

玩转Java的CLASSPATH (二) Classpath工具 PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/145/2021\\_2022\\_\\_E7\\_8E\\_A9\\_E8\\_BD\\_ACJava\\_c104\\_145239.htm](https://www.100test.com/kao_ti2020/145/2021_2022__E7_8E_A9_E8_BD_ACJava_c104_145239.htm) 二、可能出现的问题 不管他们是否愿意承认，初学者和富有经验的Java开发者都一样，他们都曾经在某些时候（通常是在那些最糟糕的情形下）被冗长、复杂的classpath欺骗。应用程序所依赖的第三方类和用户定义类的数量逐渐增长，classpath也逐渐成了一个堆积所有可能的目录和档案文件名的地方。此时，类装载机首先装载的究竟是哪一类也就不再显而易见。如果classpath中包含重复的类入口，这个问题尤其突出。前面已经提到，类装载机总是装载第一个它在classpath中找到的具有合适名字类，从实际效果看，它“隐藏”了其他具有合适名字但在classpath中优先级较低的类。如果不小心，你很容易掉进这个classpath的陷阱。当你结束了一天漫长的工作，最后为了让应用程序使用最好、最新的类，你把一个目录加入到了classpath，但与此同时，你却忘记了：在classpath的另一个具有更高优先级的目录下，存放着该类的另一个版本！ 三、一个简单的classpath工具 优先级问题是扁平路径声明方法与生俱来固有的问题，但它不是只有Java的classpath才有的问题。要解决这个问题，你只需站到富有传奇色彩的软件巨构的肩膀上：Unix操作系统有一个which命令，在命令参数中指定一个名字，which就会显示出当这个名字作为命令执行时执行文件的路径名。实际上，which命令是分析PATH变量，然后找出命令第一次出现的位置。对于Java的类路径管理来说，这应该也是一个好工具。在它的启发之下，我着手设计了一个Java工具JWhich。这

个工具要求指定一个Java类的名字，然后根据classpath的指引，找出类装载机即将装载的类所在位置的绝对路径。下面是一个JWhich的使用实例。它显示出当Java类装载机装载com.clarkware.ejb.ShoppingCartBean类时，该类第一次出现位置的绝对路径名，查找结果显示该类在某个目录下：> java JWhich com.clarkware.ejb.ShoppingCartBean Class  
com.clarkware.ejb.ShoppingCartBean found in

```
/home/mclark/classes/com/clarkware/ejb/ShoppingCartBean.class
```

下面是第二个JWhich的使用实例。它显示出当Java类装载机装载javax.servlet.http.HttpServlet类时，该类第一次出现位置的绝对路径名，查找结果显示该类在某个档案文件中：> java

```
JWhich javax.servlet.http.HttpServletClass
```

```
javax.servlet.http.HttpServlet found in
```

```
file:/home/mclark/lib/servlet.jar!/javax/servlet/http/HttpServlet.class
```

100Test 下载频道开通，各类考试题目直接下载。详细请访问  
[www.100test.com](http://www.100test.com)