

Linked Open Data への容易なアクセス実現のための視覚的な SPARQL クエリ編集・可視化機構の試作

Torgayev Tamirlan[†] 福田直樹[‡]

[†] 静岡大学情報学部

[‡] 静岡大学大学院情報学領域

1 はじめに

SPARQL は、自由にアクセスできるリンクされたデータである LOD (Linked Open Data) への問い合わせに用いられる。SPARQL は SQL のような問い合わせ言語であり、グラフパターンや推論などの LOD 特有の処理が可能であるが、このような専門知識や経験を有しない非専門家エンドユーザにとってはデータベースへのアクセスが難しいタスクとなる [3]。

本稿では、LOD の作成経験や SQL クエリを使った問い合わせの経験があるが、SPARQL クエリの構成経験がないユーザをターゲットとする。クエリの実行結果、途中経過がリアルタイムに反映されるというサポート機能を持つ視覚的な SPARQL クエリ編集機構を提案する。本研究の目標は、クエリ作成フローのビジュアライズ化、結果の可視化、リアルタイムフィードバックを活用して、非専門家エンドユーザに対するサポートを提供することである。

2 提案手法

図 1 に、提案手法でのビジュアルクエリの一例を示す。この例では、DBpedia と DBLP という二つのデータエンドポイントからデータを取得し、取得する列を SELECT ブロックで限定し、融合する。提案手法ではこのような複数のエンドポイントを使ったクエリは、SPARQL 1.1 Federated Query¹に変換される。TABLE ブロックが最後に接続されているため、最終結果が表として可視化される。

SPARQL クエリの構文がブロックという単位に分割

An Experimental Visual SPARQL Query Editing and Visualizing System for Easier Access to Linked Open Data

Tamirlan TORGAYEV[†], Naoki FUKUTA[‡]

[†]Faculty of Information, Shizuoka University
432-8011, Hamamatsu, Japan

[‡]College of Informatics, Academic Institute, Shizuoka University
432-8011, Hamamatsu, Japan

torgayev.14@shizuoka.ac.jp, fukuta@inf.shizuoka.ac.jp

¹<https://www.w3.org/TR/2013/REC-sparql11-federated-query-20130321/>

され、ユーザがこのブロックを組み合わせ、入出力をつなげることで視覚的にクエリが得られる。これを、本稿ではビジュアルクエリと呼ぶ。提案手法を用いることで、作成したビジュアルクエリは自動的に SPARQL クエリに変換され、実行される。ビジュアルクエリの構成に使用できるブロックには、エンドポイントブロック、制約などを行うブロック (SELECT など)、出力形式 (表、散布図など) がある。類似する手法である Haagらの手法 [2] では、ユーザインタフェース (UI) がグラフパターンに着目して設計されているが、本研究では、SQL を連想させるような構成要素に着目するため、ビジュアルクエリの要素は構文を表現するブロックにしている。

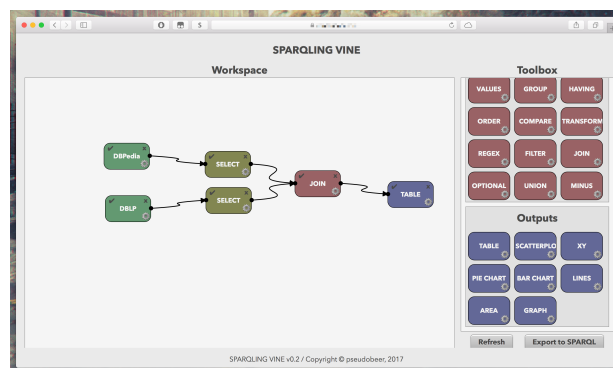


図 1: 提案手法でのビジュアルクエリの一例

提案手法でのクエリ作成フローについて説明する。右にあるツールボックスをクリックするとブロックが左にあるワークスペースに追加される。ブロックを追加、設定し、接続することでクエリを作成することが可能になっている。各ブロックの左上に実行状態が表示され、右下の歯車アイコンをクリックするとブロックの詳細設定や実行結果が表示される。各ブロックについて、そのブロックまでの実行結果や実行状態 (実行中、実行成功、実行失敗) がリアルタイムで更新される。ブロックの実行状態と実行結果を活用して、想定通りの処理が行われているか、どのブロックがクエ

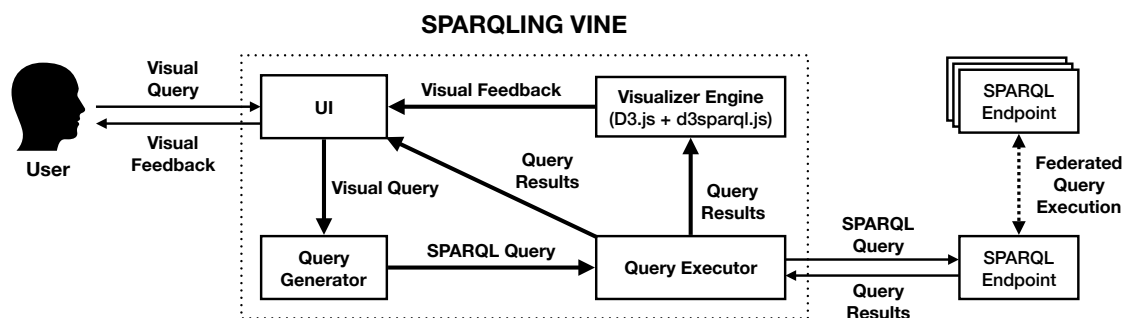


図 2: 提案手法の構成

りの失敗原因となっているかが確認できる。このように、ユーザに対してリアルタイムフィードバックを行うことで、クエリ作成におけるサポートを提供する。

提案手法の構成図を図 2 に示す。ユーザによって UI 上で作成されたビジュアルクエリが、バックグラウンドで動的に SPARQL クエリに変換され、実行される。ブロックの配置、接続、詳細設定に変更が生じれば、クエリ生成ロジック (Query Generator) に対してビジュアルクエリが送信される。生成された各クエリがクエリ実行ロジック (Query Executor) に送信される。クエリ実行後、実行状態及び実行結果が UI に返され、UI で反映される。可視化のクエリに関しては、可視化ロジック (Visualizer Engine) に送信され、可視化の結果が UI に反映される。

3 関連研究

QueryVOWL [2] では、GUI でグラフパターンや FILTER による制限などができ、クエリの実行結果がリアルタイムで反映される。QueryVOWL では、グラフの部分一致を動的に探索できる。QueryVOWL のようなシステムを利用するためには、視覚的なインターフェースの裏側にある構文やセマンティクスを理解する必要があるため、エンドユーザーに向いていないとされている [3]。

SparqlFilterFlow [1] は、SPARQL 問い合わせをデータフローとして表現している。SparqlFilterFlow は、ビジュアルパイプを用いて RDF データを処理するというアプローチに関連する [1]。SparqlFilterFlow はオントロジーとグラフパターンをメインに扱うことが特長であり、提案手法は結果の可視化や途中結果のリアルタイム反映が特長である。

4 まとめ

LOD の問い合わせ言語である SPARQL 言語を使った処理は、エンドユーザにとっては困難なタスクである [3]。本稿では、SPARQL クエリの作成過程、実行結果を可視化する機構を提案した。SPARQL 1.1 の拡張規格で導入された Federated Query は、複雑な構文では理解とデバッグが困難な場合があるが、クエリ作成の柔軟性を増す。本稿の提案手法では、複数エンドポイントを使ったビジュアルクエリは、Federated Query に変換され、クエリの句、データエンドポイント及び可視化形式をブロックとして表現している。ユーザーはこれらのブロックを使用してビジュアルクエリを作成し、ビジュアルクエリが自動的に SPARQL クエリに変換されて実行される。

参考文献

- [1] Haag, F., Lohmann, S. and Ertl, T.: *SparqlFilterFlow: SPARQL Query Composition for Everyone*, pp. 362–367, Springer International Publishing (2014).
- [2] Haag, F., Lohmann, S., Siek, S. and Ertl, T.: *QueryVOWL: A Visual Query Notation for Linked Data*, pp. 387–402, Springer International Publishing (2015).
- [3] Soyulu, A., Giese, M., Jimenez-Ruiz, E., Kharlamov, E., Zheleznyakov, D. and Horrocks, I.: Ontology-based end-user visual query formulation: Why, what, who, how, and which?, *Universal Access in the Information Society*, Vol. 16, No. 2, pp. 435–467 (2017).