



# Migration Toolkit for Virtualization 2.2

## Migration Toolkit for Virtualization のインストールおよび使用

VMware vSphere または Red Hat Virtualization から Red Hat OpenShift Virtualization  
への移行



# Migration Toolkit for Virtualization 2.2 Migration Toolkit for Virtualization のインストールおよび使用

---

VMware vSphere または Red Hat Virtualization から Red Hat OpenShift Virtualization への移行

Enter your first name here. Enter your surname here.

Enter your organisation's name here. Enter your organisational division here.

Enter your email address here.

## 法律上の通知

Copyright © 2022 | You need to change the HOLDER entity in the en-US/Installing\_and\_using\_the\_Migration\_Toolkit\_for\_Virtualization.ent file |.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux<sup>®</sup> is the registered trademark of Linus Torvalds in the United States and other countries.

Java<sup>®</sup> is a registered trademark of Oracle and/or its affiliates.

XFS<sup>®</sup> is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL<sup>®</sup> is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js<sup>®</sup> is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack<sup>®</sup> Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

## 概要

Migration Toolkit for Virtualization (MTV) を使用すると、仮想マシンを VMware vSphere または Red Hat Virtualization から OpenShift Container Platform で実行している OpenShift Virtualization に移行できます。

## 目次

多様性を受け入れるオープンソースの強化 .....	4
<b>第1章 MIGRATION TOOLKIT FOR VIRTUALIZATION について .....</b>	<b>5</b>
1.1. コールド移行とウォーム移行 .....	5
1.1.1. コールド移行 .....	5
1.1.2. ウォーム移行 .....	5
<b>第2章 前提条件 .....</b>	<b>7</b>
2.1. ソフトウェア要件 .....	7
2.2. ストレージサポートおよびデフォルトモード .....	7
2.3. ネットワークの前提条件 .....	8
2.3.1. ポート .....	8
2.4. ソース仮想マシンの前提条件 .....	9
2.5. RED HAT VIRTUALIZATION の前提条件 .....	9
2.6. VMWARE の前提条件 .....	9
2.6.1. VDDK イメージの作成 .....	10
2.6.2. vCenter ホストの SHA-1 フィンガープリントの取得 .....	11
2.6.3. ESXi ホストの NFC サービスメモリーの拡大 .....	11
2.7. ソフトウェア互換性ガイドライン .....	12
<b>第3章 MTV OPERATOR のインストール .....</b>	<b>13</b>
3.1. OPENSIFT CONTAINER PLATFORM WEB コンソールを使用した MTV OPERATOR のインストール .....	13
MTV Web コンソール URL の取得 .....	13
3.2. コマンドラインインターフェースからの MTV OPERATOR のインストール .....	14
MTV Web コンソール URL の取得 .....	15
<b>第4章 MTV WEB コンソールを使用した仮想マシンの移行 .....</b>	<b>17</b>
4.1. プロバイダーの追加 .....	17
4.1.1. VMware ソースプロバイダーの追加 .....	17
4.1.1.1. VMware ソースプロバイダーの移行ネットワークの選択 .....	18
4.1.2. Red Hat Virtualization ソースプロバイダーの追加 .....	19
4.1.3. OpenShift Virtualization プロバイダーの追加 .....	19
4.1.3.1. OpenShift Virtualization プロバイダーの移行ネットワークの選択 .....	20
4.2. ネットワークマッピングの作成 .....	20
4.3. ストレージマッピングの作成 .....	21
4.4. 移行計画の作成 .....	22
4.5. 移行計画の実行 .....	23
4.6. 移行計画のオプション .....	24
4.7. 移行のキャンセル .....	25
<b>第5章 コマンドラインからの仮想マシンの移行 .....</b>	<b>26</b>
5.1. 仮想マシンの移行 .....	26
5.2. 移行のキャンセル .....	31
<b>第6章 高度な移行オプション .....</b>	<b>33</b>
6.1. ウォーム移行のプレコピー間隔の変更 .....	33
6.2. VALIDATION サービスのカスタムルールの作成 .....	33
6.2.1. Rego ファイルについて .....	33
6.2.2. デフォルトの検証ルールの確認 .....	34
6.2.3. Inventory サービス JSON の取得 .....	34
6.2.4. 検証ルールの作成 .....	41
6.2.5. インベントリールールバージョンの更新 .....	43

第7章 MIGRATION TOOLKIT FOR VIRTUALIZATION のアップグレード .....	45
第8章 MIGRATION TOOLKIT FOR VIRTUALIZATION のアンインストール .....	47
8.1. OPENSIFT CONTAINER PLATFORM WEB コンソールを使用した MTV のアンインストール	47
8.2. コマンドラインインターフェースからの MTV のアンインストール	47
第9章 トラブルシューティング .....	49
9.1. アーキテクチャー	49
9.1.1. MTV カスタムリソースおよびサービス	49
9.1.2. 移行ワークフローの概要	50
9.1.3. 詳細な移行ワークフロー	50
9.2. ログとカスタムリソース	51
9.2.1. 収集されるログおよびカスタムリソース情報	51
9.2.2. Web コンソールからのログおよびカスタムリソース情報のダウンロード	52
9.2.3. コマンドラインインターフェースからのログおよびカスタムリソース情報へのアクセス	53



## 多様性を受け入れるオープンソースの強化

Red Hat では、コード、ドキュメント、Web プロパティにおける配慮に欠ける用語の置き換えに取り組んでいます。まずは、マスター (master)、スレーブ (slave)、ブラックリスト (blacklist)、ホワイトリスト (whitelist) の 4 つの用語の置き換えから始めます。この取り組みは膨大な作業を要するため、今後の複数のリリースで段階的に用語の置き換えを実施して参ります。詳細は、[弊社の CTO、Chris Wright のメッセージ](#) を参照してください。



# 第1章 MIGRATION TOOLKIT FOR VIRTUALIZATION について

仮想マシンを、project-first を使用して VMware vSphere または Red Hat Virtualization から OpenShift Virtualization with the Migration Toolkit for Virtualization (MTV) に移行できます。

## 関連情報

- [VMware vSphere から OpenShift Virtualization に移行するためのパフォーマンスに関する推奨事項](#)
- [Red Hat Virtualization から OpenShift Virtualization に移行するためのパフォーマンスに関する推奨事項](#)

## 1.1. コールド移行とウォーム移行

MTV は、Red Hat Virtualization からのコールド移行と、VMware vSphere からのウォーム移行をサポートしています。

### 1.1.1. コールド移行

コールド移行は、デフォルトの移行タイプです。ソース仮想マシンは、データのコピー中にシャットダウンします。

### 1.1.2. ウォーム移行

ほとんどのデータは、ソース仮想マシン (VM) の実行中にプレコピー段階でコピーされます。

次に、VM がシャットダウンされ、残りのデータはカットオーバー段階でコピーされます。

#### プレコピー段階

仮想マシンはプレコピー段階ではシャットダウンされません。

仮想マシンディスクは、[変更ブロック追跡 \(CBT\)](#) スナップショットを使用して増分コピーされます。スナップショットは、デフォルトで1時間間隔で作成されます。**forklift-controller** デプロイメントを更新して、スナップショットの間隔を変更できます。



#### 重要

各ソースVMおよび各VMディスクに対してCBTを有効にする必要があります。

仮想マシンは、最大28 CBT スナップショットをサポートします。ソースVMのCBTスナップショットが多すぎて、**Migration Controller** サービスが新規スナップショットを作成できない場合、ウォーム移行に失敗する可能性があります。スナップショットが不要になると、**Migration Controller** サービスは各スナップショットを削除します。

プレコピー段階は、カットオーバー段階を手動で開始するか、開始がスケジュールされるまで実行されます。

#### カットオーバー段階

仮想マシンはカットオーバー段階でシャットダウンされ、残りのデータは移行されます。RAMに保存されるデータは移行されません。

MTV コンソールを使用してカットオーバー段階を手動で開始するか、**Migration** マニフェストでカットオーバー時間をスケジュールできます。

## 第2章 前提条件

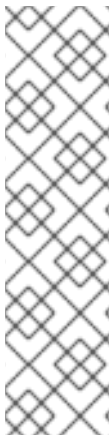
以下の前提条件を確認し、環境が移行用に準備されていることを確認します。

### 2.1. ソフトウェア要件

[互換性のあるバージョン](#)の OpenShift Container Platform および OpenShift Virtualization をインストールする必要があります。

### 2.2. ストレージサポートおよびデフォルトモード

MTV は、サポートされているストレージに以下のデフォルトのボリュームおよびアクセスモードを使用します。



#### 注記

OpenShift Virtualization ストレージが [動的プロビジョニング](#) をサポートしない場合、MTV はデフォルト設定を適用します。

- **Filesystem** のボリュームモード  
**Filesystem** ボリュームモードは、**Block** ボリュームモードよりも遅くなります。
- **ReadWriteOnce** アクセスモード  
**ReadWriteOnce** アクセスモードは、ライブ仮想マシンの移行をサポートしません。

表2.1 デフォルトのボリュームおよびアクセスモード

プロビジョナー	ボリュームモード	アクセスモード
kubernetes.io/aws-ebs	Block	ReadWriteOnce
kubernetes.io/azure-disk	Block	ReadWriteOnce
kubernetes.io/azure-file	Filesystem	ReadWriteMany
kubernetes.io/cinder	Block	ReadWriteOnce
kubernetes.io/gce-pd	Block	ReadWriteOnce
kubernetes.io/hostpath-provisioner	Filesystem	ReadWriteOnce
manila.csi.openstack.org	Filesystem	ReadWriteMany
openshift-storage.cephfs.csi.ceph.com	Filesystem	ReadWriteMany

プロビジョナー	ボリュームモード	アクセスモード
openshift-storage.rbd.csi.ceph.com	Block	ReadWriteOnce
kubernetes.io/rbd	Block	ReadWriteOnce
kubernetes.io/vsphere-volume	Block	ReadWriteOnce

## 2.3. ネットワークの前提条件

以下の前提条件はすべての移行に適用されます。

- IP アドレス、VLAN、およびその他のネットワーク設定は、移行前後で変更できません。仮想マシンの MAC アドレスは、移行時に保持されます。
- ソース環境、OpenShift Virtualization クラスタ、およびレプリケーションリポジトリ間のネットワーク接続は、信頼性と中断された状態でなければなりません。
- 複数の送信元ネットワークと宛先ネットワークをマッピングする場合は、追加の宛先ネットワークごとに [ネットワーク接続定義](#) を作成する必要があります。

### 2.3.1. ポート

ファイアウォールは、以下のポートでトラフィックを有効にする必要があります。

表2.2 VMware vSphere からの移行に必要なネットワークポート

ポート	プロトコル	ソース	宛先	目的
443	TCP	OpenShift ノード	VMware vCenter	VMware プロバイダーインベントリー ディスク転送認証
443	TCP	OpenShift ノード	VMware ESXi ホスト	ディスク転送認証
902	TCP	OpenShift ノード	VMware ESXi ホスト	ディスク転送データのコピー

表2.3 Red Hat Virtualization からの移行に必要なネットワークポート

ポート	プロトコル	ソース	宛先	目的
443	TCP	OpenShift ノード	RHV Engine	RHV プロバイダーインベントリー ディスク転送認証

ポート	プロトコル	ソース	宛先	目的
443	TCP	OpenShift ノード	RHV ホスト	ディスク転送認証
54322	TCP	OpenShift ノード	RHV ホスト	ディスク転送データのコピー

## 2.4. ソース仮想マシンの前提条件

以下の前提条件はすべての移行に適用されます。

- ISO/CDROM ディスクをアンマウントする必要があります。
- 各 NIC には、IPv4 アドレス および IPv6 アドレスが1つずつ、またはいずれか一方が1つ含まれている必要があります。
- 仮想マシン名には、小文字 (**a-z**)、数字 (**0-9**)、またはハイフン (-) のみが含まれ、最大 253 文字にする必要があります。最初と最後の文字は英数字にする必要があります。この名前には、大文字、スペース、ピリオド (.)、または特殊文字を使用できません。
- 仮想マシン名は、OpenShift Virtualization 環境の仮想マシンの名前と重複させてはなりません。
- 仮想マシンオペレーティングシステムは、[OpenShift Virtualization のゲストオペレーティングシステム](#) としての使用 および [virt-v2v](#)での KVM への変換 に対して認定およびサポートされません。

## 2.5. RED HAT VIRTUALIZATION の前提条件

以下の前提条件は Red Hat Virtualization の移行に適用されます。

- [互換性のあるバージョン](#) の Red Hat Virtualization を使用する必要があります。
- Manager の CA 証明書が必要です。  
ブラウザで [https://<engine\\_host>/ovirt-engine/services/pki-resource?resource=ca-certificate&format=X509-PEM-CA](https://<engine_host>/ovirt-engine/services/pki-resource?resource=ca-certificate&format=X509-PEM-CA) に移動して、CA 証明書を取得できます。

## 2.6. VMWARE の前提条件

以下の前提条件は VMware の移行に適用されます。

- [互換性のあるバージョン](#) の VMware vSphere を使用する必要があります。
- [VMware ツール](#) をすべてのソース仮想マシン (VM) にインストールする必要があります。
- ウォーム移行を実行している場合は、仮想マシンおよび仮想マシンディスクで [変更ブロック追跡 \(CBT\)](#) を有効にする必要がある。
- VMware Virtual Disk Development Kit (VDDK) イメージを作成する必要があります。
- vCenter ホストの SHA-1 フィンガープリントを取得する必要があります。

- 同じ移行プランの ESXi ホストから 10 台を超える仮想マシンを移行する場合は、ホストの NFC サービスメモリーを増やす必要があります。

### 2.6.1. VDDK イメージの作成

Migration Toolkit for Virtualization (MTV) は、VMware Virtual Disk Development Kit (VDDK) SDK を使用して、VMware vSphere から仮想ディスクを転送します。

VMware Virtual Disk Development Kit (VDDK) をダウンロードして、VDDK イメージをビルドし、VDDK イメージをイメージレジストリーにプッシュする必要があります。その後、VDDK イメージを **HyperConverged** カスタムリソース (CR) に追加します。



#### 注記

VDDK イメージをパブリックレジストリーに保存すると、VMware ライセンスの条件に違反する可能性があります。

#### 前提条件

- [OpenShift Container Platform イメージレジストリー](#)
- **podman** がインストールされている。
- 外部レジストリーを使用している場合、OpenShift Virtualization はこれにアクセスできる必要がある。

#### 手順

1. 一時ディレクトリーを作成し、これに移動します。

```
$ mkdir /tmp/<dir_name> && cd /tmp/<dir_name>
```

2. ブラウザーで、[VMware VDDK ダウンロードページ](#) に移動します。
3. 最新の VDDK バージョンを選択し、**Download** をクリックします。
4. VDDK アーカイブファイルを一時ディレクトリーに保存します。
5. VDDK アーカイブを展開します。

```
$ tar -xzf VMware-vix-disklib-<version>.x86_64.tar.gz
```

6. **Dockerfile** を作成します。

```
$ cat > Dockerfile <<EOF
FROM registry.access.redhat.com/ubi8/ubi-minimal
COPY vmware-vix-disklib-distrib /vmware-vix-disklib-distrib
RUN mkdir -p /opt
ENTRYPOINT ["cp", "-r", "/vmware-vix-disklib-distrib", "/opt"]
EOF
```

7. VDDK イメージをビルドします。

```
$ podman build . -t <registry_route_or_server_path>/vddk:<tag>
```

- VDDK イメージをレジストリーにプッシュします。

```
$ podman push <registry_route_or_server_path>/vddk:<tag>
```

- イメージが OpenShift Virtualization 環境からアクセスできることを確認します。

## 2.6.2. vCenter ホストの SHA-1 フィンガープリントの取得

**Secret** CR を作成するには、vCenter ホストの SHA-1 フィンガープリントを取得する必要があります。

### 手順

- 以下のコマンドを実行します。

```
$ openssl s_client \
  -connect <vcenter_host>:443 \ ①
  </dev/null 2>/dev/null \
  | openssl x509 -fingerprint -noout -in /dev/stdin \
  | cut -d '=' -f 2
```

- vCenter ホストの IP アドレスまたは FQDN を指定します。

### 出力例

```
01:23:45:67:89:AB:CD:EF:01:23:45:67:89:AB:CD:EF:01:23:45:67
```

## 2.6.3. ESXi ホストの NFC サービスメモリーの拡大

同じ移行プランの ESXi ホストから 10 台を超える仮想マシンを移行する場合は、ホストの NFC サービスメモリーを増やす必要があります。有効にしない場合、NFC サービスメモリーの同時接続は 10 台に制限されているため、移行に失敗します。

### 手順

- ESXi ホストに root としてログインします。
- `/etc/vmware/hostd/config.xml` で `maxMemory` の値を **1000000000** に変更します。

```
...
<nfcsvc>
  <path>libnfcsvc.so</path>
  <enabled>true</enabled>
  <maxMemory>1000000000</maxMemory>
  <maxStreamMemory>10485760</maxStreamMemory>
</nfcsvc>
...
```

- `hostd` を再起動します。

```
# /etc/init.d/hostd restart
```

ホストを再起動する必要はありません。

## 2.7. ソフトウェア互換性ガイドライン

互換性のあるソフトウェアバージョンをインストールする必要があります。

表2.4 互換性のあるソフトウェアバージョン

Migration Toolkit for Virtualization	OpenShift Container Platform	OpenShift Virtualization	VMware vSphere	Red Hat Virtualization
2.2	4.9	4.9.1	6.5 以降	4.3 以降



## 第3章 MTV OPERATOR のインストール

OpenShift Container Platform Web コンソールまたはコマンドラインインターフェース (CLI) を使用して MTV Operator をインストールできます。

### 3.1. OPENSIFT CONTAINER PLATFORM WEB コンソールを使用した MTV OPERATOR のインストール

OpenShift Container Platform Web コンソールを使用して MTV Operator をインストールできます。

#### 前提条件

- OpenShift Container Platform 4.9 がインストールされている。
- OpenShift Virtualization Operator がインストールされている。
- **cluster-admin** パーミッションを持つユーザーとしてログインしている必要があります。

#### 手順

1. OpenShift Container Platform Web コンソールで、**Operators** → **OperatorHub** の順にクリックします。
2. **Filter by keyword** フィールドを使用して **mtv-operator** を検索します。
3. **Migration Toolkit for Virtualization Operator** をクリックしてから **Install** をクリックします。
4. **Install Operator** ページで、**Install** をクリックします。
5. **Operators** → **Installed Operators** をクリックして、**Migration Toolkit for Virtualization Operator** が **Succeeded** のステータスで **openshift-mtv** プロジェクトに表示されることを確認します。
6. **Migration Toolkit for Virtualization Operator** をクリックします。
7. **Provided APIs** で **ForkliftController** を見つけ、**Create Instance** をクリックします。
8. **作成** をクリックします。
9. **Workloads** → **Pods** の順にクリックし、MTV Pod が実行されていることを確認します。

#### MTV Web コンソール URL の取得

OpenShift Container Platform Web コンソールを使用して MTV Web コンソール URL を取得できません。

#### 前提条件

- OpenShift Virtualization Operator がインストールされていること。
- MTV Operator がインストールされていること。
- **cluster-admin** 権限を持つユーザーとしてログインしている必要があります。

#### 手順

1. OpenShift Container Platform Web コンソールにログインします。
2. **Networking** → **Routes** の順にクリックします。
3. **Project**: 一覧で **openshift-mtv** プロジェクトを選択します。
4. **forklift-ui** サービスの URL をクリックして、MTV Web コンソールのログインページを開きます。

## 3.2. コマンドラインインターフェースからの MTV OPERATOR のインストール

コマンドラインインターフェース (CLI) から MTV Operator をインストールできます。

### 前提条件

- OpenShift Container Platform 4.9 がインストールされている。
- OpenShift Virtualization Operator がインストールされている。
- **cluster-admin** パーミッションを持つユーザーとしてログインしている必要があります。

### 手順

1. openshift-mtv プロジェクトを作成します。

```
$ cat << EOF | oc apply -f -
apiVersion: project.openshift.io/v1
kind: Project
metadata:
  name: openshift-mtv
EOF
```

2. **migration** という **OperatorGroup** CR を作成します。

```
$ cat << EOF | oc apply -f -
apiVersion: operators.coreos.com/v1
kind: OperatorGroup
metadata:
  name: migration
  namespace: openshift-mtv
spec:
  targetNamespaces:
    - openshift-mtv
EOF
```

3. Operator の **Subscription** CR を作成します。

```
$ cat << EOF | oc apply -f -
apiVersion: operators.coreos.com/v1alpha1
kind: Subscription
metadata:
  name: mtv-operator
  namespace: openshift-mtv
```

```
spec:
  channel: release-v2.2.0
  installPlanApproval: Automatic
  name: mtv-operator
  source: redhat-operators
  sourceNamespace: openshift-marketplace
  startingCSV: "mtv-operator.2.2.0"
EOF
```

4. **ForkliftController** CR を作成します。

```
$ cat << EOF | oc apply -f -
apiVersion: forklift.konveyor.io/v1beta1
kind: ForkliftController
metadata:
  name: forklift-controller
  namespace: openshift-mtv
spec:
  olm_managed: true
EOF
```

5. MTV Pod が実行中であることを検証します。

```
$ oc get pods -n openshift-mtv
```

#### 出力例

```
NAME                                READY STATUS RESTARTS AGE
forklift-controller-788bdb4c69-mw268 2/2   Running 0      2m
forklift-operator-6bf45b8d8-qps9v    1/1   Running 0      5m
forklift-ui-7cdf96d8f6-xnw5n        1/1   Running 0      2m
```

### MTV Web コンソール URL の取得

コマンドラインから MTV Web コンソール URL を取得できます。

#### 前提条件

- OpenShift Virtualization Operator がインストールされていること。
- MTV Operator がインストールされていること。
- **cluster-admin** 権限を持つユーザーとしてログインしている必要があります。

#### 手順

1. MTV Web コンソール URL を取得します。

```
$ oc get route virt -n openshift-mtv \
-o custom-columns=:.spec.host
```

#### 出力例

```
https://virt-openshift-mtv.apps.cluster.openshift.com.
```

2. ブラウザーを起動し、MTV Web コンソールに移動します。

## 第4章 MTV WEB コンソールを使用した仮想マシンの移行

MTV Web コンソールを使用して、仮想マシン (VM) を OpenShift Virtualization に移行できます。



### 重要

すべての [前提条件](#) を満たしていることを確認する必要があります。

## 4.1. プロバイダーの追加

MTV Web コンソールを使用してプロバイダーを追加できます。

### 4.1.1. VMware ソースプロバイダーの追加

MTV Web コンソールを使用して VMware ソースプロバイダーを追加できます。

#### 前提条件

- VMware 管理者権限が必要です。
- vCenter SHA-1 フィンガープリント。
- すべてのクラスターがアクセスできるセキュアなレジストリーでの VMware Virtual Disk Development Kit (VDDK) イメージ。

#### 手順

1. VDDK イメージを **HyperConverged** CR に追加します。

```
$ cat << EOF | oc apply -f -
apiVersion: hco.kubevirt.io/v1beta1
kind: HyperConverged
metadata:
  name: kubevirt-hyperconverged
  namespace: openshift-cnv
spec:
  vddkInitImage: <registry_route_or_server_path>/vddk:<tag> 1
EOF
```

- 1 作成した VDDK イメージを指定します。
2. MTV Web コンソールで、**Providers** をクリックします。
3. **Add provider** をクリックします。
4. **タイプ** 一覧から **VMware** を選択します。
5. 以下のフィールドに入力します。
  - **名前**: プロバイダー一覧に表示する名前
  - **ホスト名または IP アドレス**: vCenter ホスト名または IP アドレス

- **ユーザー名:** vCenter 管理ユーザー (例: **administrator@vsphere.local**)
  - **Password:** vCenter 管理パスワード
  - **SHA-1フィンガープリント:** vCenter SHA-1 フィンガープリント
6. **Add** をクリックしてプロバイダーを追加し、保存します。  
ソースプロバイダーはプロバイダーのリストに表示されます。

#### 4.1.1.1. VMware ソースプロバイダーの移行ネットワークの選択

VMware ソースプロバイダーの MTV Web コンソールで移行ネットワークを選択して、ソース環境へのリスクを低減し、パフォーマンスを向上させることができます。

移行に管理ネットワークを使用すると、ネットワークに十分な帯域幅がないためにパフォーマンスが低下する可能性があります。この状況は、ディスク転送操作がネットワークを飽和状態にし、ソースプラットフォームで悪影響を及ぼす可能性があります。

#### 前提条件

- 移行ネットワークには、ディスク転送用に十分なスループット (最低速度 10 Gbps) が必要である。
- 移行ネットワークは、デフォルトゲートウェイを介して OpenShift Virtualization ノードにアクセスできる必要がある。



#### 注記

ソース仮想ディスクは、ターゲット namespace の Pod ネットワークに接続されている Pod によってコピーされます。

- 移行ネットワークでは、ジャンボフレームを有効にする必要がある。

#### 手順

1. MTV Web コンソールで、**プロバイダー** をクリックします。
2. **VMware** タブをクリックします。
3. プロバイダーの横にある **ホスト** 列のホスト番号をクリックし、ホストの一覧を表示します。
4. 1つ以上のホストを選択し、**移行ネットワークの選択** をクリックします。
5. **ネットワーク** を選択します。  
デフォルトネットワークを選択して、選択を解除できます。
6. ソースプロバイダーが VMware の場合は、以下のフィールドを実行します。
  - **ESXi host admin username:** ESXi ホスト管理ユーザー名を指定します (例: **root**)。
  - **ESXi host admin password:** ESXi ホストの admin パスワードを指定します。
7. ソースプロバイダーが Red Hat Virtualization の場合は、以下のフィールドを実行してください。
  - **Username:** Manager ユーザーを指定します。

- **Password:** Manager のパスワードを指定します。
8. **保存** をクリックします。
  9. 各ホストのステータスが **Ready** であることを確認します。  
ホストのステータスが **Ready** ではない場合は、移行ネットワーク上でホストにアクセスできないか、認証情報が正しくない可能性があります。ホスト設定を変更し、変更を保存することができます。

### 4.1.2. Red Hat Virtualization ソースプロバイダーの追加

MTV Web コンソールを使用して Red Hat Virtualization ソースプロバイダーを追加できます。

#### 前提条件

- Manager の CA 証明書。

#### 手順

1. MTV Web コンソールで、**Providers** をクリックします。
2. **Add provider** をクリックします。
3. **タイプ** の一覧から **Red Hat Virtualization** を選択します。
4. 以下のフィールドに入力します。
  - **名前:** プロバイダー一覧に表示する名前
  - **ホスト名または IP アドレス:** Manager ホスト名または IP アドレス
  - **ユーザー名:** Manager ユーザー
  - **パスワード:** Manager パスワード
  - **CA 証明書:** Manager の CA 証明書
5. **Add** をクリックしてプロバイダーを追加し、保存します。  
ソースプロバイダーはプロバイダーのリストに表示されます。

### 4.1.3. OpenShift Virtualization プロバイダーの追加

MTV をインストールしたプロバイダーであるデフォルトの OpenShift Virtualization プロバイダーに加えて、OpenShift Virtualization プロバイダーを MTV Web コンソールに追加できます。

#### 前提条件

- **cluster-admin** 権限を持つ OpenShift Virtualization [サービスアカウントトークン](#) が必要です。

#### 手順

1. MTV Web コンソールで、**Providers** をクリックします。
2. **Add provider** をクリックします。

3. **Type** 一覧から **OpenShift Virtualization** を選択します。
4. 以下のフィールドに入力します。
  - **Cluster name**: ターゲットプロバイダーの一覧で表示するクラスター名を指定します。
  - **URL**: クラスターの API エンドポイントを指定します。
  - **Service account token cluster-admin** サービスアカウントトークンを指定します。
5. **Check connection** をクリックして認証情報を確認します。
6. **Add** をクリックします。  
プロバイダーがプロバイダーのリストに表示されます。

#### 4.1.3.1. OpenShift Virtualization プロバイダーの移行ネットワークの選択

MTV Web コンソールで OpenShift Virtualization プロバイダーのデフォルト移行ネットワークを選択して、パフォーマンスを向上させることができます。デフォルトの移行ネットワークは、ディスクが設定された namespace にディスクを転送するために使用されます。

移行ネットワークを選択しない場合、デフォルトの移行ネットワークは **pod** ネットワークで、ディスク転送に最適ではない可能性があります。



#### 注記

移行計画の作成時に別のネットワークを選択して、プロバイダーのデフォルト移行ネットワークを上書きすることができます。

#### 手順

1. MTV Web コンソールで、**Providers** をクリックします。
2. **OpenShift Virtualization** タブをクリックします。
3. プロバイダーを選択し、**移行ネットワークの選択** をクリックします。
4. 利用可能なネットワークの一覧からネットワークを選択し、**選択** をクリックします。
5. プロバイダーの横にある **ネットワーク** 列のネットワーク番号をクリックし、選択したネットワークがデフォルトの移行ネットワークであることを確認します。

## 4.2. ネットワークマッピングの作成

MTV Web コンソールを使用して、ソースネットワークを OpenShift Virtualization ネットワークにマッピングすることで、1つ以上のネットワークマッピングを作成できます。

#### 前提条件

- Web コンソールにソースプロバイダーおよびターゲットプロバイダーが追加されている。
- 複数のソースおよびターゲットネットワークをマッピングする場合、追加の OpenShift Virtualization ネットワークごとに独自の **ネットワーク割り当て定義** が必要です。

#### 手順



1. **Mappings** をクリックします。
2. **Network** タブをクリックしてから、**Create mapping** をクリックします。
3. 以下のフィールドに入力します。
  - **Name**: ネットワークマッピング一覧に表示する名前を入力します。
  - **Source provider**: ソースプロバイダーを選択します。
  - **Target provider**: ターゲットプロバイダーを選択します。
  - **Source networks**: ソースネットワークを選択します。
  - **ターゲット namespaces/ネットワーク**: ターゲットネットワークを選択します。
4. オプション: **Add** をクリックして追加のネットワークマッピングを作成するか、複数のソースネットワークを単一のターゲットネットワークにマッピングします。
5. 追加ネットワークのマッピングを作成する場合は、ネットワーク接続定義をターゲットネットワークとして選択します。
6. **作成** をクリックします。  
ネットワークマッピングは、**Network mappings** 画面に表示されます。

### 4.3. ストレージマッピングの作成

MTV Web コンソールを使用して、ソースデータストアを OpenShift Virtualization ストレージクラスにマッピングすることで、ストレージマッピングを作成できます。

#### 前提条件

- Web コンソールにソースプロバイダーおよびターゲットプロバイダーが追加されている。
- 仮想マシンの移行をサポートするローカルおよび共有の永続ストレージ。

#### 手順

1. **Mappings** をクリックします。
2. **Storage** タブをクリックし、**Create mapping** をクリックします。
3. ストレージマッピングの **Name** を入力します。
4. **Source provider** および **Target provider** を選択します。
5. ソースプロバイダーが VMware の場合は、**Source datastore** および **Target storage class** を選択します。
6. ソースプロバイダーが Red Hat Virtualization の場合は、**Source storage domain** および **Target storage class** を選択します。
7. オプション: **Add** をクリックして追加のストレージマッピングを作成するか、複数のデータストアまたはストレージドメインを単一のストレージクラスにマッピングします。
8. **作成** をクリックします。

マッピングは **ストレージマッピング** ページに表示されます。

## 4.4. 移行計画の作成

MTV Web コンソールを使用して移行計画を作成できます。

移行計画により、一緒に移行する仮想マシンまたは同じ移行パラメーターの仮想マシン (一定の割合のクラスターのメンバーやアプリケーション全体など) をグループ化できます。

移行計画の指定された段階で Ansible Playbook またはカスタムコンテナイメージを実行するようにフックを設定できます。

### 前提条件

- MTV がターゲットクラスターにインストールされていない場合は、Web コンソールの **Providers** ページでターゲットプロバイダーを追加する必要があります。

### 手順


1. Web コンソールで、**Migration plans** をクリックしてから **Create migration plan** をクリックします。
2. 以下のフィールドに入力します。
  - **Plan name:** 移行計画一覧に表示する移行計画名を入力します。
  - **Plan description:** オプション: 移行計画の簡単な説明。
  - **Source provider:** ソースプロバイダーを選択します。
  - **Target provider:** ターゲットプロバイダーを選択します。
  - **ターゲット namespace:** 既存のターゲット namespace 入力して検索するか、新しい namespace を作成することができます。
  - **Select a different network** をクリックして、一覧からネットワークを選択し、**Select** をクリックして、この計画の移行転送ネットワークを変更できます。  
OpenShift Virtualization プロバイダーの移行転送ネットワークを定義しており、ネットワークがターゲット namespace にある場合、そのネットワークはすべての移行計画のデフォルトネットワークになります。それ以外の場合には、**pod** ネットワークが使用されず。
3. **次へ** をクリックします。
4. ソース仮想マシンのリストをフィルタリングするオプションを選択し、**Next** をクリックします。
5. 移行する仮想マシンを選択し、**次へ** をクリックします。
6. 既存のネットワークマッピングを選択するか、新しいネットワークマッピングを作成します。新規のネットワークマッピングを作成するには、以下の手順を実施します。
  - ソースネットワークごとに、ターゲットネットワークを選択します。
  - オプション: **マッピングを保存して再度使用する** を選択し、ネットワークマッピング名を入力します。

7. **次へ** をクリックします。
8. 既存のストレージマッピングを選択するか、新しいストレージマッピングを作成します。  
新規のストレージマッピングを作成するには、以下を実行します。
  - VMware データストアまたは Red Hat Virtualization ストレージドメインにターゲットストレージクラスを選択します。
  - オプション: **マッピングを保存して再度使用する** を選択し、ストレージマッピング名を入力します。
9. **次へ** をクリックします。
10. 移行のタイプを選択し、**Next** をクリックします。
  - コールド移行: データのコピー中にソースの仮想マシンが停止します。
  - ウォーム移行: データが増分コピーされる間にソース仮想マシンが実行されます。後に、仮想マシンを停止し、残りの仮想マシンデータとメタデータをコピーするカットオーバーを実行します。ウォーム移行は、Red Hat Virtualization ではサポートされません。
11. オプション: 移行フックを作成して、移行前または移行後に Ansible Playbook を実行できます。
  - a. **Add hook** をクリックします。
  - b. フックの実行時のステップを選択します。
  - c. フック定義を選択します。
    - **Ansible Playbook**: Ansible Playbook を参照するか、フィールドに貼り付けます。
    - **カスタムコンテナイメージ**: デフォルトの **hook-runner** イメージを使用しない場合は、イメージパス **<registry\_path>/<image\_name>:<tag>** を入力します。



#### 注記

レジストリーは OpenShift Container Platform クラスタからアクセスできる必要がある。

12. **次へ** をクリックします。
13. 移行計画を確認し、**Finish** をクリックします。  
移行計画は移行計画の一覧に保存されます。
14. 移行計画の Options メニュー  をクリックし、**View details** を選択して移行計画の詳細を確認します。

## 4.5. 移行計画の実行

MTV Web コンソールで移行計画を実行し、進捗を表示できます。


### 前提条件

- 有効な移行計画。

## 手順

1. **移行計画** をクリックします。  
Migration plans リストには、ソースプロバイダーとターゲットプロバイダー、移行する仮想マシン (VM) の数、および計画のステータスが表示されます。
2. 移行計画の横にある **Start** をクリックして移行を開始します。  
ウォーム移行のみ:
  - プレコピー段階が開始されます。
  - **Cutover** をクリックして移行を完了します。
3. 移行計画を展開して、移行の詳細を表示します。  
移行の詳細画面では、移行の開始時間、終了時間、コピーされたデータ量、移行中の各仮想マシンの進捗パイプラインが表示されます。
4. VM を展開して、移行ステップ、各ステップの経過時間、およびその状態を表示します。

## 4.6. 移行計画のオプション

MTV Web コンソールの **Migration plans** ページで、移行計画の横にある Options メニュー  をクリックして以下のオプションにアクセスできます。

- **Edit**: 移行計画の詳細を編集します。移行計画の実行中または正常に完了した後は、移行計画を編集できません。
- **Duplicate**: 既存の計画と同じ仮想マシン (VM)、パラメーター、マッピング、およびフックを使用して、新しい移行計画を作成します。この機能は、以下のタスクに使用できます。
  - VM を別の名前空間に移行します。
  - アーカイブされた移行計画を編集します。
  - 移行計画を別のステータスで編集します (例: 失敗、キャンセル、実行中、クリティカル、または準備完了)。
- **Archive**: 移行計画のログ、履歴、メタデータを削除します。計画を編集または再起動することはできません。閲覧のみ可能です。



## 注記

**Archive** オプションは元に戻せません。ただし、アーカイブされた計画を複製することはできます。

- **Delete**: 移行計画を完全に削除します。実行中の移行計画を削除することはできません。



## 注記

**Delete** オプションは元に戻せません。

移行計画を削除しても、**importer** Pod、**conversion** Pod、設定マップ、シークレット、失敗した VM、データボリュームなどの一時的なリソースは削除されません。(BZ#2018974) 一時的なリソースをクリーンアップするために、移行プランを削除する前にアーカイブする必要があります。

- **View details:** 移行計画の詳細を表示します。
- **Restart:** 失敗またはキャンセルした移行計画を再起動します。
- **Cancel scheduled cutover:** ウォーム移行計画に対してスケジュールされたカットオーバー移行をキャンセルします。

## 4.7. 移行のキャンセル

MTV Web コンソールを使用して、移行計画が進行中に、一部またはすべての仮想マシン (VM) の移行をキャンセルすることができます。

### 手順

1. **移行計画** をクリックします。
2. 実行中の移行プランの名前をクリックし、移行の詳細を表示します。
3. 1つ以上の仮想マシンを選択し、**Cancel** をクリックします。
4. **Yes, cancel** をクリックして、キャンセルを確定します。  
**Migration details by VM** 一覧では、キャンセルした仮想マシンのステータスは **Canceled** になります。移行していない仮想マシンと移行済みの仮想マシンへの影響はありません。

**移行計画** ページの移行計画の横にある **Restart** をクリックして、キャンセルした移行を再開できます。

## 第5章 コマンドラインからの仮想マシンの移行

コマンドラインから仮想マシンを OpenShift Virtualization に移行できます。



### 重要

すべての [前提条件](#) を満たしていることを確認する必要があります。

### 5.1. 仮想マシンの移行

MTV カスタムリソース (CR) を作成して、仮想マシン (VM) をコマンドライン (CLI) から移行します。



### 重要

クラスタースコープの CR の名前を指定する必要があります。

namespace スコープの CR の名前と namespace の両方を指定する必要があります。

#### 前提条件

- **cluster-admin** 権限を持つユーザーとしてログインしている。
- VMware のみ: すべてのクラスターがアクセスできるセキュアなレジストリーに VMware Virtual Disk Development Kit (VDDK) イメージを用意しておく。

#### 手順

1. VMware のみ: VDDK イメージを **HyperConverged** CR に追加します。

```
$ cat << EOF | oc apply -f -
apiVersion: hco.kubevirt.io/v1beta1
kind: HyperConverged
metadata:
  name: kubevirt-hyperconverged
  namespace: openshift-cnv
spec:
  vddkInitImage: <registry_route_or_server_path>/vddk:<tag> 1
EOF
```

- 1 作成した VDDK イメージを指定します。

2. ソースプロバイダーの認証情報の **Secret** マニフェストを作成します。

```
$ cat << EOF | oc apply -f -
apiVersion: v1
kind: Secret
metadata:
  name: <secret>
  namespace: openshift-mtv
type: Opaque
stringData:
  user: <user> 1
  password: <password> 2
```

```
cacert: <engine_ca_certificate> 3
thumbprint: <vcenter_fingerprint> 4
EOF
```

- 1 base64 でエンコードされた vCenter 管理ユーザーまたは RHV Manager ユーザーを指定します。
- 2 base64 でエンコードされたパスワードを指定します。
- 3 RHV のみ: Manager の base64 でエンコードされた CA 証明書を指定します。これは [https://<engine\\_host>/ovirt-engine/services/pki-resource?resource=ca-certificate&format=X509-PEM-CA](https://<engine_host>/ovirt-engine/services/pki-resource?resource=ca-certificate&format=X509-PEM-CA) で取得できます。
- 4 VMware のみ: vCenter SHA-1 フィンガープリントを指定します。

### 3. ソースプロバイダーの **Provider** マニフェストを作成します。

```
$ cat << EOF | oc apply -f -
apiVersion: forklift.konveyor.io/v1beta1
kind: Provider
metadata:
  name: <provider>
  namespace: openshift-mtv
spec:
  type: <provider_type> 1
  url: <api_end_point> 2
  secret:
    name: <secret> 3
    namespace: openshift-mtv
EOF
```

- 1 使用できる値は **ovirt** および **vsphere** です。
- 2 API エンドポイントの URL を指定します (vSphere の場合は [https://<vCenter\\_host>/sdk](https://<vCenter_host>/sdk)、RHV の場合は [https://<engine\\_host>/ovirt-engine/api/](https://<engine_host>/ovirt-engine/api/))。
- 3 プロバイダー **Secret** CR の名前を指定します。

### 4. VMware のみ: **Host** マニフェストを作成します。

```
$ cat << EOF | oc apply -f -
apiVersion: forklift.konveyor.io/v1beta1
kind: Host
metadata:
  name: <vmware_host>
  namespace: openshift-mtv
spec:
  provider:
    namespace: openshift-mtv
    name: <source_provider> 1
```

```
id: <source_host_mor> 2
ipAddress: <source_network_ip> 3
EOF
```

- 1 VMware の **Provider** CR の名前を指定します。
- 2 VMware ホストの 管理対象オブジェクト参照 (MOR) を指定します。
- 3 VMware 移行ネットワークの IP アドレスを指定します。

5. 移行元および宛先ネットワークをマッピングする **NetworkMap** マニフェストを作成します。

```
$ cat << EOF | oc apply -f -
apiVersion: forklift.konveyor.io/v1beta1
kind: NetworkMap
metadata:
  name: <network_map>
  namespace: openshift-mtv
spec:
  map:
    - destination:
      name: <pod>
      namespace: openshift-mtv
      type: pod 1
      source: 2
        id: <source_network_id> 3
        name: <source_network_name>
    - destination:
      name: <network_attachment_definition> 4
      namespace: <network_attachment_definition_namespace> 5
      type: multus
      source:
        id: <source_network_id>
        name: <source_network_name>
  provider:
    source:
      name: <source_provider>
      namespace: openshift-mtv
    destination:
      name: <destination_cluster>
      namespace: openshift-mtv
EOF
```

- 1 使用できる値は **Pod** および **multus** です。
- 2 **id** パラメーター または **name** パラメーターのいずれかを使用して、ソースネットワークを指定することができます。
- 3 VMware ネットワーク MOR または RHV ネットワークの UUID を指定します。
- 4 追加の OpenShift Virtualization ネットワークごとにネットワーク接続定義を指定します。
- 5 OpenShift Virtualization ネットワーク割り当て定義の namespace を指定します。



6. **StorageMap** マニフェストを作成し、ソースおよび宛先ストレージをマッピングします。

```
$ cat << EOF | oc apply -f -
apiVersion: forklift.konveyor.io/v1beta1
kind: StorageMap
metadata:
  name: <storage_map>
  namespace: openshift-mtv
spec:
  map:
    - destination:
      storageClass: <storage_class>
      accessMode: <access_mode> ❶
      source:
        id: <source_datastore> ❷
    - destination:
      storageClass: <storage_class>
      accessMode: <access_mode>
      source:
        id: <source_datastore>
  provider:
    source:
      name: <source_provider>
      namespace: openshift-mtv
    destination:
      name: <destination_cluster>
      namespace: openshift-mtv
EOF
```

- ❶ 使用できる値は **ReadWriteOnce** および **ReadWriteMany** です。
- ❷ VMware データストレージ MOR または RHV ストレージドメインの UUID を指定します (例: **f2737930-b567-451a-9ceb-2887f6207009**)。

7. オプション: **Hook** マニフェストを作成し、**Plan** CR で指定されたフェーズ中に VM でカスタムコードを実行します。

```
$ cat << EOF | oc apply -f -
apiVersion: forklift.konveyor.io/v1beta1
kind: Hook
metadata:
  name: <hook>
  namespace: openshift-mtv
spec:
  image: quay.io/konveyor/hook-runner ❶
  playbook: | ❷

LS0tCi0gYmFtZTogTWFFbGogIGhvc3RzOiBsb2NhbGhvc3QKICB0YXNrczoKICAtIG5hbWU6I
Exv

YWQgUGxhbgogICAgW5jbHVkZV92YXJzOgogICAgICBmaWxIOiAiL3RtcC9ob29rL3BsYW4u
eW1s

lgogICAgICBuYW11OiBwbGFuCiAgLSBuYW11OiBMb2FkIFdvcmtsbn2FkCiAgICBpbmNsdWRlX
```

```
3Zh
```

```
cnM6CiAgICAgIGZpbGU6IClvdG1wL2hvb2svd29ya2xvYWQueW1slgogICAgICBuYW1lOiB3b
3Jr
  bG9hZAoK
EOF
```

- 1 デフォルトの **hook-runner** イメージを使用するか、カスタムイメージを指定することができます。カスタムイメージを指定する場合は、Playbook を指定する必要はありません。
- 2 オプション: base64 でエンコードされた Ansible Playbook。Playbook を指定する場合は、**image** は **hook-runner** である必要があります。

## 8. 移行の **Plan** マニフェストを作成します。

```
$ cat << EOF | oc apply -f -
apiVersion: forklift.konveyor.io/v1beta1
kind: Plan
metadata:
  name: <plan> 1
  namespace: openshift-mtv
spec:
  warm: true 2
  provider:
    source:
      name: <source_provider>
      namespace: openshift-mtv
    destination:
      name: <destination_cluster>
      namespace: openshift-mtv
  map:
    network: 3
      name: <network_map> 4
      namespace: openshift-mtv
    storage:
      name: <storage_map> 5
      namespace: openshift-mtv
  targetNamespace: openshift-mtv
  vms: 6
    - id: <source_vm> 7
      - name: <source_vm>
        hooks: 8
          - hook:
              namespace: openshift-mtv
              name: <hook> 9
            step: <step> 10
EOF
```

- 1 **Plan** CR の名前を指定します。
- 2 VMware のみ: 移行がウォームまたはコールドであるかどうかを指定します。**Migration** CR マニフェストで **cutover** パラメーターの値を指定せずにウォーム移行を指定する場合は、プレコピー段階のみが実行されます。ウォーム移行は RHV ではサポートされていません。

- 3 複数のネットワークマッピングを追加することができます。
- 4 **NetworkMap** CR の名前を指定します。
- 5 **StorageMap** CR の名前を指定します。
- 6 **id** パラメーター または **name** パラメーターのいずれかを使用して、ソース仮想マシンを指定することができます。
- 7 VMware 仮想マシン MOR または RHV 仮想マシンの UUID を指定します。
- 8 オプション: 仮想マシンのフックを最大 2 つ指定できます。各フックは個別の移行ステップで実行する必要があります。
- 9 **Hook** CR の名前を指定します。
- 10 移行完了後の移行プランの開始前または **PostHook** の前の許容値は、**PreHook** です。

#### 9. **Plan** CR を実行するための **Migration** マニフェストを作成します。

```
$ cat << EOF | oc apply -f -
apiVersion: forklift.konveyor.io/v1beta1
kind: Migration
metadata:
  name: <migration> 1
  namespace: openshift-mtv
spec:
  plan:
    name: <plan> 2
    namespace: openshift-mtv
  cutover: <cutover_time> 3
EOF
```

- 1 **Migration** CR の名前を指定します。
- 2 実行している **Plan** CR の名前を指定します。 **Migration** CR は、移行される VM ごとに **VirtualMachine** CR を作成します。
- 3 オプション: **2021-04-04T01:23:45.678+09:00** などの UTC 時間オフセットで、ISO 8601 形式に応じたカットオーバー時間を指定します。

複数の **Migration** CR を単一の **Plan** CR に関連付けることができます。移行が完了しない場合は、**Plan** CR を変更せずに新規 **Migration** CR を作成して残りの仮想マシンを移行できます。

#### 10. 移行の進捗をモニタリングするための **Migration** CR を取得します。

```
$ oc get migration/<migration> -n openshift-mtv -o yaml
```

## 5.2. 移行のキャンセル

コマンドラインインターフェース (CLI) から、移行の進行中に、移行全体または個々の仮想マシン (VM) をキャンセルできます。

## 移行全体のキャンセル

- **Migration CR** を削除します。

```
$ oc delete migration <migration> -n openshift-mtv 1
```

- 1 **Migration CR** の名前を指定します。

## 個別の仮想マシンの移行の取り消し

1. **Migration** マニフェストの **spec.cancel** ブロックに個別の VM を追加します。

```
$ cat << EOF | oc apply -f -
apiVersion: forklift.konveyor.io/v1beta1
kind: Migration
metadata:
  name: <migration>
  namespace: openshift-mtv
...
spec:
  cancel:
    - id: vm-102 1
    - id: vm-203
    - name: rhel8-vm
EOF
```

- 1 **id** キーまたは **name** キーを使用して VM を指定できます。

**id** キーの値は、VMware VM の場合は **管理対象オブジェクト参照**、RHV VM の場合は **VM UUID** です。

2. 残りの VM の進捗をモニタリングするための **Migration CR** を取得します。

```
$ oc get migration/<migration> -n openshift-mtv -o yaml
```

## 第6章 高度な移行オプション

### 6.1. ウォーム移行のプレコピー間隔の変更

**ForkliftController** カスタムリソース (CR) にパッチを適用して、スナップショットの間隔を変更できます。

#### 手順

- **ForkliftController** CR にパッチを適用します。

```
$ oc patch forkliftcontroller/<forklift-controller> -n openshift-mtv -p '{"spec":
{"controller_precopy_interval": <60>}}' --type=merge 1
```

- 1 プレコピーの間隔を分単位で指定します。デフォルト値は **60** です。

**forklift-controller** Pod を再起動する必要はありません。

### 6.2. VALIDATION サービスのカスタムルールの作成

**Validation** サービスは Open Policy Agent (OPA) ポリシールールを使用して、移行に対する各仮想マシン (VM) の適合性を確認します。**Validation** サービスは、各 VM の **concerns** 一覧を生成します。これは、**Provider Inventory** サービスに VM 属性として保存されます。Web コンソールには、プロバイダーインベントリー内の各 VM の **concerns** が表示されます。

カスタムルールを作成して、**Validation** サービスのデフォルトルールセットを拡張することができます。たとえば、VM に複数のディスクがあるかどうかを確認するルールを作成できます。

#### 6.2.1. Rego ファイルについて

検証ルールは、Open Policy Agent (OPA) のネイティブクエリ言語である **Rego** で記述されます。ルールは、**Validation** Pod の `/usr/share/opa/policies/io/konveyor/forklift/<provider>` ディレクトリーに **.rego** ファイルとして保存されます。

各検証ルールは、個別の **.rego** ファイルに定義され、特定の条件をテストします。条件が **true** と評価された場合、ルールは `{"category", "label", "assessment"}` ハッシュを **concerns** に追加します。**concerns** のコンテンツは、VM のインベントリーレコードの **concerns** キーに追加されます。Web コンソールには、プロバイダーインベントリー内の各 VM の **concerns** キーのコンテンツが表示されます。

次の **.rego** ファイルの例では、VMware VM のクラスターで有効になっている分散リソーススケジューリングを確認します。

#### drs\_enabled.rego の例

```
package io.konveyor.forklift.vmware 1

has_drs_enabled {
  input.host.cluster.drsEnabled 2
}

concerns[flag] {
```

```

has_drs_enabled
flag := {
  "category": "Information",
  "label": "VM running in a DRS-enabled cluster",
  "assessment": "Distributed resource scheduling is not currently supported by OpenShift
Virtualization. The VM can be migrated but it will not have this feature in the target environment."
}
}

```

- 1 各検証ルールはパッケージ内で定義されます。パッケージの名前空間は、VMware の場合は **io.konveyor.forklift.vmware**、Red Hat Virtualization の場合は **io.konveyor.forklift.ovirt** です。
- 2 クエリーパラメーターは、**Validation** サービス JSON の **input** キーに基づいています。

### 6.2.2. デフォルトの検証ルールの確認

カスタムルールを作成する前に、**Validation** サービスのデフォルトルールを確認して、既存のデフォルト値を再定義するルールを作成しないようにする必要があります。

例: デフォルトのルールに **default valid\_input = false** の行が含まれていて、**default valid\_input = true** の行が含まれるカスタムルールを作成した場合、**Validation** サービスは起動しません。

#### 手順

1. **Validation** Pod のターミナルに接続します。

```
$ oc rsh <validation_pod>
```

2. プロバイダーの OPA ポリシーディレクトリーに移動します。

```
$ cd /usr/share/opa/policies/io/konveyor/forklift/<provider> 1
```

- 1 **vmware** または **ovirt** を指定します。

3. デフォルトポリシーを検索します。

```
$ grep -R "default" *
```

### 6.2.3. Inventory サービス JSON の取得

**Inventory** サービスクエリーを仮想マシン (VM) に送信して **Inventory** サービス JSON を取得します。出力には **"input"** キーが含まれます。このキーには、**Validation** サービスルールによってクエリーされるインベントリー属性が含まれます。

検証ルールは、**"input"** キーの任意の属性に基づいて作成できます (例: **input.snapshot.kind**)。

#### 手順

1. **Inventory** サービスルートの取得

```
$ oc get route <inventory_service> -n openshift-mtv
```

2. プロバイダーの **UUID** を取得します。

```
$ GET https://<inventory_service_route>/providers/<provider> ❶
```

- ❶ プロバイダーで使用できる値は **vsphere** および **ovirt** です。

3. プロバイダーの VM を取得します。

```
$ GET https://<inventory_service_route>/providers/<provider>/<UUID>/vms
```

4. VM の詳細を取得します。

```
$ GET https://<inventory_service_route>/providers/<provider>/<UUID>/workloads/<vm>
```

## 出力例

```
{
  "input": {
    "selfLink": "providers/vsphere/c872d364-d62b-46f0-bd42-16799f40324e/workloads/vm-431",
    "id": "vm-431",
    "parent": {
      "kind": "Folder",
      "id": "group-v22"
    },
    "revision": 1,
    "name": "iscsi-target",
    "revisionValidated": 1,
    "isTemplate": false,
    "networks": [
      {
        "kind": "Network",
        "id": "network-31"
      },
      {
        "kind": "Network",
        "id": "network-33"
      }
    ],
    "disks": [
      {
        "key": 2000,
        "file": "[iSCSI_Datastore] iscsi-target/iscsi-target-000001.vmdk",
        "datastore": {
          "kind": "Datastore",
          "id": "datastore-63"
        },
        "capacity": 17179869184,
        "shared": false,
        "rdm": false
      },
      {
        "key": 2001,
        "file": "[iSCSI_Datastore] iscsi-target/iscsi-target_1-000001.vmdk",
```

```

    "datastore": {
      "kind": "Datastore",
      "id": "datastore-63"
    },
    "capacity": 10737418240,
    "shared": false,
    "rdm": false
  }
],
"concerns": [],
"policyVersion": 5,
"uuid": "42256329-8c3a-2a82-54fd-01d845a8bf49",
"firmware": "bios",
"powerState": "poweredOn",
"connectionState": "connected",
"snapshot": {
  "kind": "VirtualMachineSnapshot",
  "id": "snapshot-3034"
},
"changeTrackingEnabled": false,
"cpuAffinity": [
  0,
  2
],
"cpuHotAddEnabled": true,
"cpuHotRemoveEnabled": false,
"memoryHotAddEnabled": false,
"faultToleranceEnabled": false,
"cpuCount": 2,
"coresPerSocket": 1,
"memoryMB": 2048,
"guestName": "Red Hat Enterprise Linux 7 (64-bit)",
"balloonedMemory": 0,
"ipAddress": "10.19.2.96",
"storageUsed": 30436770129,
"numaNodeAffinity": [
  "0",
  "1"
],
"devices": [
  {
    "kind": "RealUSBController"
  }
],
"host": {
  "id": "host-29",
  "parent": {
    "kind": "Cluster",
    "id": "domain-c26"
  },
  "revision": 1,
  "name": "IP address or host name of the vCenter host or RHV Engine host",
  "selfLink": "providers/vsphere/c872d364-d62b-46f0-bd42-16799f40324e/hosts/host-29",
  "status": "green",
  "inMaintenance": false,

```



```
"managementServerIp": "10.19.2.96",
"thumbprint": <thumbprint>,
"timezone": "UTC",
"cpuSockets": 2,
"cpuCores": 16,
"productName": "VMware ESXi",
"productVersion": "6.5.0",
"networking": {
  "pNICs": [
    {
      "key": "key-vim.host.PhysicalNic-vmnic0",
      "linkSpeed": 10000
    },
    {
      "key": "key-vim.host.PhysicalNic-vmnic1",
      "linkSpeed": 10000
    },
    {
      "key": "key-vim.host.PhysicalNic-vmnic2",
      "linkSpeed": 10000
    },
    {
      "key": "key-vim.host.PhysicalNic-vmnic3",
      "linkSpeed": 10000
    }
  ],
  "vNICs": [
    {
      "key": "key-vim.host.VirtualNic-vmk2",
      "portGroup": "VM_Migration",
      "dPortGroup": "",
      "ipAddress": "192.168.79.13",
      "subnetMask": "255.255.255.0",
      "mtu": 9000
    },
    {
      "key": "key-vim.host.VirtualNic-vmk0",
      "portGroup": "Management Network",
      "dPortGroup": "",
      "ipAddress": "10.19.2.13",
      "subnetMask": "255.255.255.128",
      "mtu": 1500
    },
    {
      "key": "key-vim.host.VirtualNic-vmk1",
      "portGroup": "Storage Network",
      "dPortGroup": "",
      "ipAddress": "172.31.2.13",
      "subnetMask": "255.255.0.0",
      "mtu": 1500
    },
    {
      "key": "key-vim.host.VirtualNic-vmk3",
      "portGroup": "",
      "dPortGroup": "dvportgroup-48",
      "ipAddress": "192.168.61.13",
```

```

        "subnetMask": "255.255.255.0",
        "mtu": 1500
    },
    {
        "key": "key-vim.host.VirtualNic-vmk4",
        "portGroup": "VM_DHCP_Network",
        "dPortGroup": "",
        "ipAddress": "10.19.2.231",
        "subnetMask": "255.255.255.128",
        "mtu": 1500
    }
],
"portGroups": [
    {
        "key": "key-vim.host.PortGroup-VM Network",
        "name": "VM Network",
        "vSwitch": "key-vim.host.VirtualSwitch-vSwitch0"
    },
    {
        "key": "key-vim.host.PortGroup-Management Network",
        "name": "Management Network",
        "vSwitch": "key-vim.host.VirtualSwitch-vSwitch0"
    },
    {
        "key": "key-vim.host.PortGroup-VM_10G_Network",
        "name": "VM_10G_Network",
        "vSwitch": "key-vim.host.VirtualSwitch-vSwitch1"
    },
    {
        "key": "key-vim.host.PortGroup-VM_Storage",
        "name": "VM_Storage",
        "vSwitch": "key-vim.host.VirtualSwitch-vSwitch1"
    },
    {
        "key": "key-vim.host.PortGroup-VM_DHCP_Network",
        "name": "VM_DHCP_Network",
        "vSwitch": "key-vim.host.VirtualSwitch-vSwitch1"
    },
    {
        "key": "key-vim.host.PortGroup-Storage Network",
        "name": "Storage Network",
        "vSwitch": "key-vim.host.VirtualSwitch-vSwitch1"
    },
    {
        "key": "key-vim.host.PortGroup-VM_Isolated_67",
        "name": "VM_Isolated_67",
        "vSwitch": "key-vim.host.VirtualSwitch-vSwitch2"
    },
    {
        "key": "key-vim.host.PortGroup-VM_Migration",
        "name": "VM_Migration",
        "vSwitch": "key-vim.host.VirtualSwitch-vSwitch2"
    }
],
"switches": [
    {

```

```

    "key": "key-vim.host.VirtualSwitch-vSwitch0",
    "name": "vSwitch0",
    "portGroups": [
      "key-vim.host.PortGroup-VM Network",
      "key-vim.host.PortGroup-Management Network"
    ],
    "pNICs": [
      "key-vim.host.PhysicalNic-vmnic4"
    ]
  },
  {
    "key": "key-vim.host.VirtualSwitch-vSwitch1",
    "name": "vSwitch1",
    "portGroups": [
      "key-vim.host.PortGroup-VM_10G_Network",
      "key-vim.host.PortGroup-VM_Storage",
      "key-vim.host.PortGroup-VM_DHCP_Network",
      "key-vim.host.PortGroup-Storage Network"
    ],
    "pNICs": [
      "key-vim.host.PhysicalNic-vmnic2",
      "key-vim.host.PhysicalNic-vmnic0"
    ]
  },
  {
    "key": "key-vim.host.VirtualSwitch-vSwitch2",
    "name": "vSwitch2",
    "portGroups": [
      "key-vim.host.PortGroup-VM_Isolated_67",
      "key-vim.host.PortGroup-VM_Migration"
    ],
    "pNICs": [
      "key-vim.host.PhysicalNic-vmnic3",
      "key-vim.host.PhysicalNic-vmnic1"
    ]
  }
]
},
"networks": [
  {
    "kind": "Network",
    "id": "network-31"
  },
  {
    "kind": "Network",
    "id": "network-34"
  },
  {
    "kind": "Network",
    "id": "network-57"
  },
  {
    "kind": "Network",
    "id": "network-33"
  }
]

```

```
        "kind": "Network",
        "id": "dvportgroup-47"
    }
],
"datastores": [
    {
        "kind": "Datastore",
        "id": "datastore-35"
    },
    {
        "kind": "Datastore",
        "id": "datastore-63"
    }
],
"vms": null,
"networkAdapters": [],
"cluster": {
    "id": "domain-c26",
    "parent": {
        "kind": "Folder",
        "id": "group-h23"
    },
    "revision": 1,
    "name": "mycluster",
    "selfLink": "providers/vsphere/c872d364-d62b-46f0-bd42-16799f40324e/clusters/domain-c26",
    "folder": "group-h23",
    "networks": [
        {
            "kind": "Network",
            "id": "network-31"
        },
        {
            "kind": "Network",
            "id": "network-34"
        },
        {
            "kind": "Network",
            "id": "network-57"
        },
        {
            "kind": "Network",
            "id": "network-33"
        },
        {
            "kind": "Network",
            "id": "dvportgroup-47"
        }
    ]
},
"datastores": [
    {
        "kind": "Datastore",
        "id": "datastore-35"
    },
    {
        "kind": "Datastore",
```

```

        "id": "datastore-63"
      }
    ],
    "hosts": [
      {
        "kind": "Host",
        "id": "host-44"
      },
      {
        "kind": "Host",
        "id": "host-29"
      }
    ],
    "dasEnabled": false,
    "dasVms": [],
    "drsEnabled": true,
    "drsBehavior": "fullyAutomated",
    "drsVms": [],
    "datacenter": null
  }
}
}
}

```

#### 6.2.4. 検証規則の作成

ルールを含む設定マップカスタムリソース (CR) を **Validation** サービスに適用して、検証ルールを作成します。



##### 重要

- 既存のルールと同じ名前で作成すると、**Validation** サービスはルールとの **OR** 操作を実行します。
- デフォルトのルールと矛盾するルールを作成すると、**Validation** サービスは開始されません。

#### 検証規則の例

検証ルールは、**Provider Inventory** サービスが収集する仮想マシン (VM) 属性に基づいています。

たとえば、VMware API はこのパスを使用して、VMware VM に NUMA ノードアフィニティが設定されているかどうかを確認します: **MOR:Virtual Machine.config.extra Config["numa.node Affinity"]**。

**Provider Inventory** サービスは、この設定を簡素化し、リスト値を持つテスト可能な属性を返します。

```

"numaNodeAffinity": [
  "0",
  "1"
],

```

この属性に基づいて **Rego** クエリーを作成し、それを **forklift-validation-config** 設定マップに追加します。

```

`count(input.numaNodeAffinity) != 0`

```

## 手順

1. 以下の例に従って設定マップ CR を作成します。

```
$ cat << EOF | oc apply -f -
apiVersion: v1
kind: ConfigMap
metadata:
  name: <forklift-validation-config>
  namespace: openshift-mtv
data:
  vmware_multiple_disks.rego: |-
    package <provider_package> 1

    has_multiple_disks { 2
      count(input.disks) > 1
    }

    concerns[flag] {
      has_multiple_disks 3
      flag := {
        "category": "<Information>", 4
        "label": "Multiple disks detected",
        "assessment": "Multiple disks detected on this VM."
      }
    }
  }
EOF
```

- 1** プロバイダーパッケージ名を指定します。使用できる値は、VMware の場合は **io.konveyor.forklift.vmware**、Red Hat Virtualization の場合は **io.konveyor.forklift.ovirt** です。
- 2** **concerns** の名前と Rego クエリーを指定します。
- 3** **concerns** の名前と **flag** パラメーターの値を指定します。
- 4** 使用できる値は **Critical**、**Warning**、および **Information** です。

2. **forklift-controller** デプロイメントを **0** にスケーリングして、**Validation** Pod を停止します。

```
$ oc scale -n openshift-mtv --replicas=0 deployment/forklift-controller
```

3. **forklift-controller** デプロイメントを **1** にスケーリングして、**Validation** Pod を起動します。

```
$ oc scale -n openshift-mtv --replicas=1 deployment/forklift-controller
```

4. **Validation** Pod ログをチェックして、Pod が起動したことを確認します。

```
$ oc logs -f <validation_pod>
```

カスタムルールがデフォルトのルールと競合する場合、**Validation** Pod は起動しません。

5. ソースプロバイダーを削除します。

```
$ oc delete provider <provider> -n openshift-mtv
```

6. ソースプロバイダーを追加して、新規ルールを適用します。

```
$ cat << EOF | oc apply -f -
apiVersion: forklift.konveyor.io/v1beta1
kind: Provider
metadata:
  name: <provider>
  namespace: openshift-mtv
spec:
  type: <provider_type> ①
  url: <api_end_point> ②
  secret:
    name: <secret> ③
    namespace: openshift-mtv
EOF
```

- ① 使用できる値は **ovirt** および **vsphere** です。
- ② API エンドポイントの URL を指定します (vSphere の場合は **https://<vCenter\_host>/sdk**、RHV の場合は **https://<engine\_host>/ovirt-engine/api/**)。
- ③ プロバイダーの **Secret** CR の名前を指定します。

カスタムルールを作成した後、ルールバージョンを更新して、**Inventory** サービスが変更を検出し、VM を検証できるようにする必要があります。

### 6.2.5. インベントリールールバージョンの更新

**Provider Inventory** サービスが変更を検出して **Validation** サービスをトリガーするように、ルールを更新するたびにインベントリールールのバージョンを更新する必要があります。

ルールバージョンは、各プロバイダーの **rules\_version.rego** ファイルに記録されます。

#### 手順

1. 現在のルールバージョンを取得します。

```
$ GET https://forklift-validation/v1/data/io/konveyor/forklift/<provider>/rules_version ①
```

#### 出力例

```
{
  "result": {
    "rules_version": 5
  }
}
```

2. **Validation** Pod のターミナルに接続します。

```
$ oc rsh <validation_pod>
```

3. `/usr/share/opa/policies/io/konveyor/forklift/<provider>/rules_version.rego` ファイルでルールバージョンを更新します。
4. **Validation** Pod ターミナルからログアウトします。
5. 更新されたルールバージョンを検証します。

```
$ GET https://forklift-validation/v1/data/io/konveyor/forklift/<provider>/rules_version 1
```

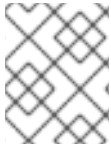
#### 出力例

```
{  
  "result": {  
    "rules_version": 6  
  }  
}
```



## 第7章 MIGRATION TOOLKIT FOR VIRTUALIZATION のアップグレード

OpenShift Container Platform Web コンソールを使用して新しいバージョンをインストールすることで、MTV Operator をアップグレードできます。



### 注記

リリースを省略せずに次のリリースにアップグレードする必要があります (例: 2.0 から 2.1 または 2.1 から 2.2)。

### 手順

1. OCP Web コンソールで、**Operators** → **Installed Operators** → **Migration Toolkit for Virtualization Operator** → **Subscription** をクリックします。
2. 更新チャンネルを **release-v2.2.0** に変更します。  
OpenShift Container Platform ドキュメントの [更新チャンネルの変更](#) を参照してください。
3. **Upgrade status** が **Up to date** から **Upgrade available** に変わります。そうでない場合は、**Catalog Source** Pod を再起動します。

- a. カタログソース (例: **redhat-operators**) に注意してください。
- b. コマンドラインで、カタログソース Pod を取得します。

```
$ oc get pod -n openshift-marketplace | grep <catalog_source> 1
```

**1 1 1** カタログソースを指定します (例: **redhat-operators**)。

- c. Pod を削除します。

```
$ oc delete pod -n openshift-marketplace <catalog_source_pod>
```

**Upgrade status** が **Up to date** から **Upgrade available** に変わります。

**Subscriptions** タブで **Update approval** を **Automatic** に設定すると、アップグレードが自動的に開始されます。

4. **Subscriptions** タブで **Update approval** を **Manual** に設定すると、アップグレードが承認されます。  
OpenShift Container Platform ドキュメントの [保留中のアップグレードの手動での承認](#) を参照してください。
5. Web コンソールにログインする前に、**forklift-ui** Pod が **Ready** 状態になっていることを確認します。

```
$ oc get pods -n openshift-mtv
```

### 出力例

```
NAME                                READY STATUS RESTARTS AGE
forklift-controller-788bdb4c69-mw268 2/2   Running 0     2m
```

forklift-operator-6bf45b8d8-qps9v	1/1	Running	0	5m
forklift-ui-7cdf96d8f6-xnw5n	1/1	Running	0	2m

## 第8章 MIGRATION TOOLKIT FOR VIRTUALIZATION のアンインストール

OpenShift Container Platform Web コンソールまたはコマンドラインインターフェース (CLI) を使用して Migration Toolkit for Virtualization (MTV) をアンインストールできます。

### 8.1. OPENSIFT CONTAINER PLATFORM WEB コンソールを使用した MTV のアンインストール

OpenShift Container Platform Web コンソールを使用して **openshift-mtv** プロジェクトおよびカスタムリソース定義 (CRD) を削除し、Migration Toolkit for Virtualization (MTV) をアンインストールできます。

#### 前提条件

- **cluster-admin** 権限を持つユーザーとしてログインしている。

#### 手順

1. Home → Projects をクリックします。
2. **openshift-mtv** プロジェクトを作成します。
3. プロジェクトの右側で、Options メニュー  から **Delete Project** を選択します。
4. **Delete Project** ペインでプロジェクト名を入力し、**Delete** をクリックします。
5. **Administration** → **CustomResourceDefinitions** の順にクリックします。
6. **検索** フィールドに **forklift** を入力し、**forklift.konveyor.io** グループで CRD を見つけます。
7. 各 CRD の右側で、Options メニュー  から **Delete CustomResourceDefinition** を選択します。

### 8.2. コマンドラインインターフェースからの MTV のアンインストール

**openshift-mtv** プロジェクトおよび **forklift.konveyor.io** カスタムリソース定義 (CRD) を削除して、コマンドラインインターフェース (CLI) から Migration Toolkit for Virtualization (MTV) をアンインストールできます。

#### 前提条件

- **cluster-admin** 権限を持つユーザーとしてログインしている。

#### 手順

1. プロジェクトを削除します。

```
$ oc delete project openshift-mtv
```

2. CRD を削除します。

■

```
┆ $ oc get crd -o name | grep 'forklift' | xargs oc delete
```

3. OAuthClient を削除します。

```
┆ $ oc delete oauthclient/forklift-ui
```

## 第9章 トラブルシューティング

このセクションでは、一般的な移行の問題をトラブルシューティングするための情報を提供します。

### 9.1. アーキテクチャー

このセクションでは、MTV カスタムリソース、サービス、およびワークフローについて説明します。

#### 9.1.1. MTV カスタムリソースおよびサービス

Migration Toolkit for Virtualization (MTV) は OpenShift Container Platform Operator として提供されます。これは、以下のカスタムリソース (CR) およびサービスを作成し、管理します。

##### MTV カスタムリソース

- **Provider** CR は、MTV がソースおよびターゲットプロバイダーに接続し、対話できるようにする属性を保存します。
- **NetworkMapping** CR は、ソースおよびターゲットプロバイダーのネットワークをマッピングします。
- **StorageMapping** CR は、ソースおよびターゲットプロバイダーのストレージをマッピングします。
- **Provisioner** CR は、サポートされているボリューム、アクセスモードなどのストレージプロビジョナーの設定を保存します。
- **Plan** CR には、同じ移行パラメーターと関連するネットワークおよびストレージマッピングを持つ仮想マシンの一覧が含まれます。
- **Migration** CR は移行計画を実行します。  
一度に実行できる **Migration** CR は、移行プランごとに1つのみです。単一の **Plan** CR に複数の **Migration** CR を作成できます。

##### MTV サービス

- **Inventory** サービスは以下のアクションを実行します。
  - ソースプロバイダーおよびターゲットプロバイダーに接続します。
  - マッピングおよび計画に関するローカルインベントリを維持します。
  - 仮想マシンの設定を保存します。
  - 仮想マシンの設定の変更が検出されたら、**Validation** サービスを実行します。
- **Validation** サービスは、ルールを適用して移行の適合性を確認します。
- **User Interface** サービスは、以下のアクションを実行します。
  - MTV CR を作成および設定できます。
  - CR のステータスと移行の進捗を表示します。
- **Migration Controller** サービスは移行のオーケストレーションを行います。  
移行計画の作成時に、**Migration Controller** サービスは計画を検証し、ステータスラベルを追

加します。計画の検証に失敗した場合には、計画のステータスは **Not ready** となり、計画を使用して移行を行うことができません。計画が検証をパスすると、計画のステータスは **Ready** となり、移行を実行するために使用することができます。移行に成功すると、**Migration Controller** サービスは計画のステータスを **Completed** に変更します。

- **Kubevirt Controller** および **Containerized Data Import (CDI) Controller** サービスは、ほとんどの技術操作を処理します。

### 9.1.2. 移行ワークフローの概要

高レベルのワークフローは、ユーザーの観点から移行プロセスを示しています。

1. ソースプロバイダー、ターゲットプロバイダー、ネットワークマッピング、およびストレージマッピングを作成します。
2. 以下のリソースを含む **Plan** カスタムリソース (CR) を作成します。
  - ソースプロバイダー
  - ターゲットプロバイダー (MTV がターゲットクラスターにインストールされていない場合)
  - ネットワークマッピング
  - ストレージマッピング
  - 1つ以上の仮想マシン (VM)
3. **Plan** CR を参照する **Migration** CR を作成して移行計画を実行します。  
何らかの理由ですべての VM 移行できない場合は、すべての VM が移行されるまで、同じ **Plan** CR に対して複数の **Migration** CR を作成できます。
4. **Plan** CR の各 VM について、**Migration Controller** サービスは **VirtualMachine** CR を作成し、**Migration** CR に VR の移行の進捗を記録します。  
すべての VM が移行されると、**Migration Controller** サービスは **Plan** CR のステータスを **Completed** に更新します。各ソース VM の電源状態は、移行後も維持されます。

### 9.1.3. 詳細な移行ワークフロー

詳細な移行ワークフローを使用して、失敗した移行のトラブルシューティングを行うことができます。

ワークフローでは、以下の手順について説明します。

1. **Migration** カスタムリソース (CR) を作成して移行計画を実行すると、**Migration Controller** サービスは各ソース仮想マシン (VM) ごとに **VirtualMachine** CR を作成し、各ソース VM ディスクごとに **DataVolume** CR を作成します。  
各仮想マシンディスクの場合:
2. **Containerized Data Importer (CDI) Controller** サービスは、**DataVolume** CR で指定されるパラメーターに基づいて永続ボリューム要求 (PVC) を作成します。■■
3. 永続ボリューム (PV) は、**StorageClass** プロビジョナーによって動的にプロビジョニングされます。■■
4. **CDI Controller** サービスは **Importer** Pod を作成します。
5. **Importer** Pod は VM ディスクを PV にストリーミングします。

仮想マシンディスクが転送された後:

6. **Migration Controller** サービスは、PVC が割り当てられた **conversion** Pod を作成します。**Conversion** Pod は **virt-v2v** を実行します。これは、ターゲット VM の PVC にデバイスドライバーをインストールし、設定します。
7. ターゲット VM の電源がオンになると、**KubeVirt Controller** サービスは **virt-launcher** Pod および **VirtualMachineInstance** CR を作成します。**virt-launcher** Pod は、VM ディスクとして割り当てられた PVC で **QEMU-KVM** を実行します。

## 9.2. ログとカスタムリソース

トラブルシューティングのためにログおよびカスタムリソース (CR) の情報をダウンロードできます。詳細は、[詳細な移行ワークフロー](#) を参照してください。

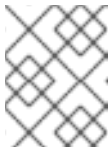
### 9.2.1. 収集されるログおよびカスタムリソース情報

MTV Web コンソールまたはコマンドラインインターフェース (CLI) を使用して、以下のターゲットのログおよびカスタムリソース (CR) **yaml** ファイルをダウンロードできます。

- 移行計画: Web コンソールまたは CLI。
- 仮想マシン: Web コンソールまたは CLI。
- 名前空間: CLI のみ。

**must-gather** ツールは、以下のログおよび CR ファイルをアーカイブファイルで収集します。

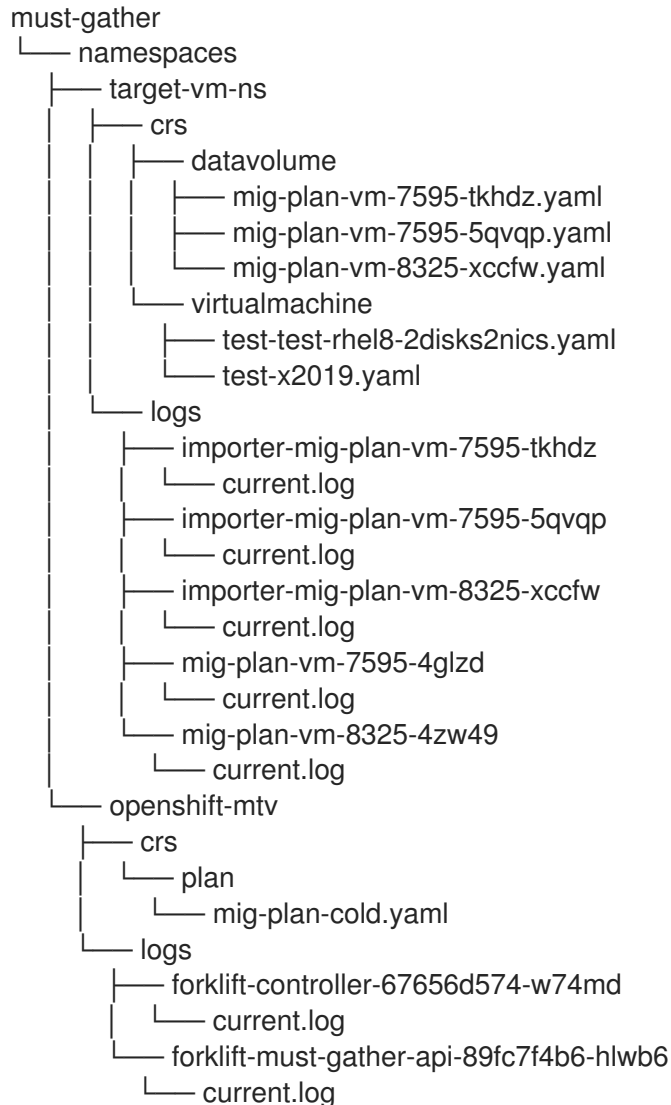
- CR:
  - **DataVolume** CR: 移行された VM にマウントされているディスクを表します。
  - **VirtualMachine** CR: 移行された VM を表します。
  - **Plan** CR: VM およびストレージおよびネットワークマッピングを定義します。
- ログ:
  - **Importer** Pod: ディスクからデータへのボリューム変換ログ。**Importer** Pod の命名規則は **importer-`<migration_plan>-<vm_id><5_char_id>`** です。たとえば、**importer-mig-plan-ed90dfc6-9a17-4a8btnfh** の場合、**ed90dfc6-9a17-4a8** は切り捨てられた RHV VM ID であり、**btnfh** は生成された 5 文字の ID です。
  - **conversion** Pod: VM の変換ログ。**conversion** Pod は **virt-v2v** を実行します。これは、VM の PVC にデバイスドライバーをインストールし、設定します。**conversion** Pod の命名規則は **<migration\_plan>-<vm\_id><5\_char\_id>** です。
  - **virt-launcher** Pod: VM ランチャーログ。移行した VM の電源がオンになると、**virt-launcher** Pod は VM ディスクとして割り当てられた PVC で **QEMU-KVM** を実行します。
  - **forklift-controller** Pod: ログは **must-gather** コマンドで指定される移行計画、仮想マシン、または名前空間に対してフィルター処理されます。
  - **forklift-must-gather-api** Pod: ログは **must-gather** コマンドで指定される移行計画、仮想マシン、または名前空間に対してフィルター処理されます。



## 注記

空または除外されたログファイルは、**must-gather** アーカイブファイルには含まれません。

### VMware 移行計画の must-gather アーカイブ構造の例



### 9.2.2. Web コンソールからのログおよびカスタムリソース情報のダウンロード

MTV Web コンソールを使用して、完了、失敗、またはキャンセルされた移行計画、または移行された仮想マシン (VM) のログとカスタムリソース (CR) に関する情報をダウンロードできます。

#### 手順

1. Web コンソールで、**Migration plans** をクリックします。
2. 移行計画名の横にある **Get logs** をクリックします。
3. **Get logs** ウィンドウで **Get logs** をクリックします。  
ログが収集されます。**Log collection complete** メッセージが表示されます。
4. **Download logs** をクリックしてアーカイブファイルをダウンロードします。



5. 移行された VM のログをダウンロードするには、移行計画名をクリックして、VM の横にある **Get logs** をクリックします。

### 9.2.3. コマンドラインインターフェースからのログおよびカスタムリソース情報へのアクセス

**must-gather** ツールを使用して、コマンドラインインターフェースからカスタムリソース (CR) のログおよび情報にアクセスできます。**must-gather** データファイルをすべてのカスタマーケースに割り当てる必要があります。

フィルターオプションを使用して、特定の名前空間、完了、失敗、またはキャンセルされた移行計画、移行された仮想マシン (VM) のデータを収集できます。



#### 注記

フィルターされた **must-gather** コマンドで存在しないリソースを指定する場合、アーカイブファイルは作成されません。

#### 前提条件

- **cluster-admin** ロールを持つユーザーとして OpenShift Virtualization クラスタにログインしている必要があります。
- [OpenShift Container Platform CLI \(oc\)](#) がインストールされていること。

#### 手順

1. **must-gather** データを保存するディレクトリーに移動します。
2. **oc adm must-gather** コマンドを実行します。

```
$ oc adm must-gather --image=registry.redhat.io/migration-toolkit-virtualization/mtv-must-gather-rhel8:2.2.0
```

データは **/must-gather/must-gather.tar.gz** として保存されます。このファイルを [Red Hat カスタマーポータル](#) で作成したサポートケースにアップロードすることができます。

3. オプション: **oc adm must-gather** コマンドに以下のオプションを指定して実行し、フィルターされたデータを収集します。

- Namespace:

```
$ oc adm must-gather --image=registry.redhat.io/migration-toolkit-virtualization/mtv-must-gather-rhel8:2.2.0 \
  -- NS=<namespace> /usr/bin/targeted
```

- 移行計画:

```
$ oc adm must-gather --image=registry.redhat.io/migration-toolkit-virtualization/mtv-must-gather-rhel8:2.2.0 \
  -- PLAN=<migration_plan> /usr/bin/targeted
```

- 仮想マシン:

```
$ oc adm must-gather --image=registry.redhat.io/migration-toolkit-virtualization/mtv-must-gather-rhel8:2.2.0 \  
-- VM=<vm_name> NS=<namespace> /usr/bin/targeted ❶
```

- ❶ VM ID ではなく、**Plan** CR に表示される VM の **名前**を指定する必要があります。