

欢迎来到Java世界

experiment

第0章课程介绍

主讲人：刘嘉欣

邮件：liujx@nankai.edu.cn

电话：13802187792

为什么要学习Java

了解 Web 的重要性
后欲构建 Web
应用

想认识 Java 在
Web 中的角色



想成为 Java
Programmer —
展身手

学习本课的预备知识



具备
C/C++编程经验



能够使用
一种文本
编辑器创
建文本文
件



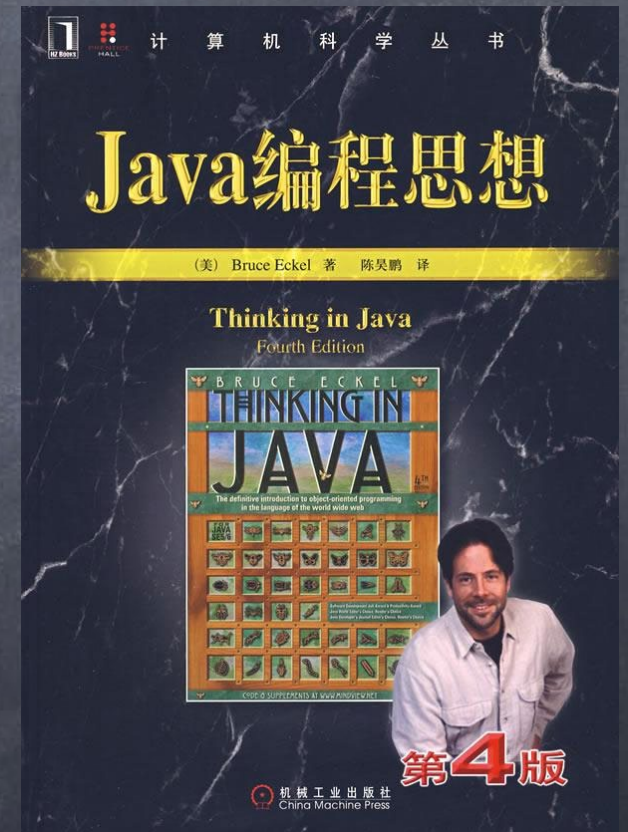
会使用基
本的
DOS命
令

教学目标

- 理解面向**对象编程技术**
- 了解Java**语言**的基本概念、**术语**
- **编写**基本的Java**应用程序**
- 会使用**图形、图像、声音**等元素**编写图形界面程序**
- **应用I/O类库，进行文件操作**
- **网络编程**

参考书目

- Java编程思想 (Think In Java)
- Java API文档



Java从哪里来？



开发团队(Green Team)

James Gosling



Java从哪里来？

- 1991年 Green Team创建oak,需要面对很多问题
 - 各种各样的制造商
 - 内存一般很小
 - 不能“重新启动”
 - 安全
 - 简单

Java从哪里来？

- 1992年夏StarSeven (*7) 问世
- 1993年 WWW兴起
 - WebRunner与Applet使互联网第一次拥有了动态效果
- 1995年 第一届JavaOne大会 Java出生



Java 崛起之技术背景

- WWW 应用日益普遍
- 静态网页无法满足用户需求
- Client 希望更复杂生动的表达方式
- 用户希望 Internet 上能作更多的事情
- 时代召唤“跨平台”的语言

Java的发展

- 1995年3月公布第一个全公开的alpha版本的Java源代码“1.0a2”
- 1996年 JDK1.0问世
- 1998年 JDK1.2更名为Java2
- 2014年3月18日，发布Java 8最终版本
- 2022年3月22日，OpenJDK 18
- <http://jdk.java.net>

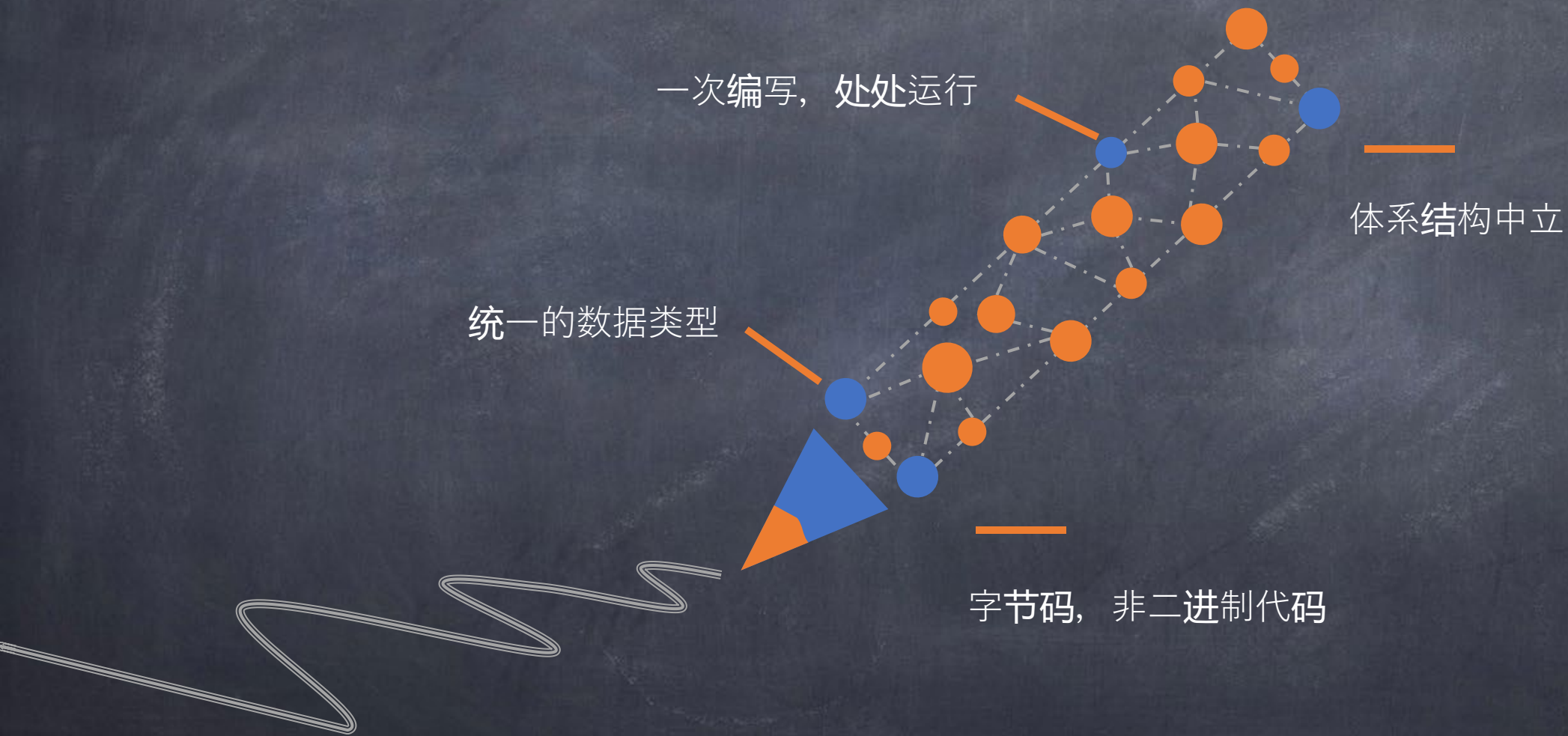
Java VS C++

	Java	C++
性能	慢	快
复杂度	简单	复杂
目标	网络、跨平台	Windows应用
风格	面向对象	兼容C

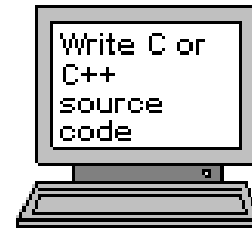
Java语言的特征

- Java是一门
 - 简单与面向对象的
 - 分布式、安全与鲁棒的
 - 解释执行与高性能的
 - 多线程与动态执行的
 - 平台无关与无缝移植的
- 编程语言

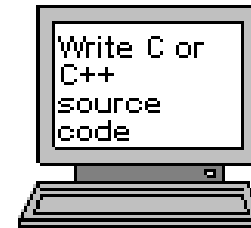
平台无关性与无缝移植



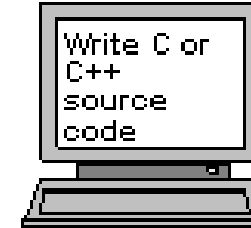
Create & Modify in C



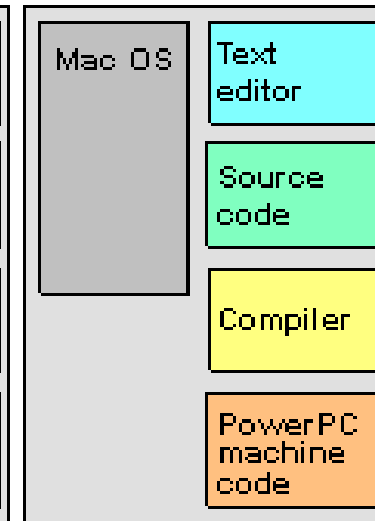
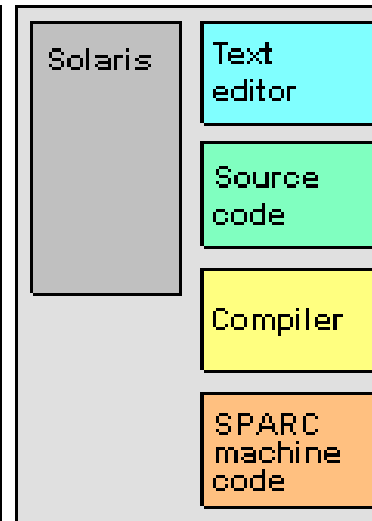
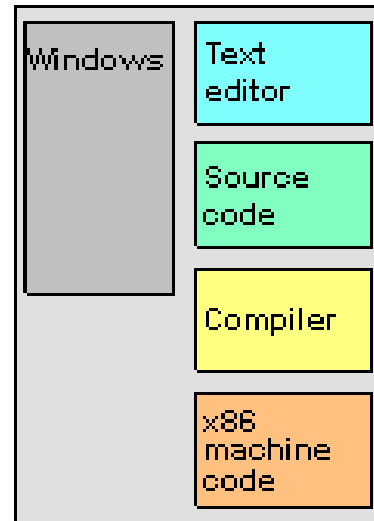
Intel x86



Sun SPARC

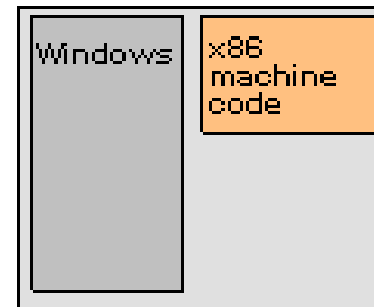


Mac PowerPC

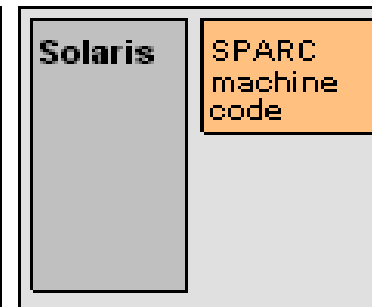


Run

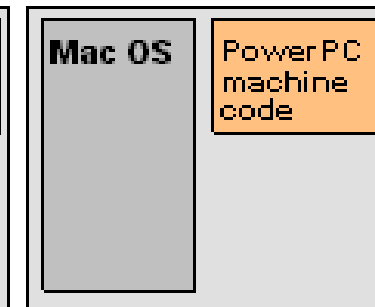
Intel x86



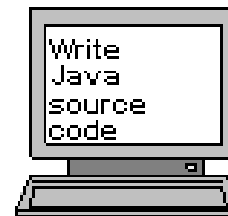
Sun SPARC



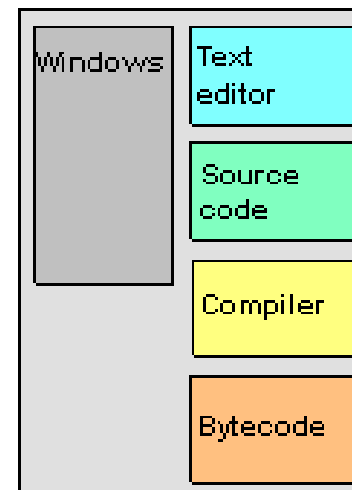
Mac PowerPC



Create & Modify in Java

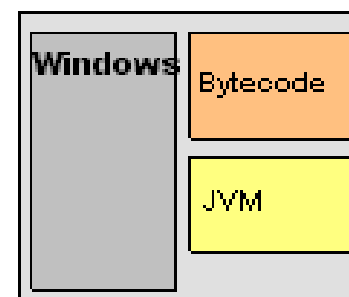


Intel x86

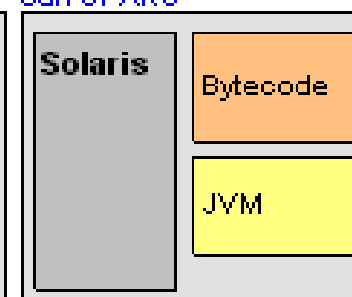


Run

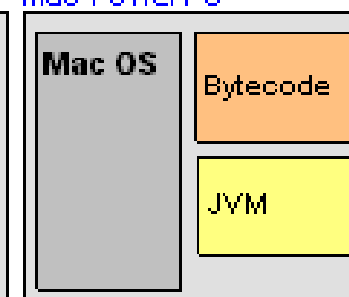
Intel x86



Sun SPARC



Mac PowerPC



简单而强大

- 非常像C++，但更加简单
 - 去掉容易出错的地方
 - 指针, 结构, typedef, goto, 预处理, 手工释放内存, 多重继承, 全局变量等
 - 强调面向对象和网络特性
- 小而精
 - 基本解释器只需要200KB RAM

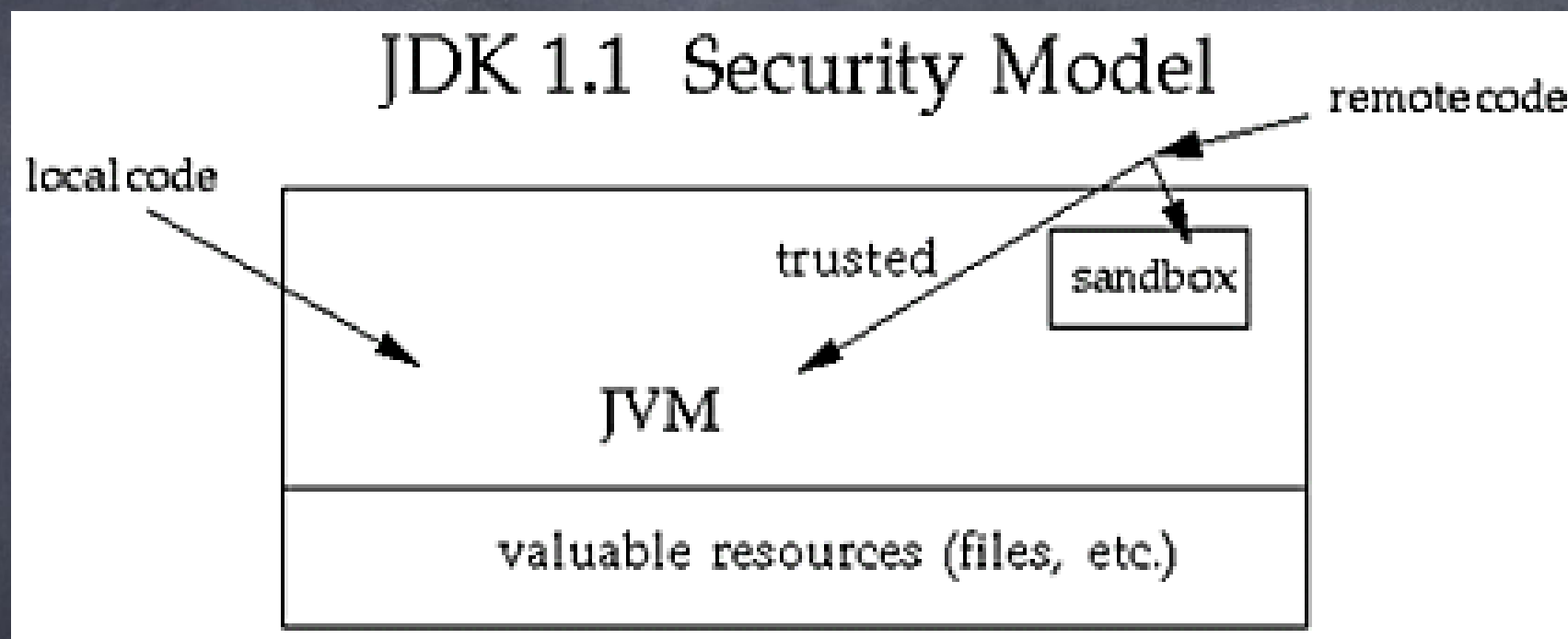
分布式

- 丰富的网络编程能力
 - 内嵌多种网络协议, TCP/IP, HTTP
- 丰富的客户端编程能力
- 丰富的服务器端编程能力

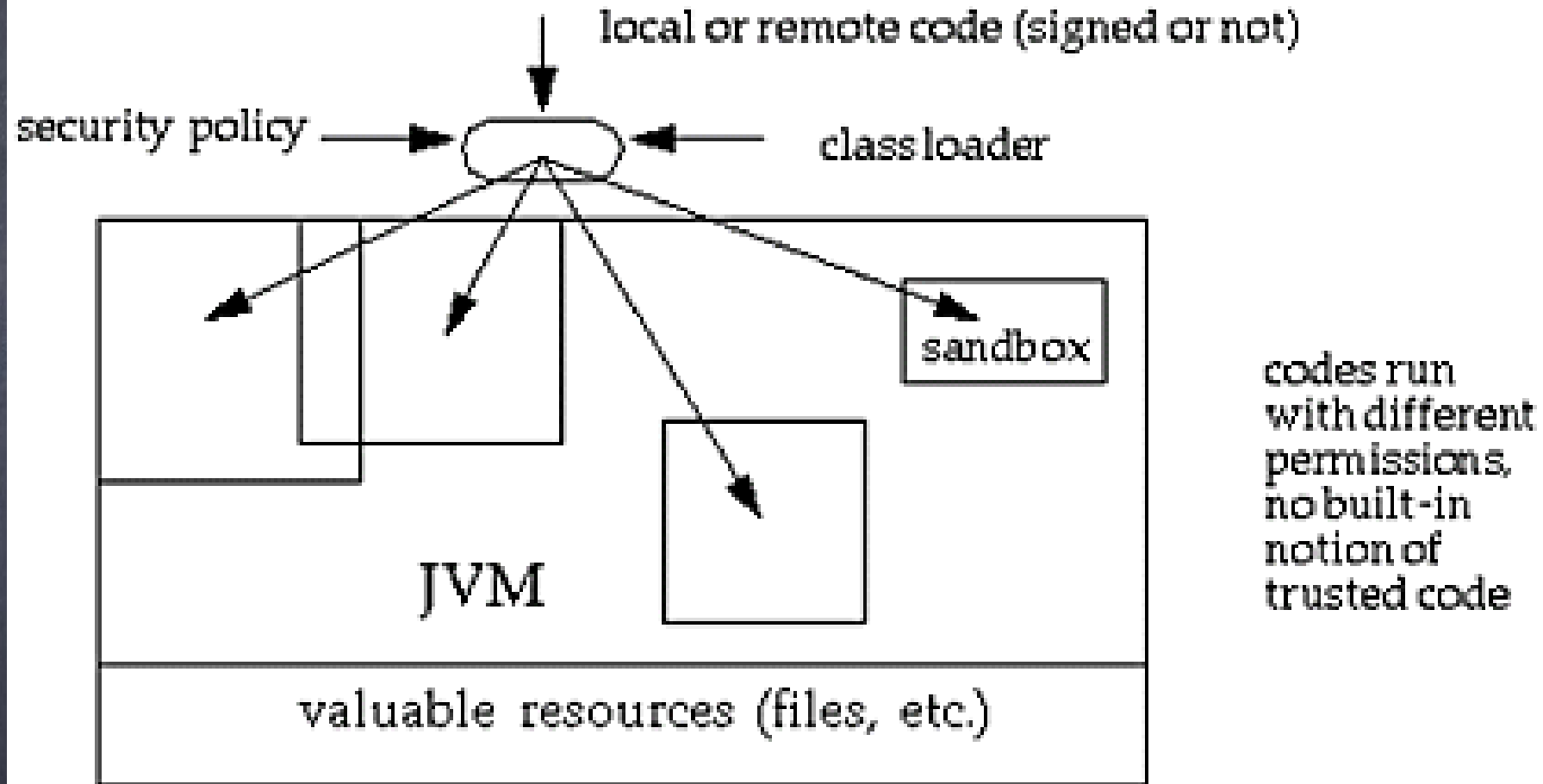
安全性

- 类型安全与字节码验证
- 由Java负责内存分配
- 沙箱模型
- 数字签名

沙箱模型



JDK 1.2 Security Model



鲁棒性

- 不死机
 - 异常处理
 - 自动垃圾收集，无内存泄漏
 - 静态检查，动态检查
 - 去掉指针

解释执行

- 1.编译成更接近本地代码的“字节码”
- 2.运行时，由解释器将“字节码”翻译成“本地代码”
- 比传统的解释性语言快
- 实现跨平台
- 资源消耗少

高性能

- 字节码精心设计
- 编译器自动优化
- Hotspot使Java与C++一样快

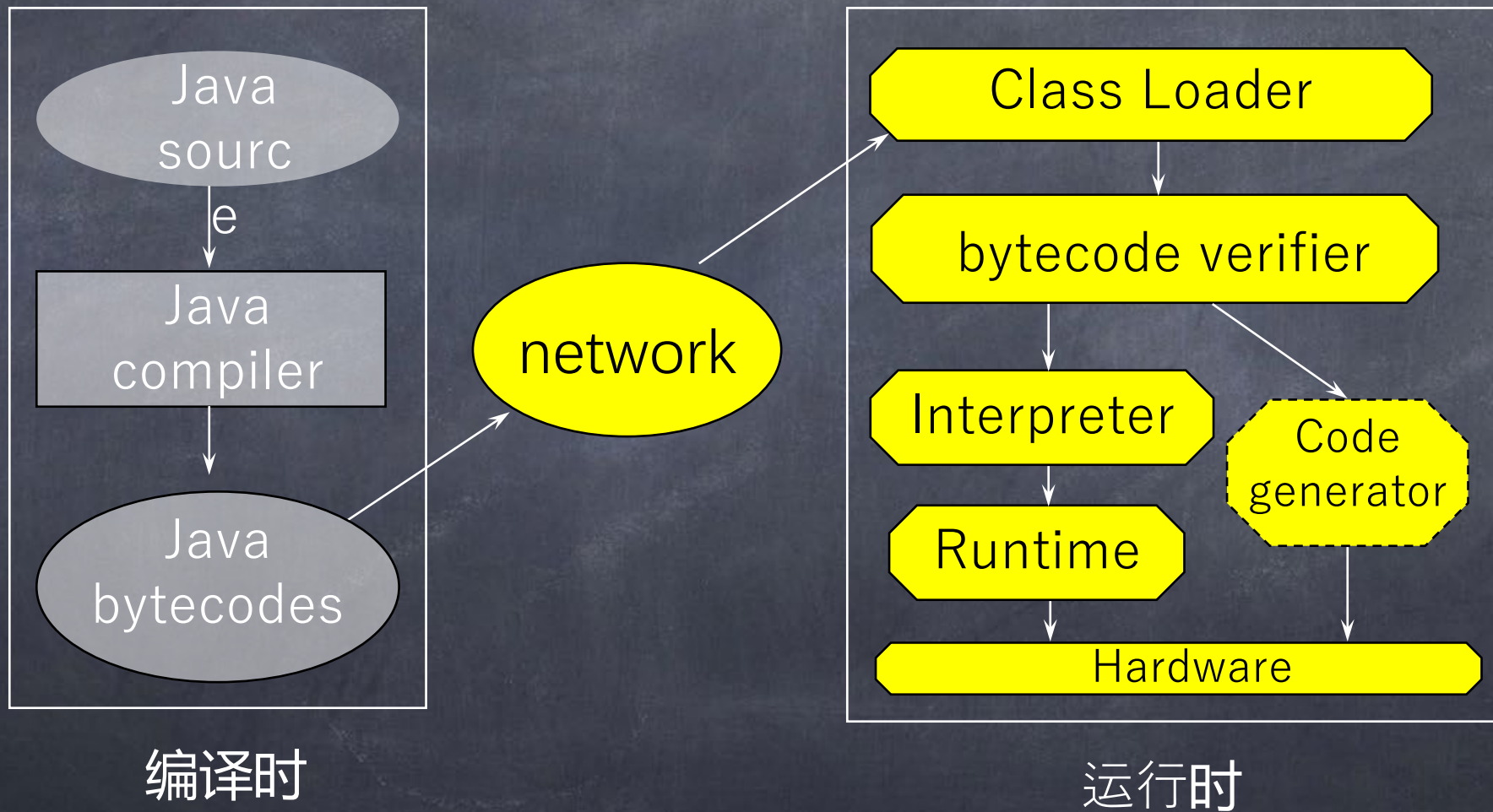
多线程支持

- 语言级别的多线程支持
- 简化了多线程开发
 - 用对象来表示线程
 - 提供限制性资源锁定功能
- 充分利用硬件资源

动态执行

- 动态装载执行代码
- 自动更新
- 不需重新编译
- 举例
 - 国际化/本地化
 - Internationalization, Localization(i18n/L10n)

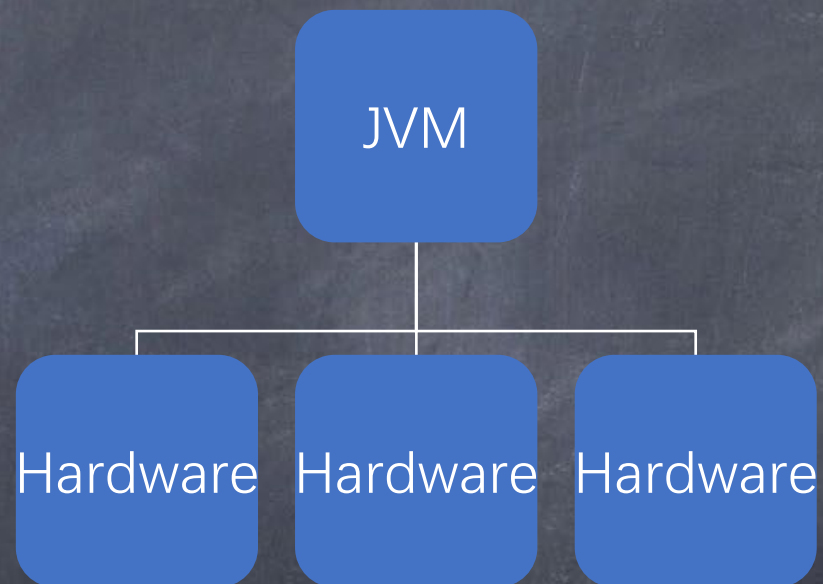
Java运行结构



什么是JVM

⑩ Java虚拟机 (Java Virtual Machine)

- Java定义的虚拟CPU
- 包括对指令集、寄存器、class文件格式、堆栈、垃圾回收堆、内存区域的定义



- 字节码 (ByteCode)
 - JVM的机器指令
- 垃圾回收 (Garbage – Collection)
 - Java中**动态**分配的内存无需在程序中**释放**, Java 会自动**收回**这些内存。
- Java API
 - Java类**库**的**应用编程**接口
- JRE (Java运行**环境**, Java RunTime)
 - 由JVM和java平台的核心支持类**库**构成

- JDK (Java Developer's Kits)
 - Java开发工具
 - 包括JRE, JVM, Java API类库, 调试工具以及配置工具等
- Java平台
 - 包括JavaEE, JavaSE, JavaME
 - JavaEE : Java企业版, 应用于服务器编程
 - JavaSE : Java标准版, 应用于桌面编程
 - JavaME : Java微型版, 应用于嵌入式系统