SECR

Software Engineering Conference Russia

November 14-15, 2019. Saint-Petersburg

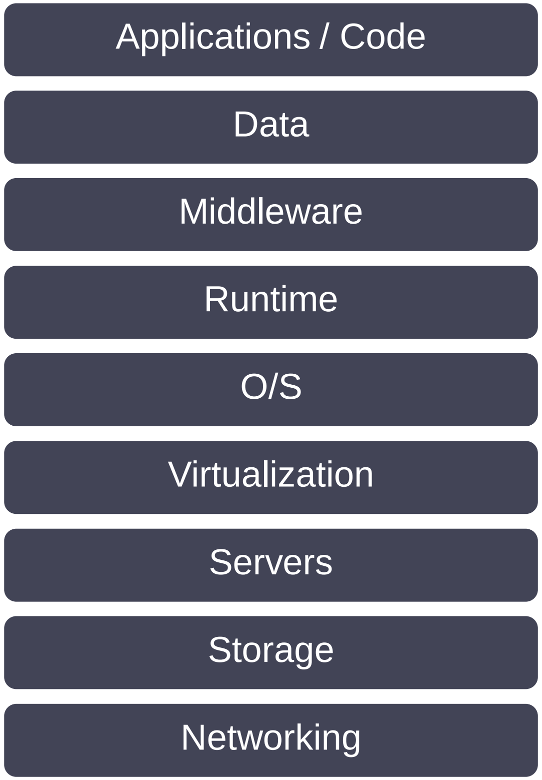
A brief history of (multi)Cloud

A container story

Franck Descollonges

IBM Cloud Developer Advocate - Europe

Back to basics : The IT Stack



Applications / Code

Data

Middleware

Runtime

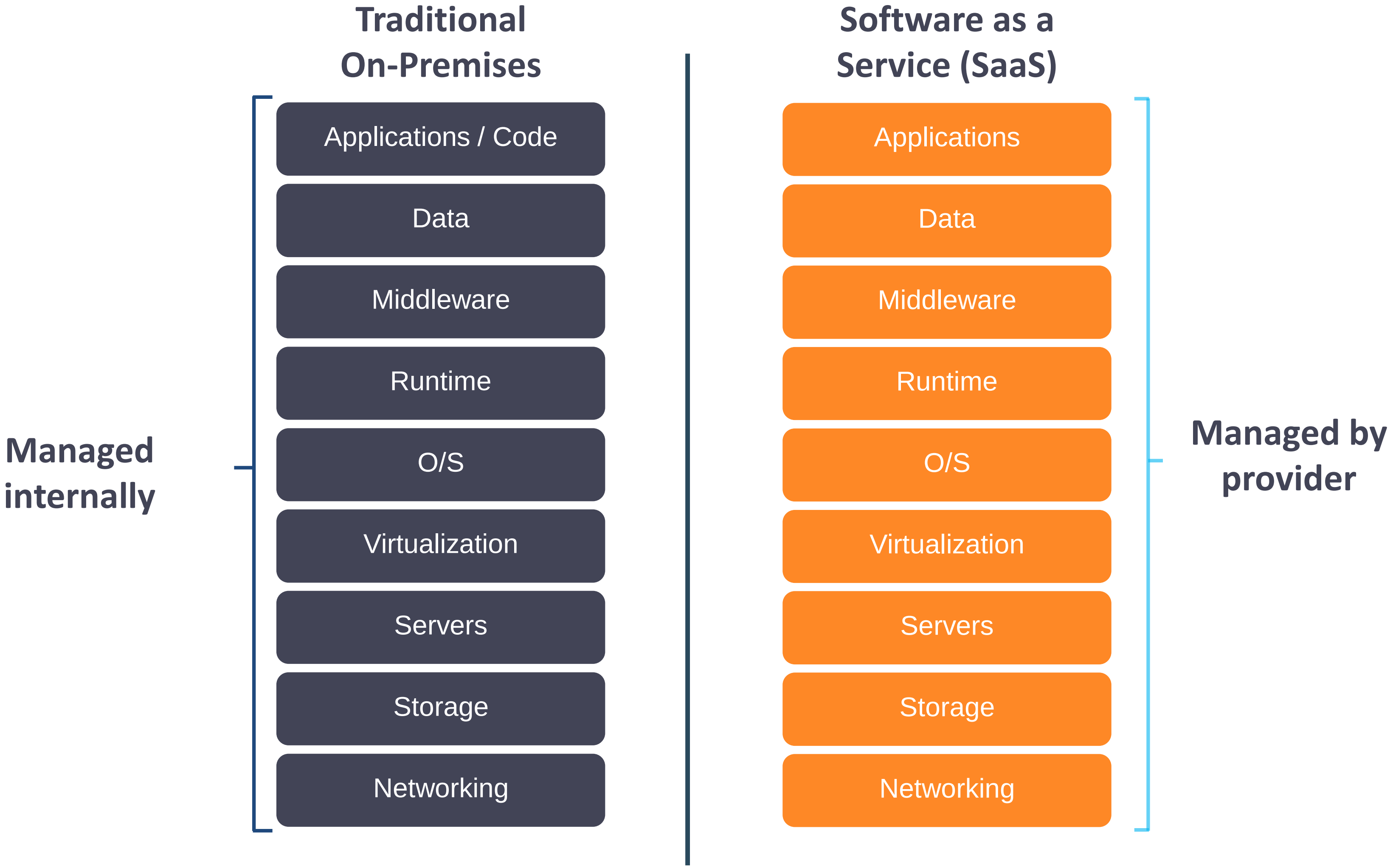
O/S

Virtualization

Servers

Storage

Networking



Traditional On-Premises

Applications / Code

Data

Middleware

Runtime

O/S

Virtualization

Servers

Storage

Networking

Managed internally

Software as a Service (SaaS)

Applications

Data

Middleware

Runtime

O/S

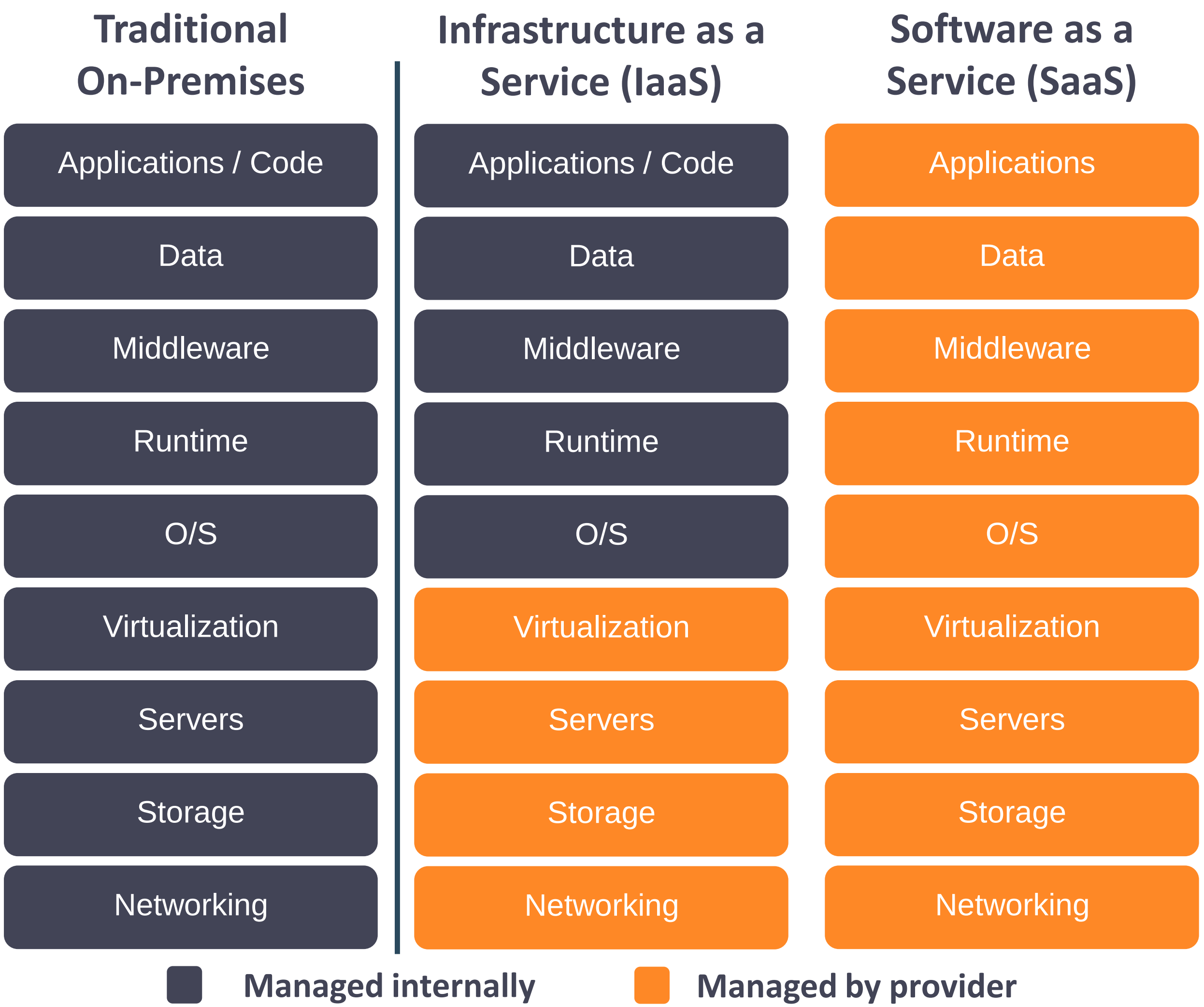
Virtualization

Servers

Storage

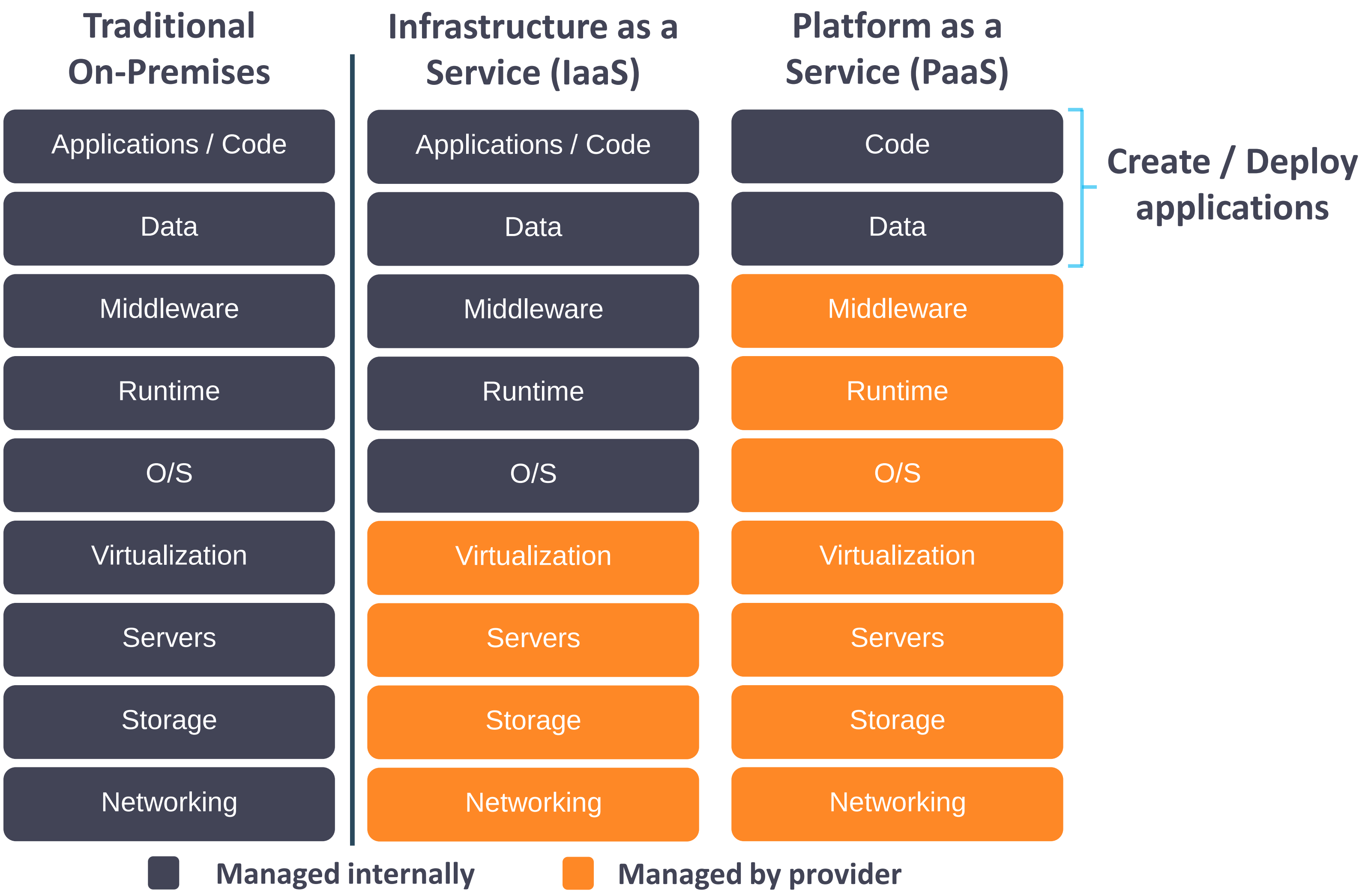
Networking

Managed by provider



Traditional On-Premises	Infrastructure as a Service (IaaS)	Software as a Service (SaaS)
⬛ Applications / Code	⬛ Applications / Code	🟧 Applications
⬛ Data	⬛ Data	🟧 Data
⬛ Middleware	⬛ Middleware	🟧 Middleware
⬛ Runtime	⬛ Runtime	🟧 Runtime
⬛ O/S	⬛ O/S	🟧 O/S
⬛ Virtualization	🟧 Virtualization	🟧 Virtualization
⬛ Servers	🟧 Servers	🟧 Servers
⬛ Storage	🟧 Storage	🟧 Storage
⬛ Networking	🟧 Networking	🟧 Networking

⬛ **Managed internally** 🟧 **Managed by provider**



Traditional On-Premises	Infrastructure as a Service (IaaS)	Platform as a Service (PaaS)
⬛ Applications / Code	⬛ Applications / Code	⬛ Code
⬛ Data	⬛ Data	⬛ Data
⬛ Middleware	⬛ Middleware	🟧 Middleware
⬛ Runtime	⬛ Runtime	🟧 Runtime
⬛ O/S	⬛ O/S	🟧 O/S
⬛ Virtualization	🟧 Virtualization	🟧 Virtualization
⬛ Servers	🟧 Servers	🟧 Servers
⬛ Storage	🟧 Storage	🟧 Storage
⬛ Networking	🟧 Networking	🟧 Networking

Create / Deploy applications

⬛ Managed internally 🟧 Managed by provider

IT is being transformed by Containers driven by agility and economics

Malcolm McLean - 1937
Intermodal Shipping Containers

↓

Standardized Building Block
90% Decrease in Costs

↓

Flexibility
Portability
Efficiency

↓

Revolutionized International Trade

Open Source (Apache) - 2013
OS-level Virtualization Containers

↓

Standardized Building Block
48% Decrease in Costs

↓

Flexibility
Portability
Efficiency

↓

Revolutionizing IT

Why it works? Separation of concerns

Deb - The Developer

Mike - The Ops Guy

Worries about what's « **inside** » the container

- Her code
- Her libraries
- Her Package Manager
- Her Apps
- Her Data

To her, all Linux servers look the same

Worries about what's « **outside** » the container

- Logging
- Remote access
- Monitoring
- Network config

All containers start, stop, copy, attach, migrate… the exact same way

Why so much interest in Containers?

#1 : Application Portability

Isolated containers package the application, dependencies and configurations together. These containers can then seamlessly move across environments and infrastructures.

#2 : Ship More Software

Accelerate development & deployment, CI and CD pipelines by eliminating headaches of setting up environments and dealing with differences between environments. On average, containers users ship software 7X more frequently.

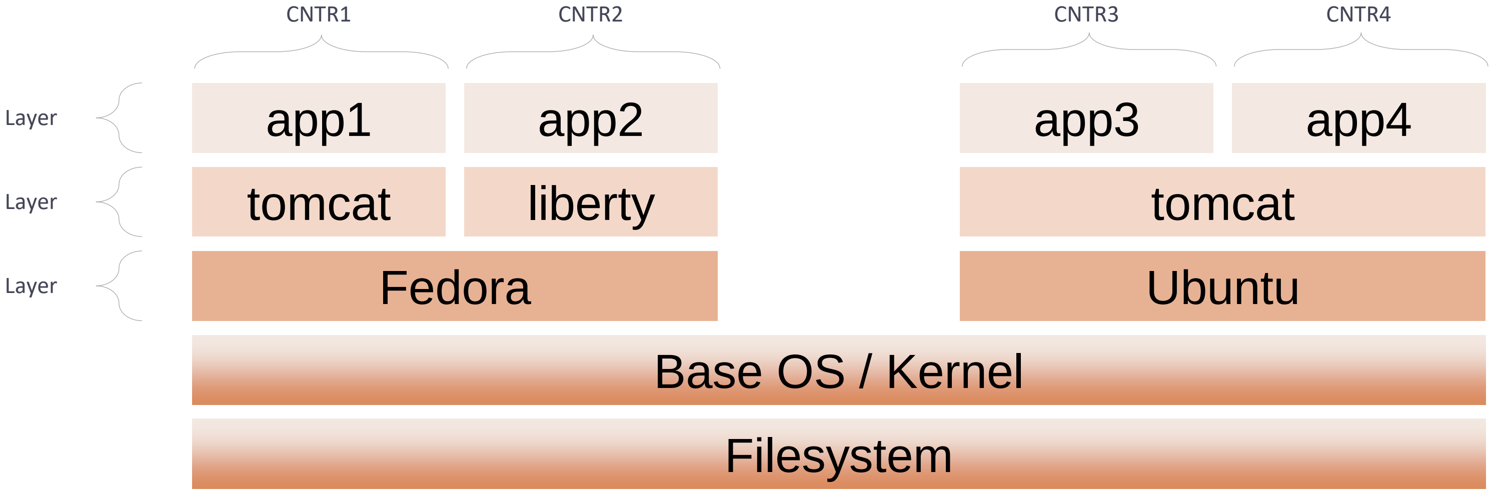
#3 : Resource Efficiency

Lightweight containers run on a single machine and share the same OS kernel while images are layered file systems sharing common files to make efficient use of RAM and disk and start instantly.

Containers

A technical view into the *shared and layered* file systems technology

Docker uses a copy-on-write (union) filesystem
New files(& edits) are only visible to current/above layers



Layers allow for reuse

- More containers per host
- Faster start-up/download time – base layers are "cached"

Images

- Tarball of layers (each layer is a tarball)

Why do you need Container Orchestration?

Vs.

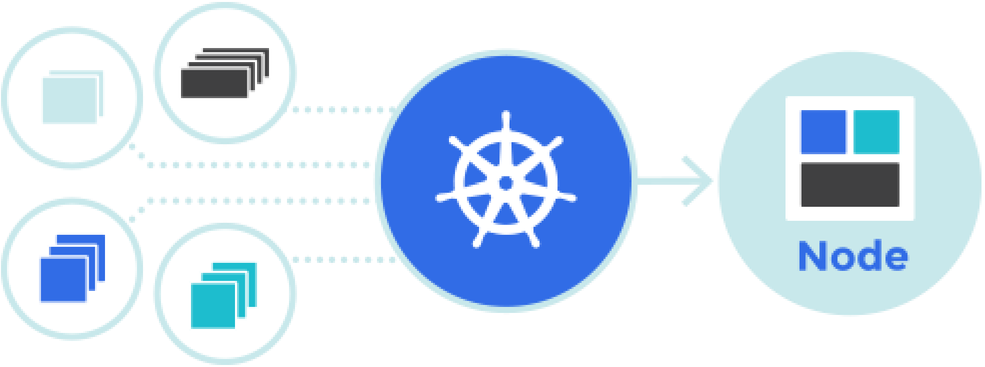
Pets

Cattle

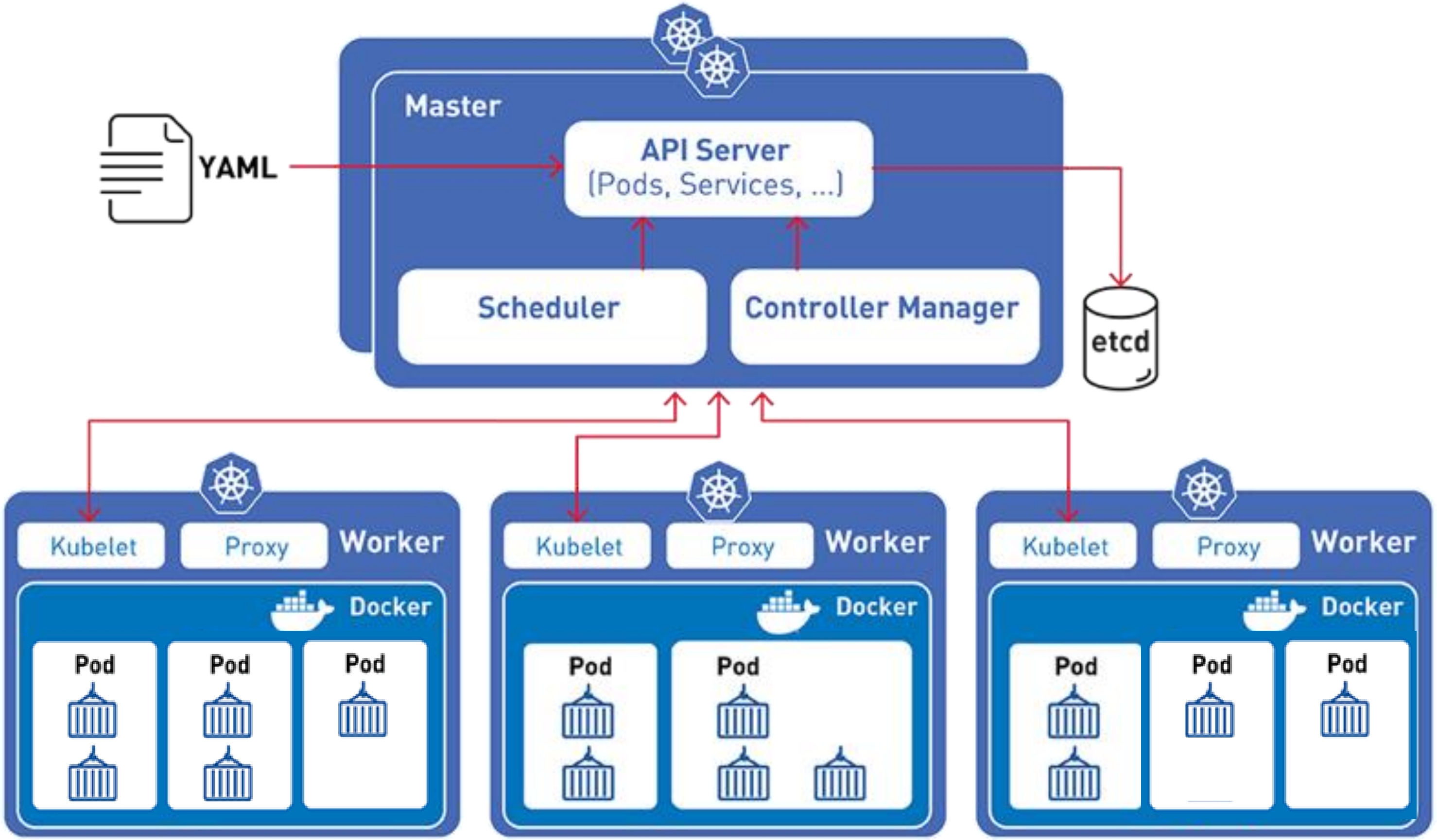
What is Kubernetes ?

“Kubernetes is an *open-source* platform for automating deployment, scaling, and management of containerized applications across clusters of nodes”

CLOUD NATIVE
COMPUTING FOUNDATION



Kubernetes – Simplified architecture



Kubernetes Capabilities

Intelligent Scheduling

Self-healing

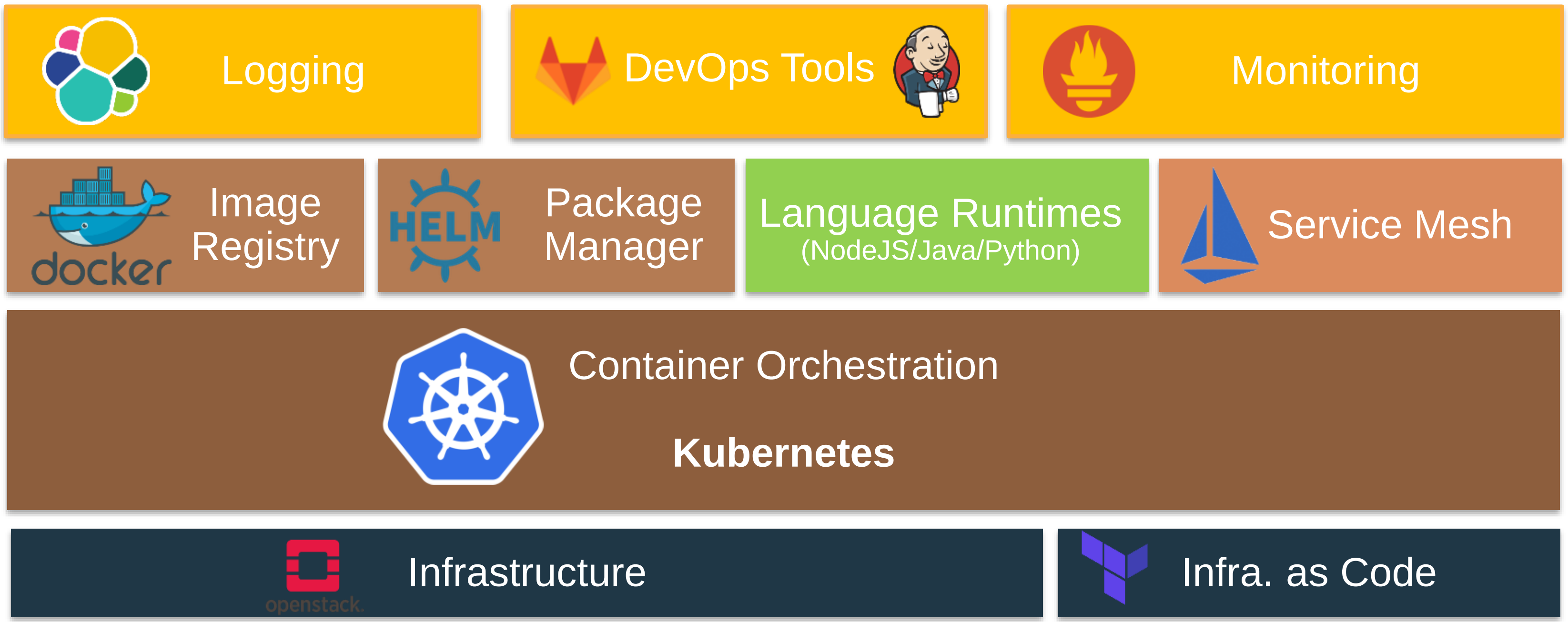
Horizontal scaling

Service discovery & load balancing

Automated rollouts and rollbacks

Secret and configuration management

A few Kubernetes companions (based on Open Technologies)



Logging

DevOps Tools

Monitoring

Image Registry

Package Manager

Language Runtimes
(NodeJS/Java/Python)

Service Mesh

Container Orchestration

Kubernetes

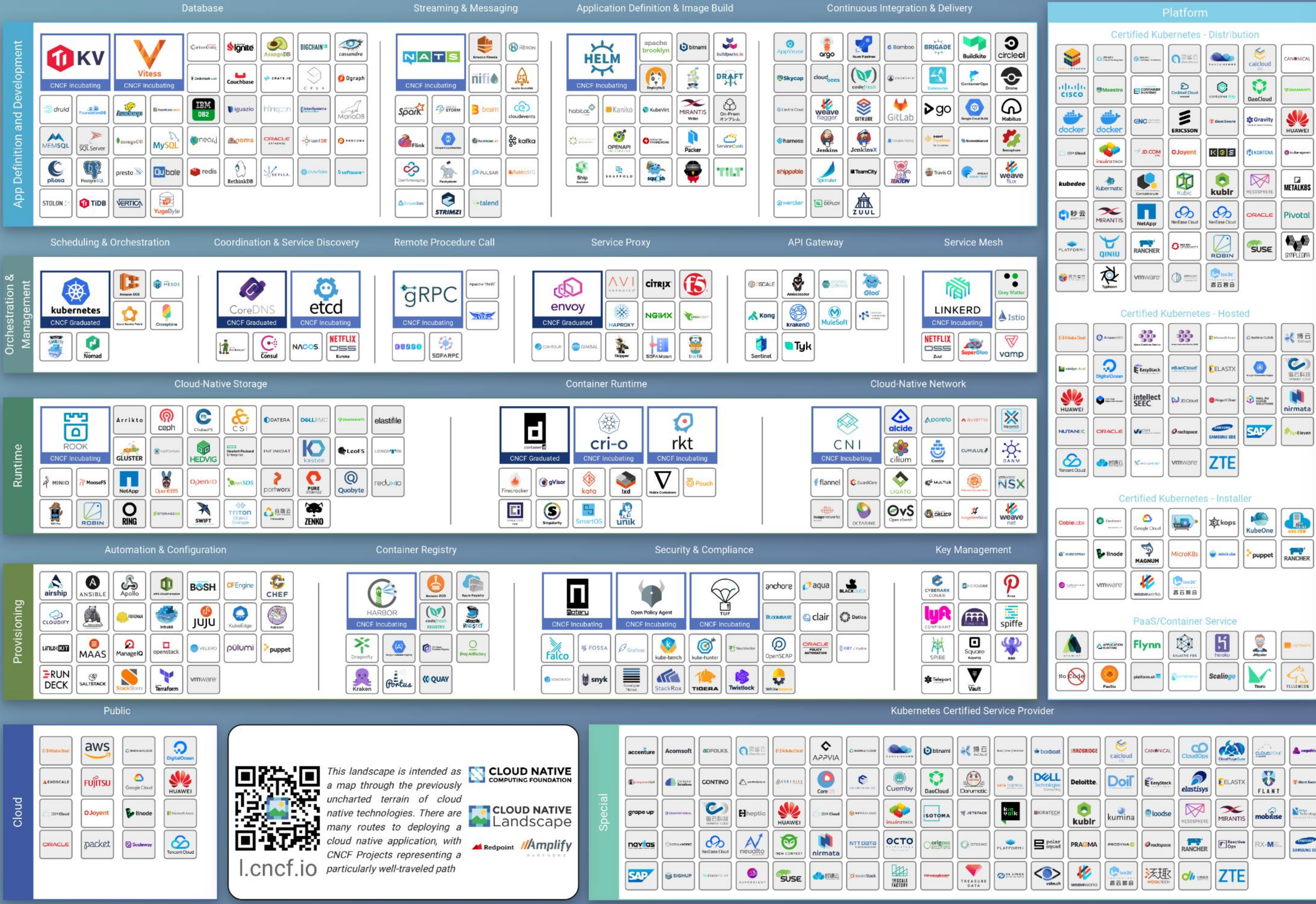
Infrastructure

Infra. as Code

CNCF Cloud Native Landscape

2019-06-11T02:27:14Z 6c8a4cf

Overwhelmed? Please see the CNCF Trail Map. That and the interactive landscape are at l.cncf.io



App Definition and Development

Database

Streaming & Messaging

Application Definition & Image Build

Continuous Integration & Delivery

Orchestration & Management

Scheduling & Orchestration

Coordination & Service Discovery

Remote Procedure Call

Service Proxy

API Gateway

Service Mesh

Runtime

Cloud-Native Storage

Container Runtime

Cloud-Native Network

Provisioning

Automation & Configuration

Container Registry

Security & Compliance

Key Management

Platform

Certified Kubernetes - Distribution

Certified Kubernetes - Hosted

Certified Kubernetes - Installer

PaaS/Container Service

Cloud

Public

This landscape is intended as a map through the previously uncharted terrain of cloud native technologies. There are many routes to deploying a cloud native application, with CNCF Projects representing a particularly well-traveled path

l.cncf.io

CLOUD NATIVE COMPUTING FOUNDATION

CLOUD NATIVE Landscape

Redpoint

Amplify PARTNERS

Special

Kubernetes Certified Service Provider

Overwhelmed? Please see the CNCF Trail Map. That and the interactive landscape are at l.cncf.io

Greyed logos are not open source

Streaming & Messaging

NATS
CNCF Incubating

Application Definition & Image Build

HELM
CNCF Incubating

Continuous Integration & Delivery

Platform

Certified Kubernetes - Distribution

Certified Kubernetes - Hosted

Certified Kubernetes - Installer

PaaS/Container Service

Observability and Analysis

Monitoring

Prometheus
CNCF Graduated

Logging

fluentd
CNCF Graduated

Tracing

JAEGER
CNCF Incubating

OPENTRACING
CNCF Incubating

Chaos Engineering

iscovery

etcd
CNCF Incubating

Remote Procedure Call

gRPC
CNCF Incubating

Service Proxy

envoy
CNCF Graduated

API Gateway

Service Mesh

LINKERD
CNCF Incubating

Container Runtime

containerd
CNCF Graduated

cri-o
CNCF Incubating

rkt
CNCF Incubating

Cloud-Native Network

CNI
CNCF Incubating

Container Registry

HARBOR
CNCF Incubating

Security & Compliance

Notary
CNCF Incubating

Open Policy Agent
CNCF Incubating

TUF
CNCF Incubating

Key Management

Serverless

Kubernetes Certified Service Provider

Special

Kubernetes Training Partner

This landscape is intended as a map through the previously uncharted terrain of cloud native technologies. There are many routes to deploying a cloud native application, with CNCF Projects representing a particularly well-traveled path

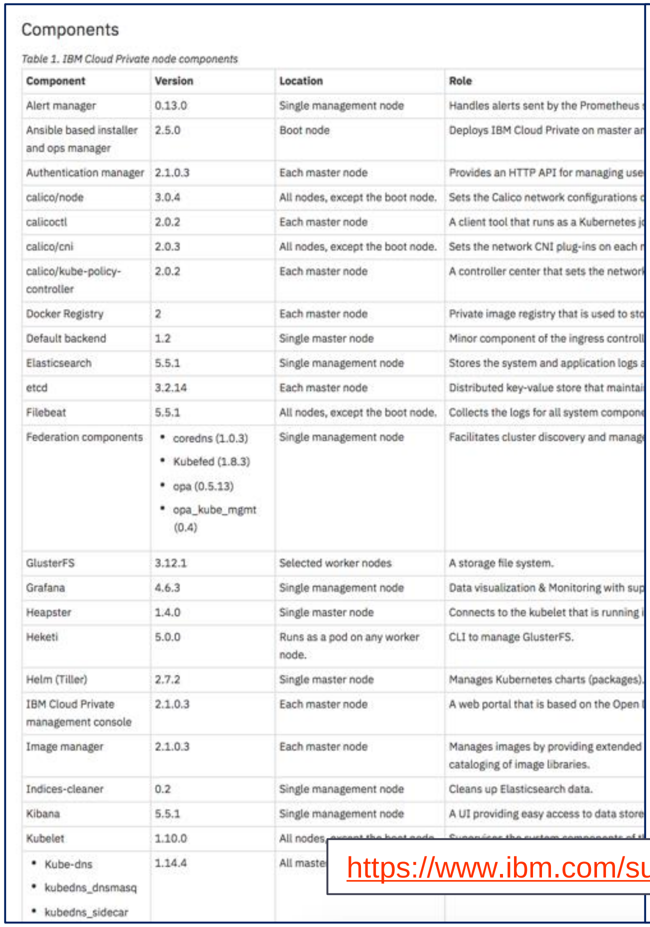
CLOUD NATIVE COMPUTING FOUNDATION

CLOUD NATIVE Landscape

Redpoint

Amplify PARTNERS

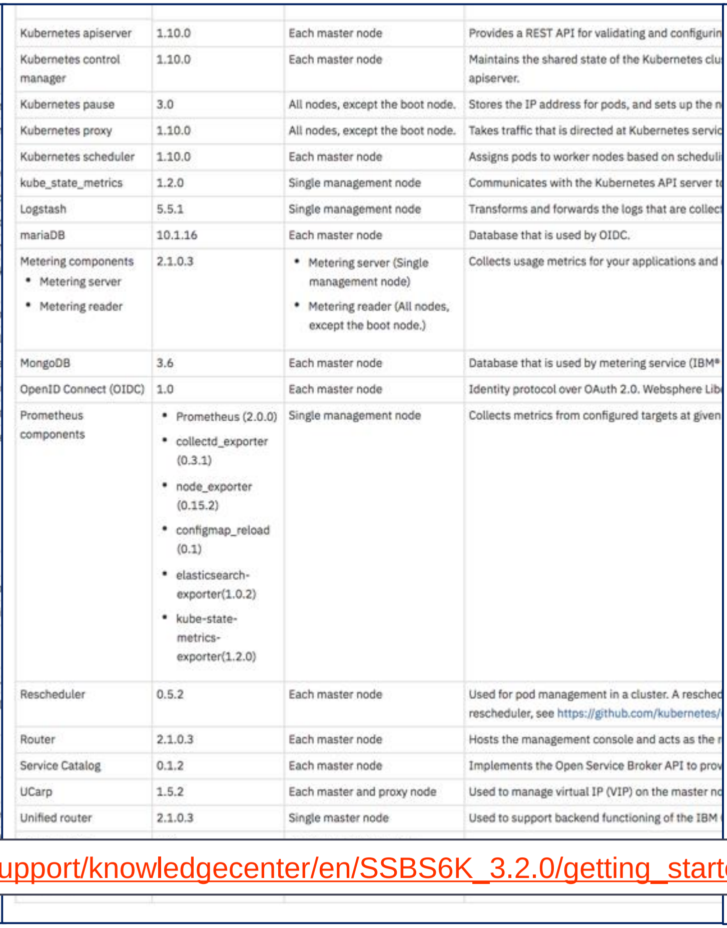
A production Kubernetes environment is at last 50 Components



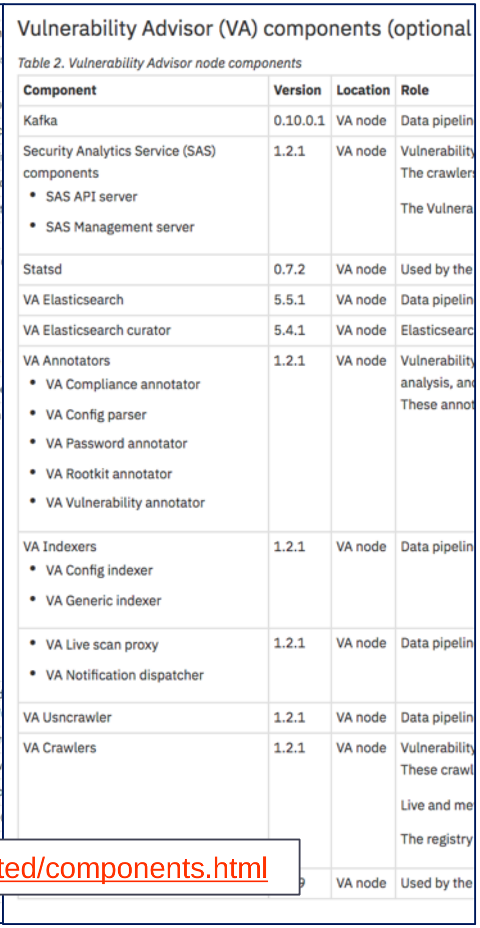
Components

Table 1. IBM Cloud Private node components

Component	Version	Location	Role
Alert manager	0.13.0	Single management node	Handles alerts sent by the Prometheus
Ansible based installer and ops manager	2.5.0	Boot node	Deploys IBM Cloud Private on master ar
Authentication manager	2.1.0.3	Each master node	Provides an HTTP API for managing use
calico/node	3.0.4	All nodes, except the boot node.	Sets the Calico network configurations c
calicoctl	2.0.2	Each master node	A client tool that runs as a Kubernetes j
calico/cni	2.0.3	All nodes, except the boot node.	Sets the network CNI plug-ins on each r
calico/kube-policy-controller	2.0.2	Each master node	A controller center that sets the network
Docker Registry	2	Each master node	Private image registry that is used to sto
Default backend	1.2	Single master node	Minor component of the ingress controll
Elasticsearch	5.5.1	Single management node	Stores the system and application logs a
etcd	3.2.14	Each master node	Distributed key-value store that maintai
Filebeat	5.5.1	All nodes, except the boot node.	Collects the logs for all system compone
Federation components	• coredns (1.0.3) • Kubefed (1.8.3) • opa (0.5.13) • opa_kube_mgmt (0.4)	Single management node	Facilitates cluster discovery and manag
GlusterFS	3.12.1	Selected worker nodes	A storage file system.
Grafana	4.6.3	Single management node	Data visualization & Monitoring with sup
Heapster	1.4.0	Single master node	Connects to the kubelet that is running i
Heketi	5.0.0	Runs as a pod on any worker node.	CLI to manage GlusterFS.
Helm (Tiller)	2.7.2	Single master node	Manages Kubernetes charts (packages).
IBM Cloud Private management console	2.1.0.3	Each master node	A web portal that is based on the Open
Image manager	2.1.0.3	Each master node	Manages images by providing extended cataloging of image libraries.
Indices-cleaner	0.2	Single management node	Cleans up Elasticsearch data.
Kibana	5.5.1	Single management node	A UI providing easy access to data store
Kubelet	1.10.0	All nodes, except the boot node	Supervises the system components of th
• Kube-dns • kubedns_dnsmasq • kubedns_sidecar	1.14.4	All maste	



Kubernetes apiserver	1.10.0	Each master node	Provides a REST API for validating and configurin
Kubernetes control manager	1.10.0	Each master node	Maintains the shared state of the Kubernetes clu apiserver.
Kubernetes pause	3.0	All nodes, except the boot node.	Stores the IP address for pods, and sets up the n
Kubernetes proxy	1.10.0	All nodes, except the boot node.	Takes traffic that is directed at Kubernetes servic
Kubernetes scheduler	1.10.0	Each master node	Assigns pods to worker nodes based on scheduli
kube_state_metrics	1.2.0	Single management node	Communicates with the Kubernetes API server to
Logstash	5.5.1	Single management node	Transforms and forwards the logs that are collect
mariaDB	10.1.16	Each master node	Database that is used by OIDC.
Metering components • Metering server • Metering reader	2.1.0.3	• Metering server (Single management node) • Metering reader (All nodes, except the boot node.)	Collects usage metrics for your applications and
MongoDB	3.6	Each master node	Database that is used by metering service (IBM®
OpenID Connect (OIDC)	1.0	Each master node	Identity protocol over OAuth 2.0. Websphere Lib
Prometheus components	• Prometheus (2.0.0) • collectd_exporter (0.3.1) • node_exporter (0.15.2) • configmap_reload (0.1) • elasticsearch-exporter(1.0.2) • kube-state-metrics-exporter(1.2.0)	Single management node	Collects metrics from configured targets at given
Rescheduler	0.5.2	Each master node	Used for pod management in a cluster. A resched rescheduler, see https://github.com/kubernetes/
Router	2.1.0.3	Each master node	Hosts the management console and acts as the r
Service Catalog	0.1.2	Each master node	Implements the Open Service Broker API to prov
UCarp	1.5.2	Each master and proxy node	Used to manage virtual IP (VIP) on the master no
Unified router	2.1.0.3	Single master node	Used to support backend functioning of the IBM



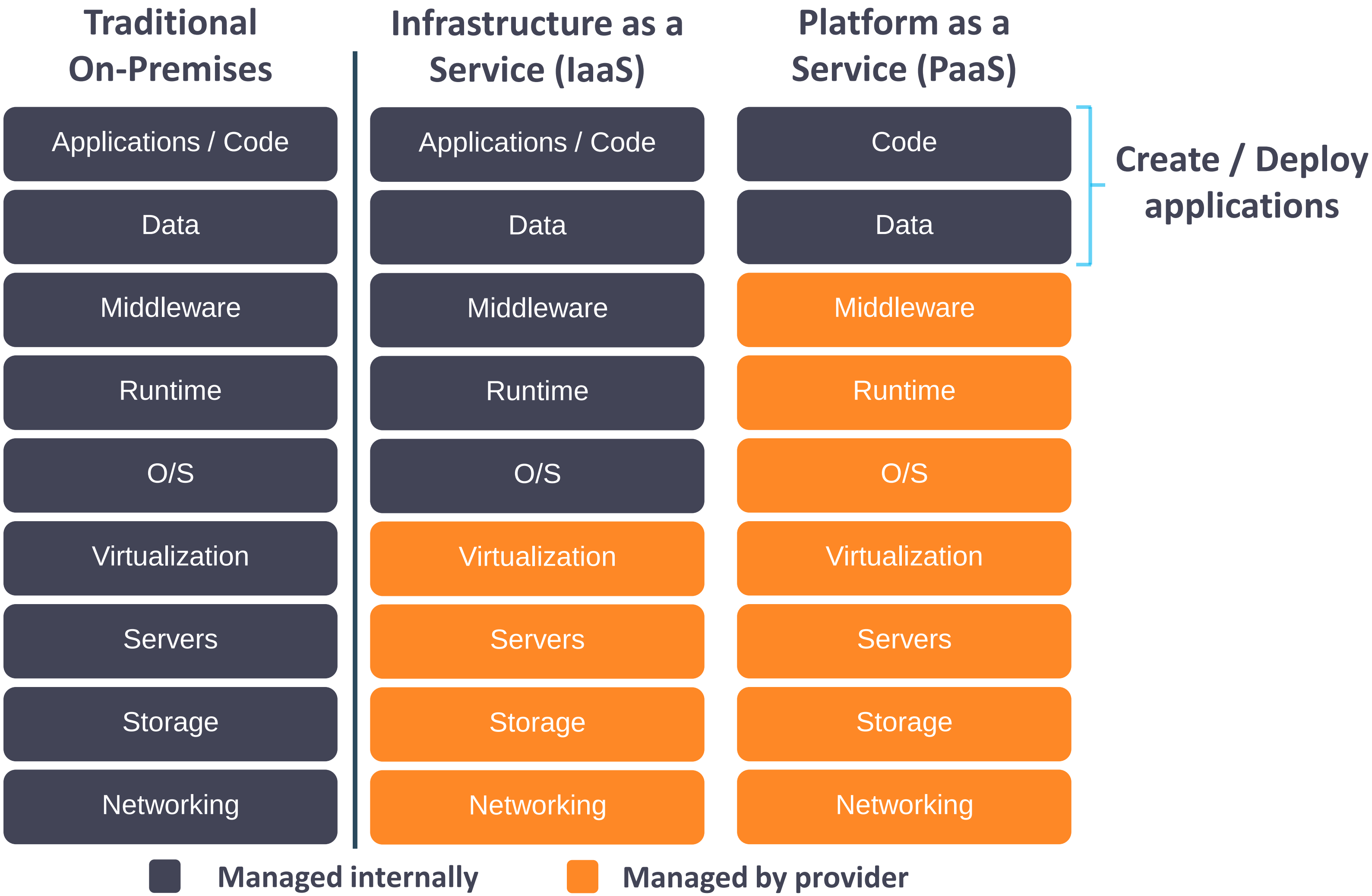
Vulnerability Advisor (VA) components (optional

Table 2. Vulnerability Advisor node components

Component	Version	Location	Role
Kafka	0.10.0.1	VA node	Data pipelin
Security Analytics Service (SAS) components • SAS API server • SAS Management server	1.2.1	VA node	Vulnerability The crawlers The Vulnera
Statsd	0.7.2	VA node	Used by the
VA Elasticsearch	5.5.1	VA node	Data pipelin
VA Elasticsearch curator	5.4.1	VA node	Elasticsearc
VA Annotators • VA Compliance annotator • VA Config parser • VA Password annotator • VA Rootkit annotator • VA Vulnerability annotator	1.2.1	VA node	Vulnerability analysis, and These annot
VA Indexers • VA Config indexer • VA Generic indexer	1.2.1	VA node	Data pipelin
• VA Live scan proxy • VA Notification dispatcher	1.2.1	VA node	Data pipelin
VA Usncrawler	1.2.1	VA node	Data pipelin
VA Crawlers	1.2.1	VA node	Vulnerability These crawl Live and me The registry
	9	VA node	Used by the

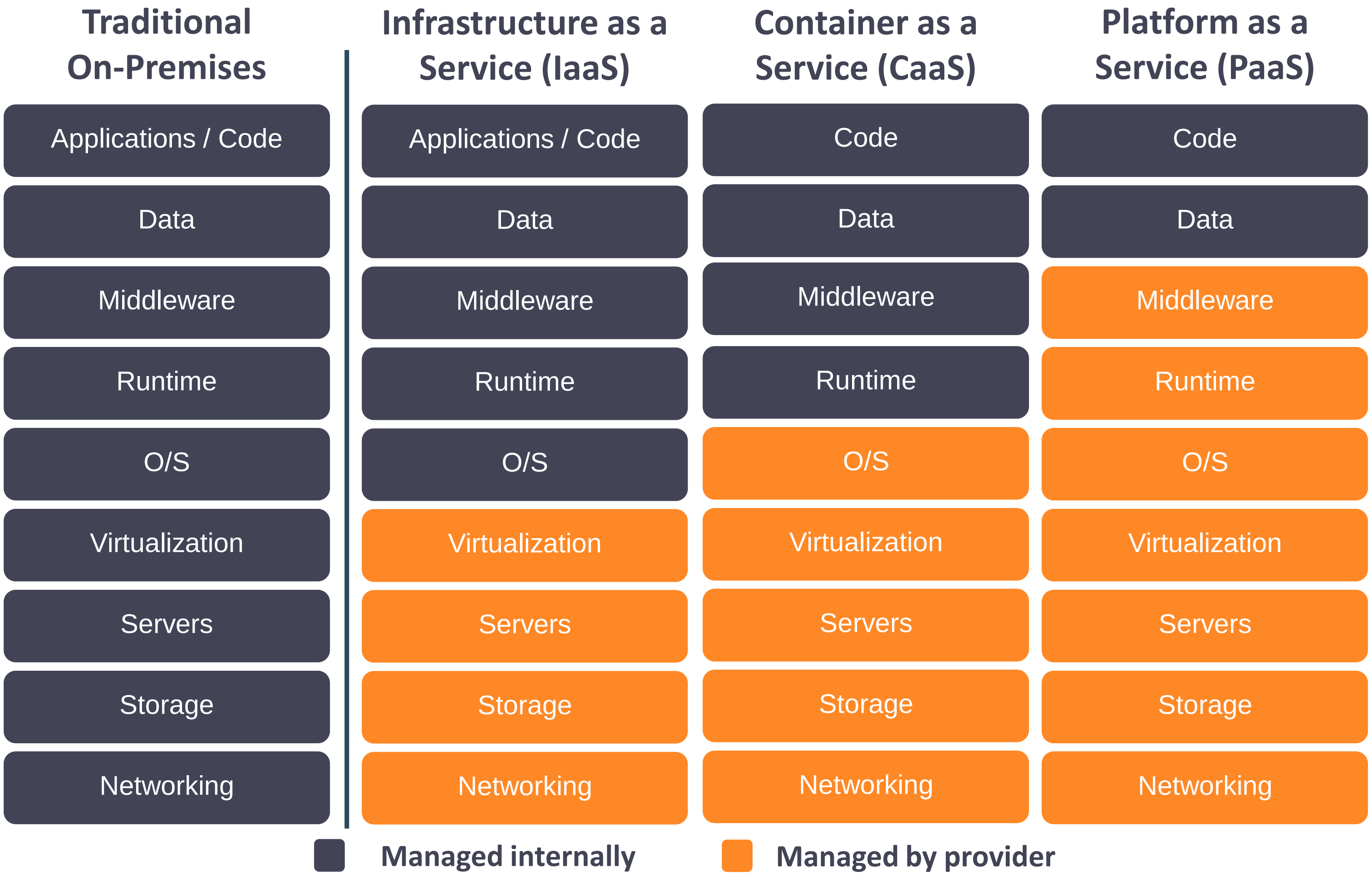
https://www.ibm.com/support/knowledgecenter/en/SSBS6K_3.2.0/getting_started/components.html

Do you want to build it yourself?



Traditional On-Premises	Infrastructure as a Service (IaaS)	Platform as a Service (PaaS)
Applications / Code	Applications / Code	Code
Data	Data	Data
Middleware	Middleware	Middleware
Runtime	Runtime	Runtime
O/S	O/S	O/S
Virtualization	Virtualization	Virtualization
Servers	Servers	Servers
Storage	Storage	Storage
Networking	Networking	Networking

⬛ Managed internally 🟧 Managed by provider

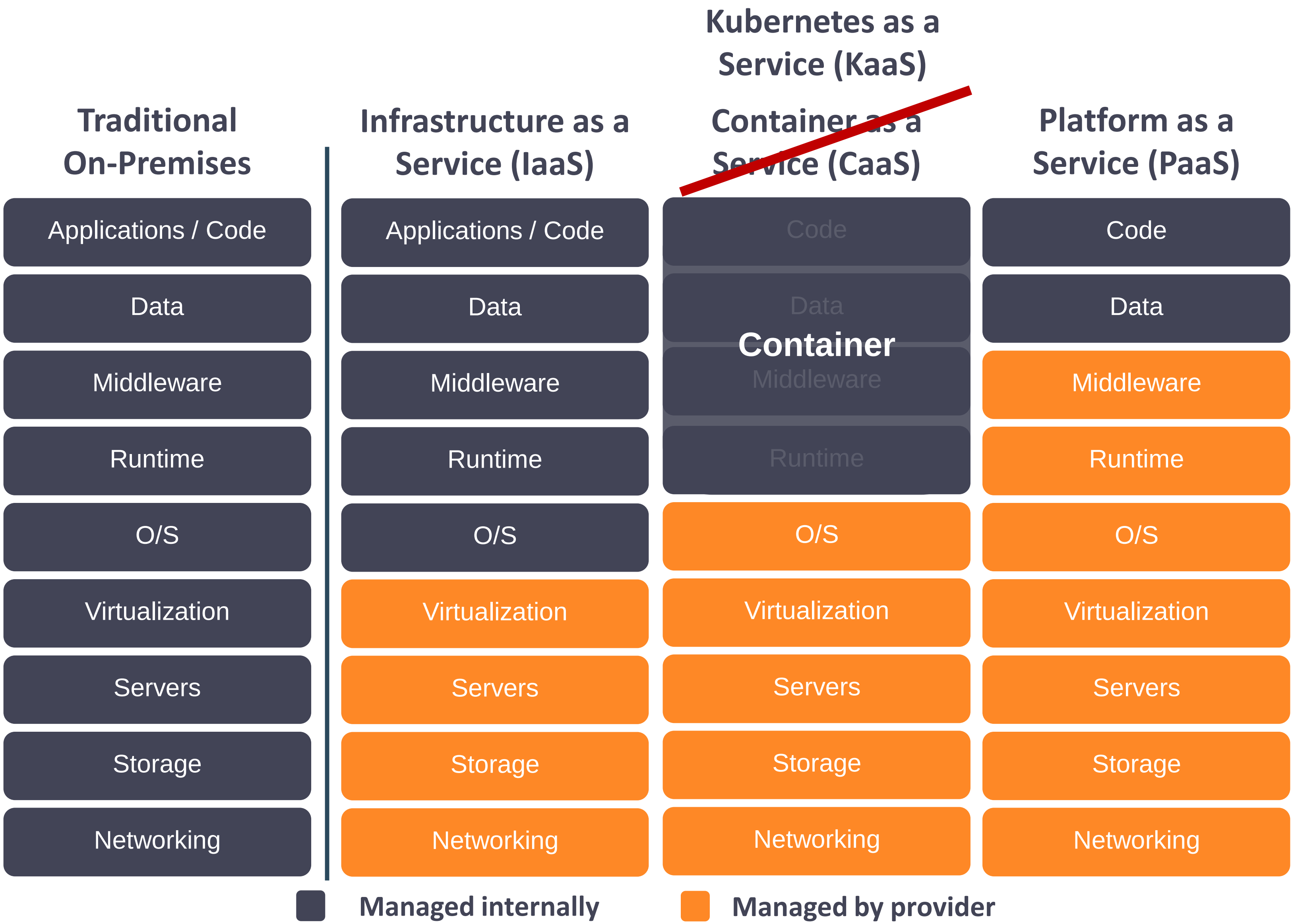


Traditional On-Premises	Infrastructure as a Service (IaaS)	Container as a Service (CaaS)	Platform as a Service (PaaS)
⬛ Applications / Code	⬛ Applications / Code	⬛ Code	⬛ Code
⬛ Data	⬛ Data	⬛ Data	⬛ Data
⬛ Middleware	⬛ Middleware	⬛ Middleware	🟧 Middleware
⬛ Runtime	⬛ Runtime	⬛ Runtime	🟧 Runtime
⬛ O/S	⬛ O/S	🟧 O/S	🟧 O/S
⬛ Virtualization	🟧 Virtualization	🟧 Virtualization	🟧 Virtualization
⬛ Servers	🟧 Servers	🟧 Servers	🟧 Servers
⬛ Storage	🟧 Storage	🟧 Storage	🟧 Storage
⬛ Networking	🟧 Networking	🟧 Networking	🟧 Networking

⬛ Managed internally 🟧 Managed by provider

Traditional On-Premises	Infrastructure as a Service (IaaS)	Container as a Service (CaaS)	Platform as a Service (PaaS)
⬛ Applications / Code	⬛ Applications / Code	⬛ Code	⬛ Code
⬛ Data	⬛ Data	⬛ Data Container	⬛ Data
⬛ Middleware	⬛ Middleware	⬛ Middleware	🟧 Middleware
⬛ Runtime	⬛ Runtime	⬛ Runtime	🟧 Runtime
⬛ O/S	⬛ O/S	🟧 O/S	🟧 O/S
⬛ Virtualization	🟧 Virtualization	🟧 Virtualization	🟧 Virtualization
⬛ Servers	🟧 Servers	🟧 Servers	🟧 Servers
⬛ Storage	🟧 Storage	🟧 Storage	🟧 Storage
⬛ Networking	🟧 Networking	🟧 Networking	🟧 Networking

⬛ **Managed internally** 🟧 **Managed by provider**



Traditional On-Premises	Infrastructure as a Service (IaaS)	Kubernetes as a Service (KaaS) ~~Container as a Service (CaaS)~~	Platform as a Service (PaaS)
⬛ Applications / Code	⬛ Applications / Code	⬛ Code	⬛ Code
⬛ Data	⬛ Data	⬛ Data Container	⬛ Data
⬛ Middleware	⬛ Middleware	⬛ Middleware	🟧 Middleware
⬛ Runtime	⬛ Runtime	⬛ Runtime	🟧 Runtime
⬛ O/S	⬛ O/S	🟧 O/S	🟧 O/S
⬛ Virtualization	🟧 Virtualization	🟧 Virtualization	🟧 Virtualization
⬛ Servers	🟧 Servers	🟧 Servers	🟧 Servers
⬛ Storage	🟧 Storage	🟧 Storage	🟧 Storage
⬛ Networking	🟧 Networking	🟧 Networking	🟧 Networking

⬛ Managed internally 🟧 Managed by provider

IBM Cloud Kubernetes Services

certified kubernetes 1.10 · certified kubernetes 1.11 · certified kubernetes 1.12 · certified kubernetes 1.13

A **certified, managed Kubernetes service**

Built-in **security and isolation** to enable rapid delivery of apps.

Available in six IBM regions WW, including **40+ datacenters**.

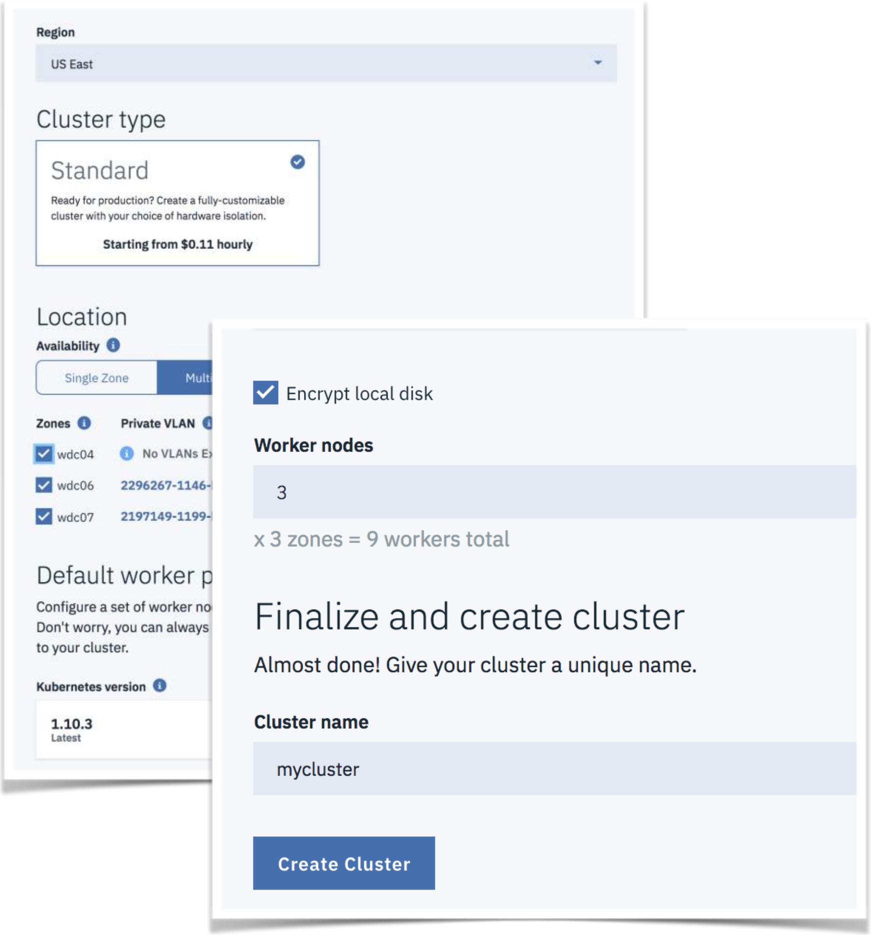
Fully **dedicated, single tenant clusters** deployed within customer account and network

Seamless integration with IBM Cloud services

Portability with native Kubernetes experience and **full API support**

Try it at : https://cloud.ibm.com

IBM **Cloud**
Kubernetes Service



Region
US East

Cluster type

Standard ✓
Ready for production? Create a fully-customizable cluster with your choice of hardware isolation.
Starting from $0.11 hourly

Location
Availability

Zones	Private VLAN
☑ wdc04	No VLANs E
☑ wdc06	2296267-1146-
☑ wdc07	2197149-1199-

Default worker p
Configure a set of worker no
Don't worry, you can always
to your cluster.

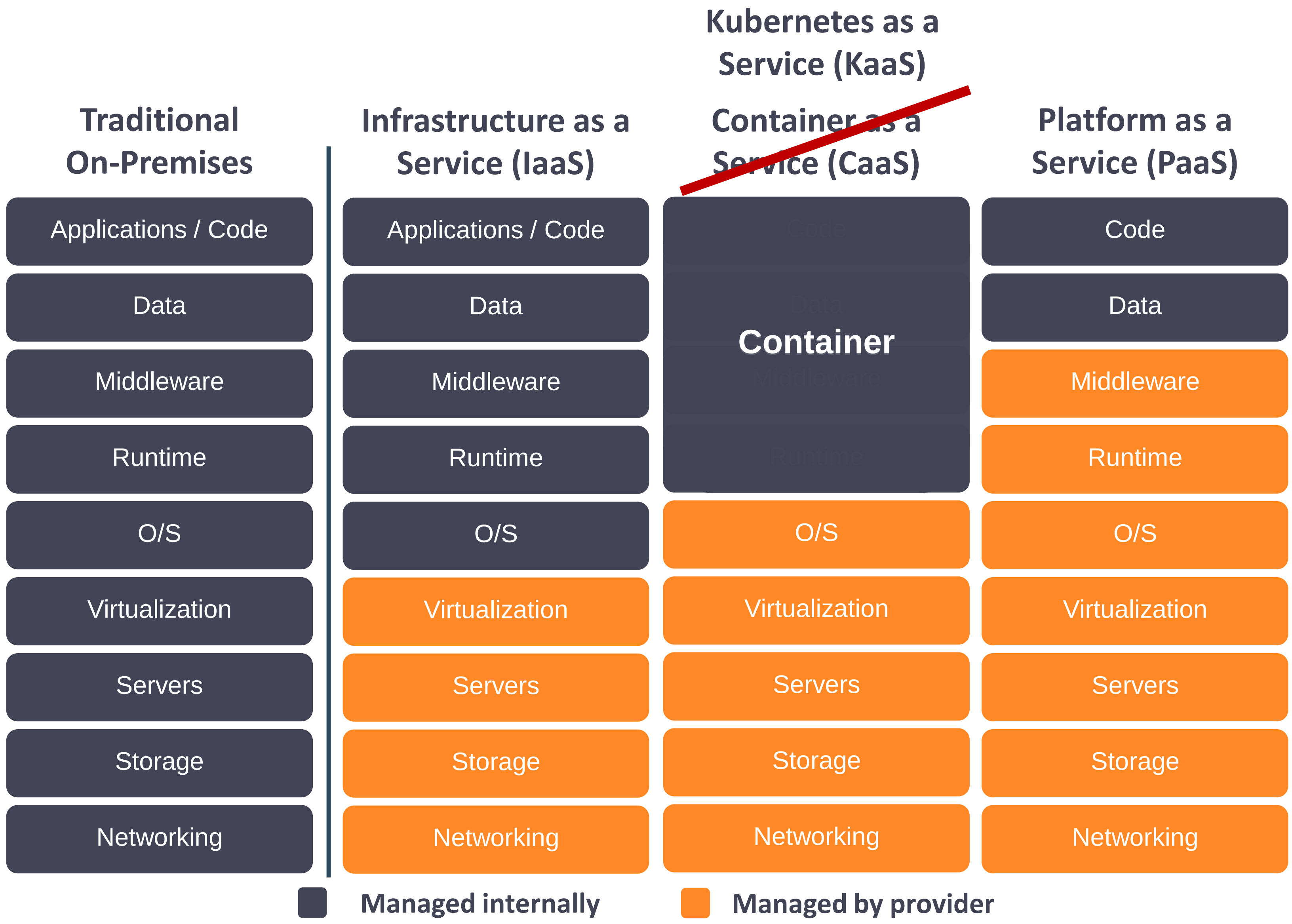
Kubernetes version
1.10.3
Latest

☑ Encrypt local disk

Worker nodes
3
x 3 zones = 9 workers total

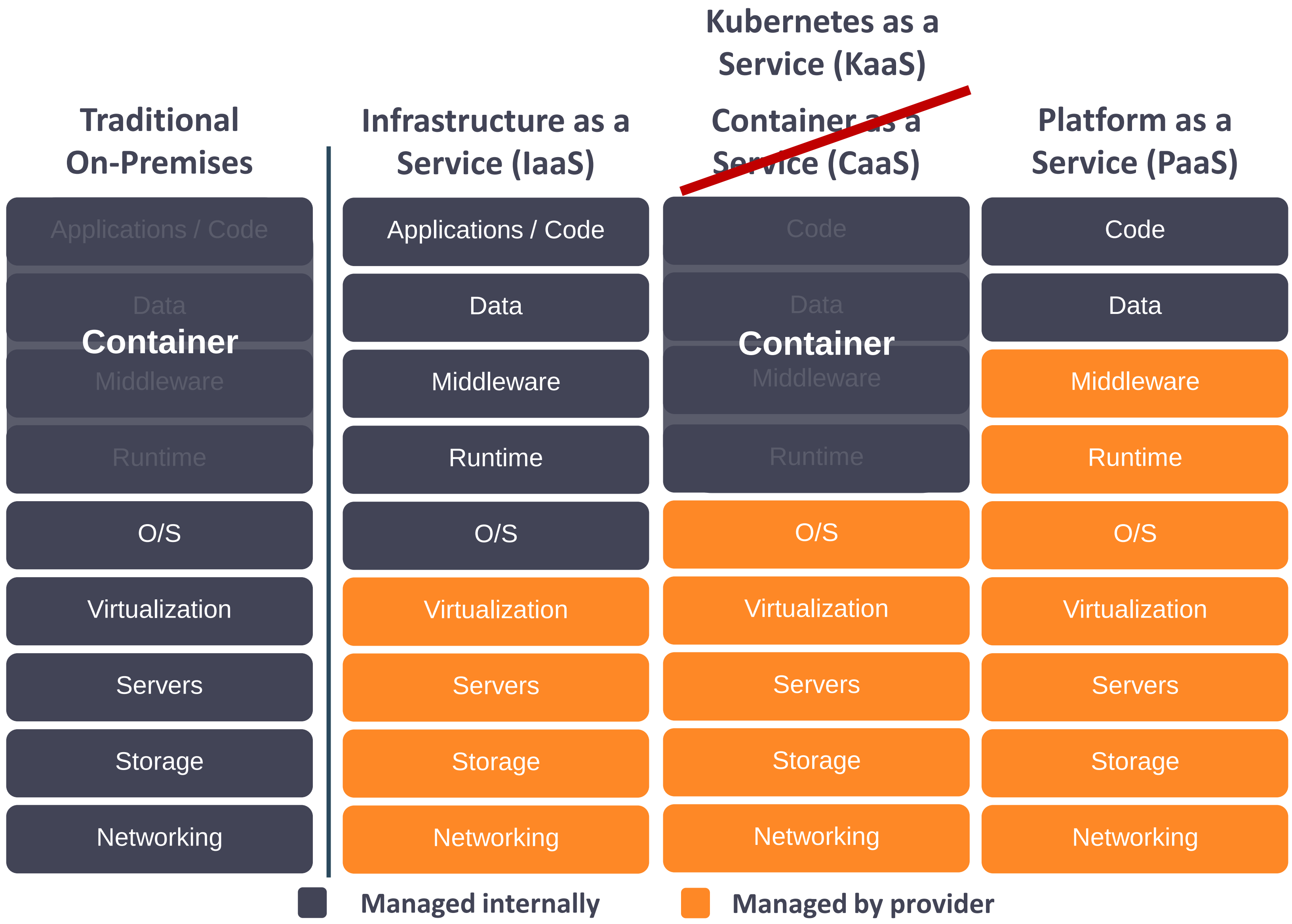
Finalize and create cluster
Almost done! Give your cluster a unique name.

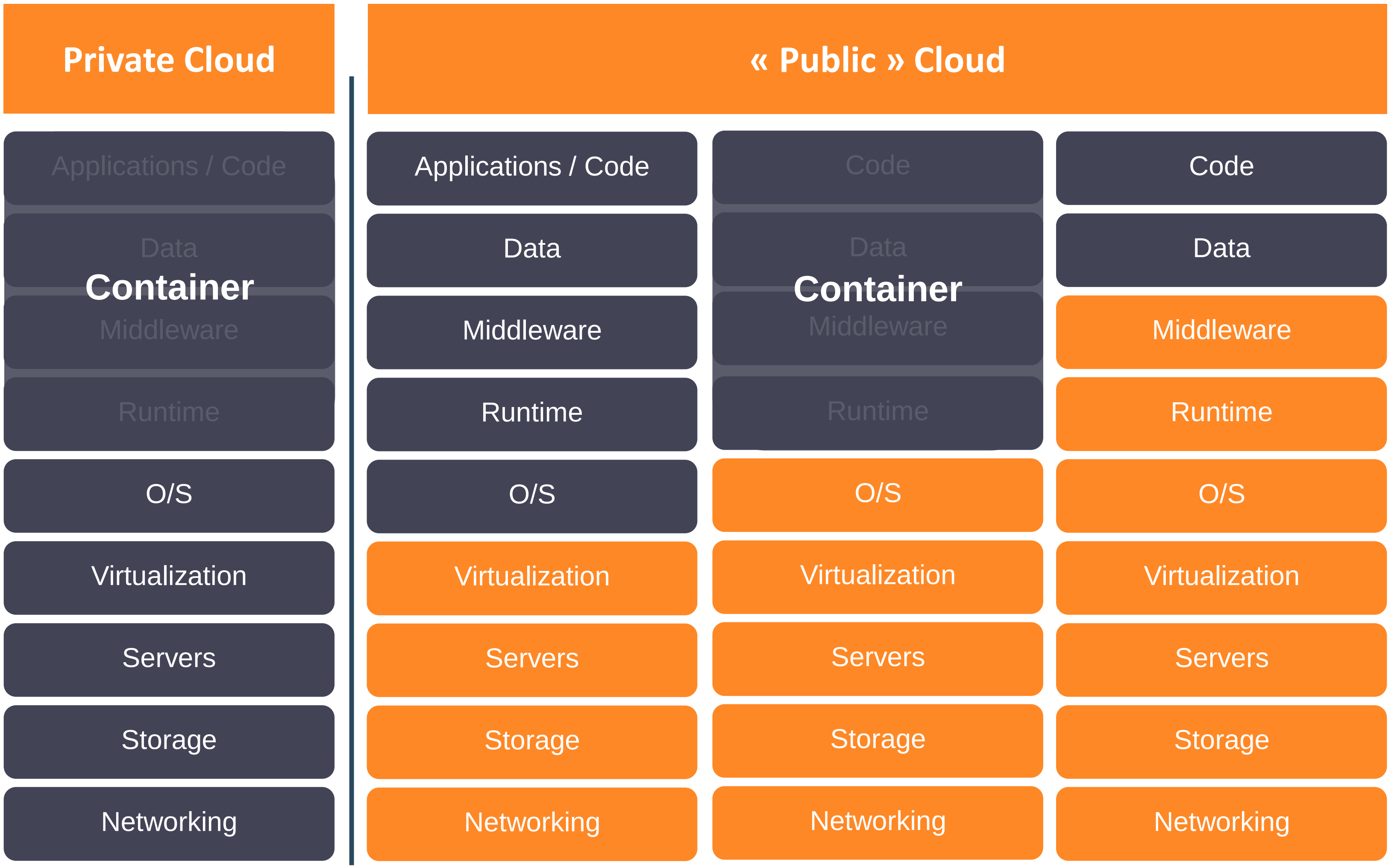
Cluster name
mycluster



Traditional On-Premises	Infrastructure as a Service (IaaS)	Kubernetes as a Service (KaaS) ~~Container as a Service (CaaS)~~	Platform as a Service (PaaS)
⬛ Applications / Code	⬛ Applications / Code	⬛ Container Code	⬛ Code
⬛ Data	⬛ Data	⬛ Data	⬛ Data
⬛ Middleware	⬛ Middleware	⬛ Middleware	🟧 Middleware
⬛ Runtime	⬛ Runtime	⬛ Runtime	🟧 Runtime
⬛ O/S	⬛ O/S	🟧 O/S	🟧 O/S
⬛ Virtualization	🟧 Virtualization	🟧 Virtualization	🟧 Virtualization
⬛ Servers	🟧 Servers	🟧 Servers	🟧 Servers
⬛ Storage	🟧 Storage	🟧 Storage	🟧 Storage
⬛ Networking	🟧 Networking	🟧 Networking	🟧 Networking

⬛ Managed internally 🟧 Managed by provider



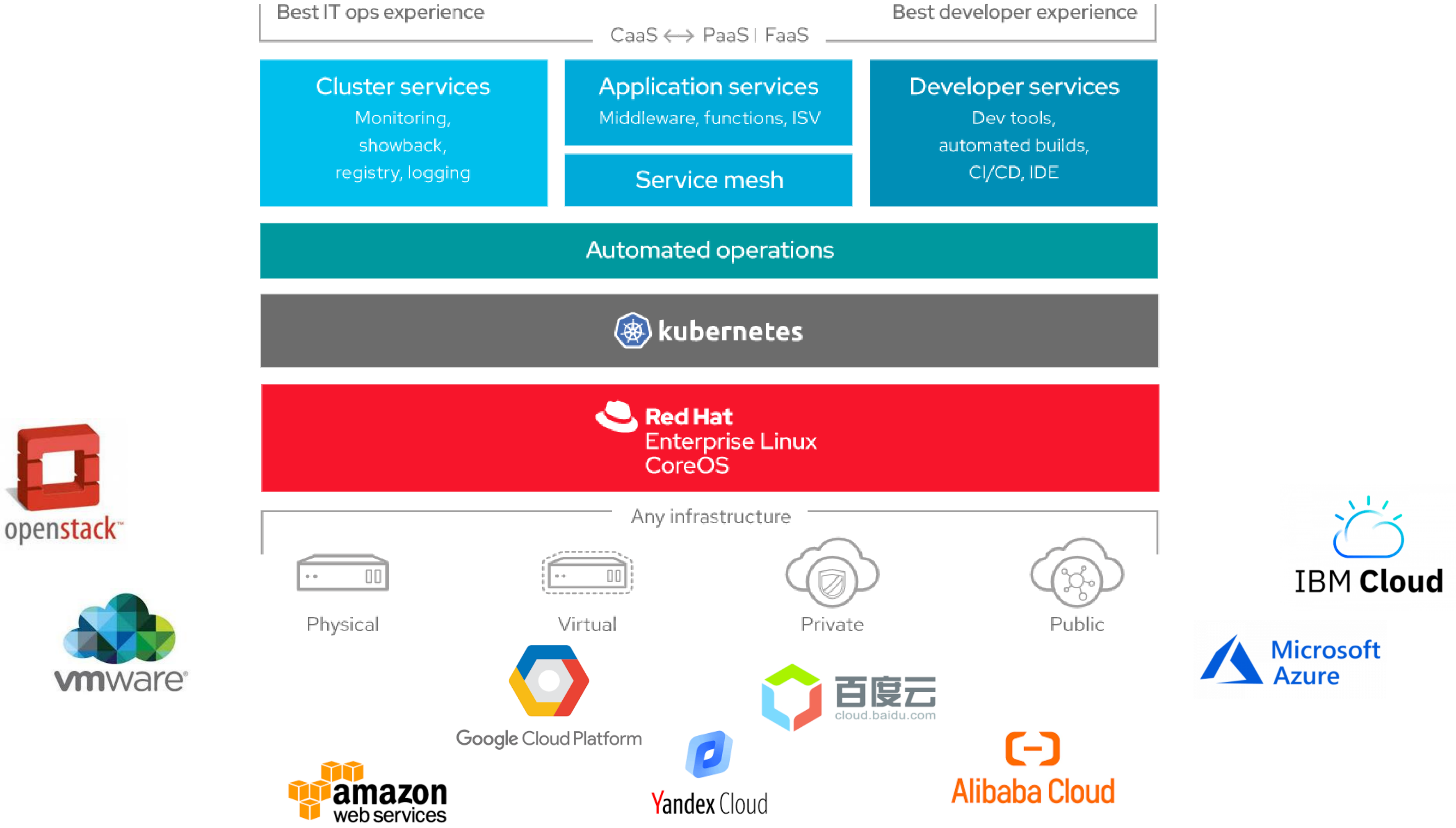


Private Cloud	« Public » Cloud		
■ Applications / Code	■ Applications / Code	■ Code	■ Code
■ Data Container	■ Data	■ Data Container	■ Data
■ Middleware	■ Middleware	■ Middleware	🟧 Middleware
■ Runtime	■ Runtime	■ Runtime	🟧 Runtime
■ O/S	■ O/S	🟧 O/S	🟧 O/S
■ Virtualization	🟧 Virtualization	🟧 Virtualization	🟧 Virtualization
■ Servers	🟧 Servers	🟧 Servers	🟧 Servers
■ Storage	🟧 Storage	🟧 Storage	🟧 Storage
■ Networking	🟧 Networking	🟧 Networking	🟧 Networking

■ Managed internally 🟧 Managed by provider

Red Hat OpenShift?

Red Hat OpenShift is a hybrid cloud, enterprise Kubernetes application platform.



Best IT ops experience

CaaS ⟷ PaaS | FaaS

Best developer experience

Cluster services
Monitoring, showback, registry, logging

Application services
Middleware, functions, ISV

Service mesh

Developer services
Dev tools, automated builds, CI/CD, IDE

Automated operations

kubernetes

Red Hat
Enterprise Linux
CoreOS

Any infrastructure

Physical

Virtual

Private

Public

Multiple K8S on multiple Cloud?

Amazon EKS

RED HAT® OPENSHIFT Container Platform

Google Container Engine (GKE)

Google Container Registry

IBM Cloud Kubernetes Service

Alibaba Cloud

Яндекс Облако

Yandex Managed Service for Kubernetes

Azure Kubernetes Service (AKS)

Introducing Hybrid Cloud

Multi-cloud is the key to organizational agility

8 out of 10 committing to Multi-Cloud

71% use 3 or more clouds

Public

Getting new value from third parties

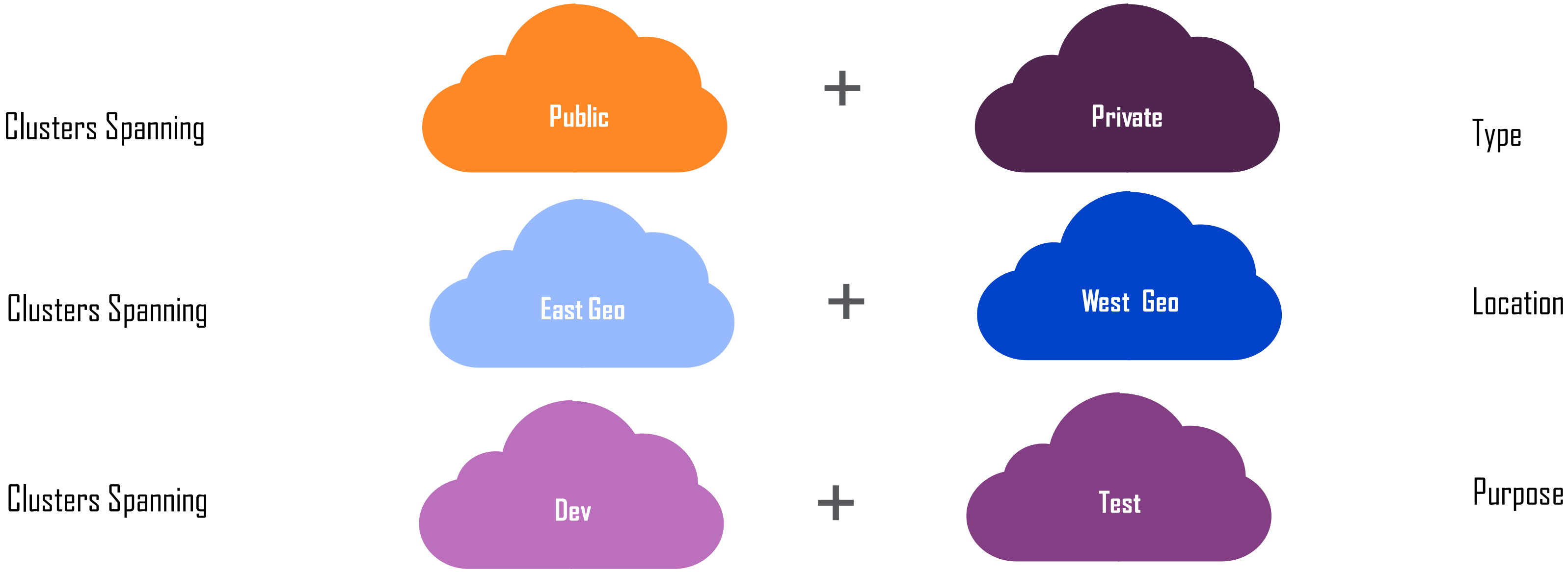
+

Private

Extracting value from your entire business

Organizations are deploying multiple clusters

9 out of 10 enterprises adopting Kubernetes have already deployed multiple clusters



Clusters Spanning Public + Private Type

Clusters Spanning East Geo + West Geo Location

Clusters Spanning Dev + Test Purpose

As organizations modernize and deploy ***containerized*** *clusters on multiple clouds, new challenges are introduced….*

I need broad Visibility

I need automated Governance

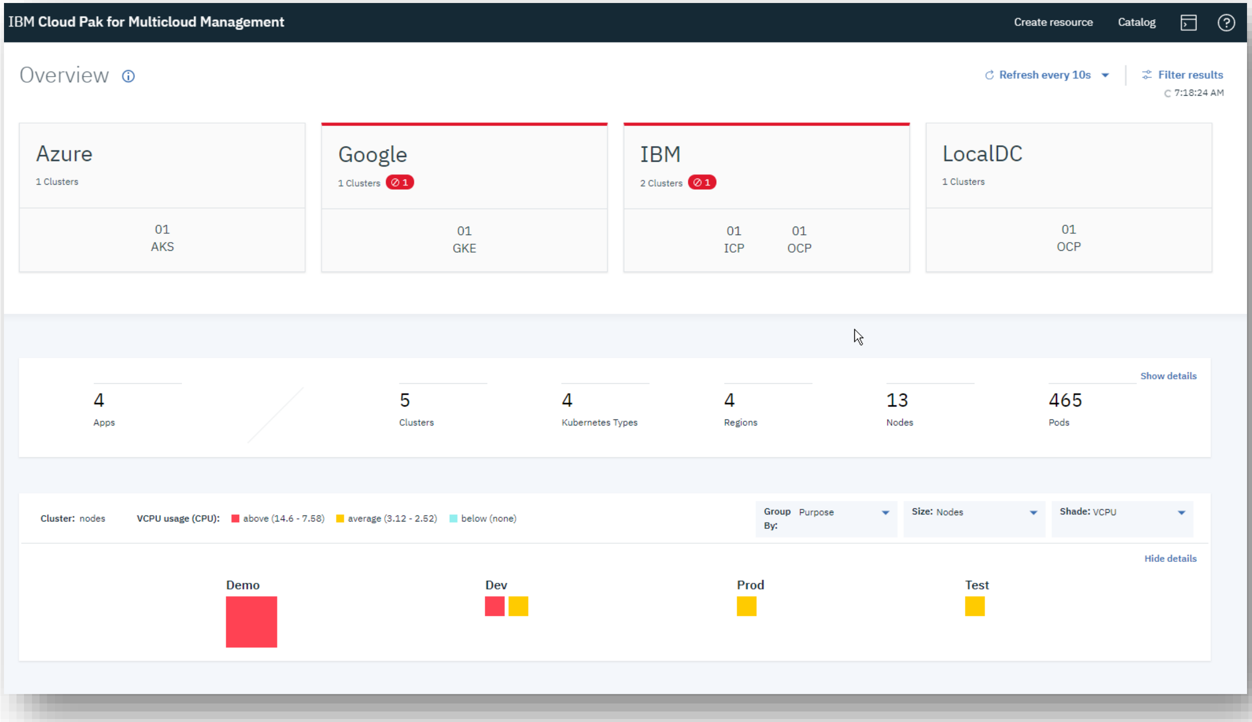
I need seamless Application Management

Visibility: Clear insight into any environment, any application, any cluster

One Dashboard, 360° view : See health, usage, policy adherence **on any cluster, any environment**

I need visibility and control

- Where are the failed components?
- Where are my services running?
- How can I monitor applications across clusters and clouds?
- How can I manage clusters as if they were one environment?
- How do I monitor usage across clouds?



Azure
1 Clusters
01 AKS

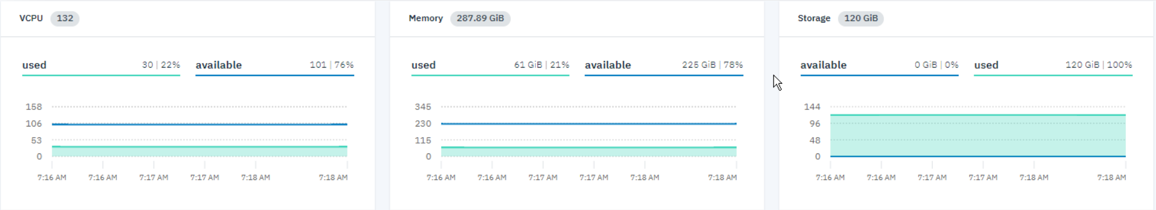
Google
1 Clusters ⊘ 1
01 GKE

IBM
2 Clusters ⊘ 1
01 ICP
01 OCP

LocalDC
1 Clusters
01 OCP

4 Apps
5 Clusters
4 Kubernetes Types
4 Regions
13 Nodes
465 Pods

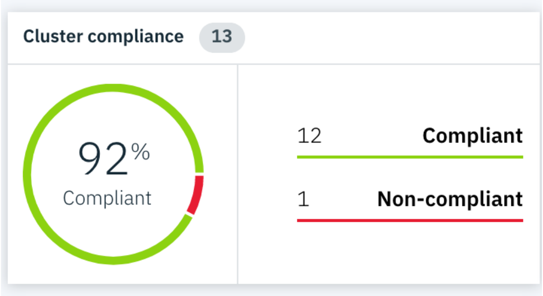
Cluster: nodes
VCPU usage (CPU): 🟥 above (14.6 - 7.58) 🟨 average (3.12 - 2.52) 🟦 below (none)
Group By: Purpose
Size: Nodes
Shade: VCPU


VCPU 132
used 30 | 22%
available 101 | 76%

Memory 287.89 GiB
used 61 GiB | 21%
available 225 GiB | 78%

Storage 120 GiB
available 0 GiB | 0%
used 120 GiB | 100%

Governance: Maintain controls across applications & clusters with policies



Cluster compliance 13

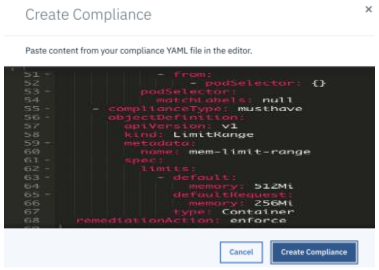
92% Compliant

12	Compliant
1	Non-compliant

I need automated Governance

- How do I set consistent security policies across environments?
- Which clusters are in compliance with our policies?
- How can I manage configuration across this large environment?
- How can I place workloads based on capacity, policy?

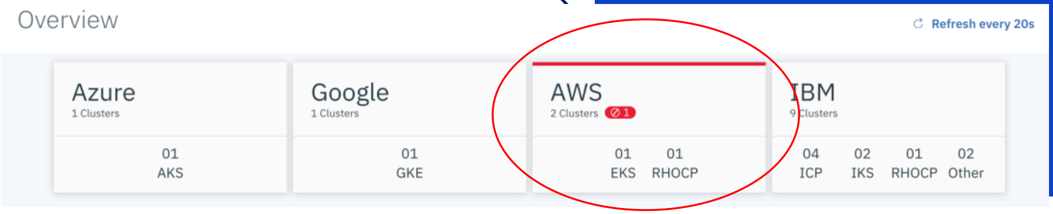
Create **Placement** & **Security policies** directly from MCM console and push to all clusters **with a click**



Create Compliance

Paste content from your compliance YAML file in the editor.

```
          - from:
              - podSelector: {}
        podSelector:
          matchLabels: null
    - complianceType: musthave
      objectDefinition:
        apiVersion: v1
        kind: LimitRange
        metadata:
          name: mem-limit-range
        spec:
          limits:
            - default:
                memory: 512Mi
              defaultRequest:
                memory: 256Mi
              type: Container
  remediationAction: enforce
```

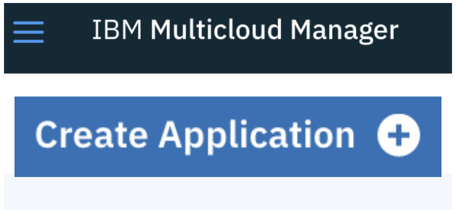


Overview

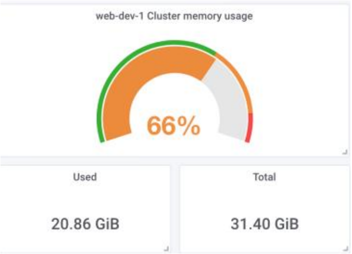
Refresh every 20s

Azure 1 Clusters	Google 1 Clusters	AWS 2 Clusters ⊘1	IBM 9 Clusters
01 AKS	01 GKE	01 EKS 01 RHOCP	04 ICP 02 IKS 01 RHOCP 02 Other

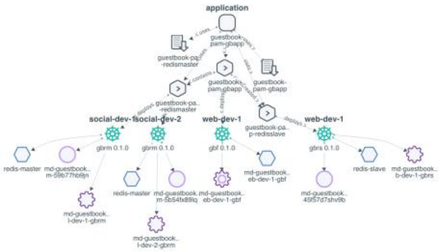
Application Management: Create, Monitor, Manage and Backup



Create **an Application** across environments **all in one place**



Monitor your application with **Grafana/Prometheus**

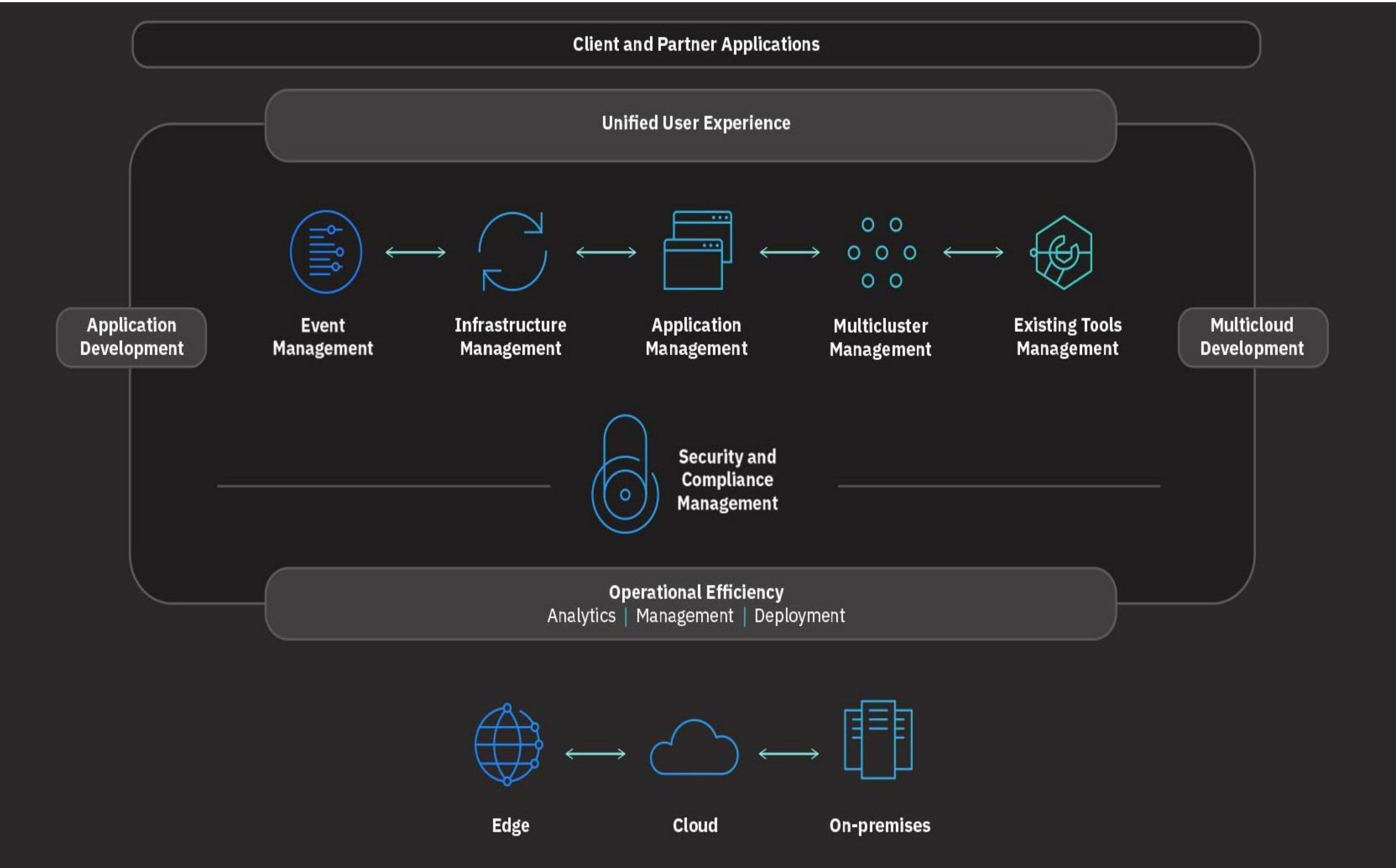

application
guestbook-pam-gbapp
guestbook-pa..-redismaster
guestbook-pam-gbapp
guestbook-pam-gbapp
guestbook-pa..-redismaster
social-dev-1
social-dev-2
web-dev-1
guestbook-pa..p-redisslave
web-dev-1
gbrm 0.1.0
gbrm 0.1.0
gbf 0.1.0
gbrs 0.1.0
redis-master
md-guestbook..m-59b77hb9jn
redis-master
md-guestbook..m-5b54fx89lq
md-guestbook..eb-dev-1-gbf
md-guestbook..eb-dev-1-gbf
md-guestbook..45f57d7shv9b
redis-slave
md-guestbook..b-dev-1-gbrs
md-guestbook..l-dev-1-gbrm
md-guestbook..l-dev-2-gbrm

View all relevant information, deployments and placement policies for your applications

I need help with Application Management

- How do I deploy applications across these environments?
- How do I move workloads across environments?
- How can I backup my applications?
- How do I do Business Continuity?

Introducing
IBM Cloud Pak for Multicloud Management



Client and Partner Applications

Unified User Experience

Application Development

Event Management

Infrastructure Management

Application Management

Multicluster Management

Existing Tools Management

Multicloud Development

Security and Compliance Management

Operational Efficiency

Analytics | Management | Deployment

Edge

Cloud

On-premises

THANKS

Franck Descollonges

IBM Developer Advocate Europe

Twitter : @fdescolpro

Linkedin : https://www.linkedin.com/in/franck-descollonges/