



CLOUD NATIVE
— BERGEN —

The Paved Path to Observability

Kasper Borg Nissen, Principal Developer Advocate at  dash0

Who?

Kasper Borg Nissen

Principal Developer Advocate at Dash0

KubeCon+CloudNativeCon EU/NA 24/25 Co-Chair (former)

CNCF Ambassador

Golden Kubestronaut

CNCG Aarhus, KCD Denmark Organizer

Co-founder & Community Lead Cloud Native Nordics



**GOLDEN
Kubestronaut**



Observability course for Platform Engineering

Gain a solid foundation of modern observability in platform engineering, from essential concepts to practical tooling.



Sign up for free

<https://university.platformengineering.org/observability-for-platform-engineering>



Michele Mancioppi

Head of Product @ Dash0



Kasper Borg Nissen

Developer Relations @ Dash0

Observability should feel **invisible**,
but never be **absent**





Why Platform Engineering?



» A digital platform is a foundation of self-service APIs, tools, services, knowledge and support arranged as a compelling internal product. «

- Evan Bottcher

Platform Engineering Principles



Self-Service Experience



Explicit and Consistent APIs



Golden Paths



Modularity

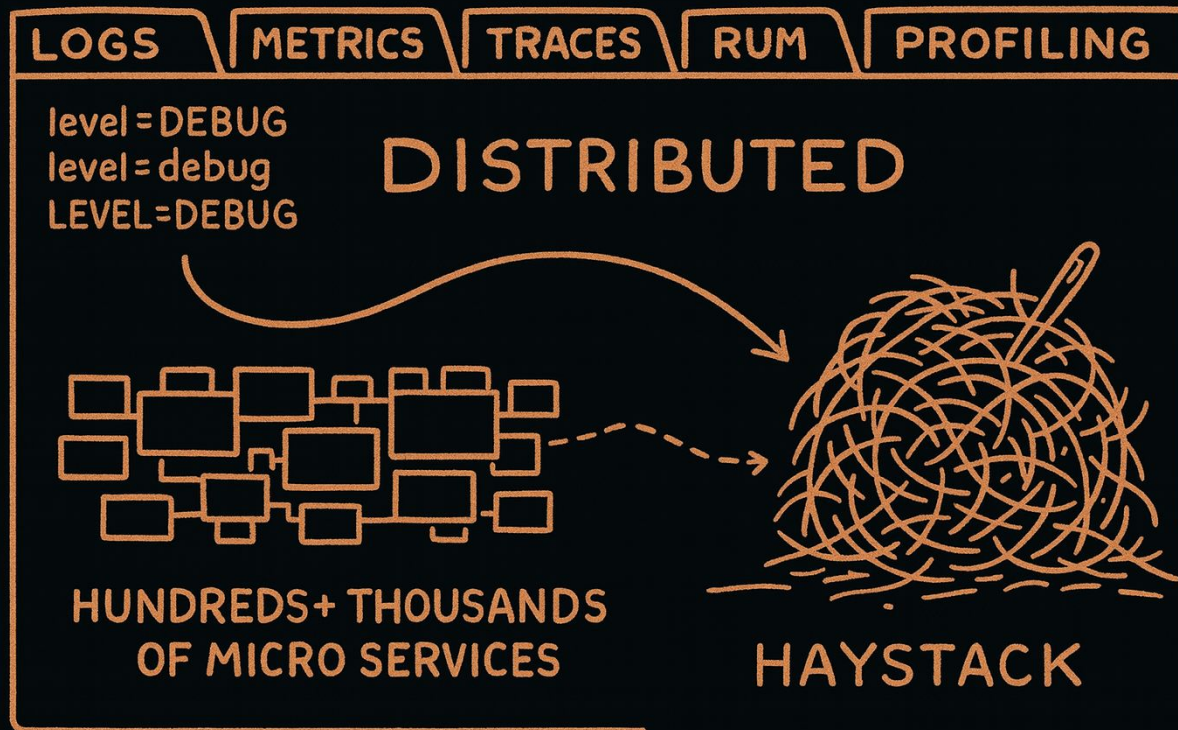


Platform as a Product



Core Requirements

How does this apply to Observability?



This fragmentation, leads to

This fragmentation, leads to



Complex Query
Languages

This fragmentation, leads to



Complex Query
Languages



Vendor lock-in

This fragmentation, leads to



Complex Query
Languages



Vendor lock-in



Metadata Inconsistency



This fragmentation, leads to



Complex Query
Languages



Vendor lock-in



Metadata Inconsistency



No instrumentation due to
high complexity

This fragmentation, leads to



Complex Query
Languages



Vendor lock-in



Metadata Inconsistency



No instrumentation due to
high complexity



Lack of unified insights

As a Platform Engineer, you have a dual role

01

Observing the platform

(cloud, cluster, CI/CD, shared DBs, etc)

02

Enable developers

(auto-instrumentation, logs, metrics, traces)

01

Observe the platform



The foundation



OpenTelemetry

OpenTelemetry (OTel) is an open source project designed to provide standardized tools and APIs for generating, collecting, and exporting telemetry data such as traces, metrics, and logs

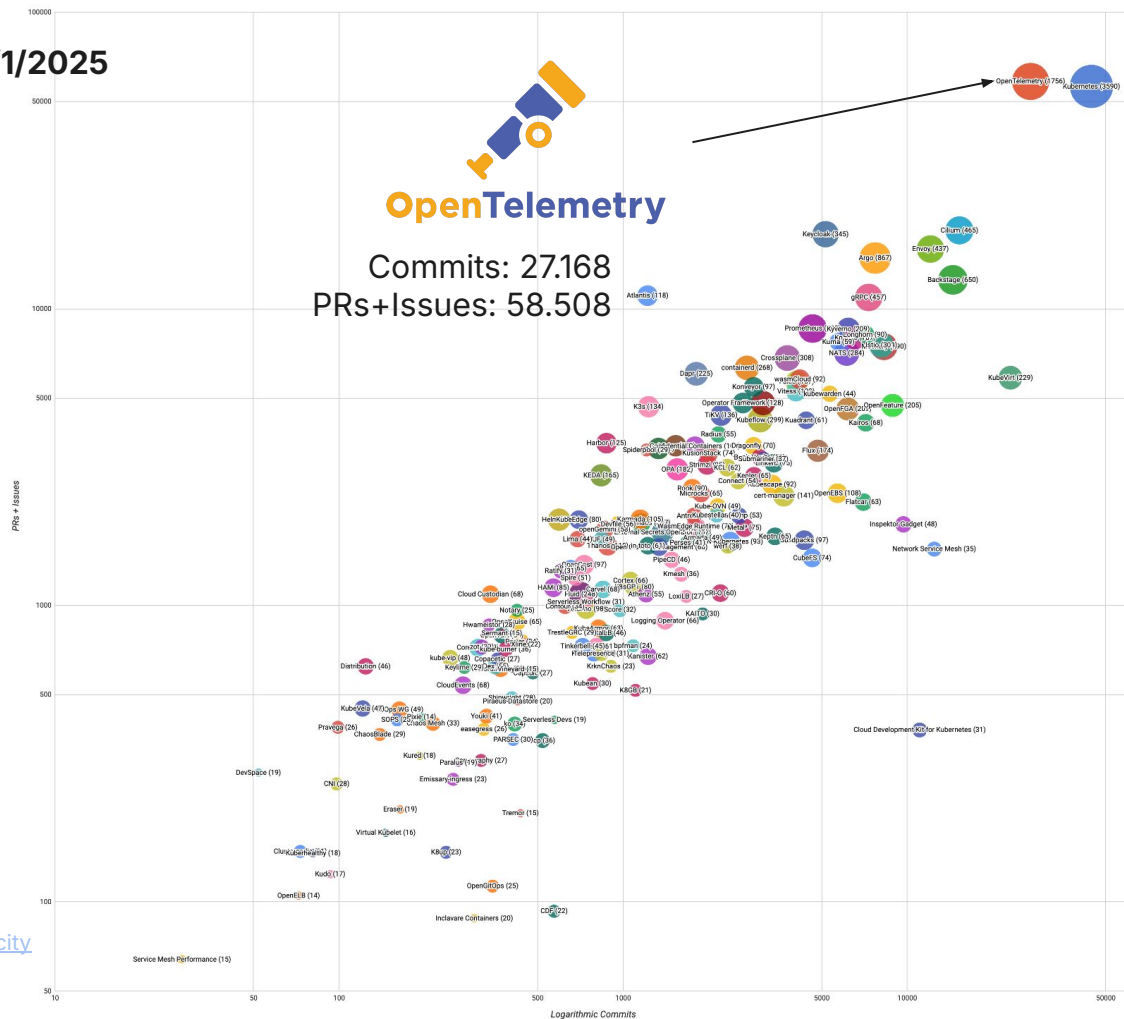
The de-facto standard for distributed tracing, also supports metrics and logs (soon profiling and RUM)

The main goals of the project are:

- Unified telemetry
- Vendor-neutrality
- Cross-platform



1/1/2024-1/1/2025



- Kubernetes (kubernetes.io) 3590 authors
- OpenTelemetry (opentelemetry.io) 1756 authors
- Argo (argoproj.github.io) 867 authors
- Backstage (backstage.io) 650 authors
- Prometheus (prometheus.io) 508 authors
- Cilium (cilium.io) 466 authors
- gRPC (grpc.io) 437 authors
- Envoy (www.envoyproxy.io) 437 authors
- Mesher (layers.io/meshery) 390 authors
- Keycloak (keycloak.org) 343 authors
- Crossplane (crossplane.io) 308 authors
- Istio (istio.io) 301 authors
- KubeFlow (kubeflow.org) 299 authors
- NATS (nats.io) 284 authors
- Containerd (containerd.io) 268 authors
- Fluentd (fluentd.org) 268 authors
- Fluid (github.com/fluid-cloudnative) 248 authors
- KubeVirt (kubewirt.io) 229 authors
- Dapr (dapr.io) 225 authors
- Kyverno (kyverno.io) 209 authors
- OpenFGA (openfga.dev) 204 authors
- OpenFeature (openfeature.dev) 205 authors
- Knative (knative.dev) 187 authors
- OPA (openpolicyagent.org) 182 authors
- Flux (github.com/fluxcd) 174 authors
- Falco (falco.io) 167 authors
- External Secrets Operator (external-secrets.io) 167 authors
- KEDA (keda.sh) 165 authors
- Helix (helm.sh) 159 authors
- etcd (etcd.io) 152 authors
- Jaeger (jaeger.readthedocs.io/en/latest) 150 authors
- cert-manager (cert-manager.io) 141 authors
- TKV (tikv.org) 136 authors
- K3s (k3s.io) 134 authors
- Operator Framework (operatorframework.io) 128 authors
- Harbor (goharbor.io) 128 authors
- Atlantis (runatlantis.io) 125 authors
- Thanos (thanos.io) 110 authors
- OpenEBS (openebs.io) 108 authors
- LitmusChaos (litmuschaos.io) 107 authors
- Karmada (karmada.io) 105 authors
- Vriess (vriess.io) 103 authors
- Confidential Containers (github.com/confidential-containers)
- Volcano (volcano.sh) 99 authors
- Budapester (budapester.io) 97 authors
- OpenCost (kubecost.com) 97 authors
- Konveyor (konveyor.io) 97 authors
- Strimzi (strimzi.io) 95 authors
- OVN-Kubernetes (ovn-kubernetes.io) 93 authors
- WarmCloud (warmcloud.com) 92 authors
- Kubeescape (github.com/kubeescape) 92 authors
- Longhorn (github.com/longhorn) 90 authors
- Rook (rook.io) 90 authors
- Spiffe (spiffe.io) 87 authors
- HAM (project-ham.io) 85 authors
- K8sPT (k8spt.ai) 80 authors
- KubeEdge (kubedge.io) 80 authors
- WarmEdge Runtime (warmedge.org) 79 authors
- Linkerd (linkerd.io) 76 authors
- MetaQ (metaq.io) 75 authors
- CubeFS (cube.io) 74 authors
- KusionStack (kusionstack.io) 74 authors
- Dragonfly (dfly.io/en-us) 70 authors
- Karos (karos.io) 68 authors
- Cloud Custodian (cloudcustodian.io) 68 authors

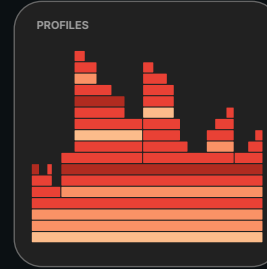
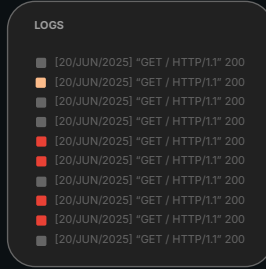
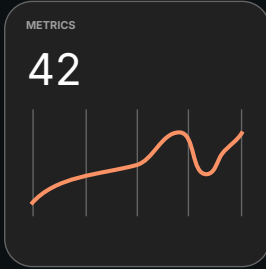


kubernetes

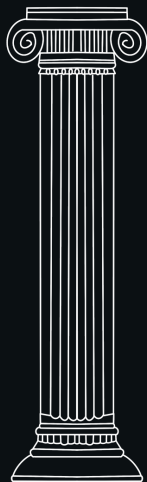
Commits: 44.486

PRs+Issues: 56.299

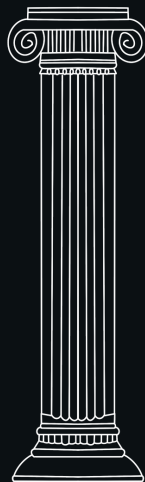
Signals



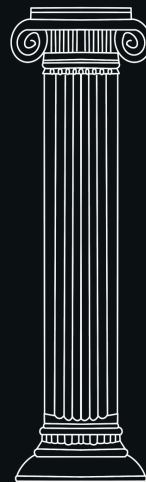
Not pillars ...



Metrics



Logs



Traces

Not pillars ...



**We don't have a *metrics* problem,
or a *tracing* problem.
We have *systems* problems.**

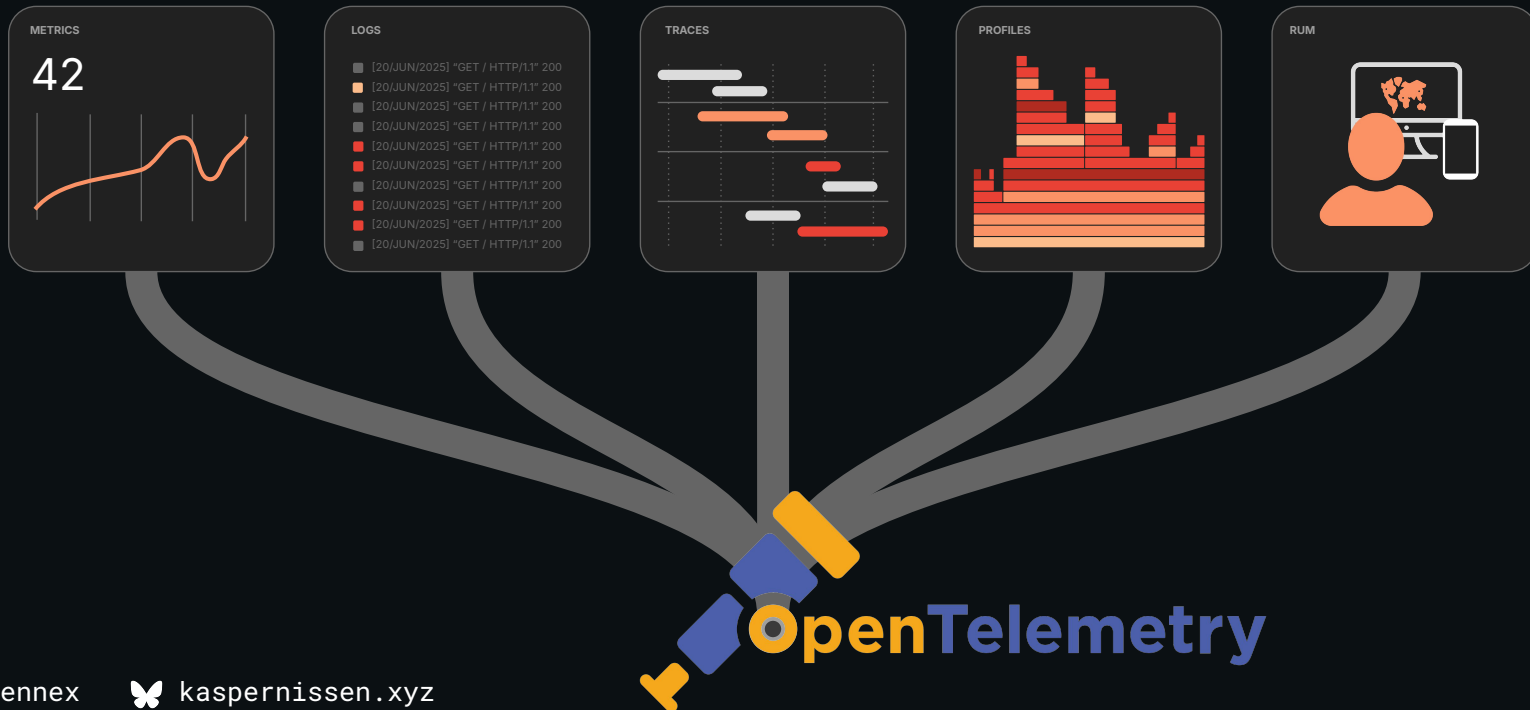


Metrics

Logs

Traces

Correlation is the superpower



02

Enable developers



No-touch instrumentation on Kubernetes



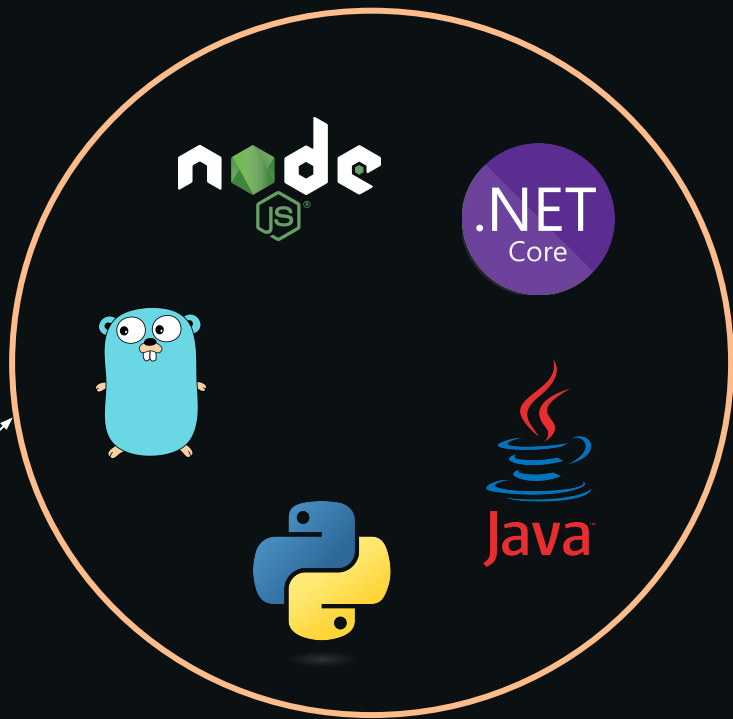
Instrumentation

*Instructs how to inject
auto-instrumentation*

*Injects
instrumentation in
to the pod*

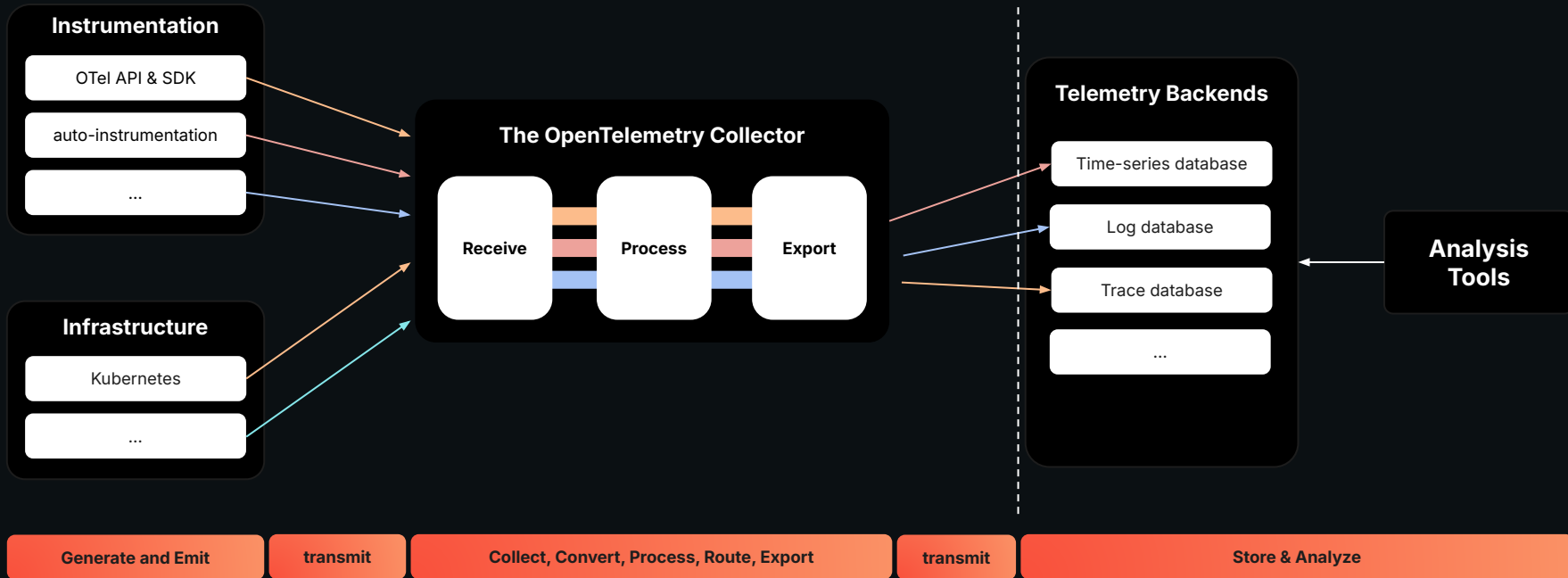


**OpenTelemetry
Operator**



```
instrumentation.opentelemetry.io/inject-go: "<ns>/<instrumentation>"
instrumentation.opentelemetry.io/inject-java: "<ns>/<instrumentation>"
instrumentation.opentelemetry.io/inject-dotnet: "<ns>/<instrumentation>"
instrumentation.opentelemetry.io/inject-nodejs: "<ns>/<instrumentation>"
instrumentation.opentelemetry.io/inject-python: "<ns>/<instrumentation>"
```

OpenTelemetry: the standard plumbing



OpenTelemetry: the standard plumbing

Collection of Telemetry is
standardized



"The last observability agent you will ever install"

Vendor space



... and many more.

Generate and Emit

transmit

Collect, Convert, Process, Route, Export

transmit

Store & Analyze

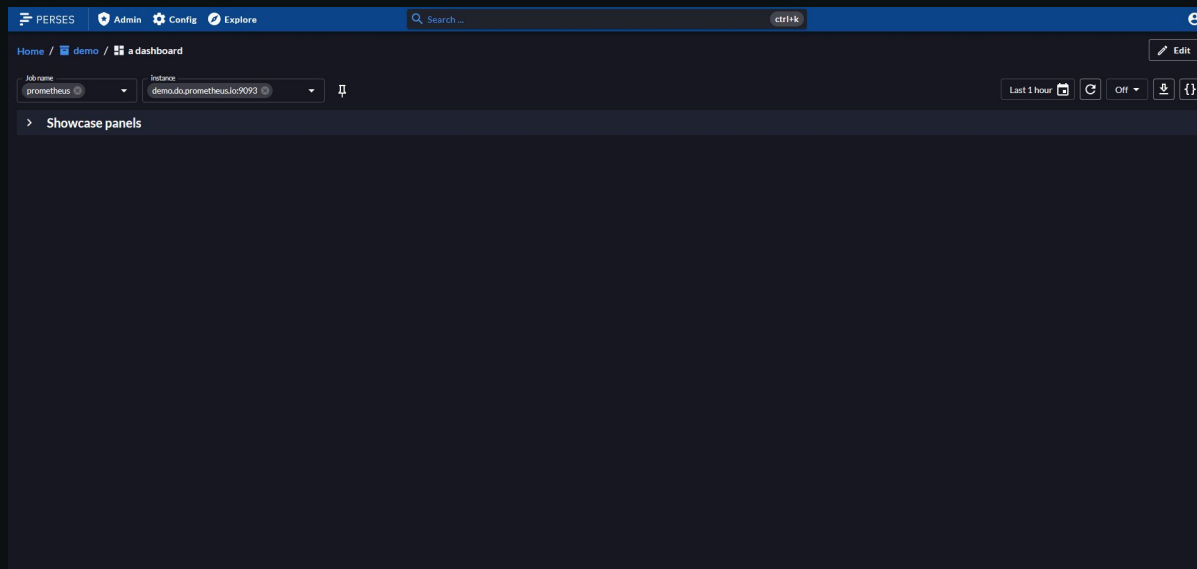


Observability as code



An open specification for
dashboards.

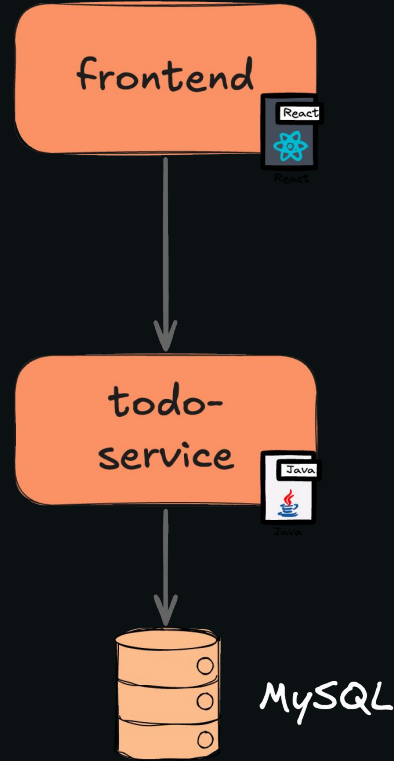
CNCF Sandbox project

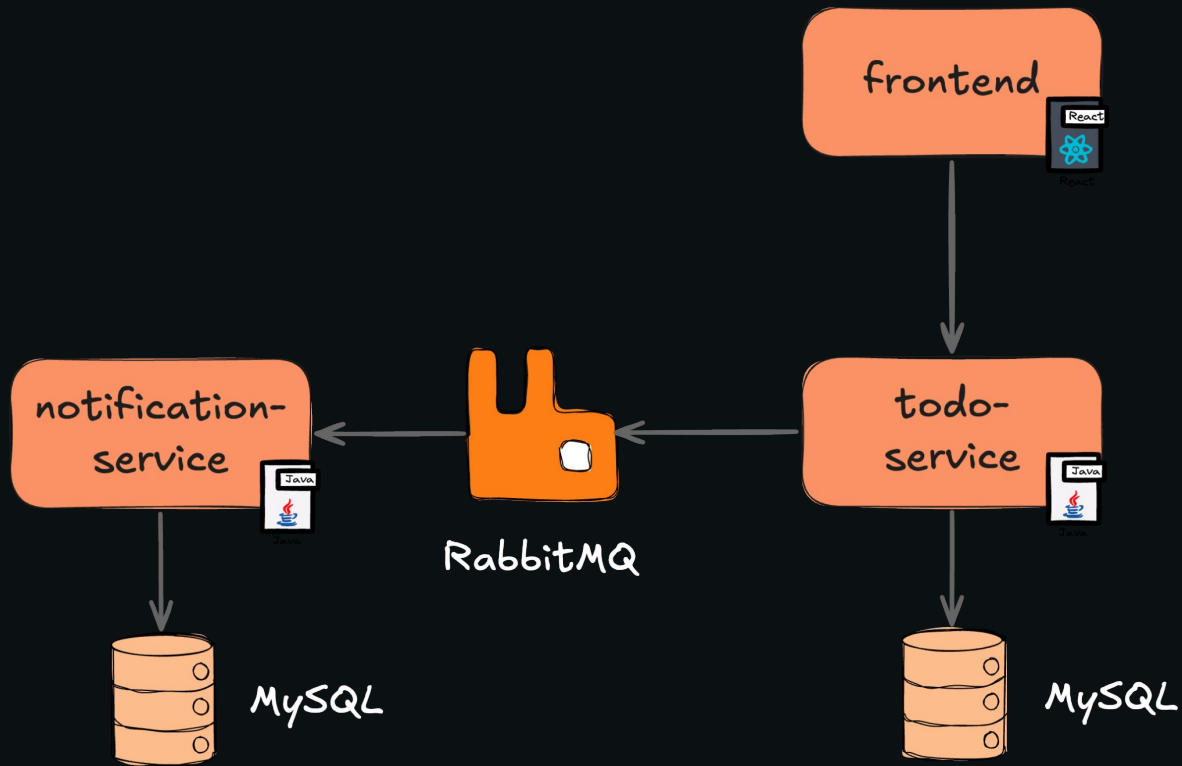


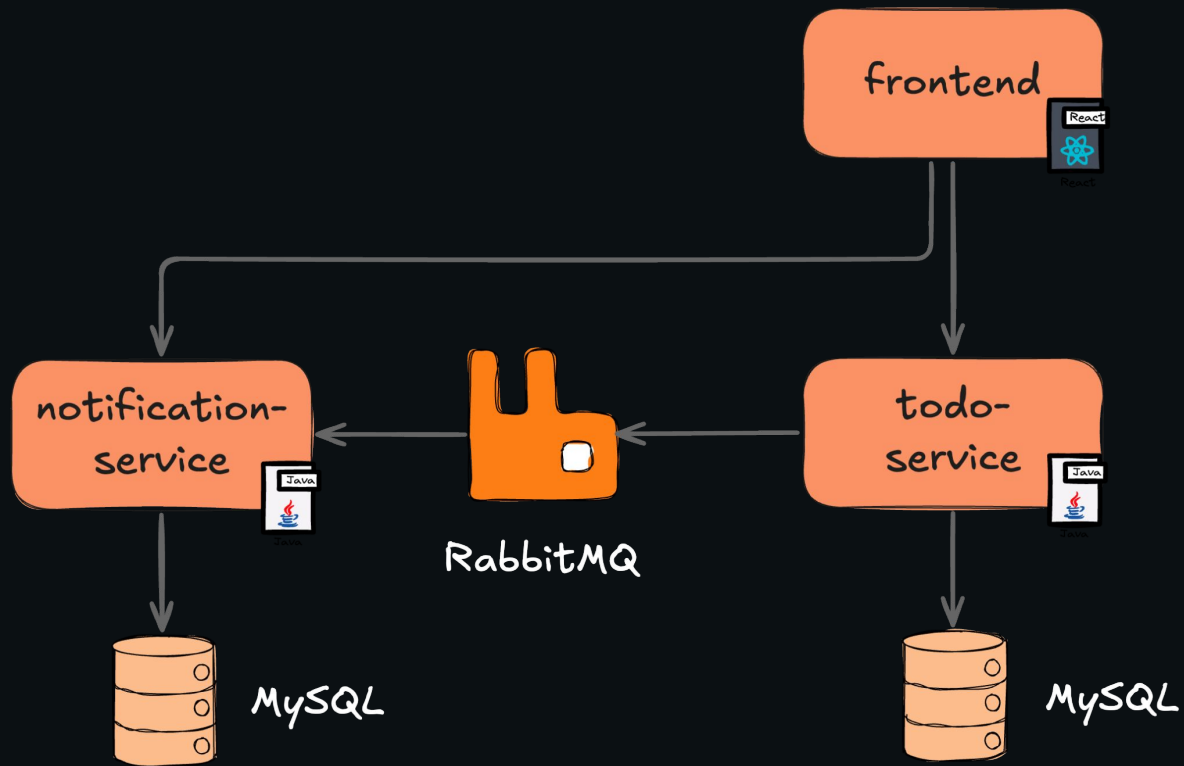
Putting it all together...

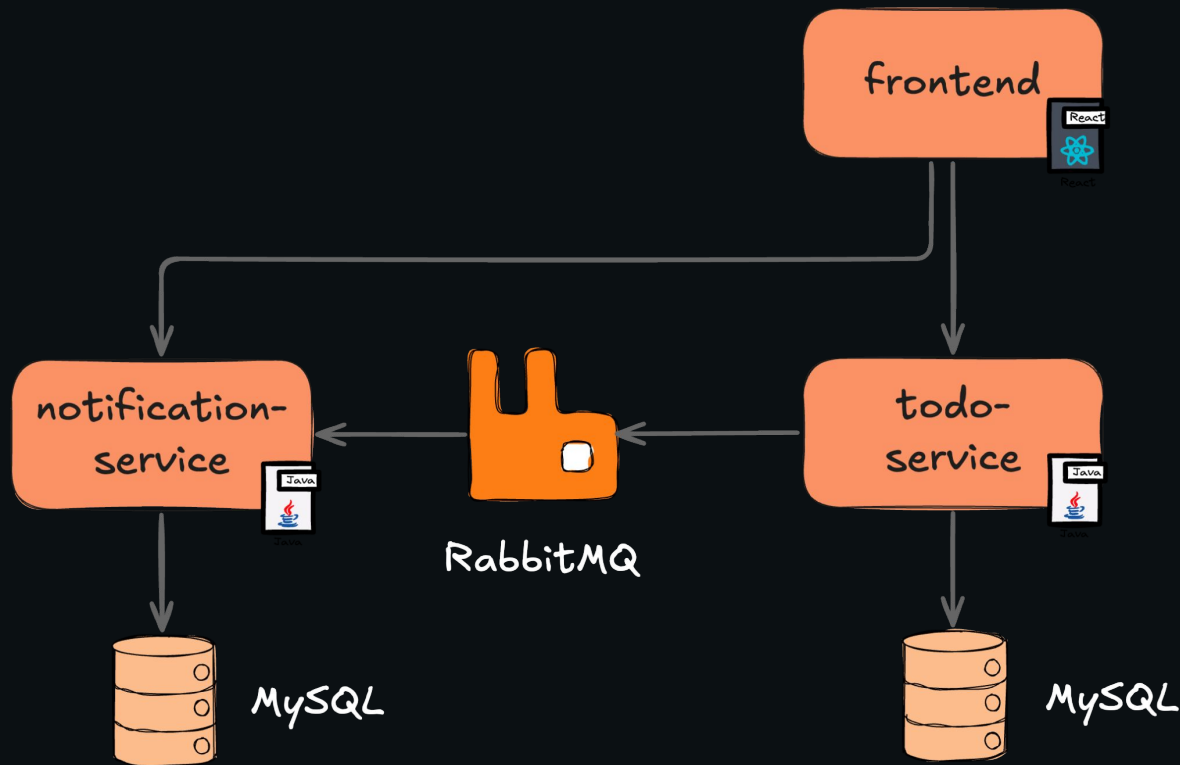
frontend





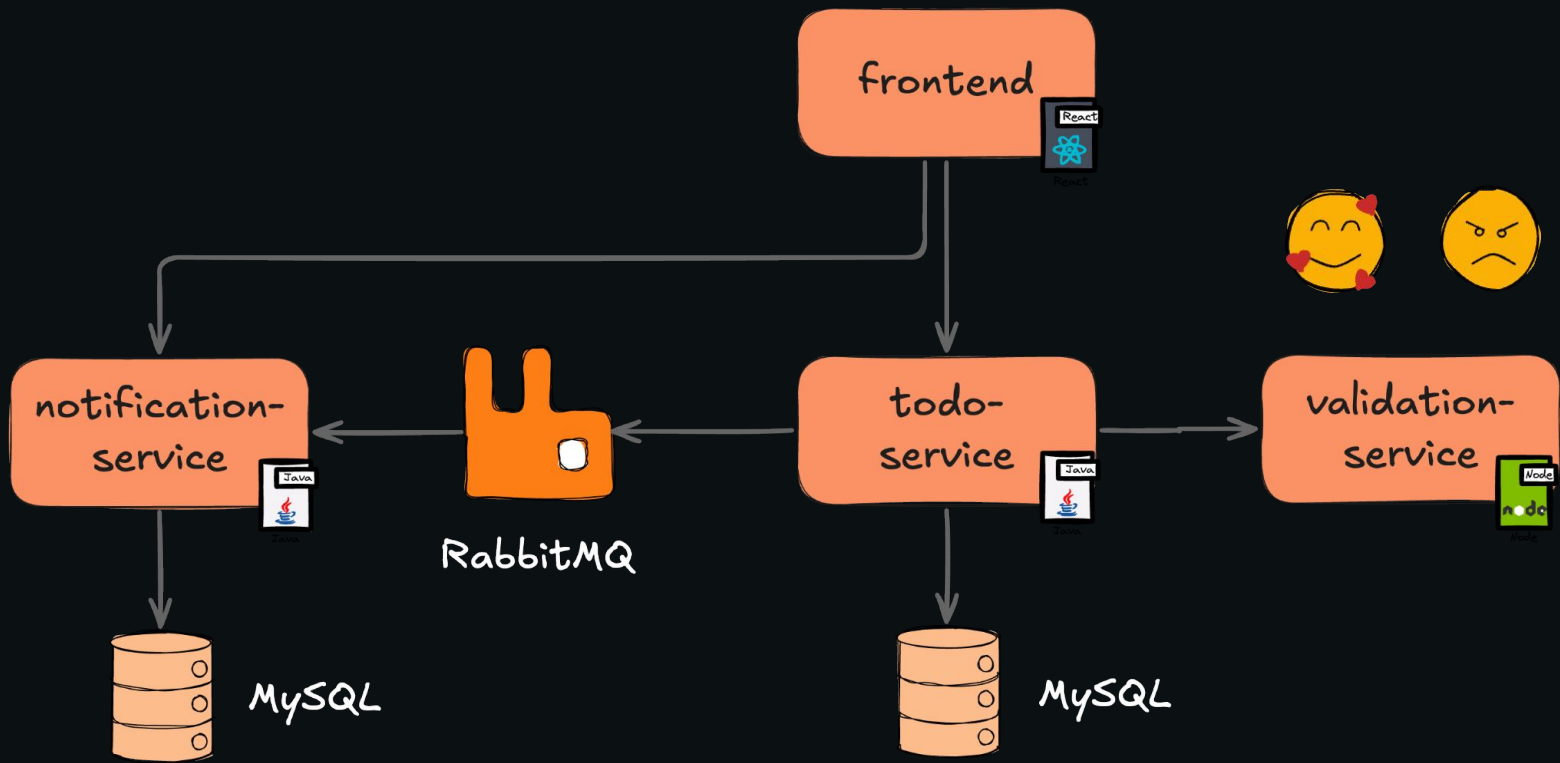




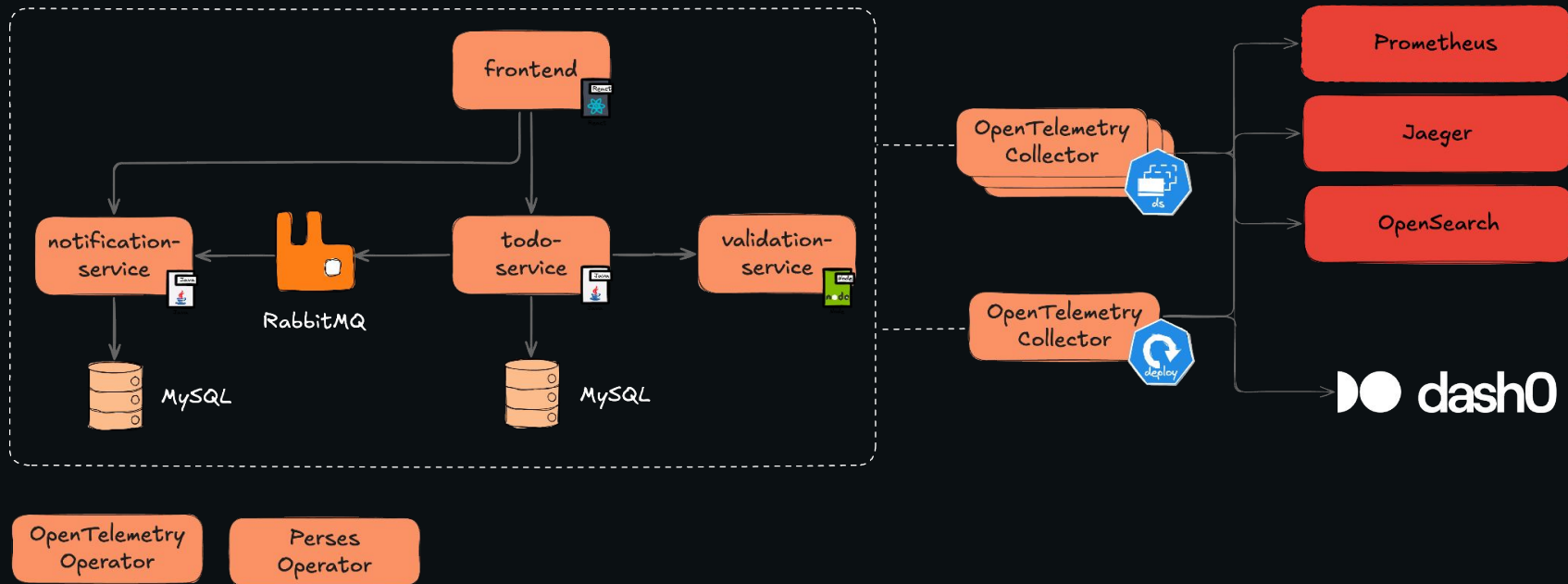


TASK:

Implement a new service that validates the todos and checks them for bad words.



Putting it all together...



Golden defaults for developers



Observability is evolving - fast.

OpenTelemetry is **standardizing** telemetry collection.

Applying **Platform Engineering** principles
can transform **observability** into a seamless,
scalable, and **developer-friendly experience**.



Treat Observability as a **platform product**

Thank you!

Get in touch!

Kasper Borg Nissen, Principal Developer Advocate at  dash0



Demo can be found here!
<https://github.com/dash0hq/otel-platform-demo>