

OO'99 シンポジウム・モデリングワークショップ報告

羽生田栄一 †

オブジェクト指向'99シンポジウムの一環として1999年7月22日に行われたモデリングワークショップの実施報告を行う。「販売管理システム」例題を対象にして、4つのOO開発技法による設計の前段階までのモデリングを比較するという試みを行った。結果として、現在の最新OO開発技法の技術的・実践的な到達点を具体的に把握できた。今回はBOFミーティングによる議論の時間の補填という新しい試みも行った。

A Report on a Modeling Workshop at OO'99 Symposium

Hanyuda Eiiti, OGIS-RI Co., Ltd., Object Technology Center

A modeling workshop was held at July 22th, 1999 as a session of OO'99 symposium. We used the reference problem "Sales management system" as a modeling target. We selected 4 typical OO development methods in Japan and compared their models, artifacts and processes. We have found the essence of current OO methods and also their typical problems through discussions with the modelers.

1. はじめに

オブジェクト指向'99シンポジウムの一環として1999年7月22日に行われたモデリングワークショップの実施報告を行う。なお、この報告では、酒匂氏の事前レジュメおよび千葉氏の議事メモを一部利用した。

1.1. 開催の経緯

オブジェクト指向シンポジウムにおいてモデリングワークショップは恒例となり、楽しみにしている参加者も多く存在し、定着してきた感がある。今年のワークショップでは、去年と同じく、新潟工科大学の青山先生の作成された「販売管理システム」例題[1]を流用させていただいたが、去年とは違った目的を設定した。去年は、SLagoonの中谷さんがモデラーとなって、特にユースケースによる要求定義およびそこから分析クラスの導出プロセスをステップを追って検討し、ユースケース分析と分析モデル作成の様子を段階的にトレースしていった。

せっかくの現実に近い設定の例題でもあり、できればこの例題をオブジェクト指向開発プロセスを客観的に議論する際の標準問題の1つとして育てていければ、それはオブジェクト指向シンポジウムからソフトウェア産業界への具体的な貢献となるのではないかという淡い期待もあって、継続して同一例題を利用してみようと考えた。

今回は、現在、日本で利用しうる代表的なオブジェクト開発技法を4つほど取り上げ、それらの技法をある程度の規模と複雑さを備えた現実的な同一の例題に

対して適用し、そのモデリング過程を、段階的な中間成果物およびそれを得るに至ったガイドラインとともに比較検討しようというものである。今回は、特に、要求定義と分析の段階におけるモデリングを中心に、各技法の特徴と利用モデルおよびプロセスを比較することを目的とした。

たまたま声を掛けて快く参加を了解いただいた以下の4チームがそれぞれ異なるオブジェクト開発技法をもって同一例題のモデリングに臨んだ(順不同)。

チーム	モデリング手法
日立	コンポーネント指向業務設計技法[2]
NEC	NEC開発手法[3]
富士通研	OO-Design2[6]
OGIS	Unified Process[4][5]OGISカスタマイズ版

とはいえ、この例題のボリュームでは当日その場でのいうわけにはいかないので、事前に分析モデルまでのすべての成果物を作成して頂いた。

1.2. 当日の様子

ワークショップ当日は、以下のような役割分担でワークショップを進行した。司会以外に、外野の視点からいろいろな突っ込みやコメント、アドバイスを入れるコメントータという役割を去年同様、酒匂氏にお願いし、会場の参加者の代表という立場と、各チームに開発をお願いしているクライアントという立場を模倣的に演じてもらった。

役割	担当(敬称略,*は当日の回答者)
問題提供	青山幹雄
司会	羽生田栄一
コメントータ	酒匂寛

†株式会社オージス総研オブジェクト技術ソリューション事業部開発技術コンサルティング室

書記	千葉寛之, 羽生田栄一
日立チーム	米田豊満*
NEC チーム	石井慎一郎, 佐野建樹*, 橋本祐介
富士通研チーム	山本里枝子*, 上原忠弘
OGIS チーム	重見剛*, 鷺島淳一, 林亨

各チームに順番に、

1. 利用した OO 開発技法
2. 業務ドメインおよび要求に対するモデリング
3. システムの仕様に関してのモデリング
4. 設計へマッピング

の各テーマで報告と質疑を行っていき、適宜、司会、コメンテータ(神の声ともいう場合があります)および会場からの質問・アドバイス・批評にも応じる、という形式で進行した。

去年、会場とのディスカッションに時間がとれず、中途半端に終わってしまったという反省から、今年は新しい試みとして、正規のワークショップ時間帯の後に、希望者のみ残って、夜の部として BOF (Birds Of Feather) ミーティングを行うことになった。結果として 18:30 から 21:20 まで議論は続き、とりあえず要求分析から設計前段階までの開発プロセスに関して予定していたテーマでの質疑はひとつおりのり行うことができた。おおよそその参加者(聴衆)は以下の通り。

- | | |
|--------------|---------|
| 14:30 開始 | 60-80 人 |
| 18:00 公式終了 | 50 人 |
| 18:30 BOF 開始 | 25 人 |
| 21:20 BOF 終了 | 20 人 |

ただテーマのせい参加者はアカデミズムからは少数で産業界からが大半で、せつかくの実モデルを前にしての対話が不発に終わったのが残念である。

1.3. 用いた例題について

今年のワークショップでも昨年と同じく、青山先生の作成された「販売管理システム」の例題を利用してモデリングを適用することにした。例題の内容については、[1]を参照して欲しい。作業の際に例題に対しいくつかの不明点・疑問が指摘された[1]が、昨年の成果物との比較容易性や各チームでの解釈や技法による特徴を反映しやすくするという意味で、例題の記述自体はそのまま利用することにし、要求の独自解釈・仕様の独自決定に関しては記録を残している。

また、例題の提示の仕方として最初から具体的なシステムイメージを固定してしまわず、現実の業務ヒアリングやくり返し型開発における段階的な要求と仕様の確定という実際的な状況を考慮して、マスターファイル

仕様や本来個々の部署で異なる業務ロジックのレベルの記述は第 2 段階で提示することにし、最初の問題提示では業務の概念レベルの説明にしておいてはどうか、という意見もあった。それは今後の課題として残り、今回は、複数チームということで作業時間的にも制約が厳しく、去年と課題を揃えてとりあえず複数の OO 技法でのワンショットのモデリング過程と成果物を比較し OO 技法の現状と課題を洗い出す、という点を第一目標にしようということで作業を進めた。

1.4. ワークショップで議論に利用された成果物

今年のワークショップの準備として、以下に掲載する成果物を各チームに作成していただいた。各チームともに、以下の 4 つのパッケージから構成される。

目的	文書
技法について	開発技法説明資料
モデリング成果物	ドメイン・業務・要求に対するモデル
	システム仕様モデル(分析モデル)
	設計に移行するための情報
その他	モデリングプロセス説明資料

フルセットのモデル成果物は膨大な量となるのが想定されるのと、モデリングワークショップの目的はフルセットの最終成果物をプレゼンすることが目的ではないので、今回のワークショップ当日のディスカッションを進めるに当たって必要なものを適宜選択してコンパイルした。

なお、今回の成果物を作成にするあたり要した工数は、各チームそれぞれ、延べ約 1-3 人週である。

1.5. 今回のワークショップの狙い

今回のワークショップでは、これら 4 チームが各々作成した成果物を持ち寄ってパネル形式で議論を行った。おおきく以下の 4 カテゴリ

- (1) 利用した OO 開発技法の特徴の説明
- (2) 業務ドメインおよび要求に対するモデリングの手順と成果の説明
- (3) システムの仕様に関してのモデリングの手順と成果の説明
- (4) 設計へマッピングするための情報・指針の説明のテーマについて取り上げた。

特に、テーマ 2 では、業務分析やユースケースの扱い方・記述法、テーマ 3 では、クラスの識別を含めたオブジェクトモデリングの際のノウハウおよび業務ロジックの表現、テーマ 4 では、そうして生成された分析モデリング成果物を設計にどう繋げていくかについての各技法の基本スタンス、を議論した。また、全般に、モデ

表1. ワークショップで取り上げた4つの OO 開発技法の比較

モデリング上のポイント	各チームで用いた技法の特徴			
	日立チーム技法	NEC チーム技法	富士通研チーム技法	OGIS チーム技法
技法のキーワード	コンポーネント指向	データ中心, 画面指向 チーム指向	パターン指向, シナリオ駆動, リスク駆動	リスク駆動繰り返し, ユースケース駆動, アーキテクチャ重視
技法の特徴	業務改革の観点, シナリオからメソッド抽出, 分散オブジェクト環境を想定したコンポーネント概念, 独自図表にもとづく徹底した標準化・部品化	データ中心技法を併用し OOA の障壁を除く, 設計はアーキテクチャ専門部隊, 実装は極力業務固有部のみ」フレームワークの構築と利用	作業・文書・管理を現場向けに具体化, パターンとして開発ノウハウを定義運用, リスク駆動で繰り返し開発を手順化, UML とユースケース利用の標準化	体系性・UML 準拠, リスクとユースケースにもとづく管理された繰り返し開発, 実行可能アーキテクチャによるモデル推敲, UML ツールとスクリプトの徹底利用
開発スタイル	改良ウォーターフォール	改良ウォーターフォール (繰り返しも可能)	繰り返し型	繰り返し型
適用ライフサイクル	ほぼ全体	ほぼ全体	ほぼ全体	ほぼ全体
プロセスの詳細さ	詳細な手順あり	おおまか	詳細な手順あり	おおまか(カスタマイズ時に指定可能)
実開発の適用実績	あり	あり(OO'98[3]参照)	あり	あり(多数)
想定適用分野	ビジネスアプリ	ビジネスアプリ	ビジネスアプリ	ほぼすべて(IT から制御系まで実績あり)

ルの整合性・妥当性に対する考え方、要求とモデルとシステムの間の特異性に関する位置付け、部品化や再利用性に対する取り組み、チーム構成と提案技法の関係なども興味深い話題として取り上げた。

1.6. コメンテータの視点

ちなみにコメンテータ(当人は野次馬と自称していましたが)からは、以下のような観点から今回のワークショップを眺めたいと事前の表明があった:

- (1) 開発ガイドライン作成のためにどのような試行、情報収集、分析、評価を行ったか
- (2) 表記法としての UML に何を加えたか、除いたか
- (3) 開発プロセスを駆動する原理 (Waterfall, Incremental, Iterative, ... etc) は何か
- (4) 開発プロセスを駆動する単位 (UseCase, Scenario, Class, ... etc) は何か
- (5) チーム構成の原理 (専門家型、役割分担型、... etc) は何か
- (6) どのようなマトリックスが想定されているか
- (7) 問題を解く際に動員した知識は何か
- (8) 利用した具体的なツール

これらの項目に関し、各チームともあらかじめ準備をある程度していたと思われる。

2. 各チームの開発技法の特徴

2.1. OO 方法論の代表例4技法の特徴

これについては、概要のみ表1にまとめた。日立は、独自の図表による徹底した標準化・文書化、NEC は、GUI 指向のデータ中心技法と OO の組み合わせとフレームワーク構築による生産性、富士通研は先行チームによるパターン定義の後攻への申し送り方式、OGIS は UML にもとづくオーソドックスなユースケース駆動の開発プロセスのツールとスクリプトによる現実化、等に特徴があるといえる。詳しくは、[1][2][3][4][5][6]を参照して欲しい。

どの技法も業務モデリングからシステム設計・実装までほぼすべてのライフサイクルをカバーしており、なおかつそれなりの実開発への適用事例が複数存在する、という意味で現在日本で入手可能な代表的 OO 開発技法といえると思われる。

2.2. OO プロジェクトと OO 方法論に関する視点

今年のワークショップの特徴として、4つの異なる開発技法をベースにしたモデリングプロセスが比較できるという点が挙げられる。しかし、ここでの興味は、各技法の細かな性格づけというよりも、各技法が実際の開発プロセスにおいて、どの程度現場の各要員の実

作業に対して駆動力・制約・具体的な指針として機能しているか、逆にいうと各チームで各技法の内容を超えてどの程度オリジナルに考えねばならないことがあったか、という点に関心があった。ここでは、コメントータが挙げた以下のポイントを中心に各チームに確認を行った上で、現状実現できていない部分がどこか、現場からの声を踏まえて課題を議論した。

(1) 技術移転性

導入容易性:原理が理解しやすい、教材が揃っている、人を育てやすい。

駆動原理の形式性:作業を進めて行く際の成果物や、各作業工程の形式的な定義と、意味定義、判断基準などが提示されている

(2) 予測可能性

再現性:同じ問題領域の開発作業ならば、同じような作業と

工数、工期予測性:同じ問題領域の開発作業ならば、どの程度の開発期間と工数が必要かがある程度予測できる。

(3) 変更可能性

プロジェクトからのフィードバックでプロセスが修正可能。その効果も定性定量的に評価可能

2.3. ディスカッションと課題

日立、NEC はじめどの技法も実開発の中で使いこみながら洗練させてきたという経緯をもち、それなりに実践的である。たとえば富士通の OO-Design2 は今の姿になるまでに3年、それまでに3度技法自体を改版しているし、他も同様である。技法とはプロセス自体のフレームワークであるから、Ralph Johnson いうように最低3回の見直しというのは必要なことなのであろう。また他3技法はビジネスアプリが対象だが、OGIS技法は、制御系や組込み系への適用実績も存在し興味深い。手順やモデリング上のどこかでビジネス系と明かな違いがあるか突っ込めなかった。

どのチームの技法も、どのタイミングでどんなダイアグラムや文書を中間成果物として残さねばならないか、という点は明確に示している。しかし、モデル内容について踏み込んでガイドしているかについては疑問が残った。やはり、その部分は、属人性や能力・経験の差が現れてしまう。その意味で、NEC や富士通研のようにあらかじめ難しい作業工程やモデル作成時にはそれなりの特別な要員を割当て、さらにその結果を他のメンバーにテンプレートとして使わせるように開発技法の中にあらかじめ「パターン作成プロセス」自体を組

み込んでおく、というのは注目される。

ただし、プロジェクトとして個別にでなく、マトリクスを開発技法の中で明示的に採集・利用し、プロジェクト管理、モデルやプロセスの改善に役立てるというまでに各チームともいっていない点、今後の課題である。

また、記法としての UML および要求モデリング手法としてのユースケース/シナリオ概念は広く普及しつつあることが感得された。現在 UML を使っていない技法も次版での採用を示唆していた。

3. 問題ドメインおよび要求のモデリングに関する技法の特徴と課題

3.1. モデリング結果の特徴

基本的に各チームともに、いわゆる分析モデルを作成する前に、業務ドメインの理解を行う、というフェーズなり心構えを導入しているのが注目される。通常のオブジェクト指向の教科書的な記述では、最初にユースケース図を描いてシステムの基本サービス(いわゆる基本機能)を識別するところがトップにくるのだが、現実の分析はそんなに甘くないということだろう。どのチームも必ず、業務組織の構造やワークフローのスケッチ、業務ドメインの概念モデルといった、通常のシステム分析モデルよりも上流の作業をラフな形であれ導入しているのである。これによって、システム対象を明確化するとともに業務ドメインに対する第0次近似を与えて、以後のより厳密なモデル化の方向付けをしているものと理解できる。

そして、要求モデリングの中心は、現在の OO 技法においてはユースケース分析が占める。それを反映し、4 つ中 3 技法ではユースケース相当の記述が要求モデルとして行われていた。

3.2. ディスカッションと課題

昨年の OO'98 のモデリングワークショップでは、ユースケースに関して、その存在が「機能」に密着し過ぎており、オブジェクト指向に反するのではないかと、という点で大いに議論が行われた。そこで明らかになったのは、アカデミズムと産業界でのユースケースの実用性に関する認識のズレである。確かに、ユースケースとオブジェクトモデルの間の論理的な整合性とトレーサビリティに関し厳密に議論しきれない部分が残っているのは事実だが、産業界の現場では、ユースケースをモデルの要素としてだけではなく繰り返し開発やテストケースを制御するための単位としてより広い観点から捉えており、その意味でユースケースの実用性は疑いのないものだとして理解している。そうした認識が、今回

表2. 4つの技法のフェイズごとの主要なモデリング成果物セット

モデリングの目的	今回の各チームのモデリング成果物			
	日立チーム技法	NECチーム技法	富士通研チーム技法	OGISチーム技法
ドメイン分析	業務活動体系図 管理実体系図 業務活動/管理実体マトリクス, 業務活動/組織関連マトリクス	対象領域図 業務階層図 業務フロー図	業務分析書(業務構造, 管理対象, 業務関連)・業務用語辞書	業務フロー(アクティビティ図) 概念モデル(クラス図)用語辞書
アーキテクチャ分析	アプリケーションコンポーネント図	(3層アーキテクチャベース)	システム構成仕様書	アーキテクチャ文書(パッケージ図, サブシステム間相互作用図, 配置図), リスク一覧
業務分析	業務ルール定義書 業務処理バリエーション定義書, 業務動作定義書(操作シナリオ), オペレーション定義書(画面帳票スケッチ)	画面レイアウト 画面処理仕様書	ユースケース分析書(ユースケース図, ユースケース依存関係図, ユースケース記述)	ユースケース図 ユースケース記述シナリオ
システム分析	拡張シーケンス図 オブジェクトモデル	クラス図 シーケンス図	画面イメージ図 シーケンス図, クラス図 分析パターン定義書(クラス図, シーケンス図)	クラス図 相互作用図 ステートチャート図
オブジェクト仕様	ビジネスコンポーネント定義書	クラス仕様書 メソッド仕様書	クラス定義書 データ辞書	クラス仕様書
アーキテクチャ設計	システム方式設計	3層アーキテクチャ プレゼン層制御 アプリ層制御	方式設計作業(システム設計書 コンストラクション計画, 設計作業規約)	アーキテクチャモデル(クラス図, 相互作用図, 配置図) 実行可能コード
その後の作業	構造モデル設計 実装モデル設計	実装アーキテクチャ構築, アプリケーション開発 環境構築, アプリケーション開発	詳細設計作業 実装作業 テスト作業	上記詳細化 テスト仕様書

の4つの技法の中での位置付けの重要性から再確認できたといえる。

ただし、ユースケースもいいことばかりではなく、ユースケース識別の妥当性特に機能分割との差異、ユースケース記述の粒度、ユースケース記述の構造化、ユースケースになじまない非機能的要求の捕捉、そして上でも議論になったユースケースとオブジェクトモデルの対応関係等課題は残る。しかし、これらについても、各チームでは、あらかじめ概念モデルを用意することでユースケース記述の精度を揃えたり、記述テンプレートを表化して用意したり、Unified Processのようにユースケースをコラボレーションを介してそのロールとしてのオブジェクトと対応付けたりと、さまざまな工夫が伺えた。しかし、システムの基本サービスの把握という目的でのユースケース図は機能分割に陥りやすく使いづらいという認識もあるようで、日立、NECなどはシーケンス図作成と対にしてのみ考えており、要件機能の管理は別途行っていた点興味深い。

4. システム仕様のモデリングに関する技法

の特徴と課題

4.1. モデリング結果の特徴

この部分での各チームのモデリング結果はどれも似たようなものになった。それだけ、OOでの分析モデリングというものも一定レベルの実務者が行えば近いモデルになるだけの成熟を示しているともいえる。

その際、一番大きく貢献しているのが、オブジェクトの分類とそれにもとづく論理的なパッケージング(論理アーキテクチャ)の導入だろうと思われる。日立では、プロセス系(一種の制御オブジェクト)/データ系/トランザクション系の3つに、他のチームもUnified Processで設定されているようなバウンダリ/コントロール/ドメインというオブジェクトのカテゴリ分類のバリエーションを利用していた。そのため、オブジェクトの識別と各オブジェクトの責務・役割の設定に一定の枠がはめられ、その分モデル構成が安定するという効果が見られた。

4.2. ディスカッションと課題

会場より、システムのモデル化においてオブジェクトを、プロセス系、データ系などに分割するのはオブジェクト指向のカプセル化に反するのでは?という質問

が投げられたが、その場では深い議論にならなかったのが残念だった。これは、オブジェクト指向システムとは何かという問題と、オブジェクトへの適切な責務の分配に関する指針の不在という問題の提起と思われる。実践的には 4.1 で述べたような効用が得られている点、分析を設計に繋ぐ一種の論理アーキテクチャの構成要素としてこれらのカテゴリは説明できるだろう。

ここではさらに再利用コンポーネントの識別や、クラス仕様の定義に関して議論があったが省略する。

5. 分析モデルを設計に移行する際の課題

5.1. モデリング結果の特徴

アーキテクチャ設計は、複雑なシステム開発工程を見とおしく分割制御するために欠かせない。その点、今回、興味深く感じたのは、いくつかのチームで、要求モデリングと並行して、非常にラフなレベルながらもアーキテクチャの検討を行うプロセスを走らせていることである。特に Unified Process ではこれを「アーキテクチャ分析」と呼んでシステム分析と並立している。

5.2. ディスカッションと課題

実際問題として、短期サイクルの繰り返し開発を行うためには、分析の最中にちょっとズルをして設計上の課題やリスクも頭の片隅に入れつつモデリングを進めるのがコツ、ということの現われなのだろう。

当日は、データベース設計のタイミングや RDB を想定した場合の具体的なオブジェクト設計ノウハウ、トランザクション設計ポリシーの検討タイミング等について突っ込んだ議論がなされた。再利用の単位やプロセス・知識の再利用についても質疑がなされた。

できれば設計・実装まで含めたモデリングを前提として、分析モデルと設計モデルの相互作用、アーキテクチャ設計のノウハウを分析モデル・設計モデルの使用情報とどううまくリンクしていくか、等の具体的議論が今後望まれる。

6. 今後に向けて

今回の結果を踏まえて、今後もモデリングワークショップは継続していく意義があると考ええる。その場合、次のような点を考慮していただきたいと希望する。

1. 今回は、モデルの内容を細部までレビューしながら、開発技法や開発プロセスにおける手順や指針とその産出物としてのモデルの内容の対応を検討する余裕がなかった。その理由の大半は 4 チームも同時に取り上げたということに尽きるが、やはり残念である。BOF 等を利用し特チームに

関し、別途モデルレビューを公開で実施したい。

2. 今回は設計前段階までのモデリングで中座している。次回は、設計以後のモデリングを実装まで一部含めて議論できると面白い。ただし、その場合は、モデルは 1 系統に絞る必要があるだろう。
3. さらに、分析・設計・実装のインクリメンタルなプロセス、中途での仕様変更への対応プロセス等をマトリクスも含めて実感できるような提示の仕方が考案できるとインパクトがあるかもしれない。
4. 実際に聴衆を含めた参加者が手と頭をその場で動かす本来のワークショップ形式を模索してみるのも 1 つの冒険として考えられる。
5. ビジネスアプリ以外の、たとえば制御系や組込み系でのモデリング特性の検討もしてみたい。
6. パネル形式で一度、ユースケースの位置付け、オブジェクト分類の問題、分析と設計の対応問題等、「方法論」の概念問題を総括すべき。

ただし工数と負荷を考えると多くは望めないのだが

上記とは別の教育・普及上の問題だが、現在、日本で独自に現実的な問題を対象としてオブジェクト指向モデリング過程が具体的に例示された実用的な教材がなさに等しい。もし今回の各技法の成果物をうまくテキストにまとめられれば有意義であろうと考える。

参考文献

- [1] オブジェクト指向'99 シンポジウム モデリングワークショップ資料, 1999
- [2] コンポーネント指向業務設計技法 HIPACE / AGORA の開発と改良(日立), 団野他, オブジェクト指向'99 シンポジウム 論文集, 情報処理学会, 1999
- [3] パネル: 開発事例に学ぶ 流通業店舗システムへのオブジェクト指向の適用(NEC), オブジェクト指向'98 シンポジウム資料集, 1998
- [4] The Unified Software Development Process, Ivar Jacobson, Addison-Wesley, 1998
- [5] The Rational Unified Process An Introduction, Philippe Kruchten, Addison-Wesley, 1998
- [6] UML によるオブジェクト指向開発実践ガイド, 吉田他, 技術評論社, 1999