



Microservices Observability Patterns

Paulo Alberto Simões

Developer Evangelist, Data-Driven Microservices with Converged Database

What are Microservices?

If every service has to be updated at the same time it's not loosely coupled

“loosely coupled service-oriented architecture with bounded contexts.”

If you know too much about the surrounding services you don't have a bounded context

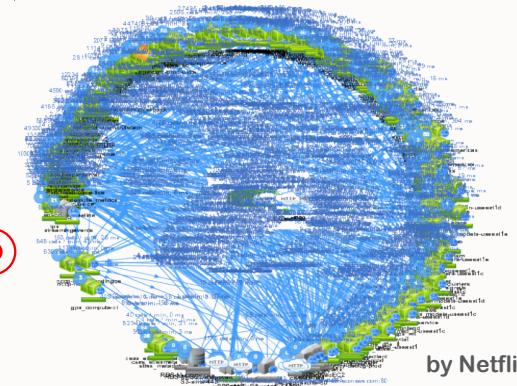
Adrian Cockroft, while at Netflix

Cloud Native Design Principles



Containerized

Open
Source
Software

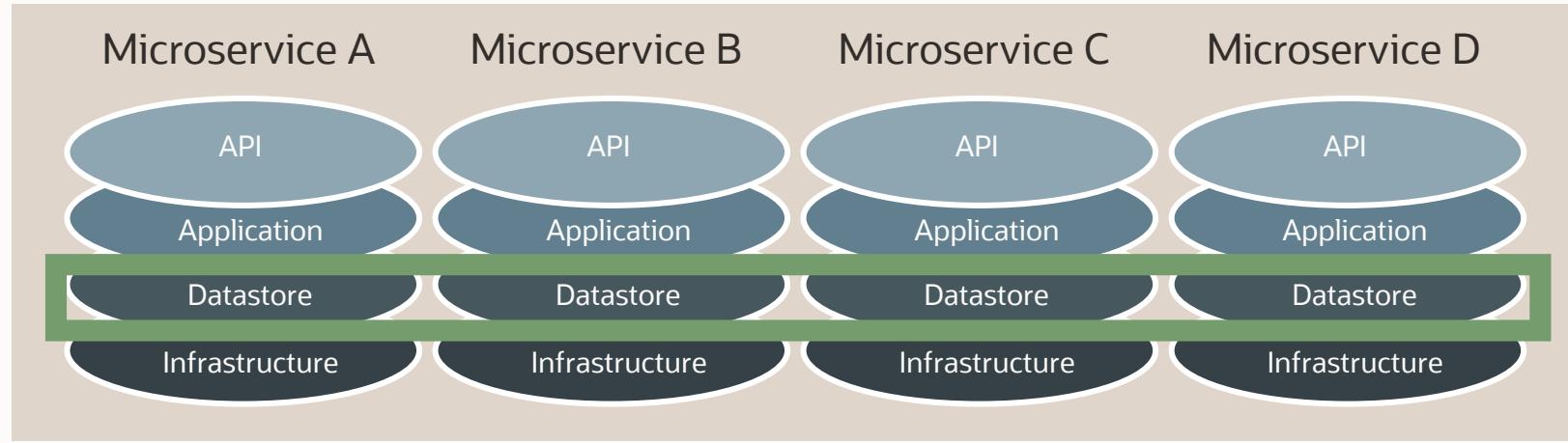


Microservices Oriented



Dynamically orchestrated

Microservices Forces Move To Distributed Computing

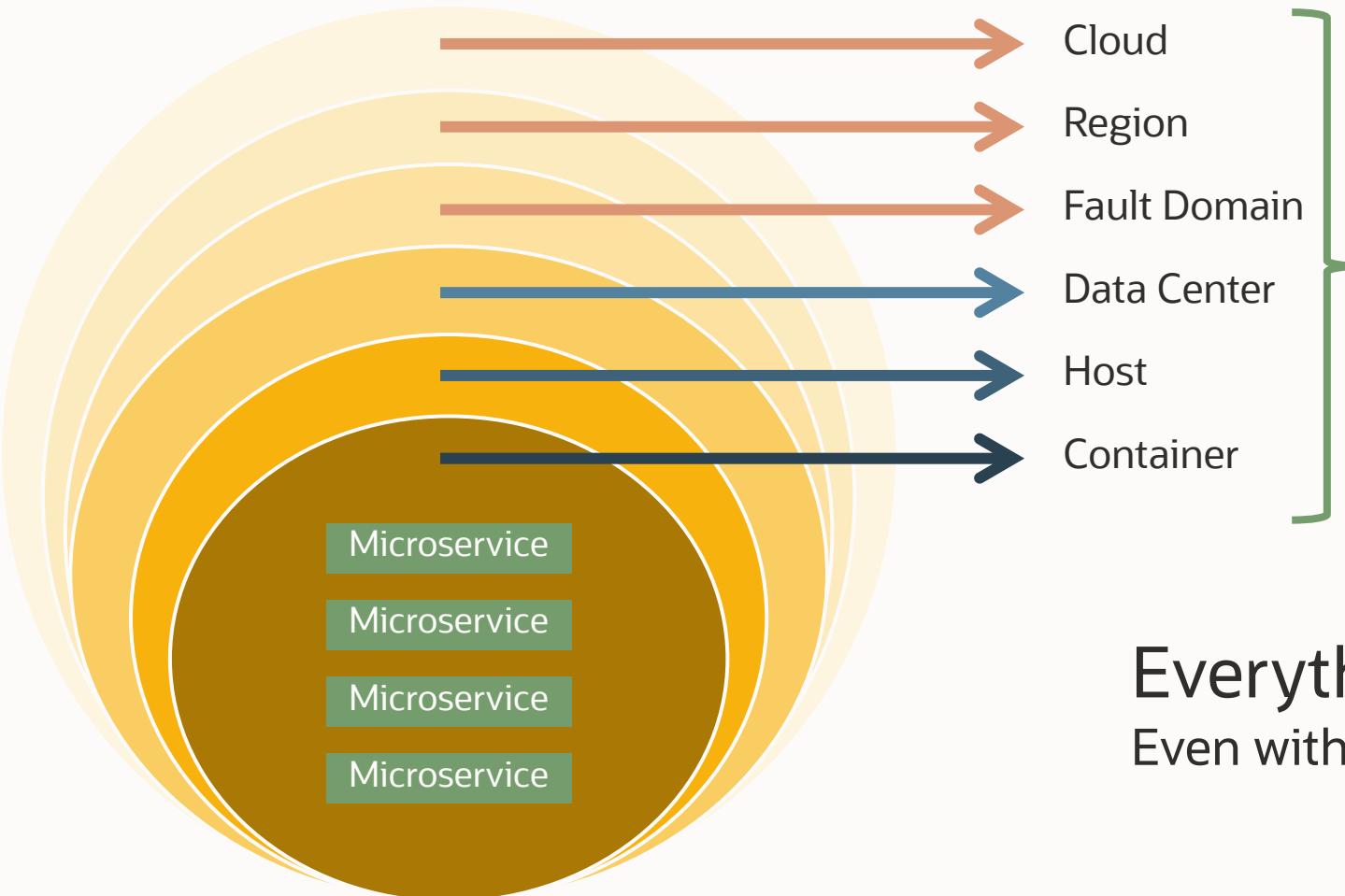


- Distributed computing is a natural consequence of microservices because each microservice has its own datastore
- Sharing datastores across microservices introduces coupling – very bad!
- There will always be latency between microservices
- Latency = eventual consistency

- All data exchange between microservices must be through API layer or messaging – no accessing datastores cross-microservices
- There are advantages to implement high-speed messaging between microservices. REST + HTTP may not be fast enough
- May end up duplicating data across datastores – e.g., a customer's profile



Distributed Computing forces Everything distributed



Run your applications across multiple data centers, fault domains, regions, etc

Hundreds of milliseconds of latency

Everything is now distributed
Even within the same data center



l.cnfcf.io

This Cloud Native Trail Map is a recommended process for leveraging Open Source, cloud native technologies



CLOUD NATIVE TRAIL MAP

The Cloud Native Landscape l.cnfcf.io has a large number of options. This Cloud Native Trail Map is a recommended process for leveraging open source, cloud native technologies. At each step, you can choose a vendor-supported offering or do it yourself, and everything after step #3 is optional based on your circumstances.

HELP ALONG THE WAY

A. Training and Certification

Consider training offerings from CNCF and then take the exam to become a Certified Kubernetes Administrator or a Certified Kubernetes Application Developer

cncf.io/training

B. Consulting Help

If you want assistance with Kubernetes and the surrounding ecosystem, consider leveraging a Kubernetes Certified Service Provider

cncf.io/kcsp

C. Join CNCF's End User

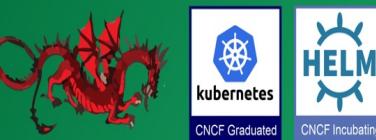
Cloud Native Trail Map

- Commonly done with Docker containers
- Any size application and dependencies (even PDP-11 code running on an emulator) can be containerized
- Over time, you should aspire towards splitting suitable applications and writing future functionality as microservices



3. ORCHESTRATION & APPLICATION DEFINITION

- Kubernetes is the market-leading orchestration solution
- You should select a Certified Kubernetes Distribution, Hosted Platform, or Installer: cncf.io/ck
- Helm Charts help you define, install, and upgrade even the most complex Kubernetes application



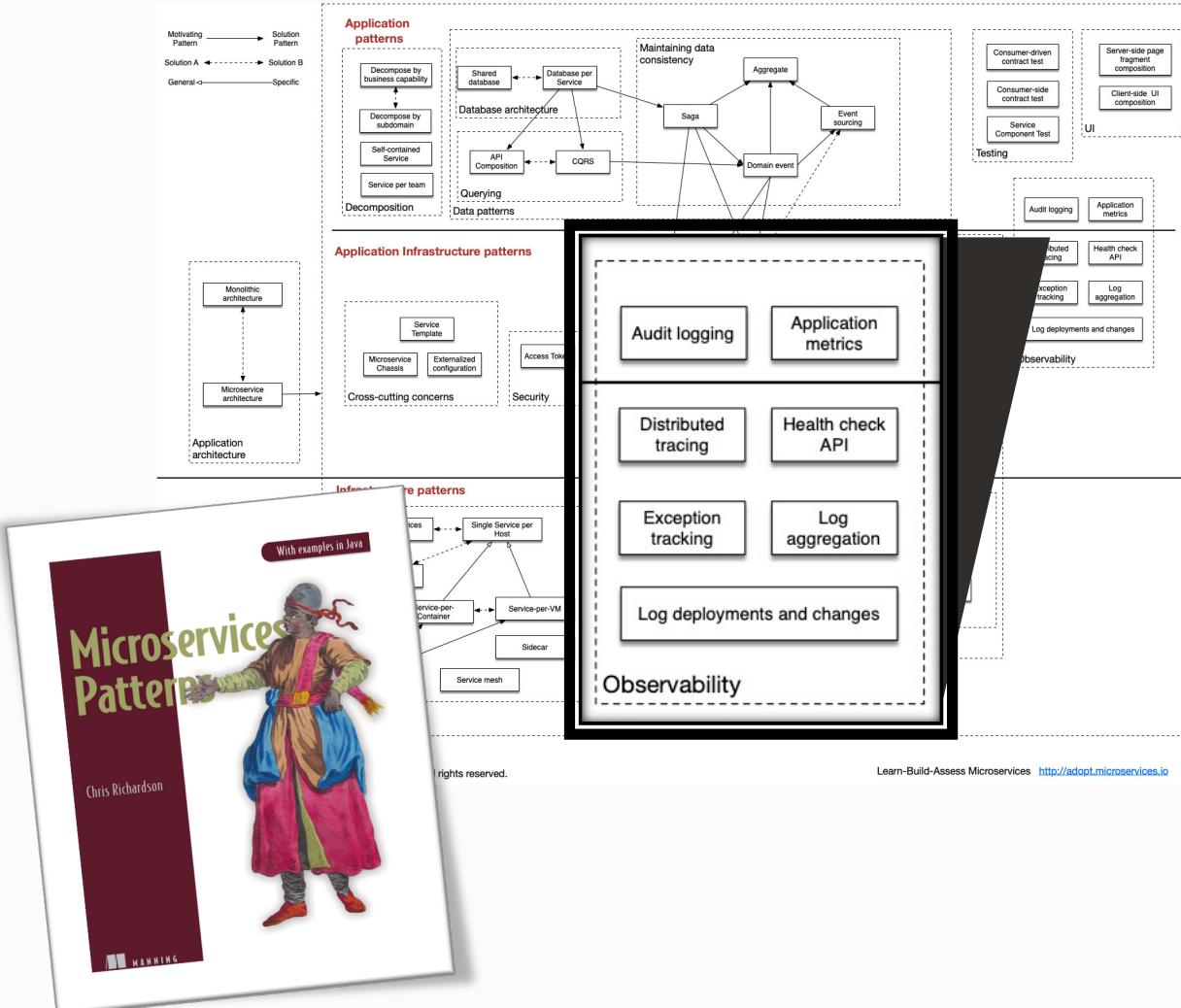
5. SERVICE PROXY, DISCOVERY, & MESH

- CoreDNS is a fast and flexible tool that is useful for service discovery
- Envoy and Linkerd each enable service mesh architectures
- They offer health checking, routing, and load balancing



Observability Patterns

Microservice Patterns

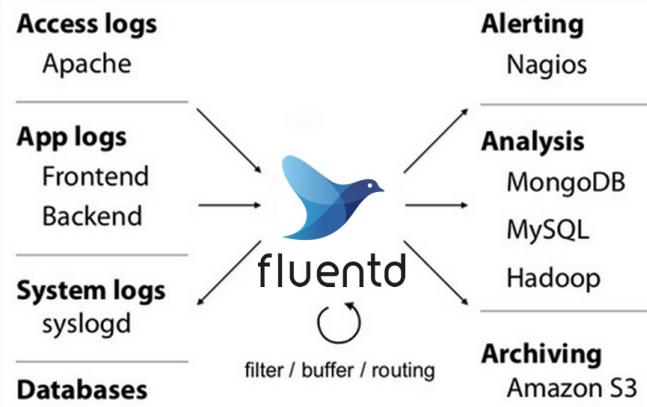
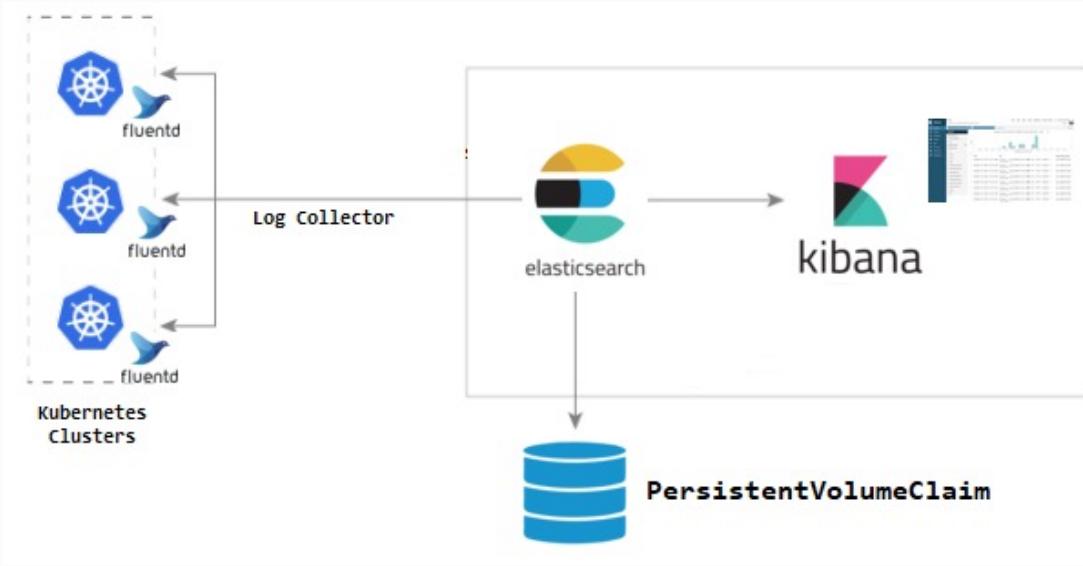


According to microservices architecture and modern systems design there are **7 patterns** to help understand the behavior of an application and troubleshoot problems:

- Log aggregation** - aggregate application logs
- Application metrics** - instrument a service's code to gather statistics about operations
- Audit logging** - record user activity in a database
- Distributed tracing** - instrument services with code that assigns each external request an unique identifier that is passed between services. Record information (e.g. start time, end time) about the work (e.g. service requests) performed when handling the external request in a centralized service
- Exception tracking** - report all exceptions to a centralized exception tracking service that aggregates and tracks exceptions and notifies developers.
- Health check API** - service API (e.g. HTTP endpoint) that returns the health of the service and is intended to be pinged, for example, by a monitoring service
- Log deployments and changes** – CI/CD focus.

Log aggregation

How to understand the behavior of an application and troubleshoot problems?



Oracle Cloud Infrastructure Logging

The screenshot shows the Oracle Cloud Infrastructure Logging interface. At the top, it features the Oracle Cloud Infrastructure logo and the 'fluentd' and 'cloudevents' logos. Below this is a search bar with fields for 'FILTER BY FIELD OR TEXT SEARCH' (containing 'type') and 'SELECT LOGS TO SEARCH' (containing 'uitest'). There are buttons for 'Reset Search', 'Create Service Connector', and a 'Search' button. On the right, there are 'AUTOREFRESH' settings (set to 'OFF') and an 'Export Log Data (JSON)' button. The main area displays a chart titled 'Number of Log Events Per Minute' showing data from 14:22 to 17:22. Below the chart is a section titled 'Log Data' with two log entries from September 28, 2020, at 17:18:57 UTC, both reporting failed delivery to OSS with exception messages.

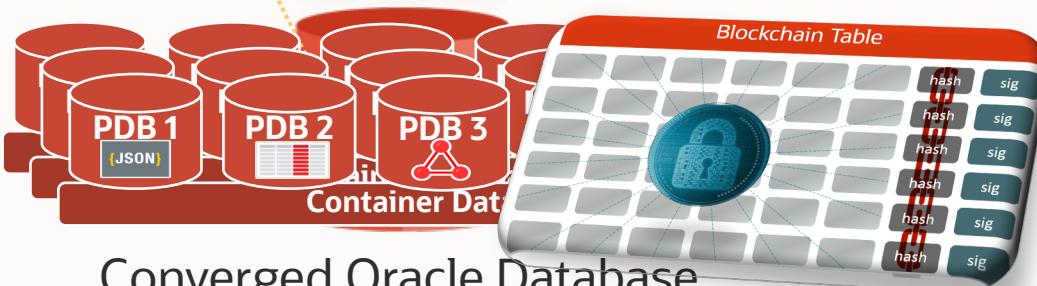


Audit Logging

Keep tracking of actions a user has recently performed (Customer Support, Compliance, etc)



Convergence | Consolidation | Virtualization

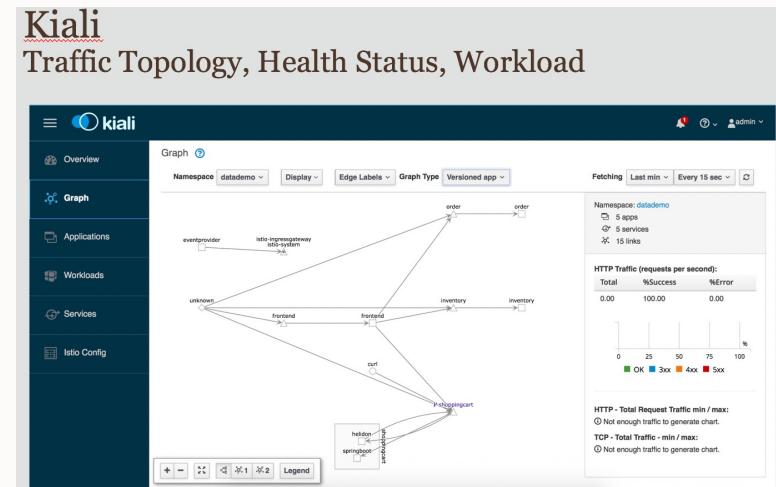


Converged Oracle Database



Observability and Distributed Tracing

- Observability: distributed tracing/logging across services
 - Trace, Span, SpanContext, SpanId
- OpenTelemetry: OpenTracing + OpenCensus
- Kiali, Jaeger, and Grafana stop at the edge of the DB
 - Looking into end-to-end distributed tracing



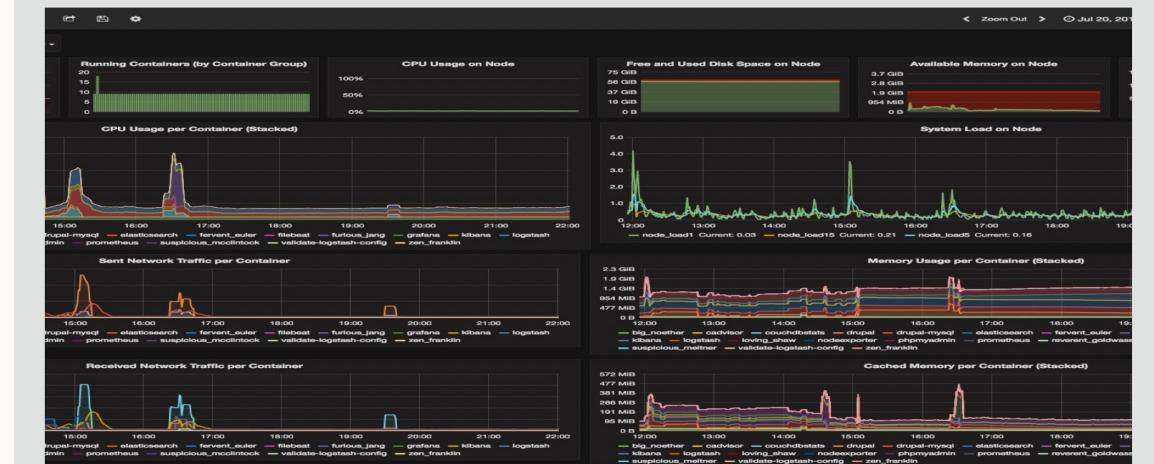
OpenTelemetry - Jaeger

Tracing across full call path of activity
Trace from WebLogic to Helidon MP to Helidon SE (soon to Oracle DB and AQ as well)...

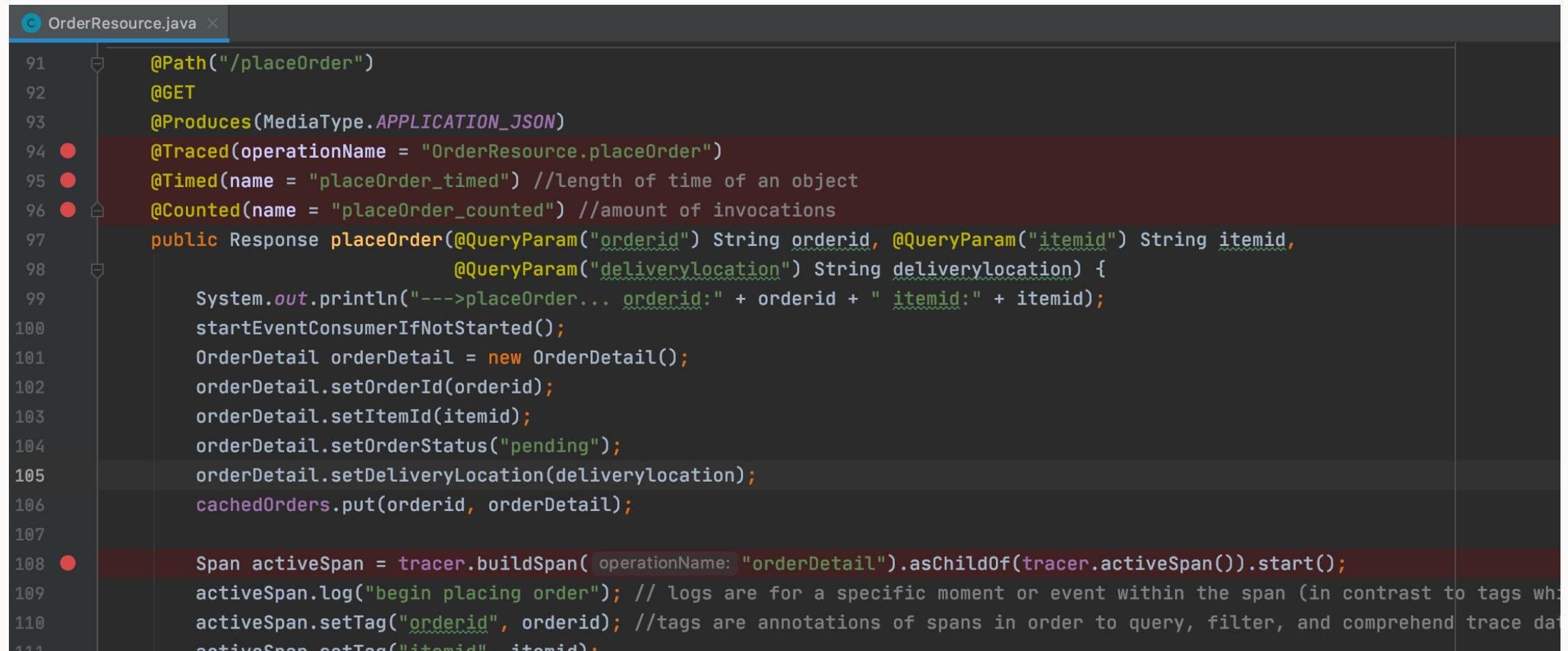


Grafana

Analytics, monitoring, alerting for time-series data



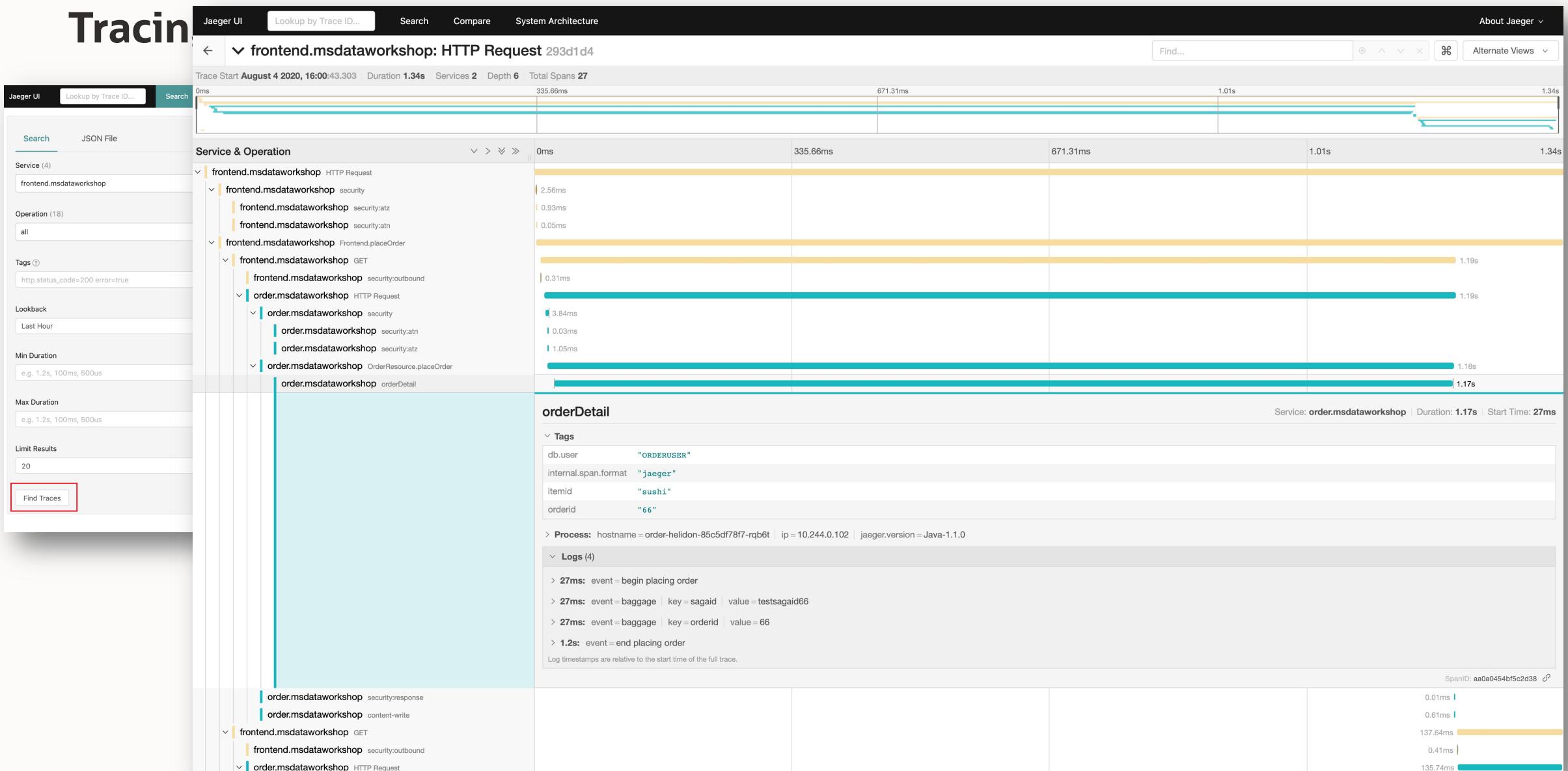
Tracing Using Jaeger



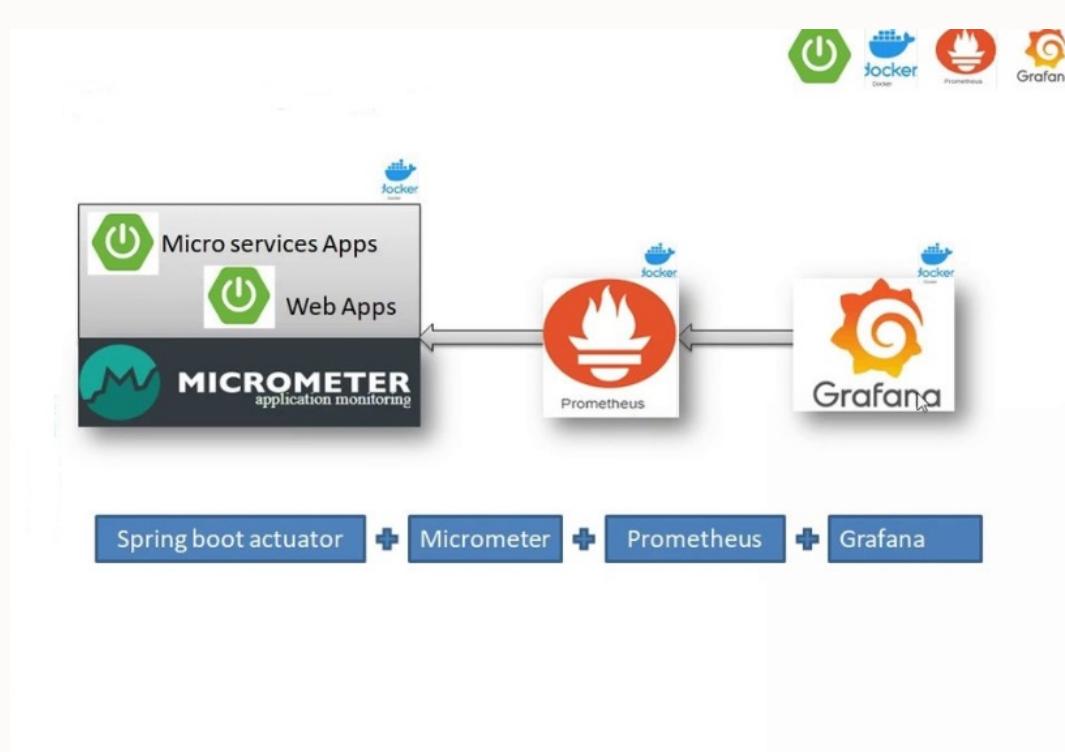
The screenshot shows a Java code editor with the file `OrderResource.java` open. The code defines a REST endpoint for placing an order. Several annotations are used for tracing:

- `@Path("/placeOrder")`: Path annotation for the endpoint.
- `@GET`: Method annotation for the GET request.
- `@Produces(MediaType.APPLICATION_JSON)`: Produces annotation for JSON output.
- `@Traced(operationName = "OrderResource.placeOrder")`: Tracing annotation with operation name.
- `@Timed(name = "placeOrder_timed")`: Timed annotation for the length of time of an object.
- `@Counted(name = "placeOrder_counted")`: Counted annotation for the amount of invocations.
- `Span activeSpan = tracer.buildSpan(operationName: "orderDetail").asChildOf(tracer.activeSpan()).start();`: Starts a new span for the `orderDetail` operation.
- `activeSpan.log("begin placing order");`: Logs an event within the span.
- `activeSpan.setTag("orderid", orderid);`: Sets a tag for the span.
- `activeSpan.setTag("itemid", itemid);`: Sets another tag for the span.

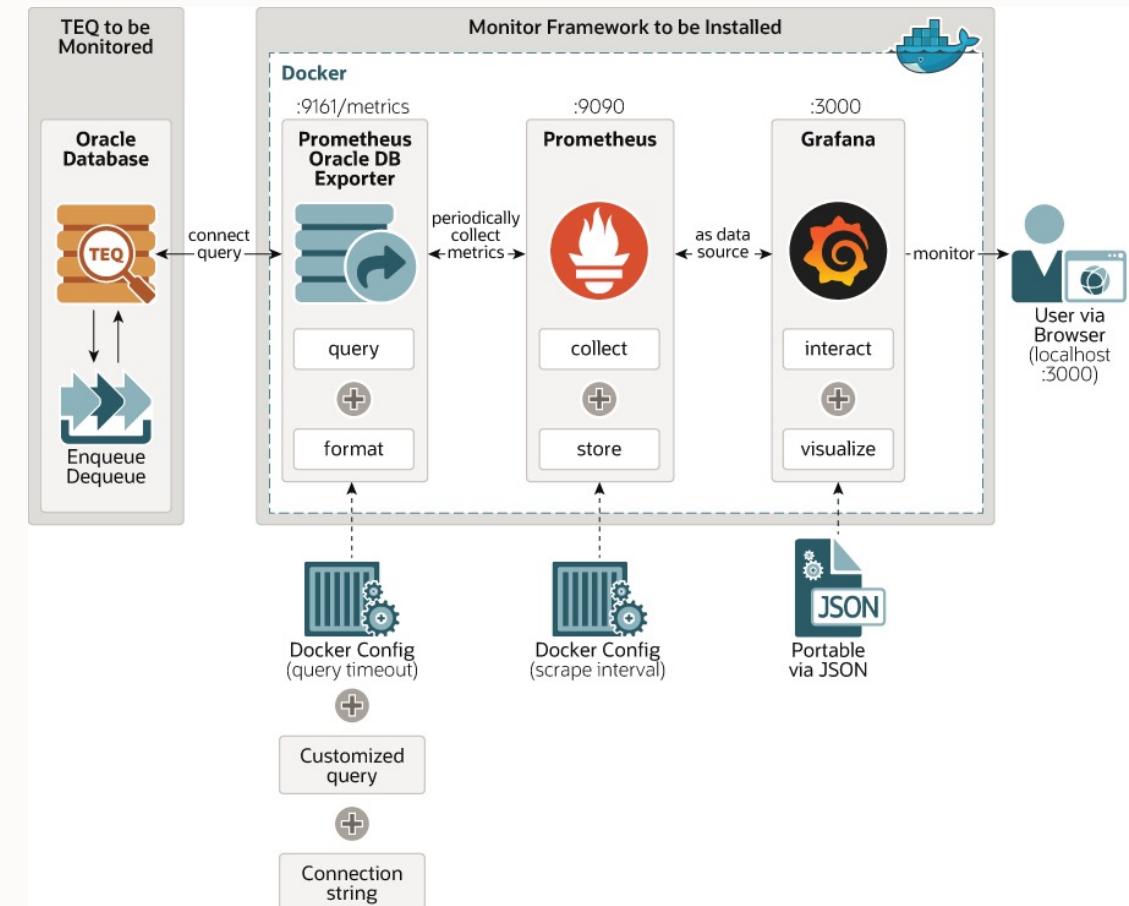
Tracing



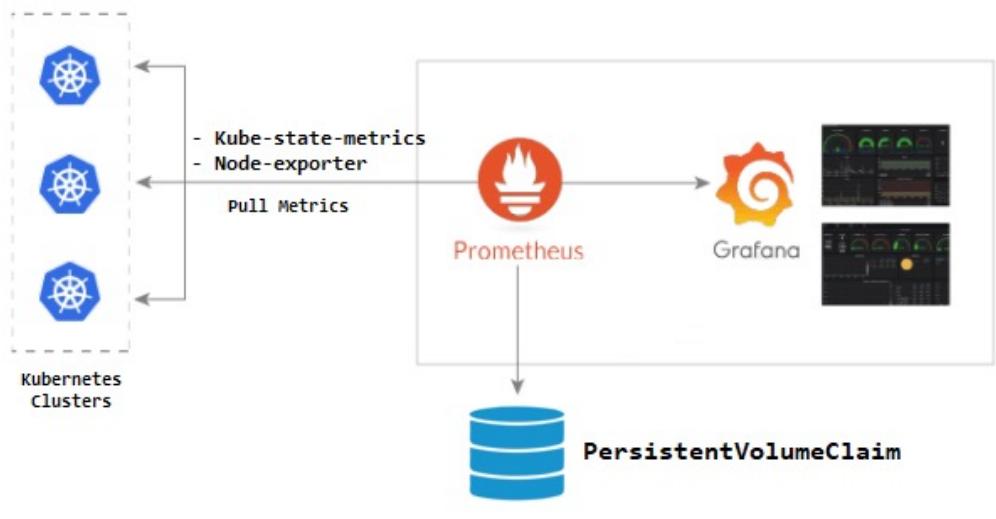
Monitoring Application Microservices & Events (Streams)



Monitoring Oracle DB Transaction Event Queue



Monitoring Infrastructures

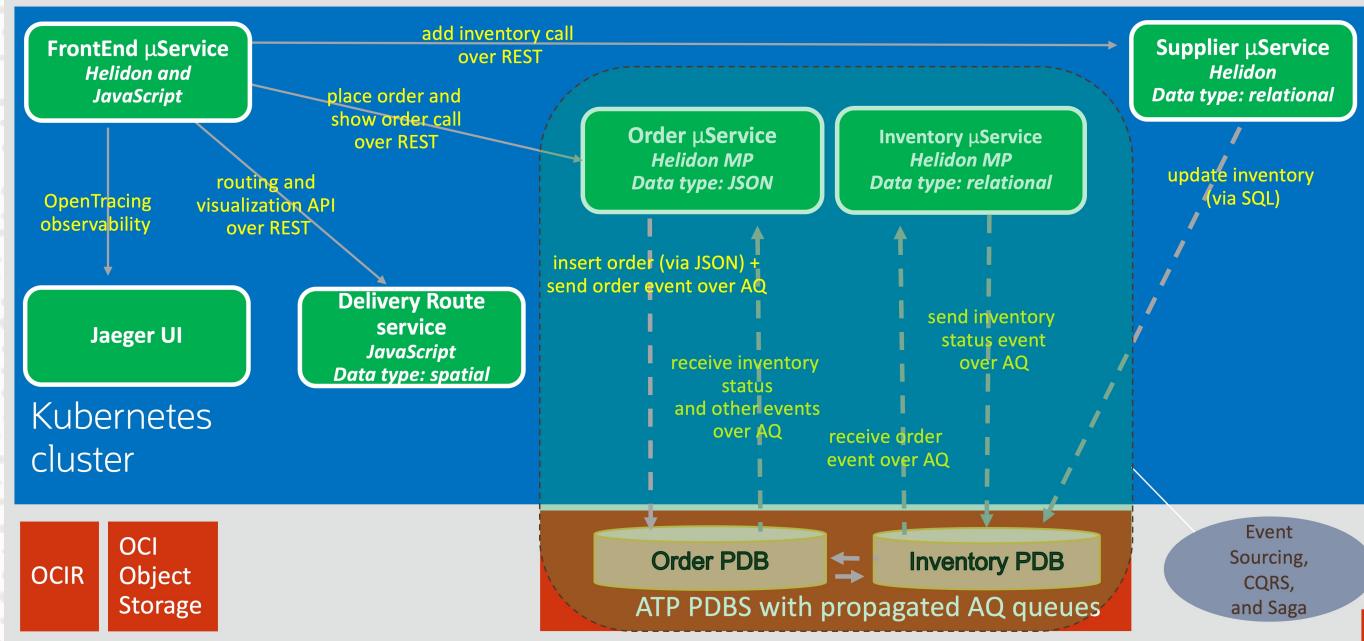


Building Microservices with Oracle Converged Database Workshop

Modernize and simplify your data-driven applications with microservices architecture using Database Cloud and Helidon on OCI.

Workshop length: 2 hours

“GrabDish” Food Order App Architecture



Workshop Outline

- Setup OCI, OKE and ATP
- Build and deploy microservices
- Setup database connection secrets, tables and AQ messaging
- Data-centric microservices walkthrough with Helidon MP
- Polygot Microservices
- Scaling the Application
- Create an APEX App to make sense of the data

<https://bit.ly/simplymicroservices>



Paulo Alberto Simões

Developer Evangelist,
Data-Driven Microservices with Converged Database at Oracle

Mentor, [Oracle for Startups](#) | Ambassador, [Oracle Academy](#)
[Cloud-Native Computing Foundation](#) Ambassador

Stay Connected

LinkedIn, Twitter, Instagram, GitHub, Clubhouse, twitch.tv, dev.to : **pasimoes**



ORACLE

Our mission is to help people see data in new ways,
discover insights, unlock endless possibilities.