



## Red Hat Fuse 7.13

# 在 Springboot Standalone 上管理 Fuse

使用 Fuse 控制台管理 Fuse 应用程序



## Red Hat Fuse 7.13 在 Springboot Standalone 上管理 Fuse

---

使用 Fuse 控制台管理 Fuse 应用程序

## 法律通告

Copyright © 2024 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<sup>®</sup> is the registered trademark of Linus Torvalds in the United States and other countries.

Java<sup>®</sup> is a registered trademark of Oracle and/or its affiliates.

XFS<sup>®</sup> is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL<sup>®</sup> is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js<sup>®</sup> 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<sup>®</sup> 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.

## 摘要

当您部署 Fuse 应用程序时，您可以使用 Fuse 控制台来监控并与红帽 Fuse 集成进行交互。

---

# 目录

前言 .....	3
使开源包含更多 .....	4
第 1 章 关于 FUSE 控制台 .....	5
第 2 章 访问 SPRING BOOT 2.X 的 FUSE 控制台 .....	6
第 3 章 自定义 FUSE 控制台品牌 .....	8
第 4 章 保护 FUSE 控制台 .....	9
第 5 章 确保数据在 FUSE 控制台中正确显示 .....	10
第 6 章 连接到远程 FUSE 应用程序 .....	11
6.1. 连接到远程 JOLOKIA 代理 .....	11
6.2. 设置数据移动首选项 .....	11
6.3. 查看 JVM 运行时信息 .....	12
第 7 章 查看并管理 APACHE CAMEL 应用程序 .....	13
7.1. 启动、暂停或删除上下文 .....	13
7.2. 查看 CAMEL 应用程序详情 .....	13
7.3. 查看 CAMEL 路由列表并与之交互 .....	14
7.4. 调试路由 .....	15
第 8 章 查看并管理 JMX 域和 MBEANS .....	17
第 9 章 查看并管理 QUARTZ SCHEDULES .....	18
第 10 章 查看诊断 .....	19
第 11 章 查看线程 .....	20
附录 A. FUSE 控制台配置属性 .....	21



---

## 前言

Red Hat Fuse 提供两个企业监控工具，用于查看和管理 Fuse 集成：

- Fuse 控制台是一个基于 Web 的控制台，您可以从浏览器访问以监控和管理正在运行的 Fuse 容器。Fuse 控制台基于 Hawtio 开源软件(<https://hawtio.io/>)。本指南论述了如何使用 Fuse 控制台。
- Prometheus 为 Fuse 分发存储系统和集成级别指标。您可以使用 Grafana 等图形分析接口来查看和分析存储的历史数据。如需了解更多有关使用 Prometheus 的信息，请参阅 [Prometheus 文档](#)。

本指南的受众是 JBoss EAP 管理员上的红帽 Fuse。本指南假定您熟悉红帽 Fuse 平台、Apache Camel 以及您所在机构的处理要求。

## 使开源包含更多

红帽致力于替换我们的代码、文档和 Web 属性中存在问题的语言。我们从这四个术语开始：master、slave、黑名单和白名单。这些更改将在即将发行的几个发行本中逐渐实施。详情请查看我们的 [CTO Chris Wright 信息](#)。



## 第 1 章 关于 FUSE 控制台

红帽 Fuse 控制台是一个基于 Hawtio 开源软件的 Web 控制台。如需支持的浏览器列表，请转至 [支持的配置](#)。

Fuse 控制台提供了一个中央接口，用于检查和管理一个或多个部署的 Fuse 容器的详细信息。您还可以监控红帽 Fuse 和系统资源，执行更新，以及启动或停止服务。

当您在 OpenShift 上安装 Red Hat Fuse standalone 或使用 Fuse 时，可以使用 Fuse 控制台。您可以在 Fuse 控制台中查看和管理的集成取决于正在运行的插件。可能的插件包括：

- Camel
- JMX
- OSGI
- Runtime
- 日志

## 第 2 章 访问 SPRING BOOT 2.X 的 FUSE 控制台

您可以访问独立 Fuse Spring Boot 2.x 发行版的 Fuse 控制台。

### 流程

1. 在 Fuse 应用程序的 **pom.xml** 文件中添加以下依赖项：

```
<dependency>
  <groupId>io.hawt</groupId>
  <artifactId>hawtio-springboot</artifactId>
</dependency>
```

请注意，您不需要指定确切的版本，因为它是由 Maven BOM 提供的。

2. 编辑 **src/main/resources/application.properties** 文件：

- a. 设置以下属性：

- **management.endpoints.web.exposure.include=hawtio,jolokia**
- **hawtio.authenticationEnabled=false**
- **management.endpoint.hawtio.enabled=true**
- **management.endpoint.jolokia.enabled=true**

- b. （可选）设置 **management.endpoints.web.base-path** 属性。

默认情况下，对于 Spring Boot 2.x，Fuse Console 的 URL 包括管理端点的上下文路径 (**/actuator**)。例如：

<http://localhost:10001/actuator/hawtio/index.html>

要更改此默认 URL，例如指定 <http://localhost:10001/hawtio>，请设置 **management.endpoints.web.base-path** 属性，如下所示：

```
management.endpoints.web.base-path=/
```

您的 **application.properties** 设置应类似以下示例：

```
# ports

server.port=8080

management.server.port=10001

# enable management endpoints for healthchecks and hawtio

management.endpoints.enabled-by-default = false

management.endpoint.hawtio.enabled = true

management.endpoint.jolokia.enabled = true

management.endpoints.health.enabled = true
```

```
management.health.defaults.enabled=false

camel.health.enabled=false

camel.health.indicator.enabled=true

management.endpoints.web.exposure.include=hawtio,jolokia

hawtio.authenticationEnabled=false

# change the URL so that it does not include the actuator folder

management.endpoints.web.base-path=/
```



### 注意

默认情况下，Spring Boot 上 Fuse 控制台的身份验证被禁用。另外，您可以通过编写特定于 Fuse 控制台分布的代码来启用身份验证。以下是可用于指导的示例：

<https://github.com/hawtio/hawtio/tree/master/examples/springboot-authentication>

### 3. 运行 Fuse 应用程序：

```
mvn spring-boot:run
```

### 4. 要确定 Fuse 控制台 URL 的端口号，请通过查看 `src/main/resources/application.properties` 文件中设置的值来获取 `management.server.port` 值。例如：

```
management.server.port = 10001
```

### 5. 要在浏览器中打开 Fuse 控制台，请使用以下 URL 语法，其中 `nnnnn` 是 `management.server.port` 属性的值：

<http://localhost:nnnnn/actuator/hawtio>

例如，如果 `management.server.port` 属性值为 **10001**，且您没有设置 `management.endpoints.web.base-path` 属性，则 URL 为：

<http://localhost:10001/actuator/hawtio/index.html>

## 第 3 章 自定义 FUSE 控制台品牌

您可以通过在 Spring Boot 独立应用程序的 Fuse 中添加 **hawtconfig.json** 文件来自定义 Fuse 控制台品牌信息，如标题、徽标和登录页面信息。

### 流程

1. 在 Spring Boot standalone 应用的 **src/main/webapp** 目录中的本地 Fuse 中创建一个名为 **hawtconfig.json** 的 JSON 文件。
2. 在您选择的编辑器中打开 **src/main/webapp/hawtconfig.json**，然后添加以下内容：

```
{
  "branding": {
    "appName": "Red Hat Fuse Console",
    "appLogoUrl": "img/Logo-Red_Hat-Fuse-A-Reverse-RGB.png",
    "companyLogoUrl": "img/Logo-RedHat-A-Reverse-RGB.png"
  },
  "login": {
    "description": "",
    "links": []
  },
  "about": {
    "title": "Red Hat Fuse Console",
    "productInfo": [],
    "additionalInfo": "",
    "copyright": "",
    "imgSrc": "img/Logo-RedHat-A-Reverse-RGB.png"
  },
  "disabledRoutes": [
    "/camel/source",
    "/diagnostics",
    "/jvm/discover",
    "/jvm/local"
  ]
}
```

3. 更改 [表 A.1 “Fuse 控制台配置属性”](#) 中列出的配置属性的值。
4. 保存您的更改。
5. 使用以下命令在 Spring Boot 上运行 Fuse：

```
mvn spring-boot:run
```

6. 在网页浏览器中，使用此 URL 打开 Fuse 控制台：  
<http://localhost:10001/actuator/hawtio/index.html>



### 注意

如果您已在 Web 浏览器中运行 Fuse 控制台，品牌将存储在浏览器的本地存储中。要使用新的品牌设置，您必须清除浏览器的本地存储。

## 第 4 章 保护 FUSE 控制台

在 Spring Boot 上保护 Fuse 控制台：

- **在部署到 AWS 时禁用 Fuse Console 的代理 servlet**  
如果要将独立 Fuse 应用程序部署到 Amazon Web Services (AWS)，您应该通过将 **hawtio.disableProxy** 系统属性设置为 **true** 来禁用 Fuse 控制台的代理 servlet。



### 注意

当您禁用 Fuse Console 代理 servlet 时，Fuse Console 的 **Connect** 选项卡被禁用，您无法从 Fuse 控制台连接到其他 JVM。如果要在 AWS 上部署多个 Fuse 应用程序，您必须为每个应用程序部署 Fuse 控制台。

- **将 HTTPS 设置为所需的协议**  
您可以使用 **hawtio.http.strictTransportSecurity** 属性要求 Web 浏览器使用安全 HTTPS 协议来访问 Fuse 控制台。此属性指定尝试使用 HTTP 访问 Fuse 控制台的 Web 浏览器必须自动转换请求以使用 HTTPS。
- **使用公钥保护响应**  
您可以使用 **hawtio.http.publicKeyPins** 属性来保护 HTTPS 协议，方法是告诉 Web 浏览器将特定的加密公钥与 Fuse 控制台关联，以减少对伪证书进行“man-in-the-middle”攻击的风险。

### 流程

1. 设置 **hawtio.http.strictTransportSecurity** 和 **hawtio.http.publicKeyPins** 属性，如下例所示：

```
public static void main(String[] args) {
    System.setProperty("hawtio.http.strictTransportSecurity", "max-age=31536000;
includeSubDomains; preload");
    System.setProperty("hawtio.http.publicKeyPins", "pin-
sha256=cUPcTAZWKaASuYWhhneDttWpY3oBAkE3h2+soZS7sWs"; max-age=5184000;
includeSubDomains");
    SpringApplication.run(YourSpringBootApplication.class, args);
}
```

2. （仅用于在 AWS 上部署）要禁用 Fuse 控制台的代理 servlet，请设置 **hawtio.disableProxy** 属性，如下例所示：

```
public static void main(String[] args) {
    System.setProperty("hawtio.disableProxy", "true");
}
```

### 其他资源

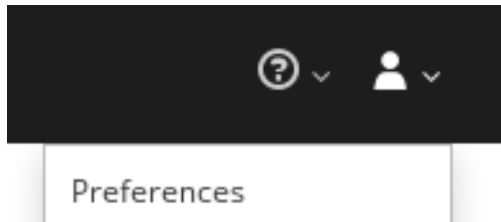
- 有关 **hawtio.http.strictTransportSecurity** 属性的语法的描述，请参阅 [HTTP Strict Transport Security \(HSTS\) 响应标头的描述页面](#)。
- 有关 **hawtio.http.publicKeyPins** 属性的语法的描述，包括如何提取 Base64 编码公钥的说明，请参阅 [HTTP 公钥 Pinning 响应标头](#) 的描述页面。

## 第 5 章 确保数据在 FUSE 控制台中正确显示

如果在 Fuse 控制台中显示队列和连接缺少队列、缺少连接或显示不一致的图标，请调整 Jolokia 集合大小参数，用于指定响应中 Jolokia marshals 中的最大元素数。

### 流程

1. 在 Fuse 控制台右上角，单击用户图标，然后单击 **Preferences**。



2. 增加 **Maximum collection size** 选项的值（默认值为 50,000）。
3. 单击 **Close**。

## 第 6 章 连接到远程 FUSE 应用程序

Fuse 控制台使用 Jolokia，这是一种基于代理的 Java 管理扩展(JMX)的方法，要求在客户端上安装额外的软件（代理）。默认情况下，红帽 Fuse 包括一个 jolokia 代理。

使用独立 Fuse 控制台发行版本，您可以连接到已在其中运行的 jolokia 代理(<https://jolokia.org/>)的远程集成。如果您要连接的进程没有 jolokia 代理，请参阅 jolokia 文档(<http://jolokia.org/agent.html>)。

### 流程

Fuse Console 的代理 servlet 使用白名单主机保护，默认情况下，Fuse 控制台只能连接到 localhost。如果要将 Fuse 控制台连接到其他远程 Fuse 实例，您需要在 Spring Boot 应用程序的 `main ()` 方法中配置 `hawtio.proxyWhitelist` 系统属性：

```
System.setProperty("hawtio.proxyWhitelist", "localhost, 127.0.0.1, myhost1, myhost2, myhost3");
```

### 6.1. 连接到远程 JOLOKIA 代理

开始之前，您需要知道远程 Jolokia 代理的连接详情（主机名、端口和路径）。

Spring Boot 上 Jolokia 代理的默认连接 URL 是 <http://<host>:8080/jolokia>

作为系统管理员，您可以更改这个默认值。

通常，远程连接到 Jolokia 代理的 URL 是打开 Fuse Console 加 `/jolokia` 的 URL。例如，如果打开 Fuse 控制台的 URL 是 <http://<host>:1234/hawtio>，则远程连接的 URL 可能是 <http://<host>:1234/hawtio/jolokia>。

要连接到远程 Jolokia 实例，以便您可以检查其 JVM：

1. 点 **Connect** 选项卡。
2. 单击 **Remote** 选项卡，然后单击 **Add connection**。
3. 键入 **名称**、**Scheme** (HTTP 或 HTTPS) 和 **主机名**。
4. 单击 **Test Connection**。
5. 点击 **Add**。



#### 注意

Fuse 控制台会自动探测 localhost 和 127.0.0.1 以外的本地网络接口，并将它们添加到白名单中。因此，您不需要手动将本地机器的地址注册到白名单中。

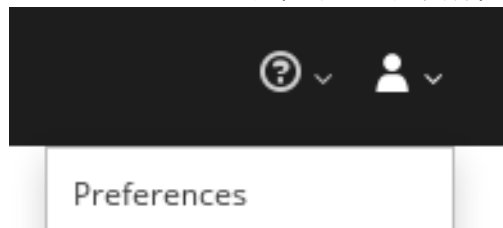
### 6.2. 设置数据移动首选项

您可以更改以下 Jolokia 首选项，例如，如果要更频繁地刷新 Fuse Console 中显示的数据。请注意，增加数据更新的频率会影响网络流量，并增加对服务器的请求数。

- **更新率** - 轮询到 Jolokia 以获取 JMX 数据之间的周期（默认为 5 秒）。
- **最大深度** - Jolokia 将在返回前将对象放入 JSON 的级别数（默认为 7）。
- **最大集合大小** - 响应中 Jolokia marshals marshals 的最大元素数（默认值为 50,000）。

更改这些设置的值：

1. 在 Fuse 控制台右上角，单击用户图标，然后单击 **Preferences**。



2. 编辑选项，然后单击 **Close**。

### 6.3. 查看 JVM 运行时信息

要查看 JVM 运行时信息，如系统属性、指标和线程，请单击 **Runtime** 选项卡。



## 第 7 章 查看并管理 APACHE CAMEL 应用程序

在 Fuse Console 的 **Camel** 选项卡中，您可以查看和管理 Apache Camel 上下文、路由和依赖项。

您可以查看以下详情：

- 所有正在运行的 Camel 上下文列表
- 每个 Camel 上下文（如 Camel 版本号和运行时静态）的详细信息
- 每个 Camel 应用程序及其运行时统计中的所有路由列表
- 正在运行的路由的图形表示以及实时指标

您还可以通过以下方法与 Camel 应用程序交互：

- 启动和暂停上下文
- 管理所有 Camel 应用程序及其路由的生命周期，以便您可以重启、停止、暂停、恢复等。
- 运行路由的实时追踪和调试
- 浏览消息并将其发送到 Camel 端点

### 前提条件

只有在连接到使用一个或多个 **Camel** 路由的容器时，Camel 选项卡才可用。

### 7.1. 启动、暂停或删除上下文

1. 在 **Camel** 选项卡的树视图中，单击 **Camel Contexts**。
2. 选中列表中一个或多个上下文旁边的框。
3. 点 **Start** 或 **Suspend**。
4. 删除上下文：
  - a. 停止上下文。
  - b. 单击 ellipse 图标，然后从下拉菜单中选择 **Delete**。



#### 注意

当您删除上下文时，您可以从部署的应用程序中删除它。

### 7.2. 查看 CAMEL 应用程序详情

1. 在 **Camel** 选项卡的树视图中，点 Camel 应用程序。
2. 要查看应用属性和值的列表，请单击 **Attributes**。
3. 要查看应用程序属性的图形表示，请点 **Chart**，然后点 **Edit** 以选择要在图表中看到的属性。
4. 要查看动态和被阻止交换，请点击 **Exchanges**。

5. 要查看应用端点，请单击 **Endpoints**。您可以根据 **URL**、**Route ID** 和 **方向** 来过滤列表。
6. 要查看、启用和禁用与 Camel 内置类型转换机制相关的统计信息，用于将消息正文和消息标头转换为不同的类型，请单击 **Type Converters**。
7. 要查看和执行 JMX 操作，如从 XML 添加或更新路由或查找 classpath 中的所有 Camel 组件，请点 **Operations**。

## 7.3. 查看 CAMEL 路由列表并与之交互

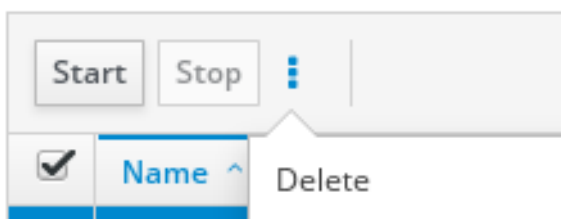
1. 查看路由列表：
  - a. 点 **Camel** 选项卡。
  - b. 在树视图中，点应用程序的路由文件夹：

### Routes

<input type="checkbox"/>	Name ^	State
<input type="checkbox"/>	_route1	Started
<input type="checkbox"/>	_route2	Started

2. 启动、停止或删除一个或多个路由：
  - a. 选中列表中一个或多个路由旁边的框。
  - b. 点 **Start** 或 **Stop**。
  - c. 要删除路由，您必须首先停止它。然后，单击 ellipse 图标，然后从下拉菜单中选择 **删除**。

### Routes



#### 注意

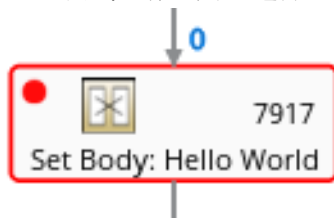
- 删除路由时，您可以将其从部署的应用程序中删除。
- 您还可以在树视图中选择特定的路由，然后单击右上角的菜单来启动、停止或删除它。

3. 要查看路由的图形图，请点 **Route 图表**。
4. 要查看动态和被阻止交换，请点击 **Exchanges**。

5. 要查看端点，请点 **Endpoints**。您可以根据 URL、Route ID 和方向过滤列表。
6. 点 **Type Converters** 查看、启用和禁用与 Camel 内置类型转换机制相关的统计信息，该机制用于将消息正文和消息标头转换为不同的类型。
7. 与特定路由交互：
  - a. 在 **Camel** 选项卡的树视图中，选择一个路由。
  - b. 要查看路由属性和值的列表，请单击 **Attributes**。
  - c. 要查看路由属性的图形表示，请点 **Chart**。您可以点 **Edit** 来选择要在图表中看到的属性。
  - d. 要查看动态和被阻止交换，请点击 **Exchanges**。
  - e. 点 **Operations** 查看并在路由上执行 JMX 操作，如将路由转储为 XML 或获取路由的 Camel ID 值。
8. 通过路由追踪信息：
  - a. 在 **Camel** 选项卡的树视图中，选择一个路由。
  - b. 选择 **Trace**，然后单击 **Start tracing**。
9. 将信息发送到路由：
  - a. 在 **Camel** 选项卡的树视图中，打开上下文的端点文件夹，然后选择端点。
  - b. 单击 **Send** 子选项卡。
  - c. 以 JSON 或 XML 格式配置消息。
  - d. 单击 **Send**。
  - e. 返回到路由的 **Trace** 选项卡，以查看消息通过路由流。

## 7.4. 调试路由

1. 在 **Camel** 选项卡的树视图中，选择一个路由。
2. 选择 **Debug**，然后单击 **Start debugging**。
3. 要添加断点，请在图中选择一个节点，然后单击 **Add breakpoint**。节点中会出现一个红色的点：



节点添加到断点列表中：

Breakpoints	
setBody1	×
log1	×

4. 单击向下箭头，以进入下一节点或 **Play** 按钮，以恢复运行路由。
5. 单击 **暂停** 按钮，以暂停路由的所有线程。
6. **完成后点 Stop debug**。所有断点都被清除。

## 第 8 章 查看并管理 JMX 域和 MBEANS

Java 管理扩展(JMX)是一种 Java 技术，允许您在运行时动态管理资源（服务、设备和应用程序）。资源由名为 MBeans（用于 Managed Bean）的对象表示。您可以在资源创建、实施或安装后立即管理和监控资源。

通过 Fuse Console 上的 JMX 插件，您可以查看和管理 JMX 域和 MBeans。您可以查看 MBean 属性、运行命令和创建显示 MBeans 统计信息的图表。

**JMX** 选项卡提供活动 JMX 域和以文件夹组织的 MBeans 的树状视图。您可以在 MBeans 上查看详情和执行命令。

### 流程

1. 查看并编辑 MBean 属性：
  - a. 在树视图中，选择一个 MBean。
  - b. 单击 **Attributes** 选项卡。
  - c. 点一个属性查看其详情。
2. 执行操作：
  - a. 在树视图中，选择一个 MBean。
  - b. 单击 **Operations** 选项卡，展开列出的操作之一。
  - c. 单击 **Execute** 以运行该操作。
3. 查看图表：
  - a. 在树视图中，选择一个项目。
  - b. 点 **Chart** 选项卡。

## 第 9 章 查看并管理 QUARTZ SCHEDULES

quartz (<http://www.quartz-scheduler.org/>)是一个功能丰富的开源作业调度库，您可以在大多数 Java 应用程序内集成。您可以使用 Quartz 为执行作业创建简单或复杂的计划。作业被定义为一个标准 Java 组件，您可以几乎执行该组件来对其进行编程。

如果您的 Camel 路由部署了 **camel-quartz2** 组件，Fuse 控制台会显示 **Quartz** 选项卡。请注意，您可以通过 JMX 树视图访问 Quartz mbeans。

### 流程

1. 在 Fuse 控制台中，单击 **Quartz** 选项卡。  
**Quartz** 页面包含 Quartz Schedulers 和 **Scheduler**、**Triggers** 和 **Jobs** 选项卡的树状视图。
2. 要暂停或启动调度程序，请单击 **调度程序选项卡上的按钮**。
3. 点 **Triggers** 选项卡查看决定作业何时运行的触发器。例如，触发器可以指定在一天（到 millisecond）、在指定天数或指定次数或特定时间重复启动作业。
  - 要过滤触发器列表，请从下拉列表中选择 **State**、**Group**、**Name** 或 **Type**。然后，您可以通过选择或键入 fill-on 字段来进一步过滤列表。
  - 要暂停、恢复、更新或手动触发触发器，请单击 **Action** 列中的选项。
4. 点 **Jobs** 选项卡查看正在运行的作业列表。您可以根据表中的列对列表进行排序：**Group**、**Name**、**Durable**、**Recover**、**Job ClassName** 和 **Description**。

## 第 10 章 查看诊断

使用 **Diagnostics** 选项卡，通过 JVM 诊断和 HotspotDiagnostic 接口查看有关 JVM 的诊断信息。



### 注意

这个功能与 Java Mission Control (jmc) 或命令行工具 jcmd 中的 **Diagnostic Commands** 视图类似。在某些情况下，该插件将提供对应的 jcmd 命令。

### 流程

1. 要检索已加载类的实例数量以及它们所消耗的字节数，请单击 **Class Histogram**。如果操作重复，选项卡会显示自上次运行以来的区别。
2. 要查看 JVM 诊断标志设置，请单击 **JVM 标志**。
3. 对于正在运行的 JVM，您还可以修改标志设置。

### 其他资源

支持的 JVM 依赖于平台，如需更多信息，请访问以下源之一：

- <http://www.oracle.com/technetwork/java/vmoptions-jsp-140102.html>
- <http://openjdk.java.net/groups/hotspot/docs/RuntimeOverview.html>

## 第 11 章 查看线程

您可以查看和监控线程状态。

### 流程

1. 单击 **Runtime** 选项卡，然后单击 **Threads** 子选项卡。**Threads** 页面列出了每个线程的活跃线程和堆栈追踪详情。默认情况下，线程列表以降序 ID 顺序显示所有线程。
2. 要通过增加 ID 进行排序，请点 **ID** 列标签。
3. （可选）根据线程状态（如 **Blocked**）或线程名称过滤列表。
4. 要深入查看特定线程的详细信息，如锁定类名称和该线程的完整堆栈追踪，请在 **Actions** 列中点击 **More**。



## 附录 A. FUSE 控制台配置属性

默认情况下，Fuse Console 配置在 `hawtconfig.json` 文件中定义。您可以自定义 Fuse 控制台配置信息，如标题、徽标和登录页面信息。

表 A.1 “Fuse 控制台配置属性” 提供属性的描述，并列出了每个属性是否需要值。

表 A.1. Fuse 控制台配置属性

部分	属性名称	默认值	描述	必需？
关于	标题	Red Hat Fuse Management Console	在 Fuse 控制台的 About 页面中显示的标题。	必填
	productInfo	空值	在 Fuse 控制台的 About 页面中显示的产品信息。	选填
	additionalInfo	空值	在 Fuse 控制台的 About 页面中显示的任何其他信息。	选填
	版权	空值	在 Fuse 控制台 About 页面中显示的版权信息。	选填
	imgSrc	<b>img/Logo-RedHat-A-Reverse-RGB.png</b>	在 Fuse 控制台的 About 页面中出现的镜像。	必填
品牌	appName	Red Hat Fuse Management Console	应用程序的名称。此名称显示在 Fuse 控制台的标题栏中。	必填
	appLogoUrl	<b>img/Logo-Red_Hat-Fuse-A-Reverse-RGB.png</b>	在 Fuse Console <code>{navigation}</code> 栏中显示的应用程序徽标镜像文件的路径。该值可以是相对于 Hawtio 状态 URL 或绝对 URL 的路径。	必填

部分	属性名称	默认值	描述	必需?
	CSS		外部 CSS 风格表的 URL，可用于风格应用程序。它可以是相对于 Hawtio 状态 URL 的路径，也可以是一个绝对 URL。	选填
	companyLogoUrl	<b>img/Logo-RedHat-A-Reverse- RGB.png</b>	您的公司徽标图像文件的路径。	必填
	favicon		favicon 的 URL，通常显示在 Web 浏览器选项卡中。它可以是相对于 Hawtio 状态 URL 的路径，也可以是一个绝对 URL。	选填
login	description	<i>空值</i>	在 Fuse Console Login 页面中显示的描述性文本（例如 <a href="http://localhost:8181/hawtio">http://localhost:8181/hawtio</a> ）。	选填
	links	[ ]	指定 "url" 和 "text" 对的数组，以向用户获取更多信息或帮助的页面提供额外的链接。	选填
disabledRoutes	<i>none</i>	[ ]	在控制台中禁用特定路径（如插件）。不要更改此部分。OpenShift 以外的发行版不支持任何更改。	选填