

XMI を利用した UML モデル間の整合性検証支援システム

浜田祐丞[†] 白井靖人[‡]

[†] 静岡大学大学院情報学研究科 [‡] 静岡大学情報学部

1 はじめに

オブジェクト指向ソフトウェア開発では、UML によって分析して複数のモデルを作成する。ここで作成されるモデルは互いに密接な関係を持つが、それらのモデル間の関係を十分に検証するための手法は確立されていない。設計や実装の段階で不整合が見つかったら、それを修正するために多くのコストがかかるため、分析の段階で整合性を検証することが望ましい。

本研究では、UML モデルの構造的な側面に注目して整合性を検証するための検証項目とその基準を定め、検証作業を行うシステムを実装した。なお、UML モデル間の意味的な整合性については考慮していない。

関連した研究として、UML1.x の主要な 6 つのモデル間の整合性検証に関する研究 [1] がある。この研究では、要素につけられたラベルの文字列を比較して整合性の検証を行っている。不整合を発見した場合、どのような順序で再検証すれば効率がよいかを示すことができるといった特徴がある。

本研究では、XMI を用いることで、要素につけられたラベルの文字列比較だけではなく、本質的にそれらが同じものを示しているかどうかを判断することができる。これにより、モデルからプログラミングコードに変換する際の本質的な不整合を防ぐことができるといったメリットがある。また、UML2 で定義されている 13 のモデル全てに対応している。

2 研究概要

本研究では、各モデルの要素間での対応を調べることによって整合性の検証を行う。そのため、まず、各モデル間での整合性の検証項目と検証基準を設定した。次に、それらに沿って検証を行うシステムを構築した。

対象とする XMI はバージョン 2.1 であり、テストケースとして Enterprise Architect[3] で作成した XMI

A System for Verifying among UML Models

Yusuke HAMADA[†] and Yasuto SHIRAI[‡]

[†] Graduate School of Informatics, Shizuoka University, Hamamatsu, 432-8011, Japan

[‡] Faculty of Informatics, Shizuoka University, Hamamatsu, 432-8011, Japan

表 1: 整合性検証項目の一例

クラス図	シーケンス図
クラス	参加要素
メソッド	メッセージ

表 2: 検証基準

警告基準	エラー基準
・メッセージの名前とメソッドの名前が一致する。	・メッセージに対応するメソッドが存在する。 ・メッセージの送り先と送り元のクラスに関連がある。

ファイルを使用した。

2.1 整合性検証項目

まず、UML2 で定義されている 13 のモデルにおいて、特に密接な関係を持つモデルに注目し、そのモデル同士の持つ関係（以下、整合性検証項目）を列挙した。例として、クラス図とシーケンス図間の整合性検証項目を表 1 に示す。

次に、各整合性検証項目においてどのような基準を満たせばいいかを設定する。基準として、最低限満たさなければいけない基準（以下、エラー基準）と、満たすことが望ましい基準（以下、警告基準）を設定した。エラー基準は、モデル間で要素の対応が取れていないなど、実質的に整合性が取れていない重度の不整合がないかを判断するための基準である。警告基準は、同じ要素を指しているのにモデル間で名前が異なっているなど、実質的な整合性は取れているが軽微な不整合がないかを判断するための基準である。例として表 1 のクラス図のメソッドとシーケンス図のメッセージとの対応についての基準を表 2 に示す。

こうして、モデル間の整合性検証項目を 36 個設定し、エラー基準を 24 個、警告基準を 23 個設けた。

2.2 整合性検証支援システム

2.1 節で設定した項目を検証するためのシステムを、Java™ Platform SE6 を用いて実装した。

入力にあたる XMI ファイル内の各モデルの要素タグを検索し、各整合性検証項目の基準に沿って属性値を比較・検証する。検証した結果、警告とエラーを表

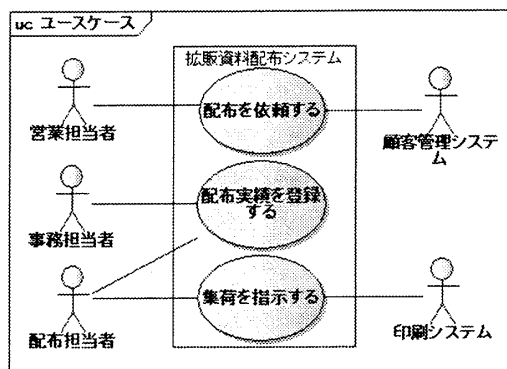


図 1: ユースケース図

示する。

3 実験・検証

Uテストケースを作成し、本システムを利用して整合性の検証を行った。ここで作成したテストケースは拡販資料配布システム [2] を拡張し、UML2 で定義されている 13 のモデル全てを用いてモデリングしたものである。検証のためにわざと一部に不整合な箇所を含むものを複数用意した。

一例として、ユースケース図 (図 1) とアクティビティ図 (図 2) に関する検証結果を図 3 に示す。この例では、(1) ユースケース図のアクタ「事務担当者」がアクティビティ図のアクティビティパーティションとして存在していない、(2) ユースケース図ではアクタ「配布担当者」とユースケース「配布実績を登録する」が関連しているがアクティビティ図ではアクティビティパーティション「営業担当者」とアクション「配布実績を登録する」が関連している、という二つの不整合が含まれている。

検証結果から、正しく検証が行われていることが確認できる。

4 おわりに

本研究では、UML2 で定義された 13 モデルの要素間での整合性の検証項目を設定し、それに基づいたシステムを構築・検証した。

改善点として検証後のモデルの修正支援が挙げられる。現状では、不整合箇所をどう直せば整合性が確保できるかは、モデル作成者自身がエラーメッセージから読み取るしかない。

改善案として、モデリングツールと連携して不整合箇所を視覚的に表示して修正方法を提示するといった、不整合箇所と修正方法の提示を行うことが挙げられる。

今後の展望としては UML モデリングツールとの統

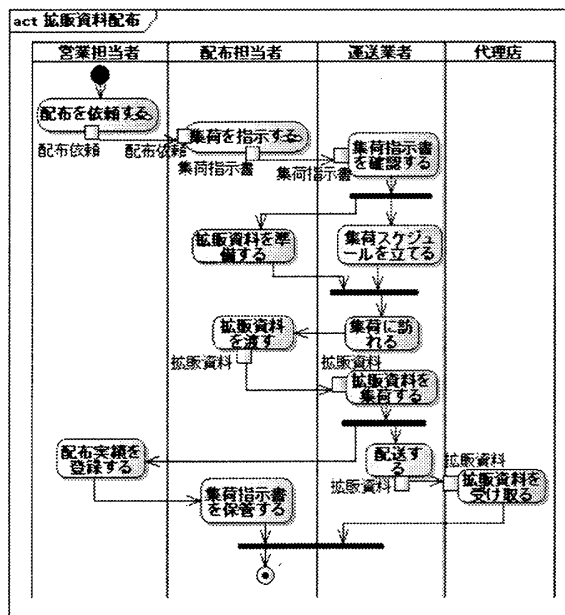


図 2: アクティビティ図

警告数: 1 エラー数: 1

警告
ユースケースモデル: アクタ「事務担当者」と名前が一致するアクティビティパーティションがアクティビティ図にありません

エラー
ユースケースモデル - アクティビティ: アクタ「配布担当者」とユースケース「配布実績を登録する」の関係とパーティション「配布担当者」とアクション「配布実績を登録する」の関係が対応していません

図 3: 検証結果

合が考えられる。本来 UML のモデル作成者にとって XMI は必修のものではないので、XMI を介していることをユーザが意識することなく整合性の検証を行えることが望ましい。UML モデリングツールと統合すれば、UML モデルをモデリングツールで作成し、検証ボタンを押すだけで検証結果が返ってくるという動作が実現できる。

参考文献

- [1] 大西淳, "UML におけるモデル整合性検証支援システム", 電子情報通信学会論文誌, Vol.J84-D-I, No.6, pp.671-681, 2001
- [2] 浅井麻衣 他, 現場の UML -モデルベース開発の全て-, ソーテック社, 2006
- [3] SPARX SYSTEMS, <http://www.sparxsystems.jp/>, 2007