.withArgsMixin($_util.mapToFlags($_querier_args)) +
  $_util.resourcesRequests('1', '12Gi') +
  $_util.resourcesLimits(null, '24Gi') +
  $_jaeger_mixin +
  container.withEnvMap({
    JAEGER_REPORTER_MAX_QUEUE_SIZE: '1024', // Default is 100.
  }),
```

Outage Walkthrough

Demo time!

Things to lookout for



- Queueing and Queries piling up

LIMIT EVERYTHING


3. Prove from first principles that

$$\left. \begin{array}{l} \lim_{x \rightarrow 0^+} \frac{5x^2 + 1}{x} = +\infty \\ \lim_{x \rightarrow 0^-} \frac{5x^2 + 1}{x} = -\infty \end{array} \right\} \text{Hence } \frac{5x^2 + 1}{x} \text{ has no limit at } 0.$$

Per user overrides



```
overrides: {
  '362': super.medium_user,
  '540': super.medium_user,
  '600': super.medium_user,
  '842': super.medium_user,
  '5313': super.medium_user + {
    ha_cluster_label: 'cluster',
    ha_replica_label: 'prometheus_replica',
  },
  '5879': super.medium_user,
  '6187': super.medium_user,
  '7625': super.medium_user,
  '5978': super.medium_user,
  '7776': super.big_user,
  '320': super.big_user,
  '327': super.big_user,
  '512': super.big_user,
  '5465': super.big_user,
  '7024': super.big_user,
  '7472': super.big_user + {
    max_label_names_per_series: 40,
  },
  '461': super.super_user,
  '7319': super.super_user + {
    accept_ha_samples: true,
    ha_cluster_label: 'prom_ha_cluster',
    ha_replica_label: 'prom_ha_instance',
  },
},
```

- `store.max-query-length=744h`
- **`store.max-query-length=12000h`** 
- `store.cardinality-limit=2000000`
- `querier.max-samples=100000000`

- `store.cache-lookups-older-than=36h`
- `querier.query-ingesters-within=12h`

Summary

- Use the mixin (build alerts and dashboards in general)
- Scale up with usage
- Jaeger is love
- Limit everything

Questions!

 @putadent