

価値に基づく Web サービスの動的連携ブローカとその評価

中村 一仁 柘植 亮人 青山 幹雄

南山大学 数理情報学部 情報通信学科

本研究では異なるインタフェースで実装された Web サービスを動的に連携させるために、価値に基づくサービス変換ブローカを提案する。Web サービスの動的連携では、連携する複数の企業間で Web サービスのインタフェースが異なることやサービスの信頼性や接続相手の信頼性を定量的に保証できない問題がある。本研究では、サービスブローカによるサービスの仲介によりインタフェースの差異を解消する方法を提案する。仲介における情報の信頼性や欠落の問題を解決するために「サービス価値」の概念を提案する。サービス価値を利用することで、サービスの品質を保証できる。

WS-Routing を Java を用いて実装し、サービス変換ブローカのプロトタイプを開発した。さらに、SOAP メッセージを拡張し、信頼度を付加できるようにした。この実装に基づきサンプルサービスを作成し、サービス変換と信頼度の有用性を実証、評価した。

Value-Based Dynamic Collaboration of Web Services

Kazuto Nakamura Akihito Tsuge Mikio Aoyama

Department of Information and Telecommunication Engineering,
Faculty of Mathematical Sciences and Information Engineering, Nanzan University

This article proposes a service broker based on the value to enable the Web services of different interfaces collaborate dynamically. The problems arise due to the behavioral differences of interfaces of Web services and lack of quantitative expression of the confidence of service partners. The proposed service broker mediates the difference in interfaces by service conversion based on the confidence. For this, we propose a concept of value of service, with which the broker can guarantee the quality of service.

We developed a prototype of service broker with WS-Routing implemented in Java. We extended SOAP message format to include confidence of service. And we evaluated the effectiveness of proposed service broker and confidence of services.

1. はじめに

Web サービスは、XML をベースとした技術を用いてインターネット上の様々な資源を、ソフトウェアシステムが利用可能とする技術である[1, 16]。Web サービスを用いることにより、企業間でシステムの柔軟な連携や各種サービスを組み合わせた高度なサービスを構築することが可能となる。

現在の技術では、インタフェース情報を定義した WSDL(Web Services Description Language)[7]とサービスの組み合わせ方を定義した BPEL4WS (Business Process Execution Language for Web Services)[2]によりサービスの連携が可能である。さらに、UDDI(Universal Description, Discovery and Integration)[5]レジストリを用いることにより、サービスの動的な発見と利用ができる。

しかし、サービスの意味が定義できないため、動的に連携できるサービスは同じポート型のインタフェースで実装されたサービスに限定される。

また、Web サービスの利用形態として、サービスブロー

カを仲介した方法が期待されている。サービスリクエストとサービスプロバイダ間を、サービスブローカを経由してサービスを利用することにより、様々な付加価値の提供が可能となる。

しかし、インターネットを利用するソフトウェアは、従来から信頼性の問題がある。これは、インターネットの分散性や匿名性に起因する。Web サービスの場合はサービスブローカによる仲介や複数サービスを組み合わせることが可能なため、問題はより複雑となる。システム連携に Web サービスを用いる場合にも動的という Web サービスの特性を活かせない。信頼性の問題を解決することが Web サービスを実用化する鍵となる。

本研究の目的は、異なったインタフェースで実装された意味的に類似のサービスを動的連携することである。さらに、各サービスが提供するサービスの情報量や信頼性などの非機能的特性を「価値」として包括的に定義し、価値を用いたサービスの品質保証と動的連携の実現を目標とする。

本稿では、インタフェースを変換するサービス変換を用いて異なったサービスを連携する方法と、価値空間を用いた品質保証の一つとしてサービスの信頼性を取

り上げ、サービスそのものの信頼性やサービス変換に伴う情報欠落の問題を信頼度を用いて解決する方法を提案する。この方法を検証するため、単語帳システムと Calc システムという価値の評価基準の異なる二つのプロトタイプを作成し、評価を行った結果を示す。

2. サービス指向アーキテクチャ上でのサービスの動的連携の問題点

2.1. サービス指向アーキテクチャ

Web サービスはサービスリクエスト、サービスブローカ、サービスプロバイダから成るサービス指向アーキテクチャをとる。Web サービスの利用の流れは次のようになる。サービスを提供するサービスプロバイダは、Web サービスのインタフェースを決定して WSDL で記述し、UDDIレジストリに登録する。一方、サービスを利用するサービスリクエストは UDDI レジストリを検索して WSDL を取得し、提供されるサービスを利用する。

Web サービスによって提供される情報は、利用者の計算機内にあるのではなく、インターネット上の不特定のサービスプロバイダによって処理される。サービスを人間の手を介すことなく自動的に連携し、利用者の期待する結果を得るためには以下の三つの問題がある。

2.2. 連携するサービスのインタフェースの違い

WSDLによって定義できる情報はサービスのインタフェース情報や使用プロトコルの指定のみである。サービスの意味情報が明確ではないため、サービスの動的連携を必ずしも保証できない。サービスの意味を定義する研究はセマンティック Web の分野で進められているが、Web サービスのインタフェースとしては実用段階に至っていない。

一方、アプリケーションドメイン毎にサービスのインタフェースを標準化する試みがある。しかし、Web サービスのインタフェースを統一することは難しい。

2.3. 期待するサービスの発見・選択の問題

Web サービスをシステムが自動的に連携する場合には、リクエストの期待するサービスを発見し連携するための基準が必要である。この基準には、サービスの機能的特性と非機能的特性の二つがある。機能的特性は、サービスのインタフェースやレジストリによって得ることができる。一方、非機能的特性は、信頼性や情報の深さ、応答スピードが挙げられる。現在の Web サービス技術では、非機能的特性は定義されていない。最適な Web サービスを動的に発見し利用するためには、機能的特性と非機能的特性の両方を知ることが必要である。非機能的特性を定義する方法と、二つの特性を利用してリクエストの期待するサービスを選択し、利用するための仕組みが必要である。

2.4. サービスの品質保証の問題

Web サービスを用いて、BtoB(Business to Business)電子商取引などにおけるシステム連携を行う場合、サービスプロバイダにおける処理の安全性や信頼性の問題がある。さらに Web サービスでは、サービスブローカを経由した利用形態や、複数のサービスを組み合わせて利用することが考えられるため、リクエストとプロバイダ間の信頼性を確保し、サービスの品質を保証する仕組みが必要である。

最近では、WS-Security 仕様[4]によるメッセージの暗号化やプロバイダの認証、WS-Reliable Messaging 仕様[6]による重複メッセージの発見やメッセージ処理の順序の確保が可能となった。しかし、これらの技術はサービスそのものの品質や信頼性を保証するものではない。

現在は、サービスの品質や信頼性を保証するため、事前に人間の手によって契約を結ぶことで解決している。しかし、この利用形態では Web サービスの利点であるシステムの動的連携性を活かさない。

3. サービスの動的連携のアプローチ

3.1. サービス変換

Web サービスを動的連携させる前提条件として、連携させるサービスの特性を考えた場合、意味的に類似したサービスを動的連携させる必要性の方が一般的に高いと考えられる。例えば、図書館の図書を検索するサービスリクエストが連立方程式の解を求めるサービスを利用することは期待されていない。しかし、このリクエストが図書を販売するショッピングサイトのサービスを動的利用できたならば、図書館の本だけでなくショッピングサイトの本を検索でき、利用性は高くなる。

我々はこの点に着目し、異なるサービスを既存のリクエストプログラムが利用できるサービスへ変換することでサービスを動的に利用可能ではないかと考える。これを実現するため、型変換を拡張してサービスのインタフェースを変換するサービス変換を提案する。これにより、既存のリクエストプログラムはサービスの名前空間の違いを意識することなく、異なったサービスを利用可能となる。

3.2. 価値

3.2.1. 価値の定義

「価値」とはサービスに関する非機能的特性を定義したもので、サービスの発見・選択と品質保証の問題を解決できると考える。価値は、次の二つに分類できる。

(1) リスク: サービスの信頼性や計算の精度、サービスレベル[8]などが該当し、サービスの利用によって生じるリスクを表す。これを使用して、ある一定以上のリスクを負うサービスを制限できる。

(2) 付加価値: リスクとは逆に、正の価値を表すものであり、他のサービスやサービスプロバイダが提供する情報よりも優れている点を表現できる。

3.2.2. 価値空間

価値を用いて、サービス選択の問題と信頼性の問題は、図 1 の価値空間の概念を用いて定義する。

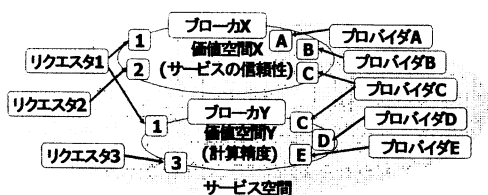


図 1 サービスの価値空間

特定のサービスを提供するサービスプロバイダと、それを利用するサービスリクエストは、一つのサービス空間に属する。さらに、このサービス空間に一定の価値を保証する価値空間を定義する。ある価値空間に属するサービスプロバイダは提供する情報に一定の価値を保証しなければならない。

例えばサービスの信頼性を 99.999999%以上保証する価値空間(図の価値空間 X)を定義すると、この価値空間に属するプロバイダ A~C は、サービスの結果に 99.999999%以上の信頼性を保証しなければならない。一方、価値空間 Y は計算精度を 30 桁保証する価値空間と定義すると、信頼性を 99.999999%保証し、かつ計算精度を 30 桁保証するサービスは両方の価値空間に属すプロバイダ C になる。

このように、価値空間の概念を利用すれば、一定の基準を満たすサービスの選択と品質保証が可能となる。

3.2.3. 価値ブローカを用いたサービスの品質保証

価値空間の概念を用いると、リクエストからのサービス要求に対し、利用するサービスを選択しサービスの品質を保証する価値ブローカ(図 2)が実現できる。

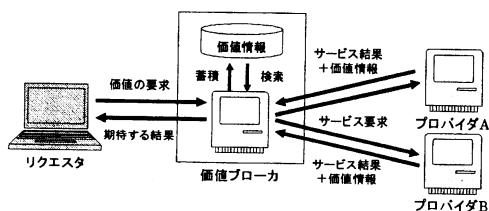


図 2 価値ブローカ

リクエストからのサービス要求に「期待する価値」の情報を付加し、価値ブローカへサービス要求を行うと、価値ブローカは蓄積した価値情報を元に利用するサービスやサービスプロバイダを選択する。選択したサービスを利用したサービスの結果には価値情報が付加されて

おり、この価値情報とリクエストの要求する価値を比較することによりサービス品質を保証する。

3.3. 価値空間を用いたサービスの信頼性の解決

3.3.1. 信頼度の定義

Web サービスの信頼性の問題を「サービスの信頼性」という価値空間を用いて解決するためには、サービスの信頼性を定量的に表現する必要がある。

従来のクライアント/サーバシステムと異なり、Web サービスの場合には接続するサービスプロバイダが動的に変化することや、サービスブローカを経由した利用形態が考えられるため、サービスの信頼性を表現することは困難である[3]。さらに、前述のサービスを変換を行う場合に、その変換が完全に行えるとは限らない。異なるサービスではインタフェースの定義域が異なるためサービス変換を行うと情報が失われる場合がある。

これを考慮して、Web サービスの信頼性を保証する三つの信頼度を定義する。

- (1) プロバイダの信頼度: リクエストがプロバイダをどれだけ信頼できるかを表す指標である。リクエストごとにプロバイダの URL リストとその信頼性を対として定義する。
- (2) サービス変換の信頼度: サービスブローカが Web サービスのインタフェースの変換を行う場合、その変換の信頼性を表す指標である。サービス変換のプログラムが値を決定し、変換が行われた SOAP メッセージの順方向、逆方向のメッセージに埋め込まれる。
- (3) コンテンツの信頼度: サービスプロバイダが提供する情報の信頼性を表す指標である。プロバイダプログラム内で値を決定し、逆方向の SOAP メッセージに埋め込まれる[9]。この値を用いて、従来のリクエストとプロバイダが直接接続する利用形態でも、サービスの品質が保証できる。

3.3.2. 信頼度の利用方法

前節で提案した三つの信頼度を支援する Web サービスを利用すると、図 3 に示すように、プロバイダが送信する逆方向の SOAP メッセージにコンテンツの信頼度を埋め込んでブローカに送信できる[9]。ブローカは自らが持つプロバイダの信頼度とメッセージ中のコンテンツの信頼度を参照し、新たな信頼度の計算を行いメッセージ中のコンテンツの信頼度を更新する。さらにブローカがサービス変換を行った場合は、サービス変換の信頼度をメッセージに埋め込み、リクエストへ返信する。リクエストが受け取った返信メッセージには、コンテンツの信頼度、サービス変換の信頼度が埋め込まれている。さらにリクエスト自身が持つプロバイダ(ブローカ)の信頼度を利用することによって、Web サービスの品質を保証できる。信頼性が低いサービスは、他のサー

ビスプロバイダからの情報と照合し情報の正当性を確認することにより、サービスの信頼性を保証できる。

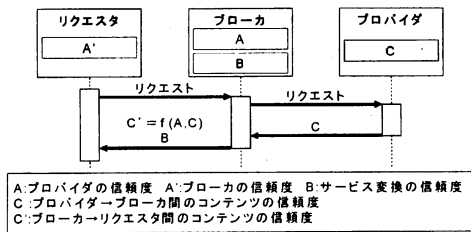


図 3 信頼度の利用方法

4. サービスの動的連携の方法

Web サービスのインタフェースを変換するサービス変換の方法として次の二つが考えられる。

4.1. サービスブローカを利用する方法

リクエストプログラムは変更せず、仲介を行うサービスブローカがサービス変換を行う方法である(図 4)。

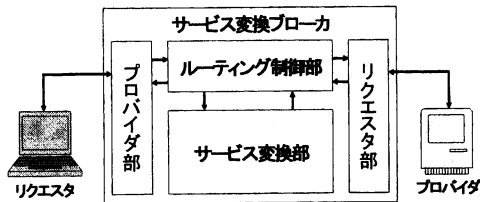


図 4 サービスブローカのアーキテクチャ

サービスブローカは、リクエストプログラムが送信した SOAP メッセージを変換プログラム内のプロバイダ部で受け取り、これをサービス変換部で異なる Web サービスに対応するメッセージへ変換し、リクエスタ部を用いて目的のサービスプロバイダへ送信する。

この方法の利点は、リクエスタが Web サービスのインタフェースの違いを意識することなく異なる名前空間の Web サービスを使用できることにある。Web サービスを追加して連携する場合にも、サービスブローカの変換プログラムのみ作成すれば対応できる。

さらに、図 5 に示すように一度、共通の名前空間を介してサービス変換を行うことで、直接変換できない Web サービスを変換可能になる。

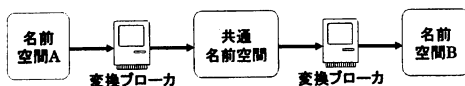


図 5 共通名前空間を使った変換

サービスブローカで SOAP メッセージを変換する方法はセキュリティに関して脆弱である。ただし、メッセージの暗号化とプロバイダの認証を行う WS-Security 仕様 [3] などによって改善が期待できる。

4.2. リクエスト内で変換モジュールを利用する

Web サービスを変換するためのプログラムをモジュール化し、それをフレームワーク化したリクエストプログラム内に自動的にダウンロードして使用方法である(図 6)。

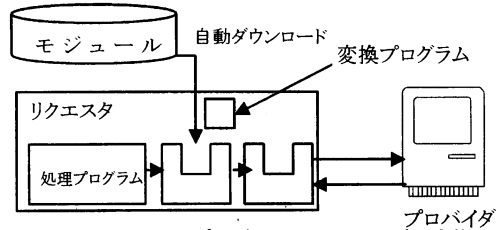


図 6 リクエストプログラム内でのサービス変換

この方法では、リクエスト内に二つのフレームワークが必要である。一つは図中の右側のものであり、WSDL 記述から Web サービスを利用するためのプロキシを作成し実行する。これは WSDL4J などによって実現されている。二つ目のフレームワークは、サービスの動的連携を行う部分である。このフレームワークを利用するためには、リクエストプログラムの構造を知っている設計者が、Web サービスのプロキシと処理プログラムとの橋渡し部分をモジュール化しインターネット上に公開する必要がある。リクエスタが Web サービスを使用する時には、フレームワークがインターネット上の Web サービス用のモジュールを自動的にダウンロードし実行する。

この方法の利点は、サービスブローカを通らないためセキュリティ面で優れている。ただし、一つの Web サービスに対してリクエスタ毎にモジュールを作成しなければならないと効率的でない。またモジュールをダウンロードするコストとリクエスタがモジュールの使用を支援していなければならない、リクエスタの高機能化が必要となる。

4.3. 動的連携方法

前者の方法は、コスト面で有利である点と、サービスに価値を付加できる点で、拡張性に富んでおり、我々はサービスブローカを利用する方法を採用した。

サービスブローカを実現するため、メッセージの経路情報を定義できる WS-Routing 仕様 [12] を Java を用いて実装し、サービス変換ブローカを試作した。試作ソフトウェアの規模は 4,223 行となった。

5. 信頼度による信頼性の保証方法

5.1. 信頼度のメタモデル

価値空間を用いた品質保証の具体例としてサービスの信頼性を定義する信頼度を取り挙げる。

コンテンツとサービス変換の信頼度はメッセージ中の SOAP ヘッダーに付加され、ブローカを経由しリクエスタへ送信される。このメタモデルを図 7 に示す。

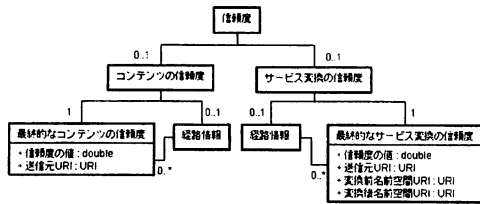


図 7 信頼度のメタモデル

信頼度の最上位要素は「信頼度」である。そのサブクラスとして「コンテンツの信頼度」と「サービス変換の信頼度」がある。コンテンツの信頼度は、「信頼度の値」と「送信元 URI」を属性として持つ。一方、サービス変換の信頼度は「信頼度の値」、変換を行う「ブローカの URI」、そして変換前後のサービスの名前空間の URI を属性として持つ。それぞれの信頼度には「経路情報」というオプションクラスが含まれており、複数のサービス変換ブローカを経由した場合にコンテンツの信頼度やサービス変換の信頼度をスタックに保存できる。

5.2. 信頼度を扱うアーキテクチャ

信頼度を含んだメッセージを受け取り、信頼度を評価するアーキテクチャを図 8 に示す。

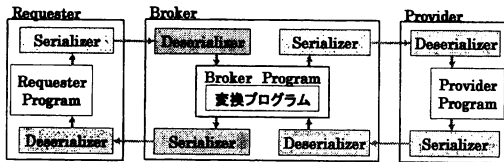


図 8 信頼度を評価するアーキテクチャ

図 7 で示した信頼度の構造は、リクエスタ、ブローカ、プロバイダ内では、プログラムが扱いやすいようプログラミング言語に依存したオブジェクトに変換して扱うことにする。このため、XML のメッセージとオブジェクトを相互に変換する仕組みが必要となり、図中の Serializer と Deserializer がこれを行う。さらに、Serializer はビジネスロジック(変換ブローカでは変換プログラム)から、コンテンツの信頼度やサービス変換の信頼度を受け取りメッセージに付加する。一方 Deserializer は、メッセージ中のコンテンツの信頼度と自らが持つブローカの信頼度を用いて、コンテンツの信頼度の値を更新する。

この信頼度を評価するアーキテクチャを実現するため、信頼度のメタモデルを XML Schema で定義し、メッセージ処理を Apache Axis を用いて実装した。開発したソフトウェアの規模は 2,066 行となった。

6. プロトタイプの作成と評価

サービス変換ブローカと信頼度の有用性を、性質の異なる二つのサービスを試作して評価した[11]。

6.1. 単語帳システムの作成と評価

単語帳システムは、単語の意味を調べるシステムである。結果をリクエスタの画面に表示する。得られる情報に高い精度の信頼性は必要がなく、人間が読んで解釈できる程度のサービス変換を行えば実用上問題は無いと考えられる。単語帳システムはすでに複数の分野の Web サービスが運用されている。それを動的連携することにより様々な視点からの単語検索が可能となる。

評価にあたり、データ構造やインタフェースの異なる次の三つのサービスを利用した。

- (1) NetDicV06: 三省堂が試験運営する辞書サービスであり、英和、和英、国語辞典から約19万項目の単語の意味を検索できる。
(<http://btonic.est.co.jp/NetDic/NetDicv06.asmx>)
- (2) Insider's computer Dictionary (ICD) サービス: @IT が運営する、コンピュータ用語に限定した事典である。
(<http://www.iwebmethod.net/icd1.0/icd.asmx>)
- (3) SimpleWordBook: 我々の研究室で運用している単語帳システムであり、学生が登録したソフトウェア工学に関する単語の意味を検索できる。

このサービスを連携する二つのサービス変換ブローカを試作した(図 9)。ソフトウェアの規模は 7,571 行である。

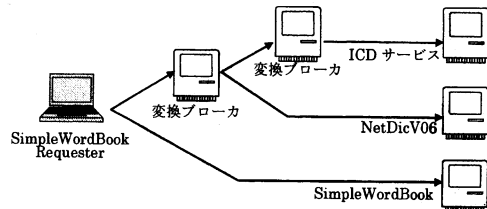


図 9 単語帳システムの構造

SimpleWordBook リクエスタプログラムから単語検索を行った結果を表 1 に示す。SimpleWordBook サービスのみを利用して検索できる単語の数は限られているが、サービス変換ブローカを用いて異なる辞書のサービスを組み合わせると、より価値のある検索結果を得ることが可能になった。

表 1 単語の検索結果

検索語	SimpleWord	NetDicV06	ICD
Apple	1 語	8 語	2 語
Java	3 語	4 語	14 語
home	0 語	50 語	1 語

次に、この三つの辞書サービスの専門領域は異なる。SimpleWordBook サービスは「ソフトウェア工学の専門知識」、NetDicV06 は英和・和英辞書サービスのため「一般」、ICD サービスは「IT 分野における専門知識」として分類できる。異なる分野の単語帳システムを利用して得られる単語の意味情報の関係を調べるため

に、表 1 中の「Apple」を例として検証する。

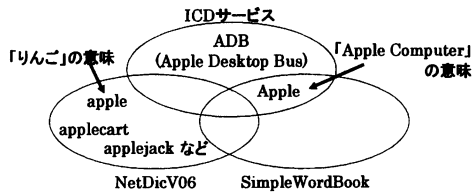


図 10 「apple」の検索結果

図 10 より、ICD サービスと SimpleWordBook サービスからは共通の「Apple Computer」の意味が検索できた。各々のサービスを単独で利用した場合と比べサービスを連携することにより、次の価値があるといえる。

- (1) 意味の深化: 二つのサービスによって得た説明の詳細度は SimpleWordBook が数行程度に対し、ICD サービスでは歴史を含めた詳しい説明が記述されており、ICD サービスを連携することによって意味的に深い情報を取得できた。一方、NetDicV06 サービスからは「りんご」の意味や、「Apple」を拡張した単語、慣用語を検索でき、幅広い意味を取得できた。
- (2) 意味の正しさの向上: SimpleWordBook に登録されている単語情報は、研究室の学生が誰でも登録できるため、正確さに欠ける。しかし、サービス連携により、ICD サービスの情報と比較することにより、SimpleWordBook に登録された情報の正しさを確認できた。

単語帳システムの評価は、異なる価値の Web サービスの動的連携が「意味の深化」や「正しさ」という新たな価値の創造や個々のサービスからは得られない新たな視点の提供が可能となることを示唆している。

6.2. Calc システムの作成と評価

Calc システムは、演算(加算、減算、乗算、除算)を行うシステムであり結果に高い精度が求められる。このような特性のサービスには、天文学や原子力などの分野で利用する精密な計算やクレジットカード決済サービスなどがある。Calc システムでは、「品質の保証」と「サービスの経路選択」の二つの評価を行った。

(1) 品質保証の評価

品質評価にあたり、int 型演算を行う CalcService と double 型の演算を行う DoubleCalcService を作成し、さらにサービス変換ブローカを作成した(図 11)、試作システムの規模は 2,466 行となった。

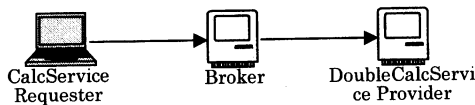


図 11 Calc システムの構造

DoubleCalcService のリクエストプログラムを利用して、CalcService を利用した結果を表 2 に示す、

表 2 サービスの利用結果

演算	in1	in2	結果	コンテンツの信頼度	サービス変換の信頼度
加算	4	2	6	0.8	1.0
除算1	4	2	2	0.8	1.0
除算2	4	3	1	0.8	0.6

精度が落ちない演算(加算、除算 1)を利用した場合には、サービス変換の信頼度が「1.0」となっておりサービスの利用結果が保証されていることがわかる。しかし、除算 2 を利用した場合には、DoubleCalcService において小数点を含む結果が出力されるため誤差が生じ、サービス変換の信頼度が低くなる。このように、完全にサービスを変換できない場合があり、リクエストがサービス変換を行った結果を完全に信用することは危険である。信頼度を SOAP メッセージに付加し、リクエストに変換の精度を知らせることにより、リクエストはサービス変換を利用するか判断可能となる。

(2) サービスの経路選択の評価

Calc サービスの二つ目の評価は、変換や計算の精度が異なる二つのサービス変換ブローカとサービスプロバイダを図 12 のように接続し、信頼度の値を用いて利用するブローカやプロバイダを選択できることを評価した。このシステムの中で、BrokerB は BrokerA よりも高精度にサービス変換することが可能であり、CalcServiceB は CalcServiceA よりも高精度の演算を行うことが可能である。

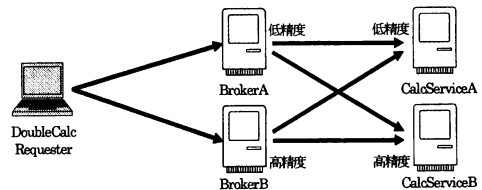


図 12 Calc システムの構造 2

このシステムを用いて、サービスを実行した結果を表 3 に示す。

表 3 サービスの利用結果

Provider	Broker	in1	in2	結果	信 1	信 2
A	A	4.2	3.6	1.000	0.125	0.5
A	B	4.2	3.6	1.166	0.200	1.0
B	A	4.2	3.6	1.000	0.300	0.5
B	B	4.2	3.6	1.167	0.480	1.0

※信1=コンテンツの信頼度 信2=サービス変換の信頼度

表 3 のサービス変換の信頼度(信 2)を見ると、BrokerA に比べて BrokerB がサービス変換をより正確に行っていることがわかる。ただし、このサービス変換の信頼度はサービスブローカが自ら申告した信頼度であるため、これをリクエストプログラムが利用する場合にはリクエストプログラムが持つブローカ信頼度の値で評価した結果を利用するのがよい。一方、コンテンツの信頼度(信 1)を見ると、CalcServiceA に比べて CalcServiceB の方がより正確な演算結果を返すことを知ることができる。

信頼度を利用するとリクエストもしくはブローカが自動的にどの経路を使って Web サービスを呼び出すとよいかを判断でき、より高品質なサービスを探すことが可能となる。ただし、信頼度はリクエストプログラムが持つブローカの信頼度に影響されるため、ブローカの信頼度を信頼できる機関から取得する必要がある。

このように、価値空間の概念に基づいてサービスの非機能的特性を評価することによって、利用するサービスや経由するブローカの選択が可能となり、サービスの結果の品質保証が可能となる。

7. 考察

7.1. サービス変換ブローカを用いた動的連携

サービス変換ブローカの提案により、意味的に類似の Web サービスをインタフェースの違いを吸収して連携可能となった。これにより、サービスの意味的な深化や信頼性の保証が可能となり、各サービスを別々に利用するよりも高い価値を得ることができる。さらに、従来はサービスの機能的特性のみに着目しサービスの連携が行われてきたが、サービスの価値の定義により、信頼性や付加価値といったサービスの非機能的特性を用いた連携が可能となる。

一方、サービス変換や価値を用いたサービスの動的連携をサービスブローカが提供することにより、サービスに付加価値を付けることが可能である。例えば、サービスブローカ上で、リクエストが多い単語の検索結果を上位に移動させることやファイヤウォールで守られているような、直接利用できないサービスも利用可能となる。

サービスブローカを用いてインタフェースの違いを解消し、サービスの非機能的特性を用いた動的連携を実現することにより、Web サービスを機能的特性だけでなく、非機能的特性を用いた様々な視点から連携可能になると考えられる。

7.2. サービス変換ブローカのアーキテクチャ

サービス変換ブローカのアーキテクチャは図 4 に示すようになる。このアーキテクチャを用いて単語帳システムの評価で用いたサービス変換ブローカを試作した。プロバイダ部とリクエスト部のプログラムは、変換する二

つのサービスの WSDL を用いて自動生成可能である。さらに、Apache Axis などの Web サービスフレームワークによって SOAP メッセージはプログラミング言語に依存したオブジェクトに変換される。従って、サービス変換ブローカの開発は、サービス変換部のみコーディングすればよく、開発者はサービス変換に集中できる。

7.3. サービスの価値

サービスの価値を定義することにより、非機能的特性を用いた Web サービスの動的な選択と品質保証が可能となる。

現在は Web サービスのオープン性のみを利用した、静的なシステム連携に留まっている。しかし、価値を利用して動的に品質保証が可能になったことで、動的なシステム連携が可能となり、Web サービス本来の動的特性が最大限に発揮できると考えられる。

また、価値を用いることによりリスクや付加価値の情報をサービスに付加できるようになる。機能的には同じサービスを提供していながらも、他のサービスプロバイダと差別化を図ることが可能となる。従来は意味的に類似のサービスであっても、他社のサービスとデータ構造やインタフェースで差別化を図っていたものが、価値情報を用いた差別化が進み、インタフェースがある程度統一されると考えられる。

8. 関連研究

Web サービスをシステムが動的に連携する技術は様々なものが研究されている[1]。

BPEL4WS などの開発により、ビジネスプロセスを用いて複数のサービスを組み合わせる利用が可能となった。さらに、サービスを動的に連携する研究として、ビジネスプロセスの情報をモデル化し、モデル間の関係や制約条件を用いてサービスを動的に連携する研究[13]やトランザクションに着目しフレームワークを用いて連携する研究[14]がある。

また、サービスの検索においても、UDDI レジストリを用いた意味検索を行う研究[10,17]が行われており、これらにより Web サービスの意味を用いたサービスの動的連携が可能となる。

一方、サービス品質についても、e-Business の分野で研究が進められており[15]、今後 Web サービスにおいてもサービスの品質保証が重要になると考えられる。

9. 今後の課題

以下に、価値を用いた Web サービスの動的連携における、主要な課題を示す。

(1) 価値の定義と利用: サービスの価値はサービスごと

- に異なっていると考えられ、サービスに適した価値を定義する必要がある。また、定義した価値をリクエスト、ブローカ、プロバイダで共有できるように、何らかの形で公開する方法が必要である。
- (2) 価値ブローカの実現: サービスの価値を利用し Web サービスの動的連携を実現するには、サービスブローカから得たサービスの価値を蓄積し、価値空間に基づきサービスリクエストの期待するサービスを動的連携する価値ブローカの実現が必要である。
 - (3) プロバイダの信頼度の初期値: 信頼度による Web サービスの信頼性を保証する方法は、サービス結果の信頼度とプロバイダの信頼度の両方の値に影響され変化する。サービス結果の信頼度が低くてもプロバイダの信頼度が高ければ高信頼度のサービスとして認識されてしまう問題がある。これを解決するために、プロバイダの信頼度の初期値は信頼できる機関から取得する必要があり、新たな問題となる。
 - (4) 信頼度の動的変化: 信頼度は固定値ではなく変化させることにより利用価値の高いものとするのが考えられる。例えば、プロバイダの信頼度の初期値を低く設定し、サービスの利用実績に応じ信頼度を高めることで信頼度を有効活用することが考えられる。

10. まとめ

本研究では、意味的に類似の Web サービスを動的連携することを目的として、ネットワーク上で Web サービスのインタフェースの変換を実現するサービス変換ブローカの提案と評価を行った。サービス変換ブローカを用いると、今までは不可能であった異なる名前空間の Web サービスを連携させることが可能となり、Web サービスの利用性が高まる。

また、インターネットに潜在する信頼性の問題とサービス変換によって生じる情報の欠落の問題を解決するため、サービスの品質保証の仕組みが必要である。そこで、価値空間の概念を用いて、サービスの品質保証を行う方法を提案し、その一つとして信頼性の問題を取り上げた。この問題は信頼度を用いてリクエストやブローカへサービスの品質を定量的に表す方法により解決した。

さらに、単語帳システムと Calc システムの二つのプロトタイプを用いて評価を行い、サービス変換の有用性と信頼度を用いた品質保証の有用性の確認を行った。

参考文献

- [1] T. Alonso, et al., *Web Services*, Springer, 2004.
- [2] T. Andrews, et al., *Business Process Execution Language for Web Services, Version 1.1*, May. 2003, <http://www-106.ibm.com/developerworks/webservices/library/ws-bpel/>.
- [3] 青山幹雄, Web サービスの信用創造・信用仲介モデル, 情報処理学会ソフトウェア工学研究会, Vol. 2002-SE-139, No.11, Oct. 2002, pp. 63-68.
- [4] B. Atkinson, et al., *Web Services Security (WS-Security), Version 1.0 05*, Apr. 2002, <http://www-106.ibm.com/developerworks/webservices/library/ws-secure/>.
- [5] T. Bellwood, et al. (eds.), *UDDI, Version 3.0.1*, Oct. 2003, <http://uddi.org/pubs/uddi-v3.0.1-20031014.htm>.
- [6] R. Bilorusets, et al., *Web Services Reliable Messaging Protocol*, Mar. 2003, <http://www-106.ibm.com/developerworks/library/ws-rm/>.
- [7] R. Chinnici, et al., *Web Services Description Language (WSDL), Version 2.0*, Mar. 2001, <http://www.w3.org/TR/wsdl>.
- [8] A. Cockcroft, サービスレベルの定義とインタラクション, Apr. 1999, <http://jp.sun.com/blueprints/0499/method1.pdf>.
- [9] M. Gudgin, et al., *SOAP, Version 1.2*, Jun. 2003, <http://www.w3.org/TR/soap12-part1/>.
- [10] T. Kawamura, et al., *Preliminary Report of Public Experiment of Semantic Service Matchmaker with UDDI Business Registry*, Proc. ICSOC (Service-Oriented Computing) 2003, LNCS 2910, Springer, 2003, pp.208-224.
- [11] 中村一仁, 柘植亮人, 信頼性を用いた Web サービスの動的連携に関する研究, 南山大学数理情報学部情報通信学科卒業論文, Jan. 2004.
- [12] H. F. Nielsen and S. Thatte, *Web Services Routing Protocol (WS-Routing)*, Oct. 2001, <http://msdn.microsoft.com/ws/2001/10/Routing/>.
- [13] B. Orriens, et al., *Model Driven Service Composition*, Proc. ICSOC (Service-Oriented Computing) 2003, LNCS 2910, Springer, 2003, pp. 75-90.
- [14] P. F. Pires, et al., *Building Reliable Web Services Compositions*, Proc. WS-RSD (Int'l Workshop on Web Services-Research, Standardization, and Deployment) '02, LNCS 2593, Springer, 2002, pp. 59-72.
- [15] M. Scannapieco, et al., *Data Quality in e-Business Applications*, Proc WES 2002 (Int'l Workshop on Web Services, E-Business, and the Semantic Web), LNCS 2512, Springer, 2002, pp.121-138.
- [16] 嶋本 正ほか, *Web サービス完全構築ガイド*, 日経 BP 社, 2001.
- [17] Y. Wang and E. Stroulia, *Semantic Structure Matching for Assessing Web-Service Similarity*, Proc. ICSOC (Service-Oriented Computing) 2003, LNCS 2910, Springer, 2003, pp. 194-207.