

自動販売機制御ソフトウェアの再開発

～自動販売機制御ソフトウェアのログ解析支援ツールのアーキテクチャ設計～

M2007MM011 加藤大地

指導教員：野呂昌満

1 はじめに

ソフトウェアの信頼性を保つことは、リリース時点だけでなく保守段階でも非常に重要であり、それは組込みソフトウェアも例外ではない。しかし、組込みソフトウェアはハードウェアと組み合わせることで製品として成立するので、一般的なソフトウェア開発にはない独特の制約が存在し、保守にも影響を及ぼす。組込みソフトウェア開発の保守における制約としては、常時監視することが困難である点、ハードウェアとの関連が強く製品の表面的な情報だけで製品内部の詳細な動作を判断することが困難である点が挙げられる。

以上のような制約が存在する組込みソフトウェア開発の保守では、ログを用いたソフトウェアの動作解析が、製品の動作を把握するための手段として広く用いられている。ソフトウェアログの解析には、多くの工数を要するという問題が存在する。ソフトウェアログの解析は、ログデータをもとにソフトウェアやハードウェアの動作や状態をトレースするので、ログデータとして表現する情報は多種多様である。製品毎にデータの種類や解析のための手法が存在し、系統的にログデータを解析する方法が確立されていないことから、自動化が困難である。これらの要因から、開発の現場におけるログの解析は、多種多様かつ大量のログを開発者が直接目で見て行わざるを得ない状況になっている。さらに、近年のソフトウェアは変更が頻繁に行われ、ログデータにも変更が及ぶ。このことが解析に要する工数の増大に拍車をかけている。ソフトウェアログの解析における問題の解決策として、ログ解析支援ツールの開発が挙げられる。

課題の背景からツールには次の2点が必要な条件となる。

- 解析手法やアルゴリズムに適応的であること
- ログや解析のための情報、対象のログをになどの取り扱うデータに柔軟であること

以上の条件を満たすためには、再利用性、柔軟性に富んだログ解析支援ツールの開発を行う必要がある。近年、再利用性、柔軟性に富んだソフトウェアを開発する開発方法論としてPLSE (ProductLineSoftwareEngineering)[1]が提案されている。PLSEを適用したプロセスで中心とするアーキテクチャの構築に、アスペクト指向[2]を用いるというアプローチで開発を行うことで、より再利用性、柔軟性に富んだログ解析支援ツールの開発を行い、問題の解決を図る。

1.1 本研究のメタ技術

本研究でのメタ技術はPLSEである。本研究ではPLSEの適用にあたってアーキテクチャの構築プロセスモデルを作成し、これを中心として開発を進めた。図1に作成

したプロセスモデルを示す。

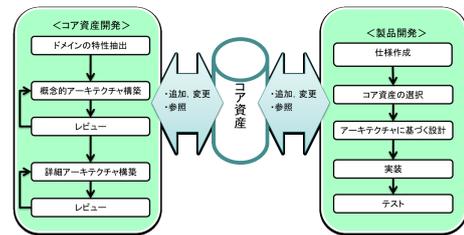


図1 PLSEに基づくアーキテクチャ構築プロセスモデル

1.2 目的

本研究の目的は、ログ解析支援ツールの開発を題材にアスペクト指向アーキテクチャを用いたPLSEに基づく開発プロセスを実施し、本研究の開発プロセスが課題解決に有用であることを検証することである。

1.3 関連研究と本研究の比較

文献[3]では正規表現を用いてログのパターンを定義し、パターンマッチングを行うことでログデータを検索するというツールを開発している。文献[4]では出力されるログデータを内部系に変換し、統計的に視覚化するツールを開発している。文献[3]と文献[4]では共に、解析手法をアダプタブルに実現するための開発を行っていないので、他の手法をツール化するためには、別のツールとして新たに開発を行わなければならない。

本研究では、ソフトウェアのログの解析にはどのような方法があるかを視点で整理し、ツール化することで、解析手法についてアダプタブルなツールとして実現することを可能とした。

1.4 OJL

本研究は、文部科学省の事業の一つである先導的ITスペシャリスト育成推進プログラム「OJLによる最先端技術適応能力を持つIT人材育成拠点の形成」としておこなった。我々は、OJLとして富士電機リテイルシステムズ株式会社と連携し、自動販売機制御ソフトウェアのログ解析支援ツールの開発をおこなった。

2 研究対象

本研究では、組込みソフトウェアの典型的な例である自動販売機制御ソフトウェアを研究対象とする。

協力企業では、ログ解析にログデータの多さとソフトウェアの頻繁な更新が原因で、多くの工数がかかるという問題を抱えている。協力企業のログ解析では、4バイト単位のログデータを表計算ソフトのマクロ機能でアセンブリ

相当の自然言語に変換し、ログの可読性を向上させているが、開発者が直接ログデータを見ることには変わりがないので大きな工数の削減はできていない。図2にログの例を示す。

00	10	23	71		コイン投入	}	販売動作
A1	3F	01	00		搬出		
4F	3E	05	A1		返金	}	故障動作
A1	6b	3c	24		紙幣投入		
2a	91	77	01		返金		
11	00	01	11		故障コード		
....		

図2 ログの例

協力企業におけるログ解析の問題は、前章で述べたソフトウェアにおけるログ解析の問題と同様である。

3 ログ解析支援ツールの概要

本研究で開発したログ解析支援ツールでは、主にログの抽出と加工の機能を実現している。抽出機能は、時間やログの種類等を指定し、抽出や除去を行うことで解析に必要な情報の選別を行う。加工機能は、指定した種類のログデータに色づけを行ったり、値の集計などを行うことにより、ログデータに情報を付加する。さらにそれぞれの設定をマクロとして保存することが可能である。図3にツールのイメージを示す。

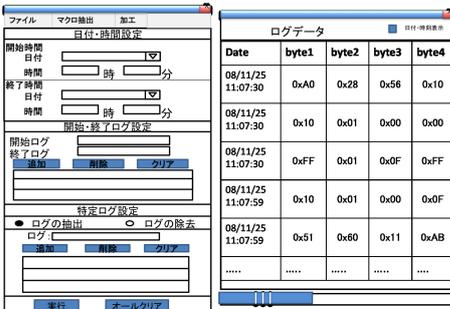


図3 ツールのイメージ

4 ログ解析支援ツール開発のためのプロセス

本研究では作成したプロセスモデルに基づき、ドメイン工学におけるプロセスを構築し、実施した。このプロセスではほとんどのプロセスでコア資産に対しての参照、追加、変更を考慮して行う。構築したプロセスを図4に示す。本研究では、概念的アーキテクチャ構築までを実施し、現在はドメインテスト開発の途中である。概念的アスペクト指向アーキテクチャは開発全体で中心となる参照用のアーキテクチャのことをあらわす。

4.1 コア資産

本研究ではPLSEにおけるコア資産は次のようなものがあると考えた。

- 対象製品（ドメイン）の特徴
- 概念的なアーキテクチャ

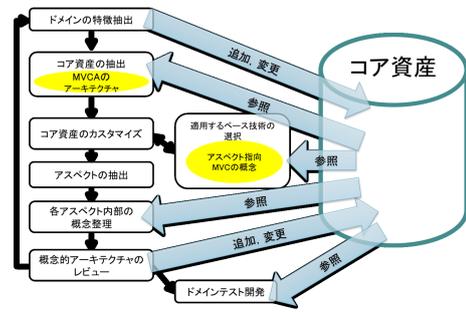


図4 開発プロセス

- 適用するベース技術のノウハウ

4.2 特徴抽出

開発するログ解析支援ツールの特徴は次の4点であると考えた。

- GUIを用いる
- 画面構成がツールの構成の中心
- データとデータに対する処理が様々
- 対象となるログデータが変更されやすい

4.3 概念的アスペクト指向アーキテクチャの構築

GUIを用いるツールであることから、前期OJLで作成した自動販売機ディスプレイ制御ソフトウェアのアーキテクチャであるMVCAのアーキテクチャをコア資産として抽出し参照した。さらに開発するツールの特徴から適用するベース技術と適用方法を考えることでコア資産をカスタマイズし、概念的アスペクト指向アーキテクチャの構築を行った。次に特徴と適用したベース技術について示す。

- 画面構成がツールの構成の中心であるので画面構成をコアコンサーンとするアーキテクチャを構築（アスペクト指向）
- 取り扱うデータとデータに対する処理が様々であることから、MVCアーキテクチャのVIEW内部にMODELのコンサーンの確認（MVCアーキテクチャ[5]、アスペクト指向）
- 対象となるデータが変更されやすいことから柔軟性と再利用性が高いアーキテクチャ（アスペクト指向ソフトウェアアーキテクチャ）

4.4 概念的アスペクト指向アーキテクチャ

概念的アスペクト指向アーキテクチャを構築した。図5に示す。ディスプレイ制御ソフトウェアにおいてMVCAアーキテクチャを参考にツールにおけるコンサーンを分離した。ツールの中心である画面構成をあらわすVIEWMODELアスペクト、ツールの見た目に関する処理をあらわすVIEWアスペクト、入力管理するCONTROLLERアスペクト、アプリケーション固有の処理をあらわすMODELアスペクトで構成されている。それぞれのアスペクトはアスペクト間記述でつながっている。各アスペクトは独立しているため、アスペクト内ではそれぞれのコンサー

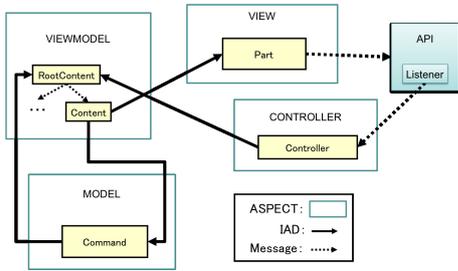


図5 概念的アスペクト指向アーキテクチャ

ンに適した手法を実現することが可能になる。各アスペクト内部の概念を整理したことによって、VIEWMODELアスペクトではツールの表示部品の構成が木構造が適用していることや、MODELアスペクトではログデータを中心して様々な処理が行われることから、処理を手続きとしてモジュール化することができると思えることができた。

5 ログ解析手法の整理

概念的アスペクト指向アーキテクチャを構築した結果として、ツールの構造とログ解析手法を独立して考えることが可能となった。特にMODELアスペクトはログ解析の手法をモジュール化する部分なので、アスペクト内でさらに詳細に概念を整理する必要がある。そこで、ログ解析の手法の整理を行った。

5.1 ログ解析手法

協力企業におけるログ解析は次の手順で行われる。

1. バイト列のログデータを表計算ソフトに取り込みアセンブリ相当の自然言語に変換
2. 不具合の原因を発見するために、表計算ソフトに取り込んだデータに対して操作を加える。具体的な操作の代表的なものは次に挙げる通りである。
 - 特定の種類や範囲のログデータの切り出し
 - 探している原因に関係の無いログデータの削除
 - 目的のログデータの検索
 - ログデータの種類別にセルに色付け

3. 操作後のデータを開発者が直接見て不具合を特定する。特定できなければ2.に戻る

以上より、ログ解析はログデータに対して、「扱いやすい形式への変換」、「必要に応じての切り出しや削除」、「情報の付加」という大きく3つの手順で行われることが分かった。

5.2 ログ解析の処理

ログ解析はどのように行われるかを考察した結果、ツール特有の処理として「データ変換」、「抽出」、「加工」の3つに分類できると考えた。これらの処理は共通のログデータに対して行われる。

- データ変換
実際のログデータとログデータの形式の読み込み、実際のログデータをツールの内部形式に変換する。

- 抽出

大量のログデータの中から必要なデータを抽出することで、可視性を向上させる。必要なデータを抽出するために、時間やログの種類などの条件を指定に対し、抽出と除去を行う。

- 加工

色づけなどをして表示しているログデータの表現で情報を拡張させる。

6 考察

次の4点について考察をおこない今回のアプローチについて整理する。

- 概念的アスペクト指向アーキテクチャをPLSEに基づく開発に用いた場合の利点
- アーキテクチャを中心とした開発プロセスの利点
- ログ解析手法の整理
- 問題解決のアプローチとしての妥当性

6.1 概念的アスペクト指向アーキテクチャをPLSEに基づく開発に用いた場合の利点

本研究で構築した概念的アスペクト指向アーキテクチャでは、概念が分離してあることによって、変更要求とアーキテクチャの対応関係が明確化できるという利点がある。次に例を示す。

- 部品の配置（構成）を変更したい
→ VIEWMODELアスペクトを中心に変更
- 部品のサイズを変更したい
→ VIEWアスペクトを中心に変更
- 解析手法を追加したい
→ MODELアスペクトを中心に変更

以上のようにアスペクト指向でアーキテクチャを構築し、コンサーンの抽出を行ったことによって、ツールに対する変更要求とアーキテクチャの対応関係が明確化できた。PLSEに基づいて開発を行う際に重要であるトレーサビリティを向上させることができたという点において、PLSEに基づく開発を行う際に中心とするアーキテクチャはアスペクト指向を用いて構築することが有用であると言える。

6.2 アーキテクチャを中心とした開発プロセスの利点

アーキテクチャを中心とした開発プロセスを実施したことで、開発の初期の段階で開発対象に対する認識を統一することができた。結果として、協力企業や開発者同士の認識のずれによる手戻りが発生しなかった。アスペクト指向でアーキテクチャを構築したことにより、各アスペクトを個別に開発することが可能になり、開発者の割当が容易に行え、進捗管理の見通しが良くなった。開発プロセスとして、手戻りの発生の回避や進捗管理の容易性が利点であるといえる。さらに概念的にアーキテクチャを整理した結果、ログ解析手法はMODELアスペクトに集約されることが分かったため、ツール全体のアーキテクチャから独立して解析手法を概念的に整理した上で、設計に反映することができた。

6.3 ログ解析手法の整理

本研究におけるログ解析手法は、自動販売機制御ソフトウェアにおけるログに対するものであったが、「データ変換」、「抽出」、「加工」とドメインとは独立した基準で分類、整理を行えた。「データ変換」、「抽出」、「加工」は組込みソフトウェア以外のサーバプログラム等のログ解析にも同様の基準で分類、整理が行えるといえる。

自動販売機のログデータ以外のログの例を示すことで、他のログデータへの応用可能性を考察する。ソフトウェアログとして代表的な Web サーバ (apache[6]) のアクセスログの一部を挙げる。次にログデータの項目の例を挙げる。これらの項目は値の列として、コマンドで区切られた形で出力される。

- 日付, 時刻
- クライアントの IP アドレス
- リクエストされたファイル
- 処理時間
- HTTP ステータスコード

次に解析のケースを挙げる。

1. エラーのステータスコードを含むログだけを出力させたい
→ステータスコードの種類を指定して抽出 (抽出の概念)
2. 特定の時間帯のアクセス数を数えたい
→時刻の範囲を指定で抽出 (抽出の概念) + 件数を集計 (加工の概念)

以上のより他の種類のログデータにも対応可能な概念であることがいえる。

6.4 問題解決アプローチとしての妥当性

本研究ではソフトウェアのログ解析における問題の解決アプローチの1つとしてアスペクト指向アーキテクチャを中心としたログ解析支援ツールの開発を行った。上記2つの考察点により、我々の行った開発は、ソフトウェアのログ解析における課題であるツール化、ログ解析手法の整理を実際に解決することができるアプローチであると言える。

7 おわりに

本研究では、ログ解析手法の整理とログ解析ツールの開発を行った。モジュールの独立性が高いアスペクト指向ソフトウェアアーキテクチャに基づきツールの開発を行い、考察することで、我々の実施した開発プロセスの有用性を示すことができた。

7.1 本研究の OJL としての位置づけ

本研究の OJL としての位置づけは、協力企業におけるログ解析の問題をツールの開発によって解決することである。

7.1.1 適用したベース技術

本 OJL で適用したベース技術を以下に述べる。

- ソフトウェアアーキテクチャ技術
 - ・アスペクト指向ソフトウェアアーキテクチャ
 - コンサーンの分離基準に利用
 - ・MVC アーキテクチャ
 - ツールにおけるベースのアーキテクチャとして利用

7.1.2 メタ技術

本研究の OJL としてのメタ技術は PLSE である。研究として中心となるプロセスモデルを作成し、それに基づいた具体的な開発プロセスを構築し、実施した。

7.2 今後の課題

今後の課題として次の3点を考える。

- 他の製品やドメインへの適用
本研究では、対象を自動販売機制御ソフトウェアとしたが、その他の組込ソフトウェアやドメインへの適用を行うことで、本研究で提案したアプローチが適用可能であるかを検証する必要がある。
- ドメインテスト開発の完了
現在、ログ解析支援ツールの開発はドメインテスト開発の途中段階である。アプローチの妥当性を完全に示すには開発をテスト終了の段階まで進め、有用性を示す必要がある。
- 次期開発においてのコア資産の有用性の検証
今回の開発は、ドメインとしては1度目の開発であったので、次期開発においてのコア資産の有用性は示せていない。同じドメインでの開発の際には今回構築したコア資産を利用することによる有用性を示す必要がある。

参考文献

- [1] L.M.Northrop “SEI’s Software Product Line Tenets,” ,IEEE Software,Vol.19 No.4, pp.32-40,2007.
- [2] M.Noro,A.Sawada,Y.Hachisu,and M.Banno “E-AoSAS++ and its Software Development Environment” *Proceedings of the 14th Asia-Pasific Software Engineering Conference(APSEC2007)*, pp.206-213,Dec.2007.
- [3] 森本亮太,若林隆行,高田広章 “リアルタイムシステムのための正規表現を用いたログ検索システムの構築とその評価”, 情報処理学会研究報告.SLDM, システム LSI 設計技術,vol.2002,No.18,pp9-15,2002.
- [4] 久野啓,大澤範高,弓場敏嗣 “並列プログラムのための視覚的性能デバッグ”, 電子情報通信学会技術研究報告.CPSY, コンピュータシステム,vol.94,No.383,pp.65-71.1994.
- [5] F.Buschmann,R.Meunier,H.Rohnert,P.Sommerlad, M.Stal “Pattern-Oriented Software Architecture Volume 1: A System of Patterns”,*John Wiley & Sons*,1997.
- [6] Apache <http://apache.org/>