
学校的理想装备
电子图书·学校专集
校园网上的最佳资源
中小学信息科学知识
数据库系统
BOOK
内部资料 非卖品

总　序

本世纪初叶，商务印书馆王云五先生得到胡适之、蔡子民、吴稚晖、杨杏佛、张菊生等 30 余位知名学者、社会贤达鼎力相助，编纂出版了《万有文库》丛书。是书行世，对于开拓知识视野，营造读书风气，影响甚巨，声名斐然，遗响至今不绝。

1000 多年以前，南朝齐梁学者钟嵘在《诗品》中以“照烛三才，晖丽万有”来指说天地人间的广博万物。今天，我们全国各地的数十家出版发行单位与数千名作者以高度的历史责任感，联袂推出《中华万有文库》，并向社会各界读者，特别是青少年读者做出承诺：

传播万物百科知识，营造有益成功文库。

我们之所以沿用《万有文库》旧名，并非意图掠美。首先，表明一个信念：承继中国出版界重视文化积累、造福社会、传播知识的优秀传统，为前贤旧事翻演新曲，把旧时代里已经非常出色的事情在新时代里再做出个锦上添花。其次，表明我们这套丛书体系与内容的鲜明特点。经过反复论证，我们决定针对中小学生正在提倡素质教育的需要和农村、厂矿、部队基层青年在提高文化与科学修养的同时还要提高劳动技能的广泛需要，以当代社会科学与自然科学的基础知识为基本立足点，编纂一套相当于基层小型图书馆应该具备的图书品种数量与知识含量的百科知识丛书。万有的本意是万物。百科知识是人类从自然界万物与社会万象之中得到的最重要的收获。而为表示新旧区别，丛书之名冠以中华。这就是我们这套丛书的缘起与名称的由来。

《中华万有文库》基本按照学科划分卷次，各卷之下按照内容分为若干辑，每一辑大体相当于学科的一个二级分支，各卷辑次不等；各辑子目以类相从，每辑 10 至 20 种不等，每种约 10 万字左右，全书总计约 300 辑 3000 种。《中华万有文库》不仅有传统学科的基本知识，而且注意吸收与介绍相关交叉学科、新兴学科知识；不仅强调学科知识的基础性与系统性，而且注重针对读者的年龄特点、知识结构与阅读兴趣而保持通俗性和趣味性；不仅着眼于帮助读者提高文化素质与科学修养，而且还注重帮助读者提高社会生存能力与劳动技能。

每个时代，图书最大的读者群是 10 至 20 岁左右的青少年。每个时代能够影响深远的图书，是那些可以满足社会需要，具有时代特点，在最大的读者群中启蒙混沌、传播知识、陶冶情操、树立信念的优秀图书。我们相信，只要我们老老实实地做下去，经过几个甚至更为漫长的寒暑更迭，将会有数以百万计的青少年读者通过《中华万有文库》而打开眼界，获取知识；《中华万有文库》将会在他们成长的道路上留下鲜明的痕迹，伴随他们一同走向未来，抵达成功的彼岸。

天高鸟飞，海阔鱼跃。万物霜天，凭知识力量，竞取成功，争得自由。在现代社会中，任何人都没有任何理由拒绝为了获取力量而读书。这是《中华万有文库》编纂者送给每一位本书读者的忠告。

追求完美固然是我们的愿望，但是如同世间只有相对完善一样，《中华万有文库》卷帙庞大，子目繁多，难免萧兰并撷，珉玉杂陈。这些不如人意之处，尚盼大家幸以教之。我们虚心以待。是为序。

《中华万有文库》编委会

数据库系统

第一章 绪论

在远古时代，人们就从现实世界中抽象出了数的概念，从原始的结绳记事，到罗马人、阿拉伯人发明了今天仍在使用的数字，人们渐渐学会用数字来描述现实世界里的事物，用数据运算表示现实世界的变化。随着人类文明的进步，社会活动的更加活跃，数据运算越来越频繁，越来越复杂，人类从而又有了利用机器实现运算的机械化、自动化的需求。中国古代的算盘和西方 19 世纪的机械计算装置反映了人类的这种追求。自上个世纪末以来，由于工业革命带来的对社会的巨大推动作用，社会呈现加速发展的趋势。本世纪 40 年代，逐步兴起的电子技术，以及为了应付二战战术数据处理的需要，人们开始了对电子自动工具的研究。1946 年，第一台电子计算机诞生了。从此，人类在对数据处理能力上，有了质的飞跃。50 年来，计算机以人们始料不及的速度发展着，以让人眼花缭乱的强大功能迅速渗透到社会的各行业各场合，并且仍在不断地以新面目面向世人。

计算机的计算能力和存储量，几十年来有了翻天覆地的变化，这些变化导致了计算机的作用从单纯的数字计算演变为对电子数据的处理上来。这种转变是一个划时代的转折，它使计算机从少数的天才科学家手中的怪物转变为广大科技人员和管理人员工作的有力工具和得力助手，它为推动信息化社会转变起到了决定性的作用。

1.信息、数据和数据处理

计算机处理的电子数据，来自于现实世界的信息。信息是现实世界在人脑中的抽象反映，是通过人的感官感知出来并经过人脑的加工而形成的反映现实世界中事物的概念。这里所说的“ 事物 ”不仅是那些看得见、摸得着的物体，而且也包括那些不可触及的抽象概念，如产量、质量、速度等。因此信息可以看作是现实世界的真实反映。信息不仅为人们所认识和理解，而且能够把它作为知识来进行推理、加工和传播，从而达到认识世界、改造世界的目的。

在用计算机处理信息的时候，要将信息转比为计算机可以识别的符号，也就是数据。数据是表示信息的一种手段。比如对一本书的认识——信息，可以在计算机中表示为标题、作者、开本、价格、出版社、提要和全文等等符号化的数据。

事物——信息——数据，实际上贯穿了三个世界，即现实世界——信息世界——计算机世界。

现实世界存在着各种物体的集合，如某个班集体、书、零件等，这些可以称为事物类，事物类也可以是某种抽象概念的集合，如成绩。这是现实世界中进行管理的基础。每一个事物类都有具体的事物组成，如某班集体的学生，可以是张三、李四等。每一个具体的事物又具有自己的内涵，如张三具有姓名、性别等内涵。事物类、事物、内涵构成三个层次，与事物、内涵相对应的是实体和属体。

在数据世界，即计算机世界中，与三个层次对应的概念分别是文件、记录和字段。三个世界的类比关系可由表 1。

表 1.1 三个世界的类比关系

现实世界	信息世界	数据世界
事物类	实体集	文件
事物	实体	记录
内涵	属性	字段

表 1.2 数据世界的事物表示形式

姓名	性别	出生日期	籍贯	地址
张三	男	1969.11	北京	北京师范大学
李四	男	1966.08	山西	北京大学
…	…	…	…	…

认识到了现实世界中的事物可以表示成计算机世界的数据，计算机就有了进行数据处理的基础。数据处理正是对各种形式的数据进行收集、储存、加工和传播的一系列活动的总和。其目的是从大量的、原始的数据中抽取、推导出对人们有价值的信息，作为行动和决策的依据；是为了借助计算机科学地保存和管理复杂的大量的数据，以便人们能方便而充分地利用这些宝贵的信息资源。比如说，对天文观测的数据处理，可以预报天气的变化；对经济运行中产生的数据处理，可以帮助制定科学的经济政策；当计算机保存并处理了图书馆的书刊数据时，就可以迅速从浩瀚的书海中迅速找到所需的资料。借助于计算机网络，数据处理可以使人们在社会活动中如虎添翼。如在北京可以迅速得知上海的商业行情；从中国可以得到美国的某篇文献的原文。这一切均来自于计算机的对电子数据的处理功能。

现在让我们来思考一下，上文中所述及的计算机完成的种种功能如何才能实现呢？要想对数据进行处理，首先要作的是对数据的分类、组织、储存、索引和维护等。这些工作是数据处理的基本环节，这些工作统称为数据管理。

2.数据管理技术的发展

数据管理的水平是和计算机硬件、软件的发展相适应的，是随着计算机技术的发展而发展的。人们的数据管理技术经历了三个阶段的发展：

·人工管理阶段；

·文件系统阶段；

·数据库系统阶段。

1.人工管理阶段

这一阶段，大致是在 50 年代中期之前。此时计算机技术相对落后。这时的计算机主要用于科学计算。硬件方面，计算机的外存只有磁带、卡片、纸带，没有磁盘等直接存取的存储设备，存储量非常小；软件方面，没有操作系统，没有高级语言，数据处理的方式是批处理，也即机器一次处理一批数据，直到运算完成为止，然后才能进行另外一批数据的处理，中间不能被打断，原因是此时的外存如磁带、卡片等只能顺序输入。

这一阶段数据管理的特点是：

（1）数据不保存。在需要计算时，利用卡片、纸带等将数据输入，经过

运算得到运算结果，数据处理的过程就结束了。

（2）数据不能独立。数据是作为输入程序的组成部分，即程序和数据是一个不可分隔的整体，数据和程序同时提供给计算机运算使用。对数据进行管理，就像现在的操作系统可以以目录、文件的形式管理数据。程序员不仅要知道数据的逻辑结构，也要规定数据的物理结构，程序员对存储结构，存取方法及输入输出的格式有绝对的控制权，要修改数据必须修改程序。要对100 组数据进行同样的运算，就要给计算机输入 100 个独立的程序，因为数据无法独立存在。

（3）这一时期，尚没有文件的概念。数据的组织完全由程序员自行设计。即使人们发现了这样作的弊病，也无可奈何。因为此时计算机的外存能力是很弱的。

（4）数据是面向应用的。一组数据对应一个程序。不同应用的数据之间是相互独立、彼此无关的，即使两个不同应用涉及到相同的数据，也必须各自定义，无法相互利用，互相参照。数据不但高度冗余，而且不能共享。

综上所说，所以有人也称这一数据管理阶段为无管理阶段。

2.文件系统阶段

从 50 年代后期到 60 年代中期，数据管理发展到文件系统阶段。此时的计算机不仅用于科学计算，还大量用于管理。外存储器有了磁盘等直接存取的存储设备。在软件方面，操作系统中已有了专门的管理数据软件，称为文件系统。从处理方式上讲，不仅有了文件批处理，而且能够联机实时处理，联机实时处理是指在需要的时候随时从存储设备中查询、修改或更新，因为操作系统的文件管理功能提供了这种可能。这一时期的特点是：

（1）数据长期保留。数据可以长期保留在外存上反复处理，即可以经常有查询、修改和删除等操作。所以计算机大量用于数据处理。

（2）数据的独立性。由于有了操作系统，利用文件系统进行专门的数据管理，使得程序员可以集中精力在算法设计上，而不必过多地考虑细节。比如要保存数据时，只需给出保存指令，而不必所有的程序员都还要精心设计一套程序，控制计算机物理地实现保存数据。在读取数据时，只要给出文件名，而不必知道文件的具体的存放地址。文件的逻辑结构和物理存储结构由系统进行转换，程序与数据有了一定的独立性。数据的改变不一定要引起程序的改变。保存的文件中有 100 条记录，使用某一个查询程序。当文件中有1000 条记录时，仍然使用保留的这一个查询程序。

（3）可以实时处理。由于有了直接存取设备，也有了索引文件、链接存取文件、直接存取文件等，所以既可以采用顺序批处理，也可以采用实时处理方式。数据的存取以记录为基本单位。

上述各点都比第一阶段有了很大的改进。但这种方法仍有很多缺点，主要是：

（1）数据冗余大。当不同的应用程序所需的数据有部分相同时，仍需建立各自的独立数据文件，而不能共享相同的数据。因此，数据冗余大，空间浪费严重。并且相同的数据重复存放，各自管理，当相同部分的数据需要修改时比较麻烦，稍有不慎，就造成数据的不一致。比如，学籍管理需要建立包括学生的姓名、班级、学号等数据的文件。这种逻辑结构和学生成绩管理所需的数据结构是不同的。在学生成绩管理系统中，进行学生成绩排列和统计，程序需要建立自己的文件，除了特有的语文成绩、数学成绩、平均成绩

等数据外，还要有姓名、班级等与学籍管理系统的数据文件相同的数据。数据冗余是显而易见的，此外当有学生转学走或转来时，两个文件都要修改。否则，就会出现有某个学生的成绩，却没有该学生的学籍的情况，反之亦然。如果系统庞大，则会牵一发而动全身，一个微小的变动引起一连串的变动，利用计算机管理的规模越大，问题就越多。常常发生实际情况是这样，而从计算机中得到的信息却是另一回事的事件。

（2）数据和程序缺乏足够的独立性。文件中的数据是面向特定的应用的，文件之间是孤立的。不能反映现实世界事物之间的内在联系。在上面的学籍文件与成绩文件之间没有任何的联系，计算机无法知道两个文件中的哪两条记录是针对同一个人的。要对系统进行功能的改变是很困难的。如在上面的例于中，要将学籍管理和成绩管理从两个应用合并成一个应用中，则需要修改原来的某一个数据文件的结构，增加新的字段，还需要修改程序，后果就是浪费时间和重复工作。此外，应用程序所用的高级语言的改变，也将影响到文件的数据结构。比如 BASIC 语言生成的文件，COBOL 语言就无法如同是自己的语言生成的文件一样顺利地使用。总之数据和程序之间缺乏足够的独立性是文件系统的一个大问题。

文件管理系统在数据量相当庞大的情况下，已经不能满足需要。美国在 60 年代进行阿波罗计划的研究。阿波罗飞船由约 200 万个零部件组成。分散在世界各地制造。为了掌握计划进度及协调工程进展，阿波罗计划的主要合约者罗克威尔（Rockwell）公司曾研制了一个计算机零件管理系统。系统共用了 18 盘磁带，虽然可以工作，但效率极低，维护困难。18 盘磁带中 60%是冗余数据。这个系统一度成为实现阿波罗计划的严重障碍。应用的需要推动了技术的发展。文件管理系统面对大量数据时的困境促使人们去研究新的数据管理技术，数据库技术应运而生了！例如，最早的数据库管理系统之一 IMS 就是上述的罗克威尔（rockwell）公司在实现阿波罗计划中与 IBM 公司合作开发的，从而保证了阿波罗飞船 1969 年顺利登月。

3.数据库系统阶段

从 60 年代后期开始，数据管理进入数据库系统阶段。这一时期用计算机管理的规模日益庞大，应用越来越广泛，数据量急剧增长，数据要求共享的呼声越来越强。这种共享的含义是多种应用、多种语言互相覆盖地共享数据集合。此时的计算机有了大容量磁盘，计算能力也非常强。硬件价格下降，编制软件和维护软件的费用相对在增加。联机实时处理的要求更多，并开始提出和考虑并行处理。

在这样的背景下，数据管理技术进入数据库系统阶段。

现实世界是复杂的，反映现实世界的各类数据之间必然存在错综复杂的联系。为反映这种复杂的数据结构，让数据资源能为多种应用需要服务，并为多个用户所共享，同时为让用户能更方便地使用这些数据资源，在计算机科学中，逐渐形成了数据库技术这一独立分支。计算机中的数据及数据的管理统一由数据库系统来完成。

数据库系统的目标是解决数据冗余问题，实现数据独立性，实现数据共享并解决由于数据共享而带来的数据完整性、安全性及并发控制等一系列问题。为实现这一目标，数据库的运行必须有一个软件系统来控制，这个系统软件称为数据库管理系统（Database Management System，DBMS）。数据库管理系统将程序员进一步解脱出来，就像当初操作系统将程序员从直接控制

物理读写中解脱出来一样。程序员此时不需要再考虑数据中的数据是不是因为改动而造成不一致，也不用担心由于应用功能的扩充，而导致程序重写，数据结构重新变动。在这一阶段，数据管理具有下面的特点，这些特点正是数据库的改进之处：

（1）数据结构不是面向单一的应用，而是面向全组织。仍以学校管理为例，要想避免数据冗余和数据程序之间的依赖性，就要将学生学籍及成绩两类不同的数据之间彼此建立关系。如图 1.1。

当需要增加新的应用，比如学生的体质状况管理，则只要再增加新的联系。

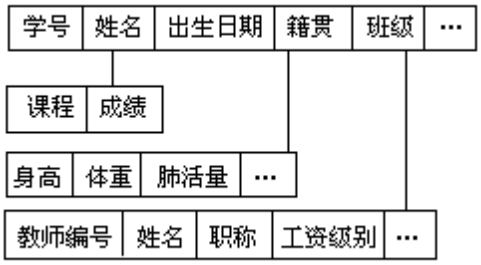


图 1.1 关系化的数据

这种思想只是数据库方法的雏形，它从文件内部的记录的结构比，扩大到不同的文件记录之间建立一种联系。但是它还有局限性，因为它还是从应用的角度去看待数据，还应进一步从整个组织的数据结构考虑。假设所考虑的这个组织——学校，就还应该有教师人事信息、教务信息、教学关系等。不同应用考虑的是整个数据集合的某个有用的子集。整个组织的数据是结构比的。这样描述数据时不仅描述数据本身，还有描述数据之间的联系。

数据的结构化是数据库主要特征之一。这是数库与文件系统的根本区别。至于这种结构化是如何实现的，则与数据库系统采用的数据模型有关，后面会有较详细的描述。

2.数据冗余小，易扩充。数据库从整体的观点来看待和描述数据，数据不再是面向某一应用，而是面向整个系统。这样就减小了数据的冗余，节约存储空间，缩短存取时间，避免数据之间的不相容和不一致。对数据库的应用可以很灵活，面向不同的应用，存取相应的数据库的子集。当应用需求改变或增加时，只要重新选择数据子集或者加上一部分数据，便可以满足更多更新的要求，也就是保证了系统的易扩充性。

（3）数据独立于程序。数据库提供数据的存储结构与逻辑结构之间的映象或转换功能，使得当数据的物理存储结构改变时，数据的逻辑结构可以不变，从而程序也不用改变。这就是数据与程序的物理独立性。也就是说，程序面向逻辑数据结构，不去考虑物理的数据存放形式。数据库可以保证数据的物理改变不引起逻辑结构的改变。

数据库还提供了数据的总体逻辑结构与某类应用所涉及的局部逻辑结构之间的映象或转换功能。当总体的逻辑结构改变时，局部逻辑结构可以通过这种映象的转换保持不变，从而程序也不用改变。这就是数据与程序的逻辑独立性。举例来讲，在进行学生成绩管理时，姓名等数据来自于数据的学籍部分，成绩来自于数据的成绩部分，经过映象组成局部的学生成绩，由数据库维持这种映象。当总体的逻辑结构改变时，比如学籍和成绩数据的结构发

生了变化，数据库为这种改变建立一种新的映象，就可以保证局部数据——学生数据的逻辑结构不变，程序是面向这个局部数据的，所以程序就无需改变。

（4）统一的数据管理功能，包括数据的安全性控制、数据的完整性控制及并发控制。

数据库是多用户共享的数据资源。对数据库的使用经常是并发的。为保证数据的安全可靠和正确有效，数据库管理系统必须提供一定的功能来保证。

数据库的安全性是指防治非法用户的非法使用数据库而提供的保护。比如，不是学校的成员不允许使用学生管理系统，学生允许读取成绩但不允许修改成绩等。

数据的完整性是指数据的正确性和兼容性。数据库管理系统必须保证数据库的数据满足规定的约束条件，常见的有对数据值的约束条件。比如在建立上面的例子中的数据库时，数据库管理系统必须保证输入的成绩值大于0，否则，系统发出警告。

数据的并发控制是多用户共享数据库必须解决的问题。要说明并发操作对数据的影响，必须首先明确，数据库是保存在外存中的数据资源，而用户对数据库的操作是先读入内存操作，修改数据时，是在内存在修改读入的数据复本，然后再将这个复本写回到处存的数据库中，实现物理的改变。比如，某学生的语文和数学的成绩都有输入错误，语文老师和数学老师同时进行修改。操作流程如图 1.2。

从图中可以看出错误的原因。所以数据库管理系统对数据的并发控制要有一定的限制。数据库管理系统对上述各个方面均提供有效的管理，进一步解放了程序员。

由于数据库的这些特点，它的出现使信息系统的研制从围绕加工数据的程序为中心转变到围绕共享的数据库来进行。便于数据的集中管理，也提高了程序设计和维护的效率。提高了数据的利用率和可靠性。当今的大型信息管理系统均是以数据库为核心的。数据库系统是计算机应用中的一个重要阵地。

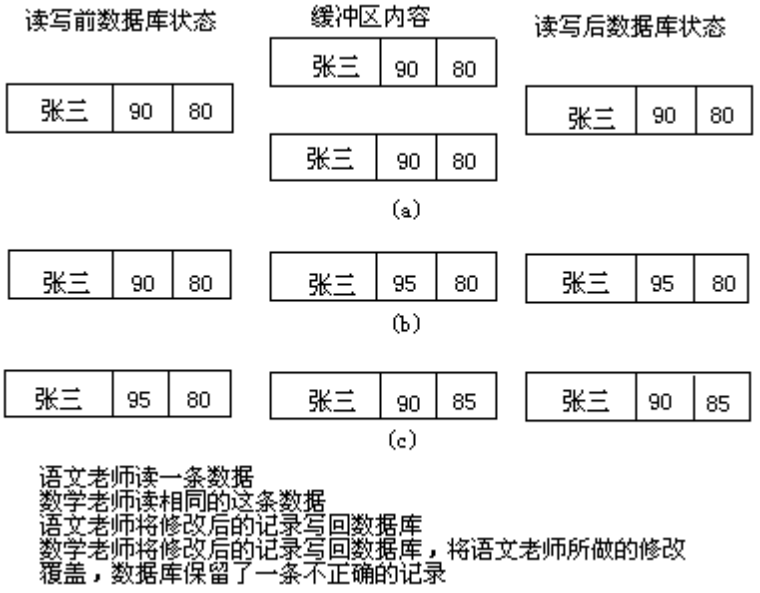


图 1.2 无并发控制的读写操作

总之，数据库技术正是研究如何科学地组织和储存数据，如何高效地获取和处理数据。数据库技术是到目前为止发展成熟的数据管理的最新技术。它的发展趋势和最新进展在第七章中讲述。

第二章 数据库系统

1.数据库系统

本书的第一、二章讲述的内容为数据库的基本理论。需要说明的是，我们常见的个人微机平台的数据库管理系统，如 FOXBASE，DBASE，严格说来，尚不能叫做真正的数据库系统。如果拿它的功能来对照理解我们在这里讲述的数据库理论，会发现这些系统只是实现了数据库系统的部分功能，比如不能自动维持数据的完整性和一致性，不能自动进行并发控制，而是需要由程序员控制等。原因是它们管理的是小型的数据库，所以只提供了部分数据库管理的功能。在理解数据库理论时，完全与这样的数据库软件相对照，有时是不完全吻合的。

1.数据库的体系结构

尽管实际的数据库系统的商业产品多种多样：支持不同的数据模型，使用不同的数据库语言，建立在不同操作系统平台之上，但是绝大多数数据库系统在总的体系结构上都具有三级模式的结构特征。三级模式指的是外模式、模式和内模式。

模式描述的是数据的全局逻辑结构，外模式涉及的是数据的局部的逻辑结构，通常是模式的子集。内模式是数据在数据库内的物理存放方式和结构。

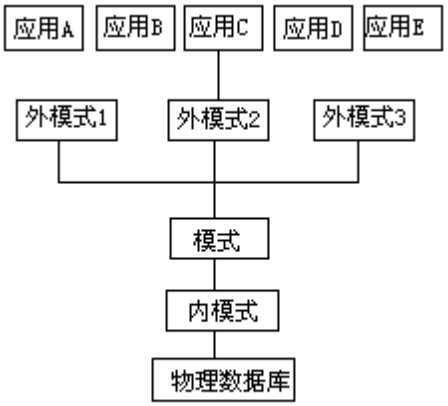


图2.1 数据库的三级模式

数据库系统的三级模式是对数据的三个抽象级别。用户面对的是外模式数据，数据库管理系统（DBMS）负责数据的具体组织，数据库在三级模式之间提供了两层映象和转换：

·外模式——模式

·模式——内模式

这两层映象保证了数据库的数据独立于程序。这两层映象分别实现了数据的逻辑独立性和物理独立性。

(1) 外模式。亦称子模式或用户模式。是数据库的用户看到的数据视图，是与某一种应用有关的数据的逻辑表示。不同用户的外模式可以互相覆盖，同一外模式可以为某一用户的多个应用所启用，一个应用只能启用一个外模式。不同的用户其需求不同，看待数据的方式可以不同，对数据保密的要求可以不同，完成应用使用的程序设计语言也可以不同。比如，学生成绩管理

和学籍管理可以用不同的程序设计语言来建立应用。对同一个学生成绩数据，老师可以写，学生却只能读。因此不同用户的外模式的描述是不同的。

（2）模式。模式是数据库中全部数据的一个逻辑表述，既要定义数据的名字、数据类型、大小，还要说明数据之间的关系，数据的安全性、完整性要求等。既要定义记录的结构，还要定义数据项之间的联系。

（3）内模式。它是数据库的数据在物理磁盘中如何保存的描述。用来定义数据的存储方式和物理结构。

当模式改变时，外模式——模式的映象要改变，当数据库的存储结构改变时，模式——内模式的映象也要改变。所有的改变，对最终用户来讲无需关心，第一种改变，开发人员要了解这种改变，建立新的映象，以保证外模式不变，建立在外模式上的程序不必改变。

2.数据库系统

要了解数据库，就要对数据库有一个全面的认识。一个完整的数据库系统是由计算机系统、数据库、数据库管理系统、应用程序集合及数据库管理人员组成的。

（1）计算机系统。计算机系统指的是进行数据管理的计算机硬件资源和基本软件资源。硬件资源就是计算机中央处理器、大容量内存和外存以及必要的输入输出设备。现在用于数据库管理的计算机有大、中、小、微各种机型，还有工作站级的计算机。一般来讲，在面向多用户的系统中，用于中心管理数据库的面向数据库管理人员的，和用于查询面向用户的终端计算机是不同档次的。此外，在计算机系统中还包含软件资源，比如操作系统、网络管理软件以及下面要讲的数据库管理系统和应用程序。

（2）数据库。数据库正是数据库系统要管理的对象，通过前面的说明，我们知道它们是以一定的组织方式存储在一起的、能为多用户共享的、与应用程序彼此独立的相合关联的数据集合。在来自于不同厂家的数据库系统中，数据库的物理存储形式是不同的。在 XBASE 型微机数据库中，并没有一个叫做数据库的实体，可以见到的只是组成数据库的一个个数据文件和索引文件，索引文件需要用户来更新。而在 AC-CESS 中，数据库是以一个*.MDB 文件的形式存在，没有独立的数据文件和索引文件，数据库中有的是一个个的表，大致类似于 XBASE 中的数据文件。索引是数据库自动维护的，不需要用户自动更新。其他大型数据库系统的数据库则还可能有其他的物理形式存在。读者在学习使用数据库的过程中，会逐渐对数据库有更感性的认识。

（3）数据库管理系统（DBMS）。用户一般不直接加工或使用数据库中的数据，而必须通过数据库管理系统。DBMS 的主要功能是维持数据库系统的正常活动，接受并响应用户对数据库的一切访问要求，包括建立和删除数据文件、检索、统计、修改和组织数据库中的及为用户提供对数据库的维护手段等。通过使用 DBMS，用户可以逻辑地、抽象地处理数据，而不必关心这些数据在计算机中的具体存放方式，以及计算机处理数据的过程细节。这样，把一切处理数据的具体而繁杂的工作交给 DBMS 去完成。就好像在计算机的发展过程中，操作系统的出现，解脱了用户，不必关心数据的实际存放和读取，而只需给出文件名和路径一样。

DBMS 是一个以统一的方式管理、维护数据库的软件的集合。具体来说，就是厂家发行的用于数据库管理的系统软件，如 FOXBASE，ACCESS，SYBASE

等。DBMS 在操作系统的支持与控制下运行，DBMS 完成三部分功能：

1）语言处理功能。DBMS 必须能理解用户的需求来描述数据，比如数据之间的联系，数据的完整性约束等；还要能理解用户的操纵数据的请求，比如，用户要插入还是要检索？有的数据库管理系统提供自己的语言，比如，用 FOXBASE 编程，使用的就是 FOXBASE 这个数据库管理系统提供的语言。也有的是利用某种程序设计语言，这种语言提供数据库操纵语句。如 VB、VC 等。

2）系统运行控制功能。包括系统总控；并发、数据安全性、检查数据完整性等控制程序；数据访问、通讯程序。

3）系统维护功能，包括数据备份（转储）、作日志、系统自动恢复等功能。微机平台的简单的 DBMS 不具备上面的全部功能；

对一些微机平台的 DBMS 而言，如 FOXBASE、FOXPRO 等，因为它提供的语言很简单，用户可以很容易的掌握，可以直接使用数据库管理系统，来操纵数据库。但对一些大型的复杂的数据库管理系统而言，用户不能够直接操作 DBMS 来管理数据，一般还要由程序设计人员进一步开发出应用程序，来更方便地满足用户的需求。包括设计出更容易使用的友好界面上，用户在上面输入数据、输入查询要求、输出处理结果等。这就是下面要讲的数据库系统的应用程序部分。

（4）应用程序集合及数据库管理人员。应用程序是计算机专业人员开发的面向最终用户的软件。它是在 DBMS 基础上实现的。也就是说，数据库应用程序不能脱离数据库管理系统环境。要先启动数据库管理系统，然后再启动应用程序。它的使用，完全是为了方便用户。因为对各行各业的用户而言，学会控制计算机的算法语言，掌握数据库的原理，维护数据库的正常运转，是困难的和不现实的。而应用程序，一般具有友好的界面，便于用户表达自己的需求。比如，要想查询自己的成绩，应用程序可能设计成引导用户用鼠标或键盘选择查询点，姓名还是学号，然后程序提示在合适的位置用键盘输入名字或学号。程序去与 DBMS 打交道，完成查询的过程，并将查询结果显示在屏幕上。事实上，有些应用程序使用起来比这还要简单，比如，超级市场的收银员，只是将光笔在商品上一扫，剩下的工作，包括从库存中减分一件，在营业额中增加一笔收入，将客户的商品及应付款打印出来等一系列工作，全部由应用程序完成。

在一个安全性较高的大型的数据库管理系统中，比如金融部门等，必须有专门的管理人员，随时作监视应用程序、维护硬件设备、定时备份等工作。他们也是一个数据库管理系统中不可缺少的一个重要部分。

整个数据库系统可用图 2.2 表示：

图 2.2 数据库系统

图2.2 数据库系统

3.数据库技术发展

在我们今天的生活中，数据库技术的应用非常广泛。计算机的商业应用几乎都与数据库有关。小到一个通讯录的管理，大到银行业务的处理，都是数据库在发挥着作用。特别是因特网的发展，更使数据库克服了时空的限制，使得人人都可以得到它的服务，它的影响力得到了进一步的扩大，我们今天可以从网上获得信息，实际上都是由于有无数个数据库系统在工作。数据库技术推动了信息社会的到来，可以称作是信息社会的坚硬基石。

正是数据库产生的深远影响力，也使得数据库得到了更迅猛的发展。在数据库技术的发展过程中，先后出现了基于层次模型和网状模型的数据库管理系统，确定并建立了数据库系统的许多概念、方法和技术。1970 年，IBM 公司圣·约瑟（San Jose）研究实验室的研究员 E.F.科德（Codd）发表了题为“大型共享数据库数据的关系模型”的论文，论述了数据库的关系模型。开创了数据库关系方法和关系数据理论的研究，为数据库技术奠定了理论基础。E.F.科德是一个不应被遗忘的人物，由于他的杰出贡献，他于 1981 年获 ACM 图灵奖。这之后，数据库技术又有了巨大的发展，出现了许多商品数据库系统。这些商用系统的运行使数据库技术日益广泛地应用到企业管理、交通运输、情报检索、军事指挥、政府管理和付诸决策等各个方面，深入到人类生产和生活的各个领域，数据库技术成为实现和优化信息系统的基本技术。在计算机领域，人们称 70 年代为“数据库时代”，关系方法的理论研究和软件系统的研制获得了很大成功。80 年代，几乎所有的新开发的系统均是关系型的。随着微机的崛起，微机数据库管理系统也越来越丰富，性能越来越好，功能越来越强，应用遍及各个领域。

今天，在数据库系统软件的研制方面，除了将数据库应用于管理，还开始应用到了工程设计、图形图象和声音等多媒体处理、自动控制、计算机辅助设计、统计等新的应用领域。这些领域所涉及的数据和管理领域中数据的格式有极大的区别。

数据库技术研究不仅涉及应用系统的设计方法，而且涉及数据库系统的模型、实现技术等方面。分布式数据库、面向对象的数据库系统、多媒体数据库系统、数据仓库等方向的研究迅速兴起并取得了巨大的成果。

在数据库的设计方法，设计工具和理论的研究，计算机辅助数据库设计方法和软件系统的研究等各方面，也取得了很大的进展。数据库理论，包括关系的规范化理论，关系数据理论是研究的焦点。相邻学科的发展，不可避

免地推动着数据库技术的发展，与人工智能等这些新技术相互融合，形成了数据库和逻辑，逻辑演绎和知识推理等理论研究，以及演绎数据库、知识库系统的研制等新的研究方向。

总之，作为计算机技术的一支重要分支，数据库技术的发展会越来越迅速，作用会越来越大，会帮助人们在社会生活中解决更多的难题。

2.数据模型

现实世界五彩缤纷，目前任何一种科学技术手段都还不能将现实世界按原样进行复制并管理起来。这样，计算机在处理现实世界的信息时，只能根据需要，选择某个局部世界，并抽取这个局部世界的主要特征，特别是数据之间的结构关系，构造一个能反映这个局部世界的数据模型。在数据库领域，目前广泛应用的数据模型主要有三类：

·层次模型；

·网状模型；

·关系模型。

（1）层次模型。层次模型是将现实世界的实体集彼此之间抽象成一种自上而下的层次关系。例如一个学校的学生组织情况，如图 2.3。

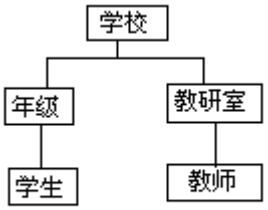


图2.3　层次模型

在层次模型中，每个构造单元称为记录型。在图中，学校、年级、学生就是记录型。一个上层记录型对应下层一个或多个记录型。每个记录型有一个或多个记录值，上层记录值对应下层一个或多个记录植，下层记录植只能对应上层一个记录值。如中学记录型有“光明中学”一个记录值，它对应下层记录型“年级”的“初一年级”“初二年级”等多个记录值。

（2）网状模型。对于现实世界的另外一些问题，它们就不符合层次模型的关系，层次模型就不能正确有效地反映。例如在讨论学校中教师、学生和开设课程这类问题时，可以构造出图 2.4 的模型。

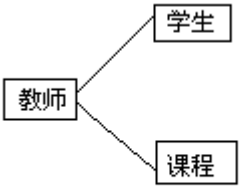


图2.4 网状模型

在这个模型中，教师、学生和课程彼此都有联系。这种模型称为网状模型。网状模型每个记录型对应一个或多个其他记录型，每个记录型也存在一个或多个记录值，每个记录值可能对于一个或多个其他记录型的记录值。

网状模型和层次模型都是成功的数据模型，基于这些模型构造了一些成

功的数据库管理系统。但是，这两种模型共同的缺点是用户在处理数据库中的数据时，必须非常清楚数据之间的网状（或层次）联系，实现较困难。如果我们把这种数据操作看成是在数据库的数据海洋中航行的话，用户必须时刻注意自己的位置和航向。所以基于这两类数据模型的数据库管理系统都称为“导航”式系统。当用户的需求发生变化，就可能修改数据模型结构，严重时危机整个系统。比如在上面的例子中，当管理学生时，层次关系是可以反映真实情况的，但如果将教师和课程加入到系统中，就必须将数据模型修改成网状的，随之整个系统都要改变。

（3）关系模型。现在使用更为普遍的是关系模型。现在的数据库管理系统几乎都是支持关系模型的。数据库领域的研究工作，也大都集中在关系方法中。在关系模型中，现实世界的数据组织成一些二维表格，在关系模型中，这些表格称为关系，用户对数据的操作抽象为对关系的操作。如图 2.5 所示。

图2.5关系模型

学号	姓名	成绩	
		语文	数学
01	张三	90	80
02	李四	80	88
03	五王	70	80

图 2.6 不规范的关系

每个关系，也就是一张表，有一个关系名；从纵向看，表中的一行称为一个元组，每行数据也称为一个记录；关系在每个横向上由若干个数据项组成，称为属性或字段。此外，表中有一个或几个属性，它们的值唯一地确定一个元组，这样的属性或属性组称为主码。

关系模型的概念简单清楚，所有数据及其关系均反映在关系——二维表上，不像层次模型或网状模型，记录与记录之间的联系非常复杂。关系模型的关系要求为规范化的，即表中不能有表，每一个数据项不能再分。在关系模型中对数据的操作，都简化为同样的表操作，用户的要求统一变为从原来的表中得出一个需要的新表。用户只需说明“找什么”，而不需要说明“怎么找”，提高了操作效率。关系模型中的数据操作是集合操作，有严格的数学基础，并在此基础上发展了关系数据理论，所以，关系模型在诞生以后，成为发展迅速，最受欢迎的数据模型。

3.网状数据库

采用了网状数据模型的数据库系统就是网状数据库系统。网状数据库的典型代表是 DBTG 系统，它不是实际机器的软件系统，但是它所提出的基本概念、方法和技术具有普遍意义。它对于网状数据库的研制和发展起了重大的

作用。

DBTG 模型的数据结构是由数据项、记录、系等对象组成的网状结构，其中：

·数据项（Data ltem）是命名的最小数据单位。

·记录（Record）是数据项的有序集合，表示描述的实体。

·系（set）表示记录之间一对多的联系。

DBTG 系统用记录的概念描述实体，用系的概念描述实体之间一对多的联系。系是 DBTG 中一个重要概念。在 DBTG 中，系既是实体之间逻辑联系的表示，又是存取数据库数据库时可遵循的存取路径。因此，用存取路径来表示记录之间的联系是 DBTG 系统的基本特点。

DBTG 系统提供了子模式 DDL（Data Defination Lan-guage 数据定义语言），模式 DDL DSDL（Data Storage De-scription Language 数据存储描述语言）、DML（Data Manipu-lation Language 数据操纵语言）四种语言。

模式 DDL 描述数据库的整体数据结构和完整性约束条件等。它独立于任何高级程序数据语言。模式 DDL 中还有描述存取路径和存储安排的内容。它相当于模式到内模式的映象。

子模式 DDL 描述用户所涉及的数据结构和完整性约束条件。子模式 DDL 是面向某一程序设计语言的。

DSDL 定义数据库的存储模式。存储模式不会影响应用程序的执行结果而只会影响运行效率。

DML 定义对模式和子模式所描述的网状数据库中数据在记录级（Record—Level）的操作集合。它是程序员用来检索和更新数据库的语言。

前三种语言是由 DBA（Database Administrator 数据库管理员）建立，用以定义模式字模式和存储模式，用户不必了解。用户只需了解与自己有关的子模式。子模式是用户的数据视图，DML 是用户存取数据库数据的工具。DBTG 的 DML 语句是一组宏命令，它不是独立的查询语言，程序员必须使用某种高级语言和嵌入高级语言的 DML 语句编写应用程序完成对数据库的操作。也就是说用户需编写程序使用 DML 实现对数据库的存取。

随着应用环境的扩大，用户的数据视图将变得越来越复杂和不够清晰，加上用户对数据库的存取必须沿着存取路径到达目标数据库，这就必须随时注意数据库在各个范围中的当前值，从而加重了用户的负担。这是 DBTG 系统的主要缺陷。

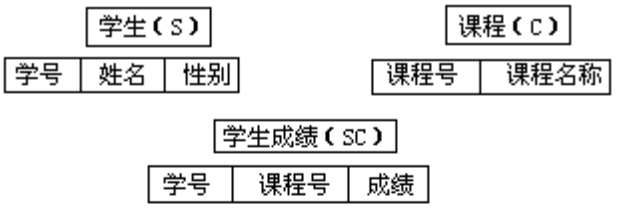
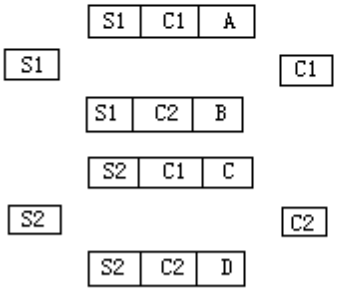


图2．6　在学生S、课程C、学生成绩SC三个记录型之间形成了两个系S-SC、C-SC



S-SC系的系值　C-SC系的系值

图2．7系值

4.层次数据库

支持层次模型的数据库系统为层次数据库系统。典型的层次型数据库有IBM 公司研制的 IMS 系统。

IMS 中数据不可分隔的最小单位是字段（Field），若干字段组成片断（Segment）。片断是 IMS 中应用程序对数据库访问的基本单位。也就是说，IMS 中描述一个实体的是片断，它相当于 DBTG 系统中记录的概念。IMS 中描述实体属性的是字段，相当于 DBTG 系统中记录的数据项。

IMS 的基本数据结构是由若干相关联的片断组成的一个层次结构，或者称为一棵树。一个 IMS 的整体数据模型是若干棵树的集合。

IMS 中把片断型的层次序列结构称为 PDBR（PhysicalData Base Record TyPe）；记为 PDBR 型。一个根片断及其后代片断值构成此 PDBR 型的一个值，称为一个数据库记录。一个物理数据库 PDB 就是一个 PDBR 型的全部值的有序集。例如，一个大学有 16 个系，则物理数据库就是以 16 个系为根片断值的 16 个数据库记录组成。IMS 中一个数据库是若干个 PDB 的集合。例如，一个学校的数据库可以有关于本科生的 PDB，有关于研究生的 PDB，有关于教师的 PDB。

IMS 中每一个物理数据库 PDB 及向存储结构的映象用数据库描述 DBD（Database Desciption 数据库描述）来定义。

用户应用程序所使用的数据的逻辑描述称为程序描述块 PSB（Programe Specification Block），它是一组程序通信块（Programe Communication Block）的集合。PSB 相当于外模式加上有关外模式到模式的映象。

IMS 的数据子语言是 DL/1，IMS 是宿主语言系统，用户把 DL/1 语句嵌入宿主语言，编写应用程序，实现对数据库的存取和对数据的处理。

IMS 的数据库的存储结构反映数据之间的层次关系。实

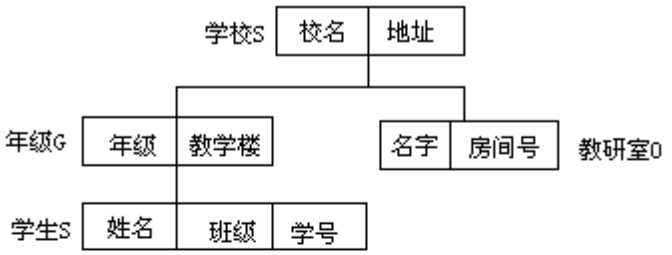


图2．8 层次数据库的型

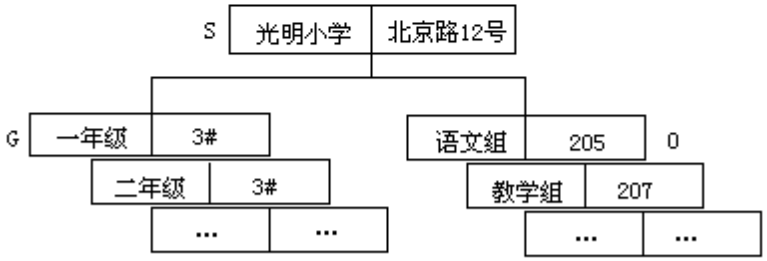


图2．9 层次数据库的一个值

现的方法有邻接法、链接法。

IMS 和 DBTG 都属于格式化模型。它们有很多共同特点，都用存取路径来表示数据之间的联系。用户对数据的存取都必须按照明确定义了的存取路径进行；必须清楚地了解当前的数据库的当前位置；对数据库的操作都是一次一个记录的存取方式；程序和数据都有高度的物理独立性，但逻辑独立性不高。网状和层次型数据库由于目前使用的不广泛，所以，对它们没有进行更详细的描述，读者如有兴趣，可参阅其他书籍。现在流行的是关系型的数据库系统，我们将在后面章节专门论述。

5.数据库保护

人们之所以放弃文件管理系统，转而使用数据库系统，一个很重要的原因就在于它可以保证数据的安全、正确和可靠。数据库的数据保护功能，主要包括数据库的安全性、完整性、并发控制和数据恢复。在目前流行的微机版本的数据库管理系统，在这个方面系统提供的支持较少。

1.安全性

数据安全性是指保护数据库以防止不合法的使用所造成的数据泄漏、更改或破坏。采取的措施主要有：

（1）用户身分审核。一般是系统提供一定的方式让用户出示用户名和口令。通过系统审核合格后，再提供用户数据使用权。

（2）存取控制。对于有权使用数据的用户，还要规定它可以操作的对象，可以是表、索引、视图和字段等。此外，还要规定在这些对象上的操作类型，新建、插入、查找、修改、删除等。用户无法操作未获授权的对象，在已经获得授权的对象上，也不能执行未得到授权的操作。

（3）数据加密。以防止窃听、偷盗及一些恶性违法事件的发生。

2.完整性

数据的完整性是指数据的正确性和相容性。数据库系统要依照完整性约束条件进行完整性检查。以防止数据库中存在不符合语义的数据，防止无效操作及错误结果。完整性约束条件主要有以下几种：

（1）值的约束。数值的取值范围、精度等的规定。比如年龄必须小于100，年份为四位数，精确度为小数点后几位等。

（2）数据之间联系的约束。保证实体完整性和参照完整性。例如维护数据之间的相等以及其他复杂关系，主码非空，学生选课必须是课程关系中的子集，工资级别与工资数额之间的正比关系等情况。

（3）动态约束。数据改变时的约束。如增加工资时，约束新工资值大于旧工资值。

3.并发控制

由于数据库是一个共享资源，所以系统必须提供并发控制功能，保证数据处理的正确和高效，维护数据库的完整性。事务是并发控制的单位，对一个事务的处理称为交易（Transaction）。事物是一个操作序列，事务提交给系统后，要么全部完成，要么全部不作。具体的实现是通过对数据对象的加锁来完成的。事务执行时，对数据加锁，事务撤销后解锁。让我们回忆一下在第一章中举过的例子。在这个例子中，语文老师和数学老师同时修改同一个学生的记录，结果造成了错误的结果。解决的办法是，将从读取学生记录到内存，到修改后写回数据库的过程定义为一个事务，这样一个完整的事务，通过加锁的方法，数据库保证事务执行中间不被打断，要么全部完成，要么全部退回。如果已经有一个老师读取了某条记录准备修改，在他将数据写回数据库之前，不允许其他的读或写操作插入，这样就保证了数据的正确。

4.数据恢复

由于计算机的故障、停电、工作人员的失误等突发事件，可能会造成数据库中的数据错误、数据丢失、事务非正常中断等严重后果，所以数据库的数据恢复功能是不可缺少的。数据恢复功能是指将数据库从某一错误状态中恢复到某一尽可能最近的正确状态。

转储是数据库恢复采用的基本技术。转储是 DBA 定期地将整个数据库复制到磁盘或磁带上保存起来的过程。转储定期地保留数据库的副本。做日志是另一重要手段。日志文件记录了系统的每一次操作。将备份和日志文件相结合，就能够从作备份的时刻开始，根据日志文件恢复作过的每一次操作，从而使损失降低到最小程度。如图 2.10 所示：

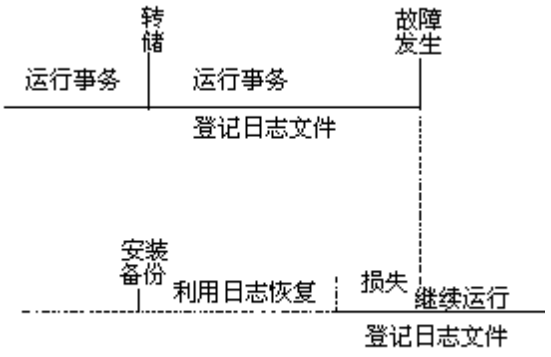


图2．10 利用转储和日志恢复数据库

第三章 关系数据库

1.关系数据语言

1.关系模型

在第一章中我们已经初步认识了关系模型，现在让我们了解一些更多的关于关系模型的知识。

（1）概念。关系模型是建立在集合代数的基础上的。关系可以简单地认为是一个二维表。表的每列对应集合代数中的域的概念，称为属性。表的每行对应元组，即域的笛卡尔积的任意一个元素。元组中的每一个值称为分量。关系中属性的数目称为关系的目。关系是笛卡尔积的子集。下面我们给出两个域的 D1 和 D2：

D1＝姓名＝ {张三，李四}

D2＝年龄＝ {20，24}

它们的笛卡尔积是：

D1×D2＝ {（张三，20），（张三，24），（李四，20），

（李四，24） }

笛卡尔积可表示成二维表，它的子集构成了两目的关系，关系的属性分别为姓名的年龄：

姓名	年龄
张三	20
李四	24

在数据库中我们要求关系的每一个分量必须是不可分的数据项，也即必须是规范化的关系，简称为范式。数据库中的关系具有以下的特性：

· 关系中的每一列的分量是同一类型的数据，来自于同一个域；

· 不同的列可以出自同一个域，每一列称为属性，要给予不同的属性名加以区分。例如：在上面的关系中，添加语文成绩和数学成绩，这两个属性来自于同一个成绩域；

· 列的顺序可以交换；

· 行的顺序可以交换；

· 关系中不能存在完全相同的两行；

· 每一分量不可分。例如，不能存在成绩这样的列，在这个列的下面又分成几个分量。在关系模型中，无论是实体还是实体间的联系均由关系——这一单一的结构类型来表示。下面是有关关系模型的几个概念：

主码：关系中唯一标识一个元组的一个或一组属性，称为主码。

关系模式：关系的描述称为关系模式，由关系名、属性名、属性间数据的依赖关系等组成。

关系数据库：对数据库的描述，即域的定义和域上的关系的集合是数据库的型的概念。数据库某个时刻对应的关系的集合称为数据库的值。

需要指出的是，关系模式是稳定的，而关系，即数据库的值是不断更新的。

（2）关系操作。关系模型决定了关系操作的特点。关系语言的特点是高

度非过程化。操作的对象是关系，操作的结果也是关系，这种操作是集合操作。用户不必关心存取路径，这正式网状和层次模型的缺点。关系模型中，关系操作有关系代数和关系演算两种形式，这两种形式的功能是等价的，一个是代数表示，一个是逻辑表示。关系操作用关系代数表示，常用的有θ选择、投影、θ连接、除、并、交、差等。θ表示＞、＜等比较运算符。

（3）关系模型的完整性。关系模型要求实体完整性、参照完整性和用户定义的完整性。

1）实体完整性，简单地讲就是主码不能为空。这样作的意义是保证关系中不会出现无意义的元组。比如在学生关系中，每个学生都有不同的学号，学生的学号就是主码，实体的完整性就防止了没有学号的学生数据的出现。

2）参照完整性存在于两个关系之间。可以理解为，关系 A 中的属性 a，与关系 B 中的主码 b 相对应，则关系 A 中所有元组的属性 a 的值必须对应关系 B 中某一个元组的主码 b 的值。这样作主要是为防止止无意义的数据的出现和存在。例如，在学生数据和学生的成绩数据之间，成绩关系中的学号不能是学生关系中没有的学号。

3）用户定义的完整性，是用户根据数据库的实际情况制定的数据库的约束条件。当这种约束条件定义好之后，用户就不必用程序检验，而是用关系模型提供的系统的方法去处理。

2.关系数据语言

数据库管理系统（DBMS）的功能有：①定义数据库；②操纵数据库；③管理数据；④建立和维护数据库；⑤数据通讯。它向用户提供的操纵数据库中数据的语言，称为数据操纵语言（DLL），数据库操纵语言必须解决的问题有：

·查询操作；
·插入操作；
·删除操作；
·修改操作；
·控制并发访问操作，以及打开、关闭数据库等操作。

在关系数据库中，关系数据语言具有将数据描述、数据操纵和数据控制合而为一的特征。关系数据库语言主要有下面的功能。

（1）数据定义。包括关系表中每个字段的名字、类型、长度、完整性和安全性定义。

（2）数据操纵。基本操纵功能：查询、插入、删除和修改

输出功能：将操作结果形成一定的形式，如报表在显示器或打印机上打印输出。

简单计算功能能够将检索结果进行简单的加减乘除算述运算，或是求和、平均值、最大和最小值。

（3）数据控制。包括给某个用户授权和从用户收回检索、修改的权利。系统一般用触发器的方法，保持数据完整性，触发器的作用是当用户有某种数据操作时，系统自动执行另外的和这个操作相关联的一系列其他操作。触发器是用户定义的。

数据操纵语言中的查询表达方式是最主要的部分。关系的数据操纵语言主要有关系代数和关系演算两大类。关系代数和关系演算均是抽象的查询语

言，与实际的 DBMS 中实现的实际语言还有一定的距离。在关系数据库中广泛使用并形成标准的是 SQL——结构化查询语言。

3.关系数据库的基本操作

（1）数据查询。垂直查询，又称投影操作，这是一个单目运算，对一个关系或多个关系表，给定字段名，构成新的关系表。

水平查询在给定的关系表中，选取满足某些条件的行构成新的关系，又称“选择操作”。

当上述操作设计多个关系时，需将两个关系合并成一个，如有多个关系，还需将合并结果与第三个合并，依次合并，形成一个关系表。

所以数据查询包括“投影”、“选择”和“合并”三种操作。

（2）数据更新。

1）数据插入。整行数据插入到关系表中。

2）数据修改。修改某些行的某些字段值。

3）数据删除。删除某些行。

2.关系代数

关系代数是关于关系的操作集，是关系数据语言的数学基础。操作的对象为关系，操作结果也是关系。关系操作集，可分为两组。

1.集合运算

这类运算将关系看成元组的集合，是从关系的水平方向即行的角度来进行的。

并运算：并运算的含义是，所有属于关系 R 和关系 S 的元组（行）组成集合 U，称为 R 和 S 的并，去掉重复之元组（行）。记作∪。

差运算：关系 R 中去掉属于关系 S 的元组所剩下的元组集合，称为差运算。记作-。

交运算：既属于关系 R，又属于关系 S 的元组组成的关系，称为交运算。记作∩。

笛卡尔积：两个分别具有 n 目和 m 目的关系 R 和 S，它们的笛卡尔积是一个（n＋m）目的元组，前 n 个分量来自关系 R 的一个元组，后 m 个分量来自关系 S 的一个元组。记作×。

交、差、并运算分别用下表说明：

R

A	B	C
a1	b1	c1
a1	b1	c2
a2	b2	c2

S

A	B	C
a1	b1	c1
a2	b2	c2
a3	b3	c3

R　S

A	B	C
a1	b1	c1
a1	b1	c1
a3	b3	c3
a2	b2	c2

R-S

A	B	C
a1	b1	c1

R　S

A	B	C
a1	b1	c2
a2	b2	c2

关系R和关系S的笛卡尔积表示如下：

R

A	B	C
a1	b1	c1
a2	b2	c2
a3	b3	c3

S

D	E
d1	
d2	

A	B	C	D	E
a1	b1	c1	d1	e1
a1	b1	c1	d2	e2
a2	b2	c2	d1	e1
a2	b2	c2	d2	e2
a3	b3	c3	d1	e1
a3	b3	c3	d2	e2

2.特殊的关系运算

关系运算部件不仅涉及行而且涉及列。

选择：是对关系中的行而言，指在一个指定的关系中按一定的逻辑条件选取若干元组，操作结果仍为一个关系。例如，“ 列出 1980 年以后出生的学生名单 ”，就是要找出那些符合条件的行。选择运算结果表示如下：

姓名	出生日期	性别
张三	1981.3	男
李四	1979.11	男
王五	1983.9	男
赵六	1979.8	男

选择运算结果

姓名	出生日期	性别
张三	1981.3	男
王五	1983.9	男

投影：按照指定的若干属性名及顺序选择列，并去掉重复元组后的所组成的新的关系体。在某些情况下，用户只对某些域感兴趣，比如，一个查询，要求在关系 R 中只查询所有学生的姓名和性别。投影结果如下：

原关系

姓名	出生日期	性别
张三	1981.3	男
李四	1979.11	男
王五	1983.9	男
赵六	1979.8	男

投影运算结果

姓名	性别
张三	男
李四	男
王五	男
赵六	男

连接：当一个查询需要来自两个或多个关系的数据时就要用连接操作。连接是从两个关系的笛卡尔积中选取属性间满足一定条件的元组。相比较的属性是可比的属性。当要满足的条件为相等时，称为等值连接。

自然连接是一种特殊而常用的连接。若关系 R 和 S 具有相同的属性组 B，则自然连接的就是要两个关系中相等的分量必须是相同属性组，而等值连接不必。另外自然连接要在结果中将相同的属性去掉，而等值连接不必。

下图分别表示了关系 R 和 S 的连接、等值连接和自然连接的情况。

关系 R

学号	语文
01	90
02	88
03	80

关系 S

学号	数学
01	90
02	80
03	88
04	90

R 和 S 连接,满足“ 语文<数学 ”

学号	语文	S．学号	数学
02	88	01	90
02	88	04	90
03	80	01	90
03	80	03	88
03	80	04	90

R 和 S 连接，满足“ 语文=数学 ”

学号	语文	S．学号	数学
01	90	01	90
01	90	04	90
02	88	03	88
03	80	02	80

R 与 S 的自然连接

学号	数学	
01	90	90
02	88	80
03	80	88

在上面介绍的这两组运算中，最基础的是并、差、笛卡尔积、投影和选

择，其他的运算均可用这五种基本运算来表达，引进它们并不增加语言的能力。有时一个请求往往都要结合使用全部这三种操作。

关系的数据操纵语言除了使用关系代数还有关系演算。关系演算是以数理逻辑的谓词演算为基础的。关于谓词演算的内容，在此不再细述。

3.结构化查询语言——SQL

SQL（Srtuctured Query Language）语言是 IBM 公司于 1974 年首先实现使用的。由于它功能强大、使用简单、易于掌握，大受计算机界人士的欢迎。并陆续形成了 SQL 语言作为关系型数据库语言的美国国家和国际标准。现在的数据库产品的各个厂家都使自己的产品支持 SQL 语言，也就是说不管出自那个厂家的产品，都有用同样的语言——SQL 去操纵它们的可能，所使用的 SQL 语法是大同小异的。所以我们有必要了解一些关于 SQL 语言的知识。

SQL 语言，作为关系数据库语言，具有很丰富的功能。它的功能包括有数据库定义、查询、控制等。SQL 具有下面的特点：

（1）SQL 能够完成定义关系模式、建立数据库、插入数据、查询数据、更新数据、删除数据、安全性控制等功能。具有集 DDL、DML、DCL 为一体的特点。

SQL 的使用有两种形式，一种是直接用于操作数据库。比如，在一些数据库管理系统中，提供了直接用 SQL 语句操作数据库的功能。还有一种使用方式是嵌入一种程序设计语言中，如常用的开发工具 VB、PB 等。

（2）在使用 SQL 语句时，只需要指出“ 干什么 ”，而无需关心“ 怎么干 ”。用户不必考虑存取路径等问题。该语言是一种高度非过程化的语言。

（3） SQL 语言使用类似于英语的语法，易于使用和看懂。SQL 语言只是使用了有限的几个动词，易于掌握。

1.数据定义功能

有关数据定义功能的 SQL 语句，它们分别用来定义表、定义视图、定义索引、删除表、删除视图、删除索引、修改表结构：

CREATE TABEL　　CREATE VIEW　　CREATE INDEX　DROP TABEL　　DROP VIEW　　DROP IN-DEX　ALTER TABEL

在下面这个例子中，用 SQL 语言定义一个表：

```
CREATE TABEL s（S#CHAR（2）NOT NULL，
SN CHAR（8），
SEXCHAR（2））；
```

执行这条语句后，就在数据库中建立了一个表，有关这个表的数据字典中就保存在了数据库中，可能是以系统表的形式保存。在上面的例子中，S#是学号，NOT NULL 表示不能为空，SN 是姓名，SEX 是性别，数据类型都是字符型，长度分别为 3、8、2 个字节。NULL 表示空值，空值不是 0 或空格，而是不能使用的值，除非在建表时特别指明（如 S#域），否则，任何列可以有 NULL 值。建成的表结构如下：

S

S#（学号）	SN（姓名）	SEX（性别）

其他有关定义的语句不再举例。

2.SQL 语句的数据操纵功能

SQL 语言使用 SEIECT 语句完成查询功能，用 IN-SERT、DELETE、UPDATE 语句完成增加（插入）、删除、修改的功能。

（1）SELECT 语句的语法是：

SELECT 目标列 FROM 表

[WHERE 条件表达式]

[GROUP BY 列名 1 [HAVING 内部函数表达式]

[ORDER BY 列名 [ASC｜DESC]；

上面的 SQL 语句中，SELECT 子句表达从基本表中选取那些列组成结果表；WHERE 子句表达从基本表中选取那些行的条件；GROUP 子句是将选取结果按照列名 1 分组，分组的附加条件用 HAVING 加函数表达式给出；ORDER 子句是将结果集排序，升序 ASC 或降序 DESC。“□”内的内容为可选项。

SELECT 语句的成分丰富多样，使用非常灵活和便利，下面是一个使用 SELECT 语句进行查询的例子。从上面建立的表中查询性别为男的学生的学号和姓名：

SELECT S#, SN FROM S WHERE SEX＝‘男’

SELECT 子句中可以用“*”代表选取表中的所有列，例如：

SELECT * FROM S WHERE SEX＝‘男’

当查询涉及两个表时，称为连接查询。一般是自然连接或等值连接。连接谓词的比较符是“＝”时，就是等值连接的情况。如果在相同目标列中去掉相同的字段名，则为自然连接。连接查询是 SELECT 语句的一个很重要的查询功能。在下面的表 R 和表 S 中，我们要查询所有学生的语文成绩。

SELECT R·S#，R·SN，S·YW

FROM R，S

WHERE R·S#＝S·S#

R

S#（学号）	SN （姓名）
01	张三
02	李四
03	王五

S

SN （姓名）	YW （语文）
01	90
02	88
03	80
04	70

结果

S#（学号）	SN （姓名）	YW （语文）
01	张三	90
02	李四	88
03	王五	30

SELECT 语句还有更多的用法，我们不再举例。下面我们看看插入、删除更新数据的情况。

（2）SQL 使用下面的语句插入数据：

INSERT INTO 表名[（字段名，[字段名]……）]

VALUES（常量[，常量]…）

例如：

INSERT INTO S

VALUES（‘05’，‘赵六’，‘男’）

这个 SQL 语句在前面定义的表可插入一条数据。一般的数据库管理系统在执行插入时，会检查完整性，当完整性检查通过时，执行插入，否则拒绝执行。

（3）使用 DELETE 语句执行删除：

DELETE FROM 表名[WHERE 子句]

例如：

DELETE FROM S WHERE S#＝‘03’

在表 S 中将学号为三的记录删掉。删除时，系统也要检查完整性。如果删除记录会破坏完整性，删除将不会被执行。

（4）使用 UPDATE 修改记录：

UPDATE 表名

SET 字段＝表达式 [，字段＝表达式] …

[WHERE 子句]

例如：

UPDATE S

SET SN＝‘张一’

WHERE S#＝‘01’

将学号为“01”的记录的姓名改为“张一”。

SQL 语句的使用是非常灵活的，语法也很丰富。在此无法一一列举，只能够看看 SQL 语句的简单的使用方法。需要说明的是，在不同公司的产品中 SQL 语句的语法是不完全相同的。前面我们列举的这些例子都是对表进行操作，但是，实际上，SQL 语句还可以运用于视图、快照等

（5）视图的概念在数据库中也是很重要的。它是从一个或多个表中选取某些行和列组成的表，它和表都是数据库的外模式的组成，是面向用户的。它的存在可以使安全性控制和使用非常灵活。比如，对某些用户的权限可以规定在视图上，而视图可以是表的某些列组成的，这样，就向用户隐藏了某些需保密的数据。这只是需要使用视图的一个例子，其他还有很多。概括起来有：

1）增加了数据的逻辑独立性，避免了因为数据库结构的改变而引起的程序的改变。

2）减小了用户的负担，用户可以在视图上操作，不用关心数据库其他数据的结构。

3）增加了管理的方便程度。比如像我们刚刚举过的例子，在安全管理上的方便。还有，不同的用户使用同一数据时，可以各自建立方便自己使用的视图，互不干扰，这样作显然是灵活的。

第四章 数据库设计

1.规范化

在前面的章节中，我们已经述及了数据库系统的一般概念。但是如何构造一个适合的数据模式的问题还未提及。例如，在关系数据库中，给定一组数据，应该构造几个关系，每个关系由那些属性组成，这就是数据库的逻辑设计的问题，关系数据库的规范化理论就是在进行数据库逻辑设计时有力工具。

关系数据库是以关系模型为基础的数据库。它利用关系来描述世界，一个关系既可以用来描述一个实体，又可以用来描述实体间的联系。关系实质上是一张二维表。表的每一行叫做一个元组，每一列称为一个属性。一个元组就是该关系所涉及的属性集的笛卡尔积的一个元素。关系是元组的集合的一个子集。关系模式就是这个元组集合结构上的描述。

规范化是关系数据库设计的步骤之一。规范理论研究的核心问题是：用分解关系模式的方法来消除关系模式中的数据冗余，以便于删除、修改等操作灵活地进行，并确保数据的完整性。规范理论属于关系数据理论的范畴，和我们第三章的内容密切相关。

1.数据依赖

现实世界的实际存在决定了关系必须满足一定的完整性约束条件。这些约束表现在对属性取值范围的限制，比如，人的年龄不能超过 1000 岁；还表现在属性值之间的相互关联（主要是相等与否）上。后者称为数据依赖。它是数据库模式设计的关键。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，它是现实世界属性间相互联系的抽象，是数据内在的性质。数据依赖的类型主要有函数依赖和多值依赖。

函数依赖可以通过下面的例子说明。在描述一个学生的关系中，有学号（S#），姓名（SN），班级（GC）等几个属性。由于一个学号对应一个唯一的学生，所以，S#确定之后，SN 和 GC 也就确定了，就像自变量 x 确定，函数 f（x）就确定了一样。我们说 SN、GC 函数 S#，记为：S#⇨SN，S#⇨GC。

现在假设要建立一个数据库来描述学生、班级、班主任（TC）、课程（CN）和成绩（G），将会有下面的函数依赖关系：

S#⇨GC，GC⇨TC，（S#，CN）⇨G

如果只考虑函数依赖这一种数据依赖，就会得到下面表 4.1 的关系：

表 4.1 关系一

班级	班主任	年龄	职称	学号	姓名	课程	成绩
初一	张老师	40	中级	01	张三	语文	100
初一	张老师	40	中级	01	张三	数学	90
初一	张老师	40	中级	01	张三	英语	99
初一	张老师	40	中级	01	李四	语文	80
初一	张老师	40	中级	01	李四	数学	90

这样的数据库设计显然是不好的。

1.插入异常。当学生的课程尚未安排时，班主任老师的有关信息不允许插入（在该关系中，课程属性是主码，不能为空，删除主码时，整个元组也要删除），引起插入异常。

2.删除异常。当学生毕业时，班主任的信息不能保留。

3.修改复杂。当张老师的职称改变时，虽然仅仅改变了一个属性，却要改变所有的有关元组。

4.数据冗余大。

那么怎样才能设计一个好的数据库模式呢？这就是规范化理论讨论的内容。

2.规范化与范式

1971 年，E·F·科德提出了规范化理论。

关系必须是规范化的，关系应满足：每个元组不可分，不存在异常插入，不存在异常删除，修改简单，冗余小。

我们按照属性间依赖情况，可以区分关系规范化的程度为第一范式、第二范式、第三范式和第四范式等。

范式就是对关系模式的限制条件，满足最低要求的叫第一范式，进一步满足另一些要求叫第二范式，依此类推，还有第三范式，第四范式等。分别写为 1NF，2NF，3NF，4NF。

上面的例子是满足关系模式的最基本要求的，是合法的、允许的。但是由于插入、删除、数据冗余等问题，需要消除数据依赖中的不合理的部分，需要进行规范化。一个低一级范式的关系模式，通过模式分解可以转换为若干个高一级范式的关系模式，这种过程就叫规范化。

在前面举过的那个例子，经过规范化后，形成三个关系，由一个仅仅符合 1NF 的关系规范为符合更高级范式的关系：

关系二

学号	课程	成绩
01	语文	100
01	数学	90
01	英语	99
02	语文	80
02	数学	90
…	…	…

关系三

学号	姓名	班级
01	张三	初一
…	…	…

关系四

班级	班主任	生日	职称
初一	张老师	1940.11	中级
…	…	…	…

规范化后的关系改善了关系数据模式，部分消除了插入、删除异常和数据冗余。我们再看关系四，在这个关系中，由于存在着传递依赖，仅仅是符合 2NF，而不符和 3NF，在某些情况下，存在插入异常、删除异常和数据冗余的现象，读者可以自己进行分析。所以需要进一步规范化为下面两个关系。

关系五

班级	班主任
初一	张老师
…	…

关系六

老师名字	生日	职称
张老师	1940.11	中级
…	…	…

这样经过规范化，彻底消除了插入、删除异常、修改复杂和数据冗余等问题。

在关系规范化的过程中，还要考虑多值依赖的情况。

2.关系数据库的设计

数据库设计是指对于一个给定的应用环境，构造最优的数据库模式，建立数据库及其应用系统，使之能够有效地存储数据，满足各种用户的应用需求。数据库设计的目标是能够正确反映现实世界。

数据库设计属于软件工程的范畴。一个大型数据库的设计是一个庞大的工程，涉及到多种技术和多学科，主要是计算机科学、程序设计、软件工程、数据库理论与技术等。数据库设计应与应用系统的设计相结合。数据库设计的目标大致有以下几点：

- 减少有害的数据冗余，提高共享程序；
- 消除异常插入、删除；
- 保存数据的独立性，可修改，可扩充；
- 访问数据库的时间要短；
- 数据库的存储空间要小；
- 要保证数据的安全性和保密性；
- 易于维护。

1.数据库设计方法

有相当长的一段时间，数据库设计主要采用手工试凑法。数据库的设计水平和与设计人员的经验有直接关系。数据库设计只是一种经验的反复实施，而不能称为是一门科学，缺乏科学分析理论基础和工程手段的支持，所以设计质量很难保证。以至于数据库投入运行后，才发现很多问题，需要不断地从头修改，这样，就增加了成本，也带来很多的隐患。此后，人们努力探索提出了许多数据库设计方法。这些方法主要应用了软件工程的成果，提出了一系列的设计规范，形成了规范设计法。

规范设计法主要是将设计的步骤分为需求分析、概念设计、逻辑设计和物理设计等向个步骤，并采用了许多规范化的手段和工具完成每个阶段的任务。比如基于 E-R 模型的数据库设计方法，基于 3NF（第三范式）的设计方法，基于抽象语法规则的设计方法等，就是在数据库设计的各个过程中采用的具体的技术与方法。

规范设计法仍旧是一种手工方法。现在，人们进一步研制了很多系统，用于数据库设计，甚至应用编程。前提是设计人员必须采用规范化的设计手段，规范化的设计会给后期的开发带来很大的方便。对于一个大型的项目而言，设计阶段的工作量，远远大于开发和维护阶段的工作量。对于大型的项目，规范化是必须遵循的设计思想。

2.数据库的设计步骤

数据库的设计过程与应用系统的设计是不能隔离的。一个数据库设计人员对程序设计技术完全不懂是不行的。数据库的设计过程也是应用系统的设计过程。在这个过程中，充分利用软件工程的研究成果，与用户充分地交流，搞清楚应用环境，把数据和数据处理的需求收集、分析、抽象、设计等工作在各个设计阶段都相互参照和相合补充，以完善两方面的设计。

按照上述原则和规范化的设计方法，数据库设计步骤分为六个阶段。

（1）需求分析。这个阶段的工作是要充分调查研究，了解用户需求；了解系统运行环境，制定将要设计的系统的功能；收集基础数据，包括输入、处理和输出数据；在这个过程中，要从系统的观点出发，既要调查数据，又要考虑数据处理，也就是数据库和应用系统同时进行设计。

结构化分析方法（SA）是常用的分析用户需求的规范化的方法，表达用户需求的是数据字典和数据流图（DFD），这些文档成为下个阶段的概念设计的基础，也是将来系统维护的基础。对规范化的设计来说，这些文档是必不可少的。

（2）概念结构设计。概念结构是整个系统的信息结构。它是现实世界的真实反映。包括实体与实体之间的关系。概念结构同时是易于理想的，可以拿它和用户交换意见，而用户的意见是至关重要的。概念结构是独立于各种数据模型的，它是各种数据模型的基础，易于向关系、网状、层次模型转换。概念结构设计的有力工具是 E-R 图。

下面是几个 E-R 图的例子。在 E-R 图中，长方形表示实体，椭圆形表示实体的属性，菱形表示实体间的关系。

概念结构设计的第一步是对需求分析阶段收集到的数据进行分析，参照数据流图和数据字典，逐步确定实体、实体

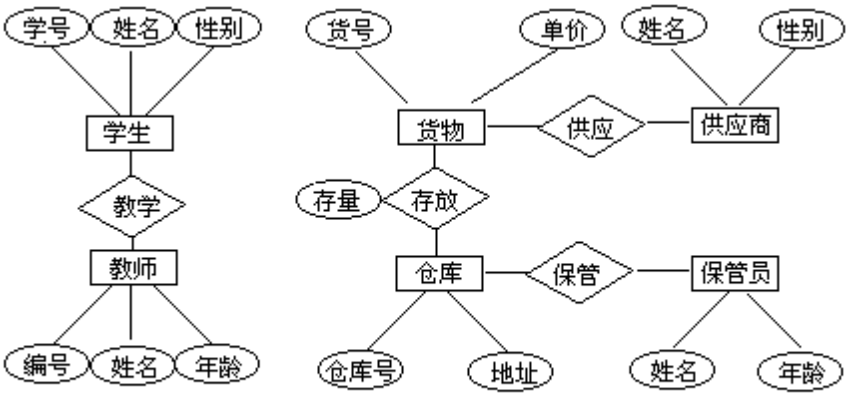


图4．1 E—R图

的属性、实体间的联系（一对一或一对多等），设计出局部 E-R 图。第二步是将多个分 E-R 图逐步集成。集成的过程是一个合并调整的过程，在这个过程中，要消除各种冲突，例如，年龄的表示，在各个分 E-R 图中可能有不同的表示方法，有的用年龄，有的用出生日期；又如同一个实体，有不同的名字，或反过来，不同的实体用了同一个名字。还要消除冗余的数据和联系，冗余会给系统的维护带来困难。最后生成基本 E-R 图。

（3）逻辑结构设计。这个阶段的任务是将概念结构转换成与所选用的 DBMS 所支持的数据模型相符合的过程。一般情况下，应该是向适合概念模型

的数据模型转换，然后再挑选合适的软件 DBMS 和机器。但是实际情况往往不是这样当概念模型向数据模型转化时，一个实体型转换为一个关系模式，而是实体的属性就是关系的属性，联系转换为一个关系模式。

数据库逻辑设计的结果不是唯一的，还要对数据模型进行优化，优化是指适当地修改、调整数模型的结构，提高数据库应用系统的性能。主要措施有记录的垂直分隔、水平分隔、适当增加冗余（提高速度等）。

规范化理论就是用于优化数据库的逻辑设计。广泛地用于逻辑结构结构设计阶段，用模式分解的概念和算法指导设计。用规范化理论分析关系模式的合理程度。

（4）数据库物理设计。这个阶段的任务是为一个给定的逻辑数据模型选取一个合适的物理结构，并对物理结构进行评价。评估的内容包括存储空间、响应时间等，如符合要求，则转向物理实施；不符合要求时，还要从前面的某一阶段开始再次重复上述过程，修改数据模型、重新设计、修改物理结构等。

在进行物理设计时，必须要了解 DBMS 的功能，了解应用环境，理解设备的特性，扬长避短。物理设计的主要内容有数据库的存放策略、数据库结构等。在设计完成后，还要进行性能评估和预测。物理设计过程需要对时间、空间效率、维护代价和各种用户要求进行权衡，对多种方案进行比较和细致的评价，最终选择一个较好的方案。

（5）数据库实施和维护。进入这个阶段后，就要按照逻辑设计和物理设计的结果利用 DBMS 的数据定义语言把数据库描述出来,采用某种设计语言设计应用程序，经过反复调试生成目标模式，然后组织数据入库并试运行。

（6）数据的载入。数据的载入是一个复杂的工作。可以人工输入，也可以利用原来的数据转录，但是数据质量的控制是很重要的，这种检验由应用程序和数据库完整性检查来完成。试运行的主要工作是检查应用程序的功能，测量系统的性能指标，在物理设计阶段所做的评估是否正确，此时可以得到检验。在试运行时，数据量一定要从小到大，以免不必要的重复劳动。试运行时发现问题，随时改正问题，并且不断增加数据，一个系统从试运行到稳定运行是需要一定的时间的。

当数据库正投入运行之后，数据库的开发任务完成，数据库的运行和维护阶段开始。投入运行并不表示万事大吉了。此后的主要工作还有：系统性能监视、改进，系统的转储和恢复，数据库的重组，调整数据库运行等。数据库的维护是整个数据库设计过程的一个有机的组成部分，丝毫也不亚于任何其他阶段的工作，不能认为是附属的不重要的而予以轻视。

第五章 分布式数据库

70 年代中期以来，由于计算机网络通信的迅速发展，以及地理上分散的公司、团体和组织对于数据库更为广泛引用的需求，在集中式数据库系统成熟技术的基础上产生和发展了分布式数据库系统。分布式数据库管理系统是数据通讯和数据库管理相结合的成果。现在，在这个领域里，分布式数据库系统的技术已逐步成熟，产品化的时代已经到来。在 J.马丁（James Martin）的专著《数据库管理的基本原理》中，称 70 年代为数据库年代。而今有更多的专家确信 80 年代是分布式数据处理的年代。事实表明，处理机和存储器的成本继续下降，基于光纤技术的低成本的通讯、微波和卫星通讯等的发展，已经并且还将进一步加速分布系统的开发和广泛使用。

1.数据通讯

这一节的目的不是深入讨论数据通讯的问题，而只是帮助读者理解数据通讯的一些基本问题，提供必须的背景资料。

1.基本概念

采用适当的通信手段，把地理上分散的多个计算机系统连接起来，使它们彼此进行通信和相互利用资源，这就构成了计算机网络，为分布数据处理系统提供了物质基础。计算机网络可以认为是有节点和连接节点的通信路径所构成。

节点是指处理部件，它既有加工能力又有存储能力。它可以是大型计算机，或微型计算机。在节点上，运行的软件必须包括某些通信软件，通常还有支撑软件（操作系统和数据库管理系统）。

分布系统中的另一类部件是连接各个节点的通信路径或链路。通信路径的两个主要指标是带宽和通讯方式。

带宽是每秒钟线路能够传输的信息量（bps）。通信方式有单工通信方式、半双工通信方式、全双工通信方式。

单工线路的成本相对便宜，但使用不够灵活。两种双工方式由于能往每一个方向传输，这就提供了较多的灵活性。半双工方式在改变传输方向时会引起延迟。全双工方式虽然比较昂贵。但不会引起延迟。拨号线路通常是半双工的，租用专线是全双工的。

与整个系统相关的另外一些性能方面的特点还有传输速率、路径建立时间、网络延迟和可靠性等。用户根据需要确定系统的性能，确定哪些因素对他们是最重要的。

2.通信技术

通信技术的重要性表现在不同的技术所能提供的性能/价格方面的不同。推动分布处理特别是分布式数据库管理的因素之一就是在处理和通信之间的性能价值方面的权衡。对于采用音频级线路的通信系统和采用光纤技术为基础的通信系统，所考虑的因素就大不一样，系统设计的差别也是悬殊的。

通信技术有三级。第一级是机内通信，它所涉及的是单个计算机的各个部件之间数据的传送。这样一类的通信通常是在计算机系统中每一对部件之

间进行，是很高速的、位并行的、同步的。这样的通信需要昂贵的计算机电缆，长度受限制。这种通信用于特殊情况，分布系统一般都不采用。但由于新技术的发展，这种情况也在变化之中。

通信技术的发展正在使局部网络成为第二级通信方式。这种方式性能/价格方面是机内通信和远程网络的折中，这些网络地理上是局限于大学、工厂、公司、政府大楼等。由于局域网采用较昂贵的技术实现较好的性能，所以这些技术难于用于大规模的网络系统。局部网络技术的发展，应用的迅速推广和日益受到人们的重视，是与微型机和通讯技术的发展分不开的。计算技术的发展，其趋势是从集中化走向分散化。以局部网络为基础的分布数据库系统，从 70 年代末开始受到人们的广泛注意。

第三类通信系统及时通常所说的分布处理所采用的。这些相对低速的技术，用在公共通信系统中。这些系统称为远程网络。上述这些通信系统和处理技术的性能/价格比将差异决定分布系统采取什么样的设计方案。

2.分布处理

分布处理一词通常是用来描述具有多个处理机的系统。分布处理必须要有数据通讯。然而，分布处理所包含的方面远远超过数据通讯的范围。分布数据库管理系统只是分布处理的一种类型。所以，在了解分布数据库管理系统之前，有必要先掌握分布处理的比较广泛的概念。

1.分布计算机系统

分布计算机系统是由多个分散的计算机经过互连网络构成的统一计算机系统。其中各个物理和逻辑资源元件既相互配合，有高度自治地在全系统范围内实现资源管理和在动态基础上实现任务或功能分配，且能并行地运行分布式程序。

由此可见，分布式计算机系统具有：

（1）模块性。系统的多个分布物理资源和逻辑资源经过互连网络连成单一系统，既相互独立，又互相联系，使系统具备整体控制的条件。

（2）并行性。既分散的系统元件可以合作起来解决一个公共问题，在一个高级操作系统的控制下，实现资源重复或时间重叠等不同形式的并行性。

（3）自治性。既系统资源是高度自治的，不存在主从控制，又能利用处理的局部化原则以减少各节点之间的数据通讯量。

分布处理的主要技术目标是改善下列性能。

（1）可靠性和坚定性。由于模块性系统容易实现资源和路径的冗余；由于自治性，系统避免了集中控制所造成的薄弱环节；加之系统的动态重构能力，使得系统遭受局部损害的情况下仍能继续运行。例如航空订票系统，连续运行是至关重要的。集中式处理通过多处理机或双机系统来提高可靠性，但是，当碰到火灾等破坏性事故时，集中式处理仍不能保证系统安全。使系统分布于不同场地，提供适当的数据冗余，就能提供更好的可靠性。提供路径的冗余。一个节点或一条通讯线路的故障，不会防碍任何功能的执行。

（2）快速响应和较低的费用。分布处理使计算机资源更加靠近用户。分散的小用户能获得计算机的快速响应和直接服务，大用户可充分利用整个系统的能力，从而实现大型机的处理能力和小型机的使用方便两方面的特点。

在通讯费用是昂贵的前提下，常见的手段是将局部性的数据放在最常用的节点上，还有把数据在各个节点上加以复制，由于可以实现局部的检索，就能提供较快的响应时间和较低的开销。问题是更新要保持同步。如果更新的频率高于检索的频率，那么保持同步所要的开销将抵销由于能够局部地实现检索的效益，那么，分布式处理就得不到什么好处了。

（3）增量式地扩展处理能力和系统规模。当用户需求增长，需要扩展规模或更新设备时，可以以低廉的价格以模块作为扩展增量，方便地纳入系统。

2.分布处理的方案

影响分布处理的体系结构的因素有许多。包括数据分布的类型，分布在许多节点的数据是否重复，因为没有重复就没有多重副本需要同步；数据的通讯，要在通讯费用和延迟之间作出折中，要在通信费用和处理费用之间进行权衡。此外，还有系统支持的数据模型和语言，应用的类型等，不同类型的问题需要不同类型的分布系统，这一点尤其要强调。

（1）具有远程作业录入的集中式的处理和存储。计算机刚刚问世的时候，所有的处理、存储、输入和输出都必须在中心场地来完成。最初只是将打卡机和打印机加以分布，利用远程的作业录入在各个局部场地可以直接读取程序和数据，并传输到中心场地进行处理。然后，输出被直接送到各个局部场地。但作业的递交与完成仍然以批处理方式为基础。

（2）具有联机终端的集中式处理和存储。对于这种事务处理系统，所有的处理和存储资源仍旧是集中式的，在任何位置的用户都能请求并引用中心场地的任何处理能力和资源。因为所有的数据和请求都要传送到中心场地，然后再将结果传送回来，所以通信费用是很高的。现在这种情况很少见了。

（3）具有分布数据录入的集中式处理和存储。在前面方法的基础上，将终端改为具有一定处理能力的智能终端或微机。大多数数据录入和编辑由本地节点完成。当准备登录和加工时，才被送往中心场地并更新数据库或文件。在中心场地处理时，需要进一步的编辑，这类编辑有核查字段值的合理性，几个字段值的一致性，核查数据与数据库是否一致，比如，新顾客号码是否唯一。

（4）具有中心数据库局部分隔的分布式数据录入和编辑。这种方法是前面的简单的扩充。在某个节点上提供附加的存储能力。所有的编辑工作在局部完成。推迟该节点与中心场地通信的时刻。这种方法，数据库的一部分仅仅因为编辑的目的而局部地存放。所有实际的处理和更新都仍然是在中心场地进行。局部副本只是特定时刻部分数据库的快照（快照是数据库部分数据的副本）。与中心数据库并不同步修改。

（5）局部与中心场地链接的分布处理和存储。对上一种处理作一些改善，把局部副本作为某一时刻的快照对待，避免复杂的更新同步的问题，对某些应用来说，不需要副本的秒级同步。新方案是把数据库的一部分永久性地放在局部。这种情况，节点具有局部处理能力，有相当规模的次级存储，数据和程序均可以永久性地存储。仓库管理系统就是一个例子，每一个仓库都存储和维护自己的库存数据。按一定的周期把更新发送到中心场地，修改公司的记录。限制是任何通信都是在局部节点和中心之间进行的，节点之间没有通信。

（6）局部节点互连的分布处理和存储。上一种方案的逻辑扩充就是允许

各个局部节点相互通信。例如，某仓库有一项订货要 100 件，库存只有 50 件，就要从另外的仓库调来 50 件。在前述系统中，要将请求发往中心，再转发给第二个仓库。允许局部场地之间通信，会有一些收益。在大多数情况下，用户必须知道数据存放在何处以及如何去存取它们。系统并不具有确定数据存放位置和自动存取它们的专门的支撑软件，这一层次的增加是属于将系统发展成为分布式数据库管理系统的一部分工作。在没有分布式数据库管理系统能力的情况下，用户必须一次一次分别向每一个仓库发出询问，查看是否有足够的存货，当用户找到一个具有足够熟练的仓库之后，就可以请求填写订货单了。

（7）分布式数据库管理系统。这个方案需要有最大量的软件支持。用户应当能存取系统内任何地方的数据，而无需指定它们的位置。在前面的例子中，分布式数据库管理系统允许用户请求其库存至少有 100 件的仓库。系统向许多局部场地发出请求以便找到一个这样的仓库，并且请求那个仓库提供全部工具。系统作了大量的工作，而用户仅仅是发出了需要 100 件的请求。即使对于这种方案，系统对用户的支持也有着许多不同的层次。一个具体的分布式数据库管理系统的设计允许或不允许数据的重复，然而，一般都有重复，这就要求有许多处理来维护各个副本之间的一致性。下一节将详细介绍分布式数据库管理系统的一些概念。

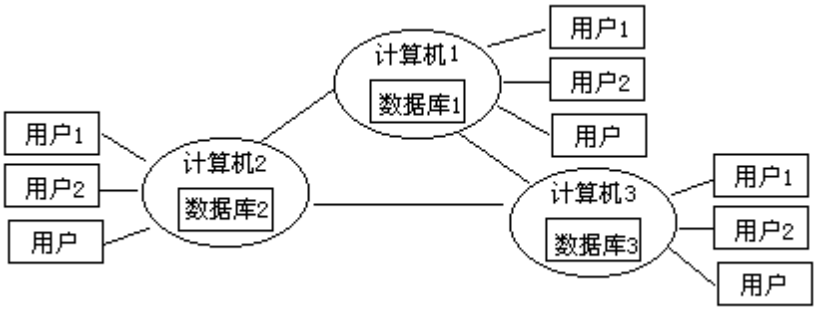
3.分布式数据库管理系统

下面我们对当今日益重要起来的分布式数据库管理系统做一个简单的介绍，而对分布式数据库系统的体系结构、模式结构，分布式数据库系统的设计与实现，包括数据分片、功能分片、更新同步、查询处理与优化、分布事务管理和并发控制等技术细节，不作详细介绍。

1.什么是分布式数据库管理系统。什么样的数据库系统才算是分布式数据库系统呢？分布式数据库是由一组数据组成的，这些数据分布在计算机网络的不同节点（亦称场地）上，逻辑上是属于同一系统的。网络中的每个节点具有独立处理的能力（称为场地自治），可以执行局部应用。同时，每个节点也能通过网络系统执行全局应用。这个定义强调了以下几个方面：

（1）分布性。数据库的数据不是存储在同一场地，更确切地讲，不存储在同一计算机的存储设备上。这是和集中式数据库相区别的。

（2）场地自治性和自治场地之间的协作性。每个场地是独立的数据库系统：它有自己的数据库，自己的一组终端，自己的中央控制器，运行自己的局部 DBMS，执行局部应用，具有高度的自治。同时又相互协作组成一个整体，这种整体性的含义是，对于用户来说，一个分布式数据库系统逻辑上看如同一个集中式数据库系统，用户可以在任何一个场地执行全局应用。如图 5.1。



图中的三台计算机，每台都有自己的数据库系统，三台计算机之间用网络相连。每台计算机有自己的若干终端，用户可以通过终端对本节点中的数据库执行某引起应用（局部应用），也可以通过终端对二个或二个以上节点中的数据库执行某些应用（全局应用或分布应用）。这样的系统是分布式数据库系统，而不支持全局应用的系统不能称为分布式系统。一个典型的全局应用的例子是银行转帐。这个应用要求从一个分行的帐户（DB1）中转若干到另一个分行的帐户（DB2）中去，因此要同时更新两个节点上的数据库。

2.分布式数据库系统的特点

分布式数据库系统是在集中式数据库系统成熟技术的基础上发展起来的，但不是简单地把集中式数据库分散地实现，它具有自己的性质和特征。集中式数据库系统的许多概念和技术，如数据独立性、数据共享和减少冗余度、并发控制、完整性、安全性和恢复等在分布式数据库系统中都有了不同的、更加丰富的内容。

（1）数据独立性。数据独立性是数据库方法追求的主要目标之一。在集中式数据库中，数据独立性包括两方面：数据的逻辑独立性和物理独立性。其意义在于程序和数据的逻辑结构和数据的存储结构无关。在分布式系统中，数据库独立性除了上面所说之外，还有数据分布独立性亦称分布透明性，即用户不必关心数据的逻辑分片，不必关心数据的物理位置分布的细节，也不必关心重复副本（冗余数据）的一致性问题。有了分布透明性，用户的应用程序书写起来就如同数据没有分布一样。在集中式数据库中，数据的独立性是通过系统的三级模式和它们之间的二级映象得到的。分布式数据库，分布透明性是由于引入新的模式和模式之间的映象得到的。

（2）集中与自治相结合的控制结构。数据库是供用户共享的，在集中式数据库中，为保证数据的安全性和完整性，对数据库的控制是集中的。由数据库管理员（DBA）负责监督和维护系统的正常运行。

在分布式数据库中，数据的共享有两个层次：一是局部共享，即在局部场地上存储局部用户的共享数据。二是全局共享，即在分布式数据库的各个场地也存储可供网络中其他场地的用户共享的数据，支持全局引用。因此，相应的控制结构也具有两个层次：集中和自治。各局部的 DBMS 可以独立地管理局部数据库，具有自治的功能。同时，系统又设有集中控制机制，协调各局部 DBMS 的工作，执行全局应用。

（3）适当增加数据冗余度。在集中式数据库中，尽量减少冗余度是系统目标之一。其原因是，冗余数据浪费存储空间，而且容易造成个副本之间的不一致性。减少冗余度的目标是用数据共享来达到的。而在分布式系统中却希望增加冗余数据，在不同的场地存储同一数据的多个副本。其原因是提高

系统的可靠性和性能，当某一场地出现故障，系统可以对另一场地上的相同副本进行操作，不会造成系统的瘫痪。系统可以根据距离选择离用户最近的数据副本进行操作，减少通信代价。但是增加冗余会碰到集中式数据库同样的问题，即不利于更新，增加了系统维护代价，需要在这些方面作出权衡。

（4）全局的一致性、可串行性和可恢复性。分布式数据库中各局部数据库应满足集中式数据库的一致性、可串行性和可恢复性。除此以外，还要保证数据库的全局一致性、可串行性和可恢复性。例如，在前面提到的银行转帐事务中，包括两个节点上的更新操作，当其中一个节点出现故障，应使全局事务回滚，在一个节点撤销已经执行的操作等。

3.分布式数据库系统的目标

研制分布式数据库系统的动机、目的，主要包括技术和组织两方面的目标。

（1）降低费用。使用数据库的单位在组织上往往是分布的（部门、科室），在地理上也是分布的。分布式数据库系统的结构符合这种分布的要求。允许用户在自己的本地录用、查询、维护等操作，实行局部控制，降低通信代价，提高响应速度。

（2）提高系统可靠性。将数据分布于多个场地，并增加适当的冗余度可以提供更好的可靠性。在一些可靠性要求高的系统中，这一点尤其重要。避免了因为某个场地的故障而造成全部瘫痪的后果。

（3）保护投资。当在一个企业中已经建成了若干个数据库之后，为了相互利用资源，为了开发全局应用，就要研制分布式数据库系统。否则，就要把现有的数据库集中起来重建一个更大的集中式数据库，将是困难和不经济的。所以，利用分布式数据库充分利用现有数据库资源，提高利用率。

（4）易于扩展处理能力和系统规模。当一个企业增加了新的部门时，分布式数据库系统的结构可以很容易地扩展系统，甚至是唯一的途径：在分布式数据库中增加一个新的节点，不影响现有系统的正常运行。这样比扩大集中式系统要灵活经济。在集中式系统中扩大系统和系统升级，由于有硬件不兼容和软件改变困难等缺点，升级的代价常常是昂贵和不可行的。

4.现状与前景

尽管在过去的时间里，分布式数据库已经取得了很显著的研究成果，但是，成功地进入商品化运行的软件却仍为数不多。

集中系统的数据库设计是比较复杂的，而分布式数据库的设计就更为复杂了。它除了集中式数据库设计的所有复杂性，还有数据分布的决策、更新同步以及查询分解等的复杂性。另外，还有设计通信系统的问题。

大多数的数据库管理系统也许走一条从集中到分布的道路。首先是跨越数个节点定义数据库，避免不同节点数据的更新同步问题，许可局部和远程查询，回避了复杂的查询处理问题。进一步的工作是增加有限的重复，如果最新的数据并不是最重要的情况下，这样提高了检索的性能。最后，就是完全的分布式数据库管理。系统的功能能够处理复杂的查询，有较好的并发控制机制和保证数据的更新同步。

对分布数据管理的研究有两个方面。一是单项的研究。比如数据的分布问题，通信问题等。在研究一个问题时，假定其他因素是不变的，得出研究

成果。此处还要研究的是要将各种因素综合起来，研究它们的相互作用和结果。数据库设计和更新同步之间就有密切的联系，对于更新要求，依据不同的更新同步方案，对通信系统的要求也随着不同。因此，就要对这些因素综合地考虑。

分布式数据库系统的研究领域还包括对计算机网络的研究。计算机网络技术的迅速发展，已经很大程度地影响到了数据库和分布数据库的领域。不管是在远程网络还是局域网领域，都发生了很多的变比。局域网和远程网之间的处理差别，必然会导致处理数据库和分布数据库问题的显然不同的一些原则和方法。

第六章 面向对象数据库

1.面向对象数据库

面向对象的思想首先出现在程序设计语言中。“面向对象”是一种认识客观世界和模拟客观世界的方法，它将客观世界看成是由许多不同种类的对象构成的，每个对象都有自己的内部状态和运动规律，不同对象之间的相互联系和相互作用就构成了完整的客观世界。面向对象方法学所引入的对象、方法、消费、类、实例、继承性、封装性等一系列概念，为我们认识和模拟客观世界，设计和实现大型软件系统奠定了坚实的基础。

随着研究的深入和发展，现在面向对象技术已经应用到计算机软件的各个领域，如面向对象的分析、面向对象的设计、面向对象的操作系统、面向对象的数据库系统、面向对象的专家系统、面向对象的开发工具、面向对象的用户界面等。

数据库系统是信息系统的核心。一般地说，综合的信息系统就是大型数据库应用系统。

将面向对象技术应用到数据库系统中，这是数据库应用发展的迫切需要。也是面向对象技术和数据库技术发展的必然结果。面向对象技术在数据库系统中的应用主要体现在数据库管理系统和数据库应用开发工具两个方面，即面向对象的数据库系统和面向对象的数据库应用开发工具。

数据库管理系统是建立信息系统的基础。

将面向对象技术应用到数据库管理系统中，使数据库管理系统能够支持面向对象数据模型，这对于提高数据库系统模拟客观世界的能力，扩大数据库应用领域具有重要的意义。

数据库应用开发工具是信息系统开发的必备环境，将面向对象技术应用到数据库应用开发工具中，使数据库应用开发工具能够支持面向对象的开发方法并提供相应的开发手段，这对于提高应用开发效率、增强应用系统界面的友好性、系统的可伸缩性、可扩充性等具有重要的意义。

1.面向对象的数据库系统

（1）面向对象数据库乃是面向对象技术与数据库技术相结合的产物。在涉及面向对象数据库的论题之前有必要先考察一下“面向对象”的含义。

“面向对象”是近年来计算机领域中的一个出现频率颇高的词汇，在软件工程实践者看来，它常常与如下一些概念密切相关：数据抽象、封装性、继承性、多态性、可扩充性、类属程序设计、信息隐蔽、代码重用、模块化等等，甚至还可以列出更多的相关概念，但最终到底什么是“面向对象”还是不清楚。

往往一谈到“面向对象”，人们就自觉或不自觉地将它与面向对象的程序设汁语言关联起来，而面向对象的程序设计语言有许许多多，而且风格迥异，所以单从语言角度出发来理解“面向对象”是不行的。

不少计算机专家认为，“面向对象”是一种世界观和方法论。

首先，“面向对象”是一种认识客观世界的世界观，这种世界观将客观世界看成是由许多不同种类的对象构成的，每个对象都有自己的内部状态和运动规律，不同对象之间的相互联系和相互作用就构成了完整的客观世界。

其次，“ 面向对象 ”是从结构组织去模拟客观世界的一种方法，这种方法的基本着眼点是构成客观世界的那些成分——对象。

用面向对象的观点去认识世界，用面向对象的方法去模拟客观世界就构成了“ 面向对象 ”的完整含义。从系统实现的角度看，“ 面向对象 ”的实质也许可以看成是支持“ 数据抽象 ”，而且是“ 分层的数据抽象 ”。

所谓“ 数据抽象 ”，粗略地说，就是把数据对象的内部表示和它所允许的操作汇集并封装起来，使得对于数据对象的访问只能通过引用其接口中所规定的外部可见的操作来进行。

把‘ 面向对象 ’概念融合到数据库中去就形成了所谓‘ 面向对象数据库 ”。但是，把哪些概念融合进去以及怎么融合进去？到目前为止都是未解决的问题，看法很不一致，以致于难以面向对象数据库下一个准确的定义。

（2）应用的需求。数据库技术自 60 年代后期问世以来，无论从理论上、技术上，还是应用上，都经历了一个飞速发展的过程。现在，大型信息系统一般都是以数据库系统作为其核心的。

从数据库系统采用的数据模型来看，70 年代广为流行的是网状模型和层次模型的数据库系统。自 80 年代以来，由于关系模型有严格的数学基础，概念简单清晰，非过程化程度高，数据独立性，因此关系型数据库系统的发展非常迅速，所以，计算机厂商新推出的数据库管理系统几乎都是支持关系模型的。

随着数据库技术的发展，数据库应用领域已从传统的商务数据处理扩展到许多新的应用领域，例如计算机辅助设计（CAD）、计算机辅助软件工程（CASE）、图象处理、超文本应用等，关系数据库管理系统很难适应这些新应用领域中模拟复杂对象，模拟对象的复杂行为的需求。甚至在传统的商务数据处理应用中，也提出了新的处理需求，例如存储和检索保险索赔案件中的照片、手写的证词等，这些要求也是传统的关系数据库系统难以满足的。新的应用需求的主要特征是：

1）数据模型要能描述复杂对象，能表达更丰富的语义。

2）数据类型从单一的格式化数据扩展为多媒体（图象、图形、声音）等非格式化的多种类型。

3）支持长事务（以小时、天计）、版本管理和动态模式修改等。

数据库工作者为了给新的应用建立适合的系统，进行了艰苦的探索。所采用的办法一是对传统的 DBMS 针对不同应用进行不同的扩充。但结果表明，这种办法，系统效率常常不能令人满意，而且发现应用开发中关系查询语言和程序设计语言之间不协调、不匹配的问题，为此人们试图把逻辑程序设计和数据库相结合，把函数程序设计和数据库相结合。最近发现把面向对象的程序设计方法和数据库相结合是最有希望的方法。它提供了表示、管理程序的数据两者的统一框架。把面向对象的方法和数据库技术结合起来建造向对象的数据库系统（Object-Oriented Database Systems，简称 OODBS）。

OODBS 在逻辑上和物理上都从面向记录或元组上升为面向对象——面向具有复杂结构的一个逻辑整体。它允许以自然方法并且结合数据抽象机制在结构和行为上对复杂对象建立模型。从而大幅度地提高管理效率，降低用户使用复杂性，并为版本管理、动态模式修改等功能的实现创造了条件。数据库的许多新领域都和面向对象的领域有关系，这些领域包括：语义数据模型、嵌套关系、可扩展的数据库系统、数据库程序设计语言等等。

（3）面向对象数据库系统的特性。面向对象数据库系统的研究始于 80 年代中后期，对 OODB 的研究，实际上是沿着不同的方向、采用不同的方法进行的。例如 OODB 数据模型的研究是沿着三条路线展开的：一条是以 RDB 和 SQL 为基础，扩展关系模型（ERM）；一条是以面向对象的程序设计语言为基础，扩充 OO 模型（EOM）；一条是建立新的 OODB 数据模型（ODM）。因此在 80 年代中后期，关于 OO 概念、OODB 概念、OODB 数据模型和语言等一系列基本概念、术语的定义和理解呈现了百花齐放、百家争鸣的局面。

对于什么是面向对象的数据库系统，目前尚缺乏权威性的统一标准。然而，对于面向对象数据库系统应该具备的基本特性，国际数据库学术界已取得了大体一致的共同认识。

首先，OODB 必须支持面向对象的数据模型，具有面向对象的特性。这些特性主要包括：支持复杂对象，具有对简单对象运用各种对象构造符组成复杂对象有的能力；具有对象标识，对象独立于它的值而存在；具有封装性，数据库对象中既封装数据又封装程序，从而达到信息隐蔽，同时也是逻辑数据独立性的一种形式；支持类型和类的概念，类型概括具有相同特性的一组对象的共同特性；支持类或类型的层次结构，从而支持继承性这一有力的建模工具；允许过载，即将同一名字用于不同类型上的数据操作；通过与现有程序设计语言的合理连接来达到计算完备性；并具有可扩充性。

其次，OODB 必须是一个数据库管理系统，具有数据库管理系统的基本功能。主要包括：持久性，数据库中的数据是持久保存的；外存管理，包括索引管理、数据聚集、数据缓冲、存取路径选择、查询优化等；并发性，系统应该提供和目前的数据库管理系统同样级别的，对多个用户并发操作数据库的支持；故障恢复，系统应该提供和目前的数据库管理系统同样级别的，将数据库从故障后的错误状态恢复到某一正确状态的功能；以及即席查询功能，查询功能应该是非过程化的、高效的、独立于应用的。

面向对象的数据库系统除了必须具备上述面向对象特性和数据库管理系统基本功能外，最好还能具备新应用领域所需要的一些进一步的特性，例如模式演化、版本管理、长事务和嵌套事务、分布式计算等。

（4）面向对象数据库系统的优越性。面向对象数据库系统将面向对象的能力赋予了数据库设计人员和数据库系统的应用开发人员，从而可以大大扩展数据库系统的应用领域，并且提高开发人员的工作效率和应用系统的质量。

1）复杂对象构造能力使得对于客观世界的模拟能力强、方式自然。

关系数据库系统强迫用户多个关系的元组来表示层次数据、嵌套数据或复合数据。例如，职工有职工号、姓名、性别、工资、部门等属性，而部门又有部门号、部门名、部门性质、部门经理等属性。关系数据库中属性的取值只能是基本数据类型，这样，职工元组中的部门属性取值只能是部门号。要查询某职工及其所在部门的信息就需要做“职工”和“部门”这两个关系的连接。这样的表示方式既不自然，又影响查询的速度。面向对象数据库中对象的属性的取值可以是另外一个对象，一个职工对象的部门属性的取值就可以是该部门对象，实际储存的是该对象的对象标识，这样的表示方式自然、易理解，而且在查询某职工及其所在部门的信息时可以通过该部门的对象标识直接找到那个部门，提高了查询的速度。

2）封装性向开发人员和最终用户屏蔽复杂性和实现细节，降低了数据库

应用系统开发和维护的难度。

对象封装将程序封装在一起作为存储和管理的单位，也是用户使用的单位，从外部只能看到它的接口，而看不到实现的细节，对象内部实现的修改不影响对对象的使用，因此使应用系统的开发和维护都变得更加容易。关系数据库系统现在支持存储的过程，即允许程序用某种过程性语言编写并存入数据库中以备以后装载和执行。但是，存储的过程并不和数据封装在一起，即它们不和任何关系或关系的元组相关联，构成一个整体，其信息隐蔽和易维护性显然不如 OODB 中封装起来的对象。

3）继承性使得数据库设计和应用编程成为可重用的。

在面向对象的数据库系统中，类的定义和类库的层次结构体现了系统分析和数据库设计的结果，即体现了客观世界中对象的内部结构及对象之间的联系。同时，类定义中封装的方法保存了数据库应用编程的结果。应用开发人员可以在已建立的类库的基础上派生出新的类，继承已存在的类的属性和方法。例如定义“销售人员”类作为已存的“职工”的职工号、姓名、性别等属性，重用了数据库设计的结果，还可以继承“职工”的计算工资额、显示奖惩记录等方法，重用了应用编程的结果，数据库设计和应用编程的重用对于建立大型复杂的数据库应用系统具有重要意义。

（5）RDB 与 OODB。传统关系数据库中管理的是一张张的二维表，表中每一项是一个元组（记录），而面向对象数据库中管理的是对象的集合，也可以看成是一张张的二维表，但表中每一项是一个对象。

关系数据库中表达现实世界的实体及其联系都是用表，而面向对象数据库中表达现实世界的对象用表，而联系用直接引用机制，这种直接引用机制是通过对象标识符加上（复杂对象）索引机制来实现的，这导致了面向对象数据库与关系数据库相比有很大不同。

用关系数据库表示非传统应用（例如 CAD、CASE、OIS 等）领域中的复杂实体就很不方便而且效率太低,因为元组之间的联系必须通过 JOIN 运算运态地建立，要访问一个复杂对象需经历大量的 JOIN 运算，造成系统效率严重下降，所以再用关系数据库就不行了。而面向对象数据库采用了直接引用机制，所以在存取一个复杂对象时避免了大量的 JOIN 运算。此外，关系数据库的传统应用是短事务应用，即每个事务的执行时间都非常短，所以可以采用封锁机制来实现并发访问。

非传统应用有长期性和合作性的特点，例如；一个设计事务可能要延续很长时间，而且可能两个人在设计同一个子系统，其中只要有一个人的设计事务正确提交就算整个事务正常完成（提交）。在这种情况下，关系数据库中并发控制的封锁策略就不再适用了，因为一个事务若长期封锁一部分对象将导致别的事务长期等待，这是不允许的。

此外，事务的原子性（事务不能嵌套）也不再适用了，因为非传统应用在很多场合下都可以表现成嵌套事务的执行。从设计方法学的角度看，人们已经普遍认识到面向对象的方法学优于过去常用的所谓“自顶向下、逐步求精”的基于功能分解的方法学，如果用户使用面向对象的方怯进行系统的分析与设计，而我们还只能为他们提供关系数据库，则势必在设计与实现之间造成一个语义断层，导致方法应用的不连贯性和语义丢失问题。所以用面向对象的数据库来支持整个面向对象的方法学的应用是必然的要求。

2.面向对象的数据库应用开发工具

（1）应用的需求。为提高应用开发人员的生产率，增强数据库应用系统的界面友好性、可维护性、易扩充性，就必须有数据库应用开发工具来支持数据库应用系统的开发。

十余年来，数据库厂商和工具开发商在数据库应用开发工具上投入了大量的人力和物力，推出了若干个建立在关系数据库系统之上的应用开发工具。早期的开发工具多是字符界面的、集中式的（非客户机/服务器结构的），一般是与特定的 DBMS 配套的。

随着客户机/服务器体系结构的发展,以及对全企业范围数据库应用系统的需求，数据库应用开发人员对应用开发工具提出了新的要求，要求它们支持图形化用户界面（GUI）开发，软件部件重用，开发组的工作方式，应用系统的可伸缩性、可扩充性等。与这些要求相呼应，数据库厂商和工具开发商开始研究将面向对象技术应用到数据库应用开发工具中，并开始推出面向对象的数据库应用开发工具。

（2）面向对象的数据库应用开发工具的特性。面向对象数据模型具有丰富的语义。一方面这使得用户能够对具有复杂数据结构的应用建模，另一方面却又使对数据库的逻辑设计和物理设计变得复杂。

由于 OODB 模式具有类层次结构和聚合结构,在 OODBS 中查询和更新语义比 RDB 复杂，这给应用开发和用户使用带来了难度。以上两个方面的原因说明，要使 OODBS 实用化，与 RDBS 相比更需要数据库设计的辅助工具，更需要在核心层外开发工具层，以提高用户的生产率。

近年来，许多公司在 OODBS 产品中均作了努力。

与关系数据库查询语言有 SQL 标准不同，数据库应用开发工具五花八门，没有统一的标准，当然更没有面向对象的数据库应用开发工具的统一标准。然而，我们还是可以列举出它应该具备的一些基本特性。

首先，作为数据库应用开发工具，它应该提供对应用开发的全面支持，包括图形化的界面描绘工具，应用建立工具，高度工具，图示工具，以及强有力的数据库访问能力和浏览工具等。

其次，作为面向对象的开发工具，它应该支持面向对象的开发方法，包括一个可扩充的面向对象编程语言定义、类的层次结构、继承性、多态性等。

此外，这样的开发工具还应该是客户机/服务器结构的，最好具有与多种数据库服务器的开放联接。以及支持开发组的工作方法，支持应用分割等进一步特性。

面向对象的数据库应用开发工具形成了这样一种环境，允许先进的面向对象技术与成熟的关系数据库技术在同一环境中工作，支持数据库应用系统的开发。

2.面向对象数据库的现状与未来趋势

1.面向对象数据库系统的现状与发展趋势

近十年来，面向对象数据库系统一直是数据库学术界和工业界研究的热点之一。

自从 1987 年以来，已陆续有多个 OODB 产品投入市场。这些产品中的某些已经占有一定的市场，另一些尚处于评估和初步的原型应用开发阶段。总

的说来，当前世界 OODB 市场只占整个数据库产品市场的很小一部分。作为数据库产品 OODB 还是不够成熟的。原因是缺乏某些数据库基本特性，例如完全非过程化的查询语言、视图、授权动态模式变化、参数化的性能调整等。这些都是数据库用户已经熟知的，因而希望提供的。此外，关系数据库产品还提供触发器，元数据管理，数据完整性约束等，而目前大多数的 OODB 产品不提供这样的支持。

尽管面向对象数据库有很吸引人的潜力和市场，但是存在于其中的难题非常之多，主要表现在两个方面，其一是理论方面，包括：面向对象数据模型的数学基础是什么？查询模型、优化模型、并发控制模型、存储模型是怎么样的？等等。理论方面的难点集中在有结构的类型系统和对象中的行为抽象部分。其二是技术方面，由于表示的灵活性导致了系统操作语义的复杂性，这种复杂性对实用性造成了严重威胁。随着产品的面市，甚至现在数据库厂商们正在逐步放弃“面向对象的”这一术语，而且，它们越来越重视自己的关系数据库产品如何支持新的、非关系型数据类型。

面向对象数据库系统的实际工作效率到目前为止还是一个令人头痛的问题。特别值得一提的是面向对象数据库系统损失了关系数据库系统的很多优点，这种损失有些是由于技术尚未成熟而造成的，有的则是由于方法本身使得事情本质上困难了。

目前 OODB 产品的应用还很不普遍，主要应用在一些特殊的行业，特殊的应用领域中。而在传统的商务处理中很少应用。

现在人们普遍认为，OODB 和 RDB 关系不同于 70 年代初 RDB 与网状层次数据库的关系。那时的争论是在同一主要应用领域（即商业事务应用）中究竟用关系数据库还是非关系数据库，也就是谁取代谁的问题。现在，是在肯定关系数据库基本适合商业事务处理的前提下,对非传统的应用用 OODB 来弥补其不足。OODBS 将成为一代数据库系统的典型代表，并和 RDB 共存（而不是替代）。新一代数据库系统应是包括面向对象特性的，与关系数据库系统兼容的（即其语言必须是 SQL 的超集）成熟的数据库系统。它们将在不同的应用领域支持不同的应用要求。

2.面向对象数据库应用开发工具的现状与发展

如上所述，面向对象数据库系统现在应用还很不广泛，将来的趋势也不是取代关系数据库系统。关系数据库系统在现在和将来的相当长时间内仍将是应用的主流。因此，基于关系数据库系统的面向对象应用开发工具对于数据库应用的发展具有十分重要的意义。

目前，已有一些面向对象的数据库应用开发工具推向市场，并由于它们出色的特性而受到用户的欢迎。例如因弗米克斯(Informix)公司的 Informix NewEra，宝兰德

(Borland)公司的 Delphi,威力软件(Powersoft)公司的 Power-Builder 等。预计还会有新的面向对象的数据库应用开发工具陆续推出，并且这些工具在在向对象特性的支持、与多种数据库服务器连接的能力、高效性、易用性等方面将有进一步改进和提高，从而为在关系数据库系统的应用开发中采用面向对象技术提供更有力的支持。

第七章 数据库技术的发展

1.数据库技术发展综述

60 年代，由于计算机的主要应用领域从科学计算转移到数据事务处理，促使数据库技术应运而生，使数据管理技术出现一次飞跃。E.F.科德提出关系数据库模型，在数据库技术和理论方面产生了深远的影响。经过大批数据库专家十余年的不懈努力，数据库领域在理论和时间上取得令人瞩目的成就，它标志着数据库技术的逐渐成熟，使数据管理技术出现了又一次飞跃。然而，人类前进的步伐是不会停止的，数据库技术正面临着新的挑战。

1.数据库技术面临挑战

（1）信息爆炸可能产生大量垃圾。随着社会的信息化进程的加快，信息量剧增，大量的信息来不及组织和处理。例如，美国宇航局近年来从空间收集了大量的数据，美国“陆地”卫星每两周就可以拍摄一次整个地球表面，该卫星运行近 20 年来的 95%的信息还没有人看过。现在还没有这样的数据库可供存储和检索如此大量的数据。再有，美国国会通过一个 30 亿美元的计算，准备构造全人类基因组的 DNA 排列图谱。每个基因组的 DNA 排列长达几十亿个元素，每个元素又是一个复杂机构的数据单元、据估计，人类的基因组约有 5～6 万种，如何表示、访问和处理这样的图谱结构数据，是数据库面临的难题。进入 90 年代，像这样的数据并不罕见，传统的数据库技术受到了挑战。

（2）数据类型的多样化和一体化要求。传统的数据库技术基本上是面向记录的，以字符表示的格式化数据为主，这远远不能满足多种多样的信息类型需求。新的数据库系统应能支持各种静态和动态的数据，如图形、图象、语音、文本、视频、动画、音乐等。

在许多计算机应用中，例如地图、地质图、空间或平面布置图、机器人控制、人工视觉、无人驾驶、医学图象等，常涉及到许多空间属性，例如方向、位置、距离是否覆盖或重叠等。目前，这类数据的表示和处理都由应用程序解决，数据库给予的直接支持很少。两者之间缺少亲近性，随着这类应用的增多，数据量的扩大和共享程度的提高，有必要由数据库系统来管理，这就需要发展相应的数据模型、数据语言和访问方法。

更为重要的，人们对信息的使用常常是综合性的，图形、图象、语音、文本、数据之间常常发生交叉调用，需能运多种手段（图标、声音、表格、命令、语言）综合进行存储检索、管理，这是计算机系统和信息系统逐步走向多媒体化的自然要求。从数据库系统来说，要解决多媒体数据的管理问题。DBMS 虽然以支持多媒体数据作为其研制的主要目标之一，但是投入实用还有相当大的困难，尤其在性能上还很难满足多媒体数据一体化处理的要求。目前，多媒体数据基本上靠嵌在关系模式中的文件系统或记录来支持，但数据量大了，数据结构复杂了，共享的要求高了，靠文件系统显然是很难适应的。研制实用化的多媒体数据库对关系数据模型和单一数据类型提出了严峻的挑战。

（3）当前的数据库技术还不能处理不确定或不精确的模糊信息。目前，一般数据库的数据，除空值外都是确定的，而且认为是现实世界的真实反映。

但是实际生活中要求在数据库中能表示、处理不确定和不精确的数据。例如，有些数据不知道确定值，只知道它属于某一集合或某一范围；也有些数据是随机性的，只知道它的不同值出现的概率；还有些数据是模糊的，它的值只是它的“可能”值，或者用自然语言值表达。推而广之，一个元组、一个关系，甚至整个数据库都可能是模糊的。要支持这类数据，必须对确定数据模型做相应的扩展，甚至要对数据库理论来一场革命。人们对数据库查询的要求也不再是简单的有解（完全符合查询条件的结果）和无解，而可能是模糊解或不确定解，提供模糊查询结果。

（4）数据库安全。数据库系统的发展方向是在大范围内集成，向广大用户提供方便的服务。今年来便携式计算机大量涌现，因特网扩展延伸，用户将可通过计算机网随时随地访问数据库，这就带来严重的数据库安全和保密问题。不解决这个问题，上述的目标将无法实现。现有的数据库安全措施远不能满足这个要求。在数据库安全模型、访问控制、授权、审计跟踪、数据加密、密钥管理、并发控制等方面都还没有形成明确的主流技术策略。例如，不管是按数据对象分别给用户授权，还是按数据级和用户密级决定能否访问，都不能可靠地防止泄密。比较可靠的办法是数据加密。但在最近，美国的 RSA 数据安全公司，为迫使美国政府放松密码产品的出口限制，发起了一项名为“秘密密钥挑战”的竞赛。因特网上有数万人加入到破解密码的行动中，采用穷举方法，终于在 96 天之后破解了 56 位 DES 加密算法。这令舆论哗然，也令使用这种加密方法的公司、机构不寒而栗。数据库管理系统的安全机制还涉及到对操作系统安全的要求。

（5）对数据库理解和知识获取的要求。目前，粗略地说，全世界平均每天诞生 100 个数据库，每 5 年信息量就要翻一番。正如奈斯比特在《大趋势》一书中所描述的：“我们正在被信息所淹没，但我们却由于缺乏知识而感到饥饿。”但是，我们对数据库的使用还停留在操作员查询一级，只能利用数据库去查询已经存放在库中的一些具体的特定的数据。即使这样，查询前用户还必须熟悉有关的数据模式及其语义，为了了解这方面的情况常常要向数据管理员（DBA）请教。这样无法解决语义的歧义问题，更不能为决策者理解数据库的整体特性服务。高层决策者常常希望把自己的数据库作为知识源，从中提取一些中观的、宏观的知识，希望数据库具有推理、类比、联想、预测能力，甚至能从中得到意想不到的发现，希望数据库能主动而不是被动地提供服务。如商品数据库能根据销售量主动提出调整价格的建议，或者提醒采购库存量已经很少的货物。

2.数据库技术的研究方向

近年来硬软件，特别是硬件的发展，为迎接上述挑战提供了技术基础。对数据库技术来说，下面的技术是有意义的：盘、磁盘组、大规模并行处理技术、光纤传输和高速网、高性能微处理器芯片、人工智能和逻辑程序设计、多媒体技术的发展和推广、面向对象程序设计、开放系统和标准化等。近年来在数据库技术方面形成了下面四个主攻方向：

（1）分布式数据库系统。由于通用操作系统对 DBMS 性能的限制，以及硬件价格的下降和高速网的发展，用专用数据库服务器已变得越来越合理了。专用数据库服务器的操作系统是面向数据库的，因此可以减少许多不必要的开销，可以支持大量的实时事务处理。为了提高服务器的性能，可以采

用磁盘组和大规模并行处理技术。多个数据库服务器连网，也可以构成分布式数据库系统。

分布式数据库系统有两种：一种是物理上分布的，但逻辑上却是集中的。这种分布式数据库只适宜于用途比较单一的、不大的单位或部门。另一种分布式数据库系统在物理上和逻辑上都是分布的，也就是所谓联邦式分布数据库系统。由于组成联邦的各个子数据库系统是相对“自治”的，这种系统可以容纳多种不同用途的、差异较大的数据库，无全局数据模式概念，比较适宜于大范围内数据库的集成。

构成联邦式分布数据库系统的成员可以是集中式数据库、数据库服务器、逻辑集中式分布数据库，也可以是另一个联邦式分布数据库系统，也就是联邦中还可以有联邦。从这个意义上说，联邦式分布数据库系统结构是分布式数据库系统的普遍结构。90 年代，分布式数据库系统将被普遍使用。形形色色的分布式数据库系统都可以被看成上述普遍结构的一个实例。

（2）面向对象的数据库管理系统。数据库管理系统历来是数据库技术的凝聚点，也是数据库技术研究的排头兵，要迎接上述挑战，在现有 DBMS 的基础上改进几乎是不可能的，但现在还没有到研制新一代 DBMS 产的时候，在此之前还需要新一轮的基础研究。

当前在 DBMS 方面，最活跃的研究是面向对象数据库系统。1984 年班西仑（Bancilhon）等人发表面向对象数据库系统宣言是一个重要标志。它将数据与操作方法一体化为对象的概念，数据和过程一起封装。现已出现了一些借鉴了面向对象程序设计的思想和成果的原型和产品，可以看成是在 DBMS 中革新数据模型的重要的尝试和实践；在数据模型方面，对象、封装、对象有识别符、类层次、子类、继承概念和功能已初步形成；在数据库管理方面，提出了持久性对象、长的事务处理、版本管理、方案进化、一致性维护和分散环境的适应性问题；在数据库访问界面上，提出了消息扫描、持久性程序设计语言、计算完备性等概念。总之，面向对象数据库系统的形象正逐步明朗起来。

（3）多媒体数据库。多媒体数据库从本质上说，要解决三个难题。第一是信息媒体的多样化，不仅仅是数值数据和字符数据，要扩大到图形、图象、语音、视频、动画、音乐数据等，形成超文本。当前市场上各种多媒体卡（视频卡、语音卡等）侧重解决实时处理和信息压缩两个问题，并没有解决多媒体数据的存储组织、使用和管理，这就需要提出与之相关的一整套新的理论，作为关系数据库基石的关系代数理论远远不够了。第二要解决多媒体数据集成或表现集成，实现多媒体数据之间的交叉调用和融合。集成粒度越细，多媒体一体化表现才越强，应用的价值也才越大。如果输入和输出的媒体形式是一样的，只能称之为记录和重放。第三是多媒体数据与人之间的实时交互性。没有交互性就没有多媒体，要改变传统数据库查询的被动性，而以多媒体方式主动表现。显然，像 SQL 查询语言是过分的单调和远远的不适应了。例如，能从数据库检录出某人的照片、声音及文字材料，对其音容笑貌有个综合形象，也许还是多媒体数据库的初级应用。通过交互特性使用户介入到多媒体数据库中某个特定条件（范围）的信息过程中，甚至进入一个虚拟的现实世界（VirtualReality），这才是多媒体数据库交互式应用的高级阶段。

（4）数据库中的知识发现。人工智能和数据库技术相结合是很重要的发展趋势，各种各样的智能数据库、演绎数据库和专家系统，促进了数据库中

的知识发现（KDD）研究。特别是从 1989 年开始，国际上已形成了一个朝气蓬勃的主攻方向，用数据库作为知识源，把逻辑学、统计学、机器学习、模糊学、数据分析、可视化计算等学科成果综合到一起，进行从数据库中发现知识的研究，使得数据库不仅仅能任意查询存放在库中的数据，而且上升到对数据库中数据的整体特征的认识，获得一些与数据库数据相吻合的中观或宏观的知识。这不仅有利于数据库自身的增长和管理，而且大大提高了数据库的利用率，使之有可能成为决策支持系统的基础，特别是使用模糊学和自然语言值，通过隶属云和语言原子模型来沟通定性分析和定量分析。例如通过一个地区人口普查数据库可望得出有助于人口控制的政策；通过一个商品数据库发现有利于价格调整的知识；通过一个公安局刑事犯罪数据库，提出对新案例的侦破建议等。KDD 方法绕过了专家系统中知识获取的瓶颈，充分利用了现有的数据库技术成果，形成了用数据库作为知识源的一整套新的策略和方法。在这个领域，目前讨论的热点集中在数据仓库和数据挖掘。

（5）专用数据库系统。在地理、气象、科学、统计、工程等应用领域，需要适用于不同的环境，需要解决不同的问题，在这些领域应用的数据库管理完全不同于商业事务管理，并且日益显示其重要性和迫切性。工程数据库、科学与统计数据库等近年来得到了很大的发展，这是由于常规的商用数据库系统不能有效地支持这些应用，而常规数据库的研究出发点又不是专业数据库必须支持的。这些领域数据各具特色，必须专门地去研究和开发。目前它们已经取得了很大的进展。

正是计算机科学、数据库技术、网络、人工智能、多媒体技术等的发展和彼此渗透结合，不断扩展数据库新的研究和应用领域。上述的四个主攻方向不是孤立的，它们彼此促进，互相渗透。人们期待着 21 世纪在信息处理技术上新的重大突破，数据管理技术的第三次飞跃即将到来。

3.数据库系统结构的发展

几十年来，运行数据库的计算机系统结构依次发生了下面的变化。

（1）主机式系统。大型机、小型机和高性能工作站被用来作数据库的原始宿主机。在宿主机中包括多用户的操作系统，DBMS 本身，访问数据库的各种应用程序，与用户终端之间发送接受数据的通信设施等。用户终端多是由没有处理能力的哑终端充当，或是由承担些处理屏幕图形和用户输入的 PC 机充当。所有的处理工作在宿主机的集中式系统中完成。

和集中式的运算模式相匹配的数据库系统称为集中式数据库系统，在这样的传统平台上建立的数据库管理系统就是人们一般了解的集中式数据库系统，如 SQL/DS DB2 早期版本的 lngres 和 Oracle 等。

（2）文件共享式系统。80 年代流行将很多 PC 机互联成一个局域网（LAN），LAN 中要共享的数据放在网上的一台计算机——文件服务器上。在这种方式下，所有的数据工作是在运行数据库应用程序的 PC 机上完成，文件服务器只是负责搜索文件并将其发送给合适的用户。为了维护数据的完整性和安全性，用户要更新、修改记录或数据文件时要加锁，当同时有多个用户要访问数据库时，会发生用户间的使用冲突。又因为是对整个文件进行传送，当对数据库的访问频繁时，网络负担加重，形成网络传输瓶颈。随着用户的增加，并发事务的处理冲突，网络的传送限制，和 PC 机的处理能力限制，都导致系统性能下降和复杂行增加。目前广泛使用的 Novell 局域网就是这种模

式的典型。

（3）C/S 结构系统。C/S 数据库的一般形式是将数据的处理分成两个部分：客户机和服务器。前者通常由 PC 机来担任，运行访问、更新、删除数据库的应用程序。后者由 UNIX（RISC）工作站、小型机、超级服务器或高档 PC 机担任。运行网络操作系统（NOS）和全部或部分 DBMS，操作数据库的更新、查询、删除、传送等。文件服务器继续为应用程序提供可代享的数据，也可以和数据库服务器用一台机器。客户机利用 PC 机的运算能力，采用良好的 GUI 界面，处理所有的输入输出，以及部分查询算法的优化、转化，查询结果的排序、报表生成等功能。DBMS 服务器完成客户机的查询、磁盘访问、返回查询结果等操作。C/S 方式下，处理工作恰当地分布在客户机和服务器两端，充分利用网上的各种机器资源，网上传送的不再是整个文件，而是查询与查询结果，流通量减少。

在典型的客户机/服务器结构中，把数据库的工作，例如数据修改、分类，安全性确认，事务恢复和对共享数据的访问管理，全部放在服务器上进行。这样一来，事务逻辑所涉及到的安全性、数据完整性和逻辑完整性都可以集中在服务器上来统一由系统解决，而不是让访问该数据的每个应用程序自己解决，从而有利于提高性能和完善控制，并减少应用程序开发和维护的开销。

（4）分布式处理。当 C/S 数据库系统需要和其他 C/S 系统或中心宿主机共享数据时，就形成了分布式处理系统。在一个分布式处理系统中，用户只要向本地数据库服务器发出请求，本地服务器确定它没有该数据后，就把该请求送入网络，从适当的数据库服务器中取得数据，并把数据和本地机上的数据一起发回给用户。

和运算模式的变化相关，数据库系统经历了集中式数据库系统（Centralized DBS）、分散式数据库系统（DistributedDBS）、C/S 式数据库系统和分布式数据库系统的演变。在数据库应用领域，由于集中式数据库系统要求终端和主机不能相隔太远，而在实际应用中，却有这种要求，比如东城储蓄所和西城储蓄所之间，解决这个问题的是分散式数据库系统。在分散式数据库系统中，每个节点有自己的数据库系统，中心计算机只保留一些类似存款总数等的关键性帐目，而不保留每个节点的帐目副本。

像分散式 DBS 一样，分布式（Dietributed DBS）DBS 也是由计算机网络联结的一组结点组成。不同的是这组网络联结的一群结点共同构成了一个统一的分布式数据库，由一个统一的 DDBMS 来管理。对 DDBS 来说，其每一个结点上都是一个多用户系统。将 PC 机联网，由于每个结点上都是单用户的 DBMS，不能算真正的 DDBMS。某结点上的一个用户所存取的数据可能物理地存放在网络上的其他结点上，甚至该用户所提交事务的计算工作也是在某个另外的结点上完成。但是该用户感觉不到这种物理上的分布性。这就是结点透明性（Site Tranparency）。无论用户在哪个结点上，他都感到整个 DDBS 就处于他所在的结点。

容易想象，这样的分布式数据库系统是非常复杂而庞大的软件系统。国际上对 DDBMS 的研究始于 70 年代中期，最有名的先驱系统是美国计算机公司（CCA）公司为美国国防部制的 SDD-1 系统。对此后全球的 DDBMS 的研制产生了决定性的影响。这种完全透明的 DDBMS 是完全分布式的，不存在任何中心控制。例如 SDD-1 控制上百个军事基地，若其中某些军事基地被炸毁则余下的部分照常运转。反过来若要增加一些基地，则整个系统也立即可平滑地扩

大。

C/S 数据库系统是近几年计算机界的主要趋势之一，至今方兴未艾。C/S 结构汹涌澎湃的原因分析起来有几个方面。首先是微机以及工作站自 80 年代初面世以来的飞快发展速度。其处理速度、存储量及各种功能都已远超过当年的大型机（如 IBM 370 168），在其上已可配置众多的系统软件和工具。此外，在数据库管理系统方面有人们熟悉的 XBASE 系列、FOXPRO、ACCESS 等，操作系统则装备有无人不知的 DOS、Windows 3.1 等。严格看来，以上在微机上流行的系统不“像”典型的操作系统和数据库管理系统。这些系统缺乏许多 OS 和 DBMS 的典型特征，有些更像是文件系统。但是应用和市场却毫不顾虑这些“不像”，普及全球数达几千万份之多。可是事实也证明了“真正的”操作系统和数据库管理系统所实现的体系结构、可靠性、安全性、理论基础的严谨性等等并非无的放矢，困扰微机环境的病毒，不可靠性、不安全性等已对软便件环境提出了强烈的改进需求。但是微机究竟规模有限，要做好所有事情似不可能也无必要，C/S 结构是一条可取的途径。还有，现在的局域网（LAN）和广域网（WAN）的广泛普及已成大势所趋。网络所联的微机群的性能和功能有了极大的提高，大可与大型机相媲美，但其价格却仅为几十甚至上百分之一，这是一项无可匹敌的优势。许多公司和组织都在考虑规模向下适化（Downsizing）以降低成本，提高效率。在这种微机联网的结构中 C/S 是一种十分优秀的体系。

有了以上的解释之后，C/S 结构的一个决定性原因就清楚了。在今天全球应用计算机已达半个世纪，所积累的信息数据财富有上万亿美元之巨，且其中一些数据的价格难以用金钱衡量，在各公司、组织的软硬件环境的升级换代中，保护投资和压缩规模和复杂程序是最经济合算的。如何保证在这种转移中使数据财富安全可靠地、平滑地过渡到新环境，显然是各主管考虑的第一要务（Lagacy Application）。不容置疑，C/S 结构是最优候选结构，从集中式的 DBS 往下过渡或向上升级，C/S 结构显然较分布式 DBS 更为自然。如此巨大的需求，它对于 C/S DBS 的推动力之惊人就不足为奇了。但是，C/S 结构并非什么从天而降的技术，它的产生有其背景，它与人们经历的过时的技术有平滑的密切的联系。

4.数据库的并行处理技术

并行处理是提高数据库系统对事务快速响应能力的一个十分有效的方法。也是当前数据库系统普遍采用技术手段。并且也是大张旗鼓地进行商业宣传的素材。

从硬件角度，并行处理是设置若干个能同时工作的部件和设备，如中央处理部件与外部设备并行，多个外部设备并行，以及多个处理机并行等等。

从软件角度，并行处理是设置若干个可以同时运行的单位。通过这些单位的运行，可以完成相同或不同的预定功能。已经十分通用的进程（Process）以及最近几年流行起来的线程（thread）就是这种运行单位的典型代表。这些运行单位可以在单处理机上交替运行，也可以在多处理机组成的系统中同时在多个处理机上运行。

并行处理技术已经渗透到计算机技术的各个领域。在数据库系统中，多线程（Multithread）并行技术和虚拟服务器结构可以大大提高系统的性能。

（1）多线程并行技术

1）多线程的基本概念。对于线程，目前尚无统一的定义。它是从并行处理的进程概念发展变化而来的，所以有时也称其为轻型进程（Lightweight Process）。

为了减少系统开销，提高系统效率，人们把传统的进程进一步细分为多个可并行执行的单位，称为线程。

在这种既有进程又有线程的系统中，进程是对计算机资源进行分配的基本单位，它具有一个地址空间，一组寄存器，一个程序计数器和一个堆栈。而线程则是系统运行的基本单位。同一进程中的多个线程共享该进程的地址空间，但各线程保持自己的一组寄存器、程序计数器和堆栈，其中含有处理器运行的状态，线程睡眠的原因以及换入、换出信号等。UNIX 的进程相当于这里的单线程进程。

多线程控制机制十分适合并行计算模型。如果有多个 CPU，则多线程可真正并行工作。例如：一个文件服务进程中有多个线程，每个线程最初都在等待用户请求。当有一个请求时，其中某一个线程还可以共享数据区，并用信号灯机制来控制同步。多线程的发展思路与并行处理的多进程思路是一致的。

上述文件服务器如果设置成单一进程（无线程划分），则会出现一个请求被处理而进程等待 I/O 时，别的请求无法被接收。当然，也可以设置成多个进程，但这时空间需求大大增加，由于进程空间的多次切换而大大降低系统的效率和加大系统的响应时间，而且由于没有共享数据区而无法用信号灯机制进行同步。

线程之间的通讯是通过消息（message）和端口（port）来实现的。消息通过端口传送给另一个线程。消息头含有消息种类及目的端口等消息。消息的传递可以是同步的也可以是异步的。

多线程并行的设计有一些特定的要求，如代码可重入、工作区浮动等。

2）数据库中的多线程。在数据库系统中，对于每个用户一个进程的结构，为处理增加的用户事务所包含的开销会呈指数增长。影响开销增加的因素主要有：进程数的增加，存储空间的增加，以及锁数量的增加。在这种系统中，增加一个用户需要增加大约 1MB 的存储开销，由此带来的页式开销、进程转换开销以及 OS 系统级的开销，这些都直接影响系统的性能（主要指吞吐量和响应时间）。

在多线程结构的系统中，服务器作为一个单一进程运行，并且调度、任务转换、盘缓冲、锁及事务处理均由服务器管理而不要花费任何 OS 开销，每增加一个用户只需增加一个线程，存储开销只要增加 34KB，是上述每用户一进程结构的 1/30。于是，可以有更多的存储空间用于磁盘缓冲（caching）和过程缓冲（caching），从而减少 I/O 开销；由于所用存储器空间减少，自然使页式开销减少；线程的转换只是简单的运行状态的改变，而不会引起进程空间的切换，再加上原来的 OS 管理的一些功能可由服务器来实现，从而减少了系统方面的开销。

综合这些因素，使得单进程多线程结构中事务处理的响应时间和系统的吞吐能力比在每用户一进程结构中有十分明显的改进，特别是在用户数大量增加时这种优点更为突出。这也正是为什么在多线程结构用户数增至成百上千而系统性能（主要指事务响应时间）仍然很少下降的原因所在。

（2）虚拟服务器结构（VSA）。虚拟服务器结构是一种适用于对称多处

理机（SMP）配置的数据库结构。它的基本思想是，在每个指定的 CPU 上设一个引擎（engine）作为一个服务器。这些服务器在功能上是完全相同的。通过它们的密切协同工作，成为一个逻辑服务器，处理外部的事务要求。

每个引擎有一个相应的进程，而这个进程具有多线程结构，于是在这种结构中便形成了一种多进程多线程结构。每一个外部事件要求，由某个运行于固定 CPU 上的引擎所对应的进程中的一个线程来处理。由于 CPU 的对称性和每个引擎功能的一致性，事务由哪个线程处理其结果是完全一样的。这样一来，便使多 CPU 的并行处理能力得到充分发挥，使系统的吞吐量大大增加而响应时间相对缩短。

2.多媒体数据库

多媒体数据库是数据库技术的新兴领域。它研究的对象已从传统的单一的字符类型的信息媒体发展为包括图形、图象、声音和字符的多种类型的信息媒体。由于研究对象的多样化，因而多媒体数据库技术提出了很多比传统数据库技术更为复杂和更为新颖的研究课题。

多媒体数据库技术的出现和形成，一方面是由于有实际的应用需求，而另一方面也基于现代计算技术发展的新成果。

由于现实世界的复杂性，因而其表现的形式也就会是多样的，作为信息传播的形式，除了通常传播媒体文字和符号外，当然也时常见到上述的以图形、图象和声音等媒体的表现形式，以及它们的相互组合。传统的数据库技术在文字和符合的输入、存储、处理、检索和输出等方面已有较成熟的技术，还有相应的理论成果。当初，它的应用主要在事务处理和商业领域。随着计算机应用领域的扩展和技术的发展，人们已不满足于单一的信息表现形式，或单一的信息表现形式已不能满足实际应用的需要，而提出了对多种信息媒体的利用和管理的需要。随着现代计算技术的发展，存储技术，如光存储技术方面，出现了大容量的光盘；输入/输出手段的更新，如摄象技术、数字化仪、扫描仪、高分辨率的图形、图象监视器的应用；彩色图形、图象转换设备的完善以及计算机本身处理能力的提高和数据模型理论的发展和完善都为多媒体数据库的实现提供了可能性。

多媒体（Multimedia，或译为多媒介/多媒质）的术语在 1983 年正式使用，1984 年在新加坡召开的超大型数据库（VLDB）第 10 届国际会议上就对多媒体数据库进行了讨论。

1.多媒体数据模型

一般认为，数据模型化是数据库技术的基础和核心。如果广义地理解，数据模型化包括了概念模型、逻辑模型和物理模型的建立。其中概念模型是数据库设计者对现实世界的抽象，逻辑模型是对概念模型的逻辑表示，而物理模型是对逻辑模型的机器表示。要把复杂的现实世界正确地描述出来，并将其数据及关系在数据库中进行存储和管理，关键地一步是要把现实世界抽象为概念模型。多媒体数据库所依托的是多媒体数据模型，首先是需要把各种媒体所建立的概念模型结合为一有机的统一整体，使概念模型一体化，以形成一个“多媒体概念模型”，再以某种符号系统加以表示，而后形成多媒体数据模型的基础。

多媒体数据模型应具有以下特性：

（1）能支持媒体的独立性。这是因为多媒体数据库的目标应能实现诸如媒体的混合、媒体的扩充、媒体的互换。即应能使用户最大限度地可忽略各种媒体间的差别，而实现对复杂数据对象的管理和使用。

（2）要支持数据模型的三个基本要素：数据的结构性质，能描述实体及实体间的联系；具有与数据库相关的语义完整性限制；体现数据的操作特性，亦即要通过对各种媒体的符号化、抽象化，使得用户可以对各种媒体数据进行统一的处理和一致性管理。对不同的内部表示的数据用同样的数据库语言进行操作，并提供能用于多媒体数据库的语言接口。

实现多媒体数据模型的方式是多样的，当前所涉及的方法有：

（1）基于关系数据模型的方法，即在关系数据模型中引入抽象数据类型，并对数据类型定义所必要的数据表示形式及其操作定义加以扩充。

（2）基于语义数据模型的方法，语义数据模型能提供更自然地处理现实世界的数据及其联系的能力，并在实体类型的表示及其联系上具有特点。当然还有其他的方法，如基于面向对象的建模方法等。当然，对于多媒体数据模型的研究还很不充分，目前仍然缺乏完整的、具有普遍意义的理论。

2.多媒体数据库管理系统

数据库管理系统是为实现数据库的建立、操作和控制的软件系统。对于多媒体数据库管理系统与传统的数据库系统一样，要提供对数据的管理、查询和事务处理等功能。除此之外，对于多媒体数据库管理系统必须要求它有独立于媒体的变化；由于其具有面向对象的特征，而往往需要根据不同的对象而提供不同的用户接口和存储结构。

多媒体数据库管理系统的体系结构可以划分为三种类型：

（1）单一型的数据库管理系统体系结构。它是由一个单独的数据库来管理各种不同媒体的数据库以及由不同媒体数据组合的对象数据库。

（2）主从型数据库管理系统体系结构。不同媒体的数据库由几个从属数据库管理系统自己管理，而这些从属数据库管理系统则由一个主数据库管理系统统一管理。用户面向主数据库管理系统，并管理由不同媒体数据组合的对象数据库。

（3）协作型数据库管理系统体系结构。各个媒体数据库由各个分散的成员数据库系统进行管理，各成员数据库管理系统提供通信、查询和处理的接口，用户可以通过任一成员数据库管理系统面向整个多媒体数据库，用户组合媒体数据的对象数据库也分散管理，并允许重复管理。

传统的数据库结构与多媒体数据库结构的差别在于从以记录为中心的存取变成了以对象为中心的存取。一般是若干个记录的组合才能表示一个可存取的对象，因此多媒体数据库的存储结构应该是一种能高速存取的存储结构。

3.多媒体数据库的用户接口

数据库系统的用户界面/接口的友好性在很大程度上影响着系统的成败。多媒体数据库系统是一类检索对象复杂、用途多样、用户面广的数据库系统，因而对用户界面的友好性的要求更为迫切，这就给系统的研制和开发工作带来更高和更多的要求。多媒体数据库的用户界面除了应保持传统的数

据库所提供的功能外，无疑应该有能处理图形、图象等视觉化的界面。

（1）当前高功能工作站的出现，给多媒体数据库用户介面的改善带来了较大的影响。这些工作站具有高分辨率的按位显示、鼠标定位、窗口及图标等功能，利用这些功能可提高和完善窗口管理的质量和能力，从而支持多媒体查询的应用和介面的友好性。

（2）使用图形、图象、等媒体的视象接口。提供视象界面的系统可以划分为三类：查询视觉化；查询结果或数据库表示以及存取视觉化；兼有上述两方面的功能和内容。

（3）多媒体数据库的自然语言界面。利用自然语言做为多媒体数据库的用户界面很适应多媒体数据库系统的数据对象多样、表示抽象化的特点，因为自然语言接口的本身具有变化度大，抽象性强，可以表示不确定性要求等优点。当然，实现自然语言接口有很大的难度。

（4）具有一定智能的接口。适用于：①用户对系统了解甚少；②只能表达部分意图的初级用户。可以为多媒体数据库的应用提供方便。

4.多媒体数据库的硬件环境

多媒体数据库是在数字、符号的基础上，加上图形、图象、声音和它们的组合等种种不同类型媒体的数据，并对它进行统一管理的数据库。由于存储对象的扩充，因此多媒体数据库与以前的以数字和符号为管理对象的数据库相比，对 硬件环境的要求是不完全相同的，主要在输入/输出及存储设备等环境方面有不同的要求。

输入处理是多媒体数据库的重要课题之一。首先，多媒体信息的机内表示，同它的输入处理方式直接有关。目前计算机内部表示方法有多种：对于容易形式化的内容以编码的方式来表示，对于非形式化的内容则通过位串作为图象信息来表示，而其语义则由用户自己去定义。为了进行输入处理，用以读入文字、图形、图象等多媒体构成的原信息所采用的输入设备有自动型和手动型两类：自动型的用光扫描文字及图画，进行阅读并自动输入。扫描所得的多媒体模拟数据，要把每一象素变换成数字数据。这类自动型的设备有传真机、图象扫描仪、电视摄象机等。手动型的设备是由操作人员直接读取操作位置坐标方式输入，如用图象输入板，它通过特殊笔尖所放位置的磁场变化而得出坐标位置，而鼠标器则是通过球的转动方向和转动量来定出坐标。

存储系统的硬件环境与传统的数据库相比有更高的要求。多媒体数据库的特点是要求设备有很大的存储容量。例如，用计算机进行图纸保存，每一张图面的信息量约为 100KB，而图的数量可能达到几万张及致上百万张，这样总的存储量就可能达到几 GB 乃至上百 GB。如果多媒体数据库的存储设备，仍采用以前数据库系统通常采用的磁盘或磁带存储设备，在容量、成本或速度上都可能会带来困难。

最近，光存储技术的应用，光盘做为多媒体储设备的前景十分可喜。目前，三种类型的光盘即只读型光盘、一次性可写型光盘和可多次写入的光盘，都可做为多媒体数据库的候选存储设备。光盘具有容量大，保存时间长（可保存 10-20 年），存取速度快等特点。满足多媒体信息的存储、利用的要求，是一种多媒体数据库技术很理想的存储设备。

当然，在构成多媒体数据库的存储结构时，也可以同时使用多种存储设

备。如在文件的存储中，可能将原文件本身（一次信息）作为永久保持信息存放在一次性可写光盘中，而把它索引或关键词等二次信息存放在磁盘中，并建立它们之间的联系。

输出处理是将检索结果，根据需要进行必要转换，并加以视象化。用于视象化的输出设备有显示设备和打印设备。显示设备的种类很多，对于显示多媒体数据，可根据应用领域和应用目的的不同而加以选择。打印设备种类亦很多，同样需要根据应用领域和使用目的来选用。如有 XY 绘图仪，高速输出图纸的激光打印机和显示画面的彩色拷贝机等。

多媒体做为一种很有应用前景的计算机技术内容，目前仍然还是处于发展的初级价段。据介绍目前还没有成熟的多媒体数据库管理系统推出；其基础理论，如多媒体数据库的数据模型理论仍还不成熟，需要做更多的工作。

3.专用数据库技术——工程数据库

70 年代以来，数据库应用不仅在商用事务处理方面得到广泛的发展，而且逐渐向工程技术领域（如 CAD、地理地图、军事指挥等）渗透。工程数据库与商用数据库有着根本的不同。在工程技术领域，数据模型的构造是从小到大，从简单到复杂，随之数据装入是逐步增大的。可能一开始是某个阶段或某个层次，随着设计者的深入，不断地加入语义信息。动态地修改模式或子模式是经常的，也是必要的。这些工作不仅仅是数据库管理单元事情，而可能是设计师本身。一旦在设计试探中，发现模型错误，应能及时返回到某一设计阶段或层次，沿着另一条路继续设计下去。在这个过程中要充分考虑到工程领域的交互性和实时性。要设计的客体采用什么样的数据结构往往取决于设计人员的构思和设计目标，要求在设计之前（像商用数据库那样）提出一个合适的概念模式是不切实际的。工程领域既要处理复杂的数据结构客体，又要满足快速实时响应、人机交互显示要求。存储结构既要存储图形信息，又要表达复杂关系的客体。所以工程数据库技术研究引起了数据库专家的兴趣。

近十年来，国外已开发的工程数据库有 IPIP、MLDB、TORNADO 等，它们的数据模型是关系和网状混合模型、扩充关系模型、语义网络模型、扩充网状模型等。我国工程数据库的研究，改造商用数据库管理系统，或者商用数据库管理和文件管理向结合的方式处理工程环境的结构化或非结构化的信息，也开发自己的工程数据库系统。

工程数据库的研究成果集中应用于机械制造，这是目前最活跃的领域。有关飞机、汽车、船舰等产品的 CAD 系统，以及通用计算机集成制造系统，都采用数据库技术来管理数据。这些数据库管理系统有的还不完全成熟；VLSI 的设计中，由于大规模集成电路的设计要通过系统描述、功能设计、逻辑设计、线路设计和布图等几个阶段来完成；而且，在各个阶段还需进行模拟和测试，因此对于一个芯片的设计，数据量是十分巨大的。而且设计各个阶段要重复使用，对于这些设计数据组织和管理需要有数据库的支持；在其他方面，如土建施工图，市政信息管理方面也都采用数据库来组织和管理，它们要求数据库能够支持图形，这些也属于工程数据的范畴。

工程数据库除采用网状、关系模型外，有的系统兼顾网络模型和关系模型的优点，采用关系/网状模型。工程数据库支持动态模式，支持分布式处理，

因为一个大型项目的设计（如飞机、计算机等），都是很多设计人员共同合作才能完成的，它们在设计过程中需要使用公共数据，相互之间需要交换，所以工程数据库的运行必须是分布式的。工程数据库采用结构实体，支持语义信息的嵌入此，还有版本管理的功能。所谓版本，是一个工程设计过程中，由于设计方案的差异，在设计的各个阶段形成同一工程实体不同的设计版本；设计过程有时是一个反复试控的过程，所以要求数据库能保留尽可能多的设计版本。传统的数据库技术要多个数据库来管理多个设计版本，这样就会有大量数据冗余，也不便于比较。为此采用版本管理，解决以上问题。

工程数据库是工程设计与制造领域中实用价值较高的软件系统，是当代工程自动化的核心软件系统。工程数据库是高级数据库技术之一，也是多学科的总和。工程数据库技术的发展必将对工程技术自动化产生极大影响。

4.专用数据库技术——科学数据库

80 年代以来，数据库技术已在各个领域中得到广泛应用，使数据库处理从原来简单的数据加工开始为信息化的社会服务。

建立科学数据库是把计算机数据库方法应用到科学技术数据处理领域。在科学数据库中存放的信息是一种知识成分，是专业科技人员在基础研究、应用研究、科学实验及新技术研究与开发等各项活动中产生与积累的数据的集中。这里将保存许多有用的事实、方法、数据、理论及其运用。收集、整理这些科学技术数据，在高性能、大容量、高速度的中、大型计算机上建立数据库群体，存储与管理这些数据，通过通讯网络传送数据，向最终用户提供联机查询及处理，为科学技术活动、社会及经济活动服务，为知识增长和培养人才服务。

1.科学技术数据的分类和处理

由于科学技术数据来自于不同的学科领域，它们的数据类型、数据用途、数据间的结构关系是各不相同的。世界科学技术协会、数据委员会（CODATA）依据数据特点曾提出了科学技术数据分类方法。

现实社会中的许多科技活动都要依据他人所生产的科技数据。建立科学数据库系统是以专业化的科技数据的收集为基础的，数据收集过程可分为三个阶段，首先由实验室、原始报告或出版物中按一定的表示方法进行数据生成。然后对获得的原始数据进行整理、分析、压缩、组织。在当今信息化的社会中，数据代表了信息，数据就是财富。科技数据信息是科学研究的主要成果，也是确定新的研究方向，选择研究课题，推动研究工作和发展应用的基本参考信息。

计算机处理科技术数据是指用计算机采集、评价、归档、存储、检索、显示、传递数据。计算机系统包含如下四部分工作。

1）数据输入。包括数据源的采集，输入数据集的选择与定义；对数据进行检查编辑，使之与系统标准相符合；转换数据格式与数据库要求的格式相一致。

（2）建立数据库。包括设计数据结构，组织并存储数据；设计对数据库访问的权力和限制措施，以便在执行对数据库的检索、插入、更新和删除等操作时予以控制；设计数据库的保护和维护措施，防止数据库破坏，保证在

发生意外故障时恢复数据库；设计监督手段及统计算法，以便监督与管理数据库的运行。

（3）数据检索与加工。包括接收用户使用数据的提问要求；组配这些要求，形成一定的查找表达式；设计算法从数据库中检索用户所要求的数据；将检索得到的数据在终端上显示，供用户浏览、选择；保存用户检索得到的数据或进一步加工处理。

（4）数据输出。包括根据用户的特点要求或用途挑选输出项目；确定并选择数据的组织和格式以适应数据传输的规定；提供传输数据的方法，直接打印数据加工的结果等等。

2.科学数据库的类型

由于各种实际应用的需要，在不同的学科及领域中发展起了多种类型的数据库。依据服务对象可将科学数据库划分为面向具体专业科技人员的专业性数据库及面向社会广泛用户的综合性信息数据库。依数据库中收录的数据类型、内容和用途又可将数据库划分为三大类，即：文献数据库、数值数据库和管理数据库。其中文献数据库的内容是以科技文献资料及图书为主，库中数据多是一次文献经分析加工而来，一般包括题目、作者、出版单位、出版年月、文摘、关键词等。这是从原文献产生的数据，文献库容量非常大，用户使用其检索得到的结果仅仅是文献的线索，再由此线索去找所需的原始文献。数值数据库的内容极其广泛，涉及了多种学科领域的科技数据及一些事实材料，这些数据是科学技术成果以最精确的数字形式表示的结晶，数据入库前要经过鉴定、评选。把经过专门评估的数据或发表的科技数据加工成为计算机可读的形式装入数据库。例如各种化学谱图，化学分子式及结构数据，各种物理常数，物质的各种特性，原子及分子特性，材料及其性能，各种观测数据，地质资料，自然资源资料，图象数据等等。在数值数据库中应有详细描述实体及其属性的数据，以满足用户希望直接获得科学、计算或工程方面的详尽的信息要求，专业人员从数值库比从文献库更能直接得到有用的信息。管理数据库的内容是以服务管理数据为主，如科研课题、科技机构、科研成果、科研器材及一些管理与决策信息。由于社会各界对科技信息的需求在增长及计算机技术和通讯网络技术的飞速发展，各类数据库都发展很快，从 70 年代末期以来，数值数据库的使用次数和增长速度已大大超过文献数据库。

3.科学数据库管理系统

在 70 年代商用数据库管理系统已广泛应用到科学技术界,使用这种软件的主要好处是它提出了数据模型化的方法，对理解数据结构有好处。

另外，无论是信息的提供者和信息需求者都可不必了解计算机的复杂性。例如 IBM 公司的 IBS 是一个层次模型的数据库管理系统，已用于为医学研究建立的嗜血杆菌数据库；欧洲经济合作与开发组织（OECD）的核能数据库（NEADB）使用了在 PDP11/70 计算机上的 DBMS-11（CODASYL 网状 DBMS）。

剑桥结晶学数据中心利用了 ADABAS（德国 AG 软件公司生产的 DBMS）开发并建立金属有机化合物质子坐标信息数据库。

加拿大国家重力数据库使用了 SYSTEM2000（层次模型的 DBMS）。

美国的戴尔斯伯里（Daresbury）实验室的科学和工程研究委员会用关系

数据库管理系统 RAPPORT 建立了数据采集系统,英国森林委员会也用 RAPPORT 管理了树木测量方面的观测数据及实验数据，可直接用交互式查询语言，也可用高级程序设计语言扩充其功能。

美国斯坦福大学开发的 SPLRAS 是一个汇编成的层次数据库系统,已在美国、加拿大和英国建立了各种科学数据库等。但是，科学数据库系统与商用数据库系统是有很大差别的。商业数据库系统的典型应用是企业、工厂的管理控制，如材料、帐单、价格、市场信息等综合数据的管理，建数据库的目的是为集中控制过程用的数据资料，建库部门可直接从中得到收益。

数据来源是公司和企业生产部门的管理用和控制用数据，这样的数据记录一般不特别长，基本由字母数据串组成，数值数据通常是整数、浮点数。用户使用数据的查询要求一般是对数据项定义值的询问，可包括报表显示及绘图输出，准确度和保密性要求高，绝对不允许有错。

科学数据库的典型应用是联机数据查询、数据计算、数据编辑。建库的目的以服务为主，提供已有的科学技术数据信息，促进了社会各部门的发展。建库部门不能直接从中得到效益。

数据来源于科技工作者的实验室、观测台及已有的出版物，数据类型可为固定长或可变长，一个单个记录可以很长，如微生物性状库数据记录可达 1 万字节，晶体结构数据库记录可达 7 千字节，基本粒子物理数据库中记录可达 8 万字节。数据可以是位串、矢量、数组或图象。查询用户可提出允许误差的范围检索，输出的显示要求可能是字母数字串，此上、下标的多维表及图形。由于它们以公益服务性为主，所以保密要求一般不高。

科学技术数据的性质比商用数据复杂，使用这类数据库的科学家和工程师要求数据相当精确，对数据库的管理也更为复杂。建库时需要仔细研究各因素和进行系统化的分析。在数据库中放置科学技术信息并不难，但获取有用的信息输出往往非常困难，目前还没有一个能适用于建立各科学数据库的管理系统，正如前面介绍过的，不少科学数据库的设计者与开发者正在使用现有的商用数据库管理系统，也有的部门结合具体的专用要求，开发了相应的数据管理程序。这样的科学数据库系统分为单一的数据库系统、多用户单一应用的数据库系统和多用户多应用的数据库系统。它们一般都包括如下三类程序，即数据检验和数据装入过程程序；数据库查询程序；与外部程序包如与统计分析包，分析算法包，报表及图形，数据显示及印刷输出等程序接口。

4.发展趋势

当今科学数据库系统向两个方向发展，一种是专业性数据库，系统规模视具体的需求和服务目标而异，既有小型系统，也有大型系统。如美国生物医学数据库中心的 MEDLINE 系统，是世界上最大的专业性数据库。另一种综合性数据库系统向以主机为中心的大型化发展，提供有多种学科的数据信息服务，如美国的 DIALOG，OBDIT，BRS 系统，欧洲的 ESA/IRS，日本的 JOIS 系统等都是国际性的联机数据库，其中位于美国加州帕罗阿尔托（Palo Alto）市的 DIALOG 系统是规模最大的数据库服务系统，供世界范围内 90 个国家的 10 万多个用户取用，现已有 350 个数据库，数据量达 675GB，每年正以 20%～30%的比例增加。

从用户方面看，联机检索系统提供的功能大都很近，但各个数据库都有

着不同的文档组织结构及检索命令语言，这使得用户在使用不同系统时必须重新学习，所以，各种联机科学数据库用法的标准化已提到日程上来，这包括进入系统标准化，键盘用法标准化，检索语言和命令标准化，截断及其符号用法标准化，中断功能标准化等。

此外，由于光盘（CD-ROM）的出现及发展，在一台 PC 机上就可存放 500MB 以上的信息，用户可购买光盘使用自己的 PC 机进行检索。还有些科学数据库在加强对检索结果的处理，以便更适合专业方面的服务。

建立科学数据库及其信息系统是计算机硬件和软件专家、专业科学家、数据收集者的智力劳动结合的产物，系统发展是计算机网络发展与资源共享的必然结果，特别是现代商业系统网（如美国的 Telenet 和 Tymnet）已允许数据库系统联机共享，使世界范围内的任何用户都可做到检索多个系统的数据库。

5.专用数据库技术——统计数据库

近年来，统计数据库的研究得到很大的发展，这是由于常规的商用数据库系统不能有效地支持统计与科学数据的存储及应用，而常规数据库所支持的基于事务处理的并发运算又不是统计与科学数据所必须支持的。因此开发新型的统计数据库是完全必要的。由于统计数据与科学数据在数据性质与所需的运算上都有十分相似的地方，所以可以将它们合并在一起用一个数据库管理系统来支持。统计数据库主用于统计与科学试验数据的存储以及进行统计分析和归纳。它与常规数据库有许多不同之处。

1.统计与科学数据的特点

（1）数据可分成分类属性集合与汇总属性集合。分类属性集合是包括汇总属性集合在内的整个记录的组合键。如在人口统计数据中，统计的年代、省、地、县、乡、职业、民族、性别等属于分类属性集合，而相应的人口值则属于汇总属性。由于在统计数据中只需将分类属性集中于一个属性的值进行变化，如性别的变化就可以更改成一个新的记录，所以采用常规的记录形式进行存储就会造成很大的重复。

（2）统计数据中的分类属性与科学试验数据中测量参数属性很相似。而它的汇总属性又与科学数据中的测量值属性又很相似，因此，科学数据也可借用分类属性及汇总属性。加之科学数据的整理也需要借助于整套统计算法，为此，我们将这 2 种数据用同一种数据库管理系统来支持。

（3）数据量一般很大。以人口统计数据而言，显然其数量是很大的，对科学数据来说，某些科学试验装置每天所产生的数据量可达数千兆字节。

（4）汇总属性中常常有大量相同的值（通常为零），只要将分类属性进行适当的排列，就可以将汇总属性中零值或相同的值聚合在一起，这就有利于对这些稀疏数据利用标头压缩技术进行压缩。

（5）统计运算往往只涉及一个或少数几个属性上的数据。如有常规数据库所采用以记录为单位对数据进行存储。这样每次存取就要存取整个记录，而实际使用只是其中一小部分，这是不合适的。所以应当在存储格式上采用了与常规记录格式完全不同的、以同一属性的值存放在一起的转置文件格式。

（6）所支持的运算与常规数据库也不一样。它除了一般的检索外还需要支持大量的统计运算和其他特有的操纵。

（7）统计数据库所保存的数据一般都反应已发生事件的记录。它们都是很稳定的，一般都不需要随时进行更新。在个别情况下即使要进行更改其频率也是很低的，而且更改前后的数据一般都需要记载下来。在常规数据库中，由于数据频频更新的要求，花费在并发控制上，进行事务处理的开销是很大的，而这种开销在统计数据库中是完全可以省去的。这就能使统计与科学数据在这方面的功能大为简化。

（8）具有元数据组织。描述有关数据集合的轮廓的数据称为元数据，无数据包含数据集合的生成者，生成的时间、生成的理由等信息，由于统计数据库数据量大，数据复杂，用户很容易把这方面的信息遗忘而影响使用，因此元数据的组织是很重要的。

（9）用户可抽取较小的数据集合以构成汇总集合。汇总集合通常可以采用类似选择操作的运算以选出有用的记录，可用类似投影操作的运算以选出有用的属性。

2.数据模型

为了适应统计与科学数据的特点，应当采用与常规数据库不同的数据模型。这种数据模型的要点是对于其分类属性采用由叉乘节点（简称 X 节点）及集结节点（简称 C 节点）所构成多维表格形式，也可以树的形式来表示。叉乘节点用于表示分类属性中多维的本质，如只用行与列则可表示属性的二维特点。鉴于一般都使用二维的表格，所以我们可以在根节点下人为地加上二个叉乘节点，以构成只有行和列的二维表格结构。集结节点（简称 C 节点）用于表示它们之间的族集特点。如行政区中的省市则有北京市、河北省等等。这样在下面所给出的简单人口统计表中的分类属性就可以用相应的由叉乘节点和集结节点构成的多维表格式或树形结构来表示。汇总属性数据则按照其相应的分类值进行定位，并根据不同的分类属性分别存储而构成转置文件。

由于在实际应用中人们遇到的统计与科学数据往往比较复杂，它们既十分接近上述数据模型但又不能有效地用上述模型来支持。在进行了较广泛的调查研究以后，我们提出了一种新的混合模型，可以支持这类数据。

统计数据库除了应具有与常规数据所类似的数据操纵，还应当有它自己所特有的数据操纵。如分类属性的位置变更、重新划分以及聚集运算等。由于机构变更等原因，会使分类属重新划分，而使其相应汇总属性值的归属重新安排。如海南建省，则与海南有关的数据应从广东省中分离出来，重新排列。

聚集运算的特点是：如在集结点下面给定其中某个节点则系统将针对该节点进行聚集，而在叉乘节点下未给定的节点，则应对其下面的所有节点进行聚集。

统计数据库的统计算法应当是相当完善的。一般应有矩阵算法、基于统计量计算、回归分析、统计参数估计、分布参数检验、非参数检验、数据平滑与滤波、经验分布曲线的选配、统计用表、相关分析等。统计与科学数据的二维输出格式是很复杂的。这种基本的表格输出格式也应由数据库管理系统提供。

3.数据压缩

数据压缩的优点有：能减少存储空间、增加数据传输率、加强保密性、减少后备副本和恢复费用等；其缺点是：需要支付压缩及还原的开销，破坏了数据的固有特性，降低可靠性，需要存储有关压缩的技术的数据，有时会使数据长度不能预定。压缩的方法有多种，如顺序保持法、霍夫曼法、标头压缩法等。由于统计数据库中往往有大量的相同的数据（如零值）相邻地集结在一起，对于这种数据采用标头压缩技术最为合适。标头压缩技术的基本思想是采用标头来标明所存数据串的类别，即标明是集结在一起的数据还是随机数据，是压缩前的数据顺序数，还是压缩后数据串末尾所处的字节数。这样压缩后对相邻地集结在一起的许多相同的数据可只存放一个数据。从而达到压缩的目的。标头压缩技术也可分为用于多相邻接数据和变长度数据的双计数压缩方案，以及单邻接数据和定长度数据的单计数压缩方案等。

4.安全措施

统计数据库又一个特点是它既要为各类用户方便提供有关群体的各种统计数据，又不能因此而泄露个体的数据，包括防止用户采用推导的办法而获得个体的信息，从而达到保密的目的。例如全国的或某省市的某种产品的产量是不保密的，但涉及某个具体工厂的这种产量是不应泄露的。在统计数据库中虽然不允许一般用户直接访问个体的机密信息，但在每个统计结果中都会有残留着原始个体数据的痕迹，如不采取适当的措施，用户就可以利用合法的统计数据推导而获得个体信息。为了防止泄露一般可采用下列安全措施。

（1）限制查询集合的大小。如果统计查询集合的大小低于或高于预定的阈值，则该查询将被拒绝。虽然限制查询集合的大小可以直接防止个体信息的泄露，但用户可以采用多次查询而推导出个体机密信息。

（2）限制查询集合交的大小。采用这种方法时，系统应当保存建库以来所有进行过的统计查询的痕迹，在每进行一次新查询时都要检查该新查询与进行的查询的交集的大小是否低于阈值，以决定是否执行或拒绝这个查询。

这种方法虽可确保个体信息不被泄露，但由于每个查询是否可执行都要追溯到自从数据库建立以来有关的情况，所以，系统要增加附加的存储信息。一般说来它只适用于规模较小的和用户较少的数据库。

（3）随机取样查询。采用这种方法，每次查询时，查询集体是随机地在数据库抽样而取得。这样即使通过多次查询也不可能正确地推导出个体信息，从而防止个体信息的泄露。这种技术适用于大型数据库。

（4）对数据值进行微扰。为了防止泄露个体信息，对统计查询的结果可用微小的数值进行打扰。这样就不会通过多次查询推导出正确的个体信息。所采用的办法可在查询时对输出的数据进行打扰，或者对存储在数据库中的数据进行打扰。它的问题是微扰的值应当取得很适当，不会因打扰所引入的误差太大而影响使用，又不能太小以至能推导出较正确的个体信息。

（5）对数据库进行划分。在许多应用中，可以预先规定若干个个体群，而只在这些群体上进行统计以满足查询的需要。由于个体群是预先规定的，所以就不能随意地进行查询而推导出所需个体机密信息。

这种方法的问题是：有时会因所定义的群体太小（只包含数量较少的个体）而为推导出个体信息提供方便。

统计数据库的安全措施的种类很多，除上述以外还可采用结合具体的数据模型、数据结构的各种安全措施。为了确保安全性，可根据具体使用环境以及所采用的数据模型等选用适当的安全措施。

在科学数据库中访问个体数据一般是允许的。因此，对于这类数据库一般不需要采用的特殊的安全措施。

5.结束语

从上面所介绍的情况可见，统计数据库都与常规数据库有很大的不同。因此开发和使用统计数据库管理系统是完全不同的。在实际应用环境中，应用所介绍的含有叉乘节点及集结节所构成的模型，有时不能很好存放过于稀疏的数据。为此，在系统中增加了一个混合模型，就拓宽了应用范围。

统计数据库管理系统可用来支持气象、水文、地质、地震等数据库，以及各企事业单位有关统计数据的数据库，人口统计数据库，各种大型实验装置的科学数据库。

6.数据仓库与数据挖掘

1.什么是数据仓库

数据仓库（Data Warhouse）是计算机应用领域里的一个崭新方向，它是一种信息管理技术，其研究的主要宗旨是通过通畅、合理、全面的信息管理，来达到对管理决策的支持。与联机事物处理（OLTP）相比，它完全是另一种类型的信息管理方式。

从概念上说，数据仓库是基础十分广泛，具有一种能力。它实质上是把运作数据转换成商业信息，帮助公司解决许多不同的复杂商业难题。从技术上说，数据仓库是企业内部各单位的运作数据和事务数据的中央仓库，这些数据经过了归化、平衡、协调和编辑。它是为最终用户进行分析处理而专门设计的，使最终用户可以针对任何一个经营单位或整个企业、用任何一个需要的参数去存取市场数据以及客户、产品或事务的信息。这种能力明显有别于以前的其他方法。那些方法实际上是把客户数据锁在一直被叫做“数据监狱”的数据库里。数据库已演变成分散的、独立的子系统，没有能力从统一的角度提供客户的有关信息，或指出哪些服务和产品与所有客户的关系最密切。

数据仓库有能力对整个企业各部门送来的各种信息进行统一和综合，这实际上是决策支持和客户管理的一次革新。以数据仓库在银行的应用为例，银行可以用它来取得各个重要方面的数据与分析结果，例如利润、市场分析和风险管理等，进而改善银行的自身管理。比如，数据仓库用户可以立即得到其单位当前所处地位的准确报告；了解其公司面临的风险，包括备项事务及整个银行所有业务面临的风险；对市场和法规条例的需要迅速作出反应。此外，数据仓库对于客户管理和营销还有许许多多的好处。由于银行能够看到所有帐户和每个人的信息，因而银行有能力真正了解客户并更好地向他们提供服务。另外，之所以要把涉及每件事情的大量信息都集中到数据库，总的想法是要在客户的各个生活阶段中知道该客户能使银行前进到哪里，并提供所谓的“预期服务”。换句话说，银行将能够在客户还未认识到他的某种需要之前就预测到他的需要。比方说，银行将知道客户的汽车已用了 4 年，

所以将建议向他提供一笔汽车贷款，帮助他更快地买到新车；银行将知道新生儿出世的消息，并向其家庭建议一个更高层次的教育计划。了解到的客户信息越多，银行就越能够更好地预测下一个潜在的业务，并通过交叉推销来提供更多的服务。只有数据仓库环境（它也包含其他的外界信息，如市场统计信息等）才能提供这种信息。

数据仓库不是一个静止不变的产品，而是一个动态的、不停变化的过程——这个过程为全企业的管理系统奠定信息基础。该系统可用来测算利润、管理和分析风险、进行市场分析、帮助规划和加强客户服务计划及市场推进计划。与现买现装的产品不同，成功的数据仓库实际上是一个过程。它要求公司仔细分析本公司的基本原则，决定需要哪些运作数据和外部数据源，然后利用一种严密的方法把所有的数据集中起来，再变换成有用的信息。

数据仓库过程一旦开始实施，就没有终结的时候。它的可用性和中肯性在极大程度上来自于其信息的新鲜性。因此，公司必须不断对它进行更新，馈入新的统计信息和新的事务档案。

有人认为在今后，数据仓库将成为保持竞争优势的因素之一。如何更好、更快地把产品推入市场？如何更好地为客户服务？如何更好地揽到客户？”越来越多的公司选择数据仓库作为答案。数据仓库是测算利润、管理和分析风险、进行市场分析，以及加强客户服务与营销活动等的催化技术；它在支持和管理突飞猛进的商业变化以及在保持这种竞争优势方面已日益扮演着举足轻重的角色。

2.数据仓库的主要功能和特点

数据仓库的主要功能是提供企业决策支持系统（DSS）或行政和信息系统（EIS）所需要的信息，它把企业日常营运中分散不一致的数据经归纳整理之后转换为集中统一的、可随时取用的深层信息，这种信息虽然也是按关系数据库的存储结构存储起来的，但与面向逐条记录的 OLTP 不同，在数据仓库中的一条记录，有可能是基础数据中若干个表、若干条记录的归纳和汇总。

数据仓库的基本特点为：

（1）数据仓库存储的信息是面向主题来组织的。它根据所需要的信息，分不同类、不同角度等主题把数据整理之后存储存来（按横向对数据进行分类存储）。

（2）数据仓库中要有一处专门用来存储 5 至 10 年或更久的历史数据，以满足比较、预测之用的数据需求（按纵向对数据进行分类存储）。

（3）不论数据来源于何处，进入数据仓库之后都具有统一的数据结构和编码规则，数据仓库中的数据具有一致性的特点。

（4）数据仓库是一个信息源，它只是为在其上开发的 DSS 或 EIS 等提供数据服务，因此它应是只读数据库，一般不轻易做改动，只能定期刷新。

3.数据仓库的基本结构

数据仓库中的信息存储，是根据对数据的不同深度处理来分成不同层次的。其结构一般划分为以下几个方面：

（1）历史性详细数据层。它存储历史数据，供分析、建模、预测之用。

（2）当前详细数据层。存储最新详细数据，是进一步分析数据的基础。

（3）不同程序的归纳总结信息层。可包含多个层次，根据所需分类和归

纳的不同深度而定，如按周、月、年统计的数据。

（4）专业分析信息层。进一步专业分析的结果，如统计分析、运筹分析、时间序列分析以及表面数据的内在规律分析等。

（5）结构信息。数据仓库的内部结构信息，反映各种信息在数据仓库中的位置分布和处理方式等，以便检索查询之用。

4.数据仓库环境

从传统的应用环境向信息启动的、以数据仓库为中心的环境转移是必然的。问题只是何时转移。

随着这一转移而来的问题是：数据仓库环境的确切意义是什么，以及它是如何工作的？

数据仓库是设计来为整个机构的信息需求服务的。为实现这一点，数据仓库在不同的粒度层次存储数据——从当前的详尽数据到高度汇总的数据。作为规律，越新的数据，越是马上就要用。一般而言，当前详尽数据支持日常决策，而历史数据支持趋势分析和长期决策。

对数据仓库环境的要求之一是积累和管理大量数据的能力。因此，为仓库中的数据适当选择粒度层次和汇总是很重要的。为管理仓库中大量数据而考虑的其他设计方法和技术还有：在多个存储媒体上存储数据，当数据变得陈旧时做汇总，按人工做的条目存储数据关系，在合适处对数据进行编码和建立基准，为独立管理和索引对数据进行分区等。

下文对构成数据仓库结构中每个构件的作用进行入门性的介绍。

（1）当前详尽数据数据仓库环境的核心是当前详尽数据，是大量数据留驻的地方，常常存在并行处理器上。当前详尽数据是直接由正在运转的传统环境送来的，代表着整个公司，而不是某个应用。当前详尽数据是按隶属脉络组织起来的。

当前详尽数据中的每个数据单位可以看作是一个瞬象，此时一个时间单位可以识别出一瞬间，在此瞬象是精确的。当前详尽数据代表着数据仓库环境中可找到的最低层次的数据粒度。它可以作为原始数据工作为概貌存储——它代表原始数据的聚合。当前详尽数据通常为 2～5 年。它按环境的需要不断地按日、周或月进行更新。

（2）过去详尽数据这是档案数据或通常超过 2 年的数据存储的地方。通常在过去详尽数据层次上存储着大量的数据，它们的存取概率不高。过去详尽数据与当前详尽数据粒度上处于同一层次。随着数据进入过去详尽等级，为了对它进行浓缩，数据可以集合在一起或作概貌加工。由于数据结构随时间的变化而变化，所以过去详尽数据一般都包含着同一数据结构的多种版本。它可以存储在多种不同的媒体上。

（3）部门/数据市场。轻度汇总数据是数据仓库中部门成分的印记。部门层次是经定制而适应拥有此数据部门之需求。定制是在数据从当前详尽送到部门层次时完成的；部门层次是专门由当前详尽层次馈送的。在任何一个给定部门数据库中数据量要比当前详尽数据少提多。部门层次包含了详尽与汇总两种数据。数据送入部门层次时的汇总过程是元数据的重要部分。数据的部门层次方便地使用关系技术进行多维分析。

（4）高度汇总数据。在数据仓库环境中数据的高度汇总层次是为高级主管人员设计的，应该允许通过向下挖掘的过程访问层次不断提高的详尽数

据。高度汇总数据来自部门层次的数据或当前详尽数据层次。此处找到的数据量远少于其他数据仓库层次上的数据量，它代表一种折中的集合，支持多种多样的需求和利益。

（5）记录系统。建立数据仓库的最初阶段，记录系统是在馈送和支持数据仓库的应用程序中可找到的数据。记录系统应始终代表着一家公司拥有‘最佳”数据，此处“最佳”被定义成最及时、完整和精确的数据，拥有集成数据模型的最佳结构一致性，并留驻在最靠近工作环境的入口处。记录数据系统并不一定是十全十美的，但当它进入数据仓库时，它经历了重要的编辑、净化和重新格式化的过程。

（6）集成/传输程序。当数据从记录系统进入数据仓库时，它经历了一组集成和传输的程序，这组程序把应用程序特有的数据转变成公司数据。这些程序执行下列一些功能：重新格式化、重新计算、修改关键结构、增加时间单元、识别缺省值、提供多个数据源之间进行选择的逻辑、做汇总、清点和合并多个来源的数据等。集成和传输程序在每次工作环境或数据仓库环境改变时都需要作修改。

（7）数据仓库环境的最后一个成分是元数据，或关于数据的数据。它留驻在数据仓库内数据的所有层次上，但它是在与数据仓库内的其他数据不同维度上存在和运行的，由于这个原因，元数据常常被误认为是授予的或错误理解的。

元数据是数据仓库环境最重要的方面之一。它存在于仓库开发和最终用户应用两个层次。元数据由数据仓库开发人员用于管理和控制数据仓库的生成与维护。对于最终用户，元数据留驻在本身的数据仓库平台上，并可以作为访问和分析数据仓库的正常的一部分。

为了成功，数据仓库系统必须是易建立、易管理、易使用。因此，重要的是了解它的目标和要求，以及确定供应商的产品能在多大程度上满足数据仓库设计人员、管理人员和业务用户的需求。

数据仓库系统的关键部件包括下列内容：

1）定义部件，用于定义和建立数据仓库系统；

2）数据采集部件，用于把数据从源文件和数据库复制到数据仓库数据库；

3）管理部件，用于管理数据仓库工作；

4）数据分配部件，用于把仓库数据外输到外部系统；

5）信息目录部件，用于提供有关储存在数据仓库里的数据库中数据的信息；

6）DBMS（数据库管理系统）部件，用于管理、维护和存取仓库数据；

7）数据存取和分析部件，用于向业务最终用户提供存取和分析仓库数据所需的工具。

（1）定义部件。定义部件由（数据）仓库设计人员和管理人员用来；①设计和定义数据仓库的数据库；②定义由仓库数据获得的数据来源；③确定在将数据从源系统向数据仓库的数据库复制时进行的数据清理和增强的规则。此部件的输出作为信息目录部件的元数据储存。

（2）数据采集部件。数据仓库系统的主要目标之一是要以业务用户容易理解和使用的形式存放公司数据。数据采集部件通过把数据从源系统中提取出来并依据定义部件定义的规则清理和交换数据，从而达到上述目标。清理

可能需要对记录或字段的重建结构、去除只与日常运作有关的数据、对字段值的译码和解释、提供丢失的字段值，以及检查数据的完整性和一致性、变换可能涉及增加时间字段（如果在源数据中没有）来反映数据、数据摘要或推导值计算的现时性。一旦源数据已被清理和变换，它就被映象到目标仓库的数据库、输送到数据仓库系统以及装入（或更新）相应的仓库数据库。仓库数据库的装入（或更新）是由 SQL（假设采用关系 DBMS）或者数据库装入实用程序完成的。

采集数据并把它复制到数据仓库系统有很多种方法。工业界的走向是混合使用编码生成器和数据复制工具。

（3）管理部件。管理部件由一组供其他仓库部件使用和管理仓库数据集合的服务组成。数据集合是一组对特定用户或用户群感兴趣的数据。数据集合是从数据采集部件生成的基础数据中得到的。管理部件提供的服务包括对从仓库的基本数据推导出来的新数据集合的数据维护服务、把仓库数据外送至分散的仓库数据库服务器和其他最终用户决策支持系统的分配服务。管理部件还提供对库数据和数据集合的安全、归档、备份和恢复以及监测等项的处理服务。这些服务常常要用到基本操作系统和数据库软件提供的一些功能。

（4）信息目录部件。数据仓库的信息目录部件包含了关于仓库数据库中数据的信息（称作元数据）。信息目录的主要长处是有助于业务用户了解仓库存放了哪些信息以及如何访问和使用它们。

信息目录的三大部分是技术目录、业务目录以及信息引导器。

技术目标包含了有关供仓库设计人员和管理人员使用仓库数据信息的。它拥有关于数据源、目标、清理规划、交换规则以及数据源和仓库数据库之间映象的信息。技术目录中的多数信息是在仓库设计人员定义数据源和目标时，以及定义把数据复制到仓库中运用的规则时生成的。它也可以由外部系统送入，如第三代语言的复制书库、DBMS 系统目录表或 CASE 工具。

有关仓库中的数据量和生成或更新的日期的信息也应存储在目录中。理想的情况是，这种信息应该由使用的工具收集，以便从源系统采集数据并把它送到仓库数据库。关于最终用户如何访问和使用仓库数据的信息也应该进行捕捉，并加到技术目录上，以使设计人员和管理人员凋整和增强数据仓库。

业务目录包含了给予最终用户仓库中数据容易理解的视图的信息。这种信息为：

1）用于访问仓库数据的业务术语及有关的技术名称和别名；

2）仓库数据源，导出规则和数据的当前值；

3）有关数据拥有者的联系信息；

4）预定义的查询和报表的细节；

5）合法性要求。

这种业务信息通常是由仓库管理员生成的，但它也可以从外部系统，如 CASE 工具或查询与报表书写工具输入。

6）信息引导器使最终用户容易访问业务目标和仓库数据。引导器应该提供：

①查询与引导功能，以访问和通过业务目标中的信息查找。②利用固定查询或通过访问助理建立新的查询，生成暂时的或永久的仓库数据集合之能力。③向仓库管理员发送新的数据采集请求之通信功能。④向数据分配部件

发送请求以把已有的仓库数据集合输送给另一个数据仓库或者外部系统之功能。⑤与数据分配和数据访问部件的天衣无缝般的接口。

迄今，供应厂商对信息目录中三大部件的支持还是有限的。此领域中的一些重要开发工作正在进行之中。

随着数据仓库使用的扩大，具有业务目录及相关的信息引导器的综合信息目录功能将成为最终用户全面使用数据仓库能力之基础。此功能将成为区别各数据仓库产品的关键因素。

（5）DBMS 部件。DBMS 部件由维护和检索仓库数据的数据库软件组成。在为数据库仓库系统选购数据库产品时两项关键考虑是可扩缩性和速度性能。一旦仓库的价值被认识后，仓库数据库通常就非常快速增长，DBMS 能否高度可扩缩便是至关紧要的了。由于仓库数据库可以含有大量千兆字节的数据，故数据库产品在处理这些极大的数据库数据时必须能提供高速性能。

为在装载、访问和分析大量数据时解决性能问题，各公司推出了并行处理数据库产品。

（6）数据访问和分析部件。数据访问和分析部件提供了让用户研究和分析数据仓库的工具，以便让他们改进决策和增强竞争优势。这些工具包括从查询生成工具到复杂数据分析的多维产品以及数据采集工具等。

5.数据仓库所面临的主要问题

数据仓库是随着企业对于 DSS 或 EIS 不断增长的市场需求，以及现实中存在的大量重复工作等问题应运而生的。目前，大多数 DSS 或 EIS 的数据处理工作都是由应用程序本身完成的，这是一种极不经济和效率低的做法。一个好的决策支持系统，其 90%以上的数据处理工作应在数据仓库中完成。然而，要建造一个实用的数据仓库，必须首先解决以下几个问题：

（1）对大量的不同格式、跨越不同软件平台的企业中一般营运数据要能及时、有效地访问到。

（2）对访问到的基本数据要能进行有效的分类、合并、归纳、整理以及深层次的分析和处理。

（3）必须具备一个合理的数据存储结构。

（4）建造的数据仓库具有开放性，使其不仅能为某一专门系统提供服务，更能被其他应用系统访问到，成为众多信息系统的物理信息源。

6.数据挖掘

（1）什么是数据挖掘。在今天的市场上，信息的利用至关重要，各行各业面临激烈的竞争及经济压力，产品的生命周期缩短，需要为顾客提供更好的服务。在过去几年中，各公司为了取得必要的市场战略信息及对付市场方面的各种压力，已经开始采用数据仓库技术。各公司为了确定所要开发的产品模式及了解市场走势，需要提取数据仓库数据，包括联机事务处理（OLTP）数据，并与外部的人口统计数据及心理数据结合，从中“挖掘出”最终结果。利用这种数据仓库信息源，知识工作者在他们的办公室内可根据所取得的数据，就可以进行决策。数据仓库直接影响事关公司命运的决策。

上述过程被称为数据挖掘（data mining），实施这一过程的基本设施是数据仓库。这是一种关键性、涉及范围很广的技术手段。利用数据挖掘技术可使潜在的效益得到最大的发挥。数据仓库是一种数据集成战略，目的是促

进最终用户利用企业数据，同时保护公司的数据财富——关键任务的可操作数据——安全性和完整性。

只要安排妥当，数据仓库就能发挥它的重要作用，即人们可以很快地作出决策。因此，数据仓库是实施公司战略的一种技术手段。

一般来说，构筑数据仓库是一个频繁的查阅过程，它可分为若干阶段，其中包括需求分析、数据仓库的设计、操作数据的提取、不相容数据的集成、数据仓库的装填、最终交付用户使用。在后续期内，还应该对数据仓库作定期更新。

数据挖掘对发挥数据仓库的作用有很大影响，因通过它可以识别出商务中的模式与趋势，而仅通过分析数据仓库数据是无法得出的。当知识工作者运用结构化查询语言（SQL）对数据仓库查询所需的信息时，查询中的歧义性常常涉及到与答案集有关的一系列知识。相反地，数据挖掘可以揭示出非常有价值的信息，这些信息在实施分析之前，知识工作者是无法得知的。这种新技术，有助于使公司取得较大的市场份额，建立更好的形象并推动公司向前发展。

（2）数据挖掘的应用。数据挖掘的应用非常广泛，下面介绍一些的应用可以说明数据挖掘的用途。

1）商品销售。商业部门把数据视作一种竞争性的财富可能比任何其他部门显得更为重要，为此需要把大型市场营销数据库演变成一个数据挖掘系统。科拉福特（Kraft）食品公司（KGF）是应用市场营销数据库的公司之一，该公司搜集了购买它商品的 3000 万个用户的名单，这是（KGF）通过各种促销手段得到的。KGF 定期向这些用户发送名牌产品的优惠券，介绍新产品的性能和使用情况。该公司体会到了解自己商品的用户越多，则购买和使用这些商品的机会也就越多，公司的营业状况也就越好。

2）制造。许多公司不仅决策支持系统用于支持市场营销活动，而且，由于市场竞争越演越烈，这些公司已使用决策支持系统来监视制造过程，有制造商声称已经指示它的各个办事机构，在三年内把制造成本每年降低 25%。不言而喻，该制造商经常收集各部件供应商的情况。因为，它们也必须遵循该制造商降低成本的战略。为了对付来自各方的挑战，该制造商已拥有一套“成本”决策支持系统，可以监视各供应商提供的零部件成本，以实现所制定的价格目标，这种应用需要收集有关各厂商连续一年来的产品成本信息，以便确定这种组织方式能否满足原先制定的有关降价的战略目标。

3）金融服务/信用卡。通用汽车公司（General Motors）已经采用信用卡——GM 卡，在该公司的数据库中已拥有 1200 万个持有信用卡的客户。公司通过观察，可以了解他们正在驾驶什么样的汽车，下一步计划购买什么样的汽车及他们喜欢哪一类车辆。譬如说，一个持有信用卡的客户表示对一种载货卡车感兴趣，公司就可以向卡车部门发出一个电子邮件，并把该客户的信息告诉有关部门。

4）远程通讯。许多远程通讯的大公司近来突然发现它们面临极大的竞争压力，这在几年前是不存在的。在过去，业务上并不需要他们密切注视市场动向，因为顾客的挑选余地有限，但是这种情况近来发生很大变化。各公司当前都在积极收集大量的顾客信息，向他们现有的客户提供新的服务，开拓新的业务项目，以扩大他们的市场规模。从这些新的服务中，公司在短期内就可以取得更大的效益。