



Red Hat OpenShift Container Storage 4.8

Red Hat OpenShift Container Storage アーキテ クチャー

Red Hat OpenShift Container Storage とそのコンポーネントおよびワークフローの
アーキテクチャーの概要。

Red Hat OpenShift Container Storage 4.8 Red Hat OpenShift Container Storage アーキテクチャー

Red Hat OpenShift Container Storage とそのコンポーネントおよびワークフローのアーキテクチャーの概要。

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/Red_Hat_OpenShift_Container_Storage_architecture.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[®] 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

このドキュメントでは、OpenShift Container Storage アーキテクチャーの概要を説明します。

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 CONTAINER STORAGE の紹介 | 5 |
| 第2章 OPENSIFT CONTAINER STORAGE アーキテクチャーの概要 | 6 |
| 第3章 OPENSIFT CONTAINER STORAGE OPERATOR | 7 |
| 3.1. OPENSIFT CONTAINER STORAGE OPERATOR | 7 |
| 3.1.1. コンポーネント | 7 |
| 3.1.2. 設計ダイアグラム | 7 |
| 3.1.3. 責任 | 8 |
| 3.1.4. Resources | 8 |
| 3.1.5. 制限 | 9 |
| 3.1.6. 高可用性 | 9 |
| 3.1.7. 関連する設定ファイル | 9 |
| 3.1.8. 関連するログファイル | 9 |
| 3.1.9. ライフサイクル | 10 |
| 3.2. ROOK-CEPH OPERATOR | 10 |
| 3.2.1. コンポーネント | 10 |
| 3.2.2. 設計ダイアグラム | 11 |
| 3.2.3. 責任 | 11 |
| 3.2.4. Resources | 12 |
| 3.2.5. ライフサイクル | 13 |
| 3.3. NOOBAA OPERATOR | 13 |
| 3.3.1. コンポーネント | 14 |
| 3.3.2. 責任およびリソース | 14 |
| 3.3.3. 高可用性 | 16 |
| 3.3.4. 関連するログファイル | 16 |
| 3.3.5. ライフサイクル | 16 |

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 である Chris Wright のメッセージ](#) を参照してください。

RED HAT ドキュメントへのフィードバック (英語のみ)

弊社のドキュメントについての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 (コンポーネント) として **Documentation** を使用します。
 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 CONTAINER STORAGE の紹介

Red Hat OpenShift Container Storage 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および管理を容易に実行するために Operator としてパッケージ化され、Red Hat OpenShift Container Platform サービスカタログの一部として利用できます。

Red Hat OpenShift Container Storage 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。

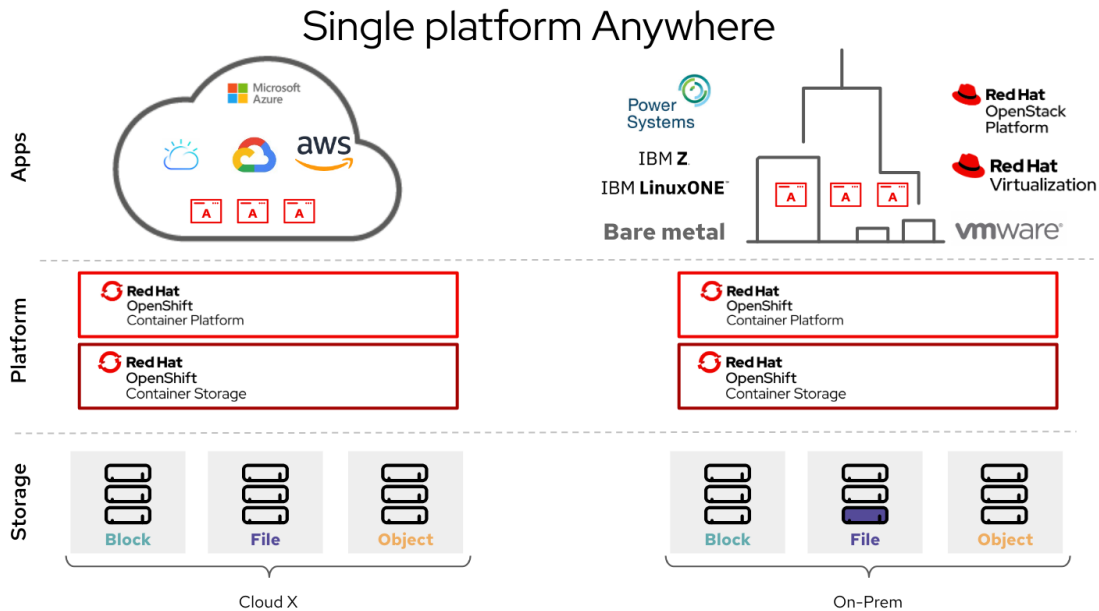
Red Hat OpenShift Container Storage バージョン 4.x は、以下を含むソフトウェアプロジェクトのコレクションを統合します。

- 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 Container Storage、Rook-Ceph、および NooBaa Operator。OpenShift Container Storage サービスを初期化し、管理します。

第2章 OPENSIFT CONTAINER STORAGE アーキテクチャーの概要

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

Red Hat OpenShift Container Storage アーキテクチャー



Red Hat OpenShift Container Storage 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 クラスター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](#) について参照してください。Red Hat OpenShift Container Storage と Red Hat OpenShift Container Platform のコンポーネントの相互運用性についての詳細は、[相互運用性マトリックス](#) 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。

第3章 OPENSIFT CONTAINER STORAGE OPERATOR

Red Hat OpenShift Container Storage は、タスクとリソースの特性を簡単に自動化できるように、管理タスクとカスタムリソースをコード化する次の3つのオペレーターで設定されます。

- **ocs-operator**
- **Rook-Ceph**
- **NooBaa**

管理者はクラスターの必要な最終状態を定義し、OpenShift Container Storage 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、またはその状態に近づけるようにします。

3つの Operator のすべてのデプロイメントは、Operator Lifecycle Manager (OLM) によって管理される **ocs-operator** バンドルで定義されます。

3.1. OPENSIFT CONTAINER STORAGE OPERATOR

ocs-operator は、OpenShift Container Storage の3つの主要なオペレーターの1つであり、「メタ」オペレーターとして説明できます。これは、他の Operator に影響を与え、他の Operator が提供する機能の設定ゲートウェイとして機能する Operator を指します。他の演算子を直接管理しません。

ocs-operator には、以下の主要機能があります。

- 他の Operator をトリガーしてそれらに対して調整するカスタムリソース (CR) を作成します。
- Ceph および Multicloud Object Gateway 設定を抽象化し、それらを Red Hat が検証し、サポートする既知のベストプラクティスに制限します。
- サポートポリシーに従って、コンテナ化された Ceph および NooBaa をデプロイするのに必要なリソースを作成し、調整します。

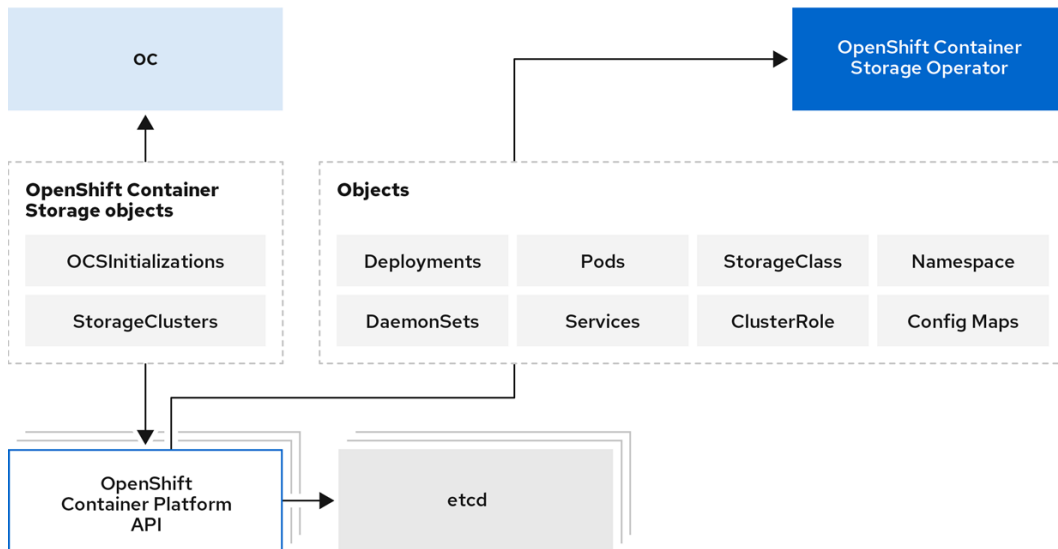
3.1.1. コンポーネント

ocs-operator には依存コンポーネントがありません。ただし、Operator には、**ClusterServiceVersion** (CSV) で定義される他の Operator からすべてのカスタムリソース定義 (CRD) の存在に依存しています。

3.1.2. 設計ダイアグラム

この図は、OpenShift Container Storage が OpenShift Container Platform とどのように統合されているかを示しています。

OpenShift Container Storage Operator



171_OpenShift_0721

3.1.3. 責任

2つの **ocs-operator** CRD は次のとおりです。

- **OCSInitialization**
- **StorageCluster**

OCSInitialization は、Operator レベルで適用されるカプセル化操作に使用されるシングルトン CRD です。オペレーターは、1つのインスタンスが常に存在することを確認します。CR は以下をトリガーします。

- OpenShift Container Storage に必要な初期化タスクを実行します。必要な場合は、**OCSInitialization** CRD を削除して、これらのタスクを再度実行するようにトリガーすることができます。
 - OpenShift Container Storage に必要なセキュリティーコンテキスト制約 (SCC) が存在することを確認します。
- 高度なトラブルシューティングおよび復旧操作を実行するために使用される Ceph ツールボックス Pod のデプロイメントを管理します。

StorageCluster CRD は、OpenShift Container Storage の全機能を提供するシステムを表します。これは Operator をトリガーし、CRD **Rook-Ceph** および **NooBaa** の生成および調整を行うことができます。**ocs-operator** アルゴリズムは、**StorageCluster** 仕様の設定に基づいて CRD **CephCluster** および **NooBaa** を生成します。Operator は **CephBlockPools**、**Routes** などの追加の CR も作成します。これらのリソースは、OpenShift Container Storage のさまざまな機能を有効にするのに必要です。現時点で、OpenShift Container Platform クラスターごとに1つの **StorageCluster** CR のみがサポートされます。

3.1.4. Resources

ocs-operator は、定義する CRD の仕様に応じて、次の CR を作成します。これらのリソースの一部の設定は上書きでき、生成された仕様の変更を許可したり、新規に作成しなくて済むようにできます。

一般的なリソース

イベント

調整に対応するために必要に応じてさまざまなイベントを作成します。

永続ボリューム (PV)

PV はオペレーターが直接作成するものではありません。ただし、Operator は Ceph CSI ドライバーで作成されたすべての PV を追跡し、サポートされる機能に適切なアノテーションが PV にあることを確認します。

クイックスタート

OpenShift Container Platform コンソールのさまざまなクイックスタート CR をデプロイします。

Rook-Ceph リソース

CephBlockPool

デフォルトの Ceph ブロックプールを定義します。Ceph オブジェクトストアの **CephFilesystemPrometheusRulesoute**。

StorageClass

デフォルトのストレージクラスを定義します。たとえば、**CephBlockPool** および **CephFilesystem** 用です。

VolumeSnapshotClass

対応するストレージクラスのデフォルトボリュームスナップショットクラスを定義します。

Multicloud Object Gateway リソース

NooBaa

デフォルトの Multicloud Object Gateway システムを定義します。

リソースの監視

- Metrics Exporter Service
- Metrics Exporter Service Monitor
- PrometheusRules

3.1.5. 制限

ocs-operator は、OpenShift Container Storage の他の Pod をデプロイも調整もしません。**ocs-operator** CSV は、Operator Deployment および Operator Lifecycle Manager (OLM) などの最上位のコンポーネントを定義します。

3.1.6. 高可用性

高可用性は、他のほとんどの Operator と同様、**ocs-operator** Pod の主な要件ではありません。通常、プロセスのディストリビューションに必要な操作や利点はありません。OpenShift Container Platform は、現在の Pod が利用できなくなるか、または削除されるたびに、置換 Pod を迅速に立ち上げます。

3.1.7. 関連する設定ファイル

ocs-operator 設定は CSV によって完全に指定され、CSV のカスタムビルドなければ変更することができません。

3.1.8. 関連するログファイル

OpenShift Container Storage を理解し、問題のトラブルシューティングを行うには、以下を参照してください。

- Operator Pod ログ
- StorageCluster のステータスおよびイベント
- OCSInitialization ステータス

Operator Pod ログ

各 Operator は、調整および発生したエラーについての情報が含まれる標準の Pod ログを提供します。これらのログには多くの場合、正常な調整に関する情報があり、これを除外して無視することができません。

StorageCluster のステータスおよびイベント

StorageCluster CR は、CR のステータスに調整の詳細を保存し、関連するイベントを持ちます。ステータスには、予想されるコンテナイメージのセクションが含まれます。これは、他の Operator からの Pod に存在する必要があるコンテナイメージと、現在検出するイメージを示します。これは、OpenShift Container Storage のアップグレードが完了したかどうかを判断するのに役立ちます。

OCSInitialization ステータス

このステータスは、初期化タスクが正常に完了したかどうかを示します。

3.1.9. ライフサイクル

OpenShift Container Storage がインストールされている限り、**ocs-operator** が存在する必要があります。これは、OLM による OpenShift Container Storage CSV の調整の一部として管理されます。Pod の少なくとも 1 つのインスタンスが **Ready** 状態にある必要があります。

CRD などの Operator オペランドは Operator のライフサイクルには影響しません。**OCSInitialization** CR は常に存在しているはずで、Operator が存在しない場合はこれを作成します。StorageClusters の作成および削除は Operator の制御外の操作であり、管理者によって開始するか、または適切な API 呼び出しによって自動化される必要があります。

3.2. ROOK-CEPH OPERATOR

Rook-Ceph オペレーターは、OpenShift Container Storage の Ceph の Rook オペレーターです。Rook を使用すると、Ceph ストレージシステムを OpenShift Container Platform で実行できます。

Rook-Ceph Operator は、ストレージクラスターを自動的にブートストラップし、ストレージデーモンを監視してストレージクラスターが正常であることを確認する単純なコンテナです。

3.2.1. コンポーネント

Rook-Ceph オペレーターは、OpenShift Container Storage デプロイメントの一部として多数のコンポーネントを管理します。

Ceph-CSI ドライバー

この Operator は、CSI ドライバー (RADOS ブロックデバイス (RBD) および Ceph ファイルシステム (CephFS) のそれぞれに対するプロビジョナーを含む)、ならびにその各ドライバーにボリュームプラグイン **daemonset** を作成し、更新します。

Ceph デーモン

Mons

モニター (mons) は Ceph のコアメタデータストアを提供します。

OSD

オブジェクトストレージデーモン (OSD) は、データを基礎となるデバイスに保存します。

Mgr

マネージャー (mgr) はメトリクスを収集し、Ceph の他の内部機能を提供します。

RGW

RADOS Gateway(RGW) は、オブジェクトストアに S3 エンドポイントを提供します。

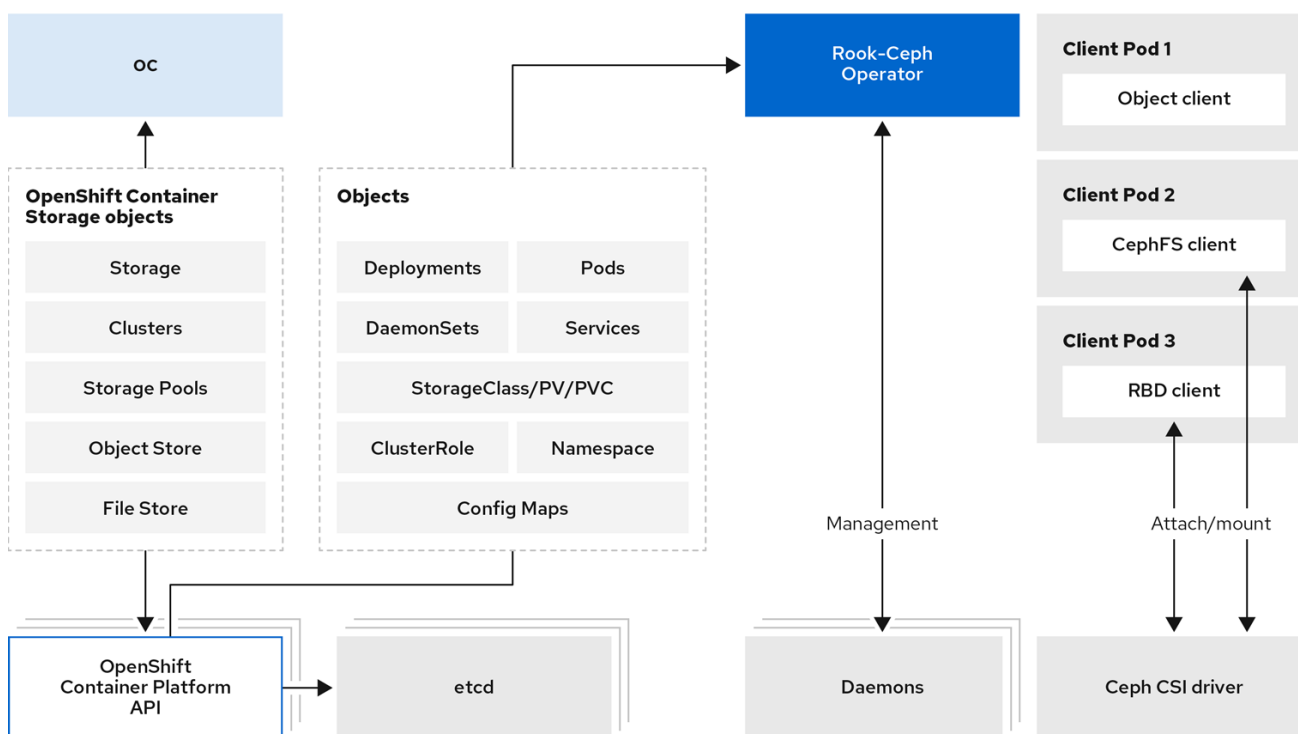
MDS

メタデータサーバー (MDS) は CephFS 共有ボリュームを提供します。

3.2.2. 設計ダイアグラム

以下の図は、Ceph Rook を OpenShift Container Platform と統合する方法を示しています。

Rook-Ceph Operator



171_OpenShift_0721

OpenShift Container Platform クラスタで Ceph が実行されている場合、OpenShift Container Platform アプリケーションは、Rook-Ceph によって管理されるブロックデバイスおよびファイルシステムをマウントしたり、オブジェクトストレージに S3/Swift API を使用できます。

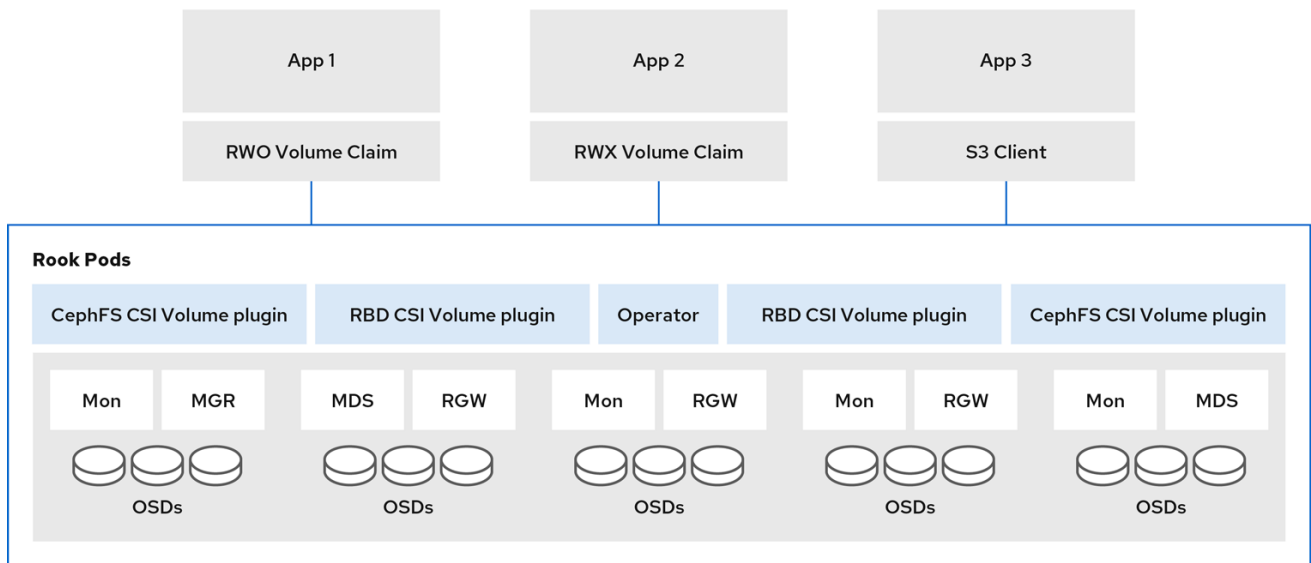
3.2.3. 責任

Rook-Ceph Operator は、ストレージクラスターをブートストラップし、監視するコンテナです。以下の機能を実行します。

- ストレージコンポーネントの設定を自動化する

- Ceph モニター Pod および Ceph OSD デーモンの開始、監視、および管理を行い、RADOS ストレージクラスターを提供する
- Pod およびその他のアーティファクトを初期化し、以下を管理するサービスを実行する
 - プールの CRD
 - オブジェクトストア (S3/Swift)
 - ファイルシステム
- Ceph mons および OSD を監視して、ストレージが利用可能で、正常な状態にあることを確認する
- クラスターサイズに基づいて mon 設定を調整しつつ、Ceph mons 配置をデプロイし、管理する
- API サービスによって要求される必要な状態変更を監視し、変更を適用する
- ストレージの使用に必要な Ceph-CSI ドライバーを初期化する
- ストレージを Pod にマウントするように Ceph-CSI ドライバーを自動的に設定する

Rook-Ceph Operator アーキテクチャー



171_OpenShift_0721

Rook-Ceph Operator イメージには、クラスターを管理するのに必要なすべてのツールが含まれます。データパスへの変更はありません。ただし、Operator はすべての Ceph 設定を公開しません。配置グループやクラッシュマップなどの Ceph 機能の多くはユーザーに表示されなくなり、物理リソース、プール、ボリューム、ファイルシステム、およびバケットにおけるユーザーエクスペリエンスが改善されました。

3.2.4. Resources

Rook-Ceph Operator は、**openshift-storage** namespace に作成するすべてのリソースに所有者参照を追加します。クラスターがアンインストールされると、所有者参照により、リソースがすべてクリーンアップされるようになります。これには、**configmaps**、**secrets**、**services**、**deployments**、**daemonsets** などの OpenShift Container Platform リソースが含まれます。

Rook-Ceph Operator は CR を監視し、**CephCluster**、**CephObjectStore**、**CephFilesystem**、**CephBlockPool** などの OpenShift Container Storage によって決定されます。

3.2.5. ライフサイクル

Rook-Ceph Operator は、Ceph クラスタで以下の Pod のライフサイクルを管理します。

Rook Operator

クラスタの調整を所有する単一 Pod。

RBD CSI ドライバー

- 単一デプロイメントで管理される 2 つのプロビジョナー Pod。
- **daemonset** によって管理されるノードごとに 1 つのプラグイン Pod。

CephFS CSI Driver

- 単一デプロイメントで管理される 2 つのプロビジョナー Pod。
- **daemonset** によって管理されるノードごとに 1 つのプラグイン Pod。

モニター (mons)

3 つの mon Pod (それぞれに独自のデプロイメント)。

ストレッチクラスタ

5 つの mon Pod (Arbiter ゾーンに 1 つとその他の 2 つのデータゾーンにそれぞれ 2 つずつ) が含まれます。

マネージャー (mgr)

クラスタに単一の mgr Pod があります。

ストレッチクラスタ

2 つの非調停ゾーンのそれぞれに 1 つずつ、2 つの mgr Pod (OpenShift Container Storage 4.8 以降) があります。

オブジェクトストレージデーモン (OSD)

3 つ以上の OSD がクラスタに最初に作成されます。クラスタが拡張すると、さらに OSD が追加されます。

メタデータサーバー (MDS)

CephFS メタデータサーバーには単一の Pod があります。

RADOS ゲートウェイ (RGW)

Ceph RGW デーモンには単一の Pod があります。

3.3. NOOBAA OPERATOR

NooBaa (Multicloud Object Gateway と呼ばれます) オペレーターは、OpenShift Container Storage オペレーターおよび Rook-Ceph オペレーターと共に、OpenShift Container Storage のオペレーターです。NooBaa Operator はアップストリームをスタンドアロン Operator として利用できます。

NooBaa Operator は以下のプライマリー機能を実行します。

- OpenShift Container Storage 内の Multicloud Object Gateway (MCG) コンポーネントを制御および調整します。
- Object Bucket Claim(オブジェクトバケット要求、バケットクラス、バックングストアなどの新規ユーザーリソース)を管理します。
- デフォルトですぐに使用可能なリソースを作成します。

いくつかの設定と情報は、OpenShift Container Storage オペレーターを介して NooBaa オペレーターに渡されます。

3.3.1. コンポーネント

NooBaa Operator にはサブコンポーネントがありません。ただし、これは、それによって制御されるさまざまなリソースの調整ループで設定されます。

NooBaa オペレーターにはコマンドラインインターフェイス (CLI) があり、OpenShift Container Storage の一部として利用できます。これにより、各種リソースの作成、削除、およびクエリーが可能になります。この CLI は、YAML ファイルを直接適用するのではなく、設定が適用される前に、入力サニタイゼーションおよびステータス検証のレイヤーを追加します。

3.3.2. 責任およびリソース

NooBaa Operator はカスタムリソース定義 (CRD) および OpenShift Container Platform エンティティについて調整されます。

- バックングストア
- namespace ストア
- Bucket クラス
- Object Bucket Claim(オブジェクトバケット要求)
- NooBaa、Pod ステートフルセット CRD
- Prometheus ルールおよびサービスモニタリング
- Horizontal pod autoscaler (HPA)

バックングストア

お客様が MCG コンポーネントに接続されているリソース。このリソースは、プロビジョニングされたバケットのデータを MCG に保存する機能を提供します。

デフォルトのバックングストアは、OpenShift Container Platform が実行しているプラットフォームに応じてデプロイメントの一部として作成されます。たとえば、OpenShift Container Platform または OpenShift Container Storage が Amazon Web Services (AWS) にデプロイされる場合は、AWS::S3 バケットであるデフォルトのバックングストアになります。同様に、Microsoft Azure の場合、デフォルトのバックングストアは Blob コンテナなどです。

デフォルトのバックングストアは、OpenShift Container Platform に含まれるクラウド認証情報 Operator の CRD を使用して作成されます。MCG に追加できるバックングストアの量に制限はありません。バックングストアは、バケットの異なるポリシーを定義するためにバケットクラス CRD で使用されます。バックングストアとしてサポートされるサービスまたはリソースのタイプを特定するには、特定の OpenShift Container Storage バージョンのドキュメントを参照してください。

namespace ストア

namespace バケットで使用されるリソース。デプロイメント時に、デフォルトは作成されません。

Bucketclass

新しくプロビジョニングされたバケットのデフォルトまたは初期ポリシー。以下のポリシーは、バケットクラスに設定されます。

配置ポリシー

バケットに割り当てられるバックングストアを示し、バケットのデータの書き込みに使用します。このポリシーはデータバケットに使用され、キャッシュポリシーでローカルキャッシュの配置を指定します。配置ポリシーのモードは2つあります。

- Spread。定義されたバックングストア全体でデータを削除します。
- Mirror。各バックングストアで完全なレプリカを作成します。

Namespace ポリシー

集約に使用されるリソースと、書き込みターゲットに使用されるリソースを定義する namespace バケットのポリシー。

キャッシュポリシー

これはバケットのポリシーで、ハブ (信頼できるソース) と、キャッシュアイテムの存続時間 (TTL) を設定します。

デフォルトのバケットクラスはデプロイメント中に作成され、デフォルトのバックングストアを使用する配置ポリシーで設定されます。追加できるバケットクラスの数に制限はありません。

サポートされているポリシーのタイプを特定するには、特定の OpenShift Container Storage バージョンのドキュメントを参照してください。

Object Bucket Claim(オブジェクトバケット要求)

S3 バケットのプロビジョニングを有効にする CRD。MCG では、OBC は任意のバケットクラスを受け取り、バケットの初期設定を書き留めておきます。バケットクラスが指定されていない場合は、デフォルトのバケットクラスが使用されます。

NooBaa、Pod ステートフルセット CRD

DB Pod、コア Pod、エンドポイントなどの NooBaa デプロイメントの異なる Pod を制御する内部 CRD。この CRD は内部であるため、変更しないでください。この Operator は以下のエンティティを調整します。

- DB Pod SCC
- OpenShift Container Platform と NooBaa ユーザーインターフェイスとの間の SSO シングルサインオンを許可するロールバインディングおよびサービスアカウント
- S3 アクセスのルート
- OpenShift Container Platform によって取得および署名され、S3 ルートに設定されている証明書

Prometheus ルールおよびサービスのモニタリング

これらの CRD は、MCG でサポートされている Prometheus およびアラートルールのスクレイピングポイントを設定します。

Horizontal pod autoscaler (HPA)

これは MCG エンドポイントと統合されます。エンドポイント Pod は、CPU の負荷 (S3 トラフィックの量) に応じてスケールアップおよびスケールダウンします。

3.3.3. 高可用性

Operator として提供される唯一の高可用性は、OpenShift Container Platform が障害のある Pod を再スケジューリングすることです。

3.3.4. 関連するログファイル

NooBaa Operator の問題のトラブルシューティングを行うには、以下を確認します。

- Operator Pod ログ (must-gather でも利用できます)
- must-gather で利用できる各種 CRD またはエンティティ、およびそのステータス。

3.3.5. ライフサイクル

NooBaa オペレーターは、OpenShift Container Storage がデプロイされた後、アンインストールされるまで実行および調整します。