

## 価値に基づく Web サービスの動的連携と評価

中村 一仁† 青山 幹雄‡

南山大学大学院 数理情報研究科† 南山大学 数理情報学部 情報通信学科‡

本稿では、Web サービスの動的連携を目的として、Web サービスを価値交換としてモデル化する方法とモデル記述言語である VSDL(Value-based Service Description Language)を提案する。価値に基づく動的連携を実現する技術として、価値モデルと価値ブローカを提案する。価値モデルは、サービスの品質特性を包括的に定義する。価値ブローカは、価値モデルに基づくサービスの動的連携と価値保証サービスを提供する。サービス選択に価値を用いる有効性を確認するため、価値ブローカのプロトタイプを実装し、三つの辞書サービスを用いて評価した。

## Value-Based Dynamic Composition and Evaluation of Web Services

Kazuto Nakamura† Mikio Aoyama‡

†Graduate School of Mathematical Sciences and Information Engineering, ‡Faculty of Mathematical Sciences and Information Engineering, Nanzan University

This article proposes VSDL (Value-based Service Description Language) as a service description language for composing services at a high level of abstraction. To realize dynamic service composition using VSDL, we propose a value model of Web services and architecture of value-added service brokers. The value model comprehensively defines quality of Web services. The value-added service broker provides value-based service selection, composition and assurance of services. We implemented a prototype of value-added service broker and demonstrated the effusiveness of a value-based service composition using three dictionary services.

### 1. はじめに

SOA(Service-Oriented Architecture)[1]は、ソフトウェアコンポーネントをサービスとして定義し、大規模システムを柔軟に構築できる技術として注目されている。なかでも Web サービスは基盤技術(SOAP[10], UDDI[7], WSDL[6])の成熟により普及しつつある。

しかし、現在の技術では、実行時に柔軟にサービスを組み替えることは困難である。ユーザにとって最適なサービスを自動的に連携し、要求を満たす複合Webサービスを生成して品質保証を行う技術が必要である。

本稿では、Web サービスへの入出力データを価値として抽象化し、サービスを価値交換としてモデル化して定義する言語として VSDL (Value-based Service Description Language)を提案する。これを実現する技術として、価値モデルと価値ブローカを提案する。価値モデルとは、サービス選択に必要な品質属性をモデル化したものであり、VSDL記述において価値提供と価値要求のテンプレートとして利用される。価値ブローカは、価値モデルを評価しサービスを動的連携させて、複合 Web サービスをサービスリクエストへ提供する。価値ブローカでサービス実行をモニタリングすることにより、価値保証サービスが提供できる。最後に、価値ブロー

カのプロトタイプを Java を用いて実現し、サービス選択に価値を用いることの有用性を三つの辞書システムを用いて評価した。

### 2. Web サービスの提供モデル

#### 2.1. 5段階の要求の連鎖によるサービス提供

従来の Web サービス技術では、サービス選択は Web サービスとして実現されたプロビジョニング[5]の機能特性に基づいている。しかし、サービスの特徴はプロビジョニングだけではなく、多くの要因に依存していると考えられる。そこで、Web サービスの提供を、図1に示す5段階の要求の連鎖としてモデル化する。

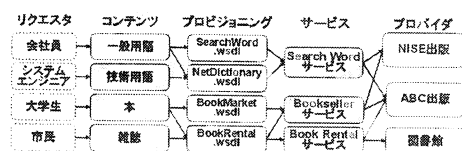


図 1 Web サービスの提供モデル

本研究では、ユーザ要求を、サービスをおこなった結果入手可能な商品や情報などのコンテンツに対する要求であると考え、コンテンツはサービスの提供

手段であるプロビジョニングにより提供されており、サービスはプロバイダによって提供されている。

このモデルにより、サービス選択のプロセスは、コンテンツの選択、プロビジョニングの選択、サービスの選択、プロバイダの選択となり、それぞれの要素に評価尺度を定義することで、サービスの提供要素を総合的に評価してサービス選択が実現できる。

## 2.2. サービスの動的利用と動的連携

複数の Web サービスを連携させて利用するには、以下の四つのステップが必要と考える。

- 1) サービスの検索と発見: サービスリクエストが期待するサービスを検索し発見する。UDDI によりサービスの機能特性からサービスの検索と発見ができる。
- 2) サービス選択: 発見したサービスの候補から、サービスリクエストの要求にあったサービスを選択する。
- 3) サービス組み合わせ: 一つのサービスでサービスリクエストの要求に応えられない場合には、複数の Web サービスを連携し、複合サービスを構築する。
- 4) サービスへの接続と利用: サービス提供者の URL を取得し、サービスへ接続して利用する。

この四ステップのうち、Web サービスの特徴として一般に言われる「動的」とは、四つ目のステップを実行時に行うことを指す。本研究ではこれを動的利用と定義する。一方、本研究が目指す動的連携は、四つのステップをすべてサービスの利用時に行うことと定義する。

## 3. Web サービスの動的連携の問題点

サービス連携の四つのステップのうち、発見と利用のステップは Web サービスの基盤技術により実現可能である。そこで、Web サービスをシステムが動的連携し利用する問題点として以下のものが挙げられる。

### 3.1. 非機能特性によるサービス選択の必要性

Web サービスの動的連携には、リクエストの要求するサービスを発見し連携する基準が必要である。この要求基準には、機能特性と非機能特性の二つがある。機能特性は、サービスのインタフェースや UDDI により定義できる。一方、非機能特性はサービスのプロビジョニングやプロバイダの属性などに依存し多様であると考えられる。非機能特性の定義は現在の技術では困難である。そこで、非機能特性をモデル化する新たな方法と、リクエストの要求するサービスを自動的に選択し、利用する仕組みが必要である。

### 3.2. 高い抽象度でのサービス記述の必要性

SOA によるシステム開発プロセスでは、要求分析工程においてビジネスプロセスを導出し、そのビジネスプロセスを実現する Web サービスを組み合わせるシステ

ムを構築する。しかし、組み合わせ時に参照される WSDL によるインタフェース記述は、ビジネスプロセスに比べ抽象度が低い。

Web サービスをシステムが動的に組み合わせる場合には、抽象度の違いを吸収する必要がある。そこで、高い抽象度で Web サービスを定義し、ビジネスプロセスからサービス抽出を行う技術が必要である。

### 3.3. Web サービスの品質保証の必要性

動的連携された Web サービスをビジネスで利用するには、サービスの提供経路を総合的に評価する仕組みが必要である[3]。Web サービスの価値保証は図1のサービス提供モデルに基づき次の四つに分類できる。

- 1) サービスプロバイダの保証: Web サービスにはインターネットの匿名性と分散性により信頼性の問題がある。プロバイダでのセキュリティや信頼性を確保する新たな仕組みが必要である。
- 2) サービスの保証: 複数のサービスを連携し利用する場合、サービス品質は SLA(Service Level Agreement)により保証している。しかし、SLA は接続相手を動的に変更することを想定していないため、動的連携で用いるのは困難である。
- 3) 提供手段の保証: Web サービスの提供手段を保証する既存の仕様として、メッセージの暗号化とプロバイダ認証を行う WS-Security[12]や送受信メッセージの順序保証を行う WS-Reliable Messaging[4]などがある。しかし、保証対象は通信路であり、ソフトウェアシステムは保証されない。
- 4) コンテンツの保証: コンテンツに対しても品質や信頼性の保証が必要である。しかし、現在の技術では品質の保証は困難である。ユーザの品質要求を評価し、コンテンツの品質保証を行う技術が必要である。

## 4. 価値に基づくサービスの動的連携

### 4.1. 価値交換によるサービス連携アプローチ

SOA では、サービスを利用者にとって意味のあるソフトウェアコンポーネントとして定義している。サービスへの入出力も意味のあるデータであると考えられる。サービスへの入出力データを価値として抽象化し、入力データを価値要求、出力データを価値提供として定義する。サービスは、要求した価値を提供した価値に交換する価値交換の活動としてモデル化できる(図2)。

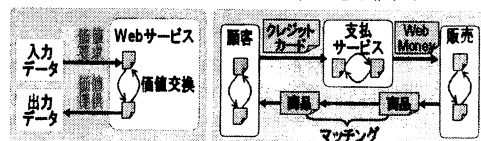


図2 価値交換によるサービス定義とサービス連携

サービスの利用はサービスリクエストとサービスプロバ

イダ間の価値提供と価値要求のマッチングとして定義できる。さらに、サービス連携は価値交換の連鎖として定義できる。

#### 4.2. 価値モデルと価値ブローカ

価値交換によるサービス連携の実現には、サービスを総合的に評価する基準が必要である。サービスを包括的に価値として定義し、価値に基づくサービスの動的連携と品質保証を実現する価値モデルと価値ブローカを提案する。

- (1) **価値モデル**: サービスの機能特性と非機能特性を価値として総合的に評価するモデル。図1のサービスの提供モデルに基づき価値モデルは四つの提供要素ごとに定義する。
- (2) **価値ブローカ**: 価値モデルを評価し、価値交換に基づくWebサービスの動的連携とサービス結果の価値保証を行う仲介サービスを提供する。

サービスの特性を登録し検索する方法として UDDI がある。しかし、価値は図 1 に示した要求の連鎖として登録する必要があるため、UDDI のデータ構造は適さない。価値モデルを格納する価値レポジトリを提案する。価値レポジトリは価値ブローカ上に配置する。

#### 4.3. 価値に基づく Web サービスの利用の流れ

価値に基づく Web サービスの利用は登録フェーズと利用フェーズに分類でき、利用の流れを図 3 に示す。

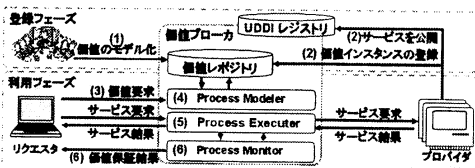


図 3 価値モデルのモデル化と利用

##### 1) サービスの価値のモデル化

サービス選択に有効な価値はアプリケーションドメインやプロバイダの業種により異なる。そこで、価値のスコップを名前空間として定義し、スコップ内で有効な価値を価値要素として抽出し、価値モデルの構造や意味を定義する。価値モデルは価値レポジトリに登録する。

##### 2) サービス定義と価値モデルのインスタンス化

プロバイダは次の手順でサービスを公開する。1) サービスのインタフェースを定義し UDDI へ登録する。2) サービスの価値を定義し価値ブローカへ申請する。3) 価値ブローカは価値を評価し価値レポジトリへ登録する。

##### 3) 価値要求の作成

リクエストは価値モデルに基づきユーザのコンテキストに応じた価値要求と価値提供を作成し、価値ブローカにサービス要求として送信する。

##### 4) 複合サービスの生成と提供

価値ブローカは価値レポジトリと UDDI レジストリを検

索しサービスリクエストの価値要求を満たすサービスを選択し、複合サービスを生成して提供する。

##### 5) サービス要求の生成と実行

サービスリクエストはサービス要求を生成し価値ブローカに送信する。価値ブローカは要素サービスを起動するためプロバイダにサービス要求を行い、サービス結果を統合してサービスリクエストに返す。

##### 6) 複合 Web サービスの価値保証

価値ブローカはリクエストとプロバイダ間で交換されるメッセージを監視し価値を評価する。これによりサービスリクエストに対して価値保証サービスを提供できる。

## 5. 価値モデル

本研究では、価値モデルの定義と利用にあたり次の手順で、価値モデルを定義した(図 4)。

- 1) **価値モデルの要求条件の整理**: 価値モデルが満たさなければならない条件を整理する。
- 2) **価値メタモデルの定義**: 価値モデルのフレームワークとして、価値モデルの構造化方法や評価方法を定義した価値メタモデルを定義する。
- 3) **価値のモデル化と記述法の提案**: サービスの価値のモデル化手法と構造化手法を考え、価値モデルとして定義するための記述法を提案する。
- 4) **価値インスタンスの利用**: 価値ブローカで価値交換の連鎖として Web サービスの動的連携を実現するため、価値の利用法について提案する。

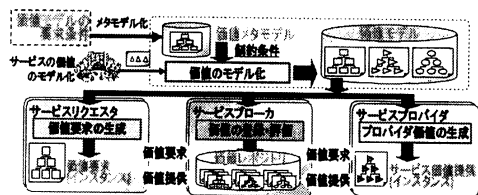


図 4 価値モデルの定義と利用法

##### 5.1. 価値モデルの要求条件

最適サービスの選択と価値保証の実現には、価値モデルは次の要求条件を満たす必要がある。

##### 1) サービスの価値の抽象化:

サービス選択には様々な要因が関係する。サービスの価値を抽象化し評価尺度を定義することで、価値ブローカで価値によるサービス選択と価値保証ができる。

##### 2) 価値要素間の関係を形式的に表現可能

価値要素は相互に複雑に関係しうる。価値要素とその間の関係をモデル化し、XML を用いて価値モデルを形式的に記述する必要がある。

##### 3) ソフトウェアによって評価可能

価値モデルはサービスの実行時に価値ブローカによって評価される。よって、価値要素には段階的な評価値を定義し、共通の尺度にマッピングして、ソフトウェアによって体系的に評価できる必要がある。

#### 4) 変化するユーザ要求に対応可能

サービスリクエストの価値要求は、ユーザの置かれている場所や時刻などのコンテキストにより変化する。変化するユーザ要求をリアルタイムに捉え、価値モデルで評価できる必要がある。

#### 5.2. 価値のメタモデル

価値と価値の利用に影響を与える要因を定義するため価値メタモデルを図5に示すように定義する。

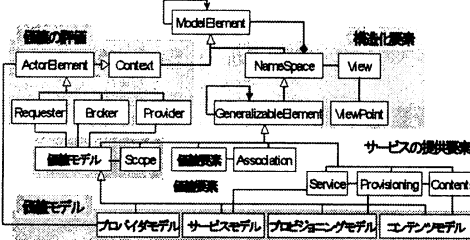


図5 価値メタモデル

#### (1) 四つの価値モデル

図1に示すサービスの提供モデルに基づきサービス選択の評価尺度となる四つの価値モデルを定義する。

- 1) プロバイダ価値モデル: サービスプロバイダのプロファイルを定義。例えば、プロバイダの信頼性や評判、経営状態が有効な価値要素となり得る。
- 2) サービス価値モデル: サービス品質を定義。例えば、商品のデリバスピードやサービスの評判。
- 3) プロビジョニング価値モデル: サービスの提供方法の属性をモデル化。例えば、ソフトウェアシステムの安全性や操作性が有効な価値要素となり得る。
- 4) コンテンツ価値モデル: サービスの結果リクエストが得られるコンテンツの品質特性をモデル化。ショッピングサービスでは、商品の品質や価格。

#### (2) 価値要素と価値モデルの構造化要素

価値モデルは価値要素と要素間の関連により定義する。価値モデルを形式的に定義するため、価値要素を異なるステークホルダの視点から分類する名前空間と価値要素間の関係を関連と継承を用いて表現する。

また、価値モデルのスコープを定義し、他の価値モデルとの関係を考慮する必要がある。この関係として集約関係や継承関係を用いる。例えば、図書販売サービスはショッピングサービスの特殊化として定義できる。

#### (3) コンテキストに応じた価値評価

価値モデルの登録や評価を行うサービスプロバイダやサービスリクエストなどのアクタは、様々なコンテキストを持つと考えられる。例えば、サービスリクエストのコンテキストに影響を与える要因として、サービスの利用時間や利用場所、ユーザの好みなどが挙げられる。これらのコンテキストをリアルタイムに捉え価値として表現し、価値モデルの評価を行う。

#### 5.3. 価値モデル

価値要素とその間の集約関係に着目し、価値モデルの構造を木構造として定義した。価値モデルの表記法として、UMLのクラス図を拡張した図記述と、XMLによるテキスト記述としてVM-XML(Value Model XML)を提案する。商社の価値モデルの図的記述の例を図6に示す。

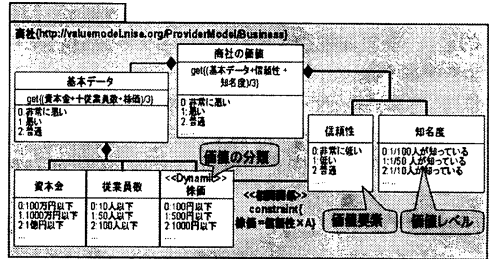


図6 価値モデルの例

価値モデルのスコープはパッケージで表現し、価値要素はクラスとして表現する。ある価値要素はより詳細な特性を集めたものとして考えられるため、価値モデルの木構造を集約関係で表現する。価値要素は、価値レベルとして品質属性の尺度を持ち、分類はステレオタイプで記述する。また、木構造の節点となる価値要素は操作「get()」を持ち、下位の価値要素の価値レベルをボトムアップ的に評価して返す。

また、価値モデル記述を行うグラフィカルエディタをEclipseプラグインとして開発した(図7)。このツールにより、価値モデルの図的表現からVM-XML記述が生成可能である。開発規模は、76クラス7,962行である。

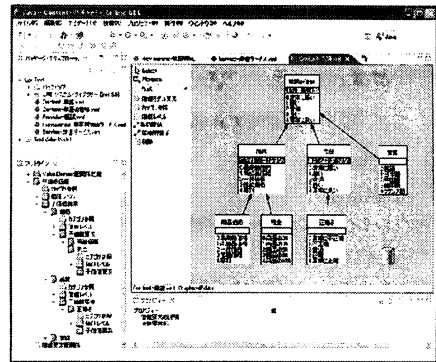


図7 価値モデルのグラフィカルエディタ

価値モデルのVM-XML記述は価値ブローカ内の価値レポジトリに、価値要求と価値提供のテンプレートとして格納される。

#### 5.4. 価値インスタンス

価値モデルで定義されたサービスへの入出力価値を用いて、サービスリクエストやサービスプロバイダは、価値インスタンスを生成し、価値ブローカでのWebサービ

スの動的連携と価値保証に利用する。価値インスタンスは、価値要求と価値提供に分類できる。

### 5.4.1. 価値要求

価値要求は、利用するサービスの価値提供に対する要求であり、価値レベル値に条件を記述することにより定義する。条件の水準として、次の二つを定義する。

- 1) 当たり前価値要求: 必ず満たさなければならない条件であり、条件を満たさないサービスは利用候補から除外される。
- 2) 魅力的価値要求: 満たすことを期待する条件であり、サービス利用の優先度を決定するのに利用する。

図6の価値モデルの価値要求の例を図8に示す。

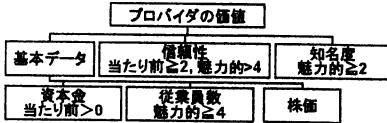


図8 図6の価値モデルに対する価値要求の例

この例の信頼性の価値要素では、価値レベル値が2以上であることが必須条件であり、4以上のサービスを期待していることが記述されている。

### 5.4.2. 価値提供

価値提供はサービスが提供する価値を定義するものであり、木構造の葉の価値要素の価値レベル値を指定して定義を行う。木構造の節の価値要素は、価値モデルで定義済みの get()メソッドにより、ボトムアップ的に価値レベル値を計算により求めることが可能である。価値提供の例を図9に示す。

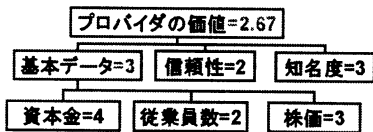


図9 図6の価値モデルに対する価値提供の例

この価値提供の例を図8に示した価値要求で評価すると、当たり前価値要求の信頼性と資本金は条件を満たしているため、この価値提供を持つサービスは利用候補となる。しかし、魅力的価値要求の信頼性と従業員数は条件を満たしていないため、これらを満たす価値提供を持つサービスが他にあった場合にはサービス利用の優先順位は下がる。

## 6. 価値交換によるサービスの動的連携

### 6.1. 価値に基づくWebサービス記述(VSDL)

価値交換の連鎖としてサービスの動的連携を実現するため、価値に基づくWebサービス記述言語としてVSDL (Value-based Service Description Language)を提案する。e<sup>3</sup>Value Model[9]の表記法

を拡張したVSDLの図的記述を図10に示す。

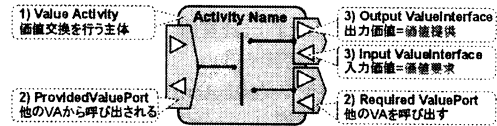


図10 VSDLの図的記述

### 1) Value Activity

Value Activityは、図1に示すWebサービスの提供モデルのプロビジョニング、サービス、プロバイダを抽象化した価値交換を行う主体であり、Value Portのセットを持つ。Value Portの違いにより、一つのプロビジョニングは複数のVAを持つ可能性がある。

### 2) Value Port

他のValueActivityと価値の交換を行うValueInterfaceの集合である。他のValueActivityから起動されるProvided Portと他のValueActivityを起動するRequired Portの2種類がある。

### 3) Value Interface

Webサービスの入出力をコンテンツの価値で定義する。サービスへの入力価値要求としてInput Value Interfaceで定義し、サービスからの出力は価値提供としてOutput Value Interfaceで定義する。

VSDLのXML記述の例を図11に示す。

```
<?xml version="1.0" encoding="UTF-8"?>
<ValueActivity id="uuid:aaaa-bbbb" name="TestActivity"
provisioningID="uuid:aaaa-bbbb-cccc-dddd-eeee-0000"
xmlns="http://ValueActivity.nise.org"/>
  <ProvidedInterfaceSet>
    <ValuePort>
      <InputValueInterface requestID="uuid:aaaa-cccc"/>
      <OutputValueInterface provideID="uuid:aaaa-dddd"/>
    </ValuePort>
  </ProvidedInterfaceSet>
  <RequiredInterfaceSet>
    <ValuePort>
      <OutputValueInterface provideID="uuid:aaaa-eeee"/>
      <InputValueInterface requestID="uuid:aaaa-0000"/>
    </ValuePort>
  </RequiredInterfaceSet>
</ValueActivity>
```

図11 VSDLのXML記述

### 6.2. 価値交換による複合Webサービスの自動生成

価値ブローカは、価値レポジトリ内に格納されたVSDL記述を参照し、図12の手順で価値交換の連鎖を実現するValue Exchangeプロセスを生成する。

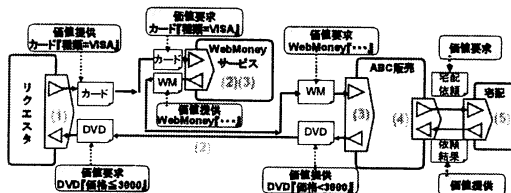


図12 複合Webサービスの生成手順

### (1) サービスリクエストの ValueActivity 定義

サービスリクエストからの価値要求と価値提供を受け取り Required Value Port を持つサービスリクエストの ValueActivity を定義する。

### (2) 価値要求を持つ ValueActivity の検索

価値レポジトリを検索しサービスリクエストの価値要求を満たす価値提供を持つ ValueActivity を検索する。図の例では、DVD を価値提供に持つ ValueActivity を検索しサービス利用の候補とする。

### (3) 価値要求の条件による絞り込み

(2)の候補 ValueActivity の中から、価値要求の当たり前価値条件による絞り込みと、魅力的価値条件による ValueActivity の優先度による並び替えを行う。

### (4) 価値提供との比較

(2)で絞り込まれた ValueActivity の価値要求とリクエストの価値提供をマッチングする。マッチングが不成功の場合は(2)から(4)の手順を繰り返し価値交換のループを完成させる。

### (5) Provided Port の解決

(4)で完成したループ内で Provided Value Port を持つ ValueActivity があつた場合には、(2)から(4)の手順を繰り返し完全なループを完成させる。

## 7. 価値ブローカのアーキテクチャ

Web サービスプラットフォームを拡張し、価値を評価する価値ブローカのアーキテクチャを図 13 に示す。

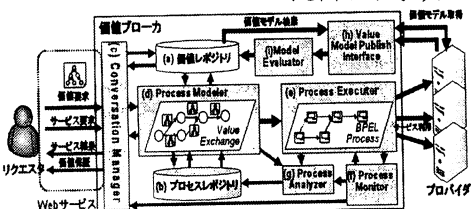


図 13 価値ブローカのアーキテクチャ

#### (a) 価値レポジトリ

VM-XML で記述された価値モデルと VSDL で記述された価値によるサービス記述を格納する木構造の XML データベースである。

#### (b) プロセスレポジトリ

利用履歴による最適サービスの選択精度向上と Value Exchange プロセスの再利用のため、実行済みプロセスを四階層に分けて格納する。

#### (c) Conversation Manager

一つのリクエストの価値要求で利用するサービスが一意に決定しない場合があるため、リクエストと価値ブローカ間の対話をサポートする。

#### (d) Process Modeler

Value Exchange プロセスを自動生成する。Process Modeler は、以下の三つのデータベースを検索できる。

1)サービスの機能特性を検索する UDDI レジストリ。2)

価値検索を行う価値レポジトリ。3)プロセスの再利用に使うプロセスレポジトリ。

#### (e) Process Executer

Value Exchange プロセスから実行可能な WS-BPEL[2]プロセスを生成し、BPEL エンジンを用いて実行する。

#### (f) Process Monitor

サービスの価値保証の実現のため、Process Executer で実行される複合 Web サービスでやり取りされる価値を監視する。

#### (g) Process Analyzer

複合サービスのプロセスを Process Monitor での価値の評価に基づき分析しプロセスレポジトリに利用履歴として格納する。

#### (h) Value Model Publish Interface

プロバイダに対して価値提供や価値要求の検索や登録サービスを提供するインタフェースであり、Web サービスで実現する。

#### (i) Model Evaluator

プロバイダにより申請された VSDL 記述を価値ブローカの担当者が評価するためのユーザインタフェースを提供する。

提案した価値ブローカを Java, Apache Axis, XIndex, BPEL4J, Oracle BPEL を用いて実装した。開発規模は、46 クラス, 6,810 行である(図 14)。

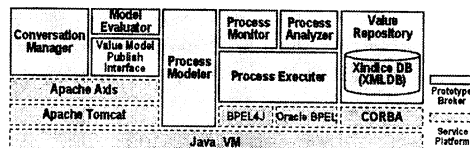


図 14 価値ブローカの実装とプラットフォーム

## 8. プロトタイプによる評価

価値による Web サービスの動的連携の有用性を評価するため、三つの辞書サービスを用いて評価した。

#### 1) NetDicV08

三省堂が提供する辞書サービスであり、英和、和英、国語辞書から 19 万語が利用できる。

(<http://btonic.est.co.jp/NetDic/NetDicv08.aspx>)

#### 2) Insider's Computer Dictionary (ICD)

@IT が運営する辞書サービスであり、IT の専門用語全般が検索できる。

(<http://www.iwebmethod.net/icd1.0/icd.aspx>)

#### 3) Simple Word Book

研究室で開発した辞書サービスであり、ソフトウェア工学の専門用語が検索できる。

このうち、NetDicV08 の英和と和英は日本語と英語の言語の変換を行っていると考えられる。その他は、単語の意味の深化を目的としている。そこで、「単語」と

「単語の意味」の二つのコンテンツの価値モデルを定義し VSDL による記述を行った。これを図 15 に示す。

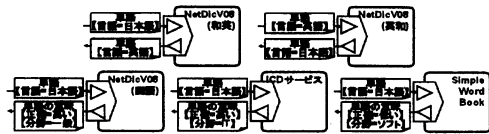


図 15 三つの辞書サービスの VSDL 記述

これらの Web サービスを価値交換に対応させるため C#を用いてラッパーブローカを作成し、価値ブローカを用いて動的連携させ評価した。開発規模は、1,708 行である。図 16 に連携の様子を示す。

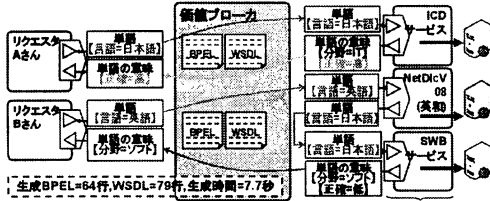


図 16 価値による辞書システムの動的連携

図 17 は、三つの辞書サービスを使って「apple」を検索した結果である。NetDicV08 サービスからはりんごの意味が検索できたが、他のサービスからは Apple Computer の意味が検索できた。同じ機能特性を持つサービスであっても、その分野によって意味の深さや正確性が異なることが確認できた。

そこで、価値モデルによりプロバイダの信頼性やサービス結果の単語の分野や意味の正確性の評価尺度を定義し、価値によるサービス選択を行うと、同一辞書サービスでもユーザの利用状況や職業などのコンテキストから、ユーザに適した辞書が提供できた。価値モデルにより辞書サービスの利用価値が高まったといえる。

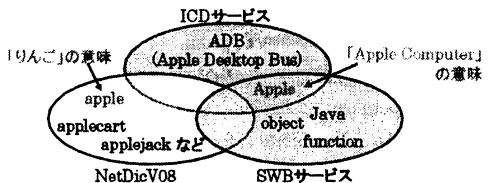


図 17 三つの辞書を使った Apple の検索結果

また、価値ブローカに単語の意味を要求し、英語の単語を提供すると、価値交換の連鎖を自動的に生成し NetDicV08 の英和辞書サービスと SWB サービスを連携できた。リクエストはサービス連携を意識せずに、英単語から SWB サービスに登録された意味が検索できる。価値による Web サービスの動的連携により、SWB サービスの利用範囲が拡大したといえる。

なお、サービスを動的連携させるために自動生成された複合 Web サービスの WS-BPEL 文書は 64 行、WSDL 文書は 79 行であり、Pentium 4 3GHz のマシンで 10 回実行した生成時間の平均は 7.8 秒である。

## 9. 価値のモデル化についての考察

サービスプロバイダの信頼性や知名度、サービスの品質特性などのサービス選択に影響を与える要因は、業種やアプリケーションドメインなどのカテゴリによって異なっていることが考えられる。また、これらの要因は互いに複雑に関係し合っていることが考えられる。よって、様々な視点からのモデル化が可能である。視点の例として、価値要素間の関係に着目したグラフによるモデル化、価値の提供の視点から見たサービスの魅力的な価値をアピールできるモデル化がある。

しかし、価値モデルにより定義された価値は、サービスリクエストによってサービス要求の一環として利用される。ユーザ要求は人によって要求の正確性や深さが異なると考えられる。例えば、図 6 に示した価値モデルで、あるサービスリクエストは資本金や株価などの詳細な要求を望んでいるかもしれないが、他のサービスリクエストは、細かい要求には興味がなく基本データとしての要求で十分である場合がある。よって、価値モデルをユーザ視点から考え、価値要素の木構造としてモデル化したことにより、サービスリクエストが必要としている抽象度での価値の要求が可能となり、サービスリクエストが期待するサービスの発見と提供が容易になった。

## 10. 関連研究

Web サービスをシステムが動的連携する技術は様々な技術が研究されている[11]。

サービスの非機能的特性を使ったサービスの動的連携の研究として、Zengらの研究[13]がある。この研究では、サービスの品質特性として、費用、サービス時間、評判、成功確率、可用性の六つを定義している。そして、品質特性を利用したサービスの動的連携の方法として、Local Optimization と Global Planning の手法を提案している。この研究により、サービスの品質特性を利用したサービス連携が可能となった。

しかし、サービスの品質特性はサービスプロバイダの信頼性やコンテンツの品質などが複雑に関係していると考えられるが、品質特性のモデル化は考慮されていない。価値モデルによる品質特性とその間の関係のモデル化が必要である。また、この研究では、複合サービスのテンプレートを用いて品質属性によるサービス選択を実現している。しかし、テンプレートが無いユーザ要求に対応できないため、本研究で提案した価値交換の連鎖によるサービスの動的連携手法が必要である。

また、サービスの意味情報を定義しサービスの動的連携を行う研究として、Fujiiらの研究[8]がある。この研究では、サービスの意味情報を定義する CoSMoS (Component Service Model with Semantic)を提案している。CoSMoS では、Semantics Domain, Data Type Domain, Logic domain, Component

domain の四つのドメインを用いて意味定義を行い、この意味情報を SeGSeC (Semantic Graph based Service Composition)により評価することで、サービスの実行パスの生成と意味の整合性のチェックを実現している。

しかし、サービスの意味定義は RDF やオントロジーを利用しており、非機能要求とは対応していない。また、サービスの動的連携にはサービスの意味定義だけでなく、プロバイダの信頼性など提供要因の非機能特性が重要であり、本研究で提案した価値のモデル化と価値によるサービス選択が必要である。

一方、要求工学分野では、Gordijn らの研究[9]がある。この研究では、ネットワーク上のステークホルダ間の活動を価値の交換としてモデル化し、ビジネスモデルの明確化と高い抽象度の要求条件の引き出しを実現している。これを Web サービスの動的連携に用いる場合には、プロビジョニングの品質属性の考慮と、モデルから Web サービスへのマッピング技術が必要である。そこで、本研究で提案した Web サービスの提供モデルによるサービスの提供要因のモデル化と価値ブローカによるサービス連携フレームワークが必要である。

## 11. 今後の課題

以下に、価値モデルを用いた Web サービスの動的連携における主要な課題を挙げ、今後検討する。

- (1) 多次元価値モデル: 現在の価値モデルは価値要素の集約関係の観点でモデル化した。しかし、サービスの価値は様々な分類方法が考えられるため、多次元価値のモデル化と評価方法が必要である。
- (2) ユーザ要求の価値モデルへのマッピング: コンテキストにより変化するユーザの価値要求をリアルタイムに捉え、価値モデルにマッピングし評価する方法の提案が必要である。

## 12. まとめ

本研究は、Web サービスの動的連携を目標とし、サービスを価値交換としてモデル化し定義する言語として VSDL (Value-based Service Description Language)の提案を行った。また、これを実現する技術として価値モデルと価値ブローカの提案を行った。価値モデルとは、サービスの価値を包括的に表現するものであり、価値要求と価値提供のテンプレートとして利用される。価値ブローカは、価値に基づく複合サービスの自動生成と価値保証サービスの提供を行う。最後に、三つの辞書システムを用いてサービス選択に価値を用いることの有用性の評価を行った。

## 参考文献

- [1] G. Alonso, et al., *Web Services*, Springer, 2004.
- [2] T. Andrews, et al., *Business Process Execution Language for Web Services (WS-BPEL)*, 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] R. Bilorusets, et al., *Web Services Reliable Messaging Protocol (WS-ReliableMessaging)*, Feb. 2005, <http://chemas.xmlsoap.org/ws/2005/02/rm/>.
- [5] T. Chandu, *Use Web Services Provisioning to Control Access, Usage, and Billing on Your Site*, Dec. 2002, <http://msdn.microsoft.com/msdnmag/issues/02/12/WebservicesProvisioning/>.
- [6] R. Chinnici, et al., *Web Services Description Language (WSDL)*, Version 2.0, Aug. 2005, <http://www.w3.org/TR/wsdl20>.
- [7] L. Clement, et al., *UDDI*, Version 3.0.2, Oct. 2004, <http://www.oasis-open.org/committees/uddi-spec/doc/spec/v3/uddi-v3.0.2-20041019.htm>.
- [8] K. Fujii and T. Suda, Dynamic Service Composition Using Semantic Information, *Proc. ACM ICSOC (Service-Oriented Computing) 2004*, Dec. 2004, pp39-48.
- [9] J. Gordijn, et al., E-Business Value Modeling Using the E3-value Ontology, W. Currie (ed.), *Value Creation*, Elsevier, 2004, pp. 98-127.
- [10] M. Gudgin, et al., *SOAP*, Version 1.2, Jun. 2004, <http://www.w3.org/TR/soap>.
- [11] T. Kawamura, et al., Preliminary Report of Public Experiment of Semantic Service Matchmaker with UDDI Business Registry, *Proc. ICSOC (Service-Oriented Computing) 2003*, LNCS, Vol. 2910, Springer, 2003, pp.208-224.
- [12] A. Nadalin, et al., *Web Services Security (WS-Security)*, Ver. 1.0, Mar. 2004, <http://docs.oasis-open.org/wss/2004/01/oasis-200401-ws-soap-message-security-1.0>.
- [13] L. Zeng, et al., QoS-Aware Middleware for Web Services Composition, *IEEE Trans. on Software Engineering*, Vol. 30, No. 5, May. 2005, pp. 311-327.