Opencensus

Observability

About me

- Name: Iliia Khaprov
- Twitter: dead_trickster
- Github: deadtrickster
- Other contributions: Prometheus.erl/ex, beam-dashboards

Opencensus

Observability

Observability

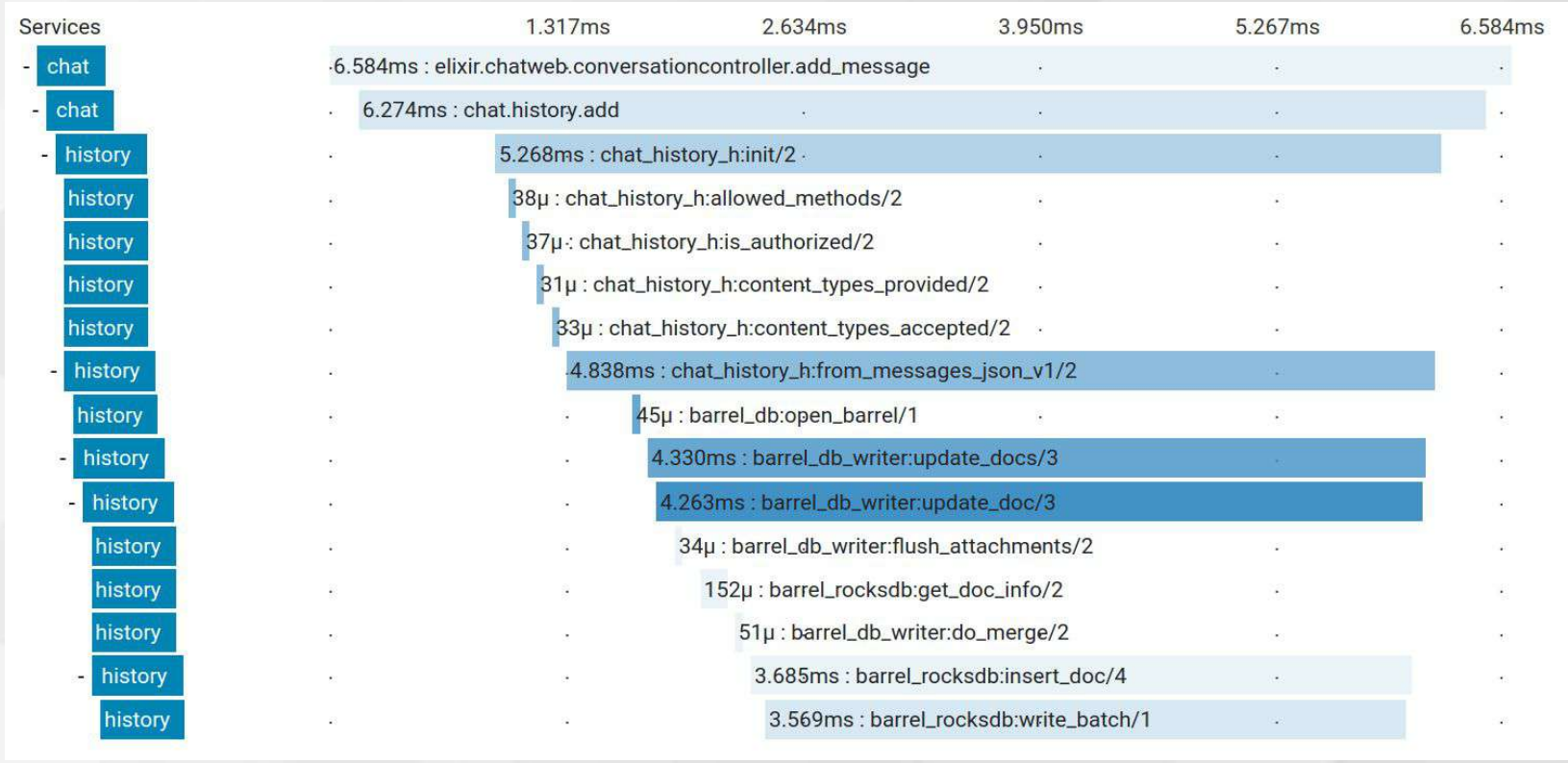
- **Metrics**
- **Logs**
- **Distributed Tracing**
- Analysis
- Visualization

Opencensus

A single distribution of libraries that collect metrics and distributed traces from your services

Tracing

- Span
  - Name
  - SpanId
  - TraceId
  - Start/End time
  - …
- TraceContext
  - TraceId
  - ParentSpantId
- Sampling

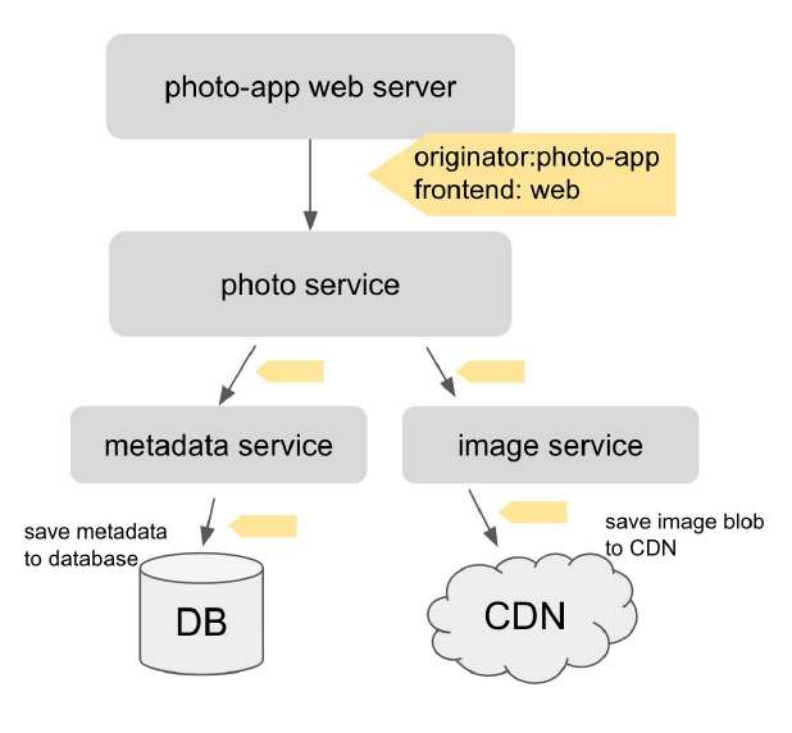


Tags

Key-value pairs to provide contextual information about request.

As request propagates through distributed system context enriched with tags.

Used my stats, routing etc.



Propagation

- Trace Context

  https://www.w3.org/TR/trace-context/

  Defines standard headers and value format to propagate context information that enables distributed tracing scenarios.

- Correlation Context

  https://github.com/w3c/correlation-context

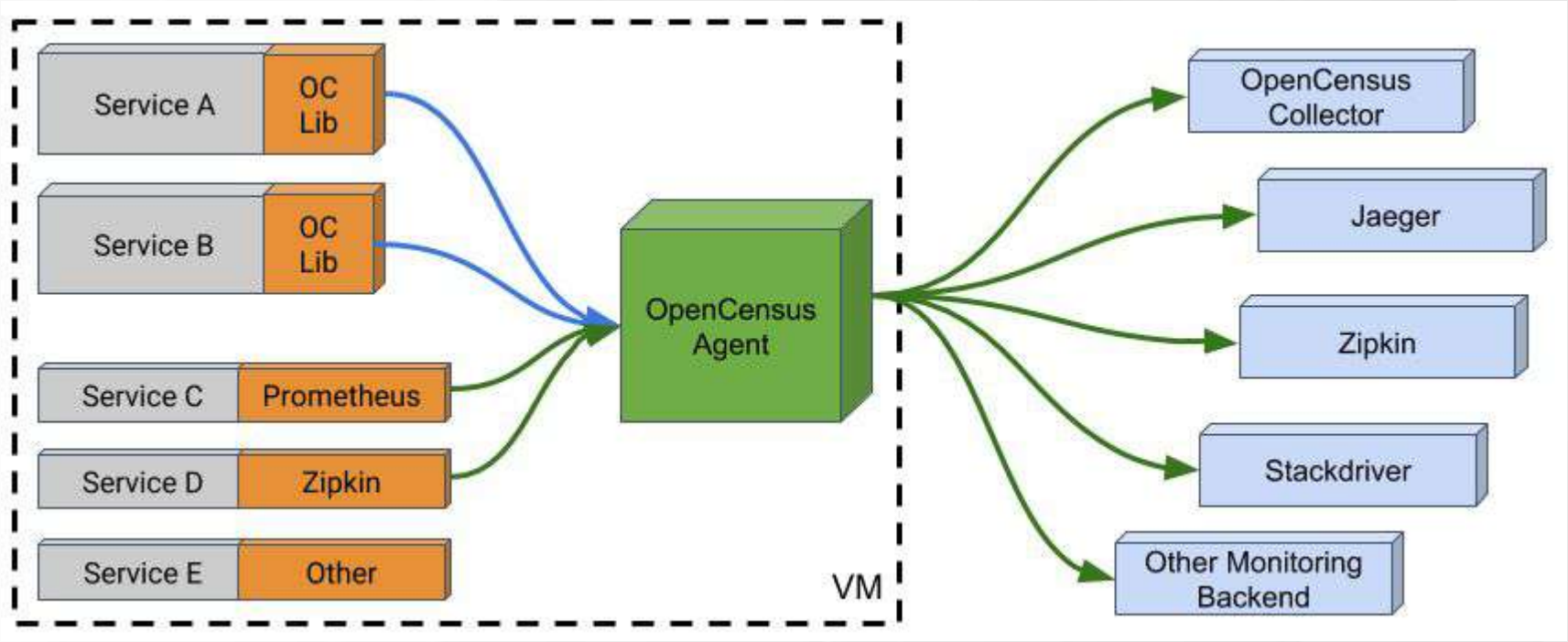
  Defines event correlation using trace context information across systems

Stats/Metrics

- OS/Runtime metrics
- Stats:
  - Multidimensional (tags or labels)
  - Measures decoupled from aggregations via views
    - Count
    - Distribution
    - Sum
    - LastValue
  - Views integrated with TagContext

Service

Set of components that can collect traces and metrics, do aggregation/smart sampling and export to different backends



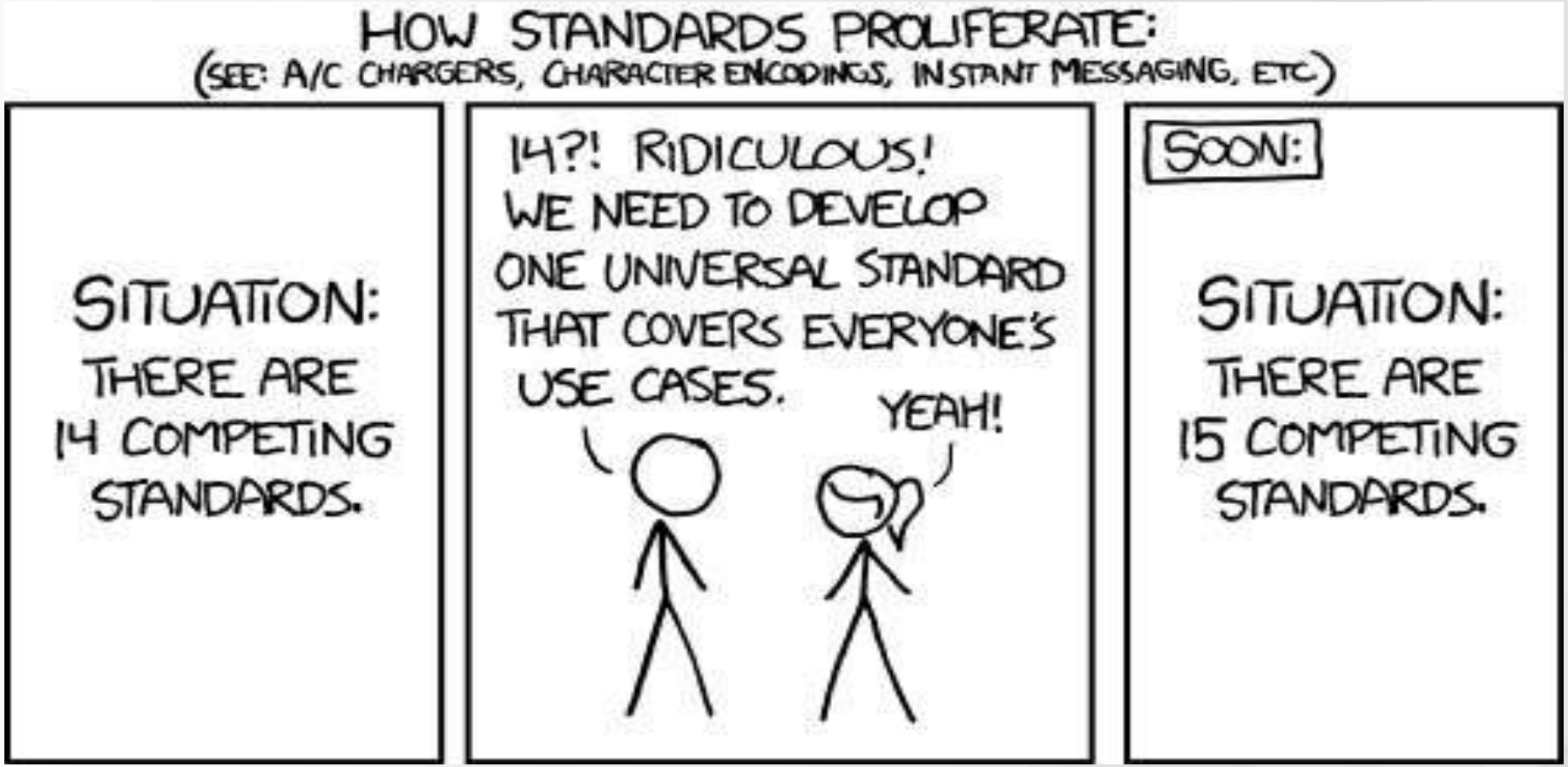
Observability

What's more

- Opencensus.io
  - Tutorials for various languages
  - Blogs
  - Community resources
- Github:census-ecosystem (additional exporters, demo, etc)

Opencensus & Opentracing

Opencesus & Opentracing



Opencensus & Opentracing

- Opencensus is more decoupled – there are separate APIs for tracing, metrics and _soon_ logs. Opentracing OTOH encourages using the same tracing API for everything
- Opentracing leaving implementation of tracers to vendors. This might be good but often behavior is different enough. Opencensus has implementations with exporters.
- Opencensus relies on standard propagation format.

Opencensus & Opentracing

Still too similar?

Opencensus & Opentracing

Merge as OpenTelemetry!

OpenTelemetry

- A new, unified set of libraries and specifications for observability telemetry.
- The Java reference implementation is available, and cross-language work will began on May 8th, 2019.
- By the September 2019, the plan is to reach parity with existing projects for Erlang, C#, Golang, Java, NodeJS, and Python. Please contribute!
- When each language reaches parity, the corresponding OpenTracing and OpenCensus project will be sunset. OC and OT support will continue for two years, via a backwards compatibility bridge.

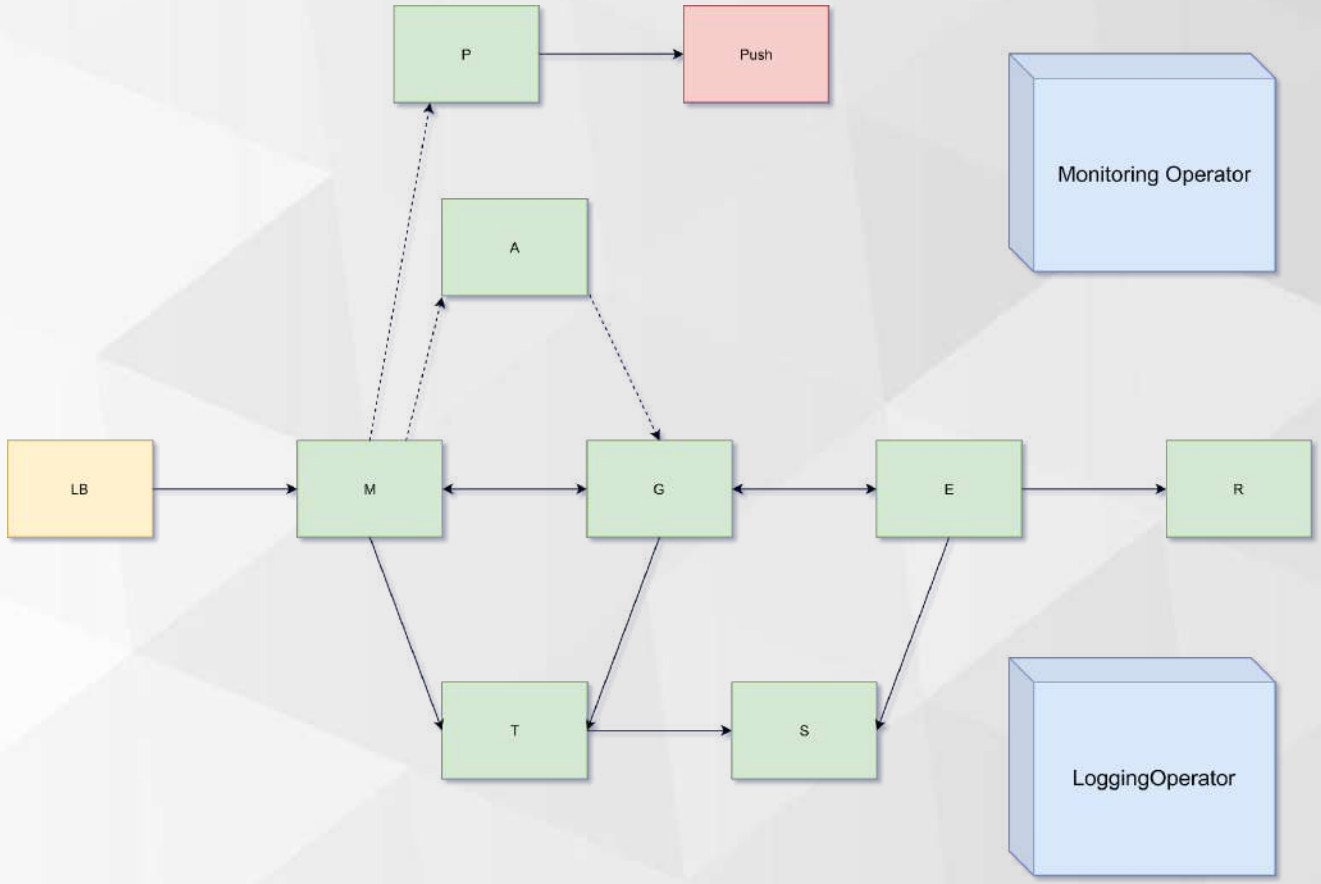
How a system becomes observable

All connections, similarities are accidental (all possible small fonts follow).

A Monolith

- Supervisor process checks if pid alive
- Prometheus OS agents
- Basic alerts: disk space, RAM, etc
- Elasticsearch-Logstash-Kibana
- **Logs are useless?**

Microservices



Microservices + Opencensus

- We are fully comfortable with metrics, and dashboards.
- We started to see execution path. And more bugs and bottlenecks.
- When we turned on Distributed Tracing everyone was so excited so our tracing backend couldn’t keep up with data.
- Too many **boring** traces!

BEAM is amazing!

dead_trickster

CodeBEAMSTO

opencensus-beam

Conclusions so far

- Operating is hard! We tried:
  - Jaeger/Cassandra
  - Zipkin/Elasticsearch
- Sampling is hard
  - Adaptive sampling?
  - Tail sampling?
- Managing configuration and data flows is hard
  - oc-agent may help
- Context is awesome!

It’s not only about microservices

Dependencies are important too!

- ~50 dependencies
- ~4 have instrumentation hooks
- Cowboy has middlewares
- Elixir is in a better shape

Community

- Opencensus.io
- Spec: github:census-instrumentation/opencensus-proto
- Gitter: census-instrumentation/Lobby
- Library: **github:census-instrumentation/opencensus-erlang**
- Hex.pm: opencensus
- Integrations: **github:opencensus-beam**
- Slack: #opencensus (erlanger, elixir-lang)
- Dashboards: github:deadtrickster/beam-dashboards

Opencensus BEAM

Integrations

- Elli
- Cowboy
- SpaceTime-IoT/pgo
- tsloughter/grpcbox
- Plugs
- Phoenix
- Ecto
- Absinthe
- Tesla
- Telemetry
- logger

Exporters/Reporters

- Prometheus
- Stackdriver
- Zipkin
- Jaeger
- Datadog
- Opencensus Service
- InfluxDB

What's missing?

- More integrations
- More exporters
- ZPages (maybe observer_cli integration?)
- More patterns
  - Context propagation
    - Pools
    - Web servers
    - gen_*
  - Using span fields properly
  - Deploying

Observability WG

EEF Observability WG: Objectives

- Improve runtime observability through integration with tools like Jaeger, LightStep, Stackdriver — without vendor lock-in
- Improve state of whitebox monitoring of BEAM applications
- Review possibilities to advance blackbox monitoring of BEAM (dtrace, eBPF)
- Provide common interfaces to gathering and viewing VM and application statistics and traces
- Cooperate with major OAM providers like New Relic to provide proper commercial support for Erlang monitoring

Observability WG: Participants

- Opencensus
- New Relic
- Spandex
- Telemetry
- Lager
- Prometheus.erl

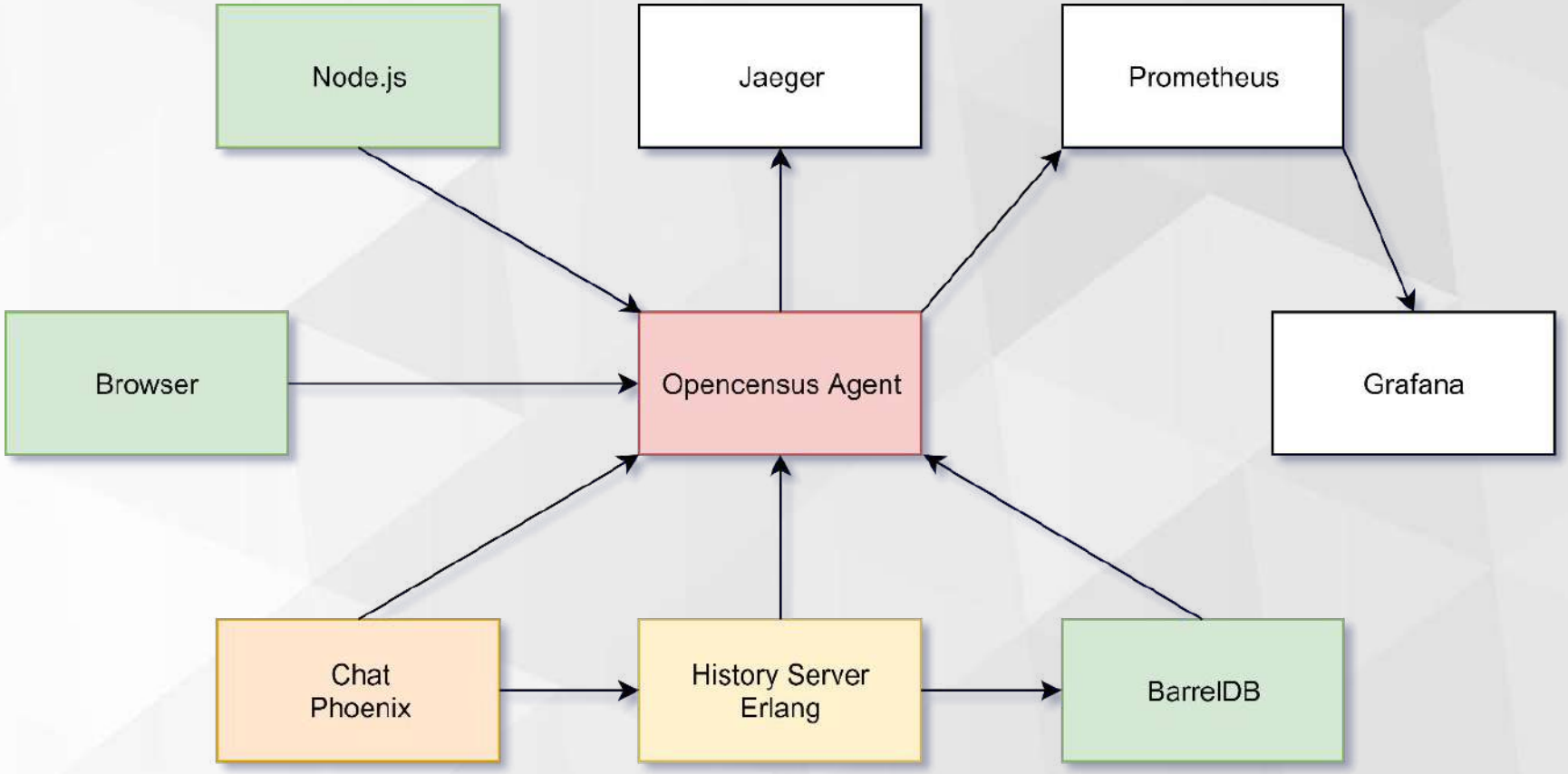
Thanks to Tristan Sloughter for leading the effort

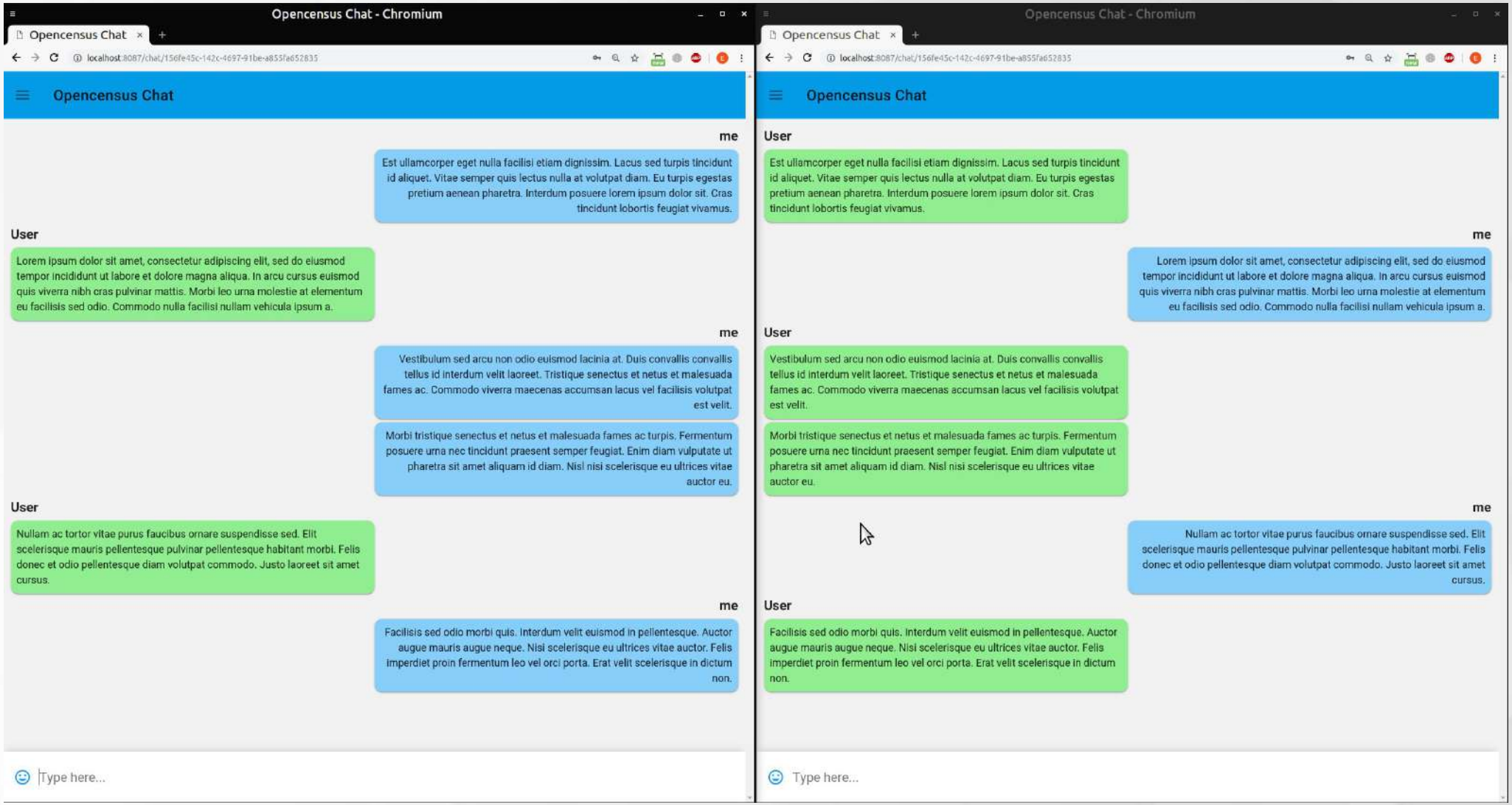
An Example

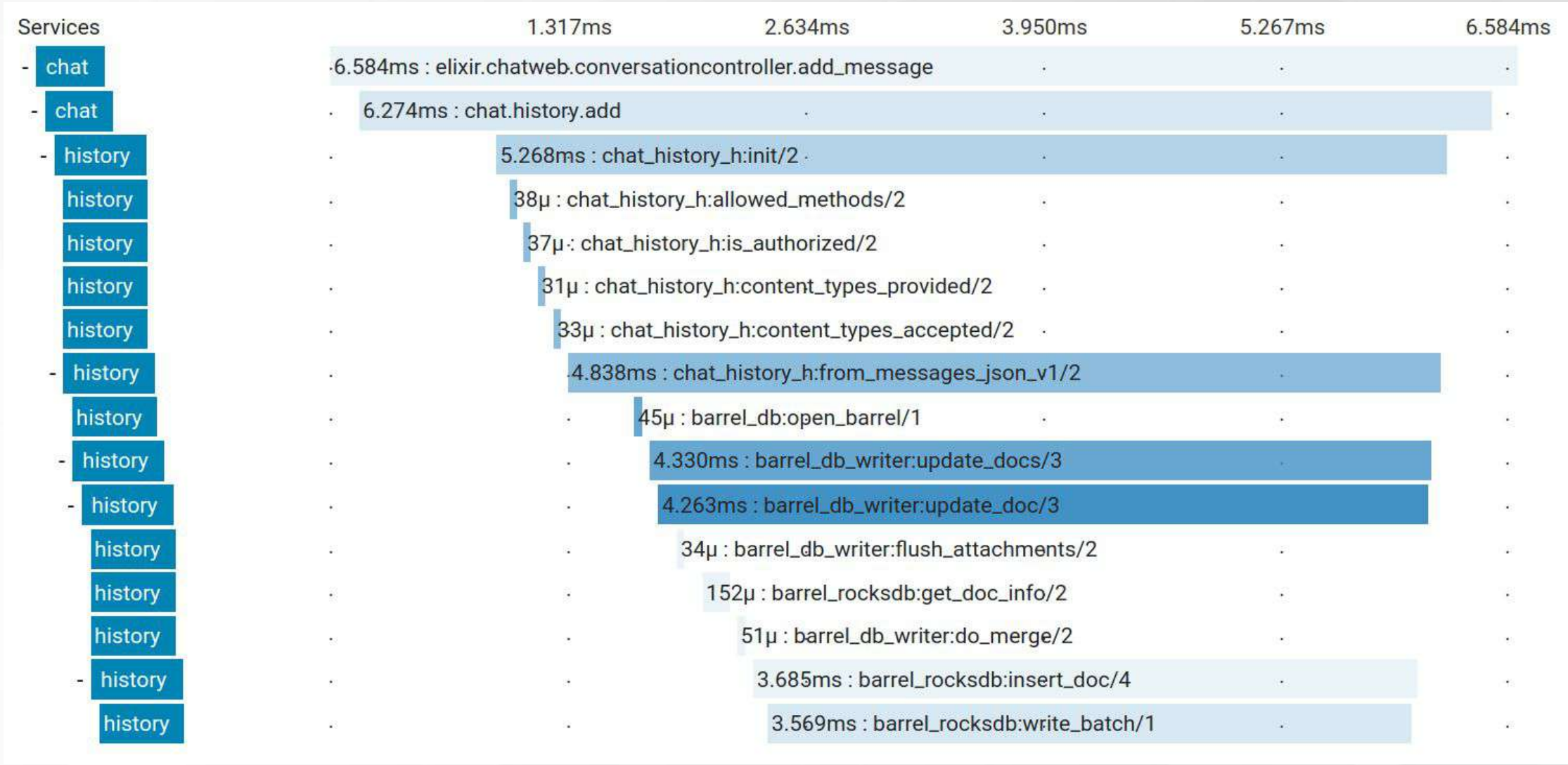
A chat app

https://github.com/opencensus-beam/opencensus-chat

Chat: diagram




Opencensus Chat - Chromium
Opencensus Chat
localhost:8087/chat/156fe45c-142c-4697-91be-a855fa652835
Opencensus Chat
me
Est ullamcorper eget nulla facilisi etiam dignissim. Lacus sed turpis tincidunt id aliquet. Vitae semper quis lectus nulla at volutpat diam. Eu turpis egestas pretium aenean pharetra. Interdum posuere lorem ipsum dolor sit. Cras tincidunt lobortis feugiat vivamus.
User
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. In arcu cursus euismod quis viverra nibh cras pulvinar mattis. Morbi leo urna molestie at elementum eu facilisis sed odio. Commodo nulla facilisi nullam vehicula ipsum a.
me
Vestibulum sed arcu non odio euismod lacinia at. Duis convallis convallis tellus id interdum velit laoreet. Tristique senectus et netus et malesuada fames ac. Commodo viverra maecenas accumsan lacus vel facilisis volutpat est velit.
Morbi tristique senectus et netus et malesuada fames ac turpis. Fermentum posuere urna nec tincidunt praesent semper feugiat. Enim diam vulputate ut pharetra sit amet aliquam id diam. Nisl nisi scelerisque eu ultrices vitae auctor eu.
User
Nullam ac tortor vitae purus faucibus ornare suspendisse sed. Elit scelerisque mauris pellentesque pulvinar pellentesque habitant morbi. Felis donec et odio pellentesque diam volutpat commodo. Justo laoreet sit amet cursus.
me
Facilisis sed odio morbi quis. Interdum velit euismod in pellentesque. Auctor augue mauris augue neque. Nisi scelerisque eu ultrices vitae auctor. Felis imperdiet proin fermentum leo vel orci porta. Erat velit scelerisque in dictum non.
Type here...
Opencensus Chat - Chromium
Opencensus Chat
localhost:8087/chat/156fe45c-142c-4697-91be-a855fa652835
Opencensus Chat
User
Est ullamcorper eget nulla facilisi etiam dignissim. Lacus sed turpis tincidunt id aliquet. Vitae semper quis lectus nulla at volutpat diam. Eu turpis egestas pretium aenean pharetra. Interdum posuere lorem ipsum dolor sit. Cras tincidunt lobortis feugiat vivamus.
me
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. In arcu cursus euismod quis viverra nibh cras pulvinar mattis. Morbi leo urna molestie at elementum eu facilisis sed odio. Commodo nulla facilisi nullam vehicula ipsum a.
User
Vestibulum sed arcu non odio euismod lacinia at. Duis convallis convallis tellus id interdum velit laoreet. Tristique senectus et netus et malesuada fames ac. Commodo viverra maecenas accumsan lacus vel facilisis volutpat est velit.
Morbi tristique senectus et netus et malesuada fames ac turpis. Fermentum posuere urna nec tincidunt praesent semper feugiat. Enim diam vulputate ut pharetra sit amet aliquam id diam. Nisl nisi scelerisque eu ultrices vitae auctor eu.
me
Nullam ac tortor vitae purus faucibus ornare suspendisse sed. Elit scelerisque mauris pellentesque pulvinar pellentesque habitant morbi. Felis donec et odio pellentesque diam volutpat commodo. Justo laoreet sit amet cursus.
User
Facilisis sed odio morbi quis. Interdum velit euismod in pellentesque. Auctor augue mauris augue neque. Nisi scelerisque eu ultrices vitae auctor. Felis imperdiet proin fermentum leo vel orci porta. Erat velit scelerisque in dictum non.
Type here...



dead_trickster

CodeBEAMSTO

opencensus-beam

Chat: Dependencies

```
{opencensus, "~> 0.6"},
{opencensus_cowboy, "~> 0.2"},
{opencensus_erlang_prometheus, "~> 0.3"},
{opencensus_jaeger, "~> 0.0.1"},
{opencensus_zipkin, "~> 0.1.0"},

{prometheus, "~> 4.0"},
{prometheus_cowboy, "~> 0.1"},
{prometheus_httpd, "~> 2.1"},
{prometheus_process_collector, "~> 1.3"}
```

```
{:opencensus, "~> 0.9.0"},
{:opencensus_cowboy, "~> 0.3.0"},
{:opencensus_plug, "~> 0.3.0"},
{:opencensus_phoenix, "~> 0.2.0"},
{:opencensus_jaeger, "~> 0.0.1"},
{:opencensus_zipkin, "~> 0.1.0"},

{:prometheus_ex, "~> 3.0"},
{:prometheus_plugs, "~> 1.1"},
{:opencensus_erlang_prometheus, "~> 0.3.2"}
```

Chat: configuration&integration

```
{opencensus, [
  {reporters, [
    {oc_reporter_zipkin, [
      {address, "http://localhost:9411/api/v2/spans"},
      {local_endpoint, #{<<"serviceName">> => <<"history">>}}
    ]}
  ]}
]}

cowboy:start_clear(
  chat_history,
  [{port, as_integer(Port)}, {num_acceptors, Acceptors}],
  #{
    env => #{dispatch => dispatch()},
    middlewares => [
      opencensus_cowboy2_context,
      cowboy_router,
      cowboy_handler
    ],
    metrics_callback => fun prometheus_cowboy2_instrumenter:observe/1,
    stream_handlers => [cowboy_metrics_h, cowboy_tracer_h, cowboy_stream_h]
  }
)
```

```
config :opencensus, :reporters,
  oc_reporter_zipkin:
    [address: 'http://localhost:9411/api/v2/spans',
     local_endpoint: %{"serviceName" => "chat"}]

config :opencensus, :sampler, {:oc_sampler_always, []}

defmodule ChatWeb.Observability.Plug.MetricsExporter do
  use Prometheus.PlugExporter
end

defmodule ChatWeb.Observability.Plug.Metrics do
  use Opencensus.Plug.Metrics
end

defmodule ChatWeb.Observability.Plug.Traces do
  use Opencensus.Plug.Trace
end

plug ChatWeb.Observability.Plug.Metrics
plug ChatWeb.Observability.Plug.Traces
```

Chat: goals

Create more microservices to show more integrations!

- Http servers, frameworks interfaces (Elli, Rax)
- More databases (Ecto, drivers)
- Different http clients
- Pools, queues, etc

Deployment patterns

- Kubernetes
- Service Meshes

Thank you!

- Evgeny Derevianko
- Benoit Chesneau
- Tristan Sloughter
- Erik Dahmen

Background designed by vexels.com

Hashtag icon made by Freepik from www.flaticon.com

Standards XKCD https://xkcd.com/927/

Twitter and Github logos are from respective official sets

Diagrams made in draw.io

Questions?