

13th Symposium on Software Performance 2022

Cloud-Native Scalability Benchmarking with Theodolite

Sören Henning, Benedikt Wetzel, Wilhelm Hasselbring



Kiel University
Christian-Albrechts-Universität zu Kiel

Scalability Benchmarking of Cloud-Native Applications

- Scalability is a main driver for adopting cloud-native applications and microservices

(Knoche & Hasselbring 2019; Soldani et al. 2018; Kratzke & Quint 2017)

Scalability Benchmarking of Cloud-Native Applications

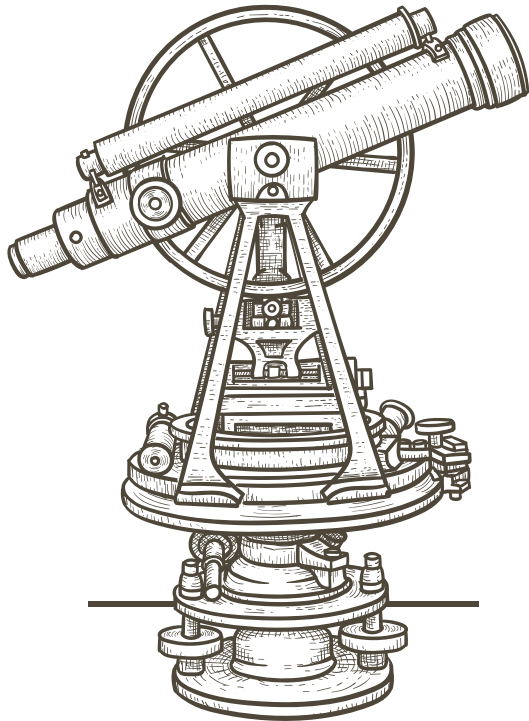
- Scalability is a main driver for adopting cloud-native applications and microservices
(Knoche & Hasselbring 2019; Soldani et al. 2018; Kratzke & Quint 2017)
- Engineers and researchers use benchmarking to evaluate and compare quality of frameworks, configuration, etc.
(Kounev et al. 2020; Hasselbring 2021)

Scalability Benchmarking of Cloud-Native Applications

- Scalability is a main driver for adopting cloud-native applications and microservices
(Knoche & Hasselbring 2019; Soldani et al. 2018; Kratzke & Quint 2017)
- Engineers and researchers use benchmarking to evaluate and compare quality of frameworks, configuration, etc.
(Kounev et al. 2020; Hasselbring 2021)
- However, no commonly used method for benchmarking scalability of cloud-native applications
(Henning & Hasselbring 2022)

Goal:
Make Scalability Benchmarking
more Usable and Reproducible!

The Theodolite Scalability Benchmarking Framework



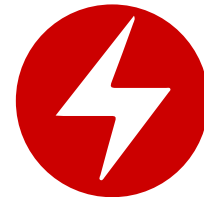
Theodolite



Creating benchmarks based on existing **Kubernetes** resource files



Defining SLO based on existing **Prometheus** metrics via **PromQL** queries



Kubernetes Operator for declarative definition of benchmarks and executions

In Search of Microservice Show Case...



Categories

Black Tea

Pure black tea and blends

Green Tea

From China and Japan

Herbal Tea

Helps when you feel sick

Rooibos

In many variations

White Tea

If green tea doesn't agree with you

Black Tea

Pure black tea and blends

Green Tea

From China and Japan

Herbal Tea

Helps when you feel sick

Rooibos

In many variations

White Tea

If green tea doesn't agree with you

Black Tea

Pure black tea and blends

Green Tea

From China and Japan

Herbal Tea

Helps when you feel sick

Rooibos

In many variations

White Tea

If green tea doesn't agree with you

Green Tea



Sencha (loose)

Price: 66,70 \$

Great Green Tea: Sencha (loose)

Add to Cart



Earl Grey Green (15 bags)

Price: 48,39 \$

Great Green Tea: Earl Grey Gree...

Add to Cart



Matcha 100 g

Price: 118,59 \$

Great Green Tea: Matcha 100 g

Add to Cart



Sencha (loose), v1

Price: 76,58 \$

Great Green Tea: Sencha (loose)

Add to Cart



Earl Grey Green (15 bags), v1

Price: 74,83 \$

Great Green Tea: Earl Grey Gree...

Add to Cart



Sencha (15 bags)

Price: 3,48 \$

Great Green Tea: Sencha (15 bags)

Add to Cart



Earl Grey Green (25 bags)

Price: 82,44 \$

Great Green Tea: Earl Grey Gree...

Add to Cart



Gunpowder Tea (loose)

Price: 32,74 \$

Great Green Tea: Gunpowder Tea...

Add to Cart



Sencha (15 bags), v1

Price: 81,23 \$

Great Green Tea: Sencha (15 bags)

Add to Cart



Earl Grey Green (25 bags), v1

Price: 69,06 \$

Great Green Tea: Earl Grey Gree...

Add to Cart



Sencha (25 bags)

Price: 109,35 \$

Great Green Tea: Sencha (25 bags)

Add to Cart



Matcha 30 g

Price: 10,18 \$

Great Green Tea: Matcha 30 g

Add to Cart



Gunpowder Tea (15 bags)

Price: 83,61 \$

Great Green Tea: Gunpowder Tea...

Add to Cart



Sencha (25 bags), v1

Price: 11,93 \$

Great Green Tea: Sencha (25 bags)

Add to Cart



Matcha 30 g, v1

Price: 85,38 \$

Great Green Tea: Matcha 30 g

Add to Cart



Earl Grey Green (loose)

Price: 80,63 \$

Great Green Tea: Earl Grey Gree...

Add to Cart



Matcha 50 g

Price: 5,65 \$

Great Green Tea: Matcha 50 g

Add to Cart



Gunpowder Tea (25 bags)

Price: 71,50 \$

Great Green Tea: Gunpowder Tea...

Add to Cart



Earl Grey Green (loose), v1

Price: 86,99 \$

Great Green Tea: Earl Grey Gree...

Add to Cart



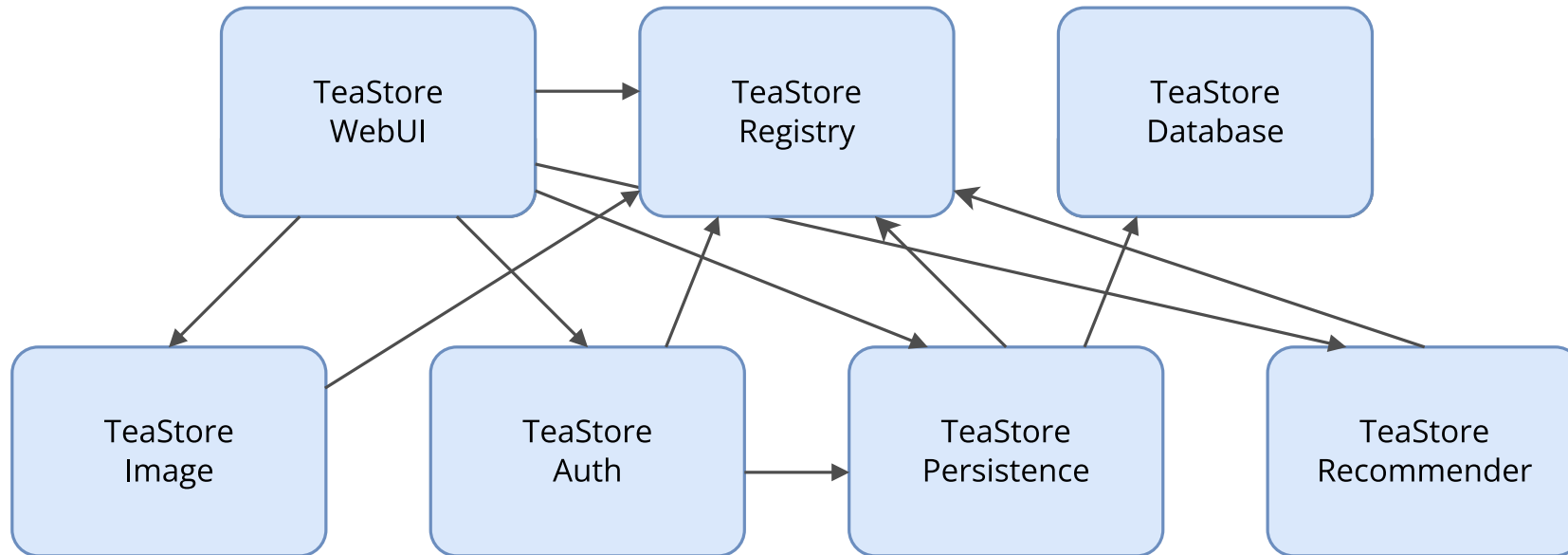
Matcha 50 g, v1

Price: 13,35 \$

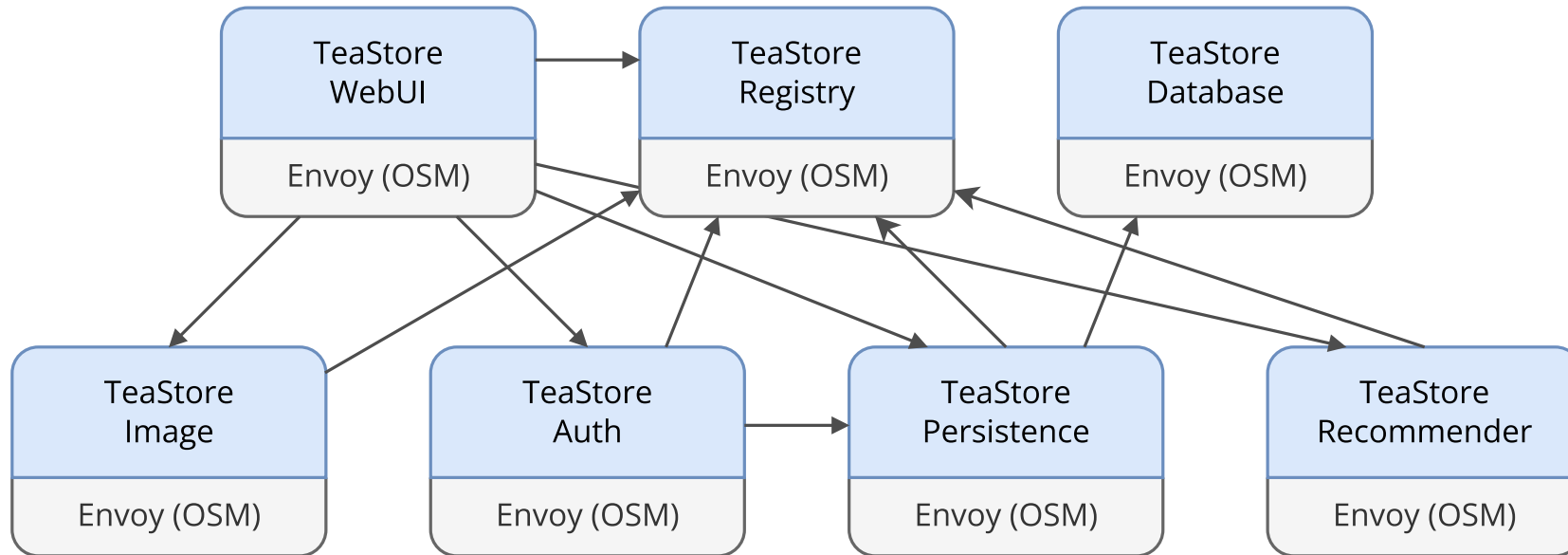
Great Green Tea: Matcha 50 g

Add to Cart

TeaStore Architecture



TeaStore Architecture

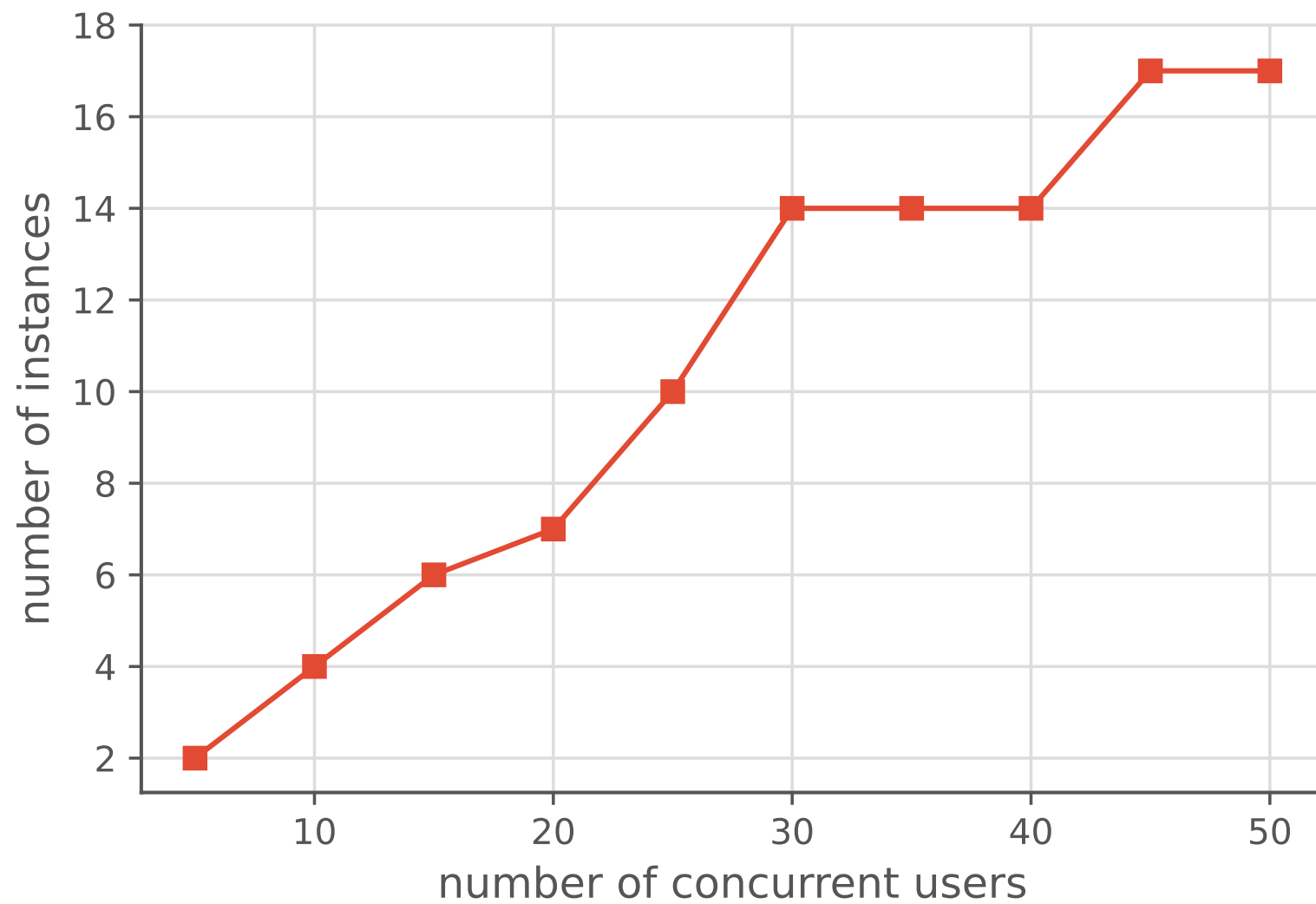


kubernetes



Open Service Mesh

Desired Outcome



Designing a Scalability Benchmark for the TeaStore

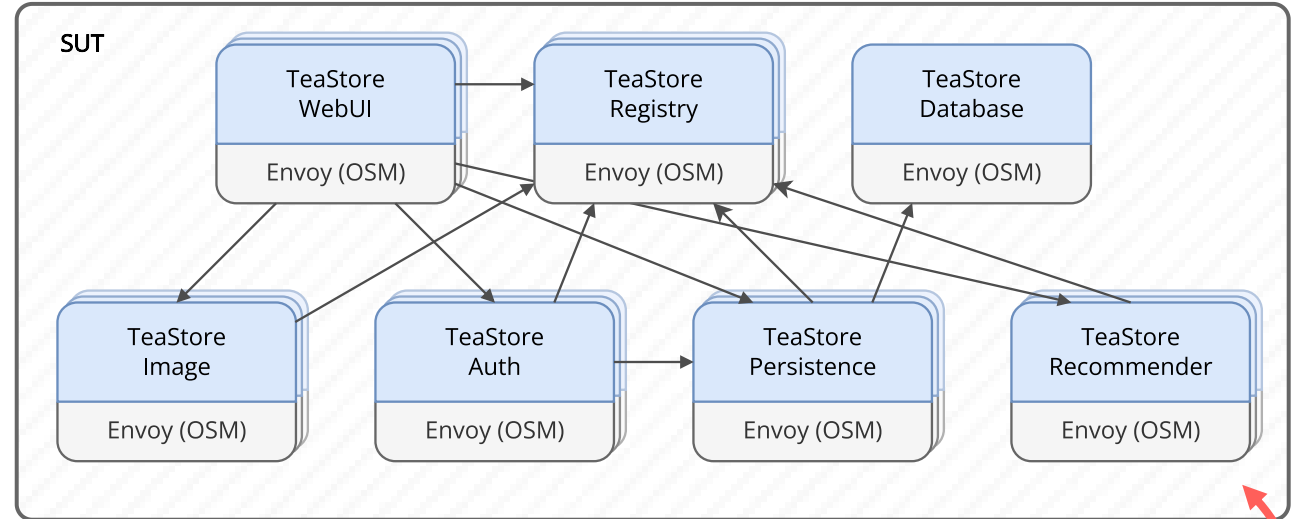
```
TeaStore - benchmark.yaml
1  apiVersion: theodolite.rocks/v1beta1
2  kind: benchmark
3  metadata:
4    name: teastore
5  spec:
6    waitForResourcesEnabled: true
7    sut:
8      resources:
9      - configMap:
10        name: teastore-deployment
11        files:
12        - teastore-auth-deployment.yaml
13        - teastore-auth-service.yaml
14        - teastore-db-deployment.yaml
15        - teastore-db-service.yaml
16        - teastore-image-deployment.yaml
17        - teastore-image-service.yaml
18        - teastore-persistence-deployment.yaml
19        - teastore-persistence-service.yaml
20        - teastore-recommender-deployment.yaml
21        - teastore-recommender-service.yaml
22        - teastore-registry-deployment.yaml
23        - teastore-registry-service.yaml
24        - teastore-webui-deployment.yaml
25        - teastore-webui-nodeport.yaml
26  loadGenerator:
27    resources:
28    - configMap:
29      name: teastore-imeter-deployment
```

```

TeaStore - benchmark.yaml
1  apiVersion: theodolite.rocks/v1beta1
2  kind: benchmark
3  metadata:
4    name: teastore
5  spec:
6    waitForResourcesEnabled: true
7  sut:
8    resources:
9      - configMap:
10         name: teastore-deployment
11         files:
12           - teastore-auth-deployment.yaml
13           - teastore-auth-service.yaml
14           - teastore-db-deployment.yaml
15           - teastore-db-service.yaml
16           - teastore-image-deployment.yaml
17           - teastore-image-service.yaml
18           - teastore-persistence-deployment.yaml
19           - teastore-persistence-service.yaml
20           - teastore-recommender-deployment.yaml
21           - teastore-recommender-service.yaml
22           - teastore-registry-deployment.yaml
23           - teastore-registry-service.yaml
24           - teastore-webui-deployment.yaml
25           - teastore-webui-nodeport.yaml
26  loadGenerator:
27    resources:
28      - configMap:
29         name: teastore-jmeter-deployment
30         files:
31           - jmeter.yaml
32  resourceTypes:
33    - typeName: Instances
34      patchers:
35        - type: ReplicaPatcher
36          resource: teastore-auth-deployment.yaml
37        - type: ReplicaPatcher
38          resource: teastore-image-deployment.yaml
39        - type: ReplicaPatcher
40          resource: teastore-persistence-deployment.yaml

```

System under Test (SUT)



TeaStore - recommender-deployment.yaml

```

1  apiVersion: apps/v1
2  kind: Deployment
3  metadata:
4    name: teastore-recommender
5  spec:
6    replicas: 15
7    selector:
8      matchLabels:
9        app: teastore
10       run: teastore-recommender
11  template:
12    metadata:
13      labels:

```

Load Generator



```
TeaStore - benchmark.yaml
1  apiVersion: theodolite.rocks/v1beta1
2  kind: benchmark
3  metadata:
4    name: teastore
5  spec:
6    waitForResourcesEnabled: true
7    sut:
8      resources:
9      - configMap:
10        name: teastore-deployment
11        files:
12        - teastore-auth-deployment.yaml
13        - teastore-auth-service.yaml
14        - teastore-db-deployment.yaml
15        - teastore-db-service.yaml
16        - teastore-image-deployment.yaml
17        - teastore-image-service.yaml
18        - teastore-persistence-deployment.yaml
19        - teastore-persistence-service.yaml
20        - teastore-recommender-deployment.yaml
21        - teastore-recommender-service.yaml
22        - teastore-registry-deployment.yaml
23        - teastore-registry-service.yaml
24        - teastore-webui-deployment.yaml
25        - teastore-webui-service.yaml
26
27    loadGenerator:
28      resources:
29      - configMap:
30        name: teastore-jmeter-deployment
31        files:
32        - jmeter.yaml
33
34    resourceTypes:
35    - type: Instances
36      patchers:
37      - type: ReplicaPatcher
38        resource: teastore-auth-deployment.yaml
39      - type: ReplicaPatcher
40        resource: teastore-image-deployment.yaml
41      - type: ReplicaPatcher
42        resource: teastore-persistence-deployment.yaml
```

```
TeaStore - jmeter.yaml
1  apiVersion: apps/v1
2  kind: Deployment
3  metadata:
4    name: jmeter
5    annotations:
6      openservicemesh.io/sidecar-injection: enabled
7  spec:
8    replicas: 1
9    selector:
10      matchLabels:
11        app: jmeter
12    template:
13      metadata:
14        labels:
15          app: jmeter
16      spec:
17        containers:
18        - name: jmeter
19          image: justb4/jmeter:5.4
20          command: [ "/bin/bash", "-c", "--" ]
21          args:
22            - >
23              java -jar bin/ApacheJMeter.jar \
24                -t profile/teastore_browse_nogui.jmx \
25                -Jhostname teastore-webui -Jport 8080 \
26                -JnumUser $NUM_USERS -JrampUp 1 \
27                -l mylogfile.log -n
28          env:
29            - name: NUM_USERS
30              value: "20"
31          volumeMounts:
32            - name: jmeter-data
33              mountPath: /data
```

Resource Dimension

(1) Horizontal Scalability

```
10   name: teastore-deployment
11   files:
12   - teastore-auth-deployment.yaml
13   - teastore-auth-service.yaml
14   - teastore-db-deployment.yaml
15   - teastore-db-service.yaml
16   - teastore-image-deployment.yaml
17   - teastore-image-service.yaml
18   - teastore-persistence-deployment.yaml
19   - teastore-persistence-service.yaml
20   - teastore-recommender-deployment.yaml
21   - teastore-recommender-service.yaml
22   - teastore-registry-deployment.yaml
23   - teastore-registry-service.yaml
24   - teastore-webui-deployment.yaml
25   - teastore-webui-nodeport.yaml
26   loadGenerator:
27     resources:
28     - configMap:
29       name: teastore-jmeter-deployment
30       files:
31       - jmeter.yaml
32   resourceTypes:
33   - typeName: Instances
34     patchers:
35     - type: ReplicaPatcher
36       resource: teastore-auth-deployment.yaml
37     - type: ReplicaPatcher
38       resource: teastore-image-deployment.yaml
39     - type: ReplicaPatcher
40       resource: teastore-persistence-deployment.yaml
41     - type: ReplicaPatcher
42       resource: teastore-recommender-deployment.yaml
43     - type: ReplicaPatcher
44       resource: teastore-webui-deployment.yaml
45   - typeName: "PodResources"
46     patchers:
47     - type: ResourceLimitPatcher
48       resource: teastore-auth-deployment.yaml
49     properties:
50       container: teastore-auth
51       limitedResource: cpu
52       format: m
53     - type: ResourceLimitPatcher
```

```
TeaStore - recommender-deployment.yaml
1  apiVersion: apps/v1
2  kind: Deployment
3  metadata:
4    name: teastore-recommender
5  spec:
6    replicas: 15
7    selector:
8      matchLabels:
9        app: teastore
10       run: teastore-recommender
11    template:
12      metadata:
13        labels:
14          app: teastore
15          run: teastore-recommender
16      spec:
17        terminationGracePeriodSeconds: 0
18        containers:
19        - name: teastore-recommender
20          image: descartesresearch/teastore-reco
21          mmender:development
22          imagePullPolicy: Always
23          ports:
24            - containerPort: 8080
```

Resource Dimension

(2) Vertical Scalability

```
41 - type: ReplicaPatcher
42   resource: teastore-recommender-deployment.yaml
43 - type: ReplicaPatcher
44   resource: teastore-webui-deployment.yaml
45 - typeName: "PodResources"
46   patchers:
47   - type: ResourceLimitPatcher
48     resource: teastore-auth-deployment.yaml
49     properties:
68       properties:
69         container: teastore-auth
70         limitedResource: memory
71         format: Mi
72         factor: 2
73   - type: ResourceLimitPatcher
74     resource: teastore-image-deployment.yaml
75     properties:
76       container: teastore-image
77       limitedResource: cpu
78       format: m
79   - type: ResourceLimitPatcher
80     resource: teastore-image-deployment.yaml
81     properties:
82       container: teastore-image
83       limitedResource: memory
84       format: Mi
85       factor: 2
86   - type: ResourceLimitPatcher
87     resource: teastore-recommender-deployment.yaml
88     properties:
89       container: teastore-recommender
90       limitedResource: cpu
91       format: m
92   - type: ResourceLimitPatcher
93     resource: teastore-recommender-deployment.yaml
94     properties:
95       container: teastore-recommender
96       limitedResource: memory
97       format: Mi
98       factor: 2
99   - type: ResourceLimitPatcher
100     resource: teastore-webui-deployment.yaml
101     properties:
102       container: teastore-webui
```

TeaStore - recommender-deployment.yaml

```
1  apiVersion: apps/v1
2  kind: Deployment
3  metadata:
4    name: teastore-recommender
5  spec:
34    failureThreshold: 3
35    httpGet:
36      path: "/tools.descartes.teastore.recommender/rest/train/isready"
37      port: 8080
38      scheme: HTTP
39    resources:
40      limits:
41        memory: 1000Mi
42        cpu: 500m
43    nodeSelector:
44      env: dev
```


Load Dimension

```
94   properties:
95     container: teastore-recommender
96     limitedResource: memory
97     format: Mi
98     factor: 2
99   - type: ResourceLimitPatcher
100     resource: teastore-webui-deployment.yaml
101     properties:
102       container: teastore-webui
103       limitedResource: cpu
104       format: m
105   - type: ResourceLimitPatcher
106     resource: teastore-webui-deployment.yaml
107     properties:
108       container: teastore-webui
109       limitedResource: memory
110       format: Mi
111       factor: 2
112   loadTypes:
113   - typeName: NumUsers
114     patchers:
115     - type: EnvVarPatcher
116       resource: jmeter.yaml
117       properties:
118         container: jmeter
119         variableName: NUM_USERS
120   slo:
121   - sloType: generic
122     name: uiLatency
123     prometheusUrl: "http://prometheus-operated:9090"
124     offset: 0
125     properties:
126       externalSloUrl: "http://localhost:8082"
127       promQLQuery: "histogram_quantile(0.95, sum(irat
128 e(osm_request_duration_ms_bucket{destination_name='tea
129 store_webui'}[1m])) by (le, destination_name))"
130       warmup: 600 # in seconds
131       queryAggregation: max
132       repetitionAggregation: median
133       operator: lte
134       threshold: 200
```

```
TeaStore - jmeter.yaml
1  apiVersion: apps/v1
2  kind: Deployment
3  metadata:
4    name: jmeter
5    annotations:
6      openservicemesh.io/sidecar-injection: enabled
7  spec:
8    replicas: 1
9
10 containers:
11   - name: jmeter
12     image: justb4/jmeter:5.4
13     command: [ "/bin/bash", "-c", "--" ]
14     args:
15     - >
16       java -jar bin/ApacheJMeter.jar \
17       -t profile/teastore_browse_nogui.jmx \
18       -Jhostname teastore-webui -Jport 8080 \
19       -JnumUser $NUM_USERS -JrampUp 1 \
20       -l mylogfile.log -n
21     env:
22     - name: NUM_USERS
23       value: "20"
24     volumeMounts:
25     - name: jmeter-profile
26       mountPath: /opt/apache-jmeter-5.4.3/profile
27     ports:
28     - containerPort: 50000
29     - containerPort: 60000
30     - containerPort: 1099
31     volumes:
```

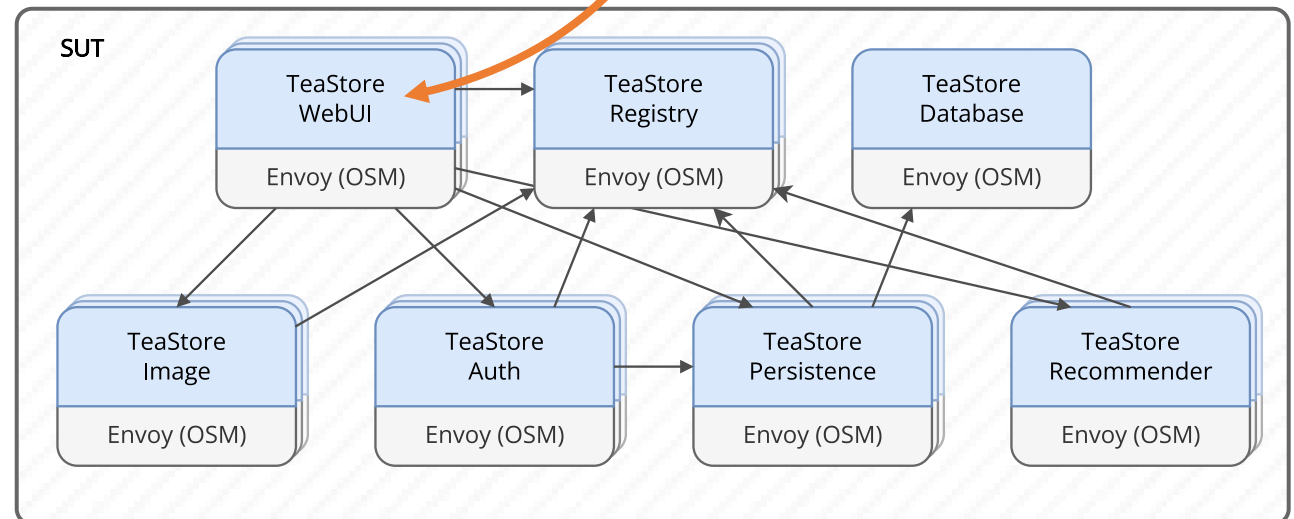
```

94     properties:
95       container: teastore-recommender
96       limitedResource: memory
97       format: Mi
98       factor: 2
99   - type: ResourceLimitPatcher
100     resource: teastore-webui-deployment.yaml
101     properties:
102       container: teastore-webui
103       limitedResource: cpu
104       format: m
105   - type: ResourceLimitPatcher
106     resource: teastore-webui-deployment.yaml
107     properties:
108       container: teastore-webui
109       limitedResource: memory
110       format: Mi
111       factor: 2
112   loadTypes:
113   - typeName: NumUsers
114   patchers:
115   - type: EnvVarPatcher
116     resource: jmeter.yaml
117     properties:
118       container: jmeter
119       variableName: NUM_USERS
120   slos:
121   - sloType: generic
122     name: uiLatency
123     prometheusUrl: "http://prometheus-operated:9090"
124     offset: 0
125     properties:
126       externalSloUrl: "http://localhost:8082"
127       promQLQuery: "histogram_quantile(0.95,sum(irat
128         e(osm_request_duration_ms_bucket{destination_name='tea
129         store_webui'}[1m])) by (le, destination_name))"
130       warmup: 600 # in seconds
131       queryAggregation: max
132       repetitionAggregation: median
133       operator: lte
134       threshold: 200

```

SLO

p95 latency <= 200 ms ?



Deployment

```
TeaStore - benchmark.yaml
1  apiVersion: theodolite.rocks/v1beta1
2  kind: benchmark
3  metadata:
4    name: teastore
5  spec:
6    waitForResourcesEnabled: true
7    sut:
8      resources:
9      - configMap:
10        name: teastore-deployment
11        files:
12        - teastore-auth-deployment.yaml
13        - teastore-auth-service.yaml
14        - teastore-db-deployment.yaml
15        - teastore-db-service.yaml
16        - teastore-image-deployment.yaml
17        - teastore-image-service.yaml
18        - teastore-persistence-deployment.yaml
19        - teastore-persistence-service.yaml
20        - teastore-recommender-deployment.yaml
21        - teastore-recommender-service.yaml
22        - teastore-registry-deployment.yaml
23        - teastore-registry-service.yaml
24        - teastore-webui-deployment.yaml
25        - teastore-webui-nodeport.yaml
26  loadGenerator:
27    resources:
28    - configMap:
29      name: teastore-jmeter-deployment
30      files:
31      - jmeter.yaml
32  resourceTypes:
33  - typeName: Instances
34    patchers:
35    - type: ReplicaPatcher
36      resource: teastore-auth-deployment.yaml
37    - type: ReplicaPatcher
38      resource: teastore-image-deployment.yaml
39    - type: ReplicaPatcher
40      resource: teastore-persistence-deployment.yaml
```

```
TeaStore - bash
$ kubectl apply -f benchmark.yaml
benchmark.theodolite.rocks/teastore created
```

Deployment

```
TeaStore - benchmark.yaml
1  apiVersion: theodolite.rocks/v1beta1
2  kind: benchmark
3  metadata:
4    name: teastore
5  spec:
6    waitForResourcesEnabled: true
7    sut:
8      resources:
9      - configMap:
10        name: teastore-deployment
11        files:
12        - teastore-auth-deployment.yaml
13        - teastore-auth-service.yaml
14        - teastore-db-deployment.yaml
15        - teastore-db-service.yaml
16        - teastore-image-deployment.yaml
17        - teastore-image-service.yaml
18        - teastore-persistence-deployment.yaml
19        - teastore-persistence-service.yaml
20        - teastore-recommender-deployment.yaml
21        - teastore-recommender-service.yaml
22        - teastore-registry-deployment.yaml
23        - teastore-registry-service.yaml
24        - teastore-webui-deployment.yaml
25        - teastore-webui-nodeport.yaml
26  loadGenerator:
27    resources:
28    - configMap:
29      name: teastore-jmeter-deployment
30      files:
31      - jmeter.yaml
32  resourceTypes:
33  - typeName: Instances
34    patchers:
35    - type: ReplicaPatcher
36      resource: teastore-auth-deployment.yaml
37    - type: ReplicaPatcher
38      resource: teastore-image-deployment.yaml
39    - type: ReplicaPatcher
40      resource: teastore-persistence-deployment.yaml
```

```
TeaStore - bash
$ kubectl apply -f benchmark.yaml
benchmark.theodolite.rocks/teastore created

$ kubectl get benchmarks
NAME          AGE   STATUS
teastore      25s   Ready
```

Running the Benchmark

```
TeaStore - execution-horizontal.yaml
1  apiVersion: theodolite.rocks/v1beta1
2  kind: execution
3  metadata:
4    name: teastore-horizontal
5  spec:
6    benchmark: teastore
7    load:
8      loadType: NumUsers
9      loadValues: [5, 10, 15, 20, 25, 30, 35, 40, 45, 50]
10   resources:
11     resourceType: Instances
12     resourceValues: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11,
13 12, 13, 14, 15, 16, 17, 18, 19, 20]
14   slos:
15     - name: uiLatency
16       properties:
17         warmup: 600 # in seconds
18         threshold: 200
19   execution:
20     strategy:
21       name: RestrictionSearch
22       restrictions:
23         - LowerBound
24       searchStrategy: LinearSearch
25     duration: 1200 # in seconds
26     repetitions: 1
27   configOverrides: []
```

```
TeaStore - bash
$ kubectl apply -f execution-horizontal.yaml
execution.theodolite.rocks/teastore-horizontal created
```

Running the Benchmark

```
TeaStore - execution-horizontal.yaml
1 apiVersion: theodolite.rocks/v1beta1
2 kind: execution
3 metadata:
4   name: teastore-horizontal
5 spec:
6   benchmark: teastore
7   load:
8     loadType: NumUsers
9     loadValues: [5, 10, 15, 20, 25, 30, 35, 40, 45, 50]
10  resources:
11    resourceType: Instances
12    resourceValues: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11,
13 12, 13, 14, 15, 16, 17, 18, 19, 20]
14  slos:
15    - name: uiLatency
16      properties:
17        warmup: 600 # in seconds
18        threshold: 200
19  execution:
20    strategy:
21      name: RestrictionSearch
22      restrictions:
23        - LowerBound
24      searchStrategy: LinearSearch
25    duration: 1200 # in seconds
26    repetitions: 1
27    configOverrides: []
```

```
TeaStore - bash
$ kubectl apply -f execution-horizontal.yaml
execution.theodolite.rocks/teastore-horizontal created

$ kubectl get executions
NAME                  STATUS    DURATION    AGE
teastore-horizontal  Pending             3s
```

Running the Benchmark

```
TeaStore - execution-horizontal.yaml
1  apiVersion: theodolite.rocks/v1beta1
2  kind: execution
3  metadata:
4    name: teastore-horizontal
5  spec:
6    benchmark: teastore
7    load:
8      loadType: NumUsers
9      loadValues: [5, 10, 15, 20, 25, 30, 35, 40, 45, 50]
10   resources:
11     resourceType: Instances
12     resourceValues: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11,
13 12, 13, 14, 15, 16, 17, 18, 19, 20]
14   slos:
15     - name: uiLatency
16       properties:
17         warmup: 600 # in seconds
18         threshold: 200
19   execution:
20     strategy:
21       name: RestrictionSearch
22       restrictions:
23         - LowerBound
24       searchStrategy: LinearSearch
25     duration: 1200 # in seconds
26     repetitions: 1
27   configOverrides: []
```

```
TeaStore - bash
$ kubectl apply -f execution-horizontal.yaml
execution.theodolite.rocks/teastore-horizontal created

$ kubectl get executions
NAME                STATUS    DURATION    AGE
teastore-horizontal Pending    3s

$ kubectl get executions
NAME                STATUS    DURATION    AGE
teastore-horizontal Running     9s          13s
```

Running the Benchmark

```
TeaStore - execution-horizontal.yaml
1 apiVersion: theodolite.rocks/v1beta1
2 kind: execution
3 metadata:
4   name: teastore-horizontal
5 spec:
6   benchmark: teastore
7   load:
8     loadType: NumUsers
9     loadValues: [5, 10, 15, 20, 25, 30, 35, 40, 45, 50]
10  resources:
11    resourceType: Instances
12    resourceValues: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11,
13    12, 13, 14, 15, 16, 17, 18, 19, 20]
14  slos:
15    - name: uiLatency
16      properties:
17        warmup: 600 # in seconds
18        threshold: 200
19  execution:
20    strategy:
21      name: RestrictionSearch
22      restrictions:
23        - LowerBound
24      searchStrategy: LinearSearch
25    duration: 1200 # in seconds
26    repetitions: 1
27    configOverrides: []
```

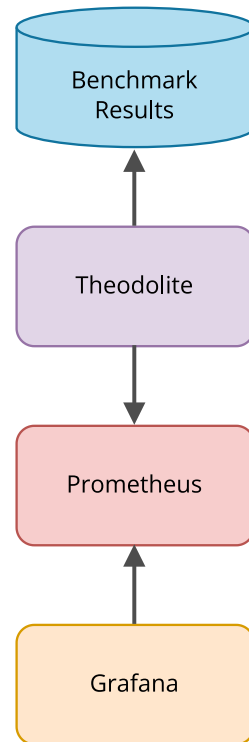
```
TeaStore - bash
$ kubectl apply -f execution-horizontal.yaml
execution.theodolite.rocks/teastore-horizontal created

$ kubectl get executions
NAME                STATUS    DURATION    AGE
teastore-horizontal Pending    3s

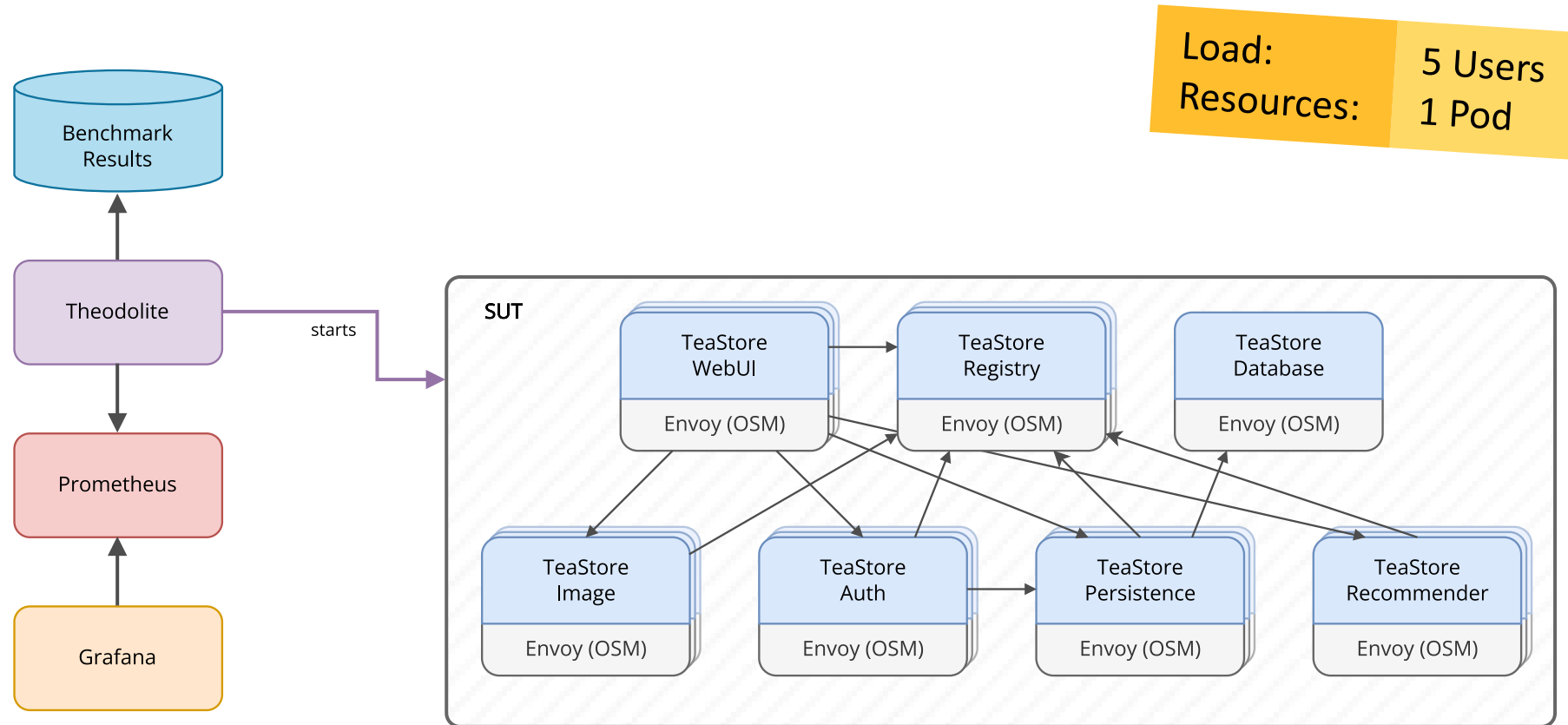
$ kubectl get executions
NAME                STATUS    DURATION    AGE
teastore-horizontal Running    9s          13s
```



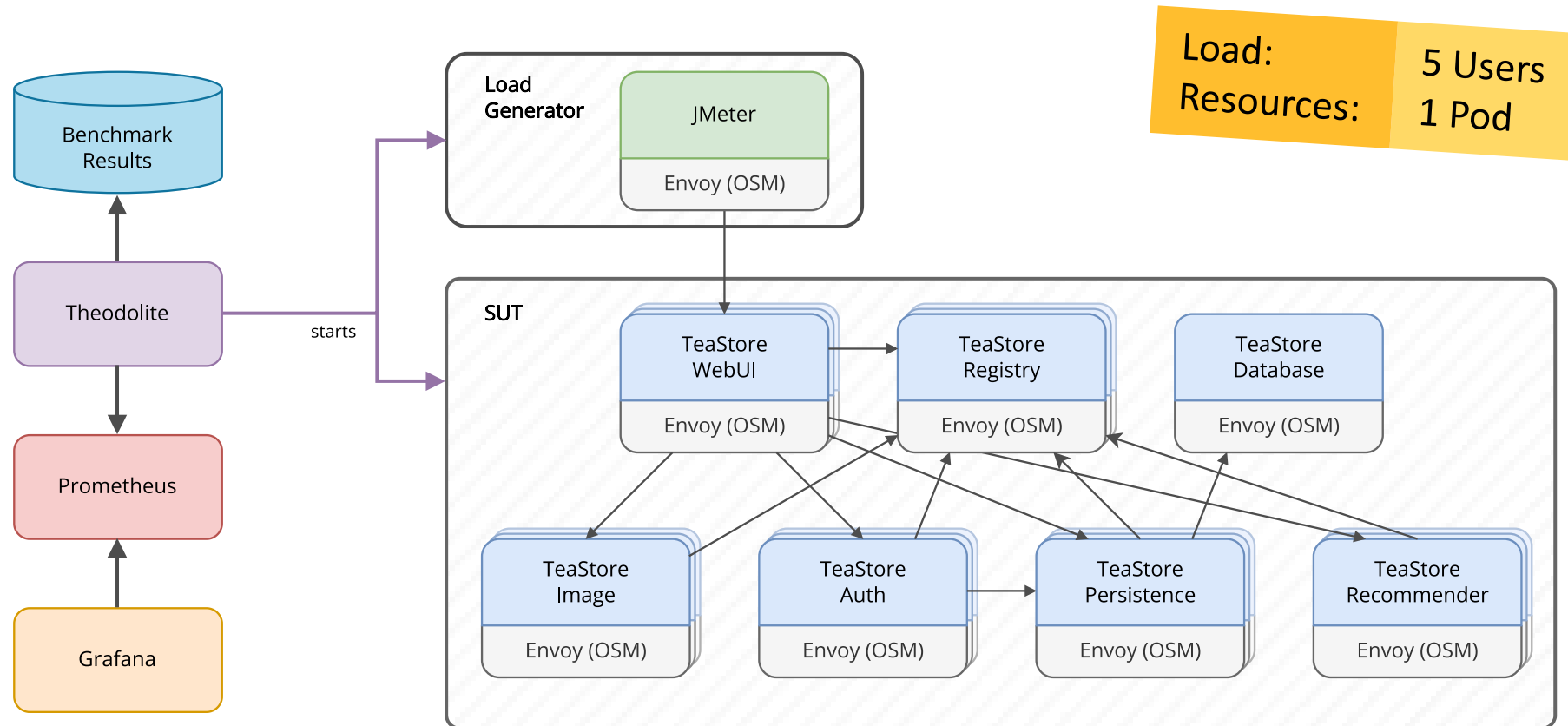
So, what's going on inside?



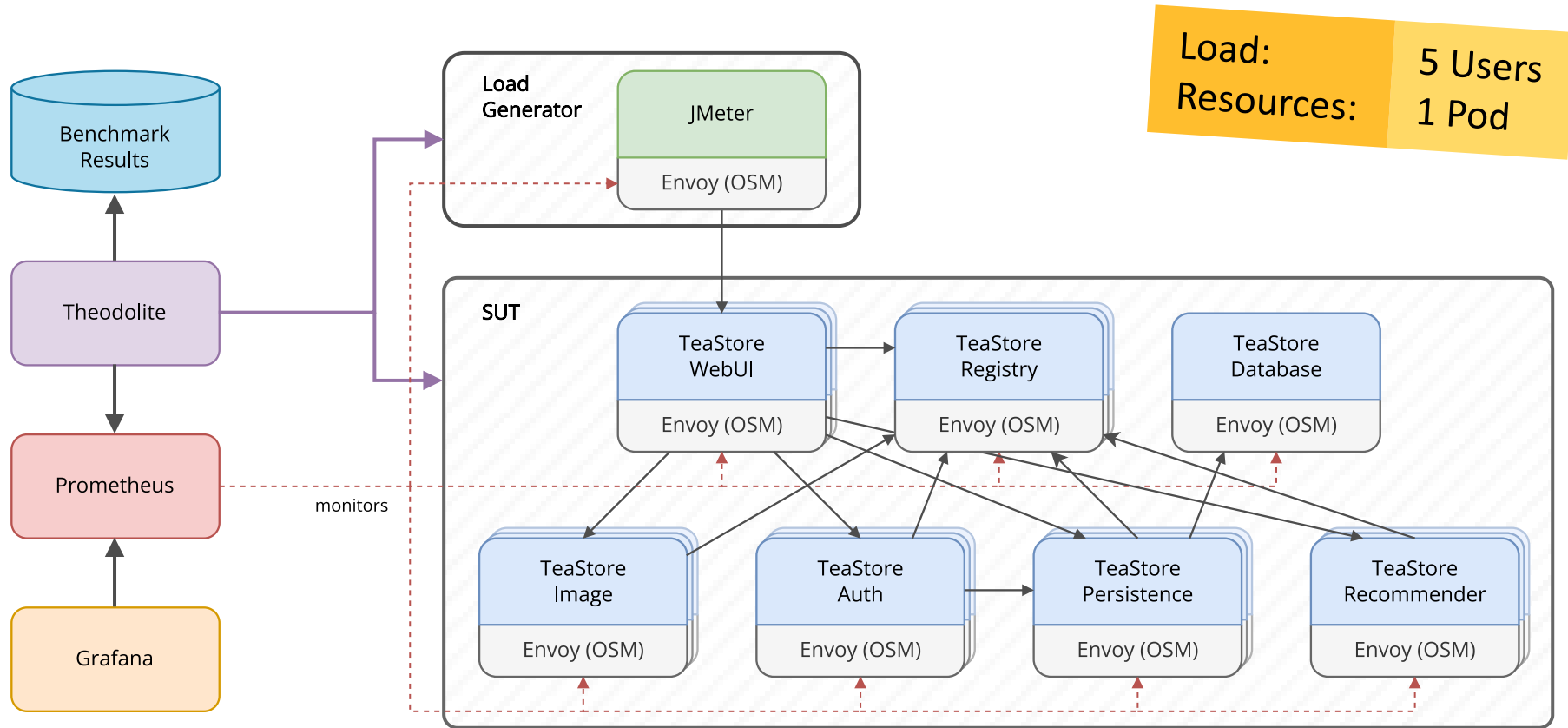
So, what's going on inside?



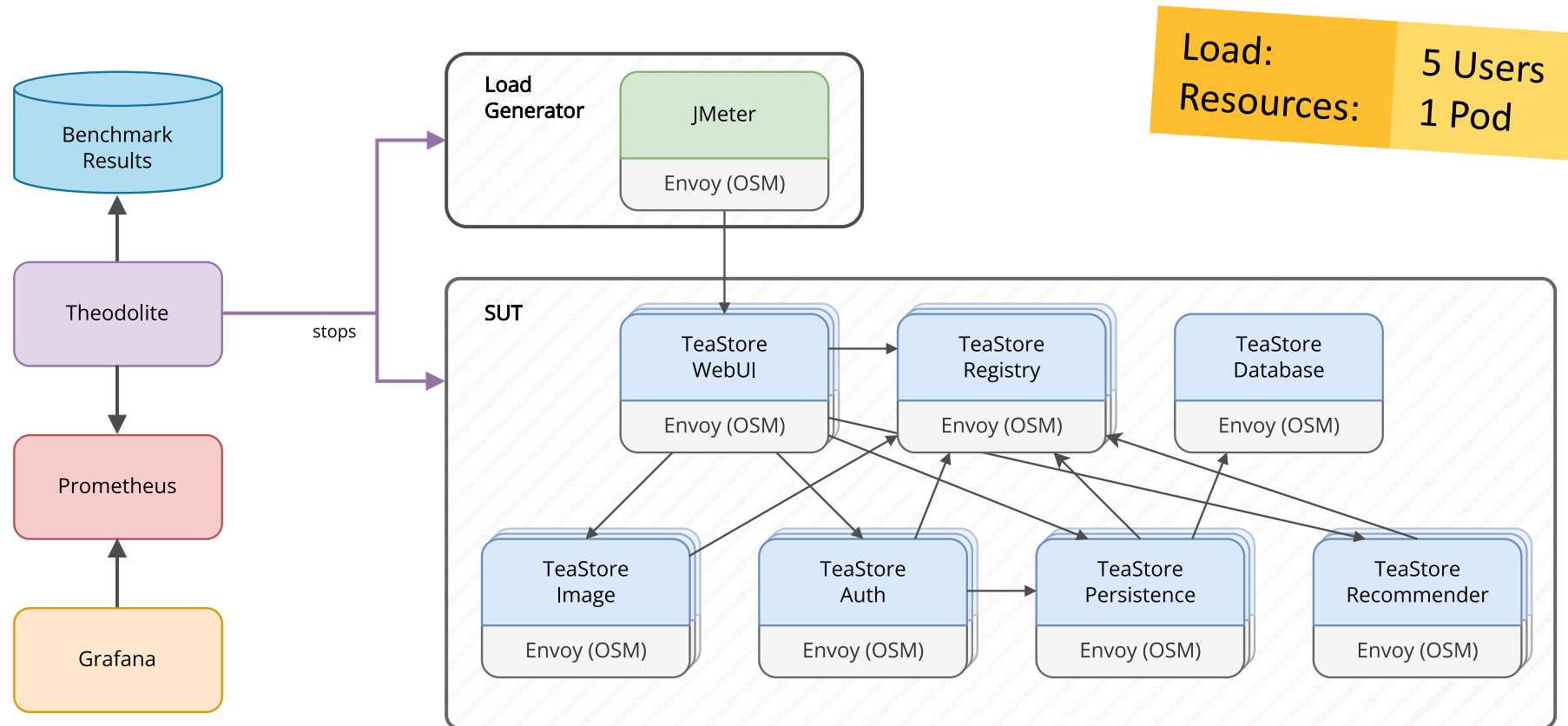
So, what's going on inside?



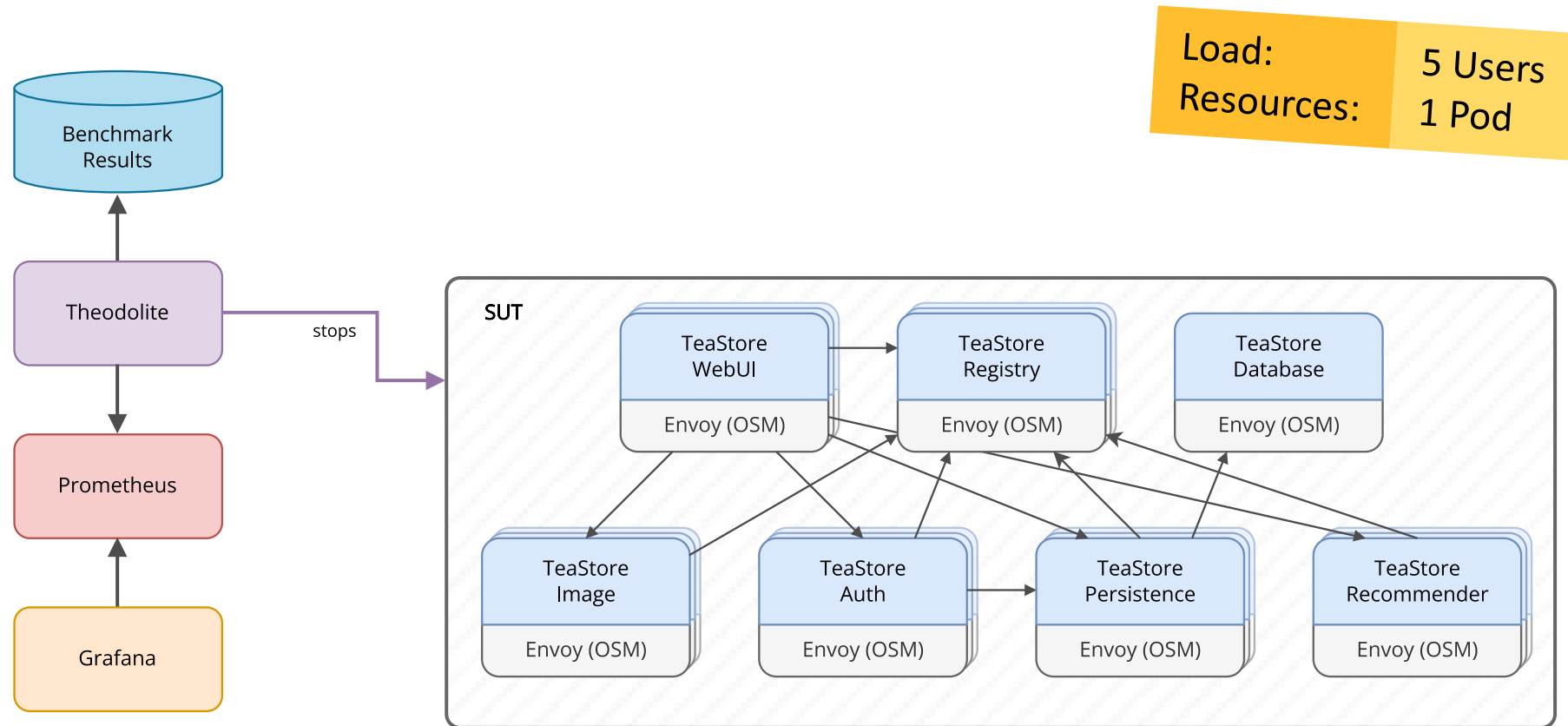
So, what's going on inside?



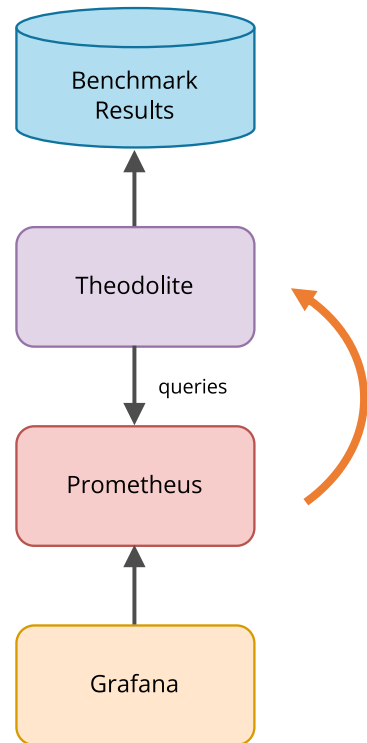
So, what's going on inside?



So, what's going on inside?



So, what's going on inside?



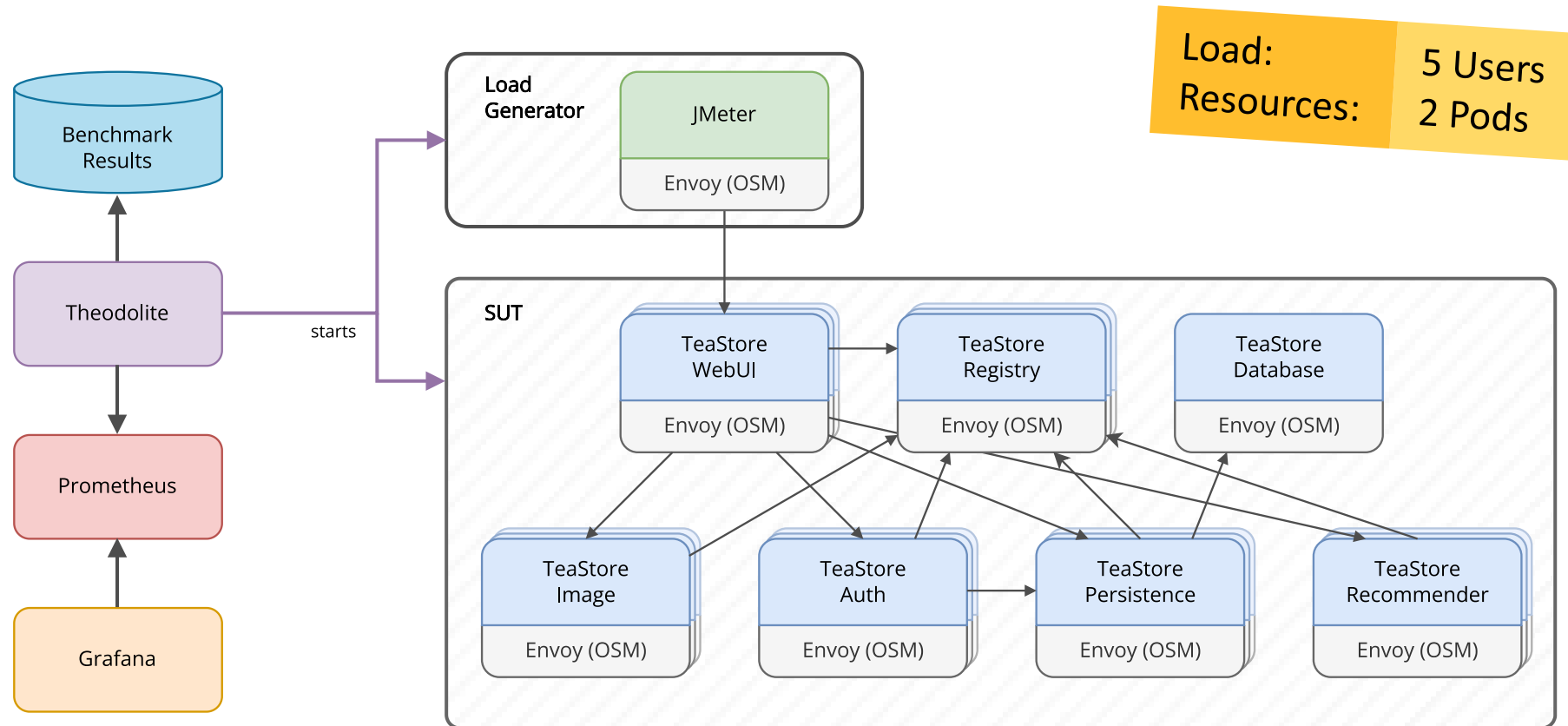
Load:
Resources:

5 Users
1 Pod

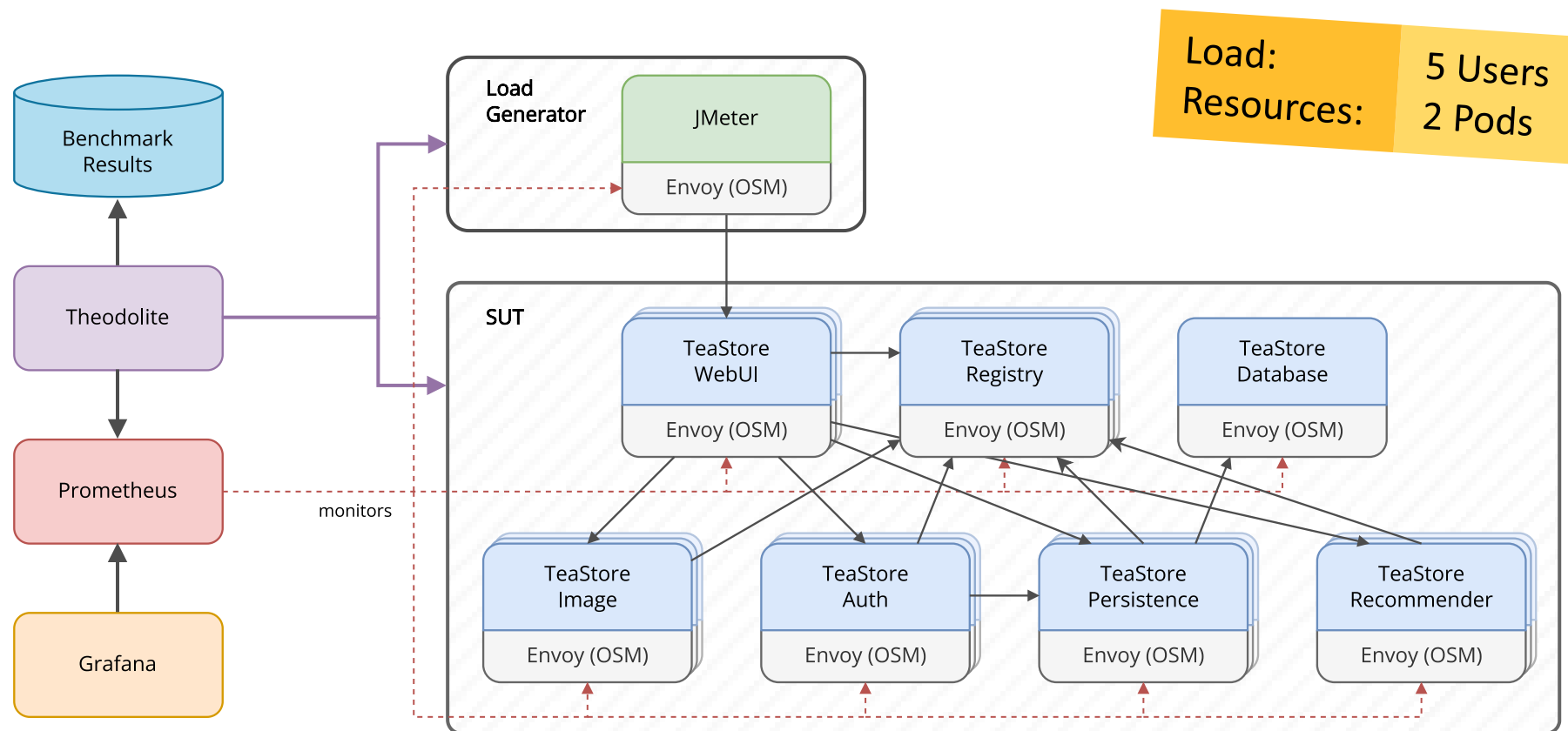
p95 latency = 298 ms



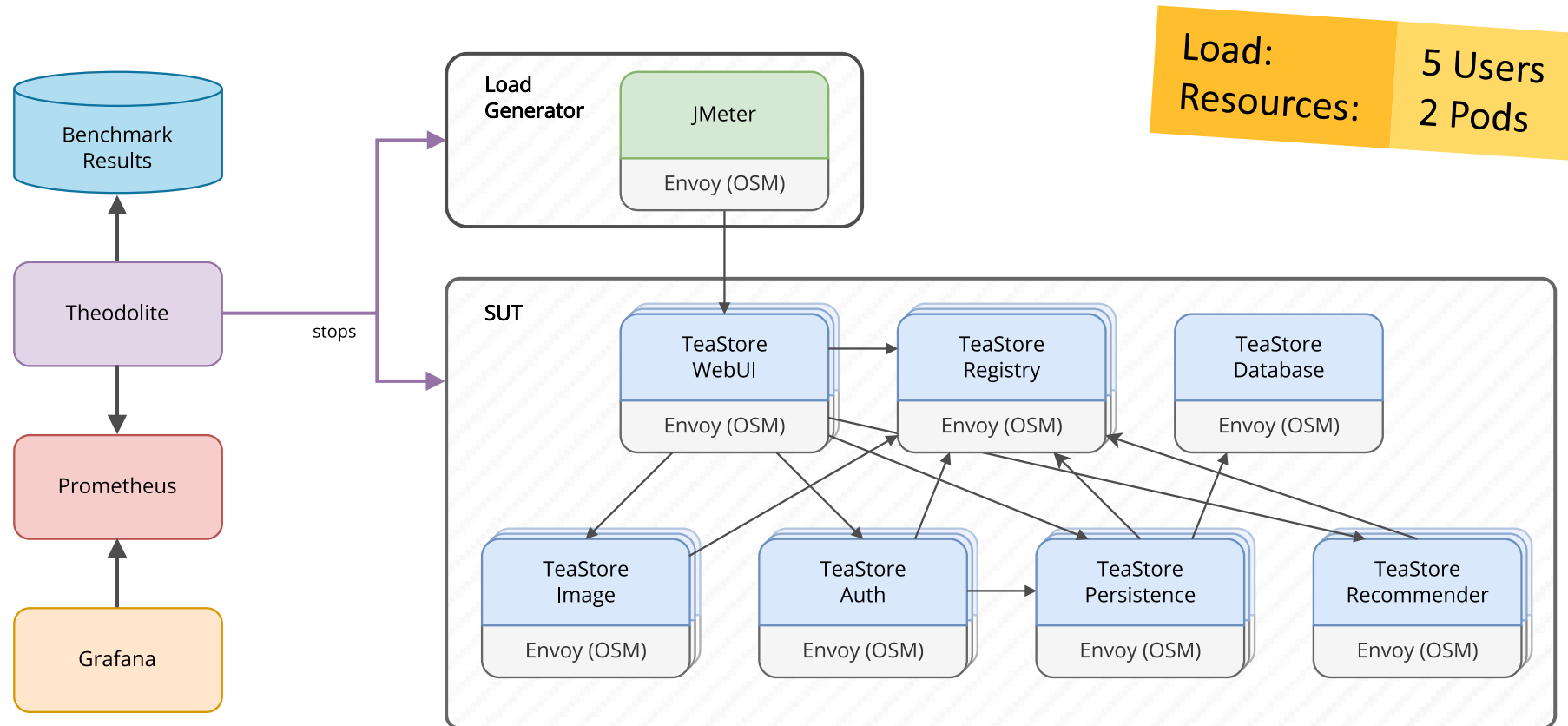
So, what's going on inside?



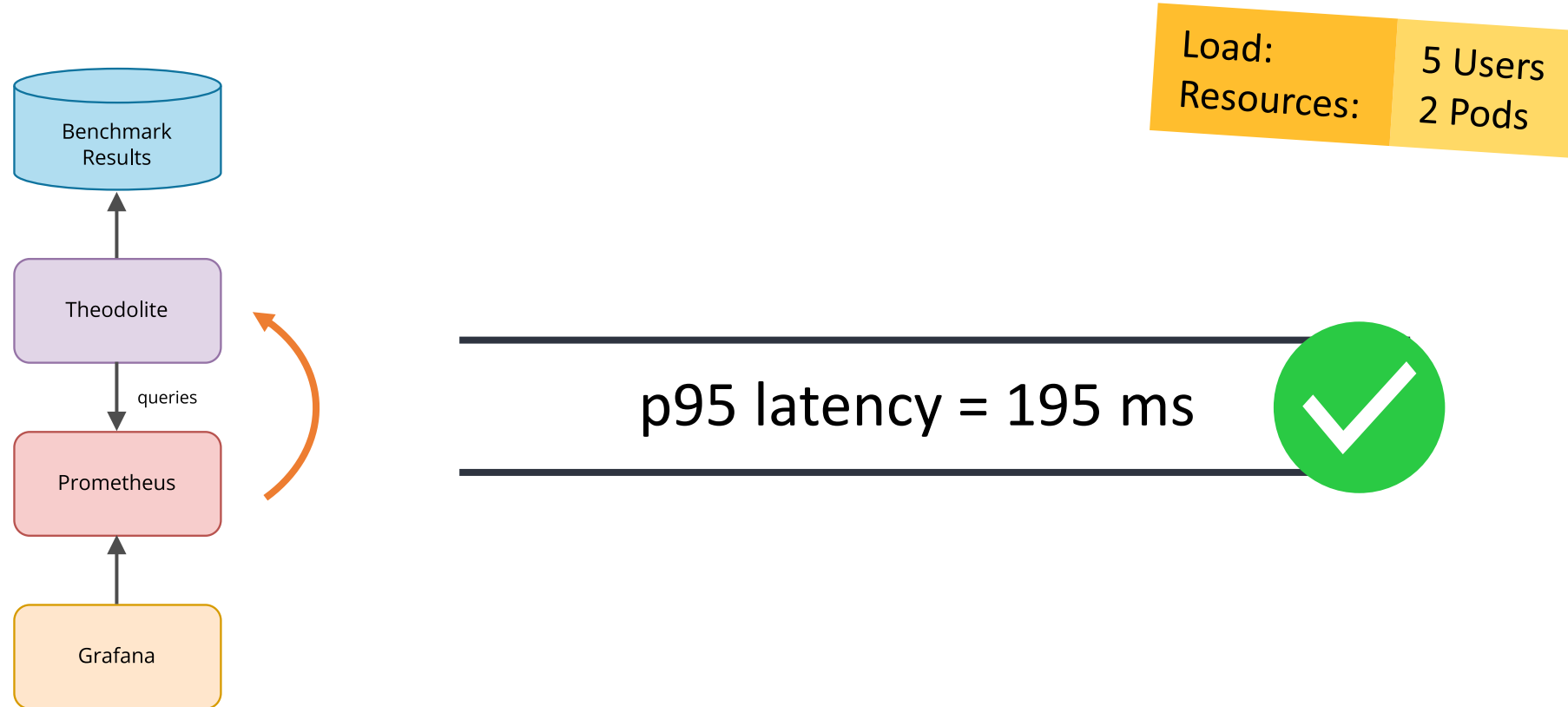
So, what's going on inside?



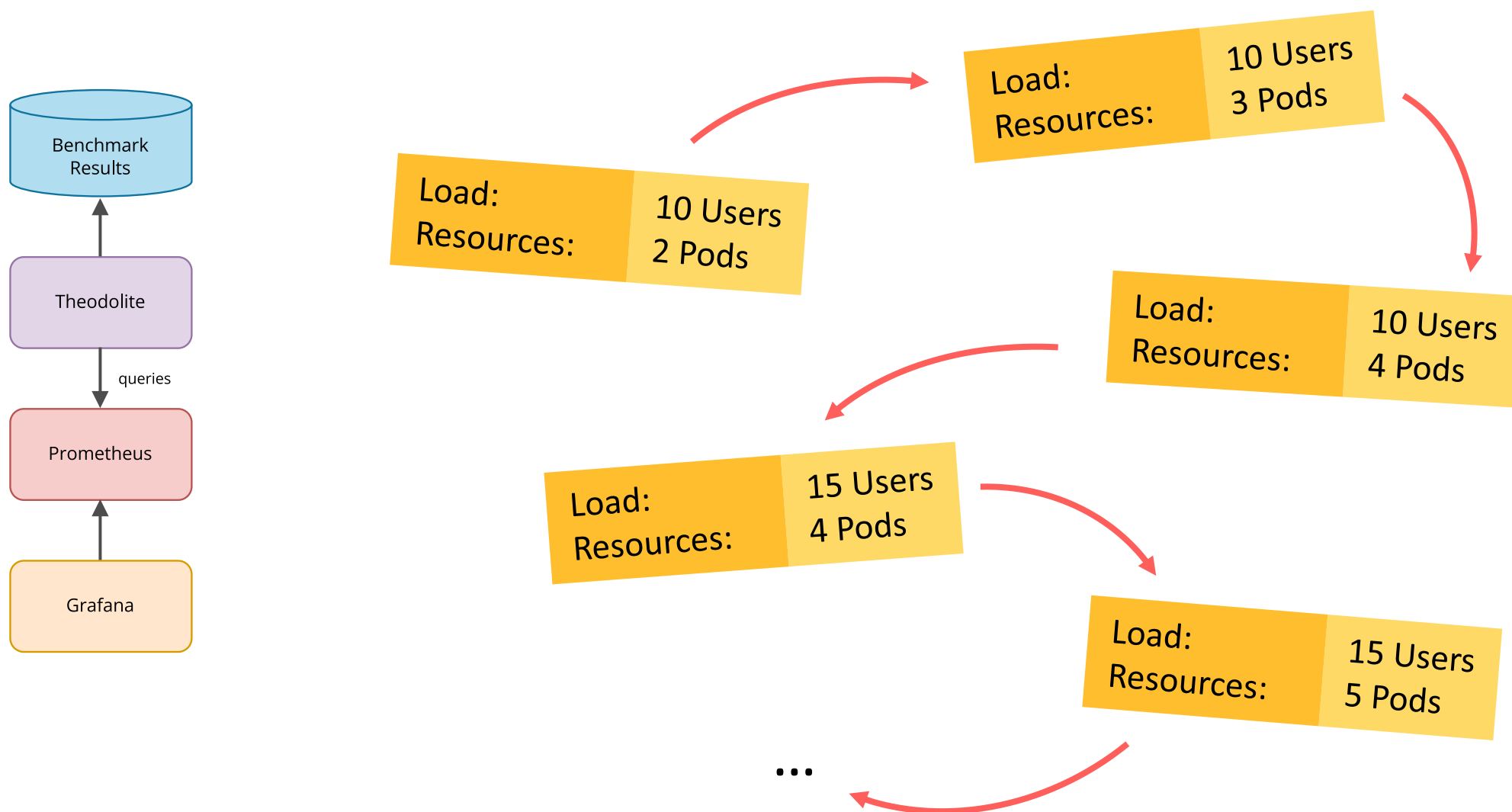
So, what's going on inside?



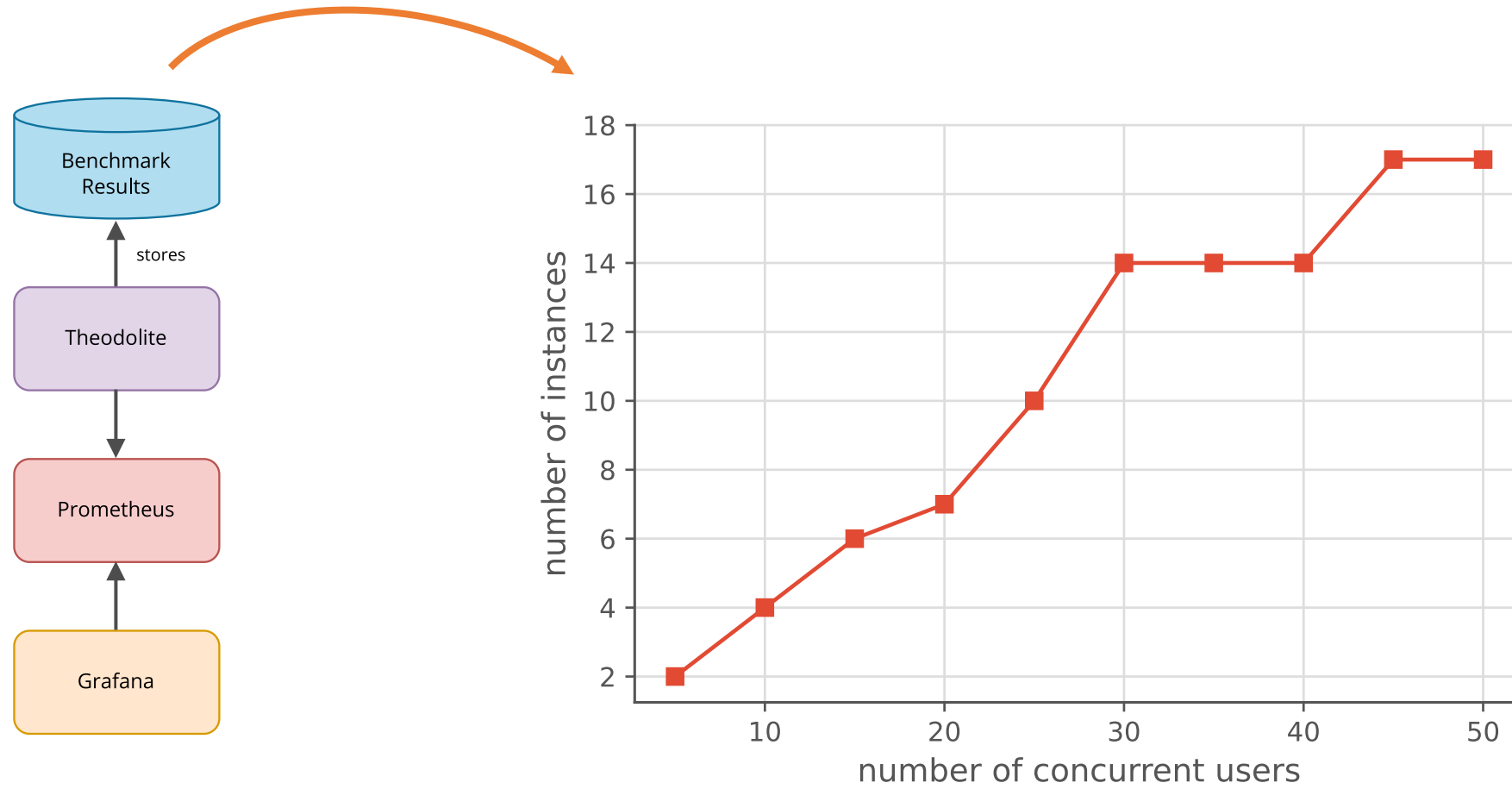
So, what's going on inside?



So, what's going on inside?

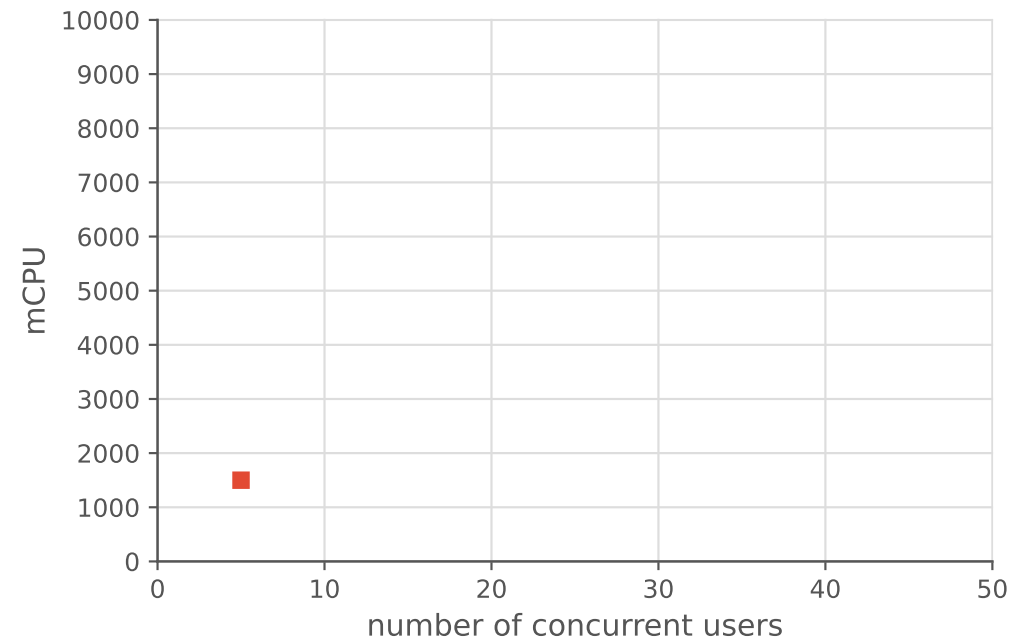


...and the result is:



Benchmarking Vertical Scalability

```
TeaStore - execution-vertical.yaml
1  apiVersion: theodolite.rocks/v1beta1
2  kind: execution
3  metadata:
4    name: teastore-vertical
5  spec:
6    benchmark: teastore
7    load:
8      loadType: NumUsers
9      loadValues: [5, 10, 15, 20, 25, 30, 35, 40, 45, 50]
10   resources:
11     resourceType: PodResources
12     resourceValues: [500, 1000, 1500, 2000, 2500, 3000,
13     3500, 4000, 4500, 5000, 5500, 6000, 6500, 7000, 7500, 8
14     000, 8500, 9000, 9500, 10000]
15   slos:
16     - name: uiLatency
17       properties:
18         warmup: 600 # in seconds
19         threshold: 200
20   execution:
21     strategy:
22       name: RestrictionSearch
23       restrictions:
24         - LowerBound
25       searchStrategy: LinearSearch
26     duration: 1200 # in seconds
27     repetitions: 1
28     configOverrides: []
```

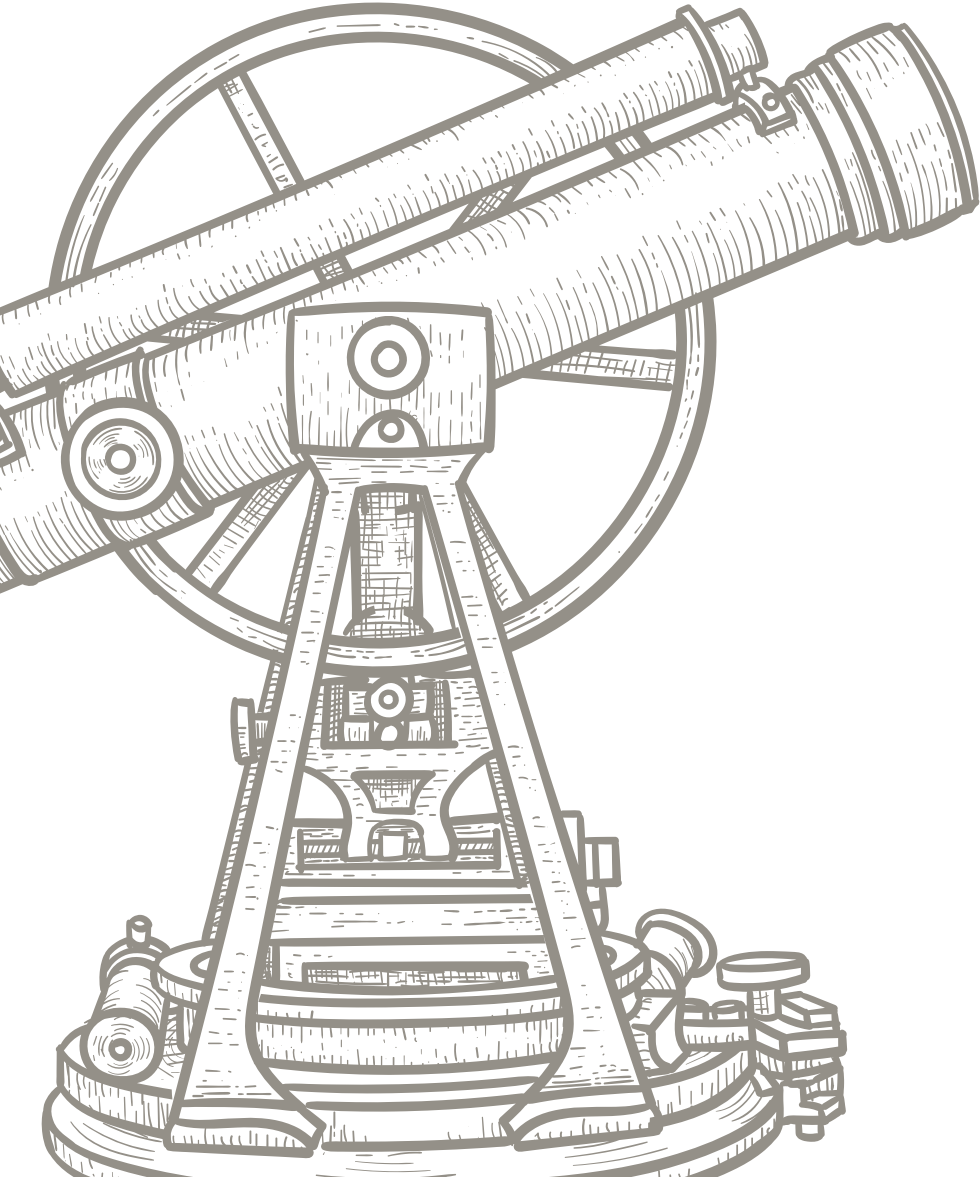


Conclusions & Lessons Learned



- Favor **open workload** over closed workload models
- Finding **good SLOs** is hard
- **Cloud-native-ness** of the TeaStore could be improved
- **Interested?** Theodolite's TeaStore benchmark is now ready to use!

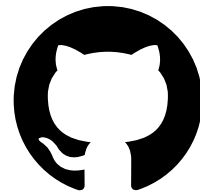
Getting Started



Theodolite



[https://www.
theodolite.rocks](https://www.theodolite.rocks)



[https://github.com/
cau-se/theodolite](https://github.com/cau-se/theodolite)