

# 操作系统

主讲教师：向勇

电话：6278.5609

Email：xyong@tsinghua.edu.cn

助教：张凯捷、杨敏

2001-02-19

1

## 参考书

- William Stallings, Operating Systems(3rd edition), 清华大学出版社, 1998年6月
- 张尧学, 史美林, 计算机操作系统教程, 清华大学出版社, 1993年9月

## 参考书(续)

- Uresh Vahalia, UNIX高级教程—系统技术内幕, 清华大学出版社, 1999年5月;
- David A. Solomon, Mark E. Russinovich, Inside Microsoft Windows 2000, 3rd Edition, Microsoft Press, 2000, ISBN 0-7356-1021-5
- David A. Solomon, Windows NT 技术内幕, 清华大学出版社, 1999年7月

## 课程主要内容介绍

操作系统的功能

- 管理系统软硬件资源、
- 扩展计算机的功能、
- 向用户提供服务。

- 用户接口
- 进程管理
- 处理机管理
- 存储管理
- 文件系统
- 外部设备管理

## 课程要求

- 作业：10%
- 上机实验：10%
- 考试或课程报告：70%
  - 第8周提交课程报告方案
  - 第16周完成
- Linux内核代码报告：10%
  - 要求：接口功能描述、数据结构分析、流程图、代码注释）

## 第一章 绪论

### 1.1 什么是操作系统

### 1.2 操作系统的发展历史

### 1.3 操作系统的分类

### 1.4 操作系统的特征

### 1.5 操作系统的功能

### 1.6 操作系统的结构

### 1.7 常用的操作系统

## 1.1 什么是操作系统

### 1.1.1 操作系统的地位和目标

### 1.1.2 操作系统的作用和组成

### 1.1.3 操作系统举例

## 1.1.1 操作系统的地位和目标

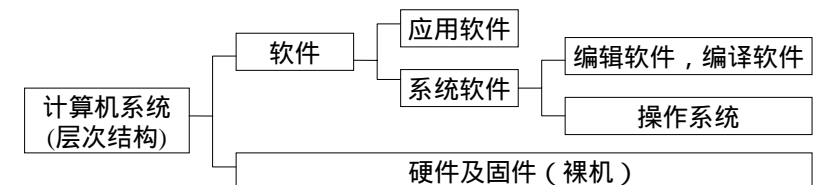
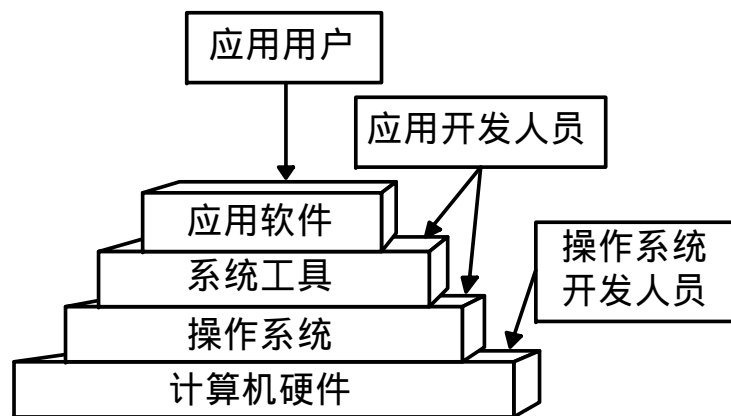


图1.1 计算机系统的组成

[返回](#)

## 操作系统在计算机系统中的地位



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)

## 引入操作系统的目标

- 有效性（系统管理人员的观点）：管理和分配硬件、软件资源，合理地组织计算机的工作流程
- 方便性（用户的观点）：提供良好的、一致的用户接口，弥补硬件系统的类型和数量差别
- 可扩充性（开放的观点）：硬件的类型和规模、操作系统本身的功能和管理策略、多个系统之间的资源共享和互操作

### 1.1.2 操作系统的作用

#### 操作系统的作用(1)

OS是计算机硬件、软件资源的管理者。

- 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）。

#### 操作系统的作用(2)

OS是用户使用系统硬件、软件的接口。

- 系统命令（命令行、菜单式、命令脚本式、图形用户接口GUI）；
- 系统调用（形式上类似于过程调用，在应用编程中使用）。

## 操作系统的作用(3)

OS是扩展机(extended machine)/虚拟机(virtual machine)。

- 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）；
- 另外，为合理组织工作流程：作业管理、进程管理。

操作系统的非形式化定义（关键点）：系统软件，程序模块的集合，资源管理和用户接口功能

## 1.1.3 操作系统举例

- MS OS: MS DOS, MS Windows 3.x, Windows 95, Windows NT, Windows 2000
- UNIX: BSD, SRV4, OSF1, SCO UNIX, AIX, Solaris, Linux
- NOS: Novell Netware
- RTOS: VxWorks, pSoS, Nucleus

## 1.2 操作系统的发展历史

### 1.2.1 推动操作系统发展的主要动力

### 1.2.2 手工操作

### 1.2.3 单道批处理系统(simple batch processing)

### 1.2.4 多道批处理系统(multiprogramming system)

### 1.2.5 分时系统(time-sharing system)

### 1.2.6 实时系统(real-time system)

[返回](#)

## 1.2.1 推动操作系统发展的主要动力

“需求推动发展”

- (1) 提高资源的利用率和系统性能：计算机发展的初期，计算机系统昂贵，用作集中计算
- (2) 方便用户：用户上机、调试程序，分散计算时的事务处理和非专业用户（商业和办公、家庭）
- (3) 器件的发展：CPU的位宽度（指令和数据）、快速外存

## 1.2.2 手工操作

1946 ~ 50年代（电子管），集中计算（计算中心），计算机资源昂贵；

- 工作方式
  - 用户：用户既是程序员，又是操作员；用户是计算机专业人员；
  - 编程语言：为机器语言；
  - 输入输出：纸带或卡片；
- 计算机的工作特点
  - 用户独占全机：不出现资源被其他用户占用，资源利用率低；
  - CPU等待用户：计算前，手工装入纸带或卡片；计算完成后，手工卸取纸带或卡片；CPU利用率低；

- 主要矛盾
  - 计算机处理能力的提高，手工操作的低效率（造成浪费）；
  - 用户独占全机的所有资源；
- 提高效率的途径
  - 专门的操作员，批处理

## 1.2.3 单道批处理系统

(simple batch processing, uniprogramming)

50年代末 ~ 60年代中（晶体管）：利用磁带把若干个作业分类编成作业执行序列，每个批作业由一个专门的监督程序（Monitor）自动依次处理。可使用汇编语言开发。

- 批处理中的作业的组成：
  - 用户程序
  - 数据
  - 作业说明书（作业控制语言）
- 批：
  - 供一次加载的磁带或磁盘，通常由若干个作业组装成，在处理中使用一组相同的系统软件（系统带）

## 两种批处理方式(1) 联机批处理

- 用户提交作业：以纸带或卡片为介质；
- 操作员合成批作业：结果为磁带介质；
- 批作业处理：对批作业中的每个作业进行相同的处理：从磁带读入用户作业和编译链接程序，编译链接用户作业，生成可执行程序；启动执行；执行结果输出。
- 这时的问题：慢速的输入输出处理仍直接由主机来完成。输入输出时，CPU处于等待状态。

## 两种批处理方式(2)

### 脱机批处理

利用卫星机完成输入输出功能。主机与卫星机可并行工作。

- 卫星机：完成面向用户的输入输出（纸带或卡片），中间结果暂存在磁带或磁盘上。
- 作业控制命令由监督程序(monitor)来执行，完成如装入程序、编译、运行等操作。
- 优点：同一批内各作业的自动依次更替，改善了主机CPU和I/O设备的使用效率，提高了吞吐量。
- 缺点：磁带或磁盘需要人工装卸，作业需要人工分类，监督程序易遭到用户程序的破坏（由人工干预才可恢复）。

## 通道和中断技术

60年代初，发展了通道技术和中断技术，这些技术的出现使监督程序在负责作业运行的同时提供I/O控制功能。

- 通道：用于控制I/O设备与内存间的数据传输。启动后可独立于CPU运行，实现CPU与I/O的并行。
  - 通道有专用的I/O处理器，可与CPU并行工作
  - 可实现 I/O联机处理
- 中断是指CPU在收到外部中断信号后，停止原来工作，转去处理该中断事件，完毕后回到原来断点继续工作。
  - 中断处理过程：中断请求，中断响应，中断点（暂停当前任务并保存现场），中断处理例程，中断返回（恢复中断点的现场并继续原有任务）
  - 可处理算术溢出和非法操作码，死循环（利用时钟中断进行超时限定）
- 监督程序发展为执行系统(executive system)，常驻内存

## 单道批处理的主要问题

- CPU和I/O设备使用忙闲不均（取决于当前作业的特性）。
  - 对计算为主的作业，外设空闲；
  - 对I/O为主的作业，CPU空闲；

## 1.2.4 多道批处理系统 (multiprogramming system)

60年代中～70年代中（集成电路），利用多道批处理提高资源的利用率。

- 多道批处理的运行特征
  - 多道：内存中同时存放几个作业；
  - 宏观上并行运行：都处于运行状态，但都未运行完；
  - 微观上串行运行：各作业交替使用CPU；

在当前运行的作业需作I/O处理时，CPU转而执行另一个作业。（I/O完成后是否立刻恢复执行？要等到其他程序再次I/O时）

## 多道批处理系统的特点

- 优点：
  - 资源利用率高：CPU和内存利用率较高；
  - 作业吞吐量大：单位时间内完成的工作总量大；
- 缺点：
  - 用户交互性差：整个作业完成后或中间出错时，才与用户交互，不利于调试和修改；
  - 作业平均周转时间长：短作业的周转时间显著增长；

批处理：交互性差 - - 提高对CPU利用率；

分时处理：用户与应用程序随时交互，控制程序运行，适于商业和办公事务处理 - - 缩短响应时间

## 1.2.5 分时系统 (time-sharing system)

70年代中期至今

- “分时”的含义分时是指多个用户分享使用同一台计算机。多个程序分时共享硬件和软件资源。
  - 多个用户分时：单个用户使用计算机的效率低，因而允许多个应用程序同时在内存中，分别服务于不同的用户。有用户输入时由CPU执行，处理完一次用户输入后程序暂停，等待下一次用户输入 - - 时走时停
  - 前台和后台程序(foreground & background)分时：后台程序不占用终端输入输出，不与用户交互 - - 现在的图形用户界面(GUI)，除当前交互的程序(输入焦点)之外，其他程序均作为后台
  - 通常按时间片(time slice)分配：各个程序在CPU上执行的轮换时间。

## 抢先式和非抢先式

- 抢先式和非抢先式(preemptive & non-preemptive)：出让CPU是OS强迫或程序主动
  - 抢先式：OS强迫出让CPU；
  - 非抢先式：程序主动出让CPU；

## 分时系统的特点

- 人机交互性好：在调试和运行程序时由用户自己操作。
- 共享主机：多个用户同时使用。
- 用户独立性：对每个用户而言好象独占主机。

现在的许多操作系统都具有分时处理的功能，在分时系统的基础上，操作系统的发展开始分化，如实时系统、通用系统、个人系统等。

## 1.2.6 实时系统(real-time system)

用于工业过程控制、军事实时控制、金融等领域，包括实时控制、实时信息处理

- 要求：响应时间短，在一定范围之内；系统可靠性高
- 任务的类型：
  - 周期性实时任务：
  - 非周期性实时任务：截止时间(deadline)，开始截止时间（最晚开始时间）和完成截止时间（最晚完成时间）

目前的操作系统，通常具有分时、实时和批处理功能，又称作通用操作系统。可适用于计算、事务处理等多种领域，能运行在多种硬件平台上，如 UNIX系统、Windows NT等。 - - 通用化、小型化

## 1.3 操作系统的分类

操作系统分类主要讨论操作系统的内部特征。

### 1.3.1 批处理操作系统

### 1.3.2 分时操作系统

### 1.3.3 实时操作系统

### 1.3.4 多处理操作系统

### 1.3.5 网络操作系统

### 1.3.6 分布式操作系统

### 1.3.7 个人计算机操作系统

[返回](#)

## 1.3.1 批处理操作系统 (Batch Processing Operation System)

- 作业的处理流程
  - 作业提交：作业的输出；
  - 作业执行
  - 作业完成：作业的输出；

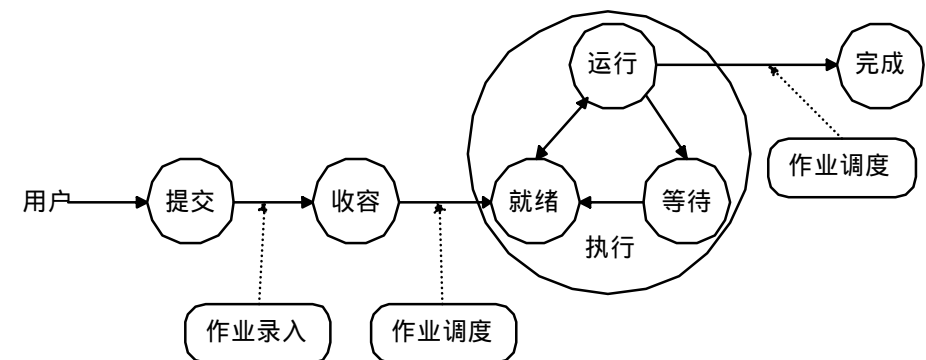


图1 批处理系统中作业处理及状态



## 单道(uniprogramming)和多道批处理的比较

|      | 单道      | 多道             |
|------|---------|----------------|
| 内存使用 | 每次一个作业  | 每次多个作业（充分利用内存） |
| 作业次序 | 顺序，先进先出 | 无确定次序          |

多道程序系统和多处理系统(multiprocessing system)的区别：前者指多个程序同时在内存中交替运行，后者指多个处理器。

## 批处理的主要特征

- 用户脱机使用计算机：作业提交后直到获得结果之前，用户无法与作业交互。
- 作业成批处理
- 多道程序并行：充分利用系统资源。

## 多道批处理系统的资源利用效率特征

多道批处理系统的资源利用效率特征是基于各作业对系统资源的需求差异得到的。

例如：有3个作业A、B、C，分别为计算、检索和打印作业，单道运行时间分别为5分、15分和10分钟。它们可并行在15分钟内完成3个作业。各资源的利用效率为：

|    | CPU | 打印机 | 磁带   |
|----|-----|-----|------|
| 单道 | 17% | 33% | 50%  |
| 多道 | 33% | 66% | 100% |

## 多道批处理系统上的技术

- 作业调度：作业的现场保存和恢复 - - 上下文切换
- 资源共享：资源的竞争和同步 - - 互斥(exclusion)和同步(synchronization)机制
- 内存使用：提高内存使用效率（为当前由CPU执行的程序提供足够的内存） - - 覆盖(overlay)，交换(swap)和虚拟存储(virtual memory)
- 内存保护：系统存储区和各应用程序存储区不可冲突 - - 存储保护
- 文件非顺序存放、随机存取

出现：作业管理、处理机管理、存储管理、设备管理、文件系统管理(file system)

## 1.3.2 分时操作系统 (Time Sharing Operating System)

### 分时的定义

把计算机的系统资源（尤其是CPU时间）进行时间上的分割，每个时间段称为一个时间片（time slice），每个用户依次轮流使用时间片。

### 分时系统的特征

- 多路性：多个用户同时工作。
  - 共享系统资源，提高了资源利用率。节省维护开支，可靠性高：笨终端 - 至今仍在使⤵用。促进了计算机的普遍应用，提高资源利用率：远地用户通过终端（较便宜）联机使用。
- 独立性：各用户独立操作，互不干扰。
- 交互性：系统能及时对用户的操作进行响应，显著提高调试和修改程序的效率：缩短了周转时间。

### 分时系统的类型

- 单道分时：调入-调出(Roll-in/Roll-out)，I/O开销太大（在有卫星机处理I/O时，单道分时是有用的）
- 前台后台分时：后台存放批处理作业，内存的划分是固定的，不灵活
- 多道分时：需要解决加载程序时地址空间重定位的问题

### 分时系统的主要问题

- 及时接受输入：多个I/O端口，设立多路缓冲区
- 及时响应：
  - 提高对换速度（快速外存）、
  - 限制用户数目、
  - 缩短时间片（可能引起对换次数增多，开销增大，程序总运行时间增大）
- 减少对换信息量：
  - 可重入代码(re-entrant code)；
  - 请求页式存储管理：只对换部分程序

### 1.3.3 实时操作系统 (Real Time Operating System)

实时操作系统主要用于过程控制、事务处理等有实时要求的领域，其主要特征是实时性和可靠性。

- 实时系统的特征
  - 实时时钟管理：提供系统日期和时间、定时和延时等时钟管理功能；
  - 过载保护：缓冲区排队，丢弃某些任务，动态调整任务周期；
    - 过载是指进入系统的任务数目超出系统的处理能力。
  - 高度可靠性和安全性：容错能力（如故障自动复位）和冗余备份（双机，关键部件）；

### 实时系统与批处理系统和分时系统的区别

- 专用系统：许多实时系统是专用系统，而批处理与分时系统通常是通用系统。
- 实时控制：实时系统用于控制实时过程，要求对外部事件的迅速响应，具有较强的中断处理机构。
- 高可靠性：实时系统用于控制重要过程，要求高度可靠，具有较高冗余。如双机系统。
- 事件驱动和队列驱动：实时系统的工作方式：接受外部消息，分析消息，调用相应处理程序进行处理。
- 可与通用系统结合成通用实时系统：实时处理前台作业，批处理为后台作业。

### 1.3.4 多处理操作系统 (Multi-processor Operating System)

多处理操作系统的出现是为了提高计算机系统性能和可靠性。提高性能有两条途径：提高各个组成部分的速度、增大处理的并行程度。1975年前后，出现多处理机系统(multi-processor)。

- 多处理机系统的特点
  - 增加系统的吞吐量：N个处理器加速比达不到N倍（额外的调度开销，算法的并行化）
  - 提高系统可靠性：故障时系统降级运行

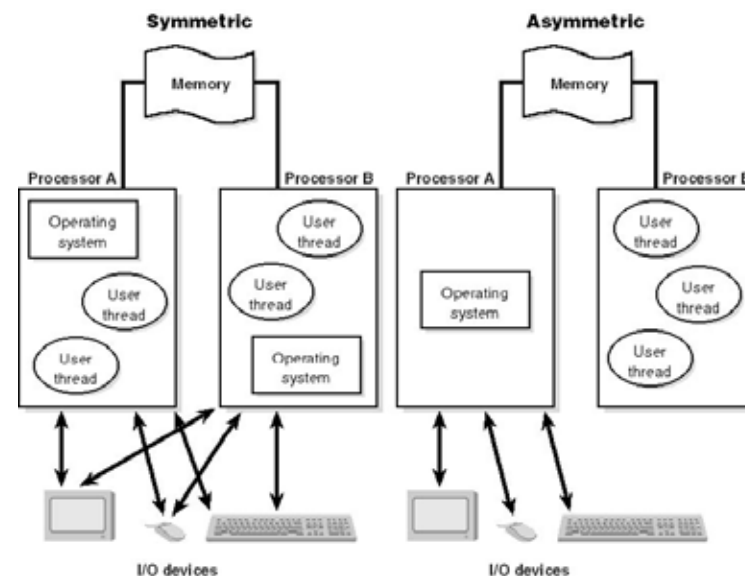
### 多处理机系统的类型

- 紧密耦合(tightly-coupled)：各处理机之间通过快速总线或开关阵列相连，共享内存，整体系统由一个统一的OS管理（一个OS核心）。
- 松散耦合(loosely-coupled)：各处理机带有各自的存储器、I/O设备和操作系统，通过通道或通信线路相连。每个处理机上独立运行OS。

## 多处理操作系统的类型

要运行OS本身、I/O处理（如中断响应）、应用程序

- 非对称式多处理(Asymmetric Multiprocessing, ASMP)：又称主从模式(Master-slave mode)。
  - 主处理器：只有一个，运行OS。管理整个系统的资源，为从处理器分配任务；
  - 从处理器：可有多个，执行应用程序或I/O处理。
  - 特点：不同性质任务的负载不均，可靠性不够高，不易移植（通常要求硬件也是"非对称"）。
- 对称式多处理(Symmetric Multiprocessing, SMP)：OS交替在各个处理器上执行。任务负载较为平均，性能调节容易 - - "傻瓜式"



对称多处理机与非对称多处理机的比较

### 1.3.5 网络操作系统 (NOS, Network Operating System)

网络操作系统是在通常操作系统功能的基础上提供**网络通信和网络服务功能**的操作系统。网络操作系统为网上计算机进行方便而有效的**网络资源共享**，提供网络用户所需各种服务的软件和相关规程的集合。

**网络功能与操作系统的结合程度**是网络操作系统的重要性能指标。早期的作法是通常操作系统**附加网络软件**，过渡到网络功能成为操作系统的**有机组成部分**。它们的区别在于：网络功能的强弱、使用是否方便等。

## 计算机网络

一些自主的计算机系统，通过通信设施相互连接，完成信息交换、资源共享、互操作和协同工作等功能。

引入计算机网络的目的：完成新的应用（进行自动的信息交换），提高性能 - 价格比（共享昂贵资源）

## 计算机网络的分类

- 按网络拓扑（传输媒体和网络设备所连接的几何形状）：星形、环形、总线形、不规则形等；
- 按地理范围：局域网（LAN, Local Area Network）、城域网(MAN, Metropolitan Area Network)、广域网(WAN, Wide Area Network)；
- 按网络标准 - - 内在特性，决定其性能（如效率随负载的变化、实时性、优先级）Ethernet, Token Ring, FDDI, X.25, ATM

## 网络操作系统的功能

- 通常操作系统的功能：处理机管理、存储器管理、设备管理、文件管理等；
- 网络通信功能：通过网络协议进行高效、可靠的数据传输；
- 网络资源管理：协调各用户使用；
- 网络服务：文件和设备共享，信息发布；
- 网络管理：安全管理、故障管理、性能管理等；
- 互操作：直接控制对方比交换数据更为困难；

## 1.3.6 分布式操作系统 (Distributed Operating System)

分布式系统：处理和控制的分散（相对于集中式系统）

分布式系统是以计算机网络为基础的，它的基本特征是处理上的分布，即功能和任务的分布。

分布式操作系统的所有系统任务可在系统中任何处理机上运行，自动实现全系统范围内的任务分配并自动调度各处理机的工作负载。

## 分布式操作系统与网络操作系统的比较

- 耦合程度：
  - 分布式系统是紧密耦合系统：分布式OS是在各机上统一建立的"OS同质"，直接管理CPU、存储器和外设；统一进行全系统的管理；
  - 网络通常容许异种OS互连，各机上各种服务程序需按不同网络协议"协议同质"。
- 并行性：
  - 分布式OS可以将一个进程分散在各机上并行执行"进程迁移"；
  - 网络则各机上的进程独立。
- 透明性：用户是否知道或指定资源在哪个机器上（如CPU、内存或外设）。
  - 分布式系统的网络资源调度对用户透明，用户不了解所占有资源的位置；
  - 网络操作系统中对网络资源的使用要由用户明确指定；
- 健壮性：分布式系统要求更强的容错能力（工作时系统重构）

### 1.3.7 个人计算机操作系统 (Personal Computer Operating System)

针对单用户使用的个人计算机进行优化的操作系统。

- 个人计算机操作系统的特征
  - 应用领域：事务处理、个人娱乐，
  - 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。
- 常用的个人计算机操作系统
  - 单用户单任务：MS DOS
  - 单用户多任务：OS/2, MS Windows 3.x, Windows 95, Windows NT, Windows 2000 Professional
  - 多用户多任务：UNIX(SCO UNIX, Solaris x86, Linux, FreeBSD)

## 1.4 操作系统的特征

### 1.4.1 操作系统的特征

### 1.4.2 操作系统的服务

[返回](#)

### 1.4.1 操作系统的特征

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

## 并发(concurrency)

多个事件在同一时间段内发生。操作系统是一个并发系统，各进程间的并发，系统与应用间的并发。操作系统要完成这些并发过程的管理。并行(parallel)是指在同一时刻发生。

- 在多道程序处理时，宏观上并发，微观上交替执行（在单处理器情况下）。
- 程序的静态实体是可执行文件，而动态实体是进程（或称作任务），并发指的是进程。

## 共享(sharing)

多个进程共享有限的计算机系统资源。操作系统要对系统资源进行合理分配和使用。资源在一个时间段内交替被多个进程所用。

- 互斥共享（如音频设备）：资源分配后到释放前，不能被其他进程所用。
- 同时访问（如可重入代码，磁盘文件）
- 资源分配难以达到最优化

## 虚拟(virtual)

一个物理实体映射为若干个对应的逻辑实体 - - 分时或分空间。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。

- CPU - - 每个用户（进程）的"虚处理机"
- 存储器 - - 每个进程都占有的地址空间（指令 + 数据 + 堆栈）
- 显示设备 - - 多窗口或虚拟终端(virtual terminal)

## 异步性(asynchronism)

也称不确定性，指进程的执行顺序和执行时间的不确定性；

- 进程的运行速度不可预知：分时系统中，多个进程并发执行，"时走时停"，不可预知每个进程的运行推进快慢
- 判据：无论快慢，应该结果相同 - - 通过进程互斥和同步手段来保证
- 难以重现系统在某个时刻的状态（包括重现运行中的错误）
- 性能保证：实时系统与分时系统相似，但通过资源预留以保证性能

## 1.4.2 操作系统的服务

- 服务类型
  - 程序执行和终止（包括分配和回收资源）
  - I/O操作
  - 文件系统操作
  - 通信：本机内，计算机之间（通常通信服务的使用者为进程，而不是笼统说"主机"）
  - 配置管理：硬件、OS本身、其他软件
  - 差错检测
- 服务提供方式：系统命令和系统调用

## 1.5 操作系统的功能

### 1.5.1 处理机管理

### 1.5.2 存储管理

### 1.5.3 设备管理

### 1.5.4 信息管理

### 1.5.5 用户接口

[返回](#)

## 1.5.1 处理机管理

完成处理机资源的分配调度等功能。处理机调度的单位可为进程或线程。

- 进程控制：创建、撤销、挂起、改变运行优先级等 - - 主动改变进程的状态
- 进程同步：协调并发进程之间的推进步骤，以协调资源共享； - - 交换信息能力弱
- 进程通信：进程之间传送数据，以协调进程间的协作； - - 交换信息能力强，也可以用来协调进程之间的推进
- 进程调度：作业和进程的运行切换，以充分利用处理机资源和提高系统性能； - - 未必是进程控制操作所引起（可能是时间片轮转、I/O操作）
  - 同一类型内的公平性、高效率（吞吐量大）、作业周转时间等

## 1.5.2 存储管理

管理目标：提高利用率、方便用户使用、提供足够的存储空间、方便进程并发运行。

- 存储分配与回收
- 存储保护：保证进程间互不干扰、相互保密；如：访问合法性检查、甚至要防止从"垃圾"中窃取其他进程的信息；
- 地址映射（变换）：进程逻辑地址到内存物理地址的映射；
- 内存扩充（覆盖、交换和虚拟存储）：提高内存利用率、扩大进程的内存空间；

## 1.5.3 设备管理

设备管理的目标是：方便的设备使用、提高CPU与I/O设备利用率；

- 设备操作：利用设备驱动程序（通常在内核中）完成对设备的操作。还需处理外设的IRQ。
- 设备独立性(device independence)：提供统一的I/O设备接口，使应用程序独立于物理设备，提高可适应性；在同样的接口和操作下完成不同的内容（如FAX Modem作为Windows上的打印机设备）。
- 设备分配与回收：在多用户间共享I/O设备资源。
  - 虚拟设备(virtual device)：设备由多个进程共享，每个进程如同独占。
- 缓冲区管理：匹配CPU和外设的速度，提高两者的利用率（单缓冲区、双缓冲区和公用缓冲区）



## 1.5.4 信息管理

解决软件资源的存储、共享、保密和保护。

- 文件存储空间管理：解决如何存放信息，以提高空间利用率和读写性能。
- 目录管理：解决信息检索问题。文件的属性（如文件名）、单一副本赋予多文件名
- 文件的读写管理和存取控制：解决信息安全问题。系统设口令"哪个用户"、用户分类"哪个用户组"、文件权限"针对用户或用户组的读写权"
- 软件管理：软件的版本、相互依赖关系、安装和拆除等

## 1.5.5 用户接口

目标：提供一个友好的用户访问操作系统的接口。操作系统向上提供两种接口：

- 系统命令：供用户用于组织和控制自己的作业运行。命令行、菜单式或GUI"联机"；命令脚本"脱机"
- 编程接口：供用户程序和系统程序调用操作系统功能。系统调用和高级语言库函数；

## 1.6 操作系统的结构

随着操作系统的发展，功能越强，OS自身代码量越大 - - 采用良好的结构：有利于保证正确性以及自身修改和扩充。

### 1.6.1 整体或模块结构

### 1.6.2 分层结构或虚拟机

### 1.6.3 客户/服务器模型或微内核结构

[返回](#)

## 操作系统的设计原则

- 可维护性：容易修改与否称为可维护性；有三种可能的维护：
  - 纠错性维护：改正已发现的错误；
  - 适应性维护：修改软件，使之适应新的运行环境（硬件环境和软件环境）；如：操作系统的移植。
  - 完善性维护：增加新功能；
- 可靠性：可靠性包括两方面：
  - 正确性：正确实现所要求的功能和性能；
  - 稳健性：对意外（故障和误操作）作出适当的处理；
- 可理解性：易于理解，以方便测试、维护和交流；
- 性能：有效地使用系统资源；尽可能快地响应用户请求；

### 1.6.1 整体或模块结构

monolithic system or modular system

整个系统按功能进行设计和模块划分。系统是一个单一的、庞大的软件系统。这种结构思想来源于服务功能观点，而不是资源管理的观点。

- 模块结构的特点：模块由众多服务过程（模块接口）组成，可以随意调用其他模块中的服务过程
  - 优点：具有一定灵活性，在运行中的高效率
  - 缺点：功能划分和模块接口难保正确和合理；模块之间的依赖关系（功能调用关系）复杂（调用深度和方向），降低了模块之间的相对独立性 - - 不利于修改

### 1.6.2 分层结构或虚拟机

layered system or virtual machine

从资源管理观点出发，划分层次。在某一层次上代码只能调用低层次上的代码，使模块间的调用变为有序性。系统每加一层，就构成一个比原来功能更强的虚拟机。有利于系统的维护性和可靠性。

### 分层结构的特点

- 优点：
  - 功能明确，调用关系清晰（高层对低层单向依赖），有利于保证设计和实现的正确性
  - 低层和高层可分别实现（便于扩充）；高层错误不会影响到低层；避免递归调用
- 缺点：降低了运行效率

各系统对具体划分多少层次有不同的看法。

### 分层原则

- 被调用功能在低层：如文件系统管理 - - 设备管理 - - 设备驱动程序
- 活跃功能在低层：提高运行效率
- 资源管理的公用模块放在最低层：如缓冲区队列、堆栈操作
- 存储器管理放在次低层：便于利用虚拟存储功能
- 最低层的硬件抽象层：与机器特点紧密相关的软件放在最低层。如Windows NT中的HAL - - 单处理、多处理
- 资源分配策略放在最外层，便于修改或适应不同环境

调用跨越的层次：相邻层（最严格）、所有下层、部分下层

### 1.6.3 客户/服务器模型或微内核结构 client-server model or microkernel

把操作系统分成若干分别完成一组特定功能的服务进程，等待客户提出请求；而系统内核只实现操作系统的基本功能(如：虚拟存储、消息传递)。

- 微内核(micro-kernel)：将更多操作系统功能放在核心之外，作为独立的服务进程运行；
  - 服务进程（或称作“保护子系统”）
  - 客户进程（系统客户和应用客户） - - 需支持多进程
- 本地过程调用 (LPC, Local Procedure Call)：一种进程之间请求-应答式的消息（Message）传递机制。
- 消息：是一定格式的数据结构。发起调用，送出请求消息  
请求消息到达并进行处理 送出回答消息 整理回答消息，返回结果；如：对文件create, read, write

## 微内核模式的特点

- 优点：
  - 良好的扩充性：只需添加支持新功能的服务进程即可
  - 可靠性好：调用关系明确，执行转移不易混乱
  - 便于网络服务，实现分布式处理：以同样的调用形式，在下层可通过核心中的网络传送到远方服务器上（远地过程调用 RPC, Remote Procedure Call)
- 缺点：
  - 消息传递比直接调用效率要低一些（但可以通过提高硬件性能来补偿）

RPC的过程：RPC应用程序 - - RPC Stub(client) - -  
Network - - RPC Server - - 进行本地调用

## 1.7 常用的操作系统

### 1.7.1 MS DOS

### 1.7.2 MS Windows 3.x, Windows 95, Windows NT, Windows 2000

### 1.7.3 UNIX

[返回](#)

## 1.7.1 MS DOS

IBM PC, CPU 8088/8086, BIOS  
单用户单任务，简单分层结构，16位

## MS DOS的历史

- 1981年：PC-DOS 1.1：IBM PC，只支持软盘的个人操作系统；
- 1983年：DOS2.0：PC XT，支持硬盘和目录的层次结构，并提供丰富的系统命令；
- 1984年：DOS3.0：PC AT (Intel 80286 CPU)，它把286作为一个快速的8086使用；
- 1987年：DOS3.3：提供对IBM PS/2的支持（如3.5"软驱），提供了更多的应用；
- 1988年：DOS4.0：支持大于32M的硬盘；
- 1991年：DOS5.0：改进对扩展内存的支持；

## MS DOS的结构



- DOS BIOS(Basic Input/Output System)：由一组与硬件相关的设备驱动程序组成，实现基本的输入/输出功能；
- DOS核心：提供一套独立于硬件的系统功能：内存管理、文件管理、字符设备和输入/输出、实时时钟等；
- 命令处理程序：对用户命令进行分析和执行；

## MS DOS的特点

- 字符用户界面。作业管理：命令行，批处理程序（BAT文件），菜单式。编程时通过软中断调用(int 21h)来使用系统功能。不区分用户。
- "准多任务"：通过内存驻留程序TSR(Terminated and Stay Resident)来实现，通过时钟中断或键盘中断"热键hotkey"来激活其他任务。
- 不支持虚拟存储，没有存储保护。采用段式分配(内存块)，可直接访问的最大地址空间为1MB。其余的内存只能通过作为扩展内存(XMS)或扩充内存(EMS)来使用。
  - XMS是段式分配，通过内存数据搬移来使用XMS区域
  - EMS是页式分配，通过页面的映射来使用EMS区域
  - 或者用支持保护方式的编程工具
- 文件系统为FAT(File Allocation Table)格式（磁盘卷，多级目录，文件名8+3个字符；分区容量最大为2GB）；有文件属性，没有区分用户的访问权限保护。
- 设备驱动程序在系统起动时加载。分为字符设备和块设备。

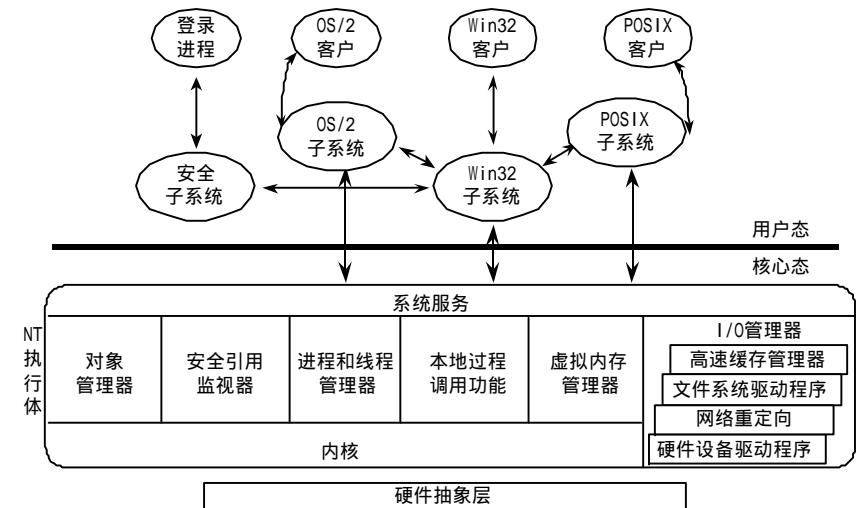
## 1.7.2 MS Windows 3.x, Windows 95, Windows NT, Windows 2000

CPU 80386

单用户多任务（分时系统），16位/16和32位混合/32位

## Windows的历史

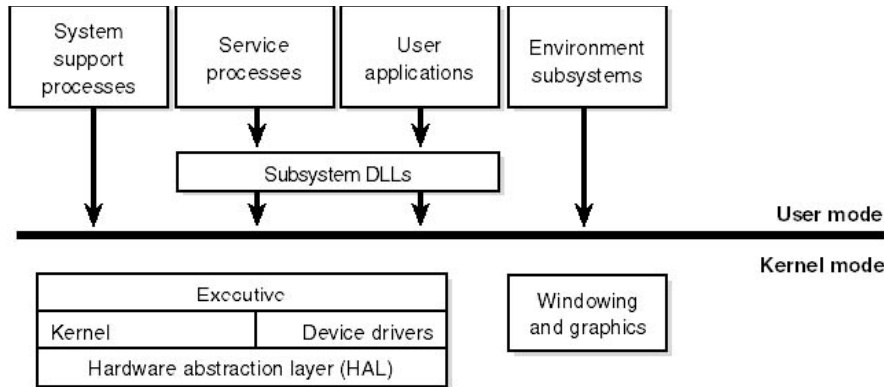
- 1990年：Windows 3.0（成功版本），16位OS，借见Apple Macintosh给出友好的用户界面；
- 1993年：Windows NT 3.1, 32位OS，支持DOS和Windows应用程序；
- 1999年12月：Windows 2000(Professional, Server, Advanced Server)，32位OS；



Windows NT体系结构

# Windows 2000的特点

- 支持对称多处理机
- 真正的32位操作系统：除16位应用的支持代码，没有16位的代码；
- 完全的代码可重入(reentrant)：同一段代码可由多个应用同时访问；
- 图形用户界面GUI（和字符用户界面）。
- 抢先式多任务和多线程。支持动态链接。
- 虚拟存储：段页式(有存储保护)。
- 兼容16位Windows应用：
- 文件系统：NTFS(HPFS)，支持安全控制
- 设备驱动程序：VxD(virtual driver)。
- 可移植：适用于多种硬件平台。
- 容错能力。
- 面向对象特性，用对象来表示所有资源。



简化的Windows2000体系结构

## 1.7.3 UNIX

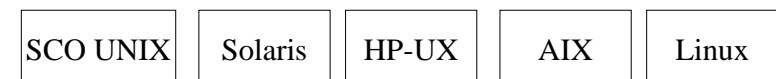
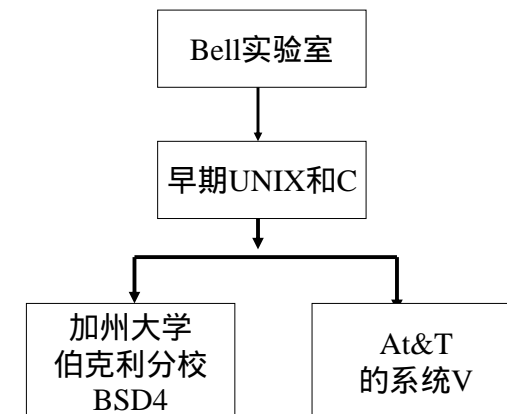
多用户多任务，16/32/64位

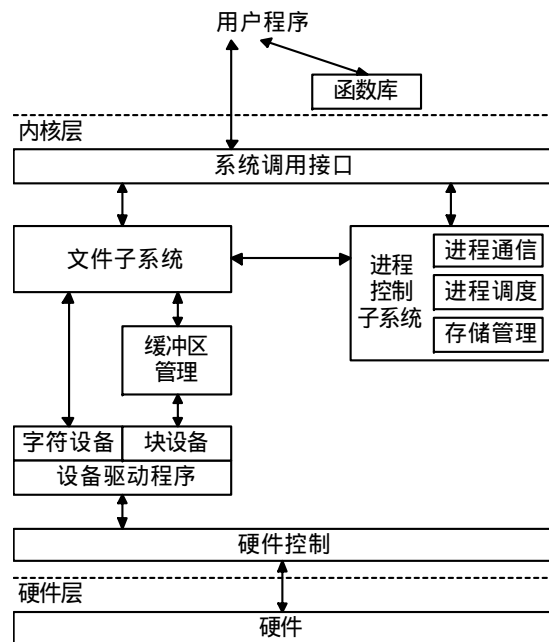
BSD, SVR4（模块式结构），OSF/1（微内核结构）

## UNIX的历史

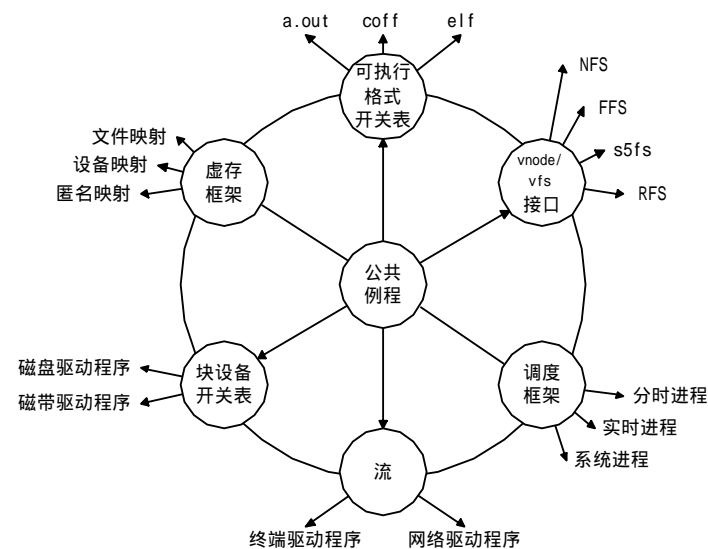
- 1965年：MIT的Multics，由于规模和进展而没有达到目标；
- 1969年：AT&T，PDP-11上的16位操作系统；
- 1974年：UNIX系统正式发表(第五版)，在大学得到使用和好评；
- 1980年：University of California at Berkeley为VAX11发表BSD4.0；以后，UNIX就以AT&T和Berkeley为主分别开发，有多种变种；
- 1989年：UI (UNIX International)发表UNIX system V Res4.0；使BSD和System V在用户界面上统一；
- 1991年芬兰大学生Linus Benedict Torvalds开发了第一个Linux版本。
- 1994年，Linux 1.0，现在的最新内核版本是2.4

UNIX系统：可运行UNIX应用程序的操作系统。





传统的UNIX结构



现代UNIX结构

## UNIX系统的特点

- 字符用户界面和图形用户界面GUI(X Window)。
- 抢先式多任务，多线程。支持动态链接。支持对称式多处理。
- 虚拟存储：段页式，有存储保护。
- 文件系统：多级目录，文件卷可以在子目录下动态装卸。无文件属性，可有别名。
- 采用设备文件的形式（读写，参数控制）。设备驱动程序修改后需要重新编译连接生成内核。
- 支持多种硬件平台。
- 易移植：主要代码用C语言写成；
- 变种很多，很难标准化。

## 小结

- OS地位、目的、作用和组成
- OS发展：主要动力
- OS分类：批处理、分时、实时、（通用）、多处理、网络和分布式、PC
- OS的结构：模块 - - 层次 - - Client-Server
- OS的特征和服务
- OS功能

## 作业（任选一题）

- 分析MS DOS操作系统的演变过程及其与计算机硬件技术和操作系统技术的关系；
- 分析MS Windows操作系统的演变过程及其与计算机硬件技术和操作系统技术的关系；
- 分析几种主流UNIX操作系统的演变过程及其与计算机硬件技术和操作系统技术的关系；

每周的作业在下周的第二次上课时提交，过期不计成绩。每个班指定一个联系人，为大家服务。