基于关系数据库映射的模糊数据 XML 建模

严 丽[1)] 马宗民[2)] 刘 健[2)] 张 富[2)]

[1)](东北大学软件学院 沈阳 110819)

[2)](东北大学信息科学与工程学院 沈阳 110819)

摘 要 在很多现实世界应用中存在着大量的不精确性和不确定性信息,正是由于这个原因,基于各类数据模型的模糊数据建模的研究已经广泛展开.当前互联网上存在大量的电子数据资源,XML 已经成为 Web 上信息表示和交换的标准,但现有的 XML 研究成果难以满足 Web 环境下智能化数据管理的需求.文中从数据库信息建模的角度入手,基于模糊集和可能性分布理论,识别出 XML 文档中的多粒度数据模糊性,进一步研究了模糊 XML 模型.在形式化给出映射规则的基础上,通过给出转化算法实现了从模糊关系数据库到模糊 XML 模型的转化.最后,通过例子证实了所提转化方法的有效性.文中所提出的转化方法为全面构建模糊 XML 数据管理体系提供了理论基础.

关键词 XML;模糊集;可能性分布;关系数据库;映射

中图法分类号 TP311 **DOI 号**: 10.3724/SP.J.1016.2011.00291

XML Modeling of Fuzzy Data with Relational Databases

YAN Li[1)] MA Zong-Min[2)] LIU Jian[2)] ZHANG Fu[2)]

[1)](*School of Software*, *Northeastern University*, *Shenyang* 110819)

[2)](*College of Information Science and Engineering*, *Northeastern University*, *Shenyang* 110819)

Abstract Information imprecision and uncertainty exist in many real-world applications and for this reason fuzzy data modeling has been extensively investigated in various data models. Currently, huge amounts of electronic data are available on the Internet, and XML has been the de-facto standard of information representation and exchange over the Web. Current efforts fall short in their ability to satisfy the requirement of intelligent management over XML data. Therefore, beginning with an investigation of data modeling, the authors identify multiple granularity of data fuzziness in XML and fuzzy XML data model based on the theory of fuzzy set and possibility distributions. The authors further investigate the formal conversions from the fuzzy relational databases to the fuzzy XML model, which is established by introducing a series of mapping rules, and demonstrate the validity of the proposed approach by using a practical case. The conversions from the fuzzy relational databases to the fuzzy XML model lay a theoretical foundation for establishing the overall management system of fuzzy XML data.

Keywords XML; fuzzy sets; possibility distributions; relational databases; mapping

---

收稿日期:2009-04-27;最终修改稿收到日期:2010-10-15.本课题得到国家自然科学基金(60873010,61073139)、教育部新世纪优秀人才支持计划(NCET-05-0288)和教育部中央高校基本科研业务费专项(N090504005,N100604017,N090604012)资助.**严 丽**,女,1964 年生,博士,副教授,主要研究方向为数据库与智能信息处理.**马宗民**,男,1965 年生,博士,教授,博士生导师,主要研究领域为智能数据与知识工程.E-mail: mazongmin@ise.neu.edu.cn.**刘 健**,男,1984 年生,博士研究生,主要研究方向为数据库与 XML 数据管理.**张 富**,男,1983 年生,博士研究生,主要研究方向为数据库建模与描述逻辑.

1 引 言

随着 Web 的广泛使用并产生海量的电子数据，Web 上的数据表示和交换已变得越来越重要，当前 XML 已经成为了 Web 上信息表示和交换的标准[1]. XML 及其相关标准的推出，为需要在 Web 上进行数据交换的应用开发(例如电子商务和供应链管理等)提供了技术支持，并因此产生了有关 XML 数据管理的需求，例如 XML 文档的存储和查询等. 为了存储、查询和更新 XML 数据，实现 XML 和数据库的集成是必要的[2]，包括关系数据库、面向对象数据库和对象-关系数据库在内的各类数据库已经用于与 XML 文档的相互映射[3-12]，例如 XML 到数据库的映射和数据库到 XML 的映射. 在这些种类的数据库中，关系数据库以其成熟的技术和广泛的应用，成为了实现 XML 数据管理的首选[6-9,12]. 此外，由于 XML 缺乏表示现实世界数据以及它们之间复杂内在语义关系的充足能力，因此使用其它方法描述数据范例、开发出概念数据模型、最后把这样的模型转换成 XML 是非常必要的. 最近的研究工作已经致力于 XML 文档模式[13-17]和 XML Schema[18-19]的概念数据建模，例如在文献[13]中，UML 类图被用于 XML DTD(document type definition)的设计.

在现实世界应用中，信息通常是不精确和不确定的，模糊集是一种广泛使用的表示不精确和不确定信息的方法[20]. 基于关系数据模型的模糊关系数据库在文献中已被广泛研究[21-23]，而为了表示模糊对象、模糊复杂值属性以及复杂的模糊对象关系，当前对模糊数据建模的研究则主要集中在含不精确和不确定信息概念数据模型[24-25]和面向对象数据库[26]上. 在电子商务和供应链框架下，已有研究文章讨论信息的模糊性，并揭示了模糊集理论是实现 Web 基商务智能非常重要的技术手段[27-29]. 在人工智能及知识工程领域，模糊知识的表示与处理是该领域一个重要的研究课题，基于 XML 的模糊推理及决策支持系统可方便地实现 Web 应用中该类系统在不同开发者或用户之间的共享、交换、重用和集成[30]. 在语义 Web 中，为了表示和处理不精确和不确定数据与知识，其核心内容的本体及逻辑基础描述逻辑已被模糊扩展[31]，作为语义 Web 本体语言 OWL 建模手段的 XML，其语法形式需要模糊扩展. 除此之外，经由对各类数据源进行概要与评估信息查询和抽取形成的 XML 文档，其中可能含有大量的模糊信息. 可以看出，智能 Web 的构建以及基于 Web 的智能数据处理的实现和智能系统的开发需要 XML 具有模糊信息处理的能力，模糊 XML 数据模型在这些领域具有十分广泛的适用性.

应当指出的是，虽然 XML 已成为 Web 上数据表示与交换的标准，Web 技术的发展与现实应用也提出了在 XML 中管理不精确和不确定信息的需求，但是当前的 XML 还不能表示和处理不精确和不确定数据. 与数据库领域中含不精确和不确定信息数据库被广泛研究的情况不同，当前对表示和查询含不精确和不确定信息 XML 的研究还非常少，只有含不完全信息的 XML[32]和含概率信息的 XML[33-37]在研究文章有所讨论. 文献[38]基于 XML 开发了模糊面向对象建模技术，以表示需求规范和容纳 stereotype 概念，便于不精确需求的建模. 最新的研究文章[39]提出了使用 UML 类图设计模糊 XML 模型、使用关系数据库存储模糊 XML 的方法.

本文将识别 XML 中的多粒度数据模糊性，并基于可能性分布理论提出模糊 XML 数据模型. 本文将重点研究从模糊关系数据库到模糊 XML 模型的形式化转换方法，以实现基于模糊关系数据库的 XML 数据建模. 本文第 2 节介绍基于模糊集和可能性分布理论的模糊关系数据库相关基础知识；第 3 节讨论 XML 文档中的数据模糊性，给出模糊 XML 数据的表示模型；第 4 节给出从模糊关系数据库到模糊 XML 模型转换的形式化方法；第 5 节总结全文.

2 不精确和不确定信息及模糊关系数据库

在现实世界应用中，信息通常是不清楚和不明确的，因此不同种类的非完整(imperfect)信息已被广泛引入到数据库中[20]. 数据库系统中的不一致(inconsistent)信息、不精确(imprecise)信息、含糊信息(vague)、不确定(uncertain)信息和不明确(ambiguous)信息是 5 种基本的非完整(imperfect)信息[40-41]，它们各自的含义如下：

(1) 不一致信息代表一类语义传统，表明现实世界的某相同方面在一个数据库或多个不同数据库中被不一致地表示多次，比如张三的年龄被同时记录为 34 岁和 37 岁. 信息的不一致性通常来源于信息的集成[42].

(2) 信息的不精确性和含糊性与属性值的内容相关，表明要从一个给定的取值范围(可以是区间或者是集合)里选择一个值，但是当前不知道选择哪一

个，通常含糊信息表示成语言值. 例如张三的年龄为一个集合{18,19,20,21}，此为不精确信息，而李四的年龄为一个语言常量“年轻”，此为含糊信息.

(3) 信息的不确定性与属性取值的真值度相关，表明对一个或一组给定的值分配相信程度，例如李四当前的年龄是 35 的可能性是 98%. 有关用概率理论表示的随机不确定性不在本文的讨论范围.

(4) 信息的不明确性表示的是信息缺乏完全的语义，导致了多种可能的解释.

通常，一个信息可能同时存在几种类型的非完整性，例如张三的年龄为一个集合{18,19,20,21}，各个值的可能性分别是 70%，95%，98%和 85%. 信息的不精确性和不确定性是非完整性信息的两种主要形式，文献中已提出多种方法来表示不精确和不确定信息[20]，这些方法可归结为两大类，分别是符号方法和定量方法. 由 Zadeh 提出的模糊集理论[43]，就是一种被广泛使用的不精确和不确定信息的定量表示方法.

2.1　模糊集理论

设 U 是论域，F 是 U 上的一个模糊集. F 的定义需要一个隶属函数. $\mu_F: U \to [0,1]$，其中对于任意的 $u \in U$，$\mu_F(u)$ 表示 u 属于模糊集 F 的隶属度，模糊集 F 表示如下：

$$F = \{\mu_F(u_1)/u_1,\ \mu_F(u_2)/u_2, \cdots, \mu_F(u_n)/u_n\}.$$

当 U 不是离散域的时候，模糊集 F 则表示如下：

$$F = \int_{u \in U} \mu_F(u)/u.$$

要说明一点，在本文中 μ_F 用于表示模糊集 F 的隶属函数，而 $\mu_F(u)$ 用于表示 u 属于模糊集 F 的隶属度，并且只讨论离散模糊集. 实际上，上面的 $\mu_F(u)$ 也可以解释成一个变量 X 值为 u 的可能性度量，这里 X 取 U 中的值，此时一个模糊值可以用一个可能性分布 π_X 来表示[44].

$$\pi_X = \{\pi_X(u_1)/u_1, \pi_X(u_2)/u_2, \cdots, \pi_X(u_n)/u_n\}.$$

这里，对于任意的 $u_i \in U$，$\pi_X(u_i)$ 表示 u_i 为真的可能性. 一个模糊集是一个概念的表示，而可能性分布与分布内一个值出现的可能性相关联. 设 π_X 和 F 分别是一个模糊值可能性分布表示和模糊集表示，则 π_X 和 F 可看作是等同的，即 $\pi_X = F$[23].

借助于模糊集和可能性分布，U 上的一个模糊值可以用一个模糊集或一个可能性分布表示. 此外，信息的模糊性也可以借助于域元素中的类似(similarity)关系来表示[21]，此时，模糊性来自于论域中个体值之间的类似关系，而不是对象自身的状态. 类似关系用于描述同一论域中个体值间的类似度，一个模糊值表示成一个集合，该集合的元素是论域中的一些值，并且论域中存在类似关系. 类似关系具有自反性、对称性和传递性的特性，也有一些用于描述信息模糊性的关系，例如接近(proximity[45] 和 closeness[46])关系和近似(resemblance)关系[47]，它们只具有自反性和对称性的特性.

2.2　模糊关系数据库

由上面的讨论可知，模糊数据主要有 3 种表示形式，即模糊集表示、可能性分布表示和相似关系表示. 与 3 种表示形式相关联，已有多种在关系数据库中表示模糊数据的方法，导致了几种模糊关系数据库模型. 第 1 种模糊关系数据库模型基于模糊关系[23]和类似关系[21](或接近关系[45]和近似关系[47])，第 2 种模糊关系数据库模型基于可能性分布[22]，它又可以进一步分成 2 类：元组与可能性相关联；属性值由可能性分别表示. 与这 3 种模糊关系数据库模型相关联，它们的元组形式可分别表示如下：

$t = \langle p_1, p_2, \cdots, p_i, \cdots, p_n \rangle$

(对应第 1 种模糊关系数据库模型)，

$t = \langle a_1, a_2, \cdots, a_i, \cdots, a_n, d \rangle$

(对应第 2 种模糊关系数据库模型)，

$t = \langle \pi_{A1}, \pi_{A2}, \cdots, \pi_{Ai}, \cdots, \pi_{An} \rangle$

(对应第 3 种模糊关系数据库模型).

这里 $p_i \subseteq Dom(A_i)$，$Dom(A_i)$ 是属性 A_i 的值域，对于每个 $Dom(A_i)$，均有一个相似(或接近)关系. 并且 $a_i \in Dom(A_i)$，$d \in (0,1]$，π_{Ai} 是属性 A_i 在 $Dom(A_i)$ 上的可能性分布，对于 $x \in Dom(A_i)$，$\pi_{Ai}(x)$ 表示 x 是属性值 $t[A_i]$ 实际值的可能性.

基于上面提到的基本模糊关系数据库模型，还有几种扩展的模糊关系数据库模型. 很显然，可以把属性值是可能性分布、元组与隶属度相关联的 2 种基于可能性分布的模糊关系数据库结合到一起[48]，其它可能的扩展形式则是把可能性分布同类似关系(或接近、近似关系)结合到一起. 文献[46,49]中提出的基于扩展可能性分布的模糊关系数据库，其可能性分布和接近关系同时出现在数据库中. 本文讨论其元组具有下面形式的模糊关系数据库模型：

$$t = \langle \pi_{A1}, \pi_{A2}, \cdots, \pi_{Ai}, \cdots, \pi_{An}, d \rangle.$$

可以看出，在这种模糊关系数据库模型中，每个元组可能与一个可能度相关联，而属性值可以是由可能性分布表示的属性值. 这种模糊关系数据库模型的形式化定义如下.

定义 1.　关系模式 $R(A_1, A_2, \cdots, A_n, A_{n+1})$ 上的一个模糊关系实例 r 是笛卡尔积 $Dom(A_1) \times Dom(A_2) \times \cdots \times Dom(A_n) \times Dom(A_{n+1})$ 的一个子

集,其中 $Dom(A_i)(1\leq i\leq n)$ 可以是一个模糊子集或模糊子集的集合,$Dom(A_{n+1})$ 为 $(0,1]$. A_{n+1} 是特殊的属性,记作 PD.

应当指出的是,文献中根据模糊数据的表示形式已经提出多种模糊关系数据库模型,但是模糊关系数据库中模糊性只有两种形式,即属性值上的模糊性和元组上的模糊性.

3 模糊 XML 数据模型

3.1 XML 文档中的模糊性

模糊关系数据库中存在 2 类模糊性:一类与元组的成员度有关,另一类是用可能性分布表示的属性值.一个与元组相关联的成员度被解释成元组是相应关系实例成员的可能性,而表示属性值的可能性分布则意味着我们不知道属性的精确值,只知道该属性可能的取值范围以及每个可能值为真的可能性.作为数据结构化的 XML,它能够提供以自然方式表示不精确和不确定信息的手段,此时,成员度可与 XML 的元素相关联,而可能性分布也可与元素的值相关联.

现在来解释与 XML 元素相关联的成员度的含义,这里元素可以被元素嵌套,并且这些元素中的多个元素均可含有相关联的成员度.与一个元素相关联的成员度表明了现实世界状态中,包括该元素以及以该元素为根元素的子树的可能性.对于一个带子树的元素,子树中的每个节点不是独立的,是依赖于根到节点的链而存在的.源 XML 文档中的每个可能性是以相应父元素确切存在的事实为条件而指派的,换句话说,该可能性是一个基于其父元素存在的可能性精确为 1.0 假设的相对可能性.为了计算一个元素的绝对可能性,必须考虑其父元素的相对可能性.通常,一个元素的绝对可能性可以通过这样的方式得到:沿该元素到根元素路径上的所有相对可能性相乘,而所有这些相对可能性均可在源 XML 文档中得到.缺省状态下,相对可能性当作 1.0 处理.

考虑一个从根节点 A 到节点 C 的关系链 $A\rightarrow B\rightarrow C$,假设源 XML 文档中节点 C、B、A 相应的相对可能性分别为 $Poss(C|B)$、$Poss(B|A)$ 和 $Poss(A)$,则有

$$Poss(B)=Poss(B|A)\times Poss(A),$$

$$Poss(C)=Poss(C|B)\times Poss(B|A)\times Poss(A),$$

这里节点 C、B、A 的相对可能性 $Poss(C|B)$、$Poss(B|A)$、$Poss(A)$ 可以从源 XML 文档中直接得到.

现在考虑 XML 元素间由于复杂调用而出现环的时候如何计算带环元素的绝对可能性.XML 中的环可分为 2 种:XML 中由于元素被循环调用,换句话说由于子元素嵌套调用父元素而形成的环被称作嵌套环,而 XML 中由于元素被多个(2 个或 2 个以上)父元素非循环调用而形成的环被称作非嵌套环,本文只考虑非嵌套环的情况.当 XML 中存在非嵌套环时,从某一节点出发可通过多个不同的路径到达相同的节点.此时,带环元素沿不同的路径可计算得到多个可能不相同的绝对可能性,而其中的最大值将作为该带环元素的最终绝对可能性.

假设从根节点 A 到节点 D 有关系链 $A\rightarrow B\rightarrow D$ 和 $A\rightarrow C\rightarrow D$,很显然 D 为一个带非嵌套环的元素.设源 XML 文档中节点 D、B、A 相应的相对可能性分别为 $Poss(D|B)$、$Poss(B|A)$ 和 $Poss(A)$,节点 D、C、A 相应的相对可能性分别为 $Poss(D|C)$、$Poss(C|A)$、$Poss(A)$,则有

$$Poss(D)=\max(Poss(D|B)\times Poss(B|A)\times Poss(A),\ Poss(D|C)\times Poss(C|A)\times Poss(A)).$$

对于元素的值,XML 限定其为单值的,但是不难发现,这种限制并不总是正确的.通常的情况是,有些数据项有多个值,并且这些值可能是完全未知的,只能用可能性分布来说明.以一个人的 e-mail 地址为例,由于可能同时有几个 e-mail 地址,因此要用多个字符串表示.在不完全知道“Tom Smith”这个人 e-mail 地址的情况下,其 e-mail 地址为“TSmith@yahoo.com”的可能性是 0.60,为“Tom_Smith@yahoo.com”的可能性是 0.85,为“Tom_Smith@hotmail.com”的可能性是 0.85,为“TSmith@hotmail.com”的可能性是 0.55,为“TSmith@msn.com”的可能性是 0.45.与上述情况相反,有些数据项则只能取单值,例如一个人的年龄是一个单值非负的整数.如果一个人的年龄值当前未知,则可以用一个可能性分布{0.4/23,0.6/25,0.8/27,1.0/29,1.0/30,1.0/31,0.8/33,0.6/35,0.4/37}来表示.基于上述讨论可以看出,一个可能性分布表示的模糊数据有 2 种解释:模糊析取数据和模糊合取数据.

综上,XML 文档中有 2 类模糊性:

(1)第 1 类是元素的模糊性,可使用成员度与这样的元素关联;

(2)第 2 类是元素属性值的模糊性,用可能性分布表示这样的值.

要注意的是,对于后者又有 2 种类型的可能性分布,分别是析取可能性分布和合取可能性分布,它

们在祖先-后代链中可以出现在有或者没有自身孩子元素的子元素中.

图 1 给出了一个带模糊信息的 XML 文档片断[39].

```
1.〈universities〉
2. 〈university UName=“Oakland University”〉
3.  〈Val Poss=0.8〉
4.  〈department DName=“Computer Science and Engineering”〉
5.  〈employee FID=“85431095”〉
6.  〈Dist type=“disjunctive”〉
7.  〈Val Poss=0.8〉
8.  〈fname〉Frank Yager〈/name〉
9.  〈position〉Associate Professor〈/position〉
10.  〈office〉B1024〈/office〉
11.  〈course〉Advances in Database Systems〈/course〉
12.  〈/Val 〉
13.  〈Val Poss=0.6〉
14.  〈fname〉Frank Yager〈/name〉
15.  〈position〉Professor〈/position〉
16.  〈office〉B1024〈/office〉
17.  〈course〉Advances in Database Systems〈/course〉
18.  〈/Val 〉
19.  〈/Dist〉
20.  〈/employee〉
21.  〈student SID=“96421027”〉
22.  〈sname〉Tom Smith〈/name〉
23.  〈age〉
24.  〈Dist type=“disjunctive”〉
25.  〈Val Poss=0.4〉23〈/Val〉
26.  〈Val Poss=0.6〉25〈/Val〉
27.  〈Val Poss=0.8〉27〈/Val〉
28.  〈Val Poss=1.0〉29〈/Val〉
29.  〈Val Poss=1.0〉30〈/Val〉
30.  〈Val Poss=1.0〉31〈/Val〉
31.  〈Val Poss=0.8〉33〈/Val〉
32.  〈Val Poss=0.6〉35〈/Val〉
33.  〈Val Poss=0.4〉37〈/Val〉
34.  〈/Dist〉
35.  〈/age〉
36.  〈sex〉Male〈/sex〉
37.  〈email〉
38.  〈Dist type=“conjunctive”〉
39.  〈Val Poss=0.60〉TSmith@yahoo.com〈/Val〉
40.  〈Val Poss=0.85〉T.Smith@yahoo.com〈/Val〉
41.  〈Val Poss=0.85〉T.Smith@hotmail.com〈/Val〉
42.  〈Val Poss=0.55〉TSmith@hotmail.com〈/Val〉
43.  〈Val Poss=0.45〉TSmith@msn.com〈/Val〉
44.  〈/Dist〉
45.  〈/email〉
46.  〈/student〉
47.  〈/department 〉
48.  〈/Val〉
49. 〈/university〉
50. 〈university Uname=“Wayne State University”〉
51. 〈/university〉
52.  〈/universities〉
```

图 1　带模糊数据的 XML 文档片断

上面的例子描述的是一个国家指定城市所在地的大学情况，所讨论的是美国密歇根州底特律(Detroit)地区的大学.密歇根州立大学位于底特律市区内，它属于底特律地区大学的可能性是 1；奥克兰大学位于密歇根州近底特律市的奥克兰(Oakland)郡，对于奥克兰大学是否属于底特律地区大学，则取决于如何定义底特律地区，是只包括底特律市，还是指大底特律地区(the Greater Detroit Area).假设当前对底特律地区的确切定义未知，则奥克兰大学属于底特律地区大学的可能性将不为 1，例如指定为 0.8.此外，奥克兰大学计算机科学与工程系教师 Frank Yager 当前处于提职阶段，其职称可能是副教授或者是教授，他作为副教授讲授“Advances in Database Systems”课程、在办公室编号为 B1024 办公的可能性为 0.8，而他作为教授讲授“Advances in Database Systems”课程、在办公室编号为 B1024 办公的可能性则为 0.6.奥克兰大学计算机科学与工程系学生 Tom Smith 的“age”和“email”属性值是模糊的，分别用析取和合取可能性分布表示.

3.2　模糊 XML 的表示模型

通过上面的例子可以清楚地看出，一个取值为[0,1]的可能性属性“Poss”应当首先引入，它与一个称作“Val”的模糊构造子共同用于说明一个给定元素存在于 XML 文档的可能性.让我们看图 1 中的第 3 行，〈Val Poss=0.8〉表明给定的大学这个元素是“奥克兰大学”的可能性等于 0.8.对于可能性为 1.0 的元素，标签对〈Val Poss=1.0〉和〈/Val〉可以从 XML 文档中省略掉.

基于标签对〈Val Poss〉和〈/Val〉，元素的可能性分布可以表示出来，而可能性分布也可以用于表示模糊的元素值.为此，需要引入一个称作“Dist”的模糊构造子，以说明一个可能性分布.一个 Dist 元素典型地有多个 Val 元素作为孩子元素，每一个孩子元素带有一个相关联的可能度.由上面的讨论已知，存在 2 种类型的可能性分布，因此 Dist 构造子应当指明可能性分布的类型是析取的还是合取的.再看图 1 中的例子，第 24～34 行表示的是学生“Tom Smith”年龄的析取 Dist 构造子，第 38～44 行表示的是学生“Tom Smith”电子邮件地址的合取 Dist 构造子.要注意的是，第 24～34 行和第 38～44 行中的可能性分布在祖先-后代链中均在叶子节点上，而实际上非叶子节点上也可以有可能性分布.让我们来看图 1 中第 6～19 行的析取 Dist 构造子，它表示 ID 为 85431095 雇员的 2 个可能的状态，其中第 7～12 行的状态值带有 0.8 的可能性，第 13～18 行的状态值带有 0.6 的可能性.

为了实现模糊数据的 XML 建模，XML 文档必须进行扩展，扩展的结果是引入了若干个模糊构造子.很显然，为了容纳这些模糊构造子，XML 文档的模式应当做相应的修改.下面给出用于模糊 XML 数据建模的模糊 DTD 形式化描述.

首先，相对于经典 DTD，模糊 DTD 中增加了

2 个新元素,它们分别是 Val 元素和 Dist 元素.下面分别给出 Val 元素和 Dist 元素的形式化定义.

(1) Val 元素定义如下:

〈!ELEMENT Val (#PCDATA|original-definition)〉
〈!ATTLIST Val Poss CDATA "1.0"〉.

(2) Dist 元素定义如下:

〈!ELEMENT Dist (Val+)〉
〈!ATTLIST Dist type (disjunctive|conjunctive)
"disjunctive"〉.

其次,在模糊 DTD 中除了引入新的 Val 元素和 Dist 元素之外,还需要对经典 DTD 中的元素定义进行修改,从而使得这些元素在模糊 DTD 中可以使用可能性分布(Dist).模糊 DTD 中的普通元素(即非 Val 和 Dist 的其它元素)可分成叶子元素和非叶子元素,下面分别给出它们的形式化定义.

(1) 对于只含文本或 #PCDATA 的叶子元素(比如说 leafElement),它在模糊 DTD 中的定义从经典的

〈!ELEMENT leafElement (#PCDATA)〉

变成

〈!ELEMENT leafElement (#PCDATA|Dist)〉.

该定义的含义是,叶子元素存在 2 种情况,一种情况是叶子元素是精确的(例如图 1 中的 sname 和 student),此时上述定义就转变成如下经典 DTD 中普通叶子元素的定义

〈!ELEMENT leafElement (#PCDATA)〉.

另一种情况是叶子元素是模糊的,取可能性分布表示的值(例如图 1 中学生的 age),此时上述定义转变成如下的定义形式:

〈!ELEMENT leafElement (Dist)〉.

对于这样的模糊叶子元素,接下来需要分别使用 Dist 元素定义和 Val 元素定义进行进一步的定义,从而得到如下的定义形式:

〈!ELEMENT leafElement (Dist)〉
〈!ELEMENT Dist (Val+)〉
〈!ATTLIST Dist type (disjunctive|conjunctive) "disjunctive"〉
〈!ELEMENT Val (#PCDATA)〉
〈!ATTLIST Val Poss CDATA "1.0"〉.

(2) 对于非叶子元素(比如说 nonleafElement),它在模糊 DTD 中的定义首先从经典的

〈!ELEMENT nonleafElement (original-definition)〉

变成

〈!ELEMENT nonleafElement (original-definition|Val+|Dist)〉.

该定义的含义是,非叶子元素存在 3 种情况,一种情况是非叶子元素是精确的(例如图 1 中的 student),此时上述定义就转变成如下经典 DTD 中普通非叶子元素的定义

〈!ELEMENT nonleafElement (original-definition)〉.

另一种情况是非叶子元素是模糊的,并且元素取与可能度相关联的值(例如图 1 中的 university),此时上述定义转变成如下的定义形式:

〈!ELEMENT nonleafElement (Val+)〉.

对于这样的模糊非叶子元素,接下来需要使用 Val 元素定义进行进一步的定义,从而得到如下的定义形式:

〈!ELEMENT nonleafElement (Val+)〉
〈!ELEMENT Val (original-definition)〉
〈!ATTLIST Val Poss CDATA "1.0"〉.

最后一种情况是非叶子元素是模糊的,并且元素取值为集合,而集合中的每个值与一个可能度相关联(例如图 1 中的 employee),此时上述定义转变成如下的定义形式:

〈!ELEMENT nonleafElement (Dist)〉.

对于这样的模糊非叶子元素,接下来需要分别使用 Dist 元素定义和 Val 元素定义进行进一步的定义,从而得到如下的定义形式:

〈!ELEMENT nonleafElement (Dist)〉
〈!ELEMENT Dist (Val+)〉
〈!ATTLIST Dist type (disjunctive|conjunctive) "disjunctive"〉
〈!ELEMENT Val (original-definition)〉
〈!ATTLIST Val Poss CDATA "1.0"〉.

考虑图 1 中的模糊 XML 文档,通过使用上面给出的模糊 DTD 扩展形式,得到该模糊 XML 文档相应的模糊 DTD 定义,如图 2 所示.

```
〈!ELEMENT universities (university*)〉
〈!ELEMENT university (Val+)〉
  〈!ATTLIST university UName IDREF #REQUIRED〉
〈!ELEMENT Val (department*)〉
  〈!ATTLIST Val Poss CDATA "1.0"〉
〈!ELEMENT department (employee*, student*)〉
  〈!ATTLIST department DName IDREF #REQUIRED〉
〈!ELEMENT employee (Dist)〉
  〈!ATTLIST employee FID IDREF #REQUIRED〉
〈!ELEMENT Val (fname?, position?, office?, course?)〉
  〈!ATTLIST Val Poss CDATA "1.0"〉
〈!ELEMENT student (sname?, age?, sex?, email?)〉
  〈!ATTLIST student SID IDREF #REQUIRED〉
〈!ELEMENT fname (#PCDATA)〉
〈!ELEMENT position (#PCDATA)〉
〈!ELEMENT office (#PCDATA)〉
〈!ELEMENT course (#PCDATA)〉
〈!ELEMENT sname (#PCDATA)〉
〈!ELEMENT age (Dist)〉
〈!ELEMENT Dist (Val+)〉
  〈!ATTLIST Dist type (disjunctive)〉
〈!ELEMENT sex (#PCDATA)〉
〈!ELEMENT email (Dist)〉
〈!ELEMENT Dist (Val+)〉
  〈!ATTLIST Dist type (conjunctive)〉
〈!ELEMENT Val (#PCDATA)〉
  〈!ATTLIST Val Poss CDATA "1.0"〉
```

图 2 图 1 中模糊 XML 文档的 DTD

接下来，我们用例子来说明上面提出的模糊扩展 DTD 可用于表示带非嵌套环的模糊 XML 文档. 考虑的是一个表示某地区公司(company)与某类银行(bank)之间存款(deposit)和贷款(loan)业务关系的 XML DTD. 首先假设 XML 文档中不含有模糊信息，其 DTD 如图 3 所示. 不难看出，该 DTD 由于元素 bank 被 2 个父元素 deposit 和 loan 非循环调用，因而形成了非嵌套环.

```
〈!ELEMENT companies (company*)〉
〈!ELEMENT company (c-address?, deposit*, loan*)〉
  〈!ATTLIST company CName IDREF #REQUIRED〉
〈!ELEMENT c-address (#PCDATA)〉
〈!ELEMENT deposit (balance?, bank*)〉
  〈!ATTLIST deposit DID IDREF #REQUIRED〉
〈!ELEMENT balance (#PCDATA)〉
〈!ELEMENT loan (amount?, bank*)〉
  〈!ATTLIST loan LID IDREF #REQUIRED〉
〈!ELEMENT amount (#PCDATA)〉
〈!ELEMENT bank (b-address?, type?)〉
  〈!ATTLIST bank BName IDREF #REQUIRED〉
〈!ELEMENT b-address (#PCDATA)〉
〈!ELEMENT type (#PCDATA)〉
```

图 3　一个带非嵌套环的 DTD 例子

现在我们假设 XML 文档中含有模糊信息，模糊信息可能出现在：(1) 由于地区定义的不精确，不能完全确定一个公司个体是否属于指定的地区；(2) 由于类别定义的不精确，不能完全确定一个银行个体是否属于指定的类别；(3) 一个公司在一家银行的存、贷款额是模糊的(例如出于保密的原因分别表示为高和低). 为容纳这些模糊信息，通过使用上面给出的模糊扩展 DTD 形式，得到图 4 所示的带非嵌套环的模糊 DTD.

```
〈!ELEMENT companies (company*)〉
〈!ELEMENT company (Val+)〉
  〈!ATTLIST company CName IDREF #REQUIRED〉
〈!ELEMENT Val (c-address?, deposit*, loan*)〉
  〈!ATTLIST Val Poss CDATA "1.0"〉
〈!ELEMENT c-address (#PCDATA)〉
〈!ELEMENT deposit (balance?, bank*)〉
  〈!ATTLIST deposit DID IDREF #REQUIRED〉
〈!ELEMENT balance (Dist)〉
〈!ELEMENT Dist (Val+)〉
  〈!ATTLIST Dist type (disjunctive)〉
〈!ELEMENT loan (amount?, bank*)〉
  〈!ATTLIST loan LID IDREF #REQUIRED〉
〈!ELEMENT amount (Dist)〉
〈!ELEMENT Dist (Val+)〉
  〈!ATTLIST Dist type (disjunctive)〉
〈!ELEMENT bank (Val+)〉
  〈!ATTLIST bank BName IDREF #REQUIRED〉
〈!ELEMENT Val (b-address?, type?)〉
  〈!ATTLIST Val Poss CDATA "1.0"〉
〈!ELEMENT b-address (#PCDATA)〉
〈!ELEMENT type (#PCDATA)〉
```

图 4　对应图 3 的带非嵌套环的模糊 DTD

由图 4 可以看出，上面给出的模糊扩展 DTD 可用于表示带非嵌套环的模糊 XML 文档，换句话说为容纳模糊信息而对 DTD 进行的模糊扩展不受非嵌套环存在的影响，这一点与经典 XML 环境下非嵌套环的存在与表示不影响 DTD 语法形式是一致的. 作为对经典 DTD 的扩展，模糊扩展 DTD 在没有模糊信息存在时完全可以转化为经典的 DTD.

基于上面给出的 DTD 中各部分的模糊扩展描述，借鉴文献[50]中给出的 DTD 形式化描述方法，下面给出模糊扩展 DTD 的形式化定义.

定义 2(模糊 DTD).　一个模糊 DTD 是一对 (P,r)，其中 P 是一个元素类型定义的集合，r 是根元素类型用于指定某一特定的模糊 DTD. 每一个元素类型定义形如 $E\rightarrow(\alpha,A)$，其中 E 是指被定义的元素类型，α 被称为是元素 E 的内容模型(content model)，A 是一个属性表达式用于指定元素 E 所具有的属性. α 和 A 是通过下面的抽象语法所定义的表达式：

$$\alpha::=S\mid empty\mid(\alpha_1\mid\alpha_2)\mid(\alpha_1,\alpha_2)\mid\alpha?\mid\alpha^*\mid\alpha+\mid\text{any},$$

$$A::=(AN,AT,VT),$$

其中：

(1) S 是指基本数据类型 #PCDATA 或者元素类型 E，其中 E 包括元素类型 Val 和 Dist.

(2) $empty$ 表示空元素，“|”表示 union，“,”表示 concatenation，“?”表示 0 或 1 次，“*”表示 0 或多次，“+”表示 1 或多次，any 表示上述任意形式.

(3) AN 表示元素的属性名(Attribute Name)；AT 表示属性的类型(Attribute Types)，可以是 CDATA，ID，IDREF，IDREFS 和枚举类型$\{v_1\mid\cdots\mid v_n\}$；$VT$ 是属性值的类型(Value Types)，可以是 #REQUIRED，#IMPLIED，#FIXED “value”，“value”，或者是可能性分布 disjunctive 和 conjunctive.

4　从模糊关系数据库到模糊 XML DTD 的转换

为完成关系数据库到 XML 模式的转换，文献[8]定义了一种嵌套(nest)操作，该操作的结果是把关系数据库转化成非第一范式的嵌套关系数据库形式.

定义 3(nest).　设 r 是一个包含属性集 C 的 n 维关系表，进一步设 $A\in C$ 并且 $\overline{A}\in C-A$. 对于每一个 $(n-1)$ 元组 $t\in\prod_{\overline{A}}(r)$，定义一个 n 元组 t' 如下：

$$\begin{cases} t'[\overline{A}]=t \\ t'[A]=\{s[A]\mid s\in r\wedge s[\overline{A}]=t\} \end{cases},$$

那么，$nest_A(r)=\{t'\mid t\in \prod_{\overline{A}}(r)\}$.

执行 $nest_A(r)$操作以后，如果属性 A 只有一个含单一值的集合$\{v\}$，则嵌套失败，此时把$\{v\}$ 和 v 看作是可交换的(即$\{v\}=v$).因此当嵌套失败时，有 $nest_A(r)=r$ 成立.否则，如果属性 A 有一个含多个值的集合$\{v_1,\cdots,v_k\}(k\geq 2)$，则嵌套成功.

对于模糊关系数据库到 XML 的转换，分两步完成，首先是不考虑模糊信息的存在，完成关系数据库到 XML DTD 的转换，之后考虑关系表中可能含有的模糊信息，对得到的 DTD 树进行适当的修改.对于第一步转换，有下面的 12 条规则.

规则 1. 对于一个关系数据库模式，在相应的 XML 模式中生成一个根节点元素，其 DTD 描述为

〈!ELEMENT root (element*)〉.

规则 2. 对于每一个关系，对应生成一个 DTD 非叶子元素节点.

规则 3. 对于关系中的主键，直接用 DTD 中的属性声明，表示为

〈!ATTLIST Ename Aname ID …〉.

规则 4. 对于关系中的外键，如果关系的外键是对单个 ID 的参考使用，则用 DTD 中的属性声明加以描述，表示为

〈!ATTLIST Ename Aname IDREF …〉.

如果关系的外键是对多个 ID 的参考使用，则用 DTD 中的属性声明，表示为

〈!ATTLIST Ename Aname IDREFS …〉.

规则 5. 对于关系中非主键并且也是非外键的属性列，用 DTD 中的元素声明，表示为

〈!ELEMENT Ename (original-definition)〉.

规则 6. 对于关系中属性值不为空的限制，可以用 DTD 中的属性声明来加以限制，表示为

〈!ALLIST Ename Aname original-definition #REQUIRED〉.

规则 7[9]. 对于满足主键不包含外键以及主键包含一个以上外键的关系，直接在根元素下面生成其相应的 DTD 描述，表示为

〈!ELEMENT root (element*)〉.

规则 8[9]. 对于满足主键包含一个外键的关系 r_1，设其父关系为 r_2，此时直接把 r_1转换成 r_2的子元素，表示为

〈!ELEMENT element2 (element1*)〉.

规则 9[9]. 如果从关系 r_1到关系 r_2仅存在一个多对一的关系，此时 r_1到 r_2的外键不为空，则把 r_1直接转换作为 r_2的子元素，表示为

〈!ELEMENT element2 (element1*)〉.

规则 10[9]. 如果从 r_0到关系 $r_1,\cdots,r_k$存在多个多对一的关系情况，则把 r_0分别转换作为 $r_1,\cdots,r_k$的子元素，表示为

〈!ELEMENT element1 (element0*)〉

…

〈!ELEMENT elementk (element0*)〉.

规则 11. 如果从关系 r_1到关系 r_2存在一个多对多的关系，则把 r_1和 r_2转换为根元素的子元素，表示为

〈!ELEMENT root (element1*, element2*)〉,

之后在 element1 和 element2 中分别使用属性声明 ID 和 IDREF 加以描述.

规则 12. 如果关系 r_1存在一个多对多的关系，则把 r_1转换为根元素的子元素，表示为

〈!ELEMENT root (element1*)〉,

之后在 element1 中使用属性声明 ID 加以描述.

规则 13. 对于关系 r，经过 nest 运算后，最终表示成 $r(A_1,\cdots,A_{k-1},A_k,\cdots,A_n)$，其中属性$(A_1,\cdots,A_{k-1})$是嵌套结构.如果 $k=1$，即没有嵌套关系，则直接把属性列翻译成子元素，表示为

〈!ELEMENT Ename (original-definition)〉,

如果 $k>1$，即存在嵌套关系，则区分以下两种情况：

(1)对于每一个属性 $A_i(1\leq i\leq k-1)$，如果 A_i定义为可以为空，则可以把元素内容表示成 A_i^* 或是 A_i^+，即

〈!ELEMENT Ename (Aname1*, Aname2+,…)〉.

(2)对于每个属性 $A_j(k\leq j\leq n)$，如果 A_j定义为可以为空，则可以把元素内容表示成 A_i?或是 A_i[8]，即

〈!ELEMENT Ename (…, Aname1?, Aname2)〉.

规则 14. 对于关系中属性默认值的限制，可以用 DTD 中的属性声明来加以限制，表示为

〈!ALLIST element Aname original-definition “default”〉.

综合使用上述 14 条规则，可实现关系数据库到 XML DTD 的转换，转换的算法如下：

(1)不考虑表示模糊信息属性列，把关系中的其它属性作 nest 运算；

(2)应用规则 1，生成 DTD 树的根节点元素；

(3)应用规则 2 和规则 7，找到一个合适的关系生成根节点的子元素，为说明上的方便，假设这个合适的关系为 r_1；

(4)对于 r_1 中的属性列，按照规则 3、4、5、6、13 和 14 生成其相应形式；

(5)对于参照 r_1 的关系 r_2，…，r_n，应用规则 8、9、10、11 和 12 生成相应的 r_1 的子元素；

(6)递归遍历剩余未转换的关系，根据规则 8、9、10、11 和 12 找到其对应的父元素.

现在考虑关系中可能含有的模糊信息对生成的 DTD 的影响，此时需要在通过使用上述算法生成的 DTD 树的基础上做适当的修改. 当关系 r 中包含模糊信息时，有下面的规则.

规则 15. 对于含有 PD 属性的关系，转换过程如下：

(1)在关系 r 转换成的 DTD 子元素内部生成一个非叶子节点，即 Val 子元素节点，其元素出现的次数定义为＋.

〈!ELEMENT element (Val＋)〉.

(2)在 Val 子元素内部生成关系 r 的属性列内容，其处理方法跟不带模糊值的转换处理过程一致，同时声明 Val 默认值为 1.0.

〈!ELEMENT Val (element1*, elementk＋,
element1?, element1,…)〉
〈!ATTLIST Val Poss CDATA "1.0"〉.

规则 16. 对于属性值以可能性分布表示的关系，如果它最终生成的 DTD 文档是叶子节点子元素，则它的转换过程如下：

(1)在关系表 r 转换成的 DTD 子元素内部生成一个 Dist 子元素节点，得到

〈!ELEMENT element (Dist)〉.

(2)在 Dist 子元素内部生成 Val 子元素，其元素出现的次数定义为＋，形成

〈!ELEMENT Dist (Val＋)〉.

(3)在 Val 子元素内部生成关系表 r 的属性列内容，其处理方法跟不带模糊值的转换处理过程一致，但要同时声明 Val 默认值为 1.0，形成

〈!ELEMENT Val (original-definition)〉
〈!ATTLIST Val Poss CDATA "1.0"〉.

规则 17. 对于属性值以可能性分布表示的关系，如果它最终生成的 DTD 文档是非叶子节点子元素，则它的转换过程如下：

(1)在关系表 r 转换成的 DTD 子元素内部生成一个 Dist 子元素节点，

〈!ELEMENT element (Dist)〉.

(2)在 Dist 子元素内部生成 Val 子元素，其元素出现的次数定义为＋.

〈!ELEMENT Dist (Val＋)〉.

(3)在 Val 子元素内部生成关系表 r 的属性列内容，其处理方法跟不带模糊值的转换处理过程一致. 同时声明 Val 默认值为 1.0.

〈!ELEMENT Val (element1*, elementk＋,
element1?, element1,…)〉
〈!ATTLIST Val Poss CDATA "1.0"〉.

下面用例子来说明应用上面给出的转换规则，实现模糊关系数据库到模糊 XML DTD 转换的过程. 假设有表 1、表 2、表 3 和表 4 所示的 4 个关系，分别为 University、Department、Employee 和 Student，它们的主键分别为 Uname、Dname、EID 和 SID，并且 Uname 是关系 Department 的外键，Dname 同时是关系 Employee 和关系 Student 的外键. 4 个表中的主键用黑体表示，外键用斜体表示.

表 1　University

Uname	Address	PD
Oakland University	Detroit	0.8
Wayne State University	Detroit	1.0

表 2　Department

Dname	***Uname***	Location
Computer Science and Engineering	Oakland University	Oakland County

表 3　Employee

EID	***Dname***	Ename	Position	Office	PD
85431095	Computer Science and Engineering	Frank Yager	Associate Professor	B1024	0.8
	Computer Science and Engineering	Frank Yager	Professor	B1024	0.6

表 4　Student

SID	***Dname***	Sname	Sex	Age
20023056	Computer Science and Engineering	Tom Smith	Male	young

首先根据规则 1～14 生成一个不带模糊值的简单 DTD 描述. 在生成根节点元素以后，从关系表中找到一个合适的表作 root 的子元素，University 表无外键属性，可作为根节点的子元素，直接在根节点下翻译其 DTD(这里先不考虑带有模糊信息的 PD 属性列). 接着找到以 University 表中主键 *Uname* 为参照的 Department 表，在 university 元素下生成其相应的子元素，Employee 表，Student 表参照 Department表的*Dname*属性列，所以把表 Employee，Student 作为 Department 的子元素进行转换. 对于 Employee 表中带有模糊列信息的 PD 属性列同样先不考虑. 这样就得到了下面的 DTD 形式：

〈!ELEMENT root (university*)〉
〈!ELEMENT university (address+, department*)〉
〈!ATTLIST university Uname ID #REQUIRED〉
〈!ELEMENT department (location+,employee*,student*)〉
〈!ATTLIST department Dname ID #REQUIRED〉
〈!ATTLIST department Uname IDREF #REQUIRED〉
〈!ELEMENT employee (ename?, position?, office?)〉
〈!ATTLIST employee EID ID #REQUIRED〉
〈!ATTLIST employee Dname IDREF #REQUIRED〉
〈!ELEMENT student (sname?, sex?, age?)〉
〈!ATTLIST student SID ID #REQUIRED〉
〈!ATTLIST student Dname IDREF #REQUIRED〉
〈!ELEMENT address (#PCDATA)〉
〈!ELEMENT location (#PCDATA)〉
〈!ELEMENT ename (#PCDATA)〉
〈!ELEMENT position (#PCDATA)〉
〈!ELEMENT office (#PCDATA)〉
〈!ELEMENT sname (#PCDATA)〉
〈!ELEMENT sex (#PCDATA)〉
〈!ELEMENT age (#PCDATA)〉.

然后,根据规则 15～17 对上面没有考虑模糊信息存在生成的 DTD 进行修改.

(1) University 表中的 PD 是带有单一成员度值的属性.根据规则 15 进行修改:

〈!ELEMENT university (address+, Val+)〉
〈!ELEMENT Val (department*)〉
〈!ATTLIST Val Poss CDATA "1.0"〉.

(2) Student 表中的 age 属性是一个可能性分布,并且它是叶子节点子元素.根据规则 16 修改如下:

〈!ELEMENT age (Dist)〉
〈!ELEMENT Dist (Val+)〉
〈!ELEMENT Val (#PCDATA)〉
〈!ATTLIST Val Poss CDATA "1.0"〉.

(3) Employee 表中的 PD 是带有多个成员度值的属性,它是一个非叶子节点子元素.根据规则 17,进行修改:

〈!ELEMENT element (Dist)〉
〈!ELEMENT Dist (Val+)〉
〈!ELEMENT Val (ename?, position?, office?)〉
〈!ATTLIST Val Poss CDATA "1.0"〉.

经过修改,最终得到如下带有模糊信息的 DTD 形式:

〈!ELEMENT root (university*)〉
〈!ELEMENT university (address+, Val*)〉
〈!ATTLIST university Uname ID #REQUIRED〉
〈!ELEMENT Val (department*)〉
〈!ATTLIST Val Poss CDATA "1.0"〉
〈!ELEMENT department (location+,employee*,student*)〉
〈!ATTLIST department Dname ID #REQUIRED〉
〈!ATTLIST department Uname IDREF #REQUIRED〉
〈!ELEMENT employee (Dist)〉
〈!ATTLIST employee EID ID #REQUIRED〉
〈!ATTLIST employee Dname IDREF #REQUIRED〉
〈!ELEMENT Dist (Val+)〉
〈!ELEMENT Val (ename?, position?, office?)〉
〈!ATTLIST Val Poss CDATA "1.0"〉
〈!ELEMENT student (sname?, sex?, age?)〉
〈!ATTLIST student SID ID #REQUIRED〉
〈!ATTLIST student Dname IDREF #REQUIRED〉
〈!ELEMENT address (#PCDATA)〉
〈!ELEMENT location (#PCDATA)〉
〈!ELEMENT ename (#PCDATA)〉
〈!ELEMENT position (#PCDATA)〉
〈!ELEMENT office (#PCDATA)〉
〈!ELEMENT sname (#PCDATA)〉
〈!ELEMENT sex (#PCDATA)〉
〈!ELEMENT age (Dist)〉
〈!ELEMENT Dist (Val+)〉
〈!ELEMENT Val (#PCDATA)〉
〈!ATTLIST Val Poss CDATA "1.0"〉.

讨论. 模糊关系数据库到模糊 DTD 的转换经由两个阶段完成,首先是不考虑关系数据库中模糊信息的存在,通过使用规则 1～14 形成初始的 DTD,之后考虑元组级别及属性值级别上的模糊性,通过使用规则 15～17 对形成的初始 DTD 进行修正,最终得到模糊关系数据库对应的模糊 DTD.因此,完成模糊关系数据库到模糊 DTD 转换的规则的有效性和完备性,就由上述两个阶段各自的有效性和完备性所决定.对于第 1 个阶段:不考虑关系数据库中的模糊信息,规则 1～14 在转换时考虑了关系数据库中的关系、属性(主键、外键以及非主键和外键属性)以及关系之间的联系完整的内容,涵盖了这些内容的转换,并充分考虑了转换对关系数据库中主键、外键、空和非空等完整性约束的支持,以及对关系数据库中一对一、多对一和多对多联系的支持,这些转换的有效性和完备性在文献中(例如文献[7,9])已用实例进行了验证.至于第 2 个阶段,模糊关系数据库与经典关系数据库从模型的角度来看,其区别仅体现在结构和内容两个方面:(1)元组级别上存在模糊性,导致关系模式包含一个附加属性 PD;(2)属性值级别上存在模糊性,导致属性值以可能性分布表示.规则 15 和规则 17 给出了含 PD 属性模糊关系数据库到模糊 DTD 的转换方法,规

则 16 给出了含模糊属性值模糊关系数据库到模糊 DTD 的转换方法，因此第 2 个阶段的转换是完备的，而转换的结果符合模糊 DTD 规范，因而是有效的，上面给出的转换例子验证了这一点.

5　结束语

Web 的广泛使用已经产生了海量可用电子数据，基于 Web 的信息表示与交换因而变得十分重要. 当前 XML 已经成为 Web 信息表示与交换的标准，形成了有关 XML 的新的数据管理需求，例如 XML 文档的存储和查询等等. 另一方面，模糊集和可能性分布已广泛用于处理现实世界应用中信息的不精确性和不确定性，以智能数据处理为目地的模糊数据库建模正受到研究者越来越多的关注.

为了在 XML 中管理模糊关系数据库中的数据，本文研究了模糊 XML 数据建模. 基于可能性分布理论，本文首先识别出 XML 文档多粒度数据模糊性，提出了可包含这些模糊性的模糊 XML 数据模型，在此基础上给出了从模糊关系数据库到模糊 XML 模型的形式化转化方法. 应当指出的是，由于 XML DTD 是描述 XML 实例文档最常用的方法，因此本文对 XML 的模糊扩展只讨论了模糊 XML DTD. 但是 XML DTD 缺乏恰当描述高结构化数据的充足表达能力，而 XML Schema 为数据的描述提供了丰富的结构、类型和约束，因此未来研究工作之一将集中在基于 XML Schema 的模糊 XML 数据建模上. 另一个未来工作是深入研究带嵌套环 XML 模糊数据的表示与处理.

参　考　文　献

[1] Bray T, Paoli J, Sperberg-McQueen C M (Eds). Extensible Markup Language (XML) 1.0, W3C Recommendation. http://www.w3.org/TR/1998/REC-xml-19980210, 1998

[2] Bertino E, Catania B. Integrating XML and databases. IEEE Internet Computing, 2001, 5(4): 84-88

[3] Chung T S, Park S, Han S Y, Kim H J. Extracting object-oriented database schemas from XML DTDs using inheritance//Proceedings of the 2nd International Conference on Electronic Commerce and Web Technologies. Munich, Germany, 2001: 49-59

[4] Johansson T, Heggbrenna R. Importing XML schema into an object-oriented database mediator system [M.S. dissertation]. Computer Science, Uppsala University, Sweden, 2003

[5] Kanne C-C, Moerkotte G. Efficient storage of XML data//Proceedings of the 2000 International Conference on Data Engineering. California, USA, 2000: 198-198

[6] Kappel G, Kapsammer E, Rausch-Schott S, Retschitzegger W. X-Ray-towards integrating XML and relational database systems//Proceedings of the 19th International Conference on Conceptual Modeling (ER2000). Utah, USA, 2000: 339-353

[7] Fong J, Wong H K, Cheng Z. Converting relational database into XML documents with DOM. Information & Software Technology, 2003, 45(6): 335-355

[8] Lee D W, Mani M, Chiu F, Chu W W. Nesting-based relational-to-XML schema translation//Proceedings of the 4th International Workshop on the Web and Databases. California, USA, 2001: 61-66

[9] Liu C F, Vincent M W, Liu J X. Constraint preserving transformation from relational schema to XML schema. World Wide Web, 2006, 9(1): 93-110

[10] Runapongsa K, Patel J M. Storing and querying XML data in object-relational DBMSs//Proceedings of the 8th International Conference on Extending Database Technology Workshops (EDBTW2002). Prague, Czech Republic, 2002: 266-285

[11] Surjanto B, Ritter N, Loeser H. XML content management based on object-relational database technology//Proceedings of the 1st International Conference on Web Information Systems Engineering. Hong Kong, China, 2000: 70-79

[12] Du F, Amer-Yahia S, Freire J. ShreX: Managing XML documents in relational databases//Proceedings of the 2004 International Conference on Very Large Data Bases. Toronto, Canada, 2004: 1297-1300

[13] Conrad R, Scheffner D, Freytag J C. XML conceptual modeling using UML//Proceedings of the 19th International Conference on Conceptual Modeling (ER2000). Utah, USA, 2000: 558-571

[14] Elmasri R, Wu Y C, Hojabri B, Li C, Fu J. Conceptual modeling for customized XML schemas//Proceedings of the 21st International Conference on Conceptual Modeling (ER2002). Tampere, Finland, 2002: 429-443

[15] Mani M, Lee D W, Muntz R R. Semantic data modeling using XML schemas//Proceedings of the 20th International Conference on Conceptual Modeling (ER2001). Yokohama, Japan, 2001: 149-163

[16] Psaila G. ERX: A data model for collections of XML Documents//Proceedings of the 2000 ACM Symposium on Applied Computing. Como, Italy, 2000: 898-903

[17] Xiao R G, Dillon T S, Chang E, Feng L. Modeling and transformation of object-oriented conceptual models into XML schema//Proceedings of the 12th International Conference on Database and Expert Systems Aplications (DEXA2001). Munich, Germany, 2001: 795-804

[18] Bernauer M, Kappel G, Kramler G. Representing XML Schema in UML—A comparison of approaches//Proceedings of the 4th International Conference on Web Engineering. Munich, Germany, 2004: 440-444

[19] Routledge N, Bird L, Goodchild A. UML and XML schema//Proceedings of the 2002 Australasian Database Conference on Database Technologies. Melbourne, Australia, 2002

[20] Parsons S. Current approaches to handling imperfect information in data and knowledge bases. IEEE Transactions on Knowledge and Data Engineering, 1996, 8(2): 353-372

[21] Buckles B P, Petry F E. A fuzzy representation of data for relational databases. Fuzzy Sets and Systems, 1982, 7(3): 213-226

[22] Prade H, Testemale C. Generalizing database relational algebra for the treatment of incomplete or uncertain information and vague queries. Information Sciences, 1984, 34(2): 115-143

[23] Raju K V S V N, Majumdar A K. Fuzzy functional dependencies and lossless join decomposition of fuzzy relational database systems. ACM Transactions on Database Systems, 1988, 13(2): 129-166

[24] Ma Z M, Zhang W J, Ma W Y, Chen G Q. Conceptual design of fuzzy object-oriented databases using extended entity-relationship model. International Journal of Intelligent Systems, 2001, 16(6): 697-711

[25] Yazici A, Buckles B P, Petry F E. Handling complex and uncertain information in the ExIFO and NF2 data models. IEEE Transactions on Fuzzy Systems, 1999, 7(6): 659-676

[26] Ma Z M, Zhang W J, Ma W Y. Extending object-oriented databases for fuzzy information modeling. Information Systems, 2004, 29 (5): 421-435

[27] Petrovic D, Roy R, Petrovic R. Supply chain modeling using fuzzy sets. International Journal of Production Economics, 1999, 59(1-3): 443-453

[28] Yager R R, Pasi G. Product category description for Web-shopping in e-commerce. International Journal of Intelligent Systems, 2001, 16(8): 1009-1021

[29] Yager R R. Targeted e-commerce marketing using fuzzy intelligent agents. IEEE Intelligent Systems, 2000, 15(6): 42-45

[30] Tseng C, Khamisy W, Va T. Universal fuzzy system representation with XML. Computer Standards & Interfaces, 2005, 28(2): 218-230

[31] Stoilos G, Simou N, Stamou G, Kollias S. Uncertainty and the semantic Web. IEEE Intelligent Systems, 2006, 21(5): 84-87

[32] Abiteboul S, Segoufin L, Vianu V. Representing and querying XML with incomplete information//Proceedings of the 20th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems. California, USA, 2001: 150-161

[33] Abiteboul S, Senellart P. Querying and updating probabilistic information in XML//Proceedings of the 10th International Conference on Extending Database Technology. Munich, Germany, 2006: 1059-1068

[34] Hung E, Getoor L, Subrahmanian V S. PXML: A probabilistic semistructured data model and algebra//Proceedings of the 19th International Conference on Data Engineering. Bangalore, India, 2003: 467-478

[35] Nierman A, Jagadish H V. ProTDB: Probabilistic data in XML//Proceedings of the 28th International Conference on Very Large Data Bases. Hong Kong, China, 2002: 646-657

[36] van Keulen M, de Keijzer A, Alink W. A probabilistic XML approach to data integration//Proceedings of the International Conference on Data Engineering. Tokyo, Japan, 2005: 459-470

[37] Zhao W Z, Dekhtyar A, Goldsmith J. A framework for management of semistructured probabilistic data. Journal of Intelligent Information Systems, 2005, 25(3): 293-332

[38] Lee J, Fanjiang Y Y. Modeling imprecise requirements with XML. Information and Software Technology, 2003, 45(7): 445-460

[39] Ma Z M, Yan Li. Fuzzy XML data modeling with the UML and relational data models. Data and Knowledge Engineering, 2007, 63(3): 970-994

[40] Bosc P, Prade H. An introduction to fuzzy set and possibility theory based approaches to the treatment of uncertainty and imprecision in database management systems//Proceedings of the 2nd Workshop on Uncertainty Management in Information Systems: From Needs to Solutions. California, USA, 1993

[41] Smets P. Imperfect information: Imprecision-uncertainty. Uncertainty Management in Information Systems: From Needs to Solutions. Boston: Kluwer Academic Publishers, 1997: 225-254

[42] DeMichiel L G. Resolving database incompatibility: An approach to performing relational operations over mismatched domains. IEEE Transactions on Knowledge and Data Engineering, 1989, 1(4): 485-493

[43] Zadeh L A. Fuzzy sets. Information and Control, 1965, 8(3): 338-353

[44] Zadeh L A. Fuzzy sets as a basis for a theory of possibility. Fuzzy Sets and Systems, 1978, 1(1): 3-28

[45] Shenoi S, Melton A. Proximity relations in the fuzzy relational databases. Fuzzy Sets and Systems, 1989, 31(3): 285-296

[46] Chen G Q, Vandenbulcke J, Kerre E E. A general treatment of data redundancy in a fuzzy relational data model. Journal of the American Society of Information Science, 1992, 43 (4): 304-311

[47] Rundensteiner E A, Hawkes L W, Bandler W. On nearness measures in fuzzy relational data models. International Journal of Approximate Reasoning, 1989, 3(3): 267-298

[48] Umano M, Fukami S. Fuzzy relational algebra for possibility-distribution-fuzzy-relational model of fuzzy data. Journal of Intelligent Information Systems, 1994, 3(1): 7-27

[49] Ma Z M, Mili F. Handling fuzzy information in extended possibility-based fuzzy relational databases. International Journal of Intelligent Systems, 2002, 17(10): 925-942

[50] Calvanese D, Giacomo G D, Lenzerini M. Representing and reasoning on XML documents: A description logic approach. Journal of Logic and Computation, 1999, 9(3): 295-318

YAN Li, born in 1964, Ph. D., associate professor. Her research interests include intelligent data processing.

MA Zong-Min, born in 1965, Ph. D., professor, Ph. D. supervisor. His research interests include intelligent data and knowledge engineering.

LIU Jian, born in 1984, Ph. D. candidate. His research interests include databases and XML data management.

ZHANG Fu, born in 1983, Ph. D. candidate. His research interests include database modeling and description logics.

Background

Smooth conversion among models is one of the most fundamental features that a well-defined data model is expected to provide, which has received extensive discussions in the research of XML modeling. Recent efforts have shown that XML lacks sufficient power in modeling real-world data and their complex inter-relationships in semantics. Furthermore, XML is not able to represent imprecise and uncertain data which widely exist in human knowledge and natural language. Current efforts have been mainly made on the problems of representing incomplete, probabilistic data rather than fuzzy data information. In order to establish a firm foundation for publishing and managing the histories of fuzzy data on the Web, in this paper, we identify multiple granularity of data fuzziness in XML and extend the ability of XML that represents imprecise and uncertain data without changing the current XML standard. Based on possibility distribution theory, we further develop the fuzzy XML data model. We finally present a general framework for the smooth conversions from the fuzzy relational databases to the fuzzy XML model.

This work is supported by the National Science Foundation of China (60873010), the Program for New Century Excellent Talents in University (NCET- 05-0288) and the Fundamental Research Funds for the Central Universities (N090504005, N100604017 and N090604012), and in part by the National Science Foundation of China (61073139).