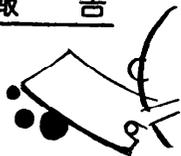


報 告

パネル討論会

マルチメディアデータベースシステム



パネリスト

川越 恭二¹⁾, 大町 一彦²⁾, 牧之内顕文³⁾
 増永 良文⁴⁾, 小島 功⁵⁾
 司会 穂鷹 良介⁶⁾

穂鷹(司会) それでは、「マルチメディアデータベースシステム」という題名のパネル討論を始めたいと思います。

スピーカの方には最初10分の持ち時間でワラウンド話をしていただきます。5人の方がお話になりますと、だいたい3時40分ごろに終わる予定と思いますが、そこでコーヒープレイクに入りたいと思います。

そのあと5分以内で、もうワラウンド各パネリストの方に意見の追加、あるいは他のスピーカの意見との比較などを行っていただきたいと思います。

実は先日筑波で開かれた11月のデータベース研究会で「マルチメディアデータベース」というパネルを本日とはほぼ同じメンバーで行いました。そこでは各スピーカはだいたいどのぐらいの年月でマルチメディアデータベースがものになるか、というつもりでお話になるかということをお聞きしたわけです。それだけをきょうの最初のインプットとして申し上げます。

川越さんは、きょうは態度が変わるかもしれませんが、だいたい7年から10年ぐらいあとにマルチメディアデータベースはものになるという予想でございます。大町さんは、当日は木村さんがご出席になっておりましたが、5年後という感じで捉えている。牧之内さんは、3年ぐらいのところプロトタイプが終わり、商品化が5年後ぐらいであろうと。増永先生もやはり5年ぐらいであろうと。小島さんも、3年から5年後ぐらいにマルチメディアデータベースというのは出現するのではないかという感じで考えている。各スピーカはこういうような実現の具合ということで、考えておられるようでございます。



それでは、川越さん、大町さん、牧之内さん、増永さん、小島さんの順番でご発表をお願いします。ではよろしく願います。

川越 日本電気の川越です。

先ほど7年から10年というお話がありましたけれども、それはマルチメディアデータベースをどう定義するかということに依存する話だと思います。マルチメディアデータベースシステムには、かなりのチャレンジをする必要があるだろうという認識をもっています。いまの意味のデータベースシステムであれば3年ぐらいでできるかと思いますが、完全なものならかなりのギャップが現在あるという気がしております。

きょうは、「マルチメディアデータベースとは」ということと、「従来のDBMSとどう違うのか」という話をしたいと思います。

マルチメディアデータベースの定義の前に、データベース自体の定義をみますと、データベースとは一つ以上の実体の集まりで、内容を高度に構造化することで検索、更新の効率化を図ったものとなっています。

この定義を利用させていただいてマルチメディアデータベースを定義しますと、実体のところがメディア情報になり、「内容を高度に構造化する」というところが「暗黙に表現された意味を利用して、必要に応じてその構造化、関連づけを行う」ということに変えます。

結局その違いは、実体をメディア情報にしたことと、構造化という点で、その可能性があるかとか、あるいはそのレベル、それをどういうタイミングでするのか、ということが違ってきます。

そういう違いを表で説明したのがこの表(表-1)です。要点だけいいますと、構造化を行うタイミングが利用時であるということ、データ量が少し膨大に



↑日時 昭和60年12月11日

場所 機械振興会館大ホール(地下2階)

1) 日電, 2) 日立, 3) 富士通研, 4) 図書館情報大, 5) 電総研,

6) 筑波大

表-1 従来のデータベースとのちがい

	従来のDB	マルチメディアDB	課題
データ	抽象度の高いデータ	抽象度の低いデータ	意味抽出
データの意味	データ構造=意味	人間のみ意味理解可能	意味表現モデル
データ構造	構造変更なし	利用により構造化	動的構造化
データ量	1件1-1,000バイト	1件1-100Mバイト	データ圧縮
ビュー	閉じた世界	開いた世界	メディア変換
効率	用途によりバッチ処理可	主に対話処理	効率的DB処理と構造化
インタフェース	DML/EML	EML 中心	ユーザ適応インタフェース
DB構築	設計後生成	DB生成後構造化	動的構造化

なるから、その対策が必要であるということ、ユーザビューというものがかなり開いた世界で柔軟に対応できる機能をもつ必要がある、ということです。

それでは、従来のデータベース管理システムで実現できないかという問題を考えます。答えは実現できると思います。しかし問題はいくつか多分出てくるでしょう、その一つは、パフォーマンス。ほかに情報の質、それから更新時のインテグリティ、あるいはある利用者には有効であるけれどもそれ以外の利用者に関しては全く使えないようなデータベースになる可能性があるという意味での、共有問題が発生するでしょう。

そうすると、この問題を解決するにはどうすればいいかといいますと、モデルの問題を解決すること、データベースの構成を解決すること、格納構造の問題、インタフェースの問題、それから、メディア変換の問題です。

これに対応させて少し私の意見を述べますと、モデルに関しては、要するにメディア間の関係をとるということと、メディアの情報の意味をモデル化する必要があるということ、データベース構成についてはメディア独立性を新たに考える必要があるということ、格納構造に関しては、よりデータの特性というものを考えて、それに応じた配置を考えていく必要があるでしょう。

インタフェースについては、EUL と DML が本質的にマルチメディアの場合は違うんだという認識をもつ必要があります。

変換に関しては、構造化でカバーできるだろうと感じています。

一つ例を使って説明させていただきますと、あるマークがあります。普通の人にはそれをどういうマークかわからないけれど、私は理解できるものであります。これは、カリフォルニア大学にあるローレンスバークレー研究所のマークです。このようにマークを知ってる人にとってみたら、それを見ただけですぐ

判断がつかますが、全くこれに関係ない人にとっては、なんのこともかさっぱりわからない。つまりこれはイメージ情報なんですけれども、それに意味情報というものが内在していると考えられます。この情報を構造化ということで抜き出してやって、ローレンスバークレーと書いてあるとか、ユニバーシティ・オブ・カリフォルニアと書いてあるとか。このことからいろんな情報が抜き出せる。たとえば、カリフォルニアというのはユニバーシティ・カリフォルニアがある場所を示しているし、バークレーとその間の関係も当然設定できるでしょう。これを利用時、利用する直前に、あるいはデータを登録するときに利用者が入れないと使えないだろうということを考えているわけです。

すなわち、この構造化というものがマルチメディアDBでは非常に本質的な意味合いをもつと思います。

それから、入力の場合の動的な構造化と同様、アウトプットの場合でも複数のメディア情報を融合化させるという問題がマルチメディアデータベースには必要なだろうということです。

また、構造化されたものをどう保存するかということに関して、私はセマンティックネット的なもので十分対応できると思います。最近オブジェクトということが話題になっていますが、セマンティックネットの各ノードをオブジェクトで対応づければ、うまくカバーできるだろうと思っています。

最後に私の主張をまとめてみます。マルチメディアシステムというのは、結局マルチメディアデータベース管理システムをもたないといけない。このマルチメディアデータベース管理システムでは、基本的には複数の応用を想定する必要があるでしょう。その基本的なコンポーネントとして、先ほどいいました動的構造化、それからセマンティックネット的なものを使って関連あるいは意味情報をもたせるということ、動的構造化が非常にユーザの本質に関するものですので、エンドユーザ・インタフェースというもの、これらを

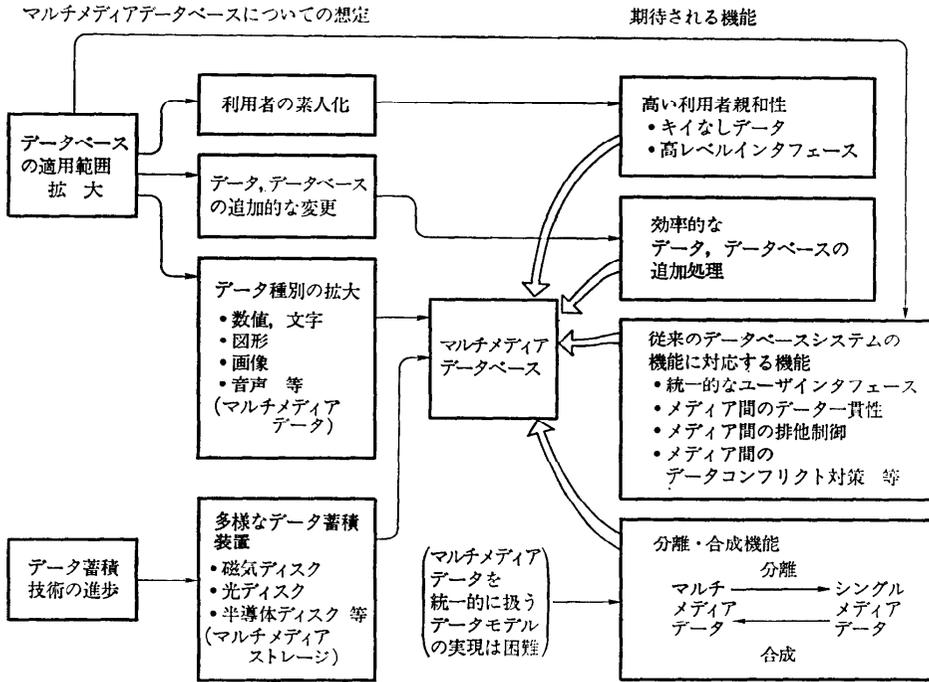


図-1 マルチメディアデータベースシステムへの期待

DBMS 自体が管理する必要があります。

それから、メディア独立性というものをまた別の次元で考える必要があって3層スキーマでの物理、論理概念というレベルと、また別の違った軸でメディア独立を考えていく必要があるんじゃないか、というのが私の意見です。以上です。

穂鷹 次に、大町さんお願いいたします。

大町 マルチメディアデータベースというのは、私自身よくわからないものですから、マルチメディアデータベースということに関して、こういうものがほしい想定されているのではないかとということをもとにして、多分こういう機能が必要なのではないかとというようなお話をしたいと思います (図-1)。



データベースの、適用範囲が非常に拡大してきたということで、利用者が素人になってくるだとか、データベースの種類が増えることほかに、従来のデータと違うと思うんです。従来のデータは、あと何回も読まれるのであろうからしまわれるんだということで、リード/ライト比が、たとえばリードが5に対してライトが1だとか、30対1だとか、そういったこと

でリードが中心だったと思います。マルチメディアになってきますと、たとえばデータを、われわれ一般に扱っている書類だとか、契約書だとか、そういうものについては改ざんをしてはいけない、アップデートをしてはいけないという世界があると思うんです。

そういう世界に対してそういうものを扱おうということになりますと、データのライトというのはインサートが中心になってくる。インサートなんですが、先々使うかどうかかわからないような情報、とにかくデータがたくさんあることが大切であって、使うか使わないかわからないような種類の情報、たとえば、国会図書館の情報量は非常に多いわけですが、あれを隅から隅まで利用する人はほとんどいなくて、あそこになくさん情報があるということ自体に意味があるということで、リード/ライト比がむしろライトのほうが多いという状況が一つあるのではないかと感じる私はもっています。

適用範囲は、従来は先ほどいいましたリード系が中心だったんですが、範囲が増えるにつれて、インサートが多い世界、このへんもちゃんとサポートしてやらなければいけないんじゃないかという状況が出てくる気がします。これがデータの追加的な変更という内容

です。

それからもう一つ、いろいろな DBMS の上にいろいろな種類のデータベースシステムが構築されておりまして、朝の植村先生のお話ではないんですが、それがお互いに結ばれて有機的な情報処理を行うことになりまして、違った世界でつくられたデータベースが、ネットワークに追加される。データベース自身が増えてくるようなことがあるのかなと。別の世界で構築されたデータベースが、ネットワークに組み入れられて、そういうものを扱わなければいけなくなるのかなというのが、ここに書いたデータベース自身の追加、それからデータの追加というようなことです。

それからもう一つ、傾向としては、扱うデータだけでなく、どうも蓄積のデバイスのほうもいろいろ出てきて、こちらも含めて世の中でマルチメディアというふうにいわれているのかなと。もちろん通信系はありますけれども、データベースということで、データ蓄積系をいま着目しているわけです。

従来の数値、文字、図形、画像、音声といったものを十把ひとからげにマルチメディアデータといま呼びまして、それから蓄積系として磁気ディスクだとか、半導体メモリだとか、光ディスク、これを仮にマルチメディアストレージと呼ぶとしますと、このマルチメディアストレージ上にこのマルチメディアデータをデータベース化した——このデータベース化したというのが非常にいいかげんな表現だと思うんですが、そういうものがマルチメディアデータベースではなかるうかと考えているわけです。

もう一つの想定として、このマルチメディアデータを統一的に扱うデータモデルの実現は、かなりむずかしいのではないかと考えておるわけです。

先ほどのご説明でも、メタレベルでの総合はマルチメディアと直接関係ないのではないかとというご意見と両方あると思うんですが、マルチメディアデータを統合的に扱うデータモデルは、なかなかむずかしいという想定をしますと、それから期待される機能として、こちら側にあげた機能群があるのではないかと。

利用者が素人になりますと、当然高いユーザフレンドリーネスというのが必要になって、それはよくいわれる高いレベルのユーザ言語だとか、キーなしデータ、これはわれわれが日ごろつくるメモを適当にファイルするときに、そのファイルにいちいちキーをつけないということがありまして、素人が情報をたくさん持つ場合に、いちいちインデックスをつくれるか、つく

れないケースもなんとか扱わないといけないんじゃないか、これは内容検索にかかわる問題だと思えます。

それから、先ほどの追加に関しては、アップデートじゃなくてインサートというものに着目すると、何かうまく処理が出てくるのではないかと。そのところが効率的になるんじゃないか。

従来のデータベースシステムも持っている機能で、対象をマルチメディアデータにしたときに要求されるであろう、つまり、従来のデータベースでいろんな機能も持っているわけですが、対象がマルチメディアになっても、やはり同じような機能が当然要求されるのではないかと。

ここで、ユーザインタフェースだとか、データ一貫性だとか排他制御もそうなんですが、このへんは非常にポピュラなので説明は省きます。このデータコンフリクトについて説明しますと、先ほどのネットワークに別の世界でつくられたデータベースがどんどん追加されることになると、意味上のぶつかり合い、データベースとデータベースとの間で意味上のぶつかり合いが出てくるのではないかと。ウィリアム・ケントの「データアンドリアティ」じゃないんですが、本という意味でも、たとえば、「夏目漱石全集」が全3巻からなっていたときに、それを3と数えるのか、1と数えるのかとか、本の中に雑誌を入れるのか、入れないのかとか、いろんな解釈の仕方があってデータベースとして別の世界でつくられていると、こちらのデータベースでいう本と、別のデータベースでいう本が意味的にぶつかってしまう。そういうものを扱えないといけない。

これはマルチメディアにかかわらず、異種のデータベースシステムを結合するときの問題なわけですが、マルチデータベースになると、そういう異種のデータベースがぶつかり合う危険がより増えるのではないかと考えるわけです。

もう一つ、マルチメディアデータを統一的に扱うのは非常にむずかしいんじゃないか、かなり消極的なんですが、実際的にはこれをシングルメディアの集合として扱わざるをえないのではないかと思うわけです。そうしますと必要機能としては、マルチメディアをなんらかの意味で最終的な要素に分離する、あるいは先ほどのご発表にあったように、抽象化だとか、汎化だとかいった意味で統合する機能、こういうものではないかと思えます。ここから先、じゃ具体的になんなのかということがまだよくわかりませんので、このぐらいでやめさせていただきます。

穂鷹 どうもご苦労さまでした。それでは、牧之内さん。

牧之内 富士通研究所の牧之内です。マルチメディアデータベースについて、私がどう考えているかについて簡単にお話ししたいと思います。



マルチメディアデータベースとは何かは、いま大町さんがおっしゃられたようにまだ確立しておりませんので、いまなら勝手なことがいえるわけです。それで勝手なことをいいたいと思うんですが私の考え方はきのうの三菱電機の和田さん、きょうの IBM の洪さん、日立の島田さんのご発表を聞いた後では方向としては間違っていないと確信しました。

- 数値、文字列
- テキスト
- グラフ
- イメージ
- 音声

に加えて

- 手続的「知識」

図-2 マルチメディアデータ

スクリーンに写っているように(図-2)数値、文字列、テキスト、グラフ、イメージ、音声等々、いわゆる計算機で扱いたいデータが種々あるわけです。私の主張は何かといいますと、マルチメディアデータといったとき、それらデータに手続的知識を加えてマルチメディアデータと呼ぶべきだということです。

もう少し詳しくいいますと、マルチメディアデータの特徴は、各種メディアデータがそれぞれ特有の構造と演算操作を持っていることです。たとえば、いままでの数値・文字ですと、その演算は非常に限られたものでした。しかしながら、たとえば整数と浮動小数点では演算の意味が違っていただけです。ところが、それ以外に図形であるとか、画像が入ってきますと、そういう演算というものがそれぞれより分化・複雑化し、またアプリケーションにより依存したものになるであろう。

2番目に、非常に複雑な構造を持つメディアがある。けれどもそれは、いわゆるプログラム言語というデータ構造と特に異なるわけではない。データ構造はいくらだって複雑にできるからです。

3番目に、しかしながら、基本的概念をもちこんで「何でもできます。さあどうぞ!」といったところでユーザは困るわけです。というのも、利用者はエンドユーザでありまして、自分の業務については興味があ

るけれども、計算機については全然知らない。プログラムなんて、なおさら書く気がしないわけです。したがって、そういうデータがあったときに、こういう使い方をすればいいよ、こういう業務をするには、こういうものを使ってこうやればいいよ、ということをお教えしてやらなければいけない。すなわち、メディア利用に関する情報も、またメディアとして入れておけばよろしい。

これを従来のデータベースと比べてみますと、いままではデータがプログラムと独立にありました。それが売り物だったわけですが、しかしユーザにはどうもそれだけではたりないんで、いわゆる DD/D を導入しなければならない状況が出てきます。その DD/D によってデータとアプリケーションプログラムの両方ともを管理しなければならないという考え方があるわけです。これはデータとプログラムをいわゆる知識ベースという形で統合化する考え方につながります。マルチメディアデータベースというのは、いわゆる知識ベースです。知識ベースの一つの分野であるというのが、私の主張です。

それじゃ、それをどうやって実現するかということなんですが、実現するためのアーキテクチャをここに示します。

まず、下には、関係データベースをもってくる。それも単なる関係データベースだけではいけないので、拡張します。

その上には、オブジェクトレイヤということで、オブジェクトという抽象的なものをもってきます。その上に応用レイヤを乗せる。応用レイヤは何かといいますと、アプリケーションの個別の知識を入れるためのレイヤです。それぞれにインタフェースを設ける。

マルチメディアデータは RDB 上で実現しますのでも既存のデータベースもアクセス可能になります。

最後に、オブジェクトレイヤで具体的に何を実現するかといいますと、フレームによるいわゆる知識表現システムをここに載せます。そのためオブジェクトをフレームで表現します。また、抽象化機構を導入して、類化、汎化、集約、進化などの抽象化機構を導入してやればいいんじゃないかと考えております。

それから、個々のオブジェクトの動作を表現するのに、メソッドを導入する、そしてオブジェクト間のメッセージの通信によりオブジェクトの集まりが全体として統一された動きをします。

以上簡単ですけれども、私が考えておりますマルチ

メディアデータベースシステムの考え方について発表しました。

穂鷹 次に、増永先生お願いします。

増永 図書館情報大学の増永です。



1970年代に入って、リレーショナルデータベースモデルの提案、あるいは CODASYL の共通言語仕様の発表があって、DBMS 技術が長足に進歩したわけです。その結果ビジネスデータ処理も、リレーショナルモデルでカバーできるまで成長し、その後 CAD に代表されます非ビジネスデータ処理にもリレーショナルモデルが適用できないかという形でデータベース技術が展開してきたと思います。それと平行して記憶媒体、通信媒体、入出力媒体技術がいろいろ進歩しまして、いま実世界全部をデータベース化できないかというような視点が出てきていると考えているわけです。それがマルチメディアデータベースの出発点であると。

対象とする実世界をすべてデータベース化したいということで、マルチメディアデータベースを考えますと、私は三つの課題があるのではないかと考えています。まず第一が、シングルメディアベースシステムでも苦しんでる問題だと思いますが、実世界をどのように知覚認知して、概念モデルを構築していくかと。

次に、第二の課題ですが、そのようにして構築されたさまざまなシングルメディアデータベースを、いかに統合していくかということです。

第三番目は、そのようにしてでき上がったマルチメディアデータベースをどう効率よく管理していくかという問題です。

まず、第一番目の問題です。メディアとは、記号(とくにアイコン表記的な)そのものの物理的・化学的屬性とそれが成立する世界と定義します。

つまり、従来ですと文字・数値を使って実世界を記述し、モデル化していたわけですが、その記述に使える記号として新たに図形とか画像とか音とか多様なものが許されるということです。そして、マルチメディアデータベースは、そういう実世界の記号化を極てでき上がったモデルをさらにコンピュータデータベースにするために論理表現モデル—これをビューと呼ぼうと思います—に変換し、最後にさまざまなビューを統合してマルチメディアデータベースができ上がるという具合に考えています。

二番目の統合の問題なんですけれども、私は、オブジェクト概念に基づいた統合を考えているということです。これに第一番目の問題とも関連し若干の説明を加えさせていただきたいんですが、たとえば、ここに1冊の本、たとえば「ピーターパン」という本があり、二つのメディアがあったと。一つは画像メディア、もう一つは文字・数値メディアだったとします。

実世界の概念モデル化にあたっては、メディアごとにアクセプタがあるとします。アクセプタというのは、いうなれば一つのフィルタみたいな装置で、ブルーと赤で書いてる絵を赤のセロファンで見ますと、赤は何も見えなくてブルーの部分だけ見えるという、そういう意味をもっているとします。そうしますと、この例ではピーターパンの絵の部分だけを認識するアクセプタがあり、その結果一つの画像ビューができ上がる。一方もう一つのアクセプタは文字部分しか見えないわけですから、そこではピーターパンの文章部分だけが抽出されて、文字・数値ビューができ上がる。

さて、この二つのビューをどう統合化するかですけれども、この例は非常に簡単ですが、そのよりどころを、実世界、すなわちピーターパンという本もっているオブジェクトの構造に求めるということです。

具体的には、誰かメタ的な認識者がいまして、それが統合デザイナーであるわけですけれども、その統合デザイナーが、このピーターパンという本はオブジェクトの観点からは構造がこういう具合になっていると。たとえば、それはページ1、ページ2、ページNというオブジェクトから成り立っている。さらにページ1は絵の部分というオブジェクトと文章部分というオブジェクトから成り立っていると。

そうすると、ページ1のこのオブジェクトは、実はビューのこのデータ群を指しているという具合にオブジェクトデータベースと各ビュー間にそういう関連を与えてゆける。それが実は統合になっているのではないかと。

もう少し詳しくは、実世界というのは、各メディアごとに認識された最も基本的なデータ群、すなわちビューの上に、いわゆるデータアブストラクションのメカニズムで、抽象データ部が構築されて、それらの全体として一つの拡張ビューが構築され、次にさまざまな拡張ビューに加えて、それらを統合するためのオブジェクトベースが加わり、関連性が与えられ全体として一つのマルチメディアデータベースをなしていくのではないかと考えているわけです。

なお、統合のためのオブジェクトの空間は多様なアプリケーションの視点を許容するものでなければならない。ユーザにはどのようなマルチメディアデータベースが見えるのかという、拡張ビューとオブジェクトデータベースの統合体が見えるわけですからオブジェクトデータベースの要素も各拡張ビュー中のデータと同じく、ユーザアクセスの対象にももちろんなっています。データ言語としてはスーパー言語という統合型の言語を定義していますが、時間の関係でその説明は省略します。

最後にシステムのアーキテクチャですが、基本的に三つの形を提案しています。一つは、いわゆる単一のDBMSアーキテクチャと呼んでいるもので、構築されたマルチメディアデータベースを管理するマルチメディアDBMSをつつくるというものです。

ただしこの場合は、ある意味でオーダメード的なDBMSをつくらないといけないということです。そうしますと、DBMSが管理するマルチメディアのデータベースが変わりますとそれも変わってこないといけないという意味で、DBMSのメディアの独立性が損なわれるわけです。

そこで中間的な段階としまして、まず個々の拡張ビューを管理するDBMSが一つあって、全体としてマルチメディアデータベースを管理するというハイアラキカルなアーキテクチャが考えられると思います。

さて、きょうの招待講演のところで植村さんが、マルチメディア分散データベースシステムという概念をお話になっていましたけれども、マルチメディアと分散というのは切っても切れない関係にありまして、そのような観点から考えますと、第三に提示します協調形のDBMSのアーキテクチャは、大変有効ではないかと思います。すなわちメディアごとのデータベースを管理するメンバDBMS群があって、それらが協調して統合体としてのマルチメディアデータを管理するというアーキテクチャです。

メンバDBMSは管理するデータベースが異なっても共通のインタフェースを持つということで、お互いにコミュニケーションが可能であり、したがってお互いに仕事を分担して、マルチメディアデータベースのアクセス、更新、あるいはメタデータの管理を行っていく体系です。これにはDBMSの参照モデルという考え方が有効かと考えています。以上が私のマルチメディアデータベースに関するイメージです。

穂鷹 それでは最後のスピーカですが、小島さんお

願います。

小島 電子技術総合研究所の小島です。マルチメディアデータベースシステム実現上の問題点について話させていただこうと思います。



マルチメディアのデータを管理したいというのが、マルチメディアデータシステムの目的なんですが、先ほど来のご発表にもありましたとおり、マルチメディアに関しては非常に多くの概念が入り組んでいます。ここでは、問題を二つの面から考えます。まず、マルチメディアデータ管理のためのデータモデルがあるかどうかということと、光ディスクなども含めて、いろんな物理メディアを管理できるようなマルチメディアデータベースシステムの新しいアーキテクチャがあるかどうかと、この2点に関して私の立場を説明させていただこうと思います。

マルチメディアの概念については、画像や音声が取られるのをマルチメディアという。あるいはテキストデータや光ディスクなども新しいメディアとして扱われているわけで、非常にいろいろな概念がまじりあっているわけです。これらをすべて統合管理するのが、マルチメディアデータベースシステムといわれているわけですが、一般のデータベースシステムで、絵が出て、音声が出て、テキストなどが扱われればよいというだけではないと思います。

マルチメディアデータベースシステムという概念が出てきた背景には、たとえば光ディスクであるとか、あるいは高機能のビットマップディスプレイみたいな計算機環境の発達があるわけですが、当然それにもなうデータベースの応用分野の拡大が考えられます。つまり、CAD、OAをはじめとするマルチメディア応用、一般のデータベースシステムでマルチメディアが扱えるだけでなく、それを使った新しい応用みたいなものが考えられて、それを扱えるのをマルチメディアデータベースシステムと呼んでいるのではないかと思います。

そうしますと、新しい応用をなんでもマルチメディアと考えられないこともないわけですが、ここでは光ディスクみたいな新しいデバイスを入れることがある程度その応用に対して有効である応用を、マルチメディア応用といおうと考えています。

したがって一般のデータベースシステムの機能のほかにマルチメディアのデータも扱う応用を扱えるよう

な機能をデータベースシステムが支援しているということが、システムの実現に関しては重要になると思います。

マルチメディアというのは、なんとなくデータの表現形式として考えることができるわけですが、そのマルチメディアをデータ表現として扱うのは、データモデルにおいて行う。さらに、それらを用いた応用に関しては、実現するシステムの機能でカバーしようというのがここでの考え方で。

システム内で扱うデータ型は、いわゆる非コード化データ、音声ですとか画像、あるいは長大レコードとして考えられるものがあります。固有の表現、あるいは固有の操作を持つようなデータ、ファクスのデータみたいに圧縮されたものや、テキストのようなデータ型の組合せによってつくられた複合データや構造データといわれるものも入ります。そういう新しいデータ型、こういうのを扱える必要があります。

システムの機能としては、先ほどあげましたとおりのような応用があるわけですが、基本的には応用依存性が非常に大きい。つまり、あるメディアがあったら、それに対する操作が固有に決まってしまうので、ある応用によってはこのシステムはうまくいくけれども、ある応用にとってはうまくいかないというような依存性が非常に大きいことが考えられます。

したがって、ここではメディアを統合して、変換して、管理する機能や、ある程度異なった応用に関して共通な場を提供するためのビューの支援方法などは基本的な機能として考えています。

さらに光ディスクみたいな新しい物理デバイスの効率的な利用方法も考えられます。ここで考えているシステムは、データベースに関して統一した視点を提供しようとする統合アプローチに基づくもので、画像とかテキストによって固有の視点を提供するものではありません。

この場合、マルチメディア応用は、応用に関する依存性が大きいので、汎用のマルチメディアDBMSは、どういう機能を持つのかは問題となります。

応用依存性の問題に対しては、いろいろなデータ、画像が扱えるとか、音声か扱えるとか、複合データを扱えるとか、マルチメディアデータに関して汎用性があるということと、それを使っているような応用が考えられるんですけど、そのアプリケーションに関して、たとえば履歴データが扱えるとか、そういう機能の要求の面での汎用性が必要になると思うんですが、その

二つで考えた場合に、前者に関しては、マルチメディアデータを扱うデータモデルで扱いますが、後者に関しては、種々の機能を支援するという点で、基本的には応用に依存してしまうと思います。

次に、マルチメディアデータベースのためのデータモデルについて話したいと思います。

基本的には、マルチメディアデータベースのためのデータモデルとして、新しい議論があるかという疑問が出てくるわけですが、私はあると思っています。

たとえば、データ表現を考えると、ユーザが新しいメディアを持ってきた場合に、そのデータタイプとして支援する場合は、従来の文字型、数値型の固定したデータタイプでなくて、ユーザが定義してそれを拡張することができるという、拡張性が必要なわけですし、メディア自身は、たとえば画像データに対する図形の拡大、縮小といった、固有のデータ操作を持つわけですが、それはいままでの組み込み関数のアプローチでは問題があります。したがってなんらかの拡張性が必要ですし、そういう操作には、当然ある操作可能なデータ集合というのを持っていますから、定義域のようなものが扱えないといけないとも考えています。

ここでは意味的データモデル、先ほど来何回も出てきていますけれども、セマンティックデータモデルによるアプローチをとっています。簡単に説明をしますが、まずオブジェクト指向、つまり、あるメディアを固有のデータ表現及び固有のデータ操作を持ったオブジェクトとして考えます。

次に、オブジェクト固有の操作に対して関数な表記を用います。つまり、図形の拡大であるとか、縮小であるとか、あるいはそれに対する面積のようなものも全部あるファンクションとして捉えます。

ここでの大きな特徴としては、そのオブジェクト、あるいはオブジェクト操作の間に、高機能の抽象化を許すことです。たとえば、データ操作の抽象化も許しています。

こういうのは、基本的なセマンティックデータモデルに関しては、若干の拡張になります。その理由としては、オブジェクト指向であって、しかも高度なデータ抽象化を支援するようなデータモデルは、マルチメディア応用に関して不可欠であろうと考えています。

具体的にどういうふうにするかという話をしたほうがいいと思いますが、基本的には、マルチメディアを扱う場合には、内部表現形式の異なった、たとえば図形でもビットイメージであったり、線図形であるよう

に異なったメディアで表現されているオブジェクトがあるわけです。

それに対しては固有の操作が決まりますからオブジェクトとして扱うことは自然だと思えます。さらにユーザに対して、図形という内部表現に独立な視野を提供する場合には、オブジェクトの表現としての抽象化と、もう一つは、ある操作に関して、つまりマルチメディアに関してはメディアごとに操作は固有ですから、オブジェクト表現が抽象化されたときにどういう関係になるのか、明確に定義されていないといけないわけです。たとえば、ビットイメージを打ち出す関数と、線図形を打ち出す関数が汎化されていると、内部表現に独立な出力操作は可能になります。これは実体のない関数であっても、実体があってもなんでもかまわないわけですけれども、そういう形で関数による表記法とそれに対する高度な抽象化の支援が、マルチメディアの応用には不可欠であろうと考えています。

現在は、以上のような立場で考えた場合に、オブジェクトのデータ表現であるとか、データ操作を定義したり、ユーザを改良したりする設計の基準のようなものを考えています。関連する問題としては、データ操作にともなっていくゆるコンピューテッドリレーションみたいな空の属性集合を持つものなどがあるわけですけれども、マルチメディアの応用で基本的な意味的データモデルだけでは不十分であるという立場をとっていますから、これは別に違った結果が得られると思います。

穂鷹 どうもありがとうございました。

ここで10分間のインタミッションとしたいと思えます。その後、各スピーカにもう一度登場していただきます。

(10分休憩)

穂鷹 それではツアーラウンド目を、各スピーカにお願いいたします。先ほど時間がなくて少し急いだ分の補足とか、あるいはほかのスピーカの話聞いたところ、少し考えが違うとか、そういう意見を5分以内ぐらいでやっていただきまして、そのあとフロアから質疑応答その他に移りたいと思えます。

それでは、誠に恐れ入りますが、また同じ順番で、川越スピーカからお願いいたします。

川越 補足させていただくところはマルチメディアデータベースとシングルメディアデータベースがどう違っているかという点です。

まず最初に、マルチメディアデータモデルとして新

しいモデルが必要だとは思っていません。むしろ従来の、先ほどいいましたオブジェクトオリエンテッド、あるいはセマンティックネット的なモデルで十分対応できるだろうと考えています。したがって、そういうところを管理するようなデータ管理部分が、実際のデータベースシステムで加えることが必要だと考えています。

それから、この情報をつくるための手段、構造化という手段がやはり必要で、それはシングルの場合でも重要であるし、マルチメディアデータベースの場合でも重要だと思います。ただそこには、先ほどの例でお話しましたが、あらかじめ構造ができるようなものと、人間が介入しないとできないようなものがあります。そういうものを対話的、あるいは動的に構造化していく必要があると思います。

それから、入力系と出力系との融合、これは複数のメディアを融合させて一つの媒体に出すという意味で、マルチメディア的な色彩があるでしょう。

ですから、マルチメディアデータベースで共通なファシリティというのは、結局構造化と融合、こういったものが必然的に出てくるだろうという認識に立っています。

マルチメディアインタフェースに関しては、いろんなレベルがあるんですけど、一つはビューがサポートされて、いろんなマルチメディアが統合化されて、プログラムなりユーザなりが扱うレベルです。もう一つは構造化されたものに対するインタフェースレベルです。構造化されたものと、元の情報とをこのレベルでインタフェースととれると考えております。

たとえば、応用の自由度が少ないのが問題だという話がありましたけれども、アプリケーションの位置関係をうまく工夫することと、すべてアプリケーション側に任せて、ここでは基本的に構造化や融合という機能、統合的なデータ管理部分というコンポーネント化さえちゃんとやれば、うまくいくという気がします。

次に、さきほど、最後にメディア独立という話をしましたが、これはDBMS自体がメディア独立である必要は必ずしもなくて、メディア独立の部分と、それからメディアに依存したところ、この二つのものを概念層レベルで分離しておく必要がある。当然、分離することによって、インタフェースが違ってきます。要するに、メディア独立な概念構造と、メディアに依存したような概念構造とでは、インタフェースが全く違う。メディアに依存した概念構造のインタフェース

は、当然メディア依存になってしまっている。それに対応した処理も当然出てくる。メディア独立の概念構築のインタフェースは、先ほどおっしゃられたオブジェクトやセマンティックネットワーク的なもので十分対応できるだろうと思っています。

それから、関連情報を持たせることはセマンティックネットワーク的なものでもできるということですが、これは必ずしも同一空間上でやる必要はなくて、あらかじめ関連を分類しておく必要があると思っています。たとえば、空間軸上といいますか、空間を空間とするような場で、複数のメディアを統合化するとか、あるいは時間軸上で統合化するとか、あるいはオブジェクト軸、あるいは属性軸というような軸を設定して、その軸上でオブジェクト間の、あるいはメディア間の関連をとっていくというふうに、グルーピングする必要があると思っています。

穂鷹 どうもありがとうございます。それでは続きまして、大町さんお願いします。

大町 いまの川越さんのお話に少し関連するわけですが、メディア独立というのとはどういうことなのかなということ、川越さんのお話とか、小島さんのお話を伺っているときに考えたんです。小島さんのお話ですと、応用によって、もうオペレーションが決まってしまうんだと、それから、メディアごとにだいたい操作が固有なんだというお話が中に入ったと思うんですが、データベースマネジメントシステムとして、従来の文字・数値情報では、いわゆるデータ独立といったことが一つの目的になっていて、程度の問題なんでしょうけれども、呼ばれているわけですが、それとのアナログというところがおかしいんですが、メディア独立ということがよくいわれているわけです。

たとえば、メディアが文字と音声だとすると、メディア独立というのとはなにかわかるような感じもするんですが、たとえば、画像と文字とか、画像と音声ということになると、メディア独立というのとはどういうことなのかなと。もうメディアの種類が決まってしまうと、決まってしまうというか、画像と文字ということに固定すると、独立ということを一生涯実現しても、ほとんど意味のない、意味のないという言いすぎかもしれませんが、ことなのかなと。

ですから、メディア独立というのは、マルチメディアデータベースの中でどういう意味があるのかなというのは、ここにすわって少し疑問になりましたので、もしどなたか教えていただければ、パネルが質問

するとおかしいかもしれませんが、教えていただければと思います。

穂鷹 非常に結構です。牧之内さん、どうぞ。

牧之内 いま技術部門で、たとえば論理設計をしている部門があるわけですが、そういう部門ではいわゆる論理設計などの図面の管理、共有ということが問題になっているわけです。計算機化はされているわけですが、データベース化はされていないということで、データベース化するにはどうしたらいいんだろうということがいわれているわけです。

ただし、私はただ図面をデータベース化して、管理・共有すればすむものではないんじゃないかと考えているわけです。つまり、設計図を共有するだけじゃなくて、設計のノウハウを共有するようなことでなくてはいけないんじゃないかと、考えています。したがって、物そのものじゃなく、物プラスその物が意味するもの、つまりデータプラスデータが表現するノウハウ、あるいはデータが表現しないノウハウも含めて、そういうノウハウの共有ということにいかないといけないんじゃないかと考えています。これがマルチメディアデータベースが知識ベースであるという主張の背景です。

ところで、私たちが考えるマルチメディアデータベースのシステムイメージはどういうものなのかということ、図に書いてみたいと思いますが(図-3)、いまやメインフレーム1台あればいいということじゃなくて、種々のコンピュータがネットワークでつながれている。そこにはメインフレームもあるでしょうし、いわゆる高機能ワークステーションもあるでしょうし、あるいは光ディスクを使ったような図形処理プロセッサなどもあるでしょう。そういうプロセッサがそれぞれ機能を分担しながらつながっている。ユーザはそういうものを自由にアクセスできなくてはいけません。

ただし、ユーザにとっては、そういうものが分散していることは意識しなくてよるしい。したがって、分散データベースのイメージが一番下にあるわけです。その上にそれらを統合するデータベースがあります。さらにデータに特有のアプリケーションがあります。

そういうものを統合するものとして、オブジェクトがあります。これを私たちは、フレームオブジェクトによって実現します。そして、それを使ってまた、その上にアプリケーションがのります。これがたとえばDSSであったり、文書処理であったり、CADであっ

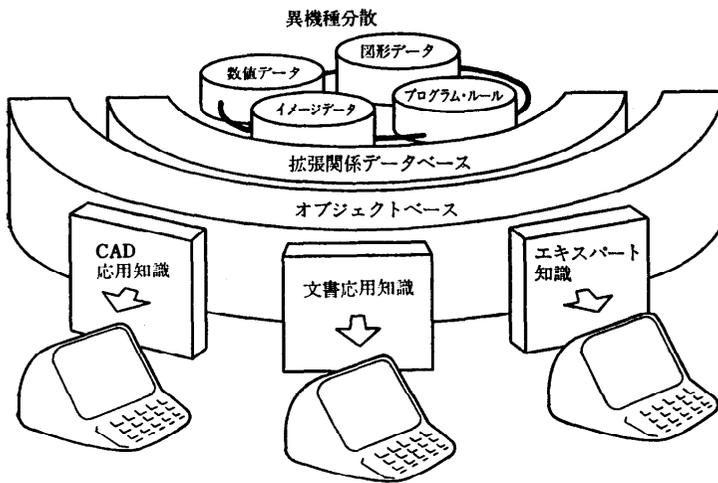


図-3

たり、オフィス業務であったり、プラント設計であったりするわけです。そういうものは固有の知識をもっています。こういう知識というものが、オブジェクトで表されているということです。

システム開発者の立場からいいますと、こういうシステム全体が、セルフデスクリプションでつくれないといけない。したがって、強力な機構と概念が必要である。そういうものをわれわれはフレームに求めているわけです。

リレーショナルデータベースもやはりセルフデスクリプションということから少し拡張したい。入れ子構造とか、ロングフィールドを実装しないといけないと考えています。

一番下は、分散データベース管理システム的なことになっています。各専用プロセッサの能力であるとか、ハウツーユースなんていうのも、やはりオブジェクトで表現するわけです。

どう使ったらいいかということは、ユーザは知らなくてよしい。それはみんな知識としてオブジェクトで表されている。

オブジェクトフレームの強力な表現形式によって大部分がセルフデスクリプションできるということで、つくりやすいシステムじゃないかと考えているわけです。

穂鷹 増永先生、手短かに。

増永 従来のデータベースアクセスは、いわゆる名札中心だったわけです。たとえば、画像を検索するときに、キーの番号とか、いわゆる文字・数値の情報を名札としてつけておいて、それを頼りに所望のデータ

を引っ張ってきていたわけです。ところがマルチメディアデータベースでは、内容による検索というのが必須であると。そう考えましたときに、従来の技術的な展開をみてもみますと、一つはこういう中間段階がある。私はスペシャルインデックスと名づけるのがよいと思えますけど、たとえば、ある空間と交わっているような物体を求めたいという質問を効率よく処理するために、名札なんですけれども、スペシャルな、空間的なインデックス体系を付与しようということです。

その延長線として、最終的に内容中心のアクセスを実現するためにオブジェクトを識別して、それを頼りにデータベースをアクセスしていくことになるのではないか。したがって、オブジェクトオリエンテッドの位置づけなんですけれども、従来の名札中心というアクセスを、いわゆる内容中心にする一つの理想的な姿になっているのではないかと考えているわけです。

先ほど大町さんから、メディア独立というのはいったいどういう具合に考えられるかというお話がございましたけれども、マルチメディアデータベースシステムでは、二つのメディア独立性があるのではないかと考えています。一つは、いわゆる実世界が概念モデル化されて、それが次にビューとして論理表現されるわけですが、異なるビューが統合可能であるためには、ビューはメディアとは独立した記述になっていないといけない。その意味で、ビューはメディア独立でないといけないという観点の一つある。

もう一つは、マルチメディア DBMS というのは、メディア独立でないといけない。その意味は管理するマルチメディアデータベースがどのように変わろうとも、DBMS のメタデータの管理機能、マルチメディア環境での質問処理機能、そしてトランザクション管理の体系、それらは不変であると。そういうメディア独立です。

穂鷹 それでは最後に、小島さんお願いします。

小島 2点ばかり。まず最初は、メディア独立に関してなんですけれども、私が考えていますメディア独立は、二つの概念からなっています。

一つは、データ独立とそれほど変わりはないんです

が、メディア内部実現形式に独立なデータベース表現を用いている。つまり、ユーザがあるデータベースの操作言語を知っていれば、内部的にどのようなスキーマ表現をもっていても関係がないことです。

たとえば関係データベースシステムで画像データベースを実現する場合に、非常に問題になるのは、具体的にラインをどう埋め込むかとか、図形をどういう形で表すかということですが、こういうデータ型に依存した特定のスキーマ表現をとってしまうと、それに対してある操作を行いたいときに、当然データ形式を決めて操作を支援するわけですから、非常に応用に対する依存性が大きくなってしまいます。

もう一つは、メディアに対する拡張性ということですが。たとえば、画像応用専用のデータベースシステムならば、画像と文字・数値という形で固定してもかまわないと思うんですが、汎用のデータベースシステムを考える場合には、ユーザが新たなメディア、新たなデータ型データ操作を組み込むことは十分に考えられます。

そこで、データベースシステム自身がユーザが考えだしたデータタイプやデータ操作を、データベースシステムの枠内で支援できる、という意味で、メディアに対する拡張性があると考えます。この二つの条件をできるだけ満たすのが、メディア独立の目標だと考えています。

次に、この立場から汎用のマルチメディアデータベースシステムを考える場合に、どういう機能がデータベースシステムの基本機能として必要かという問題があります。つまり、一般にいろいろな応用をマルチメディアの名のもとでカバーしようとするとき、いくらかでも機能上の要求が出てくると思います。そこで基本的にマルチメディアの応用に対するシステムの基本機能についての基準みたいなものが重要だと思います。

穂鷹 どうもありがとうございます。それではこれから、パネリスト並びにフロアの方からのディスカッションに移りたいと思います。

まず、横浜国大の有澤先生。

有澤 横浜国立大学の有澤と申します。いまのお話を伺って、やはりよくわからない気がするんですが、先ほど小島さんが、二つの座標軸からお考えになりました。それはマルチメディアデータベースのアーキテクチャという側面。これはおそらくほかの方もご指摘になりましたように、マルチメディア DBMS という以上、シングルのいろいろなメディアを統合化する。

統合化する以上は、その統合化された社会のアーキテクチャが大事になるだろうと、これはよくわかるわけですが、しかし一方、小島さんの指摘されましたように、そういうアーキテクチャをどういうふうにしていくかという、座標とは別に、それではそういうものを統合化する場合のモデルは何かあるんだろうかという議論。

これは前の議論でも何人かの方が、マルチメディア用のデータモデルということをお使いになっておられたと思います。あるいは、それは必要ないと明確におっしゃった方もおられました。

そのモデルに関してご質問したいわけですが、もしも統合化されたアーキテクチャの上に動くマルチメディア DBMS が特別の、つまり従来の DBMS が持っていなかったといえますか、基づいていなかったモデル、あるいはオペレーションというものが必要であるとしたら、それはいったいどういうものなのか、できれば具体的に示していただけないでしょうか、というのが質問の内容であります。

先ほど牧之内さんの話の中で、リレーショナルデータベースの中に拡張フィールドを埋め込むという話がありました。これもまた一つかと思いますが、では単に拡張フィールドという形でデータをエクスパンドしたというだけで済むのか、その場合にリレーショナルオペレーションというもの自体に変更が生じてくるのか、こないのか、そういう問題ですね。

それから、増永先生は、スペシャルインデックスという言葉をお話の中でおっしゃったと思います。スペシャルインデックスというものと、それはモデルとしていままでのモデルではサポートしていなかったようなオペレーションなり、あるいは構成方式なりを必要とするのかどうか、できれば具体的にお話が出ればと思います。

それから、火つけ役であります小島さんにも当然このことに対してどういうふうにお考えになっていらっしゃるか、モデルの話、時間があればとおっしゃいましたけれども、とうとう時間がなかったみたいなのであまりお話しできませんでしたので、その点を。

できれば司会者である穂鷹先生にも、ぜひこれに関してコメントを一ついただければと思います。

このへんのお話は、皆さん非常にホットな話題であるがゆえに、場合によりますと、論文を書いてからというふうには、アイデアを出しおしめされているかとも思いますけれども、できれば忌憚のないお話をお伺い

できればと思います。

穂鷹 いま3人ぐらい弾が当たったと思うんですけど、小島さんが先に当たったようですね。じゃ、お願いします。

小島 データモデル自身がメディア独立を支援することが、マルチメディアのためのデータモデルに関する条件だと思えます。

その枠内で考えたときには、オブジェクト指向ということと抽象化は不可欠であろうと思っています。

有澤 何かオペレーションとして、特別なものが必要であるとか、そういうことはございますか。

小島 応用依存性が非常に大きいので、これが必要であるという形ではいえないと思えます。

むしろ、利用者が一般にデータベースシステムで必要とされているデータ操作以上の操作をつくって、システム内に組み込んで使おうとするときに、システムあるいはモデルがどういう形で支援するかが重要になるんだと思えます。

牧之内 ご質問の主旨は、RDBを拡張するとき、たとえばロングフィールドを入れることによって、何か新しいオペレーションを入れなくちゃいけないんじゃないかということかと思いますが、多分そうだと思います。ただし、私の個人的な考えでは、いわゆるリレーショナルデータベースというような枠組の中での整合性というものは捨てたいと思っています。

したがって、たとえばファンクショナルデペンデンシであるとか、RDBにおける意味論というものは、もうやめたい。われわれがRDBを使うのは、いわゆるインターナルスキーマといいますか、データを格納するファイルとしてです。インタフェースが非常にきれいですので、それを使えば、その上に物をつくるとき非常に楽じゃないかというのが発想の一つです。

したがって、必要なオペレーションはどんどんつけ加えていきたいと考えております。以上です。

増永 スペーシャルインデックスというのは、CADデータベースへのアクセスなどに発想の原点があると思えます。たとえば、ある図面があって、ユーザがある単位的な空間をいつも問題にして、たとえばそういうスペースとインタセクトするような図面中のオブジェクトがあるかないか頻りに聞いてくるとします。これは全くアプリケーションオリエンテッドになってしまうわけですが、この場合そういう形のアクセスがあるということが、最初からわかってるわけですから、その質問を高速に処理するためには、図面中の

オブジェクトと単位スペースが、どういうインタセクトの関係にあるかということ、あらかじめインデックスとして持っていることは大変有効だということです。

有澤 そういう考え方があるということは存じていたわけですが、ほかの講演者は、どちらかというとそういうものはユーザがどんどん組み入れればよいという考え方だったのに対しまして、増永先生の話は、どちらかというとそういうもののうちの何種類かはマルチメディアDBMSのプリミティブであるというニュアンスだったものですから、どれとどれとどれがプリミティブだ、ということまで伺いたいと思ったんです。

増永 具体的なプリミティブの切り出し操作はまだ全くやっていませんが、たとえばマルチメディアDBMSでは、どのようなスペーシャルインデックスでも要求があれば付けてやれる能力が備わっていないければならないということです。

穂鷹 モデルのことについてだけちょっといわせていただきますと、私は自分のモデルを持っているものですから、自分のモデルでマルチメディアDBが扱えないかどうかということに常に反省しているわけですが、なんら問題なくいくと、練習問題としてはちょっと歯ごたえはありますけれども、練習問題のうちというふうに考えています。

もともと私のモデルは、ロジカルなところを論議していますので、そのへんはノートラブルだったと思えます。

小林 産業能率大学の小林でございます。

小島さんのご意見に全く賛成なんです、一つ質問したいと思えます。

おっしゃったことは、結局リレーショナルといいますか、いままでのデータベースの枠組みではたりないところがある。特にリレーショナルですとたりないところがあるのはわかるんで、これはたとえば、ファーストオーダーロジックで書きますと、ファンクションが入れられる。ファーストオーダーロジックのユニバースオブディスクというものは、どんなものでも入るんで、これはマルチメディアであろうが、なんでであろうがかまわないんで、基本的にはどんなデータをもってきてかまわないということなんです、それがリレーショナルになっちゃうと、ファンクションは使えないし、リレーションで表現されている以外のプレジケートというのは、6つのリレーショナルオペレータに制限されるという、これがきわめてきついんじゃない

いかと思っています。

ファンクションのところ、イントリンスティックであっちゃ困るというお話があったわけですが、リレーショナルですと、イントリンスティックなものさえもないわけですし、そいつをイントリンスティックだけでもたりないし、それをエキスパンドブルにしないといけないというお話だろうと伺って、全くそのとおりだと思うわけです。

ADT というアプローチがあるんですが、そのADT のアプローチとおっしゃったことはだいたい同じで、一つ加わっているのは、インヘリタンスを加えると。なぜインヘリタンスと ADT の両方が必要かというと、ファーストオーダのロジックは、ファンクションに対するインヘリタンスを記述することができない。したがって、ファンクションに対するインヘリタンスを記述するためには、どうしてもそこらへんのところを拡張しなければならないということで、いままでの議論と違うところは、ファンクションに対するインヘリタンスをちゃんと拡張したモデルが必要なんだということじゃないかと、私は理解しています。

そういう意味で、おっしゃったことは非常にいいんですが、よくよく考えてみると、それとマルチメディアデータベースとどこが関係があるのか、というのが最後の質問なんです。

小島 インヘリタンスに関する拡張性については小林先生のご指摘のとおりだと思います。後の質問に関しては、マルチメディア応用を支援するには少なくともこれだけの拡張は必要だと思います。これは、データベースの抽象的な理論の場としてはマルチメディアから少し離れた話になる場合もあると思いますが、ここでの議論の結果は具体的にマルチメディアデータベースシステムを実現するときに役に立つものであると考えています。

上林 九州大学の上林でございます。

いまマルチメディアのいろいろお話があったんですけど、きょうの地図のデータベースのときの話のように、使われるためには効率が問題だという話。もう一つは、実際に使われるという、一番最初に何年後に使われるかという話がありましたが、結局経済性が問題であると。

そちらの話がきょうの場合むしろ、モデル化であるとか、どういう機能があるかというところで——ちょっと小島さんの話のところで、応用依存性というのを非常に強調されて、あと大町さんが、いわゆるメディア

独立の意味はどうかということ、やっぱりこれは応用依存性の話だと思うんですが、ほかの方はゼネラルな、マルチメディアシステムはこうあるべきであるという話だったんですけど、現実に3年とか5年とかいわれたときに、そのゼネラルができるというイメージでいわれたのか、それとも個別の、いわゆる経済的かつ効率が高いという意味なのかと。

ですから、マルチメディアデータベースに関する立場がちょっと違ったので、初めの何年というのも少し違うんじゃないかということで、そのあたりもし何かご意見があったら伺いたいと思います。

穂鷹 全員答えなければいけないようですが、では川越さんから。

川越 メディア情報を分類しますと、いわゆるメディア情報の原情報といいますか、たとえばイメージでしたらピクセルの集合体、あるいは文書でいいますと文字の集合体、そういう意味での構造化されていない情報と構造化された情報があります。

従来のデータベースシステムでこういうメディアを扱っているシステムは、どちらかというと、非構造化のデータベースをそのまま管理して、構造化情報を若干つけ加えているというシステムが多いと思います。

マルチメディア情報の本質は、結局構造化情報をどう結合していくか、マルチメディア化されたときにどう結合するかということですが、従来のデータベース管理システムをまず使っていくというアプローチが、まず重要だと思います。

そうしますと、各メディアごとに最適化された格納構造が当然使われるでしょう。たとえば CAD ですと、パフォーマンスのネックがいろいろ処理の場合に発生します。そうすると、その処理ネックに対応したような図形特有の格納構造を当然つくることができそうです。

そうすると、その図形というメディアを扱うDBMS 自体は、非常に最適化された格納構造を持たせることができます。しかし、これはいわゆる結合用のデータベース管理システムの範疇じゃないわけです。

私がいいたいのは、そういうような従来の個々のデータベース管理システムを使って個々に最適化させても、結構使えるものが出てくるんじゃないかと思えます。そういう意味ですと、実用化まで3年から5年だと思います。

ただ、それを包含した形でマルチメディアデータベースを考えていこうとすると、マルチメディア格納

構造最適化という問題に直面して、やはり7年から10年は必要じゃないかという気がします。

大町 前回の予行演習のときに、私は実は欠席いたしました。いま考えますと、川越さんの意見とかなり近いんですが、現状でCADだとか、テキスト処理だとか、専用の従来の非コード化データのアプリケーションが随分進んでるわけですが、ただマルチメディアデータベースの発端であるそれを統合して、エンドユーザが使いたいということを取りあえず実現すると、具体的にそういった各システムをなんらかの意味で結びつけるというようなことでの実現性は、5年くらいに出ないといけないんじゃないか。

さらに、統合的に扱えるような全く新しいものという、いつになるかは残念ながらわかりません。

牧之内 性能の問題についてお答えします。私どもが考えているマルチメディアデータベース管理システムで性能上問題になる点は、まずネットワーク、つまり専用プロセッサ、あるいは汎用プロセッサがネットワークでつながれていますから、ネットワークでいわゆるマルチメディアをどれだけ早く送れるかという問題。

それから、各マルチメディア、たとえば文書は非常に複雑な構造をもっていますので、それをプリミティブなデータ間の関係として表現しないといけない。これは多分下のレイヤで関係データベース上のタップ間の関係として表現されますから、関係データベースの性能の問題。

たとえば画像をとってみますと、画像を取り出してそれを回転するとか、いろいろな処理をするその処理の問題。この三つの問題に分けられるかと思います。

関係データベースに関しては、いまの関係データベースでは遅いであろう。したがって、われわれ自身で早いをつくりそれでこの問題を解決したいと思っています。

ネットワークに関しては、そのうち早いのができてくるんじゃないかと思っています。

画像処理については、実際、今いろいろなところでやっていますから、それも結構早いのが出てくるんじゃないか。そういう意味で、3年後あるいは5年後をみますと、それなりに性能的にはいいものが出そろわないんじゃないかと考えています。そういう意味では心配していません。

やはり一番の問題は、個々のコンポーネントの性能よりそれらをどう統合して、ユーザが使えるようなも

のにするかということかと思っています。

増永 大学は企業と違ってシステム構築能力が弱いものですから、システムをつくり上げるよりはまずマルチメディアはどうあるべきかという話を、できる限りやりたいと考えています。そしてこれは、植村さんがインオペの説明でなさいましたけれども、インオペはシステムをつくることよりも、そのための技術を開発するほうが大きな意義があるとおっしゃっていましたが、私はそういう側面に大いに賛成であります。

5年といったのは、そのぐらいじゃないかと、そういう感じです。

小島 システム実現の手法については応用依存性がありますから、個々に効率的な手法が考案されて、解決されるものだと思います。

ただ、データベースのシステムの集合演算の枠内で考えるのであれば、オブジェクトの表現が変わったりしながらデータが操作されますから、それ自身の独立した議論ができるものと考えています。

システムの実現の期限については、汎用データベースシステムとして出るよりも、たとえばワープロとかそういう形で、音声が入るようになってデータベース機能がついたというように、個々の応用システムでマルチメディアが扱えるようになって、それにデータベース的な機能がついてくるという形のほうが早いのではないかという気がします。

三浦 三井造船の三浦です。

先ほど応用依存性が強すぎるとおっしゃった方、小島さんと増永先生だと思いますが、にお伺いしたいんですが、ある分野を限定してデータベースの問題としてみたときに、プリミティブなファクタがまだないから、データベースを拡張したい、あるいはデータモデルを拡張したい。これは非常に問題としては考えやすいんですが、いまのようにマルチメディアということになりますと、問題を考え出そうとしたら、むしろ応用依存性が強いということで、汎用的な答えがあるのかどうかよくわからないんですね。

DBの問題かということが基本的に私の質問なんですが、たとえば、計算機言語に抽象データ型操作といったものを、つまり強いメカニズムを入れて、むしろそちらのほうから応用依存性を吸収するというアプローチのほうがいいのかなという意見がありそうな気がします。そうじゃなくて、データベースの問題があるというふうに認識されておられる理由をお伺いしたいんですが。

増永 ぼく感じでは、ADT で吸収していくのはきつと言語の処理系が重くなりすぎて、どうしようもないんじゃないかと。

川越 基本的には、マルチメディアデータベース管理システムの基本機能としては、メディアに依存した機能と、メディア独立の機能に分かれると思いますが、それ以外に考えられるファシリティとして、先ほど申しました動的な構造化ツール、あるいは融合するようなファシリティがメディアに共通なツールとしてあると思います。

それを DBMS に入れるかどうかというのは、DBMS の定義の問題であって、どちらに入れたとしてもいいのですが、ただ、そういう機能は当然共通だと思えます。

もう一つの共通機能は、ユーザインタフェースの部分だと思えます。基本的にユーザインタフェースというのは、マルチメディアデータベースシステムという観点からみると、やはり共通だと思えます。しかしそれ以外で応用に依存するところはアプリケーションとして DBMS から独立させたほうがいいでしょう。

したがって共通という点で、先ほどいきました融合化、構造化、インタフェースというのは、データベースシステムの共通部分としては必要です。

先ほどおっしゃられた抽象型データプログラミング言語でサポートされたらいいかという話は、私はよくわかりませんが、それはデータベース的な発想ではなくて、プログラミング的な発想ですので、多くのディスカッションがいます。

小島 汎用のマルチメディアデータベースシステムはどういう機能をもっていなければいけないかについての確かな基準は私自身も持っていません。

プログラミング言語からのアプローチはどうかということに関しては、先ほど川越さんがいわれたとおり、データベース的な考え方でないと思えます。

具体的には、いろいろな応用の異なったメディアなどをデータベースシステムで管理していくときに、たとえば、これはよい設計であるとか、よいデータベースであるとかという形で示すことのできるクラスを与えることが重要だと思えます。

そういうのはプログラミング言語では難しいわけですから、そういうクラスが、たまたまデータベースの汎用と目されるある機能をもった、拡張されたデータベースの言語あるいは管理システムみたいのであればよいというアプローチをとっています。

穂鷹 議論はつきないんですけど、一応5時までということになっていますので、若干司会からまとめを述べまして、終わらせていただきたいと思えます。

全部のまとめではないのですけれども、きょうのパネルディスカッションだけではなくて、発表者の方からも非常に重要な二つの概念が区別されたと思えます。名前はいろいろ違うんですが、一つはイメージ情報というもので、一つはコード情報というものである。このコード情報というのは、オブジェクトというものをアイデンティファイする名前のようなもので、たとえば文字というの、数値というのあるいは、それが構造化されたテキストという形をとる。それに対して、イメージ情報はものそのもので、非文字というの、イメージというものでさまざまに表現される。また、メディア情報に対してはその構成を表現するデータがある。

いままでのデータベースは、主にコード情報を扱っていた。マルチメディアで名前のほかにものそのものが入ってきて、これも扱わねばならない、こういう問題だろうと思えます。

そのときにこの二つのクラスの中のデータの一方から一方の変換を何人かの方が議論されました。イメージのほうから、従来のオブジェクトをアイデンティファイするものへの変換が構造化、あるいは増永先生のビューというものかと思われま。

それから逆に、コード情報テキストのほうからイメージ情報に行くのを、富士通の牧之内さんはフォーマッタ、あるいは川越さんの言葉で、融合という言葉であったように思えます。

この二つの概念は一方を記号系、一方を非記号系といっても良いかもしれませんが、記号系あるいは名前というのが従来のDBの対象であった。そこに非記号系、メディアが新たにDBの対象として入ってきた。記号系に対しての操作は今までであったわけですが、問題は、非記号系に対しての操作がなんであり、モデルがなんであるかということが議論だったかと思えます。それがどういものであるかは、これからの数多くの実験でだんだんと性格が明らかにされるものと思われま。

きょうは、いろいろと活発なご意見をいただきまして、まことにありがとうございました。これでマルチメディアのセッションを終わりたいと思えますが、最後に発表者ならびに議論に加わっていただいたフロアの方に拍手をお願いしたいと思います。(拍手)