

“層”の概念を導入したデータベースの教育環境への応用

2Y-5

大矢野 潤† 柏木 将宏‡

千葉短期大学

1 はじめに

近年、INTERNETの利用形態が多様化したことにもない、ユーザがあるサービスを利用する状況と、そのサービスの仮定する前提条件との間にギャップが生じてきている[1]。例えば、教室にPCを持ち込み授業のノートを取り、そのノートを使用して自宅でレポートを書く場合に、そのPCは教室と自宅で異なるネットワークに接続される。この時、IPアドレス、デフォルトルータ、その他各種プロキシサーバのアドレスなどはDHCP(Dynamic Host Configuration Protocol)などを利用することにより自動的に対応することができる。しかし、利用するサービスが電子掲示板、出席確認、レポート提出などと、さらに複雑なものになってきた場合には、サーバがクライアントに対するサービスに変化をつけるための条件は、そのPCが接続されているネットワークアドレスや、学生か教員かなどのユーザの分類などの単純なものでは十分ではない。しかし、これらの場合分けと、その組み合わせに対応してサービスの種類の変化を手続き的に記述していく作業は繁雑であり、誤りを混入させやすい。

本論文では、サービスをコースの集合としてとらえ、その集合から“層”を用いてコースを切り出すことを試みる。そして、層の構成を変更することによりサービスにバリエーションをつけることができることを示す。

2 コースの記述

本論文では、サービスを記述する場合に“コース”の概念を用いる。そして、ユーザはコースの始点から終点までたどっていくことによりサービスを受けることができる。コースはラベル付き遷移システム T として次のように定義する。

$$T = (S, L, \{\overset{l}{\rightarrow} : l \in L\})$$

ここで、 T は状態の集合 S 、ラベルの集合 L 、そして $l \in L$ と $S \times S$ 上の遷移関係 $\overset{l}{\rightarrow} \subseteq S \times S$ によって構成される。また、ここで、各々のコースの記述方法はCCS(Calculus of Concurrent System)[4]のプロセスの記述から借りてくるものとする。例えば、コースのある状態 C で a というアクションを選択をした場合に C' に遷移することを $C \stackrel{\text{def}}{=} a.C'$ と書く。同様にコースの選択(プロ

セスの選択)を $C_i + C_j$ 、コースの合成(プロセスの合成)を $C_i | C_j$ と書くことにする。また、それぞれ略記として、 $C_{i_0} + \dots + C_{i_n}, (0 \leq j \leq n, i_j \in I)$ を $\Sigma_{i \in I} C_i$ 、 $C_{i_0} | \dots | C_{i_n}, (0 \leq j \leq n, i_j \in I)$ を $\Pi_{i \in I} C_i$ と書くことにする。ここで、特別に $\Sigma_{i \in E} E_i \stackrel{\text{def}}{=} 0$ とし、これ以上遷移のできないプロセス、つまり、コースの終了を意味する。更に、 C に対してラベルの制限関数 $C \setminus L$ とはラベルの集合 L に出現するラベルを使っての外部のプロセスとの通信の禁止、ラベル変更関数 f の C への適応 $C[f]$ はコース C に出現する選択肢の名前の変更を意味する。

次の例を考えてみよう。まず、下のよう定義されるコース $C \equiv C_0$ があるものとする。

$$\begin{aligned} C \equiv C_0 &\stackrel{\text{def}}{=} a.C_1 + b.C_2 + c.C_3 \\ C_1 &\stackrel{\text{def}}{=} d.C_4 \\ C_2 &\stackrel{\text{def}}{=} e.C_4 + f.C_5 \\ C_3 &\stackrel{\text{def}}{=} g.C_5 \end{aligned}$$

このプロセス C にラベル制限関数 $\setminus L, L = \{c, f, g\}$ を適応し、 0 に関するルール $C + 0 = C$ を考慮するとコース C' を選択することができる(図1参照)。

$$\begin{aligned} C' \equiv C'_0 &\stackrel{\text{def}}{=} a.C'_1 + b.C'_2 \\ C'_1 &\stackrel{\text{def}}{=} d.C'_4 \\ C'_2 &\stackrel{\text{def}}{=} e.C'_4 \end{aligned}$$

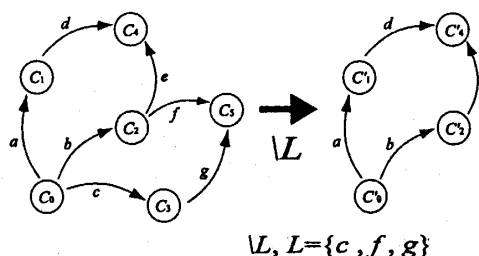


図1: コースの選択

このことは、ラベル制限関数は、図1のように、一般的なコース C から特別なコース C' を切り出すことができることを示している。

Database with “Layer” Concept for Educational Environment

† Jun Oyano, ‡ Masahiro Kashiwagi

Chiba Junior College

1-3-1 Konodai, Ichikawa, Chiba 272-8512, Japan

つぎに、 $C(y) \stackrel{\text{def}}{=} in(x).\overline{out}(\langle x : y \rangle).C(y)$ とする。また、 $C_n \frown C_m$ を $(C_n | C_m)[f] \setminus L, f = \{\overline{in}_m / \overline{out}_n\}, L = \{\overline{in}_m, \overline{in}_m\}$ とする。ここで、 $\langle \langle \dots \langle x_0 \rangle : x_1 \rangle \dots \rangle : x_n = \langle x_0 : \dots : x_n \rangle$ とすると、 $C_i(x) \frown C_j(y)$ は x と y によって構成されるタプル $\langle x : y \rangle$ をもつセルであるということが出来る。ここでタプル $\langle t_0 : \dots : t_i : \dots : t_k : \dots : t_n \rangle$ をもつセルに対してラベル変更関数 $[f], f = \{\overline{in}_k / \overline{in}_{i+1}\}$ を適応すると、 $\langle t_0 : \dots : t_i : t_k : \dots : t_n \rangle$ というタプルを生成することができる。

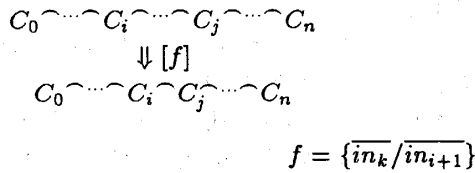


図 2: コースの射影

本論文では、このラベル制限関数 $\setminus L$ と、ラベル変更関数 $[f]$ を組み合わせたものを“層”とよび、この層を利用して一般的なコースからそれぞれの層が表す条件によってコースのサブセットを切り出すことに使用する。

3 データベースとの対応

教育環境において、成績管理、時間割管理などはデータベースシステムを利用して行なわれる。しかし、すべてのデータを単一のデータベースに格納し、データの一元化をはかるのは現実的に困難である。これはセキュリティ上の問題や、データベースの管理母体の違い、データ内に存在するデータの質などが異なることによる。また、実際にデータベースを利用する形態が、ネットワーク上のWWW(World Wide Web)などを利用しての検索が多く利用されるため、例えば *Select * from A, B* というSQL文に埋め込まれるデータベース A と B のそれぞれの物理的な格納位置、そして問い合わせを行なったクライアントが同一システム内に存在するとは限らない。このため、従来の関係データモデルなどよりも粒度の細かいデータモデルが必要である。

前節の議論で、コースの状態に値を持てるようにしたCCSを応用すると、コースの選択、タプルの組みかえができることがわかった。これは関係データベースにおける選択関数 σ と射影関数 π を表現していると言い替えることができる。このことはあるコースにおいて、「コースの終了状態に達した時には必ず終了時の状態が持つ値が集合 A に等しい」ということが言えれば、既存のデータベースシステムと置き換えて良いことがわかる。ここで Ψ をコースの終了状態を示す述語、 S をコースが返した値の集合、 A をあるデータベースのテーブルがもつ値の集合であるとする。 μ -calculus^[3]の言葉で表現すると、あるコースが、 $\Phi \stackrel{\text{def}}{=} \mu.Z((\Psi \wedge S = A) \vee (-))$ という性質を持つ

ときに、このコースと、前述のSQLの意味を持つデータベースと置き換えてよい。

4 応用

ここではこれまでの議論の応用として、揭示版システムを考えてみる。まず、それぞれの揭示の項目に対応する B_i を考える。 B_i は簡便のため初期値 \emptyset をあらかじめ持っているものとする。この時、 B_i は W によって値をセットされると、新たに値をセット (write) し直されるか、もしくは一度だけクライアント C からのデータの読みとり (read)を許すものとする。

$$\begin{array}{l}
 B \stackrel{\text{def}}{=} \prod_{i \in I} B_i \\
 B_i(x) \stackrel{\text{def}}{=} \overline{out}_i(x).in_i(y).B(y) + in(z).B(z) \\
 W \stackrel{\text{def}}{=} \sum_{i \in I} \overline{write}_i.W \\
 C \stackrel{\text{def}}{=} \sum_{i \in I} \overline{read}_i.C \\
 DB \stackrel{\text{def}}{=} (C | B | W)[f] \setminus L \\
 \text{where } f = \{\overline{write}_i / \overline{in}_i, \overline{read}_i / \overline{out}_i, i \in I\} \\
 L = \{\overline{in}_i, i \in I\}
 \end{array}$$

5 まとめ

本論文では関係データベースとネットワーク上に展開される複雑な問い合わせに対応するために既存のデータモデルを補完するための方法、特にある性質を持つコースとデータベースを同一視し、コースを選択する“層”をデータベースに応用することを試みた。しかしこの試みは同時に、パフォーマンスと、停止性などに代表される並列プロセスの問題も取り込むことになる。このため、安全で、かつ最適化された既存のデータベースと、プロセス論理に基づくコースとの取捨選択のためには、実際の応用に基づく実例と、プロセス論理の両面からの検証と、コース記述言語の開発が課題となる。

参考文献

- [1] 柏木 将宏, 大矢野 潤, “多層化されたサービスを可能にするインフラストラクチャの構築”, 情報処理学会第 57 回全国大会講演論文集, 1998
- [2] 大矢野 潤, “コンピュータ上の投票システムの開発”, 公共選択の研究 第 27 号, 1996
- [3] Julian Bradfield, “Verifying Temporal Properties of Systems”, Birkhauser, 1992
- [4] Robin Milner, “Communication and Concurrency”, Prentice Hall, 1989
- [5] Jeffrey D. Ullman, “PRINCIPLES OF DATABASE AND KNOWLEDGE-BASE SYSTEMS I, II”, Computer Science Press, 1989