ISSN 1000-9825 Journal of Software 软件学报 2000,11(7):908~917

一种数据仓库的多维数据模型*

李建中 高宏

(哈尔滨工业大学计算机科学与工程系 哈尔滨 150001)

E-mail: jzl@hlju.edu.cn

摘要 数据模型是数据仓库研究的核心问题之一.很多研究表明,传统数据模型(如实体联系模型和关系模型)不能有效地表示数据仓库的数据结构和语义,也难以有效地支持联机分析处理(on-line analysis processing,简称OLAP).最近,人们提出了几种多维数据模型.但是,这些多维数据模型在表示数据仓库的复杂数据结构和语义以及OLAP操作方面仍显不足.该文以偏序和映射为基础,提出了一种新的多维数据模型.该数据模型能够充分表达数据仓库的复杂数据结构和语义,并提供一个以OLAP操作为核心的操作代数,支持层次结构间的复杂聚集操作序列,能够有效地支持OLAP应用.该数据模型支持聚集函数约束的概念,提供了表示层次结构间聚集函数约束的机制.

关键词 数据仓库,数据模型,多维数据模型,联机分析处理(OLAP).

中图法分类号 TP311

数据模型是数据仓库研究的核心问题.虽然最近人们在数据仓库方面开展了大量的研究工作,但主要还集中在实体化视图设计、存储和维护、OLAP(on-line analysis processing)操作的有效算法等几个方面,数据模型的研究还很不够,多数研究工作都以关系数据模型和关系数据库为基础.研究表明,关系数据模型不能有效地表示数据仓库的数据结构和语义,也难以有效地支持OLAP的应用[1].Codd提出的OLAP标准指出,OLAP操作具有多维性特征[2].因此,多维数据模型引起了人们的注意.最近几年,人们提出了几种多维数据模型[3~10].这些数据模型把数据集合视为多维空间中的点集,把数据集合的属性分为维和度量两类.维属性用来描述度量属性,是多维空间的维度.度量属性的值用来进行分析处理,是多维空间中的点.这些数据模型虽然具有多维特点,但是它们有的不能表示维层次结构,有的只能表达简单的维层次结构(即只有一条路径的层次结构),最好的模型也只能表示满足具有代数格特征的维层次结构.在实际应用领域中,很多数据集合的维具有复杂层次结构,并不具有代数格的特征.下面,我们讨论一个具有复杂维层次结构的数据集合.这个数据集合是我们为黑龙江省移动电话局建立数据仓库系统时遇到的具有复杂维层次结构的诸多数据集合中的一个简单实例.

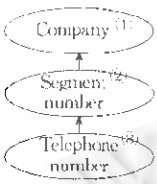


Fig. 1
图1

(1)公司,
(2)万段号,
(3)电话号码.

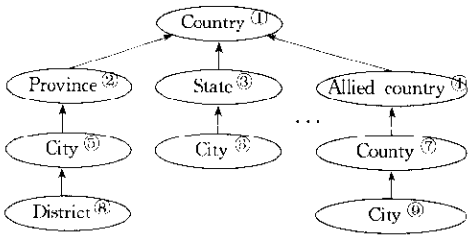
在黑龙江省移动电话局的数据仓库中有一个支持客户话务特征分析的数据集合.这个数据集合包括3个维:地区、时间、呼叫源和3个度量属性:通话次数、通话时长、话务费.呼叫源维由属性集合{公司,万段号,电话号码}构成,其结构如图1所示.地区维由属性集合{国家,省,地市,县区,州,县,市,加盟国,邮或大区,区或市}构成,其结构如图2所示,不同分枝表示不同国家的地区层次,图中仅列出了中国、美国和英国的地区层次.时间维由属性集合{年,季,月,周}构成,其结构如图3所示.由于每天客户通话信息高达10GB,目前的联机存储设备难以保存大量日数据,所以时间维无法包括最低层次"日"的数据.显然,地区维和时间维的层次结构不是

* 本文研究得到国家自然科学基金(No. 69873014)、国家863高科技项目基金(No. 863-511-9846-004)和国家973高科技项目基金(No. G1999032704)资助.作者李建中,1951年生,教授,博士生导师,主要研究领域为数据库系统,并行计算.高宏,女,1966年生,博士生,主要研究领域为数据库系统.

本文通讯联系人:李建中,哈尔滨 150001,哈尔滨工业大学计算机科学与工程系

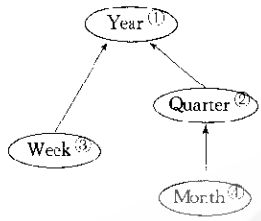
本文1999-11-24收到原稿,2000-04-14收到修改稿

代数格.现有的多维数据模型不能表示这个数据集合.



①国家,②省,③州,④加盟国,⑤地市,⑥市,⑦郡或大区,⑧县区,⑨区或市.

Fig. 2
图2



①年,②季,③周,④月.

Fig. 3
图3

从上面的讨论可以看出,现有的多维数据模型在表达复杂数据结构方面还不完善,需要加以改进.除此之外,现有的多维数据模型所支持的OLAP操作也各有不足,没有一个模型能够支持完整的OLAP操作集合.

针对现有的多维数据模型问题,本文提出了一种新的多维数据模型.与现有的多维数据模型相比,该数据模型具有如下特点:(1) 以偏序关系和映射概念为基础,提供了很强的复杂维层次结构表达能力,能够有效地表达数据仓库的各种复杂层次结构和语义;(2) 包括一个以完整的OLAP操作集合为核心的操作代数,可以有效地支持OLAP应用;(3) 引进了聚集函数约束概念,提供了表达维层次结构间聚集函数约束的机制;(4) 允许在多维数据集合的任一维的同一个层次链上使用不同的聚集函数执行Roll-up和Drill-down操作,支持同一层次结构上的复杂聚集操作序列.我们提出的这个多维数据模型已经用于一个并行数据仓库原型系统.实践证明,本文提出的多维数据模型是一个具有很强表达能力的、切实可行的多维数据模型.

1 数据模型

定义 1.1. 设 S 是一个有限集合,$\rho\subseteq 2^S$.如果下列条件成立,则称 ρ 为 S 的一个划分:

(1) $S=\bigcup_{P\in\rho}P$;

(2) 对于任意 $P_i,P_j\in\rho$,若 $i\neq j$,则 $P_i\cap P_j$ 为空集合.

定义 1.2. 设 $\alpha=\{S_1,S_2,\ldots,S_n\}$ 是一个有限集族,$S_i(1\leqslant i\leqslant n)$ 是任意集合,$\leqq$ 是如下定义的二元关系:对于 α 中任意集合 S_i 和 S_j,如果存在一个函数 $F:\rho\to S_j$(ρ 是 S_i 的划分),则称 S_i 和 S_j 满足 $\leqq$,记作 $S_i\leqq S_j$,F 称为聚集函数.如果 $\leqq$ 满足下列条件,则称 $\leqq$ 为 α 上的聚集关系:对于 α 中任意集合 S_i 和 S_j,如果 $S_i\leqq S_j$,$S_j\leqq S_i$,则 $S_i=S_j$.

引理 1. 集族 α 上的聚集关系是一个偏序.

证明:略.

以下我们称 $\leqq$ 为聚集偏序,称 $(\alpha\cup ALL,\leqq)$ 为聚集偏序集族,其中 $ALL=\{all\}$,而且 $\forall S\in\alpha,S\leqq ALL$.在数据仓库中,聚集偏序规定了数据集合的抽象级别.ALL 在聚集偏序集族中具有最高抽象级别.在下边的讨论中,我们假设所有集族都包含集合 ALL.

定义 1.3. 设 α 是一个集族,$S\in\alpha$.若对任意 $R\in\alpha(R\neq S)$,$S\leqq R$ 和 $R\leqq S$ 都不成立,则称 S 是 α 中的奇异点.

定义 1.4. 设 $\alpha=\{S_1,S_2,\ldots,S_n\}$ 是一个有限集族,$S_i(1\leqslant i\leqslant n)$ 是任意集合.如果 α 上存在一个聚集偏序 $\leqq$,而且 α 中无奇异点,则称 $(\alpha,\leqq)$ 为非奇异聚集偏序集族.

定义 1.5. 设 $(\alpha,\leqq)$ 是一个非奇异聚集偏序集族,$\theta=\{(S_i,S_j,\varphi_{ij})\mid S_i,S_j\in\alpha,S_i\leqq S_j,\varphi_{ij}=\{\Psi_1,\Psi_2,\ldots,\Psi_m\},\Psi_k$ 是聚集函数集合$\}$.$(\alpha,\leqq,\theta)$ 称为约束非奇异聚集偏序集族,θ 称为 $(\alpha,\leqq)$ 的约束.

给出多维数据集合模式定义以后,我们再来讨论约束 θ 的实际意义.

定义 1.6. 一个 n 维数据集合模式是一个三元组 $R=(D,M,Dstr)$,其中

(1) $D=\{d_1,d_2,\ldots,d_n\}$,称为维集合,d_i 称为维;

(2) $M=\{M_1,M_2,\ldots,M_k\}$,称为度量属性集合;

(3) $Dstr=\{(\alpha_1,\leqq,\theta_1),(\alpha_2,\leqq,\theta_2),\ldots,(\alpha_n,\leqq,\theta_n)\}$,称为维结构集合,$(\alpha_1,\leqq,\theta_1),(\alpha_2,\leqq,\theta_2),\ldots,(\alpha_n,\leqq,\theta_n)$是 n 个约束非奇异聚集偏序集族,$(\alpha_i,\leqq,\theta_i)$定义了维 d_i 的层次结构和聚集约束,α_i 中的每个集合称为维 d_i 的一个维层次属性;

(4) 度量属性集合 M 函数依赖于维集合 D,即 D 和 M 之间存在函数 $F:DOM(d_1)\times\ldots\times DOM(d_n)\rightarrow DOM(M_1)\times\ldots\times DOM(M_k)$,其中 $DOM(d_i)$是维 d_i 的值域(详见定义 1.9),$DOM(M_j)$是度量属性的值域.

现在,我们来说明定义 1.5 和定义 1.6 中约束的实际意义.如果在定义 1.6 的数据集合模式 R 中,$S_p,S_q\in\alpha_i$,$S_p\leqq S_q$,而且 F 是把 S_p 的划分 ρ 中的每个子集合映射为 S_q 的某个元素的聚集函数,我们可以根据 F 把 S_p 的元素分组,并将每组元素$\{s_1,\ldots,s_j\}$映射到 S_q 的一个元素 $F(\{s_1,\ldots,s_j\})$上,同时使用 k 个聚集函数$\{f_1,f_2,\ldots,f_k\}$把 k 个度量属性的 k 个值集合中与$\{s_1,\ldots,s_j\}$对应的 k 个子集合聚集为 k 个数值(f_j 用于第 j 个度量属性),从而取消维层次属性 S_p,使 R 具有更高层次的抽象语义.我们称这个过程为维层次聚集操作(严格定义在后面).维层次聚集操作是非常重要的 OLAP 操作.对于 $1\leqslant i\leqslant n$,约束 θ_i 的意义如下:如果$(S_p,S_q,\varphi_{pq})\in\theta_i$,则在进行从 S_p 到 S_q 的维层次聚集操作时,对 k 个度量属性使用的 k 个聚集函数的集合$\{f_1,f_2,\ldots,f_k\}$必须属于 φ_{pq},即 θ_i 规定了 α_i 中任何两个满足聚集关系的集合之间的聚集操作所允许使用的聚集函数的集合.我们称 φ_{pq}为 $S_p\leqq S_q$ 的聚集约束,$\{\theta_1,\theta_2,\ldots,\theta_n\}$为多维数据集合 R 的聚集约束.

定义 1.7. 设 $R=(D,M,Dstr)$是一个多维数据集合,其中 $D=\{d_1,d_2,\ldots,d_n\}$,$M=\{M_1,M_2,\ldots,M_k\}$,$Dstr=\{(\alpha_1,\leqq,\theta_1),(\alpha_2,\leqq,\theta_2),\ldots,(\alpha_n,\leqq,\theta_n)\}$.设 $S_{m+1}=ALL$.对于 $1\leqslant i\leqslant n$,如果下列条件成立,则$\{S_1,\ldots,S_m,S_{m+1}\}\subseteq\alpha_i$ 称为维 d_i 的一个层次链(记作 $S_1\leqq_{\varphi_1}S_2\leqq_{\varphi_2}\ldots\leqq_{\varphi_{m-1}}S_m\leqq_{\varphi_m}S_{m+1}$):

(1) $S_1\leqq S_2\leqq\ldots\leqq S_m\leqq S_{m+1}$,而且对于 $1\leqslant j\leqslant m$,φ_j 是 $S_j\leqq S_{j+1}$的聚集约束,即$(S_j,S_{j+1},\varphi_j)\in\theta_i$.

(2) α_i 中不存在满足条件(1)的子集合 $\beta\supset\{S_1,\ldots,S_m,S_{m+1}\}$.

定义 1.8. 一个数据仓库模式是一组多维数据集合模式,记作$\langle(D_1,M_1,Dstr_1),(D_2,M_2,Dstr_2),\ldots,(D_m,M_m,Dstr_m)\rangle$.

显然,当 M 为空集合而且$(\alpha_i,\leqq)$中的 α_i 是不具有层次结构的简单集合时,多维数据集合模式即为关系数据模型中的关系模式.当一个数据仓库模式中的所有多维数据集合模式都是关系模式时,这个数据仓库模式则为关系数据库模式.不难看出,我们的多维数据模型包含了关系数据模型(注意,后面的数据操作定义也包括了关系代数操作).下边我们来定义多维数据集合实例的概念.

定义 1.9. 设 $DS=(D,M,Dstr)$是一个数据集合模式,$D=\{d_1,d_2,\ldots,d_n\}$,$M=\{M_1,M_2,\ldots,M_k\}$,$Dstr=\{(\alpha_1,\leqq,\theta_1),(\alpha_2,\leqq,\theta_2),\ldots,(\alpha_n,\leqq,\theta_n)\}$,对于 $1\leqslant i\leqslant n$,$\alpha_i=\{S_{i1},S_{i2},\ldots,S_{im}\}$中具有 h_i 个层次链.对于 α_i 的第 j 个层次链 $S_{ij1}\leqq S_{ij2}\leqq\ldots\leqq S_{ijt_{ij}}$,令 $L_{ij1}=S_{ijt_{ij}}$,$L_{ij2}=S_{ijt_{ij}}\times S_{ijt_{ij}-1}$,$\ldots$,$L_{ijt_{ij}-1}=S_{ijt_{ij}}\times S_{ijt_{ij}-1}\times\ldots\times S_{ij2}$,$L_{ijt_{ij}}=S_{ijt_{ij}}\times S_{ijt_{ij}-1}\times\ldots\times S_{ij1}$.我们称 $DOM(d_i)=\bigcup_{1\leqslant j\leqslant h_i,1\leqslant p\leqslant t_{ij}}L_{ijp}$为维 d_i 的值域.$DOM(d_i)$中的每个元素称为维 d_i 的一个值.

度量属性 M_i 的值域即为 M_i 的取值范围.M_i 的值域对于在 M_i 上的所有合法的聚集函数都是封闭的,即如果 $x\in M_i$,F 是 M_i 上的一个聚集函数,则 $F(x)\in M_i$.为了叙述方便,以下我们使用 $DOM(D)$表示 $DOM(d_1)\times DOM(d_2)\times\ldots\times DOM(d_n)$,使用 $DOM(M)$表示 $DOM(M_1)\times DOM(M_2)\times\ldots\times DOM(M_k)$.

定义 1.10. 设 $DS=(D,M,Dstr)$是一个数据集合模式,$D=\{d_1,d_2,\ldots,d_n\}$,$M=\{M_1,M_2,\ldots,M_k\}$,$Dstr=\{(\alpha_1,\leqq,\theta_1),(\alpha_2,\leqq,\theta_2),\ldots,(\alpha_n,\leqq,\theta_n)\}$,对于 $1\leqslant i\leqslant n$,$\alpha_i=\{S_{i1},S_{i2},\ldots,S_{im}\}$中具有 h_i 个层次链.对于 α_i 的第 j 个层次链 $S_{ij1}\leqq S_{ij2}\leqq\ldots\leqq S_{ijt_{ij}}$,令 $L_{ij1}=S_{ijt_{ij}}$,$L_{ij2}=S_{ijt_{ij}}\times S_{ijt_{ij}-1}$,$\ldots$,$L_{ijt_{ij}-1}=S_{ijt_{ij}}\times S_{ijt_{ij}-1}\times\ldots\times S_{ij2}$,$L_{ijt_{ij}}=S_{ijt_{ij}}\times S_{ijt_{ij}-1}\times\ldots\times S_{ij1}$.$DS$ 的一个实例 ds 是一个映射 $F:\lambda\rightarrow DOM(M)$,$\lambda\subseteq DOM(D)$.$ds$ 满足:对于 $1\leqslant i\leqslant n$,$1\leqslant j\leqslant h_i$,$\Pi_{d_i}ds=\bigcup_{1\leqslant j\leqslant h_i}ds_{ij}$,$ds_{ij}$是空集或存在一个 $t_0(1\leqslant t_0\leqslant t_{ij})$,使得 $ds_{ij}\subseteq L_{ijt_0}$,其中 $\Pi_{d_i}ds$ 是 ds 的维 d_i 的值集合.

定义 1.11. 在数据集合模式 $DS=(D,M,Dstr)$的实例 ds 中出现的维 d_i 的值的集合是 $DOM(d_i)$的子集合,称为数据集合 DS 的维 d_i 的当前值域,简记作 $CDOM(d_i)$.在 ds 中出现的度量属性 M_j 的值的集合是$DOM(M_j)$

的子集,称为 M_j 的当前值域,简记作 $CDOM(M_j)$.

以后,我们使用 $CDOM(D)$ 表示 $CDOM(d_1)\times CDOM(d_2)\times\ldots\times CDOM(d_n)$,使用 $CDOM(M)$ 表示 $CDOM(M_1)\times CDOM(M_2)\times\ldots\times CDOM(M_k)$.

数据集合模式 DS 的实例 ds 可以视为一个 $n+k$ 元组的集合.每个元组的前 n 个分量对应于 n 个维的值,第 i 个分量值取自 $DOM(d_i)$;后 k 个分量对应于 k 个度量属性的值,第 j 个分量值取自 $DOM(M_j)$. $(t_1,\ldots,t_n,m_1,\ldots,m_k)$是 ds 的一个元组当且仅当 $F(t_1,\ldots,t_n)=(m_1,\ldots,m_k)$.显然,由于多维数据集合实例是一个映射,$n$ 个维的值函数确定了 k 个度量属性的值.

请注意,我们在这里定义的数据集合实例只是我们定义的多维数据模型的一种语义,是一个逻辑数据结构,不是物理数据存储结构.在使用这个多维数据模型实现一个数据仓库管理系统时,我们可以使用多种物理数据存储结构和方法来实现和存储任何数据集合实例.为此,我们将另文讨论数据集合实例的实现和存储方法.

定义 1.12. 设 $DW=\{(D_i,M_i,Dstr)\mid 1\leqslant i\leqslant m\}$是一个数据仓库模式.$DW$ 的实例是 $dw=\{ds_i\mid 1\leqslant i\leqslant m,ds_i$ 是 $(D_i,M_i,Dstr)$的实例$\}$.

2 代数操作

本节讨论数据仓库上的数据操作,给出一个多维数据集合族上的操作代数.在下边的讨论中,我们分别用 $Sch(E)$ 和 $Ins(E)$ 表示代数操作表达式 E 所对应的多维数据集合的模式和实例.

2.1 基本代数操作

定义 2.1(数据集合同构). 设 $R=(D_R,M_R,Dstr_R)$ 和 $S=(D_S,M_S,Dstr_S)$是两个数据集合.如果 R 和 S 满足下列条件,则称 R 与 S 同构:

(1) $|D_R|=|D_S|$, $|M_R|=|M_S|$.

(2) 存在一个一一映射 $F_D:Dstr_R\to Dstr_S$. $F_D((\alpha_R,\leqq,\theta_R))=(\alpha_S,\leqq,\theta_S)$当且仅当下列条件成立:

(a) $|\alpha_R|=|\alpha_S|$;

(b) $\forall(S_{R1},S_{R2},\varphi_R)\in\theta_R$ 当且仅当存在一个 $(S_{S1},S_{S2},\varphi_S)\in\theta_S$ 与之对应,$\varphi_R=\varphi_S$,S_{R1} 的元素与 S_{S1} 的元素具有相同数据类型,S_{R2} 的元素与 S_{S2} 的元素具有相同数据类型.

(3) 度量属性之间存在一个一一映射 $F_M:M_R\to M_S$,使得如果 $F_M(m_R)=m_S$,则 $DOM(m_R)$ 的元素与 $DOM(m_S)$ 的元素具有相同的数据类型.

在下边的定义 2.2、定义 2.3 和定义 2.4 中,$R=(D_R,M_R,Dstr_R)$和 $S=(D_S,M_S,Dstr_S)$是两个同构的多维数据集合,$Ins(R)=F_R$,$Ins(S)=F_S$.

定义 2.2(集合交). R 与 S 的交 $R\cap S$ 是一个如下定义的多维数据集合:

(1) $Sch(R\cap S)=(D_{R\cap S},M_{R\cap S},Dstr_{R\cap S})$,$(D_{R\cap S},M_{R\cap S},Dstr_{R\cap S})$与 R 和 S 同构.

(2) $Ins(R\cap S)$是映射 $F_{R\cap S}:CDOM(D_{R\cap S})\to CDOM(M_{R\cap S})$,其中 $CDOM(D_{R\cap S})=\{e\mid\forall e((e\in CDOM(D_R)\cap CDOM(D_S))\wedge(F_R(e)=F_S(e)))\}$,$CDOM(M_{R\cap S})=\{m\mid\exists e((e\in CDOM(D_R)\cap CDOM(D_S))\wedge(F_R(e)=F_S(e)=m))\}$,而且 $\forall e\in CDOM(D_{R\cap S})$,$F_{R\cap S}(e)=F_R(e)=F_S(e)$.

定义 2.3(集合并). R 与 S 的并 $R\cup S$ 是一个如下定义的多维数据集合:

(1) $Sch(R\cup S)=(D_{R\cup S},M_{R\cup S},Dstr_{R\cup S})$,$(D_{R\cup S},M_{R\cup S},Dstr_{R\cup S})$与 R 和 S 同构.

(2) $Ins(R\cup S)$是映射 $F_{R\cup S}:CDOM(D_{R\cup S})\to CDOM(M_{R\cup S})$,其中 $CDOM(D_{R\cup S})=CDOM(D_R)\cup CDOM(D_S)$,$CDOM(M_{R\cup S})=CDOM(M_R)\cup CDOM(M_S)$,而且 $\forall e\in CDOM(D_{R\cup S})$,若 $e\in CDOM(D_R)$,$F_{R\cup S}(e)=F_R(e)$,若 $e\in CDOM(D_S)$,$F_{R\cup S}(e)=F_S(e)$.

定义 2.4(集合差). R 与 S 的差 $R-S$ 是一个如下定义的多维数据集合:

(1) $Sch(R-S)=(D_{R-S},M_{R-S},Dstr_{R-S})$,$(D_{R-S},M_{R-S},Dstr_{R-S})$与 R 同构.

(2) $Ins(R-S)$是映射 $F_{R-S}:CDOM(D_{R-S})\to CDOM(M_{R-S})$,其中 $CDOM(D_{R-S})=\{e\mid\forall e(e\in CDOM(D_R)\wedge(e\notin CDOM(D_S)\vee(e\in CDOM(D_S)\wedge F_R(e)\neq F_S(e))))\}$,$CDOM(M_{R-S})=\{m\mid\exists e((e\in CDOM(D_{R-S}))\wedge$

$(F_R(e)-m))\}$,而且,$\forall e\in CDOM(D_{R-S}), F_{R-S}(e)=F_R(e)$.

定义 2.5(笛卡尔积). 设 $R=(D_R,M_R,Dstr_R)$和 $S=(D_S,M_S,Dstr_S)$是两个多维数据集合,$Ins(R)=F_R$,$Ins(S)=F_S$. R 与 S 的笛卡尔积 $R\times S$ 是一个如下定义的多维数据集合:

(1) $Sch(R\times S)=(D_{R\times S},M_{R\times S},Dstr_{R\times S})$, $D_{R\times S}=D_R\cup D_S$, $M_{R\times S}=M_R\cup M_S$, $Dstr_{R\times S}=Dstr_R\cup Dstr_S$.

(2) $Ins(R\times S)$是映射 $F_{R\times S}:CDOM(D_{R\times S})\to CDOM(M_{R\times S})$,其中 $CDOM(D_{R\times S})=CDOM(D_R)\times CDOM(D_S)$, $CDOM(M_{R\times S})=CDOM(M_R)\times CDOM(M_S)$,而且$\forall e=e_1e_2\in CDOM(D_{R\times S}), e_1\in CDOM(D_R), e_2\in CDOM(D_S)$, $F_{R\times S}(e)=F_R(e_1)\times F_S(e_2)$.

定义 2.6(选择). 设 $R=(D,M,Dstr)$, $Ins(R)=F_R$. R 上的选择操作表示为 $Select(R,P)$,其中 P 是定义在 R 的维层次属性和度量属性上的选择条件. $Select(R,P)$是一个如下定义的多维数据集合:

(1) $Sch(Select(R,P))=(D_{sel},M_{sel},Dstr_{sel})$, $(D_{sel},M_{sel},Dstr_{sel})$与 R 同构.

(2) $Ins(Select(R,P))$是映射 $F_{Select}:CDOM(D_{sel})\to CDOM(M_{sel})$,其中 $CDOM(D_{sel})=\{e\mid\forall e(e\in CDOM(D)\wedge P(e\times F_R(e))=真)\}$, $CDOM(M_{sel})=\{m\mid\exists e(e\in D_{sel}\wedge P(e\times F_R(e))=真\wedge F_R(e)=m)\}$,而且$\forall e\in CDOM(D_{sel})$, $F_{Select}(e)=F_R(e)$.

定义 2.7(投影). 设 $R=(D,M,Dstr)$, $Ins(R)=F_R$. R 上的投影操作表示为 $Project(R,\pi)$,其中 $\pi\subseteq M$ 是投影属性集合. $Project(R,\pi)$是一个如下定义的多维数据集合:

(1) $Sch(Project(R,\pi))=(D,\pi,Dstr)$.

(2) $Ins(Project(R,\pi))$是映射 $F_{Project}:CDOM(D)\to CDOM(\pi)$,而且$\forall e\in CDOM(D)$,如果 π 为非空集合,$F_{Project}(e)=F_\pi(e)$,其中 $F_\pi(e)$是 $F_R(e)$在度量属性集合 π 上的值,如果 π 为空集合,$F_{Project}(e)$为空.

注意,投影操作的投影属性集合限制在度量属性上.这是因为,如果在一个多维数据集合中去掉维属性,则度量属性值需要进行聚集计算.如果我们需要去掉某些维属性,可以使用后面定义的聚集操作.先使用聚集操作,再使用投影操作,我们可以达到既去掉某些维属性又去掉某些度量属性的目的.如果一个数据集合不包含度量属性,我们可以定义与关系模型中的投影操作完全相同的投影操作.大家都很熟悉这种投影操作,本文不再给出详细定义.

定义 2.8(维聚集). 设 $R=(D,M,Dstr)$, $D=\{d_1,d_2,\ldots,d_n\}$, $M=\{M_1,M_2,\ldots,M_k\}$, $Dstr=\{(\alpha_1,\leqq,\theta_1),(\alpha_2,\leqq,\theta_2),\ldots,(\alpha_n,\leqq,\theta_n)\}$, $Ins(R)=F_R$. R 在维 d_j 上的维聚集操作表示为 $Dagg(R,d_j,\Psi)$,其中 $d_j\in\{d_1,d_2,\ldots,d_n\}$称为聚集属性. $\Psi=\{f_1,f_2,\ldots,f_k\}$是聚集函数集合,$f_i$ 作用于度量属性 M_i. $Dagg(R,d_j,\Psi)$是一个如下定义的多维数据集合:

(1) $Sch(Dagg(R,d_j,\Psi))=(D-\{d_j\},M,Dstr-\{(\alpha_j,\leqq,\theta_j)\})$.

(2) $Ins(Dagg(R,d_j,\Psi))$是映射 $F_{Dagg}:CDOM(D-\{d_j\})\to f_1(M_1)\times f_2(M_2)\times\ldots\times f_k(M_k)$. $f_i(M_i)$定义如下:$\forall(x_1,\ldots,x_{j-1},x_{j+1},\ldots,x_n)\in CDOM(D-\{d_j\})$,令 $S(x_1,\ldots,x_{j-1},x_{j+1},\ldots,x_n)=\{z_i\mid F_R(x_1,\ldots,x_{j-1},x_j,x_{j+1},\ldots,x_n)=(z_1,\ldots,z_i,\ldots,z_k)\in CDOM(M)\}$, $f_i(M_i)=\{y_i\mid\forall(x_1,\ldots,x_{j-1},x_{j+1},\ldots,x_k)\in CDOM(D-\{d_j\}), y_i=f_i(S(x_1,\ldots,x_{j-1},x_{j+1},\ldots,x_k))\}$. F_{Dagg}满足:$\forall e=(x_1,\ldots,x_{j-1},x_{j+1},\ldots,x_k)\in CDOM(D-\{d_j\})$, $F_{Dagg}(e)=(y_1,y_2,\ldots,y_k)$,其中 $y_i=f_i(S(x_1,\ldots,x_{j-1},x_{j+1},\ldots,x_n))$.

定义 2.9(层次聚集). 设 $R=(D,M,Dstr)$, $D=\{d_1,d_2,\ldots,d_n\}$, $M=\{M_1,M_2,\ldots,M_k\}$, $Dstr=\{(\alpha_1,\leqq,\theta_1),(\alpha_2,\leqq,\theta_2),\ldots,(\alpha_n,\leqq,\theta_n)\}$, $Ins(R)=F_R$, R 是某个数据集合 R'在维 d_i 上经过 t 次层次聚集操作后的结果,$S_1\leqq S_2\leqq\ldots\leqq S_t\leqq S_{t+1}\leqq\ldots\leqq S_{t+p}$是在 R'的 d_i 维上第 1 次进行聚集操作之前维 d_i 的一个层次链. $S_{t+1}\leqq S_{t+2}\leqq\ldots\leqq S_{t+p}$是 R 的维 d_i 的一个层次链. R 在这个层次链上的层次聚集操作表示为 $Hagg(R,d_i,(S_{t+1}\leqq S_{t+2}\leqq\ldots\leqq S_{t+p}),\Psi)$,其中 $\Psi=\{f_1,f_2,\ldots,f_k\}$是聚集函数集合,$\Psi\in\varphi$, $(S_{t+1},S_{t+2},\varphi)\in\theta_i$, f_j 作用于度量属性 M_j. $HHagg(R,d_i,(S_{t+1}\leqq S_{t+2}\leqq\ldots\leqq S_{t+p}),\Psi)$是一个如下定义的多维数据集合:

(1) $Sch(Hagg(R,d_i,(S_{t+1}\leqq S_{t+2}\leqq\ldots\leqq S_{t+p}),\Psi))=(D_{Hagg},M_{Hagg},Dstr_{Hagg})$,其中 $D_{Hagg}=D$, $M_{Hagg}=M$, $Dstr_{Hagg}=Dstr-\{(\alpha_i,\leqq,\theta_i)\}\cup\{(\alpha'_i,\leqq,\theta'_i)\}$, $\alpha'_i=\{S_{t+2},\ldots,S_{t+p}\}$, $\theta'_i=\{(S_v,S_u,\varphi)\mid(S_v,S_u\in\alpha'_i)\wedge((S_v,S_u,\varphi)\in\theta_i)\}$.

(2) $Ins(HHagg(R,d_j,(S_{t+1}\leqq S_{t+2}\leqq\ldots\leqq S_{t+p}),\Psi))$是映射 $F_{Hagg}:CDOM(D_{Hagg})\to f_1(M_1)\times f_2(M_2)\times\ldots\times$

$f_k(M_k).CDOM(D_{Hagg})\subseteq CDOM(d_1)\times\ldots\times CDOM(d_{i-1})\times((CDOM(d_i)-S_{i+p}\times\ldots\times S_{i+2})\cup(S_{i+p}\times\ldots\times S_{i+2}))\times CDOM(d_{i+1})\times\ldots\times CDOM(d_n)$. 对于 $1\leqslant i\leqslant k$, $f_i(M_i)=\{v|\forall(x_1,\ldots,x_i,\ldots,x_n)\in CDOM(D_{Hagg}),x_i\notin S_{i+p}\times\ldots\times S_{i+2},F_R(x_1,\ldots,x_i,\ldots,x_n)=v\}\cup AS_i$. AS_i 定义如下:$\forall(x_1,\ldots,x_{i-1},(y_{i+p},\ldots,y_{i+2}),x_{i+1},\ldots,x_n)\in CDOM(D_{Hagg})$,令 $S(x_1,\ldots,x_{i-1},(y_{i+p},\ldots,y_{i+2}),x_{i+1},\ldots,x_n)=\{z_i|\forall e=(x_1,\ldots,x_{i-1},(y_{i+p},\ldots,y_{i+2},y_{i+1}),x_{i+1},\ldots,x_n)\in CDOM(D),F_R(e)=(z_1,\ldots,z_i,\ldots,z_k)\in CDOM(M)\}$, $AS_i=\{v_i|\forall(x_1,\ldots,x_{i-1},(y_{i+p},\ldots,y_{i+2}),x_{i+1},\ldots,x_n)\in CDOM(D_{Hagg}),v_i=f_i(S(x_1,\ldots,x_{i-1},(y_{i+p},\ldots,y_{i+2}),x_{i+1},\ldots,x_n))\}$. F_{Hagg} 满足:$\forall e=((x_1,\ldots,x_i,\ldots,x_n)\in CDOM(D_{Hagg})$,如果 $x_i=(y_{i+p},y_{i+p-1},\ldots,y_{i-2})\in S_{i+p}\times\ldots\times S_{i+2}$, $F_{Hagg}(e)=(v_1,v_2,\ldots,v_k)$,其中 $v_i=f_i(S(x_1,\ldots,x_{i-1},(y_{i+p},\ldots,y_{i+2})),x_{i+1},\ldots,x_n))$,否则 $F_{Hagg}(e)=F_R(e)$.

2.2 宏操作

2.2.1 连接、广义维聚集、切片、切块和数据方体

定义 2.10(连接). 设 $R=(D_R,M_R,Dstr_R)$和 $S=(D_S,M_S,Dstr_S)$是两个多维数据集合,$Ins(R)=F_R$, $Ins(S)=F_S$. R 与 S 的连接定义为 $Join(R,S,P)=Select(R\times S,P)$,其中 P 是连接条件.

定义 2.11(广义维聚集). 设 $R=(D,M,Dstr)$是一个数据集合. R 上的广义维聚集操作表示为 $Gagg(R,\{d_1,d_2,\ldots,d_m\},\Omega)$,其中 $\{d_1,d_2,\ldots,d_m\}\subseteq D$ 是聚集属性集合,$\Omega=\{\Psi_1,\Psi_2,\ldots,\Psi_m\}$是聚集函数集合族,$\Psi_i$ 是对 d_i 进行维聚集时使用的聚集函数集合. 广义聚集操作可以递归地定义如下:

$Gagg(R,\{d_1\},\{\Psi_1\})=Dagg(R,d_1,\Psi_1)$,

$Gagg(R,\{d_1,d_2,\ldots,d_m\},\Omega)=Dagg(Gagg(R,\{d_1,d_2,\ldots,d_{m-1}\},\{\Psi_1,\ldots,\Psi_{m-1}\}),d_m,\Psi_m)$.

定义 2.12(数据方体). 设 $R=(D,M,Dstr)$. R 上的数据方体操作表示为 $Cube(R,\Omega)$,其中 $\Omega=\{\Psi_S|S\in 2^D\}$是聚集函数集族,$\Psi_S$ 是对 D 的子集合 S 进行聚集操作时使用的聚集函数族. $Cube(R,\Omega)=\bigcup_{S\in 2^D}Gagg(R,S,\Psi_S)$.

定义 2.13(切片). 设 $R=(D,M,Dstr)$, $D=\{d_1,d_2,\ldots,d_n\}$, $M=\{M_1,M_2,\ldots,M_k\}$, $Dstr=\{(\alpha_1,\leqslant,\theta_1),(\alpha_2,\leqslant,\theta_2),\ldots,(\alpha_n,\leqslant,\theta_n)\}$, $\alpha_i=\{A_1,\ldots,A_m\}$. 在 R 的第 i 维上的切片操作表示为 $Slice(R,A_1=a_1,\ldots,A_m=a_m)=Select(R,P)$,其中 $P=(A_1=a_1)\wedge\ldots\wedge(A_m=a_m)$, a_i 是常量.

定义 2.14(切块). 设 $R=(D,M,Dstr)$, $D=\{d_1,d_2,\ldots,d_n\}$, $M=\{M_1,M_2,\ldots,M_k\}$, $Dstr=\{(\alpha_1,\leqslant,\theta_1),(\alpha_2,\leqslant,\theta_2),\ldots,(\alpha_n,\leqslant,\theta_n)\}$, $\alpha_i=\{A_1,\ldots,A_m\}$. 在 R 的第 i 维上的切块操作表示为 $Dicing(R,a_1\leqslant A_1\leqslant b_1,\ldots,a_m\leqslant A_m\leqslant b_m)=Select(R,P)$,其中 $P=(a_1\leqslant A_1\leqslant b_1)\wedge\ldots\wedge(a_m\leqslant A_m\leqslant b_m)$, a_i 和 b_i 是常量.

2.2.2 层次 Roll-up 和 Drill-down

层次 Roll-up 和 Drill-down 操作的目的是按照一个多维数据集合的指定维的一个层次链上下浏览具有不同详细程度的数据集合. 在执行层次 Roll-up 和 Drill-down 之前,需要指定数据集合的一个维层次链,并定义一个控制浏览的层次游标. 层次链的指定和游标的定义由下边的定义游标操作完成. 层次游标是一个四元组 $(Dname,Dim,Horder,Hpointer)$. Dname 是游标所属的数据集合名. Dim 是维名字,指定了层次 Roll-up 和 Drill-down 操作在哪一维上进行. Horder 是 Dim 的一个层次链,说明了层次 Roll-up 和 Drill-down 操作所遵循的维层次结构. Hpointer 是指针,指向目前要浏览的数据集合在层次链中的位置. 层次 Roll-up 和 Drill-down 操作返回 Hpointer 指针对应的数据集合. 层次 Roll-up 和 Drill-down 操作可以执行多次. 每次执行完层次 Roll-up 操作,Hpointer 沿 Horder 指定的层次链向上滚动一层. 每次执行完层次 Drill-down 操作,Hpointer 沿 Horder 指定的层次链向下滚动一层.

定义 2.15(定义层次游标). 设 $R=(D,M,Dstr)$, $D=\{d_1,d_2,\ldots,d_n\}$, $M=\{M_1,M_2,\ldots,M_k\}$, $Dstr=\{(\alpha_1,\leqslant,\theta_1),(\alpha_2,\leqslant,\theta_2),\ldots,(\alpha_n,\leqslant,\theta_n)\}$. R 的层次游标定义操作表示为 $Hcursor(flag,R,d_i,(S_0\leqslant S_1\leqslant\ldots\leqslant S_{m-1}),(\Psi_1,\Psi_2,\ldots,\Psi_{m-1}),Hcs)$,其中 $flag=RU$ 或 DM, RU 表示定义 Roll-up 游标,DM 表示定义 Drill-down 游标,$S_0\leqslant S_1\leqslant\ldots\leqslant S_{m-1}$ 是维 d_i 的一个层次链,$\Psi_i=\{f_{i1},f_{i2},\ldots,f_{ik}\}$是聚集函数集合,$\Psi_i\subseteq\varphi,(S_{i-1},S_i,\varphi)\in\theta_i$. 该操作定义了层次游标 $Hcs=(R,d_i,(S_0\leqslant S_1\leqslant\ldots\leqslant S_{m-1}),S)$(其中 $S=S_0$,如果 $flag=RU$;或 $S=S_{m-1}$,如果 $flag=DM$),并计算出如下 m 个数据集合:$R_0=R$, $R_1=Hagg(R_0,d_i,(S_0\leqslant S_1\leqslant\ldots\leqslant S_{m-1}),\Psi_1)$, $R_2=Hagg(R,d_i,$

$(S_1 \subseteq S_2 \subseteq \ldots \subseteq S_{m-1}), \Psi_2), \ldots, R_{m-1} = Hagg(R_{m-2}, d_i, (S_{m-2} \subseteq S_{m-1}), \Psi_{m-1})$.

定义 2.16(层次 Roll-up). 设 $R=(D,M,Dstr)$与定义 2.15 中的数据集合相同. R 上的层次 Roll-up 操作表示为 $Hroll\text{-}up(R,Hcs)$,其中 Hcs 是预先定义的游标. 设游标 Hcs 的当前值为$(R,d_i,(S_0 \subseteq S_1 \subseteq \ldots \subseteq S_{m-1}),S_j)$, $Hroll\text{-}up(R,Hcs)=R_j(R_j$ 在游标 Hcs 定义时已经计算出). $Hroll\text{-}up(R,Hcs)$操作执行完成后,$Hcs=(R,d_i,(S_0 \subseteq S_1 \subseteq \ldots \subseteq S_m),S_{(j+1) \bmod m})$.

定义 2.17(层次 *Drill-down*). 设 $R=(D,M,Dstr)$与定义 2.15 中的数据集合相同. R 上的层次 Drill-down 操作表示为 $Drill\text{-}down(R,Hcs)$,其中 Hcs 是预先定义的游标. 设游标 Hcs 的当前值为$(R,d_i,(S_0 \subseteq S_1 \subseteq \ldots \subseteq S_{m-1}),S_j)$, $Hdrill\text{-}down(R,Hcs)=R_j(R_j$ 在游标 Hcs 定义时已经计算出). $Hdrill\text{-}down(R,Hcs)$操作执行完成后,$Hcs=(R,d_i,(S_0 \subseteq S_1 \subseteq \ldots \subseteq S_{m-1}),S_{(j-1) \bmod m})$.

2.2.3 维 Roll-up 和 Drill-down

类似于层次 Roll-up 和 Drill-down 操作,维 Roll-up 和 Drill-down 的目的是按照多维数据集合的维的子集合之间的集合包含序上下浏览具有不同详细程度的数据集合. 执行维 Roll-up 和 Drill-down 之前,需要定义一个控制浏览的维游标. 维游标是一个三元组(*Dname*, *Dorder*, *Dpointer*). Dname 是游标所属的数据集合名. Dorder 是维的子集合之间的集合包含序,说明了维 Roll-up 和 Drill-down 操作所遵循的层次结构. Dpointer 是指针,指向目前要浏览的数据集合在层次结构中的位置. 维 Roll-up 和 Drill-down 操作返回 Dpointer 指针对应的数据集合. 维 Roll-up 和 Drill-down 操作可以执行多次. 每次执行完 Roll-up 操作,Dpointer 沿 Dorder 指定的层次链向上滚动一层. 每次执行完 Drill-down 操作,Dpointer 沿 Dorder 指定的层次链向下滚动一层.

定义 2.18(定义维游标). 设 $R=(D,M,Dstr)$, $D=\{d_1,d_2,\ldots,d_n\}$, $M=\{M_1,M_2,\ldots,M_k\}$, $Dstr=\{(a_1,\leqq,\theta_1),(a_2,\leqq,\theta_2),\ldots,(a_n,\leqq,\theta_n)\}$. R 的维游标定义操作表示为 $Dcursor(flag,R,(S_0 \supseteq S_1 \supseteq \ldots \supseteq S_{m-1}),(\Psi_0,\Psi_1,\Psi_2,\ldots,\Psi_{m-1}),Dcs)$,其中 $flag=RU$ 或 DM, RU 表示定义 Roll-up 游标, DM 表示定义 Drill-down 游标, S_i 是 D 的子集, $S_0 \supseteq S_1 \supseteq \ldots \supseteq S_{m-1}$称为 R 的一个维序, $\Psi_i=\{f_{i1},f_{i2},\ldots,f_{ik}\}$是聚集函数集合. 该操作定义游标 $Dcs=(R,(S_0 \supseteq S_1 \supseteq \ldots \supseteq S_{m-1}),S)$(其中 $S=S_0$,如果 $flag=RU$;或 $S=S_{m-1}$,如果 $flag=DM$),并计算出如下 m 个数据集合: $R_0=Gagg(R,D-S_0,\Psi_0)$, $R_1=Gagg(R_0,S_0-S_1,\Psi_1),\ldots,R_{m-1}=Gagg(R_{m-2},S_{m-2}-S_{m-1},\Psi_{m-1})$.

定义 2.19(维 Roll-up). 设 $R=(D,M,Dstr)$与定义 2.18 中的数据集合相同. R 的维 Roll-up 操作表示为 $Droll\text{-}up(R,Dcs)$,其中 Dcs 是预先定义的游标. 设游标 Dcs 的当前值为$(R,(S_0 \supseteq S_1 \supseteq \ldots \supseteq S_{m-1}),S_j)$,则 $Droll\text{-}up(R,Dcs)=R_j(R_j$ 在游标 Dcs 定义时已经计算出). $Droll\text{-}up(R,Hcs)$操作执行完成后,$Dcs=(R,(S_0 \supseteq S_1 \supseteq \ldots \supseteq S_{m-1}),S_{(j+1) \bmod m})$.

定义 2.20(维 Drill-down). 设 $R=(D,M,Dstr)$与定义 2.18 中的数据集合相同. R 的维 Drill-down 操作表示为 $Drill\text{-}down(R,Dcs)$,其中 Dcs 是预先定义的游标. 设 Dcs 的当前值为$(R,(S_0 \supseteq S_1 \supseteq \ldots \supseteq S_m),S_j)$,则 $Droll\text{-}up(R,Dcs)-R_j(R_j$ 在游标 Dcs 定义时已经计算出). $Ddrill\text{-}down(R,Hcs)$操作执行完成后,$Dcs=(R,(S_0 \supseteq S_1 \supseteq \ldots \supseteq S_{m-1}),S_{(j-1) \bmod m})$.

2.3 数据集合维护操作

数据集合维护操作分为两类,一类是数据维护操作,另一类是模式维护操作. 由于数据维护比较简单,本文不作讨论. 下面我们给出部分数据集合模式维护操作的定义.

定义 2.21(增加维). 设 $R=(D,M,Dstr)$, $D=\{d_1,d_2,\ldots,d_n\}$, $M=\{M_1,M_2,\ldots,M_k\}$, $Dstr=\{(a_1,\leqq,\theta_1),(a_2,\leqq,\theta_2),\ldots,(a_n,\leqq,\theta_n)\}$, $Ins(R)=F_R$. R 上的增加维操作表示为 $Adim(R,d,(\alpha,\leqq,\theta))$,其中 $d \notin D$ 是新增维,$(\alpha,\leqq,\theta)$是 d 的结构. $Adim(R,d,(\alpha,\leqq,\theta))$是一个如下定义的多维数据集合:

(1) $Sch(Adim(R,d,(\alpha,\leqq,\theta)))=(D \cup \{d\},M,Dstr \cup \{(\alpha,\leqq\theta)\})$.

(2) $Ins(Adim(R,d,(\alpha,\leqq,\theta)))$是一个映射 $F_{Adim}:CDOM(D) \times CDOM(d) \to CDOM(M)$, $\forall e=(x_1,x_2,\ldots,x_n,x) \in CDOM(D) \times CDOM(d)$, $F_{Adim}(e)=F_R(x_1,x_2,\ldots,x_n) \times \{$空值$\}$.

注意,新维属性的值初始地确定为空值,以后可以使用数据维护操作进行修改.

定义 2.22(删除维). 设 $R=(D,M,Dstr)$, $D=\{d_1,d_2,\ldots,d_n\}$, $M=\{M_1,M_2,\ldots,M_k\}$, $Dstr=\{(a_1,\leqq,\theta_1),(a_2,\leqq,\theta_2),\ldots,(a_n,\leqq,\theta_n)\}$, $Ins(R)=F_R$. R 上的删除维操作表示为 $Ddim(R,d_i,\Psi)$,其中 $d_i \in D$ 是删除的维,

Ψ 是聚集函数集合. $Ddim(R,d_i,\Psi)=Dagg(R,d_i,\Psi)$.

定义 2.23(增加度量属性). 设 $R=(D,M,Dstr)$, $D=\{d_1,d_2,\ldots,d_n\}$, $M=\{M_1,M_2,\ldots,M_k\}$, $Dstr=\{(\alpha_1,\leqq,\theta_1),(\alpha_2,\leqq,\theta_2),\ldots,(\alpha_n,\leqq,\theta_n)\}$, $Ins(R)=F_R$. 在 R 上增加度量属性的操作表示为 $Amea(R,Meas)$, 其中 $Meas$ 不在 M 是新增度量属性. $Amea(R,Meas)$ 是一个如下定义的多维数据集合:

(1) $Sch(Amea(R,Meas))=(D,M\cup\{Meas\},Dstr)$.

(2) $Ins(Amea(R,Meas))$ 是映射 $F_{Amea}:CDOM(D)\rightarrow CDOM(M\cup\{Meas\})$, $\forall e\in CDOM(D)$, $F_{Amea}(e)=F_R(e)\times\{$空值$\}$.

注意,新增度量属性的值初始地确定为空值,以后可以使用数据维护操作进行修改.

定义 2.24(删除度量属性). 设 $R=(D,M,Dstr)$, $D=\{d_1,d_2,\ldots,d_n\}$, $M=\{M_1,M_2,\ldots,M_k\}$, $Dstr=\{(\alpha_1,\leqq,\theta_1),(\alpha_2,\leqq,\theta_2),\ldots,(\alpha_n,\leqq,\theta_n)\}$, $Ins(R)=F_R$. 删除 R 的一个度量属性的操作表示为 $Dmea(R,M_i)$, 其中 $M_i\in\{M_1,M_2,\ldots,M_k\}$ 是被删除的度量属性. $Dmea(R,M_i)=Project(R,M-M_i)$.

定义 2.25(增加维层次属性). 设 $R=(D,M,Dstr)$, $D=\{d_1,d_2,\ldots,d_n\}$, $M=\{M_1,M_2,\ldots,M_k\}$, $Dstr=\{(\alpha_1,\leqq,\theta_1),(\alpha_2,\leqq,\theta_2),\ldots,(\alpha_n,\leqq,\theta_n)\}$, $Ins(R)=F_R$. 在 R 的 d_i 维的约束非奇异聚集偏序集族 $(\alpha_i,\leqq,\theta_i)$ 中增加一个新集合 S 的操作,表示为 $Adha(R,d_i,S,UP,LOW,\Phi_1,\Phi_2)$, 其中 $UP,LOW\subseteq\alpha_i$, $\forall up\in UP$, $\forall low\in LOW$, $up\leqq S\leqq low$, UP 和 LOW 不能同时为空集合, Φ_1 是 UP 中元素与 S 之间的聚集约束集合, Φ_2 是 S 与 LOW 中元素之间的聚集约束集合. $Adha(R,d_i,S,UP,LOW,\Phi_1,\Phi_2)$ 是一个如下定义的多维数据集合:

(1) $Sch(Adha(R,d_i,S,UP,LOW,\Phi_1,\Phi_2))=(D,M,Dstr-\{(\alpha_i,\leqq,\theta_i)\}\cup\{(\alpha_i\cup\{S\},\leqq,\theta_i\cup\{(up,S,\varphi_1)\mid\forall up\in UP,\varphi\in\Phi_1$ 是 $up\leqq S$ 的聚集约束$\}\cup\{(S,low,\varphi_2)\mid\forall low\in LOW,\varphi_2\in\Phi_2$ 是 $S\leqq low$ 的聚集约束$\})\})$.

(2) $Ins(Adha(R,d_i,S,UP,LOW,\Phi_1,\Phi_2))$ 是映射 $F_{Adha}:CDOM(D)\rightarrow CDOM(M)$, $\forall e=(x_1,\ldots,x_{i-1},x_i,x_{i+1},\ldots,x_n)\in CDOM(D)$, 如果 x_i 与任何包含 S 的层次链相关,则 $F_{Adha}(e)=($空值,空值,$\ldots$,空值$)$, 否则 $F_{Adha}(e)=F_R(e)$.

注意,因为与包含新增维层次属性的层次链相关的度量值已经无意义,故被初始地确定为空值,以后可以使用数据维护操作进行修改.

3 与相关工作的比较

到目前为止,人们一共提出了 9 种多维数据模型[3~10]. 这 9 种多维数据模型可以分为 3 类:简单的多维数据模型、结构化的多维数据模型和统计对象模型.

简单多维数据模型[5,8,10]把数据集合视为多维空间中的点集,把数据集合的属性分类为维和度量(或事实)属性两类. 维属性用来描述度量属性,是多维空间的维度. 度量属性的值用来进行分析处理,是多维空间中的点. 简单多维数据模型具有一个致命的弱点,即没有维层次结构的概念和语义,不能表示维层次结构. 前面给出的黑龙江省移动电话局数据仓库中的多维数据集合实例就不能由简单多维数据模型来表示.

结构化多维数据模型[3,6,7,9]考虑了如何表示多维数据集合的维层次结构的问题. 文献[3,6]提出的多维数据模型只是部分地间接支持维层次结构的表示,而不能直接地表示多维数据集合的完整维层次结构. 文献[6]提出的数据模型通过组合多个维关系的方法表示维层次结构. 文献[3]提出的数据模型通过维合并函数来表示维层次结构. 文献[7]提出的数据模型能够明确地支持维层次结构的表示. 但是,它只允许每个维具有单层次路径(即我们的数据模型中定义的层次链). 图 2 和图 3 表示的多维数据集合的维都具有两个以上的层次路径,不能由文献[3,6,7]提出的多维数据模型来表示. 文献[9]提出的数据模型能够支持维层次结构,而且能够表示一个维的多个层次路径. 但是,文献[9]提出的数据模型要求维层次结构必须是一个代数格. 图 2 和图 3 表示的多维数据集合的维不是代数格. 这个数据集合不能由文献[9]提出的数据模型来表示. 这样的数据集合在实际应用领域是很常见的.

统计对象模型[4]支持结构化的分类层次. 但是,每个结构化的分类层次必须与一个特定的聚集函数相关. 而且,每个结构化的分类层次只能定义在一个度量属性上,用来回答特定的统计分析查询. 显然,统计对象模型具有很大的局限性,缺少灵活性.

从上面的讨论可以看到,现有的多维数据模型在数据结构的表达能力上具有很大的不足.除此之外,现有的多维数据模型都没有提供完整的OLAP操作代数,操作能力不够完善,也不利于OLAP查询的优化处理.尤其值得注意的是,现有的多维数据模型没有或极少提供与维层次结构相关的OLAP操作.

针对现有多维数据模型在数据结构表达能力和数据操作能力方面的问题,本文基于偏序和映射的概念提出了一个新的多维数据模型,克服了现有的多维数据模型的不足.

4 结论与继续研究的问题

本文提出了一种新的数据仓库的多维数据模型.该模型以偏序和映射概念为基础,提供了很强的复杂维层次结构表达机制,能够有效地表达数据仓库的各种复杂层次数据结构和语义.这个模型包括一个以OLAP操作为核心的操作代数,可以有效地支持OLAP应用.它引进了层次结构的聚集约束概念,提供了表达聚集约束的机制.它允许在多维数据集合的任一维的同一个层次链上使用不同的聚集函数执行Roll-up和Drill-down操作,允许维数据中包括描述度量属性的层次聚集语义的信息.与其他多维数据模型相比,本文提出的多维数据模型是一个具有很强表达能力和完整OLAP操作的多维数据模型.目前,我们正在研究基于本文提出的数据模型的数据定义和操纵语言、基于该数据模型的数据查询优化和处理方法.

参考文献

1 Coliat G. OLAP, relational, and multidimensional database system. SIGMOD Record, 1996,25(3):64~69

2 Codd E F. Providing OLAP (on-line analytical processing) to user-analysts: an IT mandate. Technical Report, TR-9300011, E. F. Codd and Associates, 1993

3 Agrawal R, Gupta A, Sarawagi S. Modeling mulidimensional databases. In: Jackson M, Pu C eds. Proceedings of the 13th International Conference on Data Engineering. Los Alamitos, CA: IEEE Society Press, 1997. 105~116

4 Rafanelli M, Shoshani A. STORM: a statistical object representation model. Data Engineering Bulletion, 1990,13(3):12~18

5 Gyssens M, Lakshmanan L V S. A foundation for multi dimensional databases. In: Dayal U, Gray P M D, Nishio S eds. Proceedings of the the 23rd Conference on very large data bases. San Francisco, CA: Morgan Kaufmann Publishers, Inc., 1997. 106~115

6 Li C, Wang X S. A data model for supporting on-line analytical processing. In: Barker K, Manitoba U eds. Proceedings of the 5th International Conference on Information and Knowledge Management. New York: Springer-Verlag, 1996. 81~88

7 Lehner W. Modeling large scale OLAP scenarios. In: Jorg H, Ramos I eds. Proceedings of the 6th International Conference on Extending Database Technology. New York: Springer-Verlag, 1998. 153~167

8 Datta A, Thomas H. A conceptual model and algebra for on-line analytical processing in decision support databases. In: Thomas H eds. Proceedings of the 7th Annual Workshop on Information Technologies and Systems. San Mateo, CA: Morgan Kaufman Publishers, Inc., 1997. 91~100

9 Pedersen T B, Jensen C S. Multidimensional data modeling for complex data. In: Kitsuregawa M, Maciaszek L, Papazoglou M *et al* eds. Proceedings of the 15th International Conference on Data Engineering. Los Alamitos, CA: IEEE Society Press, 1999. 336~345

10 Gray J, Bosworth A, Layman A *et al*. Data cube: a relational aggregation operator generalizing group-by, cross tab and sub-totals. Data Mining and Knowledge Discovery, 1997,1(1):29~54

Multidimensional Data Modeling for Data Warehouses

LI Jian-zhong GAO Hong

(*Department of Computer Science and Engineering Harbin Institute of Technology Harbin* 150001)

Abstract Data model is a basic aspect in the research field of data warehouses. It has been argued that traditional data models, such as the ER model and the relational model, are in principle not powerful enough for modeling the data structure and semantics of data warehouse and supporting OLAP (on-line analysis processing). As a result, several multidimensional models based on multidimensional view of data have emerged. However, these multidimensional data models still fall short of ability to model complex data in some real-world application domains and to support complete OLAP operations. In this paper, the authors propose a new multidimensional data model based on the concepts of partial order and mapping. This model addresses supporting for complex data structure and semantics of data warehouses, especially complex hierarchies in dimensions. Along with the model, they also present an associated algebra that includes a complete set of OLAP operations and supports complex aggregation, roll-up and drill-down along hierarchies in dimensions. A new concept of aggregation function constraint is also presented in this paper, and the mechanism for expressing and checking the aggregation function constraint is included in the model.

Key words Data warehouse, data model, multidimensional data model, OLAP (on-line analysis processing).