

視覚情報を用いたオブジェクト指向によるRDB検索法

加藤哲朗，鈴木幸市，織田敬三，外川政夫
NTT 電気通信研究所

本報告では、最新のビットマップディスプレイ技術を背景に、視覚情報を用いてオブジェクト指向の Relational Data Bases (RDB) 検索法を実現した OBAQ (Organized Bitmap Assist Query) を提案する。

OBAQでのRDB検索操作は、「データベース接近のための操作」と「データベース編集のための操作」からなる。前者は、データディクショナリ上のデータベースをマウスにより直接指定するための操作であり、VDD (Visual Data Dictionary) により実現している。後者は、マウス、メニュー、フォームを用いて表形式データベースの表編集処理を行うための操作であり、VDE (Visual Data Editor) により実現している。

本検索法は、マウス、アイコン、図形情報を駆使し、人間の操作の思考に近い問合せ応答形式の視覚操作インタフェースを提供することによって、従来 of SQL, QBE (Query - By - Example) が提供するキーボード入力をベースとしたマンマシンインタフェースの欠点を克服している。

A VISUAL INTERFACE FOR AN OBJECT-ORIENTED QUERY
IN RELATIONAL DATA BASES

Akio KATO, Kouichi SUZUKI, Keizo ODA and Masao TOGAWA

NTT Electrical Communications Laboratories, NTT Corporation

1-2356, Take, Yokosuka, Kanagawa, 238-03 Japan

This paper proposes OBAQ (Organized Bitmap Assist Query), an object-oriented query with visual user interface using bitmap graphics.

OBAQ consists of VDD (Visual Data Dictionary) and VDE (Visual Data Editor) because RDB (Relational Data Bases) query can be categorized into two components, access operations and edit operations.

VDD provides access operations to the data base by means of pointing visualized objects in a data dictionary by a mouse. VDE provides edit operations on each table by using a mouse, menus and forms effectively in similar way to editing tables in word processing.

This system, which provides the human interface of a Q-A interaction using mouse/icons/graphics tools on a screen, is easier to use than keyboard languages such as SQL and QBE (Query-By-Example).

1. はじめに

OA化の進展と共に、従来のホストセンタ（HC）での集中・定形的なデータベース処理に加え、新たに非定形業務用のデータベースなどを高速構内網（LAN）内に散在するワークステーション（WS）やファイルステーション（FS）に分散配置し、それらに関係データベースで分散管理する形態が出現しつつある。また、データベース利用者層も拡大し、データ処理従事者のみならずオフィスワーカーなどの一般利用者が、ネットワーク内の任意端末から簡易な操作でデータベースを自由に入出力できることが要求されている⁽¹⁾。既に、関係データベースでは、データベースの定義・操作を行うための自然なマンマシンインタフェースとして、文字ベースの端末を対象にしたSQL、QBE（Query-By-Example）が開発され、高い評価を受けている。

本論文では、最新のビットマップディスプレイ技術を背景に、マウス、アイコン、図形情報を駆使した高度なRDB検索用のマンマシンインタフェース（OBAQ：注1）を提案する。さらに、本提案方式は、人間の操作の思考に近い問合せ応答形式のオブジェクト検索や、操作性に優れた表作成編集操作を可能としており、SQL、QBEの従来方式に比べ、マンマシンインタフェースが優れていることを示す。

2. 従来の検索技術

関係データベースの代表的なコマンド操作検索法にSQL⁽²⁾、ビジュアル操作検索法にQBE⁽³⁾がある。

SQLをマンマシンの観点から改善したのがQBEであるが、QBEでは、表示がビジュアルになったが、オブジェクト名、検索条件の入力は、従来のSQLと同様キーボードから直接入力する必要があり、また、検索のため入力するオブジェクト名の取得方法までは、規定していないという意味で、更に、改善の余地が残されている⁽⁴⁾。以下に、SQL、QBEの検索技術を概括する。

2.1 SQL

SQLは、コマンドによるデータベース検索言語であり、コマンドをタイプ入力することによって、検索結果を表形式で出力する。代表的なSELECTステートメントのコマンド例は、以下の通りである。

```
SELECT カラム名 [,カラム名…]  
FROM テーブル名 [,テーブル名…]  
WHERE カラム名 OP. '値'
```

ここで、OP. 比較演算 (=, <, >等)

このように、SQLコマンドを記述するためには、SELECT等のキーワード、データ操作の対象を示すテーブル/ビュー名、テーブル/ビューの中のどのカラムを表示するかを示すカラム名、及び検索条件としてカラム名と比較演算子及びその値を入力する必要がある。

その結果、SQLでは、コマンドの入力構文を覚えたり、入力時スペル誤りが多発したり、検索結果を更に絞り込む操作ができないなどの欠点がある。

2.2 QBE

QBEは、SQLとほぼ同一機能水準をもつ、表形式によるデータベース検索言語である。QBEの特徴は、図2.1に示すような表骨格（Table skeleton）を画面に表示させ、これに画面エディタを用いて骨格に表されている関係と属性に名前を与え、また、各行に検索コマンド、及び検索条件を示す組を与え問合せを行う⁽⁵⁾。

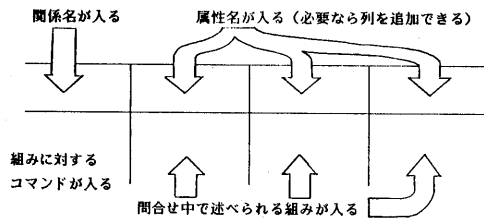


図2.1 QBEの表骨格

例えば、下記SELECTステートメントは、図2.2のように問合せする。

```
SELECT NAME  
FROM EMP  
WHERE SAL>10000,DEPT=TOY
```

EMP	NAME	SAL	DEPT
	P.	>10000	TOY

図2.2 QBEにおける問合せ例

このように、QBEでは、検索や表示を行うために、表形式を用いているため、SQLより自然にデータベース操作が可能であるが、テーブル名、カラム名、コマンド、条件式をキーボード入力しなければならないという点では、本質的にSQLと変わらず、入力時の煩わしさ、検索テーブル名の取得法などに問題がある。

(注1) Organized Bitmap Assist Query

3. OBAQの設計概念

3.1 狙い

OBAQは、SQL、QBEが持つマンマシン上の操作性の欠点を克服するため、ビットマップディスプレイ、マウス、マルチウィンドウを積極的に利用することによって、コンピュータ操作に馴染んでいない利用者向けに、視覚的かつ対話形式で関係データベースの入出力及び編集操作を可能とすることを旨とする。さらに、一層の操作性向上を狙いに以下の目標を設定し、SQLと同等の機能サポートを目標とする。

- ① SQL構文などの文法知識を不要にすること。
- ② キーボード操作は、データ入力などの必要最小限とすること。

上記に示すような高度な関係データベース操作機能を実現するため、OBAQでは次の2つの基本概念を導入する。

- ① オブジェクト指向
- ② 2次元空間操作

3.2 オブジェクト指向

データベース設計におけるスキーマの階層構造を図3.1に示す。

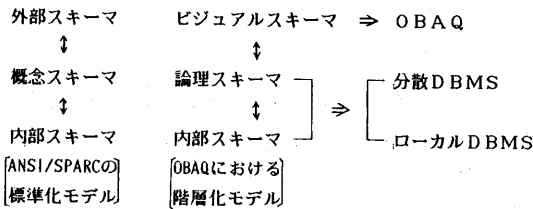


図3.1 スキーマの階層構造

関係データベースの概念スキーマは、2次元の表構造で表現し、関係データベース操作は、その表操作により行う。更に、関係データベースは、ビューという副スキーマ（概念スキーマの一部の抽象モデル）を定義可能であり、同様に2次元の表構造で表現する。このことから、関係データベースは、構造的側面からは、全て表という単一の構造体の集合体で表現し、視覚上はテーブルもビューもオブジェクトという統一した抽象概念で扱うことが可能である。

従って、関係データベースの操作は、図3.2に示すような視覚化されたオブジェクト（ビジュアルスキーマ）に対する操作で置き換えることができる。

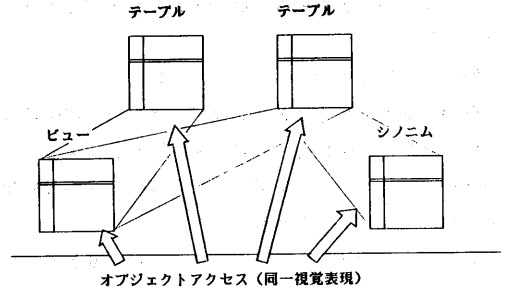


図3.2 オブジェクトに対する操作

以上より、OBAQでは、関係データベースの操作は、視覚表現化されたオブジェクト（ビジュアルスキーマ）からの特定オブジェクトの選択操作、及びその選択操作によって可視化されるオブジェクト実体（データベース実体）に対するデータ操作で構成することができる。

3.3 2次元空間操作

ユーザフレンドリなインターフェースの実現においては、視覚表現方法、操作指示方法が問題となる。

3.3.1 視覚表現方法

(1) 図形表現

視覚表現法には、抽象的な幾何学図形と具象的な幾何学図形を用いる方法とがある。種別が多い場合は、直感的で分かり易く名称入力等の補助情報が不要な具象的な幾何学図形を利用するのが良い。最近、パソコン等で良く利用されているアイコンは、ほとんど具象的な幾何学図形を利用している。

このため、OBAQでは、視覚表現のために具象的な幾何学図形を用いる。

(2) ビジュアルスキーマの視覚表現

ビジュアルスキーマは、オブジェクトの集合体であり、オブジェクトには、ビュー、テーブル、シノニムがある。

これらをディスプレイのような限られた2次元の視覚空間の中で表現する場合には、ユーザは、内部スキーマ（実際の格納構造）や概念スキーマに関係なく、使用特性や嗜好に合わせ、自由にオブジェクトのアクセス構造（ビジュアルスキーマ）を設定できることが望ましい。則ち、直接利用するユーザが、データベースのアクセス頻度、見易さ等を考慮し、自由にグループ化、階層化できるようにするのが良い。

一般に、オブジェクトの集合体は、共通の属性を有するものをクラスで分類し、必要ならば上位方向/下位方

向にクラスの階層化を行い認知の容易化を図る手法を採っている。

例えば、分散データベースのように膨大なオブジェクト群からなる世界では、ビジュアルスキーマの構造は、ロケーション、所有者、DB情報内容などによりクラス分けし、さらにアクセス頻度等によりクラスの多層化を行う方法などが採られる。

このため、OBAQでは、OBAQが対象とするオブジェクト及びそれらのクラスの視覚構造表現体として、以下のシンボル（複写が可能であるが、自らはDB実体を持たない視覚構造体）の定義を許す。

- ・クラス : フォルダのシンボル
- ・オブジェクト : テーブル、ビュー、シノニム
のシンボル

また視覚空間の限定されるディスプレイは、図3.3に示すように、クラスを表示の基本単位とする。

これにより、マルチウィンドウ表示を可能とするディスプレイでは、複数のクラスやオブジェクト実体を重畳表示することが可能なことから、ビジュアルスキーマ全体を同時に同一画面上に展開することができ、これらのクラスやオブジェクトの選択は、全てマウス操作で実行することができる。

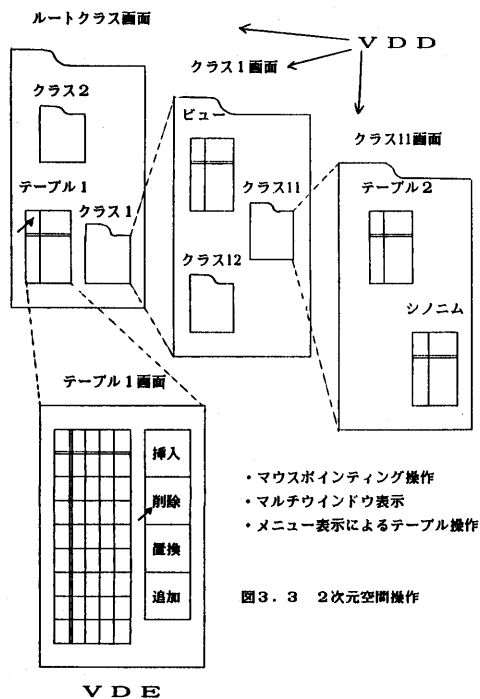


図3.3 2次元空間操作

(3) オペレーションの視覚表現

関係データベースの操作（オペレーション）とは、オブジェクトの定義・生成・参照・更新などのSQLコマンドを指示することをいう。

このSQLコマンドの指示は、オブジェクトの存在それ自身に関する指示と、オブジェクト実体が有する情報の中身に対する指示とに分けることができる。

一般に、オブジェクトの定義・生成は前者であり、オブジェクトの参照・更新は後者である。

これらのオペレーションの視覚表現法には、① アイコン表示、② メニュー表示があり、これらは、表3.1に示すような特長を有している。

表3.1 アイコンとメニューの特長

項目	アイコン表示	メニュー表示
種類	<ul style="list-style-type: none"> ・抽象幾何学図形 ・具象幾何学図形 	<ul style="list-style-type: none"> ・プルダウンメニュー ・ポップアップメニュー
特長	<ul style="list-style-type: none"> ・図形表示 ・移動/並び換え自由 ・非操作時でも表示（通常） 	<ul style="list-style-type: none"> ・文字表示 ・テーブル形式で固定 ・操作時のみ表示（通常）
適用	<ul style="list-style-type: none"> ・オペレーションの視覚表現 ・オブジェクトの視覚表現 	<ul style="list-style-type: none"> ・オペレーションの視覚表現
利害特質	<ul style="list-style-type: none"> ・独自の表示スペースが必要 ・直接選択のため操作簡単 ・アイコンの移動によりアイコン間の干渉が可能 	<ul style="list-style-type: none"> ・独自の表示スペースが不要 ・メニュー表示後選択が必要

OBAQでは、その特長を最大限に引き出すため、オブジェクトそのものの操作にかかわるものは、数も少ないことからアイコン表示で行い、オブジェクト内のデータ操作のようなものは、メニュー表示で行うこととする。

3.3.2 操作指示方法

(1) マウスの基本操作

マウスの効用は、自由に任意の位置をポインティングできる点にあり、視覚表現化したシンボル、アイコン、メニューの選択・移動用に用いる。

マウスのボタン数は1~3個のものが良く利用されており、このボタンで種々の操作指示を行う。特に多用されているのが図3.4に示すボタン2個のマウスであるボタンの操作規則は、次の4基本操作からなる。

- ・ ボタンを押す（押したままにする）
- ・ ボタンを離す（押したままの状態から離す）
- ・ ボタンをクリックする（押してすぐに離す）

- ・ ボタンをダブルクリックする（続けて2回クリックする）

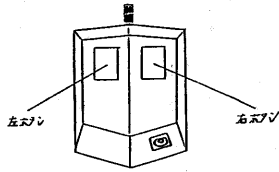


図3.4 マウス

(2) OBAQの操作指示

OBAQでは、上記4基本操作をマウスの右ボタンと左ボタンで使い分けることにより、操作指示を行う。操作は、表3.2に示すような基本パターンで構成する。

表3.2 OBAQにおけるマウス操作

オペレーション	マウス操作
オブジェクトの選択	左ボタンのクリック
オブジェクトの移動	左ボタンを押す、離す（ドロースルーと呼ぶ）
シンボルの複写	左ボタンを押す、離す（ドロースルーと呼ぶ）
アイコンの選択	左ボタンのクリック
アイコンの移動	左ボタンを押す、離す（ドロースルーと呼ぶ）
アイコンサービスを起動	左ボタンのダブルクリック
メニューの表示・選択	右ボタンを押す、離す（ドロースルーと呼ぶ）

3.4 OBAQの処理構造

関係データベースをディスプレイ上で操作するためには、つぎの2つの基本処理が必要である。

- ① 検索処理：処理対象オブジェクトの可視化
- ② 編集処理：可視オブジェクトに対する編集

それぞれの基本処理では、以下の条件が、要求される。

(1) 検索処理

検索処理は、アクセス対象のオブジェクトを格納媒体から読出し、可視状態にすることである。その検索過程で必要となるデータベースのロケーション情報、リモートアクセスのための通信手段情報、DBMSのデータディクショナリなどはシステム内に隠べいし、直接ユーザから見えないようにするのが良い。

また、アクセスしたいオブジェクトの可視化は、表示されたオブジェクト群の中から直接ポインティングすることにより可能とするのが良い。

(2) 編集処理

可視化されるオブジェクトは関係データベースであるので、表形式の視覚表現構造で表現するのが自然である。このオブジェクトに対するデータ操作は、操作を単純化するため、直接データ入力する場合を除いて、オブジェクトあるいは操作メニューへのマウスによる直接ポインティング操作により実行するのが良い。

4. OBAQの実現方式

OBAQは、設計概念で述べた基本処理構造により、VDD (Visual Data Dictionary) とVDE (Visual Data Editor) の2階層のソフトウェア構成で実現する。以下にその実現方式について述べる。

4.1 VDD

4.1.1 主要機能

ネットワークデータベース内のオブジェクト（テーブル、ビュー等）に視覚的表現を与え、ユーザがマウスで指定したオブジェクトに対する直接的な検索手段を提供する。このため、VDDは、視覚構造表現体と各ノードの関係DBMSのデータディクショナリが管理するオブジェクト実体との間の対応づけを管理する。

VDDは、オブジェクト実体と視覚構造表現体を陽に関係づけてその管理を行うために、以下の主要な機能を具備する。

- ・ 視覚構造表現体（シンボル、アイコン）の定義手段
- ・ 視覚構造表現体（シンボル）とオブジェクト実体との対応づけ
- ・ 視覚構造表現体（アイコン）の意味するオペレーションの実行

VDDでは、オブジェクトはシンボルで、オペレーションはアイコンで視覚表現化する。VDDが視覚表現化したシンボル、アイコンを図4.1、表4.1に示す。また、操作画面イメージを図4.2に示す。

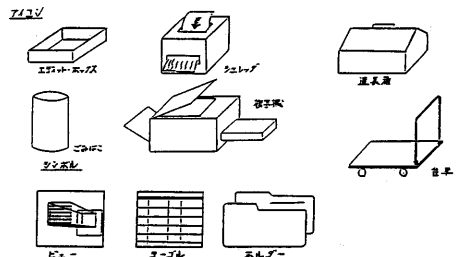


図4.1 VDDのシンボルとアイコン

表4.1 VDDの視覚構造型表現

種別	視覚構造型表現	意味内容
シンボル	フォルダ	クラス（ファイルキャビネット）を表現
	テーブル	テーブルを表現
	ビュー	ビューを表現
アイコン	エディットボックス	VDE（編集処理）への移行処理
	複写機	シンボルのコピー（生成）
	シュレッダ	シンボル及びオブジェクト実体の消去
	ごみ箱	シンボルの消去（オブジェクト実体は保存）
	道具箱	工具類（ユーティリティ類）の実行
	台車	シンボル（含オブジェクト実体）の移動

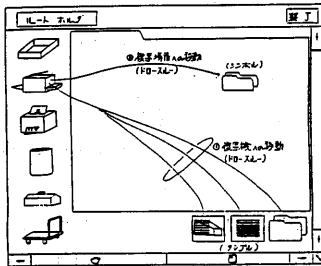


図4.2 VDDの操作画面イメージ

4.1.2 基本操作

VDDの画面上の操作は、ほとんどマウスにより行い、以下の基本操作の組合せで行う。

① 移動

画面上で、アイコン、シンボルを別の位置に移動させる。このとき、シンボルの移動先がアイコンであれば、「そこに入れる」あるいは「何らかの仕事させる」ことを意味する。移動は、マウスをドラッグすることで行う。

② 指定

画面上で、マウスによりシンボルを選択あるいはアイコンサービスを実行させる。指定は、マウスをダブルクリックすることで行う。

③ 登録

複写機でコピーしたシンボルとオブジェクト実体との対応づけを、そのシンボルにテーブル名、ビュー名等のオブジェクト名付与で行う。オブジェクト名入力は、キーボード入力である。

まず、VDDが起動されると、図4.2のような初期画面が表示される。このとき、既に、シンボルが登録されていれば、図3.3に示すようなクラス画面が表示されるので、必要なオブジェクトをマウスで指定すれば、目的とするオブジェクトにアクセスできる。

VDEによりオブジェクトを編集したいときは、対象となるオブジェクトをエディット・ボックスへ移動し、エディット・ボックスを指定する。

4.2 VDE

4.2.1 主要機能

VDEは、VDDにより検索・表示されたオブジェクトの実体（テーブルあるいはビュー）に対して、データを挿入したり、更新したり、削除したりすることにより、オブジェクト実体の編集操作を行う手段を提供する。

この場合、編集操作のためのオペレーションは、表示される操作指示コマンドのポップアップメニューを、マウスで選択・実行することによって、操作性向上を図っている。

実際の編集操作においては、ポップアップメニューで編集操作オペレーションを指示する他に、操作対象オブジェクト上の操作（カラム操作、タブ操作など）やSQLコマンドの条件式の指定等が必要となる。

これらは、それぞれの操作オペレーションモードの中で、マウスによるオブジェクト上のデータ操作や補助メニュー表示・選択などを用いて実現する。

また、複数オブジェクトの同時操作（テーブル間の操作）をする場合には、各オブジェクトを別ウィンドウに表示し、ウィンドウ間の転送機能等を利用して編集処理を行う。

VDEの操作画面イメージを図4.3に、VDEがサポートする主要なSQLコマンドを表4.2に示す。これらのコマンドは、ポップアップメニューで表示・選択される。

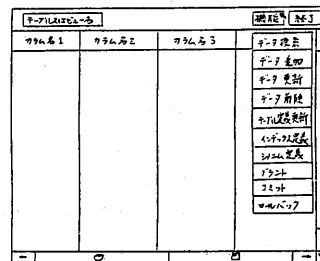


図4.3 VDEの操作画面イメージ

表4.2 VDEの操作オペレーション

SQLコマンド	実行処理
CREATE	テーブル、ビュー、シノニム、インデックスの生成
ALTER	テーブルのカラム追加
GRANT	テーブル、ビューのグラント
COMMIT	トランザクションのコミット
SERECT	テーブル、ビューのデータ照会
INSERT	テーブル、ビューのデータ追加
UPDATE	テーブル、ビューのデータ変更
DELETE	テーブル、ビューのデータ消去

4.2.2 基本操作

VDEは、リレーショナルデータベースをカラム(列)とタプル(行)から構成される「表」として操作する。すなわち、「表」上のカラム、タプルに情報を設定したり、そこから情報を読み取ったりして、リレーショナルデータベースの照会や更新を行う。このように、VDEでは、データベースのデータそのものを取扱うので、データ投入や変更のためのキーボード操作がかなりのウェイトを占める。以下に、基本操作について述べる。

① ポップアップメニューの選択

表4.2に示した各オペレーションの選択を行う。本操作は、マウスを右ボタンでドロースルーして指定する。

② テーブル/ビュー内のカラム選択

カラム移動、削除等の操作の際、どのカラムを選ぶか指定する。本操作は、マウスをカラム位置に移動し、左ボタンクリックで行う。

③ フォーム内の項目設定

カラム属性や検索条件を設定する。本操作は、文字入力に関しては、文字入力枠でマウスの左ボタンクリックした後、キーボードより文字入力する。また、項目選択に関しては、マウスの左ボタンで選択項目をクリックする。

④ データ入力

テーブル内のカラム、データベースの入力を行う。本操作は、テーブル内の入力フィールドをマウスの左ボタンクリックで指定した後、キーボードより入力する。

4.2.3 検索条件の指定

検索条件は、SQLのWHERE文で指定される条件であり、その関係演算は、カラム名、関係演算子(比較演算等)、及びデータ値により記述できる。

カラム名は、表示されているテーブルのカラム名をマウスで指定することにより可能であるが、関係演算子、及びデータ値は、外部から入力する必要があり、何らかの効率的な入力方法を設計する必要がある。

VDEでは、図4.4に示すようなフォームを表示し、マウスによる関係演算子の選択、キーボードによるデータ値入力枠へのデータ値入力を行うこととした。

図4.4 検索条件の選択フォーム

5. OBAQの評価

5.1 総合評価

以上述べてきたOBAQと、SQL、QBEを総合評価すると、表5-1のようになる。

OBAQは、SQLに比べ機能、性能をそれほど劣化させずに操作性向上を追求した点が特徴である。

表5-1 OBAQの評価

比較項目	SQL	QBE	OBAQ
操作性	×	△	○
移植性	○	△	×
機能	○	○	△
性能	○	△	△
開発費	○	△	×

5.2 OBAQの制限条件

OBAQは、文書処理ソフトウェアが取扱う表編集処理操作とSQL操作との類似性に着目し、SQL操作を表編集処理操作で置換することにより実現する。従って、リレーショナルDBMSにアクセスするためには、VDEを利用してユーザが操作するRDB操作をSQL言語

に変換する必要があり、SQL 言語の制約を受ける。

表5-2にOBAQの制限条件をしめす。

また、検索条件については、図4.4に示すフォームだけでは、SQLのすべての検索条件がサポート出来ないという問題もある。

尚、検索条件の充足性不備については、直接 WHERE 文をキーボード入力することで補うことで可能としている。

表5-2 SQL 言語上の制約条件

表編集処理操作	SQL 操作	VOE
サーチ	SELECT FROM [WHERE]	○
行追加	INSERT INTO	○
行消去	DELETE	○
行変更	UPDATE	○
カラム追加	ALTER	○
カラム削除	不可	×
カラム並び換え	SELECT カラム名 ..	○
行並び換え	不可	×
表生成	CREATE TABLE + INSERT	○

6. まとめ

本論文が提案するOBAQは、マウス、アイコン、図形情報を駆使し、人間の操作の思考に近い問合せ応答形式の視覚操作インタフェースを提供している点に特長がある。これにより、従来のSQL、QBEが提供するキーボード入力をベースとしたマンマシンインタフェースの欠点を克服することが期待される。

一方、複雑な検索条件指定によるRDB操作をサポートしようとする、かえってマンマシンの煩雑さを招く結果となること、OBAQを実現するためには、新しいウィンドウ管理やフォーム管理システムの操作支援環境が必要であり、それらを利用するためにマシン依存性が強くなり移植性が悪くなることなどの問題もある。

このため、OBAQの導入に際しては、マシン環境、ユーザ要望に合わせて、適用領域、サポート機能を選択していく必要がある。

謝辞

日頃からご指導いただくNTT複合通信研究所事業所

通信研究部、釜江部長、柏村首席研究員、分散処理プログラム研究室、宮沢室長及び関係各位に謝意を表します。

参考文献

- (1) 外川・鈴木・川瀬：複合分散型データベース設計法に関する一考察，第33回情報全大，7H-2，1986
- (2) M.W.Blasgen: "System R: An Architecture Overview," IBM System J., Vol.20, No.1, 1981, pp.41-62.
- (3) M.M.Zloof: "Query-by-Example: A Database Language," IBM System J., Vol.16, No.4, 1977, pp.324-343.
- (4) P.Reisner: "Human Factors Studies of Database Query Languages: A Survey and Assessment," Computing Survey, Vol.13, No.1, March 1981, pp.13-31.
- (5) J.D.Ullman 著，國井，大保 訳："データベースの原理," 日本コンピュータ協会，1985