2D-6

ネイティブ XML データベースを用いた XML 文書管理システムの有用性の検証

濱 野 泰 男† 北 川 亘††

近年，XML を用いた文書管理システムの案件が増加しており，関係データベースを用いた従来の XML 文書管理システムには格納・部分的な更新・各要素に対する検索の性能や柔軟性に対していくつかの問題が存在する．本論文では，ネイティブ XML データベースを用いることにより，これらの問題点が解決され，XML 文書管理システムが高効率化されることを検証した．

1. はじめに

XML の普及とともに各業界でデータモデルの標準化が進み，XML フォーマットを使った文書管理ソリューションの案件が増加している．日本アイ・ビー・エム，ソフトウェア開発研究所では大手新聞社に対して，ニュース記事管理の国際標準フォーマットである NewsML [2] を用いた文書管理ソリューションの提供を行っている．筆者らの提供する NewsML ソリューションでは，次のような手法で文書管理システムを開発している．XML 文書の中でユーザからの問合せ対象となる部分を分解し，関係データベース (RDB) に写像して格納する．それに加えて，XML 文書の全文を Character Large Object (CLOB) 形式で格納する．

筆者らの文書管理システム (以下，写像システム) では RDB に写像することによるオーバヘッドが存在する．XML のスキーマから RDB のスキーマへの変換は，具体的にはテーブルに写像するため，XML を扱うシステムには避けて通れないコストの高い処理を余儀なくされている．これらの問題を回避するために，XML 文書を木構造のまま扱えるネイティブ XML データベースが望まれている．本論文では，DB2 V9 のネイティブ XML データベース機能 (pureXML 機能) を利用することによって，上述の問題が解決されることを検証する．

2. 写像システムの問題点

写像作業のオーバヘッド: XML 文書は木構造であり RDB は表構造である．この構造の違いにより，XML 文書を RDB へ写像する作業には多くの工数を要する．XML 文書から RDB への写像アルゴリズムは多く提案されているが，筆者らの開発する NewsML による標準化ではお客様ごとに XML の拡張タグが定義されており，高速な検索のための RDB の関係スキーマ設計が個別のシステムで必要になる．

写像による柔軟性の損失: XML のスキーマを設計することはシステムの利便性，拡張性を決定付ける重要な設計になる．XML 文書を RDB に写像する場合は，格納する XML 文書のスキーマが固定されてしまうため，その変更が容易でなくなる．一度検定したスキーマの変更は，関係するテーブル再設計を必要とし，システム全体への影響が高くなることから，通常はスキーマ変更をしないように設計段階で多くの工数および時間をかけているのが現状である．

XQuery から SQL へ変換するオーバヘッド: 写像システムでの XML 文書への問合せは XQuery によって記述される．その際 XQuery から SQL への変換が必要になる．この変換は，XML 文書の要素・属性がどのテーブルに写像されているか考慮しなければならず，非常に複雑で多くの工数を要する．図 1 は，変換の際にロケーションステップごとにテーブルの結合が必要になる例を表している．

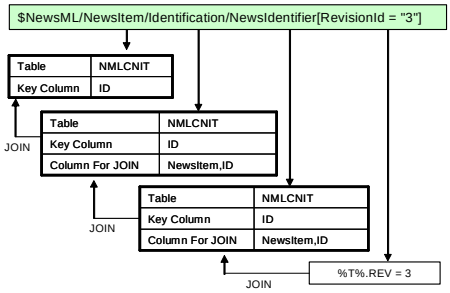

図 1 XQuery から SQL への変換例

参照を含む問合せ: NewsML 文書は他の NewsML 文書に対する参照を持つことができる．この参照情報を用いて，ある文書が参照する子孫の文書を取得したいという利用者の要求がある．既存の文書管理システムでは，NewsML 文書の参照関係を表したリンクテーブルを用いた手法 [3] によって，このような問合せ要求に対応している．しかし，リンクテーブルを用いた手法では子文書の問合せにしか対応しておらず，孫以降の文書の問合せは実行できない．また，参照を表す SQL 問合せは結合が深くなり，パフォーマンスを劣化させる要因となる．

上記の実システムにおける問題を pureXML 機能でどのように解決できるかを，第 3 節で検証した．

3. DB2 V9 による文書管理システムの効率化

DB2 V9 は XML 文書を木構造のまま扱うことができるので，RDB への写像の作業が必要ない．よって，写像システムでの写像作業のオーバヘッドが無くなる．また，XML 文書のスキーマが変更される際，関係スキーマの再設計をする必要がないので，XML の持つ柔軟性を活かすことができる．DB2 V9 は，XQuery を SQL に変換することなく直接実行することができる．これにより，XQuery から

† 日本アイ・ビー・エム株式会社，ソフトウェア開発研究所
†† レノボ・ジャパン株式会社，ノートブック開発研究所 (執筆時の所属は日本アイ・ビー・エム株式会社，ソフトウェア開発研究所)

SQL への変換作業が必要なくなり，工数の減少につながる．参照を含んだ XML 文書の問合せは，XQuery で容易に記述できる．例えば「id 番号 001 を持つ文書が参照している子文書を取得する」という問合せは，図 2 のような XQuery で表すことができる．

```
for   $docs in db2-fn:xmlcolumn('DB.TABLE')
      /root[id = 001],
      $ref in $docs/ref/text()
return <result>$docs/root[id = $ref]</result>
```

図 2 子文書を取得する XQuery 問合せ

次に，写像システムと pureXML システムの性能比較を行った．実験環境は Thinkpad R52，CPU Intel Pentium M 740，主記憶容量 1 GB，HDD 回転数 5,400 rpm のマシンを使用した．

実験 1: 写像作業が不要になったことで XML 文書の格納速度がどの程度向上したかを示す．1 件約 50 KB の XML 文書 1000 件を写像システムと pureXML システムにそれぞれ格納し，かかった時間を測定する．5 回の試行を行い，その平均値をとる．

表 1 XML 文書の格納にかかった時間 (ミリ秒)

写像システム	pureXML システム
310,731	37,956

表 1 は実験 1 の結果を表している．pureXML システムでは写像システムに比べて速度が約 8 倍向上した．

実験 2: 実験 1 の速度の差が写像処理の有無によるものかを確かめる．データ量はほぼ同じで，写像されるテーブルの数を変えた XML 文書を用意し，格納にかかる時間を測定する．Tn は n 個のテーブルに写像される XML 文書を表す．各文書を 100 件ずつ格納し，かかった時間を測定する．3 回の試行を行い，平均値をとる．

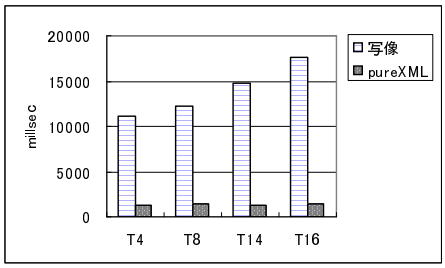

図 3 テーブルの数と格納時間の関係

図 3 は実験 2 の結果を表している．写像システムでは写像されるテーブルが増えるに従って実行時間も増えている．一方，pureXML システムでは写像されるテーブルの数に関わらず，実行時間は一定である．この結果から，pureXML 機能を使うことでテーブルへの写像の処理が不要になり，XML 文書の格納時間が短縮されたことがわかる．

実験 3: 参照を含む XML 文書の問合せ処理時間を比較する．事前に 10000 件の XML 文書をデータベースに格納しておく．そのうち参照を持つものは 100 件で，それぞれ 10 件の子文書を参照している．実行した問合せは，参照を持つ文書の子文書をすべて取得するものである．返される結果は 1000 件である．写像システムではリンクテーブルを用いた．100 回の問合せを行い，平均値を取る．表示のための時間は実行時間に含めていない．

表 2 参照を含む問合せの実行時間 (ミリ秒)

写像システム	pureXML システム
4,766	828

表 2 は実験 3 の結果を表している．写像システムに比べて pureXML システムでは速度が約 6 倍向上した．

4. XML 文書の部分更新

写像システムでの XML 文書の部分更新は，更新したい箇所を XPath で指定し，その部分に対応するテーブルと CLOB 形式のデータを更新する処理をアプリケーション側で実装することで実現している．一方 DB2 V9 では XML 文書の部分更新は提供されていない．しかし，ストアド・プロシージャを用いることで部分更新に対応できる [1]．このストアド・プロシージャは部分更新の処理を記述しやすくするが，背後では文書全体を更新しており，更新にかかる時間は文書の大きさに比例して増加する．本節では，pureXML システムでの部分更新の性能を検証した．

実験 4: 著者らのシステムで実際に使われている典型的な XML 文書 (サイズは約 50 KB) に対して，部分更新の処理速度を測定する．10000 件の XML 文書が格納されている状態で，そのうち 5000 文書に対して部分更新を行う．5 回の試行を行い，その平均値をとる．実験環境は第 3 節のものと同じである．

表 3 部分更新の実行時間 (ミリ秒)

写像システム	pureXML システム
376,581	389,265

表 3 は実験 4 の結果を表している．この結果から，典型的な XML 文書では pureXML システムは写像システムと同等の性能が得られることが確認できた．よって，pureXML システムでの部分更新は実用上十分な性能を持つと言える．

5. おわりに

本論文では，DB2 V9 の pureXML 機能によって XML 文書管理システムが効率化できることを実証した．今後の課題として，pureXML システムへの移行ツールの作成と評価などがあげられる．

謝　辞

本論文の方針の決定や執筆にあたって，多大な助言をいただいた日本アイ・ビー・エム，ソフトウェア開発研究所の橋本光治氏，中野和人氏に深く感謝いたします．

参 考 文 献

1) XML application migration from DB2 8.x to DB2 Viper, Part 1: Partial updates to XML documents in DB2 Viper. http://www-128.ibm.com/developerworks/db2/library/techarticle/dm-0605singh/, 2006.
2) NewsML チーム. NewsML レベル 1 解説書 (第 1.0.3 版), 2003.
3) 北川亘. マルチリファレンスを適用した NewsML の高速アクセス手法の提案. In *FIT 2005*, 2005.