Site Reliability Engineering with Prometheus

Eddy Reyes • eddy@mindsight.io

SRE vs Devops

Devops

- Set of Practices and Principles for Better Cloud Applications
- Broad, Transformational, Managerial

SRE

- Narrower Focus on Reliability
- A Way to Implement Devops, with Specific Recommendations

Reliability At Scale

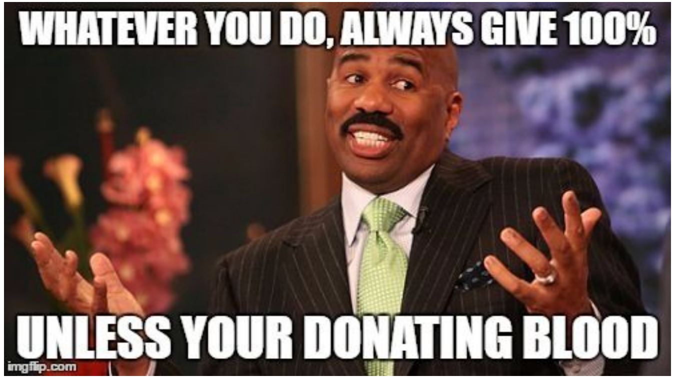
Metrics

- Rates
- Errors
- Distributions / Durations

Process

- Service Level Objectives
- Blameless Postmortems

100% … Good Luck!



Not Really Attainable

- Your dependencies are <100%
- Never change anything again
- Prohibitive cost for little additional value

Service Level Objectives (SLOs)

- Percentage (<100%)
- Over Period of Time
  - e.g. "30 days"

Based on Acceptable Quality to Your **Customers**

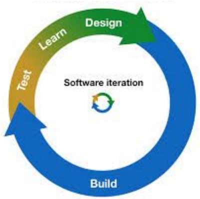
SLAs vs. SLOs

SLA

- Contractual obligation of availability or refund

SLO

- Internal metric, based on observation
- Can change / be iterated upon



SLO Benefits

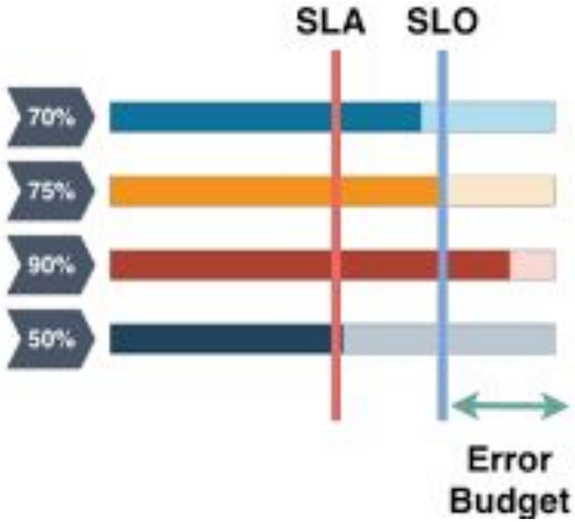
Org Interop

- Business vs. Engineering Criteria for Success
- Customer-focused and business-driven

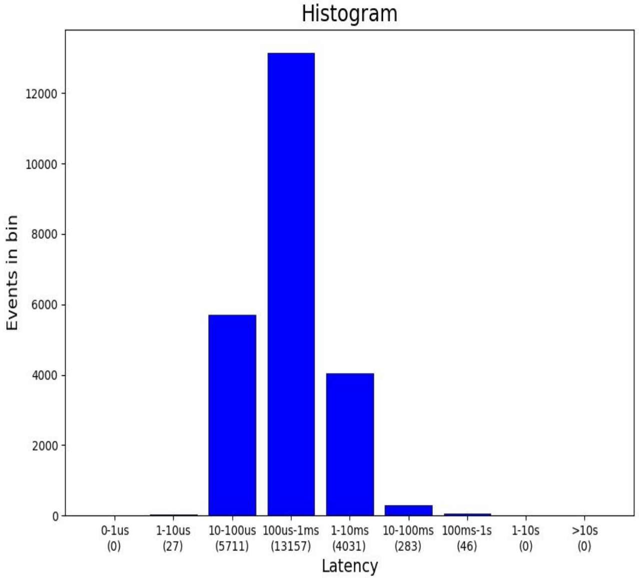
Engineering Enabler

- <100% -> Acceptable Level of Failure
  - Learning / Iteration
- SLO Failure Is Now the Only Alerting That Matters
  - Alert Fatigue

Error Budgets



Latency Error Budgets



How To Calculate

- Choose Maximum Latency X
- Error budget:
  - #Samples * 1-SLO < #Samples(>X)

Error Budget Benefits

Process Safety

- Is it safe to deploy right now?
- Does an alert need to wake me up at 3am?

Experimentation

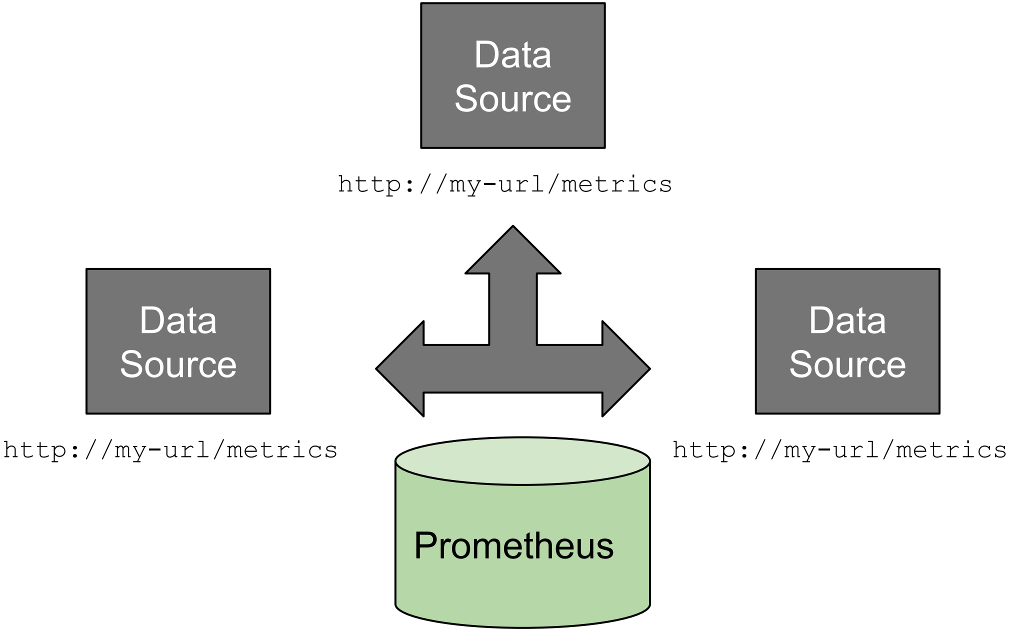
- Chaos Engineering
- Example: Facebook
  - Intentional crashing for UX research

Where Does the Data Come From?

Service Level Indicators (SLIs)

- Telemetry Used To Compute SLOs
- Error Rates:
  - Success / total (over SLO period)
- Latencies
  - 95% of all latencies below X

Prometheus Pulls Metrics



Data Source

http://my-url/metrics

Data Source

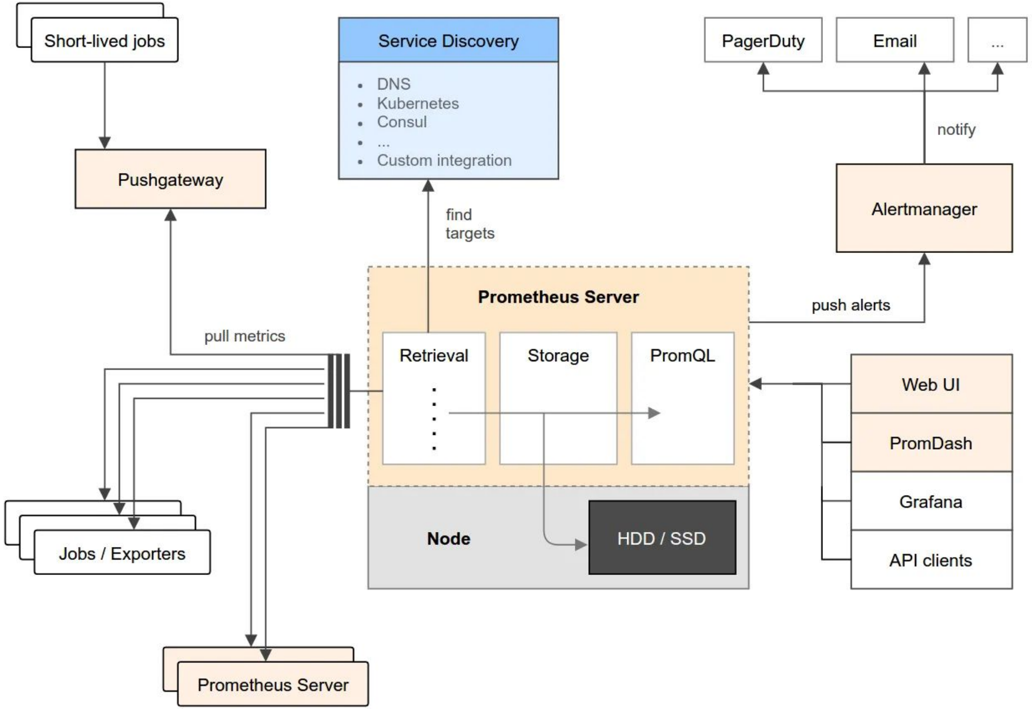
http://my-url/metrics

Data Source

http://my-url/metrics

Prometheus

Prometheus


Short-lived jobs
Service Discovery
DNS
Kubernetes
Consul
...
Custom integration
PagerDuty
Email
...
notify
Pushgateway
find targets
Alertmanager
Prometheus Server
push alerts
pull metrics
Retrieval
Storage
PromQL
Web UI
PromDash
Grafana
API clients
Jobs / Exporters
Node
HDD / SSD
Prometheus Server

Prometheus Data Model

Queries - PromQL

```
series_name{tag1=”A”}
```

Result (Instant Vector):

```
series_name{tag1=”A” job=”mytest”} 10.81
series_name{tag1=”A” job=”othertest”} 100.17
```

Prometheus Data Model

Data Types

- Counters
  - Monotonically increasing
  - e.g. request counters
- Gauges
  - Instantaneous reading
  - e.g. CPU usage %
- Histograms / Samples
  - Values kept as distributions
  - e.g. Request latency

SLIs With Prometheus

Error Rate

```
sum(rate(slo_errors[10m])) by (job) / sum(rate(slo_requests[10m]))
by (job)
```

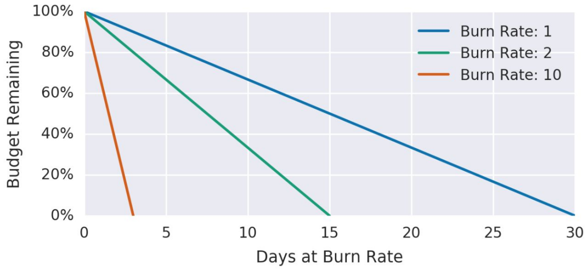
Latency Quantile

```
histogram_quantile(0.9,
sum(rate(http_request_duration_seconds_bucket[10m])) by (job, le))
```

SRE-Friendly Alerts

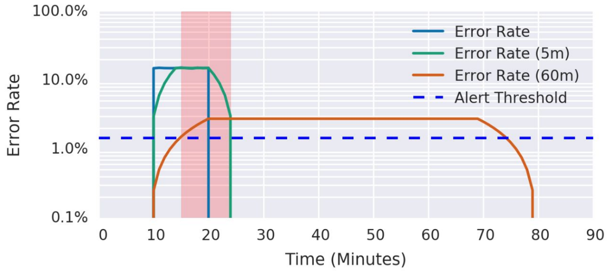
- Given Error Budget, Compute the Burn Rates
- Am I About to Fail SLO In:
    - 1 hour? WAKE UP!!
    - 1 day? WAKE UP!!
    - 1 week? Look on Monday...

Alert On Burn Rate



Burn rate	Error rate for a 99.9% SLO	Time to exhaustion
1	0.1%	30 days
2	0.2%	15 days
10	1%	3 days
1,000	100%	43 minutes

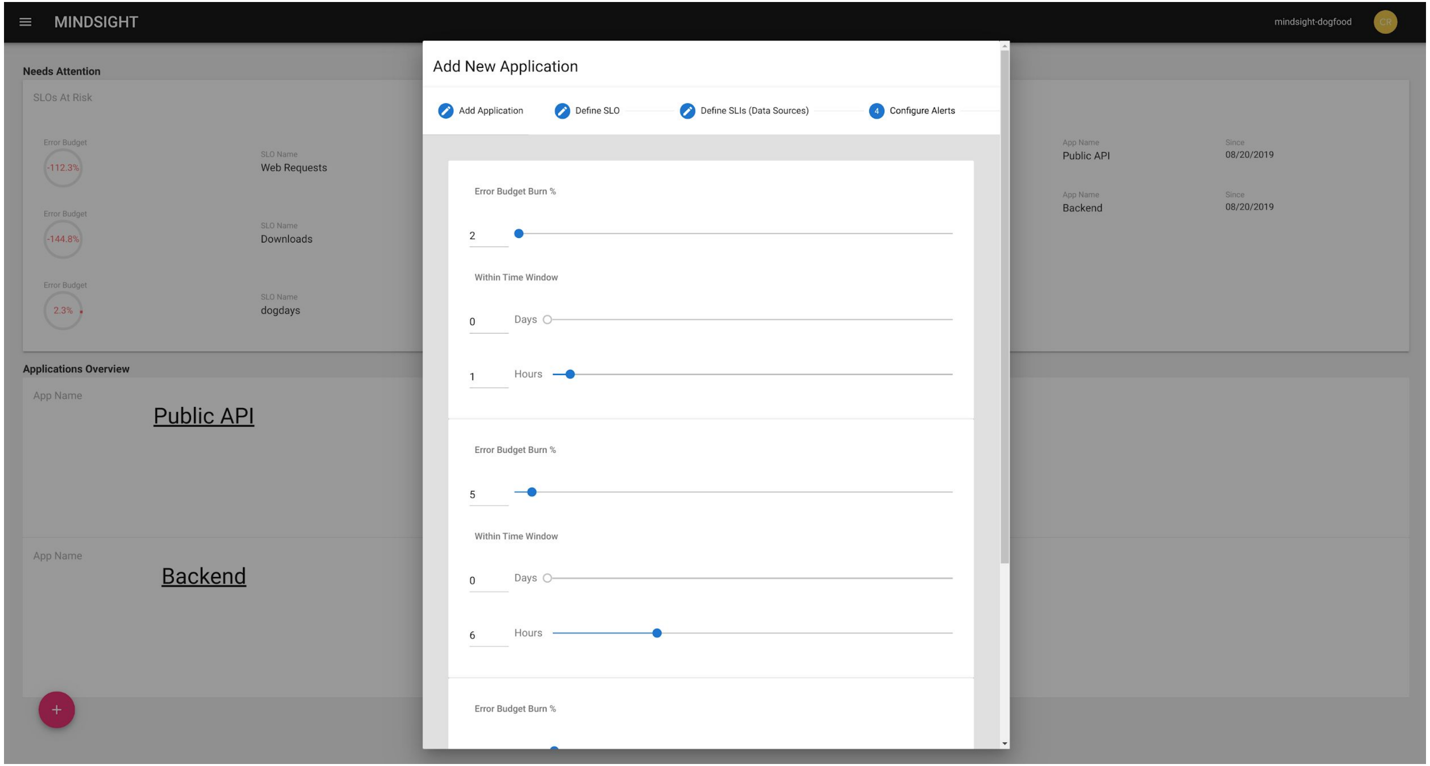
Alert Sensitivity



```
expr: (
        job:slo_errors_per_request:ratio_rate1h{job="myjob"} > (14.4*0.001)
      and
        job:slo_errors_per_request:ratio_rate5m{job="myjob"} > (14.4*0.001)
      )
    or
      (
        job:slo_errors_per_request:ratio_rate6h{job="myjob"} > (6*0.001)
      and
        job:slo_errors_per_request:ratio_rate30m{job="myjob"} > (6*0.001)
      )
severity: page

expr: (
        job:slo_errors_per_request:ratio_rate24h{job="myjob"} > (3*0.001)
      and
        job:slo_errors_per_request:ratio_rate2h{job="myjob"} > (3*0.001)
      )
    or
      (
        job:slo_errors_per_request:ratio_rate3d{job="myjob"} > 0.001
      and
        job:slo_errors_per_request:ratio_rate6h{job="myjob"} > 0.001
      )
severity: ticket
```

Mindsight Simplifies SRE



MINDSIGHT

mindsight-dogfood

CR

Needs Attention

SLOs At Risk

Error Budget	SLO Name
-112.3%	Web Requests
-144.8%	Downloads
2.3%	dogdays

App Name	Since
Public API	08/20/2019
Backend	08/20/2019

Applications Overview

App Name
Public API

App Name
Backend

Add New Application

1. Add Application
2. Define SLO
3. Define SLIs (Data Sources)
4. Configure Alerts

Error Budget Burn %
2

Within Time Window
0 Days
1 Hours

Error Budget Burn %
5

Within Time Window
0 Days
6 Hours

Error Budget Burn %

Instrumenting Infrastructure Into Prometheus

- Many tools help here
- Promising telemetry technology:
    - Kubernetes + Istio
    - Available on Google Cloud

Thank You!!

Eddy Reyes

eddy@mindsight.io

@0x000edd1e