

オントロジーに基づくWebアプリケーション構築と評価

Developing Web Applications Based on Ontologies

近藤恵一[†] 森田武[†] 和泉憲明[‡] 山口高平[†] 橋田浩一[‡]

[†] 慶應義塾大学 [‡] 産業技術総合研究所

要旨

オントロジーを基にした Web アプリケーション開発手法を提案する。ビジネスプロセス、ドメイン構造、法規の3つの側面を捉えたビジネスオントロジーを構築し、ソフトウェアシステムのユーザーと開発者との開発における共通理解を促す。対象ドメインのモデルをオントロジーにより表現し、設計仕様としての記述力不足を Web アプリケーションの3層構造に基づいたモデル、Web プロセス定義を用いて補う。提案手法を事例に適用した実験により、本手法はユーザーと開発者の共通理解に有用であることを確かめた。

1. はじめに

近年、ソフトウェアは大規模化しており、その利用用途も拡大している。これと同時に、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として、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、ユーザー（発注者）からのソフトウェアに対する要求を分析する行程では、UML の様な形式的な言語は存在せず、各開発者グループ独自のやり方で要求分析を行っている。このような現状では、ソフトウェアをよく理解していない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、ソフトウェア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。

ここで、本稿の対象である業務アプリケーションのソフトウェア開発においては、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。さらに、政府や自治体の業務は、一般的な業務とはことなり、より強く法律に制限を受けている。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莫大なコストがかかっている。

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[1]や、エジンバラ大学で研究されたエンタープライズオントロジー[2]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、業務アプリケーション開発にオントロジー技術を応用する。そして、ユーザーと開発者の共通理解を促し、ユーザーの要求を反映した Web アプリケーションの開発を目指す。

2. オントロジーに基づく Web アプリケーションのモデル化

ここでは、提案手法として、オントロジーに基づき Web アプリケーション開発を支援する手法について述べる。提案手法は、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、Web アプリケーションの開発に必要な情報を構築することである。提案手法は、下図1のような構成になっている。

まず、対象業務であるドメインを3つの側面（構造的構成、動作的構成、規約的構成）から整理し、ビジネスオントロジーを構築する。これをドメインモデルとして採用し、Web アプリケーションのモデルである 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構造を与えるも

ので、Web アプリケーションの開発仕様として必要な要素を表形式で一元的に記述するものである。開発者は、最終的に、この Web プロセス定義に基づき Web アプリケーションを実装する。

以上のように、Web アプリケーション開発のドメイン分析がオントロジーに基づくことにより、分析から、設計、開発までの仕様の一貫性と再利用性が高まり、保守や改修にも活用できると期待している。

以下に、提案手法の各要素について Web アプリケーションの構造に従い、ボトムアップに説明する。

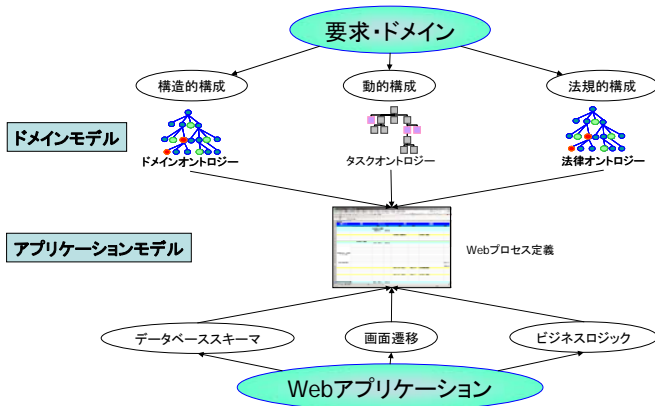


図 1 提案手法の概要

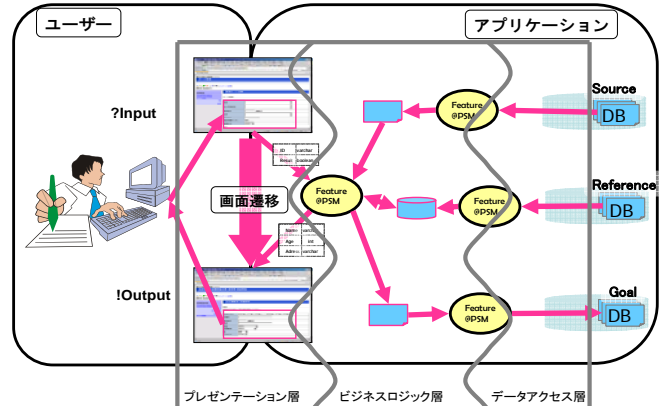


図 2 3 階層の Web アプリケーション構造

2.1. 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、おおよそ、図 2 のように、プレゼンテーション、ビジネスロジック、データアクセスの層から構成される。

本手法では、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し、Web アプリケーション開発の開発方法論 FDD (Feature Driven Development)におけるフィーチャーをオントロジーにより構成する。この 3 層構造の考え方を取り入れ、Web アプリケーションを図 2 のように 3 つに分割して捉える。具体的な構成は、一般的な捉え方と同じで、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からなる。本手法では、プレゼンテーション層に対象のタスク、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に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、データアクセス層に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 (オントロジー) をフレームワークとして定義する。

2.2. 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くビジネスオントロジーを用いて、Web アプリケーションの対象領域を書き下すことにより、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 3)。

図 2 と図 3 の対応は次のように与えられる。左側から画面に関するプレゼンテーション層の要素が並び、右側にデータベースに関わるデータアクセス層の要素が並び、中間にそれらを繋ぐビジネスロジック層の要素が並んでいる。これらの要素を用いることにより、業務において Web アプリケーションがどの様に用いられるのかを、画面を基準にして記述する。そして、画面の記述を繋げることにより、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。実際の業務は、すべて Web アプリケーションを用いて行われるわけではないため、本手法ではビジネスプロセスの単位を Web アプリケーション上で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。

2.3. ビジネスオントロジーの構成

上で述べた Web プロセス定義は Web アプリケーションモデルであるが、業務分析のフォーマットとして業務・システム最適化計画などで用いられる業務フロー図から直接得ることは難しい。そこで、本手法では、ドメイン分析をビジネスオントロジーに基づいて行い、Web プロセス定義を埋めるために必

要な情報をオントロジーに与え、オントロジーのノードが選択されて Web プロセス定義の内容となる。

ここで、Web アプリケーションのモデル化に必要な記述項目は、Web アプリケーションの3層構造と同様、異なる記述観点が必要となる。さらに、Web アプリケーション開発の基準となるドメインを記述する際も、複数の側面が必要となる[3]。そこで、本稿では、以下に挙げるオントロジーを構成する。

(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り構築する。本手法では、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、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 のタイプを付与することにより、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。

(2) タスクオントロジー

ドメインオントロジーは静的な対象領域の構造を表現することに対して、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クラスがどのように、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。タスクオントロジーのスキーマは、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の属性により定義する。

(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。そこで、本手法では、法規的な概念を表現するために法律オントロジーを構築する。あるタスクにおいて、なぜ、入力の実要求やその保存が必要なのかを、法律オントロジーへのリンクを与えることにより、明確にすることが目的である。Web プロセス定義では、図3のシートにおいて、一番右側に法律オントロジーとのリンクを与えている。Web プロセス定義の各項目が何故必要なのかは、法律オントロジーとのリンクにより明確化される。

タスク	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			注
	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	
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		データベースに入力	申請フォーム	r_form	write	2-3
				Age	>=16	データベースに入力			write	2-4
				form		データベースに入力	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 3 Web プロセス定義

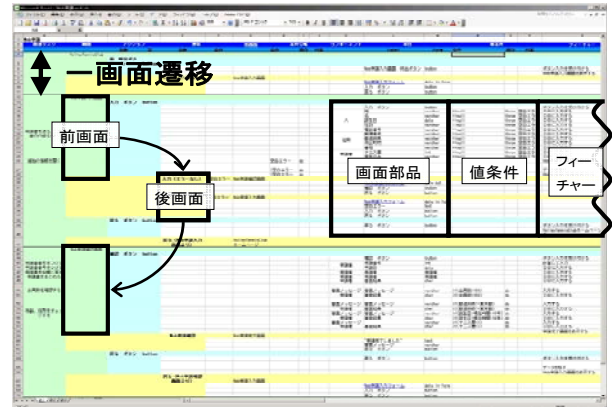


図 4 構築した Web プロセス定義

3. 事例適用の結果と検討

ここでは、事例に適用した結果について、オントロジーの有用性の観点から提案手法を検討する。さらに、大規模なシステムへの適応事例[6]の結果を照会し、そのスケーラビリティに関して検討する。

3.1. 事例における提案手法適用の結果と検討

手法を実施する事例として、Dietz のビジネスプロセス[4]を採用する。あるテニスクラブにおける入会処理の業務で、事前条件では全ての業務を手作業で行っている。

この事例に提案手法を適用し、DODDLE-OWL[5]を用いたビジネスオントロジー構築、Web プロセス定義の記述を行った。結果的に全9ページの Web アプリケーションとなった。上記の図 4 に完成した

Web プロセス定義を示す。ビジネスオントロジー、Web プロセス定義を使用した実装は非常に素早くでき、1 ページを1 時間程度で実装することができた。規模や期間に関しては次節で検討するものとし、ここではオントロジーの機能について検討する。

Web アプリケーションの実装に関して、産総研による Java フレームワークでは、一般のフレームワークと同様、データアクセス層の実装はドメインオントロジーを基に構築したデータベーススキーマから自動的に JavaBean を生成できる環境を用意している。データベーススキーマを先に構築できるので、ほぼ、自動で実装可能となった。次に、プレゼンテーション層の実装は Velocity を使用し、Web プロセス定義の記述から、ほぼ自動的にテンプレートを生成できた。

全体的には、開発者はプレゼンテーション層とデータアクセス層をつなぐビジネスロジック層の実装に集中できるので、効率的な実装が可能になったと思われる。本手法では、プログラムの詳細なロジックは、開発者の実装に委任している。ここで、ユーザーと開発者との共通理解という観点からは、Web プロセス定義は必要十分な記述力を有すると考えられる。

3.2. 自治体の業務システムにおける検討

ここでは、本手法を自治体の業務システム開発に適用した結果について述べる。

本手法を適用した業務システムと従来手法の業務システムの開発を比較した。ここで、両業務システム全体の開発規模は同等の約 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。ここでは、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ができた。

開発システムが大規模化した場合のビジネスオントロジーの可読性に関しては、規模が大きくなると、全体としての見通しが悪くなりという課題が確認された。一方、ユーザーは開発者よりも業務内容を熟知しているため、部分を限定しても業務に関する共通理解の確立が可能であった。

また、法律オントロジーは、法が改正されるまで有効であるため、再利用性が非常に高いことが分かった。さらに、法が改正された場合でも改正箇所をオントロジーのノード単位で示すことができる。Web プロセス定義の一要素として法律オントロジーを参照させているので、システム上での改正箇所が特定できる。同時に設計変更に対して、ユーザーは画面遷移レベルで変更箇所の理解が可能となる。

4. 結論

本稿では、Web アプリケーションの開発手法を提案した。本手法はドメインモデルとして構造的、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構築した。そして、Web アプリケーションのモデルとして Web プロセス定義とビジネスオントロジーを用いた定義法、ならびに、実装法を提案した。Web プロセス定義は Web アプリケーションの 3 層構造を基にしており、Web アプリケーションに必要な要素を一元的に記述できる。また、業務アプリケーション開発に本手法を適用した結果、Web アプリケーション開発において、ユーザーと開発者との共通理解を助けることができた。

今後は、より大規模なビジネスオントロジーの検討と開発への適用から、より高度な支援を目指す。

参考文献

- [1] M.Ushold, et al: The Enterprise Ontology, Knowledge Engineering Review, Vol.13, Special Issue on Putting Ontologies
- [2] Organizing Business Knowledge: The MIT Process Handbook, Thomas W. Malone, Kevin Crowston, and George A. Herman, Cambridge, MA: MIT Press, 2003
- [3] N.Izumi, T.Yamaguti, "Integration of Heterogeneous Repositories Based on Ontologies for EC Applications Development," International Journal of Electronic Commerce Research and Applications, vol.1, no.1, pp77-91, 2002
- [4] Jan L.G.Dietz: Enterprise Ontology –Theory and Methodology-, Springer-Verlag, Berlin Heidelberg, 2006
- [5] 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, 信学技報 (知能ソフトウェア工学研究会), KBSE2006-1, 2006.5
- [6] 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, no.35 pp.176-185, Tallinn, Estonia, August 2006