

依存関係管理に基づくオントロジー分散構築支援システムの試作

A Prototype System for Distributed Ontology Development Based on Dependency Management

砂川 英一
Eiichi Sunagawa

古崎 晃司
Kouji Kozaki

來村 徳信
Yoshinobu Kitamura

溝口 理一郎
Riichiro Mizoguchi

大阪大学 産業科学研究所
I.S.I.R., Osaka Univ.

Abstract. This paper describes a method for distributed ontology development and its implementation. In a distributed manner, we assume a situation where a target ontology is divided into several component ontologies which are constructed individually (perhaps in parallel) by different developers. Our goal is to support such a way of ontology development. In addition to general requirements for distributed development, we especially note the way for keeping the consistency of ontologies. We first investigate the dependency between component ontologies. And, based on it, we next consider a way for keeping their consistencies against the influence of the change which may destroy their consistency. Our system helps developers in many situations such as collaborative development, cooperative development and so on.

1. はじめに

知識システムの基盤となるオントロジーは様々なシステムやモデルに適用されるようになり、その価値や有用性は広く認識されつつある。特にここ数年は Semantic Web が次世代の Web として注目を集めており、そこで応用されるオントロジーは、世界レベルで研究活動が盛んに行われている。オントロジーの研究分野は多岐にわたるが、中でも構築の難しさから、その支援環境や方法論についての研究は急務であると言える。

本研究では、オントロジーは複数の部分的なオントロジーから構成されると捉え、それら部分となるオントロジーを個別また並列的に構築することにより、目的とするオントロジー全体を完成させるという「オントロジーの分散構築」の方法を検討している。このような形の構築過程を包括的に支援する計算機システムを開発することが、本研究の目的である[Sunagawa 03]。

分散構築はオントロジー開発の様々な段階や状況に適用可能であり、我々の提供する枠組みは開発者にとって大きく役立つと思われる。

2. オントロジーの分散構築

一般にオントロジーは複数のオントロジーに分割することが可能であり、また逆に、複数のオントロジーを統合して単一のオントロジーを構築することも可能である。この時、部分となるオントロジーは、含まれる概念のレベルや種類によって認識される。例えば、オントロジーを構築する際、既存のトップレベルオントロジーを利用し、その下に専門分野固有の概念を付け加えていく事は少なくない。このとき、構築されたオントロジー全体は、既存のオントロジーを再利用した部分と、新たに概念を追加した部分という2つのオントロジーの統合と見なすことができる。また、追加した部分のオントロジーは、新たに定義された概念の種類によって分類することが可能である。

このようにオントロジーの“部分”を考え、それぞれを個別のオントロジーとして扱うことのメリットには、①概念体系の全体像を把握しやすくする ②協調構築の布石となる ③オントロジーの再利用性向上に貢献する、などがある。具体的には、次のような分

散構築の形態を想定することができる。

- 構築前に概念を大まかに分類し、それぞれを別のオントロジーとして複数の構築者による分担作業で構築する
- 構築途中で上と同様のオントロジーの分割を行う
- 構築するオントロジーの一部として、他のオントロジーから切り出したものを再利用する

以下では、このような形態の構築を行うに当たっての方針や検討課題について述べる。

2.1 オントロジー間の依存関係管理

本研究が特に焦点を当てているのは、オントロジーの整合性に関する問題である。分割した部分的なオントロジーは、もともとは単一概念体系の部分であり、互いに影響を及ぼしあっている。よって、それらを個別に構築する際には、オントロジー間で不整合が生じやすくなると考えられる。これを解決するアプローチとして、我々はオントロジー間に存在する依存関係に着目し、依存関係の整合性を管理することによって、オントロジー全体の整合性を維持する方法を検討した[砂川 02]。

(1) オントロジー間の依存関係

一般に、オントロジーで概念を定義するときには、他の概念の定義内容を必要とする場合がほとんどである。上位概念からの定義の継承、部分概念に対するクラス制約などがそれに当たる。この時、それらの概念間には依存関係があるとみなすことができる。

異なるオントロジーに含まれる概念同士でこのような依存関係を持つとき、それをもとにオントロジー間の依存関係を捉えることが可能であり、本研究ではオントロジー間の依存関係として次の2種を定義した。

- ① **継承関係**: 一方のオントロジーで定義される概念が他方に含まれる概念の定義を継承しているとき、この二つのオントロジー間における関係を継承関係と定義する。この際、上位概念側のオントロジーを“上位オントロジー”、下位概念側のオントロジーを“下位オントロジー”と定義する。
- ② **参照関係**: 一方のオントロジーで定義される概念の部分概念や属性概念に対するクラス制約として、別のオントロジーに含まれる概念が使われている場合、この二つのオントロジー間には“参照関係”が成り立つとし、他のオントロジーの概念を使用している側を“参照オントロジー”、クラ

連絡先: 大阪大学 産業科学研究所 知識システム研究分野,
〒567-0047 大阪府茨木市美穂が丘 8-1,
Tel: 06-6879-8416, Fax: 06-6879-2123,
Mail: sunagawa@ei.sanken.osaka-u.ac.jp

ス制約として使用される概念を含む側を“被参照オントロジー”と定義する。

(2) 整合性保持のための依存関係管理

あるオントロジーの概念定義を別のオントロジーで使用した場合、その元になっている概念の定義が変化した際に、2つのオントロジーの関係に整合性がとれなくなり、構築しようとしているオントロジー全体に矛盾が生じる可能性がある

よって、オントロジー間の依存関係を構成する概念の定義に何らかの変更を加える際には、その依存関係が適切に保たれるように支援する必要がある。これには、変化するオントロジーの側でその変化の仕方々に制限を加える方法と、変化の影響を受ける側で整合性を保つために追従的な変化を行う方法の2つが考えられる。本論文では後者の手法を主として扱う。

あるオントロジーが変化した際、その変化の影響を受ける側(つまり、依存している側)のオントロジーで取ることでできる対応には以下の5通りのものがある。

1. 変化を受け入れ、それに合わせる形でオントロジーに追従的な変化を加える。追従方法は、変化の種類に依存する。
2. 変化を受け入れ、整合性に問題がない場合、依存している側での変更は行わない。
3. 変化を拒否し、それを打ち消すようにオントロジーに追従的な変化を加える。追従方法は、変化の種類に依存する。
4. 変化を拒否し、前バージョンのオントロジーに依存し続ける。オントロジー間の依存関係は保持し、更なる変化によっては再び最新のバージョンと依存関係を結ぶ余地を残す。
5. 変化を拒否し、依存関係を切り離す。影響を与えた概念を取り込み、独立して概念体系を構成できるオントロジーへと変化する。依存関係は切り離されるので、4. と異なり依存関係を結び直すことはできず、以後の変化の影響は全く受け付けない。

変化の種類	追従方法				
	1	2	3	4	5
継承関係					
概念レベルの操作					
概念の消去	1	-	1	1	1
下位概念の追加	2	1	-	1	1
定義レベルの操作					
ラベルの変化	1	1	-	1	1
スロットの消去	1	-	1	1	1
スロットの追加	2	1	-	1	1
...					
参照関係					
概念レベルの操作					
概念の消去	1	-	1	1	1
下位概念の追加	1	1	-	1	1
定義レベルの操作					
ラベルの変化	1	1	-	1	1
...					

表1. 概念変化のパターンと、それに対する追従方法の数(一部)

4と5はあらゆる変化の種類に共通であるが、1~3の適応可能性、及び、その具体的な追従方法は変化の種類に依存する。そこで変化の種類を分類するために、まずは影響を与える概念の操作についてパターンを考える。これは依存関係の種類で大別され、さらに概念そのものに対する操作のレベル(概念レベル)と概念の定義内容に関する操作レベル(定義レベル)に分かれる。例えば、概念レベルの操作には「概念の消去」「下位概念の追加」があり、定義の変化には「スロット(部分概念や属性概念)の消去」などがある。結果として、本研究では変化の種類を17パターンに分類した。そして、それぞれのパターンについて上記1~5の適応の可否やその詳細を考えたところ、合計で67通りの追従方法があるという結果が得られた。表1は、その一部を示したものであり、数字はその方法の数を示している。

例えば、“継承関係において、依存していた概念が上位オントロジーで消去された”というケースでは、表1に従って、全部で4つの選択肢が可能である。その詳細を述べると、1は「消去された概念の上位概念と依存関係を結びなおす」、3は「消去された概念を下位オントロジーで再び定義する」という追従方法である。

2.2 その他

本節では、依存関係に基づいた整合性保持問題以外に、オントロジーを構築するにあたって必要な機能を、構築される各オントロジーの管理、協調構築に向けてのオーサ管理、という2点から述べる。

(1) オントロジー管理

●バージョン管理

前述の依存関係管理を実現するため、システムには必要に応じて古いバージョンのオントロジーが保持されている。よって、システムが管理している依存関係情報に、それらのオントロジーのバージョン情報を加える事で、バージョンを含めたオントロジー管理は容易に行える。

一方、複数の部分的なオントロジーを統合する際に複数のバージョンのオントロジーが存在しては、概念体系の整合性が破綻する恐れがある。このような問題の解決は今後の課題となっている。

●各オントロジーの更新と変化の影響の伝播

本研究では、“個々のオントロジーはローカルに編集され、共有場に公開された各オントロジーを更新することによって分散構築が進む”という方針をとる。各オントロジーは、編集開始時やオーサの要求した時に、公開された他のオントロジーの最新バージョンにアクセスする。これによって他のオントロジーからの変化が伝わり、オーサは整合性を保持するために必要な編集を加えることが可能になる。

●オントロジーの再利用

本研究におけるオントロジーの再利用とは、ある分散構築過程で部分として存在するオントロジーを、別のオントロジー構築で用いることを指す。

オントロジー構築には開発者間の合意の形成が欠かせないため、分散構築においても、開発者のグループは目標を共有し、相互のコミュニケーションの下で各オントロジーを管理、また構築するべきである。よって、オントロジーを再利用するためのアプローチの一つは、再利用する側の開発者と元来の開発者との合意形成を支援し、全員で協調して分散構築を行うことである。しかし、これはコミュニケーション支援の範疇に入るもので、本研究の直接の対象ではない。別のアプローチは、ある段階のオントロジーを用い、それ以後は、再

利用する側の開発者で独自の編集を加える方法である。本研究では、主としてこちらの形態の再利用支援を考えている。この場合、独自に編集されたオントロジーは、元来のオントロジーの派生としてではなく、別に構築されたオントロジーとして扱われ、管理される。そして、依存関係管理の下、再利用が実現される。

(2) 協調構築とオーサ管理

複数のオーサが並行して同じオントロジーを構築すると、整合性を保持するのが極めて難しくなる。そのため本研究では、各オントロジーを構築するのは基本的には一人のオーサであると想定し、オーサ毎にオントロジーへのアクセスを管理している。しかし、並行作業を避ける形での簡単な協調構築を実現できるように、本システムでは各オントロジーでメインのオーサを決定し、その許可によって他のオーサもそのオントロジーを編集可能とする。

3. 「法造」における分散構築支援システムの試作

本章では、2章で議論した内容を実装したシステムについて述べる。まず、本研究室で開発されているオントロジー構築支援環境「法造」の概要を示し、分散構築がそこでどのように実現されるかの概要を述べる。続いて、それを実現させる2つのツールについて説明する。

3.1 法造

「法造」[古崎 02]は、オントロジー(=“法”)を構築する(=“造”)為の計算機環境で、「オントロジーエディタ」、「概念工房」(オントロジー構築ガイドシステム)、「オントロジーサーバー」、そして「オントロジーマネージャ」の4システムから構成される(図1)。

オントロジーエディタは、基礎理論に基づいたオントロジーをグラフィカルに記述する環境を提供する。意味定義は、定義の継承のみならず、部分概念の担うロール概念や、関係概念についても記述可能であり、これらはシステムが動的に管理する。概念工房[石川 02]は、オントロジー構築方法 AFM (Activity-First Method)に基づき、専門家等による自然言語のドキュメントを基にし、そこから語彙を抽出し、ガイドラインに沿ったオントロジー構築の過程を支援する。オントロジーサーバーは、上記の

2つのシステムで構築されたオントロジーやモデルを管理する。またオントロジーで定義された公理に基づく整合性の検証機能も備えられている。オントロジーマネージャは、前節で述べた考察に基づき設計・拡張されたシステムで、部分として存在する各オントロジーを仮想的に統合し、構築するオントロジーの全体像をオーサに提供する。これは本研究の中心的存在であり、3.3節で詳細を説明する。

3.2 分散構築の実現

(1) 分散構築の流れ

分散構築は各オントロジーのオーサが以下のステップを繰り返す事で進められる:

- ① 法造にログインする。
- ② オントロジーマネージャを起動し、構築するオントロジーの全体像や相互の依存関係を把握する。
- ③ 編集するオントロジーをオントロジーマネージャから選択し、オントロジーエディタで開く。この時、オントロジーマネージャは依存している他のオントロジーにも自動的にアクセスし、依存関係の整合性や更新情報がチェックされる。
- ④ 依存関係のあるオントロジーが更新されていれば(あるいは必要があれば)、整合性保持のための追従変化を行う。
- ⑤ オントロジーを編集する。なお、依存関係のチェックは編集中いつでも可能である。
- ⑥ 編集が終了したら、オントロジーをオントロジーサーバー上に保存し、必要に応じて他のオーサがアクセスできるように公開する。

(2) データ構造とその利用

システムは、各オントロジー内に保持されている以下の情報に基づいて依存関係を管理する。

- オントロジーの名前、バージョン、オーサ、最終更新日時
- 依存している他のオントロジーの名前とバージョン
- 依存している概念定義をコピーしたデータ

他のオントロジーと依存関係を結び、他のオントロジーの概念を利用する際、システムは依存する側のオントロジー内にその概念の定義をコピーしたデータを作成する。このデータは、依存関係を結んだ後に、その概念に変化があったかどうかを検出

するために用いられる。オリジナルの概念とコピーのデータとを比較し、その差分から概念に加えられた変化を検出することができる。変化を検出した後は、表1の変化のパターンと照らしあわせ、必要に応じて整合性保持のための支援を行う。

(3) 分散構築実現のためのオントロジー操作

オントロジーマネージャによって、オーサには以下の4つの操作が提供される: ①部分としての新規オントロジーの作成 ②オントロジーの分割 ③オントロジーの統合 ④オントロジーの再利用。これらの操作は主として、オントロジーの構成とその間の依存関係を先に規定しておき、その後で各オントロジーの詳細

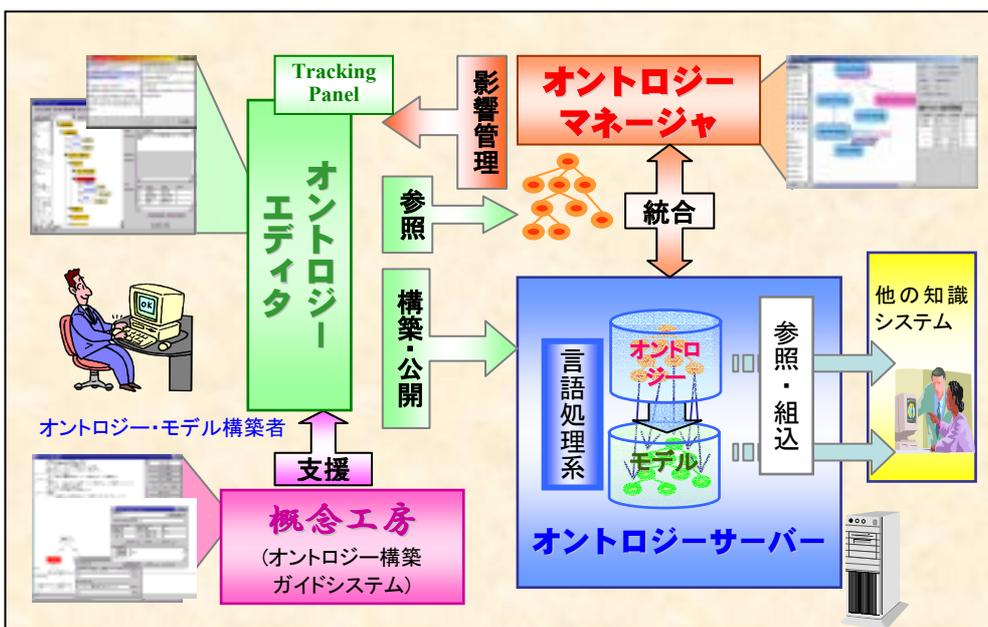


図1. 法造の全体像

を構築する場合に用いられる操作である。

一方で、オントロジーの全体像が定まる前に各オントロジーの構築を開始し、構築途中必要に応じて依存関係を結ぶ状況も考えられる。この場合は、オントロジーエディタによって ⑤他のオントロジーの概念を検索し、依存関係を結ぶ操作が提供される。

3.3 オントロジーマネージャ

オントロジーマネージャ(図2)は、4つのパネルから構成される。

- **Ontology List:** オントロジーサーバーに公開されているオントロジーの一覧を(選択的に)表示する。ユーザーがここから選択したオントロジーに関する必要な情報は、他の3つの画面に表示される。
- **Ontology Viewer:** 構築される各オントロジーと、そのオントロジーが持つ依存関係のうち継承関係がノードとリンクで表示される。
- **Ontology Information Panel:** 選択されたオントロジーの名前、バージョン、オーサ、最終更新日時が表示される。
- **Dependency Panel:** 選択されたオントロジーと依存関係を持つオントロジーの一覧が、その関係の種類毎に表示される。分類は上位・下位・参照・被参照の4種で、これらは Tab で表示を切替える事が可能である。一覧には、オントロジー名、依存関係を担う概念、オントロジーのバージョン、変更の有無が表示される。

これらのパネルによってユーザーに情報を提供するため、オントロジーマネージャは以下の3つの機能を果たす。

(1) 依存関係管理

オントロジーマネージャは、個別に構築される各オントロジーを仮想的に統合し、その全体像を **Ontology Viewer** でオーサに提示する。オーサはこのパネルによって、構築するオントロジーの全体像を容易に把握することが可能である。また、依存関係の詳細は **Dependency Panel** によって知ることができる。これら二つのパネルは、整合性保持のための追従変化を行うための最初のステップでも利用される。これについては3.4節で述べる

(2) バージョン管理

オントロジーマネージャは、オントロジー同士を関連付ける際に、そのバージョン情報も扱う。選択されたオントロジーが古いバージョンのオントロジーに依存している場合、そのことは **Ontology Viewer** 上でグラフィカルに表示される。

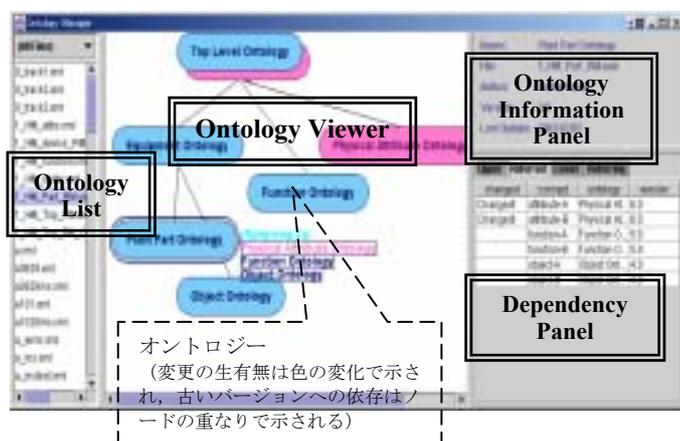


図2. オントロジーマネージャ

公開されたバージョンのオントロジーはオントロジーサーバーに保存されており、オントロジーマネージャはサーバー内を検索して必要なオントロジーにアクセスする。

(3) オーサとアクセス管理

オントロジーサーバーには、分散構築によって開発されるオントロジーについて、それを構成する各オントロジーとそのオーサについての情報が保持されている。システムはこの情報に沿って法造にログインしたオーサを識別し、オーサが分散構築に加わっているオントロジーや、その中でオーサが編集可能な部分を管理する。

3.4 Tracking Panel

Tracking Panel は、依存しているオントロジーに変化があった際、整合性保持のための追従変化で用いられるオントロジーエディタ内のツールである(図1)。

あるオントロジーに変更が加えられたことは、**Ontology Manager** の **Ontology Viewer** や **Dependency Panel** でオーサに示される。その具体的な内容は、変化の影響を受けたオントロジーのオーサが、オントロジーエディタでそのオントロジーを開いたときに **Tracking Panel** によって表示される。**Tracking Panel** には、加えられた変更がリストアップされ、それぞれに対応するための追従変化の方法が提示される。オーサは適切な選択肢を選ぶだけで、システムによって追従変化が半自動的になされ、影響を受けたオントロジーの整合性が保たれる。

4. まとめ

本稿ではオントロジーの分散構築を行うに当たって必要な点を検討し、その実現方法と支援システムについて述べた。試作段階でのシステムの実装は完了しており、基本的な機能は既に動作している。しかしながら、必要な点が全て論じられたとは言えず、今後の課題としてさらなる支援機能の検討や改良を引き続き行っていく。

参考文献

- [Sunagawa 03] E. Sunagawa, K. Kozaki, T. Kitamura, and R. Mizoguchi: Management of Dependency between Two or More Ontologies in an Environment for Distributed Development, Proc. of the International Workshop on Semantic Web Foundations and Application Technologies (SWFAT2003), pp.35-41, Nara, Japan, March 12, 2003
- [砂川 02] 砂川, 古崎, 来村, 溝口: 分散開発環境における複数オントロジー間の依存関係管理, 人工知能学会研究会資料(SIG-KBS-A202), pp.53-58, 2002
- [古崎 02] 古崎, 来村, 池田, 溝口: 「ルール」および「関係」に関する基礎的考察に基づくオントロジー記述環境の開発, 人工知能学会誌, vol.17, No.3, pp. 196-208, 2002
- [石川 02] 石川, 久保, 古崎, 来村, 溝口: タスク・ドメインルールに基づくオントロジー構築ガイドシステムの設計と開発—石油精製プラントを例として—, 人工知能学会論文誌, Vol. 17, No. 5, pp. 585-597, 2002