

专业语言：实现Java平台的三种方式 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/144/2021_2022__E4_B8_93_E4_B8_9A_E8_AF_AD_E8_c104_144378.htm

广泛地说，只要能执行 Java bytecode 者，就可以称为 Java 平台（Java platform）。大致上，实现 Java 平台的方式有三种，分别是：Java 虚拟机（Java Virtual Machine, JVM）Java 操作系统（Java Operating System）Java 芯片（Java Chip）以下就此三种 Java 平台提出解说，并比较之。Java 虚拟机所谓的 Java 虚拟机，指的是在操作系统上执行的一种程序，此程序可以解读 Java bytecode，并在操作系统的帮助下执行 Java bytecode。这样的执行架构，大致上可以分成四层，分别是：第一层（最上层）的 Java bytecode 第二层的 Java 虚拟机 第三层的操作系统 第四层（最底层）的芯片 芯片执行上面的操作系统，操作系统执行上面的 Java 虚拟机，Java 虚拟机执行上面的 Java bytecode。对于 Java bytecode 来说，它只接触到 Java 虚拟机，所以当然「误以为」Java 虚拟机就是它的平台。实例：Pentium III Chip Windows 2000 OS Sun HotSpot Client VM Together/ Java Application。Java 操作系统如果把 Java 虚拟机整合进既有的操作系统中（例如：Mac OS X），或是打造一个全新的操作系统专门执行 Java bytecode（例如：JOS），这类能直接执行 Java bytecode 的操作系统也可视为 Java 平台。这样的执行架构，大致上可以分成三层，分别是：第一层（最上层）的 Java bytecode 第二层的 Java 操作系统 第三层（最底层）的芯片 Java 芯片如果把 Java 操作系统和既有的芯片整合起来成为一组新的芯片（例如：ARM 7、ARM 9），或

者打造一个全新的 Java 专用芯片（例如：ajile aj-100）那么此一种类的芯片也是 Java 平台。这样的执行架构，大致上可以分成两层，分别是：第一层（最上层）的 Java bytecode 第二层（最底层）的 Java 芯片 三种方式的比较 这三种都是 Java 平台，各有优缺点，很难说那一种方式比较好，必须看你的需求而定。透过虚拟机器来执行 Java 是最常见的架构方式，一方面是因为虚拟机器的 Java 平台最容易被实作出来（相较于 Java 操作系统和 Java 芯片而言），二方面是因为只要有虚拟机器就可以让在不更动实体平台（操作系统和 CPU）的情况下多了执行 Java 程序的功能。目前，多数主流的实体平台（包括 Windows、Linux、Mac、Solaris、PalmOS、EPOC...）都已经有 Java 虚拟机器了。虚拟机器的架构因为分了太多层级，所以效率上当然会比较差，这是虚拟机器架构最为人所抨击的地方。但是由于虚拟机器的技术一直在进步，所以执行效率一直都有持续提升。目前已经相当逼近 Java 操作系统的架构方式，所以虚拟机器的方式应该会继续成为 PC 上的 Java 平台主流。一般来说，透过 Java 操作系统来执行 Java bytecode，效率会比虚拟机器的方式好。所以 Java 虚拟机器被整合进操作系统中，来提升该操作系统的附加价值，这种作法有可能会成为一种趋势（例如 MacOS X 就把 Java 和操作系统作了很好的整合）。但如果设计一套只能执行 Java 的全新操作系统，可能不容易成功，因为全新的操作系统势必面临驱动程序（device driver）不足的严重困扰。因为此原因，加上事实证明操作系统的 Java 平台，比虚拟机器的 Java 平台效率提升不如预期多，所以 IBM 和 Sun 合作的纯 Java 操作系统「JavaOS」计画已经停摆。利用芯片的方式来实作 Java 平台

，对于 Java 程序的执行速度会有很大的提升，毕竟硬件的速度比软件快许多。这种方式以后会成为嵌入式系统的主流，许多 Java 的 IA 产品（例如：Set-Top Box）都会开始使用 Java 芯片。但是这种方式已经把 Java 整个变成硬件平台，不适合再称为跨平台了。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