

設計意図を表現可能なデータベース設計手法とその設計支援ツール

A Database Design Method expressing the Intention of Designer and its Authoring Tools

家富 誠敏[◆] 有澤 博[▲]

Masatoshi IETOMI Hiroshi ARISAWA

データベース設計者は、実世界の情報をモデル化した結果としてデータベーススキーマを設計するが、最終的に設計されたスキーマからは設計者がどのような考え方で実世界を捉えたか、どのような意図に従ってモデル化したかという情報を推測することは困難である。

それ故に、他人が設計したデータベーススキーマは時として理解し辛く、設計者以外による高度なデータベースの利用を阻害する要因となってしまう。

これを解決するために、本研究ではデータベース設計時に設計者の意図を蓄積し、これを再利用するデータベース設計手法を提案し、設計意図を視覚的に表示・編集できるインタフェースを実現する。

A database designer designs a database schema as a result which modeled the information on the real world. It is difficult to guess the intention of a designer from only the schema. How do the designer caught and modelize the real world?

So, the database schema which others designed is difficult to sometimes understand, and it is interfering with advanced using the database.

In order to solve this problems, in this research, we propose a method that the intention of a designer is accumulated at the time of a database design and realize the interface which can display and edit a design intention visually.

1. はじめに

データベース設計者は、実世界の情報をモデル化した結果としてデータベーススキーマを設計するが、最終的に設計されたスキーマからは設計者がどのような考え方で実世界を捉えたか、どのような意図に従ってモデル化したかという情報を推測することは困難である。

しかし、データベースを高度に利用するためには、表層的なスキーマの情報だけでなく、その設計思想を十分に把握した上で行なう必要がある作業も多く存在する。例えば、スキーマを変更する上では、現状の設計思想を十分理解した上で、変更内容を吟味するのは勿論のことであるし、アプリケーションを製作する上でも、データベースの思想を理解した上でそれを活かして製作することが望ましいだろう。

[◆] 正会員 城西国際大学経営情報学部

ietomi@jiu.ac.jp

[▲] 正会員 横浜国立大学大学院環境情報研究院

arisawa@ynu.ac.jp

これを実現するためには、設計者だけが知っているデータベーススキーマに関する情報、とりわけ、どのような意図や思想に基づきスキーマが設計されたのかという情報を設計者以外に伝える方法が必要であったが、説明文など言語を用いた伝達手法では、情報を記述する手間のわりに理解される情報の量は少なかった。

ところで近年、情報視覚化(IV)[1][2][3][4][5][6][7]の研究が盛んに行われるようになると、データベースの設計意図を視覚化して表示することによって、設計者以外の人間に情報を伝えるという可能性が開けてきた。

勿論、意図や思想といった目に見えないものを、そのまま視覚化することはできないが、設計時における設計者の試行錯誤のプロセス(特に、スキーマ設計思想に関わる重要な意思決定のプロセス)を視覚化できれば、設計者の思考の過程を追体験することによって、より高度な設計意図を汲み取ることが可能になるだろう。

例えば「設計途中のスキーマでは、ある特定のインスタンスが表現できない問題があった」「初期に考えていた表現法は、ある致命的な表現上の問題があって破棄した」といった情報および、それをどのようにして解決していったかが判れば、完成したスキーマからだけでは決して伺い知ることの出来ない設計段階での設計者の意図を知ることができ、これをスキーマの再設計などに役立てることができるようである。

しかし、この考えに基づき、設計時の意図として、設計者の試行錯誤のプロセスを記録し、視覚化するシステムの実現を試みる場合、試行錯誤のプロセス、とりわけ試行段階のスキーマにどのような問題点があり、それをどのように解決していったかを如何にして視覚化するかが大きな問題となる。

本研究では、設計時におけるスキーマ作成の履歴を保存した上で、重要と考えられる作成途中のスキーマのみを抜き出し、この各段階のスキーマに対する問題点を、設計者が実際にスキーマの検定に用いたインスタンスおよびそれに対する評価(どこに問題があったのか)をセットにして蓄積することにより表現する手法を提案する。これにより、設計者が実際に想像したインスタンスを考えたと、どこに問題があったのかを把握でき、次の試行段階のスキーマではそれがどのように変更されていったのかを、やはりインスタンスを通して把握することが可能になる。

本稿では、まず提案手法の概念について述べ、次に本提案手法に基づき作成した途中スキーマの履歴や検定用インスタンスおよびそれに対する評価を、スキーマ設計時に蓄積するための設計支援ツールについて述べる。

この設計支援ツールでは、設計者はスキーマを画面上で図的に編集するだけで、スキーマ履歴の保存等の操作を自動的に行なう。これにより設計者は通常の設計を行なった上で、ちょっとした操作を加えるだけで、自分の設計時の履歴と考え方の推移を蓄積することができ、これを必要に応じて再利用することが可能になる。

2. 関連研究

厳密には設計意図とは異なるが、データベーススキーマに関する情報をより詳細に表現するという観点では、これまでいくつかの研究が報告されている。

代表的なものとしては、オントロジー[8][9][10]やUML[11]等の概念を用いて、スキーマの構造定義だけでなく詳細なメタスキーマ情報および意味制約など、より高度なデータベーススキーマの記述を行なうというアプローチがあ

るが、それらは最終的に完成したスキーマに関する付加情報でしかなく、どのような試行のプロセスを経て、そのスキーマが設計されたかといったことを含めた高度な設計意図を汲み取ることは非常に困難である。

また、設計ツールとしては、スキーマを図的にデザインするツール[12][13][14]も提案されているが、これらは所詮、視覚的デザインを目的としたもので、それらの操作のログを記録しておきスキーマ編集の履歴を後から参照することは出来ても、なぜ、そのような履歴を辿ったのか、そこにはどういった設計者の思考・意図があったのかといったことは何い知れない。

本研究では、このような問題を解決するために、完成したスキーマに関する情報だけでなく、スキーマ設計時の履歴を大局的に記録するとともに、何故、そのような履歴を辿ったのかがわかるように、設計者の設計途中スキーマに対する見解(評価や問題点)を記録しておく手法を提案し、これを実現する設計支援ツールを構築する。

3. 設計プロセスを用いた設計意図の表現

3.1 提案手法の基本概念

本研究では、設計時における設計者の試行錯誤のプロセス(どのような過程を経て、それぞれの段階で何を考えていたか)を蓄積しておき、これを再現し、設計者の思考を追体験してもらうことで設計者以外の利用者にも設計者の意図を掴んでもらうという考えに基づき、設計時の試行錯誤のプロセスを蓄積する手法を考えていく。

その基本的な考え方は以下のようになる。

- ・スキーマの履歴を保存しておく
- ・重要な段階の途中スキーマに対し、
 - 検定に用いたインスタンスを記録しておく
 - 検定の結果を記述しておく

これによって、設計者以外の利用者が、スキーマの履歴を参照しながら、設計の重要な局面・段階において、設計者が作成中のスキーマに対してどのようなインスタンスを想定し、どのような評価を下したか、すなわち「このインスタンスが正常に表現できるからこのスキーマは良くできている」もしくは「こんなインスタンスが表現されてしまうスキーマは良くない」といった判断を行なった、という事実を知ることができる。

3.2 設計プロセスの定義

このような考えに基づき、本手法ではスキーマ設計時の設計プロセスを次のような要素の組みによって表現し、これを記録している。

設計プロセス = { S, D, I, J }

- S は試行時に作成したスキーマの集合
- D は S に属するスキーマ間の依存関係の集合
- I は各スキーマに対して検定時に想像したインスタンスの集合
- J は各インスタンスに対する設計者の判断(検定結果)

それぞれの要素の詳細を以下で順に説明する。

スキーマ集合 S

設計者が対象をモデル化し、その結果として作成されたスキーマ(設計途中段階のスキーマ)のうち特に設計上重要であると判断されるものを S_i と表記する。 S_i にはデータベースの構造定義の情報だけでなく、そのメタスキーマや満たすべ

き意味制約および基本オペレータセットの情報も含まれる。

この時、最終スキーマ S_n を作成するまでに、途中で作成した全てのスキーマ S_i を集めた集合 $\{S_1, S_2, S_3, \dots, S_n\}$ をスキーマ集合 S と定義する。

依存関係 D

これらのスキーマ集合 S の間には、 S_i を参考にして S_j を作成したという依存関係がある。

この依存関係をここでは $S_i \rightarrow S_j$ と表記するが、スキーマ集合 S に属する全てのスキーマに対する依存関係をもれなく集めたものを依存関係集合 D と定義する。

検定インスタンス集合 I

スキーマ集合 S の要素であるスキーマ S_i に対して、検定時に設計者が実際に考えてみたインスタンスを I_{ij} と表記する。

スキーマ S_i に対して検定に用いた全てのインスタンスの組 $\{I_{i1}, I_{i2}, \dots, I_{im}\}$ をスキーマ S_i に対する検定インスタンス組 I_i と呼ぶ。

またスキーマ集合 S に属する全てのスキーマに対する検定インスタンス組をもれなく集めたもの $\{I_1, I_2, \dots, I_n\} = \{ \{I_{11}, I_{12}, \dots\}, \{I_{21}, I_{22}, \dots\}, \dots, \{I_{n1}, I_{n2}, \dots\} \}$ を検定インスタンス集合 I と定義する。

検定結果集合 J

スキーマ S_i に対する検定インスタンス組 I_i に属する各インスタンスに対して、あるインスタンス I_{ij} を用いてスキーマ S_i を検定した結果を検定結果 J_{ij} と表記する。

検定結果 J_{ij} は各インスタンス I_{ij} に対して 1 つずつ存在し、そのインスタンス I_{ij} を考えてみた場合のスキーマ S_i に対する設計者の判断(そのインスタンスがスキーマで表現可能か否か? またそのインスタンスが表現できる or 出来ないことはスキーマとして問題がないか? 等)を表す。

検定インスタンス集合 I と同様に、あるスキーマに対する各インスタンスへの検定結果 I_{ij} の組をすべてのスキーマに対して集めたもの $\{ \{J_{11}, J_{12}, \dots\}, \{J_{21}, J_{22}, \dots\}, \dots, \{J_{n1}, J_{n2}, \dots\} \}$ を検定結果集合 J と定義する。

3.3 設計プロセスの表現例

このように定義される $\{S, D, I, J\}$ の組によって、設計プロセスを表現し、それによって設計者の意図を推測できるようにすることが本手法のねらいである。

実際のスキーマ設計を想定して、設計プロセスを表現した例が図1である。

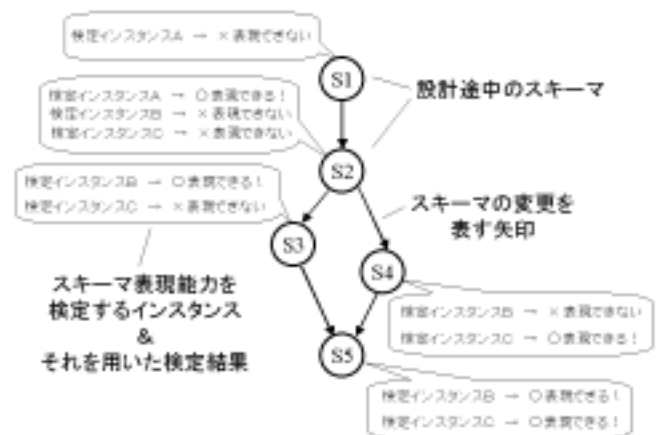


図1 設計プロセスの表現例

Fig.1 The example of expression of a design process

図1では、設計途中で作成された設計上の重要な途中スキーマSiをノードとして、その間のどのスキーマの作成にどのスキーマを参考にしたのかという依存関係Jijをノード間のアーク(有向)で図示している。このような情報を蓄積しておくことによって、設計者による設計スキーマの推移を知ることができる。

また、各スキーマを表すノードには、そのスキーマを設計した際に設計者が想像したインスタンスおよびそれに対する設計者の評価が付加されている。これを見ることで、設計者のそのスキーマに対する考えを推測することができる。

設計されたスキーマの推移と、各スキーマに対する設計者の考えを組み合わせると、どのような考えに基づいてスキーマを改良していったのかを推測することができる。この図の例では、スキーマS1にはインスタンスAが表現できない問題があり、それが次の段階ではスキーマをS2のように改良することによって解決している。しかし、このS2ではインスタンスBおよびCが表現できないという問題があり、これを解決するためにS2の改良版スキーマS3とスキーマS4を参考にして、S5のような形のスキーマを設計したことが示されている。

このように表現された設計プロセスによって、設計者が辿った試行錯誤のプロセスを追うことができ、設計意図を理解する上での大きな助けになると考えられる。

4. 設計支援インタフェースの実現

これまで述べてきた概念に基づき、設計時に設計者の意図(設計プロセス)を蓄積し、これを後から自由に参照することができる設計支援インタフェースを構築した。

このシステムでは、設計者はER図[15]を用いて視覚的に画面上に表示されたスキーマをマウス等を用いて編集することにより、スキーマを設計することができる。スキーマ設計時に作成途中のスキーマを要所要所で記録してもらうことによって、設計者が大局的にどのような試行を経て、最終的なスキーマにたどり着いたのかを保存する。

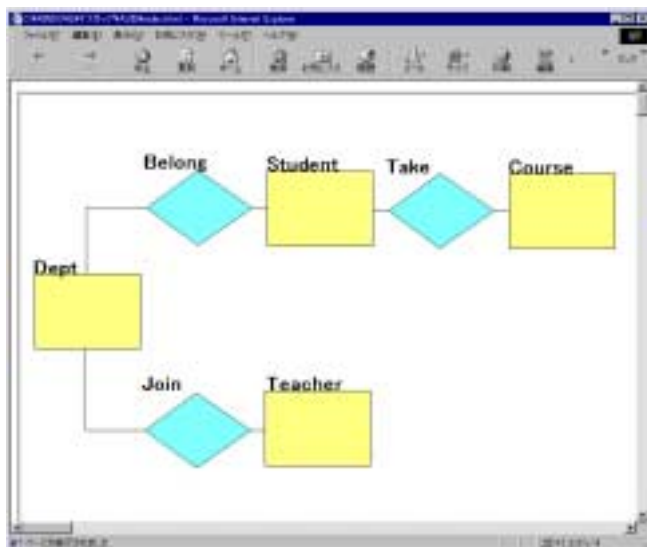


図2 ER図を用いたスキーマの表現

Fig.2 Expression of the schema by ER diagram

このとき、設計者は設計途中のスキーマを検定するために、

そのスキーマに対する検定用インスタンスを入力することができる。検定用インスタンスの入力は画面上のスキーマに重ねてインスタンス図を描くことによって行ない、この視覚的に表現されたインスタンス図を見ながら、そのインスタンスを用いたスキーマ検定結果を設計者のコメントやインスタンスの各部に対するカラーリング等を組み合わせて表現し、これを蓄積することができる。

- 例えば、カラーリングによって
 - ・インスタンスの一部に青く着色

現在のスキーマでは表現できないが、実際には表現したい内容であることを示す

- ・インスタンスの一部に赤く着色

現在のスキーマでは表現できてしまうが、実際には表現したくない内容であることを示す

というようにインスタンスの着目すべき部位を視覚的に表現し、そこに詳細なコメント(どうして問題があるのか等)を記述しておくことで、設計者が設計時にどのような考えをスキーマに抱いていたのかを蓄積することができる(図3)。

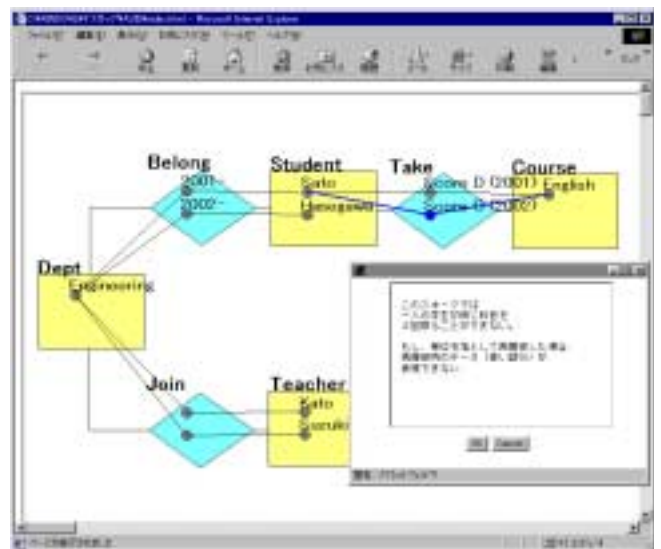


図3 検定インスタンスの表示

Fig.3 A display of instance to examine

このような設計プロセスを設計時に蓄積しておくことによって、設計者以外の利用者は後に以下のような操作を行うことができる。

- ・設計時における設計途中スキーマの推移を有向グラフによる表現や実際の設計操作を簡易アニメーションで表現したもによって捉えることができる

- ・スキーマの推移を見ながら、必要に応じて、各途中スキーマで、メタデータ等の詳細情報を表示したり、どのような検定が行われたのかを視覚化された検定インスタンスおよびそこに付加された検定結果から知ることができる。

これにより、設計者が設計時にどのような問題に直面し、その問題に対してどのような解決を図っていったかを推測することが可能になる。

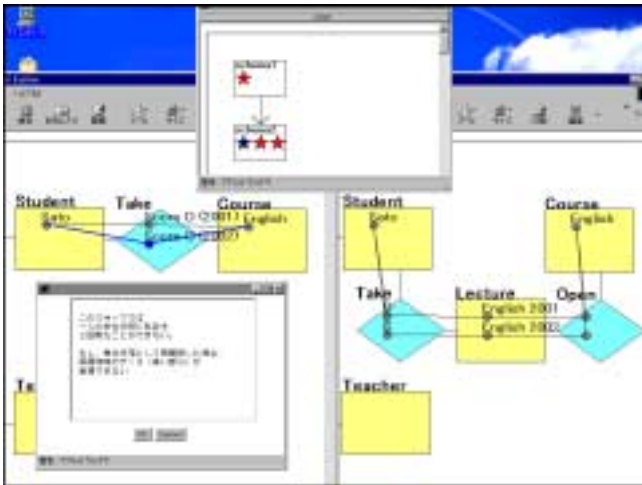


図4 設計プロセスの表示

Fig.4 display of a design process

図4は実際にデータベースを設計した際の設計プロセスを画面に表示した例である。画面上中央にあるのが、スキーマ履歴を表したグラフであり、ここでは設計者が schema1 を参考にして schema2 を作成していった場合の例であることを表している。このスキーマグラフ上で、スキーマを表す矩形や検定インスタンスを表す星型をマウスでクリックすると、そのスキーマやインスタンスが画面に図で表示される。この例では、2つの検定インスタンスを画面に表示させているが、画面左が schema1、画面右が schema2 の検定インスタンスである。ここでは schema1 に付加された検定インスタンスおよびコメントによって、設計者がどのような問題点に着目していたのかが示され、schema2 に付加された検定インスタンスによって、設計者がこの問題をどのように解決したのかが実例で示されている。さらに詳細を知りたい場合には、設計者の行なったスキーマ変更操作を細かくアニメーションで見えていくこともできるようになっている。

5. まとめと今後の課題

本研究では、設計者の設計意図を表現するために、設計プロセスを蓄積・視覚化する手法を提案し、そのためのインタフェースの構築を行なった。

現状では、設計プロセスを蓄積した上で、それを視覚化するだけのインタフェースだが、それ以外の再利用方法、スキーマ修正時に設計者のログを参照して、過去において問題があるとされている変更であった場合にそれを警告するといったような応用についても、今後は研究していきたい。

[文献]

- [1]Stuart K. Card : “Visualizing retrieved information: A Survey”, IEEE Computer Graphics and Applications, Vol. 16, No. 2, pp. 63-67, 1996
- [2]Daniel A. Keim, Hans-Peter Kriegel : “VisDB: Database Exploration Using Multidimensional Visualization”, IEEE Computer Graphics and Applications, Vol. 14, No. 5, pp. 40-49, 1994
- [3]佐野 綾一, 波多野 賢治, 田中 克己 : “自己組織化マップを用いたWeb文書の対話的分類とその視覚化”, 情報処理学会研究報告, Vol. 98, No. 57, 98-DBS-116(1)-5, pp.

33-40, 1998

[4]磯部 成二, 黒川 清, 塩原 寿子, 飯塚 哲也 : “視覚化による多次元データ分析システム : INFORVISER”, 情報処理学会論文誌, Vol. 40, No. 5, 1999

[5]武田 浩一, 野美山 浩 : “テキスト情報の可視化を利用した情報検索”, 情報処理, Vol. 41, No. 4, 2000

[6]古畑 理香, 藤代 一成, 市川 哲彦, 竹島 由里子 : “GADGET/IV: 情報可視化の半自動設計支援環境”, 第11回データ工学ワークショップ(DEWS 2000), 2000

[7]尾下 真樹, 牧之内 顕文 : “オブジェクト指向データベースの半自動可視化環境”, 電子情報通信学会技術研究報告, Vol. 100, No. 31, DE2000-5, pp. 33-40, 2000

[8]Bernhard Thalheim : “Component Construction of Database Schemes”, Conceptual Modeling - ER2002, Springer LNCS 2503, pp.20-34, 2002.

[9]Giancarlo Guizzardi, et al : “On the General Ontological Foundations of Conceptual Modeling”, Conceptual Modeling - ER2002, Springer LNCS 2503, pp.65-78, 2002.

[10]Farshad Hakimpour, Andreas Geppert : “Global Schema Generation Using Formal Ontologies”, Conceptual Modeling - ER2002, Springer LNCS 2503, pp.199-213, 2002.

[11]Sergio Lujan-Mora, et al : “Multidimensional Modeling with UML Package Diagrams”, Conceptual Modeling - ER2002, Springer LNCS 2503, pp.199-213, 2002.

[12]A.Grau, et al : “The TROLL Approach to Conceptual Modelling: Syntax, Semantics and Tools”, Proceedings of the International Conference on Conceptual Modelling, pp.277-290, 1998.

[13]Sudha Ram, et al : “DISTIL: A Design Support Environment for Conceptual Modeling of Spatio-temporal Requirements”, Conceptual Modeling - ER2001, Springer LNCS 2224, pp.70-83, 2001.

[14]Virginie Detienne, Jean-Luc Hainaut : “CASE Tool Support for Temporal Database Design”, Conceptual Modeling - ER2001, Springer LNCS 2224, pp.208-224, 2001.

[15]P. P. Chen : “The Entity-Relationship Model: Toward a Unified View of Data”, ACM Transactions on Database Systems, Vol.1, No.1, 1976.

家富 誠敏 Masatoshi IETOMI

平成 13 年横浜国立大学大学院工学研究科博士課程修了, 工学博士。同大学エコテクノロジーシステムラボラトリ講師(中核的研究機関研究員)を経て、現在は城西国際大学経営情報学部講師として活動中。次世代 DBMS のアーキテクチャおよびデータベースの視覚的インタフェースを研究テーマとしている。日本データベース学会正会員。

有澤 博 Hiroshi ARISAWA

昭和 48 年東京大学理学部物理学科卒業。富士通(株)を経て、昭和 50 年横浜国立大学工学部に奉職。現在同大学院環境情報研究院「社会環境と情報部門」教授。工学博士。1991 年アメリカ・オレゴン州立大学計算機学科客員教授。データベース理論, マルチメディアデータベースシステムを研究テーマとしている。日本データベース学会正会員。