

複合 Web サービスのモデル駆動開発方法の提案

福永 遂重[†] 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 数理情報学部[‡]

1. はじめに

本研究では、ユーザの機能、非機能要求を満たす複合 Web サービスのモデル駆動開発方法を提案する。機能・非機能特性が表現可能で拡張性の高いサービスモデルを定義し、それに基づく複合 Web サービス構築方法とその支援環境を提案する。また例題を用いて、評価した結果を示す。

2. 関連研究

モデルに基づき Web サービスを組み合わせる研究が多く行われている[2, 3]。文献[3]では、UML Profile の QoS Profile と SPT Profile の記述方法を用いて QoS 特性を WSDL に付加したメタモデルを提案している。しかし、Web サービスの動的連携に対応していない。本研究では、サービスの機能・非機能特性を付加したモデルを提案し、動的連携に対応したモデル駆動開発方法を提案する。

3. 複合 Web サービスのモデル駆動開発

3.1. MDA による複合 Web サービス開発

MDA に基づく、サービスモデルから複合 Web サービスを構築するモデル駆動開発方法を図 1 に示す。サービスの機能・非機能特性を表現するメタモデルを定義する。以後、このモデルをサービスモデルと呼ぶ。

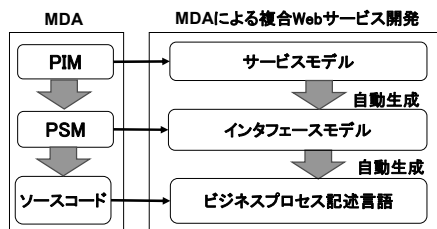


図 1 Web サービスにおけるモデル駆動開発

サービスモデルに基づき、Web サービスのインタフェースに機能、非機能特性を付加したモデルを定義する。このモデルをインタフェースモデルと呼ぶ。このモデルを用いて、機能・非機能要求に基づくサービスの選択、動的連携を可能にする。

3.2. モデル駆動開発プロセスの提案

図 2 に示すモデル駆動開発プロセスでは、モデル定義とサービス実行の 2 段階で実行する。

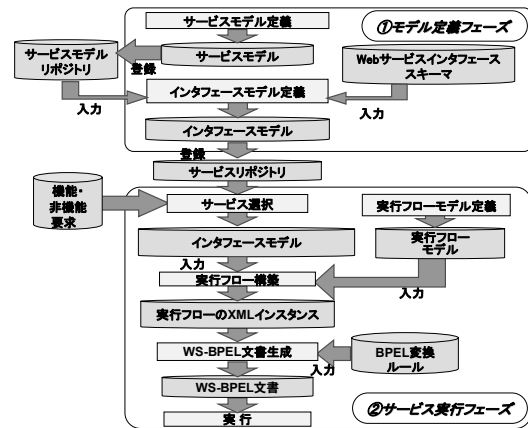


図 2 モデル駆動開発プロセス

3.2.1. モデル定義フェーズ

1) サービスモデル定義フェーズ

サービスの機能・非機能特性を定義し、サービスモデルリポジトリに登録する。

2) インタフェースモデル定義フェーズ

サービスモデルと、Web サービスインタフェーススキーマに基づき機能・非機能特性を付加したインタフェースモデルを定義し、サービスリポジトリに登録する。

3.2.2. サービス実行フェーズ

ユーザの機能・非機能要求を満たすサービスをサービスレジストリから選択し、サービス開発者が事前に定義した実行フローモデル[1]と組み合わせ、WS-BPEL を自動生成し、実行する。

3.3. サービスモデル定義

図 3 に示すサービスメタモデルに基づき、サービスモデルを定義する。サービスモデルは提供する機能と、その機能が提供するコンテンツの 2 つの特性で構成する。

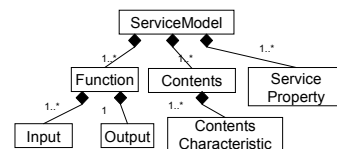


図 3 サービスメタモデル

Function は、提供する機能を抽象的に定義し、子要素として Input, Output 要素を持つ。Input と Output 要素はインタフェースの整合性を取るための型と変数を指定する。Contents は、サービスが提供する商品、情報を定義し、子要素としてその特性を定義する ContentsCharacteristic 要素を持つ。ContentsCharacteristic 要素としてコンテンツのコストに関する特性 Cost, コンテンツの

A Model-Driven Development Method 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

情報量，種類に関する特性 Type を定義する． ServiceProperty は，名前空間などを定義する．

3.4. インタフェースモデル定義

サービスモデルと Web サービスインタフェーススキーマから XML スキーマで定義されるインタフェースモデルを生成する． Web サービスインタフェーススキーマに WSDL の XML スキーマを適用し，図 4 にサービスモデルと WSDL の XML スキーマから生成した WSDL インタフェースモデルを示す． WSDL インタフェースモデルでは，サービスモデルは WSDL の XML スキーマの”Service”要素の子要素として生成される．このインタフェースモデルに基づきインスタンスを作成し，サービスリポジトリに登録する．

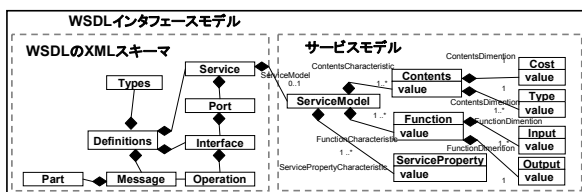


図 4 WSDL インタフェースモデル

インタフェースモデルの属性は，SPT Profile で定義されている属性 value，unit，source，type，direction を持つ． Function の value は WSDL のオペレーションを記述する． Input，Output の value は XPath に基づいて変数を記述し，unit はデータの型を記述する．

4. プロトタイプの実装と評価

4.1. プロトタイプのアーキテクチャと実装

サービスモデルとインタフェースモデルを記述するプロトタイプ ServiceModeler と，ユーザ要求を入力とし， Web サービスの選択・連携・実行を行うプロトタイプ ProcessController を開発した． ProcessController のアーキテクチャを図 5 に示す．複合 Web サービスの実装言語は WS-BPEL を用いる． ProcessController は Java で開発し，総コード数は 5004 行となった．

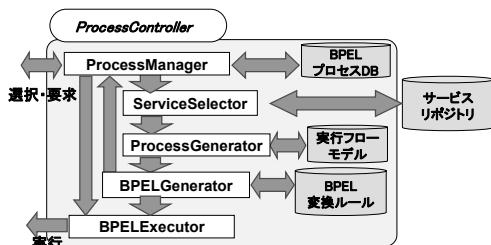


図 5 ProcessController のアーキテクチャ

4.2. 例題を用いた評価

例題として，@IT の ICD サービス，三省堂の NetDicV09 サービス，本研究で開発した SimpleSearch サービスの 3 つの辞書サービスを用いる． ServiceModeler を用いて図 6 に示す辞書サービスのサービスモデルは入力に単語，出力に単語の意味を返す検索機能を Function として定義

し，検索機能が提供する単語を Contents として定義する．このサービスモデルをサービスモデルリポジトリに登録する．定義したサービスモデルを ICD サービスに適用したインタフェースモデルの例を図 7 に示す． ServiceModeler によりサービスモデルからインタフェースモデルスキーマを自動生成し，属性の値を記述してインタフェースモデルを構築する．

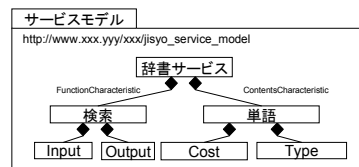


図 6 辞書サービスのサービスモデル

Function”検索”が，複数のオペレーションを組み合わせる必要がある場合，必要なオペレーションを全て ServiceModel”辞書サービス”の子要素として定義する． Contents”単語”の value 属性は辞書の分野， Contents”単語”の子要素 Type の value 属性は単語数を定義し，サービスリポジトリに登録する． NetDic， SimpleSearch サービスに対しても Function”検索”， Contents”単語”の属性を定義する． ProcessController は選択したインタフェースモデルを並行処理する WS-BPEL を生成する．

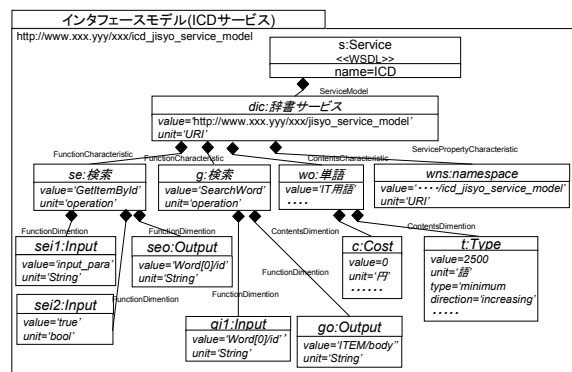


図 7 ICD サービスのインタフェースモデル

5. まとめと今後の課題

本研究では，サービスの機能・非機能特性を表現したサービスモデルとインタフェースモデル定義方法を示し，このモデルに基づき，ユーザ要求を満たすサービス選択・動的連携に対応する複合 Web サービス開発方法を提案した． 今後は，QoS 特性も考慮したサービスモデルを検討する．

参考文献

- [1] 福永 遂重，複合 Web サービスのモデル駆動開発方法と支援環境， SES2006， Oct. 2006， pp. 221-222.
- [2] K. Baina, et al., Model-Driven Web Service Development, *Proc. of CAiSE 2004*, Jun. 2004, pp. 290-306.
- [3] A. D'Ambrogio, et al., A Model-driven WSDL Extension for Describing the QoS of Web Services, *Proc. of ICWS 2006*, pp. 789-796.