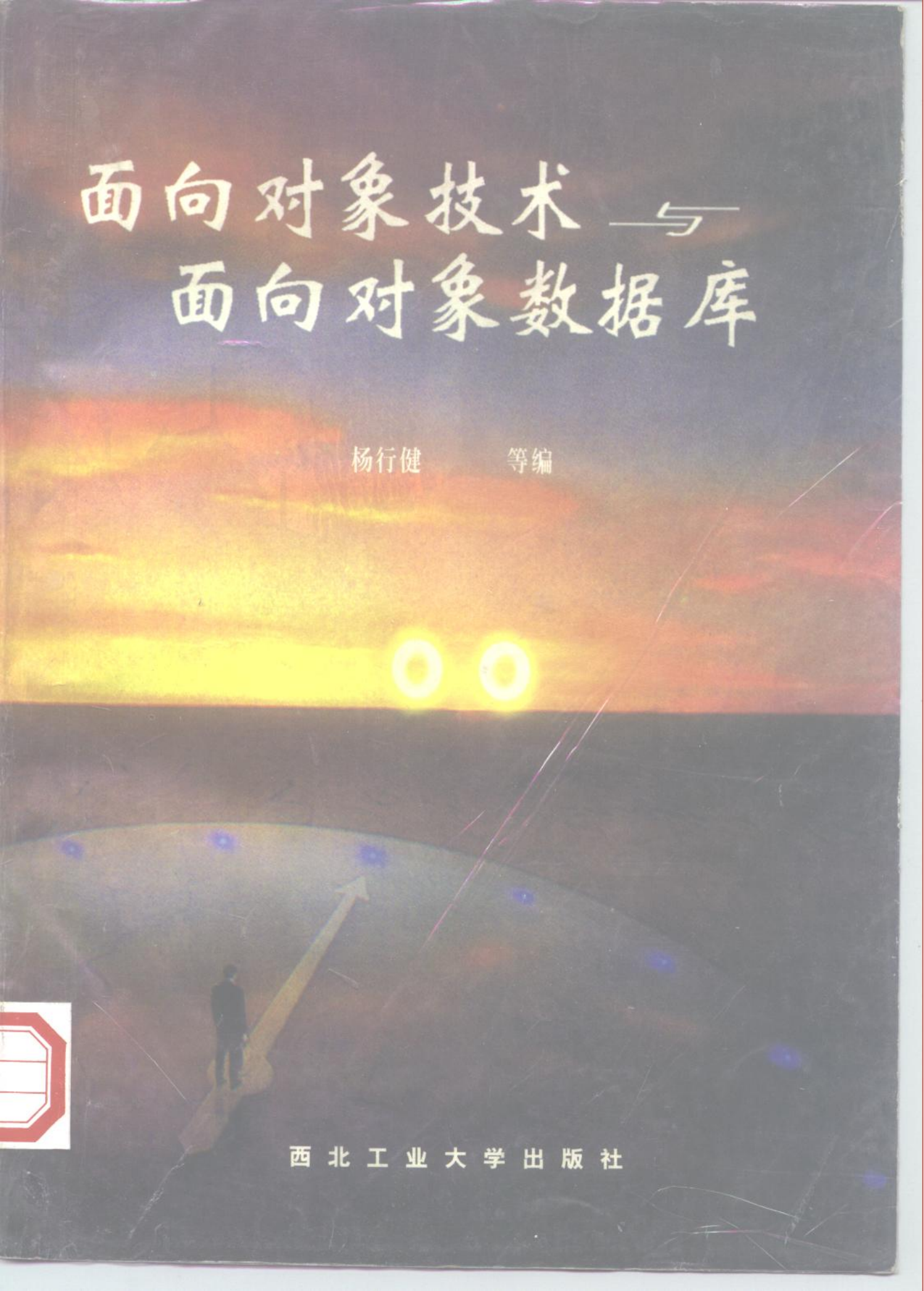
面向对象技术与面向对象数据库

杨行健　　等编



西北工业大学出版社

面向对象技术与面向对象数据库

杨行健　等编

西北工业大学出版社
1996年1月　西安

（陕）新登字 009 号

【内容简介】 面向对象技术与传统的结构化软件技术相比，在提高软件开发效率和质量、改善软件可靠性与重用性方面都有着不可比拟的优点，所以它是 90 年代计算机界关心的焦点。本书对面向对象的分析与设计方法及面向对象数据库技术作了较全面的介绍。

全书共分三大部分，第一部分为面向对象的基本概念，对面向对象的分析与设计方法、面向对象语言 C++及面向对象数据库等的一些特点及基本概念作了简要的介绍。第二部分为面向对象数据库导论，结合 ORION 系统较详细地介绍了面向对象数据库的数据模型、语言、结构及实现方法等。第三部分为一些面向对象数据库系统的简介，对目前国外高校及工业部门的一些著名的原型系统及市场上销售的一些商品化系统作了较详细的介绍。

本书可供高等院校计算机软件等有关专业的师生以及各部门的计算机软件工作者，科研、工程技术人员参考。

面向对象技术与面向对象数据库

杨行健 等编

责任编辑 刘 红

责任校对 钱伟峰

*

（710072 西安市友谊西路 127 号 电话 8493844）

陕西省新华书店经销

西北工业大学出版社印刷厂印装

ISBN 7-5612-0786-7/TP・91

*

开本：787×1092 毫米 1/16 开本 印张：22.125 字数：535 千字

1996 年 1 月第 1 版 1996 年 1 月第 1 次印刷

印数：3 000 册 定价：26.00 元

前　言

面向对象技术被人们誉为软件技术的一场革命，它与传统的结构化的软件技术相比具有许多优点。应用面向对象技术可以有效地提高软件的开发效率和质量，改善软件的可靠性和重用性，所以今天许多软件和硬件销售商都在其产品上贴上“面向对象”的标签，都说他们的产品是 OO（Objected - Oriented）的或具有 OO 的特点，似乎 OO 与 good 变成同义词。面向对象已成为目前计算机界关心的焦点，面向对象的概念与应用已经渗透到计算机的许多领域，如程序设计与开发、CAD/CAM、图形处理、动态仿真、数据库系统、用户界面、分布式系统、人工智能、体系结构等。一些新的工程概念及其实现，如应用集成平台、综合集成工程、网络工程、并行工程等也引入了面向对象的概念。

本书是航空工业总公司组织开发的 CIGMA（Computer Integrated Geometry Modelling, Manufacturing and Application）系统中的一个新技术跟踪项目，旨在探讨 90 年代 CAD/CAM 系统的软件开发环境。由于面向对象涉及的范围较广，本书仅侧重于对面向对象的程序设计方法及面向对象数据库技术作较全面的阐述。

本书由三大部分组成，第一部分的标题为“面向对象的基本概念”，对面向对象的概念和特点、面向对象的分析与设计方法、面向对象的语言及面向对象的数据库作一概要的介绍。第二部分的标题为“面向对象数据库导论”，结合 MCC（Microelectronics and Computer technology Corporation）公司开发的面向对象数据库系统 ORION 的研究成果，对面向对象数据库系统的数据模型、数据库语言、数据库体系结构、所存在问题及发展方向作了较全面的叙述。第三部分标题为“一些面向对象数据库系统简介”，对目前国际上一些高校和研究机构开发的较著名的原型系统及一些商品化的系统，如 Versant，Objectstore 等作简单的介绍。

本书第一、二、三、十二、二十四、二十五章由杨行健编写，第四、五、六章由陈良忠编写，第七、十一章由方惟一编写，第八、九章由张学宏编写，第十、十三章由武君胜编写，第十四章由杨爱国编写，第十五章由常刚编写，第十六章由张玲英编写，第十七、十八章由汤幼宁编写，第十九、二十、二十一章由朱明编写，第二十二、二十三章由鲁民清编写。杨行健汇编和校审了全书。

在本书编写过程中得到航空工业总公司科技局计算机处徐微同志的许多指导和帮助，仅在此表示感谢。

本书可作为高等学校数据库、人工智能、图形学及 CAD/CAM 等专业高年级学生、研究生的教学参考书，也可供从事与这方面有关的专业的科研人员、工程技术人员及其他有关人员参考。

由于我们的水平有限，书中难免有错误和不妥之处，欢迎读者批评指正。

编　者

1994 年 11 月于航空工业总公司第 631 所

目 录

第一部分 面向对象的基本概念

第二部分 面向对象数据库导论

第三部分 一些面向对象数据库系统简介

第一部分　面向对象的基本概念

第一章　面向对象的分析与设计方法

1.1 引　　言

在计算机应用领域中，软件开发的步履维艰与硬件技术发展的日新月异形成了鲜明的对照。自 90 年代以来，软件与硬件之间的差距至少有两代处理器之多，而且差距还在不断扩大。软件是一种抽象的逻辑思维的产品，所以软件的研制过程实质上是软件研究人员的“思考”过程。有人说软件就像泥巴一样，你愿意捏成什么样就是什么样。正由于其“容易”修改，因而没有一个软件在投入运行之后就不再进行修改扩充的。软件人员可以按各自的爱好进行工作，没有统一的标准可循，管理人员事前又难以精确地估计项目所需的经费、时间，技术人员在项目完成之前也难以预料系统是否成功；更糟的是，系统失败后又往往无法挽回，除非再从头做起，但由于时间和经费的限制这又是不可能的。所以研制过大型软件系统的工程师们回首往事时，经常把自己所完成的一些工程称为“遗憾工程”。

国外在研制一些大型软件系统时，遇到了许多困难，最著名的例子是 IBM 公司的 OS/360 系统和美国空军某后勤系统。这两个系统都花费了数千人几年的努力，历尽艰辛，但结果却令人失望。OS/360 系统负责人 Brooks 生动地描述了研制过程中的困难和混乱：“……像巨兽在泥潭中作垂死地挣扎，挣扎得越猛，就陷得越深，最后没有一个野兽能逃脱淹没的命运……程序设计就像这样一个泥潭……一批批程序员在泥潭中挣扎……没有料到问题竟会这样的棘手……”这就是所谓的“软件危机”现象。

造成这种危机的根本原因是我们目前采用的冯·诺依曼机类型计算机的求解问题的方法空间结构与人们认识问题的问题空间结构很不一致。冯·诺依曼机的基本特征是顺序地执行指令，按地址访问线性的存储空间。在这种机器上所能直接接受的是面向过程的语言，这是一种比较难以理解的、与人们的自然语言相距甚远的语言。近 20 年来，人们为了克服软件危机、控制软件的开发质量和提高软件的生产效率，对软件开发的方法进行了大量深入的研究，提出了用管理工程项目的方法开发软件工程的方法。到目前为止，最典型的传统方法是结构化的需求分析、结构化的系统设计方法。这种方法从本质上看仍具有冯·诺依曼机系统结构的特点，是软件开发人员从开发软件的立场出发而确定的，并不是从人们认识客观世界的过程和方法出发的。

我们虽已进入 90 年代，但软件的开发并没有取得突破性的进展，人们为了使问题空间结构和方法空间结构取得一致，正在两个方向上进行努力，一种努力是使未来的计算机能接受、理解和处理人的自然语言，但到目前为止，还存在着许多理论和技术问题有待解决。目前人们为了部分地、有限地缓解软件危机、缩小语言鸿沟，正热衷于面向对象方法的研究。另一种努力是在研制各种非冯·诺依曼机的体系结构，使计算机更接近于人的思维方式，例如松耦合（分布主存）结构的处理机系统、神经网络计算机系统，以及为面向对象方法而设计的计算机体系结构也正在研究之中。

可以预料，面向对象软件的体系结构将在90年代占据主导地位。向这种新型体系结构的过渡正在进行，这体现在已出现的面向对象语言、数据库、界面、操作系统及开发环境上。新的数据类型、分布式处理、多媒体应用程序以及终端用户计算机都是实现面向对象软件蓝图的推动力。近年来实现面向对象系统方面已有了很大进展。如今的图形界面使用户熟悉了对象操作。面向对象语言中，C++已成为事实上的标准。大多数主流的高级语言大多实现了面向对象的扩充。操作系统也正在被扩充以支持基于对象的应用程序设计。最后，90年代不仅将面向对象程序设计引入开发人员，而且也将引入最终用户。

1.2 面向对象技术的形成与发展

面向对象并非是一个新的概念，实际上已有二十几年的历史。寻其根源可追溯到60年代的挪威，当时挪威计算中心的Kristen Nygaard和Ole-Johan Dahl开发了一种称作Simula 67的仿真语言。Simula 67首次引入了类、协同程序和子类的概念，这很像今天的面向对象语言。

以后，在70年代中期Alan Key在施乐公司的PARC实验室工作时设计开发了Smalltalk语言，这个名字取自“少说话(talk small)”，意义是你通过很少话语就可以完成许多工作。该语言的每个元素都被作为一个对象来实现。Smalltalk的程序设计环境及其相关的各种方面都是面向对象的。即使今天，Smalltalk仍被认为是最纯的面向对象语言。Simula 67和Smalltalk的开发为今天的研究开发工作奠定了基础。

面向对象的概念已经渗透到几个不同的领域：编程语言、用户接口、人工智能和数据库等方面。编程语言的研究者们沿着两种路径开发面向对象编程的方法。一种是新的面向对象语言的开发，如像Smalltalk，Traits，Eiffel和Trellis/Owl。另一种是传统语言的扩充，如像Flavors；Object Lisp；Oaklisp；Loops以及LISP扩充的Common Loops；PASCAL扩充的CLASCAL以及C扩充的Objective C和C++（值得一提的是C++与一般其他的面向对象语言相比有许多独到之处，由于C++与C完全兼容，并保证内部一致性、高效率等，这就使已大量采用C语言开发的编程人员、系统、环境容易向C++扩展。尤其是经过大量实践，已证明在源程序和连接等阶段无严重的与C不兼容情况，也未出现程序在运行时间或空间过载现象。因此C++在短短的几年就获得广泛地使用。在“技术预测”发展趋势时，当问及“哪一种编程语言会成为今后几年最流行、最通用的语言?”时，参加预测的专家一致公认是C++。)。图1.1给出了面向对象语言的发展情况。苹果APPLE公司的Macintosh MacDraw和Hypercard是目前流行的面向对象的用户接口。自从Marvin Minsky引入知识表示模式框架以来，人工智能的研究者们已经开发了如像KEE，ART等基于框架的知识表示和操作的面向对象特点的语言。在数据库领域，关于语义数据模型的研究已经将数据模型的概念引入面向对象编程和知识表示语言中。

今后，基本的面向对象的概念是基于框架的知识表示和推理系统、面向对象编程环境以及先进的面向对象人机接口系统所采用的共同方法。它们可能是将来建立高性能的智能编程系统的一种形式。

近些年来面向对象技术是一个相当活跃的研究领域，举行的面向对象编程（Ecop）会议，推理和面向对象数据库（DOOD）会议，以及面向对象数据库（OODB）专业讨论会等已显现出面向对象概念的软件越来越普及且显得越来越重要。甚至，一种新杂志“Journal of

Objectoriented Programming" 几年前已经出版。

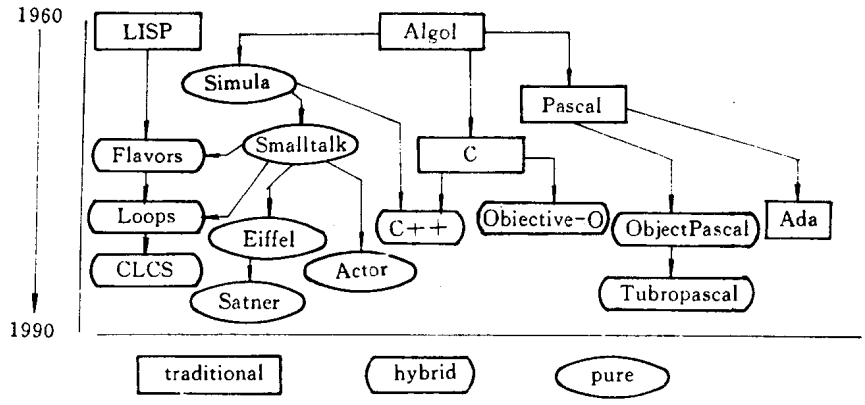


图 1.1　面向对象语言的发展情况

1.3　面向对象方法与结构化方法的比较

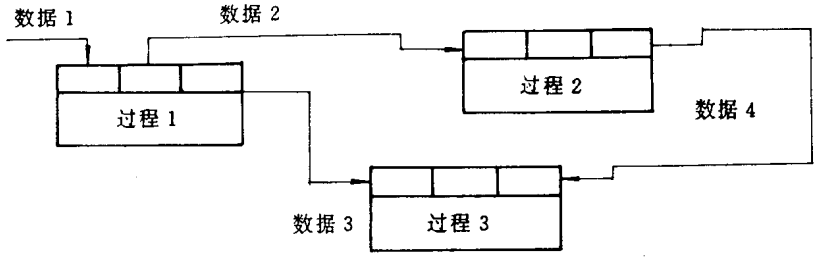
结构化方法强调了功能抽象与模块化，程序（program）是一些过程（procedure）或子程序（Subroutine）的集合，这些过程通过参数传送数据。每个过程操纵它的参数，有时修改它们，也可能返回一个值。因而结构化方法看作处理一系列的过程，也就是以模块（即过程）为中心的开发方法，如图 1.2（a）所示。

在面向对象的系统中，世界被看成是独立对象的集合，对象之间通过过程（在面向对象术语中称之为"消息"）相互通讯，对象具有"智能化"的结构，它将数据和消息"封装"（Encapsulation）在一起，对一个对象的存取或修改仅通过其外部的接口子程序，其内部的实现细节、数据结构及对它的操作是不可见的（在面向对象术语中称之为"信息隐藏"（Information Hiding））。对象是主动实体而过程（消息）是被动实体，过程和它们的参数一起从一个对象传递到另一个对象。一个对象研究对它提出的请求（即过程或消息）并激活它们，即面向对象是以数据为中心的开发方法。以对象为中心的开发方法较以过程为中心的开发方法优越。具体表现在如下几个方面：

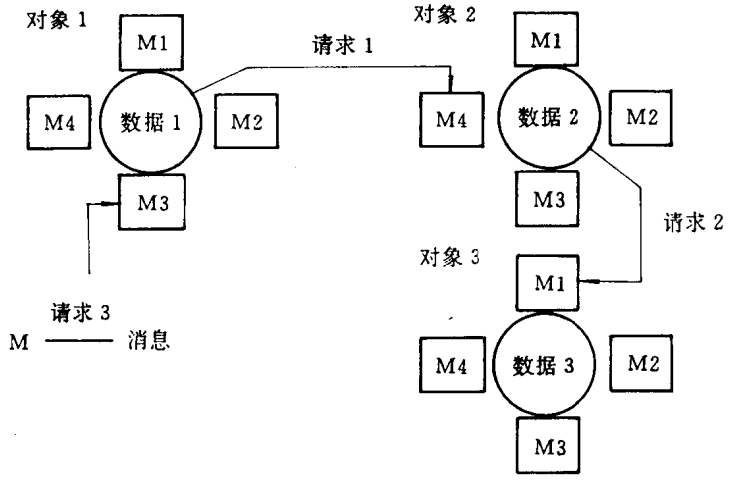
a. 采用对象为中心的开发方法能更自然更直接地反映真实世界的问题空间，因此按面向对象方法进行系统分析、设计时，由于对象、类、子类都自然对应于实际问题的物理或逻辑实体，其编程工作量仅是将问题译成代码。这样使问题转换工作量达最小程度。事实上，采用面向对象方法较之结构化方法在编程量方面可降低 40%至 90%。

b. 以对象为中心开发方法采用消息传递机制作为对象之间相互通讯的唯一方式。对象本身具有很强的自治性和独立性，接收消息的对象有权负责响应与处理所收到的消息，按消息名激活本对象内相应的操作（称为方法）并执行该操作，返回适当的结果。发送消息的对象还可将同一条消息同时发送至多个接收消息的对象中，并允许这些接收同一条消息的对象按照自身的适当方式加以响应。这样就使消息传递的机制能很自然地与分布式并行程序、多机系统、网络通信等模型取得一致，从而强有力地支持复杂大系统的分析与运行。

c. 以对象为中心的解题方法具有独特的继承性和更丰富的多态性，使这种开发方法更易于扩充，能很好地适应复杂大系统不断发展与变化的要求。



(a) 结构化方法



(b) 面向对象的方法

图 1.2 结构化方法与面向对象方法的比较

1.4 面向对象的一些基本概念

过去的哲学家和思想家们对人类是怎样构造问题空间提出了分类理论，在大英百科全书关于“分类学理论”中提出：

人类在认识和理解现实世界的过程中，普通运用着三个构造法则：

a. 区分对象及其属性，例如，区分一棵树和树的大小或空间位置。

b. 区分整体对象及其组成部分，例如，区分一棵树和树枝。

c. 不同对象类的形成及区分，例如，所有树的类和所有石头的类的形成和区分。

这三个方法分别对应于对象和属性，等级结构及分类结构。这总结了人类认识问题的两

种方法，一种是从一般到特殊的演绎方法，也是一种分类方法，面向对象方法恰好符合这一演绎方法。例如，动物世界可以由分类图示（图 1.3）表征，其中每个结点表示一类动物，括号内表该类对象的属性。另一种是从特殊到一般的归纳方法。面向对象系统将一大批相似或相同的对象归出类的过程恰是这样一个归纳过程。例如，一条黄狗是一个对象，一条白狗也是一个对象，这两个对象除了在颜色上不同外，其他狗的特征完全一样，这样，我们便可构造出一个类——“狗”。其中描述了狗的所有共同特征，比如，会叫、具有犬齿、嗅觉灵敏、有颜色、忠实等等，而黄狗与白狗都是这个狗类的实例（如图 1.4 所示）。

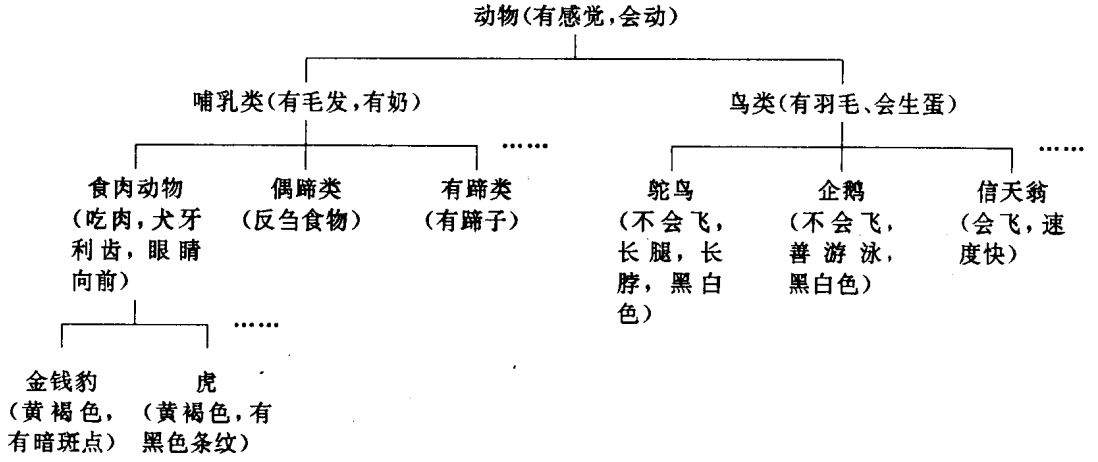


图 1.3　动物世界的演绎分解

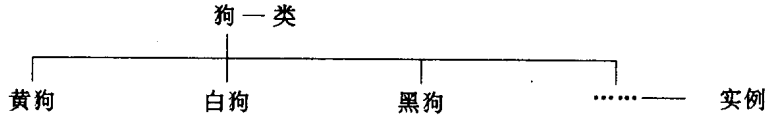


图 1.4　狗类的归纳总结

面向对象方法的优异特性体现在抽象、封装、继承性、多态性等几个主要概念方面，现在我们分别来介绍这些概念。

1.4.1　对象（Object）

世界上一切事物都是对象，小至一根针，大至一个工厂，一所学校，从哲学概念上讲，它们都是客观对象。我们大脑中的概念和认识也是一些对象，是主观对象。一个对象无非是这样一个实体，它具有一个名字标识，并有自身的状态和自身的功能。世界上所有的事物就是如此简单，这恰是面向对象技术所追求的目标——将世界上的问题求解尽可能地简单化。

一个对象之所以能够独立存在，是因为它具有自身的状态（State）。比如，五脏六腑就是人的内部状态。一个对象所具有的这些状态并非完全是直接为外界服务，但它们本身又是能够为外界服务的基础，好比一个人没有好的身体就不能很好地为社会服务。在面向对象系统中，一个对象的状态是通过域来描述的，也称为私有存储单元。在 C++中叫做成员变量，都

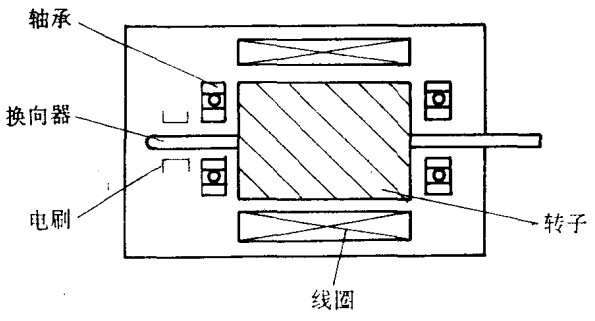
是用于存放一个对象状态的。对象的一个重要特征表现在它的私有存储单元只能由它自己的操作进行处理。这个特征保证了对象和实现只依赖于它本身的状态。

1.4.2　消息（Message）

我们仍以人这样的对象来进行分析，一个人不是生活在真空中，总是要和其他人交往，请求（invoke）他人帮助解决一些问题。这里的“请求”便是一个人与其他人进行交往的手段。在面向对象技术的专业术语中，将这些请求称之为“消息”。日常生活中不仅有请求，而且还会有命令，命令也是一种消息。在面向对象技术中，消息是对象之间交互的手段，是外界能够引用对象操作及获取对象状态的唯一方式。这个特征保证了对象的实现只依赖于它本身的状态和所能接受的消息，而不依赖于其他对象。

1.4.3　协议与封装（Protocol and Encapsulation）

协议或称规格说明（Specification），是一个对象对外服务的说明，它告知一个对象可以为外界做什么。外界能够并且只能向该对象发送协议中所提供的消息，请求该对象服务。因此它是由一个对象能接受并且愿意接受的所有消息构成的对外接口。也就是说，请求对象进行操作的唯一途径就是通过协议中提供消息进行。即使一个对象可以完成某一功能，但它没有将该功能放入协议中去，外界对象依然不能请求它完成这一功能。所以协议是一个对象所能接受的所有消息的集合。



对象、消息和协议是面向对象的支柱概念，同时这些概念一起又引入了一个新概念——封装。从字面上看，封装就是将某件事物包围起来，使外界不必知道实际内容。每个对象都把状态和行为（behaviour）封装在一起。对象的状态是该对象属性值的集合，而对象的行为是在对象状态上操作的方法的集合。封装的意义在于对象的访问只能按对象提供给外界的协议接口进行，并且只能通过协议上提供的操作向该对象发送消息请求它工作。图 1.5 给出一个封装与协议的例子。

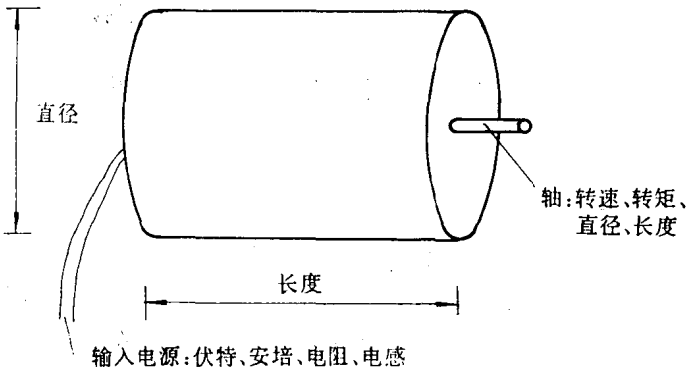


图 1.5　马达的封装与协议