

DB情報ビジュアル化へのOODBの適用法*

5W-8

塩原 寿子[†] 鬼塚 真[‡] 磯部 成二[§]NTT情報通信研究所[¶]

1 はじめに

データベースに蓄積された大量の文字情報を戦略的に活用するためには、図形等によりビジュアルに表現することが効果的である。我々は、このようなビジュアル化アプリケーションの構築を支援するためにINFOVISER[1][2]¹を提案している。INFOVISERでは、ビジュアル化情報の格納にOODBを採用し、ユーザが要求するビジュアル化パターンへの柔軟な対応を可能とした。本稿では、このためのOODBの適用法について述べる。

2 INFOVISERのビジュアル化パターン

文字情報の持つ意味をビジュアルに表現するために、INFOVISERでは、ビジュアル化モデルとして“ノード・ラインビューモデル”を採用した[2]。これは、ビジュアル化したい実体の情報をノード型とライン型という2種類の単純なオブジェクトで表現するモデルである。このモデルでは、ユーザは下記のビジュアル化条件を指定できる。

1. 情報源DBのどの実体をビジュアル化するかを選択
2. 選択した実体をどのオブジェクト型（ノード/ライン）で表示するかを指定
3. 表示オブジェクトのレイアウトパターン、および属性（形状や座標等）が表現する実体情報

さらに、表示に用いるGUIツールを自由に選択できる。図1に、このモデルを通信網に適用したビジュアル化パターンの変更例を示す。

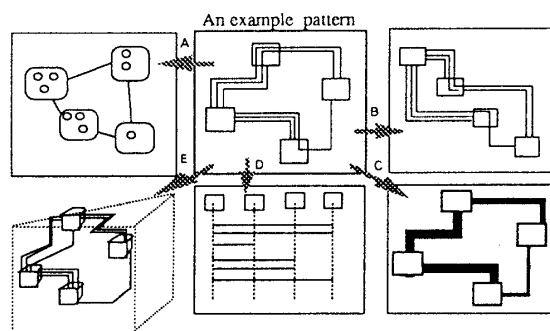


図1: 表示パターンの変更例

3 OODB適用の課題

表示のためのオブジェクト情報を格納するため、INFOVISERではOODBを採用した。この時、オブジェクトのクラス定義には下記の情報が必要である。

- ビジュアル化する実体の属性（実体属性）と実体間の関係
- 指定された実体のオブジェクト型とその表示に必要な属性項目（表示属性；形、色、大きさ、画面上の座標等）
- 3の表示属性の値を生成するときに参照する実体属性
- GUIツールに固有形式の表示情報

しかし、2章で述べたビジュアル化条件の指定を、ユーザが自由に変更するためには、上記1~3の定義情報を、動的に変更できなければならない。しかし、クラス定義を変更すると再コンパイルのためにビジュアル表示に時間がかかりすぎるという問題があった。

4 オブジェクトクラスの構成

上記の問題に対し、ビジュアル化条件の指定変更によるクラス定義への影響を最小限とするため、INFOVISERでは、次の3つのオブジェクトクラスを導入した。図2に概観を示す。

*Applying OODB to Database Information Visualization

[†]Shiohara Hisako[‡]Onizuka Makoto[§]Isobe Seiji[¶]NTT Information and Communication Systems Laboratories¹INFORMATION VISualization EnviRonment

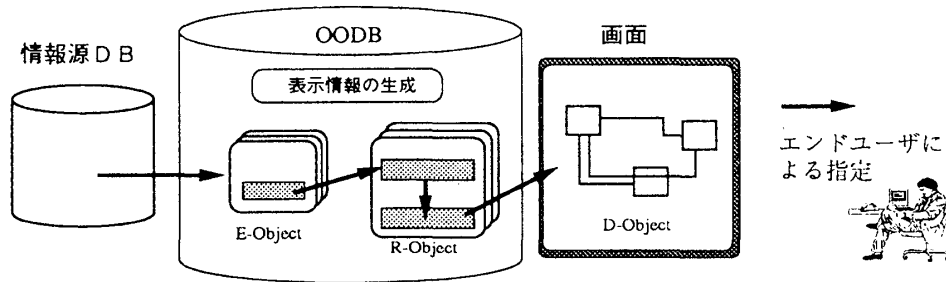


図 2: INFOVISER のアーキテクチャ

E-object ユーザがビジュアル化したい実体の属性情報と実体間の関係を表すオブジェクト。このインスタンスは、OODB に格納される。

R-object D-object の表示に必要な情報を生成するクラス。ユーザの指定したビジュアル化条件にしたがって、E-object の属性値を元に表示情報を生成する。

D-object 画面上に表示される図形オブジェクト。基本的にはノード型とライン型の 2 種類で、R-object 中の表示情報が、GUI ツール固有の表現に変換されている。

これにより、各ビジュアル化条件は独立に変更され、変更の影響を最小限にすることが可能となった。

5 R-object のクラス設計

R-object には、ユーザの指定した実体属性を参照し、表示情報を生成するメソッドが実装される。したがって、参照関係を変更したときに、その影響が最小限となる設計が求められる。

5.1 表示情報生成メソッドの独立化

本メソッド中にユーザによって変更される実体属性との関係を定義すると、変更の都度メソッドのコンパイルが必要となる。これを避けるため、参照属性を導入し実体属性のポインタを格納することにより、間接的に参照関係を定義する方式とした。これにより、メソッド定義は固定化され、ビジュアル化条件変更の影響が最小限となった。

5.2 E-object とのリンク

表示情報を生成するには、E-object の情報が必要であるが、これを R-object にどう取り入れるかに関して、

1. E-object のクラスから継承し、R-object の中に情報を全て持つ。
2. 委譲を用いて E-object のインスタンスに対し、リンクを持つ。

の二つの方法が考えられる。1 の場合は、コンパイルが 1 度ですむが、複数のユーザ間で表示実体を共有する場合、情報が冗長になるという欠点がある。2 の場合は、E-object と R-object を両方コンパイルしなければならない反面、R-object は、E-object のポインタを持つだけでよく、共有が容易であるという特徴がある。したがって、E-object に関しては委譲を採用した。

6 まとめ

DB 情報を効果的にビジュアル化する INFOVISER において、ユーザの要求に応じてビジュアル化パターンを可変にするための OODB の適用法とオブジェクト設計法について考察した。また、今後の課題として、ユーザの意図に合った E-object インスタンス間のリンク構成法について検討していく。

参考文献

- [1] Isobe et al. "Object-Oriented Information Visualization Environment for Network Management" ECOOP'94 Workshop 7 Proceeding, 1994.
- [2] 磯部成二他 "ネットワーク情報ビジュアル化方式の検討" 電子情報通信学会 IN 研究会, SSE94-34, 1994