

# 分散データベースにおける 透過的データ利用環境の実現

植田和憲\*, 神矢健一

\*高知工科大学 工学部  
〒 782-8502 高知県香美市土佐山田町宮ノ口 185

E-mail: \*ueda.kazunori@kochi-tech.ac.jp

要約：学際的な研究プロジェクトを遂行するにあたり、他機関の研究者と協力して研究を進める場合がある。このような場合、各研究者が保持する情報は物理的に分散しており、かつ、研究領域によってはその量が膨大となることがある。情報ネットワーク技術が発達し、遠隔地の情報を活用できるようになってきているとはいえ、現状では膨大なデータを拠点間で転送するにはかなりの時間を要する。そこで、本研究グループでは、分散した拠点に散在するデータを透過的に扱うための枠組みを提案する。ここでの透過的とは、データが物理的に分散していることを意識せずに済むことを指す。

## 1. はじめに

学際的な研究プロジェクトでは、各研究機関に存在する研究用データからなるデータベースを構築することがある。このデータベースには、テキストファイルといった数KBのデータや三次元画像や映像といった数MBから数GBものデータなどが含まれる。実際には、このデータベースに含まれるデータは各拠点に存在するため、統合的に利用するためには分散データベースシステムを構築する必要がある。

本研究の目的は、高知工科大学総合研究所において行われているプロジェクトの一環として構築される分散データベースにおける大容量データの転送時間に関する問題を解決し、データベース利用者に対し透過的なデータ利用環境を提供することである。ここで言う透過的な利用環境とは、集中データベースに近い使用感が得られるような環境のことであり、本研究では、データの転送時間を短縮による実現を目指す。

## 2. 分散データベースシステムの透過性

本研究では、分散データベースシステムにおける透過的なデータ利用環境の実現を目的としている。分散データベースとはネットワークを介して物理的に分散した複数の集中データベースを接続

し、ユーザにはあたかもひとつの大きな集中データベースであるかのように利用できるようにしたものである。ユーザにデータベースの物理的分散を意識させず、集中データベースであるかのように利用できることを透過的であるといい、分散データベースシステムは透過性を実現するためにいくつかの機能を提供する。分散データベースシステムが提供する透過性の種類としては、位置に対する透過性、移動に対する透過性、分割に対する透過性などがある<sup>1)</sup>。

位置に対する透過性とは、データベースがネットワークにより接続されて物理的に分散していることを、ユーザに意識させないで利用可能とすることである。移動に対する透過性とは、運用の都合や性能向上の目的で分散されたデータや表の格納サイトが変更した後も、ユーザにそれらの移動を意識させることなく業務プログラムや操作手順を変更しなくても、移動したデータにアクセス可能とすることである。分割に対する透過性とは、一つとして表現されるデータが複数のデータベースサイトに分割して格納されていても、ユーザに意識させることなく利用可能とすることである。分散データベースが提供する透過性の種類には、さらに、重複に対する透過性、障害に対する透過性、データモデルに対する透過性などがある。

いずれの透過性も「ユーザがデータおよびデータベースの分散を意識しなくてもよい」ように利用できる環境を目指しており、分散データベースシステムにおける透過性を実現することは非常に重要である。

### 3. データ取得時間の短縮

分散データベースにおけるデータ取得時間を短縮することにより、ユーザにはデータベースが物理的に分散していることを意識させないようにする。これは透過的にデータを利用できる分散データベース環境を実現したといえる。具体的には一度使用したデータを手元に保存、また近い将来に参照すると予想されるデータをあらかじめダウンロードしておくことによって、ユーザのデータ取得時間の短縮を図る。先読みとキャッシュという技術を併用することで透過的環境を実現できると考えられる。

先読みとは、ユーザがネットワークを使用していないうちに、近い将来アクセスが予想されるデータをあらかじめダウンロードしておく技術である。先読みをしておくことで、ユーザのデータへのアクセス時間を短縮することができる。先読みしたデータはキャッシュに保存しておく。ユーザがアクセスを要求したデータがすでに先読みされていれば、ネットワークを使ってダウンロードする時間が不要なく、すぐにそのデータにアクセスできる。本研究においては、参照順の傾向を指定しておき、その指示に従って「ユーザが使用中のデータに関連し、且つ近い将来にアクセスが予想されるデータ」をネットワークのアイドル時間を利用して先行ダウンロードしておくものである。

キャッシュとは、一度使用したデータを、データを使用する場所に近い高速な記憶装置に保存しておくことで、二回目以降のデータへのアクセス時間を短縮する技術、または装置のことである。キャッシュはさまざまな場所で使われており、典型的な例がWEBで使用されているキャッシュ技術である。WEBブラウザは一度ダウンロードしたページのデータをハードディスクに保存し、二回目以降のサーバアクセスを高速化している。プロキシサーバでもキャッシュ技術が使用されており、一度アクセスしたページをサーバ内のハードディスクに保存し、二回目以降はインターネットにアクセスせずにページデータを渡すことができる。本研究においては、一度使用したデータを、ユーザが初めに接続するコンピュータ(キャッシュサーバ)に一時的に保存しておくことである。

## 4. 透過的データ利用環境の実現

### 4.1 システム設計

提案システムでは、拠点間データ転送を管理することで透過的なデータ利用環境を提供する。具体的には、先読みとキャッシュによって、データの利用時の転送を回避する。図1に全体図を示す。拠点ごとに存在するストレージを分散データベースとみなし、データ利用者は拠点内ストレージを介し分散データベース内のすべてのデータへアクセスする。このとき、利用するデータの実体ができるだけ拠点内ストレージへ存在している状態を確保するように、アクセスしたデータをキャッシュとして保存すると共に、関連するデータを事前に転送する。

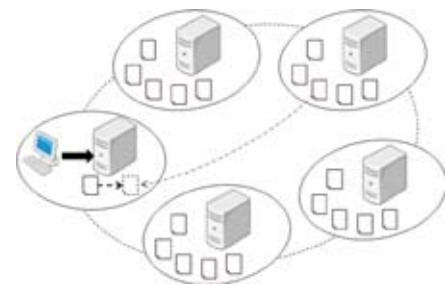


図1 拠点間データ転送管理

データの利用形態は利用者・研究内容・アプリケーションによってさまざまであり、キャッシュアルゴリズムや先読みアルゴリズムを一元的に決定することは不可能である。よって、それぞれのアルゴリズムとして普遍的なものをういようとするアプローチではなく、アクセス傾向とそれに合わせた先読みアルゴリズム・キャッシュアルゴリズムをモジュールとして用意し、それぞれの利用形態に合わせてアルゴリズムを変更するアプローチを採用する。

### 4.2 プロトタイプシステムによる検証実験

4.1節で述べたように、キャッシュアルゴリズム・先読みアルゴリズムは利用形態に合わせたものを用意する必要がある。そこで、提案システムのプロトタイプとして、関連するプロジェクトで行われている研究を対象としたシステムを構築した。使用するアプリケーションは、CTスキャン画像を処理するものでデータ量が多く、ネットワークを介して利用するには長い時間がかかるものである。図2にアプリケーションの画面を示す。

今回のプロトタイプでは、拠点内ストレージにはすべての利用可能なデータの参照先が格納されており、基本的にデータ検索などを行わずにデータが利用できるものとする。データ利用者は拠点内

ストレージ上のデータをアクセスするようにアプリケーションを利用するものとする。これは、データ位置に関する透過性についても考慮するためであるが、用いるアプリケーションにも特別な工夫を必要としないというメリットもある。先読みアルゴリズムとして文字列の類似度を考慮するアルゴリズムを、キャッシュアルゴリズムとして最終アクセス時刻を考慮するものを用い、実際にデータが利用可能となるまでの時間を測定した。

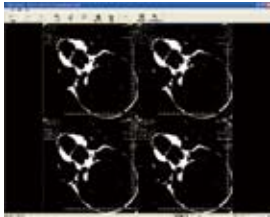


図2 CT スキャン画像

文字列の類似度判断にはレーベンシュタイン距離<sup>2)</sup>を基にする方法を用いた。レーベンシュタイン距離とは、2つの文字列がどの程度異なっているかを示す数値である。具体的には、二つの文字列において、文字の変更、削除、追加の操作を行い同一の文字列にすることができる最小の操作回数である。今回は参照中のディレクトリ名とのレーベンシュタイン距離が一番小さなディレクトリを先読みさせるものとする。これにより、ユーザが参照中のディレクトリの名前を基に先読みデータ(ディレクトリ)を決定し、ディレクトリ単位で先読みするシステムとなる。これは、CT スキャン画像を参照するためのアプリケーションが、ファイル単位でなくディレクトリ単位で読み込むといった特徴があったため、それを踏まえたシステムとするためである。

先読みされた実データがキャッシュに蓄積されていくと、やがてキャッシュ容量が限界になる。新たに先読みされたデータによってキャッシュ容量の限界を超えた場合、アクセスされた時刻がもっとも古いキャッシュデータを削除していき、先読みしたデータを保存する。削除されたキャッシュデータは再び参照先を示すデータに置き換える。今回の実験では、ユーザがアクセスしたディレクトリとその日時をキャッシュリストとして記録しておき、キャッシュの個数が一定数を超えた場合、キャッシュリストの記録を基にアクセスした日時が最も古いキャッシュデータを消去していくこととした。

### 4.3 実験結果

提案システムを用い拠点内ストレージによってデータを利用することができるかどうかの検証実

験を行った。また、今後の参考のため、プロトタイプを用いた場合、通常のファイル共有によって各データにアクセスした場合、データをローカルに配置した場合とでデータが利用可能になるまでの時間を計測した。データセット A(533KB・909 枚)の場合、通常のファイル共有では約2分30秒、プロトタイプシステムでは約49秒、ローカルでは約27秒であった。データセット B(774KB・40 枚)の場合、通常のファイル共有では約8秒、プロトタイプシステムでは約3秒、ローカルでは約2秒であった。データセット C(533KB・381 枚)の場合、通常のファイル共有では約1分16秒、プロトタイプシステムでは約15秒、ローカルでは約13秒であった。

これらより、提案システムのプロトタイプとして想定する機能が動作していることの検証ができた。ただし、拠点内ストレージを介する方法では拠点内の転送速度の影響を受けるため、ローカルに近い利用環境の提供のために利用者のコンピュータへの機能実装や高速なネットワークインフラの整備が必要であることも分かった。

## 5. まとめ

本研究では、データへのアクセス高速化技術として、先読みとキャッシュを組み合わせることで効率的なデータ利用が可能なシステムを提案した。また、提案システムの動作を検証するために仮想的な分散データベースシステムのプロトタイプを構築し検証実験を行った。その結果、提案システムにおいて分散データベースにおける透過的なデータ利用環境を実現可能であることが分かった。

今後の課題として、プロジェクトにおけるデータの利用形態を調査し、それぞれの場合に合わせた先読みアルゴリズム・キャッシュアルゴリズムを設計することやローカルコンピュータへの適用方法の検討などが考えられる。

## 謝辞

この研究は、日本私立学校振興・共済事業団から私立大学等経常費補助金の特別補助によって一部援助を受けた。

## 文献

- 1) 疋田定幸, “図解分散型データベースシステム入門,” 株式会社オーム社, May, 1989.
- 2) V. Levenshtein, "Binary Codes Capable of Correcting Insertions, Deletions and Reversals," *Cybernetics and Control Theory*, pp. 707-710, 1966.

# **Proposal of Network Environment for Distributed Database System**

**Kazunori Ueda \* and Ken-ichi Kamiya**

\* Faculty of Engineering, Kochi University of Technology  
185 Miyanokuchi, Tosayamada, Kami city, Kochi 782-8502 JAPAN

E-mail: \*ueda.kazunori@kochi-tech.ac.jp

**Abstract:** In some occasions, interdisciplinary research project consists of researchers of several institutes in the world. In such case, data used by the researchers are distributed at the institutes. Although computer networking technologies have been progressed in recent years, it takes long time to transmit huge data such as 3D modeling data or high resolution movies. To solve this problem, we propose a new network environment that enables transparency of distributed database systems.