

情報共有と意思決定のための実用的なモデル差分比較ツール

池田 滝飛[†] 松浦 佐江子[‡]

芝浦工業大学 システム理工学部 電子情報システム学科^{†‡}

1. はじめに

開発現場で問題視されていることの一つに UML (Unified Modeling Language) をはじめとする仕様書の品質の問題がある。UML のクラス図は最終的なプログラムの静的構造を表すため、他のモデルによる分析結果を統合する基盤となる。

このクラス図の品質を維持する為には、繰り返し推敲し、複数名でレビューや意思決定を行い、それを共有することが重要である。その為、クラス図に関して、意思決定の際に注目すべき所が分かたり、変更する際に及ぶ影響の範囲が読み取れたり、共有すべき箇所が明示されたりすると便利である。

本学の Project Based Learning を取り入れた実験科目である情報実験Ⅱでは、ユースケース分析などをもとに最初のバージョンのクラス図を作成する。そしてこのクラス図をたたき台として、各メンバーが自分の役割にそって変更をかける。こうすることで、各メンバーの意見が詰まった複数のクラス図ができる。この時、最初のバージョンのクラス図と各々の意見の詰まった複数のクラス図を比較して、各メンバーの意見がクラス図のどの部分に集中し、どんな議論をする必要があるかを読み取る。そしてこれを、新しいバージョンのクラス図を作成する際に行う意思決定の論点決定などに用いる。

作成された新しいバージョンのクラス図はこの後、それぞれの意見を取りこぼさないか、この変更によって影響する範囲がすべて更新されているか等をチェックする必要がある。このチェックは各々の意見が詰まった複数のクラス図と新しいバージョンのクラス図を比較して行う。

こうしてチェックが終わった後、変更箇所の見落としや認識の違いを起こさない情報共有の為に、新しいバージョンのクラス図と共に変更内容を配布する。この変更内容を得るためには新しいバージョンと最初のバージョンを見比べる必要がある。

このように開発プロセス内では、注目すべき所、変更の影響範囲、情報共有すべき所等を得るためには、比較して差分を出すことが有効である。

代表的な UML モデリングツールとして Astah[1]、Enterprise Architect[2]、Pattern Weaver[3]等が挙げられるが、これらには差分比較ツールが備わっている。しかしこのツールを、注目すべき所等の情報を得るために用いるにはいくつかの問題点がある。

本研究では Astah を例に挙げて、クラス図についての差分比較ツールの問題点を改善する。そして、改善したツールによって変更の影響範囲やクラス図をはじめとするモデルの注目すべき所の読み取りを視覚的に支援し、意思決定や情報共有の効率を上げることを目的とする。

Practical Model Difference Comparison Tool for Information Sharing and Decision Making

[†] Takito Ikeda [‡] Saeko Matsuura

^{†‡}Department of Electronic Information Systems, Collage of Systems Engineering and Science, Shibaura Institute of Technology

2. 現在の問題点とアプローチ

2.1. 表示情報粒度とノイズ混入

一つ目の問題点は、表示される情報の粒度とノイズである。図1は実際に Astah で差分を出した時の画面である。このようにクラス図が本来持つ意味とは無関係な位置情報などがノイズとして入ることで、差分情報の中から変更の影響範囲や注目すべき点等の読み取りを阻害しているといえる。また、現在ではクラス内のどの部分が変更されたか表示できない。変更箇所のある時など、変更内容をより明確に知りたい場合に問題となる。

そこで、部品位置の情報など差分の読み取りに不必要な情報を非表示にできる機能と、クラスのどのクラス属性、メソッド、フィールドが変更されたかという情報を表示できるようにする機能を追加する。

図2はこれらの機能を実装したツールによって、図1と同じクラス図を比較した際に出力される差分の表示画面である。削除を赤、追加を緑、編集を黄で表して見やすくしている。このケースの場合、改善画面では差分から情報を読み取りやすくなっていることが見て取れる。

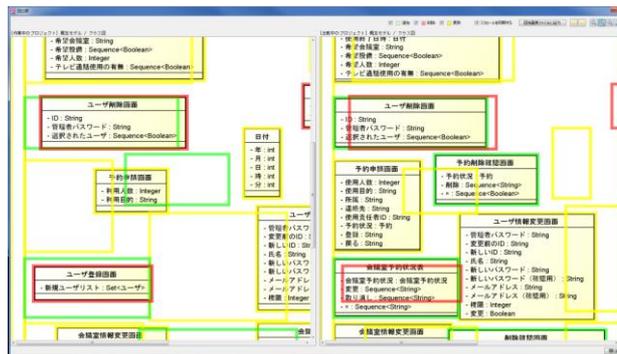


図1 Astah の簡易比較画面

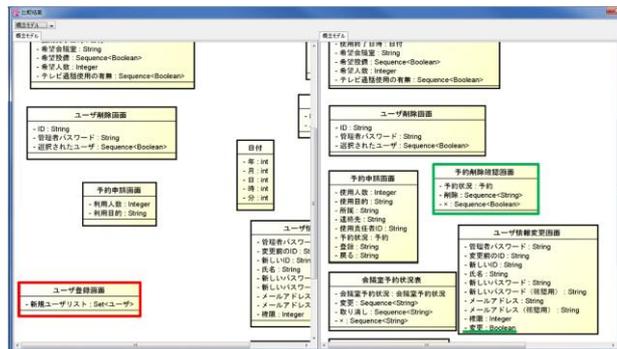


図2 位置情報を非表示に改善した画面

2.2. 比較対象の紐付け

二つ目の問題は、部品に対して内部で付与されている ID が変わると差分が出せない点である。ツール内部では各部品の作成時に ID を付与している。比較する際は、比較する両者の同じ ID の部品同士を比較している。しかし、人間が操作する際には ID を意識しない。その為、図3のような部品差し換え等、内部 ID が変わる操作が行われることがある。この状態で Astah の機能を用いて比較する

と必ず図4のように「削除した」と「追加した」という差分が出力される。このように差分の有無に関わらず同じ出力になる為、読み取りに使用できなくなる。

そこで、クラスが持つ情報を総合的に用いて比較する機能を追加する。これは内部で保持するIDが部品差し換えなどで変わることによって比較ができない場合、IDの他にクラス名、保有メソッドやフィールドなどの情報を用いて自動的に比較対象を検索する機能である。また、自動で比較対象を算出することができない場合や、算出が間違っている場合は、その紐付けをユーザが変更できる機能を追加する。これによりIDが異なる部品がある場合でも比較ができるようになり、比較対象を自動算出するより精度の高い紐付けが可能になると考えられる。

図5は紐付け機能を実装したツールで図4と同じクラス図を比較した時の出力結果である。両者の内容は同じなので差分表示がない。よって正しく比較できていると読み取れる。

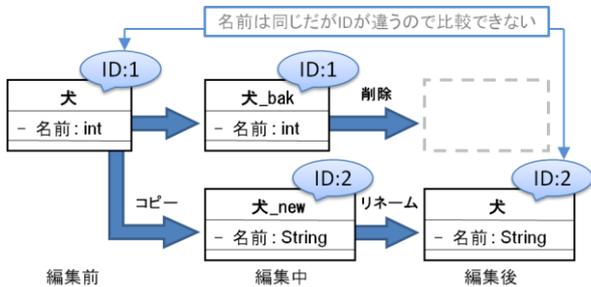


図3 部品差し換えとIDの変化

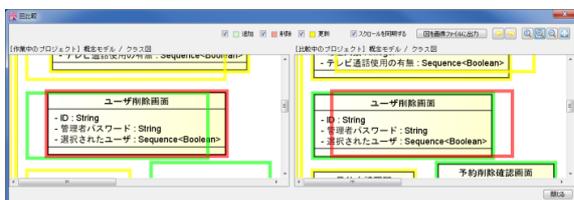


図4 紐付けが切れているときの出力結果

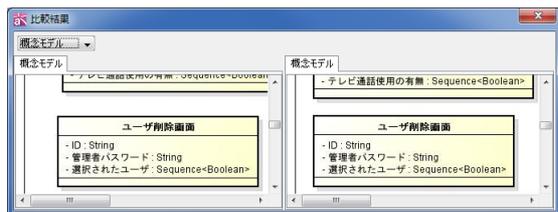


図5 自動紐付けを行った場合の比較結果

2.3. 複数比較

三つ目の問題は、複数のクラス図を同時に比較できないという点である。前述したように、変更の影響範囲や注目すべき所を読み取るには複数のクラス図を見比べる機会が多い。しかし、現在のAstahに実装されるツールでは必ず1対1で比較を行うため、複数のクラス図を比較するには、複数回出力する必要がある。これでは操作に時間がかかり、必要な情報を再度自分でまとめなければならない為、あまり現実的な方法ではない。

そこで、1対多比較を実現する機能を追加する。人間が複数のクラス図を比較する際は全クラス図を同時に見比べることが多いが、本来やっていることは、たたき台となるクラス図と各々が作った各クラス図の比較である。すなわち1対多の比較をしている。その為、基準となるクラス図と複数のクラス図の1対1比較を複数回行い、

その差分を基準となるクラス図に統合する機能とする。

図6は複数比較をした際に出力される差分の表示画面である。このケースの場合、左側が基準となるクラス図、右側が複数のクラス図である。各図はタブに格納されていて、選択することで表示できる。右のクラス図は緑下線は追加された部分、黄下線は変更された部分、緑四角は追加されたクラスで表現されている。メソッドやフィールドの内容が更新されたクラスは青四角で、クラスが追加された場合は全体が青色のクラスで基準となるクラス図にプロットすることで注目点を強調している。

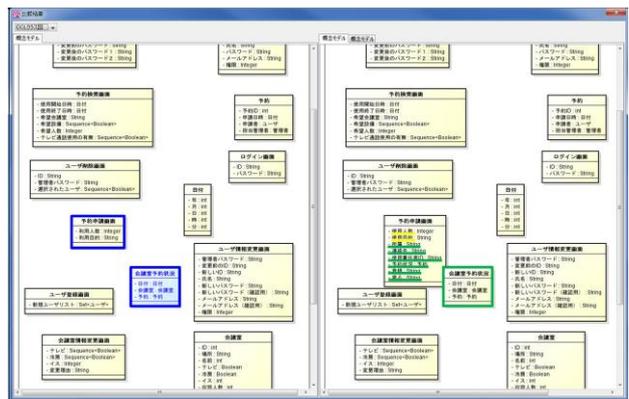


図6 複数比較を行った画面

3. ツールの実装

本研究で行った、機能を拡張した差分比較ツールの実装は、Astahのプラグイン機能として開発した。その際、Astahのプラグイン開発用に用意されたAstah APIを用いた。

4. ツールの検証法

本ツールの実用性検証について次のような方法をとる。2013年度と2014年度に本学の情報実験IIで行われた複数のプロジェクトで作られたクラス図に対して、改修したツールを当ててバージョンごとの差分を取る。この差分を実験に関わっていない被験者が読んだとき、どのように変更されてきたのかを読み取れるかどうか調べる。これによって、改善したツールで出力した差分が実際に使用できるかどうかを判断することができると思う。

5. まとめと今後の課題

本研究は、クラス図の品質を向上させる際に行う比較を自動的に実行し差分出力するツールの問題点を挙げ、その解決法を提案した。

今後の課題として、前述のツール検証を実施して、この改善提案を取り入れたツール実用度を検証する。

また、本来、クラス図を変更した際の影響範囲等はクラス図と他のモデルでは対応がいないため、クラス図内とどまることはまずない。その為、今回取り入れなかった、別のモデルと連携させて差分を表示し、影響範囲などをトレースする方法の検討を行う。

その他、実際にはここに上げきれない問題点についても検討と改善案の提案を行い、機能を組み込むことでツール自体の有用性を高める必要がある。

参考文献

[1]UML and Modeling Tools Astah.net, <http://astah.net>
 [2]Enterprise Architect, <http://www.sparxsystems.jp>
 [3]Pattern Weaver, <http://pw.tech-arts.co.jp>