

# 異種オントロジー構築に基づく情報システム開発支援 Development Support of Information Systems Based on Heterogeneous Ontologies

近藤 恵一\*<sup>1</sup> 森田 武史\*<sup>1</sup> 和泉 憲明\*<sup>2</sup> 山口 高平\*<sup>1</sup> 橋田 浩一\*<sup>2</sup>  
Keiichi Kondo Takeshi Morita Noriaki Izumi Takahira Yamaguchi Koiti Hashida

\*<sup>1</sup> 慶應義塾大学 理工学部  
Science and Technology, Keio University

\*<sup>2</sup> 産業技術総合研究所  
National Institute of Advanced Industrial Science and Technology

In this paper, we propose a development method for Web application systems based on heterogeneous ontologies. Our method models objective business (domain) as ontologies from three aspects: legal aspect, business process, and software design. And to provide specifications of the system, we adopt Web page transitions as the base of Web application model, named Web Process Architecture. Ontologies are combined and applied as factors of Web Process Architecture.

## 1. はじめに

現在、企業をはじめとして、業務に情報システムを用いることが当然となり、ユーザーの多様なニーズにより、システムに用いられるソフトウェアは複雑化している。このような業務アプリケーションの開発においては、ユーザーの要求だけでなく、ビジネスルールや各種法規に合致したソフトウェアを開発する必要がある。また、本研究の主な対象である、自治体や官公庁の業務は、一般的な業務とは異なり、より強く法律により制限を受けている。法律とソフトウェアの対応関係が無い状況では、法改正に伴うシステム変更にも莫大なコストがかかっている。

上述したユーザーの要求などによるソフトウェアの複雑化に対し、ソフトウェア開発の効率化を目的として開発手法やソフトウェア部品、フレームワークが考案され効果をあげている。例えば、ソフトウェアの設計仕様を記述する言語である UML (Unified Modeling Language) が、ソフトウェア開発における設計の統一記述法として定着しつつある。しかし、UML は開発者向けの記述法であり、ソフトウェアを知らないユーザーは、その記述内容を理解することは難しい。こうした現状で、ユーザーはソフトウェア開発に参加するのが困難であり、開発のほとんどを開発者に任せてしまうことが多い。結果として、開発においてユーザーと開発者との共通理解がうまくいかず、ユーザーの要求が開発成果に反映されないという問題を引き起こしている。

そこで、本稿では、ユーザーの要求や法律といった概念を表現する手段として、オントロジーに注目する。オントロジーはある対象領域(ドメイン)に存在する知識を体系化するものとして、人工知能の分野で研究が行われてきた。近年、セマンティック Web の分野では機械可読な語彙体系、Web 上に知識を表現する手段として利用され、幅広い応用が期待されている。関連研究として、ビジネスプロセス記述へのオントロジー応用の研究がいくつか発表されている。例えば、MIT の e-ビジネスプロセスハンドブック [Malone 03] やエジンバラ大学のエンタープライズオントロジー [Ushold 98] などが挙げられる。しかし、これらはソフトウェア開発にそのまま利用できるほどの形式性を持っていない。

以上のように、現在のソフトウェア開発において、ユーザーの要求や法律といったソフトウェア開発の直接の対象ではない概念を仕様として形式的に記述することが難しく、ユーザーと開発

者との共通理解がうまくいかないことが多い。そこで、本稿では、Web アプリケーションに特化して、情報システム開発にオントロジー技術を応用する。業務のモデルとしてオントロジーを構築し、ソフトウェアの仕組みを知らないユーザーでも理解容易な画面遷移のレベルまでオントロジーに基づいた設計を行う。こうすることで、ユーザーと開発者との共通理解を促し、ユーザーの要求を反映させた情報システムの開発を目指す。

## 2. オントロジーの構築

ここでは、オントロジー構築に基づく業務モデリングの方法を述べる。また、それに先だって、手法全体の概要を以下に示す。

### 2.1 提案手法の概要

本手法の目的は、オントロジーに基づいてユーザーの要求を記述することにより、情報システム開発に必要な情報を提供することである。本手法の全体図を以下の図1に示す。

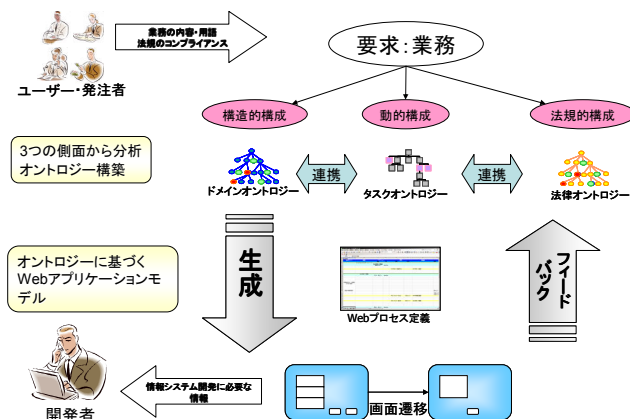


図1 手法の概要

まず、対象業務であるドメインを3つの側面(構造的構成、動的構成、規約的構成)から整理し、ビジネスオントロジーを構築して、これをドメインモデルとして採用する。つづいて、ビジネスオントロジーをもとに、Web アプリケーションの画面遷移構造を基にしたモデルである、Web プロセス定義を構成する。Web プロセス定義は Web アプリケーションの開発仕様として必要な

連絡先: 慶應義塾大学理工学研究科, 〒223-8522 神奈川県横浜市港北区日吉 3-14-1 矢上キャンパス 24-619A

要素を表形式で一元的に記述するものである。構築したオントロジーの要素は、この Web プロセス定義を記述するために用いる。そして、ユーザーと開発者は、この Web プロセス定義において画面遷移レベルで共通理解をし、最終的に、Web プロセス定義をもとに実装する。

## 2.2 ビジネスオントロジーの構成と構築

まず、対象業務から Web アプリケーションを開発するために、対象業務を 3 つの側面(構造的構成、動作的構成、規約的構成)から分析し、オントロジーを構築する。本手法では、それぞれの側面から構築したオントロジーを順番に、ドメインオントロジー、タスクオントロジー、法律オントロジーと呼ぶ。以下の図 2 に各オントロジーの概要と、既存ドキュメントとの対応関係を示す。

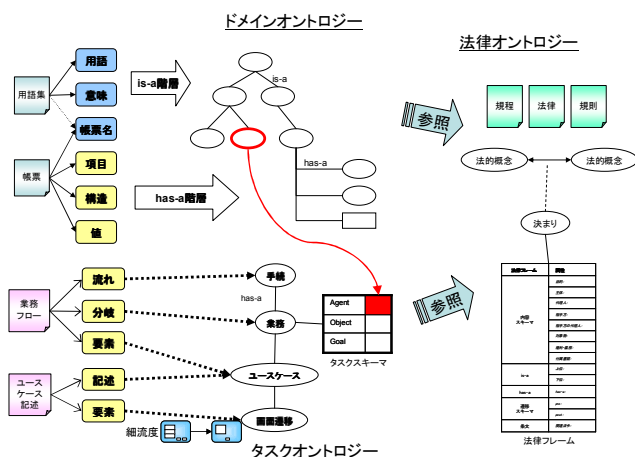


図 2 異種オントロジーの概要

オントロジーを構築するにあたって、ソースとなるものは、業務マニュアルや関連法規、ユーザーからのヒアリングなど様々なものがある。本手法では、半自動オントロジー構築ツールである DODDLE-OWL[森田 06]を使用して、ドキュメント類から初期オントロジーを構築し、そこから手動で洗練を進める方法をとる。

以下に、3 つの側面から構築するオントロジーの特徴をそれぞれ述べる。

### (1) ドメインオントロジー

ドメインオントロジーは、対象領域(ドメイン)を記述するオントロジーであり、本手法ではドメインの構造的(静的)側面をオントロジーによって表す。ドメインオントロジーに求める形式性は必要最小限にとどめており、図 2 の左上のように、is-a 関係と has-a 関係を採用する。ここで、ドメインオントロジーには、各属性に SQL のタイプを付与し、データベーススキーマ設計の土台として活用する。この結果、ドメインオントロジーの記述を、UML における概念クラス図の記述に相当するレベルから、実装に近いレベルまで、統一して利用することが可能となる。それに伴って Web アプリケーション開発における語彙を統制する役割も担う。

### (2) タスクオントロジー

タスクオントロジーは、操作や処理などの動的な概念を記述することにより、ワークプロセスを定義するオントロジーである。ここで、ワークプロセスとは、業務におけるアプリケーションの操作手順などを規定するものであり、ワークフローに具体的な操作を定義したものである。Web アプリケーションにて実行されるタスク毎に、ドメインオントロジーの要素がどのように、Web 上の操作

を構成するのかを、スキーマにより記述するものがタスクオントロジーである。

Web アプリケーションの画面遷移をタスクの最小単位と捉え、これを基準として、細かな要素はアプリケーション側のプロセス、大きな要素は人が行う業務やユースケースに該当する。そして、タスクオントロジーのスキーマはドメインオントロジーの要素を使用して定義をする。Web アプリケーションを使ってタスクを行う Agent、タスクの対象物である Object、タスクの前提条件など、タスクを表現する属性を用意しタスクの意味を表す。画面遷移レベルのタスクでは画面遷移モデルを記述するために詳細なスキーマ定義が必要である。

### (3) 法律オントロジー

Web アプリケーションの構築には、ユーザーの要求だけでなく、法律や規定などとの関連を明確にすることが求められる。このような、法規的な概念を表現するためのオントロジーが法律オントロジーである。

図 2 の右側のように、法律や規則などのドキュメントにおける法的概念の定義と、概念間の動作や行動に関わる制約に関わる条文を、法律フレームに当てはめることで体系化する。

法律オントロジーはドメインオントロジーにおける要素やタスクオントロジーにおけるタスクが存在する根拠となる。また、ドメインオントロジーとタスクオントロジーの上位部分は法律オントロジーの構造を適用することができる。

## 3. オントロジーに基づく Web アプリケーションのモデリング

ここでは、オントロジーに基づいた Web アプリケーションのモデルである Web プロセス定義について述べる。

### (1) Web アプリケーションのアーキテクチャ

Web アプリケーションは、しばしば、3 層からなる構造で表現される。おおよそ、以下の図 3 のように、左から、Web ブラウザに表示され、ユーザーから内容が見えるプレゼンテーション層、画面遷移を司り画面からの入力処理や受け渡しを行うビジネスロジック層、データの実体が存在しデータベース入出力を行うデータアクセス層と定義される。

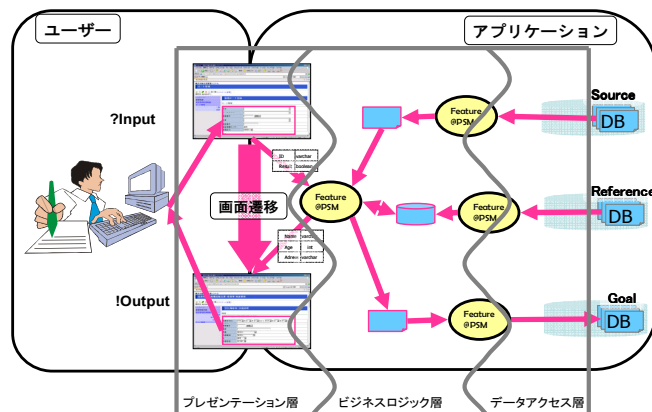


図 3 Web アプリケーションのアーキテクチャ

本手法でも、この捉え方を採用し、ユーザーから見た画面遷移を中心に Web アプリケーションをモデル化する。そして、このモデルの要素がオントロジーと対応するように、オントロジーを構成する。

ここで、本手法でフレームワークとして定義する概念を以下に示す。

- ・プレゼンテーション層
  - 最小単位のタスク、画面のコンポーネント(ラベル、ボタン、フォーム、リストなど)、画面遷移
- ・ビジネスロジック層
  - 画面遷移のコントロール、画面入出力のコントロール、入出力値条件、例外処理、フィーチャー
- ・データアクセス層
  - データベースへの入出力、対象データベーステーブル、データ構造(ドメインオントロジー)

## (2) Web アプリケーションのモデル

上述の 3 層構造のアーキテクチャに基づき、オントロジーの要素を用いて、フレームワークとして定義した概念を並べていくことで、Web アプリケーション開発のためのモデルを得ることができる。本手法では、表形式で Web アプリケーションのモデルを記述し、これを Web プロセス定義と呼ぶ(図 4)。

プレゼンテーション層			ビジネスロジック層			データアクセス層					
タスク	前ページ	後ページ	部品	要素	値条件	フィーチャー	オントロジー	データベース	I/O	法則参照	
会員登録	Page1		送信ボタン	ボタン		入力を受け取る					
			申請フォーム	Name	IsNull	データベース入力	申請フォーム	r_form	write	2-3	
				Age	>=16	データベース入力			write	2-4	
				form		データベース入力	申請台帳	ledger	write	1-2	
		Page2				ページを表示					
error	Page2		申請フォーム(リスト表示)								
			確認ボタン	ボタン							
			キャンセルボタン	ボタン							
error	Page3		申請エラー	エラー							

図 4 Web プロセス定義

この Web プロセス定義において、ソフトウェアのことを知らないユーザーであっても、最低限プレゼンテーション層である、画面の部品や遷移を把握できれば、開発者と同じ土台でコミュニケーションをとることができる。

## 4. Web プロセス定義に基づく実装

図 4 の Web プロセス定義は図 3 の Web アプリケーションのアーキテクチャと対応させて記述する。まず、オントロジーの要素を用いて、業務における Web アプリケーションの振る舞いを画面単位で記述する。そして、画面単位の記述を繋げることで、Web アプリケーションの画面遷移モデルを表現する。

ここで、本手法では、ビジネスロジック層における一般的なワークフロー処理は、画面遷移として取り扱われる。つまり、ビジネスプロセスの単位を Web アプリケーション上にて実行されるタスクであると見なす。Web アプリケーションの画面を基準とすることにより、ビジネスプロセスは画面遷移として説明可能となる。

### 4.1 Web プロセス定義に基づく実装

Web プロセス定義には、Web アプリケーションの画面遷移を実装するために必要な、画面部品や画面遷移コントロールが記述されている。そのため、細かなロジックや画面レイアウトを除けば、Web アプリケーションを実装するのに十分な情報量がある。

実装するにあたり、まず、ドメインオントロジーから、データベーススキーマを定義する。ドメインオントロジーを構築した時点

で、SQL のタイプを付加しているため、データベーススキーマとオントロジーを対応させることは容易である。そして、Web プロセス定義から画面遷移を実装すれば、Web アプリケーションの 3 層構造が完成する。その後、開発者は、残りの細かなロジックの実装に注力することができる。

## 5. 業務システム開発における実証実験

ここでは本手法を自治体の台帳管理システム開発に適用した結果[4]と、現在進行中の大規模システムへの適用実験を紹介する。

### 5.1 自治体業務システムの開発への適用結果

本手法を元に、業務マニュアルや帳票からオントロジーを構築し、オントロジーに基づき Web プロセス定義を記述した(図 5)。ドメインオントロジーの概念数は約 200、Web プロセス定義は 64 ユースケース分になり、各々数ページの画面遷移が含まれる。

本手法の評価として、Web プロセス定義とオントロジーから実装したシステム開発と、本手法を適用していないシステム開発を比較した結果を述べる。比較対象の両業務システムは、全体の開発規模が同等の約 18 人月で、かつ業務の対象分野は類似している。両システムの開発を比較した結果、提案手法の適用により、分析コストは約 40%に縮小された。この結果より、本手法により、ユーザーと開発者は実装に至るまでの期間、要求分析・設計に関する共通理解に効果があったと言える。

特に語彙統制や開発者の法規理解、ユーザーとの画面レベルでの共通理解に有効であったとの見解を開発者から得た。

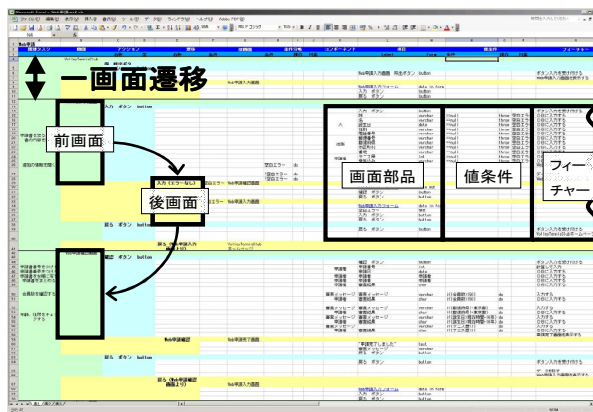


図 5 自治体の台帳管理システムにおける Web プロセス定義

### 5.2 大規模システム開発への適用

もう一つのケーススタディとして、産総研の財務会計システムへの適用と、これに伴うオントロジーの利用可能性について述べる。

#### (1) 大規模な業務オントロジーの構築

財務会計システムのオントロジー構築にはソースとなるドキュメントが複数ある。まず、構造的構成を含む帳票と用語集、動的な構成を含む業務フローとユースケース記述、法規的構成については、会計基準や業務規程、独立行政法人法など複数に分



かれる。これらのドキュメントから業務モデリングのためにオントロジーを構築していった。

まず、DODDLE-OWL を使用しドメインオントロジーの is-a 関係を構築した後、帳票の構造から has-a 関係を抽出し、オントロジーに加えて洗練した。法律オントロジーは、条文の構成をもとに法律フレームを定義し、フレームに条文を当てはめることで体系化した。タスクオントロジーはドメインオントロジーの要素を用いて、画面遷移に対応する粒度のタスクはユースケースをもとに、それ以上の大きいタスクは法律オントロジーと業務フローをもとに、体系化した(図 6)。

オントロジーの規模としては、ドメインオントロジーの概念数が約 1000、画面遷移レベルのタスク数が約 1400、法律オントロジーの概念数が約 200 となった。

今後、このオントロジーに基づき、Web アプリケーションのモデルである Web プロセス定義を記述していく。

(2) 大規模業務システムへの適用

今後、財務会計システムも医務システムのケースと同様に、オントロジーと Web プロセス定義に基づき開発を行っていく。ここでは、発展的なオントロジーと Web プロセス定義の利用について議論する。

まず、ドメインオントロジーとデータベースの対応があげられる。オントロジーにはデータベース固有の主キーなどの概念がないため、直接ドメインオントロジーをデータベースの論理モデルとすることはできないが、オントロジーの構造から自動的に論理モデルを生成し、そこからデータベースとして洗練することができる。

Web プロセス定義に関しては、現状のクロス表形式の記述から、Web アプリケーションのプロトタイプを生成することが考えられる。Web プロセス定義には画面上の部品や遷移の情報が含まれているため、細かいプロセスやインターフェースの仕様以外の項目は生成できる。また、今後、細粒度のタスクオントロジーから、直接 Web アプリケーションの画面を生成することも検討する。オントロジーから画面を作るルールを定めることで、Web プロセス定義の内容を自動生成し、画面を作ることができると思われる。そして、画面上でアプリケーションの動作や部品を直接編集し、オントロジーにフィードバックを与えることができれば、ユーザー参加型の開発を支援できると考えている。

そのほか、オントロジーの要素を用いて、既存のドキュメントを再構成することも可能である。ドメインオントロジーの要素名をドキュメントの記述に使用すれば、ぶれの無い記述にすることができる。また、タスクオントロジーにおけるタスクのつながりをフロー形式で表し、業務フローとして使用することも考えられる。

6. 結論

本稿では、オントロジーに基づいた業務モデリングと、Web アプリケーションに特化したシステム開発支援を提案した。事例では、本手法を用いて、構造的、動的、法規的の 3 つの側面から捉えたビジネスオントロジーを業務のモデルとして構築した。そして、Web アプリケーションのモデルである Web プロセス定義に構築したオントロジーの要素を集約し、設計仕様とした。医務システムの開発事例では、要求分析のコストを下げられたことにより、ユーザーと開発者との共通理解を助けることができた。

大規模システムの事例においては、本手法が大規模な開発にも適用できる事を検討した。現在、新たなシステム開発事例に適用中で、

今後の方針としては、今まで手で管理してきたオントロジーに対し、形式性を与え、自動的に整合性を保つことを目指す。それに伴い、Web プロセス定義とオントロジーも関連させ、オントロジーから簡単に画面遷移のモデルを記述できるようにする。また、現状で、ほとんど手動構築に頼っているドメインオントロジーの has-a 関係や、タスクオントロジーに関して、DODDLE-OWL の拡張により、半自動構築することを検討している。

文 献

[Malone 03] Thomas W. Malone, Kevin Crowston, and George A. Herman : Organizing Business Knowledge: The MIT Process Handbook, MIT Press (2003)  
 [Ushold 98] Mike Ushold, Martin King, Stuart Moralee, Yannis Zorgios : The Enterprise Ontology, Knowledge Engineering Review Vol.13, Cambridge University Press (1998)  
 [森田 06] 森田武史, 山口高平 : 日本語概念を対象にした領域オントロジー構築支援環境 DODDLE の機能拡張, 電子情報通信学会 2006 No.47, 信学技報(知能ソフトウェア工学研究会) (2006)  
 [Kondo 06] K.Kondo, S.Hoshii, T.Morita, N.Izumi, T.Yamaguti and K.Hasida : Semantics Driven Development of Software Systems Based on Business Ontologies, Joint Conference on Knowledge-Based Software Engineering 2006 pp.176-185, IOS-press (2006)

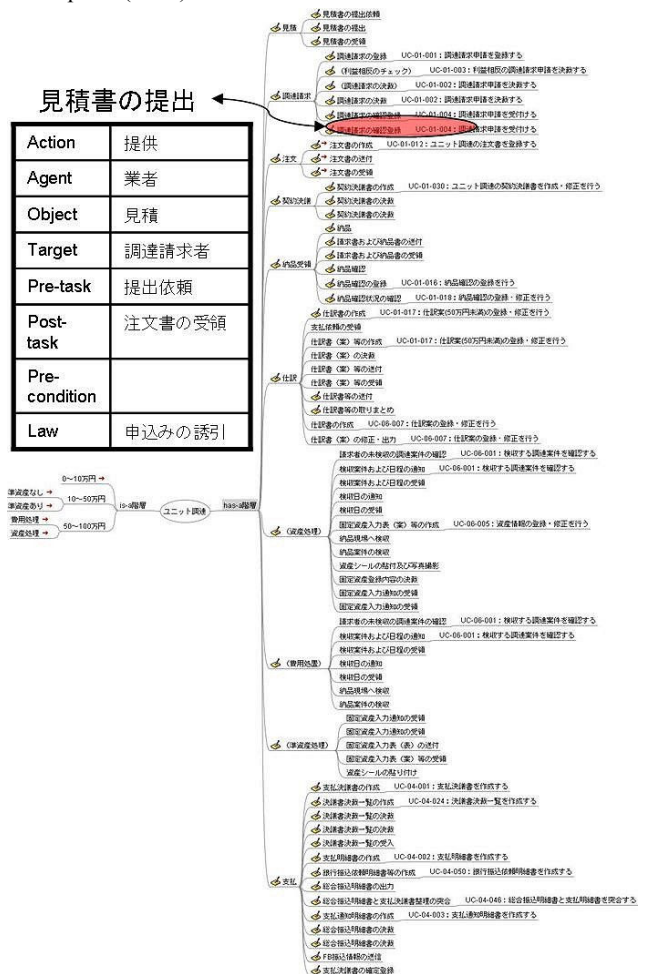


図 6 大規模システムにおけるタスクオントロジー