Observability with OpenTelemetry

2024.02

V1

Observability

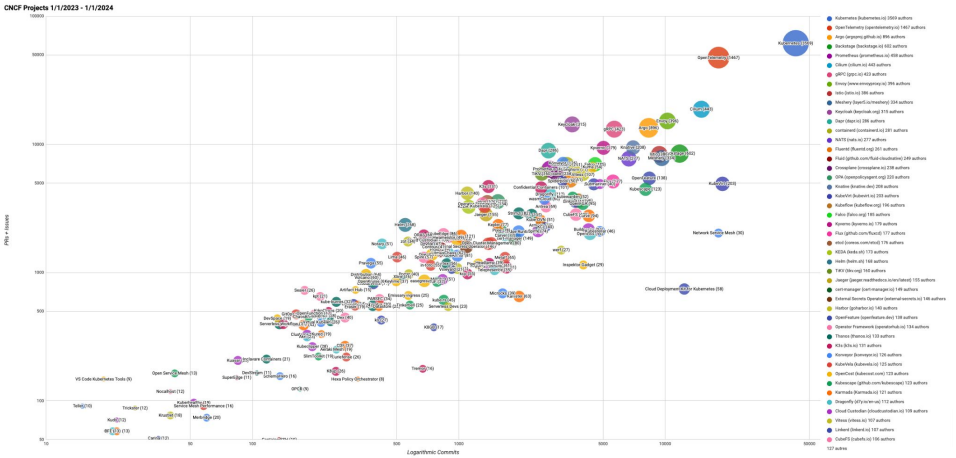
Pillars Of Observability

TEMPLE (from Yuri Shkuro)

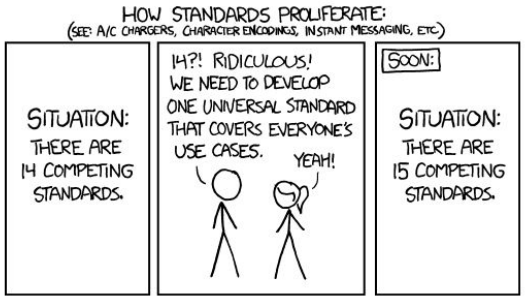
- **Metrics (the original pillar)**: Golden signals with RED (Endpoints and Services) and USE (Applications & Resources)
- **Logs (the ancient pillar)**: Application, Runtime and Traffic
- **Traces (the "new cool kid on the block" pillar)**: Distributed Tracing in a Cloud Native world
- **Events (the misunderstood pillar)**: events that are external to the observed system that cause some changes in that system
- **Profiles (the geek pillar)**: specialize in performance and efficiency optimizations
- **Exceptions (the forgotten pillar):** a specialized form of structured logs

OpenTelemetry

- **CNCF** project, comes from OpenTracing, OpenCensus
- Collection of tools, APIs and SDK to instrument, generate, collect and exporter telemetry data
- Single vendor-agnostic library per programming language
- **Specifications**:
  - API
  - SDK
  - Data (Interface Data Language)
- **OpenTelemetry status**



- **Vendor Neutrality**: provides a unified set of APIs, SDKs, and instrumentation libraries that work across different programming languages and frameworks
- **Standardization and Interoperability**: aims to provide a common standard for observability instrumentation :
  - OpenTelemetry Specification (OTel, v1.30.0): it provides the APIs, SDKs, and data models that all other OTEL standards are derived from
  - OpenTelemetry Protocol (OTLP v1.1.0): describes a common wire protocol for delivering observability data
  - OpenTelemetry Semantic Conventions (1.24.0): define a common set of (semantic) attributes which provide meaning to data when collecting, producing and consuming it
- **Extensibility and Customization**: official binaries or build owns
- Xkcd #927



- **Specification**:
  - API: Defines data types and operations for generating and correlating tracing, metrics, and logging data.
  - SDK: Defines requirements for a language-specific implementation of the API. Data: Defines the OpenTelemetry Protocol (OTLP)
- **Collector**: a vendor-agnostic proxy that can receive, process, and export telemetry data
- **SDKs**: generate telemetry data with your language of choice and export that data to a preferred backend.
- **Instrumentation Libraries**: supports a broad number of components that generate relevant telemetry data from popular libraries and frameworks for supported languages
- **Automatic Instrumentation**: provide a way to instrument your application without touching your source code
- **K8S Operator**: manages the OpenTelemetry Collector and auto-instrumentation of the workloads using OpenTelemetry

OpenTelemetry Components Lifecycle

- **Draft** components are under design, and have not been added to the specification.
- **Experimental** components are released and available for beta testing.
- **Stable** components are backwards compatible and covered under long term support.
- **Deprecated** components are stable but may eventually be removed.

OpenTelemetry Signals

- **Traces**: what happens when a request is made to an application
- **Metrics**: a measurement about a service, captured at runtime
- **Logs**: a timestamped text record, either structured (recommended) or unstructured, with metadata.
- **Baggage:** Contextual informations that's passed between spans
- **Events**: LogRecords and Events are both represented using the same data model.
- **Profiles**: https://github.com/open-telemetry/oteps/pull/237

OpenTelemetry Semantic Conventions

Defined for the following areas:

- General: General Semantic Conventions.
- Cloud Providers: Semantic Conventions for cloud providers libraries.
- CloudEvents: Semantic Conventions for the CloudEvents specification.
- Database: Semantic Conventions for database operations.
- Exceptions: Semantic Conventions for exceptions.
- FaaS: Semantic Conventions for Function as a Service (FaaS) operations.
- Feature Flags: Semantic Conventions for feature flag evaluations.
- HTTP: Semantic Conventions for HTTP client and server operations.
- Messaging: Semantic Conventions for messaging operations and systems.
- Object Stores: Semantic Conventions for object stores operations.
- RPC: Semantic Conventions for RPC client and server operations.
- System: System Semantic Conventions.

OpenTelemetry Collector

Vendor-agnostic way to receive, process and export telemetry data.

- **Receiver**: obtain the telemetry data (push or pull model)
- **Processor**: run on data between being received and being exported
- **Exporter**: get the telemetry data out of the collector (push or pull model)
- **Pipelines**: a chain of receiver, zero or more processors, and exporters
- **Connectors**: both an exporter and receiver (span to logs, spans to metrics, …)
- **Extensions**: available primarily for tasks that do not involve processing telemetry data (healthcheck, auth/z, …)

OpenTelemetry Distribution

- A collection of components (receivers, processors, ...) assembled in a binary (the list)
- **Official**:
  - Core
  - Contrib
- From **vendor**:
  - AWS Distro for OpenTelemetry (ADOT)
  - Azure Monitor OpenTelemetry Distro
  - Grafana Agent
  - Splunk
  - Sumologic
  - ...
- Custom Builder using the OpenTelemetry Collector Builder

OpenTelemetry Collector Builder

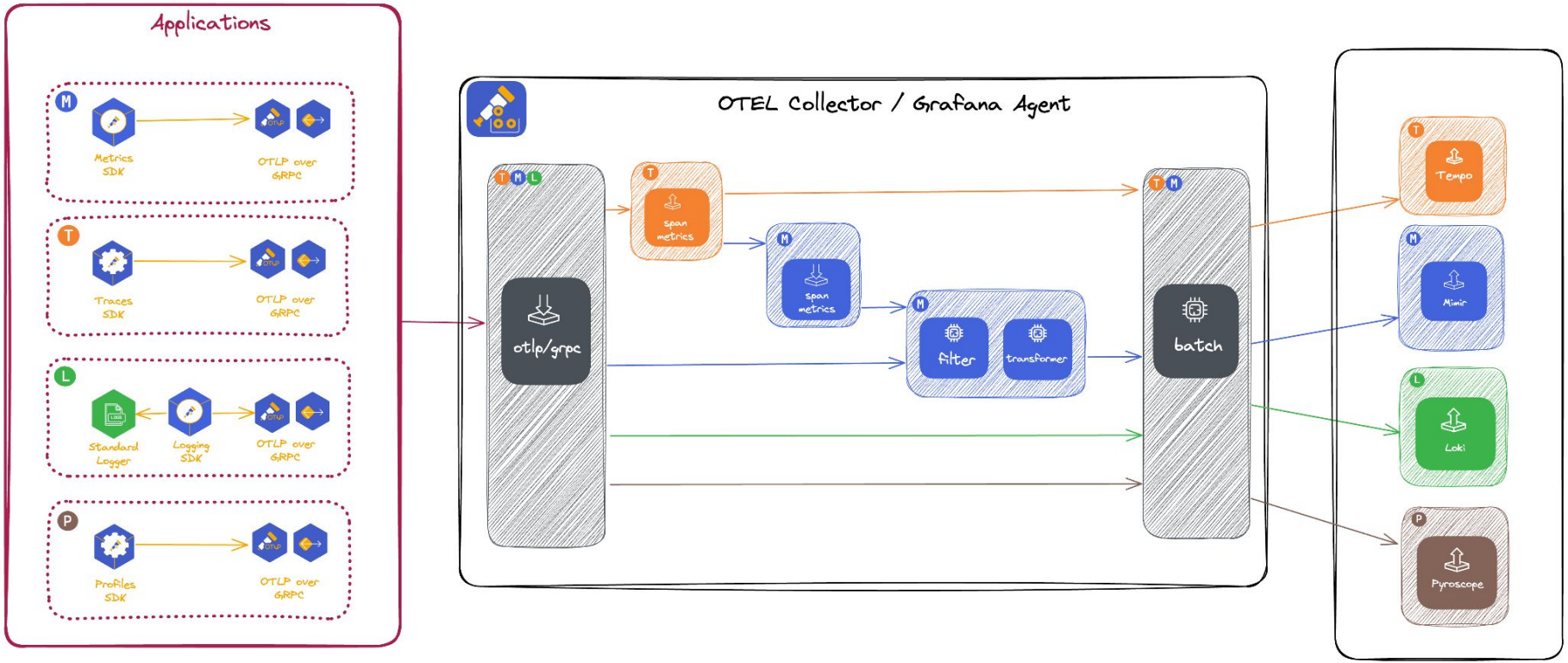
- Install the builder
- Create a builder manifest file
- Use the OTel Registry to find components
- Generating the Code and Building your Collector's distribution:

```
./ocb --config builder-config.yaml
```

Grafana Agent

Is a vendor-neutral, batteries-included telemetry collector with configuration inspired by **Terraform**.

- Every signal: Collect telemetry data for metrics, logs, traces, and continuous profiles.
- Deployments: Static, Operator or Flow
- Write programmable pipelines with ease, and debug them using a **built-in UI**.


Applications
Metrics SDK
OTLP over GRPC
Traces SDK
OTLP over GRPC
Standard Logger
Logging SDK
OTLP over GRPC
Profiles SDK
OTLP over GRPC
OTEL Collector / Grafana Agent
otlp/grpc
span metrics
span metrics
filter
transformer
batch
Tempo
Mimir
Loki
Pyroscope

Questions