

## 構造化分析を利用したオブジェクト指向設計の適用

4S-2

堀 正弘<sup>1</sup>、河原 仁<sup>1</sup>、伊藤光恭<sup>2</sup>、高木一裕<sup>3</sup>

NTT<sup>1</sup>ソフトウェア研究所<sup>2</sup>移動体通信事業部<sup>3</sup>ソフトウェア開発センタ

### 1. はじめに

ソフトウェアの維持・保守に要するコストを抑え、修正・改造が容易なソフトウェアを設計するための設計手法として、オブジェクト指向設計法(OOD)<sup>[1][2]</sup>が提唱されている。リアルタイム通信処理システムへの具体的なOODの適用手法を確立することを狙いとして、構造化分析法(SA)<sup>[3]</sup>を活用してオブジェクトを抽出する方法について試行適用を実施した。本論文では、無線通信システムについて試行適用したOOD手法について報告する。

### 2. オブジェクト

#### (1) オブジェクトの定義

本論文では、型やデータとそれに関する操作(手続き)をカプセル化し、1つのプログラム単位にまとめたものをオブジェクトと呼ぶ。また、オブジェクトを決定する工程を「オブジェクトの抽出」と呼ぶ。

#### (2) 上位オブジェクトと下位オブジェクト

比較的抽象度の高いデータや操作をもつオブジェクトを上位オブジェクトと呼ぶ。上位オブジェクトのもつ操作は、「呼を接続する」といった抽象度の高い、ともすれば曖昧なものである。これを直接一つの手続きで実現することは、規模が非常に大きくなり複雑すぎるため、よりプリミティブなオブジェクトを用いて実現することになる。

一方、比較的抽象度の低いデータや操作をもつオブジェクトを下位オブジェクトと呼ぶ。

下位オブジェクトの操作は、例えば「モアムを起動する」といった即物的・具体的な操作であり、容易に1つの手続きとして実現できるものである。

### 3. オブジェクト抽出法

OODは次のステップからなる。

1. システムの要求分析
2. オブジェクトの抽出
3. オブジェクトの操作の抽出
4. オブジェクトの関連づけ

ここで述べる方法は、第1ステップにSAを使う方法である。

#### 3.1 初期の試み

オブジェクトの抽出法として、当初以下の方法を試みた。

まずSAを用いて要求分析を行ってデータフロー図(DFD)(図1)を作成し、データストアあるいはプロセス間を流れるデータからオブジェクトを抽出する。次に抽出したオブジェクトの関連に着目して、全体のオブジェクト

構成を決定する。

しかし、試行過程で以下の問題が明らかになった。

#### (問題点)

下位レベルのオブジェクトは具体的なものに対応し、データストアから容易に抽出することができる。しかし上位レベルのオブジェクトは概念が確立しているわけではなく、またDFDで使われている言葉にも個人差があるため、単にDFDのデータストアやプロセス間のデータから機械的に抽出するのは困難である。すなわち、対象となるシステムのモデル化が必要である。

そこで、上位オブジェクトをシステム設計書から直接抽出することにより、上記の上位オブジェクトの決定の曖昧さを解決することとした。

#### 3.2 改良版

改良したオブジェクト抽出法の概要は、以下の通りである。

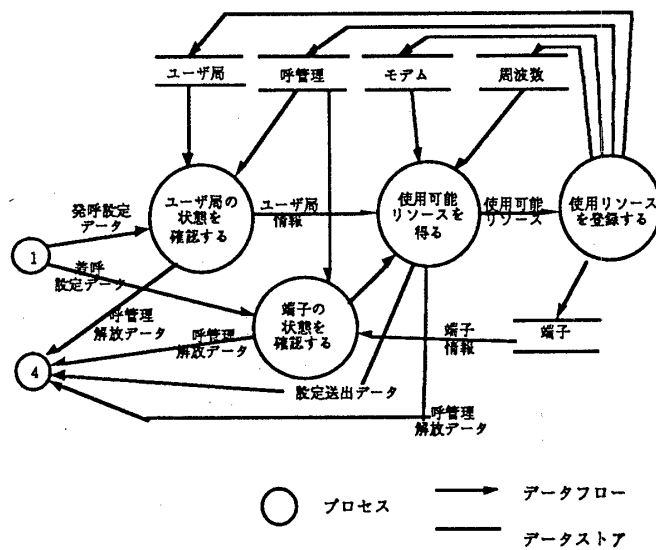
#### (概要)

まずSAを用いて要求分析を行ってDFDを作成し、データストアに着目して下位オブジェクトを抽出する。次に自然言語で記述されたシステム設計書から方式用語として明確化されている概念を抜き出し、それを上位オブジェクトとする。最後に、抽出した下位オブジェクトと上位オブジェクトとを融合して、システムを実現する。

#### (1) 下位オブジェクトの抽出

##### (a) SAによる要求分析

対象システムが大規模であり、そのままでは扱いにくいため、システムをサービス機能単位(呼処理機能、装置監視機能等)に分割し、サービス機能単位で要求分析を行い、分析結果を統合することとした。



○ プロセス → データフロー  
 ≡ データストア

図1.DFDの例

An Object Oriented Design Using Structured Analysis

Masahiro Hori, Hitoshi Kawahara, Mitsutaka Ito, Kazuhiro Takagi

NTT

(b) 下位オブジェクトの決定

(a)で作成したDFD中に出現するデータストアとそれにアクセスしているプロセスをグルーピングしてオブジェクトとする。

(2) 上位オブジェクトの抽出

(a) 方式用語の抽出

自然言語で記述されているシステム設計書中で使用されている方式用語とその用語を含む文とを全て抽出する。

(b) 上位オブジェクトの決定

(a)で抽出した方式用語をオブジェクトとし、その用語を含む文をオブジェクトに対する操作とする(例1)。

(3) 上位オブジェクトと下位オブジェクトの融合

(a) オブジェクト操作間の関連の決定

最上位のオブジェクトに対する各操作からはじめて、その操作を実現する上で必要となる、より下位のオブジェクトの操作を決定する。

(b) オブジェクト間の参照関係の決定

あるオブジェクトの各操作を参照する操作をもつ他のオブジェクトを抽出することにより、オブジェクト間の参照関係を決定する(図2)。

例1 上位オブジェクトの抽出例

通信回線未設定状態において「ユーザ局」から「回線設定要求信号」を受信すると「回線状態のチェック」を行い、OKの場合、停止中の「モデム」の起動を指示する。「タイマ」を開始する。

太字 : 方式用語 (オブジェクト)  
アンダーライン : 操作

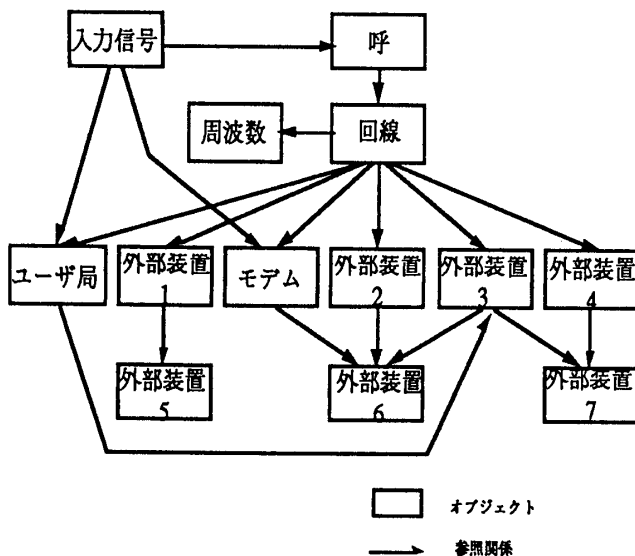


図2.オブジェクトの参照関係

4. オブジェクト抽出の技法

前章で述べた手順は一般的な物であり、オブジェクト抽出のためには、適用経験に基づく技法が必要である。適用過程で得られた技法を述べる。

(1) 操作が属するオブジェクトの決定

システム設計書の1文中に複数の方式用語を含んでいる場合、例えば「AをBにCする」という文があった場合、操作「Cする」がオブジェクト「A」に対する操作か「B」に対する操作かを決定する必要がある。ここでは、A、Bのうち抽象度の高い方のオブジェクトに対する操作とする。ただし、この基準は絶対的なものではなく、後の段階でオブジェクト関連図による全体像の把握を行い、不自然な関連を無くすように調整する。

(2) オブジェクトの統合

類似の概念が別のオブジェクトとして抽出されることがある。この場合、概念的に一つの用語で表わすことのできるものは、本質的に同一のものであると考えられるので、一つのオブジェクトにまとめる。

例えば、呼処理機能で抽出される周波数オブジェクトと、装置監視機能で抽出される周波数オブジェクトは、異なる操作を持つが、同じ概念を表わすものと考えられるので、一つのオブジェクトにまとめる。

(3) 抽象化のレベルの相違

システム全体をサービス機能単位で分割し、複数人で並行して要求分析を行っていくため、同一の情報が異なる見方をするデータストアとして出現することがある。例えば呼処理機能においては「外部装置1」「外部装置2」というように、装置単位にデータストアが作られる。一方、装置監視機能では、「装置故障状態」といったデータストアが作られる。これらは操作を分割していき、共通のデータストアに合わせる。例えば、「装置故障状態」というデータストアに対して、「外部装置1の故障状態を確認する」「外部装置2の状態を故障状態から通常状態に変える」といった操作が考えられる場合、このデータストアは、実は「外部装置1の故障状態」と「外部装置2の故障状態」の2つの要素を持つことがわかる。その後、上記(2)の手法により、オブジェクトの統合を行う。

5. まとめ

SAを利用したオブジェクト抽出の技法を提案した。SAは下位オブジェクトの抽出には有効であり、Boochの方法のようにトップダウンにオブジェクト抽出を行う方法に比べ、オブジェクト抽出の見通しが良いという利点がある。

参考文献

[1]Booch G, "Software Engineering with Ada", Benjamin/Cummings, 1983  
 [2]Sally Shlaer, Stephen J.Mellor, "Object-Oriented Systems Analysis", Yourdon Press, 1988  
 [3]Tom Demarco, "Structured Analysis and System Specification", YOURDON, 1989.  
 高梨智弘、黒田純一郎 監訳、「構造化分析とシステム仕様」、日経マグロウヒル社