

# ウェブオントロジ言語を利用した知的エージェント間の照会応答方式

野口 敦史† 多田 政美†

日本アイ・ビー・エム(株) ソフトウェア開発研究所††

## 1. はじめに

セマンティック・ウェブ(以下「SW」)で利用される知識ベースをウェブオントロジと呼び、その記述言語としてW3Cが標準化を進めているOWLがある。また、知的エージェント間(以下「エージェント」)の通信は、FIPAによってメッセージ形式としてACL[1]が提案されているが、そのメッセージ内容(以下「ACL内容」)を記述する言語に関しては規定していない。

SWの実現においては、エージェント間の通信が必須であり、その照会言語としてOWL-QL[2]が提案されている。OWL-QLは、ACL内容をOWLで記述することを提案している。

一方、SWとウェブ・サービス(以下「WS」)を統合するセマンティック・ウェブ・サービス(以下「SWS」)のフレームワークは参考文献[3]にあるような方式が提案されている。また、OWLによってWSにメタ・データを付加するためのスキーマとしてOWL-S[4]も提案されている。こうしたフレームワークの実現には、複数のエージェントが協調して作業を行い、ユーザーからの要求に対して適切なWSを発見して実行することが必要である。

本論文では、SWSのフレームワーク上のエージェント間のACL内容を役割別にOWLのクラスとして定義し、その一部にOWL-QLを採用した照会応答方式を提案する。

## 2. セマンティック・ウェブ・サービスとエージェント

### 2.1. セマンティック・ウェブ・サービス

本論文で前提としたSWSのフレームワークを図1に示す。ここでは、4種類のエージェントが互いに送受信するメッセージを実線の矢印で示している。4種類のエージェントとは、ユーザーに対するフロントエンドとして機能するパーソナル・エージェント、適切なWSを発見して必要に応じてそのオペレーションの実行まで行うブローカ・エージェント、エージェントのイエローページサービスとして機能するディレクトリ・ファシリテータ、WSの実行を代行するWSエージェントである。

各エージェントは、必要に応じてOWLで記述されたオントロジとOWL-Sで記述された各WSのサービス記述を参照して、ユーザーの要求に対する適切なWSを発見し実行する。

一般的にエージェントは、知識ベースとしてのオントロジを参照すべき場合が多く、SWではその標準としてOWLがある。したがって、SW上のエージェントはOWLを処理するための機能(OWLプロセッサ)を持つことが望ましいと考えられる。本方式のようにACL内容全体をOWLのみで記述することには、これをそのままOWLプロセッサで処理できるという利点がある。OWL-QLの例[5]のように、XML文書中にOWL断片を埋め込む方式ではこの利点は得られない。

OWL Messaging Method for Intelligent Agents

†Atsushi Noguchi, Masami Tada

†† Yamato Software Development Laboratory, IBM Japan

1 OWL-QLはDQLの、OWL-SはDAML-Sの後継仕様である。

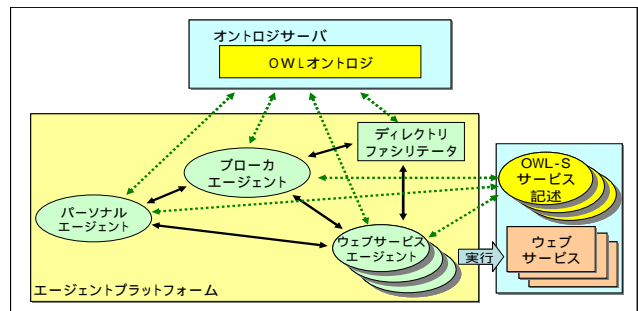


図 1 セマンティック・ウェブ・サービスのフレームワーク

### 2.2. 各メッセージ・クラスの役割

図2にシーケンスの一例を示す。図中で矢印に添えられたメッセージ名は、ACL内容で利用される同名のOWLのクラスとして定義される。これらをメッセージ・クラスと呼び、以下<>付きで表記する。例えば、<Query>は Query クラスのメッセージを表す。本論文で提案するのは図2中に示される8種類のメッセージ・クラスである。各々の役割を図2のシーケンスに従って説明する。

WSエージェントは起動時に、ディレクトリ・ファシリテータに<Register>を送り、自分自身を登録する。ユーザーからの自然文による入力があったとき、パーソナル・エージェントはそれを<Query>に変換してブローカ・エージェントに送る。ブローカ・エージェントは<Query>に対する応答を用意するのに利用可能なWSエージェントを検索するために、ディレクトリ・ファシリテータに<Search>を送る。ディレクトリ・ファシリテータは、そのディレクトリに登録されているWSエージェントで該当するものを<SearchResult>で返す。ブローカ・エージェントは必要に応じて、WSを呼び出すために<Execute>をWSエージェントに送って実行させ、その結果を<ExecuteResult>として取得する。ブローカ・エージェントは、その結果として得られたデータを元に、元の<Query>の制限条件等で絞り込みを行い、最終的な結果を<Answer>としてパーソナル・エージェントに返す。(3.1を参照のこと。) パーソナル・エージェントは必要に応じて、<Answer>の中に含まれるWSエージェントのIDとOWL-Sプロ

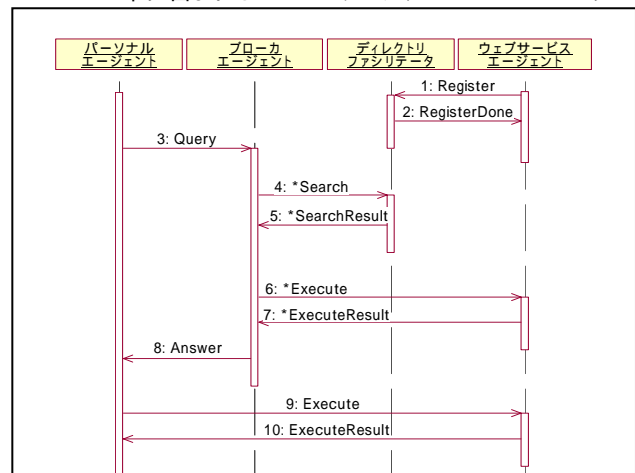


図 2 エージェント間メッセージのシーケンス例

セスのID等を元に、<Execute>をWSエージェントに送り、これを実行する。

### 3. メッセージ・クラスの定義

<Query>、<Answer>には、OWL - QLの提案する照会応答の考え方を取り入れ、これをOWLのみで記述した。これはOWL - QLの考え方が、定型化できない照会内容を表現するのに適しているからである。それ以外のメッセージ・クラスは、OWL - QLでの照会という概念には該当しないので、これらを独自にOWLのクラスとして定義した。

#### 3.1. Query/Answer クラス

メッセージ・クラスの一例として図3に<Query>の利用例を示す。Query クラスのインスタンスとしてACL内容全体を定義し、そのプロパティとしてpremise、queryPattern、mustResolveを持つ。各々はOWL - QLで提案されている概念をOWLで新たに定義したものである。さらに、照会対象に対して行うべき操作を表すプロパティとして queryAction を独自に追加した。OWL - Sでは、OWL - SプロセスをWSの一個以上のオペレーションに対して関連付けることを提案しているが、筆者らはOWL - Sプロセスに「操作」とその「対象」を表す2種類のメタ・データを付加することで、この queryAction との関連概念によるマッチングを行うように拡張した。これはユーザーからの入力文が単に「対象」の検索ではなく、その「操作」を希望している場合にも対応できるという利点がある。

例えば、図3に示した<Query>は、ユーザーからの入力文が「小倉北区で開かれるホームページ作成講座を受けたい」の場合である。パソコン講座の受講申し込み用のOWL - Sプロセスに対しては、「対象」として「パソコン講座」、「操作」として「受講する」が付加されているとする。この両者に対してそれぞれ「ホームページ作成講座」、「受ける」が関連概念であることが、オントロジを参照することで判断でき、マッチングが可能となる。

図3の Query クラスによって照会しているのは、premise 内の QueryTarget クラス(網掛け部分)のインスタンスである。これは「ホームページ作成講座」クラスと、「地域を持つ」プロパティに「福岡県北九州市小倉北区」という値を持つ Restriction クラスの積集合として定義している。また、queryPattern プロパティと

```
<owl:Query>
  <owl:premise>
    <owl:Class rdf:ID="QueryTarget">
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Class rdf:about="education;ホームページ作成講座" />
        <owl:Restriction>
          <owl:onProperty rdf:resource="&root;地域を持つ" />
          <owl:hasValue rdf:resource="&address;福岡県北九州市小倉北区" />
        </owl:Restriction>
      </owl:intersectionOf>
    </owl:Class>
  </owl:premise>
  <owl:queryPattern>
    <QueryTarget rdf:ID="x">
      <root:地域を持つ rdf:resource="&address;福岡県北九州市小倉北区" />
    </QueryTarget>
  </owl:queryPattern>
  <owl:mustResolve rdf:resource="#x" />
  <owl:queryAction rdf:resource="&root;受ける" />
</owl:Query>
```

図3 Queryクラスのインスタンスの例<sup>2</sup>

<sup>2</sup> 例に示したOWL文書は簡単のため省略して表記した部分がある。

mustResolve プロパティで QueryTarget クラスのインスタンスが照会の対象であると定義している。

ブローカーエージェントは回答の候補となるすべてのインスタンスを取得してから、OWLプロセッサを利用して QueryTarget クラスのインスタンスをすべて取得することで、条件による絞り込みを行う。その結果を Answer クラスのインスタンスとして生成して<Answer>としてパーソナル・エージェントに送り返す。

#### 3.2. メッセージ・クラス間の関係の定義

その他のメッセージ・クラスも同様に、その目的ごとにOWLのクラスとして定義した。また、各クラスの持つプロパティに cardinality の指定を付けることで、必須とオプションの違いも明確に定義した。

すべてのメッセージ・クラスは Query/Answer クラスのようにリクエストとそのレスポンスの関係にあるペアを成している。こうしたメッセージ・クラス間の関係もOWLで定義したので、あるメッセージ・クラスを受け取った場合に、応答として返すべきメッセージ・クラスを容易に特定することができる。

これは新規のメッセージ・クラスを追加したい場合に、エージェントを変更することなく、メッセージ・クラスのスキームを入れ替えるだけで対応が可能になるべきである、という将来的な方向性を考慮したものである。

### 4. まとめ

SWSのフレームワーク上で稼動するエージェント間の各種メッセージを、その役割ごとにOWLのクラスとして定義する方式を提案した。その一部として、OWL - QLの方式を採用した場合の、OWLのみによる表現方法と処理方法の一例を示した。この方式は検証システムとして実装され、その実用性が確認済みである。

今後の課題としては、OWL - QLの提案する「対話」の考え方を取り入れ、回答の全個数や全応答時間が不明な場合でも先に見つかった回答から順次応答するような、より効率的な照会応答方式を実現したいと考えている。

#### 謝辞

本研究は総務省の委託研究「ITを活用した企業間連携・起業支援プラットフォームに係わる要素技術に関する調査研究」の一部として実施したものである。研究の機会を与えていただいたことに感謝する。

#### 参考文献

- [1] ACL, <http://www.fipa.org/repository/aclspecs.html>, FIPA
- [2] Fikes, R., Hayes, P. and Horrocks, I., "OWL-QL - A Language for Deductive Query Answering on the Semantic Web", [http://ksl.stanford.edu/KSL\\_Abstracts/KSL-03-14.html](http://ksl.stanford.edu/KSL_Abstracts/KSL-03-14.html), KSL(Stanford Univ.) (2003).
- [3] 村上佐枝子, 槌谷一, 豊島浩文, "ユビキタス環境におけるセマンティック・ウェブ・サービスの提案", 情報処理学会第64回全国大会講演論文集, 平成14年3月
- [4] OWL - S, <http://www.daml.org/services/owl-s/1.0/>, DAML
- [5] OWL-QL some example queries, [http://ksl.stanford.edu/projects/owl-ql/owl-ql\\_queries.txt](http://ksl.stanford.edu/projects/owl-ql/owl-ql_queries.txt), KSL(Stanford Univ.)