

オブジェクト間の演算可能関係

原嶋 秀次 永瀬 恵子 岡宅 泰邦

東芝 システム・ソフトウェア技術研究所

演算の結果、異なる型のオブジェクトが生成されることを考慮したオブジェクト指向データモデルを提案する。オブジェクト指向データモデルでは、データの操作はメソッドによって実行される。しかし、オブジェクトの構造（データやメソッドの構成）に対する操作（これを演算と呼ぶ）については、一般的方法が提案されるに至っていない。ここでは、クラス間の演算可能関係なる概念を導入し、これに基づいて演算を定義する。これにより、オブジェクトの構造変化を伴う演算の定義が可能となった。また、オブジェクトの構造変化について考察し、関連する演算を提案する。これらの演算により、ユーザの視野の変化に柔軟に対応できるオブジェクト指向データベースの構築が可能となる。

The Operational Relation among Objects

Shuuji HARASHIMA, Keiko NAGASE and Yasukuni OKATAKU

TOSHIBA Corporation, Systems & Software Engineering Laboratory

Yanagi-cho 70, Saiwai-ku, Kawasaki, Kanagawa, 210, JAPAN

We propose an object-oriented data model in which structures of objects are variable. In an object-oriented database, data can be manipulated through appropriate method. But there is no way to modify structures of objects. The proposed data model introduces operational relations among object's classes, and defines operations for object's structures based on this relation. Using these operations it becomes possible to realize more flexible object-oriented database system whose structure varies in accordance with the changes of environments.

1. はじめに

オブジェクト指向データベースは、データ記述の柔軟性から、近年注目されており^{[1]、[2]}、多くの研究成果が報告されている。

ここでは、オブジェクトの持つデータ構成や手続きに対する演算（これはデータ型に対する演算の拡張と考えることが出来る。以後、この意味での操作を単に演算と呼ぶ）について考察し、オブジェクトの構造変化を考慮したモデルを提案する。

このモデルでは、全ての演算をクラス間の「演算可能関係」にもとづいて定義する。これにより、オブジェクトの持つデータの性質や手続きの変化を表現することが可能になる。特に複合オブジェクトの構造変化が表現できるようになり、データの見方が時間や利用者によって変化する場合のデータの扱いが容易になる。

2章では、例をもとにオブジェクトに対する演算を考察する。

3章では、データベースの形式的定義を与える。

4章では、いくつかの演算の定義とその意味について述べる。

2. オブジェクトの演算について

オブジェクト指向データベースの特徴はデータと手続きのカプセル化であり、研究の中心もこの点にある。

これまでは、ある時点のデータの表現法がテーマであり、クラスとインスタンス、*is-a*関係、*part-of*関係、複合オブジェクトなど、データ表現に関する重要な成果が報告されている。

ところで、データベースは多くのユーザが様々な視野（ビュー）で利用するので、データに要求される性質も動的に変化する。例えば、ある会社の社員リレーションに対して、あるユーザはある社員の情報を知りたいかもしれないし、別のユーザは年収が1,000万円以上の社員の情報を知りたいかもしれない。これまでこのような視野の違いには、様々なデータ操作法を提供する事に対応してきた。例えばリレーショナルデータベースにおける関係代数の提供がこれに相当する。

しかし、オブジェクト指向データベースでは、この操作は新たな構造のオブジェクトを作ることに対応する。このようなオブジェクトの構造変化の必要性について、次の例で考えてみる。

例2.1 自動車は、生産工場ではエンジンやボディをオブジェクトとする複合オブジェクトとして表現されるとする。

ここで、この自動車が売られ修理工場で燃料系統一式を取り出して機能テストする場合を考えると、燃料系統は独立したオブジェクトとして表現するのが望ましい。しかし、燃料系統の部品は、生産工場ではそれぞれ異なる組立ユニットに組み込まれているかもしれない。つまり、燃料系統なるオブジェクトは、生産工場での表現中には存在しなかったもので、このような視野に対して適当な構造のオブジェクトを新たに作ることができる機構を提供してやる必要がある。（図2.1.参照） □

このような視野の変化に対応するためには、オブジェクトの持つデータ構成や手続きなどの構造を変化させる演算が必要となる。しかし、これまでは

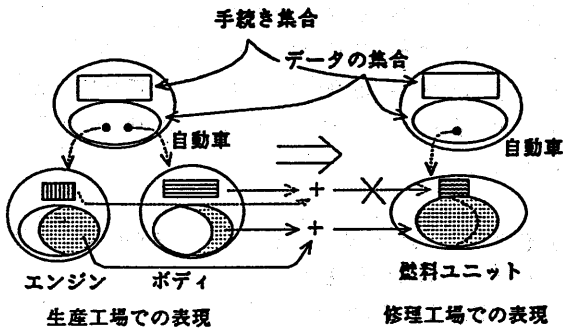


図2.1. 生産工場と修理工場での自動車の表現

このような演算についての研究があまりなされておらず、関係モデルをもとにしたデータを対象としたいくつかの演算の提案がなされているのみである[3][4]。

本報告では、演算可能関係に基づき演算の概念をまとめる。演算可能関係は、どのクラスのオブジェクトにどの演算を施すと、どのクラスに属するオブジェクトが生成されるかを定義するものであり、オブジェクトの持つデータの性質と手続きの変化、つまり一般化された型変化を扱うことが可能となる。

3. 定義

本章では、いくつかの基本定義とともに、データベースを定義する。

定義 3.1 基本データ (Atomic Data) :
数、文字、文字列が基本データである。 □

定義 3.2 データ (Data) :

1. 基本データはデータである。
2. データの集まり (集合、タプル、リスト) は、データである。
3. 次に定義するオブジェクトは、データである。 □

定義 3.3 オブジェクト (Object) : オブジェクトとは、データとそれに対する操作「メソッド」の集合から成る。オブジェクトは、一意な識別子 (ID) をもつ。 □

オブジェクトは、クラスかインスタンスに分類でき、インスタンスには、必ず対応するクラスが存在する。

定義 3.4 クラス (Class) :

クラス X とは、

$(M(X), A(X), D(X), DV(X))$

なる 4 項組である。ここで、

$M(X)$: メソッドの集合

$A(X)$: インスタンス変数と呼ばれる、データ記述の集合

$D(X)$: クラス変数と呼ばれる、このクラスに属するオブジェクトに共通なデータ記述の集合

$DV(X)$: $D(X)$ のデータ値である。 □

定義 3.5 インスタンス (Instance) :

クラス X のインスタンス x とは、 $(M(X), A(X), AV(x), D(X), DV(X))$

なる 5 項組である。ここで、

$AV(x)$: $A(X)$ のデータ値

である。 □

定義より、各インスタンスは、メソッド集合、インスタンス変数の集合とその型、クラス変数の集合とその値をクラスから継承 (Inherit) していることがわかる。

以下、インスタンスをオブジェクトと呼ぶことにする。クラス X のオブジェクト x を、 X に属する x と呼ぶ。オブジェクトとしては、集合型、タプル型[4]などを考える。また、オブジェクトのデータにもとづき、複合オブジェクトを定義する。

定義3.6 複合オブジェクト (Complex Object): オブジェクト x 内のいずれかのデータがオブジェクト y である時、 x を複合オブジェクトと呼ぶ。□

定義3.7 [概略] オブジェクト等価性 (Object Equality): オブジェクト x と y は、互いのデータ及び手続きが等しいとき、オブジェクト等価 (等価) である。複合オブジェクトでは、それを構成するオブジェクト間の親子関係が等しく、それぞれ対応するオブジェクト同士が等価であるとき、オブジェクト等価である。□

例3.1 図3.1(a), (b)の2つの複合オブジェクトは、等価である。これに対し、(a)と(c)は等価ではない。これは、例2.1の生産工場と修理工場における車の違いに相当する。□

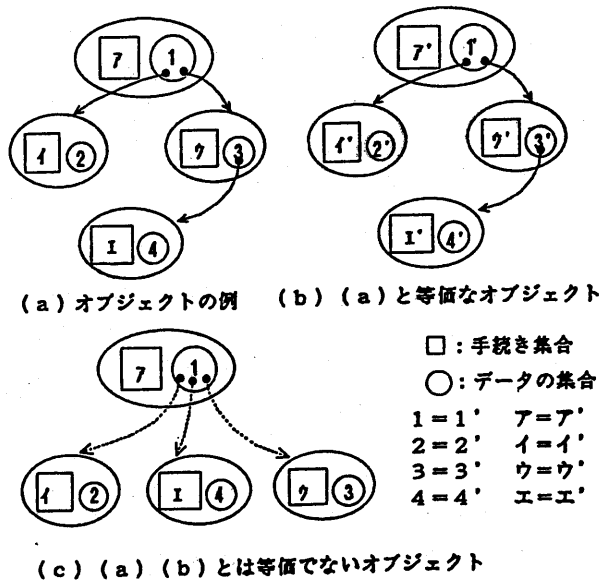


図3.1. オブジェクトの等価性

演算可能関係導入のねらいの1つは、このような視野の違いに応じたオブジェクトの構造変化の記述の必要性にある。つぎにデータベースシステムを定義する。

定義3.8 データベース (Database): データベース DB とは、

(C, I, OR, CR, OP)

なる5項組である。ここで、

C : クラスの集合

I : オブジェクトの集合

OR : 演算可能関係の集合

$C^+ \times OP \rightarrow C^*$

但し、 $C^* = \{\lambda, C, C \times C, \dots\}$

$C^+ = C \times C^* = \{C, C \times C, \dots\}$

λ : 要素の無い集合

\times : 直積

CR : オブジェクト間関連の集合

$\{C \cup I\}^+ \rightarrow \{C \cup I\}^+$

但し、 U : 集合の和演算

OP : 演算の集合

である。 OR の定義において、右辺が C^* であるのは、オブジェクト削除演算を考慮したためである。□

演算可能関係は、おのおのあるクラスに属するオブジェクト群に、ある演算を施すと、演算可能関係で決められたクラスに属するオブジェクト群が生成されることを定義する関係である。

以下、関係名であるか、演算名であるか明かな場合は特に記述しないことにする。

4. 演算

、本章では、データベースにおける演算について述べる。

定義4.1 演算(operation)：データベースの演算可能関係の集合をORとする。

$$R: C_1 \times \dots \times C_n \times r \rightarrow C'_1 \times \dots \times C'_m \quad R \in OR$$

であるとき、演算 r によって C_1, \dots, C_n に属するオブジェクトから C'_1, \dots, C'_m に属するオブジェクトが生成される。また、 C'_1, \dots, C'_m の各オブジェクトのデータ (d'_1, \dots, d'_m) は、もとの n 個のオブジェクトのデータ (d_1, \dots, d_n) から $(d'_1, \dots, d'_m) = fr(d_1, \dots, d_n)$ として定義される。 fr は、演算 r とともに定義される関数である。□

例4.1：水素と酸素の酸化物は水である。このことは、クラス「水素」、「酸素」と「水」の間に次の演算可能関係「酸化」を定義することで記述できる。

$$\text{酸化： 酸素クラス} \times \text{水素クラス} \times \text{酸化} \rightarrow \text{水クラス}$$

また、 H, O をそれぞれ水素、酸素オブジェクトとすると、 H と O に対して酸化演算を実行することにより水オブジェクト W が生成される。また、 a, b, x をそれぞれ酸素、水素、水のモル数であるとする、 x はつぎのような関数 f 酸化で与えられる。

$$x = f \text{ 酸化} (a, b) = \min(a, 2b)$$

図4.1にこの状況を示す。□

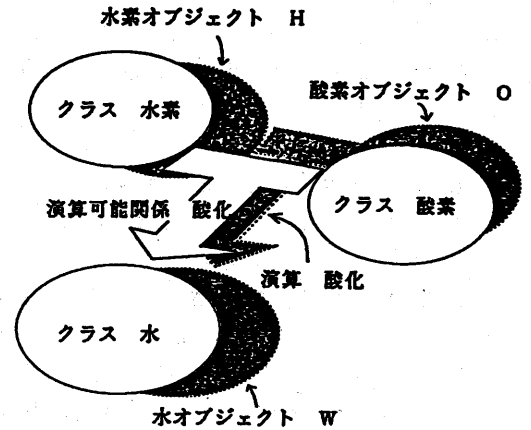


図4.1. 演算可能関係酸化とその演算

演算可能関係の概念にもとづき[3]、[4]などで提案している関係演算をベースとした演算も矛盾なく定義することが可能である。

例2.1.で示したように、オブジェクト指向データモデルにおいては、複合オブジェクトを含むオブジェクトの構造の概念が重要な意味を持つ。以下では演算可能関係の概念にもとづく、オブジェクトの構造に関する各種演算を定義する。

最初の2つは、複合オブジェクトにおけるオブジェクト間の親子関係に関するもので、親子関係の作成と消滅をおこなう演算である。

定義4.2[概略] 関係付け(make relation) ← : クラス W, X, Y, Z の間に $W \times X \times (\leftarrow) \rightarrow Y \times Z$

なる演算可能関係「関係付け」が存在するとき、 w の x との関係付けとは、(オブジェクトとしての) x を w のデータとし、それに伴って必要となる w の手続き集合の変更を行う(変更後のクラスを Y とする)演算である。また、

子供となったオブジェクト x の手続き集合の変更後のクラスが Z である (x は変化しないと考え、右辺を $Y \times X$ とすることもある)。

定義 4.3 [概略] 独立 (independence)
 : クラス W, X, Y, Z の間に
 $Y \times Z \times | \rightarrow W \times X$

なる演算可能関係「独立」が存在するとき、 y からの z の独立とは、 y の (オブジェクトとしての) データ z と y の関係付けをなくし、 y の必要な手続きの変更を行う (変更後のクラスを W とする) 演算である。また、独立後の z の手続き集合の変更後のクラスが X である (z は変化しないと考え、右辺を $Y \times Z$ とすることもある)。

次の定義は、オブジェクトの統合と分割をおこなう演算である。

定義 4.4 [概略] 取り込み (acceptance)
 $<$: クラス X, Y, Z の間に
 $X \times Y \times (<) \rightarrow Z$

なる演算可能関係「取り込み」が存在するとき、 x の y の取り込みとは、 y のデータを x のデータの一部とし、それに伴って必要となる x の手続き集合の変更を行う (変更後のクラスを Z とする) 演算である。

取り込み演算の概念は、集合型オブジェクトにおける和演算、タプル型演算における結合演算の概念を含んだものである。

定義 4.5 [概略] 分離 (separation) $>$:
 クラス X, Y, Z の間に
 $Z \times > \rightarrow X \times Y$

なる演算可能関係「分離」が存在するとき、 z からの y の分離とは、 z のデータの一部を分離し必要な手続き集合の変更をおこない (この結果、クラス X のオブジェクト x が生成される)、さらにこの時、分離したデータをデータとするクラス Y のオブジェクト y を生成する演算である。

図 4.2 に各演算の概要と演算間関係を示す。

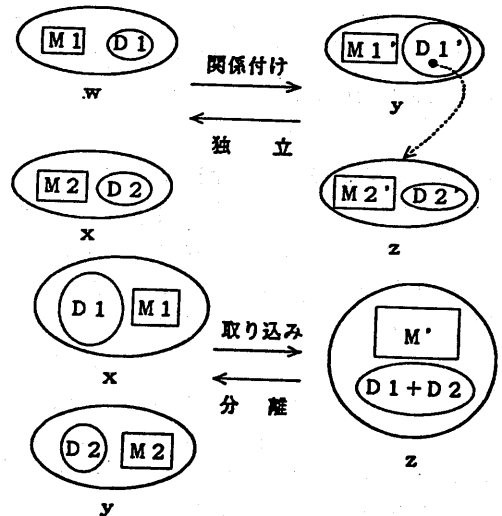


図 4.2. 各演算の概要とその関係

例4.2 例2.1について考える。生産工場から出た車（エンジンやボディといったオブジェクトで構成される複合オブジェクト）に対して、取り込み演算を実行し、1つの（複合オブジェクトでない）オブジェクトにする。その後、分離演算を実行し燃料系をオブジェクトとして持つ複合オブジェクトを生成することにより、手続き集合の変化を含めて、それぞれの視野を提供することが可能となる。 □

このように、演算可能関係の概念の導入により、手続き集合の変更を含めて、異なる視野を提供することが可能となる。

さらに、「現在存在する材料から何がどれだけ生成できるか？」など、これまでユーザがトランザクション内で作成していた視野を提供することも可能となった。

5.まとめ

ここでは、クラス間に演算可能関係の概念を導入したオブジェクト指向データモデルの提案をおこなった。

演算可能関係をもとに演算を定義することにより、

- ・データの性質の変化を伴う演算のデータベース中での表現が可能となった。
- ・同時に演算によって生成されたオブジェクトの手続き構成を定義することが可能となった。
- ・取り込み、分割演算の導入により、複合オブジェクトの構造の変更が可能となった。

これらにより、現実世界でのデータの扱い方や見方の変化を、データモデル中で扱うことが可能となった。

現在、知的分散システム^[5]上に基本システムとして、本モデルのサブセットとなるオブジェクト指向拡張分散RDBを構築中である。

今後、データ操作言語など、実用化に関する詳細研究を進める予定である。

参考文献

- [1] J. Peckham et. al.: "Semantic Data Models", ACM Computing Surveys, Vol. 20, No. 3, pp. 153-189, Sept. 1988.
- [2] M. Atinson et. al.: "The Object-Oriented Database System, Manifesto", Proc. of The First International Conf. on Deductive and Object-Oriented Databases (DOOD89), pp. 40-57, Kyoto, Dec. 1989, ACM TODS Vol. 12, No. 4, pp. 525-565, Dec. 1987.
- [3] F. Bancilhon and S. Khoshafian: "A Calculus for Complex Objects", Proc. of ACM PODS, pp. 53-59, March, 1986.
- [4] K. Tanaka and T. S. Chang: "On Natural Joins in Object-Oriented Databases", Proc. of DOOD89, pp. 279-293, Kyoto, Dec. 1989.
- [5] S. Tamura et. al.: "DISTRIBUTED COMPUTER SYSTEMS IN FMS -INTELLECTUAL DISTRIBUTED PROCESSING SYSTEM-", Proc. of The Japan Society of Mechanical Engineers MSET 21, pp. 497-502, Tokyo, May, 1990.