SIG-SWO-051-02

ナレッジグラフの構築レベルの整理

A Level of Knowledge Graph Building for Efficient Requirements Definition

小柳佑介[1*]　西野文人[1]　朱成敏[2]　武田英明[2]
Yusuke KOYANAGI[1]　Fumihito NISHINO[1]　Sungmin JOO[2]　Hideaki TAKEDA[2]

[1] 株式会社 富士通研究所
[1] Fujitsu Laboratories Ltd.
[2] 国立情報学研究所
[2] National Institute of Informatics

Abstract: In the initial stage of system development, the system engineer plans the budget, and period of time, and define the system requirements. In the case of a system that utilizes a knowledge graph, it is necessary to define the requirements of the knowledge graph and estimate the building cost. It is useful for system engineers, in particular, who do not have much experience in the building of knowledge graphs, to understand how much they need to build a knowledge graph for what they want to achieve and how much it costs to build it. However, it is not clear how much the knowledge graph should be constructed in order to realize what it wants to realize, and what requirements the knowledge graph should meet and the building costs for "What you want to achieve". In this paper, as a first step to improve the efficiency of system development utilizing knowledge graphs, we define the building level of knowledge graph and clarify "what it realizes" and "difficulty" for each level of knowledge graph. At first, we investigate the procedure of building a knowledge graph. After defining the level of knowledge graph, "what it realizes" and "difficulty" of the actual knowledge graph is shown.

1　はじめに

システムエンジニアが，顧客に価値を提供するシステムを開発するためには，そのシステムで何ができるかを提示することや，それを実現するためのコストの見積もりが重要である．それは，ナレッジグラフを活用するシステムを開発する場合でも同様であり，ナレッジグラフを使うことで何ができるかを提示することや，それを実現するためのナレッジグラフ構築のコスト見積もりが必要である．また構築コストを下げるためにナレッジグラフ構築において自動化・効率化できる部分の把握も重要である．

ナレッジグラフの構築においてまず重要なことは，一般的なソフトウェア開発と同様，ナレッジグラフの要件定義を明確化することである．また，ナレッジグラフを活用するシステムを開発する場合，最初から何でもできる完璧なナレッジグラフを目指して構築するのはコストが大きくなるため，システムによって「実現したいこと」が実現できる範囲のみでナレッジグラフを構築するというのが現実的である．その場合,「実現したいこと」を実現するための「ナレッジグラフの具体的な要件」に対して，ナレッジグラフをどの範囲まで構築すべきかを判断する必要がある．また，既存のナレッジグラフを活用したシステムを構築する場合でも，既存のナレッジグラフでは何が保証され,「実現したいこと」に対して何が足りないかを判断することで開発計画を立てることができる．また,「実現したいこと」に対して「ナレッジグラフの具体的な要件」を明らかにされ，今後どんな作業が必要で，その作業のコストがどの程度なのかが明らかにされていれば，それらは，顧客の要望をナレッジグラフ構築の設計にブレイクダウンするために有用である．

上記を実現するにあたって，現状では，ナレッジグラフの構築時の要件の落とし方が確立されておらず，また「実現したいこと」に対してナレッジグラフの満たすべき要件が整理されていない．また，それぞれの「実現したいこと」に対してどんなコストがあるかもこれまでに整理されていない．

*連絡先：株式会社富士通研究所
〒 211-8588 川崎市中原区上小田中 4-1-1
E-mail: koyanagi.yusuke@fujitsu.com

本研究では,「実現したいこと」に対して，ナレッジグラフの構築に関わる人が，要件や必要な作業，構築コストを判断できるようにすることを目的として，特にテーブルデータからのナレッジグラフの構築を想定し，その手順における「ナレッジグラフのレベル」を整理し，それぞれのレベルに対して「保証されること」,「要件」,「構築コスト」を整理する．

2　本研究で想定するナレッジグラフ構築

2.1　ナレッジグラフの定義

本研究において想定するナレッジグラフの定義は，以下のものとする．

1. ある実世界の知識をエンティティ間の関連性で定義したものであり，
2. それを機械処理が可能な形式にしたもの．

上記の定義を満たすための表現形式の一つとしてRDFが挙げられる．後述の事例において，RDF形式のものを挙げて説明しているが，本研究において定義する構築レベルは，特にRDFに限定するものではない．

2.2　ナレッジグラフ構築手順の想定

本研究では，複数のテーブルデータからのナレッジグラフの構築を想定する．これを想定する理由としては，筆者がこれまで経験した構築が，既に存在する複数のデータを統合して行うものであったことが挙げられる．また，テーブルデータは構造データの中でも比較的シンプルで，かつ，様々なデータソースで広く用いられる形式である．そのため，特にテーブルデータからの構築を想定することとした．

以下に，本研究で想定する，ナレッジグラフ構築の手順を記載する．

１．調査・設計 ナレッジグラフの要件を満たすナレッジグラフの設計．

２．データ収集 ナレッジグラフの元となるデータの収集．

３．データ変換 収集したデータを設計通りに変換．必要に応じて，正規化処理や同一性判定処理などを実現．

また，上記には記載していないが,「調査・設計」の前には,「ナレッジグラフの要件定義」のステップが存在する．これは，ナレッジグラフを活用するシステムの要求定義をブレイクダウンして，ナレッジグラフに対する要件を定義するステップである．本研究では，このステップを対象外とするため，ナレッジグラフ構築手順から除外した．

3　ナレッジグラフの構築レベル

ナレッジグラフとして何がされているのかという「ナレッジグラフの要件」をレベル分けし，それぞれのレベルで「保証されること」，そのレベルのナレッジグラフを構築する上での「手間・難しさ」，そのナレッジグラフを構築する上で利用できる「ツール」と「技術」について，表1の通り，整理した．整理した内容について，以下でそれぞれ説明する．

3.1　レベル1：構造を考えないRDF，値の正規化・整形

レベル1として想定するナレッジグラフは，単に表形式を単純に形式だけRDFに変換した「構造を考えないRDF」である．これはW3Cのドキュメントcsv2rdf(https://www.w3.org/TR/csv2rdf/)に従った形式である．csv2rdfは，表形式のデータをRDFに変換するときに適用される手順とルールを定義している．このレベルのものは，単に形式を変換しただけのものなので，変換処理は完全に自動化することが可能である．また，単純に形式を変換したものであるため，元データからの情報に忠実であることが保証される．レベル1は，単純に変換したものであるが，実際には値の正規化・整形などが行われる．これをレベル1+とする．

レベル1+は上記の変換の際に，値の正規化・整形を施したものである．値の正規化・整形の前提として，まずは値のタイプ分けがある．値が数値なのか，日付なのか，文字列なのか，あるいは文字列であってもそれが住所なのか郵便番号なのかあるいは特定のコードなのかを把握し，それぞれに合った正規化・整形を行う．正規化・整形の例としては，数字の桁区切りの除去，単位系の統一，異体字の変換，カタカナ・ひらがななどの統一，表記のゆれの吸収，住所表記の統一などがある．これらの処理の中にはやや複雑なものや事情に精通していないと変換できないものもあるが（例えば海外を含めた住所表記の統一など），基本的には比較的単純なツールで対応できるものである．そして，このような処理を行ったナレッジグラフは，正規化されているので，リテラルレベルでの同一性は保証され，簡単な集計処理などが行えるようになる．

表 1: ナレッジグラフの構築レベル

Lv	KGの要件	保証されること	手間・難しさ	ツール /Service	構築 自動化 技術
1	構造を考えないRDF （https://www.w3.org/TR/csv2rdf/）	データの中身に忠実	（完全自動）	（完全自動）	（完全自動）
1+	値の正規化・整形	簡単な集計 (同名インスタンスの識別などは×)	・各列にどんな正規化・整形処理を適用するかの判断が手間 ・複雑な整形は処理の実現が、やや難（例えば住所の分かち書きなど）	OpenRefine	正規化技術
2	ID/URI付与	インスタンス識別した上での分析 他のデータと連携した活用	・同一性判定処理の設計と実現が、データによっては難	Silk	同一性判定技術
3	独自設計のRDFスキーマで 表現されたKG	スキーマの異なるデータを統合した活用 （自分の周りでの活用のみ）	・スキーマを考える手間 ・スキーマにデータをマッピングする手間 ・大きく異なるスキーマの場合、やや難	OpenRefine	スキーマ マッチング 技術
4	標準語彙による表現、標準的なデザインパターンに基づいた表現	標準語彙・標準的なデザインパターンに対して定義されたツール活用 標準語彙で定義されたデータとの連携 標準語彙に対して定義されたルールの活用	・どの語彙で表現するか、その表現が適切か、を判断するのが困難 ・大きく異なるスキーマの場合、やや難 ・複雑なデザインパターンの変換が難	LOV, BioPortal	オントロジ マッピング技術
5	スキーマ・オントロジの制約チェック	意味的制約を充たす データの正確性・完全性が把握できる	・データが制約を満たすかどうかのチェックが難	制約チェックツール	制約チェックツール

レベル 1/1+のナレッジグラフは，表形式のデータのままでも表現できるものであり，他のデータとの連携のための一時的なものであるなどの特殊事情がない限り，特にナレッジグラフ化するメリットはないものである．

3.2 レベル 2：ID/URI 付与

レベル 2 として想定するナレッジグラフは，ナレッジグラフの各エンティティに ID などをベースとした適正な URI を付与したナレッジグラフである．そもそもレベル 1 で想定しているナレッジグラフが RDF 形式ということから，レベル 1 のナレッジグラフでも URI は付与されているので，適正な URI というのがポイントである．すなわち，適当な ID 体系を持った URI を設定したり，適当な LOD のエンティティと owl:sameAs のような述語でリンクすることで，そのエンティティが何を示すかを明示した URI を使用するということを意味している．

このようなレベル 2 のナレッジグラフを構築するには，何を同一とみなすかの設計と，同一性判定処理の実現が必要になってくる．

例えば，法人や自治体あるいは会社内の組織を考えたときに，吸収合併や名称変更，組織変更の前後において，それらは同一のものなのか異なるものなのかを決める必要がある．あるいは，ある部品を考えたときに，仕様が同じなら同一なのか，仕様が同じでもメーカが異なればそれは分けるのか，メーカ内に複数工場があったときは，それらの工場ごとに分けるのか，あるいは個々の部品 1 点ごとに別のものとするのかなどを考える必要がある．これは，何を処理したいのかが明確でないと，何を同一とみなすのが良いのかが定まらない．

このレベルを実現する上での手間・難しさは，正確な同一性判定処理を実現することである．もし，変換前のデータにおいて ID/URI が付与されていて，その ID/URI が設計に合っていれば，手間や難しさは少なくなる．

ナレッジグラフにおいて，何を同一のものとみなすかを明確にして，エンティティに適正な URI を付与することにより，レベル 1+で行った集計処理がより明確で正しいものになる．また，URI の意味づけがはっきりするので，他のデータから本 URI を参照しやすくなる．

3.3 レベル 3：独自設計の RDF スキーマで表現された KG

レベル 3 として想定するナレッジグラフは，独自のものでも構わないので，スキーマ設計をしっかり行い，そのスキーマに基づいて表現されたナレッジグラフである．この後述べるレベル 4 のナレッジグラフとの違いは，標準語彙や標準デザインパターンなどに基づいているか否かである．レベル 4 で想定するナレッジグラフは，RDF の語彙やデザインパターンに精通した人が設計することを想定しているのに対して，このレベル 3 のナレッジグラフは，一般のシステムエンジニア

が設計できるスキーマを想定したものであり，ER 図でデータベースを設計する作業とよく似たものである．

標準語彙・オントロジーや標準的なデザインパターンに基づかないため，再利用性や拡張性に何はあるかもしれないが，外部との連携がなく，当面再利用や拡張が不必要ならば，このレベルのナレッジグラフで十分である．データをナレッジグラフ化するには，スキーマに合わせてデータを変換する必要があるが，これは従来の RDB 等でシステムを構築するときも行われてきたことで，ナレッジグラフであるからといって，特別に困難な作業が増えるわけではない．

3.4 レベル 4：標準語彙による表現

レベル 4 として想定するナレッジグラフは，標準語彙・オントロジーを使い，標準的なデザインパターンを使ったスキーマによるナレッジグラフである．ナレッジグラフのデザインパターンとしては，例えば以下のようなものがあげられる．

- すべてのエンティティには rdf:type でタイプを付与する
- すべてのエンティティには rdfs:label でラベルを与える
- なるべく既存の述語を利用する
- 新たな述語を定義した場合には，その述語の説明，定義域，値域を明示する

あるいは，(s, v, o) の 3 つ組に対して情報を付与したい場合には，Reification や Singleton predicate などの知られたテクニックの中から適切なものを選んで表現するようなこともここに含まれる．

このようなナレッジグラフを構築するには，スキーマ設計において標準語彙やデザインパターンを熟知した人の助けが必要であるが，典型的な事例についてのサンプルやアニュアルを整備することで，ごくわずかな特殊ケースを除いて，一般的なシステムエンジニアでも設計が容易になる．

レベル 3 のナレッジグラフでは独自の語彙を使用しているので，その語彙の意味が記述されていない，あるいはナレッジグラフと別のところに自然言語などによって記述される（独自語彙を使用していてもその意味が明確に記述されているならば，それはもはやレベル 5 ではない）．しかしレベル 4 のナレッジグラフでは，標準語彙や標準的なデザインパターンを利用することで，データの意味がこのナレッジグラフに含まれていることになり，その語彙やデザインパターンを知っている人や機械はこのナレッジグラフを解釈できるということになる．外部に公開するナレッジグラフ，あるいは再利用や拡張を考える場合には，レベル 4 以上のナレッジグラフが望ましい．

また，標準語彙で定義された既存のデータとの連携した活用が可能となる．同じ語彙に対して作成されたクエリを活用することも可能となる．

3.5 レベル 5：スキーマオントロジーの制約チェック

レベル 5 として想定するナレッジグラフは，スキーマ上で意味的な制約を記述するとともに，ナレッジグラフが意味的制約を充たすように作成されており，データの正確性・完全性などもわかり，それらに基づいて高度な推論を可能とするようなナレッジグラフである．

既存のデータからこのようなナレッジグラフを作成するには，データのプロファイリングをとり，例外事項への対処なども必要になり，大きな労力が必要になる．

4 ナレッジグラフの構築レベルの実例

本章では，実際のナレッジグラフに対して，ナレッジグラフの実現したいことと構築レベルを示す．ここでは，実際のナレッジグラフとして，日本の法人 LOD，日本の法令 LOD，DBpedia/DBpedia Japanease を例に，それぞれの構築レベルについて述べる．目的に対して必要な構築レベルを達成しているかどうかで判断するため，各ナレッジグラフのアプリもそれぞれ想定して構築レベルを判断している。

4.1 日本の法人 LOD

日本の法人 LOD[1]は，国税庁公開の法人番号情報を LOD 化したものである．法人番号情報に含まれている，法人番号，法人名，法人所在地，変更履歴情報に加え，DBpedia へのリンク情報や，政府各省庁と企業の調達関係の情報が含まれている．

このデータは,「企業情報分析 Web アプリ」[2]で活用している．このアプリの目的は企業情報分析であり，法人の概要，関連企業情報，法人の吸収合併情報，府省との契約情報や，地域毎に法人に関する統計情報を表示することである．また，LOD4ALL フロントエンドを活用することで，企業間や企業-地域間の遷移を実現している．

[1] http://idea.linkdata.org/idea/idea1s1417i
[2] https://www.fujitsu.com/jp/group/labs/resources/tech/announced-tools/lod4all-frontend/

このデータの構築レベルは，レベル 4 である．アプリの表示対象である法人と地域に URI を付与し，それらの関係を表現するためのスキーマを設計している．法人については，既に組織情報を表現する Core organization ontology（http://www.w3.org/ns/org#）を活用し，地域情報については，独自のオントロジーを作成している．

本 LOD の構築にあたってかかった工数の割合を，表 reftable:ex2 に示す．特に，正規化処理，同一性判定の実現，データ調査に時間がかかっていることが分かる．法人情報には元々法人番号が付与されているので，法人番号情報の URI 化には工数はかかっていない．ただし，DBpedia など各データにおける企業情報へのリンクの際には，名称・所在地による同一性判定を行った上で，同一性を示すリンク情報を生成している．リンク先の情報に法人番号が記載されていればこのコストは削減される．

4.2 日本の法令 LOD

日本の法令 LOD[3]は，電子政府の総合窓口から提供されている日本の法令データを RDF に変換したものである．法令文の解析により，法令間の参照関係のリンクや，用語定義の抽出，文章構造の解析結果も含まれている．また，概要をつかみやすくするために構造化した HTML とも関連付けている．活用目的としては，法令に関する QA チャットボットや，日本の法令 LOD ダッシュボード[4]での構造解析結果の確認がある．

このデータの構築レベルは，レベル 5 である．オープンアノテーションのオントロジーのスキーマに従って作成している．発行年も URI 化の対象に含まれ，元号と西暦の両方で表現されている．また，今回のユーザは日本人を対象にしているため，rdfs:label は日本語のみに留めている．また，法令文の概要表示を実現するにあたって，データ側に概要表示の情報を記載するのではなく，アプリ側でアノテーションに応じた表示方法を定義し，データ側にはアノテーション情報のみを表現するようにしている．

4.3 DBpedia / DBpedia Japanese

開発するシステムにおいて、既に作成されているナレッジグラフを活用するという観点で，良く知られている LOD である DBpedia の構築レベルも確認する．「企業情報分析 Web アプリ」では DBpedia Japanese を活用しており，このアプリで DBpedia と DBpedia Japanese を活用するという観点で見る．

[3]https://github.com/lod4all/e-laws-lod
[4]http://lod4all.net/frontend/index/applicationtop?app_id=lawKGFrontend

このデータの構築レベルは，レベル 4 である．基本的な事柄を示すためには既存のオントロジーを使いつつ，必要に応じて DBpedia 独自オントロジーを設計している．正規化や ID/URI 化は,「企業情報分析 Web アプリ」に対して充分と言えない部分もあった．例えば，主要株主情報や子会社情報は URI ではなくリテラルで示されているため，アプリで企業間の関係を構造化するために，リテラルを URI 化するための処理が必要であった．また，従業員数や資産情報などの時間変化情報は，ある時点での値しか書かれていない上に，その時間情報を同じプロパティの二つのトリプルで示している．そのため、企業間の関係情報としては，レベル 1〜2 である部分もあった．

5 ナレッジグラフ構築コストの見積もり観点

ナレッジグラフを活用するシステムの開発プロジェクトにおいて，ナレッジグラフ構築にかかるコスト・工数を見積もることは重要である．ここでは，ナレッジグラフ構築のコスト・工数の見積りにおいて重要になる要素として，以下の観点を挙げる．

ナレッジグラフの構築レベル： 前節で示したように，ナレッジグラフに何を期待しているのかによって，要求されるナレッジグラフの基本的な構築レベルが異なる．ナレッジグラフの構築レベルが高ければ，それだけコスト・工数も限られるし，また対応できる人材も限られてくる．

ナレッジグラフ（スキーマ）の複雑さ： スキーマにおける基本エンティティタイプの数が一つの指標になる．また，スキーマのグラフの複雑さも関連してくる．

ベースとなるナレッジとその品質： ナレッジグラフ化するもとになるナレッジが存在するのか，存在するとしてそれがどのような形で存在するのか，あるいはその品質の程度はどうなのかなどが問題になる．例えば，ID が付与されていれば，それに基づいて同一性判定を行うことができ構築コストは少なくなる．

ナレッジの量： 機械的に変換するだけであれば，ナレッジの量はそれほど問題にはならないが，量が増えてくると例外事項も増えてくる．また，速度品質への対応や，表示の見やすさへの対応などが必要になるかもしれない．

要求されるナレッジの品質： ナレッジグラフにおいて多少の誤りを許容するのか，完全性を追求するのかなどが問題になる．

表 2: 日本の法人 LOD の構築の工数割合

1．調査・設計	体系知識のクラス定義：		3.5%
	体系知識の関係の定義：		3.5%
2．データ収集	データセットのリスト作り	基軸となるデータセットの選定	5.9%
		スキーマ調査	5.9%
		各項目と体系知識とのマッピング	1.2%
	更新スケジュールの設計とスクリプト化		1.2%
	データ取得		2.4%
3．データ変換	適切なオントロジーの選定		3.5%
	Webからの情報抽出		14.1%
	元データのデータ構造の変換		2.4%
	基軸となるデータセットの変換・正規化		28.2%
	基軸IDとの突き合わせ：同一性判定		28.2%

運用の複雑さ： ナレッジグラフを１回構築するだけで終了なのか，ナレッジグラフ構築後に新たなナレッジが追加されたり，削除されたりするのか．それらにどう対応するのかによって，ナレッジグラフの設計にも大きくかかわってくる．

ナレッジグラフ開発のコスト要因の洗い出しと，それらを定量的に見積もる方法の検討は今後の課題である．

6　既存研究との比較

本節では，ナレッジグラフの構築レベルと関連する既存研究として，Linked Data の品質に関する論文について述べる．

Zaveri らは，2002 年から 2014 年の Linked Data の品質に関する論文を調査し，18 の次元と 69 の指標にまとめ上げている [1]．Farbar らは，DBpedia, Freebase, OpenCyc, Wikidata, YAGO を対象として，品質に関する 34 の指標を挙げている [2]．また，RDFS/OWL 層において，McGurk らは，オントロジーの品質評価に関する論文を調査し，28 の指標を提案している [3]．Mader らは，SKOS による語彙の品質に関わる項目を 15 個挙げている [4]．

また，Dodds らは，Linked Data Patterns を提案している [5]．Linked Data に現れるデータの 56 のパターンを列挙し，5 つのカテゴリに分類している．Linked Data の特徴や性質が各パターンで示されている．

これらの既存研究で述べられる品質や提案されている指標は,「実現したいこと」に対する充分性という観点は入っておらず,「実現したいこと」に対してどこまで構築すべきかを判断するには不十分である．また，手間や難しさなどのコストにつながる要素が入っておらず，システムエンジニアが構築コストの見積もりを判断するには不十分である．

本研究で整理した構築レベルについては，各レベルにおいて「保証されること」を一緒に整理している．それによって,「保証されること」に対して，どの構築レベルまでを実現すればよいかを決める手がかりとすることができる．また，手間・難しさも示しているため，各構築レベルを実現するためにどんなコストが発生するかを把握することができる．本研究によって，システムナレッジグラフを活用するシステムで「実現したいこと」から，構築すべきナレッジグラフのレベルとそのコストを判断するために有用な情報を整理できた．

7　むすび

本研究では，様々なナレッジグラフの定義が存在する中，ナレッジグラフを５つのレベルに階層分けし，それぞれのレベルのナレッジグラフができること，それらのナレッジグラフが保証すること，またそれらのナレッジグラフを構築するためのコストの整理を実施した．

各構築レベルにおいて「保証されること」は整理したが，システムへの要求をナレッジグラフの要件にブレイクダウンする方法についてはまだ定義していない．今後，ナレッジグラフを活用するシステムの「実現したいこと」に対してどのレベルまで構築すべきかを定義することで，システムへの要求をナレッジグラフの要件にブレイクダウンできるようにすることが必要である．

また，ナレッジグラフの構築は，ソフトウェアの開発と共通するところが多い．今後はソフトウェア開発の手法を取り入れながら，工業生産的なナレッジグラフ開発手法を整理していく．

参考文献

[1] Zaveri, A., Rula, A., Maurino, A., Pietrobon, R., Lehmann, J., Auer, S.: Quality assessment for Linked Data: A Survey, *Semantic Web*, Vol. 7, No. 1, pp. 63–93, (2016)

[2] Färber, M., Bartscherer, F., Menne, C., Rettinger, A.: Linked Data Quality of DBpedia, Freebase, OpenCyc, Wikidata, and YAGO, *Semantic Web*, Vol. 9, No. 1, pp. 77–129, (2018)

[3] Mc Gurk, S., Abela, C., Debattista, J.: Towards Ontology Quality Assessment, *4th Workshop on Linked Data Quality at ESWC2017.*

[4] Labra Gayo, J. E., Kontokostas, D., Auer, S.: Multilingual Linked Data Patterns, *Semantic Web*, Vol. 6, No. 4, pp. 77–129, (2015)

[5] Dodds, L., Davis, I.: Linked Data Patterns: A pattern catalogue for modelling, publishing, and consuming Linked Data, http://patterns.dataincubator.org, (2012)