

1. 为什么多处理机系统能显著提高计算机性能?

广义上说,使用多台计算机协同工作来完成所要求的任务的计算机系统都是多处理机系统。

传统的狭义多处理机系统的作用是利用系统内的多个 CPU 来并行执行用户的几个程序,以提高系统的吞吐量或用来进行冗余操作以提高系统的可靠性。多个处理机(器)在物理位置上处于同一机壳中,有一个单一的系统物理地址空间和每一个处理机均可访问系统内的所有存储器是它的特点。

2. csma/cd

CSMA/CD 载波监听/冲突检测,属于计算机网络以太网的工作类型,即在总线上不断的发出信号去探测线路是否空闲,如果不空闲则随机等待一定时间,在继续探测。直到发出信号为止。

多点接入:说明是总线型网络,许多计算机以多点接入的方式连接在一根总线上。

载波监听:发送前先监听,即每一个站在发送数据之前先要检测一下总线上是否有其他站在发送数据,如果有则暂时不发送数据,要等待信道变为空闲时再发送。

碰撞检测:便发送边监听,即适配器发送数据边检测信道上的信号电压的变化情况,以便判断自己发送数据时其他站是否也在发送数据。当几个站同时在总线上发送数据时,总线上的信号电压将会增大。

3.模拟信号数字化基本原理

声音的数字化包括三大步骤:取样、量化、编码

以下是我找到的具体内容:

一、取样

对连续信号按一定的时间间隔取样。

奈奎斯特取样定理认为,只要取样频率大于等于信号中所包含的最高频率的两倍,则可以根据其取样完全恢复出原始信号,这相当于当信号是最高频率时,每一周期至少要采取两个点。

但这只是理论上的定理,在实际操作中,人们用混叠波形,从而使取得的信号更接近原始信号。

二、量化

取样的离散音频要转化为计算机能够表示的数据范围,这个过程称为量化。

量化的等级取决于量化精度,也就是用多少位二进制数来表示一个音频数据。一般有 8 位, 12 位或 16 位。量化精度越高,声音的保真度越高。以 8 位的举例稍微说明一下其中的原理。若一台计算机能够接收八位二进制数据,则相当于能够接受 256 个十进制的数,即有 256 个电平数,用这些数来代表模拟信号的电平,可以有 256 种,但是实际上采样后的某一时刻信号的电平不一定和 256 个电平某一个相等,此时只能用最接近的数字代码表示取样信号电平。

三、编码

对音频信号取样并量化成二进制,但实际上就是对音频信号进行编码,但用不同的取样频率和不同的量化位数记录声音,在单位时间中,所需存贮空间是不一样的。波形声音的主要参数包括:取样频率.量化位数.声道数.压缩编码方案和数码率等,未压缩前,波形声音的码率计算公式为:波形声音的码率=取样频率*量化位数*声道数/8。波形声音的码率一般比较大,所以必需对转换后的数据进行压缩。常见的方案有如下几种:

4.网络多路复用机制中, ADSL 采用的是哪一种

ADSL (Asymmetric Digital Subscriber Line, 非对称数字用户环路) 是一种新的数据传输方式。它因为上行和下行带宽不对称, 因此称为非对称数字用户线环路。原理它采用频分复用技术把普通的电话线分成了电话、上行和下行三个相对独立的信道, 从而避免了相互之间的干扰。

5. TCP 传输建立连接时为什么使用三次握手机制

TCP 是一种可靠的面向连接的传送服务。它在传送数据时是分段进行的, 主机交换数据必须建立一个会话。

- (1) 对每次发送的数据量跟踪进行协商, 使数据段的发送和接收同步
- (2) 根据所接收到的数据量而确定的数据确认数及数据发送
- (3) 接收完毕后撤销联系, 并建立虚连接

6. 分布式处理涉及哪些关键技术

分布式处理系统与并行处理系统都是计算机体系结构中的两类。并行处理系统是利用多个功能部件或多个处理机同时工作来提高系统性能或可靠性的计算机系统, 这种系统至少包含指令级或指令级以上的并行。并行处理系统的研究与发展涉及计算理论, 算法, 体系结构, 软硬件多个方面, 但它与分布式处理系统有密切的关系, 随着通信技术的发展, 两者的界限越来越模糊。广义上说分布式处理也可以认为是一种并行处理形式。而分布式处理系统将不同地点的或具有不同功能的或拥有不同数据的多台计算机用通信网络连接起来, 在控制系统的统一管理控制下, 协调地完成信息处理任务的计算机系统。一般认为, 集中在同一个机柜内或同一个地点的紧密耦合多处理机系统或大规模并行处理系统是并行处理系统, 而用局域网或广域网连接的计算机系统是分布式处理系统。松散耦合并行计算机中的并行操作系统有时也称为分布式处理系统。

分布式处理系统包含硬件, 控制系统, 接口系统, 数据, 应用程序和人等六个要素。而控制系统中包含了分布式操作系统, 分布式数据库以及通信协议等。

分布式计算环境是在具有多地址空间的多计算机系统上进行计算和信息处理的软件环境。而分布式软件系统是支持分布式处理的软件系统, 它包括分布式操作系统, 分布式程序设计语言及其编译系统, 分布式文件系统和分布式数据库系统等。而 CORBA, COM+ 等是设计分布式软件系统的一些技术。

7. 简述虚电路和数据报

(1) 从电路设置看, 虚电路需要进行电路设置, 数据报无需;

从地址设置看, 虚电路每个分组含有一个短的虚电路号, 数据报有完整地址; 从路由选择及影响来看, 虚电路建好时, 路由就已确定, 所有分组都经过此路由, 数据报的每个分组独立选择路由。路由器失败时, 所有经过路由器的虚电路都将被终止, 数据报服务则除了崩溃时全丢失分组外, 无其他影响; 在拥塞控制方面, 若有足够的缓冲区分配给已经建立的每条虚电路, 较容易控制, 而数据报服务难以控制。

1. 数据报服务的特征: (1) 不需建立连接; (2) 采用全网地址; (3) 要求路由选择; (4) 数据报不一定能按序到达目的节点; (5) 对故障的适应性强; (6) 易于平衡网络流量。

虚电路服务的特征: (1) 要求先建立连接; (2) 全网地址; (3) 路由选择; (4) 按顺序传送分组; (5) 可靠性较高; (6) 适用于交互方式环境。

8. 什么是 IP 地址? 逻辑上 IP 地址如何构成?

IP 地址 (Internet Protocol Address) 用于确定因特网上的每台主机, 它是每台主机唯一性的标识。联网设备用物理地址标识自己, 例如网卡地址。

TCP / IP 用 IP 地址来标识源地址和目标地址, 但源和目标主机却位于某个网络中, 故源地址和目标地址都由网络号和主机号组成, 但这种标号只是一种逻辑编号, 而不是路由器

和计算机网卡的物理地址。对于一台计算机而言, IP 地址是可变的, 而物理地址是固定的。

9. 半双工总线型局域网的覆盖范围取决于什么因素?

传输方式目前无线局域网采用的传输媒体主要有两种, 即微波与红外线。采用微波作为传输媒体的无线局域网按调制方式不同, 又可分为扩展频谱方式与窄带调制方式。

10. 通常路由协议有距离向量法和链路状态法, RIP 和 OSPF 分别是哪一种?

Rip 距离向量法, ospf 是链路状态法。

11. 计算机网络和分布式系统的异同。

我们将地理位置不同, 具有独立功能的多个计算机系统, 通过通信设备和线路互相连接起来, 使用功能完整的网络软件来实现网络资源共享的大系统, 称为计算机网络。

换句话说, 计算机网络既可以用通信线路将几台计算机系统连成简单的网络, 实现信息的收集、分配、传输和处理, 也可以将成千上万的计算机系统和数千公里乃至数万公里的通信线路联成全国或全球的计算机网络。按照网络覆盖的地区不同, 可把计算机网分成局域网 (LAN)、广域网 (WAN)、都市网 (MAN) 以及网间网 (INTERNET) 等。

分布式软件系统(Distributed Software Systems)是支持分布式处理的软件系统,是在由通信网络互联的多处理机体系结构上执行任务的系统。它包括分布式操作系统、分布式程序设计语言及其编译(解释)系统、分布式文件系统和分布式数据库系统等。

分布式操作系统负责管理分布式处理系统资源和控制分布式程序运行。它和集中式操作系统的区别在于资源管理、进程通信和系统结构等方面。

分布式程序设计语言用于编写运行于分布式计算机系统上的分布式程序。一个分布式程序由若干个可以独立执行的程序模块组成,它们分布于一个分布式处理系统的多台计算机上被同时执行。它与集中式的程序设计语言相比有三个特点: 分布性、通信性和稳健性。

分布式文件系统具有执行远程文件存取的能力,并以透明方式对分布在网络上的文件进行管理和存取。

12. 网际互联常用的网间连接器有哪些? 分别对应哪一层的协议转换?

网桥 (Bridge) 是一个局域网与另一个局域网之间建立连接的桥梁。网桥是属于网络层的一种设备, 它的作用是扩展网络和通信手段, 在各种传输介质中转发数据信号, 扩展网络的距离, 同时又有选择地将有地址的信号从一个传输介质发送到另一个传输介质, 并能有效地限制两个介质系统中无关紧要的通信。

路由器

路由器 (Router) 是用于连接多个逻辑上分开的网络。路由器具有判断网络地址和选择路径的功能, 它能在多网络互联环境中建立灵活的连接, 可用完全不同的数据分组和介质访问方法连接各种子网。路由器是属于网络应用层的一种互联设备, 只接收源站或其他路由器的信息。

(1) 物理层 (即常说的第一层、层 L1) 中继系统, 即转发器 (repeater)。

(2) 数据链路层 (即第二层, 层 L2), 即网桥或桥接器 (bridge)。

(3) 网络层 (第三层, 层 L3) 中继系统, 即路由器 (router)。

(4) 网桥和路由器的混合物桥路器 (brouter) 兼有网桥和路由器的功能。

(5) 在网络层以上的中继系统, 即网关 (gateway)。

13. TCP/IP 和 OSI 网络体系结构的异同。

(1) 相同点: 它们都采用了层次结构的概念, 层的功能大体相似, 都具有基于独立协议栈的概念。

(2) 不同点: ①ISO/OSI 参考模型将服务、接口与协议的概念明确化; TCP/IP 参考模型在服务、接口与协议的区别上不很清楚。②ISO/OSI 参考模型先有层的划分, 后有各层协议; TCP/IP 参考模型先有协议, 后有层的划分。③ISO/OSI 参考模型划分为七层; TCP/IP 参考

模型划分为四层。④ISO/OSI 的网络层支持无连接和面向连接的通信，传输层只支持面向连接的通信；TCP/IP 的网络层只支持无连接的通信，传输层提供无连接和面向连接的通信。

14. 什么是网桥？

网桥工作在数据链路层，将两个局域网（LAN）连起来，根据 MAC 地址（物理地址）来转发帧，可以看作一个“低层的路由器”（路由器工作在网络层，根据网络地址如 IP 地址进行转发）。它可以有效地联接两个 LAN，使本地通信限制在本网段内，并转发相应的信号至另一网段，网桥通常用于联接数量不多的、同一类型的网段。

网桥通常有透明网桥和源路由选择网桥两大类。

（1）透明网桥

简单的讲，使用这种网桥，不需要改动硬件和软件，无需设置地址开关，无需装入路由表或参数。只须插入电缆就可以，现有 LAN 的运行完全不受网桥的任何影响。

（2）源路由选择网桥

源路由选择的核心思想是假定每个帧的发送者都知道接收者是否在同一局域网（LAN）上。当发送一帧到另外的网段时，源机器将目的地址的高位设置成 1 作为标记。另外，它还在帧头加进此帧应走的实际路径。

15. 谈谈你了解的几种网络安全技术，你认为他们分别适用于什么场合？

包括防火墙技术,安全认证技术,病毒防治,或是其他一些和网络安全相关的技术,入侵检测

16. Internet 划分成很多自治系统 AS，在 AS 内部和 AS 之间的路由通常采用什么协议实现路由选择？

一个 AS（Autonomous System，自治系统，指一个互连网络，就是把整个 Internet 划分为许多较小的网络单位，这些小的网络有权自主地决定在本系统中应采用何种路由选择协议）内的路由协议称为内部网关协议（interior gateway protocol），AS 之间的路由协议称为外部网关协议（exterior gateway protocol）。这里网关是路由器的旧称。现在正在使用的内部网关路由协议有以下几种：RIP-1，RIP-2，IGRP，EIGRP，IS-IS 和 OSPF。其中前 4 种路由协议采用的是距离向量算法，IS-IS 和 OSPF 采用的是链路状态算法。对于小型网络，采用基于距离向量算法的路由协议易于配置和管理，且应用较为广泛，但在面对大型网络时，不但其固有的环路问题变得更难解决，所占用的带宽也迅速增长，以至于网络无法承受。因此对于大型网络，采用链路状态算法的 IS-IS 和 OSPF 较为有效，并且得到了广泛的应用。IS-IS 与 OSPF 在质量和性能上的差别并不大，但 OSPF 更适用于 IP，较 IS-IS 更具有活力。IETF 始终在致力于 OSPF 的改进工作，其修改节奏要比 IS-IS 快得多。这使得 OSPF 正在成为应用广泛的一种路由协议。现在，不论是传统的路由器设计，还是即将成为标准的 MPLS（多协议标记交换），均将 OSPF 视为必不可少的路由协议。

（1）路由协议大概分为 2 种，IGP 和 BGP

（2）IGP(interior Gateway Protocols)内部网关协议

内部网关协议（IGP）是一种专用于一个自治网络系统（比如：某个当地社区范围内的一个自治网络系统）中网关间交换数据流转通道信息的协议。网络 IP 协议或者其他的网络协议常常通过这些通道信息来决定怎样传送数据流。目前最常用的两种内部网关协议分别是：路由信息协议（RIP）和最短路径优先路由协议，目前的 IGP 有 RIP、OSPF、IGRP、EIGRP、IS-IS 等协议。（OSPF）。

（3）BGP (Border Gateway Protocol,边界网关协议)是用来连接 Internet 上的独立系统的路由选择协议。它是 Internet 工程任务组制定的一个加强的、完善的、可伸缩的协议，其中有可以分有 IBGP 和 EBGP

（4）路由协议交换信息的过程简单的说，就是一个路由协议的度量 和 另外一个不同

路由协议度量 一个换算的过程，通过重分发命令进行，打个比方，你用 RIP，默认 RIP 按跳数计算路由，最大 16 条，现在，你要把 RIP 重新分配到 OSPF 中，OSPF 的度量值是根据 SPF 算法计算 COST 值，那么本身 OSPF 不理解 RIP 的跳数度量值，那么要需要通过手工的方法让 RIP 的度量跳数变成 COST 进入 OSPF 度量中

17. 简述数据链路层和传输层在流量控制上的区别。

虽然两层都有流量控制，但是控制的对象是不同的。这两层工作的方式不同，传输的数据对象也不同。数据链路层传输的是帧，面对的是物理设备；而传输层传输的是包，面对的是高层的应用。这样，他们分别要应对不同的速度和不同的介质，因而就需要分别的独立的流量控制。

18. 决定局域网特征的主要技术有哪些？

主要技术要素是：网络拓扑、传输介质与介质访问控制方法

网络拓扑结构分类：星型拓扑 环型拓扑 总线型拓扑 不规则型

介质访问控制方法：总线结构的带冲突检测的载波侦听多路访问 CSMA/CD 方法、环型结构的令牌环（Token Ring）访问控制方法和令牌总线（Token Bus）访问控制方法。

七、面向对象（参照面向对象历史回顾）

面向对象方法的“封装性”有什么问题？怎样解决？

封装性就是把对象的属性和服务结合成一个独立的相同单位，并尽可能隐蔽对象的内部细节，包含两个含义

◇ 把对象的全部属性和全部服务结合在一起，形成一个不可分割的独立单位（即对象

◇ 信息隐蔽，即尽可能隐蔽对象的内部细节，对外形成一个边界（或者说形成一道屏障），只保留有限的对外接口使之与外部发生联系。

封装的原则在软件上的反映是：要求使对象以外的部分不能随意存取对象的内部数据（属性），从而有效的避免了外部错误对它的“交叉感染”，使软件错误能够局部化，大大减少查错和排错的难度。

封装性是保证软件部件具有优良的模块性的基础。

面向对象的类是封装良好的模块，类定义将其说明（用户可见的外部接口）与实现（用户不可见的内部实现）显式地分开，其内部实现按其具体定义的作用域提供保护。

对象是封装的最基本单位。封装防止了程序相互依赖性而带来的变动影响。面向对象的封装比传统语言的封装更为清晰、更为有力。

2. 为什么说面向对象设计所获得的软件结构同问题本身的结构是一致的？

软件体系结构是具有一定形式的结构化元素，即构件的集合，包括处理构件、数据构件和连接构件。处理构件负责对数据进行加工，数据构件是被加工的信息，连接构件把体系结构的不同部分组合连接起来。

3. 继承机制是面向对象技术重要特征，谈谈继承机制软件技术所产生的作用。

避免重复地实现某个功能，用继承能省很多事。

4. 简述面向对象方法的概念

（1）面向对象程序设计的基本概念

面向对象程序设计中的概念主要包括：对象、类、数据抽象、继承、动态绑定、数据封装、多态性、消息传递。通过这些概念面向对象的思想得到了具体的体现。

1) 对象

对象是运行期的基本实体，它是一个封装了数据和操作这些数据的代码的逻辑实体。

2) 类

类是具有相同类型的对象的抽象。一个对象所包含的所有数据和代码可以通过类来构造。

3) 封装

封装是将数据和代码捆绑到一起，避免了外界的干扰和不确定性。对象的某些数据和代码可以是私有的，不能被外界访问，以此实现对数据和代码不同级别的访问权限。

4) 继承

继承是让某个类型的对象获得另一个类型的对象的特征。通过继承可以实现代码的重用：从已存在的类派生出的一个新类将自动具有原来那个类的特性，同时，它还可以拥有自己的新特性。

5) 多态

多态是指不同事物具有不同表现形式的能力。多态机制使具有不同内部结构的对象可以共享相同的外部接口，通过这种方式减少代码的复杂度。

6) 动态绑定

绑定指的是将一个过程调用与相应代码链接起来的行为。动态绑定是指与给定的过程调用相关联的代码只有在运行期才可知的一种绑定，它是多态实现的具体形式。

7) 消息传递

对象之间需要相互沟通，沟通的途径就是对象之间收发信息。消息内容包括接收消息的对象的标识，需要调用的函数的标识，以及必要的信息。消息传递的概念使得对现实世界的描述更容易。

3. 面向对象语言

一个语言要称为面向对象语言必须支持几个主要面向对象的概念。根据支持程度的不同，通常所说的面向对象语言可以分成两类：基于对象的语言，面向对象的语言。

基于对象的语言仅支持类和对象，而面向对象的语言支持的概念包括：类与对象、继承、多态。举例来说，Ada 就是一个典型的基于对象的语言，因为它不支持继承、多态，此外其他基于对象的语言还有 Alphard、CLU、Euclid、Modula。面向对象的语言中一部分是新发明的语言，如 Smalltalk、Java，这些语言本身往往吸取了其他语言的精华，而又尽量剔除他们的不足，因此面向对象的特征特别明显，充满了蓬勃的生机；另外一些则是对现有的语言进行改造，增加面向对象的特征演化而来的。如由 Pascal 发展而来的 Object Pascal，由 C 发展而来的 Objective-C，C++，由 Ada 发展而来的 Ada 95 等，这些语言保留着对原有语言的兼容，并不是纯粹的面向对象语言，但由于其前身往往是有一定影响的语言，因此这些语言依然宝刀不老，在程序设计语言中占有十分重要的地位。

4. 面向对象程序设计的优点

面向对象出现以前，结构化程序设计是程序设计的主流，结构化程序设计又称为面向过程的程序设计。在面向过程程序设计中，问题被看作一系列需要完成的任务，函数（在此泛指例程、函数、过程）用于完成这些任务，解决问题的焦点集中于函数。其中函数是面向过程的，即它关注如何根据规定的条件完成指定的任务。

在多函数程序中，许多重要的数据被放置在全局数据区，这样它们可以被所有的函数访问。每个函数都可以具有它们自己的局部数据。下图显示了一个面向过程程序中函数和数据的关系。

图 1 面向过程程序设计中函数和数据的关系示例

这种结构很容易造成全局数据在无意中被其他函数改动，因而程序的正确性不易保证。面向对象程序设计的出发点之一就是弥补面向过程程序设计的一些缺点：对象是程序的基本元素，它将数据和操作紧密地连结在一起，并保护数据不会被外界的函数意外地改变。下图显示了一个面向对象程序中对象与函数和数据的关系。

图 2 面向对象程序设计中函数和数据的关系示例

比较面向对象程序设计和面向过程程序设计，还可以得到面向对象程序设计的其他优点：

1) 数据抽象的概念可以在保持外部接口不变的情况下改变内部实现, 从而减少甚至避免对外界的干扰;

2) 通过继承大幅减少冗余的代码, 并可以方便地扩展现有代码, 提高编码效率, 也减低了出错概率, 降低软件维护的难度;

3) 结合面向对象分析、面向对象设计, 允许将问题域中的对象直接映射到程序中, 减少软件开发过程中中间环节的转换过程;

4) 通过对对象的辨别、划分可以将软件系统分割为若干相对为独立的部分, 在一定程度上更便于控制软件复杂度;

6) 以对象为中心的设计可以帮助开发人员从静态(属性)和动态(方法)两个方面把握问题, 从而更好地实现系统;

7) 通过对象的聚合、联合可以在保证封装与抽象的原则下实现对象在内在结构以及外在功能上的扩充, 从而实现对象由低到高的升级。

面对对象的程序设计方法

在数据输入模块内部设计中, 采用面向对象的设计方法。[6]面向对象的基本概念如下:

对象:对象是要研究的任何事物。从一本书到一家图书馆, 单的整数到整数列庞大的数据库、极其复杂的自动化工厂、航天飞机都可看作对象, 它不仅能表示有形的实体, 也能表示无形的(抽象的)规则、计划或事件。对象由数据(描述事物的属性)和作用于数据的操作(体现事物的行为)构成一独立整体。从程序设计者来看, 对象是一个程序模块, 从用户来看, 对象为他们提供所希望的行为。在对内的操作通常称为方法。

类:类是对象的模板。即类是对一组有相同数据和相同操作的对象的定义, 一个类所包含的方法和数据描述一组对象的共同属性和行为。类是在对象之上的抽象, 对象则是类的具体化, 是类的实例。类可有其子类, 也可有其它类, 形成类层次结构。

消息:消息是对象之间进行通信的一种规格说明。一般它由三部分组成: 接收消息的对象、消息名及实际变元。

面向对象主要特征:

封装性:封装是一种信息隐蔽技术, 它体现于类的说明, 是对象的重要特性。封装使数据和加工该数据的方法(函数)封装为一个整体, 以实现独立性很强的模块, 使得用户只能见到对象的外特性(对象能接受哪些消息, 具有那些处理能力), 而对象的内特性(保存内部状态的私有数据和实现加工能力的算法)对用户是隐蔽的。封装的目的在于把对象的设计者和对象者的使用分开, 使用者不必知晓行为实现的细节, 只须用设计者提供的消息来访问该对象。

继承性:继承性是子类自动共享父类之间数据和方法的机制。它由类的派生功能体现。一个类直接继承其它类的全部描述, 同时可修改和扩充。继承具有传递性。继承分为单继承(一个子类只有一父类)和多重继承(一个类有多个父类)。类的对象是各自封闭的, 如果没继承性机制, 则类对象中数据、方法就会出现大量重复。继承不仅支持系统的可重用性, 而且还促进系统的可扩充性。

多态性:对象根据所接收的消息而做出动作。同一消息为不同的对象接受时可产生完全不同的行动, 这种现象称为多态性。利用多态性用户可发送一个通用的信息, 而将所有的实现细节都留给接受消息的对象自行决定, 如是, 同一消息即可调用不同的方法。例如: **Print** 消息被发送给一图或表时调用的打印方法与将同样的 **Print** 消息发送给一正文文件而调用的打印方法会完全不同。多态性的实现受到继承性的支持, 利用类继承的层次关系, 把具有通用功能的协议存放在类层次中尽可能高的地方, 而将实现这一功能的不同方法置于较低层次, 这样, 在这些低层次上生成的对象就能给通用消息以不同的响应。在 **OOP** 中可通过在派生类中重定义基类函数(定义为重载函数或虚函数)来实现多态性。

综上所述，在面向对象方法中，对象和传递消息分别表现事物及事物间相互联系的概念。类和继承是适应人们一般思维方式的描述范式。方法是允许作用于该类对象上的各种操作。这种对象、类、消息和方法的程序设计范式的基本点在于对象的封装性和类的继承性。通过封装能将对象的定义和对象的实现分开，通过继承能体现类与类之间的关系，以及由此带来的动态联编和实体的多态性，从而构成了面向对象的基本特征。

面向对象设计是一种把面向对象的思想应用于软件开发过程中，指导开发活动的系统方法，是建立在“对象”概念基础上的方法学。对象是由数据和容许的操作组成的封装体，与客观实体有直接对应关系，一个对象类定义了具有相似性质的一组对象。而每继承性是对具有层次关系的类的属性和操作进行共享的一种方式。所谓面向对象就是基于对象概念，以对象为中心，以类和继承为构造机制，来认识、理解、刻画客观世界和设计、构建相应的软件系统。按照 Bjarne Stroustrup 的说法，面向对象的编程范式：

1. 决定你要的类；
2. 给每个类提供完整的一组操作；
3. 明确地使用继承来表现共同点。

由这个定义，我们可以看出：面向对象设计就是“根据需求决定所需的类、类的操作以及类之间关联的过程”。

面向对象设计方法的特点和面临的问题

面向对象设计方法以对象为基础，利用特定的软件工具直接完成从对象客体的描述到软件结构之间的转换。这是面向对象设计方法最主要的特点和成就。面向对象设计方法的应用解决了传统结构化开发方法中客观世界描述工具与软件结构的不一致性问题，缩短了开发周期，解决了从分析和设计到软件模块结构之间多次转换映射的繁杂过程，是一种很有发展前途的系统开发方法。

但是同原型方法一样，面向对象设计方法需要一定的软件基础支持才可以应用，另外在大型的 MIS 开发中如果不经自顶向下的整体划分，而是一开始就自底向上的采用面向对象设计方法开发系统，同样也会造成系统结构不合理、各部分关系失调等问题。所以面向对象设计方法和结构化方法目前仍是两种在系统开发领域相互依存的、不可替代的方法。