

ソフトウェアサイクルへのUMLの適用

6Q-3

高山 裕志 古河インフォメーション・テクノロジー株式会社
 和田 憲明 富士通株式会社

1. 研究の背景と目的

システム開発の現場では、オブジェクト指向分析/設計は普及していないと言える。最近脚光を浴びているUMLを導入すれば、普及させることができるのだろうか。我々はUMLについて2つの疑問を持った。

- (1) 通常のシステム開発現場では、オブジェクト指向の経験者と未経験者が混在している。その状況下において、オブジェクト指向分析/設計を採用し、UMLを効果的に適用するためにはどのような支援が必要なのだろうか。ガイドラインを作成することでそれが提示できるのではないかな。
- (2) UMLには、成果物の表記法を統一する利点があるが、既存の表記法を変更してまで適用させるだけの説得材料となるのだろうか。もう一步踏み込み、オブジェクト指向分析/設計をベースとした開発方法論と共に議論することで、より効果的な使用方法が見えてくるのではないだろうか。

我々はLS研（富士通ユーザの研究会）において、「ソフトウェアサイクルへUMLを適用できるかどうか可能性を見極めること」をテーマとして研究を進めた。本稿ではその研究成果の概要を報告する。

2. 研究の成果

今回の研究では、ビジネスアプリケーションへの適用にターゲットを絞り、UML普及の足かせとなっていると判断した「ドキュメント作成方法の不明確さ」を補うための「リファレンス」を作成した。またドキュメント間の関連を明確にすることを目的とした独創的な「ドキュメントマップ」も作成することができた。最終的にこのリファレンスとドキュメントマップは、オブジェクト指向での分析/設計手法の各フェーズでのドキュメントの作成基準や、下流工程へのドキュメントの影響なども取り込んだ開発方法論へと発展させることができた。我々はこれをURCO(UML Rapid Compass)法＝「UML最速誘導法」と命名した。

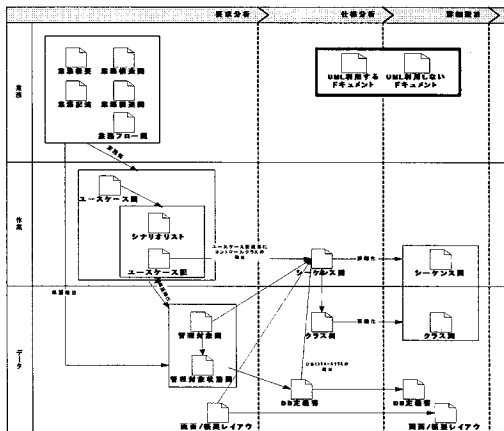


図1 URCO法ドキュメントマップ。ドキュメント間での情報の再利用の流れを線で明確に示している。このマップにより、ドキュメント作成時の参照ドキュメントが常に明らかになり、容易に作成できるよう支援する。

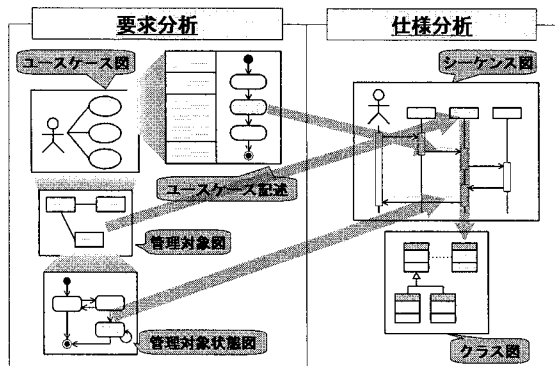


図2 URCO法でのドキュメント構成要素の関連の一部を図解した。リファレンスでは、このように再利用の手順を具体的に示し、ユースケース図、管理対象図などの上流ドキュメントからシーケンス図、クラス図などの下流ドキュメントを、ある程度機械的に作成できるように支援して、オブジェクト指向の未経験者でも設計作業を進めることができる。

3. そして実践

我々はURCO法で定めたりファレンスに沿って実際に分析/設計を実践した。適用したシステムはケーブル敷設工事のための受発注管理システムであり、Java Servlet を利用した Web アプリケーションである。

本事例では、上流工程のドキュメントからシーケンス図を作成するまでの手順を紹介する。

上流工程のドキュメントとしてユースケース図、ユースケース記述(アクティビティ図)、管理対象図、状態図、画面・帳票レイアウトなどを作成する(図3~5)。

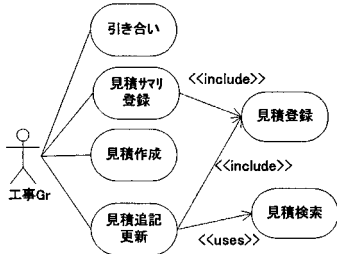


図3 ユースケース図(工事見積業務)

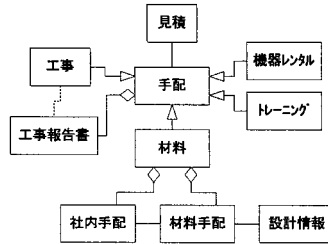


図4 管理対象図

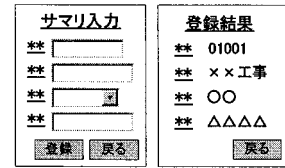


図5 画面レイアウト

これらの作成された上流工程のドキュメントからURCO法にしたがってシーケンス図を導出する。その手順は大きく分けてクラス(オブジェクト)の抽出とクラス間のメッセージの抽出から構成されている。

クラスの抽出は処理の流れを制御するコントローラ、画面や帳票などをビュー、データに対応するドメインとして分類する。見積サマリ入力機能におけるクラスの抽出例を表1に示す。

クラス	抽出方法	クラス抽出例
コントローラ	1 ユースケース1 コントローラ	見積サマリ入力、見積登録
ビュー	画面・帳票レイアウトから抽出	サマリ入力、登録結果
ドメイン	管理対象図から抽出	見積

JAVA Servlet を用いた WEB アプリケーションではコントローラとビューの分離が困難であるため、実装時にはそれらをセットとしてクラスを定義している

メッセージの抽出は状態図やユースケース記述(アクティビティ図)から利用する。イベントやアクティビティを各クラス間のメッセージとして抽出する。アクティビティ図の例を図6に示す。

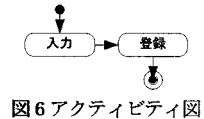


図6 アクティビティ図

このようにして作成されたシーケンス図を図7に示す。表1で示したクラスの手順によって業務に関わるクラスは機械的に抽出できた。また、メッセージについてもアクティビティ図から作業情報(「入力」「登録」)や状態図から遷移情報(「生成」「登録」)の内容を反映させ、コントローラからの画面表示やデータベースへの登録の手順をフレームワーク化することにより、容易に抽出できることを確認した。

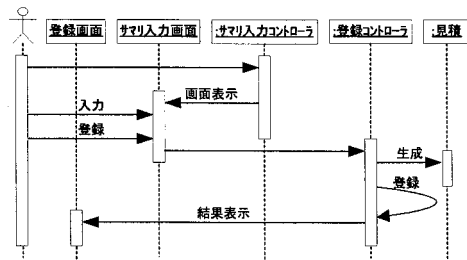


図7 シーケンス図 (見積サマリ登録)

4. おわりに

UMLは、開発方法論と組み合わせることによって、その利点を最大限に発揮できることが実践により明確になった。この成果は、迷える開発者達への一筋の光明となるに違いない。我々は、オブジェクト指向設計の普及と発展を推進する当事者として今後も努力していきたい。