os

introduction os

role of os

1. 用户与计算机硬件系统之间的接口
2. 计算机资源的管理者
3. 扩充机器/control problem ,虚拟机

定义：操作系统是一组控制和管理计算机软硬件资源，合理对各类作业进行调度以及方便用户的程序的集合

层次模型：一种经典的操作系统的结构模型

1. 最高层：接口
2. 中间层：对对象进行操作和管理的软件集合
3. 最底层：OS操纵和管理的对象，包括各种软硬件资源

操作系统设计的目标

1. 方便性

2. 有效性
3. 可扩充性
4. 开放性 方便性和有效性是操作系统最重要的两个目标

系统的启动：Power-on->Bootstrap:BIOS->BootLoader:GRUB->OS:Linux

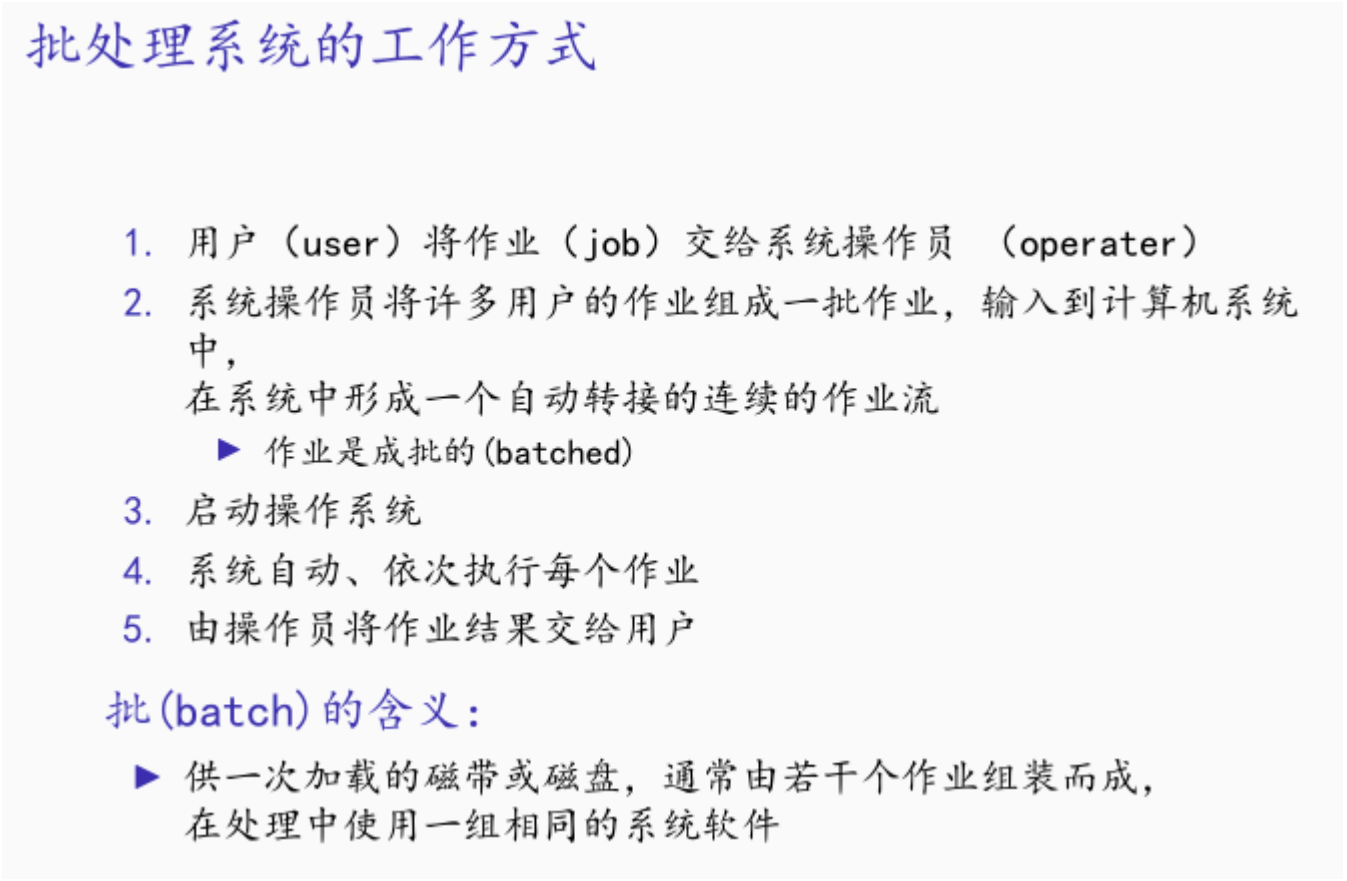
BIOS：

OS history

操作系统的发展动力

1. 不断提高计算机资源的利用率的需要
2. 方便用户
3. 器件的不断更新换代
4. 计算机体系结构的不断发展

历程：无OS时代->批处理系统->分时系统->实时系统->PC->分布式和并行系统



批的含义：共一次加载的磁带或磁盘，通常由若干个作业组装而成，在处理中使用一组相同的系统软件

批处理系统的引入是为了提高系统资源的利用率和吞吐量，特征为自动行，顺序性，单道性

脱机I/O 目的：解决人机矛盾和CPU与I/O设备之间速度不匹配的矛盾 方法：利用低速的外围机进行，纸带(卡片)->磁带(磁盘)

脱机的内涵：程序和数据都在脱离主机控制下，由外围机控制完成。

多道批处理系统的特征

1. 多道性
2. 无序性
3. 调度性

分时系统的特征：多路性，独立性，及时性，交互性

设计目标：及时响应，其依据是响应时间

实时系统有实时任务和非实时任务

实时任务的分类

1. 按任务执行是否呈现周期性来划分 周期性的，有规律； 非周期性的，无规律，但有截止时间 开始截止时间 vs. 完成截止时间
2. 根据对截止时间的要求来划分 硬实时任务 vs. 软实时任务

实时操作系统追求的设计目标：

1. 满足实时性要求：
2. 对外部请求在严格时间范围内作出反应
3. 高可靠性

一个实际的操作系统，往往兼有上述三种基本操作系统类型的功能（批，分时，实时）

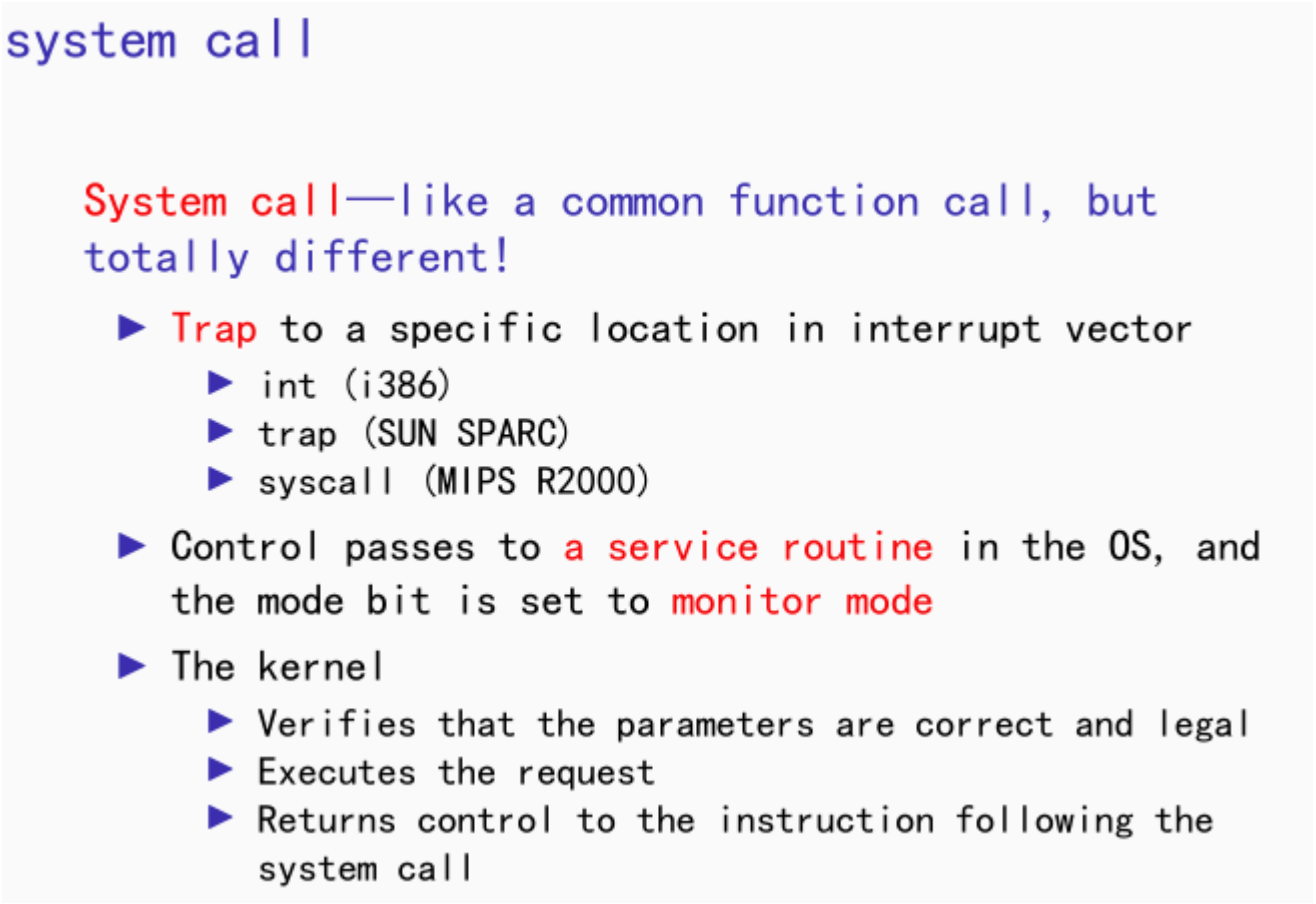
smp 对称多处理器

Some use two-step process: a simple bootstrap loader fetches a more complex boot program from disk, which in turn loads the OS

Master Boot Record, MBR, 主引导记录 ▶ the first sector on a hard drive, a special type of boot sector

▶ Hardware must provide protection

1. Dual-Mode Operation
2. I/O protection
3. Memory protection
4. CPU protection



os structure

操作系统的组成

1. 进程管理
    1. 进程控制
    2. 同步
    3. 进程间的通信
    4. 作业和进程的调度
2. 内存管理
3. I/O管理：设备独立性，program与设备无关，增加了程序的可移植性
4. 文件管理
5. 辅存/外存管理
6. 命令解释系统
    1. 命令接口
    2. 程序接口

    3. 图形接口
7. 保护
8. 网络

操作系统提供服务的最基本方式-系统调用

参数传递的方式

1. 通用寄存器传参
2. 使用块或者表，通过寄存器传递其起始地址
3. 栈

系统调用的类型

1. 进程控制类
2. 文件管理类
3. 设备管理类
4. 通信类
5. 信息维护类

操作系统的特征

1. 并发：并行是指两个或多个时间在同一时刻发生，并发是两个或多个时间在统一时间间隔内发生，并发包括并行

2. 共享：系统中资源可供内存中多个并发执行的进程共同使用

    1. 互斥共享：一段时间内只允许一个进程访问资源
    2. 同时访问

3. 虚拟：通过某种技术把一个物理实体变为若干逻辑上的对应物

4. 异步：运行进度不可预知

并发与共享是操作系统最基本的特征

程序与进程：程序是静态实体，进程是动态实体。

进程间通信机制：信号，信号量，管道，套接字，消息队列

操作系统的结构

1. 无结构
2. 单一大内核

3. 模块化结构
4. 层次结构
5. 微内核
6. 混合内核
7. 外核

现在很多os很多结构都有，鸿蒙是微内核

机制与策略相分离：机制决定如何去做（定时器），策略决定做什么（为每个用户决定将定时器设置为多长）

process

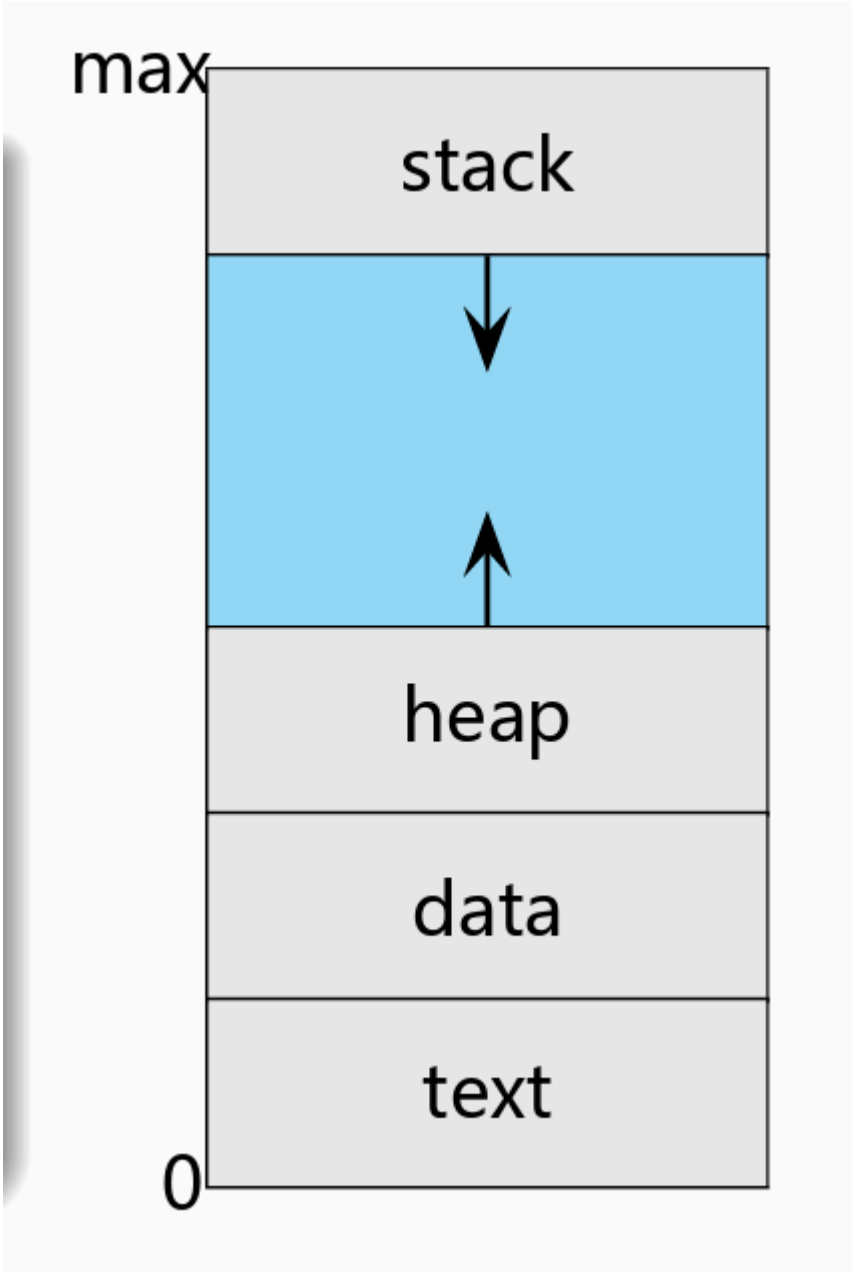
程序顺序执行的特征

1. 顺序性
2. 封闭性
3. 可再现性

程序并发执行的特征

1. 间断性
2. 失去封闭性
3. 不可再现性

必须防止不可在现性：R(p1)andW(p2)orR(p2)andW(p1)orW(p1)andW(p2)（bernstein



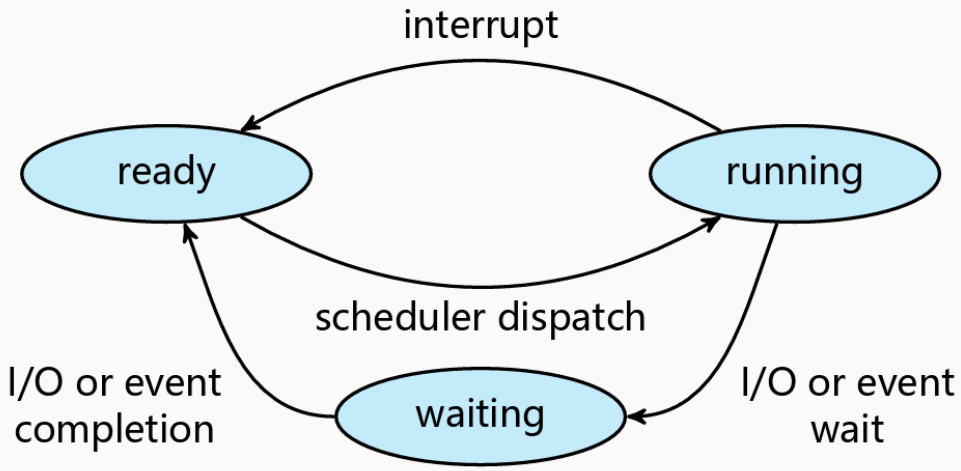
条件

进程的五大特征

1. 动态性（最基本的一点，具有生命期
2. 并发性
3. 独立性
4. 异步性
5. 结构特征

1 “三状态”模型

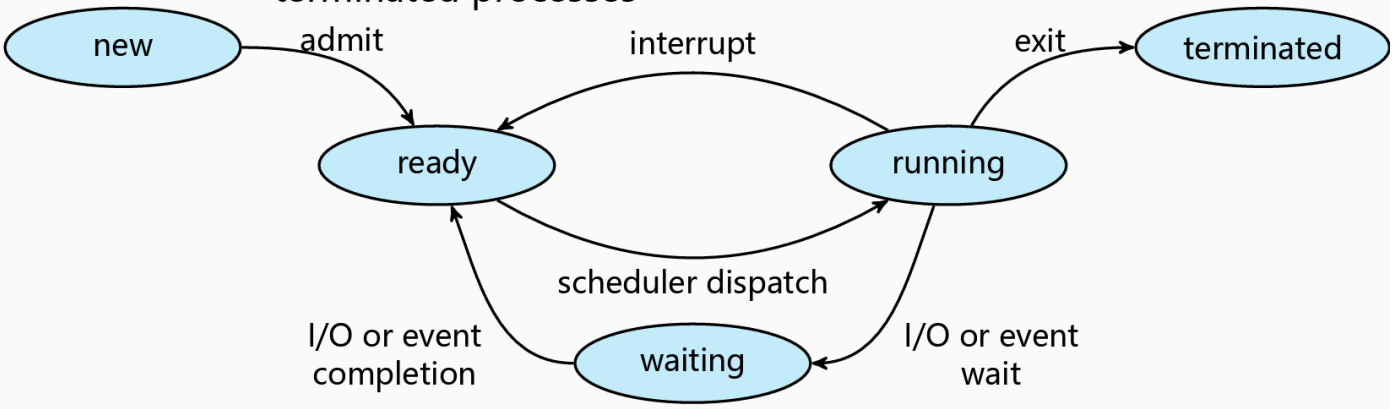
- 三种最基本的状态
  1. **ready** (就绪): “万事具备，只欠CPU”
  2. **running** (执行)
  3. **waiting** (等待, also blocked(阻塞), sleeping(睡眠))



4 types of state transferring

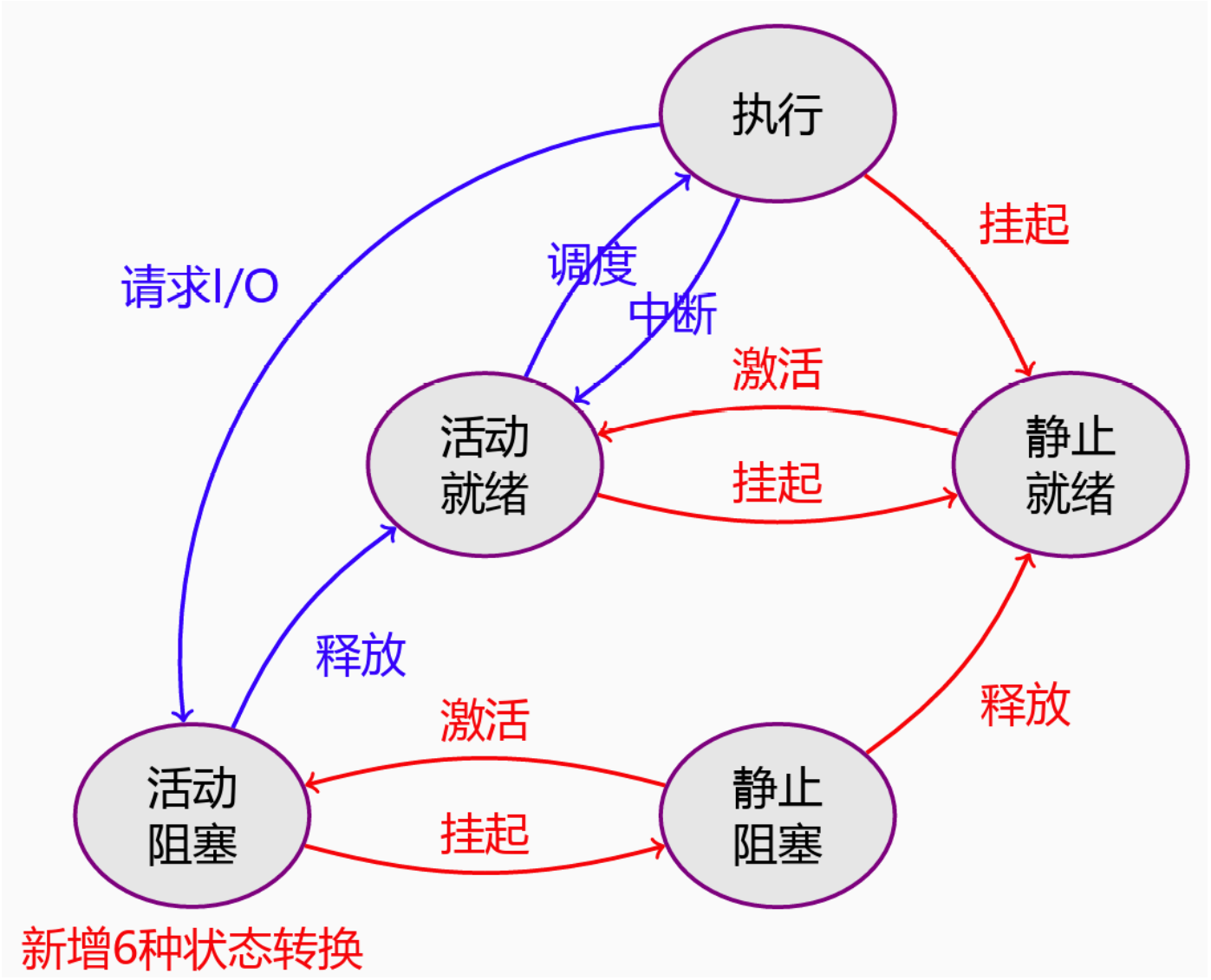
2 “五状态”模型

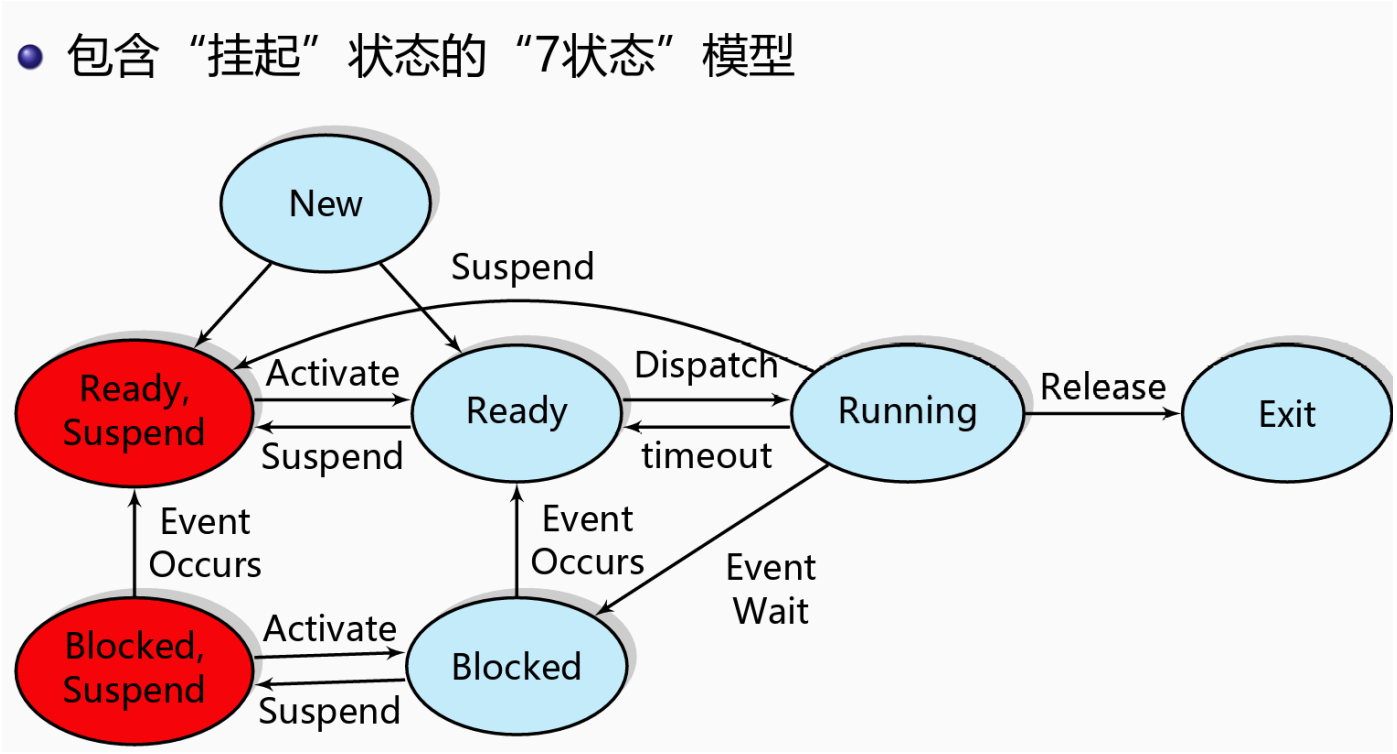
- Two more states is added to the “three state” model.
  1. **new** (新生状态): The process is beig created
     - ★ initialization, resource preallocation, etc.
  2. **terminated** (终止状态): The process has finished execution, normally or abnormally.
     - ★ removed from ready queue, but still not destroyed.
     - ★ other process may gather some information from the terminated processes



6 types of state transferring

外部原因可能导致挂起，挂起是一类状态





PCB是进程存在的唯一标志

长期调度：选择进程到就绪队列

短期调度：选择进程cpu执行

多道程序度：系统内部进程的个数，由长期调度决定

I/O密集型，CPU密集型，好的长期调度会混合他们

中期调度：引入挂起状态时会有，也会影响多道程序度

上下文切换的时间依赖于硬件支持

进程间通信：

1. 消息传递（传递和接受都是阻塞的就叫集合点
2. 共享内存

cpu scheduling

Process execution = n(CPU execution + I/O wait)+CPUexecution

抢占调度与非抢占调度

抢占调度需要定时器，导致异步

可抢占内核：并非系统支持抢占调度，而是系统执行系统调用，中断处理的时候可以抢占

Dispatch latency– time it takes for the dispatcher to stop one process and start another running

调度算法的指标

1. cpu利用率
2. 吞吐率
3. 周转时间：进程提交到进程完成
4. 等待时间：就绪队列中等待所花时间之和
5. 响应时间

调度算法

1. fcfs,平均等待时间很长
2. sjf(最小平均等待时间是最优的，可抢占叫srtf,缺点是未来的时间未知)
3. rr(有抢占)

护航效应：所有其他进程都在等待一个大进程完成（短进程等待长进程

饿死现象：解决方案：aging，等待时间越长，优先级越高

多级队列：队列间的分配，队列内的分配

多级反馈队列，涉及优先级的升降

Processor affinity(处理器 亲和性):a process has an affinity for the processor on which it is currently running.(软亲和性，硬亲和性)

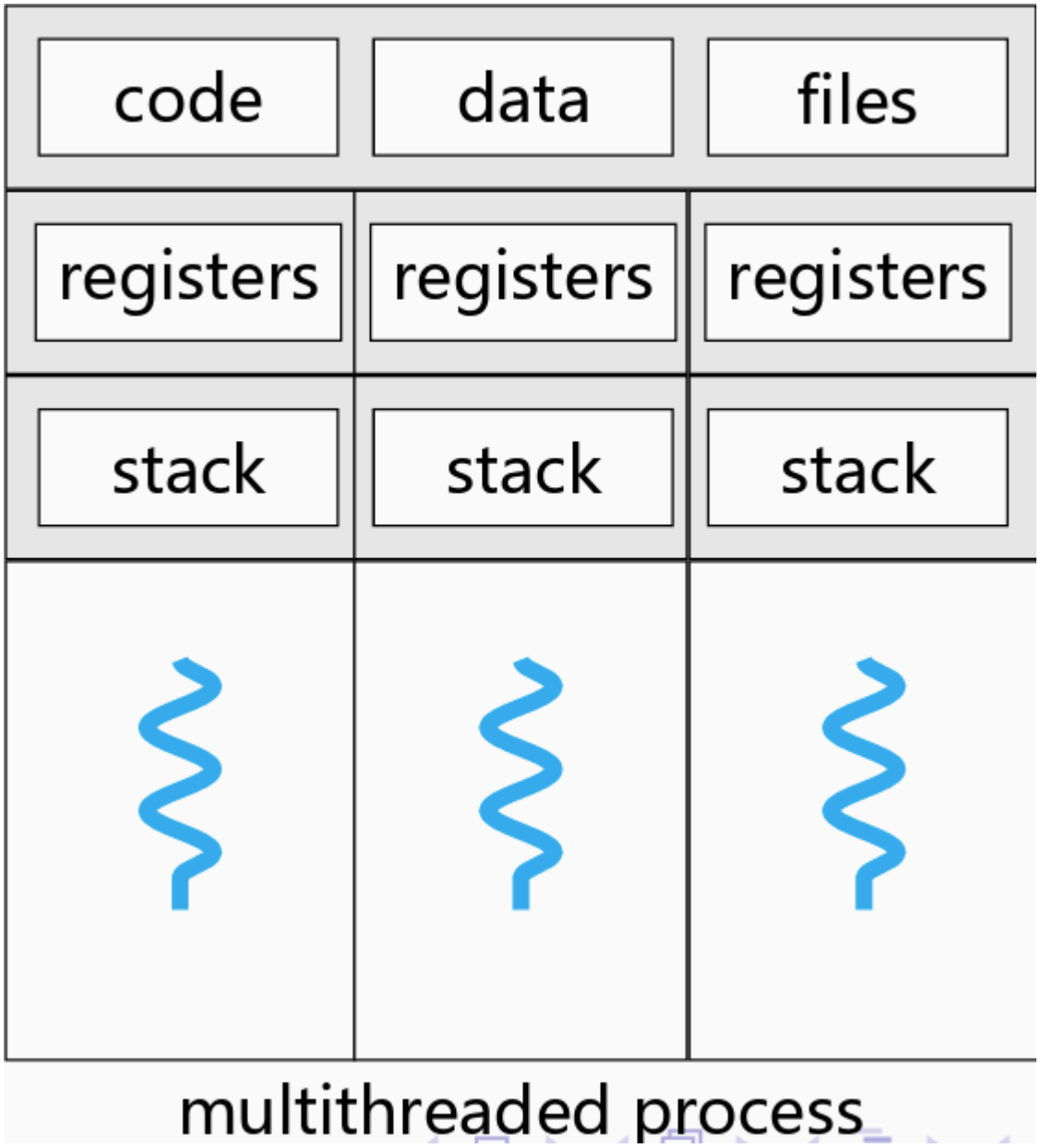
优先级反转问题：低优先级任务先来，占领资源，高优先级任务来了之后要等待资源释放，这时候来了中优先级任务，不需要资源，所以中优先级任务先于高优先级任务，两个解决方案

1. 优先级继承，高优先级等待低优先级释放资源的时候，把低优先级的优先级提高到和自己相同
2. 天花板策略，多个高优先级的任务准备申请资源的时候，低优先级的优先级在拥有资源的时候优先级是上述任务里最高的优先级。

调度算法的评估

1. 确定性建模
2. 排队模型
3. 模拟
4. 实现

threading



A thread is a basic unit of CPU utilization; ▶it comprises a threadID, a program counter,a register set, and a stack. ▶It shares with other threads belonging to the same process the code section,the data section,and other OS resources, such as open files,signals,etc

优点

1. 响应性
2. 资源共享
3. 经济
4. 缩放性

多线程模型

1. 多对一
2. 一对一
3. 多对多

process synchronization

竞争条件：当多个进程并发访问和操作同一数据并且执行结果取决于特定的访问顺序

临界资源：在一段时间内只允许一个进程访问的资源

临界区问题的解决方案

1. 互斥
2. 空闲让进
3. 有限等待

信号量

1. 整形信号量
    1. 计数形信号量
    2. 二进制信号量
    3. 同步工具
2. 记录形信号量（值正负的含义

生产，读者，背

deadlock

死锁产生的条件

1. 互斥（资源不能被同时访问
2. 持有并等待
3. 不剥夺
4. 循环等待

死锁的解决方案

1. 预防（请求时）和避免（分配时）
2. 检测和恢复（等待图，判断有没有死锁
3. 忽略

银行家算法

memory

内部碎片

外部碎片：总的可用内存之和可以满足请求但不连续

可重定位方案

动态分区的算法

1. 首次适应
2. 循环首次适应
3. 最佳适应
4. 最差适应

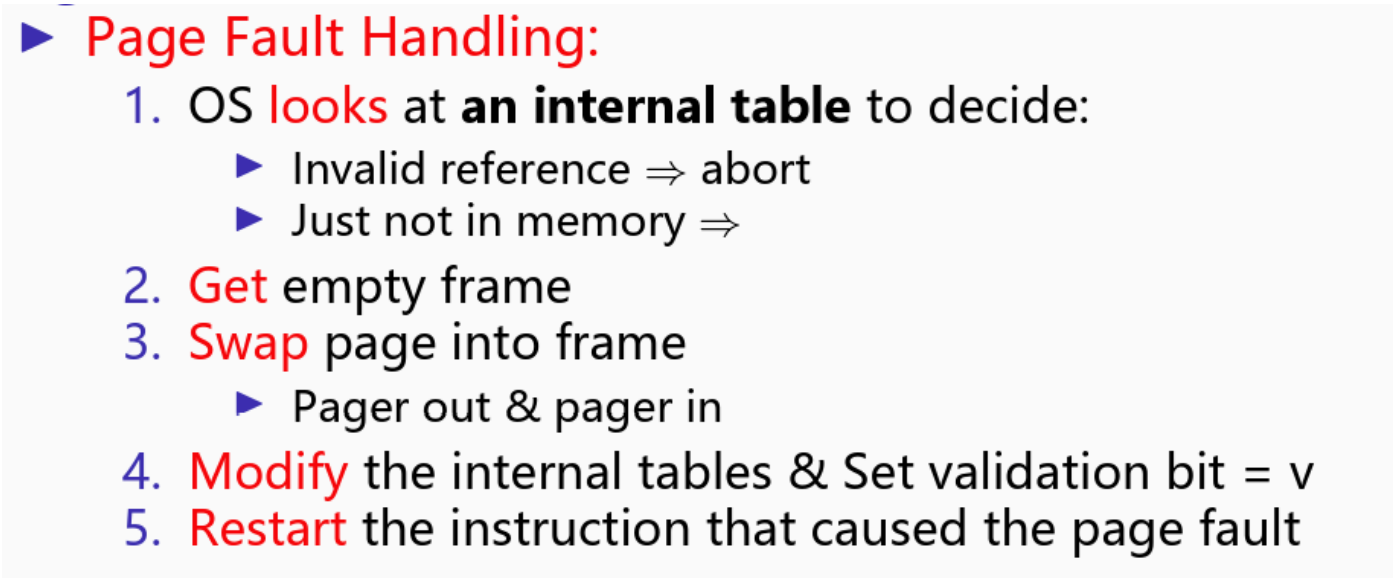
紧凑需要和动态可重定位技术结合，只能在运行时进行

对换技术和中期调度有关系

virtualmemory

虚拟存储器的特征

1. 多次性
2. 对换性
3. 虚拟性



- **To keep the fault time low**
  1. Swap space, faster then file system
  2. Only dirty page is swapped out, or
  3. Demand paging only from the swap space, or
  4. Initially demand paging from the file system, swap out to swap space, and all subsequent paging from swap space
- **Keep the fault rate extremely low**
  - Localization of program executing
    - Time, space

页面置换的算法

- Basic Page Replacement
  1. **Find** the location of the desired page on disk
  2. **Find** a free frame:
     - If there is a free frame, use it
     - If there is no free frame, use a page replacement algorithm to select a victim frame
  3. **Bring** the desired page into the (newly) free frame; **Update** the page and frame tables
  4. **Restart** the process

贝来蒂异常

分配给进程的最小页框取决于isa

帧的分配：

1. 平均分配
2. 比例分配

置换

1. 全局置换（允许拿别人的
2. 局部置换

抖动：高度的页面调度活动成为抖动，抖动产生的根本原因是不合理的多道程序度

工作集模型：

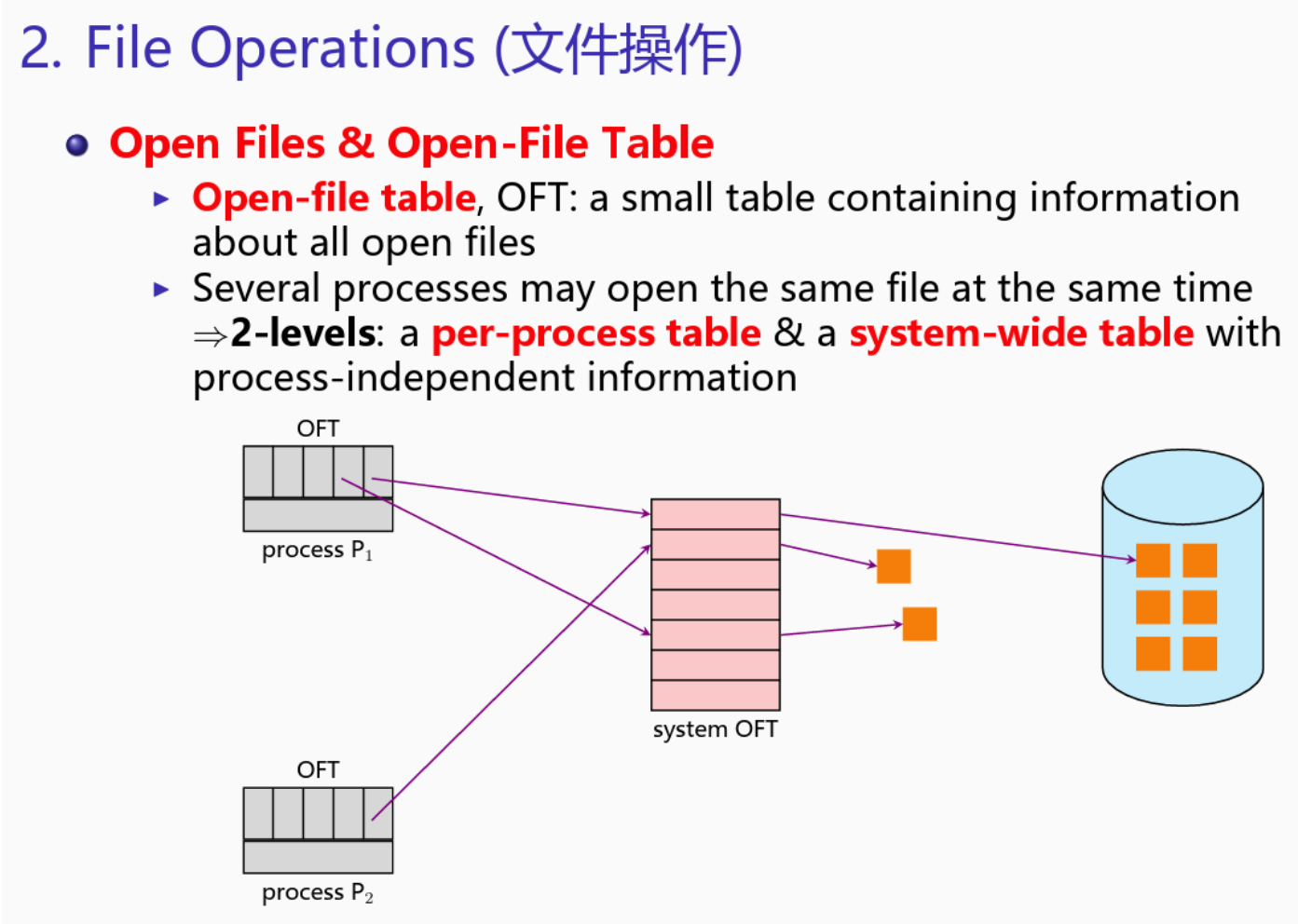
Memory-mapped file：讲磁盘文件的全部或部分内容与进程虚拟地址空间的某个区域建立映射关系，便可以直接访问被映射的文件，而不必执行文件的io操作也无需对文件内容进行缓存处理

页大小取决于

1. 碎片
2. 页表大小
3. Io时间
4. 局部性

tlb范围：通过tlb访问的内存量

file structure



OFT：open file table,记录了文件的io资源，两个opt，每个进程有perprocess table ，记录了与进程相关的文件信息，还有文件的公共信息存储在system-wide table

文件（逻辑上）结构

1. None
2. 简单
3. 复杂结构

文件访问方式

1. 顺序
2. 直接（定长便于计算
3. 索引、

硬链接，使用文件别名来链接，也就是无环图，增加一个路径名(删除的时候只删除路径，所有路径都删除文件删除)

符号链接，新建一个文件，存放别的文件的路径名（删除只删除链接，文件本身被删除保留链接

文件的保护

1. 可靠性（物理损坏
2. 安全性（隐私

FS implimentation

虚拟文件系统

1. 没有文件系统，虚拟出来一个文件系统
2. 为用户提供了文件系统操作的统一接口，屏蔽了不同文件系统的差异和操作细节

目录实现

1. 线性表
2. 哈希表

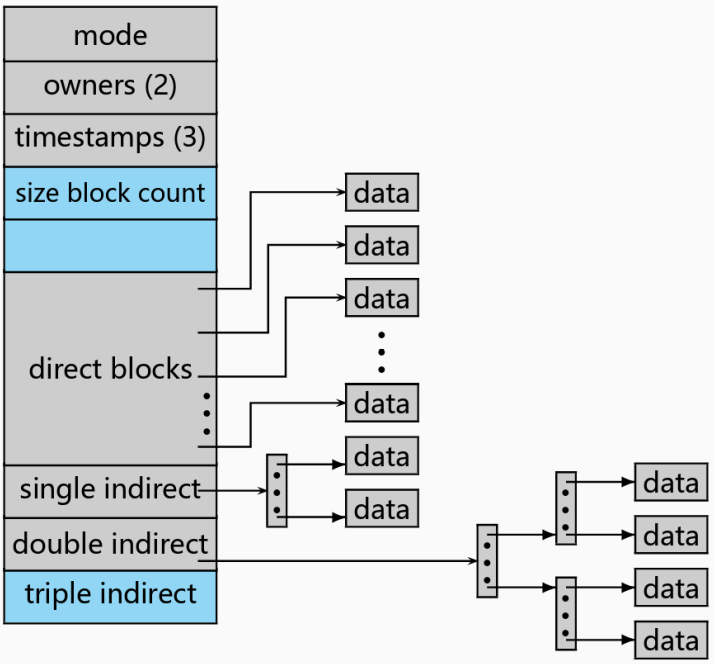
文件的分配方法

1. 连续分配
2. 连接分配
    1. 显式
    2. 隐式

3. 索引分配
4. 组合分配

FAT file allocation table

4. Combined Scheme (组合方式): UNIX (4K bytes per block) I



mass storage structure

磁盘的低级格式化：在磁盘可以存储数据之前它必须分成扇区以便磁盘控制器能够读写

磁盘调度算法

1. FCFS
2. SSTF
3. SCAN
4. C-SCAN
5. C-LOOK（不会到头

有坏块的解决方法

1. 人工方法：fat表中给坏块分配
2. 备用
3. 滑动，把坏块滑动过去

磁盘阵列RAID

1. raid 0-6

io

控制方式

1. 轮询（握手协议
2. 中断
3. DMA方式（握手协议

周期窃取：当DMA控制器占用内存总线时，cpy被暂时阻止访问内存，但是任然可以访问著缓存或者辅助缓存内的数据项

使用buffer的原因

1. 处理数据流的生产者和消费者之间的速度不匹配
2. 协调传输大小不一致数据的设备
3. 支持应用程序io复制语义

缓冲机制

1. 单缓冲
2. 双缓冲
3. 循环缓冲
4. 缓冲池（公用，前面都是私用

假脱机：外部设备联机并行操作