

操作系统是什么？

操作系统是计算机系统中的—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**地使用计算机，使整个计算机系统高效率运行

这就是操作系统的一个定义 那么操作系统完成各种各样的功能



解读几个关键词

有效：系统效率，资源利用率

CPU利用率充足与否？**I/O**设备是否忙碌？

合理：

各种软硬件资源的管理是否公平合理



如果不公平、不合理，则可能会产生问题？

方便使用：

两种角度：用户界面 与 编程接口

两个典型的接口是 一个是用户界面，也就是通过命令行，用户和操作系统打交道



解读几个关键词

有效：系统效率，资源利用率

CPU利用率充足与否？**I/O**设备是否忙碌？

合理：

各种软硬件资源的管理是否公平合理

Q 如果不公平、不合理，则可能会产生问题？

方便使用：

两种角度，用户界面 与 编程接口

还有一个呢是编程接口，也就是程序员通过开发软件，在软件中与操作系统打交道



操作系统的三个作用

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



我们总结出典型的操作系统的三个作用 第一个作用，操作系统是资源的管理者

OS是资源的管理者

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、网卡等)

软件资源:

磁盘上的文件、各类管理信息等

有很多很多的管理信息也是操作系统所管理的 所以操作系统是资源的管理者。



怎样管理资源？

- ◎ 跟踪记录资源的使用状况



一个软件有数据结构、 算法 这些成分，所以数据结构是用于记录资源的使用情况

怎样管理资源？

- ◎ 跟踪记录资源的使用状况
 - 如：哪些资源空闲，分配给谁使用，允许使用多长时间等
- ◎ 确定资源分配策略——算法



有了数据结构之后 我们还需要算法，也就是说，操作系统要确定资源分配策略

怎样管理资源？

- ◎ 跟踪记录资源的使用状况
 - 如：哪些资源空闲，分配给谁使用，允许使用多长时间等
- ◎ 确定资源分配策略——算法

给它用多久，给它用 用多少，这都是所谓资源分配策略



怎样管理资源？

- ◎ 跟踪记录资源的使用状况
 - 如：哪些资源空闲，分配给谁使用，允许使用多长时间等
- ◎ 确定资源分配策略——算法
 - 静态分配策略



所谓静态分配策略就是说一个程序在执行前 就把它所需要的各种资源拿到手

怎样管理资源？

- ◎ 跟踪记录资源的使用状况
 - 如：哪些资源空闲，分配给谁使用，允许使用多长时间等
- ◎ 确定资源分配策略——算法
 - 静态分配策略



那么这个程序在执行过程中就不再需要新的资源了 就可以一直执行完成。

怎样管理资源？

- ◎ 跟踪记录资源的使用状况
 - 如：哪些资源空闲，分配给谁使用，允许使用多长时间等
- ◎ 确定资源分配策略——算法
 - 静态分配策略



当然这里头有一个问题 这个资源在执行过程中有的时候不需要用资源的时候呢，
这个资源就被
浪费了。

怎样管理资源？

- ◎ 跟踪记录资源的使用状况
 - 如：哪些资源空闲，分配给谁使用，允许使用多长时间等
- ◎ 确定资源分配策略——算法
 - 静态分配策略
 - 动态分配策略
- ◎ 实施资源的分配和回收



因此，很多的操作系统，或者说绝大多数操作系统都采用的是一个动态分配策略

怎样管理资源？

- ◎ 跟踪记录资源的使用状况
 - 如：哪些资源空闲，分配给谁使用，允许使用多长时间等
- ◎ 确定资源分配策略——算法
 - 静态分配策略
 - 动态分配策略
- ◎ 实施资源的分配和回收



那么这样的好处就是，资源的利用率得到了充分的提高 充分的提高。

怎样管理资源？

- ◎ 跟踪记录资源的使用状况
 - 如：哪些资源空闲，分配给谁使用，允许使用多长时间等
- ◎ 确定资源分配策略——算法
 - 静态分配策略
 - 动态分配策略
- ◎ 实施资源的分配和回收
- ◎ 提高资源利用率
- ◎ 保护资源的使用
- ◎ 协调多个进程对资源请求的冲突

这就是操作系统怎样管理一个资源的一个一般性的过程



从资源管理的角度 — 五大基本功能

- ◎ 进程/线程管理 (CPU管理)

进程线程状态、控制、同步互斥、通信、调度、.....

- ◎ 存储管理

分配/回收、地址转换、存储保护、内存扩充、.....

- ◎ 文件管理

文件目录、文件操作、磁盘空间、文件存取控制、.....

- ◎ 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

操作系统的第二个作用呢 是向用户提供各种服务。



OS是各种系统服务的提供者

- 在操作系统之上，从用户角度来看：

操作系统为用户提供了一组功能强大、方便易用的命令或系统调用

- 典型的服务

进程的创建、执行；文件和目录的操作；I/O设备的使用；各类统计信息；.....

设备的使用啊，还有提供各种各样的统计信息
这里简单地举了几个例子，大家有那么一个感性认识



OS是硬件之上的第一层软件

应用程序

----- 虚拟机器界面

操作系统

----- 物理机器界面

硬件

执行，那么这是我们说操作系统是硬件基础上的第一层软件



与硬件相关的工作复杂、繁琐

◎ 请问：下列哪一个操作更简单、方便？

“从某个文件读一个数据块” 与 “移动磁头、等待放下”

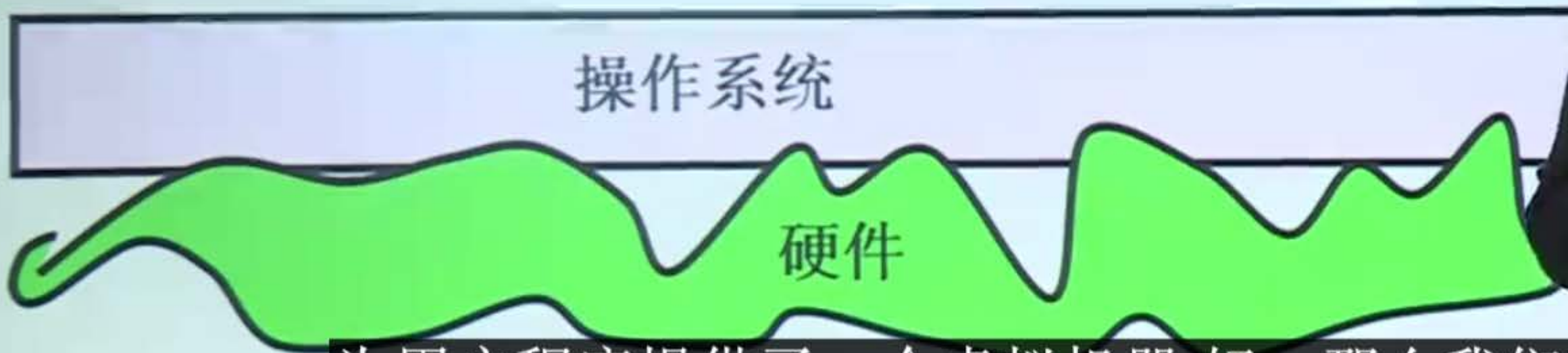
◎ 例子：软盘I/O操作

- 控制芯片NEC PD765，有16条命令
- 每一条命令向设备控制寄存器写入长度从1 到9 字节的特定数据（读写数据、移动磁头臂、格式化磁道，及初始化、检测状态、复位、标准控制器及设备等等）
- 以READ为例：13个参数
要读取的磁盘块地址、磁道的扇区数、物理介质的记录格式、扇区间隙、对已删除数据地址标识的处理方法
- 操作结束时，控制器芯片在7个字节中返回23个状态及出错字段
- 程序员还要保持注意步进电机的开关状态



OS对硬件机器的扩展

- ◎ 操作系统在应用程序与硬件之间建立了一个等价的扩展机器（虚拟机）
- ◎ 对硬件抽象，提高可移植性；比底层硬件更容易编程



为用户程序提供了一个虚拟机器 好，那么我们最后再



明确了操作系统的三个作用

- 有效地管理资源
- 通过命令接口、编程接口等为用户提供各种功能服务
- 对硬件机器的扩展

第三个作用呢，就是屏蔽硬件细节，为用户提供一个扩展的虚拟机器



下面我们来介绍一下操作系统的定义及作用 那么操作系统是什么？怎样理解它的作用？操作系统是什么呢？操作系统是计算机系统中的一个系统软件 这个系统软件呢是一些程序模块儿的结合 这些程序模块儿相互协作、相互配合来完成以下这些功能 首先，它们能以尽量有效、合理的方式组织和管理计算机的 各种软硬件资源。为了 支持程序的执行，操作系统 要合理地组织计算机的工作流程，控制程序的执行 并在程序执行的过程中 向程序提供各种各样的服务功能。第三 操作系统必须采用各种各样的方式使得用户能够非常灵活、方便地使用计算机 采用各种各样的技术使整个的计算机系统能够高效率地运行 这就是操作系统的一个定义 那么操作系统完成各种各样的功能 我们解读一下儿在这个定义当中有些关键的 关键的词汇 第一个词是有效 那么操作系统有管理各种各样的资源 那么这些资源像 CPU，那我们总是希望 CPU 不空闲，能够充分利用 CPU，有很多很多的程序在 CPU 上跑 那么还有内存、磁盘，我们希望这些内存、磁盘呢 应该能够充分利用这些空间，能够得到很好的管理 比如说各种各样的设备，我们希望这些设备能够充分地忙碌起来 总而言之，我们希望通过 操作系统能够把各种各样的资源的利用率提高 以提高整个的操作系统，整个的计算机系统的效率 第二个关键词是合理 因为要管理资源，所以呢各种各样的策略就会有各种各样的策略。那么这些管理资源的这种策略是不是公平，是不是合理呢？那么如果不公平、不合理，则会产生什么样的问题呢？我们呢也要解决这样的问题。第三个关键词是方便使用 我们要和操作系统打交道 通常希望能够非常容易地使用 操作系统提供给我们的各种接口。两个典型的接口是一个 是用户界面，也就是通过命令行，用户和操作系统打交道 还有一个呢是编程接口，也就是程序员 通过开发软件，在软件中与操作系统打交道 那么这两种接口都希望能够容易使用、容易学习 操作系统的作用是什么？我们总结出典型的操作系统的三个作用 第一个作用，操作系统是资源的管理者 操作系统是资源的管理者是从什么角度来看呢？是从自底向上，也就是从操作系统内部的角度来看 操作系统是资源的管理者。哪些是操作系统该管的资源呢？这都是我们熟悉的硬件资源和软件资源 CPU, 内存 各种各样的设备，包括我们说的 I/O 设备，打印机啊 扫描仪啊，磁盘、时钟 还有像网卡等等。那么还有 很多软件资源，比如说磁盘上有很多很多的文件 有很多很多的管理信息也是操作系统所管理的 所以操作系统是资源的管理者。从这样一个角度出发 我们来看一下怎么去管理各种各样的资源 操作系统为了管理资源，首先需要跟踪记录每一种资源的使用状况 那么这就是通过数据结构 一个软件有数据结构、算法 这些成分，

所以数据结构是用于记录资源的使用情况 那么一个数据结构都记录哪些信息呢？我们可以记录哪些资源是空闲的 哪些资源已经分配出去了，分配给谁使用了 并且允许使用多长时间等等这些信息 有了数据结构之后 我们还需要算法，也就是说，操作系统要确定资源分配策略 操作系统要决定把资源分给哪一个进程 用它用多久，给它用多少，这都是所谓资源分配策略 那么资源分配策略通常呢有几个典型的 一个是静态分配策略。所谓静态分配策略就是说一个程序在执行前 就把它所需要的各种资源拿到手 那么这个程序在执行过程中就不再需要新的资源了 就可以一直执行完成。当然这里头有一个问题 这个资源在执行过程中有的时候不需要用资源的时候呢，这个资源就被 浪费了。因为它拿着资源，别的进程不能用 因此，静态分配策略会导致整个资源的浪费 因此，很多的操作系统，或者说绝大多数操作系统都采用的是一个动态分配策略 那么动态分配策略就是说在一个 进程执行的过程中，什么时候需要资源 什么时候提出申请。那么操作系统就会根据目前系统中资源的 使用状况，也就是刚才所说的数据结构 来决定这个资源是否分配给它，是否分配给它 那么这样的好处就是，资源的利用率得到了充分的提高 充分的提高。我们举一个典型的例子 比如说我们是一个页式的存储管理系统 那么内存中有，我们把内存，物理内存划分出很多的物理页面 那我们可以用一个非常简单的数据结构位示图，或者位图 bitmap 来记录这个物理内存的使用情况 一位对应了物理内存的一页 如果这一页是空闲的，那么我们这里头就记为 0 如果这一位是分出去了，那我们就记为 1 那么我们的分配算法就实际上就是在这个数据结构上来执行 那么下面就要去实施资源的分配和回收 那么分配的过程实际上就是 按照策略，按照刚才制定的这种资源分配策略选中 要给分配资源的这个进程，然后去决定分配给它多少 然后去修改相应的数据结构。比如说刚才，那么如果我们把一页 内存分配给了一个进程，那么这个位示图就会把这个原来的 0 改成 1 而回收，当一个进程使用完资源之后，那么就要回收，系统要把它回收 那么回收的时候呢，也是要修改相应的数据结构，比如说就把原来的 1 换成 0 这就是实施资源分配的和回收的过程 除此之外，操作系统 还要采用各种各样的技术来提高资源的利用率 当有多个进程都要使用资源的时候呢，还要保护资源 那么一个进程使用资源的过程不被另外一个进程所干扰 如果在一个系统中有多个进程 都对资源有申请，那么产生了冲突的情况下，需要操作系统来协调 这就是操作系统怎样管理一个资源的一个一般性的过程 那么我们以后所有的资源其实

都符合 这样的一般性的过程，当然有一些特殊的问题需要解决 从资源管理的角度 那么我们来看，操作系统呢总结出操作系统的五大基本功能 进程/线程管理，有的时候也称为 CPU 管理，因为它们进程和线程是在 CPU 上执行 这里包括了进程线程的状态设计、控制操作 同步互斥、通信还有调度 存储管理主要是做 存储空间的分配和回收 完成地址的转换，还有存储保护、内存扩充 文件管理，那么当然是 主要包括了文件目录管理，文件的各种各样的操作 磁盘空间的管理，还有文件的存取控制管理 设备管理呢，主要在设备的驱动程序 设备的分配、回收过程 还有为了提高设备的各种各样的资源利用率呢采用的各种技术，比如说缓冲技术等等 那么一个操作系统在使用过程中，必然提供 给用户非常方便使用的接口，因此呢，作为操作系统呢我们要讨论 接口的，典型的接口，系统命令，还有编程接口 操作系统的第二个作用呢 是向用户提供各种服务。那么这些服务呢是方便用户使用的 那么我们可以说 操作系统是各种系统服务的一个提供者 那么这个系统服务呢，如果操作系统 提供功能更强大，更容易使用的这样的一系列的命令或系统调用 我们就认为这个操作系统提供了更好的服务。那有哪些典型的服务呢？我们来看一下，典型的像进程的创建呀 执行啊，文件和目录的各种操作呀 各种 I/O 设备的使用啊，还有提供各种各样的统计信息 这里简单地举了几个例子，大家有那么一个感性认识 操作系统的第三个作用呢是对机器硬件的一个扩展 具有一个扩展能力。我们来看一个示意。那么我们在这里头我们知道操作系统 实际上是硬件基础上的第一层软件 那么操作系统完成各种各样的功能 在操作系统之上是应用程序在执行 操作系统为应用程序的执行提供了，像我们说的 各种各样的接口，用户的接口和编程的接口 那么这些就组成了一个虚拟机器界面。操作系统之下呢 是硬件，也就是操作系统屏蔽了一些硬件的细节 用户程序不需要直接和硬件打交道 通常它都是通过操作系统来控制这些硬件的 执行，那么这是我们说操作系统是硬件基础上的第一层软件 那么为什么要把，用操作系统把硬件屏蔽呢？那我们来看一下与硬件相关的一些工作 是非常复杂和繁琐的。我们有两个操作 我们现在介绍两个操作，那么想问一下，这两个操作当中的哪一个操作 你觉得简单，你觉得很方便 第一个操作呢是说 从一个文件当中读一个数据块 第二个操作呢是说你要移动磁头，要等待这个磁头放下 那么我们对程序员来讲，第一个操作 显然要比第二个操作更简单，更方便使用 那我们来看看如果你要考虑第二个操作，你要实现第二个操作 那你要考虑哪些问题，是不是很复杂，是

不是很琐碎？我们举一个非常简单的例子，软盘 I/O 操作 虽然软盘呢我们现在都很难见到了 但是因为软盘和硬盘很相似，但是它又非常简单，所以我们就举 它作为例子。那么硬盘比它要复杂得多，所以我们更多的问题需要去解决 有这么一个非常简单的软盘控制器 那么这个软盘控制器呢可以执行 16 条命令 这 16 条命令呢实际上的长度是不一样的，是从 1 到 9 个字节的，长度不等 而这些命令包括了什么呢？我们简单来看一下，比如说读写数据 然后移动磁头臂，格式化磁道 还有些初始化呀，检测软盘的状态呀，复位呀等等等等 那么这些呢都是非常细节的一些命令，我们很难记住 那么这些命令呢要把它写入到设备的控制寄存器里头去来控制这个设备的执行 我们以这个 read 命令为例，read 命令呢 需要 13 个参数，那我们还要记住 13 个参数 每个参数呢，像 read 命令的 13 个参数呢包括了你要读取的磁盘地址 磁盘的扇区数，磁道的扇区数，还有很多跟 物理介质相关的一些物理特性，软盘的一些物理特性，你都要了解 因此对于程序员来讲就很不容易来知道每一个细节 另外呢，这个软盘 工作的是不是正常，是不是完成了你的这种控制要求？因此这个软盘的这个控制芯片呢还会返回给 23 个状态或者出错信息，这些 信息呢 又记录在了 7 个字节的设备 状态寄存器里头。那么还要了解这 23 个不同的状态，什么样的情况，发生了什么事情 除此之外，还有一个很特殊的 就是程序员还要保持步进电机的一个开关状态 那么我们知道软盘不能总是在 启动的状态，那么时间久了，软盘就会损坏 因此呢，如果长时间不用，它就会停止 但是一旦要用，读数据了，要写数据了，要启动这个软盘 让它旋转到一定的转速那才能去读 因此呢，程序员要注意这样一些步进开关的这种 步进电机一个开关的一些状态，还要考虑这些细节 从以上介绍其实我们已经看出来了，说如果让我们来 编程的时候，考虑这么琐碎、复杂的事情，那么程序员呢会非常非常头疼 所以我们说与硬件相关的这些工作特别地琐碎 特别地繁杂，而且不同的硬件呢 这些还不一样。如果我们换了一个 驱动器的，驱动器，那么这个命令就可能是另外的样子，每条命令就不一样，参数也是不一样的 因此，操作系统就做了这样一个事情 操作系统就是在硬件之上 在应用程序与硬件之间 建立了这么一个等价的一个扩展机器，虚拟机 虚拟机，那么通过对硬件的抽象 那么提高了整个系统的可移植性 那么对于程序员来讲呢，他去对底层硬件编程 更麻烦，而提供了这样一个操作系统这一层之后，那么 程序员比对这个编程就会比底层硬件的这种更容易 那么我们举一个例子来 描述一下这个图

所表达的含义 我们通常我们买，可以买一套房子 那么这套房子呢我们是一个毛坯房，毛坯房，那么你去看房子的時候一看这个墙 非常地难看，凸凹不平，都是那个砖嘛，凸凹不平 那怎么办呢？那这时候呢我们就需要装修 那么装修当然就是说先用腻子把墙抹平，然后再刷上 几遍漆，这样的话你就可以住进非常温馨的家里去了 但是这个装修一般来讲，不可能是说每家每户自己来装修 通常在我们这儿，都是有个装修队，有一个装修公司来帮助你来装修 那么这就是说装修公司起的作用就像操作系统起的作用一样 那么它把这些比较琐碎、复杂的一些硬件屏蔽在它之下 然后向用户提供了一个非常良好的，非常优雅的、清晰的 一致的这样一个接口，那么供用户去使用 那么我们说操作系统是对硬件机器的一个扩展 为用户程序提供了一个虚拟机器 好，那么我们最后再 考虑这个操作系统的三个作用，再强调一下 第一个作用，就是有效地管理计算机系统的各种资源 第二个作用呢，就是通过接口，命令接口 编程接口，为用户提供各种各样的服务功能 第三个作用呢，就是屏蔽硬件细节，为用户提供一个 扩展的虚拟机器