

## 再利用の新しい枠組みを求めて： オブジェクト指向開発でのパターンシステム

山本里枝子†, 上原忠弘†, 中山裕子†, 大橋恭子†, 吉田裕之†

再利用の枠組みとして、体系化された再利用部品群と、部品化作業と再利用作業を含む開発プロセスを組み合わせる提供することが重要である。本稿では、オブジェクト指向開発技術に関して、体系化された再利用部品群にパターンシステムが有効であることを指摘し、開発プロセスと組み合わせる適用した事例を報告する。

### Toward A New Reuse Framework: A Pattern System for Object Oriented Development

Rieko Yamamoto †, Tadahiro Uehara †, Yuko Nakayama†, Kyouko Ohashi†, Hiroyuki Yoshida†

The combination of systematic reusable components and development processes, which include building-components activities and reusing-component activities, is important for the reuse technology. We propose a "pattern system" for reuse in object oriented development and report a case study.

#### 1. はじめに

「再利用技術」は、ソフトウェア工学の歴史の中で繰り返し試みられてきた技術である。本稿では、再利用技術の枠組みを「再利用部品群」と「開発プロセス」の組み合わせで捉える。オブジェクト指向開発技術に焦点をあて、オブジェクト指向開発技術が「再利用技術」にもたらすメリットは「再利用部品」にあるとした上で、再利用部品開発の問題点とそれらへの解として「パターンシステム」の利用を提案する。併せて、開発現場での試みを報告する

#### 2. 再利用技術とオブジェクト指向

##### 2.1. 再利用の枠組み

我々は、再利用の枠組みとして、(1)体系化された再利用部品群とその使い方を説明する何らかの手段、(2)部品化作業と再利用作業を含む開発プロセスと支援環境、を組み合わせることが重要だと考える。

##### 2.2. 体系化された再利用部品群

再利用対象には、ソースレベルからビジネスモデルレベルまで、様々なレベルがある。全てのレベルに共

通して、一貫した基本方針に従っていること、部品の組み合わせを考慮して体系化されていることが重要である。また、部品そのものの説明に加えて、サンプルコードやシナリオなどで「部品の使い方」を利用者に示すと再利用時の作業効率が大きく向上する。

##### 2.3. 開発プロセス

再利用をソフト開発プロジェクトの方針の一つとしたなら、プロジェクトが採用する開発技法の一部として、部品化作業と再利用作業を含む開発プロセスを提示すべきである。既に構造化手法でも、一アプリケーションを構築する際に、先にアプリ内で再利用する部品を開発し、その後他の固有な部分を開発する手順が推奨されている。

開発技法に従った、作業支援用の各種ツールを組み合わせた開発環境が提供されると、作業効率がより向上する。

##### 2.4. 再利用部品と開発プロセスの組み合わせ

再利用部品と開発プロセスの組み合わせは、主にソースレベルを対象に構造化手法で試みられてきたし、また、オブジェクト指向開発でも行われている。フレームワークを再利用部品体系とし、「フレームワーク開発部隊」が先行して部品を開発し、「アプリケーション開発部隊」がその部品を再利用する、開発プロセスを採用したプロジェクトが報告され、その有効性が述べられている[1]。

†(株)富士通研究所 ソフトウェア研究部  
Software Laboratory, Fujitsu Laboratories Ltd.

## 2.5. オブジェクト指向により再利用の枠組みの何が変わるのか？

オブジェクト指向開発でも再利用に必要な基本的枠組みは変わらないと考えられる。が、再利用部品に関してはメリットが期待できる。

例えばシステム構築時、アプリケーションは、プラットフォーム等に関わらずビジネスロジック等のドメイン毎に決まる情報と、ドメインに共通な情報、ミドルウェアの制御方式などのアプリの基盤に関する情報、等の様々なドメインの情報を含む。これらは知識を持つ人も、また適用範囲も異なる。各ドメインのエキスパートたちの中には、経験から再利用したいと意識する情報を持ち、それらが「オブジェクト」という入れ物で考え通りに定義できるようになる、という人がある。また、異なるドメインに関してもクラスという共通の理解、統一的な分割と組み合わせの機構はメリットとなる。

但し、再利用部品を、「どの範囲に関して」、「どんな粒度で」、「どう体系化するか」の3点の問題は、オブジェクト指向であろうとなかろうと、普遍的な課題である。

範囲としては、ドメイン別が一般的であろう。アプリケーション規模もドメインを分ける指標になる。粒度と体系化に関して、我々は、オブジェクト指向に基づくパターンシステムが一つの解になると考えている。

パターンはエキスパートのノウハウを定義する良い手段である。パターンは部品を「こういう場合にはこうする」という視点で整理できる手段なので、ドメインのエキスパートが部品化を考える際の思考にあっている。パターンシステムは、これらパターンをある規則で関連づけて体系化したものである。体系化すると、再利用をする際に想定する部品を検索し易く、また、エキスパートが問題を詳細化していく道筋も再利用できる。

## 3. パターンシステムによる再利用事例

本節ではオブジェクト指向開発に関して、体系化された部品群として「パターンシステム」を、開発手順として部品化作業と再利用作業を取り入れたプロセスを採用した事例を報告する。

本事例は、分析モデルのクラス数が27の比較的規模が小さなビジネスアプリである。設計時に9クラス追加され、設計モデルのクラスは36個であった。

### 3.1. パターンシステム

パターンは「繰り返し起こる問題とその解決の組」であり、80年代後半に建築分野で提唱され近年ソフトウェア分野で再利用技術のひとつとして注目されている技術である。現在パターンの適用範囲は拡大しつつあ

り、GoF[2]のような設計時に適用するパターンだけでなく、分析時や実装時に適用するなど、さまざまなレベルのパターン[3]が提唱されている。

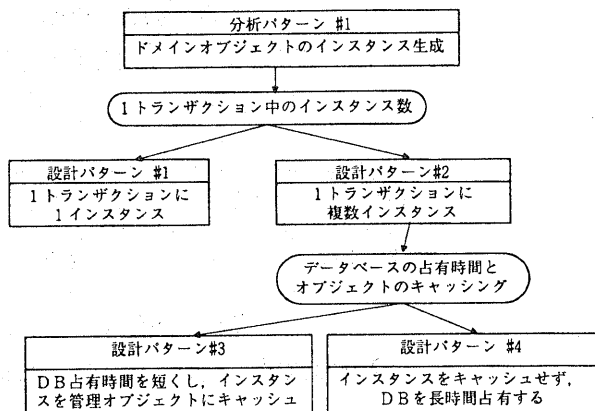


図1 パターンシステムの一部

「パターンシステム」は、ある一貫した基本方針で統一されたパターンを関連づけて体系化したものである。「パターンシステム」を、「パターン」と「パターン間の関係」を節点とする有効グラフの形式で定義する。事例では、問題を詳細化するという規則で、「パターン間の関係」に開発作業上の考慮点(issues)を採用して、考慮点間の関係、考慮点とそれに対する選択肢となるパターンを定義して、体系化した。このパターンシステムは、オブジェクトのライフサイクルに関するパターンシステムの一部を図1に示す。図中、四角がパターンを、角が丸い四角が考慮点を示す。あるパターンの結果に対して、別なパターンを適用して成果物を得ることを繰り返してアーキテクチャを詳細化していく。図1中の分析パターン#1の定義の一部を図2に示す。こ

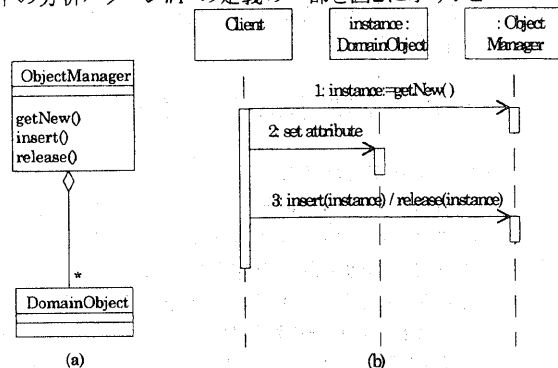


図2 分析パターン#1

これはモデル内の全てのドメインオブジェクトに関してインスタンスを得る際のオブジェクトの構造と振舞いを規定する。また設計パターン#3の定義の一部を図3に示す。分析パターン適用を前提として、「1トランザクション中で複数インスタンス数を扱う」、「DBの占有時間は短く、オブジェクトはキャッシングする」と判断した際に適用する設計パターンである。図3は振舞いを規定する。

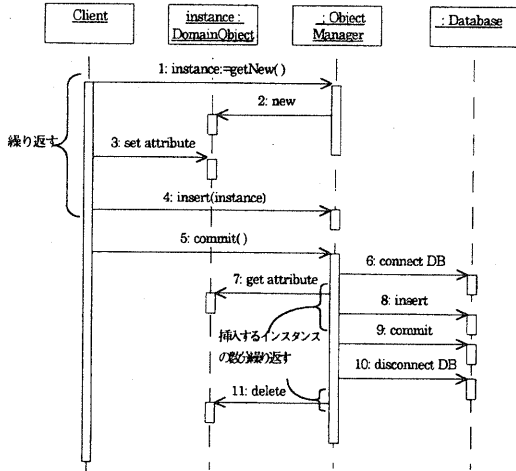


図3 設計パターン #3の振舞い

一般に適用容易で洗練されたパターンを作成するのは非常に難しいが、このように文脈を限定していくことで、過度な一般化が不要となり、プロジェクト内で必要なパターンが比較的容易に作成できる。オブジェクトのライフサイクルに関連するパターンシステムは、分析パターン数が5、設計パターン数が9となった。

### 3.2. 開発プロセス

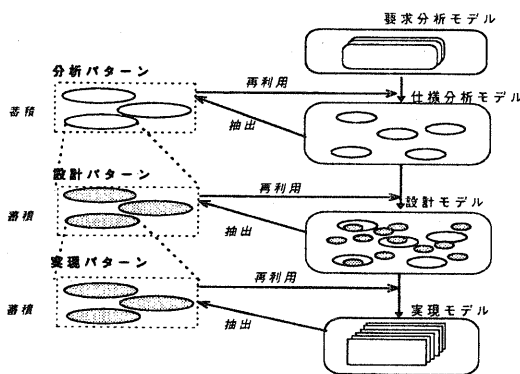


図4 パターン指向開発

開発プロセスとして、パターン指向開発を特徴とする OO-DESIGN2[4]を適用し、アーキテクチャ定義を目的としたパターン定義作業を少人数のグループが先行して行い、その後アプリケーション本体の開発を多人数の開発部隊が行う、開発プロセスを実施した。パターンはアプリケーションの各開発モデル毎に抽出し、各作業の対象となるモデルに繰り返し適用した(図4)。

### 3.3. パターンシステムの適用可能性

設計モデルにおけるパターン適用率を計測した。適用率はクラスのメソッドを対象とし、全てのメソッド数に対するパターンが適用された数の割合、とした。本事例の適用率は 52.4%であり、パターンシステムが再利用部品として十分に有用であることが示せた。但し、この数値は予想外に高い。これは、本アプリケーションの大部分がデータアクセスで、ビジネスロジックを殆ど持たなかったためであり、通常は今回のパターンシステムでは 2, 3 割の適用率が妥当だと考える。

なお、設計パターンは、先行チームが定義した文書をネットワークで共有し、開発部隊に常に最新のパターンを利用するよう奨励した。設計レビューの前と後での適用率の変化も計測したが大きな差はなかった。パターンシステムを提供すれば、開発者が自主的に再利用可能であることがわかる。

## 4. まとめ

再利用の新しい枠組みに関して、体系化した再利用部品群と、部品化作業と再利用作業を含む開発プロセスを組み合わせることを提案した。オブジェクト指向開発において、再利用部品としてパターンシステムを、開発プロセスとしてパターン指向開発を適用した事例を報告した。

パネルでは、オブジェクト指向開発での再利用上の問題点とそれらへのアプローチに関して議論したい。

## 参考文献

- [1] Sparks, S. et al. Managing Object Oriented Framework Reuse, *IEEE Computer*, Vol. 29, No.9, pp. 52-61(1996)
- [2] Gamma,E .et al. *Design Patterns*, Addison-Wesley, 1995
- [3] Bushmann, F. et al. A SYSTEM OF PATTERNS, John Wiley & Sons Ltd., 1996
- [4] 山本, 他. パターン指向とリスクドリブンの特徴とするビジネスアプリ向けオブジェクト指向開発技法, オブジェクト指向シンポジウム,1998