

複合 Web サービスのモデルベース開発方法と支援環境の提案

福永 遼重† 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‡

1. はじめに

本研究では、WS-BPEL(Web Services Business Process Execution Language: 以下 BPEL と略記)[1]を用いて、共通のモデル[2]に基づく異なる非機能特性の Web サービスを動的に連携させて、リクエストの非機能要求も満たす複合 Web サービスの構築方法とその支援環境を提案し、例題を用いて評価した結果を示す。

2. 複合 Web サービス開発へのアプローチ

2.1. 属性による類似サービスのモデル定義

異なる非機能要求のサービスを組み合わせるために複数のサービスに対してインタフェースの整合をとる共通のモデルを提案する。サービス間で各 WSDL のインタフェースにモデルとの対応付けを定義するメタ情報を属性として付加する。サービス間のインタフェースの整合をとるメタ情報として WSDL にインタフェース定義の type, message, operation を新たに属性として付加する。WSDL に準拠しているため、全ての WSDL によるインタフェース定義に適用可能である。例として、サービス A, サービス B と、BPEL プロセスとの対応関係を図 1 に示す。

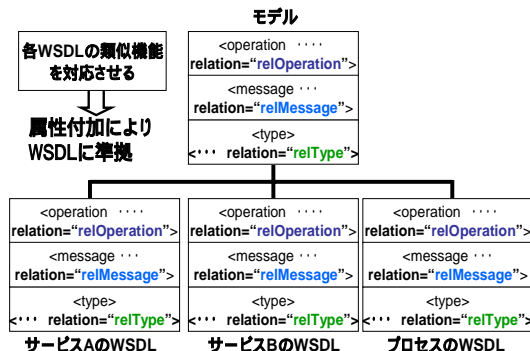


図 1: 属性付加による WSDL の対応関係

2.2. BPEL プロセス自動生成のプロセス

プロバイダがリクエストのサービス要求を受信してから BPEL プロセスを自動生成するまでのプロセスを図 2 に示す。

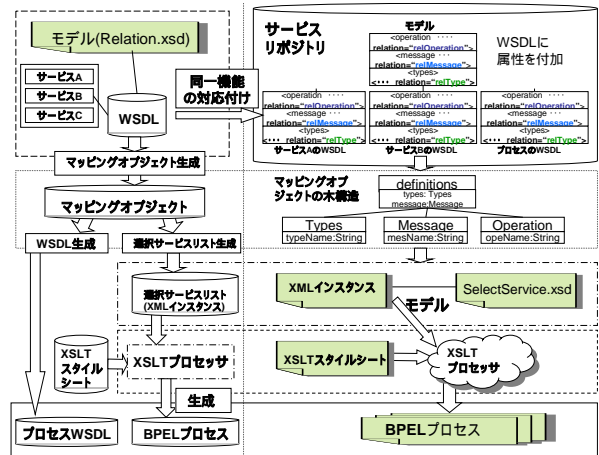


図 2: BPEL プロセスの自動生成のプロセス

3. 複合 Web サービスの構築支援モデル

BPEL プロセスを自動生成するために、選択したサービスを実行時にリンクするためのモデルと、選択したサービスから BPEL プロセスの実行フローを構築するためのインタフェースモデルを提案する。

3.1. 辞書サービスにおけるモデル化

例として、辞書サービスを用いて、選択したサービスを実行時にリンクするモデルを示す。属性による類似サービスのモデル定義から、各 WSDL の operation と入出力のデータ構造を抽出する。次に動的に複合 Web サービスを構築するために、この 2 つの要素から、図 3 に示す辞書サービスのモデルを定義する。このモデルから辞書サービスの WSDL を入力としてモデルのインスタンスを生成し、サービスリポジトリに登録する。登録したモデルのインスタンスから駆動する複合 Web サービスを自動生成する。図 3 は、Service クラスのステレオタイプで辞書サービスを指定し、子要素として 1 つ以上の operation を持つ。その子要素は入出力のデータ構造を持ち、機能間の対応はステレオタイプを用いて表現する。

A Model-Based Development Method and its Support System for Composite Web Services

† Yukishige Fukunaga, Graduate School of Mathematical Sciences and Information Engineering, Nanzan University

‡ Mikio Aoyama, Faculty of Mathematical Sciences and Information Engineering, Nanzan University

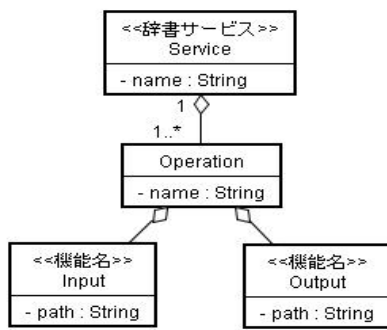


図 3：辞書サービスのモデル

3.2. モデルから BPEL プロセス自動生成

実行時に BPEL プロセスの実行フローを構築するために、選択したサービスのドメイン毎に共通のインタフェースを提供する必要がある。辞書サービスの例では、入力には単語が必要で、その出力に単語の意味が返ってくる。この入力と出力を共通のインタフェースとしてモデル化できる。辞書サービスが選択されたときは、このデータモデルから辞書サービスに対応するリクエストへのインタフェースが実現できる。定義したデータモデルを図 4 に示す。

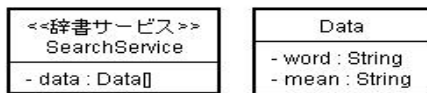


図 4：単語帳サービスのデータモデル

4. 例題による実験と評価

4.1. 辞書サービス

本研究では BPEL プロセスで扱うサービスの例として辞書サービスを利用する。使用するサービスはコンピュータ用語の ICD[3]、三省堂の辞書サービスの NetDic[4]、本研究で作成したサービスの SimpleSearch の 3 つを使用する。

4.2. 実験方法

表 1 に示す開発、実行環境上で複数の辞書サービスを並行実行する BPEL プロセスを自動生成するツールを開発した。ツールは XSLT と DOM の 2 種類で実装した。

表 1 開発、実行環境

OS	WindowsXP
CPU/メモリ	Pentium 4(2.66GHz), 1GB RAM
Web サーバ	Tomcat 5.0.28, Apache Axis 1.1
BPEL エンジン	BPEL Process Manager
Java 開発環境	J2SDK 1.4.2
XML パーサ	xerces-1_4_4
開発規模	1163 行(XSLT を用いた実装) 553 行(DOM を用いた実装)

4.3. BPEL プロセス自動生成の評価

サービスを追加する拡張性と BPEL プロセスの実行フロー変更の柔軟性の視点でツールのプロトタイプを評価する。DOM と XSLT による自動生成ツールの 2 つに対して、3 つのサービスを使用して比較をした。サービスの追加毎に生成される総コード数を図 5 に示す。DOM のツールでは 1 つのサービス追加のコード数が大きい。サービスの実行順序が動的に変更される場合、構造を柔軟に変更できない。よって DOM よりも XSLT を用いたツールのほうが優れていると言える。この結果から、本研究の BPEL プロセスの生成方法は柔軟性と拡張性があると評価できる。

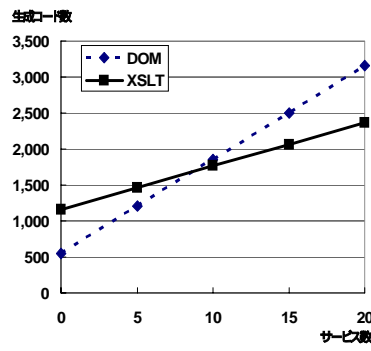


図 5 より、サービス追加数が 7 つを越えると、XSLT ツールのほうが総コード数が少なくなる。

図 5：追加サービス毎に生成される総コード数

5. まとめと今後の課題

本研究では類似機能のサービス連携に注目し、インタフェースの機能と要素間に対応させる共通のモデルに基づく BPEL プロセスの自動生成方法と、開発支援環境を提案した。

従来方法[2]では、サービスの機能間の対応付けに限られているので、サービスの非機能特性や内容に基づく動的選択、結合に対応していない。

今後、実行コンテキストやサービス品質要求に応じてサービスの動的合成を支援するモデル駆動型 Web サービス開発方法論を検討する。

参考文献

- [1] 福永 遂重, ほか, BPEL を用いた複合 Web サービスのモデルベース開発支援, 情報処理学会ソフトウェア工学研究会, Vol. 2005-SE-148, Mar. 2004, pp. 9-16.
- [2] B. Orriens, et al., Model Driven Service Composition, *Proc. of IC3SOC 2003*, LNCS, Vol. 2910, Springer, Dec. 2003, pp. 75-90.
- [3] NetDictionary ICD Web サービス仕様 1.0, <http://www.webservice.jp/netdict/index.html>.
- [4] 三省堂 デイリー コン サ イ ス, <http://www.btonic.com/ws/vs.htm>.