

基于函数依赖的 XML 文档的研究

周 莉,王 珏

(华东交通大学 信息工程学院,江西 南昌 330013)

摘要:从数据库设计的角度出发,对 XML 数据的函数依赖进行深入的研究,提出了基于树模型和路径表达式上的 XML 函数依赖的定义及推理规则。对于利用 XML 实现异构数据库的交换技术上,不仅在理论上具有指导价值,而且在实际应用中也具有现实意义和实用价值。

关键词:XML;约束;DTD;函数依赖

中图分类号:TP311

文献标识码:A

可扩展的标记语言 XML(Extensible Markup Language, XML)已经成为 Internet 上的主流数据。自描述、可扩展等优势使其适于充当 Internet 环境下数据描述标准与异构应用间数据交换载体的角色,这就对 XML 数据库模式提出了更高的要求,但 XML 文档是半结构化的,虽然它很容易表达来自不同源的数据,但是其所能表示的语义信息却相对有限,对于给定的关系模式中的约束条件及结构很难实现。由于 XML 数据库模式没有从数据库设计的角度来评价,这必将对以后的 Web 数据处理带来很大的麻烦,势必造成数据的大量冗余和插入、删除、更新等异常现象。XML 文档的语义结构非常复杂,很难从已有的文档中提取完整约束条件;要想解决这些异常现象,必须对 XML 从关系数据库设计角度来进一步研究,而函数依赖是关系数据库中保持语义的重点概念,也是范式研究得基础,XML 模式的理论远不如关系数据库理论成熟,还处于初级阶段,因此,要使 XML 模式数据库规范化^[1],首先就要对 XML 模式函数依赖进行深入研究,也是当前 XML 数据库理论工作者热门的研究课题。

1 XML 文档树模型

一棵 XML 树^[2] T 可以定义成为一个六元组 $T = (V, lab, ele, att, parent, val)$, 其中:

V 代表有限的结点集。

lab 是一个函数,它将返回结点 v 的标签。如果 $lab(v) \in E$, 则 v 是元素结点;如果 $lab(v) \in A$, 则 v 是属性结点;如果 $lab(v) = S$, 则 v 是文本结点。

ele 是一个函数。如果 v 是元素结点,则 $ele(v)$ 返回 v 所有元素和文本孩子组成的序列,即 $ele(v) = [v_{x1}, \dots, v_{xk}]$, 其中 $v_{x1}, \dots, v_{xk}$ 都是元素结点或是文本结点,且都是 v 的孩子。由于元素结点和文本结点是考虑次序的,所以 $ele(v)$ 是有次序的;如果 v 不是元素结点,则 ele 没有定义。

att 是一个函数。如果 v 是元素结点,则 $att(v)$ 返回 v 所有属性孩子组成的集合,即 $att(v) = \{a_1, \dots, a_m\}$, 其中 $a_1, \dots, a_m$ 都是属性结点,且都是 v 的孩子。由于属性结点不考虑次序,所以 $att(v)$ 是一个集合,里面的元素没有先后次序;如果 v 不是元素结点,则 att 没有定义。

val 是一个函数。如果 v 是元素结点,则 val 没有定义,如果 v 是属性结点或是文本结点,则 $val(v)$ 返回它的文本值。

$parent$ 是一个函数,它返回给定结点的双亲, v 中仅有一个结点无双亲,其它的结点仅有一个双亲。

在元素结点中唯一的一个结点 $root$ (代表树的根), 有 $root \in v$ 。且 $parent(root) = \Phi$, 用 r 来代表根结点。

收稿日期:2009-04-09

基金项目:江西省重点攻关项目(061216);江西省教育厅科技项目(GJJ08255)

作者简介:周 莉(1977-),女,江西南昌人,讲师,硕士,主要研究方向为数据库技术。

“相等”是函数依赖中的一个重要概念,在XML文档树上,定义如下。

定义1 (值相等):一棵XML树中两个结点 v_1 和 v_2 是值相等的,用 $v_1 = v_2$ 来表示,当且仅当下列条件成立:

1) $lab(v_1) = lab(v_2)$; 2) 如果 v_1, v_2 是属性结点或者是文本结点,则要求有 $val(v_1) = val(v_2)$; 3) 如果 v_1, v_2 是元素结点,则要求有 $att(v_1) = att(v_2); ele(v_1) = ele(v_2)$ 。

2 路径表达式

在XML树上,若从 v 到 v_1 有树上的一条边,称 v 是 v_1 的父节点, v_1 是 v 的子节点。若存在序列 $(v_0, v_1, \dots, v_n)$, 其中 v_i 是 $v_{i-1} (i = 1 \dots n)$ 的子节点, 则称 v_0 是 v_n 的祖先节点。将 v_{i-1} 到 v_i 的边依次连接, 构成树上的惟一的从 v_0 到 v_n 的路径^[3], 记为 $v_0 - v_n$ 。DTD 可以决定符合它的XML文档上的路径形式, 这就是路径表达式的概念。DTD 上的一个路径表达式可能对应文档中的若干条路径, 也就是说, 给定起始节点, 经由路径表达式所能到达的节点是一个集合。

定义2 D上的路径表达式P, 有两种表达形式: 一种是简单路径表达式PL_s; 另一种是正则路径表达式PL。其形式为:

$$PL_s: \rho ::= \epsilon \mid l. \rho \text{ 或 } PL: P ::= \epsilon \mid l \mid P.P' \mid _*$$

其中, $l \in E \cup A \cup \{S\}$, 是一个结点的标签; 使用符号 ϵ 来表示空路径表达式; “.” 是一个二元操作符, 它代表两个路径表达式的连接; “ ϵ ” 可以匹配任意标签 l ; “ $*$ ” 表示是 Kleene 闭包, 两者的组合 “ $_*$ ” 可以匹配任意路径。值得注意的是, 对一个路径 p 来说, 有 $P.\epsilon = \epsilon.P = P$ 。

一个简单路径表达式PL_s 指用一些标签所组成的序列, 其中不带任何通配符和 Kleene 闭包。由简单路径表达式定义的路径为简单路径。

一个正则路径表达式PL 可能包含 “ $_*$ ”, 其中的 “ $_*$ ” 就可以匹配任意路径。由正则路径表达式定义的路径为正则路径。

如果 ρ 是一个合法的简单路径, 当且仅当对 ρ 中的任意标签 l , 如果, $l \in A \cup \{S\}$, 则 l 必是 ρ 的最后一个标签。类似地, 可以定义合法的正则路径。

包含关系: 一个正则路径表达式定义的是一个简单路径的集合。用 $\rho \in P$ 表示 ρ 属于由正则路径表达式P定义的集合。

可达性: 在一个XML树T 中一个结点 v_2 可以由结点 v_1 经过简单路径 ρ 可达, 用 $T \mid \rho = \rho(v_1, v_2)$, 当且仅当

- 1) 如果 $\rho = \epsilon$, 则 $v_1 = v_2$
 - 2) 如果 $\rho = \rho'.l$, 则在T 中存在一个结点 v , 使得 $T \mid \rho' = \rho'(v_1, v)$, 且 v_2 是 v 的一个孩子, v_2 的标签是 l 。
- 结点集: 设 v 是XML树T 中的一个结点, 用 $v[\rho]$ 表示在树T 中, 从 v 开始, 经过路径 ρ 可达的所有结点的集合, 即 $v[\rho] = \{v' \mid T \mid \rho = \rho(v, v')\}$ 。类似地, 可以定义 $v[P] = \{v' \mid T \mid P = P(v, v')\}$ 。
- 值交集: 设 $v_1[P], v_2[P]$ 为两个结点集, 它们的值交集 $v_1[P] \cap v_2[P]$ 定义为:
- $v_1[P] \cap v_2[P] = \{(z, z') \mid \exists \rho \in P, z \in v_1[\rho], z' \in v_2[\rho], z = z'\}$ 。

3 XML上的函数依赖的定义

函数依赖是关系数据库设计理论的主要内容之一, 是范式研究的基础。数据依赖是指数据之间存在着各种内在的联系, 譬如键就是一种特殊的依赖。在关系模式中, 函数依赖的形式化定义为:

定义3 (函数依赖) 设有关系模式 $R(U)$, X 和 Y 是属性集 U 的子集, 函数依赖是形为 $f: X \rightarrow Y$ 的一个命题, 只要 r 是 R 的当前值, 对 r 中任意两个元组 t, s , 若有 $t[X] = s[X]$ 成立, 则必有 $t[Y] = s[Y]$ 成立, 那么函数依赖 $X \rightarrow Y$ 就在关系模式 $R(U)$ 中成立。

为XML 提供表述类似约束的能力, 引入关系模式中的概念到XML 中, 称其为XML 上的函数依赖^[4]。函数依赖是数据语义的重要组成部分, 对于键的定义、完整性约束、查询优化和数据集成等有非常重要的

作用。基于前面给出的 DTD, XML 的树模型和路径表达式等概念定义, 下面给出形式化的 XML 上函数依赖的定义。

定义 4 设 D 是一个 DTD。在 D 上的一个 XML 函数依赖定义成一个如下的表达式:

$(Q, [P_{x1}, P_{x2}, \dots, P_{xn} \rightarrow P_y])$

其中, Q 是该函数依赖的头路径, 它是由一个简单的 XPath 表达式表示 (即不含通配符和 Kleene 闭包), 并且这个 XPath 表达式一定是从 XML 文档的根开始的, $P_{xi} (1 \leq i \leq n)$ 是一个左部实体类型, 它由一个元素结点名和一些属性结点 (可有可无) 组成, P_y 是一个右部实体类型, 它由一个元素结点名和一个属性结点 (可有可无) 组成。一个 XML 函数依赖 $(Q, [P_{x1}, P_{x2}, \dots, P_{xn} \rightarrow P_y])$ 描述如下: 对任意两个由 Q 标识的子树, 如果它们在 $P_{x1}, P_{x2}, \dots, P_{xn}$ 上相等且 P_y 存在的话, 则它们在 P_y 上也相等。

如果对于一个 XML 文档 T , 以上的 XML 函数依赖 δ 是成立的, 则说该文档是满足该函数依赖 δ 的, 记作 $T \models \delta$ 。而且, 对于一个函数依赖集 Σ 中任意一个函数依赖 δ , 如果都有 $T \models \delta$, 则说该文档是满足该函数依赖集的, 记作 $T \models \Sigma$ 。

定义 5 对于由一个 DTD 中的元素结点组成的集合 L , 它被称为是一个线性系, 当且仅当在 L 中所有结点组成的结合 $(v_{x1}, v_{x2}, \dots, v_{xk})$ 构成一个线性序, 满足 v_{xi} 是 v_{xi+1} 的双亲 $(1 \leq i \leq k-1)$ 。

定义 6 考虑以下的 DTD:

$\langle ! \text{ELEMENT } H_1(H_2^*) \rangle$

.....

$\langle ! \text{ELEMENT } H_m(E_1^*) \rangle$

$\langle ! \text{ELEMENT } E_1(E_2^*) \rangle$

.....

$\langle ! \text{ELEMENT } E_x(E_y^*) \rangle$

对于一个 XML 函数依赖 $(Q, [P_{x1}, P_{x2}, \dots, P_{xn} \rightarrow P_y])$, 其中 $Q = /H_1/\dots/H_m$, $P_{xi} (1 \leq i \leq n)$ 由集合 $\{E_1, \dots, E_x\}$ ($E_1, \dots, E_x$ 是不同的) 中的元素结点名字再带可选的属性结点组成, P_y 由 E_y 带一个可选的属性结点组成。这个 XML 函数依赖被称为是良构的, 当且仅当它符合下列条件:

- 1) $Q, P_{x1}, P_{x2}, \dots, P_{xn}$ 和 P_y 中的元素结点按顺序构成一个线性系;
- 2) $P_{x1}, P_{x2}, \dots, P_{xn}$ 中不能包含任何冗余左部实体类型。

定义 7 考虑以下的 DTD:

$\langle ! \text{ELEMENT } H_1(H_2^*) \rangle$

.....

$\langle ! \text{ELEMENT } H_{m-1}(H_m^*) \rangle$

$\langle ! \text{ELEMENT } H_m(E_1, \dots, E_x, E_y) \rangle$

对于一个 XML 函数依赖 $(Q, [P_{x1}, P_{x2}, \dots, P_{xn} \rightarrow P_y])$, 其中 $Q = /H_1/\dots/H_m$, $P_{xi} (1 \leq i \leq n)$ 。由集合 $\{E_1, \dots, E_x\}$ ($E_1, \dots, E_x$ 是不同的) 中的元素结点名字再带可选的属性结点组成, P_y 由 E_y 带一个可选的属性结点组成。如果 $P_{x1}, P_{x2}, \dots, P_{xn}$ 不含有任何冗余的左部实体类型, 则这个 XML 函数依赖被称为是扁平的。

4 XML 上的函数依赖推理规则

一条路径 P 被另一条路径 Q 所包含, 可以记作 $P \subseteq Q$, 当且仅当对任意的 XML 树 T , 以及 T 中任意的结点 v , 有 $v[P] \subseteq v[Q]$ 。如果 $P \subseteq Q$ 且 $Q \subseteq P$, 可以记作 $P = Q$ 。以下是路径包含规则:

- 对一个正则路径 P , 有 $P \cdot \epsilon \subseteq P, P \subseteq \epsilon \cdot P, P \cdot \epsilon \subseteq P$ 和 $P \subseteq P \cdot \epsilon$, 其中 ϵ 代表的是空路径。
- 对一个正则路径 P , 有 $P \subseteq P$ 。
- 对一个正则路径 P , 有 $P \subseteq *$ 。
- 对于路径 P 和 P' , Q 和 Q' , 如果有 $P \subseteq P', Q \subseteq Q'$, 则有 $P \cdot Q \subseteq P' \cdot Q'$ 。
- 对于路径 P, Q 和 R , 如果有 $P \subseteq Q, Q \subseteq R$, 那么有 $P \subseteq R$ 。

XML 函数依赖推理规则总共有七条, 分别如下所示。

第1条(自反律):在XML数据库中,函数依赖 $(Q, [P_{x1}, P_{x2}, \dots, P_{xk} \rightarrow P_{xi}])$ 成立,其中 $1 \leq i \leq k$ 。

第2条(外延性):若XML数据库中有一个函数依赖 $(Q, [P_{x1}, P_{x2}, \dots, P_{xk} \rightarrow P_y])$ 成立,且 $P \in \text{root}[Q]$,则 $(Q, [P, P_{x1}, P_{x2}, \dots, P_{xk} \rightarrow P_y])$ 成立。

第3条(传递律):在XML数据库中,函数依赖 $(Q, [P_{x1}, P_{x2}, \dots, P_{xk} \rightarrow P_{y1}])$ 成立且 $(Q, [P_{x1}, P_{x2}, \dots, P_{xk}, P_{y1} \rightarrow P_{y2}])$ 成立,则 $(Q, [P_{x1}, P_{x2}, \dots, P_{xk} \rightarrow P_{y2}])$ 成立。

第4条(合并律):在XML数据库中,函数依赖 $(Q, [P_{x1}, P_{x2}, \dots, P_{xk} \rightarrow P_{y1}])$ 和 $(Q, [P_{x1}, P_{x2}, \dots, P_{xk} \rightarrow P_{y2}])$ 成立,则 $(Q, [P_{x1}, P_{x2}, \dots, P_{xk} \rightarrow \{P_{y1}, P_{y2}\}])$ 成立。

第5条(分解性):在XML数据库中,函数依赖 $(Q, [P_{x1}, P_{x2}, \dots, P_{xk} \rightarrow \{P_{y1}, P_{y2}\}])$ 成立,则 $(Q, [P_{x1}, P_{x2}, \dots, P_{xk} \rightarrow P_{y1}])$ 成立。

第6条(域收缩性):在XML数据库中,函数依赖 $(Q, [Q_1, P_{x1}, P_{x2}, \dots, P_{xk} \rightarrow P_y])$ 成立,且 Q, Q_1 合法,则 $(Q, [Q_1, P_{x1}, P_{x2}, \dots, P_{xk} \rightarrow P_y])$ 。

第7条(化简规则):在XML数据库中,函数依赖 $(Q, [Q_1, P_{x1}, P_{x2}, \dots, P_{xk} \rightarrow P_y])$ 成立,且 $Q_1 \subseteq Q$,则 $(Q_1, [P_{x1}, P_{x2}, \dots, P_{xk} \rightarrow P_y])$ 成立。

5 结束语

随着XML成为Web上的数据表示和数据交换^[5]的标准,需要通过Web交换和处理的XML数据在大篇幅的增加,这就对XML数据库的模式提出了更高的要求。从数据库设计的角度出发,采用基于路径的表示方法对XML数据进行函数依赖的规范化研究,直接对Web数据进行处理,从而得到规范化的XML数据库模式^[6],这样不仅完整地保留了XML文档数据中的语义和结构信息,而且满足了数据库设计的要求。因此,对XML模式中的函数依赖进行研究具有重要的理论意义和实用价值。

参考文献:

- [1] 张忠平,曹顺良.基于约束的XML模式规范化研究[C].第二十届全国数据库学术会议论文集.北京:中国计算机学会,2003:35-40.
- [2] 陈桦,张亮.XML文档在关系数据库中的存储研究[J].黑龙江电力,2008,30(3):185-188.
- [3] 赵相国,王国任,等.XML的函数依赖[J].东北大学学报,2008,29(1):57-59.
- [4] 李高仕,刘先锋,黄海燕.XML文档规范化算法研究[J].计算机技术与发展,2008,18(7):49-52.
- [5] 王海波,耿晖,等.基于XML数据交换的实现[J].计算机应用,2001,21(4):67-68.
- [6] 刘翔,程文青,等.一种从XML建立关系数据库的模式映射方法[J].计算机技术与发展,2007,17(2):2-4.
- [7] 莫佳.XML数据的关系存储技术[J].重庆工学院学报(自然科学版),2007,21(9):128-132.

Research of XML Documents Based on Functional Dependency

ZHOU Li, WANG Yu

(School of Information Engineering, East China Jiaotong University, Nanchang 330013, China)

Abstract: Based on design of databank, we study XML functional dependency. XML functional dependency is presented based on path expression and tree model, and a sound complete set of inference rules of XML functional dependency is given. Implementing of data exchange between heterogeneous databases with XML has important theoretic significance and practical value.

Key words: XML, constraint, DTD, functional dependency

(责任编辑:吴泽九)