Deploying and running your first application on K8s

Alexander Reelsen
alex@elastic.co | @spinscale

Today's goal

> How to **build**, **run** & **maintain**
> a modern java web application
> with minimal resources
> on K8s

Rocket-science free zone!

- ⚠️ Level: Intro
- ⚠️ Perspective: Developer, user of existing K8s cluster
- ⚠️ Journey from nothing to downtime free rollout during peak traffic

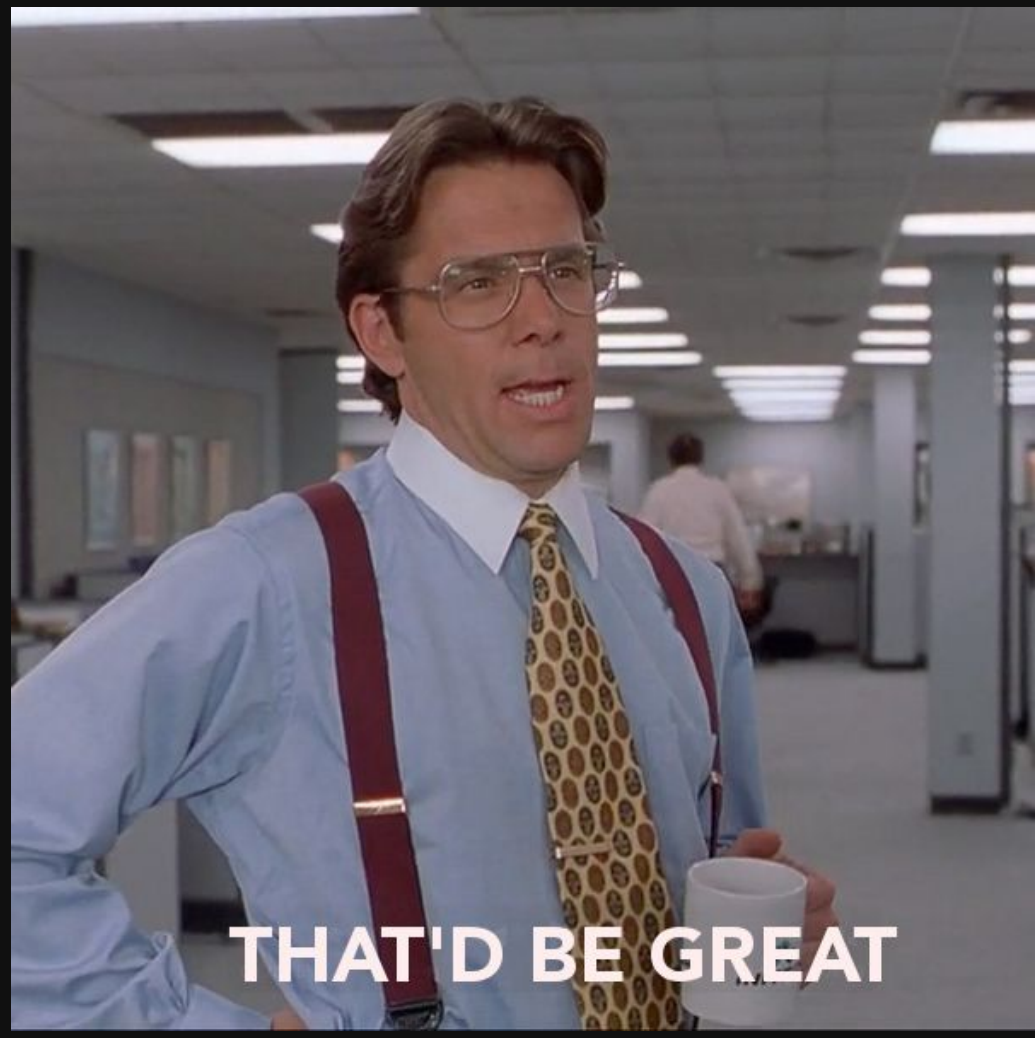
About me

- Developer & Advocate @Elastic
- PaaS fan, IaC fan
- K8s skeptic: Primitives 👍 level of abstraction ❓
- First rule of SWE: *Don't write code, if you don't want to maintain it*...

Elastic Community Conference

- Organized by the Elastic Community Team
- Virtual
- Around the clock
- Several languages
- No talks from Elastic Community Team members
- 2021 was a success, 70 talks

2022: ElasticCC Registration via Elastic Cloud



Discussion

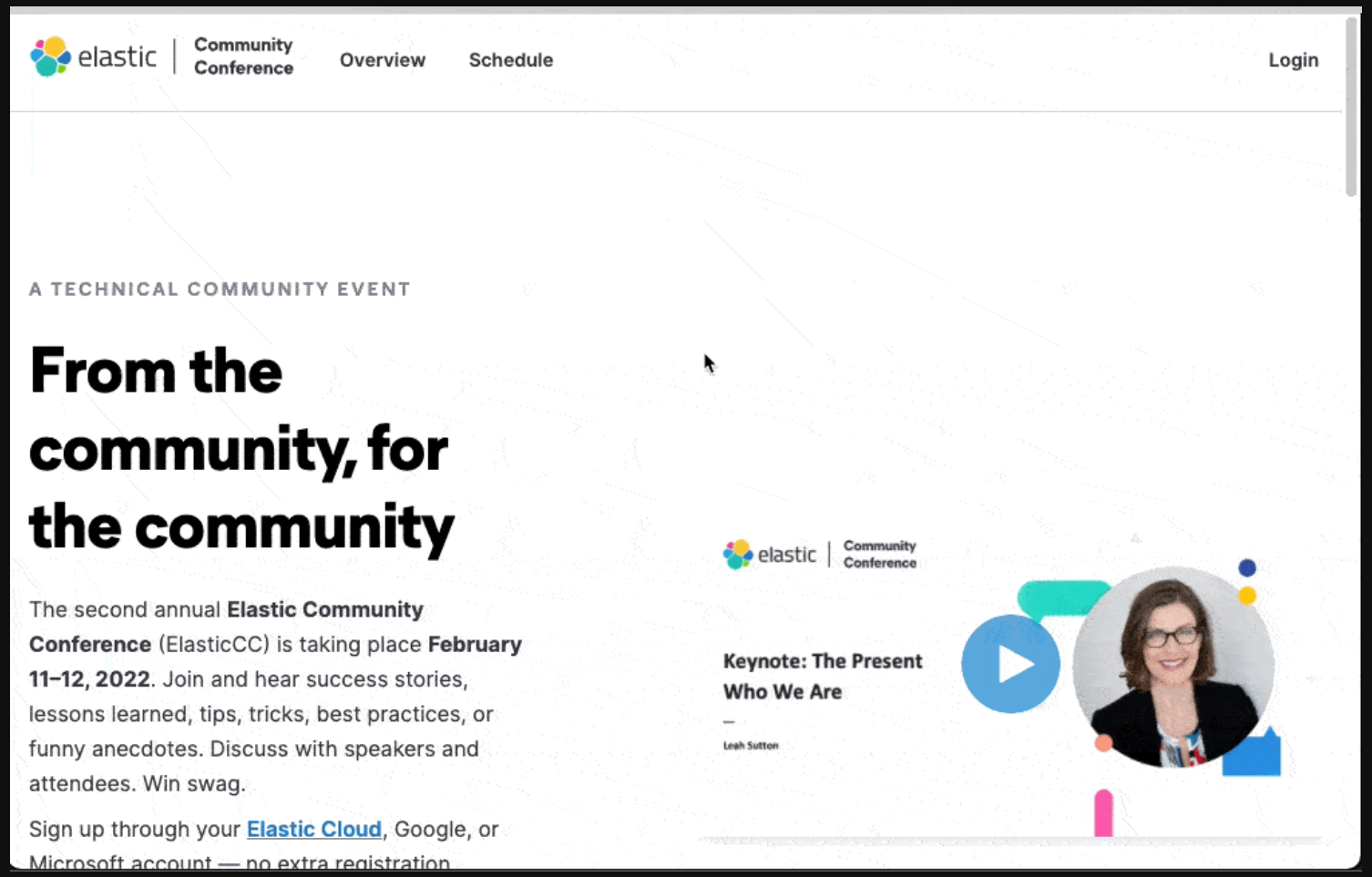
- Decision: `Build` vs. Buy (Registration, Live Streaming)
- Platform: PaaS vs. `K8s` (no approval required)
- Datastore: SQL vs. `Elastic Cloud` vs. `API`
- **Let's do this**: Own web application

Discussion

- Decision: `Build` vs. Buy (Registration, Live Streaming)
- Platform: PaaS vs. `K8s` (no approval required)
- Datastore: Sql vs. `Elastic Cloud` vs. `API`
- **Let's do this**: Own web application

> Use your own technologies in production
>
> *--Me*

Login via Cloud



Schedule

elastic | Community Conference

Overview Schedule Login

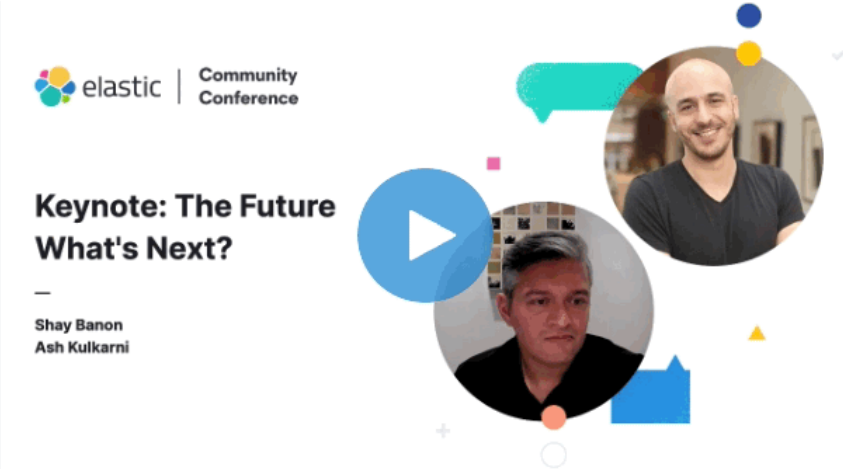
A TECHNICAL COMMUNITY EVENT

From the community, for the community

The second annual **Elastic Community Conference** (ElasticCC) is taking place **February 11–12, 2022**. Join and hear success stories, lessons learned, tips, tricks, best practices, or funny anecdotes. Discuss with speakers and attendees. Win swag.

Sign up through your Elastic Cloud, Google, or Microsoft account — no extra registration required.

Join us for free



Feedback

elastic | Community Conference Overview Schedule Profile Logout

Keynote: The Past — 10 Years Elastic

February 11, 2022

17:00 — 17:35 (EUROPE/BERLIN)
16:00 — 16:35 (UTC)

Feedback Take the quiz Add to my Agenda

Keynote: The Past — 10 Years Elastic

Q&A with the Elasticsearch founders — back when the company wasn't even called Elastic yet. Some of the questions we are asking them are:

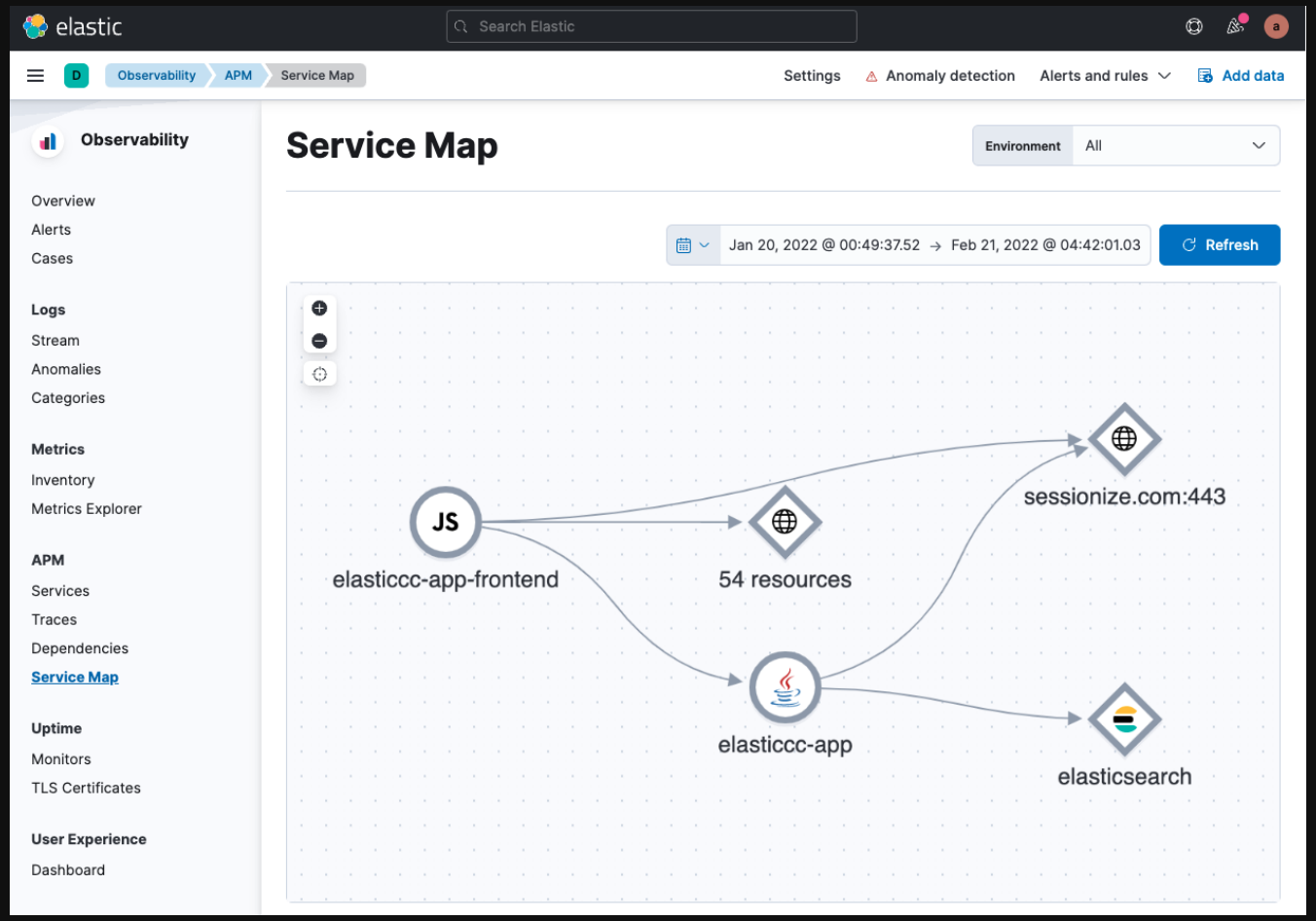
What was your goal in founding Elasticsearch? What is the reason for Elastic's success? Is there anything you would have done differently at Elastic? Is there any product specific feature you're particularly proud of? Is there a feature of the Elastic Stack that you think is underrated?

Full session

Shay Banon
Founder, CTO | Elastic

Simon Willnauer
Founder | Elastic

Architecture



How to build, run & maintain

- No other teams involved after initial setup
- Collective ownership within the team
- Well tested

… a modern java web application

- Javalin as a framework
- Latest Java version
- Latest GC (ZGC)
- pac4j for SAML based authorization
- Frontend for backend developers with htmx and hyperscript
- New Elasticsearch Java Client
- Elastic APM Agent

… with minimal resources

- Small pods
- Fast rollouts
- No one working full time on this
- No user accounts/passwords should be stored
- Easy rollout for everyone in the community team

… on K8s

- Utilizing company wide resources
- Rollout: `docker build && docker push && kubectl restart ...`
- `imagePullPolicy: Always`

Secrets with Vault

```yaml
apiVersion: vaultproject.io/v1
kind: SecretClaim
metadata:
  name: elasticcc-app
  namespace: community
spec:
  type: Opaque
  path: secret/k8s/elasticcc-app
  renew: 3600
```

Secrets with Vault

```yaml
apiVersion: apps/v1
kind: Deployment
spec:
  template:
    spec:
      containers:
        - name: elasticcc-app
          env:
          - name: ELASTICSEARCH_PASSWORD
            valueFrom:
              secretKeyRef:
                name: elasticcc-app
                key: elasticsearch_password
```

Secrets with Vault

```
vault write secret/k8s/elasticcc-app \
   elasticsearch_password=S3cr3t \
   key=value
```

Rollouts without downtime

```yaml
apiVersion: apps/v1
kind: Deployment
spec:
  replicas: 1
  strategy:
    rollingUpdate:
      maxUnavailable: 0
    type: RollingUpdate
```

Rollouts without downtime

- Just start more pods... **not so easy**
- Requests are distributed via round robin
- Javalin is a Servlet based web framework with a notion of sessions...
- ... each user gets a session cookie with a corresponding map of attributes on the server side
- Server side: `User user = ctx.sessionAttribute("user")`
- Instance shutdown kills session
- Session fixation? Works until shutdown...

Rollouts without downtime

```
this.app = Javalin.create(cfg -> {
  cfg.sessionHandler(() -> createSessionHandler(elasticsearchClient));
});

public static SessionHandler createSessionHandler(ElasticsearchClient client) {
  SessionHandler sessionHandler = new SessionHandler();
  // session handler setup here...

  SessionCache sessionCache = new NullSessionCache(sessionHandler);
  sessionCache.setSaveOnCreate(true);
  sessionCache.setFlushOnResponseCommit(true);
  sessionCache.setSessionDataStore(new ElasticsearchSessionDataStore(client));

  sessionHandler.setSessionCache(sessionCache);
  return sessionHandler;
}
```

Rollouts without downtime

- Every request writes its session data to Elasticsearch when finished
- Bad idea! The internet consists of bots... a lot
- 100k requests per hour **before** the announcement due to security scanners
- Solution: Only persist session if a login/logout has happened prior
- **Major** reduction of Elasticsearch write operations, resulting in faster responses

No announcement, but 100k req/hour?

```yaml
apiVersion: extensions/v1beta1
kind: Ingress
metadata:
  name: elasticcc-app-ngx
  namespace: community
  annotations:
    kubernetes.io/ingress.class: nginx
    cert-manager.io/cluster-issuer: letsencrypt-production
```

Probes

```yaml
livenessProbe:
  failureThreshold: 3
  periodSeconds: 30
  httpGet:
    path: /monitoring/health
    port: 8080
readinessProbe:
  failureThreshold: 15
  initialDelaySeconds: 10
  periodSeconds: 5
  httpGet:
    path: /monitoring/health
    port: 8080
```

Setting JVM memory

```yaml
resources:
  requests:
    cpu: 2.0
    memory: 1Gi
  limits:
    cpu: 2.0
    memory: 1Gi
```

```
def jvmOptions = ["-XX:+UseZGC", "-Xmx768m"]

startScripts {
  defaultJvmOpts = jvmOptions
}
```

Monitoring

```yaml
spec:
  template:
    metadata:
      annotations:
        watcher.alerts.slack: "#community-downtime-notifications"
      labels:
        app: elasticcc-app
        watcher: enabled
```

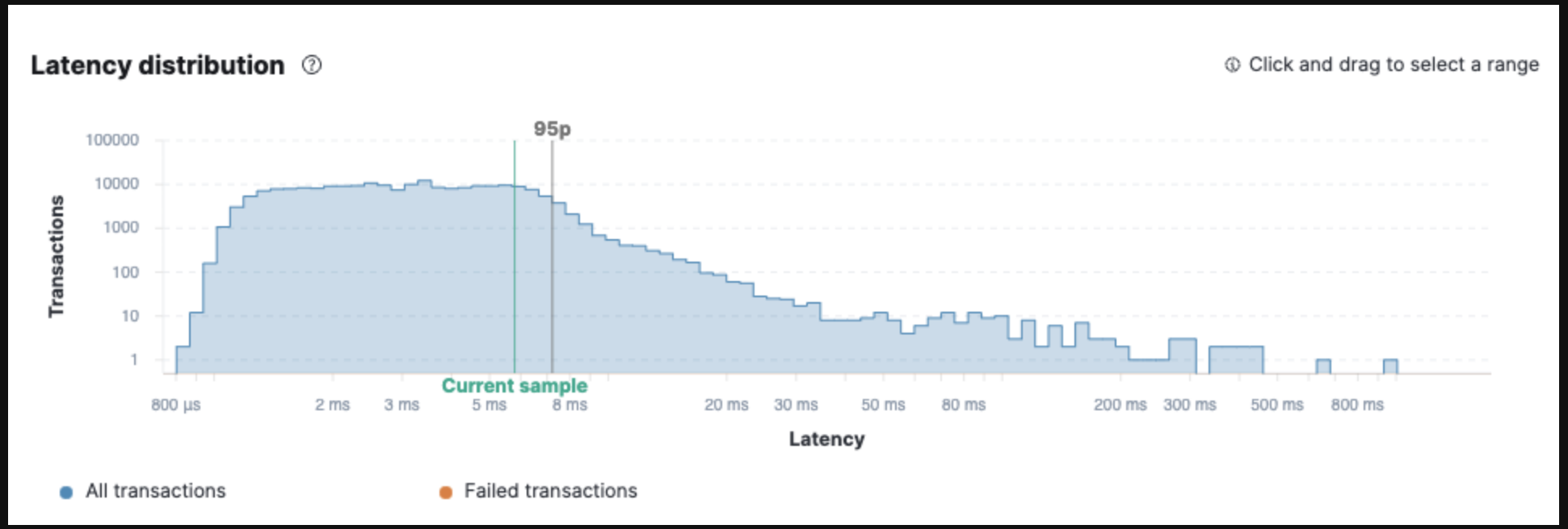
elastic-apps APP 05:05

community.elasticcc-app has 1 not ready pod(s) [ack] [docs]

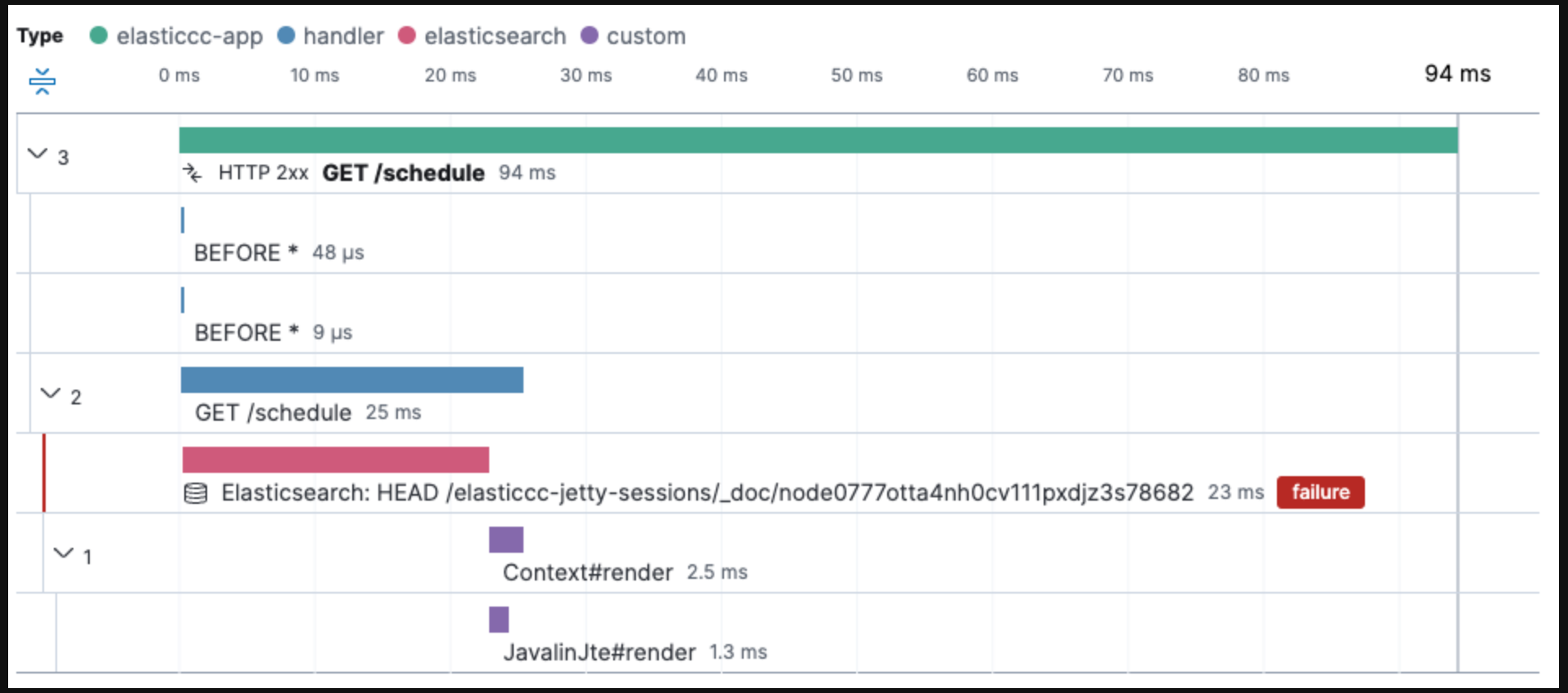
Observability

- Tradeoff
  - GraalVM for speed and lower memory footprint
  - APM agents require bytecode instrumentation

Observability



Observability



elastic

Search Elastic

Observability > APM > Services > elasticcc-app-front... > Transactions > /schedule

Settings | Anomaly detection | Alerts and rules | Add data

Observability

- Overview
- Alerts
- Cases

Logs

- Stream
- Anomalies
- Categories

Metrics

- Inventory
- Metrics Explorer

APM

- Services
- Traces
- Dependencies
- Service Map

Uptime

- Monitors
- TLS Certificates

User Experience

- Dashboard

Latency

All transactions | Failed transactions

Trace sample 2 of 500

Investigate | View full trace

20 days ago | 2,832 ms (100% of trace) | https://community-conference.elastic.co/schedule | Chrome (98.0.4758.80)

Timeline | **Metadata** | **Logs**

Services elasticcc-app-frontend elasticcc-app

0 ms | 500 ms | 1,000 ms | 1,500 ms | 2,000 ms | 2,500 ms | 2,832 ms

59 **/schedule** 2,832 ms

1 Requesting and receiving the document 3.0 ms

3 HTTP 2xx **GET /schedule** 5.5 ms

BEFORE * 20 µs

BEFORE * 6 µs

1 GET /schedule 3.2 ms

1 Context#render 2.9 ms

JavalinJte#render 1.9 ms

Parsing the document, executing sync. scripts 880 ms

https://community-conference.elastic.co/static/elastic-20211203.css 753 ms

https://cdnjs.cloudflare.com/ajax/libs/htmx/1.6.1/htmx.min.js 2.0 ms

https://community-conference.elastic.co/static/logo.svg 994 ms

https://community-conference.elastic.co/static/vidyard-v4.js 749 ms

Longtask(self) 98 ms

https://unpkg.com/hyperscript.org@0.9.4 2.0 ms

https://assets.website-files.com/5d35f5b2989a23dd99c4cb9a/5d35fd62752f2207ba611f84_Inter-SemiBol... 3.0 ms

https://assets.website-files.com/5d35f5b2989a23dd99c4cb9a/5d35fd62334923710184a551_Inter-Regula... 3.0 ms

https://assets.website-files.com/5d35f5b2989a23dd99c4cb9a/5f44209adcd5058fd1aa191d_MierB-ExtraB... 2.0 ms

Debugging

- Logs were not on the same instance, adding friction
- Logs required k8s configuration change in our case, tedious
- Component that shipped logs over the network would have been great
- Do you really need logs, when exceptions are logged? 🎑

Missing

- Automatic rollouts
- Stateful services outsourced
- Setup-as-code (i.e. via terraform to also include Elasticsearch cluster)
- APM tooling can be tricky, hard to distinguish single service memory spikes when running several pods

Conference day

- APM early detected an exception thrown when a template was rendered
- Rolled out before main traffic was coming in
- No issue during the 12 hours of the conference
- > 170k valid requests served in total, 1.7 mio in total
- 95th percentile:
  - `/schedule`: 8.8ms
  - `/speaker/{id}`: 5.0ms
  - `/session/{id}`: 5.5ms

Agility

- Log4Shell: From slack notification to assessing to rollout in 14 minutes
- Impact: Dropped the little one later to kindergarten

Summary

- 10/10 Would do again!
- Don't go crazy on automation (i.e. push on rollout etc)
- Go with Cookie based session store?
- Go crazy on IaC!
- Logs should be easily accessible, just like APM data
- Level of abstraction: 🎲

Summary: Level of abstraction

- Primitives are designed for operations (CPU, Memory)
- When to scale up/out? Application hint required:
  ❌ # of concurrent requests
  ❌ duration of requests
  ✅ wait time until processed
- Scaling strategy: Start pods if one is overloaded? Or all?
- Talk to developers about this, the discussions within your company (especially with legacy apps) will be a great exercise for everyone

Thanks for listening

Q & A

Alexander Reelsen
alex@elastic.co | @spinscale

Discussion

- `What technologies would you use?`
- `Where did I go wrong?`
- `Alex, this is not how you do it in k8s world!11!!elf!` - I'm sure, please talk to me 😀

Thanks for listening

Q & A

Alexander Reelsen
alex@elastic.co | @spinscale