

JAVA基础：Java启动器如何查找类 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_JAVA_E5_9F_BA_E7_A1_80_c104_145457.htm Java 启动器 java 将初始化 Java 虚拟机。虚拟机随即按以下顺序搜索和加载类：自举类 - 构成 Java 平台的类，包括 rt.jar 和 i18n.jar 中的类。扩展类 - 使用 Java 扩展机制的类。它们被捆绑为 .jar 文件，位于扩展目录中。用户类 - 开发人员和第三方定义的类，不使用扩展机制。在命令行上使用 -classpath 选项（常用方法）或使用 CLASSPATH 环境变量可识别这些类的位置（参见设置 Windows 或 Solaris 的 Classpath）。实际上，这三个搜索路径将被连为一个简单的类路径。这有点象以前使用的“单级”类路径，但现在的模型有一些重要差别：偶然“隐藏”或省略自举类相对困难一些。通常只需指定用户类的位置。系统将“自动”查找自举类和扩展类。工具类现在位于独立的归档中 (tools.jar)。只有当它包含在用户类路径中时才可用（稍后会有解释）。Java 启动器如何查找自举类自举类是实现 Java 1.2 平台的类。自举类在 /jdk1.2/jre/lib 中的 rt.jar 和 i18n.jar 归档中。这些归档由存储在 sun.boot.class.path 系统属性中的自举类路径值指定。该系统属性仅供引用，不应直接修改。需要重新定义自举类路径的可能性较小。极少数情况下需要使用另一组核心类时，非标准选项 -Xbootclasspath 允许重新定义自举类路径。注意：实现 JDK 开发工具类与自举类在不同归档中。工具归档为 /jdk1.2/lib/tools.jar。调用启动器时，开发工具将该归档添加到用户类路径中。然而，这个扩大的用户类路径仅用于执行工具。处理源代码的工具 javac 和

javac 将使用初始类路径，而不是扩大后的类路径（有关详细信息，参见下面的 Javac 和 Javadoc 如何查找类）。Java 启动器如何查找扩展类扩展类是扩展 Java 平台的类。扩展目录中的每个 .jar 文件即被认为是扩展，且都使用 Java 扩展框架加载。我们不提供改变扩展目录位置的选项。Java 启动器如何查找用户类用户类是在 Java 平台上构建的类。查找用户类时，启动器需要使用用户类路径，它是一张目录、JAR 归档和包含类文件的 ZIP 归档的列表。类文件有一个反映类全限定名的子路径名。例如，如果类 com.mypackage.MyClass 存储在 /myclasses 下，则 /myclasses 一定位于用户类路径中，且类文件的完全路径一定为

/myclasses/com/mypackage/MyClass.class。如果类存储在名为 myclasses.jar 的归档中，则 myclasses.jar 一定位于用户类路径中，且类文件一定作为 com/mypackage/MyClass.class 存储在归档中。在 Solaris 中，我们用字符串指定用户类路径，而用冒号 (:) 分隔类路径项。java 启动器将用户类路径字符串放置在 java.class.path 系统属性中。该值可能的源为：缺省值 “.” 意味着用户类文件是当前目录中的所有类文件（如果是在包中，则为当前目录下的）。CLASSPATH 环境变量的值，它覆盖缺省值。命令行选项 -cp 或 -classpath 的值，它覆盖缺省值和 CLASSPATH 值。-jar 选项指定的 JAR 归档，它覆盖所有其它值。如果使用该选项，则所有用户类都来自指定的归档。Java 启动器如何查找 JAR-CLASS-PATH 类JAR 文件通常包含一个“清单” - 该文件列出 JAR 的内容。清单可以定义 JAR-CLASS-PATH，而后者进一步扩展类路径（仅当从该 JAR 中加载类时）。查找 JAR-CLASS-PATH 访问的类时将按

以下顺序进行：通常，查找 JAR-CLASS-PATH 项引用的类时，即将其当作 JAR 文件的一部分。出现在 JAR-class-path 中的 JAR 文件将在 JAR-CLASS-PATH 前面的类路径项之后及 JAR-CLASS-PATH 后面的类路径项之前进行搜索。然而，如果 JAR-CLASS-PATH 指向一个已搜索过的 JAR 文件（例如扩展或在类路径前面列出的 JAR 文件），则不再搜索该 JAR 文件（这种优化方式提高了效率，防止了循环搜索）。这样的 JAR 文件将在它出现的地方（类路径中更靠前的地方）进行搜索。如果 JAR 文件作为扩展安装在 jdk 的 ext 子目录中，则忽略它所定义的 JAR-CLASS-PATH。我们假定扩展所需的所有类都是 JDK 的一部分或已作为扩展安装。OldJava 启动器如何查找类oldjava 启动器不支持 Java 扩展框架。下列情况下，它提供向后兼容性：有一个实现 1.1 风格的安全管理器（与 1.2 类加载方法不兼容）的应用程序，或者正在加载的类已生成，或以与 1.2 类结构不兼容的方式（例如用模糊器）管理。（有关这些问题的详细信息，参见 Windows 或 Solaris 的 java 启动器参考页。） 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