

## RDF データベースにおける視点の変化に対応した知識の構造化

大槻 明<sup>i</sup>, 岡田謙一<sup>i</sup>  
<sup>i</sup> 慶應義塾大学大学院理工学研究科

組織における知識情報は、組織活動に比して膨大化、細分化及び複雑化の一途をたどり、その全体像の把握は容易ではなくなる。しかし、組織内の解決されるべき数多の課題に対応するためには、組織内の知識情報を縦断に役立てる合理的・体系的な手法が必要である。その手法の一つとして知識の構造化が重要視されている。知識の構造化とは、要素と要素の関係性を明らかにすることであり、本論文では、非巡回有向グラフを対象として、RDF スキーマを応用し、要素間の関係性を参照及び継承情報によって明確化することにより視点を変えて表現できる知識の構造化手法について提案する。

### Structurizing of knowledge corresponding to change in aspect in RDF data base

Akira Otsuki<sup>i</sup>, Kenichi Okada<sup>i</sup>  
<sup>i</sup> Science and Engineering, Keio University

The knowledge information in the organization keeps being made hugely, being subdivided and being complicated with the organizational activity, and it becoming not easy to understand the whole image. However, you have to deal with a lot of problems in the organization. For that, a systematic technique for using the knowledge information in the organization is necessary. Therefore, structurizing knowledge is important as this technique. The structurizing knowledge clarified the relation between the elements. In this study, it proposes the structurizing technique of the knowledge that can correspond to the change in the aspect by applying the RDF schema, and clarifying the relations between elements.

#### 1 はじめに

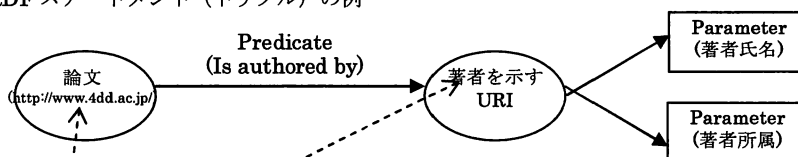
近年、情報共有技術の発達により大量のメタデータが生成・蓄積されるようになってきたため、メタデータによる情報の管理が重要性を増している。メタデータは様々な形式で記述されるが、RDF に代表されるグラフ構造はその代表的なフォーマットの一つである[1-5]。本論文では、ノード（頂点）間を結ぶすべてのエッジ（辺）に片側矢印または両側矢印をもつ矢線グラフである非巡回有向グラフを対象として、RDF スキーマを応用し情報間の関係性や継承関係を表現することにより視点の変化に対応できる知識の構造化手法について提案する。

#### 2 先行・関連研究

##### 2.1 RDFについて

RDF (Resource Description Framework) とは、ウェブ上にある「リソース」を記述するための統一された枠組みであり、W3C により規格化がなされており、特に、メタデータについて記述することを目的としたものである。また、RDF の構造は、主語 (subject) 述語 (predicate) 目的語 (object) の3つの要素でリソースに関する関係情報を表現するもので、このトリプルはグラフ理論におけるグラフで表現するものである<図1>。この構造を利用することにより、情報間の関係性や継承性を表現することが可能になる。

<図1> RDF ステートメント (トリプル) の例



RDF が定義しているのは、このトリプルに基づく抽象構文であり、具象構文としては XML を利用した RDF/XML が別に定義されている。

RDF における主語は、URI で示されたリソースか、URI を持たず直接参照できない空白ノードのどちらかである。述語は URI で示される。目的語は Unicode の文字列か (URI で示された) リ

ソースか空白ノードのいずれかである[2].

## 2.2 先行・関連研究

W3C, HP, MIT の合同プロジェクトである SIMILE (Semantic Interoperability of Metadata and Information in unLike Environments) が発表した Welkin[6]は, RDF 可視化ツールであり, RDF/XML ファイルを読み込ませると, ノードとアークをグラフィカルに表示することができる. しかし, 本ツールは URI データをビジュアルライズに表示する機能にのみとどまっており, 利用者が構築するメタデータを満足に表現するには至らない. さらに, Leo Sauermann の RDF Gravity (RDF Graph Visualization Tool)[7]は, Welkin と同様, RDF のノードとアークをグラフィカルに表示してくれるツールであるが Welkin よりも完成度が高い. Welkin と同様に, ノードをタイプごとに異なるシンボルで表現したり, 自由に動かしたりスクランブルできるだけでなく, アークのプロパティラベルを ON/OFF できる. さらに, RDQL を使ってトリプルを絞り込むと, そのパターンにマッチしたグラフだけが表示される機能がある. しかし, 本ツールもメタデータをランダムに関連づけた知識マップの表現に留まっており, 視点の変化を柔軟に表現するには至らない.

また, これまでにいくつかの知識の構造化に関するツールが提案されているが[8-11], 何れも, 知識マップを形成し, 個人及びグループの作業支援を目的とするものが多い. 日立システムアンドサービスが発表している知のコンシェルジュ[8]は, Mindjet Corporation 社の思考マップ作成ツール「Mind Manager」を応用し, 作成した複数のマップを重ね合わせた知識を活用する機能を強化したものであり, 個人で持っていた業務上の知識を組織で共有して活用するツールである. アッター・ソフトウェア社が発表している XpertRule[9]は, 知識工学をベースとしたナレッジマネジメントシステムである. 決定木により知識表現を行い, 例外木により知識を獲得する. これらの木構造にて知識の構造化を図り課題解決のための知識を管理する. 友部らは[10], データベースとして蓄積された議事録集合を構造化し, 関連する議事録集合を統合することで議事録マップを生成し, 閲覧者の知識発見を支援するシステムについて述べている. 何れの知識マップも, 知識を構造化するに留り, 構造化された各個別の知識間の関連性を表現するまでには至っていない. また, 構造知識化研究所が提唱する, SKY (Structured Knowledge Yielding, 構造化知識生産) 手法[11]は, 組織活動において必要とされる知識を再利用する目的とその目的のために表現される知識の再利用上の性質 (分節性, 一般性, 関係性) を定義し, その目的並びに性質に基づいて知識を構造化する. しかし, ここでいう関係性は, 「原因→結果」の因果関係のみで構成されているが, 本発表では, HasPertOf, AttributeOf, KindOf, Reference, UseOf, Equal, Positive, Negative, Creative などの関連性や継承性を表現する要素を新設しており, さらなる拡張性を目指している. さらに, 砂川らは概念間の依存性に基づくロール概念について提唱しており[12], この概念は本発表における継承関係の表現に類似する点も多いが, 砂川らのロール概念は, 概念のみの発表に留まっており, しかるべき検証や実証がされているわけではなくシステムとして具現化されているわけでもない. また, ロール概念は, 親子表現によるツリー構造であるのに対し, 本発表は, 利用者の視点に基づき知識マップを表現することに主眼を置いており, 親子表現にもツリー構造にも特化するものではない.

## 3 研究目的及び提案手法

### 3.1 研究目的

知識の構造化とは, 要素と要素の関係性を明らかにすることであるが, 本研究は, 非営利団体という組織を想定し, 議事録, 関連資料及び財務関係資料など, 組織を運営する上で必須の各資料を要素として用い, また, RDF (Resource Description Framework) における RDF スキーマを応用し, その要素間に継承関係を表現する情報を持たせてその関係性や継承性を明らかにし, さらに, 利用者の視点の変化に対応した知識の構造化及びその表現手法である知識マップについて提案することを目的とする.

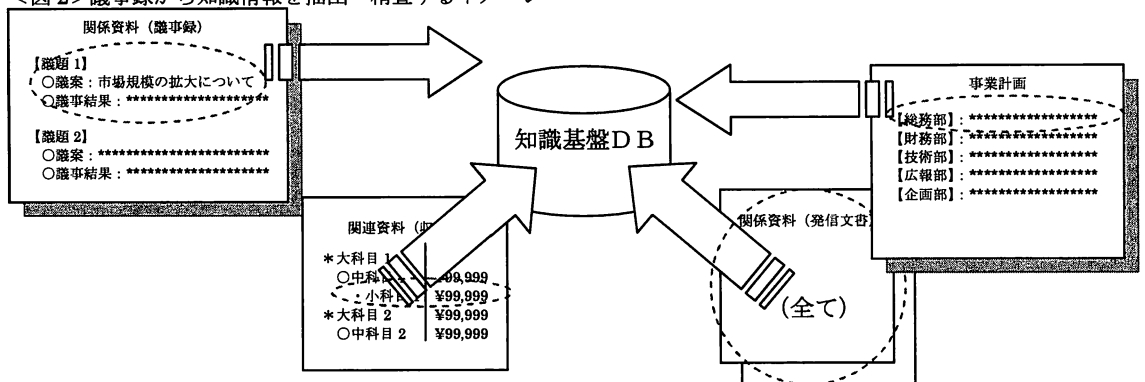
### 3.2 提案手法

#### 3.2.1 メタデータの記述・格納

本発表では, メタデータとして格納する情報として, 組織における事業計画, 資料, 議事録, 受発信文書及び財務関係書類を想定し, 図 2 にそれらの格納イメージを示す. ただし, 関連資料や発信文書は 1 ファイル単位 (つまり, 1 ファイル全て) で格納する場合もありえることとする. 例えば, 議事録や財務情報などの当該部分の情報のみを抽出してメタデータとして格納するものについては, 当該箇所を抽出し後述する RD (Relational Division) ナンバーを付して格納する. なお, 本発表では, 格納するメタデータの単位をノードと表現する. また, 関連資料や発信文書などの 1 ファイルごとに格納するようなものについては, 1 ファイル単位で RD (Relational Division) ナ

ンバーを付して格納する。

＜図 2＞ 議事録から知識情報を抽出・精査するイメージ



### 3.2.2 関係性や継承関係の表現

本発表では、ノード間における関係性や継承関係性を表現するために、RD (Relational division) ナンバーを設けた。＜表 1＞にその RD 要素一覧をまとめる。大きくは 3つの要素から構成されており、subClassOf は、ノード間の親子関係性を表現し、SubPropertyOf はノードのプロパティ要素 (親子関係の詳細など) を表現し、そして、reletionOf はノード間関係性や継承性を表現する。この RD は、RDF におけるクラスの継承 (rdfs : subClassOf) プロパティの継承 (rdfs : subPropertyOf) に相当し、要素情報をクラス (rdfs:Class)、要素情報のプロパティを (rdf:Property) と定義することにより、その継承関係を表現することに応用した。

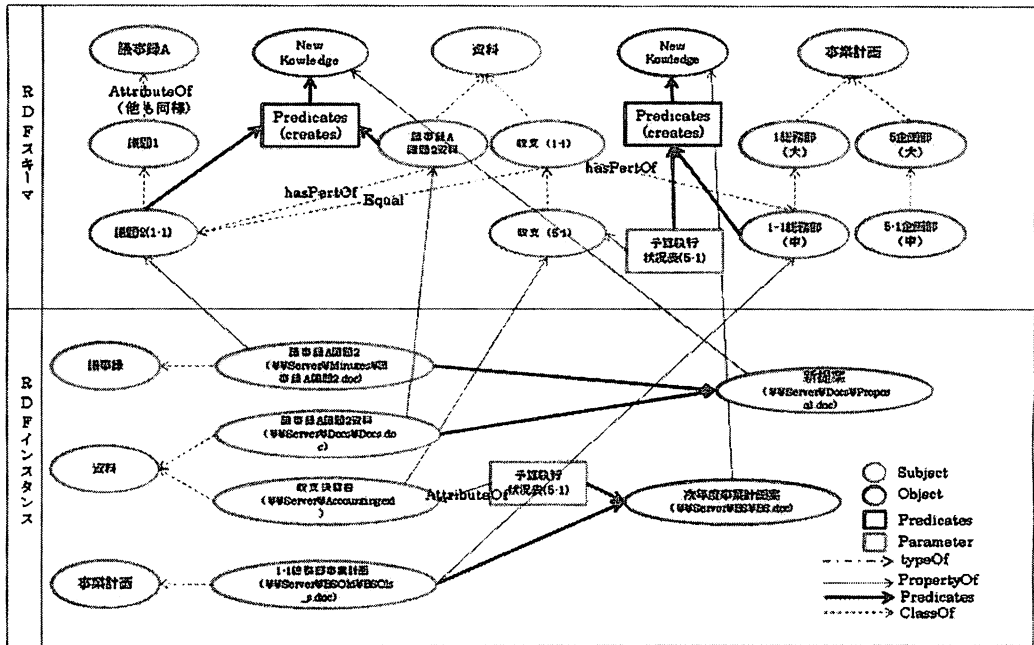
＜表 1＞ RD による継承関係の表現

項	RD 例	内容
1	subClassOf	
	(1) SuperClass	上位ノードであることを意味する。
	(2) SubClass	下位ノードであることを意味する。
2	subPropertyOf	
	(1) HasPartOf	部分的に持っている機能であることを意味する。 (例：車ならエンジン)
	(2) AttributeOf	属性であることを意味する。(例：重さ)
	(3) KindOf	一種であることを意味する。
3	reletionOf	
	(1) Create	複数のノードにより創造したことを意味する。
	(2) ReferenceOf	参照関係を意味する。
	(3) UseOf	使用したことを意味する。
	(4) Equal	同輩であることを意味する。
	(5) Positive	成功関係であることを意味する。
	(6) Negative	失敗関係であることを意味する。

### 3.2.3 RDF スキーマ及びインスタンス

全項までの内容から、RDF スキーマ及びインスタンスについて考察する。＜図 3＞に RDF スキーマ及びインスタンスの例を示す。図の上部は RDF Schema を用いて定義した RDF スキーマであり、図の下部は、RDF スキーマ中で定義されたクラスなどのインスタンスである資源とそれらの関係性を示した RDF データである。

＜図 3＞ RDF スキーマ及び RDF インスタンスの例



### 3.2.4 構造化の視点

本構造化の基軸となる視点として、次の3つの視点を例示する。このような視点を変える毎に新たな知識の構造化が構築される。

#### a) 構成要素（ノード）単位の視点

複数の要素がリレーションを持ち、新たな要素を創出する場合に、元要素（ノード）を単位とする視点である。つまり、類似要素からどのような要素が導き出されたのか、といった情報抽出の視点を想定する。この具体的な一例として、年度途中及び事業計画遂行後における事業分析を対象とし、セグメンテーションによる事業分析を想定する。つまり、事業を戦略分析に適した単位に分けて分析するイメージである。

#### b) 構成要素から導き出された要素（ノード）単位の視点

リレーションを持った複数の要素から創出される新たな要素を単位とする視点である。つまり、ある要素を導出する類似過程や元要素を抽出するような視点を想定する。この具体的な一例として、企画会議や新商品開発会議における議題をキーとして、関係する資料や過去の議事録から類似する要素を抽出し、その関係資料等の情報を抽出する。この概念を応用すると、例えば、科学の分野において、部分的な化学反応単位での関係情報が抽出でき、このケースでは無害な化学反応が発生するが、このケースでは有害な化学反応が発生するなど、参考となるべき関連情報を文献等から抽出できる仕組みへの応用が期待できる。

#### c) トリプル単位の視点

類似するリレーションを持った複数の要素のトリプルをキーとして類似情報を抽出する視点である。これは、上記 b)と同様であるが、抽出する対象がノード単位ではなくトリプル単位となる。

## 4 システム構成

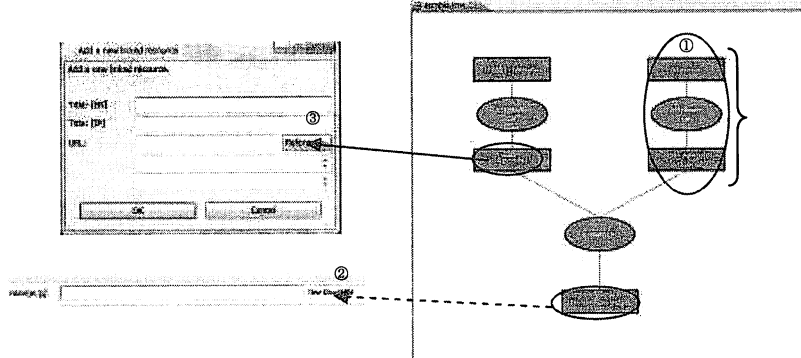
### 4.1 システム概要

本システムは、3つの機能から構成される。一つ目は、メタデータの登録機能であるが、これはエディタを通してノード単位及びトリプル単位でメタデータを登録する。二つ目は、上記登録機能により登録したメタデータを検索する機能であり、同様にノード単位及びトリプル単位で検索できる。三つ目は、検索されたメタデータを近くマップにて表示する機能である。この近くマップを上述の視点により切り替えることができる。メタデータの記述及び検索に用いるクライアントシステムはJava6.0及びeclipse3.3で構築し、データベースはMySQL5.0.51aにて構築し、検索処理は形態素解析に対応しているApache Lucene Java 2.3.1を採用して構築した。

## 4.2 メタデータの記述

<図 4>はメタデータとして、「概念ノード」+「関連ノード」+「概念ノード」のトリプルを2つ（「Agenda1」+「AttributeOF」+「H2OS1\_Minute」と「H2OS1\_Agenda1\_Material」+「AttributeOF」+「Material」）と、その2つのトリプルから新たなトリプル（「Agenda1」and「H2OS1\_Agenda1\_Material」+「CreateOF」+「New Knowledge」）を記述した例である。トリプルは、パレットから選択する形でエディタに張り付けたり ①、各ノードの名前を記述したり ②、各ノードのリソース（ファイルなど）をリンクしたり ③ することができる。また、このように記述された RDF ファイルの出力例を<図 5>に示す。

<図 4>メタデータ記述例



<図 5>トリプル RDF ファイルの出力例

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:xsd="http://www.w3.org/2000/10/XMLSchema#"
>
  <rdfs:Class rdf:ID="WholnessConcept">
    <rdfs:label>WholnessConcept</rdfs:label>
  </rdfs:Class>
  <rdfs:Class rdf:ID="RelationalConcept">
    <rdfs:label>RelationalConcept</rdfs:label>
  </rdfs:Class>
  <rdfs:Property rdf:ID="hasPart">
    <rdfs:label>hasPart</rdfs:label>
  </rdfs:Property>
  <rdfs:Property rdf:ID="hasAttribute">
    <rdfs:label>hasPart</rdfs:label>
  </rdfs:Property>
  <rdfs:Class rdf:ID="Material">
    <rdfs:label>Material</rdfs:label>
    <rdfs:subClassOf rdf:resource="#WholnessConcept" />
  </rdfs:Class>
  <rdfs:Class rdf:ID="H2OS1_Agenda1_Material">
    <rdfs:label>H2OS1_Agenda1_Material</rdfs:label>
    <rdfs:subClassOf rdf:resource="#Material" />
  </rdfs:Class>
  <rdfs:Class rdf:ID="H2OS1_Minute">
    <rdfs:label>H2OS1_Minute</rdfs:label>
    <rdfs:subClassOf rdf:resource="#WholnessConcept" />
  </rdfs:Class>
  <rdfs:Class rdf:ID="Agenda1">
    <rdfs:label>Agenda1</rdfs:label>
    <rdfs:subClassOf rdf:resource="#H2OS1_Minute" />
  </rdfs:Class>
  <rdfs:Class rdf:ID="New Knowledge">
    <rdfs:label>New Knowledge</rdfs:label>
    <rdfs:createOf rdf:resource="#H2OS1_Agenda1_Material" />
    <rdfs:createOf rdf:resource="#Agenda1" />
  </rdfs:Class>
</rdf:RDF>
```

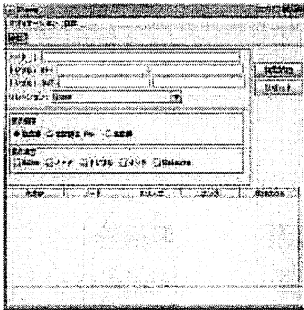
Name...Class

Name...subClassOf

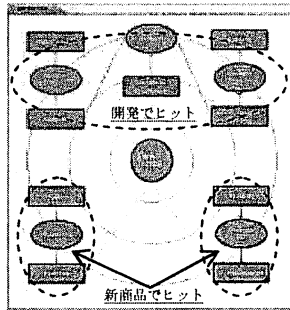
## 4.3 視点の変化に対応した知識の構造化

検索画面イメージを<図 6>に示す。この利用例としては、<図 4>のエディタにて新知識を創造時に、当該ノードやトリプルの類似知識を検索するときにあげられる。また、<図 4>のエディタを使用せず、この検索画面のみで検索することも可能である。<図 7>は「新商品の開発（上記 3.2.4 の b）に相当」というノードをキーとして検索した結果であり、「新商品」と「開発」でそれぞれヒットした知識が表示されている。また、<図 8>は「PartsA」と「PartsB」から「New Item」を導出する過

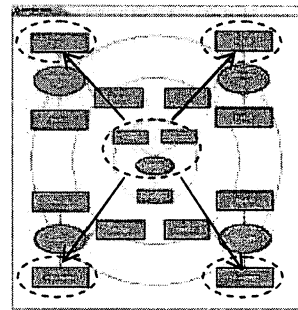
程（手法）をキーとしており、つまりは、上記3.2.4のc)に相当するトリプルをキーとして類似手法を検索し、その類似手法が導出する新知識について表示している。このように、キーとなる事項を視点と位置付け、その視点を切り替えることによって違った結果（知識）を検索・表示することができる。



<図 6>



<図 7>



<図 8>

## 5 今後の課題

今後の課題としては、企業など実際の現場を想定して本システムの有効性・実行性検証を行うことであり、以下にその一例（概要）を示す。

- **速度的優位性の検証**  
例えば、本システムを利用した場合としなかった場合における課題解決までの時間比較など。
- **有効性検証**  
利用（実証）者へのアンケートによる有効性検証など。
- **定量的検証**  
例えば、本システムを利用した場合としなかった場合における知識排出数の比較検証など。
- **類似ツールとの比較検証**  
上述の各検証について、類似ツールと比較検証する。

## 6 おわりに

今回は、利用者の視点の変化に対応する知識の構造化及びそれを表現する知識マップの構築について、RDFスキーマを応用する形で試みた。組織活動を行ううえでは、散在する現状や現場、さらにはセクションや業種の違いにも柔軟に対応するために、多岐にわたる角度からの視点を取り入れながら考察していく場面が多々存在するため、今後は、本システムを実際の現場に対応すべくさらなる発展を目指していきたいと考える。

### 【参考文献】

- [1] Q.Li and B.Moon. "Indexing and Querying XML Data for Regular Path Expressions" in the VLDB journal, pp.361-370.2001
- [2] "Resource Description Framework(RDF) Schema Specification 1.0" , <http://www.w3.org/TR/2000/CR-rdf-schema-20000327/>, W3C Candidate Recommendation 27 March 2000
- [3] David Beckett, "The Design and Implementation of the Redland RDF Application Framework", WWW10, May 2-5, 2001, Hong Kong
- [4] Brian McBride, "Jena: Implementing the RDF Model and Syntax Specification", In proceedings of the second international workshop on the Semantic Web 2001' Hongkong, China' May 2001
- [5] Jeen Broekstra, Arjohn Kampman, Frank van Harmelen, "Sesame: A Generic Architecture for Storing and Querying RDF and RDF Schema (2002)", In The Semantic Web - ISWC 2002, volume 2342 of Lecture Notes in Computer Science, pages 54-68. Springer, 2002
- [6] SIMILE Project, 「welkin」, <http://simile.mit.edu/welkin/>
- [7] Leo Sauermann, 「RDF Gravity」, <http://semweb.salzburgresearch.at/apps/rdf-gravity/index.html>
- [8] 日立システムアンドサービス, 「知のコンシェルジェ」, <http://www.hitachi-system.co.jp/press/2008/pr080128.html>
- [9] アッター・ソフトウェア, 「XpertRule」, [http://www.xpertrule.com/jpages/jinfig\\_x.htm](http://www.xpertrule.com/jpages/jinfig_x.htm)
- [10] 友部博教ほか, 「ディスカッションマイニング：議事録集合からの知識発見」, 情報処理学会第67回全国大会
- [11] 構造知識化研究所 HP, <http://www.ssm.co.jp/sky04.html>
- [12] 砂川英一ほか, 「コンテキスト依存性に基づくロール概念組織化の枠組み」, 人工知能学会論文誌, Vol.20 No.6 pp.461-472 (2005)