

領域オントロジを用いた言語グリッドの Web サービス連携

境 智史[†] 村上 陽平[‡] 石田 亨^{†††}

[†]京都大学工学部情報学科 s-sakai@ai.soc.i.kyoto-u.ac.jp

[‡]情報通信研究機構 yohei@nict.go.jp

^{†††}京都大学情報学研究科 ishida@i.kyoto-u.ac.jp

1. はじめに

現在，サービス指向アーキテクチャを具現化する Web サービスが注目され，インターネット上で提供され始めている．さらに，複数のサービスを連携させ，新たに有用なサービスを生み出す技術に期待が高まっている．しかし，現在はワークフロー作成のコストが大きいため，サービス連携は容易ではない．

現在，NICT，京都大学，和歌山大学を中心とした言語グリッドプロジェクト[1]では，Web に存在する多くの言語サービスをインターフェースの標準化に基づいて Web サービス化する活動を行っている．しかし，サービスを連携させるためには，利用したいサービスの選択を行う必要がある．さらに，連携が期待通りの動作を行うように，Web サービスの置き換えを行う必要があり，これらの作業には大きなコストがかかっている．そこで，言語サービスのオントロジを用いることにより，サービスの選択や置き換えを容易にするというのが本研究の目的である．

まず，2 章で，サービス連携のプロセスを述べ，3 章でどのようにしてオントロジを用いるのかを述べる．本論文では抽象的な Web サービスをサービスと記述し，実行可能な Web サービスのみを Web サービスと記述する．

2. ワークフロー作成プロセス

現在，言語グリッドを用いたワークフローの作成は以下の手順で進められる．

- I. **利用したい Web サービスを選択**
Web サービス一覧の中から利用する Web サービスを選択する．
- II. **Web サービスの実行フローを作成**
連携する際の Web サービスの実行順序や，制御構造などを記述する．
- III. **Web サービスの入出力の対応付け**
変数に値を割り当て，サービス間の入出力を対応付ける．

Web service composition using domain ontology in language grid

[†]Satoshi Sakai, Department of Information Science, Kyoto University

[‡]Yohei Murakami, National Institute of Information and Communication Technology

^{†††}Toru Ishida, Department of Social Informatics, Kyoto University

IV. 作成したワークフローのチューニング

連携が期待通りの結果を出力するように，ワークフロー中で呼び出される Web サービスを入れ替える．また必要に応じて，II, III の手順を繰り返し，実行フローや入出力の対応付けに対し変更を行う．

言語グリッドにおいて，利用する Web サービスのインターフェースは標準化されており，同じインターフェースを持つサービスへの入れ替えは容易である．しかし，サービスの種類が多岐にわたる場合，適切なサービスの選択には大きなコストが存在し，利用したいサービスに実現可能な Web サービスが存在しない場合には，サービスの種類を広げて再び同様の選択が必要となる．また，異なるインターフェースを持つサービスへの入れ替えに伴う入出力の対応の変更は全て人手で行われるため，ワークフロー作成のコストも大きい．

3. オントロジによるワークフロー作成支援

本研究では，ワークフロー作成の際に，利用したいサービスの選択及び Web サービスを入れ替えた際の入出力の対応付けを半自動的に行うことを目標にしている．

そのために，言語サービスのオントロジを用いた以下の 2 つの機能により Web サービス連携の効率化を行う．本研究で用いる言語サービスのオントロジ例を図 1 に示す．

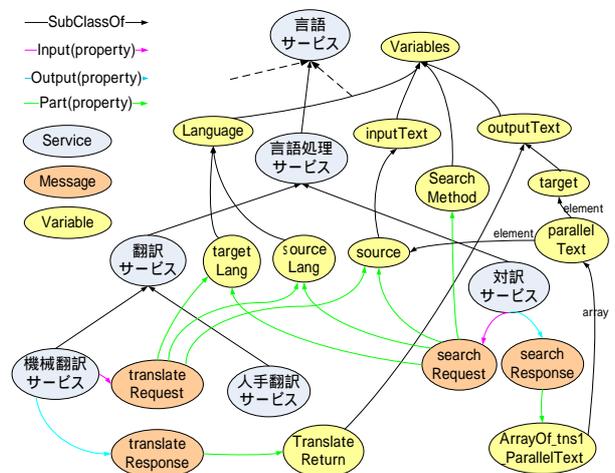


図 1 言語サービスのオントロジ例の一部

3.1 代替サービスの提案

まず、ユーザ（ここでいうユーザはワークフロー作成者を指す）から指定されたサービスの種類を元に、全てのクラスのサブクラスを探索し、指定されたサービスの役割を実行するサービスの種類をすべて見つけ出す。そして、それらのサービスを実行する Web サービスをユーザに提案することにより、Web サービス選択の効率化を図る。

また、ユーザが意図したサービスが提案されない場合、一つ上位のサービスに探索範囲を広げることで、より多くのサービスを探索する。

例えば図 1 のオントロジが与えられたとする。ユーザが翻訳サービスを実行する場合、そのサブクラスである機械翻訳サービスと人手翻訳サービスも該当するので、それらの 3 つのサービスを実現する Web サービスの提案を行う。もし、この 3 つのサービスを実行する Web サービスが存在しない場合には、探索範囲を一つ上位の言語処理サービスにすることで、新たに対訳サービスを含めた 5 つのサービスを実現する Web サービスの提案を行う。

3.2 Web サービスの連結可能性の検証

すでにワークフローが完成した状態で、実行する Web サービスを変更しようとする際、先ほどの方法で Web サービスを再び提案し、選択を行う。ここで、サービスの選択は 2 つのパターンに分けることができる。

- ・以前のサービスと同クラスのサービスに変更
- ・以前のサービスと別クラスのサービスに変更

同クラスの場合は、インターフェースがそろっていることから、入出力を変更することなく終了することができる。

しかし、別クラスに変更を行う場合は、Web サービスのインターフェースが異なるため、まずその Web サービスの入出力の変数に対して再び割り当てを行う必要がある。そこでオントロジを用いることにより以前の Web サービスの変数が、新しい Web サービスに置き換えたときにどの変数に割り当てるのが良いのかを判断する。

まず、サービスのプロパティを見ることで、変更前のサービスと、変更後のサービスで同じクラスの変数を用いているかを調べる。同じクラスの変数が使われていた場合は、それらの変数の割り当てに変更は加えない。別のクラスの変数を用いられていた場合は、以前の変数から新しい変数に変更する。変更する変数が 2 つ以上ある場合には、それぞれ、オントロジの階層グラフ上での変数の距離が近い組み合わせを変更する変数とする。

例えば図 1 の例において、ユーザが機械翻訳サービスを対訳サービスに置き換えを行い、そのときにシステムが行う出力値の割り当ての提案を考

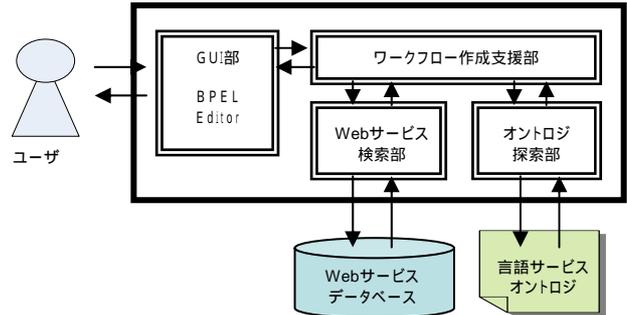


図 2 システム構成図

える。機械翻訳サービスの出力では translateReturn という変数が用いられており、対訳サービスでは、target と source という変数が用いられている。そこで、translateReturn から target と source のそれぞれの距離を計算する。そして、距離の小さい target という変数に割り当てを変更することを決定する。

この割り当ての変更を Web サービスが利用する変数すべてに対して行う。

この結果を用いて、ユーザに変数割り当ての変更を提案し、ユーザが提案どおりに変更を行うかを判断することにより、Web サービス変更時の入出力の対応付けの効率化を図る。

4. 実装

本研究では、3 章で述べた 2 つの機能を実装する。

システム構成図を図 2 に示す。GUI 部は eclipse のプラグインとして提供されている BPEL Designer Editor を使用する。Web サービス検索部はサービスの種類を与えると、適切なサービスを返す。オントロジ探索部は、サービスのサブクラスを探索する機能と、サービスの変数を調べる機能を持つ。ワークフロー作成支援部は Web サービス検索部とオントロジ探索部を用いて 2 つの機能を GUI 部に提供する。

5. おわりに

本研究では、言語サービスのオントロジを用いて、Web サービスの検索と Web サービス入れ替えに伴う入出力の対応付けの変更を効率化する手法を提案し、それらを実装した。この手法を用いることで、ワークフローのチューニングを行う際に、Web サービスの変更を半自動的に行うことが可能となる。今後の課題としては、言語グリッドプロジェクト[1]で作成された Web サービス群に対して、提案手法の性能の評価を行っていく。

参考文献

- [1] Ishida, T.: Language Grid: An Infrastructure for Intercultural Collaboration, *IEEE/IPSJ Symposium on Applications and the Internet (SAINT-06)*, pp.96-100 (2006).