

プログラムの視覚的理解に向けての1つの試み - UML の3次元化 -

2001MT035 稲葉 光俊 2001MT085 大矢 真実
指導教員 真野 芳久

1. はじめに

近年、システム開発において、オブジェクト指向言語が注目され広く利用されている。その開発設計で、UML によるモデリングが行われている。

UML は本来多数の側面を持つ実体を、個々の側面からの視点で2次元平面に投影した図であり、各側面に特化した視点を得ることができる。

これに対して、複数の関連した側面を多次的に捉える方法も考えられる。この方法ではUMLの2次元的な図形より実体に近い情報を表現できる。よって、この方法の有効性を検討する価値がある。

そこで、本研究では複数の側面からのビューを同時に表現できるツールを試作し、その有効性を検討する。尚、検討する図はUMLを拡張した図であり、1つ側面から見れば既存のUML図と一致するものである。

2. 諸技術の概要

2.1. UML(Unified Modeling Language)

ソフトウェアにおける開発論は、システムの複雑化や大規模化によって1980年代終わりごろから多くの提案がなされてきた。代表的なものを挙げると、Grady Booch によるBooch法、James Rumbaugh によるOMT法、Iver Jacobson によるOOSE法などがあり、これらの方法論は類似した概要にもかかわらず表記法が異なっていた。

これらのオブジェクト指向開発方法論を継承し、表記法を統一したものとしてUML0.9が1996年に作成され、その翌年UML1.1がオブジェクト指向技術標準化団体であるOMG(Object Management Group)により認定された。

2003年にはUML1.5が承認、同年12月にUML2.0の草案が出され、現在各研究機関で意見が出されて調整されているところである。UML2.0はOMGが同じく提唱するMDAへの対応を強化した変更が多くなされている。

本研究では、多くの企業において提供されているツールがUML1.5を用いているため、それを利用する。

次に本研究で取り扱う図の紹介をする。

2.1.1. コミュニケーション図

コミュニケーション図はシーケンス図と合わせて相互作用図と言われ、オブジェクトのメッセージのやり取りを表現

する。コミュニケーション図では特にオブジェクトを中心としてメッセージのやり取りを表現する。UML1.5ではコラボレーション図と呼ばれていたが、UML2.0でコミュニケーション図と名称が変更されている。

オブジェクトの下には基本的に下線を引く。黒三角矢印は同期メッセージ、破線矢印は同期メッセージの戻り、矢印メッセージは非同期メッセージを示している。図2-1に例を挙げる。

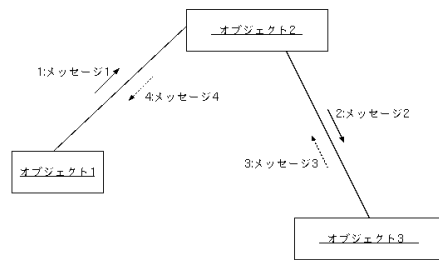


図2-1: コミュニケーション図

2.1.2. シーケンス図

シーケンス図はオブジェクトのメッセージのやり取りを時系列に沿って表現する。シーケンス図は相互作用に関するオブジェクトとそのオブジェクトが存在することを示すライフラインで成り立っていて、そのライフラインの間に引いた矢印とラベルで表現する。図2-2に例を挙げる。

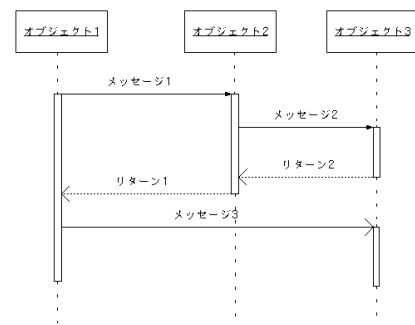


図2-2: シーケンス図

2.1.3. タイミング図

タイミング図はオブジェクトがある期間中にどのような振る舞いをするかを詳しく表現する。図の左端に状態や条件を並べ、図の一番下に時間を記す。状態タイムラインと呼ばれる線を用いて、オブジェクトの状態変化を表現する。状態遷移の発生点にイベント等を記し、変化の理由を表現する。タイムライン間の矢印はオブジェクト間のメッセージを表現する。図2-3に例を挙げる。

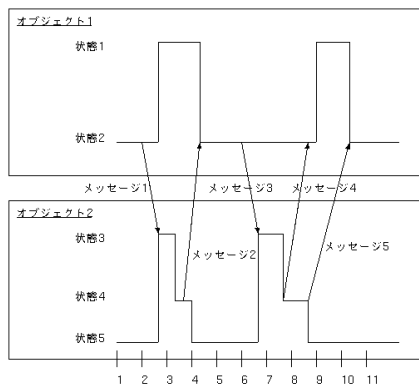


図2-3: タイミング図

2.2. XMI(XML Metadata Interchange)

XMIはUML図を異なるツール間で交換するための方法でOMGにより標準化された仕様である。本来UMLは表記法であり、そのデータ形式は図で表現されるために他のツールや企業間でのデータ交換は難しいものがあった。しかし、その図形データを多くの企業で標準的に利用されているXMLを利用した表現に変換することで異なるツール間や企業間でのデータ交換が可能となった。

具体的にはXMIはMOF(Meta Object Facility)というメタデータが標準化されたリポジトリと記述の定義を利用している。XMIはこのMOFの定義により、UMLモデルをXML DTDとストリームベースのXMLベースに展開することで、UMLモデルのインポートやエクスポートを可能としている。

本研究では、作成したUMLモデルの受け渡しのためにXMIを利用する。

2.3. OCL(Object Constraint Language)

OCLはUMLモデルでは表現できない細かな制約を正確に表現するための言語である。この言語が成立される以前は複雑な数式(束縛変数を使った限定)で記述されることが多く、数学に精通した人でないと理解できないことがあったために、英語の文章で表現される。

OCLはUML1.1から導入され、UMLのセマンティクス記述にも積極的に使われている。OCLはUMLメタモデルに

対してだけでなく、一般の分析設計者がモデルを正確に表したいときに、従来のビジュアルなモデル記述に注釈として追加する形でモデルの厳密性を増すのに使うこともできる。

本研究では整合性の証明が必要な場合に、OCLを利用する。

3. UML図の3次元化に関する検討

3.1. 次元に関する検討

2次元モデルは多数の側面を持つ実体を個々の側面からの視点で2次元平面に投影したモデルである。そのため、各視点に特化した情報を表現することができる。しかし、全体を表現するためには複数の視点が存在するので、各視点に対して各々別の図として表現される。さらに、視点の異なると同一の対象が異なる形で表現されることがしばしば起こる。ユーザは全体を把握するために各図の整合性を保った1つのメンタルモデルを再構成する必要が出てくる。

3次元モデルでは1つのモデルで複数の視点を提供することができる。そのため、複数の関係を同時に把握しつつ、視点を変更することによって各関係に焦点をあわせることができる。さらに、複数の視点を同時に捉えることができ、図の数を減らすことができるので、メンタルモデルを再構成する負担が少なくなる。

3次元モデルのメリットは2次元モデルよりも多くの情報をもつことが挙げられる。これはより実体に近いモデルを表現できるといえる。また、メンタルモデルを再構成する負担が少なくなるので、ユーザに対する負担が少なくなる。

通常の場合ある図での変更は他の図へも影響を及ぼすが、複数の図を同時に表現した1つの図の場合はその可能性がなくなる。そして図の変更をする場合も1つの図を操作するだけですべての範囲での変更操作と同じ意味を持つこととなる。

この反面で3次元モデルは操作性の難しさ、データとしての扱いにくさ、元図形との整合性が取れているかなどの克服すべき問題点が挙げられる。

3.2. UML図の3次元化手法に関する検討

UML図は各図がすべて独立しているのではなく、同一情報を表しながら異なる視点で図を表現しているものもある。

本研究ではその部分を利用し、同一情報でまとめることが可能な部分を1つの平面で表現し、異なった情報をあらわしている部分をその平面に垂直な軸で表現し合成する。利用するUML図はUML1.5とUML2.0の2つの中から考え、実装できそうなパターンをいくつか検討する。最低限の条件で元の図との整合性があることは守る必要がある。

まず図が静的であるか動的であるかによって分ける。これは構造の違いから合成するのに制約がかかってくるため

に制限する。その中で同様な情報をもつ図を選び出すと、動的図からは相互作用図であるコミュニケーション図、シーケンス図、タイミング図の3つが挙げられる。相互作用概要図はアクティビティ図の機能を利用して、すでにある意味での合成図といえるため除外する。

上記の3つの図は「オブジェクト」と「オブジェクト間の相互作用」を共通に持ち、シーケンス図では相互作用の時系列を、コミュニケーション図ではオブジェクト間の関係を、タイミング図では時間軸に沿って、それぞれのオブジェクトの状態変化とその間の関係をそれぞれ中心に表現している。よって、これらの図を2つずつ組み合わせる図を検討する。組み合わせパターンは4通りあるが、本研究ではその中で以下の2つを検討する。

3.2.1. コミュニケーション図とシーケンス図の合成

2章のUML図の説明でも記述したがコミュニケーション図はオブジェクトとメッセージの関係を、シーケンス図はオブジェクトとメッセージの時間関係を表現する。この2つの図においてオブジェクトとメッセージが共通項目であり、1つの図に同時に表現することが可能である。

記述の方法はオブジェクト同士の関係を1つの平面に描き、その平面に垂直となる方向に時間軸をとり、シーケンス図の情報からメッセージを時間順に描く。このシーケンス図部分で用いられるメッセージは、コミュニケーション図部分のオブジェクト間のリンクとしても表現される。

この合成図は各UML図を同時に見ることができ、角度を変えれば片方のUML図だけを見ることが出来る。

また自由に視点を変更できるために、通常のUML図では得ることができない様々な見方をする事や、全体像の表現も部分的な拡大もでき図の理解を促すこととなる。

図3 1 にイメージ図を挙げる。

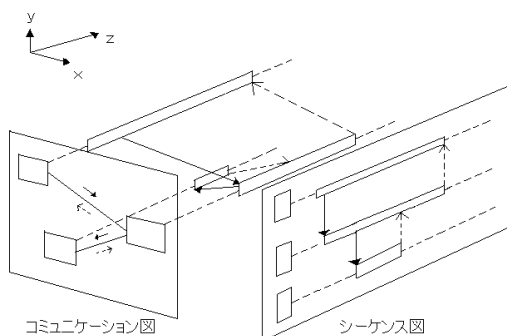


図3 1: コミュニケーション図とシーケンス図の合成イメージ

3.2.2. コミュニケーション図とタイミング図の合成

コミュニケーション図はオブジェクトとメッセージの関係を、タイミング図はオブジェクトの状態変化とメッセージのやり取りを表現する。この2つの図においてもオブジェクトとメッセージが共通項目であり、1つの図に同時に表現することが可能である。

記述の方法はオブジェクト同士の関係を1つの平面に描き、その平面に垂直となる方向にタイミング図の情報からオブジェクトの状態変化とメッセージを時間順に描く。このタイミング図部分で用いられるメッセージは、コミュニケーション図部分のオブジェクト間のリンクとしても表現される。

図3 2 にイメージ図を挙げる。

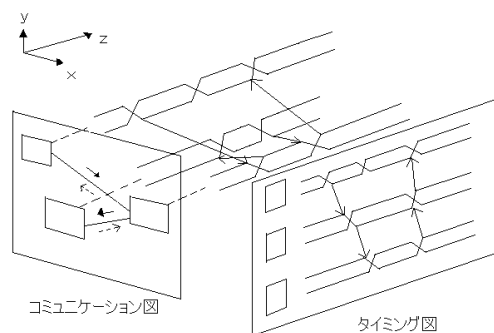


図3 2: コミュニケーション図とタイミング図の合成イメージ

尚、本研究では使用するツールの関係上UML1.5で行うため、UML2.0で新たに追加されたタイミング図の利用は不可能であるため、2つ目のコミュニケーション図とタイミング図の合成に関しては提案のみという形にする。

3.3. 整合性に関する検討

複数のUML図を扱うには、UML図間の整合性を示す必要が出てくるが、その整合性は2つに分けられる。

1つ目はモデルの意味にまで踏み込んだ整合性であるセマンティクス、2つ目はモデルの記述面の整合性であるシンタックスがある。

本研究で扱うUML図は、元々1つの振る舞いを異なる視点で捉えた図であるので、合成してもセマンティクスは保たれる。シンタックスについてもそれぞれの図の情報をそのまま過不足無く表現するので整合性は保たれる。もし、シンタックスについて細かく調べるならOCLを利用した証明方法もある。[6]

4. 試作するツールの詳細

4.1. 既存のツールの説明

本研究では、作業を円滑に進めるために既存のツール

「Poseidon for UML」[4]と「じゅん」[5]を利用する。

「Poseidon for UML」はUML図の描画とXMIデータの出力の機能を持つフリーのUML描画ツールである。

「じゅん」はオープンソースの3D描画ライブラリである。

4.2. プロセスの検討

ユーザはまず「Poseidon for UML」を用いてコミュニケーション図とシーケンス図を描く。そして作成したUML図をXMIデータに出力して保存する。次にそのXMIデータを本研究で作成するコンバータで読み込み、「じゅん」のフォーマットに合うデータ構造へと変換する。そしてコンバータによって作成されたデータは「じゅん」を利用した3D描画システムに読み込まれOpenGLによりユーザへと出力される。ユーザはその画像に自由に操作を加えることができる。

図4-1に全体の流れ図を挙げる。

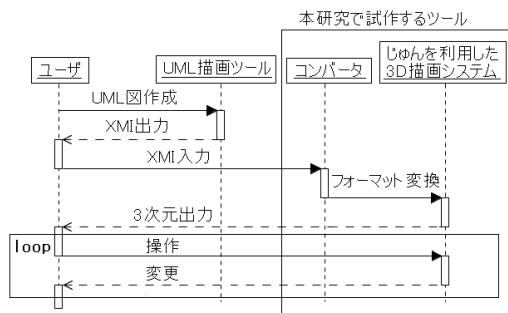


図4-1: 全体の流れ図

本研究で試作するツールは図4-1にあるようにXMIデータの変換、3D描画の部分を行う。

4.3. ツールの機能

ツールに持たせるべき機能としては、まず第1に視点の移動・回転が挙げられる。3次元図であるので、視点を移動させる必要があるからである。

次に3D図形に対する直接操作が挙げられる。図形の配置を変更することでより見やすい図にすることができるからである。

他にはオブジェクト名等のポップアップ表示が挙げられる。3D図形なので文字の表現が難しく、直接表示では見難くなってしまう。図の理解のためには説明が必要である。選択オブジェクトに対するポップアップを利用することによって、この2つの矛盾点を克服することができる。

最後にXMIデータへの出力である。図に対して直接変更を加えることにより元図形を変更する必要がある。この際に、XMIデータへ出力することができれば、元画像を変更することができるからである。

4.4. 試作ツールの現状

本研究で試作する部分はコンバータと「じゅん」を利用した3D描画部分である。現状は設計の段階であり、実装にはいたっていない。コンバータのデータ変換方法や3D描画部の3次元モデルに対する操作方法等を検討している。

4.4.1. コンバータの設計

XMIデータからオブジェクト、リンクを読み込んで簡単なデータベースを作成する。その際、座標データを「じゅん」のフォーマットに合うように変換する。

4.4.2. 3D描画部の設計

コンバータにより作成されたデータベースの情報をもとに、「じゅん」を用いて3次元モデルを表示する。主にマウスを用いて、物体の移動や視点の移動等の操作を行う。

5. まとめと今後の課題

試作ツールは未完成で直接検証することはできていないが、他の3次元描画ツールや作成した3次元モデルより、2次元図形より多くの情報が表現でき、また2次元図形では得られない視点を得ることができた。

これより、3次元図を用いて表現することで、プログラムに対する理解をより深めることができると言えるであろう。

今後の課題としては、XMIデータの解析、作成する図形ビューの改善、ツール機能の検討を引き続き行っていく必要がある。

また、各種技術のバージョンアップに伴うツールの制限や、UML図形の変更による整合性や互換性に対し、新たに対応していく必要がある。

参考文献

- [1] 小池英樹, 石井威望: 3次元ソフトウェア視覚化の枠組と実例による有効性の評価, 情報処理学会論文誌, Vol33, No.6, (1992).
- [2] Scott W .Amber: アジャイルモデルのエッセンス, アジャイルに作れる成果物
<http://www.ogis-swe.jp/process/am-res/am/artifact/>
- [3] OMG <http://www.omg.org/>, UML <http://www.uml.org/>, XMI <http://www.omg.org/technology/documents/formal/xmi.htm>
- [4] Gentleware, just model <http://www.gentleware.com/>
- [5] じゅん <http://osb.sra.co.jp/Jun/>
- [6] 吉積邦浩, 「整合性を考慮したOCLからSQLへの変換に関する研究」, 北陸先端科学技術大学院大学情報科学研究科2002年修士論文