

# 操作系统A

## Principles of Operating System

北京大学计算机科学技术系 陈向群

Department of computer science  
and Technology, Peking University

2020 Autumn

# 本章要求掌握的概念

---

操作系统定义

并发性

共享性

虚拟性

随机性

SPOOLing

多道程序设计

OS/360

MULTICS

操作系统结构

.....

# 大纲

---

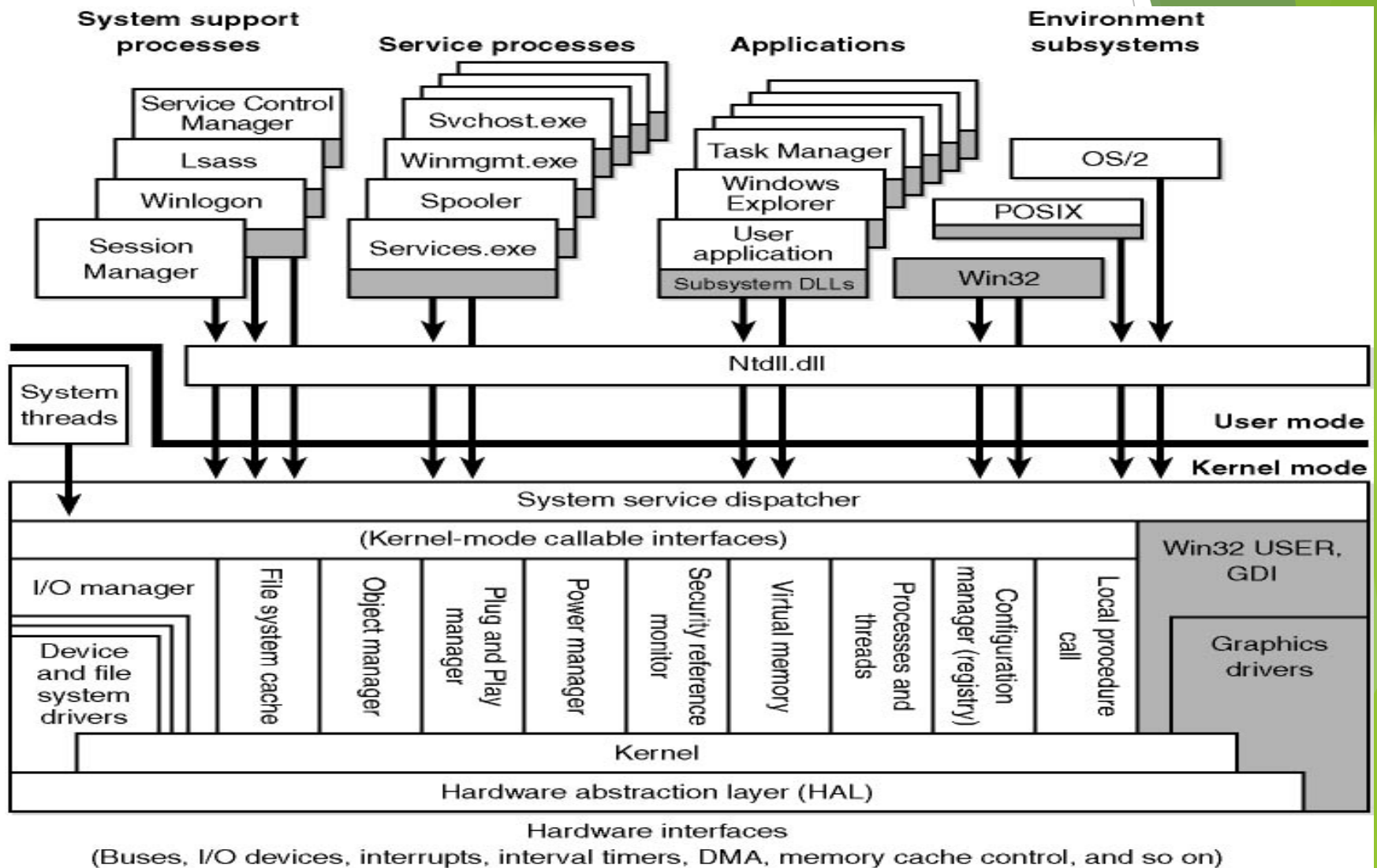
- ▶ 典型的操作系统架构
- ▶ 操作系统是什么?
  - ▶ 定义（描述）、解读操作系统定义
- ▶ 各种角度认知操作系统
- ▶ 操作系统的特征
- ▶ 操作系统的发展历程及启示
- ▶ 操作系统的分类
- ▶ 操作系统结构

*Windows、Linux、Android、.....*

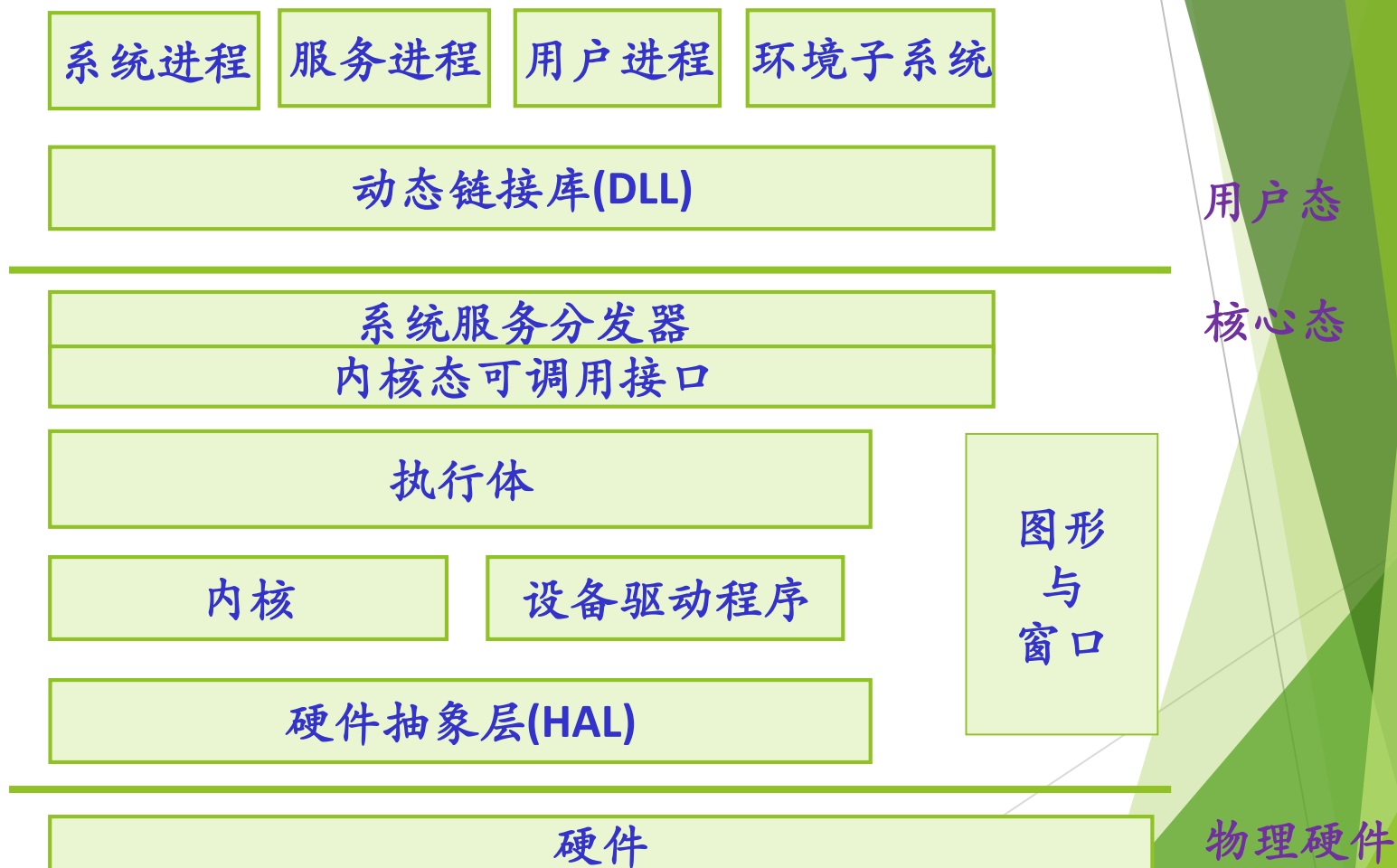
# *典型的操作系统*

# Windows 架构

# Windows操作系统的体系结构

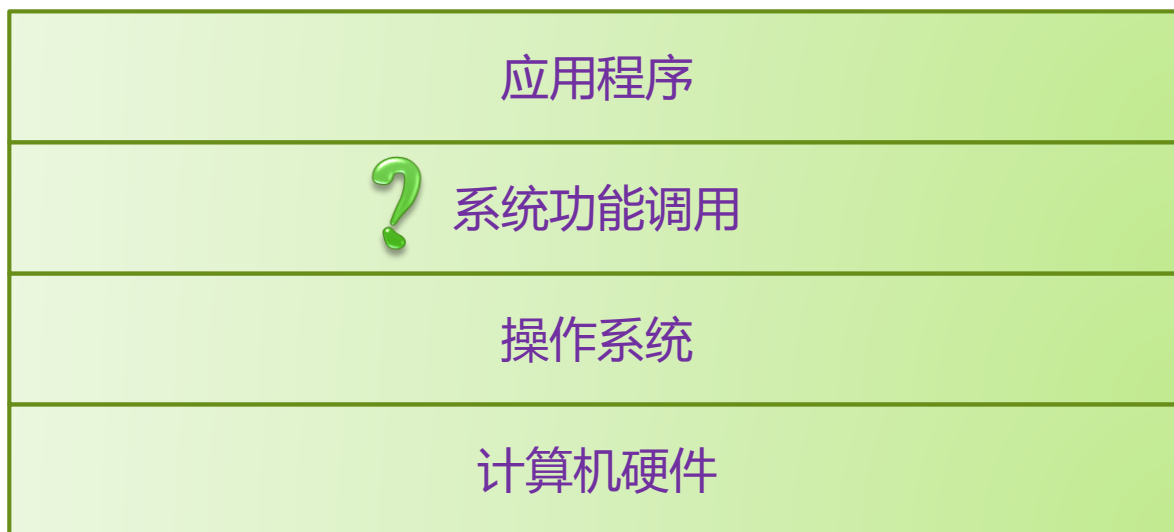


# Windows 操作系统的体系结构



# Windows 操作系统的体系结构

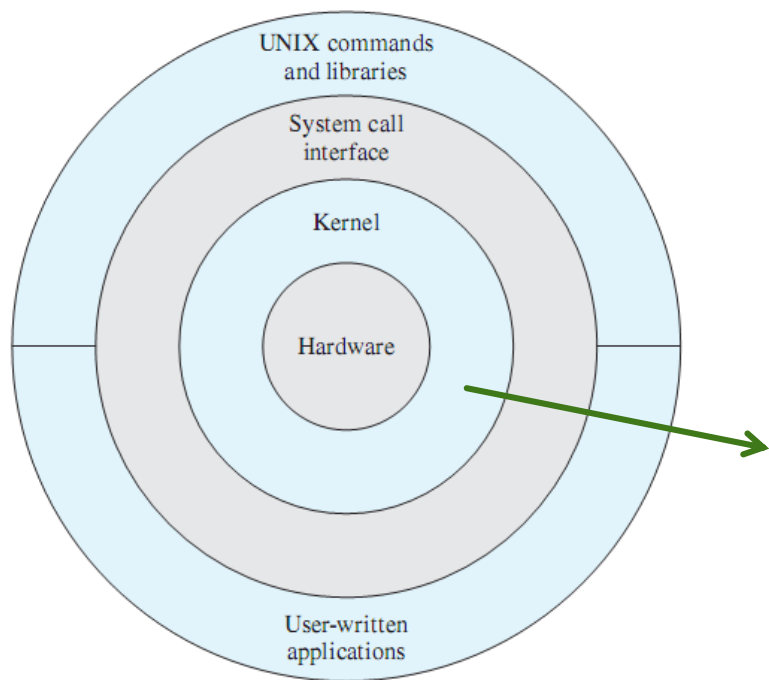
---





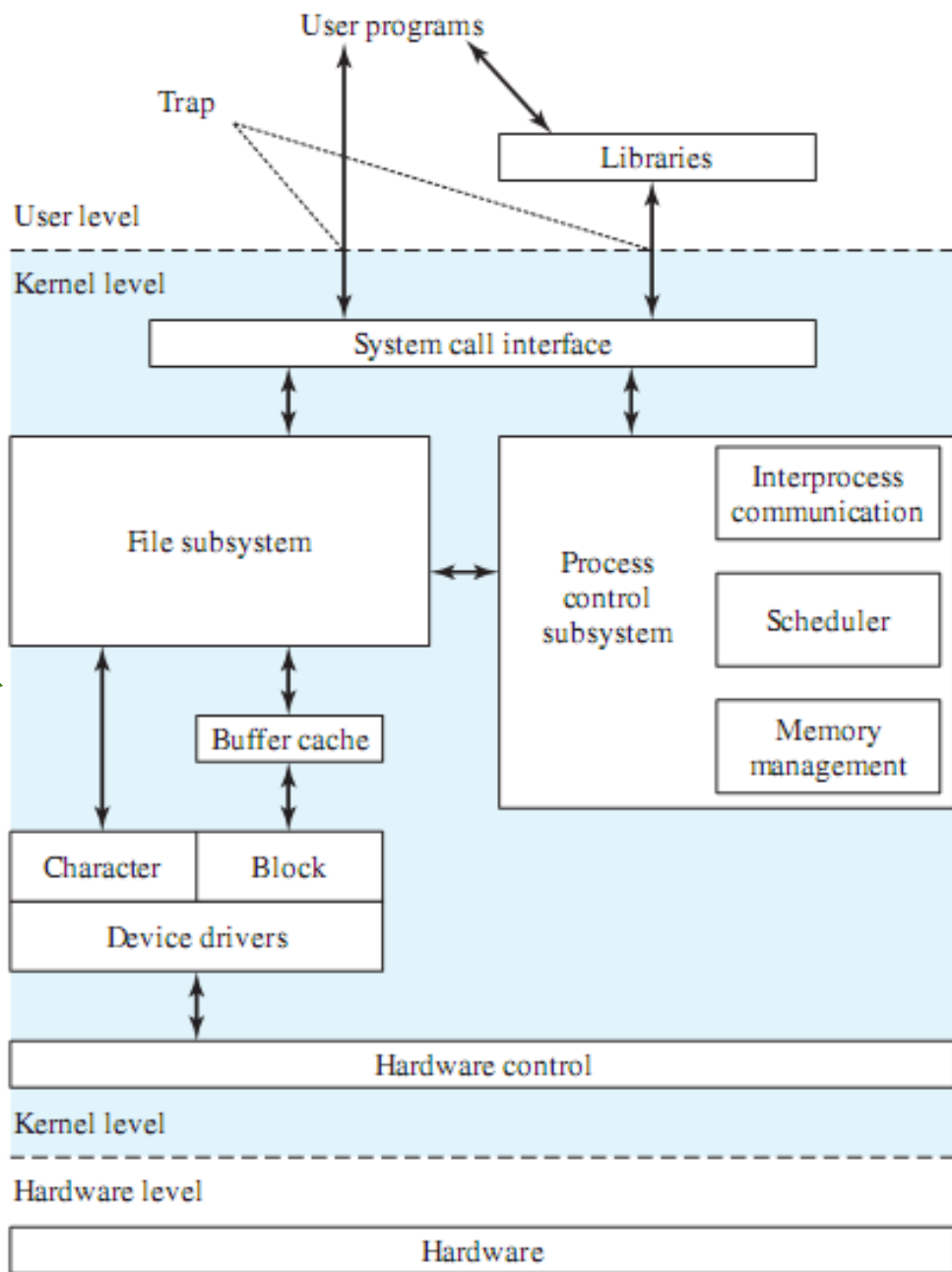
# UNIX 架构

# UNIX操作系统



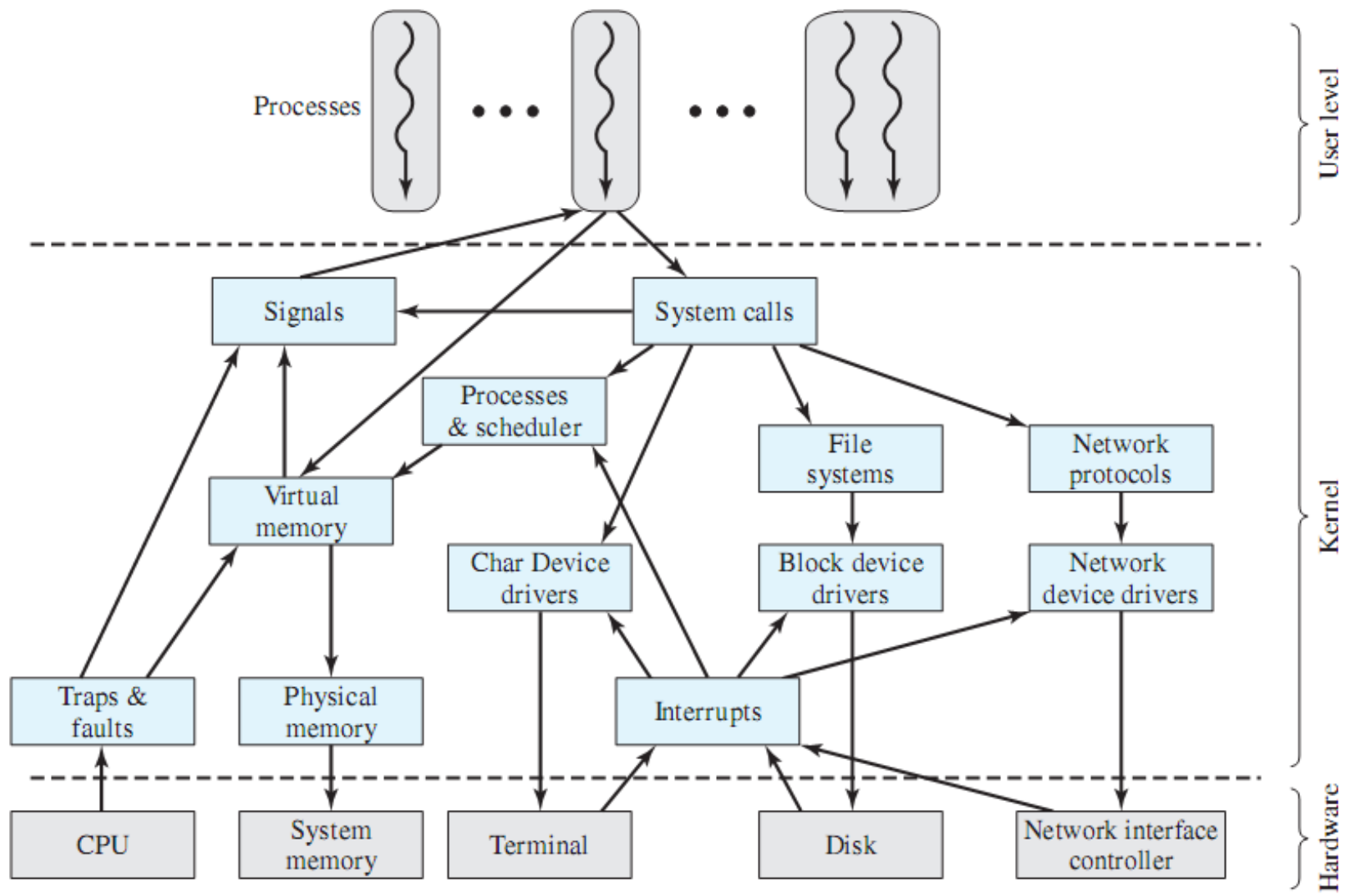
层次结构

内核结构



# LINUX 架 构

# Linux 内核组件



# Linux操作系统内核



# Android 架构

# Android操作系统的整体架构

## Android应用程序

Email客户端, SMS短消息程序, 日历, 地图, 浏览器, 联系人管理等

## 应用程序框架

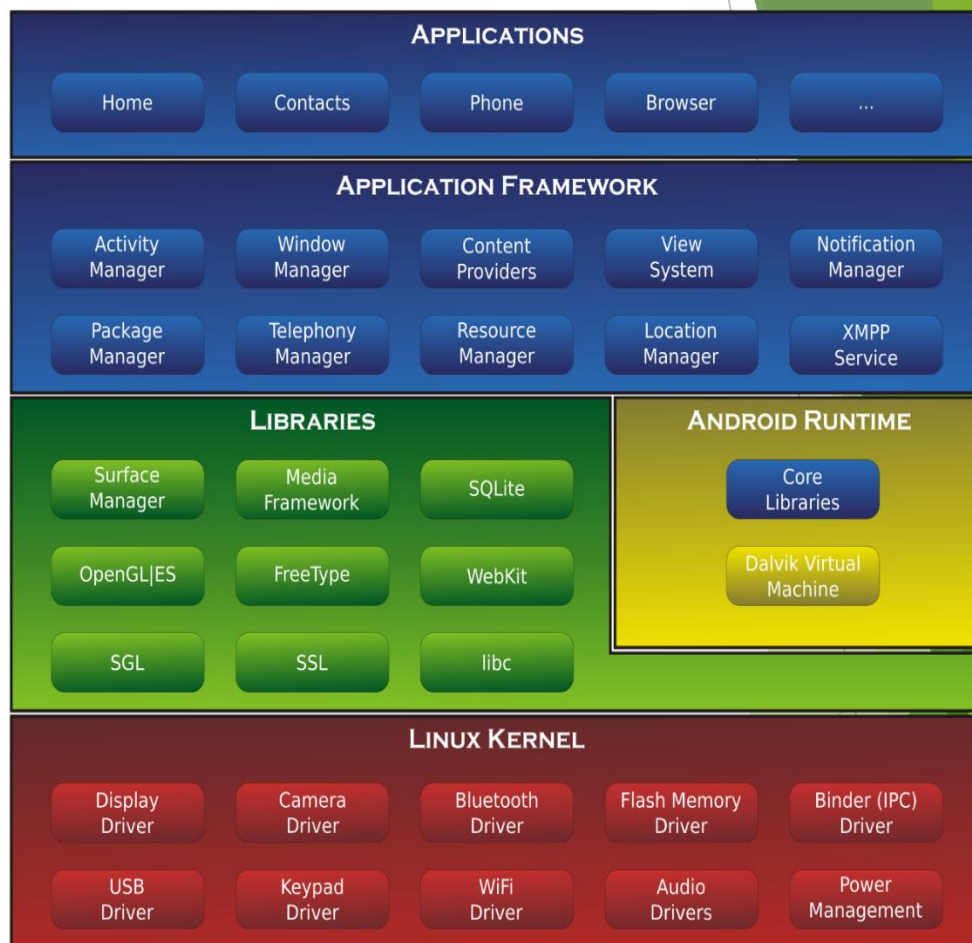
开发者可以完全使用核心应用程序所使用的框架APIs  
视图、内容提供者、资源管理等

## 库

Android包含一个C/C++库的集合, 供Android系统的各个组件使用。  
如: 系统C库、3D库、SQLite、媒体库等

## Linux内核

提供核心系统服务, 例如: 安全、内存管理、进程管理、网络堆栈、驱动模型



## 思考一下

从WINDOWS、UNIX和LINUX的系统结构图中得出什么结论？

从ANDROID架构图中得出什么结论？



操作系统是什么？怎样理解？特征.....

# 操作系统的定义

# 一、操作系统是什么？

---

操作系统是计算机系统中的—一个系统软件，  
是一些程序模块的集合——

- ▶ 它们能以尽量**有效**、**合理**的方式组织和管理计算机的软硬件资源
- ▶ 合理地组织计算机的工作流程，控制程序的执行并向用户提供各种服务功能
- ▶ 使得用户能够灵活、**方便**的使用计算机，使整个计算机系统能高效地运行

关键词解读

# 明确了三个目标

---

- ▶ 资源的管理者 → 有效
- ▶ 向用户提供各种服务 → 方便使用
- ▶ 机器的扩展 → 扩展能力

# 理解操作系统的作用

---

- ▶ 操作系统将一个物理资源(如处理器、内存或磁盘), 转换成一个更通用、更强大、更易用的虚拟形式, 将操作系统称为虚拟机
- ▶ 为了让用户告诉操作系统该做什么以及利用虚拟机的特性(比如运行程序、分配内存或访问文件), 操作系统还提供了一些可供用户调用的接口(APIs), 操作系统提供了应用程序的标准库
- ▶ 因为虚拟化允许多程序运行(如共享CPU), 允许多程序并发地访问自己的指令和数据(如共享内存), 允许多程序访问设备(如共享磁盘等等), 因此操作系统有时被称为资源管理器

# 设计与实现目标

---

- ▶ 抽象
  - ▶ 模块化
  - ▶ 使用高级语言 (C) 而非汇编
- ▶ 性能 (最小化开销)
  - ▶ 最小化额外的时间 (指令)
  - ▶ 最小化额外的空间 (内存/磁盘)
- ▶ 应用之间、操作系统与应用之间的保护
  - ▶ 隔离进程
- ▶ 可靠性
- ▶ 节能
- ▶ 安全性 (保护的扩展)
  - ▶ 防止恶意应用的入侵, 尤其是在高度网络化的时代
- ▶ 移动性
  - ▶ 使操作系统能够运行在越来越小的设备上

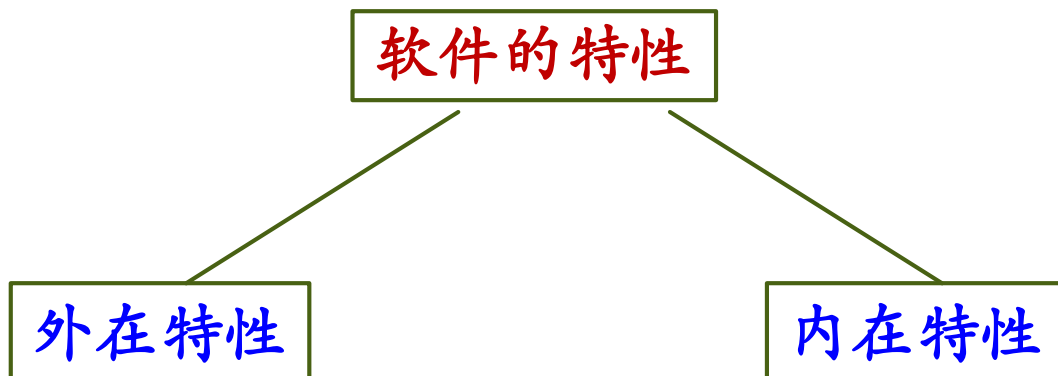
## 二、从不同角度认知操作系统

---

- ▶ 作为软件来看的观点
- ▶ 资源管理的观点
- ▶ 进程的观点
- ▶ 虚机器观点
- ▶ ……其他各种观点
  - 服务提供者、魔幻制造者、历史教员、家长、政府、仲裁者

# 1. 作为软件来看的观点

---



软件即是服务

界面/接口：使用方式  
如：命令、系统调用等

软件的结构

- ✓ 由哪几个部分组成？
- ✓ 各部分之间的关系？

## 2. 资源管理的观点

---

自底向上 → 操作系统 是 **资源的管理者**

硬件资源：

CPU，内存，设备（I/O设备、磁盘、时钟、网络接口等）

软件资源：

磁盘上的文件、信息

资源管理的目的：实现资源共享、提高资源利用率

两种方式实现复用（共享）：时间 及 空间



# 怎样管理资源?

公平竞争使用  
防止非法使用

- ▶ 跟踪记录资源使用状况
  - ▶ 如：哪些资源空闲，好坏与否，被谁使用，使用多长时间等
- ▶ 分配和回收资源（资源分配策略与算法）
  - 静态分配策略
  - 动态分配策略 ✓
- ▶ 提高资源利用率
- ▶ 保护
- ▶ 协调多个进程对资源请求的冲突

# 从资源管理的角度——五大基本功能

---

## ▶ 进程和线程管理（CPU管理、调度）

进程控制、同步互斥、通信、调度

## ▶ 存储管理

分配/回收、地址映射、存储保护、内存扩充

## ▶ 文件管理

文件目录、磁盘空间、文件系统布局、文件操作、存取控制

## ▶ 设备管理

设备驱动、分配回收、缓冲技术

## ▶ 用户接口

系统命令、编程接口

## 3. 进程的观点

---

从操作系统运行的角度动态的观察操作系统

按照这一观点：

► 操作系统 是

由一些可同时、独立运行的进程  
和一个对这些进程进行协调的核心组成

进程：完成某一特定功能的程序  
是程序的一次执行过程  
动态的、有生命的，存在/消亡

## 4. 虚机器观点

---

从操作系统内部结构来看：

- ✓ 把操作系统分成若干层
- ✓ 每一层完成其特定功从而构成一个虚机器，并对上一层提供支持
- ✓ 通过逐层功能扩充，最终完成整个操作系统虚机器
- ✓ 而操作系统虚机器向用户提供各种功能，完成用户请求

分层结构

并发、共享、虚拟、随机.....

# 操作系统的特征

# 操作系统的特征(1/4)

---

## 并发(concurrency):

处理多个同时性活动的能力

由并发引起的问题：活动切换、保护、相互依赖的活动间的同步

在计算机系统中同时存在多个程序

宏观上：这些程序是同时在执行

微观上：任何时刻只有一个程序在执行(单CPU)，即微观上这些程序在CPU上轮流执行

并行(parallel): 与并发相似，但多指有硬件支持

# 操作系统特征(2/4)

---

## 共享(sharing):

操作系统与多个用户程序共同使用计算机系统  
中的资源

- 共享有限的系统资源
  - 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 (如打印机)
  - ▶ 同时访问 (如可重入代码, 磁盘文件)
- 引出问题: 资源分配难以达到最优化, 保护

# 操作系统特征(3/4)

---

## 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

→ 虚拟技术是操作系统管理系统资源的重要手段，可提高资源利用率

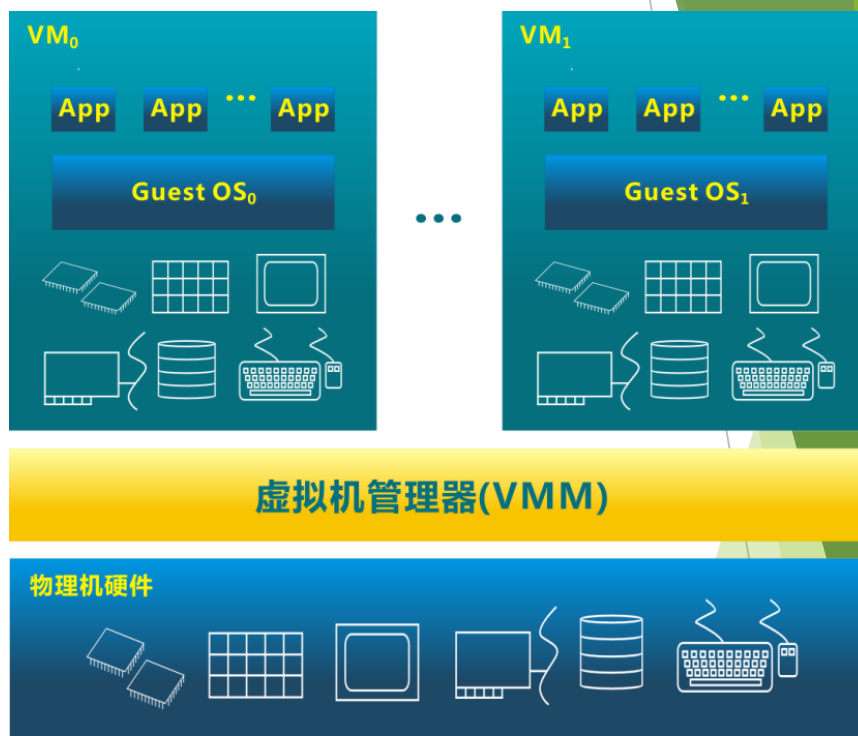
- ◆ CPU——每个用户（进程）的"虚处理机"
- ◆ 存储器——每个进程都占有的地址空间（代码+数据+堆栈）
- ◆ 显示设备——多窗口或虚拟终端



# VMM (虚拟机管理器)



无虚拟机：单操作系统拥有所有硬件资源



有虚拟机：多操作系统共享硬件资源

# 操作系统特征(4/4)

---

## 随机性:

操作系统必须随时对以不可预测的次序发生的事件进行响应

- ◆ 进程的运行速度不可预知：多个进程并发执行，“走走停停”，无法预知每个进程的运行推进快慢
- ◆ 难以重现系统在某个时刻的状态（包括重现运行中的错误）

划分为几个阶段

# 操作系统发展历史

# 为什么？启示/概念的重用、回归

---

技术变化 导致某些思想过时并迅速消失

但 技术的另一种变化还可能使它们复活

操作系统中的例子

- ✓ 磁盘上文件分配—连续文件（**CD-ROM**文件系统）
- ✓ 硬件保护
- ✓ 动态链接（**MULTICS**首先提出）
- ✓ 计算服务（**MULTICS**，以大量的、附有相对简单用户机器的、集中式Internet服务器形式回归） →  
云计算

# 操作系统的发展历程

---

- ▶ 操作系统发展是随着计算机硬件技术、应用需求的发展、软件新技术的出现而发展的

目标：充分利用硬件  
提供更好的服务

大型机 → PC机 → 网络 → 移动计算 → 云计算 → 泛在计算（物联网） → 大数据 → AI

# 第1阶段 (1948-1970)

---

硬件昂贵，人工便宜

更有效地利用硬件资源

缺乏用户和计算机之间的交互

- ▶ 控制台：一次一个用户（独占资源）
- ▶ 批处理：装入程序→运行→打印输出（保护：无）
- ▶ 操作系统必须管理所有程序的交接、运行→复杂
- ▶ 数据通道、中断：I/O和计算重叠
- ▶ 多道程序设计：多个程序同时运行，多个用户共享系统（需要存储保护）
- ▶ SPOOLing技术

## 第2阶段 (1970-1985)

---

硬件便宜，人工昂贵

- ▶ 交互、分时

利用便宜的终端 → 多个用户同时与系统交互  
牺牲CPU时间 → 用户得到更好的响应时间

- ▶ 用户可以在线工作：开发、调试、编辑等

- ▶ 引出的问题

增加用户时 → 系统性能降低  
(响应时间、抖动)

# 第一个分时操作系统CTSS

---

- ▶ 分时系统的思想——1959年在MIT提出
  - ✓ 每个用户有一个联机终端
  - ✓ 调试程序的用户常常只发出简短的命令  
很少有长的费时命令
  - ✓ 计算机能够为许多用户提供交互式、快速服务  
同时在CPU空闲时还能在后台运行大作业

第一个分时系统（CTSS）由MIT的Fernando Corbato等1961年在一改装的IBM 7090/94机上开发成功（有32个交互式用户）

指标：32K内存，系统用5K，用户用27K

用户存储映像在内存和一台磁鼓之间切换

1962年Manchester大学的Atlas计算机投入运行（运行速度200 kFLOPS）

第一个有虚拟存储器(virtual memory)和页面调度(paging)的机器



# 首次面对重大失败

---

- ▶ IBM 的 OS/360 发布时  
带着已知的 1000 个错误
- ▶ Multics 于 1963 年开始  
直至 1969 年才发布

(MULTiplexed Information and Computing Service )

ESSAYS ON SOFTWARE ENGINEERING



# 人月神话

The Mythical Man-Month: Essays on Software Engineering, Anniversary Edition (2nd Edition)

(美) Frederick P. Brooks, Jr.





# 成功案例

- ▶ 一群计算机迷在贝尔实验室开发出UNIX
- ▶ 初衷：可以在一台无人使用的DEC PDP-7 小型计算机上玩星际探险游戏

**Ken Thompson, Dennis Ritchie**  
**1983年图灵奖获得者**  
**1999年4月 美国国家技术金奖**



## 第3阶段 (1981-)

---

硬件非常便宜，人工非常昂贵

面临挑战 → 如何利用计算机  
充分发挥人的时间

个人计算时代

- ▶ 开始 PC硬件资源有限 一次运行一个程序 **OS是一个例程库** 回归简单
- ▶ 逐渐 PC资源丰富 OS又成为一个庞然大物（大型OS）：**存储保护、多道程序设计再次出现**

## 第4阶段 (1981-)

---

### 分布式

- ▶ 网络：允许不同机器很容易地相互共享资源（打印机、文件服务器、Web服务器）
- ▶ 解决问题：共享，安全

## 第5阶段 (1995-)

---

### 移动计算时代

- ▶ 各种移动终端的出现  
(笔记本、平板、手机、机顶盒、可穿戴设备等)
- ▶ 特点：小型、移动、便宜  
但 有限能力

# 第6阶段 (2006-)

## 云计算时代

- ▶ IT基础设施
- ▶ 提供可无限扩展的、可随时获取的、按需使用、按使用付费的资源网络
- ▶ 云计算操作系统——云计算后台数据中心的整体管理运营系统，是指构架于服务器、存储、网络等基础硬件资源和单机操作系统、中间件、数据库等基础软件管理的海量的基础硬件、软资源之上的云平台综合管理系统
- ▶ 云计算操作系统：大规模基础软硬件管理、虚拟计算管理、分布式文件系统、业务/资源调度管理、安全管理控制等模块
- ▶ 云计算操作系统的作用：一是能管理和驱动海量服务器、存储等基础硬件，将一个数据中心的硬件资源逻辑上整合成一台服务器（治众如治寡）；二是为云应用软件提供统一、标准的接口；三是管理海量的计算任务以及资源调配



## 第7阶段 (200? —)

---

### ► 泛在计算/普适计算/物联网

许多联网设备为许多人提供个性化的服务

传统分类、Tenenbaum 分类

# 操作系统分类

# 操作系统的分类

---

- ▶ 批处理操作系统（多道批处理）
- ▶ 分时系统
- ▶ 实时操作系统
- ▶ 个人计算机操作系统
- ▶ 网络操作系统
- ▶ 分布式操作系统
- ▶ 嵌入式操作系统

通用操作系统

# 1. 批处理操作系统

---

## ► 工作方式

1. 用户将作业交给**系统操作员**
2. 系统操作员将许多用户的作业组成一批作业，输入到计算机系统中，在系统中形成一个自动转接的连续作业流
3. 启动操作系统
4. 系统自动、依次执行每个作业
5. 由操作员将作业结果交给用户

# 批处理操作系统

---

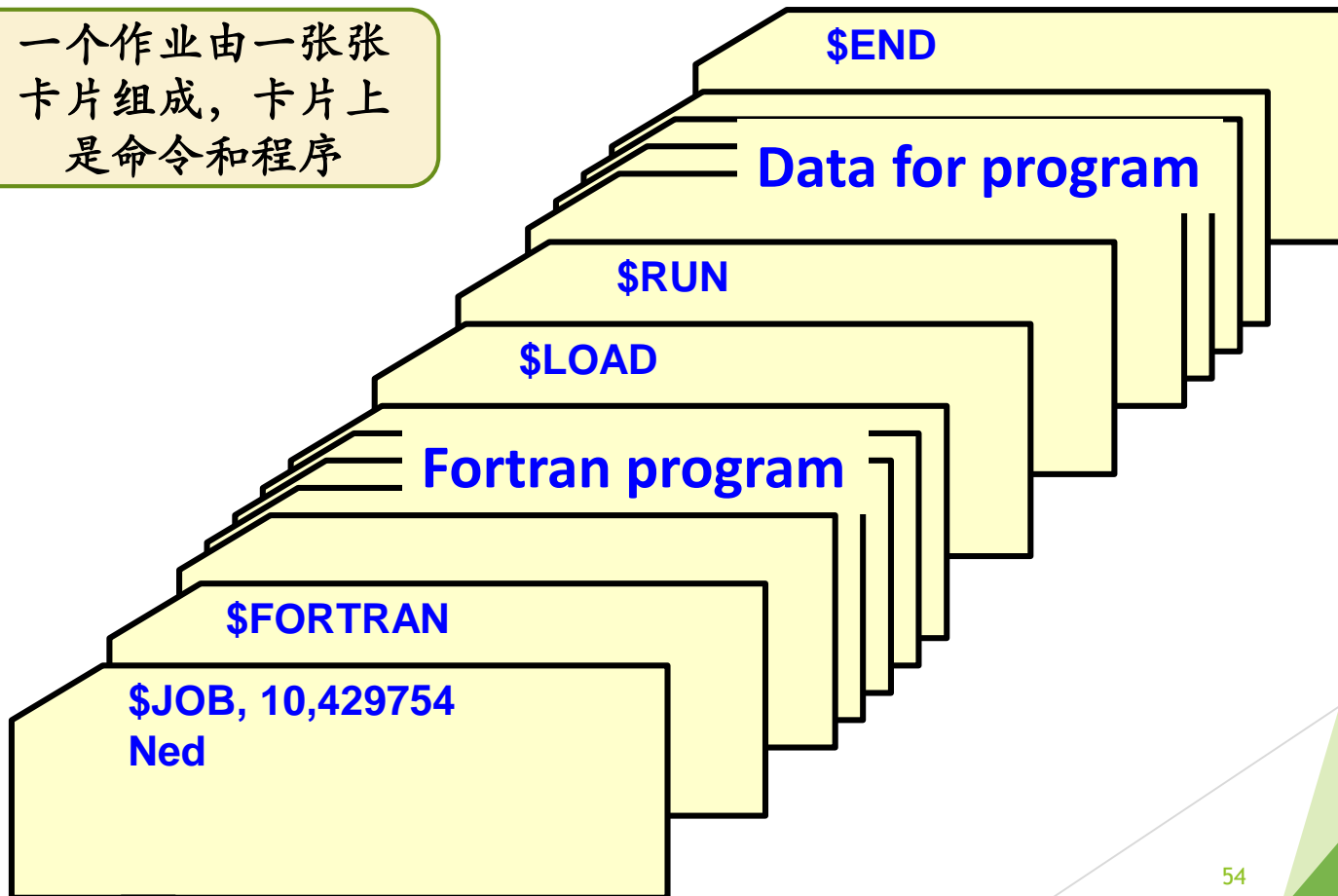
- ▶ 批处理系统中的作业的组成：
  - 用户程序
  - 数据
  - 作业说明书（作业控制语言）
- ▶ 批：

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）
- ▶ 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

# 典型的FMS JOB结构

一个作业由一张张卡片组成，卡片上是命令和程序



# 批处理操作系统

---

## ► 问题:

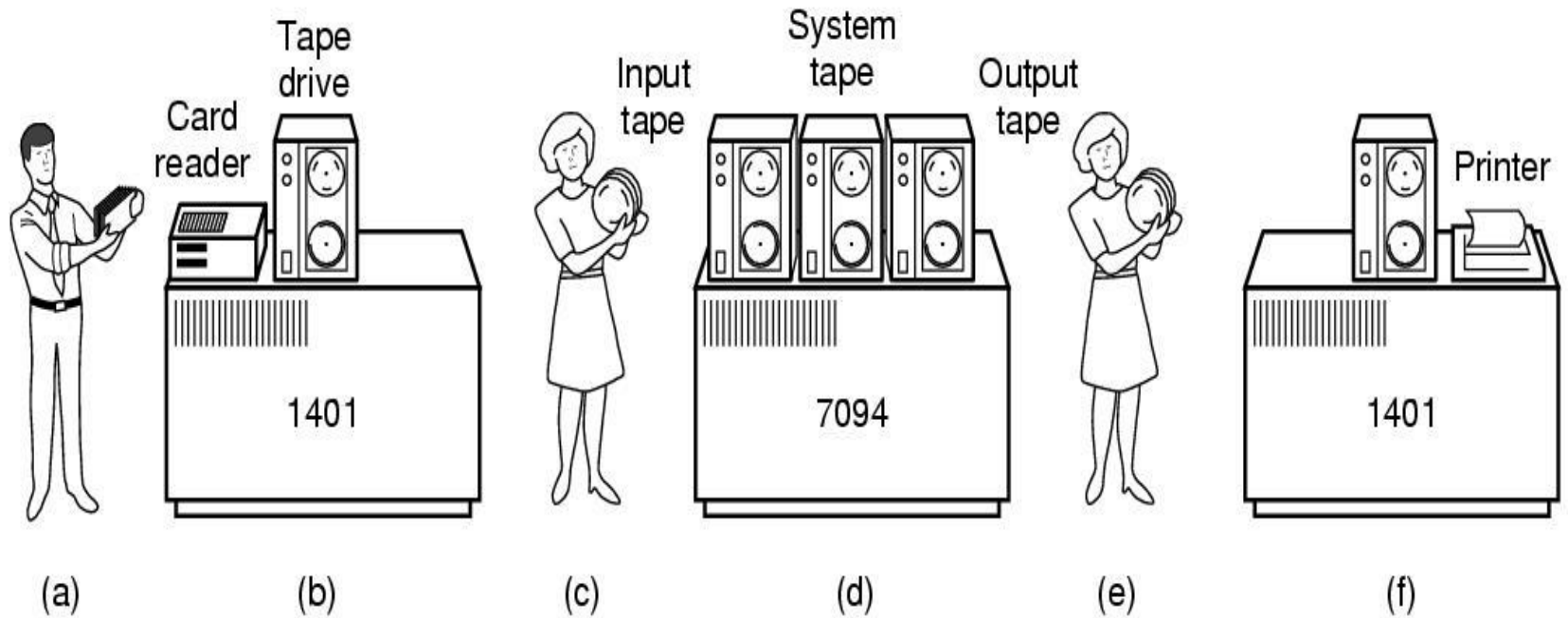
慢速的输入输出处理直接由主机来完成，输入输出时，CPU 处于等待状态

## ► 解决:

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

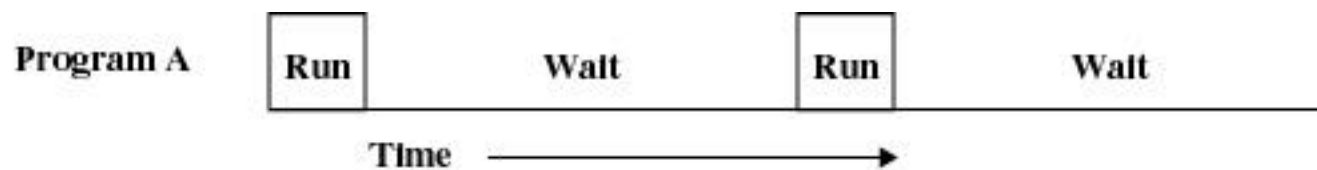
- ✓ 单道批处理系统（simple batch processing, uni-programming）
- ✓ 多道批处理系统（multiprogramming system）

# 早期的批处理系统

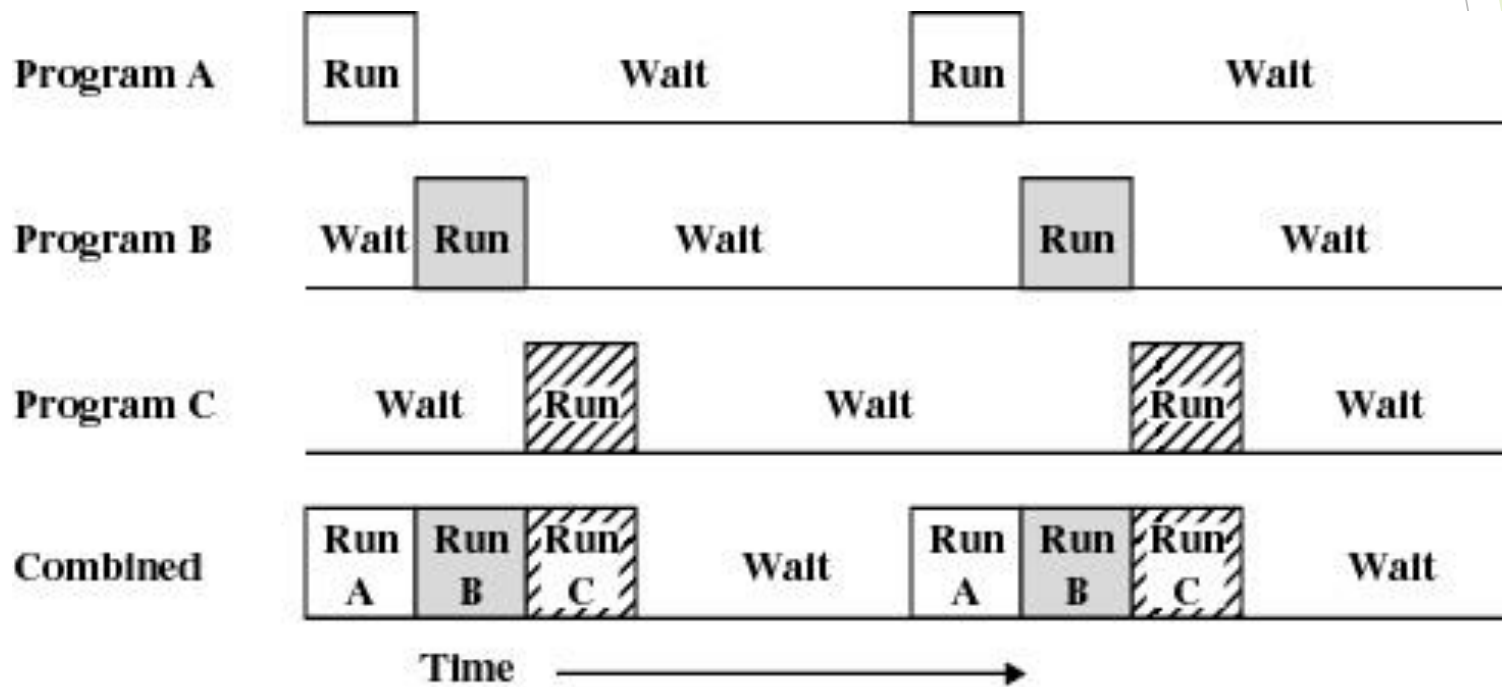




# 例子



单道程序设计



多道程序设计

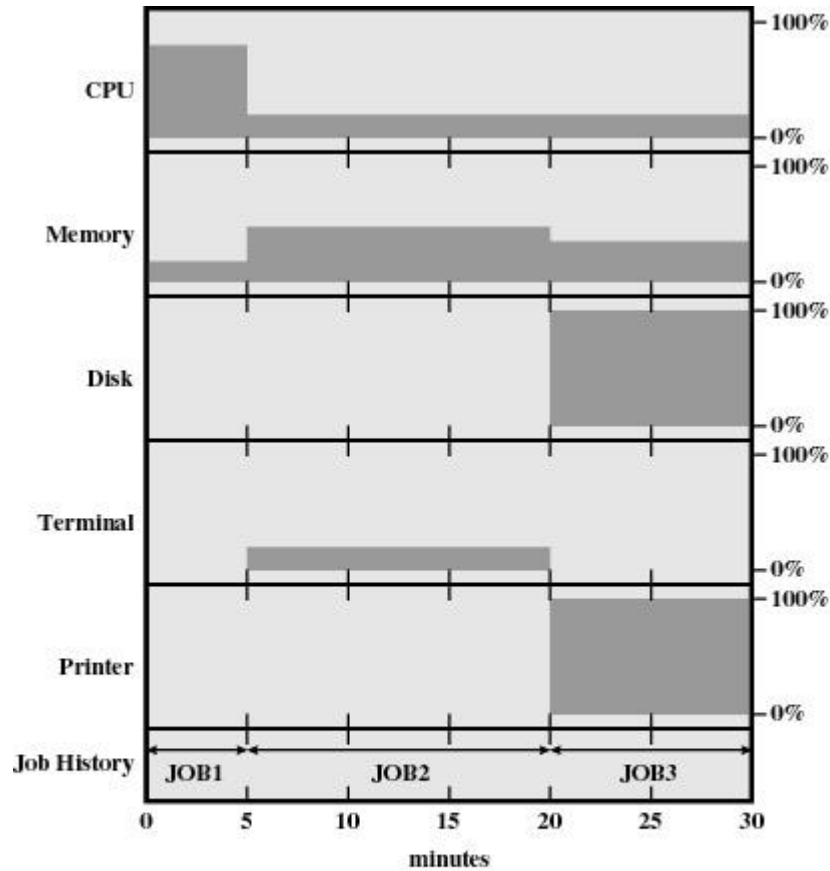
ms

	JOB1	JOB2	JOB3
Type of job	Heavy compute	Heavy I/O	Heavy I/O
Duration	5 min.	15 min.	10 min.
Memory required	50K	100 K	80 K
Need disk?	No	No	Yes
Need terminal	No	Yes	No
Need printer?	No	No	Yes

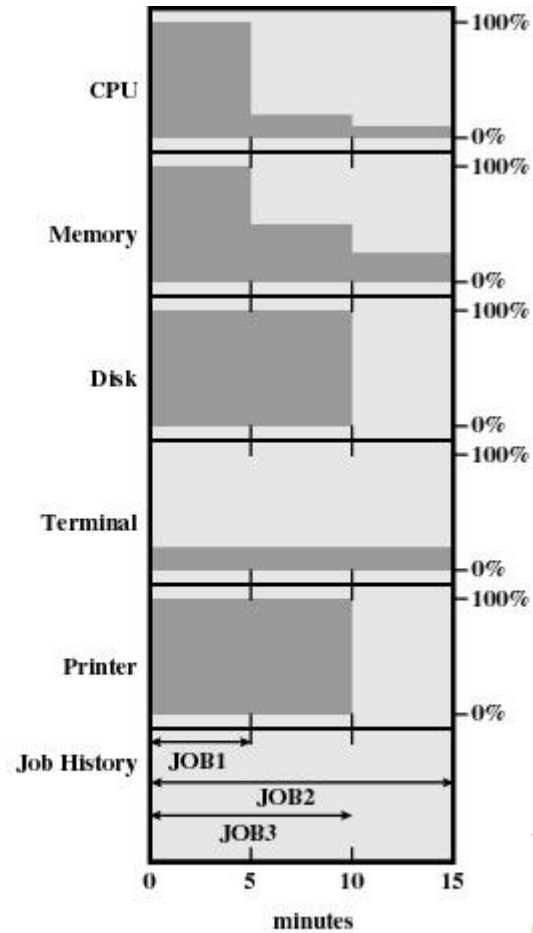
计算

	Uniprogramming	Multiprogramming
Processor use	22%	43%
Memory use	30%	67%
Disk use	33%	67%
Printer use	33%	67%
Elapsed time	30 min.	15 min.
Throughput rate	6 jobs/hr	12 jobs/hr
Mean response time	18 min.	10 min.

# 效果



(a) Uniprogramming



(b) Multiprogramming

# SPooling系统（技术）

---

- ▶ 1961年，英国曼彻斯特大学，Atalas机
- ▶ Simultaneous Peripheral Operation On-Line

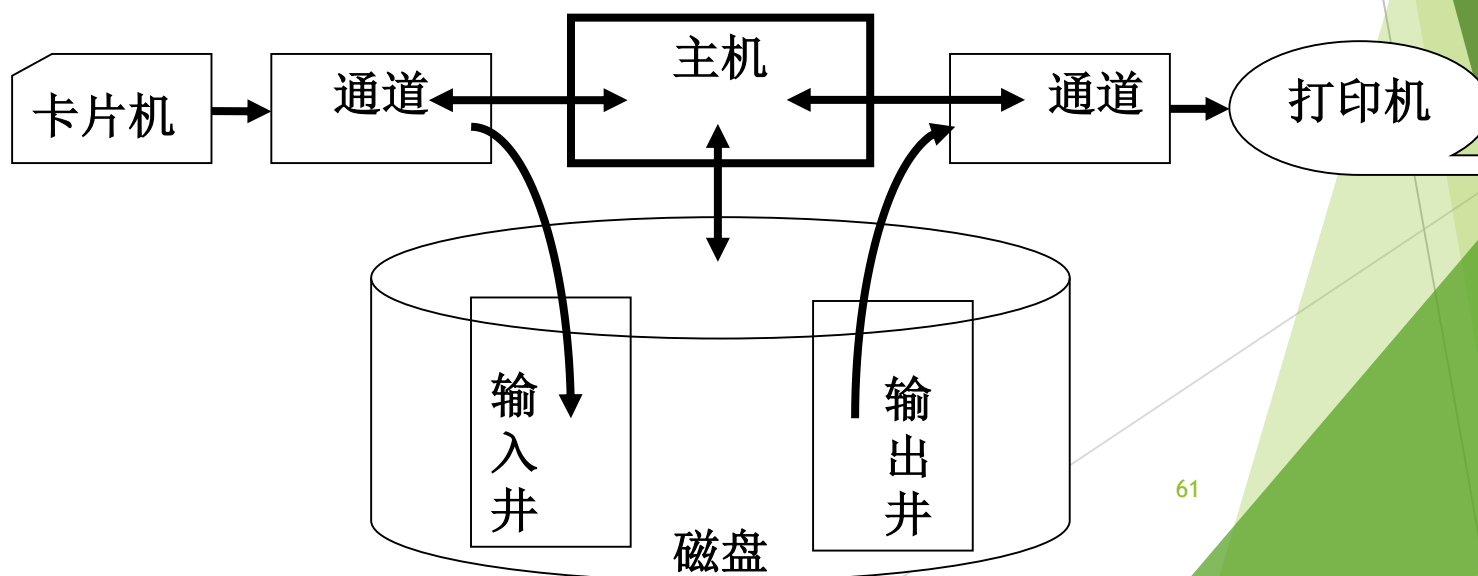
同时的外围设备联机操作 -- 假脱机技术

- ▶ 思想：

利用磁盘作缓冲，将输入、计算、输出分别组织成独立的任务流，使I/O和计算真正并行

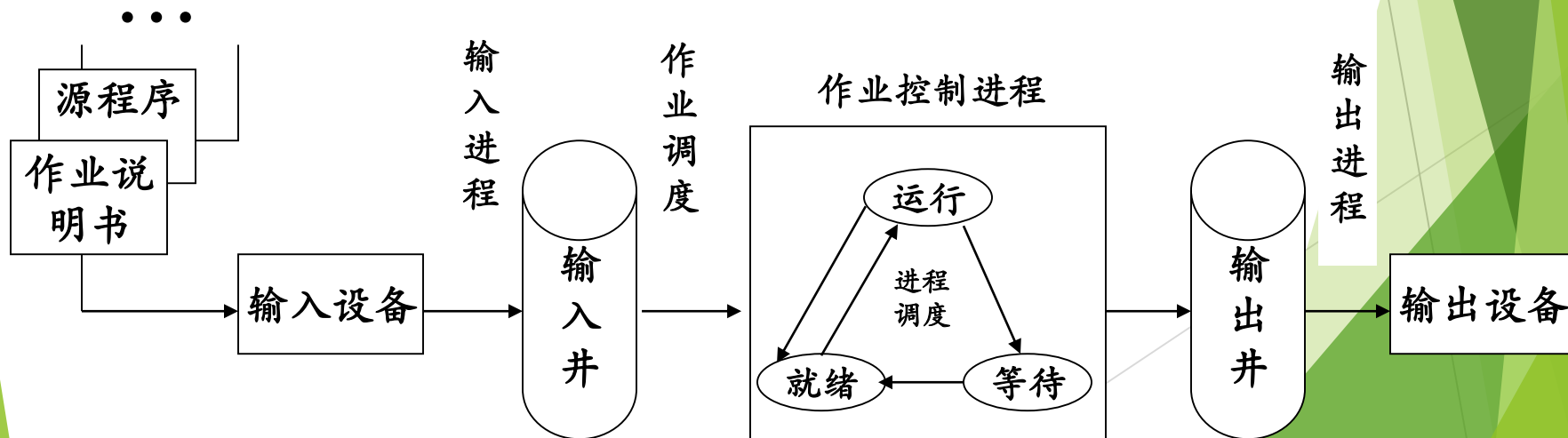
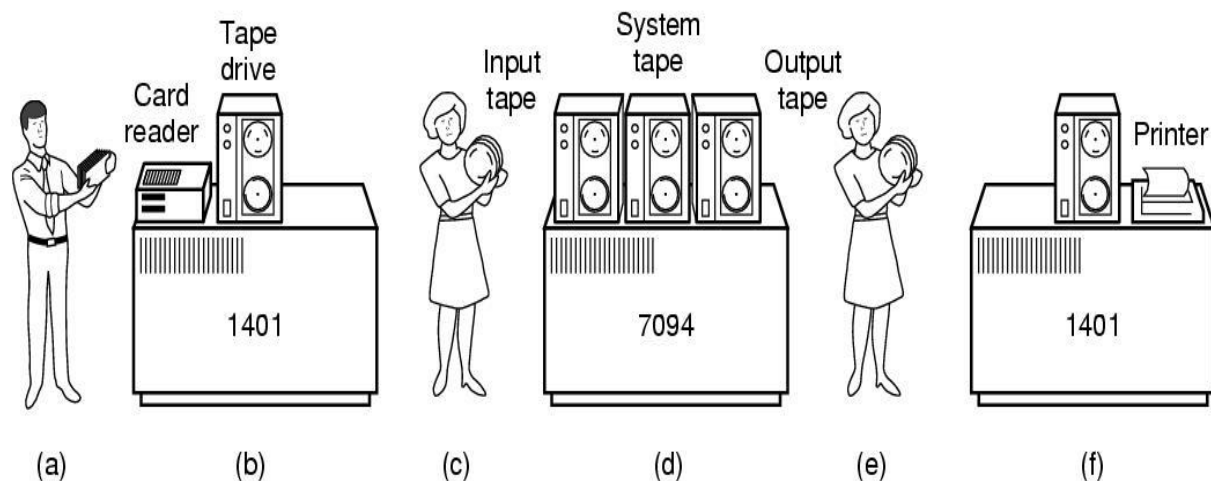
# SPOOLing系统工作原理

- ▶ 作业进入到磁盘上的输入井
- ▶ 按某种调度策略选择几种搭配得当的作业，并调入内存
- ▶ 作业运行的结果输出到磁盘上的输出井
- ▶ 结果从磁盘上的输出井送到打印机



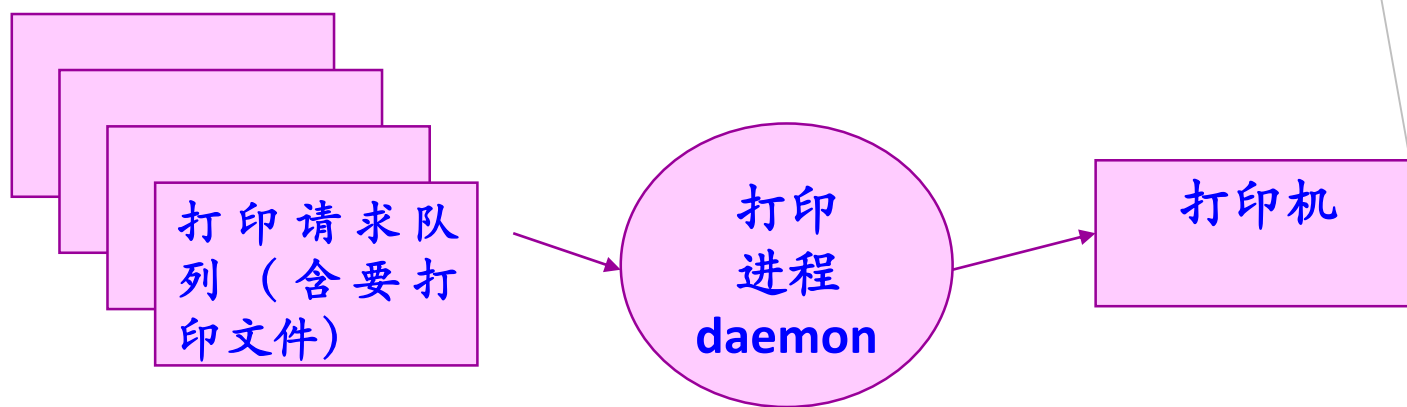
# 多道批处理操作系统

## 假脱机技术

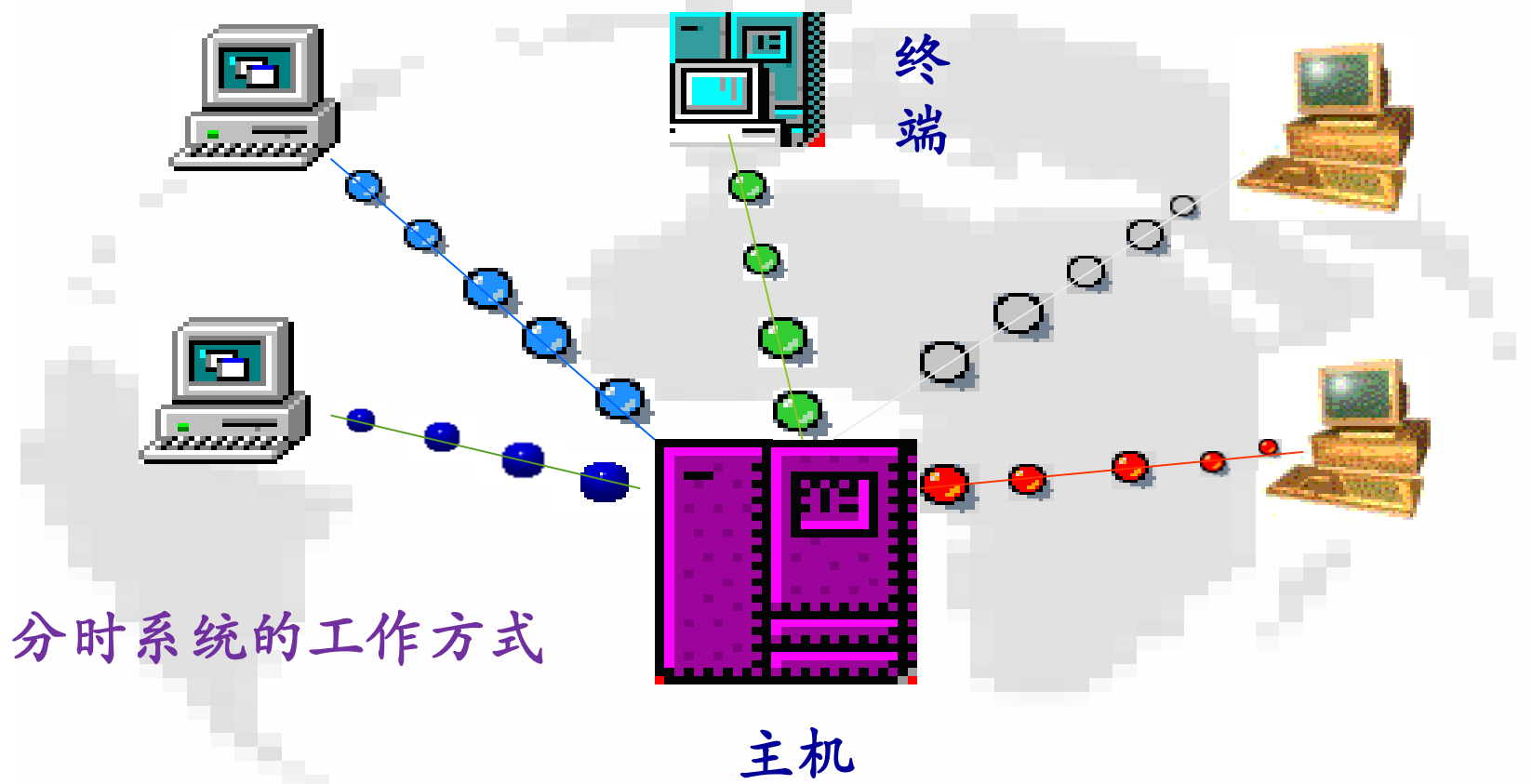


# 以 SPOOLing 方式使用外设

---



## 2. 分时操作系统 (time-sharing system)





# 分时操作系统

---

## ► 时间片 (time slice)

操作系统将CPU的时间划分成若干个片段，称为时间片

- ✓ 操作系统以时间片为单位，轮流为每个终端用户服务
- ✓ 每次服务一个时间片

(其特点是利用人的错觉，使人感觉不到)

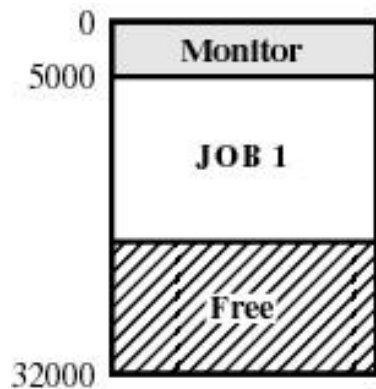
追求目标：

及时响应(依据是响应时间)

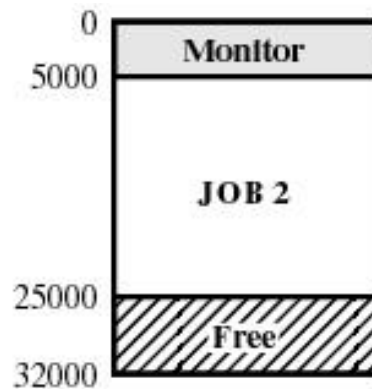
响应时间：

从终端发出命令到系统给予回答所经历的时间

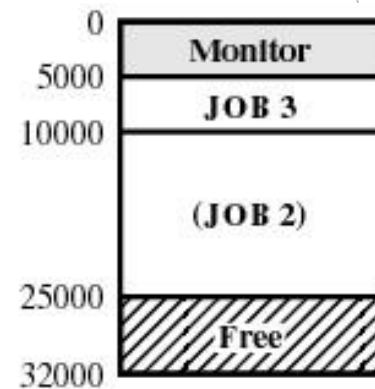
# CTSS分时系统



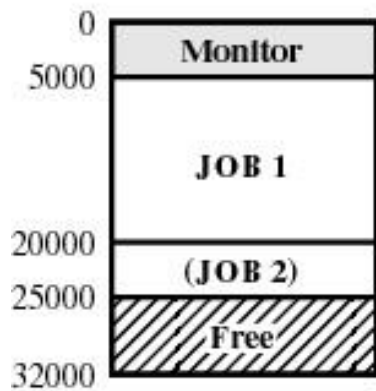
(a)



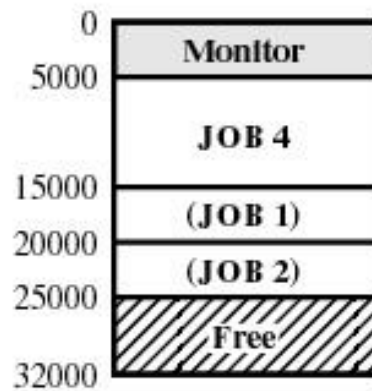
(b)



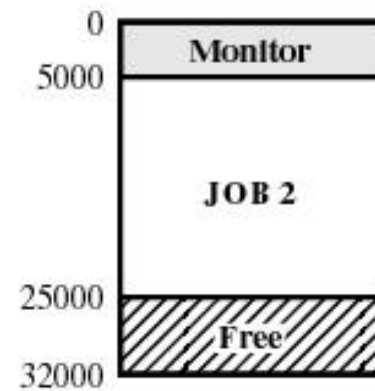
(c)



(d)



(e)



(f)

# 通用操作系统

---

- ▶ 分时系统与批处理系统结合
- ▶ 原则：分时优先，批处理在后
  - “前台”：需频繁交互的作业
  - “后台”：时间性要求不强的作业

### 3. 实时操作系统

---

是指使计算机能及时响应外部事件的请求，在规定的严格时间内完成对该事件的处理，并控制所有实时设备和实时任务协调一致地工作的操作系统

分类：

► **第一类：实时过程控制**

工业控制，军事控制，...

► **第二类：实时通信（信息）处理**

电讯（自动交换），银行，飞机订票，股市行情

# 实时操作系统

---

追求目标：

- ▶ 对外部请求在严格时间范围内作出反应
- ▶ 高可靠性

特征：

关键参数 是 时间

例子：工业过程控制系统——汽车装配线

## ◆ 硬实时系统（例子？）

某个动作绝对必须在规定的时刻或时间范围完成

## ◆ 软实时系统（例子？）

接受偶尔违反最终时限

## 4. 个人计算机操作系统

---

► 计算机在某一时间内为单用户服务

► 追求目标:

界面友好, 使用方便  
丰富的应用软件

## 5. 网络操作系统

---

基于计算机网络

在各种计算机操作系统上

按网络体系结构协议标准开发的软件

- ▶ 功能：网络管理，通信，安全，资源共享和各种网络应用
- ▶ 追求目标：相互通信，资源共享

## 6. 分布式操作系统

---

- ▶ 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的分布（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



# 分布式操作系统的特征

---

## 1. 是一个统一的操作系统

若干个计算机可相互协作共同完成一项任务

## 2. 资源进一步共享

## 3. 透明性

资源共享，分布对用户来讲是不知道的

## 4. 自治性

处于分布式系统的多个主机处于平等地位，无主从关系

## 5. 处理能力增强、速度更快、可靠性增强

## 7. 嵌入式操作系统

### ► 嵌入式系统

- 在各种设备、装置或系统中，完成特定功能的软硬件系统
- 它们是一个大设备、装置或系统中的一部分，这个大设备、装置或系统可以不是“计算机”
- 通常工作在反应式或对处理时间有较严格要求环境中

### ► 嵌入式操作系统 (Embedded Operating System)

运行在嵌入式系统环境中，对整个嵌入式系统以及它所操作、控制的各种部件装置等等资源进行统一协调、调度、指挥和控制的系统软件

美国加州伯克利大学研制：微型智能传感器，安装TinyOS



# 操作系统的另一种分类 (Tanenbaum)

---

- ▶ 大型机操作系统
- ▶ 服务器操作系统
- ▶ 多处理机操作系统
- ▶ 个人计算机操作系统
- ▶ 实时操作系统
- ▶ 嵌入式操作系统
- ▶ 智能卡操作系统

# 智能卡操作系统

---

智能卡：一种包含有一块CPU芯片的信用卡

## ► 特点

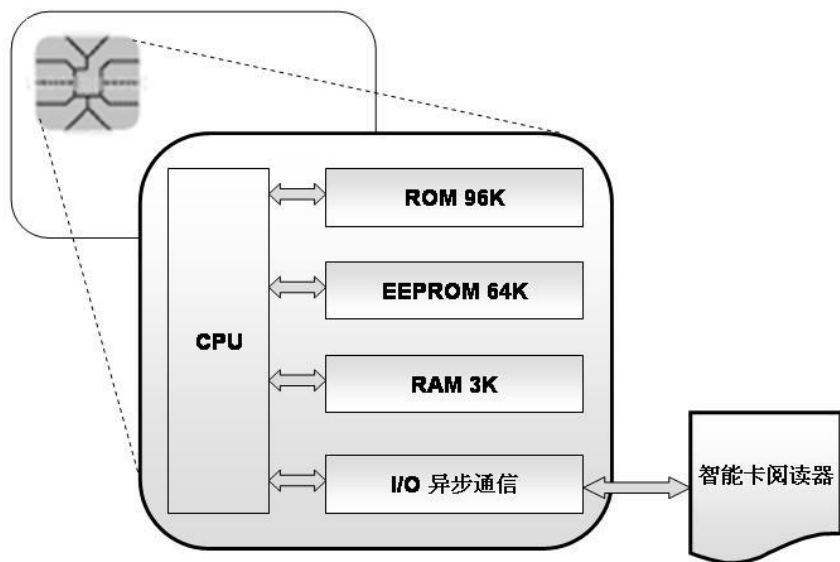
非常严格的运行能耗和存储空间的限制

有些智能卡只有单项功能，诸如电子支付

## ► 专用的操作系统

有些智能卡是面向Java的。其含义是在智能卡的ROM中有一个Java虚拟机解释器。Java小程序被下载到卡中并由JVM解释器解释。有些卡可以同时处理多个Java小程序，这就是多道程序，并且需要对它们进行调度。在两个或多个小程序同时运行时，资源管理和保护就成为突出的问题。这些问题必须由卡上的操作系统处理

# 智能卡操作系统



- 在读写器与智能卡之间通过“命令-响应对”方式进行通信和控制
- 读写器发出操作命令，智能卡接收命令
- 操作系统对命令加以解释，完成命令的解密与校验
- 操作系统调用相应程序来进行数据处理，产生应答信息，加密后送给读写器

# 重点小结

---

- ▶ 操作系统的概念
  - ▶ 理解操作系统的不同角度
  - ▶ 操作系统的主要特征
- ▶ 典型的、历史/现实中有重要意义的操作系统
- ▶ 重要的操作系统技术及相关技术
  - ▶ 多道程序设计、中断、通道、SPOOLing技术
- ▶ 操作系统的分类
- ▶ 操作系统结构

**SPOOLing技术    操作系统结构    多道程序设计**

**OS/360    MULTICS    并发    共享    虚拟    随机性**

# 作业1

---

1、何谓“多道程序设计”？阅读“Three Easy Pieces”的Introduction中“The Era of Multiprogramming”，结合本讲义57-59页内容，查找相关资料，总结你对多道程序设计技术的理解。

2、阅读<http://pages.cs.wisc.edu/~remzi/OSTEP/dialogue-threeeasy.pdf>，简要回答为什么这本教材叫“Three Easy Pieces”。

3、简要总结<http://pages.cs.wisc.edu/~remzi/OSTEP/intro.pdf>的2.1和2.2。

提交时间：2020年10月4日晚23:30

*Thanks*

*The End*