
ORACLE®

ORACLE
CLOUD PLATFORM

Oracle Application Container Cloud

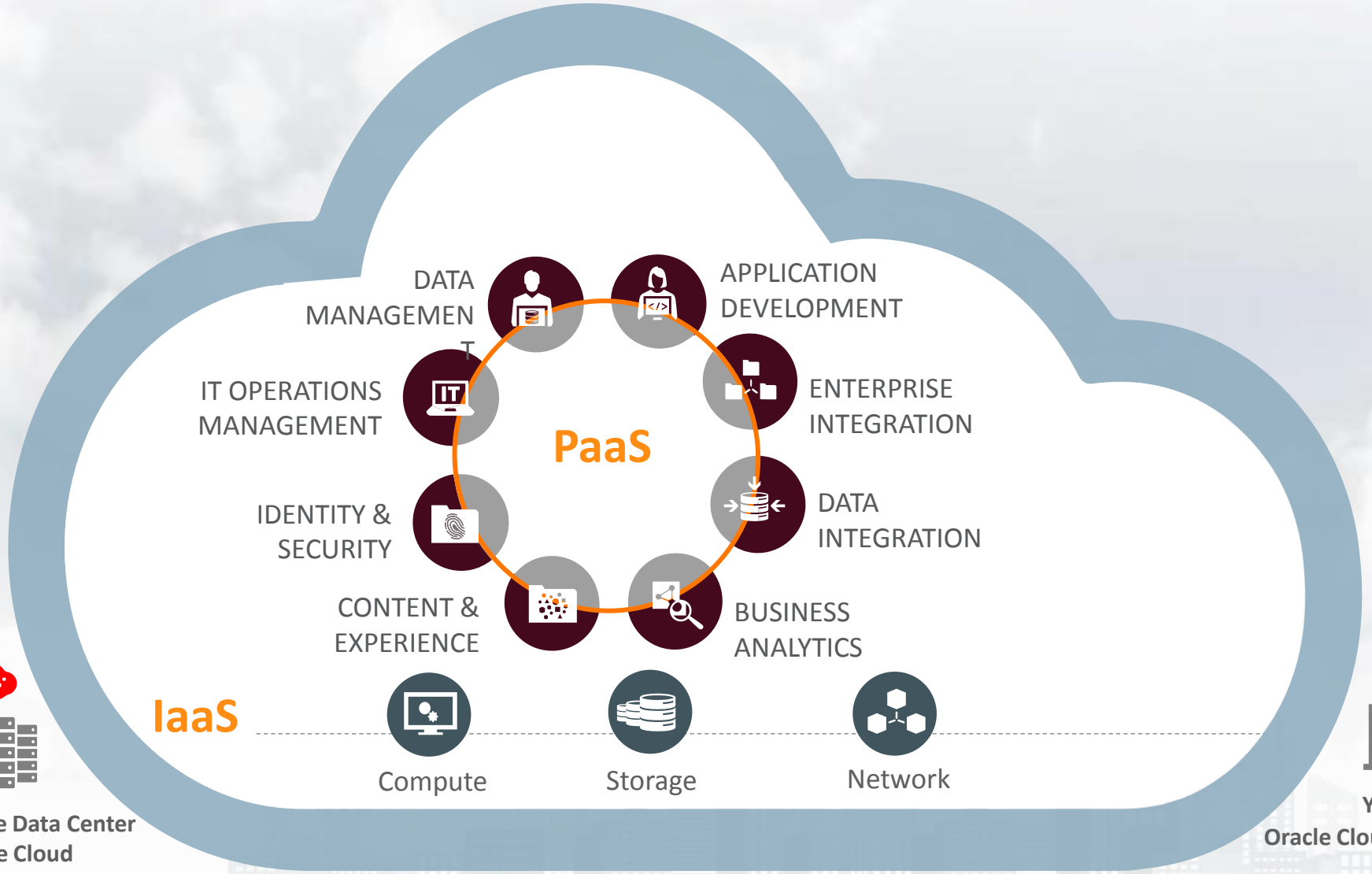
Wonjo Yoo
Principal Sales Consultant
Oracle Middleware

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.

Oracle Cloud Platform



Hybrid Cloud

Comprehensive

Integrated

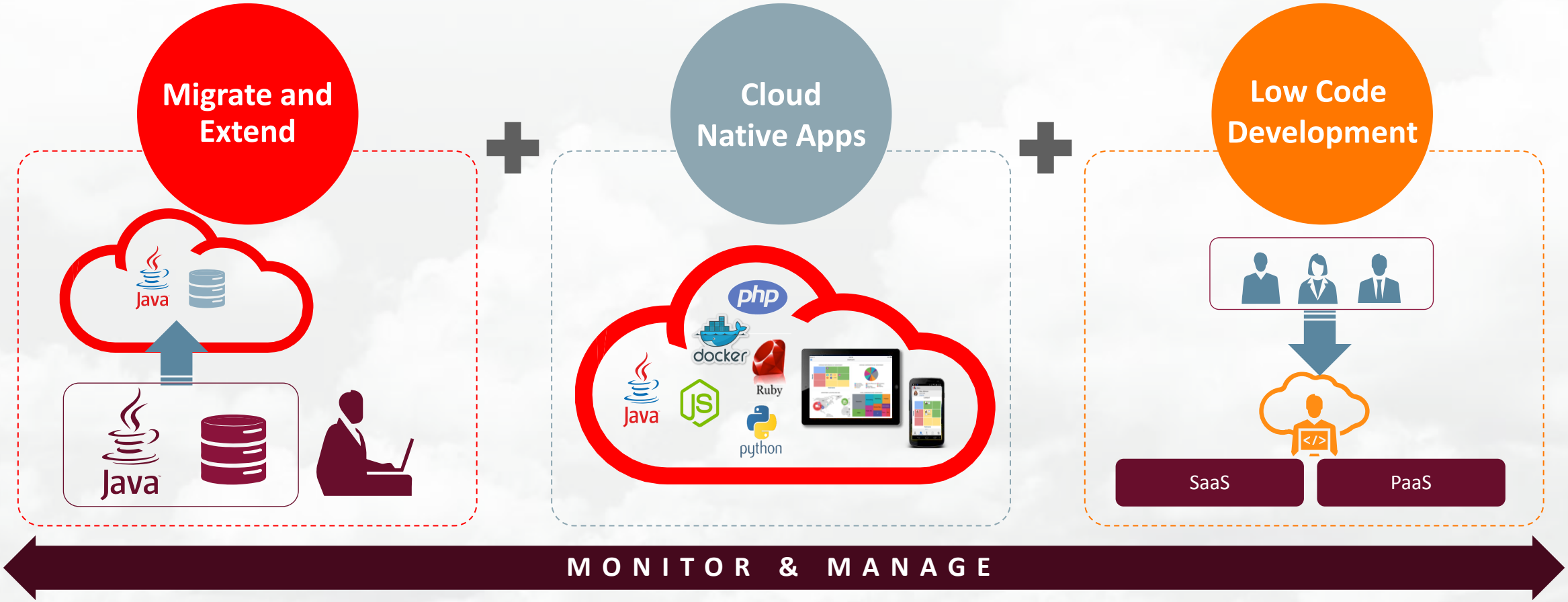
Open

Oracle Data Center
Oracle Cloud

Your Data Center
Oracle Cloud at Customers

Why is Oracle Different and Better at AppDev?

One Solution that Solves Migrate & Extend, Cloud Native and Low Code



Oracle Cloud Platform: Application Development Services

Unique in Blending Traditional, Cloud Native and Low Code with End to End PaaS

Oracle Platform for Cloud Application Development

BACK-END SERVICES

- Application Container
- Java
- Functions*
- Mobile

FRONT-END TOOLING

- SDKs (3rd Party, JET)
- App Builder

COMMON SERVICES

- Developer Cloud (CICD)
- Container Cloud (Docker, Orchestration)
- APM/Log Analytics
- Event Hub* (Kafka) Cloud
- Messaging Cloud
- API Management*
- Identity*

DATA SERVICES

- Database
- MySQL
- NoSQL

INTEGRATION SERVICES

- Integration
- SOA
- Internet of Things

Compute (VM, Bare Metal) | Network | Storage

Infrastructure as a Service

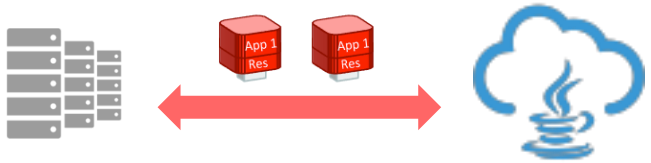
* Coming CY17

ORACLE

Oracle Cloud: A Modern AppDev Platform

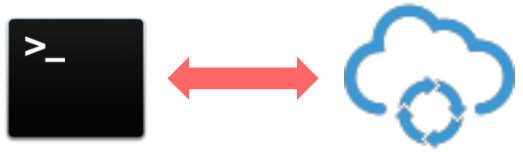
Polyglot
Lightweight Apps

Enterprise Java/DB
Automation in the Cloud

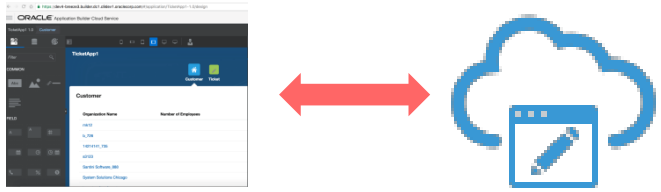


App2Cloud Migration

DevOps Automation
Developer Collatoration



Low Code
Composer – AppBuilder



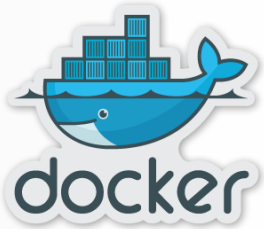
APM, Log Analytics
IT Analytics

Mobile and IoT
Platform

Requirements for Cloud Native Development

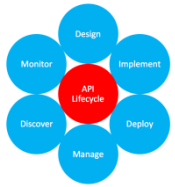
Scalable Elastic Polyglot Microservices

Containers For Simplicity & Extensibility



Automated DevOps for Agility

API First for Service Consumption



Instrument for APM and Diagnostics

Mobile First and Modern Web UX

What is Oracle's Cloud Native AppDev PaaS Solution?

Developer Cloud

Continuous Integration/Continuous Delivery

Container Cloud

Docker Container Management & Orchestration

Application Container Cloud

Polyglot Lightweight Applications

API Platform Cloud

Publish/Manage Service APIs

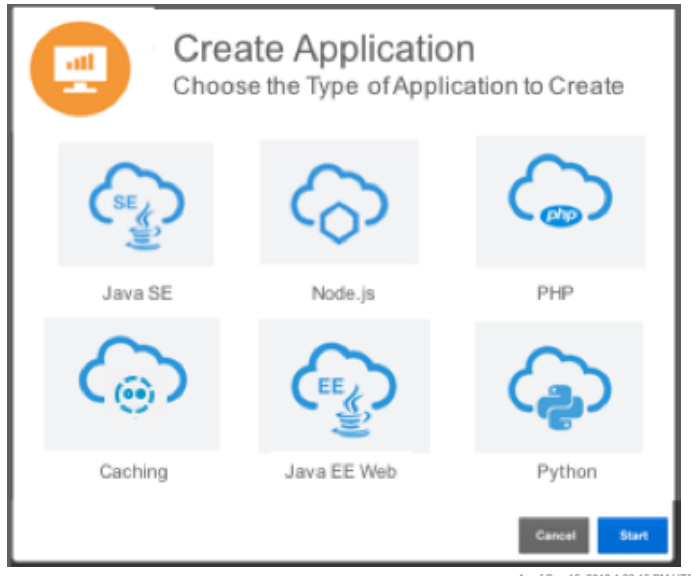
Management Cloud

APM, Security and Log Analytics

Application Container Cloud – Polyglot Cloud Native Apps

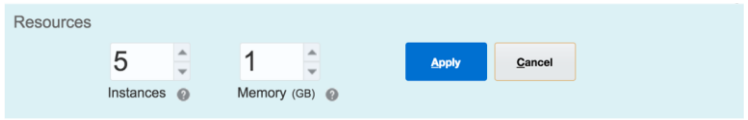
Java SE, Java EE, Node, PHP, Caching …

Cloud native, polyglot Applications



Built on Docker with Autoscaling/Elasticity

Integrated Cloud Native PaaS Services



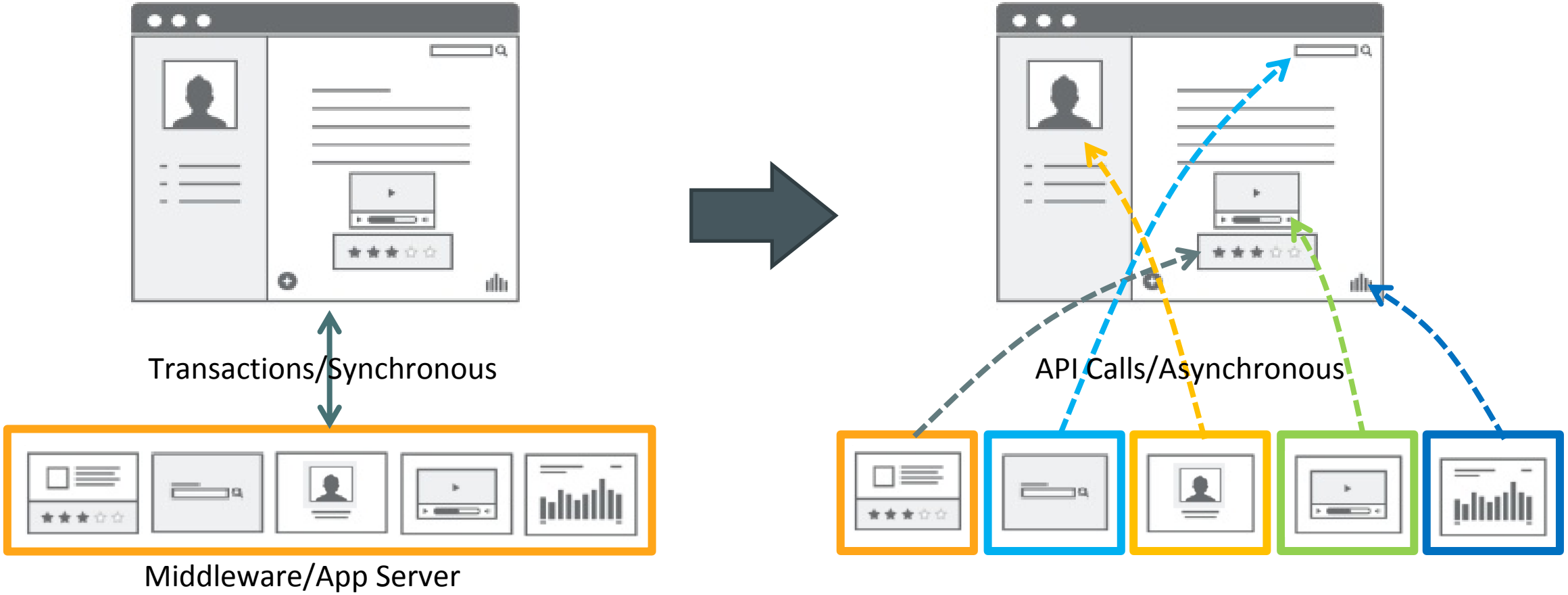
Declarative

Integrates to Existing Oracle Workloads

Application Container Cloud Service

Java SE, Node.JS, PHP

Microservices



Reference: http://www.slideshare.net/RichardHarvey7/micro-services-and-containers

Microservices란?

독립적으로 배포되는 최소화된 기능의 서비스지만 광범위한 사용사례를 위해 서로 상호작용하는 서비스

Monolithic Applications

- 단일, Monolithic App
- 전체 App을 함께 배포해야 함
- 전체 App이 단일 DB 사용
- 기술 계층 중심으로 구성
- 각각의 런타임 인스턴스에 상태를 보관
- 전체 App이 하나의 기술 스택을 채택
- SOAP 사용

Microservices

- 다수의, 더 작아진 최소 기능의 Microservices
- 각 Microservice를 독립적으로 배포
- 각 Microservice는 각자의 데이터 저장소를 가짐
- 비즈니스 기능 중심으로 구성
- 상태는 외부에 보관
- 각 Microservice가 각자의 기술을 선택
- HTTP를 통한 REST, Messaging, 또는 Binary

You're Doing Microservices If You...

Can build a microservice independently

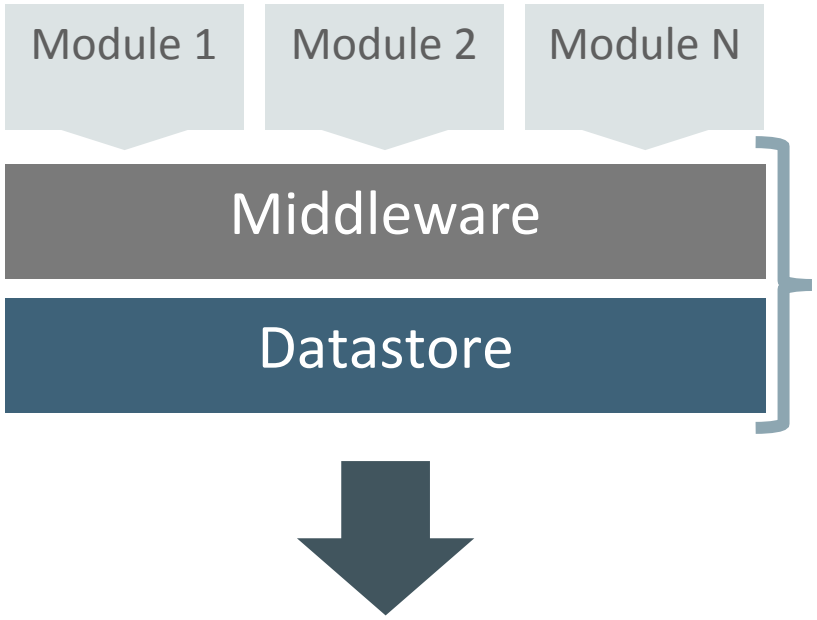
Can release each microservice independently

Don't share a datasource across microservices

“Micro” in Microservices != Runtime Weight

애플리케이션을 많은 작은 조각으로 나누어 마켓에 필요한 기능을 신속하게 획득

Monoliths

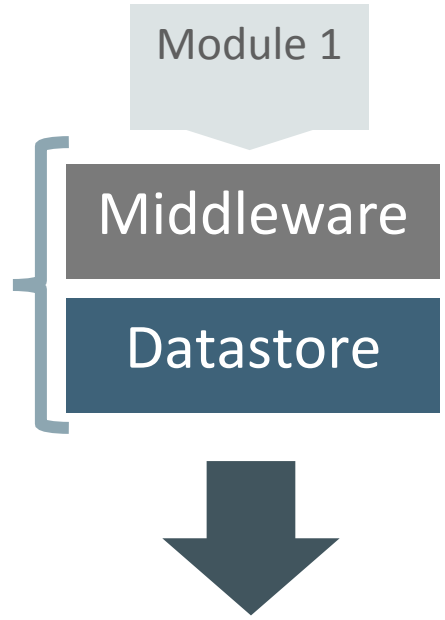


모든 모듈의 요구사항을 전부 지원해야 함

모든 케이스를 다 지원하는
완벽한 기능의 런타임

모듈 한 개의 요구사항을 지원하면 됨

Microservices



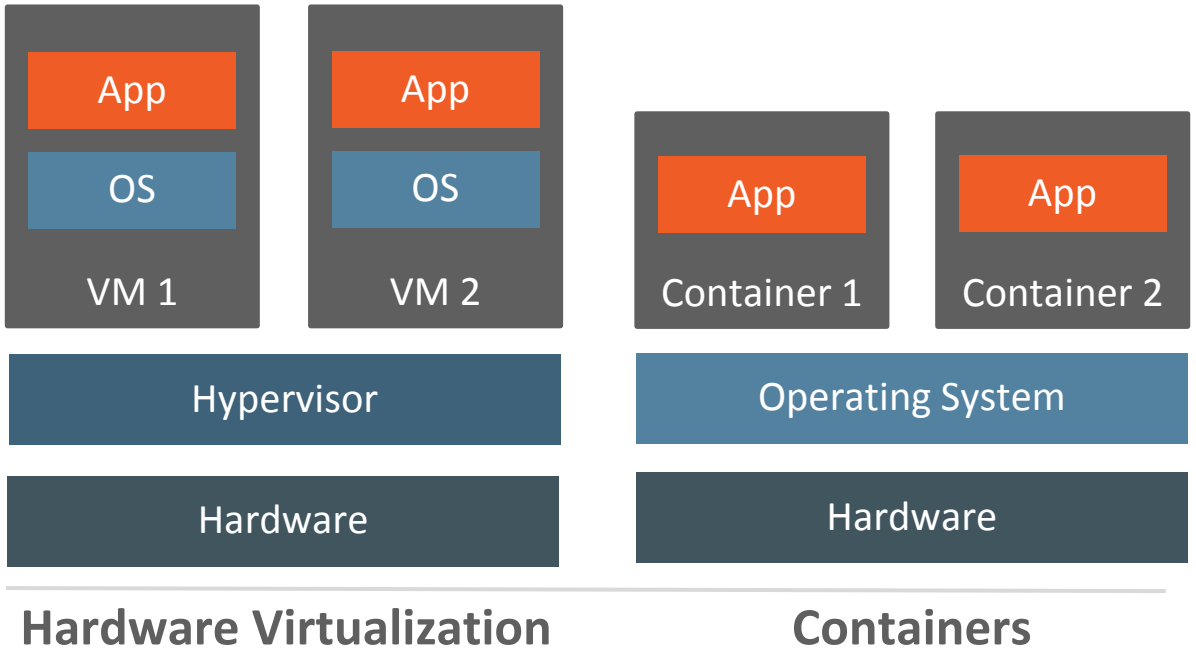
특정 기능을 수행하는
가벼운 런타임

Lightweight **Container**

스크립트(Code) 만으로 구동 – everything is 100% Code.

매우 빠르게 구동 – try for under 10 seconds.

깔끔하게 중지 – ephemeral

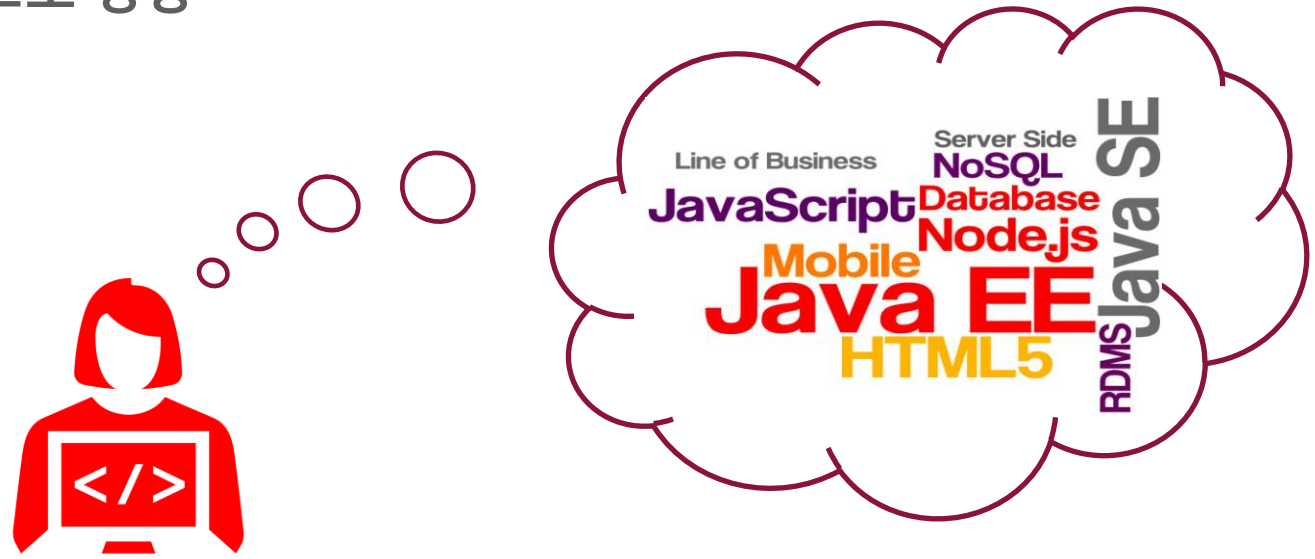


Higher density
Easy to start/stop
Portability

docker

개발 언어 : Polyglot

Polyglot 환경은 계속적으로 성장



Apache play jetty:// HIBERNATE eclipse Jersey

eclipselink vaadin JRuby Jython

Oracle Application Container Cloud

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

Application Container Cloud

클라우드 상에서 non-Java EE 업무를 운영하기 위한 플랫폼

- lightweight microservices를 위한 기반
  - Polyglot: Java SE, Node.js, JRuby, Groovy, Scala, Ruby*, Python*, PHP*, Web Profile*
- 사용이 매우 쉬움
  - 간소화된 최소한의 UI & REST API
- Build, Zip, Deploy
  - 인스턴스 수, 인스턴스 당 메모리 선택
  - 서비스를 위한 Docker 컨테이너를 자동으로 생성
- 백업이 필요 없음— 애플리케이션은 비상태 유지형
- 자동적인 로드 발란싱

Application Container Cloud Service

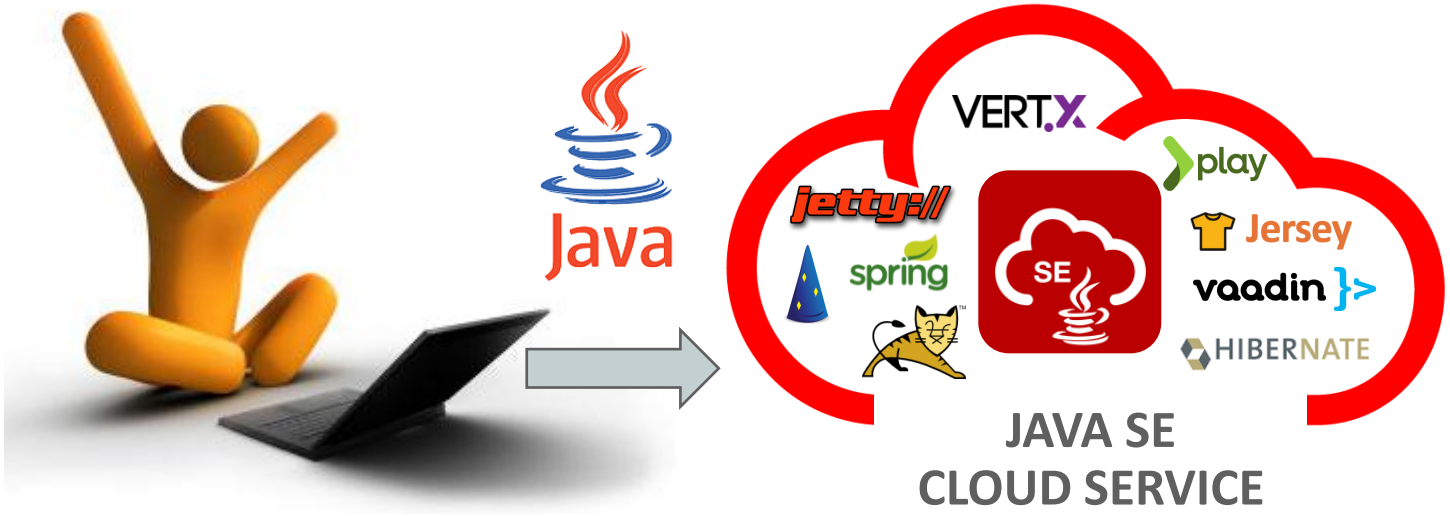
non-Java EE 실행을 위한 클라우드 플랫폼

- 가벼운 서비스를 위한 플랫폼
  - 다양한 언어: Java SE, Node.js, JRuby, Groovy, Scala, Ruby*, Python*, PHP*, Web Profile*
- Oracle Cloud Platform과 통합
  - Database CS
  - Java CS
  - Caching CS
  - Messaging CS
  - Storage CS
  - …
- Developer Cloud Service 를 통한 Continuous Delivery

* Roadmap Item

Application Container Cloud Service - Java SE

가벼운 마이크로서비스 아키텍처



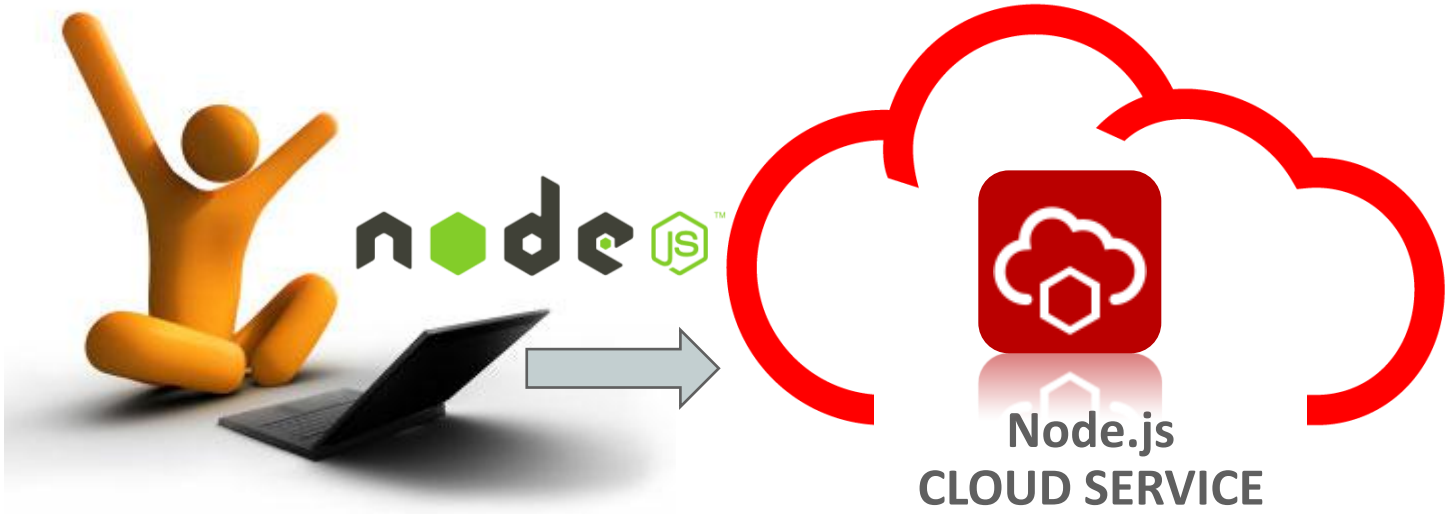
Java Support와 업그레이드를 제공하는 유일한 Cloud 벤더

- Java 7/8 - 어떤 오픈소스나 상용 Java framework* 을 실행 가능
- Java SE Advanced 상용 기능:
  - Flight Recorder, Mission Control
- Oracle Cloud Platform과 통합:
  - Database CS
  - Java CS
  - Caching CS
  - Messaging CS
  - Storage CS
- Oracle Traffic Director 로 부하분산

*JRuby, Jython, Scala 또는 다른 Java기반 언어 포함

Application Container Cloud Service - Node.js

가벼운 마이크로서비스 아키텍처



- Node.js 최신 안정화 버전 제공 – 유명 Node 라이브러리 포함
- 오라클 Node.js 확장 모듈
  - 예: Node.js Oracle Database Driver
- Oracle Cloud Platform과 통합 :
  - Database CS
  - Messaging CS
  - Java CS
  - Storage CS
  - Caching CS
- Oracle Traffic Director 로 부하분산

Application Container Cloud Service

장점 과 기능

빠른 Provision

셀프서비스, Zero Commands
버전 선택, size & capacity

간단한 운영

업그레이드 

Java Flight Recorder 

크기조정 

DR 장애복구 

부하에 따른 빠른 대응

Scale out/in

Scale up/down

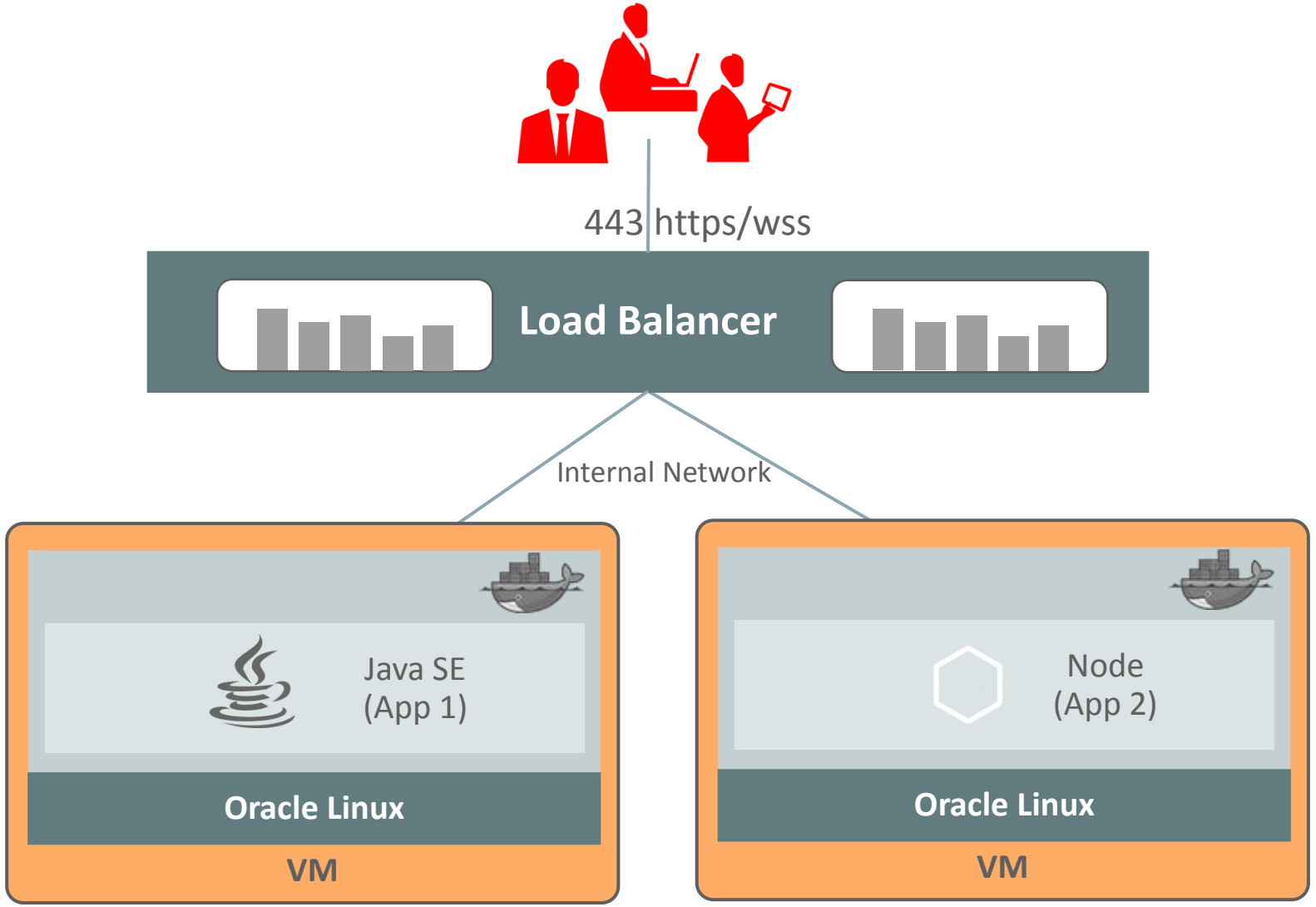
신속,연속 배포

Developer Cloud Service 를 통한
Continuous Delivery

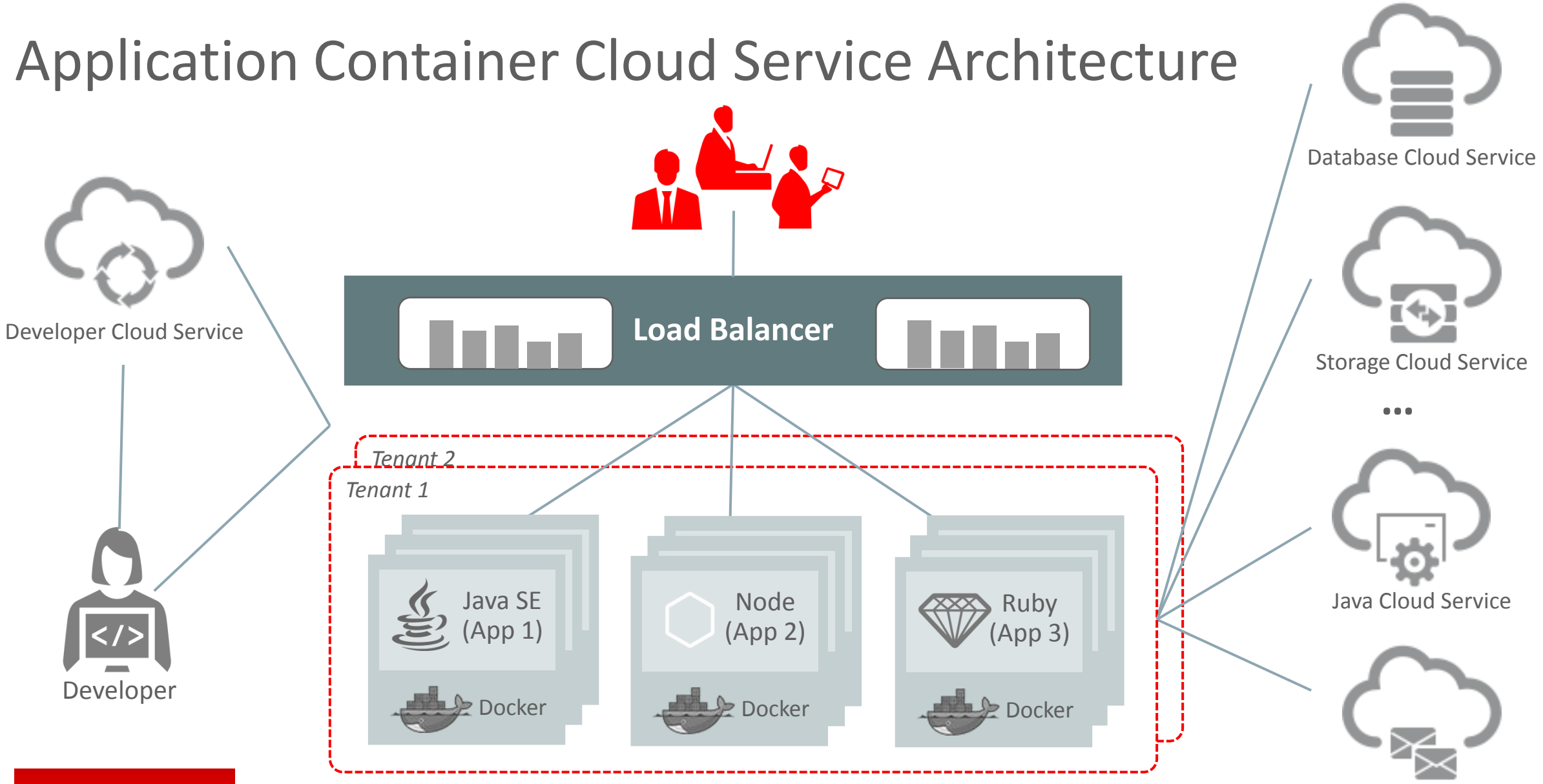
Application **Container** Cloud : Current Release

Language Support	Platform Features	Integrations
Java SE	12 Factor	DBCS
Node.js	Load Balancing	JCS
PHP	Elastic Scaling	Developer CS
	Service Binding	MySQL CS
	PSM CLI	
	Scale 32 x 20GB	

Application Container Cloud Architecture



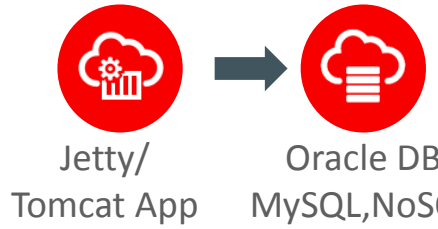
Application Container Cloud Service Architecture



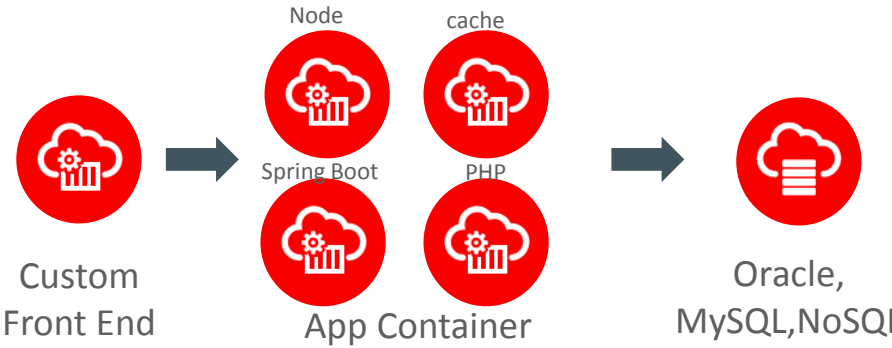
Typical AppDev Use Cases

Focus: Developing New Lightweight Applications

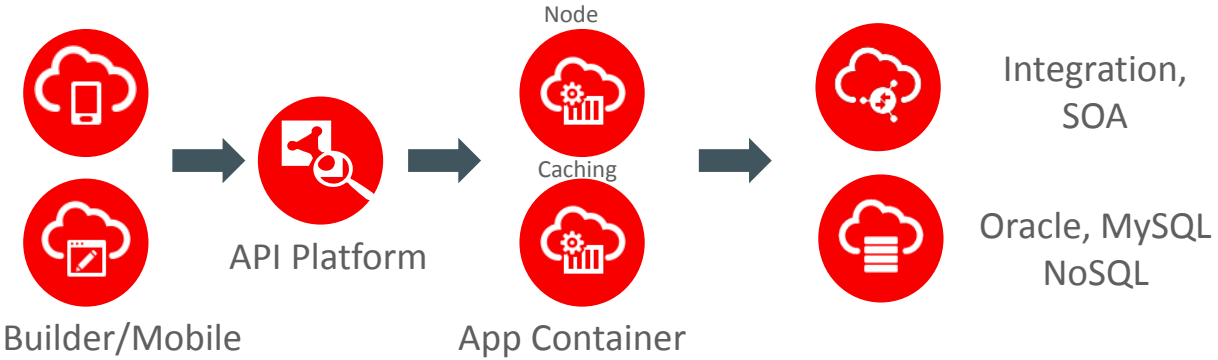
1. Simple lightweight apps running on To mcat, Jetty, Wildfly



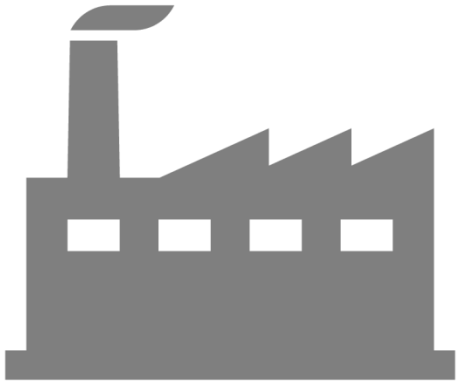
2. Composite Polyglot Apps – Basic

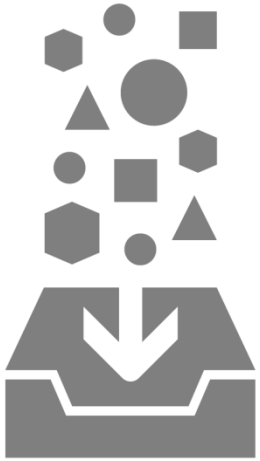


3. Composite Polyglot Apps – Advanced

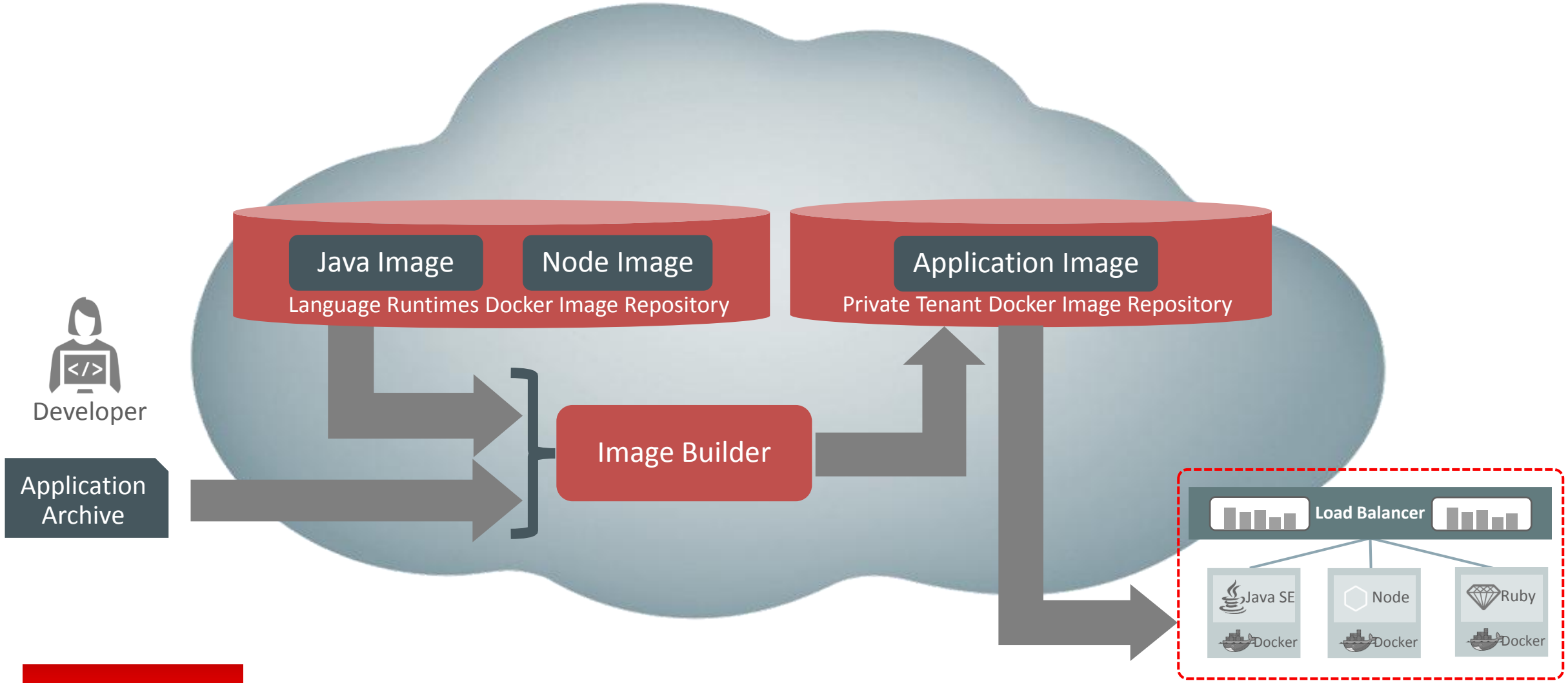


Build Zip Deploy!



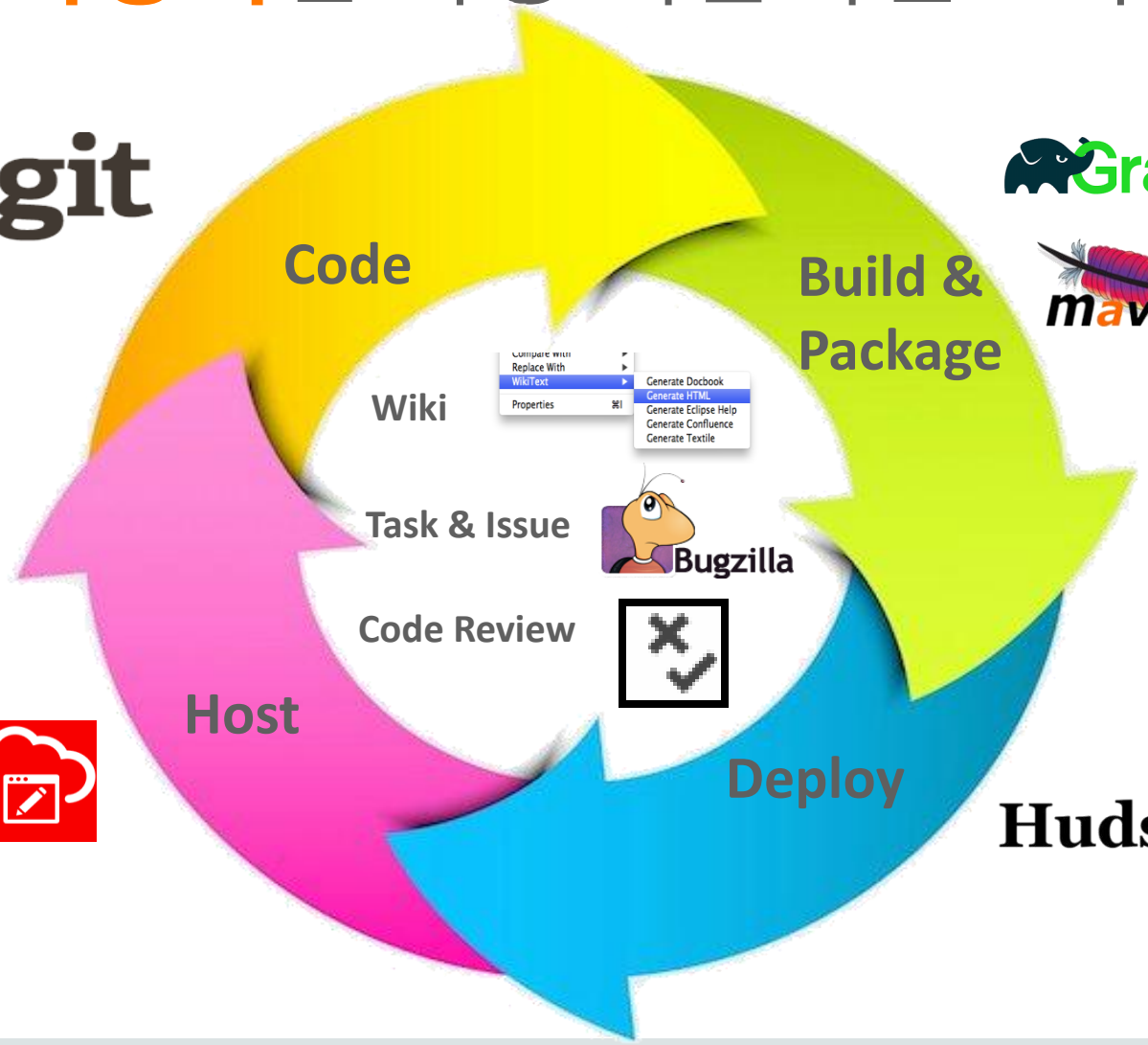


Application Deployment

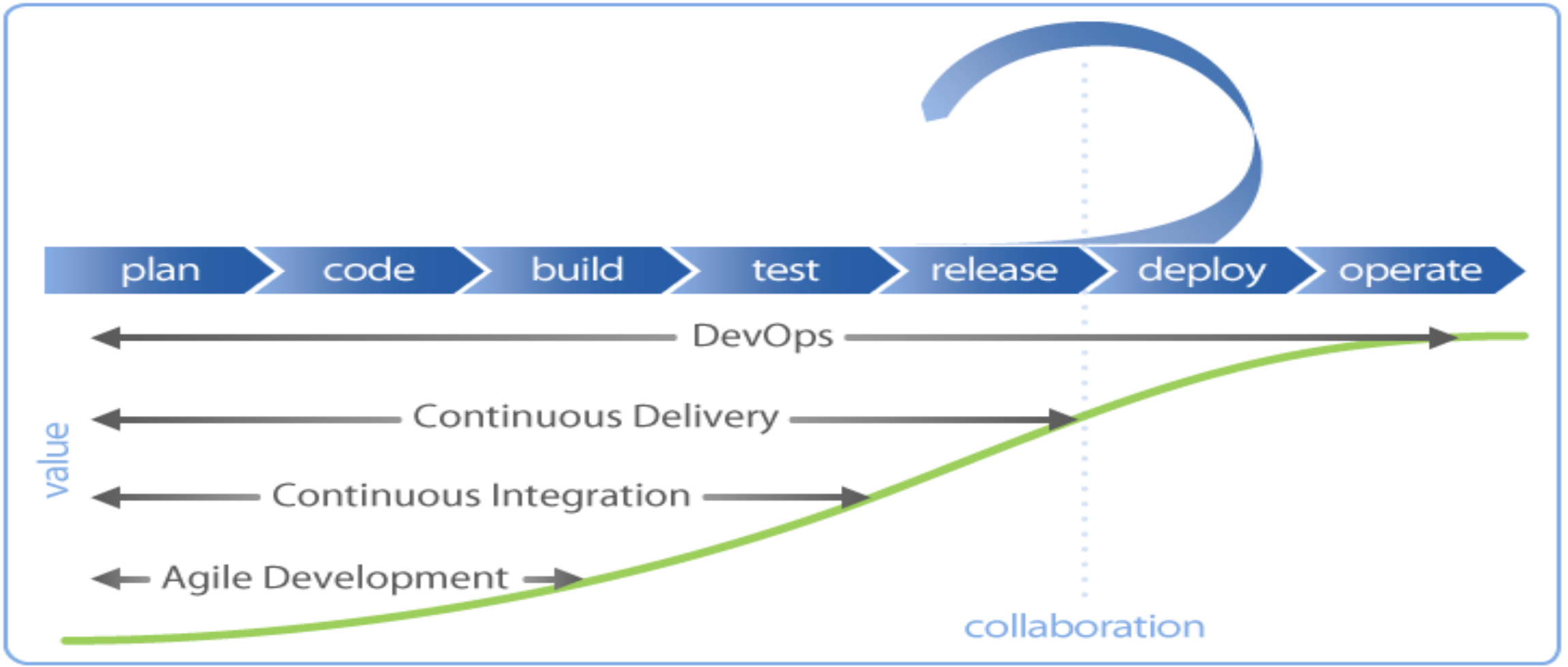


Developer Cloud Service

Automation - 자동화는 가장 기본적인 요구사항

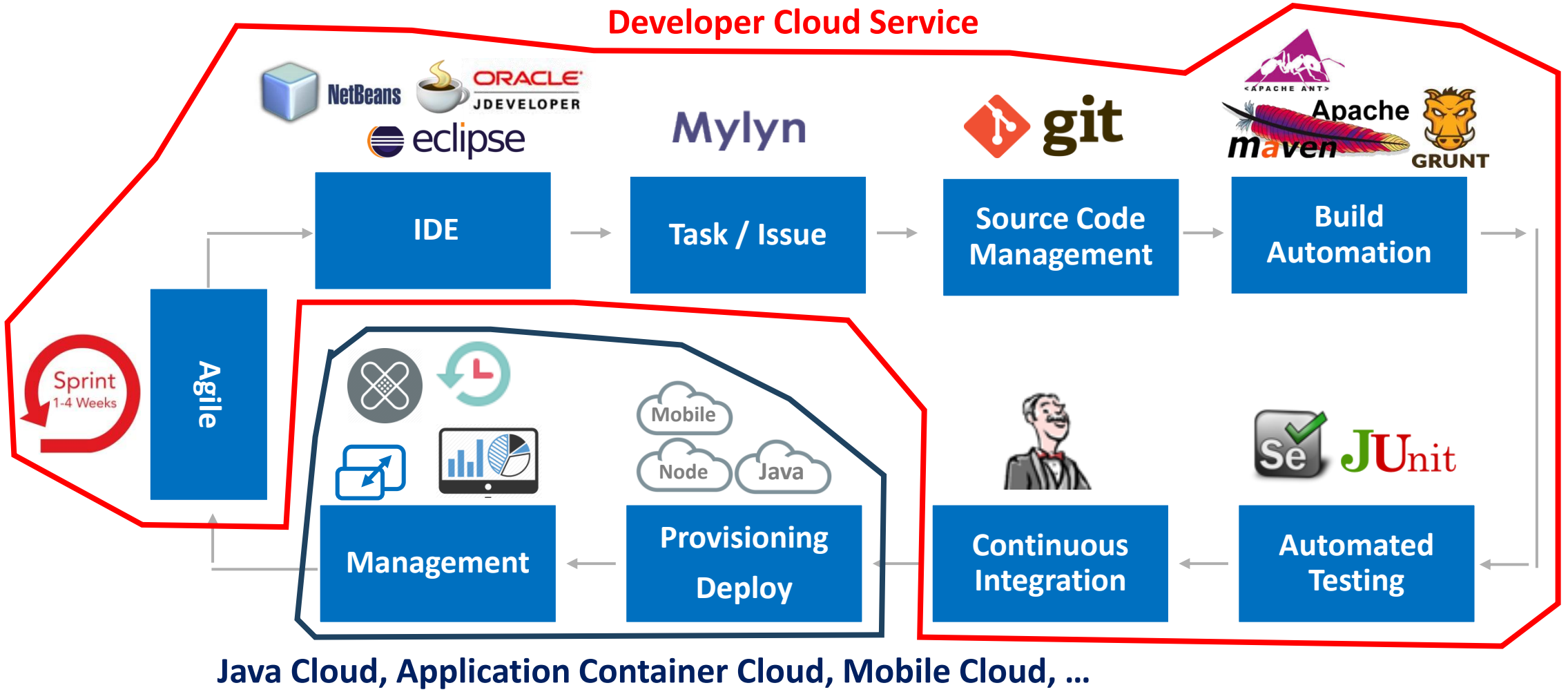


CI, CD, DevOps

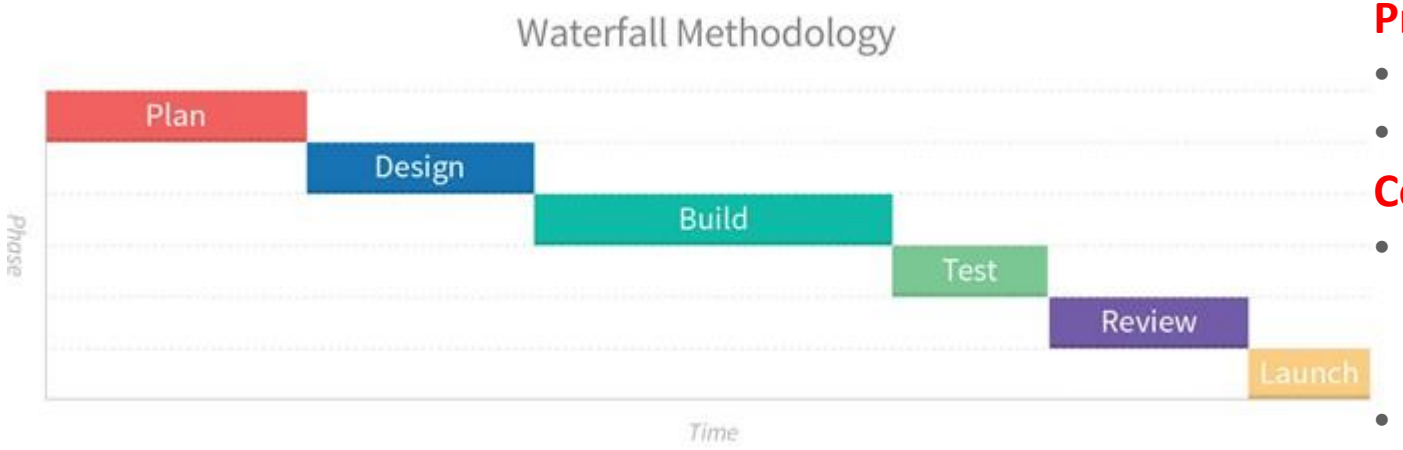


http://www.collab.net/solutions/devops

애플리케이션 라이프사이클 **자동화**



Project Management - **Agile** Methodologies



Pros
- 계획과 설계가 매우 직관적임
- 진척 상황 측정이 쉬움

Cons
- 고객이 요구사항을 초반부에 잘 형상화 하지 못하는 경우, 고객이 제품 품질에 만족하지 못할 수 있음
- 변경이 어렵거나 많은 비용이 필요할 수 있음

Pros
- 진행되는 상황을 자주 확인할 수 있고, 변경이 용이
- Time to Market이 중요한 애플리케이션의 경우 초기 버전으로 출시가 가능

Cons
- 고객의 참여가 많이 요구됨
- Time box 기반으로 우선순위 재조정이 계속 일어남

Source : http://www.screenmedia.co.uk/blog/2014/08/what-is-agile-development-a-brief-introduction/

Developer Cloud: For Continuous Delivery

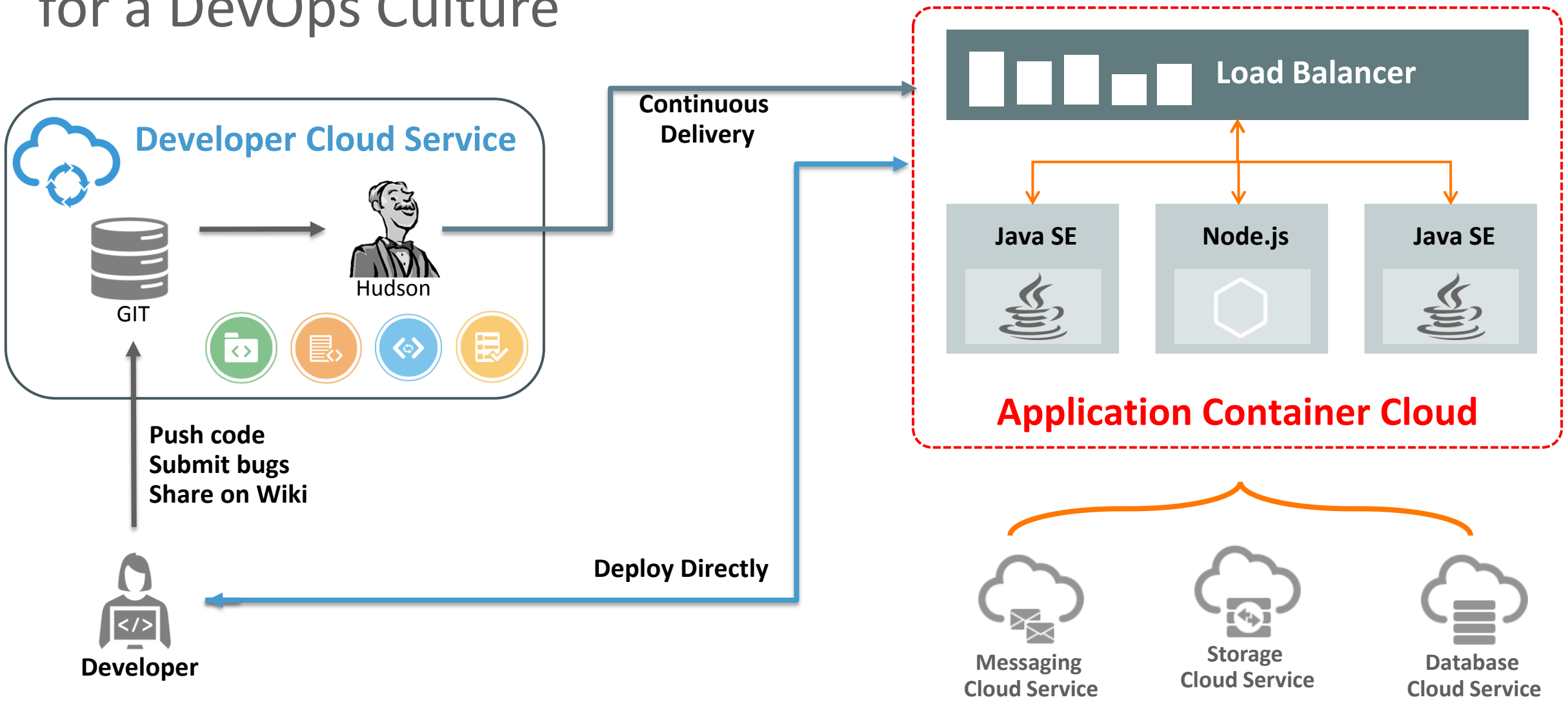
- Java Cloud, Application Container Cloud, Mobile Cloud와 함께라면 무료
- 특징:
  - 프로젝트 기반, 멀티 테넌트
  - wiki server 통합
  - task/defect service 통합
  - 코드 리뷰
  - Git 저장소
  - Maven & Ant 통합
  - Hudson Continuous 통합
  - IDE 통합 (Eclipse, NetBeans, Jdeveloper)

Developer Cloud Service – 쉬운 적용/통합

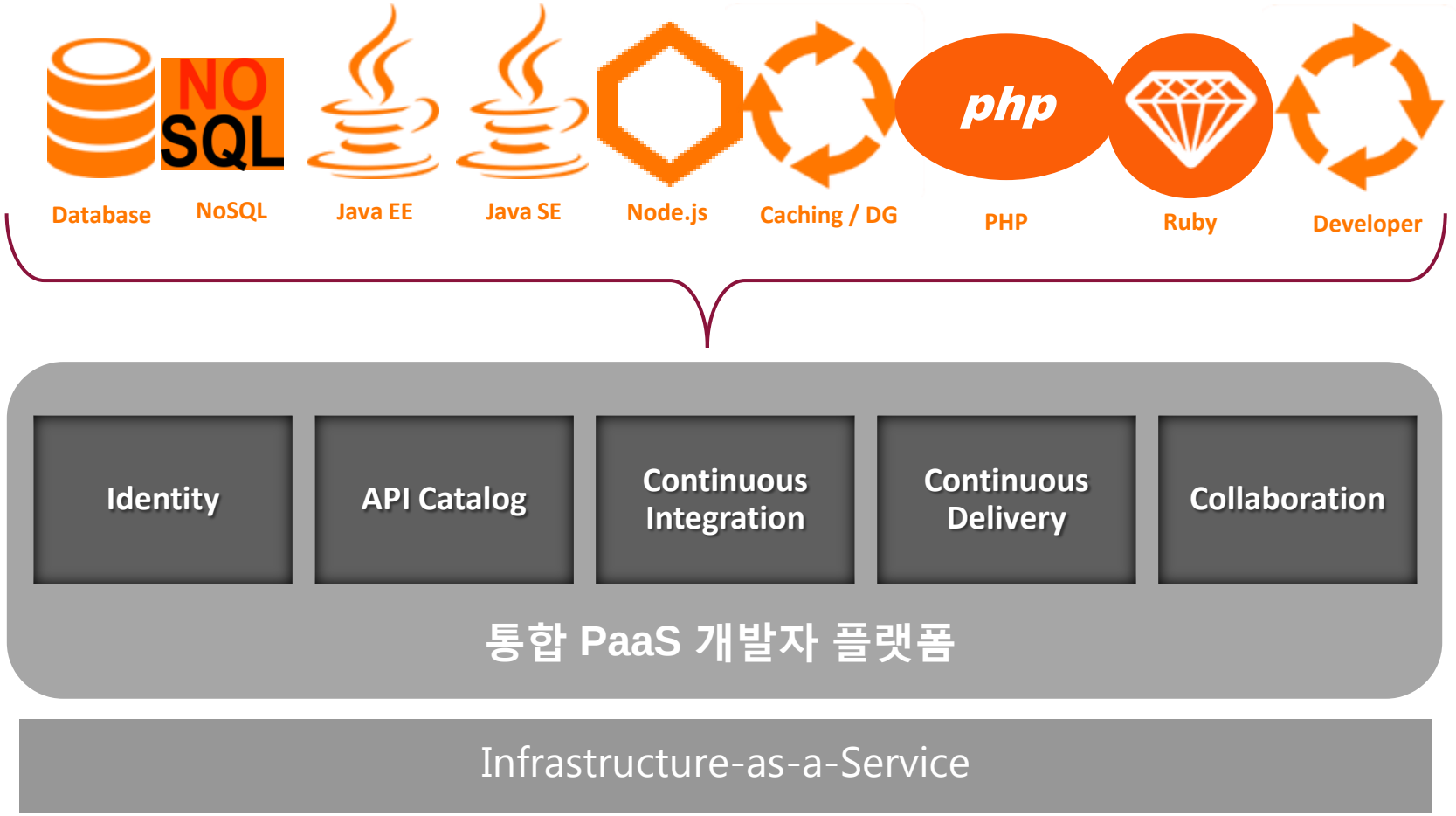
클라우드 통합 개발 기술

- 표준 기반
  - Git, Maven, Hudson, Ant, etc.
- IDE 연동기능 내장
  - Eclipse, NetBeans, JDeveloper
- 유연한 소스 위치
  - Hosted Git 또는 GitHub 제공
- 배포 Target의 선택
  - Oracle Cloud 또는 on-premise

Integrated Oracle Cloud Services for a DevOps Culture



Application Container Cloud Service



Integrated Cloud

Applications & Platform Services


ORACLE®