Red Hat Integration 2021.Q1

Red Hat Integration 2021.Q1 リリースノート

Red Hat Integration の新機能
Red Hat Integration の新機能

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 © 2021 | You need to change the HOLDER entity in the en-US/Release_Notes_for_Red_Hat_Integration_2021.Q1.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

Red Hat Integration プラットフォームについて説明し、本リリースの新機能について最新情報を提供します。
目次

第1章 RED HAT INTEGRATION ........................................................................... 3

第2章 本リリースの新機能 ................................................................................. 4
  2.1. インテグレーションの新機能 ................................................................ 4
  2.2. コンポーネントの新機能 ................................................................ 4

第3章 DEBEZIUM リリースノート .................................................................. 5
  3.1. DEBEZIUM データベースコネクター .............................................. 5
  3.2. DEBEZIUM でサポートされる構成 .................................................. 6
  3.3. DEBEZIUM のインストールオプション ............................................. 6
  3.4. DEBEZIUM の新機能 ................................................................ 6

第4章 CAMEL K リリースノート ..................................................................... 8
  4.1. CAMEL K の新機能 ................................................................ 8
    4.1.1. プラットフォームおよびコアコンポーネントのバージョン .......... 8
    4.1.2. テクノロジーレビューの機能 .................................................. 8
    4.1.3. Camel K Operator のメタデータ ............................................. 9
    4.2. CAMEL K の既知の問題 ............................................................. 9
    4.3. CAMEL K の修正された問題 .................................................... 10
      4.3.1. CVE セキュリティー問題の修正 ...................................... 10

第5章 CAMEL KAFKA CONNECTOR リリースノート ...................................... 11
  5.1. CAMEL KAFKA CONNECTOR の新機能 ....................................... 11
    5.1.1. プラットフォームおよびコアコンポーネントのバージョン ......... 11
    5.1.2. テクノロジーレビューの機能 .................................................. 11
    5.1.3. テクノロジーレビューの Camel Kafka コネクター .................. 12
  5.2. CAMEL KAFKA CONNECTOR の修正された問題 ......................... 13
    5.2.1. CVE セキュリティー問題の修正 ...................................... 13

第6章 RED HAT INTEGRATION の OPERATOR ............................................ 14
  6.1. OPERATOR とは ................................................................ 14
  6.2. RED HAT INTEGRATION OPERATOR ...................................... 14
    6.2.1. サポートライフサイクル ..................................................... 15
  6.3. RED HAT INTEGRATION コンポーネント OPERATOR .............. 15
    6.3.1. 3scale Operators .................................................................. 15
    6.3.2. AMQ Operator ..................................................................... 15
    6.3.3. Camel K Operator ................................................................. 15
    6.3.4. Fuse Operator ..................................................................... 15
    6.3.5. Service Registry Operator .................................................... 15
第1章 RED HAT INTEGRATION

Red Hat Integration は、ハイブリッド環境およびマルチクラウド環境全体でコンテナベースの統合サービスを作成、拡張、デプロイするための包括的な統合およびイベント処理技術です。Red Hat Integration は、デジタル環境で必要となるアプリケーションとシステム間でデータを接続および共有するために組織が使用できる、アジャイルで API 中心の分散ソリューションを提供します。

Red Hat Integration には、以下の機能が含まれています。

- API の接続
- データの変換
- サービスの構成とオーケストレーション
- リアルタイムのメッセージング
- データセンター間のメッセージストリーミング
- API 管理

その他のリソース

- エンタープライズ統合について理解する
第2章 本リリースの新機能

ここでは、Red Hat Integration 2021.Q1の主な新機能の概要を説明し、異なるコンポーネントで利用可能な新機能の詳細へのリンクを提供します。

注記

本リリースノートには、Red Hat Integration 2021.Q1でのみ更新されたコンポーネントの詳細が含まれています。Service Registryなどの他のコンポーネントの最新バージョンに関する詳細は、『Red Hat Integration Release Notes for 2020-Q4』を参照してください。

2.1. インテグレーションの新機能

Red Hat Integration Operator

- Red Hat Integration Operator 1.0 で Red Hat Integration コンポーネントを管理する OpenShift Operator のインストールおよびアップグレード

データの統合

-Debezium 1.4 での変更データキャプチャーやリアルタイムイベント（新しい DB2 コネクターや Service Registry との統合を含む）

Serverless Camel K

- Red Hat Integration - Camel K (テクノロジープレビュー) による Camel 3.7 をベースとしたサーバーレスアーキテクチャーのクラウドネイティブインテグレーション

イベント駆動型の Camel Kafka コネクター

-Camel Kafka Connector (テクノロジープレビュー) により Camel コンポーネントを外部システムへの Kafka コネクターとして利用可能

2.2. コンポーネントの新機能

Red Hat Integration 2021.Q1 コンポーネントの新機能の詳細は、以下を参照してください。

- Red Hat 3scale API Management
  - オンプレミス型 Red Hat 3scale API Management 2.9 向けリリースノート
  - SaaS 版 Red Hat 3scale API Management リリースノート

- Red Hat AMQ 2021.Q1 の製品ドキュメント

- Red Hat Fuse 7.8 リリースノート
第3章 DEBEZIUM リリースノート

Red Hat Integration 2021.Q1 には、Debezium オープンソースプロジェクトをベースとした、Debezium on OpenShift の GA (一般提供) リリースが含まれています。Debezium は、データベースの操作を追跡し、データ変更イベントをストリーミングする、分散型変更データキャプチャープラットフォームです。Debezium は Apache Karaf に構築され、AMQ Streams とデプロイおよび統合されます。

Debeziumによって、データベーステーブルの行レベルの変更がキャプチャーされ、対応する変更イベントコードが AMQ Streams に渡されます。アプリケーションはこれらの変更イベントストリームを読み取り、発生順に変更イベントにアクセスできます。

リリースの詳細は以下を参照してください。

- 「Debezium データベースコネクター」
- 「Debezium でサポートされる構成」
- 「Debezium のインストールオプション」
- 「Debezium の新機能」

3.1. DEBEZIUM データベースコネクター

Debezium は、以下の共通データベースの Kafka Connect をベースとしたコネクターを提供します。

- Db2
- MongoDB
- MySQL
- PostgreSQL
- SQL Server

注記

- Db2 コネクターには、Linux 用の Db2 の標準部分として利用できる抽象構文表記 (ASN) ライブラリーを使用する必要があります。
  - ASN ライブラリーを使用するには、IBM InfoSphere Data Replication (IIDR) のライセンスが必要です。
  - ライブラリーを使用するために IIDR をインストールする必要はありません。
- 現在、MongoDB 4.2 で Debezium MongoDB コネクターのトランザクションメタデータ機能を使用することはできません。
- Debezium PostgreSQL コネクターでは、PostgreSQL バージョン 10 以降のデフォルトである pgoutput 論理デコーディング出力プラグインを使用する必要があります。

その他のリソース

- Getting Started with Debezium
3.2. DEBEZIUM でサポートされる構成
サポートされるデータベースバージョンなどの、Debezium でサポートされる構成の詳細は「Debezium 1.4 Supported Configurations」のページを参照してください。

3.3. DEBEZIUM のインストールオプション
AMQ Streams で Debezium を OpenShift または RHEL にインストールできます。
- 『Installing Debezium on OpenShift』(GA、一般提供)
- 『Installing Debezium on RHEL』(テクノロジープレビュー)

重要
テクノロジープレビューの機能は、Red Hat の本番環境のサービスレベルアグリーメント (SLA) ではサポートされず、機能的に完全ではないことがあります。Red Hat は、本番環境でのテクノロジープレビュー機能の実装は推奨しません。テクノロジープレビューの機能は、最新の技術をいち早く提供して、開発段階で機能のテストやフィードバックの収集を可能にするために提供されます。サポート範囲の詳細は、「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。

3.4. DEBEZIUM の新機能
Debezium 1.4 には、以下の更新が含まれています。

GA (一般提供) への格上げ
これまでのリリースでテクノロジープレビューとして提供されていた以下の機能が、一般提供 (GA) で利用できるようになりました。

Debezium Db2 コネクター
IBM Db2 データベース用のコネクター (LUW)

コンテンツベースルーター
イベントコンテンツに基づいてデータ変更イベントレコードをトピックに再ルーティングするための単一メッセージ変換 (SMT、Single Message Transformation)。

フィルター SMT
各変更イベントの式を評価し、評価結果に基づいてイベントを削除または出力します。

Avro シリアライゼーション
Avro を使用してメッセージキーおよび値をシリアライズするDebezium コネクターの設定をサポートします。

テクノロジープレビューの機能
CloudEvents コンバーターコンバーターコンバーター
CloudEvents 仕様に準拠する変更イベントレコードが出力されます。Avro エンコーディングタイプが CloudEvents エンベロープ構造でサポートされるようになりました。

送信トレイ (Outbox) イベントルーター
複数の (マイクロ) サービス間でデータを安全かつ確実に交換するための送信トレイ (Outbox) パターンをサポートする SMT。
Debeziumのドキュメント

- Kafka Connectの自動トピック作成を有効化およびカスタマイズする方法。
- Db2およびSQL Serverのキャプチャジョブ設定がサーバーの負荷とレイテンシーにどのように影響するか。
第4章 CAMEL K リリースノート

Red Hat Integration - Camel K は、Red Hat Integration 2021.Q1 ではテクノロジープレビューのコンポーネントとして使用できます。Camel K は、OpenShift のクラウドでネイティブで実行される Apache Camel K からビルトされる軽量のインテグレーションフレームワークです。Camel K は、サーバーレスおよびマイクロサービスアーキテクチャー向けに特別に設計されています。Camel K を使用すると、Camel Domain Specific Language (DSL) で書かれたインテグレーションコードを直接 OpenShift で即座に実行することができます。

Camel K を OpenShift Serverless および Knative とともに使用すると、コンテナーは必要な場合のみ自動的に作成され、負荷時の自動スケーリングやゼロへのスケーリングが行われます。これにより、サーバーレスのプロビジョニングとメンテナンスのオーバーヘッドがなくなり、アプリケーションの開発に集中することができます。

Camel K を OpenShift Serverless および Knative Eventing とともに使用すると、システムのコンポーネントがサーバーレスアプリケーションのイベント駆動型アーキテクチャーで通信する方法を管理できます。これにより、イベントプロダクターとコンシューマー間の関係が切り離されたパブリッシュ/サブスクリプションモデルまたはイベントストリーミングモデルを使用すると、柔軟性を提供し、効率化を実現できます。

重要な注意事項

テクノロジープレビューの機能は、Red Hat の本番環境のサービスレベルアグリーメント (SLA) ではサポートされず、機能的に完全ではないことがあります。Red Hat は、本番環境でのテクノロジープレビュー機能の実装は推奨しません。

テクノロジープレビューの機能は、最新の技術をいち早く提供して、開発段階で機能のテストやフィードバックの収集を可能にするために提供されます。サポート範囲の詳細は、「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。

4.1. CAMEL K の新機能

テクノロジープレビューの Camel K は、以下の主要機能でクラウドネイティブインテグレーションを提供します。

4.1.1. プラットフォームおよびコアコンポーネントのバージョン

- OpenShift Container Platform 4.6 または 4.7
- OpenShift Serverless 1.13
- Quarkus 1.7 Java runtime
- Apache Camel K 1.3
- Apache Camel 3.7
- Apache Camel Quarkus 1.5
- OpenJDK 11

4.1.2. テクノロジープレビューの機能

- 自動スケーリングおよびゼロへのスケーリングを行うための Knative Serving
イベント駆動型アーキテクチャーのための Knative Eventing
デフォルトで Quarkus ランタイムを使用するパフォーマンスの最適化
Java、XML、または YAML DSL です書かれた Camel インテグレーション
Visual Studio Code での開発ツール
OpenShift で Prometheus を使用したインテグレーションの監視
新しい Transformations や SaaS が含まれるクイックスタートチュートリアル
AWS、Jira、Salesforce など、外部システムへのソースコネクターのための Kamelet Catalog

その他のリソース
- Camel K インテグレーションの OpenShift へのデプロイ

4.1.3. Camel K Operator のメタデータ
テクノロジープレビューの Camel K には、OpenShift OperatorHub から Camel K をインストールするために使用される、更新された Operator メタデータが含まれています。この Operator メタデータには、OpenShift Container Platform 4.6 以降との使用を目的に設計されたリリースパッケージの Operator バンドル形式が含まれます。

その他のリソース
- OpenShift ドキュメントに記載されている Operator バンドル形式。

4.2. CAMEL K の既知の問題
テクノロジープレビューの Camel K に存在する既知の問題を以下に示します。

ENTESB-15306 - CRD conflicts between Camel K and Fuse Online
古いバージョンの Camel K が同じ OpenShift クラスターにインストールされたことがある場合、カス タムリソース定義との競合により、OperatorHub から Camel K のインストールに失敗します。たとえ ば、これには Fuse Online で以前使用できた古いバージョンの Camel K が含まれます。

この問題を避けるには、Camel K を別の OpenShift クラスターにインストールするか、Camel K をイ ンストールする前に以下のコマンドを入力します。

```
$ oc get crds -l app=camel-k -o json | oc delete -f -
```

ENTESB-15787 - kamel run command does not work for remote files on Windows
Windows では、リモート Camel インテグレーションファイルの実行を試みると kamel run コマンドで エラーが発生します。以下に例を示します。

```
> kamel run https://raw.githubusercontent.com/apache/camel-k/b29333f0a878d5d09fb39e565be8fe586d77dd95d0/e2e/common/files/yaml.yaml
panic: runtime error: invalid memory address or nil pointer dereference...
```

ENTESB-15858 - Added ability to package and run Camel integrations locally or as container images
ローカルでの Camel インテグレーションのパッケージ化および実行、またはコンテナイメージとしての Camel インテグレーションのパッケージ化および実行は、現在テクノロジープレビューの Camel K には含まれておらず、コミュニティーのみによるサポートの対象となります。

詳細は「Apache Camel K community」を参照してください。

ENTESB-15893 - Camel K catalog contains camel-quarkus-spark reference and cannot deploy integrations with Apache Spark

Camel K カタログには、Camel Quarkus エクステンションの BOM (Bill of Materials) には含まれなくなった camel-quarkus-spark コンポーネントが含まれています。Camel Quarkus の Spark コンポーネントを使用して Camel K インテグレーションをデプロイしようとすると、この依存関係がないため、インテグレーションをコンパイルできません。

詳細は「Spark component in Camel Quarkus」を参照してください。

ENTESB-15930 - Camel K dependency autoloading not working correctly with YAML format

YAML で書かれた Camel インテグレーションで -route 属性を使用すると、Camel K 依存関係の自動読み込みは正しく機能しません。しかし、代わりに -from 属性を使用すると、Camel K 依存関係の自動読み込みは正常に機能します。

詳細は「Apache Camel K community」を参照してください。

4.3. CAMEL K の修正された問題

テクノロジープレビューの Camel K では、以下の問題が修正されました。

4.3.1. CVE セキュリティー問題の修正

ENTESB-14997 - CVE-2020-25649 jackson-databind: FasterXML DOMDeserialization insecure entity expansion is vulnerable to XML external entity

FasterXML Jackson Databind で、エンティティ拡張のセキュリティーが適切に保護されていないという不具合が発見されました。この不具合により、XML 外部エンティティー (XXE) 攻撃に対して脆弱な状態でした。この脆弱性では、データの整合性が最も懸念されました。
第5章 CAMEL KAFKA CONNECTOR リリースノート

Camel Kafka Connector は、Red Hat Integration 2021.Q1 のテクノロジープレビュー コンポーネントとして使用できます。

注記：サポート対象外のテクノロジープレビュー リリースを超える Camel Kafka Connector リリースのプランはありません。

Camel Kafka Connector を使用すると、Kafka Connect で標準の Camel コンポーネントをコネクターとして設定できます。これにより、可能なインテグレーションの範囲が、Kafka Connect のみによってサポートされる外部システム以外にも広がります。

Camel Kafka Connector を使用すると、Kafka Connect フレームワークで Camel コンポーネントを直接設定するのがより簡単になります。Camel Kafka シンクまたはソースコネクターを接続元または接続先とすることで、Camel コンポーネントを利用して異なるシステムとの統合を行うことができます。コードを作成する必要はありません。また、適切なコネクター JAR を Kafka Connect イメージに含めることが可能であり、カスタムリソースを使用してコネクターオプションを設定できます。

Camel Kafka Connector は、Apache Camel オープンソースコミュニティのサブプロジェクトである Apache Camel Kafka Connector 上に構築されます。Camel Kafka Connector は、OpenShift Container Platform、AMQ Streams、および Kafka Connect と完全に統合されています。OpenShift 上のクラウドネイティブインテグレーションでは、Red Hat Integration - Camel K ディストリビューションにて Camel Kafka Connector を利用できます。

5.1. CAMEL KAFKA CONNECTOR の新機能

テクノロジープレビューの Camel Kafka Connector には、以下の主要機能が含まれています。

5.1.1. プラットフォームおよびコアコンポーネントのバージョン

- OpenShift Container Platform 4.6 または 4.7
- Red Hat Enterprise Linux 8.x
- AMQ Streams 1.6
- Apache Kafka Connect 2.6
- Apache Camel Kafka Connector 0.7.1
- Apache Camel 3.7
- OpenJDK 11

5.1.2. テクノロジープレビューの機能

重要

テクノロジープレビューの機能は、Red Hat の本番環境のサービスレベルアグリーメント (SLA) ではサポートされず、機能的に完全ではないことがあります。Red Hat は、本番環境でのテクノロジープレビュー機能の実装は推奨しません。

サポート範囲の詳細は、「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。

5.1. CAMEL KAFKA CONNECTOR の新機能

テクノロジープレビューの Camel Kafka Connector には、以下の主要機能が含まれています。

5.1.1. プラットフォームおよびコアコンポーネントのバージョン

- OpenShift Container Platform 4.6 または 4.7
- Red Hat Enterprise Linux 8.x
- AMQ Streams 1.6
- Apache Kafka Connect 2.6
- Apache Camel Kafka Connector 0.7.1
- Apache Camel 3.7
- OpenJDK 11

5.1.2. テクノロジープレビューの機能

重要

テクノロジープレビューの機能は、Red Hat の本番環境のサービスレベルアグリーメント (SLA) ではサポートされず、機能的に完全ではないことがあります。Red Hat は、本番環境でのテクノロジープレビュー機能の実装は推奨しません。

サポート範囲の詳細は、「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。
- 一部の Camel Kafka コネクター
- シンクおよびソースコネクターの Camel データフォーマットのマーシャリング/アンマーシャリング
- シンクコネクターの集約
- コネクターを拡張するための Maven archetype

5.1.3. テクノロジープレビューの Camel Kafka コネクター

表5.1 テクノロジープレビューの Camel Kafka コネクター

<table>
<thead>
<tr>
<th>コネクター</th>
<th>シンク/ソース</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amazon Web Services (AWS2) Kinesis</td>
<td>シンクおよびソース</td>
</tr>
<tr>
<td>Amazon Web Services (AWS2) S3</td>
<td>シンクおよびソース</td>
</tr>
<tr>
<td>Amazon Web Services (AWS2) SNS</td>
<td>シンクのみ</td>
</tr>
<tr>
<td>Amazon Web Services (AWS2) SQS</td>
<td>シンクおよびソース</td>
</tr>
<tr>
<td>Azure Storage Blob</td>
<td>シンクのみ</td>
</tr>
<tr>
<td>Azure Storage Queue</td>
<td>シンクのみ</td>
</tr>
<tr>
<td>Cassandra Query Language (CQL)</td>
<td>シンクおよびソース</td>
</tr>
<tr>
<td>Elasticsearch</td>
<td>シンクのみ</td>
</tr>
<tr>
<td>File</td>
<td>シンクのみ</td>
</tr>
<tr>
<td>Hadoop Distributed File System (HDFS)</td>
<td>シンクのみ</td>
</tr>
<tr>
<td>Hypertext Transfer Protocol (HTTP)</td>
<td>シンクのみ</td>
</tr>
<tr>
<td>Java Database Connectivity (JDBC)</td>
<td>シンクのみ</td>
</tr>
<tr>
<td>Java Message Service (JMS)</td>
<td>シンクおよびソース</td>
</tr>
<tr>
<td>MongoDB</td>
<td>シンクおよびソース</td>
</tr>
<tr>
<td>RabbitMQ</td>
<td>シンクおよびソース</td>
</tr>
<tr>
<td>SQL</td>
<td>シンクおよびソース</td>
</tr>
<tr>
<td>SSH</td>
<td>シンクおよびソース</td>
</tr>
</tbody>
</table>
その他のリソース

・『Camel Kafka Connector のスタートガイド』

5.2. CAMEL KAFKA CONNECTOR の修正された問題
テクノロジープレビューの Camel Kafka Connect では、以下の問題が修正されました。

5.2.1. CVE セキュリティー問題の修正

ENTESB-14997 - CVE-2020-25649 jackson-databind: FasterXML DOMDeserializer insecure entity expansion is vulnerable to XML external entity

FasterXML Jackson Databind で、エンティティー拡張のセキュリティーが適切に保護されていないという不具合が発見されました。この不具合により、XML 外部エンティティー (XXE) 攻撃に対して脆弱な状態でした。この脆弱性では、データの整合性が最も懸念されました。

<table>
<thead>
<tr>
<th>コネクター</th>
<th>シンク/ソース</th>
</tr>
</thead>
<tbody>
<tr>
<td>Syslog</td>
<td>シンクおよびソース</td>
</tr>
<tr>
<td>Timer</td>
<td>ソースのみ</td>
</tr>
</tbody>
</table>

5章 CAMEL KAFKA CONNECTOR リリースノート
第6章 RED HAT INTEGRATION の OPERATOR

Red Hat Integrationは、OpenShiftでRed Hat Integrationコンポーネントのデプロイメントを自動化するために、operatorを提供します。Red Hat Integration Operatorを使用して、複数のコンポーネントoperatorを管理できます。そのため、各コンポーネントoperatorを個別に管理できます。ここではoperatorを紹介し、operatorを使用してRed Hat Integrationコンポーネントをデプロイする方法の詳細へのリンクを記載します。

6.1. OPERATORとは

operatorは、Kubernetesアプリケーションをパッケージ化、デプロイ、および管理する方法です。operatorは運用上の知識を仕様し、これをコンシューマーと簡単に共有できるソフトウェアにエンコードして、一般的なタスクや複雑なタスクを自動化します。

OpenShift Container Platform 4.xでは、Operator Lifecycle Manager (OLM)を使用すると、ユーザーはすべてのOperatorとクラスター全体で実行される関連サービスをインストールおよび更新することができ、それらのOperatorと関連サービスのライフサイクルを管理できます。これは、Kubernetesのネイティブアプリケーション(operator)を効果的かつ自動化されたスケーラブルな方法で管理するために設計されたオープンソースツールキットであるOperator Frameworkの一部です。

OLMはOpenShift Container Platform 4.xでデフォルトで実行されます。これは、クラスター管理者がクラスターで実行しているOperatorをインストールおよびアップグレードし、アクセスを付与するのに役立ちます。OpenShift Container Platform Webコンソールは、クラスター管理者がOperatorをインストールし、クラスターで利用可能なOperatorのカタログを使用するために特定のプロジェクトにアクセスを付与するための管理画面を提供します。

OperatorHubは、OpenShiftクラスター管理者がOperatorを検出、インストール、およびアップグレードするために使用するグラフィカルインターフェースです。1回クリックするだけで、これらのOperatorをOpenHubからプルし、クラスターにインストールすることができ、OLMでOperatorを管理することで、開発、テスト、および実稼働環境のソフトウェアをエンジニアリングチームが独自に管理することが可能です。

その他のリソース

- Operatorに関する詳細は、OpenShiftのドキュメントを参照してください。

6.2. RED HAT INTEGRATION OPERATOR

Red Hat Integration Operatorを使用して、複数のRed Hat IntegrationコンポーネントOperatorをインストールおよびアップグレードできます。

- 3scale
- 3scale APIcast
- AMQ Broker
- AMQ Interconnect
- AMQ Streams
- API Designer
- Camel K
6.2.1. サポートライフサイクル
サポート対象の設定を維持するには、最新の Red Hat Integration Operator バージョンをデプロイする必要があります。Red Hat Integration Operator の各リリースバージョンは 3カ月間のみサポートされます。

その他のリソース

- 複数の Red Hat Integration コンポーネント Operator の管理に関する詳細は、「Installing the Red Hat Integration Operator on OpenShift」を参照してください。

6.3. RED HAT INTEGRATION コンポーネント OPERATOR
たとえば、3scale Operator や Camel K Operator などを使用して、各 Red Hat Integration コンポーネント Operator を個別にインストールおよびアップグレードできます。

6.3.1. 3scale Operators
- 3scale Operator
- 3scale APIcast Operator

6.3.2. AMQ Operator
- AMQ Broker Operator
- AMQ Interconnect Operator
- AMQ Streams Cluster Operator
- AMQ Online Operator

6.3.3. Camel K Operator
- Camel K Operator (テクノロジープレビュー)

6.3.4. Fuse Operator
- Fuse on OpenShift - Samples Operator
- Fuse on OpenShift - Fuse Console Operator
- Fuse on OpenShift - API Designer Operator
- Fuse Online Operator

6.3.5. Service Registry Operator
Service Registry Operator

その他のリソース

複数の Red Hat Integration コンポーネント Operator の管理に関する詳細は、「Installing the Red Hat Integration Operator on OpenShift」を参照してください。