

要求仕様とモデルの共通性・可変性分析による ソフトウェアプロダクトライン構築

†松本 卓郎

†久保 淳人

‡鷺崎 弘宜

†深澤 良彰

†早稲田大学

‡国立情報学研究所

1 ソフトウェアプロダクトライン開発 とは

ソフトウェアプロダクトライン開発とは、製品の系列において共通に使用するソフトウェアの部品を再利用資産として開発し、その資産を利用して個別の製品を開発する手法である[1]。以下で製品とは、製品のソフトウェア部分を指す。プログラム、要求仕様書、モデルといった製品の資産を入力として、製品間の共通性、可変性を分析し、製品系列で共通なソフトウェアの構造であるプロダクトラインアーキテクチャを構築し、それに基づき再利用資産を開発する。

入力となるプログラム、モデル等の資産は、それぞれ独立に共通性、可変性が分析されるが、その結果をそれぞれ単独に利用しただけではソフトウェアの有効な再利用には結びつかない。その一因として、要求とモデル・プログラム間の追跡性を保つことは一般に困難であることが挙げられる。しかし、入力となる製品の資産がまとまって得られることがあるにもかかわらず、プログラムを除いた各要素の対応関係を考慮して共通性、可変性を分析する作業を支援する仕組みが存在しない。

そこで我々は、製品の要求仕様書の共通性・可変性を分析し、モデルの共通性・可変性分析との対応関係からプロダクトラインを構築する手法を提案する。提案手法により、製品の資産を有効に活用したプロダクトライン構築を支援し、ソフトウェアの再利用性の向上を可能とする。

2 提案手法

我々は、製品の要求仕様書の共通性・可変性を分析し、モデルの共通性・可変性分析との対応関係からプロダクトラインを構築する手法を提案する。対象とするモデルは、オブジェクト指向モデルにおける静的構造を持ったクラスモデルとする。要求仕様書中のある一つの単語や段落等と、モデル中のある一つのモジュールとを自動的に 1 対 1 に対応付けることは困難であるため、要求仕様書の共通性と可変性を、モデルの共通性と可変性と照合することにより、自動的なプロダクトライン構築を支援する。提案手法の概略を図 1 に示す。

ある既存製品の要求仕様書とモデルのセットが複数あるものとする。要求仕様書に形態素解析を行い、品詞ごとの単語に分けたのち、全ての要求仕様書間で共通に出現する単語を抽出し、その集合を共通部とする。共通部に属さない、ある要求仕様書特有の単語の集合は可変部とする。モデルも同様に、全てのモデル間で共通に使用されているモジュールを特定し、その集合を共通部とする。共通部に属さない、全てのモデルに

Building Software Product Line based on Commonality and Variability Analysis of Models and Requirement Specifications:
†MATSUMOTO Takuro, KUBO Atsuto, FUKAZAWA Yoshiaki, Waseda University.

‡WASHIZAKI Hironori, National Institute of Informatics.

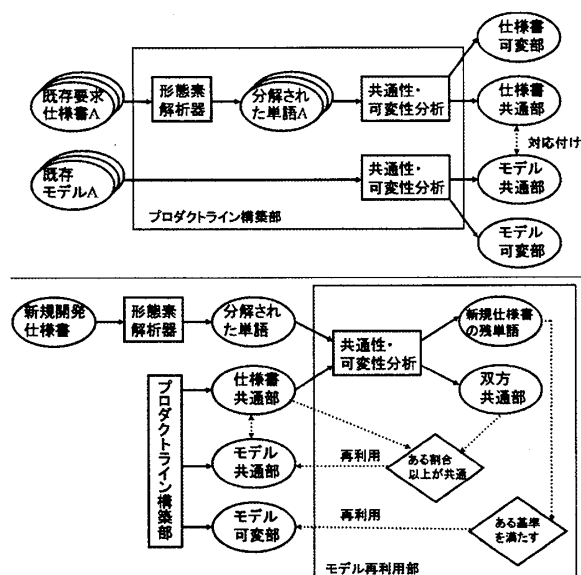


図 1: 提案手法の概略

は出現しないモジュールの集合は可変部とする。要求仕様書間の共通部とモデル間の共通部には対応関係があるとみなし、自動的に 1 対 1 に対応付ける。それぞれの可変部に対しては、あるモジュールに対し、ある一つおよび複数の語を対応付けるのは困難であるため、対応付けない。以上をプロダクトライン構築部とする。

新規製品を開発する際にその製品の要求仕様書を入力とすると、その要求仕様書中に構築部における要求仕様書群の共通部中の単語が何割出現するかが分析され、その値がある閾値以上である場合は、対応するモデルを再利用する。構築部におけるモデル間の可変部に対しては、先ほどの分析時に共通でなかった、新規製品の要求仕様書中の単語を利用するアルゴリズムを作成し、可能な限り特定する。

提案手法を適用する例を以下に示す。

エレベータシステムを例にとる。図 2 を要求仕様書の例、図 3 をそれに対応するモデルとし、A, B を既存製品と仮定しそれぞれの資産をシステムに入力する。要求仕様書に形態素解析を行い、品詞ごとの単語に分けたのち、要求仕様書間で共通に出現する単語を抽出すると、「フロア」、「ボタン」等の単語が得られ、その共通語の集合が共通部となる。共通部に属さないその他の単語の集合は可変部となる。モデルも同様に分析し、モデル間で共通に使用されている「階」、「上方向ボタン」、「下方向ボタン」のモジュールを特定し、その共通モジュールの集合を共通部となる。共通部に属さない、モジュールの集合は可

要求仕様書A ・各階には、上または下へ行きたいことをリクエストするボタンがある。 ・エレベータ停止時に、フロアの上昇(下降)ボタンを押すと、ドアが開く。 ・...
要求仕様書B ・各フロアにはエレベータを呼ぶためのボタンがふつう二つついており、これらを通してエレベータを呼ぶことができる。 ・各フロアには、スピーカがついている。これはエレベータの到着を音で知らせる。 ・...
要求仕様書C ・フロアにある、上ボタン、下ボタンを押すと、最適なエレベータがそのフロアまで移動し、停止するようにスケジューリングされる。 ・スケジュールにあわせてエレベータが移動、停止する。 ・...

図 2: 要求仕様書の例 [2] [3] [4]

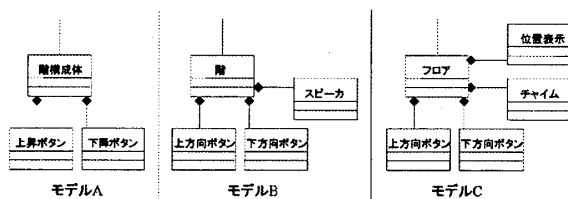


図 3: 図 2 に対応するドメインモデルの例 [2] [3] [4]

変部となる。このとき、要求仕様書群の共通部とモデル群の共通部は自動的に 1 対 1 に対応付けられる。

C を新規に開発する製品とし、その要求仕様書をシステムに入力すると、先ほど得られた要求仕様書の共通集合と「フロア」等の単語が一致し、その割合が事前に定めた閾値以上であるので、要求仕様の共通部に対応付けられたモデルが再利用可能であると判定される。構築部におけるモデル間の可変部に対しては、あるアルゴリズムによって「スピーカ」がその製品に使用されると特定される。最終的に得られるモデルのひな形を図 4 に示した。図 4 より、提案手法がモデル間で共通するモジュールを再利用し、適切に可変部も再利用していることが確認できる。

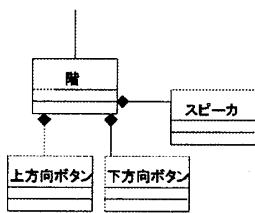


図 4: 提案手法により得られるモデルの例

3 評価実験

[2] [3] [4] において web に公開されている、要求仕様書とモデルの組を使用し実験した。

一般的なエレベータシステムを仮定し、要求仕様書とモデルの組を 3 件用意した。うち 2 件を既存製品、他の 1 件を新規開発する製品と仮定し、提案手法を適用した際に、新規開発製品のモデルのうち再利用資産から得られる割合を計測した。

実験の詳細な条件を以下に記す。クラスモデルは XMI を解析し、クラス図間でクラス名どうしが完全に一致するものを共通であるとし、そうでないものは可変であるとした。形態素解析には ChaSen [5] を使用した。品詞ごとに分解された要求仕様書の単語のうち、クラスの候補になりうる語である名詞のみを抽出し共通性、可変性分析の対象とした。要求仕様書の共通部と新規開発製品の要求仕様書との共通性判定の閾値は仮に 50% とした。モデルの可変部に対しては、クラス名と新規開発製品の要求仕様書の可変部の名詞とが完全に一致する場合、特定できたとみなし、再利用した。

実験により得られた結果を表 1 に示す。

表 1: 実験結果

新規開発製品とするサンプル	A	B	C	平均
他2つのサンプルの仕様書群との共通性	86%	60%	86%	77%
再利用資産から得られたモデルの割合	8%	21%	27%	19%
クラス名を整えた場合の再利用資産から得られたモデルの割合	83%	64%	60%	69%

要求仕様書の共通部と新規開発製品の要求仕様書との共通性はそれぞれ高く、各々モデル間の共通部が再利用されたが、最終的に再利用資産から得られたモデルの割合は低かった。これは、既存製品のクラス名に、「階」と「フロア」のように意味は同じであるが揺らぎがある単語が存在し、名前が完全に一致しないため上記のアルゴリズムでは共通なクラスであると判定されないためである。

試験的に、役割が同じであるクラスを単語の揺らぎがないものに改名させ統一し、再度同様に提案手法を適用したところ、最終的に再利用資産から得られたモデルの割合は高くなった。

この結果により、既存製品のクラス図のクラス名に揺らぎがなければ、提案手法を用いてソフトウェアの資産からプロダクトラインを得られることが確認できた。

4 おわりに

我々は、製品の要求仕様書の共通性・可変性を分析し、モデルの共通性・可変性との対応関係からプロダクトラインを構築する手法を提案し、その有効性を検証した。モデルの共通性、可変性の分析の手法に改良が必要であり、文字列の類似度、類語といったものを活用し、揺らぎをなくす仕組みを調えることが早急な課題である。また、可変部の対応関係を抽出することなどを今後の課題としたい。

参考文献

- [1] P. Clements and L. M. Northrop, *Software Product Lines : Practice and Patterns*, Addison-Wesley, 2001.
- [2] <http://www.unisys.co.jp/tec.info/tr60/6014.pdf>
- [3] <http://www.ics.t.u-tokyo.ac.jp/tmori/lectures/is/2005/elevator/elevator.html>
- [4] <http://www.ogis-ri.co.jp/otc/hiroba/technical/DDJ/index.html>
- [5] <http://chasen-legacy.sourceforge.jp/>