ORACLE

Small is not always beautiful – moving enterprise applications to the Cloud

—

Paul Jenkins

Product Manager @ Oracle Cloud

Tony Vertenten

CTO @ Intris NV

Paul Jenkins

Senior Principal Product Manager
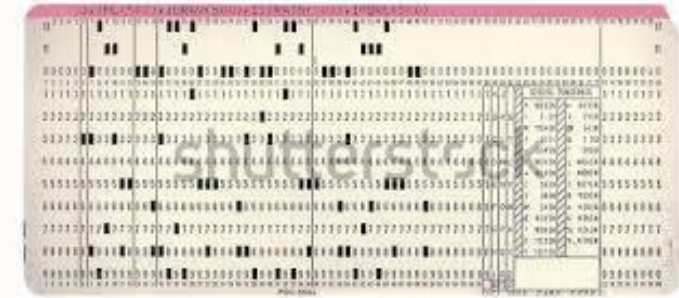Oracle Cloud Infrastructure
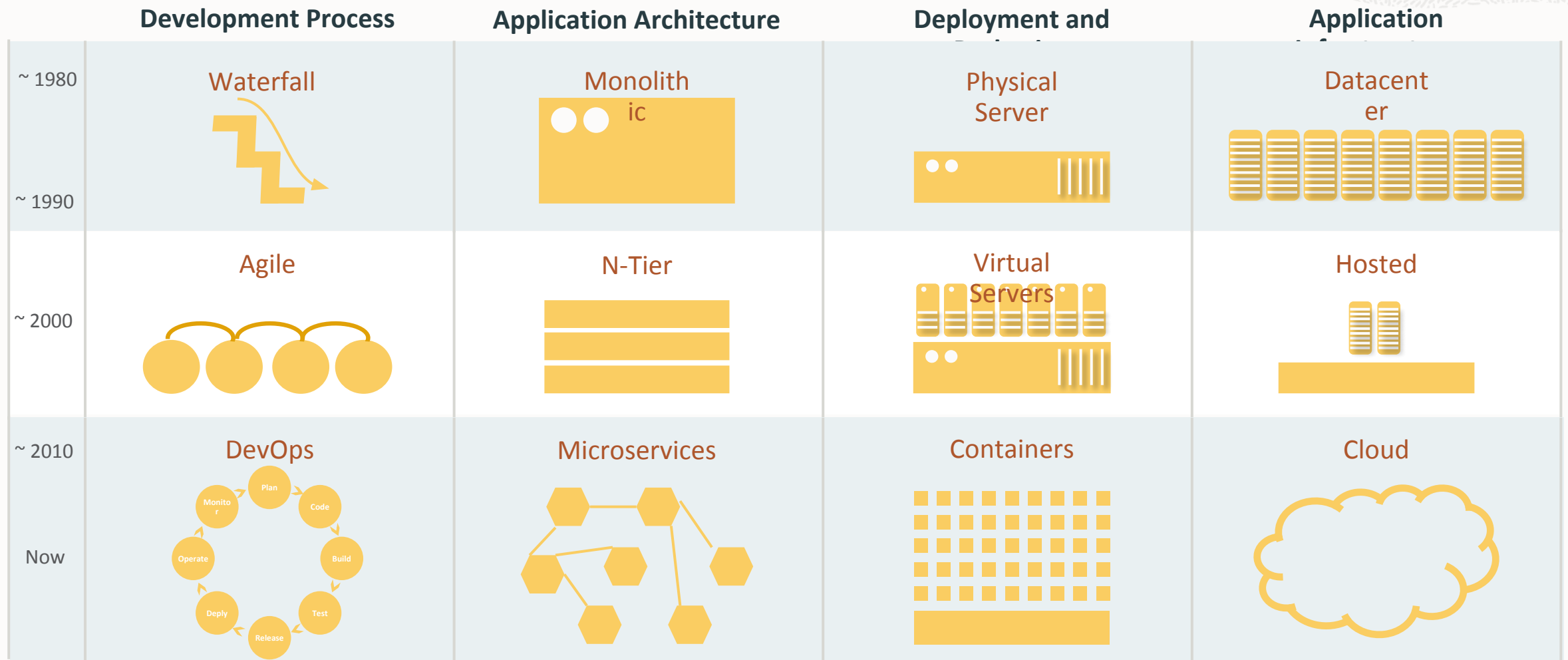
twitter.com/jenksho
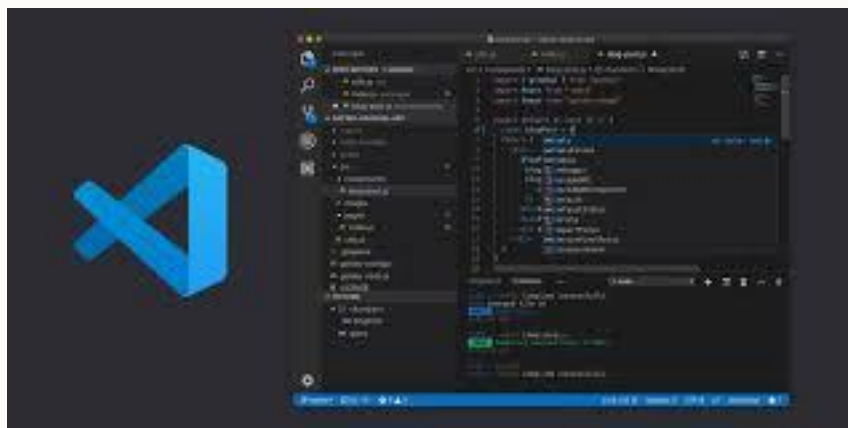
linkedin.com/in/jenksho

Tony Vertenten

Co-Founder and CTO
Intris

Evolution of Development & Deployment

	Development Process	**Application Architecture**	**Deployment and**	**Application**
~ 1980 / ~ 1990	Waterfall	Monolithic	Physical Server	Datacenter
~ 2000	Agile	N-Tier	Virtual Servers	Hosted
~ 2010 / Now	DevOps	Microservices	Containers	Cloud

The **Desires**

"We want to go serverless !"

"How do we do DevOps ?"

"What about micro-services ?"

"How can we reduce our delivery time ?"
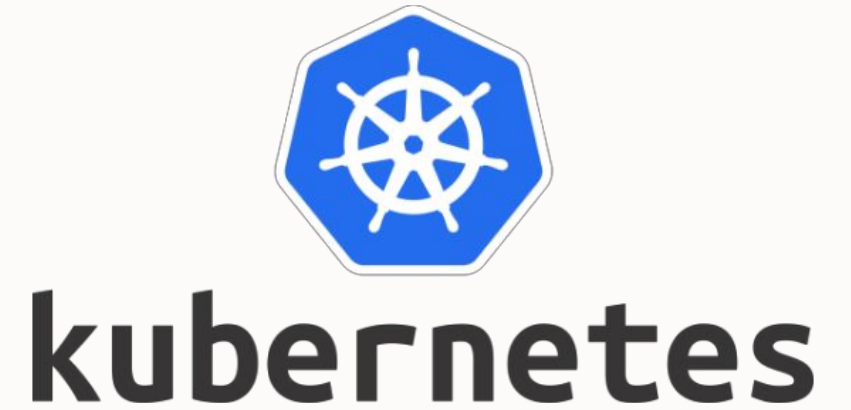
The **realities**

Complexity of existing systems – technical debt

Pace layering – not all systems need constant change

Skills – enterprise application servers

Organisation and culture - silos

A Journey

Copyright © 2020, Oracle and/or its affiliates

Transition

- Customer, industry trends to cloud native technology

- Embrace Kubernetes as the preferred deployment platform
    - Support containerized WebLogic and Coherence today
    - Simplify deployment of applications on Kubernetes

- Provide microservices evolution path
    - Helidon and other microservices technologies
    - Migrate traditional application to microservices

Kubernetes on Public Cloud

Migrate

Physical & VMs On Premise

Kubernetes on Premises
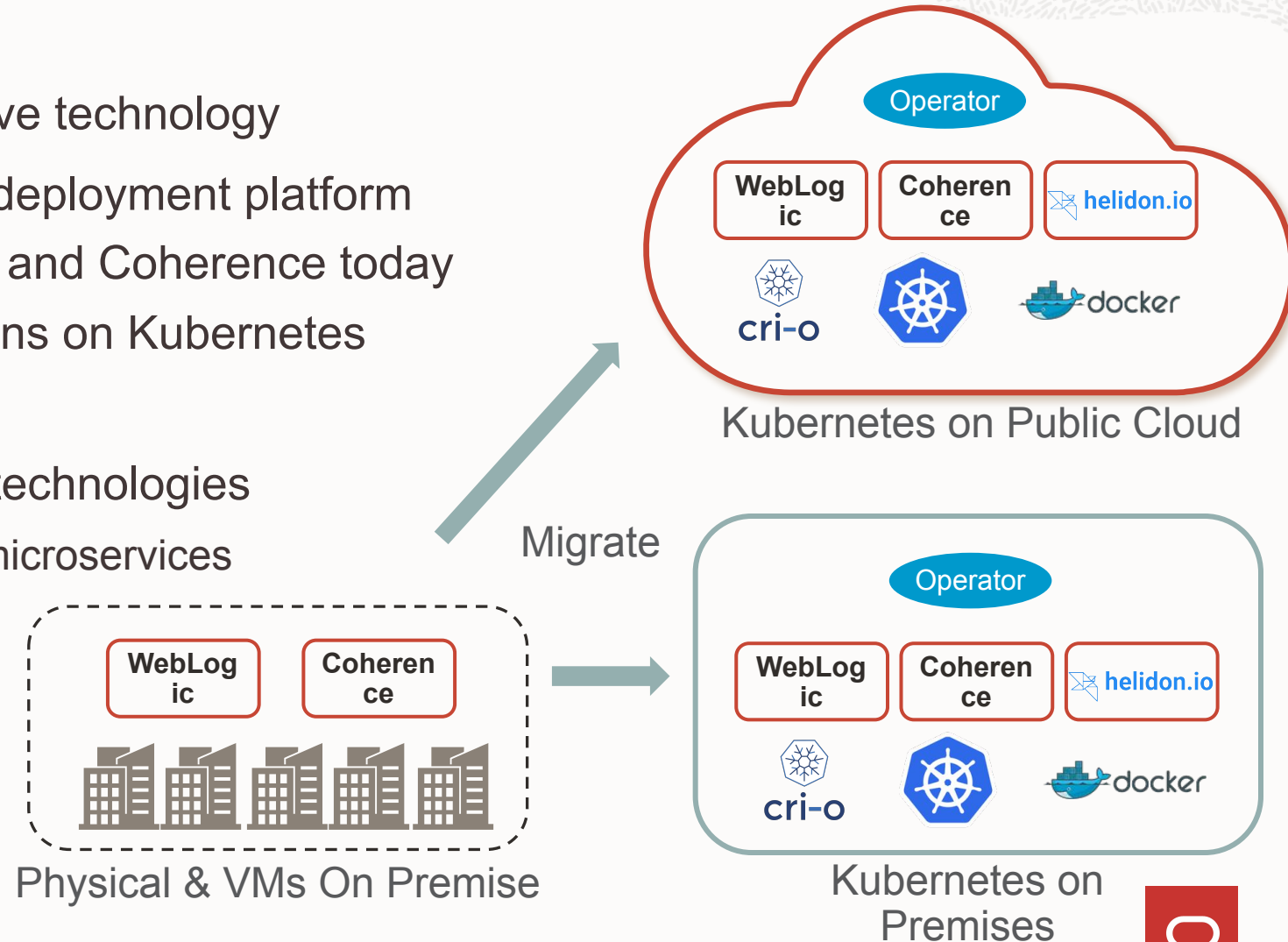
Kubernetes Operator

- Makes use of existing skill sets
- Focus on applications rather than Kubernetes
- Contains built-in knowledge about how to perform lifecycle operations on a domain
- Uses Kubernetes APIs to automate lifecycle operations.
- Transition to modern tool-chains and methods

Copyright © 2020, Oracle and/or its affiliates
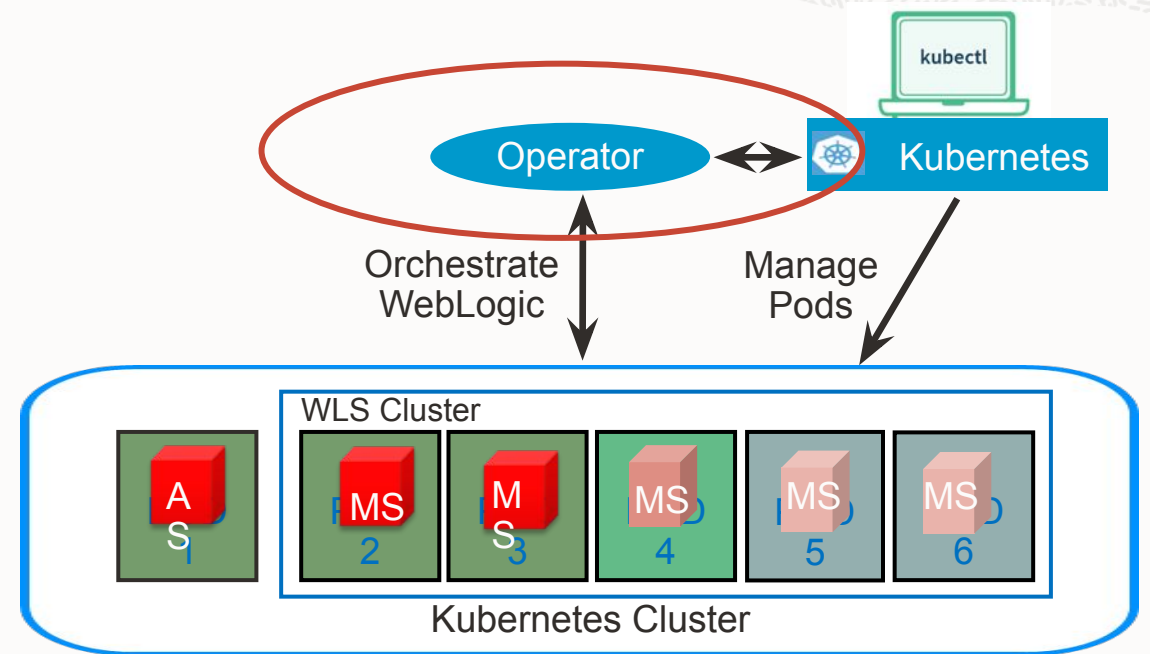
WebLogic Kubernetes Operator

1 Manages lifecycle operations (start, stop, scale, rolling restart, etc.) in Kubernetes

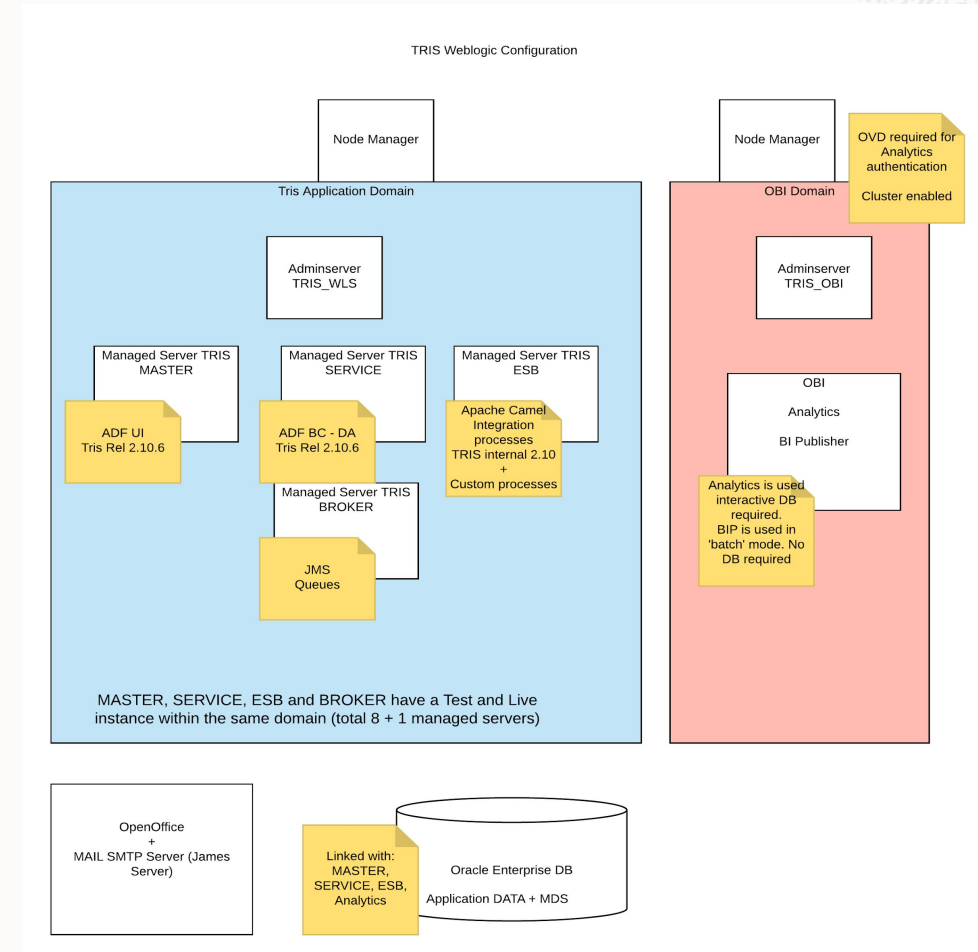**2** Automate configuration, e.g. clustering, channels/ports, configuration overrides

3 Supports standard k8s idioms like sidecars, init containers, custom resources

Open source and fully supported
https://github.com/oracle/weblogic-kubernetes-operator

Intris

- Independent Software Vendor
  - Java EE application for logistics service providers
  - 300+ customers. Smallest 2 users largest 450 users
  - Embedded Software License => separate installation for each customer
  - New release every 6 weeks
  - Development team
    - 12 application developers
    - 4 integration

TRIS Weblogic Configuration

Node Manager

Node Manager

OVD required for Analytics authentication

Cluster enabled

Tris Application Domain

OBI Domain

Adminserver TRIS_WLS

Adminserver TRIS_OBI

Managed Server TRIS MASTER

Managed Server TRIS SERVICE

Managed Server TRIS ESB

OBI

Analytics

BI Publisher

ADF UI
Tris Rel 2.10.6

ADF BC - DA
Tris Rel 2.10.6

Apache Camel Integration processes
TRIS internal 2.10
+
Custom processes

Analytics is used interactive DB required.
BIP is used in 'batch' mode. No DB required

Managed Server TRIS BROKER

JMS Queues

MASTER, SERVICE, ESB and BROKER have a Test and Live instance within the same domain (total 8 + 1 managed servers)

OpenOffice
+
MAIL SMTP Server (James Server)

Linked with: MASTER, SERVICE, ESB, Analytics

Oracle Enterprise DB

Application DATA + MDS

Challenges
Mix of environments to deploy on

Problem: Customers all have different environments
- on premise and in the cloud
- Performance varies certainly for on premise infrastructure
- Deployment can not be fully automated

Solution: Provide standard cloud environment
- Uniform environment (linux, cpu, memory, …)
  - Standardized deployment scripts
  - Predictable performance
- 1 compute instance for small customers, 3 compute instances for larger customers
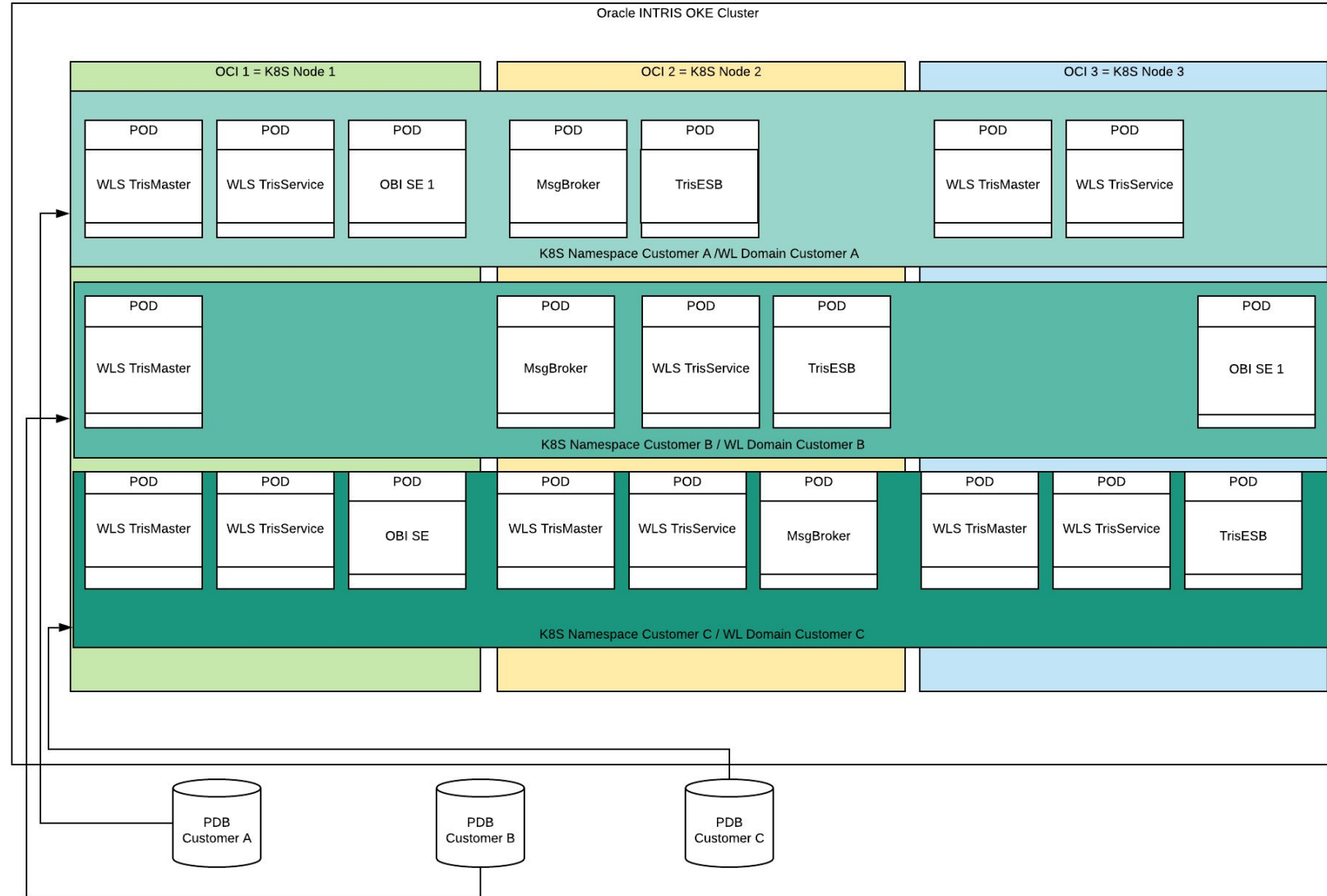
Challenges
Deployment of new release to every customer

—

Problem

- New release every 6 weeks but deployment to customers takes 8 weeks

Solution : Kubernetes

- Automated deployment connected to CI/CD pipeline
- Blue and green scenario
- Better use of (compute) resources
- Scalable
- Available in the cloud and on premise

Challenges

Docker and Kubernetes have their own challenges

Problem

- Docker and Kubernetes require a new set of skills and knowledge
- No experience with Ops, focus on application development
- Monitor and manage

Solution : Use existing tools

- WebLogic Kubernetes Operator
- WebLogic Deployment Tool
- WebLogic Image Tool
- Elasticsearch, Fluentbit, Kibana (EFK)
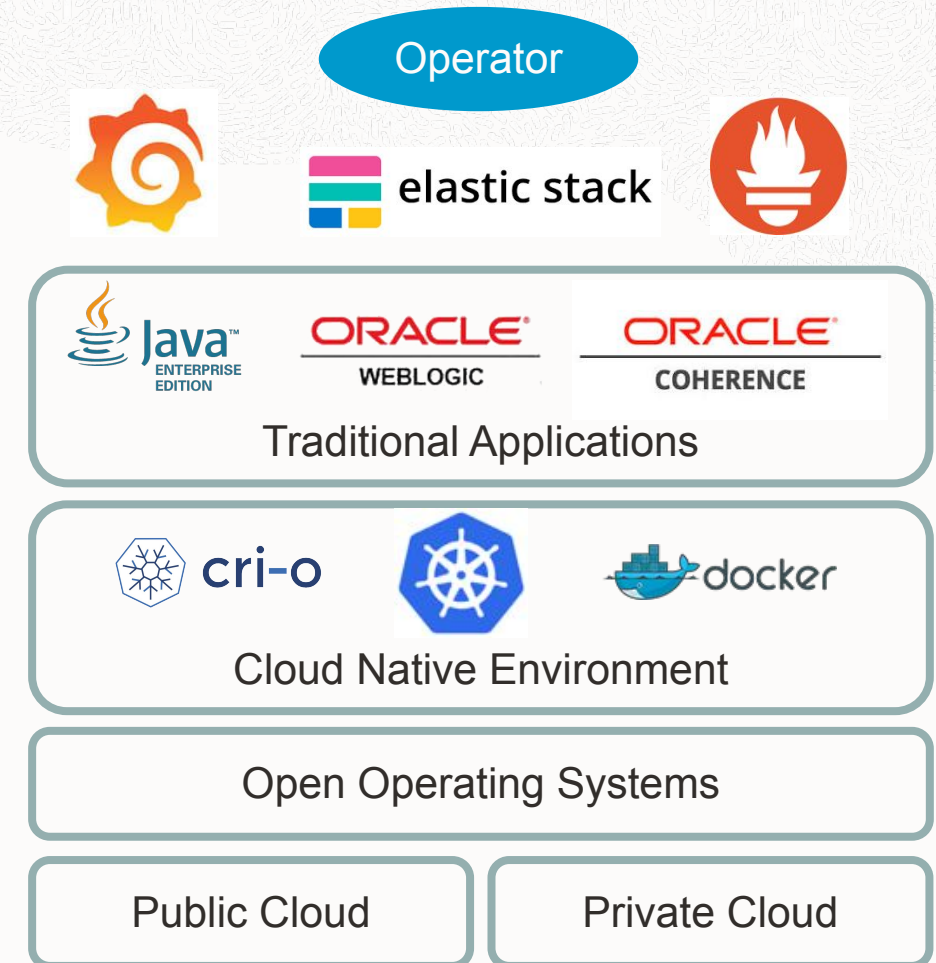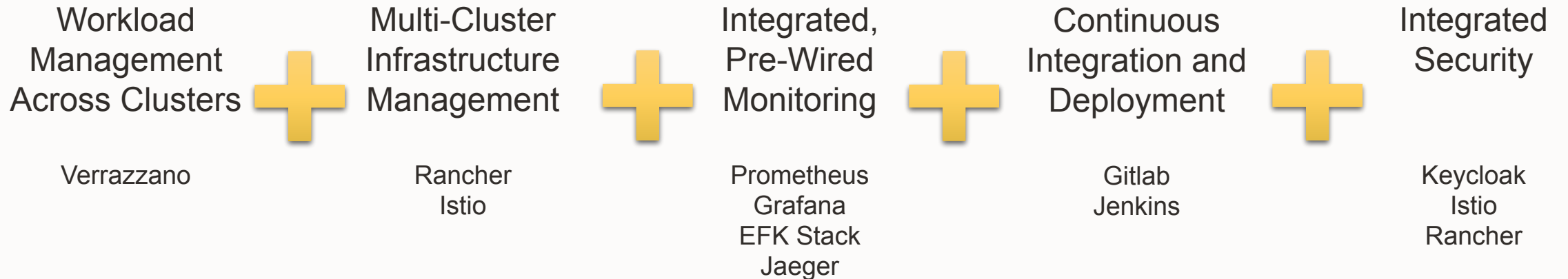- Prometheus
- Grafana

Next steps

- Microservices
- Helidon: https://helidon.io
- GraalVM: https://www.graalvm.org

Enterprise Needs

- Enterprise applications have complex interdependencies

- Applications need systems, resources, connectivity, access control and monitoring/management

- Enterprises want to run enterprise applications and microservices in a shared environment

Operator

elastic stack

Java ENTERPRISE EDITION | ORACLE WEBLOGIC | ORACLE COHERENCE

Traditional Applications

cri-o | docker

Cloud Native Environment

Open Operating Systems

Public Cloud | Private Cloud

Enterprise Container Tooling

Workload Management Across Clusters
+
Multi-Cluster Infrastructure Management
+
Integrated, Pre-Wired Monitoring
+
Continuous Integration and Deployment
+
Integrated Security

Verrazzano

Rancher
Istio

Prometheus
Grafana
EFK Stack
Jaeger

Gitlab
Jenkins

Keycloak
Istio
Rancher

Kubernetes	Kubernetes	Kubernetes
Public Cloud	Private Cloud	Multi-Cloud

More than Kubernetes

Traditional Applications

Java Microservices

Polyglot microservices

Cloud Native Environment

Operators

Open Operating Systems

Public Cloud

Private Cloud

Multi-Cloud

Thank you

Paul Jenkins

Product Manager @ Oracle Cloud

twitter.com/jenksho

linkedin.com/in/jenksho

Tony Vertenten

CTO @ Intris NV