

分散型データベースシステムの検索方法及び性能評価

広瀬直美 勝本道哲 柴田義孝

東洋大学工学部情報工学科

E-mail: {naomi,katsu,shibata}@yosemite.sb.cs.toyo.ac.jp

広域分散型マルチメディアデータベースシステムのプロトタイプとして、ISDNにより相互接続されたLAN上にデザイン画像データベースシステムの構築を行った。本システムは、ダイナミックハイパーメディアシステムを基本としており、大規模データベースに対応するために、ユーザとサーバの間にエージェントを配置している。そこで、エージェント-サーバ間の、複数データベースサーバに対応した検索プロトコルとして、Burst Filtering法とSelective OID法の2つを提案し、その評価を行い、比較、検討について報告する。

1 はじめに

近年情報システムの発達により、画像や音声、テキストなどの異なったメディアの情報をワークステーション上で扱うことが可能となり、データベースシステムにおいてもこれまでのようにテキストだけでなく、テキストと画像を意味的に統合し、情報を提供することが可能となった。このような背景のもとに、繊維産業などでは素材や製品を電子化し、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データベースを統括し、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。

そこで著者らは、テキストと画像より構成される、広域分散型マルチメディアデータベースシステムの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の構築を行い、その性能評価を行った[1, 2]。このプロトタイプシステムは実用化を考慮して、Ehternet上だけでなくISDN上にも構築されており、文献[1, 2]ではデータベースサーバが1台の場合の、Ehternet環境とISDN環境における比較評価を行った。その結果、Ehternet環境においても、また転送速度の遅いISDN環境においても十分満足のいくレスポンスタイムが得られ、実用的であることが確認された。そこで大規模実用化を押し進めるために、新たにユーザとサーバの間にエージェントを配置し、エージェント-サーバ間の、複数のデータベースサーバに対応した新たな検索プロトコルを提案し、データベースサーバが複数台の場合の評価を行い、その実用性を確認する。

本稿では、エージェント-サーバ間の検索プロトコルとしてBurst filtering法とSelective OID法の2つを提案し、それらの検索方法の性能評価を行った結果について述べる。

2 デザイン画像データベースシステム

本システムは図1のように、行田市の繊維工業試験場、人間・秩父の各支場及び川越の東洋大学の各LANをISDNで相互接続した広域な環境を想定しており、各LAN上に分散格納されているテキスト、及び画像データを、どのワークステーションからも透過的に検索できるようなシステムである。本システム

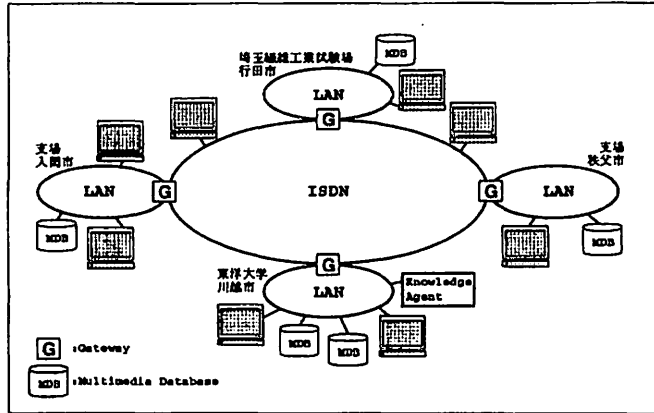


図 1: デザイン画像データベースシステム

は、これまで本研究室で開発されてきたダイナミックハイパーメディアシステム [1] を用いて構築されている。

デザイン画像データベースシステムは埼玉県の地場産業である繊維工業の画像データベースであり、デザイナーのためのデザイン支援システムである。これまで各地域で構築、管理されていたデータベースを統合し、分散型データベースとして管理する。

本システムは、埼玉県内の各地で実働している既存のデータベースを統合し、透過的なアクセスが可能なことを目指しているため、通常なら Bottom-up の構築アプローチをとるところだが、これらのデータベースは統合されることを考慮し構築されていたので、Top-down 的なアプローチが可能であった。そのため、既存の全てのデータベースが、データモデルは関係データベースモデルを基本とし、データ言語は SQL 言語を用いており、統合後のデータベースの形態は Homogeneous なデータベースである。さらに、ローカルスキーマは全てのサイトで同じものを使用しているため、格納されているデータは水平分散をしていると考えられ、そのため、統合スキーマの生成が必要なく、構築は比較的容易である。画像データは、そのファイル名に相当するオブジェクト識別子 (Object identifier:OID) によって管理されている。また、格納されているデータの OID は重複がなく、分散しているデータベース空間においてユニークであるが、信頼性の向上や地域的特色性、あるいは過去の検索の履歴により、重複して格納されているデータもある。

2.1 感性検索法

これまで行われてきた一般的な検索は、素材や柄、パターン、系統色等オブジェクトに既に割り当てられているキーワードにより検索していたが、本システムはデザイナーの支援を目的に構築されたもので、デザイナーは、「エキゾチックな」や「ダンディな」といった、感性に訴えるようなキーワードで検索できることを望んでいる。しかしこのような感性語とデザイン画像の関係づけは、データベース構築者の主観によってインデックスづけが行われ、検索を行う個人の主観が反映されず、ユーザの期待したような画像が検索されないことがある。そこで本システムでは、キーワードによる検索のみではなく、曖昧な表現や、感性語等による検索を行う感性検索を可能にしている。これはユーザ個人の感性を、ユーザモデルを用いて平均的なユーザからの「ずれ」で表現することにより、個々のユーザの主観を反映した検索を行うことが可能になる [3]。

本システムでは、感性語は大きく 15 の形容詞に分割され、180 ある感性語はそれらを含む形容詞にクラスタリングされている。さらにその感性語には、それぞれその感性語に相当する 3 色配列が割り当てられている。画像の代表色と 3 色配列の色を比較することにより、各画像と感性語を対応させることが出来る。

実際のデータベースの検索ではその配列中の色による Query を発行する。すなわち、ユーザからの検索要求 (User Query) は感性語によるもので、ユーザモデルを用いて変換したあとのデータベースへの検索要求 (Agent Query) は、その感性語に割り当てられている三色配列の色番号によるものである。表 1 に感性語と 3 色配列の一例を示す。

表 1: 感性語と色配列の組合せの例

形容詞	感性語	3 色配列		
プリティ	甘い	R_P	Y_Vp	YR_P
		R_P	YR_Vp	RP_P
カジュアル	かわいい	Y_P	Y_Vp	RP_B
	あざやかな	TP_V	G_V	Y_V
	カジュアルな	R_Dp	N_9	B_Dp

2.2 ダイナミックハイパーメディアシステム

ダイナミックハイパーメディアシステム (Dynamic Hypermedia System:DHS) とは、マルチメディア情報ネットワークシステムを実現するための基本概念であり、クライアント-エージェント-サーバモデルを導入しており、クライアントエージェント (Client Agent)、知識エージェント (Knowledge Agent)、マルチメディアデータベース (Multimedia Database:MDB) から構成される (図 2)。

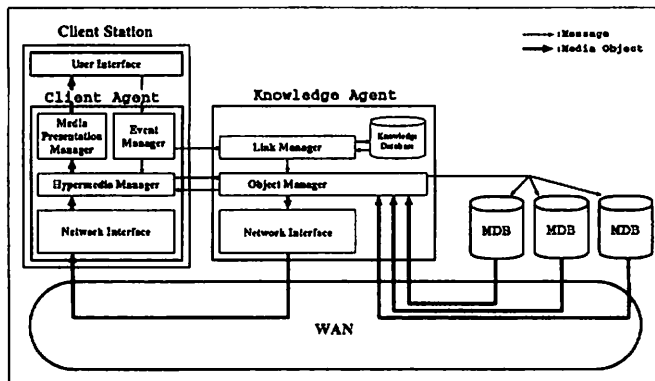


図 2: ダイナミックハイパーメディアシステム

これまでのクライアント-クライアント-サーバモデルでは、データベースが分散している場合にはユーザがそれらデータベースの位置を認識し、個別にアクセスしなければならなかった。各データベースに格納されているデータの格納方法や属性を知らなければ効率的な検索は行えなかった。また、データベースの追加、除去といった場合には、それらの情報の更新を行わなければならない。また、同時実行制御を行わなければならないので、システムが複雑になる。

そこでクライアントとサーバの間にユーザに代わって仕事を支援するエージェントを存在させ、データベースの位置情報、データの格納法や属性の保持、及び管理を実行させることによって、分散しているデータベースへのアクセス等を代行し、ユーザの負担を軽減することが出来る。すなわち、ユーザはエージェントさえ知っていれば個々のデータベースに関する知識を持つ必要がなく、エージェントにアクセスすることによって複数のデータベースから情報を引き出すことが出来る [4]。

2.3 クライアントステーション

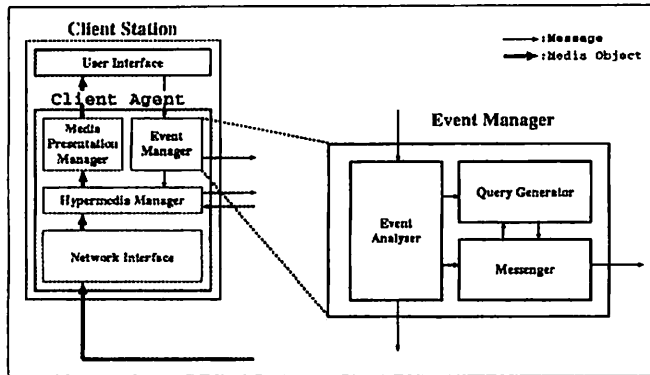


図 3: クライアントステーション

クライアントステーションでは、ユーザからの要求の処理等ユーザとのインタラクションを管理し、検索要求の発行、知識エージェントとの通信を行う。またオブジェクトを表現するためのインタフェースを提供し、知識エージェントから受けとったメディアデータの表示等を行う。

ユーザの要求は、提示された感性語から自分の欲する画像の条件を複数指定する。例えば、ユーザが感性語「かわいい」を選択し、検索要求を行ったとすると、User Query としては、感性語「かわいい」が知識エージェントへと伝えられる。この処理は、クライアントエージェント内のイベントマネージャ(Event Manager)によって行われる。

2.4 知識エージェント

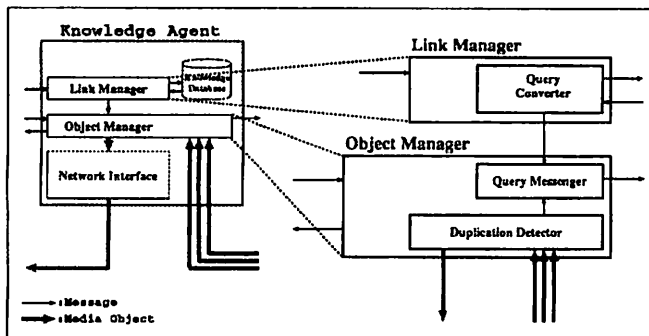


図 4: 知識エージェント

知識エージェントでは、デザイン画像データベースにおける感性検索のためのユーザモデルの管理、及び更新を、リンクマネージャ(Link Manager)、及び知識ベース(Knowledge base)で行い、それらを用いて User Query から Agent Query への変換を行う。また、Query の発行、及び重複したデータの除去等をオブジェクトマネージャ(Object Manager)で行う。さらに、複数の知識エージェントを存在させ、その知識エージェント間で協調することによって知識の拡張を行い、大規模データベース空間を実現している。

User Query を受けとったリンクマネージャでは、それが感性検索による検索要求だと判断した場合にはクエリコンバータ(Query Converter)を介して知識ベースへのアクセスを行い、その検索要求を発行した

ユーザのユーザモデルを用いて感性のずれを補正する。その後、感性語と色に関する知識ベースを用いて、補正されたユーザの感性を、色とその割合に変換する。すなわち、User Query として渡された「感性語」を、ユーザモデルを用いて補正し、知識ベースを用いてそのユーザの感性に相当した色とその割合に変換する。その「色とその割合」が Agent Query となりオブジェクトマネージャに渡される。

リンクマネージャから Agent Query を受けとったオブジェクトマネージャでは、クエリメッセンジャ (Query Messenger) によって、複数の MDB へマルチキャスト対応のリモートプロシージャコール (Multicast Remote Procedure Call[4]) を用いて Agent Query を発行している。

また、各 MDB に格納されているオブジェクトは重複している場合もあるので、検索結果をそのままユーザに提供したのでは、検索システムとしては冗長的であるので、この重複を取り除いてからユーザに提供する必要がある。そこで、オブジェクトマネージャ中のデュプリケーションディテクター (Duplication Detector) で、その重複を除去する。ここでは除去する方法として、検索してから重複を除去する方法 (Burst Filtering 法) と、重複しないように Agent Query を発行する方法 (Selective OID 法) を提案している。

オブジェクトマネージャは、重複のないデータをネットワークインタフェース (Network Interface) に渡し、クライアントエージェントに転送する。

2.5 マルチメディアデータベース

マルチメディアデータベース (Multimedia Database:MDB) では、知識エージェントからの検索要求に合致したデータの検索、提供を行う。

格納されている各オブジェクトは OID を持っており、システムはこの OID によりオブジェクトを一意に決定することが出来る。また、各画像がどの色をどれくらい含んでいるかは、あらかじめ簡単に調べることが出来るので、その画像の面積比上位 5 色とその面積比率がデータベースに登録されている。さらに MDB では、リレーショナルデータベースによって各データを管理しており、画像等のバイナリデータはデータベースのエントリとして画像のファイル名を用いている。

本システムで用いる画像は、画像の一覧に用いるブラウジング用の画像と、その詳細を示す詳細表示用の画像の 2 種類がある。ブラウジング用の画像は、100 × 71(pixel) の大きさで 256 色のピクセルデータである。また詳細表示用の画像は、400 × 282(pixel) の大きさでフルカラー画像を JPEG によって圧縮したものが格納されている。これは、ISDN を介したサービスを行うので、詳細用の大きな画像データを転送するには ISDN の遅い転送速度がネックになると考えられるので、相対的なデータ転送量を減らし、ネットワークの負荷を軽減させるものである。従って、クライアントステーションにおいて伸長が、さらにクライアントステーションの属性によってはディザによる色変換が必要となる。

3 検索方法

前節のようなデザイン画像データベースにおけるエージェント-サーバ間の検索プロトコルとして、以下の 2 つの方法を提案する。

3.1 Burst Filtering 法

この方法は、クライアントから発行された User Query を、知識エージェントにおいて Agent Query に変換した後、各 MDB にその Agent Query を発行する。Agent Query を受けとった MDB は、その Query に合致するオブジェクトのデータ全てを知識エージェントに転送する。MDB からのデータを受けとった知識エージェントでは、各データ中の OID をチェックし、まだ検索されていないデータであったらクライアントへ転送し、既に検索されていたデータであったら知識エージェントで破棄し、クライアントへは転送しないことにより、検索されたデータの重複を除去する (図 5)。

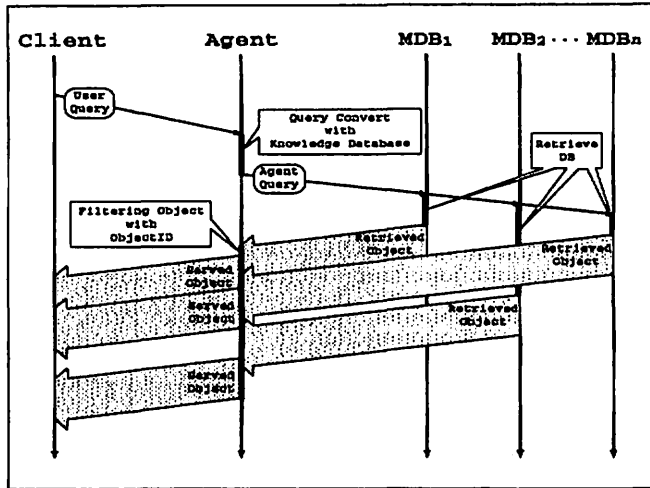


図 5: BurstFiltering 法

この方法は、検索時における知識エージェント-MDB 間の通信が一度で済み、検索プロトコルは単純である。しかし、検索されたデータにおける重複率が高い場合には、ネットワークトラフィックが増大し、さらに知識エージェントにおける重複を除去するための処理 (Filtering) の時間が大きくなると考えられる。

3.2 Selective OID 法

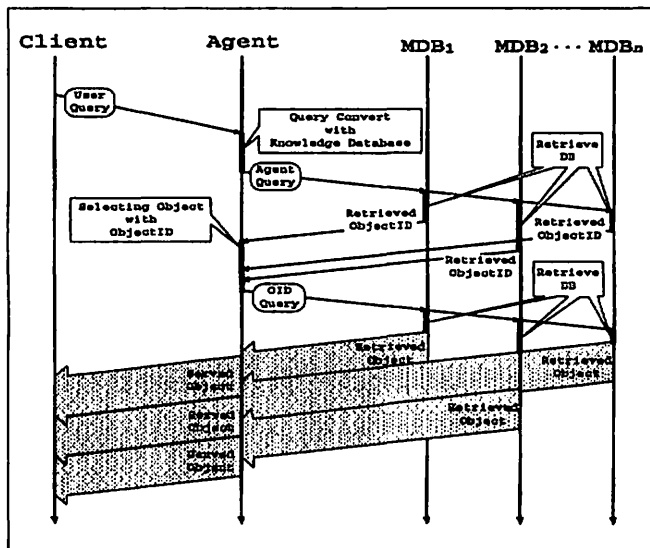


図 6: SelectiveOID 法

この方法は、クライアントから発行された User Query を知識エージェントにおいて Agent Query に変換し、各 MDB に Agent Query を発行するまでは上記の Burst Filtering 法と同じである。知識エージェントからの Agent Query を受けとった MDB は、その Query に合致するオブジェクトのデータ全てを知識エージェントに転送するのではなく、データ中の OID のみを Agent Query の検索結果として転送する。

Agent Query の検索結果を受けとった知識エージェントでは、その OID をチェックし、重複がないかを調べる。重複が除去できたら、各 OID を検索したメンバDB に、選択された OID による Query(OID Query) を発行する (図 6)。

この方法は、オブジェクトデータを転送する前に重複が調べられるので重複したデータの転送がなく、余計なトラフィックを押えることが出来る。しかし、データを転送するまでの検索要求が 2 度になり、また知識エージェントで OID を選択し、さらにその OID を検索した MDB に OID による検索要求を発行しなければならないので検索プロトコルが複雑である。

表 2 に両プロトコルの比較を示す。

表 2: 両プロトコルの比較

	Burst Filtering	Selective OID
重複の除去の仕方	検索してから除去する	重複しないように Query を発行する
Query の発行	1 度 (Agent Query)	2 度 (Agent Query と OID Query)
Agent Query による MDB の動作	検索後データを転送する	OID のみを転送する
データを受けとった Agent の動作	データ中の OID をチェックし 重複を除去してから転送する	そのまま転送する
重複率が高い場合	余計なデータ転送が増大する	冗長なデータ転送がない
重複率が低い場合	Filtering の処理が増える	

4 性能評価

本稿では、Burst Filtering 法の実用性を研究するために、以下のような環境で性能評価を行った (図 7)。

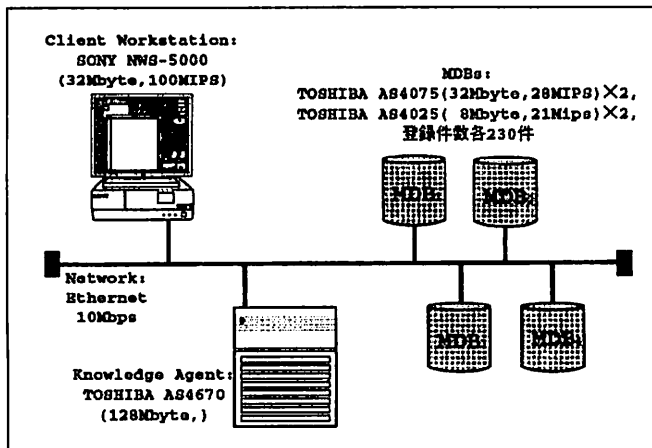


図 7: 性能評価プロトタイプ

クライアントステーション、知識エージェント各 1 台、MDB は 4 台でリレーショナルモデルを採用し、SQL 言語によって記述されており、同じスキーマを持ったデータベースが格納されている。登録件数は各データベース 230 件である。ネットワークは Ethernet を用いた。データベースには各画像の OID、ブラウジング用と詳細用の画像ファイル名、面積比上位 5 色の色番号とその面積比率が含まれている。

評価方法は、MDB が 1 台の場合と複数台の場合で比較する。ユーザからの要求件数が同じである時、MDB が複数台の場合に、検索されるオブジェクトの重複率を変化させたらどのような傾向になるかを評価

したものである。画像データは、データベースに格納されているファイル名のバイナリデータを転送している。今回用いた画像は、400 × 282(pixel) の 256 色の詳細表示用のピクセルデータで、性能評価のため固定サイズの無圧縮の画像を使用している。画像とテキストを含めた 1 オブジェクトのデータ量は約 100Kbyte である。性能評価としては、ユーザが検索要求を発行してから、全ての画像データの受信を完了するまでであり、オブジェクトデータの転送、知識エージェントでの重複したデータの除去等を含んだ End-to-End のレスポンスタイムである。

図 8 は、MDB が 3 台の時の End-to-End のレスポンスタイムであり、x 軸は検索されたオブジェクト数、y 軸はレスポンスタイム、グラフ中の 0~100% は検索されたオブジェクトの知識エージェントでの重複率を示している。

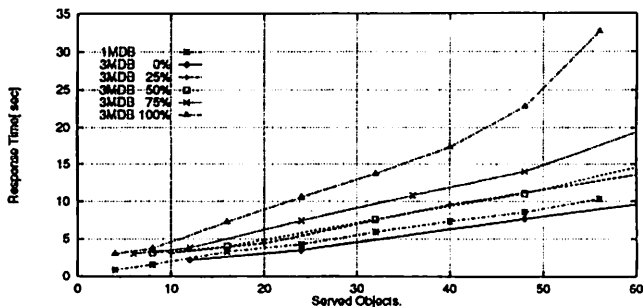


図 8: End-to-End レスポンスタイム (MDB 3 台)

これを見ると、データの重複率が 0%、すなわちデータの重複がない場合には、MDB が 1 台の時とはほぼ同程度のレスポンスを示している。これは、MDB での検索処理やオブジェクトデータの転送が、複数の MDB で並列に行われ、その分 MDB が 1 台の場合よりもレスポンスタイムが多少向上したと考えられる。しかし、データに重複があると、1 台の時よりもレスポンスタイムは増加していく。これは、データに重複があると、ネットワークへの負荷が増大し、知識エージェントにおける冗長なデータのフィルタリング処理を必要とするので、それが影響を及ぼしていると考えられる。

次に示すグラフはそれぞれ、サーバの数を 1~4 台と増加させた時の End-to-End のレスポンスタイムを示している。図 9 は、検索されたオブジェクト数が同じである時、重複率が変化したらどのようなようになるかを、図 10 は、知識エージェントでのオブジェクトの重複率が同じである時、検索されるオブジェクトの数が変化したらどのようなようになるかを示したグラフである。x 軸は MDB の数を、y 軸はレスポンスタイムを示している。

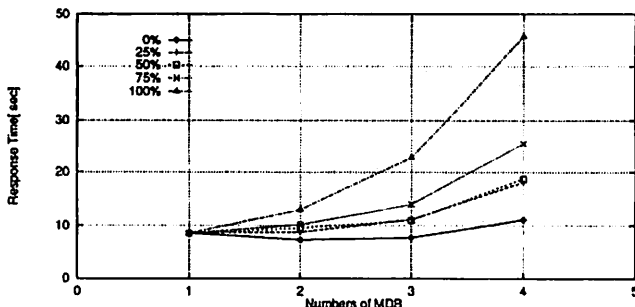


図 9: End-to-End レスポンスタイム (オブジェクト数一定:48 個)

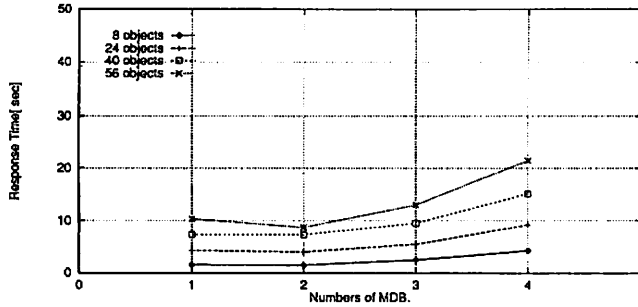


図 10: End-to-End レスポンスタイム (重複率一定:25%)

上の2枚のグラフを見ると、どちらの場合もMDBの数が増えるに従いレスポンスタイムが悪くなっていくが、図9では重複率が低いほど、図10では検索されたオブジェクト数が少ないほど、MDBの数による影響を受けていない。これは、重複率が低い場合には、各MDBで検索処理やデータの転送処理が並列に行われるため、その分レスポンスが早くなり、重複率が高くなるにつれMDBにおける冗長な処理が多くなり、さらにネットワークへの負荷も増大し、知識エージェントでのフィルタリング処理時間も大きくなるためであると考えられる。また、図10のように、知識エージェントにおける重複率が25%の場合でも、検索されたオブジェクト数が少ない場合にはネットワークへの影響が少ないためMDBの数による影響を受けにくいと考えられる。

5 まとめ

本稿では、エージェント-サーバ間の検索プロトコルとして Burst filtering 法と Selective OID 法の2つを提案し、MDBが1~4台の場合における Burst Filtering 法の性能評価を行った。その結果、MDBが3台の場合は重複したデータがない時に、MDBが1台の場合と同程度の性能を得ることが出来た。しかし、オブジェクトに重複があるとその分余計な負荷をネットワークにかけることになり、さらに、知識エージェントにおける重複したデータのフィルタリング処理が増加する。従ってこのような分散環境においては、データベース空間全体でデータの重複率をある一定レベルにする分散データの管理法が必要とされる。また、検索されるオブジェクトのしほり込みを行うことにより、実際に転送されるオブジェクトの数を抑えることも必要である。

今回はデータベースサーバを複数台にし Burst Filtering 法の性能評価を行ったので、今後の課題としては、さらに考察を行い、データベースの分割の割合、オブジェクトの重複の割合等をどれくらいにしたら、効率的な検索が行えるかを調べる必要がある。また、実用化をふまえた ISDN 環境における性能評価、細かいコンポーネント毎の測定、それによるボトルネックの分析、解明等を行う予定である。また、Selective OID 法を実装しその性能評価を行い、Burst Filtering 法との比較評価を行う。それにより、どちらがどのような場合に効率的かを考察する予定である。

参考文献

- [1] M.Katsumoto and Y.Shibata, "Dynamic Hypermedia System Using Knowledge Agent for Multimedia Information Network", JWCC-8,C2-2-1~C2-2-8,Dec.1993.
- [2] 勝本, 入江, 広瀬, 瀬田, 柴田: ISDN による広域デザイン画像データベースの構築と性能評価, 情報研報 94-DPS-64, Vol.94, No.19, Mar.1994.
- [3] 勝本, 入江, 柴田: デザイン画像データベースシステムにおけるヒューマンインタフェースの研究, 情報処理学会第48回全国大会, 2E-6, Mar.1994.
- [4] 鈴木 隆之: 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