

科学研究費助成事業 研究成果報告書

平成 27 年 6 月 12 日現在

機関番号：37111

研究種目：基盤研究(C)

研究期間：2012～2014

課題番号：24500042

研究課題名(和文) プロダクトライン開発の派生開発からの導入と適用範囲拡大に関する研究

研究課題名(英文) A Study on Introducing Software Product Line through Derivative Development and Extending Its Scope

研究代表者

中西 恒夫 (NAKANISHI, Tsuneo)

福岡大学・工学部・教授

研究者番号：70311785

交付決定額(研究期間全体)：(直接経費) 4,100,000円

研究成果の概要(和文)：組込みシステムの開発のほとんどは派生製品開発の連続となっている。このような派生製品を効率的に開発する方法論としてプロダクトライン開発と派生開発が知られている。前者は軌道に乗れば品質、コスト、納期の大きな改善が期待できる反面、導入障壁が高い。後者は導入障壁は低いが、ソフトウェアの複雑さの増大やアーキテクチャの崩壊に対応できる技術ではない。本研究では、導入障壁の低い派生開発からプロダクトライン開発に移行するプロセスを定義するとともに、その過程における不確実性を管理するフレームワークを提唱した。

研究成果の概要(英文)：Derivative products have been developed in the development site of embedded systems. Software product line and derivative development process have been expected to develop derivative products efficiently. Software product line improves quality, cost, and lead time drastically if it is introduced successfully; however, its adoption brings many challenges. On the other hand, derivative development process has a lower adoption barrier, while it never guarantees to prevent complexity of software from increasing and software architecture from corrupting. This research defines a process to migrate from derivative development process with a lower adoption barrier to software product line gradually and provides a framework to manage uncertainty in that process.

研究分野：ソフトウェア工学

キーワード：プロダクトライン 派生開発 導入障壁 不確実性

1. 研究開始当初の背景

(1) 今日の組込みシステム開発現場では、ほとんどの開発は既存製品に対して機能の追加や変更を施す、派生製品の開発となっており、スクラッチから製品を開発することはあまりない。派生製品開発における品質、コスト、工期の改善ではソフトウェア資産の再利用が鍵となる。

(2) ソフトウェア資産を製品間で共有、再利用するパラダイムとしてソフトウェアプロダクトライン(SPL: Software Product Line)が知られている。SPLは米国、欧州において1990年代後半から研究、実践され、その理論はほぼ完成されていると考えてよい。SPLは我が国においても2000年代後半から産業界において注目されるようになった。我が国の企業は世界中に膨大な種類の製品を供給し、当然 SPL へのニーズは高いはずであるが、現場における実践例はあまり報告されていない。SPLを実施するには、既存ソフトウェア資産の全体理解、現在ならびに将来の製品群の共通性と相違性の分析、ソフトウェアアーキテクチャの定義、ソフトウェアアーキテクチャで使用されるコンポーネントの開発と管理、開発組織とプロセスの変更が遅かれ早かれ必要であり、SPLの導入障壁は高いと言わざるを得ない。さらに現場でのリスクテークがし難い我が国の大企業の組織的風土もあり、SPLに興味を持ち、またその利点を理解しつつも導入に及び腰となっている企業が多い状況であった。

(3) 一方、既存製品に対する追加ならびに変更の要求に始まり、それら要求を充たすべく既存ソフトウェア資産の関連箇所を集中的に調査し、設計ならびに実装の追加/変更を進めることで派生製品を開発する、派生開発方法論(XDDP: eXtreme Derivative Development Process)の実践例が自動車産業をはじめとする我が国の産業界から昨今多く報告されていた。XDDPは、SPLと目的こそ共通しているものの、ソフトウェア資産の全体理解を必要とせず、またアーキテクチャの概念は希薄であり、コア資産の概念も持たず、SPLとは全く異なる開発プロセスである。(さらに言えば、SPLは開発プロセスというよりも、開発プロセスを律するパラダイムである。)既存ソフトウェア資産の全体理解を必ずしも必要としないXDDPは、明らかにSPLよりも低いコスト、短い工期での派生製品の開発を可能にする。しかしながら、XDDPは度重なる派生開発による、ソフトウェアの複雑さの増大やアーキテクチャの崩壊に対応できる技術ではない。長期に渡って派生製品が開発され続けるのであれば、SPLを導入し、プロダクトラインの製品間で共通アーキテクチャを定め、コア資産を整備していくほうが確実に品質、コスト、納期の改善を達成できる。

(4) 本研究申請当初、(研究代表者を含む)我が国のSPLのコミュニティでは、SPLと

XDDPとどちらがよりよく派生製品の開発に向いているのか、それぞれの共通点と相違点は何なのか、それぞれの長所と短所は何なのかという議論がなされるようになり、またXDDPのコミュニティとの技術的交流もなされるようになった。SPL導入の通過点にXDDPを位置づける発想は、このような背景のもと、当然に出てくるものであった。

2. 研究の目的

上述の背景を踏まえて、本研究では、まずXDDPからSPLへ移行する方法論の確立することとした。

研究立案当初、XDDPとSPLのもっとも大きな違いは、XDDPが既存製品のソフトウェア資産に対する追加/変更部分に関係する部分の理解に集中し既存ソフトウェア資産の全体理解を必ずしも求めないのに対して、SPLはそもそもアーキテクチャ指向のパラダイムでありソフトウェア資産の全体理解を求めることにあるように思われた。また、それがSPLの導入障壁を高め、開発現場へのSPLの普及を困難にしている主たる要因に思われた。そのため、XDDPによる派生開発を繰り返す中で、既存ソフトウェア資産に対する理解を深め、SPLの部分適用を経て全体適用へ拡大する方法論の確立を目指すこととした。

3. 研究の方法

(1) 上述の通り、XDDPは基本的に変更駆動型、擦り合わせ開発指向、部分理解容認型のボトムアップ型の開発方法論であるのに対して、SPLは計画駆動型、組み合わせ指向、全体理解前提型のトップダウン型のパラダイムである。XDDPからSPLへ移行するには、変更駆動から計画駆動へ、擦り合わせ開発から組み合わせ開発へ、部分理解から全体理解への転換を図る必要がある。

(2) SPLでは、プロダクトラインの全体像の理解と記述の手段として、フィーチャモデリングが半ばデファクトスタンダードに行われている。また、コア資産の管理はフィーチャからそのフィーチャの実現に寄与するコンポーネントを追跡可能とすることによって、すなわちフィーチャを「索引語」として用いることによって行われている。そこで、本研究では、既存ソフトウェア資産に対する理解、ならびに既存ソフトウェア資産のコア資産化をフィーチャによって管理することとした。

4. 研究成果

(1) XDDPによる派生製品の開発を繰り返す中で既存ソフトウェア資産に対する理解を深め、アーキテクチャの定義とコア資産の発掘を進め、SPLの部分適用に移行し、さらには全体適用に拡大していくための開発プロセス、XDDP4SPLを定義した。

XDDP4SPLの発想そのものについては、

派生開発の自然な延長となっており、既存の SPL や派生開発の方法論や技術を大きく塗り替えるようなことはしていない。むしろ、提案プロセスが開発現場に受け入れられるためにはそのようなことはしてはならない。本研究成果の真価は、XDDP4SPL から SPL への移行プロセスのひな形を与え、同プロセスの各工程における課題を整理したことにある。

(2) XDDP4SPL のプロセスでは、フィーチャ概念が移行管理の重要な役割を担う。

フィーチャモデリングは SPL の実施には必須の工程であるが、大規模、複雑な既存システムを理解しフィーチャを見出し整理するには相当の時間を要する。そこで、まずは XDDP による派生開発の際に行う変更要求、変更仕様の記述、ならびに既存コードの局所的リバースエンジニアリングの結果に基づいて局所的なボトムアップ型のフィーチャモデリングを実施することとした。一方で、既存ソフトウェア資産を調査することなく、ドメイン知識に基づいてトップダウン型のフィーチャモデリングを実施することも可能である。本研究では、ボトムアップ型フィーチャモデリングの結果見出されたフィーチャを局所識別フィーチャ、トップダウン型フィーチャモデリングの結果見出されたフィーチャを大域識別フィーチャと呼ぶ。局所識別フィーチャは既存ソフトウェア資産による裏付けは得られているものの、プロダクトライン全体の中での位置づけは明らかになっていないフィーチャである。派生製品開発のたびに既存ソフトウェア資産に対して理解の及ぶ範囲が広がり、局所フィーチャモデルの統合が進んでいく。局所識別フィーチャに関わる部分のアーキテクチャが明らかになれば、当該フィーチャは大域識別フィーチャに昇格し、当該フィーチャを含む局所フィーチャモデルが大域フィーチャモデルに組み込まれるようにした。

これまでフィーチャを、製品間での再利用性や機能、重要度、ステークホルダ等の観点で分類することは SPL のコミュニティにおいて一般的に行われてきたが、上述のように全体理解、部分理解のいずれに基づいて見出されたかを明らかにするために分類することは行われてこなかった。本研究では、派生開発から SPL への移行、あるいは SPL の部分的適用に資するべく、フィーチャを大域識別、局所識別の観点で分類することを提案したが、これは新規性の高い研究成果であると考えている。

(3) 既存ソフトウェア資産に関する理解の状況をフィーチャ単位で管理するのみならず、既存ソフトウェア資産からのコア資産発掘もフィーチャ単位で管理するようにした。

SPL 開発への移行の途上においてはプロダクトライン開発への移行、つまりはコア資産の蓄積の現況把握が欠かせない。

この目的のために本研究では、コア資産管

理のために階層化トレーサビリティマトリックスを導入することとした。トレーサビリティマトリックスはフィーチャの実現にどの既存ソフトウェア資産が寄与しているかを示す表である。階層化トレーサビリティマトリックスではソフトウェア資産の階層構造が表に反映されており、フィーチャから異なる抽象度のソフトウェア資産へのトレーサビリティが記述できる。そのため、既存ソフトウェア資産に対する理解が進んでいない段階では高い抽象度での粗いトレーサビリティを、理解が進むにつれて低い抽象度での精密なトレーサビリティを記述することが可能となる。

一般のトレーサビリティマトリックスは単に、各フィーチャの実現に各ソフトウェア資産が寄与しているか否かを記録するのみであるが、階層化トレーサビリティマトリックスでは、各フィーチャの実現にソフトウェア資産の全体が寄与しているのか、部分が寄与しているのか、全く寄与していないのか、寄与しているかどうか不明なのかを区別して記述する。フィーチャの実現にソフトウェア資産が寄与しているか不明であることを明示的に記述することにより、既存ソフトウェア資産の理解の程度が一目瞭然で把握できるようになる。

既存ソフトウェア資産の理解が進めば、それにあわせてフィーチャモデルも変更される。フィーチャの追加、削除、改名はもちろん、フィーチャそのものは変わらなくとも、その意味が追加、削除されることが頻繁に生じる。フィーチャモデルの変更にあわせて階層化トレーサビリティマトリックスの変更が必要となる。本研究では、フィーチャモデルの変化に伴う階層化トレーサビリティマトリックスの保守の手順も述べた。

トレーサビリティマトリックスの階層化そのものは当業者であれば自然に考案する発想である。しかしながら、既存ソフトウェア資産の理解が及んでいないところを明示的に記述し「見える化」することは、本研究における重要な主張である。また、フィーチャモデルの変更、特にフィーチャの意味変化による階層化トレーサビリティマトリックスの変更プロセスは、新規性の高い研究成果と考えている。形式仕様記述を導入し、フィーチャの意味を形式記述することによって、この成果をさらに理論的に発展させることが期待できる。

(4) 当初計画されていなかった派生的な研究成果であるが、本研究の実施過程において、システムの開発過程における不確実性を取り扱う概念的フレームワークが着想された。

あらゆるシステムの開発活動は相互に関連し束縛しあう意思決定の連続である。不確実性は後工程での仕様変更、すなわち手戻りのリスクとなるので除去されるべきものであるが、システムの開発や運用において予測できないことがある以上は不確実性を避け

ることはできない。システム開発の停滞，遅延を避けるためには不確実性と共存しつつ開発を進めていくよりほかはない。本研究では，システムの開発や運用における不確実性に関する諸概念を整理し，不確実性を扱うためのフレームワークを定義した。具体的には，不確実性の定義ならびに分類（外部不確実性，内部不確実性；一次不確実性，二次不確実性；多肢選択型，パラメータ型，複合型）を与えた。また，不確実性を決定木として体系化してモデリングする方法論を示し，さらに半世紀近くにわたって，製造現場で使われてきた FMEA (Failure Modes and Effects Analysis) を拡張し，不確実性の影響評価，解決策の検討，不確実性から開発資産へのトレーサビリティ確保を行うプロセスを示した。本研究では，ケーススタディの一環として農業用自律走行車両の開発を題材に選択していたが，当該車両のモータ制御システムを題材に，同フレームワークの適用を行った。

本研究成果は，ソフトウェアのみならず，あらゆるシステムの開発に適用できるものであり，特に機械・電気分野ではなじみの深い FMEA を拡張していることから，今後，システム工学分野に発展させていけるものと考えており，またそうする予定である。

(5) 派生開発から SPL への移行もまた，不確実性のある中での意思決定の連続である。研究計画当初は考えていなかったことであるが，最終年度には，派生開発から SPL への移行の本質はこちらであるように考えられた。そこで，(4) で述べた不確実性フレームワークを派生開発から SPL への移行に適用することを着想した。

XDDP4SPL のプロセス実施における不確実性の下での意思決定，たとえば既存ソフトウェア資産のスペックアウト（リバースエンジニアリング）を行うか，SPL のパラダイムによる開発を実施するか，局所識別フィーチャを定義するか，局所識別フィーチャを大域識別フィーチャに昇格するか，フィーチャの定義を変更するか・・・といった意思決定の典型例を定義したうえで，XDDP4SPL における不確実性フレームワークの適用を検討した。

残念ながら，農業用自律走行車を題材とした派生開発から SPL への移行のケーススタディの進捗がリソース不足に因り芳しくなく，リアルな題材による評価にまでは至らず，小さな例題による評価しかできなかった。同ケーススタディは他の課題も絡めて継続実施するので，本研究助成終了後の成果発表を待たれたい。

5. 主な発表論文等

(研究代表者、研究分担者及び連携研究者には下線)

[学会発表](計9件)

中西 恒夫, 久住 憲嗣, 福田 晃, 「派生開発からプロダクトライン開発への移行のための不確実性フレームワーク」, 情報処理学会ソフトウェア工学研究会, 2015年3月13日, 化学会館(東京都千代田区).
田上 宗明, 中西 恒夫, 久住 憲嗣, 福田 晃, 「モデルベース開発のためのモデル進化プロセス」, 情報処理学会ソフトウェア工学研究会, 2015年3月13日, 化学会館(東京都千代田区).

Tsuneo Nakanishi, "A Generative Wireless Sensor Network Framework for Agricultural Use," (Invited Talk), 2014 IEEE Int. Conf. on Electrical Engineering and Informatics (MICEEI), 2014年11月27日, マカッサル(インドネシア).

中西 恒夫, 馬 立東, 久住 憲嗣, 福田 晃, 「システム開発のための不確実性フレームワーク」, 情報処理学会組込みシステム研究会, 2014年3月15日, ICT文化ホール(沖縄県石垣市).

中西 恒夫, 「派生開発からプロダクトライン開発への漸次的移行プロセス XDDP4SPL」(招待講演), 派生開発推進協議会・第4回アフォード・フォーラム, 2013年11月19日, (株)日立情報制御ソリューションズ(東京都台東区).

中西 恒夫, 「派生開発とプロダクトライン開発」(招待講演), 日本 SPI コンソーシアム・ソフトウェアプロセス改善カンファレンス 2013, 2013年10月16日, タワーホール船堀(東京都江戸川区).

Tsuneo Nakanishi, Kenji Hisazumi, and Akira Fukuda, "Soft Landing on Software Product Line through Derivative Development," SPL Symp. MODULARITY: aosd13, 2013年3月28日, 福岡国際会議場(福岡県福岡市).

中西 恒夫, 久住 憲嗣, 福田 晃, 「派生開発からプロダクトライン開発への漸次的移行プロセス XDDP4SPL におけるコア資産管理手法」, 情報処理学会組込みシステム研究会, 2013年3月13日, 対馬市交流センター(長崎県対馬市).

中西 恒夫, ハンス・ヴェルナー グリーペントローク, クラエス イェーガー・ハンセン, 「派生開発方法論 XDDP からのプロダクトライン開発導入」, 電子情報通信学会ソフトウェアサイエンス研究会, 2012年5月10日, 愛媛大学(愛媛県松山市).

6. 研究組織

(1) 研究代表者

中西 恒夫 (NAKANISHI, Tsuneo)
福岡大学・工学部電子情報工学科・教授
研究者番号: 70311785