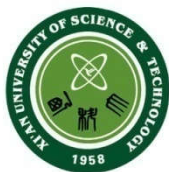


西安科技大学
XI' AN UNIVERSITY OF SCIENCE TECHNOLOGY

第一章 计算机操作系统概论

刘晓建

2018年8月

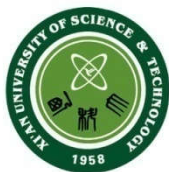


操作系统的普遍性、基础性和核心性

计算机操作系统是当今最复杂的系统软件之一，它是几乎所有复杂应用软件的基础。操作系统已经深度融入到几乎所有与信息处理相关的系统之中，小到各种传感器、手机、掌上电脑，大到整个互联网。它的应用如此广泛，以至于我们感觉不到它的存在。

课程偏重于基本概念、原理 和方法

讲解操作系统的基本概念、原理和方法，因此内容具有一般性和普适性。掌握了这门课程的内容之后，可以将它们应用到具体操作系统的学习和应用开发之中，比如手机操作系统、嵌入式操作系统和云操作系统等。



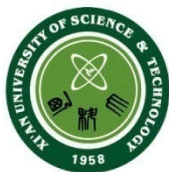
教学和考试方式

理论教学 +

课后/课堂作业 (20%) +

上机作业 (10%) +

期末考试 (70%)



<https://ansafe.xust.edu.cn>

->Teaching 教学课件

The screenshot shows a web browser window displaying the homepage of the Institute of Software Engineering at Xi'an University of Science and Technology. The browser's address bar shows the URL <https://ansafe.xust.edu.cn/index.htm>. The website features a navigation menu with items: Home, Research, Graduate, Project, Teaching, Tools, Resource, Base, and Share. The main content area is titled "Resume" and lists the profile of Associate Professor Liu Xiaojian. To the right, there is a section for "Photo" (照片) with a portrait of the professor, and an "Academic Introduction" (学术简介) section detailing his research areas and achievements.

Resume

副教授：刘晓建

- 2013年1月—至今 西安科技大学计算机学院软件工程系，副教授。
- 2006年7月—2012年12月 山东科学院自动化研究所副研究员，山东省汽车电子重点实验室副主任。
- 2001年9月—2006年6月 西北工业大学计算机科学与技术博士后流动站，博士后研究员。
- 2005年1月—2005年10月 国际软件技术研究所 (<http://www.iist.unu.edu>)，fellow。
- 2000年2月—2001年6月 西安电子科技大学软件工程研究所，博士研究生。
- 1997年7月—1999年12月 西安大唐电信有限公司，研发员，项目经理。
- 1994年9月—1997年6月 南京大学数学系，计算数学专业，硕士研究生。
- 1990年9月—1994年6月 南京大学大气科学系，本科。

照片墙

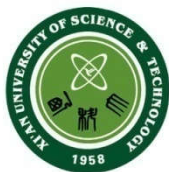
学术简介

研究领域

- 软件形式化方法。
- 嵌入式软件系统。
- 软件可靠性。

学术成果

- 获得1项实用新型专利，5项软件著作权。
- 2011年起发表第一作者论文20篇，第二作者5篇。
- 2006年以来 主持各类



西安科技大学

XI' AN UNIVERSITY OF SCIENCE TECHNOLOGY

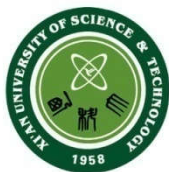
本章学习内容和基本要求

- ◆从三个角度理解计算机操作系统的概念；
- ◆在学习操作系统发展历史的基础上，重点掌握**多道程序设计**的概念、原理和实现；掌握**分时系统**的基本原理；
- ◆了解操作系统的**典型结构**，重点掌握层次结构、微内核结构的优点和不足；对**虚拟机结构**有所了解。



操作系统本质上是一组**系统级程序**，它管理和控制着其它**应用程序**的执行，并充当应用程序和计算机硬件之间的接口。一般来说，操作系统应满足如下三个应用需求：

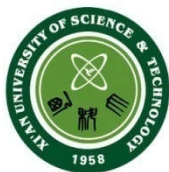
- ◆ 方便性——从用户使用计算机的角度
- ◆ 有效性——从计算资源分配和管理角度
- ◆ 可扩展性——从自身设计和构造角度
- ◆ 可信赖性——从运行保障的角度



可以从三个角度来看待操作系统

1、从用户使用角度来看，OS表现为一组可用的功能和提供这些功能的接口/使用界面。操作系统为用户提供了这些功能：

- ◆ 程序开发
- ◆ 程序运行
- ◆ I/O设备访问
- ◆ 文件访问和控制
- ◆ 并发控制和系统保护
- ◆ 错误检测和响应
- ◆ 日志和记账



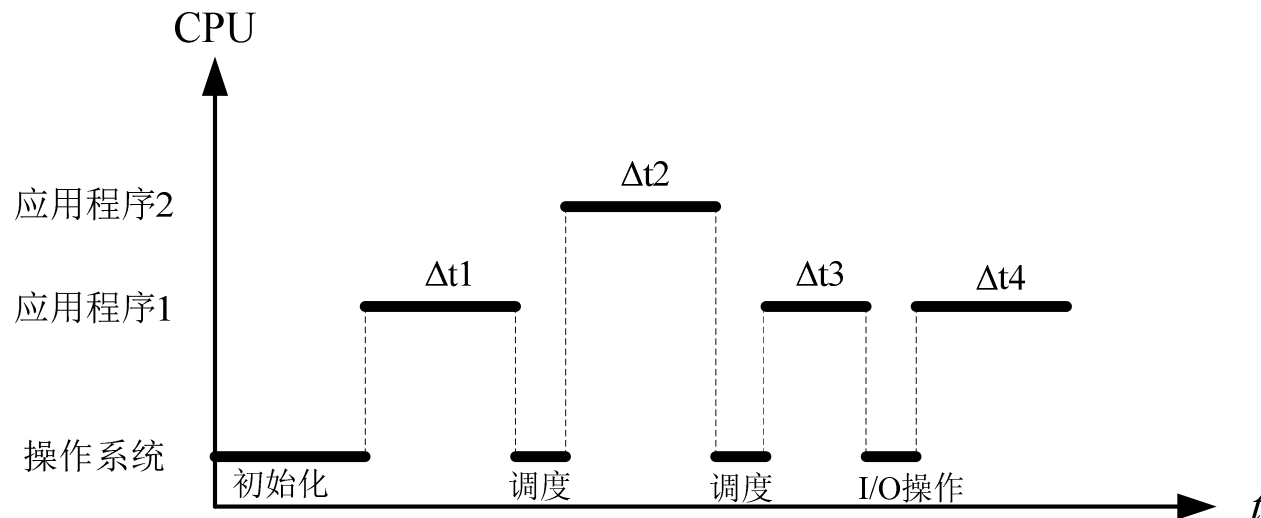
2、从计算资源管理和控制角度来看，OS表现为计算资源的管理者，其核心是**最优化问题**，也就是如何在有限资源的约束下，满足每个应用的需求，同时使整个系统的效能达到最优。计算资源包括：

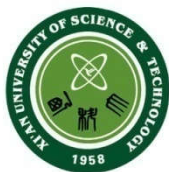
- ◆ 处理器（CPU）
- ◆ 存储资源，包括内存和外存
- ◆ I/O
- ◆ 实用程序、关键数据和文档等软件资源



OS管理和控制计算资源的特殊性表现在：

- ◆ OS和被管控的应用程序一样，都是软件程序，但操作系统是管理程序的程序，因此是“元程序”（Meta program）
- ◆ OS既是资源的管控者，同时也是资源的使用者，它必须耗费和占有计算资源
- ◆ OS对应用程序的管控是通过对计算资源的掌控和释放来实现的。



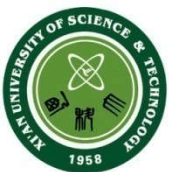


西安科技大学

XI' AN UNIVERSITY OF SCIENCE TECHNOLOGY

3、从计算环境的角度来看，OS为相互独立运行的用户和程序提供隔离的虚拟计算环境，为用户造成自己私人占有整个计算机的假象，使得用户和程序的逻辑独立性得以实现。

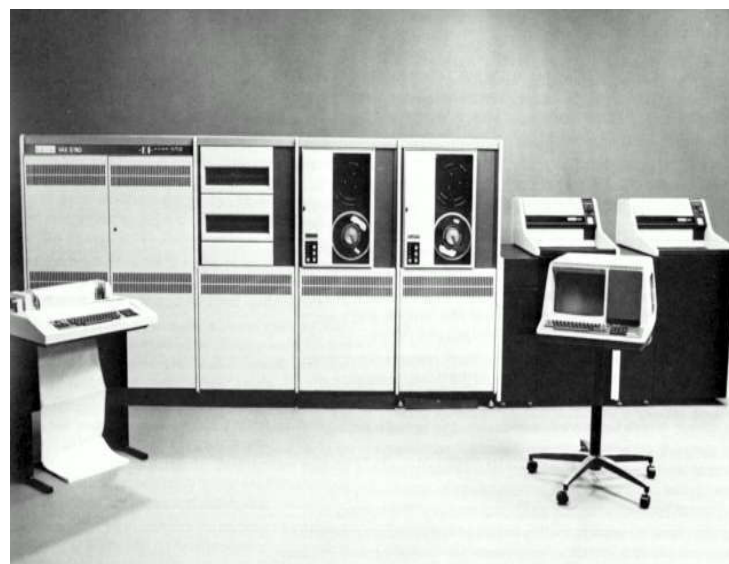
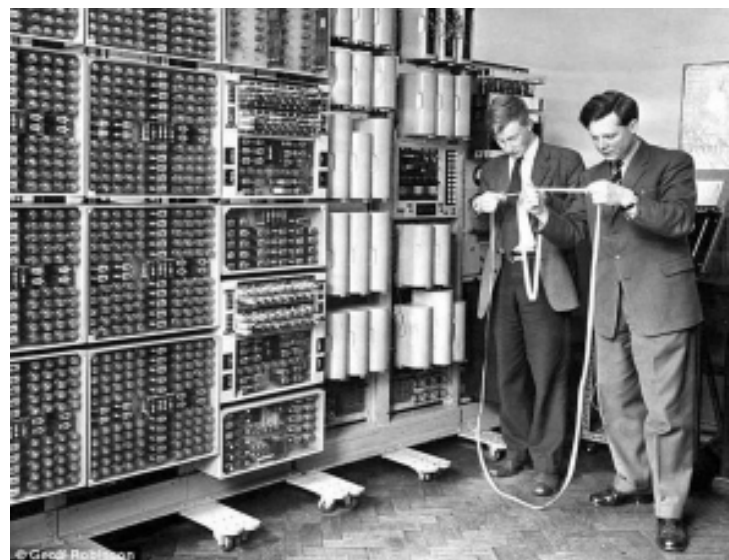
比如，在分时系统中，运用时分复用原理，为每个用户分配周期性时间片资源，如果周期足够小，就会为用户造成好像自己独占计算资源的假象；再如，在目前广泛使用的Android手机操作系统中，每个应用程序都运行在一个Dalvik虚拟机基础上，使得应用程序相互隔离。当一个程序出错时，错误被限定在自己的计算环境中，不会干扰其它应用程序，从而提高了系统的安全性和可靠性。

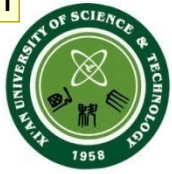


1.2 操作系统的发展历史

操作系统大致经历了这样几个阶段：

- ◆ 串行处理阶段
- ◆ 简单批处理系统
- ◆ 多道程序批处理系统
- ◆ 分时系统
- ◆ 分布式集群系统
- ◆ 互联网操作系统
- ◆

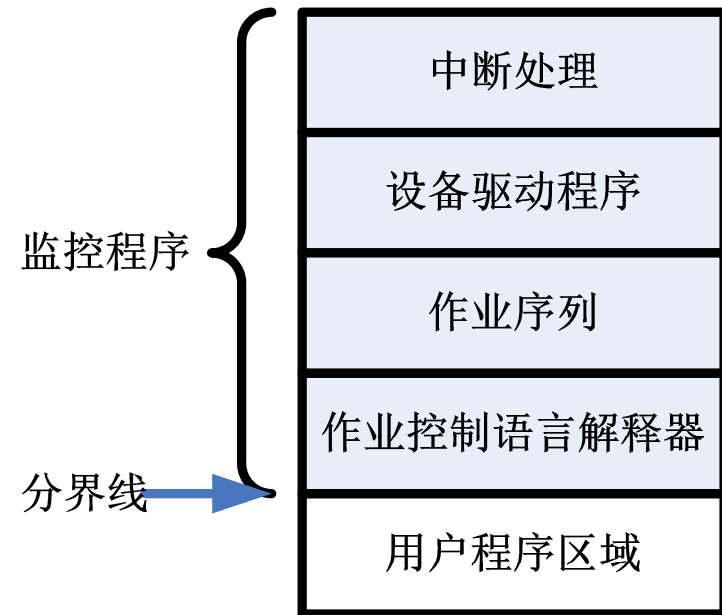


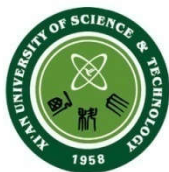


2、简单批处理系统中出现了OS的雏形

- 1、用户把作业程序（磁带或卡片）提交给计算机操作员
- 2、操作员根据每个作业的步骤要求，通过编写JCL程序，把所有作业组织成顺序处理任务
- 3、把JCL程序交给监控程序。监控程序执行JCL程序，执行每个作业
- 4、监控程序执行一个作业时，把作业调入用户程序区，并把计算资源释放给该作业
- 5、一个作业完成时，通过中断通知给监控程序，由监控程序回收资源，并调入下一个作业，然后又把资源释放给它，让它继续执行。如此循环直到所有作业执行完毕

这里有哪些三类程序？
它们的关系如何？





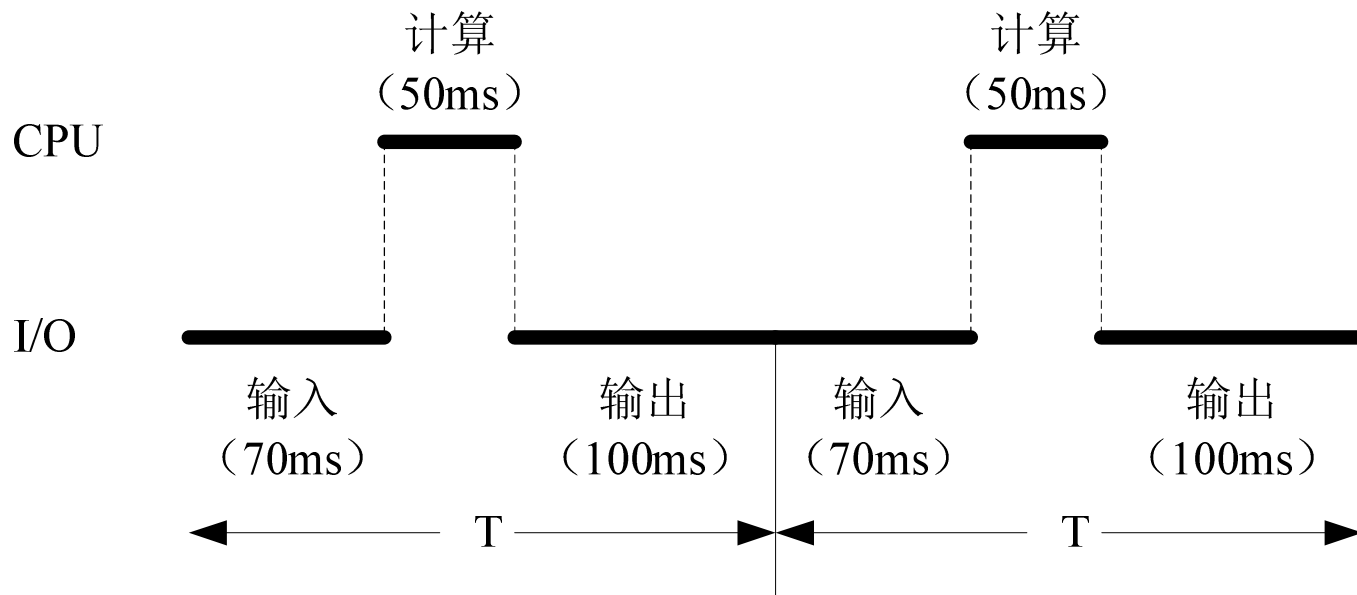
3、多道程序批处理系统：内存中同时存放了多道程序，它们都处于开始与结束点之间。

从宏观上看，多道程序并发运行，它们都处于运行过程中，但都未运行结束。从微观上看，多道程序的执行是串行的，各道程序轮流占用处理器，交替执行。

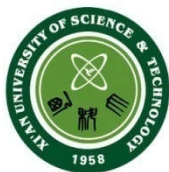
硬件基础：

- ◆ 中断处理——使得几个程序在未完成之前可以被打断，从而把执行权切换到其它程序
- ◆ I/O通道技术——把复杂的I/O处理过程代理给通道处理机去执行，从而解放了CPU

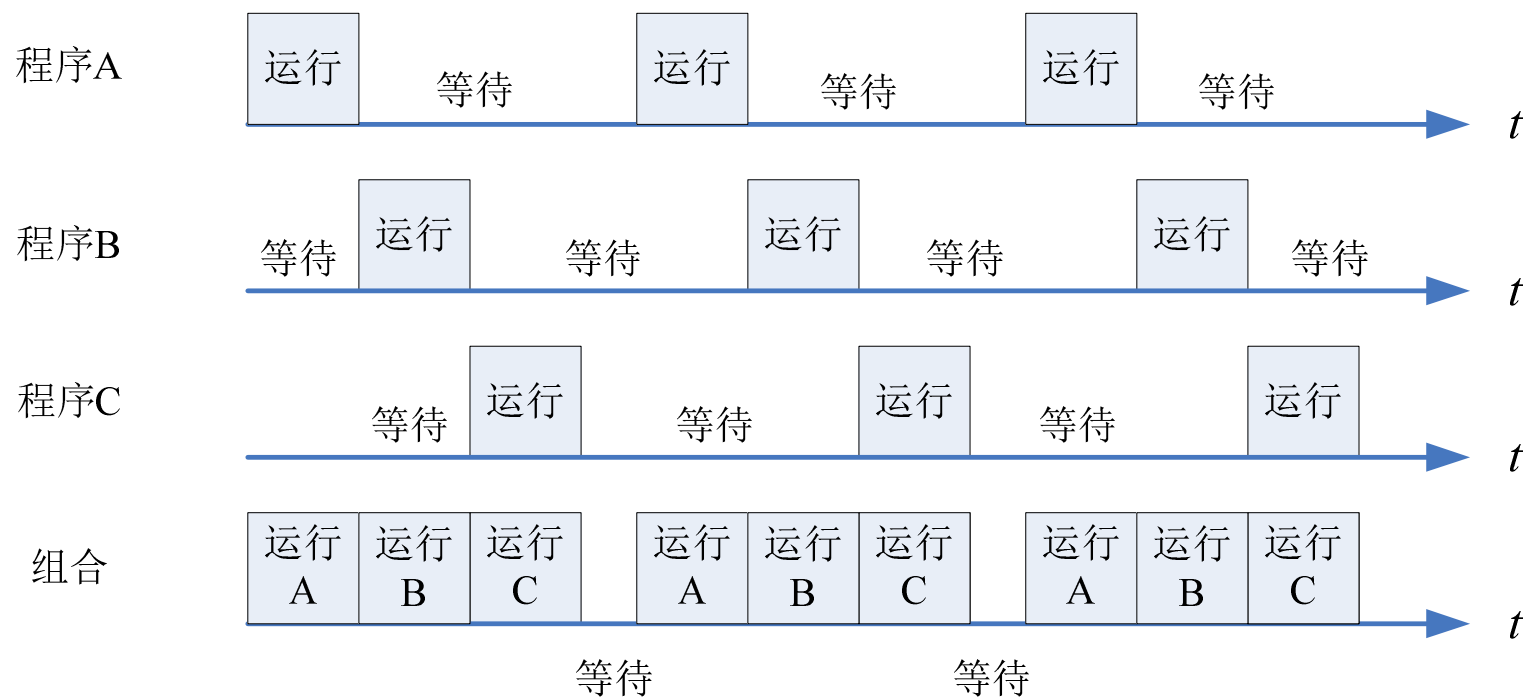
【例1-1】某个数据处理问题P1，要求从输入机上输入500个字符（花费70ms），经处理器处理50ms后，将结果的2000个字符存到磁带上（花费时间100ms），重复这个过程，直至数据全部处理完毕。试计算这个问题中CPU的利用率。

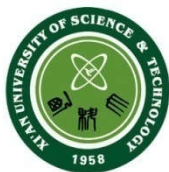


CPU的利用率= $50 / (70 + 50 + 100) = 23\%$ 。可见单道程序运行时CPU的利用率较低，主要原因是I/O的执行速度远低于CPU的执行速度，使得CPU大部分时间都处于等待I/O完成的状态，宝贵的CPU资源被浪费了。



【例1-2】在图1-5中，假定程序A、B和C的运行模式是相同的，在一个周期内运行时间为50ms，输入输出的时间（即等待时间）为100ms。试计算单道、两道和三道程序运行时，CPU的利用率。



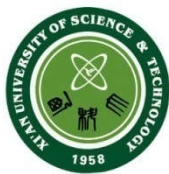


西安科技大学

XI' AN UNIVERSITY OF SCIENCE TECHNOLOGY

调度问题：

一般的，给定一个作业集合 J 以及资源约束条件 C 。如果 S 是对 J 中作业执行的一个编排，并且满足资源约束 C ，那么称 S 为作业集合 J 上的一个**调度** (Schedule)。



【例1-3】某系统供用户使用的内存空间为100K，系统配有4台磁带机。一批作业的运行和资源需求信息如下所示。试给出对这个作业集合的一种调度。

作业	进入时间	估计运行时间	内存需求	磁带机需求
J1	10:00	25min	15K	2台
J2	10:20	30min	60K	1台
J3	10:30	10min	50K	3台
J4	10:35	20min	10K	2台
J5	10:40	15min	30K	2台

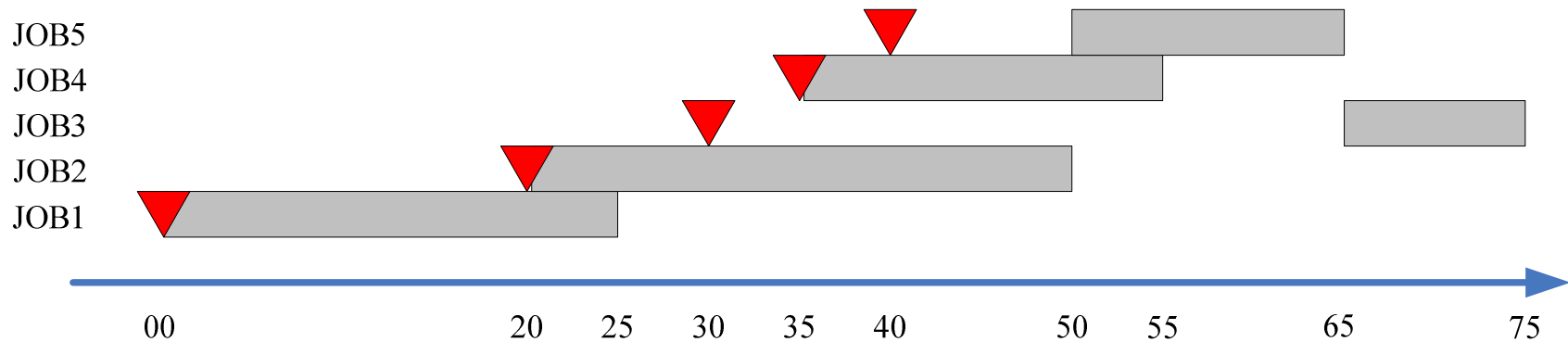


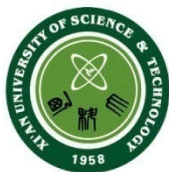
解：作业集合 $\mathcal{J} = \{J_1, J_2, J_3, J_4, J_5\}$

约束 C : $M(J_i) + M(J_j) + \dots + M(J_k) \leq 100K$

$$P(J_i) + P(J_j) + \dots + P(J_k) \leq 4$$

其中， J_i, J_j, \dots, J_k 是当前内存中的作业， $M(J_i)$ 和 $P(J_i)$ 分别表示 J_i 所占的内存和打印机资源。





4、分时系统

频分、时分、码分

基本思想是：对处理器资源进行时分复用（Time-sharing）。在时间域上，将处理器分若干时隙（Time slice），在每个时隙上，处理器为一个用户服务，如果时隙划分的足够小，从宏观来看，处理器就同时为多个用户提供交互计算服务。

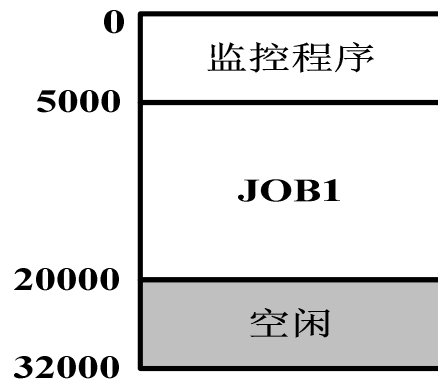
在分时系统中，如果 n 个用户同时请求服务，每个用户平均只能得到计算机有效速度的 $1/n$ 。但是由于人的响应时间相对较慢，所以一个设计良好的系统，其响应时间能够满足人们的交互式需要。

第一个分时操作系统是由麻省理工学院开发的兼容分时系统（CTSS, Compatible Time Sharing System），源于多路存取计算机项目，该系统最初是在1961年为IBM 709开发的，后来移植到IBM 7094中。

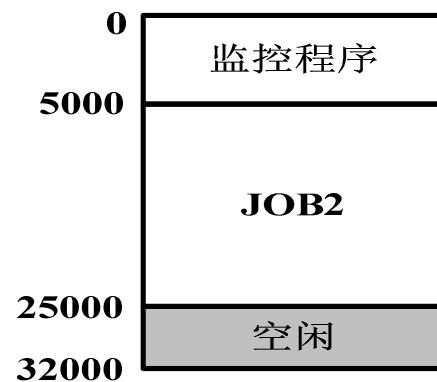


JOB1: 15000
JOB2: 20000
JOB3: 5000
JOB4: 10000

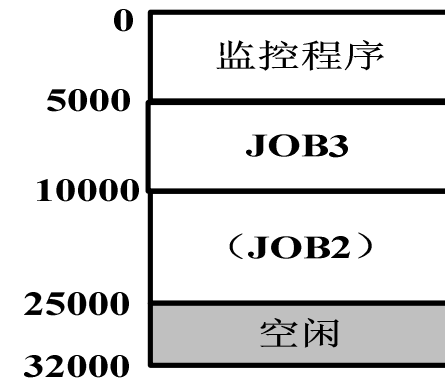
假设有4个交互用户，其存储器需求如下：



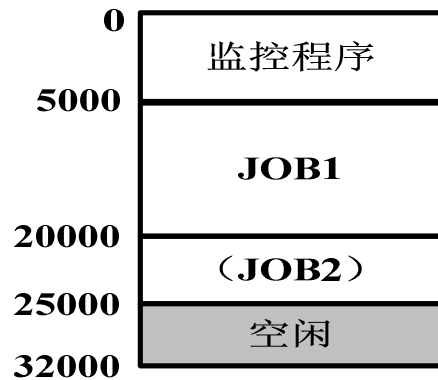
a)



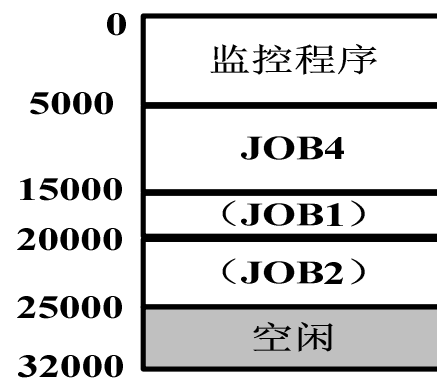
b)



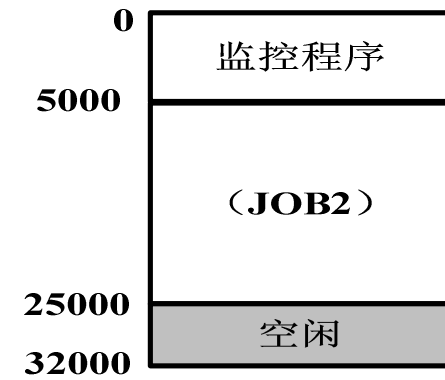
c)



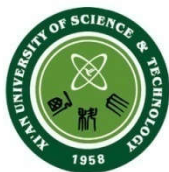
d)



e)



f)



1.2 操作系统的体系结构

1、简单结构

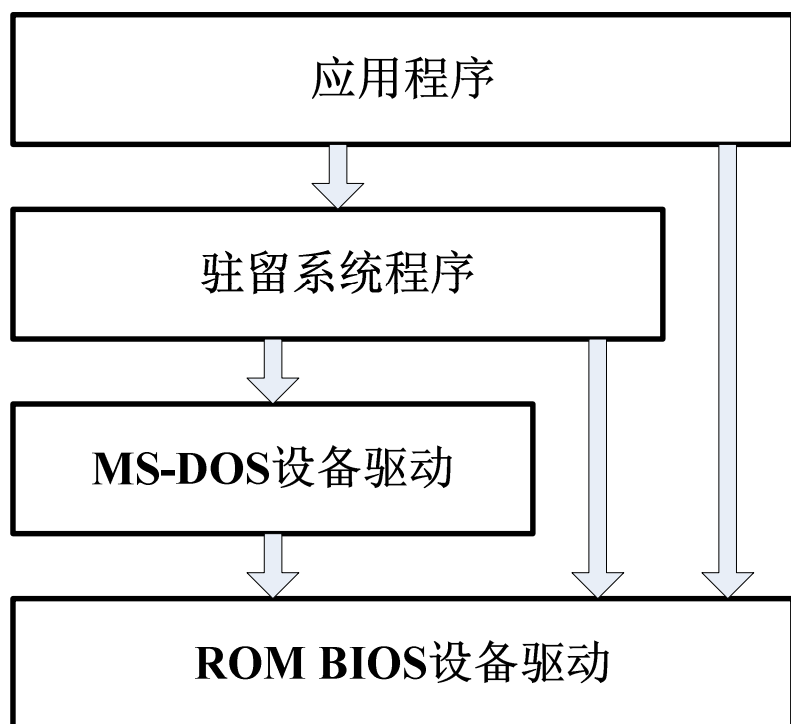
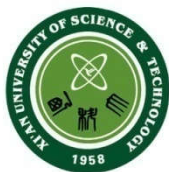


图1-8 MS-DOS层次化结构



图1-9 Unix系统结构



2、层次化结构

理想的层次化结构具有这样几个**优点**:

- ◆ 操作系统对计算机硬件以及使用这些硬件的应用程序施加了更严格的控制。
- ◆ 充分利用了信息隐蔽 (Information hiding) 基本原理
- ◆ 便于更改和维护操作系统内部工作方式, 有利于创建模块化操作系统

缺点:

- ◆ 难以合理的划分和定义各个层次
- ◆ 执行效率不高

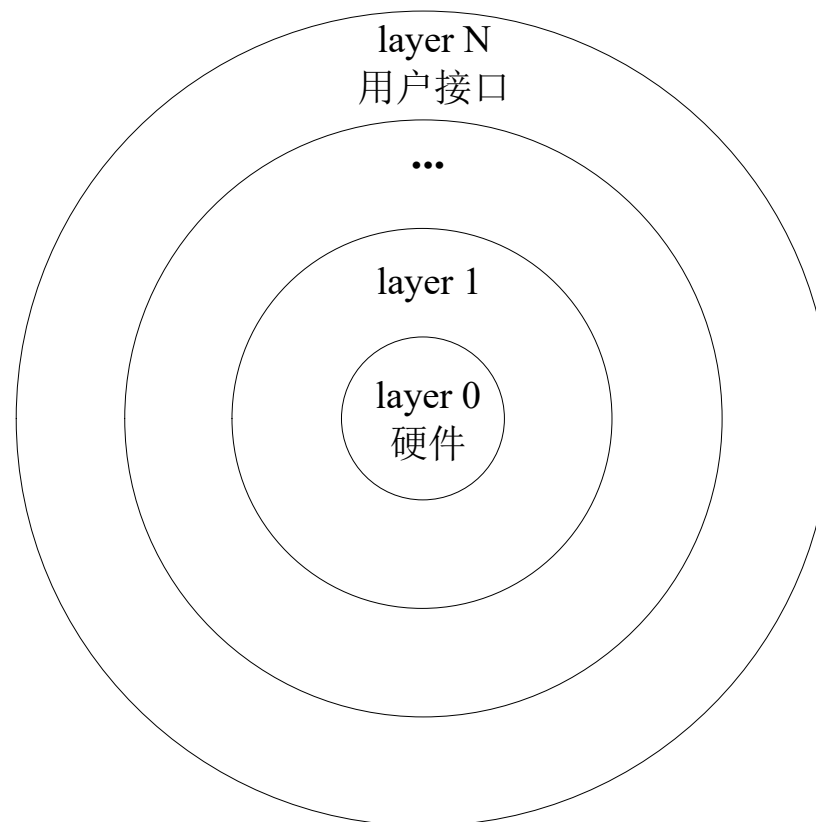


图1-10理想的层次化操作系统



3、微内核结构

基本思想：把OS中非核心组件从内核空间移动到用户空间，内核被大大简化，变成了微小内核。内核提供一个最小功能集：进程管理、内存管理和通信机制。

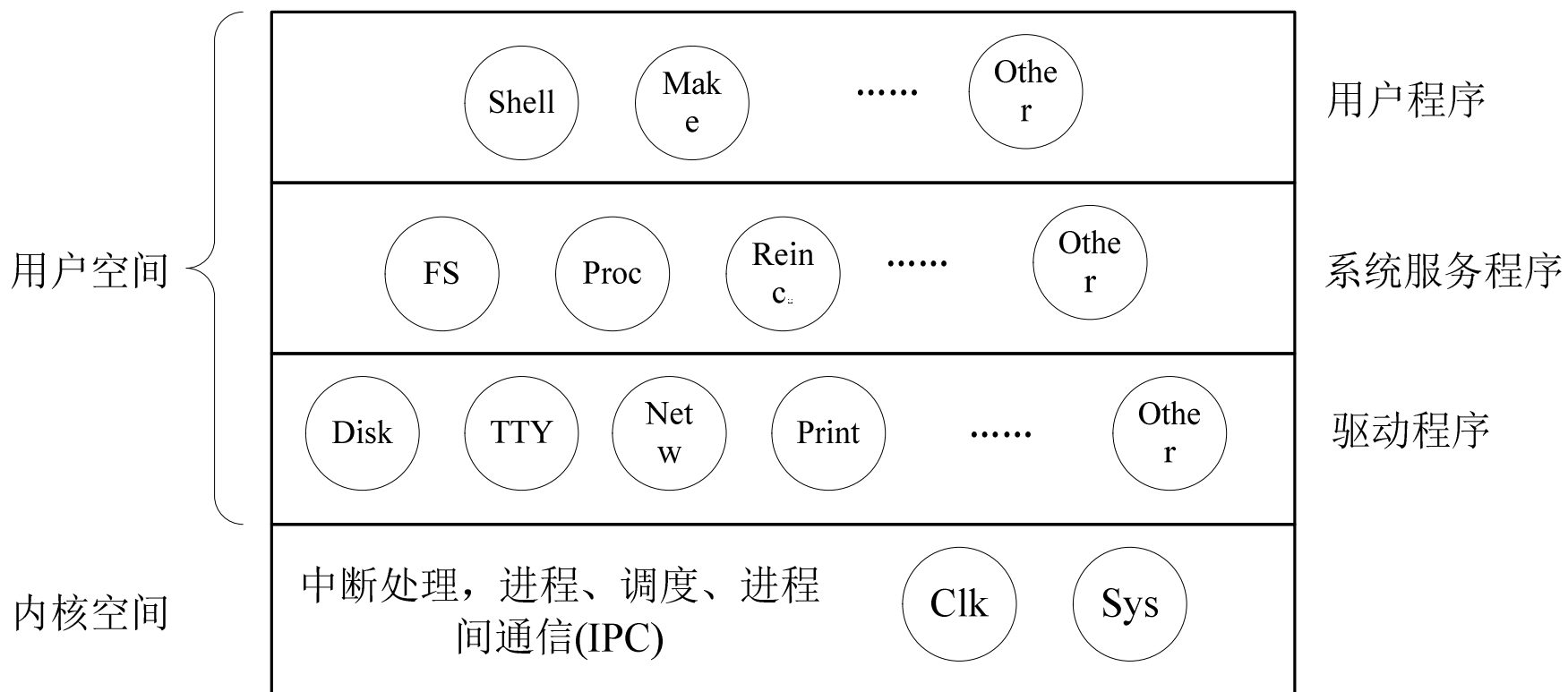
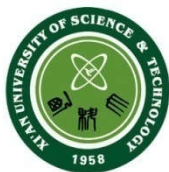


图1-11 Minix 3操作系统结构

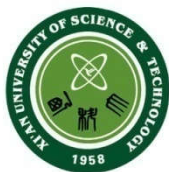


微内核结构的主要优点是易于实现操作系统的扩展：

- ◆ 所有的新服务被添加到用户空间，对内核不产生影响；
- ◆ 由于内核较小，当需要修改内核时，其改变也相应较少；
- ◆ 服务之间通过消息通信机制间接交互，因此服务之间的关联较为松散，有利于实现服务的更新和维护；
- ◆ 微内核系统更易于移植；
- ◆ 微内核系统提供了更高的安全性和可靠性。由于每个服务都是以一个用户态进程的形式运行的，如果一个服务失效，不会影响操作系统内核的运行

缺点：

- ◆ 执行效率问题：消息传递存在拷贝数据的开销；消息传递通常涉及进程切换的开销



4、模块结构

基本思想：由内核提供核心服务，其它特定服务功能以**动态**模块的形式提供。比如，可以把特定硬件的驱动程序添加到内核中，而把支持不同文件系统的模块以可加载模块的形式添加到系统中。

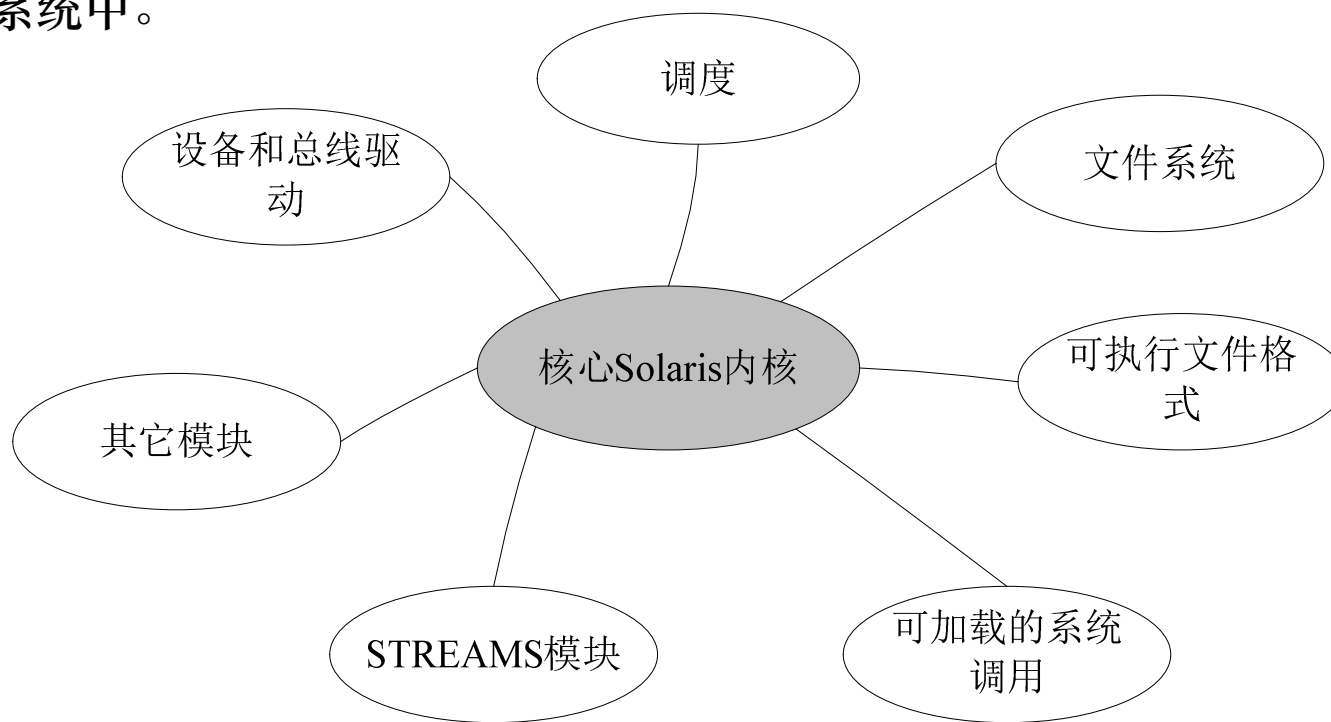
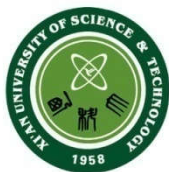


图1-12 Solaris可加载模块

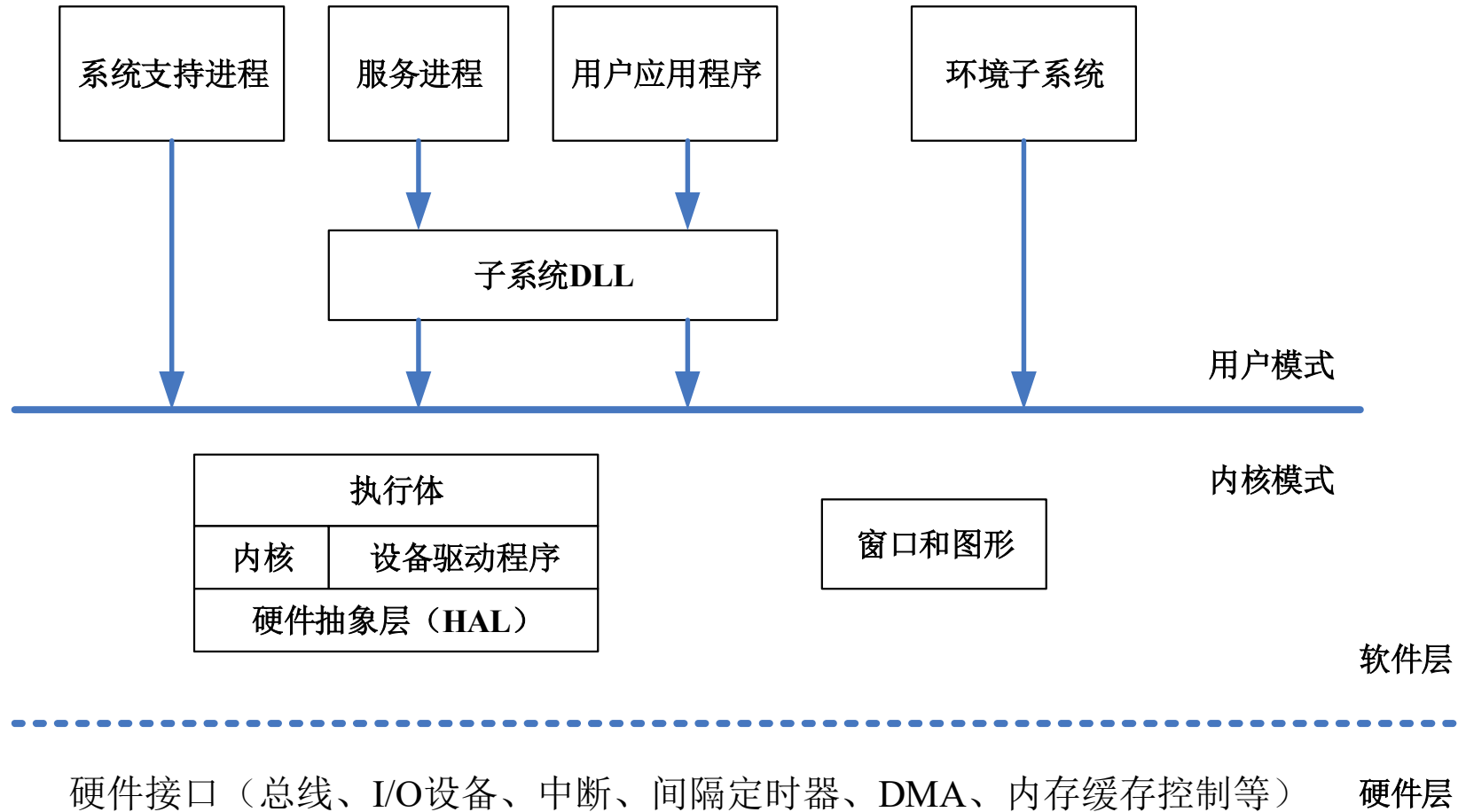


5、虚拟机

虚拟机 (VM, Virtual machine) 就是一个**计算机硬件系统的软件实现**，它能够**模拟和仿真特定计算机硬件**，并提供一套与底层硬件完全相同的接口，这样运行在物理计算机上的操作系统以及应用程序也同样可以运行在虚拟机上。

构造系统虚拟机的目的：

- ◆ 在实际硬件还没有得到之前就运行和测试应用程序
- ◆ 在一台计算机上运行虚拟机的多个实例，仿真多个计算环境，以提高计算资源的使用效率
- ◆ 提高系统的安全可靠性



精简的Windows操作系统架构

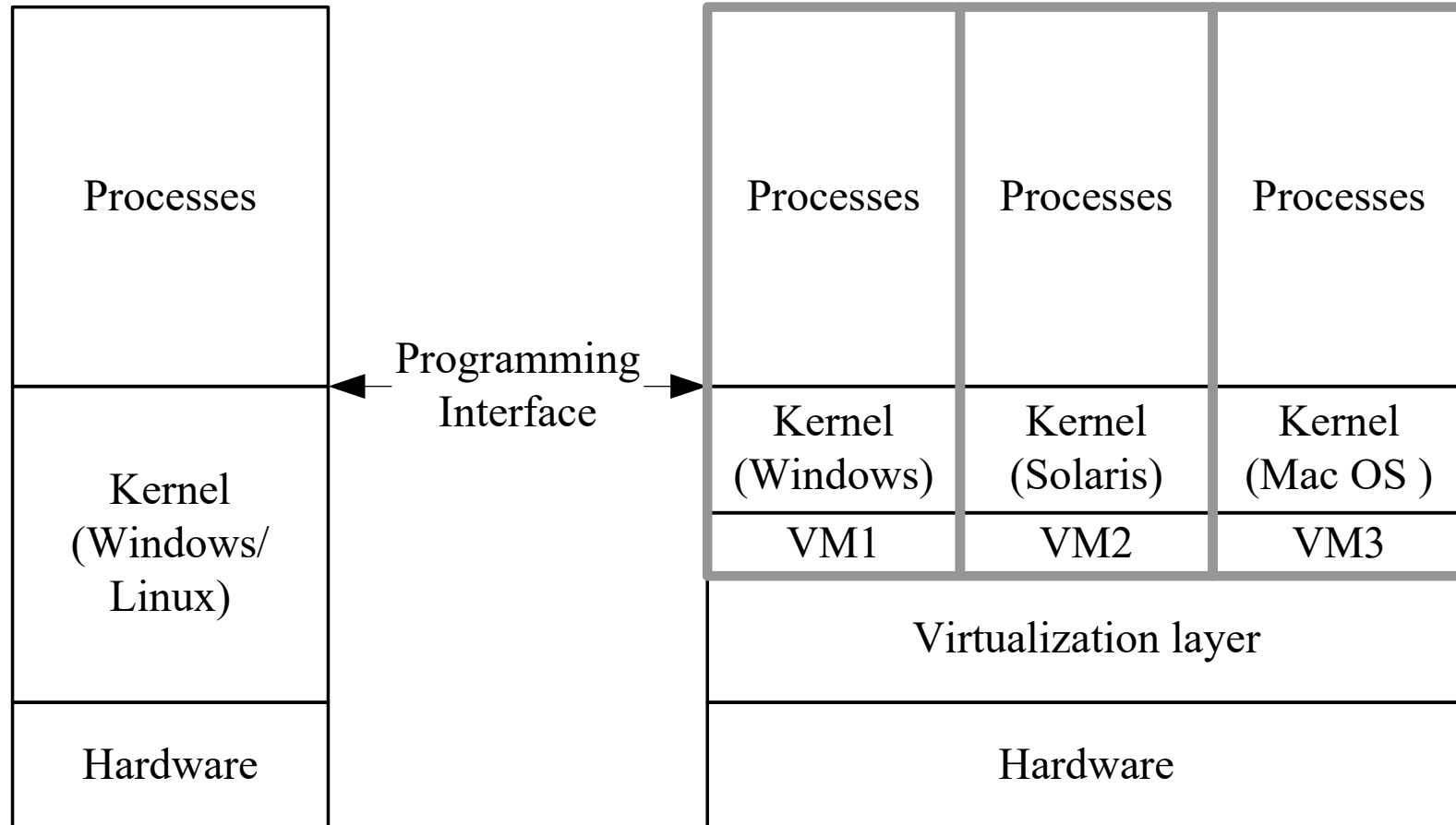


图1-13 采用虚拟机的操作系统结构



虚拟机也可以建立在宿主操作系统上 (Host OS)

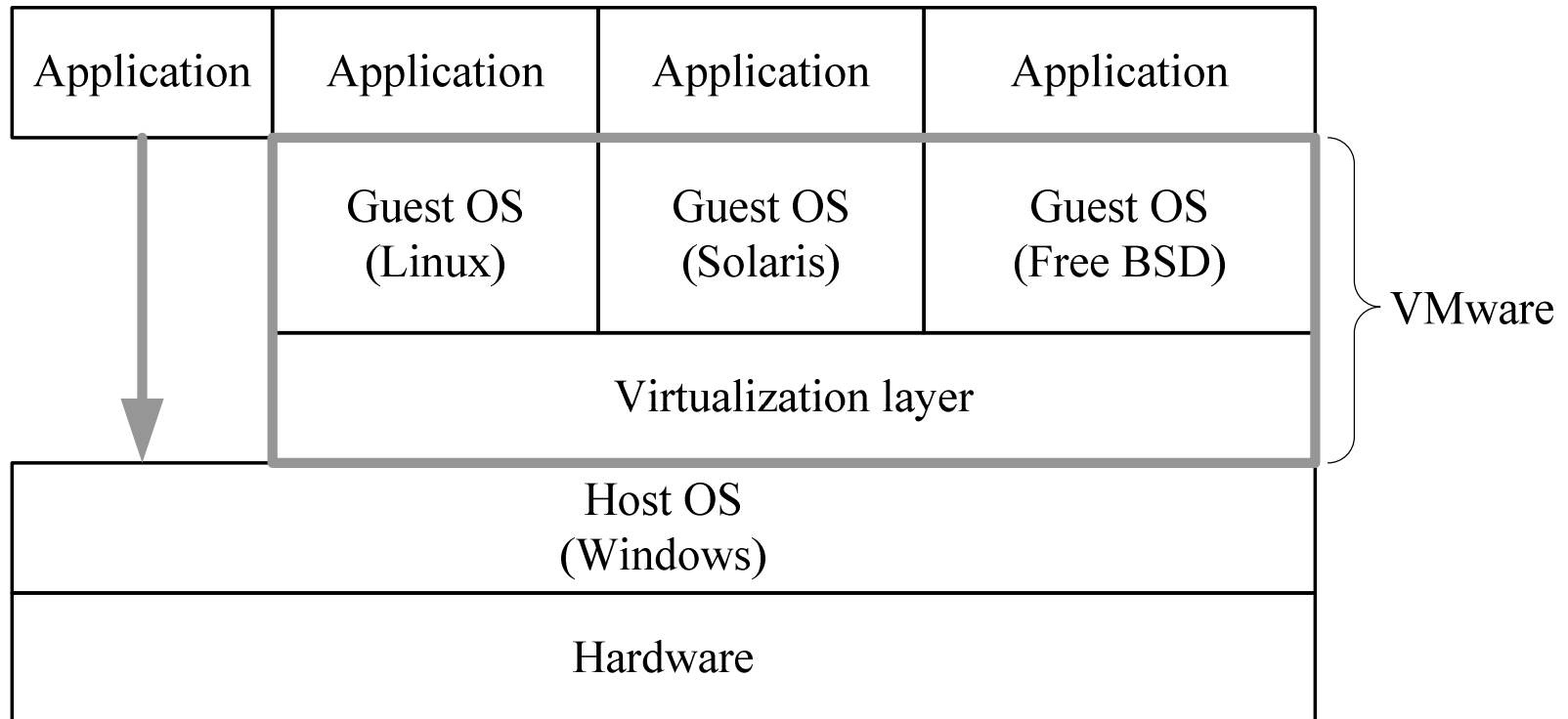
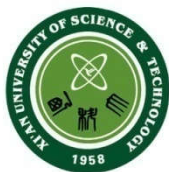


图1-14 VMware的结构



使用虚拟机的好处是：

- ◆每个虚拟机都与其它虚拟机完全隔离，对各种系统资源提供一个完全的保护；
- ◆多个虚拟机实例共享相同的硬件，构造了一个异质计算平台，提高了计算资源的使用效率；
- ◆虚拟机为操作系统研究和开发提供了一个理想的载体；
- ◆计算资源的虚拟化是云计算模式的关键技术。云计算模式要求在数据中心对各种计算资源进行统一管理，每个客户准备计算时，需要向云端发起任务请求，计算任务实际上是在云端完成的，这就要求云必须要为每个客户建立一个虚拟计算环境，并且保证每个虚拟环境的隔离性和安全性。



西安科技大学
XI' AN UNIVERSITY OF SCIENCE TECHNOLOGY

■ 作业

P20 4、5、6