



これからはじめる Red Hat OpenShift

Red Hat K.K.

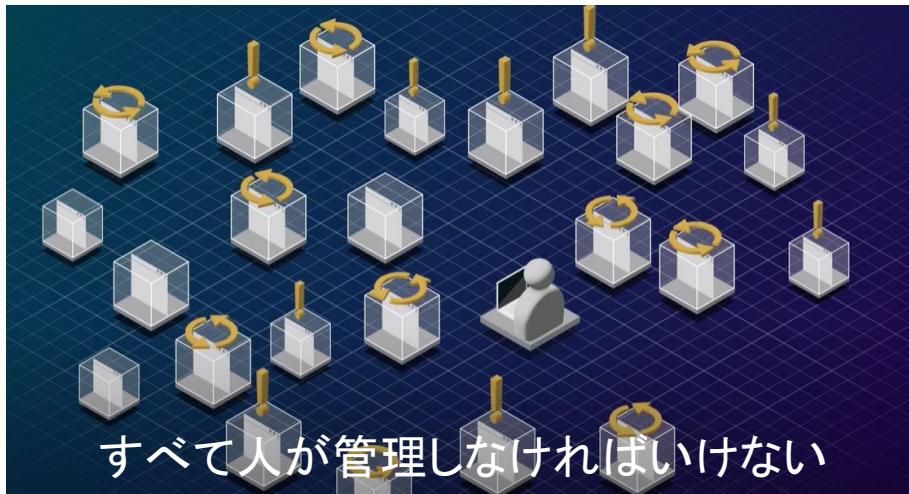
Solution Architect / Sales

OpenShiftとは

Why we require OpenShift?

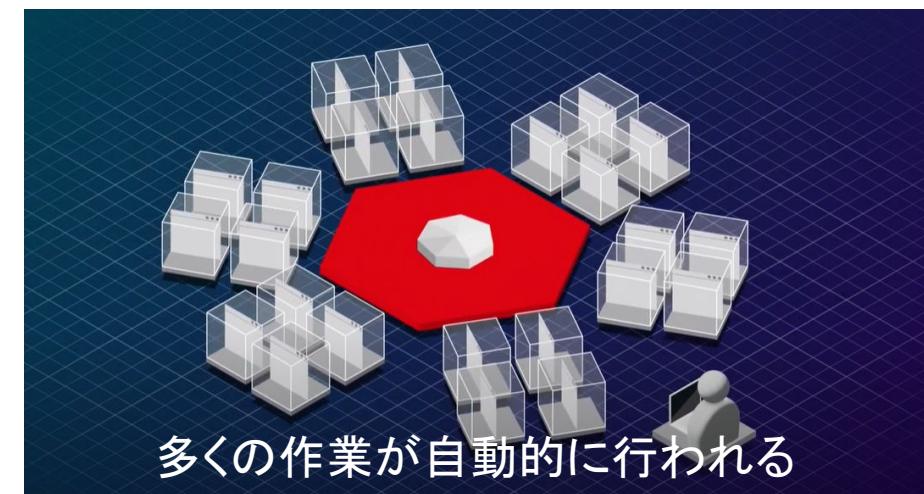
コンテナオーケストレーション

マイクロサービス化によってコンテナの数が増えれば増えるほど、管理が複雑化します。
コンテナオーケストレーションは、**コンテナの管理・運用を自動化**するためのソリューションです。



既存の運用スタイル

- ・属人的な障害復旧オペレーション
- ・手動によるのコンテナ変更作業
- ・アプリケーションごとの設定管理
- ・定期的な監視作業

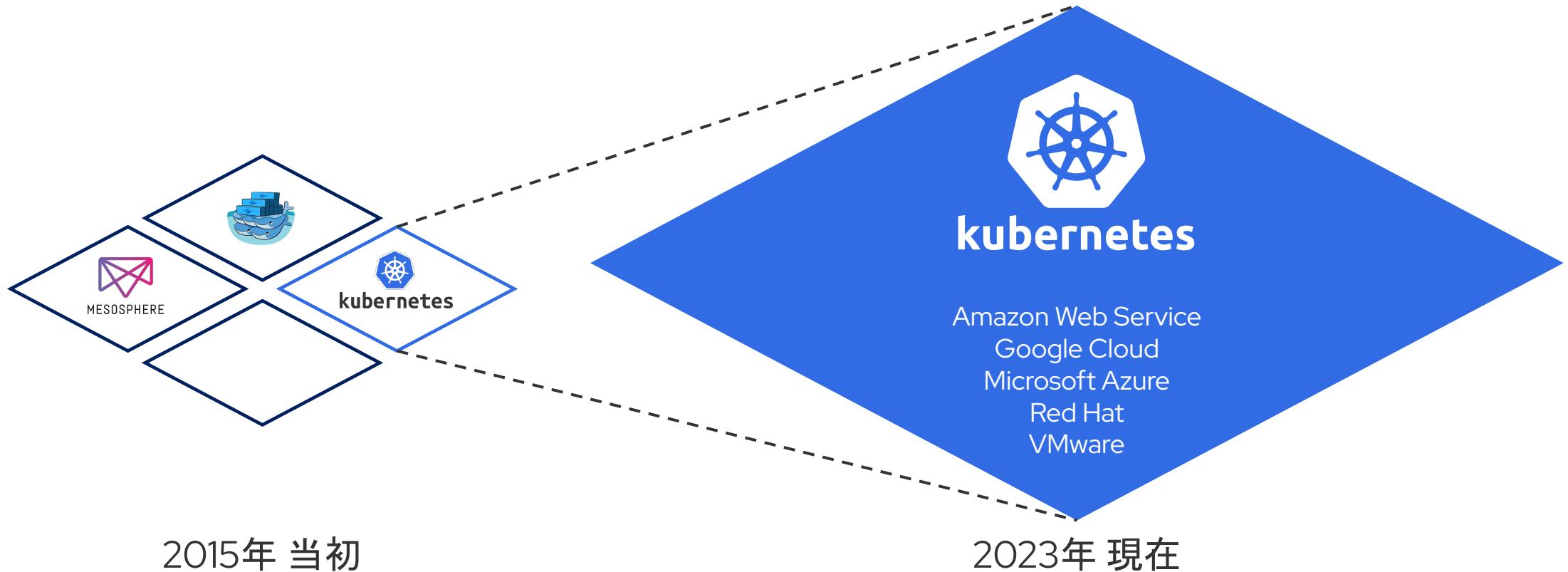


今後の運用スタイル

- ・ビジネス変化に応じた適切なリソース調整

コンテナオーケストレーションの歴史

企業のコンテナオーケストレーションの選択肢として「Docker」と「Mesosphere」は競い合ってきました。その後、Kubernetesが急激に支持を広げ、今では**デファクトスタンダードの地位**を得ています。



Kubernetesができること

Kubernetes(k8s)とは、**コンテナの運用操作を自動化するオープンソースのコンテナオーケストレーション**です。

Kubernetesを使用することにより、コンテナ化されたアプリケーションのデプロイやスケーリングに伴う、運用負担を軽減することができます。



アクセス負荷分散



コンテナの死活監視



リソースの制御



Bare metal



Virtual



Private cloud



Public cloud

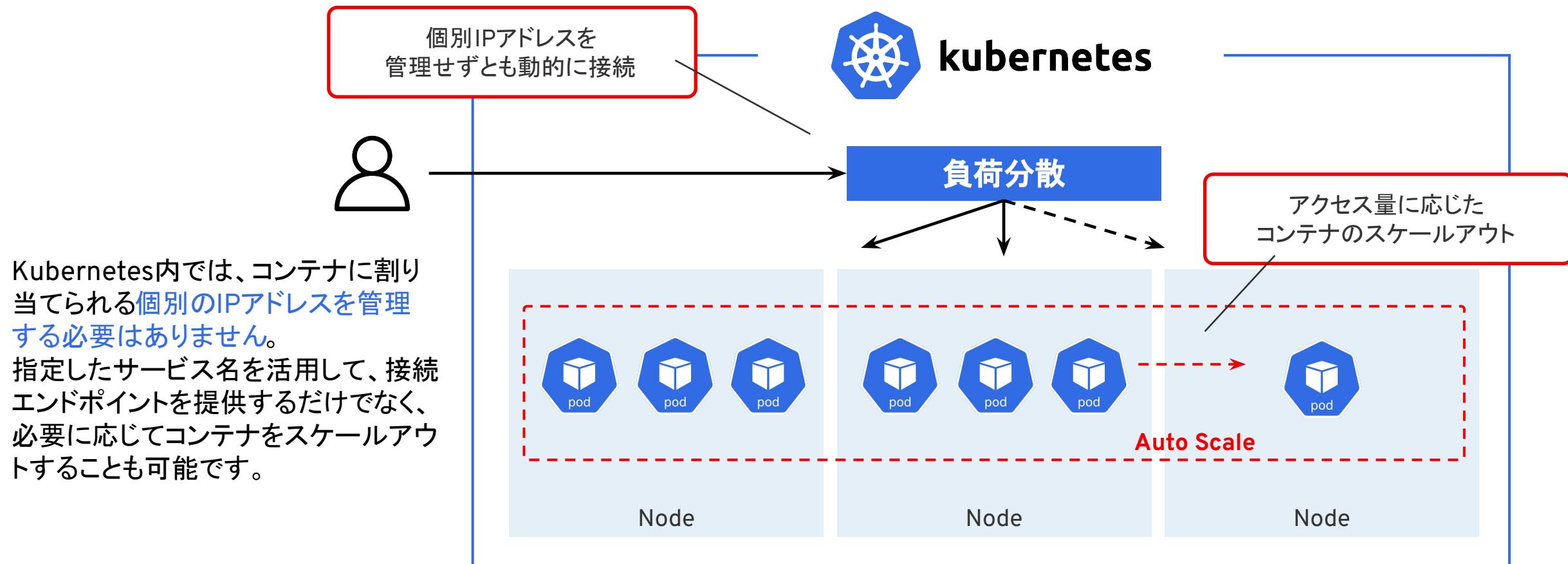


Edge

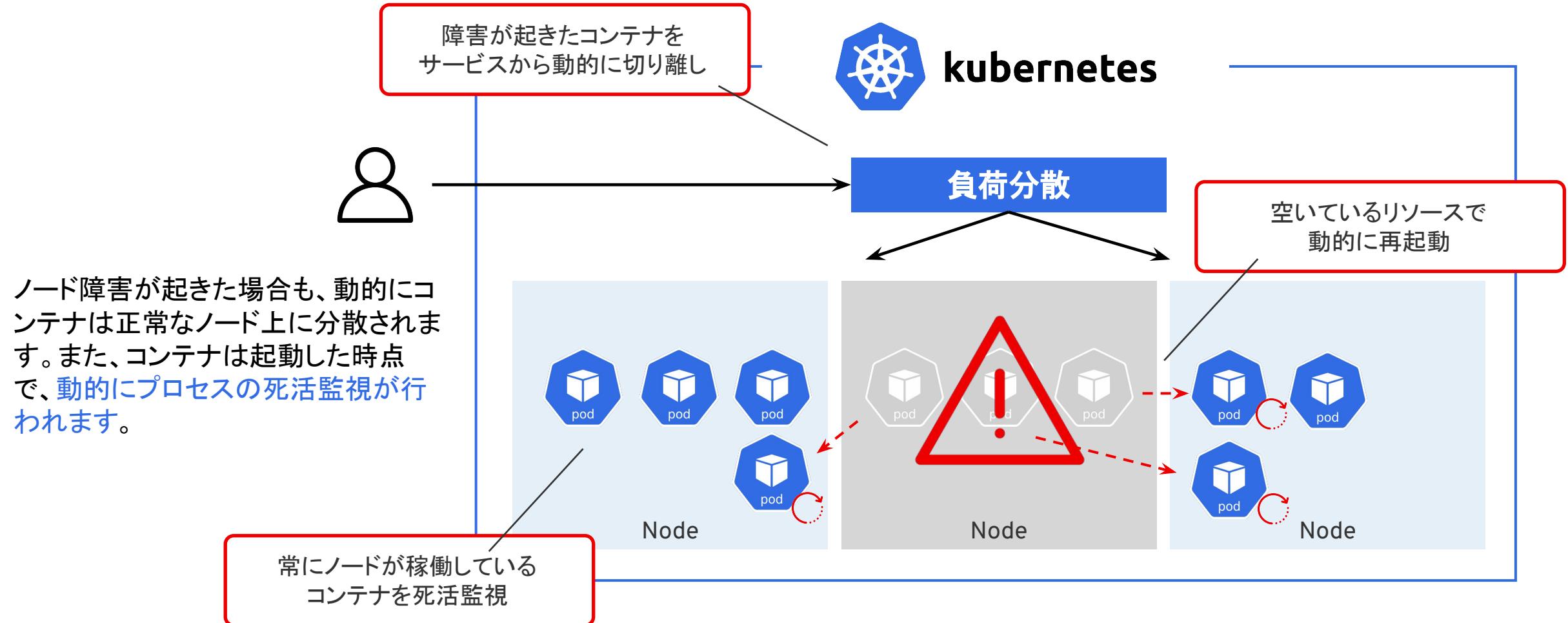


アクセス負荷分散

Kubernetesのアクセス負荷分散



Kubernetesのコンテナの死活監視

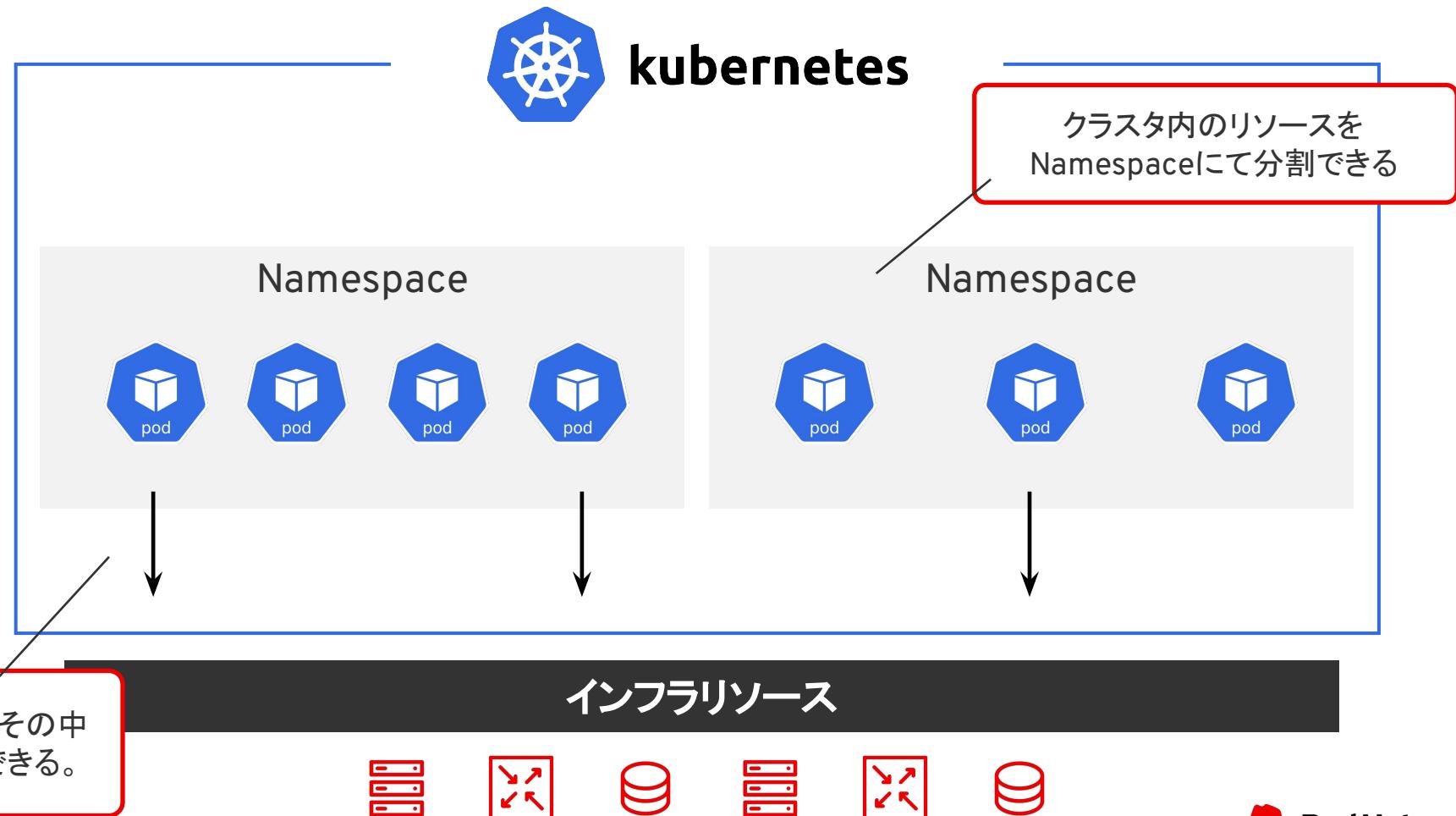




リソースの制御

Kubernetesのインフラリソース配置

個別のインフラリソース(ストレージやネットワーク)に対して、ロールベースの細かな権限管理(RBAC管理)ができます。これによってリソースの払い出し単位を、仮想マシンなどのインスタンス単位ではなく、リソースプールとして受け渡しできます。



Kubernetesだけではできないこと

Kubernetesはコンテナの管理、運用に役立つ機能を提供しますが、それ単体だけではできることもあります。コンテナのビルドやミドルウェアの管理には、Kubernetes以外のツールの連携が必要です。

Kubernetesでは提供されない機能

コンテナの動的
ビルド/デプロイ

ミドルウェア
の管理

クラスタの
ロギングや監視

コンテナの
セキュリティ

クラスタ
アップグレード



kubernetes

Linux



Bare metal



Virtual



Private cloud



Public cloud



Edge

オープンソースや他のクラウドサービスで補完する

ライセンス費用の増加

障害の切り分け責務

ツールごとの保守調達



GKE



AKS
Kubernetes



Azure

EKS



aws



Kubernetes上のサービス管理がビジネスに直結

Agility

開発アジャリティの向上

Steady

安定したサービス維持継続

Optimize

トータルコスト(TCO)の削減

ビジネス価値に直結する機能

コンテナの動的
ビルド/デプロイ

ミドルウェア
の管理

クラスタの
ロギングや監視

コンテナの
セキュリティ

クラスタ
アップグレード



kubernetes

インフラリソース管理に関する運用の自動化

- Kubernetesがデファクトスタンダード -



Bare metal



Virtual



Private cloud



Public cloud



Edge

Red Hat OpenShift

エンタープライズに求められる機能をKubernetesに付随し、サポートすることで、ビジネス価値に直結する機能を提供しています。**アプリケーション開発の効率化に重きを置く**か、まずはインフラ運用の効率化に取り組むか、という点がKubernetes単体と大きく異なる点です。



コンテナの動的
ビルド/デプロイ

ミドルウェア
の管理

クラスタの
ロギングや監視

コンテナの
セキュリティ

クラスタ
アップグレード



Bare metal



Virtual



Private cloud



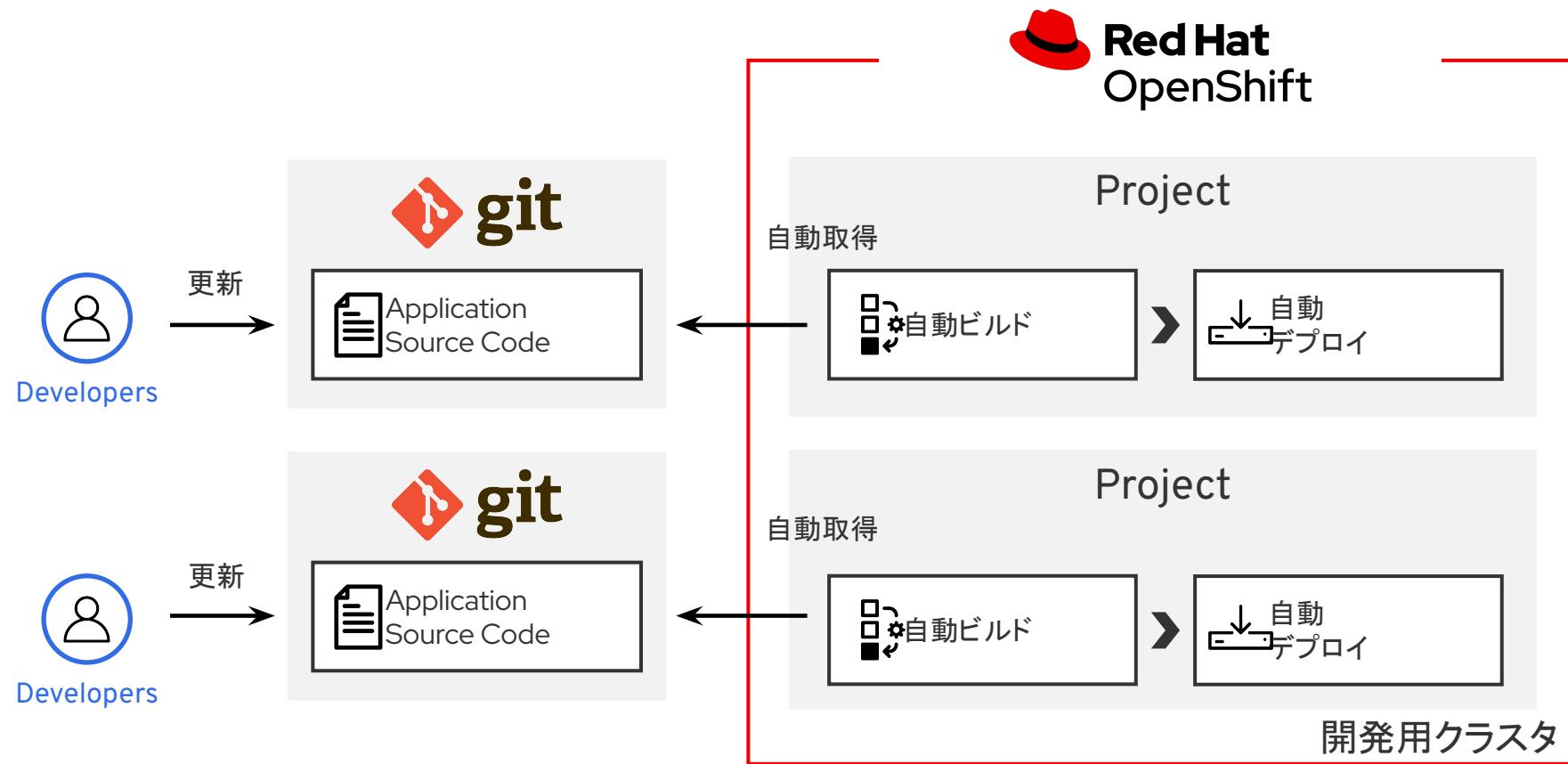
Public cloud



Edge

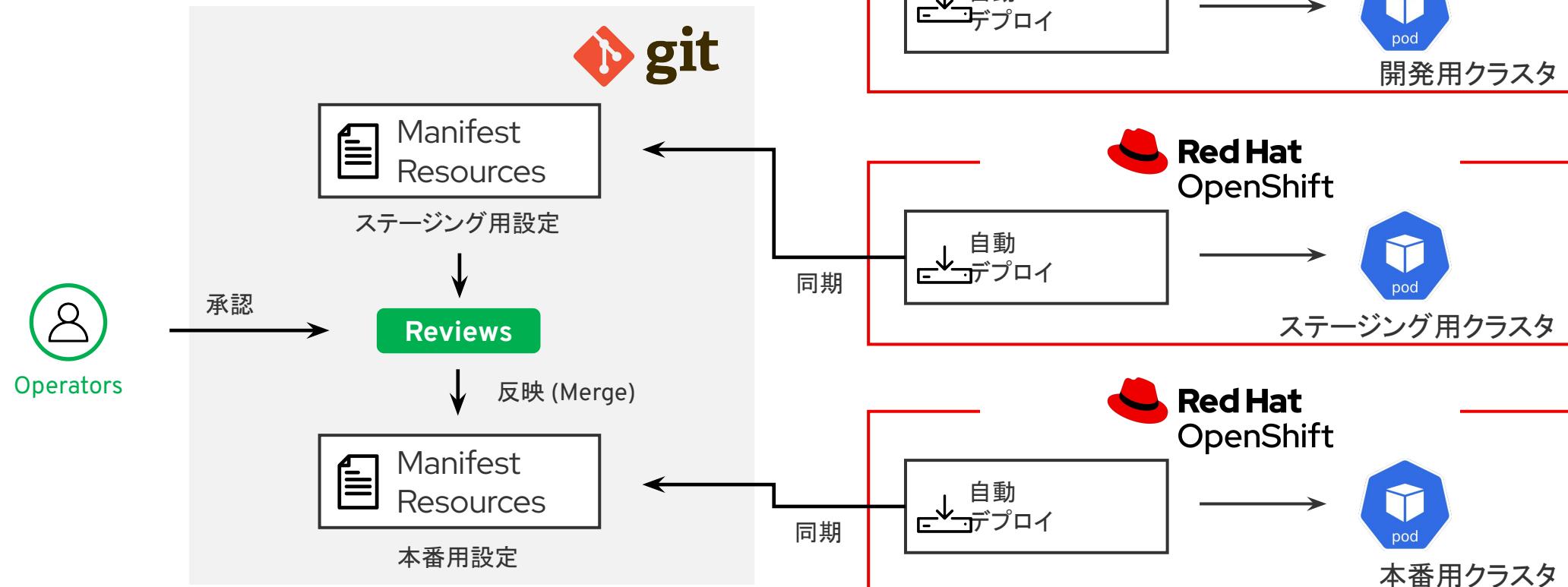
OpenShiftを活用したコンテナ開発

コンテナ開発では、開発者が新しく作ったソースコードをGitリポジトリに格納するだけで、動的にビルド、テスト、デプロイが行われます。従来のように、運用者に開発リソースを個別に用意してもらうことがないため、**開発スピードが飛躍的に向上します。**



OpenShiftを活用した継続的デリバリ

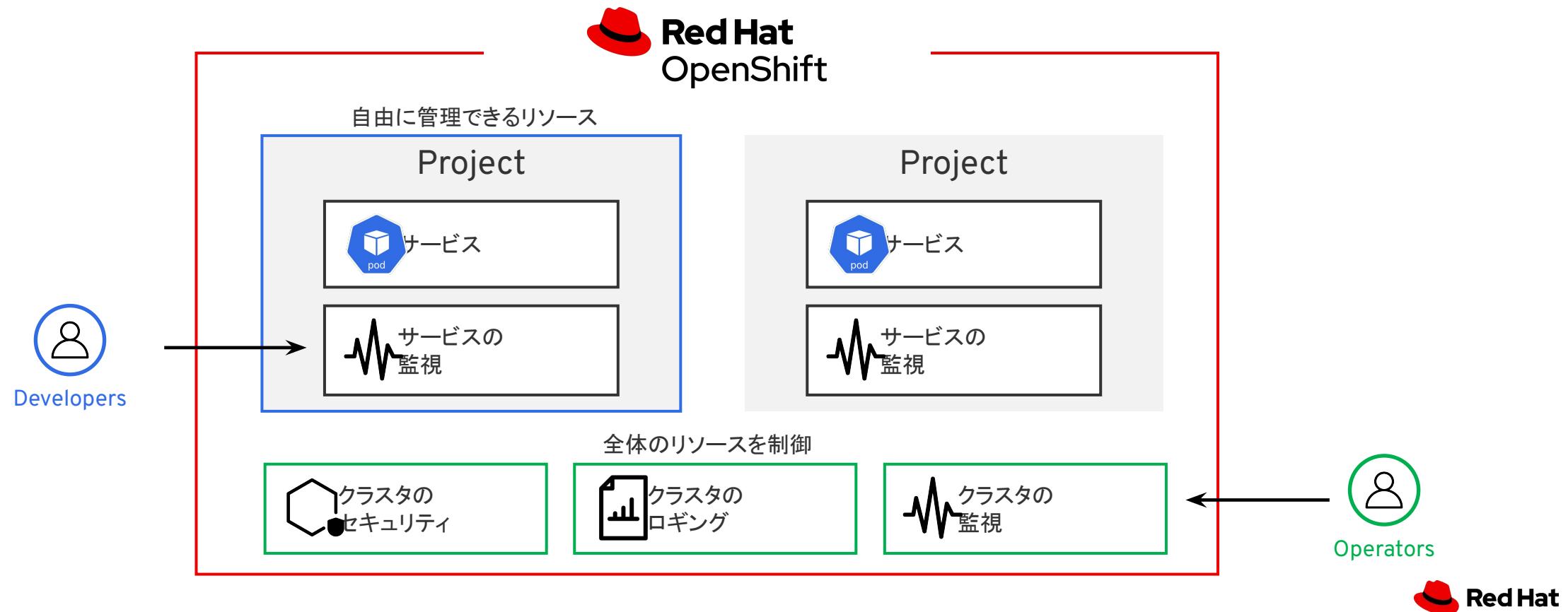
運用者はデプロイ設定内容をコード上(マニフェスト)で確認し、承認ボタンを押すことで動的に各環境に同期します。



従来の手動作業はなくなりますが、人の判断をもとにした安全な動的デプロイが可能です。

OpenShiftを活用したサービスの管理

開発者は自身が関わるサービスの安定性を各チームで見ることができます。運用者はクラスタの管理に専念できます。
従来のようにアプリごとの環境差異や監視設定を運用者が統合する必要がなく、**適材適所にリソースをセキュアに管理できます。**



OpenShiftが導き出すビジネス価値

導入企業14社から年間の平均ビジネス価値を算出

An IDC Business Value White Paper, sponsored by Red Hat

5年間のROI

636%

アプリ開発サイクル

29%迅速化

投資回収期間

10ヶ月

必要なVMの数

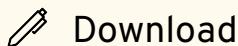
22%削減

開発チームの生産性

20%向上

インフラチームの稼働率

21%向上



Red Hat OpenShift のビジネス価値

<https://www.redhat.com/ja/resources/The-Business-Value-of-Red-Hat-OpenShift>



Red Hat OpenShiftの ビジネス価値

RESEARCH BY:



Nancy Gohring
Research Director, Future of Digital Innovation, IDC



Larry Carvalho
Research Director, Platform as a Service, IDC



Gary Chen
Research Director, Software Defined Compute, IDC



Matthew Marden
Research Director, Business Value Strategy Practice, IDC



OpenShiftの特徴

What does OpenShift provide?

Red Hat OpenShiftの5つの特徴

エンタープライズに求められる機能をKubernetesに付随し、サポートすることで、ビジネス価値に直結する機能を提供しています。

1

コンテナの動的
ビルド/デプロイ

2

ミドルウェアの管理

3

クラスタの
ロギングや監視

4

コンテナの
セキュリティ

5

クラスタアップグレード

Source to Imageビルド

アプリケーションソースコードとベースイメージを動的にビルドする機能(s2i)があるため、**開発者はソースコード開発に専念**できます。



Red Hatから提供されるベースイメージ

Create Sample Application

Name *

java-sample

A unique name given to the component to

Builder Image version *

IST openjdk-11-eI7

1. アプリケーション開発言語のベースイメージを指定



Red Hat OpenJDK 11 (RHEL 7)

BUILDER JAVA OPENJDK

Build and run Java applications using Maven and OpenJDK 11.

Sample repository: <https://github.com/>

2. アプリケーションソースコードが入ったリポジトリを指定

Git repo URL

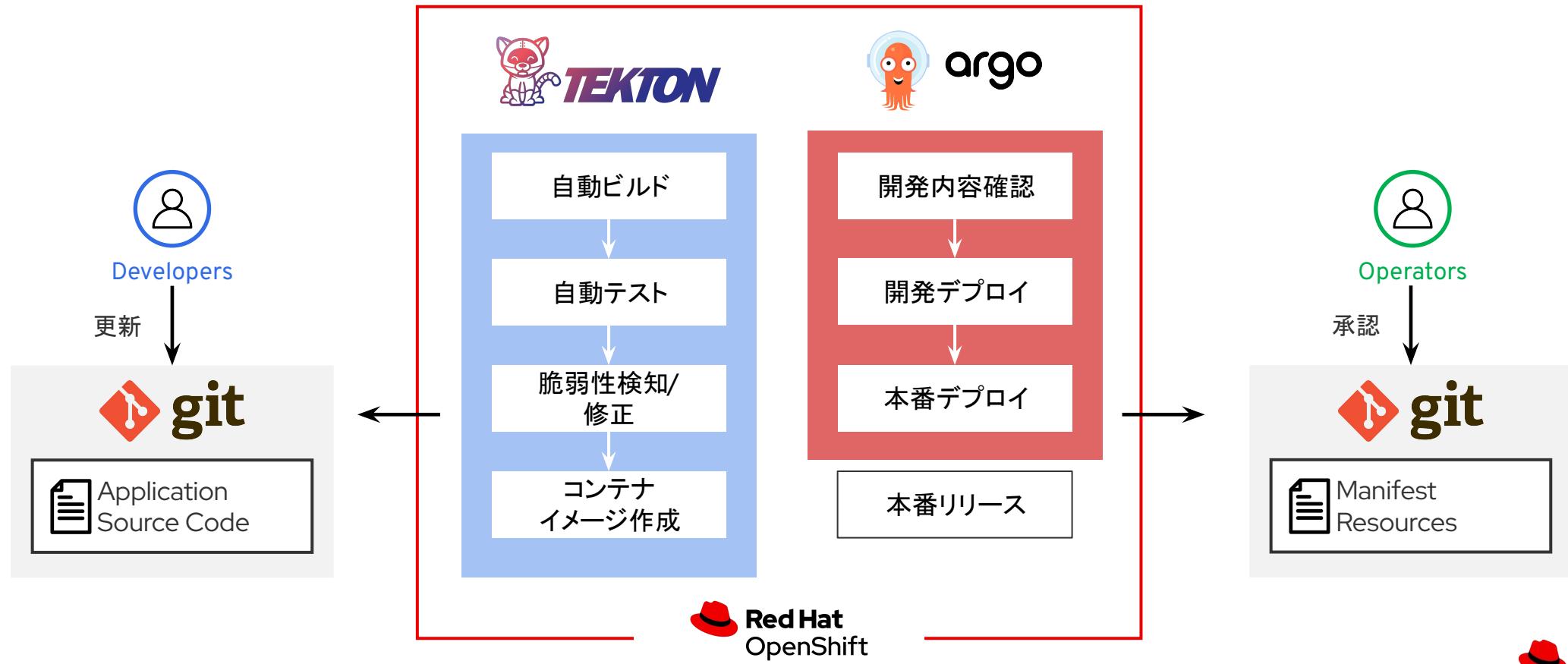
<https://github.com/jboss-openshift/openshift-quickstarts>

Create

Cancel

コンテナ環境に適したCI/CD

OpenShiftでは、開発者がアプリケーションに変更を加えたときの動的ビルドやテスト作業、運用者が確認を行う安全なデプロイ作業をCI/CDパイプラインによって完全自動化します。



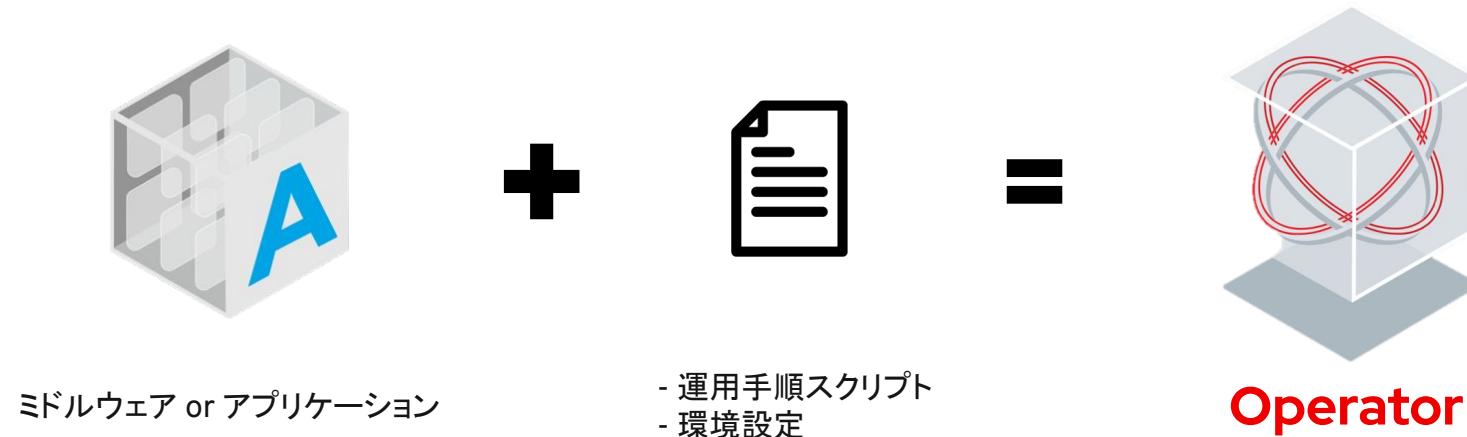
ミドルウェア運用の自動化

OpenShiftでは、ミドルウェアの運用自動化をKubernetes Operator(Operator)で実装しています。

Kubernetes Operator – コンテナがコンテナを管理する時代に-

Operatorとは、Kubernetes上でミドルウェアやアプリケーションを動的にデプロイ、管理するための手段です。

Kubernetesのリソースとコントローラーをベースとしており、運用者に代わってミドルウェアの複雑なインストールや設定、運用、管理作業を代替します。



OperatorHub

OperatorHubによって各種ミドルウェアが提供されます。

Operatorを利用することによって、各ミドルウェアのインストール、設定、監視、アップグレードなどの運用操作を自動化できます。

OpenShiftのサブスクリプションに含まれる
ミドルウェア

- CI Pipelines (Tekton)
- GitOps (Argo)
- Cluster Monitoring (Prometheus)
- Cluster Logging (Loki + Vector)
- Service Mesh (Istio)
- Serverless (Knative)
- Tracing (Jaeger / Kiali)
- API Security (Gatekeeper)
- etc

The screenshot shows the Red Hat OpenShift Container Platform interface. The top navigation bar includes the Red Hat logo, a search bar, and user account information. The left sidebar has a dark theme with white text and features sections for '管理者' (Administrator), 'Home', 'Operators' (which is currently selected), 'Workloads', and 'Networking'. Under 'Operators', there are links for 'OperatorHub' (which is highlighted) and 'Installed Operators'. The main content area is titled 'OperatorHub' and contains a brief introduction about discovering operators from the Kubernetes community and Red Hat partners. It features a grid of cards for different operators:

- Advanced Cluster Management for Kubernetes** (provided by Red Hat): Described as advanced provisioning and management of OpenShift and Kubernetes clusters.
- Advanced Cluster Security for Kubernetes** (provided by Red Hat): Described as the Red Hat Advanced Cluster Security (RHACS) operator provisions the services necessary...
- Ansible Automation Platform (early access)** (provided by Red Hat): Described as the Ansible Automation Platform Resource Operator manages everything Automation.

Operatorの種類

ミドルウェアを自動管理するOperatorには、以下の3種類があり、各プロダクトごとにサポートが異なります。

種類	概要
Red Hat Operator	Red Hat によってパッケージ化され、出荷されるRed Hat 製品。 Red Hat によってサポートされます。
認定 Operator	大手独立系ソフトウェアベンダーが提供する製品。Red Hat は ISV とのパートナーシップにより、パッケージ化および出荷を行います。 ISVによってサポートされます。
コミュニティーOperator	<u>operator-framework/community-operators</u> GitHub リポジトリ上の各プロジェクトによってメンテナンスされるソフトウェア。 正式なサポートはありません。

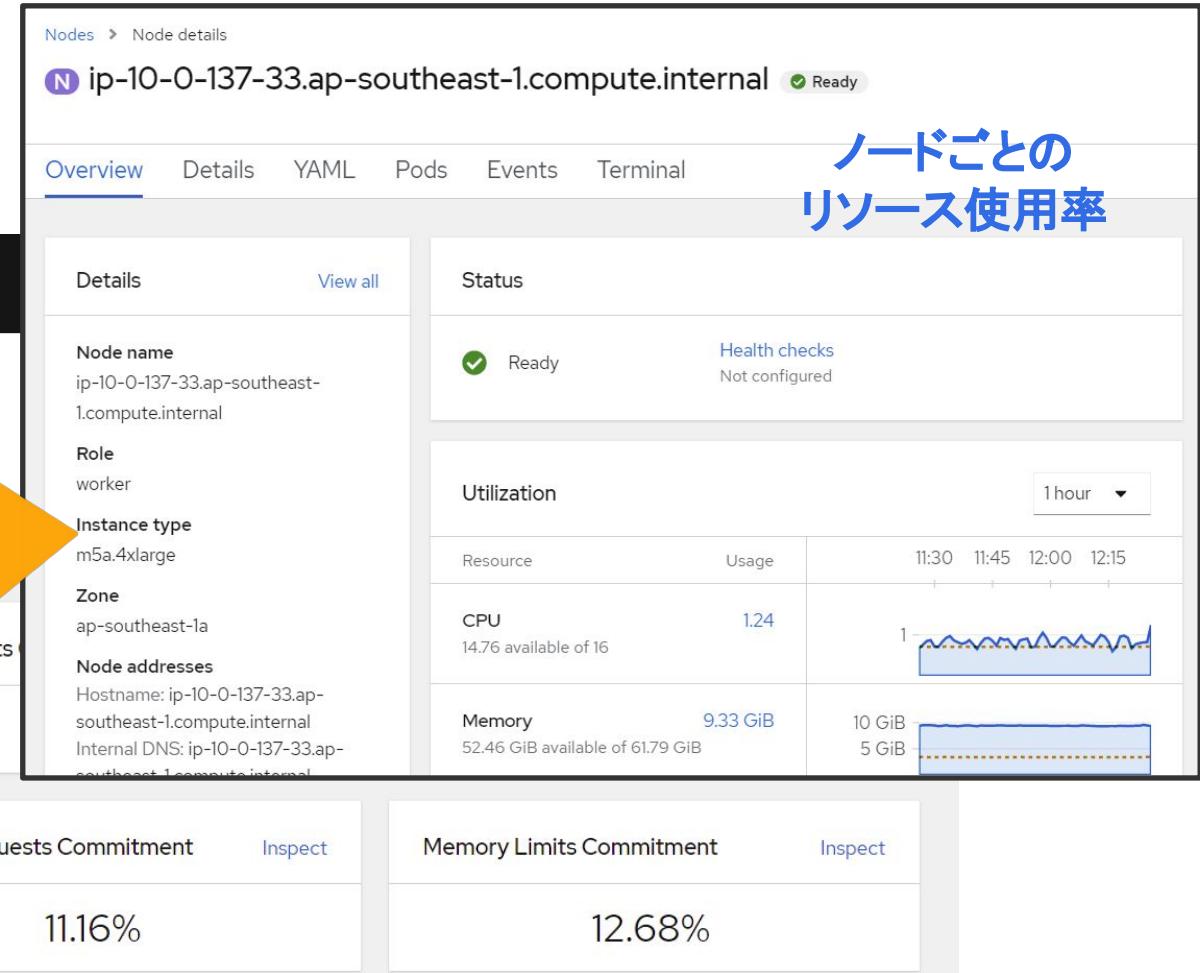
Cluster Monitoring

OpenShiftをインストールした時点で、クラスタに関する監視は Cluster Monitoring(Prometheus)によって設定済みです。(アラート含む)

初期設定や更新作業は自動で設定されます。

The screenshot shows the Red Hat OpenShift Container Platform dashboard. On the left, there's a navigation sidebar with options like Operators, Workloads, Networking, Storage, Builds, Monitoring (which is expanded to show Alerting and Metrics), and Dashboards. The main area is titled "Dashboards" and "Cluster". It displays four cards: "CPU Utilisation" (8.02%), "Memory Utilisation" (18.82%), "CPU Requests" (highlighted with a red arrow), and "Memory Requests Commitment" (11.16%). A large blue box highlights the "CPU Requests" card.

クラスタ全体のリソース使用率



Cluster Logging

Cluster Loggingを有効化することによって、各Pod、Clusterから出力されるログをLoki上に保管できます。保存されたログはApp、Infra、Auditという単位で分類されコンソールから参照することができます。

Logs

Date ↓ Message

- > 2023年2月3日 18:30:10.719 2023-02-03T09:30:10.688Z|02980|ovsdb_server|INFO|memory trimming after compaction enabled.
- > 2023年2月3日 18:30:10.706 ovs|02624|connmgr|INFO|br-int<->unix#2: 11 flow_mods in the 5 s starting 10 s ago (6 adds, 5 deletes)
- > 2023年2月3日 18:30:10.652 I0203 09:30:10.623369 1 job_controller.go:510] enqueueing job openshift-monitoring/osd-rebalance-infra-nodes-27923610
- > 2023年2月3日 18:30:10.584 2023-02-03T09:30:00.691Z|02979|ovsdb_server|INFO|memory trimming after compaction enabled.
- > 2023年2月3日 18:30:10.543 ovs|02526|connmgr|INFO|br-int<->unix#2: 73 flow_mods in the 5 s starting 10 s ago (37 adds, 36 deletes)
- > 2023年2月3日 18:30:10.541 ovs|02479|connmgr|INFO|br-int<->unix#2: 7 flow_mods in the 5 s starting 10 s ago (4 adds, 3 deletes)
- > 2023年2月3日 18:30:10.506 ovs|02745|connmgr|INFO|br-int<->unix#2: 7 flow_mods in the 5 s starting 10 s ago (4 adds, 3 deletes)
- > 2023年2月3日 18:30:10.502 ovs|02490|connmgr|INFO|br-int<->unix#2: 7 flow_mods in the 5 s starting 10 s ago (4 adds, 3 deletes)
- > 2023年2月3日 18:30:10.470 ovs|02978|connmgr|INFO|br-int<->unix#2: 7 flow_mods in the 5 s starting 10 s ago (4 adds, 3 deletes)
- > 2023年2月3日 18:30:10.66 I0203 09:30:10.036310 1 logr.go:249] dispatcher "msg"="Handling request" "request"="/regularuser-validation?timeout=2s"
- > 2023年2月3日 18:30:09.999 I0203 09:30:09.616908 1 job_controller.go:510] enqueueing job openshift-monitoring/osd-rebalance-infra-nodes-27923610
- > 2023年2月3日 18:30:09.934 I0203 09:30:09.921322 1 logr.go:249] dispatcher "msg"="Handling request" "request"="/regularuser-validation?timeout=2s"
- > 2023年2月3日 18:30:09.867 I0203 09:30:09.835173 1 logr.go:249] dispatcher "msg"="Handling request" "request"="/regularuser-validation?timeout=2s"

Pods > Pod details

P java-sample-1-build Completed

Actions ▾

Details Metrics YAML Environment Logs Events Terminal

Log stream ended. sti-build Current log ▾

143 lines

Show in Kibana | Wrap lines | Raw | Download | Expand

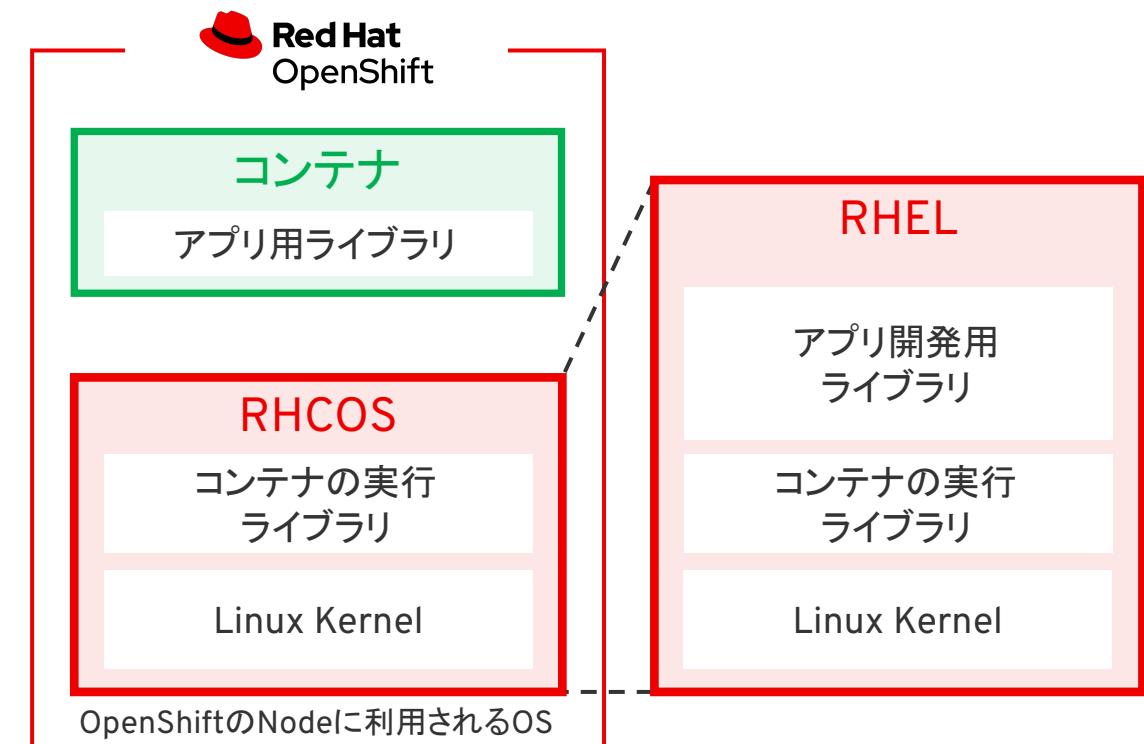
```
org/maven2/org/sonatype/sisu/sisu-inject-bean/1.4.2/sisu-inject-bean-1.4.2.jar (153 kB at 362 kB/s)
org/maven2/org/apache/maven/maven-repository-metadata/3.0/maven-repository-metadata-3.0.jar
org/maven2/org/sonatype/sisu/sisu-guice/2.1.7/sisu-guice-2.1.7-noaop.jar (472 kB at 812 kB/s)
org/maven2/org/apache/maven/maven-model-builder/3.0/maven-model-builder-3.0.jar
org/maven2/org/apache/maven/maven-settings/3.0/maven-settings-3.0.jar (47 kB at 62 kB/s)
org/maven2/org/apache/maven/maven-aether-provider/3.0/maven-aether-provider-3.0.jar
org/maven2/org/apache/maven/maven-repository-metadata/3.0/maven-repository-metadata-3.0.jar (30 kB)
org/maven2/org/sonatype/aether/aether-impl/1.7/aether-impl-1.7.jar
org/maven2/org/apache/maven/maven-settings-builder/3.0/maven-settings-builder-3.0.jar (38 kB at 46 kB/s)
org/maven2/org/sonatype/aether/aether-spi/1.7/aether-spi-1.7.jar
```

コンテナ基盤を支えるRHCOS

コンテナ利用に最適化されたRHCOS(Red Hat Enterprise Linux CoreOS)を利用することによって、より安全かつ安定したコンテナ環境を提供します。

RHCOSは、RHELのKernelを利用しコンテナ実行に必要なライブラリだけを載せたコンテナ専用軽量OSです。
従来のRHELと同等の利用でサポートされます。

- OpenShiftと連携し、動的なUpgradeをOne-Clickで実現
- ライブラリが少ないため、セキュリティホールを生む可能性が極めて低い
- ライブラリがないため、多くのプログラムがOSの中で動かすことができない



RHELによるコンテナの信頼性

コンテナイメージとして展開されるUBI(Universal Based Image)は、RHEL(Red Hat Enterprise Linux)のライフサイクルに基づいてサポートされます。

Universal Base Image

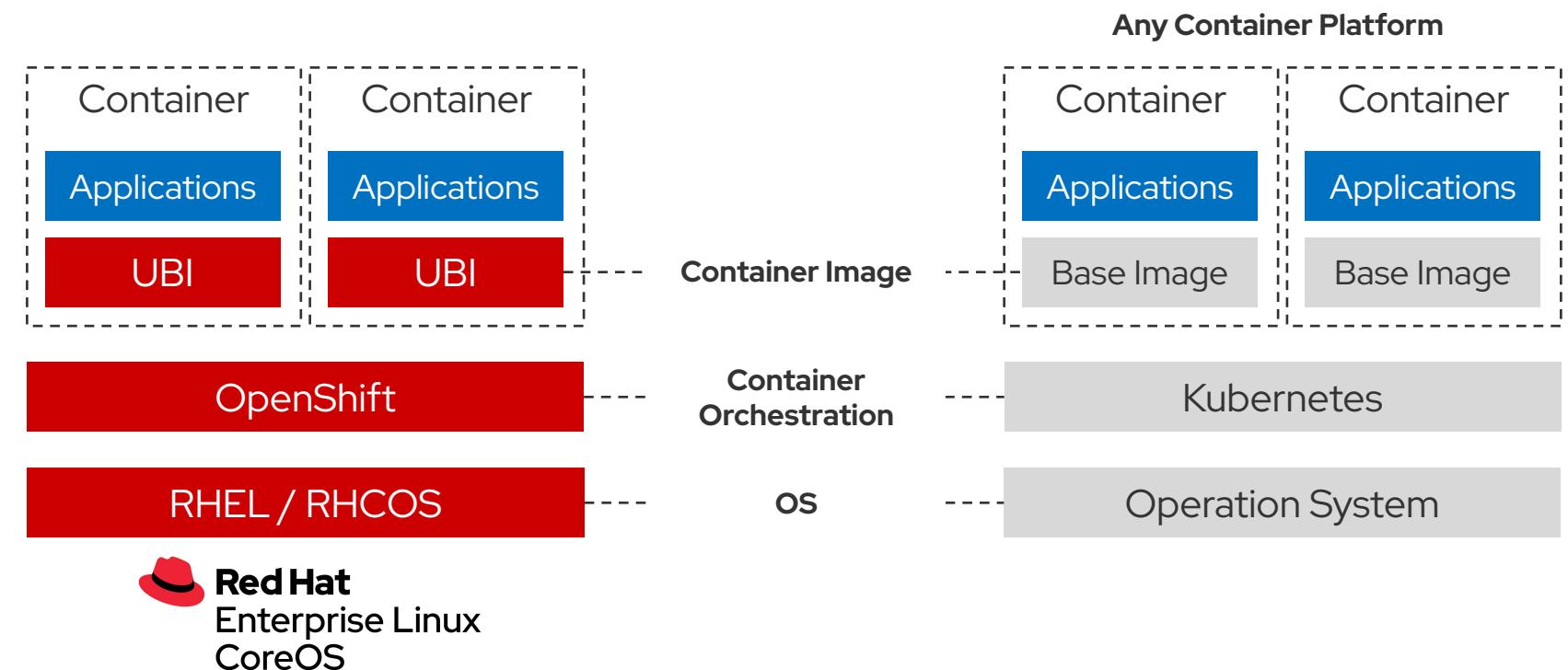
Red Hatのコンテナ実行環境を利用する場合、UBIの使用を完全にサポート

OpenShift

ホストOSであるRHEL/RHCOSのサポートを含む

RHEL/RHCOS

コンテナランタイムとしての稼働をサポート



ベースイメージの提供

コンテナベースイメージ(アプリケーションランタイム / SDK)をRed Hat Ecosystem Catalogにて提供しています。
セキュリティ脆弱性診断にも対応しており、**安心してコンテナイメージを利用可能です。**

The screenshot shows the Red Hat Ecosystem Catalog interface. On the left, a search bar contains "python". Below it, filters for "Provider" (Couchbase, IBM, New Relic, Red Hat, Inc.) and "Category" (Application Delivery) are shown. The main area displays search results for "python", including "Red Hat Python 2.7" and "Red Hat Python 3.8". A large orange arrow points from the Python 3.8 card to its detailed view on the right.

Container images
Container images offer lightweight and self-contained software to enable deployment at scale.

Home > Software > Container images

python X Search

Provider

- Couchbase
- IBM
- New Relic
- Red Hat, Inc.

Category

- Application Delivery

Sort by: Relevance

Red Hat Python 2.7
by Red Hat, Inc.
Platform for building and running Python 2.7 applications
Updated 20 days ago

Red Hat Python 3.8
by Red Hat, Inc.
Platform for building and running Python 3.8 applications
Updated 21 days ago

Red Hat Python 3.8
latest 2.7 2.7-155
Overview Security Technical Information Packages Dockerfile Get this image

Health index i
This image does not have any unapplied Critical or Important security updates.
The Container Health Index analysis is based on RPM packages signed and created by Red Hat, and does not grade other software that may be included in a container image.

Release category i Generally Available
Advisory i RHBA-2021:3113
Privilege mode i Unprivileged

OpenShiftサブスクリプションに含まれるベースイメージ

「Software Collections(for RHEL7)」および「Application Streams (for RHEL8/9)」のコンテナイメージのサポートがOpenShiftのサブスクリプションに含まれています。

Red Hat Enterprise Linux 7 Software Collections Product Life Cycle



<https://access.redhat.com/ja/node/4654951>

Red Hat Enterprise Linux 8/9 Application Streams



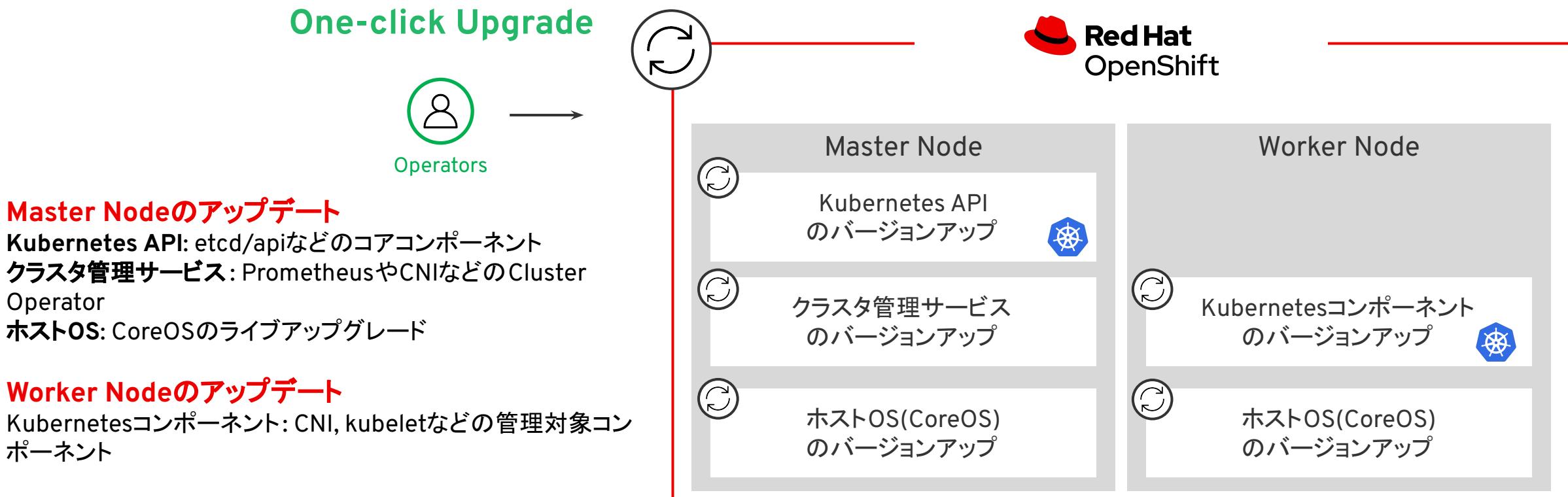
<https://access.redhat.com/ja/node/4167391>

Application Streamの一部抜粋

Application Stream	Release Date	Retirement Date
mariadb 10.5	May 2021	May 2026
postgresql 13	May 2021	May 2026
python 3.9	May 2021	May 2024
redis 6	May 2021	May 2024
dotnet 6.0	Nov 2021	Nov 2024
nginx 1.20	Nov 2021	Nov 2023
perl 5.32	May 2022	Apr 2025
php 7.4	Nov 2020	May 2029

OpenShiftのバージョンアップ

ワンクリックでOpenShiftクラスタをアップデートする「Over-the-Air (OTA) Update」機能を提供しており、コンポーネントごとの管理が不要です。



Master Nodeのアップデート

Kubernetes API: etcd/apiなどのコアコンポーネント

クラスタ管理サービス: PrometheusやCNIなどのCluster Operator

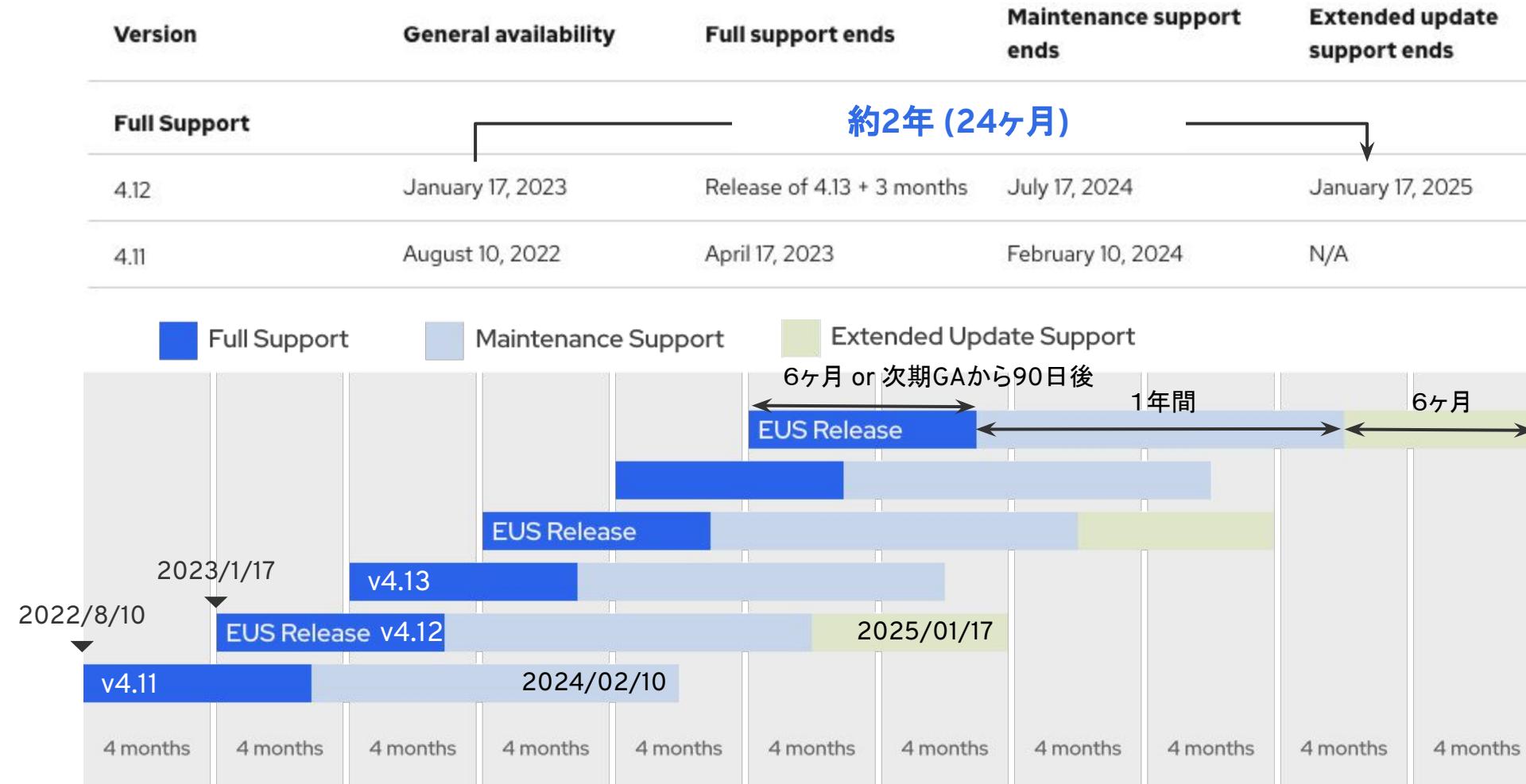
ホストOS: CoreOSのライブアップグレード

Worker Nodeのアップデート

Kubernetesコンポーネント: CNI, kubeletなどの管理対象コンポーネント

OpenShiftのサポートライフサイクル

Red Hat OpenShiftはKubernetesプロジェクト同様に長期のサポートライフサイクルで提供されています。



OpenShiftの活用

How can we utilize OpenShift?

Kubernetesのコンポーネント



Master Node (3 nodes)

Master Nodesには、コンテナを制御するコンポーネントと、クラスタの状態を構成管理するデータ(etcd)があります。

Controllers: リソースのデプロイを行う機能

Schedulers: リソースの空き状況を管理する機能

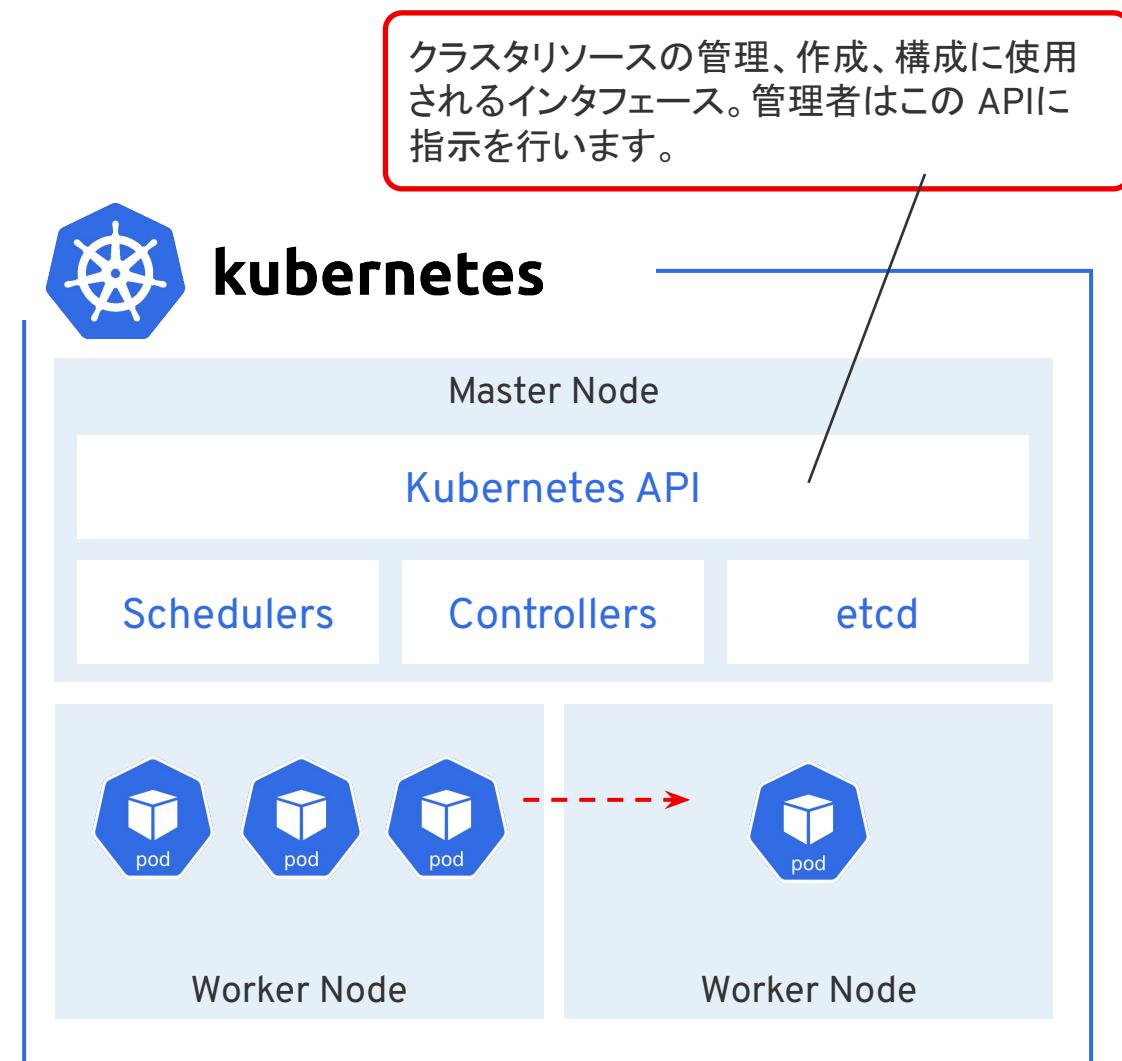


Worker Node (2~N nodes)

Worker Nodesでは、スケジュールされたコンテナがデプロイされ、コンテナを死活監視します。

リソース容量が足りない場合は、Worker Nodesを追加します。

クラスタリソースの管理、作成、構成に使用されるインターフェース。管理者はこの APIに指示を行います。



OpenShiftのコンポーネント

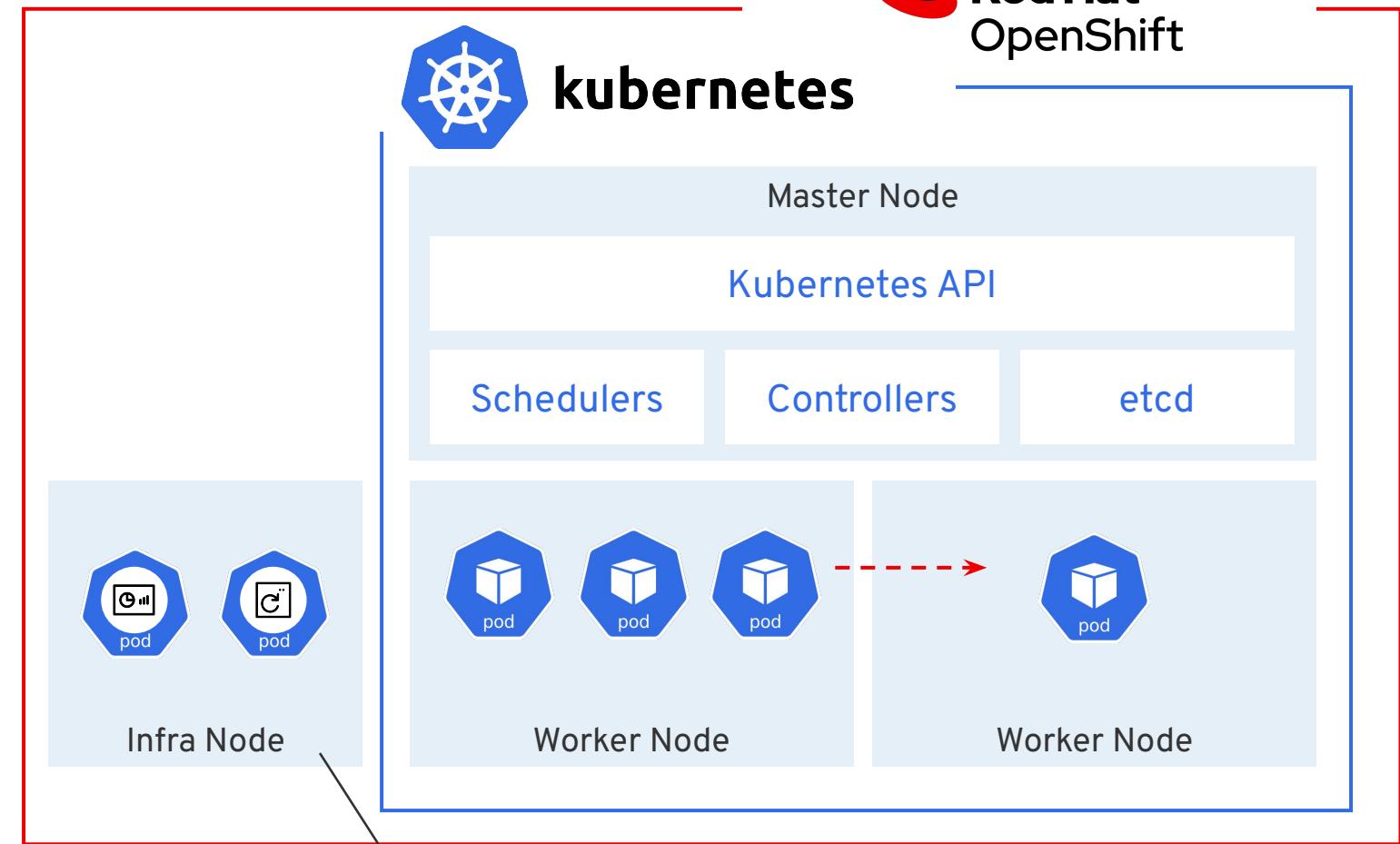
Infra Node (3 nodes)

Infra Nodesには、OpenShiftの追加サービスを管理するためのコンポーネントが配置されます。

Infra Nodesに配置できる機能

- Router
- OpenShift-included Registry
- OpenShift cluster monitoring
- OpenShift log aggregation

など



* 設定上はWorker Nodesにラベルが付いたものです。

OpenShiftのSubsでサポートする機能

機能	OpenShiftの機能名	概要
Embedded OS	Red Hat CoreOS	コンテナ専用のセキュアな軽量OS、動的にバージョンアップが可能 Kubernetesに最適なコンテナランタイム(CRI-O)を活用
Web Portal		クラスタや各Projectの管理を行うためのWeb UI
Virtualization	OpenShift Virtualization	kubevirtを利用した仮想マシンのコンテナ化支援
CI	OpenShift Pipelines	Tekton Pipelines, Tekton Triggersによる継続的インテグレーション
CD	OpenShift GitOps	Argo CDを活用したGitOpsベースの継続的デリバリ
Ingress	HAProxy Ingress Controller	Ingressコントローラ
Service Mesh	OpenShift Service Mesh	Istio (Maistra), Kiali, Jaegerを活用したサービスメッシュ
Serverless	OpenShift Serverless	Knativeベースのサーバーレス
API Security	Gatekeeper Operator	OPAを使用したAPIリクエスト検証用のアドミッションコントローラ

* OpenShiftのSubscriptionでサポートできるもののおよそ半分をしています。



Feature summary : https://docs.openshift.com/container-platform/latest/welcome/oke_about.html

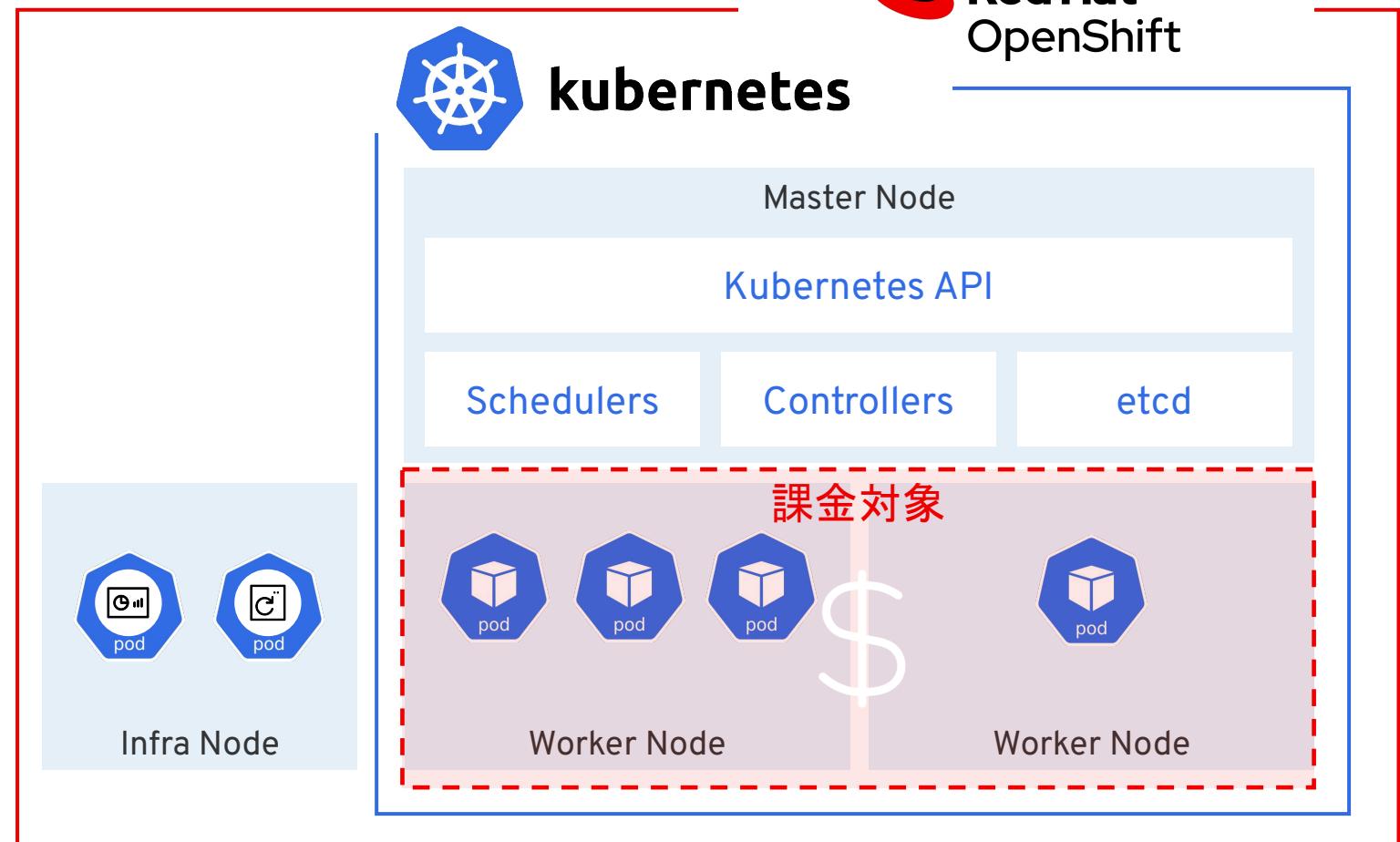
OpenShiftの課金対象

OpenShiftのサブスクリプションは、「Worker Nodes」のvCPU数で課金されます。

ex) 3 Nodes * 12vCPU = 36vCPU

Infra Nodesを作成することによって、負荷の高いワークロードを分離できます。

- 1) サブスクリプション数に対する請求コストの発生を防ぐ
- 2) 保守対象を分離する



OpenShiftをインストールできる環境

専用のインストーラーを利用してOpenShiftクラスタを構築する場合、以下のプラットフォームにインストールできます。



On Public Cloud

- Amazon Web Services (AWS)
- VMware Cloud on AWS
- Microsoft Azure
- Alibaba Cloud
- Google Cloud Platform (GCP)

** +Platform agnostic



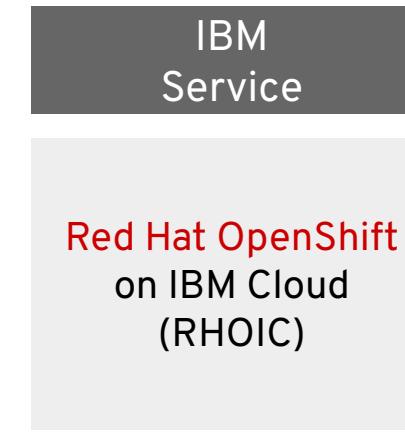
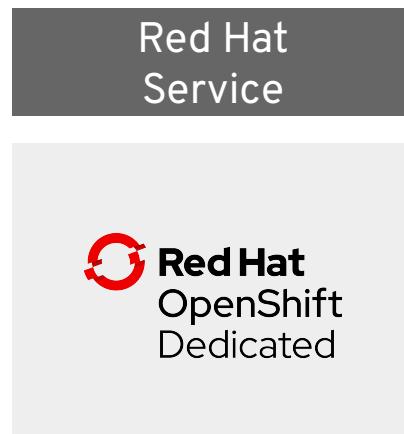
On Your Datacenter

- BareMetal
- VMware vSphere
- Red Hat Virtualization (RHV)
- Red Hat OpenStack Platform (RHOSP)
- IBM Z, Power Systems
- Azure Stack Hub
- Nutanix AOS

インストーラーを利用してインストールすることによって、クラウドのAPIを活用して、OpenShiftクラスタに必要なリソースを動的に作成します。

マネージドOpenShiftの活用

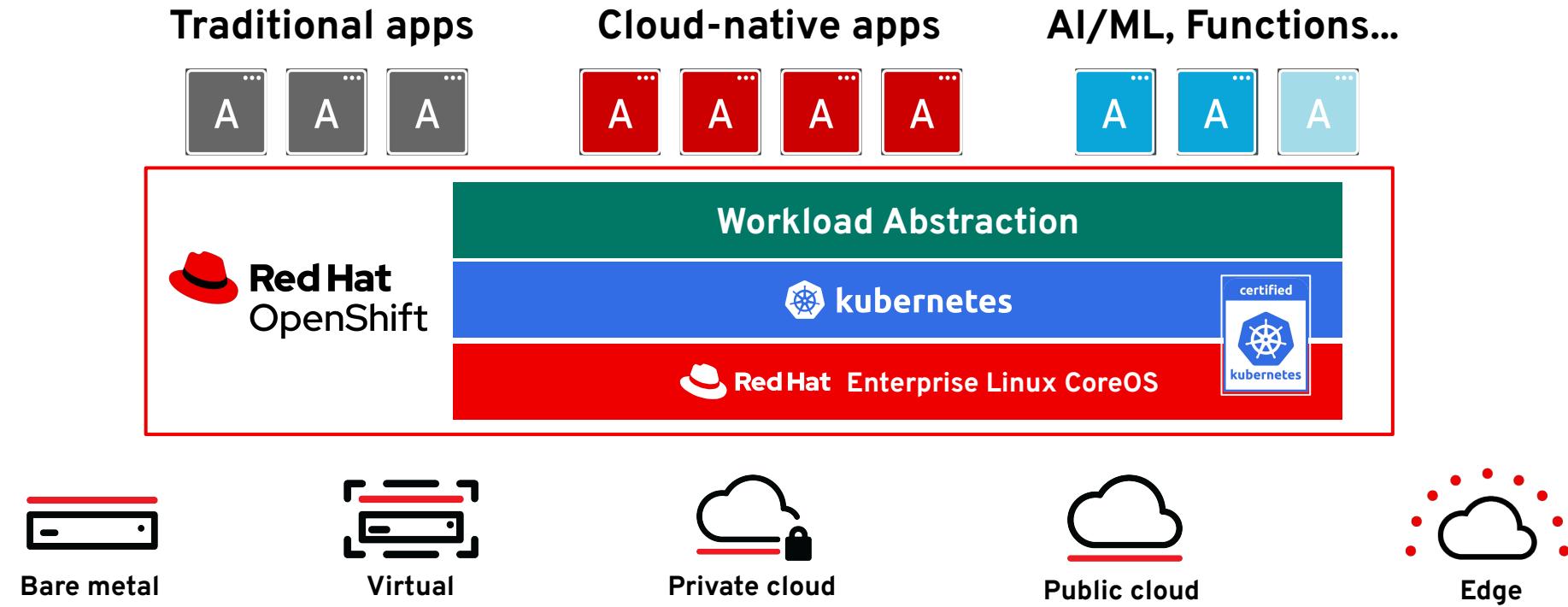
マネージドOpenShiftを利用することによって、OpenShiftのクラスタ運用をパブリッククラウドベンダーにオフロードできます。すでにクラウドを活用している開発者が、使い慣れたクラウド上でOpenShiftを試したいという要望が増えています。



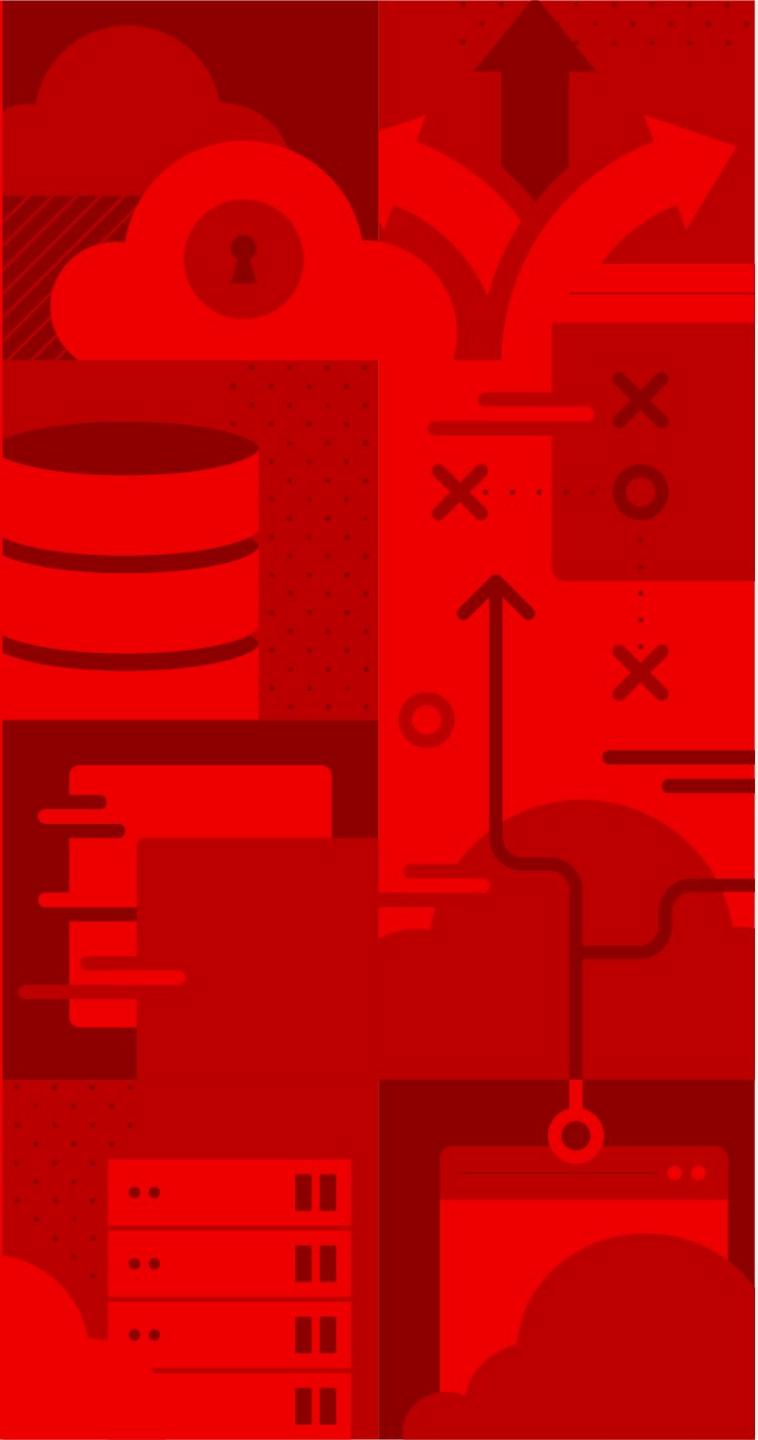
まとめ

Summary

OpenShiftが目指す世界



環境を意識せずに、ビジネス価値を提供できる世界



Thank you

Red Hat is the world's leading provider of enterprise open source software solutions. Award-winning support, training, and consulting services make Red Hat a trusted adviser to the Fortune 500.



[linkedin.com/company/red-hat](https://www.linkedin.com/company/red-hat)



[facebook.com/redhatinc](https://www.facebook.com/redhatinc)



[youtube.com/user/RedHatVideos](https://www.youtube.com/user/RedHatVideos)



twitter.com/RedHat

Appendix A

Container / OpenShift ハンズオン環境のご紹介

本日の101でご紹介した内容を実際に試すことができるハンズオン環境を準備しています。

①ハンズオンのまとめページへアクセス

OpenShiftハンズオンへようこそ

目的

- Container（Docker）の操作を体験します
- OpenShiftの超初級的内容を体験します

ハンズオンに必要な環境

- ブラウザ（Firefox, Chrome）

コース数と難易度

- Container編：2コース ★☆☆☆☆
- OpenShift編：6コース ★☆☆☆☆

難易度	
とても難しい	★★☆☆☆
難しい	★★☆☆☆
普通	★★★☆☆
少し難しい	★★★★☆
難しい	★★★★★

<https://github.com/loungeplus/moku2-public/blob/main/README.md>

コンテナやOpenShiftについて難易度ごとに複数のコンテンツを準備しています。

②コンテンツを選択しハンズオン実施

ターミナル画面

Getting Started a Container ハンズオンの説明

この演習環境は下図のようになっています。図の右側の「コンテナを稼働させるホスト（自ホスト）」が、このコースの左側に表示されているTerminalに対応しています。

コンテナレジストリ コンテナを稼働させるホスト（自ホスト）

コンテナレジストリ

コンテナを稼働させるホスト（自ホスト）

停止中のコンテナ

イメージ格納領域

ランタイム

この環境は既にDockerが利用できる状態になっていますので、Dockerを利用してコンテナを起動して、動作を確認してみましょう。

↓このように表示されている部分をクリックするとコマンドがコピーされるので、左側のターミナルにペーストして実行してください。

↓以下を実行してください

Next

コースにより環境の立ち上げに最大20分程度かかります。起動したら説明に従いハンズオンを実施下さい。

*: 本ハンズオン環境はハンズオン以外の目的で使用することを想定していません。PoC等の目的でご利用することはお控え下さい。

**: ハンズオンコンテンツの内容については予告無く変更（公開の中止を含む）することがあります。あらかじめご了承下さい。