

演繹データベースへの階層概念の導入

高橋千恵

(財)日本情報処理開発協会

横田一正

(財)新世代コンピュータ技術開発機構

演繹データベース (DDB) にオブジェクト指向概念を組み込む演繹・オブジェクト指向データベース (DOOD) が注目されている。ここでは、オブジェクト指向概念の計算モデル的側面をDDBに組み込むことを提案する。それはDDBの問合せ処理をオブジェクト間のメッセージ・パッシングによって実行する。このために、DDBを一つのオブジェクトに対応させ、継承関係を持ったそれらの階層関係を考える。これが階層型DDBである。我々は最初、項表現を拡張した言語CRLで記述されたDDBに階層構造を導入した。まず、その実現方法を述べる。この経験を踏まえ、設計中のDOOD言語Juanに階層概念を言語の基本機能として導入した。

A Deductive Database with Hierarchical Structure

Chie Takahashi

Japan Information Processing
Development Center(JIPDEC)
Kikaishinkou-Kaikan, 3-5-8,
Shiba-Kouen, Minato-ku, Tokyo
105, JAPAN
e-mail: takahash@icot.or.jp

Kazumasa Yokota

Institute for New Generation
Computer Technology (ICOT)
21F., Mita-Kokusai Bldg., 1-4-28,
Mita, Minato-ku, Tokyo 108,
JAPAN
e-mail: kyokota@icot.or.jp

Deductive databases (DDBs) with object-oriented concepts have been noticed. We introduce DDBs in which computational aspects of object-oriented concepts are embedded. In the DDB, a query is processed by message-passings between objects. For this idea, we take a DDB as an object, and embed a set of objects in a hierarchical structure with inheritance relation. This is called a deductive database with hierarchical structure. In this paper, first, we introduce a hierarchical structure into DDBs written with CRL. We explain a way of the implementation. Next, we introduce this concept into a DOOD language Juan as one of basic features.

1 概要

演繹データベース (DDB) のさまざまな拡張が提案されているが、その中でも演繹データベースにオブジェクト指向概念を組み込む演繹・オブジェクト指向データベース (DOOD) が注目されている^[1]。ここでは、オブジェクト指向概念の計算モデル的側面をDDBに組み込むことを提案する。つまり、DDBの問合せ処理をオブジェクト間のメッセージ・パッシングによって実行することである。そのために、DDBを一つのオブジェクトに対応させ、継承関係を持ったそれらの階層関係を考える。これが階層型DDB^[2]である。

実世界の知識を見ると、知識はバラバラなものではなく、それらはある単位で分類、モジュール化されている。たとえば、数学の知識を考えると、それらは自然数論、集合論、など理論という単位でモジュール化されており、それらが (もちろん立場によって異なっているが) 互いに継承関係を持っている。そして、そのモジュール化された知識が互いに関係しながら、問題解決や質問処理にあたる。DDBは、DOODの枠組の中で知識表現としても大幅に拡張されており、そのような知識モジュールの表現として有力であると考えられる。したがって、複数のDDB (オブジェクト) の集合に階層 (厳密には有向非巡回グラフ) 概念を導入した階層型DDBは、知識モジュールのモデルとして自然である。各DDBは、オブジェクト指向での能動的オブジェクトに対応しており、自律的な問合せ処理能力を持っていると考える。したがって問合せに対しては、各DDBが階層関係に対応して動的な継承を行いながら、メッセージ・パッシングにより協調して処理を行う。この応用としては、数学や分子生物情報などの大規模知識ベースのモジュール化、CAIでの各人の学習の発展過程の管理、あるいは法的推論のような専門家システムでの仮説推論などを考えている。

このようなDDBの拡張のために、我々は最初、項表現を拡張した言語CRL^[3]で記述されたDDBに階層構造を導入し、そのシステムを試作した^[4]。その実現方法を2節に述べる。その経験を踏まえ、我々は現在設計中のDOOD言語Juan^[5]に、そのモジュール概念を言語の基本機能として導入している。その概要を3節で説明する。最後に4節では簡単に今後の課題/関連研究に言及する。

2 CRLの経験

多値関係に対応した論理型言語CRLに基づくDDBに、階層概念を導入した。

2.1 階層概念の導入

定義は以下のように与えられる。

1. DDB

否定を含まない、単調なCRL項からなるDDB。

2. 世界

識別子を持つDDB。

3. 宇宙

互いに同一の識別子を持たない世界の集合で、反射律、反対称律、推移律を満たす半順序関係を持つもの。

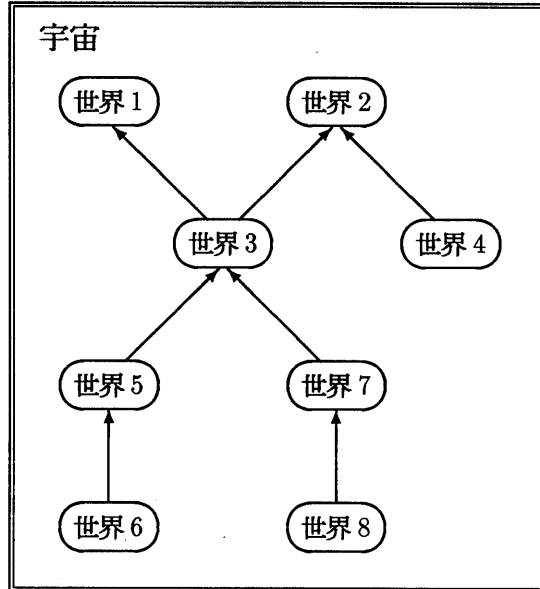


図 1: 世界と宇宙の例

宇宙と世界の意味は、単一のDDBとの対応で定義する。

世界の意味は、宇宙における半順序関係を外延階層とし、上位の世界の全てのルールと事実を継承したDDBに対応していると考ええる。

宇宙の意味は、各世界のDDBの和集合と考え、その宣言的意味論とする。この時、各世界のルールと事実は世界の識別子まで考え、互いに排他的と考えている。

以上のように考えると、半順序関係にない別の世界同士には、相反したルールや事実が干渉し合うことなく存在でき、宇宙にはさまざまな異種の「知識」が共存することができる。これが階層構造を持ったDDBである。

2.2 質問

宇宙に対し、質問を行う。それは宇宙の動的な再構成機能を持ち、宇宙に含まれている複数の世界で質問処理を行う。

質問は以下の3つからなる。

- 質問対象とする世界の識別子の集合、あるいは変数
- ゴール（述語の集合）
- 宇宙に対する更新情報（世界ごとの半順序と規則の追加情報）

質問によって与えられる「更新情報」は、データベースに対する更新とは異なり、通常のDDBと同様に問合せの拡張として位置付けている。つまり、個々の質問ごとにその対象となる宇宙を動的に再構成していると考ええる。また、ゴールは質問対象とする世界の識別子を付け、各世界で問いを求めるように展開される。

上記の質問の定義によって可能となる質問の種類は以下の通りである。

- 特定の世界に対する質問
- 世界を求める質問
- 継承関係追加した時の質問
- 世界に知識を追加した時の質問

2.3 分散問合せ処理

質問が与えられると、宇宙と各世界がそれぞれ独立したプロセスとなり、プロセス間通信によって協調的に処理される。質問処理の概要は以下の通りである。

1. 問合せに対し、必要なルールを各世界で抽出。
2. 抽出されたルールに問合せの束縛情報を sideways information passing [6] で伝播。
3. 束縛情報が伝播されたルールを問合せ処理の効率化のために一般マジック集合法 [7] で変換。
4. 変換されたルールの述語 (CRL項) 間の依存グラフを生成し、半順序関係を持つ評価単位を生成。
5. 最小評価単位から順序関係に従って、世界間で協調しながら順次評価単位をセミナイーブ評価法 [8] で評価。

これらについて、以下に詳しく述べる。

2.3.1 ルールの抽出

宇宙に質問が与えられると、宇宙プロセスが生成され、質問の「更新情報」を使って宇宙を動的に再構成する。次に新しい宇宙の中で極大の知識を持っている、継承関係で極下位の世界が自動的に選択される。そして以下に述べる質問処理が、選択された世界に対し順次行われる。

1. 宇宙プロセスは、選択された極下位の世界と継承関係で上位の世界のプロセスを生成する。
2. 各世界プロセスは、自分が持っている演繹データベースを調べ、その情報を宇宙プロセスに知らせる。
3. 宇宙プロセスは、各世界プロセスから送られてくる情報を整理し、これを使って以下の質問処理を行う。
4. 宇宙プロセスは、ゴールを評価できる世界プロセスに投げる。
5. ゴールを投げられた世界プロセスは、それを評価できるルールを調べサブゴールを求め、宇宙プロセスに返す。
6. 宇宙プロセスは、戻ってきたサブゴールを評価できる世界プロセスを求め、サブゴールを該当する世界プロセスに投げる。このことをサブゴールがなくなるまで繰り返す。

7. 世界プロセスは、宇宙プロセスとのゴールの遣り取りの間に、質問を評価するのに必要なルールを自分の I D B から抽出する。

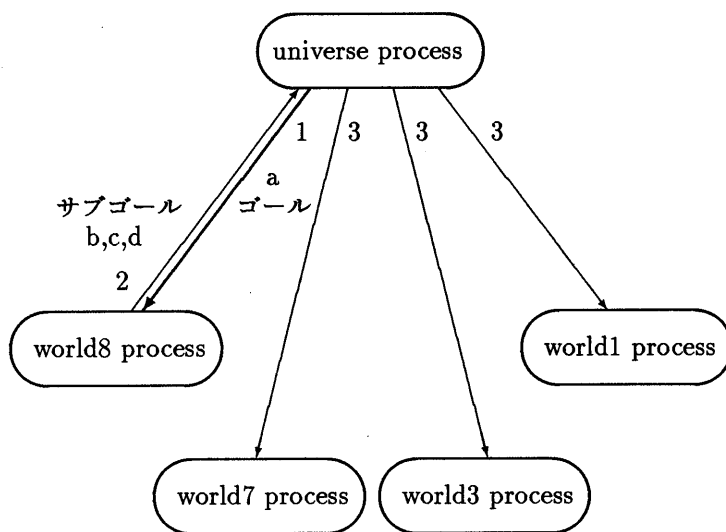


図 2: 宇宙プロセスと世界プロセスとのゴールの遣り取り

2.3.2 束縛情報の伝播

次にマジック集合法でルールの変換を行うために、質問の束縛情報を各世界プロセスに伝播する。ここでは、非正規関係に対応した知識表現言語 C R L を用いているので、束縛情報は束縛されていない変数の属性名のパスを用いている。

伝播方法は上記のゴールの遣り取りと同じように、ゴールと束縛情報を一緒にして宇宙プロセスと各世界プロセスが遣り取りを行う。

2.3.3 一般マジック集合法での変換

各世界プロセスで、束縛情報付きルールを一般マジック集合法で変換し、マジックルールの情報を宇宙プロセスに送る。シードは、ゴールを評価できるルールを持つ世界プロセスのうち一つの世界プロセスが生成する。

2.3.4 依存グラフの生成

宇宙プロセスは各世界プロセスから、ルールの中の述語間の関係を示す情報を得る。それを使って、半順序関係を持つ評価単位を生成する。

2.3.5 評価

宇宙プロセスは依存グラフの最小評価単位を取り出し、それに含まれるルールを持つ世界プロセスに評価するように指示する。その時、そのルールの述語を使うルールも依存グラフで調べ、結果のテーブルをどこの世界プロセスに投げるかをも指示する。

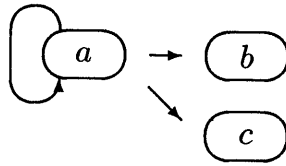


図 3: ルール a : - a, b, c の述語間の関係

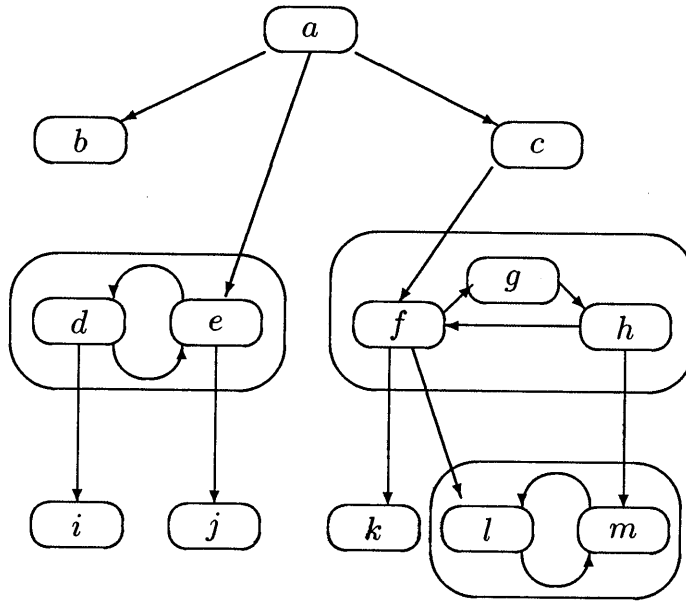


図 4: 半順序を持つ評価単位

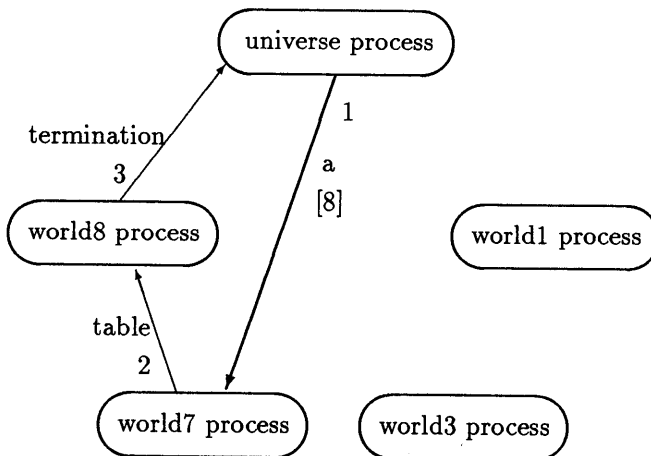


図 5: 評価

2.4 補助情報と下降評価

CRL演繹データベースで現実のさまざまな問題を扱おうとするとき、いくつかの機能の拡張が必要である。たとえば、都市間のアークからなるグラフを考えると、ある都市から別の都市に行く経路情報の他、その料金、時間などの集約情報を表現したい。また、経路はループする可能性があるのでその停止情報も組み込みたい。現在のCRLでは、トップの属性名にHiLog風のパラメータを持つことによって、今までの構文をきたなくすることなく、リスト処理や算術演算などに基づく集約関数を実現した。

$$\begin{aligned} path(route/[X,Y],fare/M,time/S)/[fr/X,to/Y] : -arc/[fr/X,to/Y,fare/M,time/S]. \\ path(route/[X|L],fare/M+N,time/S+T)/[fr/X,to/Y] : - \\ \quad arc/[fr/X,to/Z,fare/M,time/S], \\ \quad path(route/[L],fare/N,time/T)/[fr/Z,to/Y]. \end{aligned}$$

現在は、上昇評価で全解を求めた後、個別に上記のルールの評価を下降評価している。

3 Juan のモジュール化機能

前節で述べたCRL階層型DDBの階層概念は、いわゆる概念の外付けとして導入されたものであった。その試作の経験から、DDBのモジュール化機能の重要性を一層確信した。そこで、現在設計中のDOD言語Juan^[5]では、基本概念としてその機能を取り込んでいる。本節では、モジュール化に焦点を絞り、その概要を説明する。

3.1 構文と言語構造

Juanの基本構造は以下のような6種類のレベルからなっている。

実体	定義
世界	= モジュール化データベースのモジュール
データベース	= ルールの集合
ルール	= (属性項, { 属性項, ..., 属性項 })
属性項	= (オブジェクト項, { 属性, ..., 属性 })
オブジェクト項	= (基本オブジェクト, 識別属性) (オブジェクト変数, 識別属性)
基本オブジェクト,	オブジェクト変数

更に、データベースから世界を構成するために、以下の構文を持っている。

$$\begin{aligned} \langle \text{データベース} \rangle & ::= \{ \langle \text{世界ルール} \rangle \text{ リスト} \} \\ \langle \text{世界ルール} \rangle & ::= [\langle \text{世界ラベル} \rangle \text{ “} \Rightarrow \text{”}] \langle \text{ルール} \rangle \\ & \quad | \langle \text{世界関係} \rangle \\ \langle \text{世界ラベル} \rangle & ::= \{ (\langle \text{世界識別子} \rangle [\text{“} ; \text{”} \langle \text{ルール識別子} \rangle]) \} \\ & ::= | \langle \text{世界ラベル} \rangle \text{ “} / \text{”} \langle \text{世界ラベル} \rangle \\ \langle \text{ルール} \rangle & ::= \langle \text{属性項} \rangle [\text{“} \Leftarrow \text{”} \langle \text{属性項} \rangle \text{ リスト}] \\ \langle \text{世界関係} \rangle & ::= \langle \text{世界識別子} \rangle \text{ “} : \text{”} \{ \langle \text{世界識別子} \rangle \text{ リスト} \} \end{aligned}$$

但し、‘<X>リスト’とは、‘<X> {“,” <X>}’のことである。

各ルールには世界ラベルが付加されているが、これは syntax sugar として以下のように書くことができる。

$$\begin{aligned} w_1 &: \{r_{11} : \{\phi_{111}, \dots\}, \dots, \phi_{1n}, \dots\}, \\ w_2 &: \{\dots\}, \\ &\dots \end{aligned}$$

ここで w_i, r_{jk}, ϕ_{lmn} はそれぞれ世界識別子、ルール識別子、ルールを表している。ルール識別子は世界内で一意である必要はなく、ルールのグループを指すことができる。また、ルール識別子を持たないルールを書くことも可能で、この場合は、継承において例外の対象になりえないだけである。

3.2 継承機能と例外の扱い

3.1 に従って、継承機能と例外について簡単に説明する。

世界識別子を w, w_1, \dots, w_n とし、世界 w と他の世界 w_1, \dots, w_n の関係を以下のように指定する。

$$w : \{w_1, \dots, w_n\}$$

この意味は、

$$w \preceq w_1 \wedge \dots \wedge w \preceq w_n$$

であり、世界 w は上位世界として w_1, \dots, w_n を持ち、ルールをそれら上位世界から継承することを表している。複数の世界間の関係は、上記の各々の世界と他世界との関係に従って半順序関係に構成される。

一方、ルール識別子を持った世界ラベルは、継承の際の例外機能や他世界からの明示的参照のために利用される。 w_i, r_j, ϕ_k をそれぞれ世界識別子、ルール識別子、ルールとすると

$$\begin{aligned} w_2 &: \{r_2 : \{\phi_1, \phi_2, \dots\}, \dots\} \\ w_1 &: \{r_1 / (w_2 / r_2) : \{\phi_{1'}, \phi_{2'}, \dots\}, \dots\} \\ w_1 &: \{w_2, \dots\} \end{aligned}$$

は、 w_1 は上位世界 w_2 の r_2 で識別されるルール集合 $\{\phi_1, \phi_2, \dots\}$ を継承せず、代わりに $\{\phi_{1'}, \phi_{2'}, \dots\}$ を使用することを意味している。この意味で $(w_1, r_1) / (w_2, r_2)$ を例外ラベル、 (w_2, r_2) を書き換えラベルと呼ぶ。

3.3 意味論と問合せ処理

継承関係を含む世界の宣言の意味論は、基本的には CRL と同様に与える。

1. 世界構造を半順序とする。
2. 継承関係で飽和した状態を考える。この際例外も考慮される。
3. 各世界のルールは世界識別子を識別属性としてもつ。

これによって、継承関係にない世界間では、‘矛盾’した知識を互いに影響し合うことなく持つことができる。

問合せ処理としては、上記(2.)の結果を対象にするのではなく、動的継承を行いながらの意味論を検討中である。同時に、マジック集合法等の問合せ最適化手法がどこまで適用可能かも検討している。

4 おわりに

Juan は、演繹・オブジェクト指向データベースのための知識表現言語を目指して現在設計中である。そのために、分子生物情報データベースや法的推論の事例ベースの記述実験や、CRLでの階層型演繹データベースの経験を通して、言語の必要要件を明確にしつつある。本稿では、そのモジュール化機能の枠組について報告したが、今後検討すべき点が多く残されており、さらに効率的実装方式を含め検討を続ける予定である。

謝辞 最後に本研究に関し多くの示唆を下さったDOO-I会議、ICOTの「演繹+オブジェクト指向データベース・ワーキンググループ」(DOO-WG)のメンバの方々に深く感謝致します。

参考文献

- [1] 横田一正, 西尾章治郎, “演繹・オブジェクト指向データベース”, 情報処理, Vol.31, No.2, pp.234~243(1990)
- [2] 横田一正, “階層構造をもった演繹データベース”, ICOT 内部メモ, 1989
- [3] K.Yokota, “Deductive Approach for Nested Relations”, in *Programming of Future Generation Computers II*, K.Fuchi and L.Kott, eds., North-Holland, 1988
- [4] 高橋, 横田, 横塚, 梶山, “拡張項と階層構造に基づいた演繹データベースの試作”, 情報処理学会全国大会, 2C-7, 福岡, 10月(1989)
- [5] K.Yokota, “The Outline of a Deductive and Object-Oriented Database Language *Juan*”, 情報処理学会第78回データベース・システム研究会, 札幌, 7月(1990)
- [6] J.D.Ullman, “Implementation of Logic Query Languages for Databases”, *ACM TODS*, vol.10, no.3, 1985
- [7] C.Beer and R.Ramakrishnan, “On the Power of Magic”, *ACM PODS*, 1987
- [8] F.Bancilhon, “Naive Evaluation of Recursively Defined Relations”, in *On Knowledge Base Management Systems*, M.L.Brodie, et al, eds., Springer, 1986