

1Q-1

抽象データ型を用いたデータベースにおける高度な表示機能の実現方法

有川正俊 九州大学大型計算機センター
上林彌彦 九州大学工学部

1. まえがき

最近, 視覚的プログラミング, 科学計算における視覚化 (ViSC) など, 人間の優れた視覚能力を生かすための計算機との会話形態が非常に注目されている. この動向は, 計算機との会話装置が文字型から高機能グラフィック型へと変遷していく過程で必然的に起こったと考えられる.

データベースでも, データの視覚化機能が重要視され始めている. 現在の市販のデータベースの表示機能は, 文字データに対するものが中心であり, 選択された文字や数値データを文字図形として出力する, あるいは1レコードごとに予め設定しておいた表示形式に出力するものがほとんどである. 地図やグラフのような高度な表示は, データベースから選択されたデータを応用プログラムに通す方式で実現されてきた. 選択したデータ (あるいは, データの組合せ) に適した応用プログラムを利用者が選択することは, 実に煩わしいことである. 一般的な表示方法が決まっているデータに対しては, 抽象データ型を用いて, そのデータの振舞いとして, デフォルト的な表示方法を規定すべきである. 本論文では, このような観点から, データベースの高度な表示機能を実現するために, どのように抽象データ型を適用するかについて述べる. 議論の例として, 文字・数値データであるが, 視覚データとしての性質が強い地理・統計データを用いる.

2. 表示を強化したデータベースの利用形態

表示機能を強化したデータベースとは, 具体的に, どのようなものであるかを, 図1の関係"県"を用いて説明する.

Table with 4 columns: 名前, 温度, 形状. Rows: 福岡 (20), 佐賀 (21), 長崎 (23).

図1. 関係"県"

関係"県"は, 属性"名前(文字型)", "平均気温(数値型)", "形状(数値可変長配列型)"により構成されている. この関係に対し, 次のようなSQL風の各種質問とその答え(表示)の例を, 図2~5に示す.

select 名前, 温度
from 県;

Table showing query results for 'select 名前, 温度 from 県;'. Columns: 名前, 温度. Rows: 福岡 20, 佐賀 21, 長崎 22.

図2. 文字型属性の表示

select 形状
from 県
where 名前="福岡";

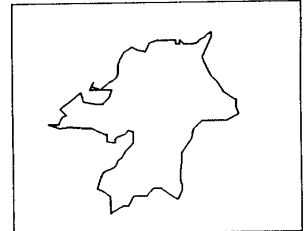


図3. 数値可変長配列型属性の表示

select 名前, 形状
from 県
where 名前="福岡";



図4. 2属性1組の表示

select 名前, 温度, 形状
from 県;

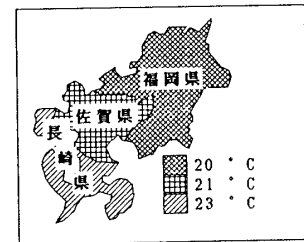


図5. 3属性3組の表示

3. 1スロットオブジェクトの表示

一般のデータベースでは, 文字, 整数, 実数などの組み込み型が用意されている. この組み込み型の役割は, 2進表現のデータの解釈方法を規定することにある. 例えば, 文字型データに対する表示操作は, そのデータ値に対応する文字図形を表示装置に出力することであり, 数値型データに対する表示操作は, そのデータを文字型データに変換し, この変換された文字型データを表示装置に出力することである. このように, 同じ"表示"操作でも, 型が違えば起動する手続き(以下, メソッド)が異なるのである.

オブジェクト指向データベース[1]では, 組み込み型以外に, 利用者が新たな型(抽象データ型)を定義できる機能が用意されている. 抽象データ型の定義は, その型に適用できる操作の定義, それぞれの操作に対するメソッドとデータ構造の定義から構成されている. 以下に, 抽象データ型の定義例を記す.

(抽象データ型 Polygon の定義例)
define abstract\_data\_type Polygon
slots
points: list of Numeric\_Type\_Data

operations

```
display: draw_polygon( points, BOLD )
```

この抽象データ型"Polygon"を、図1の関係"県"の1属性"形状"の"型"として用いると、図3のような表示を実現できる。属性"形状"に対する表示操作は、単に多くの数値を表示させるよりも、むしろ全体的な形状を見せるほうが自然であるので、この属性に抽象データ型"Polygon"を適用すべきである。

ここで、以後の議論のために用語を簡単に説明する。

スロット：関係データベースでの"属性"に対応する。

データの入れ物の基本単位。

オブジェクト：スロットの集合。実体に相当する。

型：スロットとオブジェクトの振舞いを規定する。

クラス：同じ型のオブジェクトの集合。Smalltalkでの

クラスのように、型を表す概念として用いない。

上記の用語を用いると、図3は、クラス"県"に属する1オブジェクトの1スロット"形状"の値を表示した例である。このように、実際の表示は、オブジェクト単位で行うのではなく、オブジェクトを構成する全スロットの部分集合で行う。選択操作により生じた、表示の単位としてのオブジェクトを"表示オブジェクト"と呼ぶ。この節では、表示オブジェクトが1スロットから構成されている場合について述べた。

#### 4. 複数スロットオブジェクトの表示

表示オブジェクトが1スロットだけで構成されているのは特殊であり、一般には複数スロットで構成されている。複数のスロットは、複数の部分図形に対応し、次の原則に従って表示を実現する必要がある。

「同じオブジェクトの複数のスロットのそれぞれの図形を表示する場合は、それらの図形が同一オブジェクトを表現しているのが分かるようにかためて配置する。」

図2では、同一オブジェクトのスロットの文字図形同士はかためて、別のオブジェクトのスロットの文字図形同士は分離して、表示を構成している。図4では、同一オブジェクトの2つのスロット値を視覚化した面図形と文字図形を、どのように配置すべきかを示している。この場合は、文字図形を面図形の中に配置させている<sup>[2]</sup>。このような表示を実現するためには、単に1スロットだけに対する抽象データ型を用意するだけでは不十分である。つまり、図4の場合、面図形型スロットと文字型スロットで構成される表示オブジェクトに対する型（つまり、表示メソッド）を定義する必要がある。既に、2つのスロットに対する視覚化の方法は、各スロットに対する型で定義されているので、この2つのスロットを組み合わせた表示オブジェクトの型では、それぞれの部分図形の相対配置方法を記述する。

同一オブジェクトの複数スロットを1つの（部分）図形として表示する場合もある。例えば、図5は、スロット"温度"の値を（文字）図形で表すのではなく、多角形の図形属性"内部パターン"として用いることにより、スロット"形状"とスロット"温度"の2つのスロット値から1つの部分図形を構成した例である。

このように、選択されたスロットの組合せにより、それぞれのスロットの表示表現が異なってくる。図1のクラス"県"の中のスロット"形状"は、多角形としてしか表現されないが、スロット"温度"は、表示する組合せにスロット"県"が含まれる場合は、多角形の内部パターンとして表現され、

組合せが文字型スロット"名前"だけの場合は、文字図形として表現される。この例では、関係"県"の全スロットの中で、スロット"形状"が表現の決定力が最も強い。現実のメソッドの選択方法には、データの優先順位が関係してくる。

#### 5. 複数オブジェクトの表示

1回の表示で、複数のオブジェクトが表示される方が一般的である。4節では、同じオブジェクトの中の複数のスロットの表示図形は、1つのオブジェクトを表現するようにかためるという原則を上げた。逆に、異なるオブジェクトのスロットの表示図形は、区別できるように分離して配置しなければならない。図2のように、同じオブジェクトの中に含まれる2スロットの表示（文字）図形は、同じ列に接近させて配置させ、異なるオブジェクトのスロットの表示図形は、分離させて表示させている。

また、複数オブジェクトの表示では、図5のように、表示すべき複数の異なるオブジェクト全てが表示域に含まれるように表示装置の座標系を設定する必要がある。

4節・5節で述べたように、抽象データ型の記述は、1オブジェクト（スロット）だけでは、不十分であり、複数のオブジェクト間で定義すべきものである。このような概念を参考文献[3]では、オブジェクト空間に対する抽象データ型として扱っている。

#### 6. 複数クラスの表示

現実の地図は、1つのクラスのオブジェクトだけで、表示が構成されていることはない。例えば、クラス"県"、"川"、"道"、"市"など、複数のクラスの複数のオブジェクトから構成される。複数クラスを表示する場合に、複数オブジェクトを表示する場合には考慮する必要がなかった問題が生じる。例えば、ある川とある道が重なるような問題であり、どのクラスのオブジェクトを優先的に上に重ねるかを決定するための総括的な（知識的な）処理を必要とする。その他、クラス間の不整合性の問題も生じる。例えば、クラス"福岡市内の道"とクラス"合衆国の主要都市"を選択した場合に、これらの2つのクラスを同時に1枚の画面に表示するのは不適當である。このように、現在表示しているオブジェクトとは不整合な関係にあるクラスのオブジェクトを表示する場合は、現表示を消す、あるいは別の表示領域に表示するなどの処理が必要である。

#### 7. むすび

スロット、オブジェクト、クラスという概念を用いて、表示機能に対する抽象データ型（データの選択操作に対する、表示のためのメソッドの起動のさせ方）を議論した。我々は、地理データベースの構築方法を研究しており、オブジェクト指向が、地理データに、図形の自動合成機能を持たせるのに必要な概念であると考えている。

#### 参考文献

1. L. A. Rowe and M. R. Stonebracker, "The POSTGRES Data Model", Proc. 13th Int. Conf. on Very Large Databases, Boston, 1987, pp83-96.
2. 今井浩, 青沼裕美 他, "図形データベースにおける属性文字配置問題について", 昭和62年電子情報通信学会情報・システム部門 全国大会, 2-254頁.
3. D. G. Bobrow and et. al., "CommonLoops", OOPSLA' 86 Proc., 1986, pp17-29.
4. A. H. Robinson and et. al., "Element of Cartography", John Wiley & Sons, 1978.