

データベース設計支援システムDESSERTのERエディタ

3R-8

西山 智

小花 貞夫

国際電信電話株式会社

1. はじめに

近年、データベースの利用は極めて多様化、複雑化、大規模化してきており、それに伴ってデータベースの設計構築も困難となってきた。これまでに、データベースの設計に関する多くの手法が提案されてきたが、概念設計における概念スキーマ作成を支援するものは少ない。筆者らは実体/関連モデル(ERモデル)^[1]に基づきデータベース設計の全段階を統合的に支援するデータベース設計支援システムDESSERT^[2]の構築を行っているが、そこではERモデルの作図・検証機能を提供するERエディタにより概念スキーマ作成を支援する。

本稿では、このDESSERTのERエディタについて報告する。

2. データベース概念設計における問題点

一般に概念設計では、データベースの利用者(または応用プログラム)が必要とするデータの概念的あるいは抽象的な構造(概念スキーマ)とそれに対する操作(概念操作)を設計する。概念設計においては、利用者の要求仕様をもとにどれが実体でどれが関連でどれが属性であるといった概念スキーマを決定することは困難なことがある。このような決定は熟練した設計者により試行錯誤的に生成される場合が多い。これまでに、概念スキーマの生成を自動化しようという試みがいくつかなされてきた^[3]が、その概念自体を理解して自動生成を行うのではないため、実用化には問題が多い。そこで筆者らは、概念スキーマを自動生成するのではなく、設計者の試行錯誤を支援することにより、概念スキーマ設計を支援することとし、その試行錯誤支援を実現するERエディタの検討・実装を行った。

3. ERエディタ

3.1 目的

本ERエディタは、データベース概念設計において概念スキーマ決定のために設計者が行う試行錯誤的設計を支援する。すなわち、

- 1) 概念モデルによる概念スキーマの作図の支援を行う。概念モデルとしてERモデルを使用、ERダイアグラムを対話的に作成、変更、ドキュメント化するためのエディタ機能を提供する。
- 2) 作成した概念スキーマ自身の一貫性検証(例えば各実体/関連の名前の一意性など)や概念操作と概念スキーマとの一貫性の検証(例えば、概念操作中の実体や関連が正しく存在している等)により、概念スキーマ設計段階での誤りの検出を行う。

3.2 DESSERTにおける位置付け

DESSERTはデータベースの概念設計・論理設計を統合的に支援する。ERエディタは概念設計の段階で、個々の設計者による局所概念スキーマの生成に使用される。

3.3 ERエディタの提供する機能

ERエディタは以下の方針で機能設計を行った。

- 1) 概念モデルとしてERモデルを使用する。但し、将来ERモデルにより汎化/専化、集約(aggregation)などの高位概念を付加したEER(拡張ER)モデル^[4]、またはその拡張版を概念モデルとして使用することを考慮し、拡張性のある実装を行う。図1にERエディタで今回用いたERダイアグラムを示す。
- 2) 入力装置としてキーボードとマウスを想定し、アイコン操作による良好な利用者インタフェースを提供する。
- 3) 外部記憶装置との入出力は、DESSERTで作成される他の支援ツールとの接続性を考慮する。
- 4) 概念スキーマに関する検証を行う。
- 5) 概念スキーマと概念操作との一貫性検証は今回は実装を行わない。

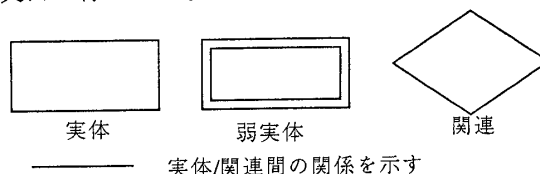


図1 ERダイアグラム

これらの方針に基づき以下の機能を提供する。

- a) アイコン操作
 - ・各オブジェクト(実体/弱実体/関連)の生成/展開/削除/移動/複写
 - ・実体/関連間の関係付け/削除
 - ・各オブジェクトの展開により、名前、コメント、属性、キー情報等各オブジェクトに必要な情報を設定可能
 - ・属性を展開することにより、属性の名前、型、コメント等各属性に関する情報が設定可能
- b) 仮想画面
 - ・実画面にとらわれない広大な仮想画面とその画面内の移動の機能
- c) ファイル入出力
 - ・外部記憶との間で設計データの入出力機能
- e) ハードコピー機能

・作成した概念スキーマのハードコピーを作成

d) 概念スキーマに関する検証機能

・各オブジェクトに設定された情報を用いて概念スキーマ構造の一貫性検証を行う。検証は、①各実体/弱実体/関連の名前が一貫であること、②各実体のキー定義がなされていること、③各関係セットのカーディナリティが定義されていること、④孤立した実体/弱実体/関連が存在しないこと、等について行う。

3.4 実装

PC9800シリーズ(OS:MSDOS)とその上の汎用ウィンドウシステム:MS-WINDOWを用いてERエディタの実装を行った。移植性を考慮し、システムの記述にはC言語を使用した。ERエディタが提供するサービスを図2に示す。

3.3節で述べた機能のうち、ハードコピー機能についてはMS-WINDOWの画面コピー機能を使用することとした。また、概念スキーマ検証機能のうち、アイコン操作時に検出できる一貫性違反(例えば、関係セットによる関連付けにキーとして用いられている属性を実体で削除しようとした等)については、操作のたびにチェックを行う。

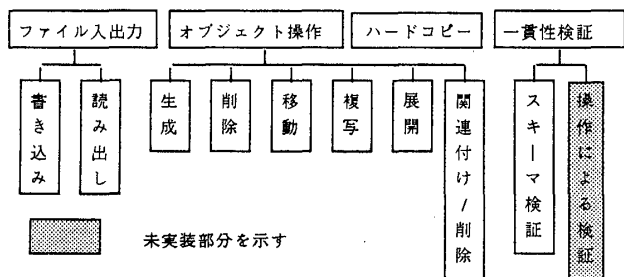


図2 サービス項目

3.5 評価

図3にERエディタを用いて概念モデルを設計した例を示す。若干画面の解像度の関係で多数のERダイアグラムを実画面で表示できないが、仮想画面機能により、実用規模(約500オブジェクト程度)の概念スキーマが編集できる。

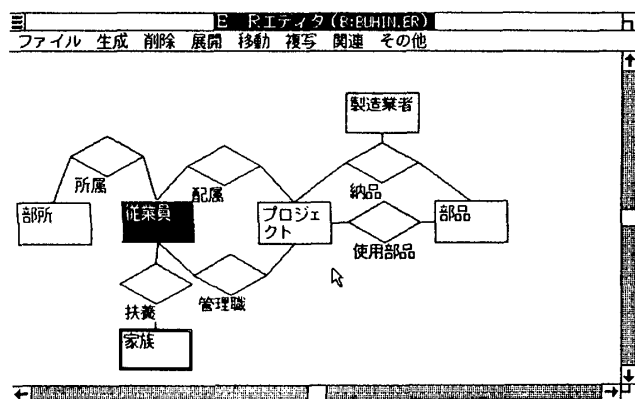


図3 作図例

4. 考察

1) 実装したERエディタにより概念スキーマの試行錯誤的設計の支援という目的は達成できた。複数の

の設計者がERエディタを使用することにより、ERエディタの出力を設計における標準記法として使用することができ、意志の疎通の向上が図れる。

2) 本実装では、表示能力等の不足により、例えば、実画面の解像度が悪いことや、図形移動時の図形軌跡の表示ができない等若干の点で利用者インタフェースに問題がある。また、複数の概念スキーマを画面上で編集することを考慮するとマルチウィンドウ機能が必要となる。これらの問題点を解決するためには汎用のウィンドウシステムを持つワークステーションへの移植が適切であろう。

3) 実際のデータベース設計においては、ERモデルで表される概念に加えて、汎化/専化、集約、導出データ、履歴データなどの概念が必要な場合が多い。実用的観点からはERエディタの概念モデルは、EERモデルにさらに導出データ、履歴データ等の概念を拡張したものが望ましい。

4) 概念スキーマの設計方法としては、ERエディタのように実体/関連等を決定し、それから属性やキー情報を設計していくというトップダウンの方法と、属性間の関数従属性や多値従属性から実体/関連等を決定し設計するボトムアップ的方法の2種類が考えられる。利用者からの要求仕様によっては両方の方法を用いたい場合もある。このような場合、例えば、ボトムアップ的方法を支援するツールの出力をERエディタに入力することで、設計者はこれらの方法を同一環境で使用できる。

5) 今回は、実装を行っていないが、概念スキーマと概念操作間の一貫性検証を行うためには、概念操作を一貫に記述する必要がある。この方法として、形式的な記述言語(例えばSQL的な記述言語)による入力と設計した概念スキーマを示したウィンドウ上で対話的に指定する方法の2種類が考えられる。但し、後者の場合もドキュメント化するために形式的な記述言語は必要になると考えられる。

5. おわりに

本稿では、実体/関連モデルに基づくデータベース設計支援システムDESSERTにおいて、試行錯誤による概念スキーマ設計を支援するERエディタについて述べた。今後は、ERエディタの拡張を進めるとともに、DESSERTの構築を進めていく予定である。

最後に日頃御指導頂くKDD上福岡研究所 小野所長、湯口次長、コンピュータ通信研究室 柳平室長、鈴木主任研究員に感謝します。

参考文献

- [1] Chen, P., "The entity-relationship model -Toward a unified view of data," ACM TODS 1, 1, Mar., 1976
- [2] 西山, 小花, "ERモデルに基づくデータベース設計支援システム DESSERTの構想", 本全国大会予稿
- [3] 川口, 溝口 他, "データベースの論理設計を支援する知的インテリジェンスシステム", 情報処理学会研究報告 AI-48, 1986
- [4] Elmasri, R., et al., "The category concept: An extension to the entity-relationship model," Data Knowl. Eng. 1, 1, 1985