

分散型デザイン画像データベースシステムの検索方法

5Q-4

広瀬直美 勝本道哲 柴田義孝

東洋大学工学部情報工学科

1 はじめに

近年では繊維産業などでは素材や製品を電子化、データベース化しそれらを有効活用しようという動きがある。また繊維産業は地場産業として地域の特性が大きいので、それらの情報は各地域で自律的に管理されたデータベースに格納されている場合が多い。従って、そのように自律分散したマルチメディアデータベースを統合し、透過的にアクセスできるようなシステムの構築が望まれる。そこで本稿では分散型データベースシステムの具体的なプロトタイプとしてデザイン画像データベースシステムを挙げ、本システムにおける効果的なデザイン画像の転送を可能とする2つの検索プロトコルを提案し、その性能を考察する。

2 デザイン画像データベースシステム

図1は本研究で想定しているデザイン画像データベースシステムのプロトタイプ環境である。本システムではクライアント-エージェント-サーバモデルを採用し、本研究室で開発したダイナミックハイパーメディアシステム [1] に基づき、複数のクライアントとマルチメディアデータベース、1つの知識エージェントから構成される。

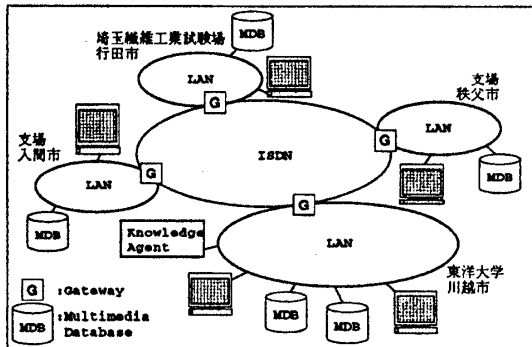


図1: Design Image Database System

本システムで格納されているオブジェクトはテキストと画像から構成され、テキストデータが約100byte、ブラウジング画像データ(100×100[pixel], 256色ピクセルデータ)が10Kbyte、詳細画像データ(400×282[pixel], フルカラーJPEG圧縮)が平均28Kbyteである。

個々のマルチメディアデータベース中では、テキストデータはオブジェクト識別子(object identifier:OID)によって一意に識別、管理され、画像

データもOIDによって一意に決定されるファイル名によって管理されている。個々のマルチメディアデータベース内のオブジェクトの重複はないが、システムの信頼性の向上やオブジェクトの可用性等の理由によりマルチメディアデータベース間では重複して格納されているオブジェクトもある。

3 画像データのバッファリング

本システムではテキストと画像を組み合わせたオブジェクトを想定しており、クライアントにおいては画像データの閲覧(ブラウジング)を行う。その際には全てのオブジェクトをブラウジングするのではなく、数十個のブラウジング画像を含むページを切り替えながらブラウジングする。このため全ての画像をバッファリングする必要があるが、これにはクライアントで行う方法と知識エージェントで行う方法の2つが考えられる。クライアントにおいてバッファリングを行う方法では、ブラウジングページの切り替えの際に画像のネットワーク転送が必要ないので、スムーズなページの切り替えを行うことができる。しかしクライアントにおいては必ずしも最初に全てのブラウジング画像データが必要になるとは限らない。知識エージェントで画像データのバッファリングを行う方法では、検索された結果が膨大な件数であった場合にクライアントのリソースを余分に消費することがないので有効であると考えられる。しかしその場合には、クライアントでブラウジングのページを切り替えた場合にそのページの画像データを知識エージェントに要求し、受け取る、という処理を行わなければならないのでクライアントにおいてバッファリングを行う方法と比べてブラウジングのレスポンスが悪くなると考えられる。

本システムではユーザへのレスポンスを重視し、クライアントでの画像データのバッファリングを採用する。

4 検索プロトコル

上述のシステムにおいて以下の2つの検索プロトコルを提案する。

この2つの検索プロトコルは、クライアントから知識エージェントにUser Queryを発行し、知識エージェントはMulticast RPC[2]を用いてAgent Queryをマルチメディアデータベースに発行するまでは同じ処理を行うが、Agent Queryを受け取った後のマルチメディアデータベースの処理が異なり、それによってマルチメディアデータベースと知識エージェント間で送信されるオブジェクトの画像データ量が異なる。

1). Burst Filtering Protocol

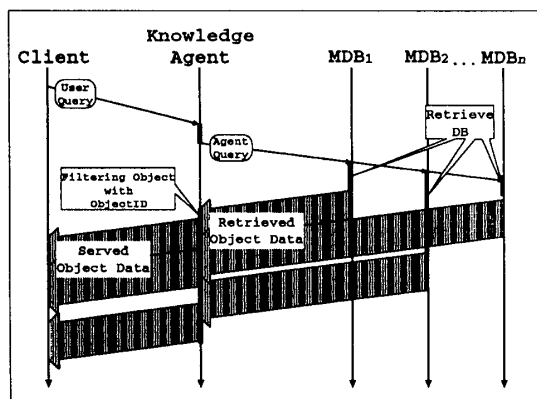


図 2: Burst Filtering Protocol

本プロトコルでは、Agent Query を受け取ったマルチメディアデータベースは、条件に合致するオブジェクトの画像データを含んだ全てのデータを知識エージェントに送信する。オブジェクトデータを受け取った知識エージェントは、各マルチメディアデータベースから受け取ったオブジェクトデータ中の OID を比較することによって重複を除去し、重複のないものをクライアントに送信する(図2)。

このプロトコルは、知識エージェントからマルチメディアデータベースへの要求が1度で済み検索プロトコルとしては単純であるが、マルチメディアデータベース間の重複率が高い場合には冗長なオブジェクトデータの転送によってネットワークトラフィックが増大するためレスポンスが悪くなると考えられる。また検索されたものから順次オブジェクトデータを転送することができるので、画像のブラウジングに関しては良いレスポンスが期待できる。以上のことよりマルチメディアデータベース間の重複率が低い、あるいは重複率がある程度高くても基盤となっているネットワークの帯域が十分であるならば有効であると考えられる。

2). Selective OID Protocol

このプロトコルでは、Agent Query を受け取ったマルチメディアデータベースは、条件に合致するオブジェクトのテキストデータのみを知識エージェントに送信する。テキストデータを受け取った知識エージェントは、各マルチメディアデータベースから受け取ったテキストデータ中の OID を比較することによって重複を除去し、それぞれのマルチメディアデータベースへそれぞれが検索したオブジェクトの画像データを要求する。画像データの要求を受け取ったマルチメディアデータベースは、それらの画像データを知識エージェントに送信し、画像データを受け取った知識エージェントは、それをクライアントに送信する(図3)。

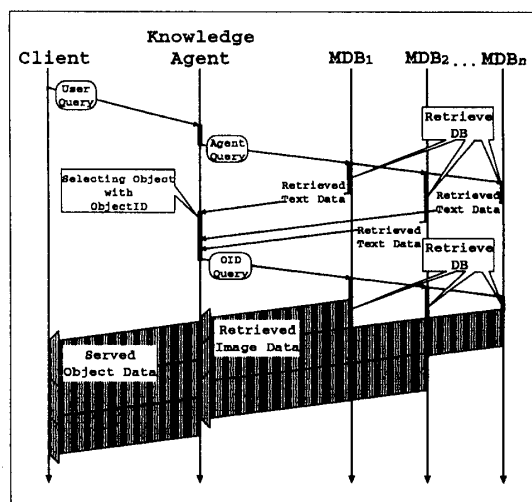


図 3: Selective OID Protocol

このプロトコルは重複したオブジェクトによる冗長な画像データの転送を行う必要がないが、画像データの要求がテキストデータを受け取った後に行われるためクライアントにおけるブラウジングはややレスポンスが悪くなると考えられる。また知識エージェントからマルチメディアデータベースの要求が2度になりプロトコルは複雑になる。以上のことからマルチメディアデータベース間の重複率が高い場合、あるいはシステム全体での検索件数が少ない場合に有効であると考えられる。

5 まとめ

本稿では分散型データベースシステムの具体的なプロトタイプとしてデザイン画像データベースを挙げ、マルチメディアデータベース間でのオブジェクトの重複、及び画像データのバッファリングに関して考慮した Burst Filtering Protocol と Selective OID Protocol の2つの検索プロトコルを提案し、どのような場合に有効であるかを考察した。現在上記のプロトコルを実装し、実際にどのような場合にどちらのプロトコルが有効であるかを評価している。

参考文献

- [1] 勝本; マルチメディア情報ネットワークのためのダイナミックハイパーメディアシステムの研究, 1995年度 東洋大学大学院博士論文
- [2] 鈴木; マルチメディア情報ネットワークのための Multimedia Remote Procedure Call の研究, 1994年度 東洋大学大学院修士論文