Java语言的语法与C语言和C++语言很接近,使得大多数 程序员很容易学习和使用。另一方面, Java丢弃了C++中 很少使用的、很难理解的、令人迷惑的那些特性,如操作 Java语言的设计目标之一是适应于动态变化的环境。 符重载、多继承、自动的强制类型转换。特别地,Java语 Java程序需要的类能够动态地被载入到运行环境,也可 言不使用指针,而是引用。并提供了自动的废料收集,使 以通过网络来载入所需要的类。这也有利于软件的升 得程序员不必为内存管理而担忧。 级。另外,Java中的类有一个运行时刻的表示,能进行 Java语言是简单的 运行时刻的类型检查。 Java语言是动态的 Java语言提供类、接口和继承等原语,为了简单起见,只支持类之间的单继 承,但支持接口之间的多继承,并支持类与接口之间的实现机制(关键字为 在Java语言中,线程是一种特殊的对象,它必须由Thread类或其子 implements ) 。Java语言全面支持动态绑定,而C++语言只对虚函数使用 (孙)类来创建。通常有两种方法来创建线程:其一,使用型构为 动态绑定。总之,Java语言是一个纯的面向对象程序设计语言。 Thread(Runnable)的构造子将一个实现了Runnable接口的对象包 Java语言是面向对象的 装成一个线程,其二,从Thread类派生出子类并重写run方法,使 用该子类创建的对象即为线程。值得注意的是Thread类已经实现了 Runnable接口,因此,任何一个线程均有它的run方法,而run方法 中包含了线程所要运行的代码。线程的活动由一组方法来控制。 Java语言支持多个线程的同时执行,并提供多线程之间的同步机制 Java语言支持Internet应用的开发,在基本的Java应用编程接口中 (关键字为synchronized)。 有一个网络应用编程接口(java net),它提供了用于网络应用编 Java语言是多线程的 程的类库,包括URL、URLConnection、Socket、ServerSocket 等。Java的RMI(远程方法激活)机制也是开发分布式应用的重要 手段。 Java语言是分布式的 2、java主要特性 与那些解释型的高级脚本语言相比, Java的确是 高性能的。事实上, Java的运行速度随着JIT(Just-In-Time)编译器技术的发展越来越接近于C++。 Java的强类型机制、异常处理、垃圾的自动收集等是Java 一 公众号java-mindmap Java是高性能的 程序健壮性的重要保证。对指针的丢弃是Java的明智选 择。Java的安全检查机制使得Java更具健壮性。 Java语言是健壮的 如前所述, Java程序在Java平台上被编译为字节码格 式,然后可以在实现这个Java平台的任何系统中运 行。在运行时, Java平台中的Java解释器对这些字节 Java通常被用在网络环境中,为此,Java提供了一个安全机制以防恶意代 码进行解释执行,执行过程中需要的类在联接阶段被 码的攻击。除了Java语言具有的许多安全特性以外,Java对通过网络下载 载入到运行环境中。 的类具有一个安全防范机制(类ClassLoader),如分配不同的名字空间 Java语言是解释型的 以防替代本地的同名类、字节代码检查,并提供安全管理机制(类 SecurityManager) 让Java应用设置安全哨兵。 Java语言是安全的 这种可移植性来源于体系结构中立性,另外, Java还严格规定了各个基本数据类型的长度。 Java系统本身也具有很强的可移植性 , Java编译 Java程序(后缀为java的文件)在Java平台上被编译为 器是用Java实现的, Java的运行环境是用ANSI C 体系结构中立的字节码格式(后缀为class的文件),然 实现的。 后可以在实现这个Java平台的任何系统中运行。这种途 Java语言是可移植的 径适合于异构的网络环境和软件的分发。 Java语言是体系结构中立的