

## データフロー型可視化システムにおける 問合せ及び処理関数の設計環境

渡辺 知恵美<sup>†</sup> 織飼 規子<sup>†</sup> 古畑 理香<sup>†</sup> 市川 哲彦<sup>‡</sup>  
<sup>†</sup>お茶の水女子大学大学院 人間文化研究科 数理・情報科学専攻  
<sup>‡</sup>お茶の水女子大学 理学部 情報科学科

〒 112-8610 東京都文京区大塚 2-1-1 Tel:03-5978-5399

Email:{chiemi, noriko, rika2}@imv.is.ocha.ac.jp, ichikawa@hyalos.is.ocha.ac.jp

### 概要

マルチメディアデータベースへの問合せでは、メディア固有の特徴抽出処理やパラメタ設定が重要な役割を果たす。そこで我々は、そのような処理の対話的かつ効果的なプログラミング手法が必要であると考へ、科学技術可視化システムにおいて普及しているデータフロー型の対話的な環境を、問合せ関数の設計に利用するシステムを開発している。開発環境としては可視化システムにソースの公開されている Advanced Khoros 2.2 を、また DBMS としては拡張可能 DBMS の一つである Postgres 6.2.1 を使用している。今回は、Khoros のビジュアルプログラミングの環境を問合せの設計に利用する機能を新たに追加し、システムの強化を図っている。本稿ではシステムの概要と問合せの視覚的な設計環境について述べる。

## Supporting Development of Multimedia Database Queries and Functions via a Modular Visualization System

Chiemi Watanabe<sup>†</sup> Noriko Ukai<sup>†</sup> Rika Furuhata<sup>†</sup> Yoshihiko Ichikawa<sup>‡</sup>

<sup>†</sup>Division of Mathematics and Information Sciences, Graduate School  
of Humanities and Sciences, Ochanomizu University

<sup>‡</sup>Department of Information Sciences, Faculty of Science, Ochanomizu University

### ABSTRACT

The recent proliferation of the multimedia information has increased the needs of quickly supporting newly developed media and technologies in database management systems (DBMSs).

We have been developing a system that integrates a modular visualization system (MVS) and an extensible DBMS; the current prototype has been built on top of PostgreSQL 6.2.1, and Advanced Khoros 2.2.1. Our system has the capability to translate Khoros module definitions to the corresponding PostgreSQL user-defined functions. As well as modules written in C, a network of modules in a restricted form can be converted. In addition to a simple SQL-module issuing SQL-statements, we have designed a module-based query formulation interface, which helps users formulate queries in the modular programming environment of khoros.

## 1 はじめに

近年、画像や音声などのマルチメディアデータが幅広く普及し、それに伴ってマルチメディアデータベース (MMDB) やデータ自身の内容に基づく検索 (CBIR) の研究が盛んに行なわれている。CBIR では一般的に、データから抽出した特徴に基づいて特徴空間を構成し、その特徴空間で定義された距離 (非類似性) を基準にしてデータ選択を行なうという手法が用いられる [8, 15, 6]。

拡張可能 DBMS [9, 14] では、これらの処理を行う関数をユーザ定義関数として登録し、検索に利用することが可能ではある。しかし、ユーザが拡張可能 DBMS でマルチメディアデータの検索を行なう場合、次の問題が生じる。

- ユーザが処理関数を定義するには、関数に対する知識に加え、各々の DBMS が提供する拡張機能やデータ形式について把握しなければならない。また関数の記述のためにはプログラミング技術が必要である。
- マルチメディアの問合せはユーザの視覚的な評価によって決定される。そのため、視覚的かつ対話的な問合せ環境が必要となる。

一方、視覚情報処理の分野では、データフローパラダイムに基づいた可視化システムとして MVS (モジュール型可視化支援システム) が普及している。代表的なものに AVS [16], Khoros [17], apE [5] などがある。MVS はデータフローの概念を用いて可視化プログラムを設計するシステムである。MVS は画像処理のためのモジュールを数多く提供し、ユーザはモジュールを処理の流れにそって繋ぎ合わせネットワークを構成することによって、プログラムを作成することができる。

そこで我々はこの MVS の視覚的・対話的な設計環境を、マルチメディアデータ検索のための関数設計や問合せ設計のために利用するシステムを提案した [1]。本システムでは実装環境として、MVS に Khoros 2.2.1<sup>1</sup> を、拡張可能 DBMS に PostgreSQL 6.2.1<sup>2</sup> を使用している。

本システムは、マルチメディアデータ検索に必要な以下の二段階の設計環境を MVS によって与えている。

- 問合せに用いる処理関数 (特徴抽出関数、距離関数など) の設計環境
- 問合せの設計環境

<sup>1</sup>Khoros は Khoral Research Inc. の登録商標である。

<sup>2</sup>機能については <http://www.postgresql.org> 参照されたい。また、BLOB の実装と関数引数の個数については若干変更を加えている。

今回、問合せの設計環境のための機能を強化し、Khoros の設計環境をより効果的に利用することを可能にした。本稿では、本システムが提供する問合せ及び処理関数の設計環境について述べる。

## 2 Khoros の設計環境

### 2.1 Khoros でのプログラム設計

図 1 は Khoros でのプログラム設計の様子である。このプログラム設計ツールは Cantata と呼ばれる。

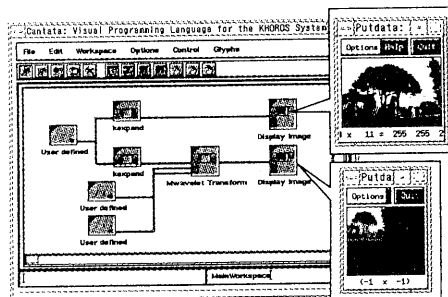


図 1: Khoros におけるプログラム設計の様子

プログラム設計の基本要素となるモジュールを Khoros ではグリフとよぶ。Khoros は基本演算や画像処理、可視化処理など多くのグリフを提供している。利用者は、処理に必要なグリフを Cantata 上に取り出し、それらをつなぎあわせて処理プログラムを設計していく。またグリフはそれぞれ利用者対話部分を持ち、これによってパラメタ情報を与えることができる。プログラムを実行すると、ネットワーク上の流れに沿ってグリフが起動され、データ処理が進められる。利用者はここで実行結果を確認し、必要に応じてグリフを付け変えたり、パラメタを変更しながら対話的に設計を行うことができる。

また、利用者は構成したグリフネットワークの一部をマクロ化し、一つのグリフとして定義することができる。Khoros はこのグリフを “procedure” と呼ぶが、ここでは混乱を避けるためマクログリフと呼ぶことにする。このマクログリフを用いることによって、グリフネットワークを再利用したり、より複雑な処理プログラムを設計することができる。

### 2.2 データ操作における Khoros の構造

プログラミングの基本的な処理を与えるグリフ (以下基本グリフと呼ぶ) は、C などのプログラミング言語で記述されている。これらの基本グリフは Khoros が提供するデータフローおよびデータ操作

のライブラリによってコーディングされている。図2は、図1のグリフネットワークの「断面」を概念的に示したものである。

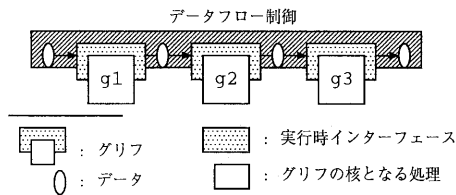


図2：グリフ間のデータフローと制御層の関係

グリフの実行は制御層によって監視されており、グリフネットワークに沿って適宜グリフが起動される。この制御層はデータ生成の終了イベントを監視しているため、グリフ間を結合するチャンネルに流れるデータが完全に生成された時に限り、次のグリフが起動される。また、起動されるグリフのデータアクセスや利用者対話は、Khoros が提供するサービスを經由して行われている。データアクセスサービスは、特定の物理フォーマットで編成されたデータを、より抽象的な配列形式や幾何データ系列として取り扱う機能を提供している。

### 3 システム構成

本システムは前節で紹介した Khoros の対話的な設計環境を、PostgreSQL におけるマルチメディアデータ検索に必要な処理関数及び問合せ設計に用いる。そこで、まず PostgreSQL のプログラム開発手順を Khoros におけるプログラム設計手順に対応づけた。図3はその対応関係を表したものである。PostgreSQL での C や SQL による SQL 関数定義、問合せ構成という開発手順は、Khoros 上においては、基本グリフのプログラム、マクログリフによる基本グリフのマクロ定義、グリフネットワークによる問合せ構成、と対応づけることができる。

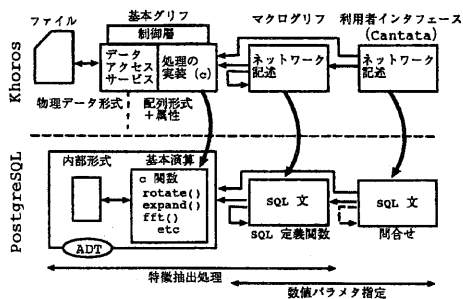


図3：PostgreSQL と Khoros におけるプログラム開発手順の対応関係

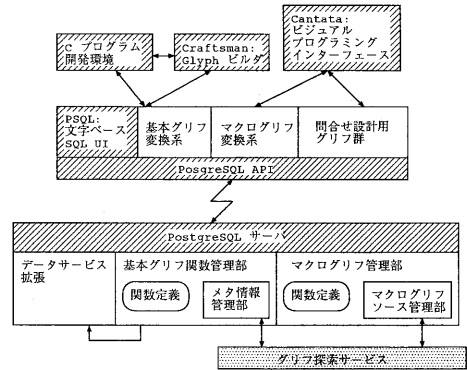


図4：システム構成

この方針に基づいた本システムの構成を図4に示す。斜線を引いた箇所は Khoros あるいは PostgreSQL で既に提供されている機能である。なおグリフ検索サービス(7節参照)は未実装である。本システムは基本グリフ及びマクログリフの変換機能を提供する。これにより、Cantata で設計された処理関数を PostgreSQL のユーザ定義関数として登録し、問合せに用いることができる。またこれらの機能によって変換された関数のデータ処理をサポートするものとして、PostgreSQL 側にデータアクセスサービスを提供している。これは、Khoros のデータアクセスサービスに対応したものであり、変換された関数のデータアクセス要求を処理している。また、グリフネットワークによって問合せを構成するために、PostgreSQL のリレーションに対して関係演算を施すグリフを提供している。第4節と第5節でこれらの機能についてさらに詳しく説明していく。

### 4 グリフ変換機能

基本グリフから SQL 関数への変換処理を図5に示す。基本グリフのプログラムは C などのプログラミング言語で記述され、スタブ生成ツールが生成したファイルに、開発者が必要な処理を書き込む作りになっている。そのため、自動生成された部分と開発者が記述したデータ処理の部分は既定義のコメント文によって区別されている。これを利用して、スタブ生成部分のコードを取り出し、SQL 関数の定義に必要なとされるコードの形式に変換する。ここで、関数の引数の数や型は、基本グリフの UI 仕様ファイルから入出力情報とパラメタ情報を得ることにより決定している。データ処理のコードはそのまま利用する。生成されたプログラムはコンパイルされ、SQL 関数として登録されるが、これと同時に、グリフと関数間の対応関係とグリフのパラメタ情報

がメタ情報として記録される。これらの情報はマクログリフ変換処理の際に利用される。

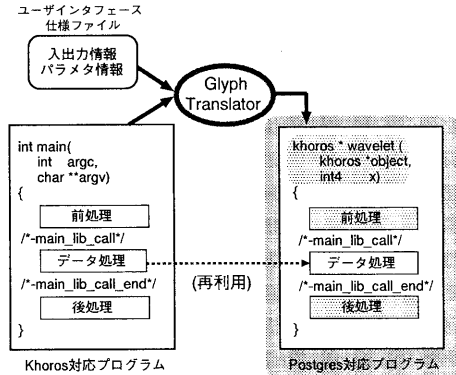


図5：基本グリフの内部変換処理

マクログリフからSQL関数への変換処理を図6に示す。ユーザがCantata上で作成したマクログリフの情報はファイルに記述されている。このファイルには、マクログリフを構成しているグリフの名前とパラメタ情報、それらのグリフ間の接続情報などが記録されている。ここから必要な情報を抽出し、マクログリフを構成しているグリフと予めDBに登録されたSQL関数との対応関係を調べ、マクロに対応したSQL関数を合成していく。

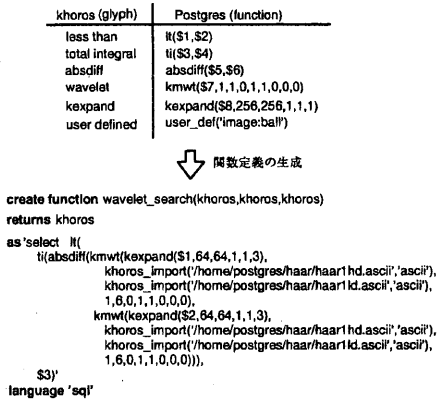


図6：マクログリフの内部変換処理

図7はこの様子をユーザの視点から表したものである。変換したSQL関数をCantata上で利用可能にするためのグリフが自動生成されている。このグリフと他のグリフを組み合わせて新たにマクログリフを生成し、さらにそれをSQL関数として登録することができる。この作業を繰り返すことにより、ユーザの意図に合った関数を設計することが可能である。

またこのグリフはグリフで実行可能な関数名を出力するポートを持つ。これは本システムで問合せを

設計する時に用いられる。

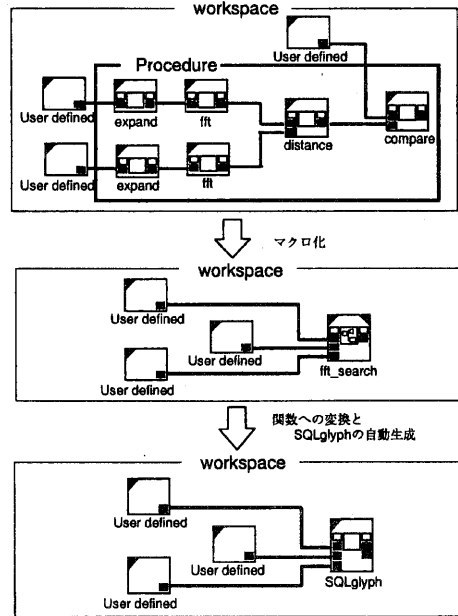


図7：利用者側から見たマクログリフの変換処理

一連の変換処理はC言語を用いて実装している。なお、本機能の実装上の制限等については、文献[1]を参照されたい。

## 5 問合せ設計環境

本システムにおける問合せ設計の様子を図8に示す。Cantataにおけるマルチメディアデータ問合せの視覚的構成の実現のために、本システムではリレーションを選択するグリフと、リレーションに対して関係演算を施すグリフを提供している。

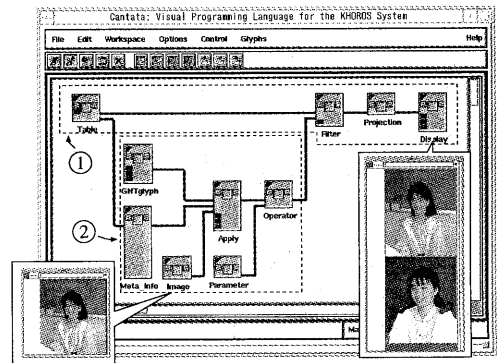


図8：本システムにおける問合せ設計の様子

利用者は、図8①のように関係演算のグリフをつなぎあわせ、問合せのグリフネットワークを構成す

る。また、問合せ条件を図8②のように記述し、関係演算の入力引数とする。このグリフネットワークを実行することで、利用者はネットワークに従ってリレーションに関係演算が施されていくかのような感覚で、目的にあったリレーションの検索を行うことが出来る。

本節では、このような問合せ設計環境を実現する上で問題となる Cantata の機能上の制約について、その概略を述べ、続いて、本システムでどのようにそれらの制約を回避して問合せ設計環境を構築したかについて述べる。

## 5.1 グリフネットワークによる問合せ設計の問題点

Cantata 上で、グリフネットワークによる問合せ設計環境を実現する場合、Khoros のもつ性質上、次のような制約を考慮しなければならない。

- (1) グリフは集合値をサポートしていない  
Cantata 上で問合せ処理を設計する場合、グリフの処理対象はリレーションである。しかし、Khoros は科学技術可視化のシステムとして設計されており、集合値やタプル値には対応していないため、リレーションを扱うのは困難である。
- (2) リレーションの処理にかかる負荷が大きい。  
関係演算グリフは、リレーションに対する処理を PostgreSQL 側で行う。そのため各グリフは演算を施すたびに、Khoros と PostgreSQL の間でリレーションを受渡する必要がある。また、Khoros のデータフロー制御ではグリフが処理を完全に終了するのを待って、次のグリフに一度にデータを渡すことになる。そのため、set-at-a-time の処理を Cantata 上でそのまま実行したのでは、Khoros - Postgres 間やグリフ間におけるプロセス間通信の負担が大きくなってしまう。
- (3) 問合せ条件を入力引数に用いることができない。

関係演算に用いる問合せ条件は、グリフネットワークとして構成する必要がある。しかし、グリフはグリフネットワークのような1階パラメタを取ることができず、グリフネットワークによる処理の結果のみサポートしている。そのため、問合せ条件をグリフネットワークで表しても、選択演算グリフの入力ポートには問合せ条件ではなく、問合せ条件を施した結果のデータが渡されてしまう。

このような問題から、文献[1]のプロトタイプでは Cantata 上で問合せを行う方法として、SQL 文発行グリフを提供していた。SQL 文発行グリフは、利用者が SQL 文を書き、それを Postgres に発行するグリフである。このグリフでは、パラメタ指定や表示の際に Khoros のビジュアルプログラミング機能を利用でき、Cantata 上で視覚的に問合せを行うことが可能ではあった。しかし、この方法は Khoros の対話的な設計環境を効果的に利用することが出来なかった。

## 5.2 本システムにおける問合せ環境

今回、本システムでは上述の問題に対処し、グリフネットワークで問合せを構成することのできる環境を提供している。本項では、本システムがどのようにこれらの問題に対処したかについて述べる。

図9は図8のシステムにおけるグリフの処理の様子を表したものである。まず、グリフ間ではリレーションの代わりに問合せを渡している。関係演算のグリフは、実際に演算を行うのではなく、渡された問合せの書き換え処理を行っている(図9①)。表示グリフは渡された問合せを PostgreSQL に発行し、検索されたリレーションの一覧を表示する。利用者は、関係演算グリフに表示グリフを接続することによってリレーション一覧を表示させることが出来るので、あたかもグリフ間でリレーションが渡され、それに対して関係演算が施されていくかのように問合せを設計することが出来る。この方法により、グリフでリレーションを扱う必要がなくなるため、(1),(2)の問題に対処することが出来る。

また、問題(3)に対しては、条件式構成グリフを提供することで対処している(図9②)。これらのグリフは入力ポートから受け取った情報をもとに条件式を構成していく。これにより、グリフネットワークという1階のデータが、条件式を表した文字列という0階の表現に置き換えられ、グリフの引数として用いることが出来る。本システムは条件式を記述するために必要な以下のグリフを提供している。

- データ入力グリフ … 閾値や比較データなど、問合せに用いる要素を入力するためのグリフ。本システムは画像ファイル入力や、数値入力など、入力データの種類別にそれぞれグリフを提供している。これらのグリフは、文字列や数値など、データ自身が問合せに用いることが出来る場合にはデータの内容を出力し、画像データなど、識別子を問合せに用いる種類のデータについては、識別子を出力している。

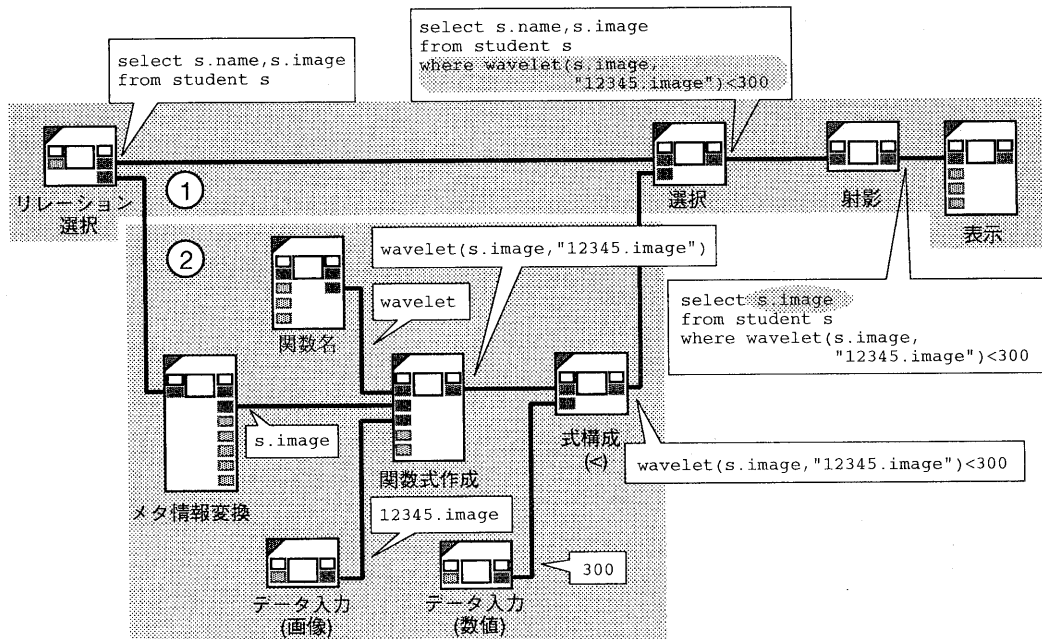


図9：図8における各グリフの処理の様子

- 関数名グリフ… これは、マクログリフ変換機能により生成されたグリフを用いている。ここでは関数名のみを利用している。
- メタ情報変換グリフ… リレーションの情報から属性情報を取り出す。このグリフは渡された問合せ文から属性情報のみを抽出し、用意されている複数の出力ポートに属性情報を振り分けて出力する。
- 関数式作成グリフ… 関数式を構成するグリフ。このグリフは、関数名と引数にするデータ情報を受け取り、関数の引数情報を基に関数式を構成する。関数の引数の型と、入力されたデータの型が一致しない場合にはエラーを返す。グリフ変換機能により変換された関数の情報は、DBに記録されている。このグリフはDBの記録から関数の引数情報を取得している。
- 式構成グリフ… 比較演算子や論理演算子を用いて、入力ポートから受け取ったデータから式を構成する。ここで入力されたデータの型が一致しない場合にはエラーを返す。

これらの手法を用いることによって、問合せをグリフネットワークで構成することが可能となり、利用者はCantata上で対話的に問合せ設計を行うことが出来る。しかし、これらのグリフはデータ自身ではなくデータに対する情報を扱っているため、通常のグリフとは性質が異なる。そのため、通常のグリフを問合せ設計グリフとして用いることが出来ず、

また、問合せ設計用グリフを画像処理設計に用いることも出来ない、という問題点もある。

しかし、マクログリフ変換機能により生成されたグリフについては、処理の結果とメタ情報（関数名）の両方を出力ポートに渡せるようにしているのので、画像処理設計と問合せ処理設計の両方に用いることが出来る。また、逆にKhorosには画像処理用のピクセル単位の論理演算が多数登録されているので、ブール式の構成に名称の異なる専用のグリフを用いることは利用者の混乱を避ける効果があると考えられる。

## 6 関連研究

マルチメディアデータベースの利用は

- (1) 問合せに必要な関数の設計
- (2) 問合せの設計
- (3) 結果の表示と問合せの動的な構成

という段階で行われると考えられる。このうち、マルチメディアデータベース検索の開発者は(1)と(2)を、末端利用者は(2)と(3)を利用する。本システムはMVSを利用することで(1)、(2)の利用を支援している。今回は問合せの視覚的な構成を可能にし、(2)の支援を強化した。(2)の効果的な利用を実現するために、MVSの持つ制約を回避する方向で対処したが、今後ビジュアルプログラミング

の視点で可視化システムを強化し、この制約に対応することも考えなくてはならない。一般的なビジュアルプログラミングシステムとしては VisaVis [11] などがある。これは関数型言語 FP の処理に対してアイコンを対応させることで、高階関数の利用を可能にしたシステムであり、今後本システムを強化するための参考になるとと思われる。

また(2)の問合せの視覚的構成を支援するシステムとして Tioga [13, 2], VQE[4], GQL[10] が挙げられる。Tioga は本システムと同様にデータフローを用いて、関係演算による問合せを視覚化している。また VQE は、E-R ダイアグラムから対話的にデータセットを選択する機能を情報視覚化システム Visage に統合することで、選択されたデータセットの対話的な可視化を可能にしている。また、GQL でも同様なデータセット選択の機能を提供している。VQE では視覚化された情報に対して利用者が探索操作を行うと、その操作が自動的に内部の問合せ処理に反映される動的問合せ機能を持つ。

このような動的問合せの機能は情報の視覚化システムに典型的に見られるもので、近年盛んに研究が行われている [3]。また、SpotFire<sup>3</sup> のように商品化されたものもある。これらのシステムは DB 全体のデータ構成を表示し (overview)、そのなかでの探索処理 (focus and detail-on-demand) を行うことで、ユーザの意図にあったデータの検索を行う。利用者は (3) によってのみデータベースを利用し、(2) の問合せ設計という概念はシステム内部に隠されている。

上述のシステムは末端利用者の利用を対象としており、本システムの目的とは少々異なっている。しかし、本システムの機能をさらに効果的に用いるために、これらのシステムを参考に (3) を強化することが必要であると考えている。ただし、これらのシステムが、少なくとも現状では、文字や数値情報のような (システムレベルでもまた末端利用者にとっても) 意味論の確立したデータのみを扱っている点には注意が必要である。効果的な表示や動的な検索を、マルチメディアデータベースを対象して行おうとした場合には、意味論の決定方法や表示の仕方、また利用者の学習カーブの問題<sup>4</sup>が発生する。この点は (3) を強化する場合に検討すべき課題となると思われる。

<sup>3</sup><http://www.ivee.com>

<sup>4</sup>表示された内容を利用者が理解するまでにかかる時間が文字情報などに比べ視覚情報では著しく増大する問題。

## 7 おわりに

MVS をマルチメディアデータ検索のための関数及び問合せ設計環境として用いたシステムについてその概要を述べ、問合せ設計に MVS を効果的に利用する機能について説明した。このシステムを利用することにより、マルチメディア検索のための一連の設計を行うために、MVS の視覚的かつ対話的な環境を効果的に利用することが出来る。

今後の課題は以下のとおりである。まず、本システムの間合せ設計環境に関しては、検索結果の効果的な表示方法を検討する必要がある。本システムでは、リレーションの表示方法として表形式を用いたが、特徴空間における各データの分布を用いたデータの表示方法なども考えられる。他には利用者のフィードバックを考慮した設計方法も検討される。

また、システムにおけるパフォーマンス強化も必要である。現在これに対する工夫はされていないが、今後本システムの現実的な利用を考えるに当たって、処理の高速化は必要不可欠である。これを実現する方法としては、空間索引 [6]、式計算の最適化 [12]、などが考えられる。

さらに、効率的な利用を行うために、検索に用いられる重みづけパラメータや閾値パラメータを利用の履歴から自動的に設定する機能も考えられる。また、可視化処理に関するメタ情報を DB に記録することで、文献 [7] で提案されたような、可視化技法を含めたグリフ検索サービスを提供すること可能であると考えられる。

## 参考文献

- [1] 市川哲彦, et al. データフロー型可視化システムを用いたマルチメディアデータ処理関数の開発環境。データ工学ワークショップ, 98(35), 1998.
- [2] A. Aiken, et al. Tioga-2: A direct manipulation database visualization environment. In *Proceedings of the 12th International Conference on Data Engineering, New Orleans, LA*, pp. 208-217, February 1996.
- [3] T. Catarci and I. F. Cruz, eds. Special section on information visualization. *ACM SIGMOD RECORD*, 25(4), December 1996.
- [4] M. Derthick, S. F. Roth, and J. Kolojechick. Coordinating declarative queries with a direct manipulation data exploration environment. In *Proceedings of IEEE Information Visualization '97 Symposium, Phoenix, AZ*, pp. 65-72, October 1997.
- [5] D. S. Dyer. A dataflow toolkit for visualization. *IEEE CG & A*, 10(4):60-69, July 1980.
- [6] C. Faloutsos. *Searching Multimedia Databases by Contents*. Kluwer Academic Publishers, 1996.

- [7] I. Fujishiro, et al. GADGET: Goal-oriented application design guidance for modular visualization environments. In *Proceedings of IEEE Visualization '97, Phoenix, AZ*, pp. 245–252, October 1997.
- [8] V. N. Gudivada and V. V. Raghavan. Content-based image retrieval systems. *IEEE Computer*, 28(9):18–23, 1995.
- [9] W. Kim. *Introduction to Object-Oriented Databases*. MIT Press, 1990.
- [10] A. Papantonakis and P. J. H. King. Syntax and semantics of gql, a graphical query language. In *Journal of Visual Languages and Computing*, pp. 3–25, 6 1995.
- [11] J. Poswig, G. Vrankar, and C. Morara. VisaVis: a higher-order functional visual programming language. *Journal of Visual Language and Computing*, 5(1):83–111, 1994.
- [12] M. L. Praveen Seshandri and R. Ramakrishnan. The case for enhanced abstract data types. In *Proceedings of the 23rd International Conference on Very Large Data Bases, Athens, Greece*, pp. 66–75, August 1997.
- [13] M. Stonebraker, et al. Tioga: Providing data management support for scientific visualization applications. In *Proceedings of the 19th International Conference on Very Large Data Bases, Dublin, Ireland*, pp. 25–38, August 1993.
- [14] M. Stonebraker and D. Moore. *Object-Relational DBMSs: the Next Great Wave*. Morgan Kaufmann, 1996.
- [15] V. S. Subrahmanian and S. Jajodia, eds. *Multimedia Database Systems: Issues and Research Directions*. Springer-Verlag, 1996.
- [16] C. Upson, et al. The application visualization system: A computational environment for scientific visualization. *IEEE CG & A*, 9(4):30–42, July 1989.
- [17] M. Yound, D. Argiro, and S. Kubica. Cantata: Visual programming environment for the Khoros system. *ACM Computer Graphics*, 29(2):22–24, May 1995.