数十テラバイトのXMLデータを高速検索できるXMLデータベースTX1$_{TM}$の分散並列検索技術

Distributed Parallel Search Function of TX1$_{TM}$ XML Database Realizing High-Speed Searching of Tens of Terabytes of XML Data

幸田　和久　　田中　　雅
■ KODA Kazuhisa　■ TANAKA Satoshi

東芝ソリューション(株)は，非定型データの効率的な管理のために，XML (Extensible Markup Language) データをそのまま格納でき，テラ (T：10^{12}) バイト級のXMLデータでも高速検索できるXMLデータベースTX1を2005年に商品化した。その後，企業内で扱う情報量は急速に増え，より大容量のXMLデータに対しても高い検索性能が求められるようになった。

このニーズに応えるため，TX1では分散並列検索技術DPS (Distributed Parallel Search) を開発し，従来は難しかった数十Tバイト級の大容量XMLデータに対しても高速検索を実現した。また，クラスタソフトウェアDNCWARE ClusterPerfect$_{TM}$ EXと連携し，DPSシステムの効率的で安定した稼働を実現した。

Toshiba Solutions Corporation released the TX1 extensible markup language (XML) database in 2005, which can retrieve data from large volumes of XML data of several terabytes in size for efficient management of semistructured data. With the rapid proliferation of business databases in recent years, demand has been growing for enhancement of retrieval performance with larger volumes of XML data.

In response to this situation, we have developed a distributed parallel search (DPS) function for the TX1 that makes it possible to search large-volume XML data of tens of terabytes in size at high speed. Experiments on searching speed confirmed that the DPS system achieves efficient and stable operation linked with DNCWARE ClusterPerfect$_{TM}$ EX (CPEX) integrated cluster software.

1　まえがき

企業情報の約8割を占めると言われる非定型データのフォーマットとして，柔軟なデータ構造を持つXML (Extensible Markup Language) の利用が進んでいる。このXMLデータを効率的に蓄積，検索，管理できるようにするため，東芝ソリューション(株)は，多種多様な階層構造を持つXMLデータをそのまま格納でき，Tバイト級のデータ量でも高速検索できるネイティブXMLデータベースTX1[(1), (2)]を2005年に商品化した。TX1は，規程文書管理公開システムや保守情報統合検索システムなど，企業情報を効率的に利活用するシステムの基盤として用いられている。

一方，近年，ビジネスの拡大や企業間の経営統合，日本版SOX法（金融商品取引法）の施行による内部統制の導入などを背景に，企業内で扱う非定型データの量は急速に増えており，より大容量のデータに対する高い検索性能が求められている。当社はこのニーズに応えるため，XMLデータを分散配置した複数のTX1に対して並列検索を行う分散並列検索技術DPS (Distributed Parallel Search) を開発し，従来1台のTX1では難しかった数十Tバイト級の大容量XMLデータに対する高速検索を実現した。

ここでは，DPSの高速検索技術と，クラスタソフトウェアDNCWARE ClusterPerfect EX (以下，CPEXと呼ぶ) との連携によりDPSシステムの効率的で安定した稼働を実現するための技術について述べる。

2　DPS

DPSシステムの概要とDPSを構成する技術について述べる。

2.1　DPSシステムの概要

システム構成を**図1**に示す。DPSシステムは，大容量XMLデータを分割して管理する複数のTX1サーバと，TX1サーバに対する検索処理全体を取りまとめるTX1コーディネータから構成される。TX1コーディネータは，複数のTX1サーバと連携してアプリケーションソフトウェアからの検索要求を処理する。DPSシステムに対する問合せ言語として，W3C (World Wide Web Consortium) で標準化されたXQuery (An XML Query Language)[(3)]の一部であるXPath2.0を提供している。XPathとは，XMLデータの特定の要素を指し示す構文であり，アプリケーションソフトウェアはXPathを利用して，複数のTX1サーバが管理するXMLデータベース群を，あたかも一つのXMLデータベースであるかのように，透過的に検索することができる。

DPSシステムでは大容量データを複数のTX1サーバで分散配置するが，データ量のいっそうの増加には，TX1サーバの増設（スケールアウト）で対応できる。DPSシステムのデータ管理にはTX1サーバ間でデータの共有は行わないシェアドナッシング型を採用している。各TX1サーバは独立して検索

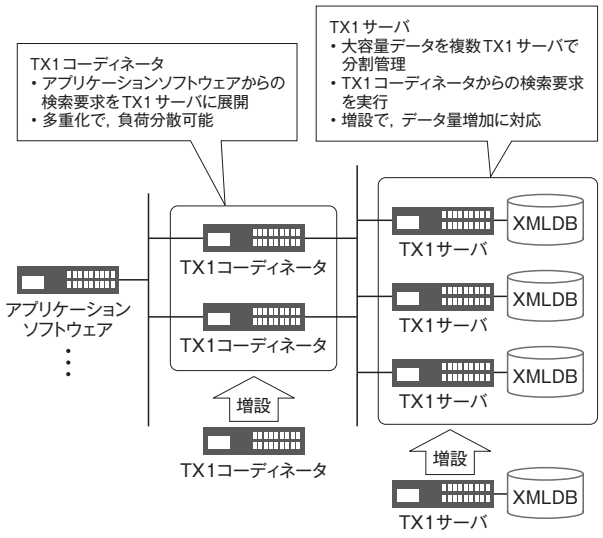


図1. DPSシステムの構成 — DPSシステムは，大容量XMLデータを分割して管理する複数のTX1サーバと，TX1サーバに対する検索処理全体を取りまとめるTX1コーディネータから構成される。

Configuration of DPS system

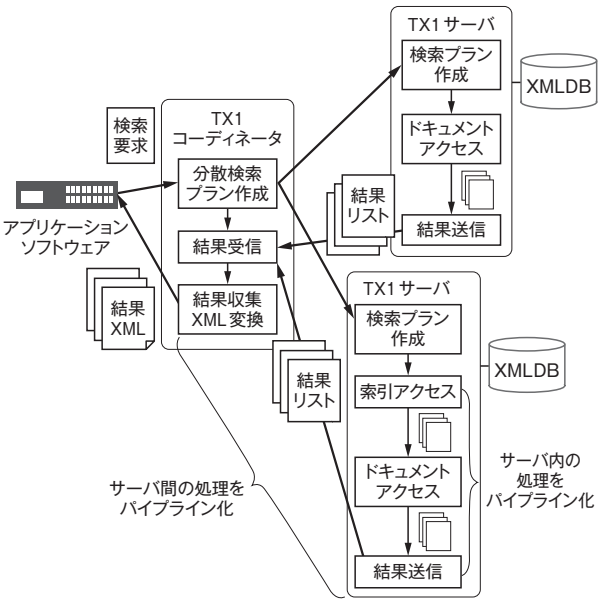


図2. 検索処理の一例 — TX1コーディネータは，複数のTX1サーバと連携してアプリケーションからの検索要求を処理する。

Example of query processing flow

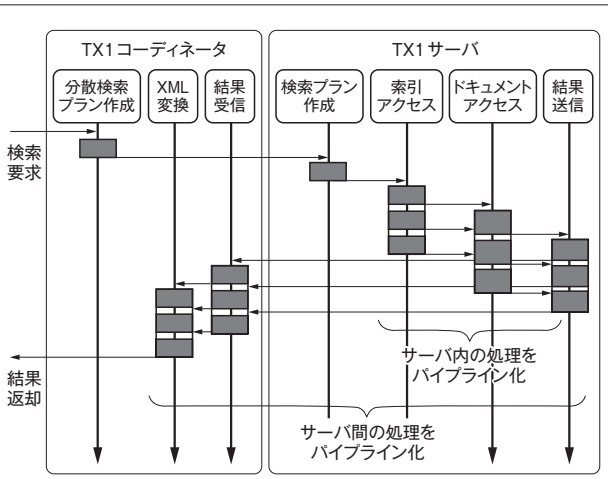


図3. 検索処理のシーケンス図 — 検索の各処理はパイプラインで処理される。

Sequence diagram of query processing

処理を行うことができるため，検索性能を低下させずにスケールアウトできる。

一方，アプリケーションソフトウェアからの同時アクセス数の増加によりTX1コーディネータが高負荷となった場合には，TX1コーディネータを増設することで負荷を分散できる。

2.2 分散並列処理による検索処理

DPSでは，アプリケーションソフトウェアからの検索要求を，各サーバで並列処理することで，高い検索性能を達成している。以下，検索処理の流れについて述べる（**図2**）。

(1) TX1コーディネータは，アプリケーションソフトウェアからの検索要求を受け付けると，各TX1サーバとの送受信処理や結果収集処理などを含めた検索処理全体の実行計画（分散検索プラン）を作成し，複数のTX1サーバに対して並列に検索要求を送信する。

(2) 各TX1サーバは，TX1コーディネータからの検索要求に対して，データベースに格納されたXMLデータの構造情報や索引の設定状況に応じて最適な検索プランを作成する。そのプランに沿って索引やデータへのアクセスなどの処理を実行する。

(3) 各TX1サーバは，(2)の検索処理によって作成した結果リストを順次TX1コーディネータに転送し，TX1コーディネータは受信した結果リストを処理する。

TX1コーディネータ及びTX1サーバで実行されるこれらの各処理は，前段処理の出力を次段処理の入力とするパイプライン処理で動作する。このパイプライン処理のシーケンスを**図3**に示す。ここではTX1コーディネータ1台，TX1サーバ1台の構成での検索処理を例として挙げる。

TX1コーディネータは，分散検索プランを作成しTX1サーバに検索を出す。TX1サーバは，各々で最適な検索プランを作成する。図3のTX1サーバの検索処理例では，索引アクセス処理に引き続きドキュメントアクセスを行う検索プランが採用されている。索引アクセス部は最初の数件分の処理が終わった時点で，ドキュメントアクセス部に結果を出力する。その結果を入力としたドキュメントアクセス部の処理と並行して，索引アクセス部は次の数件分を処理する。

また，TX1サーバは，索引アクセス部とドキュメントアクセス部によって作成する一定件数ごとの結果リストを，TX1コーディネータに順次送信する。TX1コーディネータでは，受信した結果リスト順に，検索結果となるXMLデータを作成する。このように，サーバ内とサーバ間で実行する処理をパイプライン化することで，効率的な並列処理による高い検索性能を実現した。

2.3 データ圧縮技術

DPSの検索処理の高速化には，通信データ量の削減によるデータ転送時間の短縮が不可欠である。例えば，サイズの大きなXMLデータや，XML文書の大量取得を行うと，TX1サーバとTX1コーディネータ間で大量の通信データが発生する。また，今後のXQueryのサポート範囲の拡大により，“TX1コーディネータが，あるTX1サーバから取得した中間結果を利用して，再度別のTX1サーバに検索要求する”などの検索処理が可能になることで，通信データ量の大幅増加が見込まれている。

この通信データ量を削減するには通信データを圧縮する方法がある。しかし，一般的に圧縮解凍速度と圧縮率にはトレードオフ（二律背反）の関係があり，高い圧縮率でデータ圧縮することで通信時間を短縮できても，データ圧縮処理時間で相殺されてしまい，データ転送時間全体の短縮にはつながらない。この課題を解決するために，DPSでは，圧縮解凍速度重視の独自のアルゴリズムの採用と通信データ表現の改良によって，圧縮解凍速度と圧縮率の両方を向上させ，データ転送時間の短縮を実現した。

2.4 検索結果先着方式

DPSでは，XMLデータを複数のTX1サーバで分割管理するシェアドナッシング方式を採用しているので，検索漏れをなくすためには，TX1コーディネータはすべてのTX1サーバからの検索結果を待つ必要がある。一方，検索条件の要素や属性の値を用いてXMLデータを分割配置し，その要素や属性の値で一致検索するケースでは，検索結果を返却するのは必ず一つのTX1サーバとなる。

このようなケースに効率よく対応するために，TX1コーディネータがいずれかのTX1サーバから検索結果として“1件以上ヒット”を受け取った時点で，そのTX1サーバに検索結果がすべてあるとみなし，検索結果を，そのTX1サーバからだけ取得する“検索結果先着方式”を提供する。

DPSシステムを構成するTX1サーバ数が多くなると，ネットワーク遅延などの様々な要因によりTX1サーバの応答性能にばらつきが発生し，検索時間が増加することがある。しかし，検索結果先着方式を利用すれば，検索結果を持つ一つのTX1サーバがTX1コーディネータに応答した時点で検索は終了するので，ばらつきの影響を抑え検索時間を短縮できる。

3 DPSシステムの効率的で安定した稼働を実現する技術

DPSシステムは，高い検索性能を実現するために複数のTX1サーバから成る分散構成をとる。分散構成は，多くのコンピュータやソフトウェアから構成されるため，一般的に管理負荷が増大する。この課題を解決するために，CPEXと連携しDPSシステムの効率的で安定した稼働を実現した。

3.1 CPEXによるDPSシステムの運用管理

CPEXは，大規模な分散システムにおいて，システム管理者に対してサーバの起動や，停止，監視などの機能を提供するHA（High Availability）クラスタソフトウェアである。**図4**に示すように，複数台のコンピュータで構成されるDPSシステムの状態や，各サーバの起動及び停止を一元的に管理する。システム管理者は，CPEXが提供するサーバの起動や，停止，監視などの機能を使うことで，DPSシステムの運用管理が容易になる。例えば，DPSシステムを起動するには，すべてのTX1サーバが起動した後にTX1コーディネータを起動する必要があるが，事前にこの動作をCPEXに設定することで，ユーザーは各サーバの起動順序まで意識することなくDPSシステムを起動することができる。また，DPSシステムの状態もCPEXのユーザーインタフェースで一括して確認できるため，効率的なサーバの運用管理ができる。

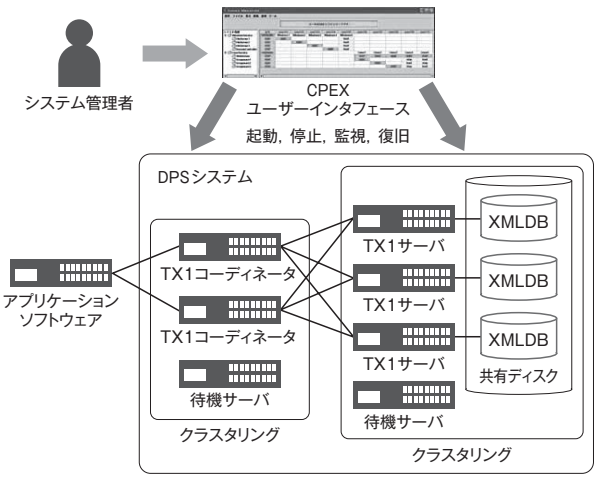


図4．CPEXによるDPSシステムの運用管理 — DPSシステムをCPEXのクラスタシステムとして構築する。

Management of DPS system by CPEX

3.2 障害復旧処理

DPSシステムにおいてCPEXは，TX1コーディネータ，TX1サーバ，及びネットワークを監視し，障害を検知すると，その検索サービスを待機系サーバへ自動的に引き継ぐことで，DPSシステムの稼働率を高めることができる。ただし，TX1コー

ディネータ，TX1サーバの検索処理はパイプライン化されているため，障害が発生したときに実行中の検索は待機サーバに引き継がずに，DPSシステム全体で検索の中止処理を行う。障害からの復旧の流れについて以下に述べる。

(1) TX1サーバで障害が発生した際の復旧処理

(a) CPEXは，あるTX1サーバでの障害を検知し，TX1コーディネータへ通知する。

(b) CPEXは障害が発生したTX1サーバの検索サービスを，待機系サーバへフェールオーバ（障害発生時の機能引継ぎ）する。

(c) TX1コーディネータは，自身が現在実行中の検索に対して中止処理を行う。

(d) TX1コーディネータはすべてのTX1サーバに検索中止を指示する。

(2) TX1コーディネータで障害が発生した際の復旧処理

(a) CPEXは，TX1コーディネータでの障害を検知し，TX1サーバへ通知する。

(b) CPEXは障害が発生したTX1コーディネータの検索サービスを，待機系サーバへフェールオーバする。

(c) TX1サーバは，自身が現在実行中の検索に対して中止処理を行う。

この処理の中でポイントとなるのは，障害発生時に実行されていた検索の中止処理完了を待たずに，CPEXによる待機サーバへのフェールオーバができる点である。したがって，障害発生時に実行されていた検索の中止処理が，システム復旧時間に影響することはない。

4 大容量XMLデータの検索性能

DPSの検索性能を評価するために，2種類の検索時間の測定を行った。

(1) 1Tバイト以上のXMLデータに対する検索性能

6,000万件（2.5Tバイト）のXMLデータを，TX1サーバ4台に1,500万件（0.6Tバイト）ずつ分散配置し，検索時間の測定を行った結果を**図5**に示す。

DSPにより，6,000万件（2.5Tバイト）を超える大容量XMLデータに対して高速検索を実現することができた。

(2) TX1サーバ増設による検索性能（拡張性）　TX1サーバ1台に130万件（3Gバイト）のXMLデータを格納し，順次TX1サーバを増設して検索対象となるデータ件数を5,200万件（120Gバイト）まで増やした場合の，検索時間を測定した結果を**図6**に示す。検索式には，検索対象が増加しても一定件数（60件）該当するような文書ID（Identification）検索を利用した。TX1サーバ台数を40台まで増設しても，検索時間には劣化は見られず，高い拡張性の確保が実現できた。

◆文書IDによる検索（94件該当）
- 検索時間：100 ms
- 検索式：
db("test")/test/document[header/Id[@root='1a2b3c4d-5e6f-7g8h-9i0j-1k2l-3m4n5o6p7q8r']]

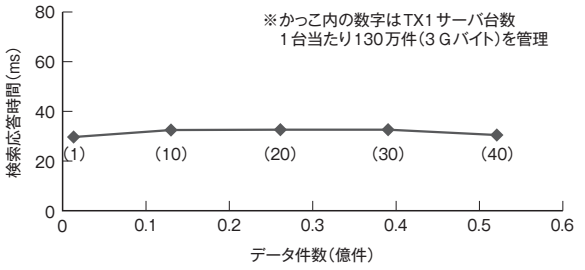
◆キーワード検索でヒットした文書の文書IDによる検索（160件該当）
- 検索時間：160 ms
- 検索式：
db("test")/test/document[header/Id[@root=
db("test")/test/document[body/contents/keyword=
"東芝20090101"]/header/Id/@root]

ハードウェア仕様

項　目		仕　様	台　数
コーディネータ	CPU	Quad-Core AMD Opteron™(注1) 2.4 GHz×2	1
	メモリ	128Gバイト	
TX1サーバ	CPU	6-Core Intel® Xeon®(注2) 2.4 GHz×2	4
	メモリ	128Gバイト	
	ストレージ	AF2500	

図5. 検索性能測定結果(1)：1Tバイト以上のXMLデータに対する検索性能 — 測定に使用した検索式とその結果である。DSPは，6,000万件（2.5Tバイト）のXMLデータに対して高速に検索できる。

Results of retrieval performance measurement (1)



ハードウェア仕様

項　目		仕　様	台　数
コーディネータ	CPU	Dual-Core Intel® Xeon® 2.33 GHz×2	1
	メモリ	16Gバイト	
TX1サーバ	CPU	Intel® Core™(注3) 2Duo 2.66 GHz	40
	メモリ	2Gバイト	
	ストレージ	内蔵ディスク	

図6. 検索性能測定結果(2)：TX1サーバ（PCサーバ）増設による検索性能（拡張性）— TX1サーバを増設した場合の測定結果である。DPSは，TX1サーバを40台増設しても検索性能は劣化しない。

Results of retrieval performance measurement (2)

これら二つの測定結果から，TX1サーバ40台にXMLデータを1,500万件（0.6Tバイト）ずつ分散配置し，文書ID検索で一定件数（160件）該当した場合の検索時間を外挿した結

(注1) Opteronは，Advanced Micro Device, Inc.（AMD）の商標。
(注2) Intel，Xeonは，米国及びその他の国における米国Intel Corporation又は子会社の登録商標又は商標。
(注3) Coreは，米国及びその他の国における米国Intel Corporation又は子会社の登録商標又は商標。

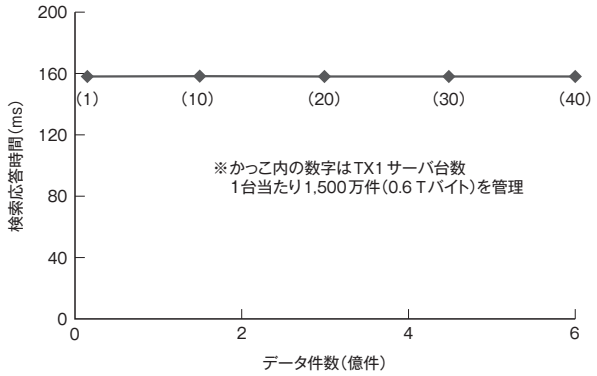


図7. データ件数と検索性能 — 検索性能の測定結果(1), (2)から外挿した結果である。DPSは，6億件（25 Tバイト）のXMLデータでも安定した検索性能を確保できる。

Relationship between volume of data and retrieval performance

果を**図7**に示す。DPSでは，XMLデータ件数の増加に合わせてTX1サーバを増設することで，6億件（25 Tバイト）のXMLデータでも安定した検索性能を確保することができる。

5　あとがき

DPSによって，従来のTX1では難しかった数十Tバイト級の大容量XMLデータの実用的な検索性能を実現した。また，CPEXと連携しDPSシステムの効率的で安定した稼働を実現した。これにより，大容量XMLデータを扱う大企業の情報検索システムの基盤としてTX1の適用ができるようになった。

今後は，複数のデータベースの更新同期を実現する2相コミットメントやデータの分散配置を支援するデータパーティションなどの分散データベース機能の強化や，XQueryによる更新を実現するXUF（XQuery Update Facility）[4]のサポートなどに取り組んでいく。また，DPSの検索機能に関しても，XQueryのサポート範囲を拡大し，複数TX1サーバからの検索結果の結合や集計など，より高度な検索機能を提供し，いっそうの適用領域拡大を目指す。

文　献

(1) 服部雅一，ほか．高速性と信頼性を両立したコンテンツ管理向けネイティブXMLデータベース．東芝レビュー．**59**，2，2004，p.54－57.
(2) 谷川　均，ほか．大規模でも高速な検索を実現するXMLデータベースTX1．東芝レビュー．**60**，7，2005，p.71－75.
(3) World Wide Web Consortiumホームページ．“XQuery 1.0: An XML Query Language”．<http://www.w3.org/TR/xquery/>，（参照2009-08-31）.
(4) World Wide Web Consortiumホームページ．“XQuery Update Facility 1.0”．<http://www.w3.org/TR/xqupdate/>，（参照2009-08-31）.
(5) 服部雅一．巨大XMLデータを管理・検索する分散XMLデータベース．東芝レビュー．**62**，10，2007，p.62－63.
(6) 服部雅一，ほか．巨大XMLデータを管理し検索できる分散XMLデータベース．東芝レビュー．**64**，4，2009，p.56－59.

幸田　和久　KODA Kazuhisa
東芝ソリューション（株）プラットフォームソリューション事業部ソフトウェア開発部主任。XMLデータベースTX1の開発に従事。
Toshiba Solutions Corp.

田中　　雅　TANAKA Satoshi
東芝ソリューション（株）プラットフォームソリューション事業部ソフトウェア開発部主任。クラスタ及びクラウドの開発に従事。
Toshiba Solutions Corp.