

コンテキストに基づくスマートデバイスの 動的機能変更に関する研究

2015SE011 八手幡拓巳 2015SE022 井上真吾

指導教員：沢田篤史

1 はじめに

スマートデバイスは持ち運びやすく、機能も豊富であることからさまざまな環境で利用されている。スマートデバイスにアプリケーションをインストールすることで、様々な機能が利用可能となる。使用するアプリケーションは持ち運んだ場所に応じて、必要となるアプリケーションが異なることがある。しかし、それぞれの場所に適したアプリケーションを瞬時に探して利用することは難しい。利用できる機能を必要な時だけ表示させることができれば、スマートデバイスがより便利になる。これを実現するためには、変化する位置情報や時間などの利用環境に基づくコンテキストに応じてスマートデバイスの機能を柔軟に制御するための枠組みが必要である。また、外部環境の変化に応じて制御をしたいのでコンテキスト指向による実現が最適である。

スマートデバイスの機能を必要な時だけ利用可能となるように制御するためには、位置情報や時間など複数のコンテキストを扱う必要がある。しかし、複数のコンテキストを扱ってプログラムを作成した場合、コンテキストに関する振舞いの記述が多くなるので膨大で複雑な記述になってしまう。変更を行う際、機能制御に関する記述がプログラム中に散らばると変更が困難である。また、個々のアプリケーションレベルで機能変更を行う試みが存在するが、場当たりのになりがちであり、コンテキストの追加や削除などに対する保守の面からは問題解決に至っていない。

本研究の目的は、スマートデバイスの機能を利用環境に応じて柔軟に制御することである。我々はコンテキストの変更を容易にするために、保守性を考慮したコンテキスト指向ソフトウェアアーキテクチャを定義することが重要であると考えた。記述が複雑になってしまう問題を解決するために、コンテキストに応じた振舞いを分離して記述できるように設計する。

このアーキテクチャを実現するために、江坂ら [3] が提案した自己適応のための PBR パターンを利用し、佐藤ら [4] のアーキテクチャを改良する。PBR パターンを適用することで機能変更に関する記述を分離し、保守性を考慮する。設計したアーキテクチャに基づいてプログラムを作成し、記述が簡便になっていることを検証する。また、すでに提案されている研究と比較して我々の提案したアーキテクチャの有用性を検証する。

2 背景技術

2.1 動的機能変更の必要性

スマートデバイスの利点により、多くのアプリケーションを扱う利用者が増えているが全ての機能を使いこな

している利用者は少ない。環境に応じて利用できる機能だけを表示することで利便性が高まる。例えば、飲食店に入店した時に使えるアプリケーションを自動で表示することは利用者にとって便利である。これを実現するためには、スマートデバイスの機能を動的に制御する必要がある。すでにスマートデバイスの機能を動的に制御し、表示を切り替えるための研究を佐藤らと青柳らが行っている。

2.2 関連研究

2.2.1 佐藤らの研究 [4]

佐藤らはスマートデバイスが利用されるさまざまな環境に応じたセキュリティの確保を課題として捉え、スマートデバイスの機能が自律的に制御されることがセキュリティ対策として重要であると考えた。そのために、佐藤らはネットワーク接続情報の変化でコンテンツの閲覧やアプリケーションの起動を制御するためのアーキテクチャを提案した。アプリケーションの機能制御の実現に向け、ID ベース暗号と呼ばれる ID 情報を公開鍵としてデータの暗号化を行う暗号技術を応用した。佐藤らが提案したシステムの構成は、機能制御クライアント、機能制御サーバ、鍵管理サーバの 3 者構成から成っている。これらを利用し、スマートデバイスの利用環境に応じてアプリケーションの起動可否を制御する。また、佐藤らはコンテンツも利用環境に応じて閲覧制御を行っている。

2.2.2 青柳らの研究 [1][2]

青柳らは佐藤らの研究を基に関数型暗号を応用することで、柔軟な制御を実現した。システムの構成は佐藤らの提案方式と大きく変わらない。

アプリケーションの起動制御を実現するために、サーバ側には機能制御ファイルの管理機能と復号鍵の生成機能、クライアント側にはアプリ制御機能と暗号化された機能制御ファイルの復号機能、利用環境の検知機能が必要である。そのために、佐藤らが提案した機能制御クライアントと機能制御連携アプリケーションを機能制御アプリケーション、コンテンツビューワアプリケーション、復号制御アプリケーションの 3 つに変更した。

機能制御アプリケーション

アプリケーションの起動可否の制御を実現するための常駐型のホームアプリケーションである。機能は主に 3 つあり、機能制御ファイルの取得・管理機能、アプリケーション制御状態の UI 表示機能、アプリケーション制御・状態通知機能である。

コンテンツビューワアプリケーション

コンテンツの閲覧制御を実施するアプリケーションである。機能は主に 2 つあり、コンテンツの取得・管

理機能，コンテンツ表示機能である．

復号制御アプリケーション

端末の利用環境に応じて，関数型暗号で暗号化されたデータを復号するアプリケーションである．機能は主に3つあり，暗号化データ復号機能，暗号化データ復号可否判定機能，環境情報管理機能である．

これらの機能によって，関数型暗号の手法が扱えるように適応させた．

青柳らはスマートデバイスが利用される環境を想定し，機能制御ファイルを導入した．

2.2.3 問題点

佐藤らの研究はスマートデバイスの機能を動的に制御しているが，ID ベース暗号の特性上，コンテキストを一つしか扱えないので利用環境の変化に対して柔軟な制御ができていない．また，コンテキストを変更したい場合，機能制御ファイルを含めた全てのシステムを書き換える必要がある．そのために，佐藤らの研究は保守性が低い．

青柳らは，関数型暗号を応用することで複数のコンテキストを利用し，柔軟な制御を実現しようとしたが，コンテキストの組合せが増えるたびに，コンテキストに関する振舞いの記述が膨大になり記述が複雑になってしまう．

3 コンテキストに基づくスマートデバイスの動的機能変更

3.1 動的機能変更実現のために利用する技術

3.1.1 PBR パターン [3]

PBR(Policy-Based Reconfiguration) パターンとは，江坂らが提案しているポリシーに基づいて静的および動的な再構成を行うコンテキスト指向とアスペクト指向を統一的に扱う自己適応のためのアーキテクチャパターンである．

3.2 アーキテクチャ設計

佐藤らが提案しているアーキテクチャを改良し，コンテキストに応じた振舞いをアスペクトとしたスマートデバイスの機能を動的に変更するアーキテクチャを提案する．本研究では，スマートデバイスの持ち運びやすいという特徴を重視し位置情報をコンテキストとし，更に利用者の生活環境に適した制御をするために時間帯もコンテキストとする．本研究で提案したアーキテクチャの静的構造と動的振舞いを図1，図2に示す．

保守性を考慮するために，PBRパターンを佐藤らのアーキテクチャに適用する．ポリシーと再構成の仕組みをそれぞれコンポーネント化し実現する．佐藤らのアーキテクチャでは1つのコンテキストを扱っているが，我々は柔軟な制御を目的としているので，複数のコンテキストを取得する．

アーキテクチャのそれぞれのコンポーネントについて説明する．また，コンテキスト指向に基づいてアーキテクチャを設計しているので，PBRパターンとの関係を示す．

機能変更ポリシー

コンテキストの変化により実行される．構成定義で

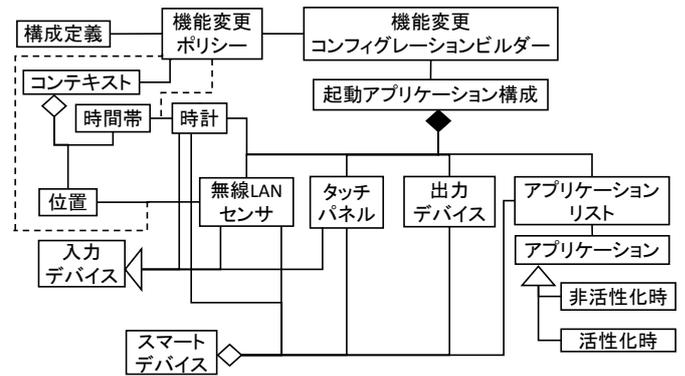


図1 保守性を考慮した動的機能変更を可能とするアーキテクチャ 静的構造

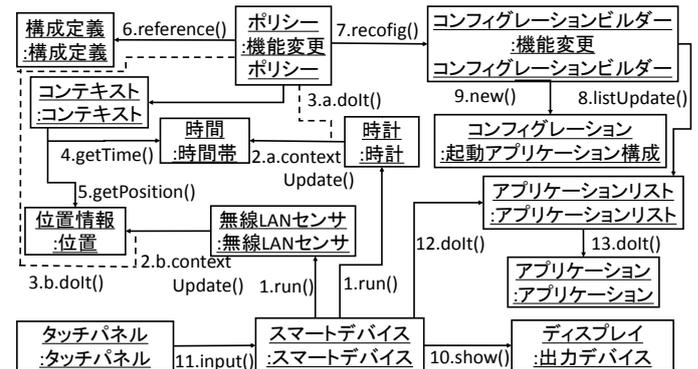


図2 保守性を考慮した動的機能変更を可能とするアーキテクチャ 動的振舞い

決められた機能制御ファイルであるアプリケーションリストを取得し復号を行う．その後，再構成を行うメッセージを機能変更コンフィグレーションビルダーに送信する．PBRパターンのPolicyに相当する．

機能変更コンフィグレーションビルダー

機能変更ポリシーから送られてきたメッセージを基に再構成を行う．PBRパターンのBehavior Activatorに相当する．

起動アプリケーション構成

各コンポーネントの情報を保持する．PBRパターンのConfigurationに相当する．

コンテキスト

has-a 関係になっているコンポーネントから値を取得．PBRパターンのContextに相当する．

時間帯

朝や昼など時間帯の情報を保持する．

位置

無線LANセンサクラスから取得したSSIDの情報を保持する．

構成定義

機能変更ポリシー起動時に，コンテキストの組合せからどのアプリケーションリストを取得するのかを決める．

スマートデバイス

スマートデバイスの動作について記述されている。スマートデバイスは時計、無線 LAN センサー、タッチパネル、出力デバイス、アプリケーションリストと has-a 関係である。

時計

現在の時刻を取得し続け時間帯を判別する。また、時間帯が変更したら時間帯に最新の時間帯の情報を送信する。
PBR パターンの Common Component に相当する。

無線 LAN センサ

接続されているアクセスポイントの SSID の情報を取得し、位置に最新の SSID の情報を送信する。また、アクセスポイントに接続されていない場合は、接続されていないという情報を送信する。
PBR パターンの Common Component に相当する。

タッチパネル

ディスプレイに表示されているアプリケーションのアイコンを押した時にメッセージを送信する。
PBR パターンの Common Component に相当する。

出力デバイス

ディスプレイに表示する。
PBR パターンの Common Component に相当する。

アプリケーションリスト

起動可能なアプリケーションと起動不可能なアプリケーションを持つ。PBR パターンの Context Component に相当する。

アプリケーション

アプリケーションの動作について記述されている。

図 2 を基に再構成の振舞いについて説明する。コンテキストが更新した時 (2.a, 2.b) に、機能変更ポリシーがメッセージを横取る (3.a, 3.b)。機能変更ポリシーの記述に従って機能変更コンフィグレーションビルダーを起動 (7) し、起動アプリケーション構成を再構成する (9)。それによりアプリケーションの起動を制御する。

機能変更ポリシーの詳細な振舞いとして、構成定義を起動 (6) することで、取得する機能制御ファイルを特定し、機能制御ファイルを取得する。その後、鍵管理サーバから復号鍵を取得し、復号処理を行う。

3.3 プロトタイプの実現

我々が提案したアーキテクチャに基づいてプロトタイプのプログラムを作成した。アプリケーションリストのフォーマットは、コンストラクタで制御対象のアプリケーションのインスタンス生成を行う、ここでアプリケーショ

ンの起動可否が決定される。メソッドでは、アイコンの表示とアプリケーションの実行について記述される。アプリケーションリストに関する記述を図 3 に示す。

```
public class Map1AppList extends AppList{
    private Application cameraApp, gameApp;

    public Map1AppList() {
        cameraApp = new ActiveApplication();
        gameApp = new ActiveApplication();
    }

    public void icon() {
        cameraApp.icon();
        gameApp.icon();
    }

    public void doIt(String msg) {
        if(msg=="cameraApp") {
            System.out.println(msg);
            cameraApp.doIt();
        }else if(msg=="gameApp") {
            System.out.println(msg);
            gameApp.doIt();
        }
    }
}
```

図 3 実行結果

振舞いとして、スマートデバイスが持つ時計と無線 LAN センサが一定時間ごとに実行を行い、コンテキストを更新する。それによって機能変更ポリシーが起動される。機能変更ポリシーは構成定義を起動させ、構成定義の中で使用するアプリケーションリストを決定する。その後、機能変更ポリシーが機能変更コンフィグレーションビルダーを起動する。機能変更コンフィグレーションビルダーは、起動アプリケーション構成の情報を取得する。使用するアプリケーションリストを起動アプリケーション構成に適用した後、再構成を行う。これにより、アプリケーションの動的機能制御を行っている。プロトタイプの実行結果を図 4 に示す。

```
時間帯 : evening
SSID : 2
実行不可能アイコン
実行不可能アイコン
cameraApp
実行できません
gameApp
実行できません
時間帯 : evening
SSID : 1
実行可能アイコン
実行可能アイコン
cameraApp
実行
gameApp
実行
時間帯 : night
SSID : 2
実行可能アイコン
実行不可能アイコン
cameraApp
実行
gameApp
実行できません
```

図 4 実行結果

アプリケーションリストには、cameraApp と gameApp の 2 つのアプリケーション起動可否の情報が記述されているものとする。コンテキストである時間帯が evening、位置である SSID が 2 の場合、2 つのアプリケーションは実行不可となっている。1 回目の更新は、SSID が更新さ

れた場合を想定している．SSID が 2 から 1 に更新されると、コンテキストの組合せから 2 つのアプリケーションが実行可能と記述されたアプリケーションリストに再構成される．2 回目の更新は、位置と時間帯が同時に更新された場合を想定している．SSID が 1 から 2、時間帯が evening から night に更新されると、cameraApp は実行可能、gameApp は実行不可と記述されたアプリケーションリストに再構成される．また、利用者がアプリケーション起動可否の情報を瞬時に把握するために、アプリケーションのアイコンの表示を切り替える．コンテキストの値とその場合に起動できるアプリケーションの組合せを図 5 に示す．

	SSID1	SSID2	SSID3	SSID4	SSID5
morning	cameraApp 可能	cameraApp 不可能 gameApp 可能			
noon	gameApp 可能	cameraApp 可能 gameApp 不可能			
evening		cameraApp 不可能 gameApp 不可能			
night		cameraApp 可能 gameApp 不可能			
midnight		cameraApp 不可能 gameApp 不可能	cameraApp 可能 gameApp 不可能		

図 5 コンテキストの値とその場合に起動できるアプリケーションの組合せ

4 考察

4.1 プロトタイプの評価

PBR パターンを利用したアーキテクチャに基づいてプログラムを記述した結果、機能変更ポリシーと構成定義の記述を変更するだけでコンテキストに関する振舞いを変更できるようになり、コンテキストの追加、削除が容易になった．

我々が設計したアーキテクチャに基づいて記述するプログラムはコンポーネント間の関係が明確であり、コンテキストに関する振舞いの記述が分離されている．そのために、コンテキストの変更を行う場合、変更場所が明確なので変更しやすく保守性が高い．

コンテキストの数が増えると構成定義の記述が複雑になる．それぞれの利用者に適したコンテキストのみで構成定義を記述することで、コンテキストの組合せが膨大になることを抑えることができる．今後、構成定義に記述の保守性について考察し、改善する必要がある．

4.2 関連研究との比較

本研究をアーキテクチャと保守性の観点から青柳らの研究と比較する．

アーキテクチャの観点から比較すると青柳らは暗号化手法に適したシステム構成図を設計している．我々はスマートデバイスの機能制御のためのアーキテクチャを設計した．我々の提案したアーキテクチャの機能変更ポリ

シーを変更するだけで、青柳らの暗号化手法を用いた機能制御も行える．

保守性の観点から比較すると青柳らはコンテキストの組合せが増えるたびに、コンテキストに関する振舞いの記述が膨大となり、記述が複雑になるので、コンテキストの変更が容易でなく保守性が低い．本研究は機能変更ポリシーと構成定義の記述を変更するだけでコンテキストに関する振舞いを変更できるので、保守性が高い．

5 おわりに

スマートデバイスは持ち運びやすく機能も豊富であるという利点からさまざまな環境で利用されている．しかし、それぞれの場所に適したアプリケーションを瞬時に利用することは難しい．我々は利用できる機能を必要な時だけ表示させることが利用者にとって便利であると考えた．これを実現するには、変化する位置情報や時間などの利用環境に応じてスマートデバイスの機能を柔軟に制御することが重要である．スマートデバイスの機能を柔軟に制御するためには、複数のコンテキストを扱う必要がある．しかし、コンテキストに関する振舞いの記述が増えると膨大で複雑な記述になってしまう問題がある．

本研究は、スマートデバイスの機能を利用環境に応じて柔軟に制御するために、保守性を考慮したスマートデバイスの動的機能変更を可能とするコンテキスト指向ソフトウェアアーキテクチャを定義した．その際、保守性を考慮するために、PBR パターンを利用し、佐藤らのアーキテクチャを改良した．その後、プロトタイププログラムを作成した．結果として、本研究をアーキテクチャと保守性の観点から青柳らの研究と比較することでアーキテクチャの有用性と保守性が高いことを示した．

今後の課題として、スマートデバイスに我々の提案したアーキテクチャに基づいて記述されたプログラムを導入し、実際に稼働させ、事例検証を行う必要がある．

参考文献

- [1] 青柳真紀子, 知加良盛, 伊坂広明: 関数型暗号を適用したスマートデバイス機能制御システムにおける暗号化条件最適化の検討, コンピュータセキュリティシンポジウム 2014 論文集, Vol. 2014, No. 2(2014), pp. 1089-1096.
- [2] 青柳真紀子, 知加良盛, 伊坂広明, 栢口茂: 関数型暗号を適用したスマートデバイス機能制御の実装, コンピュータセキュリティシンポジウム 2014 論文集, Vol. 2014, No. 1(2014), pp. 758-764.
- [3] 江坂篤侍, 野呂昌満, 沢田篤史: インタラクティブシステムのための共通アーキテクチャの設計, コンピュータソフトウェア, Vol. 35, No. 4(2018), pp. 3-15.
- [4] 佐藤亮太, 知加良盛, 奥田哲矢, 栢口茂: スマートデバイスにおける利用環境に応じた機能制御機構の提案とその考察, 情報処理学会論文誌, Vol. 55, No. 1(2014), pp. 267-279.