

Common Jersey Use Cases

中文标题【通用 Jersey 用户案例】

Glassfish 中基于 Servlet 的应用

如果你使用的是 Glassfish 作为你应用服务器，你不需要在你的引用中包含引用任何东西，所有你需要的都已经包含进去了。

你只需要定义 JAX-RS API 以便于你能够对你的应用进行编辑，使用 (provided) 依赖。

```
<dependency>
  <groupId>javax.ws.rs</groupId>
  <artifactId>javax.ws.rs-api</artifactId>
  <version>2.1</version>
  <scope>provided</scope>
</dependency>
```

如果你需要使用 Jersey 的一些特定特性，你需要基于你的 Jersey 目录进行添加。

```
<dependency>
  <groupId>org.glassfish.jersey.containers</groupId>
  <artifactId>jersey-container-servlet</artifactId>
  <version>2.27</version>
  <scope>provided</scope>
</dependency>
<!-- if you are using Jersey client specific features without the server side -->
<dependency>
  <groupId>org.glassfish.jersey.core</groupId>
  <artifactId>jersey-client</artifactId>
  <version>2.27</version>
  <scope>provided</scope>
</dependency>
```

基于Servlet 的服务器端应用

下面的依赖通常应用到应用服务器上 (servlet 容器)，同时这个应用服务器上并没有整合任何 JAX-RS 的实现。

因此，这个应用服务器需要包含有 JAX-RS API 和 Jersey 实现，同时部署到服务器上。

```
<dependency>
  <groupId>org.glassfish.jersey.containers</groupId>
  <!-- if your container implements Servlet API older than 3.0, use "jersey-container-servlet-core" -->
  <artifactId>jersey-container-servlet</artifactId>
  <version>2.27</version>
</dependency>
<!-- Required only when you are using JAX-RS Client -->
<dependency>
  <groupId>org.glassfish.jersey.core</groupId>
  <artifactId>jersey-client</artifactId>
  <version>2.27</version>
</dependency>
```

JDK 上的客户端应用

如应用是运行在 JDK 上的话，你只需要使用 JAX-RS 中的客户端部分就可以了，这个根据你使用的客户端有所调整。

这里有一系列的模块是可以供你使用的，例如 grizzly 或 apache 或 jetty connector (请参考下面的的依赖表格)。Jersey 客户端默认使用 JDK 进行运行 (使用的是 `URLConnection`)。

请参考 [Chapter 5, Client API](#) 获得更多的细节。

```
<dependency>
  <groupId>org.glassfish.jersey.core</groupId>
  <artifactId>jersey-client</artifactId>
  <version>2.27</version>
</dependency>
```

当前可用的连接器：

```
<dependency>
  <groupId>org.glassfish.jersey.connectors</groupId>
  <artifactId>jersey-grizzly-connector</artifactId>
  <version>2.27</version>
</dependency>

<dependency>
  <groupId>org.glassfish.jersey.connectors</groupId>
  <artifactId>jersey-apache-connector</artifactId>
  <version>2.27</version>
</dependency>

<dependency>
  <groupId>org.glassfish.jersey.connectors</groupId>
  <artifactId>jersey-jetty-connector</artifactId>
  <version>2.27</version>
</dependency>
```

服务器端应用支持的容器

基于 JAX-RS Servlet-based 部署的一部分标准，能运行在任何支持 Servlet 2.5 和更高标准的容器上。Jersey 提供支持程序化部署在下面的容器中：Grizzly 2（HTTP 和 Servlet），JDK Http server，Simple Http server 和 Jetty Http server。

本章节仅仅提供了需要 Maven 提供的依赖，有关更多的细节，请参考[Chapter4, Application Deployment and Runtime Environments](#)页面中的内容。

```
<dependency>
  <groupId>org.glassfish.jersey.containers</groupId>
  <artifactId>jersey-container-grizzly2-http</artifactId>
  <version>2.27</version>
</dependency>

<dependency>
  <groupId>org.glassfish.jersey.containers</groupId>
  <artifactId>jersey-container-grizzly2-servlet</artifactId>
  <version>2.27</version>
</dependency>

<dependency>
  <groupId>org.glassfish.jersey.containers</groupId>
  <artifactId>jersey-container-jdk-http</artifactId>
  <version>2.27</version>
</dependency>

<dependency>
  <groupId>org.glassfish.jersey.containers</groupId>
  <artifactId>jersey-container-simple-http</artifactId>
  <version>2.27</version>
</dependency>

<dependency>
  <groupId>org.glassfish.jersey.containers</groupId>
  <artifactId>jersey-container-jetty-http</artifactId>
  <version>2.27</version>
</dependency>

<dependency>
  <groupId>org.glassfish.jersey.containers</groupId>
  <artifactId>jersey-container-jetty-servlet</artifactId>
  <version>2.27</version>
</dependency>
```