这部分主要是计算机基础知识的概念介绍，相关系列文章如下所示。

操作系统时计算机系统的核心系统软件，其他软件均建立在其上，其分类包括：单用户操作系统和批处理操作系统、分时操作系统和实时操作系统、网络操作系统和分布式操作系统、嵌入式操作系统。其4大特征为并发性、共享性、虚拟性和不确定性，此外操作系统的5大功能为：进程管理、文件管理、存储管理、设备管理和作业管理。

**处理机与进程管理**

进程是资源分配和独立运行的基本单位，是程序的一次执行，该程序可以和其他程序并发执行，通常有PCB进行管理。进程的状态转化与控制如下图所示。



进程同步的相关概念

进程间同步：异步环境下的一组并发进程间互发消息、相互合作、相互等待，使得各进程按一定的速度执行的过程称为进程同步。

进程间互斥：一组并发进程中的一个或多个程序段，因共享某一共有资源而使得他们必须以一个不允许交叉的顺序执行。

临界资源：系统中有些资源可以供多个进程同时使用，有些资源一次只能供一个进程使用，称为临界资源。

临界区：是进程中对临界资源实施操作的那段程序。

信号量机制与PV操作。

进程的调度和死锁

进程调度即处理器调度（又称上下文切换），它的主要功能时确定把处理器在什么时候分配给哪一个进程。

调度方式：指当有更高优先级进程到来时如何分配CPU，分为可剥夺和不可剥夺2种。

调度算法：常用的有先来先服务、时间片轮转、优先级调度和多级反馈调度算法。

死锁：指两个以上的进程相互都因请求对方已经占有的资源，无限期的等待并无法执行下去的情况。

死锁产生的原因是资源竞争及进程推进顺序非法。产生死锁的4个必要条件是互斥条件、请求保持条件、不可剥夺条件和环路条件。解决死锁的策略包括：鸵鸟策略、预防策略、避免策略和检测与解除死锁策略，最后一种最为常见。

**存储管理**：存储组织的功能时在存储技术和CPU寻址技术许可的范围内组织合理的存储结构，使得各层次的存储器都处于均衡的状态，其依据是访问速度匹配、容量要求和价格等。一般存储器结构有“寄存器—缓存—主存—外存”。其相关概念还包括：逻辑地址、物理地址、存储空间和地址重定向。存储管理的功能有主存储器的分配和回收、提高主存储器的利用率、存储保护、主存扩充。存储管理的方式有分区存储管理，分页存储管理、分段存储管理、段页式存储管理和虚拟存储管理。

**设备管理**:其任务是保证在多道程序环境下，当多个进程竞争使用设备时，按一定策略分配和管理各种设备，控制设备的各种操作，完成输入/输出设备和主存之间的数据交换。设备管理的主要技

术包括：中断技术、DMA技术、缓冲技术、虚设备与SPOOLING技术。此外，传统的磁盘调度算法有：先来先服务、最短寻道时间优先、扫描算法等。

**文件管理**:文件的访问方法是指对读写文件存储设备上的一个物理快的方法，常用的有顺序访问和随机访问。此外，文件系统对空闲空间的管理方法包括位图法、索引法和链接法。

数据库是指长期存储在计算机内的、有组织的、可共享的数据集合。数据库系统从广义上来讲是由数据库、硬件、软件和人员组成，管理的对象是数据。其采用三级模式、两级映像（模式/内模式映像，外模式/模式映像）来保证数据的逻辑和物理独立性。

实体-联系(E-R)模型，其中E-R图的主要构件如下表所示。

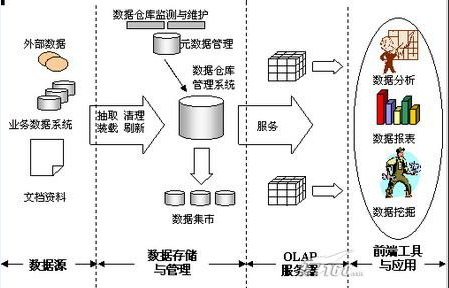
|  |  |
| --- | --- |
| 构件 | 说明 |
| 矩形 | 表示实体集 |
| 菱形 | 表示联系集 |
| 椭圆 | 表示属性 |
| 线段 | 将属性与相关的实体集连接，或将实体机与联系集相连 |
| 双椭圆 | 表示多值属性 |
| 虚椭圆 | 表示派生属性 |
| 双线 | 表示一个实体全部参与到联系集中 |

数据规范化：关系模式中个属性之间的依赖关系及其对关系模式性能的影响。关系型数据库设计理论的核心是数据间的函数依赖，衡量的标准是关系规范化的程度及分解的无损连接和保持函数依赖性。数据依赖是通过一个关系中属性间值的相等与否体现出来的数据间的相互关系，是现实世界属性间联系和约束的抽象，是数据内在的性质，是语义的体现。函数依赖则是一种最重要、最基本的数据依赖。包括函数依赖、非平凡的函数依赖、平凡函数依赖、完全函数依赖、部分函数依赖、传递依赖、码、主属性和非主属性、外码、值依赖定义、函数依赖的公理系统。

范式是关系模型满足的确定约束条件，范式有1NF、2NF、3NF、BCNF（巴克斯范式）、4NF和5NF。

此外，数据库的ADID属性和并发控制比较复杂，这儿滤过，请见<http://www.cnblogs.com/wanliwang01/p/TSQL_Base04.html>。

BI商业智能：企业对商业数据的搜集、管理和分析的系统过程，目的是使企业的各级决策者获得知识或洞察力，帮助他们做出对企业更有利的决策。它是数据仓库、联机分析处理(OLAP)和数据挖掘等相关技术走向商业应用后形成的一种应用技术。其包括数据预处理、建立数据仓库、数据分析和数据展现4个主要阶段。



计算机网络可以分为：广域网（WAN），局域网(LAN)和城域网(MAN)。

开放系统互连的参考模型(OSI/RM)由低层到高层分别为物理层、数据链路层、网络层、传输层、会话层、表示层和应用层。

网络互联的设备根据其工作的协议层分类为：中继器、网桥、路由器、网关和交换机等。传输介质包括:双绞线、同轴电缆和光纤灯；无线介质为微波、红外线和激光等。

TCP/IP协议簇包括多种不同的协议，如下表所示。

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ISO/OSI模型 | TCP/IP协议 | | | | | TCP/IP模型 |
| 应用层 | 文件传输协议FTP | 远程登陆协议SMTP | 电子邮件协议SMTP | 网络文件服务协议NFS | 网络管理协议SNMP | 应用层 |
| 表示层 |
| 会话层 |
| 传输层 | TCP,UDP | | | | | 传输层 |
| 网络层 | IP | ICMP | ARP,RARP | | | 网际层 |
| 数据链路层 | Enternet IEEE | FDDI | Token-Ring/IEEE 802.3 | ARCnet | PPP/SLIP | 网络接口层 |
| 物理层 | 硬件层 |

**负载均衡技术**：由多台服务器以对称的方式组成一个服务器集合，每台服务器都具有等价的地位，都可以单独对外提供服务而无须其他服务器的辅助。其实现方式包括2中：把大量的并发访问或数据流量分配到多个设备上分别处理，以减少用户等待的时间；将单个重负载的运算分摊到多个设备上做并行处理，再将每个设备的运行结果汇总后返回给用户。

一个网络的负载均衡，一般情况下从传输链路聚合、采用更高层网络交换技术和设置服务器集群策略三个角度来实现。常用的负载均衡技术有操作系统自带的负载均衡服务、基于特定服务器软件的负载均衡、基于DNS的负载均衡、反向代理负载均衡、基于NAT的负载均衡等。

**服务器集群技术**：由一组相互独立的服务器在网络中表现为单一的系统，并以单一系统模式进行管理，为客户提供可靠服务，并大幅提高服务器的安全性。一个Cluster包含多台拥有共享数据存储空间的服务器，任何一台服务器在运行一个应用时，应用数据被存储在共享的数据空间中。每台服务器的IS和应用程序文件存储在本地存储空间。集群中的计算机使用同一个名称，通过内部局域网通讯，当一台故障时，其应用程序将被自动接管。

媒体是承载信息的载体，其有两层含义，分别是：信息的载体，如文本、图像；存储信息的实体，如磁盘、U盘等。媒体的种类主要包括如下5种。

感觉媒体：人们接触信息的感觉形式，如视觉、听觉、触觉和味觉。

表示媒体：信息的表现形式，如文字、图像、动画、音频、视频。

显示媒体：表现和获取信息的物理设备，如输入的键盘、鼠标，输出的显示器、打印机等。

存储媒体：存储数据的物理设备，如磁盘、光盘、内存等。

传输媒体：传输数据的物理载体，如电光缆和交换设备。

国际标准：静态图像压缩编码的标准JPEG；运动图像压缩标准MPEG和H261等。

多媒体的研究内容：数据压缩、数据的组织和管理、多媒体信息的展现和交互、多媒体通信与分布处理、虚拟现实技术、智能多媒体技术等。

常见软件开发模型：瀑布模型、原型模型、螺旋模型、基于可重用构建的模型、基于面向对象的模型、基于第四代技术的模型。

接下来，对现在非常流行的敏捷开发做一个简单介绍。其特点为：适应性和预设性，面向人而非面向过程；核心思想：敏捷方法是适应性，而非可预测性；以人为本，而非以过程为本；迭代增量式的开发过程；4个核心价值观：沟通、简单、反馈和勇气；12条过程事件规则：简单设计、测试驱动、代码重构、结对编程、持续集成、现场客户、发行版本小型化、系统隐喻、代码集体所有制、规划策略、规范代码、40小时工作机制；主要敏捷方法:XP极限编程，开放式代码(Git)，SCRUM

Tip:需要技术团队有一定的实力。

**需求管理：**软件需求开发的最终文档经过评审批准后，则定义了开发工作的需求基线(baseline)，需求管理的整体流程如下图所示。



需求管理的目标：为软件需求建立一个基线，提供给软件工程和管理使用；软件计划、产品和活动与软件需求保持一致。

**开发管理**

1. 项目的范围、时间、成本

在初步项目范围说明书中已文档化的主要交付物、假设和约束条件的基础上准备详细的项目范围说明书，是项目成功的关键。此外，范围定义的输入包括，项目章程、项目范围管理计划、组织过程资产、批准的变更申请。

项目时间管理中的过程包括：活动定义、活动排序、活动的资源估算、活动历时估算、订制进度计划以及进度控制。

成本管理包括：成本估算，编制一个为完成项目各活动所需资源成本的近似估算；成本预算，将总的成本估算分配到各项活动和工作包上，来建立一个成本的基线；成本控制，控制项目预算的变更。

1. 配置管理、文档管理

产品配置是指一个产品在其生命周期各个阶段所产生的各种形式和各种版本的文档、计算机程序、部件及数据的集合。该集合中的每一个元素称为该产品配置中的一个配置项(Configuration Item)，配置项主要包括两类：属于产品组成部分的工作成果，如需求文档、设计文档、源代码和测试用例等；属于项目管理和机构支撑过程域产生的文档，如工作计划、项目质量报告、项目追踪报告等。

1. 软件开发的质量和风险

软件质量：成功的项目管理是在约定的时间和范围、预算的成本以及要求的质量下，达到项目干系人的期望。ISO9000对项目质量的定义是：一组固有特性满足需求的程度，需求指明示的、通常隐含的活必须履行的需求和期望。

软件开发风险：项目风险既包括对项目目标的威胁，也包括促进项目目标的机会。风险源于所有项目之中的不确定性因素。

**常见设计方法**：结构化分析与设计、面向对象的分析设计，后者的设计过程如下图所示。



最后是软件的重用、逆向工程与重构工程，这部分内容在现在高节奏迭代的场景下的应用实在太少，一笔带过了。

提到企业信息系统，最著名的莫过于ERP系统了， 企业运作的供需链图如下。

**ERP**系统主要包括以下模块：生产预测；销售管理；经营计划；主生产计划；物料需求计划；能力需求计划；车间作业计划；采购与库存管理；质量与设备管理；财务管理；ERP有关扩展应用模块。其功能包括：支持决策的功能；为处于不同行业的企业提供针对性的IT解决方案；从企业内部的供应链发展为全行业和跨行业的供应链。

**客户关系管理**CRM(Customer Relationshio Management)是涵盖构建良好客户关系所应具备所有要素的一门科学。主要包括4个内容：提供的信息要有利于更好地理解客户；流程管理要为客户提供高效、适当的体验；允许员工使用以上知识的软件；培训并改变管理要素，使员工和企业了解并有能力提供哪些构建强有力管理、提高客户忠诚度的体验。其目的是提高企业收入，核心思想是以客户为中心。

CRM的主要模块包括：销售自动化；营销自动化；客户服务和支持；商业智能。成功实施CRM的关键：分析与规范；项目计划与管理；系统配置和修改；原型、兼容测试和系统重复运行；主导系统和质量保证测试；最后实施和推广；持续支持；系统的持续管理。其价值在于：提高内部员工的工作效率，节省日常开支；提高客户的满意度；提高客户的忠诚度。

信息安全的相关技术繁多，其关键技术如下表所示。

|  |  |
| --- | --- |
| 技术 | 诠释 |
| 加密解密技术 | 对电子数据的攻击方式分为主动和被动两种形式：被动攻击，例如截获信息；主动攻击包括，中断、篡改和伪造。  对称密码体系的常见加密算法为DES(56位密钥)、IDEA(126位密钥)等，使用分组加密方式。  非对称加密体系常见的算法为RSA，其依赖于大素数分解，有耗时较长的缺点。 |
| 散列函数和数字签名 | 常见的散列函数包括MD5、SHA、HMAC，其中MD5最为常见，被称为信息摘要算法，对输入的任意长度消息产生128位的散列值。  数字签名可以解决否认、伪造、篡改等问题，常见的签名算法包括RSA和DSA、椭圆曲线等。  数字水印的主要应用领域包括：版权保护、加指纹、标题与注释、篡改提示和使用控制。 |
| 密钥分配中心与公钥基础设施 | 现代的密码体系中，算法本身的保密已经不重要，对数据的保密主要依赖于对密钥的保密，其相关的基础包括密钥分配中心（KDC）、数字证书和公钥基础设施(PKI)。 |
| 访问控制 | 访问控制是建立在身份认证的基础之上的，常见的身份认证技术包括：用户名与口令认证、令牌认证、生物识别和3因素(用户知道什么？口令、拥有什么？私钥和令牌、是什么？生物特征)认证。  访问控制分类：入网访问控制、网络权限控制、目录级安全控制、属性安全控制和网络服务器的安全控制。 |
| 安全协议 | 常见协议包括：IPSec、SSL、PGP协议。  IPSec:在**IP层**对数据包进行高强度的安全处理提供数据源验证、无连接数据完整性、数据机密性、抗重播和有限通信流机密性等安全服务。  SSL：是在传输层实现的一种安全协议，非常常用，采用公开密钥和私有密钥两种加密方式。  PGP:针对电子邮件在Internet上通信的安全问题而设计的一种混合加密系统。 |
| 数据备份 | 常见的备份类型包括：完全备份、差异备份、增量备份和按需备份。 |
| 计算机病毒和免疫 | 计算病毒的分类：引导区病毒、文件感染病毒、宏病毒、特洛伊木马和蠕虫病毒，其基础特征：感染性、潜伏性、可触发性、破坏性、人为性、衍生性。常见免疫方式包括：针对一个病毒进行的计算机病毒免疫；基于自我完整性检查的计算机病毒的免疫方法。 |

Tip:信息安全管理相干知识

安全管理包括：安全设备管理、安全策略管理、安全风险控制和安全审计等。主要解决：集中化的安全策略管理、实时安全监控、安全联动机制、配置与补丁管理和统一权限管理。

公安部制定的信息安全制度：计算机信息系统安全保护等级划分标准；计算机信息安全保护条例；计算机病毒防治管理办法。

信息安全保障体系的建立包括：建立统一的身份认证体系；建立统一的信息安全管理体系了；建立规范的信息安全保密体系(统一密钥管理)；建立完善的网络边界防护体系(不同的域)。

如何在现有的.NET项目中这么做，减少重复代码。

使用面向方面概念的原因：代码分散现象；关于代码分散的分析；一个模块化的新因素；横切功能的综合；非功能的服务和方面；依赖性的颠倒。

AOP技术可以说是OOP的补充和完善，其利用一种称为“横切”的技术，剖揭开封装的对象内部，并将那些影响了多个类的公共行为分装到一个可重用模块，称为Aspect方面。AOP的实现技术，主要包括两类：一种采用动态代理技术，利用截取信息的方法，对消息进行装饰，以取代原有对象行为的执行；二是采用静态织入的方式，引入特定的语法创建“方面”，从而使得编译期间织入有关“方面”的代码。

1. join point（连接点）:是程序执行中的一个精准执行点，例如类中的一个方法，它是一个抽象的概念，在实现AOP时，并不需要去定义一个join point。
2. point cut（切入点）:本质上是一个捕获连接点的结构。
3. advice（通知）:是point cut的执行代码，是执行方面的具体逻辑。
4. aspect（方面）:point cut和advice结合起来就是aspect，它类似于OOP中定义的一个类，但它代表的更多的是对象间横向的关系。
5. introduce（引入）：对对象引入附加的方法或属性，从而达到修改对象结构的目的，也称为mixing。

AOP的特性为：可扩展性、可重用性、易理解性和易维护性。

常见的AOP实现包括：AspectJ、AspectWerkz、JBoss AOP和Spring AOP，之后将尤其重点介绍Spring AOP。其中AspectJ可能是已知最好的AOP实现，其是一个语言规范，从Java语言中扩展开来（类似Scala?），AspectJ编译器生成的是符合Java字节码规范的class文件。而Spring AOP使用起来更加的方便便捷，spring大体的组织结构如下图所示。



Spring AOP用纯Java实现，不需要特别的编译过程，区别于AspectJ的实现，SpringAOP不需要控制类装载器，因此适用于J2EE Web容器或应用服务器。特别指出，Spring默认使用JDK动态代理实现AOP代理。这使得任何接口或接口的集合能够被代理。Spring也可以是CGLIB代理，如果业务对象没有实现一个接口，CGLIB将被默认使用。接下来介绍Spring AOP松散耦合的几种方式：创建通知；在Spring中定义切入点；用ProxyFactoryBean创建AOP代理；自动代理。

嵌入式系统时一种以应用为中心，以计算机技术为基础，可以适应不同应用，对功能、可靠性、成本、提及和功耗等方面的要求，集可配置可剪裁的软、硬件为一体的专用计算机系统，其特点是：系统专用性强、实时性强、软硬件依赖性强、处理器专用、多种技术紧密结合、系统透明、资源有限。其内核主要包含3个公共部件：调度器、内核对象和内核服务。



常见的内核调度算法包括：任务优先级分配方法、时间片轮转调度、任务操作。比较有名的嵌入式系统为VxWorks，其包括板级支持包BSP，微内核Wind、网络系统、文件系统和I/O系统。