

# 携帯電話制御ソフトウェアへのPLSEの適用

2003MT014 早崎 有哉

2003MT077 野田 博之

指導教員 野呂 昌満

## 1 はじめに

携帯電話制御ソフトウェアは、ユーザ要求の多様化によって、多機能かつ複雑な携帯電話制御ソフトウェアが存在する。製品の改版が頻繁に行われる場合に開発コストを削減する目的で、PLSE(Product Line Software Engineering)[1]が、注目されている。再利用性を目的に系統的にソフトウェア開発をおこなう技術の一つである。携帯電話の特徴から、携帯電話制御ソフトウェアをPLSE適用領域の典型的な例として考えた。

PLSEを適用した開発においては、構造と要求の対応づけが不明確という問題が挙げられる。携帯電話制御ソフトウェア開発においては、製品ごとに開発がおこなわれており、PLSEへの試みは不十分である。

本研究の目的は、携帯電話制御ソフトウェアを事例として、本研究室で提案されているE-AOSAS++とPLSE(E-AOSAS++とフィーチャ図)の対応関係を用いて開発を行う。得られたアーキテクチャから対応関係の妥当性を分析し、考察することである。E-AOSAS++(組込みソフトウェアのアスペクト指向アーキテクチャスタイル)を用いることで、様々なフィーチャの選択に対応し、より系統的に開発ができると考えられる。また、より再利用性の高い携帯電話制御ソフトウェア開発を行うことができると考えられる。

本研究は、以下のように進めた。

- 携帯電話のソフトウェアモデルの作成
- 携帯電話のフィーチャ図の作成
- PLSE, E-AOSAS++を適用し、携帯電話制御ソフトウェアのアーキテクチャの構築
- 携帯電話特有のアーキテクチャの定義
- 再利用性に関する考察

## 2 関連研究

PLSEとE-AOSAS++について説明する。

### 2.1 Product Line Software Engineering

PLSEは、ソフトウェア製品開発において再利用性を重視し、系統的な開発を支援する手法の一つとして提案されている。アーキテクチャを中心としてソフトウェアを開発するための工学である。PLSEの活動を図1を用いて以下に示す。

1. ドメイン (応用領域) におけるソフトウェア群の分析, コア資産の構築
2. アプリケーションエンジニアリング  
コア資産をもとに、ソフトウェア製品を開発
3. 管理  
ドメインの情報、開発方法などの管理

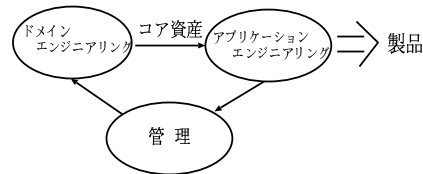


図1 PLSEの3つの活動

### フィーチャ図

フィーチャ図は、フィーチャの共通性と変更容易性を分析することが可能となり、PLSEの適用において重要な分析である。FORM(Feature Oriented Reuse Method)[1]で提案されているフィーチャ図は、以下の4つの階層に分割して作成する。

- 特性層  
組込みソフトウェアの機能特性や非機能特性を示す
- 操作環境層  
組込みソフトウェアのハードウェアに関する部分を示す
- ドメイン技術層  
組込みソフトウェア開発に特化した技術を示す
- 実現技術層  
一般的なソフトウェアなどの実現に利用される技術を示す

### 2.2 E-AOSAS++

E-AOSAS++では、組込みソフトウェアは並行に動作する複数の状態遷移機械の集合とみなす。各並行状態遷移機械(CSTM)は並行処理アスペクト、状態遷移アスペクト、コアアスペクトと、アスペクト間記述(Inter Aspect Description)から構成される。複数の並行状態遷移機械は、互いにイベントを送り合い、協調して動作する。

並行に動作する状態遷移機械では、状態遷移に関する処理と振舞に関する処理が横断的に関連する。状態遷移に関する処理と振舞に関する処理の再利用性を高める目的で、状態遷移に関する処理と振舞に関する処理を分離した。アスペクトの関連を図2に示す。

各アスペクトに記述する処理を以下に示す。

- 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  
状態遷移機械の状態の遷移に関する処理
- コアアスペクト  
アプリケーションロジックを含む主要コンサーンを実現するオブジェクト群

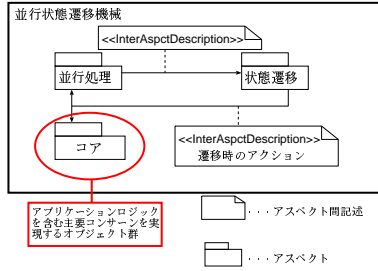


図2 E-AOSAS++における並行状態遷移機械の構造

### 構成管理 (コンフィギュレーションコントロール)

複雑化する携帯電話制御ソフトウェアの実現には、非機能特性としてコンフィギュレーションコントロール (以下, CC) を考慮する必要がある。CC の実現には並行状態遷移機械を使い, CC に関する状態遷移をメタ状態遷移, 各サブシステムでの状態遷移をベース状態遷移と呼ぶ。

CC を実現する方法として, メタ状態遷移とベース状態遷移を別の並行状態遷移機械に分け, メタ並行状態遷移機械 (CCPolicyCSTM) でベース並行状態遷移機械を切替える方法を考える。メタ並行状態遷移機械でベース並行状態遷移機械を切替える構造を図3に示す。

各並行状態遷移機械には Sleep 状態および Active 状態がある。Sleep 状態にある並行状態遷移機械は, 他の並行状態遷移機械から処理しなければならないイベントを受けると Active 状態になり, 処理を開始する。一方, 処理できないイベントを受け取ると, そのイベントは破棄され, Sleep 状態のままである。

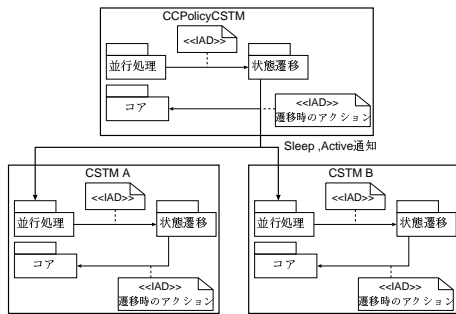


図3 メタ並行状態遷移機械とベース並行状態遷移機械

この場合, メタ状態遷移とベース状態遷移が分割されるので, 状態遷移の管理が容易になる。サブシステムを追加すると, 状態遷移の変更は局所化される。状態遷移の変更による変更箇所も局所化されるので, 追加・変更に関する柔軟性が高い。

CCPolicyCSTM が CSTM A を Active 状態から Sleep 状態に, CSTM B を Sleep 状態から Active 状態にして, コンフィギュレーションを変更するさいの処理の流れを図4に示す。なお, CPolicyCSTM は常に Active 状態である。

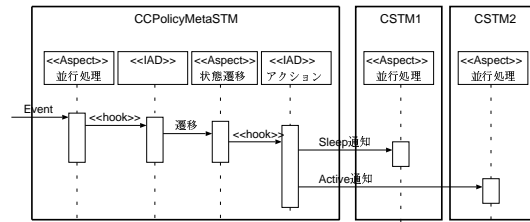


図4 コンフィギュレーション変更時のシーケンス図

### 3 E-AOSAS++ に基づいた 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。しかし, フィーチャの選択のパターンは多種多様で設計・実現が困難であると考えられる。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。E-AOSAS++ に基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++ とフィーチャ図の対応関係は図5のように提案されている。

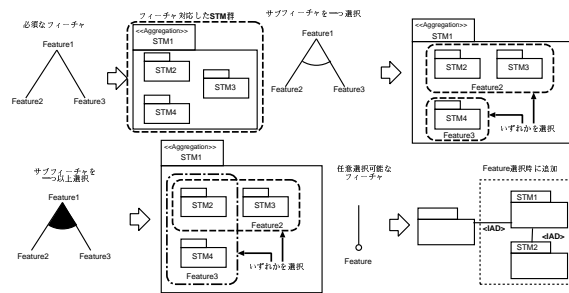


図5 フィーチャ図と E-AOSAS++ の対応関係

### 4 携帯電話制御ソフトウェアのアーキテクチャと PLSE

携帯電話制御ソフトウェアのアーキテクチャの構築をおこなう。構築のために携帯電話制御ソフトウェアの構成について分析をおこなう。分析から PLSE と E-AOSAS++ の対応関係を用いてアーキテクチャの構築をおこなう。

#### 4.1 携帯電話のソフトウェアモデル

オブジェクト指向設計による携帯電話のソフトウェアモデルを図6に示す。

#### 4.2 携帯電話の仕様

本研究では, フィーチャの選択として通話システム, 電話帳システム, Mail システムの機能を選択する。E-AOSAS++ に基づき, CC を取り入れた携帯電話制御ソフトウェアのアーキテクチャを構築し, 容易にカスタマイズ可能な携帯電話を実現する。

#### 4.3 フィーチャ図によるユーザ要求分析

ユーザ要求を整理する目的で, 携帯電話制御ソフトウェアを構成するフィーチャの関連を図7に示す。

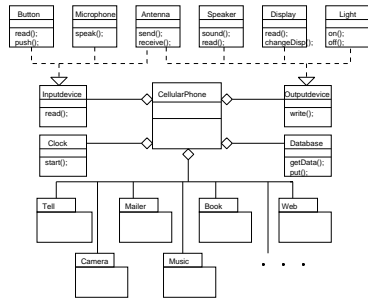


図6 携帯電話のソフトウェアモデル

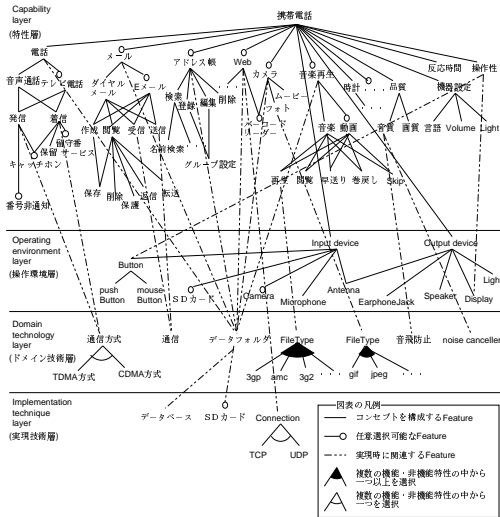


図7 携帯電話のフィーチャ図

#### 4.4 PLSE, E-AOSAS++ の適用

PLSE, E-AOSAS++ を適用し、構築した携帯電話制御ソフトウェアのアーキテクチャを図8に示す。

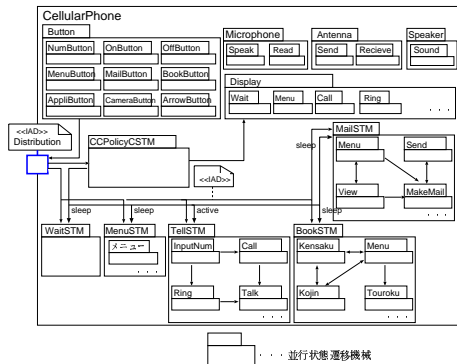


図8 携帯電話制御ソフトウェアのアーキテクチャ

図8では、携帯電話における各部品を並行状態遷移機械としている。フィーチャ図から、携帯電話の各機能はOptionalフィーチャとして考えられる。対応関係を適用するので、各機能をベース並行状態遷移機械として作成することができる。また、CCを考慮して実現するのでメタ並行状態遷移機械としてCCPolicyCSTMを作成し、これらをアスペクト間記述で繋ぐことで実現した。

ボタンからのイベントに対して、入口である Distribution で、CellularPhone 内部の全ての並行状態遷移機械にイベントを送る。コンフィギュレーション変更時のイベントであれば、メタ並行状態遷移機械 (CCPolicyCSTM) がイベントを受け取り、Active, Sleep 通知を各ベース並行状態遷移機械に通知することで、CCを実現している。

#### コンフィギュレーションの切替え

待受状態 (WaitSTM) から通話システム (TellSTM) にコンフィギュレーションが切り替わるさいの、イベントの流れを図9を用いて示す。待受状態で番号ボタンが押されると、そのイベントは Distribution によって、内部の全ての並行状態遷移機械に通知される。現在、Active 状態の WaitSTM では、このイベントは処理できないので破棄される。TellSTM は Sleep 状態なので当然破棄される。このイベントはコンフィギュレーション変更のイベントなので、CCPolicyCSTM がイベントを受け取る。WaitSTM には Sleep 通知を、TellSTM には Active 通知をおこない、コンフィギュレーションを切替えている。

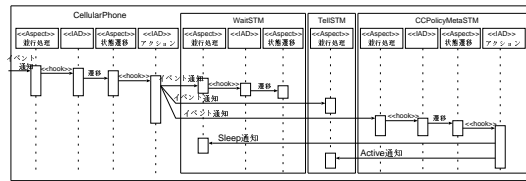


図9 コンフィギュレーション変更時のシーケンス図

### 5 考察

実現した携帯電話制御ソフトウェアについて、本研究では提案されている対応づけの適用に関する考察をおこなう。

#### 5.1 携帯電話特有のアーキテクチャの定義

定義したアーキテクチャを図10に示す。

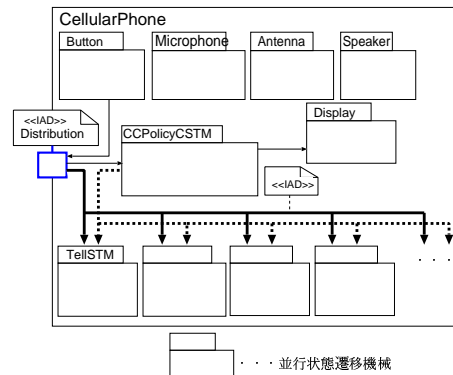


図10 携帯電話特有のアーキテクチャ

本研究では携帯電話を例に研究をおこなってきた。携帯電話の特徴として機能の追加・変更・削除があげられる。これは、フィーチャ図のOptionalフィーチャの強調として考えられる。以上より、本研究では携帯電話制

御ソフトウェア特有のアーキテクチャを図 10 のように定義することができた。

### 再利用できるアーキテクチャ

本研究で構築したアーキテクチャは、並行状態遷移機械で実現しているの、各部品を共通な構造で実現することができた。新しく機能を追加するさい、共通な構造を用いて、一部分を変更するだけで実現可能である。共通な構造を図 11 に示す。構造が共通なことからステートマシン図から共通な構造を自動的に作成することが可能と考えられる。

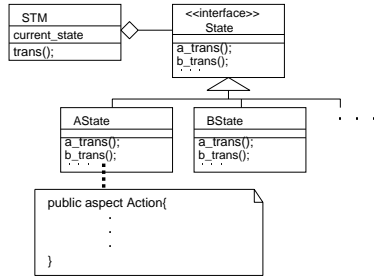


図 11 共通な構造

## 5.2 再利用性に関する考察

上記で定義したアーキテクチャをもとに、実際に機能を追加したさいのアーキテクチャを考える。

### 独立した機能の追加

通話システム、電話帳システム、Mail システムの機能を持つ携帯電話に、電卓システムを追加したさいの構成を図 12 に示す。

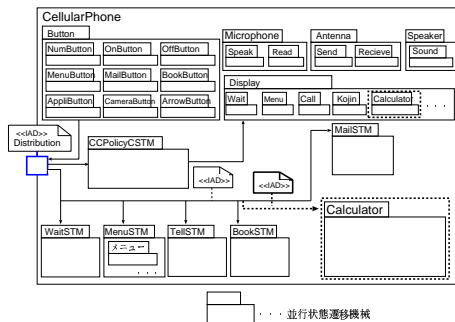


図 12 電卓機能を追加したさいの構成

ベース状態遷移機械として CalculatorSTM を作成し、CC を管理する CCPolicyCSTM にコンフィギュレーションを追加する。DisplaySTM にも、電卓用の並行状態遷移機械を追加し、アスペクト間記述で各部品間を結びつけることで実現可能となる。他の独立した機能を追加するときにも、同様に考えられる。

### 連携した機能の追加

通話システム、の機能を持つ携帯電話に、電話帳システムを追加したさいの構成を図 13 に示す。

ベース並行状態遷移機械として BookSTM を作成し、CC を管理する CCPolicyCSTM にコンフィギュレーションを追加する。ButtonSTM, DisplaySTM に電話帳用の並行状態遷移機械を追加し、アスペクト間記述で

各部品間を結びつけることで実現可能となる。他の連携した機能を追加するときにも、同様に考えられる。

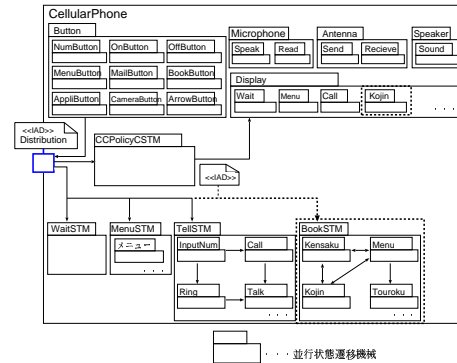


図 13 電話帳システムを追加したさいの構成

電話帳システムを追加したさい、新たな機能として電話帳発信が考えられる。電話帳システム、通話システムは互いに疎な関係であり、メタ並行状態遷移機械で CC を実現しているの、電話帳システムの一部に、アスペクト間記述で新たに追加するだけで、容易に可能である。しかし、フィーチャ図では、各機能を Option として考えているの、各アスペクト間の連携は記述できない。実現するさいには、開発者がフィーチャの選択によって連携を考える必要がある。

実際に、我々が実現したアスペクト間記述の一部を図 14 に示す。

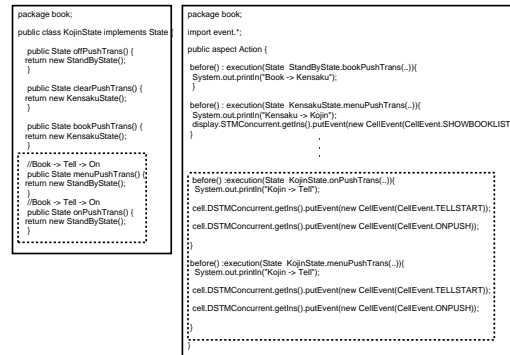


図 14 追加されたアスペクト間記述

## 6 おわりに

本研究では、携帯電話制御ソフトウェアに PLSE を適用し、さらに、E-AOSAS++ を対応づけることでアーキテクチャを構築した。構築したアーキテクチャをもとに考察をおこなった。

今後の課題としてステートマシン図から共通構造を自動生成する自動生成ツールの作成が挙げられる。

## 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, July/August.2002.