Got the Need for Speed? Hit the Gas Pedal and Accelerate Your Prometheus Dashboard Using Trickster

Shilla Saebi @shillasaebi

James Ranson @ranson

COMCAST

James Ranson, Senior Principal Architect, Comcast
Trickster Creator and Maintainer
@ranson

Shilla Saebi, Open Source Program Manager, Comcast
Chief dispenser of Trickster champagne
@shillasaebi

Comcast Open Source Program Office

Opened in 2017

Support
7k+ developers

141+ project repos on GitHub

4-person team

Our Open Source Journey

- OSS consumption

- Internal creation of RDK for X1
- Contributions to Apache HttpComponents
- Signed CCLA with ASF

- OSAC (Open Source Advisory Counsel formed)
- Signed CCLA for OpenStack

- Traffic Control open sourced
- Top contributors to OpenStack
- Won OpenStack SuperUser Award

- OSPO created
- Joined LF, Yocto, OpenChain, ONAP, CNCF
- Launched comcast.github.io
- Signed Cloud Foundry CLA
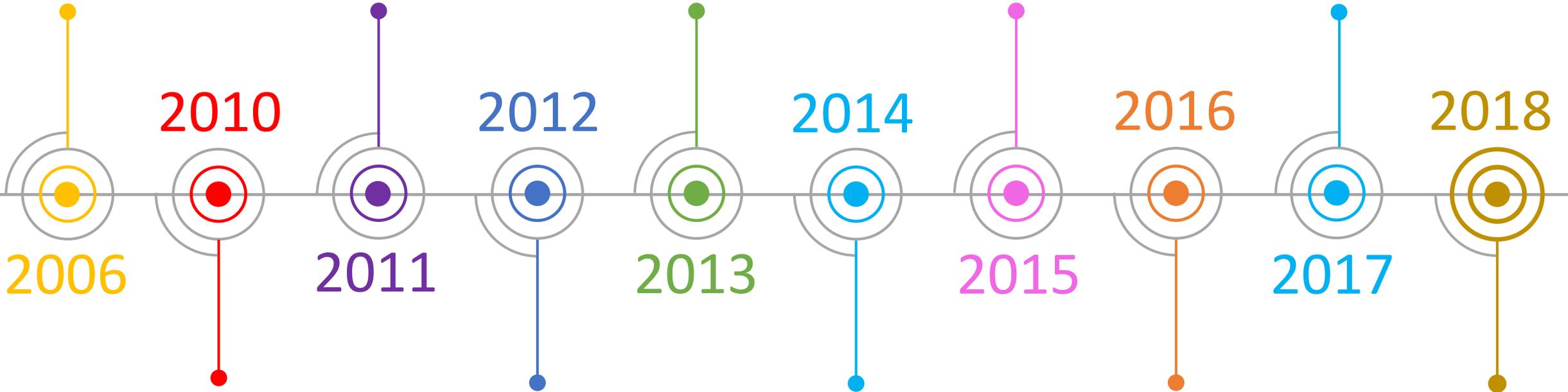
2006

2010

2011

2012

2013

2014

2015

2016

2017

2018

- Contributed caching module to Apache HttpClient

- Launched 1st 2 OSS projects on github.com/comcast
- Start of CDN project

- Consortia for RDK formed w/TWC, Charter, & Liberty Global
- Contributed IPv6 functionality to OpenStack

- Traffic Control enters Apache Incubator
- Joined ASF

- 1st Philly OS Conference
- 50M RDK DLs, 30 PB content through ATC daily
- Launched OSFP (Open Source Fellowship Pilot)
- 180 contributions this year
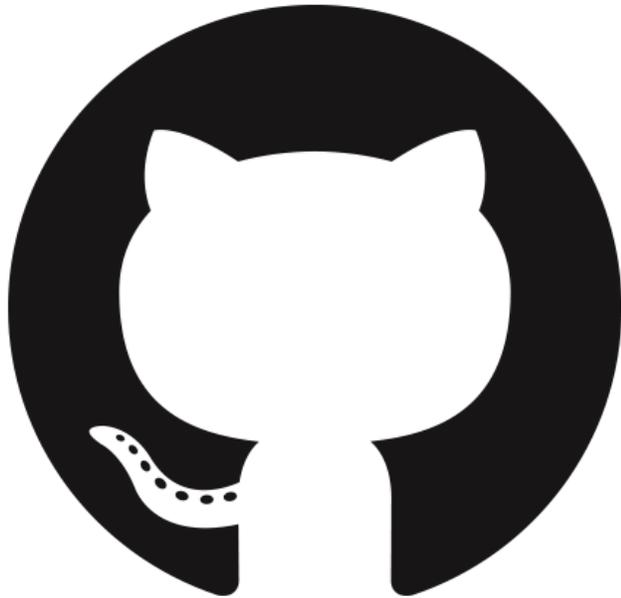- Won 1st ever Cloud Foundry Community Champion award

Trickster, developed at Comcast in 2018, is an open source project written in Go
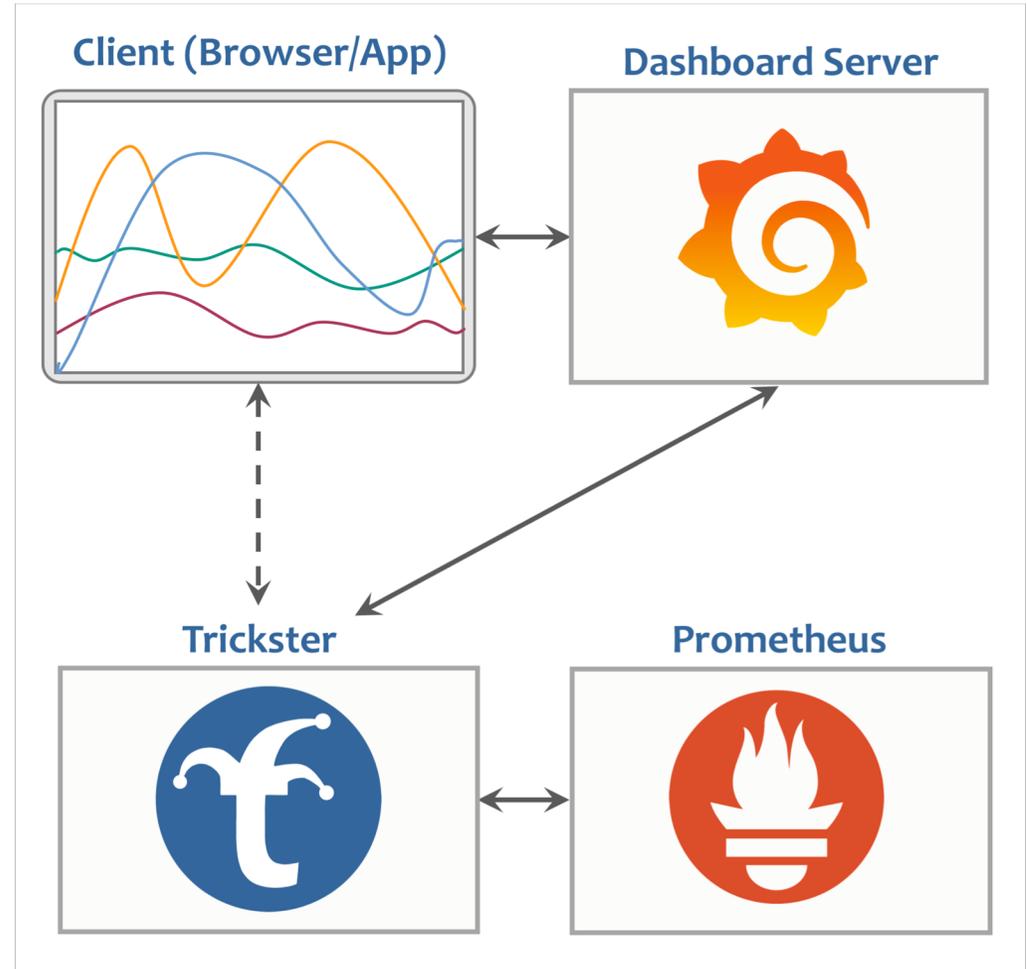
Community

Trickster Metrics

- 233 commits
- 15 releases
- 2/3 of our contributors are not from Comcast
- 519 stars
- 43 forks

Deep Dive into Trickster

Trickster dramatically accelerates dashboard rendering times for any series queried from Prometheus.

Why We Created Trickster

- Dashboard apps repeatedly process and download the same data sets, with only minor differences based on relative time range at request time.
  - Per user, per-refresh

- Most dashboard refreshes ask for ~300X more data than actually needed, taking significantly longer to serve than necessary.

- Trickster eliminates repetitive computations against Prometheus
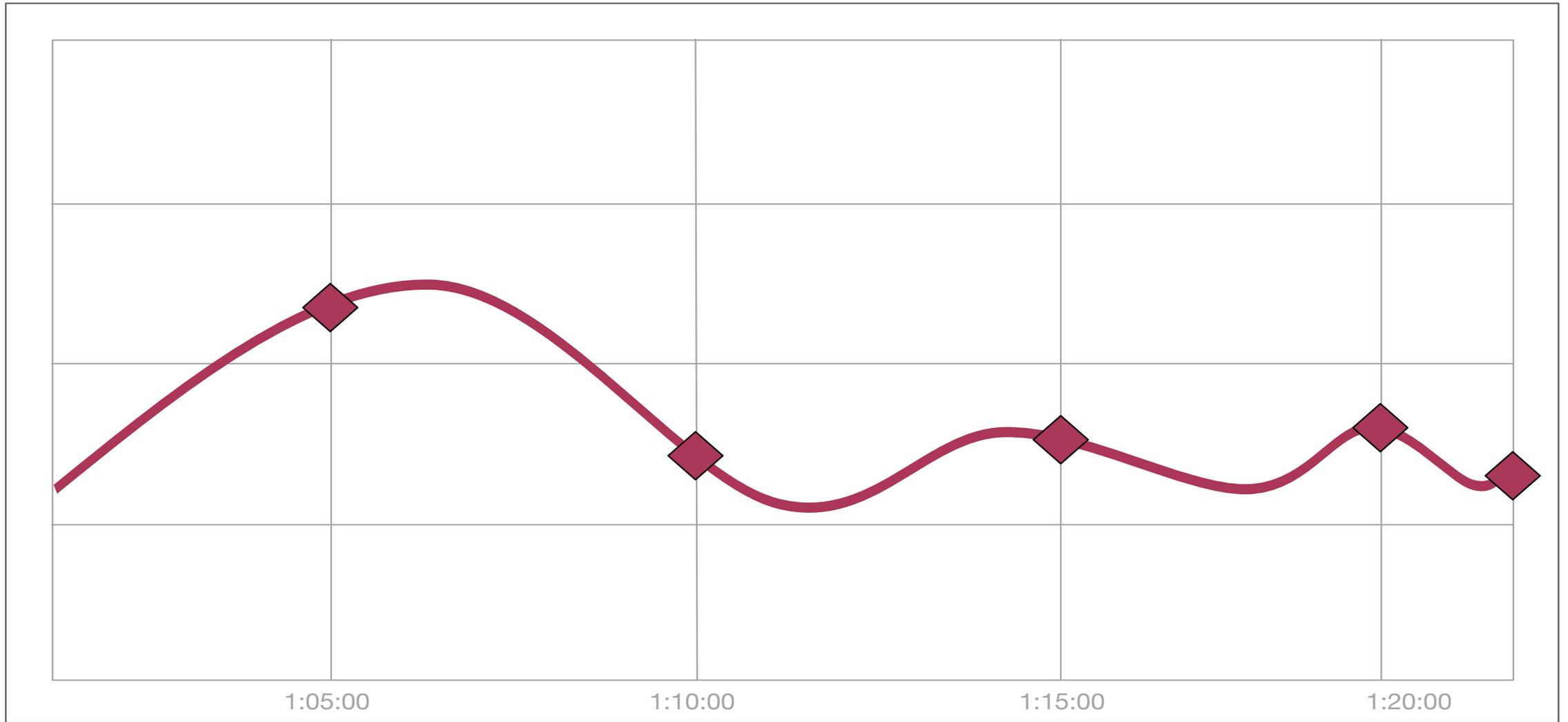
DELTA CACHE

Client requests range 44-56 from Trickster

Trickster determines it has range 45-54

Trickster requests ranges 44-44 & 55-56 from Prometheus

Trickster merges new range into cached range

Trickster extracts range 44-56 and serves to Client
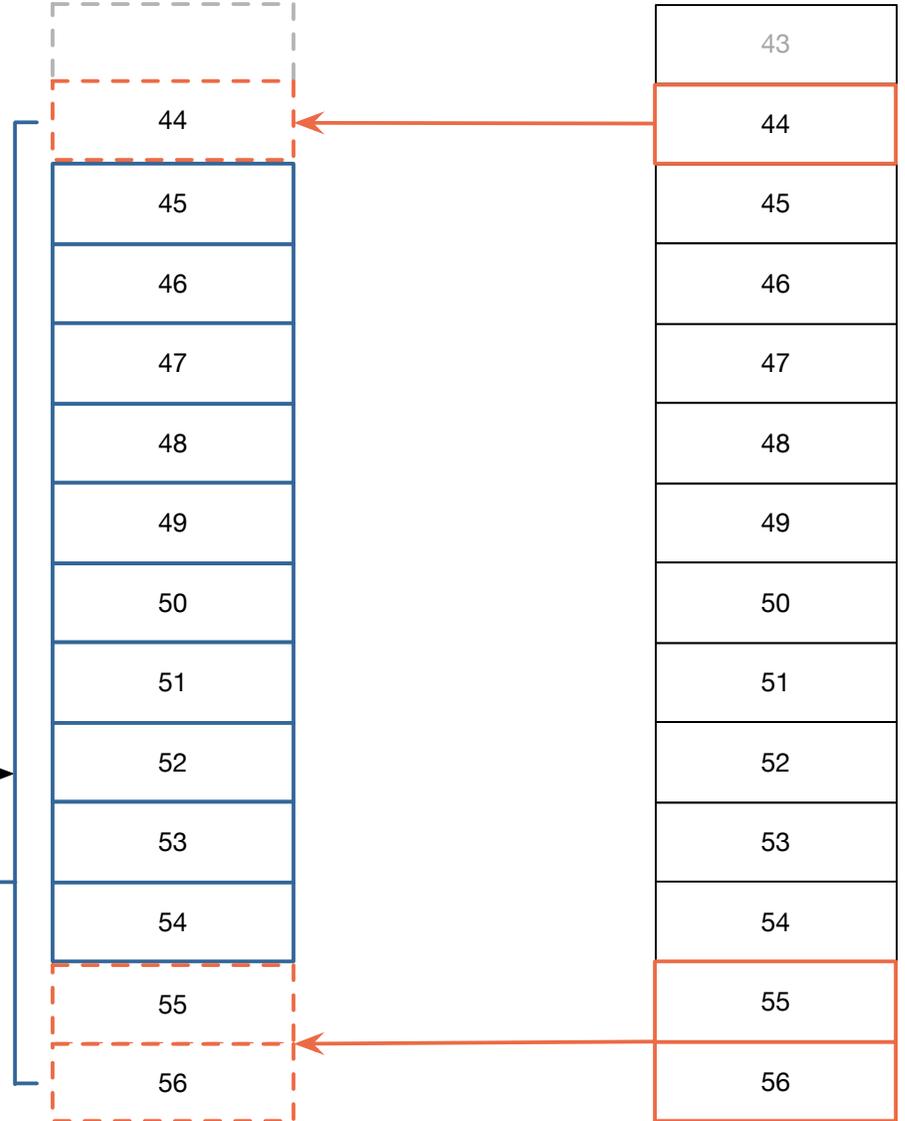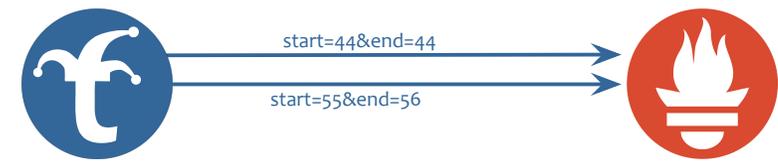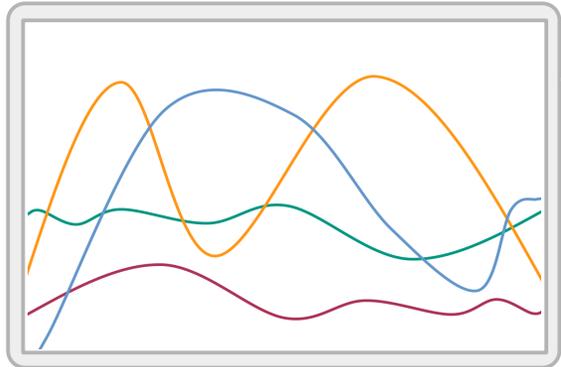
STEP BOUNDARY NORMALIZATION
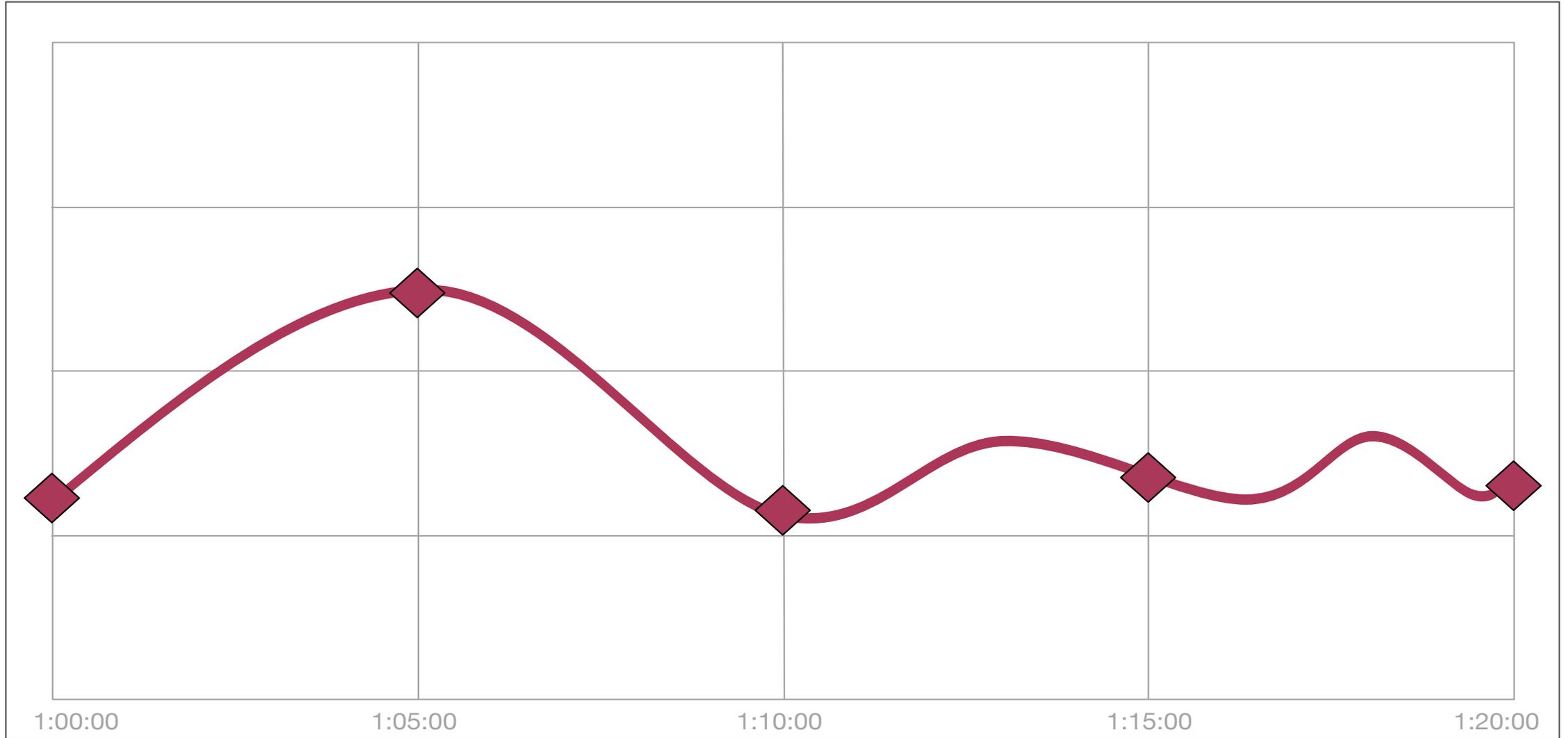
FAST FORWARD (REAL-TIME DATA)

TRICKSTER USE CASES

- **Cardinality Creep** slows down dashboards over time
  - Label Values may not be controlled by the Prometheus or Application Owner
  - Diversity of label values like Device Type and Version grows over time

- Incidental Heavy Dashboard Usage (E.g., Outages)
  - Dashboards see highest usage during outages, requiring significantly more resources than when the monitored system is healthy.
  - **Dashboards are generally least performant during their time of greatest need.**

- Ongoing Heavy Dashboard Usage (e.g., multi-tenant Observability services)
  - Companies provide "as a service" centralized Observability platforms having many users in many Orgs

Live Demo

ROADMAP

- Move to Interfacing Model to support more upstream types
  - InfluxDB
  - ElasticSearch
  - Circonus

- Add Distributed Tracing Support
  - Should support all major players (OpenTracing, OpenCensus, Zipkin)

- High Availability
  - Query multiple HA Prometheus concurrently and merge datasets to fill any gaps

https://github.com/Comcast/trickster/issues

SUPPORTED CACHES

- In-Memory sync.Map

- Filesystem Cache

- bbolt Cache

- Redis (deployed/operated separately from Trickster)

- Cache supports snappy compression (enabled by default)

METRICS & MONITORING

- Trickster is fully instrumented with Prometheus /metrics http endpoint
  - Separate HTTP listener from Proxy, with it's own configurable Port #

- Metrics available about cache size, hit rates, proxy durations
  - `trickster_requests_total{method="query_range", "status"="hit"}`

- Trickster provides /health endpoint for health check clients.

- Grafana Dashboard Template for Trickster Performance & Heath

SUPPORTED DEPLOYMENTS

- Standalone Binary

- Docker Container (hub.docker.com)

- Kubernetes (Example deployment files provided in the project)

- Kube w/ Helm (Example charts provided in the project)

GET INVOLVED

- Clone, Fork and Contribute
  - https://github.com/comcast/trickster

- Join #trickster on the Gophers Slack instance

- Add you name to our Users list in the Readme by submitting a PR

It's important to have a friendly ramp up process with documentation

It's important to provide clear and consistent support

THE LINUX FOUNDATION
OPEN SOURCE SUMMIT
August 29-31, 2018 | Vancouver Convention Centre
OPEN SOURCE. FOR EVERYONE.

8 MAY
Past Meetup
Intro to Trickster, an open source dashboard accelerator for Prometheus
Hosted by Shilla
From Philadelphia Prometheus Meetup
Public group

22 MAR
Past Meetup
GO- Monthly Denver Meetup [Hosted at WeWork on 4th Thursday]
Hosted by Stacie Graves and 3 others
From Mile High Gophers - Denver
Public group

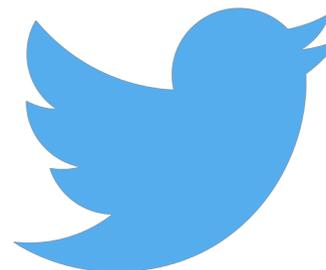KubeCon | CloudNativeCon
North America 2018

Trickster
@TricksterIO — Following ⌄

HEY! We just released Trickster v0.1.0, our first Minor Release, which adds BoltDB cache support by way of @CoreOS's bbolt project. Thanks to all the #TeamTrickster contributors for helping get this one over the finish line!

5:23 PM - 6 Aug 2018

@tricksterio

Announcing Trickster, an Open-Source Dashboard Accelerator for Prometheus

by Shilla Saebi | MAY 30, 2018

Trickster
@TricksterIO — Following ⌄

Here's a great Trickster testimonial from Github: "I have trickster deployed now and wow, what a difference it makes for #grafana." Trickster is so fast and so easy to setup, you can have it accelerating your @PrometheusIO dashboards in under 10 minutes!

comcast.github.io

Aside — ROI of a Sticker

Bulk order of 1,000 stickers costs $347
Based on exposure there's a 1/10 "usage" of stickers
Say 1/100 people are speakers
Get a sticker on a speaker's laptop?
Say they speak at 10 conferences a year.
Average audience size of 1000.
Assume 50% recorded with 5x people watching online.
35,000 views in a year.
Get a keynote speaker?
Average audience size of 10,000.
350,000 views in a year.

Cost per sticker = $0.35
Cost per view = between $0.01-0.001

Cost to produce a piece of content that gets 35,000 views?
1 week of employee ($100,000 a year) is $1,923.
Cost to product a piece of content that gets 350,000 views?
2 weeks of 2 employee ($100,000 a year) is $7,692.

Cost per view = between $0.05-0.076

Stickers offer between
5x and 76x greater return

@mbbroberg

Thank you!

Shilla Saebi @shillasaebi

James Ranson   @ranson