第3章

联机分析处理

在数据仓库系统中，联机分析处理（OLAP）是重要的数据分析工具。OLAP的基本思想是企业的决策者应能灵活地，从多方面和多角度以多维的形式来观察企业的状态和了解企业的变化。

3.1 OLAP概念

在信息爆炸的时代，信息过量几乎成为人人都需要面对的问题。如何才能不被信息的汪洋大海所淹没，从中及时发现有用的知识或者规律，提高信息利用率呢？要想使数据真正成为一个决策资源，只有充分利用它为一个组织的业务决策和战略发展服务才行，否则大量的数据可能会成为包袱，甚至成为垃圾。OLAP是解决这类问题最有力的工具之一。

OLAP专门设计用于支持复杂的分析操作，侧重对分析人员和高层管理人员的决策支持，可以应分析人员的要求，快速、灵活地进行大数据量的复杂查询处理，并且以一种直观易懂的形式将查询结果提供给决策制定者，以便他们准确掌握企业（公司）的经营状况，了解市场需求，制定正确方案，增加效益。OLAP软件以它先进的分析功能和以多维形式提供数据的能力，正作为一种支持企业关键商业决策的解决方案而迅速崛起。

3.1.1 OLAP的定义

在决策活动中，决策人员需要的数据往往不是单一指标的单一的值，他们希望能够从多个角度观察某个指标或者某个值，或者找出这些指标之间的关系。例如，决策者可能想知道“东北地区和西南地区今年一季度和去年一季度在销售总额上的对比情况，并且销售额按10万元～50万元、50万元～100万元，以及100万元以上分组”。上面的问题是比较有代表性的，决策所需数据总是与一些统计指标如销售总额、观察角度（如销售区域、时间）和不同级别的统计有关，可以将这些观察数据的角度称为维。可以说决策数据是多维数据，多维数据分析是决策分析的主要内容。但传统的关系数据库系统及其查询工具对于管理和应用这样复杂的数据显得力不从心。

OLAP是在OLTP的基础上发展起来的，OLTP是以数据库为基础的，面对

的是操作人员和低层管理人员，对基本数据的查询和增加、删除、修改等进行处理。而OLAP是以数据仓库为基础的数据分析处理。它有两个特点：一是在线性(Online)，体现为对用户请求的快速响应和交互式操作，它的实现是由客户/服务器这种体系结构在网络环境上完成的；二是多维分析(Multidimensional Analysis)，这也是OLAP的核心所在。

OLAP超越了一般查询和报表的功能，它是建立在一般事务操作之上的另一种逻辑步骤，因此，它的决策支持能力更强。在多维数据环境中，OLAP为终端用户提供了复杂的数据分析功能。通过OLAP，高层管理人员能够通过浏览、分析数据发现数据的变化趋势、特征以及一些潜在的信息，从而更好地帮助他们了解商业活动的变化。目前，普遍为人们所接受的OLAP的定义有两种。

1. OLAP理事会给出的定义

OLAP是一种软件技术，它使分析人员能够迅速、一致、交互地从各个方面观察信息，以达到深入理解数据的目的。这些信息是从原始数据转换过来的，按照用户的理解，它反映了企业真实的方方面面。

企业的用户对企业的观察自然是多维的。例如销售，不仅可以从生产这方面看，还与地点、时间等有关，这就是为什么要求OLAP模型是多维的原因。这种多维用户视图通过一种更为直观的分析模型进行设计和分析。

OLAP的大部分策略都是将关系型的或普通的数据进行多维数据存储，以便于进行分析，从而达到联机分析处理的目的。这种多维数据库也被看作超立方体，沿着多个维存储数据，为用户沿着任意多个维事务的便利地分析数据。

2. OLAP简单定义

近来，随着人们对OLAP理解的不断深入，有些学者提出了更为简要的定义，即联机分析处理是共享多维信息的快速分析(Fast Analysis of Shared Multidimensional Information，FASMI)，它体现了4个特征。

(1) 快速性(Fast)：用户对OLAP的快速反应能力有很高的要求。系统应能在5s内对用户的大部分分析要求做出反应，如果终端用户在30s内没有得到系统的响应，则会变得不耐烦，改变分析主线索，影响分析的质量。

(2) 可分析性(Analysis)：OLAP系统应能处理与应用有关的任何逻辑分析和统计分析。尽管系统需要一些事先的编程，但并不意味着系统事先已将所有的应用都定义好了。

(3) 多维性(Multidimensional)：多维性是OLAP的特点。系统必须提供对数据分析的多维视图和分析，包括对层次维和多重层次维的完全支持。

(4) 信息性(Information)：不论数据量有多大，也不管数据存储在何处，OLAP系统都应能及时获得信息，并且管理大容量的信息。

用于实现OLAP的技术主要包括网络环境上客户/服务器体系结构、时间序列分析、面向对象、并行处理、数据存储优化等。

3.1.2 OLAP 准则

1985 年以来,关系数据库需求始终受到 E. F. Codd 提出的 12 条规则的影响。1993 年,E. F. Codd 在 *Providing OLAP to User Analysts* 一书中又提出了有关 OLAP 的 12 条准则,用来评价分析处理工具,这也是他继关系数据库和分布式数据库提出的两个"12 条准则"后提出的第三个"12 条准则"。由于这些规则最初是对客户研究的结果,所以业界对这个 12 条准则褒贬不一。但其主要方面,如多维数据分析、客户/服务器结构、多用户支持及一致的报表性能等方面还是得到了大多数人的认可。E. F. Codd 在文中系统阐述了有关 OLAP 产品及其所依赖的数据分析模型的一系列概念及衡量标准,这对 OLAP 产品的辨别及后来的发展方向的确立都产生了重要的作用。如今,这 12 条规则也成为大家定义 OLAP 的主要依据,被认为是 OLAP 产品应该具备的特征。如今 OLAP 的概念已经在商业数据库领域得以广泛使用,E. F. Codd 提出的 OLAP 主要的 6 条准则如下。

1. 多维概念视图

从用户分析员的角度来看,用户通常从多维角度来看待企业,企业决策分析的目的不同,决定了分析和衡量企业的数据总是从不同的角度进行,所以企业数据空间本身就是多维的。因此,OLAP 的概念模型也应是多维的。用户可以简单、直接地操作这些多维数据模型。例如,用户可以对多维数据模型进行切片、切块、改变坐标或旋转模式中的联合(概括和聚集)数据路径。

2. 一致稳定的报表性能

报表操作不应随维数增加而削弱,即当数据维数和数据的综合层次增加时,提供给最终分析员的报表能力和响应速度不应该有明显的降低,这对维护 OLAP 产品的简易性至关重要。即便是企业模型改变,关键数据的计算方法也无须更改。也就是说,OLAP 系统的数据模型对企业模型应该具有"鲁棒"性。只有做到这一点,OLAP 工具提供的数据报表和所做的预测分析的结果才是可信的。

3. 客户/服务器体系结构

OLAP 是建立在客户/服务器体系结构之上的。这要求它的多维数据库服务器能够被不同的应用和工具所访问,服务器端以最小的代价完成同多种服务器之间的挂接任务,智能化服务器必须具有在不同逻辑的和物理的数据库之间映射并组合数据的能力,还应构造通用的、概念的、逻辑的和物理的模式,从而保证透明性和建立统一的公共概念模式、逻辑模式和物理模式。客户端负责应用逻辑及用户界面。

4. 维的等同性

每一数据维在其结构和操作功能上必须等价。可能存在适用于所有维的逻辑结构,提供给某一维的任何功能也应提供给其他维,即系统可以将附加的操作能力授给所选维,但必须保证该操作能力可以授给任意的其他维,即要求维上的操作是公共的。该准则实

际上是对维的基本结构和维上的操作的要求。

5. 动态的稀疏矩阵处理

OLAP 服务器的物理结构应完全适用于特定的分析模式，创建和加载此种模式是为了提供优化的稀疏矩阵处理。当存在稀疏矩阵时，OLAP 服务器应能推知数据是如何分布的，以及怎样存储才更有效。

6. 多用户支持能力

当多个用户在同一分析模式上并行工作，或是在同一企业数据上建立不同的分析模型时，OLAP 工具应提供并发访问、数据完整性及安全性等功能。

实际上，OLAP 工具必须支持多用户也是为了适合数据分析工作的特点。应该鼓励以工作组的形式来使用 OLAP 工具，这样多个用户就可以交换各自的想法和分析结果。

3.1.3　OLAP 的基本概念

OLAP 是针对特定问题的联机数据访问和分析。通过对信息进行快速、稳定一致和交互性的存取，允许管理决策人员对数据进行深入观察。为了对 OLAP 技术有更深入的了解，这里主要介绍在 OLAP 中常用的一些基本概念。

1. 变量

变量是数据的实际意义，即描述数据“是什么”。例如，数据 100 本身并没有意义或者说意义未定，它可能是一个学校的学生人数，也可能是某产品的单价，还可能是某商品的销售量，等等。一般情况下，变量总是一个数值度量指标，例如，“人数”“单价”“销售量”等都是变量，而 100 则是变量的一个值。

2. 维

维是人们观察数据的特定角度。例如，企业常常关心产品销售数据随着时间推移而产生的变化情况，这时是从时间的角度来观察产品的销售，所以时间是一个维（时间维）。企业也时常关心自己的产品在不同地区的销售分布情况，这时是从地理分布的角度来观察产品的销售，所以地理分布也是一个维（地理维）。其他还有产品维、顾客维等。

3. 维的层次

人们观察数据的某个特定角度（即某个维）还可以存在细节程度不同的多个描述方面，通常称这多个描述方面为维的层次。一个维往往具有多个层次。例如，描述时间维时，可以从日期、月份、季度、年等不同层次来描述，那么日期、月份、季度、年等就是时间维的层次；同样，城市、地区、国家等构成了地理维的层次。

4. 维成员

维的一个取值称为该维的一个维成员。如果一个维是多层次的，那么该维的维成员

是由各个不同维的层次的取值组合而成的。例如,考虑时间维具有日期、月份、年这3个层次,分别在日期、月份、年上各取一个值组合起来,就得到了时间维的一个维成员,即“某年某月某日”。一个维成员并不一定在每个维的层次上都要取值,例如,“某年某月”“某月某日”“某年”等都是时间维的维成员。对一个数据项来说,维成员是该数据项在某维中位置的描述。例如,对一个销售数据来说,时间维的维成员“某年某月某日”就表示该销售数据是“某年某月某日”的销售数据,“某年某月某日”是该销售数据在时间维上位置的描述。

5. 多维数组

一个多维数组可以表示为(维1,维2,…,维n,变量)。例如,若日用品销售数据是按时间、地区和销售渠道组织起来的三维立方体,加上变量“销售额”,就组成了一个多维数组(地区,时间,销售渠道,销售额),如果在此基础上再扩展一个产品维,就得到一个四维的结构,其多维数组为(产品,地区,时间,销售渠道,销售额)。

6. 数据单元(单元格)

多维数组的取值称为数据单元。当多维数组的各个维都选中一个维成员时,这些维成员的组合就唯一确定了一个变量的值。数据单元就可以表示为(维1维成员,维2维成员,…,维n维成员,变量的值)。例如,在产品、地区、时间和销售渠道上各取维成员“牙膏”“上海”“2004年12月”“批发”,就唯一确定了变量“销售额”的一个值(假设为100 000),则该数据单元可表示为(牙膏,上海,2004年12月,批发,100 000)。

3.2 OLAP的数据模型

建立OLAP的基础是多维数据模型,多维数据模型的存储可以有多种不同的形式。MOLAP和ROLAP是OLAP的两种主要形式,其中MOLAP(Multidimensional OLAP)是基于多维数据库的OLAP,简称多维OLAP;ROLAP(Relation OLAP)是基于关系数据库的OLAP,简称关系OLAP。还有几种OLAP,如WOLAP(Web OLAP)代表网络OLAP,HOLAP(Hybrid OLAP)代表混合OLAP。

3.2.1 MOLAP数据模型

MOLAP数据模型是基于多维数据库的OLAP,多维数据库(Multidimensional Database,MDDB)是以多维方式组织数据,即以维作为坐标系,采用类似于数组的形式存储数据。多维数据库中的元素具有相同类型的数值,如销售量。例如,MDDB(二维数组,即矩阵)的数据组织如表3.1所示。它代表不同产品(衣服、鞋、帽)在不同地区(北京、上海、广州)的销售量情况。

表 3.1　MDDB(二维)的数据组织

产品名	地　区		
	北京	上海	广州
衣服	600	700	500
鞋	800	900	700
帽子	100	200	80

在查询中除查询一般的“衣服在广州的销售量”外,有时查询像“衣服的总销售量”等类问题,它涉及多个数据项求和,如果采取临时进行累加计算,会使查询效率大大降低。为此,需要增加汇总数据项。在多维数据库中只需要按行或列进行求和,增加“总和”的维成员即可,如表 3.2 所示。

表 3.2　MDDB 中含综合数据的数据组织

产品名	地　区			
	北京	上海	广州	总和
衣服	600	700	500	1800
鞋	800	900	700	2400
帽子	100	200	80	380
总和	1500	1800	1280	4580

MDDB 的数据组织形式不同于关系数据库的组织形式,关系数据库是以“属性—元组(记录)”形式组织数据。对表 3.1 中的数据按关系数据库组织,数据如表 3.3 所示。

表 3.3　关系数据库的数据组织

产品名	地区	销售量	产品名	地区	销售量
衣服	北京	600	鞋	广州	700
衣服	上海	700	帽子	北京	100
衣服	广州	500	帽子	上海	200
鞋	北京	800	帽子	广州	80
鞋	上海	900			

可见,MDDB 比关系数据库表达更清晰且占用的存储少。在关系数据库中增加综合数据项,如表 3.4 所示。这些综合数据项一般在建立数据库的同时计算出来。这样在查询时,不必临时进行计算,提高了查询效率。对于多维数据库的综合数据项明显比关系数据库的综合项更有效果。

表 3.4 关系数据库中含综合数据的数据组织

产品名	地区	销售量	产品名	地区	销售量
衣服	北京	600	鞋	广州	700
衣服	上海	700	鞋	总和	2400
衣服	广州	500	帽子	北京	100
衣服	总和	1800	帽子	上海	200
鞋	北京	800	帽子	广州	80
鞋	上海	900	帽子	总和	380

3.2.2 ROLAP 数据模型

ROLAP 是基于关系数据库的 OLAP,如表 3.3 所示。它是一个平面结构,用关系数据库表示多维数据时,采用星形模型,即用两类表:一类是事实表,存储事实的实际值,如销售量;另一类是维表,对每个维至少有一个表来存储该维的描述信息,如产品的名称、分类等。星形模型完全用二维关系表示了数据的多维观念。

通过关系数据库实现多维查询时,通过维表的主码对事实表和每个维表做连接操作,一次查询就可以得到数据的具体值以及对数据的多维描述(即对应在各维上的维成员)。但是,因为对每个维都需要进行一次连接操作,所以系统的性能就成了 ROLAP 实现的最大的一个问题,特别是当维数增加和事实表增大时,必须采用有效的查询优化技术(特别是表连接策略),利用各种索引技术来提高系统的性能。

当存在多层次的复杂维时,需要采用雪花模型,用多张表来描述一个复杂维。对于存在综合数据时,需要建立汇总事实表,采用星网模型来描述。

3.2.3 MOLAP 与 ROLAP 的比较

MOLAP 通过多维数据库引擎从关系数据库(DB)和数据仓库(DW)中提取数据,将各种数据组织成多维数据库,存放到 MDDB 中,而且将自动建立索引并进行预综合(见 3.4.4 节)来提高查询存取性能,如图 3.1 所示。

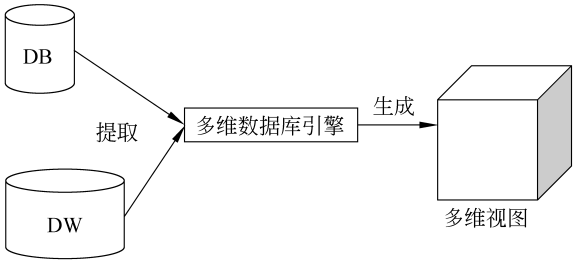

图 3.1 MOLAP 结构

ROLAP 从关系数据库(DB)和数据仓库(DW)中提取数据,按 ROLAP 的数据组织

存放在关系数据库管理系统(Relational Database Management System,RDBMS)服务器中。最终用户的多维分析请求,通过 ROLAP 服务器的多维分析(OLAP 引擎)动态翻译成 SQL 请求,将查询结果经多维处理(将关系表达式转换成多维视图)返回用户,如图 3.2 所示。

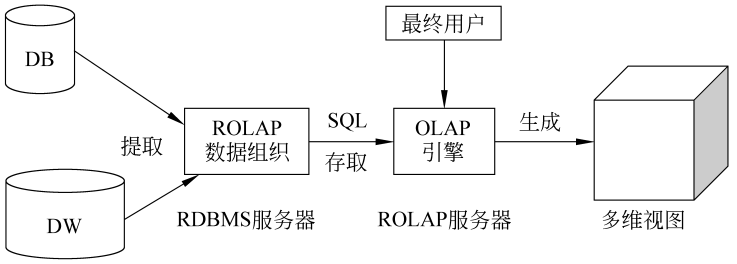

图 3.2　ROLAP 结构

虽然这两种技术都满足了 OLAP 数据处理的一般过程,即数据装入、汇总、建索引和提供使用,但 MOLAP 要比 ROLAP 简明一些。MOLAP 的索引及数据综合可以自动进行;然而 ROLAP 的实现较为复杂,但灵活性较好,用户可以动态实现统计或计算方式。

下面详细深入分析 MOLAP 与 ROLAP。

1. 数据存取速度

ROLAP 的多维数据是以星形模型等关系数据库(平面形式)存储,并不直接体现“超立方体”形式。在接收客户 OLAP 请求时,ROLAP 服务器需要将 SQL 语句转化为多维存储语句,并利用连接运算临时拼合出多维数据立方体。因此,ROLAP 的响应时间较长。

目前,关系数据库已经对 OLAP 做了很多优化,包括并行存储、并行查询、并行数据管理、基于成本的查询优化、位图索引、SQL 的 OLAP 扩展等,大大提高了 ROLAP 的速度。

MOLAP 是专为 OLAP 所设计的,能够自动地建立索引,并且有良好的预计算能力,能够使用多维查询语句访问数据立方体,因此 MOLAP 在数据存储速度上性能好,响应速度快。

2. 数据存储的容量

ROLAP 使用的传统关系数据库的存储方法,在存储容量上基本没有限制。但是,需要指出的是,在 ROLAP 中为了提高分析响应速度,常常构造大量的中间表(如综合表),这些中间表带来了大量的冗余数据。

MOLAP 通常采用多平面叠加成立体的方式存放数据(这样访问速度快),由于受操作系统平台中文件大小的限制,当数据量超过操作系统最大文件长度时,需要进行数据分割。随着数量的增大,多维数据库进行的预运算结果将占用大量的空间,此时可能会导致“数据爆炸”的现象。因此,多维数据库的数据量级难以达到太大的字节级。

3. 多维计算的能力

MOLAP 能够支持高性能的决策支持计算,包括复杂的跨维计算、行级的计算,而在

ROLAP中,SQL无法完成部分计算,并且ROLAP无法完成多行的计算和维之间的计算。

最近发展起来的多维数据分析的MDX语言能更有效地进行多维数据分析。

4. 维变化的适应性

MOLAP需要在建立多维数据库前确定各维以及维的层次关系。在多维数据库建立后,如果要增加新的维,则多维数据库通常需要重新建立。新增维数据会剧烈增加。而ROLAP增加一个维,只是增加一张维表并修改事实表,系统中其他维表不需要修改,因此ROLAP对于维表的变更有很好的适应性。

5. 数据变化的适应性

由于MOLAP通过预综合处理来提高速度,当数据频繁变化时,MOLAP需要进行大量的重新计算,甚至重新建立索引乃至重构多维数据库。在ROLAP中,预综合处理通常由设计者根据需求制定,因此灵活性较好,对于数据变化的适应性强。

6. 软硬件平台的适应性

关系数据库已经在众多的软硬件平台上成功地运行,即ROLAP对软硬件平台的适应性很好,而MOLAP相对较差。

7. 元数据管理

元数据是OLAP和数据仓库的核心数据,OLAP的元数据包括层次关系、计算转化信息、报表中的数据项描述、安全存取控制、数据更新、数据源和预计算综合表等,目前在元数据的管理上,MOLAP和ROLAP都没有成形的标准,MOLAP产品将元数据作为其内在数据,而ROLAP产品将元数据作为应用开发的一部分,由设计者来定义和处理。

MOLAP和ROLAP在技术上各有优缺点。MOLAP以多维数据库为核心,在数据存储和综合上有明显的优势,但它不适应太大的数据存储,特别是对有大量稀疏数据的存储将会浪费大量的存储空间。ROLAP以RDBMS为基础,利用成熟的技术为用户的使用和管理带来方便。

MOLAP和ROLAP在数据存储、技术和特征方面的比较如表3.5所示。

表3.5 MOLAP和ROLAP在数据存储、技术和特征的比较

类型	数据存储	技术	特征
MOLAP	详细数据用关系表存储在数据仓库中;各种汇总数据保存在多维数据库中;从数据仓库中询问详细数据,从多维数据库中询问汇总数据	由MOLAP引擎创建;预先建立数据立方体;多维视图存储在陈列中,而不是表格中;可以高速检索矩阵数据;利用稀疏矩阵技术来管理汇总的稀疏数据	询问响应速度快;能轻松适应多维分析;有广泛的下钻和多层次/多视角的查询能力

续表

类型	数 据 存 储	技　　术	特　　征
ROLAP	全部数据以关系表存储在数据仓库中；可获得细节的和综合汇总的数据；有非常大的数据容量；从数据仓库中询问所有的数据	使用复杂 SQL 从数据仓库中获取数据；ROLAP 引擎在分析中创建数据立方体；表示层能够表示多维的视图	在复杂分析功能上有局限性，需要采用优化的 OLAP；下钻较容易，但是跨维下钻比较困难

3.3　多维数据的显示

3.3.1　多维数据的显示方法

多维数据一般采用多维数据库和关系数据库两种方式存储。多维数据的显示只能在平面上展现出来。对于二维数据采用多维数据库形式显示时，如表 3.1 所示。二维数据采用关系数据库形式显示时，如表 3.3 所示。若增加一维——时间维，仍然可以显示出来，如表 3.6 所示。

表 3.6　三维数据的关系数据库显示

产品名	地区	时间	销售量
衣服	北京	1 月	100
衣服	北京	2 月	200
衣服	北京	3 月	300
衣服	上海	1 月	200
衣服	上海	2 月	300
衣服	上海	3 月	400
衣服	广州	1 月	150
衣服	广州	2 月	250
衣服	广州	3 月	300
鞋	北京	1 月	150
鞋	北京	2 月	300
鞋	北京	3 月	350
鞋	上海	1 月	200
鞋	上海	2 月	300
鞋	上海	3 月	400
鞋	广州	1 月	150
鞋	广州	2 月	250
鞋	广州	3 月	300
⋮	⋮	⋮	⋮

用关系数据库可以显示更多维的数据,即用星形模型的事实表形式显示。但是,用事实表显示多维数据时,重复数据很多,也显得很烦琐。

用多维数据库显示时,虽然不能同时显示三维以上数据,由于显示的数据很精炼,因此仍然用多维数据库的方式来显示多维数据。一般在多维数据库中,固定一些维成员,重点显示二维数据。如在表3.6三维数据中,固定地区维是"北京地区"时的二维数据的显示如表3.7所示。

表3.7 北京地区销售情况表

北京地区	1月	2月	3月
衣服	100	200	300
鞋子	150	300	350
⋮	⋮	⋮	⋮

3.3.2 多维类型结构

为了有效地表示多维数据,E. Thomsen引入了多维类型结构(Multidimensional Type Structure, MTS)。有些专家称其为多维域结构(Multidimensional Domain Structure,MDS)。表示方法:每个维用一条线段来表示。维中的每个成员都用线段上的一个单位区间来表示。例如,用3个线段分别表示时间、产品和指标的三维MTS如图3.3所示。

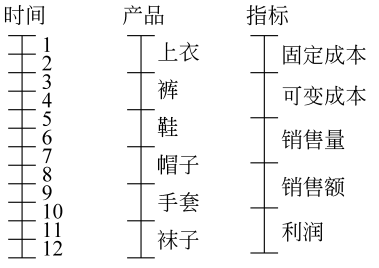

图3.3 三维MTS实例

在图3.3中,指定时间维成员是3月,产品维成员是鞋,指标维成员是销售量,这样它代表了三维数据的一个空间数据点,如图3.4所示。

在MTS中,在原有多维数据中增加一个维是很容易的,例如在图3.3的三维中增加一个商店维,这时需要增加一条线段表示商店维,如图3.5所示(注:对图3.3中的指标维中,把销售量、销售额和利润分别改为直接成本、间接成本和总销售)。

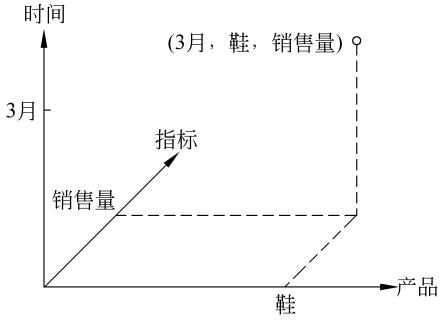

图3.4 多维类型结构中的空间数据点

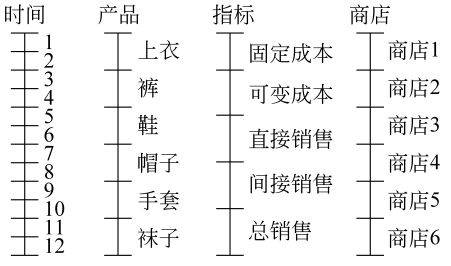

图3.5 四维MTS实例

3.3.3　多维数据的分析视图

在平面的屏幕上显示多维数据，是利用行、列和页面三个显示组来表示的。例如，对上例的四维 MTS 实例，在页面上选定商店维的“商店 3”，在行中选定时间维的“1 月、2 月、3 月”共三个成员，在列中选定产品维中的“上衣、裤、帽子”三个成员，以及指标维中的“固定成本、直接销售”两个成员。该四维数据的显示如图 3.6 所示。

商店3	上衣		裤		帽子	
	直接销售	固定成本	直接销售	固定成本	直接销售	固定成本
1月	450	350	550	450	500	400
2月	380	280	460	360	400	320
3月	400	310	480	410	450	400

图 3.6　四维数据的显示

对于更多维的数据显示，需要选择维及其成员分布在行或列中。在页面上可以选定多个维，但每个维只能显示一个成员。在行或列中一般只选择两个维，每个维都可以多个成员。例如，对六维数据，它的 MTS 如图 3.7 所示。

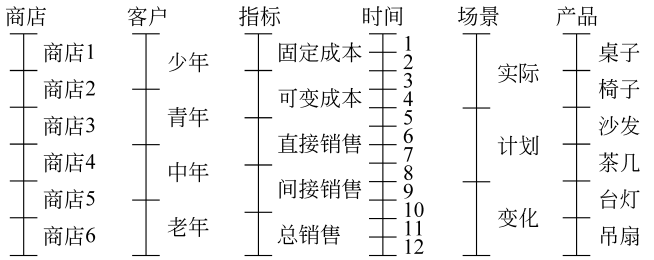

图 3.7　六维 MTS 实例

对以上六维数据，在页面上设定商店维成员是“商店 3”，客户维成员是“老年”。行维含时间维和产品维共两个维，其中时间维成员为“1 月、2 月、3 月”，产品维中成员为“桌子、台灯”。列维含指标维和场景维共两个维，其中指标维成员为“直接销售、间接销售、总销售”，场景维成员为“实际、计划”。具体六维数据的显示如图 3.8 所示。

商店3，老年		直接销售		间接销售		总销售	
		实际	计划	实际	计划	实际	计划
1月	桌子	250	300	125	150	375	450
	台灯	265	320	133	160	400	480
2月	桌子	333	400	167	200	500	600
	台灯	283	340	142	170	425	510
3月	桌子	350	420	175	210	525	630
	台灯	250	300	125	150	375	450

图 3.8　六维数据的显示

由于整个屏幕的空间是有限的，将维嵌套在行或列中相对于放在页维中会占据更多

的屏幕空间。用于显示维的空间越多,用于显示数据的空间就会越少。随着显示数据空间的减少,为了查看同样的数据,就需要做更多的卷屏操作。卷屏操作的增加也加大了理解正在寻找的数据的难度。一些经验规则如下。

(1) 将维尽量放在页中,除非确定需要同时看到一个维的多个成员。让屏幕上的信息尽量相关。

(2) 当维嵌套在行或列中时,考虑到垂直空间比水平空间更有用,所以将维嵌套在列中比嵌套在行中要好。一个经典的显示方法就是在行上有一个维,而在列上嵌套一～三个维,而其他的维则放在页中,如图 3.6 所示。

(3) 在决定数据的屏幕显示方式之前,应该首先弄清楚需要查找和分析比较的内容。例如,如果需要比较某个产品和某类客户在商品和时间上的实际成本情况,就可以将产品和客户放在页维中,而在屏幕上则可以按商店和时间来显示实际成本,如图 3.9 所示。

商店	时间			
	1月	2月	3月	4月
商店1	125	170	157	114
商店2	200	195	129	157
商店3	136	158	132	144

图 3.9　按照商店和时间比较成本的数据组织

页维:

产品维成员“鞋”,指标维成员“成本”,场景维成员“实际”,客户维成员“青年”。

3.4　OALP 的多维数据分析

3.4.1　多维数据分析的基本操作

OLAP 的目的是为管理决策人员通过一种灵活的多维数据分析手段,提供辅助决策信息。基本的多维数据分析操作包括切片、切块、旋转、钻取等。通常把在多维数据分析中加入数据分析模型和商业分析模型称为广义 OLAP。

随着 OLAP 的深入发展,出现了多维数据聚集计算的数据立方体和多维数据分析的 MDX 语言。

1. 切片

选定多维数组的一个二维子集的操作称为切片(Slice),即选定多维数组(维 1,维 2,…,维 n,变量)中的两个维:如维 i 和维 j,在这两个维上取某一区间或任意维成员,而将其余的维都取定一个维成员,则得到的就是多维数组在维 i 和维 j 上的一个二维子集,称这个二维子集为多维数组在维 i 和维 j 上的一个切片,表示为(维 i,维 j,变量)。

切片就是在某两个维上取一定区间的维成员或全部维成员,而在其余的维上选定一个维成员的操作。这里可以得出两点共识:

维是观察数据的角度,那么切片的作用或结果就是舍弃一些观察角度,使人们能在两

个维上集中观察数据。因为人的空间想象能力毕竟有限，一般很难想象四维以上的空间结构。所以对于维数较多的多维数据空间，数据切片是十分有意义的。

图 3.10 为一个按产品维、地区维和时间维组织起来的产品销售数据，用三维数组表示为(地区，时间，产品，销售额)。如果在地区维上选定一个维成员(设为“上海”)，就得到了在地区维上的一个切片(关于“时间”和“产品”的切片)；在产品维上选定一个维成员(设为“电视机”)，就得到了在产品维上的一个切片(关于“时间”和“地区”的切片)。显然，切片的数目取决于每个维上维成员的个数。

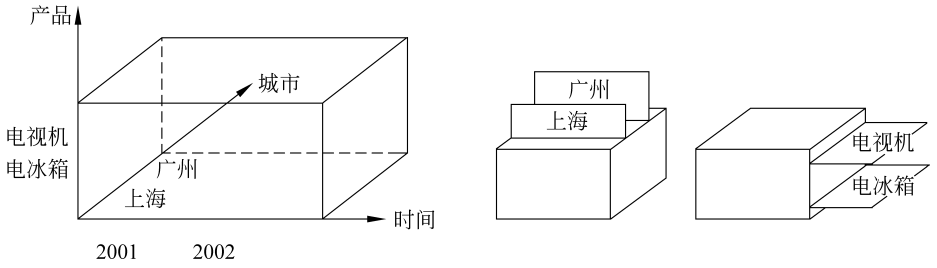

图 3.10　三维数据切片

2. 切块

切块(Dice)有如下两种情况。

(1) 在多维数组的某个维上选定某一区间的维成员的操作。切块可以看成是在切片的基础上确定某个维成员的区间得到的片段，即由多个切片叠合起来的。对于时间维的切片(时间取一个确定值)，如果将时间维上的取值设定为一个区间(例如，取“2001—2005年”)，就得到一个数据切块，它可以看成由 2001—2005 年 5 个切片叠合而成的。

(2) 选定多维数组的一个三维子集的操作。在多维数组(维 1，维 2，…，维 n，变量)中选定三个维，维 i、维 j、维 k，在这三个维上分别取一个区间，或任意维成员，而其他维都取定一个维成员。例如，在三维数组(地区、时间、产品、销售额)中，地区维取“上海、广州”两个维成员，产品维取“电视机、电冰箱”两个维成员，时间维取“2003—2005”(三个维成员)组成三维立方体，如图 3.11 所示。

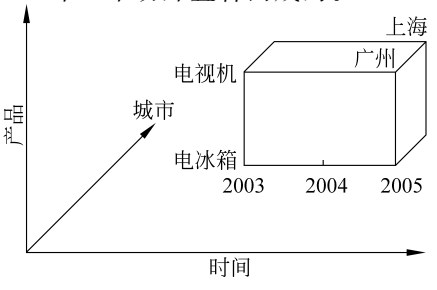

图 3.11　三维数据切块

3. 钻取

钻取(Drill)分为下钻(Drill Down)和上钻(Drill Up)操作。下钻是使用户在多层数据中能通过导航信息而获得更多的细节性数据，而上钻获取概括性的数据。例如，2009 年各部门销售数据如表 3.8 所示。

表 3.8 部门销售数据

部门	销售	部门	销售
部门 1	900	部门 3	800
部门 2	600		

在时间维进行下钻操作,获得新表 3.9。

表 3.9 部门销售下钻数据

	2009 年			
部门	一季度	二季度	三季度	四季度
部门 1	200	200	350	150
部门 2	250	50	150	150
部门 3	200	150	180	270

相反的操作为上钻。钻取的深度与维所划分的层次相对应。

4. 旋转

通过旋转(Pivot)可以得到不同视角的数据。旋转操作相当于平面数据将坐标轴旋转。例如,旋转可能包含了交换行和列,或是把某个行维移到列维中去,或是把页面显示中的一个维和页面外的维进行交换(令其成为新的行或列中的一个),如图 3.12 所示。

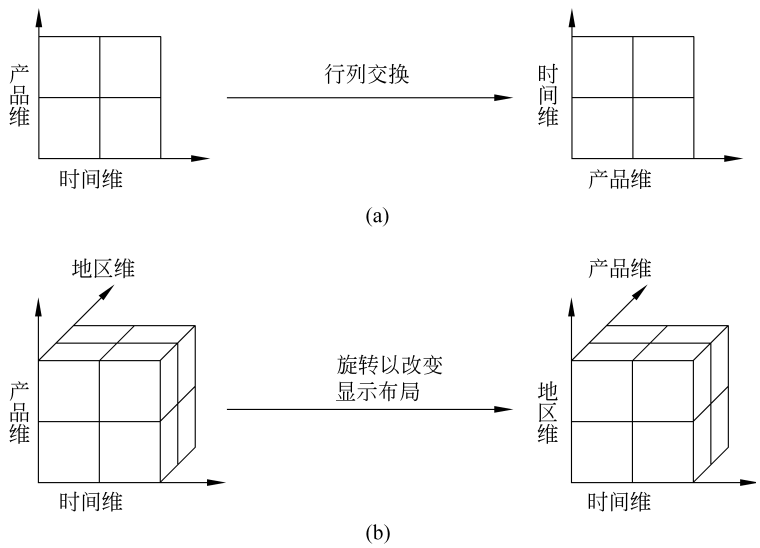

图 3.12 旋转操作

图 3.12(a)是把一个横向为时间、纵向为产品的报表旋转成为横向为产品、纵向为时间的报表。

图 3.12(b)是把一个横向为时间、纵向为产品的报表旋转成一个横向仍为时间而纵向为地区的报表。

3.4.2　多维数据分析实例

1. 切片

为了对广东省全省营业税和个人所得税在 2006、2007 两年的纳税情况进行全面了解，需要对全省税收数据按城市进行切片显示，部分城市数据如表 3.10 所示。

表 3.10　广东省各市营业税和个人所得税表　　单位：亿元

城市	税　收			
	2006 年营业税	2006 年个人所得税	2007 年营业税	2007 年个人所得税
广州市	199.1	96.4	231.9	122
东莞市	53.4	25.4	70.3	31.6
珠海市	23.9	9.1	34.9	13.9
佛山市	55.7	29.3	72.5	34.4

由表 3.10 中数据可知，广州市营业税增加 32.8 亿元，增长率为 16.5%；广州市个人所得税增加 25.6 亿元，增长率为 26.6%；东莞市营业税增加 16.9 亿元，增长率为 31.65%；东莞市个人所得税增加 6.2 亿元，增长率为 24.4%。

对营业税而言，增长量最大的城市是广州市，增加速度较快的城市是东莞市(31.65%)。

2. 下钻

为了更深入分析东莞市的各行业的营业税情况，需要对东莞市营业税数据下钻分析。2006、2007 两年部分行业的纳税情况如表 3.11 所示。

表 3.11　东莞市各行业的营业税表　　单位：百万元

行　业	税　收	
	2006 年营业税	2007 年营业税
农、林、牧、渔业	15.6	10.1
房地产业	1204	1510.5
制造业	85.5	112.8
餐饮业	327.9	363.8
金融业	475.7	698.1
采矿业	0.28	0.26

由表 3.11 中数据可知，东莞市农、林、牧、渔业 2007 年下降了 5.5 百万元，下降率为 35.2%。采矿业下降 0.02 百万元，下降率为 7.1%。房地产业增加 306.5 百万元，增长率

为25%。金融业增加222.4百万元,增长率为46.8%。对这4种行业营业税增减率有更直观的表示,用直方图表示,如图3.13所示。

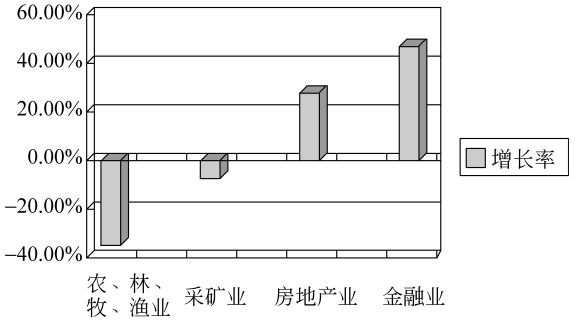

图3.13 东莞市4个行业营业税增减率的直方图

3. 数据分析

1) 宏观分析

从表3.10中的数据可以宏观地看出,东莞市的营业税增长很突出,在广东省各市中名列前位。

2) 深入分析

根据表3.11中的行业数据进行深入分析时发现,东莞市的农、林、牧、渔业下降明显,采矿业也下降,而房地产业增长明显,金融业增长突出。通过调查得出,原因是随着经济的发展,东莞市的外来合资企业越来越多,本地农民很多把地卖了或者租出去建厂房然后收租金,造成农、林、牧、渔业营业税下降,东莞市近年逐步实现产业转移,由农业更多地转向制造业和加工业。

从总体看,东莞市的房地产业和金融业税收的上升,掩盖了农、林、牧、渔业税收的下降。对领导来说,需要做出正确的决策:要继续支持还是需要调整。

3.4.3 广义OLAP功能

OLAP的切片、切块、旋转与钻取等基本操作是最基本的展示数据、获取数据信息的手段。从广义上讲,任何能够有助于辅助用户理解数据的技术或操作都可以作为OLAP功能,这些有别于基本OLAP的功能称为广义OLAP功能。

1. 基本代理操作

"代理"是一些智能性代理,当系统处于某种特殊状态时提醒分析员。

(1) 示警报告。定义一些条件,一旦条件满足,系统会提醒分析员去做分析。例如,每日报告完成或月订货完成等通知分析员做分析。

(2) 时间报告。按日历和时钟提醒分析员。

(3) 异常报告。当超出边界条件时提醒分析员。例如,销售情况已超出预定义阈值的上限或下限时提醒分析员。

2. 数据分析模型

E. F. Codd 认为，以前的数据分析主要集中在静态数据值的相互比较上。有了 OLAP 后，可以进行动态数据分析，需要建立企业数据模型。E. F. Codd 将数据分析模型分为 4 类：绝对模型（Categorical Model）、解释模型（Exegetical Model）、思考模型（Contemplative Model）和公式模型（Formulaic Model）。

（1）绝对模型。它属于静态数据分析，通过比较历史数据值或行为来描述过去发生的事实。该模型查询比较简单，综合路径是预先定义好的，用户交互少。

（2）解释模型。它也属于静态数据分析，分析人员利用系统已有的多层次的综合路径层层细化（进行下钻操作），找出事实发生的原因。

（3）思考模型。它属于动态数据分析，旨在说明在一维或多维上引入一组具体变量或参数后将会发生什么。分析人员在引入确定的变量或公式关系时，须创建大量的综合路径。

（4）公式模型。它的动态数据分析能力更强，该模型表示在多个维上，需要引入哪些变量或参数，以及引入后所产生的结果。

下面通过一个实例进行说明。

一家百货公司在建立了自己的数据仓库后，希望构造一个 OLAP 系统辅助决策。决策者最关心的一个问题是如何最大限度地扩大商品的销售量，因而他希望能尽可能地找出与销售量相关的因素，从而采取相应的促销手段。但是他能获得多大的帮助需要取决于采用何种分析模型。

（1）绝对模型只能对历史数据进行比较，并且利用回归分析等一些分析方法得出趋势信息。它能回答诸如“某种商品今年的销售情况与以往相比有怎样的变化？今后的趋势怎样？”此类问题。

（2）解释模型能够在当前多维视图的基础上找出事件发生的原因。例如，该公司按时间、地区、商品及销售渠道建立了多维数据库，假设今年销售量下降，那么解释模型应当能找出原因，即销售量下降与时间、地区、商品及销售渠道四者中的何种因素有关。

（3）思考模型可以在决策者的参与下，找出关键变量。例如，该公司决策者为了了解某商品的销售量是否与顾客的年龄有关，引入了新变量——年龄，即在当前的多维视图上增加了顾客的年龄维。解释模型就能分析出年龄的引入是否必要，即商品销售与顾客年龄有关或无关。

（4）公式模型自动完成上述各种变量的引入和分析，从而最终找出与销量有关的全部因素，并给出了引入各变量后的结果。

可以看出，这 4 种模型一个比一个深入，从描述基本事实到寻找原因，从代入变量值进行预测到寻找关键变量。

E. F. Codd 认为 OLAP 是因企业动态分析而产生的，其功能是创建、操作、激活及综合来自解释模型、思考模型及公式模型中的信息。它可以识别变量间新的或不可预测的关联，通过创建大量的维（综合路径）及指出维间计算条件、表达式来处理大量数据，获得辅助决策信息。

3. 商业分析模型

利用数据仓库中的数据进行商业分析需要建立一系列模型,用于提高决策支持能力。具体的商业分析模型如下。

1) 分销渠道的分析模型

通过客户、渠道、产品或服务三者之间的关系,了解客户的购买行为、客户和渠道对业务收入的贡献、哪些客户比较喜欢由什么渠道在何时和银行打交道、目前的分销渠道的服务能力、需要增加哪些分销渠道才能达到预期的服务水平。

为此,银行需要建立客户购买倾向模型和渠道喜好模型等。

2) 客户利润贡献度模型

通过该模型能了解每位客户对银行的总利润贡献度,银行可以依客户的利润贡献度安排合适的分销渠道提供服务和销售,知道哪些利润高的客户需要留住,采用什么方法留住客户,交叉销售改善客户的利润贡献度,哪些客户应该争取,完成个性化服务。另外,银行可以模拟和预测新产品对银行的利润贡献度,或者新政策对银行将产生什么样的财务影响,或者客户流失或留住对银行整体利润的影响。

3) 客户关系(信用)优化模型

银行对客户的每笔交易中,知道客户需要什么产品或服务,例如,定期存款是希望退休养老使用、申请信用卡需要现金消费、询问放贷利息需要住房贷款等,这些都是银行提供产品或服务最好的时机。银行需要将账号每天发生的交易明细,以实时或定时方式加载到数据仓库中,关注客户行为的变化。当发生上述变化时,通过模型计算,主动地与客户沟通并进行交叉销售,达到留住客户和增加利润的目标。

4) 风险评估模型

模拟风险和利润间的关系,建立风险评估的数学模型,在满足高利润、低风险客户需求的前提下,达到银行收益的极大化。

银行通过以上模型实现以客户为中心的数据仓库决策支持系统,才能真正实现个性化服务,提高银行竞争优势。

3.4.4 数据立方体

1. 概述

1996年,Jim Gray等首次提出了数据立方体(Data Cube)的概念,数据立方体是实现多维数据查询与分析的一种重要手段。实质上,数据立方体就是数据仓库的结构图(见图2.1)中的综合数据层(轻度和高度)。从此,基于数据立方体的生成方法一直是OLAP和数据仓库领域研究者所关注的热点问题。

多维数据集的属性分为维属性和度量属性。维属性是观察数据对象的角度,而度量属性则反映数据对象的特征。对于多维数据分析而言,本质上是沿着不同维进行数据获取的过程。在数据立方体中,不同维组合构成了不同的子立方体,不同维值的组合及其对

应的度量值构成相应的对于不同的查询和分析。因此，数据立方体的构建和维护等计算方法成为了多维数据分析研究的关键问题。

OLAP 和数据仓库通常预先计算好不同细节层次和不同维属性集合上的聚集，并把聚集的结果存储到物理磁盘上(称为物化)。把所有可能的聚集(即全聚集)都计算出来，可以得到最快的系统查询响应时间，即使不管计算聚集所花费的 CPU 处理时间，只是随着维数的增加，这样做就有可能导致数据聚集的种类剧烈增加。

数据立方体是在所有可能组合的维上进行分组聚集运算(group by 操作)的总和，聚集函数有 sum()、count()、average() 等。数据立方体中的每个元组(立方体的度量属性)称为该立方体上的格(Cell)，每个格在 n 个维属性上有相应的值，其中，在未参与 group by 操作的维属性上具有 All 值(用 * 表示)，而在参与 group by 操作的维属性具有非 All 值。

例如，对于一个具有三个维属性 A、B、C 和一个度量属性 M 的数据集 $R(A,B,C,M)$，其对应的数据立方体是在维属性集{ }，{A}，{B}，{C}，{AB}，{AC}，{BC}，{ABC}上分别对度量属性进行聚集操作后的并集。其中，{}表示进行聚集运算{ * , * , * ，聚集函数 (M)}，{A}表示进行聚集运算{A, * , * ，聚集函数(M)}等。

这些聚集运算与操作结果是数据仓库中的一种高度综合级数据，实质上是进行了数据的浓缩(压缩)，也可称为泛化。最终所获得的这些数据立方体可用于决策支持、知识发现或其他许多应用。

例如，对表 3.12 所示的超市的基本数据集 POS(product，type，counter，price)，前三个属性分别代表(产品名、类型、柜台)维属性，对度量属性 price 进行取平均值的聚集运算，则通过 Cube 操作可以得到一个具有三个维属性和一个度量属性的数据立方体 Dpos，如表 3.13 所示。同时，也可以用三维方式来体现立方体的特征(省略)。

表 3.12　基本数据集 POS

product	type	counter	price
KONKA	TV SET	01	1000
TCL	TV SET	01	1500
NOKIA	PHONE	01	2000

表 3.13　全聚集的数据立方体 Dpos

product	type	counter	M(AVG(price))
*	*	*	1500
KONKA	*	*	1000
TCL	*	*	1500
NOKIA	*	*	2000
*	TV SET	*	1250