

オントロジーに基づくソフトウェア開発上流工程支援

Supporting Software Engineering Processes with Ontologies

峰岸 巧^{*1} 石橋 昌彦^{*1} 福田 直樹^{*2} 飯島 正^{*1} 山口 高平^{*1}
 Satoshi Minegishi Masahiko Ishibashi Naoki Fukuta Tadashi Iijima Takahira Yamaguchi

^{*1} 慶應義塾大学
Keio University

^{*2} 静岡大学
Shizuoka University

The object-oriented approach is the mainstream of recent software development, and the modeling method with UML is widely used. Getting into object-oriented analysis, it has some issues for lack of knowledge in application domain. From the above-mentioned background, this paper conceptually designs a system that supports users in generating analysis class diagrams from user specific information, taking two kinds of ontologies: general ontologies and domain specific ontologies. By doing the match between nouns in use case descriptions and concepts from a domain ontology or superordinate concepts from a general ontology, the system can extract necessary concepts, data properties and object properties to class. We have done two case studies; stock management and sales management. They have shown us that this methodology can generate entity classes which relate to domain and relations between classes. Thus the proposed methodology turns out to reduce the costs of the generation of analysis class diagrams..

1. はじめに

現状のソフトウェア開発は、オブジェクト指向によるソフトウェア開発が主流であり、オブジェクト指向での開発では、モデルを用いた開発が進められることが多い。統一モデリング言語: UML(Unified Modeling Language)[1]を用い、ソフトウェア開発の各工程において、UMLを用いてモデルを構築し、設計、実装に落とすという方法で開発が行われている。現在はモデル作成、つまりモデリングの重要性が謳われ、なおかつユーザの求めている要求が複雑で、変化が非常に早いことから、そうしたユーザの要求を分析し、システムの機能を定めていく、いわゆる上流工程が注目されている。

本稿では、オブジェクト指向ソフトウェア開発の上流工程、特に、システム分析フェーズの開発支援を目的とする。より具体的にいうと、システム分析フェーズにおいて成果物となる分析クラス図作成の支援を行う。ここで、この工程の問題点を挙げると、以下のようなになる。

- (課題 1) 分析者の対象業務に関する理解不足により、モデリングが困難である点
- (課題 2) 分析者がシステム固有の情報とドメイン共通の情報を分離できない点
- (課題 3) 成果物に対する根拠の欠如によって、個々の解釈の相違が生じる点

上記の問題を解決するために、本稿では、オントロジーを利用した分析クラス図作成支援手法を提案する。人工知能研究の分野におけるオントロジーとは、「概念化の明示的な仕様」と定義され、これを利用することで、複雑な知識を構造化し、知識の獲得・利用・維持が容易となると考えられている。ユースケース記述を入力として、2種類のオントロジーを参照し、分析クラス図の初期段階を作成することにより、この工程での支援を行う。具体的には、ユースケース記述から名詞を抽出し、対象業務に特化したドメインオントロジー及び一般的な語彙を定義した汎用オ

ントロジーとのマッチングを行うことにより、クラス候補を抽出し、そのクラス候補同士の関連を導き出す。

本稿の構成として、続く2節では、基本アプローチを述べる。3節では、提案手法を詳細に説明する。4節で、ケーススタディについて述べ、5節では、まとめ・課題・今後の方向性について述べる。

2. 基本アプローチ

本文前述の問題点に関するアプローチとして、まず分析者の業務知識の理解不足に対して、業務知識を参照しながら分析を行える環境を提示することが望ましいと考えられる。また、成果物に対する根拠の欠如に対して、モデルの根拠となる意味に関して明確に記述したものを共有することで、あとからモデルの根拠を追跡することが可能になると考えられる。

以上から、本稿では分析クラス図作成の支援として、概念定義と概念間の関係を明確に記述した、オントロジーを利用することを提案する。

さらに分析者の情報を切り分ける力の欠如に対して、システム構築の対象を3つの役割に明確に分割して分析を行うアプローチを考えた。ここでいう3つの役割は、以下のとおりである。

- 各ドメインに特化しない最も一般的な概念
- システムの対象となる業界・業務に特化した概念
- ユーザに固有の独自の概念

これら3つの概念を分離して、対象の分離を図った。上記のオントロジーの利用に際し、この概念の分離を図るため、オントロジーを役割によって分類して用いることにした。

また、分析クラス図の成果物として、本来ならばクラスにオブジェクト間のやりとりを記述した操作を含むが、本稿では、クラスの操作を決定する部分の支援を行わないことにした。その理由として、次のように考えている。

システム内部のやりとりは、ドメイン共通の部分というものがなく、開発する各システムに完全に固有のやりとりが行われる。従って、各システムに共通部分をまとめて体系化するという方法を

採用することは、共通部分の存在がほぼないことから、かえってコストがかかり、支援として意味を成さないからである。

逆に、クラスの決定やクラス間の関連の決定に関しては、ドメインに依存してある程度の共通に理解できる部分が存在するため、この部分に関してオントロジーを用いて、さらにそのオントロジーを役割ごとに明確に使い分けることで、分析クラス図作成を支援できると考えている。

以上により、本稿で扱う分析クラス図を、属性とクラス間の関連を含んだクラスまでと定義する。

3. オントロジーを用いた提案手法

本節では、オブジェクト指向分析における要求分析の成果物であるユースケース記述が得られていることを前提に、この記述から抽出した名詞を対象ドメインにおける概念を体系化したドメインオントロジーと汎用オントロジーを参照して、分析クラス図を作成する工程を支援する。3.1 節で、提案手法のアーキテクチャについて述べ、3.2 節では利用する 2 種類のオントロジーについて、叙述する。

3.1 提案手法のアーキテクチャ

提案手法のアーキテクチャは、以下の 4 つの工程から構成されている。

1. ユースケース記述からの名詞抽出・英語への変換
2. ドメインオントロジーとのマッチング
3. 汎用オントロジーとのマッチング
4. 初期的な分析クラス図の作成

まず、形態素解析ツール「茶筌」[2]を用いて、自然言語で記述されたユースケースから、自動的に名詞を抽出する。この際、単純に茶筌を使って形態素解析を行うと、複合語の名詞を細かく分解してしまうため、専門用語抽出モジュール[3]を用いて形態素解析を行い、使用頻度の高いものから、複合語を含めて名詞の抽出を行う。続いて、その名詞を日本語から英語への変換を行う。この変換の理由として、利用するオントロジーの記述言語の依存性が挙げられる。今回使用する、ドメインオントロジー・汎用オントロジーは共に、概念の体系化を英語で行っていて、そのマッチングを行うためには、ここでの英語への変換は不可欠である。

続いて、ドメインオントロジーを参照することにより、抽出した単語の関係性を獲得する。ドメインオントロジーには、is-a 関係・has-a 関係以外の概念間の関係性が定義されている。この関係性は is-a 関係や has-a 関係のような一般的な関係性ではなく、より具体的かつ対象領域に合致したものである。この関係性を利用することにより、抽出単語の名詞同士が対象領域で一般的である関係性によって、結合することが可能になる。

その後、汎用オントロジーを参照することにより、ドメインオントロジーにおいてマッチングしなかった単語の関係性を獲得する。ユーザに固有な情報が、必ずしもそのドメインに共通に用いられる言葉と一致するとは限らない。このような単語に対応するために、ドメインに特化しないより一般的な概念の階層構造を利用する。具体的には、抽出した単語がドメインオントロジーの概念と一致しない場合、汎用オントロジーの階層の下位(包含関係)にある概念とのマッチングを行い、単語間の関係性を抽出する。

最後に、これまでの工程で得られた概念をクラスとして、関係性をクラス間の関連として、そして、データプロパティを属性として分析クラス図を作成する。

抽出名詞とオントロジーとのマッチング・分析クラス図作成の構造を図1で示す。

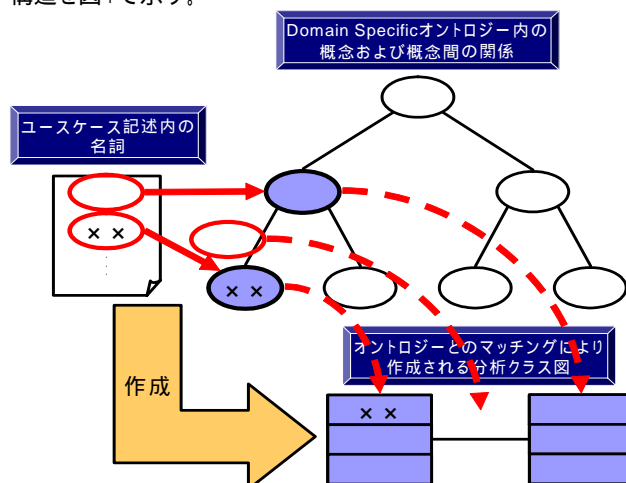


図1 概念および関係のマッチングによる分析クラス図作成

3.2 利用オントロジーについて

本稿では、前節で述べたように、分析クラス図作成において、概念のレベルごとに 3 つの役割に分類して支援を行う。その中で、各ドメインに特化しない最も一般的な概念とシステムの対象となる業界・業務に特化した概念をオントロジーとして用意する。

前者のドメインに依存しない汎用的な抽象概念を体系化したオントロジーとして、本稿では汎用オントロジーである WordNet[4]を用いる。

後者の業界・業務に特化したドメインオントロジーとしては、各システム開発に応じて、個別に用意するものとする。

適切なオントロジーが利用できなかった場合、ドメインオントロジーの構築を行う必要がある。作成するオントロジーは、ドメインに固有の概念をリストアップし、概念間に関係を記述する。この際、リストアップした固有の概念のみに関係をつけるだけでは、意味的に不自然になるので、上位概念を適度に補って、不自然に関係をつなぐことの無いようにしたほうがよい。

また、オントロジーの体系化の方法として、オブジェクトオントロジーとプロセスオントロジーの 2 つを用意する。プロセスオントロジーの方には、ドメイン内のプロセスに関する要素を含めて体系化を行い、オブジェクトオントロジーには、プロセスオントロジーの入出力やアクタなどの構成要素を、階層構造(is-a 関係)を中心に体系化を行う。プロセスオントロジーに関しては、関係をできる限り記述するようにする。具体的に作成したオントロジーに関しては、次節で説明するが、ここではあまりコストのかかるような複雑なレベルの概念定義や関連の定義を強要しない。むしろ、今回のように、システムの分析モデル作成という目的に相応しいレベルのオントロジーを考えると、必要となる概念要素がある程度の粒度以上の細かさを要求していないのである。こうしたオントロジー構築のコストと効果に関わる内容は、後ほど考察にて検討する。

4. ケーススタディ

本節では、前節で提案した手法を代表的な 2 つの業務システム開発事例に適用する。まず、2 つの事例(在庫管理、および販売管理システム)について説明し、次に、その際に用いる参照用のドメインオントロジーについて述べ、結果と考察を行う。

4.1 ケーススタディの事例

ソフトウェア工学の世界において、かつて複数の設計手法の比較のために作成された例題として、「共通問題」[5]というものがある。この共通問題である在庫管理に関するシステム(事例1)、販売管理に関するシステム(事例2)の2つの事例に関してケーススタディを行った。このシステムの詳細は省略する。

4.2 ケーススタディに用いるオントロジー

今回事例として取り上げた2つのシステムが、共に販売管理(在庫管理を含んでいる)を対象としているため、オントロジーが対象とするドメインとして、販売管理を設定した。以降、このオントロジーを販売管理オントロジーと呼ぶことにする。

販売管理オントロジーは、まず企業における販売業務を受注・出荷・売上・請求・回収と分解した。また逆に、調達業務を発注・入荷・仕入・支払いと分解した。このように分解された各業務をプロセスオントロジーの中心となるプロセスと位置づけ、各業務に関係する要素と関係を定義して、プロセスオントロジーを構築した。簡単に言えば、トランザクションを中心として、各トランザクションに関して、プロセススキーマにあたる要素を定義してオントロジーを構築している。

一方、それぞれの業務において必要となるリソースやロール、場所に関して体系化したものがオブジェクトオントロジーである。基本的に is-a 関係を用いて記述されている。一般的に必要な概念要素を、あまり細部には意識せず、多くの人間が理解できるレベルのものを意識してピックアップし、これらをオブジェクトオントロジーとしている。構築したオブジェクトオントロジーを、以下の図2に示す。

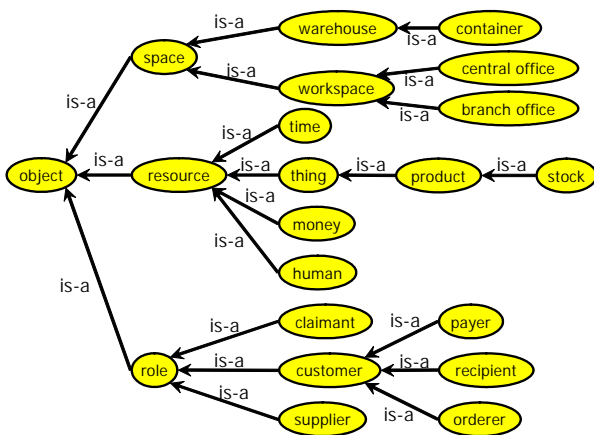


図2 オブジェクトオントロジー

そして分解された各業務が、対象となるリソースや、業務に関連するロール、そして場所を関係として定義している。プロセスオントロジーのサンプルとして、構築した出荷プロセス(業務)のプロセスオントロジーを、図3に示す。なお、今回オントロジーの構築には、代表的なオントロジー構築ツールである Protege2000[6]を用いている。

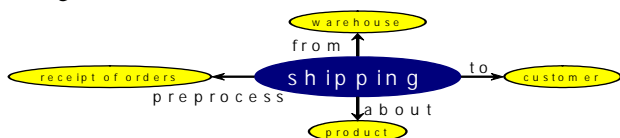


図3 プロセスオントロジー

4.3 結果と考察

この節では、提案手法を適用した結果、作成された分析クラス図を人手により作成した分析クラス図と比較し、提案手法が支援できた部分とそうでない部分を明確にする。

事例1(在庫管理システム)

前述の販売管理オントロジーと WordNet[4]を参照して、提案手法を適用した結果、以下の図4の分析クラス図が作成された。比較のために、続けて人手による一般的な分析クラス図の例を図5で示す。

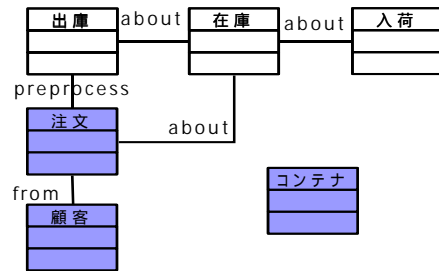


図4 提案手法により作成された分析クラス図(事例1)

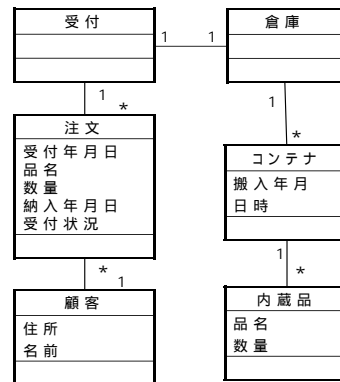


図5 一般的な分析クラス図(事例1)

提案手法により支援できた部分として、両方の分析クラス図の一致する部分を考えて、注文クラス・顧客クラス・コンテナクラスの3つのクラスと注文クラスと顧客クラスの関連を抽出することができ、この部分は支援することができた。関連については、一般的なクラス図よりも詳しい情報が盛り込まれている。

一部支援できた部分としては、提案手法では、出庫クラス・入荷クラスが作成されたが、この両者は、倉庫に関連したものであると考え、両者をひとつにまとめて、倉庫クラスと洗練することも可能である。

支援できなかった部分としては、バウンダリクラスつまり、この事例における受付クラスの抽出とこの事例に特有の内蔵品という概念のクラス抽出が挙げられる。また、属性・多重度についても、まったく支援できなかった。

事例2(販売管理システム)

事例1と同じ2種類のオントロジーを参照して、提案手法を適用した提案手法を適用した結果、以下の図6の分析クラス図が作成された。比較のために、続けて人手による一般的な分析クラス図の例を図7で示す。

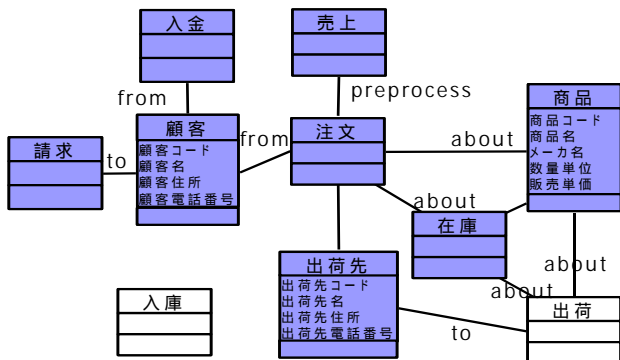


図6 提案手法により作成された分析クラス図(事例2)

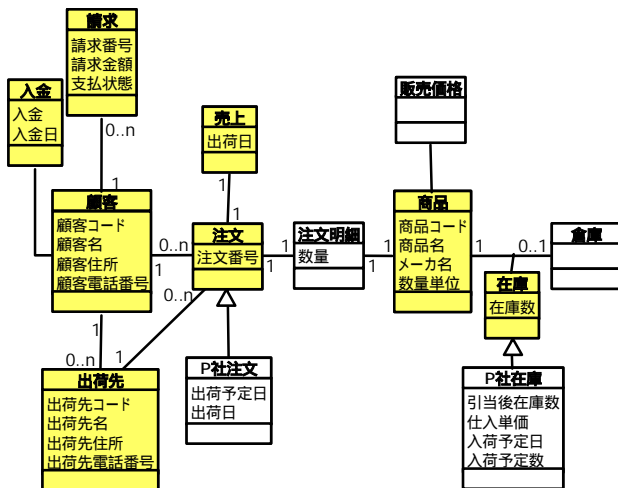


図7 一般的な分析クラス図(事例1)

この事例で支援できたのは、顧客クラス・商品クラス・出荷先クラス・在庫クラス・売上クラス・注文クラス・請求クラスの8つのクラスの抽出とそれらクラス間の関連の抽出である。

一部支援できた部分としては、事例1と同様に在庫クラス・出荷先クラスを倉庫クラスと洗練できることである。また、属性に関しても、全てを網羅することはできないが、一般的に考えられる属性だけは抽出することができた。

支援できなかった部分としては、注文明細クラスとP社注文等のシステム特有のクラスである。また、多重度についても、まったく支援できなかった。

2つの事例の結果より、提案手法によって、作成されたクラス図の特徴を把握すると共に、支援できずに人間がすべきことについての考察を行う。

支援できた部分として、エンティティ・バウンダリ・コントロールの責務のクラス抽出であるがあげられる。この責務のクラスは、実際のクラス図のクラス図のものほぼ全て抽出されていた。また、クラスの関連に関しては、オントロジー上で概念間の関係性について明確な定義を行っているので、クラス図の関連は、実際のクラス図より、厳密な定義の関係が抽出することができた。

次に、一部支援できた部分について触れる。クラスの属性については、典型的なものをデータプロパティとして記述してあるので、それについては、支援できるのだが、網羅性に欠け、人手による追加が求められる。

最後に、支援できない部分として、システム固有の情報に関する部分・クラスの多重度に関する部分・関連クラスに関する部分が挙げられる。関連クラス特に重要なトランザクション明細については、分析者の自由度が強い点も考慮し、支援の対象とは

考えないことにしたが、その重要性を考えると、支援の検討性を考えるべきである。

5. おわりに

本稿は、オントロジーベースのソフトウェア開発の実用性を探る位置づけであり、対象領域・利用オントロジーを限定した研究である。ソフトウェア開発に、オントロジーを利用することは、理論的には、十分可能であるが、具体的な使用オントロジー・適応手法・支援対象は、いまだ定まっていないのが現状である。

利用するオントロジーに関して述べると、本稿では、販売管理に関するオントロジーを作成したが、実際オントロジーを構築する際、情報量にもよるが、厳密に概念や関係を定義するとなると、コストがかかっていくことが必然的である。オントロジーを利用する際、構築におけるコストと利用におけるベネフィットは、トレードオフの関係にある。従って、オントロジー構築のコストが極力かからないことが望ましい。現在一般に公開されているオントロジーの数は、ビジネスのプロセスに注目した Process Handbook[7]など33万を超え、この数ヶ月で2倍増えているという状況、つまり日々オントロジーの数は増え続けていることを意味する。こうした中から自分に必要なオントロジーを獲得するのは難しいが、オントロジー用の検索エンジン Swoogle[8]などの登場で、日々改善されている。こうして自分に適切なオントロジーを獲得し、修正・利用することで、提案手法の適用による効果の拡大が期待できる。そして、ドメインオントロジーに複数オントロジーを適用することで、分析者に選択肢が増え、より最適なソフトウェア開発ができることが期待される。

オントロジー利用方法については、今後さらなる検討を加え、個々のユーザに要望に応じた柔軟性のある支援を行う必要がある。本研究では、成果物として分析クラス図を一種類だけ出力し、それをもとに開発者が洗練していくというスタイルをとったが、ユーザの意図により適切に応じるには、複数の案の中から選択させる方がよいと考えられる。

支援対象領域に関しては、本研究はシステム分析工程に限定したが、より上流工程の要求分析工程や業務分析工程もしくは、より下流の工程にもオントロジーを利用した開発支援が出来ることが期待される。オントロジーの利用方法は、工程ごとに異なると共に、オントロジー自体も各開発フェーズに対応した粒度である必要がある。開発フェーズに応じたオントロジー・適応手法を考案し、全ての工程においてオントロジーベースのソフトウェア開発が行われることが最終的な目標である。

参考文献

- [1] OMG Unified Modeling Language Specification, <http://www.omg.org/docs/formal/03-03-01.pdf>
- [2] 松本裕治他, 「形態素解析システム『茶釜』version 2.3.3 使用説明書」, 奈良先端科学技術大学院大学, 2003.8
- [3] 中川裕志他, 「専門用語(キーワード)自動抽出システム」, <http://gensen.dl.itc.u-tokyo.ac.jp/>
- [4] WordNet, <http://www.cogsci.princeton.edu/wn/>
- [5] The Protege Project, <http://camis.stanford.edu/projects/protege/>
- [6] 山崎利治, 情報処理学会誌「情報処理」25巻(1984)9号
- [7] MIT Process Handbook Project, <http://ccs.mit.edu/ph/>
- [8] Swoogle, <http://swoogle.umbc.edu/>