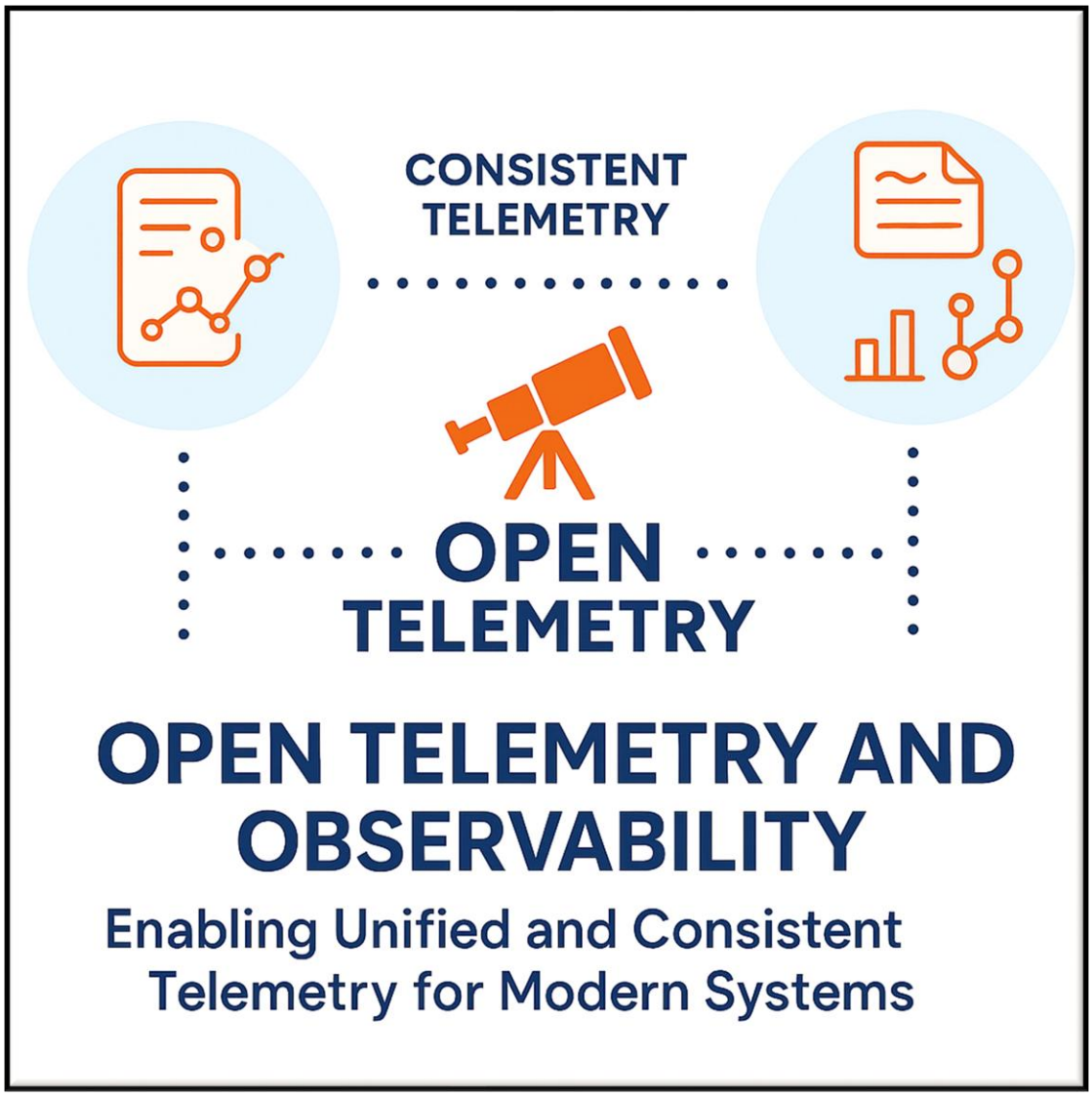

Created by:

Thomas Bronack, president

Data Center Assistance Group, LLC

bronackt@dcag.com | bronackt@gmail.com

(917) 673-6992

OpenTelemetry and Observability: Enabling Unified and Consistent Telemetry for Modern Systems

Overview: OpenTelemetry and Observability

Modern software systems—especially those built using microservices, containers, and distributed architectures—generate enormous volumes of telemetry data (logs, metrics, and traces). However, until recently, there was no consistent standard for collecting and correlating this data across services, languages, and platforms.

Why OpenTelemetry Was Created

OpenTelemetry (OTel) was developed by the Cloud Native Computing Foundation (CNCF) to unify telemetry generation under a single, vendor-neutral standard. It merged two projects—OpenTracing and OpenCensus—to simplify instrumentation, reduce duplication, and provide developers with a consistent SDK (Software Development Sit) and data model across services and platforms.

What is Observability?

Observability refers to the ability to understand the internal state of a system based on external outputs. It is enabled through three core types of telemetry data:

- **Logs:** Timestamped records of discrete events
- **Metrics:** Time-series of numerical data
- **Traces:** End-to-end request tracking across services

How OpenTelemetry Supports Observability

OpenTelemetry (OTel) provides:

- Standardized instrumentation libraries for multiple programming languages
- Automatic trace injection across supported frameworks
- Exporters to route telemetry data to backends like Prometheus, Grafana, Jaeger, and others
- Correlation of logs, metrics, and traces for enhanced root cause analysis and system understanding

Example Use Case

In a Kubernetes-based microservices environment, OTel allows:

- Tracing of user requests through services and APIs
- Correlation of performance metrics with trace events
- Export to observability platforms for visualization and alerting

Products Supporting OpenTelemetry & Observability

Open Source Tools:

- Prometheus
- Grafana
- Jaeger
- Tempo
- Loki\- Elastic Stack

Commercial Platforms:

- Datadog
- New Relic
- Splunk Observability Cloud
- Dynatrace
- Honeycomb.io
- Lightstep
- AWS Distro for OpenTelemetry (ADOT)
- Azure Monitor
- Google Cloud Operations Suite

Adoption Strategy

1. Begin with auto-instrumentation using OpenTelemetry SDKs
2. Deploy OpenTelemetry Collector to receive, process, and export telemetry
3. Send telemetry data to your chosen observability backend(s)
4. Enhance custom instrumentation as needed
5. Use dashboards and alerts to correlate signals and gain real-time insights

For additional information contact:

Thomas Bronack, president
Data Center Assistance Group, LLC
bronackt@dcag.com | bronackt@gmail.com
(917) 673-6992
Website: https://www.dcag.com

