

# 操作系统

---

## 定义

---

- **计算机程序**
- 管理计算机系统的全部硬件资源、软件资源及数据资源
- 为用户提供方便的、有效的、友善的用户界面

## 目标

---

- **方便性**：方便用户使用计算机
- **有效性**：以有效的方式使用计算机资源
- **可扩展性**
  - 硬件升级
  - 新硬件
  - 新服务
  - 纠正错误

## 功能

---

- **作为用户和计算机间的接口**
  - 程序开发
  - 程序运行
  - I/O设备访问
  - 文件访问
  - 系统访问
  - 错误检测与响应
  - 审计
- **管理计算机硬件资源**
  - 为其他程序的运行分配硬件资源

## 发展

---

1. 串行处理系统
2. 简单批处理系统
  - 每次执行一个程序
  - 运行了监控程序
    - 顺序调度，每一批中有多个程序，监控程序调度这些程序执行
    - 提供使用程序和公用函数
3. 多道程序批处理系统
  - 当一个程序需要等待I/O操作时，让出CPU资源。
  - 实现了宏观上的并行，微观上的串行
  - 需要硬件支持

- I/O中断
- DMA

#### 4. 分时系统

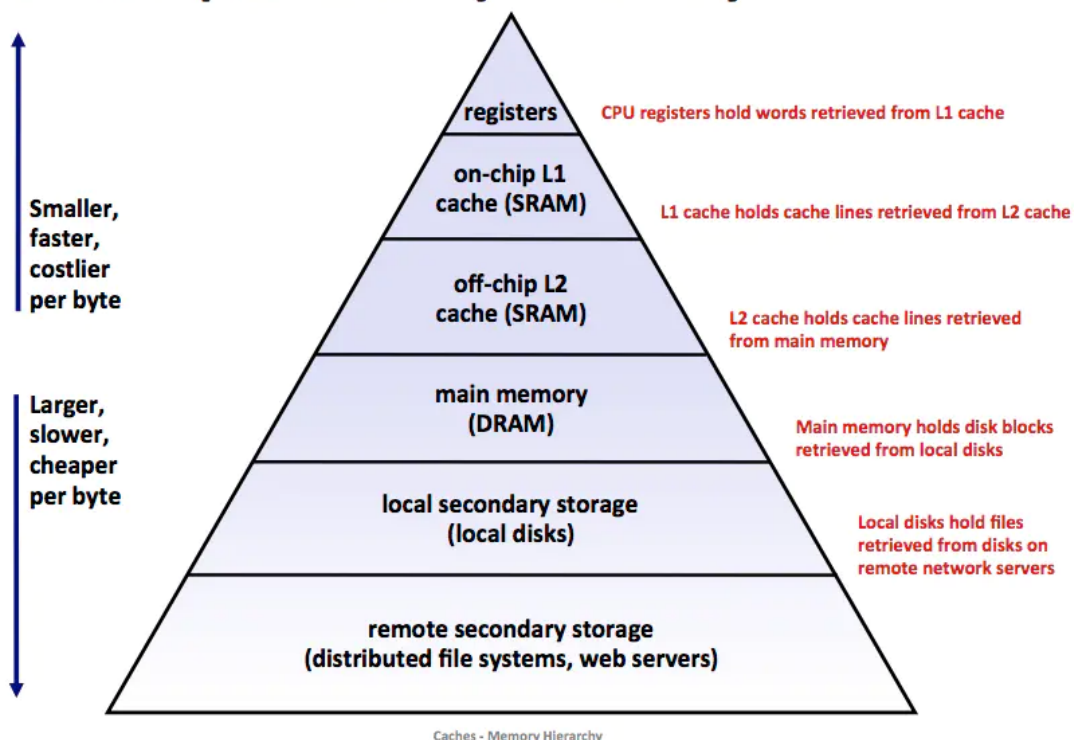
- 将CPU的时间划分成时间片，每个程序以时间片为单位执行，比如时间片为1微秒，那么每个程序都轮转执行1微秒。
- 目前发展的方向

## OS的主要成就

- 进程
- 内存管理
  - 虚拟内存
- 信息保护和安全
- 调度和资源管理
  - 公平性
  - 有差别的响应
  - 有效性：最大吞吐率，最小响应时间，最多服务用户
- 系统结构
  - 简单结构
  - 层次结构（单体内核）
    - 操作系统分为多层——》模块化，类比于OSI 7层协议
  - 微内核
- 其他
  - 多线程：一个进程可以分为多个线程
  - 对称多处理：多核处理

## 存储器的层级结构

### An Example Memory Hierarchy



- 局部性原理
  - 时间局部性：被引用过一次的内存地址很可能在不远的将来会再被多次引用。
  - 空间局部性：如果一个内存位置被引用了一次，那么程序很可能在不远的将来会引用附近的一个内存位置。
- 磁盘缓冲
  - 主存的一部分，用于临时存放磁盘数据
  - 由于从磁盘缓冲中读取数据的速度远远高于从磁盘中读取数据，结合局部性原理，可以提高程序运行速度
  - 磁盘操作以簇（Block）的方式进行
- 主存缓冲（cache）
  - CPU不直接与主存交换数据，而是与cache交换数据
  - CPU的速度高于主存的速度，cache的速度位于存储器的顶层
  - CPU与cache之间以字为单位交换数据，cache与主存以'簇'（block）为单位交换数据
  - 由于从cache中读取数据的速度高于从主存中读取数据，结合局部性原理（提前将活跃使用的数据存存放在cache中），可以提高程序运行速度
  - [参考](#)
  - 每个CPU对应一个cache

## I/O操作

---

- I/O模块负责具体的I/O操作，I/O模块中有I/O状态寄存器
- 可编程I/O（轮询）
  - 等待型I/O
  - CPU不断检查I/O模块的状态，当I/O模块ready后CPU才可以进行下一步的读写操作
- 中断驱动I/O
  - 当I/O模块ready后，发出中断，CPU开始参与下一步的读写操作
- 直接存储器访问
  - 上述两种CPU参与数据的读写过程，直接存储器不需要CPU的参与，交给DMA去进行读写操作
  - 下面是DMA操作的步骤
    1. CPU初始化DMA请求，为DMA分配数据总线
    2. DMA负责处理I/O请求，CPU继续工作（其他的事情）
    3. DMA完成后发出中断，CPU处理中断