

平成22年 2月 9日現在

研究種目： 基盤研究（C）
研究期間： 2007～2009
課題番号： 19500030
研究課題名（和文） 検証可能なモデルコンパイラに関する研究

研究課題名（英文） A Study on Verifying Model Compiler

研究代表者

鷓林 尚靖（UBAYASHI NAOYASU）
九州工業大学・大学院情報工学研究院・准教授
研究者番号： 80372762

研究成果の概要（和文）：

本研究では、拡張可能なアスペクト指向モデリング言語AspectMと検証可能なモデルコンパイラを開発した。拡張可能なアスペクト指向モデリングの概念は有効であるが、その一方で、モデルコンパイル（織り込み）の正しさを確認することは必ずしも容易ではない。この問題を解決するため、本研究では、AspectMモデルコンパイラ上に、一連の検査器、すなわち、メタモデル検査器、モデル構造検査器、アサーション検査器の機能を持たせた。

研究成果の概要（英文）：

In this research, we developed an extensible AOM (Aspect-Oriented Modeling) language called AspectM and a verifying model compiler. Although the notion of extensible AOM is useful, it is not necessarily easy to confirm the correctness of model compilation (weaving). To deal with this problem, the AspectM model compiler provides a set of verifiers consisting of a metamodel checker, a model structure checker, and an assertion checker.

交付決定額

（金額単位：円）

	直接経費	間接経費	合計
2007年度	1,100,000	330,000	1,430,000
2008年度	1,000,000	300,000	1,300,000
2009年度	1,000,000	300,000	1,300,000
年度			
年度			
総計	3,100,000	930,000	4,030,000

研究分野： 総合領域

科研費の分科・細目： 情報学・ソフトウェア

キーワード： ソフトウェア工学、モデル駆動開発、アスペクト指向、モデルコンパイラ、ドメイン特化言語、拡張可能モデリング環境、検証

1. 研究開始当初の背景

昨今、携帯電話やデジタルTVなどの普及

には目を見張るものがあるが、これらの機器に搭載されるソフトウェアには従来に無い高度な品質が求められる。このようなソフト

ウェアを短期間で開発するには今までとは違った開発手段が必要となるが、そのための方法の一つがモデル駆動開発手法（MDD：Model-Driven Development）である。モデル駆動開発手法とはソフトウェア開発のライフサイクルを通じてモデル中心で開発をおこなう手法であり、その代表的なのがOMG（Object Management Group）が主導するMDA（Model-Driven Architecture）である。MDAによる開発と従来のオブジェクト指向開発の違いは、主に設計フェーズにある。MDAでは、UML（Unified Modeling Language）による設計モデルを特定のプラットフォームに依存しないPIM（Platform Independent Model）と、依存するPSM（Platform Specific Model）に分ける。そして、PIMからPSMへはモデルコンパイラを用いて自動変換する。さらにPSMから特定のプログラミング言語に変換する。このようなMDAには、以下のようなメリットがある。

- 開発者は特定のプラットフォームにとらわれることなく、PIMの開発に全力を注ぐことができる。すなわち、従来のコーディング中心の開発からモデル中心の開発にパラダイムシフトすることが可能になる。
- 同じPIMから複数のPSMを生成することができる。すなわち、PIMモデル部品とモデル変換規則をライブラリ化することにより、様々な機能やプラットフォームに対応したプロダクト群を生成することが可能になり、プロダクトライン型開発の実現につながる。

しかしながら、現状のMDAには、モデル作成者が独自にモデル表記能力やモデル変換機能を拡張することが難しいという問題がある。そのため、アプリケーションの特性にあったモデルを明示的に記述することが容易ではない。また、仮にモデル表記能力やモデル変換機能が拡張できたとしても、それが本当に正しいか否かを確認することは容易ではない。すなわち、アプリケーションの特性にあったドメイン特化のPIMを開発したとしても、それから正しいPSMに変換できなければ、上記に示したMDAのメリットは達成されないことになる。

このような問題を解決するためには、拡張可能なモデリング言語と検証可能なモデルコンパイラを構築する技術を確立する必要がある。

2．研究の目的

我々の研究室では、従来から、MDD研究の一環としてUMLベースのアスペクト指向モデリング言語AspectMの開発に取り組んで来た。アスペクト指向とは、横断的関心事をモジュール化するための技術である。MDAが対象としている「プラットフォーム」も横断的関心事の一つである。すなわち、PIMにプラットフォームに関する記述を追加しようとするモデル上の様々な箇所に横断的に影響が及ぶ。PIMにアスペクト指向を導入することにより、プラットフォームに関する関心事をまとめてモデル化できる。モデル作成者がアスペクトを追加することにより、モデルコンパイラの機能を拡張することが可能となる。

AspectMの支援環境は、1)リフレクティブモデルエディタ、2)アスペクト指向に基づいた拡張可能モデルコンパイラ、から構成される。前者は適用ドメインに合ったモデル表記を定義、導入することが可能なモデルエディタである。後者は新たに導入したモデル表記をどのようにコンパイルするかを定義することが可能なコンパイラである。

拡張可能なモデルコンパイラは様々な利点をもつ一方、コンパイラそのものの正しさを保証することが困難となるという側面をもつ。通常のコンパイラ（プログラミング言語処理系）でも正しさを検証することは重要な研究課題の一つであるが、拡張可能性が入ると検証はより困難になる。AspectMコンパイラの場合、モデル作成者はリフレクティブモデルエディタを使用してメタモデルを拡張したり、アスペクトを新規に作成して新たなプラットフォームへのモデル変換を可能にしたりすることができる。したがって、モデル作成者による拡張方法に誤りがあると正しいモデル変換が行えない。この場合、モデル作成者はPIMに問題があるのか、それとも自らが行った拡張方法に問題があるのか、容易には判断できない。

我々は、このような問題に対処するため、「検証可能なモデルコンパイラ」を実現するための要素技術を開発することを本研究のゴールに設定した。

3．研究の方法

検証可能性を議論するには、何を検証するかが問題となる。その際、通常のプログラミングコードを対象としたコンパイラではなく、モデルを対象としたコンパイラである点に留意する必要がある。

我々は、検証項目として、以下の2つを設定した。

検証項目1： P I Mから生成された P S Mがメタモデルに違反することはないか？

検証項目2： モデル作成者の意図に沿ったモデル変換が行われているか？

検証項目1は拡張により文法的に正しくないモデル変換が行われていないかを検証するものである。モデルコンパイラ版の文法チェックと言える。検証項目2はモデル作成者が施した拡張が意図した形で行われているかを検証するものである。これは、モデルコンパイラの機能検証と言える。

本研究では、上記2つの検証機構を開発すると共に、既存の A s p e c t M支援環境 (E c l i p s e上に構築)の中に統合することにした。また、これらの検証機構が現実世界の課題に対して本当に有効かどうかを、組み込みシステムのドメインを事例に評価することにした。

4. 研究成果

(1) 検証機構の実現

前述の2つの検証を実現するため、 A s p e c t Mモデルコンパイラに以下の3つの検査器を埋め込んだ。

- メタモデル検査器
- モデル構造検査器
- アサーション検査器

メタモデル検査器とモデル構造検査器は検証項目1、アサーション検査器は検証項目2の検査をそれぞれ行う。

メタモデル検査器

メタモデルの拡張には2つの課題が存在する。まず1点目の問題は、ベースモデルが拡張前のメタモデルに適合していたとしても、拡張後のメタモデルに必ずしも適合するとは限らないことである。たとえば、ベースモデルのクラスがあるメタクラスに依存しているにも関わらず、それがメタモデルから不用意に削除された場合である。ベースモデルだけでなく、メタモデルの拡張変更にも注意を払わなければ、拡張可能なモデリング環境は実現されない。2点目の問題は、コンパイル前の P I Mがそのメタモデルに適合していたとしても、コンパイル後の P S Mがメタモデルに適合するとは限らないことである。 A s p e c t Mではアスペクトを追加することによりモデル変換機能を拡張できるが、

不用意なアスペクトを定義してしまうと、生成された P S Mはメタモデルに適合しなくなる。メタモデル検査器は、 X M Lスキーマを用いて構築した。これは、 A s p e c t Mでは、ベースモデルもメタモデルも両方とも X M L文書として保存されるからである。

モデル構造検査器

モデル構造検査器は D O Mを用いて構築した。 X M L文書としてのベースモデルを走査し、名前の衝突 (同じ名前を持つクラス、属性、操作など)、多重継承、循環継承などを検出する。

アサーション検査器

A s p e c t Mでは、モデル上に成立して欲しい性質をアサーションとして追加することができる。アサーションは述語を用いて記述する。アサーション検査器は、モデルおよびアサーションを P r o l o gに変換することにより実現している。

上記で述べた以外で、先行研究的な試みとして、 O M G標準の O C L (Object Constraint Language) と Q V T (Queries, Views, Transformations)を用いた拡張可能なモデル変換機構とそれに対応する検証機構を試作した。また、クラスレベルの検証 (クラスモデルの構造チェック)だけでなく、インスタンスレベルの検証 (オブジェクト間に不適切な結びつきがないか否かのチェック)を可能にするため、 U M Lモデルを軽量形式仕様記述言語の一つである A l l o yに変換する方法も試作した。

(2) 適用事例

我々の研究室では組み込みシステムを対象としたコンテキスト分析手法 C A M E m b (Context Analysis Method for Embedded Systems)を開発しているが、これを支援するモデリング環境を A s p e c t Mの拡張機能を用いて実際に構築した (図1)。



図1 メタモデル拡張 (上部) と CAMEmb サポートのモデルエディタ (下部)

C A M E m b用のドメイン特化言語を定義するにあたって、新たなプログラミング作業は必要とされず、A s p e c t Mが提供する拡張機能のみで実現できた。また、C A M E m bに基づいた実際の組込みシステムのモデル(今回はライントレーサ)およびその変換を先に述べた検査器を用いて検証した。

(3) 研究成果の公開と今後の展望

開発したA s p e c t M開発支援環境は我々の研究室のホームページ(<http://posl.minnie.ai.kyutech.ac.jp/>)から公開している。また、発表論文(論文誌、国際会議、ワークショップ、国内シンポジウム、国内研究会などで発表したもの)についても、このホームページから公開している。

最終年度(2009年度)は、A s p e c t Mに関する研究の総仕上げを行うと共に、次の研究ステップに進むための準備を行った。前者については、国際会議 CAiSE 2009 (21st International Conference on Advanced Information Systems) で今までの成果を発表した。CAiSE は情報システム分野に関するプレミア国際会議の一つである(採択率は10数パーセント)。後者については、モデリング技術の重要適用領域の一つであるアーキテクチャ設計とプログラムコードの間の双方向追跡性と検証を可能にする研究に着手した。具体的にはアーキテクチャが記述可能なインタフェース機構をアスペクト指向の考え方をベースに実現する方式を考案した。今後、モデリングとプログラミングをつなぐ新たな言語機構として発展する可能性が高い。まだ初期段階の研究であるが、現時点の研究成果をまとめた論文が国際会議 ICSE 2010 (32nd ACM/IEEE International Conference on Software Engineering) に採択された。ICSE はソフトウェア工学の分野では世界最高峰の国際会議であり、今後の発展が期待される。

5. 主な発表論文等

(研究代表者、研究分担者及び連携研究者には下線)

[雑誌論文](計10件)

Naoyasu Ubayashi, Jun Nomura, and Tetsuo Tamai, Archface: A Contract Place Where Architectural Design and Code Meet Together, Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering (ICSE 2010), 査読有, 2010, to appear
Naoyasu Ubayashi, Hidenori Akatoki, and Jun Nomura, Pointcut-based

Architectural Interface for Bridging a Gap between Design and Implementation, Proceedins of 6th ECOOP 2009 Workshop on Reflection, AOP and Meta-Data for Software Evolution (RAM-SE'09), 査読有, 2009, ACM Digital Library

Naoyasu Ubayashi, Genya Otsubo, Kazuhide Noda, and Jun Yoshida, An Extensible Aspect-oriented Modeling Environment, Proceedings of the 21st International Conference on Advanced Information Systems (CAiSE 2009), 査読有, Lecture Notes in Computer Science, Springer-Verlag, vol.5565, 2009, pp.17-31

Naoyasu Ubayashi, Genya Otsubo, Kazuhide Noda, Jun Yoshida, and Tetsuo Tamai, AspectM: UML-based Extensible AOM Language, Proceedings of the 23rd IEEE/ACM International Conference on Automated Software Engineering (ASE 2008), 査読有, 2008, pp.501-502

Naoyasu Ubayashi, Yuki Sato, Akihiro Sakai, and Tetsuo Tamai, Alloy-based Lightweight Verification for Aspect-oriented Architecture, Proceedings of the 6th International Conference on Software Engineering Research, Management and Applications (SERA 2008), 査読有, 2008, pp.171-178

Naoyasu Ubayashi, Toshiki Seto, Hirotohi Kanagawa, Susumu Taniguchi, Jun Yoshida, Takeshi Sumi, and Masayuki Hirayama, A Context Analysis Method for Constructing Reliable Embedded Systems, Proceedins of Workshop on Modeling in Software Engineering (MISE 2008), 査読有, 2008, pp.57-63

鶴林 尚靖, Alloy による設計記述と自動検査, システム制御情報学会誌, 査読無, vol.52, no.9, 2008, pp.316-321

Naoyasu Ubayashi, Yusaku Maeno, Kazuhide Noda, and Genya Otsubo, A Verification Mechanism for Weaving in Extensible AOM Languages, Proceedins of the 2nd International Workshop on Aspects, Dependencies and Interactions (ADI'07), 査読有, 2007, pp.36-41

Naoyasu Ubayashi, Shinji Sano, and Genya Otsubo, A Reflective Aspect-oriented Model Editor Based on Metamodel Extension, Workshop on

Modeling in Software Engineering (MISE 2007), 査読有, 2007, IEEE-CS Digital Library

鷓林 尚靖, 金川 太俊, 瀬戸 敏喜, 中島 震, 平山 雅之, コンテキストベース・プロダクトライン開発とVDM++の適用, 情報処理学会論文誌, 査読有, vol.48, no.8, 2007, pp.2492-2507

〔学会発表〕(計4件)

野村 潤, 鷓林 尚靖, アーキテクチャ記述をカプセル化するインタフェース機構 Archface, 日本ソフトウェア科学会 第16回ソフトウェア工学の基礎ワークショップ (FOSE 2009), 2009年11月20日, 箱根

野村 潤, 鷓林 尚靖, アーキテクチャ設計と実装をつなぐインタフェース機構 Archface, 電子情報通信学会ソフトウェアサイエンス研究会, 2009年8月6日, 北見工業大学

佐藤 友紀, 鷓林 尚靖, 関心事指向アーキテクチャモデリング環境, 電子情報通信学会ソフトウェアサイエンス研究会, 2008年7月31日, 公立はこだて未来大学

金川 太俊, 瀬戸 敏喜, 谷口 奨, 吉田 純, 鷓林 尚靖, 鷲見 毅, 平山 雅之, 組込みシステムの外部環境に着目した動作仕様検証, 電子情報通信学会ソフトウェアサイエンス研究会, 2007年10月22日, 宮城大学

〔その他〕

ホームページ等

<http://posl.minnie.ai.kyutech.ac.jp/>

6. 研究組織

(1) 研究代表者

鷓林 尚靖 (UBAYASHI NAOYASU)
九州工業大学・大学院情報工学研究院・
准教授
研究者番号: 80372762

(2) 研究分担者

()
研究者番号:

(3) 連携研究者

()
研究者番号: