オムロングループにおけるソフトウェアプロダクトライン（Software Product Lines）の取組み

原田　真太郎・丹羽　徹・赤松　康至・田口　正久

製品の高機能化、ハードで実現していた機能のソフトウェア化などにより、ソフトウェアが大規模・複雑化している。その結果、開発費の高騰、開発期間の長期化、品質コストの増加など、事業的な課題になっている。

この課題に対して、共通化・再利用技術の一つであるソフトウェアプロダクトライン技術に着目し、オムロングループでの効果的な展開のための技術開発、および、エキスパート人財の育成を行った。その結果、5年間で8製品群にソフトウェアプロダクトライン技術を展開し、それぞれの導入目的に沿った効果を得た。

Software Product Lines in the Omron Group

Shintaro Harada, Toru Niwa, Yasuyuki Akamatsu, Masahisa Taguchi

Software is becoming larger and more complex mainly due to higher functionality of the product, software implementation of functions what used to be realized by hardware. As a result, increasing development and quality costs, frequent schedule overruns of software are becoming a business issue.

To resolve this issue, we focused on Software Product Lines technology, which is one of commonalization, reuse technology. We developed technology for effective deployment, and trained expert human resources to lead the development. As a result, we deployed the technology to eight product groups in five years with each achieving the deployment purpose.

1. まえがき

製品の高機能化、システム化、ハードで実現していた機能のソフトウェア化などにより、組込み製品におけるソフトウェアが大規模化・複雑化している。

これはオムロングループでも例外ではなく、主力製品である制御機器・FAシステム、健康医療機器、車載電装部品、社会システム、環境関連機器などにおいても、ソフトウェアの開発量が指数関数的に増加している。その結果、開発費の高騰、開発期間の長期化、品質コストの増加など、事業的な課題になっている。

この課題に対して、我々はこれまで、ソフトウェアのQCD（Quality, Cost, Delivery）向上を狙いにソフトウェアプロセス改善活動（SPI）に取組んできた。具体的には、プロセスリファレンスモデル（CMMI®）を用いた改善活動を、それぞれのビジネスカンパニーで実施し、持続的なQCD向上を実現してきた。

一方、オムロンの主力製品では、図1に示すように、製品群ごとに5～10年程度のスパンでハードウェアを含めた大きな製品リニューアルが行われる。その後、その製品群の派生開発により製品ラインナップを増やすというサイクルで開発が行われる。したがって、その新規開発時に構築したソフトウェア構造や派生開発を行うための仕組みが、製品群のライフサイクル全体のQCDに大きな影響を及ぼす。

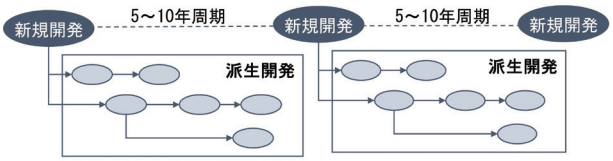

図1　新規開発と派生開発

そこで、これまでのプロセス改善活動に加えて、ライフサイクル全体を踏まえた「新規開発時のソフトウェア構造」および「派生開発を行う仕組」をより強化するための手法として、ソフトウェアプロダクトライン技術に着目した。

2. ソフトウェアプロダクトライン（SPL：Software Product Lines）

ソフトウェアプロダクトライン（SPL）とは、共通化・再利用技術の一つであり、製品を系列的に捉え、系列の中でトータルとしてQCDや顧客満足を向上させる総合的な仕組みや技術である。

連絡先：原田 真太郎　s.harada@omron.com

SPLによる開発により製品数が一定の数値（損益分岐点）を超えると、図2に示すように、製品ごとに個別に開発を行う従来の進めかた以上の費用対効果が得られると言われている[1)]。さらに、製品バリエーションを効率的に生み出すことで製品の市場対応能力向上とシェア拡大に貢献できる。

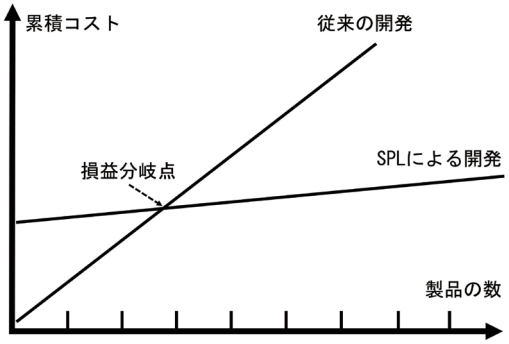

図2　SPLの費用対効果

SPLは以下の3つの活動から構成されている[1)]。

① コア資産開発
　より多くの製品開発で利用されるコア資産を整備し、派生製品を継続的に素早く開発する準備を行うことを目的とする。

② 製品開発
　顧客の要望に応えつつ、コア資産を最大限に利用し、派生製品を効率的かつ効果的に行うことを目的とする。

③ 管理
　全体が正常に立ち上がり稼働するように、組織・技術の視点からコア資産開発と製品開発を支援する。

SPLはこれらの3つの活動を組織的かつ持続的に行うことで効果を発揮する。SPLを導入すると製品開発のたびに既存のソフトウェアを改造するという従来の開発が、図3のようにコア資産を開発し、そのコア資産を活用して製品を生み出すという開発になる。

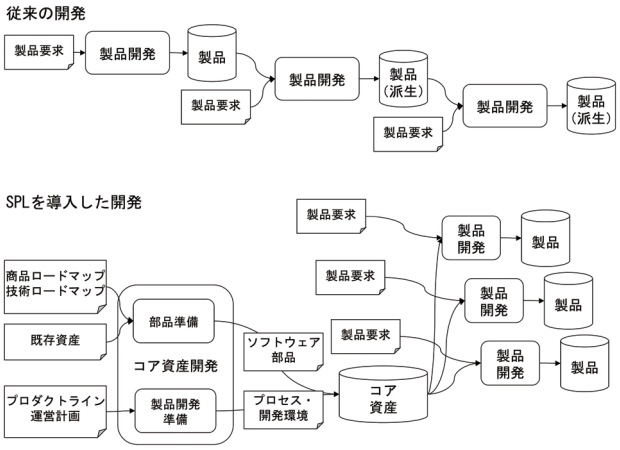

図3　SPLを導入した開発

SPLを導入したことで目覚ましい効果を上げた事例が欧米を中心に多く報告されている。一例を挙げると、米Cummins社は、ディーゼルエンジン制御システムにおいて、製品サイクルタイムを250人月から数人月に効率化した。また日本国内においても、東芝が発電監視制御システムにてコア資産を維持・活用している事例が報告されている[2)]。

この技術を新規製品でのソフトウェア開発に導入することで、その後の派生開発を含めたトータルでのQCDを革新できると考えた。しかし、個々の製品群にて取り組んだ他社事例は多く存在するが、企業全体として複数の事業ドメインを対象とした取り組み事例は少なく、オムロングループとして最適な方法を新たに考える必要があった。

3. 課題

前述のとおり製品の大幅なリニューアルは数年に一度であり、個別の開発部門では新規開発の実践頻度は少ない。従って、それぞれの新規開発でSPL開発への取り組みを行っても十分な知見が蓄積できずに、オムロングループ全社にとって効果的・効率的な取り組みに発展させられないことが想定された。また、人員の配置換えなどにより、SPLを実践した開発者が、次の新規開発も担当するかは不確かで、技術の伝承という面でも対策が必要であった。

そこで、本社機能部門にて、全社の新規開発における知見を集約し、SPL開発技術の蓄積と人財育成を行うことで、効率的・効果的な改善が可能と考え、以下の2つの課題に取り組んだ。

①オムロングループ全社のSPLの実践から得た知見の蓄積

②本社部門でのSPLエキスパート人財育成

4. 対策

「オムロングループ全社の知見の蓄積」に対して、世の中のSPLに関する技術やプロセスをベースに独自の知見を加え、「SPL技術マテリアル」として再構築した。

「エキスパート人財育成」に対して、「SPLエキスパート育成フレーム」を構築し、人財の早期育成を狙った。

「SPLエキスパート」が「SPL技術マテリアル」を活用して開発部門メンバとともに新規開発の課題を解決し、製品群ライフサイクル全体のQCD向上を実現する。その結果、「SPLエキスパート」のさらなる能力向上、「SPL技術マテリアル」の拡充と質向上が、継続的に行われている状態を目指した。

次章より上記の具体的な取り組みを記載する。

5. SPL技術マテリアル

SPLに関する技術やプロセスは、2002 年に、カーネギメロン大学・ソフトウェア工学研究所（CMU/SEI）にて、「A

Framework for Software Product Line Practice」として29の技術領域（Practice Area）が定義されている[3]。また、2013 年　に、ISO/IEC 26550：“Software and systems engineering – Reference model for product line engineering and management”として参照モデルが定義された[4]。

我々は、CMU/SEIによる29の技術領域で定義されている技術やプロセスとISO/IEC 26550で述べられている参照モデルを参考に、技術およびプロセスを再構成した。そして、オムロングループとしてSPLを効果的に導入できるように、開発の実態を踏まえたうえで、特に重要と思える観点を肉付けした。さらに、SPLの実践で得た知見も適宜取り込んだ。

このSPL技術マテリアルを、後述するSPLエキスパート育成、および、SPL導入時の課題解決に活用し、効果的なSPLの導入を実現した。

以降に、本取組みにて、特に注力した技術領域を記載する。これは、従来の開発では不足していた技術やプロセスであり、オムロンとしての最適な取り組みとすべく、特に工夫を重ねた部分である。

5.1　コンポーネント開発（可変性実装ガイド）　SPLで重要なのは、あらかじめ派生製品の違いを「可変性」として洗い出し、製品バリエーションや将来のバージョンアップにおける変更を見越した設計・実装をすることにある。つまり、それぞれの可変性に対して、いつ（バインディングタイム）、どのように（バインディング方式）実現するのかを明確にしておくことである。

この実現方式の選択が、製品の保守性や拡張性に大きく影響を及ぼすため、その選択は極めて重要となる。しかし、実現方式は多種多様である上に、その選択は設計者任せとなっており、統一した選択基準も存在していなかった。

多くのオムロンの組込製品で可変性の実現方式に大きく影響する項目として、「必要なメモリ容量（製品コストに影響）」「バイナリの種別数（生産時の効率に影響）」「コードの可読性（派生開発時の効率に影響）」が挙げられる。各項目の状態とそれに相応しい実現方式との関係を、過去の開発事例を基に可変性実装ガイドとして定義した（表1）。

表1　可変性実装ガイド抜粋

バインディングタイミング：開発時	
実現主体	開発者
主な実現方式	コンパイルスイッチ Make file等による使用ファイル選択
必要メモリ量	少ない
バイナリ数	多い
コード可読性	低い
バインディングタイミング：製造時、設置時	
実現主体	生産者
主な実現方式	設定値の書きこみ ハードウェア構成の切り替え
必要メモリ量	多い
バイナリ数	少ない
コード可読性	—
バインディングタイミング：使用時	
実現主体	ユーザ
主な実現方式	インストーラでの選択 ユーザ設定
必要メモリ量	多い
バイナリ数	少ない
コード可読性	—

これら判断基準を組込み製品で多く使われるC/C++/MBDを対象に比較表として取りまとめ活用した。

この実装ガイドを選択基準とすることで、各設計者により可変性実現方式の適切な選択ができるようになった。

5.2　アーキテクチャ評価（ATAM）　新規開発で構築したソフトウェアが、長期にわたる派生開発で使い続けられるためには、初期のソフトウェア構造（ソフトウェアアーキテクチャ）の良し悪しが重要である。しかし、個別の製品開発を主眼に置いた開発プロセスでは、中長期にわたる改修のしやすさ（保守性・移植性）に関しては、多くの場合十分に検証されない。

そこで、中長期に利用するソフトウェア構造を体系的に開発初期の段階で評価できるかの観点で、複数の手法を検討した結果、CMU/SEIが開発したアーキテクチャ評価手法であるATAM（Architecture Tradeoff Analysis Method）を導入することとした。

ATAMとは、図4に示すように、ビジネス目標を達成するための品質特性（処理速度、可用性、保守性、セキュリティ、移植性など）に関する具体的なシナリオ（品質要求事項）をもとに、アーキテクチャを分析し、ビジネス目標達成を阻害する項目をリスクとしてまとめる手法である[5]。製品ライフサイクル全体視点でのビジネス目標を加えることで、長期にわたる派生開発で使い続けることへの評価も可能になる。

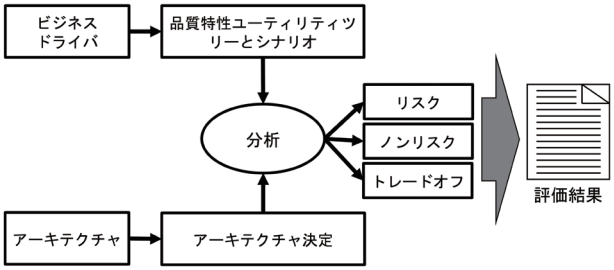

図4　ATAMの全体像

しかし、ATAMは、対象ソフトウェア構造の規模、複雑性やステークホルダーの規模、範囲によって評価期間が約1か月要する手法でもある。そこで、本手法のポイントである「品質特性を起点とした評価」を損なうこと無くプロセス全体を軽量化し、オムロン版ATAMとして体系化した。具体的には、従来手法ではアーキテクチャの理解向上に販売や運用担当など広範囲なステークホルダーの参画を求めるのに対して、オムロンの製品開発特性に合せて開発長、商品リーダ、ソフトウェア設計リーダなど少数で充足する。活動ステップも目的を絞りこんだ結果、従来では4フェーズ20活動ステップを3フェーズ9活動ステップに簡略化した。

また、品質特性の中でも特に「変更容易性」に絞った手法として、同様にオムロン版SAAM（Software Architecture Analysis Method）も体系化した。

この手法を用いてSPL導入開発にて、新規開発時のアーキテクチャ設計にて今後も使い続けられるアーキテクチャであることの確認を行った。また、続く派生開発時の企画時に、アーキテクチャの変更の必要性判断などにも本手法を用いた。

5.3　構成管理　新規開発で構築したコア資産を、長期にわたる派生開発で使い続けられるためには、コア資産を組織的に維持する仕組みが必要である。この仕組みが無いと、派生開発を重ねるごとに場当たり的にソフトウェアに変更が加えられることでコア資産が劣化し、当初目論んでいたSPLの効果が見込めなくなる。

その対策として、変更管理委員会（Change/Configuration Control Board：CCB）を設置し、コア資産への変更を管理することが推奨される。我々は、このCCBを設置・運営するための、典型的なプロセス、体制、変更時の判断基準を定め、それぞれの開発部門に適した形で導入を行った。図5は、変更時の判断基準とその結果の開発プロセスを定めたものである。

現在、複数の部門にてCCBの運用を行い、コア資産を維持管理している。

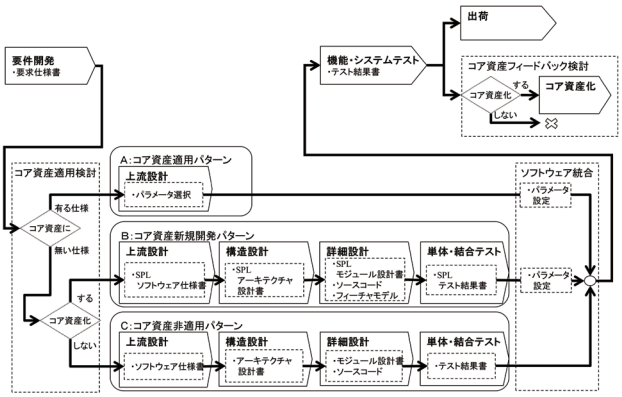

図5　コア資産の変更判断

5.4　プラクティスパターン　SPLの29の技術領域は、技術観点で分割されているため、時間軸での取組みが把握しづらい。それにより、SPLの経験が浅い技術者は、導入に向けた具体的な進め方が分からず活動が計画通りに進まないことがあった。

誰でも効果的にSPLを実践できるようにするためには、時系列で取組みを理解し、活動を進めることが望まれた。

そこで、各製品群にてSPL導入を経験した本社スタッフが集まり、典型的なオムロンにおける29の技術領域の活用の流れ（オムロンSPLパターン）を体系化した。具体的には、図6に示す4つのフェーズ（（商品企画〜コア資産開発〜製品開発〜振り返り）と、各フェーズの典型的な技術領域の関係と流れを整理し、各フェーズでの活動のポイントをまとめた。一例として、図7にSPL立上げの流れを示す。

これにより、SPLの導入がより体系化され、SPL導入の経験が浅いメンバでも、適切な支援が行えるようになった。また、経験が深いメンバにおいても、他取り組みの知見を加えて、より効率的・効果的に行えるようになった。

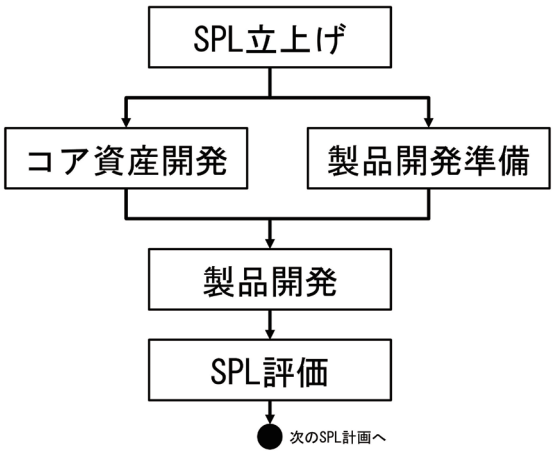

図6　オムロンSPLパターン（全体像）

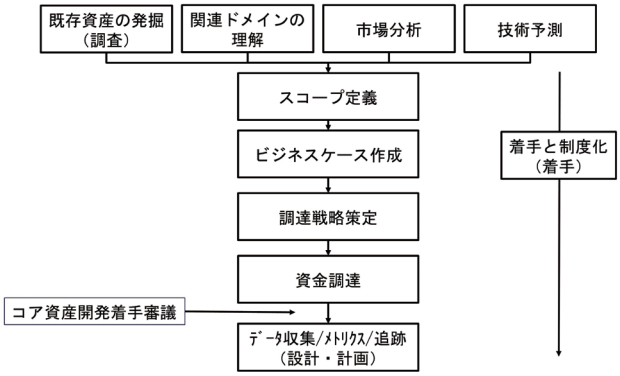

図7　オムロンSPLパターン（SPL立上げ）

6. SPLエキスパート育成フレーム

技術の蓄積と並行して、SPLを開発部門に効果的に適用する人財として、「SPLエキスパート」を育成し、本社機能部門にプールした。

SPLエキスパートとして活動するためには、SPL技術だけでなく、開発部門への働きかけが必要になる。SPL技術を提案し、合意形成を行い、それぞれの開発に適した形で技術を展開するといった、社内コンサルタントとしての能力も必要である。そこで、SPLエキスパートに必要な育成フレームとして、図8に示すように、「コア能力（コンサルティング能力）」「プロフェッショナル能力（SPL技術力）」「経験」の3つの観点で、必要な要件を定義した。

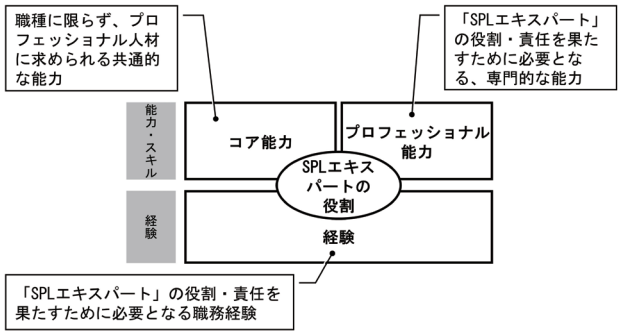

図8　SPLエキスパート育成フレーム

また、SPLエキスパートとしての能力レベルを「初級（Beginner）」「中級（Intermediate）」「上級（Advanced）」として定義し、具体的な育成計画を定めた。

このフレームに基づいて、個々人が目標を設定し、計画、実践、評価・振返りのサイクルを回すこと、および、組織として個々人の育成目標と能力に合った経験の場をアサインすることにより、効率的な人財育成を実現した。

さらに、それぞれの要素を伸ばすためのトレーニングを開発し、継続的に実施した。

この育成フレームを用いて、個々人および組織の技術力を計画的に伸ばすことができた。結果、図9に示すように7名のSPL上級エキスパートを育成した。

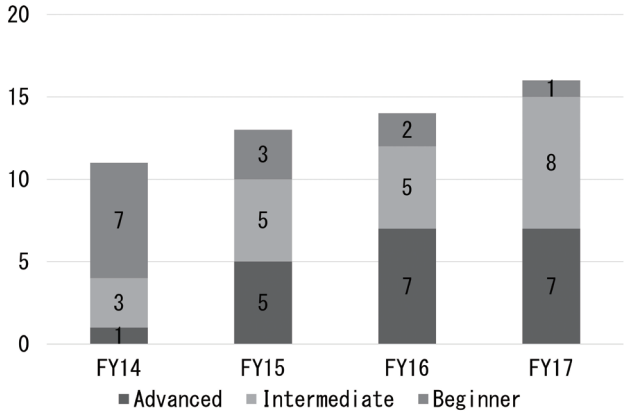

図9　SPLエキスパート認定人数

7. 成果

2013年から本格的な展開を始め、2017年までに16名のSPLエキスパート（内、上級7名）を育成した。また、実際のSPL導入活動から得た知見をベースに、全約400ページの「SPL技術マテリアル」を蓄積した。

図10は、本社機能部門に所属するSPLエキスパートの平均保有能力を示す。コア能力（Core skills）、プロフェッショナル能力（Technical skills）、総合能力（Total）を0～3の数値で表し、経年の変化を示している。この図が示す通り、「SPL育成フレーム」の運用により、組織として継続的に能力を向上させることができている。また、各年にSPLエキスパートの他部門への異動や、他部門からの新規メンバの参加が少なからず発生しているが、組織としての能力を落とすことなく成長し続けることができている。

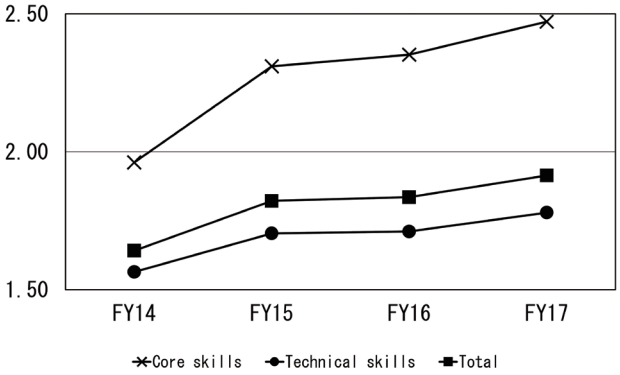

図10　本社機能部門 保有能力平均推移

その結果、8製品群に対してSPLを導入し成果が出た。

具体的には、本社スタッフで構成するSPLエキスパートチームが製品群の企画段階から製品開発に参画し、SPLを適用するかどうかのROI判断、ドメイン分析、コア資産構築、コア資産の維持方法確立などを、開発部門メンバとともに推進した。その結果、以下のような成果があった。

- 派生開発での開発費が従来の30%削減
- 6種類のソフトウェアの1本化による生産効率の大幅向上

・コア資産の事前準備による早期の客先デモによる商談獲得

5年間で8製品群の新規開発へのSPL導入による成果への貢献、および、その実践によるSPLエキスパート人財の早期育成とSPL技術マテリアルの短期間での構築を実現することができ、本取組みは有効であったと考える。

8. むすび

オムロンの組込み製品で特長的な、5～10年での新規開発とその後に続く派生開発のQCDを大幅に向上するための技術として、SPL技術をオムロンに適した形で適用し一定の成果を得た。

一方で、2つの新たな課題を認識した。1つ目は、新規かつ大規模開発においては、高度な専門性を持ったソフトウェアアーキテクトの有無が開発の成否に大きく関係し、当然ながらSPL導入の成否にも影響を及ぼすこと。2つ目は、派生開発は委託先が中心となることが多く、委託先を含めた、より高度な派生開発の在り方を定義し、展開する必要があることである。現在、この2つの取組みに対して、新たな改善活動をスタートしている。

IoT/AI/ロボティクスといった技術により、今後ますます製品におけるソフトウェアの役割が高まっていく中、ソフトウェア開発におけるQCD向上は、より大きな事業課題になる。我々は、オムロンの事業成長を確固たるものにするため、今後も継続的にソフトウェア開発技術を革新していく所存である。

参考文献

1) Paul Clements; and Linda Northrop. Software Product Lines: Practices and Patterns. Addison- Wesley. 2002. 563p
2) Software Product Line Conferences. PRODUCT LINE HALL OF FAME. http://splc.net/hall-of-fame . (accessed 2018-12-14)
3) Paul Clements; Linda Northrop et al. A Framework for Software Product Line Practice, Version 5.0. http://www.sei.cmu.edu/productlines/frame_report/index.html . (accessed 2018-12-14)
4) ISO/IEC 26550: 2013. Software and systems engineering -- Reference model for product line engineering and management
5) Paul Clements; Rick Kazman and Mark Klein. Evaluating Software Architectures: Methods and Case Studies. Addison- Wesley. 2002. 323p

執筆者紹介

原田　真太郎　Shintaro Harada
グローバルものづくり革新本部
開発プロセス革新センタ SPILIT推進部
専門：ソフトウェア工学

丹羽　徹　Toru Niwa
グローバルものづくり革新本部
開発プロセス革新センタ SPILIT推進部
専門：ソフトウェア工学
所属学会：電子情報通信学会

赤松　康至　Yasushi Akamatsu
インダストリアルオートメーションビジネスカンパニー
技術開発本部 第3技術部
専門：ソフトウェア工学

田口　正久　Masahisa Taguchi
グローバルものづくり革新本部
開発プロセス革新センタ SPILIT推進部
専門：ソフトウェア工学

CMMIは、米国CMMI Instituteの米国およびその他の国における登録商標または商標です。
本文に掲載の商品の名称は、各社が商標としている場合があります。