

# アスペクト指向アーキテクチャに基づく PLSE に関する研究

M2005MM006 久松 康倫

指導教員 野呂 昌満

## 1 はじめに

再利用性の高いソフトウェア開発手法として、プロダクトラインソフトウェアエンジニアリング (以下, PLSE) の研究が盛んにおこなわれている。PLSE は共通の特徴をもつソフトウェア群であるソフトウェアファミリを対象に分析, 部品開発をおこなう。ソフトウェアファミリにおける分析結果をもとに, 製品の開発をおこなうことで, 開発効率の向上を目指している。バリエーションが豊富で, 開発期間の短縮化が求められている組込みソフトウェア開発において, PLSE の適用が必要であると考えられている。

本研究室では, 組込みソフトウェアアスペクト指向ソフトウェアアーキテクチャスタイル (以下 E-AOSAS++) を提案している。E-AOSAS++ は, 適切なモジュール化, 統一的な記述を規定することにより, 粒度の高い再利用性を保証を目指している。組込みソフトウェアを並行に動作する状態遷移機械の集合として構造の整理をおこなう。

本研究の目的は, 組込みソフトウェア開発へ PLSE を適用した際のコア資産の整理をおこなうことである。ユーザ要求と E-AOSAS++ に基づくアーキテクチャの関連づけをおこなう。ユーザ要求とアーキテクチャの関連づけをおこなうことで, ユーザ要求の変更に柔軟に対応可能となると考える。ソフトウェア部品の自動生成の枠組の提案をおこなう。自動生成に必要な情報と枠組をコア資産として定義する。

提案した方法を用いて, 自動販売機制御ソフトウェアを開発をおこなう。ユーザ要求の変更を考え, 提案した方法の妥当性について考察をおこなう。

## 2 関連研究

本章では, PLSE, Feature-Oriented Reuse Method (以下, FORM), E-AOSAS++ について説明をおこなう。PLSE は再利用性を考慮した開発方法論である。FORM は, PLSE における分析を支援する方法論である。E-AOSAS++ は, 再利用性の高いアーキテクチャを統一的に記述することを支援する枠組である。

### 2.1 PLSE

PLSE[2] は, 近年の大規模化したソフトウェアの多機能化, 複雑さを解決するために, 考案されたソフトウェア開発方法論である。

ソフトウェア開発を, 共通の特徴をもったソフトウェア群であるプロダクトラインに対して, 系統的におこなうことで, 開発期間の短縮化を目指している。

PLSE は次の 3 つの活動を系統的に行うよう定義されている。

- ドメインエンジニアリング  
プロダクトラインの分析を行いコア資産の開発を行う。
- アプリケーションエンジニアリング  
コア資産を用いてターゲットとするアプリケーションソフトウェアを開発する。
- マネージメント  
コア資産の管理を行う。

### 2.2 FORM

FORM は, PLSE におけるドメイン分析を, 支援する方法論である。Feature-Oriented Domain Analysis[1](以下, FODA) のフィーチャ図を拡張している。FORM のフィーチャ図は, 特性層, 操作環境層, ドメイン技術層, 実現技術層の四階層で表される。

フィーチャは, 用途によって必須フィーチャ, オルタネイティブフィーチャ, オアフィーチャ, およびオプションフィーチャに分類される。必須フィーチャはプロダクトラインにおいて絶対必要となるフィーチャである。複数のフィーチャからひとつを選択するのがオルタネイティブフィーチャ, ひとつ以上選択可能なのがオアフィーチャである。オプションフィーチャは服的に扱われるフィーチャである。

### 2.3 E-AOSAS++

本研究室では, 組込みソフトウェアの研究をおこなってきた経験から, 組込みソフトウェアにおけるアスペクト指向ソフトウェアアーキテクチャスタイル (以下, E-AOSAS++) を提案している。E-AOSAS++ は, 適切なモジュール化, 統一的な記述を規定することにより, 粒度の高い再利用性を保証を目指している。組込みソフトウェアを並行に動作する状態遷移機械の集合として構造の整理をおこなう。並行に動作する状態遷移機械の構造を図 1 に示す。

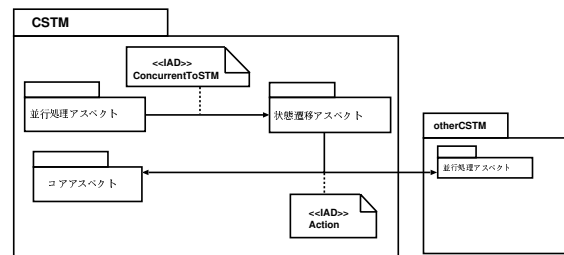


図 1 並行状態遷移機械の構造

状態遷移機械は、プライマリコンポーネントとコンポジットコンポーネントが存在する。コンポジットコンポーネントは、コンフィギュレーションコンポジットコンポーネントとアグリゲーションコンポジットコンポーネントの二種類が存在する。コンフィギュレーションコンポジットコンポーネントは、役割が異なる複数のベース状態遷移機械とポリシー状態遷移機械で構成される。ポリシー状態遷移機械によって、ベース状態遷移機械を切替えることでコンフィギュレーションコントロールを実現している。ポリシーが実時間処理ポリシー、例外処理ポリシーになることで非機能特性を扱うことができる。アグリゲーションコンポジットコンポーネントは複数のベース状態遷移機械と、アグリゲーションポリシー状態遷移機械から構成される。ベース状態遷移機械はアグリゲーションポリシーの状態遷移によって、協調動作をおこなう。

### 3 PLSE の適用

E-AOSAS++ に基づくアーキテクチャを PLSE に適用させるために、フィーチャ図とアーキテクチャの関連、ソフトウェア部品の自動生成を考える。

#### 3.1 フィーチャ図とアーキテクチャの関連

PLSE に E-AOSAS++ に基づくアーキテクチャを適用させるために、フィーチャ図とアーキテクチャの関連を考える。フィーチャ図とアーキテクチャの関連を考えることで、ユーザ要求に対するアーキテクチャの特定を容易におこなうことが可能となる。

本研究では、ひとつのユーザ要求に対して、ひとつの状態遷移機械を用意する。状態遷移機械を複合することにより、機能を実現する。

#### フィーチャとサブフィーチャ

フィーチャは、複数のサブフィーチャを複合して、表現される。フィーチャは、複数のサブフィーチャを実現した状態遷移機械の複合状態遷移機械として、構造化することが可能であると考えられる。フィーチャとアーキテクチャの関連を図 2 に示す。

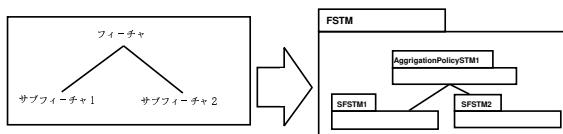


図 2 フィーチャ図とアーキテクチャの関連

#### オルタネイティブフィーチャ

オルタネイティブフィーチャは、複数のフィーチャからひとつを選択する。E-AOSAS++ のコンフィギュレーションコンポジットコンポーネントを用いることで表現できる。コンフィギュレーションコンポジットコンポーネントは、ポリシーによって、動作する状態遷移機械を変更することで、複数の役割を果たすことを可能にしている。オルタネイティブフィーチャの選択したフィーチャによって、動作する状態遷移機械を選択することが

可能となる。オルタネイティブフィーチャとアーキテクチャの関連を図 3 に示す。

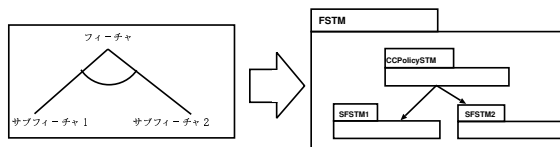


図 3 オルタネイティブフィーチャとアーキテクチャの関連

#### オアフィーチャ

オアフィーチャは、複数のフィーチャからひとつ以上を選択する。選択されるサブフィーチャによって、サブフィーチャで構成されるフィーチャの役割は異なる。選択されるフィーチャの組合せによって、異なるポリシーを順ぶする必要がある。オアフィーチャとアーキテクチャの関連を図 4 に示す。

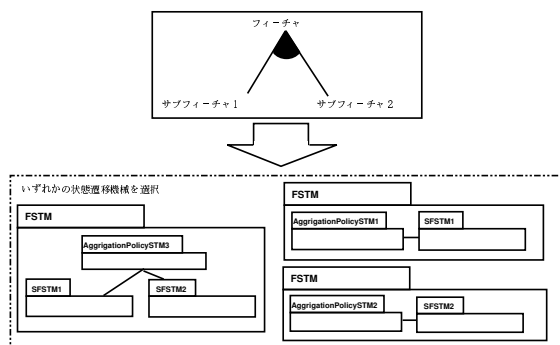


図 4 オアフィーチャとアーキテクチャの関連

#### オプションアルフィーチャ

オプションアルフィーチャは、プロダクトラインにおいて、付加的に扱われるフィーチャである。オプションアルフィーチャを選択時には、オプションアルフィーチャを実現する状態遷移機械の追加する。オプションアルフィーチャが追加される状態遷移機械のポリシーを変更する必要がある。オプションアルフィーチャと、アーキテクチャの関連を図 5 に示す。

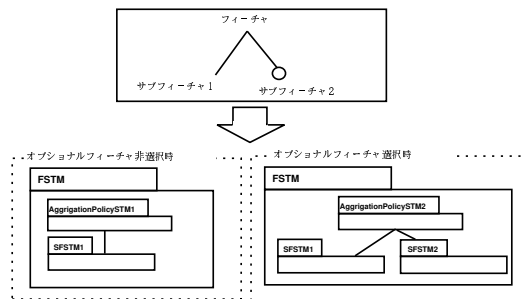


図 5 オプションアルフィーチャとアーキテクチャの関連

#### 3.2 自動生成の枠組の提案

ソフトウェア部品の自動生成の枠組の提案をおこなう。E-AOSAS++ では、ソフトウェアは並行に動作する状

状態遷移機械の集合として規定している。状態遷移機械コードの自動生成をおこなうことで、生産性が向上すると考えられる。

状態遷移機械のクラス図を図6に示す。

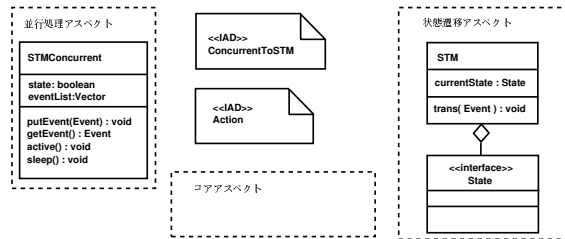


図6 状態遷移機械のクラス図

本研究では、オブジェクト指向言語として、広く使われている Java 言語のアスペクト指向拡張言語である AspectJ を用いて、実現をおこなう。実現した状態遷移機械コードを分析し、定型コード、自動生成可能コード、ユーザ記述コードの抽出をおこなう。抽出した結果を表1に示す。

表1 自動生成可能部分

並行処理アスペクト	定型コード
状態遷移アスペクト	状態遷移図から自動生成可能
コアアスペクト	ユーザ記述
並行処理アスペクトと状態遷移アスペクト間のアスペクト間記述	定型コード
アクション	状態遷移図から一部自動生成可能

#### 4 自動販売機制御ソフトウェア開発への適用

組込みソフトウェアの典型的な例のひとつである自動販売機制御ソフトウェアを提案した方法を用いて開発をおこなう。開発は自動販売機のドメイン分析をおこない、フィーチャ図を作成する。フィーチャ図とアーキテクチャの関連づけをおこなう。フィーチャの選択をおこない、プロダクトの開発をおこなう。

##### 4.1 自動販売機の分析

自動販売機制御ソフトウェアの分析をおこなう。自動販売機制御ソフトウェアのドメインに存在する全てのフィーチャの抽出をおこなう。フィーチャは現在の市販されている複数の自動販売機を分析することで抽出をおこなう。抽出したフィーチャの関連を整理したフィーチャ図を図7に示す。

##### 4.2 自動販売機のアーキテクチャ

分析した結果をもとに自動販売機制御ソフトウェアの構造を考える。自動販売機は、販売モードと管理モードが存在する。本研究では、販売モードについてだけ考えている。自動販売機はお金を取り扱う部品と、商品を取り扱う部品が存在する。

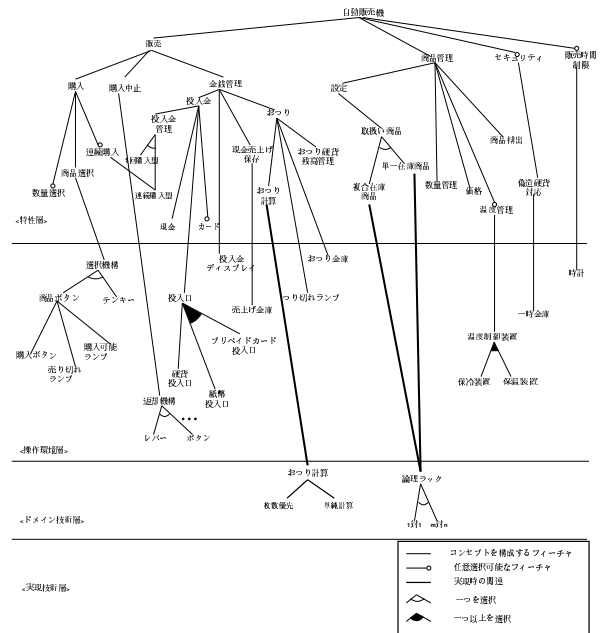


図7 自動販売機のフィーチャ図

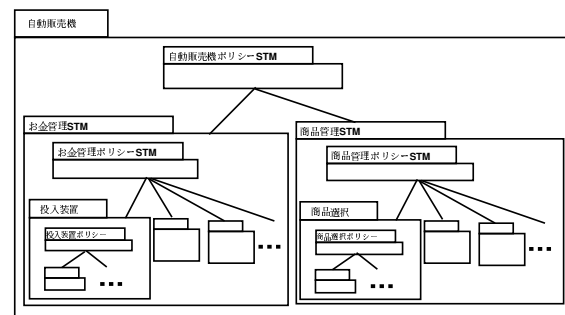


図8 自動販売機のアーキテクチャ

#### 5 考察

本研究では、コア資産関係の整理をおこなった。特にユーザ要求とアーキテクチャの関連について考えた。ユーザ要求とアーキテクチャの関連を提案することで、ユーザ要求の変更に対応が可能になると考えた。アスペクト指向アーキテクチャを適用した PLSE におけるユーザ要求とアーキテクチャの関連以外のコア資産を、以下にあげる。

- ユーザ要求 (フィーチャ図)
- アーキテクチャ
- 状態遷移図
- 再利用可能なソフトウェア部品 (状態遷移コード)
- 状態遷移コードの自動生成の枠組

コア資産として提案した、ユーザ要求とアーキテクチャの関連のルール、および自動生成の枠組を用いることで

開発効率の向上が計れると考えた。提案した方法を用いた自動販売機制御ソフトウェアの開発を用いて考察をおこなう。

### 5.1 ユーザ要求変更に関する考察

ユーザ要求の変更に対する構造の変更点について考察をおこなう。例として、自動販売機制御ソフトウェアの入金装置について考える。提案した方法をもとに、入金装置に構造を考えると図9のようなになる。

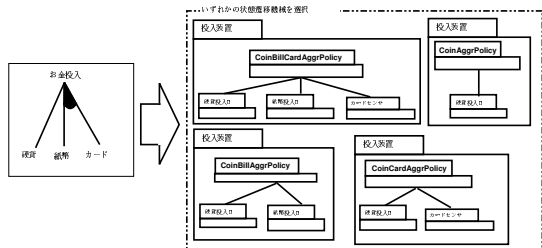


図9 入金装置のフィーチャ図とアーキテクチャの関連

図10の左側は入力装置として硬貨、紙幣、およびカードを選択したとき、右側は硬貨、紙幣を選択したときの構造である。ユーザ要求の変更によって、選択されるオアフィーチャは異なる。入力可能なお金の種類に対する要求が変更されることで、おつりの計算など、お金に関する機能が変更する。本研究では、ユーザ要求に対して、ひとつの状態遷移機械を用意し、複合状態遷移機械として、扱うことで機能の実現をおこなっている。ユーザ要求の変更に対して、複合状態遷移機械のポリシーを変更することで、機能の変更が可能である。本研究で提案した、ユーザ要求とアーキテクチャの関連のルールを用いることで、ユーザ要求に対して、アーキテクチャの変更を柔軟に対応することが可能になったと考えられる。

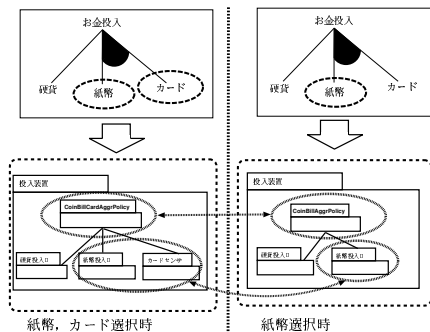


図10 フィーチャ選択時の構造の比較

### 5.2 ソフトウェア部品の生産性に関する考察

ソフトウェア部品の自動生成にともなう生産性の向上について考察をおこなう。本研究では、ユーザが記述する必要のあるコードの減少することで生産性の向上が計れたと考えた。図11のような状態遷移をもつ状態遷移機械の実現を考えたときの、コード量を表2に示す。

E-AOSAS++ を適用することで、統一的な記述でモジュール化をおこなった。状態遷移機械コードのユーザ

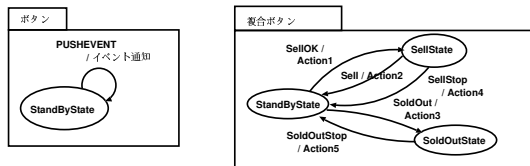


図11 ボタンと複合ボタンの状態遷移図

表2 コード比較

状態遷移機械	定型	自動生成	ユーザ記述	合計
ボタン	104	15	10	129
複合ボタン	104	35	50	189

が記述すべきコードはアクションコードとコアアスペクトコードである。単純な状態遷移をしているボタンの状態遷移機械では、ユーザが記述すべきコードは状態遷移機械コードの10%以下であった。状態の数がボタンより多い複合ボタンの状態遷移機械におけるユーザ記述コードは、アクションでおこなうことが多いこともあり、全体の約25%であった。本研究で実現した自動販売機制御ソフトウェアでは、全コードの約30%がユーザ記述コードであった。ユーザが記述すべきは、アクションとコアアスペクトも特定でき、コード量は全体の約30%となり開発の効率の向上が計れたと考えられる。

## 6 おわりに

本研究は、PLSEへのアスペクト指向ソフトウェアアーキテクチャの適用を考えた。アーキテクチャとユーザ要求の関連づけをおこなった。関連づけをすることで、ユーザ要求の変更にともなう、ソフトウェアの構造の変更を容易におこなうことが可能になった。E-AOSAS++に基づくアーキテクチャにおけるソフトウェア部品となる状態遷移機械の自動生成の枠組の提案をおこなった。

今後の課題として、提案した枠組をもとに、自動生成ツールの作成が考えられる。

### 謝辞

本研究を進めるにあたり、熱心に御指導下さいました野呂昌満教授、有益なアドバイスを頂いた蜂巢吉成先生、張漢明助教授に深く感謝致します。

### 参考文献

- [1] K. C. Kang, J. Lee, and P. Donohoe, "Feature-Oriented Product Line Engineering", *IEEE Software*, Vol. 19, No. 4, pp.58-65, 2002.
- [2] L. M. North, "SEI's Software Product Line Tenets", *IEEE Software*, Vol. 19, No. 4, pp.32-40, 2002.