

[論文]

知的熟練の持続的表出化支援システムの構築

岡部雅夫^{†‡}, 小林圭堂[†], 石川達也[†], 飯島正[†], 山口高平[†]

要旨

知的熟練の継承が大きな問題となって久しいが、これは決して一過性の問題ではなく、スリム化された組織においても機能する新しい組織的継承の枠組みが求められている。筆者らは既にオントロジーとルールベース・システムを活用した知的熟練の組織的継承の枠組みを提案し、東京電力のある現場事業所での試行およびその評価実験により、熟練者がオントロジーやルールベース・システムへ表出化した知的熟練を新人が効率的に内面化できることが支持された。ただし、知的熟練も、経営環境等の変化に伴い、経営層の意思に従い、変化する。そのため、筆者らの提案が機能していくためには、熟練者が知的熟練の変化をオントロジーおよびルールベースへ持続的に表出化していくことが前提となる。そこで、本論文では、熟練者が経営層の意思に従って知的熟練を持続的に表出化していくことを支援するシステムを提案する。具体的には、ドメイン・オントロジーによりルール・オントロジー上の業務に直接使われる日本語の「最も浅いルール」を生成し、それをルールベース上の実行ルールに変換する機能、経営環境等の変化に対する経営層の意思をルール・オントロジー上の「最も浅いルール」およびルールベース上の実行ルールに反映する機能を提案する。併せて、新人による内面化に関しても、筆者らの先行論文での評価実験を踏まえ、業務プロセス・フローの詳細化を提案する。これらの提案は、試行事業所において熟練者から高い評価を得、その有効性が支持された。

Abstract

Transfer of intelligence skill to next generation has been a big issue for some time. To resolve this issue, we need a new sustainable framework that can work in a substantially downsized organization. We already proposed a new framework using ontologies and a rule-based system, and its trial application to some actual job at Tokyo Electric Power Company supported its effectiveness. Intelligence skill, however, has to evolve, reflecting business environment change and in accordance with management decision for it, and it is necessary for domain experts to externalize their intelligence skill continuously. Hence, in this paper, we propose a system that helps domain experts externalize their intelligence skill continuously in accordance with management decision. More specifically, we propose a functionality that generates the shallowest rules in Japanese controlled by domain ontologies in a rule ontology and transforms them to executable rules in a rule base, and a functionality that reflects management decision for business environment change in the shallowest rules and the executable rules. Also as for the internalization by novices, we propose a relatively

Development of Support System for Sustainable Externalization of Intelligence Skill

Masao Okabe^{†‡}, Keido Kobayashi[†], Tatsuya Ishikawa[†], Tadashi Iijima[†], Takahira Yamaguchi[†]

[†] 慶應義塾大学 大学院 理工学研究科

[‡] 東京電力株式会社 システム企画部

[†] Graduate School of Science and Technology, Keio University

[‡] Corporate Systems Department, Tokyo Electric Power Co., Inc.

[論文] 2010年5月12日受付

© 情報システム学会

detailed business process flow based on the evaluation of our earlier proposal. Domain experts of the job to which these proposals have been applied have evaluated that the proposals are effective and this supports the effectiveness of the proposals.

1 はじめに

ベテラン技術者の大量退職期を迎え、技術・技能の継承が大きな問題となって久しい。この問題は情報システムのサポートすべき大きな領域とされ、国家的なプロジェクト[1]も推進されたが、この問題は決して一過性の問題ではない。これまで技術・技能は暗黙知のまま OJT により継承されてきたが、組織のスリム化により OJT による継承が困難になりつつあることに原因があり、スリム化された組織においても、持続的に技術・技能が継承されていく枠組みが求められている。そのため、バーチャルリアリティ等の情報技術の適用が提案されている[2 他]が、それらが主にターゲットとしているのは、高度な金型の作成技能等の所謂「匠的熟練」である。一方で、我が国の技術・技能の本質は不確実性や変化をこなすノウハウである「知的熟練」にあるとされ、その「知的熟練」ははば広く深い OJT によりはじめて獲得されるとされる[3]。また、最近の高度に自動化・統合化された製造現場においては、「匠的熟練」がコンピュータによる数値制御に置き換えられつつある一方で、「知的熟練」の重要性はますます高まっている[4]。

「知的熟練」の本質は、膨大な知識の中から、様々な状況に応じて必要な知識を適切に論理的に組み合わせることにある。従って、「知的熟練」には知識を体系化するオントロジーやその論理的組み合わせを行うルールベース・システムが有効である。筆者らはこの点に着目して、オントロジーとルールベース・システムを活用した知的熟練の組織的継承を支援するシステムを提案した[5]。これは、熟練者が知的熟練をオントロジー及びルールベース・システムに表出化・連結化し、それを新人が内面化することを支援するものである。東京電力の知的熟練を要する典型的業務への

試験適用およびその評価は、この提案により、熟練者が表出化・連結化した知的熟練を、新人が、OJT と比較して十分に短い時間で、OJT とほぼ同等なレベルで内面化できることを支持したが、いくつかの不十分な点も明らかになった。まず、熟練者の観点から見ると、知的熟練も、経営環境等の変化に伴い、経営層の意思に従い、変化していくが、その変化に合わせて熟練者が持続的に知的熟練を表出化していけるかは確認できなかった。また、新人の観点から見ると、新人が業務を遂行するうえで一番の糸口となる手順的情報が不足していることも明らかになった。そこで、本論文では、[5]に対する拡張として、経営の視点も入れ、熟練者が経営環境等の変化に対する経営層の意思に従い知的熟練をオントロジーやルールベース・システムに持続的に表出化することの支援、さらには、経営層の意思をオントロジーやルールベース・システムに直接的に範囲させることを提案する。また、経営レベルから実務レベルまでの標準的な業務プロセス・フローも導入し、新人が業務を遂行する上での糸口も与える。

本論文の構成は以下の通りである。まず、2章で先行研究・ツールを概観し、3章で知的熟練の持続的表出化支援システムを提案する。4章では、この提案の東京電力における試行への適用について述べる。5章では、まず4章での適用に対する熟練者による評価と考察を述べ、それに基づき、本提案に対する評価と考察を行う。最後に6章でまとめと今後の課題について述べる。なお、本提案は筆者らの先行論文[5]に対する拡張であるため、本論文で必要となる範囲で [5]の概要を補遺1にまとめた。

2 先行研究・ツール

2.1 ナレッジマネジメント

知的熟練の継承は、広い意味で、ナレッジマネジメントの一分野と捉えることができる。ナレッジマネジメントにおいてはSECIモデル [6] がしばしば参照される。SECIモデルでは、共同化、表出化、連結化、内面化という4つのプロセスのスパイラル的繰り返しにより、暗黙知と形式知が相互に変換され、組織の知識が創造されるとされ、また、これらの相互変換は「場」において行われるとされる [7]。

SECIモデルは、知識の共有に留まらず、組織としての知識の創造に踏み込んだ知識経営を提唱しているが、知的熟練の次世代への継承という観点では、熟練者による知識の表出化およびその新人による内面化が中心となる。

さらに、不確実性や変化をこなすためには、様々な局面に応じた知識の活用が必要であり、連結化、即ち、体系化ないし構造化も重要である。業務知識・ノウハウの表出化・連結化の最も端的な例としては、体系化された業務マニュアルがあげられる。知的熟練においてはば広く深いOJTが重視されるということは、即ち、知的熟練すべてを体系化されたマニュアルとして表出化・連結化することが困難であること、また、それがなされたとしても膨大なマニュアルをOJT抜きに内面化することの困難さを意味している。土方らはSECIモデルに基づき、2人の熟練者が暗黙知を効率的に表出化・連結化できる「場」をシステムにより提供することを提案している [8]。

筆者らは、[5]において、熟練者の暗黙知の表出化・連結化とともに、初心者による内面化も支援し、知的熟練の組織的継承を支援する「場」を提案した。

2.2 ナレッジマネジメント・ツール

ナレッジマネジメントのための商用ツールはいくつか存在するが、それらはドキュメ

ント管理を核とする場合が多い。一般的には、ドキュメントに限定されずに、Webコンテンツ、業務システムの情報、メール情報等はば広い情報を知識データベースとして一括登録管理する機能を提供するとともに、必要な時に容易にアクセスできるように、全文検索等の機能が提供される [9 他]。ただし、あくまで対象となるのはすでにドキュメント等に表出化されている知識が中心となる。

また、所謂企業情報ポータルが多くの企業において導入され、コンテンツの充実が進む中で、企業情報ポータル向けのスイート製品の中にも、ナレッジマネジメント向け機能を謳うものもある [10 他]。その場合は、むしろ、電子会議室、Q&Aシステムのような組織横断的なコミュニケーション活性化による暗黙知の表出化・共有化支援機能が中心となるが、ただし、その場合も、ドキュメント、Webコンテンツ、業務システム等のはば広い情報にアクセスできる機能が前提となっていて、目的に応じポータルに集約した情報以外は全文検索等でのアクセスが中心となる。

どちらにも共通な特徴としては、すでに表出化されている知識はそのままの形式で登録・管理し、また、電子会議室等により新たに表出化された知識もそのまま登録され、その活用のためのアクセスは全文検索等に委ね、暗黙知の体系だった表出化、表出化された知識の体系化・構造化という視点は乏しい点が上げられる。

これらのナレッジマネジメント・ツールは、知的熟練を必要とする業務に対しても有効であるが、それはあくまで熟練者に対してである。熟練者といえども、必要なすべての知識を記憶していることはあり得ず、過去の技術文書等の参照が必要となることはしばしばある。これらのツールは、熟練者が必要な技術文書等を発掘し、さらに、その文書等に対し新たな知見を付加していくには有効である。ただし、利用者が全文検索等により必要な技術文書等を検索できるためには、必要な知識を特定できていて、また、検索した技術文書を自らの状況に当てはめて活用できる必要が

あり、そのためには利用者はすでにかかなりの知的熟練を獲得していることが前提となる。

そもそもこれまで OJT により継承されてきた知的熟練は、技術文書のような形で表出化されていないことが問題であり、知的熟練の熟練者から新人への継承という観点で見た場合、まず、知的熟練そのものを体系的に表出化することが必要である。また、不確実性や変化への対応が重視される知的熟練の継承には、単に直接的に必要となる知識を提供するだけでなく、それを様々な状況において活用できるように、関連する知識を含め、はば広く体系的に提供することが必要であり、知的熟練の表出化においては、単に技術文書といった形式ではなく、体系化・構造化することが重要である。その意味で、既存のナレッジマネジメント・ツールは知的熟練の継承という目的には十分とは言えない。

2.3 ナレッジマネジメントと情報技術

ダベンポートらは、知識の創造や移転等に関して、組織の文化的側面等を踏まえつつ情報技術の活用を実践的に論じている[11]。その中で、複雑な暗黙知を形式化することには多くの労力を要すること、また、労力を費やして暗黙知を形式化しても直接対面による信頼関係の構築なしには共有がなされない場合があること等を指摘しつつも、知識の共有のために形式化は必要であり、それを支える情報技術として構造化された明示的な知識の貯蔵庫が重要であるとしている。そして、そこからの確に知識を発掘する意味的検索を実現する上でソーラスの整備の重要性を指摘している。また、エキスパートシステムが有効であるための条件として、1) 特化されたテクニカルな問題領域であること、2) 知識の形式化が容易であること、3) 知識が安定していることをあげている。

先の[8]は、2人の熟練者の暗黙知に対しシステムが事例から帰納学習した知識を含めて不整合を提示することにより、整合の取れた知識の連結化を効率的に実現できることを示したものであるが、そのためにエキスパート

システムにおける知識獲得の手法が有効であるとして、2人の熟練者が暗黙知を if then 形式で表出化することが前提となっている。

一般的にエキスパートシステムでは知識の獲得と更新がボトルネックとなるが、その要因の一つとして、エキスパートシステムの開発がドメイン・エキスパートからの知識の抽出も含め、実装方式に依存した方法によっていることがあげられる。そのため、所謂知識システムにおいても、通常の情報システムと同様な統合的開発方法論の整備が進められていて、その中の一つに CommonKADS [12]がある。CommonKADS では、知識システムを単に専門家に代わって推論等を行うものとして狭く捉えることなく、組織の知的生産性を高めるためのものとして広く捉え、通常の情報システムの分析・設計と同様、課題設定、組織モデル、業務モデル(タスクモデル)といったハイレベルは分析から、実装方式とは独立のモデリングによりブレークダウンしていく。その中で核となるのが知識モデルであるが、知識モデルは実装方式とは独立の概念レベルのモデルであるが故に、ナレッジマネジメントに対しても知識の形式化・共有等の面で有効であるとされる。知識モデルでは、そこで実行されるタスクをタスク知識、そのタスクを実行する上で必要となる推論を推論知識としてモデル化するとともに、その推論において必要となる対象世界の静的構造を領域知識としてモデル化する。タスク知識、推論知識に関しては、その再利用性・メンテナンス性を高めるために、診断、スケジューリング等の知識システムの代表的なタスク毎に、テンプレート化されている。また、領域知識には、UML のクラス図に相当するものとして領域スキーマがあるが、クラス間の制約等に関しては、領域スキーマとは別に知識ベースとして実装独立にモデル化し、メンテナンス性・再利用性を高めているのが特徴である。さらに、領域スキーマに関しては、推論知識への適用性を高めるために汎用性を意識して体系化したオントロジー(2.4を参照のこと)に拡張される。

2.4 オントロジー

オントロジーは「概念化の明示的な規格」[13]として定義されるが、より一般的には、概念を汎用性を意識しつつ体系化するとともに、各概念に対する標準的な語彙を定めたものとして捉えられる。ダベンポートらが知識の貯蔵庫から意味的検索により的確に知識を発掘するためにシソーラスの整備の重要性を指摘していることは既に述べたが、昨今、ナレッジマネジメントにおいても、シソーラスを一步進めて、オントロジーの活用が進んでいる。例えば、ある保険会社では、スキル情報の管理のために、スキル情報を記述する上で必要となる概念をオントロジー化し、社員のスキル情報記述の標準化・統制を行っている[14]。ただし、その目的はシソーラスと同様、意味的検索の実現が主眼であり、スキル情報そのものの体系化という視点はない。また、自動車ボディの製造工程におけるゆがみの発生を抑えるノウハウの再利用のためのある事例推論システムでは、その事例の記述に必要な概念がオントロジー化され、事例を記述するに当たっては、プルダウン・メニューよりオントロジーの語彙を選択して記述する[15]。その目的は、事例推論のために各事例の類似性の尺度を導入することと同時に、事例の記述の属人性を排除し、理解性・再利用性を向上させることにある。ただし、それは記述様式および語彙の標準化というレベルにとどまっている。

人間の理解性向上のための知識の体系化という観点からみると、オントロジーを構成するプリミティブは語彙が対応する概念とは限らない。例えば、MIT Process Handbook [16]は自然言語によるビジネス・プロセスの記述であるが、それらの間に構成関係や汎化・特化関係等の意味的關係が定義され体系化されているが故に、オントロジーと呼ばれることがある。MIT Process Handbook は、様々なビジネス・プロセスを体系化することにより、それらを容易に理解・比較検討できるようにし、それを基に改善された新たなビジネス・プロセスを創出することを意図している。そ

ういう意味では、前節の CommonKADS の知識モデルをナレッジマネジメントの観点から見ると、オントロジーとして体系化されるべきものには、領域スキーマだけでなく、クラス間の制約等を表す知識ベースも含まれる。

筆者らは、[5]において、知識の理解性向上、内面化支援という観点から、用語に留まらず業務ルールをプリミティブとして体系化したオントロジーとその業務ルールの一部を実行形式に変換したルールベース・システムを組み合わせ、初心者による知的熟練の内面化の支援を提案した。

3 知的熟練の持続的表出化支援システム

3.1 スコープ

本提案を具体的に述べるに前に、まず、筆者らの先行論文[5]と本提案を合わせた全体のスコープを明確にする。対象とする組織は、知的熟練を必要とするあるまとまった責務を果たしている現場事業所である。大企業の場合であれば一現場事業所が該当し、中小企業であれば企業全体の場合もある。ただし、いずれにしても、そこには、当該事業所の長(事業所長、あるいは、経営企画・管理部門)があり、その方針の下で、知的熟練を要するいくつかの業務を遂行していて、組織のスリム化に伴い OJT による熟練者から新人への知的熟練の継承が困難になりつつあり、できる限り OJT によらない知的熟練の継承が求められている。

なお、ここで、知的熟練の継承の対象である新人とは、全くの新人は想定しない。業務に対する最低限の背景知識を持ち、業務マニュアルが整備されている場合であれば業務マニュアルに記載されていることは理解できるレベル、業務マニュアルが整備されていない場合であれば、標準的な業務に対する最低限の教育は受けているレベルを想定している。現在問題となっているのは、全くの新人の教育ではなく、スリム化された組織において知的熟練を継承していくためにいかにして熟練者を育成していくかであるからである。

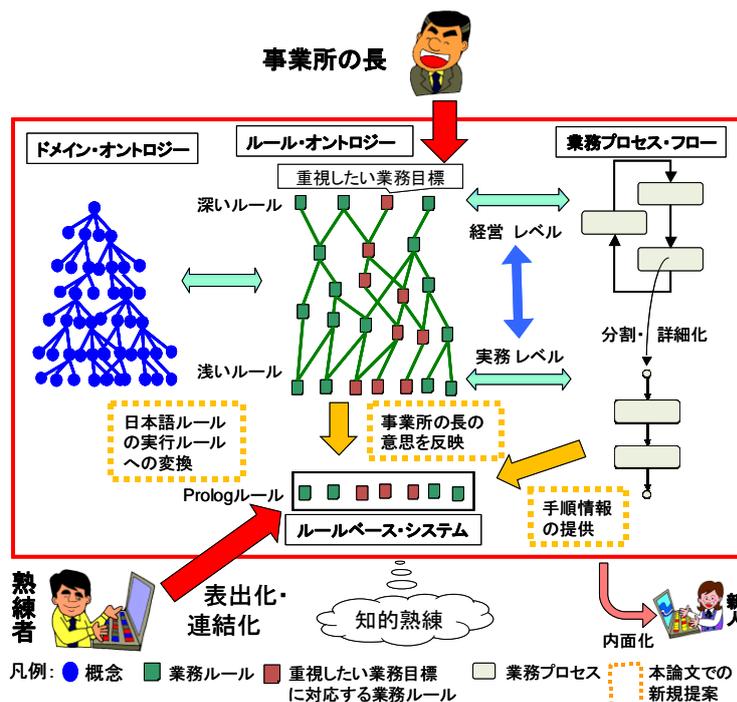


図1 提案の全体像

なお、対象となる知的熟練は、代々蓄積されてきたものであり、比較的安定しているものの、経営環境等の変化に伴い、事業所の長の意思に従って、当然変化していくものであり、熟練者による表出化・連結化は持続的なものであることが求められる。

3.2 提案の概要

筆者らの先行論文[5]を含む提案の全体像を図1に示す。

筆者らは[5]において、新人が、知的熟練を獲得し、熟練者に育っていくことを支援するために、熟練者が自らの知的熟練を表出化・連結化し、それを新人が内面化する「場」を提供するシステムを提案した。熟練者の知的熟練は、知的熟練を構成する業務ルールをプリミティブとして意味的關係を導入し体系化したルール・オントロジー、各業務ルールにおいて使用される専門概念を体系化したドメイン・オントロジー、業務ルールのうち業務に直接使われる「最も浅いルール」を実行ルールに変換したルールベース・システムに表出化・連結化され、それらを活用して新人が知的熟練を効率的に内面化できることが、試行および評価実験により支持された。詳細は補遺1を参照されたい。

ただし、経営環境等の変化に伴い、事業所の長の意思に従って、熟練者が表出化された知的熟練を持続的に更新し続けられるかは確認できていない。そこで、本論文では、熟練者が持続的に知的熟練を表出化していくことの支援を主眼とする。本システム全体の初期導入においては、熟練者に加えて、オントロジーやルールベース・システムを専門とするナレッジエンジニアが加わることにより、知的熟練の表出化・連結化は実現される。このことは、筆者らの先行論文[5]における試行においても支持された。問題はそれ以降にある。経営環境等の変化により、当然、知的熟練も変化していく。熟練者自らが、経営環境等の変化に応じて、事業所の長の意思に沿って、持続的に知的熟練を表出化していくことが求められる。本論文は、そのための支援を提案する。

具体的には、第一に、業務に直接使われる日本語の「最も浅いルール」¹をドメイン・オ

¹ ある業務ルールが他の業務ルールの正当性を説明している場合に、その関係を正当性関係と呼び、相対的に、正当性を説明されるルールを「浅いルール」、説明するルールを「深いルール」と呼ぶ。「最も浅いルール」は業務に直接使われるルールであり、「最も深いルール」は業務の目標や基本的な制約を表すものとなる。詳細は補遺1を

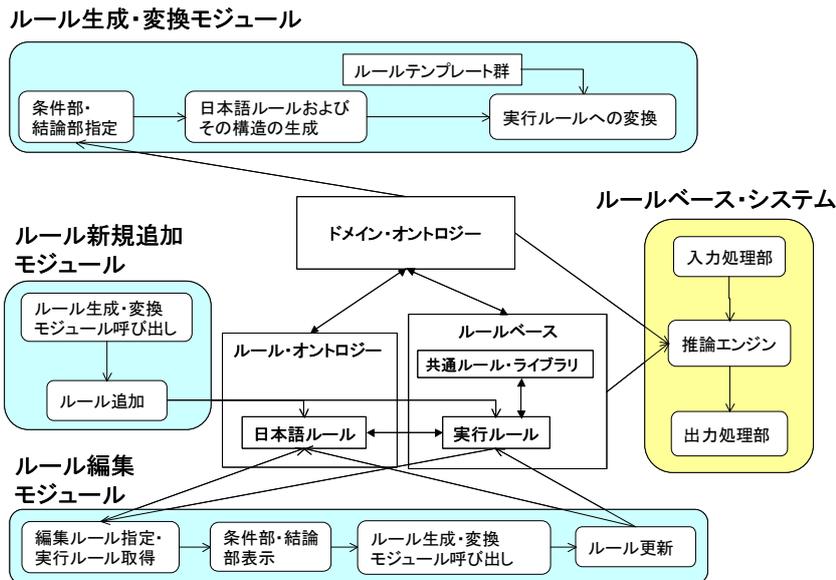


図2 日本語ルール生成と実行ルールへの変換のモジュール構成

ントロジーからの語彙の選択により構成することにより、日本語の「最も浅いルール」からルールベース上の実行ルールへの変換を可能にする。これにより、熟練者はルールベース上の実行ルールを直接意識することなく、業務に直接使われる日本語の「最も浅いルール」を管理することにより、ルールベース上の実行ルールも管理できるようになる。

第二に、事業所の長の視点を導入し、ルール・オントロジーにおける正当性関係¹を、新人の理解を支援することに留まらず、事業所の長の視点と実務者の視点を結び付けるものに拡張する。第一の提案が意味を持つためには、熟練者が、経営環境等の変化に伴い、事業所の長の意思に従い、変更すべき浅いルールを認識できることが前提となるからである。さらに一歩進め、事業所の長の意思に基づきルール・オントロジーやルールベースそのものを直接変化させることも提案する。

第三に、業務プロセス・フローを経営レベルと実務レベルを結び付けるものとして位置づけると共に、実務レベルの業務プロセス・フローに関しては、新人が業務を遂行する上での糸口を得る上で必要な詳細化は行うこととする。なお、この三点目は、熟練者による知的熟練の持続的な表出化を支援するものではなく、筆者らの先行論文[5]における評価実

験を踏まえた新人による内面化に関する改善提案である。

次節において、それぞれに関し、詳しく述べる。

3.3 提案システム

3.3.1 ドメイン・オントロジーからの日本語ルール生成と実行ルールへの変換

「最も浅いルール」の日本語記述をドメイン・オントロジーからの語彙の選択により生成するよう統制し、それにより、ルールベース上の実行ルールへの変換を可能にする。筆者らの先行論文[5]においては、ドメイン・オントロジーは、各業務ルールを構成する専門概念を体系化したものであり、ドメイン・オントロジーの語彙の多くは、専門用語である名詞ない名詞句が中心であったが、本論文では、ドメイン・オントロジーは「最も浅いルール」の日本語記述を生成するため必要な概念をすべて含むものに拡張される。ドメイン・オントロジーからの語彙の選択により日本語記述を生成できるようにするため、「最も浅いルール」は、if 条件部 then 結論部（日本語記述としては「xxx である場合は、yyy である。」等）の形式に統一され、例えば、条件部においては期間を条件指定する際に必要となる「上期」「下期」、結論部においては「同時に実施する」等の述語句的なもの、あるいは、

参照のこと。

「開始日」「終了日」等がドメイン・オントロジーに追加される。

システムの概略のモジュール構成を図 2 に示す。現状、ルールベース・システムは Prolog で開発しており、ルールベースも Prolog での記述を前提としている。ドメイン・オントロジーからの日本語ルール生成と実行ルールへの変換を担う各モジュールは Java にて開発している。

ここで、中核となるのは、ルール生成・変換モジュールである。ルール生成・変換モジュールは、ドメイン・オントロジーの語彙を選択することによりルールの条件部、結論部を構成し、そこから日本語ルールを生成する。また、ルール生成・変換モジュールは、粗粒度の日本語記述を細粒度の実行ルールに変換するために、様々な条件部、結論部に対応した Prolog のルールテンプレートを保持している。ルール生成・変換モジュールは、条件部、結論部の指定に応じて、適当なルールテンプレートを選択し、そのルールテンプレートに条件部、結論部で選択したドメイン・オントロジーの語彙を埋め込むことにより、実行ルールを生成する。必要なルールテンプレートを予め準備できるために「最も浅いルール」は類型化できる必要がある。なお、ルールテンプレートには、(年度, 月) と (年, 月) の変換、年月日とユリウス通日との変換等の共通に用いられるルールの呼び出しが含まれる。これらの共通に用いられるルールは、あらかじめ、ルールベース上に共通ルール・ライブラリとして定義してある。このルール生成・変換モジュールを用い、ルール新規追加モジュールは新たなルールの追加を行い、ルール編集モジュールは既存のルールの更新を行う。なお、どちらの場合も、生成された日本語によるルール記述に不自然さが残る場合も排除しきれないため、日本語によるルール記述に関しては後から修正できるようにしている。

3.3.2 ルール・オントロジーを活用した事業所の長の意思の反映

上記により、日本語の「最も浅いルール」からルールベース上の実行ルールへの変換がなされるとしても、それが意味を持つためには、「最も浅いルール」が持続的に適切に更新されていく必要があり、経営環境等の状況の変化に対し、事業所の長の意思を反映し、更新すべき「最も浅いルール」が適切に認識されることが前提となる。現実には、一度ルールが確立されると、本来より深いルールを達成するための手段である「最も浅いルール」自体が目的化し、経営環境等の変化により更新されなければならないにもかかわらず、そのまま残ってしまうことはよくある。

そこで、ルール・オントロジーを、新人が業務の全体象を理解するためのものだけでなく、熟練者にも、事業所の長の意思に従って、更新する必要がある「最も浅いルール」を的確に認識する上で活用するものとして積極的に位置づけることとする。

ただし、一般的には、「最も深いルール」そのものは普遍性が高く、多くの場合、経営環境等の変化があっても、「最も深いルール」そのものが入れ替わるのではなく、既存の「最も深いルール」の重要度に変化が生じる。一方で、浅いルールと深いルールの関係は多くの場合多対多の関係になり、「最も深いルール」の重要度に変化が生じた場合に、それに追随すべき浅いルールの把握が必ずしも容易ではない。そこで、多対多の関係を維持しつつも、浅いルールから見て、それも最も強く正当化している深いルールを一つ選定した。それにより、各「最も浅いルール」から見て、それを中心的に正当化する「最も深いルール」が一つに定まるようになり、また、「最も深いルール」から見ても、それが中心的に正当化している「最も浅いルール」が絞られ、把握が容易になる。

さらに、一歩進めて、経営環境等の変化に対し「最も深いルール」の重要度を変化させるという事業所の長の意思が示された場合に、ルール・オントロジーを活用して「最も浅い

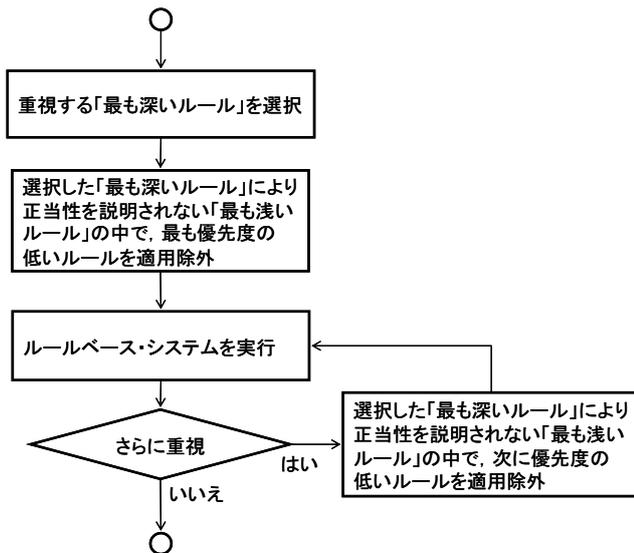


図3 ルールの適用除外アルゴリズム

ルール」およびルールベース上の実行ルールを変化させ、ルールベース・システムが事業所の長の意思に沿った結果を半自動で出力できるようにすることを提案する。「最も浅いルール」および実行ルールを変更するアルゴリズムはいくつか考えられるが、著者らは先行研究[5]において知的熟練を要する業務が様々な状況を考慮する必要があることへの対応として、「最も浅いルール」に優先度を与え、実行解が存在しない場合は、優先度の低い実行ルールから適用を除外して実行解を得るといった方式を提案したが、その方式に沿ったアルゴリズムとして、図3に示されるものがある。すなわち、選択した「最も深いルール」により正当性を説明されない「最も浅いルール」の中で、優先度の最も低いものを適用除外する。より自然には、「最も深いルール」により正当性を説明される「最も浅いルール」の優先度を高くすることも考えられるが、可能解が存在する場合に特定のルールの優先度を上げて結果は同一になるため、上記のアルゴリズムとする。

3.3.3 ルール・オントロジーを活用した事業所の長の意思の反映

最上位の経営レベルの業務プロセス・フローは、当該事業所の長が業務全体を概観する上で必要になるレベルで記述したものであり、CommonKADSにおける最もハイレベル

表1 オントロジーのプリミティブ数

	プリミティブ数	備考
ルール・オントロジー	134	内、最も浅いルール 90
ドメイン・オントロジー	292	内、設備オントロジー、作業オントロジーのプリミティブ 254

な業務モデルに相当する。一方、それをブレークダウンしていった最下位の実務レベルの業務プロセス・フローは、実務者が業務を遂行する上で必要となるレベルのものであり、CommonKADSにおける知識モデルのタスク知識、推論知識に相当する。継承しようとしているものは様々な状況を考慮する必要がある知的熟練であり、手順的な業務プロセス・フローを詳細にすべて書き上げることは現実的には難しいが、新人が業務を遂行する上で必要となる「最も浅いルール」を適用する手順に関する情報を得るために必要な詳細化は行うこととする。

4 東京電力における試行への適用

4.1 先行論文での試行およびその評価

提案システムの東京電力のある現場事業所における試行への適用を説明する前に、その上で必要となる範囲で、筆者らの先行論文[5]における東京電力での試行を概説する。詳細は補遺1を参照されたい。

試行は、「停止調整業務」と呼ばれる水力発電関係の現場事業所の業務を対象に行われた。「停止調整業務」とは、その現場事業所が管理するすべての水力発電所設備の停止を要する点検・修理の主として年間スケジュール（以下、停止スケジュール）を作成する業務である。スケジュールは、様々な制約の下に、² 水電力²を最小にするように作成される。制約は多様で、多くは長年の経験により暗黙知として形成されたもので、必ずしもすべて

² 水とは、本来発電に供されるはずであったものが、発電所の停止等により発電に供されなかった水のことであり、水電力とは、水が発電に供されたならば発電されたであろう電力量のことである。

の制約を満たす解が存在するとは限らず、状況に応じて長年の経験に基づき制約を緩和する必要がある、この業務は「知的熟練」を要すると共に、その「知的熟練」が表出化されていない典型的な業務である。

筆者らは、この業務の熟練者と一体となって、ドメイン・オントロジー、ルール・オントロジー、ルールベース・システムとしてのスケジューリング・システムを構築した。各オントロジーのプリミティブの数は表1の通りである。また、スケジューリング・システムは、状況に応じてルールを緩和する必要があるが、それぞれの状況に応じて緩和ルールを決定することは組み合わせの数が膨大になり現実的ではないため、各ルールに優先度を与え、解が存在しない場合は、優先度の低いものから適用を除外することとした。様々なケースに対し、熟練者が適切と考える解が得られるよう、試行錯誤により各ルールに優先度を与えた結果、15段階の優先度によりほとんどの場合に熟練者が適切と考える解が得られるようになった。

その上で、本プロジェクトのメンバー5名を被験者として、実際に停止スケジュールを作成する評価実験を行った。その結果、業務知識の無い状態でオントロジーおよびルールベース・システムを活用して独習した被験者は、熟練者からの直接的指導により業務知識を獲得した被験者との比較において、ほぼ1/4の学習時間で、ほぼ同等の精度の停止スケジュールを作成することができ、新人に対する知的熟練の内面化支援としての提案システムの有効性が支持された。ただし、本提案に従いオントロジーおよびルールベース・システムを活用して独習した被験者は業務手順情報が不足していることに起因する誤りが見られた。一方、熟練者による知的熟練の表出化・連結化に関しても、プロジェクト・メンバーと共同作業により実際に構築されたオントロジーやルールベース・システムが新人の知的熟練の内面化に有効に寄与したという意味において、提案システムの有効性が支持された。ただし、今回は初期構築であったため、

熟練者は本プロジェクト・メンバーと一体となってオントロジーやルールベース・システムを構築したが、今後は、熟練者自らが、経営環境等の変化に伴い、事業所の長の意思に従い、オントロジーやルールベース・システムを持続的に更新していく必要があるが、それが可能であるかどうかは確認できなかった。

4.2 提案システムの適用

以上を踏まえ、本節では、3.3 提案システムにあげた3点の本試行への適用について述べる。

4.2.1 ドメイン・オントロジーからの日本語ルール生成と実行ルールへの変換

4.2.1.1 最も浅いルールの分析

ドメイン・オントロジーからの「最も浅いルール」の生成を可能にするため、「最も浅いルール」の分析を行った。「最も浅いルール」90個の中には、点検周期に関するルールや、水電力をできるだけ少なくするための可能解の探索順に関するルールも含まれている。従前から点検周期は点検周期表により管理されており、また、可能解の探索順に関しては各河川の過去30年間平均の月別河川水量表に基づき決定されていた。どちらのルールもこれらの表に基づき管理するのが最も管理しやすい形態と考えられたため、これらの表から実行ルールを生成することとした。その上で、これらのルールを除いた「最も浅いルール」72個を類別化した。まず、ルールを単独作業ルールと相互依存作業ルールに分類した。例えば、「A発電所のB作業は12月に実施する」というルール³はA発電所のB作業に関する単独作業ルールである。また、例えば、「(電力需要の高い)夏季は(電力供給信頼度を維持するために)停止作業は行わない」といったルールも、夏季に実施されるあらゆる作業が対象となるが、それらの個々の作業に対し独立に適用されるルールであるから、単独

³ 本節で例示するルールはすべて説明用に簡略化したものであり、実際のものとは異なる。

作業ルールである。一方、「A 発電所の B 作業と C 作業は同時には実施しない」といったルールは、A 発電所の B 作業と C 作業の相互依存の制約を規定している。このようなものは、相互依存作業ルールである。また、相互依存作業ルールには、この例のように対象作業の相互依存性に対称性が成り立つ場合と、「A 発電所の B 作業は C 作業の終了後に実施する」といった対象作業の相互依存性に対称性が成り立たない場合がある。前者を対称ルール、後者を非対称ルールと呼ぶことにする⁴。なお、対称ルール、非対称ルールとも、可能性としては、3 つ以上の作業間の多項関係の制約となる可能性もあったが、実際には 2 項関係以上のものはなかった。理論上は多項関係があり得ても実際には 2 項関係までで十分であることが実務上は多いが、この場合も同様であった。

以上の分類を踏まえ、これらのルールを、“if 条件部 then 結論部”という形式にパターン化した。ここで、結論部は設備と作業を指定しただけでは特定されず、その設備のその作業の実施時期、作業日数にも依存する場合がある。そのため、条件部では、設備、作業に加えて、実施時期、作業日数も指定できるようにした。

条件部は、以下の各項の or 指定の and 解釈により指定される。

項 1：設備の指定

発電所、発電機（注：発電所によっては複数の発電機を有する。その場合、発電機も指定することにより設備を識別する。）

項 2：作業の指定

作業、回（注：作業によっては年に同一設備に対して複数回実施するものがある。その場合、1 回、2 回等、何回目の作業であるかにより識別す

る。）

項 3：実施時期の指定

上期／下期、月、旬、日の指定 ないし 期間指定

項 4：作業日数の指定

具体的日数の指定 ないし $m+7n$ ($m=1,\dots,6, n=0$ または任意の自然数) による指定（注：曜日との関係を意識したもの）

ただし、相互依存作業ルールの場合には、相互依存関係の及ぶ範囲を指定することが必要になる場合がある。対象ルールにおいては、例えば、「A 発電所の各発電機に対する B 作業が同一月に実施される場合は、(作業効率の観点から) 連続して行う」というルールにおいて、連続して行うという制約の対象となる作業は、A 発電所の各発電機の B 作業であり、かつ、実施時期は何月でも構わないが同一月にあるものである。そこで、各指定項に対し、例えば、以下のように同一月であることを指定できるようにした。

項 1：設備の指定

発電所=A 発電所

項 2：作業の指定

作業=B 作業

項 3：実施時期の指定

月=同一

また、相互依存作業ルールの内、非対称ルールに関しては、複数の対象作業の集合を指定する必要が生じるが、前述のようにルールは現実には 2 項関係までであるので、対象作業の指定は 2 つまで可能とし、それらの間の相互依存関係の及ぶ範囲は対象ルールの場合と同様にして指定することにした。例えば、「A 発電所の B 作業は同一月に C 発電所の D 作業がある場合は、C 発電所の D 作業の後に「行う」の場合は、以下のようになる。

⁴ 単独作業ルール、相互依存作業ルール・対象ルール、非対称ルールの正確な定義は補遺 2 を参照のこと。

対象作業 1	対象作業 2
項 1：設備の指定 発電所=A 発電所	項 1：設備の指定 発電所=C 発電所
項 2：作業の指定 作業=B 作業	項 2：作業の指定 作業=D 作業
項 3：実施時期の指定 月=同一	項 3：実施時期の指定 月=同一

表 2 結論部の類別によるルール分類

	単独作業 ルール	相互依存作業ルール		計
		対称ルール	非対称ルール	
期間指定	22			22
除外期間指定	12			12
曜日指定	17			17
重ねない		6	4	10
同時に行う			7	7
連続して行う		1	1	2
その他	1	1		2
計	52	8	12	72

以上のように条件部を類別化した上で、結論部の類別化を行った。結論部は、大きく以下の 7 種類に類別された。

- 期間指定
- 除外期間指定
- 曜日指定
- 重ねない
- 同時に行う
- 連続して行う
- その他

最初の 3 つ「期間指定」「除外期間指定」「曜日指定」は「単独作業ルール」に関する結論部である。ここで、「期間指定」「除外期間指定」共に、開始日に関するもの（例：「4 月上旬までに開始する」）、終了日に関するもの（例：「5 月末までに終了する」）、全期間に関するもの（例：「4 月 15 日～20 日の間に実施する」）等、いくつかのパターンが存在する。「曜日指定」は、作業日数に応じて、作業工程上の作業幅輻日との兼ね合いから、開始曜日を定めるものである。次の 3 つ「重ねない」「同時に行う」「連続して行う」は「相互依存作業ルール」に関するものである。ここで、「重ねない」は作業日程を重ねないことを意味しているが、重ねない期間に関し「全期間を重ねない」、(作業が特に輻輳する日である)「開始日、終了日を重ねない」等、やはりいくつかのパターンが存在し、対称性が成り立つ場合と成り立たない場合が存在する。また、「同時

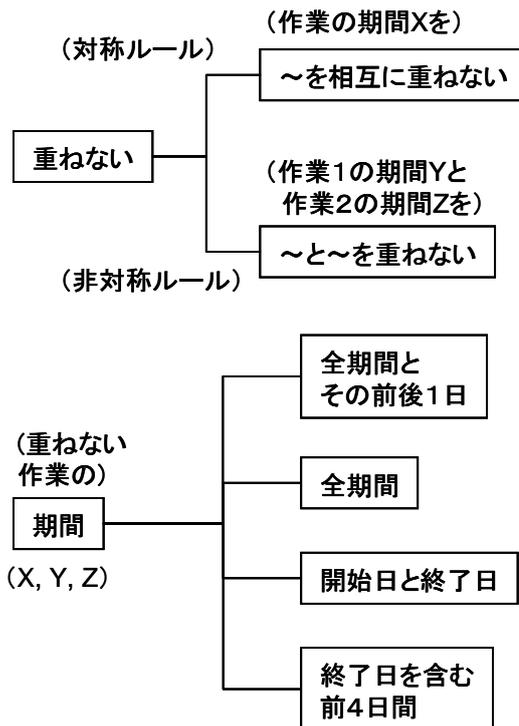


図 4 「重ねない」のパターン

に行う」に関しては、作業期間の短い作業を他の作業期間の長い作業の実施中に実施することを意味し常に非対称であり、「連続して行う」に関しては対称、非対称の双方がある。最後の「その他」は個別的な制約であって、他のルールにおいて再利用される可能性が少ないものをここに分類した。この結論部の類別に従って、72 個の実行ルールを分類したのが表 2 である。さらに、この結論部の類別に基づき、「その他」を除いた各類別のパターンを整理した。一例として、「重ねない」のパターンを図 4 に示す。

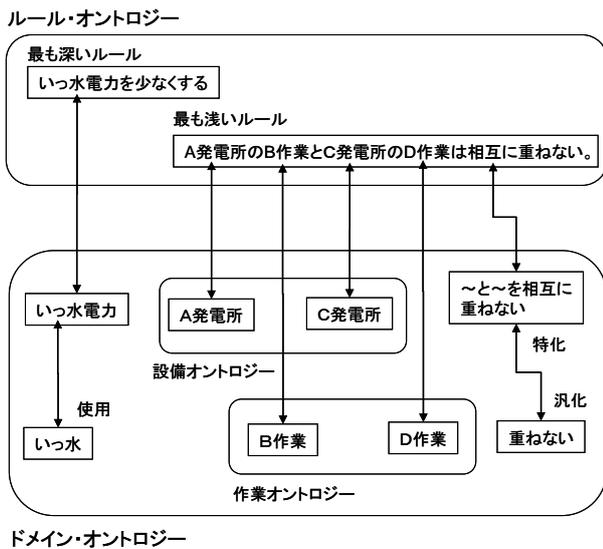


図5 ルール・オントロジーとドメイン・オントロジーの関係

4.2.1.2 ドメイン・オントロジーの拡張

以上を踏まえ、ドメイン・オントロジーの拡張を行った。停止調整業務という業務の特質から、先行論文[5]においては、ドメイン・オントロジーの主たるものは、設備に関するもの（以下、設備オントロジー）と停止作業に関するもの（以下、作業オントロジー）となったが、本提案においては、前節での最も浅いルールの分析に基づき、図4の「重ならない」等のパターンにて洗い出された各用語等、最も浅いルールを構成する上で必要となる用語をドメイン・オントロジーに組み入れた。また、深いルールに関しても、それを構成する用語の内、新人に有用と思われるものは、ドメイン・オントロジーに組み入れた。それ

らのルールとドメイン・オントロジーの関係を図5に示す。

その結果、ドメイン・オントロジーを構成するプリミティブは、設備オントロジーと作業オントロジーの254個に加え、その他の専門的概念や浅いルールの結論部を構成するための語彙として75個、計329個になった。

4.2.1.3 日本語ルールの生成と実行ルールへの変換

以上を基に、プルダウン・メニューからドメイン・オントロジーの語彙を選択し、日本語ルールを生成し、さらにそれを実行ルールに変換する。

1例として、結論部が「開始日，終了日を相互に重ならない」である対称ルールの場合を示す。図6がプルダウン・メニューからのドメイン・オントロジーを選択する画面であり、そこから生成される日本語ルール「上期に実施する作業日数が(3+7n)日のA発電所の内部点検は開始日，終了日を相互に重ならない」になる。また、そこから生成されるPrologの実行ルールは図7のようになる。この実行ルールの生成にあたっては、図8に示す対称ルールのテンプレートの一つおよび図9に示す「開始日，終了日を相互に重ならない」に対応するテンプレートを使用している。また、条件部を満たす作業を判別するうえで、上期/下期の判定を行うjiki等の予めルールベース上に定義されている共通ルール・ライブラリを使用している。



図6 実行ルールへの変換のためのプルダウン・メニューによる指定

```

rule87(ResultStock,a発電所,内部点検,3,上期,StartDate,EndDate):-!,
    rule87_2(ResultStock,a発電所,内部点検,3,上期,StartDate,EndDate).

rule87(____):- true.

rule87_2(ResultStock,Setubi,Sagyo,Length_Class,Jiki,StartDate,EndDate):-
    rule87_3(ResultStock,Setubi,Sagyo,Length_Class,Jiki,StartDate,EndDate).

rule87_2(____):- skip_rule87,true.

rule87_3([],____):- true.

rule87_3([[Setubi2,G2,Sagyo2,OtherStartDate,OtherEndDate]]R,Setubi,Sagyo,Length_Class,Jiki,StartDate,EndDate):-
    member(Setubi2,[Setubi]),
    member(Sagyo2,[Sagyo]),
    Length_Class2 is (OtherEndDate-OtherStartDate+1)%7,
    member(Length_Class2,[Length_Class]),
    julian_henkan(OtherStartDate,[_,Month2,_]),
    jiki(0,Month2,Jiki2),
    member(Jiki2,[Jiki]),
    !,
    StartDate =/= OtherStartDate,
    StartDate =/= OtherEndDate,
    EndDate =/= OtherStartDate,
    EndDate =/= OtherEndDate,
    rule87_3(R,Setubi,Sagyo,Length_Class,Jiki,StartDate,EndDate).

rule87_3([_|R],Setubi,Sagyo,Length_Class,Jiki,StartDate,EndDate):-
    rule87_3(R,Setubi,Sagyo,Length_Class,Jiki,StartDate,EndDate).
    
```

図7 変換された実行ルール

```

rule#xx(ResultStock,#setubi,#sagyo,#lengthclass,#jiki,StartDate,EndDate):-!,
    rule#xx_2(ResultStock,#setubi,#sagyo,#length_class,#jiki,StartDate,EndDate).

rule#xx(____):- true.

rule#xx_2(ResultStock,Setubi,Sagyo,Length_Class,Jiki,StartDate,EndDate):-
    #xx_3(ResultStock,Setubi,Sagyo,Length_Class,Jiki,StartDate,EndDate).

rule#xx_2(____):- skip_rule#xx, true.

rule#xx_3([],____):- true.

rule#xx_3([[Setubi2,G2,Sagyo2,OtherStartDate,OtherEndDate]]R,Setubi,Sagyo,Length_Class,Jiki,StartDate,EndDate):-
    member(Setubi2,[Setubi]),
    member(Sagyo2,[Sagyo]),
    Length_Class2 is (OtherEndDate-OtherStartDate+1)%7,
    member(Length_Class2,[Length_Class]),
    julian_henkan(OtherStartDate,[_,Month2,_]),
    jiki(0,Month2,Jiki2),
    member(Jiki2,[Jiki]),
    !,
    #結論部テンプレート,
    rule#xx_3(R,Setubi,Sagyo,Length_Class,Jiki,StartDate,EndDate).

rule #xx_3([_|R],Setubi,Sagyo,Length_Class,Jiki,StartDate,EndDate):-
    rulexx_3(R,Setubi,Sagyo,Length_Class,Jiki,StartDate,EndDate).
    
```

注：#で始まる文字列が置換される文字列

共通ルール・ライブラリの利用

図8 使用された対称ルールのテンプレート

```

StartDate =/= OtherStartDate, StartDate =/= OtherEndDate, EndDate =/= OtherStartDate, EndDate =/= OtherEndDate
    
```

図9 「開始日、終了日を相互に重ねない」のテンプレート

なお、個別的な制約であり表 2 で「その他」に分類された結論部に関しては、その結論部そのものに対応する用語をドメイン・オントロジーに個別に定義しプルダウン・メニュー

から選択できるようにするとともに、それに対応するルールテンプレートを個別に用意している。

以上の結果、少なくとも、現状の 72 個の

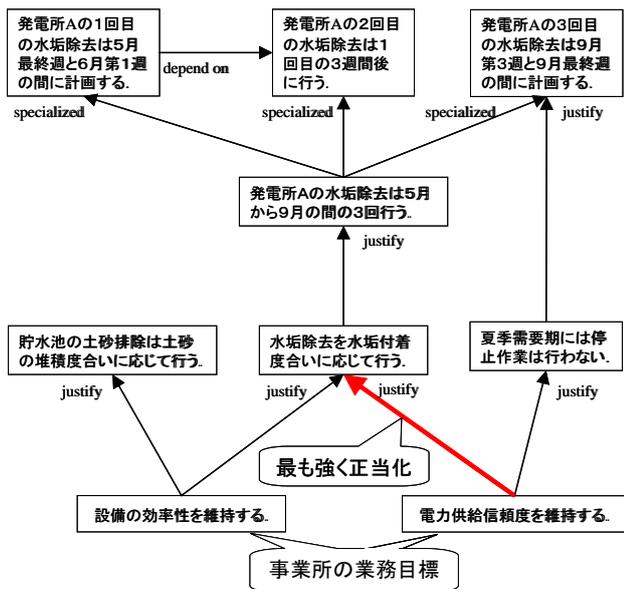


図 10 業務ルール間の意味の関係の例

「最も浅いルール」に関しては、プルダウンメニューからの選択により日本語ルールの生成および実行ルールへの変換ができるようになった。また、新たなルールに関しても、多くの場合、既存のパターンを組み合わせることにより日本語ルールが生成され、また、実行ルールへも変換されるものと期待される。

4.2.2 ルール・オントロジーを活用した事業所の長の意思の反映

ルール・オントロジーを熟練者に事業所の長の意思を伝えるものとして拡張したことに伴い、浅いルールから見て、それも最も強く正当化している深いルールを一つ選定したが、本試行への適用における例を図 10 に示す。「水垢除去を水垢の付着度合いに応じて行う」を最も強く正当化しているのは赤の太い矢線で示した「電力供給信頼度を維持する」であり、水垢付着は、設備の効率性を阻害するだけでなく、設備の停止の原因になる場合があり、水垢除去は、設備の効率性の観点以上に、電気供給信頼度を維持するために行われていることが分かる。

また、経営環境等の変化に伴い「最も深いルール」の重要度を変化させるという事業所の長の意思が示された場合に、ルール・オントロジーを活用して「最も浅いルール」と共にルー

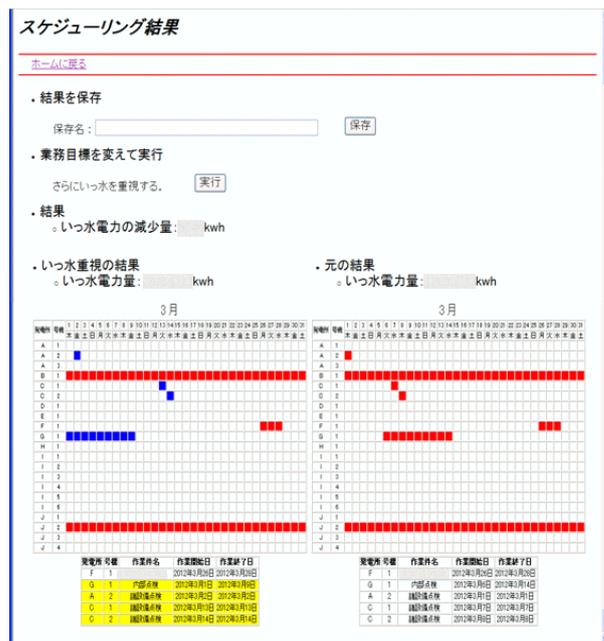


図 11 「いっ水電力を少なくする」をより重視して再作成した停止スケジュール

ルベース上の実行ルールも変化させ、スケジューリング・システムが停止スケジュールの再作成する機能に関しては、「いっ水電力を少なくする」をより重視する場合を実装した。実行ルールを変化させるアルゴリズムは図 3 に示したものである。「いっ水電力を少なくする」をより重視して停止スケジュールを再作成し、停止スケジュールおよびいっ水電力の変化を示す例を図 11 に示す。青色のバー及び黄色にハッチングされた作業が再スケジューリングにより変更された部分である。「さらにいっ水を重視する」「実行」をクリックするたびに、「いっ水電力を少なくする」により正当性を説明されないルールが、優先度の低いものから順次に適用が除外されていく。実際に、どのような状況においてどの程度いっ水電力が削減されるかを見るために、実際の典型的な3つのケースに対して、「さらにいっ水を重視する」のレベルに応じて、どの程度いっ水電力が削減されるかを調べた。その結果を表 3 に示す。例えば、この表で優先度 12 の行は、優先度 13, 14, 15 の「いっ水電力を少なくする」により正当性を説明されないルール計 7 個が適用除外され、ルールの適用除外率は $7/72=9.7\%$ であることを示している、その場

表3 いっ水電力重視によるルール適用除外率と いっ水電力削減率

if then 形式の「最も浅いルール」					いっ水電力削減率(%)		
優先度	計	「いっ水電力を少なくする」により正当性を説明されない		適用除外率(%)	Case1	Case2	Case3
		されない	される				
1	26	22	4	56.9%	-	-	-
2	5	5		50.0%	-	-	-
3	2	2		47.2%	-1.83%	-0.68%	-0.13%
4	4	4		41.7%	-1.77%	-0.66%	-0.12%
5	4	3	1	37.5%	-1.77%	-0.66%	-0.12%
6	2	2		34.7%	-1.77%	-0.66%	-0.12%
7	10	10		20.8%	-1.77%	-0.66%	-0.12%
8	5	4	1	15.3%	-1.77%	-0.66%	-0.12%
9	1	1		13.9%	-1.66%	-0.66%	-0.11%
10	1	1		12.5%	-1.66%	-0.66%	-0.11%
11	1	1		11.1%	-1.66%	-0.66%	-0.11%
12	1	1		9.7%	-1.62%	-0.29%	-0.02%
13	2	2		6.9%	-0.05%	0.00%	-0.01%
14	4	4		1.4%	0.00%	0.00%	0.00%
15	4	1	3	0.0%	0.00%	0.00%	0.00%
計	72	63	9				

合のいっ水電力の削減率は、それぞれ、1.62%、0.29%、0.02% になり、大きな開きがあることが分かる。

これは、それぞれのケースのもともとのスケジュールにおいてどの程度のルールの適用除外が発生しているかによる。例えば、Case1では、もともとのスケジュールにおいて適用除外されているルールが少ない。そのため、「いっ水電力を少なくする」により正当性を説明されない優先度の低いルールを除外すると、実行可能解空間が広がり、その中から、いっ水電力がすくなくなるよう探索をし直すため、相対的に、いっ水電力の削減率が大きくなる。一方、Case3では、もともとのスケジュールがタイトであり、既にルールの適用除外が多く発生していたため、改めて、「いっ水電力を少なくする」により正当性を説明されない優先度の低いルールを除外しても、それらのルールの多くは、実際にはスケジュール作成上既に除外されていたため、いっ水電力はほとんど削減されない。Case2はその中間にあたる。

また、すべてのケースにおいて、優先度11(即ち、優先度12以降を適用除外)、優先度12(優先度13以降を適用除外)において、いっ水電力の削減率の変動が相対的に大きいですが、これは優先度12,13ルールが状況によって実行可能解空間をいっ水電力を大きくする方向

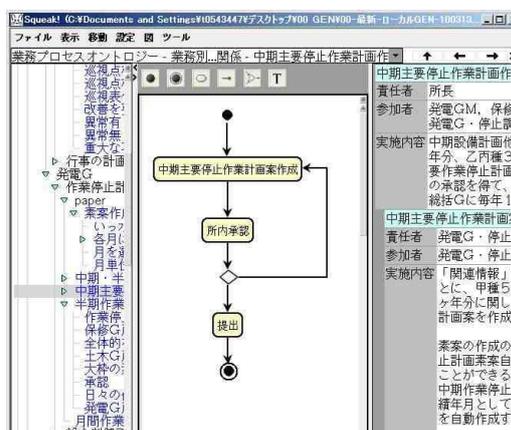


図12 停止調整業務の実務レベルの業務プロセス・フロー (当初)

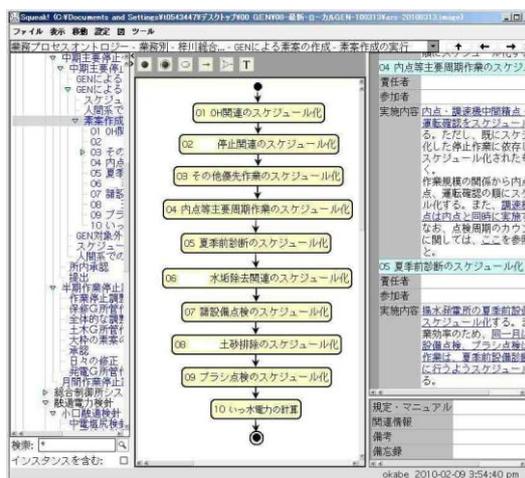


図13 停止調整業務の実務レベルの業務プロセス・フロー (改善後)

に制約するものであるためである。それぞれのケースにおいて、優先度12,13のルールがもともとのスケジュール作成においてどの程

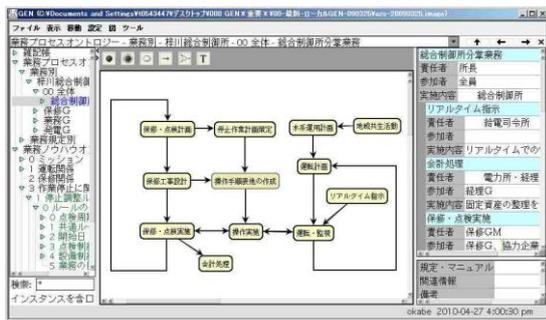


図 14 最上位の業務プロセス・フロー

度適用除外されているかの差が、その段階での各ケースのいっ水電力の削減率の大きな差を生んでいる。

4.2.3 業務プロセス・フローの詳細化

様々な状況を考慮する必要がある知的熟練では、手順的な業務プロセス・フローを詳細にすべて書き上げることは現実的には難しい。そのため、筆者らの先行論文[5]では、手順的な業務プロセス・フローはできるだけ汎用的なレベルに留め、それを補うものとして、ルール・オントロジーを導入したが、実験による評価では、業務知識の無い状態でオントロジーおよびルールベース・システムを活用して独習した被験者にとっては手順的な情報が不足し、そのために大きな手戻りが発生して、誤りの要因となっていることが明らかになった。

そこで、図 12 の業務プロセス・フローをもう一段詳細化し、当初よりも詳細な手順情報を記載することにした。それが、図 13 である。ただし、図 13 の各業務プロセスは、基本的には、いっ水電力への影響の大きい作業から優先的にスケジュール化するよう順序付けられており、解が存在せず一部の業務ルールを適用除外する必要がある場合も、以前の業務プロセスに戻る必要がないよう考慮されている。また、事業所長の視点を入れるために、図 12 をさらに抽象化し、事業所業務全体をカバーする業務プロセス・フローも導入した (図 14)。

5 評価と考察

本章では、まず、前章での試行への提案システムの適用に対する評価と考察を行い、次に、それに基づき、提案システムに対する評価と考察を行う。

5.1 提案システムの適用

前章での試行への提案システムの適用に対する評価は試行事業所における当該業務の熟練者 2 名にお願いした。

5.1.1 熟練者による持続的な表出化・連結化

まず、熟練者がプルダウン・メニューからの日本語ルールの生成とルールベース上の実行ルールへの変換機能により、将来必要となるであろうルールを生成できるかどうかを評価した。具体的には、熟練者 2 名に、その操作方法を説明した後、状況が変化した場合を想定して、自由に新たなルールを生成してもらった。生成したルールは、おおよそ 10 個である。その結果、熟練者が生成しようとするルールに「ルール生成・変換モジュール」が対応していない場合はなく、かつ、操作上も、熟練者はプルダウン・メニューからの選択により問題なくルールを生成できることが確認できた。生成された日本語ルール記述に関しても、通常日本語と全く同等とまでは言えないが、誤解を生むような記述となることはなかった。生成された日本語ルール記述とそれを熟練者が修正した例を表 4 に示す。

なお、4.2.1.1 に従い、条件部にて設備、作業に加えて、実施時期、作業日数を指定できるようにしている。一方、生成されるルールは基本的には設備に対する作業に関するルールであるため、条件部で設備、作業に加えて実施時期、作業日数を指定することに対し、熟練者が違和感を抱く可能性があった。特に、実施時期に関しては、結論部が「期間指定」「除外期間指定」(表 2 を参照のこと) の場合は、実施時期が条件部とともに結論部にも現れる。この条件部に対する熟練者による評価は、この実施時期、作業日数を含めた

表4 生成された日本語ルール記述と熟練者が修正した日本語ルール記述

生成された日本語ルール記述	熟練者が修正した日本語ルール記述
10月に実施するA発電所のすべての作業は、開始日は10月10日以降とする。	10月に実施するA発電所のすべての作業は、10日以降に開始する。
上期に実施するA発電所の内部点検は、開始日、終了日を相互に重ねない。	上期に実施するA発電所の内部点検の開始日と終了日を相互に重ねない
同一月に実施するB発電所1号機のC作業と同一月に実施するD発電所2号機のE作業は、前者の開始日と後者の全期間は重ねない。	B発電所1号機のC作業の開始日は、同一月に実施するD発電所2号機のE作業とは重ねない。

表5 条件部に実施時期、作業日数を指定するルール数

条件部	ルール数	内、結論部にて実施時期を制約
実施時期指定有り	10	7
作業日数指定有り	9	0
全数	72	-

注：条件部にて実施時期、作業日数の両方を指定するルールはない。

条件部の指定は熟練者のこれまでの思考と合っているとのことであった。実際、既存の72個のif then形式で管理されるルールの中で、条件部で実施時期、作業日数を指定しているルールは表5にあるように、19個ある。また、実施時期指定の10個の内、7個は、結論部にてさらに絞り込んだ実施時期を指定するものであった。

全体としてプルダウン・メニューからの日本語ルールの生成と実行ルールへの変換機能に対する評価は高く、熟練者からは、このようにプルダウン・メニューから日本語ルールの生成と実行ルールへの変換ができれば、自分たちでルールベース・システムを維持していけるという評価を得た。

ただし、将来的には、今回用意したパターンに収まらないルールが発生する可能性は否定できない。例えば、相互依存作業ルールの条件部に関しては、2項関係に収まらない多項関係が必要になるかもしれないし、また、結論部に関しても、現在の「その他」以外にも個別に対応しなければならないものが発生する可能性もある。このような可能性に対する対応としては、条件部に関しては極力現在の枠組みを維持し、それを超える場合は結論部側での個別に対応することを想定している。例えば、既存の枠組みでは記述できない条件Aを持つルール(if A then B)が必要になった場合は、Aを既存の枠組みで記述できる条件

A' と記述できない条件Eにより $A \equiv A' \cap E$ と分割し、(if (A' \cap E) then B) と (if A' then (if E then B))が同値であることを利用して、条件部はA'として既存の枠組みに抑え、結論部にて(if E then B)に個別対応することを基本方針とする。それにより、既存の枠組みでは対応できないルールが発生した場合でも、個別対応は結論部に限定される。結論部の個別対応を熟練者に委ねるのは無理があるにしても、その発生頻度を考えると、システム部門による対応が十分に可能であると考えられる。

また、いつ水電力低減をより重視した停止スケジュール再作成機能に関して、熟練者から肯定的な評価を得た。その理由としては、これまででも、停止スケジュールを作成した結果、いつ水電力が想定よりも多くなってしまった場合等において、いつ水電力を少なくするために停止スケジュールの再作成がよく行われていたが、再作成しても必ずしも期待したようないつ水電力の削減が得られないことも多く、それが業務量増大につながっていたことがあげられる。

今回開発したいつ水電力低減をより重視した停止スケジュール再作成機能による3ケースの検証でも、再作成によるいつ水電力の削減率に大きな開きがあることが明らかになったが、この機能により迅速に停止スケジュールを再作成し、かつ、いつ水電力の削減率も明らかにできる。このことは、熟練者のいつ水電力低減をより重視した停止スケジュール再作成に対する業務量軽減に貢献するだけでなく、事業所の長のレベルと実務者のレベルを結び付けるという点でも重要な意味を持つ。すなわち、この機能により、事業所の長は「いつ水電力をより少なくする」と

いう意思を示した後、実務レベルにおいてそれにどの程度応えられるかを直ちに知ることができ、必要であれば別の方策をとることの意味決定が迅速に行えるようになる。例えば、個々の停止作業は既定とした上での「停止スケジュール」の再作成では所定の削減が出来ないことが明らかになれば、事業所の長は、個々の停止作業の工程に踏み込んで、個々の停止作業の作業日数の短縮に取り組むという次の方策を示すこともできる。

5.1.2 業務プロセス・フローの詳細化

詳細化した業務プロセス・フロー（図 13）に関しては、新人から見ると、作業をスケジュール化する順序が明らかになり、それにより、手戻りの発生も小さくできるために、メリットがあることは明白であるが、このような手順情報を含めることは、持続的な更新をおこなっていく上での障害になることも危惧された。そこで、熟練者に評価してもらったところ、以下のような肯定的な評価を得た。

- － 熟練者自身は、初期のスケジュールを作成する場合このような手順は意識していない。
- － ただし、新人がスケジュールを作成する上では参考になる。
- － この手順は、いつ水電力への影響の大きいものを優先的にスケジュール化するようになっていて、スケジュール対象作業に大きな変更がない限り、汎用的で安定している。
- － また、スケジュール対象作業に大きな変更が生じた場合も、いつ水電力に大きな影響を与えるものを優先するという基本的考えに則れば、更新することは困難ではない。

5.2 提案システムに対する評価と考察

次に、前節での試行への適用に対する評価に基づき、提案システムに対する評価および考察を行う。

5.2.1 熟練者による持続的な表出化・連結化

プルダウン・メニューからのドメイン・オントロジーの語彙の選択によるルール・オン

トロジー上の日本語ルールの生成とルールベース上の実行ルールへの変換機能は、熟練者が、日本語の「最も浅いルール」を管理することにより、ルールベース上の実行ルールを、直接意識することなく、持続的に管理できるようにすることを意図したものである。ここで、留意すべき点は、熟練者自らによるルールベース上の実行ルールの管理が持続的に行われるためには、プルダウン・メニューによるルール・オントロジー上の日本語ルールの生成と実行ルールへの変換機能も持続的に有効であることが求められることである。

本提案機能の停止調整業務への適用において、熟練者による短期的な評価ではあるが、その有効性が示唆されたが、その前提となった停止調整業務の性質に関する事項を以下に挙げる。

まず、第一に、停止スケジュールの作成に直接使われる「最も浅いルール」が類型化され、かつ、曖昧性の少ない日本語として記述出来たことが挙げられる。「最も浅いルール」は、4.2.1.1 に述べたように類型化され、さらに、結論部が「期間指定」「重ねない」等曖昧性が少ないものであったことにより、ドメイン・オントロジーからの語彙の選択による日本語ルールの生成が可能となった。

第二に、第一の要因から派生するものであるが、日本語ルールをルールベース・システム上の実行ルールに変換する上での具体化する必要のある事項が比較的限定されていたことが挙げられる。一般的に、日本語ルールが人間にとっては明確であっても、それをルールベース・システムが実行できる実行ルールとするには、さらに具体化を要する事項が多くある。停止調整業務においては「最も浅いルール」が類型化されてことを受けて、例えば 4.2.1.3 の図 8 や図 9 に示したように、ルールテンプレートおよび共通ルール・ライブラリを組み合わせることで適用することにより、必要な具体化を行うことができた。

第三に、状況に応じてルールを緩和しなければならぬことに対し、「最も浅いルール」に優先度を与え、解が得られるまで優先度の

低いものから適用を除外することで対応できたことが挙げられる。これにより、個々の状況に応じた膨大な緩和ルールを定める必要がなくなり、実行ルールに変換する日本語ルールが限定されることになった。

最後に、停止スケジュールの作成という業務の性格から、洗い出された「最も浅いルール」の類型化が比較的安定であることが想定されたことが挙げられる。具体的には、既存の類型が当てはまらないルールが必要になった場合でも、条件部に関しては既存の枠組みを維持し、ドメイン・オントロジーに結論部に対応する語彙を追加し、また、その結論部の変換に必要なテンプレートを用意することにより対応する方針としたことは、5.1.1で述べた通りであるが、この方針が有効であると想定できたのは、このような既存の枠組みに収まらないルールの発生頻度が低いと推定されたからである。

以上より、停止調整業務においてプルダウン・メニューによるルール・オントロジー上の日本語ルールの生成とルールベース上の実行ルールへの変換機能が持続的に有効に機能することが示唆される上で、前提となった業務の性質は以下のようにまとめることが出来る。

- 「最も浅いルール」が日本語として明確に記述でき、かつ、類型化できること。
- 状況に応じて「最も浅いルール」の緩和が必要となる場合も、実行ルールへの変換とは独立に、優先度の設定等により行えること。
- 既存の類型に当てはまらない「最も浅いルール」の発生頻度が低いこと。

また、ルール・オントロジーを活用した事業所の長の意思の反映に関しては、経営環境の変化に伴い、事業所の長の意思に従い、熟練者が変更すべき浅いルールを正しく認識することを支援するものであり、さらには、それを一歩進め、事業所の長の意思に基づき

ルール・オントロジーやルールベースそのものを直接変化させるものである。

本提案機能の停止調整業務への適用においては、いつ水電力低減をより重視した停止スケジュールの再作成機能として実装された。これにより、事業所の長の意思が実務レベルにおいてどの程度反映されるかを迅速に検証することが可能になり、事業所の長の視点と実務者の視点が結び付けられ、事業所の長は必要に応じて別の方策を示すことが可能となることが示唆された。

これは、事業所の長が重視する「最も深いルール」を選択し、その「最も深いルール」により正当性を説明されない「最も浅いルール」を優先度の低いものから順次除外することにより実現されており、そのためには、以下が前提となる。

- 経営環境等の変化に対する事業所の長の意思は、「最も深いルール」そのものの変更には及ばず、既存の「最も深いルール」間の重要度の変更に留まること。
- 除外する実行ルールを選定するために「最も浅いルール」に優先度の設定されていること。

以上を踏まえると、逆に、以下のような性質を満たす業務に対しては、熟練者による持続的な表出化・連結化に対する本提案が有効に機能するものと考えられる。

- (1) 「最も深いルール」そのものは普遍性が高く、経営環境等の変化に対する事業所の長の意思は、既存の「最も深いルール」の重要度の変更によって示されうる可能性が高いこと。
- (2) 業務を遂行する上での直接必要となる「最も浅いルール」が日本語として明確に記述され、かつ、類型化できること。
- (3) 既存の類型に当てはまらない新たな「最も浅いルール」の発生頻度が低い

こと。

- (4) 状況に応じて「最も浅いルール」の緩和が必要となる場合も、個々の状況に応じて緩和ルールを定めることなく、優先度の設定等により対応できること。

この内、(1)に関しては、本提案が対象としている業務は、その知的熟練が OJT により長年に渡り暗黙知として蓄積・継承されてきた業務であるため、成熟し比較的安定していることが期待され、経営環境の抜本的な変化がなければ、満たされる可能性が高い。その上で、(2)(3)(4)が満たされる可能性が高い典型的な業務としては、今回適用した水力発電設備に関する停止調整業務に限らず、設備の点検等の作業の年間計画レベルでのスケジューリング業務が挙げられる。東京電力は、水力発電設備以外にも、様々な発電設備、流通設備等の設備を有しており、設備に対する点検等の作業のスケジューリング業務は数多く存在するが、それらの業務に対して本提案が有効に機能する可能性が高い。また、このような作業のスケジューリング業務以外においても、所謂制約充足問題で、その制約が長年の経験に基づき形成されている業務は、その制約を構成する「最も浅いルール」が非常に複雑で多岐にわたる場合を除いては、(2)(3)(4)が満足される可能性が高いと考えられる。

5.2.2 業務プロセス・フローの詳細化

本提案が対象とする業務は様々な状況を考慮する必要がある知的熟練を要する業務であり、様々な状況に応じて手順的な業務プロセス・フローを書き上げること、また、それを経営環境の変化に追従して更新していくことは現実的には困難である。また、業務プロセス・フローでは業務に対する理解が表層的なものになる危険性もある。そのため、業務プロセス・フローは無理なく記述できる標準的のものに留め、それに代わるものとしてルール・オントロジーを導入した。ただし、新人

には、ルール・オントロジーだけでは、業務を遂行する上でどういう手順で「最も浅いルール」を適用するかの情報が十分ではない。そこで、業務プロセス・フローにて、新人が業務を遂行する上で必要となる「最も浅いルール」を適用する手順を示す。

停止調整業務において今回行った詳細化は、新人に必要な「最も浅いルール」を適用する手順を示すと共に、一方で、経営環境等の変化に対しても、無理なく更新していける可能性が高いという評価を熟練者から得た。その意味で、停止調整業務における業務プロセス・フローの詳細化レベルは、先行研究[5]を含めて本体提案全体を適用する上で、業務プロセス・フローの記述の一つのガイドラインになるものである。ただし、この業務プロセス・フローの詳細化レベルは、対象とする新人の業務習熟度のレベル、対象とする業務の業務プロセスの複雑さ、多様性にも依存することに留意する必要がある。

6 まとめと今後の課題

本論文では、熟練者が自らの知的熟練をオントロジーやルールベースへ持続的に表出化することを支援するシステムを提案した。筆者らは、先行研究[5]において、あるまとまった責務を果たしている事業所を対象に、長年に渡って OJT により暗黙知のまま継承されてきた知的熟練が組織のスリム化により OJT による継承が困難になりつつあることを踏まえ、オントロジーやルールベース・システムを活用した知的熟練の組織的継承を支援する枠組みを提案し、熟練者がオントロジーやルールベース・システムに表出化した知的熟練を新人が効率的に内面化できることを特定の業務においてではあるが確認した。ただし、知的熟練も経営環境等の変化に伴い、事業所の長の意思に従い、変化する。従って、熟練者による知的熟練の表出化も、経営層の意思を反映しつつ、持続的に行われる必要があるが、これは必ずしも容易ではない。まず、熟練者がルールベース上の実行ルールを持続

的に更新していくことは技術的に容易ではない。また、知的熟練を、経営環境等の変化に応じて、事業所の長の意思に従って、変化させていくことは、ルール・オントロジー上の日本語ルールに対しても必ずしも容易ではない。知的熟練は実務レベルでの長年の経験に基づき形成されてきたものであり、経営との結びつきが強いからである。

そこで、本論文では、ドメイン・オントロジーを活用し、ルールベース上の実行ルールをルール・オントロジー上の日本語の「最も浅いルール」から変換することを提案するとともに、「最も深いルール」と「最も浅いルール」を結ぶルール・オントロジーを活用して、「最も深いルール」に対する事業所の長の意思をルール・オントロジー上の「最も浅いルール」に反映させることを提案した。これにより、熟練者は、事業所の長の意思に沿って、更新すべき「最も浅いルール」を把握し、ルールベース上の実行ルールと併せて更新することが可能になる。さらに、一歩進め、「最も深いルール」に対する事業所の長の意思に沿って、ルール・オントロジー上の「最も浅いルール」やルールベース上の実行ルールを直接変更することも提案した。

これらの提案は東京電力における試行に適用され、熟練者から高い評価を得た。東京電力における特定の業務における試行ではあるが、先行研究[5]における評価実験を踏まえた新人による内面化のための改善としての業務プロセス・フローの詳細化と併せて、本論文における提案および先行研究[5]における提案全体で、知的熟練の持続的な継承支援として、熟練者による事業所の長の意思に沿った持続的な表出化、および、新人によるその内面化の両面において有効性が確認され、知的熟練の持続的な継承の求められる業務に対して、一定の範囲で、本提案が有効であることが示唆された。

ただし、今回の試行対象業務は、知的熟練を要すると共にそれが表出化されていない典型的な業務ではあったが、スケジューリングというルールベース・システムの適用しやす

い業務であったことが幸いした面も否定できない。また、熟練者による事業所の長の意思に沿った持続的な表出化に関しては、あくまで熟練者による可能性としての評価であり、実際に長期に渡って検証したわけではない。これらの点を踏まえ、今後は、対象業務ドメインを広げつつ、実務において長期的に適用し、より確実な立証をしていくとともに、必要な改善を積み重ねていきたい。

謝辞

本研究においては、東京電力(株) 柳澤雅彦氏、上條員利氏、山崎浩志氏、山田和芳氏に当該業務の熟練者として参加いただいた。また、慶應義塾大学 吉岡亜紀子氏、竹田百合恵氏には、様々な示唆・協力を頂いた。厚く御礼申し上げます。

参考文献

- [1] 森下泰, “デジタル・マイスター・プロジェクト”, 精密工学会誌, Vol. 67 巻, 5 号, pp. 709-710, 2001 年 7 月.
- [2] 綿貫啓一, “バーチャルリアリティ技術による匠の技の伝承と人材育成”, 精密工学会誌 Vol72 No.1, pp.46-51, 2006 年 1 月.
- [3] 小池和夫, “仕事の経済学 第 3 版”, 東洋経済新報社, 2005 年 3 月.
- [4] 中馬宏之, “もの造り現場における問題発見・解決型熟練一二極分化傾向の背景”, 日本労働研究雑誌 特別号 No. 510, pp. 4-14, 2002 年 12 月.
- [5] M. Okabe, A. Yoshioka, K. Kobayashi and T. Yamaguchi, “Organizational Knowledge Transfer Using Ontologies and a Rule-Based System”, IEICE TRANS. INF & SYS, Vol. E93-D, No. 4, pp.763-773, 2010 年 4 月.
- [6] I. Nonaka and H. Takeuchi, “The Knowledge-Creating Company: How Japanese Companies Create the Dynamics of Innovation”, Oxford University Press, May, 1995 (梅本勝博訳, “知的創造企業”, 東洋経済新報社, 1996 年 3 月).
- [7] I. Nonaka and N. Konno, “The Concept

- of 'ba': Building a foundation for knowledge creation", California Management Review, Vol. 40, No. 3, pp. 40-54, Spring, 1998.
- [8] Y. Hijikata, T. Takenaka, Y. Kusumura and S. Nishida, "Interactive knowledge externalization and combination for SECI model", Proceedings of the 4th international conference on Knowledge capture, Whistler BC Canada, pp. 151-158, Oct., 2007.
- [9] KnowledgeMeister
http://www.toshiba-sol.co.jp/pro/km2/ .
- [10] INSUITE http://www.insuite.jp/ .
- [11] T. H. Davenport and L. Prusak, "Working Knowledge, How Organizations Manage What They Know", Harvard Business School Press, 1998. (梅本勝博訳, "ワーキング・ナレッジ「知」を活かす経営", 生産性出版, 2000年11月).
- [12] G. Schreiber, H. Akkermans, A. Anjewierden, R. Hoog, N. Shadbolt, W. V. Velde and B. Wielinga, "Knowledge Engineering and Management The CommonKADS Methodology", The MIT Press, 1999.
- [13] T. R. Gruber, "A Translation Approach to Portable Ontology Specifications", Knowledge Acquisition, Volume 5, Issue 2, p.199-220, June, 1993.
- [14] T. Lau and Y. Sure, "Introducing Ontology-based Skill Management at a large Insurance Company", Proceedings of the Modellierung 2002, Tutzing, Germany, 2002.
- [15] A. P. Morgan, J. A. Cafeo, D. I. Gibbons, R. M. Lesperance, G. H. Sengir and A. M. Simon, "CBR for Dimensional Management in a Manufacturing Plant", Lecture Notes in Artificial Intelligence 2080. Springer Verlag, New York, pp 597-610, Apr., 2001.
- [16] T. W. Malone, K. Crowston, and G. A. Herman, "Organizing Business Knowledge: The MIT Process Handbook", MIT press, Sep., 2003.
- [17] 石川達也, 小林圭堂, 岡部雅夫, 山口高平, "オントロジーとルールベースシステムを利用した知識変換支援", 人工知能学会 第24回全国大会, 2010年6月.
- [18] 岡部雅夫, "オントロジーによる技術・技能の組織的蓄積のサポートについて", 情報システム学会 第2回全国大会・研究発表大会, 2006年12月情報システム学会.
- [19] 岩間貴史, 立花浩, 山崎浩志, 岡部雅夫, 黒川利明, 小林圭堂, 加藤美穂, 吉岡亜紀子, 山口高平, "業務知識の組織的蓄積・活用を支援するオントロジーに対する一考察", 情報システム学会 第3回全国大会・研究発表大会, 2007年12月.
- [20] 岡部雅夫, 小林圭堂, 石川達也, 山口高平, "業務知識の組織的継承へのオントロジーおよび知識モデリングの活用", 情報システム学会 第5回全国大会・研究発表大会, 2009年12月.

補遺 1

筆者らの先行論文[5]を本論文で必要となる範囲で概説する。

1 概要

先行論文[5]の全体像を図1に示す。ここで、熟練者の暗黙知であった知的熟練を表出化・連結したものが、ドメイン・オントロジーとルール・オントロジーである。ドメイン・オントロジーは、関連研究[14, 15]と同様、対象業務ドメインを構成する概念の体系化およびそれに対する語彙の標準化からなり、知的熟練の表出化において、その記述の標準化をはかり、属人性を排除するためのものである。

一方、知的熟練は、表出化し整理することにより、比較的小さな粒度の自然言語で記述される業務ルールに分割される。ルール・オントロジーは、それらの業務ルールをプリミティブとし、新人がはば広く深い理解ができるよう体系化するため、「正当性関係」(justify)等の意味的關係を導入したものである。正当性関係とは、一方のルールが他方のルールの正当性を説明している関係であり、相対的に、正当性を説明するルールを「深いルール」、正当性が説明されるルールを「浅いルール」と呼ぶ。「最も深いルール」は対象業務に対する当該事業所としての業務目標を表

すものとなる、「最も浅いルール」は対象業務を遂行する上で直接使われるルールとなる。

業務ルールを体系化するための意味的關係としては、その他に、あるルールが別のルールの詳細化となっている「特殊化関係」(specialize)、特殊化であり、さらに、元のルールを一部変更する「例外関係」(override)、あるルールの適用には別のルールの適用が前提となる「依存関係」(depend on)が導入される。

ドメイン・オントロジー、ルール・オントロジーにより、初心者にも理解しやすい形で知的熟練が表出化・連結化されたとしても、初心者がそれを実務で適切に使用できるまでに内面化するには、できるだけ実務に近い形での適用経験が必要である。そのためにルールベース・システムを活用する。初心者は自ら実行した業務結果とルールベース・システムのアウトプットを試行錯誤的に比較することを通じて、獲得した知的熟練の十分な内面化が可能になる。また、ルールベース・システムは、熟練者が自らの知的熟練をオントロジーに表出化・連結化することの支援にも活用される。OJTによって知的熟練を獲得してきた熟練者にとっては、それを表出化することは容易ではない。とりわけ、熟練者がルール・オントロジーにおけるプリミティブとしての業務ルールをもれなく認識することは容易ではない。彼らは、状況に応じて様々な業務ルールを使い分けているが、それを明示的に意識していないからである。そこで、熟練者はルールベース・システムを利用し、ルールベース・システムのアウトプットと自ら実行した業務結果とを比較することにより、業務ルールの抜け・不完全さ等をチェックする。さらに、このルールベース・システムは、熟練者に代わって業務を半自動で実行することにより、熟練者が自らの暗黙知をルール・オントロジーとして表出化することの動機付けとなる。

2 東京電力における試行

このような「場」を提供するシステムとして、筆者らは、GEN (General knowlEdge

Navigator)と呼ばれるナレッジ・マネジメントに特化したオントロジー・リポジトリの実験システムを開発し、東京電力における知的熟練を要するある典型的な業務に試験適用した。対象業務は水力発電関係の「停止調整業務」と呼ばれる現場業務である。東京電力における水力発電は既に高度に自動化・統合化されていて、水系毎にある総合制御所がその水系にあるすべての水力発電所を遠隔無人運転するとともに、保全も担っている。「停止調整業務」とは、総合制御所が管理するすべての水力発電所設備の停止を要する点検・修理の主として年間スケジュール（以下、停止スケジュール）を作成する業務である。スケジュールは、以下のような様々な制約の下に、いつ水と呼ばれる発電に利用されない河川水量を最小にするように作成される。

- 機器毎に定められている点検周期
- 需要に見合った発電量の確保
- 農業協同組合等の流域地域団体との協定
- 作業量の平準化
- 天候・気象要因
- 構造上発生する機器間の停止依存関係等

具体的な制約の中身は様々であり、点検周期を除くほとんどのものが、長年の経験により暗黙知として形成されてきたもので、代々、OJTにより引き継がれてきている。また、すべての制約が絶対という訳ではなく、状況によっては、実行可能なスケジュールを作成するために一部の制約が緩和されるが、それは長年の経験に基づいている。以上のように、この業務は、様々な状況・制約を幅広く考慮する必要があり「知的熟練」を要すると共に、その「知的熟練」が表出化されていない典型的な業務である。

試験適用にあたって、筆者らは当該業務における初心者として熟練者にインタビューを行い、熟練者と一体となって GEN 上にオントロジーを構築した。ドメイン・オントロジーは主として設備および停止作業に関するものであり、ルール・オントロジーは停止スケ

ジュールを作成する上での業務ルールからなる。

表 3 実験結果

被験者	A, B	C, D	E	
知的熟練獲得方法	GENIによる独習	専門家による指導	体系化された知識	
平均学習時間(時間)	47	199	138	
平均正答率(%)	88	91	96	
平均誤答数(／130作業)	15.5	11.5	5.0	
原因別内訳	記述の欠如	4.0	0.0	0.0
	記述の曖昧さ	4.0	5.0	1.0
	ケアレスミス	7.5	6.5	4.0

筆者らが熟練者からヒアリングした内容を整理した結果、その内容は134個の単独で意味をなし得るレベルの単純な業務ルールに分解された。134個の内、90個が停止作業のスケジューリングに直接使われる「最も浅いルール」であり、また、当該事業所の業務目標と直接対応する最も深いルールは、「いっ水電力を少なくする」「地域社会と共生する」「電力供給信頼度を維持する」等、12個になった。さらに、熟練者による表出化、新人の内面化を支援するためのルールベース・システムとして、半自動的に停止スケジューリング・システムを構築した。既に述べたように、必ずしもすべてのルールを満たす解が存在する訳ではなく、状況に応じて長年の経験に基づきルールを緩和する必要があるが、それぞれの状況に対して緩和ルールを決定することは組み合わせの数が膨大になり現実的ではないため、各ルールに優先度を与え、解が存在しない場合は、優先度の低いものから緩和することとした。様々なケースに対し、熟練者が適切と考える解が得られるよう、試行錯誤により各ルールに優先度を与えた結果、15段階の優先度によりほとんどの場合に熟練者が適切と考える解が得られるようになった。

3 試行に対する評価

3.1 新人による内面化

本プロジェクトのメンバー5名を被験者として、実際に停止スケジューリング(2年分、計130作業)を作成する評価実験を行った。作成する停止スケジューリングは、予め、熟練者により停止ス

ケジュールを作成済みのものであるが、可能なスケジュールは必ずしも一通りとは限らないため、正解／不正解は熟練者により個別に判断した。

実験結果の概略は表3の通りである。被験者A, Bは業務知識の無い状態で本プロジェクトに最終段階から参加したメンバーでGENおよびスケジューリング・システムを利用し独習した。これに対し、被験者C, Dは本プロジェクト初期からのメンバーで、熟練者からの直接的指導により業務知識を獲得した。被験者Eはプロジェクトに途中から参加し、ある程度体系化された知識をもとに知識を獲得した。被験者A, Bは、被験者C, Dと比較して、ほぼ1/4の学習時間でほぼ同等の正答率をあげ、新人に対する知的熟練の内面化支援において、本提案が一定の効果を上げることが確認された。

ただし、GENによる独習を基本にした被験者A, Bは、他の被験者には見られない誤答として、「記述の欠如」による誤答が平均4.0件あった。その原因の一つに手順情報の不足があった。既に述べたように、本問題は必ずしも業務ルールすべてを満たす解が存在せず、状況に応じて一部の業務ルールの緩和が必要となる。熟練者の場合は、「何となく」概ね業務ルールを満たす停止スケジューリングを作成した後、業務ルールに抵触する部分を揺すって、できる限り業務ルールを満たす解を導くことが比較的容易にできるが、これは「何となく」作成する初期の停止スケジューリングが手戻りの小さいものとなっているためである。ところが、被験者A, Bはこのような初期の停止スケジューリングを与えることができず、非常に大きな手戻りが発生する場合があり、結果として緩和すべきでない業務ルールを緩和してしまう場合があった。被験者C, D, Eにはこのような誤りがなかったが、それは、被験者C, D, Eは手戻りが少なくなるよう全体への影響の大きな作業を優先的にスケジューリング化することを、プロジェクトでの経験を通じて学んでいたことによる。

3.2 熟練者による表出化・連結化

一方、熟練者による知的熟練の表出化・連結化に関しても、プロジェクト・メンバーとの共同作業により実際に構築されたオントロジーやルールベース・システムが新人の知的熟練の内面化に有効に寄与したという意味において、本提案の有効性が確認された。スケジューリング・システムは熟練者からの評価が高く、熟練者による知的熟練のオントロジーへの表出化においても、スケジューリング・システムは重要な役割を果たした。具体的には、祝日が土曜日となった場合の扱い等、単純ではあるが、熟練者といえども、ミスや適用漏れを起こしやすいルールがあるが、スケジューリング・システムはこれらのルールを確実に適用してくれることに対する熟練者の評価が高かった。これらのルールは、熟練者には自明であり意識されないが、初心者には重要なルールであり、取りも直さず、熟練者が表出化を漏らしがちなルールであった。

スケジューリング・システムによる試行錯誤は、必要なルールを漏れなく洗い出すことに貢献しただけでなく、状況に応じた一部ルールの緩和という長年の経験に基づいた暗黙知をルールの優先度という形で表出化することを可能にした。

一方で、今回は初期構築であったため、熟練者は本プロジェクト・メンバーと一体となってオントロジーやルールベース・システムを構築したが、これからは熟練者自らが持続的に状況の変化に応じて更新していけるかどうかは確認できなかった。

補遺 2

条件部と結論部の解釈に基づくルールの分類の正確な定義を記す。記述は、条件部に規定される作業集合をソートとして扱い、通常の述語論理の記法によった。なお、本定義はプロジェクト・メンバー内で正確な定義を共有するために利用したものであり、エンドユーザが必要とするものではない。

1 単独作業ルール

作業を表す変数を x 、条件部で規定される作業集合を S 、結論部で規定される制約を表す単項述語を C として、以下で表されるルール。

$$\forall x / S \quad C(x)$$

2 相互依存作業ルール

2.1 対称ルール

以下の i) ii) のいずれかで表されるルール。
i) 条件部で規定される作業集合が単一の場合

作業を表す変数を x, y 、条件部で規定される作業集合を S 、結論部で規定される制約を表す 2 項述語を C として、

$$\forall x / S \quad \forall y / S \quad (\text{if not } (x=y) \text{ then } C(x, y))$$

ii) 条件部で規定される作業集合が複数の場合

作業を表す変数を x, y 、条件部で規定される作業集合を S_i 、結論部で規定される制約を表す 2 項述語を C として、

$$\forall i / I (\forall x / S_i \quad \forall y / S_i \quad (\text{if not } (x=y) \text{ then } C(x, y)))$$

ただし、作業集合 S_i 合の添え字 i は月等を表すもので、そのソートを I (i が月の場合であれば、月を表す添え字の集合、i.e. 1~12) とする。また、作業集合の添え字が i, j 等複数になることも認める。

2.2 非対称ルール

以下の i) ii) のいずれかで表されるルール。
i) 条件部で規定される作業集合の組が一つの場合

作業を表す変数を x, y 、条件部で規定される作業集合を T, U 、結論部で規定される制約を表す 2 項述語を C として、

$$\forall x / T \quad \forall y / U \quad (\text{if not } (x=y) \text{ then } C(x, y))$$

ただし、 T と U は同一の集合であってはならない。

ii) 条件部で規定される作業集合の組が複数の場合

作業を表す変数を x, y 、条件部で規定され

る作業集合を T_i, U_i , 結論部で規定される制約を表す 2 項述語を C として,

$$\forall i / I (\forall x / T_i \forall y / U_i \text{ (if not } (x=y) \text{ then } C(x, y)))$$

ただし, 作業集合 T_i, U_i の添え字 i は月等を表すもので, そのソートを I (i が月の場合であれば, 月を表す添え字の集合, i.e. 1~12) とする。また, 作業集合の添え字が i, j 等複数になることも認める。

また, T_i と U_i は同一の集合であってはならない。

補足

対称ルールにおいては, その定義より, 制約を表す述語 C は対称 (i.e. $\forall x / S_i \forall y / S_i C(x, y) = C(y, x)$ (真理値として)) となる。

一方, 非対称ルールにおいては, 制約を表す述語 C に対して非対称性 (i.e. $\exists x / T_i \exists y / U_i \text{ not } (C(x, y) = C(y, x))$ (真理値として)) は要求しない。

従って, 対称ルールと非対称ルールは必ずしも排他関係にはなく,

例えば, 対称ルール

$$\forall x / S \forall y / S \text{ (if not } (x=y) \text{ then } C(x, y))$$

ないし

$$\forall i / I (\forall x / S_i \forall y / S_i \text{ (if not } (x=y) \text{ then } C(x, y)))$$

において, 作業集合が $S=\{a, b\}$ ないし $S_i=\{a_i, b_i\}$ と 2 つの要素からなる場合, $T=\{a\}, U=\{b\}$ ないし $T_i=\{a_i\}, U_i=\{b_i\}$ とすることにより,

非対称ルールとして

$$\forall x / T \forall y / U \text{ (if not } (x=y) \text{ then } C(x, y))$$

ないし

$$\forall i / I (\forall x / T_i \forall y / U_i \text{ (if not } (x=y) \text{ then } C(x, y)))$$

と記述もできる。

このことは, 実際, 対称ルールとして対象作業が各項の or 指定の and 解釈の単一作業集合として指定できない場合 (例えば, A 発電所の B 作業と C 発電所の D 作業 等) に, 非対称ルールとして規定するために用いられている。

著者略歴

[1] 岡部雅夫

1987 年 スタンフォード大学文理科学研究科

統計学専攻修士課程, 工学研究科オペレーションズ・リサーチ専攻修士課程修了。慶應義塾大学理工学研究科開放環境科学専攻後期博士課程在学中。東京電力株式会社システム企画部スペシャリスト (ソフトウェア開発技法)。当学会, 賛助会員 (東京電力株式会社)。

[2] 小林圭堂

2008 年 慶應義塾大学理工学部管理工学科卒業。2010 年 慶應義塾大学理工学研究科開放環境科学専攻修士課程終了。現在, 東芝ソリューション株式会社勤務。

[3] 石川達也

2010 年 慶應義塾大学理工学部管理工学科卒業。現在, 慶應義塾大学理工学研究科開放環境科学専攻修士課程在学中。

[4] 飯島正

1991 年 慶應義塾大学大学院理工学研究科博士課程単位所得退学。博士 (工学)。(株) 東芝を経て, 慶應義塾大学理工学部管理工学科・専任講師。ソフトウェア工学分野を中心に, 広く知識の表現・獲得・形成・共有・利用並びに管理等に関心を持っている。当学会, 理事。

[5] 山口高平

1979 年大阪大学工学部通信工学科卒業, 1984 年同大学院工学研究科博士後期課程修了。同年, 大阪大学産業科学研究所助手。1989 年静岡大学工学部助教授。1997 年同大学情報学部教授。2004 年より慶應義塾大学理工学部教授。定理証明の研究を経て, 知識システム, データマイニング, セマンティック Web, オントロジー, 知能ソフトウェア工学に関する研究に従事。2007 年度大川出版賞。当学会理事, 編集委員長。人工知能学会副会長。電子情報通信学会, 情報処理学会, AAI, IEEE-CS 等の会員。