

複合オブジェクトを表す  
セマンティック ID方式

\*

豊田 雅信

足立 武史

龍 忠光

富士通ネットワークエンジニアリング(株)

(株)富士通研究所

複合オブジェクトは、クラス群にリンクを付けて階層化した”階層クラス”で表される。

複合オブジェクトを表現するために、クラス構造をクラス群とインスタンス部に分けることにより簡単な記号の組で記述したセマンティックID方式を提案する。

ここでは、本方式を用いることにより複合オブジェクトが1つのコマンドとして捉えられ、複雑なモデルを容易に操作できるようになることを説明する。

S E M A N T I C I D M E T H O D T O  
E X P L A I N C O M P L E X O B J E C T S

Masanobu Toyota Takeshi Adati

Fujitsu Network Engineering Ltd.

3-2-1, Sakado, Takatsu-Ku, Kawasaki 213, Japan KSP/R&D

Tadamitu Ryu

Fujitsu Laboratoris Ltd.

1015 Kamikodanaka, Nakahara-Ku, Kawasaki 211, Japan

We can explain complex objects by hierarchic class added to link in a group of classes. We propose semantic id method which can describe complex objects with simple id pair explained class and instance part.

In this paper we describe that we can think of complex objects as a command and operate easy a complex model by using this method.

## 1. はじめに

モノの振る舞いを中心に捉えて、現実世界をモデル化するパラダイムであるオブジェクト指向が、さまざまなメリットを掲げてやっと企業レベルで取り上げられるようになってきた。しかし、飛躍的な効率アップを望む上で、現時点でのオブジェクト指向技術では、まだ満足できない点が多いのも事実である。そうした、万能ではないオブジェクト指向を有用なものとするために、ぜひとも現時点でのオブジェクト指向の改善が望まれる。克服しなければならない課題として①モデリングが困難である②複雑なリンク構造が困難である③性能が悪い④部品の再利用が困難である⑤データベースを分散で使いにくいなどがあげられる。本稿では、これらのうちの部品の再利用に焦点をあてて報告する。

我々は、部品の再利用を徹底化するために、クラスオブジェクトは、実行するためのオブジェクトではなく再利用するための部品であり、実行するためのオブジェクトがインスタンスオブジェクトであるという考えをとる。そのために必要な技術は、部品の性質を徹底的に表すメタデータの採用、部品の性質を自由に追加・変更・削除できる方法の採用、それらの部品の検索を早めるためのオブジェクトIDのコマンド化の採用などがあげられる。複合オブジェクトを支援するためには、それがどのような構成になっても、それらのオブジェクトを全体として扱える操作が提供されなければならない。そこで、1つの複合オブジェクトの操作でそのすべての構成要素に推移的に伝播するセマンティックID方式が生まれた。このIDは強力なデータ操作を行うための基本要素となり、集合、組、再帰的な複合オブジェクトを操作するための基盤となる。

本論文では、オブジェクトのコマンド化を実現するために必要なオブジェクトの本質的な捉え方を述べたあと、コマンドの高度な表現方式であるセマンティックID方式を提案する。その後、セマンティックID方式による複合オブジェクトの捉え方が実世界をモデリングする上で、いかに効率的であるかを報告する。

## 2. 実世界のオブジェクトモデル化

### 2.1 オブジェクトの世界

オブジェクトは、クラス(静的モデル)とインスタンス(動的モデル)とメソッド(機能モデル)から構成される。オブジェクトの世界は外延/内包で実世界をとらえる。外延によって内包を隠蔽し内包に機能モデルを与えることで外延のみで実世界をモデル化できるようになる。これより実世界は、社会のしくみを表現する静的世界にクラス間の因果関係を付加したものと時間的流れを表現する動的世界で表される。

すなわち、オブジェクトの本質とは静的世界を表す静的モデルがクラスや複合クラスのテンプレートの世界であり、動的世界を表す動的モデルが、静的世界で作られたクラスを使って時間的な動きをするようにインスタンス化した世界でインスタンス間はメッセージパッシングを使って通信する。また、内包の隠蔽の世界が機能モデル(メソッド)である。

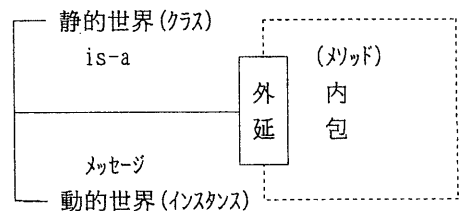


図1 : オブジェクトの世界

### 2.2 属性の拡張

現状のオブジェクト指向にとらわれず、オブジェクトというものの本質からオブジェクトを捉えるために、インスタンスのみ、メソッドのみ、それらを複合した複合オブジェクトなど、複雑な多数のオブジェクトを1つのオブジェクトモデルで置き換えた。このように、属性のみでなく、パターンも関係もすべてオブジェクトとして扱うことにより、すなわち実世界から出てくるあらゆる条件、たとえばデータ、メソッド、カプセル、メッセージなどを同格のオブジェクトコマンドで置き換えることにより、オブジェクトのコマンド化を実現した。

### 2.3 メタデータの拡張(Nature)

格納されたオブジェクト(実データ)を効率よく引き出すために、実データの付加情報としてメタデータを拡張して付与した。メタデータには、識別、表現、関連、格納場所だけではなく、テンプレート(is-a階層, part-of階層), 因果関係(時間的, 空間的, 法則的, 人為的)なども付加し, これを我々はNatureと定義する。

### 2.4 オブジェクトコマンド(Command)

メタデータは実データの説明文になっているために情報量が大変多い。先程述べた属性の拡張により, あらゆるものをオブジェクトとみなしオブジェクトIDで置き換えた。さらに, 我々はオブジェクトIDに意味付けしたのとしてオブジェクトコマンド(Command)を定義する。すなわち, オブジェクトコマンドは, 情報量の多いメタデータを圧縮記号化したものである。

### 2.5 オブジェクトデータモデル

実世界を外延と内包で受け止め, これをオブジェクトデータモデル(オブジェクトの複雑な捉え方を一律のモデルで表す概念)として以下のように捉えた。オブジェクトデータモデルは, オブジェクト指向で捉えた概念モデルを論理モデル化したもので, 捉えたデータにCommandを与え, 実データはEntityに格納し, その実データの属性やつながりをNature(リンクデータ)として各内部テーブルに保有してオブジェクトデータモデルに変換した。これにより, 外延, 内包の考えでは実世界をメタファー的にしか捉えられなかったが, オブジェクトデータモデルによりデータモデル的にも捉えられるようになった。

#### (1) Command

内部のみで動くIDでオブジェクトIDと複雑なオブジェクトを表現する(サロゲイト部)からなる。

#### (2) Nature

- ①エンティティデータの格納場所
- ②オブジェクトとしての性質
  - ・外延(インスタンス名, クラス名)
  - ・内包(メソッドなど情報隠蔽されたもの)

オブジェクトデータモデル

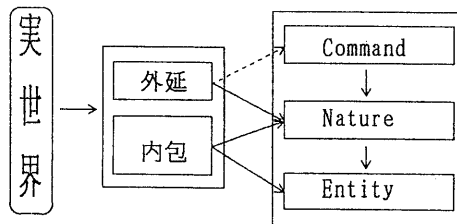


図2:オブジェクトデータモデル

#### ③オブジェクト自身もしくはオブジェクト間の関連

- ・仕組み → is-a, part-of
- ・因果関係

#### ④オブジェクトの管理データ

### (3) Entity

- ①実データ
- ②複合データ

### 2.6 オブジェクト部品の構成

オブジェクト部品はNatureとEntityで構成されており, コマンドリンクテーブルより引き出される。

#### (1) コマンドリンクテーブル

コマンドリンクテーブル処理は, 通常ディレクトリ処理とも呼ばれコマンドリンクデータを作り管理する。1つの新しい部品が出来るとき自動的にオブジェクトコマンドを割り振り, EntityやNatureの格納場所を割り付けてコマンドリンクテーブルを作成する。

#### (2) Entity

Entityは, 以下に示す2つの情報より構成される。

- ①データ
- ②メソッド

#### (3) Nature

Entityに意味を付けて説明したデータであり検索用, 管理用を使用する。このメタデータは圧縮されたコードの構造体で, オブジェクトコマンドとなる。

- ①Entityの説明文
- ②Entity関連データ
- ③Entity検索用データ

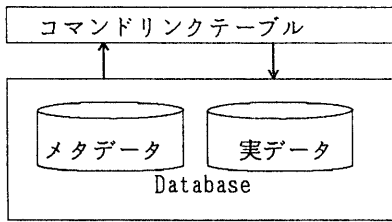


図3:オブジェクトの部品化

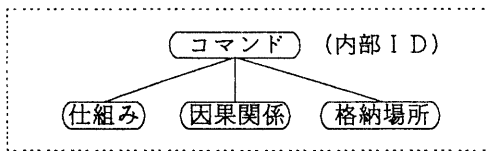
## 2.7 オブジェクト部品の関係図

オブジェクト部品は、コマンドリンクテーブルにより結びつけられる。

コマンドリンクテーブルは、オブジェクト部品の実データとメタデータをリンクしている。

モード1はメタデータにリンクしていて、仕組み、因果関係、格納場所を表す。モード2は実データそのものを表す。

コマンドリンク(モード1)



コマンドリンク(モード2)

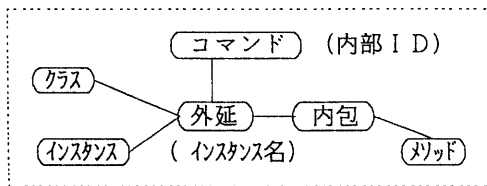


図4:コマンドリンク

## 2.8 複合オブジェクト

複合オブジェクトには静的モデルが複合オブジェクト化されたものと、動的モデルが複合オブジェクト化されたものがある。

### (1) 複合オブジェクト化(静的モデル)

#### ①part-ofの複合オブジェクト化

```
class X {
    クラスA を参照;
    クラスB を参照;
    クラスC を参照;
}
```

クラスをXとし、参照クラスをA,B,Cとするpart-ofリンクで複合オブジェクト化を行い、 $A \rightarrow B \rightarrow C$ という動作シーケンスをカプセル化して隠蔽すると、Xという外延のみで表される。

#### ②is-aの複合オブジェクト化

```
class B {
    クラスA をインヘリット;
}
```

スーパークラスをA,サブクラスをBとするis-aリンクで複合オブジェクト化を行うと $A \rightarrow B$ が成り立つ。

### (2) 複合オブジェクト化(動的モデル)

#### ①イベントの複合オブジェクト化

プログラム順序X  
 ↓  
 メッセージA  
 ↓  
 メッセージB  
 ↓  
 メッセージC

“動的世界のすなわち小さな意味のある動き”イベント”を複合化することで、メッセージA,メッセージB,メッセージCをプログラム順序Xで表し、このXを外延とする。

なお、実データは複合オブジェクトの場合が多いので、オブジェクトIDで複合化する。

## 3. オブジェクトコマンド

### 3.1 セマンティックID方式

オブジェクト指向プログラミングにおけるオブジェクトを一義的に表現するために、オブジェクトIDを付与する新しい方法として、セマンティックID方式を提案する。

### 3.2 旧方式の欠点と本方式の特徴

従来は、オブジェクトにつけるオブジェクトIDは、単なる順番を示す記号などが多く、他のオブジェクトと区別したり、同じオブジェクトIDの場合には同一なのかどうかという判断にしか使われなかった。本方式では、サロゲイト部により複雑なE-Rモデルを表し、そのインスタンス部をオブジェクトIDで表現することにより以下のようなことを可能とした。

(1) ネットワーク上で、このセマンティックID

を見るだけで、そのオブジェクトの実体を見なくてもどんな振る舞いを持っているかがわかる。このため、このセマンティックIDのやりとりで、大体の振る舞いの分析が可能となる。

- (2) このセマンティックIDにより、スペックに必要なプログラムを分散協調で集めることが可能となる。
- (3) コモンオブジェクトモデル化

### 3.2 オブジェクトコマンドの実現

オブジェクトコマンドは、オブジェクトコマンド= サロゲイト+ オブジェクトIDで表される。サロゲイトは、メタデータの代用として使用される。サロゲイトが先程述べたコマンドリンク(モード1)で引き出すコードとなっておりオブジェクトIDとして使うところが実データを引き出すコマンドリンク(モード2)となっている。サロゲイトの部分は、メタデータのそのままの意味を圧縮したコードで表すようにしておりサロゲイトを見れば実データの意味を書いたメタデータのことを、ほぼ理解できる。しかし、オブジェクトIDと呼ばれるところはインスタンスデータを表してはいるが、これはインスタンスデータそのものを表してはいない。インスタンスデータは、サロゲイトが複雑になるとR-DBの正規化されたスキーマが多数出てくるからである。よって、インスタンスデータを表すオブジェクトIDは代表IDとなっている。

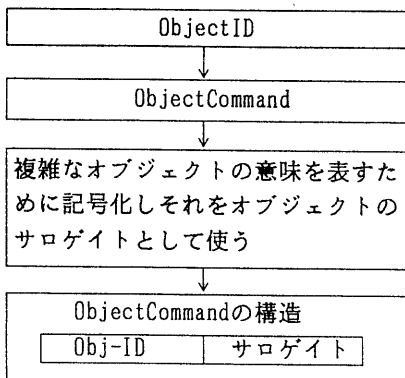


図6:オブジェクトコマンド

### 3.3 オブジェクトコマンドの構造

自動的に作成されるコマンドは、基本的に以下に示すような構成となる。

コマンドには大きく分けて5種類のコードがある。1つはヘッダオブジェクトで、どの部分にメタデータが入っているか、is-aか、part-ofかを示したデータである。2つめはis-a系統でそのままハイアラキのテンプレートを表し、3つめはpart-of 系統の条件を表す。なお、第2レイヤのコマンドリンクによりpart-of 関係の親と子のテーブルが作成され、これにより順リンクと逆リンクが取れる。4つめは因果関係を表すためのデータ 5つめは、インスタンスデータの代表IDが入っている。

- (1) オブジェクトIDの構造をクラスに対するインスタンスを表す”オブジェクトID”とクラス群すなわち階層クラスを表す”サロゲイト”の2つで構成する。

- ①オブジェクトID
  - ・インスタンス表現部
- ②サロゲイト
  - ・ヘッダ部
  - ・is-a, part-of 表現(仕組み)
  - ・因果関係

- (2) ”サロゲイト”のクラス群を表すフォーマットによって複雑な複合オブジェクトを簡単な記号の組合せで表せるようにした。

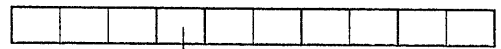


図7: コマンドフォーマット

### 3.4 コマンド管理方法

- (1) 階層化方式

サロゲイトのコードのところを、1桁~n桁のコマンドの階層で表現する。

- ① オブジェクトコマンドの桁数を判断する。
- ② n桁目の文字を判断する。
- ③ その文字のクラスをスーパークラスとしてインヘリットする。
- ④ それで集めたクラスで複雑な階層クラスを作りインスタンス部のデータを(オブジェ

クトID)によって与えられるデータを処理する。

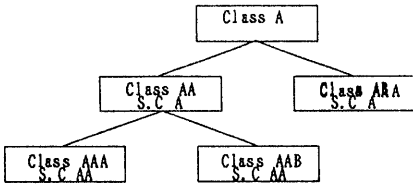


図8: 階層化方式

(2) 意味変換テーブル方式

サロゲイトのコードを2進のbitの集まりとし、それをコマンドと見て1つ1つを無関係に複合オブジェクトのコマンドに変えるために、意味変換テーブルを用意する。

- ① オブジェクトコマンドを見て、意味変換テーブルでクラス名に変換する。
- ② クラス名からその中に含まれているクラスのつながりに変換する。
- ③ 集めたクラスで階層クラスを作り、インスタンス部のデータを取り込み実行する。

オブジェクトコマンド	クラス名
0000	A
0001	AA
.	
01FF	ACDD

図9: 意味変換テーブル方式

### 3. むすび

本稿では、実世界に存在する複合オブジェクトを操作するための基盤となるセマンティックID方式を紹介した。まず、オブジェクトコマンドを導入するために必要となる基本的なオブジェクトの捉え方について述べた後、複合オブジェクトの操作方法について紹介した。

複合オブジェクトは、現実社会で直観的に集合と分かる”集合”の概念、実世界でおこる事象の順番や時系列データを表す”リストや配列”の概念、そしてある実体の性質を表す”組”の概念で表されるものであり、単純なオブジェクトを組み合わせた複雑なオブジェクトのこと

を言う。すなわち、複合オブジェクトは社会の仕組みや因果関係を表すためにクラス群にリンクを付け、クラスの階層化を行った”階層クラス”で捉えられる。このことから、セマンティックID方式により階層クラスを表すことが、複合オブジェクトを捉える上で都合がよいことがわかる。

今後はセマンティックID方式を実際に試行することによりこの有用性を確認していきたい。

### 参考文献

- [1] 龍, 佐藤, 高原: 「分散処理における新データモデルの提案 SAITA '90
- [2] 龍, 高原, 関口: 「情報ネットワークサービスの将来のために」  
応用科学学会vol. 4, No.2 1990.11
- [3] 龍, 佐藤, 若林「ネットワークシステム構築から見たオブジェクト試行データベースの提案」情報処理学会アドバンスト・データベース・シンポジウム1990.12
- [4] 村川, 龍「オブジェクト・センサ・モデルにおけるメタ・データ・アーキテクチャの提案」情報処理学会データベース・システム・シンポジウム1991. 7
- [5] 龍, 青江「分散システムにおけるファイル構造の提案」情報処理データベース・システムシンポジウム1990.7
- [6] 龍, 若林, 村川「オブジェクトの捉え方とオブジェクトセンサーモデル」  
情報処理学会データベース・システムシンポジウム 1991.7.17
- [7] 龍, 戸島, 村川, 豊田「自己組織化通信方式の提案」  
SAITA '92