Grafana Labs

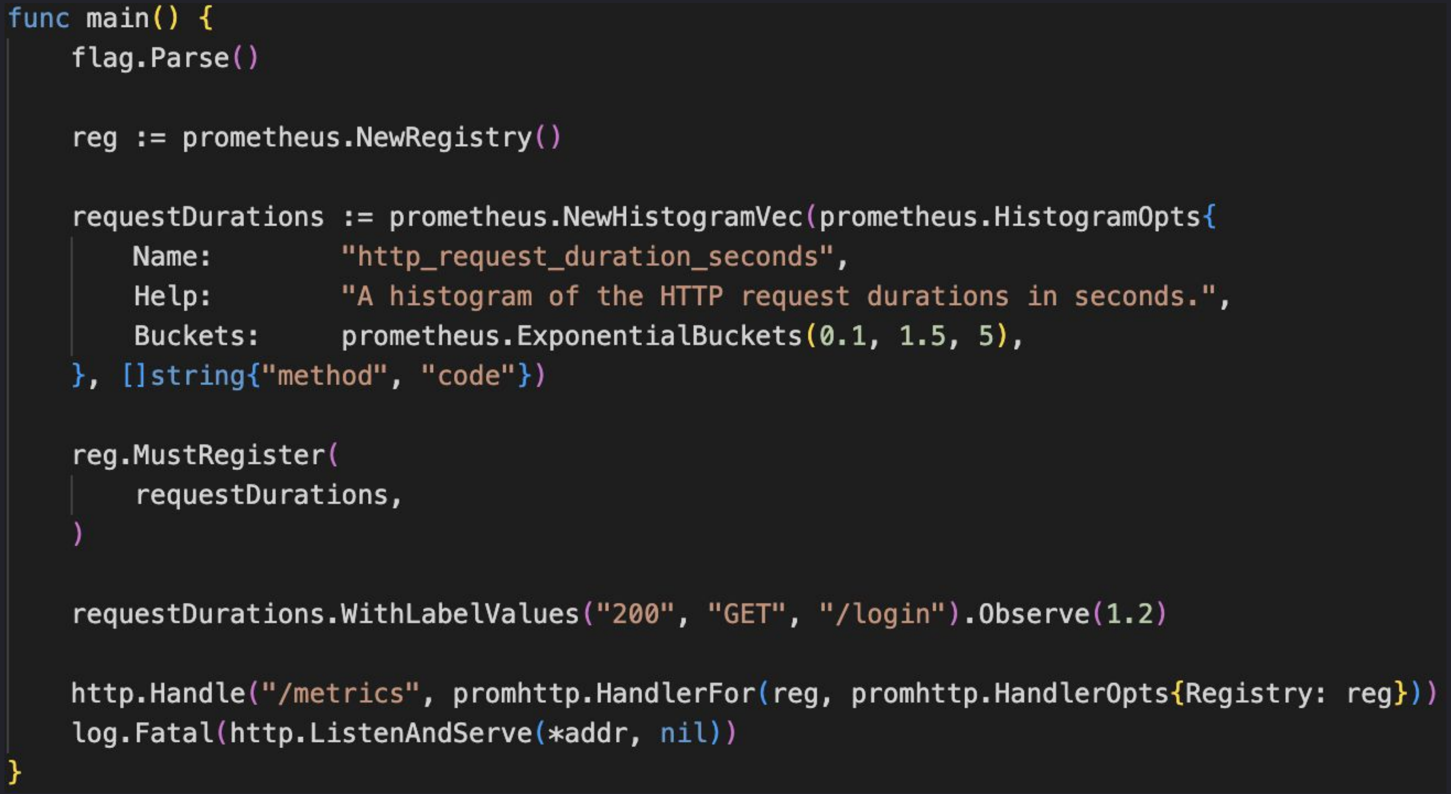
Schema driven Observability with OpenTelemetry Weaver

Dominik Süß
DevEx Engineer @ Grafana Labs

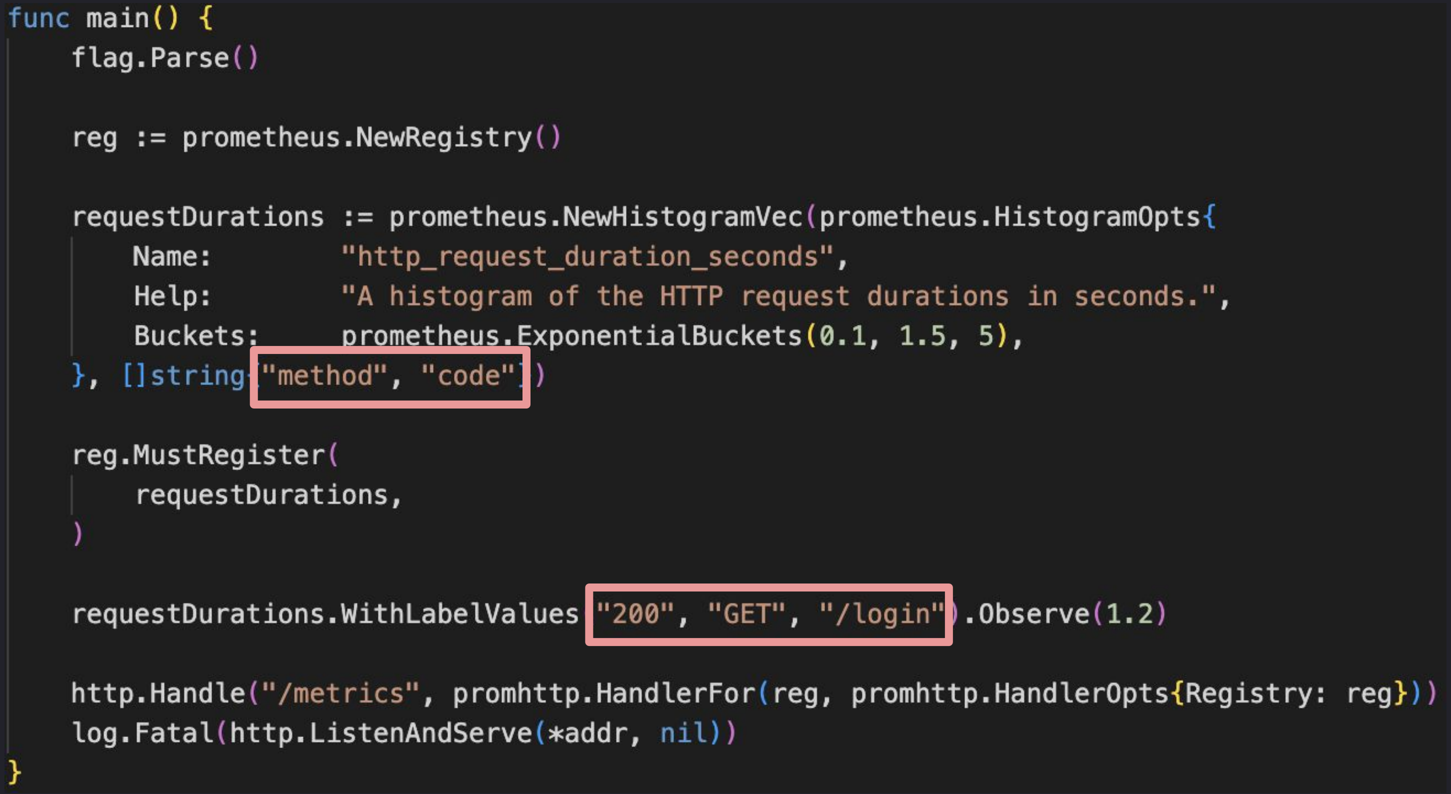
Grafana Labs

Schema driven Observability with OpenTelemetry Weaver

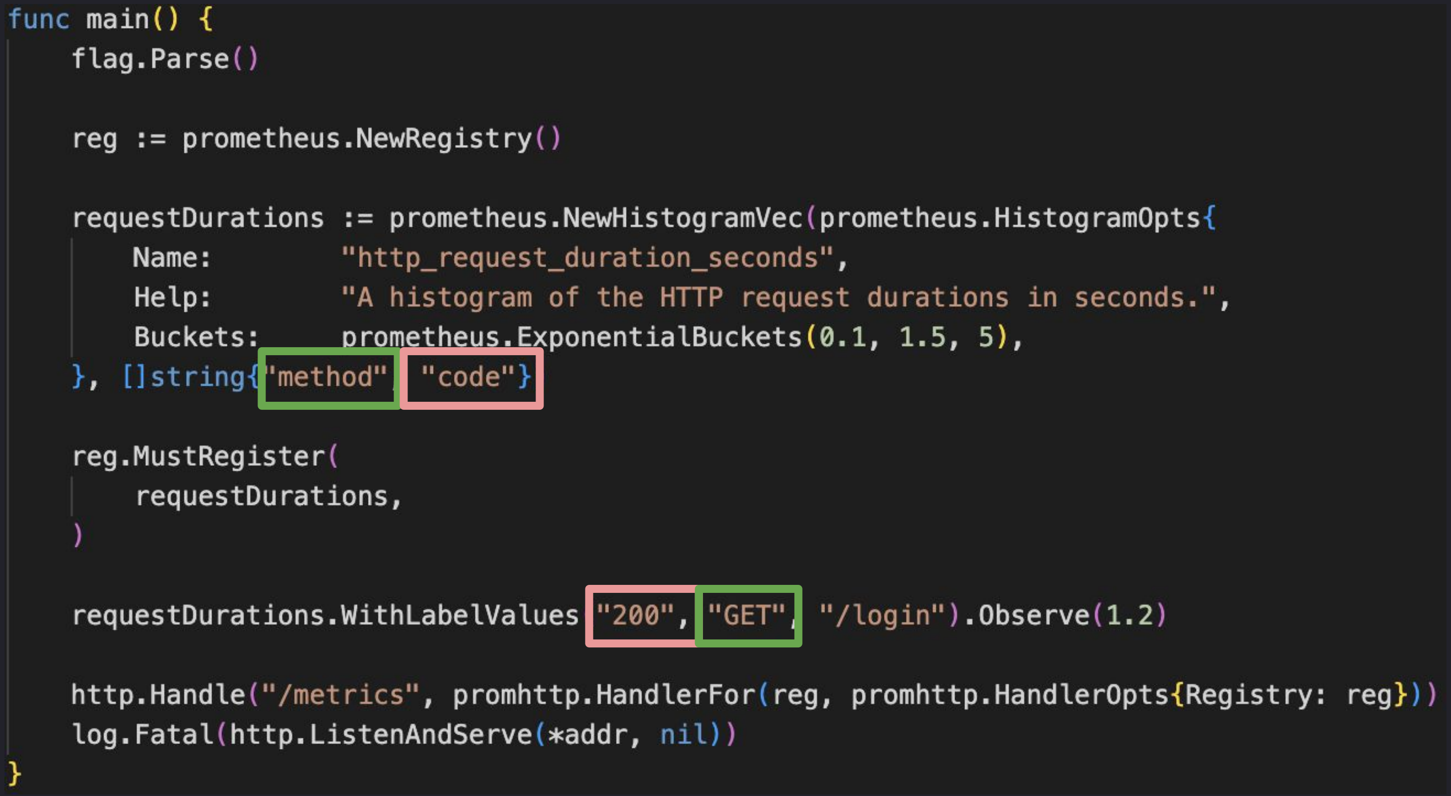
Dominik Süß
DevEx Engineer @ Grafana Labs



```go
func main() {
    flag.Parse()

    reg := prometheus.NewRegistry()

    requestDurations := prometheus.NewHistogramVec(prometheus.HistogramOpts{
        Name:        "http_request_duration_seconds",
        Help:        "A histogram of the HTTP request durations in seconds.",
        Buckets:     prometheus.ExponentialBuckets(0.1, 1.5, 5),
    }, []string{"method", "code"})

    reg.MustRegister(
        requestDurations,
    )

    requestDurations.WithLabelValues("200", "GET", "/login").Observe(1.2)

    http.Handle("/metrics", promhttp.HandlerFor(reg, promhttp.HandlerOpts{Registry: reg}))
    log.Fatal(http.ListenAndServe(*addr, nil))
}
```



```go
func main() {
    flag.Parse()

    reg := prometheus.NewRegistry()

    requestDurations := prometheus.NewHistogramVec(prometheus.HistogramOpts{
        Name:       "http_request_duration_seconds",
        Help:       "A histogram of the HTTP request durations in seconds.",
        Buckets:    prometheus.ExponentialBuckets(0.1, 1.5, 5),
    }, []string{"method", "code"})

    reg.MustRegister(
        requestDurations,
    )

    requestDurations.WithLabelValues("200", "GET", "/login").Observe(1.2)

    http.Handle("/metrics", promhttp.HandlerFor(reg, promhttp.HandlerOpts{Registry: reg}))
    log.Fatal(http.ListenAndServe(*addr, nil))
}
```



```go
func main() {
    flag.Parse()

    reg := prometheus.NewRegistry()

    requestDurations := prometheus.NewHistogramVec(prometheus.HistogramOpts{
        Name:       "http_request_duration_seconds",
        Help:       "A histogram of the HTTP request durations in seconds.",
        Buckets:    prometheus.ExponentialBuckets(0.1, 1.5, 5),
    }, []string{"method", "code"})

    reg.MustRegister(
        requestDurations,
    )

    requestDurations.WithLabelValues("200", "GET", "/login").Observe(1.2)

    http.Handle("/metrics", promhttp.HandlerFor(reg, promhttp.HandlerOpts{Registry: reg}))
    log.Fatal(http.ListenAndServe(*addr, nil))
}
```

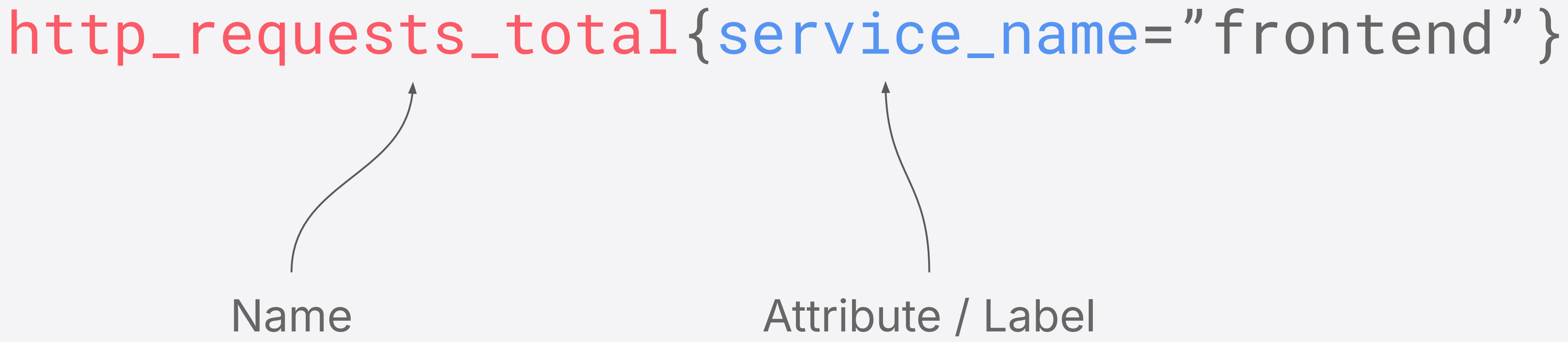
Two hard things in software development

- Naming things
- Cache invalidation
- Off-by-one errors

Two hard things in software development

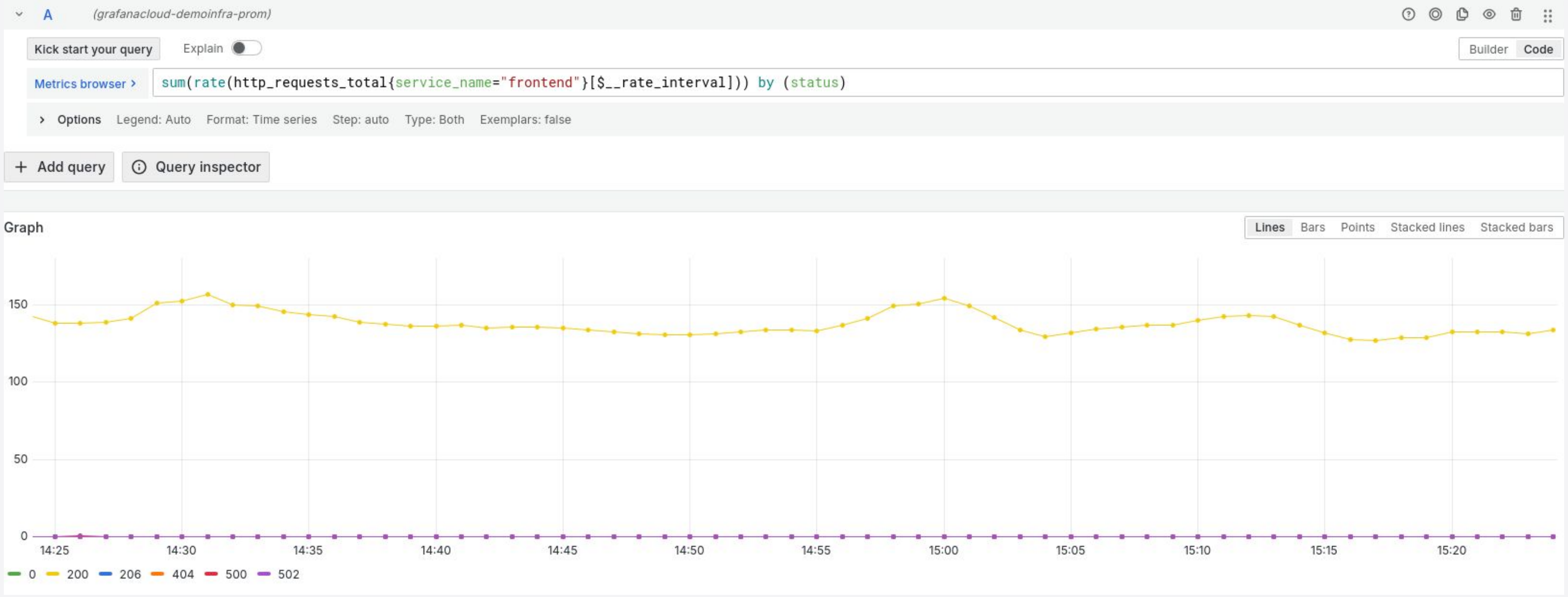
- ***Naming things***
- Cache invalidation
- Off-by-one errors

Anatomy of a (prometheus) metric



```
http_requests_total{service_name=”frontend”}
```

Telemetry sprawl


A (grafanacloud-demoinfra-prom)

Kick start your query | Explain | Builder | Code

Metrics browser ›

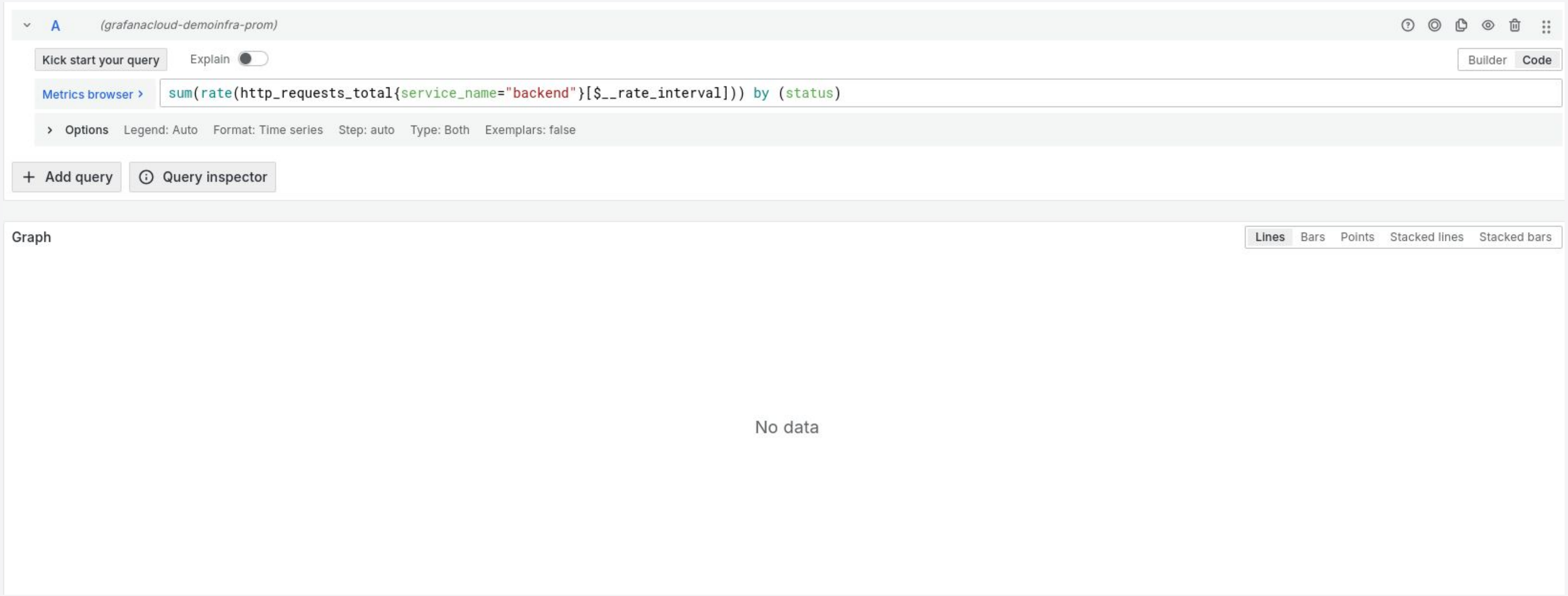
```
sum(rate(http_requests_total{service_name="frontend"}[$__rate_interval])) by (status)
```

› Options Legend: Auto Format: Time series Step: auto Type: Both Exemplars: false

+ Add query | Query inspector

Graph

Uh oh...



A (grafanacloud-demoinfra-prom)

Kick start your query   Explain

Builder   Code

Metrics browser >

```
sum(rate(http_requests_total{service_name="backend"}[$__rate_interval])) by (status)
```

> Options   Legend: Auto   Format: Time series   Step: auto   Type: Both   Exemplars: false

+ Add query   Query inspector

Graph

Lines   Bars   Points   Stacked lines   Stacked bars

No data

Grafana Labs

Schema driven Observability with OpenTelemetry Weaver

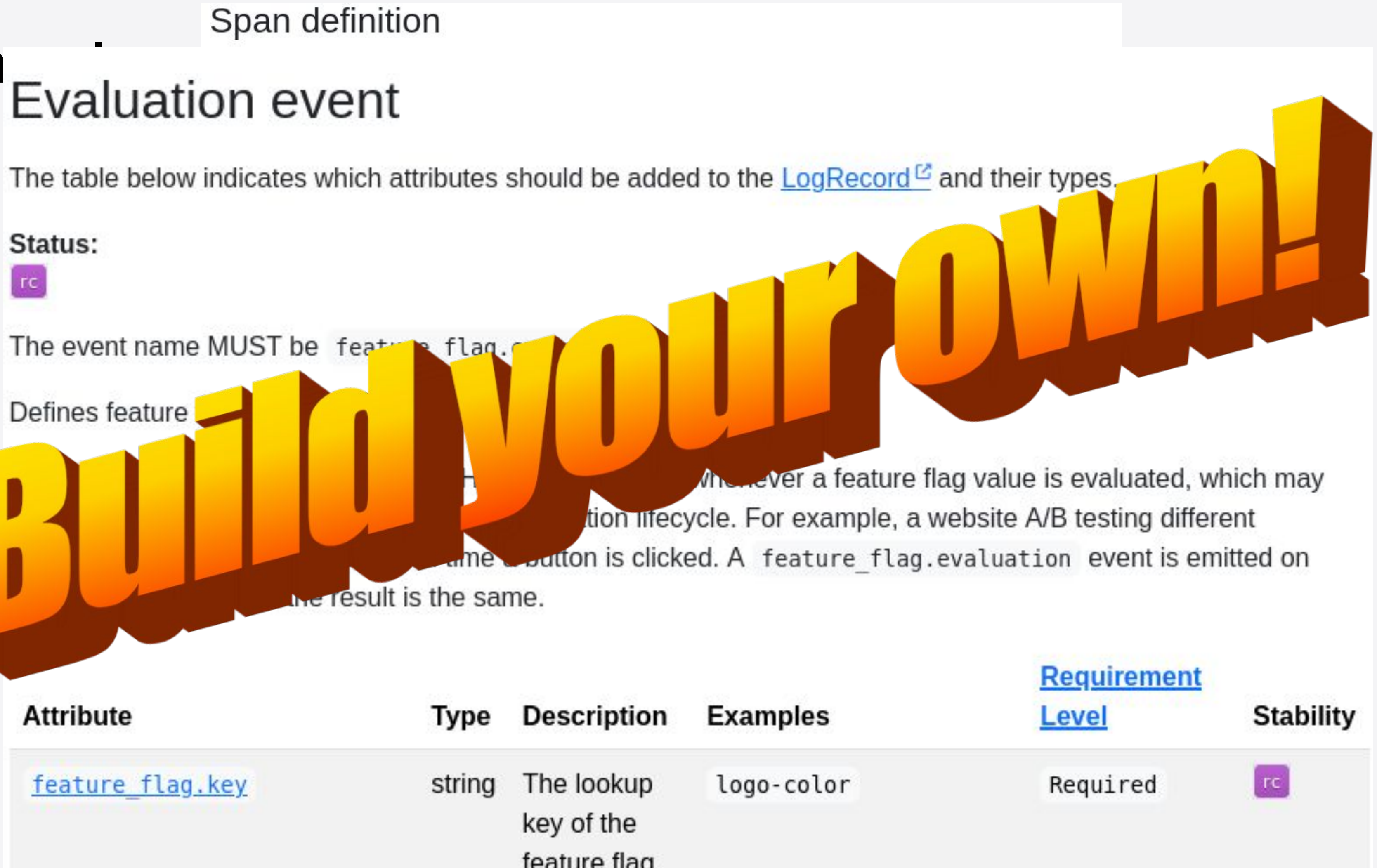
Dominik Süß
DevEx Engineer @ Grafana Labs

OpenTelemetry

Semantic Conventions

- Agreed upon names for common signals
- Versioned
- Community Maintained
- Many namespaces already populated
  - Networking, HTTP, Hardware, CICD,...

Exa



Span definition

Evaluation event

The table below indicates which attributes should be added to the LogRecord and their types.

Status:

rc

The event name MUST be `feat` `flag.`

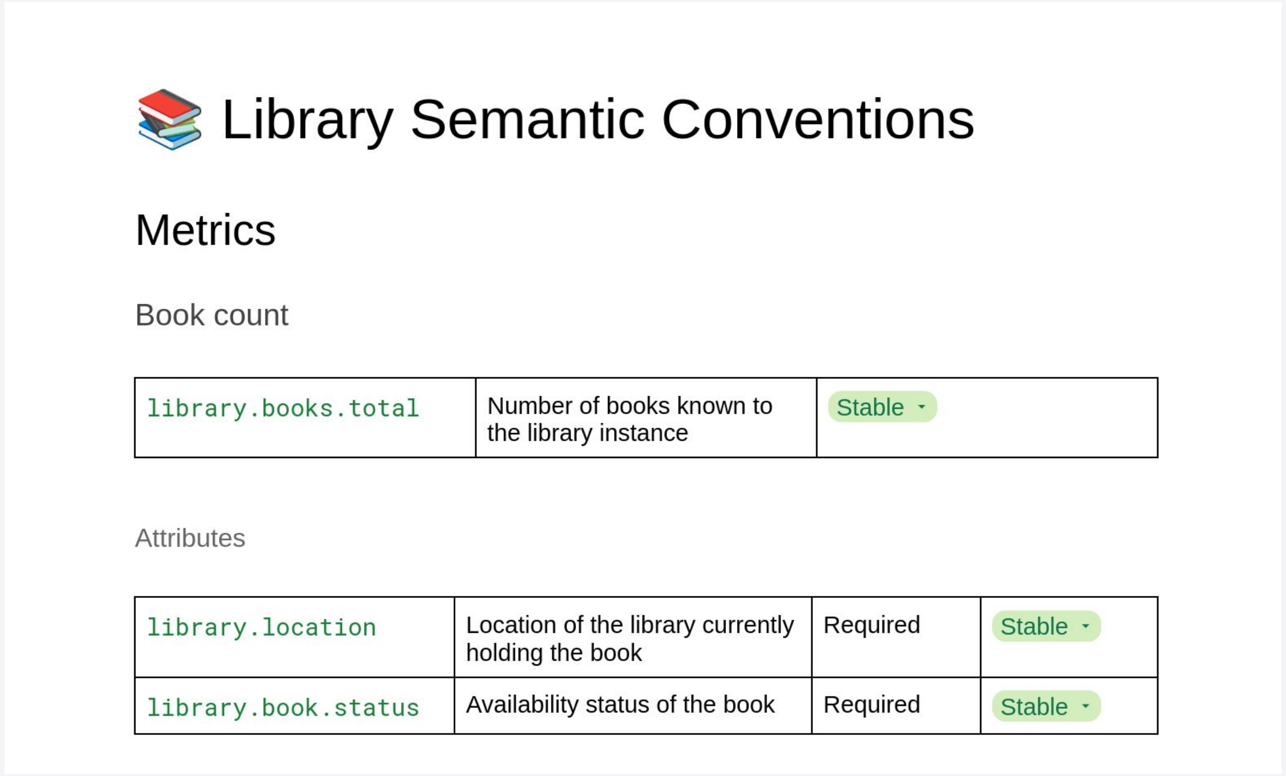
Defines feature

H whenever a feature flag value is evaluated, which may tion lifecycle. For example, a website A/B testing different me button is clicked. A `feature_flag.evaluation` event is emitted on result is the same.

Attribute	Type	Description	Examples	Requirement Level	Stability
`feature_flag.key`	string	The lookup key of the feature flag	`logo-color`	`Required`	rc

Build your own!

My first convention



📚 Library Semantic Conventions

Metrics

Book count

library.books.total	Number of books known to the library instance	Stable

Attributes

library.location	Location of the library currently holding the book	Required	Stable
library.book.status	Availability status of the book	Required	Stable

Challenges

- Cross-referencing attributes
- Keeping the schema up to date
- Ownership process
- Ensuring teams adopt schema

Grafana Labs

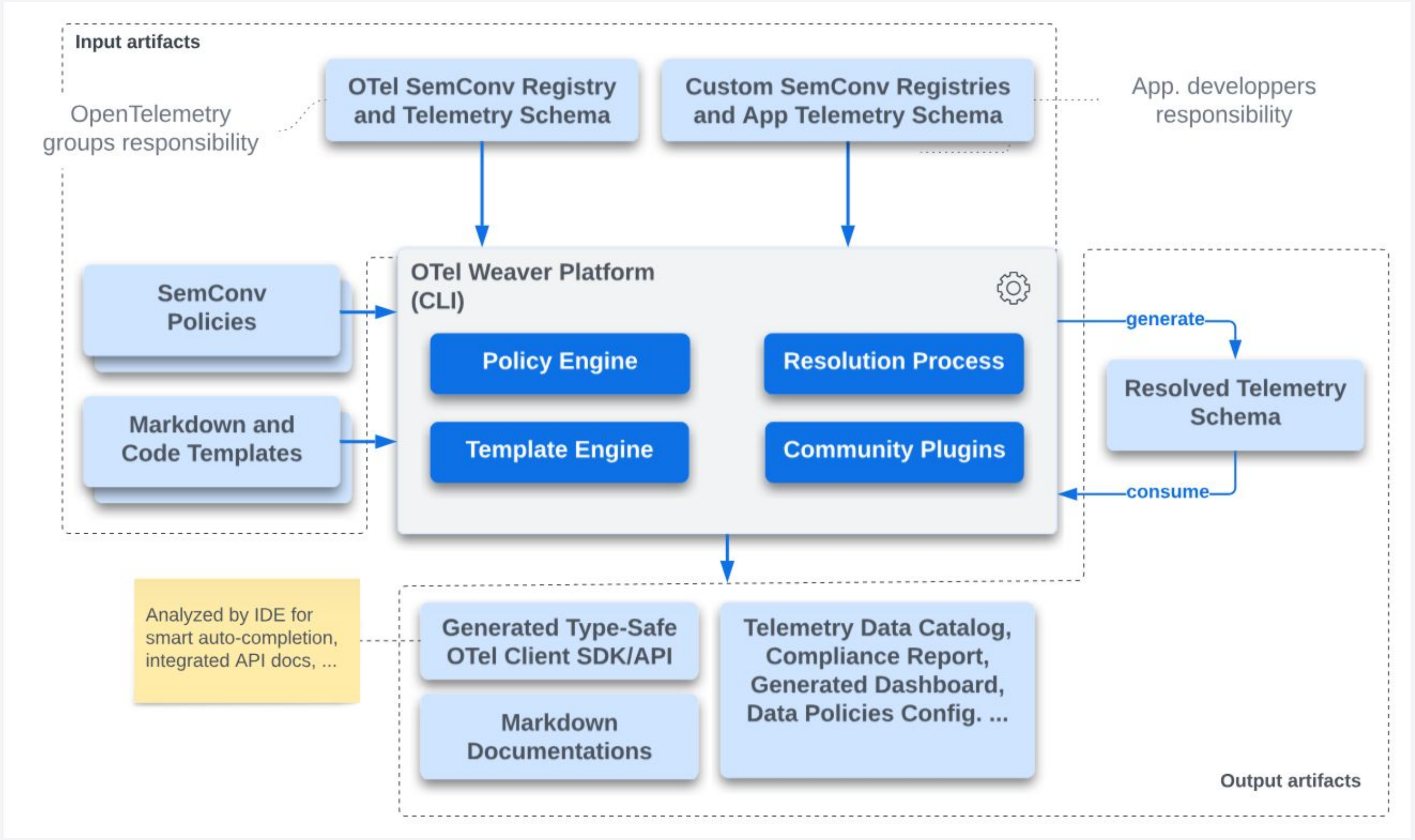
Schema driven Observability with OpenTelemetry Weaver

Dominik Süß
DevEx Engineer @ Grafana Labs

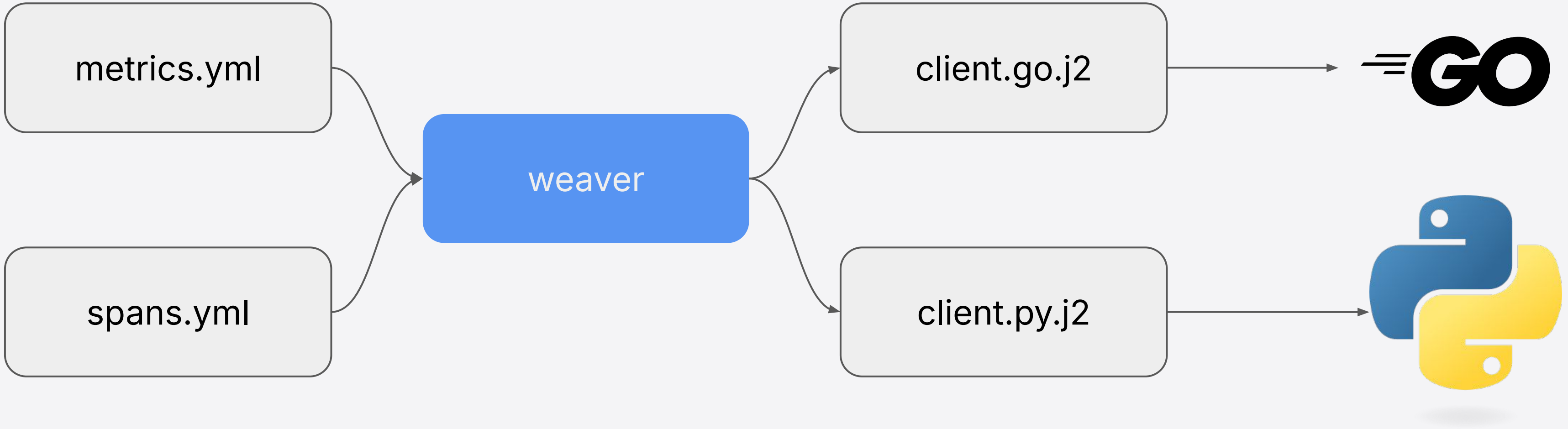
Weaver

- Semantic convention tooling
- Code generation
- Policy validation
- Live checking
- Written in rust 🦀

Weaver Architecture


Input artifacts
OpenTelemetry groups responsibility
OTel SemConv Registry and Telemetry Schema
Custom SemConv Registries and App Telemetry Schema
App. developpers responsibility
SemConv Policies
Markdown and Code Templates
OTel Weaver Platform (CLI)
Policy Engine
Resolution Process
Template Engine
Community Plugins
generate
Resolved Telemetry Schema
consume
Analyzed by IDE for smart auto-completion, integrated API docs, ...
Generated Type-Safe OTel Client SDK/API
Markdown Documentations
Telemetry Data Catalog, Compliance Report, Generated Dashboard, Data Policies Config. ...
Output artifacts

Weaver Architecture



Demo Time

Challenges

- Cross-referencing attributes
  - Seen in demo
- Keeping documentation up to date
  - Weaver can generate the docs for you!
- Ownership process
  - Schema lives in a shared git repository with well-defined code owners
- Ensuring teams adopt schema
  - Promote client usage & live checking

Recap

- Schemas help you combat sprawl
- Weaver can build clients for your teams
- OTel Semantic conventions are a good starting point
- Technical solutions can’t solve social problems (on their own)

Grafana Labs

Thanks for participating!

@dominik.suess.wtf

@thesuess@kulupu.party

hello@dominik.suess.wtf

Get involved:

#opentelemetry

open-telemetry/weaver

community.grafana.com