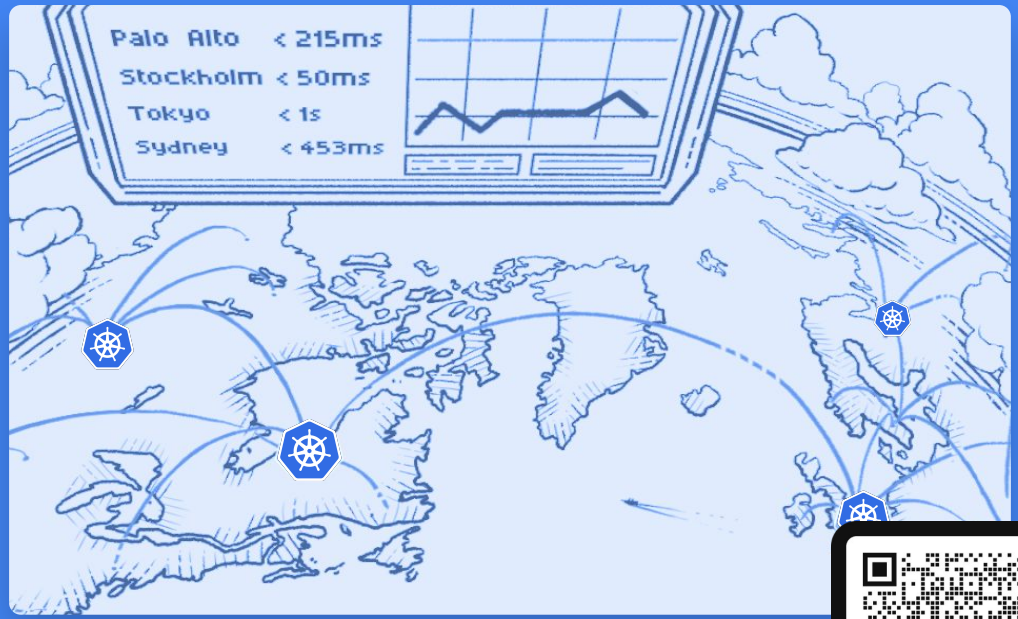
Distributed Load Testing

in Kubernetes



Paul Balogh
Developer Advocate, Grafana Labs
@javaducky

Overview

1. **What is load testing?**
2. Why distribute testing with k8s?
3. Introducing the k6-operator
4. Where do we go from here?

“”

Load testing is the process of putting **demand** on a system and **measuring** its **response**.

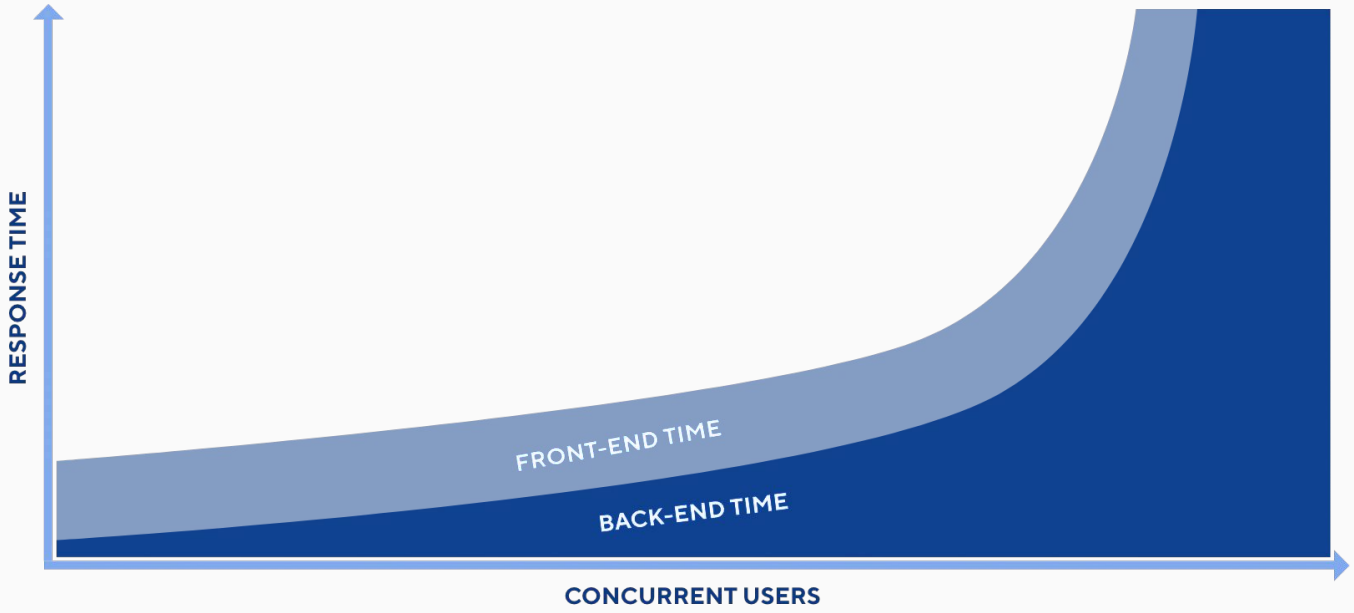
The Internet
Knower of all things

Load testing **myths**

- Performance testing **==** load testing
- It is only for **large** companies
- Is **expensive** to do
- Should **only** be in production
- Unnecessary if you have o11y

Why do load testing today?

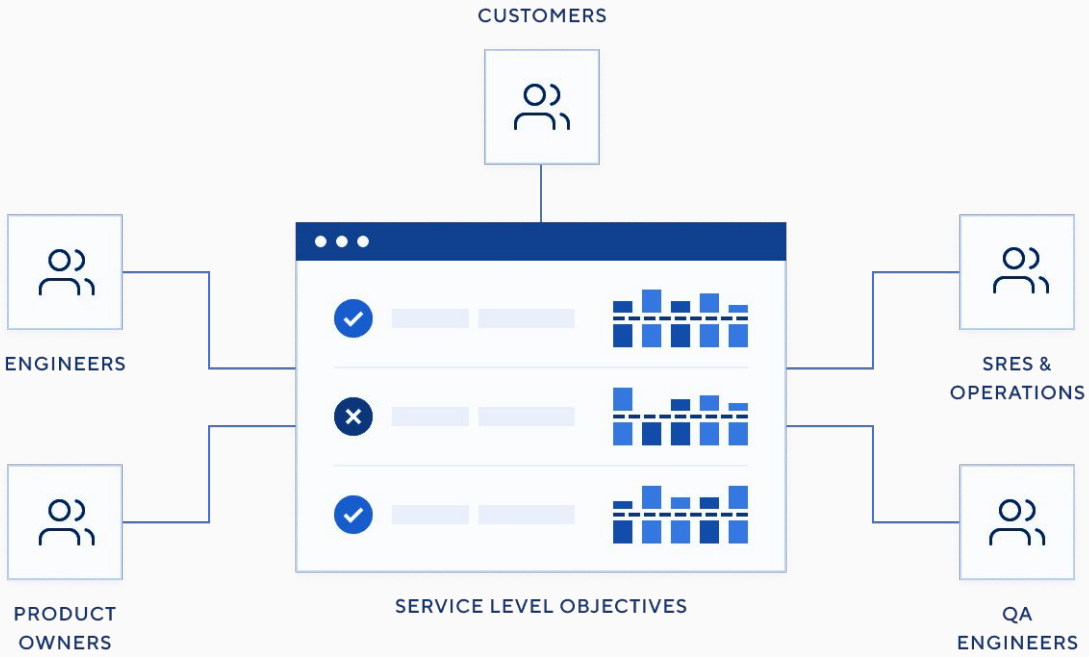
UX



Why do load testing today?

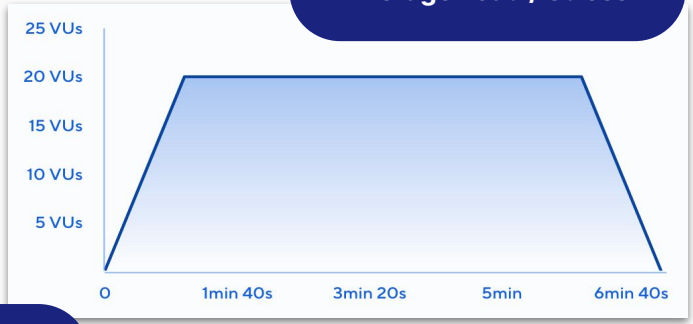
Proactively test

SLOs

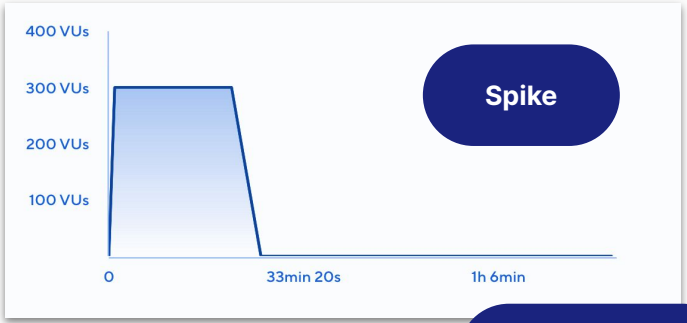


Common **types** of load tests

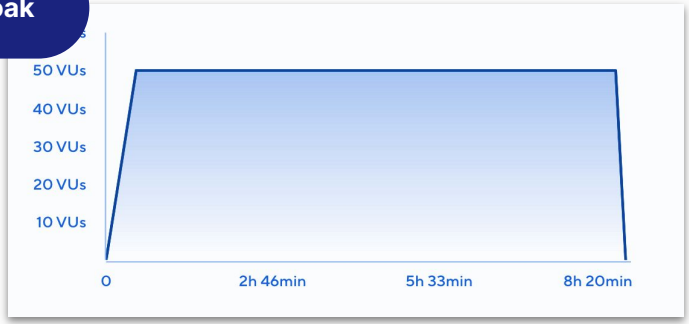
Average Load / Stress



Spike



Soak



Breakpoint

Overview

1. What is load testing?
2. **Why distribute testing with k8s?**
3. Introducing the k6-operator
4. Where do we go from here?

Why distribute testing with k8s?

Kubernetes is already the **preferred** operating environment

- Consistent with other applications
- Pre-established infrastructure for Observability (o11y)

Cannot use an **external cloud** service, like *Grafana Cloud k6**

** this is changing with Private Load Zones (PLZv2)*

Need extremely **large demand** for test load

- Optimized node can simulate 40,000 users, but that's still not enough?

Load should come from **multiple IPs**

Overview

1. What is load testing?
2. Why distribute testing with k8s?
3. **Introducing the k6-operator**
4. Where do we go from here?

Introducing **k6** and **k6-operator**

k6, a *reliability testing* tool

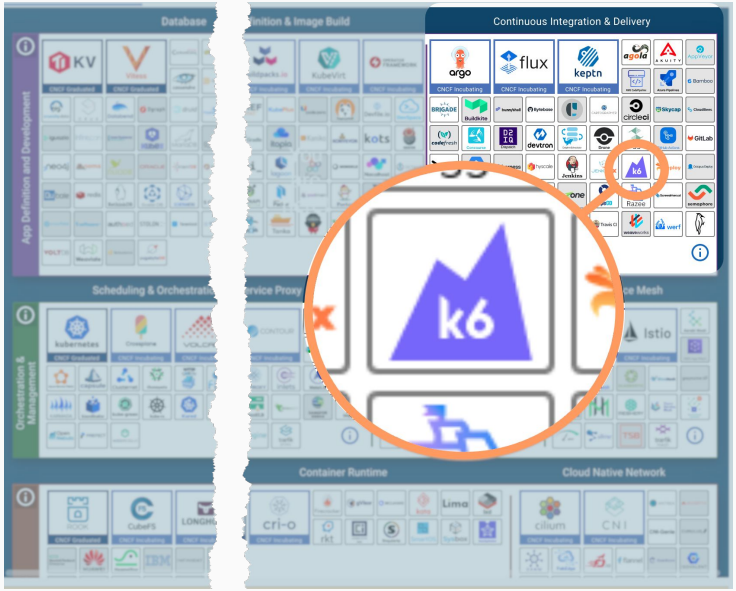
- Formerly known as *Load Impact*
- Open Source since 2016
- ~21.3k Github Stars *(as of October 2023)*
- Promotes “shift-left” testing
- Acquired by Grafana Labs in 2021

github.com/grafana/k6

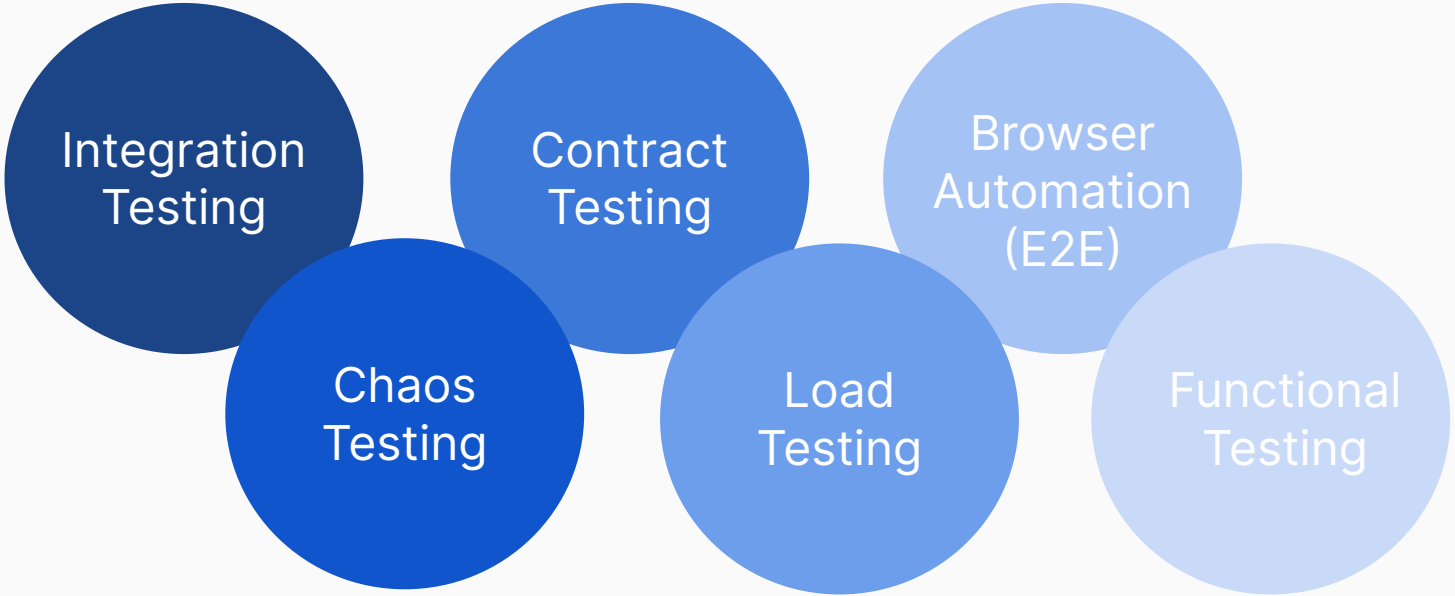
k6-operator for *k8s* distributed execution

- August 2020 operator is born
- Now becoming basis for *Grafana Cloud k6*

github.com/grafana/k6-operator



Determine your strategy



Model your load with JavaScript

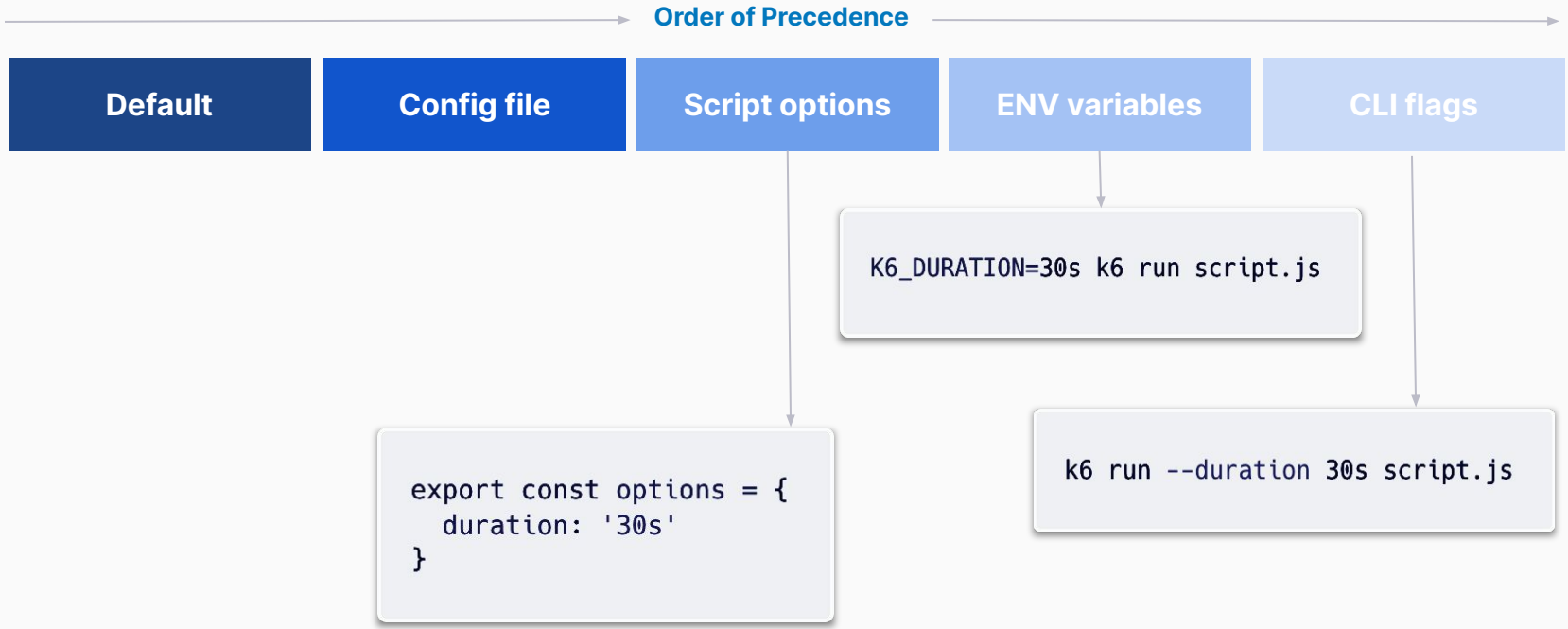
"Shape" activity with Executors

- Shared iterations
- Per VU iterations
- Constant VUs
- Ramping VUs
- Constant Arrival Rate
- Ramping Arrival Rate

```
export const options = {
  scenarios: {
    mainScenario: {
      executor: 'constant-vus',
      exec: 'myUserFlowA',
      vus: 10,
      duration: '60m',
    },
    scenarioB: {
      executor: 'shared-iterations',
      exec: 'myUserFlowB',
      startTime: '30m',
    },
  },
};
```

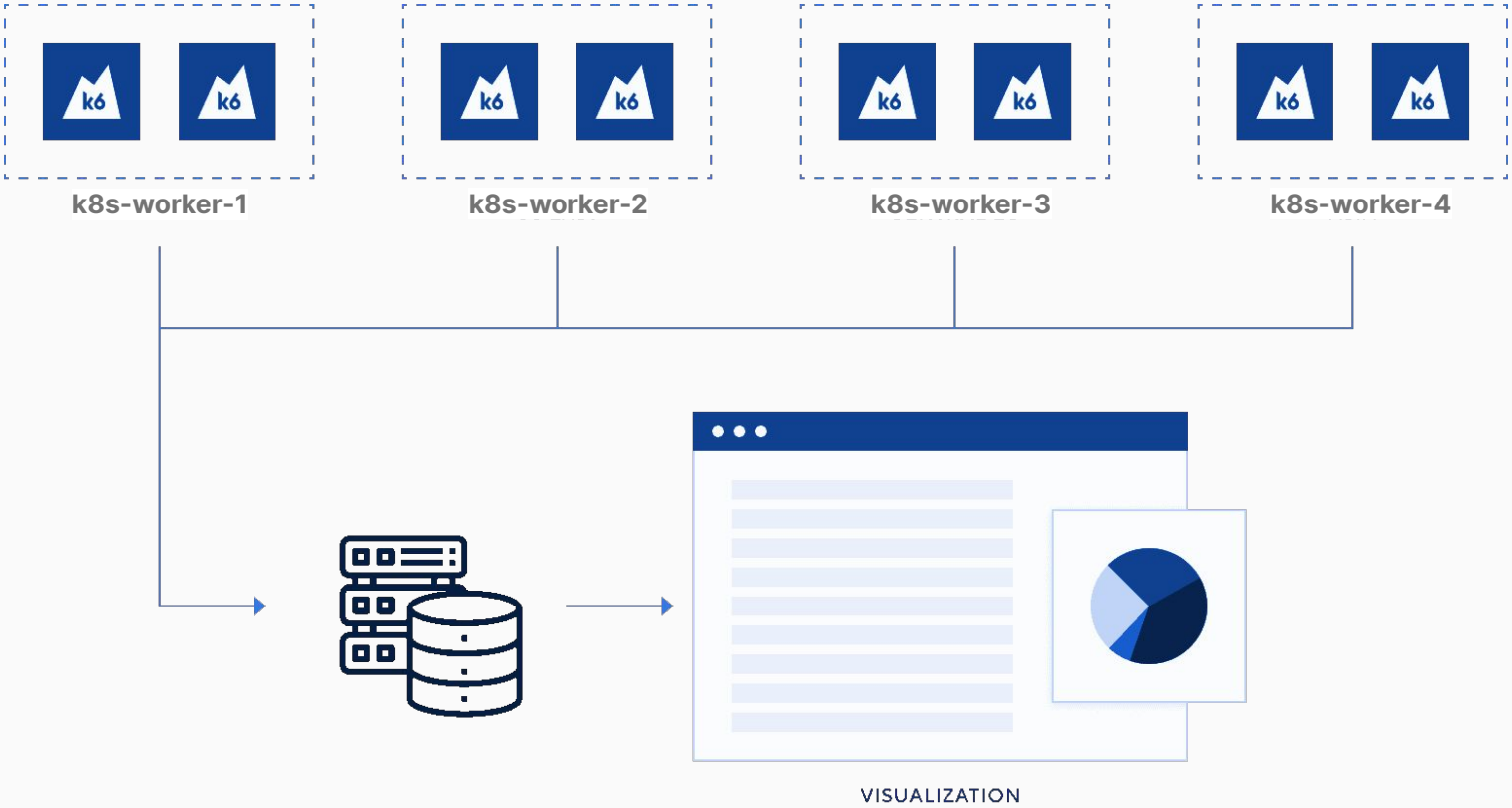
https://k6.io/docs/using-k6/scenarios/ for more details.

Configure execution options

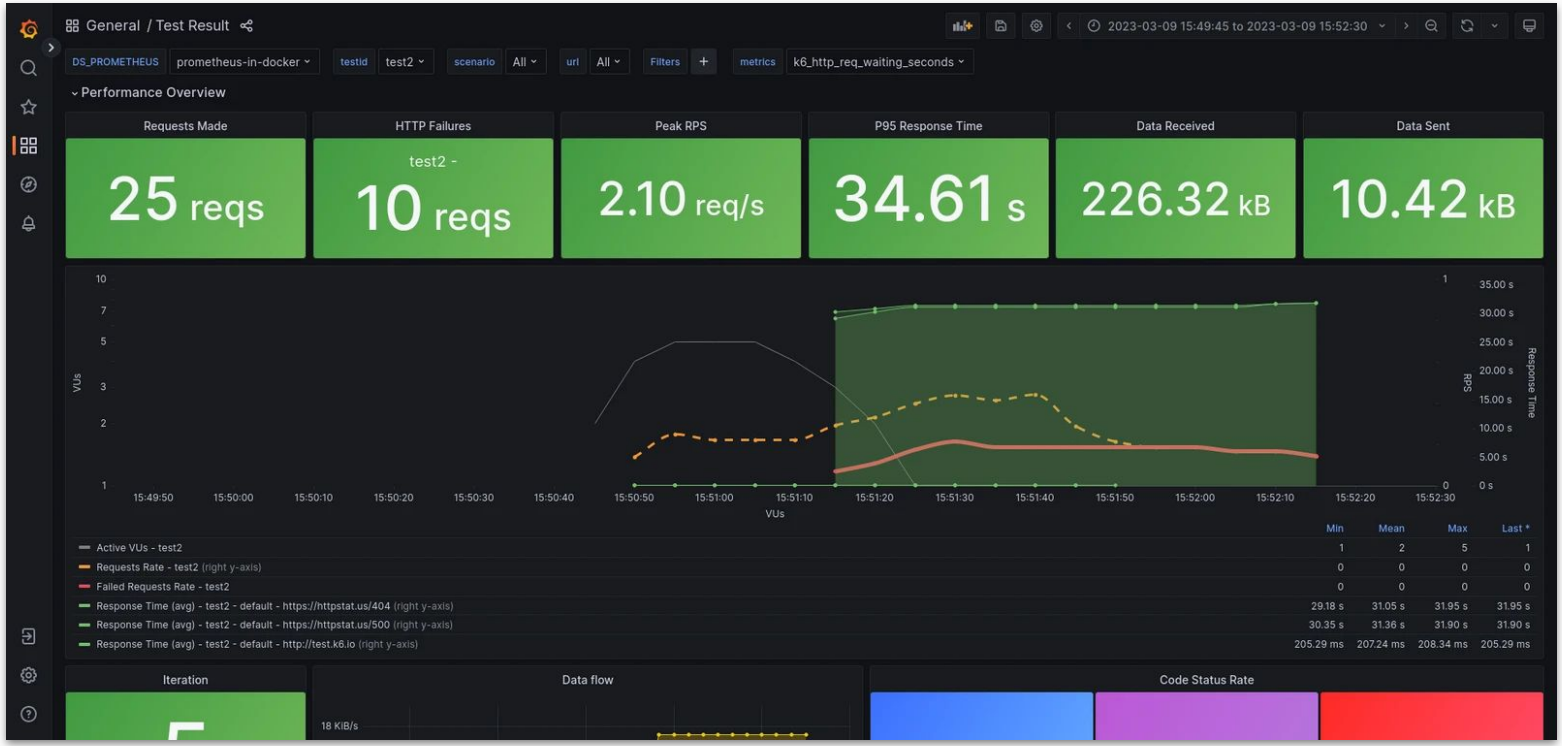


https://k6.io/docs/using-k6/k6-options/reference/ for more options.

Distribute test execution



View aggregated results in real-time



https://k6.io/docs/results-output/real-time/ for more output options.

Demo!

Overview

1. What is load testing?
2. Why distribute testing with k8s?
3. Introducing the k6-operator
4. **Where do we go from here?**

“Golden Signals” of Observability

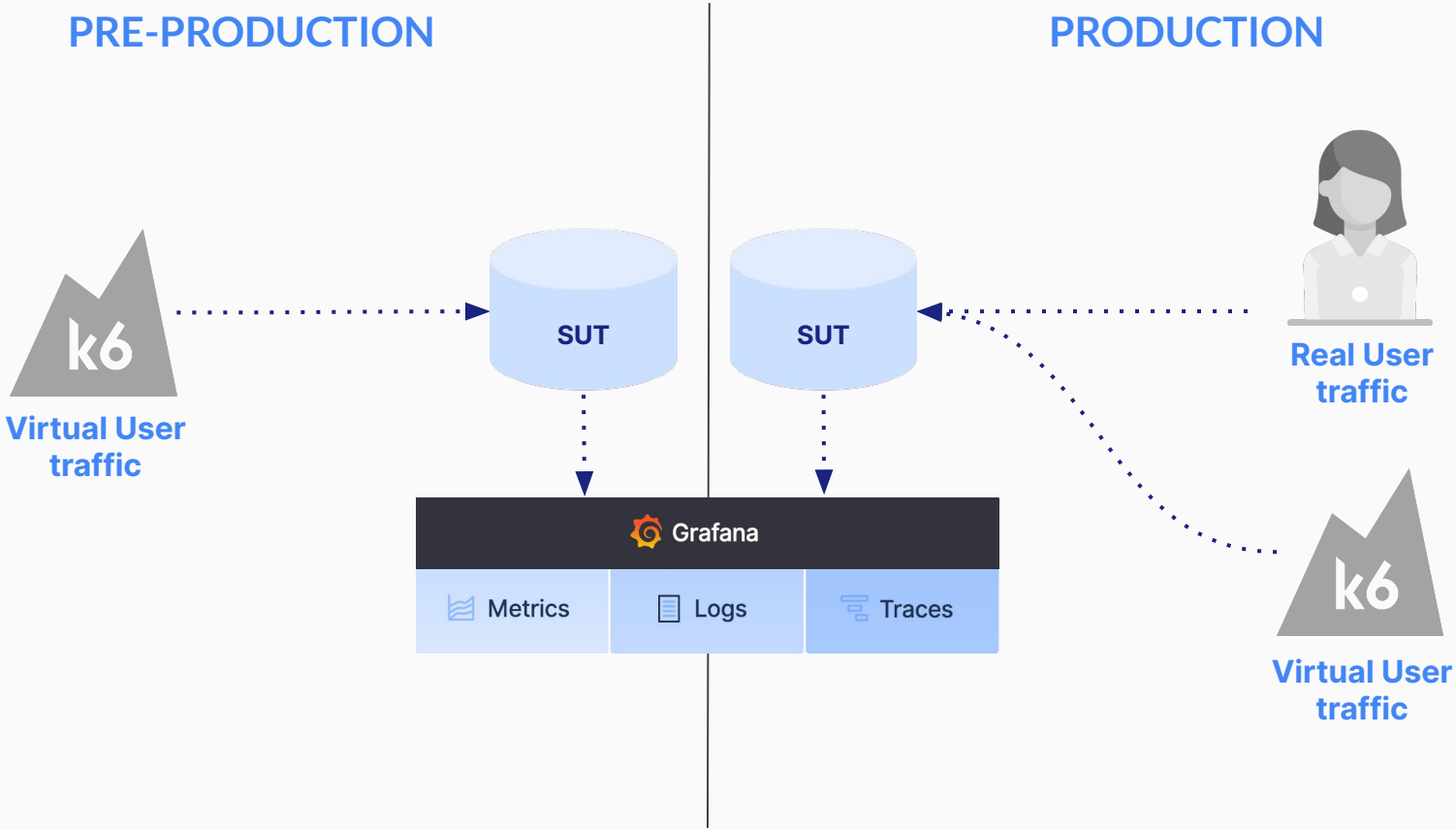
Network quality, Geolocation

Special events, Cyber attacks

Bugs, 500s, Security/Auth

Underprovisioning, Infra issues

Proactively improve reliability



Thanks for participating!

Connect with Paul as
@javaducky or **linkedin/in/pabalogh**