# XML データベース

米田 有佑，王　路易
Yusuke YONEDA，Luyi WANG

## 1　はじめに

近年，情報技術の進歩により膨大な情報が電子的に取り扱われるようになった．また，それに伴いデータベース技術も発展を遂げてきている．従来のデータベース技術では，扱うデータの構造はあらかじめ定まった形に整形し，強力な問い合わせ言語を用いることで膨大なデータを高速に扱うことを実現していた．しかし，そのようなデータベースでは構造が異なったデータの追加などの変更が生じた際に，データ全体の構造まで手を加える必要があり，大きな手間がかかっていた．現在では，IT の発展に伴って社会，ビジネスなどデータベース技術が利用されている状況は著しく変化している．そのような風潮の中で，単純にデータを速く処理できるだけでなく，より柔軟にデータの変化を扱えるデータベース技術に注目が集まっている．本稿では，柔軟にデータの変化を扱えるデータベースとして注目を浴びつつある XML データベース（以下 XMLDB）について述べる．

## 2　XML:eXtensible Markup Language

本章では，XMLDB の根幹的な技術である XML について述べる．

### 2.1　XML の概要

XML は Standard Generalized Markup Language（以下 SGML）を発展させた言語である．SGML は，異なる環境のコンピュータがネットワークで接続される中で，電子文書をスムーズに交換する目的として開発された．SGML は次の 2 つの大きな特徴を持っている．

- メタ言語
- マークアップ言語

1 つ目の特徴であるメタ言語について説明する．コンピュータで電子文書の交換を行う際には，各コンピュータに共通の言語のフォーマットを規定する必要がある．そのために電子文書交換のフォーマットを作成する機能を持った言語がメタ言語である．次に 2 つ目の特徴であるマークアップ言語とはテキストファイルの中に内容と同時に特定の記号を利用して付加情報を記述するものである．XML は SGML の「メタ言語」と「マークアップ言語」の特徴を受け継いでおり，異なる情報システム間でのデータ共有を容易にする機能を持つ．

### 2.2　XML の歴史

現在，XML は web データ記述の標準文法として広まっている．初期のインターネット普及の技術的な要因として SGML をもとに作成された HTML(HyperText Markup Language) の存在があった．しかし，その後のインターネットの更なる拡大に伴い，さまざまな情報をインターネット上で公開，交換を行おうとする意識が高まった．しかしデータ交換という点では HTML は十分な機能を備えておらず，また SGML も助長的で利便性に欠けており，不適格だった．そこで W3C[*1] によってインターネットの標準形式として SGML を簡潔に利用しやすく発展させた XML が考案された．これにより，XML の利用によって共通の情報フォーマットの作成が容易になった．

### 2.3　XML の記述

XML で記述された文書の中では，最も基本的な情報単位として要素という概念が用いられる．さらに要素に対して付加的な情報を付け加えるためには属性と呼ばれる概念が使用される．また，XML ではデータをツリー構造で表現する．XML データと，そのデータ構造の具体例として Fig.1 を示す．

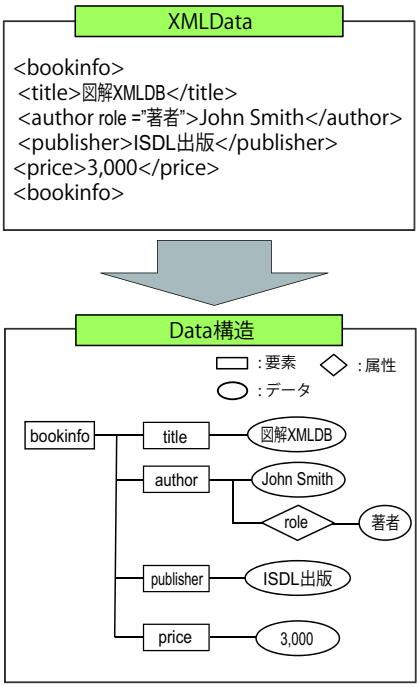


Fig.1　書籍情報を示した XML 文書 (出典：自作)

Fig.1 の XML データでは bookinfo，title， author，publisher，price が要素であり，role は属性である．ここでは bookinfo 要素の中に title, author, publisher，price

[*1] World Wide Web Consortium の略称．WWW で使用される各種技術の標準化を推進する為に設立された団体．

の要素が入れ子状になっている．これは記述されている情報を bookinfo 要素を親とした階層構造としても認識できる．従って title, author, publisher の要素は bookinfo 要素の子としてみることができ，その中に格納された個々のデータも要素の子として表される．同時に，role 属性も publisher 要素の子としてみることができる．従って，この XML データは Fig.1 で示したツリー構造で表現される．このようなツリー型のデータ構造は新たにデータを追加する際に要素を親要素に追加するだけでよく，柔軟に構造の変化に対応できる特徴を持っている．

## 3　XML データベース

本章では現在の XMLDB の発展の過程と特徴および活用分野について述べる．

### 3.1　XML データベースの概要

XMLDB とは XML データを格納できるデータベースである．しかし，XML データの格納方法については多様な種類が存在している．その中で主流となっているのがリレーショナルデータベース（以下 RDB）に XML を格納する XMLDB と，XML 形式のまま XML を格納する XMLDB である．その中で，後者の XML の情報を XML 形式のまま保存、検索、出力することができるデータベースのことを特にネイティブデータベースと呼ぶ．以下の Fig.2 は XMLDB の形態を示したものである．

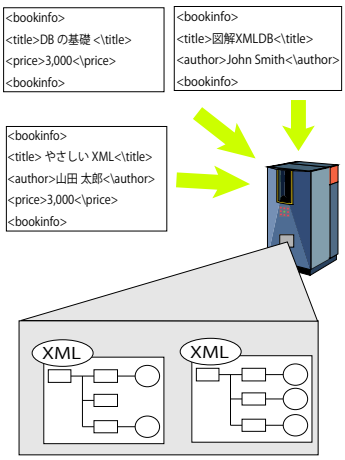


Fig.2　XMLDB 概念図 (出典：自作)

### 3.2　XML データベースの発展

XMLDB は XML の利用用途の拡大と共に，その機能が注目されてきた．しかし，XMLDB は登場時から有用性が認められていたにもかかわらず，当初は普及しなかった．本節では，XMLDB が現在注目を集めている理由を XMLDB の発展という観点から考察する．

#### 3.2.1　従来のデータベース

従来，データベースの主流として利用されてきたのは RDB であった．この RDB には表形式によるデータ表現とリレーションによる表の関連よってデータを扱うという特徴を持っている．これらに関連して，RDB ではデータベースの構造であるスキーマを厳密に定義する．また，表のデータの参照や変更などといったデータベースの操作には SQL(Structured Query Language) と呼ばれる問い合わせ言語が利用される．Fig .3 は RDB でのデータベース操作を示した図である．社員表と部門表という 2 つの表を部門番号という共通の情報を持つ列を利用して表を関連付けて処理を行っている．

社員表

| ID | 社員名 | 給与 | 部門番号 |
|---|---|---|---|
| 01 | 山田 | 10,000 | 10 |
| 02 | 田中 | 70,000 | 20 |
| 03 | 鈴木 | 3,000 | 30 |
| 04 | 伊藤 | 50,000 | 10 |

部門表

| 部門番号 | 部門名 | 所在地 |
|---|---|---|
| 10 | 経理 | 東京 |
| 20 | 営業 | 大阪 |
| 30 | 経理 | 奈良 |

社員表 ＋ 部門表

| ID | 社員名 | 給与 | 部門番号 | 部門名 | 所在地 |
|---|---|---|---|---|---|
| 01 | 山田 | 10,000 | 10 | 経理 | 東京 |
| 02 | 田中 | 70,000 | 20 | 営業 | 大阪 |
| 03 | 鈴木 | 3,000 | 30 | 経理 | 奈良 |
| 04 | 伊藤 | 50,000 | 10 | 経理 | 東京 |

Fig.3　リレーショナルデータベース (出典：自作)

RDB は，スキーマの厳密な定義によってデータを高速に扱えるが，一方でデータ構造の変化が生じた際にはスキーマの再構成などの大きな手間が生じるというデメリットも持っている．

#### 3.2.2　黎明期の XMLDB（第一世代）

XMLDB が登場したのは XML1.0 が W3C で標準化されたのとほぼ同時期である 2000 年前後であり，この当時の XMLDB は第一世代と呼ばれている．しかし，この時 XMLDB は現在ほど注目を集めていなかった．その原因に次の 3 点における欠点が考えられる．

- データ構造の柔軟性
- データの大量処理
- データの高速処理

１つ目の原因に関して，当時の XMLDB は RDB に XML データを格納していたものが多かった．このように RDB に XML を格納する XMLDB は現在でも存在しているが，当時の XMLDB はスキーマやインデックスの設定を必要とするものや，設定をしないと実際には使い物にならない製品が多数存在していた．スキーマの設定はデータベースにおいて検索などの処理パフォーマンスを改善することにつながるが，一方でデータベースの扱うデータ構造が固定的になる欠点がある．またインデックスも設定を厳密に行いすぎると検索性能の向上というメリットに加えてデータ更新性能の悪化というデメリットがある．スキーマやインデックスの設定は XML のツリー型データ構造の特徴であるデータの追加・更新の容易さというメリットを損なっていた．

２つ目の原因に関して，当時の XMLDB はデータを大量に処理する際にパフォーマンスの低下が生じるという弱点が存在した．具体的には，1GB を超える規模のデータベースは扱えない，もしくは極端にパフォーマンスが

低下するものがほとんどであった．XML データはその構造から RDB の様な単純なレコードデータよりもサイズが大きくなりやすいため，1GB 程度の規模では不十分となるケースが多かった． また同時に XMLDB 自身の問題だけでなく，データ構造やマッピングなどを設計段階で最適化して，大量のトランザクションを処理するという RDB と同様の使い方で使用したケースが多かったことも原因の１つである．

３つ目の原因に関して，当時の XMLDB は XML という木構造のデータを扱っているために検索処理が低速であるという弱点も持っていた．このように，当時の XMLDB は単純に XML データが格納可能な RDB に過ぎず，XML データのメリット以上にデメリットによって実用的には RDB に劣るデータベースであった．

#### 3.2.3 現在の XMLDB（第二世代）

2003 年頃から，大量の XML データを高速に処理できる XMLDB が市場に投入され始めた．この当時から現在にかけての XMLDB は第二世代と呼ばれている．以下の table.1 は現在の主要な XMLDB 製品をまとめた表である．

Table1　主要な XMLDB 製品 (出典：自作))

| 製品名 | 企業名 |
|---|---|
| **ネイティブ DB** | |
| Cyber Luxeon | サイバーテック |
| TX1 | 東芝ソリューション |
| Shunasaku | 富士通 |
| **XML 対応 RDB** | |
| Oracle Database | Oracle |
| MS SQL Server | Microsoft |
| DB2 | IBM |

この第二世代では前節で述べた問題点を解決する手段として２種類の XMLDB が出現した．１つは XML データをそのまま格納，検索することが可能なネイティブ XMLDB であり，もう１つは進化した XML 対応の RDB である．どちらにも共通してスキーマレス，自動化したインデックスの設定，データの大量処理能力の改善，XQuery と呼ばれる W3C より標準化された問い合わせ言語が対応になったという特徴がある．スキーマレスの特徴は，XMLDB に柔軟にデータを扱える機能を与えた．次にインデックスの自動化は，XML の特徴であるツリー型のデータ構造とスキーマレスの弱点である検索等の処理パフォーマンス低下を補い，高速化を可能にした．XQuery の標準化は XMLDB の各製品における問い合わせ言語が共通になったことを意味する．さらにクエリの最適化技術も進化し，検索処理のパフォーマンスも向上させた．同時に，データの大量処理もハード技術の発展に伴い数十から数百 GB に対応する製品が出現した．XMLDB は従来の欠点の補完し実用に耐える性能を得ただけでなく，RDB では不可能であった XMLDB 独自に柔軟にデータを扱える新しいデータベースとして発展した．このような要因から，XMLDB に対する注目は現在さらに高まっている．

### 3.3 XMLDB と RDB の比較

本節では従来におけるデータベースにおいて主流である RDB と XMLDB における機能を比較し，XMLDB の有用性について考察する．格納データの構造，スキーマの必要性，構造変更時のコスト，データの高速処理性能の４点について XMLDB と RDB の比較を行う．この比較の一覧を table.2 に記載する．

Table2　XMLDB と RDB の比較 (参考文献 [4] より引用)

| **比較項目** | **RDB** | XMLDB |
|---|---|---|
| 格納データの構造 | 定型 | 半定型 |
| スキーマの必要性 | 必須 | 任意 |
| 構造変更時のコスト | 高 | 低 |
| データの高速処理 | 高速 | 低速 |

１つ目の格納データの構造において，RDB では定型的で構造的なデータが適している．それに対し XMLDB では XML をデータとして扱うので半定型的で半構造的なデータが適している．半定型的なデータとは RDB のようにはっきりと構造を決定しないデータのことである． ２つ目のスキーマにおいて，RDB ではまず第一にスキーマの設計を行うことが前提となっている．一方で XMLDB ではスキーマがなくても XML をデータとしてデータベースに格納することが可能である．このスキーマレスという特徴は XMLDB の特筆すべき点である．３つ目の 構造変更時のコストにおいて，RDB ではデータ構造に大きな追加もしくは変更の必要が生じた際にシステム全体に大規模な修正をおこなう必要性が生じる．XMLDB ではデータ構造に変更の必要が生じてもデータベース自体の変更は生じないので変更時のコストは低くなる．４つ目の大量データの処理は，構造的なデータを得意とする RDB は高速で行える．一方で XMLDB は格納するデータが XML であり，階層構造であるという理由から大量のデータを一括で処理することは得意としない．この４点をまとめると RDB は扱うデータは硬直的でデータ構造の変更にも弱いが，非常に高速なデータ処理を行えるという性質があると言える．一方で XMLDB は扱うデータは非常に柔軟であり，データ構造の変化に対応できるが，一方で現段階では大量のデータ処理には不安が残ると言える．

### 3.4 利用分野

本節では，実際の社会において XMLDB の特徴がどのような分野で活用されているのかについて注目し，その一例として医療分野への利用について述べる．Fig.4 は，医療分野での XMLDB の利用を図式化したものである．医療情報では，患者，質病，検査ごとに多様な項目がある．また項目は検査機材の進歩や医師の診断の中で随時

追加・変更がおこなわれる．このようなデータは厳密にスキーマを設計する必要がある RDB では対応が困難であり，従来ではデータが有効活用されていなかった．しかし，XMLDB は多様な項目を持つデータや，データの追加・変更にも対応できる．この医療分野に XMLDB を用いることでデータの有効的な活用が可能になった．

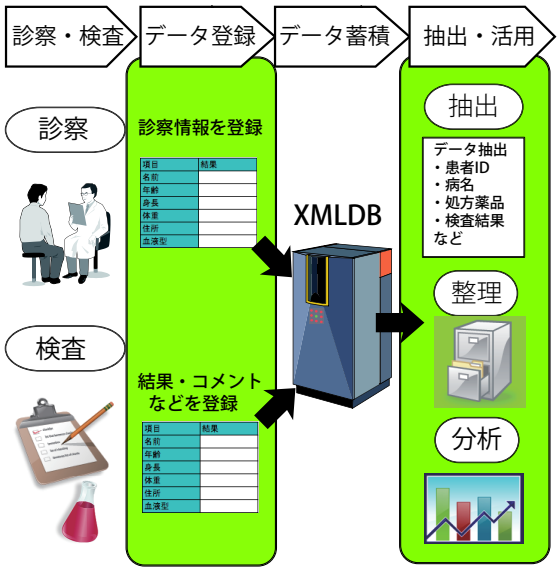


Fig.4　医療分野における XMLDB の利用 (出典：自作)

また XMLDB は医療分野以外にビジネス分野でも様々な活用がされている．第 1 にナレッジマネジメントでの利用がある．ナレッジマネジメントとは個人が持つ知識や情報を組織全体で共有，活用しようという経営手法である．この分野ではオフィスドキュメントや業務マニュアルなどの組織内で用いられていた文書を XMLDB に格納することで，見出し，文書構造での検索，新旧対照表の自動生成が可能になった．第 2 に CAD/CAM での利用がある．CAD/CAM とは設計，生産に用いられるシステムである．この分野では複雑な階層構造を持つ図面データを XMLDB でデータベース化し，web 上での共有を可能にすることで効率的なデータの管理を実現している．次に出版業界では自動組版実現への手段として利用されている．従来印刷のコンテンツは RDB ではデータ化が難しかった．しかし XMLDB を利用することでコンテンツの管理が実現された．また，電子カタログでの利用がある．カタログは商品や商品属性の更新が頻繁におこる．XMLDB ではデータの追加が容易である点からカタログの分野での利用が目覚ましい．このように，RDB では困難とされる文書や階層構造，カタログなどの頻繁に変化するデータに対して XMLDB は管理を容易にし，利用分野を拡大している．Table.3 は医療分野以外での XMLD の活用分野とその具体的な適用例を一覧で示したものである．

## 4　今後の展望

インターネットをはじめとする IT の発展により，社会を取り巻く環境の変化は現在も加速している．その

Table3　XMLDB の利用分野 (参考文献 [4] より引用)

| 分野 | 適用例 |
|---|---|
| ナレッジマネジメント | オフィスドキュメント<br>業務マニュアル<br>社内規定集 |
| CAD/CAM | 画像情報<br>設計情報 |
| 出版業界 | 自動組版ソリューション |
| 電子カタログ | 商品情報 |

流動的な環境の中で，取り扱うデータの内容だけでなく構造そのものも変化が求められている．実際に社会でも XMLDB のようなデータ構造の変化に柔軟なデータベースが必要とされる場面は拡大しており，これからも XMLDB は確実にそのシェアを広げていくだろう．しかし，現在の XMLDB が RDB のシェアに置き換わるのは困難であると予想する．なぜなら，XMLDB はインデックスの自動化，XQuery などの問い合わせ言語の進化によって従来の弱点であった大量データの高速処理という弱点を改善してきたものの，階層的なデータを扱う XMLDB は，RDB のように厳密なスキーマを持つデータベースに高速性という点では完全に不利である．現在の XMLDB では，データ変更が頻繁に必要とされる分野で拡大していくことは予想されるが，RDB も完全になくなるのではなく，固定されたデータを高速に処理する必要のある分野で用いられていくことが予想される．XMLDB は RDB では扱えなかった分野でシェアを広げていくと予想する．

## 参考文献


1) Serge Abiteboul, Peter Buneman, Dan Suciu , XML データベース入門，共立出版，2006.
2) 川越　恭二，楽しく学べるデータベース， 昭晃堂，2007.
3) 島田達巳，國友義久，小田圭二，データベース，河北印刷出版，2008.
4) XMLDB.JP
http://www.xmldb.jp/index.html
5) TECHSCORE-XML
http://www.techscore.com/tech/XML/index.html
6) XML SQUARE
http://www.utj.co.jp/xml/beg/index.html
7) UL Systems.Inc
http://www.ulsystems.co.jp/index.html
8) @IT
http://www.atmarkit.co.jp/fxml