

電子カタログにおけるネイティブ XML データベース『Sonic XIS』の活用

プログレス ソフトウェア株式会社

武末 徹也

1. はじめに

B2B やコンテンツ管理、Web サービスなど、XML という技術は様々な分野で応用されはじめている。それに伴い、蓄積、管理、再利用すべき XML データの量も増加しており、専用のデータベースへのニーズが高まっている。

ここでは、ネイティブ XML データベースの代表的製品である SonicXIS のアーキテクチャ、機能、特徴を、RDBMS 等他のデータベースと比較することで説明する。また、SonicXIS の代表的なアプリケーションイメージの一つである電子カタログを例として、SonicXIS の適用シーンと SonicXIS 活用により得られるメリットについて説明する。

2. SonicXIS

SonicXIS は、米国プログレスソフト社のビジネスユニットであるソニックソフトウェア社により開発、販売されているネイティブ XML データベースである。1998 年に最初のバージョンが発表され、現在のバージョンは 3.1.3 となっている。

2.1 SonicXIS の特徴

SonicXIS は、オブジェクト指向データベースである ObjectStore をデータベースのコアに持ち、その上に XML 処理エンジンや API、XML 関連標準の統合開発環境を搭載した製品である。以下に SonicXIS の特徴を示す。

(1) メモリデータベース

ObjectStore はデータを RDBMS のような表形式に正規化されたデータとして格納、利用するのではなく、プログラムで定義されたオブジェクトモデル、すなわちオブジェクトのインスタンスそのものとして格納する。この格納されたオブジェクトはメモリ上にマッピングされ、クライアントプログラムから利用されるため、ある意味メモリデータベースと言える。このため、データに対してメモリスピードでのアクセスが可能となり、ディスクアクセスがメインとなる他のデータベースと比較して、格段のハイパフォーマンスを提供することができる。

SonicXIS の場合、XML データをテキスト形式で格納するのではなく、W3C (World Wide Web Consortium) で規定されている XML 操作の標準インタフェースである DOM(Document Object Model) オブジェクトの形式で格納する(XML パーシスタンス)。これにより、複雑に階層化されたデータを、マッピングなどの構造変換手段を用いることなく、そのまま表現することができる。

また、SonicXIS はメモリ上の DOM オブジェクトを直接操作することができるので、データ構造そのものの更新処理(RDBMS で言うところのフィールドの増減やテーブルの増減など)を、サービスの中断なしにオンラインで行うことができる(ダイナミックモデリング)。これにより、構造変更の頻繁なデータの取り扱い、すなわちデータの拡張性を飛躍的に向上させることができる。構造変更は XML ドキュメントのレベルではなく、必要なノードのレベルで行うことができるため、プログラム処理に対するインパクトは非常に小さい。

SonicXIS では、DOM オブジェクトのハンドリングと XML 処理を行うプロセスを XML キャッシュと呼ぶ。XML キャッシュはメモリ上に展開された DOM オブジェクトと、XML 処理エンジン(DXE: Dynamic XML Engine)とで構成される。そのため、SonicXIS アプリケーションから見ると、XML キャッシュがデータベースそのものとなる。

SonicXIS ではすべてのデータベース処理を XML キャッシュを介して行う。SonicXIS では XML データの格納時点で DOM オブジェクトとするため、テキスト形式の XML データを毎回メモリ上に DOM オブジェクトとしてロード(パース)することなく、データベース上の DOM オブジェクトを直接操作することができ、メモリスピードでのリアルタイム処理が可能となる。

(2) ロードバランスとスケーラビリティ

SonicXIS には、大量アクセスに対しての負荷分散とスケーラビリティを確保するための手段として、XML キャッシュプロセスを複数分散できる機能がある(分散 XML キャッシュ)。この機能は、ObjectStore の米国特許技術である、キャッシュフォワードアーキテクチャを継承しているものである。

SonicXIS では、データベースそのものの役割となる XML キャッシュを、複数起動することができ、さらに複数の XML キャッシュは、同一のデータベースサーバ機だけではなく、複数のマシンに分散することができる。これにより、データベースがアプリケーションパフォーマンスのボトルネックとなることを回避することができる。

(3) 高機能ツール群

SonicXIS には、Windows エクスプローラのようなルック&フィールの管理ツール「DXEManager」を持つ。

また、XML 関連標準を統一したビューで開発するための IDE(統合開発環境)である「StylusStudio」も標準でバンドルしている。

RDBMS 等、他のデータソースに対するデータ連携を行うツールとして、XConnects がある。XConnects はオプションである。

(4) サポートプラットフォーム

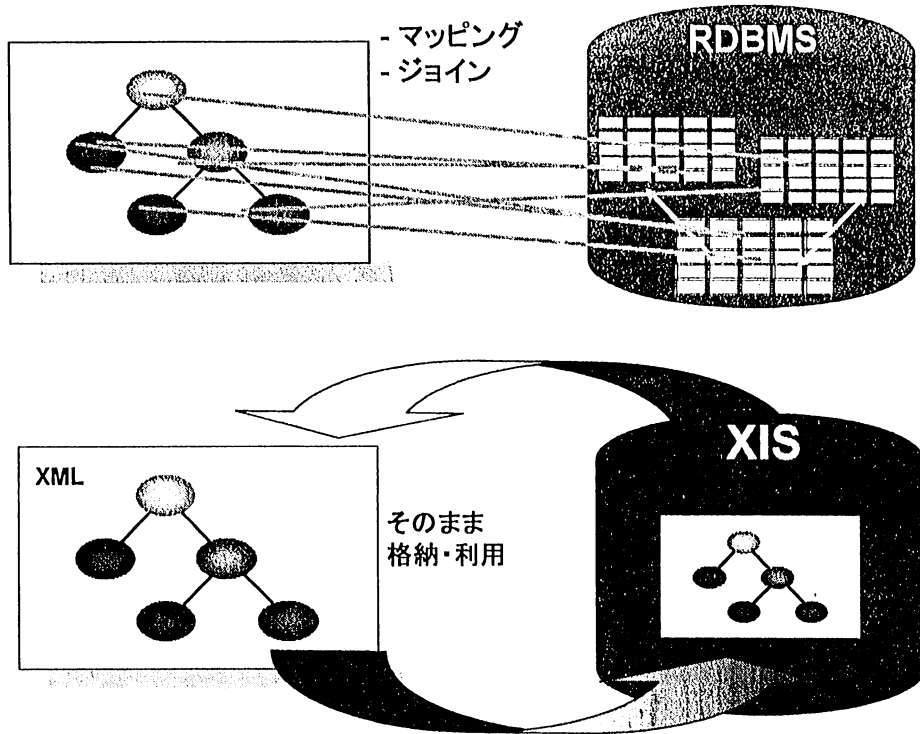
SonicXIS のサポートプラットフォームは、以下の通りである。

- Windows(NT/2000)
- Soralis
- HP-UX
- Linux

2.2 XML パーシスタンス

RDBMS の場合、XML データを利用しようとする、表形式のデータ構造と XML の階層構造とのマッピングや、複雑なテーブルジョインが必要になる。これは、テーブルジョインの多発によるパフォーマンスの低下だけでなく、表現したいデータモデルが変更された際に、毎回マッピングやプログラムのやり直し、影響範囲の調査など、開発効率の低下を招く。

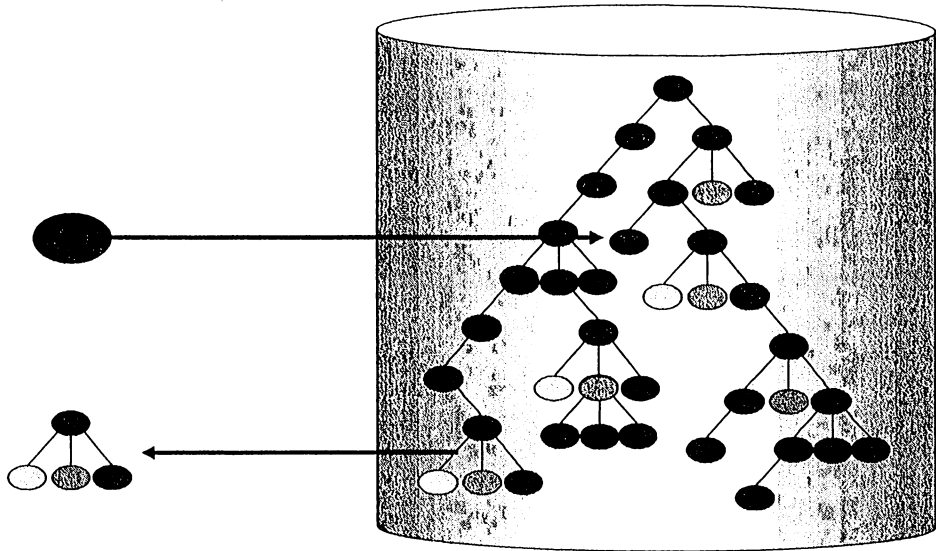
一方、SonicXIS の場合、XML の階層構造をそのまま格納することができるため、複雑なデータモデルをマッピングなしで格納することができ、さらにデータモデルへの変更にも柔軟に対応することができる。データアクセスに関しては、メモリスピードで高速アクセスができ、データへのアクセスごとに DOM パースが不要となる。



2.3 ダイナミックモデリング

SonicXIS の最大の特徴として、データモデルをオンライン/オンザフライで更新できるという特徴がある。ここで、オンラインとはシステムのサービスストップをすることなく、運用状態のままデータモデル更新ができることを言い、オンザフライとは、データベース上のデータをアプリケーションで更新し、再度データベースに上書きするような処理ではなく、直接更新を行うことができることを言う。

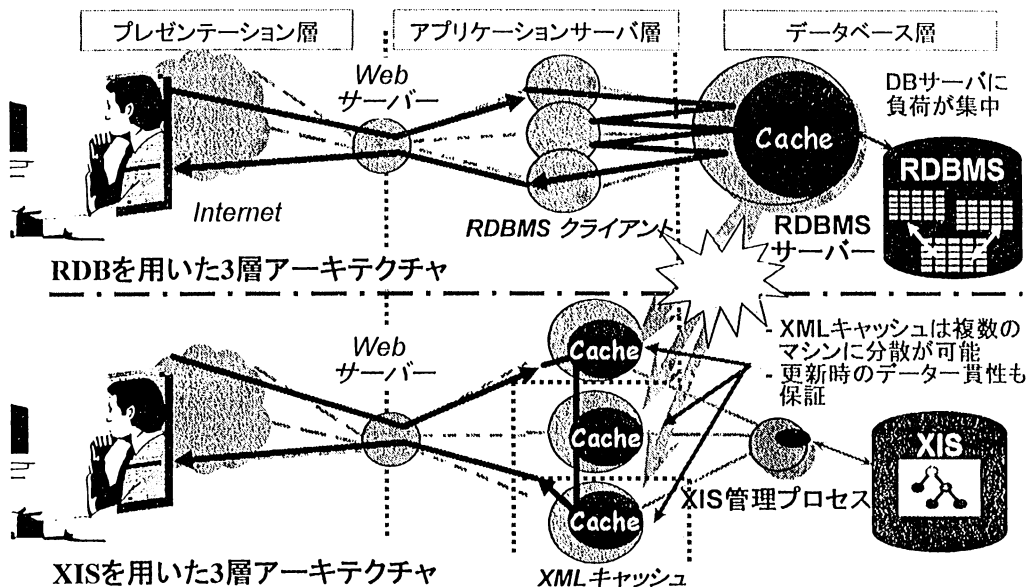
この特徴はデータそのものの拡張性を向上できるだけでなく、システム全体の可用性を向上できるという特徴を持つ。



2.4 分散 XML キャッシュ

SonicXIS は、高負荷システムへのスケーラビリティを確保するために、分散 XML キャッシュという負荷分散の機能を持つ。XML キャッシュをデータベースサーバ上、アプリケーションサーバ上、あるいはキャッシュ専用構築したサーバ上に複数個配置することができ、これによりデータベースサーバ機の CPU やメモリ等ハードウェアリソースを使うことなく、データベース処理を行うことができる。

分散した XML キャッシュ上のデータはトランザクションの一貫性が保証される。これは、ある XML キャッシュ上のデータに対して更新が発生した際に、同一リソースを担当する別のキャッシュに対して、メッセージを送ることにより実現する。データの更新が成功した場合、次に別の XML キャッシュへの読み出しアクセスがあると、アクセスされた XML キャッシュは、自身がキャッシュしている古いデータを捨て、再度データベースファイルから最新のデータをフェッチしなおすため、読み出されるデータは最新のものになる。



同一マシン上で分散 XML キャッシュを設定した場合は、クライアントからのアクセスに対して、担当が設定されたキャッシュに自動的にルーティングを行うことができる。

2.5 SonicXIS の API

SonicXIS の API は、大きく 2 種類に分かれる。1 つは DXE クライアント API と呼ばれる API で、クライアント上、例えばアプリケーションサーバ上に配置したクライアントモジュールが提供する API である。DXE クライアント API には Java API とマイクロソフト社の COM API がある。通常のデータベース操作は、この DXE クライアント API を使用して行う。

DXE クライアント API はクライアントモジュールと XML キャッシュとの間に CORBA による通信が発生する。この際にやりとりするデータのオーバーヘッドや、きめ細かいトランザクション制御を行いたいような場合には、サーバサイドの API を利用する。サーバサイド API は DXE サブレットと言い、Java API となる。DXE サブレットはいわゆる http サブレットではなく、サブレット API に準拠した処理モジュールで、XML キャッシュプロセスをコンテナとして起動、実行される。

2.6 SonicXIS のデータアクセス

SonicXIS のデータアクセス手段は、RDBMS のように SQL 言語を使用しない。検索手段は W3C の XML データのアドレッシング言語である XPath もしくは、XML への検索に関する次世代標準言語である XQuery(現在は Working Draft)を用いて行う。検索結果は XML として返却される。

更新は、XML ベースで書かれた独自の更新言語である XIS Updategram(XUG)を用いて行う。XUG は XPath でノードをアドレスし、更新内容(Insert/Replace/Delete)を指示することにより非常に簡潔に行うことができる。これにより、DOM インタフェースを使用せずに格納されている XML データに対する更新を 1 メソッドコールで行うことができる。もちろん、DOM インタフェースを用いた更新も可能である。この場合、サーバサイド API を使用して、XML キャッシュ上の DOM オブジェクトを直接編集することになる。

3. 電子カタログへの活用

上記のように、SonicXIS は複雑なデータモデルを持ち、構造変更が頻繁となるようなデータを管理するのに最適なデータベースである。そのようなケースで電子カタログは SonicXIS の特徴を最大限に利用することのできるアプリケーションの 1 つとなる。

例えば、社内購買システムのようなアプリケーションでは、文房具からパソコン、オフィス家具までを透過的に管理するようなイメージとなる。これはカテゴリが非常に複雑なツリーになることを意味する。あるカテゴリについては階層が深くなり、また、あるカテゴリでは浅い階層となる。このようなカテゴリを従来の RDBMS で管理するのは非常に困難になるが、SonicXIS では、カテゴリのツリーをそのまま格納することにより容易に管理することができる。当然、カテゴリツリーの構造変更にもオンラインで対応することができる。

また、各カテゴリで管理されるデータは管理項目がカテゴリごとに異なる。XML データで見ると、末端が微妙に異なるデータ構造となる。さらに、パソコンなど構造変更が頻繁なデータになると、サービスをストップしてデータのメンテナンスを行うことは通常データベースでは困難である。そこで、SonicXIS のダイナミックモデリング機能を使うことにより、そのようなケースに対して柔軟に対応することが容易となる。

さらに、ユーザごとに異なるビューを用意するような機能を実現したいような場合は、XML のワンソースマルチビューの機能を利用して、ユーザ独自のフィルタ(XSLT ファイル)をダイナミックに生成するような機能を実装することにより、最小限のコーディングで実現が可能となる。

電子カタログサイト等のSonicXISアプリケーションを構築するためのフレームワークとして、eCatalog Plus for XISがある。eCatalog Plus for XISは、Servlet、JSP(Java Server Pages)、JavaBeansを用いたMVCモデルで構成されるWebアプリケーション用途のPure Javaコンポーネント群で、SonicXISアプリケーション構築をクイックスタートできるようなサンプルコードである。

4. まとめ

このようなSonicXISを用いたアプリケーションのアイデアは、電子カタログだけでなく、組織というカテゴリツリーと人それぞれのデータ項目を管理するヒューマンリソースマネジメントや、文書構造をモジュール化したドキュメント管理など、多数のソリューションを提供することができる。

SonicXISのアプリケーションを考える上で重要となるのは、既存の資産をどのように活かすか、ということも一つである。例えば、既存のRDBMSのシステムがあり、そこにXMLの特徴を活かした機能を追加するような場合、RDBMSをすべてリプレースするのではなく、RDBMSの長所とSonicXISの長所をそれぞれ活かしたハイブリッドな構成とすること、などである。例えばマスターデータや在庫データなどの固定的なスキーマを持つデータに関してはそのままRDBMSを用い、製品スペックなど、ダイナミックに構造が変わるようなデータに関してはSonicXISを作る、などのアプローチである。

このように、SonicXISは、XMLのメリットを最大限に活かし、従来のアプローチでは不可能だったアプリケーションを、機能だけではなく運用・管理のフェーズにまで提供できる唯一のデータベースであると言える。