

論文内容の要旨

博士論文題目

Making XML Database Systems Scalable to Computer Resources and Data Volumes
(計算リソースとデータ量にスケーラブルな XML データベースの実現)

情報科学研究科情報生命科学専攻データベース学講座 油井誠

(論文内容の要旨)

XMLの利用の拡大により多量のXMLデータを効率的に管理することが求められている。データベースの利用者がデータベースに求める要件のうち、最も重要となる項目の一つが、データベースの計算リソースやデータ量に対して得られる性能のスケーラビリティである。本研究では、(a)データ量に対して得られる性能のスケーラビリティ、(b)無共有型計算機クラスタを利用したスケーラビリティ、(c)共有メモリ型並列計算機を利用したスケーラビリティの三つの観点で、それぞれXMLデータベースのスケーラビリティ向上のための手法を提案する。(a)のデータ量に対して得られる性能のスケーラビリティでは、表形式で内部表現されたXML文書をブロック化して二次記憶に文書順に配置し、問合せ実行中に必要なブロックを効率的に主記憶に読み込む機能の特徴とするXML問合せ処理手法を開発した。実験結果から提案手法がデータサイズに対してほぼ線形の性能が得られることを示した。(b)の無共有型計算機クラスタを利用したスケーラビリティでは、リアルタイム性が要求されるXML問合せに対して、計算機クラスタを利用した分割統治を行う場合を議論する。これまでの分散XMLデータ処理で常であった値渡しによるデータ交換には中間計算結果のシーケンスの一部の要素のみが利用される場合に不要な通信や計算が発生する問題や、オペレータ間の並列性が阻害されるという問題がある。この問題への解決策として、遅延評価を活かした参照渡しに基づく分散XML問合せ処理手法を開発し、実験結果により分散XMLデータベースシステムが参照渡しによるデータ交換を考慮に入れることの重要性を実証した。(c)の共有メモリ型並列計算機を利用したスケーラビリティでは、メニーコア計算機上でデータベース管理システムを利用する上での問題点として、バッファ管理モジュールの並行アクセス耐性の低さを取り上げ、ノンブロッキングのバッファ管理手法を提案する。64プロセッサを備えるSun UltraSPARC T2を利用した実験により、既存のバッファ管理手法が16プロセッサ以上のスケールしないのに対して、提案するバッファ管理手法が64プロセッサまで、プロセッサ数に応じた線形の性能が得られることを示した。最後に三つの提案手法を組合せてXMLネイティブデータベースシステム上に実現する上で得られた知見を述べる。

(論文審査結果の要旨)

本論文はスケーラブルな XML データ管理システムを実現するための手法について体系立てて述べている。XML の利用拡大により、科学データや Web 上の RSS/ATOM データなど大量の XML データを効率的に管理するデータ管理システムが求められている。データ管理システムにおいて、ユーザが最も重視する項目の一つが、計算リソースやデータ量に対して得られる性能のスケーラビリティである。本研究は、XML データベースのスケーラビリティについて、1) データ量に対して得られる性能のスケーラビリティ、2) 無共有型計算機クラスタを利用したスケーラビリティ、3) 共有メモリ型並列計算機を利用したスケーラビリティの3つの観点で、それぞれスケーラビリティ向上のための手法を提案している。本論文の成果は以下の3点に要約される。

1. XML をブロック化してハードディスク上に配置する上で効果的な手法を議論し、XML 文書を文書順に沿ってブロック化し、ディスク上に文書順にブロック配置することが XML データに対するアクセスパターンにおいて有効であることを示している。その上で、XML 問合せのデータアクセスパターンに適した文書順のデータ格納手法を提案している。従来、XML の部分木単位のデータ格納方法が有効であると見なされていたが、その常識が当てはまらない場合を明らかにしたという点で意義がある。

2. 分散 XML 問合せ処理において欠かすことのできない問合せ処理ノード間のデータ交換について議論している。従来の分散 XML 処理では値渡しによるデータ交換が主に行われてきたが、本論文では値渡しの問題点を挙げ、参照渡しを用いることの重要性を論じている。提案手法を用いたパイプライン処理によって値渡しでは避けることができないバースト処理によるリソースの枯渇を防ぐことができること、提案手法のオーバーヘッドが十分に低く応用範囲が広いことが実験により示されている。

3. ロックフリーのバッファ管理手法として、プロセッサ数に対する高いスケーラビリティを実現する Nb-GCLOCK を提案している。本論文は、バッファ管理における並行性の問題を明らかにし、その解決策の1つとしてノンブロッキングデータ構造をデータベースのバッファ管理に導入している。既存手法が 16 プロセッサ付近に CPU スケーラビリティの限界があるのに対して、提案手法は 64 プロセッサまでほぼ線形の性能があることが評価実験により確認されている。本提案は、バッファ管理に CPU の原子命令を利用したノンブロッキング同期を採用する初めての試みである。マルチプロセッサが急速に普及している現状を踏まえた時宜を得た研究であり、新規性、有用性、信頼性ともに不足ない。

論文の主要部分に相当する内容は、既に学術論文誌に掲載されると共に、国際会議においても公表されている。以上のように意義深い成果が得られていることから、本論文を博士(工学)の学位論文として価値あるものと認める。