
AWS
re:Invent

OPN301

Open-source observability at AWS

Michael Hausenblas

Sr. Open Source Product Developer Advocate

AWS Container Service Team

AWS

AWS re:Invent

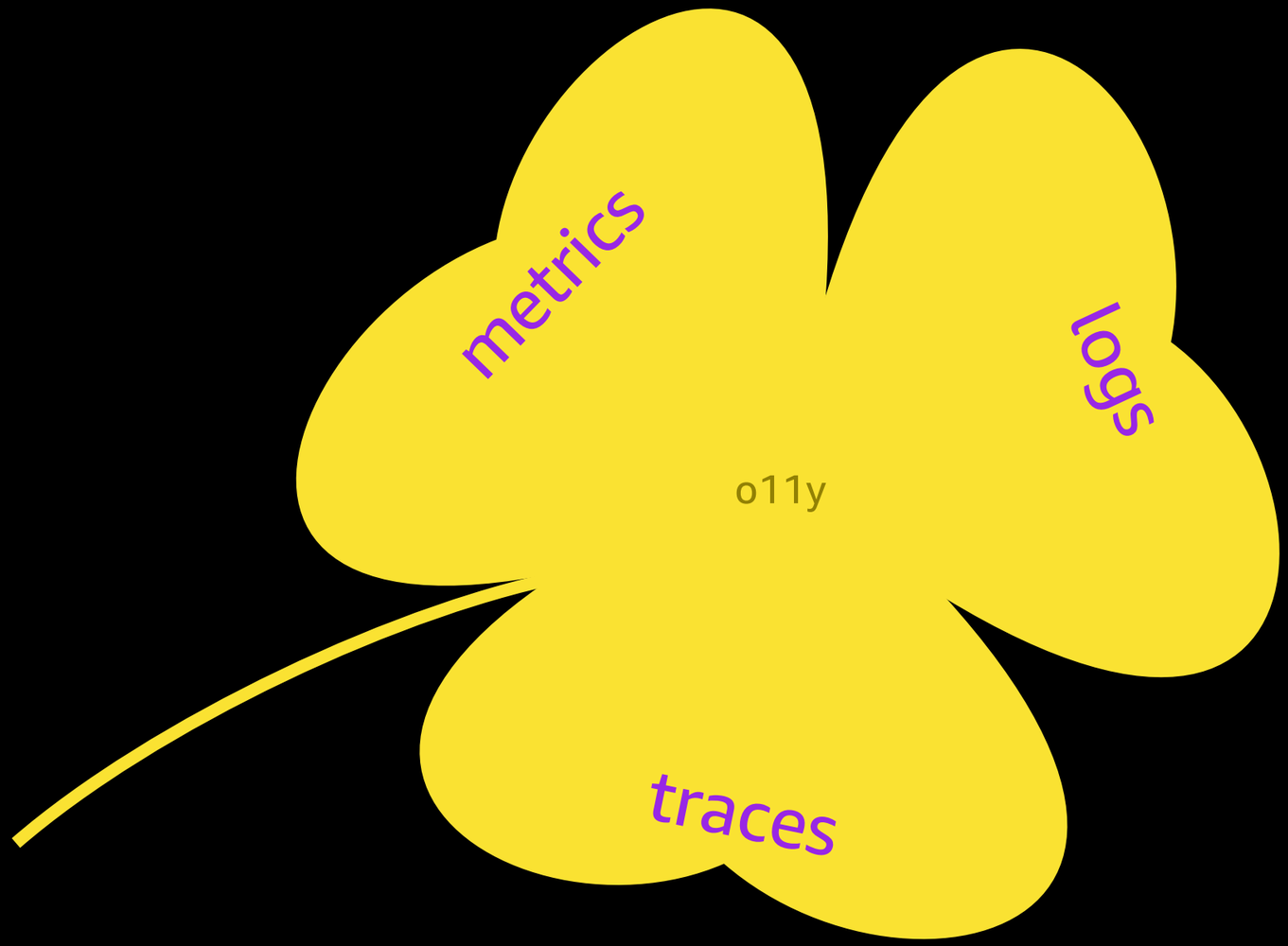
© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

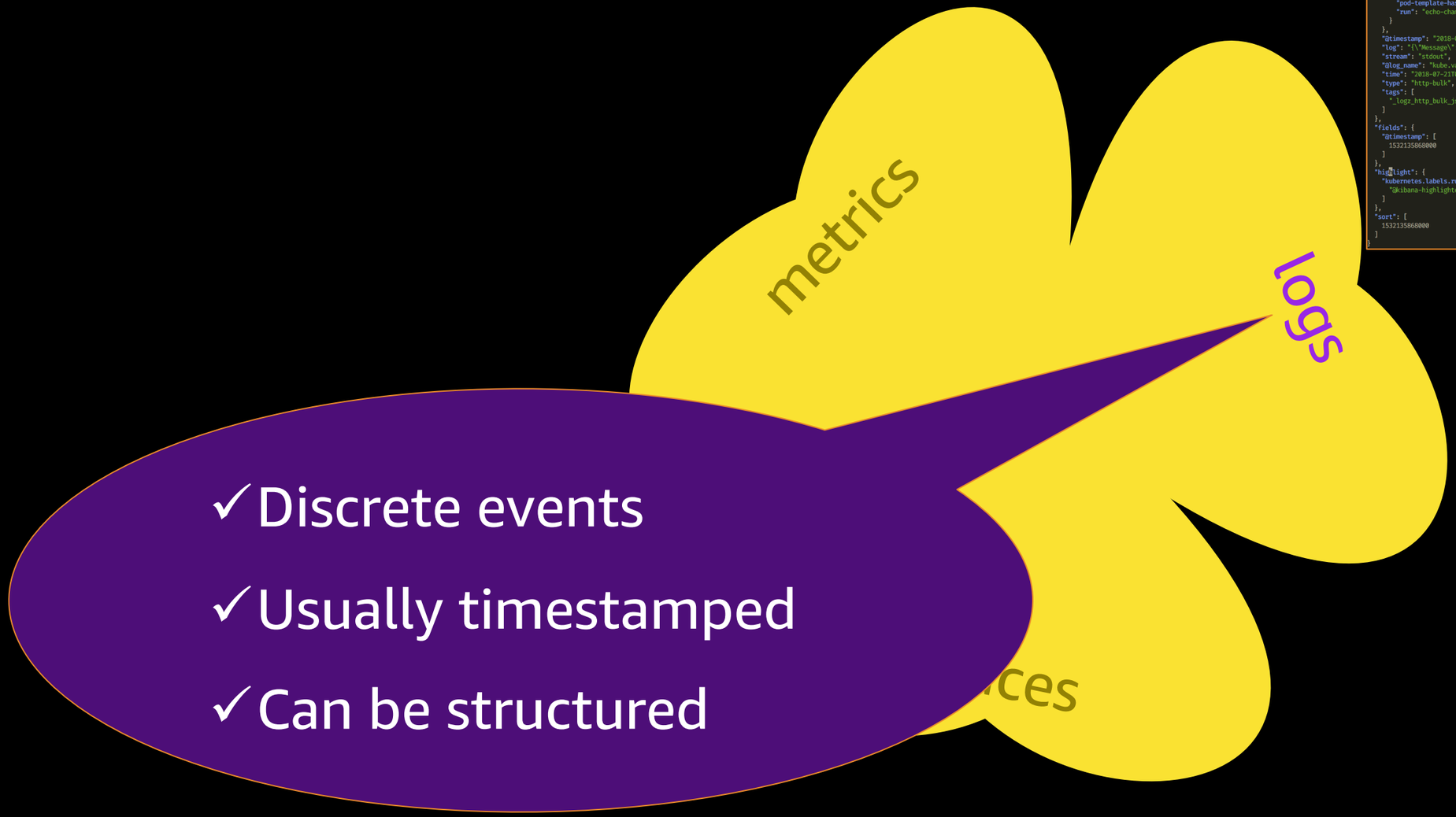
What is observability and why you *should* pay attention



AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.


metrics
logs
o11y
traces




```
{
"_index": "logzioCustomerIndex180721_v3",
"_type": "http-bulk",
"_id": "AWE6ag4I1yq4qwVVbz95.account-38207",
"_version": 1,
"_score":    ,
"_source": {
  "kubernetes": {
    "container_name": "echo-chamber",
    "host": "ip-10-41-24-84.ap-southeast-2.compute.internal",
    "annotations": {
      "kubernetes.io/limit-ranger": "LimitRanger plugin set: cpu request for container echo-chamber"
    },
    "docker_id": "d012be600cdb0f322203da4a4161da42e898cc8ce4a4dd5d79a1cad1a4d80537",
    "pod_id": "89f01ebb-8c82-11e8-b438-0a9480859cc",
    "pod_name": "echo-chamber-59bbd6495d-4n9x7",
    "namespace_name": "default",
    "labels": {
      "pod-template-hash": "1566820518",
      "run": "echo-chamber"
    }
  },
  "@timestamp": "2018-07-21T01:17:48.000+00:00",
  "log": "{\"Message\": \"JSON Parsing in Kubernetes is fun!\"}\n",
  "stream": "stdout",
  "@log_name": "kube.var.log.containers.echo-chamber-59bbd6495d-4n9x7_default_echo-chamber-d012be600cdb0f322203da4a4161da42e898cc8ce4a4dd5d79a1cad1a4d80537.log",
  "time": "2018-07-21T01:17:48.429526Z",
  "type": "http-bulk",
  "tags": [
    "_logz_http_bulk_json_8070"
  ]
},
"fields": {
  "@timestamp": [
    1532135868000
  ]
},
"highlight": {
  "kubernetes.labels.run": [
    "@kibana-highlighted-field@echo-chamber@/kibana-highlighted-field@"
  ]
},
"sort": [
  1532135868000
]
}
```



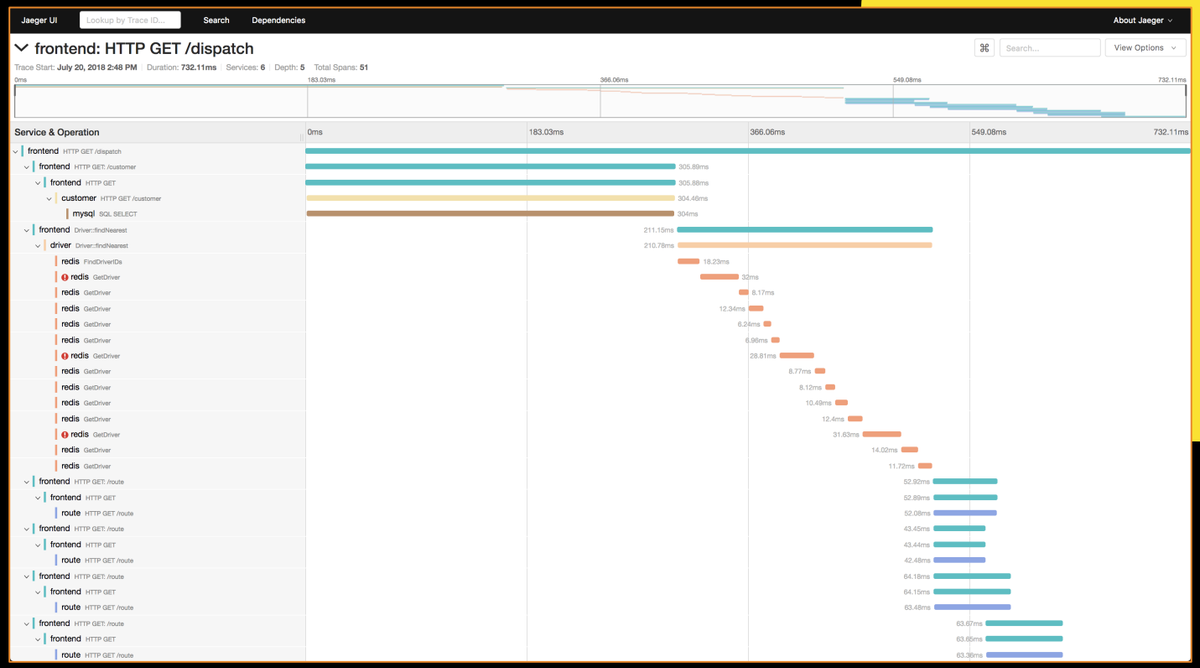
grafana.com/grafana/dashboards/1244

metrics

log

- ✓ *Regularly* sampled data points (timestamped)
- ✓ Numeric value
- ✓ Dimension/Labels

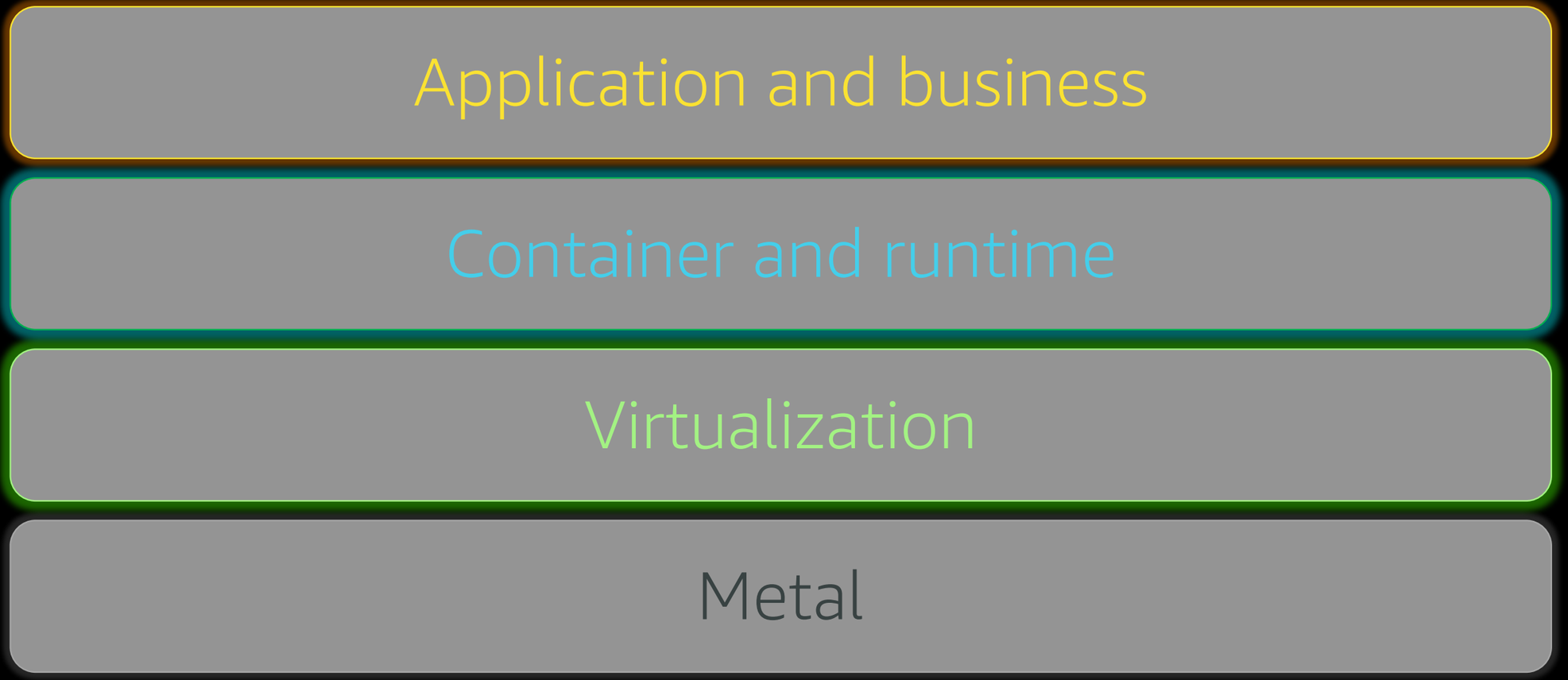
Request path across a number of services



traces

jaegertracing.io/docs/1.18/

Observability across the stack



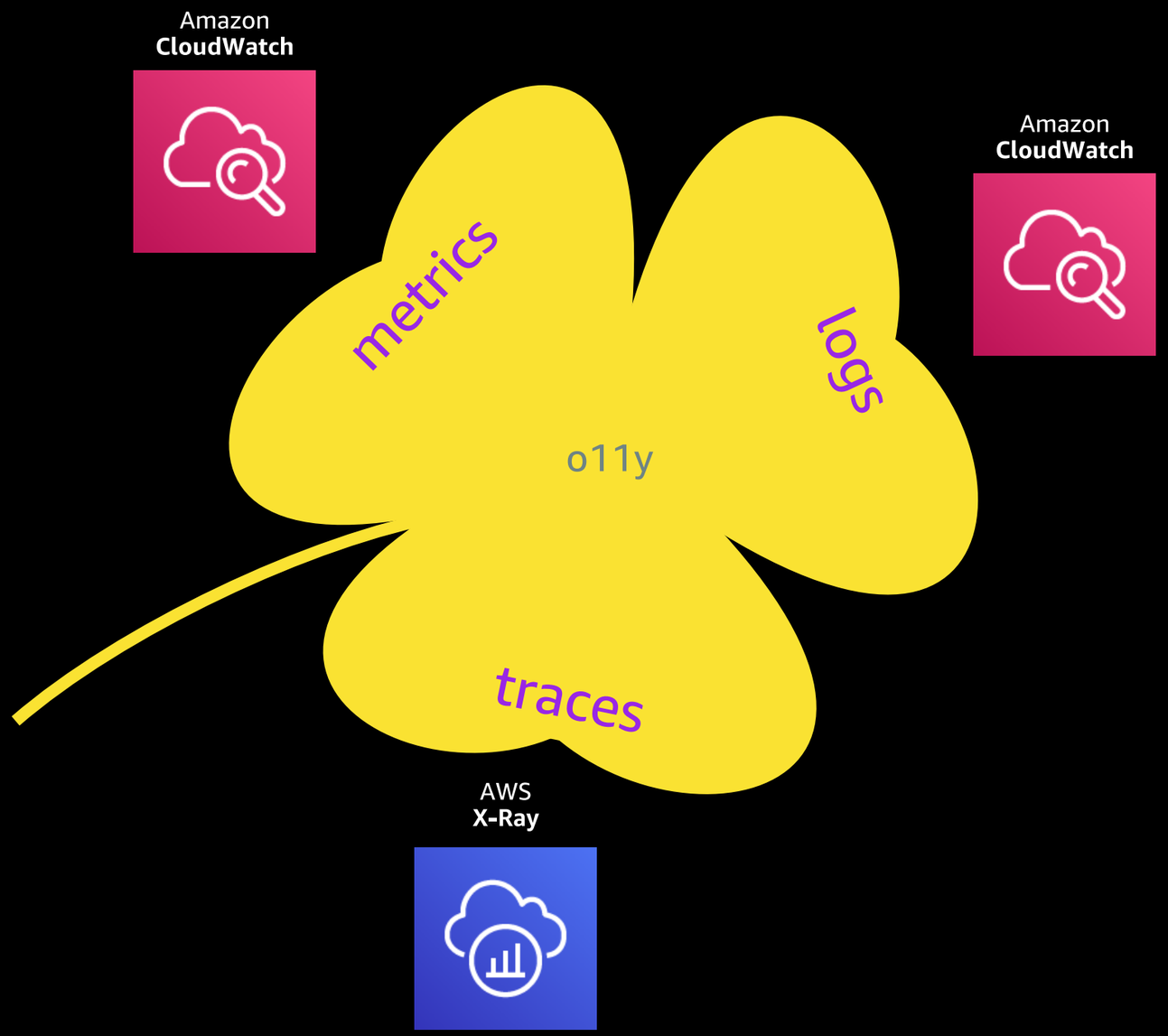
Observability at AWS



AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

How does this map to AWS services?



Some stats on our services



Amazon CloudWatch – 1,000 trillion metrics per months[1]



CloudWatch – 100 PB of logs per months[1]



AWS X-Ray – dozens of programming languages and integrations[2]

[1] twitter.com/AWSSecurityInfo/status/1202724331769470976

[2] docs.aws.amazon.com/xray/latest/devguide/aws-xray.html

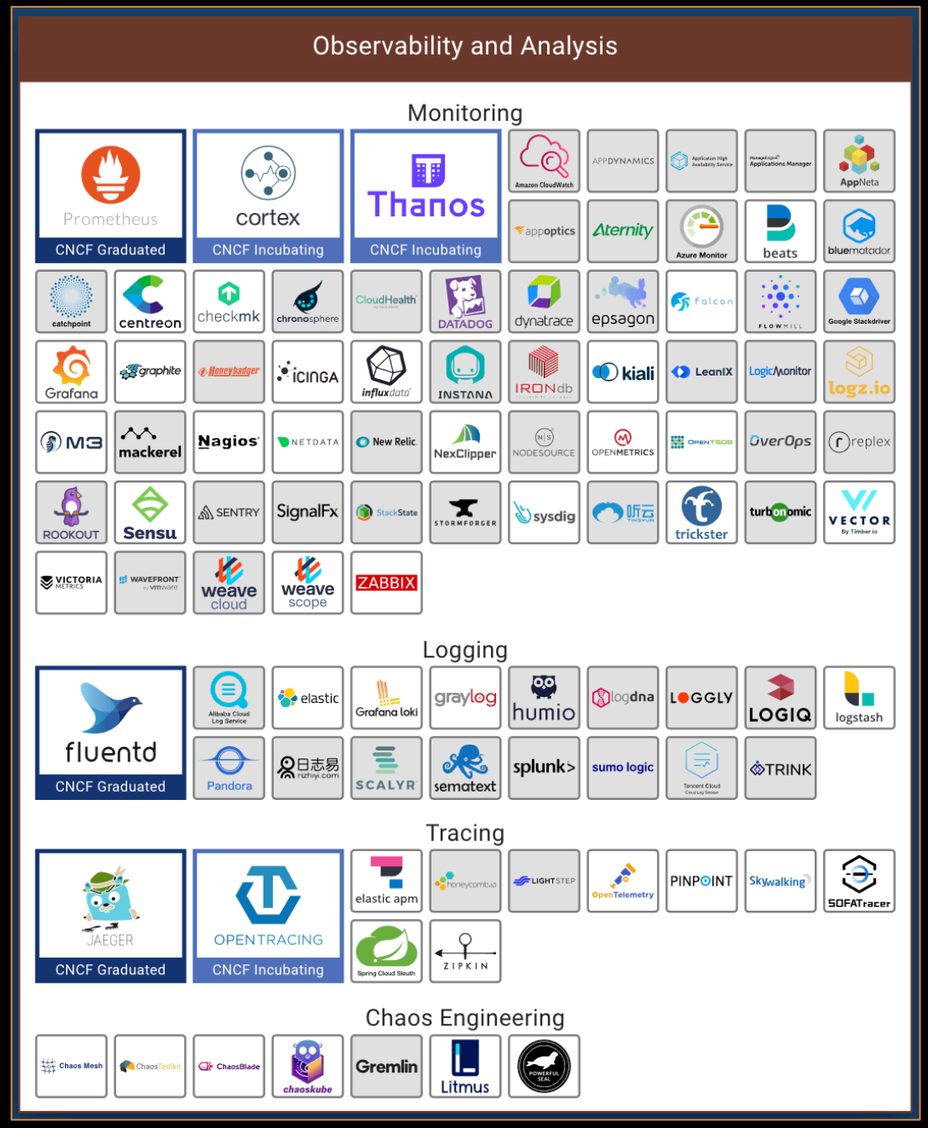
Observability at CNCF and wider open source community



AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

CNCF landscape around o11y



landscape.cncf.io

Observability activities at CNCF

- Prometheus
- Jaeger
- Fluentd

- Cortex
- Thanos

- OpenTelemetry
- OpenMetrics
- Chaos Mesh
- Litmus

Graduated

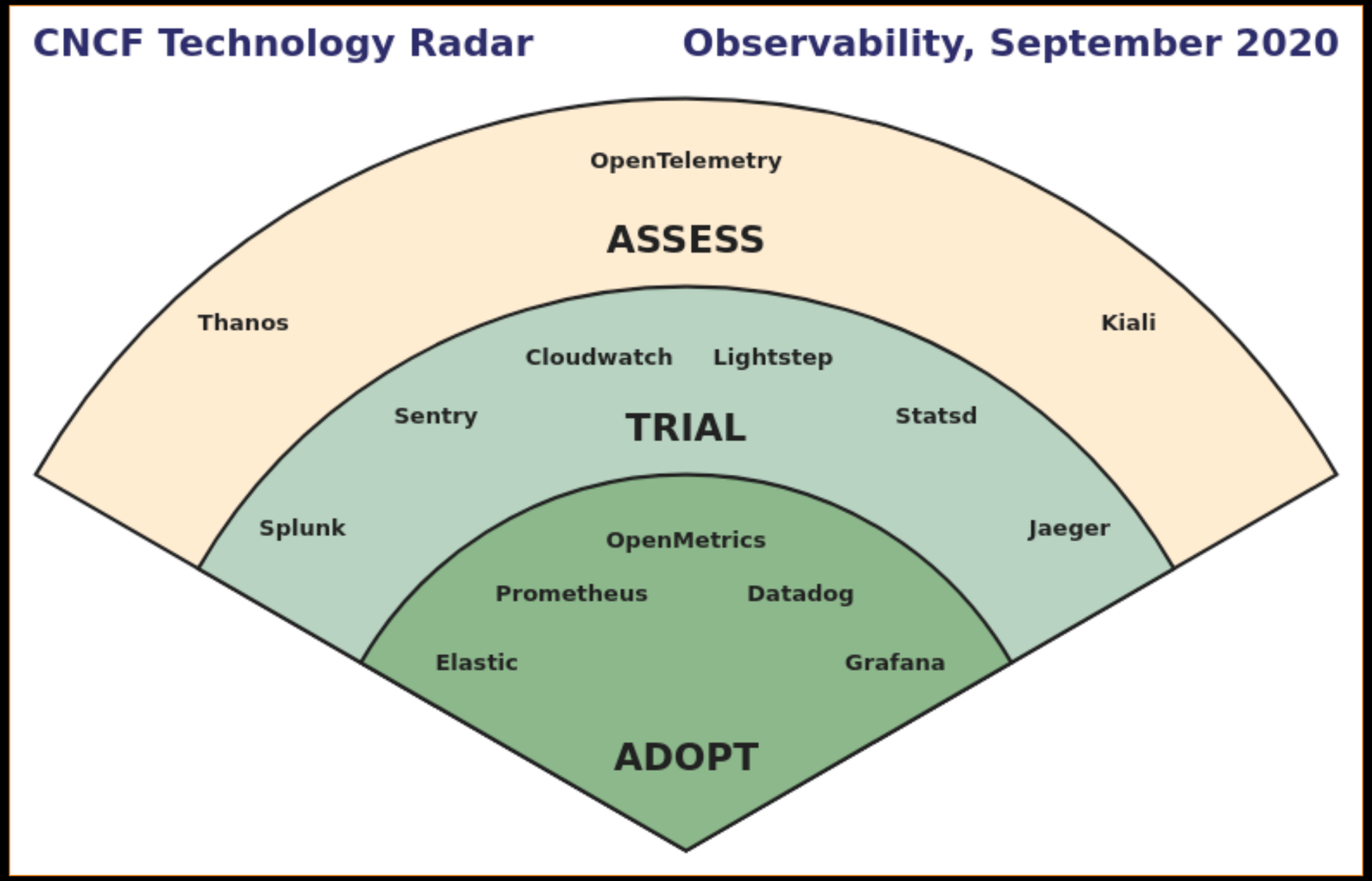
Incubating

Sandbox

SIG observability

github.com/cncf/toc/blob/master/process/graduation_criteria.adoc

CNCF end-user technology radar



cncf.io/blog/2020/09/11/cncf-end-user-technology-radar-observability-september-2020/

AWS upstream contributions



AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Some examples:

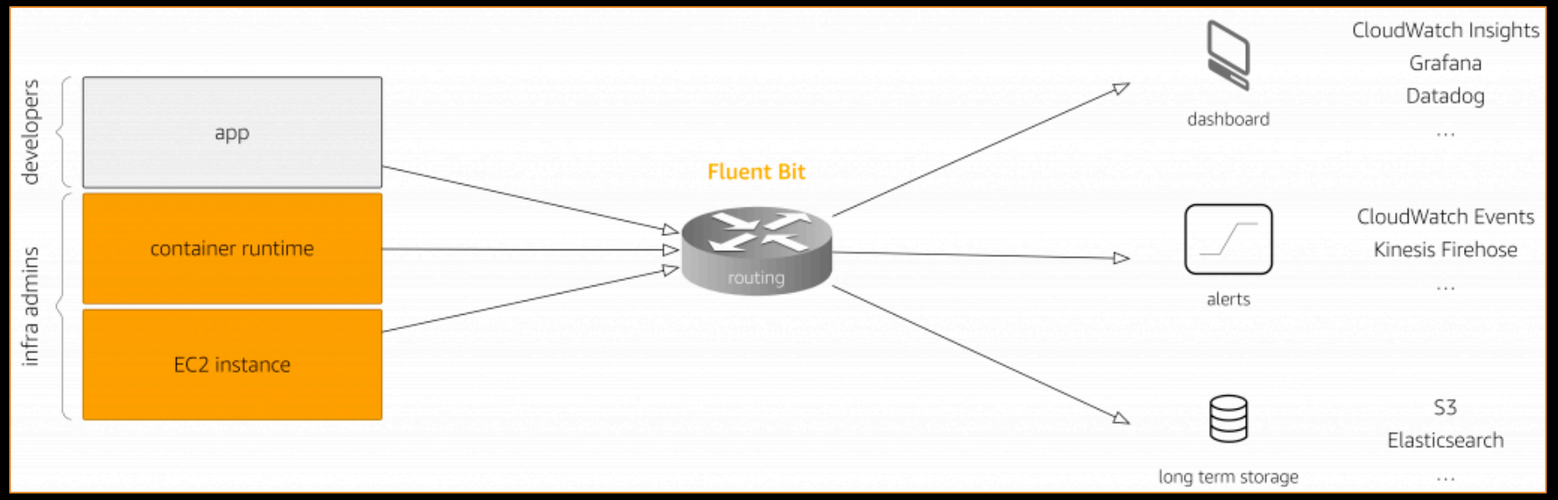
1) Logs – FireLens (Fluent Bit) log routing across orchestrators

2) Metrics – Using container insights for Prometheus metrics

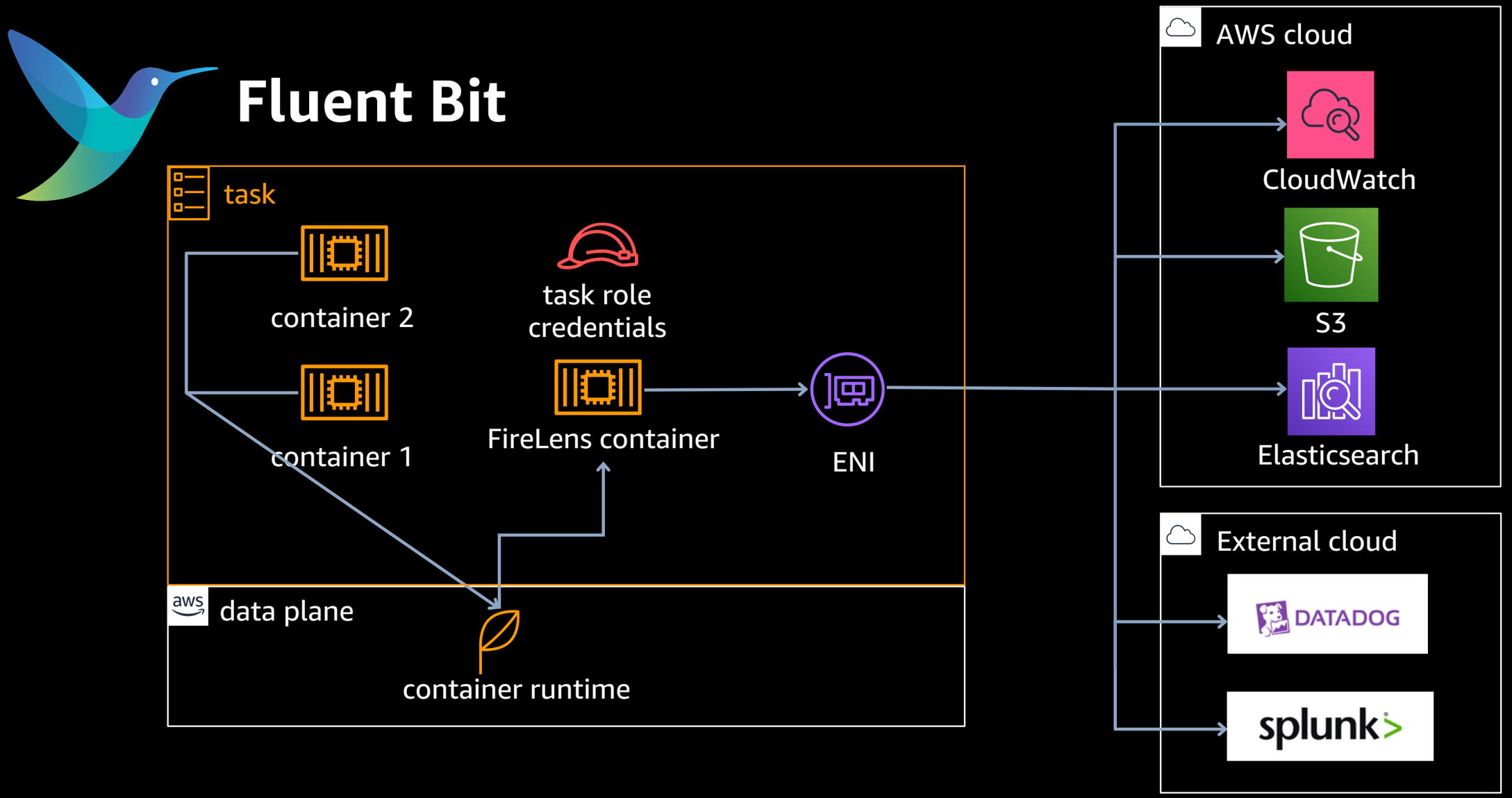
3) Traces – OpenTelemetry C++ library contributions

EX1) FireLens: OSS log routing

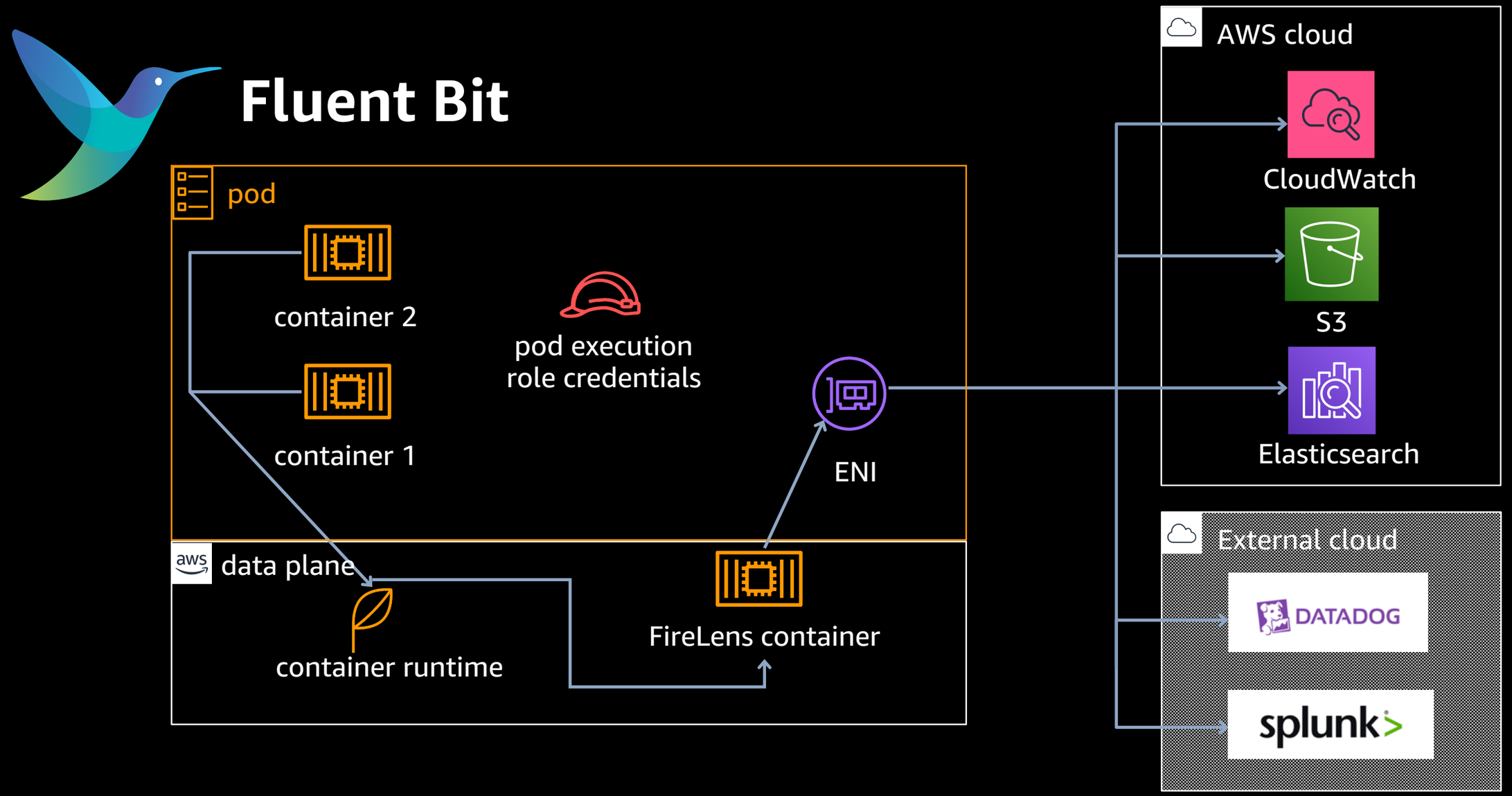
- Logs – FireLens (Fluent Bit) log routing across orchestrators
- https://aws.amazon.com/blogs/opensource/centralized-container-logging-fluent-bit/
- https://aws.amazon.com/blogs/containers/how-to-capture-application-logs-when-using-amazon-eks-on-aws-fargate/
- https://aws.amazon.com/blogs/aws/announcing-firelens-a-new-way-to-manage-container-logs/



EX1) FireLens for ECS

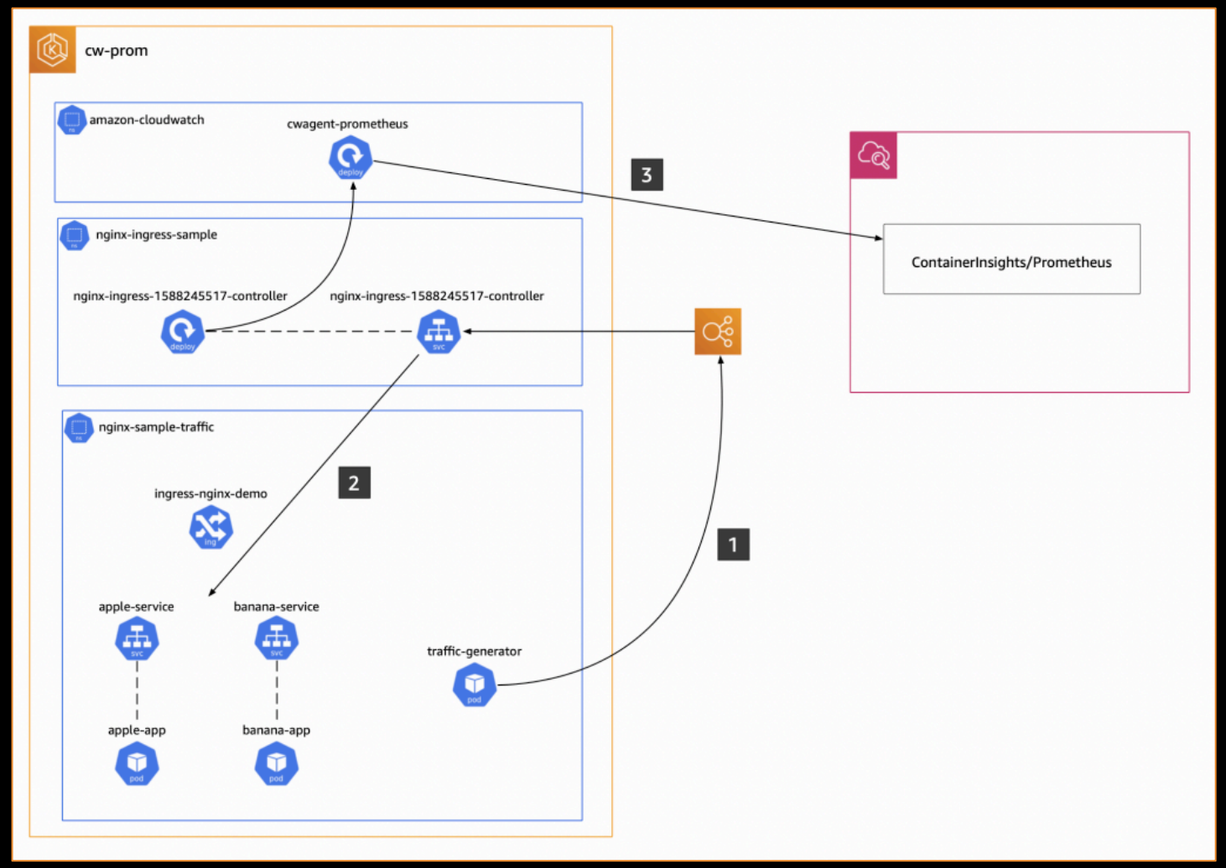


EX1) FireLens for EKS



EX2) Prometheus metrics in CloudWatch

- Metrics – using container insights for Prometheus metrics
  https://aws.amazon.com/blogs/containers/using-prometheus-metrics-in-amazon-cloudwatch/



Prometheus / Grafana / CloudWatch in action

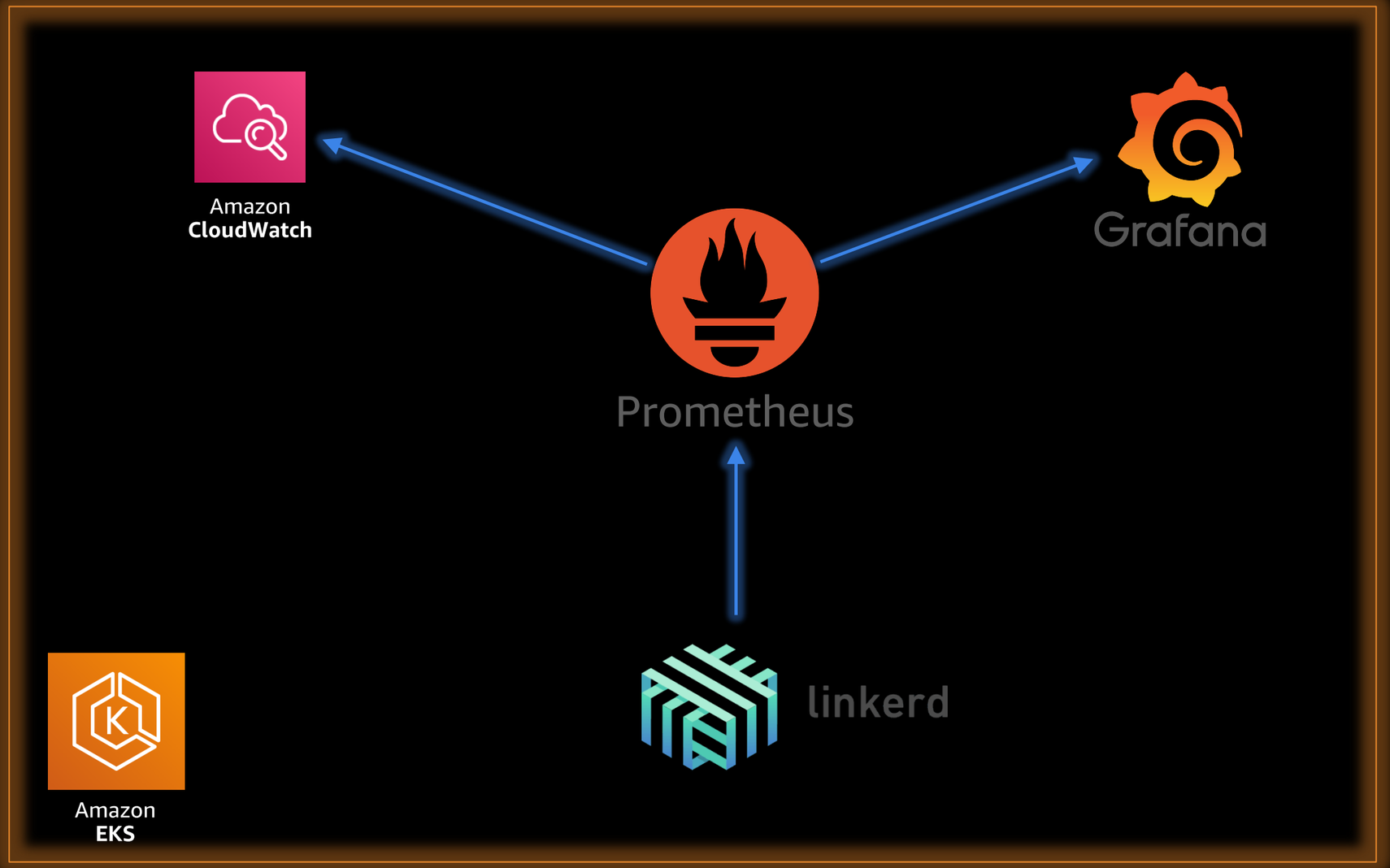
```yaml
apiVersion: eksctl.io/v1alpha5
kind: ClusterConfig
metadata:
  name: oss-o11y-demo
  region: eu-west-1
  version: '1.17'
iam:
  withOIDC: true
managedNodeGroups:
- name: default-ng
  minSize: 1
  maxSize: 2
  desiredCapacity: 2
  ssh:
    allow: true
    publicKeyPath: ~/.ssh/work-default.pub
  labels: {role: mngworker}
  iam:
    withAddonPolicies:
      imageBuilder: true
      autoScaler: true
      externalDNS: true
      certManager: true
      ebs: true
      albIngress: true
      xRay: true
      cloudWatch: true
cloudWatch:
  clusterLogging:
    enableTypes: ["*"]
```

```yaml
apiVersion: extensions/v1beta1
kind: Ingress
metadata:
  name: ingress-nginx-demo
  namespace: {{namespace}}
spec:
  rules:
  - host: {{external_ip}}
    http:
      paths:
        - path: /apple
          backend:
            serviceName: apple-service
            servicePort: 5678
        - path: /banana
          backend:
            serviceName: banana-service
            servicePort: 5678

---

apiVersion: v1
kind: Pod
metadata:
        name: traffic-generator
        namespace: {{namespace}}
spec:
        containers:
                - name: traffic-generator
                  image: ellerbrock/alpine-bash-curl-ssl
                  command: ["/bin/bash"]
                  args: ["-c", "while :; do curl http://{{external_ip}}/apple > /dev/null 2>&1; curl http://{{external_ip}}/banana > /dev/null 2>&1; sleep 1; done"]
```

Prometheus / Grafana / CloudWatch in action



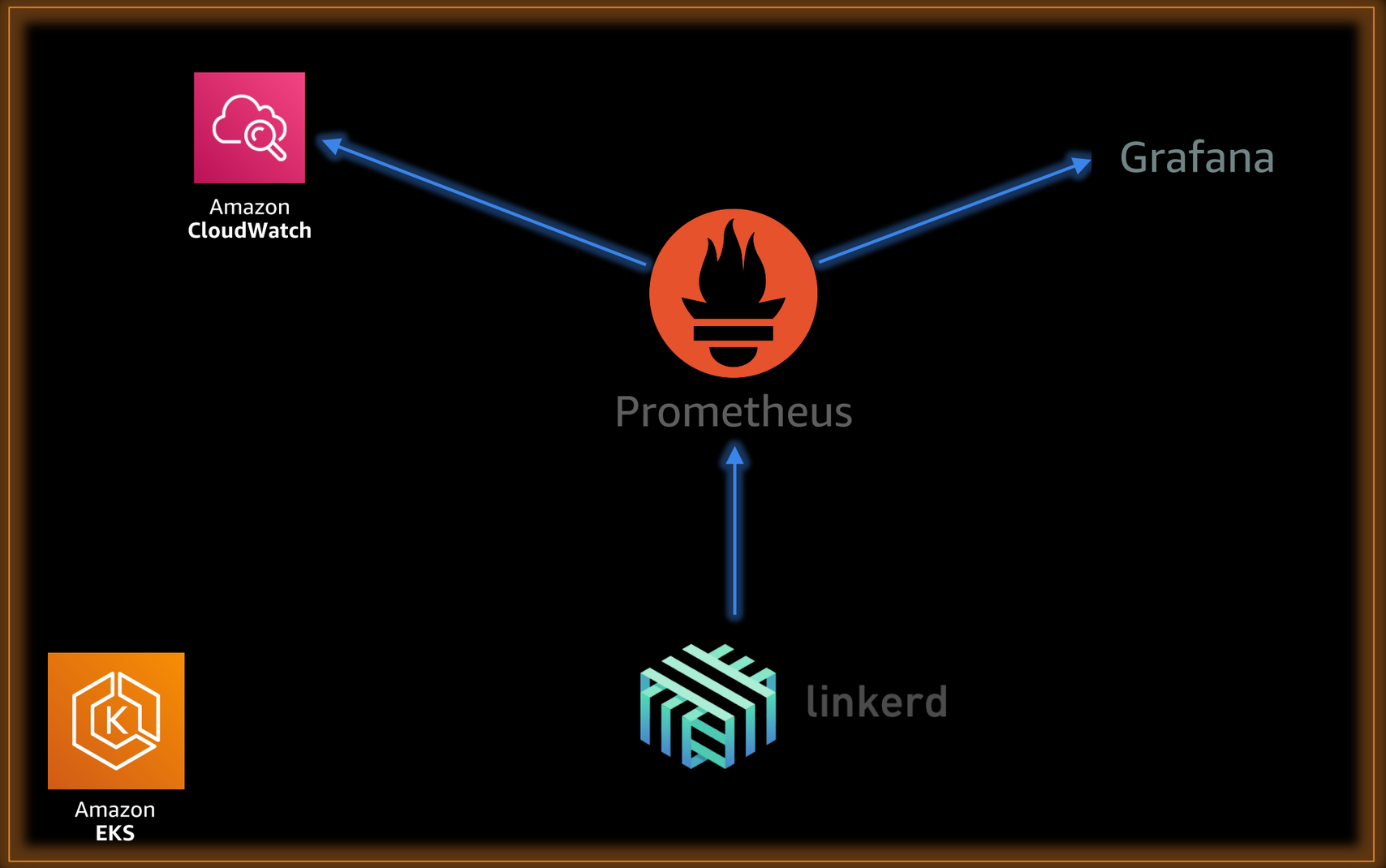
Prometheus / Grafana / CloudWatch in action

```
└─> $ kubectl get pods -A -o=wide
NAMESPACE              NAME                                                READY   STATUS    RESTARTS   AGE   IP              NODE                                                NOMINATED NODE   READINESS GATES
amazon-cloudwatch      cwagent-prometheus-694d8b9c86-7v5dt                 1/1     Running   0          45h   192.168.81.144  ip-192-168-88-186.eu-west-1.compute.internal   <none>           <none>
kube-system            aws-node-74r8t                                      1/1     Running   0          46h   192.168.88.186  ip-192-168-88-186.eu-west-1.compute.internal   <none>           <none>
kube-system            aws-node-xhkbw                                      1/1     Running   0          46h   192.168.40.166  ip-192-168-40-166.eu-west-1.compute.internal   <none>           <none>
kube-system            coredns-6987776bbd-7nbhp                            1/1     Running   0          46h   192.168.43.158  ip-192-168-40-166.eu-west-1.compute.internal   <none>           <none>
kube-system            coredns-6987776bbd-r8whq                            1/1     Running   0          46h   192.168.48.236  ip-192-168-40-166.eu-west-1.compute.internal   <none>           <none>
kube-system            kube-proxy-7h8z2                                    1/1     Running   0          46h   192.168.88.186  ip-192-168-88-186.eu-west-1.compute.internal   <none>           <none>
kube-system            kube-proxy-rnsmq                                    1/1     Running   0          46h   192.168.40.166  ip-192-168-40-166.eu-west-1.compute.internal   <none>           <none>
linkerd                linkerd-controller-f4ff89488-ftlf7                  2/2     Running   0          46h   192.168.78.202  ip-192-168-88-186.eu-west-1.compute.internal   <none>           <none>
linkerd                linkerd-destination-c69cccb69-xzl72                 2/2     Running   0          46h   192.168.48.168  ip-192-168-40-166.eu-west-1.compute.internal   <none>           <none>
linkerd                linkerd-grafana-6b56bdb978-nvv7h                    2/2     Running   0          46h   192.168.56.193  ip-192-168-40-166.eu-west-1.compute.internal   <none>           <none>
linkerd                linkerd-identity-6cdd87f69b-vxv5s                   2/2     Running   0          46h   192.168.74.108  ip-192-168-88-186.eu-west-1.compute.internal   <none>           <none>
linkerd                linkerd-prometheus-64b588d7f8-4z49b                 2/2     Running   0          46h   192.168.56.220  ip-192-168-40-166.eu-west-1.compute.internal   <none>           <none>
linkerd                linkerd-proxy-injector-567b47656f-6smfd             2/2     Running   0          46h   192.168.74.253  ip-192-168-88-186.eu-west-1.compute.internal   <none>           <none>
linkerd                linkerd-sp-validator-879d84658-rl8r8                2/2     Running   0          46h   192.168.60.240  ip-192-168-40-166.eu-west-1.compute.internal   <none>           <none>
linkerd                linkerd-tap-7c9947cf6c-29244                        2/2     Running   0          46h   192.168.66.81   ip-192-168-88-186.eu-west-1.compute.internal   <none>           <none>
linkerd                linkerd-web-5bd844f96c-fbfbf                        2/2     Running   0          46h   192.168.73.38   ip-192-168-88-186.eu-west-1.compute.internal   <none>           <none>
nginx-ingress-sample   nginx-ingress-1601464685-controller-748dbc4b79-rsj64   1/1     Running   0          45h   192.168.43.178  ip-192-168-40-166.eu-west-1.compute.internal   <none>           <none>
nginx-ingress-sample   nginx-ingress-1601464685-default-backend-858bdcb94d-47vcr   1/1     Running   0          45h   192.168.54.230  ip-192-168-40-166.eu-west-1.compute.internal   <none>           <none>
nginx-sample-traffic   apple-app                                           1/1     Running   0          45h   192.168.94.192  ip-192-168-88-186.eu-west-1.compute.internal   <none>           <none>
nginx-sample-traffic   banana-app                                          1/1     Running   0          45h   192.168.62.147  ip-192-168-40-166.eu-west-1.compute.internal   <none>           <none>
nginx-sample-traffic   traffic-generator                                   1/1     Running   0          45h   192.168.64.113  ip-192-168-88-186.eu-west-1.compute.internal   <none>           <none>
```

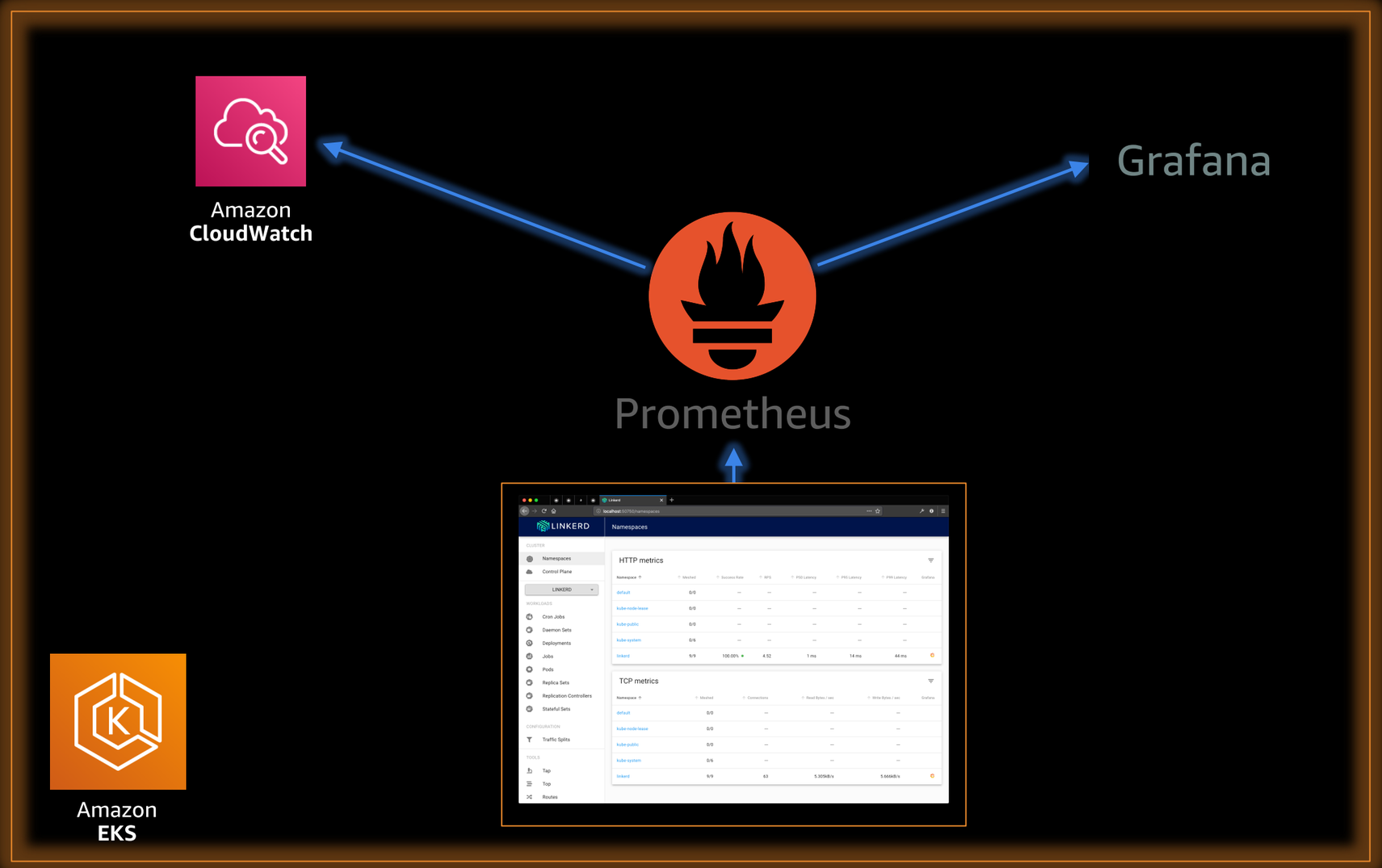
Prometheus / Grafana / CloudWatch in action

```
└─> $ aws logs get-log-events --log-group-name /aws/containerinsights/oss-o11y-demo/prometheus --log-stream-name kubernetes-service-endpoints --limit 5 | jq .
{
  "events": [
    {
      "timestamp": 1601629769880,
      "message": "{\"ClusterName\":\"oss-o11y-demo\",\"Namespace\":\"kube-system\",\"Service\":\"kube-dns\",\"Timestamp\":\"1601629769875\",\"Version\":\"0\",\"container_name\":\"coredns\",\"coredns_dns_request_type_c
mponent\":\"kube-dns\",\"instance\":\"192.168.43.158:9153\",\"job\":\"kubernetes-service-endpoints\",\"k8s_app\":\"kube-dns\",\"kubernetes_io_cluster_service\":\"true\",\"kubernetes_io_name\":\"CoreDNS\",\"kubernetes_
mpute.internal\",\"pod_name\":\"coredns-6987776bbd-7nbhp\",\"prom_metric_type\":\"counter\",\"server\":\"dns://:53\",\"type\":\"AAAA\",\"zone\":\".\"}",
      "ingestionTime": 1601629776291
    },
    {
      "timestamp": 1601629769880,
      "message": "{\"ClusterName\":\"oss-o11y-demo\",\"Namespace\":\"kube-system\",\"Service\":\"kube-dns\",\"Timestamp\":\"1601629769875\",\"Version\":\"0\",\"container_name\":\"coredns\",\"coredns_cache_size\":14,\"
s\",\"instance\":\"192.168.43.158:9153\",\"job\":\"kubernetes-service-endpoints\",\"k8s_app\":\"kube-dns\",\"kubernetes_io_cluster_service\":\"true\",\"kubernetes_io_name\":\"CoreDNS\",\"kubernetes_node\":\"ip-192-168
pod_name\":\"coredns-6987776bbd-7nbhp\",\"prom_metric_type\":\"gauge\",\"server\":\"dns://:53\",\"type\":\"success\"}",
      "ingestionTime": 1601629776291
    },
    {
      "timestamp": 1601629769880,
      "message": "{\"ClusterName\":\"oss-o11y-demo\",\"Namespace\":\"kube-system\",\"Service\":\"kube-dns\",\"Timestamp\":\"1601629769875\",\"Version\":\"0\",\"container_name\":\"coredns\",\"coredns_plugin_enabled\":1
-dns\",\"instance\":\"192.168.43.158:9153\",\"job\":\"kubernetes-service-endpoints\",\"k8s_app\":\"kube-dns\",\"kubernetes_io_cluster_service\":\"true\",\"kubernetes_io_name\":\"CoreDNS\",\"kubernetes_node\":\"ip-192-
,\"name\":\"kubernetes\",\"pod_name\":\"coredns-6987776bbd-7nbhp\",\"prom_metric_type\":\"gauge\",\"server\":\"dns://:53\",\"zone\":\".\"}",
      "ingestionTime": 1601629776291
    },
    {
      "timestamp": 1601629769880,
      "message": "{\"ClusterName\":\"oss-o11y-demo\",\"Namespace\":\"kube-system\",\"Service\":\"kube-dns\",\"Timestamp\":\"1601629769875\",\"Version\":\"0\",\"container_name\":\"coredns\",\"eks_amazonaws_com_componer
ce\":\"192.168.43.158:9153\",\"job\":\"kubernetes-service-endpoints\",\"k8s_app\":\"kube-dns\",\"kubernetes_io_cluster_service\":\"true\",\"kubernetes_io_name\":\"CoreDNS\",\"kubernetes_node\":\"ip-192-168-40-166.eu-w
"coredns-6987776bbd-7nbhp\",\"prom_metric_type\":\"gauge\",\"version\":\"go1.13.12\"}",
      "ingestionTime": 1601629776291
    },
    {
      "timestamp": 1601629805511,
      "message": "{\"ClusterName\":\"oss-o11y-demo\",\"Namespace\":\"kube-system\",\"Service\":\"kube-dns\",\"Timestamp\":\"1601629805506\",\"Version\":\"0\",\"container_name\":\"coredns\",\"coredns_cache_hits_total\"
ube-dns\",\"instance\":\"192.168.48.236:9153\",\"job\":\"kubernetes-service-endpoints\",\"k8s_app\":\"kube-dns\",\"kubernetes_io_cluster_service\":\"true\",\"kubernetes_io_name\":\"CoreDNS\",\"kubernetes_node\":\"ip-1
l\",\"pod_name\":\"coredns-6987776bbd-r8whq\",\"prom_metric_type\":\"counter\",\"server\":\"dns://:53\",\"type\":\"success\"}",
      "ingestionTime": 1601629806292
    }
  ],
  "nextForwardToken": "f/35717538195072969027091739909172639958359905420840468480",
  "nextBackwardToken": "b/35717537400475116858247106716844700882910903672239030292"
}
```

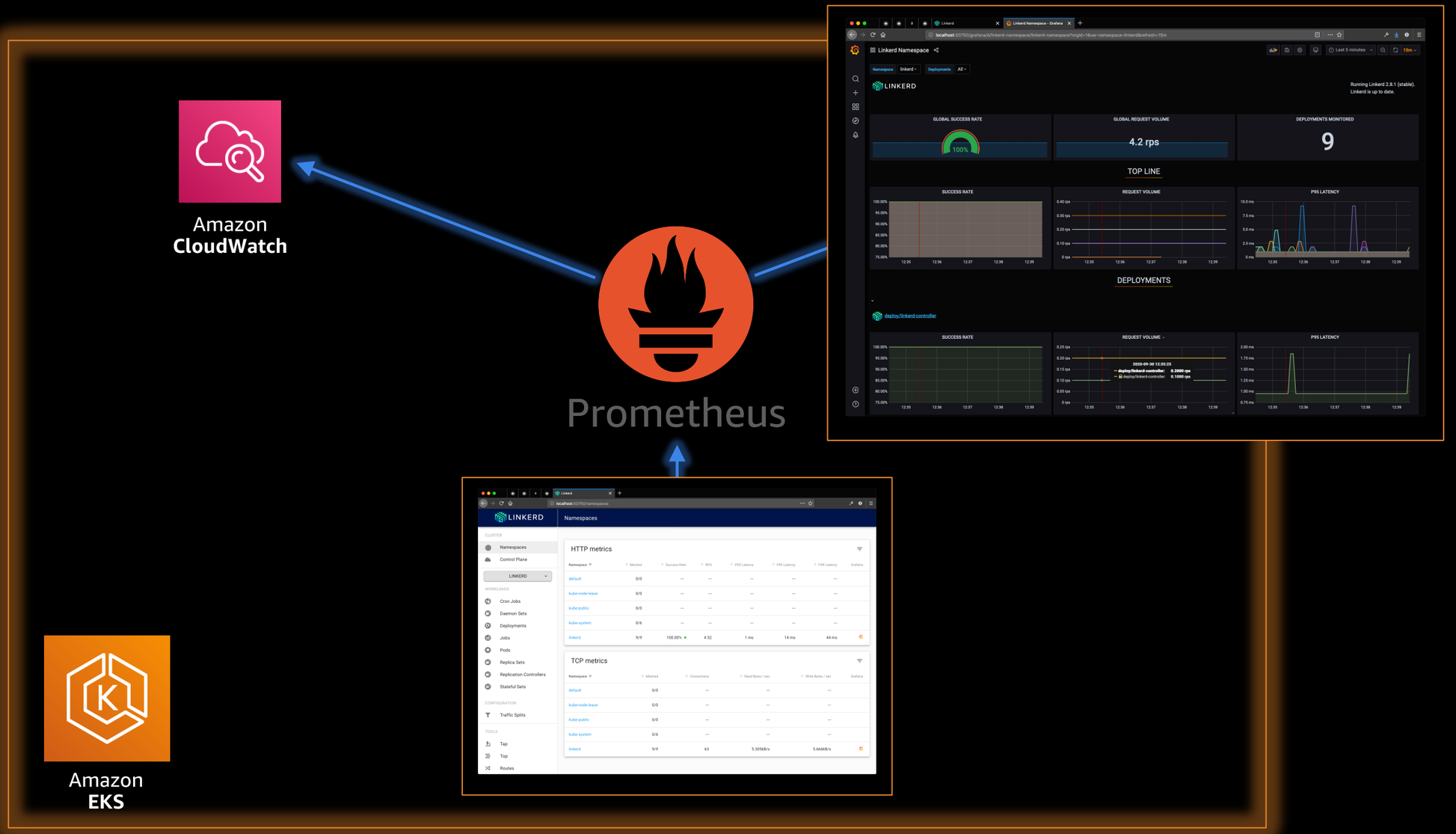
Prometheus / Grafana / CloudWatch in action



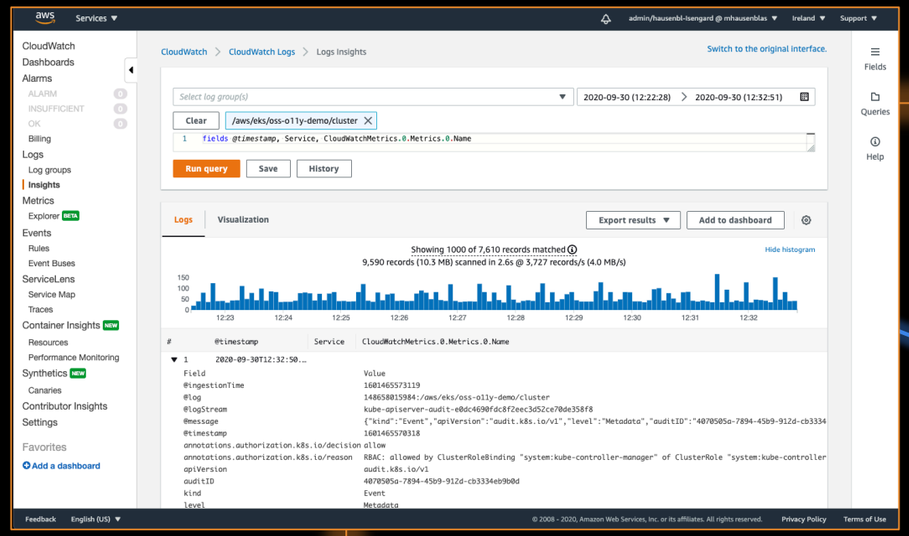
Prometheus / Grafana / CloudWatch in action

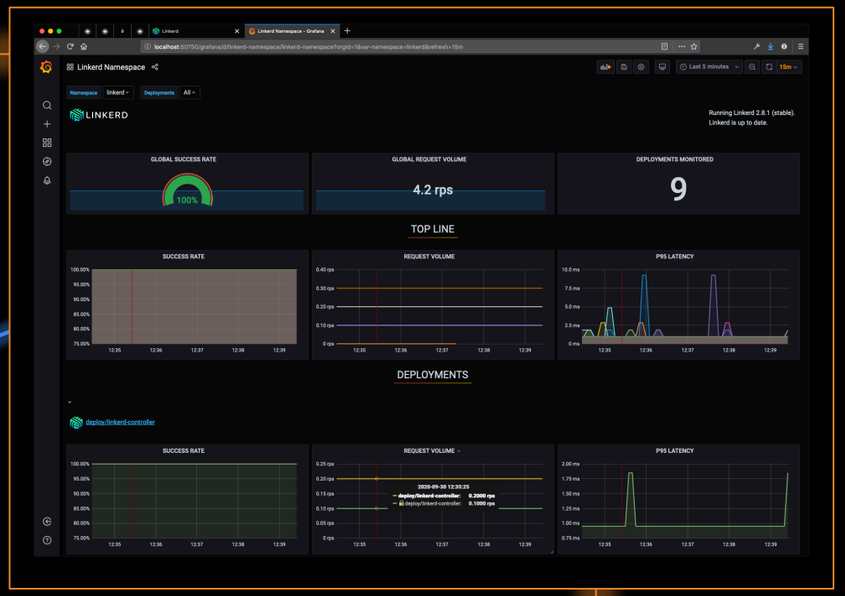


Prometheus / Grafana / CloudWatch in action

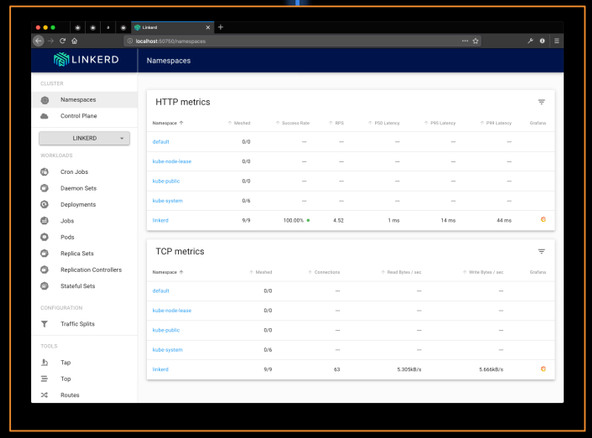


Prometheus / Grafana / CloudWatch in action



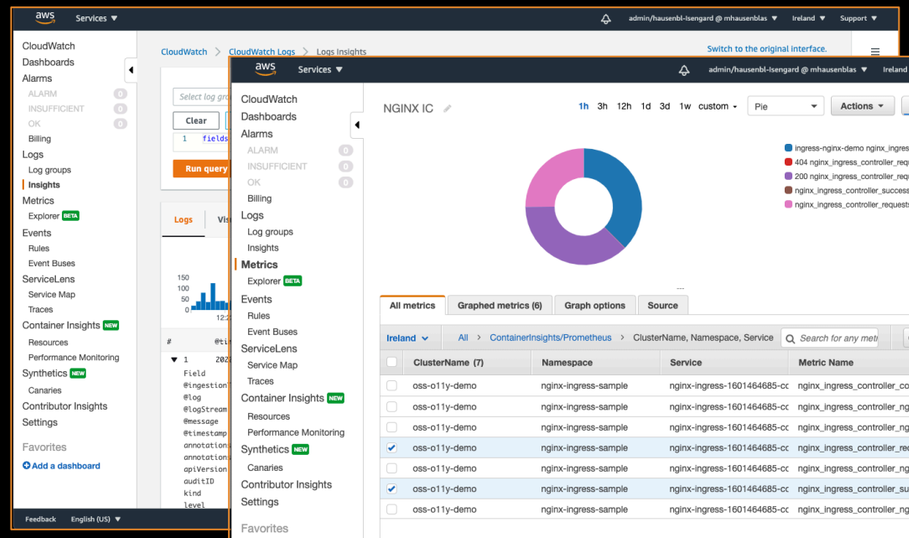


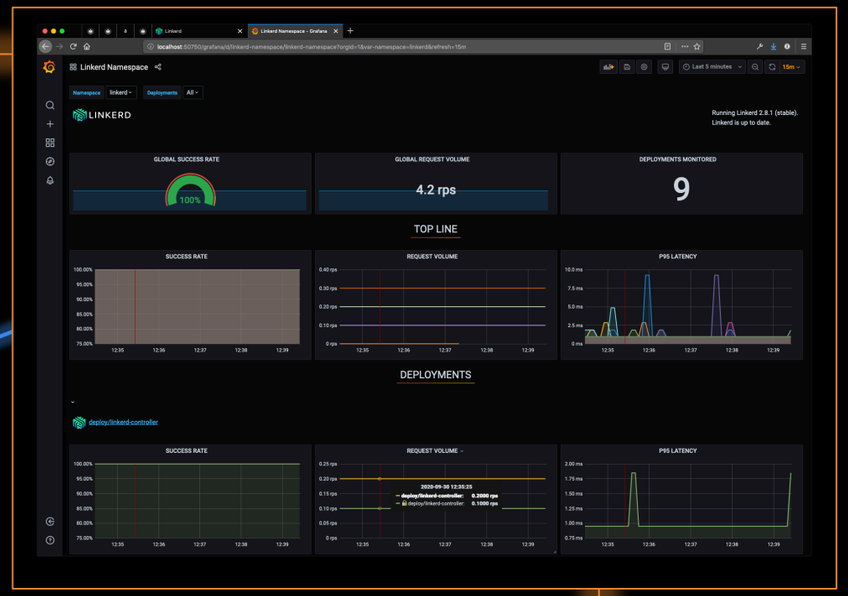
Prometheus



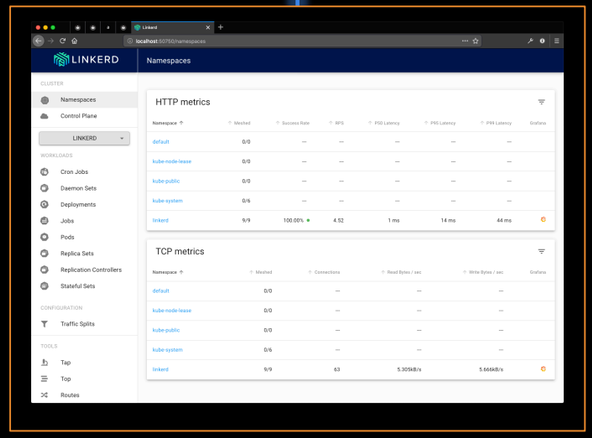
Amazon
EKS

Prometheus / Grafana / CloudWatch in action





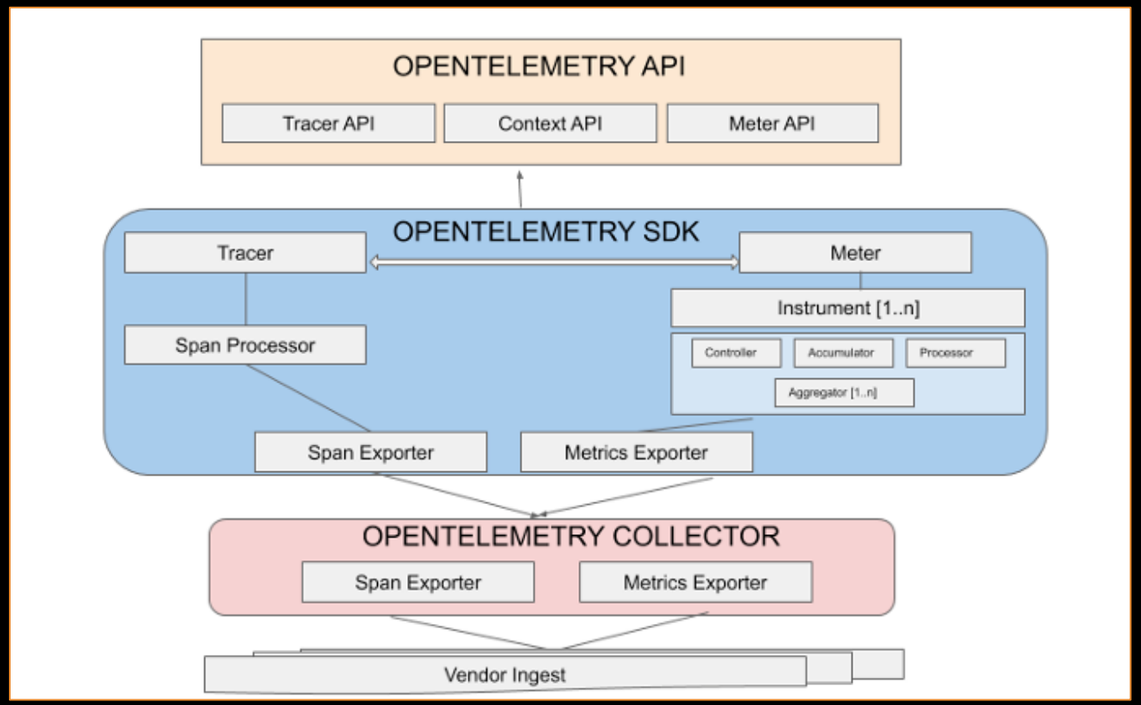
Prometheus

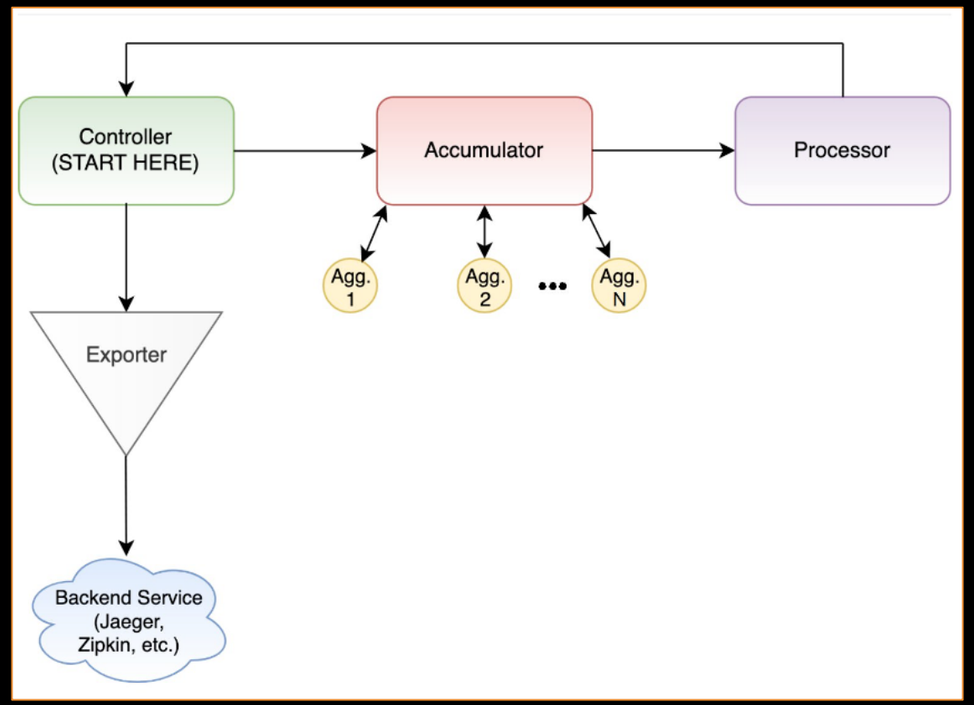


Amazon
EKS

EX3) OpenTelemetry

- Traces – OpenTelemetry C++ library contributions
  https://aws.amazon.com/blogs/opensource/aws-adds-observability-metrics-to-the-opentelemetry-c-library/





Next steps

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Challenges

1. Having to use **different SDKs** and **agents** to **collect/analyse** signals
2. Developers **instrumenting apps** using **different SDKs**, **decreasing velocity**
3. Running **multiple agents** increases resource consumption, causing **unnecessary costs**
4. **Manually correlating** traces & metrics from a range of source is **slow** and **error-prone**









The future: OpenTelemetry

OpenTelemetry enables developers to use a standardized set of open source APIs, SDKs, and agents to instrument apps once and collect correlated metrics & traces for multiple monitoring solutions.



opentelemetry.io

NEW: AWS Distro for OpenTelemetry

https://aws-otel.github.io/

AWS native



Amazon **CloudWatch**



AWS **X-Ray**

OSS-based offerings



Amazon **Elasticsearch Service**



Prometheus



Grafana



Jaeger

Instrumentation (SDKs & agent)

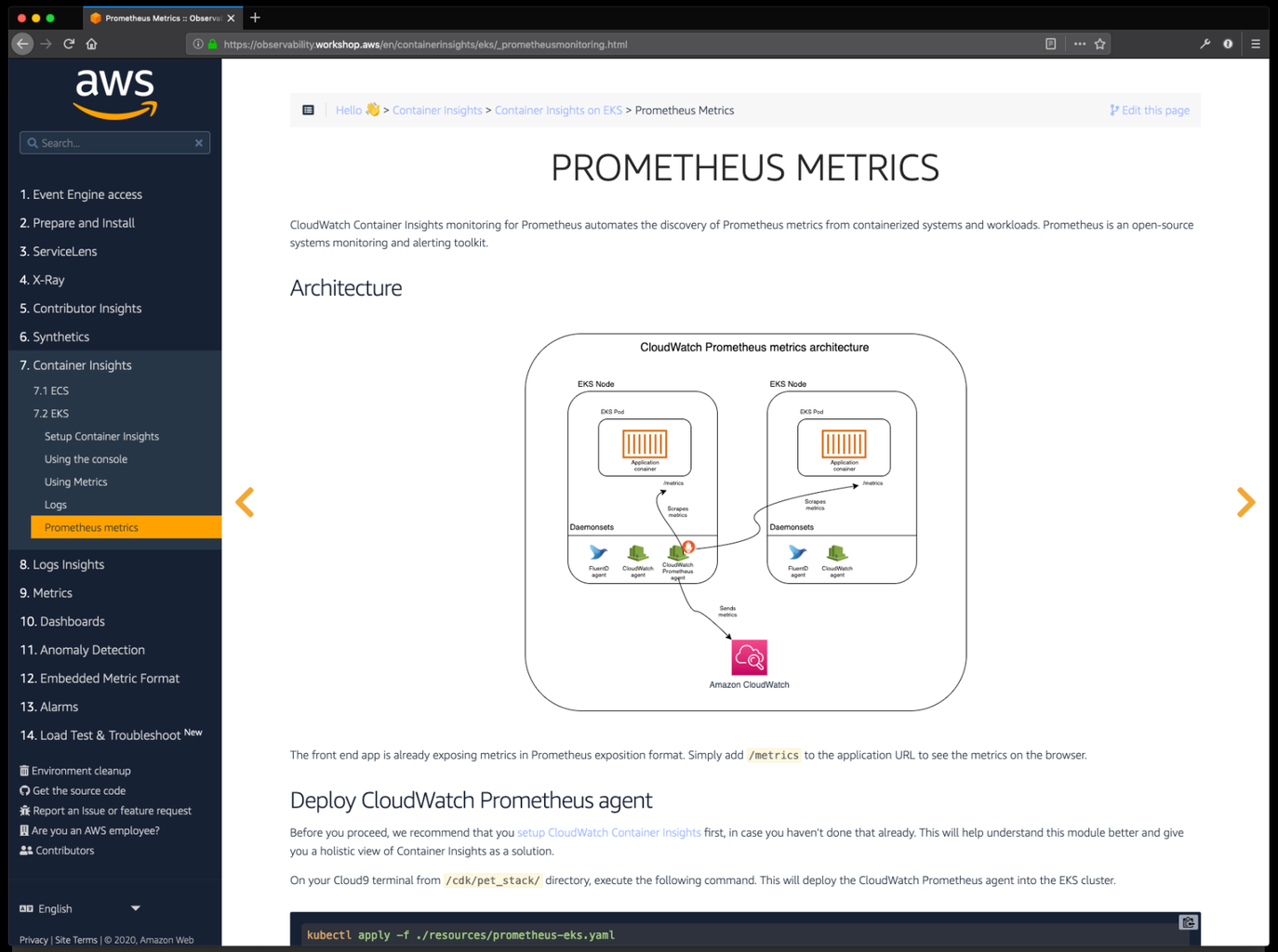
How *you* can benefit

- Freedom of **choice:** use the **best of class** for a given use case
- Rely on **open standards**, most importantly CNCF OpenTelemetry and W3C Trace Context
- With open source, build **portable** observability systems (e.g., on-premises to cloud migration)



Photo by Betty Manousos, licensed under CC BY-NC-ND

Getting started



https://observability.workshop.aws/en/

Thank you!

Michael Hausenblas

hausenbl@amazon.com (Email)

@mhausenblas (GitHub | Twitter | Twitch)

AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.



Please complete the session survey

AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.