



# 操作系统

---

教师：高雯

yayagaob@ecust.edu.cn



# 本课程的安排

---

- 理论（48学时线上学习）
- 最后成绩评定
  - 平时成绩占50%
  - 期末考试占50%



# 课程主要内容

---

👉 操作系统引论（第1章）

👉 进程管理（第2-3章）

👉 存储器管理（第4-5章）

👉 设备管理（第6章）

👉 文件管理（第7-8章）

👉 操作系统接口（第9章）

👉 Unix操作系统



# Operating System--OS

---

- 控制应用程序执行的软件
- 充当硬件系统和应用程序之间的界面的软件
- 为用户程序提供一个更好、更简单、更清晰的计算机模型
- 管理计算机系统的所有软硬件资源
- 举例



# 为什么要学习操作系统？（1）

---

- 需要总体上掌握系统设计
- 操作系统包括了所有软件设计/实现问题  
资源共享/管理、安全和身份验证、  
灵活性、健壮性、友好界面的设计
- 存在人们意识不到的大量“操作系统”
  - 掌上计算机操作系统（如：PDA；手机OS-iOS、Android、Symbian、Windows Phone、Blackberry OS）
  - 嵌入式系统（如：机顶盒；微波炉；汽车等）
  - 应用系统中的小操作系统（MINI-OS）



# 为什么要学习操作系统？（2）

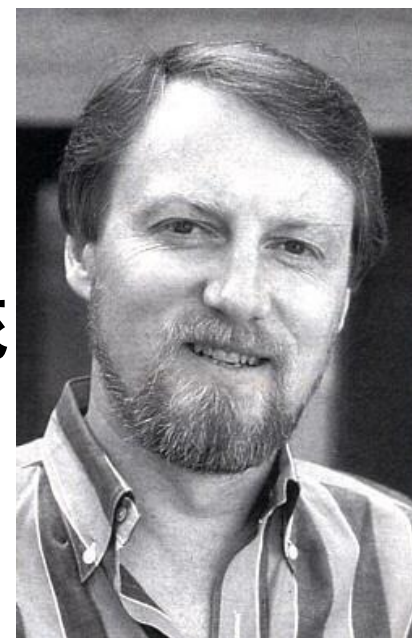
---

- 日常生活中并不总是使用Windows系列的操作  
系统（Unix、Linux、Apple Macintosh、IBM  
OS/2等）
- 常用的操作系统：
  - CP/M 操作系统
  - MS-DOS 操作系统
  - MS-Windows操作系统
  - MAC
  - UNIX 和 LINUX
  - 鸿蒙操作系统
  - VxWorks操作系统

# 为什么要学习操作系统？（3）

## ■ CP/M (Control Program/Microprocessor or Microcomputer)

- 微型计算机的第一个操作系统
- 加里·基尔代尔（开发PL/M语言）
- 诞生于1974年，8位微型机操作系统
- 主要功能
  - 能够进行文件管理
  - 具有磁盘驱动功能
  - 可以控制磁盘的输入输出
  - 可以控制显示器的显示以及打印机的输出
  - 版本：CP/M-80, CP/M-86, CP/M-68K





# 为什么要学习操作系统？（4）

---

- **MS-DOS (Microsoft Disk Operating System)**
  - 花2.5万美元从蒂姆·帕特森的QDOS购买改写的
  - 1981年8月DOS1.0~2000年9月MS-DOS8.0
  - fdisk、format、dir、cd、del、debug、copy .....
  - 我国推出基于MS-DOS的汉字磁盘操作系统CC-DOS





# 为什么要学习操作系统？（5）

---

## ■ MS-Windows系列

- 针对笔记本电脑的特有设计
- 基于应用服务的设计
- 用户的个性化设计
- 视听娱乐的优化
- 用户易用性的新引擎



# 为什么要学习操作系统？（6）

---

## ■ MAC

- 1976年，由斯蒂夫·乔布斯和斯蒂夫·沃兹尼亚克创立。世界上第一台拥有图形界面的个人电脑是由苹果公司设计的。
- 第一个开发了具有图形界面及使用鼠标和菜单的操作系统，Mac OS X v10.6 “Snow Leopard”



# 为什么要学习操作系统？（7）

---

## ■ UNIX

- 1969年Ken Thompson、Dennis Ritchie和Douglas McIlroy在AT&T的贝尔实验室开发的
- 技术成熟、可靠性高、网路和数据库功能强、伸缩性和开发性好等特色，可满足各行各业的实际需要
- 很多公司开发了自己的Unix：IBM的AIX、HP的HPUX、SUN的Solaris、加州大学伯克利分校的BSD产品：FreeBSD



# 为什么要学习操作系统？（8）

---

## ■ LINUX

- 1991年4月芬兰赫尔辛基大学的学生Linux Torvalds想设计一个代替UNIX的操作系统
- Linux的标志和吉祥物是一只名字叫做Tux的企鹅
- 有的Linux版本通过CD或USB就能使用，成为LiveCD
- Fedora Core、RedHat Linux、Ubuntu Linux、红旗Linux……



# 为什么要学习操作系统？（9）

## ■ 鸿蒙OS—华为公司陈海波

- 鸿蒙OS（HarmonyOS），2019年8月9日，华为在东莞举行华为开发者大会，正式发布鸿蒙操作系统。鸿蒙OS是一款“面向未来”的操作系统，一款基于微内核的面向全场景的分布式操作系统，它将适配手机、平板、电视、智能汽车、可穿戴设备等多种终端设备。
- 可按需扩展，实现更广泛的系统安全，主要用于物联网，特点是低时延，甚至可到毫秒级乃至亚毫秒级。

# 为什么要学习操作系统？（10）

## VxWorks操作系统

- 美国Wind River System推出的一个实时操作系统
- 嵌入式VxWorks的主要应用领域：
  - 数据网络：以太网交换机、路由器、远程接入服务器等
  - 远程通讯：电信用的专用分组交换机和自动呼叫分配器、蜂窝电话系统等
  - 医疗设备：放射理疗设备
  - 消费电子：个人数字助理等
  - 交通运输：导航系统、高速火车控制系统
  - 工业：机器人
  - 航空航天：卫星跟踪系统
  - 多媒体：电视会议设备
  - 计算机外围设备：X终端、I/O系统等演示



# 为什么要学习操作系统？（11）

---

- 涉及到计算机科学的很多领域：
  - 计算机体系结构/硬件
  - 软件设计
  - 程序设计语言
  - 数据结构
  - 算法



# 为什么要学习操作系统？（12）

---

## ■ 魔术师（抽象）

### ■ 操作系统具有一种魔力-----

- 将丑陋变成美好
- 将难变易
- 将少变为多----

提供了无限的CPUs、无限的内存 等等。

## ■ 管理者

### ■ 对所有计算机资源进行管理，目标----

- 公平
- 高效





# 为什么要学习操作系统？（13）

---

- 大型、复杂系统的系统设计

- 性能 与 方便使用

- 性能 与 简单性

- 硬件 与 软件

折衷权衡

**操作系统是目前最复杂的软件成分**



# 操作系统的规模（1）

---

- 简单的操作系统：  
一次一个应用程序在运行

例子：

- 较早期的计算机
- 早期的PC
- 嵌入式控制器（电梯，汽车，……）  
等等



# 操作系统的规模（2）

---

- **复杂的操作系统：**

- 多个应用程序共享计算机**

- 操作系统必须管理多个应用程序和多个用户之间的交互

- 例子：CPU，内存，I/O 设备：硬盘，打印机，... 操作系统管理所有这些资源，提供标准服务库

- **更复杂的操作系统：**

- 多个CPU，多个计算机**



# 第1章 操作系统引论

---

- 操作系统的目标和作用
- 操作系统的发展过程
- 操作系统的基本特征
- 操作系统的主要功能
- 操作系统的结构设计
- UNIX操作系统简介



# 1.1 操作系统的目标和作用

---

一、操作系统目的/目标

二、计算机系统组成

三、操作系统的作用

# 一、操作系统目的/目标



执行用户程序并使用户问题更易解决。

- 以一种有效的方式使用资源（**有效性**——**系统管理人员的观点**）。
- 使计算机系统更易于使用（**方便性**——**用户的观点**）。
- 采用新的OS结构, 易于功能的增、删、改（**可扩充性**）。
- 要求统一开放的环境, 各种类型的计算机硬件系统, 出自不同的厂家, 能通过网络集成并能够正确、有效地协同工作, 实现应用程序的移植（**开放性**）。



## 二、计算机系统集成

---

### 1. 计算机系统的组成

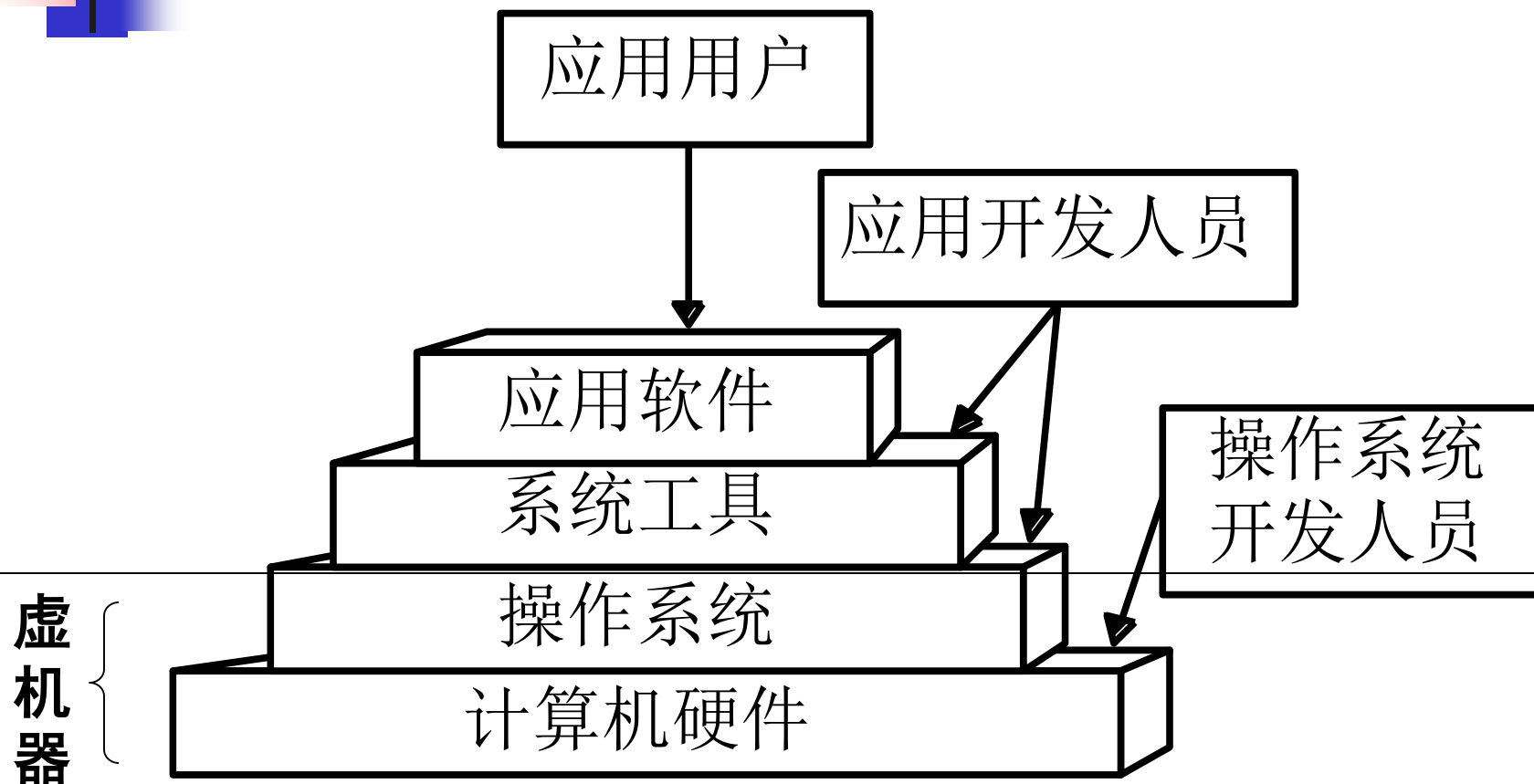
**硬件系统**（裸机）：CPU、存储器（主存、辅存）、I/O、I/O控制系统

**软件系统**：系统软件、应用软件

**系统软件**：管理计算机本身的操作。如操作系统、编译系统

**应用软件**：提供给用户进行解题。如，科学计算、事务管理

## 2. 计算机系统的层次结构



**操作系统的地位：** 紧贴系统硬件之上，所有其他软件之下（是其他软件的共同环境）





## 三、操作系统的作用（1）

---

- OS作为用户与计算机硬件系统之间的接口
  - 命令方式
  - 系统调用方式
  - 图形、窗口方式

通过以上接口OS提供如下功能：程序创建、程序执行、

I/O设备的访问、控制对文件的访问、系统访问、查错和纠错、簿记等。

# 三、操作系统的作用（1）

- OS作为用户与计算机硬件系统之间的接口

- 命令方式

- 系统

- 图形

- 通过以

- I/O

- 和结

```
命令提示符
G:\UC98>nake
G:\UC98>set path=G:\UC98\bin
G:\UC98>cl /c /IG:\UC98\include hellonsg.c
Microsoft (R) 32-bit C/C++ Optimizing Compiler Version 12.00.8168 for 80x86
Copyright (C) Microsoft Corp 1984-1998. All rights reserved.
hellonsg.c
G:\UC98>link /SUBSYSTEM:windows /machine:IX86 /LibPath:G:\UC98\lib G:\UC98\hello
msg.obj user32.lib kernel32.lib
Microsoft (R) Incremental Linker Version 6.00.8168
Copyright (C) Microsoft Corp 1992-1998. All rights reserved.
G:\UC98>
```

# 三、操作系统的作用（1）

## ■ OS作为用户与计算机硬件系统之间的接口

### ■ 命令方式

### ■ 系统调用方式

### ■ 图形、窗口方式

通过以上接口OS:

I/O设备的访问

和纠错、簿记:

Parent	Child
<pre>int main() {     pid_t pid;     char *message;     int n;     pid = fork();     if (pid &lt; 0) {         perror("fork failed");         exit(1);     }     if (pid == 0) {         message = "This is the child\n";         n = 6;     } else {         message = "This is the parent\n";         n = 3;     }     for(; n &gt; 0; n--) {         printf(message);         sleep(1);     }     return 0; }</pre>	<pre>int main() {     pid_t pid;     char *message;     int n;     pid = fork();     if (pid &lt; 0) {         perror("fork failed");         exit(1);     }     if (pid == 0) {         message = "This is the child\n";         n = 6;     } else {         message = "This is the parent\n";         n = 3;     }     for(; n &gt; 0; n--) {         printf(message);         sleep(1);     }     return 0; }</pre>



## 三、操作系统的作用（2）

---

- OS作为计算机系统资源的管理者
  - 管理对象包括：CPU、存储器、外部设备、信息（数据和软件）；
  - 管理的内容：资源的当前状态（数量和使用情况）、资源的分配、回收和访问操作，相应管理策略（包括用户权限） .....



## 三、操作系统的作用（3）

---

- OS实现了对计算机资源的抽象
  - 在裸机上添加：设备管理、文件管理、存储管理（针对内存和外存）、处理机管理（针对CPU）；
  - 即OS用作扩充机器（extended machine）/虚拟机（virtual machine）。



## 1.2 操作系统的发展过程

---

### 推动操作系统发展的主要动力

- 无操作系统的计算机系统
  - 人工操作方式
  - 脱机输入输出方式
- 批处理系统
  - 单道批处理系统
  - 多道批处理系统
- 分时系统
- 实时系统
- 微机操作系统
- 多处理机操作系统
- 网络操作系统
- 分布式操作系统
- 嵌入式操作系统
- 智能卡操作系统
- 云操作系统



## 1.2.1 推动操作系统发展的主要动力

---

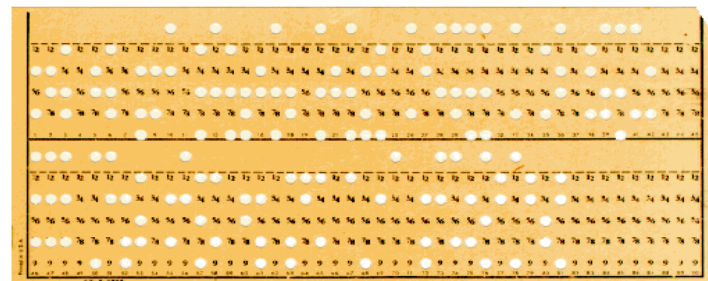
- ❑ 不断提高计算机资源利用率的需要
- ❑ 方便用户（如新服务，修补操作系统中的错误等）
- ❑ 器件的不断更新换代（如微机芯片的发展）
- ❑ 计算机体系结构的不断发展（单处理机系统→多处理机系统，计算机网络的出现）
- ❑ 不断提出的新的应用需求

## 1.2.2 无操作系统的计算机系统

### (1) 人工操作方式

#### ■ 工作流程：

- (1) 由程序员事先穿孔（对应程序和数据）
- (2) 将穿孔的纸带（卡片）装入纸带（卡片）输入机
- (3) 再启动输入机将程序和数据输入计算机，
- (4) 然后启动计算机运行
- (5) 运行完毕取走计算机结果
- (6) 下一位用户







# 人工操作方式带来的矛盾及解决方法

- **缺点：**

- (1) 用户独占全机（资源浪费）

- (2) CPU等待人工操作：装入、取走时CPU及内存等资源空闲。CPU既负责计算，又要负责传输。（属于单控制方式）

- **矛盾**

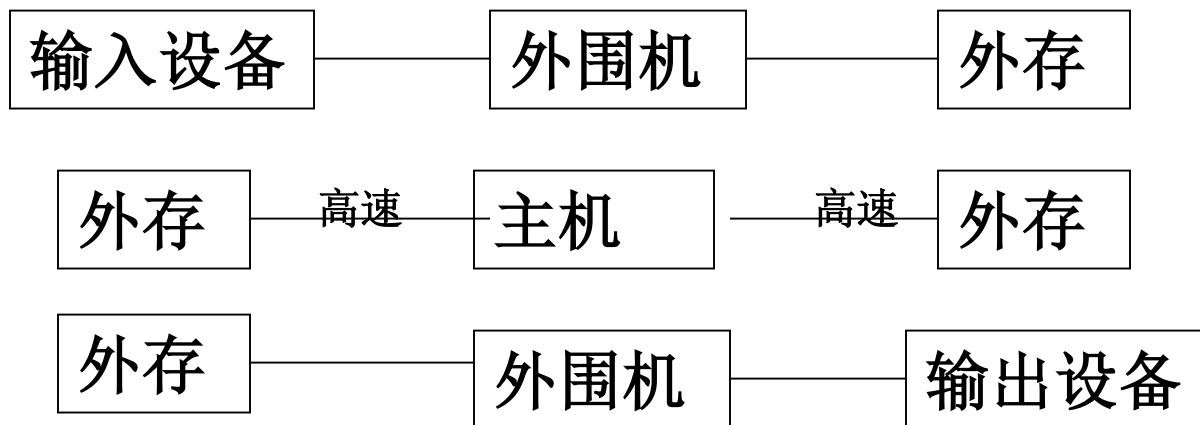
- 人机矛盾即人工操作方式与机器利用率的矛盾
  - CPU与I/O设备之间速度不匹配的矛盾

- **解决方法**

- 通道技术、缓冲技术
  - 脱机输入输出方式

## (2) 脱机输入/输出方式 (off-Line I/O)

### ■ 工作流程：（示意图）



### ■ 优点：

- (1) 减少了CPU的空闲时间。
- (2) 提高了I/O速度（与高速的磁盘打交道）

## 1.2.3 单道批处理系统

用户在一次解题或一个事务处理过程中要求计算机系统所做的全部工作。

### 1、工作流程：

- (1) **作业**合成一批输入到外存上，同时在系统中配上监控程序。
- (2) monitor将作业逐个送入内存并运行。

**2、特征：** (1) 自动性 (2) 顺序性 (3) 单道性

**3、优点：** 减少了人工操作的时间，提高机器的利用率和系统吞吐量。

**4、缺点：** 对某些作业来说，当它发出输入/输出请求后，CPU必须等待I/O的完成，特别因为I/O设备的低速性，从而使机器的利用率很低。



## 1.2.4 多道批处理系统（1）

---

- 多道程序设计技术（20世纪60年代中期引入）

- **特征：**

（1）多道性      （2）无序性      （3）调度性

- **优点：**

（1）资源（CPU、内存、I/O设备）的利用率高

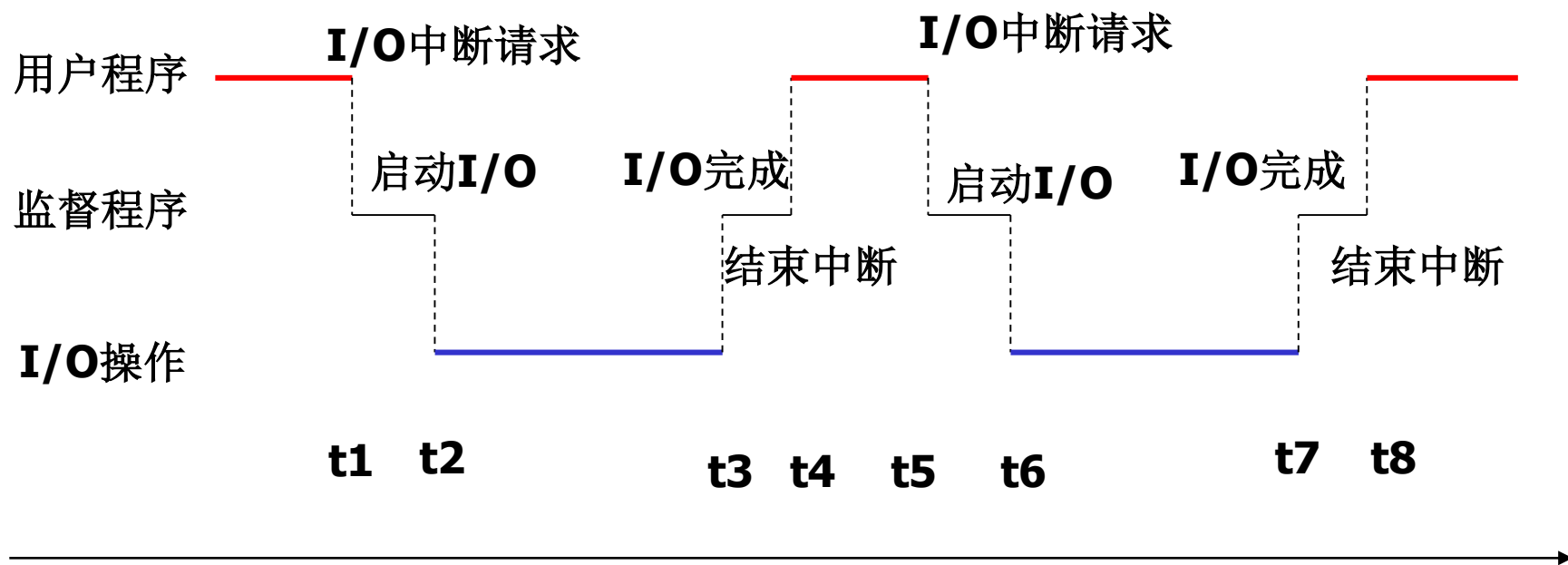
（2）系统吞吐量大

- **缺点：**（1）平均周转周期长      （2）无交互能力

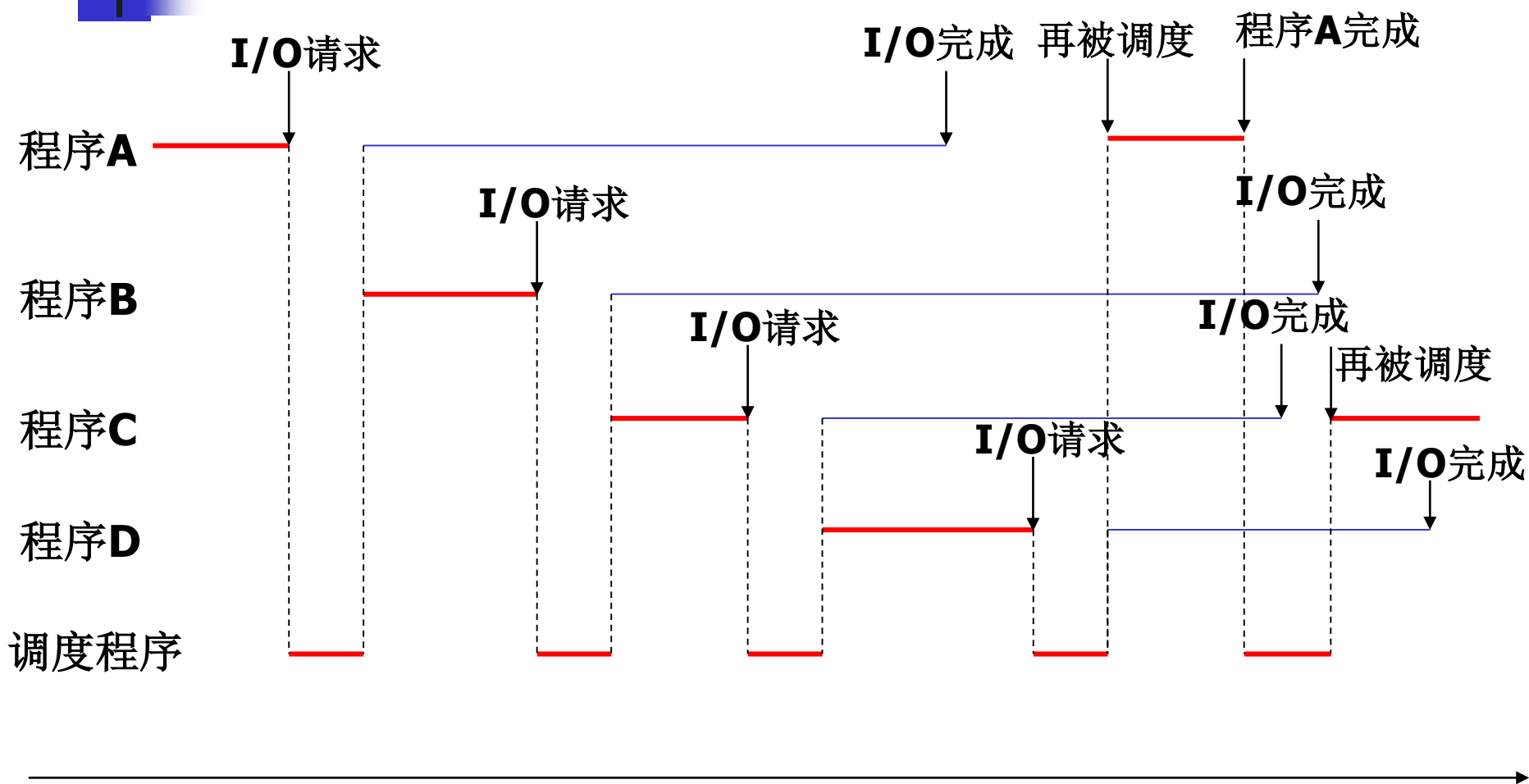
- **需解决问题**

处理机管理、内存管理、I/O设备管理、文件的组织和管理、作业管理、用户和系统的接口问题

# 单道程序运行情况



# 四道程序运行情况





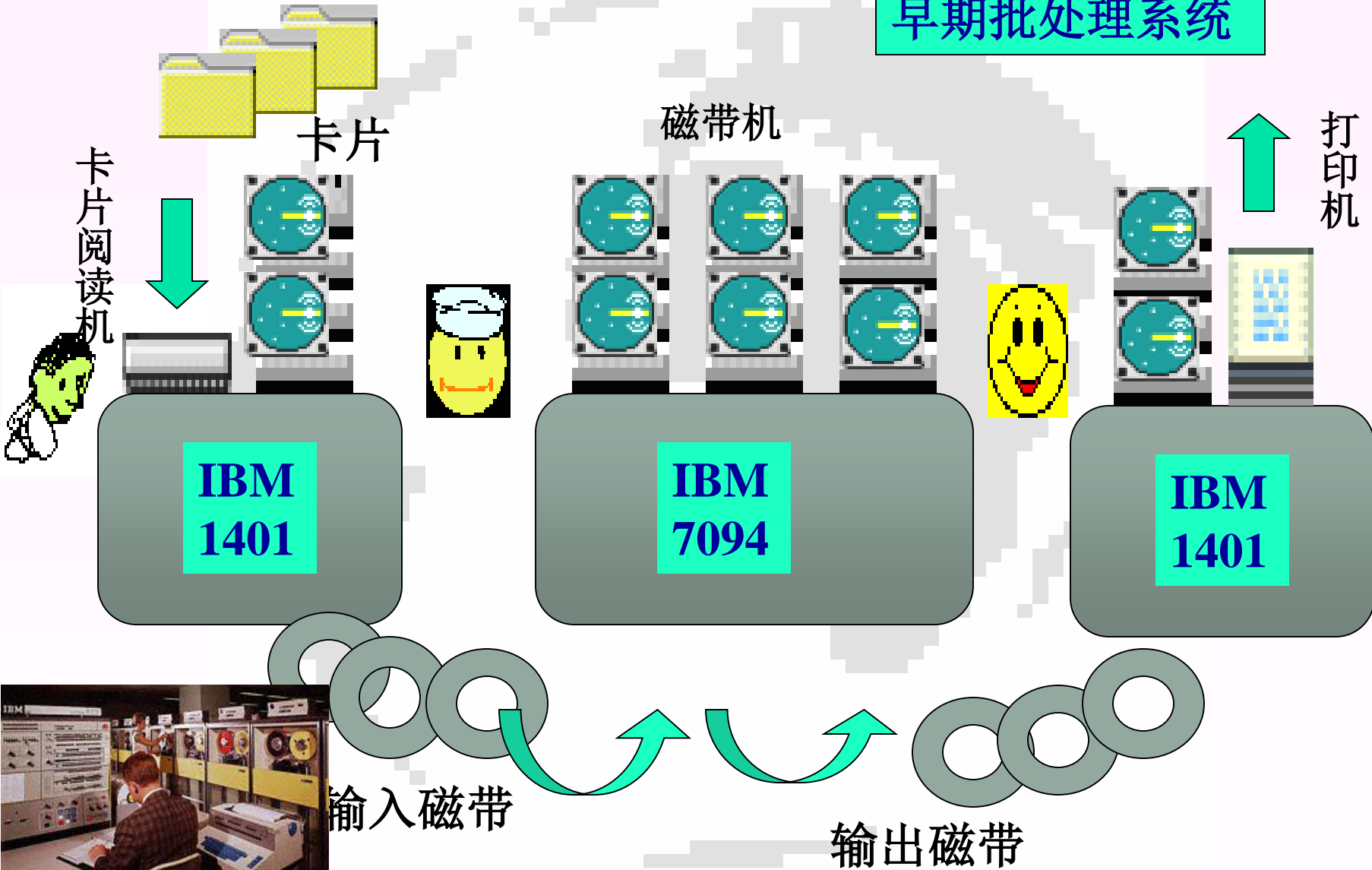
## 1.2.4 多道批处理系统（2）

### ■ 操作系统的形成

为解决上述问题，在计算机系统中增设一组软件，对这些问题加以有效、妥善地解决，这组软件应包括：控制和管理四大资源的软件、合理地各类作业进行调度的软件，以及方便用户使用计算机的软件，这样就形成了操作系统。

**操作系统：**是一组控制和管理计算机硬件和软件资源，合理地各类作业进行调度，以及方便用户使用的程序的集合。

# 早期批处理系统







# 批处理操作系统优缺点

---

- 优点：

- 作业流程自动化
- 效率高
- 吞吐量高

- 缺点：

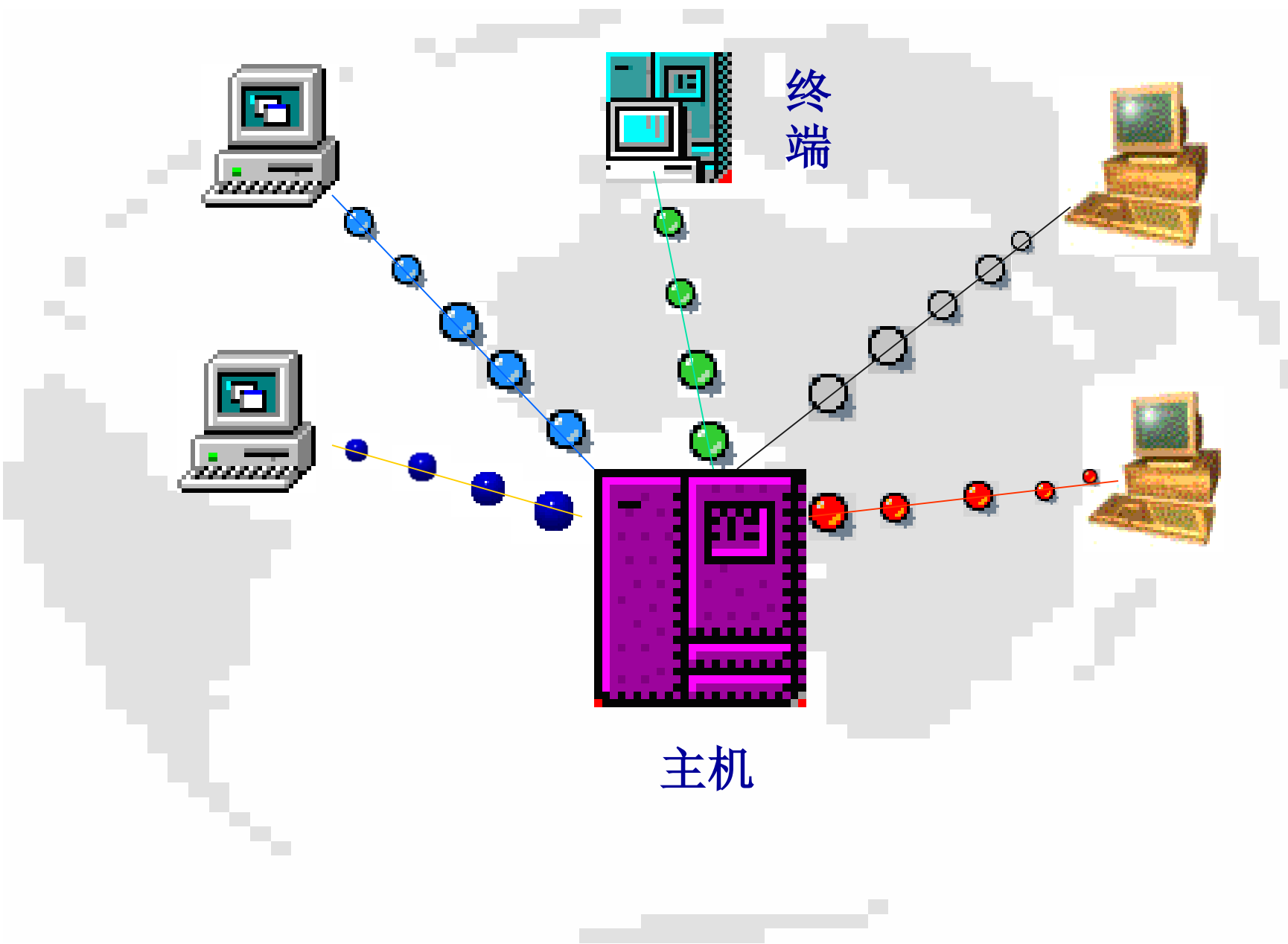
- 无交互手段
- 调试程序困难



## 1.2.5 分时系统（1）

---

- 分时系统(Time Sharing System)的产生  
人-机交互、共享主机、便于用户上机
- 分时系统的定义
  - 在一台主机上连接了多个带有显示器和键盘的终端，同时允许多个用户通过自己的终端，以交互方式使用计算机，共享主机中的资源。
- 第一台真正的分时OS (CTSS, Compatible Time Sharing System) 是由MIT开发成功的
- 见下图



终端

主机



## 1.2.5 分时系统（2）

---

### ■ 分时系统的基本思想

- 让多个用户同时通过终端使用系统，而OS则在系统内部按一定策略调度和处理用户程序。

### ■ 多道批处理系统和分时系统都使用了多道程序设计技术，但侧重点不同。

### ■ 实现中的关键问题

及时接收、及时处理

### ■ 分时系统的特征

多路性、独立性、及时性、交互性



## 1.2.6 实时系统 (Real Time System) (1)

**多道批处理系统：**令人较为满意的资源利用率

**分时系统：**令人较为满意的响应时间

**但不能满足实时控制和实时信息处理的需求**

### ■ **实时操作系统：**

指系统能及时（或即时）响应外部事件的请求，在规定的时间内完成对该事件的处理，并控制所有实时任务协调一致地运行。

### ■ **实时操作系统的应用领域**

有实时要求（有严格确定的时间限制）的领域（过程控制及事务处理）——多道批处理系统和分时系统不能满足



## 1.2.6 实时系统 (Real Time System) (2)

---

### ■ 实时系统分类：

- (1) 实时控制系统
- (2) 实时信息处理系统
- (3) 多媒体系统
- (4) 嵌入式系统

### ■ 实时任务：

周期性/非周期性实时任务（截止时间）

硬实时任务/软实时任务

### ■ 实时系统与分时系统特征的比较：

多路性、独立性、及时性、交互性、可靠性



# 三种基本的操作系统

---

## ■ 三种基本的操作系统

批处理操作系统、分时操作系统和实时操作系统

## ■ 通用操作系统

如果一个操作系统兼有批处理、分时和实时系统三者或二者的功能，则称该操作系统为通用操作系统。



## 1.2.7 微机操作系统

---

### ■ 微机操作系统的特征

- **应用领域**：事务处理、个人娱乐 等
- **系统要求**：使用方便、支持多种硬件和外部设备（多媒体设备、网络、远程通信）、效率不必很高。

### ■ 常用的微机操作系统

- **单用户单任务**：CP/M, MS- DOS
- **单用户多任务**：Windows
- **多用户多任务**：Solaris OS, Linux。





## 1.2.8 多处理机操作系统

---

- 一种获得大量联合计算能力的操作系统
- 常用的方式：将多个CPU连接成单个的系统。
- 依据连接和共享的方式的不同，这些系统称为**并行计算机、多计算机或多处理器**。
- 需要专门的操作系统：配有通信、连接和一致性等专门功能的操作系统



## 1.2.9 网络操作系统（1）

---

### ■ 计算机网络

- 将一些具有独立处理能力的计算机通过传输媒体把它们互联起来，能够实现通信、资源共享和相互合作的系统。

### ■ 计算机网络的类型（按拓扑结构分为：星型、树型、总线型、环型、网状型）

### ■ 网络操作系统

- 为网络用户提供各种服务的软件和有关协议的集合，其目的是让网络上各计算机能方便、有效地共享网络资源。



## 1.2.9 网络操作系统（2）

---

### ■ 网络OS的模式

- 集中模式：信息的处理和控制都是集中的，由分时系统加上网络功能演变而成。
- 客户端/服务器模式：具有分布处理和集中控制的特征。
- 对等模式：具有分布处理和分布控制的特征。

### ■ 网络OS的功能

通信、资源管理、网络服务、网络管理、互操作能力



## 1.2.10 分布式操作系统（1）

---

- **分布式系统**：由多个分散的计算机经互联网络连接而成的计算机系统，能将任务处理和控制进行分散/分布的系统（相对于集中式）。
- **分布式系统是多机系统特别是并行处理系统的一种新形式，是计算机网络的高级发展阶段。**
- **分布式操作系统**：为分布式计算机系统配置的操作系统，自动实现全系统范围的任务分配并自动调度各处理机的工作负载，为用户提供一个方便良好的用机环境。



## 1.2.10 分布式操作系统（2）

### ■ 分布式OS的五个主要特征

- 资源共享：硬件资源共享、软件资源共享
- 开放性：包括可伸缩性、可移植性和互操作性
- 并发性：内在的特征
- 容错性：2个基本方法——硬件冗余和软件恢复
- 透明性：包括访问透明性、位置透明性、迁移透明性、副本透明性、并发透明性和并行透明性。

**分布式系统基本特征——处理上的分布，即功能和任务的分布。**



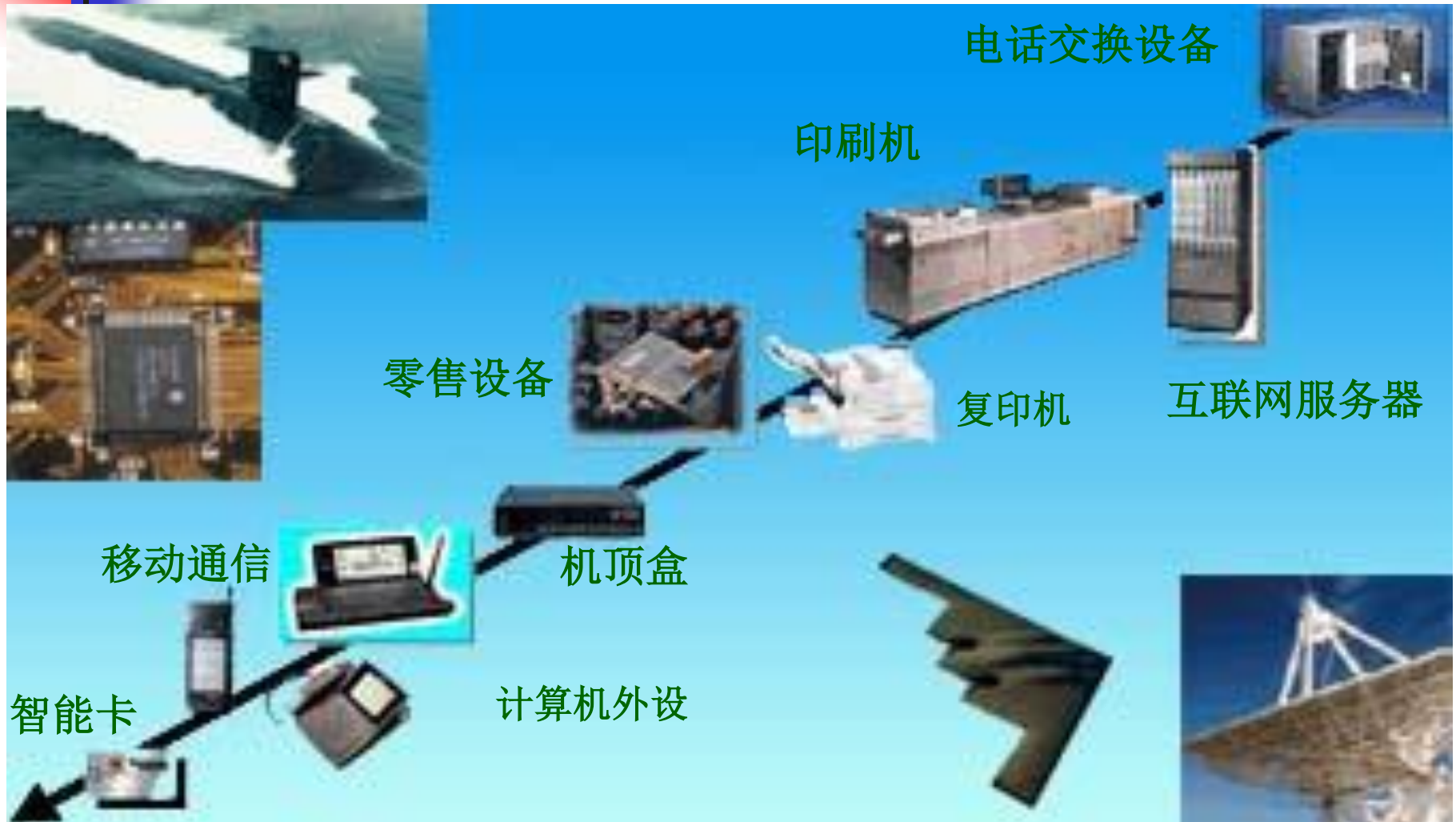
## 1.2.11 嵌入式操作系统

---

### ■ EOS, Embedded Operating System

- 用途广泛的系统软件，过去主要应用于工业控制和国防系统领域。
- 指运行在设备、装置、系统中，对整个系统及所有操作部件、装置等资源进行统一协调、处理、指挥和控制的计算机系统软件。
- 必须体现其所在系统的特征，能够通过装卸某些模块来达到系统所要求的功能。
- 除具备一般OS最基本的功能外，还有自己的特点

# 嵌入式操作系统 (Embedded OS) 的应用





## 1. 2. 12 智能卡操作系统

---

- 最小的操作系统运行在智能卡上
- 智能卡是一种包含有一块CPU芯片的信用卡，运行能耗和存储空间的限制非常严格。
  - 具有单项功能——如电子支付
  - 具有多项功能——如校园卡



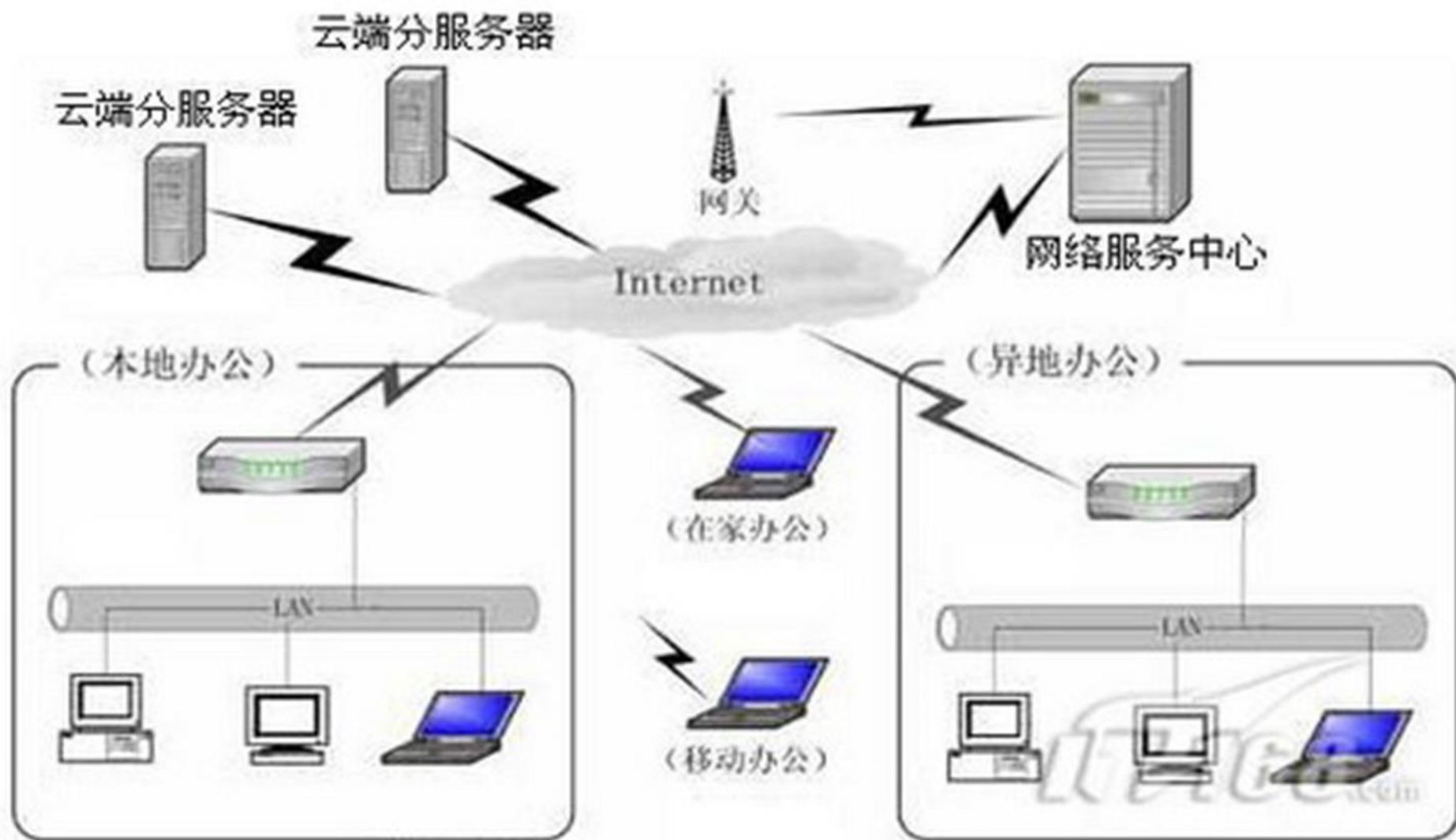


## 1.2.13 云操作系统

---

- **云计算：**通过网络以按需、易扩展的方式获得所需的资源（硬件、平台、软件）和服务。云计算是并行计算、分布式计算和网格计算的发展。提供资源的网络被称为“云”；
- **云计算的原理：**通过使计算分布在大量的分布式计算机上，而非本地计算机或远程服务器中，企业数据中心的运行将更与互联网相似，这使企业能够将资源切换到需要的应用上，根据需求访问计算机和存储系统。

## 1.2.13 云操作系统



## 1.2.13 云操作系统

### ■ 云计算操作系统：

- 非传统的操作系统，是一个虚拟的操作系统，把操作系统变成一种Web服务，核心技术是虚拟化，通过虚拟化进行相关的资源分配；
- 无需在本地计算机上安装操作系统，只需在裸机上安装特定的浏览器并且可以上网，就能通过浏览器使用云计算操作系统，像使用其他Web服务一样。





## 1. 2. 13 云操作系统

---

### ■ 云操作系统的优势

- 基于云计算和云存储，可管理和利用更庞大的资源，为用户提供更多的服务；
- 在可以上网的任何计算机上，用户都可以使用自己定制的同个云计算操作系统，无需各种繁琐的文件拷贝和资源转移；
- 云计算操作系统不与Windows、Linux、Unix等操作系统冲突，可以在这些操作系统上运行。



## 1. 2. 13 云操作系统

---

### ■ 云操作系统举例

- 2009年4月22日，推出业内首款云计算操作系统  
VMware vSphere 4;
- 微软Windows Azure;
- Google Chrome OS;
- 浪潮云海OS



## 1. 2. 13 云操作系统

---

### ■ 云OS和传统OS的比较

- 管理的资源不同：传统OS管理的是一台或多台电脑；云OS管理的是云资源
- 运行的例程不同：传统OS包括调度器、内存分配、文件系统和中断处理程序；云OS提供了更多的附加功能：虚拟机的分配和释放、任务的分配和融合等
- 接口功能不同：传统OS提供管理底层硬件的库函数；云OS提供基于网络的接口管理资源
- 包含的分布式库不同：传统OS包括标准的分布式库和软件包；云OS为分布式应用提供了自主扩展和灵活调度的软件支持



## 1.3 操作系统的基本特征

---

- 并发 (concurrency)
- 共享 (sharing)
- 虚拟 (virtual)
- 异步性 (asynchronism)



# 并发 (concurrency) (1)

---

■ **并发**：指在计算机系统中同时存在着多道运行的程序（**进程**）

宏观上：多道程序**同时**在执行

微观上：任何时刻**只有一道**程序在执行，  
即微观上多道程序在CPU上**轮流（交替）**执行  
（单机）

■ **并行** (parallel)：与并发相似，指多道程序在同一时刻执行，但需多个硬件支持。





# 并发 (concurrency) (2)

---

## ■ 引入进程

程序和进程：在多道程序系统中，程序不能独立运行，更不能和其它程序并发执行。引入进程的目的，就是为了使多个程序并发执行。

## ■ 引入线程

进程是OS中拥有资源并独立运行的基本单位。引入线程后，把进程作为分配资源的基本单位，把线程作为独立运行和独立调度的基本单位，进一步提高系统的并发性。



## 并发 (concurrency) (3)

---

**注：1、**程序的并发执行能有效改善系统资源的利用率，但使系统复杂化，因此操作系统必须对并发活动进行控制和管理。

**2、**并发是操作系统最重要的特征，其它特征均以并发为前提。



# 共享 (sharing)

---

■指系统中的资源不再为某道程序所独占，而是供多道程序共同使用。

■资源共享方式：互斥共享方式、同时访问方式

■并发和共享是操作系统的两个最基本的特征，二者互为存在条件：

- 资源的共享是以程序（进程）的并发执行为条件。
- 程序的并发执行也以资源的共享为条件。



# 虚拟(virtual)

---

- 通过某种技术把一个**物理实体**映射为若干个对应的**逻辑实体**——分时（时分复用技术）或分空间（空分复用技术）。
- 虚拟是操作系统管理系统资源的重要手段，可提高资源利用率。如：虚拟处理机（分时）、虚拟设备（分时）、虚拟磁盘（分空间）、虚拟存储器（分空间、分时）



# 异步性 (asynchronism)

---

- 也称不确定性，指进程在执行中，其执行时间、顺序、向前推进的速度和完成的时间等都是不可预知的。
- 只要在OS中配置有完善的进程同步机制，且运行环境相同，作业经多次运行都会获得完全相同的结果。所以，异步运行方式是允许的，是OS的一个重要特征。



## 1.4 操作系统的功能

---

- 处理机管理
- 存储器管理
- 设备管理
- 文件管理
- OS与用户之间的接口
- 现代OS的新功能



# 一、处理机管理的功能

---

- 处理机管理的主要任务：

- 是对处理机进行**分配**
- 对处理机运行**进行有效的控制和管理**

**注：**处理机的分配和运行以进程为基本单位，因此  
对处理机的管理可归结为对进程的管理

- 处理机管理的功能

- 进程控制
- 进程同步
- 进程通信
- 调度



## 二、存储器管理的功能

---

### ■ 主要任务

- 为多道程序的运行提供良好的环境
- 方便用户使用存储器
- 提高存储器的利用率
- 从逻辑上扩充内存

### ■ 功能

- 内存分配（静态分配、动态分配）
- 内存保护
- 地址映射
- 内存扩充





## 三、设备管理的功能

---

### ■ 主要任务

- 完成用户提出的I/O请求
- 为用户分配I/O设备
- 提高I/O设备的利用率及速度
- 方便用户使用I/O设备

### ■ 功能

- 缓冲管理
- 设备分配
- 设备处理
- 虚拟设备



## 四、文件管理的功能

---

- 主要任务

- 对用户文件和系统文件进行管理
- 方便用户使用文件
- 保证文件的安全性

- 功能

- 文件存储空间的管理
- 目录管理
- 文件的读、写管理和保护



## 五、OS与用户接口管理的功能

---

- 主要任务

- 方便用户使用操作系统

- 功能

- 用户接口（联机用户接口-命令方式、图形用户接口，脱机用户接口）
  - 程序接口（系统调用）



## 六、现代OS的新功能

---

- 增加了新功能
  - 系统安全
  - 网络的功能和服务
  - 支持多媒体



# 1.5 操作系统的结构设计

---

操作系统的结构设计经历了以下几代：

- 传统的操作系统结构
  - 无结构操作系统
  - 模块化结构OS
  - 分层式结构OS
- 客户/服务器模式
- 面向对象的程序设计
- 微内核OS结构



# 无结构操作系统

---

- OS是由众多的过程直接构成，各过程之间可相互调用，但OS内部不存在任何结构，所以这种OS是无结构的，又称为整体系统结构。

- 缺点：

既庞大又杂乱，缺乏清晰的程序结构；程序错误多，调试难、阅读难、理解难、维护难。



# 模块化结构操作系统（1）

- OS是采用“模块化程序设计”技术，按其功能划分为若干个独立的模块，管理相应的功能，同时规定好各模块之间的接口，以实现它们的交互，对较大模块又可按子功能进一步细分下去。所以这种OS称为模块化OS结构。
- 模块的独立性
  - 关键：模块的划分和规定好模块之间的接口
  - 衡量独立性的两个标准：
    - 内聚性：指模块内部各部分间联系的紧密程度
    - 耦合度：指模块间相互联系和相互影响的程度



# 模块化结构操作系统（2）

---

## ■ 优点

- 提高了OS设计的正确性、可理解性和可维护性
- 增强了OS的可适用性
- 加速了OS的开发过程

## ■ 缺点：

- 模块及接口划分较困难
- 未区别共享资源和独占资源
- 由于管理上的差异，使OS结构变得不够清晰





# 分层式结构操作系统（1）

---

- 分层式结构是对模块化结构的一种改进，它按分层式结构设计的基本原则，将OS 划分为若干个层次，每一层都只能使用其底层所提供的功能和服务，从硬件开始，在其上面一层一层地自底向上增添相应功能的软件，这种OS结构称为分层式OS结构。
- 特点：
  - 每一步设计都建立在可靠的基础上，结构更清晰
  - 调试和验证更容易，正确性更高



# 分层式结构操作系统（2）

---

- 分层式结构的优点：
  - 易保证系统的正确性
  - 易扩充和易维护
- 缺点：
  - 系统效率降低：层次结构是分层单向依赖的，必须在相邻层次之间建立层次间的通信机制，增加了系统开销。



# 客户/服务器模式（1）

---

## ■ 客户/服务器模式的组成

- 客户机：每台客户机都是一个自主计算机，客户进程在运行
- 服务器：一台规模较大的机器，为网上所有的用户提供一种或多种服务。
- 网络系统：用于连接所有客户机和服务器，实现它们之间的通信和网络资源共享。

## ■ 客户/服务器之间的交互：

- 客户发送请求消息、服务器接收消息
- 服务器回送消息、客户机接收消息



# 客户/服务器模式（2）

---

- 客户/服务器模式的优点

- 数据的分布处理和存储
- 便于集中管理
- 灵活性和可扩充性
- 易于改编应用软件

- 不足之处

- 存在着不可靠性和瓶颈问题
- 改进方法：在网络中配置多个服务器，并采取相应的安全措施。



# 面向对象的程序设计

---

- 面向对象技术的基本概念
  - 对象
  - 类
  - 继承
- 面向对象技术的优点
  - 通过“重用”提高产品质量和生产率。
  - 使系统具有更好的易修改性和易扩展性。
  - 更易于保证“正确性”和“可靠性”



# 微内核的OS结构（1）

---

- 微内核技术的主要思想

在OS内核中只留下一些最基本的功能，而将其他服务分离出去，由工作在用户态下的进程来实现，形成“客户/服务器”模式。

客户进程可通过内核向服务器进程发送请求，以取得OS的服务。



# 微内核的OS结构（2）

---

- 微内核

精心设计的，能实现现代OS核心功能的小型内核，它小而精炼，运行在核心态下，开机后常驻内存，不会因内存紧张而换出，它为构建通用OS提供了一个重要基础。

- Windows 2000/XP 采用了微内核结构



# 微内核的OS结构（3）

---

- 微内核OS的基本概念

在进行现代OS结构设计时，大多采用基于客户/服务器模式的微内核结构，将OS分为两部分：微内核和多个服务器。

- 足够小的内核
- 基于客户/服务器模式
- 应用“机制与策略分离”原理
- 采用面向对象技术





# 微内核的OS结构（4）

---

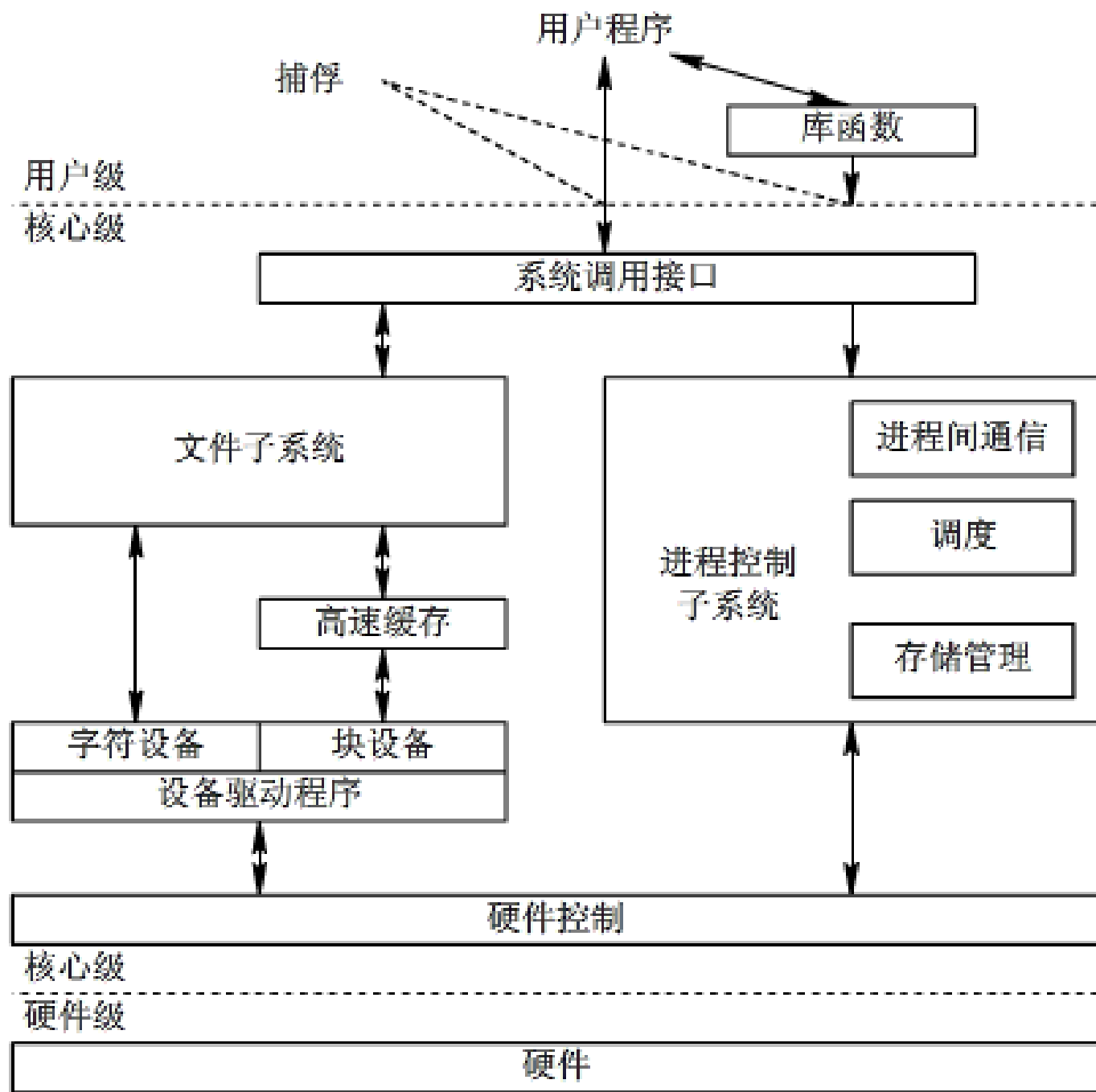
- 微内核的基本功能
  - 进程（线程）管理
  - 低级存储器管理
  - 中断和陷入处理
- 优点：微内核OS结构是建立在模块化、层次化结构的基础上的，并采用了C/S模式和OO的程序设计技术
  - 提高了系统的可扩展性
  - 增强了系统的可靠性
  - 提供了对分布式系统的支持
  - 融入了面向对象技术



## 1.6 UNIX系统简介

---

- UNIX系统的发展
- UNIX系统的特性
  - 开放性（系统遵循国际标准规范）
  - 多用户、多任务环境
  - 功能强大、实现高效
  - 提供丰富的网络功能
  - 支持多处理机的功能
- UNIX系统的内核结构（见下页）

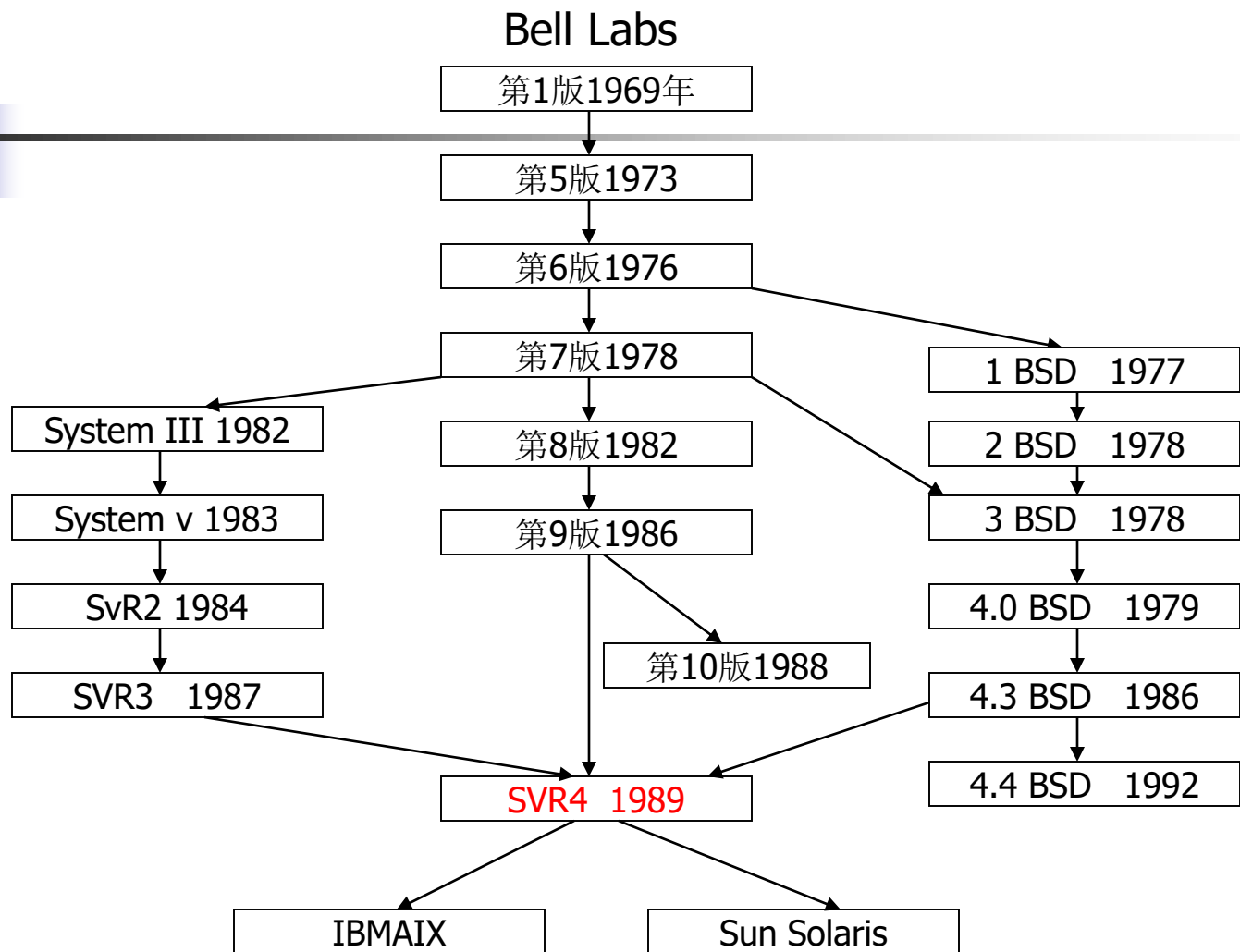
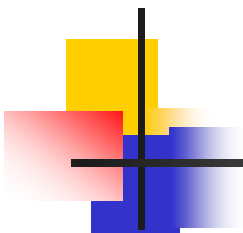




# UNIX系统的发展

---

- UNIX系统是美国贝尔实验室的两名程序员K. Thompson和D. M. Ritchie为PDP-7机器所设计和实现的一个分时操作系统。
- 最初采用汇编语言编写，后采用了C语言，并先后形成了第3、4、5、6、7版、UNIX System v2.0 (UNIX SVR 2)、UNIX SVR 3、UNIX SVR 4、UNIX SVR 4.2版本以及BSD UNIX版本系列。
- UNIX是目前世界上最成功、最流行的OS之一。
- UNIX操作系统发展历程图（下页）



## UNIX版本发展历史



## 1.7 自由软件和Linux操作系统(1)

- 商业软件    共享软件    自由软件
- 自由软件是指遵循通用公共许可证GPL  
(General Public License) 规则，保证使用上的自由、获得源程序的自由，可以自己修改的自由，可以复制和推广的自由，也可以有收费的自由的一种软件。
- 自由软件出现的意义 。



## 自由软件和Linux操作系统 (2)

---

### 自由软件的经典之作：

**C++编译器、Objective C、  
FORTRAN77、C 库、BSD email、  
BIND、Perl、Apache、TCP/IP、  
IP accounting、HTTPserver、  
Lynx Web、...**



# Linux操作系统(1)

---

- Linux是由芬兰籍科学家Linus Torvalds于1991年编写完成的操作系统内核。
- 许多人对Linux进行改进、扩充、完善，做出了关键性贡献——Linux由最初一个人写的原型变成在Internet上由无数志同道合的程序高手们参与的一场运动。





# Linux操作系统(2)

## Linux技术特点(1)

- (1) 继承了UNIX的优点，有许多改进，是集体智慧的结晶，能紧跟技术发展潮流，具有极强的生命力；
- (2) 通用操作系统，可作为Internet上的服务器；网关路由器；可用做文件和打印服务器；也可供个人使用；
- (3) 内置通信联网功能，能让异种机联网；



## Linux操作系统 (3)

### Linux技术特点 (2)

- (4) 开放源代码，有利于发展各种特色的操作系统；
- (5) 符合POSIX标准（可移植操作系统接口），各种UNIX应用可方便地移植到Linux下；
- (6) 提供庞大的管理功能和远程管理功能；
- (7) 支持大量外部设备；



# Linux操作系统(4)

## Linux技术特点(3)

(8) 支持32种文件系统, 如Ext2、Ext3、Xiafs、Isofs、Hpfs、MSDOS、UMSDOS、Proc、NFS、SYSV、Minix、SMB、Ufs、Ncp、VFAT、AFFS等;

(9) 提供GUI, 有图形接口X-Windows, 有多种窗口管理器;

(10) 支持并行处理和实时处理, 能充分发挥硬件性能;

(11) 可自由获得源代码, 在Linux平台上开发软件成本低



# Linux操作系统 (5)

## Linux技术特点 (4)

### (12) 多用户多任务系统

Linux的关键特点就是支持多用户多任务。多用户是指在同一时刻可以有多个用户同时登陆系统并完成各自的工作或操作，多任务是指在同一时刻用户可以有多个进程同时运行且完成各自的工作或操作。



## 本章要求：

---

**了解操作系统的基本概念、操作系统发展历史，操作系统的特征、操作系统应该具备的功能、OS结构设计，掌握多道批处理系统、实时系统和分时系统的工作原理，实时和分时系统的区别，UNIX操作系统基本结构。**