

SPL 型開発におけるコストモデルを用いたアーキテクチャ選択

明神 智之[†] 加藤 正恭[†] 小川 秀人[†]

SPL 型開発では、製品の計画、ラインナップ、ライフサイクルなどを考慮してアーキテクチャを選択する必要がある。アーキテクチャを選択する際は、ソフト資産の初期開発や製品開発に要するコストに応じたアプローチを採らなければならない。本研究では、アーキテクチャを選択するためのコストモデルを用いた開発コスト評価手法を提案する。

A method to select architectures using a cost model in software product line engineering

TOMOYUKI MYOJIN, TADAHISA KATO, HIDETO OGAWA

In software product line engineering, it is necessary to select architectures considering product plan, product lineup and product life cycle. When selecting architectures, it is important to decide an approach according to costs of development of both core assets and products. In this paper, we propose a method to evaluate costs of development using a cost model for selecting architectures.

1. はじめに

ソフトウェアを内在する製品では、製品数に比例して開発コストが増加する問題がある。この問題に対して、製品間で設計や実装を共通するなどして、製品全体としての開発効率の最適化を目指す開発手法として、ソフトウェアプロダクトライン(SPL)型開発がある。

従来型開発から SPL 型開発に移行する場合、製品の計画、ラインナップ、ライフサイクルなどを考慮して、アーキテクチャを選択する必要がある。しかし、共通ソフト資産の開発規模、開発時期、投入時期によって、移行コストと移行によって得られる効果が異なるため、移行前に投資対効果を見積もることが重要である。

本研究では、SPL 型開発におけるアーキテクチャを選択するための判断材料として、コストモデルを用いた評価手法を提案する。本手法により、アーキテクチャの構成要素と製品計画を元に、将来の複数世代の製品開発コストを見積もり、比較することで、どのアーキテクチャが最も効果的であるかを示すことを可能にした。

2. コストモデルを用いたアーキテクチャ選択

2.1. アーキテクチャ候補の導出

まず、SPL 移行後のアーキテクチャの候補を導出す

る。アーキテクチャ候補の導出にあたっては、事業計画や将来予測を加味して、アーキテクチャに対する要件を整理し、要件を解決するアプローチを複数定める。アプローチには、例えば、初期投資を抑えてアーキテクチャを移行する「移行コスト重視」、製品系列間での共通化を推し進める「共通化重視」、性能を最優先で考える「性能重視」などがある。

次に、アプローチ毎にアーキテクチャ候補を作成する。本研究では、従来型開発から SPL 型開発への移行を目指すため、既存製品をベースとして、アーキテクチャの構成要素を抽出し、対応する構成要素をアーキテクチャ候補毎に決定する。このとき、製品系列間で共通して利用する構成要素と、個別に開発する要素に分類する。アーキテクチャ毎に表 1 のように構成要素をマッピングができる。

表 1 アーキテクチャの構成要素マッピング

	移行コスト重視案	製品系列間 共通資産	製品系列内共通資産	
			系列A	系列B
構成要素	OS	Linux	←	←
	HW抽象層

	データ送受信
	ライブラリ	-	系列A専用	他社製X

2.2. コストモデルの構築

コストモデルの基本的な考え方は、共通ソフト資産を開発する初期開発コストと、共通ソフト資産を用いて

[†]株式会社 日立製作所 横浜研究所
Hitachi, Ltd., Yokohama Research Laboratory

開発する製品開発コストである[1]. 本研究では、図 1 に示すように、(1)現行製品を元にして共通部となる共通ソフト資産を開発する初期開発と、(2)共通ソフト資産を用いた N 世代目の製品開発および共通部保守に分けてコストを算出する。また、(3)将来予測のために、開発コスト間の関係性に製品開発計画を用いる。以下、算出方法を述べる。

1. 共通ソフト資産を開発するコストを見積もるために、従来製品の母体規模、従来製品からの流用規模、改造規模、新規開発規模を見積もる。これらの規模を元にして、社内の標準指標を用いて、規模からコストに換算する。
2. N 世代目の製品開発規模を見積もるために、共通ソフト資産を、どれだけ流用、改造し、どれだけ新規で開発するかをパラメータ化する。また、改造と新規開発規模を元に、共通ソフト資産の規模を更新する。N 世代目の製品開発コストも、共通ソフト資産の開発コスト同様に、社内の標準指標を用いて、規模からコストに換算する。
3. 製品開発計画に対して、共通ソフト資産と製品開発のコストをマッピングして、いつ、どの製品で共通ソフト資産を適用するかを反映させる。

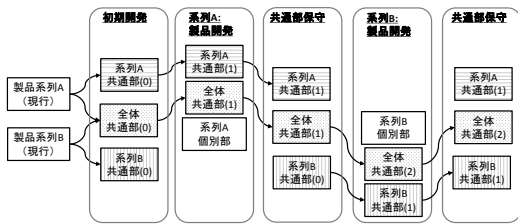


図 1 開発規模見積もり原理

規模の見積もりでは、表 2 に示すように、表 1 のアーキテクチャの構成要素をソフトウェアモジュールに詳細化して、モジュール別に、母体規模、開発規模、利用する製品系列を決定する。

表 2 ソフトウェアモジュール管理表

移行コスト重視案	系列A	系列B	資産分類	母体規模 (kstep)	開発規模 (kstep)
モジュール1	●	●	系列間共通	***	**
モジュール2	●		系列A内共通	***	*
モジュール3		●	系列B内共通	**	*
...					
モジュール4	●		系列A個別	*	*
モジュール5		●	系列B個別	*	*

2.3. 開発コストの評価とアーキテクチャ選択

アーキテクチャ毎に、前述のコストモデルを用いて開

発コストを算出し、時系列に並べて、単期での開発コストと累積での開発コストの推移を求める(図 2)。単年度での開発コストの推移を用いて、その組織が許容しうる開発コストに収まっているかを判断することができる。また、累積での開発コストの推移を用いて、初期投資した開発コストの回収時期を判断できる。例えば、初期投資を抑えたアーキテクチャ C と初期投資を十分に行うアーキテクチャ A を比較すると、2014 年下期にアーキテクチャ A の累計開発コストがアーキテクチャ C を下回り、初期投資した開発コストを回収できることを示している。

これらのコスト評価により、どのアーキテクチャを選択して SPL 型開発を進めることが最適であるかの判別が可能になる。

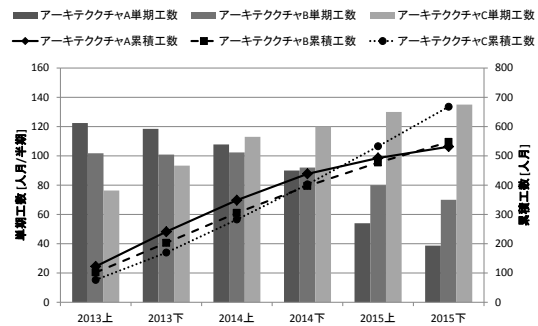


図 2 アーキテクチャ毎の開発コスト評価

2.4. 製品開発への適用と今後の課題

本手法を実際の製品開発プロジェクトに適用して、SPL 型開発への移行を検討した。その結果、共通資産を用いた開発のパラメータの選定が難しく、共通化の効果が直観と一致しない場合があった。また、移行コストの投資タイミングが製品開発計画とマッチしないため、共通ソフト資産の開発と適用を複数段階に分けるなどコストモデルの修正が生じた。

今後は製品開発プロジェクトでの SPL 型開発への移行状況を踏まえ、移行の過程で得られたデータを元に、評価の精度を上げるとともに、適用した製品系列での継続的な開発コストの見積もりへの活用を目指す。

参考文献

[1] G.Bockle, P.Clements, J.D.McGregor, D.Muthig, and J.Schmid: Calculating ROI for software product lines, IEEE Software, vol. 21, pp.23-31 (2004).