

Create a microservice via the console

# Table of Contents

Runtime tabs ..... 16

Demo ..... 22

The easiest way to configure the Pipeline for your project is via the [fabric8 developer console](#).

When you open the fabric8 console you should see a screen like this:

A **Team** is a kubernetes namespace running your development tools (like Jenkins, Nexus, JBoss Forge) and is associated with a number of environments (Testing, Staging, Production etc).

Click on the **Team Dashboard** which should take you to the Team Dashboard where you can create new apps or view your existing apps:

It?q=

applications currently available.

on

new application or try browsing the [Runtime](#) for t

It?q=

applications currently available.

on

new application or try browsing the [Runtime](#) for t

lt?q=

applications currently available.

on

new application or try browsing the [Runtime](#) for t

ations currently available.

application or try browsing the [Runtime](#) for this p

If you click the [Create Application](#) you get to the create wizard page:



+ In

f templates with the  
cret

Import an App which a

+ In

f templates with the  
cret

Import an App which a

+ In

f templates with the  
cret

Import an App which a

/forge/createProject

f templates with the  
cret

+ In  
Import an App which a

Then you get to pick what kind of project you wish to create and its name:

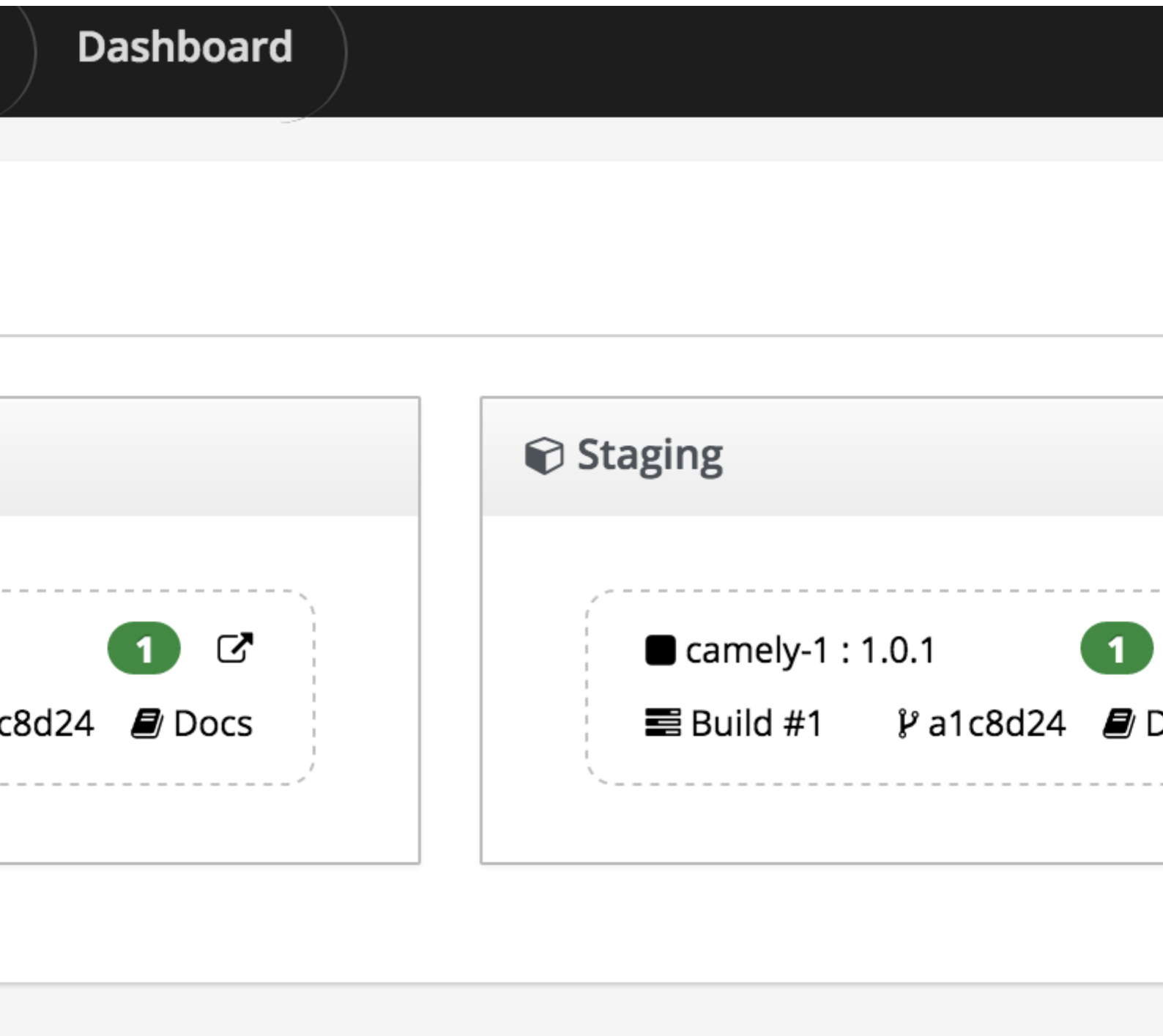
Then choose your [CD Pipeline](#):

If you choose **Copy pipeline to project** then the Jenkinsfile that defines the pipeline gets copied into your project's git repository so that you can easily edit it later on via a versioned source code change just like any other code change.

Now you will be taken to the **App Dashboard** where you can see all the environments and active pipelines along with recent commits on a single pane of glass. This is how it looks once the Canary release, Testing and Staging is complete; waiting for Promotion to Production

You can click on the **Proceed** button to promote to Production, or **Abort** to stop the pipeline.

You can easily switch between all your development tools (Gogs, Jenkins, Nexus etc) using the tool drop down menu at the top right of the screen:



[Pipelines >>](#)

## Staging

1



c8d24  Docs

■ camely-1 : 1.0.1

1

≡ Build #1

🔗 a1c8d24




D


[Pipelines >>](#)




1




c8d24


 Docs


 Staging

camely-1 : 1.0.1

1


 Build #1


 a1c8d24

 D

[Pipelines >>](#)

1




c8d24  Docs

## Staging

camely-1 : 1.0.1

1

 Build #1

 a1c8d24

 D

[Pipelines >>](#)

## Runtime tabs

The **Team** page has a **Runtime** tab that lets you browse the runtime of your development environment. Or from the home page you can click on an environment page to view its runtime.

The **Runtime** pages have a number of tabs that let you work with the various Kubernetes resources. We'll highlight the main ones you'll need to use:

## Replicas

The main tab to get a feel for what's running in your system is the **Replicas** tab which shows all the [replication controllers](#) or ReplicaSets on Kubernetes.

To scale up or down a controller to run more or less [pods](#) (containers) just increase or decrease the **Desired Replicas** value and hit **Save** and hey presto pods are created or destroyed.

×

Current Replicas	Desired Replicas	Labels
	<input type="text" value="3"/>	<span>elasticsearchController</span>
	<input type="text" value="1"/>	<span>fabric8MQ</span> <span>defaultMQGr</span>
	<input type="text" value="3"/>	<span>fabric8MQConsumer</span> <span>fab</span>
	<input type="text" value="1"/>	<span>fabric8MQCProducer</span> <span>fab</span>
	<input type="text" value="1"/>	<span>grafana</span> <span>management</span>
	<input type="text" value="1"/>	<span>influxdb</span> <span>management</span>

×

Current Replicas	Desired Replicas	Labels
	<input type="text" value="3"/>	<span>elasticsearchController</span>
	<input type="text" value="1"/>	<span>fabric8MQ</span> <span>defaultMQGr</span>
	<input type="text" value="3"/>	<span>fabric8MQConsumer</span> <span>fab</span>
	<input type="text" value="1"/>	<span>fabric8MQCProducer</span> <span>fab</span>
	<input type="text" value="1"/>	<span>grafana</span> <span>management</span>
	<input type="text" value="1"/>	<span>influxdb</span> <span>management</span>

×

Desired Replicas	Labels
<input type="text" value="3"/>	<span>elasticsearchController</span>
<input type="text" value="1"/>	<span>fabric8MQ</span> <span>defaultMQGroup</span> <span>java</span>
<input type="text" value="3"/>	<span>fabric8MQConsumer</span> <span>fabric8MQConsu</span>
<input type="text" value="1"/>	<span>fabric8MQCProducer</span> <span>fabric8MQProdu</span>
<input type="text" value="1"/>	<span>grafana</span> <span>management</span>
<input type="text" value="1"/>	<span>influxdb</span> <span>management</span>

Overview

The **Overview** tab gives you a feel for how all the various [services](#) and [replication controllers](#) interact:



# Demo

Here is a [video demonstrating how to create a microservice and then deploy and edit it via Continuous Delivery](#)