

シーケンスチャートからのオブジェクト抽出に関する一考察*

1T-11

堀川 桂太郎 小林 透†
NTT ソフトウェア研究所‡

1 はじめに

複数の組織によって遂行される協同作業の支援システムを、オブジェクト指向モデリング [1][2] にて分析する方法を検討している。ここでは、システム化の初期段階において効果的に業務の流れを表現する業務フロー図から出発して、オブジェクトを抽出する手法を提案する。その概要は、まず与えられた業務フロー図を正規化し、シーケンスチャートを作成する。次に、各プロセスで最も支配的に利用されるオブジェクトを選択し、それを探索の拠点に関連するオブジェクトを抽出する。「クラスの構築」と「クラス階層の構築」作業はオブジェクト抽出に並行して行なわれる。ここでは、「ファイル供給業務」や「問題管理業務」に代表されるソフトウェア運用業務のシステム化を対象に本手法を試行適用し、以下の観点でその有効性を考察したことを報告する。

- オブジェクト抽出の容易性。
- 分析結果としてのデータモデルの特性。

2 問題管理の業務分析とその正規化

今回は、ソフトウェアの保守運用において重要な位置を占める「問題の管理」業務を分析した。ソフトウェアの問題管理の主要な目的は、以下の3つである。

- 問題点、解析状況、対処情報を迅速に通知する。
- 問題解析者の責任の所在を明らかにする。
- 問題発生状況を迅速に周知し、未解決問題の把握を容易に行なう。

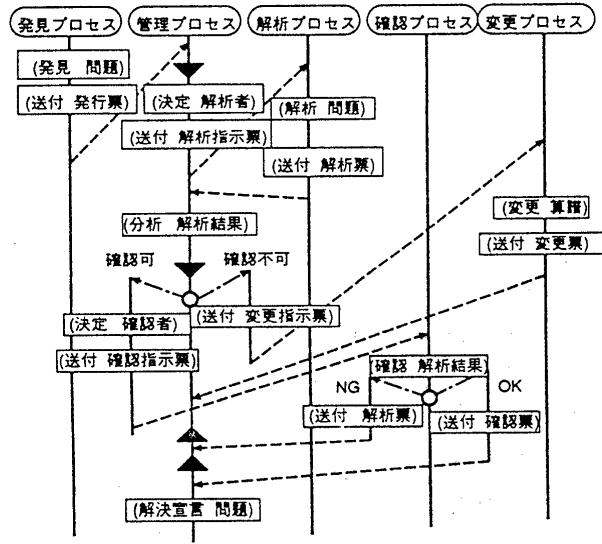


図 1: 問題管理シーケンス

*A Study of Object Extraction from Sequence Charts

†Keitaro HORIKAWA, Toru KOBAYASHI

‡NTT Software Laboratories

与えられた作業要項に基づいて業務分析を行ない、各作業単位を“(act obj)” (何をどうする) という書式で正規化して記述することで、図1に示されるようなシーケンスチャートが得られる。シーケンスチャートは段階的に詳細化され複数存在する。その一例として、発行票送付の詳細を図2に示す。この業務分析と正規化によって、実行主体であるプロセス、各作業単位を構成する対象(obj)と動作(act)、各作業単位の時間的遷移、プロセス間の相互作用などが明確に表記できる。

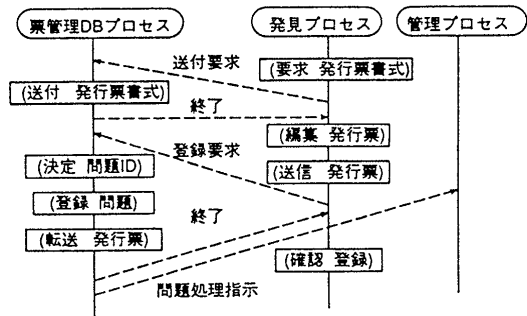


図 2: (送付 発行票) 詳細シーケンス

3 オブジェクト抽出

得られたシーケンスチャートからオブジェクトを抽出する手法を説明する。なお、具体例は次節に示す。

3.1 中心・参照オブジェクトの抽出手順

手順1: 各プロセス p_i ($i \in [1, n], n$: プロセス総数) に

ひとつの中心オブジェクト $Co(p_i)$ を抽出する: ひとつのプロセス p_i に着目し、 p_i が行なう作業単位“(act obj)”を列挙する。その中のobjの集合から以下の判断基準にもっとも適合するものを選択し、中心オブジェクト $Co(p_i)$ とする。

- 基準1: プロセス p_i が最終出力する
- 基準2: p_i が頻繁にアクセスする
- 基準3: 人工物 (artifact) のように構造が比較的自明である
- 基準4: 数値や文字列などのような不可分なオブジェクト=プリミティブオブジェクトではない。

手順2: $Co(p_i)$ を探索の拠点に参照オブジェクト $Ro(Co(p_i))$ を抽出する: 参照オブジェクト $Ro(Co(p_i))$ とは、例えば $Co(p_i)$ を構成する、 $Co(p_i)$ が参照するなど何らかの形で $Co(p_i)$ に関係するものである。これを抽出する為には、詳細化されたシーケンスを参照しながら、以下の判断基準を用いる。

- 基準5: 参照オブジェクト $Ro(Co(p_i))$ は、 p_i だけに関わるものではなく、複数のプロセスから複数個抽出できる。
- 基準6: $Ro(Co(p_i)) \ni Co(p_j) (i \neq j)$ も可能である。

手順3: 中心・参照オブジェクトの反復抽出: 既に抽出済みの参照オブジェクト Ro を、その後のオブジェクト探索における中心オブジェクト Co' とし、上述の手順2を適用し、次段階の参照オブジェクト $Ro'(Co' = Ro)$ を探索する。その参照オ

プロジェクト R_o' がすべてプリミティブオブジェクトになるまで抽出する手順3を繰り返す。

3.2 属性・クラス・クラス階層の定義

全プロセスに関してオブジェクトを抽出するのに並行して以下の分析を行なう。

- **クラス(静的側面)の定義**: 抽出されたオブジェクトの中で、共通あるいは類似した性質を持つものは、同じクラスに属する異なるインスタンスという位置付けでクラスを構築する。
- **クラス階層(静的側面)の定義**: 多くのクラスが列挙された時点で、類似クラスをあつめ、それらの共通部分からなるスーパークラスを設ける。個々のクラスはそれらのサブクラスという関係を設け、複数のクラスに構造的体系をつくる。その時、実世界の自然な関係のうち特に普遍的な is-a 関係に基づくことが有効である [3]。
- **クラスの動的側面(メソッド仕様)の定義**: 各プロセスの各作業単位“(act obj)”を参考にして、act の部分に対応する obj のクラスにおけるメソッド仕様とする。これらを段階的に詳細化しその他のクラスのメソッド仕様も定義してゆく。

4 分析結果の具体例

上述の手順を用いて、中心・参照オブジェクトを抽出した結果の一部を表1に示す。表中の R_o の欄で下線を引いたものは、プリミティブオブジェクトである。

さらに、これらのオブジェクトからクラスとクラス階層を構築した結果を図3に示した。これについては考察で述べる。

表 1: 抽出された中心・参照オブジェクト

p_i	$Co(p_i)$	$Ro(Co(p_i))$
発見	発行票	発行書式, 発見者, 記述規則, 表示規則, 問題の症状, 発行日時, タイトル, 格納 DB, 問題管理者, 送付手順, 問題 ID
管理	指示票	指示書式, 管理者, 記述規則, 表示規則, 指示内容, 指示日時, 対応発行票, 格納 DB, 送付先, 送付手順
解析	解析票	解析書式, 解析者, 記述規則, 表示規則, 解析状況, 解析結果, 解析日時, 対応発行票, 過去の類似問題, 送付手順, 格納 DB, 送付先
確認	確認票	確認書式, 確認者, 記述規則, 表示規則, 確認結果, 確認日時, 対応発行票, 送付手順, 格納 DB, 送付先
変更	変更票	変更書式, 変更者, 記述規則, 表示規則, 変更決定会合, 変更理由, 変更内容, 変更日時, 対応発行票, 変更ソース, 関連仕様, 送付手順, 格納 DB, 送付先

5 考察

5.1 オブジェクト抽出の容易性

提案したオブジェクトの抽出法に対して、以下のよな定性的考察ができる。

- **手順性**: シーケンスチャートの各プロセスからひとつの中心オブジェクトを抽出して、そこから関連するオブジェクトを抽出していくというアプローチをとったことにより、抽出作業に方向性が与えられ、探索アルゴリズム的な手順が得られた。

- **網羅性**: 上記の手順性によって、オブジェクト抽出における「もれ」や「かたより」が減少し網羅的な探索が行なえる。
- **客観性**: それぞれの手順において、中心オブジェクトや参照オブジェクトの判断基準を設けているが客観的または厳密であるとは言えない。

5.2 得られたデータモデルの特性

提案した手法を利用して得られた結果に関して以下の考察が出来る。すなわち、図3に示すように、中心オブジェクトから派生した5つのクラスに対し、スーパークラス「帳票」を設けて、継承や多相性を導入したクラス体系が構築できた。これにより、差分プログラミングやデータとプログラムの間に総称的なインターフェースが実現できる。

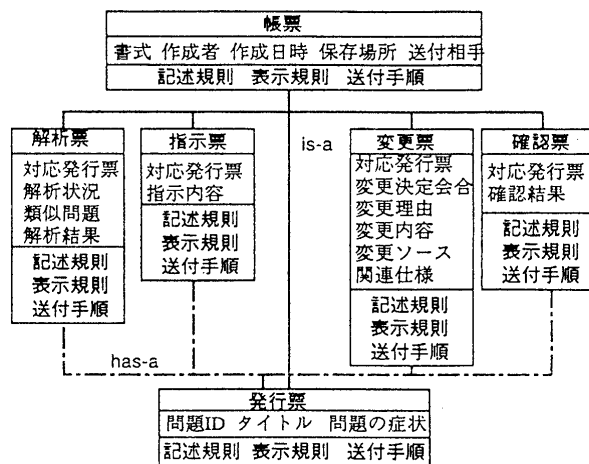


図 3: 問題管理業務におけるクラス階層の一部

6 まとめと今後の課題

シーケンスチャートを基に、今回考案した手順を適用することで、半機械的なオブジェクト抽出とクラスの体系化が行なえた。また、このようにして得られた分析結果は、異なるオブジェクトに対する類似したシーケンスを、共通メソッドとして仕様を統一できることが分かった。

今後の課題として中心・参照オブジェクトという分類結果の再利用と、類似したシーケンス情報からの共通メソッド抽出方法をさらに検討し、分析手順を厳密・多様化する。そのために、ソフトの保守運用におけるその他の業務に対し、今回の検討で用いた手法を適用し、分析手順の洗練と同時に、各業務毎に得られたオブジェクトの体系を統合する。

参考文献

- [1] Rumbaugh, et al, "Object-Oriented Modeling and Design", Prentice Hall, 1991
- [2] Shlaer, Mellor, "オブジェクト指向システム分析", 啓学出版, 1989
- [3] 青木, "オブジェクト指向ソフトウェア開発 (OOA, OOD, OOP)", 富士ゼロックス情報システム, 1990