



Welcome to the NY Oracle  
User Group Webinar

**Kubernetes, Containers,  
Microservices + Mobile with  
Oracle Database = Modern  
App Dev**

We will start in a few minutes





ORACLE

Mark Nelson

Developer Evangelist  
Oracle Database



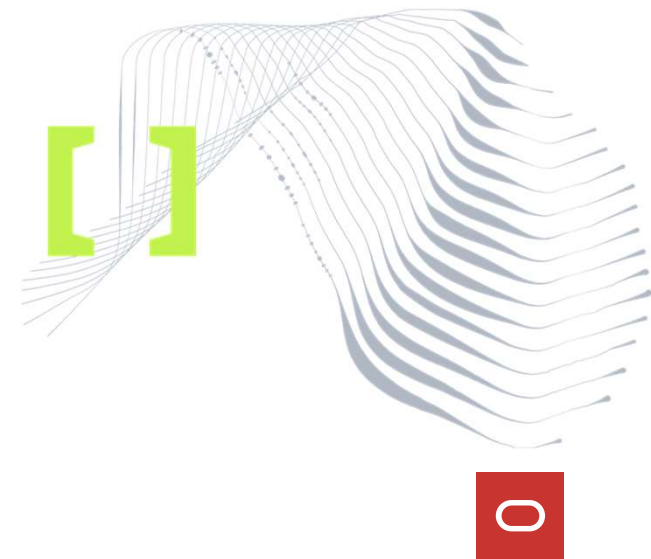
Andy Tael

Developer Evangelist  
Oracle Database

# Agenda

1. Introduction – Kubernetes, containers, microservices, mobile
2. Oracle Backend for Spring Boot
3. Feature drill down
4. Mobile app dev
5. Ways to learn more

Ask questions any time in the Q&A or chat.  
There will be dedicated Q&A time at the end.



# Introduction



# Kubernetes and containers

Coming soon – Kubernetes for DBAs –  
free training – subscribe to our  
YouTube channel to be notified  
<http://bit.ly/convergeddatabase>

Kubernetes has emerged as the *de facto* platform for modern applications:

- Containers provide benefits like portability, consistency, externalized configuration and secrets, automated builds, workload isolation and easier scalability
- Kubernetes makes it easy (well, possible!) to run containers at scale
- Kubernetes is widely regarded to be complex, but it is also a very rapidly evolving space with lots of innovation occurring

## D2iQ survey:

*“77% of organizations cited Kubernetes as a central part of their digital strategy”*

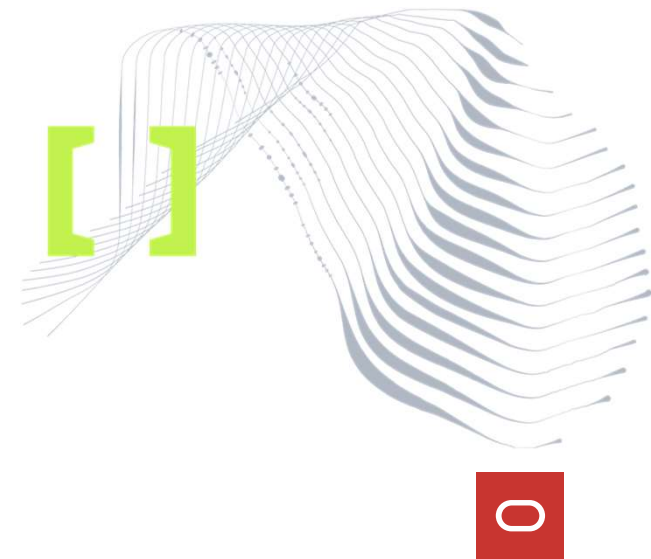
Oracle is now supporting the Oracle Database (and other products) for *production use* in Kubernetes and providing “operators” to simplify lifecycle management



# Why microservices?

Microservices architecture is a popular way to build modern, modular applications:

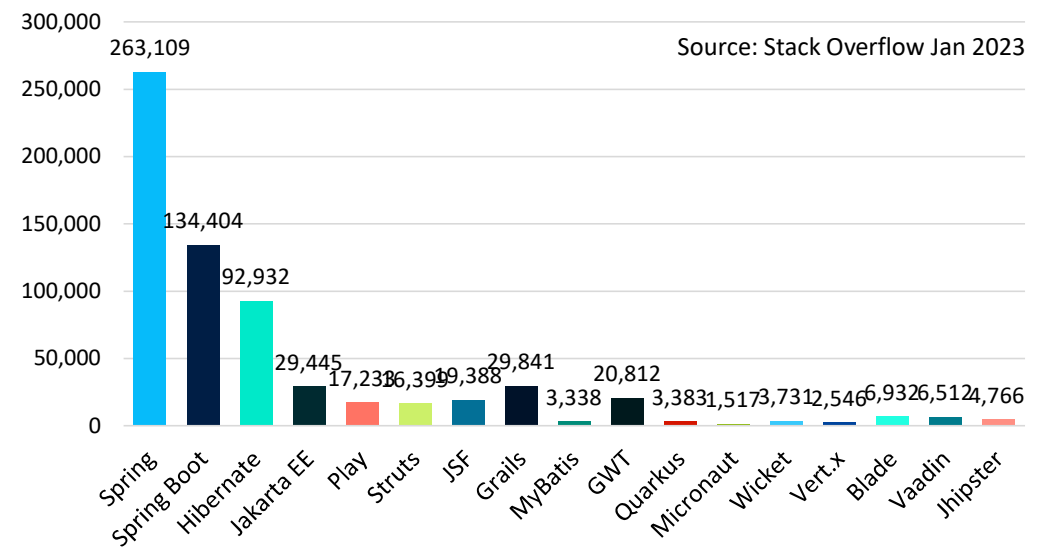
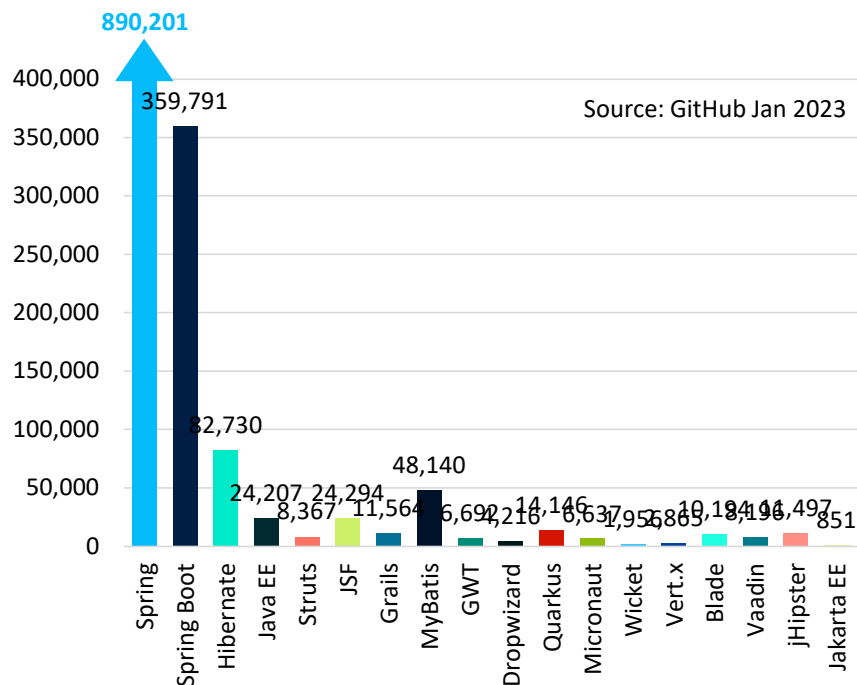
1. Applications are structured as a collection of independently deployable services that are loosely coupled, focus on a single business domain and are owned by small teams
2. Promises to make large, complex applications quicker and more reliable to deliver, maintain and evolve
3. “Monolithic architecture” is often cited as the opposite
4. Like any new architectural style, microservices is not a silver bullet, it introduces its own complexities and trade offs, but enough projects have been run for patterns to start to emerge



# Spring Boot is the most popular microservices platform

A lot of developers today use frameworks - **Spring Boot** and Spring Framework are the most popular, *by far*, based on several measurements (job advertisements, stack overflow posts, GitHub repos, Google search trends, etc.)

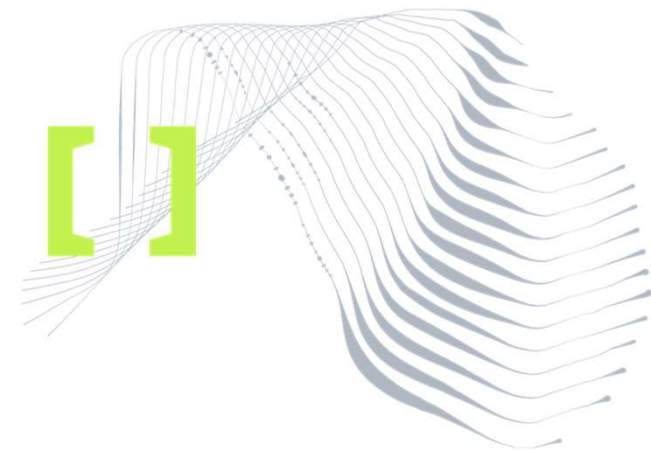
We can write less code, and get more done and there's a wealth of knowledge online



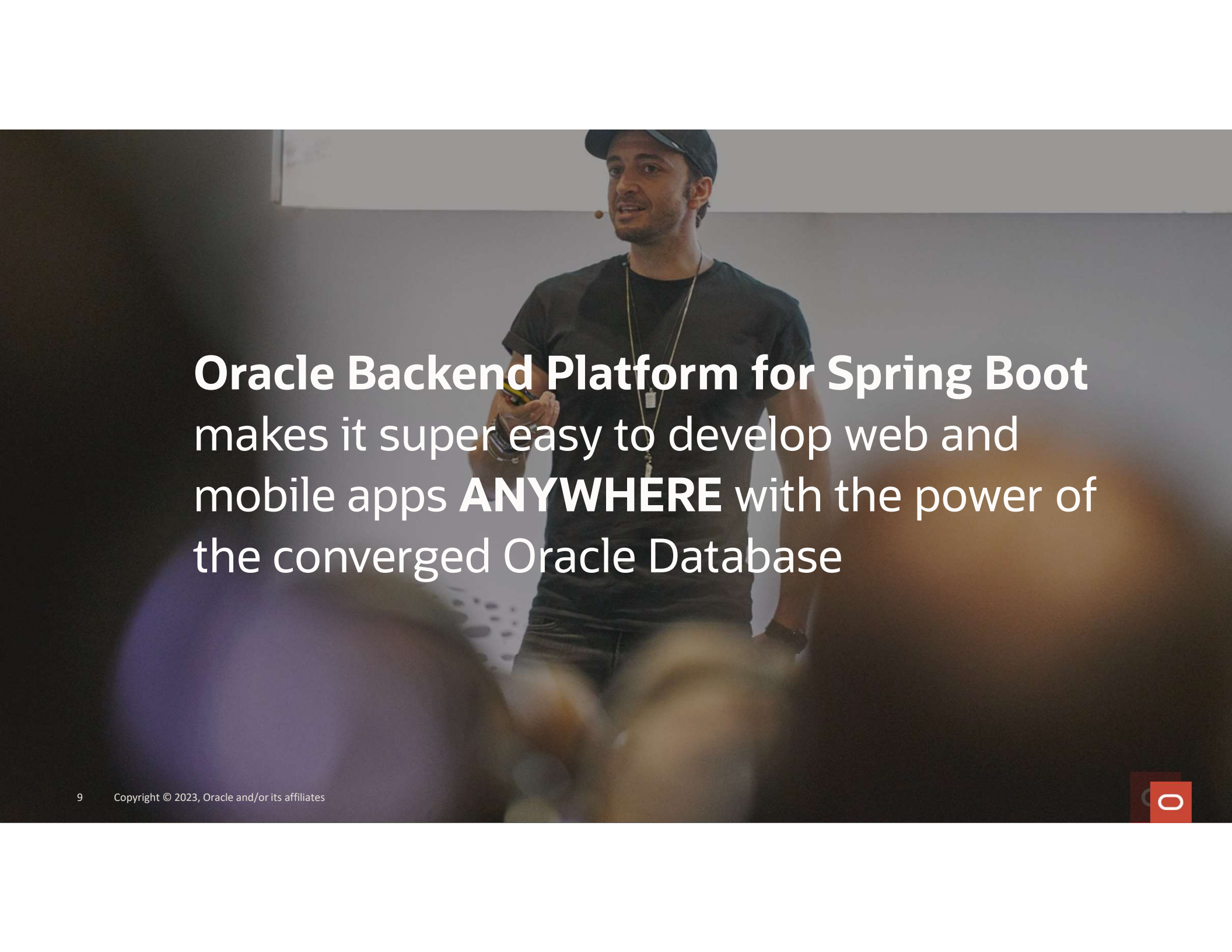
# What about mobile?

We say “mobile”, but we really mean mobile *and* web front end applications:

1. Mobile has emerged as a ubiquitous platform for application UI's
2. It's not just for B2C – there are huge B2E, B2B and IOT segments in mobile as well, especially in certain industries (healthcare and transportation for example)
3. The “**backend as a service**” pattern is very popular in mobile app dev.  
It allows the developer for focus on the app, and just consume services from a backend using APIs without needing to worry about how the backed is provisioned, managed, scaled, etc.
4. The two main SDK approaches are ecosystem-native (e.g. iOS, Android) and cross platform (e.g. React Native, Flutter)





A man wearing a black t-shirt, a black cap, and a lanyard is speaking at a conference. He is holding a small object in his right hand. The background is a blurred audience and a stage setting.

**Oracle Backend Platform for Spring Boot**  
makes it super easy to develop web and  
mobile apps **ANYWHERE** with the power of  
the converged Oracle Database



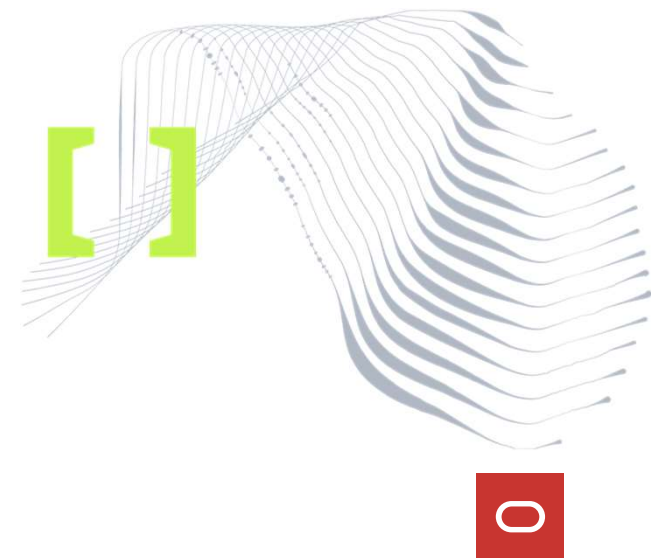
# Applications need data – and that’s where we come in!

Microservices applications and mobile/web applications all need data:

- Oracle Database 23c is all about making life **easy for developers!**
- At Oracle, we are producing new offerings, SDKs, APIs, integrations to make it easy for developers to access the data in their organizations Oracle Databases (not just 23c, the older ones too!)

## Oracle Backend for Spring Boot (and Parse Platform)

- Implements the “backend as a service” pattern for Spring Boot & mobile  
*Microservices and mobile – better together!*
- Available now as a Developer Preview in OCI Marketplace
- Also available for on-premises and Azure



## Oracle Backend for Spring Boot feature/function – microservices

Microservices/Spring Boot developers want..	We meet this need with..
Runtime environment for services	Containers and Kubernetes
Easy access to database, messaging, etc.	Spring Boot Starters for Oracle technologies
Service discovery, client-side load balancing, fault tolerance	Spring Eureka Service Registry, Resilience4j
API gateway, management	APISIX API Gateway and Dashboard
Manage data consistency across microservices	Oracle Transaction Manager for Microservices
Workflow, orchestration, choreography	Netflix OSS Conductor
Externalize configuration from applications	Spring Config Server
Observability	Prometheus, Grafana, Jaeger, OpenTelemetry
Secure secrets management	Hashicorp (and OCI) Vault
Scale on demand	OKE and ADB autoscaling
Developer tooling	CLI and VS Code extension





## Oracle Backend for Spring Boot feature/function – mobile

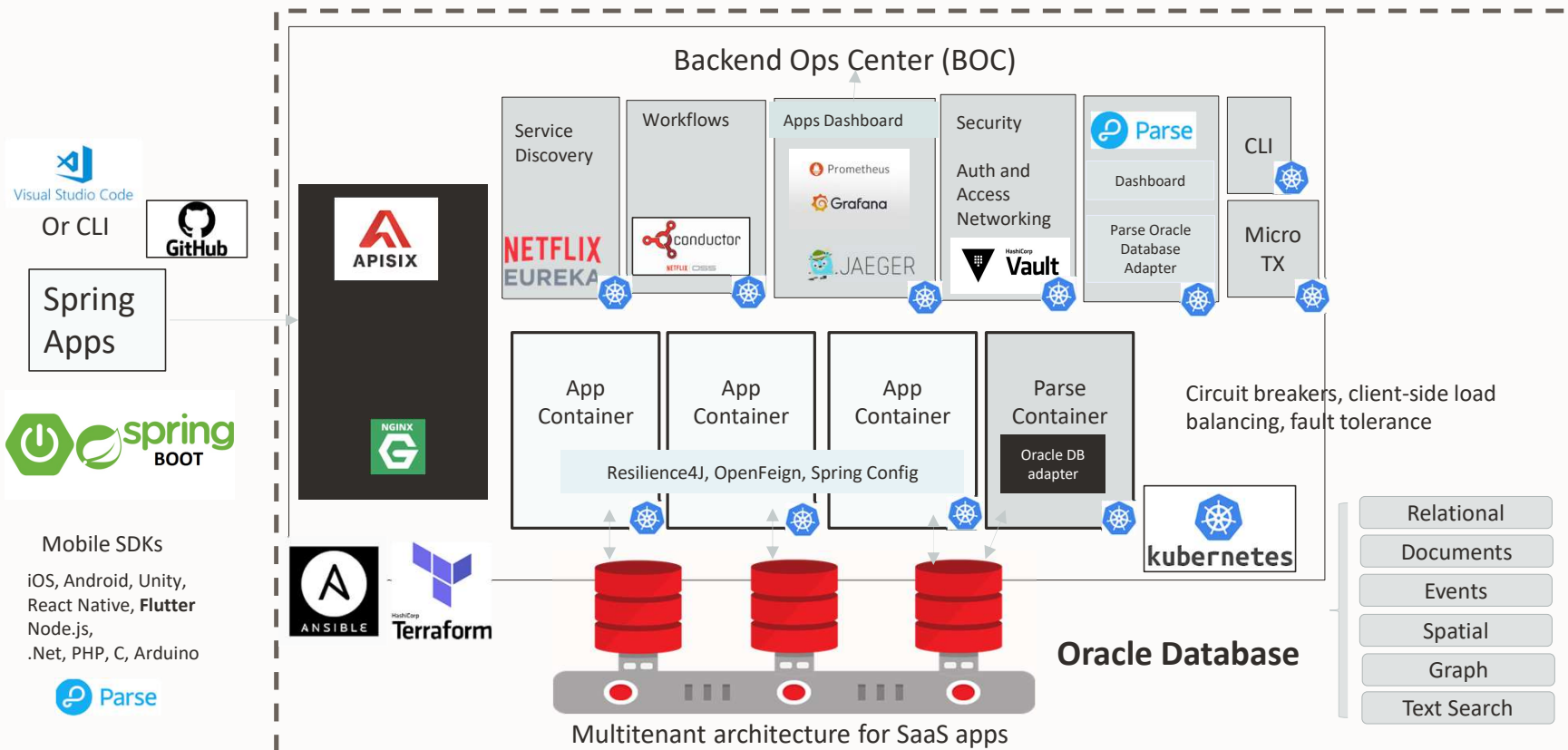
Mobile developers want..	We meet this need with..
Store/retrieve data (usually JSON documents)	
On-device offline storage	A new <i>Oracle Storage Adapter for Parse Platform</i> which lets you use Oracle Autonomous Database as the backend data store
Identity/security	
Files (too large to bundle into app, distribute by CDN)	
Caching	Can be used by any existing Parse app with zero code changes
Location/geospatial	
Config (push to device)	All existing Parse SDKs can be used with Oracle backend – iOS and Android phone/tablet/computer/TV/auto, JavaScript, React, React Native, Flutter, Unity .Net, PHP, C, Arduino, plus REST and GraphQL APIs
Cloud functions (run on server side)	
Live query	
GraphQL	
Push notifications	



# Oracle Backend for Spring Boot (and Parse Platform)

Spring Boot for Microservices Development and Deployment

## Oracle Backend for Spring Boot



Let's drill down into some of these features...

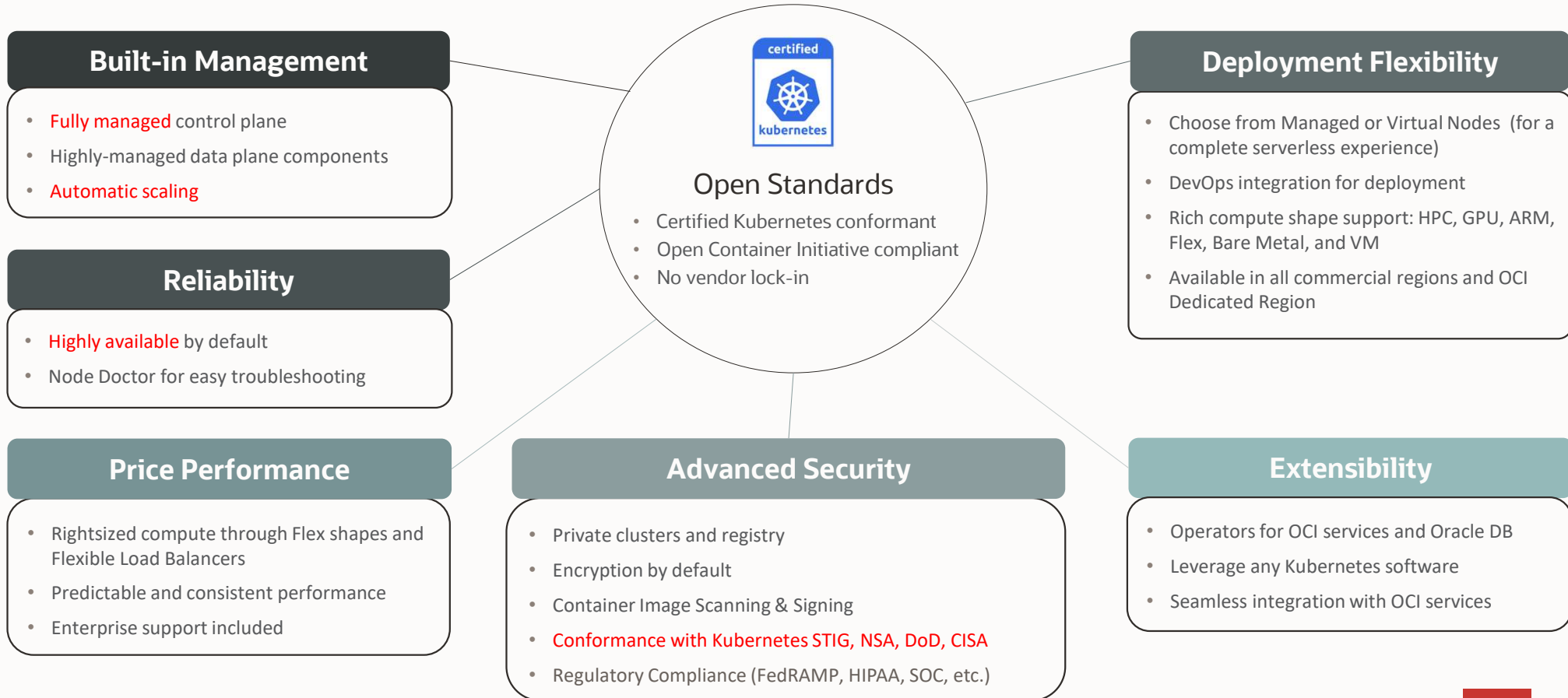
## Oracle Backend for Spring Boot feature/function – microservices

Microservices/Spring Boot developers want..	We meet this need with..
Runtime environment for services	<b>Containers and Kubernetes</b>
Easy access to database, messaging, etc.	Spring Boot Starters for Oracle technologies
Service discovery, client-side load balancing, fault tolerance	Spring Eureka Service Registry, Resilience4j
API gateway, management	APISIX API Gateway and Dashboard
Manage data consistency across microservices	Oracle Transaction Manager for Microservices
Workflow, orchestration, choreography	Netflix OSS Conductor
Externalize configuration from applications	Spring Config Server
Observability	Prometheus, Grafana, Jaeger, OpenTelemetry
Secure secrets management	Hashicorp (and OCI) Vault
Scale on demand	OKE and ADB autoscaling
Developer tooling	CLI and VS Code extension



# Runtime Environment for Services (OCI OKE, Azure & On-Prem)

## Container and Kubernetes





## Oracle Backend for Spring Boot feature/function – microservices

Microservices/Spring Boot developers want..	We meet this need with..
Runtime environment for services	<b>Containers and Kubernetes</b>
Easy access to database, messaging, etc.	Spring Boot Starters for Oracle technologies
Service discovery, client-side load balancing, fault tolerance	Spring Eureka Service Registry, Resilience4j
API gateway, management	APISIX API Gateway and Dashboard
Manage data consistency across microservices	Oracle Transaction Manager for Microservices
Workflow, orchestration, choreography	Netflix OSS Conductor
Externalize configuration from applications	Spring Config Server
Observability	Prometheus, Grafana, Jaeger, OpenTelemetry
Secure secrets management	Hashicorp (and OCI) Vault
Scale on demand	OKE and ADB autoscaling
Developer tooling	CLI and VS Code extension



## Oracle Spring Boot Starters

We are publishing **Spring Boot Starters** (for Spring Boot 2.x and 3.x) that make it easy to use Oracle technologies like:

- Universal Connection Pooling
- Database wallet
- AQ/TxEventQ using JMS APIs
- TxEventQ using Kafka APIs
- JSON collection using SODA or Mongo APIs
- Graph data using PGQL or Neo4J APIs
- Text, Spatial, ML, and **many more coming!**

Spring Boot Starters are libraries that Spring Boot developers can easily add to their applications:

- They provide any necessary dependencies
- And they may also handle auto-configuration and/or injection of various “beans” into the application
- For example, the Oracle Spring Boot Starter will inject a JDBC connection and a JMS Connection Factory auto-configured and able to share the same database transaction



## Example – using Oracle Spring Boot Starter for AQ/JMS

```
<dependency>  
  <groupId>com.oracle.database.spring</groupId>  
  <artifactId>oracle-spring-boot-starter-aqjms</artifactId>  
  <version>2.7.7</version>  
</dependency>
```

Add a dependency for the Oracle Spring Boot Starter for AQ JMS and TxEventQ

```
oracle.aq.username=pdbadmin  
oracle.aq.password=Welcome123  
oracle.aq.url=jdbc:oracle:thin:@//172.17.0.2:1521/pdb1
```

Add details for your JMS connection (or put them in the Config Server instead!)

## Example – using Oracle Spring Boot Starter for AQ/JMS

```
@Component
public class Receiver {
    @JmsListener(destination = "mailbox", containerFactory = "myFactory")
    public void receiveMessage>Email email) {
        System.out.println("Received <" + email + ">");
    }
}
```

Oracle Spring Boot Starter for AQ JMS will automatically inject both a JDBC database connection and a JMS connection factory into your service

Use the JmsListener annotation to name your queue/topic – the CLI will automatically create it for you when you deploy your service

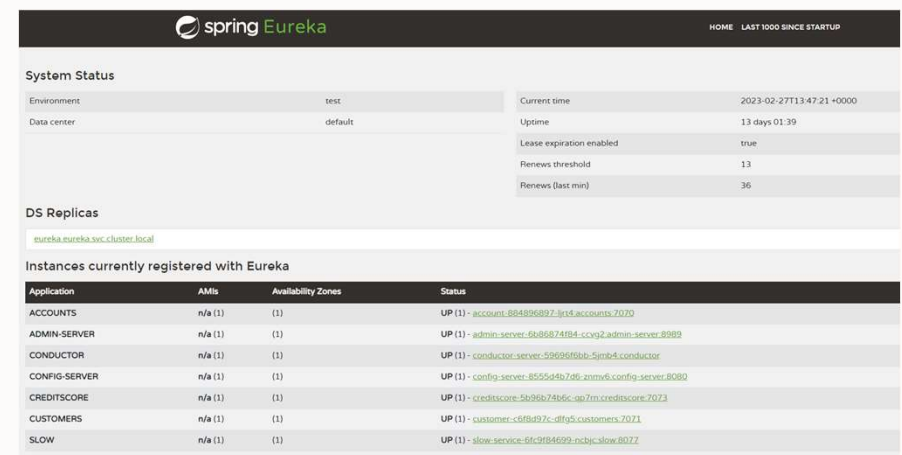
## Oracle Backend for Spring Boot feature/function – microservices

Microservices/Spring Boot developers want..	We meet this need with..
Runtime environment for services	<b>Containers and Kubernetes</b>
Easy access to database, messaging, etc.	Spring Boot Starters for Oracle technologies
Service discovery, client-side load balancing, fault tolerance	Spring Eureka Service Registry, Resilience4j
API gateway, management	APISIX API Gateway and Dashboard
Manage data consistency across microservices	Oracle Transaction Manager for Microservices
Workflow, orchestration, choreography	Netflix OSS Conductor
Externalize configuration from applications	Spring Config Server
Observability	Prometheus, Grafana, Jaeger, OpenTelemetry
Secure secrets management	Hashicorp (and OCI) Vault
Scale on demand	OKE and ADB autoscaling
Developer tooling	CLI and VS Code extension



## Spring Eureka Service Registry

- Spring Service Registry maintains a list of all active/healthy instances of a service
- Services can find other services they need by looking them up in the registry – thereby eliminating the need to know the other service's address at compile or deploy time
- The Spring discovery client provides client-side load balancing and fault tolerance capabilities
- The registry is also used by other infrastructure and operations components to discover information about running services



The screenshot displays the Spring Eureka web interface. At the top, there's a navigation bar with the 'spring Eureka' logo and a 'HOME' link. Below this, the 'System Status' section provides details about the environment (test), data center (default), current time (2023-02-27T13:47:21+0000), uptime (13 days 01:39), lease expiration enabled (true), renewal threshold (13), and renewal interval (36). The 'DS Replicas' section shows a single replica at 'eureka.eureka.svc.cluster.local'. The 'Instances currently registered with Eureka' section contains a table with columns for Application, AMIs, Availability Zones, and Status.

Application	AMIs	Availability Zones	Status
ACCOUNTS	n/a (1)	(1)	UP (1) - account-804826897-3rtd.accounts.7070
ADMIN-SERVER	n/a (1)	(1)	UP (1) - admin-server-5b86874f84-cvrg2.admin-server.8989
CONDUCTOR	n/a (1)	(1)	UP (1) - conductor-server-59696f0bb-5ymb4.conductor
CONFIG-SERVER	n/a (1)	(1)	UP (1) - config-server-8555d4b7d8-2nmv6.config-server.8080
CREDITSCORE	n/a (1)	(1)	UP (1) - creditscore-5b96b74b6c-gp7m.creditscore.7073
CUSTOMERS	n/a (1)	(1)	UP (1) - customer-c6f0497c-dffg5.customers.7071
SLOW	n/a (1)	(1)	UP (1) - slow-service-6fc9f04609-mcbjc.slow.8077





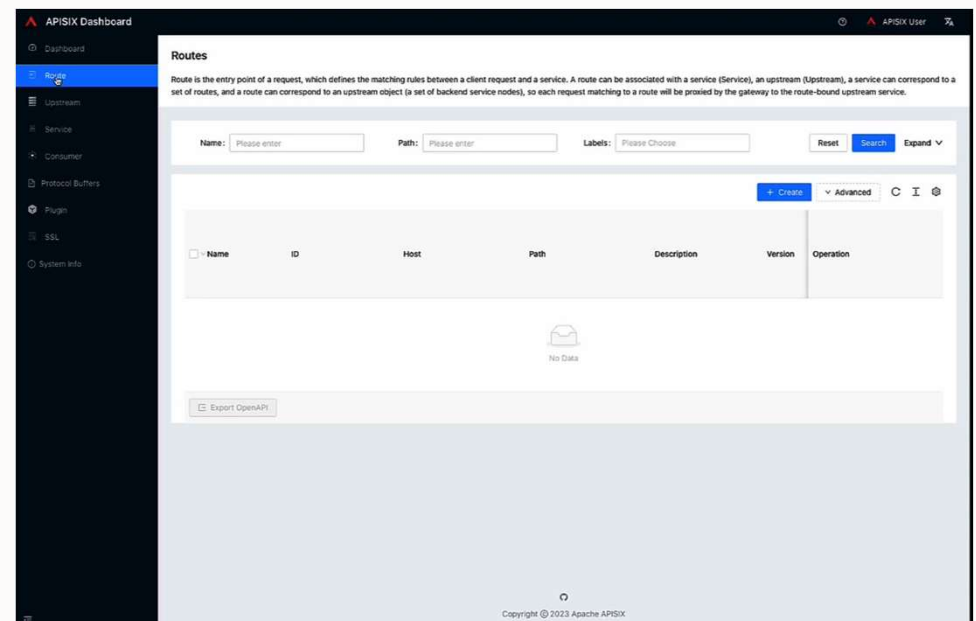
## Oracle Backend for Spring Boot feature/function – microservices

Microservices/Spring Boot developers want..	We meet this need with..
Runtime environment for services	<b>Containers and Kubernetes</b>
Easy access to database, messaging, etc.	Spring Boot Starters for Oracle technologies
Service discovery, client-side load balancing, fault tolerance	Spring Eureka Service Registry, Resilience4j
API gateway, management	APISIX API Gateway and Dashboard
Manage data consistency across microservices	Oracle Transaction Manager for Microservices
Workflow, orchestration, choreography	Netflix OSS Conductor
Externalize configuration from applications	Spring Config Server
Observability	Prometheus, Grafana, Jaeger, OpenTelemetry
Secure secrets management	Hashicorp (and OCI) Vault
Scale on demand	OKE and ADB autoscaling
Developer tooling	CLI and VS Code extension



# APISIX API Gateway

- The API Gateway is used to expose services outside the Kubernetes cluster
- It provides features like traffic management, transformation, auditing, authentication, and observability
- The API Gateway can also control what methods, protocols, headers, cookies, etc., are required and/or allowed
- It acts as a central place to manage which APIs (services) are accessible to clients outside the runtime environment
- It is integrated with Spring Service Registry and Kubernetes for service discovery





## Oracle Backend for Spring Boot feature/function – microservices

Microservices/Spring Boot developers want..	We meet this need with..
Runtime environment for services	<b>Containers and Kubernetes</b>
Easy access to database, messaging, etc.	Spring Boot Starters for Oracle technologies
Service discovery, client-side load balancing, fault tolerance	Spring Eureka Service Registry, Resilience4j
API gateway, management	APISIX API Gateway and Dashboard
Manage data consistency across microservices	Oracle Transaction Manager for Microservices
Workflow, orchestration, choreography	Netflix OSS Conductor
Externalize configuration from applications	Spring Config Server
Observability	Prometheus, Grafana, Jaeger, OpenTelemetry
Secure secrets management	Hashicorp (and OCI) Vault
Scale on demand	OKE and ADB autoscaling
Developer tooling	CLI and VS Code extension



# How is data consistency handled today in microservices?

## Manual reconciliation

- Inconsistent data view for a period
- Potential financial losses due to loss of business and customer dissatisfaction
- **Resource intensive task, which increases cost of operations**

## Developers building transaction management logic in apps

- Requires developers to have advanced skills
- **Takes valuable time away from app developers**
- Can be error prone; increases testing complexity
- Increases time and cost to market

## Use of existing Transaction Managers

- Almost all solutions are for Java apps
- **Don't provide loosely coupled / asynchronous consistency for scale**
- Lack enterprise capabilities; e.g. **integration with K8S ecosystem**
- **Not optimized for Oracle Database**

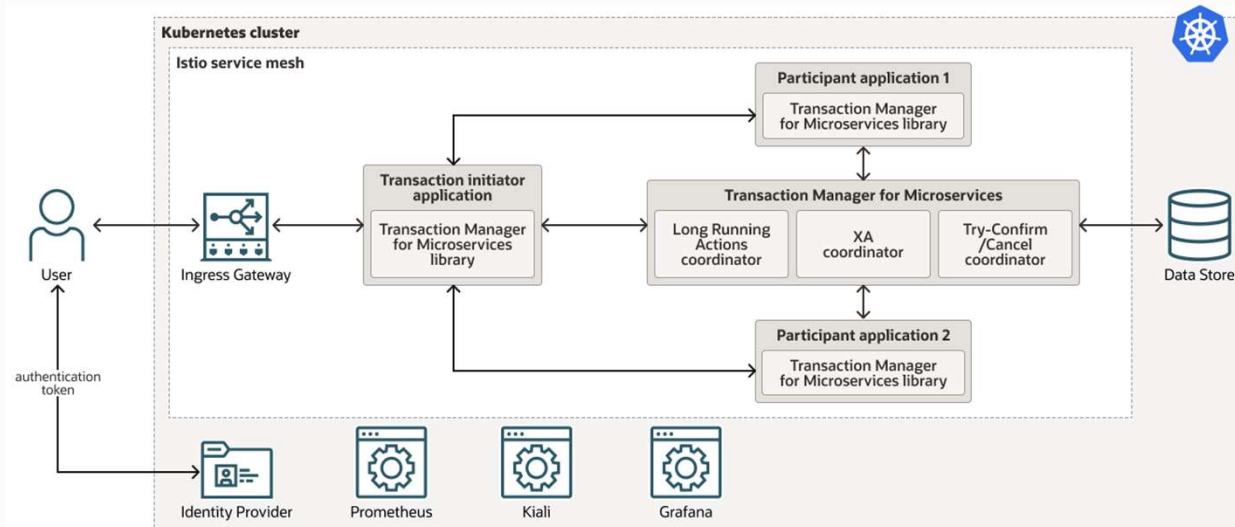
There is a need for an enterprise solution to address data consistency issues for Microservices apps



## Oracle Transaction Manager for Microservices (MicroTx)

Enables developers to use distributed transactions to ensure data consistency across microservices based apps

- Using heterogenous resource managers
- Utilizing many other existing services/applications
  - Developed in multiple programming languages
  - Implements the Eclipse Microprofile LRA coordinator for Microservices with REST
  - Deployed in Kubernetes



# Oracle Transaction Manager for Microservices (MicroTx)

Two main components

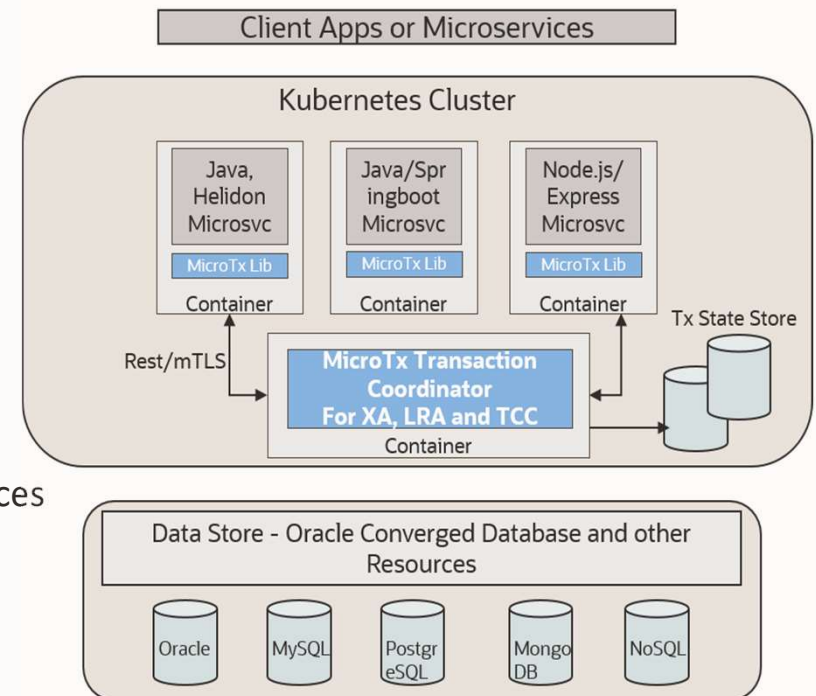
- Transaction Coordinator
- MicroTx Library
  - one for each programming language
  - utilized by each application microservice

Transaction coordinator is a microservice

- Deployed along with application microservices

REST API based communication

- No additional requirements imposed on application microservices
- Transaction state stored in etcd or Oracle Database





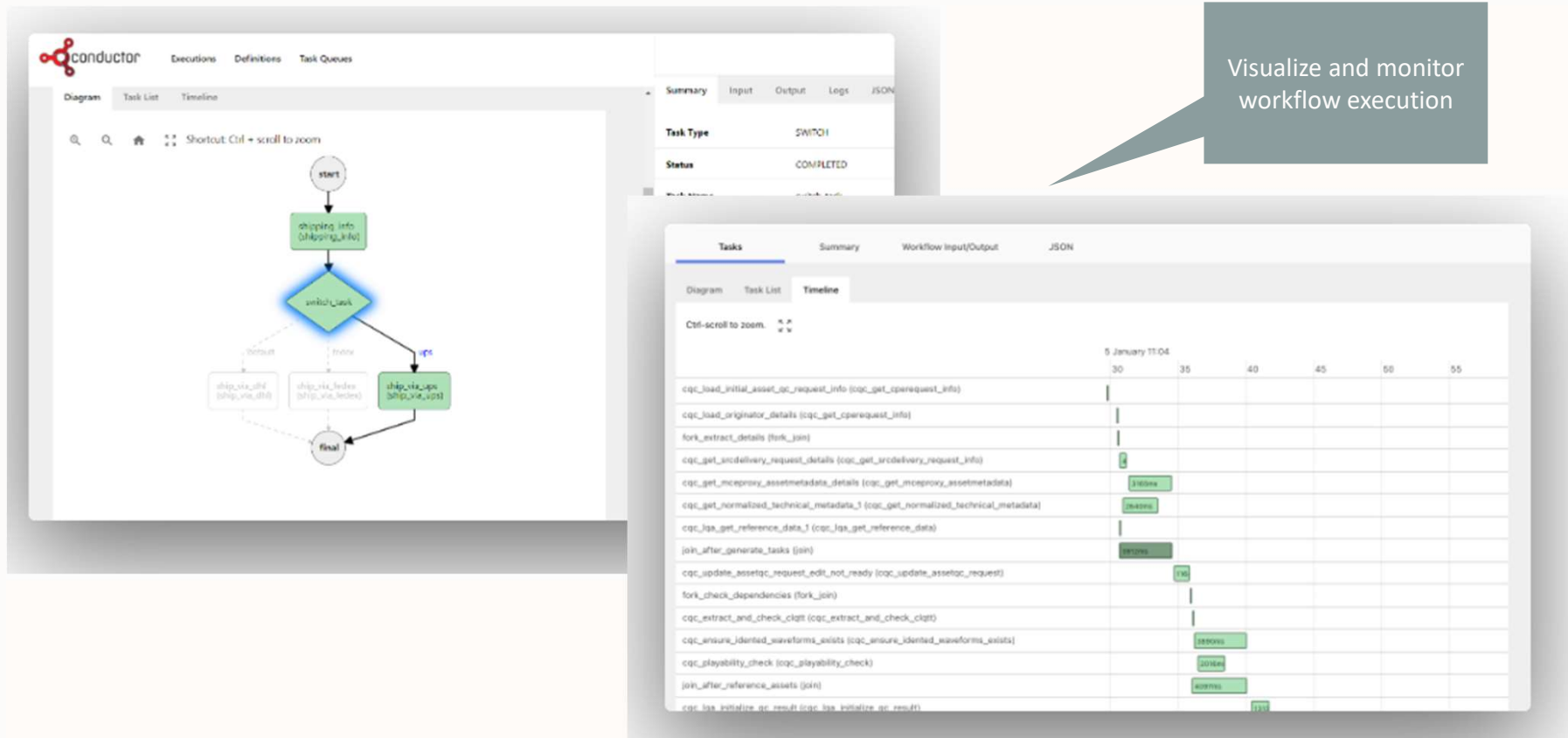
## Oracle Backend for Spring Boot feature/function – microservices

Microservices/Spring Boot developers want..	We meet this need with..
Runtime environment for services	<b>Containers and Kubernetes</b>
Easy access to database, messaging, etc.	Spring Boot Starters for Oracle technologies
Service discovery, client-side load balancing, fault tolerance	Spring Eureka Service Registry, Resilience4j
API gateway, management	APISIX API Gateway and Dashboard
Manage data consistency across microservices	Oracle Transaction Manager for Microservices
Workflow, orchestration, choreography	Netflix OSS Conductor
Externalize configuration from applications	Spring Config Server
Observability	Prometheus, Grafana, Jaeger, OpenTelemetry
Secure secrets management	Hashicorp (and OCI) Vault
Scale on demand	OKE and ADB autoscaling
Developer tooling	CLI and VS Code extension



# Workflows are important in modern apps

Netflix Conductor OSS included in Oracle Backend for Spring Boot



## Oracle Backend for Spring Boot feature/function – microservices

Microservices/Spring Boot developers want..	We meet this need with..
Runtime environment for services	<b>Containers and Kubernetes</b>
Easy access to database, messaging, etc.	Spring Boot Starters for Oracle technologies
Service discovery, client-side load balancing, fault tolerance	Spring Eureka Service Registry, Resilience4j
API gateway, management	APISIX API Gateway and Dashboard
Manage data consistency across microservices	Oracle Transaction Manager for Microservices
Workflow, orchestration, choreography	Netflix OSS Conductor
Externalize configuration from applications	Spring Config Server
Observability	Prometheus, Grafana, Jaeger, OpenTelemetry
Secure secrets management	Hashicorp (and OCI) Vault
Scale on demand	OKE and ADB autoscaling
Developer tooling	CLI and VS Code extension





## Spring Config Server

Configuration is a key challenge in microservices:

- We want to avoid “burning” configuration into a container image – both because it makes it hard to change, and because it could expose sensitive information
- The common pattern is to **externalize configuration** into the Spring Config Server, and then inject it into instances of the application at startup time, based on their environment/profile, version, etc.
- Spring Config Server stores the actual configuration data in an Oracle Database and/or vault (depending on how sensitive the data is)
- Spring has built-in “**profiles**” that make it easy to have multiple sets of configuration for different environments like development, testing, pre-production, production, etc.





## Oracle Backend for Spring Boot feature/function – microservices

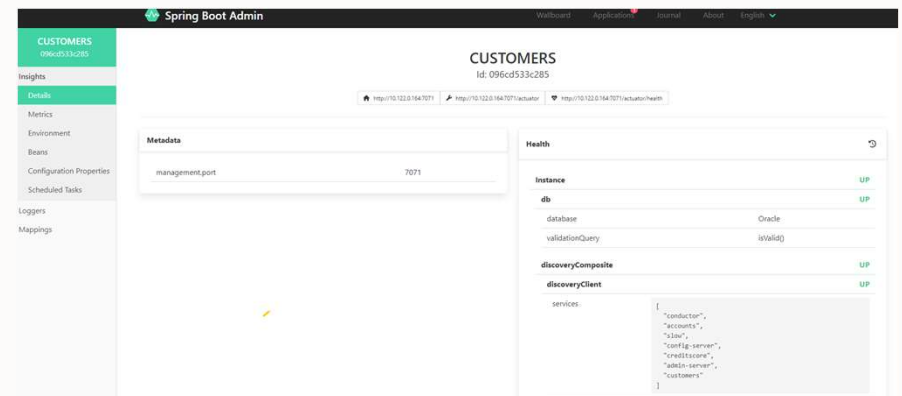
Microservices/Spring Boot developers want..	We meet this need with..
Runtime environment for services	<b>Containers and Kubernetes</b>
Easy access to database, messaging, etc.	Spring Boot Starters for Oracle technologies
Service discovery, client-side load balancing, fault tolerance	Spring Eureka Service Registry, Resilience4j
API gateway, management	APISIX API Gateway and Dashboard
Manage data consistency across microservices	Oracle Transaction Manager for Microservices
Workflow, orchestration, choreography	Netflix OSS Conductor
Externalize configuration from applications	Spring Config Server
Observability	Prometheus, Grafana, Jaeger, OpenTelemetry
Secure secrets management	Hashicorp (and OCI) Vault
Scale on demand	OKE and ADB autoscaling
Developer tooling	CLI and VS Code extension



# Spring Admin

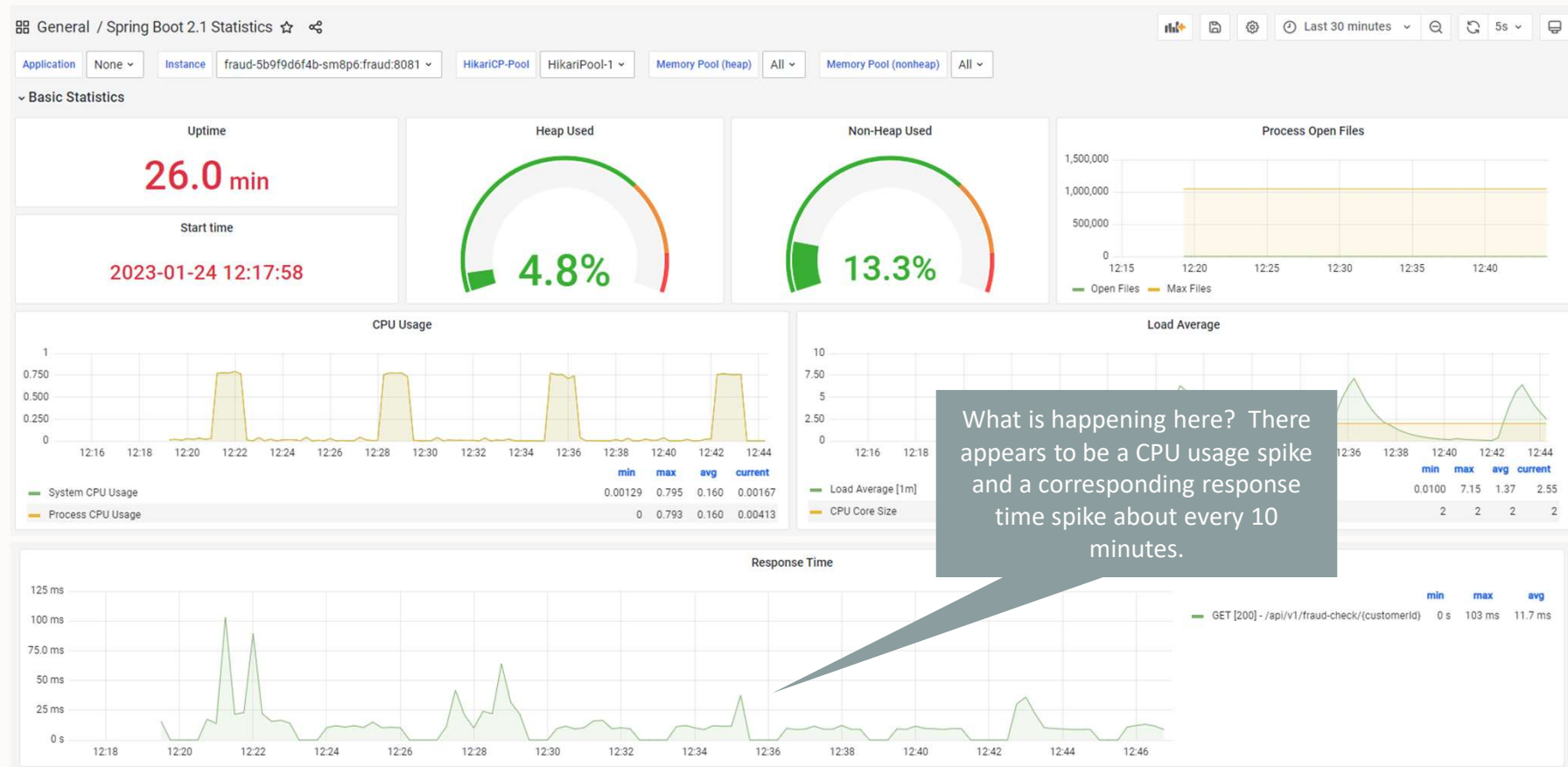
Built-in admin tools make it easy to see information about your Spring Boot applications:

- Which services are running
- Their health
- What beans and configuration they have had injected
- What endpoints they expose
- What metrics they provide
- JVM statistics
- And so on...



# Monitor Spring applications

Using the included Spring Boot Grafana Dashboard

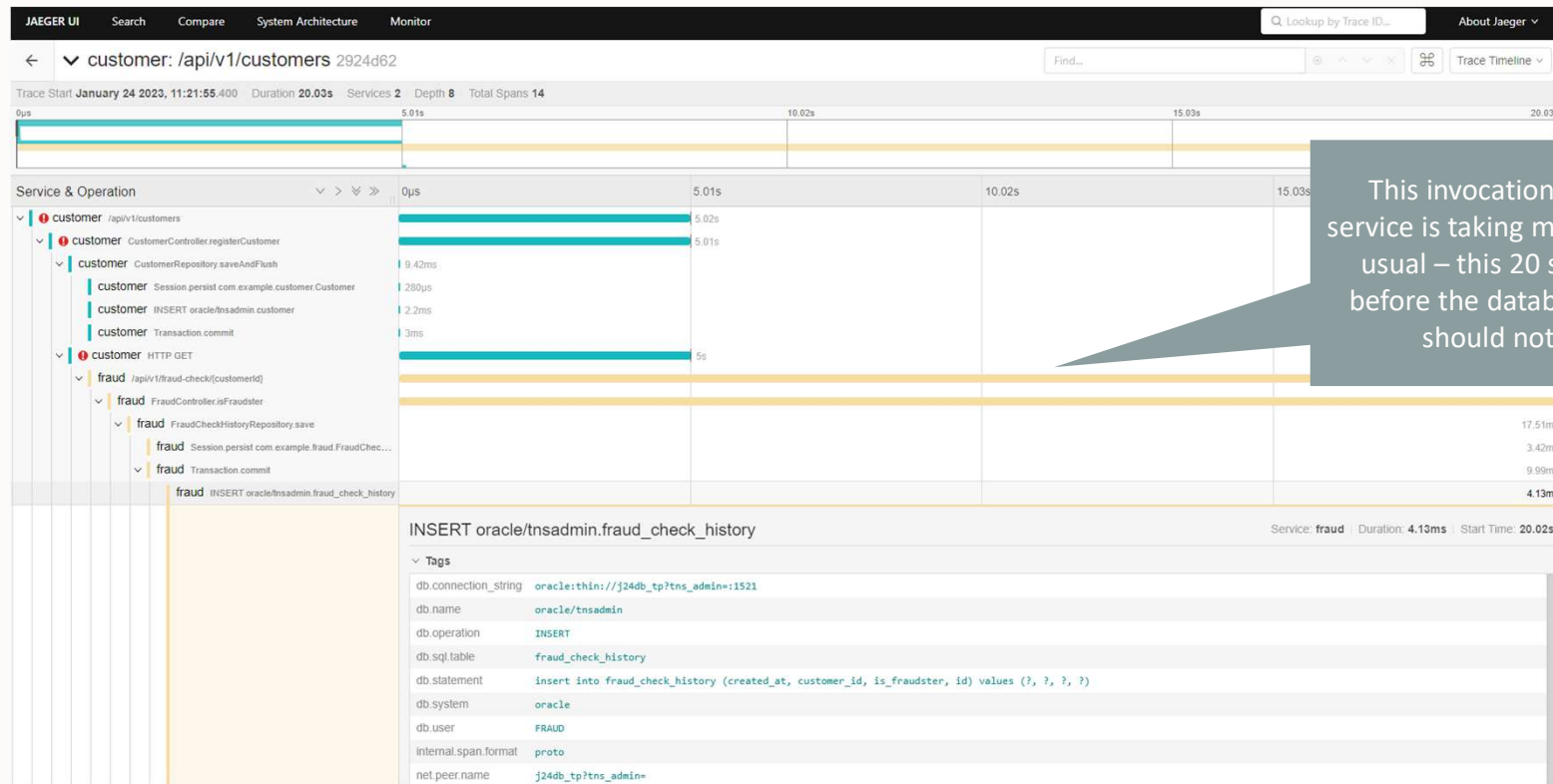


Copyright © 2023, Oracle and/or its affiliates



# Trace Spring applications (including into the database and TxEventQ)

## Using the included OpenTelemetry and Jaeger Dashboard



## Oracle Backend for Spring Boot feature/function – microservices

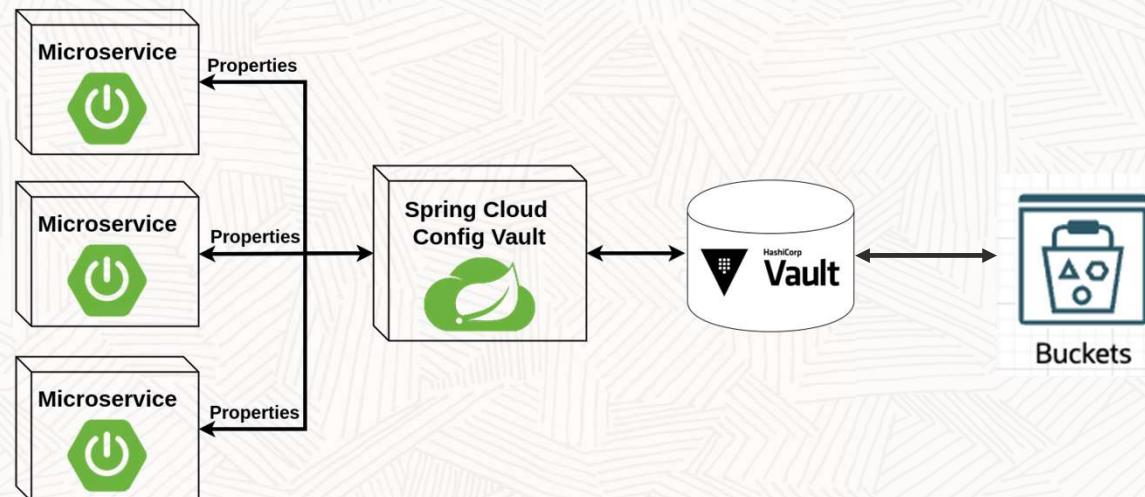
Microservices/Spring Boot developers want..	We meet this need with..
Runtime environment for services	<b>Containers and Kubernetes</b>
Easy access to database, messaging, etc.	Spring Boot Starters for Oracle technologies
Service discovery, client-side load balancing, fault tolerance	Spring Eureka Service Registry, Resilience4j
API gateway, management	APISIX API Gateway and Dashboard
Manage data consistency across microservices	Oracle Transaction Manager for Microservices
Workflow, orchestration, choreography	Netflix OSS Conductor
Externalize configuration from applications	Spring Config Server
Observability	Prometheus, Grafana, Jaeger, OpenTelemetry
Secure secrets management	Hashicorp (and OCI) Vault
Scale on demand	OKE and ADB autoscaling
Developer tooling	CLI and VS Code extension





## HashiCorp Vault and OCI Vault

- Store secrets
- Dev or Production mode
- HashiCorp Vault is using Object Storage
- OCI Vault for auto unsealing
- OCI Vault is available for applications running on Oracle Backend for Spring Boot
- Spring injection



## Oracle Backend for Spring Boot feature/function – microservices

Microservices/Spring Boot developers want..	We meet this need with..
Runtime environment for services	<b>Containers and Kubernetes</b>
Easy access to database, messaging, etc.	Spring Boot Starters for Oracle technologies
Service discovery, client-side load balancing, fault tolerance	Spring Eureka Service Registry, Resilience4j
API gateway, management	APISIX API Gateway and Dashboard
Manage data consistency across microservices	Oracle Transaction Manager for Microservices
Workflow, orchestration, choreography	Netflix OSS Conductor
Externalize configuration from applications	Spring Config Server
Observability	Prometheus, Grafana, Jaeger, OpenTelemetry
Secure secrets management	Hashicorp (and OCI) Vault
Scale on demand	OKE and ADB autoscaling
Developer tooling	CLI and VS Code extension



## Scale on Demand

A key feature of a “backend as a service” is the ability to scale on demand

- Once you deploy your applications, you don’t want to have to worry about scaling the environment
- We pre-configure the backend at your chosen starting (smallest) size, and then **scale** both the Kubernetes cluster and the Autonomous Database instance **automatically based on demand** (you can opt out of auto-scaling if you prefer)
- Within the Kubernetes cluster itself, we can also auto-scale instances of your services to make sure there are enough running instances to meet the demand

Oracle Kubernetes Engine (OKE) and Autonomous Database (ADB) autoscaling

The screenshot displays the configuration interface for Oracle Kubernetes Engine (OKE) and Autonomous Database (ADB). It is divided into two main sections: 'Node Pool' and 'Database Options'.

**Node Pool Section:**

- ☐ **Enable Horizontal Pod Scaling?** (A red arrow points to this checkbox. Below it is the text: 'Allow horizontal pod scaling within a Node Pool'.)
- Node Pool Workers:** A dropdown menu showing the value '3'. Below it is the text: 'The total number of Workers in each Node Pool.'
- Node Pool Worker Shape:** A dropdown menu showing the value 'VM.Standard.E4.Flex'. Below it is the text: 'Choose the shape of the Node Pool Workers.'
- Node Workers OCPU:** A dropdown menu showing the value '1'. Below it is the text: 'The initial number of OCPU for'.

**Database Options Section:**

- ☒ **Show database options?** (A red arrow points to this checkbox. Below it is the text: 'Shows database options.')
- Autonomous Database Network Access:** A dropdown menu showing the value 'SECURE\_ACCESS\_FROM\_EVERYWHERE'. Below it is the text: 'Choose the Autonomous Database Network Access.'
- Autonomous Database CPU Core Count:** A dropdown menu showing the value '1'. Below it is the text: 'Choose how many CPU cores will be used for ADB Instance.'
- ☐ **Allow Autonomous Database OCPU Auto Scaling?** (Below it is the text: 'Enable auto scaling for the ADB OCPU core count (x3 ADB OCPU).')



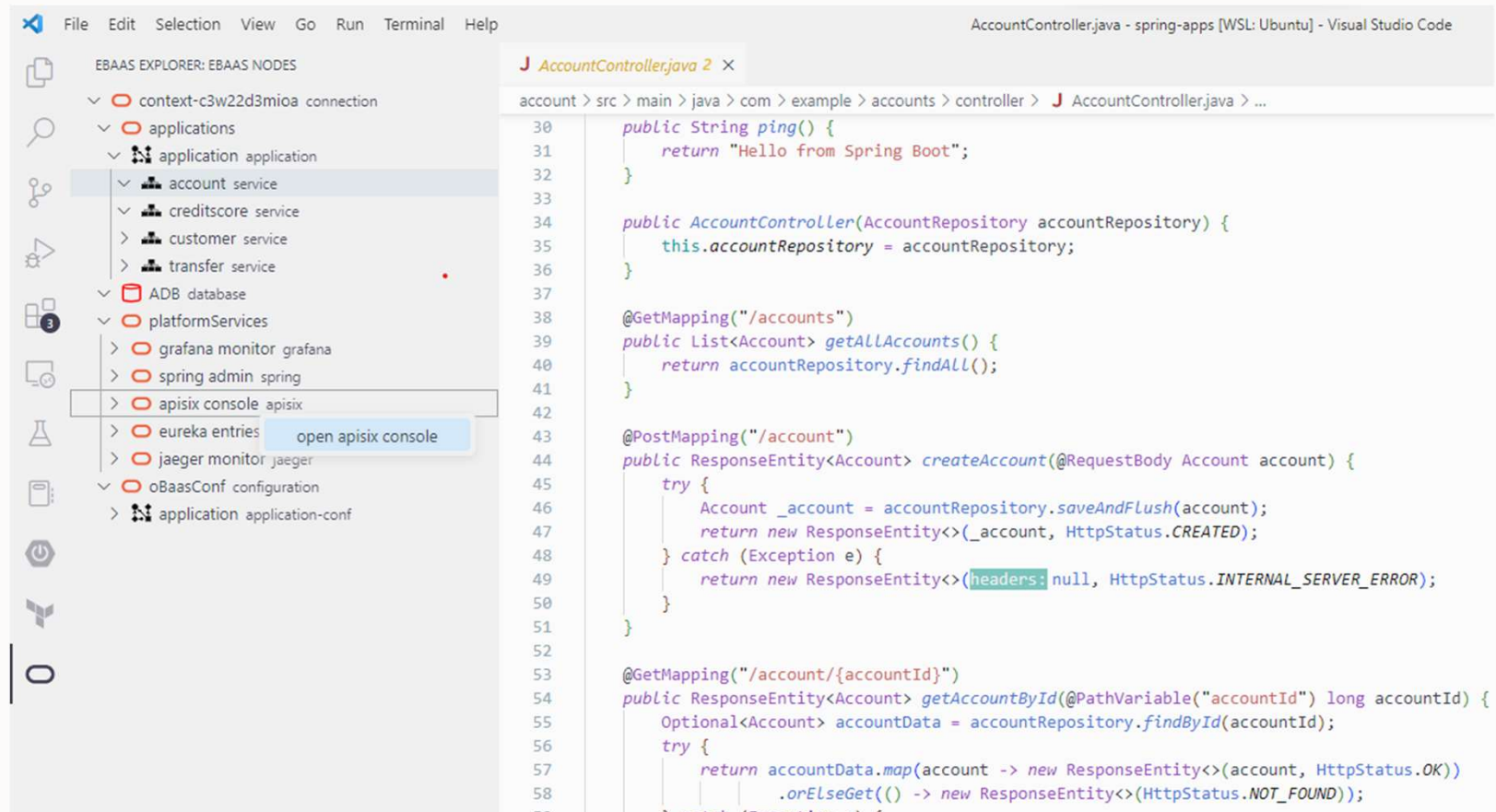


## Oracle Backend for Spring Boot feature/function – microservices

Microservices/Spring Boot developers want..	We meet this need with..
Runtime environment for services	<b>Containers and Kubernetes</b>
Easy access to database, messaging, etc.	Spring Boot Starters for Oracle technologies
Service discovery, client-side load balancing, fault tolerance	Spring Eureka Service Registry, Resilience4j
API gateway, management	APISIX API Gateway and Dashboard
Manage data consistency across microservices	Oracle Transaction Manager for Microservices
Workflow, orchestration, choreography	Netflix OSS Conductor
Externalize configuration from applications	Spring Config Server
Observability	Prometheus, Grafana, Jaeger, OpenTelemetry
Secure secrets management	Hashicorp (and OCI) Vault
Scale on demand	OKE and ADB autoscaling
Developer tooling	CLI and VS Code extension

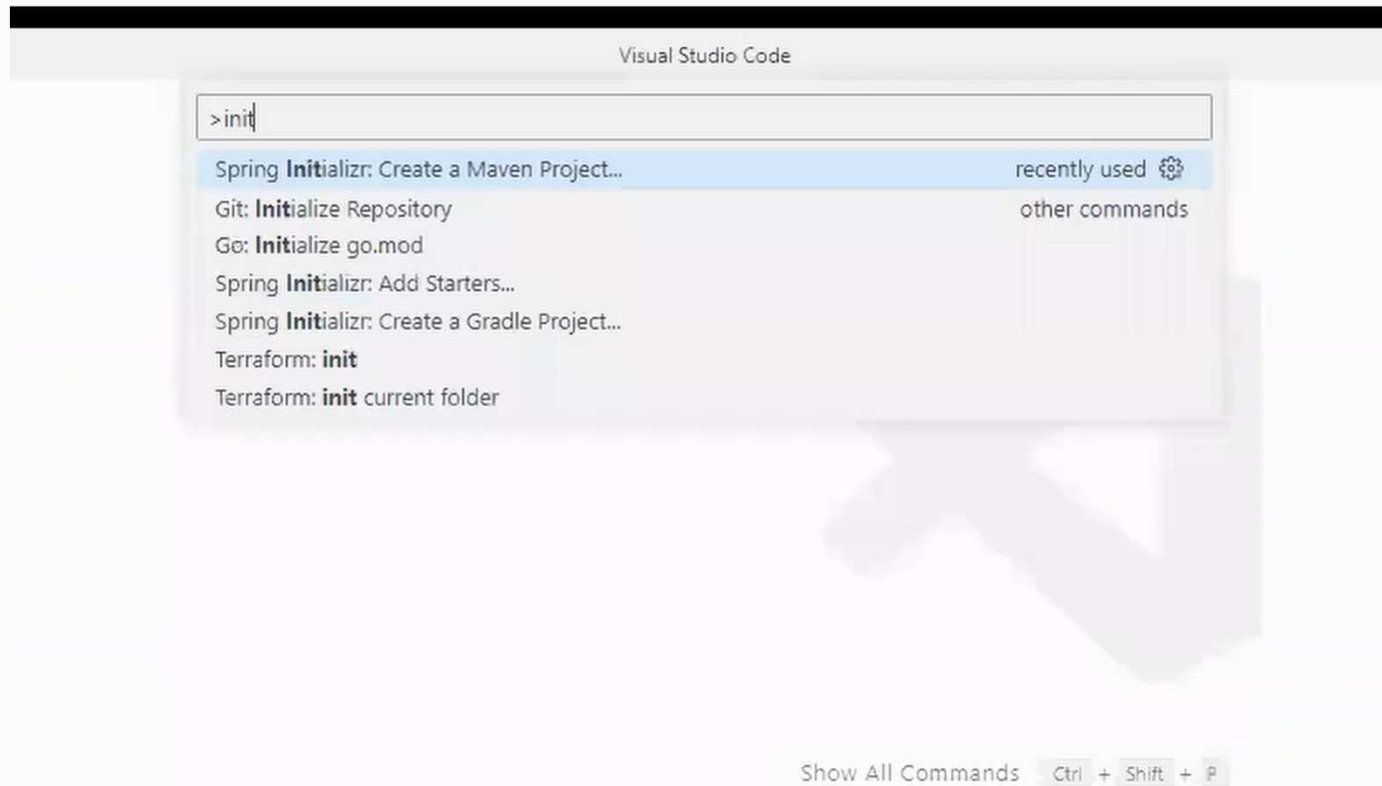


## Visual Studio Code Extension



# Create a Spring application

Using Spring Initializr in Visual Studio Code – and create the Customer Entity



## Development time features

- CLI and Visual Studio Code Extension to manage deployment and configuration
- Spring Boot Starters for Oracle technologies (e.g. UCP, AQ/TxEventQ JMS, SODA/JSON)
- Spring Data JDBC, JPA
- Spring Config
- Eureka Client, Feign, Ribbon, Resilience4j
- Spring Cloud Streams (Kafka, more coming soon)
- Integration with SQLcl and Liquibase for schema CI/CD



# Developer Tooling

## Command Line Interface (CLI)

```
atael@atael-mac ~  
→ oractl  
  
⏏ ⏏ ⏏ ⏏ ⏏ ⏏ ⏏ ⏏ ⏏ ⏏  
⏏ ⏏ ⏏ ⏏ ⏏ ⏏ ⏏ ⏏ ⏏ ⏏  
  
10:39:00.778 [main] INFO o.s.s.cli.shell.ShellApplication - Starting AOT-processed ShellApplication using Java 17.0.5 with PID 14931 (/Users/atael/bin/oractl started by atael in /Users/atael)  
10:39:00.781 [main] DEBUG o.s.s.cli.shell.ShellApplication - Running with Spring Boot v3.0.0, Spring v6.0.2  
10:39:00.781 [main] INFO o.s.s.cli.shell.ShellApplication - The following 1 profile is active: "obaas"  
10:39:00.911 [main] INFO o.s.s.cli.shell.ShellApplication - Started ShellApplication in 0.305 seconds (process running for 0.439)  
oractl:>help  
AVAILABLE COMMANDS  
  
Admin Server Commands  
  change-password: Change password for OBaaS Spring Cloud admin user.  
  connect: Connect to the OBaaS Spring Cloud admin console.  
  
Application/Namespace Commands  
  create: Create an application/namespace.  
  delete: Delete a service or entire application/namespace.  
  
Built-In Commands  
  help: Display help about available commands  
  stacktrace: Display the full stacktrace of the last error.  
  clear: Clear the shell screen.  
  quit, exit: Exit the shell.  
  history: Display or save the history of previously run commands  
  version: Show version info  
  script: Read and execute commands from a file.  
  
Informational Commands  
  list: list/show details of application services.  
  
Service Commands  
  bind: Create a schema/user and bind it to service deployment.  
  config: View and modify Service configuration.  
  deploy: Deploy a service.  
  
oractl:>
```



# Mobile app dev

Overview and demo



# What is the Parse Platform?

## Major features of an MBaaS

- Runs as (many stateless) NodeJS processes
- Schema-less (uses Oracle NodeDB library to communicate with Oracle Autonomous Database)
- Create/Delete indexes on collections, create compound indexes
- User authentication and Access Control Lists supported
- Relations 1-1 and 1-Many
- Queries (many features included)
- GraphQL support
- LiveQuery Support – subscribe to a query
- Push Notifications
- Mobile apps use the Parse APIs which are *super easy to use* and available for all the common platforms and frameworks including:
  - Android (phone/tablet/tv/auto)
  - IOS (phone/table/tv/auto/computer)
  - JavaScript
  - React/React Native
  - Flutter
  - Unity (games)
  - .Net
  - Embedded/IOT/C
  - GraphQL
  - REST





# Developer-focused Dashboard

PARSE DASHBOARD 5.0.0  
ADMIN

j2ob

Core

Browser

Push

Role

User

Jobs

Logs

Config

API Console

Logout

CLASS

Movies

objectId

String

crew

Array

genre

Array

gross

String

summary

String

list\_price

Number

ACL

ACL

movie\_id

Number

image\_url

String

updatedAt

Date

views

Number

<input type="checkbox"/>	NLMkmQqKEB	[{"job": "producer"...	["Drama"]	+74200000	August: Osage Coun...	4.99	Public Read + Write	362	https://upload.wik...	24 Jan 2023 at 13:...	1637
<input type="checkbox"/>	nbRDyoDavI	[{"job": "producer"...	["Musical"]	3300000	At War with the Ar...	4.99	Public Read + Write	355	https://upload.wik...	24 Jan 2023 at 13:...	34
<input type="checkbox"/>	OG09y7amgY	[{"job": "producer"...	["Crime", "Comedy"...	2900000	Assassination Nat...	4.99	Public Read + Write	348	https://upload.wik...	24 Jan 2023 at 13:...	660
<input type="checkbox"/>	dWSJvEaL8U	[{"job": "producer"...	["Comedy", "Drama"]	6400000	Article 99 is a 19...	0	Public Read + Write	340	https://upload.wik...	24 Jan 2023 at 13:...	77
<input type="checkbox"/>	k8AytFQ9oD	[{"job": "director"...	["Unknown"]	(null)	Armageddon Gospels...	0	Public Read + Write	333	https://upload.wik...	24 Jan 2023 at 13:...	13
<input type="checkbox"/>	mkR3P0A72U	[]	["Mystery"]	(null)	Archaeology of a W...	2.99	Public Read + Write	325	https://upload.wik...	24 Jan 2023 at 13:...	1
<input type="checkbox"/>	GqkT3Imkk	[{"job": "producer"...	["Drama", "Adventur...	+355237933	Apollo 13 is a 199...	4.99	Public Read + Write	318	https://upload.wik...	24 Jan 2023 at 13:...	2006
<input type="checkbox"/>	QG2QJLxEqm	[{"job": "producer"...	["Horror", "Sci-Fi"]	+61.808	Antiviral is a 201...	4.99	Public Read + Write	310	https://upload.wik...	24 Jan 2023 at 13:...	252
<input type="checkbox"/>	CRVVBg1wxl	[{"job": "producer"...	["Comedy", "Drama"]	5700000	Anomalisa is a 201...	0	Public Read + Write	303	https://upload.wik...	24 Jan 2023 at 13:...	563
<input type="checkbox"/>	qpShtL9VKc	[{"job": "producer"...	["Animation", "Come...	(null)	Animalympics is a ...	0	Public Read + Write	296	https://upload.wik...	24 Jan 2023 at 13:...	111
<input type="checkbox"/>	T9fCSaHkUl	[{"job": "director"...	["Drama"]	(null)	Angel and Big Joe ...	0	Public Read + Write	289	https://upload.wik...	24 Jan 2023 at 13:...	2
<input type="checkbox"/>	B2UNFLIfHU	[{"job": "director"...	["Documentary"]	(null)	And the Pursuit of...	0	Public Read + Write	281	https://upload.wik...	24 Jan 2023 at 13:...	7
<input type="checkbox"/>	mhqBGHkWh0	[{"job": "director"...	["Comedy", "Romance...	(null)	An Oversimplificat...	4.99	Public Read + Write	274	https://upload.wik...	24 Jan 2023 at 13:...	47
<input type="checkbox"/>	BydAZN5QqC	[{"job": "director"...	["Documentary"]	(null)	An Essay on Matiss...	0	Public Read + Write	267	(null)	24 Jan 2023 at 13:...	1
<input type="checkbox"/>	hBf7nTLf6E	[{"job": "producer"...	["Drama"]	(null)	Amphetamine (Chine...	1.99	Public Read + Write	260	https://upload.wik...	24 Jan 2023 at 13:...	52
<input type="checkbox"/>	FF00o1DKLA	[{"job": "director"...	["Drama"]	(null)	American Son is a ...	3.99	Public Read + Write	252	https://upload.wik...	24 Jan 2023 at 13:...	27
<input type="checkbox"/>	hgQT2v3b2P	[{"job": "producer"...	["Action", "Romance...	+2237561	American Heist is ...	0	Public Read + Write	245	https://upload.wik...	24 Jan 2023 at 13:...	196
<input type="checkbox"/>	RqC0hFERHe	[{"job": "director"...	["Thriller", "Drama...	(null)	Ambushed is a 2013...	4.99	Public Read + Write	238	https://upload.wik...	24 Jan 2023 at 13:...	22
<input type="checkbox"/>	649ENLuL8y	[{"job": "producer"...	["Crime", "Drama", "...	32100000	Alpha Dog is a 200...	0	Public Read + Write	229	https://upload.wik...	24 Jan 2023 at 13:...	1791
<input type="checkbox"/>	azxW90xyrf	[{"job": "producer"...	["Action", "Fantasy...	(null)	Allan Quatermain a...	0.99	Public Read + Write	222	https://upload.wik...	24 Jan 2023 at 13:...	35
<input type="checkbox"/>	mewEFseV5	[{"job": "director"...	["Drama"]	(null)	All at Once is an ...	0	Public Read + Write	215	https://upload.wik...	24 Jan 2023 at 13:...	8
<input type="checkbox"/>	Mhvpvx2h8j	[]	["Comedy"]	(null)	All American Orgy ...	3.99	Public Read + Write	208	https://upload.wik...	24 Jan 2023 at 13:...	9
<input type="checkbox"/>	BviSQAd3DQ	[{"job": "producer"...	["Horror", "Sci-Fi"]	(null)	Alien Abduction is...	3.99	Public Read + Write	201	(null)	24 Jan 2023 at 13:...	31
<input type="checkbox"/>	A61223j5SE	[{"job": "director"...	["Romance"]	(null)	Alex & The List is...	0	Public Read + Write	194	https://upload.wik...	24 Jan 2023 at 13:...	53
<input type="checkbox"/>	lBk4PNH74	[{"job": "producer"...	["Horror", "Thrille...	103000000	Airport 1975 (also...	3.99	Public Read + Write	187	https://upload.wik...	24 Jan 2023 at 13:...	279
<input type="checkbox"/>	hT0FeCUJFq	[{"job": "producer"...	["Drama", "Romance"]	+1000000	Ain't Them Bodies ...	2.99	Public Read + Write	180	https://upload.wik...	24 Jan 2023 at 13:...	398
<input type="checkbox"/>	46Q2GW7K01	[{"job": "producer"...	["Horror", "Film-No...	(null)	Aftermath is a 199...	0.99	Public Read + Write	173	https://upload.wik...	24 Jan 2023 at 13:...	49



# Initialize the Parse SDK in App.js

## React Native example

```
import './App.css';

function App() {

  const Parse = require('parse/react-native.js');
  Parse.setAsyncStorage(AsyncStorage);
  Parse.initialize("aL1Vq1HJYyEj8hV4xxx6Ynf5rxL4xun3Vtat2MD");
  Parse.serverURL = 'http://1.2.3.4/parse';

  // state and handlers here

  return (
    <ReactComponentsHere />
  );
}

export default App;
```

Initialize the Parse SDK and  
connect to our backend

# Use Parse in a component

## React Native example using Parse Documents API

```
const performDeposit = async (parseAddress, accountNum, userid, amount, accountType) => {  
  
  const Deposit = Parse.Object.extend('BankAccount');  
  const deposit = new Deposit();  
  deposit.set('accountNum', +accountNum);  
  deposit.set('action', 'Deposit');  
  deposit.set('amount', +amount);  
  deposit.set('userId', userid);  
  deposit.set('accountType', accountType);  
  deposit.save().then(  
    id => console.log('saved with id ' + JSON.stringify(id)),  
    error => console.log('failed to save, error = ' + error),  
  );  
};
```

Here's an example of creating a new document in the "BankAccount" collection

And saving it on the backend

# Use Parse in a component

## React Native Example of calling Cloud Code

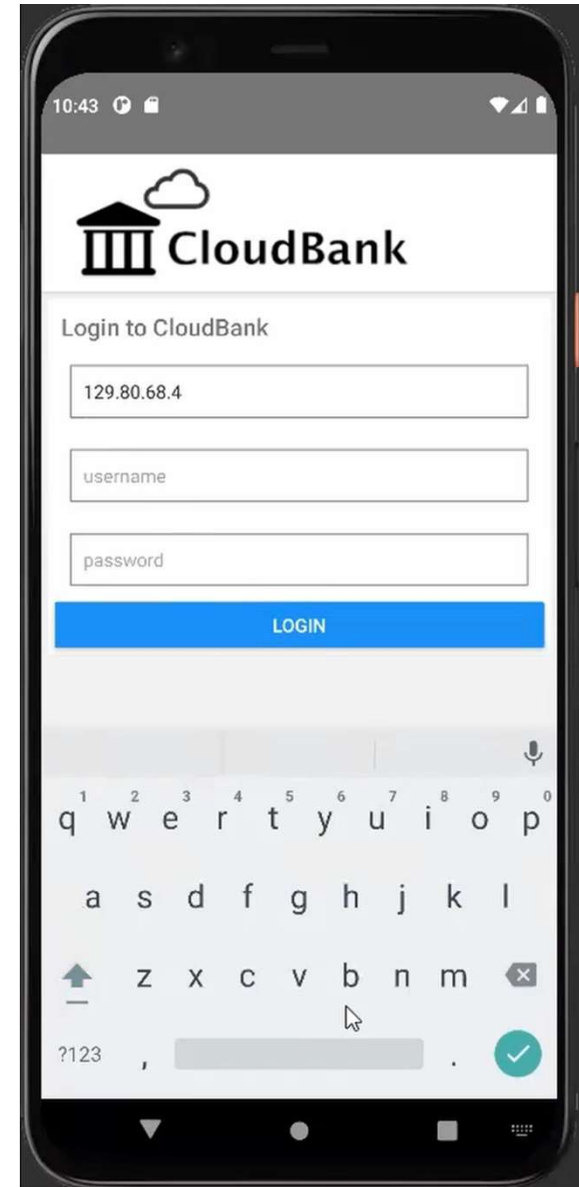
```
// get the balance of the given account  
export const getAccountBalance = async (parseAddress, accountNum) => {  
  const params = {accountNum: accountNum};  
  const balance = await Parse.Cloud.run('balance', params);  
  return balance;  
};
```

Here's an example of calling a  
"cloud function" on the  
backend

# Mobile App Demo

This is a sample/demo Cloud Banking application built with Flutter and the Parse APIs

- Developed in **Flutter**, running on Android (in this case) against the Oracle Backend for Spring Boot (and Parse Platform)
- Includes user authentication, integration with Spring Boot services via the API Gateway (for “transactional” features), and integration with Parse APIs (for “origination” features)
- Full source code available and used in our “Live Lab” where you would implement a new “Cloud Cash” feature and integrate it with Spring Boot services and a Long Running Action (Saga) in the backend!



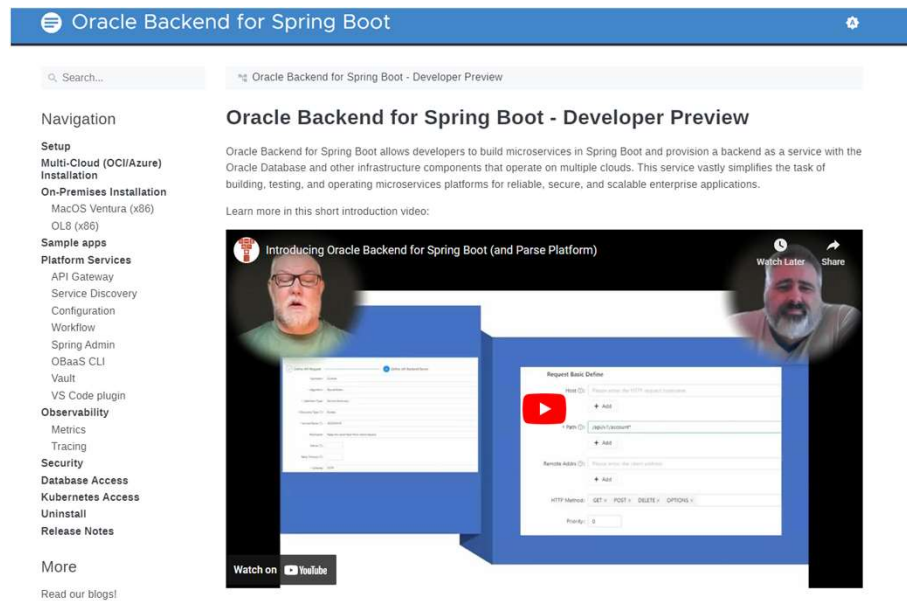


Want to learn more?



# Introductory YouTube video and channel

Go to [bit.ly/oraclespringboot](http://bit.ly/oraclespringboot) where you will find our quick intro video on the first page



We are building a new channel to publish our videos – we're just getting started – but much more to come!

<http://bit.ly/convergeddatabase>



**Introducing Oracle Backend for Spring Boot (and Parse Platform)**  
This video introduces the Oracle Backend for Spring Boot (and Parse Platform) which makes it super easy to develop, run and...



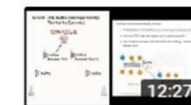
**Long Running Actions demonstration**  
Paul Parkinson demonstrates a Spring Boot microservices application that implements the Saga pattern with Long Running...



**Microservices with Oracle's Converged Database.**  
Learn about building modern microservices with Oracle's Converged Database.



**Data Refactoring Advisor demo**  
Data Refactoring Advisor is a tool that is currently under development to help refactor monolithic databases for...



**Tour of Oracle Transactional Event Queues**  
00:00 Introduction 01:34 Managing TEQ with the SQLcl command line tool 02:26 REST APIs for TEQ, provided by Oracle REST Dat...



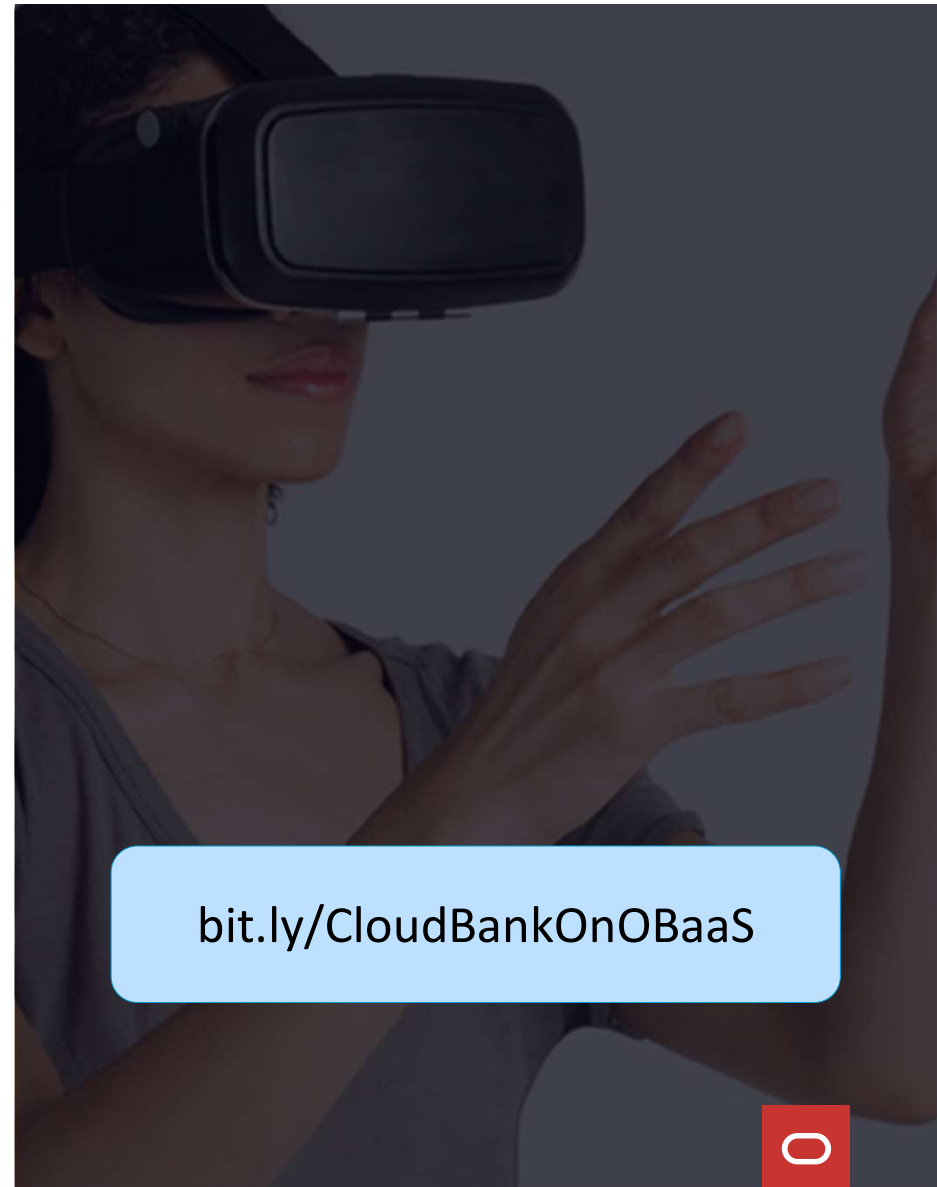
## Live Lab: Build a working Cloud Banking application

*Spring Boot microservices for customer, account, transactions, payments, credit score*

*Extend the Flutter mobile application to add “Cloud Cash” payment feature*

*Implement “Cloud Cash” payments using Long Running Action microservice transaction pattern*

“In this lab, you will build a sample application, CloudBank, by utilizing Spring Boot microservices for a fully working application that you can take with you when you leave this session. You will learn about service discovery, external configuration, workflow, API management and observability. Then you'll deploy the full application and a React front-end web application that allows customers to see their account history, make transfers, apply for credit and more.



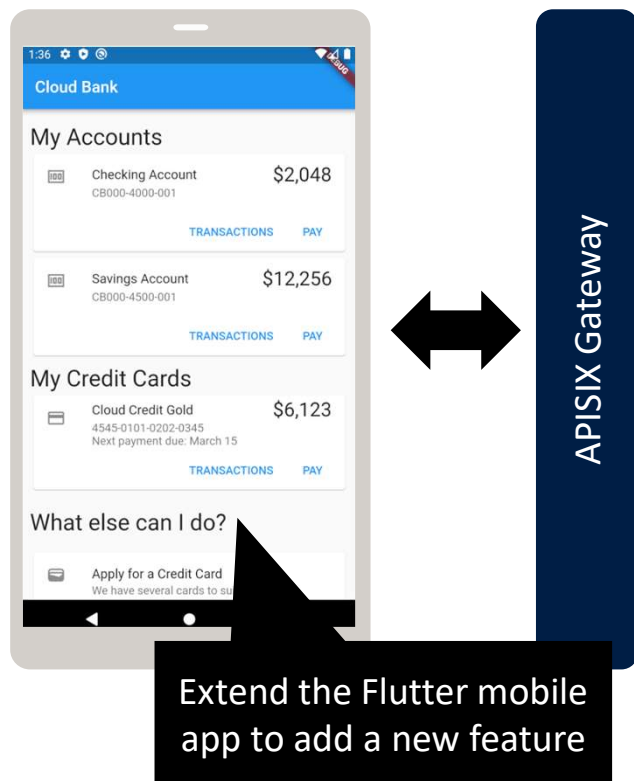
[bit.ly/CloudBankOnOBaaS](https://bit.ly/CloudBankOnOBaaS)



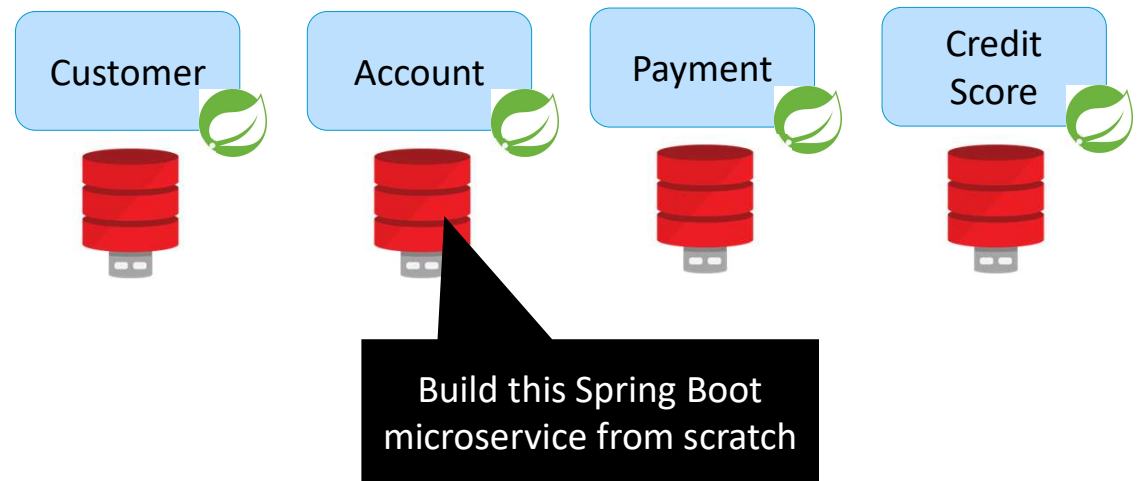
# CloudBank

The foundation of the Handson Lab

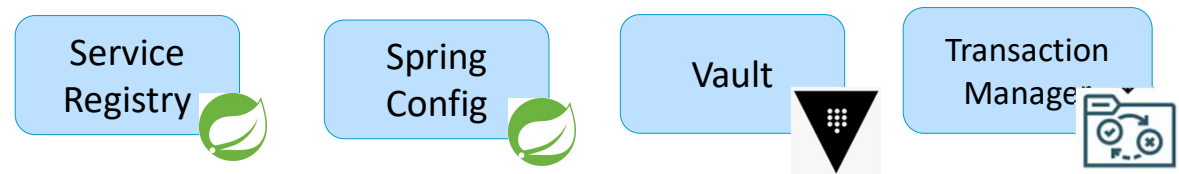
<http://bit.ly/CloudBankOnOBaaS>



## Application Microservices

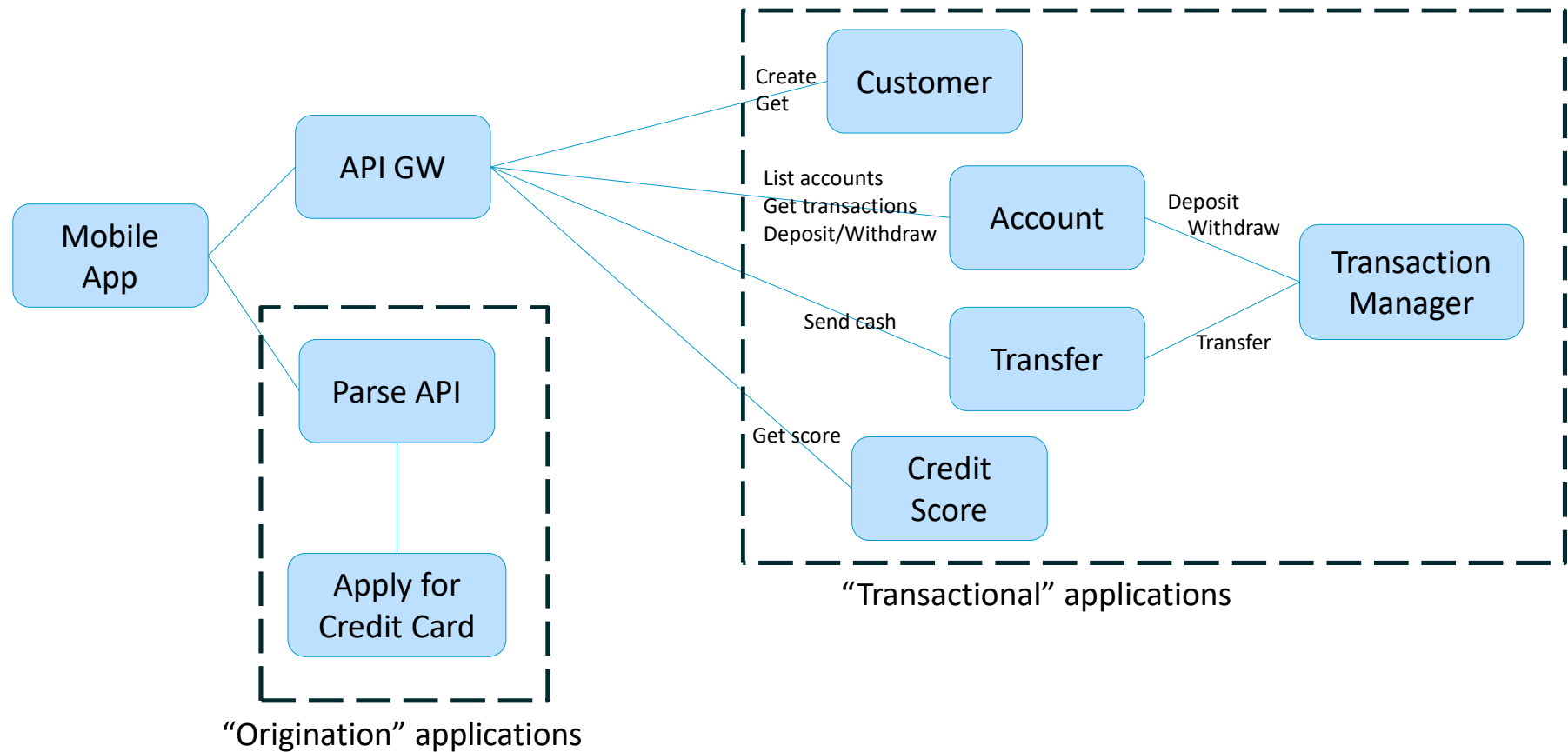


## Platform services



# CloudBank

The foundation of the Hands-on Lab





# Mini-workshop offer! Both DBAs and Developers Unite

DevOps + DataOps = App and Database Lifecycle management

We are happy to offer a personalized mini-workshop to learn about topics like:

Kubernetes, containers, microservices, mobile, polyglot development, security, converged database, etc.

We will customize the agenda to your needs -- **Onsite or online**



## Workshop Agenda 9am – 4pm

1. 9am | Simplify DevOps and DataOps with Spring Boot Backend Platform (60m)
2. 10am | Kubernetes, Containers, and Microservices with the Oracle Database (30m)
3. 10:30am | Hands-On Lab: Build a Working Application for Cloud Banking (120m)

12:30pm Lunch

4. 2pm | Polyglot development with Oracle Database - SQL, APEX, Python, Java, JavaScript, .NET (30m)
5. 2:30pm | DevOps and DataOps Deep dive: Security, HA, Service Discovery, Monitoring, Scaling (30m)
6. 3pm | Converged Data: Documents, Relational, Graph, Spatial, Streams, Events, AI/ML (30m)
7. 3:30pm | Feedback, Discussion, Q&A, and Next Steps (30m)



# Open Q&A



## Open Q&A

While we're talking...

<https://bit.ly/OracleAppSurvey>

Get in touch with us and we'd be super happy to talk to you about your needs and answer questions.



Thank you for coming – are you interested in other topics? Like these? Please let us know!

- *Oracle Database Operator for Kubernetes and Metrics*
- *Kubernetes for DBAs*
- *Twelve Patterns for Microservices Success*
- *Messaging and Streaming - Kafka and Oracle TxEventQ*
- *Kubernetes Operator for Horizontal scaling and Data sovereignty with Oracle Database*
- *Transactions across Microservices*

mark.x.nelson@oracle.com  
andy.tael@oracle.com