

# OpenShift Virtualization workshop

～ 今から始めるシステム基盤のモダナイゼーション ～

OpenShift Virtualization の基礎に始まり、別環境からの仮想マシンの移行作業までを触って体験！

# 本日のスケジュール

13:00 - 13:10	オープニング
13:10 - 13:30	<b>Lecture 1</b> <ul style="list-style-type: none"><li>- OpenShift Virtualization の基礎</li><li>- 仮想マシンの移行</li></ul>
13:30 - 14:30	<b>Lab 1</b> <ul style="list-style-type: none"><li>- OpenShift Virtualization の基礎</li><li>- 仮想マシンの移行</li></ul>
14:30 - 14:50	<b>Lecture 2</b> <ul style="list-style-type: none"><li>- ネットワーク管理／ストレージ管理</li><li>- バックアップリストア</li></ul>
14:50 - 16:00	クロージング、QA、ご相談

休憩時間は  
適宜ご自由にお取り下さい

※ハンドズオン環境は本日 18時までは自由にご利用いただけます。  
Lab2以降はご自由に実施ください。

# オープニング

# The **WORLD** is changing



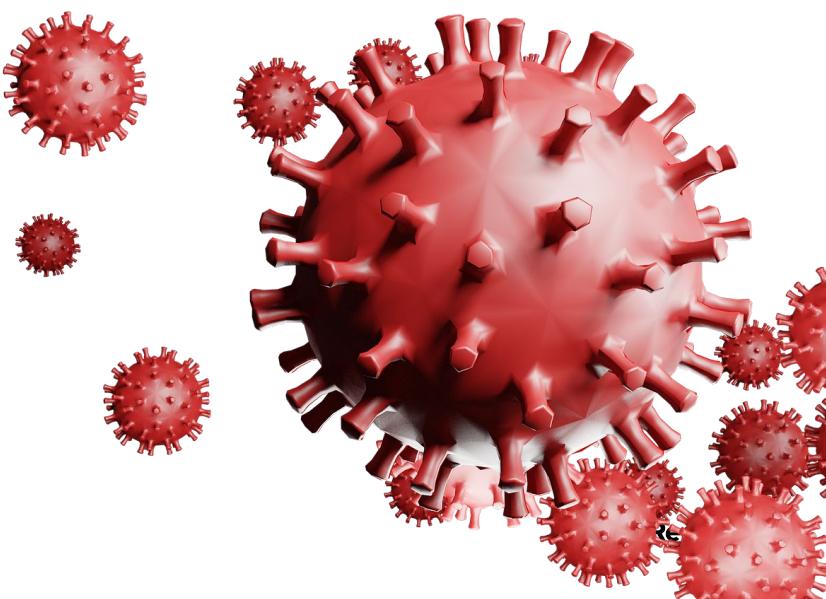
# 2005



Luca Bruno / AP



# And then COVID happened



We are living in one of the  
**BIGGEST SHIFTS** in human  
history !!

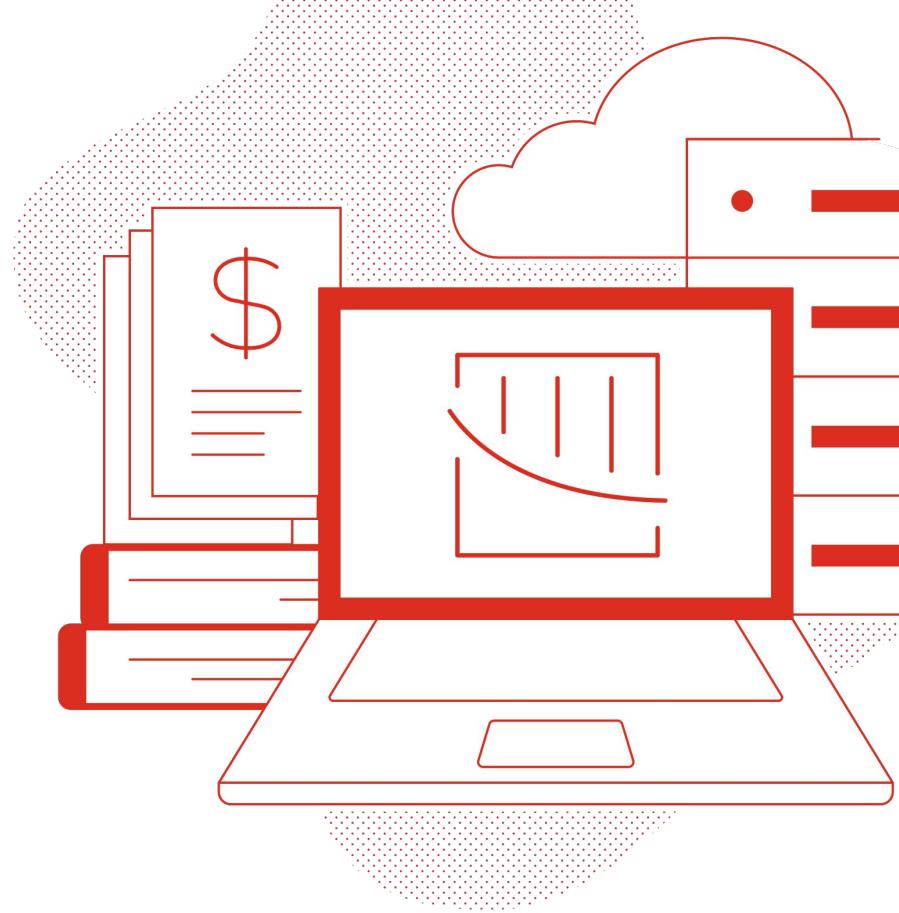


**vmware**<sup>®</sup>

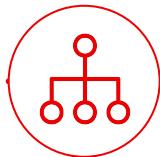
by **Broadcom**

サーバー仮想化業界は劇的に  
変化しています

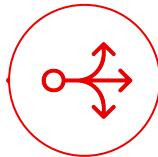
あなたの選択肢は？



# 今日のワークショップが終わると皆さんは...



仮想インフラストラクチャの  
代替ソリューションを検討する



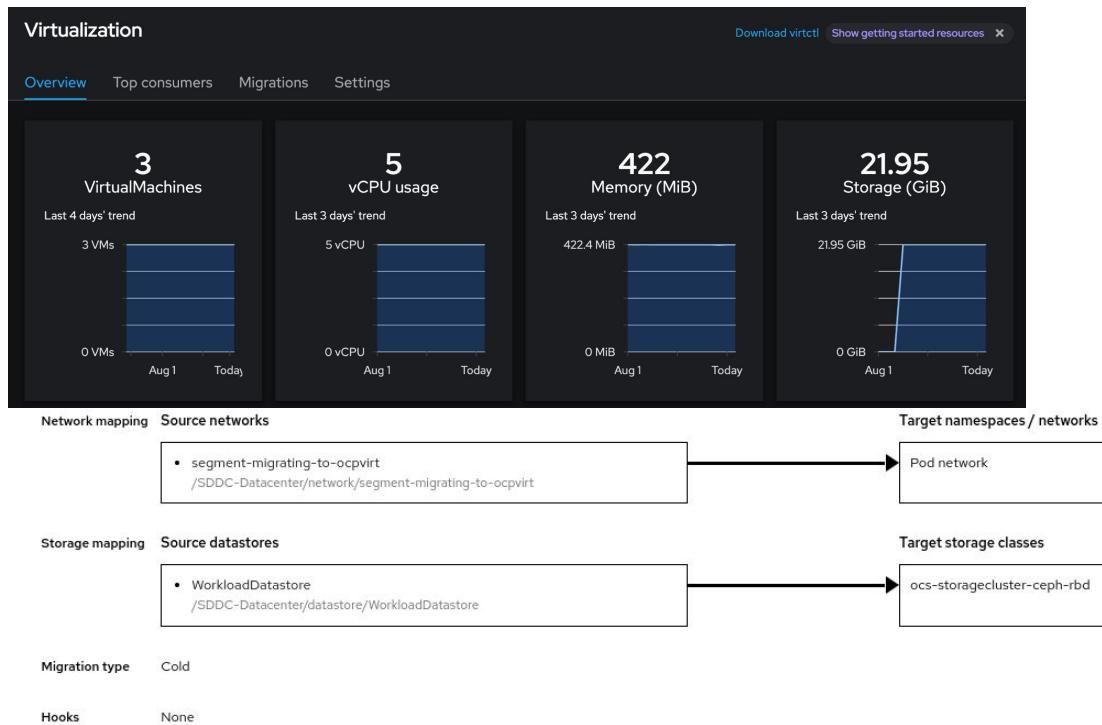
Red Hat OpenShift Virtualization を知る



インフラモダナイゼーションへの旅が今日から  
始まるることを認識する

# 本ハンズオンワークショップでカバーする範囲

- 仮想マシン
  - プロビジョニング
  - 管理
  - Live migration
- プラットフォーム
  - ストレージ
  - ネットワーク
  - ロードバランサ
- 移行
  - vSphere -> OpenShift
- バックアップ・リストア



# Lecture 1

- OpenShift Virtualization の基礎
- 仮想マシンの移行

# OpenShift Virtualization の基礎

# Red Hat OpenShift

次世代の仮想化を担うアプリケーション・プラットフォーム

## ➤ “Unified” なプラットフォーム

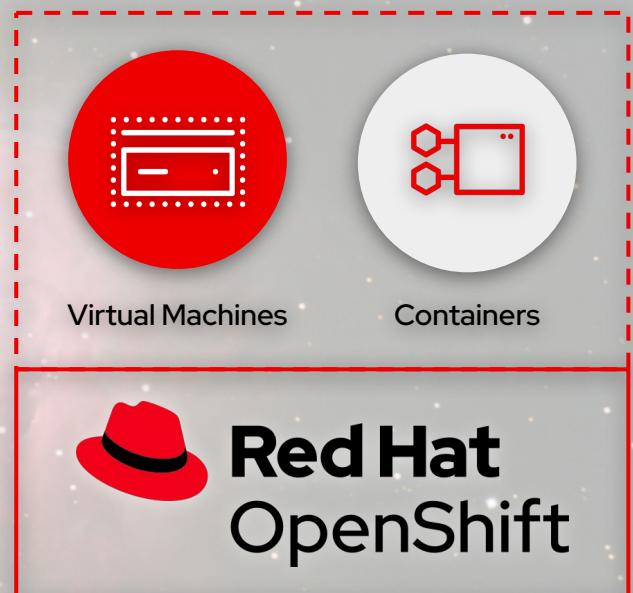
仮想マシンとコンテナ、アプリケーションの形態を問わず単一の  
プラットフォームでサポート

## ➤ インフラコストを抑制

サーバー課金<sup>\*1</sup> のサブスクリプションで、CPU リソースが多く必要な仮想  
化基盤でも経済的

## ➤ 生産性向上を支援するサービス

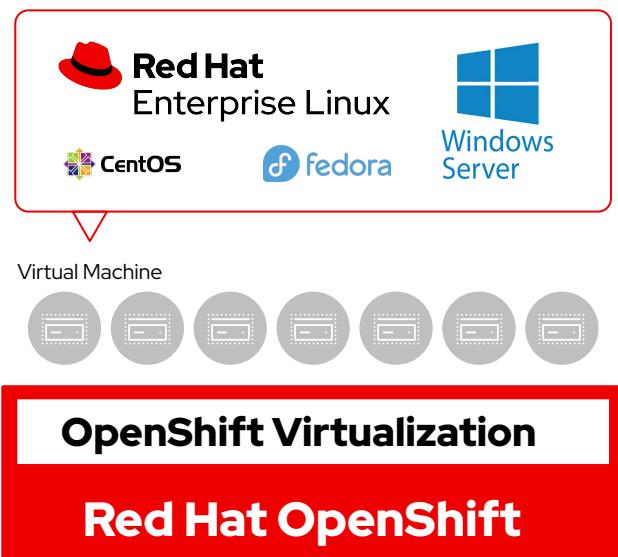
煩雑な作業を自動化し、運用者と開発者が職務上の本分に集中すること  
を支援する数々のサービス



\*1 物理サーバーを使用する場合。2 CPU・合計 64 コアまでを 1 サーバー分のサブスクリプションで使用可能。

# OpenShift Virtualization

- OpenShift が標準で提供するサーバー仮想化機能
  - Linux 仮想化機能 KVM に基づくサーバー仮想化
  - コンテナ内で実行、コンテナで管理
  - Linux, Windows 仮想マシンをサポート
- Kubernetes の作法で VM を作成
  - ポリシーに基づくスケジューリング、宣言的なデプロイ
- OpenShift のリソース／サービスとの統合
  - コンピュート: Pod, Project
    - CPU / メモリの割当、名前空間
  - ネットワーク: Service, Route
    - クラスタ内部の仮想ネットワーク、外部ネットワーク接続
  - ストレージ: Persistent Volume, Storage Class
    - 永続ボリューム
  - 運用管理系／開発系サービス
    - メトリクス監視 / ログ管理、発報、バックアップ / リストア
    - CI/CD パイプライン、GitOps



# 仮想マシンの“コンテナ化”

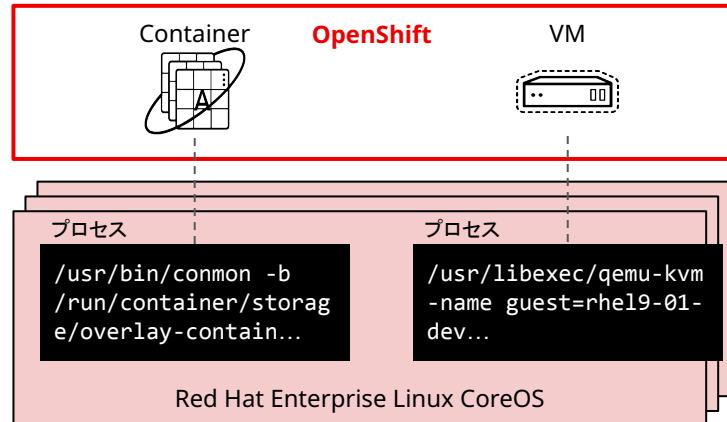
## 仮想マシンをコンテナとしてOpenShift(Kubernetes)で扱う

### KVM仮想マシンとコンテナの共通点

- どちらもLinuxホストから見るとLinuxプロセスである
- どちらもリソース(CPU / メモリ、ネットワーク、ストレージ)を必要とする



仮想マシンはコンテナと同じように扱える



# KubeVirt

Kubernetes上で仮想マシンを動かすことを目的としたオープンソースプロジェクト



- 2016年にRed Hatがプロジェクトをスタート
- 2017年初にOSSとして公開
- 2019年にCNCFプロジェクトの一部に
- 2023年7月にversion1.0をリリース



## What does v1.0 mean to the community?

The v1.0 release signifies the incredible growth that the community has gone through in the past six years from an idea to a production-ready Virtual Machine Management solution. The next stage with v1.0 is the additional focus on maintaining APIs while continuing to grow the project. This has led KubeVirt to adopt community practices from Kubernetes in key parts of the project.

v1.0のリリースは、過去6年間にコミュニティがアイデアから本番稼動可能な仮想マシン管理ソリューションまで、信じられないほどの成長を遂げたことを意味します。v1.0の次の段階は、APIを維持しながらプロジェクトを継続的に成長させることにさらに重点を置くことです。KubeVirtはプロジェクトの鍵となる箇所において、Kubernetesのコミュニティ・プラクティスを採用していきます。

KubeVirtのコミュニティブログより

# OpenShift で仮想マシンを動かすことのメリット



## クラウドのような体験で生産性を高める

- ネイティブなマルチテナントを活用したセルフサービスの実現
- 人手を廃し徹底的な自動化による短時間での仮想マシン作成



## 仮想マシンとコンテナの統一で効率を上げる

- 同じ管理インターフェース (GUI, CLI, API) を使った管理手法の統一
- インフラリソース (コンピュート, ネットワーク, ストレージ) の統一



## 様々な側面からのコスト抑制アプローチ

- 割安なサブスクリプション体系、無制限の Red Hat Enterprise Linux VM
- 標準で利用できる、運用管理およびアプリ開発を助けるサービス

# OpenShift クラスタを構成するノード



## Control Plane Node (3 nodes)

- ・ クラスタ全体にわたってリソース<sup>1</sup>を管理する役割を持つノード
- ・ OS は Red Hat Enterprise Linux CoreOS (RHCOS)

API Server: リソースのライフサイクル制御に使うインターフェース

Controllers: リソースの状態を常に監視し好ましい状態になるよう制御

Schedulers: リソースの配置するノードを決定

Authentication: ユーザー認証サービスを提供

etcd: クラスタの状態や構成を管理するデータベース

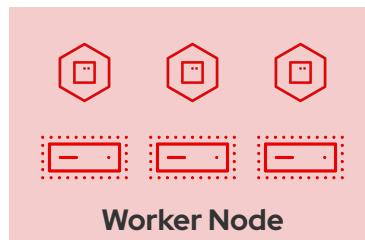
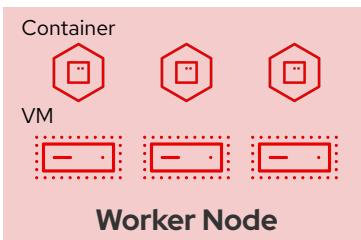
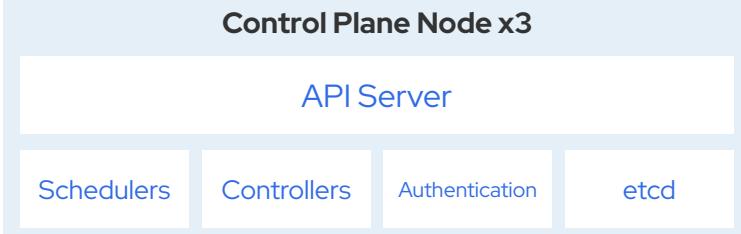


## Worker Node (2～N nodes)

- ・ スケジュールされたコンテナやVM が稼働するノード
- ・ ワークロードの量によってノード数を決める。途中で追加することも可能
- ・ OS は Red Hat Enterprise Linux CoreOS (RHCOS)

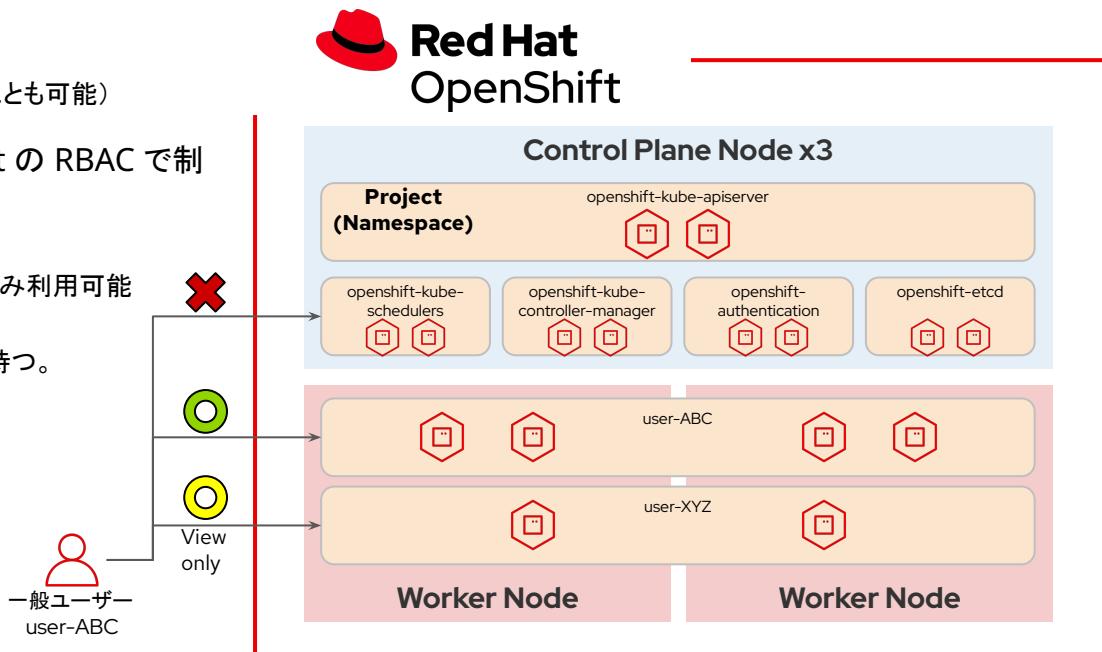


**Red Hat  
OpenShift**



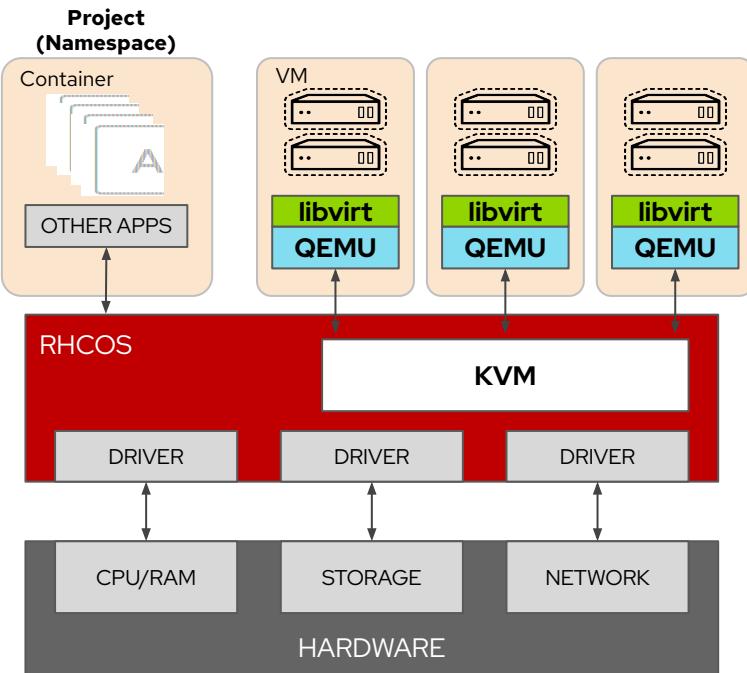
# OpenShift のマルチテナント・アーキテクチャ

- 全てのコンテナと VM は、OpenShift クラスタ内のいずれかの “Project\*1” という名前空間で実行される。
  - Project はクラスタ全体で適用される。
  - 各Project は他の Project と分離されている。
    - Project 間での通信は可能(制限することも可能)
- Project へのユーザーの権限は OpenShift の RBAC で制御される。
  - Create / View / Edit / Delete
  - 一般ユーザーは自身が権限を持つProject のみ利用可能
    - 自分で Project を作ることも可能。
  - クラスタ管理者は全てのProject の全権限を持つ。



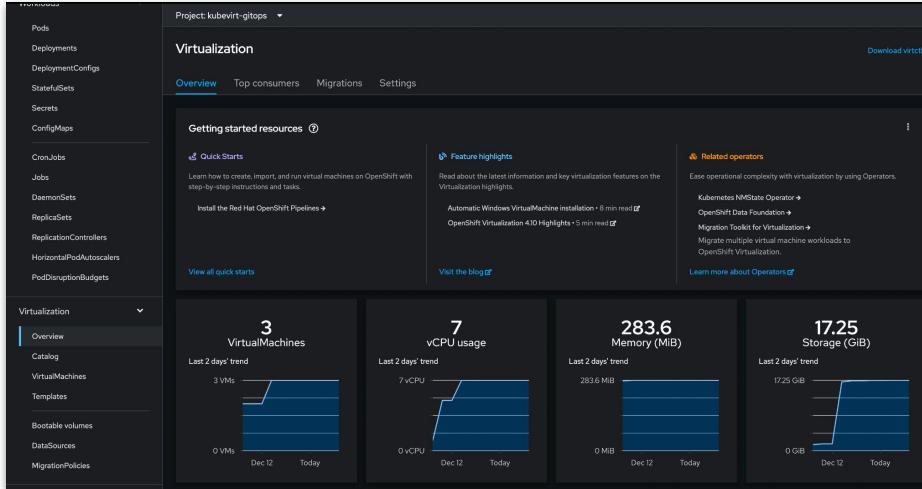
# KVM : Kernel-based Virtual Machine

- OpenShift Virtualization で使用するハイパーバイザ
  - KVM, QEMU, libvirt のスタックで仮想化
    - QEMU は仮想マシンのエミュレーションを提供
    - libvirt は仮想マシンの管理レイヤーを提供
  - Red Hat Enterprise Linux kernel のコアコンポーネント
    - 10 年以上にわたって本番利用されている実績
      - Red Hat Enterprise Linux
      - Red Hat Virtualization
      - Red Hat OpenStack Platform
- の全てで KVM+QEMU+libvirt が使用



# OpenShift Virtualization ダッシュボード

- Project の中で稼働する VM 情報のサマリを表示
  - CPUやメモリなどのインフラリソースを消費している上位の VM を表示
  - ライブマイグレーションの履歴



# VM の作成

- 様々な作成方法
  - テンプレートからクローン
  - 既存のディスクイメージの Import
  - 事前にCPU/RAMの割り当てが定義されたインスタンスタイプ
  - YAMLファイル

The screenshot shows the Red Hat OpenShift web interface. On the left, a sidebar menu includes 'Administrator', 'Home', 'Operators', 'Workloads', 'Virtualization' (selected), 'Overview', 'Catalog', 'VirtualMachines' (selected), and 'Templates'. The main area is titled 'VirtualMachines' and shows three existing VMs: 'database' (status: Running, IP: 10.128.2.43, port: 80, host: example.com), 'winweb01' (status: Running, IP: 10.129.2.191, port: 80, host: example.com), and 'winweb02' (status: Running, IP: 10.129.2.190, port: 80, host: example.com). To the right, a 'Create' dropdown menu offers three options: 'From template' (marked with a red arrow 1), 'From volume' (marked with a red arrow 2), and 'With YAML'.

# VM テンプレート

- ユーザが最も簡単に VM を作成できる方法
- いくつかの OS はデフォルトで提供される
  - Red Hat Enterprise Linux, Fedora Linux, CentOS
  - Windows Server
  - 管理者がカスタムテンプレートを作ることも可能
- テンプレートで定義する情報
  - CPUとメモリ
  - ネットワーク構成
  - ストレージ構成(PVCサイズ、Storage Class)
  - その他
    - デフォルトユーザのパスワードなど
- 定義済みの情報は VM 作成の際に上書き可能

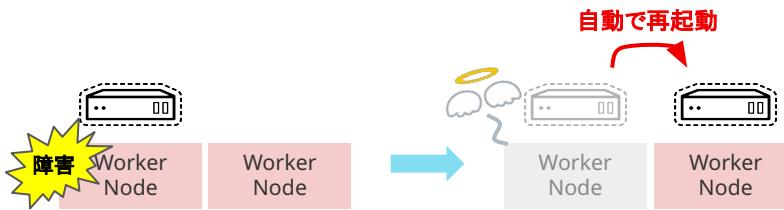
The screenshot shows the 'Create new VirtualMachine' interface. On the left, the 'Template catalog' tab is selected, displaying a list of available templates under 'Default templates'. Three Red Hat Enterprise Linux 8 VM templates are shown: 'Red Hat Enterprise Linux 8 VM (rhel8-server-small)', 'Red Hat Enterprise Linux 8 VM (rhel8-server-email)', and 'Red Hat Enterprise Linux 8 VM (rhel8-server-small)'. The middle section, 'Template info', provides detailed configuration for the selected template:

- 1 CPU | Memory:** 1 CPU | 2 GB Memory
- 2 Network interfaces (1):** Name: default, Network: Pod networking, Type: Masquerade
- 3 Disks (2):** Name: r0ndisk, c0ndisk, Drive: Disk, Disk, Size: 30 GB

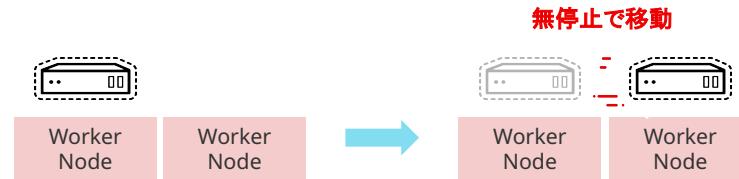
The right side of the interface includes a 'Quick create VirtualMachine' section with fields for 'VirtualMachine name' (set to 'rhel8-revolutionary-antelope') and 'Project' (set to 'unimported'). A checkbox 'Start this VirtualMachine after creation' is checked. At the bottom right are 'Quick create VirtualMachine' and 'Customize VirtualMachine' buttons, and a 'Cancel' button.

# 仮想化基盤としての機能

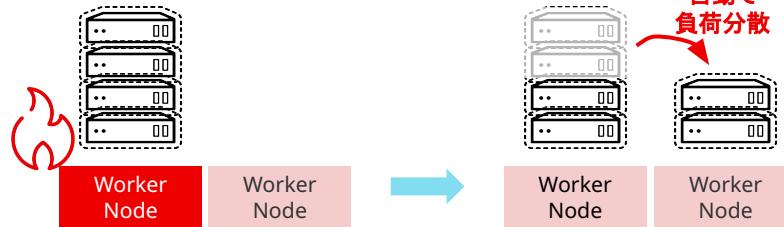
- 高可用性(HA)



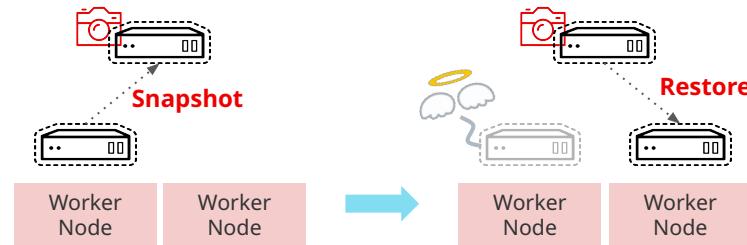
- ライブマイグレーション



- 負荷分散



- VM スナップショット / リストア



# 仮想マシンの移行

# VM を維持すること

- VM のまま移行する必要性
  - クリティカルなアプリケーションを維持する役割と責任
  - リファクタリングに時間のかかるモノリス
  - 長期的な視点でのスキルのモダナイズ
- モダンなプラットフォームにおける従来型 VM のポイント
  - 管理のコンセプトと作業
  - ネットワークとストレージ
  - 冗長化(High Availability, Fault Tolerance)
- OpenShift Virtualization への移行ツール
  - Migration Toolkit for Virtualization (MTV)



# お客様のペースでモダナイズ

従来の仮想化

仮想マシン

緩慢な進化  
⌚

増加するコスト  
💲



開発者の生産性の限界

インフラのモダナイズ初期

OpenShift  
Virtualization



仮想マシン

クラウドの柔軟性 + 拡張性



コストの削減



IT効率と信頼性の向上

Re-platform

ワークフローとインフラの  
モダナイズ

OpenShift  
Virtualization



仮想マシン and/or コンテナ



スピードにおける革新



売上への貢献



開発者のアウトプットの向上

Refactor

Refactor

# Migration Toolkit for Virtualization (MTV)

- 様々なプラットフォームで稼働する VM を OpenShift Virtualization 上に移行するツール
  - OpenShift 上でインストールして実行
- 移行元として選択できる Provider
  - VMware vSphere
  - Red Hat Virtualization
  - OpenStack
  - Open Virtual Appliances (OVA)
  - OpenShift Virtualization
- 移行元と移行先 (=OpenShift Virt) で、ネットワークとストレージのマッピングを行う
- 2種類の Migration タイプから選択
  - Cold Migration
  - Warm Migration

Migration Toolkit for Virtualization Successful

Overview YAML

Welcome

Migration Toolkit for Virtualization (MTV) migrates virtual machines at scale to Red Hat OpenShift Virtualization. You can migrate virtual machines from VMware vSphere, Red Hat Virtualization, OpenStack, OVA and OpenShift Virtualization source providers to OpenShift Virtualization with the Migration Toolkit for Virtualization (MTV).

This gives organizations the ability to more easily access workloads running on virtual machines, while developing new cloud-native applications.

Migrations are performed in a few simple steps, first by providing source and destination credentials, then mapping the source and destination infrastructure and creating a choreographed plan, and finally, executing the migration effort.

**Migrations**

Total	Running	Failed	Succeeded
2	0	0	1
1 canceled			

**Virtual Machine Migrations**

Total	Running	Failed	Succeeded
2	0	0	1
1 canceled			

Settings

Max concurrent virtual machine migrations  
20

Must gather cleanup after (hours)  
Disabled

Controller main container CPU limit  
500m

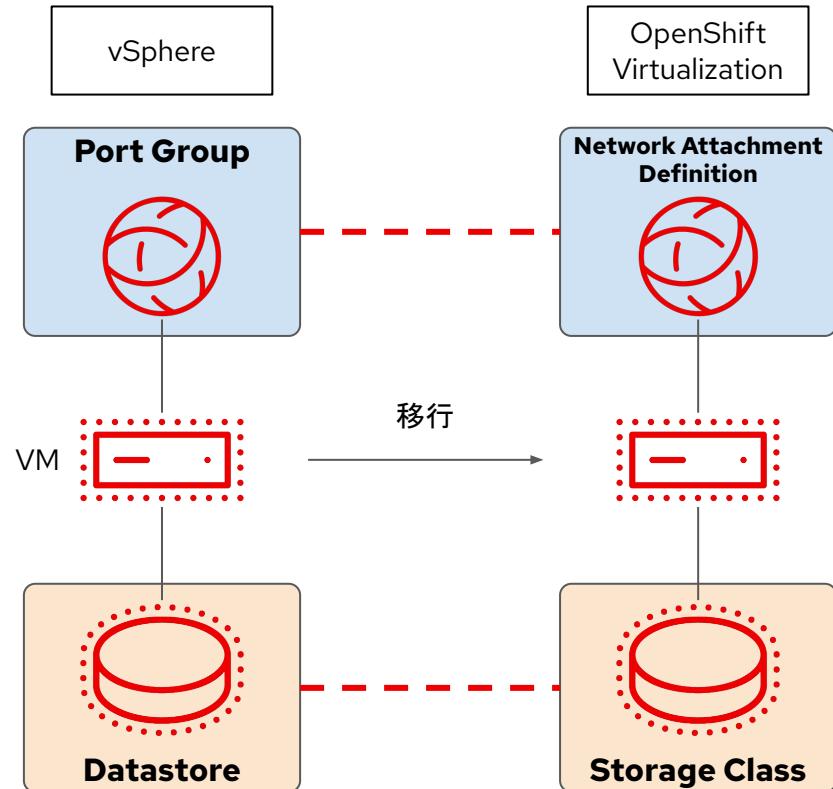
Controller main container Memory limit  
800Mi

Precopy interval (minutes)  
60

Snapshot polling interval (seconds)  
10

# ネットワーク／ストレージのマッピング

- ネットワークのマッピング
  - オーバーレイネットワークの仮想スイッチに相当するもの同士をマッピング
  - vSphere では Port Group が相当
  - OpenShift Virtualization では Bridge が相当
    - Bridge は “Network Attachment Definition (NAD)” という形で抽象化される
    - デフォルトでは Pod Network 用の NAD が作られており、任意で追加することも可能
- ストレージのマッピング
  - 仮想ディスクの格納先に相当するもの同士をマッピング
  - vSphere では Datastore が相当
  - OpenShift Virtualization では Storage Class が相当



# 仮想マシン移行のフロー

- 移行プラン (Plan) の作成
  - 対象の VM の選択
  - ネットワーク／ストレージのマッピング
  - 移行タイプの選択
  - 移行前後に自動実行する処理
- 移行スタート
  - 事前処理
  - 移行後 VM が利用する仮想ディスク (PVC) を発行
  - 移行元 VM をコピーして KVM の形式に変換
  - 仮想ディスクの中身を PVC にコピー
  - 移行後 VM を作成

Migration details by VM					
Name	Start time	End time	Data copied	Status	
① winweb01	12 Mar 2024, 11:56:...	12 Mar 2024, 14:27:...	90.00 / 90.00 GB	Complete	
Step					
② Initialize migration		00:00:40		Completed	
③ Allocate disks		00:00:00		Completed	
④ Convert image to kubevirt		00:14:55		Completed	
⑤ Copy disks		02:15:01		Completed	
⑥ Create VM		00:00:00		Completed	
① winweb02	12 Mar 2024, 11:56:...	12 Mar 2024, 14:32:...	90.00 / 90.00 GB	Complete	
① database	12 Mar 2024, 11:56:...	12 Mar 2024, 12:47:...	16.00 / 16.00 GB	Complete	

# Lab 1

- OpenShift Virtualization の基礎
- 仮想マシンの移行

# 標準モジュール

- 仮想インフラ管理者・仮想インフラユーザ向け
  - OpenShift Virtualization の基礎
  - 仮想マシンの移行

## Q&Aの方法



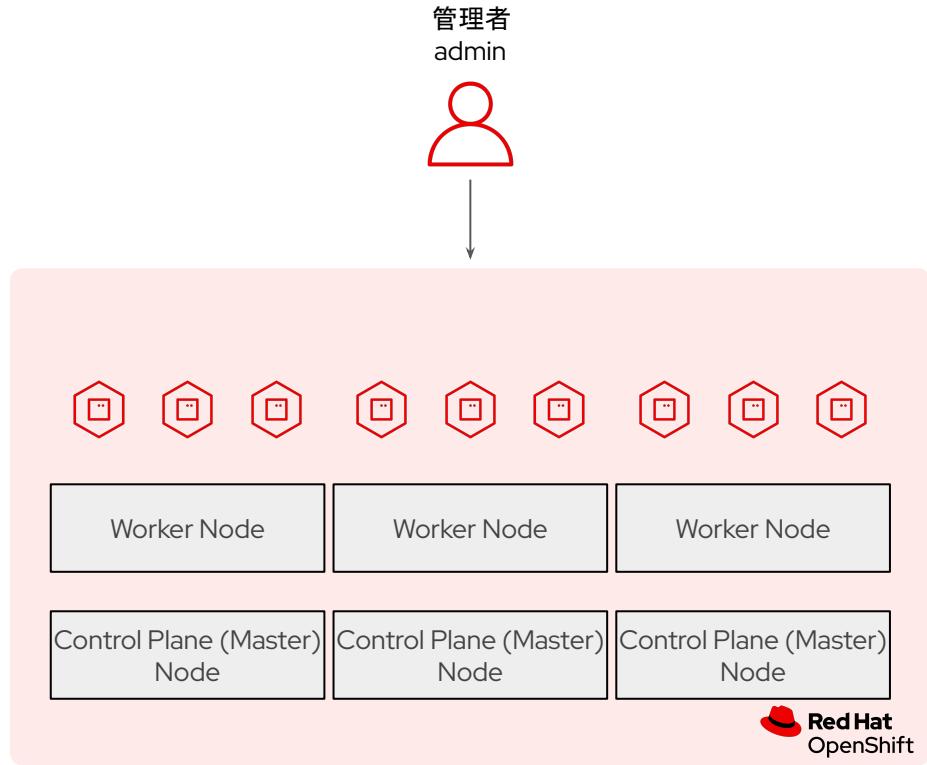
本日のご質問はEtherpad で受け付けます。  
EtherpadのURL

<https://red.ht/ocpv-ws-toinx-ep>

受講者の皆様でQ&Aを共有できるよう、Etherpad でのご質問にご協力をお願い  
いたします。

# ワークショップで使用する OpenShift クラスタ

- プラットフォーム
  - ベアメタル (Equinix Cloud)
- ノード構成
  - 3 x Control Plane (Master) ノード
  - 3 x Worker ノード
- バージョン
  - OpenShift Container Platform 4.15
- 事前に導入済みのソフトウェア／ツール
  - OpenShift Virtualization
  - Migration Toolkit for Virtualization
  - OpenShift Data Foundation
  - NMState
  - MetallLB
- ユーザー
  - 管理者ユーザ: admin



## ラボ環境の取得 (1/4)

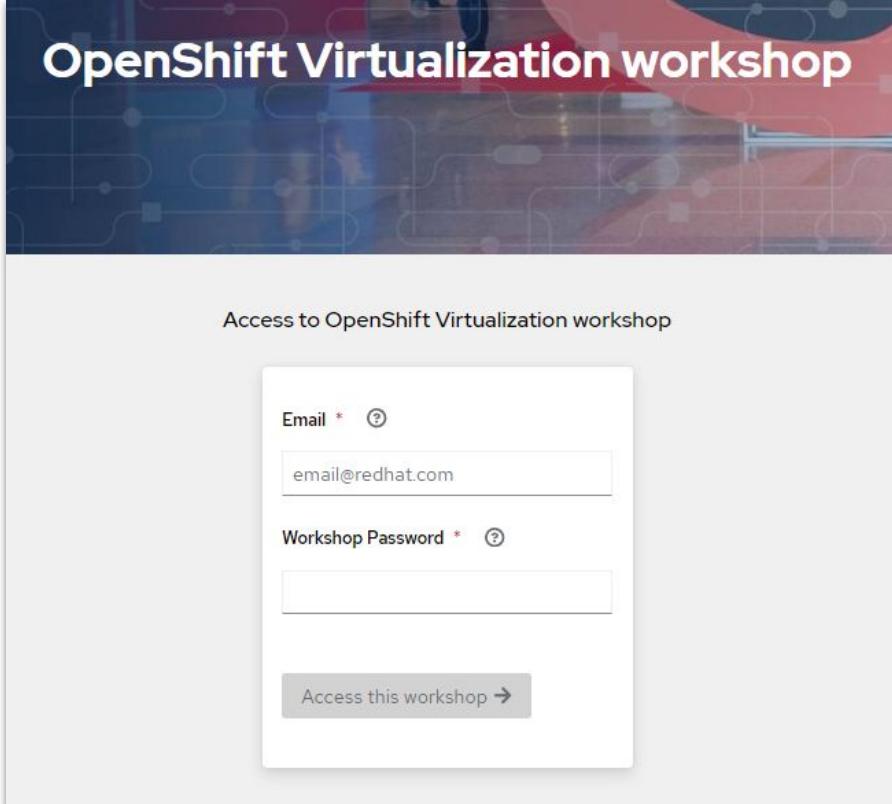
- 下の URL にブラウザでアクセスして下さい。

<https://red.ht/ocpv-ws-toinx>

## ラボ環境の取得 (2/4)

- 右のような画面が出たら、Email と Workshop Password を入力してください。
- Emailアドレスは単にユニークな ID として使うだけです。本ワークショップ以外の目的で使われることはありません。
- Workshop Password は、

ocpv4all



The screenshot shows a login form titled "Access to OpenShift Virtualization workshop". It features two input fields: "Email \*" containing "email@redhat.com" and "Workshop Password \*". Below the password field is a large empty input box. At the bottom is a button labeled "Access this workshop →". The background of the page has a dark blue and red abstract circuit board design.

OpenShift Virtualization workshop

Access to OpenShift Virtualization workshop

Email \*

email@redhat.com

Workshop Password \*

Access this workshop →

# ラボ環境の取得 (3/4)

- 右図のような画面が出たら、ラボ環境の取得は成功です。
- “Lab User Interface”に続くリンクをクリックし、下図のように日本語のテキストが表示されることを確認して下さい。

The screenshot shows the 'Introduction' page of the 'Red Hat OpenShift Virtualization workshop'. The page has a sidebar with navigation links like 'Books', 'Cheatsheets', 'Upcoming Events', 'More Tutorials', and 'Download'. The main content area features a large image of a server rack with the text 'Red Hat OpenShift Virtualization workshop'. Below the image, there are two sections: '1.はじめに' and '2.ワークショップモジュール'. The '1.はじめに' section contains text about OpenShift Virtualization and its features. The '2.ワークショップモジュール' section lists '標準モジュール' and 'オプションモジュール'. At the bottom, there's a note about OpenShift Virtualization's prerequisites.

39

The screenshot shows the 'Lab User Interface' window. It displays Japanese instructions for the workshop, including host information, DNS name, SSH user, SSH password, OpenShift web console URL, OpenShift API server URL, OpenShift Admin User, and OpenShift Admin Password. A red box highlights the 'Lab User Interface' link at the top of the window, which is also shown in the browser's address bar.

OpenShift Virtualization workshop

Instructions for OpenShift Virtualization workshop

Lab User Interface <https://showroom-showroom.apps.w8rch.dynamic.redhatworkshops.io/>

Messages

Lab instructions: <https://showroom-showroom.apps.w8rch.dynamic.redhatworkshops.io/>  
The host you will be using for lab: 136.144.49.105  
DNS name: hypervisor.w8rch.dynamic.redhatworkshops.io  
SSH User: lab-user  
SSH Password: 2Lr9fvloYLi7  
OpenShift web console: <https://console-openshift-console.apps.w8rch.dynamic.redhatworkshops.io>  
OpenShift API server: <https://api.w8rch.dynamic.redhatworkshops.io:6443>  
OpenShift Admin User: admin  
OpenShift Admin Password: eiZ:0a2MVII5  
To connect to the Bastion VM for system:admin CLI access: sudo ssh root@192.168.123.100



# ラボ環境の取得 (4/4)

- OpenShift Web コンソール (GUI) の URL にアクセスしてください。
  - “OpenShift web console:” に続く URL です。
- ログイン画面が表示されたら、"admin" ユーザでログインしてください。
  - パスワードは、“OpenShift Admin Password:” に続く文字列です。
  - ダッシュボードの画面が表示されれば、ログインに成功です。テキストに沿ってラボを始めて下さい。

The image shows two screenshots related to the OpenShift Virtualization workshop.

The top screenshot is a landing page titled "OpenShift Virtualization workshop". It contains instructions for the workshop, including the lab user interface URL (<https://showroom-showroom.apps.w8rch.dynamic.redhatworkshops.io/>), host information (IP: 136.144.49.105, DNS name: hypervisor.w8rch.dynamic.redhatworkshops.io), and SSH credentials (User: lab-user, Password: 2Lr9fvloYl7). It also lists the OpenShift web console URL (<https://console-openshift-console.apps.w8rch.dynamic.redhatworkshops.io>), API server URL (<https://api.w8rch.dynamic.redhatworkshops.io:6443>), and OpenShift Admin User/Password (admin/eiZ:0a2MViI5).

The bottom screenshot is a login screen titled "アカウントにログイン" (Log in to account). It features a Red Hat OpenShift logo and a "Red Hat OpenShift" watermark. The login form has fields for "ユーザー名" (User name) containing "admin" and "パスワード" (Password) containing ".....". A blue "ログイン" (Login) button is at the bottom.



Let's get familiar with the product

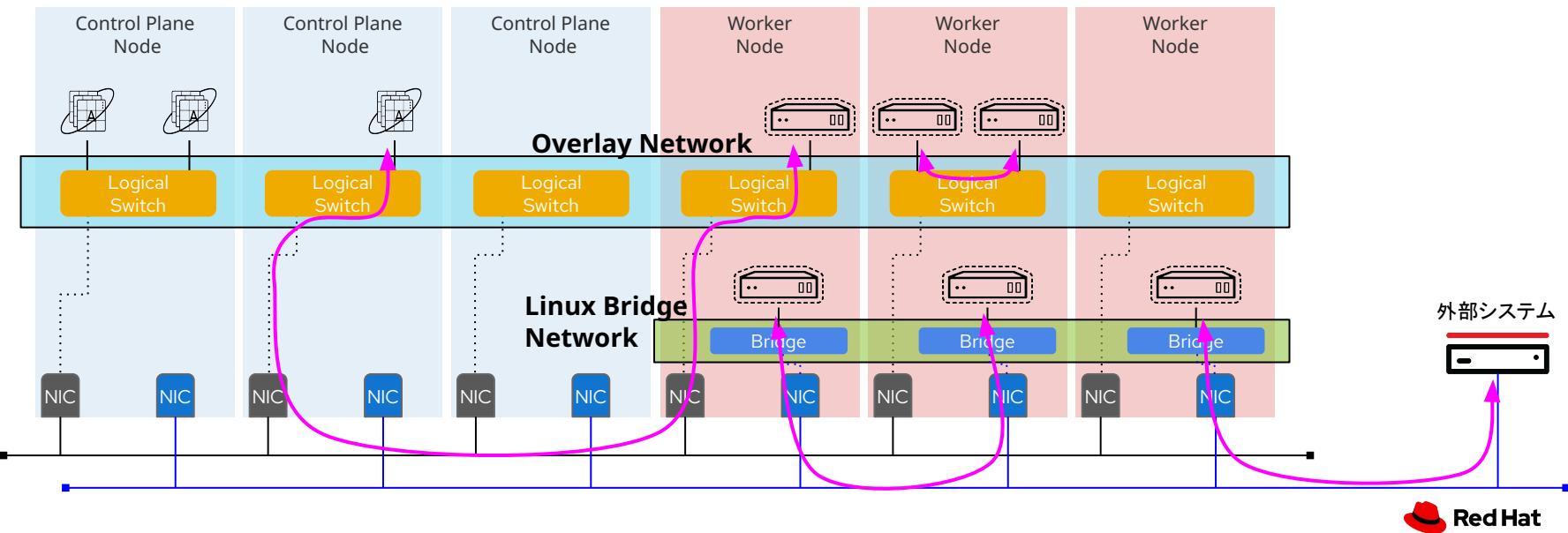
# Lecture 2

- ネットワーク管理／ストレージ管理
- バックアップとリストア

# ネットワーク管理

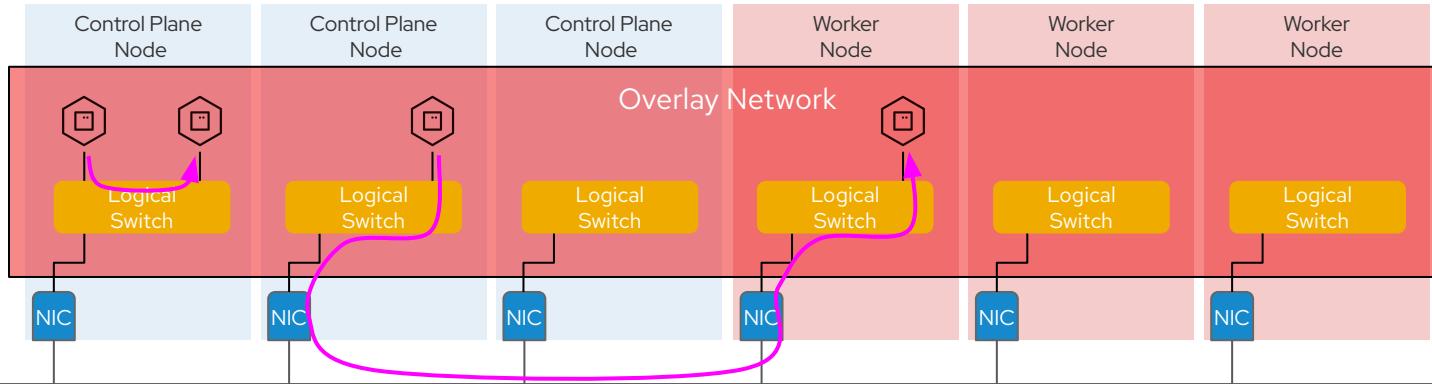
# OpenShift の仮想ネットワーク

- OpenShift デフォルトで構成される仮想ネットワーク(オーバーレイネットワーク)接続
- Bridge を使った外部ネットワークへの接続



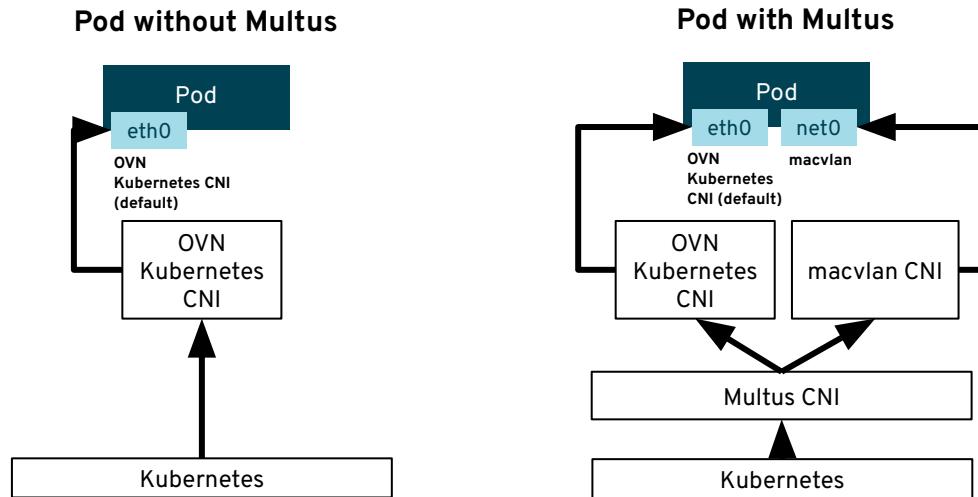
# OpenShift のオーバーレイネットワーク

- OpenShift は全てのノードにまたがった仮想ネットワーク(オーバーレイネットワーク)を内部で持つ。
- オーバーレイネットワークは SDN を使って構成される。
  - OVN-Kubernetes, OpenShift SDN, etc
- 各ノードで論理的なスイッチを持ち、ノード内の通信は論理スイッチを介して行われる。
- ノード間通信はトンネリングプロトコル (Geneve, VXLAN等) でカプセル化して行われる。
- 実装は使用する SDN によって異なる。



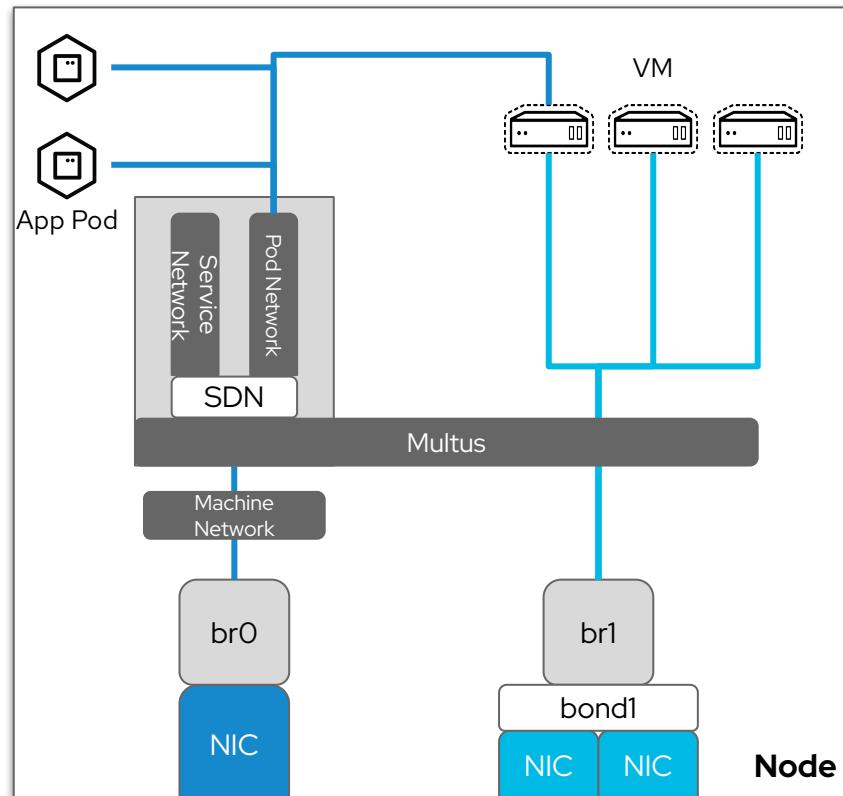
# 複数の NIC を持つための Multus

- 通常 Pod はオーバーレイネットワークに接続する NIC を 1 つだけ持つが、Multus CNI により複数の NIC を持つことができる。



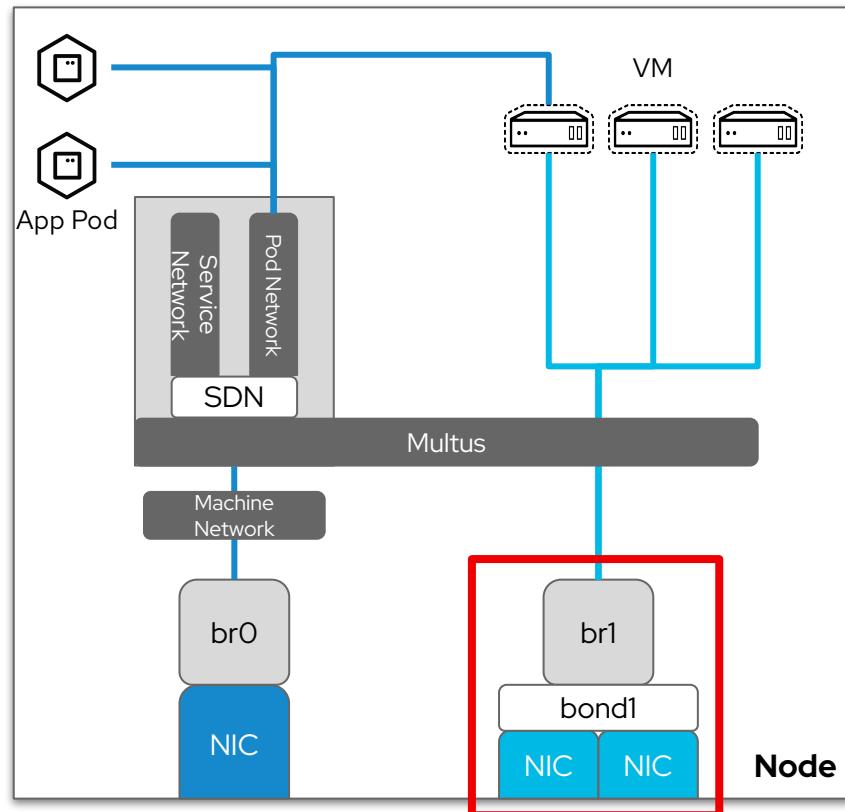
# 仮想マシンのネットワーク

- デフォルトは OpenShift クラスタ内で閉じたネットワーク(Pod Network)だけ接続可能
  - OVN-Kubernetes, OpenShift SDN
- Pod Network を経由してクラスタ外部に公開するには、Service や Route/Ingress を活用
  - コンテナアプリケーション Pod と同じ扱い
- Pod Network 以外のネットワークに VM を接続する場合は、Network Attachment Definition で定義
  - Multus CNI を使って複数の仮想 NIC を付与
    - Bridge, SR-IOV, 2nd OVN network
    - 外部ネットワーク



# NMState

- ホスト(ノード)のネットワークを設定する手法
  - 通常は OS (RHCOS) で Network Manager を操作して行う作業を、OpenShift 上から実施
  - 主に外部ネットワークを VM に接続する際に使用
    - Bridge
    - Bonding
    - VLAN
- NMState Operator を使ってカスタムリソースを定義し、宣言的に設定
  - NodeNetworkConfigurationPolicy
  - NodeNetworkState



# Node Network Configuration Policy

- ホストのネットワークインターフェースを設定
- Web コンソールのフォーム、または YAML で定義
  - Ethernet : IP アドレス(静的, DHCP)
  - Bridge : IP アドレス
  - Bonding : mode 1-6, IP アドレス
- Node Selector を使って特定のノードのみ設定をすることも可能
  - デフォルトは全ノードに設定を適用

Create NodeNetworkConfigurationPolicy Edit YAML

Node network is configured and managed by NM state. Create a node network configuration policy to describe the requested network configuration on your nodes in the cluster. The node network configuration enactment reports the network policies enacted upon each node.

Apply this NodeNetworkConfigurationPolicy only to specific subsets of nodes using the node selector

**Policy name \***  
bond1-emp5s0f0-emp5s0f1-policy

**Description**  
Bond emp5s0f0 and emp5s0f1

**Policy Interface(s) ?**  
+ Add another interface to the policy

Bonding bond1

**Interface name \***  
bond1

**Network state \***  
Up

**Type \***  
Bonding

**IP configuration**  
 IPv4

**Port**  
emp5s0f0,ens5s0f1  
Use commas to separate ports

> Copy MAC address

**Aggregation mode \***  
active-backup

+ Add Option

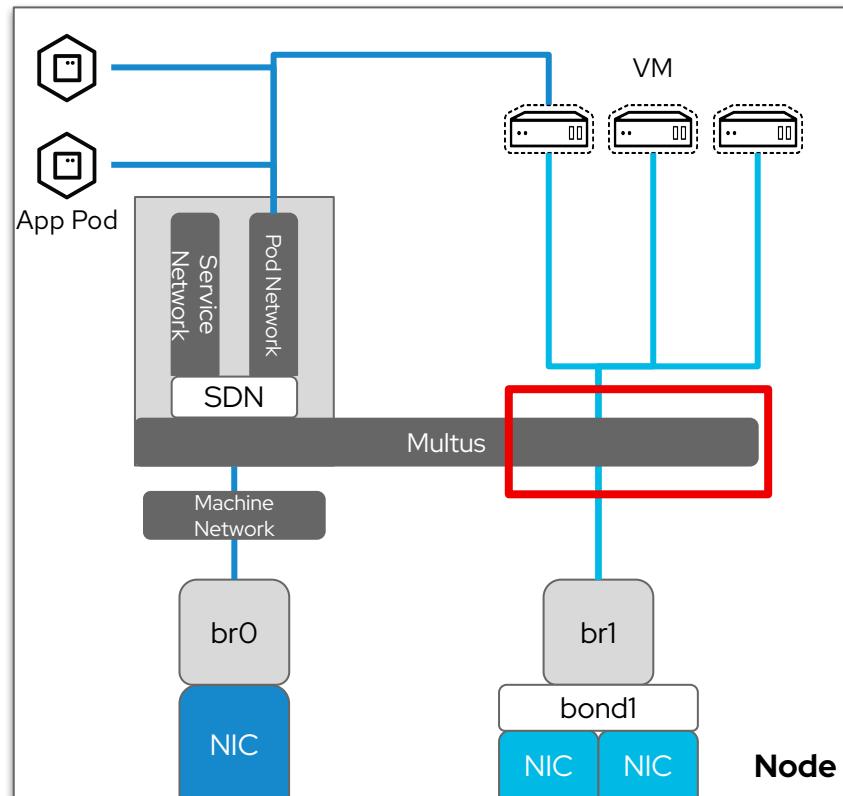
# Node Network State

- 現状のホストのネットワーク設定を表示

Network details 18 Interfaces						
Name	IP address	Ports	MAC address	LLDP	MTU	
▼ bond						
bond0 †	-	2	A0:36:9F:B9:3B:A0	<input type="checkbox"/>	1500	
▼ ethernet						
eno1 †	fe80:d022:5b76:65b0:dc0b/64	-	18:66:DA:8A:C0:F7	<input type="checkbox"/>	1500	
eno2 †	-	-	18:66:DA:8A:C0:F8	<input type="checkbox"/>	1500	
eno3 †	-	-	18:66:DA:8A:C0:F9	<input type="checkbox"/>	1500	
eno4 †	-	-	18:66:DA:8A:C0:FA	<input type="checkbox"/>	1500	
enp4s0f0 †	-	-	A0:36:9F:B9:3B:A0	<input type="checkbox"/>	1500	
enp4s0f1 †	-	-	A0:36:9F:B9:3B:A0	<input type="checkbox"/>	1500	
enp5s0f0 †	fe80:b57:80e7:c02d:7fle/64	-	A0:36:9F:B9:3C:1C	<input type="checkbox"/>	1500	
enp5s0f1 †	fe80:7977:a277:5884:888/64	-	A0:36:9F:B9:3C:1E	<input type="checkbox"/>	1500	
geneve_sys_6081 †	fe80:7461:36ff:fe719fb8c/64	-	76:6f:36:719fb8c	<input type="checkbox"/>	65000	
➤ ovs-bridge						
➤ ovs-interface						
▼ vlan						
bond0.241 †	-	-	A0:36:9F:B9:3B:A0	<input type="checkbox"/>	1500	

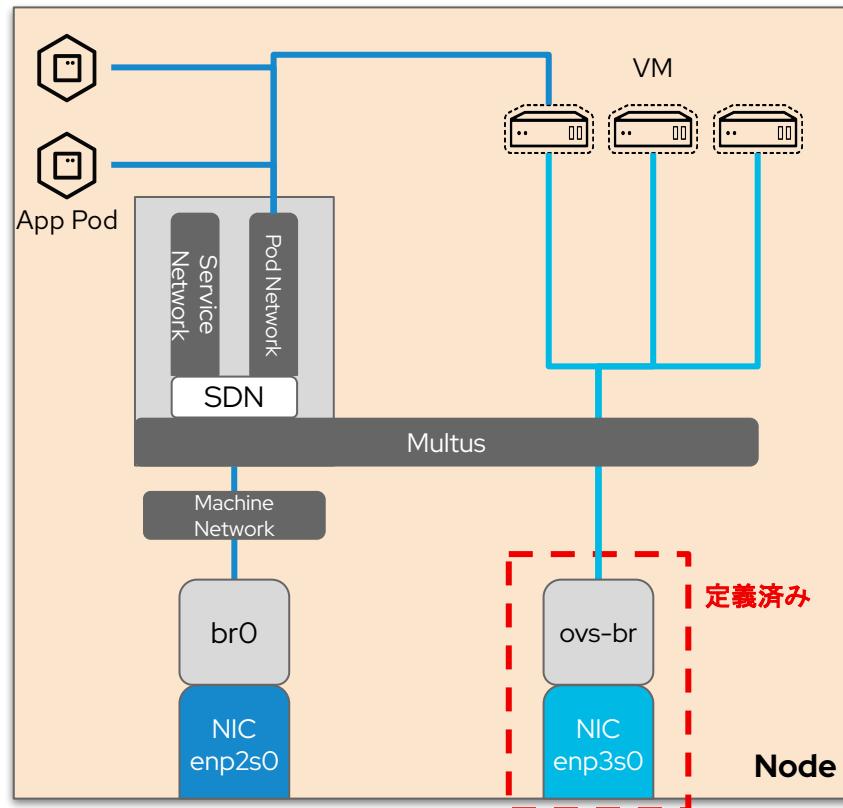
# Network Attachment Definition

- Pod Network 以外のネットワークを VM に接続する際に定義
- ホスト上の Bridge を指定して、VM が Bridge を通じてネットワークを利用できるように設定
  - オプションで VLAN を指定可能
- 作成した Project に限定して使用できるネットワーク
  - default Project で作られた Network Attach Definition は、全ての Project で利用可能



## ※Lab のネットワーク環境

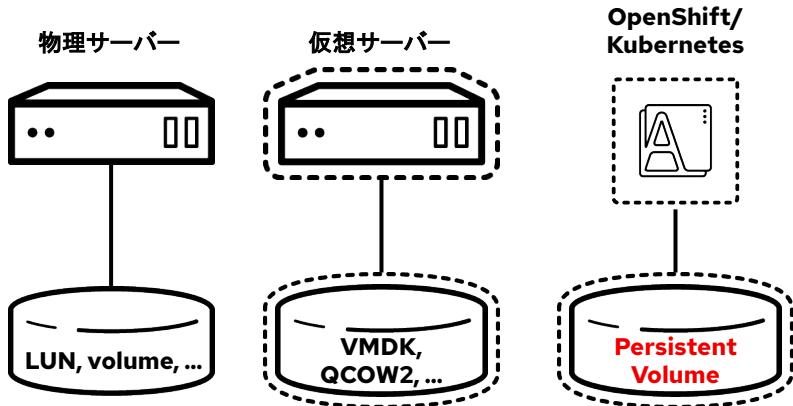
- Lab 環境では、はじめから `NodeNetworkConfiguratlonPolicy` が定義されており、“ovs-br” という Bridge がホストの NIC に対して作られています。
- Network Attachment Definition を作る際には、“ovs-br” を指定するようしてください。



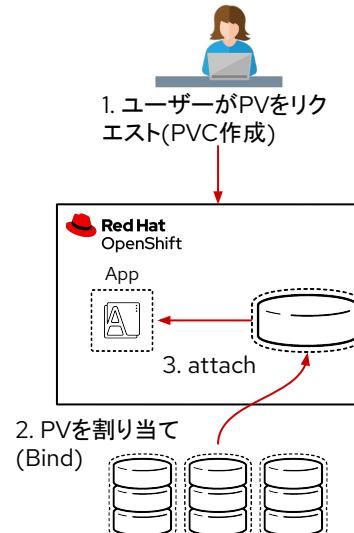
# ストレージ管理

# Persistent Volume / Persistent Volume Claim

- Persistent Volume (PV)
  - OpenShift の Pod が利用できる永続ストレージ
  - 様々な形態のストレージを抽象化した姿
- Persistent Volume Claim (PVC)
  - セルフサービスで PV を取得するためのリクエスト
  - リクエスト内容に合致する PV があれば割り当てる



- Persistent Volume Claim (PVC)
  - セルフサービスで PV を取得するためのリクエスト
  - リクエスト内容に合致する PV があれば割り当てる



PVCの例

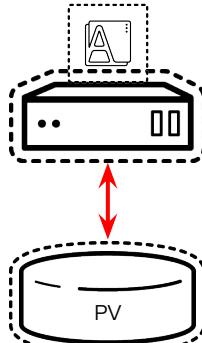
```
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: app-pvc
spec:
  accessModes:
    - ReadWriteOnce
  volumeMode: Filesystem
  resources:
    requests:
      storage: 10Gi
  storageClassName: gp3-csi
```

# Access Mode

- PVへのアクセス制御のモード
- バックエンドのストレージシステムによって、使用できる Access Mode は異なる
  - 使用できない Access Modeを指定した場合は PVCが失敗し、PVはBindされない

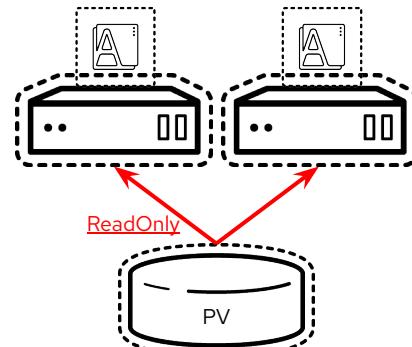
## Read Write Once (RWO)

- 1ノードからRead/Write可能
- ほぼ全てのストレージで利用可能
- 最も利用されることが多い基本のモード



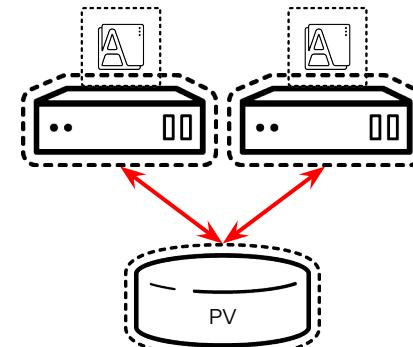
## Read Only Many (ROX)

- 複数ノードから同時にRead-Onlyでアクセス可能
- あまり使われることではなく、用途としては限定される



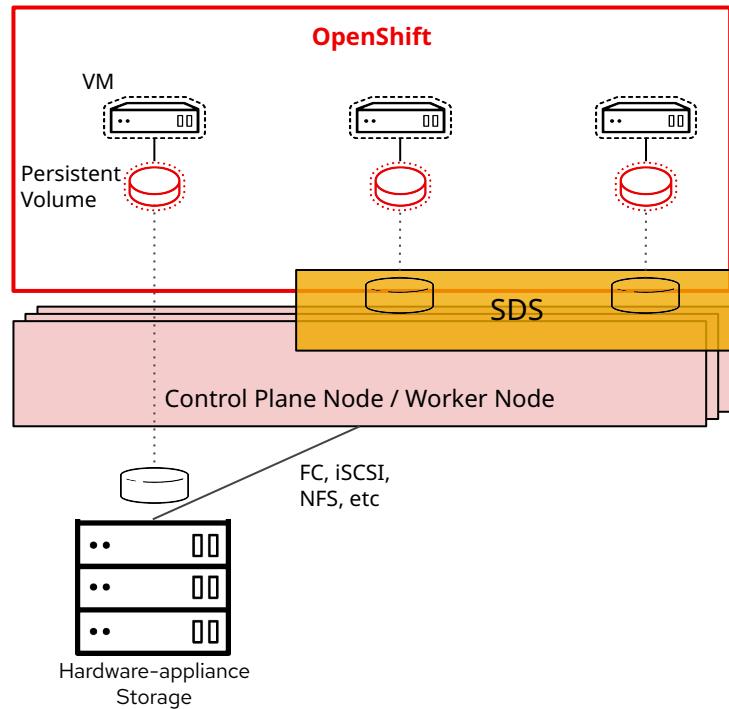
## Read Write Many (RWX)

- 複数ノードから同時にRead/Write可能
- 基本的にはファイルストレージで利用可能



# 仮想マシンのストレージ

- 一般的なストレージシステムを利用可能
  - ハードウェアアプライアンス
  - SDS(OpenShift Data Foundation, 他)
- Block Storage と File Storage の両方に対応
  - FC, iSCSI, 他
  - NFS, 他
- Persistent Volume という形式で仮想ディスクを attach
- Storage Class から Persistent Volume を作成
  - “プール”のようなイメージ
- 仮想ディスクごとに Persistent Volume = ボリュームを作成
  - VM の処理をストレージシステム側にオフロード
    - スナップショット、クローン等
  - ストレージシステムの機能を仮想ディスクに適用可能
    - リモートミラー、QoS 等



# VM が使用する仮想ディスク

- 仮想ディスクはブロックデバイスであること。
- PVC の VolumeMode によって仮想ディスクの実体は異なる。
  - Filesystem Mode  
→ ファイルシステム上に作られる Thin-provisioned な raw イメージファイル
  - Block Mode  
→ PV そのもの
- オーバーヘッドが少ない Block Mode が好ましい。
  - 多くのストレージパートナーのプロダクトは、RWX が可能な Block mode に対応している。

PersistentVolumeClaim Details	
Name	Status
rhel-rootdisk	Bound
Namespace	Capacity
NS default	20Gi
Labels	Access Modes
app=containerized-data-importer	ReadWriteMany
Annotations	Volume Mode
12 Annotations	Filesystem
Label Selector	Storage Class
No selector	managed-nfs-storage
Created At	Persistent Volume
Jul 8, 4:18 pm	PV pvc-alaac411-2e46-495a-897e-cf3bc2442199
Owner	
rhel-rootdisk	

PersistentVolume Details	
Name	Status
rhel8	Bound
Namespace	Requested capacity
NS 00-test	30 GiB
Labels	Capacity
app=containerized-data-importer app.kubernetes.io/component=storage app.kubernetes.io/managed-by=cdi-controller app.kubernetes.io/part-of=hyperconverged-cluster app.kubernetes.io/version=413.7 cdi-controller=cdi-tmp-472968b8-a5fa-4ff1-bb45-604fa274fb9b cdi.kubevirt.io=cdi-smart-clone kubevirt.io/created-by=577498f5-2f23-49cc-aca9-ae71789dbe93	30 GiB
Annotations	Used
15 annotations	0 B
Label selector	Access modes
No selector	ReadWriteMany
	Volume mode
	Block
	StorageClasses
	ocs-storagecluster-ceph-rbd

# VM スナップショット／クローン

- VM はスナップショット／クローンを取得できる。
  - ユーザが任意のタイミングで実行可能
  - スナップショット、クローンは VM 無停止で実行可能。
- CSI ドライバによって、ストレージ側でのスナップショット／クローンにオフロードできる。
  - スナップショット／クローンに対応していないバックエンドストレージを使う場合は利用不可

# ヘルパーティスク

- 副次的なディスクを使って、VM にデータを注入することができる。
  - cloud-init
  - ConfigMap
  - Secrets
  - ServiceAccount
- これらは読み取り専用のディスクで、OS が起動時にマウントして中のデータを利用する。
  - 起動後はアンマウントされており、取り外しも可能。

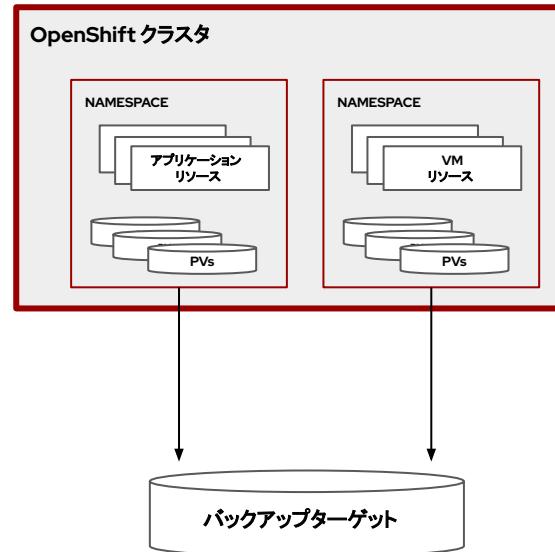
```
1  spec:
2    domain:
3      devices:
4        - disk:
5          bus: virtio
6          name: cloudinitdisk
7      volumes:
8        - cloudInitNoCloud:
9          userData: |-
10            #cloud-config
11            password: redhat
12            chpasswd: { expire: False }
13            name: cloudinitdisk
```

Name	Source	Size	Interface	Storage Class	⋮
cloudinitdisk	Other	-	VirtIO	-	⋮

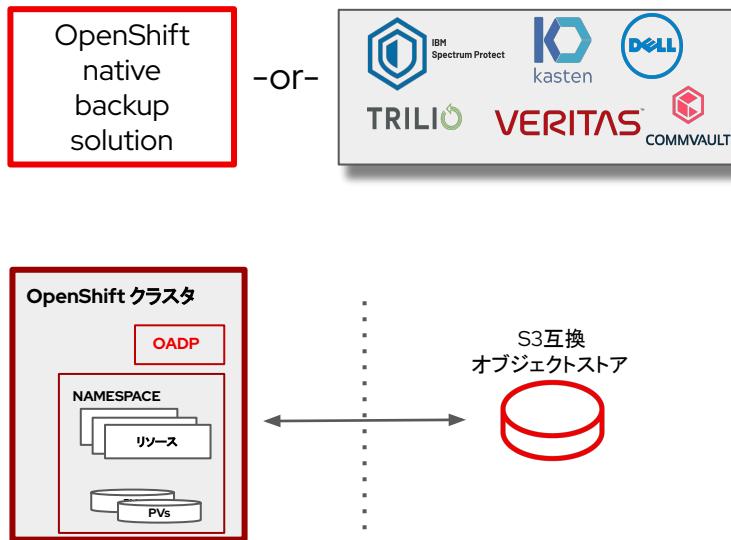
# バックアップ／リストア

# 仮想マシンのバックアップ<sup>°</sup>

- VM もコンテナアプリケーションも同様に、リソースと PVC をバックアップすることができる。
  - VirtualMachine リソース
  - VM ディスク
  - その他(ConfigMap, Secret など)
- スナップショットやクローンだけでなく、クラスタ外部にバックアップすることで、より高い耐障害性を保つことができる。



# OpenShift のバックアップソリューション



- OpenShift API for Data Protection (OADP)
  - 簡素化された OpenShift ネイティブなバックアップソリューション
  - アプリケーション単位で取得
  - バックアップパートナーのソフトウェアと連携して、より豊かなバックアップ機能も利用可能
- CSI スナップショットをサポートする全てのストレージに対応
  - CSI スナップショット未対応でもバックアップは可能
- OADP は S3 互換のオブジェクトストアをバックアップターゲットとして使用
  - クラウド・オンプレミスを問わない

# パートナーソリューションの選択基準

- OADP を単独で使うか、バックアップパートナーのソリューションと連携するかは、要件によって変化する。
- より高度なバックアップ／リストア運用や、既存のバックアップ基盤への統合などを求める場合は、パートナーソリューションを推奨する。

	OADP 単独	パートナーソリューション連携
スコープ	OpenShift のみ	OpenShift だけでなく他の基盤も統合可能
コスト	OpenShift に組み込み	SW ライセンス (+ インフラコスト)
バックアップ スケジューリング	シンプルな Cron 式 Job	ポリシーベースの包括的なスケジューリングが可能
UX	OpenShift Web コンソール	各パートナーごとのコンソール
先進機能	-	Cataloging, Indexing, advanced searching, media management
バックアップ ターゲット	S3 互換オブジェクトストレージ	豊富なメディア (NAS, S3, tape, disk, optical, public cloud ...)
ネットワーク管理 機能	シンプルにDC間／クラウドへの通信	マルチサイト、クラウド、トライフィック管理、効率的なデータ転送

# Lab 2

- ネットワーク管理／ストレージ管理
- バックアップとリストア

# オプションモジュール

- 仮想インフラ管理者向け
  - ベアメタル OpenShift
  - ストレージ管理
  - ネットワーク管理
  - バックアップ／リストア

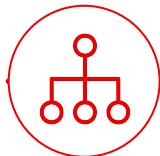


Let's get familiar with the product

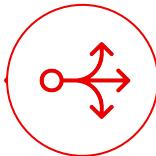
# Q & A

# クロージング

# 今日のワークショップが終わると皆さんは...



仮想インフラストラクチャの  
代替ソリューションを検討する



Red Hat OpenShift Virtualization を知る



インフラモダナイゼーションへの旅が今日から  
始まるることを認識する

アプリケーション・プラットフォームとして  
の Red Hat OpenShiftは、

**コンテナ と サーバー仮想化**

の両方において、

**開発者 と 運用者** を結びつける。



アンケートにご協力お願いします。  
QRコードよりご回答をお願いいたします。



<https://forms.gle/Zdb6wW3fvpJbyBSz9>

# 本日はありがとうございました



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



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



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



[twitter.com/RedHat](https://twitter.com/RedHat)