OpenSearchCon EUROPE

Ingesting Observability Data: OpenTelemetry Collector or Data Prepper?

Karsten Schnitter, SAP SE

SAP

Why this question matters

- Understand the OpenTelemetry collector
- Understand Data Prepper
- Understand the differences
- Understand the synergies
- Understand how to use both for building scalable ingestion

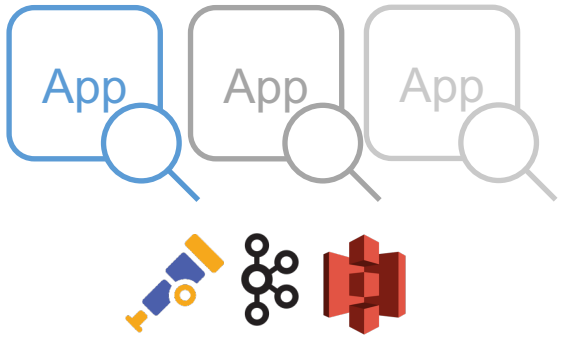
Disclaimer

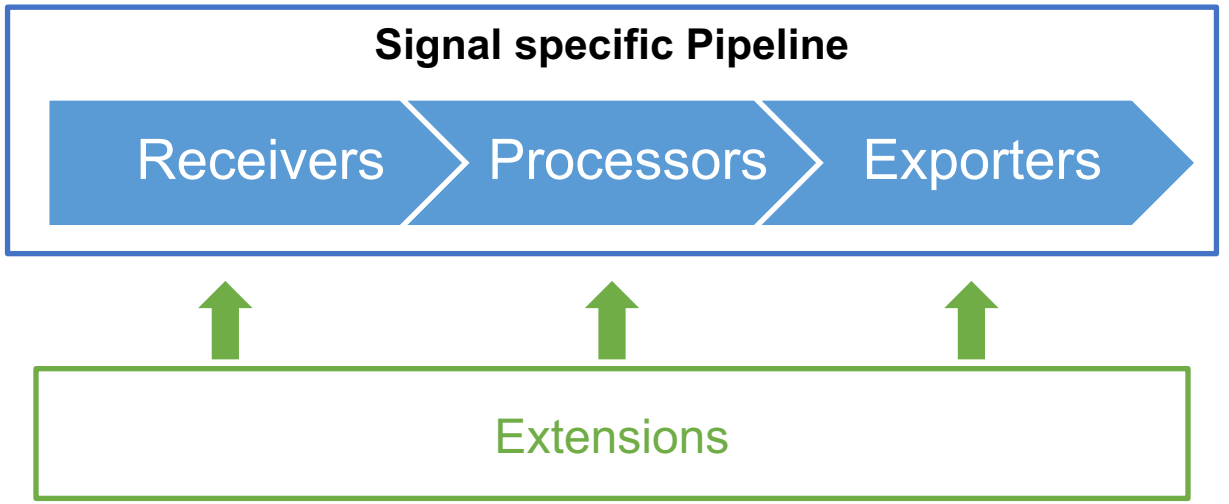
This presentation is *slightly* biased:

- Data has to go to OpenSearch
- OpenTelemetry is used as transport protocol

- Data Prepper maintainer
- TAG Observability member

OpenTelemetry Collector Basics





OpenSearch

Prometheus

OpenTelemetry Collector

OpenSearchCon EUROPE

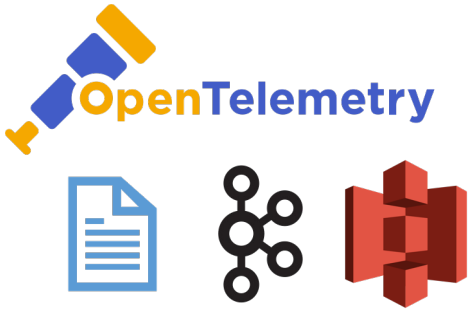
Strengths

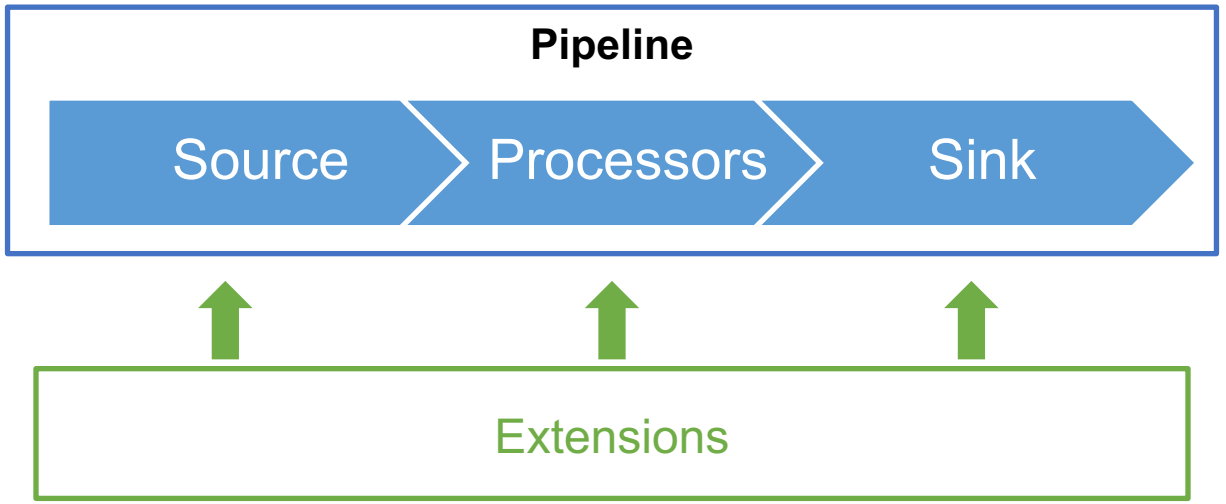
- Vendor-neutral
- Native OTLP support
- Low minimal resource requirements
- Enormous receiver/exporter ecosystem (100+ components)
- OTTL field-level transformation
- K8s resource attribute enrichment
- Runnable as sidecar / DaemonSet or standalone gateway

Areas for Improvement

- Data parsing tied to receivers
- Stateful aggregations work on single instance only
- Inconsistent E2E acks
- No dead-letter-queues
- OpenSearch exporter supports only traces and logs
- Service Map unsupported by OpenSearch

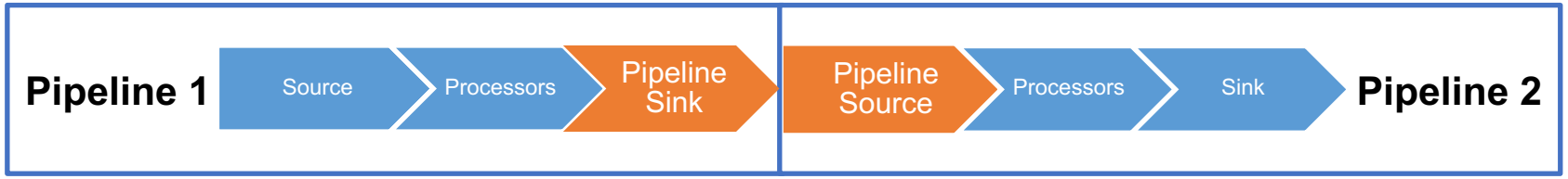
Data Prepper Basics



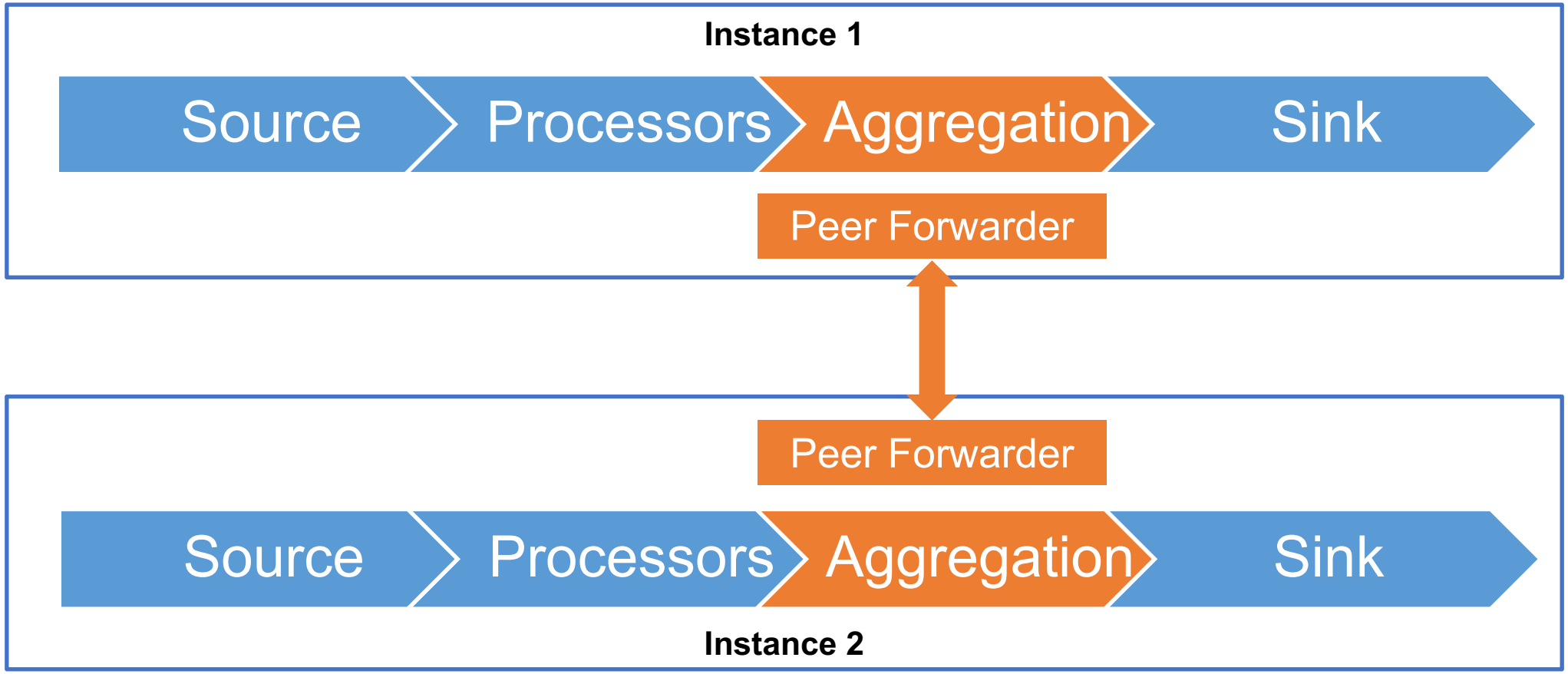


OpenSearch

Prometheus



Data Prepper Peer Forwarding



Data Prepper

Strengths

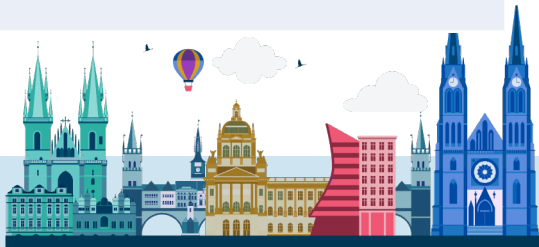
- Processor based parsing
- E2E acks
- Kafka-backed buffers
- State sharing via Peer Forwarder, e.g. allowing Tail-sampling
- Dead-letter-queues
- Advanced OpenSearch index management
- Service Map creation for OpenSearch

Areas of Improvement

- OpenSearch specific
- Low number of sources and sinks
- High minimal resource requirements
- Incomplete documentation

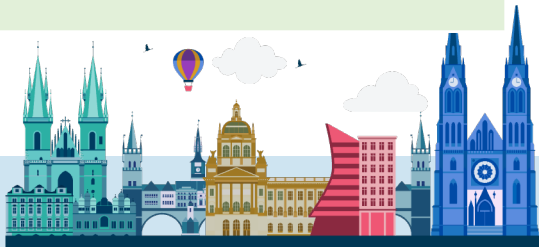
Comparison

	Open Telemetry Collector	Data Prepper
Collection & Transport	⊕ OTLP for logs, metrics, traces, profiles ⊕ Technology specific receivers ⊕ Data routing and dispatching ⊕ K8s infrastructure metrics ⊕ Automatic K8s resource attribute enrichment	⊕ OTLP for logs, metrics, traces ⊖ Restricted selection of sources and sinks ⊕ Data routing and dispatching
Processing & Ingestion	⊕ OTTL and native attribute transformers ⊖ Single instance aggreations only ⊖ OpenSearch basic ingest: bulk API ⊖ incompatible service map	⊕ Rich processor set ⊕ Parsing at any pipeline stage ⊕ Stateful aggregations across multiple instances ⊕ E2E Acknowledgements ⊕ OpenSearch advanced ingest: bulk API, ISM, data streams, retry, DLQ



Comparison

	Open Telemetry Collector	Data Prepper
Collection & Transport	⊕ OTLP for logs, metrics, traces, profiles ⊕ Technology specific receivers ⊕ Data routing and dispatching ⊕ K8s infrastructure metrics ⊕ Automatic K8s resource attribute enrichment	⊕ OTLP for logs, metrics, traces ⊖ Restricted sources and sinks
Processing & Ingestion	⊕ OTTL and native attribute transformers ⊖ Single instance aggreations only ⊖ OpenSearch basic ingest: bulk API ⊖ incompatible service map	⊕ Rich processor set ⊕ Parsing at any pipeline stage ⊕ Stateful aggregations across multiple instances ⊕ E2E Acknowledgements ⊕ OpenSearch advanced ingest: bulk API, ISM, data streams, retry, DLQ



Best of All: Hybrid Architecture

OpenSearch Observability Stack: https://observability.opensearch.org/docs/send-data/

Best of All: Hybrid Architecture

OpenSearch Observability Stack: https://observability.opensearch.org/docs/send-data/

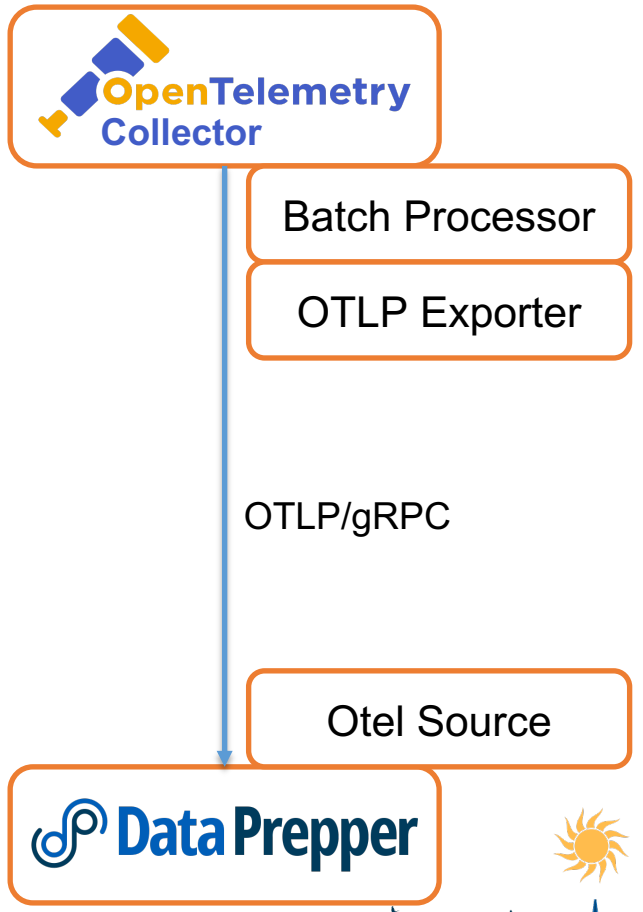
BTP Runtime

SAP

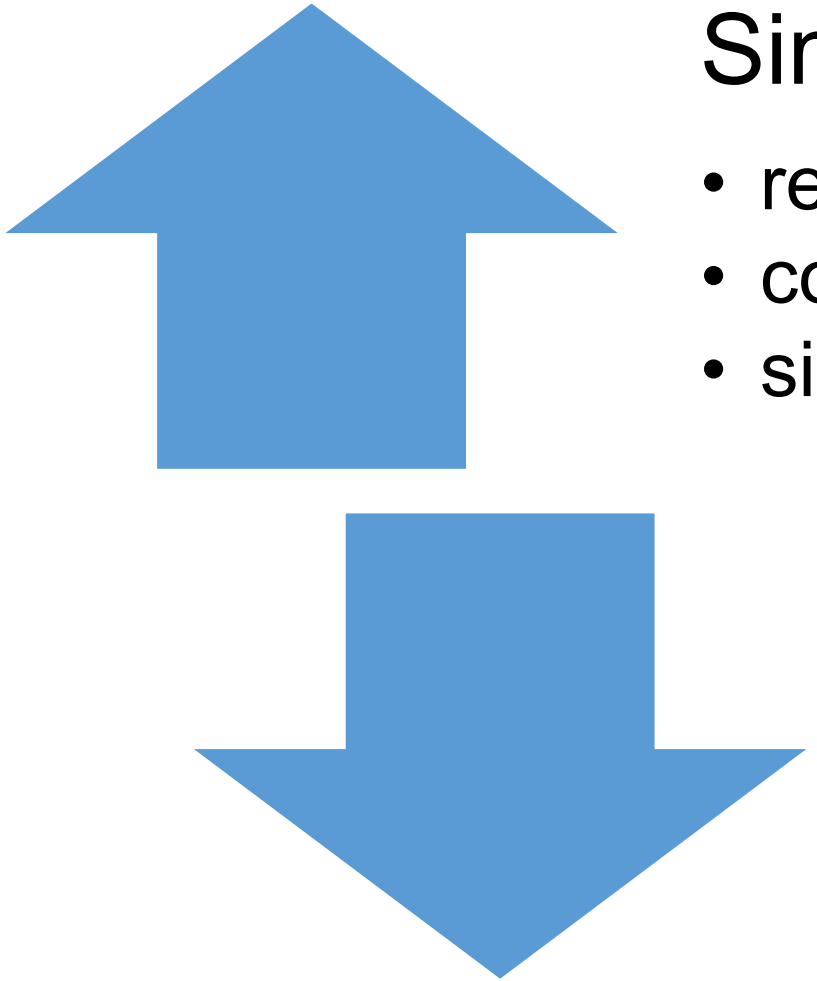
Cloud Logging

Connecting Collector and Data Prepper

- Data batching controlled by Collector
- Received batches buffered by Data Prepper
  - Back-pressure on buffer overload (circuit breaker)
  - Retryable gRPC status code
- Collector batch sizes and Data Prepper buffers need to be aligned
- E2E acknowlegments in Data Prepper leave decision to drop data with Collector
- Use durable buffer to avoid data loss



Decision Drivers



Simple Architecture

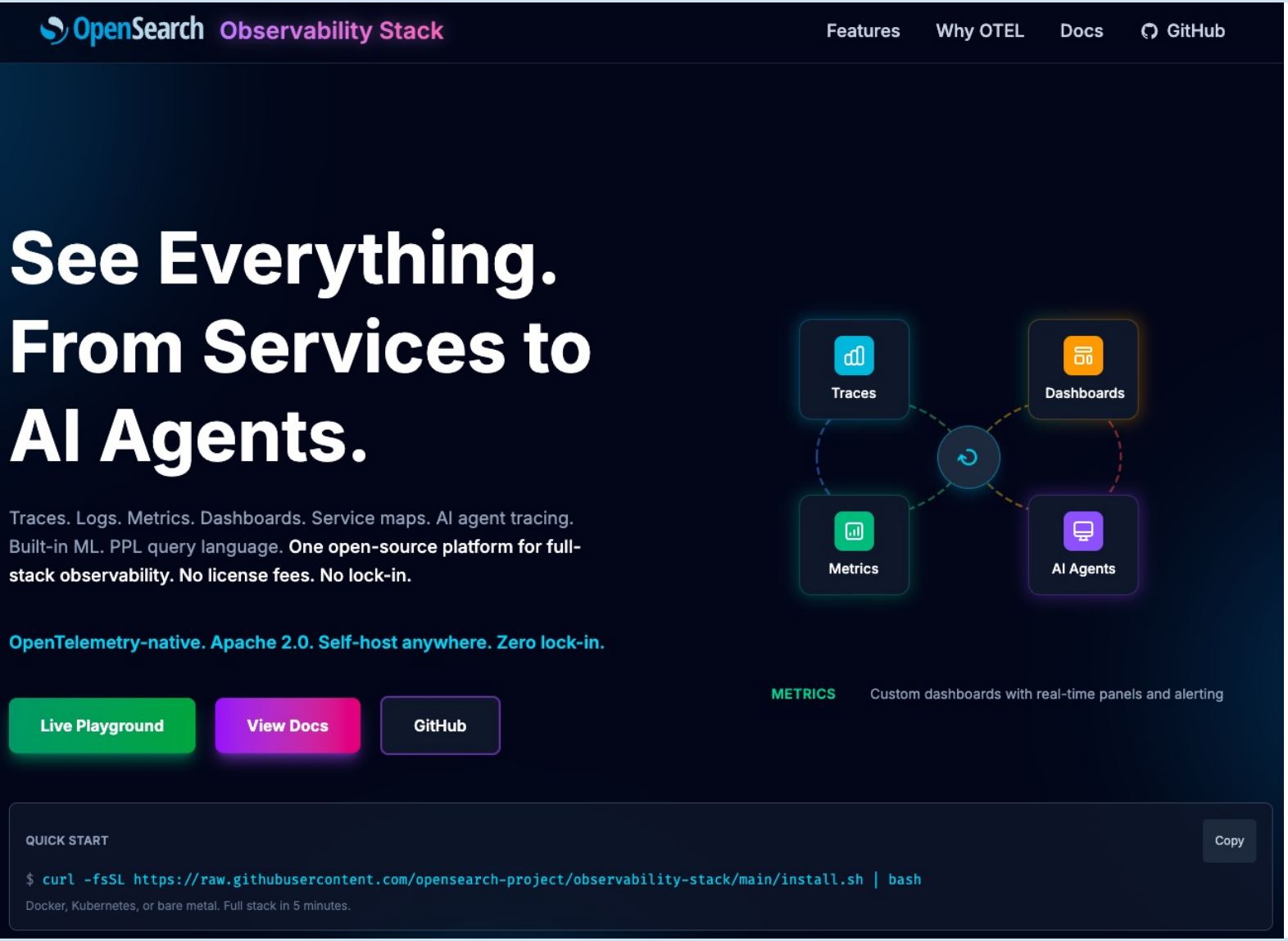
- resource usage
- cost
- simplicity

Hybrid Architecture

- centralization
- throughput
- cloud qualities

OpenSearch Observability Stack

OpenSearchCon EUROPE



- OpenTelemetry native observability
- Ready-to-go setup
- With OpenSearch 3.6 better than ever before
- Start now at: https://observability.opensearch.org/
- Join us at TAG Observability

OpenSearchCon EUROPE