データベースシステム　122－19
(2000. 7. 27)

異種文書管理のための問合せ機構の考察

横田一正　　國島丈生　　劉渤江

岡山県立大学情報工学部　　岡山理科大学総合情報学部
{yokota,kunishi}@c.oka-pu.ac.jp　　liu@mis.ous.ac.jp
〒719-1197 総社市窪木 111　　〒700-0005 岡山市理大町 1-1

XML は電子化文書の交換フォーマットとしてデファクトスタンダードの地位を獲得しつつあるが、複合オブジェクトなどのデータモデル、共有化情報と個別化情報の表現の要件、マルチメディア情報の統合、などの観点から考えると、大きな制約を持っている。

われわれは現在、異種文書管理、デジタルテーマパーク、マルチメディア情報提示の制御などの研究開発を行っているが、そこで必要とされる要件を反映するために、本稿では、XML に特殊な属性を導入することによって拡張することを提案する。これによって従来のデータモデルの構成子を使うことや意味的なエレメントの定義も可能になる。それに対応して問合せの拡張も必要になるので、巡行を含む宣言的な問合せ機構を議論する。

キーワード: XML、グラフモデル、データモデル、文書管理

Considerations on Query Processing for Heterogeneous Document Management

Kazumasa Yokota　　Takeo Kunishima　　Bojiang Liu

Okayama Prefectural University
Faculty of Conputer Science and System Engineering
{yokota,kunishi}@c.oka-pu.ac.jp
Soja, Okayama 719-1197

Okayama University of Science
Faculty of Informatics
liu@mis.ous.ac.jp
Ridai-cho, Okayama 700-0005

XML has been almost *de facto* standard for data exchange of electronic documents, however it has many restrictions to represent various information from viewpoints of constructing conventional data models such as complex objects, combining personal and shared information, and mediating multimedia information.

We have been developing XML-based systems for heterogeneous document management, digital theme parks, and interactive multimedia presentation, where there are many enhanced requests to XML. In this paper, we propose an extension of XML by introducing some special attributes, by which conventional constructors can be utilized and sementic elements can be defined. We discuss its declarative query processing including navigation.

Key words: XML, graph model, data model, document management

1 はじめに

XML は電子化文書の交換フォーマットとしてデファクトスタンダードの地位を獲得しつつあるが、複合オブジェクトなどのデータモデル、共有化情報と個別化情報の共存化、マルチメディア情報の統合、などの観点から考えると、構文的に大きな制約を持っており、その緩和が重要である。

われわれは現在、異種文書管理 [4, 5]、デジタルテーマパーク [6]、マルチメディア情報提示の制御 [3]、文学データベース [8] などの研究開発を行っているが、そこで必要とされる要件を反映するために XML を拡張する必要が生じた。これまでは QUIK[7] を拡張することを検討していた [4] が、もう少し一般性を持たせるために、本稿では、XML に特殊な属性を導入することによって拡張することを提案する。これによってより自然に従来のデータモデルの構成子を使うことや意味的に拡張されたエレメントを定義することが可能になり、さらにそれに対応して問合せ機構を宣言的に定義することができる。

まず2節では、ここで使用するグラフモデルと、いくつかの応用からの要件を議論する。3節では上の要件を反映するためのデータ構造の拡張を考え、4節ではそれに対応するデータ操作を議論する。

2 モデルへの要件

2.1 XML を考慮したグラフの枠組

XML は一般的にグラフモデルとして定義される (たとえば文献 [1])。本稿ではグラフモデルを以下のように定義する。

$$
\begin{aligned}
\langle graph\rangle \quad ::= \quad & (\langle id\rangle,\\
& \langle e\text{-}label\rangle,\\
& \{(\langle a\text{-}label\rangle, \langle a\text{-}value\rangle), \cdots\},\\
& \{\langle e\text{-}graph\rangle, \cdots\})\\
\langle e\text{-}graph\rangle \quad ::= \quad & \langle graph\rangle \mid \langle a\text{-}value\rangle \mid \langle id\rangle
\end{aligned}
$$

各グラフの $\langle graph\rangle$ は変数で等値制約の役割を持っている。同一 $\langle id\rangle$ のグラフは識別子の性質からマージされなければならないが、その部分グラフのマージで、異なった $\langle id\rangle$ はマージできないが、$\langle graph\rangle$ は異なっていてもマージ可能な点が異なっている。属性が2種類あるのは、XML を前提にしたもので、それぞれを属性 ($\langle a\text{-}attr\rangle$ と略) とエレメント属性 ($\langle e\text{-}attr\rangle$ と略) と呼ぶ。$\langle a\text{-}value\rangle$ は原子値である。$\{(\langle a\text{-}label\rangle, \langle a\text{-}value\rangle), \cdots\}$ を $\{\langle a\text{-}label\rangle = \langle a\text{-}value\rangle, \cdots\}$ と略記する。エレメント属性はグラフの集まりとして定義されるが、それぞれを部分エレメントと呼ぶ。

この構文は QUIXOTE のグラフモデル [10] に準じており、グラフ自体の意味論も同様に定義できるが、本稿ではページ制限のため議論しない。本稿では、応用からの要件を反映させるために、XML を意識した変種を議論する。

たとえば図1の XML の簡単な例を考えよう。これはグラフとしては図2のように表現できる。この中で "_" は匿名変数を示している。さらにこれをグラフとして図示すれば図3と

```
⟨book-order⟩
  ⟨customer⟩Okamoto⟨/customer⟩
  ⟨shop location="Soja"⟩ABC⟨/shop⟩
  ⟨goods⟩
    ⟨book⟩
      ⟨publish year="1999"/⟩
      ⟨name⟩Kurekure⟨/name⟩
    ⟨/book⟩
    ⟨book⟩
      ⟨publish year="2000"/⟩
      ⟨name⟩Dierdre⟨/name⟩
    ⟨/book⟩
  ⟨/goods⟩
⟨/book-order⟩
```

図 1: XML の例

$$
\begin{aligned}
g_0 &= (\_, \mathit{book\text{-}order}, \_, \{g_1, g_2, g_3\})\\
g_1 &= (\_, \mathit{customer}, \_, \{\text{Okamoto}\})\\
g_2 &= (\_, \mathit{shop}, \{\mathit{locator} = \text{“}\mathit{Soja}\text{”}\}, \{\text{ABC}\})\\
g_3 &= (\_, \mathit{goods}, \_, \{g_{31}, g_{32}\})\\
g_{31} &= (\_, \mathit{book}, \_, \{g_{311}, g_{312}\})\\
g_{311} &= (\_, \mathit{publish}, \{\mathit{year} = \text{“1999”}\}, \_)\\
g_{312} &= (\_, \mathit{name}, \_, \{\text{Kurekure}\})\\
g_{32} &= (\_, \mathit{book}, \_, \{g_{321}, g_{322}\})\\
g_{321} &= (\_, \mathit{publish}, \{\mathit{year} = \text{“2000”}\}, \_)\\
g_{322} &= (\_, \mathit{name}, \_, \{\text{Dierdre}\})
\end{aligned}
$$

図 2: XML のグラフ表現

なる。この構文的な定義は冗長に見えるかもしれないが、上記の例で $\langle goods\rangle$ のエレメント属性の構成子の性質を規定するには都合がいい。また文献 [1] では半構造データの表現と

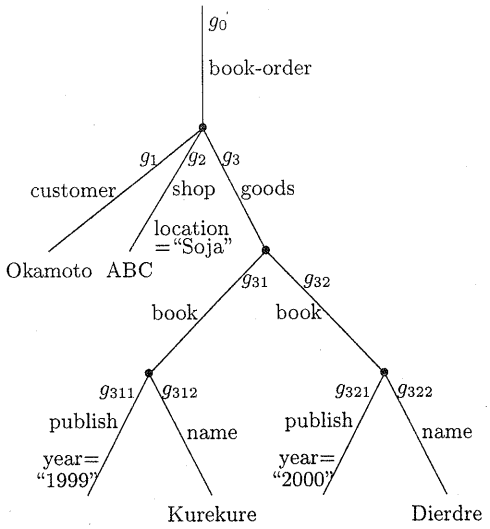

図 3: XML のグラフ表現 (2)

して ssd-expression を述べているが、本稿のモデルはこれも包含している。以下この形式にしたがい、具体的な意味を検討したい。

2.2 XML の特徴

XML の特徴には以下のものがある。

- 属性 ($\langle a\text{-}attr \rangle$)
  この特徴は、同一ラベルは一度しか現れず、属性の順序に意味はなく、属性値は原子値であることである。
- 部分エレメント ($\langle e\text{-}attr \rangle$)
  この特徴は、同一ラベルは複数あってもよいが、エレメントの順序は意味を持ち、値は意味的には集合値ではないことである。また、部分エレメントでない PCDATA が混在することがあるが、この値はエレメントの順序で、深さ優先で結合する必要がある
- 識別子
  XML は id 属性と idref 属性で参照関係を定義できるが、識別子としての意味は弱く、オブジェクト識別子のような対称的な参照関係は定義できない。

XML では意味論の定義はなく、設計者が属性として付加可能である。DTD でもタグの意味を付加できるが、これはデータモデルの意味論とは異なっており、構文的な制約である。本稿で議論するのは、たとえば、oid や等値制約の導入、エレメントに対する集合属性などの構成子の付加、PCDATA へのダミータグの挿入などである。これらは、応用の要件を考慮しながら、XML の拡張として付加することが可能である。

2.3 半構造データからの要件

半構造データは、XML データを含む幅広い概念であり、どこまでを対象にするかで要件も異なってくる [1]。ここでは以下を考慮の対象とする。

- 集合概念
  属性 (属性 - 値対) の集まりとしてレコードを記述する際、属性の順序に意味はない。したがって集合概念を導入する。
- 自己記述
  データの意味はスキーマという形では必ずしも記述できないので、属性中に自己記述させる。本稿での属性での意味記述はこれに対応している。
- 識別子と制約
  OEM[1] のようにすべての要素にオブジェクト識別子を付加することによって、データ間のリンクは明確に記述できるが、半構造データは情報の部分性が強く、必ずしも静的な識別子は好ましくないことがある。したがってそのために等値制約を導入する。

一般的に半構造データでは、XML より柔軟な構造が求められ、そのために XML の構文的制約をいかに緩和するかが重要となる。

2.4 デジタルテーマパークからの要件

デジタルテーマパーク [6] では、視点によって対象 (観光ポイント) の粒度を変更する必要があり、個々の対象間には必ずしも静的なリンクは張られていない。

- 観光ポイント
  対象はグラフのノードに対応するが、視点によって粒度を変更するために、ハイパーグラフ構造がふさわしい。観光ポイント間のコース自体が観光ポイントにもなりえて、それがノードと同様の (エレメント) 属性をもたねばならない。したがってコースとリンク (エッジ) の関係が議論となる。

- 観光コース
  観光ポイント間には複数の推奨コースがあり、さらに利用者の要求に応じて経路探索を行い、動的にコースが生成できなければならない。コースをリンクと考えれば多重グラフとなる。
- 制約
  観光ポイント間を辿るためには、空間的、時間的、費用などの制約を考慮する必要がある。したがってそれら制約の充足判定機構が必要となる。
- マルチメディア情報
  各観光ポイントには多種のマルチメディア情報が付加されているが、提示のためにそれらの間に関連を持たさせければならない。

デジタルテーマパークでは観光ポイントの提示に特有の提示方式が必要とされ、その機能の埋め込みが必要となる。そのためにはマルチメディア情報の同期制御 (たとえば文献 [3]) の組込みも検討が必要である。

2.5 異種文書管理の要件

電子化された文書管理 [4, 5] のためには、データ形式、版、設計、所在、共有性などさまざまな異種性を考慮しなければならない。そのためにとくに以下を重要視している。

- 設計の異種性
  内容的には同一ながらも設計によって物理的に異なった文書に分割されたものを論理的に同一に扱う。つまり物理的な構成に依存しない検索結果を保証する。
- 論理的エレメント
  電子化文書に対する利用者のメモのように、XML では必ずしも定義できない対象をエレメントとして定義する。このためにはエレメントの構文と意味を分離する必要がある。
- リンク
  関連文書をリンクし再利用や保守に利用することは他の応用以上に多く、従来手続き的にしか考えられてこなかった巡行を含む検索機能の明確な定義が必要である。
- 共有レベル
  上記メモにあるように、個別化、グループ化、共有化のような情報の共有化レベルの階層化が必要となる。

この応用では内容や利用者の視点からの識別や取扱のために、XML の構文と意味の分離が必要である。

3 形式化へのアプローチ

前節でのさまざまな要件を踏まえた上で、XML の意味的に拡張されたグラフモデルを検討する。

3.1 提案モデル

ここで提案するモデルは 2.1節のものであり、参照構造をもった (ハイパー) グラフ構造となっている。

複合オブジェクトやオブジェクトモデルと異なっているのは以下の点である。

- 参照関係としては識別子 $\langle id \rangle$ と等値制約 $\langle graph \rangle$ の 2 種類をもち、必要に応じて 2 種類の情報統合を行えるようにしている。
- 2 種類の属性からなっており、$\langle a\text{-}attr \rangle$ は組構成子、$\langle e\text{-}attr \rangle$ の属性は、$\langle a\text{-}attr \rangle$ の指定により、集合 (組)、リスト構成子となりうる。

集合と組は部分エレメントの形の違いによって区別する。

3.2 特殊な属性の導入

XML を意味的に拡張するために、属性として以下の特殊な属性を導入し、エレメント属性を性質を拡張する (下線が既定値)。

- $constructor$ = "$\underline{set}$" | "$list$": エレメント属性の構成子の指定
- doc_group = "yes" | "$\underline{no}$": 文書群の指定
- $user$ = "xxx": 利用者 "xxx" の個別情報
- $course$ = "$\underline{both}$" | "one": $\langle a\text{-}attr \rangle$ 中に指定されるアーク $(from, to)$ の方向性

$constructor$ (以下 $const$ と略) については以下のように定義される。

	エレメントの順序	ラベルの重複
set	なし	許さず
$list$	あり	許す

エレメントが集合値となっているので set の上の定義は制限になっていないことに注意されたい。doc_group は、部分エレメントの集合が文書群 [5] として論理的にひとつとして扱うかどうかを指定している。$user$ はここで指定された情報の所有者を示している。$course$ は観光コースの方向性を示すための特殊なラベルである。

3.3 PCDATA とテキストの結合

検索等のデータ操作を行う場合、対象とするテキストデータを明確にする必要がある。

エレメント属性の統一的記述のための特殊なタグ $\langle pc\_data \rangle$ を用いる。

$$\cdots\langle a\rangle T_1\langle b\rangle T_2\langle /b\rangle T_3\langle /a\rangle\cdots$$

というデータ (T_1 と T_3 がPCDATA) に対しては以下のように書く。

$$\cdots\langle a\rangle\langle pc\_data\rangle T_1\langle /pc\_data\rangle \langle b\rangle T_2\langle /b\rangle\langle pc\_data\rangle T_3\langle /pc\_data\rangle\langle /a\rangle\cdots$$

これによってPCDATAと部分エレメントの混在を防ぐ。

このテキストの扱いはエレメントの属性 *const* によって異なっている。テキストの抽出と文字列接合関数をそれぞれ $text, cat$ とすると

$$text((\_, pc\_data, \_, \{T\}) = T$$
$$text((\_, pc\_data, \{const = \text{“}list\text{”}\}, \{a_1, a_2, \cdots, a_n\}) = \{cat([text(a_1), text(a_2), \cdots, text(a_n)])\}$$
$$text((\_, pc\_data, \{const = \text{“}set\text{”}\}, \{a_1, a_2, \cdots, a_n\}) = \{text(a_1), text(a_2), \cdots, text(a_n)\}$$

と定義する。つまり “*list*” の場合テキストは深さ優先で結合される。検索やデータ抽出の対象にはこの *text* の結果が使われる。集合指定が入れ子になった場合、結果の *text* も入れ子になる。

3.4 論理エレメント

論理エレメントは

$$\cdots\langle a\rangle T_1\langle /a\rangle\langle b\rangle T_2\langle /b\rangle\cdots$$

という記述に対して

$$\cdots\langle a\rangle T_{11}\underline{\langle c\rangle T_{12}\langle /a\rangle\langle b\rangle T_{21}\langle /c\rangle} T_{22}\langle /b\rangle\cdots$$

のように要素 $\langle c\rangle$ がオーバラップして指定されることによって生成される。XMLの整形性 (well-formedness) を考え

$$\cdots\langle a\rangle T_{11}\underline{\langle c\rangle T_{12}\langle /c\rangle}\langle /a\rangle\langle b\rangle\underline{\langle c\rangle T_{21}\langle /c\rangle} T_{22}\langle /b\rangle\cdots$$

と分割すると、2つの $\langle c\rangle$ の論理的な扱いが必要となる。

そこで識別子と特殊な属性 *overlap* によって

$$g_0 = (\_, \_, \{const = \text{“}list\text{”}\}, \{g_1, g_2, g_3\})$$
$$g_1 = (\_, a, \{const = \text{“}list\text{”}\}, \{g_{11}, g_{12}\})$$
$$g_2 = (\_, b, \{const = \text{“}list\text{”}\}, \{g_{21}, g_{22}\})$$
$$g_3 = (\_, c, \{const = \text{“}list\text{”}, overlap = \text{“}yes\text{”}\}, \{g_{12}, g_{21}\})$$
$$g_{11} = (\_, pc\_data, \_, \{T_{11}\})$$
$$g_{12} = (\_, pc\_data, \_, \{T_{12}\})$$
$$g_{21} = (\_, pc\_data, \_, \{T_{21}\})$$
$$g_{22} = (\_, pc\_data, \_, \{T_{22}\})$$

と定義する。これでテキストの順序は保証され、重複を防ぐことができる。論理エレメントとしう視点からは、$\langle a\rangle, \langle b\rangle, \langle c\rangle$ は対等であるが、テキストの重複を防ぐために *overlap* を導入している。この属性をどこに付加するかは一意的ではないが、相互変換は容易である。

これと類似した概念が遺伝子情報の構造化で使用されている [9]。これはゲノムの配列情報が、巡回構造や重なりあいを考慮しなければならないために、XMLの拡張を行ったものである。このポイントは配列情報の絶対位置によって一貫性を保証するところにある。本稿のアプローチは、識別子によって一貫性を保証する点が異なっている。

3.5 観光ポイント

デジタルテーマパークでの観光ポイントは、

- このポイントの地理的情報
- 種々の属性
- このポイント自体のマルチメディア提示情報
- 内部の観光ポイント
- 内部の推奨コース

が必要となる [6]。上記に述べたように、ポイント間のコースもポイントになりうるので、グラフのアークとは異なり、ノードとして表現する。

$$g_0 = (c, tour\_point, \{course = \text{“}both\text{”}, from = \text{“}id\_a\text{”}, to = \text{“}id\_b\text{”}\}, \{\cdots\})$$
$$g_1 = (id\_a, tour\_point, \_, \_)$$
$$g_2 = (id\_b, tour\_point, \_, \_)$$

ここで $\langle a\text{-}attr\rangle$ 中の $from$ と to がどこの観光ポイントを結ぶかを示している。指定されるポイントは識別子であるが、両ポイントを結ぶコースは必ずしもひとつではないので、コース自体に識別子を付加して区別することも可能である。また、たとえ両ポイントの両方向の通行が可能でも、方向性

はこのコースの情報 ($\langle e\text{-}attr\rangle$ 内の情報) の提示順序を示すために必要となる。その結果、これらを逆に操作する関数も必要となる。

3.6 文書群

文書群は、論理的にはひとつとして扱いたい複数の文書を定義するものである。文書群の場合、

$$\begin{aligned} g &= (\_, doc, \{doc\_group = \text{“}yes\text{”}\}, \{g_1, g_2, \cdots, g_n\}) \\ g_i &= (\_, chapter, \_, \{d_{i1}, \cdots\}) \end{aligned}$$

と定義される。つまり $g_1, g_2, \cdots$ の文書の集まりを論理的にひとつの文書 g として扱うことを定義している。

4 データ操作

さまざま操作が考えられるが、ここでは基本的な操作だけを定義する。

4.1 グラフのマージ

分散環境では、同一のオブジェクトがさまざまな表現で散在することがよくある。検索結果としてそれらを得たとき、同一識別子のグラフはマージされなければならない。等値制約に関するマージは、新たな変数の制約が出たとき、あるいは利用者の指示によって行われる。識別子や等値制約を使用する場合、それらの有効範囲が問題になるが、ここでは単純化のためそれらは大域的であることを仮定する。

グラフの定義にあるように、識別子は単一のラベルに対して定義されるので、グラフ g_1, g_2 に対して、id, lab がグラフから識別子とエレメントラベルを取り出す関数とすると、$id(g_1) = id(g_2)$ のとき $lab(g_1) \neq lab(g_2)$ は矛盾、つまり g_1, g_2 の定義が無効 ($\bot$) となる。それ以外の場合、各属性のマージが必要となる。

- $\langle a\text{-}attr\rangle$ に対しては同一ラベルのものは同一の値を持たなければ矛盾、それ以外は、マージされる。
- $\langle e\text{-}attr\rangle$ に対しては、構成子は同一でなければならないが、set の場合合併、$list$ の場合接合されることになる。未定義の場合、既定義のものに合わされる。エレメント属性間でマージ可能なものが処理される。

たとえば

$$\begin{aligned} g_1 &= (id, el, \{const = \text{“}set\text{”}\}, \{g_{11}, g_{12}\}) \\ g_2 &= (id, el, \{shop = \text{“}Soja\text{”}\}, \{g_{21}, g_{22}\}) \end{aligned}$$

のとき、g_1 と g_2 をマージすると、

$$\begin{aligned} g_3 = (id, el, \{const = \text{“}set\text{”}, shop = \text{“}Soja\text{”}\}, \\ \{g_{11}, g_{12}, g_{21}, g_{22}\}) \end{aligned}$$

が得られる。この結果の $\langle e\text{-}attr\rangle$ も上記条件に合致すればさらにマージされる。マージする値に ψ- 項 [2] のような型指定を導入することによって、マージの可能性を高めることも考えられるが、これは将来の課題とする。

等値制約の場合は

$$\begin{aligned} g_1 &= (\_, el, \_, \{g_{11}, \cdots\}) \\ g_2 &= (\_, el, \_, \{g_{21}, \cdots\}) \end{aligned}$$

のとき、$g_1 = g_2$ の制約が与えられたとき、

$$g_1 = g_2 = (\_, el, \_, \{g_{11}, \cdots, g_{21}, \cdots\})$$

のようなマージが発生する。これは、異なった対象と考えられていたものが同定されたとき、あるいは変数の制約伝播で等値となったときなどに必要となる。

4.2 パスと識別子によるアクセス

2 種類の属性によってアクセスが異なっている。

$$\begin{aligned} \langle a\text{-}path\rangle &::= \langle graph\rangle[.\langle path\rangle] \\ &\quad | \quad \langle graph\rangle[.\langle path\rangle].\langle a\text{-}label\rangle \\ \langle path\rangle &::= [\langle path\rangle.]\langle e\text{-}label\rangle \\ &\quad | \quad [\langle path\rangle.]list(\langle num\rangle) \end{aligned}$$

で定義される。$list(\langle num\rangle)$ はリストの特定の要素を指定している。

アクセスパス a_1, a_2 に対して、$\langle path\rangle$ p が存在して $a_1.p = a_2$ のとき $a_1 \preceq a_2$ と定義する。

与えられたグラフに対して $\langle a\text{-}path\rangle$ を適用することによって、グラフの部分構造が特定される。

$$\begin{aligned} g &= (id, el, \{al_1 = av_1, \cdots\}, \{g_1, g_2, \cdots, g_n\}) \\ g_i &= (id_i, el_i, \_, \{g_{i1}, \cdots\}) \end{aligned}$$

とする。この場合、

$$\begin{aligned} g.el &= \{g_1, g_2, \cdots, g_n\} \\ g.al_i &= av_i \mid \top \\ g.el.el_i &= \{g_{i1}, \cdots\} \mid \top \\ g.list(i) &= \{g_i\} \mid \top \qquad const = \text{“}list\text{”}\text{の場合} \end{aligned}$$

と定義される。グラフの代わりに識別子 id を使用しても同じ結果を得る。これはグラフの集合に拡張することも可能で

$$\{g_1,\cdots\}.al = \{g_1.al,\cdots\}$$
$$\{g_1,\cdots\}.el = \bigcup_i (g_i.el)$$

グラフにラベル名の列を付加したものをアクセスパスという。ここで $\top$ は指定されたラベルが存在しない場合の値である。$const$ = "$list$" の場合木の幅優先に展開され $\top$ は省略できないが、$const$ = "set" の場合 $\top$ は省略され同一要素は縮退させられる。集合は複数のラベルに対応するものなので、意味論は略記としている。

たとえば図2で $g_0.book\text{-}order.goods.book.name$ は $\{g_{312}, g_{322}\}$ を指している。

4.3 基本的検索

検索は単位質問の論理的組合せであるが、各単位質問は、

$$\langle a\text{-}path\rangle \quad \langle 比較演算子\rangle \quad \langle 検索条件\rangle$$

の形をしており、アクセスパスで指定された $\langle a\text{-}attr\rangle$ あるいは $\langle e\text{-}attr\rangle$ の集合が検索の対象となる。対象となる値は3.3で生成されたものが使用される。論理的エレメントに対しても同様である。単位質問の結果は部分グラフの集合である。たとえば図2に対して $g_0.book\text{-}order.goods.book.publish = 2000$ という問合せに対しては、g_{321} が返される。

このグラフはアクセスパスに変換できることに注意する必要がある。たとえば上の g_{321} は、集合の要素を識別するために、$set(\langle num\rangle)$ あるいは $list(\langle num\rangle)$ が対応するラベルに付加されて、$g_0.book\text{-}order.goods.book[set(2)].publish$ というアクセスパスに変換できる。もし g_{321} が複数から参照されていれば複数のアクセスパスに変換できるので、変換すべき範囲を明確にする必要がある。

グラフに対する問合せでアクセスパスは正規表現を可能にすることが多いが(たとえば文献[1]での Lorel, UnQL, XML-QL)、それは可能性のあるアクセスパスに展開し、複数の基本的検索に分解し、検索条件を充足するアクセスパスを返せばよく、利用者にとっては重要だが、モデルとして本質的ではないので、ここでは省略する。

4.4 論理演算とアクセスパスの縮退

4.3節で得られたグラフ集合間の論理演算を定義する。グラフ g をアクセスパスに変換した結果を $a\text{-}path(g)$ と書くと、$\{g_1, g_2,\cdots\}$ の変換結果は $\bigcup_i(a\text{-}path(g_i))$ となる。この変換には参照関係の範囲を指定する必要がある。$a_1 \preceq a_2$ のとき、a_2 は a_1 に対してより特化した情報を持つので、アクセスパスの集合 S_1, S_2 間にスミス順序

$$S_1 \preceq_S S_2 \stackrel{\text{def}}{=} \forall a_2 \in S_2, \exists a_1 \in S_1.a_1 \preceq a_2$$

を定義する。この順序で極小となる集合を、対応する同値類の代表とする。検索の結果得られるアクセスパスの集合は、この意味での代表とする。以下このような集合のみを考える。

$a_1 \preceq a_2$ のとき、$a_1 \wedge a_2 = a_1$、$a_1 \vee a_2 = a_2$ と定義することにより、アクセスパスの集合間に論理演算を定義する。

$$S_1 \wedge S_2 = \{a_1 \wedge a_2 \mid a_1 \in S_1, a_2 \in S_2\}$$
$$S_1 \vee S_2 = \{a_1 \vee a_2 \mid a_1 \in S_1, a_2 \in S_2\}$$

得られたアクセスパスは部分グラフを特定しているが、アクセスパスを縮退させることによって部分グラフを拡張することができる。

$$\pi_{l_i}(g.l_1.l_2.\cdots.l_i.\cdots.l_n) = g.l_1.l_2.\cdots.l_i$$

のようは射影に類似した操作と定義できる。結果の集合は上のスミス順序の意味で正規化される。

4.5 文書群に対する問合せの展開

文書群は、問合せの結果が同一であることによってそれらが論理的に同一であることを保証するものである。したがって通常の問合せを文書群用に変換することが必要である[5]。3.6節の定義に対して、$\sigma_F(g)$ という問合せはまず、$\sigma_{F_1}\sigma_{F_2}\{g_1, g_2,\cdots\} = \sigma_{F_1}\{\sigma_{F_2}(g_1), \sigma_{F_2}(g_2),\cdots\}$ と変換される。σ_{F_1} は各文書からの結果をまとめあげる処理である。

4.6 コースの抽出

デジタルテーマパークでは観光ポイント間のコースを生成する。観光ポイントは一般的に $g = (a, tour\_point, \_, S)$ のように表現される。S を単純化して a の内部観光ポイントの集合と考える。すると、あるポイント a から他のポイント b までのコース L は、以下のように $course(a, b, L)$ を再帰的に定義することによって求めることができる。

$$course(a, b, [a, b]) \Leftarrow (\_, tour\_point, \_, S), a, b \in S$$
$$course(p, a, [p, a]) \Leftarrow (p, tour\_point, \_, S), a \in S$$

$$course(a,b,L') \Leftarrow course(b,a,L), reverse(L,L')$$
$$course(a,b,L) \Leftarrow course(a,Z,L_1), course(Z,b,L_2), append(L_1,L_2,L)$$

2ポイントを結ぶコースは一意的には決まらないので、

$$L_1 \preceq L_2 \stackrel{\text{def}}{=} \forall a \in L_1.a \in L_2 \wedge \forall a \in L_1, b \in L_2. (a = L_1[i] = L_2[i'] \wedge b = L_1[j] = L_2[j'] \wedge (i \leq j)) \supset i' \leq j'$$

によってコースの冗長度を定義することができる。

ここで得られたコース L に3.5節の観光ポイントとしてのコースを埋め込むことと、観光の時間的制約を反映させることが重要となる。

4.7 リンクの巡行

XML でのリンクは、id 属性の使用、XPointer/XLink などがあるが、本稿では識別子と等値制約の2種類のリンクを考えた。これは本質的に両方向のリンクであり、片方向のリンクは両方向リンクの属性として考えることにした。これは、参照文献などの応用を考えると、たとえ片方向のリンクでも両方向リンクとして扱いたいことが多いからである。

4.3節で述べたグラフのアクセスパスへの変換がリンクを巡回することに対応している。たとえば「条件 C_1 という論文を参照している条件 C_2 の論文を求める」という問合せを考えた場合、「条件 C_1 という論文」という検索と、この結果の「論文を参照している条件 C_2 という論文」の2つに分解できる。前者の結果得られたグラフを、後者の論文を含む定義域でアクセスパスに変換することで結果が得られる。

5 おわりに

XML を拡張する要求は多くの応用に見ることができる。本稿では、著者たちが研究開発に携わっている応用の要件を反映するために、XML の拡張モデルを議論した。構文的に XML を拡張するのではなく、XML に特殊な属性を持たせることによって拡張した。これはデファクトスタンダードとしての XML との互換性をできるだけ維持した上で、拡張機能を利用したいからである。

本稿のモデルは以下の特徴を持っている。

- 従来のデータモデルの様々な構成子を利用できる。
- 意味的に XML の構文に反している論理エレメントを自然に記述できる。
- 内容的にひとつに扱いたい文書群を自然に記述できる。
- リンクの巡行を手続き的に辿るのではなく、グラフのアクセスパスへの変換として宣言的に記述できる。
- 電子化文書の共有情報や個別化情報を統一的に扱える。
- デジタルテーマパークなどの応用の特殊な要求も簡単に組み込める。

今後の検討点として、実装方式、型情報の導入、利用者向け問合せ言語、などを考えている。さらに現在対話的マルチメディア情報提示の言語としての位置付けももってきた QUIK[7] との整合性を議論している。

参考文献

[1] Serge Abiteboul, Peter Buneman, and Dan Suciu, *Data on the Web — From Relations to Semistructured Data and XML*, Morgan Kaufmann, 1999.

[2] Hassan Aït-Kaci, *A Lattice Theoretic Approach to Computation Based on a Calculus of Partially Ordered Type Structures*, Dissertation, Univ. of Pennsylvania, 1984.

[3] 藤野猛士, 野宮一生, 横田一正, 國島丈生, 三宅忠明, "構造化文書に基づいた対話的戯曲提示システムの実現," 情報処理学会データベースシステム研究会, 東京, May, 2000.

[4] Takeo Kunishima, Kazumasa Yokota, Bojiang Liu, and Tadaaki Miyake: "Towards Integrated Management of Heterogeneous Documents," *Cooperative Databases and Applications '99*, pp.39-51, Springer, Sep., 1999.

[5] 國島丈生, 鈴木美沙, 宮川由香, 横田一正, "構造化文書の設計の異種性解消のための「文書群」の導入と検索機能の実現," 情報処理学会データベースシステム研究会, 東京, May, 2000.

[6] 劉渤江, 横田一正, 岡本辰夫, "デジタルテーマパークのモデリングの検討," 電子情報通信学会データ工学ワークショップ, 近江八幡, Mar., 2000.

[7] Bojiang Liu, Kazumasa Yokota, and Nobutaka Ogata, "Specific Features of the QUIK Mediator System," *IEICE Transaction on Information and System*, Vol. E82-D, No.1, pp.180-188, 1999.

[8] 三宅忠明, 横田一正, "情報処理としての作品解析 — デァドラ伝説の研究から," 英語青年, vol.146, no.1, pp.6-9, Apr., 2000.

[9] Aaron J. Stokes, Hideo Matsuda, and Akihiro Hashimoto, "GXML: A Novel Method for Exchanging and Querying Complete Genomes by Representing them as Structured Documents," 情報処理学会論文誌: データベース, vol.40, no.SIG 6, pp.66-78, 1999.

[10] Hideki Yasukawa and Kazumasa Yokota, "Labeled Graphs as Semantics of Objects," 情報処理学会データベースシステム・人工知能合同研究会, Nov., 1990.