1. 作业管理和用户接口
2. 作业的组织和管理
3. 作业

作业是用户在一次事务请求过程中要求计算机系统所做工作的综合，它是用户向系统提交工作的基本单位。

1. 作业的分类

根据系统作业处理方式的不同，分为：

1. 脱机作业

用户不能直接与计算机交互，中间需要操作员控制和干预，用于批处理系统。

1. 联机作业

用户直接与计算机交互，用于单用户微机操作系统。

1. 作业的组成

作业由程序，数据以及作业控制信息组成。当系统调用该作业时，调用命令解释程序，对作业说明书中的命令逐行解释执行

1. 作业的处理过程

一个作业从进入系统到执行结束，都要经过“输入”，“后备”，“执行”和“完成”四个阶段，因此作业有四个状态。

1. 作业的输入输出方式
2. 联机输入输出

由主机直接控制输入输出，由于主机和外设速度相差过大，会降低CPU利用率。

1. 脱机输入输出

专门设置一台外围处理机将作业线输入到辅助存储器中，然后与主机相连。

1. SPOOLING系统

在多道程序系统中，用程序模拟脱机输入输出时外围控制机的功能，便可以在主机的直接控制下实现脱机输入输出的功能，外设的I/O操作与CPU对数据的处理同时进行，这种在联机情况下实现的外设同时操作称为SPOOLONG，也叫伪脱机。

核心是利用磁盘来模拟独占设备的操作，是一台设备变为多台可并行使用的虚拟设备。SPOOLING技术是在通道技术和多道程序设计基础上产生的，由主机和相应的通道共同完成作业的输入输出。SPOOLING系统由专门负责I/O的常驻内存的进程以及输入井，输出井组成。输入井和输出井是在磁盘中的两大存储空间，输入井模拟脱机输入时的磁盘，暂存I/O设备输入的数据；输出井模拟脱机输出的磁盘，暂存用户程序的输出数据，作用是调节CPU和IO设备的速度差异。输入缓冲区和输出缓冲区是在内存中的，输入缓冲区用于暂存从输入设备送来的数据，再传给输入井；输出缓冲区用于暂存从输出井送来的数据，再送给输出设备。输入进程和输出进程是系统进程，优先级高于用户进程。

1. SPOOLING优势

提高了IO速度，对于低速IO设备，演变为了对输入井和输出井数据的读取；

将独占设备改造为共享设备，其实并没有为任何作业分配设备，只是在输入井和输出井中分配了一个存储区和IO请求表；

实现了虚拟设备功能，每个线程都认为自己独占了一个设备，其实只是逻辑上的而已；

1. 作业控制块
2. 概述

每个作业进入系统时，系统会自动为其建立作业控制块（Job Control Block，JCB），用来存放管理和控制作业所需的信息。当作业退出系统时，JCB才撤销，因此JCB是一个作业存在的唯一标志。

1. 作业后备队列

作业建立完成后，形成一个后备队列，存放在输入井中。

1. 作业调度

一个作业从提交到完成会经过高级，中级和低级三级调度。

1. 高级调度（作业调度）

选择一些后备状态的作业，为其建立进程，并进入主机。

1. 中级调度（对换调度）

负责决定进程在内存和辅助盘交互区间的对换，在内存紧张时，为了将进程调入内存，需要将内存中的一些进程切换到盘交换区以腾出空间，中级调度是为了缓解内存资源的紧张。

1. 低级调度（进程级调度）

决定内存中的哪个进程可以占据CPU，使其处于运行状态。

1. 作业调度算法的评价标砖

作业调度程序通常作为一个进程在系统中执行，在系统初始化时被创建，调度算法是调度的关键，评价标准是能否使得系统资源得到高效的利用，如下的参数都是评价标准：

1. CPU利用率

CPU利用率越高越好。

1. 吞吐量

单位时间CPU完成作业数量。

1. 周转时间

周转时间 = 等待时间 + 作业运行时间

周转系数 = 周转时间 / 运行时间

周转时间和周转系数越低越好。

1. 作业调度算法
2. 先来先服务（FCFS）

谁等的时间越长，越先得到CPU，有利于长作业。

1. 短作业优先调度（SJF）

短作业占据作业中一大部分，因此SJF会降低作业的平均等待时间，提高系统的吞吐量，但是也会造成长作业的饥饿现象。

1. 最高响应比优先调度（HRP）

为每个作业设置一个优先级，公式如下：

优先级 = （等待时间 + 作业执行时间）/ 作业执行时间

这样，长作业在等待一段时间后，一定能够得到CPU，该公式又相当于响应比，既利于短作业，又克服了饥饿现象。

1. 作业控制方式
2. 脱机作业控制

作业控制语言（JCL）是表达作业控制意图和步骤的语言。

1. 联机作业控制

操作系统提供了交互式的用户接口，用户通过控制终端输入命令，向系统提出要求，命令解释器会解释这些命令，完成操作。例如UNIX中的Shell。

1. 系统功能调用
2. 概述

系统功能调用是操作系统提供给开发者的程序级接口，开发者通过它来调用系统功能。

1. 程序状态

程序分为系统程序和用户程序，同时把操作系统程序运行的状态称为管态或系统态，把用户程序运行的状态称为算态或用户态。

1. 特权指令

特权指令指的是只能在管态下执行而不能在算态下执行的特殊指令。如传送程序状态字指令，启动外设指令等。

1. 访管指令

它不是特权指令，作用是引起访管中断，硬件开始响应中断，保护原来的程序状态字到内存固定单元，然后取出新的程序状态字送入寄存器中，由于新的程序状态字已经预设为管态，从而CPU进入管态，执行用户请求的功能，中断处理程序完成后，恢复原来的程序状态字，回到用户程序的算态。

综上，系统功能调用就是用户在程序中用访管指令调用操作系统指令。

1. 系统调用和普通过程调用的区别

系统调用是特殊的过程调用，区别如下：

1. 系统调用从算态切换到管态，再切回到算态；
2. 系统调用通过软中断进入；
3. 系统调用在采用了抢占调度方式的系统中，在被调用过程执行完后，会进行优先权的分析，只有得到了CPU才能返回原调用进程接着执行，而一般的过程调用则是直接返回调用进程。