

ツール統合による PLSE 環境の提案

—飛行船制御ソフトウェアを例題として—

2009SE085 石川拓也 2009SE314 山内達也

指導教員：野呂昌満

制御ソフトウェア開発事例を用いて検証した。

1 はじめに

ソフトウェア開発の生産性や製品の信頼性の向上のためのアプローチとして、Product Line Software Engineering(以下、PLSE)[1]が提案されている。PLSEは、製品間で再利用可能な部品をコア資産として定義して利用する。コア資産の再利用により、開発の生産性と製品の信頼性の向上が期待できる。PLSEは、特に多機能、多品種なソフトウェアの開発に有効である。したがって、共通の基盤を持つ製品の系列として開発されることが多い組込みシステムは、PLSEを適用する効果的な領域である。PLSEのコア資産には、プログラムの部品だけでなく、ソフトウェアツール(以下、ツール)やソフトウェア開発プロセス(以下、プロセス)等も含まれる。PLSEを構成する活動の1つとして、コア資産の利用の管理が存在する。管理では、コア資産の適切な再利用をおこなうための規則を定めることが求められる。ツールもコア資産であり管理の対象となるが、ツールの選択は開発の前後関係に基づいておこなわれており、選択されたツールが適切であるか保証することは困難である。コア資産としてツールを有効に利用するためには、管理の規則を定める必要がある。

選択されたツールは、製品やプロジェクトに対して適切でない可能性がある。選択したツールが適切でない場合、開発の手戻りや不要な工程が発生する。不適切なツール選択が発生する原因として、選択したツールが適切であることを保証する手段が存在しないこと、様々な機能を持つツールを適切に組み合わせる利用することが困難であることが挙げられる。

本研究の目的は、ツール統合によるPLSEの管理の支援である。ツール統合とは、ソフトウェア開発全体を通してツールを系統的に使用できる環境の構築をあらわす。ツール統合により、PLSEの管理の取り組みの一つである、ツールによる支援を実現する。

目的を達成するための課題として、ツール選択プロセスの定義とツール選択基準の決定が挙げられる。ツール選択プロセスに対し要求を与えることで適切にツールを選択できる環境を構築する。ツール選択基準を決定するために、ツール選択に影響を及ぼす要素とツールの関係を整理する。

ツール選択プロセスは、ツール選択の手順をモデル化することで表現した。モデルに対して与えられる、ツールや要求の関係をQCDを用いて整理した。その関係から、ツールを選択する仕組みを構築することで、ツールの適切な管理を実現する。その仕組みが妥当であるかは、飛行船

2 PLSEのツール選択における問題

2.1 PLSEとは

PLSE[1]とは、再利用可能な部品をコア資産として利用し、ソフトウェアを系統的に開発する開発方法論である。コア資産の再利用により、開発の生産性と製品の信頼性の向上が期待できる。PLSEはコア資産の再利用性を重視しているので、類似の製品を作ることが多い組込みシステムの開発に適していると考えられる[3]。コア資産とは、過去の開発経験等を通じて蓄積され、再利用のために整備された知識や部品のことである。

PLSEはドメインエンジニアリング、アプリケーションエンジニアリング、管理により構成される。ドメインエンジニアリングでは、製品系列を分析し、共通部分と可変部分を定義し再利用可能なコア資産を開発する。アプリケーションエンジニアリングでは、製品に求められる要求を基にコア資産から再利用可能なものを選択し、製品の開発をおこなう。管理では、製品系列の範囲の特定とコア資産の保守をおこなう。

2.2 管理

PLSEの管理を実現するための取り組みは、1) 構成管理、2) データの収集、メトリクス、追跡、3) 内作、購入、発掘、外注の分析、4) プロセス定義、5) スコープ定義、6) プロジェクト計画策定、7) プロジェクトリスク管理、8) ツールによる支援の8つからなる[2]。

本研究では、上記の取り組みの中からツールによる支援に着目する。ツールによる支援では、多様なツールを適切に選択し使用することで開発作業の円滑化を図る。コア資産としてのツールを適切に取り扱い、開発作業の支援をおこなうためには、管理の規則を定める必要がある。

2.3 ツール選択における問題

ドメインエンジニアリング、アプリケーションエンジニアリングであらわれた成果は、互いに反映し合う。PLSEでの開発活動全体を通じて適切にツール選択をおこなうためには、各活動においてツールや要求の変化の都度、ツールが選択可能な規則を定める必要がある。

3 ツール統合

本研究では、ツール統合を実現するために、ツール選択プロセスの定義とツール選択基準の決定をおこなった。

ツール選択プロセスは、ツール選択手順を一般化し、モデル化することで定義した。ツール選択プロセスに対し

要求やツールを与えることにより、プロセスが具体化される。本研究では、対象のプロセスにウォーターフォールモデルを想定し、適切なツール選択をおこなう仕組みを提案した。

ツール選択基準の決定により、各工程で使用するべきツールを提案した。ツール選択基準は、ツールの機能の分類や要求、ツール間の関係により決定した。提案したツール選択プロセスに基づきツール選択をおこなう際に、これらの要素によりツールを選別し、適切なツールを提案する。

3.1 ツール選択プロセスの定義

ツール選択プロセスとは、プロジェクトに対して適切なツールを選択するためのプロセスをあらわす。ツール選択プロセスに対して要求を与えることでツールが提案される。ツール選択プロセスに対してツールを与えることで、具体的なプロセスが構築される。

ツールの選択を適切におこなうためには、選択に影響を及ぼす要素を考慮することが必要である。選択に影響を及ぼす要素として、開発者の要求を考慮した。

ツール選択の手順を一般化し、要求やツールによってプロセスが構築される仕組みをモデル化した。定義したツール選択プロセスを図1に示す。

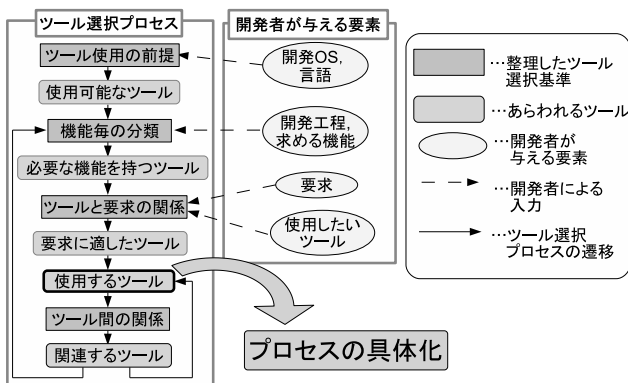


図1 ツール選択プロセスのモデル

3.2 ツール選択基準の決定

開発者の要求をツール選択の基準として捉えた。要求は、生産管理で重視すべき要素である、1) Quality(品質)、2) Cost(開発費用)、3) Delivery(納期)の3つの観点に着目して整理した。QualityではISO9126に基づいた品質特性として、信頼性、使用性、効率性、保守性、移植性を考えた。Costではツールの値段、人件費の削減、開発費用の削減を考えた。Deliveryでは開発期間の短縮、開発プロセスの管理を考えた。以上が、ツール選択に影響を及ぼす要求として考える。

要求と選択の対象となるツールの関係を表に整理した。例としてコード自動生成機能を持つツールを表1に示す。整理した表に対し要求を与えることで、より適切なツールを提案できる。表に対しツールを与えることで、使用する

ツールが以前の工程で選択したツールや求めた要求との矛盾が無いか検査できる。整理した表を表1にあらわす。表1において、◎は要求に大きく寄与する、○は要求を満たす、△は要求と関係を持たない、×は要求に不適切であることをあらわす。

表1 要求とツールの関係

		code Generator	astah	blanco Framework
Quality	信頼性	△	△	△
	使用性	△	△	△
	効率性	×	×	×
	保守性	○	○	△
	移植性	△	△	△
Cost	ツールの値段	○	×	○
	人件費	×	△	△
	開発費用	○	○	○
Delivery	開発期間	○	○	○
	開発管理	△	△	△

3.2.1 ツール選択の前提の整理

ツールが動作する条件により、選択の候補となるツールを選別できると考えた。対応する開発OSとプログラミング言語は、ツールが動作する条件として捉え、ツール選択の前提とする。ツール選択の前提を考えることで、提案されるツールの絞り込みが可能となる。

ツールと前提の関係を表2に整理した。コード自動生成機能を持つツールの例を表2に示す。表において、○は無条件で使用可能なツール、△は制限の下で使用可能なツール、×は使用不可能なツールであることを示す。表2より、開発OS、プログラミング言語の両方に○か△がついたツールが選択の候補として挙げられる。

表2 ツールが使用可能な前提

		code Generator	astah	blanco Framework
開発 OS	Windows	○	○	○
	MacOS	○	○	×
	Linux	○	○	×
プログラミン グ言語	Java	△	○	○
	AspectJ	○	△	×
	C	○	△	○
	C+○	×	△	○
	C#	×	△	○

3.2.2 ツール機能の分類

開発者が求める機能を持つツールを提案するために、ツールを利用される目的毎に分類した。想定したプロセスである、ウォーターフォールモデルにおいて各工程で求められる機能を定義し、各機能を満たすツールを整理した。

3.2.3 ツール間の関係の整理

選択されたツールによって以降のツール選択に及ぼされる影響を考慮するために、ツール間の関係を整理した。ツールには、ツールの利用の前提となるものや互いに連携して使用可能なものが存在する。それらの関係を整理することで、選択されたツールによって以降のツール選択に及ぼされる影響を考慮することが出来ると考えた。ツール間の関係として、以下の5つを定義した。

- 双方の関連： ——
ツール間で互いに連携して使用することで、開発を効率化できる
- 一方からの関連： ——→
ツールが、他の工程で利用されたツールの成果を活用できる
- 利用の前提： -----→
他のツールがあることを前提に利用される
- 追加機能： ——◇
ツールは、以前の工程で使用したツールに機能を追加することを目的に利用される
- ファイル形式の共有：
ツールは、互いの入出力するファイルの形式が共通
上記の関係をもとに、今回選択の対象とするツールのツール間の関係を図2に表現した。

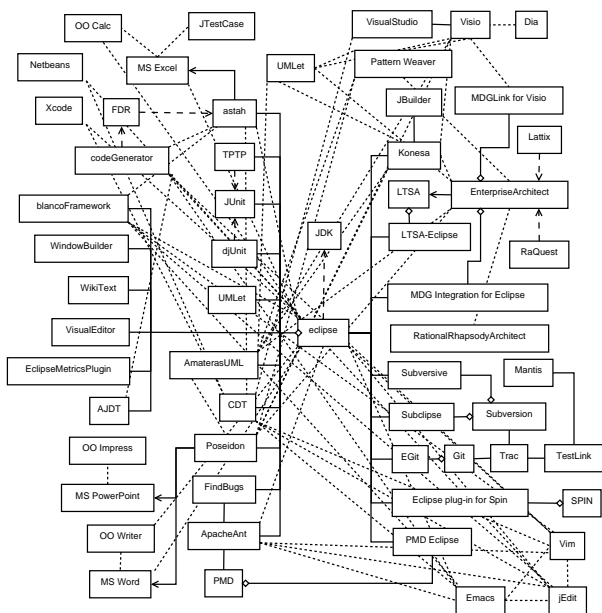


図2 ツール間の関係

ユーザーは、ツール間の関係に基づいてツール選択をおこなうことで、より適切なツールを利用できる。

4 事例検証

飛行船制御ソフトウェア開発を事例において、ツール選択を、提案手法を用いずにおこなった場合と、3章で提案

したツール選択プロセスに基づいてツール選択をおこなった場合を比較する。

4.1 飛行船開発事例の概要

飛行船制御ソフトウェアの概要を次に示す。飛行船に求められる主な機能は、定められたルートの自動航行である。飛行船に求められる非機能は、複数回に渡って開発がおこなわれるので、保守性が求められる。また、本ソフトウェアの開発に携わる人数は5人である。開発に使用するOSはWindowsであり、使用する言語はJava、AspectJである。今回の開発期間は2カ月である。飛行船制御ソフトウェアの開発は本研究室で複数回に渡っておこなわれた開発の成果を基盤とする。前回までの成果はコア資産として定義されているので、今回の開発ではコア資産の再利用をおこなう。

4.2 ツール選択の実施

事例において、我々が提案手法を用いず選択したツールはastah, MSExcel, eclipse, AJDT, Dropboxの5つである。本研究で提案したツール選択プロセスに基づき、ツール選択をおこなう。事例において、PLSE環境の下で適切にツール選択がおこなえるか検証する。前回までの飛行船開発によって得られた成果をコア資産として定義して再利用する。コア資産として1)アーキテクチャの共通部分、可変部分、2)ソースコードの共通部分、3)使用するツール(astah, eclipse, AJDT)を扱う。アプリケーションエンジニアリングの各工程におけるツール選択を実践する。事例では、開発OSがWindows、使用するプログラミング言語がJavaとAspectJである。これらに対応するツールを、機能毎に整理したツール群より選別する。事例では、要求にQualityとして保守性、Deliveryとして開発期間が求められる。アプリケーション実現における、ツール選択プロセスに基づいた選択作業を図3に示す。ツール選択基準によるツールの提案を図4に示す。また、アプリケーション要求開発においてはastah, MSExcel, アプリケーション設計ではastahを選択する。アプリケーション試験ではツールを使用せず、各工程のツール選択を終了する。

5 考察

5.1 ツール選択プロセスの妥当性

定義したツール選択プロセスの妥当性について考察する。ツール選択プロセスに対し要求を与えることでツールが提案され、ツールを選択することで具体的なプロセスが構築されると考えた。飛行船制御ソフトウェア開発事例で、定義したツール選択プロセスに基づいてPLSE[1]環境における開発をおこなった。検証の結果、各工程で要求と使用したい機能を与えることで、使用すべきツールが提案された。コア資産として定義されたツールを、開発で使用するツールとして与えることで、求める要求との矛盾が発生しないか確認することができた。ツールが選択された

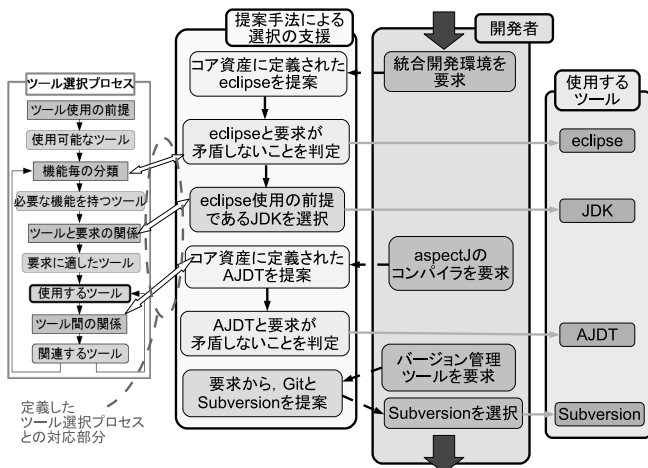


図 3 アプリケーション実現におけるツール選択プロセス

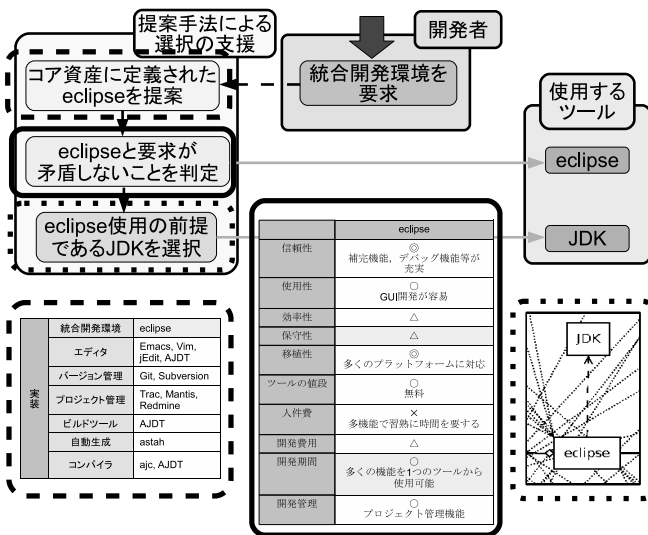


図 4 ツール選択プロセスと選択基準の関係

際に、ツール間の関係を以降のツール選択に反映することができた。以上により、定義したツール選択プロセスにより体系的なツール選択の支援が可能であることが確認できた。しかし、今回提案したツール選択プロセスでは、コア資産として再利用する要素に基づいてツールを選択することは不可能である。この問題を解決するために、再利用されるコードやアーキテクチャをツール選択に影響を及ぼす要素として考える必要があると考える。今回は、ツール選択の対象にウォーターフォールモデルを想定した。したがって、プロセスの動的な変更への対象が確認できていない。動的なプロセスモデルでは、手戻りが発生した際の影響を想定する必要がある。手戻りの先で使用したツールを管理する規則をツール選択プロセスに組み込むことで、手戻りに対応できると考える。

5.2 ツール選択基準の妥当性

本研究で決定したツール選択基準の妥当性について考察する。ツール選択基準の決定により、開発に適切なツールを提案できると考えた。飛行船制御ソフトウェア開発事例で、決定した基準に基づいてツール選択をおこない、提案手法を用いずに選択したツールと比較をおこなう。それぞれの手法によって選択されたツールは、以下のようになった。

- astah, MSExcel, eclipse, AJDT, Dropbox

- astah, MSExcel, eclipse, JDK, AJDT, Subversion

あらわれた 2 点の差異を考察する。JDK は、eclipse を使用する際に予め導入されている必要があるツールである。したがって、提案手法を用いずに選択したツールに含まれていないことは不適切である。JDK は、ツール間の関係により選択されたので、ツール間の関係を選択基準として用いることが妥当であることが確認できた。バージョン管理機能を持つツールは、Dropbox はファイルの変更と削除の履歴を保存する機能のみを持ち、Subversion はファイル統合等、多くの機能を持つという違いを持つ。チームで開発がおこなわれた点から、Subversion を利用の方が適切である。Subversion は要求により提案されたツールであるので、要求によるツールの提案が妥当であることが確認できた。しかし、本研究では要求によるツールの提案の際に、複数の要求が対立する場合には考えられていない。このような場合、開発者は優先する要求を考慮して選択するツールを判断する必要がある。この問題を解決するために、要求とツールの関係の表の点数化、要求の優先度の定義等、ツール選択基準を具体化する必要がある。

6 おわりに

本研究では、PLSE[1] の管理の支援するためのツール選択方法を提案した。ツール選択の実現のために、ツール選択プロセスの定義とツール選択基準の決定をおこなった。事例を用いて、定義したツール選択プロセスとツール選択基準の妥当性を考察した。本研究の成果として、PLSE の管理の一部としてツール選択によるプロセス構築を支援できた。今後の課題として、ツール選択に影響を及ぼす要素の追加、動的なプロセスへの対応、ツール選択をより自動化させる必要がある。

7 参考文献

- [1] K. Pohl, G. Bockle, and F. J. Linden, *Software Product Line Engineering: Foundations, Principles, And Techniques*, Springer-Verlag, 2005.
- [2] P. Clements, L. Northrop, *Software Product Lines: Practices and Patterns*, Addison-Wesley Professional, 2001.
- [3] 高田広章, 枝廣正人, 沢田篤史, 清水徹, 中島達夫, 平山雅之, *組込みシステム概論*, CQ 出版株式会社, 2008.