

倉持和彦[▼] 三浦昭浩[▼] 浦井哲哉[†] 近棟稔[▼]熊井秀憲[▼] 森俊治[▼] 上野浩一郎[▼] 高杉秀樹[▼][▼]三菱電機 (株) [†]三菱電機システムウェア (株)

1. はじめに

オブジェクト指向技術の普及に伴い、アプリケーション・フレームワーク (以下 FW) を利用したアプリケーション (以下 AP) 開発が広まっている。しかし、FW 型ソフトウェア開発を大規模に展開する場合、FW 利用者の教育に時間がかかる、FW の利用方法が不均質なため FW の導入が期待された効果を生まない等の問題が出てきている。

我々は、FW 型ソフトウェア開発を円滑に行うための方法として、アプリケーションテンプレート (以下 APT) を用いた FW の利用方法を検討している。本稿では、APT の概念とその開発手順を示す。

2. フレームワークの課題

FW を利用した AP 開発は、FW の提供する基本制御構造に従って FW の提供するクラスを拡張することで実現される。これによりソフトウェアの開発コード量を抑えつつ、機能拡張を行う手段を提供する。これまでに様々な分野で数多くの FW が開発されており、そのソフトウェア開発に与える効果も示されている [1]。

しかし、大規模なソフトウェア開発において多くの AP 開発者に FW 利用方法を正しく理解してもらい、AP 開発の均質化を図ることは、現状では必ずしも成功していない。この FW 利用の問題の大きな原因として次の 2 点を挙げることができる。

- FW 開発者にとって、FW の設計情報の伝達範囲が不明確なこと
- FW 利用者にとって、FW の機能拡張方法

が不明確であること

前者の問題に関しては、現状では個々のクラスに対するインタフェース仕様が提示されているが、FW の用途にクラス構成や改修のポイント等、理解すべきことが多いという課題がある。この解決策として、例えばデザイン・パターンの利用やソフトウェア・アーキテクチャの明示が提唱されている [2]。しかし、ドキュメント量の増加に対する抵抗も大きい。

一方、後者への解決策には FW にサンプルコードを添付する方法が主流である。しかしこの方法は FW の理解を前提としており、FW の機能拡張を実現するための改修範囲や改修量を見通すことが困難であるという問題が出ている。

この結果、FW の理解やその利用には個々の AP 開発技術者の前提知識や理解力が大きな影響を与えており、FW を大規模の AP 開発に利用する状況下では FW に期待される効果を得ることは困難である。

3. アプリケーションテンプレート

我々が提唱する APT は、AP 開発者が AP 開発を行う際の FW のホットスポット [3] の機能拡張を支援するためにある。APT は以下の 2 要素からなる。

- システムの 1 ユースケース (以下 UC) を表現する要素
我々はこれをユースケーステンプレート (以下 UCT) と呼ぶ。UCT は、エンドユーザから見える「ソフトウェアの機能」に対応して用意され、UC のシナリオ記述をもとに機能拡張が可能な部分と固定される部分を明示

し、AP 開発者が機能拡張を行う際の手がかりを与える。ひとつの UCT には次に述べるホットスポットテンプレートを複数含む。

- ホットスポットに対応するテンプレートとしてのサンプルコード

これをホットスポットテンプレート（以下 HST）と呼ぶ。HSTでは FW の提供するホットスポットに対する機能拡張事例として記述される。HSTは、機能の呼び出し順序等の、FW 利用上の遵守項目や、機能拡張箇所を明示している。

4. アプリケーションテンプレートの開発

APTは以下の手順で開発を行う。

- (1) 顧客要求の具体化を行うため、業務分析と UC 分析を行ない、基本的な機能項目を UC とし、各 UC のシナリオを作成する。
- (2) 複数の AP に対する共通性分析を行う。複数の AP に出現する UC を構成するオブジェクト群は、共通機能を実現する共通オブジェクトとして切り出す。
- (3) 共通オブジェクト群を FW とし、UC シナリオを機能固定部分と機能拡張部分とに分類し、UCT を作成する。
- (4) UCT の機能拡張部分は HST として FW のホットスポットを実装する形で用意する。ソフトウェア部品として、FW と HST を併せて利用者に提供する。

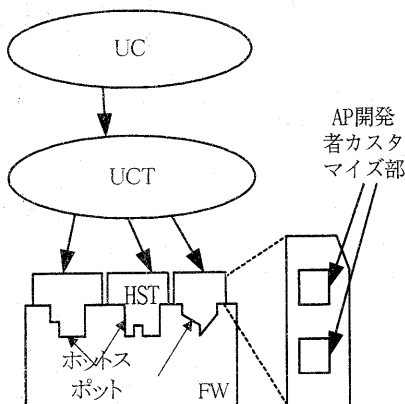


図 1 AP テンプレート概念図

5. 評価

今回はプロトタイプとして帳票作成業務について本手法を適用し、FW と APT の開発から AP 開発までを実施した。本手法を用いて APT を提供することにより UC から一貫した理解しやすい枠組みを提供でき、APP 開発者に対する教育期間を短縮できた。反面、FW の機能を効率よく利用した HST を UCT から分離する基準が重要であり、今後の課題である。

6. まとめ

本稿では、APT の開発手順と課題について述べた。今後は実システム開発への展開を図るとともに残された課題の解決策を模索していく。

[参考文献]

- [1] Lewis, T., et.al., Object-Oriented Application Frameworks, Prentice-Hall Publishing (1995).
- [2] Fayad, M.E., et.al., Building Application Framework: Object-Oriented Foundations of Framework Design, John-Wiley and Sons Publishing (2000).
- [3] Pree, W., Design Patterns for Object-Oriented Software Development, Addison-Wesley Publishing (1994).

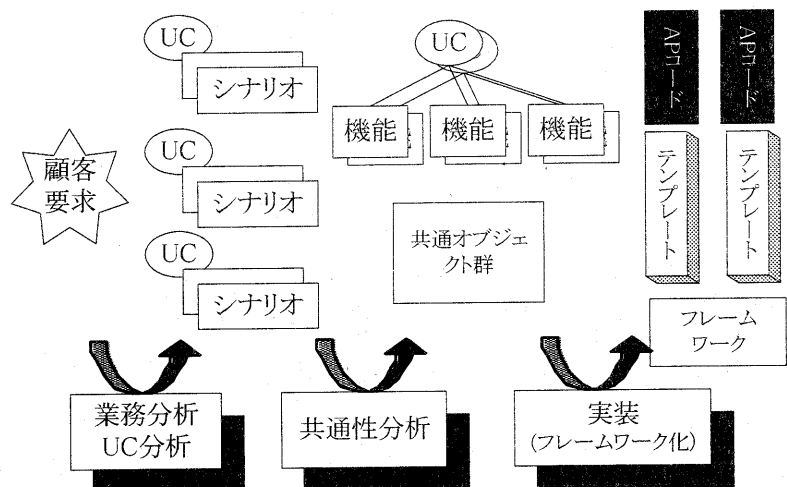


図 2 AP テンプレートの構築