関数従属性と包含従属性を用いたXML-RDBマッピング手法に関する研究*

太田壮祐 (学籍番号 201021737)
研究指導教員: 森嶋厚行
副研究指導教員: 鈴木伸崇

1. はじめに

これまで，XMLデータをRDBで管理するために，XMLデータをRDBにマップする手法が数多く研究されてきた [1]．これらの既存の手法は全て異なるアプローチを採用しているが，XML要素 (あるいは属性) からRDB属性値への1:1もしくは1:Nマッピングを行うという共通点が存在する．

しかし1:1/1:Nマッピングでは，生成されたリレーションを更新する際，XMLビュー上でのデータ一貫性制約の維持が保証されないという問題がある．これは，1:1/1:Nマッピングでは，一貫性制約が存在するXML上のデータが別々のRDB属性値にマップされてしまうため生じる問題である．

本論文では，XML-RDBマッピングの中で一貫性制約の維持を可能とするために，マッピング手法 **C-Mapping** (Consistency-conscious Mapping) を提案する．C-Mappingは，入力としてXMLデータとそのXMLデータに存在する一貫性制約 (関数従属性と包含従属性) を指定する事により，RDBへのマッピング結果としてリレーションの集合を出力する．C-Mappingのユニークな特徴は，1:1/1:Nマッピングだけでなく N:1 マッピングも実現可能である，という意味で**完全**であるという事である．加えて，C-Mappingは入力として適切な一貫性制約を与える事により，既存の主要なマッピング手法の多くをシミュレートできる．

ここで，例を用いて N:1 マッピングを説明する．N:1 マッピングとは，複数の XML 要素 (あるいは属性) を 1 つの RDB 属性値にマップすることである．N:1 マッピングの例を図 1 に示す．この例では，book.xml が「noveltitle 要素に含まれるテキストは，必ず booktitle 要素のテキストとして存在する」という一貫性制約を持つと仮定する．図 1 に見られるように，各 noveltitle 要素とそれに対応する booktitle 要素は，book リレーションの中で同じ属性値にマップされている．このように N:1 マッピングを行う事で，XML データの更新時における一貫性維持が容易になる．

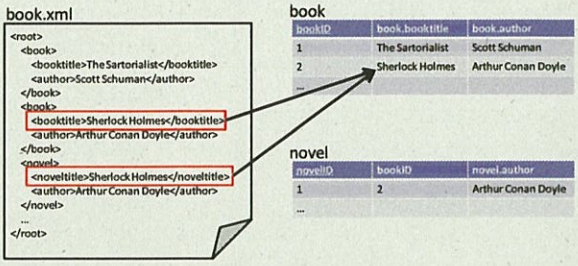

図 1　N:1 マッピングの例

2. XML の一貫性制約

2.1 XFD (XML Functional Dependencies)

XFD は論文 [2] で定義された XML における関数従属性である．具体例として，図 1 に示す book.xml における XFD の例を説明する．book.xml には，各 book 要素が決まれば，その book 要素に含まれる booktitle 要素のテキストが決まる，という制約が存在すると仮定する．その制約は次の XFD で表す事が出来る．

```
for $x in book.xml/root/book
    $x →$x/booktitle/text()
```

また，本論文では，XFD の決定子と被決定子に計算で求められる仮想的な属性の使用を許可した XFD+ と呼ばれる制約のクラスを導入する．

2.2 XIND (XML Inclusion Dependencies)

XIND は，XML における包含従属性である．具体例として，図 1 に示す book.xml における XIND の例を説明する．book.xml には，noveltitle 要素の持つ全てのテキストは booktitle 要素のテキストとして存在していなければならない，という制約が存在すると仮定する．その制約は次の XIND で表す事が出来る．

```
book.xml/root/book/booktitle/text()
 ⊇ book.xml/root/novel/noveltitle/text()
```

3. C-Mapping

C-Mapping の入出力を図 2 に示す．入力は，XML データの集合 X，X に関する XFD+ の集合 $XFDSet$，そして X に関する XIND の集合 $XINDSet$ である．出力は，X のマッピング結果であるリレーションの集合 R と，XML データの集合 X' もしくは復元時に使用するための Viewtree の集合 V である．

C-Mapping は次の手順で処理を行う．(1)

---

* "A Study on XML-RDB Mapping Methods using Functional and Inclusion Dependencies" by Sosuke OTA

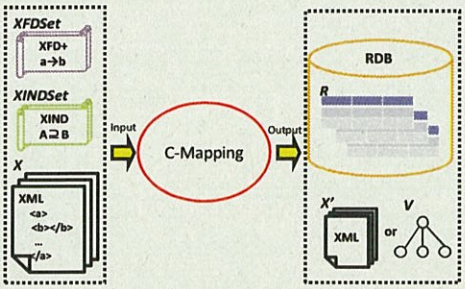

図 2　C-Mapping の入出力

$XFDSet$ と $XINDSet$ を用いて，リレーショナルスキーマ RS を生成する．(2) X を用いて，RS に従うリレーションインスタンスの集合 I と，X' もしくは V を生成する．

3.1　C-Mapping によるスキーマ生成

(1) はさらに次の 2 つのステップに分けられる．

ステップ 1．ステップ 1 では $XFDSet$ を用いて，リレーショナルスキーマ RS' を生成する．基本的には，$XFDSet$ の各 XFD+ である xfd_i に対して，xfd_i の各関数従属性の決定子に対応する主キー属性と，被決定子に対応する属性を持つリレーションスキーマを生成し RS' に追加する．

ステップ 2．ステップ 2 では，SQL データベースで一般にサポートされている**外部キー制約**を有効に利用して，$XINDSet$ をマッピング後も維持できるように RS' を RS に変形する．

外部キー制約は，参照される属性を主キー属性とする，包含従属性の一種である．例えば，$R(\underline{K_R}, \ldots, A, \ldots)$ と $S(\underline{K_S}, \ldots, B, \ldots)$ という 2 つのリレーションスキーマがある時，外部キー制約 $R[K_R] \supseteq S[B]$ とは，$S[B]$ の値は必ず $R[K_R]$ にも存在するという包含従属性を表す．一般的な RDBMS は，指定された外部キー制約をリレーションが満たすようチェックする機能を持っている．

しかし，外部キー制約には，参照される属性がキー属性でなければならないという制限が存在する．したがって，$R[A] \supseteq S[B]$ のように参照される属性 A が R におけるキー属性ではない一般の包含従属性は，SQL データベースではサポートされていない．入力 XIND“$e_1 \supseteq e_2$” に対応する，ステップ 1 のマッピング結果上の包含従属性 $U[A_{e_1}] \supseteq V[A_{e_2}]$ において，A_{e_1} が必ずしも U におけるキー属性になるとは限らない．したがって，単純な変換方法では，SQL データベースにおける外部キー制約で実装することはできない．

この問題に対処するために，ステップ 2 では，入力 XIND に対応する一般の包含従属性 $R[A] \supseteq S[B]$ を外部キー制約 $R[K_R] \supseteq S[K'_R]$ に置き換える手法を導入する．ここで，K'_R は S に追加された，K_R と同等の属性である．

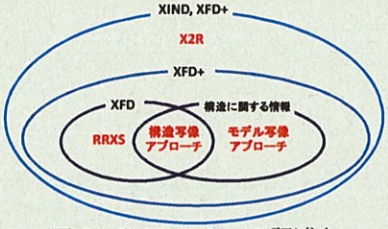

図 3　C-Mapping の記述力

元の制約のセマンティクスを保持するために，S に追加された K'_R の値は次の条件を満たすように選ぶ必要がある．すなわち，リレーション $T = R \bowtie_{K_R = K'_R} S$ において $\forall t \in T(t[A] = t[B])$ が成立する事である．この条件を満たす時，R において K_R がキー属性である事から $K_R \rightarrow A$ であるため，S において $K'_R \rightarrow B$ である．したがって，$R[K_R] \supseteq S[K'_R]$ ならば $R[A] \supseteq S[B]$ を満たす．

したがって，ステップ 2 では次の 2 つの処理を行う．(1) R のキー属性 K_R と同等の属性 K'_R を S に追加する．(2) S から属性 B を除去する．

4.　C-Mapping の記述力

C-Mapping の記述力の関係をまとめたものを図 3 に示す．図 3 では，それぞれ円で囲ったものが，入力として与える制約のクラスにより C-Mapping がシミュレート可能な既存手法を表している．

5.　実データを用いた評価

理論的に示した記述力の違いが，実データを用いた場合にどのように影響を及ぼすのかを調査した．具体的には，C-Mapping と既存手法をそれぞれ用いて，Wikipedia のページから Wikipedia のバックエンド DB のリレーションを生成できるかを調査した．その結果，C-Mapping では 82% の RDB 属性へのマッピングが可能であったのに対し，既存手法では 43% であった．このように，C-Mapping の持つ高い記述力が実用上も重要だという事がわかった．

6.　まとめ

本論文では，関数従属性と包含従属性を用いた XML-RDB マッピング手法 C-Mapping の提案を行った．C-Mapping は次の特徴を持つ．(1) 1:1/1:N マッピングだけでなく，N:1 マッピングも実現できるという意味で完全なマッピング手法である．(2) 既存の XML-RDB マッピング手法の多くを十分にシミュレート可能な記述力を持つ．

文献

[1] 天笠俊之, 吉川正俊: XML データベース技術概説. オペレーションズ・リサーチ: 経営の科学 50(6): 365-372 (2005).
[2] Yi Chen, Susan Davidson, Carmem Hara, Yifeng Zheng: RRXS: Redundancy reducing XML storage in relations, the 29th VLDB Conference: 189-200 (2003).