

4 A A - 2

## 協調作業用データベースの 階層ビューインタフェース

古川哲也

九州大学経済学部

### 1 はじめに

複数の利用者が共同で作業を行う際に、作業用データを管理し協調的に作業するためには、データベースを中心として作業環境を整える必要がある。データ全体の中から作業の対象となるデータや作業に必要なデータを取り出し、各人の作業にあったビューを構築することで作業環境が整備される。利用者がデータ全体を把握するには、データが階層的に分類されることが望ましい。本稿では、協調作業を支援するためのデータベースにおける階層構造に基づくビューについて議論する。

### 2 階層ビューへの要求

データの分類は、分類されたデータ集合をさらに分類することで階層構造となる。階層構造をビューとしたときには、次の点を考慮しなければならない。

- (1) 階層の多様性：階層の定義方法は様々であり、同一レベルであっても異なる手法で定義される場合もある。
- (2) 階層の不均一性：階層構造は、すべてのデータに対して同レベルで定義されるとは限らない。一部のデータに対してのみ階層が定義される場合もある。
- (3) 多重ビューの支援：分類のカテゴリは作業目的によって異なるため、それぞれの作業に合ったデータの階層が必要である。すなわち、複数の階層構造を提供しなければならない。
- (4) 階層の再構成：作業の進展に伴ってデータの挿入や削除が発生する。また、データに対する視点も変化する。そのような作業の進展に合わせて階層構造を変更できなければならない。

データの階層に関する研究には、データの持つ属性値やデータ間の対応関係を用いてデータを階層的に出力したり、階層構造のビューを構築するものがある。このような方法は、スキーマに基づいて階層構造を作り出すものである。しかし、スキーマに基づく階層構造の構成では、階層の多様性や不均一性に対応することはできない。データベースで集合自体を扱う研究も行

われているが、値を集合値に拡張する研究がほとんどであり、集合の階層構造をデータベースで支援しようとするものは知られていない。構造化していないデータベースに関しては、オブジェクトの対応関係を用いて階層構造を形成する研究<sup>[1]</sup>等がある。

本稿では、様々な手法で定義された階層を組み合わせることにより柔軟な階層構造のビューを構成する方法について検討する。個々の階層をデータとすることで、階層の多様性や不均一性に対応し、その合成により複数の利用目的に合った階層を支援する。また、階層をデータベースのスキーマから独立させることで、階層の再構成に対する処理も容易になる。

### 3 分類によるデータの階層

対象となるデータをオブジェクト  $o$ 、分類によって生成されたオブジェクトの集まりをグループ  $g$  とし、 $m(g)$  でグループ  $g$  に属するオブジェクトの集合を表す。  
[定義1] グループの集合  $D_g = \{g_1, g_2, \dots, g_m\}$  で、 $g_i$  ( $1 \leq i \leq m$ ) が  $m(g)$  に対して定義されているとき、 $D_g$  は  $g$  の分割であるといい、グループ  $g$  が  $S_g = \{g_1, g_2, \dots, g_m\}$  の和集合として定義されている、すなわち  $m(g) = \bigcup_{i=1}^m m(g_i)$  であるとき、 $g$  は  $S_g$  の統合であるという。□

分割と統合を複数回適用することによってデータの分類が行われる。分割と統合により、グループの包含関係の階層ができる。作業目的によっては、分割や統合の順序が異なる階層を必要とすることがある。本システムでは、独立した基準による階層をデータとして記憶し、その合成によって作業に必要な階層構造を生成する。

[例1] 全オブジェクトからなるグループ  $g_U$  に対し、独立した2つの分割の基準  $\{g_1, g_2, g_3\}$  および  $\{g_a, g_b\}$  があれば、それは独立した階層  $H_1, H_2$  となる (図1 (a), (b))。  $H_2$  の  $g_2$  と  $g_3$  を統合し  $g_{23}$  とすると、階層  $H_3$  ができる (図1 (c))。作業によっては分割の順序や分割を適用するグループの異なった階層  $H_4, H_5$  が要求されることがある (図1 (d), (e))。ここで、 $g_{xn} = g_x \cap g_n$  である。データベースには、 $H_1, H_2, H_3$  を階層データとして記憶し、 $H_4, H_5$  は階層データを合成したビューとして提供する。□

Hierarchical View Interface for Cooperative Work  
Databases

Tetsuya Furukawa

Dept. Economic Engineering, Kyushu University

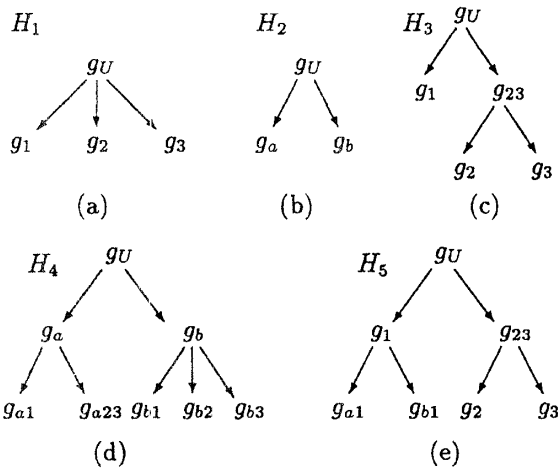


図1 階層データと階層ビュー

### 4 階層ビューの構成

データベース中の階層を用いて利用目的にあったビューを構成する。ビューは、データ集合に対して必要な視点に基づいた階層構造とする。データベース中で定義されたグループによる階層をグループ階層として定義する。

[定義2] グループ集合  $g$  に対し、 $g$  が  $g$  の分割、または  $g$  が  $g$  の統合のとき、 $(g, g)$  を原子階層、 $g$  を原子階層の親、 $g$  の要素を原子階層の子という。原子階層の集合  $H = \{(g_i, g_i) \mid 1 \leq i \leq k\}$  は、その親がすべて異なり ( $g_i \neq g_j, (i \neq j)$ )、 $H$  中の他の原子階層の子とはならない親が唯一であるとき、グループ階層であるという。 $H$  中の原子階層の親グループとはならない子グループを  $H$  の子グループといい、 $H$  の子グループの集合を  $leaf(H)$  で表す。他の原子階層の子グループとはならない親グループを  $H$  の親グループといい、 $H$  の親グループを  $root(H)$  で表す。□

階層ビューは、作業環境に応じた順序で複数個のグループ階層を組み合わせて生成する。グループ階層の合成を次のように定義する。

[定義3] グループ  $g_0$  によるグループ  $g$  の制限  $g_0 * g$  は、 $m(g') = m(g_0) \cap m(g)$  となるグループ  $g'$  である。グループ集合  $g$  に対し、 $g_0 * g = \{g_0 * g \mid g \in g\}$  とする。原子階層  $(g, g)$  の制限  $g_0 * (g, g)$  は、階層  $(g', g')$  ( $g' = g_0 * g, g' = g_0 * g$ ) である。グループ  $g_0$  によるグループ階層  $H$  の制限  $g_0 * H$  は、グループ階層  $\{(g, g) \mid (g, g) = g_0 * (g', g') (g' \neq root(H)), (g, g) = (g_0, g_0 * g') (g' = root(H)), (g', g') \in H\}$  である。グループ階層  $H, H'$  の合成  $H * H'$  は、 $H \cup \bigcup_{g \in leaf(H)} g * H'$  である。また、 $H(g_1 * H_1, \dots, g_n * H_n)$  ( $g_i \in leaf(H)$ ) で  $H \cup \bigcup_{i=1}^n g_i * H_i$  を表す。 $H_i = H' (1 \leq i \leq n), leaf(H) = \{g_1, \dots, g_n\}$  のとき  $H(g_1 * H_1, \dots, g_n * H_n) = H * H'$

である。□

グループ階層の合成演算は結合則が成り立つ。すなわち  $(H_1 * H_2) * H_3 = H_1 * (H_2 * H_3)$  である。階層ビューの定義では、グループ階層の順序のみが必要であり、合成演算の適用順序は問題とならない。

データベースの階層ビューは、グループ階層を演算  $*$  を用いて合成した結果のグループ階層である。階層ビューは仮想的なグループ階層なので、すでに定義された他の階層ビューを用いた再帰的な定義も可能となる。

[例2] 図1 (d) の階層構造  $H_4$  は、階層ビュー  $H_2(g_a * H_3, g_b * H_1)$  として定義できる。図1 (e) の階層構造  $H_5$  は、階層ビュー  $H_3(g_1 * H_2)$  である。□

### 5 階層ビューインタフェース

グループ階層はオブジェクトが実際に分類された階層構造であり、階層ビューは分類の仮想的な構造である。利用者は作業内容に適した階層ビューを構築し、必要なオブジェクトを取り出す。データベースの出力は、指定されたグループのオブジェクト集合となる。すなわち、利用者は指定した階層ビューを通じてオブジェクトを利用する。

階層ビューの会話的な利用を可能とするため、カレントグループの概念を導入する。カレントグループは、階層中の特定の1つのグループで、現在注目されているものである。カレントグループに対する操作として、次のものを準備する。

- DOWN  $g$ : 子グループ  $g$  をカレントグループとする。
- UP: 親グループをカレントグループとする。
- GET: オブジェクト集合を出力する。

会話的な階層ビューの利用では、ビューの動的な構成も可能となる。グループ階層の集合をメニューとして準備しておき、選択したグループ階層  $H$  をカレントグループ  $g$  に適用する。カレントグループが階層の子グループであれば、 $g$  は  $g * H$  によって置き換えられる。 $g$  が子グループでないとき、 $g$  以下の部分階層  $H_g$  を  $g * H$  とするか  $H_g * H$  とするかの選択がある。

### 6 むすび

協調作業データベースで必要となる階層ビューインタフェースの論理的な枠組を示した。データベース内の階層データやビュー定義の記憶法、インタフェースのグラフィカルな実現が重要である。

### 参考文献

[1] Buneman, P., et al., "A Query Language and Optimization Techniques for Unstructured Data," *Proc. ACM SIGMOD*, pp. 505-516, June 1996.