

8、操作系统发展史

1. 手工操作（无操作系统）
2. 批处理系统
 - 2.1 联机批处理系统
 - 2.2 脱机批处理系统
- 3 多道程序系统
 - 3.1 多道程序技术
 - 3.2 多道批处理系统
4. 分时系统
5. 实时系统
6. 操作系统功能小结
7. 通用操作系统
8. UNIX操作系统
 - 8.1 诞生
 - 8.2 体系结构
 - 8.3 用户界面
 - 8.4 文件系统
 - 8.5 软件开发环境

8、操作系统发展史

我们这门课研究的主题是：基于平台的软件开发。前面我们介绍过了第一种真正意义上的编程平台：CPU指令集。从现在开始，我们讲介绍最重要的编程平台：操作系统。

现在，我们买来电脑的后第一件事情就是安装操作系统，有的电脑则在我们买来的时候已经安装好了操作系统，比如说品牌机（Dell、HP、lenovo等等）。我们（最终用户）可以在操作系统上通过安装程序来看视频，听音乐，玩游戏、浏览网页，还可以弹出移动硬盘和U 盘、连接打印机打印、管理我们硬盘中的文件等等。

可是，操作系统并非时随着计算机同步出现，一开始计算机只有“光板”硬件，没有预装任何软件，程序直接调用CPU指令集。

下面我们来了解一下，是什么原因导致了操作系统的出现和发展、操作系统的那些功能是怎么来的。

1. 手工操作（无操作系统）

人们先把程序纸带装上计算机，然后启动**输入机**把程序和送入计算机，接着通过**控制台开关**启动程序运行。计算完毕，打印机输出计算结果，用户卸下并取走纸带（或卡片）。第二个用户上机，重复同样的步骤。如下图1所示：

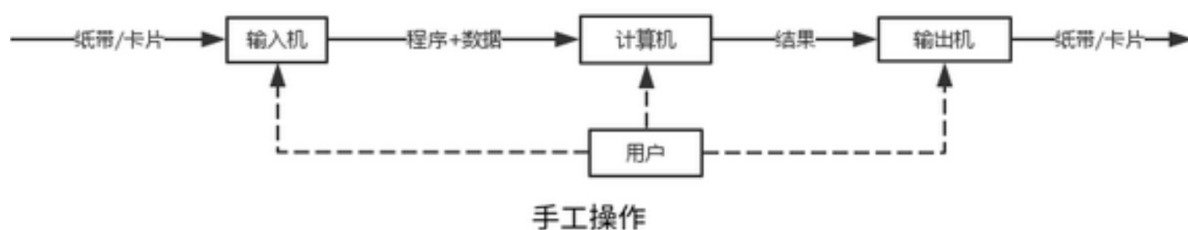


图1、手工操作计算机

特点：用户独占机器，通过控制台开关操作，CPU等待手工操作，CPU利用不充分。

由于手工操作的满速度和计算机的高速度之间形成了尖锐矛盾，手工操作的方式是计算机的资源利用率极低，唯一的解决办法只有摆脱手工操作，实现作业的自动过渡。

2. 批处理系统

批处理系统：加载计算机上的一个监督软件，在监督程序的控制下，计算机能够自动的、成批的处理一个或多个用户的作业（作业包括程序、数据、命令）。

首先出现的是联机批处理系统，即作业的输入输出由CPU来处理。

2.1 联机批处理系统

在主机和输入机之间增加两个存储设备——磁带机，在监督程序的自动控制下，计算机自动完成任务。

如下图2所示，成批的把输入机上的用户作业读入磁带，依次把磁带上的用户作业读入主机内存并执行，执行完成后把计算结果向输出机输出。完成一批作业后，**监督程序**又从输入机读取作业存入磁带机。按照上面的步骤重复处理任务。**监督程序**不停的处理各个作业，实现了作业的自动转接，减少了作业的建立时间和手工操作时间，有效的克服了人机矛盾，提高了计算机资源的利用率。

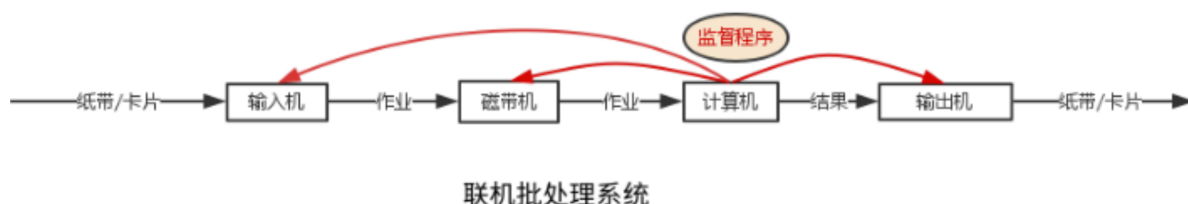


图2、联机批处理系统

问题：在输入作业和输出结果时，CPU还是会处于空闲状态，等待慢速的输入/输出设备完成工作——主机处于忙等状态。

2.2 脱机批处理系统

为了克服与缓解告诉主机与慢速外设（输入输出设备），提高CPU利用率，用又引入了脱机批处理系统，即输入输出脱离主机控制。如下图3所示。



图3、脱机批处理系统

显著特征就是：增加一台不与主机直接相连**卫星机**。卫星机用来从输入机上读取用户作业并放到磁带机上；将磁带机上的执行结果传给输出机。这样主机不再与慢速的输入输出设备连接。主机与卫星机两者并行工作，分工明确，可充分发挥主机的高速计算能力。

问题：每次主机内存中仅存放一道作业，每当它运行期间发出输入/输出（I/O）请求后，高速的CPU便处于等待低速的I/O完成状态，致使CPU空闲。

3 多道程序系统

3.1 多道程序技术

多道程序设计技术，就是指**允许多个程序同时进入内存并运行**。即同时把多个程序放入内存，并允许它们交替在CPU中运行，它们共享系统中的各种硬、软件资源。当一道程序因I/O请求而暂停运行时，CPU便立即转去运行另一道程序。多道程序与单道程序对比参见下图4：

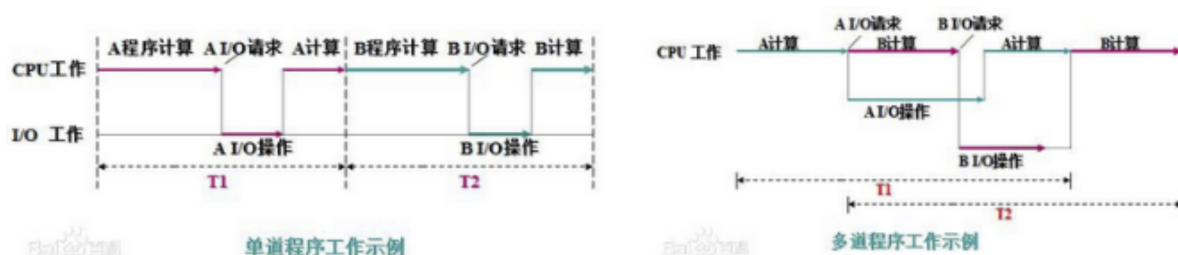


图4、单道程序与多道程序

单道程序运行过程：在A程序计算时，I/O空闲，A程序I/O操作时，CPU空闲（B程序也是同样）；必须A工作完成后，B才能进入内存中开始工作，两者是串行的，全部完成共需时间= T_1+T_2 。

多道程序运行过程：将A、B两道程序同时存放在内存中，它们在系统的控制下，可相互穿插、交替地在CPU上运行：当A程序因请求I/O操作而放弃CPU时，B程序就可占用CPU运行，这样CPU不再空闲，而正进行A I/O操作的I/O设备也不空闲，显然，CPU和I/O设备都处于“忙”状态，大大提高了资源的利用率，从而也提高了系统的效率，A、B全部完成所需时间 $<T_1+T_2$ 。

多道程序设计技术不仅使CPU得到充分利用，同时改善I/O设备和内存的利用率，从而提高了整个系统的资源利用率和系统吞吐量（单位时间内处理作业（程序）的个数），最终提高了整个系统的效率。

3.2 多道批处理系统

20世纪60年代中期，在前述的批处理系统中，引入多道程序设计技术后形成**多道批处理系统**。

多道：系统内可同时容纳多个作业。这些作业放在外存中，组成一个后备队列，系统按一定的调度原则每次从后备作业队列中选取一个或多个作业进入内存运行，运行作业结束、退出运行和后备作业进入运行均由系统自动实现，从而在系统中形成一个自动转接的、连续的作业流。

成批：在系统运行过程中，不允许用户与其作业发生交互作用，即：作业一旦进入系统，用户就不能直接干预其作业的运行。批处理系统的追求目标：提高系统资源利用率和系统吞吐量，以及作业流程的自动化。批处理系统的一个重要缺点：不提供人机交互能力，给用户使用计算机带来不便。

多道批处理系统的一个重要缺点：**不提供人机交互能力**，给用户使用计算机带来不便。虽然用户独占全机资源，并且直接控制程序的运行，可以随时了解程序运行情况。但这种工作方式因独占全机造成资源效率极低。

4. 分时系统

如下图5所示，**分时系统**是把处理机的运行时间分成很短的时间片，按时间片轮流把处理机分配给各联机作业使用。若某个作业在分配给它的时间片内不能完成其计算，则该作业暂时中断，把处理机让给另一作业使用，等待下一轮时再继续其运行。由于计算机速度很快，作业运行轮转得很快，给每个用户的印象是，好象他独占了一台计算机。而每个用户可以通过自己的终端向系统发出各种操作控制命令，在充分的人机交互情况下，完成作业的运行。

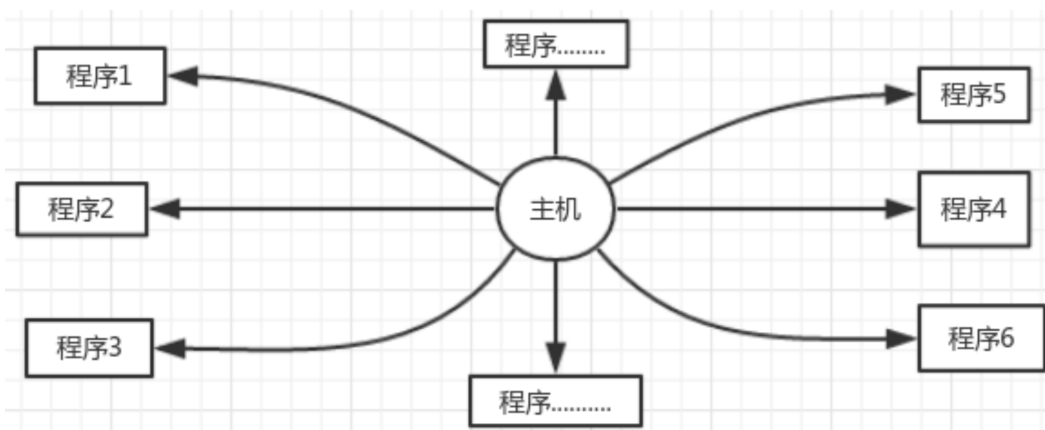


图5、分时系统

分时系统总结：

特点：(1) 同时性。若干个终端用户可同时使用计算机；(2) 独立性。用户彼此独立，互不干扰；(3) 及时性。用户的请求能在较短时间内得到响应；(4) 交互性。用户能进行人-机对话，联机地调试程序，以交互方式工作。

适用性：交互性强的多用户系统。

问题：无法对特殊任务做出及时响应。

5. 实时系统

虽然多道批处理系统和分时系统能获得较令人满意的资源利用率和系统响应时间，但却不能满足**实时控制与实时信息处理**两个应用领域的需求。于是就产生了实时系统，**即系统能够及时响应随机发生的外部事件**，并在严格的时间范围内完成对该事件的处理。实时系统可分成两类：

实时控制系统：当用于飞机飞行、导弹发射等的自动控制时，要求计算机能尽快处理测量系统测得的数据，及时地对飞机或导弹进行控制，或将有关信息通过显示终端提供给决策人员。当用于轧钢、石化等工业生产过程控制时，也要求计算机能及时处理由各类传感器送来的数据，然后控制相应的执行机构。

实时信息处理系统：当用于预定飞机票、查询有关航班、航线、票价等事宜时，或当用于银行系统、情报检索系统时，或者个人用户进行文档编辑等办公操作时，都要求计算机能对终端设备发来的服务请求及时予以正确的回答。此类对响应及时性的要求稍弱于第一类。

实时系统总结：

特点：具有实时性，**事件驱动**。

适用性：较少有人为干预的监督和控制系统，例如，监督产品线，控制流水线生产的连续过程，监督病人的临界功能，监督和控制交通灯系统、监督和抑制实验室的实验以及监督军用飞机的状态等等。

6. 操作系统功能小结

如上面内容所示，操作系统发展出三种基本类型：多道批处理系统、分时系统、实时系统。总结抽象一下：

多道批处理系统的功能是：尽量提高CPU的利用率，批量处理多个预定的计算任务。

分时系统的功能是：进一步提高CPU的利用率，同时完成多个用户的任务，包括人机交互任务。

实时系统的功能是：让系统能够实时处理由随机的外部事件触发的任务。

再进一步抽象，操作系统的功能是：

管理控制计算机的资源（CPU、内存等），并合理地组织调度计算机的工作和资源分配。

7. 通用操作系统

通用操作系统：具有多种类型操作特征的操作系统，可以同时兼有多道批处理、分时、实时处理的功能，或其中两种以上的功能。

例如：实时处理+批处理=实时批处理系统。首先保证优先处理实时任务，插空进行批处理作业。常把实时任务称为前台作业，批作业称为后台作业。

再如：分时处理+批处理=分时批处理系统。即：时间要求不强的作业放入“后台”（批处理）处理，需频繁交互的作业在“前台”（分时）处理，处理机优先运行“前台”作业。

从上世纪60年代中期，国际上开始研制一些大型的通用操作系统。这些系统试图达到功能齐全、可适应各种应用范围和操作方式变化多端的环境的目标。但是，这些系统过于复杂和庞大，不仅付出了巨大的代价，且在解决其可靠性、可维护性和可理解性方面都遇到很大的困难。

早期的通用操作系统非常多样化，生产商生产出针对各自硬件的操作系统。每一个操作系统都有很不同的命令模式、操作过程和调试工具，即使它们来自同一个生产商。最能反映这一状况的是，厂家每生产一台新的机器都会配备一套新的操作系统。

8. UNIX操作系统

8.1 诞生

早期的通用操作系统很多，这些系统试图达到功能齐全、可适应各种应用范围和操作方式变化多端的环境的目标。但是，这些系统过于复杂和庞大，不仅付出了巨大的代价，且在解决其可靠性、可维护性和可理解性方面都遇到很大的困难。

相比之下，UNIX操作系统却是一个例外，这是一个最早出现并获得成功的、通用的、多用户分时交互型的操作系统。它首先建立的是一个精干的核心，而其功能却足以与许多大型的操作系统相媲美。在核心层以外，可以支持庞大的软件系统。它很快得到应用和推广，并不断完善，对现代操作系统有着重大的影响。下图6展示了UNIX系统的产生过程：

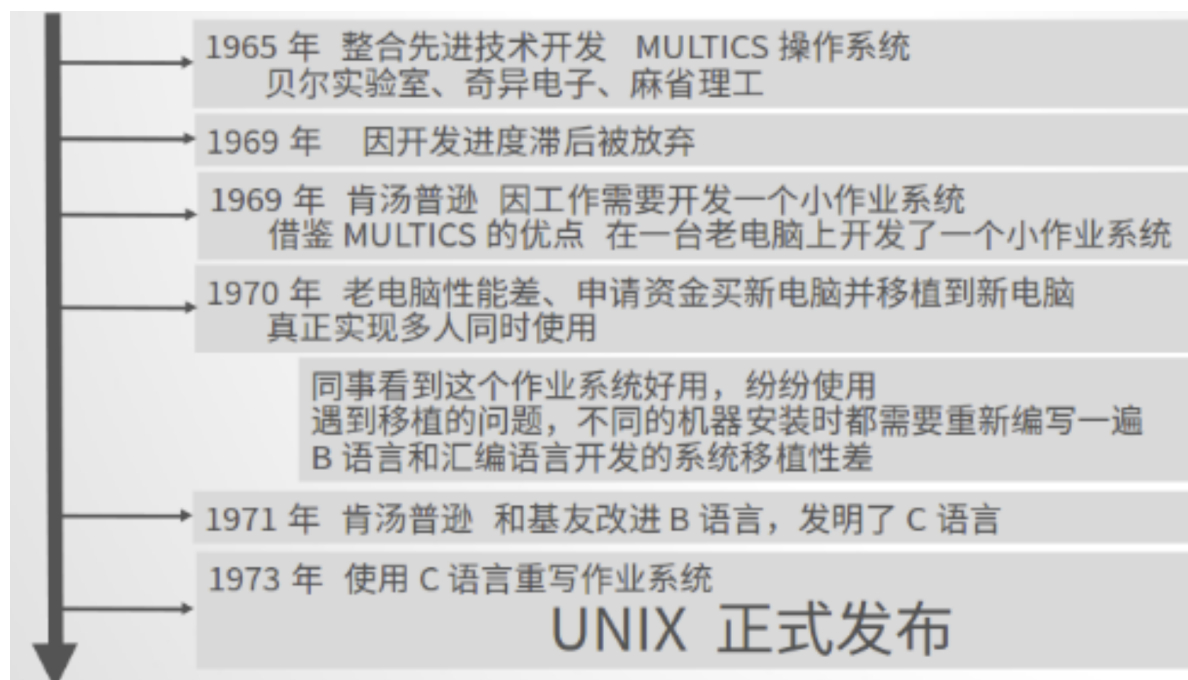


图6、UNIX的诞生

8.2 体系结构

UNIX系统在计算机操作系统的发展史上占有重要的地位。它确实对已有技术不断作了精细、谨慎而有选择的继承和改造。并且在**操作系统的总体设计构想**等方面有所发展，才使它获得如此大的成功。

UNIX系统在结构上分为核心程序(kernel)和外围程序(shell)两部分，而且两者有机结合成为一个整体。核心部分承担系统内部的各个模块的功能，即处理机和进程管理、存储管理、设备管理和文件系统。核心程序的特点是精心设计、简洁精干，只需占用很小的空间而常驻内存，以保证系统的高效率运行。外围部分包括系统的用户界面、系统实用程序以及应用程序，用户通过外围程序使用计算机。参见下图7：



图7、Unix体系结构

UNIX这种分层设计奠定了现代操作系统体系结构设计的基础，对软件开发也产生了深远的影响。

8.3 用户界面

UNIX系统提供了良好的用户界面，具有使用方便、功能齐全、清晰而灵活、易于扩充和修改等特点。UNIX系统的使用有两种形式：

一种是操作命令，即shell语言，它是一个命令解析器，类似于DOS命令。它接收用户命令，然后调用相应的应用程序。是用户通过终端与系统发生交互作用的命令行界面。

另一种是面向用户程序的界面，它不仅在汇编语言，而且在C语言中向用户提供服务，即API服务。我们基于操作系统进行软件开发，就是调用操作系统提供的这个API。

8.4 文件系统

UNIX系统的文件系统是树形结构。它由基本文件系统和若干个可装卸的子文件系统组成，既能扩大文件存储空间，又有利于安全和保密。

8.5 软件开发环境

UNIX系统包含有非常丰富的语言处理程序、实用程序和开发软件用的工具性软件，向用户提供了相当完备的软件开发环境。

注：这个东西很重要！操作系统只有向程序员提供方便强大的软件开发环境，才会吸引更多的程序员为这个操作系统开发软件，这个操作系统才会更有用，才会吸引更多用户使用它。

