

# オントロジー構築・利用環境「法造」の開発と利用—実規模プラントのオントロジーを例として—

## Development and Use of an Environment for Building/Using Ontologies “Hozo” —A Case Study on a Real-scale Plant Ontology—

古崎 晃司  
Kouji Kozaki

大阪大学産業科学研究所  
I.S.I.R., Osaka University  
kozaki@ei.sanken.osaka-u.ac.jp, <http://www.ei.sanken.osaka-u.ac.jp/~kozaki/>

来村 徳信  
Yoshinobu Kitamura

(同 上)  
kita@ei.sanken.osaka-u.ac.jp, <http://www.ei.sanken.osaka-u.ac.jp/~kita/>

佐野 年伸  
Toshinobu Sano

(同 上) †1

本松 慎一郎  
Shin'ichiro Motomatsu

(同 上) †2

石川 誠一  
Seiichi Ishikawa

(同 上) †3

溝口 理一郎  
Riichiro Mizoguchi

(同 上)  
miz@ei.sanken.osaka-u.ac.jp, <http://www.ei.sanken.osaka-u.ac.jp/members/miz/overview.html>

**keywords:** ontology, development support system, plant ontology

### Summary

Ontological engineering is a successor of knowledge engineering which has been considered as a technology for building knowledge-intensive systems. Although knowledge engineering has contributed to eliciting expertise, organizing it into a computational structure, and building knowledge bases, AI researchers have noticed the necessity of a more robust and theoretically sound engineering which enables knowledge sharing/reuse and formulation of the problem solving process itself. Knowledge engineering has thus developed into “ontological engineering” where “ontology” is the key concept to investigate. Although the necessity of an ontology and ontological engineering is well-understood, there has been few success stories about ontology construction and its deployment to date. The reason for this is that the principles of ontology design is not clear enough. Therefore, a methodology of ontology design and a computer system supporting ontology design are needed. Our research goals include a methodology of ontology design, and development of an environment for building and using ontologies. This article outlines an environment for building and using ontologies “Hozo” which is under development. Hozo is designed based on a fundamental consideration of an ontological theory. And it has been extensively used in many projects to develop various ontologies. As an example, this paper presents an activity of ontology construction and its deployment in an interface system for an oil-refinery plant operation which has been done under the umbrella of Human-Media Project for five years. And we demonstrate the ability of Hozo through the ontology building/using process.

### 1. ま え が き

オントロジー工学は、知識システムの構築技術として考えられていた知識工学を継承するものである。知識工学は専門知識を抽出・構造化し、知識ベースを構築することに貢献してきた。しかし近年は、知識の共有・再利用や問題解決プロセス自身を明確に記述することを可能

とする、より強固で健全な理論と技術、すなわち、オントロジー工学が注目されている。換言すれば、知識工学はオントロジーをキーコンセプトとしたオントロジー工学に発展したと言える。しかし、オントロジーやオントロジー工学の必要性は広く理解されつつあるにも関わらず、実規模の対象におけるオントロジーの開発や利用に関しては、今日に至るまで報告例が少ない。その一因として、オントロジーの基礎理論やオントロジー構築の方法論が未だ十分には確立していないことが挙げられ、構築・利用を支援するシステムの開発や構築方法論の整備

†1 現在、オンキヨー株式会社

†2 現在 (株) アイティブースト

†3 現在、NTTコムウェア

などが渴望されている．このような背景のもと筆者らは，オントロジーの構築・利用を支援する計算機環境として，オントロジー構築・利用環境「法造」の開発を進めると共に，様々なプロジェクトにおいてオントロジーやオントロジーベースのアプリケーションの開発を行ってきた [古崎 99a, 古崎 99b, 古崎 02]．「法造」はオントロジーの基礎理論に関する考察に基づいて設計された記述環境や，構築したオントロジーやモデルを他のシステムで利用するための機構を持ち，これまでも様々なオントロジーの構築に利用されている．本論文は「法造」の概要を示すと共に，オントロジーの構築から利用に至る一連の過程における「法造」の使用例を通して，本システムの有用性を示すことを主題としている．ここでは，旧通産省の支援のもとで，ヒューマンメディアプロジェクトの一環として 5 年間に渡り開発・利用がなされた石油精製プラントオントロジーの開発例を中心に述べる．

以下，2 章ではオントロジーベースのアプリケーション開発について述べ，3 章ではオントロジー構築・利用環境「法造」の概要を述べる．続く 4 章では，石油精製プラントオントロジーの開発・利用の過程を通して「法造」の有用性を示す．5 章では本研究の成果を関連研究との比較と併せて述べ，6 章で全体の総括と今後の検討課題を述べる．

## 2. オントロジーベースのシステム開発

### 2.1 オントロジー

知識システムは対象としている問題の範囲や前提条件など，本質的になんらかの仮定に基づいており，扱える範囲に特定の限界がある．また，問題解決に用いられる知識も同様になんらかの仮定に基づいている．しかしながら，従来の問題解決システムではこのような仮定や規約が明示化されていることはまれである．その為に，知識の共有・再利用が困難となり，扱う知識が大規模になるにつれて知識ベースの構築やメンテナンスに必要となるコストが大きくなる．また，計算機とシステムを扱う人間の間に概念レベルのギャップが生じたり，分散協調システム内での語彙の不統一などの問題が生じる [溝口 98]．

このような知識の再利用という大きな問題を解決するためには，知識の内容について議論を行い，知識の記述意図を明確にする技術が求められ [溝口 96, 溝口 97]，その鍵としてオントロジーが注目されている．知識システムの開発にオントロジーを用いることで，得られるメリットの代表的なものを以下に示す [溝口 99a]．

- (1) 暗黙情報の明示化：前述のように，問題解決システムは様々な仮定や規約に基づいているが，それらの情報は多くの場合暗黙的である．オントロジーはこのような暗黙知識を記述したものであり，それらを明示化する役割を果たす．

- (2) 共通語彙の提供：対象とする世界を記述する際に必要とされる，厳密に定義された，関係者の合意に基づく「語彙」を提供する．
- (3) 知識の体系化：知識を体系化するには，厳密に定義された合意に基づく概念や語彙を用いて様々な現象，観測事象，興味ある対象を説明する理論が記述され，知識の組織化がなされる．オントロジーはこのような知識を体系化する際の拠り所となるバックボーンとしての機能を持つ．
- (4) 標準化：オントロジーは標準概念の意味を規定するものであり標準化に貢献する．
- (5) メタモデル的機能：オントロジーはある対象をモデル化するときに必要となる概念とそれらの間に成立する関係を明示的に規定し，そのモデルはオントロジーが提供する概念と制約の下で作られる．この意味で，オントロジーはメタモデルとしての機能を持っているということができる．

### 2.2 オントロジーを利用したシステムの分類

前節で述べたオントロジーの特徴を利用して，様々なシステムが開発されている．Uschold らはそれらのオントロジーベースのアプリケーションを ontology application scenarios として以下のように 3 種類に分類し，各システムにおけるオントロジーの利用され方やシステムの枠組みを解説している．[Uschold 99]

- (1) neutral authoring：オントロジーを，アプリケーションが利用する知識を作成する際の中間生成物として用いる．利用の枠組みとしては，オントロジーやモデルを構築し，複数のアプリケーションに利用できる形式に変換する機能を必要とする．
- (2) common access to information：オントロジーを複数のアプリケーションや開発者から共通にアクセスできる，共有された知識として利用する．利用の枠組みとしては主に以下のようなものがある．
  - a ビューアーを用いてオントロジーを参照する
  - b オントロジーやモデルの変換器を用いる
  - c 共通アクセスの為に API を用いる
  - d 複数のオントロジー間のマッピングを利用する
- (3) indexing for search：オントロジーを検索エンジンに用いる知識の索引付けに利用する．

これらのオントロジーベースのシステムを開発するためには，(1) のようなオントロジーやモデルを構築する機能や，構築されたオントロジーを管理し (2) で述べられている利用の枠組みを提供する環境が必要とされる．このような背景のもと筆者らは，これら要求を満たすオントロジー構築・利用環境の開発を進めてきた [古崎 99a, 古崎 99b, 古崎 02]．次章では，筆者らが設計・開発したオントロジー構築・利用環境「法造」について述べる．

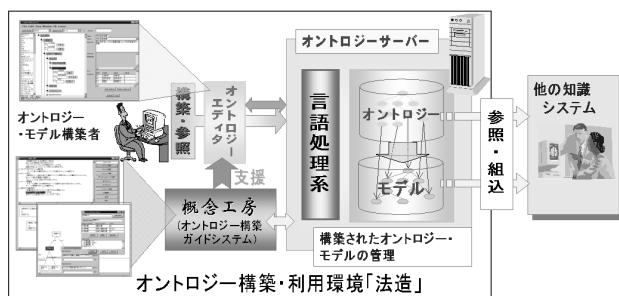


図 1 法造の全体像

### 3. オントロジー構築・利用環境「法造」

#### 3.1 「法造」の概要

オントロジーベースのシステム開発は、

- (1) オントロジーの構築
  - (2) 構築したオントロジーに基づいたモデル構築
  - (3) オントロジー・モデルを利用したシステムの開発
- という3つの過程を経てなされる。

このようなオントロジーの構築から利用に至る一連の過程を支援することを目的として、筆者等はオントロジー構築・利用環境「法造」の開発を進めてきた[古崎 99a, 古崎 99b]。「法造」とはオントロジー(=“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」の3つのシステムから構成される(図1)。

本章では「法造」が扱うオントロジーの概要を示した後、オントロジー構築・利用の各過程において「法造」が提供する機能を述べる。

#### 3.2 「法造」が扱うオントロジー

「法造」が扱うオントロジーは、以下の構成要素からなる。

- label: 概念名を表すラベル
- super: 上位概念
- axiom: 公理
- def: 自然言語による定義(コメント)
- slots: スロットの集合
  - スロット名
  - スロットの種類(part-of/attribute-of)
  - クラス制約
  - スロットの数に関する制約

part-of はその概念を構成している部分にあたる概念(部分概念)を表し、attribute-of は概念がもつ属性を表す。また、クラス制約はスロット(部分概念、属性)の値となりうるインスタンスが所属するクラスに関する制約、数に関する制約はスロットの値となるインスタンスの数に関する制約を示す。

(Participants

(?接続情報 "接続情報"  
(?ポート ("ポート" ?接続情報))  
(?接続先部品 ("接続先部品" ?接続情報))  
(?接続先ポート ("接続先ポート" ?接続情報))))

(AxiomBody

(Condition  
(equal ("Side" ?ポート) "IN"))  
(Body  
(equal ("Side" ?接続先ポート) "OUT"))

(UserUse

(部品の入力ポートは、接続している他の部品の出力ポートにつながっていないとありません!)))

図 2 公理の記述例

公理は概念が満たすべき性質を宣言的に表したもので、部分概念や属性に関する制約や関係などが記述される。オントロジーサーバーで扱う公理は、制約の定義に参加する概念のインスタンスを表す変数を記述する Participants と、制約の内容が記述される Axiom Body の2つの部分からなる。Axiom Body はさらに Condition, Body, UserUse の3つの組で記述される。Condition は公理に記述された制約が適用されるための条件で、この適用条件が成立するときに対応する Body に記述された制約内容が評価される。UserUse には、オントロジーサーバーを利用するエージェントや、ユーザー向けのメッセージが記述され、Body に書かれた制約に違反があれば、エラーメッセージと共に UserUse の内容が返される。

公理を記述するための文法については、等価関係(equal)、所属関係(include)、順序関係(order-of)などのプリミティブと、AND, OR, NOT といった論理演算子など、必要最低限なものを用意している。例えば“ある部品の入力ポートは接続している他の部品の出力ポートにつながっていないとなければならない”という公理は図2のように表現される。

#### 3.3 オントロジーの構築

オントロジー構築は、

- (1) オントロジーの設計(内容の検討)
- (2) オントロジーの記述
- (3) 整合性確認

を繰り返すことで行われる。

##### §1 オントロジーの設計

オントロジーの設計は「対象世界の理解」、「対象世界を構成する概念の分析・組織化」の2つの過程に分かれ、自己の内省や各種文献の参照、専門家との議論などを通して行われる。「対象世界の理解」では、構築するオントロジーの対象や利用目的に応じて必要な知識を収集し、オントロジーで記述すべき概念を明らかにする。そして「対象世界を構成する概念の分析・組織化」では、抽出さ

れた概念を分析し体系的に組織化する。この際、オントロジーの構成要素や概念的性質などを明らかにする基礎理論に関する考察 [溝口 99b, 古崎 02] や、構築方法論の開発が不可欠となる。

「法造」におけるオントロジーの設計は、概念工房によって支援される。概念工房はオントロジー構築方法 AFM に基づく、オントロジー構築ガイドシステムで、複数のステップに分かれた一連のガイドラインに沿ってオントロジーの設計を支援し、オントロジーの全体像を構築する機能を持つ。この詳細は別稿で報告する [石川 01]。

## §2 オントロジーの記述

オントロジーの設計がなされると、次いで、3.2 節で示したオントロジーの構成要素を用いて、設計したオントロジーの定義内容を記述する。オントロジーの記述には、それらの構成要素（記述内容）の表示・編集を支援する諸機能が必要となる。その際、より良質なオントロジーを構築する為には、オントロジーの基礎理論に基づいた扱いをサポートすることが重要となる。

「法造」におけるオントロジーの記述はオントロジーエディタを用いて行う。オントロジーエディタはユーザーの編集操作に応じて概念の「定義」、「編集」、「整合性の検証」等のオントロジーサーバーが提供する操作関数を呼び出し、オントロジーの構築を進める。操作関数としては、概念定義 (define-base-concept)、上位概念の取得 (get-superclass)、スロットの追加 (add-slot) など、オントロジーやモデルの構築に必要な 26 種類を用意している。オントロジーエディタが提供するオントロジー記述環境は、「ルール概念」と「関係」に関する考察を中心としたオントロジーの基礎理論に基づき開発されている。この基礎理論に基づいたオントロジーの扱いが「法造」の最も本質的な特徴であるが、この詳細は別稿で報告している [古崎 02]。

## §3 オントロジーの検証

記述したオントロジーの定義内容が適当か、矛盾がないかなどの検証を行う必要がある。オントロジーの定義内容そのものが適当であるかについては、オントロジー研究の本質的課題であり妥当性の確認は容易ではなく、使用を通して確認する以外に有効な方法はない。しかし、記述方法の文法エラーなど形式的な整合性の確認は、適切な言語処理系を用いて行うことができる。

「法造」ではオントロジーサーバーが整合性検証を行うための処理系を提供する。ユーザーはオントロジー構築中において任意の時に「オントロジーの整合性検証」機能（コマンド）を実行し、構築したオントロジーの形式的な整合性の確認を行う。この整合性検証機能は、本研究の中心課題ではないので現在は、

- 概念定義で上位概念やクラス制約として参照している概念が既にオントロジーに定義されているか
- is-a 階層がループしていないか

など必要最低限のものを実装しており、今後、必要に

応じて処理系の機能拡張や、既存の論理システムを利用することを検討している。

## 3.4 モデルの構築

オントロジーに基づくモデルの構築は、オントロジーで定義した概念（クラス）から個々のインスタンスを作成し、接続情報や属性値など、そのインスタンスに特有の情報を与えることで行われる。各概念や概念間の関係を表す語彙とその定義はオントロジーで用意されているので、オントロジーに基づいてモデルを作成・変更することによりモデリングの作業が容易になり、構築したモデルの一貫性が保証される。このようなオントロジーに基づくモデル構築は、

- (1) オントロジーで定義した概念や関係に基づき、個々のインスタンスを作成する（モデルの記述）
- (2) 構築したモデルがオントロジーの定義に矛盾していないかを検証する（モデルの整合性確認）

という 2 つの過程を繰り返して行われ、これらの各過程を支援する機能が必要とされる。

「法造」でモデルを構築する際には、モデルの表示・編集を行う「モデルエディタ」と、オントロジーの定義を参照する「オントロジー参照ウィンドウ」の 2 つが表示される。ユーザーはオントロジーで定義されている概念の一覧から必要な概念を選択し、そのインスタンスを作成することでオントロジーに基づいたモデルの記述を行う。その際「法造」は、オントロジーに基づき構築されたモデルの整合性を保つために、オントロジーで定義された公理やスロットに関する制約に従い次のような機能を提供する。

- 必須スロットの自動生成：概念定義で必須とされたスロットは自動的に生成し、デフォルトの値を入れる。
- スロット作成の制限：オントロジーで定義されたスロットの数や制約に従って、インスタンスが持ちうるスロットの種類や、スロットの値として入るインスタンスを制限する。
- スロットに入るインスタンスの検索：クラス制約に基づいて値となりうるインスタンスの一覧を表示する。

このようにして構築したモデルは、オントロジーサーバーの処理系を用いて整合性の検証を行う。ユーザーが（任意の時に）「モデルの整合性検証」機能を実行すると、オントロジーサーバーの処理系が編集集中のモデルに含まれる各インスタンスに関して、

- オントロジーで定義されたスロットに関する制約（クラス制約、数の制約）
- オントロジーで公理として記述された制約

が満たされているかを確認し、違反があるとユーザーにエラーメッセージを表示する。これらの機能を用いて、ユーザーはオントロジーに基づいたモデルを構築することができる。

### 3.5 オントロジーとモデルの利用

「法造」で構築されたオントロジーやモデルはオントロジーサーバーが管理し、必要に応じて外部の知識システムによって利用される。本論文では、構築されたオントロジーやモデルを、オントロジーサーバーを介して利用する外部システムを「OS 利用エージェント」と呼ぶ。OS 利用エージェントは、2.2 節で述べたように分類できる。「法造」のオントロジーエディタは、Java 言語を用いた OS 利用エージェントとして開発がなされており、この分類では neutral authoring に対応している。

オントロジーやモデルを利用する主な枠組みは、2.2 節の (2) common access to information の a~d で示されている。これを参考にして「法造」では、OS 利用エージェントがオントロジーやモデルを利用するために、オントロジーサーバーは以下の 3 通りの枠組みを用意した。

- (1) ネットワーク経由の参照：構築されたオントロジーやモデルは、ネットワークを介して外部から参照することができる。オントロジーエディタは JAVA アプリレットとして実装されており、ユーザーはインターネットを介して、構築したオントロジーやモデルにアクセスすることができる。
- (2) 特定形式への変換：「法造」は構築したオントロジーやモデルを、汎用の形式に変換して出力する機能を持つ。現在は階層化したテキスト形式、Lisp 形式、XML/DTD 形式をサポートしている。OS 利用エージェントはそれぞれに適した形式で取り込んで利用する。
- (3) 操作関数 (API) を利用したアクセス：オントロジーやモデルの構築に用いられる操作関数は、外部のシステムから利用可能な公開関数群として整理されている。OS 利用エージェントはこれらの操作関数を用いてオントロジーサーバーが提供する機能を利用することが出来る。

上記 (1), (2), (3) の枠組みはそれぞれ、2.2 節の common access to information で述べた a, b, c に対応する。d のオントロジー間のマッピングに関しては、今後の課題として検討を進める。

### 3.6 その他の諸機能

#### §1 標準化

構築したオントロジー・モデルをより多くのシステムで利用とするためには、標準への準拠について考察する必要がある。「法造」では標準化への取り組みとして、知識ベースシステムの標準仕様として広く使用されている OKBC への対応と、インターネットの標準として近年注目されている XML の利用を行っている。

OKBC は DARPA の支援のもとスタンフォード大学知識システム研究所で開発された API で、知識表現システムの知識ベースにアクセスするためプロトコルを提供

表 1 オントロジーサーバーの操作関数と OKBC Operations の対応 (一部)

| 操作関数               | OKBC Operations            |
|--------------------|----------------------------|
| define-base-class  | create-class               |
| add-slot           | create-slot    attach-slot |
| add-superclass     | add-class-superclass       |
| make-base-instance | create-frame               |

する [Farquhar96]。OKBC にはフレーム型知識表現である OKBC 知識モデルと、知識表現システムに対する標準的なインターフェースを提供する OKBC Operations が含まれる。「法造」のオントロジーサーバーの操作関数を OKBC Operations に対応させることで、OKBC ベースの様々なシステムから「法造」で構築されたオントロジーやモデルを容易に利用することが出来るようになる (表 1)。オントロジーやモデルの操作に利用する関数は全て OKBC Operations に対応可能で、整合性検証を行う関数のみ OKBC を一部拡張することで対応させる。

XML はインターネットの標準として W3C より勧告されたメタ言語で、電子的なデータ交換を行う役割も持つことから、近年広く利用されつつある。また XML 対応のツール類は既に多数開発されており、「法造」では構築されたオントロジーやモデルを XML 形式で出力することで、XML 対応のアプリケーションからの利用を可能とする。その際、オントロジーは DTD に変換して出力され、DTD が提供する文法を用いて XML によるモデルインスタンスの記述を規定することができる。しかし DTD の表現能力の限界から、オントロジーを DTD 出力する際には定義内容の一部は単なるコメントとして扱う。今後は W3C で現在制定が進められている XML Schema, RDF(S) や DAML など、より適切な言語の採用を検討している。

#### §2 ユーザー管理

オントロジーの構築・利用を支援する補助的な機能として、オントロジーサーバーはユーザー管理機構を持つ。ユーザーはグループ単位で管理され、ユーザー名とパスワードを用いて認証される。構築されたオントロジーやモデルは、パッケージという集合単位で管理され、ユーザー単位 / グループ単位で排他制御やアクセス管理などを行う。

### 3.7 開発状況

オントロジーサーバーの主要モジュールは CLOS を、一部の通信モジュールは Java を用いて開発されており、Java で開発されたオントロジーエディタと共に公開の準備が整っている。また「法造」は研究室内外の様々なプロジェクトでオントロジーやモデルの構築に利用されており、オントロジーサーバーを利用したシステムも既に開発されている。次章では「法造」を用いて開発されたオントロジーおよび、システムの開発例を述べる。

#### 4. プラントオントロジーの構築と利用

「法造」のような構築環境の評価は、既存システムに対する新規性とシステムの使用経験からなされる。本システムの新規性に関しては別稿で報告 [古崎 02, 石川 0] するが、後者の評価には長期間に渡る使用経験が必要となることが多く、客観的な評価は一般的に難しい。しかし、実際のオントロジー構築から利用に至る一連の過程でシステムを使用することで、本システムが実際の使用に必要な機能を提供していることを確認できると考えられる。そこで本章では、旧通産省の支援のもと行われたヒューマンメディアプロジェクトの一環として開発された石油精製プラントのオントロジーを例として、「法造」を用いたオントロジーの構築から利用に至るまでの一連の開発過程について述べる。さらに、その他の研究における「法造」の利用状況を述べ、これらの利用例を通して「法造」が実際のオントロジー構築および利用に耐えうる有用なシステムであることを示す。なお本論文で述べたオントロジーの開発事例では、「法造」のオントロジーエディタおよびオントロジーサーバーのオントロジーやモデルを記述する能力および、それらを利用したシステム開発が可能であることの確認に重点を置く。概念工房を用いたオントロジー構築支援に関しては、紙面の都合のため別稿 [石川 01] で報告する。

##### 4.1 ヒューマンメディアプロジェクト

工業プラントの巨大化・複雑化が進むにつれ、扱うべきプラント情報が多様化している。また生産工程の自動化に伴い、運転員がノウハウを獲得する機会が減少している。そのため産業界ではプラント運転における情報化の促進・コストの削減・安全性のさらなる向上が求められている。このような要求を満たすために、快適に・安全にプラントを運転することを支援する次世代プラント用ヒューマンインタフェースの開発が、旧通産省の支援のもと、ヒューマンメディアプロジェクトの一環として行われた [佐野 99]。

これまでのプラント運転支援システムはすべて一品生産であり、開発にかかるコストが高かった。またシステムの大規模化に伴い、知識のメンテナンスや機能追加が困難となっている。その一因は知識ベースを構築する際の仮定や前提条件といった背景情報が、暗黙的で明確化されていないことにある。このような問題を解決する有効な手段の 1 つが、オントロジーの利用である。2.1 節で述べたように、オントロジーは知識システムの開発・運用の過程において様々な貢献をする。次節では、次世代プラント用ヒューマンインタフェースにおける、プラントオントロジーの利用のされ方について述べる。

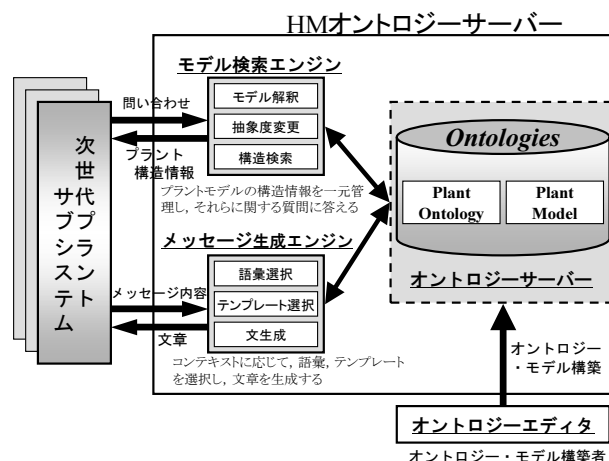


図 3 HM オントロジーサーバーの構成

##### 4.2 ヒューマンインタフェースのシステム構成

次世代プラント用ヒューマンインタフェースは、プラントの状態を直感に訴える表示を行う「仮想プラント表示エージェント (VR)」や、運転員がプラントを操作するためのインタフェースを提供する「インタフェースエージェント (IA)」など、複数のエージェントから構成される。各エージェント間はネットワークシステムを介して接続され、互いに協調することによりプラントの運転を支援する。これらのシステムが対象とする石油精製プラントのオントロジーおよびモデルは、「HM オントロジーサーバー (HM/OS)」において一元管理される。

HM/OS は 3 章で述べた「法造」のオントロジーサーバーに、OS 利用エージェントとして「メッセージ生成エンジン」、「モデル検索エンジン」を付加した集合システムとして構築されており (図 3)、インタフェースシステムにおいて以下のような機能を持つ。

- (1) メッセージ内容の表現に用いるプラントオントロジーを保管する
- (2) エージェント間で共有されるプラントモデルを構築する
- (3) コンテキストに適した説明文を生成し運転員に提供する
- (4) プラントの構造に関する質問に解答する

各エージェントは分散協調システムが提供する API を利用して、HM/OS から必要な情報を得ることができる。その際、プラントオントロジーは対象とする石油精製プラントに関する知識を明確にし、分散協調システムにおいてプラント運転員や各エージェント間での知識共有や語彙統一の役割を果たす。例えば、プラントモデルの 3 次元仮想表示を行う VR は、各部品の 3 次元空間における位置情報のみを保持している。その為、「このバルブの隣の部品は何か?」といった、プラントの論理構造に関する質問については、HM/OS のモデル検索エンジンに問い合わせる必要のある情報を取得する。HM/OS のメッセージ生成エンジンはエージェントの要求に対して、プラント

オントロジーに基づき状況に応じた語彙を選択して適切なメッセージを返す。これは各エージェントが HM/OS の管理する運転員が日常使用している語彙を用いて会話することを可能とし、計算機と人間の間の概念レベルのギャップを埋める役割を果たす。この詳細は 4.5 節で述べる。

このように 2.1 節で述べた (1) 暗黙情報の明示化、(2) 共通語彙の提供にプラントオントロジーが貢献している。また、これらのシステムにおけるオントロジーの利用され方は、2.2 節の (2) common access to information で述べた枠組みに相当する。

### 4.3 プラントオントロジーの構築

前述の背景のもとで、我々は石油精製プラントオントロジーとして、操作タスクオントロジー、プラント部品、プラント対象物に関するオントロジーを構築した。ここで注意すべき点は、概念を捉える際に本質的な問題である、視点などのコンテキストに依存して定義される「ロール概念」と、他の概念に依存せずに定義される「基本概念」の区別を明らかにすることである [溝口 99b, 古崎 02]。「法造」のオントロジーエディタは 3.3 節で述べたように、これらの概念を峻別して記述する枠組みを提供している。またグラフィカルな記述環境により、ユーザーがオントロジーを記述する際の労力を軽減することができた。以下に、我々が構築したプラントオントロジーの概要を示す。

#### §1 タスクオントロジー

タスクオントロジーはタスクの実行者（ここではプラントの運転員）の行為や対象物がそのタスクにおいて果たす役割（ロール）を表す概念から構成される。よってタスクオントロジーを先に構築することで、対象物に現れる多くの概念の中から、そのタスクを果たすのに必要な概念の範囲を絞ることが可能となる。このような考え方を基にしたオントロジー構築方法論が AFM であり、AFM に基づきオントロジーの構築を支援するシステムが「法造」のサブシステムとして開発を進めている「概念工房」である。本論文で報告するプラントオントロジーは、技術文書を参考に人手によって構築されたもので、概念工房はこの際に得られた知見に基づいて設計・開発されている [石川 01]。

今回対象としている石油精製プラントの運転は、プラントの振る舞いを監視し、異常があればその原因を診断し、異常を復旧できるように装置を操作することで行われる。この問題解決において処理を表す「運転する」「認識する」「診断する」「決定する」「列挙する」「絞り込む」といった概念を運転タスクオントロジーとして記述した。

これらのタスク概念は、全体-部分 (part-of) 関係と一般-特殊 (is-a) 関係で組織化されている。例えば「運転する」とは異常を認識し、近未来の異常を予測し、異常の原因を診断し、対策を決定することで、part-of 関係によ

るサブタスクとして「認識する」「予測する」「診断する」「決定する」を持つ。一方、「推論する」はある状態から因果関係にある新たな状態を導出することで、is-a 関係によって特殊化された「予測する」「(原因候補を) 推論する」「(原因関連状態を) 推論する」といった下位概念を持つ。これらの下位概念は「推論する」の持つ性質をすべて継承し、もとの状態と導出する状態のロールを特定したものである。「予測する」は推論により異常状態から近未来状態を、「(原因候補を) 推論する」は事例ベース推論により異常状態から原因候補を、「(原因関連状態を) 推論する」は推論により原因候補から原因関連状態を導出する処理である。

続いてタスクにおけるロール概念として、それぞれの動詞が対象とする状態や操作が問題解決コンテキストで果たす役割を表す「運転状態」「異常状態」「原因候補」「対策操作」等を運転タスクオントロジーとして記述した。「運転状態」が異常と認識されると、さらに「異常状態」という役割（ロール）を果たす。また「近未来状態」は、異常状態から予測された近未来の状態のことである。「原因」には、認識された異常状態の原因となりうる「原因候補」と、診断の結果異常状態の原因であると判断された状態である「実際の原因」がある。「原因関連状態」は、原因候補と原因-結果関係にある測定可能な状態のことであり、原因候補が直接測定できない場合は原因関連状態が実際に起こっているか判定することにより、原因候補が実際の原因であるかどうか判断する。

#### §2 ドメインオントロジー

本研究で対象とする石油精製プラントのドメインオントロジーは、対象物、属性概念、機能概念、部品に分類される。ドメイン概念にもタスクオントロジーと同様にロール概念が存在し、特に対象物を表すドメイン概念の多くはロール概念である。よって対象物のオントロジーを構築する際には、視点やコンテキストに依存しない基本概念とロール概念を分けることから始めなければならない。

対象物オントロジーを図 4 に示す。対象物はプラント中を流れる原油などの流体で、「灯油」「ガソリン」等のように視点と独立して物質の組成などから一意に決まる「視点独立対象物概念」（基本概念）と、視点によって概念のとらえられ方が変わる「視点依存対象物概念」（ロール概念）とがある。

「視点依存対象物概念」は「場所依存概念」「履歴依存概念」「状態依存概念」「役割依存概念」等、文字通り視点によって分類されている。「場所依存概念」には「塔頂製品」のように対象物の位置する場所に注目した概念が、「履歴依存概念」には「留分（蒸留成分）」のように以前なされた機能に注目した概念が用意されている。「状態依存概念」には「気体」「過熱蒸気」の様に対象物のある固有な状態を示す「固有状態依存概念」と「高温」「低沸」のように他の対象物と比較して相対的に表現するときに使用する「相対状態依存概念」がある。「役割依

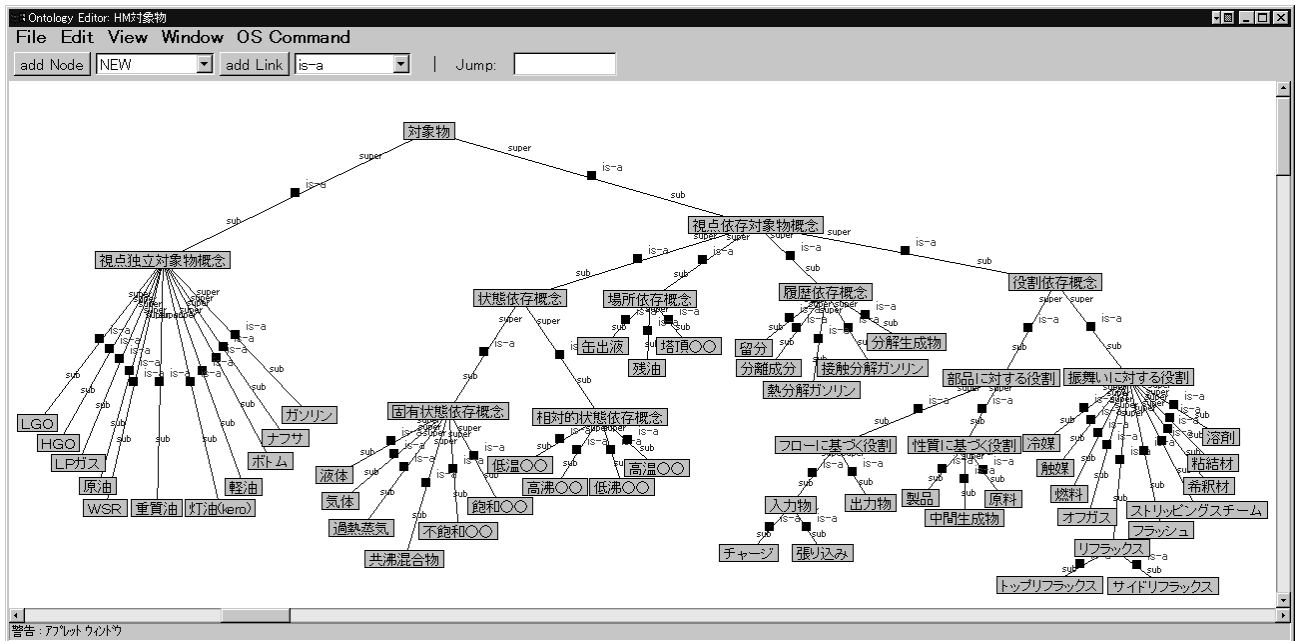


図 4 プラント対象物オントロジー

存概念」には、部品に対する役割に注目した「部品に対する役割」と、部品の振る舞いにおける対象物の役割に注目した「振る舞いに対する役割」がある。

一方、部品は装置と導管に分かれる。部品のうち対象物の位置だけを変えるパイプが導管であり、その他の熱交換部品や蒸留塔などが装置である。装置と導管はその入出力口としてポートを持ち、それらの構造情報は隣り合うポートとポートの接続関係により表現される。また「コントローラー」はパイプを流れる対象物や装置の「温度」「流量」「液レベル」といった様々な属性を測定し、その属性値を設定値に保つようにアクチュエータに対して開度を指示する装置で、他の装置や対象物と様々な関係を持つ。コントローラーが測定する対象を「測定対象物」と呼び、コントローラーが開度を指定する対象を「操作対象部品」と呼ぶ。「操作対象部品」となる装置はアクチュエータのインスタンスに限るのでクラス制約としてそのことが記述されており、モデルを作成する際に、モデル構築者がコントローラーの操作対象部品としてアクチュエータ以外のインスタンスを指定した場合、クラス制約に反していることを指摘することが出来る。また、論理的にコントローラーが動作を制御している装置を「制御対象部品」と呼ぶ。

「属性概念」は、その属性の持ち主が対象物か部品か、また時間を含む属性なのかなどによってカテゴリ分けでき、対象物や部品の持つ属性の属性名ライブラリとなる。「機能概念」は対象物やエネルギーの流れ、対象物属性の変化などに注目することによりカテゴリ分けできる。

これらの石油プラントモデルを記述するのに必要な概念を、約 400 の概念からなる石油精製プラントオントロジーとして定義した。プラントモデルはこのような部品オントロジーを枠組みとして記述され、プラント運転

状況説明文生成時に利用される。

#### 4.4 プラントモデルの構築

続いてプラントオントロジーに基づいて、対象とする石油精製プラントのモデルを構築した。このモデル構築の過程は 3.4 節で述べた「法造」の機能により「オントロジーの概念定義を基にインスタンスを作成し、半自動で生成されるスロットに適切な値を記述する」、「これらの部品間に“接続関係”を定義することで接続情報を記述する」という簡単な作業を繰り返すことで行うことができた。さらに構築したモデルに対して整合性検証機能を用いることで、ユーザーの記述ミスを防ぐことができた。また、モデル構築に必要な概念や関係の定義やインスタンスが満たすべき制約などは、オントロジーで明示され「法造」が提供する機能によりモデルに反映される。この為、ユーザーは「法造」を用いることで、オントロジーに基づく一貫性の取れたモデル構築を比較的容易に行うことができた。ここでは 2.1 節で述べた (3) 知識の体系化、(4) 標準化、(5) メタモデル的機能、といったオントロジーの性質が利用されている。

またモデルを構築する過程において、オントロジーの概念定義に必要な属性が欠けているといった定義の不備が発見される場合がいくつか見られた。このような際には、モデル構築を一度中断し、オントロジー構築の過程に戻って定義を修正することで対応した。このような変更はモデルに動的に反映され、整合性検証機能を実行することでモデルに不整合がないかを確認することができる。しかし現状では、不整合を自動的に修正するなどの機能は実装されておらず、今後の重要な課題の 1 つとして位置付けている。

本研究で構築したプラントモデルでは、装置以外のパ



イブなどの導管も全てモデル化されている。その際、部品の3次元空間内での位置など物理的な情報はモデルに含まれていないが、パイプの分岐点を境界とした全セグメントの接続情報が記述されている。またセンサなどの装置に関しては、物理的な接続情報だけでなく、測定している対象機器との論理的な接続情報を記述した。現在、日石エンジニアリング株式会社の提供資料をもとに、総部品数約2000の石油精製プラントモデルが構築されている(図5)。これは実際に運用されている石油精製プラントの一部をモデル化したものであるが、プラントの本質的な動作に必要な部品は全て表現されており、各エージェントにほぼ現実規模に近いプラントモデルを提供している。

#### 4.5 プラントオントロジーの利用

構築したプラントオントロジーとモデルは、HM/OSによってインタフェースシステムの各エージェントで共有される。ここではHM/OSの一部として開発した「メッセージ生成エンジン」を例として、プラントオントロジーがどのように利用されているかの詳細を示す。

メッセージ生成エンジンは、プラントを監視しているエージェントから送られてくる情報をもとに運転状況を把握し、その状況に応じて運転員が直感的に理解しやすい説明文を生成する機能を持つ。以下、状況に応じた語彙の選択が必要とされる背景について述べた後、メッセージ生成エンジンの詳細を示す。

##### §1 状況に適した語彙の選択

我々はプラントの専門家へのインタビューを通して、プラントの運転員がある同じ物について運転状況によって異なる呼び名で認識しているという事実気付いた。したがって、運転員にとって理解しやすい説明文を生成するためには物の呼び名をコンテキストに適した語彙で表現する必要がある。

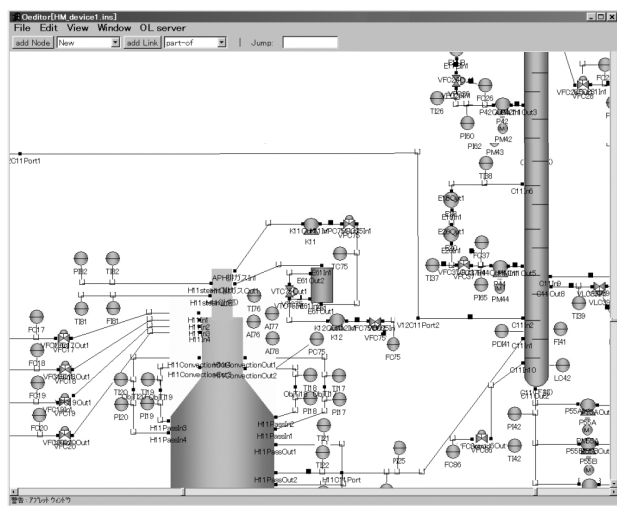


図5 プラントモデル(一部)

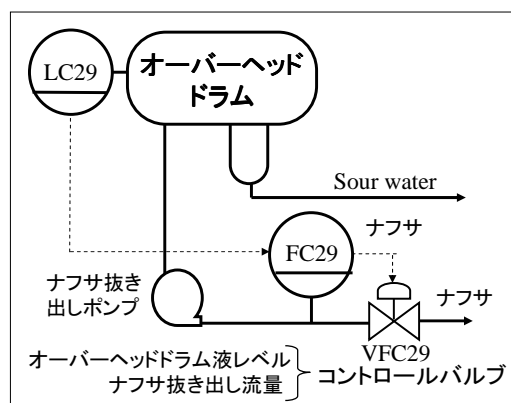


図6 2つのコントローラによるカスケード制御

例えば図6に示すカスケード制御において、流量コントローラ(FC29)と液レベルコントローラ(LC29)という2つのコントローラが、同じコントロールバルブ(VFC29)を利用している。LC29はオーバーヘッドドラムの液レベルを測定・制御し、FC29はナフサ抜き出しポンプから流出するナフサの流量を測定・制御している。この時、これらのコントローラに操作されるコントロールバルブ(VFC29)は、どちらのコントローラに注目しているかというコンテキストに応じて、「オーバーヘッドドラム液レベルコントロールバルブ」と呼ばれたり、「ナフサ抜き出し流量コントロールバルブ」と呼ばれたりする。このようにコンテキストに依存して現れる名前はロール名と呼ばれる。

ここで問題となるのは、注目されているコントローラが通常は暗黙的であるため、ある時点でどの部品が注目されており、プラント運転の過程によって注目される部品がどのように変化するかを推論することである。さらに呼び名が生成されるメカニズムを理解し、システムの柔軟性を高めることが望まれる。それぞれの概念のラベルについて、モデルの構築者は対象とするプラントのデフォルト名のみ記述し、ロール名に関してはコンテキストに応じてシステムが生成することで、モデルの構築を容易にした。特に4.3節で述べたように、プラントの対象物は装置からの視点に応じて、様々なロール名をもつ。例えば「入力物」は多くの対象物が持ちうるロール名の典型例である。

##### §2 メッセージ生成エンジン

HM/OSの「メッセージ生成エンジン」では、このようなロール名を適切に選択するために「フォーカス同定ルール(Focus tracing rules)」と「呼び名生成ルール(Role name generation rules)」という2つのルールを用意した[佐野99, Mizoguchi00]。フォーカス同定ルールは、部品の呼び名を決定する際のコンテキスト情報である「現在注目している部品(フォーカス)」を運転員との対話履歴から決定・保持する為のルールで、呼び名の対象となる部品の種類に応じて“対象物の属性”、“部品

の属性 “,” “コントロールバルブ”, “それ以外の部品” に分けて整理されている。呼び名生成ルールは, コンテキスト情報を元に部品の呼び名を選択・生成する為のルールで, “対象物用”, “コントローラーおよびコントロールバルブ用”, “その他の部品用” の 3 種類に分かれる。

これらのルールは, オントロロジーサーバーのオントロロジー・モデル参照関数を用いて記述されており, オントロロジーサーバー内のオントロロジーやモデルを参照しながら, 呼び名の選択・生成が行われる。

呼び名の選択・生成がなされると, その語彙を用いて運転員に提示する説明文の生成がなされる。石油プラント運転時に運転員に提示する説明文は, 発見された異常や, 診断過程において測定された状態, 推論により導出した状態, または対策として運転員が行うべき操作である。つまり説明文の内容は, 各タスクの実行結果として得られる状態や操作である。説明文は,

警告 : 異常発見時に発見された異常状態を提示する

対策 : 対策操作を提示する

のように, 内容タイプによって分類することで, 警告であれば「警告: [異常状態] しています。’, 対策であれば「対策として [対策操作] してください。’, のような文章テンプレートを用意することができる。テンプレートに現れる [異常状態] や [対策操作] は [状態], [操作] のロール名であり, [状態] は

- <部品> の <状態語彙>
- <部品> の <機能> が <比較値>
- <物> の <属性> が <比較値>

[対策] は

- <部品> を <部品操作>
- <物> の <属性> を <属性操作>
- <物> の <属性> を <絶対値> に

のいずれかの形で表現される。[状態] や [対策] の <部品> や <物> <属性> <機能> などの表現には, 石油プラントオントロロジーの語彙が使用される。

例えば, インタフェースエージェント (IA) は事例ベース推論の機能を持ち, プラントとオペレーターの行動の監視し, 状況に応じてオペレーターに警告や操作の指示などを提示する。この際, IA が生成する出力結果はシュミレーター用の部品番号や流体, ナフサといった汎用語彙で表記され, 留分, 液レベルコントロールバルブといった状況に適した語彙が用いられていない。HM/OS のメッセージ生成エンジンは IA の出力結果を受け取り, プラントオントロロジーに基づき状況に応じた語彙を選択して適切なメッセージを返す。

このようにオントロロジーはシステムで使用される語彙を提供し, システム内部での語彙の統一に役立つ。また, メッセージ生成エンジンによって状況に適した, 運転員に理解しやすい語彙を選択することで, 計算機と人間の間の概念レベルのギャップを埋めることに貢献する。

### § 3 ロール概念による定義

§ 1 で述べたコントロールバルブの例を, ロール概念の扱いをサポートしたオントロロジーエディタの最新版 [古崎 02] で記述すると次のようになる。図 7 はプラントオントロロジーの一部を表したもので「流量コントローラー」および「液レベルコントローラー」は共に「コントローラー」を上位概念として持ち, “測定属性” といった属性 (a/o で示す) や, “検出部品” といった部分概念 (p/o で示す) を上書きして定義されている。なお, 図中 PV (Process value) はセンサの測定値, SV (Set value) はコントローラーの設定値, MV (Manipulated value) はアクチュエータへ操作するよう指示される値を意味する。

また「コントローラー」では“操作対象部品”として定義されていた部分概念は, クラス制約が「アクチュエータ」から「バルブ」に特殊化されている。さらにそのバルブが「流量コントローラー」では「流量制御関係」から定まる“流量操作 role”, 「液レベルコントローラー」では「液レベル制御関係」から定まる“液レベル操作 role”を担い, それぞれ「流量コントロールバルブ」\*1, 「液レベルコントロールバルブ」というロールホルダーとなる\*2。 「流量制御関係」や「液レベル制御関係」は部品が果たす機能を関係概念として概念化したものである。

このオントロロジーを基に図 6 で示したモデルを作成すると「流量コントローラー」のインスタンス FC29 と「液レベルコントローラー」のインスタンス LC29 が, 一つの「バルブ」VFC29 を部分として持つことになり, そのバルブはそれぞれのコントローラから定まる 2 つの role を果たし, 注目する部品に応じて別の名前の部品 (ロールホルダー) と認識される。このようにしてロール概念を用いることで, 注目するコンテキストに依存した認識の変化を, オントロロジーの基礎理論に基づいて適切に捉えることが可能となる。

### 4.6 その他の利用例

「法造」を用いてオントロロジーやモデルの構築・利用がなされた研究には SmartTrainer/AT [金 99] や, DEGREE [Barros 01], 機能オントロロジー [来村 02] などがある。

SmartTrainer/AT は, 教材オントロロジーに基づく電力系統事故復旧操作訓練システム構築用オーサリングツールある。このシステムはオントロロジーサーバーの操作関数を呼び出すことで「法造」で構築されたオントロロジーを利用する, OS 利用エージェントとして開発がなされ

\*1 現実世界のバルブは流量を制御することが本質的な機能として設計されている為, 本来の「流量コントロールバルブ」は基本概念であるが, ここではロールホルダー名に適当なラベルが見つからず基本概念と同じラベルを用いている。

\*2 正確には, 各コントローラーが「オーバーヘッドドラム」および「ナフサ抜き出しポンプ」に接続された際に, このバルブが「オーバーヘッドドラム液レベルコントロールバルブ」および「ナフサ抜き出し流量コントロールバルブ」となるが, 煩雑になるため本論文における記述を省略している。

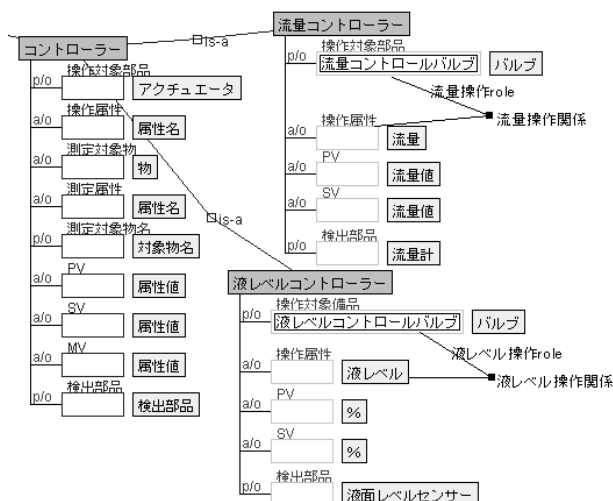


図 7 プラントオントロジーにおけるコントローラーの定義

ている。SmartTrainer/AT はオントロジーエディタで記述した教材オントロジーを基に、オントロジーサーバーが提供する操作関数を用いて教材モデルを作成する。その際、SmartTrainer/AT は教材作成に特化した独自のインタフェースをオーサに提供し、モデリング中に検出された制約違反に対しては、オントロジーサーバーの戻り値情報を、予め用意したテンプレートを用いて警告あるいはエラーメッセージに変換してオーサに提示する。例えば、“もし学習者は「初心者」で、誤り原因は「知識を知らない」、かつ学習タイプは「例題優先」であれば、原理を教える前に例題を提示するという教授行為を先に実行した方がよい。”(太字はオントロジーで定義されている概念を表す)といった公理に対する違反が警告として表示される。

一方、「DEGREE」は CSCL におけるインタラクション履歴解析システムで「法造」で構築したオントロジーを XML 形式で出力したものを読み込むことで動作する。オントロジーの構築に「法造」を用いることで、オントロジーベースのシステム開発のコスト削減や構築したオントロジーのメンテナンスに「法造」が貢献している。

これらのシステムを 2.2 節のオントロジーベースのアプリケーションの分類から見ると、SmartTrainer/AT が (2) common access to information, 「DEGREE」は、(1) neutral authoring の枠組みに相当する。

また、機能オントロジーの構築においては「法造」のオントロジーエディタが提供する基本概念とロール概念の峻別などをサポートした記述環境を用いることで、従来の記述環境では困難であったオントロジーの構築が可能になったという評価を受けている。その他にも「法造」は研究室の内外で様々なオントロジーやモデル構築に利用されており、ヒューマンメディアプロジェクトに参加したドメインの専門家から日常業務での利用を検討したいという評価も得られている。

## 5. 考察および関連研究

本章では前章で述べたプラントオントロジーの構築から利用の過程における、オントロジーおよび「法造」の貢献とその有用性について考察する。

まずプラントオントロジーを構築することで、システムが対象とするプラントに関する語彙や暗黙情報が明示され (2.1 節 (1) 暗黙情報の明示化)、4.2 節で述べたように、分散協調システム内の各エージェント間での語彙統一など知識の共有がなされた (2.1 節 (2) 共通語彙の提供)。また 4.4 節で述べたプラントモデルの構築の際には、オントロジーがモデル構築に必要な概念や関係の定義や前提条件を明示している為 (2.1 節 (3) 知識の体系化 (4) 標準化)、モデル構築のコスト削減や、構築されるモデルの一貫性の保証に貢献した。これはオントロジーの持つ (5) メタモデルの機能による。その際、基本概念とロール概念の違いを明らかにすることで、コンテキストに依存した認識の変化をオントロジーの基礎理論に基づいて適切に捉えることが可能となり、4.5 節で述べたメッセージ生成エンジンの開発で示されるように、計算機と人間の間の概念レベルのギャップを埋め親和性の高いシステム開発がなされた。

これらの過程で「法造」は前章で述べたように、オントロジーやモデルの構築・利用に必要な機能を提供し、オントロジーベースのシステム開発を容易にしている。前述の石油精製プラントの例では、構築された概念数約 400 のプラントオントロジーおよび部品数約 2000 というプラントモデルは、企業の専門家からも現実の石油精製プラントを扱うのに十分な規模を提供していると認められており、本システムが現実規模のオントロジー構築に十分耐えうるシステムであることが示されたと言える。さらに同プロジェクト内の開発システム (三菱電機 (株)、日石三菱石油 (株) が参加) において、これらのプラントオントロジーおよびモデルを利用したアプリケーションも開発された。

大規模なオントロジーの開発例については、エディンバラ大学の AIAI 研究所 [AIAI] とトロント大学 [TOVE] で開発されたエンタープライズオントロジーなど多数見られる。しかし、本論文のプラントオントロジーのように「オントロジー構築」「構築したオントロジーに基づくモデル構築」、そして「それらを利用したシステム開発」に至る一連の過程を通じた具体事例はまだ少ないのが現状である。このような点から、オントロジーの構築から利用までの一連の過程をサポートできる「法造」は、今後のオントロジー研究において重要な意味を持つと言える。なおこれ以外にも「法造」は様々なオントロジー構築に使用されており、オントロジー構築に必要な機能が提供されていることが確認されている。

一方、既存のオントロジー構築・利用環境としては、スタンフォード大学の KSL が開発を進めている Ontolin-

gua Server がある [Farquhar 96] . このシステムではネットワークを介したオントロジーの共同構築・利用がなされており, 構築されたオントロジーは様々な知識表現言語に変換がなされる. さらに, Fikes らは再利用可能な形で表現力豊かなオントロジーを扱うシステムに対する要求をオントロジー構築・利用に分けてまとめ, Ontolingua Server の拡張を進めている [Fikes 98] . また, Webベースのオントロジー構築環境としては, Swartout らの Ontosaurus [Swartout 96] , Mahalingam らの Java Ontology Editor ( JOE ) [Mahalingam 99] , Domingue らの WebOnto [Domingue 98] などがある .

「法造」は Ontolingua Server を初めとした, これらのシステムが提供している基本的な機能を実装している. これらのオントロジー構築環境と比較した「法造」の最大の特徴は, 基本概念とロール概念の峻別など, オントロジーの基礎理論に基づいた記述環境にある. この記述環境に関する詳細は別稿で報告している [古崎 02] が, 本論文ではロール概念に基づいた定義の例を含む, 「法造」を用いたオントロジーやモデル構築の具体例を通してシステムの有用性を示した .

## 6. む す び

本論文では, オントロジー構築・利用環境「法造」の概要を示し, 石油精製プラントオントロジーの構築から利用に至るまでの一連の過程を通した「法造」の使用例を報告した. 構築されたプラントオントロジーやモデルは, ほぼ現実に近い規模を持ち, これらのプラントオントロジーおよびモデルを利用したアプリケーションも開発されている. 「法造」はこれ以外にも様々なオントロジー構築に用いられており, 企業の専門家より日常の業務に使用したいという評価も得られている. これらの使用経験から, 「法造」はオントロジーの構築・利用を支援する計算機環境として, 必要な機能を提供し, 実際のオントロジー構築に耐えうるシステムであることが示された .

今後はこれらの経験から得られた知見をもとに「法造」の機能を拡張していくと共に, 一般公開の準備を進めていく .

## 謝 辞

本研究の一部は, 旧通産省産業科学技術研究開発制度として, 新エネルギー・産業総合開発機構からヒューマンメディアの研究開発として ( 財 ) イメージ情報科学研究所が委託を受け進めている研究開発の一環として行われた. また本研究は科学研究費補助金 ( 基盤研究 (B)(2)11480076 ) オントロジーの基礎理論とその開発環境に関する研究の援助のもとでなされた. なお本研究に関して様々な助言を頂いた, 大阪大学 池田満助教授に感謝致します .

## ◇ 参 考 文 献 ◇

- [AIAI] AIAI, : <http://www.aiia.ed.ac.uk/enterprise/enterprise/ontology.html>.
- [Barros 01] Barros, B., Mizoguchi, R., and Verdejo, F.: A Platform for Collaboration Analysis in CSCL. An ontological approach, in *Proceedings of AIED01*, St. Antonio, USA (2001).
- [Domingue 98] Domingue, J.: Tadzebao and WebOnto: Discussing, Browsing, and Editing Ontologies on the Web, *Proceedings of the 11th Banff Knowledge Acquisition Workshop*. (1998).
- [Farquhar 96] Farquhar, A., Fikes, R., and Rice, J.: The Ontolingua Server: a Tool for Collaborative Ontology Construction, *Proceedings of the 10th Banff Knowledge Acquisition Workshop* (1996).
- [Fikes 98] Fikes, R. and Farquhar, A.: Distributed Repositories of Highly Expressive Reusable Ontologies, *IEEE Intelligent System*, Vol. 14, No. 2, pp. 73-79 (1998).
- [石川 01] 石川, 久保, 古崎, 来村, 溝口: タスク・ドメインロ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 -, 人工知能学会誌 (投稿中) (2001).
- [金 99] 金, 林, 池田, 溝口 他: 訓練システム SmartTrainer 構築用オーサリングツール, 教育情報学会学会誌秋号, Vol. 16, No. 3, pp. 139-148 (1999).
- [来村 02] 来村, 笠井, 吉川, 高橋, 古崎, 溝口: オントロジーに基づく機能的知識の体系的記述とその機能構造設計支援における利用, 人工知能学論文会誌, Vol. 17, No. 1, pp. 73-84 (2002).
- [古崎 99a] 古崎, 久保, 来村, 池田, 溝口: オントロジー構築利用環境「法造」の設計と試作, 人工知能学会全国大会論文集 (第13回), pp. 374-377 (1999).
- [古崎 99b] 古崎, 久保, 来村, 池田, 溝口: オントロジー構築利用環境の開発 ~ 「関係」および「ロール概念」に関する基礎的考察 ~, 人工知能学会研究会資料 SIG-KBS-9803, pp. 13-18 (1999).
- [古崎 02] 古崎, 来村, 池田, 溝口: 「ロール」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 (2002).
- [Mahalingam 99] Mahalingam, K. and Huhns, M.: Java Ontology Editor (JOE) TUTORIAL (1999), <http://www.engr.sc.edu/research/CIT/demos/java/joe/>.
- [溝口 96] 溝口理一郎: 形式と内容 - 内容指向人工知能研究の勧め -, 人工知能学会誌, Vol. 11, No. 1, pp. 50-59 (1996).
- [溝口 97] 溝口, 池田: オントロジー工学序説, 人工知能学会誌, Vol. 12, No. 4, pp. 559-569 (1997).
- [溝口 98] 溝口, 来村: 工学知識のマネージメント, 第4章, pp. 155-179, 朝倉書店 (1998).
- [溝口 99a] 溝口理一郎: オントロジー研究の基礎と応用, 人工知能学会誌, Vol. 14, No. 6, pp. 977-988 (1999).
- [溝口 99b] 溝口, 池田, 来村: オントロジー工学基礎論, 人工知能学会誌, Vol. 14, No. 6, pp. 1019-1032 (1999).
- [Mizoguchi 00] Mizoguchi, R., Kozaki, K., Sano, T., and Kitamura, Y.: Construction and Deployment of a Plant Ontology, in *Knowledge Engineering and Knowledge Management - Methods, Models and Tools -*, The 12th International Conference, EKAW2000, Lecture Notes in Artificial Intelligence 1937, pp. 113-128, Juan-les-Pins, France (2000).
- [佐野 99] 佐野, 来村, 溝口: ヒューマンメディア・プロジェクトにおける石油プラントオントロジーの構築とその利用, 人工知能学会全国大会論文集 (第13回), pp. 378-381 (1999).
- [Swartout 96] Swartout, B., Patil, R., Knight, K., and Russ, T.: Toward Distributed Use of Large-Scale Ontologies, *Proceedings of the 10th Banff Knowledge Acquisition Workshop* (1996).
- [TOVE] TOVE, : <http://www.eil.utoronto.ca/tove/toveont.html>.
- [Uschold 99] Uschold, M. and Jasper, R.: A Framework for Understanding and Classifying Ontology Applications, in *Proceedings of the IJCAI-99 workshop on Ontologies and Problem-Solving Methods (KRR5)*, Stockholm, Sweden

(1999).

〔担当委員：木下哲男〕

2001 年 4 月 27 日 受理

## 著 者 紹 介



古崎 晃司(正会員)

1997 年大阪大学工学部電子工学科卒業。2002 年同大学院工学研究科博士課程修了。同年、化学工学会嘱託研究員、現在に至る。博士(工学)。現在、大阪大学産業科学研究科にて、ナノテクノロジーオントロジーの研究とオントロジーサーバーの開発、オントロジー構築利用環境の設計・開発に関する研究に従事。情報処理学会会員。



来村 徳信(正会員)

1991 年大阪大学基礎工学部情報工学科卒業。1993 年同大学院基礎工学研究科前期課程修了。同年、同大学産業技術研究所技官。1994 年同助手。現在に至る。博士(工学)。物理的システムに関するオントロジー工学的考察と、それに基づいたモデル化と推論に関する研究に従事。1996 年人工知能学会創立 10 周年記念論文賞受賞。情報処理学会会員。



佐野 年伸

2001 年 3 月大阪大学大学院工学研究科電子工学専攻修士課程修了。同年 4 月オンキヨー株式会社入社、現在に至る。在学中は、デバイスオントロジー、機能構造に関する研究に従事。



本松 慎一郎

2001 年 12 月大阪大学大学院工学研究科電子工学専攻修士課程修了。在学中は、オントロジーサーバーの設計・開発に関する研究に従事。現在(株)アイティブーストにてネットワーク/ソフトウェア技術に関する教育、およびアプリケーション開発を行う。著書『TECHNICAL MASTER はじめての Turbolinux7 server サーバ構築編』(秀和システム:2002 年)



石川 誠一

2002 年 3 月大阪大学大学院基礎工学研究科修士課程修了。同年 4 月エヌ・ティ・ティ・コムウェア株式会社入社、現在に至る。在学中は、タスク・ドメインロ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に従事。



溝口 理一朗(正会員)

1977 年同大学院基礎工学研究科博士課程修了。1978 年大阪大学産業科学研究所助手、1987 年同研究所助教授、1990 年同教授。現在に至る。工学博士。音声の認識・理解、エキスパートシステム、知的 CAI システム、オントロジー工学の研究に従事。1985 年 Pattern Recognition Society 論文賞、1988 年電子情報通信学会論文賞、1996 年人工知能学会創立 10 周年記念論文賞、1999 年 ICCE99 Best paper Award 受賞。電子情報通信学会、情報処理学会、教育システム情報学会、日本認知科学会、Intl. AI in Education(IAIED) Soc., AAAI, IEEE, APC of AACE 各会員。現在、IAIED Soc. 会長、及び APC of AACE の次期会長。