

エンドユーザ主導型アプリケーション開発技法における パターンを用いたドメインモデル構築支援

石 樽 久 嗣[†] 中 所 武 司[†]

オフィスなどにおける比較的小規模な業務アプリケーションをエンドユーザが自ら開発し、自ら保守するためのアプリケーション開発環境 M-base の実現を目指している。エンドユーザは、ビジュアルツールを用いてメッセージフローに基づく業務モデルの構築を行い、シミュレーション実行により動作確認を行う。本稿では、共通問題を M-base のアプローチに適用し、エンドユーザによる要求仕様定義プロセスについて検討する。

Enduser-Initiative Requirement Definition Process

HISASHI ISHIGURE[†] and TAKESHI CHUSHO[†]

One solution for enduser-initiative application development is given as a formula of “a domain model \equiv a computation model”. An end-user builds a domain model by using a visual tool, and then confirms the system behavior by simulation of the model. This paper describes requirement definition process for enduser-initiative application development.

1. はじめに

業務の専門家が自分に必要なアプリケーションを自ら開発し自ら保守できるような開発技法と、それを支援するための開発環境 M-base の実現を目指している¹⁾。これまでに我々は、M-base の研究アプローチを共通問題²⁾に適用し、エンドユーザによる要求仕様定義プロセスについて検討してきた。

本稿では、近年活発に議論されているパターン技術³⁾を M-base に適用することで、エンドユーザのドメインモデル構築を支援する方法について述べる。

2. M-base における要求仕様定義プロセス

M-base におけるアプリケーション開発は、次のような手順で行われる。

- (1) 業務仕様の詳細化
システム利用者を抽出し、そこから開始される機能概要を記述することで業務を詳細化する。
- (2) ドメインモデルの作成
モデリングツールを用いて「1業務1オブジェクト」を原則とした業務モデルを構築する。
- (3) ユーザインタフェースの構築
入出力情報を定義することにより UI を自動生

- 成し、遷移図による要求の洗練・検証を行う。
- (4) シミュレーション検証
3種類の図式表現を併用することにより、作成したドメインモデルの妥当性を検証する。

3. 共通問題の適用結果

共通問題記述の 11 項目のうち、前半の 5 項目である (a)~(e) に注目して適用実験を行った。まずプログラム委員長が行う業務として、スケジュール決定、CFP 作成・配布、プログラム委員選出、投稿論文管理という 4 つのユースケースを抽出して対象業務を詳細化し、そこから「1業務1オブジェクト」の割当て原則に従ったドメインモデルを構築した。さらにモデリングとシミュレーションを繰り返し行った結果、最終的には図 1 のようなドメインモデルが作成された。

4. パターンを用いたモデル構築

4.1 パターンの抽出

我々は、図 1 に示した共通問題のドメインモデルから、以下のような 5 つのパターンを抽出した。

- データ管理パターン
データを管理する必要がある場合、それを管理するための専用のコンポーネントを導入し、そこに登録・削除・照会・変更などのメソッドを導入する。
- 分岐パターン
業務フローに分岐処理が生じる場合、分岐を行う

[†] 明治大学大学院理工学研究科基礎理工学専攻情報科学系
Graduate School of Science and Technology, Major in
Sciences, Computer Science Course, Meiji University

コンポーネントにルールを記述する．ここでは，内部変数から分岐させる方法と，UI による問い合わせから分岐させる方法がある．

- 受付パターン
書類や申込書を扱うような業務アプリケーションの場合，そのデータを入力するための専用の受付コンポーネントを導入する．
- 繰返しパターン
ある業務を何度も繰返し実行したい場合，繰返しを制御するコンポーネントを導入し，そこに繰返し条件となるルールを定義する．
- 並行パターン
複数の業務を並行に実行したい場合，複数のメッセージを同時に送信するためのルールを記述する．さらに必要であれば，同期をとるための制御コンポーネントを導入する．

4.2 パターンの分類と活用方法

M-base では，ドメインモデルを構築するという観点から，パターンを以下のような3つに分類した．

- コンポーネント抽出に関するパターン
- メッセージパッシングに関するパターン
- コンポーネント接続に関するパターン

コンポーネント抽出に関するパターンは，詳細化した業務仕様からどのようにコンポーネントを抽出すれば良いかという指針である．これらのパターンにより「1業務1オブジェクト」の割当て原則に従った適切な粒度でコンポーネントを抽出するが可能となる．前節で抽出したパターンの中では，データ管理パターン，受付パターンが分類される．

メッセージパッシングに関するパターンは，業務フローの実行順序の制御に関するものである．ここでは主にフローを制御するためのコンポーネントの抽出やルールの記述方法などのノウハウが中心となる．前節で抽出したパターンの中では，分岐パターン，繰返しパターン，並行パターンが分類される．

コンポーネント接続に関するパターンは，1つのサービスを提供するコンポーネント同士を接続することにより，複合的なサービスを定義するためのパターンである．例題では，データ管理コンポーネントとメール送信コンポーネントを組み合わせることで登録通知サービスを実現している．また，受付パターンとデータ管理パターンを組み合わせることで，2層システムとみなしたシステムを構築することもできる．

5. パターンを用いた適用事例

今回抽出したパターンを共通問題記述の (f)~(h) に

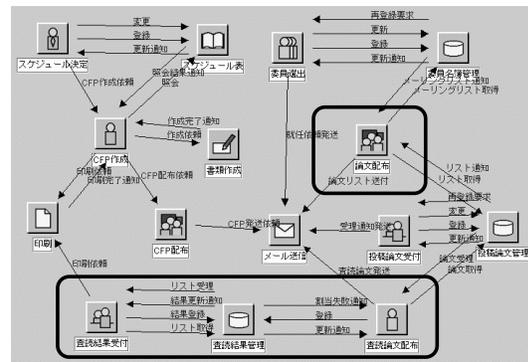


図 1 共通問題のドメインモデル(枠内は新規部分)

適用した結果について以下に示す．

- (1) データ管理パターンの適用により，査読結果を管理するための「査読結果管理」コンポーネントを導入した．
- (2) 受付パターンの適用により，査読結果を入力するための「査読結果受付」コンポーネントを導入した．
- (3) 繰返しパターンの適用により，査読結果を連続して入力できるようにした．
- (4) 分岐パターンの適用により，査読論文を割り当てるための条件をルールとして記述することで論文割り当ての妥当性検証を実現した．
- (5) 「査読結果管理」と「メール送信」を組み合わせることにより，査読結果を投稿者に通知するサービスを実現した．

以上から，M-base においてもパターンを用いることにより，エンドユーザの要求をスムーズにドメインモデルで表現することが可能となった．

6. おわりに

本稿では，エンドユーザ主導型アプリケーション開発技法において，パターンを用いることによりエンドユーザのモデル構築を支援する方法について述べた．

参考文献

- 1) 石樽久嗣, 紺田直幸, 中所武司: メッセージフローモデルに基づくエンドユーザ主導型アプリケーション構築・検証技法, オブジェクト指向 2000 シンポジウム論文集, pp. 133-140 (2000).
- 2) 大西淳: 情報処理学会要求工学ワークショップ共通問題: 国際会議のプログラム委員長の業務, <http://www.selab.cs.ritsumei.ac.jp/~ohnishi/RE/problem.html>
- 3) 中谷多哉子, 青山幹雄, 佐藤啓太 編: ソフトウェアパターン, bit 別冊, 共立出版 (1999).