

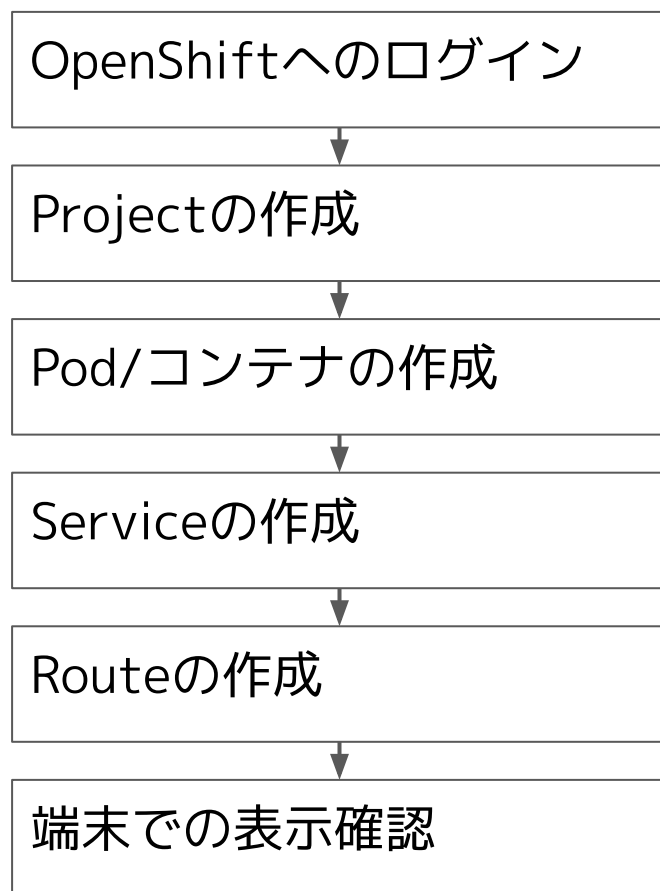
# テーマ1：コンテナの体験 ハンズオン編

レッドハット株式会社  
テクニカルセールス本部

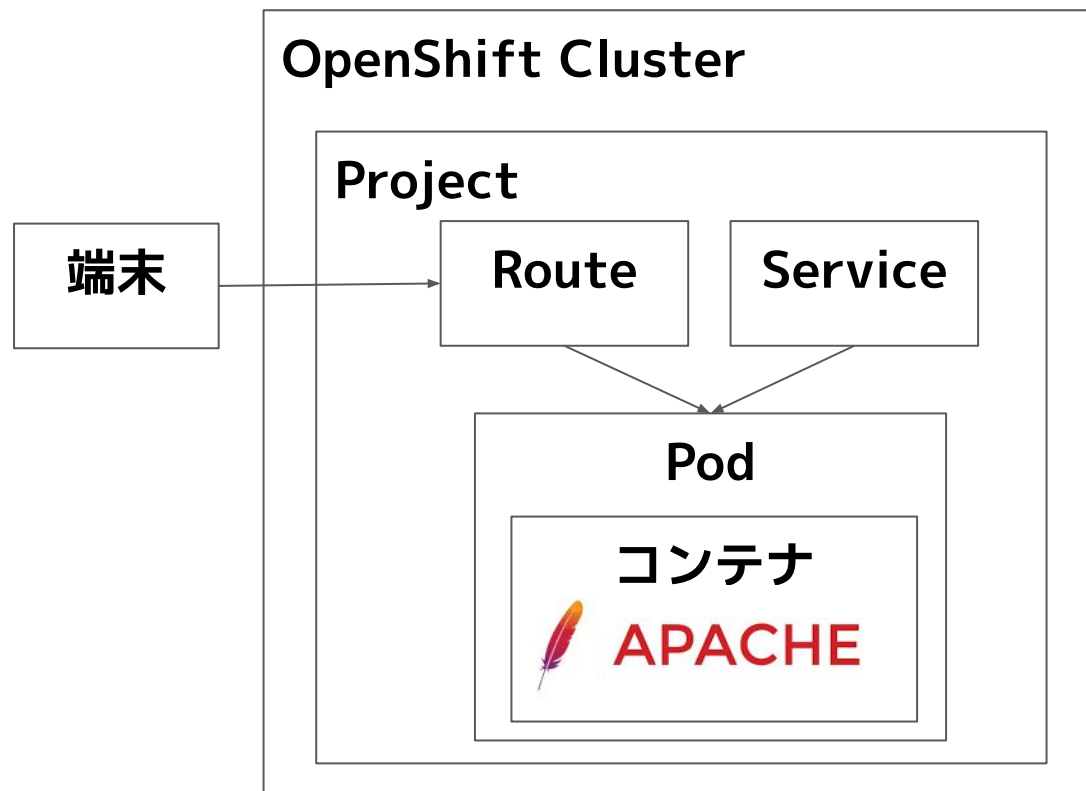
# ハンズオンの流れ

本ハンズオンでは、Apache Webサーバのコンテナをデプロイし、端末のブラウザからApache WebサーバのHTMLが表示するところまでを実施します。

## <流れ>



## <構成>



# 1. OpenShiftへのログイン

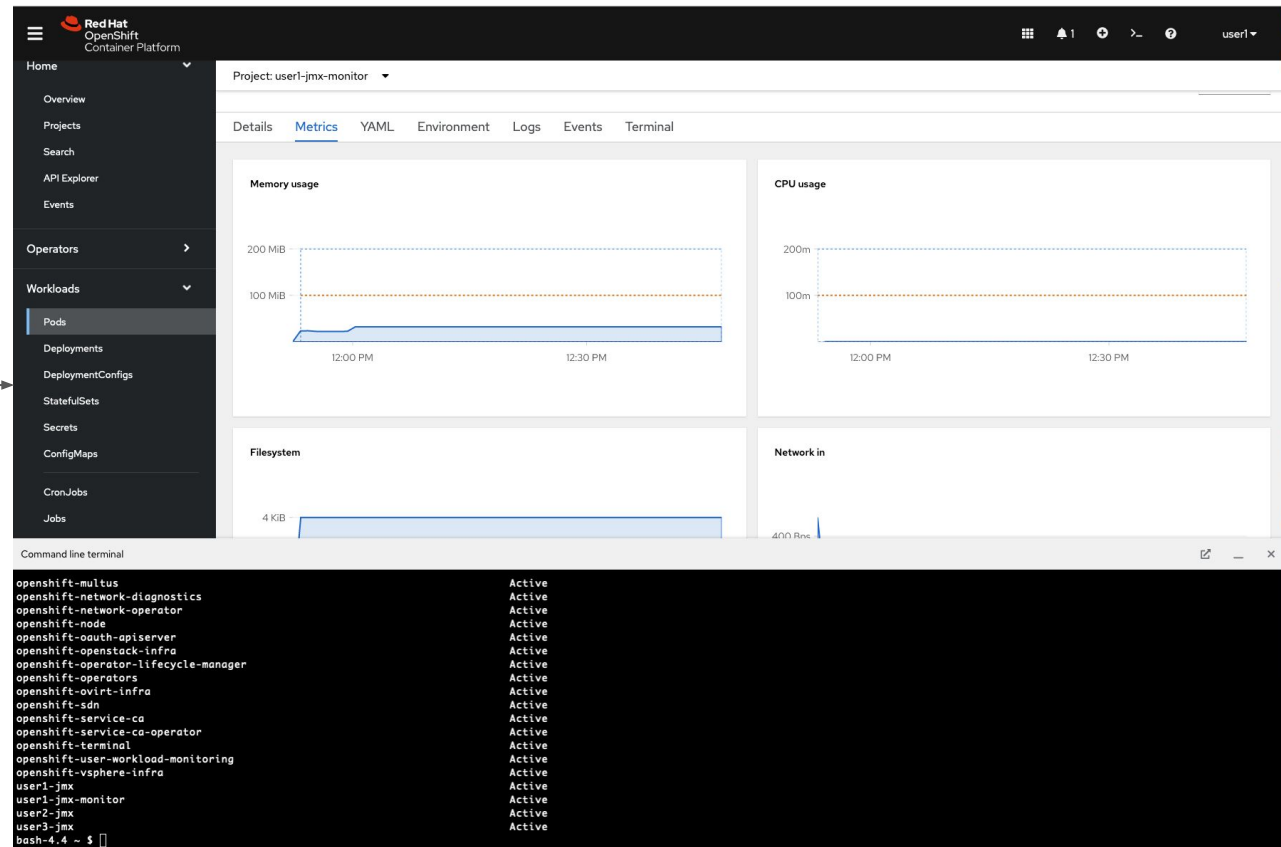
ハンズオンの操作は、全てブラウザ上で実施します。(同等の操作は、コマンド(CUI)でも可能です)  
OpenShiftの環境は、クラウド上に配備されています。

OpenShiftの環境情報：<ホワイトボードのURLをご確認ください>



受講者様

PC端末  
(Linux/Wi  
n/Mac)

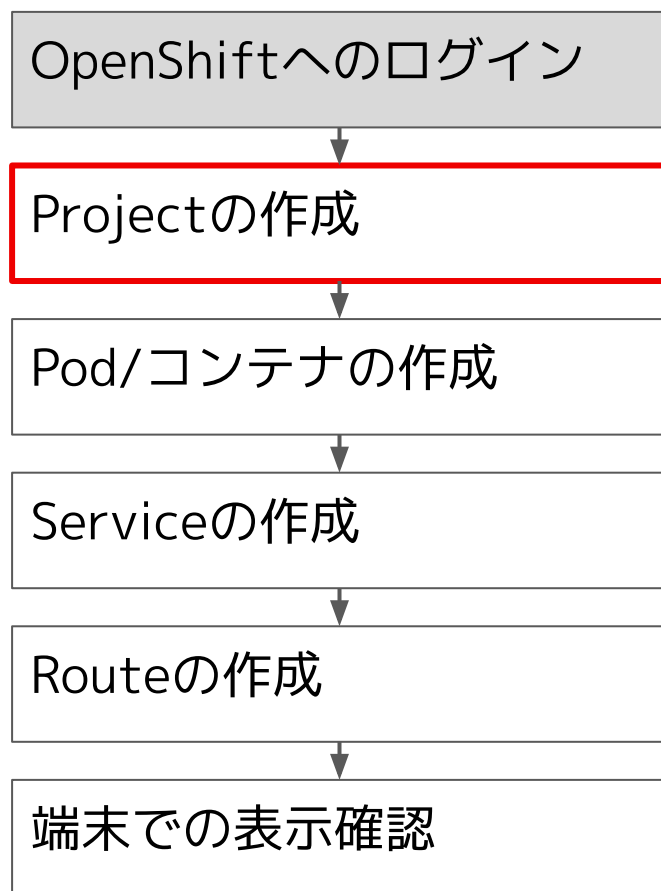


## 2.プロジェクトの作成

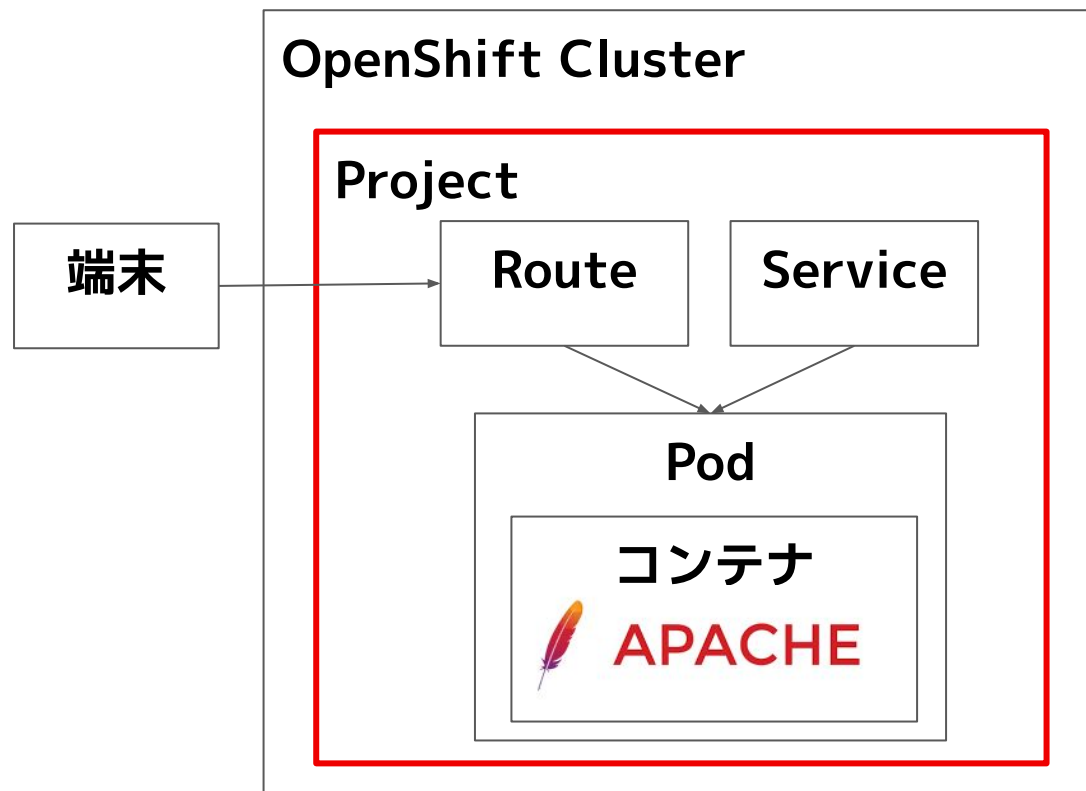
ここから、各自のペースで実施ください

まずは、新規Pod/コンテナ作成ために、プロジェクトを作成します。

### <流れ>



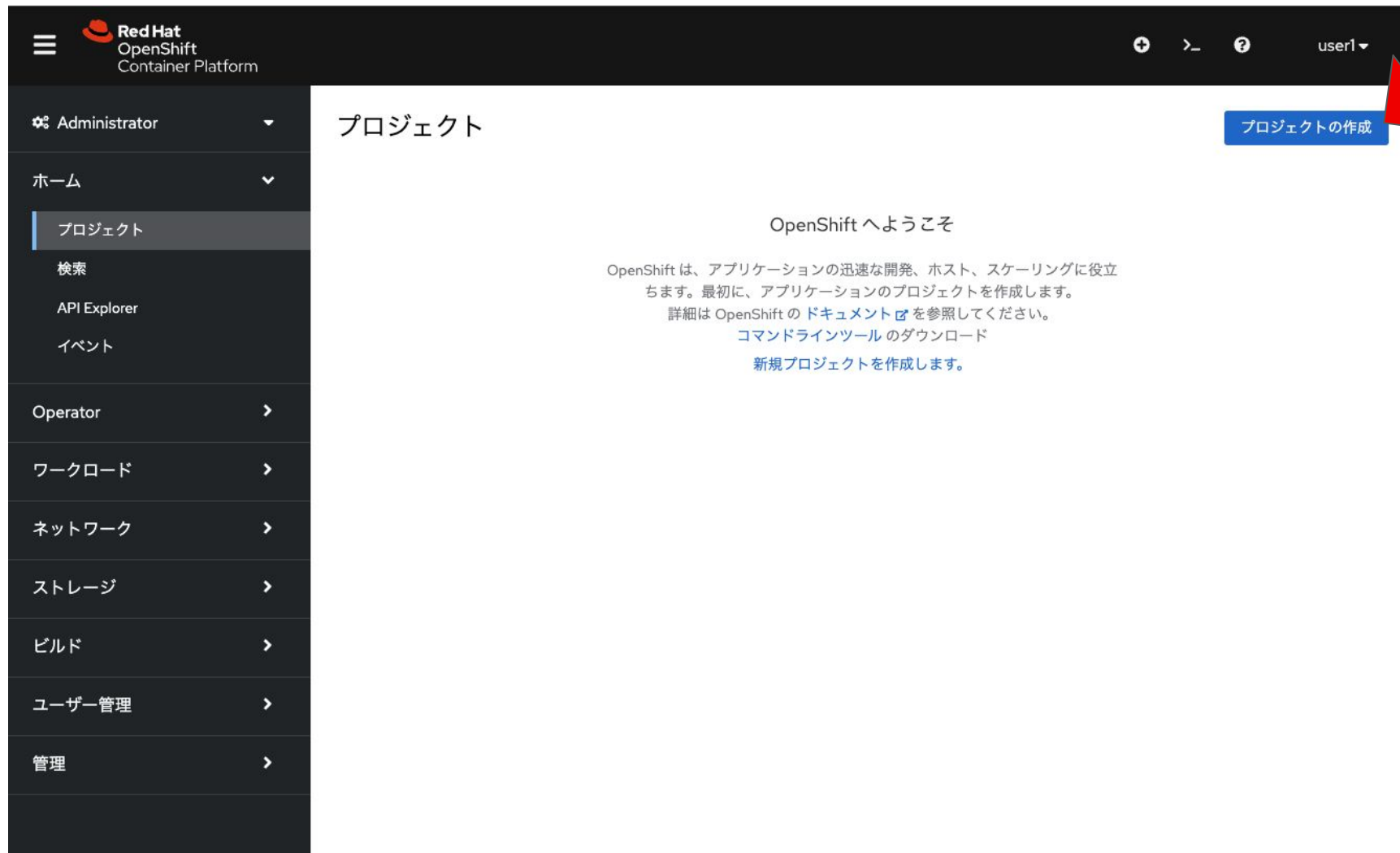
### <構成>



## 2.プロジェクトの作成

左ペインの[Administrator]⇒[ホーム]⇒[プロジェクト]を選択します。

プロジェクト画面に遷移後、「プロジェクトの作成」ボタンを押下します。



## 2. プロジェクトの作成

プロジェクトの作成画面が表示されます。

名前の欄に「project-<ユーザ名(※1)>」を入力し、「作成」ボタンを押下します。

プロジェクトの作成

OpenShift プロジェクトは、Kubernetes namespace の代わりの表現です。

プロジェクトの使用方法は、[こちらを参照してください。](#)

名前 \* ②

project-user1

表示名

説明

キャンセル 作成

※1 例: project-user1

ユーザ名は、OpenShiftにログインしたユーザIDです。

ユーザIDは、ブラウザの右上(例:user1)にも表示されています。

OpenShift(k8s)では、プロジェクト名(名前空間)は、1つのクラスターで一意的である必要があるため、今回はユーザ毎にプロジェクト名を分けています。

## 2. プロジェクトの作成

プロジェクトの作成が完了すると、プロジェクトの詳細画面が表示されます。  
プロジェクト内には、コンテナなどが存在しないため、CPUなどのリソース状況などは、まだ表示されません。

The screenshot displays the Red Hat OpenShift Container Platform console. The left sidebar contains navigation links: Administrator, Home, Project (selected), Search, API Explorer, Events, Operator, Workload, Network, Storage, Build, User Management, and Management. The main content area shows the 'project-user1' project details. The project status is 'Active'. Below the status, there are tabs for Overview, Details, YAML, Workload, and Role Binding. The 'Overview' tab is active, showing a table of resource usage. The table has columns for Resource, Usage, and a description. The resources listed are CPU, Memory, File System, Network Transfer, and Pod count, all of which are marked as 'Not Used' (利用不可). The right sidebar shows the 'Active' status and a section for 'Recent Events' (最近のイベント) with a 'Show' (表示) button.

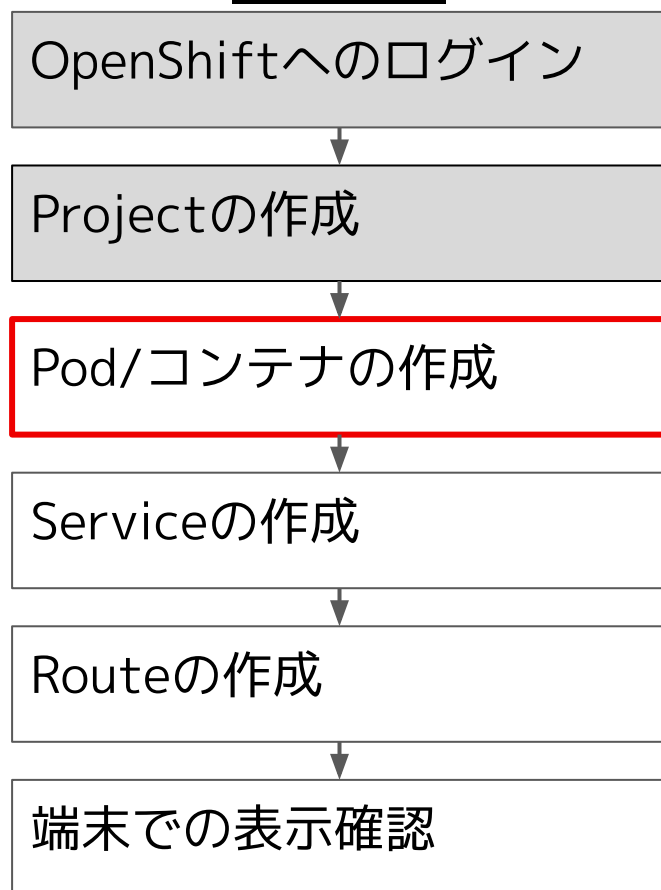
リソース	使用	1時間
CPU	利用不可	データポイントが見つかりません。
メモリー	利用不可	データポイントが見つかりません。
ファイルシステム	利用不可	データポイントが見つかりません。
ネットワーク転送	利用不可	データポイントが見つかりません。
Pod 数	利用不可	データポイントが見つかりません。

### 3.Podの作成

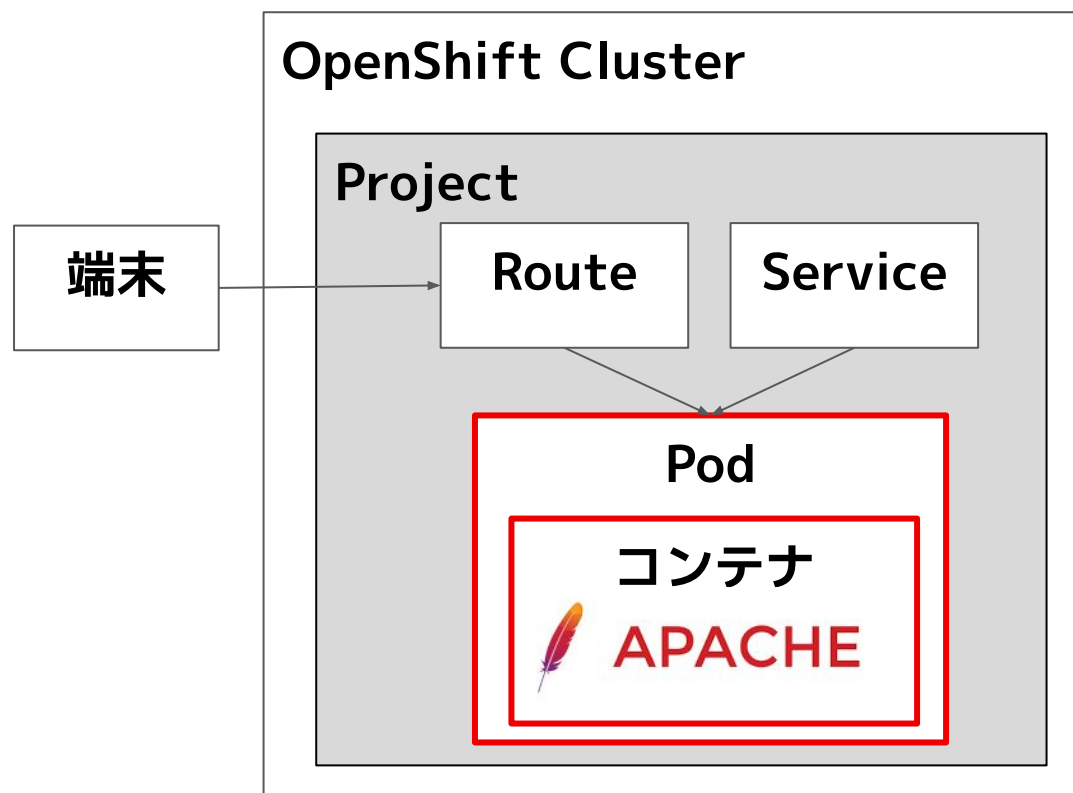
作成したプロジェクト内に、ApacheのPod/コンテナを作成します。

Podの作成時にコンテナイメージを指定するため、Pod作成時にコンテナも一緒に作成されます。

#### <流れ>



#### <構成>





### 3.Podの作成

左ペインの[Administrator]⇒[ワークロード]⇒[Pod]を選択します。  
Pod画面に遷移後、「Podの作成」ボタンを押下します。



### 3.Podの作成

Podの作成画面に遷移します。

デフォルトで、コンテナイメージが

Apacheを指定しています。

左下の作成を押下し、Podを作成します。

[Tips]

OpenShiftは、内部でコンテナレジストリを持っています。

- ・コンテナレジストリのURL

Image-registry.openshift-image-registry.svc:5000

- ・イメージのパス

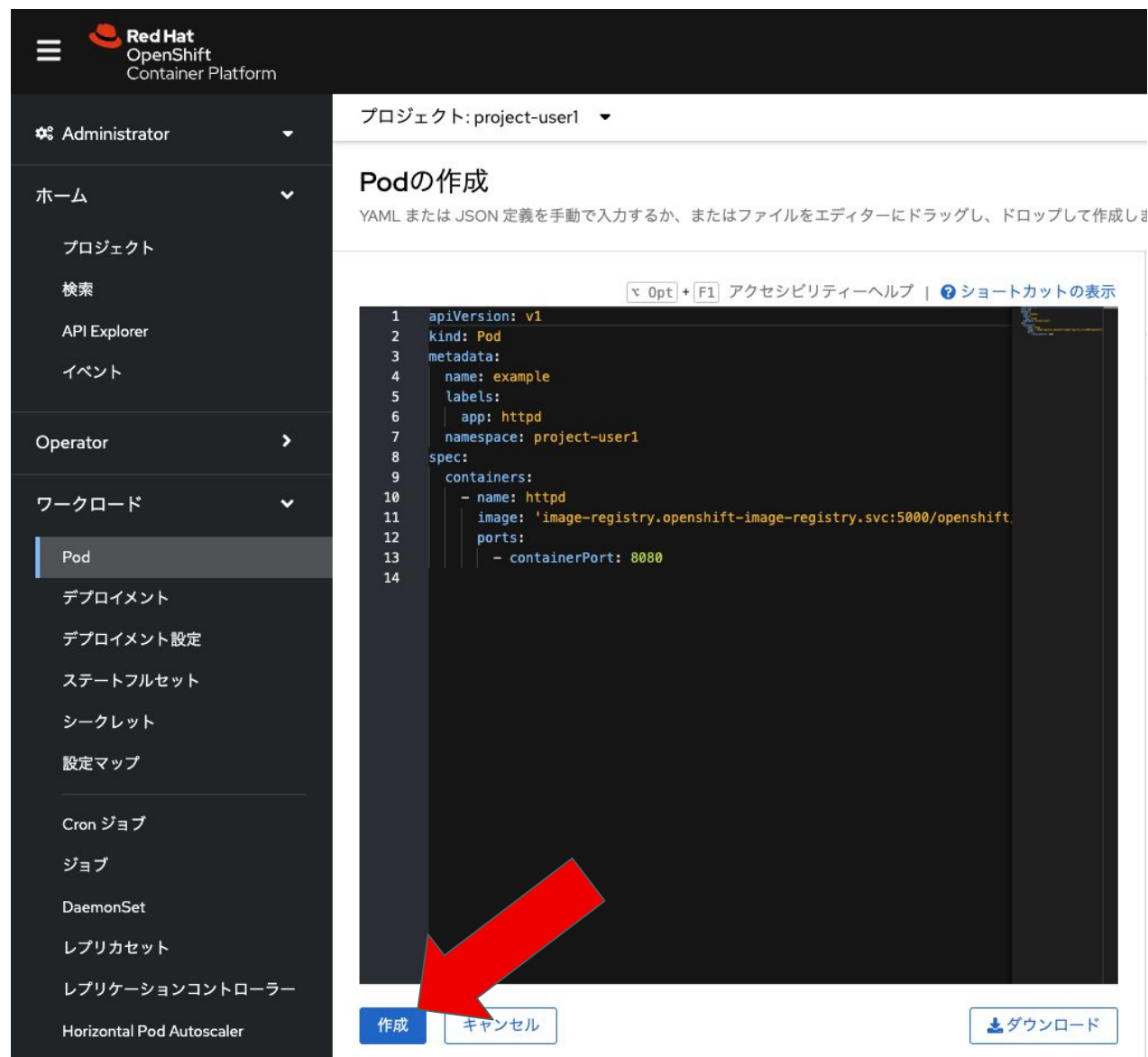
/openshift/httpd:latest

openshift: 名前空間

(デフォルトのイメージが格納されています)

httpd : Apacheのコンテナイメージ

latest : 最新のコンテナイメージを指定



# 3.Podの作成

Podの作成を押下すると、Podの詳細画面が表示されます。

Red Hat  
OpenShift  
Container Platform

Administrator

ホーム

プロジェクト

検索

API Explorer

イベント

Operator

ワークロード

Pod

デプロイメント

デプロイメント設定

ステートフルセット

シークレット

設定マップ

Cron ジョブ

ジョブ

プロジェクト: project-user1

Pod > Podの詳細

P example Running ← Runningになると、作成完了です。アクション

詳細    メトリクス    YAML    環境    ログ    イベント    ターミナル

Podの詳細

名前  
example

namespace  
NS project-user1

ラベル  
app=httpd

ノードセクター  
セクターなし

容認  
3件の容認

アノテーション  
4件のアノテーション

作成した時間:

ステータス  
Running

再起動ポリシー  
常に再起動する

有効な期限(秒)  
未設定

Pod IP  
10.129.2.32

ノード  
ip-10-0-207-70.ap-southeast-1.compute.internal

### 3.Podの作成

Podが作成できたので、CPUやメモリリソースを使っているため、Podのリソースの使用状況を確認します。  
Podの画面にある「メトリクス」を選択し、リソース状況が見れることを確認します。(ちょっと見る程度で結構です)

Red Hat OpenShift Container Platform

プロジェクト: project-user1

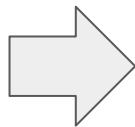
Pod > Podの詳細

**example** Running

詳細 | **メトリクス** | YAML | 環境 | ログ | イベント | ターミナル

**Podの詳細**

名前	example	ステータス	Running
namespace	project-user1	再起動ポリシー	常に再起動する
ラベル	app=httpd	有効な期限 (秒)	未設定
ノードセクター	セクターなし	Pod IP	10.129.2.34
承認	3件の承認	ノード	ip-10-0-207-70
アノテーション	4件のアノテーション		
作成した時間	5分前		
オーナー	オーナーなし		



Red Hat OpenShift Container Platform

プロジェクト: project-user1

Pod > Podの詳細

**example** Running

詳細 | **メトリクス** | YAML | 環境 | ログ | イベント | ターミナル

**メモリ使用量**

1 GiB

14:55 15:00 15:05

**CPUの使用率**

400m

200m

14:55 15:00 15:05

**ファイルシステム**

40 KiB

20 KiB

14:55 15:00 15:05

**ネットワーク IN**

1 Bps

0.5 Bps

14:55 15:00 15:05

**ネットワーク OUT**

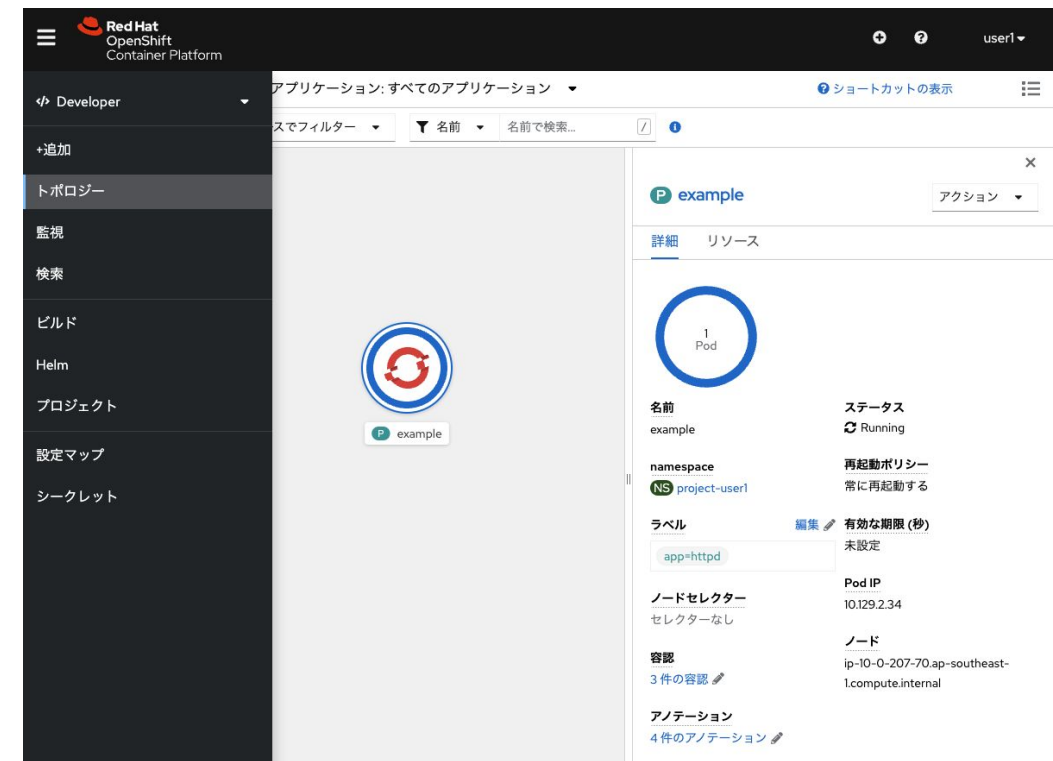
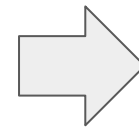
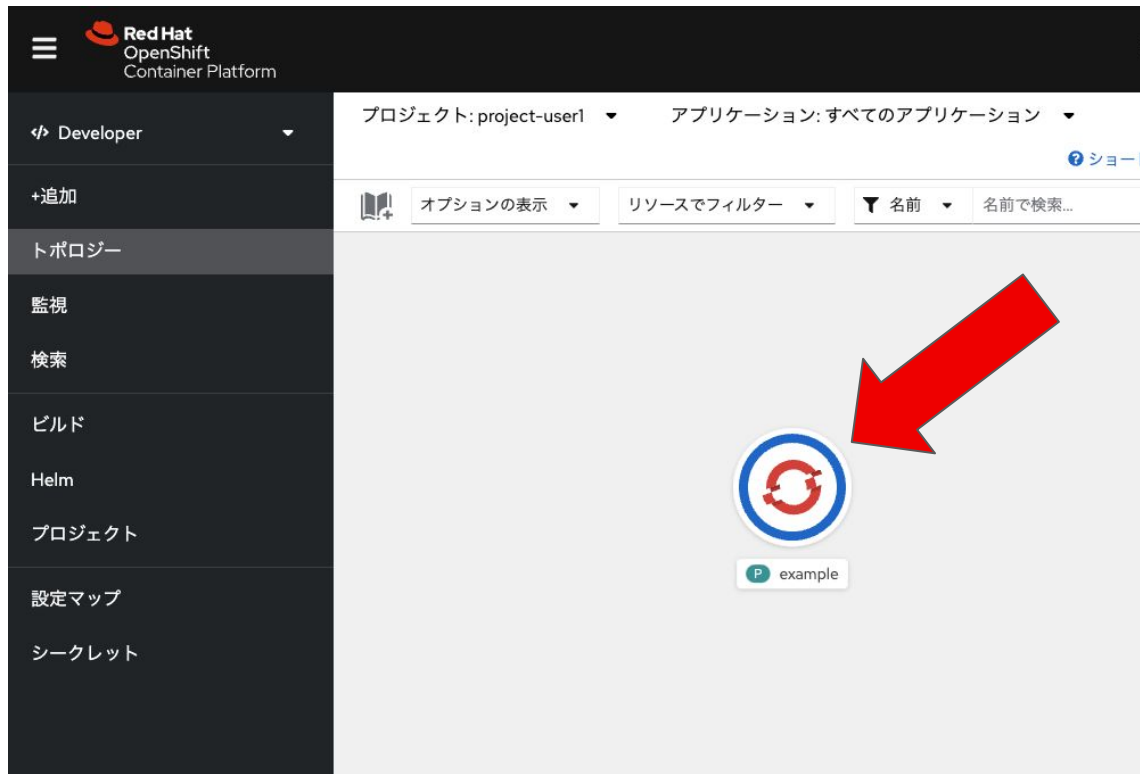
### 3.Podの作成(トポロジービューでの確認)

OpenShiftには、プロジェクト内で作成したリソースをGUIで確認する「トポロジービュー」があります。

左ペインを[Administrator]から[Developer]に切り替えて、[トポロジー]を選択します。

Podのアイコンをクリックすると、Podの情報が表示されます。

以降の手順でServiceやRouteのリソースを追加後、適宜、トポロジービューを参照します。



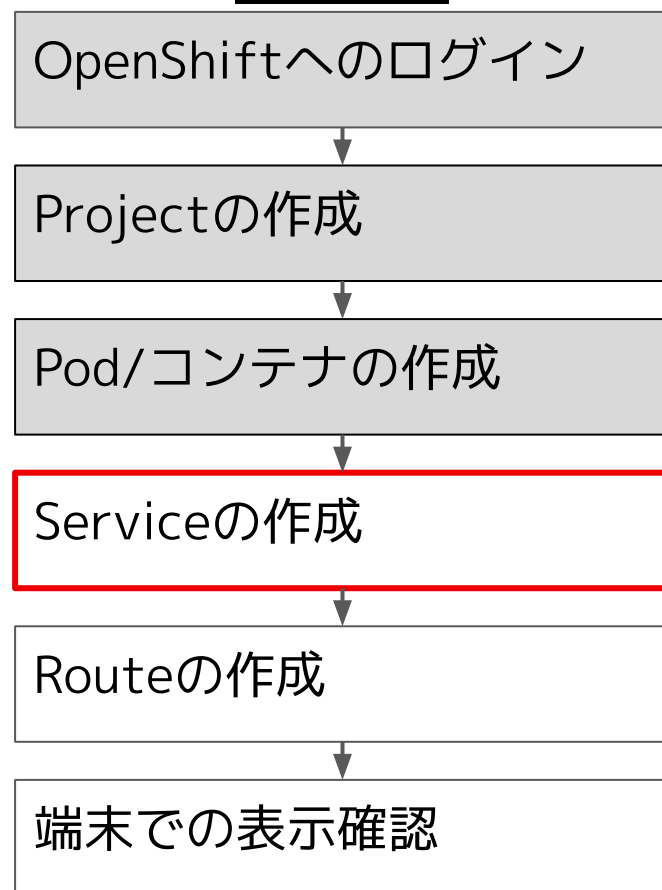
## 4.Serviceの作成

作成したApacheのPod/コンテナに対して、OpenShift内で通信するためのServiceを作成します。

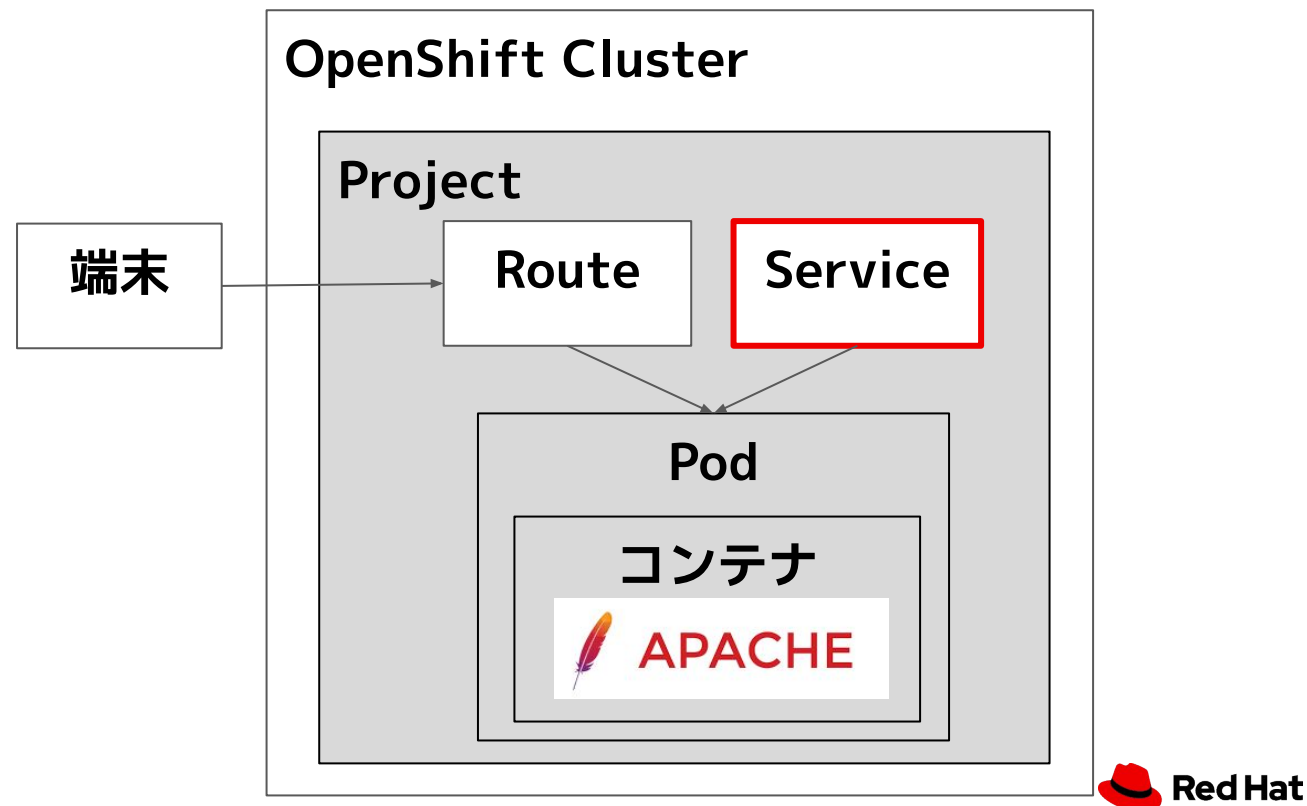
※ Routeを作成するには前提としてServiceの作成が必要です。

Serviceは、Pod/コンテナ間の通信にも利用します。

### <流れ>

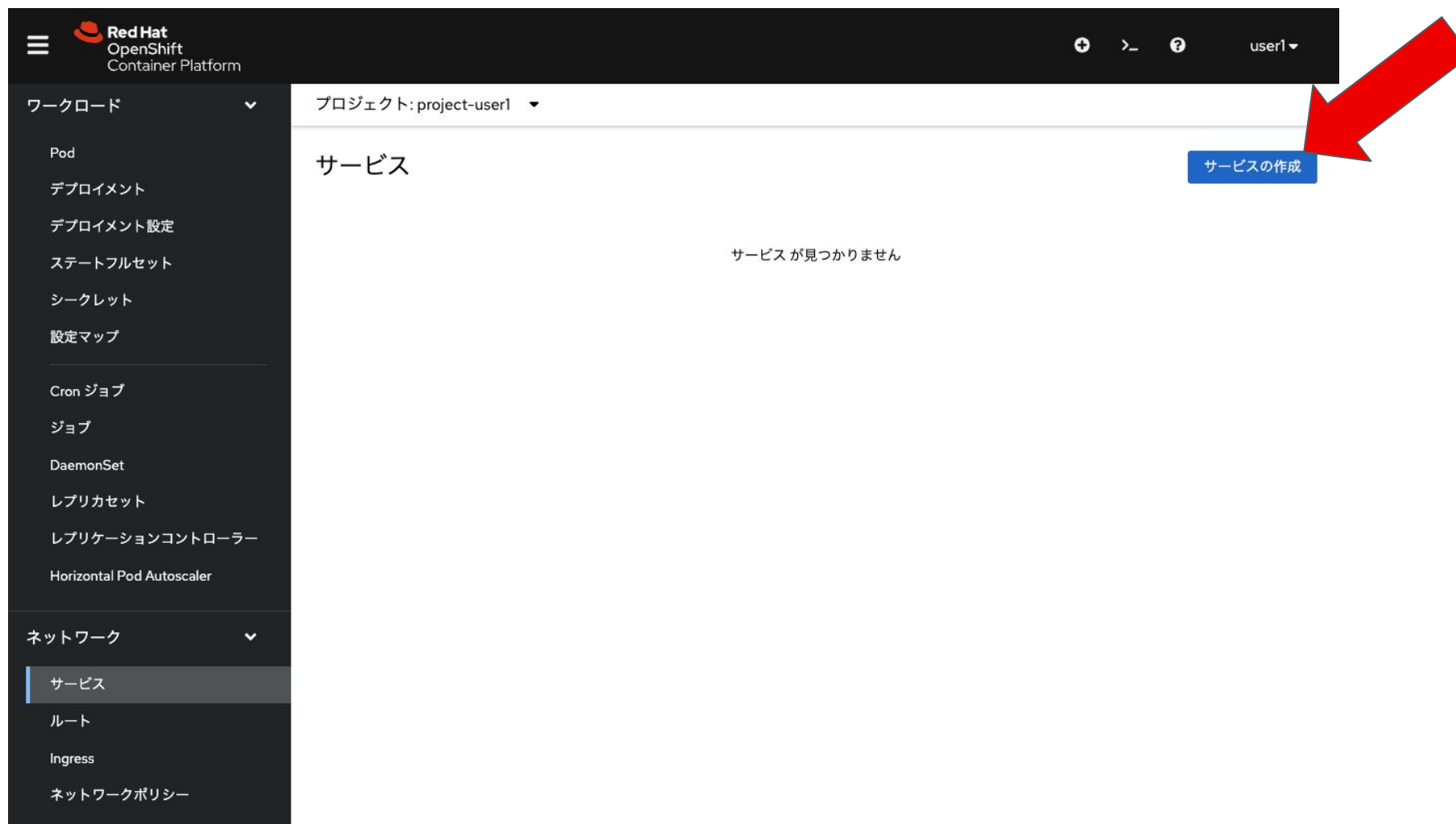


### <構成>



## 4.Serviceの作成

左ペインの[Administrator]⇒[ネットワーク]⇒[サービス]を選択します。  
サービス画面に遷移後、「サービスの作成」ボタンを押下します。



## 4.Serviceの作成

Serviceの作成画面が表示されます。

Serviceの定義は、修正する必要があります。

次ページで、修正内容を説明いたします。

Red Hat OpenShift Container Platform

プロジェクト: project-user1

### サービスの作成

YAML または JSON 定義を手動で入力するか、またはファイルをエディターにドラッグし、ドロップして作成します。

```
1 apiVersion: v1
2 kind: Service
3 metadata:
4   name: example
5   namespace: project-user1
6 spec:
7   selector:
8     app: MyApp
9   ports:
10    - protocol: TCP
11      port: 80
12      targetPort: 9376
13
```

#### サービス

##### スキーマ

Service is a named abstraction of software service (for example, mysql) consisting of local port (for example 3306) that the proxy listens on, and the selector that determines which pods will answer requests sent through the proxy.

- **apiVersion**  
string  
APIVersion defines the versioned schema of this representation of an object. Servers should convert recognized schemas to the latest internal value, and may reject unrecognized values. More info: <https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources>



## 4.Serviceの作成

Serviceの定義では、対象とするPodを指定するため、

Serviceの定義のspec -> selector -> appとspec -> ports -> targetPortの2箇所を修正する必要があります。

### < Podの定義(yaml) >

```
apiVersion: v1
kind: Pod      : リソースがPodであることを指定
metadata:
  name: example : このPodの名称
  labels:
    app: httpd   : 識別するためのラベルを app=httpdとする
  namespace: project-user1 : 所属するプロジェクト名
spec:
  containers:
    - name: httpd : コンテナの名称
      image: 'image-registry.openshift-image-registry.svc:5000/openshift/httpd:latest'
      ports:
        - containerPort: 8080 : コンテナが使うポート
```

### < Serviceの定義(yaml) >

```
apiVersion: v1
kind: Service : リソースがServiceであることを指定
metadata:
  name: example : このServiceの名称
  namespace: project-user1 : 所属するプロジェクト名
spec:
  selector:
    app: httpd : Serviceの対象とするラベルを指定 app=httpdとする
  ports:
    - protocol: TCP : Serviceの利用するプロトコル
      port: 80      : Serviceが公開するポート
      targetPort: 8080 : コンテナに転送するポート
```

## 4.Serviceの作成

Serviceの定義の修正が完了したら、「作成」ボタンを押下します。

Red Hat OpenShift Container Platform

プロジェクト: project-user1

### サービスの作成

YAML または JSON 定義を手動で入力するか、またはファイルをエディターにドラッグし、ドロップして作成しま

```
1  apiVersion: v1
2  kind: Service
3  metadata:
4    name: example
5    namespace: project-user1
6  spec:
7    selector:
8      app: httpd
9    ports:
10     - protocol: TCP
11       port: 80
12       targetPort: 8080
13
```

⌘ Opt + F1 アクセシビリティヘルプ | ⓘ ショートカットの表示

作成 キャンセル ダウンロード

## 4.Serviceの作成

Serviceの作成を押下すると、Serviceの詳細画面が表示されます。

The screenshot displays the Red Hat OpenShift Container Platform interface. The left sidebar contains navigation menus for 'API Explorer', 'イベント', 'Operator', 'ワークロード', and 'ネットワーク'. The 'ワークロード' menu is expanded, showing options like 'Pod', 'デプロイメント', 'ステートフルセット', 'シークレット', '設定マップ', 'Cron ジョブ', 'ジョブ', 'DaemonSet', 'レプリカセット', 'レプリケーションコントローラー', and 'Horizontal Pod Autoscaler'. The 'ネットワーク' menu is also expanded, showing 'サービス' (selected), 'ルート', and 'Ingress'.

The main content area shows the details of a Service named 'example' in the 'project-user1' namespace. The 'サービス' tab is selected, showing the following information:

- 名前:** example
- namespace:** project-user1
- ラベル:** ラベルなし
- Pod セレクター:** app=httpd
- アノテーション:** 0 件のアノテーション
- セッションアフィニティー:** None
- 作成した時間:** 2 分前
- オーナー:** オーナーなし

On the right side, the 'サービスルーティング' section shows the 'ホスト名' as 'example.project-user1.svc.cluster.local' and the 'サービスアドレス' table:

タイプ	場所
クラスター IP	172.30.159.157
クラスター内でのみアクセス可能	

Below this, the 'サービスポートマッピング' section shows a table with the following data:

名前	ポート	プロトコル	Pod ポートまたは名前
-	80	TCP	8080

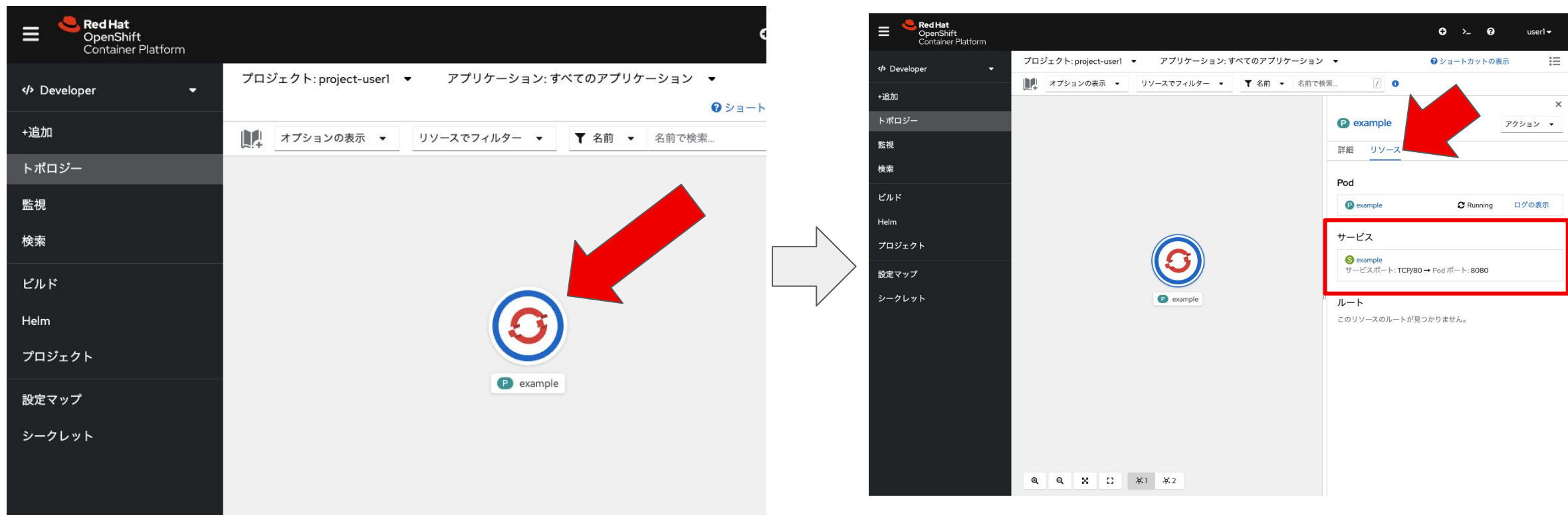
## 4.Serviceの作成(トポロジービューでの確認)

Serviceが出来たことを トポロジービューで確認します。

左ペインを[Administrator]から[Developer]に切り替えて、[トポロジー]を選択します。

Podのアイコンをクリックすると、Podの情報が表示されます。

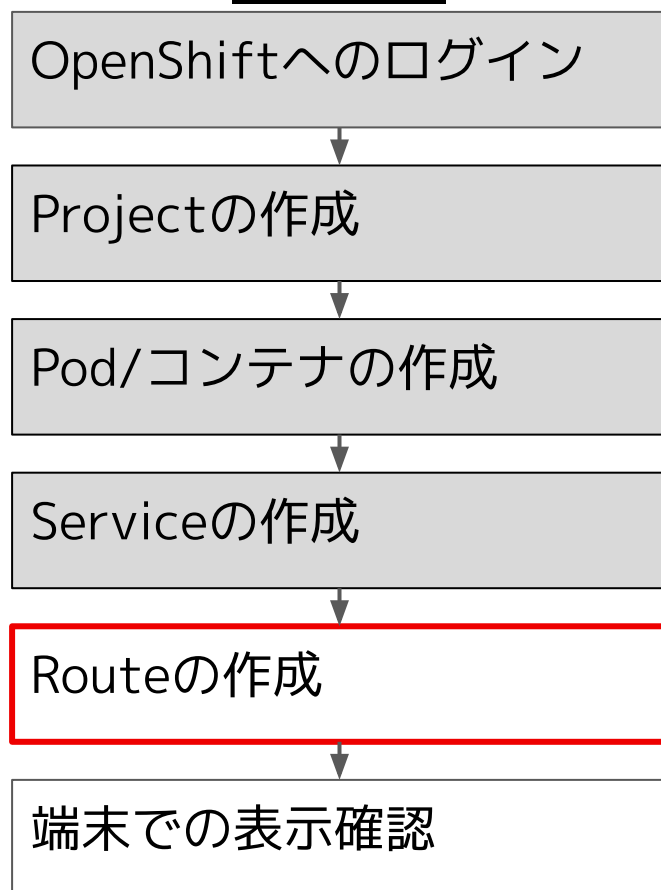
Podの情報のリソースをクリックすると、Serviceが作成されていることを確認できます。



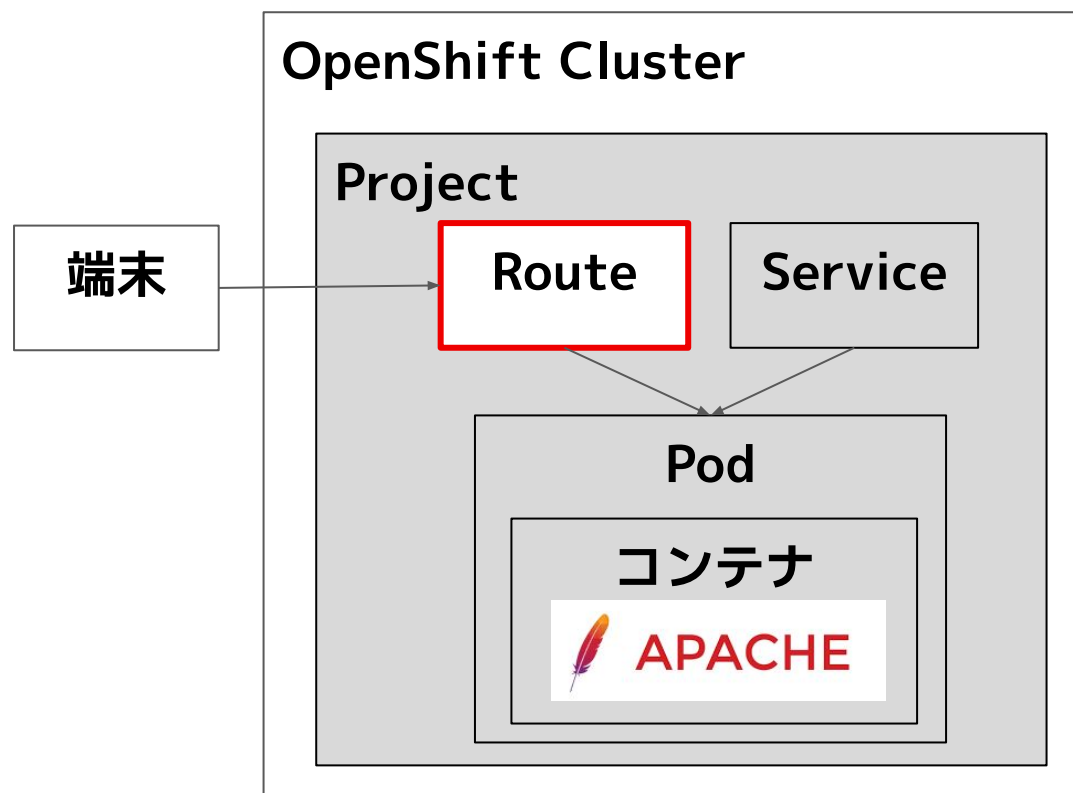
## 5.Routeの作成

作成したApacheのPod/コンテナに対して、OpenShift外で通信するためのRouteを作成します。

### <流れ>

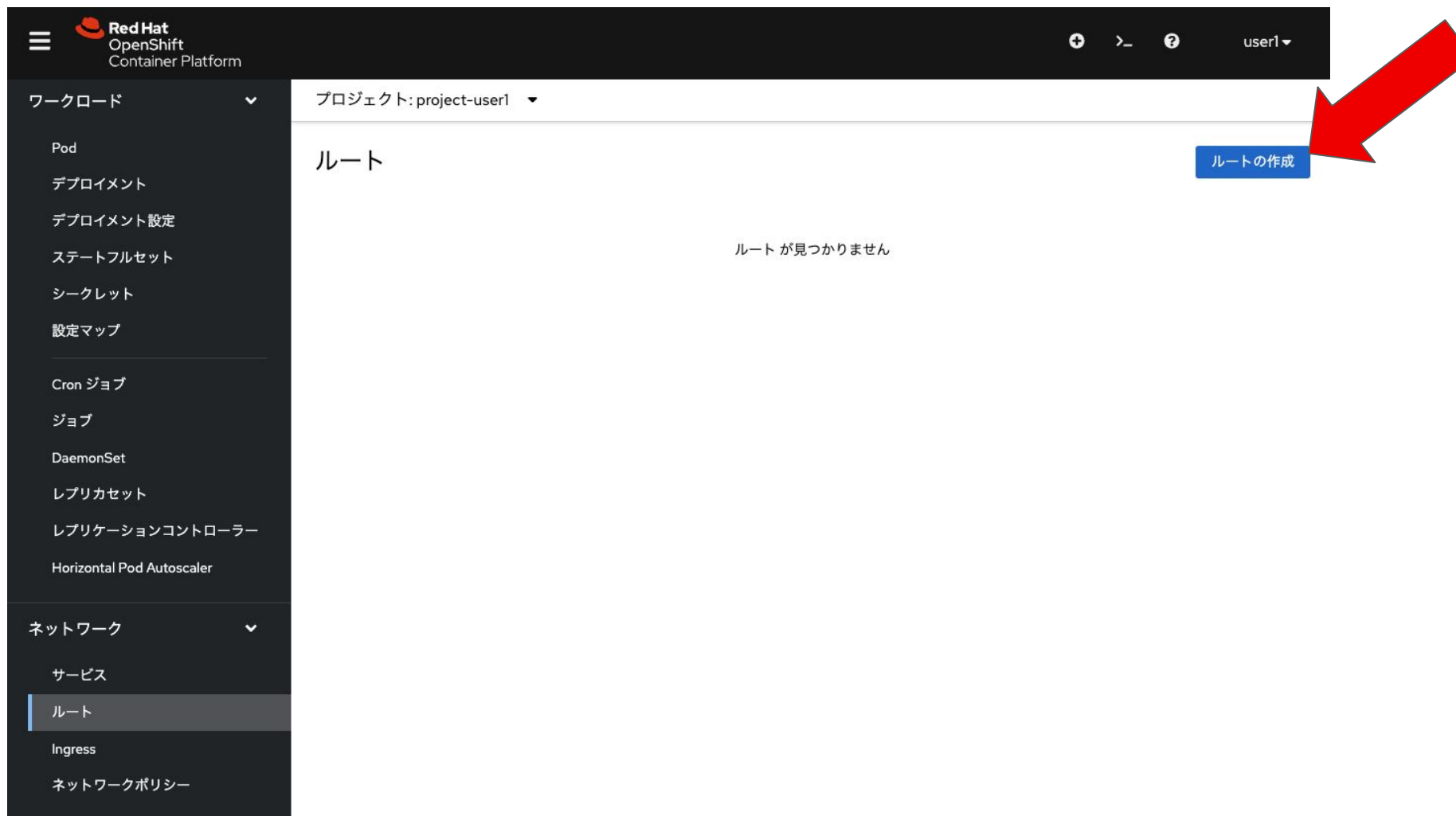


### <構成>



## 5.Routeの作成

左ペインの[Administrator]⇒[ネットワーク]⇒[ルート]を選択します。  
サービス画面に遷移後、「ルートの作成」ボタンを押下します。



# 5.Routeの作成

Routeの作成画面が表示されます。

Routeの定義は、 入力および選択する必要があります。

今回は、最低限の入力項目である「名前」の入力と「サービス」と「ターゲットポート」のみ選択します。

Red Hat OpenShift Container Platform

プロジェクト: project-user1

### ルートの作成

ルーティングはアプリケーションを一般に公開する方法です。

**名前 \***  
my-route  
プロジェクト内のルートの一意の名前

**ホスト名**  
www.example.com  
ルートのパブリックホスト名。指定されない場合は、ホスト名が生成されます。

**パス**  
/  
ルーターがトラフィックをサービスにルーティングするために監視するパス。

**サービス \***  
サービスの選択  
ルーティングするサービス。

**ターゲットポート \***  
上記のサービスを選択  
トラフィックのターゲットポート。

**セキュリティ**  
☐ セキュアなルート  
ルートのセキュリティは、証明書を提供するために複数の TLS 終端タイプを使用して保護できます。

作成 キャンセル

example と入力します

サービス \*

サービスの選択

example

Serviceのexampleを選択します

ターゲットポート \*

ターゲットポートの選択

80 → 8080 (TCP)

80→8080(TCP)を選択します

## 5.Routeの作成

Routeの定義の修正が完了したら、「作成」ボタンを押下します。

The screenshot shows the 'Routeの作成' (Create Route) page in the Red Hat OpenShift Container Platform web console. The left sidebar contains navigation links for Administrator, Home, Project, Search, API Explorer, Events, Operator, Workload, Network, Service, Route (selected), Ingress, Network Policy, Storage, Build, User Management, and Management. The main content area shows the 'Routeの作成' form with the following fields:

- 名前 \***: example
- ホスト名**: www.example.com
- パス**: /
- サービス \***: example
- ターゲットポート \***: 80 → 8080 (TCP)
- セキュリティ**: ☐ セキュアなルート

A red arrow points to the '作成' (Create) button at the bottom of the form.



## 5.Routeの作成

Routeの作成を押下すると、Routeの詳細画面が表示されます。  
次のページ以降にて、端末にて接続確認をします。

The screenshot displays the Red Hat OpenShift Container Platform web console. The left sidebar shows the navigation menu with 'Route' selected under the 'Network' section. The main content area shows the details of a Route named 'example' in the 'project-user1' namespace. The route is in an 'Accepted' state. The 'Labels' field is empty. The 'Annotations' section shows one annotation. The 'Service' is 'example'. The 'Target Port' is '8080'. The 'Created Time' is '進行中' (In Progress). The 'Owner' is 'オーナーなし' (No Owner). The 'Location' is 'http://example-project-user1.apps.cluster-rt6jf.rt6jf.sandbox1171.opentlc.com'. The 'Status' is 'Accepted'. The 'Host' is 'example-project-user1.apps.cluster-rt6jf.rt6jf.sandbox1171.opentlc.com'. The 'Path' is empty. The 'Router's Canonical Host Name' is 'router-default.apps.cluster-rt6jf.rt6jf.sandbox1171.opentlc.com'.

Red Hat OpenShift Container Platform

プロジェクト: project-user1

ルート > ルートの詳細

RT example Accepted

アクション

詳細 メトリクス YAML

ルートの詳細

名前  
example

場所  
<http://example-project-user1.apps.cluster-rt6jf.rt6jf.sandbox1171.opentlc.com>

namespace  
NS project-user1

ステータス  
Accepted

ラベル  
ラベルなし

編集

ホスト  
example-project-user1.apps.cluster-rt6jf.rt6jf.sandbox1171.opentlc.com

アノテーション  
1件のアノテーション

サービス  
S example

パス  
-

ターゲットポート  
8080

作成した時間:  
進行中

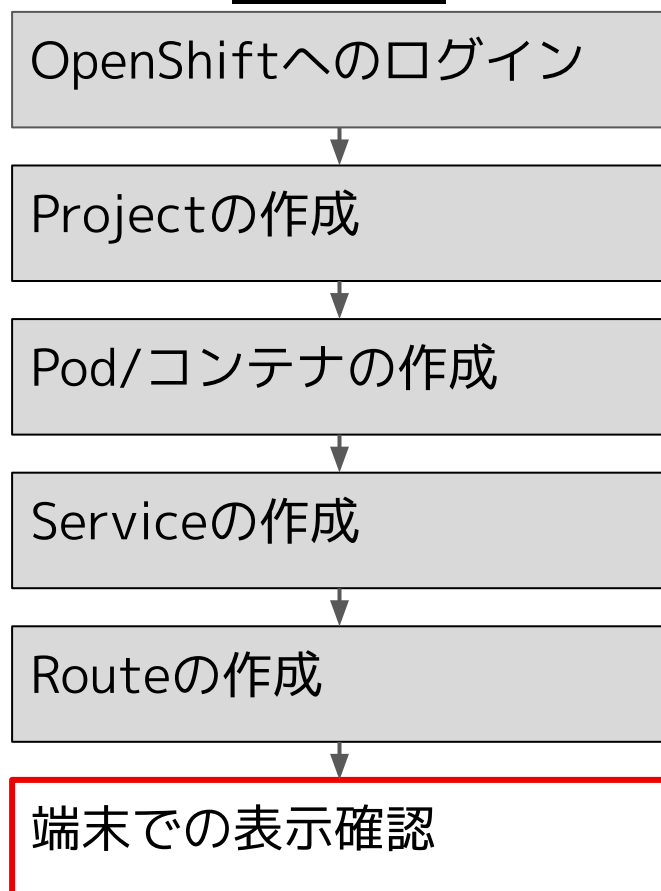
オーナー  
オーナーなし

ルーターの正規ホスト名  
router-default.apps.cluster-rt6jf.rt6jf.sandbox1171.opentlc.com

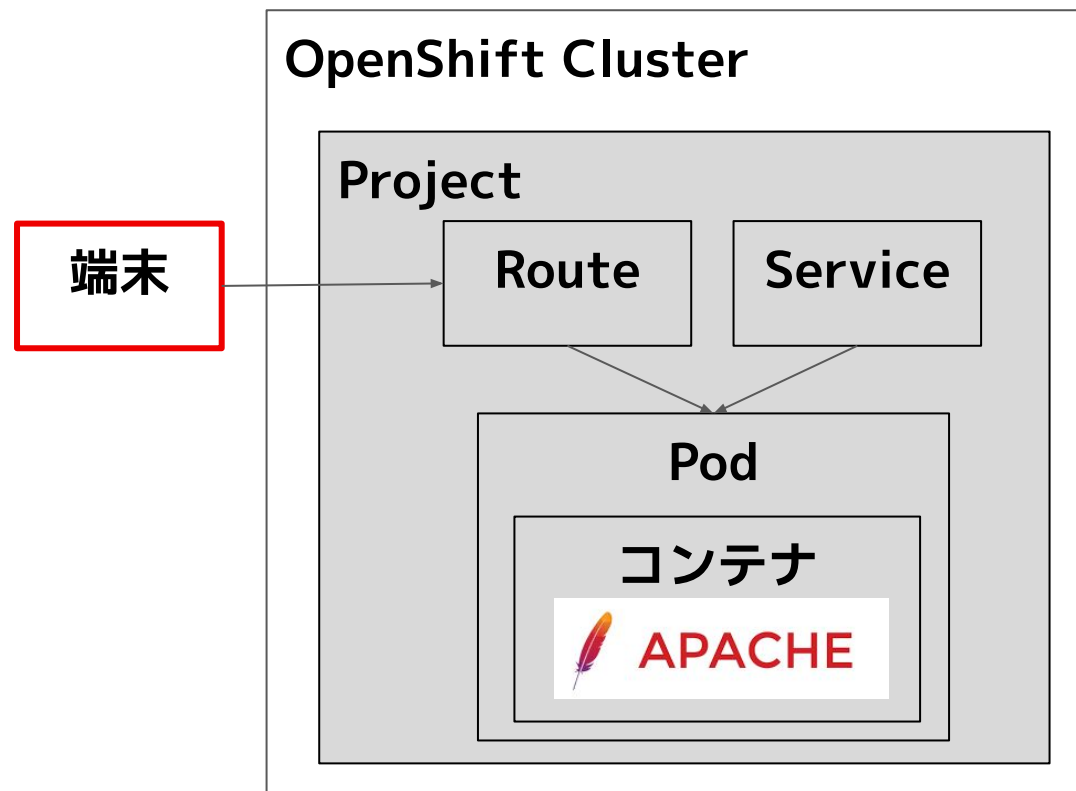
## 6. 端末での表示確認

最後に、端末での表示確認を実施します。

### < 流れ >



### < 構成 >



## 6. 端末での表示確認

Routeの詳細画面の「場所」と記載されている箇所が、RouteのURLになります。  
URLの箇所をクリックします。

The screenshot shows the Red Hat OpenShift Container Platform console interface. The left sidebar contains navigation menus for Administrator, Home, Project, Search, API Explorer, Events, Operator, Workload, Network, Service, Route, Ingress, Network Policy, Storage, Build, User Management, and Management. The main content area displays the details of a Route named 'example' in the 'project-user1' namespace. The 'Location' (場所) field is highlighted with a red arrow, showing the URL: `http://example-project-user1.apps.cluster-rt6jf.rt6jf.sandbox1171.opentlc.com`. Other fields include 'Status' (ステータス) as 'Accepted', 'Host' (ホスト) as `example-project-user1.apps.cluster-rt6jf.rt6jf.sandbox1171.opentlc.com`, 'Path' (パス) as '-', 'Router Canonical Hostname' (ルーターの正規ホスト名) as `router-default.apps.cluster-rt6jf.rt6jf.sandbox1171.opentlc.com`, 'Target Port' (ターゲットポート) as 8080, 'Created Time' (作成した時間) as 'In Progress' (進行中), and 'Owner' (オーナー) as 'None' (オーナーなし).

## 6. 端末での表示確認

端末のブラウザで以下の内容(Apacheのコンテナ)が表示されます。

### Red Hat Enterprise Linux Test Page

This page is used to test the proper operation of the HTTP server after it has been installed. If you can read this page, it means that the HTTP server installed at this site is working properly.

#### If you are a member of the general public:

The fact that you are seeing this page indicates that the website you just visited is either experiencing problems, or is undergoing routine maintenance.

If you would like to let the administrators of this website know that you've seen this page instead of the page you expected, you should send them e-mail. In general, mail sent to the name "webmaster" and directed to the website's domain should reach the appropriate person.

For example, if you experienced problems while visiting [www.example.com](#), you should send e-mail to "[webmaster@example.com](#)".



For information on Red Hat Enterprise Linux, please visit the [Red Hat, Inc. website](#). The documentation for Red Hat Enterprise Linux is [available on the Red Hat, Inc. website](#).

#### If you are the website administrator:

You may now add content to the webroot directory. Note that until you do so, people visiting your website will see this page, and not your content.

For systems using the Apache HTTP Server: You may now add content to the directory `/var/www/html/`. Note that until you do so, people visiting your website will see this page, and not your content. To prevent this page from ever being used, follow the instructions in the file `/etc/httpd/conf.d/welcome.conf`.

For systems using NGINX: You should now put your content in a location of your choice and edit the `root` configuration directive in the **nginx** configuration file `/etc/nginx/nginx.conf`.



[Apache™](#) is a registered trademark of [the Apache Software Foundation](#) in the United States and/or other countries.  
[NGINX™](#) is a registered trademark of [F5 Networks, Inc.](#)

## 6. 端末での表示確認(トポロジービューでの確認)

Routeが出来たことを トポロジービューで確認します。

左ペインを[Administrator]から[Developer]に切り替えて、[トポロジー]を選択します。

Podのアイコンをクリックすると、Podの情報が表示され、Podの右上のアイコンから、RouteのURLに遷移できます。

Podの情報のリソースをクリックすると、Routeが作成されていることを確認できます。

The screenshot shows the Red Hat OpenShift Container Platform interface in the Developer view. The left sidebar contains navigation options: Developer, +追加, トポロジー, 監視, 検索, ビルド, Helm, プロジェクト, 設定マップ, and シークレット. The main area displays the Topology view for the project 'project-user1'. A Pod named 'example' is shown with a status of 'Running'. A red box highlights the 'Route' icon in the Pod's top right corner, with an arrow pointing to the text 'RouteのURLに遷移'. The right-hand pane shows the details of the Pod, including its name 'example', status 'Running', and a link to 'ログの表示'. Below this, the 'ルート' (Route) section is highlighted with a red box, showing the Route name 'example' and its location: 'http://example-project-user1.apps.cluster-rt6jfrt6jfsandbox1171.opentlc.com'.

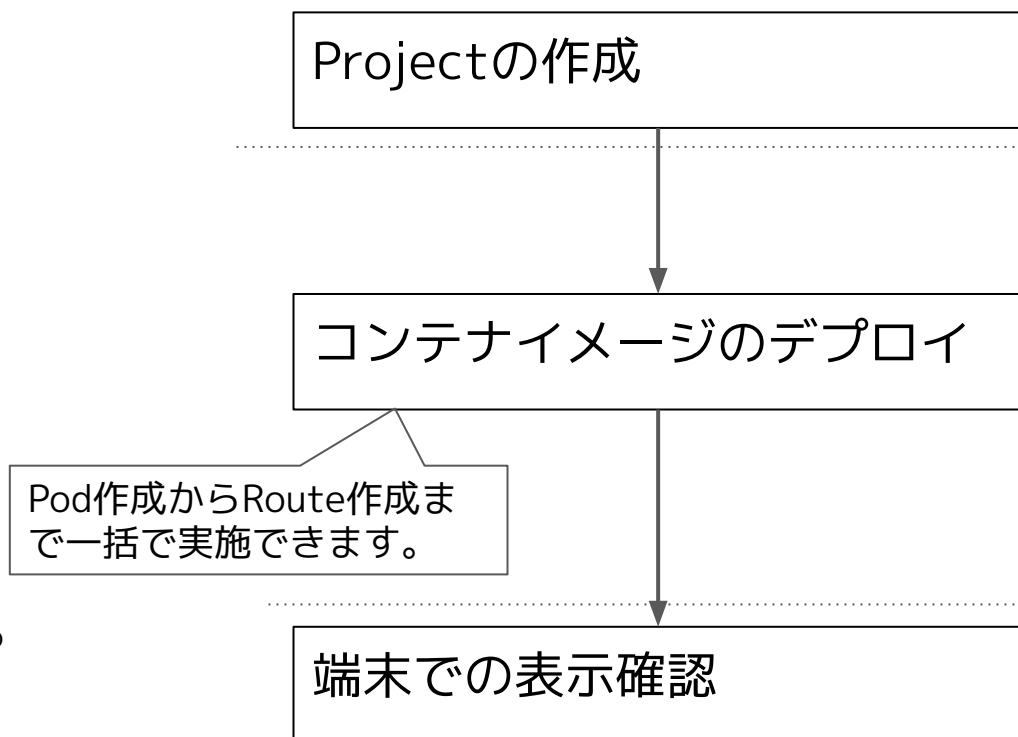
# X-1. ミドルウェアのコンテナイメージをデプロイ

以降の手順は、お時間が余った場合に可能なら実施ください。

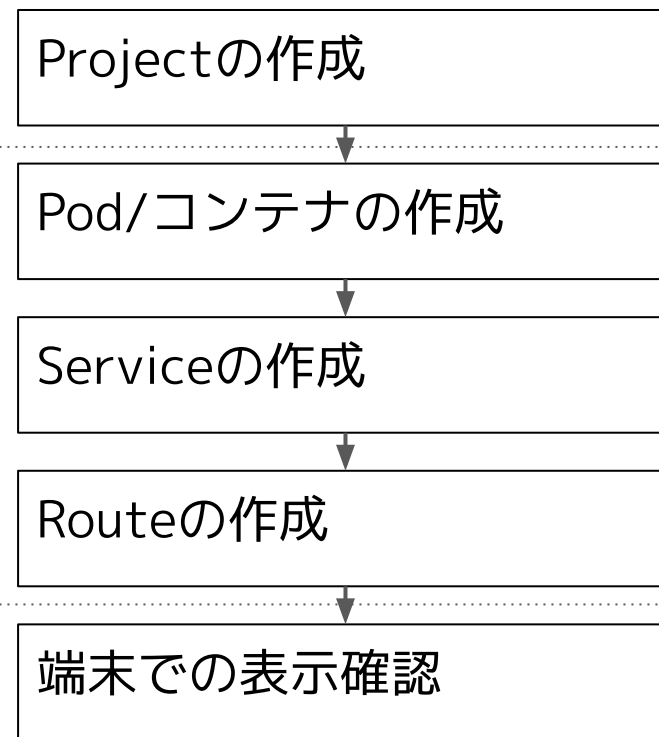
OpenShiftはJBossやTomcat等のミドルウェアのコンテナイメージや、OpenJDK用のコンテナイメージを提供しています。Red Hatは、コンテナ基盤からコンテナベースイメージ、ミドルウェアもトータルでサポートできるのが、他社にない強みになっています。

本手順では、認証・認可のOSSであるKeycloakのRed Hat製品版 Red Hat SSOのコンテナイメージをデプロイします。

## <ミドルウェアのコンテナイメージのデプロイ>



## <参考:先程のコンテナ作成の流れ>



## X-2.プロジェクトの作成

左ペインの[Administrator]⇒[ホーム]⇒[プロジェクト]を選択します。  
プロジェクト画面に遷移後、「プロジェクトの作成」ボタンを押下します。



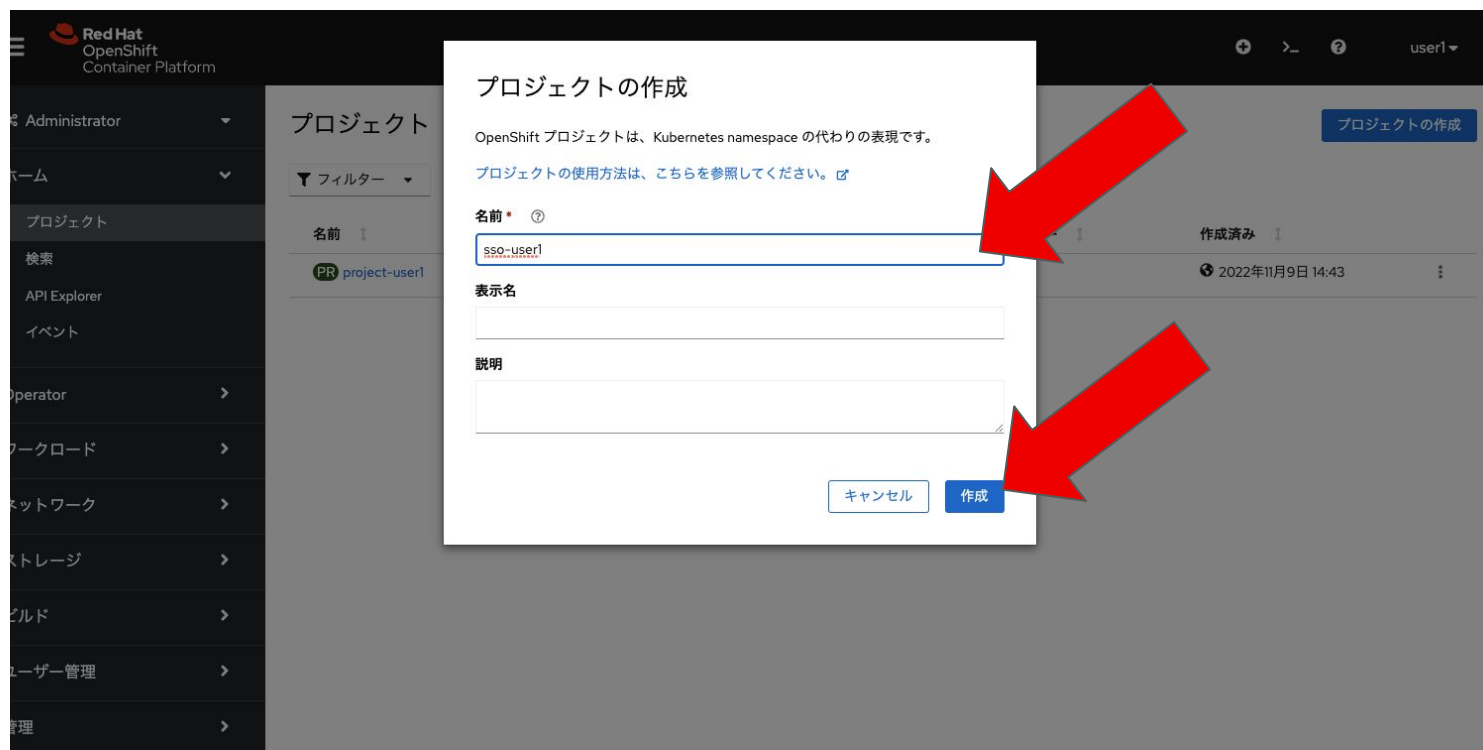
The screenshot shows the Red Hat OpenShift Container Platform interface. The left sidebar contains the navigation menu with 'Administrator', 'ホーム' (Home), 'プロジェクト' (Project), '検索' (Search), 'API Explorer', and 'イベント' (Events). The 'プロジェクト' section is expanded, showing 'Operator', 'ワークロード' (Workload), 'ネットワーク' (Network), and 'ストレージ' (Storage). The main content area is titled 'プロジェクト' and features a search bar and a table of projects. A red arrow points to the 'プロジェクトの作成' (Create Project) button in the top right corner.

名前	表示名	ステータス	リクエスター	作成済み
PR project-user1	表示名なし	Active	user1	2022年11月9日 14:43

## X-2.プロジェクトの作成

プロジェクトの作成画面が表示されます。

名前の欄に「sso-<ユーザ名(※1)>」を入力し、「作成」ボタンを押下します。



※1 例: sso-user1

ユーザ名は、OpenShiftにログインしたユーザIDです。

ユーザIDは、ブラウザの右上(例:user1)にも表示されています。

OpenShift(k8s)では、プロジェクト名(名前空間)は、1つのクラスターで一意的である必要があるため、今回はユーザ毎にプロジェクト名を分けています。

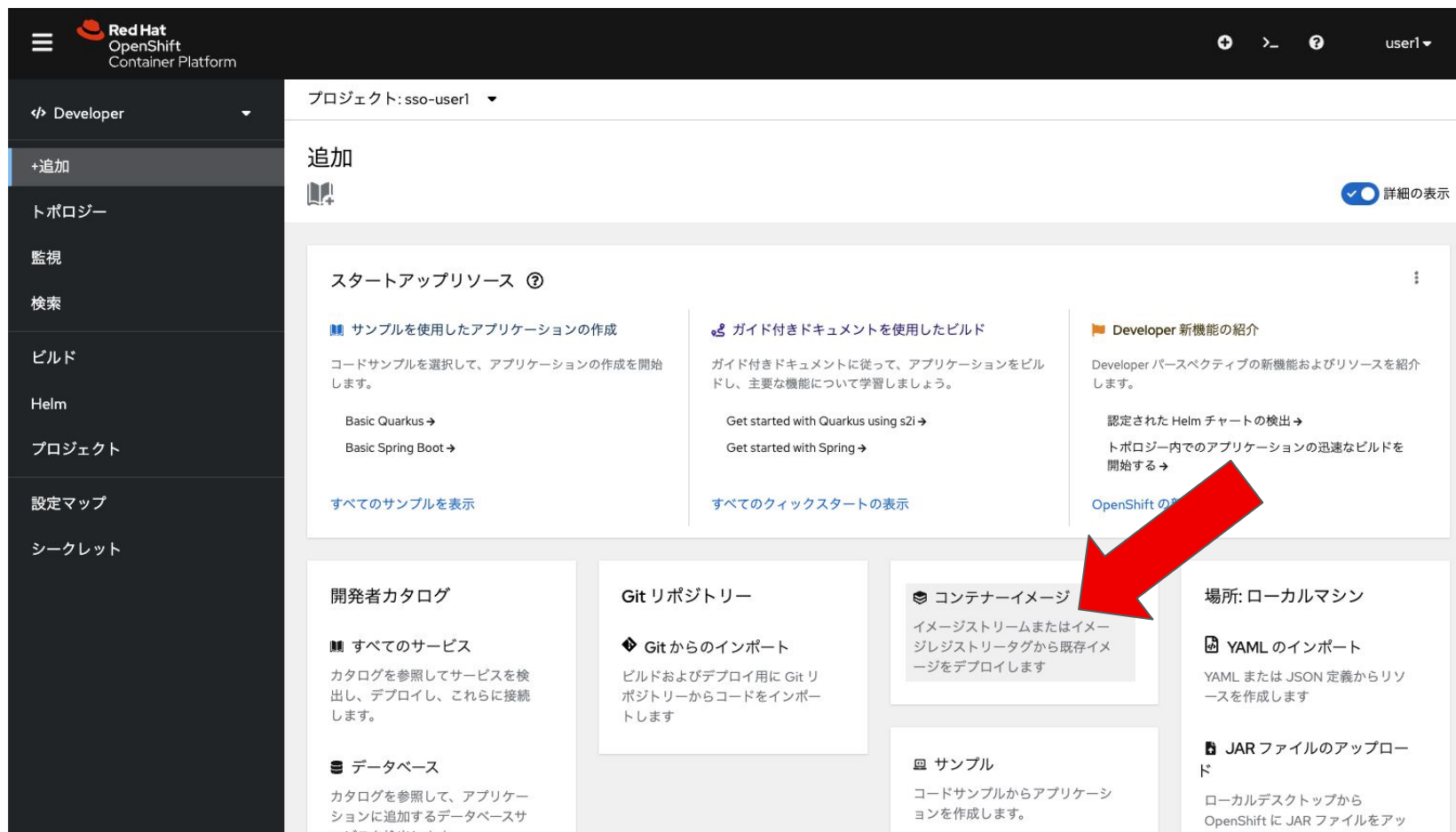


## X-3. コンテナイメージのデプロイ

OpenShiftには、プロジェクト内にアプリを追加できるGUIがあります。(CUIでも同等の操作が可能)

左ペインを[Administrator]から[Developer]に切り替えて、[+追加]を選択します。

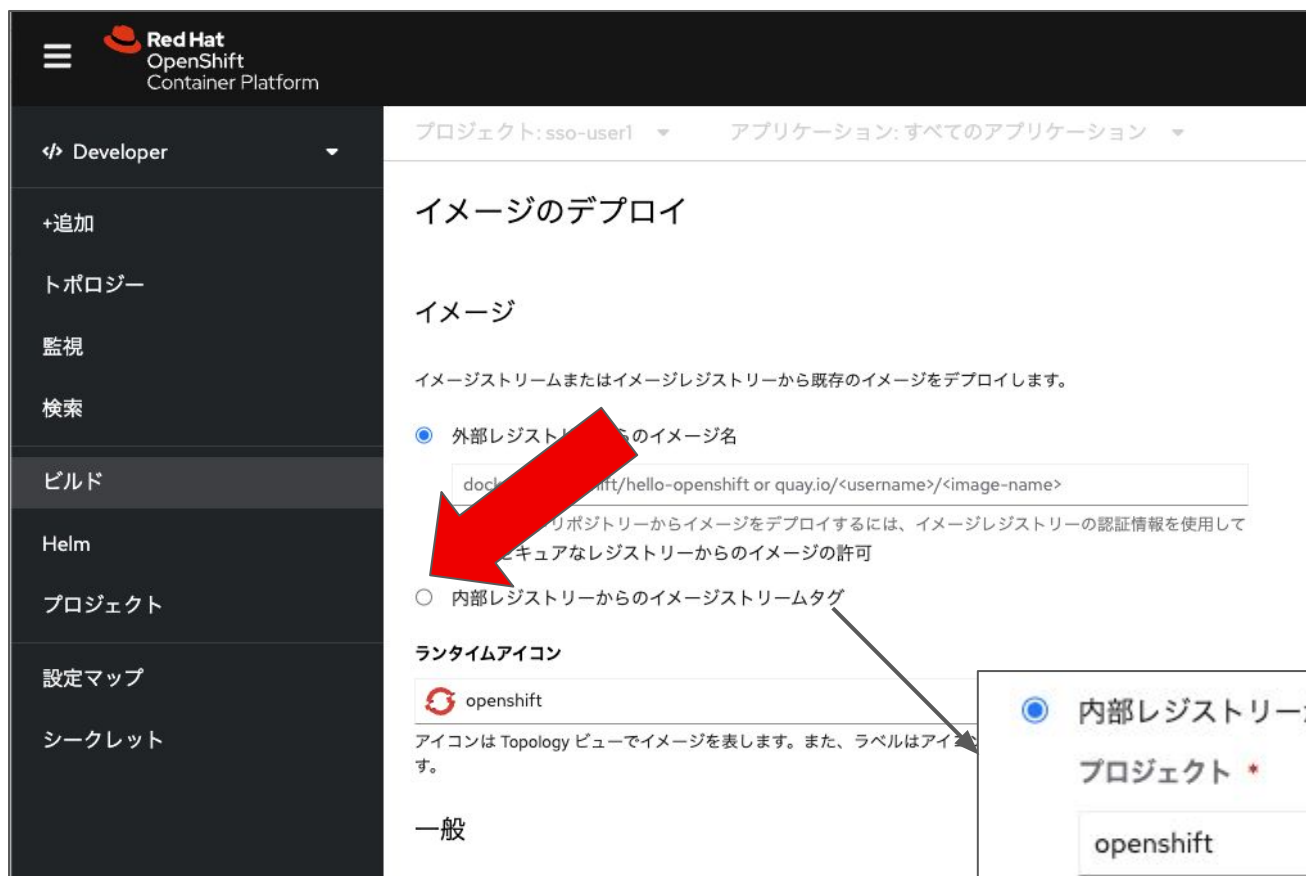
追加の画面が表示後、「コンテナイメージ」を選択します。



## X-3. コンテナイメージのデプロイ

イメージのデプロイ画面に遷移します。

「内部レジストリーからのイメージストリームタグ」のラジオボタンを押下します。



「内部レジストリーからのイメージストリームタグ」の各プルダウンで、以下を選択します。

- ・ プロジェクト: openshift
- ・ イメージストリーム: redhat-sso73-openshift
- ・ タグ: latest

## X-3. コンテナイメージのデプロイ

「内部レジストリーからのイメージストリームタグ」を選択後、アプリケーション名などが自動で設定されます。

ServiceやRouteの作成も同時に実施されます。  
画面左下の「作成」を押下してください。

Red Hat OpenShift Container Platform

プロジェクト: sso-user1 アプリケーション: すべてのアプリケーション

### イメージのデプロイ

イメージ

イメージストリームまたはイメージレジストリーから既存のイメージをデプロイします。

☐ 外部レジストリーからのイメージ名

☒ 内部レジストリーからのイメージストリームタグ

プロジェクト: openshift / イメージストリーム: redhat-ss073-openshift : タグ: latest

ランタイムアイコン

openshift

アイコンは Topology ビューでイメージを表します。また、ラベルはアイコンを定義するリソースに追加されます。

一般

アプリケーション名

redhat-ss073-openshift-app

リソースにラベルを付けるためにアプリケーションのグループに付けられた一意の名前。

名前

redhat-ss073-openshift

コンポーネントに指定する一意名で、関連リソースに名前を付けるのに使用します。

リソース

生成するリソースタイプの選択

☒ デプロイメント

apps/Deployment

Deployment は、Pod および ReplicaSet の宣言的な更新を有効にします。

☐ デプロイメント設定

apps.openshift.io/DeploymentConfig

DeploymentConfig は Pod のテンプレートを定義し、新規イメージまたは設定変更のデプロイを管理します。

詳細オプション

ターゲットポート

8080

トラフィックのターゲットポート。

☒ アプリケーションへのルートの作成

パブリック URL でのアプリケーションの公開

作成

## X-3. コンテナイメージのデプロイ

コンテナイメージの作成を押下すると、「トポロジービュー」の画面に遷移します。  
右上のアイコンから、Red Hat SSOのRouteのURLに遷移できます。

The screenshot displays the Red Hat OpenShift Container Platform console interface. The left sidebar shows the navigation menu with options like Developer, トポロジー (Topology), 監視 (Monitoring), 検索 (Search), ビルド (Build), Helm, プロジェクト (Project), 設定マップ (Settings Map), and シークレット (Secrets). The main area shows the 'Topology View' for the application 'redhat-ss073-openshift'. A red arrow points to the 'Route' icon in the application's resource list. The right sidebar provides details for the application, including a health check status, a pod count of 1, and various configuration parameters like namespace, labels, and update strategy.

Red Hat OpenShift Container Platform

プロジェクト: sso-user1    アプリケーション: すべてのアプリケーション    ショートカットの表示

トポロジー

redhat-ss073-openshift

ヘルスチェック

コンテナ redhat-ss073-openshift には、アプリケーションが正常に実行されていることを確認するためのヘルスチェックはありません。ヘルスチェックの追加

詳細    リソース    監視

1 Pod

名前: redhat-ss073-openshift    更新ストラテジー: RollingUpdate

namespace: NS sso-user1    最大の利用不可数: 25% / 1 Pod

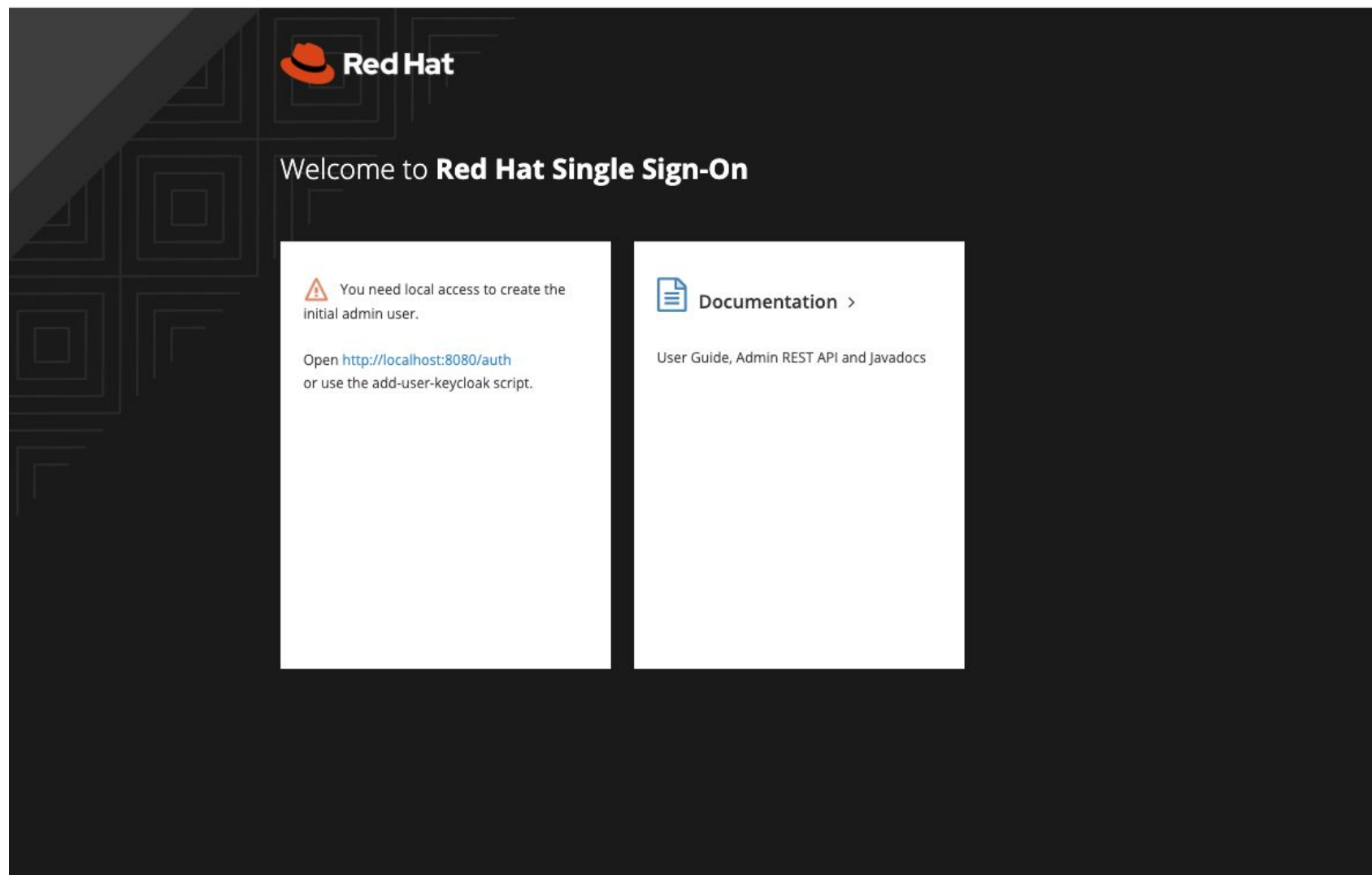
ラベル: app=redhat-ss073-openshift    最大サイズ数: 1 Pod を上回る 25%

app.kubernetes.io=redhat-ss0...    進行の期限 (秒)

## X-4. 端末での表示確認

端末のブラウザで以下の内容(Red Hat SSOのWelcomeページ)が表示されます。

今回は、Red Hat SSOのID、パスワードなどは設定しないため、ログインはできません。。



# Thank you

Red Hat is the world's leading provider of enterprise open source software solutions.

Award-winning support, training, and consulting services make

Red Hat a trusted adviser to the Fortune 500.

 [linkedin.com/company/red-hat](https://linkedin.com/company/red-hat)

 [youtube.com/user/RedHatVideos](https://youtube.com/user/RedHatVideos)

 [facebook.com/redhatinc](https://facebook.com/redhatinc)

 [twitter.com/RedHat](https://twitter.com/RedHat)