

4L-1 データベース論理設計支援系SEEDの拡張

宮尾公崇 植村俊亮
 富士総合研究所 東京農工大学

0. はじめに

データベース論理設計支援システム『SEED』¹⁾²⁾の機能を拡張して、多様な論理設計を可能にしたので報告する。

『SEED』は、グラフィックディスプレイ上でグラフ構造に基づくスキーマを編集し、Coddの提唱した第3正規形を合成するシステムである。関数従属性(FD)のほかに、包含関係(ISA)を処理することができる。『SEED』の特徴は、「拡張可能なシステム」という点にある。これを拡張して第4正規形までの正規分解を可能とした。そのために不可欠な従属性として多値従属性(MVD)を追加した。『SEED』は、Armstrongの公理系を用いて関係を合成しているが、多値従属性に関しては、分解法を導入した。

1. データベース論理設計の位置付け

総合データベース環境における論理設計の位置付けを図1に示す。

システム構築	性能測定 性能向上 調整	データベース論理設計	データベース物理設計 生成	応用システムの開発、管理	問合せの支援
試作		試行		実行	
DBMS支援		データ設計支援		応用の支援	
総合データベース環境					

図1 データベース環境の基本概念

2. 拡張できるデータベース設計支援システム『SEED』

データベース論理設計支援システム『SEED』は、1988年に電子技術総合研究所で開発された。『SEED』の特徴は、利用者が自分に必要な設計機能を組み入れたり、既存の機能を組み合わせたりしながら、それぞれに固有の論理設計環境を構築できることにある。すなわち『SEED』は、利用者による拡張可能なシステムである。

『SEED』では、利用者が画面上でグラフィ的に入力編

集した情報(関数従属性および包含関係)をもとに、合成法に基づいた算法によって、正規形の関係スキーマを合成する。

『SEED』の動作環境は、次のとおりである。

- ・ワークステーション Sun-3
- ・作業データベースシステム University INGRES
- ・ウィンドウ環境 GMW
- ・開発言語 C言語および Eque1

3. 『SEED』の拡張

『SEED』では、W.W.Armstrongの公理系と、P.A.Bernsteinの合成算法により、第3正規形の合成を行なっている。今回は、これを以下の方針で拡張した。

- (1) 多値従属性(MVD)を追加する。
- (2) (1)による第4正規形への論理設計を可能にする。
- (3) (2)の算法には、分解法を用いる。

最初は、本来の『SEED』の合成法にならって、(3)を実現しようとしたが、次の二つの理由で、分解法に変更した。

- (1) 関数従属性と多値従属性とを含んだ公理系がまだ知られておらず、したがって合成算法も知られていない。
- (2) 『SEED』の合成法の部分には、誤りがあり、そのままでは拡張できないことがわかった。

多値従属性の入力では、システムの拡張性を実現するために、従属性がデータベース形式で管理されていることを利用した。拡張後のスキーマ設計・作成モジュールの構成を図2に示す。

拡張された『SEED』の出力結果の例を図3に示す。

4. 分解法の意味論的考察

『SEED』に、多値従属性を入力可能にし、分解法を実装して、第4正規形への論理設計が可能となった。

分解法では、まずすべての属性を一つの大きい関係とみなして、そこから関係を取り出しながら分解していく。最後に、他の関係ではキーとして扱っている属

On Extension of a Database Logical Design System SEED

Kimitaka Miyao, Syunsuke Uemura***

**Fuji Research Institute Corp., **Tokyo Univ. of Agriculture and Technology*

性（集合）と、どの関係にもかかわりのなかった属性とが残る。この残った部分が一つの関係を構成する。これが意味をもつのは、最初にすべての属性の間に関係が成り立つとした仮定がそれなりの意味をもつ場合である。合成法では、いわばばらばらの属性から合成を始めるので、分解法と異なる結果を産むことがあった。これを今回の拡張の結果確認した。

謝辞

『SEED』を作成された電子技術総合研究所の小島功氏に感謝の意を表します。

[参考文献]

- 1) データ論理設計支援システム外部仕様書，内部仕様書，富士通ソーシャルサイエンスラボラトリ（1988）
- 2) 小島功，植村俊亮：拡張できるデータベース設計支援システムについて，信学技報 DE88-16（1988）

図 2 拡張後のスキーマ設計・作成モジュール構成図

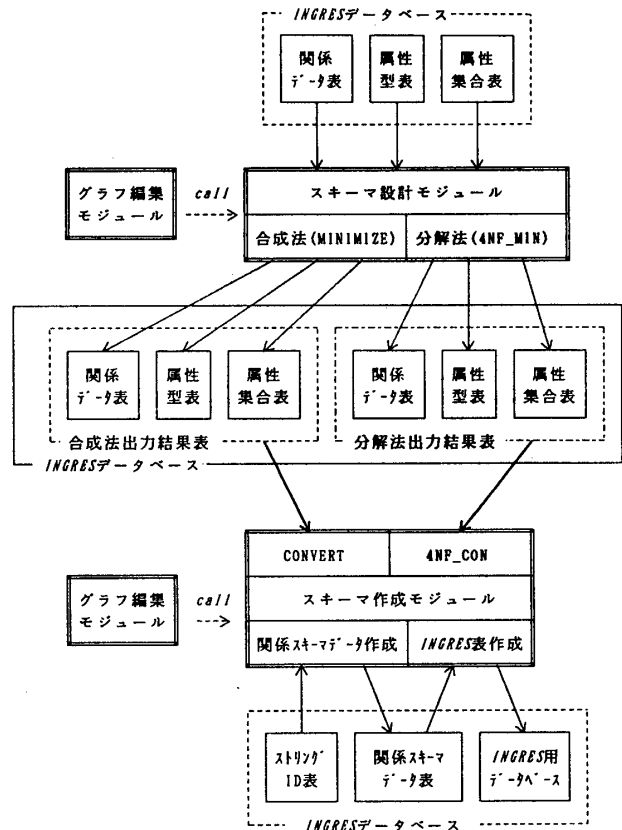


図 3 拡張された『SEED』の出力結果の例

```

junko:/dev/tty:
r5      part      y      c10
r5      project   y      c6
r6      project   y      c6
r6      employee  y      c20
-----
create r1(supplier=c20,part=c10,cost=i4)
r1 relation
|supplier      |part      |cost
|-----|-----|
create r2(project=c6,manager=c20)
r2 relation
|projec|manager
|-----|
create r3(employee=c20,salary=i4,hiredate=c6)
r3 relation
|employee      |salary      |hiredate
|-----|-----|
create r4(supplier=c20,location=c10)
r4 relation
|supplier      |location
|-----|
create r5(supplier=c20,part=c10,project=c6)
r5 relation
|supplier      |part      |projec
|-----|-----|
create r6(project=c6,employee=c20)
r6 relation
|projec|employee
|-----|
FSmainK <-- EV_CURSOR_LEAVE
FSmessageV <-- EV_MIDDLE_PRESSED
FSmainK <-- EV_CURSOR_LEAVE
            
```

Enter Database Name (12 length)
*mvd88

Set Editor Window Size.
Editor Mode Start.