

6C-2

ビジネスオブジェクトのための 永続オブジェクト自動生成ツール*

寺田 雅之[†] 内川 昌平[‡] 関根 純[§]NTT 情報通信研究所[¶]

e-mail: {terada, uchikawa, sekine}@isl.ntt.co.jp

1 はじめに

SoftwareIC[1]という言葉に象徴されるように、プログラムの再利用は比較的単純な機能を持つオブジェクト群から構成されるクラスライブラリによってなされてきた。一方、代表的な業務アプリケーションでは、オブジェクト間の関係が複雑であるため、プログラムの構造をいかに再利用するかが鍵となる。そのため、特定業務ドメインに特化することにより、構造を含めた再利用を可能とした Business Object と呼ばれるフレームワークによる再利用が注目されている。

Business Object の設計は、実際の業務を分析して抽出されたオブジェクトモデル(業務モデル)に基づいて行なわれる。しかしながら、業務モデルをそのまま実装のためのオブジェクトモデル(実装モデル)として用いることは困難であることが多い。その大きな理由の一つとして業務モデルと実装モデルとの間でアイデンティティの概念に差異が存在することが挙げられる。

本稿ではまず、アイデンティティの差異により発生する問題とそれを解決するための拡張 ID モデルについて述べ、拡張 ID モデルに基づいたオブジェクト永続化機構の自動生成ツールの実装方式について述べる。

2 拡張 ID モデル

Business Object は業務モデルに基づいて設計されたオブジェクト群から構成される。しかしながら、一般には現実世界をオブジェクトモデルとして表現した業務モデルと実際にシステムを構築する際に用いるオブジェクトモデルである実装モデルとは、その構成に隔たりがあることが多い。

この理由として、(1)業務モデルには(性能向上のための中間的オブジェクトなど)実装上にのみ必要なオブジェクトが抽出されないこと [2]、(2)クラスライブラリなどを用いるとそれによりオブジェクトの構造が業務モデルとは無関係に制約を受けること、(3)業務モデルより実装モデルのほうがオブジェクトの粒度が小さく、関連が複雑化する傾向にあること、などが挙

げられる。

Business Object はプログラム構造を含めた再利用を可能にするため、(1)(2)は Business Object の構築を困難にはするが、その再利用を阻害することはない。しかしながら、(3)については複雑化したオブジェクトを Business Object の利用者に見せることとなり、再利用性を阻害する要因となる。

我々は、(3)の問題が業務モデルと実装モデルとのアイデンティティの概念の相違によること、その理由の一つとして業務モデルでは文脈に依存して ID が設計されるのにも拘らず、実装モデルでは文脈の存在を許していなかったことが挙げられることを発見した。

例として、契約という業務モデルにおけるアイデンティティとして契約 ID を持つオブジェクトを考える。

ここで、契約 ID は実装モデルにおいてユニークなアイデンティティとなるとは限らない。例えば契約の有効期間などの情報を利用して履歴管理を行なう場合には、契約 ID だけではオブジェクトの同定するための情報として不十分であり、「X年Y月Z日の時点(日時)において有効である」という情報があわせて必要となる。すなわち、実装モデルのアイデンティティを得るために、(契約 ID という)業務モデルの ID と(日時という)文脈が必要となる。

これは業務モデルの分析が(時間の経過を意識せずに)ある時点のスナップショットという特定の文脈のもとに行なわれていることにより、文脈に依存したアイデンティティの設定がなされているためである。

一般的なオブジェクト指向設計では、このような業務モデルの分析に対しては実装モデルの設計時に実装固有のアイデンティティを付加することによって対処することが多い [3]。しかしながら、この設計は直感的な業務の理解とは異なり、システムを複雑化する原因となる。

そのため、我々は実装固有のアイデンティティを隠蔽し、業務モデルの ID と文脈によって Business Object のアイデンティティを得る拡張 ID モデルを提案する。このモデルでは、Business Object の利用者は業務モデルの ID と文脈を指定することによりオブジェクトを同定する。

例えば、前出の例は「X年Y月Z日の時点において有効な(文脈)契約 ID が ABC001(業務モデルの

* A Generator for Persistent Business Objects

[†]MASAYUKI TERADA

[‡]SHOUHEI UCHIKAWA

[§]JUN SEKINE

[¶]NTT Information and Communication Systems Labs.

ID)のオブジェクト」のように表現する。これはRDBにおける複合キーと類似するが、文脈は必ずしもオブジェクトの要素との完全一致条件で対応づけることができない点異なる。この例においても、「X年Y月Z日において有効」という文脈をオブジェクトに「有効期間」という属性を設けてそれとの完全一致により対応づけることはできず、例えば発効日時、失効日時などの属性を設けて「X年Y月Z日は発効日時と失効日時の間にある」という条件によって対応づける必要がある。

3 永続化機構の実装

本章では、拡張IDモデルに基づいたBusiness Objectのための永続化機構を自動生成するツールであるMSI-POSEの実装について述べる。

MSI-POSEは、Business Objectを定義したリポジトリからObject-Relational Mapperを自動生成するツール群である。Object-Relational Mapperは、Business Objectを格納するためのRDBMSのテーブル定義などと、RDBを永続記憶として格納されたBusiness Objectをアクセスするためのインタフェースを提供するC++クラス群から構成される。

以下では、前章で述べた拡張IDモデルを実装するための、業務IDと文脈からのオブジェクトの同定方式について述べる。また、拡張IDモデルにおいてキャッシュ機構を実装する際の問題点と、その解決方法を示す。

3.1 オブジェクトの同定方式

オブジェクトを同定する必要があるのは、以下のような局面である。

1. 業務モデルのIDによる永続記憶からのオブジェクトの取得。(検索)
2. オブジェクト間の関連の追跡。(参照)
3. オブジェクト間の同一性の判定。(比較)

検索 検索を行なうために、それぞれのオブジェクトはIDの検索条件を保持する。この検索条件は、前章の例では業務モデルのID id と(文脈を表わす)時刻 t を用いた条件式 $C(id, t) \equiv (\text{契約 } ID = id) \cap (\text{発効日時} \leq t) \cap (t < \text{失効日時})$ として記述する。

参照 参照は、参照先のIDと参照元の文脈を用いた検索として記述できる。

比較 同一性を判定するための比較については、以下の3方式が考えられる。

1. 業務モデルのIDのみが等しければ同一。

2. 業務モデルのID、文脈ともに等しい場合のみ同一。
3. オブジェクトIDが等しければ同一。

方式1では、同一であっても同値ではないという状態が発生する。また方式2では、同一のオブジェクトを比較しているにも拘らず同一ではないと判定されるという状態が発生する。一般的なプログラミング言語ではこれらの状態が発生することは考えられないため、両方式ともにプログラミングが困難になると考えられる。

そのため方式3を採用し、同一性の判定には隠蔽して保持されるオブジェクトIDを用いることとした。

3.2 キャッシュの実装

キャッシュを実現するためには、一般的にはIDをキーにしたハッシュが用いられるが、業務モデルのIDは一意性を保証するものではないため、キーとはできない。また、検索によるオブジェクト取得は業務モデルのIDと文脈の指定により行なうため、オブジェクトIDをキーにするとハッシュからのオブジェクトの取得が不可能となる。

この問題を解決するため、ハッシュのアルゴリズムがハッシュ値の計算と同一性比較の2段階から構成されることに着目した変形ハッシュ法を開発し、これによりキャッシュを実現した。

変形ハッシュでは、業務モデルのID id および文脈 T の部分¹ T' を用い、ハッシュ値 $H(id, T')$ を計算する。その後に行なう同一性比較は、同一ハッシュ値を持つキャッシュ内のオブジェクトそれぞれに対して、検索のための条件式 $C(id, T)$ を適用することによってオブジェクトを得る。

4 まとめ

現在、本方式に基づくオブジェクト永続化機構の自動生成ツールの実装を完了した。今後は本ツールで生成したBusiness Objectを用いて遠隔教育システムを構築することにより、本方式を評価する予定である。

参考文献

- [1] Cox, B. J.: *Object-Oriented Programming - an Evolutionary Approach*, Addison-Wesley, 1986.
- [2] Rumbaugh, J. et al.: *Object-Oriented Modeling and Design*, Prentice Hall, 1991.
- [3] Loomis, Mary E. S.: *Object Databases: The Essentials*, Addison-Wesley, 1995.

¹条件式 $C(id, T)$ の中で等値比較されている要素の集合