

什么是双亲委派？如何打破双亲委派？ 原创

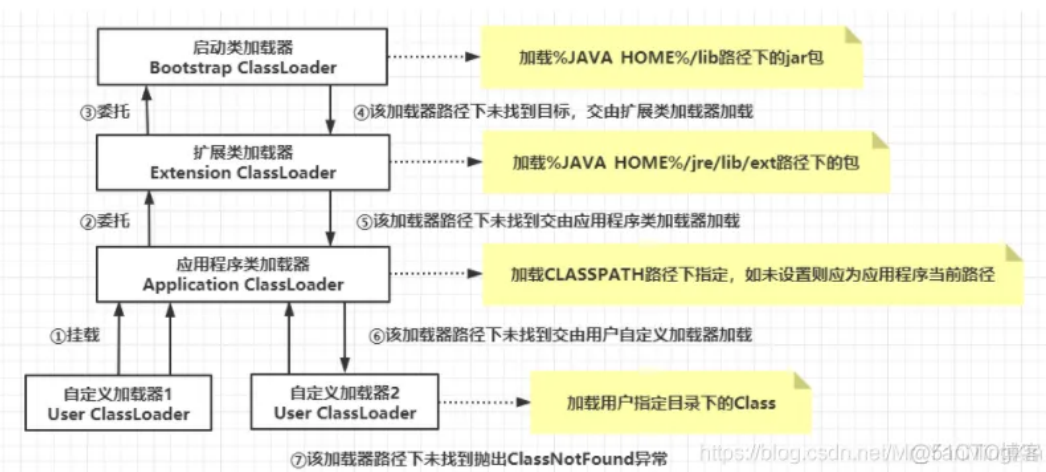
海风极客 2022-05-24 19:50:12 博主文章分类：细说Java基础 ©著作权

文章标签 多线程 java 面试 jvm 类加载器 文章分类 Java 后端开发 阅读数 2750

1 双亲委派（又称双亲委托）

作用：保证类加载的有序性和安全性

类加载器的分类：

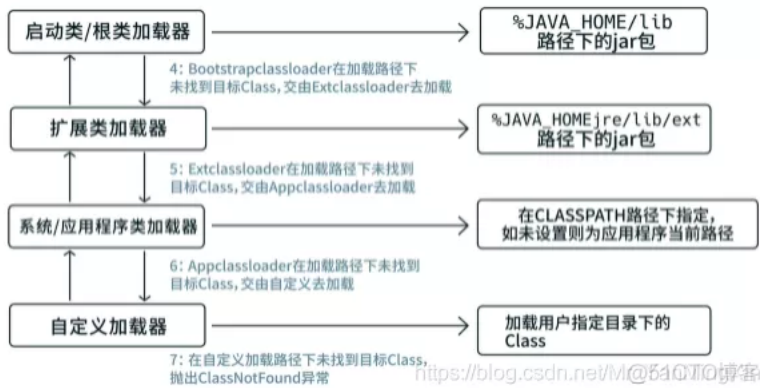


作用：保证JVM的核心类和用户的类都能得到正常加载

双亲委派机制：

向上委派：当类加载器加载一个类时，首先会向他的父类加载器进行委派，直到启动类加载器。





2 如何打破双亲委派

两个典型的方法：

- 自定义类加载器，重写loadClass方法
- 使用线程上下文类加载器

2.1 重写loadClass方法

因为双亲委派机制的实现都是通过这个方法实现的，这个方法可以指定类通过什么加载器来进行加载，所以如果我们改写他的加载规则，就相当于打破了双亲委派机制。默认的过程是这样的，先判断这个类是不是已经被当前层的类加载器加载过了，如果没有加载过就将该类委派给父类加载器，如果父加载器无法加载再向下传递，回来由自己来进行加载，重写了这个方法以后就能自己定义使用什么加载器了，也可以自定义加载委派机制，也就打破了双亲委派模型。

重写loadClass()方法也就有可能连带把findClass(方法也重写。

2.2 线程上下文类加载器

线程上下文类加载器(Thread Context ClassLoader)。这个类加载器可以通过java.lang.Thread类的setContextClassLoader()方法进行设置，如果创建线程时还未设置，他将会从父线程中继承一个，如果在应用程序的全局范围内都没有设置过的话，那这个类加载器默认就是应用程序类加载器。

有了线程上下文加载器，JNDI服务就可以使用它去加载所需要的SPI代码，也就是父类加载器请求子类加载器去完成类加载的动作，这种行为实际上就是打通了双亲委派模型层次结构来逆向使用类加载器，实际上已经违背了双亲委派模型的一般性原则，但这也是无可奈何的事情。Java中所有涉及SPI的加载动作基本上都采用这种方式，例如JNDI、JDBC、JCE、JAXB和JBI等。