Java Update

CREATE THE FUTURE

Part 1 - Java SE / Roadmap

Part 2 - Java EE / Java in the Cloud

Steve Elliott
Java Technology Lead
Oracle UK

Bristol Java Meetup
@bristoljava

May 2016

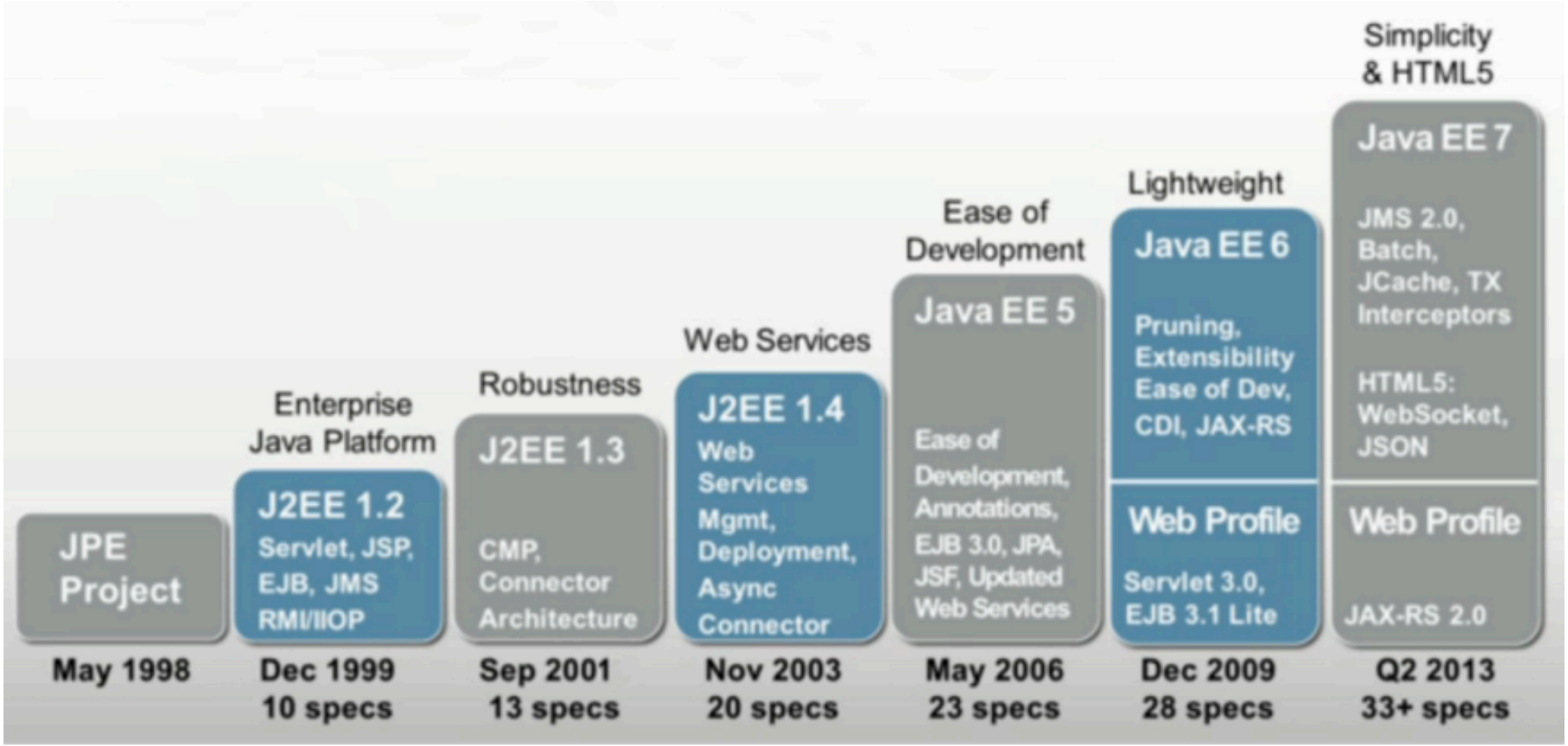
Java ORACLE

Safe Harbor Statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

From Java EE ➡ Cloud, Microservices and Beyond

Java EE Past & Present



A Brief(ish) History of AppServers (c2004)

- 1992 CERN Web Server — info.cern.ch
- 1993 NCSA Web Server
  CGI & Forms added to NCSA Server
- 1994 Sun - DOE / Neo (CORBA)
  IBM – ComponentBroker (CORBA) [later WebSphere]
- 1995 Kiva founded
  NetDynamics founded
  Weblogic founded - dbKona/T3Server/Tengah
  BEA founded - distributes and then buys Tuxedo
  Apache first release (evolution of NCSA)
- 1996 JavaSoft(Sun) creates Jeeves (Java Web Server)
  Defines Servlet API which moves to JSDK
- 1997 Netscape buys Kiva
  BEA acquires ObjectBroker from DEC (CORBA)
  Sun launches JPE (Java Platform for the Enterprise)
  - EJB, JDBC, RMI, JTS, JNDI, JMAPI, JMS...
- 1998 Sun buys NetDynamics
  BEA buys Weblogic
  WebSphere Appserver V1 (Servlets)
- 1999 Sun donates code to Apache Software Foundation
  - combines with existing Apache JServ work to form Tomcat
  Sun buys Forte
  AOL buys Netscape
  Sun/Netscape Alliance Formed
  J2EE 1.2 released (including reference implementation - RI)
  Marc Fleury creates EJB-OSS
  Later renamed to JBOSS and JBoss
- 2000 iPlanet Application Server 6 (J2EE 1.2) [1st J2EE 1.2 certified]
- 2002 Sun ONE Application Server 7 (J2EE 1.3)
- 2004 Sun Java Application Server 8 (J2EE 1.4)

2005 – GlassFish (Java EE 5)
2007 – GlassFish 2
2009 – GlassFish 3 (Java EE 6)
2013 – GlassFish 4 (Java EE 7)

2014 – WildFly 8 (Java EE 7)
2015 – WildFly 9
2016 – WildFly 10

2013 – WebSphere
Liberty Profile (Java EE7)

2011 – TomEE (Java EE 6w)

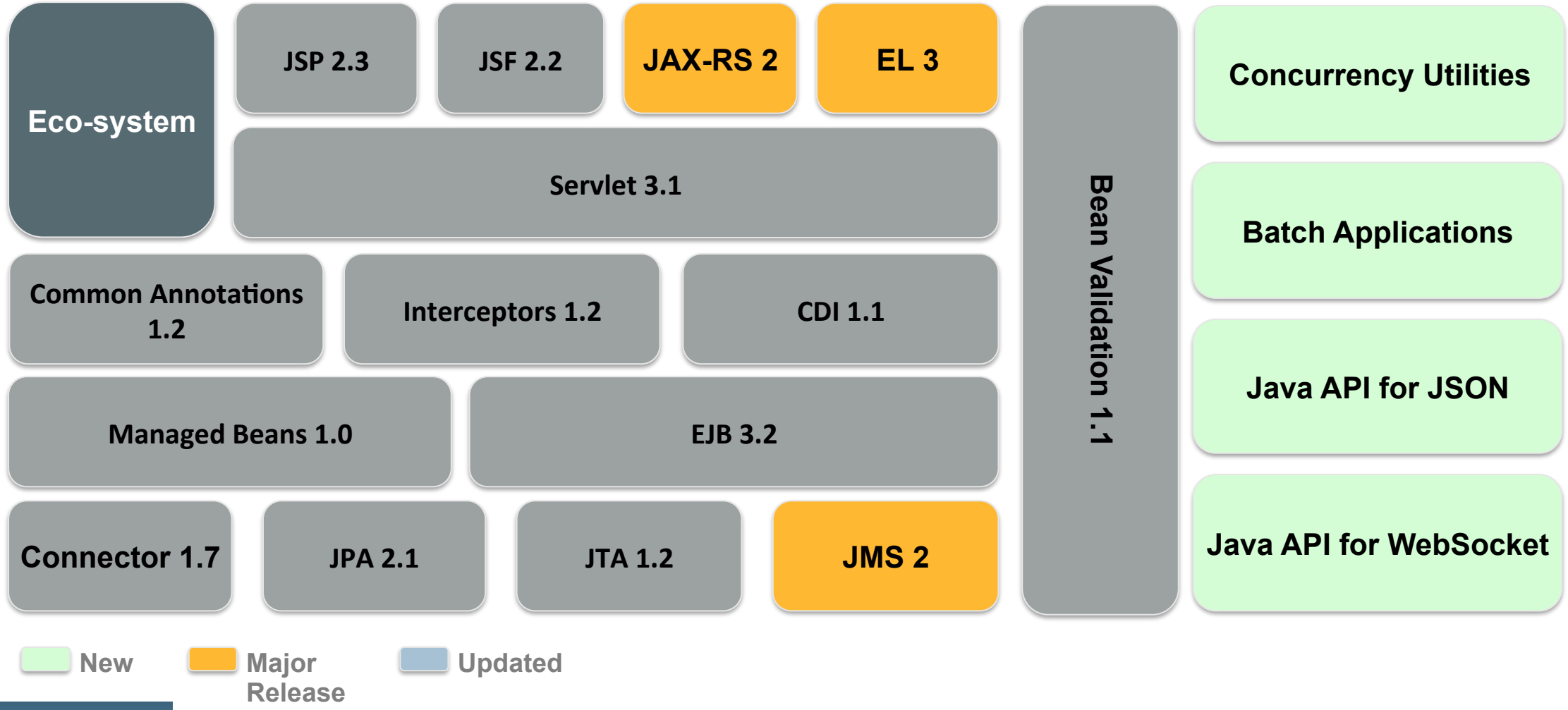
Java EE 7 Themes

- More annotated POJOs
- Less boilerplate code
- Cohesive integrated platform

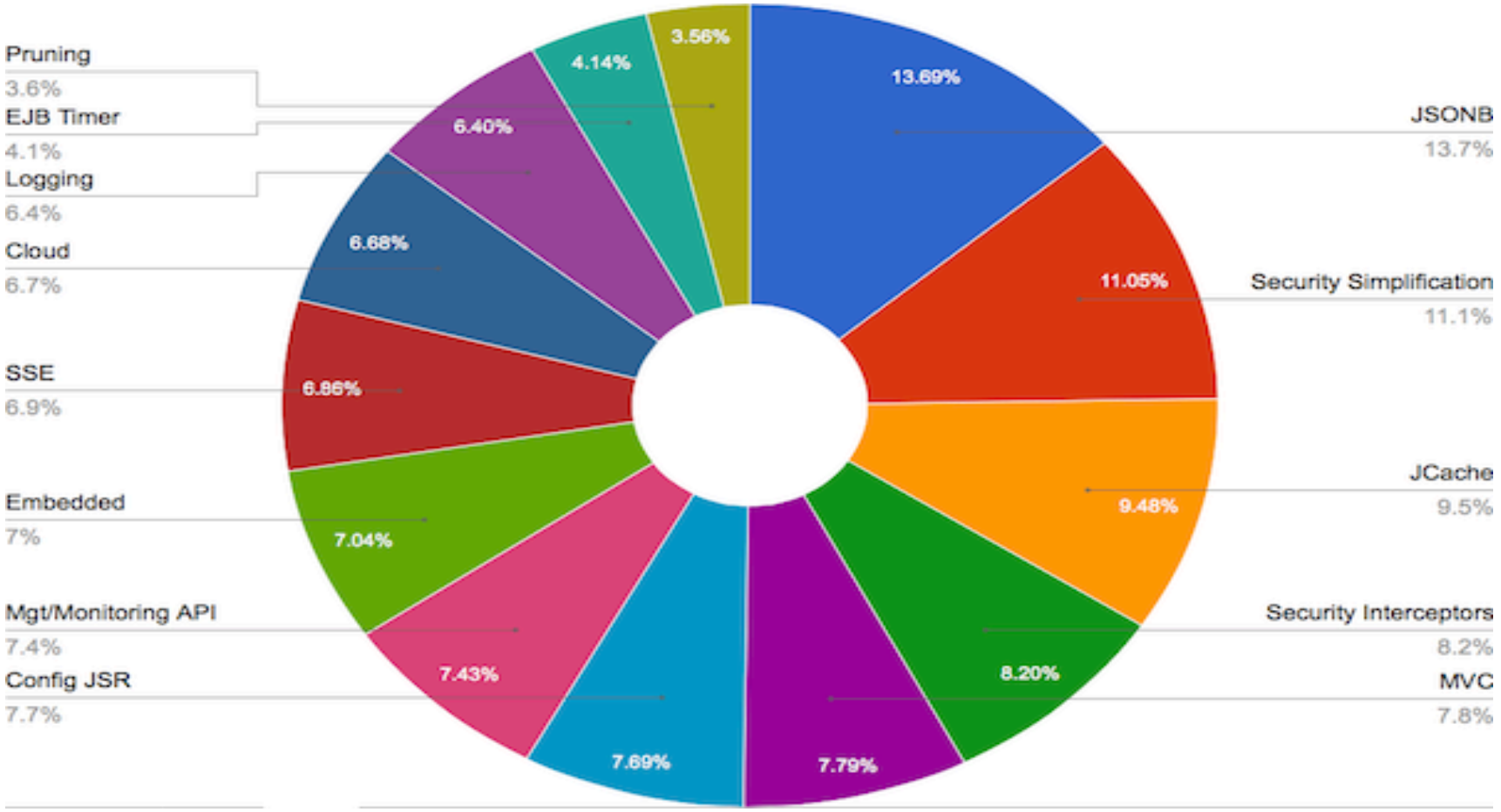
- WebSockets
- JSON
- Servlet 3.1 NIO
- REST

- Batch
- Concurrency
- Simplified JMS

Java EE 7



Java EE 8 Community Survey



https://blogs.oracle.com/ldemichiel/entry/results_from_the_java_ee

https://java.net/downloads/javaee-spec/JavaEE8_Community_Survey_Results.pdf

Java EE 8 Themes

- HTML 5 / Web Tier Enhancements
  - JSON Binding, JSON-P enhancements, SSE, action-based MVC, HTTP/2
- Ease of Development / CDI alignment
  - CDI 2, EJB services outside EJB, security interceptors, EJB pruning
- Infrastructure for running in the Cloud
  - Java EE Management 2.0, Java SE Security 1.0

Java EE Going Forward

Java Specification Request
JSR 366 – Java EE 8 Platform
JSR 365 – CDI 2.0 – (Red Hat) CDI for Java SE, modularity & events
JSR 367 – JSON-B 1.0 – JSON Binding for Java Objects
JSR 368 – JMS 2.1 – MDB Improvements, CDI Managed Bean integration
JSR 369 – Servlet 4.0 – HTTP/2
JSR 370 – JAX-RS 2.1 – NIO, Server-Sent Events, Reactive
JSR 371 – MVC 1.0 – Model View Controller, Action-Based, HTML framework
JSR 372 – JSF 2.3 – Integration with WebSocket, MVC, CDI, Java 8 DateTime
JSR 373 – Management 2.0 – REST based Management
JSR 374 – JSON-P 1.1 – Query enhancements, Java SE 8 improvements
JSR 375 – Security 1.0 – Simplifications, Cloud enhancements

Rapid Changes Over Past Few Years

Driven by increasing business needs

Microservices
Apps divided into many small pieces

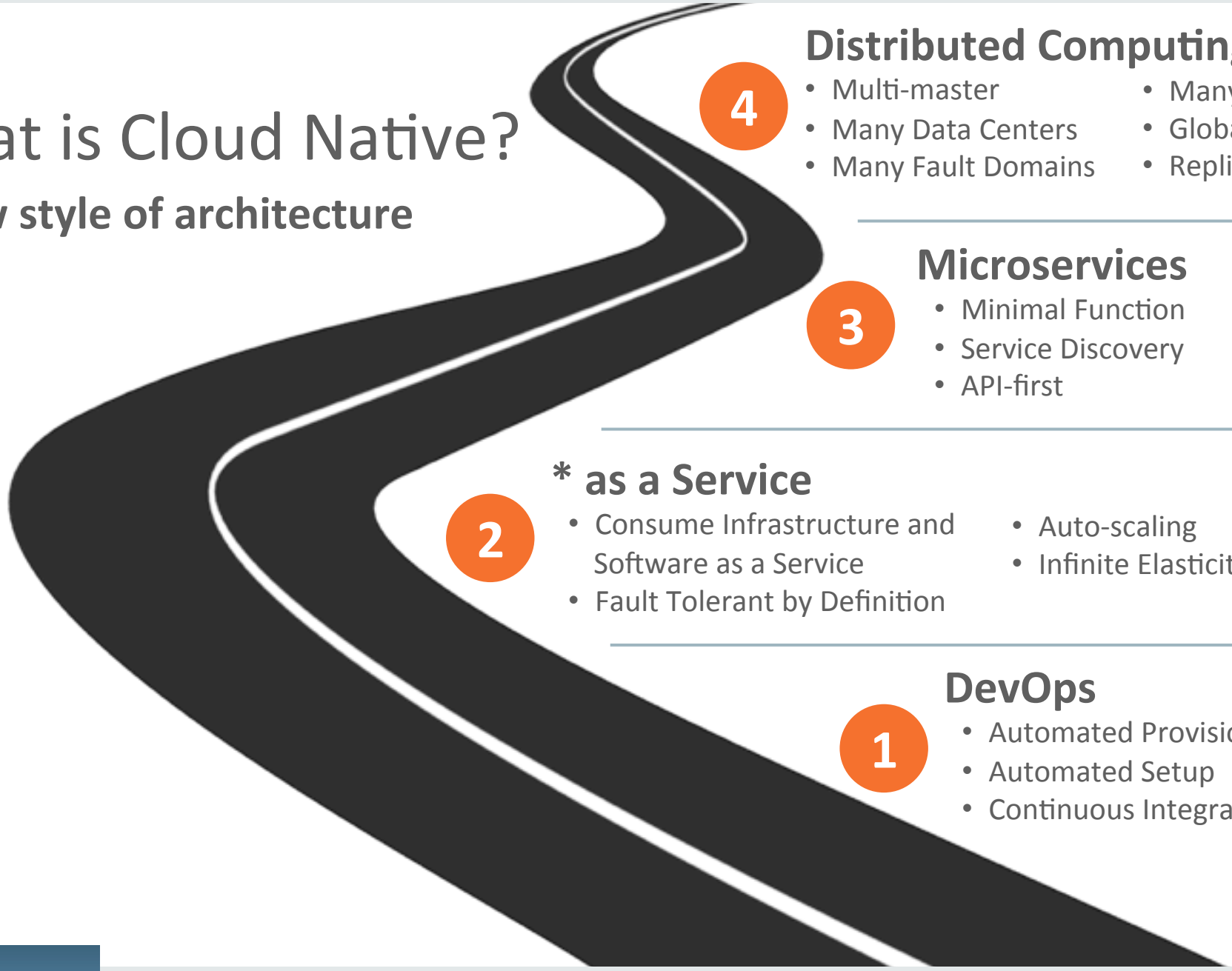
Distributed Computing
Many data centers, AZs, regions, etc

Polyglot
Java leads but use of others increasing

New Technology
Docker, Cloud, DevOps, etc

What is Cloud Native?

A new style of architecture



4

Distributed Computing

- Multi-master
- Many Data Centers
- Many Fault Domains
- Many Regions
- Global Server Load Balancing
- Replication

3

Microservices

- Minimal Function
- Service Discovery
- API-first
- Polyglot
- Choreography
- Loose Coupling

2

* as a Service

- Consume Infrastructure and Software as a Service
- Fault Tolerant by Definition
- Auto-scaling
- Infinite Elasticity

1

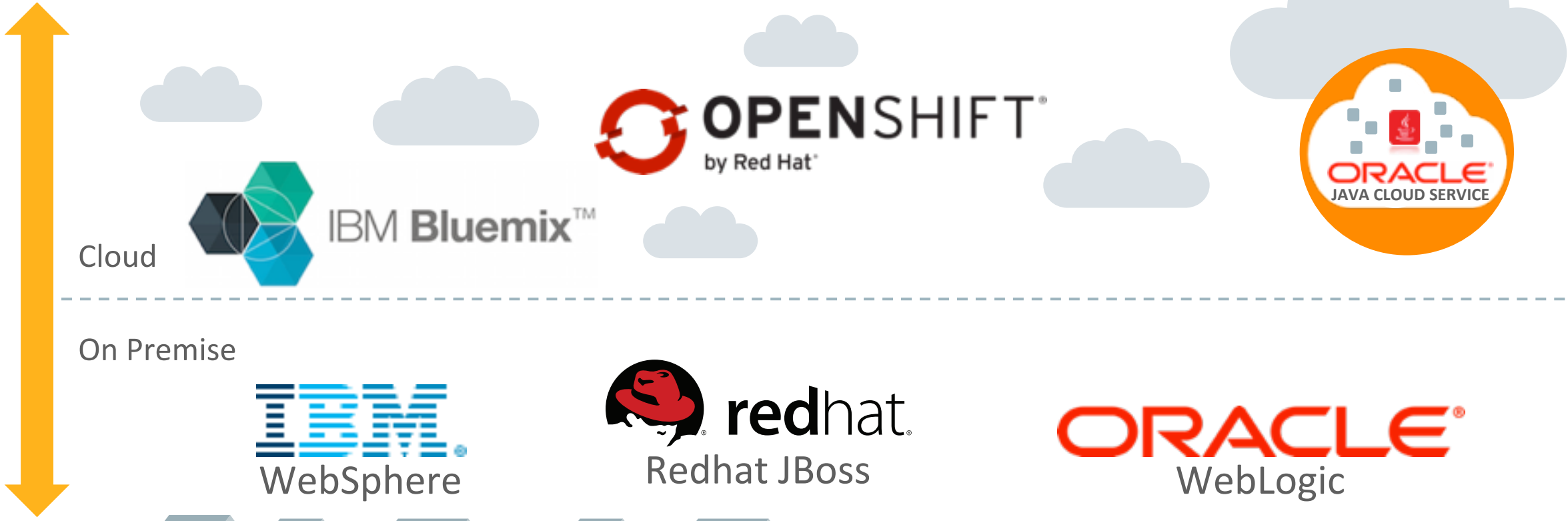
DevOps

- Automated Provisioning
- Automated Setup
- Continuous Integration
- Continuous Delivery
- Automated Testing
- Agile
- Culture Change

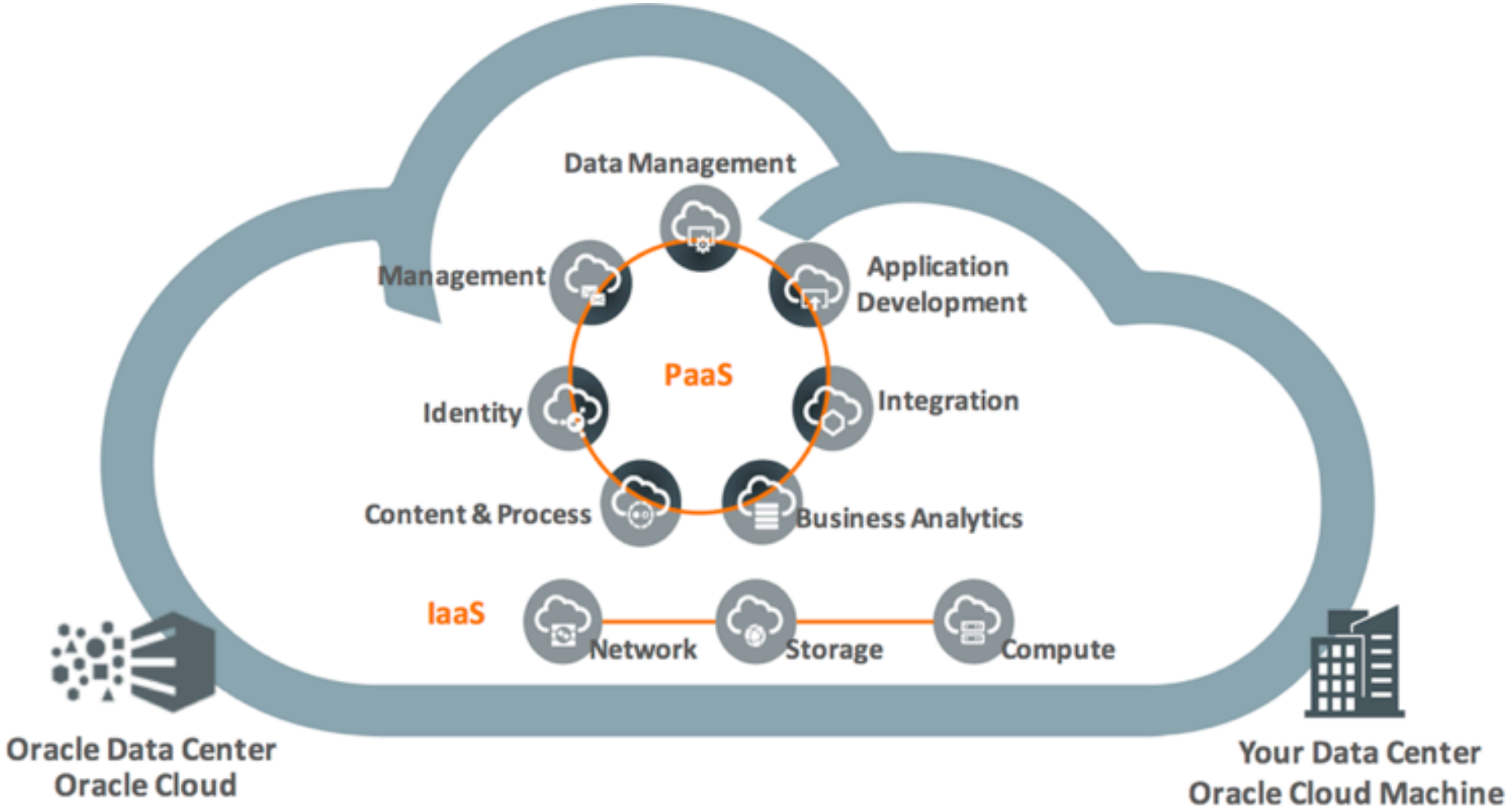
Java ORACLE

Java In The Cloud

Java EE Has Easily Transitioned into the Cloud



Oracle Cloud Platform



DATA MANAGEMENT

- Database
- NoSQL Database
- Big Data
- Big Data SQL
- Big Data Preparation
- Database Backup
- Exadata

APPLICATION DEVELOPMENT

- Java
- Application Container (Node, Java SE)
- Application Builder
- Developer
- Mobile

IT OPERATIONS MANAGEMENT

- IT Analytics
- Log Analytics
- Application Performance Monitoring

INTEGRATION

- Integration
- SOA
- API Manager
- Internet of Things
- GoldenGate

IDENTITY MANAGEMENT

- Identity

MOBILE

- Mobile
- Integration
- Internet of Things

CONTENT & PROCESS

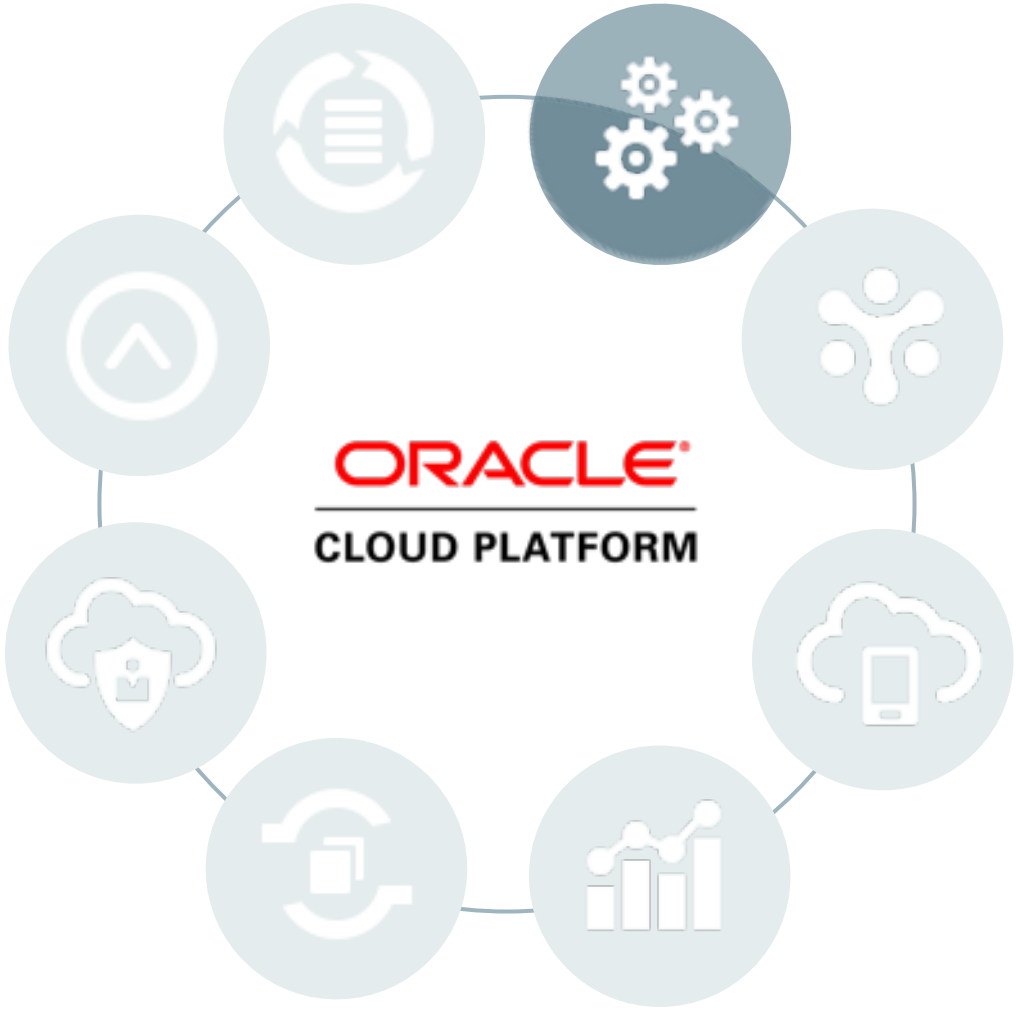
- Documents
- Social
- Process
- Sites

BUSINESS ANALYTICS

- Data Visualization
- Business Intelligence
- Big Data Discovery
- Big Data Preparation
- Internet of Things

ORACLE

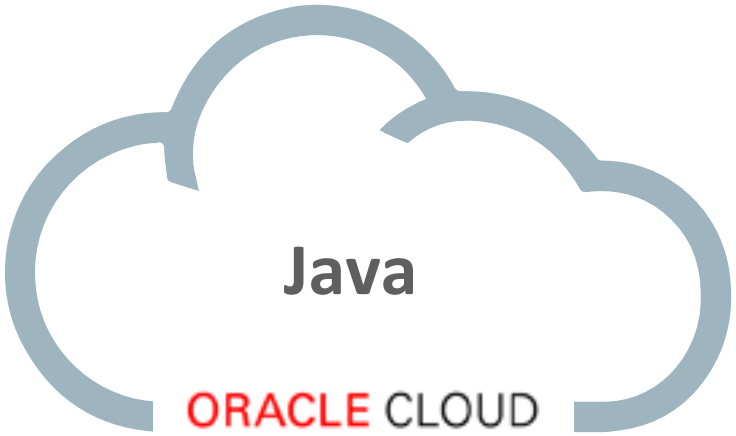
Oracle Cloud Platform: Application Development Services



APPLICATION DEVELOPMENT

- Java
- Application Container (Node, Java SE)
- Developer
- Mobile
- Application Builder

Java Cloud: For Java EE Workloads

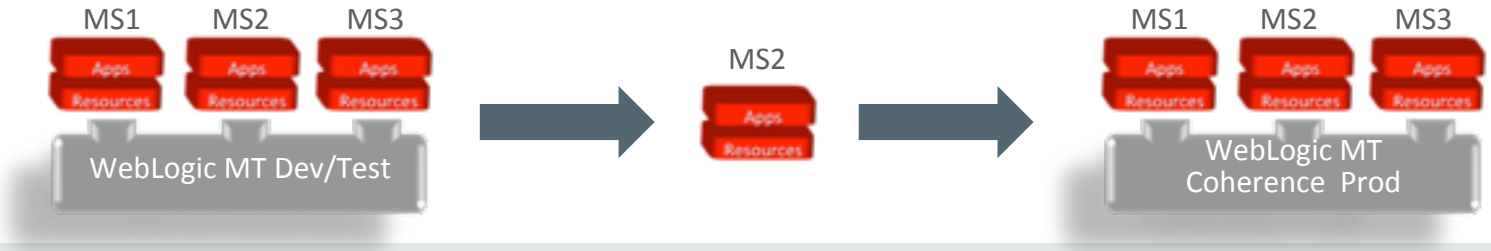


Key Features

- Full-Featured: WebLogic 11g or 12c Instance
- Clustering, In-Memory, High Availability, Elastic Load Balancing, Scale Up & Scale Out
- Lifecycle Tooling: Back Up/Restore, Patching, Application Server Management
- Full Portability: On-premise to Cloud

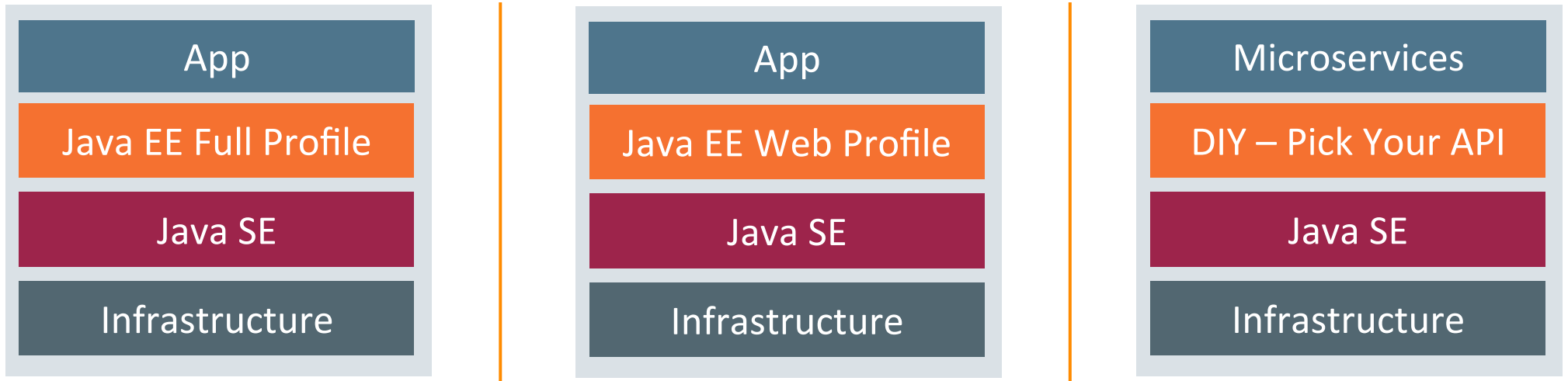
Advanced Multitenancy on Java Servers Even Enables a Modern Microservices Architecture on Java EE

- Java EE Multitenancy Characteristics
  - Partitions give service isolation (CPU/Memory)
  - Partitions have independent service lifecycles (start/stop/import/export)
  - Partitions are easily ported between environments (DevOps friendly)
  - Partitions can be aligned with pluggable DB for app/data alignment
- New applications can be modeled as services in Multitenant Partitions
  - Partition 1 = microservice 1; Partition 2 = microservice 2 etc
  - Partitions are extremely lightweight and immutable - low memory, fast startup
  - Partitions can be plugged easily into DevOps pipelines to drive fast release cycles



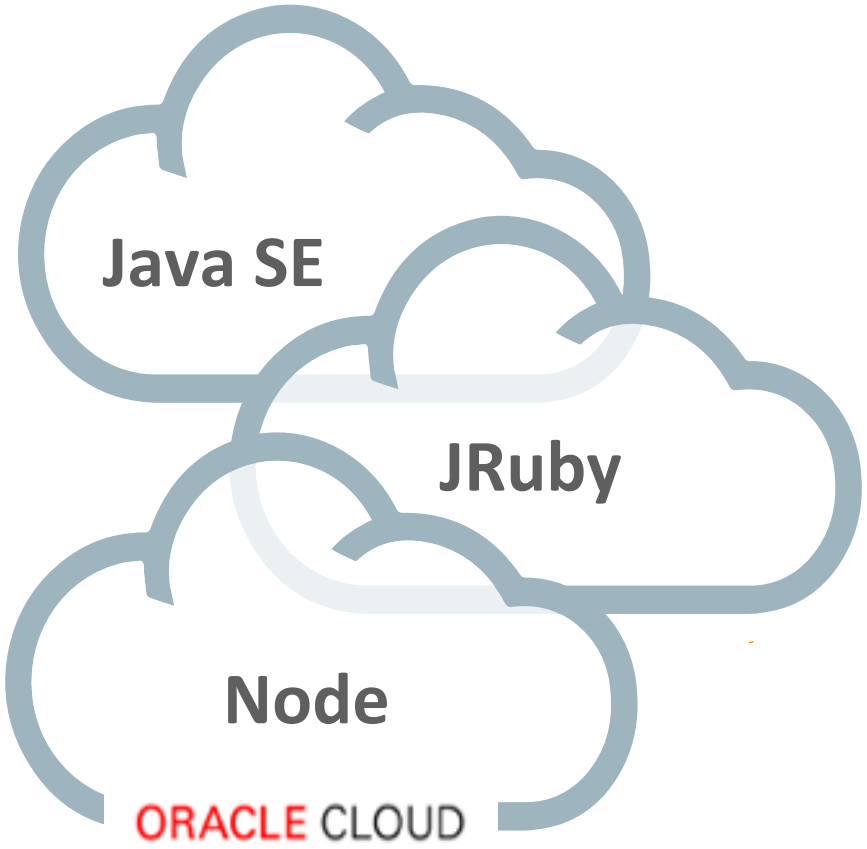
Microservices Built on Java

Some on Full Java EE, Some on Java Web Profile, Many Aiming at Fat Jar's on Java SE



Enterprise Standard Runtime for REST, JSON, Tx, JMS, EJB	**Lightweight Standard Runtime** for HTTP, REST, JSON…	**Even *Lighter* Non-standard Runtime** for microservices
GlassFish, WebLogic Wildfly, Liberty	GlassFish, WebLogic Wildfly, Liberty	Spring Boot, DropWizard, KumuluzEE etc

Application Container Cloud: For Other Workloads



An open
highly available
Docker container-based
elastic
polyglot
cloud platform

Key Features

- Useful for any Java or Node.js Framework (other languages such as Ruby, Python, PHP etc to come)
- Cloud tooling for lifecycle management
- Fully automated provisioning, patching, backup, and recovery

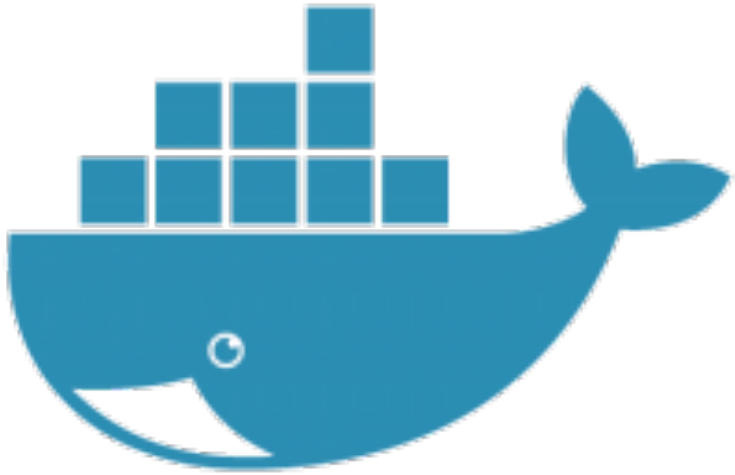
Polyglot Platform

- Deploy applications to a selection of popular language runtimes supported
  - Initial support for Java SE and Node.js
- Leverage unique Oracle Java SE features
  - Immediate access to platform upgrades, security, platform optimizations
  - Continued commercial support for Java SE versions no longer receiving public updates
- Node access to Oracle DB with open source database driver

*Python, Ruby, PHP, & Java EE coming soon

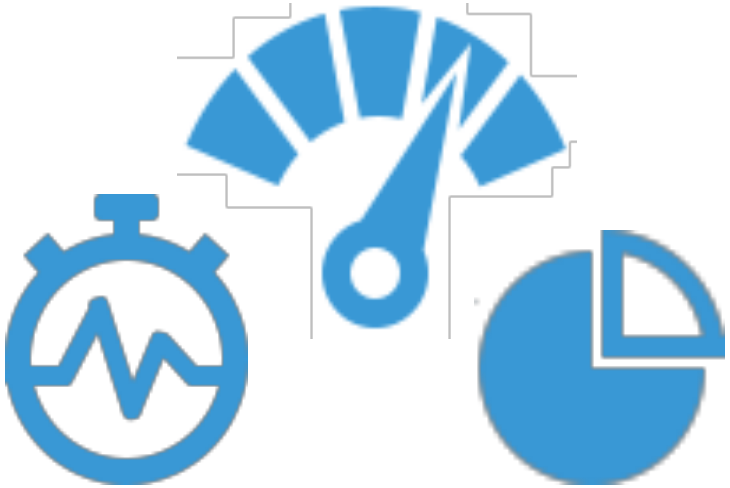
Open Platform

Container-based Application Platform as a Service



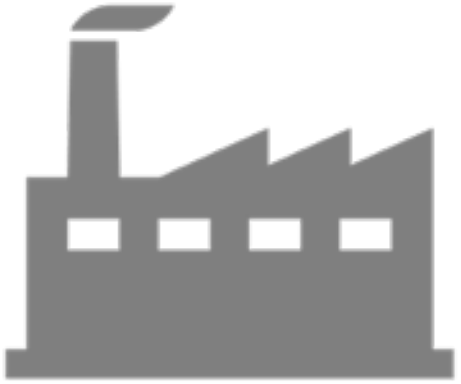
- Applications run on Oracle Linux in Docker containers
- Stateless Applications
  - Ephemeral disk
  - Permanent storage through database or storage service

Profiling

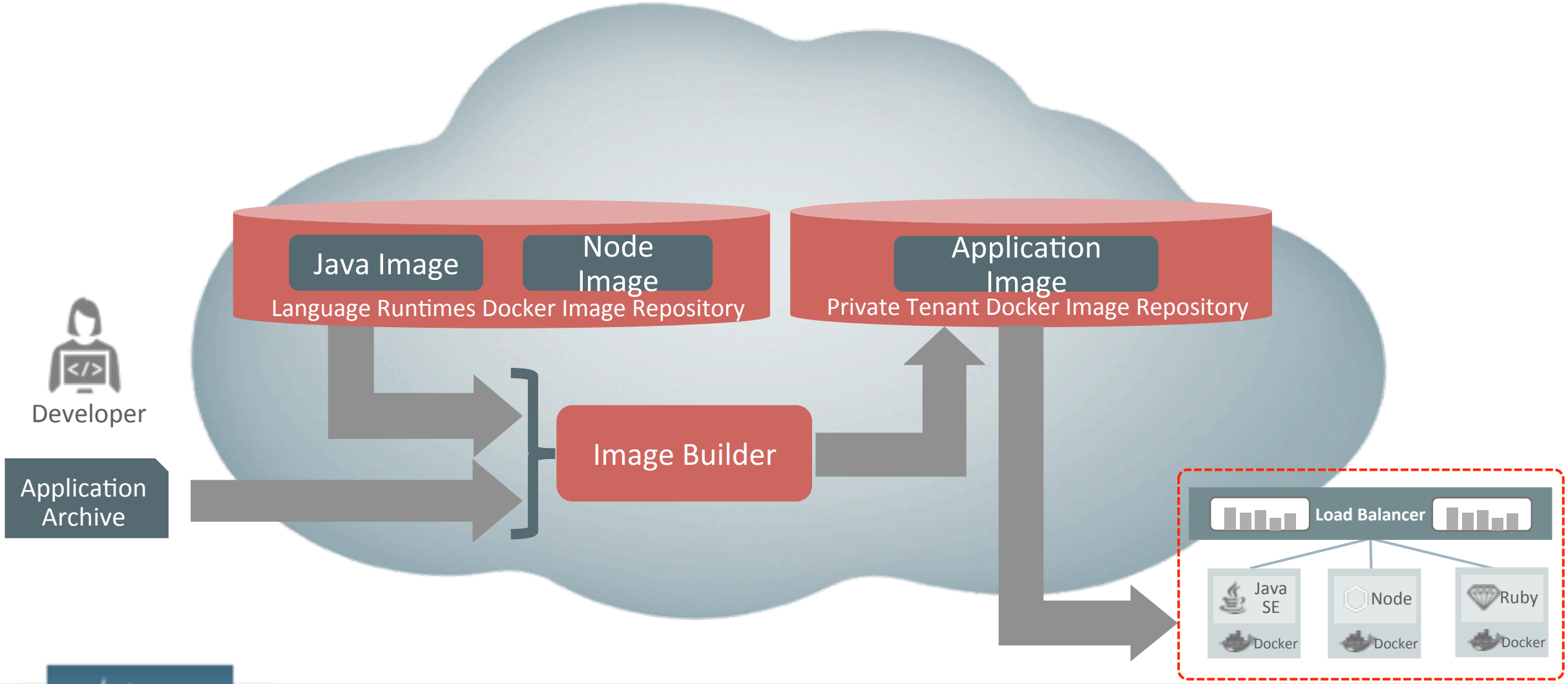


- Java application can use Java Flight Recorder to monitor application and JVM behavior and analyze in Mission Control
- Use Application Performance Monitoring Cloud Service for advanced use cases

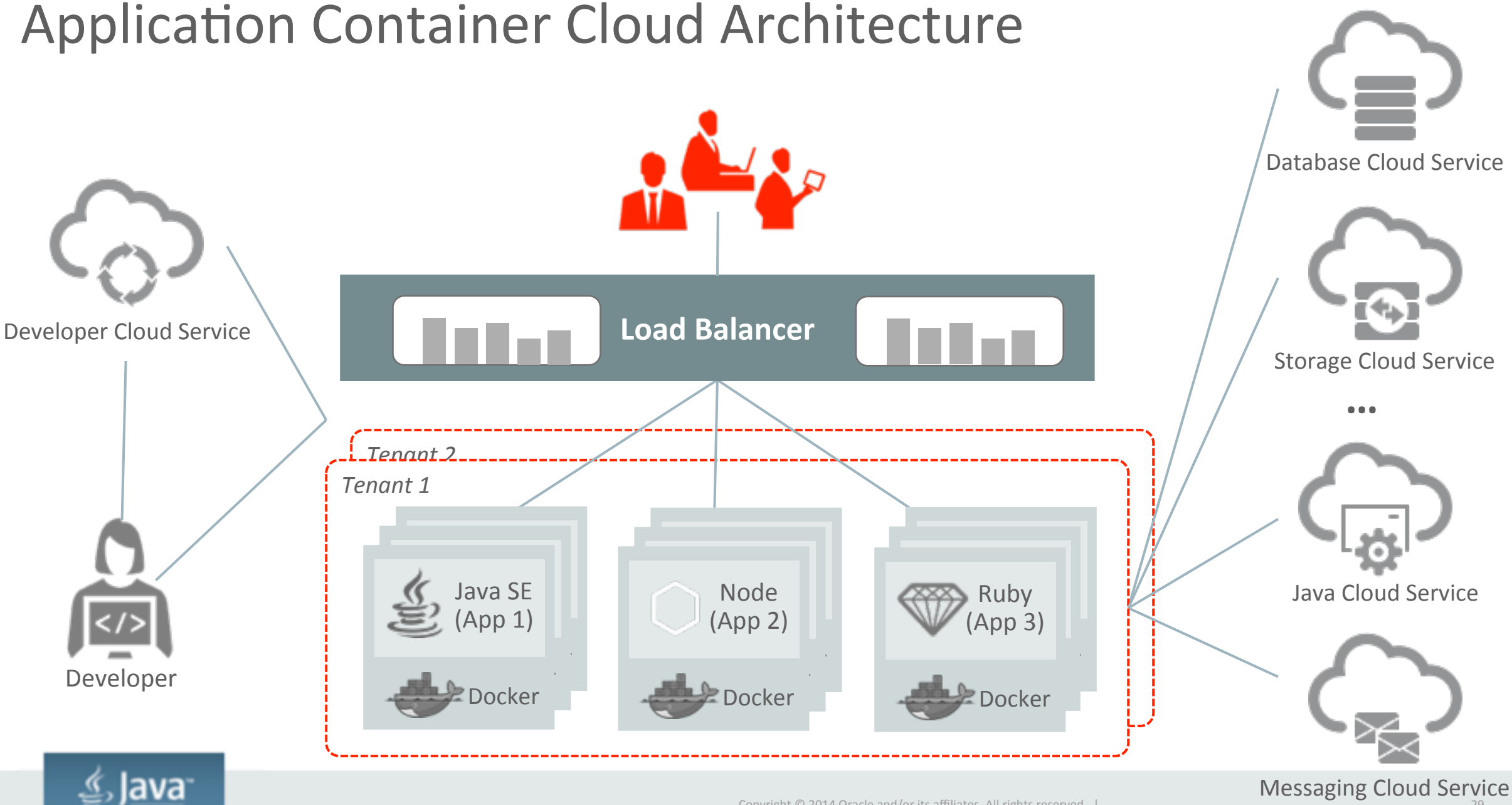
Build Zip Deploy!



Application Deployment

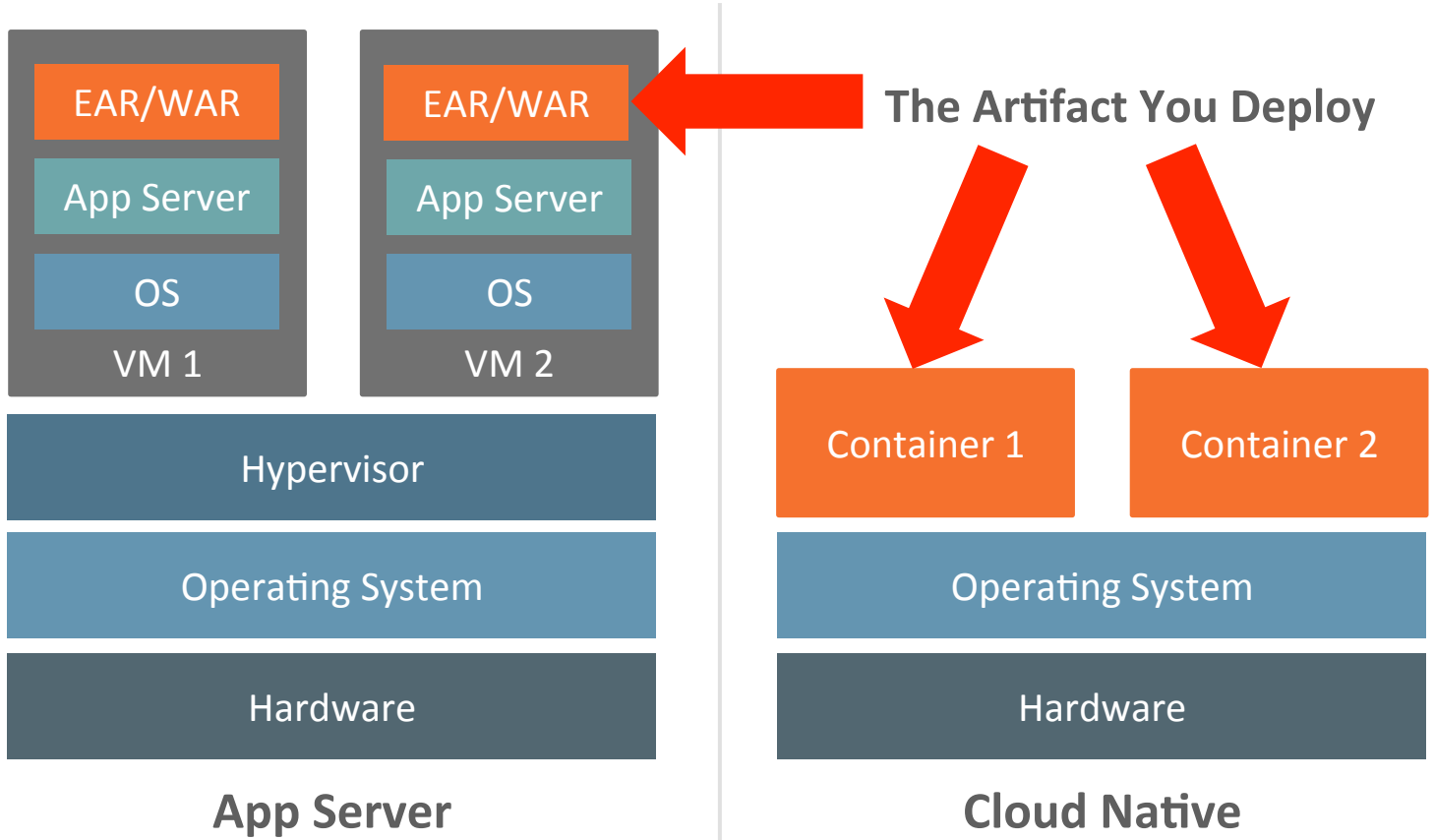


Application Container Cloud Architecture



Artifacts Are Now Immutable Containers – Not EARs, WARs

Containers can have anything in them and are highly portable



- No more installing a JVM, app server, and then deploying the artifacts to them
- Build the container once, deploy it anywhere. Can include complex environment variables, scripts, etc
- Containers should be free of state and configuration
- Containers should not assume they are able to write to a persistent local file system

Application Containers – Use What You Have Today

Characteristics of Existing Application Server

- Run a very large, complicated monolithic application
- Many advanced features
- Integrated dependency management
- Few instances per host
- Stateful

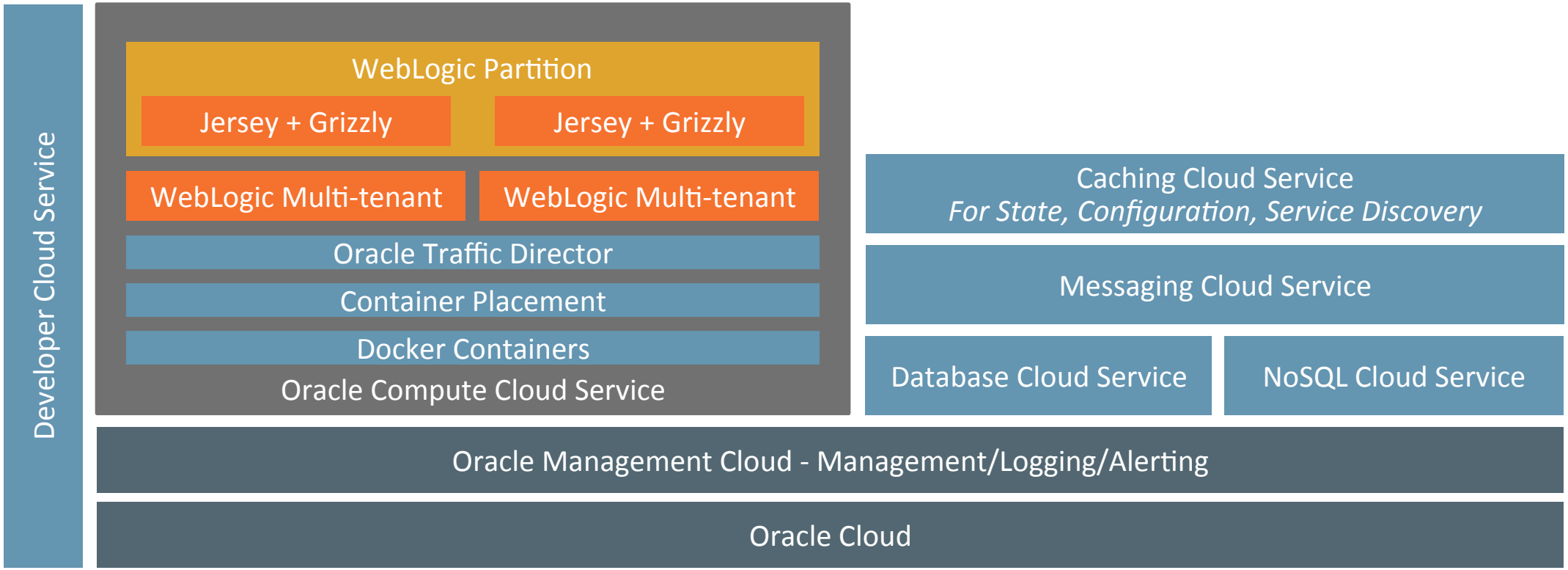
Characteristics of Microservices-friendly Application Server

- Run a smaller, microservice that does one thing really well
- Few features
- External dependency management
- Many instances per host
- Stateless

Run whatever works best – few firm requirements

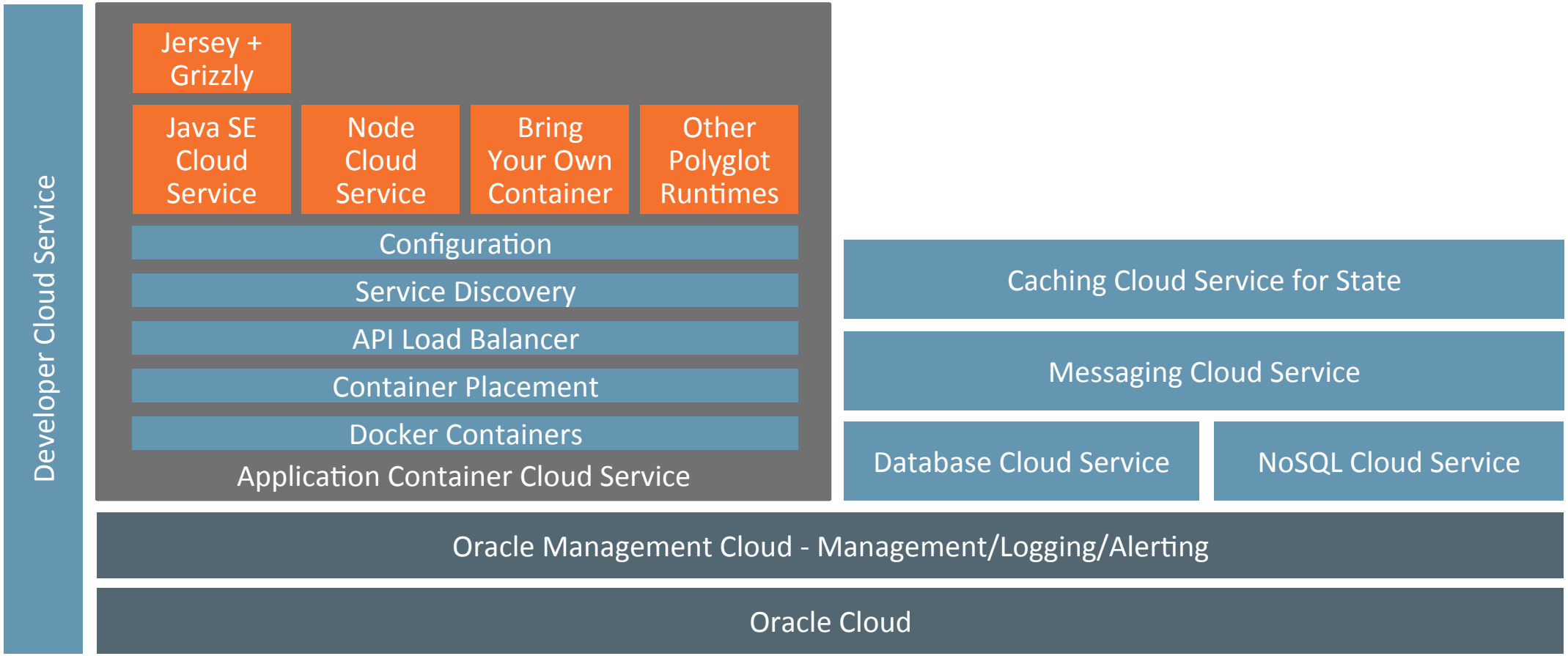
Using WebLogic + Java EE For Microservices

A proven, enterprise-grade platform for application development



Using Application Container Cloud Service For Microservices

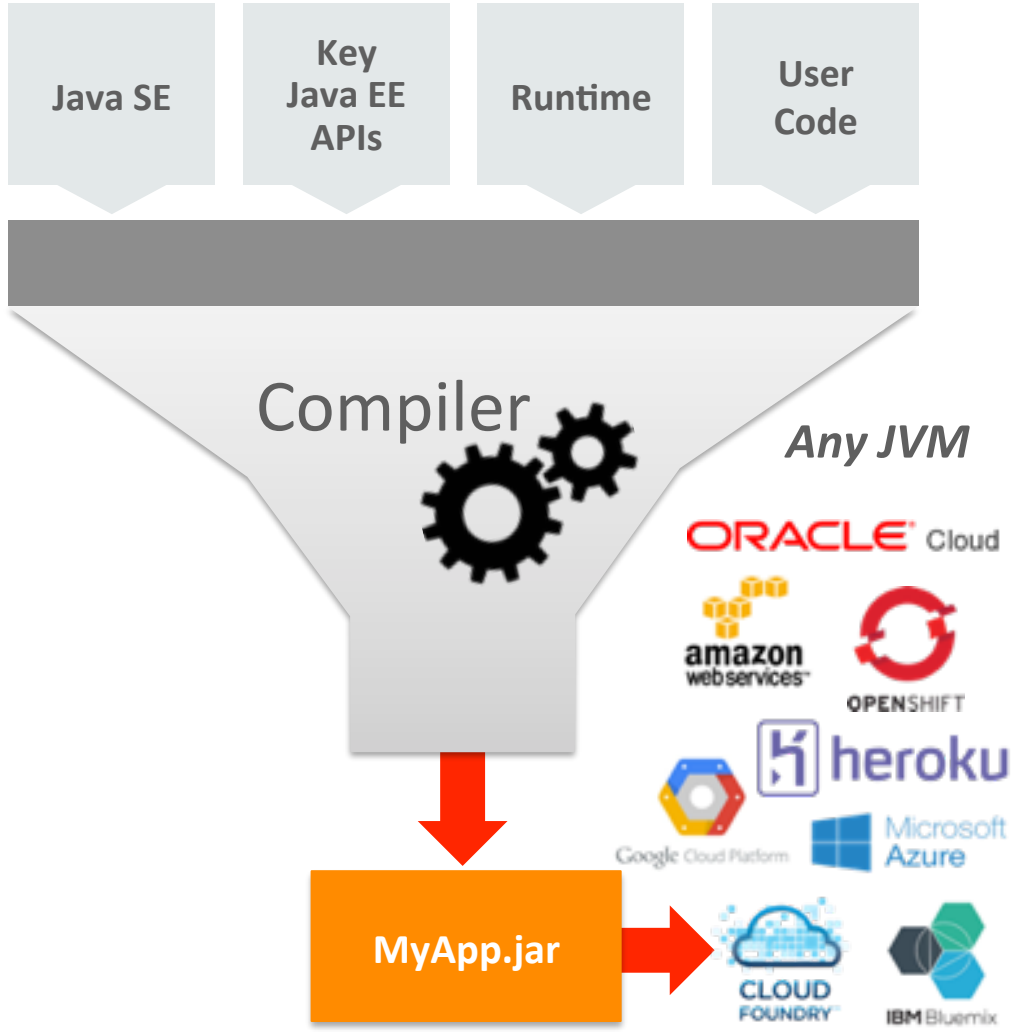
A modern platform for lightweight application development



Next Up: A Next Generation Application Runtime

As part of a larger microservices platform

- Take core Java EE APIS - Jersey/JAX-RS, Tyrus/ WebSocket and Grizzly/NIO - as a foundation
- Foundational to a new style microservices platform
- Key capabilities and design goals:
  - Integrated circuit breaker
  - Simple packaging model – Java -jar
  - Fully modular (Jigsaw) design, to allow for extensibility
  - Deep Swagger integration – optionally API first
  - Deep Maven integration
  - Easily cap resource utilization - CPU, memory
  - Programmatic builder or class-path discovery modes
  - Automatic registration of transitive Module dependencies
  - Concurrency API - context propagation, managed executor services



Oracle's Microservices Strategy

	Oracle's Microservices Roadmap	Today With Oracle Products
State	**Planned** - Oracle Cloud State Service	Oracle Coherence or Oracle WebLogic
Configuration	**Planned** - Oracle Cloud Config Service	Oracle Coherence
Runtime	**Planned** - Runtime - Jersey + Grizzly	Oracle WebLogic, Node, Java SE, etc
Eventing	**Planned** - Oracle Cloud Eventing Service	Oracle Coherence
Messaging	Oracle Messaging Cloud Service	Oracle Messaging Cloud Service
Management/Logging/Alerting	Oracle Management Cloud Service	Oracle Management Cloud Service
Datastore	Oracle Database or NoSQL Cloud Service	Oracle Database or NoSQL Cloud Service
Central Source of Truth	**Planned** - Oracle Microservices Platform	Oracle Coherence
Service Discovery	**Planned** - Oracle Microservices Platform	Oracle App Container Cloud Service
API Gateway/Load Balancer	**Planned** - Oracle Microservices Platform	Oracle App Container Cloud Service
Container Orchestration	**Planned** - Oracle Microservices Platform	Oracle App Container Cloud Service
Infrastructure	Oracle Cloud	Oracle Cloud
Build/Deploy	Oracle Developer Cloud Service	Oracle Developer Cloud Service

An All-New Microservices Platform From Oracle

Coming Soon!

All New

Java Runtime
Light/fast
Jersey-based
Easy Packaging

RAILS node.js Microsoft .NET python Scala

ORACLE Microservices Platform

Rolling Upgrades
Stateless Support
Stateful Support

Rollbacks
Service Discovery
Blue/Green Releases

Canary Testing
Health Monitoring
Language-specific Tooling

Infrastructure Agnostic
Placement Constraints

High Availability
High Density

Hybrid Cloud
Internal Load Balancing

Consume Platform as PaaS on Oracle Cloud

Install Platform On Any IaaS, On Or Off Premise (future)

ORACLE Cloud — Oracle Public Cloud

amazon web services · Microsoft Azure · Google Cloud Platform

ORACLE Cloud — Oracle Public Cloud *Machine*

PureSystems · VBLOCK · vmware · openstack

ORACLE Cloud

New!

- Config
- State
- Eventing
- Messaging
- Logging
- Monitoring
- Database
- NoSQL
- Caching
- Integration
- Big Data
- Mobile
- Process
- Developer

Services Available for Consumption Inside the Platform

Java ORACLE

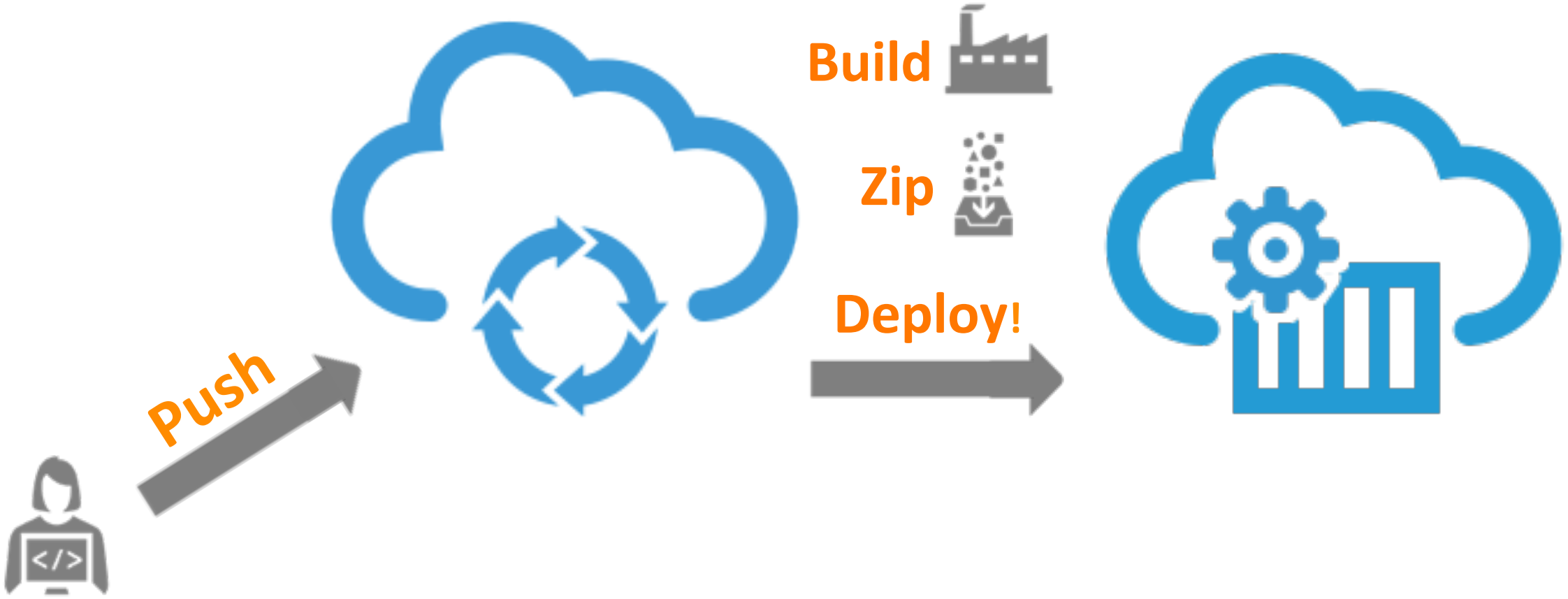
Oracle Developer Cloud Service—Included

Source Control Management

Issue Tracking

Hudson Continuous Integration

Wiki Collaboration

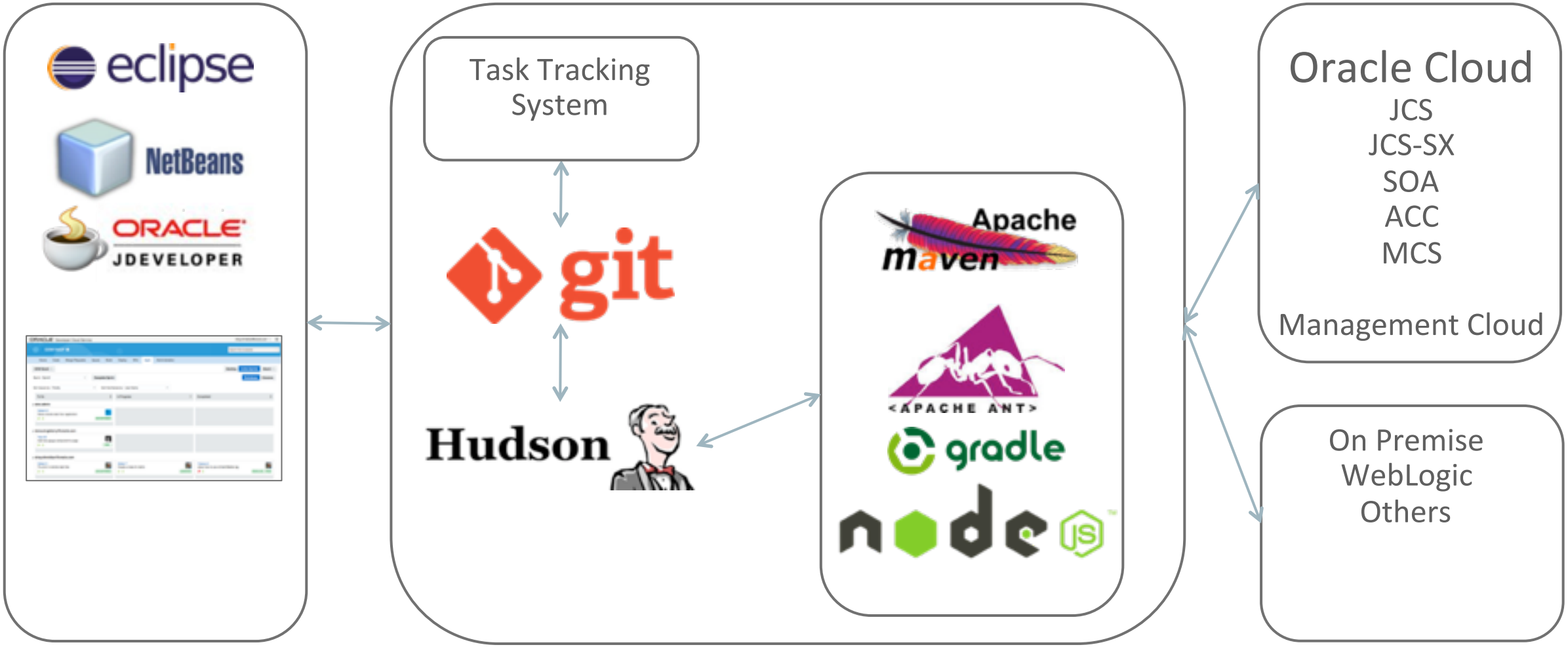


Developer Cloud Service – Easy Adoption/Integration

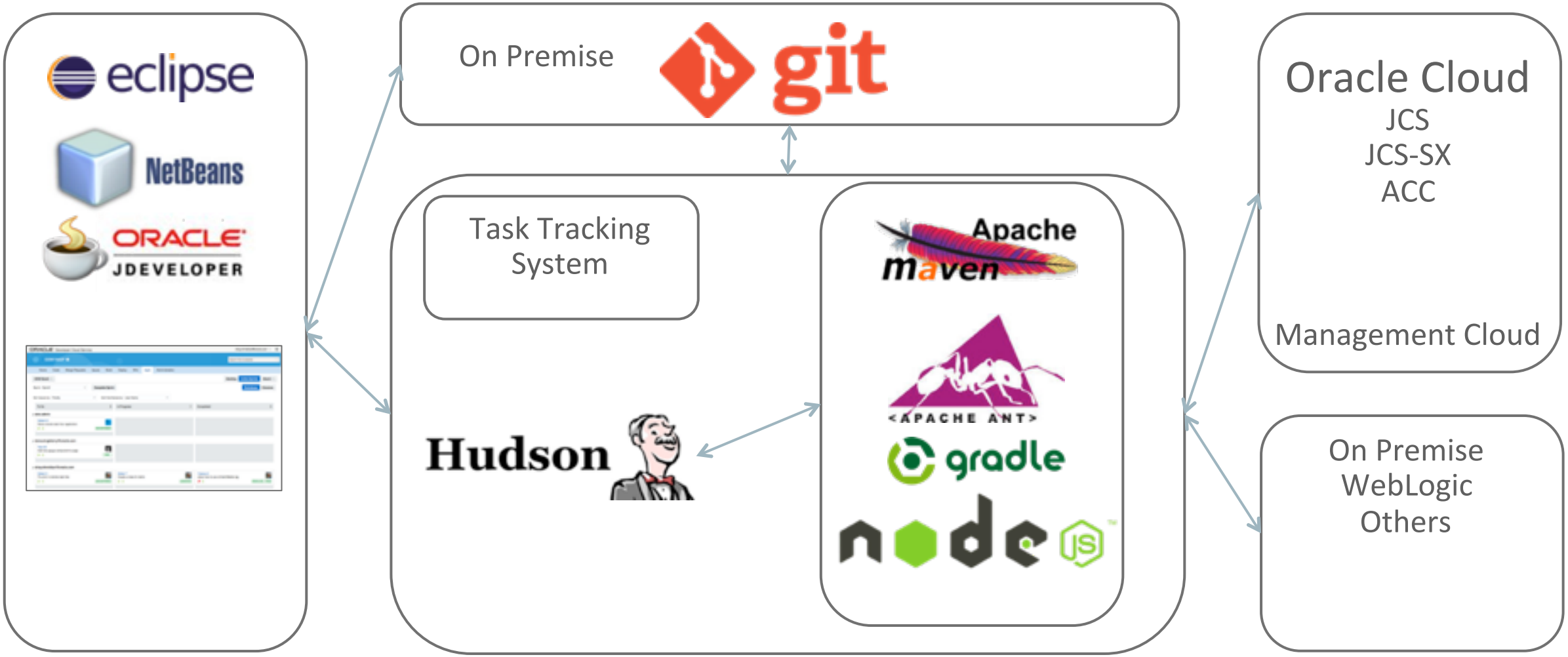
Pre-integrated development technologies in the cloud

- Standards Based
  - Git, Maven, Hudson, Ant, etc.
- Built-in IDE Integration
  - Eclipse, NetBeans, JDeveloper
- Flexible Source Location
  - Hosted Git or GitHub
- Choice of Deployment Target
  - Oracle Cloud or on-premise

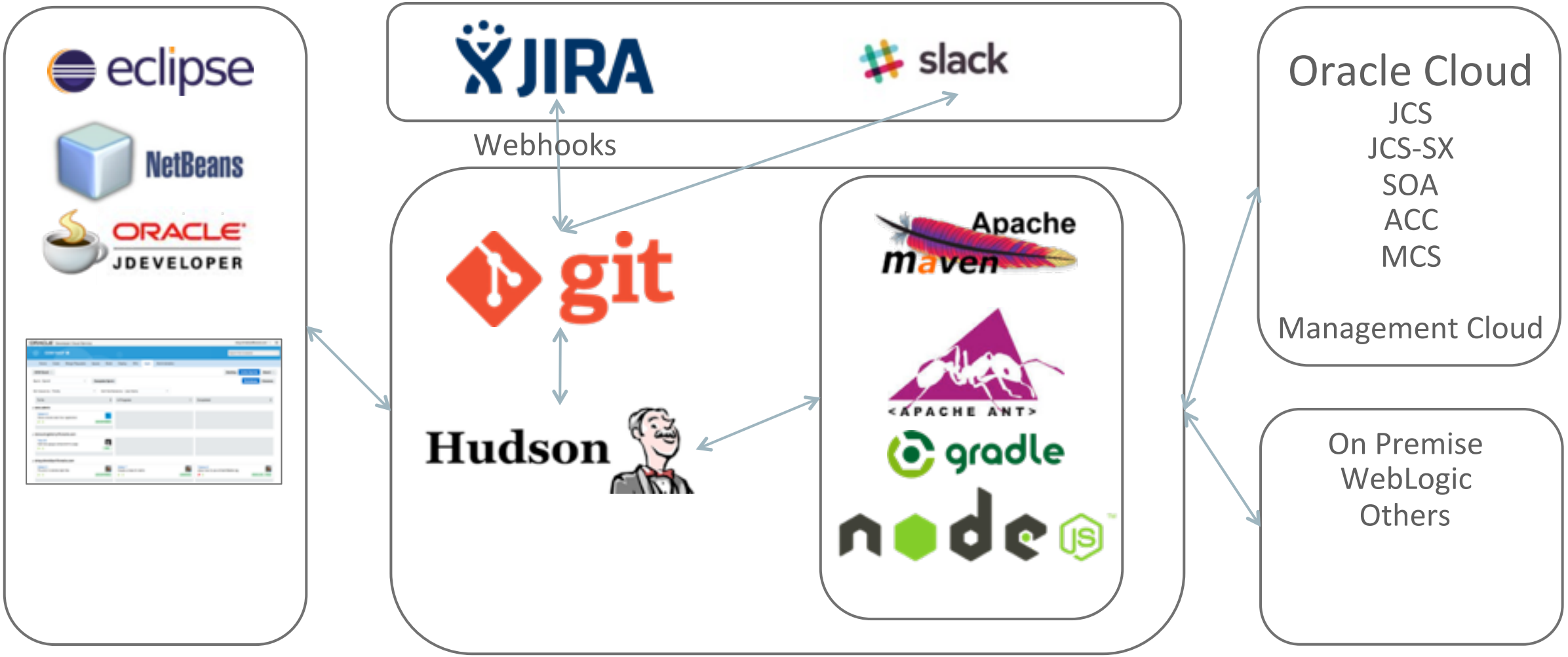
Oracle Developer Cloud Service – Base Architecture



Another Possible Implementation Architecture



Another Possible Implementation Architecture



Oracle Cloud Platform - Application Development

Cloud Native, Polyglot, Java, Mobile, HTML5, Declarative

Cloud Native, Polyglot	Modern Java	Mobile	Declarative
			
Application Container Cloud	Java Cloud/ WebLogic Server	Mobile Cloud	Application Builder Cloud

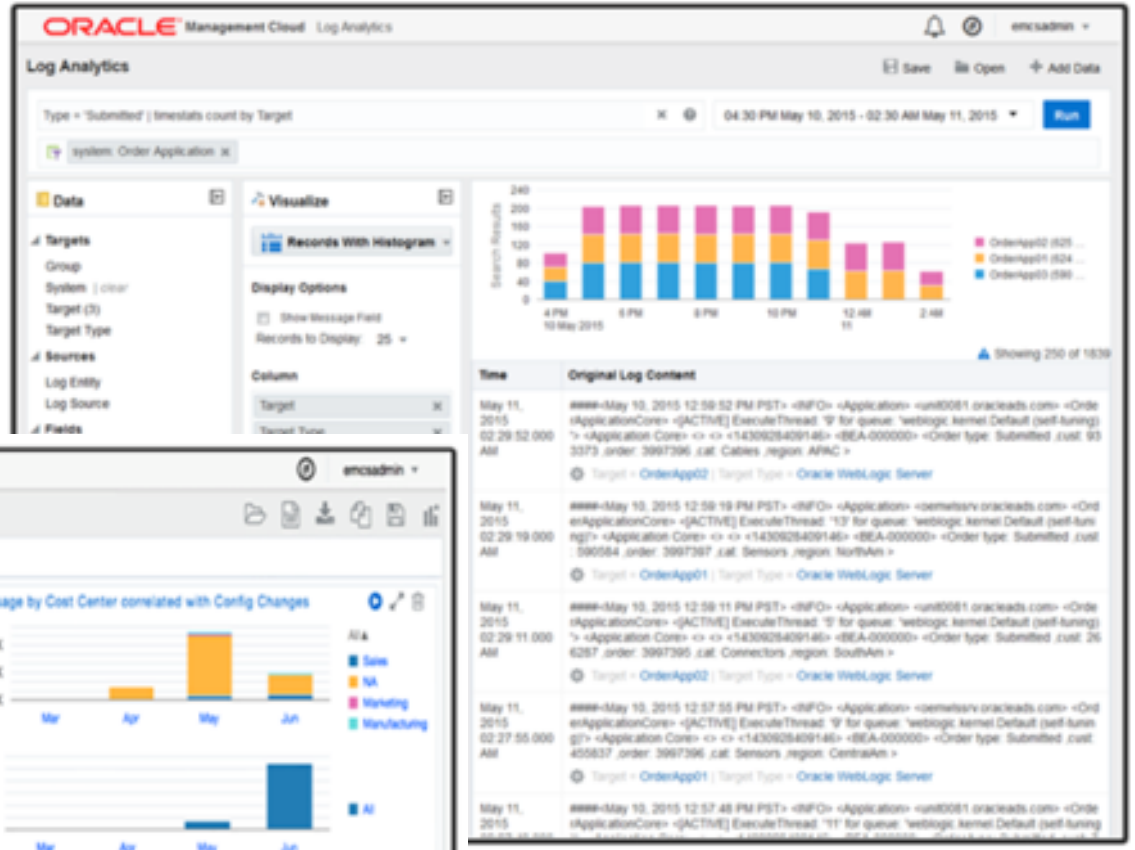
Management Cloud: For IT Operations and DevOps

Key Features

- End-user & server monitoring: web, mobile, on-premises, cloud
- Capacity and resource planning
- Estate-wide event and metric analytics
- Light-touch log aggregation with topology-aware search
- Machine Learning based anomaly detection
- Out-of-box dashboards

Benefits

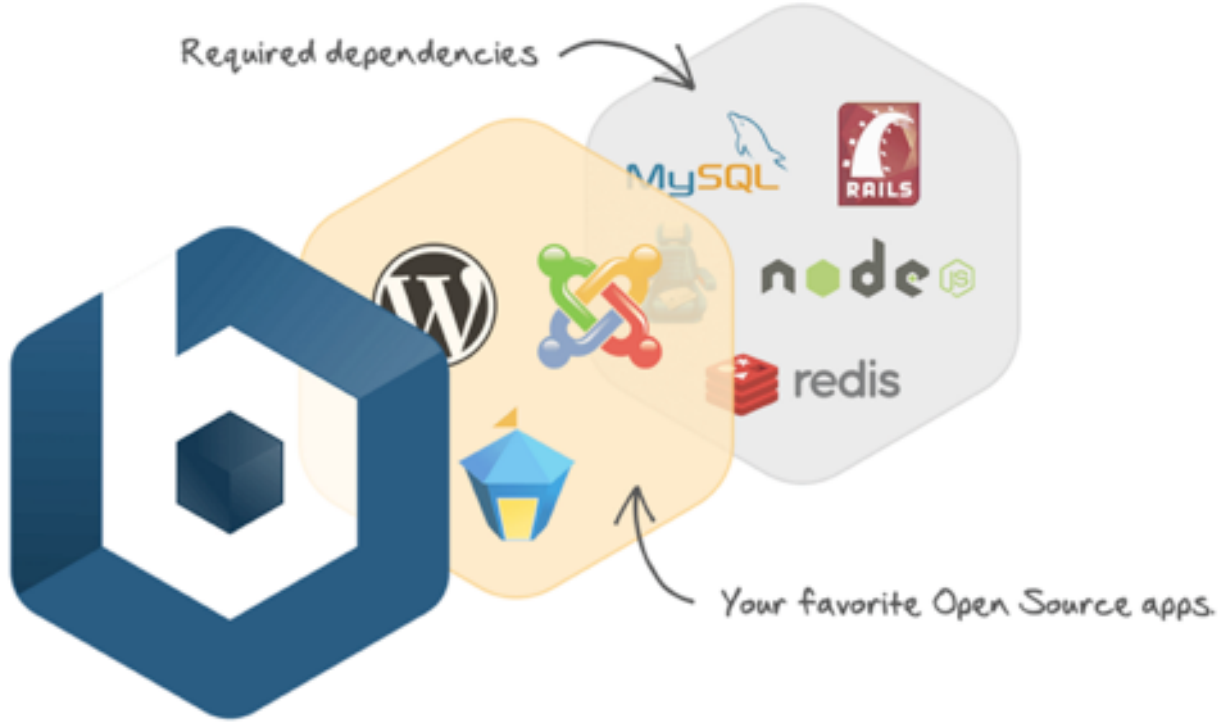
- Assure good user experience & rapidly troubleshoot issues
- Manage complex applications (heterogeneous, multi-cloud, etc.)
- Eliminate operational data silos and encourage DevOps
- Get started quickly and easily with lightweight deployment



Bitnami

Popular images, provided by Bitnami, ready to launch on Oracle Cloud Platform in one click.

https://oracle.bitnami.com



ORACLE

Product Development IT (PDIT)

Supporting enterprise, development IT, cloud and managed hosting at Oracle

- 602 Exadata
- 383 Exalogic
- 69,290 Sun x86 servers
- 6,854 SPARC servers
- 1,768 Sun ZFS appliances
- Thousands of other systems

924 Oracle Products
135,000 Employees

How Did PDIT Improve Efficiency by > 2x in 18 Months?

Culture changes enabled by technology

Culture
(Campbell Webb, Sr. VP, Oracle)

+

ORACLE PaaS

ORACLE IaaS

Cloud
The Same Products Oracle Sells

+

ORACLE ENTERPRISE MANAGER 12c

PaaS Automation

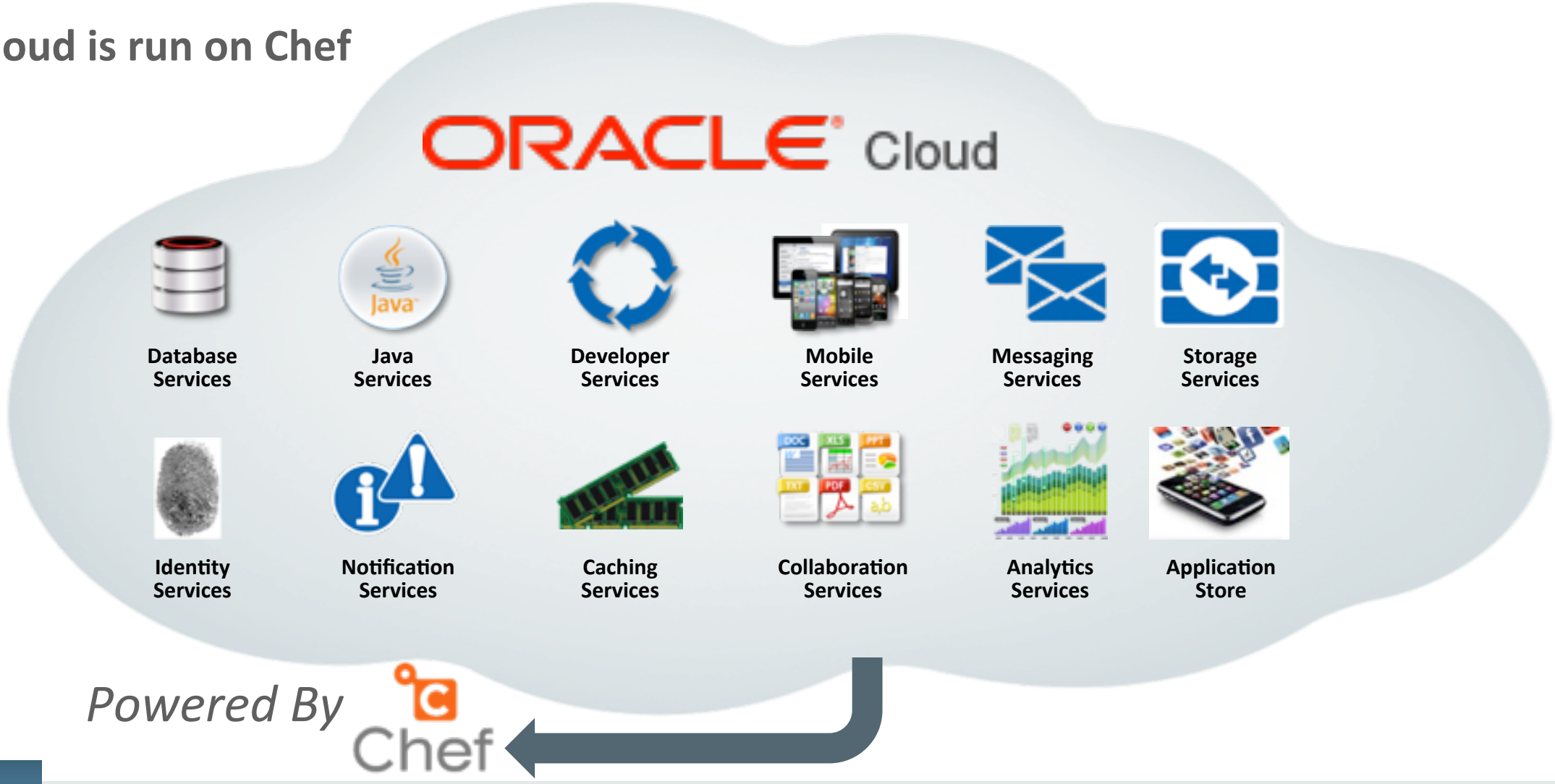
puppet labs

Chef

Infrastructure Automation

Oracle Cloud is Provisioned Using Chef

Oracle Cloud is run on Chef



Java

JCP

Lambda

Java Champions

Java Card

Collaboration

Java SE

Jigsaw

Java ME

Big Data

NetBeans

Security

Community

MOOC

OTN

Java Magazine

Java Duchess

Internet of Things

JEPs

Tooling

Adopt-a-JSR

Java Embedded

OpenJDK

Nashorn

Java EE

Cloud

APIs

Performance

Java User Groups

JavaOne
ORACLE

Q & A

Acknowledgements

- Oracle PM
  In particular Mike Lehmann, Shaun Smith,
  Kelly Goetsch http://www.slideshare.net/KellyGoetsch
- Java EE stuff from
  Linda DeMichiel, David Delabassee, Reza Rahman, Arun Gupta…
- App Server history is from a much older Sun slide I had put together
  (any errors in there are all mine ☺)