



KubeCon



CloudNativeCon

North America 2021

RESILIENCE

REALIZED

SIG Instrumentation Intro & Deep Dive

David Ashpole, Google, Tech Lead (@dashpole)

Frederic Branczyk, Polar Signals, Tech Lead (@brancz)

Elana Hashman, Red Hat, Chair (@ehashman)

Han Kang, Google, Chair (@logicalhan)

Introduction



KubeCon



CloudNativeCon

North America 2021

- What does SIG Instrumentation do?
- Current activities
 - SIG Subprojects
 - Logs
 - Traces
 - Metrics
- How to contribute
- Where to find us
- Related talks!

What do we do?



KubeCon



CloudNativeCon

North America 2021

- **Charter**: To cover best practices for cluster observability across all Kubernetes components and develop relevant components.
- **Subprojects**:
 - kube-state-metrics
 - klog
 - metrics-server
 - and more!
- Metrics
- Logs and Events
- Traces

How do we do it?



KubeCon



CloudNativeCon

North America 2021

- Triage and fix relevant instrumentation issues
 - [All open SIG Instrumentation-labelled issues and pull requests](#)
- Review all code changes for metrics
- Develop new features and enhancements
 - [Kubernetes Enhancement Proposals \(KEPs\) for SIG Instrumentation](#)
- Maintain subprojects



KubeCon



CloudNativeCon

North America 2021

RESILIENCE

REALIZED

Subprojects

Subprojects



KubeCon



CloudNativeCon

North America 2021

- kube-state-metrics
- Metrics Server
- k8s-prometheus-adapter

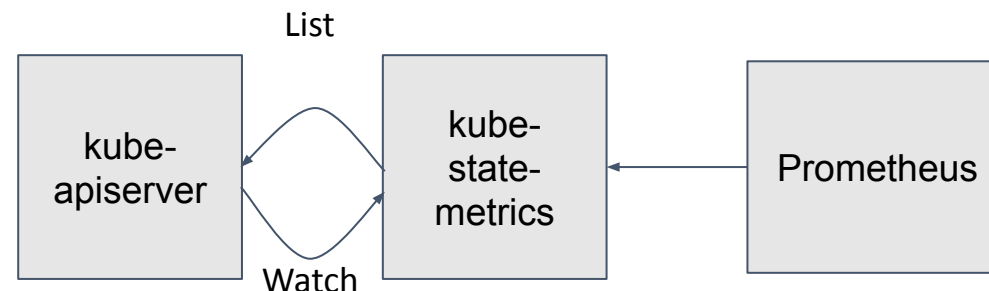
kube-state-metrics

- Generate Prometheus style metrics from Kubernetes API objects
- Pods, Deployments, StatefulSets, etc.
- v2 Release recently

Example:

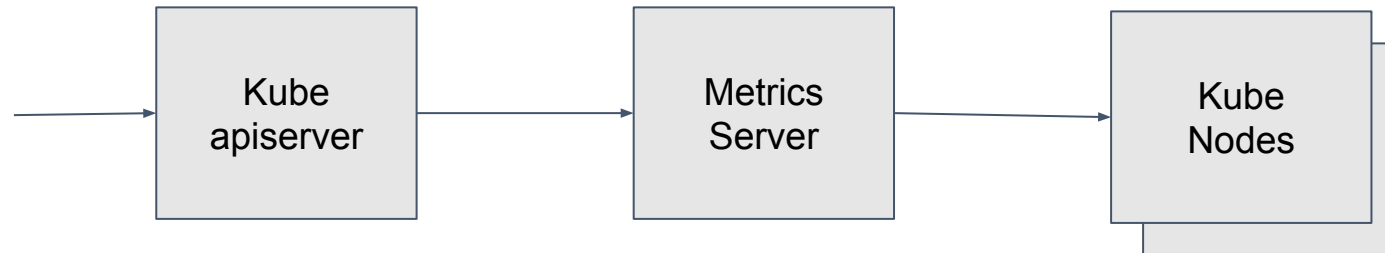
`kube_deployment_spec_replicas`

`kube_deployment_status_replicas_updated`



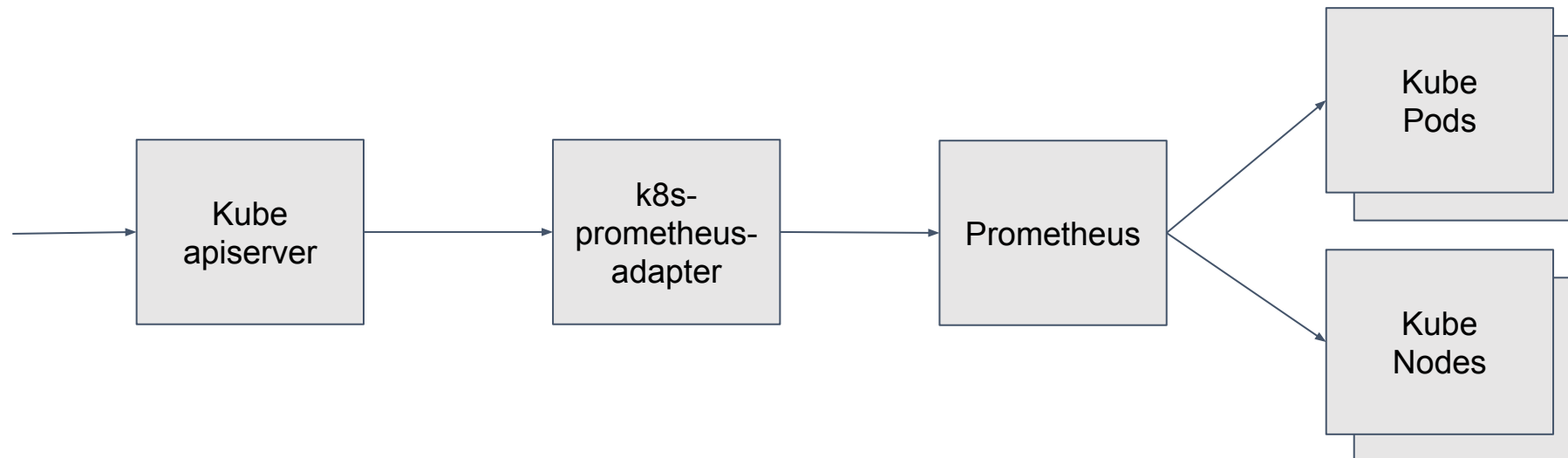
Metrics Server

- Implementation of the resource metrics API
 - Source of `$ kubectl top`
 - Source of metrics for resource based autoscaling
- Repo: <https://github.com/kubernetes-sigs/metrics-server>



k8s-prometheus-adapter

- Implementation of the resource/custom/external metrics API
 - Use custom metrics for autoscaling
- Repo: <https://github.com/kubernetes-sigs/prometheus-adapter>
 - Migration to official repos in progress





KubeCon



CloudNativeCon

North America 2021

RESILIENCE

REALIZED

Logs

Structured Logs: What?

Before:

```
I0902 03:19:16.663200      1862 kubelet.go:1856] SyncLoop (ADD, "api"):  
"busybox-user-0-fd7df2b0-44be-44d3-8263-c63607950d99_security-context-test-4141  
(10a830b4-bdff-4671-a947-451346fe13fe)"
```

After (text):

```
I0902 21:38:49.937907      1821 kubelet.go:2053] "SyncLoop ADD" source="api"  
pods=[security-context-test-832/busybox-privileged-false-77fb495d-3037-4597-868  
d-d4b0e7a3eafd]
```

After (JSON):

```
{"ts":1630623419364.0852,"caller":"kubelet/kubelet.go:2053","msg":"SyncLoop  
ADD","v":2,"source":"api","pods":[{"name":"security-context-test-832","namespac  
e":"busybox-privileged-false-77fb495d-3037-4597-868d-d4b0e7a3eafd"}]}
```

Structured Logs: When?

- Fully migrated Kubelet in 1.21 release ([#98976](#))
 - Includes static analysis to prevent regressions
- Selecting components for migration in 1.23 ([#104197](#))
- Deprecating klog-specific flags in Kubernetes components
 - Reduce maintenance burden and complexity
 - Reduce number of flags needed to be supported by JSON and other formats
 - [kubernetes/enhancements#2845](#)

Structured Logs: Who?



KubeCon



CloudNativeCon

North America 2021

- Spun off new **WG Structured Logging** to manage the structured log migration
 - **Organizers:**
 - Marek Siarkowicz (@serathius), Google
 - Shubheksha Jalan (@shubheksha), Apple
 - **Slack channel:** #wg-structured-logging
 - **Charter:** kubernetes/community/wg-structured-logging
 - **Biweekly meetings:** Thursdays at 15:00 UTC
- They need your help!

- Hide credentials and secrets from Kubernetes' logs
 - Debuted as alpha in 1.20
- Moved KEP ownership to the new SIG Security
- **Static checks:** graduating to stable in 1.23
 - [kubernetes/enhancements#1933](https://kubernetes.io/enhancements/#1933)
- **Dynamic sanitization:** targeting beta in 1.23
 - Enable with `--logging-sanitization`
 - [kubernetes/enhancements#1753](https://kubernetes.io/enhancements/#1753)

Events API redesign



KubeCon



CloudNativeCon

North America 2021

- Introduced >4 years ago
- Need to complete migration before we can declare it as graduated
 - [kubernetes/enhancements#383](https://kubernetes.io/enhancements/#383)



KubeCon



CloudNativeCon

North America 2021

RESILIENCE

REALIZED

Traces

Control-plane Tracing:

- API Server Tracing reached Alpha in 1.22
- Etcd has experimental tracing support in v3.5.0

Pod Lifecycle Tracing (coming soon):

- Kubelet Tracing: [KEP](#)
- CRI-O Tracing: [Issue](#)
- Containerd Tracing: [Issue](#)

Controller Context Propagation (coming soon): [KEP](#)

Traces: Control-Plane

Enable the `APIServerTracing` feature-gate

On the Kube API Server, set `--tracing-config-file` to a file with:

```
apiVersion: apiserver.config.k8s.io/v1alpha1
kind: TracingConfiguration
# 1% sampling rate
samplingRatePerMillion: 10000
```

On Etcd, set `--experimental-enable-distributed-tracing`

Run the OpenTelemetry collector as sidecar containers to the API Server and Etcd, and send traces to your backend of choice!

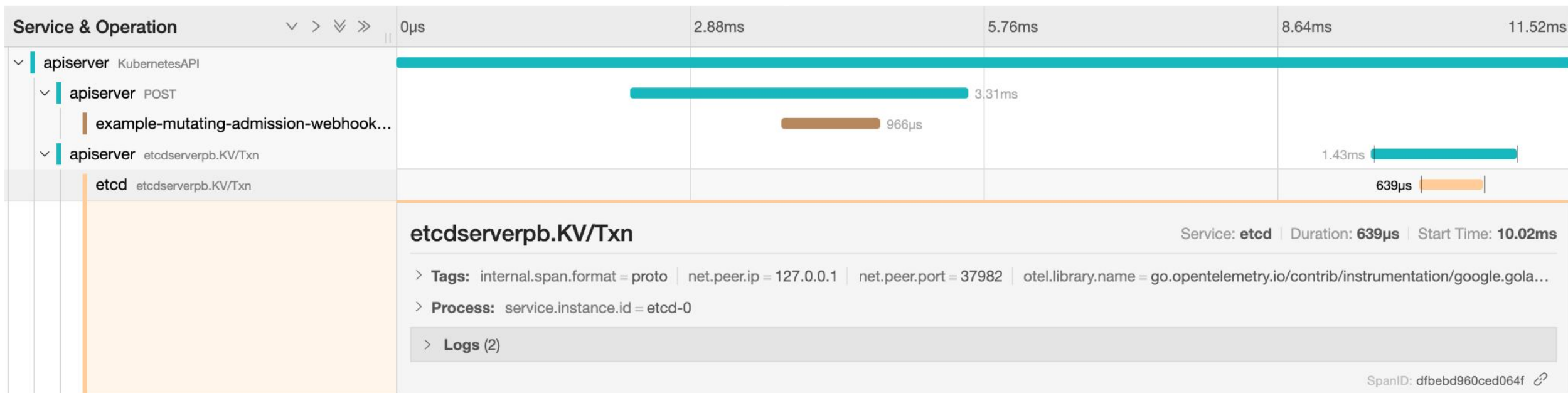
Traces: Control-Plane

Spans from:

- API Server Tracing (teal)
- Etcd Tracing (orange)
- Custom Mutating admission controller (brown)

Collected with: OpenTelemetry Collector(s)

Displayed with: Jaeger





KubeCon



CloudNativeCon

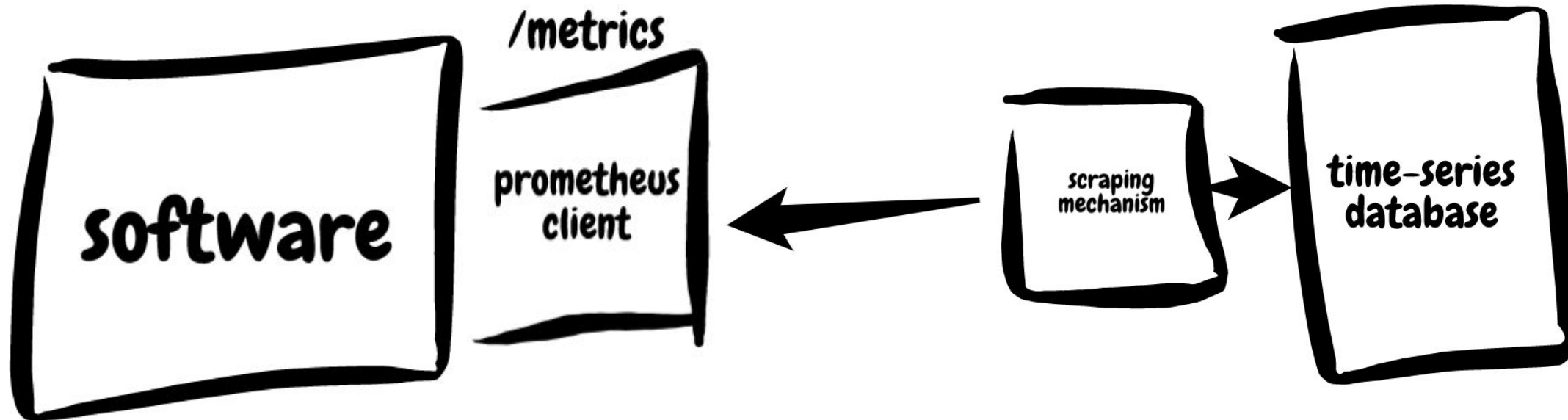
North America 2021

RESILIENCE

REALIZED

Metrics

Kubernetes components integrate with Prometheus, a time-series based monitoring and alerting toolkit.



Metrics Woes



KubeCon



CloudNativeCon

North America 2021

Leaking reflector metrics in 1.8 #53485



Closed

brancz opened this issue on Oct 5, 2017 · 12 comments · Fixed by #53586



brancz commented on Oct 5, 2017

Member



Is this a **BUG REPORT** or **FEATURE REQUEST**?:

/kind bug

What happened:

Ran Kubernetes 1.8 apiserver for 3 days and the memory usage kept going up.

`go_memstats_heap_inuse_bytes(service="kubernetes")`

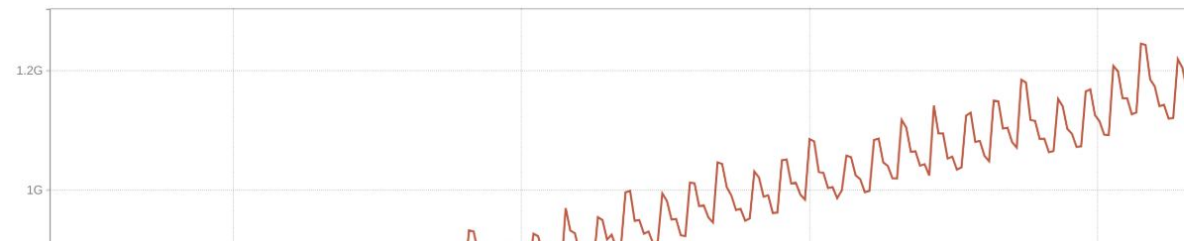
Execute

- insert metric at cursor -

Load time: 96ms
Resolution: 345s
Total time series: 1

Graph Console

1d + << Until >> Res. (s) stacked



Assignees

No one assigned

Labels

kind/bug

priority/important

sig/api-machinery

sig/scalability

Projects

None yet

Milestone

No milestone

Notifications

Metrics Woes



KubeCon



CloudNativeCon

North America 2021

Closed

High cardinality metrics coming from reflectors #52121

smarterclayton opened this issue on Sep 7, 2017 · 4 comments · Fixed by #54921



smarterclayton added **sig/api-machinery** **sig/instrumentation** labels on Sep 7, 2017



k8s-github-robot removed the **needs-sig** label on Sep 7, 2017



lavalamp commented on Sep 11, 2017

...are we storing the resource version in the metric *name*?

Member

1G

Notifications

Metrics Woes



KubeCon



CloudNativeCon

North America 2021

rename metric reflector_xx_last_resource_version #5492

Merged

k8s-github-robot merged 1 commit into `kubernetes:master` from `weiwei04:fix_reflector_last_resource_ver`

Conversation 9

Commits 1

Checks 0

Files changed 2



weiwei04 commented on Nov 1, 2017

Contributor



What this PR does / why we need it:

mv reflector name from metric name to metric label

before:

```
reflector_k8s_io_kubernetes_pkg_client_informers_informers_generated_internalversion_fac
```

after

Nov 7, 2017

Member

Notifications

Metrics Woes



KubeCon



CloudNativeCon

North America 2021

memory leak in kubelet 1.12.5 #73587

Closed

Sandor Szücs opened this issue on Jan 31 · 35 comments · Fixed by [#74636](#)



Sandor Szücs commented on Jan 31 • edited ▾

Member

+ 😊 ...

What happened:

After upgrading to kubernetes 1.12.5 we observe failing nodes, that are caused by kubelet eating all over the memory after some time.



Metrics Cardinality



KubeCon



CloudNativeCon

North America 2021

some_request_metric

labels:

VERB

CODE

PATH

PUT

200

/pods

POST

201

DELETE

404

GET

500

Metrics Stability



KubeCon



CloudNativeCon

North America 2021

- Provide a framework to expressing metric stability guarantees
- Provide automation around stability levels
- (stretch goal) Provide a mechanism to centralize instrumentation related code and instrumentation processes



Metrics Cardinality Enforcement

- **Problem:** cardinality problems keep surfacing and metrics stability isn't enough.
- **Problem:** fixes required code changes, backports... these are inherently expensive
- **Solution:** Metrics Cardinality Enforcement

Future Efforts

- Extending metrics stability
 - parity with feature stages
 - broader expressiveness for stability
 - automated documentation



KubeCon



CloudNativeCon

North America 2021

RESILIENCE

REALIZED

Get involved!

How to contribute

- Attend our SIG meetings!
- Participate in reviews, issues, and docs!
- `kube-state-metrics` is seeking new contributors
 - Contact **Lili Cosic** (@lilic)
- `metrics-server` and structured logging implementation are seeking contributors
 - Contact **Marek Siarkowicz** (@serathius)
- `promq` is seeking new contributors
 - Contacts: **Han Kang** (@logicalhan), **Yuchen Zhou** (@YoyinZyc)

Where to find us

- **SIG Meetings:**
 - [Regular meeting](#), alternating **biweekly** on **Thursdays at 9:30am Pacific Time**
 - [Triage meeting](#), alternating **biweekly** on **Thursdays at 9:30am Pacific Time**
- **Slack channel:** [#sig-instrumentation](#)
- **Mailing list:** [kubernetes-sig-instrumentation](#)
- **Chairs:** @ehashman and @logicalhan
- **Tech leads:** @brancz and @dashpole

RESILIENCE
REALIZED



KubeCon



CloudNativeCon

North America 2021