

自動販売機制御ソフトウェアへの PLSE の適用

2003MT035 加藤 大地 2003MT107 渡辺 基裕 2003MT120 後藤 洋
指導教員 野呂 昌満

1 はじめに

近年、ソフトウェアを系統的に開発するための開発方法の一つとして、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)[1] が提案されている。PLSE ではユーザ要求からアーキテクチャを系統的に構築する手法ははまだ確立されていない。

一方、本研究室では組込みソフトウェアのAspect指向ソフトウェアアーキテクチャスタイル (以下、E-AOSAS++) が提案されている。E-AOSAS++ はソフトウェア開発において、粒度の高い再利用性を保証する技術である。

本研究室ではユーザ要求からアーキテクチャを系統的に構築するために、PLSE を E-AOSAS++ にもとづいて適用することを提案している。E-AOSAS++ はアーキテクチャを粒度の高い部品として分割するのでユーザ要求に対応させやすい。われわれは、E-AOSAS++ にもとづいた PLSE が自動販売機制御ソフトウェアに適用可能であると考えられる。

本研究の目的は、E-AOSAS++ にもとづいた PLSE を自動販売機制御ソフトウェアの開発に適用し、適用の有用性を考察することである。

研究手順を以下に示す。

- 自動販売機制御ソフトウェアの要求、仕様、構造の整理
- 自動販売機制御ソフトウェアに E-AOSAS++ にもとづいた PLSE を適用
- 適用の有用性を考察

なお、加藤は主にフィーチャモデリング、後藤は主にアーキテクチャの構築、渡辺は主に应用可能性の考察を担当した。

2 関連研究

2.1 E-AOSAS++

E-AOSAS++ は、組込みソフトウェアアーキテクチャを並行に動作する並行状態遷移機械 (以下、CSTM) の集合として規定する。複数の CSTM は、互いにメッセージを送り協調動作をする。CSTM は、並行処理アスペクト、状態遷移アスペクト、コアアスペクト、アスペクト間記述で構成される。各アスペクトの関連を図 1 に示す。アスペクト同士のつながりはアスペクト間記述によって書かれている。コアアスペクトへのデータアクセスやほかの CSTM へのイベント通知はアクションに記述され

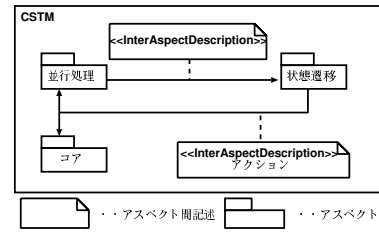


図 1 E-AOSAS++ のアスペクトの関連

ている。各構成要素の処理を以下に示す。

- 並行処理アスペクト
状態遷移機械を並行に動作させる処理
- 状態遷移アスペクト
状態の遷移に関する処理
- コアアスペクト
主要コンサーンを実現するオブジェクト群

複数の CSTM を集約して、ある機能を実現させたものを複合 CSTM とする。複合 CSTM 内にある複数の CSTM を同時に動かすさいにはアグリゲーションコンポジット CSTM を用いる。アグリゲーションコンポジット CSTM は複合 CSTM 内の CSTM を管理する。アグリゲーションコンポジット CSTM を通じてイベント通知をおこなうことで連動動作をする。アグリゲーションコンポジット CSTM の挙動を図 2 に示す。

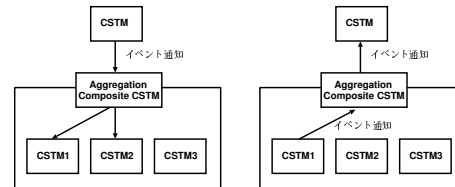


図 2 アグリゲーションコンポジット CSTM

本研究では、CSTM を独立した部品とみなし、コンポーネントと呼ぶ。

2.2 PLSE

PLSE とはソフトウェア開発を系統的におこなうための技術である。PLSE ではコア資産をもとに部品を組み合わせることで製品開発をおこなう。コア資産とは、要求分析の結果、再利用可能部品、部品化の手法およびアーキテクチャなどの開発に必要な項目全ての総称である。PLSE は以下で構成される。

- ドメイン工学
 - ユーザ要求の分析、コア資産の構築
- アプリケーション工学

- 部品を組み合わせ、製品を作成
- 管理
 - コア資産の運用、保守

2.3 フィーチャモデリング

本研究ではフィーチャモデリングを適用してユーザ要求のモデル化をおこなう。フィーチャモデリングはFORMで提案されているユーザ要求をモデル化する手法である。

- フィーチャ
 - 製品の特徴、特性、開発者側からみた実現可能なユーザ要求
- サブフィーチャ
 - フィーチャが包含する詳細なフィーチャ群

フィーチャ同士の関係をあらわすためにフィーチャ図[2][3]をもちいる。ルートは製品名をノードはフィーチャをあらわす。フィーチャの子はサブフィーチャである。フィーチャ図の例を図3に示す。

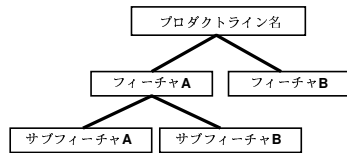


図3 フィーチャ図の例

フィーチャ間関係を図4に示す。

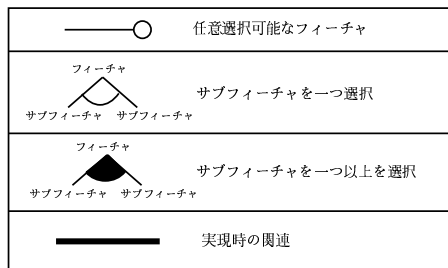


図4 フィーチャ間関係

フィーチャ図にはフィーチャの種類ごとに階層が存在する。

- 特性層 (Capability layer)
 - 機能特性や非機能特性のつながり
- 操作環境層 (Operating environment layer)
 - ハードウェアに関する部分
- ドメイン技術層 (Domain technology layer)
 - ドメインに特化した技術
- 実現技術層 (Implementation technique layer)
 - 一般的なソフトウェアの実現に利用されている技術

3 E-AOSAS++ にもとづいた PLSE

FORMではフィーチャモデルから必要なフィーチャを選択してソフトウェアを設計・実現する。フィーチャの

選択のパターンは多種多様で設計・実現が困難である。様々なフィーチャの選択に対応するためのアーキテクチャが必要である。本研究では、E-AOSAS++にもとづいてアーキテクチャを構築すると、様々なフィーチャの選択に対応できると考えた。

E-AOSAS++のコンポーネントとフィーチャの対応関係を図5に示す。

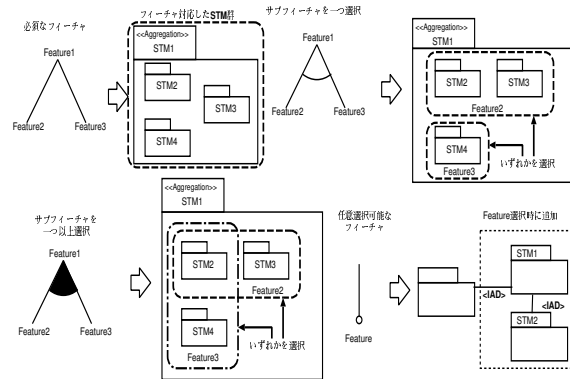


図5 フィーチャ図とアーキテクチャの関係

選択されたフィーチャに合わせてSTM群の構成を選択することによって、フィーチャとアーキテクチャを対応させる。

4 PLSEの適用

本研究では自動販売機制御ソフトウェアにPLSEの適用をおこなう。フィーチャモデリングとアスペクト指向ソフトウェアアーキテクチャの構築をおこなう。フィーチャとE-AOSAS++のコンポーネントの対応付けをおこない、選択したフィーチャをもとにコンポーネントを組み合わせる。

4.1 自動販売機制御ソフトウェアのソフトウェアモデル

自動販売機を構成している各ハードウェアに対し、それを制御するソフトウェアが必要であるとする。ハードウェア構成をもとに自動販売機制御ソフトウェアのオブジェクト指向分析をおこなった。PLSE適用の対象とする自動販売機制御ソフトウェアのソフトウェアモデルのクラス図を図6に示す。

4.2 フィーチャモデリング

自動販売機制御ソフトウェアの共通点と相異点を整理し、分析結果からフィーチャモデリングをおこなう。作成したフィーチャ図を図7に示す。特性層では自動販売機が提供する機能のフィーチャを記述した。操作環境層では自動販売機の構成するハードウェアをフィーチャとして記述した。

4.3 E-AOSAS++を適用したアーキテクチャ

E-AOSAS++を適用し、自動販売機制御ソフトウェアのアーキテクチャを構築する。ソフトウェアモデルの分

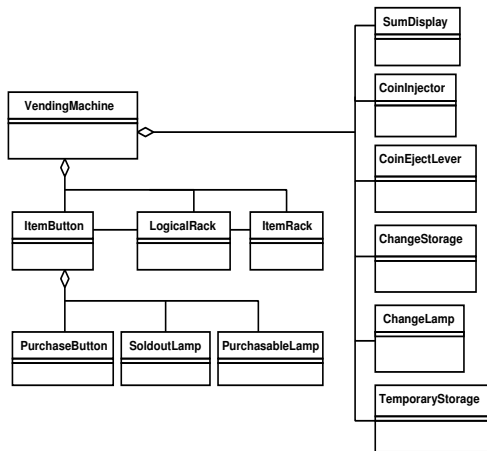


図6 自動販売機制御ソフトウェアのソフトウェアモデル

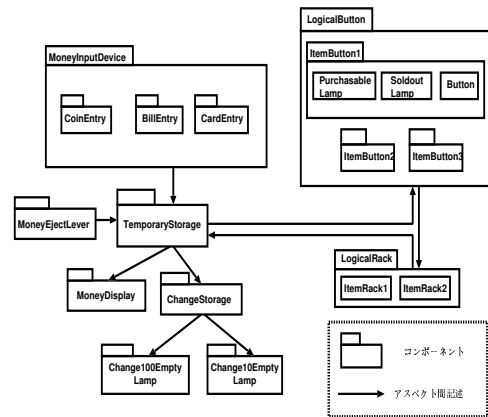


図8 自動販売機制御ソフトウェアのアーキテクチャ

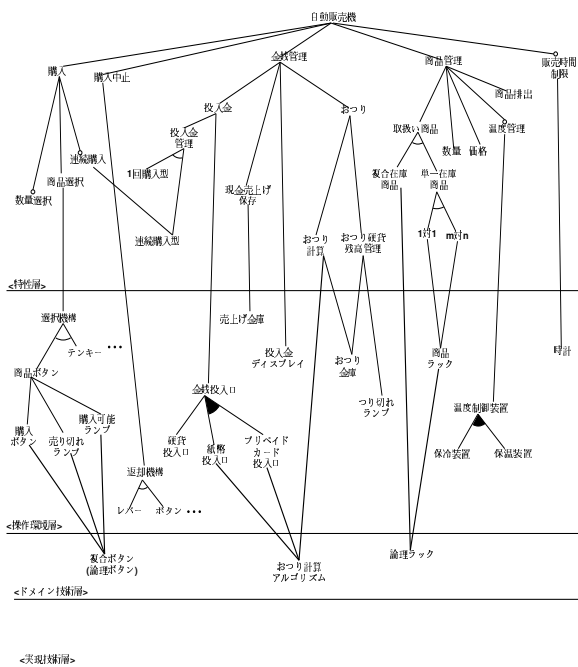


図7 自動販売機制御ソフトウェアのフィーチャ図

析結果をもとにコンポーネントを構築し、各フィーチャの関連からコンポーネントを組み合わせた。構築したアーキテクチャを図8に示す。

4.4 フィーチャとコンポーネントの対応関係

アーキテクチャの各コンポーネントはフィーチャと対応関係をもつ。対応関係の例として金銭投入口フィーチャをあげる。対応関係を示した図を図9に示す。

金銭投入口フィーチャには、硬貨投入口、紙幣投入口、プリペイドカード投入口の三つのサブフィーチャがある。硬貨投入口は CoinEntry、紙幣投入口は BillEntry、カード投入口は CardEntry に対応している。サブフィーチャを包含するフィーチャに対してアグリゲーションコ

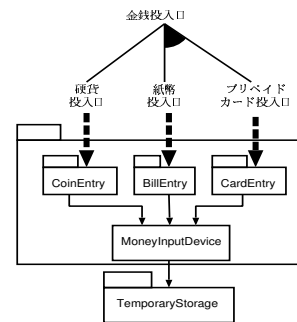


図9 フィーチャとコンポーネントの対応関係

ンポジットコンポーネントを構築する。アグリゲーションコンポジットコンポーネントとして構築することでコンポーネントの追加・変更による修正部分を局所化することができる。紙幣投入口、プリペイドカード投入口の二つのフィーチャは実現時の関連としておつり計算アルゴリズムフィーチャに関連する。

5 部品の対応関係の考察

PLSE を適用した自動販売機制御ソフトウェアについて考察する。部品の対応関係について考察する。E-AOSAS++ を適用したアーキテクチャはコンポーネント間をアスペクト間記述 (IAD) で結ぶので疎結合となり、部品の独立性が高まる。部品の独立性が高ければ、部品の追加、変更が容易になると考えられる。

5.1 部品追加に関する考察

部品の追加の例として金銭投入口フィーチャの変更をあげる。金銭投入口フィーチャには、硬貨投入口、紙幣投入口、プリペイドカード投入口の三つのサブフィーチャがある。硬貨投入口、紙幣投入口、プリペイドカード投入口の三つのフィーチャの選択に変更したときのアーキテクチャの変更部分について図10に示す。CardEntry と MoneyInputDevice を結ぶアスペクト間記述を追加し、関連する MoneyInputDevice を切替えることで CardEntry を追加することができる。

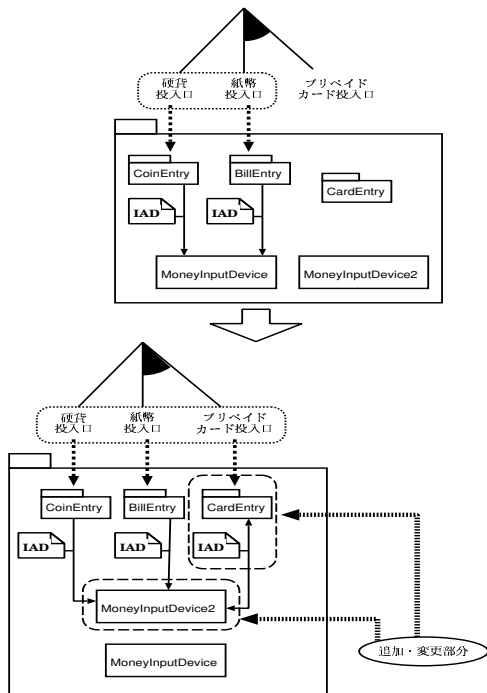


図 10 フィーチャ追加によるアーキテクチャの変更部分

5.2 部品変更に関する考察

部品の変更の例として連続購入可能な自動販売機への変更をあげる。連続購入可能とは商品購入後に残金が商品の価格分あれば、商品が選択可能な状態になることである。本研究で構築したアーキテクチャでは、TemporaryStorage が現在投入された金額を管理している。連続購入可能な自動販売機を実現するには、TemporaryStorage を変更する必要がある。変更部分を図 11 に示す。TemporaryStorage 内のアスペクト間記述を変更

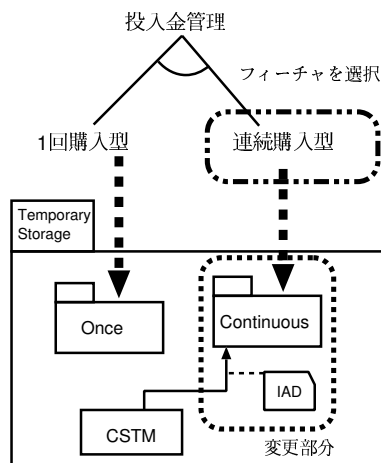


図 11 フィーチャ変更によるアーキテクチャの変更部分

し、Continuous コンポーネントにイベント通知先を変更する。TemporaryStorage 以外のコンポーネントの変更なしで、連続購入可能な自動販売機が実現できることを確認した。

5.3 ドメインの特徴

自動販売機制御ソフトウェアというドメインにおいて、対応関係には2つのパターンが存在することを確認した。その他のドメインにおいて別の種類のパターンが存在する可能性があるため、ドメインごとのパターンをコア資産として保持する必要があると考える。以下に自動販売機制御ソフトウェアにおけるパターンをしめす。

- フィーチャと部品の関係が1対1
 - フィーチャの変更が1つの部品の変更のみで対応できる場合
- フィーチャと部品の関係が1対複数
 - フィーチャの変更に対応するために複数の部品を変更する必要がある場合

6 おわりに

本研究では、自動販売機制御ソフトウェアへ E-AOSAS++ にもとづいた PLSE の適用をした。フィーチャと E-AOSAS++ の対応関係を確認し、適用可能であることが確認できた。自動販売機制御ソフトウェアに適用できたことから、ほかの組込みソフトウェアにも PLSE の適用ができると考えられる。今後の課題は PLSE の開発サイクルをくり返し、コア資産を再検討することである。

謝辞

本研究をすすめるにあたり、熱心な御指導をいただいた野呂昌満教授、有益なアドバイスをくださった張漢明助教授、大学院生の久松康倫さん、坂野将秀さん、水野耕太さん、西山遼平さん、太田将吾さんみなさんに深く感謝いたします。また2年間共にごんばってきた野呂研究室、張研究室のみなさんにも感謝致します。

参考文献

- [1] Linda M. Northrop, “SEI’s Software Product Line Tenets”, pp. 32-40, IEEE SOFTWARE, vol. 19, No. 4, July/August 2002 .
- [2] Kyo C. Kang, Jaejoon Lee, Patrick Donohoe, “Feature-Oriented Product Line Engineering”, pp. 58-65, IEEE SOFTWARE, Vol. 19, No. 4, July/August 2002 .
- [3] Krzysztof Czarnecki, Ulrich W. Eisenecker, “Generative Programming: Methods, Tools, and Applications”, pp. 38-43, Addison-Wesley, 2000 .