



Red Hat OpenShift Data Foundation 4.11

デプロイメントのプランニング

Red Hat OpenShift Data Foundation 4.11 をデプロイする際の重要な考慮事項

Red Hat OpenShift Data Foundation 4.11 デプロイメントのプランニング

Red Hat OpenShift Data Foundation 4.11 をデプロイする際の重要な考慮事項

法律上の通知

Copyright © 2023 Red Hat, Inc.

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[®] is the registered trademark of Linus Torvalds in the United States and other countries.

Java[®] is a registered trademark of Oracle and/or its affiliates.

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

MySQL[®] is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js[®] 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[®] 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

本書は、Red Hat OpenShift Data Foundation のデプロイメントを計画する際の重要な考慮事項について説明します。

目次

多様性を受け入れるオープンソースの強化	3
RED HAT ドキュメントへのフィードバック (英語のみ)	4
第1章 OPENSIFT DATA FOUNDATION の紹介	5
第2章 OPENSIFT DATA FOUNDATION のアーキテクチャー	6
2.1. OPERATOR	7
2.2. ストレージクラスターのデプロイメントアプローチ	7
2.3. ノードのタイプ	9
第3章 内部ストレージサービス	11
第4章 外部ストレージサービス	12
第5章 セキュリティーに関する考慮事項	13
5.1. FIPS-140-2	13
5.2. プロキシ環境	13
5.3. データ暗号化オプション	13
第6章 サブスクリプション	16
6.1. サブスクリプションのオフアリング	16
6.2. 障害復旧サブスクリプションの要件	16
6.3. コア対 VCPU およびハイパースレッディング	16
6.4. コアの分割	17
6.5. サブスクリプションの要件	17
第7章 インフラストラクチャーの要件	19
7.1. プラットフォーム要件	19
7.2. 外部モード要件	21
7.3. リソース要件	21
7.4. POD の配置ルール	25
7.5. ストレージデバイスの要件	26
7.6. マルチネットワークプラグイン (MULTUS) のサポート [テクノロジープレビュー]	27
第8章 障害復旧	31
8.1. REGIONAL-DR [テクノロジープレビュー]	31
8.2. METRO-DR [テクノロジープレビュー]	32
8.3. STRETCHED CLUSTER - ARBITER [テクノロジープレビュー]	32
第9章 非接続環境	34
第10章 IBM POWER および IBM Z インフラストラクチャーでサポートされている機能およびサポートされていない機能	35
第11章 次のステップ	37

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、[Red Hat CTO である Chris Wright のメッセージ](#) をご覧ください。

RED HAT ドキュメントへのフィードバック (英語のみ)

Red Hat ドキュメントに対するご意見をお聞かせください。ドキュメントの改善点があれば、ぜひお知らせください。フィードバックをお寄せいただくには、以下をご確認ください。

- 特定の部分についての簡単なコメントをお寄せいただく場合は、以下をご確認ください。
 1. ドキュメントの表示が **Multi-page HTML** 形式になっていることを確認してください。ドキュメントの右上隅に **Feedback** ボタンがあることを確認してください。
 2. マウスカーソルを使用して、コメントを追加するテキストの部分を強調表示します。
 3. 強調表示されたテキストの下に表示される **Add Feedback** ポップアップをクリックします。
 4. 表示される指示に従ってください。
- より詳細なフィードバックをお寄せいただく場合は、Bugzilla のチケットを作成してください。
 1. [Bugzilla](#) の Web サイトに移動します。
 2. **Component** セクションで、**documentation** を選択します。
 3. **Description** フィールドに、ドキュメントの改善に向けたご提案を記入してください。ドキュメントの該当部分へのリンクも追加してください。
 4. **Submit Bug** をクリックします。

第1章 OPENSIFT DATA FOUNDATION の紹介

Red Hat OpenShift Data Foundation は、Red Hat OpenShift Container Platform のクラウドストレージおよびデータサービスの集合です。これは、単純なデプロイメントや管理を容易に実行できるように Operator として同梱されており、Red Hat OpenShift Container Platform サービスカタログの一部として提供されます。

Red Hat OpenShift Data Foundation サービスは、主に以下のコンポーネントを表すストレージクラスを使用してアプリケーションで使用できます。

- ブロックストレージデバイス。主にデータベースのワークロードに対応します。主な例には、Red Hat OpenShift Container Platform のロギングおよびモニタリング、および PostgreSQL などがあります。
- 共有および分散ファイルシステム。主にソフトウェア開発、メッセージング、およびデータ集約のワークロードに対応します。これらの例には、Jenkins ビルドソースおよびアーティファクト、Wordpress のアップロードコンテンツ、Red Hat OpenShift Container Platform レジストリー、および JBoss AMQ を使用したメッセージングが含まれます。
- Multicloud オブジェクトストレージ。複数のクラウドオブジェクトストアからのデータの保存および取得を抽象化できる軽量 S3 API エンドポイントを特長としています。
- オンプレミスオブジェクトストレージ。主にデータ集約型アプリケーションをターゲットとする数十ペタバイトおよび数十億のオブジェクトにスケーリングする堅牢な S3 API エンドポイントを特長としています。これらの例には、Spark、Presto、Red Hat AMQ Streams (Kafka) などのアプリケーションや、TensorFlow や Pytorch などのマシンラーニングフレームワークを使用した行、列、および半構造化データの保存およびアクセスが含まれます。



注記

CephFS 永続ボリューム上での PostgreSQL ワークロードの実行はサポートされていないため、RADOS Block Device (RBD) ボリュームを使用することを推奨します。

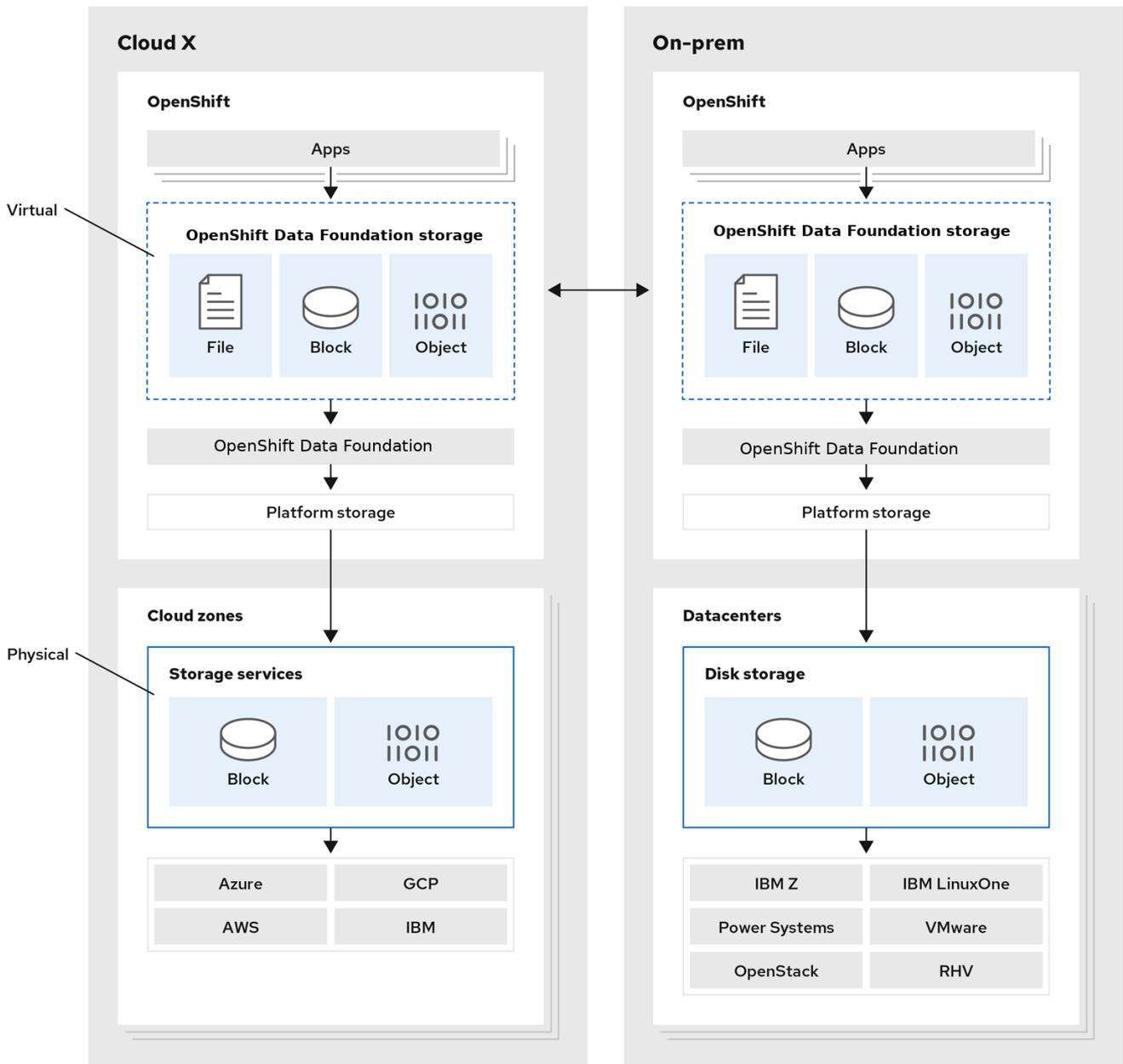
Red Hat OpenShift Data Foundation バージョン 4.x は、以下を含むソフトウェアプロジェクトのコレクションを統合します。

- Ceph。ブロックストレージ、共有および分散ファイルシステム、およびオンプレミスのオブジェクトストレージを提供します。
- Ceph CSI。永続ボリュームおよび要求のプロビジョニングおよびライフサイクルを管理します。
- NooBaa。Multicloud Object Gateway を提供します。
- OpenShift Data Foundation サービスを初期化し、管理する OpenShift Data Foundation、Rook-Ceph、および NooBaa Operator。

第2章 OPENSIFT DATA FOUNDATION のアーキテクチャー

Red Hat OpenShift Data Foundation は、Red Hat OpenShift Container Platform のサービスを提供し、Red Hat OpenShift Container Platform の内部で実行できます。

図2.1 Red Hat OpenShift Data Foundation アーキテクチャー



171_OpenShift_1221

Red Hat OpenShift Data Foundation は、インストーラーでプロビジョニングされるインフラストラクチャー、またはユーザーによってプロビジョニングされるインフラストラクチャーでデプロイされる Red Hat OpenShift Container Platform クラスタへのデプロイメントをサポートします。

これら 2 つの方法については、[OpenShift Container Platform のインストールプロセス](#) を参照してください。

Red Hat OpenShift Data Foundation と Red Hat OpenShift Container Platform のコンポーネントの相互運用性の詳細は、[Red Hat OpenShift Data Foundation のサポート性および相互運用性チェッカー](#) を参照してください。

OpenShift Container Platform のアーキテクチャーおよびライフサイクルについての詳細は、[OpenShift Container Platform architecture](#) を参照してください。

ヒント

IBM Power については、[OpenShift Container Platform-インストールプロセス](#) を参照してください。

2.1. OPERATOR

Red Hat OpenShift Data Foundation は、管理タスクとカスタムリソースを体系化する 3 つの主要な Operator で設定されているため、タスクとリソースの特性を簡単に自動化できます。管理者はクラスターの必要な最終状態を定義し、OpenShift Data Foundation Operator は管理者の介入を最小限に抑えてクラスターをその状態にするか、その状態に近づけるようにします。

OpenShift Data Foundation Operator

サポートされる Red Hat OpenShift Data Foundation のデプロイメントの推奨事項と要件を成文化し、実施するために、特定のテストされた方法で他の Operator を利用するメタ Operator です。rook-ceph および noobaa operator は、これらのリソースをラップするストレージクラスターリソースを提供します。

Rook-ceph operator

この Operator は、永続ストレージおよびファイル、ブロックおよびオブジェクトサービスのパッケージ化、デプロイメント、管理、アップグレード、およびスケーリングを自動化します。これは、すべての環境用にブロックおよびファイルストレージクラスを作成し、オンプレミス環境でオブジェクトストレージクラスおよびサービスオブジェクトバケット要求 (OBC) を作成します。

さらに、内部モードクラスターの場合、以下を表すデプロイメントおよびサービスを管理する ceph クラスターリソースを提供します。

- オブジェクトストレージデーモン (OSD)
- モニター (MON)
- マネージャー (MGR)
- メタデータサーバー (MDS)
- オンプレミスのみの RADOS オブジェクトゲートウェイ (RGW)

Multicloud Object Gateway operator

この Operator は、Multicloud Object Gateway (MCG) オブジェクトサービスのパッケージ化、デプロイメント、管理、アップグレード、およびスケーリングを自動化します。オブジェクトストレージクラスを作成し、それに対して作成された OBC にサービスを提供します。

さらに、これは NooBaa クラスターリソースを提供します。このクラスターリソースは、NooBaa コア、データベースおよびエンドポイントのデプロイメントとサービスを管理します。

2.2. ストレージクラスターのデプロイメントアプローチ

モード数 (operating modalities) が増えていることから、柔軟性が Red Hat OpenShift Data Foundation の主な特徴であることが分かります。本セクションでは、お使いの環境に最も適した方法を選択するのに役立つ情報を提供します。

Red Hat OpenShift Data Foundation は OpenShift Container Platform 内で完全にデプロイ (内部アプローチ) することも、OpenShift Container Platform 外で実行されるクラスターからサービスを利用可能な方法 (外部アプローチ) を実行することもできます。

2.2.1. 内部アプローチ

Red Hat OpenShift Data Foundation を Red Hat OpenShift Container Platform 内にすべてデプロイすると、Operator ベースのデプロイメントおよび管理からのすべての利点が得られます。グラフィカルユーザーインターフェイス (GUI) で、内部でアタッチされているデバイスのアプローチを使用して、ローカルストレージ Operator とローカルストレージデバイスで、Red Hat OpenShift Data Foundation を内部モードでデプロイできます。

デプロイメントおよび管理の容易性は、OpenShift Data Foundation サービスを OpenShift Container Platform の内部で実行することについての主な特長となっています。Red Hat OpenShift Data Foundation が完全に Red Hat OpenShift Container Platform 内で実行されている場合、以下の 2 つのデプロイメントモードを使用できます。

- Simple (単純)
- Optimized (最適化)

Simple (単純) デプロイメント

Red Hat OpenShift Data Foundation サービスは、アプリケーションと共存する形で実行されます。Red Hat OpenShift Container Platform の operator がこのようなアプリケーションを管理します。

Simple (単純) デプロイメントは、以下のような場合に最も適しています。

- ストレージ要件が明確ではない。
- Red Hat OpenShift Data Foundation サービスは、アプリケーションと共存して実行されている。
- 特定のサイズのノードインスタンスを作成することが困難である (例: ベアメタル)。

Red Hat OpenShift Data Foundation をアプリケーションと共存させるには、アプリケーションにローカルストレージデバイス、または EC2 の EBS ボリューム、VMware の vSphere 仮想ボリューム、SAN ボリュームなどのポータブルストレージデバイスを動的に接続する必要があります。



注記

PowerVC は SAN ボリュームを動的にプロビジョニングします。

Optimized (最適化) デプロイメント

Red Hat OpenShift Data Foundation サービスは、専用のインフラストラクチャーノードで実行します。Red Hat OpenShift Container Platform がこのようなインフラストラクチャーノードを管理しません。

最適化アプローチは、以下の場合に最も適しています。

- ストレージ要件が明確である。
- Red Hat OpenShift Data Foundation サービスが、専用のインフラストラクチャーノードで実行されている。
- 特定サイズのノードインスタンスの作成が容易である (例: クラウド、仮想化環境など)。

2.2.2. 外部アプローチ

Red Hat OpenShift Data Foundation は、OpenShift Container Platform クラスター外で実行されている Red Hat Ceph Storage サービスをストレージクラスとして公開します。

以下の場合に外部アプローチが最も適しています。

- ストレージ要件の規模が大きい (600 以上のストレージデバイス)。
- 複数の OpenShift Container Platform クラスターが共通の外部クラスターからストレージサービスを使用する必要がある。
- 別のチームであるサイトリライアビリティエンジニアリング (SRE)、ストレージなどは、ストレージサービスを提供する外部クラスターを管理する必要がある。(すでに存在している場合があります)

2.3. ノードのタイプ

ノードはコンテナランタイムとサービスを実行し、コンテナが実行中の状態にし、Pod 間のネットワーク通信および分離を保ちます。OpenShift Data Foundation には、3 種類のノードがあります。

表2.1 ノードの種類

ノードタイプ	説明
マスター	これらのノードは、Kubernetes API を公開し、新たに作成された Pod を監視およびスケジュールし、ノードの正常性および数量を維持し、基礎となるクラウドプロバイダーとの対話を制御するプロセスを実行します。
インフラストラクチャー (インフラ)	<p>インフラストラクチャーノードは、ロギング、メトリクス、レジストリー、およびルーティングなどのクラスターレベルのインフラストラクチャーサービスを実行します。これらは OpenShift Container Platform クラスターではオプションです。OpenShift Data Foundation レイヤーワークロードをアプリケーションから分離するには、仮想化環境およびクラウド環境で OpenShift Data Foundation に infra ノードを使用することを確認します。</p> <p>infra というラベルが付けられた新規ノードをプロビジョニングして、インフラストラクチャーノードを作成できます。詳細は、Red Hat OpenShift Data Foundation 専用のワーカーノードを使用する方法 を参照してください。</p>

ノードタイプ	説明
ワーカー	<p>ワーカーノードは、アプリケーションを実行するため、アプリケーションノードとしても知られています。</p> <p>OpenShift Data Foundation が内部モードでデプロイされている場合、3つのワーカーノードの最小クラスターが必要です。可用性を確保するために、ノードが3つの異なるラックまたはアベイラビリティゾーンに分散していることを確認してください。OpenShift Data Foundation をワーカーノードで実行するには、ローカルストレージデバイスまたはポータブルストレージデバイスをワーカーノードに動的に接続する必要があります。</p> <p>OpenShift Data Foundation が外部モードでデプロイされている場合、複数のノードで実行されます。これにより、Kubernetes は、障害が発生した場合に使用可能なノードでスケジュールを変更できます。</p>



注記

OpenShift Data Foundation には、OpenShift Container Platform と同じ数のサブスクリプションが必要です。ただし、OpenShift Data Foundation がインフラノードで実行されている場合、OpenShift はこれらのノードに OpenShift Container Platform サブスクリプションを必要としません。したがって、OpenShift Data Foundation コントロールプレーンには、追加の OpenShift Container Platform および OpenShift Data Foundation サブスクリプションは必要ありません。詳細は、[6章 サブスクリプション](#) を参照してください。

第3章 内部ストレージサービス

Red Hat OpenShift Data Foundation サービスは、以下のインフラストラクチャーで実行されている Red Hat OpenShift Container Platform の内部で利用できます。

- Amazon Web Services (AWS)
- ベアメタル
- VMware vSphere
- Microsoft Azure
- Google Cloud [テクノロジープレビュー]
- Red Hat Virtualization 4.4.x 以降 (インストーラーがプロビジョニングしたインフラストラクチャー)
- Red Hat OpenStack 13 以降 (インストーラーがプロビジョニングしたインフラストラクチャー) [テクノロジープレビュー]
- IBM Power
- IBM Z および LinuxONE

内部クラスターリソースを作成すると、OpenShift Data Foundation ベースサービスの内部プロビジョニングが実行され、追加のストレージクラスがアプリケーションで使用可能になります。

第4章 外部ストレージサービス

Red Hat OpenShift Data Foundation は、IBM Flash Systems を使用するか、外部の Red Hat Ceph Storage クラスタからのサービスを以下のプラットフォームで実行されている OpenShift Container Platform クラスタを介して利用できるようにします。

- VMware vSphere
- ベアメタル
- Red Hat OpenStack Platform (テクノロジープレビュー)
- IBM Power
- IBM Z インフラストラクチャー

OpenShift Data Foundation Operator は、外部サービスに対する永続ボリューム (PV) およびオブジェクトバケット要求 (OBC) を満たすためにサービスを作成し、管理します。外部クラスタは、OpenShift Container Platform で実行されているアプリケーションのブロック、ファイル、およびオブジェクトストレージクラスを提供できます。Operator は、外部クラスタをデプロイまたは管理しません。

第5章 セキュリティーに関する考慮事項

5.1. FIPS-140-2

FIPS-140-2 (Federal Information Processing Standard Publication 140-2) は、暗号モジュールの使用についての一連のセキュリティー要件を定義する標準です。法律は、米国政府機関および請負業者にこの基準を義務付けており、他の国際および業界固有の基準でも参照されています。

Red Hat OpenShift Data Foundation は、FIPS 検証済みの暗号化モジュールを使用するようになりました。Red Hat Enterprise Linux OS/CoreOS(RHCOS) は、これらのモジュールを提供します。

現在、暗号化モジュール検証プログラム (CMVP) は暗号化モジュールを処理します。これらのモジュールの状態は、[プロセスリストのモジュール](#) で確認できます。最新情報は、Red Hat ナレッジベースソリューションの [RHEL コア暗号化コンポーネント](#) を参照してください。



注記

OpenShift Data Foundation をインストールする前に、OpenShift Container Platform で FIPS モードを有効にします。この機能は Red Hat Enterprise Linux 7 (RHEL 7) での OpenShift Data Foundation デプロイメントをサポートしていないため、OpenShift Container Platform は RHCOS ノードで実行する必要があります。

詳細は、OpenShift Container Platform ドキュメントの [インストールガイド](#) に記載されている **FIPS モードでのクラスタのインストール** および **FIPS 暗号化のサポート** を参照してください。

5.2. プロキシ環境

プロキシ環境は、インターネットへの直接アクセスを拒否し、代わりに利用可能な HTTP または HTTPS プロキシを提供する実稼働環境です。Red Hat OpenShift Container Platform は、既存クラスタのプロキシオブジェクトを変更するか、新規クラスタについて `install-config.yaml` ファイルでプロキシを設定してプロキシを使用するように設定されます。

Red Hat では、OpenShift Container Platform が [クラスタ全体のプリキシの設定](#) に応じて設定されている場合に、プロキシ環境での OpenShift Data Foundation のデプロイメントをサポートします。

5.3. データ暗号化オプション

暗号化を使用すると、必要な暗号化キーがなければデータを読み取ることができないようにデータをエンコードできます。このメカニズムは、物理メディアが手元から離れるような物理的なセキュリティー違反の発生時にもデータの機密性を保護できます。PV ごとの暗号化は、同じ OpenShift Container Platform クラスタ内の他の namespace からのアクセス保護も提供します。データはディスクに書き込まれる際に暗号化され、ディスクから読み取られる際に復号化されます。暗号化されたデータを使用すると、パフォーマンスに小規模なペナルティーのみが発生する可能性があります。

暗号化は、Red Hat OpenShift Data Foundation 4.6 以降を使用してデプロイされる新規クラスタでのみサポートされます。外部鍵管理システム (KMS) を使用していない既存の暗号化されたクラスタは、外部 KMS を使用するように移行できません。

現時点で、HashiCorp Vault はクラスタ全体の暗号化および永続ボリュームの暗号化で唯一サポートされている KMS です。OpenShift Data Foundation 4.7.0 および 4.7.1 では、HashiCorp Vault Key/Value (KV) シークレットエンジン API (バージョン 1) のみがサポートされます。OpenShift Data Foundation 4.7.2 以降では、HashiCorp Vault KV シークレットエンジン API (バージョン 1 および 2) がサポートされるようになりました。



重要

- KMS は、永続ボリューム (PV) 暗号化に必要であり、クラスター全体の暗号化にはオプションです。
- まず、ストレージクラスの暗号化には、有効な Red Hat OpenShift Data Foundation Advanced サブスクリプションが必要です。詳細は、[OpenShift Data Foundation サブスクリプションに関するナレッジベースの記事](#) を参照してください。

Red Hat はテクノロジーパートナーと連携して、本書をお客様へのサービスとして提供します。ただし、Red Hat では、Hashicorp 製品のサポートを提供していません。この製品に関するテクニカルサポートについては、[Hashicorp](#) にお問い合わせください。

5.3.1. クラスター全体の暗号化

Red Hat OpenShift Data Foundation は、ストレージクラスター内のディスクおよび Multicloud Object Gateway 操作のすべてに対して、クラスター全体の暗号化 (保存時の暗号化、encryption-at-rest) をサポートします。OpenShift Data Foundation は、キーのサイズが 512 ビットの Linux Unified Key System (LUKS) バージョン 2 ベースの暗号化と、各デバイスが異なる暗号化キーを持つ **aes-xts-plain64** 暗号を使用します。このキーは Kubernetes シークレットまたは外部 KMS を使用して保存されます。どちらのメソッドでも相互排他的であり、メソッド間の移行はできません。

暗号化はデフォルトでは無効にされます。デプロイメント時にクラスターの暗号化を有効にできます。詳細は、[デプロイメントガイド](#)を参照してください。

クラスター全体の暗号化は、キー管理システム (KMS) なしの OpenShift Data Foundation 4.6 でサポートされますが、OpenShift Data Foundation 4.7 以降では、KMS の有無にかかわらずサポートされます。



注記

有効な Red Hat OpenShift Data Foundation Advanced サブスクリプションが必要です。OpenShift Data Foundation のサブスクリプションがどのように機能するかを知るには、[OpenShift Data Foundation subscriptions に関するナレッジベースの記事](#) を参照してください。

HashiCorp Vault KMS を使用したクラスター全体の暗号化には、次の 2 つの認証方法があります。

- **トークン:** このメソッドでは、vault トークンを使用した認証が可能です。Vault トークンを含む kubernetes シークレットは、openshift-storage namespace で作成され、認証に使用されます。この認証方法を選択した場合、管理者は、暗号化キーが保存されている Vault のバックエンドパスへのアクセスを提供する Vault トークンを提供する必要があります。
- **Kubernetes:** このメソッドでは、serviceaccounts を使用して Vault で認証できます。この認証方法を選択した場合、管理者は、暗号化キーが保存されているバックエンドパスへのアクセスを提供する Vault で設定されたロールの名前を指定する必要があります。次に、このロールの値が **ocs-kms-connection-details** Config Map に追加されます。このメソッドは、OpenShift Data Foundation 4.10 から利用できます。

現時点で、HashiCorp Vault は唯一サポートされている KMS です。OpenShift Data Foundation 4.7.0 および 4.7.1 では、HashiCorp Vault KV シークレットエンジン API (バージョン 1) のみがサポートされます。OpenShift Data Foundation 4.7.2 以降では、HashiCorp Vault KV シークレットエンジン API (バージョン 1 および 2) がサポートされるようになりました。



注記

IBM Cloud プラットフォーム上の OpenShift Data Foundation は、HashiCorp Vault KMS に加えて、暗号化ソリューションとして Hyper Protect Crypto Services (HPCS) Key Management Services (KMS) をサポートします。



重要

Red Hat はテクノロジーパートナーと連携して、本書をお客様へのサービスとして提供します。ただし、Red Hat では、Hashicorp 製品のサポートを提供していません。この製品に関するテクニカルサポートについては、[Hashicorp](#) にお問い合わせください。

5.3.2. ストレージクラスの暗号化

デバイスの暗号化キーを保存するために外部の鍵管理システム (KMS) を使用して、ストレージクラスの暗号化で永続ボリューム (ブロックのみ) を暗号化できます。永続ボリュームの暗号化は RADOS Block Device (RBD) 永続ボリュームでのみ利用できます。[永続ボリュームの暗号化を使用したストレージクラスの作成方法](#) を参照してください。

ストレージクラスの暗号化は OpenShift Data Foundation 4.7 以降でサポートされます。



注記

有効な Red Hat OpenShift Data Foundation Advanced サブスクリプションが必要です。OpenShift Data Foundation のサブスクリプションがどのように機能するかを知るには、[OpenShift Data Foundation subscriptions に関するナレッジベースの記事](#) を参照してください。

第6章 サブスクリプション

6.1. サブスクリプションのオフアリング

Red Hat OpenShift Data Foundation のサブスクリプションは、OpenShift Container Platform と同様にコアのペアをベースとして提供されます。Red Hat OpenShift Data Foundation 2 コアサブスクリプションは、OpenShift Container Platform が実行されるシステムの CPU 上の論理コア数をベースとしています。

以下の点は、OpenShift Container Platform と同様です。

- OpenShift Data Foundation サブスクリプションは、大規模なホストに対応するようにスタック可能です。
- コアは、必要に応じて多数の仮想マシン (VM) に分散できます。たとえば、10 の 2 コアサブスクリプションは 20 コアを提供し、IBM Power の場合、SMT レベル 8 の 2 コアのサブスクリプションは、任意の数の仮想マシンで使用できる 2 コアまたは 16 vCPU が提供されます。
- OpenShift Data Foundation サブスクリプションは、Premium または Standard サポートで利用できます。

6.2. 障害復旧サブスクリプションの要件

Red Hat OpenShift Data Foundation でサポートされる障害復旧機能では、障害復旧ソリューションを正常に実装するために以下の前提条件をすべて満たす必要があります。

- 有効な Red Hat OpenShift Data Foundation Advanced エンタイトルメント
- 有効な Red Hat Advanced Cluster Management for Kubernetes サブスクリプション

ソースまたは宛先としてアクティブレプリケーションに参加している PV を含む Red Hat OpenShift Data Foundation クラスターには、OpenShift Data Foundation Advanced エンタイトルメントが必要です。このサブスクリプションは、ソースクラスターと宛先クラスターの両方でアクティブにする必要があります。

OpenShift Data Foundation のサブスクリプションがどのように機能するかを知るには、[OpenShift Data Foundation subscriptions に関するナレッジベースの記事](#) を参照してください。

6.3. コア対 vCPU およびハイパースレッディング

現時点で特定のシステムが1つまたは複数のコアを消費するかどうかについての決定は、そのシステムでハイパースレッディング機能を利用できるかどうかによって異なります。ハイパースレッディングは Intel CPU のみの機能です。Red Hat カスタマーポータルにアクセスし、特定のシステムがハイパースレッディングをサポートしているかどうかを判断します。

ハイパースレッディングが有効にされており、1つのハイパースレッドが1つの利用可能なシステムコアに等しいシステムの場合、[コアの計算](#) は 2 コア対 4 vCPU の比率になります。したがって、2 コアのサブスクリプションは、ハイパースレッドシステムの 4 vCPU に対応します。大規模な仮想マシン (VM) には、4 サブスクリプションコアに相当する 8 vCPU がある場合があります。サブスクリプションは 2 コア単位で提供されるため、4 コアまたは 8 vCPU に対応するには 2 つの 2 コアサブスクリプションが必要になります。

ハイパースレッディングが有効にされていない場合や、表示される各システムのコアが基礎となる物理コアに直接関連する場合、コアの計算は 2 コア対 2 vCPU の比率になります。

6.3.1. IBM Power のコア対 vCPU および同時マルチスレッド (SMT)

現時点で特定のシステムが1つまたは複数のコアを消費するかどうかについての決定は、同時マルチスレッド (SMT) のレベルによって異なります。IBM Power は、以下の表にあるように、各コアの同時マルチスレッドレベルの1、2、4、または8を提供します。これは vCPU の数に対応します。

表6.1 さまざまな SMT レベルとそれに対応する vCPU

SMT レベル	SMT=1	SMT=2	SMT=4	SMT=8
1 コア	# vCPU=1	# vCPU=2	# vCPU=4	# vCPU=8
2 コア	# vCPU=2	# vCPU=4	# vCPU=8	# vCPU=16
4 コア	# vCPU=4	# vCPU=8	# vCPU=16	# vCPU=32

SMT が設定されたシステムでは、サブスクリプションに必要なコア数の計算は SMT レベルによって異なります。したがって、2 コアのサブスクリプションは、上記の表に示すように SMT レベル 1 の 2 vCPU に対応し、SMT レベル 2 の 4 vCPU に、SMT レベル 4 では 8 vCPU に、SMT レベル 8 では 16 vCPU に対応します。大規模な仮想マシン (VM) には 16 vCPU が含まれる場合があります。この場合、SMT レベル 8 では SMT レベルで vCPU の # を除算した計算により 2 コアサブスクリプションが必要になります (SMT-8 の場合: $16 \text{ vCPU} / 8 = 2$)。サブスクリプションは 2 コア単位で提供されるため、これらの 2 コアまたは 16 vCPU に対応するには 1 つの 2 コアサブスクリプションが必要になります。

6.4. コアの分割

奇数のコアを必要とするシステムの場合でも、2 コアのサブスクリプションを使用する必要があります。たとえば、必要なコアが1つだけであると計算されたシステムは、登録およびサブスクライブされると、最終的に 2 コアのサブスクリプションを完全に消費することになります。

2 vCPU を持つ単一の仮想マシン (VM) がハイパースレッディング機能を使用し、1 vCPU が計算される場合、2 コアサブスクリプションが必要になります。単一の 2 コアサブスクリプションは、ハイパースレッディングを使用する 2 vCPU を持つ 2 仮想マシン間で分割することはできません。詳細は、[Cores versus vCPUs and hyperthreading](#) セクションを参照してください。

そのため、仮想インスタンスは偶数のコアを必要とするようにサイズ設定することが推奨されます。

6.4.1. IBM Power 用の共有プロセッサプール

IBM Power には、共有プロセッサプールの概念があります。共有プロセッサプール内のプロセッサは、クラスター内のノード間で共有できます。Red Hat OpenShift Data Foundation に必要な集約コンピュータ容量はコアのペアの倍数である必要があります。

6.5. サブスクリプションの要件

Red Hat OpenShift Data Foundation コンポーネントは、Red Hat CoreOS (RHCOS) または Red Hat Enterprise Linux (RHEL) 8.4 のいずれかをホストのオペレーティングシステムとして使用できる OpenShift Container Platform ワーカーまたはインフラストラクチャーノードのいずれかで実行できます。RHEL 7 は非推奨になりました。OpenShift Data Foundation サブスクリプションは、すべての OpenShift Container Platform をサブスクライブするコアに 1:1 の割合で必要です。

インフラストラクチャーノードを使用する場合、OpenShift Container Platform または OpenShift Data Foundation サブスクリプションが必要でなくても、OpenShift Data Foundation のすべての OpenShift

ワーカーノードコアをサブスクライブするルールが適用されます。ラベルを使用して、ノードがワーカーノードまたはインフラストラクチャーノードであるかどうか示すことができます。

詳細は、[ストレージリソースの管理および割り当て ガイド](#)の [Red Hat OpenShift Data Foundation に専用のワーカーノードを使用する方法](#) を参照してください。

第7章 インフラストラクチャーの要件

7.1. プラットフォーム要件

Red Hat OpenShift Data Foundation 4.11 は、OpenShift Container Platform バージョン 4.11 およびその次のマイナーバージョンでのみサポートされます。

以前のバージョンの Red Hat OpenShift Data Foundation に関するバグ修正は、バグ修正バージョンとしてリリースされます。詳細は、[Red Hat OpenShift Container Platform ライフサイクルポリシー](#) を参照してください。

外部クラスターのサブスクリプション要件については、Red Hat ナレッジベースのソリューション [Which Red Hat subscriptions do I need, to get Red Hat OpenShift Container Storage support?](#) を参照してください。

サポートされているプラットフォームバージョンの完全なリストについては、[Red Hat OpenShift Data Foundation Supportability and Interoperability Checker](#) を参照してください。

7.1.1. Amazon EC2

内部 Red Hat OpenShift Data Foundation クラスターのみをサポートします。

内部クラスターは、[ストレージデバイス要件](#) の両方を満たし、aws-ebs プロビジョナーを介して EBS ストレージを提供するストレージクラスを備えている必要があります。

OpenShift Data Foundation バージョン 4.10 以降は、gp2 および gp3 CSI ドライバーをサポートします。Amazon Web Services (AWS) は、これらの CSI ドライバーを導入しました。これらのドライバーは、ストレージ拡張機能を改善し (gp2 インツリーと比較して gp2 CSI)、月額料金を引き下げます (gp3)。OpenShift Data Foundation 4.10 でストレージクラスを選択するときに、新しいドライバーを選択できるようになりました。gp2 がデフォルトです。

7.1.2. ベアメタル

内部クラスターをサポートし、外部クラスターの使用をサポートします。

内部クラスターは、[ストレージデバイスの要件](#) を満たし、ローカルストレージ Operator 経由でローカル SSD (NVMe/SATA/SAS、SAN) を提供するストレージクラスがなければなりません。

7.1.3. VMware vSphere

内部クラスターをサポートし、外部クラスターの使用をサポートします。

推奨されるバージョン:

- vSphere 6.7、Update2 以降
- vSphere 7.0 以降

詳細は、[VMware vSphere インフラストラクチャーの要件](#) を参照してください。



注記

VMware ESXi がデバイスをフラッシュとして認識しない場合は、それらをフラッシュデバイスとしてマークします。Red Hat OpenShift Data Foundation をデプロイする前に、[ストレージデバイスをフラッシュとしてマーク](#) を参照してください。

さらに、内部クラスターは [ストレージデバイスの要件](#) を満たし、以下のいずれかを提供するストレージクラスがなければなりません。

- vSAN または VMFS データストア (vsphere-volume プロビジョナー経由)
- VMDK、RDM、または DirectPath ストレージデバイス (Local Storage Operator 経由)

7.1.4. Microsoft Azure

内部 Red Hat OpenShift Data Foundation クラスターのみをサポートします。

内部クラスターは、[ストレージデバイス要件](#) の両方を満たし、azure-disk プロビジョナーを介して zzure ディスクを提供するストレージクラスを備えている必要があります。

7.1.5. Google Cloud [テクノロジープレビュー]

内部 Red Hat OpenShift Data Foundation クラスターのみをサポートします。

内部クラスターは、[ストレージデバイス要件](#) の両方を満たし、gce-pd プロビジョナーを介して GCE Persistent Disk を提供するストレージクラスを備えている必要があります。

7.1.6. Red Hat Virtualization プラットフォーム

内部 Red Hat OpenShift Data Foundation クラスターのみをサポートします。

内部クラスターは、[ストレージデバイスの要件](#) を満たし、ローカルストレージ Operator 経由でローカル SSD (NVMe/SATA/SAS、SAN) を提供するストレージクラスがなければなりません。

7.1.7. Red Hat OpenStack Platform [テクノロジープレビュー]

内部 Red Hat OpenShift Data Foundation クラスターをサポートし、外部クラスターを使用します。

内部クラスターは、[ストレージデバイス要件](#) の両方を満たし、Cinder プロビジョナーを介して標準ディスクを提供するストレージクラスを備えている必要があります。

7.1.8. IBM Power

内部 Red Hat OpenShift Data Foundation クラスターをサポートし、外部クラスターを使用します。

内部クラスターは、[ストレージデバイスの要件](#) を満たし、ローカルストレージ Operator 経由でローカル SSD (NVMe/SATA/SAS、SAN) を提供するストレージクラスがなければなりません。

7.1.9. IBM Z および LinuxONE

内部 Red Hat OpenShift Data Foundation クラスターのみをサポートします。

内部クラスターは、[ストレージデバイスの要件](#) を満たし、ローカルストレージ Operator 経由でローカル SSD (NVMe/SATA/SAS、SAN) を提供するストレージクラスがなければなりません。

7.2. 外部モード要件

7.2.1. Red Hat Ceph Storage

外部モードの Red Hat OpenShift Data Foundation および Red Hat Ceph Storage (RHCS) のサポートと相互運用性を確認するには、[Red Hat OpenShift Data Foundation サポートおよび相互運用性チェックカー](#) のラボにアクセスします。

1. **ODF as Self-Managed Service** として **Service Type** を選択します。
2. ドロップダウンから適切な **Version** を選択します。
3. Versions タブで、**Supported RHCS Compatibility** タブをクリックします。

RHCS クラスターのインストール方法は、[インストールガイド](#) を参照してください。

7.2.2. IBM FlashSystem

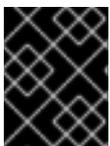
IBM FlashSystem を他のプロバイダーのプラグ可能な外部ストレージとして使用するには、最初に OpenShift Data Foundation をデプロイする必要があります。これは、IBM FlashSystem ストレージクラスをバックングストレージとして使用します。

サポートされている最新の FlashSystem 製品およびバージョンについては、[IBM ドキュメンテーション](#) の Spectrum Virtualize ファミリー製品ドキュメンテーション内のリファレンス > Red Hat OpenShift Data Foundation サポートの要約を参照してください。

OpenShift Data Foundation をデプロイする方法については、外部 IBMFlash System ストレージ用の [Creating an OpenShift Data Foundation Cluster for external IBM FlashSystem storage](#) を参照してください。

7.3. リソース要件

Red Hat OpenShift Data Foundation サービスは、ベースサービスの初期セットで設定され、追加のデバイスセットで拡張できます。これらすべての Red Hat OpenShift Data Foundation サービス Pod は、[リソース要件](#) に応じて OpenShift Container Platform ノードの kubernetes によってスケジュールされます。クラスターを (障害ドメインごとに1ノード) 3の倍数に拡張する方法は、[Pod の配置ルール](#) を簡単に満たす方法です。



重要

これらの要件は、OpenShift Data Foundation サービスのみに関連し、これらのノードで実行している他のサービス、オペレーター、またはワークロードには関連しません。

表7.1 Red Hat OpenShift Data Foundation のみの利用可能なリソース要件の集約

デプロイメントモード	ベースサービス	追加のデバイスセット
内部	<ul style="list-style-type: none"> ● 30 個の CPU (論理) ● 72 GiB メモリー ● 3 ストレージデバイス 	<ul style="list-style-type: none"> ● 6 個の CPU (論理) ● 15 GiB メモリー ● 3 ストレージデバイス

デプロイメントモード	ベースサービス	追加のデバイスセット
外部	<ul style="list-style-type: none"> ● 4 個の CPU (論理) ● 16 GiB メモリー 	該当なし

例: 単一デバイスセットを持つ内部モードデプロイメントの 3 ノードクラスターの場合、最小の $3 \times 10 = 30$ ユニットの CPU が必要です。

詳細は、[6章 サブスクリプション](#) および [CPU ユニット](#) を参照してください。

Red Hat OpenShift Data Foundation クラスターの設計に関する追加のガイダンスは、[ODF Sizing Tool](#) を参照してください。

CPU ユニット

本セクションでは、1 CPU ユニットは Kubernetes コンセプトの 1 CPU ユニットにマップされます。

- CPU の 1 ユニットは、ハイパースレッディングされていない CPU の 1 コアに相当します。
- CPU の 2 ユニットは、ハイパースレッディングされている CPU の 1 コアに相当します。
- Red Hat OpenShift Data Foundation コアベースのサブスクリプションは常にペア (2 コア) で提供されます。

表7.2 IBM Power の最小リソース要件の集約

デプロイメントモード	ベースサービス
内部	<ul style="list-style-type: none"> ● 48 CPU (論理) ● 192 GiB メモリー ● 3 つのストレージデバイス (それぞれに追加の 500GB ディスクが含まれる)
外部	<ul style="list-style-type: none"> ● 24 個の CPU (論理) ● 48 GiB メモリー

例: 内部接続デバイスモードのデプロイメントの 3 ノードクラスターの場合、最小の $3 \times 16 = 48$ ユニットの CPU、および $3 \times 64 = 192$ GB が必要です。

7.3.1. IBM Z および LinuxONE インフラストラクチャーのリソース要件

Red Hat OpenShift Data Foundation サービスは、ベースサービスの初期セットで設定され、追加のデバイスセットで拡張できます。

これらすべての Red Hat OpenShift Data Foundation サービス Pod は、[リソース要件](#)に応じて OpenShift Container Platform ノードの kubernetes によってスケジュールされます。

クラスターを (障害ドメインごとに1ノード) 3 の倍数に拡張する方法は、Pod の配置ルールを簡単に満たす方法です。

表7.3 Red Hat OpenShift Data Foundation でのみ利用可能なリソース要件を集約 (IBM Z および LinuxONE)

デプロイメントモード	ベースサービス	追加のデバイスセット	IBM Z および LinuxONE の最小ハードウェア要件
内部	<ul style="list-style-type: none"> ● 30 個の CPU (論理) <ul style="list-style-type: none"> ○ それぞれ 10 個の CPU (論理) を備えた 3 つの ノード ● 72 GiB メモリー ● 3 ストレージデバイス 	<ul style="list-style-type: none"> ● 6 個の CPU (論理) ● 15 GiB メモリー ● 3 ストレージデバイス 	1 IFL
外部	<ul style="list-style-type: none"> ● 4 個の CPU (論理) ● 16 GiB メモリー 	該当なし	該当なし

CPU

ハイパーバイザー、IBM z/VM、カーネル仮想マシン (KVM)、またはその両方で定義されている仮想コアの数です。

IFL(Linux 向けの統合機能)

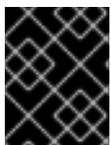
IBM Z および LinuxONE の物理コアです。

最小システム環境

- 1 つの論理パーティション (LPAR) で最小クラスターを動作させるには、6 つの IFL の上に追加の IFL が必要です。OpenShift Container Platform は、これらの IFL を使用します。

7.3.2. デプロイメントリソースの最小要件

OpenShift Data Foundation クラスターは、標準のデプロイメントリソース要件を満たしていない場合に、最小の設定でデプロイされます。



重要

これらの要件は、OpenShift Data Foundation サービスのみに関連し、これらのノードで実行している他のサービス、Operator、またはワークロードには関連しません。

表7.4 OpenShift Data Foundation のみのリソース要件の集約

デプロイメントモード	ベースサービス
内部	<ul style="list-style-type: none"> ● 24 個の CPU (論理) ● 72 GiB メモリー ● 3 ストレージデバイス

デバイスセットを追加する場合は、最小デプロイメントを標準デプロイメントに変換することが推奨されます。

7.3.3. コンパクトなデプロイメントリソース要件

Red Hat OpenShift Data Foundation は、3 ノードの OpenShift のコンパクトなベアメタルクラスターにインストールできます。ここでは、すべてのワークロードが3つの強力なマスターノードで実行されます。ワーカーノードまたはストレージノードは含まれません。



重要

これらの要件は、OpenShift Data Foundation サービスのみに関連し、これらのノードで実行している他のサービス、Operator、またはワークロードには関連しません。

表7.5 OpenShift Data Foundation のみのリソース要件の集約

デプロイメントモード	ベースサービス	追加のデバイスセット
内部	<ul style="list-style-type: none"> ● 24 個の CPU (論理) ● 72 GiB メモリー ● 3 ストレージデバイス 	<ul style="list-style-type: none"> ● 6 個の CPU (論理) ● 15 GiB メモリー ● 3 ストレージデバイス

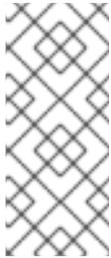
コンパクトのベアメタルクラスターで OpenShift Container Platform を設定する場合は、[Configuring a three-node cluster](#) および [Delivering a Three-node Architecture for Edge Deployments](#) を参照してください。

7.3.4. MCG のみのデプロイメントのリソース要件

Multicloud Object Gateway (MCG) コンポーネントのみを使用してデプロイされた OpenShift Data Foundation クラスターは、デプロイメントに柔軟性を提供し、リソース消費を削減するのに役立ちます。

表7.6 MCG のみのデプロイメントの総リソース要件

デプロイメントモード	Core	データベース (DB)	Endpoint (エンドポイント)

デプロイメントモード	Core	データベース (DB)	Endpoint (エンドポイント)
内部	<ul style="list-style-type: none"> ● 1 CPU ● 4 GiB メモリー 	<ul style="list-style-type: none"> ● 0.5 CPU ● 4 GiB メモリー 	<ul style="list-style-type: none"> ● 1 CPU ● 2 GiB メモリー  <p>注記</p> <p>デフォルトオートスケールは1～2です。</p>

7.3.5. ネットワークファイルシステムを使用するためのリソース要件

Red Hat OpenShift Data Foundation バージョン 4.11 のリリースにより、ネットワークファイルシステム (NFS) を使用してエクスポートを作成し、テクノロジープレビューとして OpenShift クラスタから外部からアクセスできるようになりました。この機能を使用する場合、NFS サービスは3つの CPU と 8Gi の RAM を消費します。NFS はオプションであり、デフォルトでは無効になっています。

NFS ボリュームには、次の2つの方法でアクセスできます。

- クラスタ内: Openshift クラスタ内のアプリケーション Pod による。
- cluster: Openshift クラスタ外から。

NFS 機能の詳細は、[NFS を使用したエクスポートの作成](#) を参照してください。



重要

NFS を使用したエクスポートの作成はテクノロジープレビュー機能です。テクノロジープレビュー機能は、Red Hat 製品のサービスレベルアグリーメント (SLA) の対象外であり、機能的に完全ではないことがあります。Red Hat は実稼働環境でこれらを使用することを推奨していません。これらの機能は、近々発表予定の製品機能をリリースに先駆けてご提供することにより、お客様は機能性をテストし、開発プロセス中にフィードバックをお寄せいただくことができます。

詳細は、[テクノロジープレビュー機能のサポート範囲](#) を参照してください。

7.4. POD の配置ルール

Kubernetes は、宣言型の配置ルールに基づいて Pod の配置を行います。内部クラスタの Red Hat OpenShift Data Foundation ベースサービスの配置ルールは、以下のように要約できます。

- ノードには **cluster.ocs.openshift.io/openshift-storage** キーでラベルが付けられます。
- ノードは、擬似障害ドメインに分類されます (何も存在しない場合)。
- 高可用性が必要なコンポーネントは障害ドメインに分散されます。

- ストレージデバイスはそれぞれの障害ドメインでアクセスできる必要があります。

これにより、少なくとも3つのノードがあり、既存の [トポロジーラベル](#) が存在する場合にノードは3つの異なるラックまたはゾーン障害ドメインにある必要があります。

追加のデバイスセットについては、3つの障害ドメインのそれぞれにストレージデバイスがあり、Podが消費するのに十分なリソースが必要になります。手動の配置ルールはデフォルトの配置ルールを上書きするのに使用できますが、通常この方法はベアメタルのデプロイメントにのみ適しています。

7.5. ストレージデバイスの要件

このセクションでは、内部モードのデプロイメントおよびアップグレードの計画時に考慮できる各種のストレージ容量の要件について説明します。通常、ノードごとに9以下のデバイスが推奨されます。この推奨事項により、ノードがクラウドプロバイダーの動的ストレージデバイスの割り当て制限下であり、ローカルストレージデバイスに関連してノードに障害が発生した後の復旧時間を制限できます。クラスターを(障害ドメインごとに1ノード)3の倍数に拡張する方法は、[Podの配置ルール](#) を簡単に満たす方法です。

ストレージノードには少なくとも2つのディスクが必要です。1つはオペレーティングシステム用で、残りのディスクは OpenShift Data Foundation コンポーネント用です。



注記

ストレージ容量は、インストール時に選択した容量の増分値でのみ拡張できます。

7.5.1. 動的ストレージデバイス

Red Hat OpenShift Data Foundation では、動的ストレージデバイスサイズの要求サイズとして 0.5 TiB、2 TiB または 4 TiB の容量を選択できます。ノードごとに実行できる動的ストレージデバイスの数は、ノードサイズ、基礎となるプロビジョナー制限、および [リソース要件](#) の関数です。

7.5.2. ローカルストレージデバイス

ローカルストレージのデプロイメントの場合、4 TiB 以下のディスクサイズを使用でき、すべてのディスクが同じサイズおよび種類である必要があります。ノードごとに実行できるローカルストレージデバイスの数は、ノードのサイズと [リソース要件](#) によって決まります。クラスターを(障害ドメインごとに1ノード)3の倍数に拡張する方法は、[Podの配置ルール](#) を簡単に満たす方法です。



注記

ディスクのパーティション設定はサポートされません。

7.5.3. 容量のプランニング

使用する前に、利用可能なストレージ容量を必ず確保するようにしてください。利用可能なストレージ容量が完全に使い切られる場合はリカバリーが難しく、単に容量を追加したり、コンテンツを削除したり、移行したりするよりも多くの介入が必要になります。

容量アラートは、クラスターストレージ容量が合計容量の 75% (ほぼ一杯) および 85% (一杯) になると発行されます。容量についての警告に常に迅速に対応し、ストレージを定期的を確認して、ストレージ領域が不足しないようにします。75% (ほぼフル) に達したら、スペースを解放するか、クラスターを拡張します。85% (フル) アラートに達すると、ストレージ領域が完全に不足していて、標準コマンドを使用して領域を解放できないことが示唆されます。この時点で、[Red Hat カスタマーサポート](#) にお問い合わせください。

次の表は、動的ストレージデバイスを使用した Red Hat OpenShift Data Foundation のノード設定の例を示しています。

表7.7 3つのノードで設定される初期設定の例

ストレージデバイスのサイズ	ノードあたりのストレージデバイス	合計容量	利用可能なストレージ容量
0.5 TiB	1	1.5 TiB	0.5 TiB
2 TiB	1	6 TiB	2 TiB
4 TiB	1	12 TiB	4 TiB

表7.8 30 ノード (N) で拡張された設定の例

ストレージデバイスのサイズ (D)	ノードごとのストレージデバイス (M)	合計容量 (D * M * N)	使用可能なストレージ容量 (D*M*N/3)
0.5 TiB	3	45 TiB	15 TiB
2 TiB	6	360 TiB	120 TiB
4 TiB	9	1080 TiB	360 TiB

7.6. マルチネットワークプラグイン (MULTUS) のサポート [テクノロジープレビュー]

デフォルトで、Red Hat OpenShift Data Foundation は Red Hat OpenShift Software Defined Network (SDN) を使用するように設定されています。このデフォルト設定では、SDN には以下のトラフィックタイプがあります。

- Pod 対 Pod のトラフィック
- OpenShift Data Foundation パブリックネットワークトラフィックとして知られる OpenShift Data Foundation トラフィックへの Pod
- OpenShift Data Foundation クラスターネットワークトラフィックとして知られる OpenShift Data Foundation のレプリケーションおよびリバランス

ただし、OpenShift Data Foundation 4.8 以降では、テクノロジープレビューとして、さまざまな種類のネットワークトラフィックを分離することで、Multus を使用してセキュリティーやパフォーマンスを強化する機能がサポートされます。



重要

Multus のサポートはテクノロジープレビュー機能としてのみサポートされ、ベアメタルおよび VMWare デプロイメントでテストされます。テクノロジープレビュー機能は、Red Hat 製品のサービスレベルアグリーメント (SLA) の対象外であり、機能的に完全ではないことがあります。Red Hat は実稼働環境でこれらを使用することを推奨していません。これらの機能は、近々発表予定の製品機能をリリースに先駆けてご提供することにより、お客様は機能性をテストし、開発プロセス中にフィードバックをお寄せいただくことができます。

詳細は、[テクノロジープレビュー機能のサポート範囲](#) を参照してください。

7.6.1. 複数ネットワークについて

Kubernetes では、コンテナネットワークは Container Network Interface (CNI) を実装するネットワークプラグインに委任されます。

OpenShift Container Platform は、Multus CNI プラグインを使用して CNI プラグインのチェーンを許可します。クラスターのインストール時に、**デフォルト** の Pod ネットワークを設定します。デフォルトのネットワークは、クラスターのすべての通常のネットワークトラフィックを処理します。利用可能な CNI プラグインに基づいて **追加のネットワーク** を定義し、1つまたは複数のネットワークを Pod に割り当てることができます。必要に応じて、クラスターの複数のネットワークを追加で定義できます。これにより、スイッチングやルーティングなどのネットワーク機能を提供する Pod を設定する際に柔軟性が得られます。

7.6.1.1. 追加ネットワークの使用シナリオ

データプレーンとコントロールプレーンの分離など、ネットワークの分離が必要な状況で追加のネットワークを使用できます。トラフィックの分離は、以下のようなパフォーマンスおよびセキュリティー関連の理由で必要になります。

パフォーマンス

各プレーンのトラフィック量を管理するために、2つの異なるプレーンにトラフィックを送信できます。

セキュリティー

機密トラフィックは、セキュリティー上の考慮に基づいて管理されているネットワークに送信でき、テナントまたはカスタマー間で共有できないプライベートを分離することができます。

クラスターのすべての Pod はクラスター全体のデフォルトネットワークを依然として使用し、クラスター全体での接続性を維持します。すべての Pod には、クラスター全体の Pod ネットワークに割り当てられる **eth0** インターフェイスがあります。Pod のインターフェイスは、**oc exec -it <pod_name> -- ip a** コマンドを使用して表示できます。Multus CNI を使用するネットワークを追加する場合、それらの名前は **net1**、**net2**、...、**netN** になります。

追加のネットワークを Pod に割り当てるには、インターフェイスの割り当て方法を定義する設定を作成する必要があります。それぞれのインターフェイスは、**NetworkAttachmentDefinition** カスタムリソース (CR) を使用して指定します。これらの CR のそれぞれにある CNI 設定は、インターフェイスの作成方法を定義します。

7.6.2. Multus を使用したストレージトラフィックの分離

Multus を使用するには、OpenShift Data Foundation クラスターをデプロイする前に、後にクラスターにアタッチされる Network attachment definition (NAD) を作成する必要があります。詳細は以下を参照してください。

- ベアメタルの場合は [Creating network attachment definitions](#)
- VMware の場合は [Creating network attachment definitions](#)

Multus を使用すると、ハードウェアの設定や VMWare インスタンスのネットワーク設定に応じて、以下の設定が可能です。

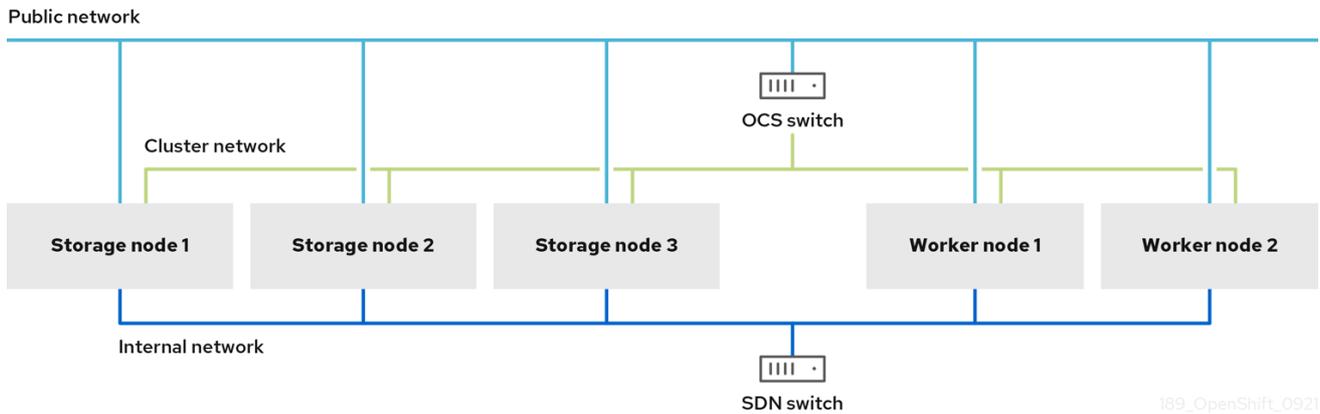
- デュアルネットワークインターフェイスを持つノードの推奨設定
 - 分離されたストレージトラフィック
 - OpenShift SDN に1つのインターフェイスを設定 (Pod 対 Pod のトラフィック)
 - すべての OpenShift Data Foundation トラフィックに対して1つのインターフェイスを設定
- トリプルネットワークインターフェイスを持つノードの推奨設定
 - 完全なトラフィック分離
 - OpenShift SDN に1つのインターフェイスを設定 (Pod 対 Pod のトラフィック)
 - OpenShift Data Foundation トラフィック (OpenShift Data Foundation public トラフィック) へのすべての Pod に1つのインターフェイスを設定します。
 - すべての OpenShift Data Foundation レプリケーションおよびリバランストラフィックに1つのインターフェイスを設定 (OpenShift Data Foundation クラスタトラフィック)

7.6.3. Multus 設定の推奨ネットワーク設定および要件

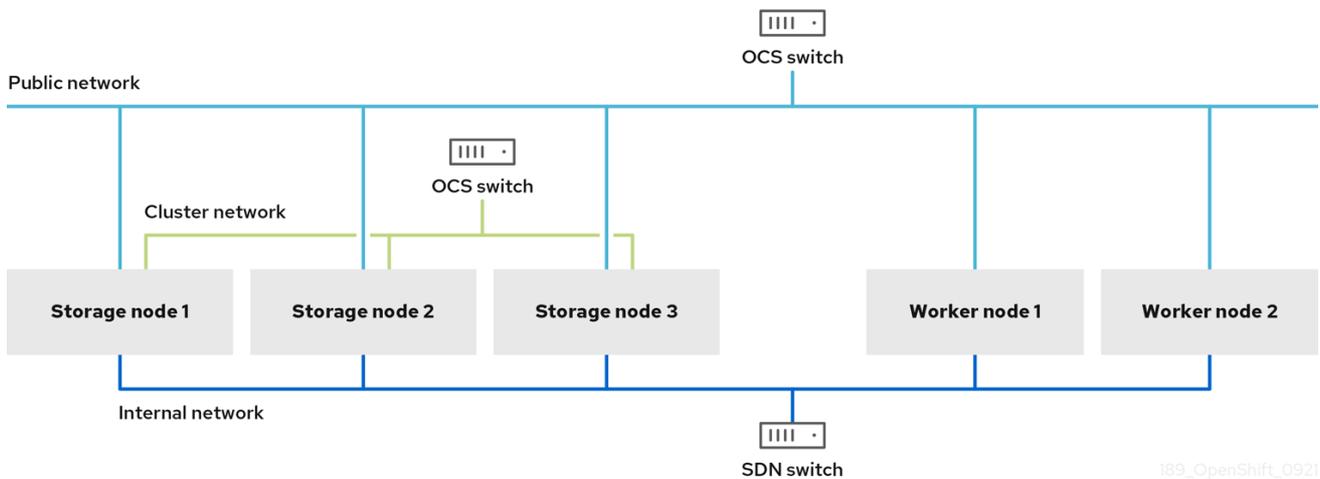
Multus 設定を利用する場合は、以下の前提条件を満たす必要があります。

- Multus 設定を完全に機能させるため、OpenShift Data Foundation のデプロイに使用されるすべてのノードには、同じネットワークインターフェイス設定が必要です。すべてのノードのネットワークインターフェイス名は同じにする必要があります。Multus パブリックネットワークおよび Multus クラスタネットワークの同じ基礎となるスイッチメカニズムに接続されている必要があります。
- Multus 設定を完全に機能させるため、永続ストレージに OpenShift Data Foundation を利用するアプリケーションをデプロイするために使用されるワーカーノードはすべて、同じネットワークインターフェイス設定を持つ必要があります。2つのインターフェイスの1つは、Storage ノードに Multus パブリックネットワークを設定するために使用されるインターフェイス名と同じである必要があります。すべてのワーカーネットワークインターフェイスが、Storage ノードの Multus パブリックネットワークに使用されるものと同じ基礎となるスイッチメカニズムに接続されている必要があります。

デュアルネットワークインターフェイスの分離設定の概略例:



デュアルネットワークインターフェースの分離設定の概略例:



注記

OpenShift Data Foundation OSD が実行されているストレージノードのみに、Multus によって設定された OpenShift Data Foundation クラスターネットワークへのアクセスが必要になります。

ベアメタルで Multus ベースの設定に必要な手順については、[Creating Multus networks](#) を参照してください。

VMware で Multus ベースの設定に必要な手順については、[Creating Multus networks](#) を参照してください。

第8章 障害復旧

障害復旧 (DR) は、中断または障害が発生する場合に、組織がビジネスクリティカルな機能または通常の運用を回復し、再開するのに役立ちます。OpenShift Data Foundation は、ステートフルアプリに高可用性 (HA) および DR ソリューションを提供します。これらのソリューションは、大きく 3 つのカテゴリーに分類されます。

- Regional-DR: 潜在的なデータ損失を最小限に抑えた地域間保護 [テクノロジープレビュー]
- Metro-DR: データ損失のない単一リージョンおよびクロスデータセンターの保護 [テクノロジープレビュー]
- Stretched Cluster - Arbiter: 単一の OpenShift Data Foundation クラスターが 2 つの異なる場所間で拡張され、ストレージインフラストラクチャーに災害復旧機能を提供しますテクノロジープレビュー

8.1. REGIONAL-DR [テクノロジープレビュー]

Regional-DR は、Red Hat Advanced Cluster Management for Kubernetes (RHACM) と OpenShift Data Foundation コンポーネントで設定され、OpenShift Container Platform クラスター全体でアプリケーションとデータのモビリティを提供します。非同期データレプリケーションに基づいて構築されているため、データが失われる可能性があります。さまざまな障害に対する保護を提供します。

Red Hat OpenShift Data Foundation は、ストレージプロバイダーとして Ceph に支えられており、そのライフサイクルは Rook によって管理されており、次の機能で強化されています。

- ミラーリングのプールを有効にする
- RBD プール間でイメージを自動的にミラーリングする
- Persistent Volume Claim ミラーリングごとに管理する csi アドオンを提供する

このリリースの Regional-DR は、さまざまなリージョンおよびデータセンターに展開されるマルチクラスター設定をサポートします。たとえば、2 つの異なるリージョンまたはデータセンターにある 2 つのマネージドクラスターでの 2 方向のレプリケーションをサポートします。このソリューションには、Red Hat Advanced Cluster Management (RHACM) と OpenShift Data Foundation Advanced SKU および関連するバンドルが含まれています。

前提条件

Red Hat OpenShift Data Foundation でサポートされる障害復旧機能では、障害復旧ソリューションを正常に実装するために以下の前提条件をすべて満たす必要があります。

- 有効な Red Hat OpenShift Data Foundation Advanced エンタイトルメント
- 有効な Red Hat Advanced Cluster Management for Kubernetes サブスクリプション

OpenShift Data Foundation のサブスクリプションがどのように機能するかを知るには、[OpenShift Data Foundation subscriptions に関するナレッジベースの記事](#) を参照してください。

詳細なソリューション要件は、[Regional-DR requirements](#) および [RHACM requirements](#) を参照してください。



重要

Advanced Cluster Management を使用した Regional-DR 用の OpenShift Data Foundation の設定は、テクノロジープレビュー機能であり、テクノロジープレビューサポートの制限が適用されます。テクノロジープレビュー機能は、Red Hat 製品のサービスレベルアグリーメント (SLA) の対象外であり、機能的に完全ではないことがあります。Red Hat は実稼働環境でこれらを使用することを推奨していません。これらの機能は、近々発表予定の製品機能をリリースに先駆けてご提供することにより、お客様は機能性をテストし、開発プロセス中にフィードバックをお寄せいただくことができます。

詳細は、[テクノロジープレビュー機能のサポート範囲](#) を参照してください。

8.2. METRO-DR [テクノロジープレビュー]

Metro-DR は、Red Hat Advanced Cluster Management for Kubernetes (RHACM)、Red Hat Ceph Storage、および OpenShift Data Foundation コンポーネントで設定され、OpenShift Container Platform クラスター全体でアプリケーションとデータのモビリティを提供します。

このリリースの Metro-DR ソリューションは、地理的に分散しているサイト間でボリュームの永続的なデータとメタデータのレプリケーションを提供します。パブリッククラウドでは、これらはアベイラビリティゾーンの障害からの保護に似ています。Metro-DR は、データセンターが利用できない場合でも、データを失うことなくビジネスの継続性を保証します。このソリューションには、Red Hat Advanced Cluster Management (RHACM) と OpenShift Data Foundation Advanced SKU および関連するバンドルが含まれています。

前提条件

Red Hat OpenShift Data Foundation でサポートされる障害復旧機能では、障害復旧ソリューションを正常に実装するために以下の前提条件をすべて満たす必要があります。

- 有効な Red Hat OpenShift Data Foundation Advanced エンタイトルメント
- 有効な Red Hat Advanced Cluster Management for Kubernetes サブスクリプション

OpenShift Data Foundation のサブスクリプションがどのように機能するかを知るには、[OpenShift Data Foundation subscriptions に関するナレッジベースの記事](#) を参照してください。

詳細なソリューション要件は、[Metro-DR 要件](#)、[arbiter を備えた Red Hat Ceph Storage ストレッチクラスタのデプロイメント要件](#) および [RHACM 要件](#) を参照してください。



重要

Advanced Cluster Management を使用した Metro-DR 用の OpenShift Data Foundation の設定は、テクノロジープレビュー機能であり、テクノロジープレビューサポートの制限が適用されます。テクノロジープレビュー機能は、Red Hat 製品のサービスレベルアグリーメント (SLA) の対象外であり、機能的に完全ではないことがあります。Red Hat は実稼働環境でこれらを使用することを推奨していません。これらの機能は、近々発表予定の製品機能をリリースに先駆けてご提供することにより、お客様は機能性をテストし、開発プロセス中にフィードバックをお寄せいただくことができます。

詳細は、[テクノロジープレビュー機能のサポート範囲](#) を参照してください。

8.3. STRETCHED CLUSTER - ARBITER [テクノロジープレビュー]

この例では、3 番目のゾーンを Arbiter の場所とした上で、単一クラスターが2つのゾーンに展開され

ます。これはテクノロジープレビュー機能であり、現時点では OpenShift Container Platform オンプレミスおよび同じデータセンターでのデプロイメントを目的としています。このソリューションは、複数のデータセンターにまたがる展開にはお勧めできません。代わりに、Metro-DR を、低遅延ネットワークを備えた複数のデータセンターに展開されたデータ損失のない DR ソリューションの最初のオプションとして検討してください。



注記

このソリューションは、主要なオンプレミスデータセンターに存在する 2 つのゾーンのロケーション間のレイテンシーが 10 ミリ秒の往復時間 (RTT) を超えない場所に展開するように設計されています。より高いレイテンシーでデプロイする予定がある場合は、[Red Hat カスタマーサポート](#) にお問い合わせください。

Arbiter ストレッチクラスターを使用するには、以下を実行します。

- 3 つのゾーンには、最低でも 5 つのノードが必要です。ここでは、以下のようになります。
 - データセンターゾーンごとに 2 つのノードが使用され、arbiter ゾーンにはノードが 1 つ含まれる、追加ゾーンが 1 つ使用されます (arbiter はマスターノード上にある場合があります)。
- すべてのノードには、クラスターの作成前にゾーンのラベルを手動で付ける必要があります。たとえば、ゾーンには以下のようにラベル付けできます。
 - `topology.kubernetes.io/zone=arbiter` (マスターまたはワーカーノード)
 - `topology.kubernetes.io/zone=datacenter1` (2 つ以上のワーカーノード)
 - `topology.kubernetes.io/zone=datacenter2` (2 つ以上のワーカーノード)

詳細は、[Configuring OpenShift Data Foundation for stretch cluster](#) で [Knowledgebase article](#) を参照してください。

OpenShift Data Foundation のサブスクリプションがどのように機能するかを知るには、[OpenShift Data Foundation subscriptions に関するナレッジベースの記事](#) を参照してください。



重要

ストレッチクラスターはテクノロジープレビュー機能で、テクノロジープレビューのサポート制限の対象となります。テクノロジープレビュー機能は、Red Hat 製品のサービスレベルアグリーメント (SLA) の対象外であり、機能的に完全ではないことがあります。Red Hat は実稼働環境でこれらを使用することを推奨していません。これらの機能は、近々発表予定の製品機能をリリースに先駆けてご提供することにより、お客様は機能性をテストし、開発プロセス中にフィードバックをお寄せいただくことができます。

詳細は、[テクノロジープレビュー機能のサポート範囲](#) を参照してください。

第9章 非接続環境

非接続環境は、Operator Lifecycle Manager (OLM) がインターネット接続が必要なデフォルトの Operator Hub およびイメージレジストリーにアクセスできないネットワークが制限された環境です。

Red Hat は、OpenShift Container Platform がネットワークが制限された環境にインストールされた非接続環境での OpenShift Data Foundation のデプロイメントをサポートします。

切断された環境に OpenShift Data Foundation をインストールするには、OpenShift Container Platform のドキュメントの [Operators ガイド](#) の [制限付きネットワークでの Operator Lifecycle Manager の使用](#) を参照してください。



注記

OpenShift Data Foundation をネットワークが制限された環境でインストールする場合は、デフォルトでインターネット接続が OpenShift Container Platform で想定され、**chrony** が ***.rhel.pool.ntp.org** サーバーを使用するように設定されるため、カスタム Network Time Protocol (NTP) 設定をノードに適用します。

詳細は、Red Hat ナレッジベースソリューションの [新しくデプロイされた OCS4 クラスターのステータスが Degraded と表示されるのはなぜですか。](#)、および、OpenShift Container Platform ドキュメントの [インストールガイド](#) の [chronytime サービスの設定](#) を参照してください。

OpenShift Data Foundation に含まれるパッケージ

redhat-operator インデックスイメージをプルリングするときは、{project-name-short} デプロイメントのパッケージの次のリストを含めます。

- **ocs-operator**
- **odf-operator**
- **mcg-operator**
- **odf-csi-addons-operator**
- **odf-lvm-operator**
これは現在、テクノロジープレビュー機能であり、切断されたインストールではサポートされていません。
- オプション: **local-storage-operator**
ローカルストレージ展開の場合のみ。
- オプション: **odf-multicluster-orchestrator**
リージョナル障害復旧 (Regional-DR) 設定の場合のみ。



重要

CatalogSource に **redhat-operators** という名前を付けます。

第10章 IBM POWER および IBM Z インフラストラクチャーでサポートされている機能およびサポートされていない機能

表10.1 IBM Power および IBM Z インフラストラクチャーでサポートされている機能およびサポートされていない機能の一覧

機能	IBM Power	IBM Z インフラストラクチャー
コンパクトなデプロイメント	サポート対象外	サポート対象外
動的ストレージデバイス	サポート対象外	サポート対象
ストレッチクラスター - Arbiter	サポート対象外	サポート対象外
Federal Information Processing Standard Publication (FIPS)	サポート対象	サポート対象外
プール圧縮メトリクスを表示する機能	サポート対象	サポート対象外
Multicloud Object Gateway (MCG) エンドポイント Pod の自動スケーリング	サポート対象	サポート対象外
オーバープロビジョニングを制御するアラート	サポート対象	サポート対象外
Ceph Monitor がスペースを使い果たしたときにアラート	サポート対象	サポート対象外
スタンドアロンの Multicloud Object Gateway コンポーネントのデプロイメント	サポート対象	サポート対象外
IBM Flashsystem などのプラグ可能な外部ストレージを可能にする拡張 OpenShift Data Foundation コントロールプレーン	サポート対象外	サポート対象外
IPV6 サポート	サポート対象外	サポート対象外
Multus	サポート対象外	サポート対象外
Multicloud Object Gateway (MCG) バケットのレプリケーション	サポート対象	サポート対象外
オブジェクトデータのクォータサポート	サポート対象	サポート対象外

機能	IBM Power	IBM Z インフラストラクチャー
最小限のデプロイメント	サポート対象外	サポート対象外
Red Hat Advanced Cluster Management (RHACM) を使用した Regional-Disaster Recovery (Regional-DR)	サポート対象外	サポート対象外
RHACM を使用した Metro-DisasterRecovery (Metro-DR) 複数クラスター	サポート対象外	サポート対象外
無線アクセスネットワーク (RAN) のシングルノードソリューション	サポート対象外	サポート対象外
ネットワークファイルシステム (NFS) サービスのサポート	サポート対象	サポート対象外
Multicloud Object Gateway (MCG) アカウントの認証情報を変更する機能	サポート対象	サポート対象外
Red Hat Advanced Cluster Management コンソールでのマルチクラスター監視	サポート対象外	サポート対象外
Multicloud Object Gateway ライフサイクルでの期限切れオブジェクトの削除	サポート対象	サポート対象外

第11章 次のステップ

OpenShift Data Foundation のデプロイを開始するには、OpenShift Container Platform 内で内部モードを使用するか、外部モードを使用して OpenShift Container Platform の外部で実行されているクラスターからサービスを使用できるようにします。

要件に応じて、それぞれのデプロイメントガイドを参照します。

内部モード

- [Amazon Web サービスを使用した OpenShift Data Foundation のデプロイ](#)
- [ベアメタルを使用した OpenShift Data Foundation のデプロイ](#)
- [VMWare vSphere を使用した OpenShift Data Foundation のデプロイ](#)
- [Microsoft Azure を使用した OpenShift Data Foundation のデプロイ](#)
- [Google Cloud を使用した OpenShift Data Foundation のデプロイ](#) [テクノロジープレビュー]
- [Red Hat OpenStack Platform を使用した OpenShift Data Foundation のデプロイ](#) [テクノロジープレビュー]
- [Red Hat Virtualization プラットフォームを使用した OpenShift Data Foundation のデプロイ](#)
- [IBM Power での OpenShift Data Foundation のデプロイ](#)
- [IBM Z インフラストラクチャーへの OpenShift Data Foundation のデプロイ](#)

外部モード

- [外部モードでの OpenShift Data Foundation のデプロイ](#)