



Red Hat Fuse 7.12

OpenShift Container Platform での Fuse Online のインストールと操作

Fuse Online のインストール、設定、アップグレード、および CI/CD パイプラインの
インテグレーションのインポートやエクスポート

Red Hat Fuse 7.12 OpenShift Container Platform での Fuse Online のインストールと操作

Fuse Online のインストール、設定、アップグレード、および CI/CD パイプラインのインテグレーションのインポートやエクスポート

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

オンサイト、クラウド、またはハイブリッドクラウドのお客様が管理する環境にて、OpenShift Container Platform で Fuse Online をインストール、管理、および操作する情報および手順。

目次

前書き	4
多様性を受け入れるオープンソースの強化	5
第1章 OCP 4.X での FUSE ONLINE のインストール	6
1.1. OCP 4.X に FUSE ONLINE をインストールするのに必要な手順の概要	6
1.2. FUSE ONLINE のインストール前にデフォルトのカスタムリソースファイルを編集する必要がある場合	8
1.3. FUSE ONLINE の設定に使用するカスタムリソース属性の説明	8
1.4. JAEGER 監視用の FUSE ONLINE の設定	15
1.5. 外部データベースを使用した FUSE ONLINE のインストール	17
1.6. FUSE ONLINE のインストール	19
第2章 FUSE ONLINE 環境の設定変更	25
2.1. インストール後に変更可能な FUSE ONLINE の設定	25
2.2. FUSE ONLINE の設定変更に関する一般的な手順	27
2.3. FUSE ONLINE の設定および 3SCALE での API の検出	28
2.4. FUSE ONLINE POD の設定	31
2.5. HTTP プロキシの設定	33
2.6. OCP で稼働している FUSE ONLINE 環境へのサンプルデータの追加	34
第3章 OCP 上の FUSE ONLINE の管理	36
3.1. FUSE ONLINE コンポーネントの監査	36
3.2. PROMETHEUS での FUSE ONLINE インテグレーションおよびインフラストラクチャーコンポーネントの監視	37
3.3. FUSE ONLINE のメータリングラベル	42
3.4. FUSE ONLINE 環境のバックアップ	44
3.5. FUSE ONLINE 環境の復元	46
3.6. FUSE ONLINE のアップグレード	48
3.7. OCP プロジェクトからの FUSE ONLINE のアンインストール	50
3.8. FUSE ONLINE が含まれる OCP プロジェクトの削除	50
第4章 FUSE ONLINE のパブリック REST API エンドポイントの呼び出し方法	52
4.1. 外部ツールによって使用される FUSE ONLINE パブリック REST API の公開	52
4.2. FUSE ONLINE パブリック REST API エンドポイントのベース URL の説明	54
4.3. パブリック REST API エンドポイントを呼び出すためのシークレットトークンの取得	55
4.4. インテグレーション ID の検索方法	56
4.5. FUSE ONLINE パブリックエンドポイントを呼び出す CURL コマンドを指定するための形式	56
第5章 外部ツールを使用した CI/CD の FUSE ONLINE インテグレーションのエクスポート/インポート	59
5.1. CI/CD のインテグレーションのマーク付け	59
5.2. CI/CD のインテグレーションのマーク付け	59
5.3. CI/CD の環境ラベルの管理	60
5.4. FUSE ONLINE パブリック API エクスポートエンドポイントの呼び出し	61
5.5. FUSE ONLINE パブリック API インポートエンドポイントの呼び出し	64
第6章 FUSE ONLINE のパブリック REST API エンドポイントの参照情報	66
6.1. インテグレーションの状態を取得するためのエンドポイント	66
6.2. インテグレーションの環境ラベルの一覧を取得するためのエンドポイント	67
6.3. インテグレーションをマーク付けし、未指定のタグを維持するためのエンドポイント	68
6.4. インテグレーションをマーク付けし、未指定のタグを削除するためのエンドポイント	70
6.5. インテグレーションをパブリッシュするためのエンドポイント	71
6.6. インテグレーションを停止するためのエンドポイント	72
6.7. インテグレーションをエクスポートするためのエンドポイント	73

6.8. インテグレーションをインポートするためのエンドポイント	75
6.9. 特定のインテグレーションから環境ラベルを削除するためのエンドポイント	77
6.10. 環境ラベルを作成するためのエンドポイント	78
6.11. 環境ラベルの一覧を取得するためのエンドポイント	79
6.12. 環境ラベルを変更するためのエンドポイント	80
6.13. すべてのインテグレーションから環境ラベルを削除するためのエンドポイント	81
6.14. コネクションのプロパティを変更するためのエンドポイント	82
第7章 FUSE ONLINE ユーザーインターフェイスのカスタマイズ	85

前書き

Fuse Online をオンサイトで OpenShift Container Platform (OCP) にインストールし、操作することができます。Fuse Online がオンサイトで稼働している場合、Fuse Online が OpenShift Dedicated で稼働しているときに提供される機能以外に追加機能を使用できます。

オンサイトという用語はお客様が管理する環境を意味します。Red Hat は OpenShift 環境を管理しません。お客様が管理する環境には、オンプレミス、クラウド、またはハイブリッドクラウドがあります。

詳細は以下のトピックを参照してください。

- [1章OCP 4.x での Fuse Online のインストール](#)
- [2章Fuse Online 環境の設定変更](#)
- [3章OCP 上の Fuse Online の管理](#)
- [4章Fuse Online のパブリック REST API エンドポイントの呼び出し方法](#)
- [5章外部ツールを使用した CI/CD の Fuse Online インテグレーションのエクスポート/インポート](#)
- [6章Fuse Online のパブリック REST API エンドポイントの参照情報](#)
- [7章Fuse Online ユーザーインターフェイスのカスタマイズ](#)

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。

第1章 OCP 4.X での FUSE ONLINE のインストール

Fuse Online は、コアインテグレーション機能をサービスとして提供する、柔軟かつカスタマイズ可能なオープンソースプラットフォームです。

Fuse Online の各インストールは、Fuse Online インスタンス (または環境) と呼ばれます。OpenShift プロジェクトには、必ず1つの Fuse Online インスタンスがあります。各 Fuse Online インスタンスには独自の URL があります。単一の OpenShift クラスターに、複数の Fuse Online インスタンスが存在することがあります。

Operator Hub からオペレーターをインストールするための適切なパーミッションを持っている場合は、OpenShift 開発者ユーザーとして Fuse Online をインストールできます。デフォルトの Fuse Online インスタンスまたはカスタマイズされた Fuse Online インスタンスをインストールできます。カスタマイズされた Fuse Online インスタンスの場合、デフォルトのカスタムリソースを編集する必要があります。



重要

Fuse Online インストールプロセスでは、コンテナイメージの Red Hat エコシステムカタログである registry.redhat.io にアクセスする必要があります。

Fuse Online のインストールに関する詳細は、以下を参照してください。

- [「OCP 4.x に Fuse Online をインストールするのに必要な手順の概要」](#)
- [「制限された環境での Fuse Online のインストールに関する考慮事項 \(OCP 4.6 以降\)」](#)
- [「Fuse Online のインストール前にデフォルトのカスタムリソースファイルを編集する必要がある場合」](#)
- [「Fuse Online の設定に使用するカスタムリソース属性の説明」](#)
- [「Jaeger 監視用の Fuse Online の設定」](#)
- [「コンテナイメージの registry.redhat.io を使用した認証」](#)
- [「Fuse Online のインストール」](#)

1.1. OCP 4.X に FUSE ONLINE をインストールするのに必要な手順の概要

OCP 4.x に Fuse Online をインストールするための主なステップは次のとおりです。

1. Red Hat コンテナイメージへの認証を設定する OpenShift シークレットを生成します。
2. OperatorHub からクラスターのプロジェクト (namespace) に Fuse Online Operator をインストールします。
3. 任意で、コネクションおよびインテグレーションの定義を永続化するために外部データベースを含める場合は、OpenShift シークレットを作成します。
4. Fuse Online インスタンスを OpenShift 4.x プロジェクトに追加します。
任意で、カスタムリソースを編集して、1つ以上のアドオン機能を有効にしたり、1つ以上のカスタム設定を実装したりします。

- 必要に応じて、他の開発者ユーザーに Fuse Online Web コンソールにアクセスできるようにパーミッションを付与します。

1.1.1. 制限された環境での Fuse Online のインストールに関する考慮事項 (OCP 4.6 以降)

制限された環境に FuseOnline をインストールする前に、次のタスクを完了する必要があります。

前提条件

- 制限された環境での OpenShift クラスタで、クラスタ管理者アクセス権がある。

手順

- プライベートネットワークで利用可能な場所に、すべての Fuse Online イメージをミラーリングします。制限された環境での OpenShift Operator のイメージのインストールに関する詳細は、OpenShift ドキュメントの [ネットワークが制限された環境での Operator Lifecycle Manager の使用](#) セクションを参照してください。
- Fuse リポジトリの内容を使用してカスタム Maven リポジトリを設定します。詳細については、Red Hat ソリューションの [How to create an offline Maven repository for Fuse 7](#) を参照してください。

- Fuse Online をインストールする前に、Fuse Online カスタムリソースを編集します。**maven:mirror** 設定を使用して、統合をビルドする際に、Maven アーティファクトにアクセスするために指定された単一の Maven リポジトリのみを参照するように Fuse Online に指示します。

次の例では、**https://customRepo** をオフラインリポジトリの URL に置き換えます。

```
components:
  server:
    features:
      maven:
        mirror: https://customRepo
```

- HTTPS_PROXY**、**HTTP_PROXY**、および **NO_PROXY** 環境変数を **syndesis-oauthproxy** に設定します。これらの環境値は、Fuse Online のインストール時またはその後に設定できます。Fuse Online をインストールした後、**syndesis/app** で環境値を設定するには、以下を行います。

- 次のコマンドを使用して、プロキシ/クラスタから値を取得します。

```
myhttpProxy=$(oc get proxy/cluster -ojsonpath='{.status.httpProxy}')
myhttpsProxy=$(oc get proxy/cluster -ojsonpath='{.status.httpsProxy}')
mynoproxy=$(oc get proxy/cluster -ojsonpath='{.status.noProxy}')
```

- 次の `oc patch` コマンドを使用して値を設定します。

```
oc patch syndesis/app --type=merge -p '{"spec": {"components": {"oauth": {"environment": {"HTTPS_PROXY": "'${myhttpsProxy}'"', "HTTP_PROXY": "'${myhttpProxy}'"', "NO_PROXY": "'${mynoproxy}'"}}}}'
```

このコマンドは、次の確認を返します。

```
syndesis.syndesis.io/app patched
```

syndesis-oauthproxy デプロイメントを変更したため、OpenShift は **syndesis-oauthproxy-1-deploy** Pod を再作成します。

- デフォルトでは、**todo** サンプルアプリケーションは無効になっています (Fuse Online カスタムリソースでは、**addon:todo:enabled** 値は **false** に設定されています)。制限された環境に Fuse Online をインストールした後に、任意で <https://github.com/syndesisio/todo-example> から **todo** サンプルアプリケーションをダウンロードできます。Readme ファイルの手順に従い、リポジトリ URL をプライベートネットワークで利用可能な場所に変更します。

1.2. FUSE ONLINE のインストール前にデフォルトのカスタムリソースファイルを編集する必要がある場合

Fuse Online のインストールには、設定可能な Fuse Online 環境アドオン機能およびパラメーター設定のデフォルトを指定するデフォルトカスタムリソースが含まれています。

インストール後の Fuse Online 環境が以下に該当する場合は、Fuse Online をインストールする前に、デフォルトのカスタムリソースを編集する必要があります。

- Fuse Online コンソールにアクセスできる OpenShift ルートのために指定する URL を使用する場合。デフォルトでは、インストールプロセスによってこのルートが算出されます。
- 外部データベースを使用して、コネクションおよびインテグレーション定義を保存する場合。デフォルトでは、内部データベースが使用されます。
- コネクションやインテグレーション定義の永続化に使用できる内部ストレージの容量を増やす場合。ほとんどの Fuse Online 環境では、デフォルトの **1Gi** で十分対応できます。

上記のいずれかが該当する場合に Fuse Online 環境を設定するには、Fuse Online のインストール時にカスタムリソースを編集する必要があります。つまり、インストール後に Fuse Online 環境の設定を変更して、上記を実装することはできません。また、上記の設定を適用して Fuse Online 環境をインストールした場合、その設定を変更することはできません。

1.3. FUSE ONLINE の設定に使用するカスタムリソース属性の説明

インストール前だけに指定できるカスタムリソース属性の他に、インストール前または後に変更できるカスタムリソース属性が複数あります。

表1には、設定可能なカスタムリソース設定の簡単な説明が記載されています。また、インストールの前または後、あるいは両方で変更可能であるかどうかを示しています。適切に Fuse Online を設定するには、以下の表の情報を活用して、インストール前にカスタムリソースをどのように変更する必要があるのか、またはインストール後にカスタムリソースをどのように変更するのを見極める必要があります。その後、該当する手順にしたがいます。

OCP 4.x の場合:

- [Fuse Online インスタンスの OpenShift 4.x プロジェクトへの追加](#)
- [Fuse Online の設定変更に関する一般的な手順](#)

表1.1 設定可能なカスタムリソース設定

機能/設定	設定可能なタイミング	仕様
<p>強化されたアクティビティ追跡</p> <p>強化されたアクティビティ追跡の追加情報はこの表の後に記載されています。</p>	<p>インストール前のみ設定</p>	<p>addons: jaeger: enabled: true clientOnly: false operatorOnly: false</p> <p>強化されたアクティビティ追跡はデフォルトで有効になっています。Jaeger 設定をカスタマイズする場合は、clientOnly または operatorOnly を true に設定します。</p>
<p>外部データベース</p> <p>外部データベースの使用に関する追加情報はこの表の後に記載されています。</p>	<p>インストール前のみ設定</p>	<p>spec: components: database: externalDbURL: postgresql://custom-postgres:5432 user: db-user-name name: db-name</p> <p>custom-postgres:5432 を PostgreSQL データベースのホスト名およびポートに置き換えます。 db-user-name をそのデータベースにアクセス可能なユーザーアカウントの名前に置き換えます。 db-name をデータベースの名前に置き換えます。</p>
<p>コネクションおよびインテグレーションの 内部ストレージ容量。</p> <p>外部データベースも指定する場合は無視されます。</p> <p>内部ストレージの増加に関する追加情報はこの表の後に記載されています。</p>	<p>インストール前のみ設定</p>	<p>spec: components: database: resources: volumeCapacity: 1Gi volumeName: my-volume</p> <p>1Gi を必要なストレージ容量に置き換えます。デフォルトは 1Gi です。</p> <p>my-volume を内部ストレージに使用するボリュームの名前に置き換えます。このパラメーターは任意です。</p>
<p>Fuse Online コンソールにアクセスするための OpenShift ルート</p>	<p>インストール前のみ設定</p>	<p>spec: routeHostname: project.route.com</p> <p>project.route.com を Fuse Online コンソールにアクセスできる OpenShift ルートに置き換えます。 例: north-project.6a63.fuse-online.openshiftapps.com</p>

機能/設定	設定可能なタイミング	仕様
<p>Memory および CPU</p> <p>1つ以上のコンポーネントが使用できるデフォルトのメモリー容量を増やします。</p> <p>各コンポーネントによって独自のメモリー要件が定義されるため、各 Pod には割り当てられるメモリー容量の制限があります。制限および要求設定の詳細は、OpenShift ドキュメントの Configuring cluster memory to meet container memory and risk requirements セクションを参照してください。</p> <p>server コンポーネントの CPU リソースを指定することもできます。</p> <p>database コンポーネントはコネクションおよびインテグレーションの定義を格納する内部データベースです。</p> <p>meta コンポーネントはサーバーがロードするコネクタなどのビジネスロジックを提供します。</p> <p>prometheus コンポーネントは、Fuse Online インフラストラクチャーコンポーネントおよび Fuse Online インテグレーションを監視します。</p>	<p>インストール前のみ設定</p>	<pre> components: server: resources: limit: memory: "1024Mi" cpu: "800m" request: memory: "512Mi" cpu: "500m" meta: resources: limit: memory: "750Mi" request: memory: "300Mi" database: resources: limit: memory: "300Mi" request: memory: "300Mi" prometheus: resources: limit: memory: "750Mi" request: memory: "750Mi" </pre>
<p>3scale の検出</p> <p>Red Hat 3scale による検出を可能にするため、Fuse Online API プロバイダーインテグレーションの API を公開します。</p> <p>詳細は、Configuring Fuse Online to enable 3scale discovery of APIs を参照してください。</p>	<p>インストール前または後に設定</p>	<pre> components: server: features: managementUrlFor3scale: https://url-for-3scale </pre> <p>3scale ユーザーインターフェイスの URL を指定します。</p>

機能/設定	設定可能なタイミング	仕様
<p>Backups</p> <p>追加の設定ステップは、Fuse Online 環境のバックアップを参照してください。</p>	<p>インストール前または後に設定</p>	<p>spec: backup: schedule: interval</p> <p>interval を希望するバックアップの間隔に置き換えます。間隔 および 事前定義のスケジュール には cron ユーティリティーの形式を使用します。間隔の前に @ 記号を指定しないでください。</p>
<p>ノードのアフィニティー と 許容 (Toleration)</p> <p>Fuse Online インフラストラクチャーコンポーネントおよびインテグレーション Pod のクラスター内のノードへの配置を決定します。</p> <p>ノードのアフィニティーを使用すると、配置先のノードのグループに対して Fuse Online Pod のアフィニティーを指定できます。</p> <p>許容 (Toleration) を使用すると、Fuse Online Pod が実行されるノードを制御し、他のワークロードがそれらのノードを使用しないようにすることができます。</p> <p>Fuse Online Pod の設定 も参照してください。</p>	<p>インストール前または後に設定</p>	<p>注記: Fuse Online インフラストラクチャーコンポーネントのデプロイメントには infraScheduling を使用します。Fuse Online インテグレーションのデプロイメントでは、infraScheduling を integrationScheduling に置き換えます。</p> <p>spec: infraScheduling: tolerations: key: value operator: value effect: value</p> <p>spec: infraScheduling: affinity: nodeAffinity: preferredDuringSchedulingIgnoredDuringExecution: weight: preference: matchExpressions: key: value operator: value values: value1, value2</p>
<p>インテグレーションの制限</p> <p>稼働中のインテグレーションの最大数を指定します。デフォルトは 0 で、稼働中のインテグレーションの数を制限しません。</p>	<p>インストール前または後に設定</p>	<p>components: server: features: integrationLimit: 0</p>

機能/設定	設定可能なタイミング	仕様
<p>データベースコネクションプール</p> <p>サーバーコネクションプール設定を調整して、データベースへのコネクションを管理できます。</p> <p>データベースプール設定の追加情報は、この表の後に記載されています。</p>	<p>インストール前または後に設定</p>	<p>components:</p> <p>server:</p> <p>connectionPool:</p> <p>connectionTimeout: 30000</p> <p>idleTimeout: 600000</p> <p>leakDetectionThreshold: 0</p> <p>maxLifetime: 1800000</p> <p>maximumPoolSize: 10</p> <p>minimumIdle: 10</p>
<p>Java オプション</p> <p>syndesis-server コンポーネントおよび syndesis-meta コンポーネントの Java オプションを指定できます。</p> <p>Java オプションに応じて、コンポーネントに異なる値を指定できます。たとえば、JVM 関連のパラメーターは syndesis-meta よりも多くのコンピューターリソースを必要とするため、syndesis-server のみに設定できます。</p>	<p>インストール前または後に設定</p>	<p>components:+ server:</p> <p>javaOptions:</p> <p>-option_name=my_value meta:</p> <p>javaOptions:</p> <p>-option_name=my_value</p> <p>-option_name を Java オプション名に置き換えます。Java オプションの接頭辞 (-D、-X、または -XX) を指定することができます。</p> <p>my_value を、オプションの値に置き換えます。</p> <p>HTTP プロキシを設定する場合 (例): components:+ server:</p> <pre>javaOptions: -Dhttp.proxyHost=10.0.0.100 - Dhttp.proxyPort=8800 meta: javaOptions: -Dhttp.proxyHost=10.0.0.100 - Dhttp.proxyPort=8800`</pre>
<p>Maven 引数</p> <p>Fuse Online インテグレーションをビルドするときに使用する追加の Maven オプションを指定します。</p>	<p>インストール前または後に設定</p>	<p>components:</p> <p>server:</p> <p>features:</p> <p>maven:</p> <p>additionalArguments:</p> <p>"typeA=stringA typeB=stringB"</p> <p>For example:</p> <p>additionalArguments:</p> <p>"-Dhttp.proxy=my_proxy -DpropA=valueA"</p>

機能/設定	設定可能なタイミング	仕様
<p>Maven Mirror</p> <p>インテグレーションをビルドするときに Fuse Online が Maven アーティファクトにアクセスするために使用する単一リポジトリを指定できます。Fuse Online がデフォルトでアクセスする Maven リポジトリを無視します。このオプションは通常、制限された環境に FuseOnline をインストールするときに使用されます。</p>	<p>インストール前または後に設定</p>	<p>components: server: features: maven: mirror: https://customRepo</p> <p>Replace `https://customRepo` をリポジトリの URL に置き換えます。</p>
<p>Maven リポジトリ</p> <p>Fuse Online がデフォルトでアクセスする Maven リポジトリの他に、Fuse Online 環境がアクセスする必要のある外部 Maven リポジトリを指定します。</p>	<p>インストール前または後に設定</p>	<p>components: server: features: mavenRepositories: customRepo1: https://customRepo1 customRepo2: https://customRepo2</p> <p>Replace customRepo をリポジトリの名前に置き換えます。 各リポジトリに URL を指定します。</p>
<p>監視</p> <p>Prometheus を使用した OCP での Fuse Online インテグレーションの監視も参照してください。</p>	<p>インストール前または後に設定</p>	<p>addons: ops: enabled: true</p>
<p>パブリック REST API</p> <p>追加の設定ステップは 外部ツールによって使用される Fuse Online パブリック REST API の公開を参照してください。</p>	<p>インストール前または後に設定</p>	<p>addons: publicApi: enabled: true routeHostname: public-syndesis.192.168.64.63.nip.io</p> <p>routeHostname を Fuse Online REST API エンドポイントを呼び出すためのパブリックアドレスに設定します。クラスターのセットアップにより、指定する必要のあるパブリックアドレスが決定されます。この例では、ルートのホスト名は minishift クラスターに対して有効です。</p>

機能/設定	設定可能なタイミング	仕様
<p>ToDo アドオン テスト用のサンプルインテグレーションアプリケーション、AMQ ブローカー、およびサンプル PostgreSQL データベースとサンプルデータベースコネクタを提供します。</p>	<p>インストール前または後に設定</p>	<p>spec: addons: todo: enabled: false</p> <p>制限された環境で Fuse Online をインストールするには、インストールの前に todo アドオンが false (デフォルト) に設定されていることを確認する必要があります。Fuse Online のインストール後に、任意で https://github.com/syndesisio/todo-example から ToDo アプリケーションをダウンロードできます。Readme ファイルの手順に従い、リポジトリ URL をプライベートネットワークで利用可能な場所に変更します。</p> <p>Adding sample data to a Fuse Online environment running on OCP も参照してください。</p>
<p>監査</p> <p>コネクタ、接続、インテグレーションに加えられた基本的な変更を表示します。</p> <p>Auditing Fuse Online components も参照してください。</p> <p>注記: これはテクノロジープレビューの機能です。</p>	<p>インストール前または後に設定</p>	<p>components: server: features: auditing: true</p>

アドオン機能と設定

- **強化されたアクティビティ追跡**

Fuse Online のインストール時に、Jaeger を使用したアクティビティ追跡はデフォルトで有効になっています。(OperatorHub またはコマンドラインスクリプトを使用して) Fuse Online をインストールする場合、OperatorHub の存在を検出し、OperatorHub のサブスクリプション機能を使用して Jaeger をインストールします。OperatorHub が利用できないという制限された状況では、Fuse Online は独自のインストール機能を使用して Jaeger をインストールします。任意で、[Jaeger 監視用の Fuse Online の設定](#) のセクションで説明されているように、Fuse Online をインストールする前に Jaeger 設定をカスタマイズできます。

- **コネクションおよびインテグレーションを永続化するための外部データベース**

Fuse Online のデフォルトインストールは、コネクションおよびインテグレーション定義を永続化するために Fuse Online によって使用される内部 PostgreSQL データベースを提供します。代わりに、OpenShift がデフォルトで提供する PostgreSQL テンプレートの1つなど、外部 PostgreSQL データベースを使用することを選択できます。

- **内部ストレージ容量**

ほとんどの Fuse Online 環境では、デフォルトの **1Gi** で十分対応できます。Red Hat テクニカルサポートが推奨する場合のみ、新しい Fuse Online インストールのこの設定を増強することが想定されます。これは、稼働している別の Fuse Online 環境で Fuse Online のサーバーエラー

が発生し、Red Hat テクニカルサポートが、デフォルトよりも容量が大きいデータベースボリュームで Fuse Online 環境をインストールする必要があると判断した場合は。

Fuse Online がすでに稼働している OpenShift プロジェクトで Fuse Online の内部ストレージ容量を増やすには、最初に Fuse Online をアンインストールする必要があります。[OCP プロジェクトからの Fuse Online のアンインストール](#) を参照してください。

● データベースコネクションプールの設定

以下の **syndesis-server** データベース接続プールプロパティを設定できます。

- **connectionTimeout - syndesis-server** がプールからのコネクションを待機する最大時間 (ミリ秒単位)。許可される最小コネクションタイムアウトは 250 ミリ秒です。デフォルトは 30000 (30 秒) です。
- **idleTimeout** - コネクションが削除されるまで、プールでアイドル状態にすることができる最大時間 (ミリ秒単位)。0 の値は、アイドルコネクションがプールから削除されないことを意味します。許可される最小値は 10000 (10 秒) です。デフォルトは 600000 (10 分) です。
- **leakDetectionThreshold** - コネクションリークの可能性を示すメッセージがログに記録される前に、コネクションがプール外にある時間 (ミリ秒単位)。0 を値として指定すると、リーク検出が無効になります。リーク検出の有効化で許容される最小値は 2000 (2 秒) です。デフォルトは 0 です。
- **maxLifetime** - プール内のコネクションの最大有効期間 (ミリ秒単位)。許可される最小値は 30000 (30 秒) です。デフォルトは 1800000 (30 分) です。
- **maximumPoolSize** - アイドルコネクションと使用中のコネクションの両方を含む、プールが到達できる最大サイズ。デフォルトは 10 です。
- **minimumIdle** - プールに保持されるアイドルコネクションの最小数。デフォルト値は **maximumPoolSize** です。

1.4. JAEGER 監視用の FUSE ONLINE の設定

Jaeger は、分散サービス間のトランザクションをトレースするためのオープンソースソフトウェアです。これは、複雑なマイクロサービス環境の監視およびトラブルシューティングで特に役立ちます。

OperatorHub またはコマンドラインスクリプトを使用して Fuse Online をインストールする場合、Fuse Online インストールは OperatorHub の存在を検出し、OperatorHub のサブスクリプション機能を使用して Jaeger をインストールします。OperatorHub が利用できないという制限された状況では、Fuse Online は独自のインストール機能を使用して Jaeger をインストールします。

デフォルトの Fuse Online 環境では、必要な Jaeger コンポーネントがすべて設定されます。任意で、Fuse Online カスタムリソースを編集して、クライアントのみ/非依存のサーバー設定、またはハイブリッド Jaeger クライアントおよび Jaeger Operator 設定を指定できます。

デフォルトの Jaeger 設定

基本的な追加設定なしの設定には、すべての Jaeger コンポーネントが含まれます。Jaeger の監視機能を試して、Jaeger の動作について学ぶことができます。デフォルト設定では、メモリーのみの制限されたバックエンドストレージ機能を提供します。

デフォルトの Jaeger 設定で Fuse Online をインストールすると、以下のような結果になります。

- Fuse Online コンポーネントには Jaeger 通信 URL があります。

- Jaeger Operator がインストールされます。
- Jaeger カスタムリソースは、アクティビティ監視のデフォルト設定で設定されます。

デフォルトの Jaeger 設定を使用した Fuse Online インストールには、以下の **syndesis** カスタムリソース仕様があります。

```
apiVersion: syndesis.io/v1beta2
kind: Syndesis
metadata:
  name: app
spec:
  addons:
    jaeger:
      enabled: true
```

clientOnly および **operatorOnly** が指定されていない場合 (デフォルトで **false** に設定されている場合)、Fuse Online は提供される Jaeger バックエンドだけでなく、Jaeger サーバー設定によって提供されるデフォルトのメモリーのみのストレージも使用します。

クライアントのみ/非依存のサーバー設定

クライアントのみ/非依存のサーバー設定では、Fuse Online と外部に設定された Jaeger バックエンドとの間の通信用にクライアント URL コネクションのみが設定されます。Jaeger バックエンドのすべての要素は外部にあり、Fuse Online 環境および **syndesis-operator** には依存しません。これには、Jaeger Operator および Jaeger カスタムリソースが含まれます。

クライアントのみの Jaeger 設定をインストールするには、以下のように Fuse Online をインストールする前にカスタムリソースを編集します。

- **clientOnly** を **true** に設定して、[Jaeger エージェント機能](#) を追加します。
- **queryUri** を、個別にインストールされた Jaeger バックエンドの [query コンポーネント](#) の URI に設定します。
- **collectorUri** を、個別にインストールされた Jaeger バックエンドの [collector コンポーネント](#) の URI に設定します。

以下に例を示します。

```
apiVersion: syndesis.io/v1beta2
kind: Syndesis
metadata:
  name: app
spec:
  addons:
    jaeger:
      enabled: true
      clientOnly: true
      queryUri: https://jaeger-query-hostname:443/api
      collectorUri: https://jaeger-collector-hostname:14268/api/traces
```

ハイブリッド Jaeger クライアントおよび Operator の設定

ハイブリッド Jaeger クライアントおよび Jaeger Operator 設定では、Fuse Online は Jaeger Operator と Jaeger クライアント機能をインストールします。Jaeger カスタムリソースがインストールされてい

ない。Jaeger サーバー設定を定義する独自の Jaeger カスタムリソースをインストールする必要があります。これにより、Fuse Online が提供する機能を利用し、独自の環境用に Jaeger 設定をカスタマイズすることができます。たとえば、データストレージに Elasticsearch または Cassandra を使用できません。

ハイブリッド Jaeger クライアントと Jaeger Operator 設定をインストールするには、以下を行います。

- Fuse Online をインストールする前に、以下の例のようにカスタムリソースを編集します。

```
apiVersion: syndesis.io/v1beta2
kind: Syndesis
metadata:
  name: app
spec:
  addons:
    jaeger:
      enabled: true
      operatorOnly: true
```

- 以下の例のように、Jaeger カスタムリソース **syndesis-jaeger** に名前を付けます。

```
apiVersion: jaegertracing.io/v1
kind: Jaeger
metadata:
  name: syndesis-jaeger
...
spec:
  ....
```

注記: Jaeger カスタムリソースに別の名前を使用する場合は、Client-only/independent サーバー設定セクションに記述されているように、Syndesis カスタムリソースで **queryUri** および **collectorUri** を設定します。

syndesis-jaeger という名前の Jaeger カスタムリソースが作成されると、Jaeger インスタンスは Fuse Online インテグレーションからデータを収集します。デフォルトでは、このデータは Fuse Online アクティビティログで表示できます。

1.5. 外部データベースを使用した FUSE ONLINE のインストール

外部データベースを使用して接続と統合の定義を永続化する Fuse Online 環境をインストールする場合、以下を実行します。

- OpenShift クラスターが検出できるホスト名を使用して PostgreSQL データベースを作成します。
- Fuse Online をインストールする前に、外部データベース用に **syndesis-global-config** という名前の OpenShift シークレットを作成します。
- Fuse Online をインストールします。
- Fuse Online をデプロイする前に、カスタムリソースを編集して、外部データベースへの接続を設定します。

前提条件

- Fuse Online がインストールされていない必要があります。
- **oc** クライアントツールがインストール済みであり、Fuse Online をインストールする OCP クラスターに接続されている必要があります。
- クラスター管理者権限を持つユーザーは、該当ユーザーがクラスターでアクセス権限を持つプロジェクトに Fuse Online をインストールするための権限を付与済みである必要があります。

手順

1. Fuse Online をインストールする権限を持つアカウントで OpenShift にログインします。以下に例を示します。

```
oc login -u developer -p developer
```

2. Fuse Online をインストールする Open Shift クラスターで使用可能で、以下の設定オプションの値を定義する PostgreSQL データベースを作成します。

- 接続ユーザー名
- データベース名
- データベースのパスワード

次の例では、OpenShift がデフォルトで提供する PostgreSQL テンプレートの1つを使用しています。

注記: 開発またはテスト目的には PostgreSQL 一時 (**postgresql-ephemeral**) テンプレートのみを使用します。実稼働環境の場合は、PostgreSQL (**postgresql**) テンプレートを使用します。

```
oc new-app postgresql-ephemeral -p POSTGRESQL_USER=syndesis -p
POSTGRESQL_PASSWORD=exdb-pwd -p POSTGRESQL_DATABASE=syndesisdb
```

このコマンドは、URL など、Fuse Online を設定するときに必要なデータベースに関する情報を返します。

データベースの作成については、[PostgreSQL topic in the OpenShift documentation](#) を参照してください。

3. 以下の内容を含むリソースファイル (例: **my-fuse-online-secret-cr.yml**) を作成して、保存します。

```
apiVersion: v1
kind: Secret
metadata:
  name: syndesis-global-config
  namespace: my-fuse-online-project
type: Opaque
stringData:
  POSTGRESQL_PASSWORD: exdb-pw
```

my-fuse-online-project を、外部データベースであるス指定する Fuse Online 環境をインストールする予定である OpenShift プロジェクトの名前に置き換えます。

exdb-pw を手順 2 のパスワードに置き換えます。Fuse Online は、外部データベースにアクセスするためのパスワードとして使用します。**stringData** は、パスワードの値を base64 でエンコードされた値 (変換を行わないように) に変換することに注意してください。

OpenShift シークレットの詳細は、[Providing sensitive data to pods](#) を参照してください。

4. 以下のように、シークレットをクラスターに追加します。

```
oc apply -f my-fuse-online-secret-cr.yml
```

OpenShift クラスターでは、外部データベースを指定するカスタムリソースとインストールされた Fuse Online 環境で **syndesis-global-config** シークレットを利用できます。

5. シークレット (ステップ 4) の **YAML** ファイルで指定した **my-fuse-online-project** プロジェクトに Fuse Online をインストールします。
 - OCP 4.6 以降の場合は、[Installing the Fuse Online Operator from the OperatorHub](#) の手順に従います。[Adding a Fuse Online instance to an OpenShift 4.x project](#) において、Fuse Online をデプロイする前にカスタムリソースを編集して、外部データベースの使用を指定します。以下に例を示します。

```
apiVersion: syndesis.io/v1alpha1
kind: Syndesis
metadata:
  name: app
spec:
  components:
    database:
      externalDbURL: "postgresql://syndesis-external-db:5432"
      user: db-user-name
      name: db-name
```

- **syndesis-external-db:5432** を外部 PostgreSQL データベースのホスト名およびポートに置き換えます。
- **db-user-name** をコネクションユーザー名に置き換えます。
- **db-name** をデータベースの名前に置き換えます。

結果

Fuse Online のインストールプロセスでは、カスタムリソースの設定を使用して、インストールされた Fuse Online 環境の設定を決定します。

正しくインストールおよびデプロイされると、Fuse Online は **syndesis-db** 以外の必要な Pod をすべて稼働します。**syndesis-db** の代わりに、**syndesis-server** は外部データベースのサービス名を使用してデータベースに接続します。外部データベースの設定が正しくない場合、**syndesis-server** は初期化に失敗し、Fuse Online デプロイメントはエラーを出して終了します。

1.6. FUSE ONLINE のインストール

Fuse Online は、コアインテグレーション機能をサービスとして提供する、柔軟かつカスタマイズ可能なオープンソースプラットフォームです。Operator Hub からオペレーターをインストールするための適切なパーミッションを持っている場合は、OpenShift 開発者ユーザーとして Fuse Online をインストールできます。

以下は、Operator を使用して Fuse Online をインストールする一般的な手順です。

1. OpenShift プロジェクト (namespace) にシークレットを作成し、Fuse Online Operator が必要なインストールテンプレートにアクセスできるように、**registry.redhat.io** との認証を設定します。
2. Fuse Online Operator を OpenShift プロジェクト (namespace) にインストールします。Operator はインストール後に、選択された namespace で実行されます。
3. インストールされた Operator から Fuse Online のインスタンスを作成します。デフォルト設定でインスタンスを作成するか、インスタンスのカスタムリソースを編集してインスタンスをカスタマイズすることができます。続いて、提供された URL から Fuse Online にアクセスできます。

Operator を使用して OCP 4.x に Fuse Online をインストールする方法の詳細は、以下を参照してください。

- [「コンテナイメージの registry.redhat.io を使用した認証」](#)
- [「OperatorHub からの Fuse Online Operator のインストール」](#)
- [「Fuse Online インスタンスの OpenShift 4.x プロジェクトへの追加」](#)

1.6.1. コンテナイメージの registry.redhat.io を使用した認証

Fuse コンテナイメージを OpenShift にデプロイする前に、**registry.redhat.io** で認証を設定します。

前提条件

- OpenShift Container Platform クラスターへアクセスできるクラスター管理者権限。
- OpenShift **oc** クライアントツールがインストール済みであること。詳細は、[OpenShift CLI のドキュメント](#) を参照してください。

手順

1. 管理者として OpenShift クラスターにログインします。

```
oc login --user system:admin --token=my-token --server=https://my-cluster.example.com:6443
```

2. Fuse をデプロイするプロジェクトを開きます。

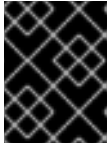
```
oc project myproject
```

3. Red Hat Customer Portal アカウントを使用して **docker-registry** シークレットを作成し、**PULL_SECRET_NAME** を **psi-internal-registry** に置き換えて以下を作成します。

```
oc create secret docker-registry psi-internal-registry \
  --docker-server=docker-registry.redhat.io \
  --docker-username=CUSTOMER_PORTAL_USERNAME \
  --docker-password=CUSTOMER_PORTAL_PASSWORD \
  --docker-email=EMAIL_ADDRESS
```

以下の出力が表示されるはずですが。

secret/psi-internal-registry created



重要

この **docker-registry** シークレットを、**registry.redhat.io** に対して認証されるすべての OpenShift プロジェクト namespace に作成する必要があります。

- シークレットをサービスアカウントにリンクして、シークレットをイメージをプルするために使用します。次の例では、**default** サービスアカウント、**builder** サービスアカウント、および **deployer** サービスアカウントを使用しています。

```
oc secrets link default psi-internal-registry
oc secrets link default psi-internal-registry --for=pull
oc secrets link builder psi-internal-registry
oc secrets link builder psi-internal-registry --for=pull
oc secrets link deployer psi-internal-registry
oc secrets link deployer psi-internal-registry --for=pull
```

サービスアカウント名は、OpenShift Pod が使用する名前と一致する必要があります。



注記

Red Hat のユーザー名とパスワードを使用してプルシークレットを作成したくない場合は、レジストリーサービスアカウントを使用して認証トークンを作成できます。

関連情報

コンテナイメージに対する Red Hat の認証に関する詳細は、以下を参照してください。

- [Red Hat コンテナイメージの認証](#)
- [Red Hat registry service accounts](#)

1.6.2. OperatorHub からの Fuse Online Operator のインストール

OpenShift Container Platform Web コンソールを使用して、OperatorHub から Fuse Online Operator をインストールできます。Fuse Online をインストールする各 OpenShift プロジェクト (namespace) について、以下の手順を実行します。

前提条件

- OpenShift クラスターの管理者権限または開発者権限を持っている必要があります。開発者ユーザーの場合、Operator Hub から operator をインストールするための適切な権限がある。
- コンテナイメージの [registry.redhat.io を使用した認証](#) で説明されているように、**registry.redhat.io** との認証を設定している。
- **oc** クライアントツールがインストール済みであり、Fuse Online をインストールする OCP クラスターに接続されている必要がある。

手順

1. Web ブラウザーで OpenShift コンソールに移動し、管理者または開発者のクレデンシャルを使用してコンソールにログインします。
2. 管理者としてログインしている場合には、**Operators** をクリックした後、**OperatorHub** をクリックします。

開発者としてログインしている場合は、**Add** をクリックしてから **From Catalog** カードをクリックします。

3. **Filter by keyword** フィールドに **Fuse Online** を入力します。
4. **Red Hat Integration - Fuse Online** カードをクリックします。 **Fuse Online Operator** のインストールページが開きます。
5. **Install** をクリックします。 **Install Operator** ページが開きます。
 - a. **更新チャンネル** は Operator の更新ストリームを定義し、サブスクライバーの更新をロールアウトするために使用されます。デフォルトのチャンネル、**latest** を受け入れます。
 - b. **Installation mode** では、クラスターの namespace のリストから namespace (プロジェクト) を1つ選択します。 **コンテナイメージの registry.redhat.io による認証** で docker レジストリーシークレットを作成したときに使用したのと同じ namespace を選択します。
 - c. **UpdateApproval** では、**Automatic** または **Manual** を選択し、Fuse Online Operator の更新が OpenShift によってどのように対処されるかを設定します。
 - **Automatic** (自動) 更新を選択した場合、新しいバージョンの Fuse Online Operator が使用できるようになると、人的な介入なしで OpenShift Operator Lifecycle Manager (OLM) によって、Fuse Online の稼働中のインスタンスが自動的にアップグレードされます。
 - **Manual** (手動) 更新を選択した場合、Operator の新しいバージョンが使用できるようになると、OLM によって更新リクエストが作成されます。クラスター管理者は、更新リクエストを手動で承認して、Fuse Online Operator を新しいバージョンに更新する必要があります。

6. **Install** をクリックし、Fuse Online Operator を指定の namespace (プロジェクト) で利用可能にします。
7. Fuse Online がプロジェクトにインストールされていることを確認するには、**Operators** をクリックした後、**Installed Operators** をクリックして **Red Hat Integration - Fuse Online** がリストに表示されることを確認します。
8. ターミナルウィンドウで、次のコマンドを入力して、(**コンテナイメージの registry.redhat.io での認証** で作成した) シークレットを Fuse Online Operator サービスアカウントにリンクします。

```
oc secrets link syndesis-operator PULL_SECRET_NAME --for=pull
```

次のステップ

Fuse Online Operator のインストール後に、Fuse Online のインスタンスを OpenShift プロジェクトに追加します。

1.6.3. Fuse Online インスタンスの OpenShift 4.x プロジェクトへの追加

Fuse Online Operator を OpenShift 4.x プロジェクトにインストールした後に、Fuse Online のインスタンスを OpenShift プロジェクトに追加します。Fuse Online インスタンスは、開発者が Fuse Online にアクセスするために使用する URL を提供します。

前提条件

- Fuse Online Operator は現在の OpenShift プロジェクトにインストールされます。
- OCP 環境では、Fuse Online が使用できる永続ボリュームが少なくとも3つ必要です。すべての永続ボリュームには、以下の設定要件が必要です。
 - `capacity.storage`: **1Gi**
 - `accessMode`: **ReadWriteOnce**
- デフォルトの Fuse Online インスタンスまたはカスタマイズされたインスタンスのどちらをインストールするかを決定します。指定できるカスタムリソース設定の詳細は、[インストール前にデフォルトのカスタムリソースを編集する必要がある場合](#) を参照してください。

手順

1. ユーザーロールに応じて **Create Syndesis** ページにアクセスするには、以下の手順に従います。
 - 管理者としてログインしている場合は、以下を実行します。
 - a. **Operators** をクリックした後、**Installed Operators** をクリックします。
 - b. **Name** 列で、**Red Hat Integration - Fuse Online** をクリックします。**Operator Details** ページが開きます。
 - c. **Provided APIs** の **Create Instance** をクリックします。
 - 開発者としてログインしている場合は、以下を実行します。
 - a. **Add** をクリックしてから **Operator Backed** カードをクリックします。
 - b. **Syndesis CRD** カードをクリックします。**Syndesis CRD** ページが開きます。
 - c. **Create** をクリックします。
Create Syndesis ページが開きます。
2. 名前を入力するか、または **app** をデフォルトのままにします。
3. **YAML view** オプションを選択して、デフォルトのカスタムリソースを表示します。
4. 任意手順:カスタムリソースを編集します。
指定できるカスタムリソース設定の詳細は、[インストール前にデフォルトのカスタムリソースを編集する必要がある場合](#) および [Fuse Online の設定に使用するカスタムリソース属性の説明](#) を参照してください。

Fuse Online のインストールプロセスでは、カスタムリソースで指定した設定を使用して、インストールされた Fuse Online 環境の設定を決定します。

5. **Create** をクリックして Fuse Online インスタンスを作成します。OpenShift は、Fuse Online の Pod、サービス、およびその他のコンポーネントを起動します。

6. Fuse Online の URL を取得するには、以下を行います。
 - a. **Networking > Routes** をクリックします。
 - b. 正しいプロジェクトが選択されていることを確認してください。
 - c. **syndesis** 行の **Location** 列で、Fuse Online の URL をクリックします。
 - d. OpenShift ログインクレデンシャルを使用して、Fuse Online 環境にログインします。
7. 他の開発者ユーザーに Fuse Online の Web コンソールへのアクセスを許可するには、管理者は以下のコマンドを使用して、各開発者ユーザーに Fuse Online がインストールされているプロジェクトの **view** ロールを付与する必要があります。

```
oc adm policy add-role-to-user view <username> -n <project-name>
```

たとえば、**myfuseonline** プロジェクトの **view** パーミッションをユーザー **jdoe** に付与するには、以下を実行します。

```
oc adm policy add-role-to-user view jdoe -n myfuseonline
```

第2章 FUSE ONLINE 環境の設定変更

オンサイトで Fuse Online を OpenShift Container Platform (OCP) にインストールした後、Fuse Online 環境の設定を変更することができます。

詳細は以下のトピックを参照してください。

- [「インストール後に変更可能な Fuse Online の設定」](#)
- [「Fuse Online の設定変更に関する一般的な手順」](#)
- Fuse Online の設定および 3scale での API の検出
 - [「3scale で API の検出を有効化する Fuse Online の設定」](#)
 - [「3scale で API の検出を無効化する Fuse Online の設定」](#)
- [「HTTP プロキシの設定」](#)
- [「OCP で稼働している Fuse Online 環境へのサンプルデータの追加」](#)

2.1. インストール後に変更可能な FUSE ONLINE の設定

オンサイトで Fuse Online を OpenShift Container Platform (OCP) にインストールした後、機能を有効または無効にしたり、設定を変更することができます。以下の表は、変更可能な機能および設定、変更の影響、および変更手順の記載場所を一覧表示しています。

syndesis カスタムリソースへの変更によって **syndesis-server** が再度デプロイされた場合、新しいサーバー Pod が OpenShift によって作成されます。新しいサーバーの準備が整うまで、最大 1-2 分ほどかかります。準備が整うまで Fuse Online コンソールを使用することはできません。新しいサーバーのデプロイ中も、稼働中のインテグレーションは稼働し続けます。新しいデプロイメントの準備が整ったら、新しいサーバーによってインテグレーションの状態がロードされます。**syndesis-server** Pod の状態が **Running** であれば Fuse Online コンソールを使用することができます。Fuse Online コンソールを更新すると、インテグレーション、コネクション、メッセージ、および稼働時間の値が表示されます。

表2.1 変更可能な機能および設定

機能/設定	変更の影響	詳細
3scale の検出	syndesis-server を再度デプロイします。	3scale で API の検出を無効化する Fuse Online の設定
バックアップ	バックアップジョブを syndesis-operator に追加します。	Fuse Online 環境のバックアップ
監査 注記: これはテクノロジープレビューの機能です。	実行中の syndesis-server の設定を更新します。再デプロイは必要ありません。	Auditing Fuse Online components

機能/設定	変更の影響	詳細
モニタリング	新しい OpenShift リソースをインストールします。	Prometheus を使用した OCP での Fuse Online インテグレーションの監視
パブリック Fuse Online REST API エンドポイント	syndesis-public-oauthproxy Pod を作成します。	外部ツールによって使用される Fuse Online パブリック REST API の公開
サンプル Todo アプリ、AMQ ブローカー、サンプル PostresSQL データベースとコネクタ	todo addon は、インテグレーションをテストするためのサンプル Todo アプリ用の新しい todo Pod を作成します。また、AMQ ブローカー、サンプルデータベースおよびサンプルデータベースコネクタも提供します。 sampledb という名前の PostgreSQL データベースを作成し、Fuse Online 環境の SampleDB コネクタに PostgresDB 接続 を追加します。 todo addon オプションを変更すると、 syndesis-server および syndesis-db Pod の再デプロイがトリガーされることに注意してください。	OCP で稼働している Fuse Online 環境へのサンプルデータの追加 Fuse Online の設定に使用するカスタムリソース属性の説明
Java オプション	syndesis-server を再度デプロイします。	Fuse Online の設定に使用するカスタムリソース属性の説明
Maven の設定	再デプロイメントをトリガーするには、 syndesis-server Pod を削除する必要があります。	HTTP プロキシの設定
OpenShift クラスターノードへの Fuse Online Pod の割り当て	Fuse Online インフラストラクチャコンポーネントおよびインテグレーション Pod を再デプロイします。	Fuse Online の設定に使用するカスタムリソース属性の説明 Fuse Online Pod の設定
Fuse Online コンポーネントで使用できるメモリー	メモリーが変更されたコンポーネントを再度デプロイします。	Fuse Online の設定に使用するカスタムリソース属性の説明
許可される稼働中のインテグレーションの数	再デプロイメントをトリガーするには、 syndesis-server Pod を削除する必要があります。	
データベースコネクションプール	syndesis-server を再度デプロイします。	

2.2. FUSE ONLINE の設定変更に関する一般的な手順

OCP にインストールされる Fuse Online 環境では、**syndesis** カスタムリソースを更新して設定を変更できます。変更を保存した後、構文が正しい場合は OpenShift によって **syndesis-operator** が更新されます。構文エラーがある場合は更新されません。

syndesis カスタムリソースの編集時、構文が正しい無効な設定を指定しないようにしてください。構文が正しい設定が無効であっても、更新された **syndesis** カスタムリソースが **syndesis-operator** によって使用されます。無効な設定を指定した場合、OpenShift の更新後でのみ判明します。更新中、OpenShift では無効な設定が削除されます。

前提条件

- オンサイトで OCP に Fuse Online がインストールされている必要があります。
- **oc** クライアントツールがインストール済みであり、Fuse Online がインストールされている OCP クラスタに接続されている必要があります。
- Fuse Online の設定を変更するプロジェクトで Fuse Online をインストールする権限が必要です。
- [Fuse Online の設定に使用するカスタムリソース属性の説明](#) を読み、インストール後に変更可能な設定と、**syndesis** カスタムリソースを更新して設定変更を行う方法について理解する必要があります。

手順

1. Fuse Online をインストールする権限を持つアカウントで OpenShift にログインします。以下に例を示します。

```
oc login -u developer -p developer
```

2. 設定を変更する Fuse Online 環境を実行している OpenShift プロジェクトに切り替えます。以下に例を示します。

```
oc project my-fuse-online-project
```

3. 任意手順:誤った内容を更新および保存した場合に備え、**syndesis** カスタムリソースに現在の設定のバックアップコピーを作成します。たとえば、以下のコマンドを実行すると、バックアップが **syndesis_app_backup.yml** ファイルに保存されます。

```
oc get syndesis app -o yaml > syndesis_app_backup.yml
```

4. 以下のコマンドを実行して、エディターで **syndesis** カスタムリソースを開きます。

```
oc edit syndesis/app
```

注記: この手順では、エディターでカスタムリソースファイルを開き、設定する方法を説明します。この他にも、**oc patch** コマンドを使用して、カスタムリソースを編集することもできます。

5. 必要に応じて **syndesis** カスタムリソースを編集します。
6. リソースを保存します。
編集が適用されるオブジェクトのタイプに応じて、更新により **syndesis-operator** がトリガーされ、リソースの保存時に Fuse Online コンポーネントを再デプロイします。
7. 2 - 3 分間待ってから、**syndesis** カスタムリソースが正しく更新されていることを確認します。

- a. 以下のコマンドを実行して、**syndesis** カスタムリソースの内容を表示します。
oc describe syndesis/app
- b. 内容を確認します。
syndesis カスタムリソースに必要な設定が含まれていない場合は、これを再度編集して有効な設定を指定します。

結果

syndesis カスタムリソースのほとんどの変更によって **syndesis-operator** がトリガーされ、Fuse Online **syndesis-server** が再デプロイされます。これは、OpenShift が新しいサーバー Pod を作成することを意味します。

新しいサーバーの準備が整うまで、最大1-2分ほどかかります。準備が整うまで Fuse Online コンソールを使用することはできません。新しいサーバーのデプロイ中も、稼働中のインテグレーションは稼働し続けます。新しいデプロイメントの準備が整ったら、新しいサーバーによってインテグレーションの状態がロードされます。以下の場合に、Fuse Online コンソールを再度使用できます。

- **syndesis-server** Pod の状態が **Running** である場合。
- Fuse Online コンソールを更新すると、インテグレーション、コネクション、メッセージ、および稼働時間の値が表示される場合。

syndesis カスタムリソースの変更によっては、Fuse Online 設定が更新されても **syndesis-server** の再デプロイメントは必要ないものがあります。以下に例を示します。

- **syndesis** カスタムリソースを更新してバックアップを指定する場合、バックアップジョブが利用可能になるまで数秒かかります。

各設定変更の影響については、[インストール後に変更可能な Fuse Online の設定](#) を参照してください。

2.3. FUSE ONLINE の設定および 3SCALE での API の検出

API プロバイダーインテグレーションを作成する場合、Red Hat 3scale でそのインテグレーションの API の検出を有効することがあります。デフォルトの動作では、API は 3scale の自動検出の対象になりません。

3scale の検出を有効にしている場合には、後で任意で無効にすることができます。

2.3.1. 3scale で API の検出を有効化する Fuse Online の設定

API プロバイダーインテグレーションを作成する場合、Red Hat 3scale でそのインテグレーションの API の検出を有効することがあります。デフォルトの動作では、API は 3scale の自動検出の対象になりません。検出を有効にするには、3scale ユーザーインターフェイスの URL を指定する必要があります。

Fuse Online をインストールする前に 3scale で API の検出を有効にするよう Fuse Online を設定するには、[Fuse Online の設定に使用するカスタムリソース属性の説明](#) を参照してください。

インストール後に **syndesis** カスタムリソースを更新すると、検出を有効にすることができます。本トピックでは、この手順を説明します。検出を有効にすると、リソースの更新時に接続されている OpenShift プロジェクトのみに適用されます。

3scale サービス検出を有効にすると、以下のようになります。

- デフォルトの動作として、3scale は API プロバイダーインテグレーションをパブリッシュします。3scale が API プロバイダーインテグレーションをパブリッシュする場合は、以下のとおりです。
 - Fuse Online は、稼働している API プロバイダーインテグレーションの外部 URL を提供しません。
 - API には 3scale からのみアクセスできます。エンドポイントを公開するために 3scale の設定が必要です。詳細は、Red Hat 3scale API Management Admin Portal Guide の [Service Discovery](#) を参照してください。
- API プロバイダーインテグレーションの作成者は、そのインテグレーションの 3scale 検出を無効にすることができます。つまり、各 API プロバイダーインテグレーション作成者は、そのインテグレーションの API を検出可能にするかどうかを選択できます。

前提条件

- オンサイトで OCP に Fuse Online がインストールされている必要があります。
- **oc** クライアントツールがインストール済みであり、Fuse Online がインストールされている OCP クラスターに接続されている必要があります。
- API の検出を有効にするプロジェクトで Fuse Online をインストールするパーミッションが必要です。
- OpenShift クラスター上の 3scale プロジェクトの名前を把握している必要があります。

手順

1. Fuse Online をインストールする権限を持つアカウントで OpenShift にログインします。以下に例を示します。
oc login -u developer -p developer
2. Fuse Online が稼働している OpenShift プロジェクトに切り替えます。このプロジェクトのみで検出を有効にします。以下に例を示します。
oc project my-fuse-online-project
3. 3scale プロジェクトによる Fuse Online インテグレーションの表示を可能にするパーミッションを追加します。
oc adm policy add-cluster-role-to-user view system:serviceaccount:<3scale-project>:amp
たとえば、OpenShift の 3scale プロジェクトの名前が **my3scale** が場合は、次のようになります。
oc adm policy add-cluster-role-to-user view system:serviceaccount:my3scale:amp
4. **syndesis** カスタムリソースを編集します。
 - a. 以下のコマンドを実行します。通常、エディターでリソースが開かれます。
oc edit syndesis
 - b. 3scale ユーザーインターフェイスの URL に **managementUrlFor3scale** を設定し、リソースを編集します。結果は以下のようになります。

```
spec:
  components:
    server:
```

```
features:
  managementUrlFor3scale: https://3scale-admin.apps.mycluster.com
```

- c. リソースを保存します。
5. 任意手順:切り替え先のプロジェクトで検出が有効になっていることを確認するには、以下のコマンドを実行します。

```
oc describe dc/syndesis-server
```

検出が有効な場合、**OPENSIFT_MANAGEMENT_URL_FOR3SCALE** 環境変数がカスタムリソースで指定した URL に設定されていることがこのコマンドで出力されます。

結果

syndesis カスタムリソースに対するこの変更により、Fuse Online のインストールを担う **syndesis-operator** が **syndesis-server** を再デプロイするようトリガーされます。切り替え先の OpenShift プロジェクトでは、新しいデフォルト動作として API が 3scale で検出の対象となります。

syndesis-server DeploymentConfig オブジェクトを編集して **OPENSIFT_MANAGEMENT_URL_FOR3SCALE** 環境変数を設定しないでください。**syndesis-operator** により変更が元に戻るため、これは動作しません。**syndesis-operator** は、常に **syndesis** カスタムリソースのみに従って Fuse Online がデプロイされるようにします。

2.3.2. 3scale で API の検出を無効化する Fuse Online の設定

[API の 3scale 検出を有効化する Fuse Online の設定](#) で説明されている手順に従っている場合、各 API プロバイダーインテグレーションの作成者は、そのインテグレーションの API が検出可能かどうかを選択できます。場合によっては、インテグレーション API が 3scale では検出できないデフォルトの動作を再設定する必要になることがあります。たとえば、Fuse Online で API プロバイダーを編集およびテストする場合などです。このような場合、**managementUrlFor3scale** キーと値を指定する行がコメントアウトされるよう Fuse Online の **syndesis** カスタムリソースを更新します。これにより、3scale による検出が無効になります。その後、検出可能な API でパブリッシュされた API プロバイダーインテグレーションを再パブリッシュします。

前提条件

- オンサイトで OCP に Fuse Online がインストールされている必要があります。
- 検出を無効にするプロジェクトで Fuse Online をインストールするパーミッションが必要です。
- **oc** クライアントツールがインストール済みであり、Fuse Online がインストールされている OCP クラスターに接続されている必要があります。
- 3scale 検出は、[API の 3scale 検出を有効化する Fuse Online の設定](#) で説明されているように OpenShift プロジェクトで有効にされている必要があります

手順

1. Fuse Online をインストールする権限を持つアカウントで OpenShift にログインします。以下に例を示します。
oc login -u developer -p developer
2. 3scale の検出が有効になっている OpenShift プロジェクトに切り替えます。以下に例を示します。
oc project my-fuse-online-project

3. **syndesis** カスタムリソースを編集します。

- a. 以下のコマンドを実行します。通常、エディターでリソースが開かれます。

```
oc edit syndesis
```

- b. **managementUrlFor3scale** キーおよび値を指定する行の最初にハッシュ記号(#)を挿入して、リソースを編集します。これにより、行がコメントになり、検出が無効になります。結果は以下のようになります。

```
spec:
  components:
    server:
      features:
        # managementUrlFor3scale: https://3scale-admin.apps.mycluster.com
```

- c. リソースを保存します。

これにより、**syndesis** カスタムリソースが更新されます。更新により、**OPENSIFT_MANAGEMENT_URL_FOR3SCALE** 環境変数が削除されます。**syndesis** カスタムリソースに対するこの変更により、Fuse Online のインストールを担う **syndesis-operator** が **syndesis-server** を再デプロイするようトリガーされます。新しいデフォルト動作では、API が 3scale で検出の対象ではなくなります。

syndesis-server DeploymentConfig オブジェクトを編集して

OPENSIFT_MANAGEMENT_URL_FOR3SCALE 環境変数を設定を削除しないでください。

syndesis-operator により変更が元に戻るため、これは動作しません。**syndesis-**

operator は、常に **syndesis** カスタムリソースのみに従って Fuse Online がデプロイされるようにします。

4. 先ほど検出を無効化したプロジェクトで作成された API プロバイダーインテグレーションで、検出が有効化されていたものがあれば、そのインテグレーションを再パブリッシュします。言い換えると、API プロバイダーインテグレーションが作成されたプロジェクトの検出が有効であるときに、そのインテグレーションの API の検出を無効にした場合は、この API プロバイダーインテグレーションを再パブリッシュする必要はありません。

結果

関連する OpenShift プロジェクトで作成された API プロバイダーインテグレーションの API が 3scale で検出できなくなります。

Fuse Online が関連するプロジェクトにある API プロバイダーインテグレーションをパブリッシュ (または再パブリッシュ) する場合、Fuse Online は API プロバイダーインテグレーションの操作を呼び出すための外部 URL を提供します。

2.4. FUSE ONLINE POD の設定

Fuse Online カスタムリソースでノードのアフィニティーおよび許容オプションを設定して、Fuse Online インフラストラクチャーコンポーネントおよびインテグレーション Pod を OpenShift クラスター内のノードに配置するように設定できます。**nodeAffinity** オプションを使用すると、配置先のノードのグループに対して Fuse Online Pod のアフィニティーを指定できます。**tolerations** オプションを使用すると、Fuse Online Pod が実行されるノードを制御し、他のワークロードがそれらのノードを使用しないようにすることができます。

これらのオプションは、Fuse Online のインストール前または後に Fuse Online カスタムリソースを編集して設定します。

インテグレーションデプロイメントとは別に、インフラストラクチャーコンポーネントデプロイメントのノードアフィニティーまたは許容の設定を指定します。

- **infraScheduling** - **infraScheduling** オプションを指定して、これらの Fuse Online インフラストラクチャーデプロイメントのノードのアフィニティーおよび許容を設定します。
 - **syndesis-db**
 - **syndesis-meta**
 - **syndesis-prometheus**
 - **syndesis-ui**

注記: **syndesis-operator** デプロイメントは Fuse Online 管理コンポーネントであるため、ノードのアフィニティーまたは許容の変更による影響を受けません。
- **integrationScheduling** - すべての Fuse Online インテグレーションデプロイメントのノードのアフィニティーおよび許容を設定するには、**integrationScheduling** オプションを指定します。

注記: Fuse Online デプロイメントがインテグレーションデプロイメントかどうかを知るには、デプロイメント名を確認します。**oc get pods** コマンドを使用して、実行中のデプロイメントの一覧を表示できます。デプロイメント名に接頭辞 **-i** が含まれる場合、これはインテグレーションデプロイメントになります (例: **i-my-integration**)。

以下の手順では、**oc patch** コマンドを使用してカスタムリソースを編集する方法を説明します。この他にも、以下のトピックで説明されているような方法で、カスタムリソースを編集できます。

- Fuse Online がすでにインストールされている場合は、[Fuse Online の設定変更に関する一般的な手順](#)の手順にしたがって、カスタムリソースを開きます。
- Fuse Online がインストールされていない場合は、OCP のバージョンに応じて、以下の手順にしたがってカスタムリソースを開きます。
 - OCP 4.x の場合: [Fuse Online インスタンスを OpenShift 4.x プロジェクトに追加](#)

前提条件

- オンサイトで OCP に Fuse Online がインストールされている必要があります。
- **oc** クライアントツールがインストール済みであり、Fuse Online がインストールされている OCP クラスタに接続されている。

手順

1. 以下の構文で **oc patch** コマンドを使用して **許容 (Toleration)** を設定します。Fuse Online インフラストラクチャーコンポーネントのデプロイメントには **infraScheduling** を、Fuse Online インテグレーションのデプロイメントには **integrationScheduling** を指定します。

```
oc patch syndesis/app --type=merge -p '{"spec":{"infraScheduling":{"infraScheduling": {"tolerations":[{"key":"_value1_","operator":"_value2_","effect":"_value3_"}, {"key":"_value4_","operator":"_value5_","effect":"_value6_"}]}}}'
```

たとえば、以下のコマンドは、Fuse Online インフラストラクチャーコンポーネントに許容を追加します。

```
oc patch syndesis/app --type=merge -p '{"spec":{"infraScheduling":{"tolerations":
[{"key":"tol_1","operator":"Equal","effect":"NoSchedule"},
{"key":"tol_2","operator":"Equal","value":"value2","effect":"NoSchedule"}]}}}'
```

- 以下の構文で **oc patch** コマンドを使用して、ノードのアフィニティーを設定します。Fuse Online インフラストラクチャーコンポーネントのデプロイメントには **infraScheduling** を、Fuse Online インテグレーションのデプロイメントには **integrationScheduling** を指定します。

```
oc patch syndesis/app --type=merge -p '{"spec":{"infraScheduling|integrationScheduling":
{"affinity":{"nodeAffinity":{"preferredDuringSchedulingIgnoredDuringExecution":
[{"weight":_value1_,"preference":{"matchExpressions":
[{"key": "_value1_", "operator": "_value2_", "values": ["_value3_", "_value4_"]}]}]}}}}}'
```

たとえば、以下のコマンドは nodeAffinity を Fuse Online インフラストラクチャーコンポーネントとインテグレーションの両方に設定します。

```
oc patch syndesis/app --type=merge -p '{"spec":{"infraScheduling":{"affinity":{"nodeAffinity":
{"preferredDuringSchedulingIgnoredDuringExecution":{"weight":11,"preference":
{"matchExpressions":{"key": "_affinity", "operator": "In", "values":
["foo", "bar"]}}}}}}, "integrationScheduling":{"affinity":{"nodeAffinity":
{"preferredDuringSchedulingIgnoredDuringExecution":{"weight":22,"preference":
{"matchExpressions":{"key": "my_affinity2", "operator": "In", "values": ["foo", "bar"]}}}}}}}'
```

- 設定を削除するには、以下のコマンド例のように、**oc patch** コマンドで **remove** オプションを使用します。
 - 以下のコマンド例は、Fuse Online インテグレーションの nodeAffinity から **preferredDuringSchedulingIgnoredDuringExecution** 設定を削除します。

```
oc patch syndesis/app --type=json -p [{"op": "remove", "path":
"/spec/integrationScheduling/affinity/nodeAffinity/preferredDuringSchedulingIgnoredDuringE
xecution" }]
```

- 以下の例では、Fuse Online インフラストラクチャーコンポーネントから許容を削除します。

```
oc patch syndesis/app --type=json -p [{"op": "remove", "path":
"/spec/infraScheduling/tolerations" }]
```

その他のリソース

許容およびノードのアフィニティーに関する詳細は、OpenShift ドキュメントの [Understanding node affinity](#) および [Controlling pod placement using node taints](#) を参照してください。

2.5. HTTP プロキシの設定

多くのサービスは、HTTP プロキシを使用して、他のサービスまたはコンポーネントとの通信をフィルタリングし、セキュリティを保護します。Fuse Online コンポーネントおよびインテグレーションで HTTP プロキシを設定するには、Fuse Online のインストール前または後に Fuse Online カスタムリソースを編集します。

前提条件

- HTTP プロキシのホストおよびポートの値を知っている必要があります。

手順

1. エディターでカスタムリソースを開きます。
Fuse Online がすでにインストールされている場合は、[Fuse Online の設定変更に関する一般的な手順](#)の手順にしたがって、カスタムリソースを開きます。

Fuse Online がインストールされていない場合は、OCP のバージョンに応じて、以下の手順にしたがってカスタムリソースを開きます。

OCP 4.x の場合: [Fuse Online インスタンスを OpenShift 4.x プロジェクトに追加](#)

2. **syndesis-server** および **syndesis-meta** コンポーネントの場合は、Java Options (**javaOptions**) パラメーターを設定し、HTTP プロキシホストおよびポートの値を指定します。**syndesis-server** および **syndesis-meta** に同じ HTTP プロキシ値を使用します。以下に例を示します。

```
spec:
  components:
    server:
      javaOptions:
        -Dhttp.proxyHost=10.0.0.100 -Dhttp.proxyPort=8800

    meta:
      javaOptions:
        -Dhttp.proxyHost=10.0.0.100 -Dhttp.proxyPort=8800
```

3. Fuse Online インテグレーションでは、Maven 引数 (additionalArguments) パラメーターを設定します。以下に例を示します。

```
spec:
  components:
    server:
      features:
        maven:
          additionalArguments: "-Dhttp.proxy=10.0.0.100 -Dhttp.proxyPort=8800"
```

4. カスタムリソースを保存します。
Fuse Online がすでにインストールされている場合、手順 2 の javaOptions の更新に応じて **syndesis-server** Pod が再デプロイされます。
5. HTTP プロキシを使用する稼働中のインテグレーションを再パブリッシュします。
 - a. Fuse Online で、更新するインテグレーションを選択します。
 - b. **Edit** を選択します。
 - c. **Publish** を選択します。

Fuse Online はインテグレーションの再ビルド時に HTTP プロキシ設定を使用します。

2.6. OCP で稼働している FUSE ONLINE 環境へのサンプルデータの追加

新しいユーザーが Fuse Online でのインテグレーションの作成方法を学べるよう、Fuse Online 環境にサンプルデータを追加することができます。

- **ToDo アプリ**
- サンプル **PostgresDB** データベースと **PostgresDB** 接続
- AMQ ブローカー

Salesforce からデータベース、AMQ から REST API、および API プロバイダーのクイックスタートサンプルインテグレーションには、このサンプルデータが必要です。

デフォルトでは、AMQ ブローカー、サンプル PostgresDB データベースと接続、および ToDo アプリは無効になっています (インストールされていません)。

Fuse Online をインストールする前にサンプルデータが含まれるように Fuse Online を設定するには、[Descriptions of custom resource attributes that configure Fuse Online](#) を参照してください。

前提条件

- オンサイトで Fuse Online が OCP にインストールされ、稼働している必要があります。
- **oc** クライアントツールがインストール済みであり、Fuse Online がインストールされている OCP クラスタに接続されている。
- Fuse Online カスタムリソースを編集する権限がある。

手順

1. Fuse Online カスタムリソースを編集する権限を持つアカウントで OpenShift にログインします。以下に例を示します。

```
oc login -u developer -p developer
```

2. Fuse Online が稼働しているプロジェクトに切り替えます。以下に例を示します。

```
oc project my-fuse-online-project
```

3. **syndesis** カスタムリソースを編集します。

- a. 以下のコマンドを実行します。通常、エディターでリソースが開かれます。

```
oc edit syndesis
```

- b. **addons:todo:enabled** フィールドを **true** に変更して、カスタムリソースを編集します。

```
spec:
  addons:
    todo:
      enabled: true
```

- c. リソースを保存します。

この変更を **syndesis** カスタムリソースに保存すると、**syndesis-operator** による新しい **todo** Pod の作成と、**syndesis-server** および **syndesis-db** Pod の再デプロイがトリガーされます。

第3章 OCP 上の FUSE ONLINE の管理

オンサイトで Fuse Online を OpenShift Container Platform (OCP) にインストールしたら、Prometheus を使用してインテグレーションアクティビティを監視し、Fuse Online 環境の復元に使用できる Fuse Online の定期的なバックアップを設定できます。必要に応じて、Fuse Online のアップグレード、Fuse Online のアンインストール、または Fuse Online が含まれる OCP プロジェクトの削除を行うことができます。

詳細は以下のトピックを参照してください。

- 「Fuse Online コンポーネントの監査」
- 「Prometheus での Fuse Online インテグレーションおよびインフラストラクチャーコンポーネントの監視」
- 「Fuse Online のメータリングラベル」
- 「Fuse Online 環境のバックアップ」
- 「Fuse Online 環境の復元」
- 「Fuse Online のアップグレード」
- 「OCP プロジェクトからの Fuse Online のアンインストール」
- 「Fuse Online が含まれる OCP プロジェクトの削除」

3.1. FUSE ONLINE コンポーネントの監査



重要

Fuse Online 監査は、テクノロジープレビュー機能のみです。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、実稼働環境での使用を推奨しません。テクノロジープレビューの機能は、最新の製品機能をいち早く提供して、開発段階で機能をテストし、フィードバックを提出できるようにすることを目的としています。Red Hat のテクノロジープレビュー機能のサポート範囲に関する詳細は、<https://access.redhat.com/ja/support/offerings/techpreview> を参照してください。

Fuse Online は、ユーザーが以下の Fuse Online コンポーネントに加えた変更の基本監査をサポートします。* **接続**: Fuse Online Web コンソールのコネクターの **Details** ページに表示される **Name** およびその他のフィールド。* **Connectors - Name** フィールド。* **Integrations - Name** フィールド。

開発者がこれらのコンポーネントフィールドのいずれかを更新する場合 (インテグレーションの名前を変更するなど)、Fuse Online は ID、ユーザー、タイムスタンプ、コンポーネント (**connection**、**connector**、または **integration**) および変更のタイプ (**create**、**modify**、または **delete**) などの情報が含まれる監査メッセージを標準出力を送信します。監査メッセージのフィールド値は 30 文字に切り捨てられることに注意してください。

デフォルトでは、Fuse Online 監査は無効になっています。Fuse Online カスタムリソースを編集することで有効にできます。Fuse Online をインストールする前に監査を有効にする場合は、Fuse Online の設定に使用するカスタムリソース属性の説明を参照してください。

前提条件

- **oc** クライアントツールがインストール済みであり、Fuse Online がインストールされている OCP クラスタに接続されている。
- Fuse Online カスタムリソースを編集する権限がある。

手順

1. Fuse Online カスタムリソースを編集する権限を持つアカウントで OpenShift にログインします。以下に例を示します。

```
oc login -u admin -p admin-password
```

2. Fuse Online 環境を実行しているプロジェクトに切り替えます。以下に例を示します。

```
oc project my-fuse-online-project
```

3. `syndesis` カスタムリソースを編集します。

- a. 以下のコマンドを実行します。通常、エディターでリソースが開かれます。

```
oc edit syndesis
```

- b. 以下の行がリソースにあることを確認します。必要に応じて編集します。

```
components:
  server:
    features:
      auditing: true
```

4. リソースを保存します。

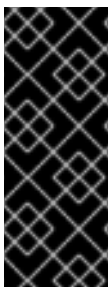
`syndesis` カスタムリソースで監査機能を有効にすると、稼働中の **syndesis-server** 設定を再読み込みし、Fuse Online は Fuse Online コンポーネントに関連する変更のロギングを開始します。

5. Fuse Online 監査ログメッセージを表示するには、次のコマンドを入力します。

```
oc logs -l syndesis.io/component=syndesis-server
```

3.2. PROMETHEUS での FUSE ONLINE インテグレーションおよびインフラストラクチャーコンポーネントの監視

Prometheus を使用して Fuse Online インフラストラクチャーコンポーネントおよび Fuse Online インテグレーションを監視できます。Grafana ダッシュボードを使用して、Prometheus によって収集されるメトリクスを可視化することもできます。



重要

Prometheus に対する Red Hat のサポートは、Red Hat 製品ドキュメントに記載されているセットアップと推奨設定に限定されます。

Grafana はコミュニティがサポートする機能です。Grafana をデプロイして Red Hat Fuse 製品を監視する設定は、Red Hat の実稼働環境におけるサービスレベルアグリーメント (SLA) の対象外です。

Fuse Online インテグレーションを監視する他に、Prometheus を使用して以下の Fuse Online インフラストラクチャーコンポーネントによって公開されるメトリクスを監視できます。

Syndesis Server

syndesis-server コンポーネントは Micrometer でインストルメント化され、デフォルトですべての JVM Micrometer メトリクスを自動的に公開します。さらに、**syndesis-server** は要求レート、エラーレート、レイテンシーなどの REST API エンドポイントに関するメトリクスを公開します。

Syndesis Meta

syndesis-meta コンポーネントは Micrometer でインストルメント化され、デフォルトですべての JVM Micrometer メトリクスを自動的に公開します。また、REST API エンドポイントについてのメトリクスも公開します。

Syndesis DB

Fuse Online Postgres データベースのメトリクスは、[サードパーティーの Prometheus エクスポート](#) を使用してエクスポートされます。

統合

integration メトリクスは、インテグレーションが作成され、デフォルトで複数の JVM メトリクスを公開する公式の JMX エクスポートを使用してエクスポートされた後に表示されます。さらに、integration メトリクスは、メッセージレートやエラーレートなどの Apache Camel 固有のメトリクスを公開します。

前提条件

- Fuse Online は、オンサイトで OCP 4.9 (またはそれ以降) にインストールされ、実行されている。
- **oc** クライアントツールがインストール済みであり、Fuse Online がインストールされている OCP クラスターに接続されている。
- OCP クラスターへの **admin** アクセスがある。
- Fuse Online インストールが **ops addon** が有効になっている状態で設定されている。必要に応じて、以下のコマンドで有効にすることができます。

```
oc patch syndesis/app --type=merge -p '{"spec": {"addons": {"ops": {"enabled": true}}}'
```

手順

1. 既存の **openshift-monitoring** 設定がある場合は、手順 2 に進みます。それ以外の場合には、ユーザーワークロードモニタリングオプションを **true** に設定する **openshift-monitoring** 設定を作成し、手順 3 に進みます。

```
oc apply -f - <<EOF
apiVersion: v1
kind: ConfigMap
metadata:
  name: cluster-monitoring-config
  namespace: openshift-monitoring
data:
  config.yaml: |
    enableUserWorkload: true
EOF
```

2. 既存の **openshift-monitoring** 設定がある場合:

- a. 既存の **openshift-monitoring** 設定をチェックして、**ユーザーワークロードモニタリング** オプションが **true** に設定されるかどうかを判別します。

```
oc get -n openshift-monitoring cm/cluster-monitoring-config -
ojsonpath='{.data.config\.yaml}'
```

結果が **enableUserWorkload: true** の場合、**ユーザーワークロードモニタリング** オプションは **true** に設定されます。ステップ 3 に進みます。

結果に他の設定が表示される場合には、次の手順に進み、ConfigMap を編集してユーザーワークロードの監視を有効にします。

- b. 以下のようにエディターで ConfigMap ファイルを開きます。

```
oc -n openshift-monitoring edit cm/cluster-monitoring-config
```

- c. **enableUserWorkload** を **true** に設定します。以下に例を示します。

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: cluster-monitoring-config
  namespace: openshift-monitoring
data:
  config.yaml: |
    enableUserWorkload: true
```

- d. ConfigMap ファイルを保存します。

3. 以下のコマンドを使用して、**openshift-user-workload-monitoring** namespace の Pod のステータスを確認します。

```
oc -n openshift-user-workload-monitoring get pods -w
```

Pod のステータスが Running になるまで待機します。以下に例を示します。

```
prometheus-operator-5d989f48fd-2qbzd 2/2 Running
prometheus-user-workload-0 5/5 Running prometheus-user-workload-1
5/5 Running
thanos-ruler-user-workload-0 3/3 Running
thanos-ruler-user-workload-1 3/3 Running
```

4. Prometheus で Fuse Online のアラートルールが有効になっていることを確認します。

- a. 内部 prometheus インスタンスにアクセスします。

```
oc port-forward -n openshift-user-workload-monitoring pod/prometheus-user-workload-0
9090
```

- b. ブラウザーを開いて **localhost:9090** にアクセスします。

- c. **Status > Targets** の順に選択します。3 つの **syndesis** エンドポイントが表示されるはずで

- d. **CTRL-C** を押して、**port-forward** プロセスを終了します。
5. OperatorHub から、Grafana Operator バージョン 4 を選択した namespace (例:**grafana-middlewre** namespace) にインストールします。更新チャンネル **v4**を使用します。
6. クラスターロールとクラスターロールのバインディングを追加して、**grafana-operator** がノードおよび namespace を一覧表示できるようにします。
 - a. **grafana-operator** Web サイトからクラスターロール YAML ファイルをダウンロードします。

```
curl https://raw.githubusercontent.com/gravana-operator/gravana-operator/v4/deploy/cluster_roles/cluster_role_grafana_operator.yaml > tmp_role.yaml
```

- b. **grafana-operator** のクラスターパーミッションを追加して、他の namespace およびノードを読み取ります。

```
cat <<EOF >> tmp_role.yaml
- apiGroups:
  - ""
  resources:
    - namespaces
    - nodes
  verbs:
    - get
    - list
    - watch
EOF
```

```
oc apply -f tmp_role.yaml
```

```
oc apply -f - <<EOF
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
  name: grafana-operator
roleRef:
  name: grafana-operator
  kind: ClusterRole
  apiGroup: ""
subjects:
  - kind: ServiceAccount
    name: grafana-operator-controller-manager
    namespace: grafana-middlewre
EOF
```

7. **DASHBOARD_NAMESPACES_ALL** 環境変数を使用して namespace を制限することで、**grafana-operator** が他の namespace から Grafana ダッシュボードを読み取れるようにします。

```
oc -n grafana-middlewre patch subs/gravana-operator --type=merge -p '{"spec":{"config":{"env":[{"name":"DASHBOARD_NAMESPACES_ALL","value":"true"}]}}}'
```

8. **grafana** Pod が再作成されていることを確認します。

```
oc -n grafana-middleware get pods -w
```

9. 必要に応じて、**grafana-operator** ログを表示します。

```
oc -n grafana-middleware logs -f `oc -n grafana-middleware get pods -oname|grep grafana-operator-controller-manager` -c manager
```

10. **Grafana カスタムリソース** を追加して、以下のように Grafana サーバー Pod を起動します。

```
oc apply -f - <<EOF
apiVersion: integreatly.org/v1alpha1
kind: Grafana
metadata:
  name: grafana-middleware
  namespace: grafana-middleware
spec:
  config:
    auth:
      disable_signout_menu: true
    auth.anonymous:
      enabled: true
    log:
      level: warn
      mode: console
    security:
      admin_password: secret
      admin_user: root
    dashboardLabelSelector:
      - matchExpressions:
        - key: app
          operator: In
          values:
            - grafana
            - syndesis
    ingress:
      enabled: true
EOF
```

11. **grafana-operator** がモニタリング情報を読み取ることを許可します。

```
oc -n grafana-middleware adm policy add-cluster-role-to-user cluster-monitoring-view -z grafana-serviceaccount
```

12. **GrafanaDataSource** を追加して、**thanos-querier** をクエリーします。

```
oc apply -f - <<EOF
apiVersion: integreatly.org/v1alpha1
kind: GrafanaDataSource
metadata:
  name: prometheus-grafanadatasource
  namespace: grafana-middleware
spec:
  datasources:
    - access: proxy
```

```

editable: true
isDefault: true
jsonData:
  httpHeaderName1: 'Authorization'
  timeInterval: 5s
  tlsSkipVerify: true
name: Prometheus
secureJsonData:
  httpHeaderValue1: "Bearer $(oc get secret $(oc get secret | grep grafana-
serviceaccount-token | awk '{print$1}')) -o=jsonpath="{.data.token}" | base64 -d)"
  type: prometheus
  url: "https://$(oc get route thanos-querier -n openshift-monitoring -
o=jsonpath="{.spec.host}")"
  name: prometheus-grafanadatasource.yaml
EOF

```

13. grafana サーバーログを表示します。

```
oc logs -f `oc get pods -l app=grafana -oname`
```

14. grafana URL にアクセスし、Fuse Online ダッシュボードを表示するには、以下を実行します。

```
echo "https://"$$(oc -n grafana-middlewre get route/grafana-route -o=jsonpath="{.spec.host}")
```

Grafana コンソールの左側のパネルで、検索ボタンをクリックします。各 Syndesis インスタンスのダッシュボードを含むフォルダー (OCP namespace 名) が表示されます。

- Fuse Online インテグレーションでは、**Integration - Camel** を選択します。このダッシュボードには、Apache Camel インテグレーションアプリケーションによって公開される標準メトリクスが表示されます。
- Fuse Online インフラストラクチャーコンポーネントでは、以下のインフラストラクチャーダッシュボードのいずれかを選択します。

Infrastructure - DB

Fuse Online の Postgres インスタンスに関連するメトリクスを表示します。

Infrastructure - JVM

syndesis-meta または **syndesis-server** アプリケーションの実行中の JVM に関するメトリクスを表示します。ダッシュボードの上部にある **Application** ドロップダウンリストから、監視するアプリケーションを選択します。

Infrastructure - REST APIs

request throughput および **latency** などの Fuse Online インフラストラクチャー API エンドポイントに関連するメトリクスを表示します。ダッシュボードの上部にある **Application** ドロップダウンリストから、監視するアプリケーションを選択します。

関連情報

Prometheus を初めて使用する場合は、https://prometheus.io/docs/prometheus/latest/getting_started/ を参照してください。

3.3. FUSE ONLINE のメータリングラベル

OpenShift の Metering Operator を使用すると、インストールされた Fuse Online の Operator とコンポーネントを分析し、Red Hat サブスクリプションに準拠しているかどうかを判断することができます。詳細は、[OpenShift ドキュメント](#) のメータリングを参照してください。

以下の表では、Fuse Online インフラストラクチャーコンポーネントおよびインテグレーションのメータリングラベルが一覧表示されています。

表3.1 Fuse Online のメータリングラベル

ラベル	使用できる値
<code>com.company</code>	<code>Red_Hat</code>
<code>rht.prod_name</code>	<code>Red_Hat_Integration</code>
<code>rht.prod_ver</code>	<code>7.8</code>
<code>rht.comp</code>	<code>Fuse</code>
<code>rht.comp_ver</code>	<code>7.8</code>
<code>rht.subcomp</code>	<code>syndesis-db</code> <code>syndesis-server</code> <code>syndesis-meta</code> <code>syndesis-ui</code> <code>syndesis-prometheus</code> <code>syndesis-operator</code> <code>i-<integration name></code>
<code>rht.subcomp_t</code>	<code>infrastructure</code> (<code>rht.subcomp = syndesis-*</code> の場合) <code>application</code> (when <code>rht.subcomp = i-*</code>)

例

- インフラストラクチャーの例 (インフラストラクチャーコンポーネントが `syndesis-db` の場合)

```
com.company: Red_Hat
rht.prod_name: Red_Hat_Integration
rht.prod_ver: 7.8
rht.comp: Fuse
rht.comp_ver: 7.8
rht.subcomp: syndesis-db
rht.subcomp_t: infrastructure
```

- アプリケーションの例 (インテグレーションのデプロイメント名が `mytestapp` の場合)

```
com.company: Red_Hat
rht.prod_name: Red_Hat_Integration
rht.prod_ver: 7.8
rht.comp: Fuse
rht.comp_ver: 7.8
rht.subcomp: i-mytestapp
rht.subcomp_t: application
```

3.4. FUSE ONLINE 環境のバックアップ

Fuse Online を定期的にバックアップするよう設定できます。

- Fuse Online がコネクションとインテグレーションを保存する内部 PostgreSQL データベース。
- **syndesis-operator** によって作成され、Fuse Online の実行に必要な OpenShift リソース。これには、設定マップ、デプロイメント設定、およびサービスアカウントが含まれますが、これらに限定されません。

Fuse Online をインストールする前に Fuse Online 環境のバックアップを設定するか、Fuse Online 環境の設定を変更してバックアップを有効することができます。

Fuse Online がバックアップを実行するように設定されている場合、Fuse Online はデータを単一のファイルに圧縮し、そのファイルを指定した Amazon S3 バケットにアップロードします。新しい Fuse Online 環境 (コネクションやインテグレーションが定義されていない) にバックアップを適用して、バックアップした Fuse Online 環境を復元できます。

前提条件

- OCP がオンサイトで稼働している必要があります。
- **oc** クライアントツールがインストール済みで、Fuse Online が稼働中または稼働予定の OCP クラスタに接続されている必要があります。
- クラスタ管理者権限を持つユーザーは、該当ユーザーがクラスタでアクセス権限を持つプロジェクトに Fuse Online をインストールするための権限を付与済みである必要があります。
- AWS アクセスキーと AWS シークレットキーが必要です。これらのクレデンシャルの取得に関する詳細は、[IAM ユーザーのアクセスキーの管理](#) を参照してください。
- アップロードする S3 バケットが存在する AWS リージョンを知っておく必要があります。
- バックアップのアップロード先の S3 バケットの名前を知っておく必要があります。

手順

1. Fuse Online をインストールする権限を持つアカウントで OpenShift にログインします。以下に例を示します。
oc login -u developer -p developer
2. バックアップを設定する Fuse Online 環境を稼働中または稼働予定の OpenShift プロジェクトに切り替えます。以下に例を示します。
oc project my-fuse-online-project
3. OpenShift シークレットを作成します。コマンドラインで以下を行います。

- 以下のコマンド形式で **syndesis-backup-s3** を指定します。
- AWS 変数を、AWS アクセスキー、AWS シークレットキー、バケットがある AWS リージョン、およびバケットの名前に置き換えます。
以下のコマンド形式を使用してシークレットを作成します。

```
oc create secret generic syndesis-backup-s3 \
  --from-literal=secret-key-id="my-aws-access-key" \
  --from-literal=secret-access-key="my-aws-secret-key" \
  --from-literal=region="aws-region" \
  --from-literal=bucket-name="aws-bucket-name"
```

このシークレットは、バックアップジョブの実行中に存在する必要があります。

4. Fuse Online がインストールされていない場合は、**default-cr.yml** ファイルを編集してバックアップを有効にする必要があります。 [Editing the syndesis custom resource before installing Fuse Online](#) を参照してください。 Fuse Online のインストール後、カスタムリソースに指定したスケジュールに応じてバックアップジョブが発生します。
Fuse Online が稼働している場合、**syndesis** カスタムリソースを編集する必要があります。

- a. 以下のコマンドを実行して、エディターで **syndesis** カスタムリソースを開きます。

```
oc edit syndesis
```

- b. 以下を **spec:** の下に追加します。

```
backup:
  schedule: my-backup-interval
```

my-backup-interval をバックアップの間隔に置き換えます。以下を参照して、バックアップの間隔を指定する方法を判断してください。

- [cron を使用して事前設定されたスケジュール](#) (英語)
- [cron で設定する間隔](#) (英語)
間隔の前に @ 記号を指定しないでください。たとえば、毎日バックアップを行うよう設定するには、カスタムリソースに以下のような記述が含まれるようにします。

```
apiVersion: syndesis.io/v1beta1
kind: Syndesis
metadata:
  name: app
spec:
  backup:
    schedule: daily
```

5. ファイルを保存します。
これにより、バックアップジョブが **syndesis-operator** に追加されます。

結果

Fuse Online がすでに稼働している場合、定義したスケジュールに応じて Fuse Online のバックアップジョブが実行されます。

次のステップ

Fuse Online をインストールする必要がある場合は、**default-cr.yml** ファイルを編集して、その他の必

要な機能を有効にしたり、他のパラメーターを設定したりします。**Default-cr.yml** ファイルにすべての設定がある場合は、OpenShift シークレットの作成時に指定したプロジェクトで Fuse Online をインストールします。

3.5. FUSE ONLINE 環境の復元

コネクションまたはインテグレーションが作成されていない新しい Fuse Online 環境で、Fuse Online 環境のバックアップを復元できます。Fuse Online 環境の復元後、復元されたコネクションを編集してパスワードを更新する必要があります。その後、復元されたインテグレーションをパブリッシュできます。

前提条件

- OCP がオンサイトで稼働している必要があります。
- **oc** クライアントツールがインストール済みであり、Fuse Online 環境を復元する OCP クラスターに接続されている必要があります。
- クラスター管理者権限を持つユーザーは、該当ユーザーがクラスターでアクセス権限を持つプロジェクトに Fuse Online をインストールするための権限を付与済みである必要があります。
- データを定期的にバックアップし、Amazon S3 にデータをアップロードするよう設定された Fuse Online 環境が必要です。
- バックアップされた Fuse Online 環境と、バックアップを復元する Fuse Online 環境で、Fuse Online のリリース番号 (例: 7.6) が同じである必要があります。
- Fuse Online バックアップが含まれる AWS バケットにアクセスする権限が必要です。
- バックアップを復元する Fuse Online 環境は新しい Fuse Online インストールである必要があります。つまり、コネクションやインテグレーションは定義されていません。Fuse Online 環境にコネクションとインテグレーションがあるプロジェクトで復元する場合、Fuse Online 環境をアンインストールし、新しい Fuse Online 環境をインストールする必要があります。

手順

1. Amazon S3 から必要なバックアップファイルをダウンロードします。詳細は、AWS ドキュメントの [S3 バケットからオブジェクトをダウンロードする方法](#) を参照してください。
2. zip ファイルの内容を展開します。たとえば、以下のコマンドラインを実行すると、**7.6-2020-03-15-23:30:00.zip** ファイルが展開され、内容が **/tmp/fuse-online-backup** フォルダーにコピーされます。
unzip 7.6-2020-03-15-23:30:00.zip -d /tmp/fuse-online-backup
3. 以下の例のように、Fuse Online データベースをデコードします。

```
base64 -d /tmp/fuse-online-backup/syndesis-db.dump > /tmp/fuse-online-backup/syndesis-db
```

4. 新しい Fuse Online 環境を稼働している OpenShift プロジェクトに切り替えます。たとえば、新しい Fuse Online 環境が **my-fuse-online-project** にある場合は、以下のコマンドを実行します。
oc project my-fuse-online-project

これ以降の手順では、新しい Fuse Online 環境が含まれるプロジェクトに切り替えたことを想定します。

5. データベース Pod の名前を取得します。
復元された Fuse Online 環境で、提供された内部 PostgreSQL データベースが使用される場合は、以下のコマンドを実行し、データベース Pod の名前を取得します。

```
oc get pods -l deploymentconfig=syndesis-db -o jsonpath='{.items[*].metadata.name}'
```

復元された Fuse Online 環境で外部データベースが使用される場合、そのデータベースの Pod の名前を取得する方法を知っている必要があります。

これ以降のコマンドでは、**DATABASE_POD_NAME** を復元された Fuse Online 環境のデータベース Pod の名前に置き換えます。

6. データベースにアクセスしているコンポーネントをスケールダウンします。
 - a. **syndesis-operator** をスケールダウンし、他のコンポーネントをスケールダウンできるようにします。

```
oc scale deployment syndesis-operator --replicas 0
```

- b. **syndesis-server** および **syndesis-meta** コンポーネントをスケールダウンします。

```
oc scale dc syndesis-server --replicas 0
```

```
oc scale dc syndesis-meta --replicas 0
```

7. データベースのバックアップファイルを Fuse Online データベース Pod に送信します。

```
oc cp /tmp/fuse-online-backup/syndesis-db DATABASE_POD_NAME:/tmp/syndesis-db
```

8. Fuse Online データベース Pod でリモートシェルセッションを開きます。

```
oc rsh DATABASE_POD_NAME
```

9. 以下のコマンドを実行して Fuse Online データベースを復元します。

psql コマンドによってデータベースのパスワードが要求され、復元された Fuse Online 環境で提供された内部 PostgreSQL データベースが使用される場合は、**syndesis-db** デプロイメント設定の **POSTGRESQL_PASSWORD** 環境変数でパスワードを確認できます。復元された Fuse Online 環境で外部データベースが使用される場合、パスワードを知っている必要があります。

```
cd /tmp
psql -c 'DROP database if exists syndesis_restore'
psql -c 'CREATE database syndesis_restore'
pg_restore -v -d syndesis_restore /tmp/syndesis-db
psql -c "SELECT pg_terminate_backend(pid) FROM pg_stat_activity WHERE datname = 'syndesis'"
psql -c 'DROP database if exists syndesis'
psql -c 'ALTER database syndesis_restore rename to syndesis'
```

Fuse Online が復元されるはずですが、RSH セッションを終了できます。

exit

10. Fuse Online コンポーネントをスケールアップします。

```
oc scale deployment syndesis-operator --replicas 1
```

syndesis-operator を 1 にスケールアップすると、スケールダウンされた他の Pod がスケールアップされます。スケールアップされない場合は手作業でスケールアップできます。

```
oc scale dc syndesis-server --replicas 1
oc scale dc syndesis-meta --replicas 1
```

11. サーバーは復元された各インテグレーションを開始しようとしませんが、最初に接続を更新する必要があります。したがって、復元されたインテグレーションが稼働していないことを確認します。
 - a. Fuse Online コンソールルートを取得します。


```
echo "https://$(oc get route/syndesis -o jsonpath='{.spec.host}')"
```
 - b. Fuse Online をインストールする権限を持つ OpenShift ユーザーアカウントで、Fuse Online コンソールにログインします。
 - c. インテグレーションのリストを表示し、すべてのインテグレーションが停止したことを確認します。稼働しているインテグレーションがある場合は停止します。
12. パスワードのある各接続を更新し、この Fuse Online 環境の正しいパスワードを適用します。以下の手順は、提供された PostgreSQL コネクションでこの作業を行う方法になります。
 - a. この復元された Fuse Online 環境が稼働しているプロジェクトの OpenShift コンソールで、PostgreSQL コネクションのパスワードを取得します。**syndesis-db** では、パスワードは環境変数にあります。
 - b. Fuse Online コンソールで接続を表示します。
 - c. PostgreSQL コネクションを編集します。
 - d. PostgreSQL コネクションのコネクション詳細で、取得したパスワードを Password フィールドに貼り付けます。
13. 各インテグレーションで **Configuration Required** が表示されないことを確認します。表示される場合は、インテグレーションを編集して問題を解決します。インテグレーションのすべてのステップが正しいければ、インテグレーションをパブリッシュします。
Build ステップの直後に、インテグレーションが **Stopped** 状態にロールバックされ続ける場合は、デプロイメントを削除して、設定が必要ないことを確認し、再度インテグレーションのパブリッシュを試行します。

ログに記録された以下のメッセージは、無視しても問題はありません。

```
Error performing GET request to https://syndesis-my-fuse-online-project.my-cluster-url/api/v1/metrics/integrations
```

3.6. FUSE ONLINE のアップグレード

時々、パッチとセキュリティ修正が組み込まれた新しいアプリケーションイメージが Fuse Online に対してリリースされます。これらの更新は、Red Hat のエラータ更新チャンネルを介して通知されます。その後、Fuse Online イメージをアップグレードできます。

OCP 4.x の場合、[Upgrading Fuse Online by using the OperatorHub](#) の手順に従って、Fuse Online 7.11 から 7.12 にアップグレードします。

Fuse Online 7.12 へのアップグレードにより、既存のインテグレーションを変更する必要があるかどうかを判断する必要があります。変更が必要ない場合でも、Fuse Online をアップグレードするときに、実行中のインテグレーションを再パブリッシュする必要があります。

3.6.1. OperatorHub (OCP 4.x) を使用した Fuse Online のアップグレード

OpenShift OperatorHub を使用して Fuse Online 7.11 から 7.12 にアップグレードします。



注記

- Fuse Online 7.12 には OpenShift Container Platform (OCP) 4.6 以降が必要です。OCP 4.5 以前を使用している場合、Fuse Online 7.12 にアップグレードするには、OCP 4.6 以降にアップグレードする必要があります。
- OCP 4.9 では、7.11 にアップグレードすると、Fuse Online Operator のアップグレードプロセス時に以下の警告が表示されます。

W1219 18:38:58.064578 1 warnings.go:70] extensions/v1beta1 Ingress is deprecated in v1.14+, unavailable in v1.22+; use networking.k8s.io/v1 Ingress

この警告は、クライアント (Fuse Online が Kubernetes/OpenShift API 初期化コードに使用する) が非推奨の Ingress バージョンにアクセスするために表示されます。この警告は、非推奨の API が完全に使用されていることを示すものではなく、Fuse Online 7.11 へアップグレードすることに問題はありません。

Fuse Online 7.11 から、または 7.12 の初期バージョンから、Fuse Online 7.12 の新しいバージョンへのアップグレードプロセスは、Fuse Online のインストール時に選択した **Approval Strategy** によって異なります。

- **Automatic** (自動) 更新の場合、新しいバージョンの Fuse Online Operator が使用できるようになると、人的な介入なしで OpenShift Operator Lifecycle Manager (OLM) によって、Fuse Online の稼働中のインスタンスが自動的にアップグレードされます。
- **Manual** (手動) 更新の場合、Operator の新しいバージョンが使用できるようになると、OLM によって更新リクエストが作成されます。クラスター管理者は、OpenShift ドキュメントの [Manually approving a pending Operator upgrade](#) セクションで説明されているように、更新リクエストを手動で承認して Fuse Online Operator を新しいバージョンに更新する必要があります。

インフラストラクチャーのアップグレード中およびアップグレード後も、既存のインテグレーションは引き続き Fuse Online ライブラリーおよび依存関係の古いバージョンで実行されます。

更新された Fuse Online バージョンで既存のインテグレーションを実行するには、インテグレーションを再パブリッシュする必要があります。

3.6.2. Fuse Online インテグレーションのアップグレード


Fuse Online 7.11 にアップグレードする場合、既存のインテグレーションに変更を加える必要があるかどうかを判別する必要があります。

[Camel Migration Considerations](#) に記載の Apache Camel の更新を確認します。

インテグレーションで変更が必要ない場合でも、インフラストラクチャーのアップグレード中およびアップグレード後に、引き続き古いバージョンの Fuse Online ライブラリーおよび依存関係で既存のインテグレーションが実行されるため、稼働中のインテグレーションを再パブリッシュする必要があります。更新されたバージョンで実行するには、再パブリッシュする必要があります。

手順

Fuse Online 環境でインテグレーションを再パブリッシュするには、以下を行います。

1. Fuse Online の左側のナビゲーションパネルで **Integrations** をクリックします。
2. インテグレーションごとに以下を行います。
 - a. インテグレーションエントリーの右側で  をクリックし、**Edit** を選択します。
 - b. Fuse Online で編集するインテグレーションが表示されたら、右上の **Publish** をクリックします。

パブリッシュを行うと、最新の Fuse Online 依存関係を使用して再ビルドが強制されます。



注記

インテグレーションのいずれかの要素に更新の必要がある新しい依存関係がある場合、Fuse Online ユーザーインターフェイスに警告が表示されます。

3.7. OCP プロジェクトからの FUSE ONLINE のアンインストール

プロジェクトやプロジェクトの内容を削除せずに、OCP プロジェクトから Fuse Online をアンインストールできます。Fuse Online のアンインストール後、稼働中のインテグレーションは継続されますが、編集または再パブリッシュできなくなります。

前提条件

- Fuse Online がインストールされている OCP プロジェクトが必要です。
- Fuse Online がインストールされている他の OpenShift プロジェクトで使用するインテグレーションをエクスポートしてある必要があります。必要な場合は、[インテグレーションのエクスポート](#) を参照してください。

手順

1. Fuse Online をインストールする権限を持つアカウントで OpenShift にログインします。以下に例を示します。
oc login -u developer -p developer
2. アンインストールする Fuse Online 環境を実行している OpenShift プロジェクトに切り替えます。以下に例を示します。
oc project my-fuse-online-project
3. Fuse Online インフラストラクチャーを削除します。
oc delete syndesis app
4. **syndesis-operator DeploymentConfig** および **ImageStream** リソースを削除します。
oc delete deployment/syndesis-operator
oc delete is/syndesis-operator

3.8. FUSE ONLINE が含まれる OCP プロジェクトの削除

Fuse Online がインストールされている OpenShift プロジェクトを削除すると、プロジェクトのすべてが削除されます。これには、定義されたすべてのインテグレーションと、稼働中のすべてのインテグレーションが含まれます。

前提条件

- Fuse Online がインストールされている OCP プロジェクトが必要です。
- Fuse Online がインストールされている他の OpenShift プロジェクトで使用するインテグレーションをエクスポートしてある必要があります。必要な場合は、[インテグレーションのエクスポート](#) を参照してください。

手順

oc delete project コマンドを実行します。たとえば、名前が **fuse-online-project** である OpenShift プロジェクトを削除するには、以下のコマンドを入力します。

oc delete project fuse-online-project

第4章 FUSE ONLINE のパブリック REST API エンドポイントの呼び出し方法

OCP で Fuse Online を稼働している場合、各 Fuse Online 環境でパブリック REST API エンドポイントを公開できます。継続的インテグレーション/継続的デリバリー (CI/CD) の外部ツールにより、これらのエンドポイントを呼び出して、Fuse Online 環境のリソースを操作することができます。

外部ツールでこれらのエンドポイントを呼び出しできるようにするには、先に各 Fuse Online 環境で、OpenShift 管理者が Fuse Online のパブリック REST API エンドポイントを公開する必要があります。API エンドポイントを呼び出すコマンドで、エンドポイントが動作する Fuse Online 環境の URL、シークレットトークン、および承認トークンを指定します。

詳細は以下のトピックを参照してください。

- [「外部ツールによって使用される Fuse Online パブリック REST API の公開」](#)
- [「Fuse Online パブリック REST API エンドポイントのベース URL の説明」](#)
- [「パブリック REST API エンドポイントを呼び出すためのシークレットトークンの取得」](#)
- [「インテグレーション ID の検索方法」](#)
- [「Fuse Online パブリックエンドポイントを呼び出す curl コマンドを指定するための形式」](#)

4.1. 外部ツールによって使用される FUSE ONLINE パブリック REST API の公開

オンサイトの OCP で Fuse Online を実行している場合、外部ツールを使用して Fuse Online 環境から別の Fuse Online 環境に Fuse Online インテグレーションをコピーすることがあります。外部 CI/CD ツールは、Jenkins ジョブ、Ansible Playbook、**cron** ベースのシェルスクリプトなどです。たとえば、Ansible Playbook ではインテグレーションを Fuse Online 開発環境からエクスポートし、Fuse Online テスト環境にインポートすることができます。

これを有効にするには、各 Fuse Online 環境の Fuse Online パブリック REST API エンドポイントを公開する必要があります。つまり、Fuse Online がインストールされている OpenShift プロジェクトごとに、この手順を繰り返す必要があります。

前提条件

- Fuse Online がインストールされている OCP プロジェクトが必要です。
- **oc** クライアントツールがインストール済みであり、Fuse Online がインストールされている OCP クラスターに接続されている必要があります。
- 外部 CI/CD ツールを使用しており、マーク付けされたインテグレーションを Fuse Online 環境から別の Fuse Online 環境にコピーする状況です。
- **syndesis-operator grant** コマンドの実行には、クラスターの管理者権限が必要です。クラスターの管理者権限はなく、OpenShift プロジェクトの管理者権限があるユーザーは、他のコマンドを実行できます。

手順

1. クラスターの管理者権限を持つアカウントで OpenShift にログインします。以下に例を示します。

oc login -u admin -p admin

クラスターの管理者権限は、**syndesis-operator grant** コマンドにのみ必要ですが、クラスター管理者がすべての手順を実行することを想定します。

2. Fuse Online が稼働している OpenShift プロジェクトに切り替えます。このプロジェクトでのみパブリック API を公開します。以下に例を示します。

oc project fuse-online-north

3. ロールを **syndesis-public-oauthproxy** サービスアカウントに付与します。OpenShift プロジェクトでは、このサービスアカウントを使用して OAuth プロキシサービスを実行します。クラスターの管理権限を持つユーザーアカウントを指定します。以下に例を示します。

syndesis-operator grant --user developer

このコマンドは、クラスターロールとクラスターのロールバインディングを作成し、これを **syndesis-public-oauthproxy** サービスアカウントに割り当てます。**ClusterRoleBinding** は、API を公開する OpenShift プロジェクトの名前を指定します。この例では、名前は **fuse-online-north** になります。

- ClusterRole: **syndesis-auth-delegator**
- ClusterRoleBinding: **syndesis-fuse-online-north-auth-delegator**

4. クライアントサービスアカウントを作成し、Fuse Online のパブリック API にアクセスする権限を付与します。
 - a. 選択した名前で OpenShift サービスアカウントを作成します。たとえば、次のコマンドを実行すると **cidc-client** サービスアカウントが作成されます。

oc create serviceaccount cidc-client

パブリック API エンドポイントを呼び出すコマンドは、このアカウントを使用してパブリック API にアクセスします。また、このサービスアカウントは、API エンドポイントへの呼び出しで指定する必要があるシークレットトークンを取得するために必要です。この説明は [パブリック REST API エンドポイントを呼び出すためのシークレットトークンの取得](#) を参照してください。

- b. Fuse Online のパブリック API にアクセスするための権限をクライアントサービスアカウントに付与します。Fuse Online がインストールされた OpenShift プロジェクトの名前が **fuse-online-north** で、API サービスにアクセスするために作成したサービスアカウントの名前が **cidc-client** である場合、以下のコマンドを実行します。

```
oc policy add-role-to-user edit system:serviceaccount:fuse-online-north:cidc-client`
```

5. **syndesis** カスタムリソースを編集します。

- a. 以下のコマンドを実行します。通常、エディターでリソースが開かれます。

oc edit syndesis

- b. リソースを編集して、パブリック API を有効にし、**routeHostname** の設定としてエンドポイントを呼び出すための Fuse Online 環境のパブリックアドレスを指定します。Fuse Online のインストールに使用された **default-cr.yml** ファイルで、パブリック API が有効化され、エンドポイントのルートホスト名が指定されている場合は、リソースを編集する必要はありません。

クラスターのセットアップにより、指定する必要があるパブリックアドレスが決定されます。詳細は、OpenShift ドキュメントの [Routes](#) を参照してください。以下の例では、ルートのホスト名は minishift クラスターに対して有効です。

-

```
spec:
  addons:
    publicApi:
      enabled: true
      routeHostname: public-syndesis.192.168.64.63.nip.io
```

- c. リソースを保存するか、そのまま閉じます。

syndesis カスタムリソースを保存することにより、Fuse Online のインストールを担う **syndesis-operator** がパブリック API OAuth プロキシサービスをデプロイするようトリガーされます。

syndesis カスタムリソースを編集する必要がなければ、パブリック API OAuth プロキシサービスはすでにデプロイされています。これは、Fuse Online のインストールに使用された **default-cr.yml** ファイルで **publicApi** が有効化され、そのルートが指定されたためです。

結果

OpenShift で、以下を確認できるようになります。

- **syndesis-public-oauthproxy** デプロイメント設定の Pod。
- **syndesis-public-oauthproxy** サービス。
- **syndesis-public-api** ルート。

この手順を実行した OpenShift プロジェクトでは、外部 CI/CD ツールで Fuse Online のパブリック REST API エンドポイントを使用して Fuse Online インテグレーションをエクスポートまたはインポートできます。

次のステップ

- 他の Fuse Online 環境にエクスポートするインテグレーションをマーク付けします。Fuse Online コンソールでインテグレーションをマーク付けしたり、インテグレーションをマーク付けする Fuse Online のパブリック REST API エンドポイントを呼び出すことができます。
- パブリック API エンドポイントが利用できることを確認するには、環境名を返す **curl** コマンドを実行します。Endpoint for obtaining a list of environment labels を参照してください。
- Fuse Online のパブリック REST API エンドポイントを呼び出すよう外部ツールを設定します。

4.2. FUSE ONLINE パブリック REST API エンドポイントのベース URL の説明

Fuse Online パブリック REST API エンドポイントのベース URL は、以下のようになります。

https://public-syndesis.192.168.64.42.nip.io/api/v1/public

Fuse Online 環境ごとに、ベース URL の最初の部分は異なります。OpenShift アプリケーションを作成して、これによりパブリック REST API エンドポイントへのアクセスを有効にする Fuse Online パブリック OAuth プロキシを実行する場合、Fuse Online 環境のパブリックアドレスを指定します。このアドレスは、Fuse Online 環境で動作するエンドポイントを呼び出すベース URL の最初の部分です。以下に例を示します。

https://public-syndesis.192.168.64.42.nip.io

ベース URL の 2 つ目の部分は、すべての Fuse Online 環境で同じです。

/api/v1/public

Fuse Online のパブリック REST API は、3 つのリソースで動作するエンドポイントを提供します。

- **/integration** は、ベース URL で識別される Fuse Online 環境にあるインテグレーションです。
- **/connections** は、ベース URL で識別される Fuse Online 環境にあるコネクションです。
- **/environments** は、ベース URL で識別される Fuse Online 環境にある環境ラベルのセットです。

4.3. パブリック REST API エンドポイントを呼び出すためのシークレットトークンの取得

Fuse Online のパブリック REST API エンドポイントを呼び出すコマンドで、シークレットトークンを指定する必要があります。このトークンは、任意の Fuse Online 環境で Fuse Online パブリック REST API を公開したときに作成したサービスアカウントのものであります。

前提条件

- オンサイトの OCP で Fuse Online が稼働している必要があります。
- エンドポイントを呼び出す Fuse Online 環境によって提供されるパブリック REST API を公開している必要があります。

手順

1. この Fuse Online 環境のパブリック REST API を公開したときに作成したサービスアカウントのシークレットトークンの名前を取得します。たとえば、**cicd-client** がサービスアカウントの名前である場合、以下のコマンドを実行します。

```
oc describe serviceaccount cicd-client
```

これにより、以下のような 2 つのトークンの名前が含まれる、**cicd-client** サービスアカウントの情報の一覧が表示されます。

```
Tokens:  cicd-client-token-gxb25  
         cicd-client-token-gxdnv
```

2. いずれかのトークンの内容を表示します。以下に例を示します。

```
oc describe secret cicd-client-token-gxb25
```

これにより、**token:** とそれに続く長いランダムな文字を表示する **Data** セクションなど、情報の一覧が表示されます。これはサービスアカウントの 2 つのシークレットトークンの 1 つです。

3. シークレットトークンをコピーし、ファイルに貼り付けて保存します。

結果

curl コマンドで指定するシークレットトークンは次のようになります。

```
-H 'Authorization: Bearer  
eyJhbGciOiJSUzI1NiIsImtpZCI6Ij9.eyJpc3MiOiJrdWJlcm5ldGVzL3NlcnZpY2VhY2NvdW50Iiwia3ViZXJ
```

```
uZXRlcy5pby9zZXJ2aWNIYWVudC9uYW1lc3BhY2UiOiJzeW5kZXNpcylsmt1YmVybmV0ZXMuaV
8vc2VydmljZWJY291bnQvc2VjcmV0Lm5hbWUuOiJzeW5kZXNpcy1jZC1jbGllbnQtdG9rZW4tMnZjNmwi
LCJrdWJlcm5ldGVzLmlvL3NlcnZpY2VhY2NvdW50L3NlcnZpY2UtYWNjb3VudC5uYW1lIjoic3luZGVzaX
MtY2QtY2xpZW50liwia3ViZXJlcy5pby9zZXJ2aWNIYWVudC9zZXJ2aWNLWFJY291bnQudW
kljoiNjUxMjYxNGMtMmYwMS0xMWU5LTk3OWEtNDI1YWNIMzY3MTcyliwic3Viljoic3lzdGVtOnNlcnZpY
2VhY2NvdW50OnN5bmRlc2l2OnN5bmRlc2l2LWNkLWNsaWVudCJ9.uKsri0JSKJDbgHoQwAhBJSNuV
KwJgjegf2QlrCkxVssSK1zIMZQaF9P5a4R7ZcWRnrZ_345UTqxYVeRlfHWVH0PqBkD-
n7PAS9dcKJIFdS1jUHOmL1FTGgc3YW-bz1SIWT93tvK1EhorZ4_-
EBfXhSAP4Uumi5qAg3_QUTMDstq233NSwBKytFOw3Pp1ys3p3y0hcaiLMimeCH60vR4iWvptqqzc5Q
DigHiPySZNWxs_5awZlwdolDvR-nSj690aC-
49UKFgyEEzdzFU4bl2W4hOyDyhN9fValAZQKeJUrJBU-
InFTHI_NAd2OwzOEBpWZuj31Za5w9fU4kf4UDGA'
```

次のステップ

保存されたファイルから、指定の Fuse Online 環境のパブリック REST API エンドポイントを呼び出すコマンドにトークンをコピーします。

4.4. インテグレーション ID の検索方法

特定のインテグレーションのみで動作する Fuse Online のパブリック REST API エンドポイントを呼び出すコマンドで、エンドポイントが操作するインテグレーションの ID を指定する必要があります。以下のいずれかを指定します。

- インテグレーションの名前
これは、Fuse Online コンソールで表示されているとおりに正確に指定する必要があります (例: **timer-to-log**)。インテグレーション名にスペースまたは特殊文字がある場合は、HTML エスケープ文字を指定する必要があります。
- 内部インテグレーション ID
この ID は、インテグレーションの概要を表示するときに Fuse Online コンソール URL に含まれます。インテグレーションの概要を表示するには、左側のナビゲーションパネルで **Integrations** をクリックします。インテグレーションのリストで、ID が必要なインテグレーションのエントリーをクリックします。

ブラウザでインテグレーションの概要が表示され、URL の末尾に **/integrations/i-Lauq5ShznJ4LcuWwiwcz** のような内容が表示されます。このインテグレーションの ID は **i-Lauq5ShznJ4LcuWwiwcz** です。

4.5. FUSE ONLINE パブリックエンドポイントを呼び出す curl コマンドを指定するための形式

Fuse Online のパブリック REST API エンドポイントを呼び出す **curl** コマンドの形式は、以下のようになります。

```
curl [options] \  
-H "Content-Type: <media-type>" \  
-H "SYNDESIS-XSRF-TOKEN: awesome" \  
-H `Authorization: Bearer <token>` \  
<base-url><endpoint> \  
[--request <HTTP-method>] \  
[-d <data>] \  
[-o <filename>]
```

表4.1 curl コマンドオプション

オプション	説明
[options]	選択した curl オプションを指定します。
<media-type>	エクスポートおよびインポートエンドポイントに、 multipart/form-data を指定します。他のすべてのエンドポイントに、 application/json を指定します。
<token>	パブリック REST API を公開するときに作成した OpenShift サービスアカウントのシークレットトークンを指定します。
<base-url>	エンドポイントが操作するインテグレーション、コネクション、または環境ラベルのある Fuse Online 環境のベース URL を指定します。
<endpoint>	呼び出すエンドポイントを指定します。
[--request <HTTP-method>]	任意で、HTTP メソッドを指定します (例: --request POST)。
[-d <data>]	任意で、呼び出されるエンドポイントに応じて、エンドポイントが必要とする引数を指定します。たとえば、インテグレーションの環境ラベルを test に変更するには、 -d `test` を指定します。
[-o <filename>]	任意で、出力が含まれるファイルの名前を指定する必要がある場合には、 curl の -o オプションとファイル名を指定します (例: -o export.zip)。

以下の **curl** コマンドは、指定する1つまたは複数の環境に対してインテグレーションをマーク付けする Fuse Online のパブリック API エンドポイントを呼び出します。

```
curl -v -k -L -H "Content-Type: application/json" -H "SYNDESIS-XSRF-TOKEN: awesome" -H
'Authorization: Bearer
eyJhbGciOiJIUzI1NiIsInR5cGU6IiwiOiJrdWJlcm5ldGVzL3NlcnZpY2VhY2NvdW50liwia3ViZXJ2
uZXRlcy5pby9zZXJ2aWVudC9uYW1lc3BhY2UiOiJzeW5kZXNpcylsmt1YmVybWV0ZXMuVW
8vc2VydmVudC9uYW1lc3BhY2UiOiJzeW5kZXNpcy1jZC1jbGllbnQtdG9rZW4tMnZjNmwi
LCJrdWJlcm5ldGVzLmlvL3NlcnZpY2VhY2NvdW50L3NlcnZpY2UtYWNjb3VudC5uYW1ljoic3luZGVzaX
MtY2QtY2xpZW50liwia3ViZXJ2aWVudC9uYW1lc3BhY2UiOiJzeW5kZXNpcy1jZC1jbGllbnQtdG9rZW4tMnZjNmwi
kljoiNjUxMjYxNGMtMmYwMS0xMWU5LTk3OWEtNDI1YWNIMzY3MTcyliwic3ViIjoic3lzdGVtOnNlcnZpY
2VhY2NvdW50OnN5bmRlc2l2OnN5bmRlc2l2LWNkLWNsaWVudCj9uKsri0JJSKJDbgHoQwAhBJSNuV
KwJgJegf2QlrCkxVssSK1zIMZQaF9P5a4R7ZcWRnrZ_345UTqxYVeRlfHWVH0PqBkD-
n7PAS9dcKJIFdS1jUHOmL1FTGgc3YW-bz1SIWT93tvK1EhorZ4_-
EBfXhSAP4Uumi5qAg3_QUTMDstq233NSwBKytFOW3Pp1ys3p3y0hcaiLMimeCH60vR4iWvptqqzc5Q
DigHiPySZNWxs_5awZlwdolDvR-nSj690aC-
49UKFgyEEEdzdFU4bl2W4hOyDyhN9fValAZQKeJUrJBU-
lnFTHI_NAd2OwzOEBpWZuj31Za5w9fU4kf4UDGA'
https://public-syndesis.192.168.64.42.nip.io/api/v1/public/integrations/timer-to-log/tags -d
["test","staging"] --request PUT
```

サンプル **curl** コマンドでは、以下のとおりです。

- コマンドの最後にある URL は、呼び出すエンドポイントの Fuse Online 環境を識別します。

- **timer-to-log** は、指定した環境の **timer-to-log** インテグレーションをマーク付けすることを示します。
- **test** および **staging** を指定すると、これらの環境の **timer-to-log** インテグレーションがマーク付けされます。

第5章 外部ツールを使用した CI/CD の FUSE ONLINE インテグレーションのエクスポート/インポート

オンサイトの OCP で Fuse Online を稼働している場合、CI/CD (継続的インテグレーション/継続的デプロイメント) パイプラインを特定のインテグレーションで実行することがあります。これを実装するには、以下のタスクを完了する必要があります。

- Fuse Online コンソールで、インテグレーションに CI/CD パイプラインをマーク付けします。
- OpenShift で、[Fuse Online のパブリック API を公開](#) します。
- 外部ツールを使用して、インテグレーションをエクスポートおよびインポートする Fuse Online パブリック API エンドポイントを呼び出します。

詳細は以下のトピックを参照してください。

- [「CI/CD のインテグレーションのマーク付け」](#)
- [「CI/CD のインテグレーションのマーク付け」](#)
- [「CI/CD の環境ラベルの管理」](#)
- [「Fuse Online パブリック API エクスポートエンドポイントの呼び出し」](#)
- [「Fuse Online パブリック API インポートエンドポイントの呼び出し」](#)

5.1. CI/CD のインテグレーションのマーク付け

オンサイトの OCP で Fuse Online を稼働している場合、パイプラインのインテグレーションを特定するために、指定する CI/CD 環境のインテグレーションをマーク付けします。これにより、タイムスタンプ付きのラベルがインテグレーションに適用されます。バックエンド CI/CD Fuse Online のパブリック API は、ラベルおよびそのタイムスタンプを使用してインテグレーションをフィルタリングし、パイプラインが動作する必要のあるインテグレーションを見つけます。

たとえば、Fuse Online で **test1** 環境のインテグレーションをマーク付けするとします。その後、Fuse Online のパブリック API エクスポートエンドポイントを呼び出して、**test1** 環境ラベルがあるインテグレーションをエクスポートできます。エンドポイントによって **test1** インテグレーションがエクスポートファイルにパッケージ化され、そのファイルが返されます。**test1** インテグレーションを Fuse Online テスト環境にコピーするには、パブリック API インポートエンドポイントを呼び出し、エクスポートした **test1** インテグレーションが含まれるファイルを提供します。

ここで、以前 **test1** 環境に対してマーク付けしたインテグレーションを繰り返し更新し、パブリッシュするとします。インテグレーションが新しいバージョンになり、更新されたバージョンを **test1** 環境にエクスポートするとします。すでにインテグレーションが **test1** 環境に対してマーク付けされている場合でも、再度マーク付けする必要があります。インテグレーションを再度マーク付けすると、**test1** 環境ラベルのタイムスタンプが更新されます。これは、インテグレーションが更新され、エクスポートできるようになったことを外部ツールに示します。

インテグレーションを再度マーク付けする、つまりインテグレーションにすでに割り当てられた環境ラベルのタイムスタンプを更新するには、[CI/CD のインテグレーションのマーク付け](#) の手順にしたがいます。CI/CD ダイアログで、対象の環境のチェックボックスがすでに選択されているはずなので、**Save** をクリックします。


5.2. CI/CD のインテグレーションのマーク付け

オンサイトの OCP で Fuse Online を稼働している場合、パイプラインのインテグレーションを特定するために、指定する CI/CD 環境のインテグレーションをマーク付けします。これにより、ラベルがインテグレーションに適用されます。

前提条件

Fuse Online がインストールされている OCP プロジェクトが必要です。

手順

1. 左側の Fuse Online ナビゲーションパネルで **Integrations** をクリックします。
2. インテグレーションのリストで、マーク付けするインテグレーションのエントリーの右にある  をクリックしてポップアップメニューを表示し、**Manage CI/CD** をクリックします。
3. 表示されるダイアログで、以下のいずれかを行います。
 - このインテグレーションをマーク付けする環境が表示されている場合、その環境の左側にあるチェックボックスを選択します。
 - このインテグレーションをマーク付けする環境がすでに選択されている場合は、その環境を選択したままにしておきます。
 - 環境のラベルを作成する必要がある場合、**Manage CI/CD** をクリックすると、新しい環境ラベルを作成するダイアログが表示されます。必要なラベルを作成してから、この手順に戻り、インテグレーションをマーク付けします。

Fuse Online によって、選択した環境ラベルがインテグレーションに適用されます。任意の数の環境ラベルをインテグレーションに適用できます。

4. **Save** をクリックします。

結果

Fuse Online は、選択した環境のリリースのインテグレーションにラベルを付けます。バックエンド API はインテグレーションをフィルタリングして、たとえば **test1** ラベルが付いたインテグレーションなどの検索を行うことができます。

次のステップ

パイプラインが特定の環境に対してマーク付けされたインテグレーションで動作できるようにするには、先に Fuse Online のパブリック API エンドポイントを公開する必要があります。[外部ツールによって使用される Fuse Online パブリック REST API の公開](#) を参照してください。

5.3. CI/CD の環境ラベルの管理

Fuse Online コンソールで CI/CD 環境ラベルを作成、変更、または削除できます。CI/CD 環境ラベルは、CI/CD パイプラインのインテグレーションに適用できる識別用のタグです。

前提条件

- オンサイトの OCP で Fuse Online を稼働している必要があります。
- 外部ツールでは、Fuse Online 環境ラベルを使用して CI/CD パイプラインのインテグレーションが識別される必要があります。

手順

1. Fuse Online の左側のナビゲーションパネルで **Integrations** をクリックします。
2. 右上の **Manage CI/CD** をクリックします。
3. **Manage CI/CD** ページで以下を行います。
 - CI/CD 環境ラベルを作成するには、以下を実行します。
 - a. **Add New** をクリックします。
 - b. **Add Tag Name** ダイアログで、**Tag Name** 入力フィールドに新しい環境ラベルの名前を入力します (例: test1)。
 - c. **Save** をクリックします。 **Manage CI/CD** ページに、追加した環境ラベルの新しいエントリーがあります。
 - CI/CD 環境ラベルを変更するには、以下を実行します。
 - a. CI/CD 環境ラベルの一覧内の、変更するラベルのエントリーで、**Edit** をクリックします。
 - b. **Edit Tag** ダイアログで、必要に応じて名前を変更します。
 - c. **Save** をクリックすると、環境ラベルのリストに更新された名前が表示されます。Fuse Online は、変更した環境ラベルですでにマーク付けされているすべてのインテグレーションに対し、更新された名前を適用します。
 - CI/CD 環境ラベルを削除するには、以下を実行します。
 - a. CI/CD 環境ラベルの一覧内の、削除するラベルのエントリーで、**Remove** をクリックします。
 - b. 確認ダイアログで、**Yes** をクリックします。Fuse Online は環境ラベルを削除し、削除される環境ラベルでマーク付けされたインテグレーションからもラベルを削除します。

次のステップ

新しい環境ラベルでインテグレーションをマーク付けします。[CI/CD のインテグレーションのマーク付け](#) を参照してください。

5.4. FUSE ONLINE パブリック API エクスポートエンドポイントの呼び出し

外部ツールを使用して Fuse Online インテグレーションを別の Fuse Online 環境からコピーできるようにするには、先に以下のタスクを実行する必要があります。

- Fuse Online では、CI/CD パイプラインのためにエクスポートするインテグレーションを特定の環境に対してマーク付けする必要があります。[CI/CD のインテグレーションのマーク付け](#) を参照してください。
この要件には例外があります。1回のエクスポート操作で Fuse Online 環境からすべてのインテグレーションをエクスポートする場合、特定の環境に対してインテグレーションがすでにマーク付けされているかどうかは重要ではありません。
- Fuse Online のパブリック API が公開済みである必要があります。[外部ツールが使用する Fuse Online のパブリック REST API の公開](#) を参照してください。

特定環境のインテグレーションをエクスポートするためのエンドポイント

特定の環境に対してマーク付けされたインテグレーションをエクスポートするために、Fuse Online は以下の **GET** メソッドエンドポイントを提供します。

/public/integrations/{env}/export.zip

{env} を、すでに作成した CI/CD 環境ラベルに置き換えてください。インテグレーションが特定の環境に対してマーク付けされている場合、Fuse Online はマーク付けされたタイミングを示すタイムスタンプで管理します。エンドポイントがマーク付けされてからエクスポートされていない場合に限り、インテグレーションをエクスポートするのがデフォルトの挙動です。たとえば、**test1** 環境に対してマーク付けされたインテグレーションをエクスポートする場合、エンドポイントは以下のようになります。

/public/integrations/test1/export.zip

このエンドポイントは、**test1** 環境ラベルがあり、以下の条件のいずれかを満たす、各インテグレーションをエクスポートします。

- インテグレーションは、最後にエクスポートされた後に **test1** 環境に対してマーク付けされた。
- インテグレーションは **test1** 環境に対してマーク付けされ、エクスポートされていない。

エンドポイントはインテグレーションを **export.zip** ファイルにパッケージ化し、そのファイルを返します。

インテグレーションがエクスポートされた後に指定の環境に対してマーク付けされたインテグレーションがない場合は、エンドポイントは HTTP **204** 応答を返し、返すものがないことを示します。

指定された環境に対してタグ付けされたすべてのインテグレーションをエクスポートするためのエンドポイント

エクスポートエンドポイントを呼び出して、指定する環境ラベルでタグ付けされた Fuse Online 環境のすべてのインテグレーションをエクスポートすることができます。エンドポイントは、インテグレーションが最後にエクスポートされた後に指定の環境に対してマーク付けされたかどうかをインテグレーションにエクスポートします。これにより、特定のタグがあるすべてのインテグレーションを別の Fuse Online 環境で簡単に複製できます。これを行うには、以下を実行します。

- 環境ラベルがすでに作成されている必要があります。[CI/CD のインテグレーションのマーク付け](#) を参照してください。
- エクスポートエンドポイントの呼び出しに **ignoreTimestamp=true** オプションを追加します。

以下に例を示します。

/public/integrations/test1/export.zip?ignoreTimestamp=true -o export.zip

すべてのインテグレーションをエクスポートするためのエンドポイント

エクスポートエンドポイントを呼び出して、単一の Fuse Online 環境ですべてのインテグレーションをエクスポートすることができます。これにより、別の Fuse Online 環境ですべてのインテグレーションを簡単に複製できます。これを行うには、以下を実行します。

- 環境ラベルがすでに作成されている必要があります。[CI/CD のインテグレーションのマーク付け](#) を参照してください。
- エクスポートエンドポイントの呼び出しに **all=true** オプションを追加します。

以下に例を示します。

```
/public/integrations/{env}/export.zip?all=true
```

{env} を CI/CD 環境ラベルに置き換えます。エンドポイントは、その環境に対してマーク付けされていない各インテグレーションに、指定した環境ラベルを割り当てます。

all=true オプションを追加する場合、エクスポートされたインテグレーションを **export.zip** ファイルにパッケージ化するように明示的に指定する必要があります。以下に例を示します。

```
/public/integrations/test1/export.zip?all=true -o export.zip
```

このエンドポイントの呼び出しによって、以下のことが実行されます。

- **test1** 環境の各インテグレーションがマーク付けされます。
- **export.zip** ファイルのすべてのインテグレーションを返します。
-o **export.zip** オプションを指定しない場合、エンドポイントは **export.zip?all=true** という名前のファイルを返します。

エクスポートエンドポイントに必要なカスタムヘッダー

エクスポートエンドポイントを呼び出すコマンドでは、以下のカスタムヘッダーを正確に指定する必要があります。

- **-H "Content-Type: multipart/form-data"**
- **-H "SYNDESIS-XSRF-TOKEN: awesome"**
Fuse Online のパブリック API では、リクエストの認証にこのヘッダーが必要です。
- **-H 'Authorization: Bearer <token>'**
<token> を、Fuse Online のパブリック API の公開に使用される OpenShift サービスアカウントの作成時にファイルにコピーしたシークレットトークンに置き換えます。

インテグレーションをエクスポートする curl コマンドの例

以下は、インテグレーションをエクスポートする Fuse Online API エンドポイントを呼び出す **curl** コマンドの例です。

```
curl -v -k -L -H "Content-Type: multipart/form-data" -H "SYNDESIS-XSRF-TOKEN: awesome" -H
'Authorization: Bearer
eyJhbGciOiJSUzI1NiIsImtpZCI6IiJ9.eyJpc3MiOiJrdWJlcm5ldGVzL3NlcnZpY2VhY2NvdW50Iiwia3ViZXJ
uZXRlcy5pby9zZXJ2aWVudC9uYW1lc3BhY2UiOiJzeW5kZXNpcyIsImt1YmVybmV0ZXMuaW
8vc2VydmljZWJyY291bnQvc2VjcmV0Lm5hbWUiOiJzeW5kZXNpcy1jZC1jbGllbnQtdG9rZW4tMnZjNmwi
LCJrdWJlcm5ldGVzLmlvL3NlcnZpY2VhY2NvdW50L3NlcnZpY2UtYWVudC5uYW1lIjoic3luZGVzaX
MtY2QtY2xpZW50Iiwia3ViZXJ2aWVudC9uYW1lc3BhY2UiOiJzeW5kZXNpcy1jZC1jbGllbnQtdG9rZW4tMnZjNmwi
kljoiNjUxMjYxNGMtMmYwMS0xMWU5LTk3OWEtNDI1YWNIMzY3MTcyliwic3ViIjoic3lzdGVtOnNlcnZpY
2VhY2NvdW50OnN5bmRlc2l2OnN5bmRlc2l2LWNkLWNsaWVudCj9uKsri0JSKJDbgHoQwAhBJSNuW
KwJggeg2QlrCkxhVssSK1zIMZQaF9P5a4R7ZcWRnrZ_345UTqxYVeRlfHWVH0PqBkD-
n7PAS9dcKJIFdS1jUHOmL1FTGgc3YW-bz1SIWT93tvK1EhorZ4_-
EBfXhSAP4Uumi5qAg3_QUTMDstq233NSwBKytFOW3Pp1ys3p3y0hcaiLMimeCH60vR4iWvptqqzc5Q
DigHiPySZNWxs_5awZlwdolDvR-nSj690aC-
49UKFgyEEzdzFU4bl2W4hOyDyhN9fValAZQKeJUrbJU-
lnFTHI_NAd2OwzOEBpWZuj31Za5w9fU4kf4UDGA'
https://public-syndesis.192.168.64.42.nip.io/api/v1/public/integrations/dev1/export.zip
```

コマンドでは、以下のようになります。


```
EBfXhSAP4Uumi5qAg3_QUTMDstq233NSwBKYtFOw3Pp1ys3p3y0hcaiLMimeCH60vR4iWvptqqzc5Q  
DigHiPySZNWxs_5awZlwdolDvR-nSj690aC-  
49UKFgyEEdzdFU4bl2W4hOyDyhN9fValAZQKeJUrJBU-  
InFTHI_NAd2OwzOEBpWZuj31Za5w9fU4kf4UDGA'  
https://public-syndesis.192.168.64.45.nip.io/api/v1/public/integrations -F data=@export.zip -F  
environment=testing --request POST
```

このコマンドでは、以下のようになります。

- コマンドの最後にある URL は、インテグレーションをインポートする Fuse Online 環境を識別します。
- **export.zip** ファイルには、インポートされるインテグレーションが含まれます。
- **environment=testing** を指定すると、エンドポイントによって、インポートされた各インテグレーションが **testing** 環境に対してマーク付けされます。
- このコマンドにより **multipart/form-data** が作成されます。

第6章 FUSE ONLINE のパブリック REST API エンドポイントの参照情報

このセクションでは、Fuse Online の各パブリック REST API エンドポイントについての参照情報を紹介します。詳細は、[Fuse Online パブリック REST API エンドポイントの呼び出し方法](#) を参照してください。

パブリック REST API エンドポイントを定義する OpenAPI ドキュメントは <https://<fuse-online-host>/api/v1/openapi.json> の Fuse Online 環境にて利用できます。ただし、本ドキュメントでは **tags** オブジェクトに3つのタグ (**public-api**、**extensions**、**integration-support**) があると定義しています。Fuse Online の OpenShift パブリック OAuth プロキシプロセスの使用時は、**public-api** タグのみにアクセスできます。他の2つのタグは無視する必要があります。

- 「インテグレーションの状態を取得するためのエンドポイント」
- 「インテグレーションの環境ラベルの一覧を取得するためのエンドポイント」
- 「インテグレーションをマーク付けし、未指定のタグを維持するためのエンドポイント」
- 「インテグレーションをマーク付けし、未指定のタグを削除するためのエンドポイント」
- 「インテグレーションをパブリッシュするためのエンドポイント」
- 「インテグレーションを停止するためのエンドポイント」
- 「インテグレーションをエクスポートするためのエンドポイント」
- 「インテグレーションをインポートするためのエンドポイント」
- 「特定のインテグレーションから環境ラベルを削除するためのエンドポイント」
- 「環境ラベルを作成するためのエンドポイント」
- 「環境ラベルの一覧を取得するためのエンドポイント」
- 「環境ラベルを変更するためのエンドポイント」
- 「すべてのインテグレーションから環境ラベルを削除するためのエンドポイント」
- 「接続のプロパティを変更するためのエンドポイント」

6.1. インテグレーションの状態を取得するためのエンドポイント

このエンドポイントは、指定されたインテグレーションの状態を返します。状態には **Running**、**Stopped**、**Pending**、または **Error** があります。

メソッドとエンドポイント

GET

`/public/integrations/{id}/state`

表6.1パラメーター

名前	タイプ	説明
必須のヘッダーパラメーター		
-H "Content-Type: <media-type>"	application/json	エンドポイントが必要とするメディアタイプ。
-H "SYNDESIS-XSRF-TOKEN: awesome"	カスタム	認証に必要です。
-H 'Authorization: Bearer <token> '	カスタム	Fuse Online のパブリック REST API を公開する OpenShift サービスアカウントのシークレットトークン。 パブリック REST API エンドポイントを呼び出すためのシークレットトークンの取得 を参照してください。
パスパラメーター		
{id}	文字列	必須のパスパラメーター。状態を取得するインテグレーションの名前または内部 ID。 インテグレーション ID の検索方法 を参照してください。

リクエストの例

以下の例では、エンドポイントは **timer-to-log** インテグレーションの状態を返します。

/public/integrations/timer-to-log/state

生成されるもの

application/json

応答の例

```
{
  "currentState": "Unpublished",
  "stateDetails": {
    "id": "i-Lc0JLrsUFtBJfR_ylfEz:5",
    "integrationId": "i-Lc0JLrsUFtBJfR_ylfEz",
    "deploymentVersion": 5,
    "detailedState": {
      "value": "BUILDING",
      "currentStep": 2,
      "totalSteps": 4,
      "namespace": "syndesis",
      "podName": "i-timer-to-log-5-build",
      "linkType": "LOGS"
    }
  }
}
```

6.2. インテグレーションの環境ラベルの一覧を取得するためのエンドポイント

このエンドポイントは、指定のインテグレーションに適用されている環境ラベル (タグ) を返します。

メソッドとエンドポイント

GET

/public/integrations/{id}/tags

表6.2 パラメーター

名前	タイプ	説明
必須のヘッダーパラメーター		
-H "Content-Type: <media-type>"	application/json	エンドポイントが必要とするメディアタイプ。
-H "SYNDESIS-XSRF-TOKEN: awesome"	カスタム	認証に必要です。
-H 'Authorization: Bearer <token> '	カスタム	Fuse Online のパブリック REST API を公開する OpenShift サービスアカウントのシークレットトークン。 パブリック REST API エンドポイントを呼び出すためのシークレットトークンの取得 を参照してください。
パスパラメーター		
{id}	文字列	必須のパスパラメーター。環境ラベルを取得するインテグレーションの名前または内部 ID。 インテグレーション ID の検索方法

リクエストの例

以下の例では、エンドポイントは **timer-to-log** インテグレーションの環境ラベルを返します。

```
/public/integrations/timer-to-log/tags
```

生成されるもの

```
application/json
```

応答の例

```
{"test":{"name":"test","releaseTag":"i-Lc5Wl16UFtBjFR_ylggz","lastTaggedAt":1554887553159,"lastExportedAt":1554887330152,"lastImportedAt":1554888047271},"staging":{"name":"staging","releaseTag":"i-Lc5Wl16UFtBjFR_ylgfz","lastTaggedAt":1554887553159}}
```

6.3. インテグレーションをマーク付けし、未指定のタグを維持するためのエンドポイント

このエンドポイントは **PATCH** メソッドを使用して、指定された環境の指定のインテグレーションをマーク付けします。インテグレーションが指定の環境に対してすでにマーク付けされている場合、エンドポイントはその環境ラベルのタイムスタンプを更新します。インテグレーションが新しいリクエストで指定されていない環境に対してマーク付けされていた場合、エンドポイントはそのタグをそのまま残し、タイムスタンプを更新しません。

この **PATCH** エンドポイントは、他の既存のタグを削除しなくてもタグが追加されるため、CI/CD ツールに便利です。これは、指定環境に対してインテグレーションをマーク付けし、リクエストで指定されていない環境のタグを削除する、**PUT** エンドポイントとは対比的です。

メソッドとエンドポイント

PATCH

/public/integrations/{id}/tags

表6.3 パラメーター

名前	タイプ	説明
必須のヘッダーパラメーター		
-H "Content-Type: <media-type>"	application/json	エンドポイントが必要とするメディアタイプ。
-H "SYNDESIS-XSRF-TOKEN: awesome"	カスタム	認証に必要です。
-H 'Authorization: Bearer <token> '	カスタム	Fuse Online のパブリック REST API を公開する OpenShift サービスアカウントのシークレットトークン。 パブリック REST API エンドポイントを呼び出すためのシークレットトークンの取得 を参照してください。
パスパラメーター		
{id}	文字列	必須。指定の環境に対してマーク付けするインテグレーションの名前または内部 ID。 インテグレーション ID の検索方法 を参照してください。
追加のパラメーター		
--request PATCH		PATCH メソッドを指定します。
-d [env{,...}]	文字列	必須。指定のインテグレーションに追加する1つ以上の環境ラベルをコンマ区切りで指定します。環境ラベルがすでに作成されている必要があります。エンドポイントはラベルを作成できません。 CI/CD のインテグレーションのマーク付け を参照してください。

リクエストの例

以下の例では、エンドポイントは **test2** および **test3** 環境に対して **timer-to-log** インテグレーションをマーク付けします。

```
public/integrations/timer-to-log/tags --request PATCH -d ["test2","test3"]
```

生成されるもの

application/json

応答の例

```
{
  "test2": {
    "name": "test2",
    "releaseTag": "i-LcXydouUFtBJfR_ylgrz",
    "lastTaggedAt": 1555365010746
  },
  "test3": {
    "name": "test3",
    "releaseTag": "i-LcXydouUFtBJfR_ylgsz",
    "lastTaggedAt": 1555365010746
  },
  "test": {
    "name": "test",
    "releaseTag": "i-Lc5Wl16UFtBJfR_ylggz",
    "lastTaggedAt": 1554887553159,
    "lastExportedAt": 1554887330152,
    "lastImportedAt": 1554888047271,
    "staging": {
      "name": "staging",
      "releaseTag": "i-Lc5Wl16UFtBJfR_ylgfz",
      "lastTaggedAt": 1554887553159
    }
  }
}
```

6.4. インテグレーションをマーク付けし、未指定のタグを削除するためのエンドポイント

このエンドポイントは **PUT** メソッドを使用して、指定された環境の指定されたインテグレーションをマーク付けします。インテグレーションが新しいリクエストで指定されていない環境に対してマーク付けされている場合、エンドポイントはその環境ラベルをインテグレーションから削除します。

未指定の環境ラベルを削除せずにインテグレーションをマーク付けするには、代わりに **PATCH** メソッドエンドポイント を呼び出します。

メソッドとエンドポイント

PUT

/public/integrations/{id}/tags

表6.4 パラメーター

名前	タイプ	説明
必須のヘッダーパラメーター		
-H "Content-Type: <media-type>"	application/json	エンドポイントが必要とするメディアタイプ。
-H "SYNDESIS-XSRF-TOKEN: awesome"	カスタム	認証に必要です。
-H 'Authorization: Bearer <token> '	カスタム	Fuse Online のパブリック REST API を公開する OpenShift サービスアカウントのシークレットトークン。 パブリック REST API エンドポイントを呼び出すためのシークレットトークンの取得 を参照してください。
パスパラメーター		
{id}	文字列	必須。マーク付けするインテグレーションの名前または内部 ID。 インテグレーション ID の検索方法 を参照してください。

名前	タイプ	説明
追加のパラメーター		
<code>--request PUT</code>		PUT メソッドを指定します。
<code>-d [env{,...}]</code>	文字列	必須。コンマ区切りの環境ラベルを1つ以上指定します。エンドポイントは、これらの環境の指定したインテグレーションをマーク付けします。環境ラベルがすでに作成されている必要があります。エンドポイントはラベルを作成できません。 CI/CD のインテグレーションのマーク付け を参照してください。

リクエストの例

以下の例では、エンドポイントは `test2` および `test3` 環境に対して `timer-to-log` インテグレーションをマーク付けします。インテグレーションが他の環境に対してマーク付けされていた場合、エンドポイントはこれらのタグをインテグレーションから削除します。

```
public/integrations/timer-to-log/tags --request PUT -d ['test2','test3']
```

生成されるもの

```
application/json
```

応答の例

```
{"test2":{"name":"test2","releaseTag":"i-LcXyw7GUFtBjFR_ylgtz","lastTaggedAt":1555365085713},"test3":{"name":"test3","releaseTag":"i-LcXyw7GUFtBjFR_ylguz","lastTaggedAt":1555365085713}}
```

6.5. インテグレーションをパブリッシュするためのエンドポイント

このエンドポイントは指定されたインテグレーションをパブリッシュします。インテグレーションがすでに稼働中である場合、エンドポイントはインテグレーションを停止し、再度パブリッシュします。

メソッドとエンドポイント

POST

```
/public/integrations/{id}/deployments
```

表6.5 パラメーター

名前	タイプ	説明
必須のヘッダーパラメーター		
<code>-H "Content-Type: <media-type>"</code>	application/json	エンドポイントが必要とするメディアタイプ。

名前	タイプ	説明
-H "SYNDEISIS-XSRF-TOKEN: awesome"	カスタム	認証に必要です。
-H 'Authorization: Bearer <token> '	カスタム	Fuse Online のパブリック REST API を公開する OpenShift サービスアカウントのシークレットトークン。 パブリック REST API エンドポイントを呼び出すためのシークレットトークンの取得 を参照してください。
パスパラメーター		
{id}	文字列	必須。パブリッシュするインテグレーションの名前または内部 ID。 インテグレーション ID の検索方法 を参照してください。

リクエストの例

以下の例では、エンドポイントは **timer-to-log** インテグレーションをパブリッシュします。

/public/integrations/timer-to-log/deployments

生成されるもの

application/json

応答の例

この例では、省略記号は一部の応答の省略を示しています。

```
{
  "id": "i-Lc0JLrsUFtBJfR_ylfEz:8",
  "version": 8,
  "createdAt": 1555365135324,
  "updatedAt": 1555365135324,
  "userId": "system:serviceaccount:syndesis:syndesis-cd-client",
  "currentState": "Pending",
  "targetState": "Published",
  "integrationId": "i-Lc0JLrsUFtBJfR_ylfEz",
  ".2c+PC9zdmc+",
  "description": "Trigger events based on an interval or a cron expression",
  "isDerived": false,
  "stepKind": "endpoint",
  "id": "i-Lc0I7wqEVfKCDDHC8Jv",
  "configuredProperties": {
    "bodyLoggingEnabled": true,
    "contextLoggingEnabled": true
  },
  "metadata": {
    "configured": true,
    "stepKind": "log",
    "name": "Log"
  },
  "continuousDeliveryState": {
    "test2": {
      "name": "test2",
      "releaseTag": "i-LcXyw7GUFtBJfR_ylguz",
      "lastTaggedAt": 1555365085713
    },
    "test3": {
      "name": "test3",
      "releaseTag": "i-LcXyw7GUFtBJfR_ylguz",
      "lastTaggedAt": 1555365085713
    }
  }
}
```

6.6. インテグレーションを停止するためのエンドポイント

このエンドポイントは、指定されたインテグレーションを停止します。

メソッドとエンドポイント

PUT

`/public/integrations/{id}/deployments/stop`

表6.6 パラメーター

名前	タイプ	説明
必須のヘッダーパラメーター		
<code>-H "Content-Type: <media-type>"</code>	application/json	エンドポイントが必要とするメディアタイプ。
<code>-H "SYNDESIS-XSRF-TOKEN: awesome"</code>	カスタム	認証に必要です。
<code>-H 'Authorization: Bearer <token> '</code>	カスタム	Fuse Online のパブリック REST API を公開する OpenShift サービスアカウントのシークレットトークン。 パブリック REST API エンドポイントを呼び出すためのシークレットトークンの取得 を参照してください。
パスパラメーター		
<code>{id}</code>	文字列	必須。停止するインテグレーションの名前または内部 ID。 インテグレーション ID の検索方法 を参照してください。

リクエストの例

以下の例では、エンドポイントは `timer-to-log` インテグレーションを停止します。

```
/public/integrations/timer-to-log/deployments stop
```

生成されるもの

```
application/json
```

応答の例

204 ステータスコードのコンテンツがない

6.7. インテグレーションをエクスポートするためのエンドポイント

このエンドポイントはインテグレーションをエクスポートします。エンドポイントが指定の環境に対してマーク付けされ、まだ1回もエクスポートされていないインテグレーションか、その環境に対して最後にマーク付けされた後にエクスポートされていないインテグレーションをエクスポートするのがデフォルトの挙動です。以下のいずれかを行う場合は、オプションを指定できます。

- Fuse Online 環境のすべてのインテグレーションをエクスポートし、指定の環境ラベルでタグ付けする。
- インテグレーションが最後にエクスポートされた後に、更新またはタグ付けされたかどうかに関わらず、指定した環境ラベルでタグ付けされたすべてのインテグレーションをエクスポートします。

Fuse Online パブリック API エクスポートエンドポイントの呼び出し も参照してください。

メソッドとエンドポイント

GET

/public/integrations/{env}/export.zip

表6.7 パラメーター

名前	タイプ	説明
必須のヘッダーパラメーター		
-H "Content-Type: <media-type>"	multipart/form-data	エンドポイントが必要とするメディアタイプ。
-H "SYNDEESIS-XSRF-TOKEN: awesome"	カスタム	認証に必要です。
-H 'Authorization: Bearer <token> '	カスタム	Fuse Online のパブリック REST API を公開する OpenShift サービスアカウントのシークレットトークン。 パブリック REST API エンドポイントを呼び出すためのシークレットトークンの取得 を参照してください。
パスパラメーター		
{env}	文字列	必須。Fuse Online コンソールで作成した環境ラベル。エンドポイントは、この環境に対してマーク付けされたインテグレーションをエクスポートします。
クエリーパラメーター		
all=true	文字列	任意手順:Fuse Online 環境にあるすべてのインテグレーションをエクスポートする場合に、このオプションを指定します。エンドポイントは、各インテグレーションの現在のバージョンをエクスポートします。指定した環境に対してインテグレーションがマーク付けされていない場合は、エンドポイントによって指定の環境ラベルがインテグレーションに追加されます。環境ラベルがすでに作成されている必要があります。エンドポイントはラベルを作成できません。

名前	タイプ	説明
ignoreTimestamp=true	文字列	指定した環境ラベルでタグ付けされたすべてのインテグレーションをエクスポートします。インテグレーションが最後にエクスポートされた後にインテグレーションが編集およびタグ付けされたかどうかに関わらず、この操作を行います。
追加のパラメーター		
-o export.zip	文字列	クエリーパラメーターを指定した場合は必須です。このオプションを指定しない場合、エクスポートされたインテグレーションは export.zip?all=true という名前のファイルに含まれます。

リクエストの例

最初の例では、エンドポイントは、**test1** 環境に対してマーク付けされ、まだ1回もエクスポートされていないインテグレーションか、その環境に対して最後にマーク付けされた後にエクスポートされていないインテグレーションをエクスポートします。

/public/integrations/test1/export.zip

以下の例では、エンドポイントは **test1** 環境に対してマーク付けされたすべてのインテグレーションをエクスポートします。前回エクスポートされた後にインテグレーションが更新されたかどうかは関係ありません。

/public/integrations/test1/export.zip?ignoreTimestamp=true -o export.zip

最後の例では、エンドポイントは各インテグレーションが **test1** 環境に対してマーク付けされていることを確かめ、**export.zip** ファイルのすべてのインテグレーションを返します。

/public/integrations/test1/export.zip?all=true -o export.zip

生成されるもの

application/octet-stream

応答

エクスポートされたインテグレーションが含まれる **export.zip** ファイル。エクスポートするインテグレーションがない場合、エンドポイントは **204** の HTTP ステータスを返します。

6.8. インテグレーションをインポートするためのエンドポイント

このエンドポイントは、提供されたファイルにあるインテグレーションをインポートします。Fuse Online [パブリック REST API インポートエンドポイントの呼び出し](#) も参照してください。

メソッドとエンドポイント

POST

/public/integrations

表6.8 パラメーター

名前	タイプ	説明
必須のヘッダーパラメーター		
-H "Content-Type: <media-type>"	multipart/form-data	エンドポイントが必要とするメディアタイプ。
-H "SYNDESIS-XSRF-TOKEN: awesome"	カスタム	認証に必要です。
-H 'Authorization: Bearer <token> '	カスタム	Fuse Online のパブリック REST API を公開する OpenShift サービスアカウントのシークレットトークン。 パブリック REST API エンドポイントを呼び出すためのシークレットトークンの取得 を参照してください。
追加のパラメーター		
data=@export.zip	文字列	必須。これは、インポートするインテグレーションが含まれるファイルです。このファイルを取得するには、これまでにエクスポートエンドポイントを呼び出したことがある必要があります。
environment={env}	文字列	必須。インポートされた各インテグレーションに追加する環境ラベル。環境ラベルがすでに作成されている必要があります。エンドポイントはラベルを作成できません。

リクエストの例

以下の例では、エンドポイントは **export.zip** ファイルにあるインテグレーションをインポートし、**testing** 環境に対してインテグレーションをマーク付けします。

```
/public/integrations -F data=@export.zip -F environment=testing --request POST
```

生成されるもの

```
multipart/form-data
```

応答の例

応答は、インテグレーションおよびコネクションなどのインポートされたリソースの一覧です。以下の例では、省略記号は応答の一部が省略されていることを示しています。

```
{"lastImportedAt":1554888047271,"results":[{"id":"i-Lc0JLrsUFtBJfR_yIfEz","version":5,"createdAt":1554800274935,"updatedAt":0,"tags":["timer"],"name":"timer-to-log","flows":[{"id":"-Lc0I5AZEvfKCDDHC8Jv","steps":[{"id":"-
```



```
Lc0I5jnEVfKCDDHC8Jv","configuredProperties":{"period":"900000"},"metadata":...
"description":"Trigger events based on an interval or a cron
expression","isDerived":false},"stepKind":"endpoint"},"{"id":"-
Lc0I7wqEVfKCDDHC8Jv","configuredProperties":
{"bodyLoggingEnabled":"true","contextLoggingEnabled":"true"},"metadata":
{"configured":"true"},"stepKind":"log","name":"Log"}]],"continuousDeliveryState":{"staging":
{"name":"staging","releaseTag":"i-
Lc5Wl16UFtBjR_ylgfz","lastTaggedAt":1554887553159},"test":{"name":"test","releaseTag":"i-
Lc5Wl16UFtBjR_ylggz","lastTaggedAt":1554887553159,"lastExportedAt":1554887330152,"lastIm
portedAt":1554887859824}}]]}}
```

6.9. 特定のインテグレーションから環境ラベルを削除するためのエンドポイント

このエンドポイントは、指定された環境ラベルを指定のインテグレーションから削除します。環境ラベル自体は引き続き存在しますが、指定されたインテグレーションのマーク付けが解除されます。

メソッドとエンドポイント

DELETE

/public/integrations/{id}/tags/{env}

表6.9 パラメーター

名前	タイプ	説明
必須のヘッダーパラメーター		
-H "Content-Type: <media-type>"	application/json	エンドポイントが必要とするメディアタイプ。
-H "SYNDESIS-XSRF-TOKEN: awesome"	カスタム	認証に必要です。
-H 'Authorization: Bearer <token> '	カスタム	Fuse Online のパブリック REST API を公開する OpenShift サービスアカウントのシークレットトークン。 パブリック REST API エンドポイントを呼び出すためのシークレットトークンの取得 を参照してください。
パスパラメーター		
{id}	文字列	必須のパスパラメーター。マーク付けを解除するインテグレーションの名前または内部 ID。 インテグレーション ID の検索方法 を参照してください。
{env}	文字列	必須のパスパラメーター。指定したインテグレーションから削除する環境ラベル。

名前	タイプ	説明
追加のパラメーター		
<code>--request DELETE</code>		DELETE メソッドを指定します。

リクエストの例

以下の例では、エンドポイントは **dev1** 環境ラベルを **timer-to-log** インテグレーションから削除します。

```
/public/integrations/timer-to-log/tags/dev1 --request DELETE
```

応答の例

204 ステータスコードのコンテンツがない

6.10. 環境ラベルを作成するためのエンドポイント

このエンドポイントは環境ラベルを作成します。Fuse Online コンソールで、またはパブリック API エンドポイントを呼び出して明示的にマーク付けするまで、インテグレーションがこのラベルでマークが付けられることはありません。

メソッドとエンドポイント

POST

```
/public/environments/{env}
```

表6.10 パラメーター

名前	タイプ	説明
必須のヘッダーパラメーター		
<code>-H "Content-Type: <media-type>"</code>	application/json	エンドポイントが必要とするメディアタイプ。
<code>-H "SYNDESIS-XSRF-TOKEN: awesome"</code>	カスタム	認証に必要です。
<code>-H 'Authorization: Bearer <token> '</code>	カスタム	Fuse Online のパブリック REST API を公開する OpenShift サービスアカウントのシークレットトークン。 パブリック REST API エンドポイントを呼び出すためのシークレットトークンの取得 を参照してください。
パスパラメーター		
<code>{env}</code>	文字列	必須。作成する環境ラベル。

生成されるもの

何も生成しませんが、応答は以下のようになります。

応答	説明
204 No Content	成功。
400 Bad Request	指定した環境名がすでに存在します。
5xx エラー	サーバーはリクエストに対応できません。

リクエストの例

以下の例では、エンドポイントが **dev1** 環境ラベルを作成します。

/public/environments/dev1 --request POST

このエンドポイントを呼び出した Fuse Online 環境に、**dev1** 環境ラベルが存在するようになります。このラベルでマーク付けされたインテグレーションはまだありません。

応答の例

応答ステータスコード **204** のコンテンツがない

6.11. 環境ラベルの一覧を取得するためのエンドポイント

このエンドポイントは、Fuse Online 環境に存在する環境ラベルの一覧を返します。Fuse Online コンソールで環境ラベルを作成します。[CI/CD の環境ラベルの管理](#) を参照してください。

メソッドとエンドポイント

GET

/public/environments[?withUses=true]

表6.11 パラメーター

名前	タイプ	説明
必須のヘッダーパラメーター		
-H "Content-Type: <media-type>"	application/json	エンドポイントが必要とするメディアタイプ。
-H "SYNDESIS-XSRF-TOKEN: awesome"	カスタム	認証に必要です。

名前	タイプ	説明
-H 'Authorization: Bearer <token> '	カスタム	Fuse Online のパブリック REST API を公開する OpenShift サービスアカウントのシークレットトークン。 パブリック REST API エンドポイントを呼び出すためのシークレットトークンの取得 を参照してください。
パスパラメーター		
withUses	ブール値	任意手順: コマンドが withUses=true を指定している場合、エンドポイントは環境ラベルと、そのラベルでマーク付けされたインテグレーションの数を返します。指定されていない場合、または false に設定されている場合、エンドポイントは環境ラベルのみを返します。

生成されるもの

application/json

リクエストの例

- `/public/environments`
 応答の例:
`["env-label-1", "env-label-2"]`
- `/public/environments/?withUses=true`
 応答の例:
`[{"name": "env-label-1", "uses": 1}, {"name": "env-label-2", "uses": 0}]`

6.12. 環境ラベルを変更するためのエンドポイント

このエンドポイントは環境ラベルを変更します。元の環境ラベルについてマーク付けされていたインテグレーションは、新しい環境ラベルに対してマーク付けされるようになります。

メソッドとエンドポイント

PUT

`/public/environments/{env}`

表6.12 パラメーター

名前	タイプ	説明
必須のヘッダーパラメーター		

名前	タイプ	説明
-H "Content-Type: <media-type>"	application/json	エンドポイントが必要とするメディアタイプ。
-H "SYNDESIS-XSRF-TOKEN: awesome"	カスタム	認証が必要です。
-H 'Authorization: Bearer <token> '	カスタム	Fuse Online のパブリック REST API を公開する OpenShift サービスアカウントのシークレットトークン。 パブリック REST API エンドポイントを呼び出すためのシークレットトークンの取得 を参照してください。
パスパラメーター		
{env}	文字列	必須。変更する環境ラベル。
追加のパラメーター		
-d {env}	文字列	必須。環境の新しいラベルを指定します。環境ラベルがすでに作成されている必要があります。エンドポイントはラベルを作成できません。
--request PUT		PUT メソッドを指定します。

リクエストの例

以下の例では、エンドポイントが **dev1** 環境ラベルを **dev2** に変更します。

```
/public/environments/dev1 -d 'dev2' --request PUT
```

dev1 環境に対してマーク付けされていたインテグレーションのタグがなくなります。このインテグレーションは、**dev2** 環境に対してマーク付けされるようになります。

応答の例

応答ステータスコード **204** のコンテンツがない

6.13. すべてのインテグレーションから環境ラベルを削除するためのエンドポイント

このエンドポイントは、指定された環境ラベルが適用されている各インテグレーションからラベルを削除します。

メソッドとエンドポイント

DELETE

```
/public/environments/{env}
```

表6.13 パラメーター

名前	タイプ	説明
必須のヘッダーパラメーター		
-H "Content-Type: <media-type>"	application/json	エンドポイントが必要とするメディアタイプ。
-H "SYNDESIS-XSRF-TOKEN: awesome"	カスタム	認証に必要です。
-H 'Authorization: Bearer <token> '	カスタム	Fuse Online のパブリック REST API を公開する OpenShift サービスアカウントのシークレットトークン。 パブリック REST API エンドポイントを呼び出すためのシークレットトークンの取得 を参照してください。
パスパラメーター		
{env}	文字列	必須。インテグレーションから削除する環境ラベル。
追加のパラメーター		
--request DELETE		DELETE メソッドを指定します。

リクエストの例

以下の例では、エンドポイントは **dev1** タグを持つインテグレーションからそれを削除します。

```
/public/environments/dev1 --request DELETE
```

応答の例

応答ステータスコード **204** のコンテンツがない

6.14. コネクションのプロパティを変更するためのエンドポイント

このエンドポイントは、指定されたコネクションのプロパティを変更します。これは、設定が必要なコネクションがあるインテグレーションをインポートした後に役立ちます。たとえば、コネクションが使用するクレデンシャルを変更する必要がある場合などが該当します。

メソッドとエンドポイント

POST

```
/public/connections/{id}/properties
```

表6.14 パラメーター

名前	タイプ	説明
必須のヘッダーパラメーター		
-H "Content-Type: <media-type>"	application/json	エンドポイントが必要とするメディアタイプ。
-H "SYNDESIS-XSRF-TOKEN: awesome"	カスタム	認証に必要です。
-H 'Authorization: Bearer <token> '	カスタム	Fuse Online のパブリック REST API を公開する OpenShift サービスアカウントのシークレットトークン。 パブリック REST API エンドポイントを呼び出すためのシークレットトークンの取得 を参照してください。
パスパラメーター		
{id}	文字列	<p>必須。{id} を、プロパティを変更する接続の ID に置き換えます。接続 ID については、以下のいずれかを指定します。</p> <ul style="list-style-type: none"> ● 接続の名前 (例: PostgresDB)。接続名にスペースまたは特殊文字がある場合は、HTML エスケープ文字を指定する必要があります。 ● 接続の詳細を表示したときの Fuse Online コンソール URL の内部接続 ID。接続の詳細を表示するには、左側のナビゲーションパネルで Connections をクリックします。次に、詳細を表示する接続をクリックします。ブラウザーに接続の詳細が表示されると、URL の末尾に /connections/i-Laupl8XznJ4LcuWwiwaz のような内容が表示されます。この接続の ID は i-Laupl8XznJ4LcuWwiwaz です。

リクエストの例

以下の例では、**PostgresDB** 接続のプロパティを変更します。**user** プロパティの新しい値は **myuser** で、**password** プロパティの新しい値は **mypassword** です。

```
/public/connections/PostgresDB/properties --request POST -d '{ "user": "myuser", "password": "mypassword" }'
```

生成されるもの

application/json

応答の例

この例では、省略記号は応答の大部分が省略されていることを表しています。

```
{ "uses": 0, "id": "i-LaOziUGpQE45nua4pfCz", "name": "TODO app", "configuredProperties":  
  { "password": "»ENC:c2cb731046372a275b76beabc92aefa061f79b43fb791fb599d9e85ec0235a7e",  
    "basePath": "/api", "host": "http://todo-syndesis.my-minishift.syndesis.io/", "specification": ...  
    "userId": "admin", "lastUpdated": 1555365796629, "createdDate": 1553066813379, "board": { "id": "i-  
    Lbj4-vqUFtBJfR_ylfCz", "metadata": { "connector-id": "i-LaOzcPZpQE45nua4pfBz", "connector-  
    version-latest": "1", "connector-version-connection": "1" }, "messages":  
    [ { "level": "WARN", "code": "SYNDESI007" }, { "createdAt": 1554494263030, "updatedAt": 15544942637  
    27, "targetResourceId": "i-  
    LaOziUGpQE45nua4pfCz", "notices": 0, "warnings": 1, "errors": 0, "isDerived": false } }
```


第7章 FUSE ONLINE ユーザーインターフェイスのカスタマイズ

Fuse Online のアプリケーション名、ロゴ、アイコン、およびその他のユーザーインターフェイスの詳細を置き換えて、Fuse Online をカスタマイズできます。これは、**syndesis-ui** デプロイメントファイルの **ConfigMap** オブジェクトとしてカスタマイズされたリソースをマウントして行います。

前提条件

- お客様が管理する環境のオンサイトの OpenShift Container Platform に Fuse Online がインストールされ、稼働中である必要があります。
- **oc** クライアントツールは、Fuse Online がインストールされている OCP クラスタに接続されている必要があります。
- ユーザーが Fuse Online をインストールできる権限を持っている必要があります。

手順

1. 以下のコマンドを実行して、Fuse Online がインストールされている **syndesis-ui** Pod の名前を取得します。

```
oc get pods
```

2. 以下のコマンドを実行して、カスタマイズ可能なロゴ、アイコン、フォント、および背景を取得します。その際、**SYNDESIS_UI_POD_NAME** を **syndesis-ui** を実行している Pod の実際の名前に置き換えます。

```
oc rsync SYNDESIS_UI_POD_NAME:/opt/app-root/src/static/media ~/
```

これにより、~/**media/** フォルダーにユーザーインターフェイスリソースが追加されます。

3. ヘッダーロゴが含まれるファイル ~/**media/syndesis_logo_full_darkbkg.9c467d96.svg** を編集し、保存します。
4. 更新されたロゴファイルが含まれる **ConfigMap** オブジェクトを作成して、変更したファイルをクラスタにアップロードします。

```
cd ~/
oc create configmap syndesis-ui-custom-logo --from-file=media/syndesis_logo_full_darkbkg.9c467d96.svg
```

5. **syndesis-ui** デプロイメントファイルを変更して、**syndesis-ui-custom-logo ConfigMap** オブジェクトをマウントします。これには、**oc edit dc/syndesis-ui** を呼び出すか、OpenShift Web コンソールを使用して **Applications > Deployments > syndesis-ui > Actions > Edit YAML** と選択します。

- a. **spec.template.containers.volumeMounts** の下に以下を追加します。

```
-name: syndesis-ui-custom-logo-volume
  mountPath: /opt/app-root/src/static/media/syndesis_logo_full_darkbkg.9c467d96.svg
  subPath: syndesis_logo_full_darkbkg.9c467d96.svg
```

- b. **spec.template.volumes** の下に以下を追加します。

```
-name: syndesis-ui-custom-logo-volume
  configMap:
    name: syndesis-ui-custom-logo
```

- c. **syndesis-ui** デプロイメントファイルを保存します。
6. `~/media` フォルダにあるカスタマイズしたい各ファイルについて、ステップ 3 から 5 を繰り返します。
7. アプリケーションタイトルが含まれるファイルを編集および保存します。

- a. ファイルを更新するディレクトリを作成します。

```
mkdir ~/syndesis-custom/src
```

- b. 以下のコマンドを実行して、アプリケーションタイトルを取得します。その際、**SYNDESIIS_UI_POD_NAME** は **syndesis-ui** を実行している Pod の実際の名前に置き換えます。

```
oc rsync SYNDESIIS_UI_POD_NAME:/opt/app-root/src/index.html ~/syndesis-custom/src
```

- c. アプリケーションタイトルを置き換えます。以下に例を示します。

```
sed -i 's/<title>.*</title>/<title>NEW_TITLE</title>/g' ~/syndesis-custom/src/index.html
```

8. 更新されたファイルが含まれる **ConfigMap** オブジェクトを作成して、変更したアプリケーションタイトルファイルをクラスターにアップロードします。

```
cd ~/syndesis-custom/src
oc create configmap syndesis-ui-custom-index --from-file=index.html
```

9. **syndesis-ui-custom-index ConfigMap** オブジェクトをマウントして、**syndesis-ui** デプロイメントファイルを変更します。

- a. **spec.template.containers.volumeMounts** の下に以下を追加します。

```
-name: syndesis-ui-custom-vindex-olume
  mountPath: /opt/app-root/src/index.html
  subPath: index.html
```

- b. **spec.template.volumes** の下に以下を追加します。

```
-name: syndesis-ui-custom-index-volume
  configMap:
    name: syndesis-ui-custom-index
```

- c. **syndesis-ui** デプロイメントファイルを保存します。

結果

syndesis-ui デプロイメントファイルを保存すると、変更内容を反映したユーザーインターフェイスで Fuse Online が自動的に再デプロイされます。Fuse Online の後続の再デプロイメントでも、変更内容が反映されます。