



Red Hat Fuse 7.8

在 JBoss EAP 上安装

在 Red Hat JBoss Enterprise Application Platform 7.3.2 上安装 Red Hat Fuse 7.8

Red Hat Fuse 7.8 在 JBoss EAP 上安装

在 Red Hat JBoss Enterprise Application Platform 7.3.2 上安装 Red Hat Fuse 7.8

法律通告

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.

摘要

安装和管理流程，支持开发在 JBoss EAP 上运行的 Fuse 应用程序。

目录

前言	3
第 1 章 在 JBOSS EAP 上安装 FUSE	4
第 2 章 在 JBOSS EAP 服务器上启动 FUSE	6
2.1. 以独立模式启动 JBOSS EAP	6
2.2. 以域模式启动 JBOSS EAP	6
第 3 章 验证 JBOSS EAP 上的 FUSE 安装	7
第 4 章 将 FUSE 用户添加到 JBOSS EAP	8
第 5 章 停止 JBOSS EAP	9
第 6 章 在 JBOSS EAP 安装中修补 FUSE	10
6.1. JBOSS EAP 上 FUSE 的热修补	10
6.2. 在 JBOSS EAP 上安装 FUSE 修补程序	10
第 7 章 在本地设置 MAVEN	12
7.1. 准备设置 MAVEN	12
7.2. 将红帽软件仓库添加到 MAVEN	12
7.3. 使用本地 MAVEN 存储库	14
7.4. 使用环境变量或系统属性设置 MAVEN 镜像	14
7.5. 关于 MAVEN 工件和协调	15

前言

管理员可以在红帽 JBoss 企业应用平台上安装红帽 Fuse，以便编程人员能够开发在 JBoss EAP 上运行的 Fuse 应用程序。有关 JBoss EAP 的详情，请查看 [JBoss EAP 简介](#)。

要在 JBoss EAP 上安装 Fuse，请参阅：

- [第 1 章 在 JBoss EAP 上安装 Fuse](#)
- [第 2 章 在 JBoss EAP 服务器上启动 Fuse](#)
- [第 3 章 验证 JBoss EAP 上的 Fuse 安装](#)
- [第 4 章 将 Fuse 用户添加到 JBoss EAP](#)
- [第 5 章 停止 JBoss EAP](#)
- [第 6 章 在 JBoss EAP 安装中修补 Fuse](#)
- [第 7 章 在本地设置 Maven](#)

第 1 章 在 JBOSS EAP 上安装 FUSE

要在 JBoss EAP 上开发 Fuse 应用程序，请安装 JBoss EAP，然后安装 Fuse。

先决条件

- 红帽客户门户网站登录凭证。
- 已安装受支持的 Java 运行时版本。
[Red Hat Fuse 支持的](#) 运行时列出了在不同操作系统上支持的 Java 运行时。

流程

1. 安装 JBoss EAP 7.3.2:

- a. 访问红帽客户门户上的 [Enterprise Application Platform Software Downloads](#) 页面。
- b. 出现提示时，登录到您的帐户。
- c. 在版本 下拉菜单中，选择 **7.3**。
- d. 点 [Red Hat JBoss Enterprise Application Platform 7.3 Installer](#) 软件包的 **Download** 链接。
- e. 运行下载的安装程序。在以下示例中，将 **DOWNLOAD_LOCATION** 替换为您系统上的 JBoss EAP 安装程序的位置：

```
java -jar DOWNLOAD_LOCATION/jboss-eap-7.3.0-installer.jar
```

- f. 接受条款和条件。
 - g. 选择您首选的安装路径，由 **EAP_HOME** 代表 JBoss EAP 运行时。
 - h. 创建管理用户，并确保记下此管理用户的凭据。您需要他们登录 Fuse 管理控制台。
 - i. 接受剩余的屏幕上的默认设置。
- #### 2. 进入 [Enterprise Application Platform Software Downloads > Patches](#) 页面并下载 JBoss EAP 7.3 Update 02 文件。 有关更新文件的详情，请查看相应的 [发行注记](#)。
- #### 3. 安装 JBoss EAP 7.3 Update 02。例如，在 Red Hat Enterprise Linux 中使用以下命令：

```
bin/jboss-cli.sh "patch apply jboss-eap-7.3.x-patch.zip"
```

以下响应表示补丁已被成功安装：

```
{
  "outcome" : "success",
  "result" : {}
}
```

- #### 4. 检查 [Red Hat Fuse 支持的配置](#) 页面，了解与 Red Hat Fuse 的 JBoss EAP 补丁兼容性的任何备注或建议。如果相关，请安装任何其他 JBoss EAP 补丁。

5. 在 JBoss EAP 7.3.2 上安装 Fuse 7.8 :
 - a. 访问红帽客户门户上的 [Red Hat Fuse Software Downloads](#) 页面。
 - b. 出现提示时，登录到您的帐户。
 - c. 在版本 下拉菜单中，选择 **7.8.0**。
 - d. 在 EAP 安装程序软件包中，点 Red Hat Fuse 7.8.0 的 **Download** 链接。
 - e. 打开 shell 提示符（或 Windows 上的命令提示符）。
 - f. 进入 **EAP_HOME** 目录，这是全新的 Red Hat JBoss Enterprise Application Platform 安装的根目录。
 - g. 运行下载的安装程序。在以下示例中，将 **DOWNLOAD_LOCATION** 替换为系统上下载的 Fuse 安装程序的位置：

```
java -jar DOWNLOAD_LOCATION/fuse-eap-installer-7.8.0.fuse-780029-redhat-00001.jar
```

6. 安装程序在没有提示的情况下运行，并将其活动记录到屏幕上。

后续步骤

- 启动 JBoss EAP，验证 Fuse 正在运行，然后将 Fuse 用户添加到 JBoss EAP。下面的章节解释了如何执行这些任务。
- 可选，但建议：设置 Maven 以用于 Fuse 项目本地使用。这在 [第7章 在本地设置 Maven](#) 中解释。

其他资源

- 有关安装 JBoss EAP 的更多信息，请参阅 [JBoss EAP 7.3 安装指南](#)。
- 有关安装 JBoss EAP 补丁的详细信息，请参阅 [JBoss EAP 7.3 补丁和升级指南](#)。

第 2 章 在 JBOSS EAP 服务器上启动 FUSE

JBoss EAP 上的 Fuse 支持单机模式和域模式。本章解释了如何以单机模式或域模式启动服务器。

2.1. 以独立模式启动 JBOSS EAP

本节中的 Fuse 说明如何将 JBoss EAP 作为单机服务器启动。

先决条件

- 已安装 JBoss EAP 7.3.2。

流程

- 对于 Red Hat Enterprise Linux, 运行以下命令：
EAP_HOME/bin/standalone.sh
- 对于 Microsoft Windows Server, 运行以下命令：
EAP_HOME\bin\standalone.bat

2.2. 以域模式启动 JBOSS EAP

本节中的命令解释了如何在 JBoss EAP 上作为域服务器启动 Fuse。

先决条件

- 已安装 JBoss EAP 7.3.2。

流程

- 对于 Red Hat Enterprise Linux, 运行以下命令：
EAP_HOME/bin/domain.sh
- 对于 Microsoft Windows Server, 运行以下命令：
EAP_HOME\bin\domain.bat

其他资源

- 要获取您可以传递给启动脚本的参数列表, 请输入 **-h** 参数。
- 有关使用替代和更高级的方法启动和停止 JBoss 企业应用平台的更多信息, 请参阅 [Red Hat JBoss Enterprise Application Platform 配置指南](#)。

第 3 章 验证 JBOSS EAP 上的 FUSE 安装

在 JBoss EAP 上安装 Fuse 后，验证安装是否成功。

先决条件

- 已安装 JBoss EAP 7.3.2 上的 Fuse 7.8。
- JBoss EAP 服务器正在运行。

流程

1. 打开以下文件来检查是否已记录任何错误信息：
EAP_HOME/standalone/log/server.log
2. 在浏览器中，打开 [Fuse 管理控制台](#)。
3. 输入您在 JBoss EAP 安装过程中创建的管理员凭据进行登录。

结果

如果 Fuse 管理控制台运行并且您可以登录，Fuse 安装可以成功。

第 4 章 将 FUSE 用户添加到 JBOSS EAP

运行 JBoss EAP **add-user** 脚本，将 Fuse 用户添加到 JBoss EAP。

先决条件

- JBoss EAP 正在运行。

流程

1. 导航到 **EAP_HOME/bin**。
2. 运行 **add-user** 脚本。例如：
./add-user.sh
3. 响应提示以创建新用户：
 - **管理用户** 是 JBoss EAP 上的 Fuse 管理用户。
 - **应用程序用户** 是 JBoss EAP 上的 Fuse 非管理用户。

第 5 章 停止 JBOSS EAP

以下流程使用管理命令行界面(CLI)来停止 JBoss EAP。

先决条件

- JBoss EAP 7.3.2 正在运行。

流程

1. 运行以下命令启动管理 CLI :

```
$ EAP_HOME/bin/jboss-cli.sh
```

2. 运行 **connect** 命令连接到服务器 :

```
[disconnected /] connect
```

3. 运行 **shutdown** 命令停止服务器 :

```
[standalone@localhost:9999 /] shutdown
```

4. 运行 **quit** 命令关闭管理 CLI :

```
[standalone@localhost:9999 /] quit
```

替代方案

以下是停止 JBoss EAP 的另一种方式 :

1. 导航到运行 JBoss EAP 的终端。
2. 按 **Ctrl+C** 停止 JBoss 企业应用平台。

第 6 章 在 JBOSS EAP 安装中修补 FUSE

本章介绍了如何将 Fuse 修补程序应用到 JBoss EAP 安装的现有 Fuse 中。它包括以下主题：

- [第 6.1 节 “JBoss EAP 上 Fuse 的热修补”](#)
- [第 6.2 节 “在 JBoss EAP 上安装 Fuse 修补程序”](#)

升级 JBoss EAP

您还可以将 JBoss EAP 的底层版本升级到 Fuse 支持的另一版本，而无需在 JBoss EAP 上重新安装和重新部署 Fuse。如需了解更多详细信息，请参阅 [JBoss EAP 补丁和升级指南](#)。



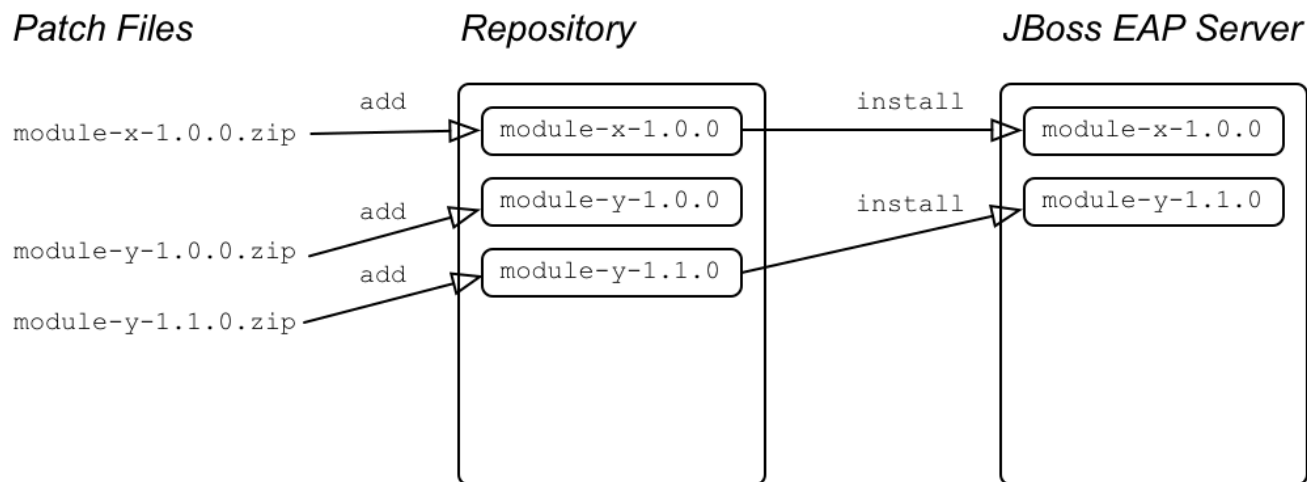
重要

您只能将 JBoss EAP 升级到在 [Fuse 支持的配置页面中](#) 支持的版本。

6.1. JBOSS EAP 上 FUSE 的热修补

Fuse 修补程序包含 JBoss EAP 安装的 Fuse 中特定文件的更新版本。它们通常仅包含一个或多个关键错误。热修补修补程序适用于您现有红帽 Fuse 分发的基础上，仅更新现有 Fuse 文件的子集。

在 JBoss EAP 上为 Fuse 应用补丁是一个两阶段的过程，首先将补丁文件添加到补丁存储库中，然后在 JBoss EAP 服务器中安装。下图显示了 JBoss EAP 上的 Fuse patching 流程概述：



补丁软件仓库

patch 存储库是 JBoss EAP 补丁上与 JBoss EAP 服务器相同的 JVM 上运行的 Fuse 的存放区域。当存储库中存在补丁时，这并不意味着它在 JBoss EAP 服务器中已安装。您必须首先将补丁添加到存储库，然后从存储库安装补丁到 JBoss EAP 服务器中。

fusepatch 工具

fusepatch 实用程序是用于在 JBoss EAP 上修补 Fuse 的命令行工具。在 EAP 软件包上安装 Fuse 后，JBoss EAP 服务器的 **bin** 目录中提供了 **fusepatch.sh** 脚本(Linux 和 UNIX)和 **fusepatch.bat** (Windows)脚本。

6.2. 在 JBOSS EAP 上安装 FUSE 修补程序

在现有 Fuse 安装之上必须安装 Fuse 热修补补丁。本节解释了如何在现有 Fuse 安装之上安装 hotfix 补丁 **fuse-eap-distro-VERSION.fuse-MODULE_ID.HOTFIX_ID.zip**，它已包含 **fuse-eap-distro-VERSION.fuse-MODULE_ID-redhat-BASE_ID**。

先决条件

- [第 6.1 节 “JBoss EAP 上 Fuse 的热修补”](#)。
- 下载红帽支持的热修补代码补丁 **.zip** 文件。
- 阅读带有 hotfix 补丁文件的 **readme.txt** 文件中的说明，以防需要执行任何其他步骤来安装该文件。
- 在应用补丁之前，在 JBoss EAP 安装上对 Fuse 进行完整备份。

流程

1. 将 hotfix 补丁文件复制到 **EAP_HOME** 目录中。
2. 确保已将正确的基础版本添加到您的补丁存储库中，并安装在 JBoss EAP 服务器上。
例如，如果一个基础模块 **fuse-eap-distro-7.8.0.fuse-sb2-780038-redhat-00001**，以检查存储库中安装的 **MODULE_ID** 和 **BASE_ID**，请输入以下命令：

```
bin/fusepatch.sh --query-repository
```

应返回以下响应：

```
fuse-eap-distro-7.8.0.fuse-sb2-780038-redhat-00001
```

为了检查 JBoss EAP 服务器上是否已安装相同的 ID，请输入以下命令：

```
bin/fusepatch.sh --query-server
```

应返回以下响应：

```
fuse-eap-distro-7.8.0.fuse-sb2-780038-redhat-00001
```

3. 根据一次性修补程序补丁文件，**fuse-eap-distro-7.7.0.fuse-770013.hf1.zip**，通过输入以下命令将其添加到您的仓库并将其与基础安装相关联：

```
bin/fusepatch.sh --add file:fuse-eap-distro-7.7.0.fuse-770013.hf1.zip --one-off fuse-eap-distro-7.8.0.fuse-sb2-780038-redhat-00001
```

4. 根据基础模块 **fuse-eap-distro-7.8.0.fuse-sb2-780038-redhat-00001**，将 JBoss EAP 服务器更新至最新版本：

```
bin/fusepatch.sh --update fuse-eap-distro-7.7.0.fuse-770013.hf1
```

5. 执行补丁说明中记录的任何安装后步骤。

其他资源

- 有关 **fusepatch** 命令的更多详细信息，请输入：

```
bin/fusepatch.sh --help
```

第 7 章 在本地设置 MAVEN

典型的 Fuse 应用程序开发使用 Maven 来构建和管理项目。

以下主题描述了如何在本地设置 Maven：

- [第 7.1 节 “准备设置 Maven”](#)
- [第 7.2 节 “将红帽软件仓库添加到 Maven”](#)
- [第 7.3 节 “使用本地 Maven 存储库”](#)
- [第 7.4 节 “使用环境变量或系统属性设置 Maven 镜像”](#)
- [第 7.5 节 “关于 Maven 工件和协调”](#)

7.1. 准备设置 MAVEN

Maven 是一个来自 Apache 的免费开源构建工具。通常，您可以使用 Maven 构建 Fuse 应用程序。

流程

1. 从 [Maven 下载页面](#) 下载最新版本的 Maven。
2. 确定您的系统已连接到互联网。
在构建项目时，默认行为是 Maven 搜索外部存储库并下载所需的工件。Maven 查找可通过互联网访问的存储库。

您可以更改此行为，以便 Maven 仅搜索位于本地网络上的存储库。也就是说，Maven 可以在离线模式下运行。在离线模式中，Maven 会在其本地存储库中查找工件。请参阅 [第 7.3 节 “使用本地 Maven 存储库”](#)。

7.2. 将红帽软件仓库添加到 MAVEN

要访问位于 Red Hat Maven 存储库中的工件，您需要将这些存储库添加到 Maven 的 **settings.xml** 文件中。Maven 在用户主目录的 **.m2** 目录中查找 **settings.xml** 文件。如果没有用户指定的 **settings.xml** 文件，Maven 将使用 **M2_HOME/conf/settings.xml** 中的系统级 settings.xml 文件。

前提条件

您知道要在其中添加红帽软件仓库的 **settings.xml** 文件的位置。

流程

在 **settings.xml** 文件中，为红帽 软件仓库 添加软件仓库元素，如下例所示：

```
<?xml version="1.0"?>
<settings>

<profiles>
  <profile>
    <id>extra-repos</id>
    <activation>
      <activeByDefault>true</activeByDefault>
    </activation>
```



```
<repositories>
<repository>
  <id>redhat-ga-repository</id>
  <url>https://maven.repository.redhat.com/ga</url>
  <releases>
    <enabled>true</enabled>
  </releases>
  <snapshots>
    <enabled>>false</enabled>
  </snapshots>
</repository>
<repository>
  <id>redhat-ea-repository</id>
  <url>https://maven.repository.redhat.com/earlyaccess/all</url>
  <releases>
    <enabled>true</enabled>
  </releases>
  <snapshots>
    <enabled>>false</enabled>
  </snapshots>
</repository>
<repository>
  <id>jboss-public</id>
  <name>JBoss Public Repository Group</name>
  <url>https://repository.jboss.org/nexus/content/groups/public/</url>
</repository>
</repositories>
<pluginRepositories>
<pluginRepository>
  <id>redhat-ga-repository</id>
  <url>https://maven.repository.redhat.com/ga</url>
  <releases>
    <enabled>true</enabled>
  </releases>
  <snapshots>
    <enabled>>false</enabled>
  </snapshots>
</pluginRepository>
<pluginRepository>
  <id>redhat-ea-repository</id>
  <url>https://maven.repository.redhat.com/earlyaccess/all</url>
  <releases>
    <enabled>true</enabled>
  </releases>
  <snapshots>
    <enabled>>false</enabled>
  </snapshots>
</pluginRepository>
<pluginRepository>
  <id>jboss-public</id>
  <name>JBoss Public Repository Group</name>
  <url>https://repository.jboss.org/nexus/content/groups/public</url>
</pluginRepository>
</pluginRepositories>
</profile>
</profiles>
```

```

<activeProfiles>
  <activeProfile>extra-repos</activeProfile>
</activeProfiles>

</settings>

```

7.3. 使用本地 MAVEN 存储库

如果您在没有互联网连接的情况下运行 Apache Karaf 容器，并且您需要部署一个具有离线不可用依赖项的应用程序，您可以使用 Maven 依赖项插件将应用的依赖项下载到 Maven 离线存储库中。然后，您可以将此自定义 Maven 离线存储库分发到没有互联网连接的机器。

流程

1. 在包含 **pom.xml** 文件的项目目录中，运行以下命令来下载 Maven 项目的存储库，如下所示：

```

mvn org.apache.maven.plugins:maven-dependency-plugin:3.1.0:go-offline -
Dmaven.repo.local=/tmp/my-project

```

在本例中，构建项目所需的 Maven 依赖项和插件将下载到 **/tmp/my-project** 目录中。

2. 将此自定义 Maven 离线存储库在内部分发到没有互联网连接的任何机器。

7.4. 使用环境变量或系统属性设置 MAVEN 镜像

在运行应用程序时，您需要访问 Red Hat Maven 存储库中的工件。这些存储库添加到 Maven 的 **settings.xml** 文件中。Maven 检查 **settings.xml** 文件的以下位置：

- 查找指定的 url
- 如果没有找到 **`\${user.home}/.m2/settings.xml** 的查找
- 如果没有找到 **`\${maven.home}/conf/settings.xml** 的查找
- 如果未找到，则查找 **`\${M2_HOME}/conf/settings.xml**
- 如果没有找到位置，则会创建空的 **org.apache.maven.settings.Settings** 实例。

7.4.1. 关于 Maven 镜像

Maven 使用一组远程存储库来访问工件，这些工件目前在本地存储库中不可用。存储库列表几乎始终包含 Maven Central 存储库，但对于 Red Hat Fuse，它还包含 Maven 红帽存储库。在某些情况下，无法访问不同的远程存储库，您可以使用 Maven 镜像的机制。镜像将特定的存储库 URL 替换为不同的存储库 URL，因此搜索远程工件时的所有 HTTP 流量都可以定向到单个 URL。

7.4.2. 在 settings.xml 中添加 Maven 镜像

要设置 Maven 镜像，请在 Maven 的 **settings.xml** 中添加以下部分：

```

<mirror>
  <id>all</id>
  <mirrorOf>*</mirrorOf>

```

```
<url>http://host:port/path</url>
</mirror>
```

如果在 **settings.xml** 文件中找不到以上部分，则不会使用 mirror。要在不提供 XML 配置的情况下指定全局镜像，您可以使用系统属性或环境变量。

7.4.3. 使用环境变量或系统属性设置 Maven 镜像

要使用环境变量或系统属性设置 Maven 镜像，您可以添加：

- 名为 **MAVEN_MIRROR_URL** 的环境变量到 **bin/setenv** 文件
- 名为 **mavenMirrorUrl** 到 **etc/system.properties** 文件的系统属性

7.4.4. 使用 Maven 选项指定 Maven 镜像 url

要使用替代的 Maven 镜像 url，在环境变量或系统属性之外，在运行应用程序时使用以下 maven 选项：

- **-DmavenMirrorUrl=mirrorId::mirrorUrl**
for example, **-DmavenMirrorUrl=my-mirror::http://mirror.net/repository**
- **-DmavenMirrorUrl=mirrorUrl**
for example, **-DmavenMirrorUrl=http://mirror.net/repository**. 在本例中，<mirror> 的 <id> 只是一个镜像(mirror)。

7.5. 关于 MAVEN 工件和协调

在 Maven 构建系统中，基本构建块是一个 *工件*。构建后，工件的输出通常是一个存档，如 JAR 或 WAR 文件。

Maven 的一个关键方面是能够找到工件和管理它们之间的依赖关系。*Maven 协调*是一组用于标识特定工件位置的值。基本协调有三个值，格式为：

groupId:artifactId:version

有时，Maven 通过 *打包值*或使用*打包*值和 *分类器*值增加一个基本的协调。*Maven 协调*可以具有以下形式之一：

```
groupId:artifactId:version
groupId:artifactId:packaging:version
groupId:artifactId:packaging:classifier:version
```

以下是值的描述：

groupId

定义工件名称的范围。您通常使用所有或部分软件包名称作为组 ID。例如，**org.fusesource.example**。

artifactId

定义相对于组 ID 的工件名称。

version

指定工件的版本。版本号最多可有四个部分：**n.n.n.n**，其中版本号的最后一部分可以包含非数字字符。例如，**1.0-SNAPSHOT** 的最后一部分是字母数字字符子字符串 **0-SNAPSHOT**。

打包

定义构建项目时生成的打包实体。对于 OSGi 项目，打包是 **捆绑包**。默认值为 **jar**。

classifier

可让您区分从同一 POM 构建但具有不同内容的工件。

工件的 POM 文件中的元素定义工件的组 ID、工件 ID、打包和版本，如下所示：

```
<project ... >
...
<groupId>org.fusesource.example</groupId>
<artifactId>bundle-demo</artifactId>
<packaging>bundle</packaging>
<version>1.0-SNAPSHOT</version>
...
</project>
```

要定义对上述工件的依赖项，您可以在 POM 文件中添加以下 dependencies 元素：

```
<project ... >
...
<dependencies>
<dependency>
  <groupId>org.fusesource.example</groupId>
  <artifactId>bundle-demo</artifactId>
  <version>1.0-SNAPSHOT</version>
</dependency>
</dependencies>
...
</project>
```



注意

不需要在前面的依赖项中指定 **bundle** 软件包类型，因为捆绑包只是特定类型的 JAR 文件，**jar** 是默认的 Maven 软件包类型。但是，如果您需要在依赖项中明确指定打包类型，您可以使用 **type** 元素。