2D-7

XML データ同時部分更新を可能にするロック委譲システムの提案

北川　亘†　　瀬田　賢治†　　橋本　光治†

日本アイ・ビー・エム株式会社†

1．はじめに

昨今の電子データ化の流れやネットワークの高速化を受けて，XML データの更新頻度，速度とも高速化が必須となってきている．このような現状において，XML データを用いたソリューションの中では，1 つの XML データを，複数のサブシステムで同時に平行して更新処理をおこなう必要性が出てきている．そこで，1 つの XML データの整合性を維持しつつ，並列処理による同時部分更新を可能とする，XML データの要素単位での部分ロック委譲のシステムを発案した．そのシステムにおける仕組み及び，有効性について述べる．

2．XML データ同時更新における問題点

2.1 データの整合性

XML 形式で示されるデータは，常に整合性を保たなければならない．

XML では，任意に独自のタグを定義していくことで，各 XML でのソリューションに用いられる XML に対する DTD を作成するため，より複雑化，煩雑化すればするほど XML のデータ量は肥大し，XML データの動的変更を余儀なくされる場合が存在しうる[1]．その変更された XML データは常に DTD に準拠しなければならない．

2.2 動的な部分更新

XML データの部分的な更新を行う際は，基本的に，XML が保管されているシステムのリポジトリから，対象となる XML 全体を一旦取り出して，変更，更新，置換したい箇所を，変更した後，再度システム内のリポジトリに保管することがほとんどである．

しかし，この手法では，XML データのある一部分を変更するだけでも，その対象となる XML 全体をロックした上で，部分的な更新を行わざるを得ない．

2.3 XML の階層構造

XML データは，階層構造を持っており，部分的な XML データの更新を行う場合，その階層構造の整合性を崩すことは出来ない．そのため，2.2 節で述べたように，今までの手法では，更新対象 XML 全体に対してのロックを取得し，その上で部分的な更新を行う必要がある．つまり，単純に各階層単位で部分的にロックを取得したとしても，図 1 に示すように，ユーザー 1 が第 2 階層 B をロックして，第 2 階層 B 内，第 3 階層 A を更新した場合，同時にユーザー 2 が第 3 階層 A をロックして，第 3 階層 A を更新する可能性が考えられる．この場合，XML データとして不整合が生じてしまい，同時更新が不可能である．

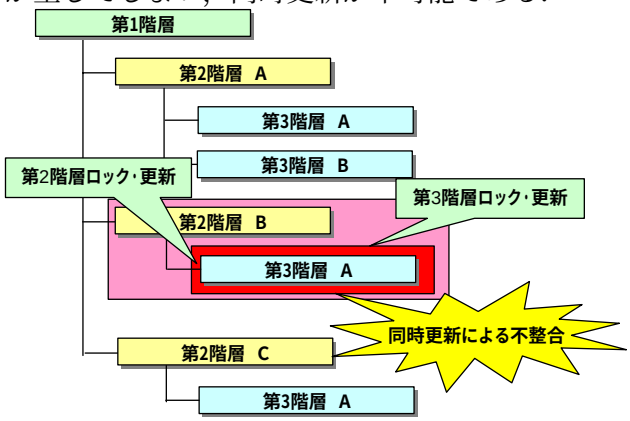

図 1　XML 階層の同時更新による不整合例

3．ロック委譲を含んだシステム構成例

3.1 ロックユーザー管理

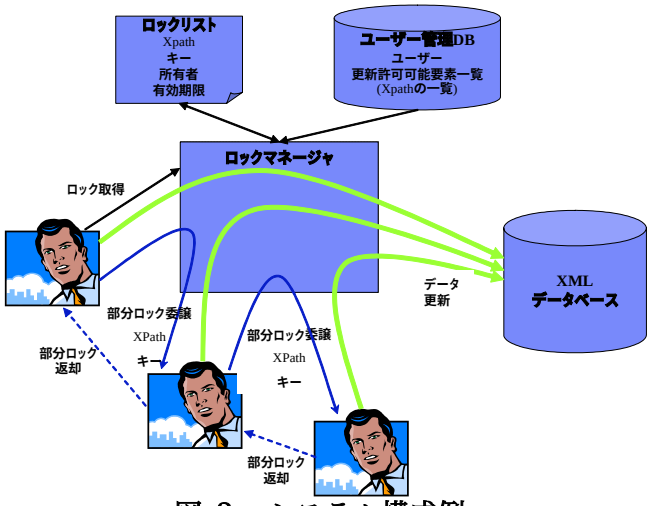

図 2　システム構成例

システム構成例を図 2 に示す．ロックの委譲システムは主に次の 3 つのコンポーネントから構成されることを想定する

Lock delegation technique for parallel updating of partial XML data by multiple users

†Wataru Kitagawa, Information Management Technology Software Development Lab., IBM Japan

a. ロックユーザー管理
b. ロックリスト
c. ロックマネージャー
まず，a.ロックユーザー管理について，以下に説明する．ロックユーザー管理は，全ユーザーまたはロック管理者のユーザーを管理し，各ユーザーに対して，ロック可能な XPath（XML のポインタ）一覧を管理するリポジトリである．つまり，
あるユーザーは，どの XML の部分ロックを取得することが可能かという要素一覧を示す機能をもつ．

3.2 ロックリスト

ロックリストの役割は，あるユーザーがロックを取得した際，そのユーザーがロックを持つ XML の XPath を持つ．さらに，XPath がその XML 内で，一意に特定できない場合でのキーと，ロック所有者，及びそのロックに対しての有効期限などを持つ．あるユーザーが，ロックを取得した場合，ロックマネージャーを通し，ロックリストに，ロックした XPath 及び，キー情報，有効期限を付与する．これにより，現状のロック状態を確認することが可能である．また，ロックが返却された場合は，このロックリストから，返却されたロックの XPath，キー情報，有効期限は削除される．

3.3 ロックマネージャー

ロックマネージャーの役割は，ユーザーとユーザー管理，ロックリスト，を制御するコンポーンネントとなる．ユーザーは全て，このロックマネージャーを通し，ロックを取得する．
前節でも説明したように，あるユーザーがロックを取得する際，ロックマネージャーに対し，ロック依頼を行う．この際，ロックマネージャーは，ユーザー管理リポジトリに問い合わせ，このユーザーがロックしたい XML に対し，ロック可能なユーザーかどうか，また，その XPath に対してもロック可能かどうかの確認を行う．もし，ロック可能なユーザーであれば，ロックリストに，ロックした XPath，キー情報，ロック所有者（ユーザー名），有効期限を書き込む．もし，このユーザーがユーザー管理リポジトリ上で，ロック不可な XML，XPath を指定した場合は，ロックマネージャーがそのユーザーに対しエラーを返す．

４．部分ロック委譲の仕組み

4.1 ロックの委譲

XML データの同時更新を実現するために，ロック権限を委譲する仕組みを提案する．通常，データベースでは，SQL の GRANT 文などを使用して，ロックを含めた様々な権限を付与する事が可能であるが，筆者らが提案する概念は，付与ではなく委譲である．つまり，ロック権限を取得した XML データに対して，その中の部分的なロックを他者に譲ることが出来る．
また，委譲した部分的なロックを必ず返却してもらうことで，更新やコミットを行うのは，ロック権限取得者となり，最終的なデータ全体の整合性は保障される．これにより，DTD の整合性を崩すことなく XML データの更新が可能となる．

4.2 更新権限の失効と更新拒否

権限を委譲した上階層のロック所有者は，委譲した XML 要素以下の XML 要素に対する更新権限を失うが，委譲部分の更新を拒否する権限を持つことで動的な更新が可能である．

3.3 委譲ロック権限所有者

権限を委譲された下階層ロック所有者は，データの更新，権限の返却，子階層以下の更新権限の他者への委譲が可能となる
委譲する XML 要素及びその子階層の指定には XPath など XML 要素を特定できるもの使用し，必要であれば，同一の要素名をもつ要素の特定のためにキー情報を使用する．よって，階層構造の整合性は保たれる．

５．結論

ロック委譲の技術を提案することにより，1 つの XML データの整合性を維持しつつ，並列作業による同時部分更新を可能とする，XML の要素単位での作業高効率化を可能とした．これは，XML データの部分更新のみならず，並行処理におけるパフォーマンスの向上にもつながる上，XML の DTD スキーマでの整合性も保障される．さらには，要素での階層の上下関係によるデータレベルでの整合性も保障し，部分更新担当者と責任者の分離によるコンテンツレベルでの整合性も保障される．また，部分更新記録などを使うことによる，XML データの更新差分取得の可能性も示唆でき，XML データのレプリケーションの高速化などに応用できる．また，この技術は，特許として開示を行った．

参考文献

[1] 橋本 光治，瀬田 賢治，渡部 敏雄，北川 亘，「素材管理データベースの XML 部分更新によるパフォーマンス向上」，情報処理学会 FIT2005，2005