



# SLO libsonnet

Matthias Loibl [[@metalmatze](#)]  
Software Engineer, Red Hat

# SLOs

Service Level Objectives

# SLIs

Service Level Indicators

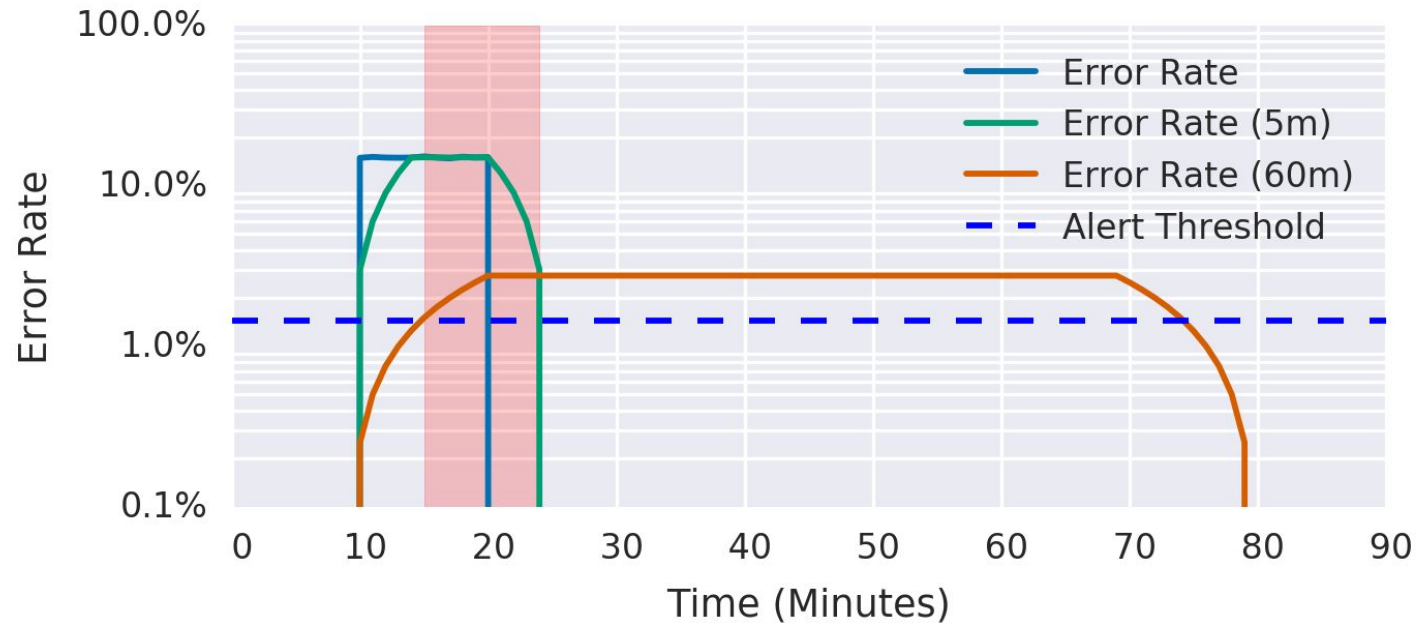
# SLIs

```
rate(http_requests_total{code=~"5.."}[5m])
```

# SLOs

```
rate(http_requests_total{code=~"5.."}[5m]) > 0.01
```

# Multi-Burn-Rate Alerts



```
alert: ErrorBudgetBurn
expr:
  (
    status_class_5xx:http_requests_total:ratio_rate1h{job="prometheus"} > (14.4*0.001000)
    and
    status_class_5xx:http_requests_total:ratio_rate5m{job="prometheus"} > (14.4*0.001000)
  )
or
  (
    status_class_5xx:http_requests_total:ratio_rate6h{job="prometheus"} > (6*0.001000)
    and
    status_class_5xx:http_requests_total:ratio_rate30m{job="prometheus"} > (6*0.001000)
  )
```

```
record: status_class_5xx:http_requests_total:ratio_rate5m
```

```
expr:
```

```
  sum(status_class:http_requests_total:rate5m{job="prometheus",status_class="5xx"})  
  /  
  sum(status_class:http_requests_total:rate5m{job="prometheus"})
```



```
record: status_class:http_requests_total:rate5m
expr:
  sum by (status_class) (
    label_replace(
      rate(http_requests_total{job="prometheus"}[5m]
    ), "status_class", "${1}xx", "code", "([0-9])..")
  )
```

Jsonnet

# SLO-libsonnet

[github.com/metalmatze/slo-libsonnet](https://github.com/metalmatze/slo-libsonnet)

```
local slo = import '../slo-libsonnet/slo.libsonnet';

{
  local errorburnrate = slo.errorburn({
    metric: http_requests_total,
    selectors: ['job="prometheus"'],

    errorBudget: 1-0.99,
  }),

  rules:
    errorburnrate.recordingrules +
    errorburnrate.alerts,
}
```

Demo

HTML 43.4%

Dart (no JS) 23.0%

Go 21.3%

Jsonnet 7.1%

Demo

Ideas



# Feedback!

...and help :D

# Thanks

to Björn Rabenstein & Aditya Konarde



[promtools.matthiasloibl.com](https://promtools.matthiasloibl.com)

[github.com/metalmatze/slo-libsonnet](https://github.com/metalmatze/slo-libsonnet)

[github.com/metalmatze/slo-libsonnet-web](https://github.com/metalmatze/slo-libsonnet-web)