

非機能特性を考慮した ESB の研究

—ESB を用いた SOA アプリケーションの開発支援—

2009SE002 足立理子 2009SE049 村上晃輝 2009SE142 黒田航平

指導教員：野呂昌満

1 はじめに

SOA に基づくシステムにおいて、サービス統合をおこない位置透過性を実現する技術として Enterprise Service Bus(以下、ESB)[1]がある。ESB はバスの概念でサービスを連携するリファレンスアーキテクチャである。利用者はバスの場所とサービス名を知っているだけで目的とするサービスを利用することができる。ESB を構成するコンポーネントは非機能特性と関連があり、これらのコンポーネントによって非機能要求が実現される。ESB を構成するコンポーネントの1つに Routing があり、実行効率や耐故障性といった要求を実現する。ESB における Routing にはルーティングアルゴリズム [2] という技術がある。ルーティングアルゴリズムは最適な経路を決定するための処理手順で、いくつかのバリエーションがありバリエーションごとに実現できる非機能要求の度合いが異なる。

ESB の Routing において、ルーティングアルゴリズムと非機能特性の対応関係が不明確という問題が挙げられる。SOA に基づくシステムにおける非機能要求の実現は、主に ESB の Routing などでおこなう。ルーティングアルゴリズムと非機能特性の対応関係が整理されていないことから、非機能要求の実現が他の非機能要求に与える影響が不明確となる。その結果、多様化する非機能要求に対応したルーティングアルゴリズムの選択が困難となる。

本研究の目的は、ESB を用いた SOA アプリケーション開発を非機能特性を考慮して支援することである。ルーティングアルゴリズムと非機能特性の対応関係を整理し、要求に適したルーティングアルゴリズムの選択手法を提案する。ルーティングアルゴリズムと非機能特性の対応関係を明確にすることで、多様化する非機能要求に柔軟に対応したルーティングアルゴリズムの選択が可能となる。ルーティングアルゴリズムの選択手法を提案することで、目的の達成を目指す。

本研究では ESB のリファレンスアーキテクチャモデルに基づいて、非機能特性とルーティングアルゴリズムの対応関係を整理する。非機能特性の整理には ISO9126[3] の Quality Model を適用する。これにより、ルーティングアルゴリズムの性質を一般的な非機能特性として整理することが可能となる。ルーティングアルゴリズムと非機能特性の対応関係の結果を基に、Analytic Hierarchy Process(以下、AHP)による一対比較をおこない、品質副特性ごとにルーティングアルゴリズムの評価値を求める。評価値をもとに非機能要求に適したルーティングアルゴリズムの選択

手法を提案する。また、同様に AHP に基づいて開発者やケースごとに非機能要求の重要度を求める。その結果、非機能要求が多様化した場合でも要求に適したルーティングアルゴリズムの選択が可能になると考えた。

2 背景技術

2.1 ESB

ESB はサービス統合をバスの概念で整理したリファレンスアーキテクチャ [1] である。レジストリ、メッセージング機能、ルーティング機能、言語や形式の変換機能などの機能があり、個々の機能は非機能特性と関連している。図 1 に示す ESB のリファレンスアーキテクチャモデルの各層の概要は以下の通りである。

- Change and control
縦に配置されており、他の全ての層を横断している。システムの提案から開発、運用、廃棄までの全ての段階において管理や監視をおこなう
- Orchestration
システムに対する要求からビジネスプロセスを定義し、それをもとにシステム内のどのサービスを利用するかを決める
- Mediation
仲介役として、言語の違いや形式を変換したり、サービスの規約の管理をおこなう
- Connection
正しい宛先に決められたルートでメッセージを送ることで通信する
- Architecture
システムが故障しないように管理、サービス連携する際の管理、接続形態の管理、拡張性の管理をおこなう

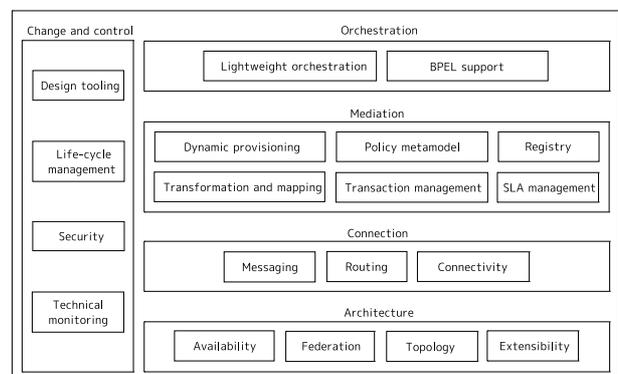


図 1 ESB のリファレンスアーキテクチャモデル

2.2 ルーティングアルゴリズム

ルーティングアルゴリズムとは、最適な経路を決定するための処理の手順である。ESB で用いられるルーティングアルゴリズムはいくつかのバリエーションがあり、そのバリエーションごとに実現できる非機能要求の度合いが異なる。ESB で用いられる Routing は Static Routing と Dynamic Routing の 2 つに分けられる。ESB で主に用いられる Dynamic Routing のルーティングアルゴリズムは以下の 4 種類である。

- CBR(Content-Based Routing)
- PBDR(Pattern-Based Dynamic Routing)
- MDHR(Multifactor-Driven Hierarchical Routing)
- ICMR(Intelligent Conceptual Message Routing)

2.3 Quality Model

ISO9126[3] の Quality Model はソフトウェアの品質の特性を表すために定められた国際規格である。Quality Model を適用して非機能特性との対応関係を整理すると、ソフトウェアの性質やソフトウェアへの要求を明確に特定することができる。ISO9126 はソフトウェアの品質を表す 6 つの品質特性に分類され、品質特性はさらに 27 の品質副特性に分類される。

3 ルーティングアルゴリズム選択手法の提案

ESB を用いた SOA アプリケーションの開発支援のために、非機能特性を考慮したルーティングアルゴリズムの選択手法を提案する。非機能特性を考慮することは、ソフトウェアの品質に影響する。非機能要求とルーティングアルゴリズムの非機能特性の対応関係を整理することで、多様化する非機能要求に対応してルーティングアルゴリズムの組換えが柔軟におこなうことが可能になると考えた。

本研究では、ESB を構成するコンポーネントの 1 つである Routing に着目する。Routing は、SOA に基づくシステムにおける非機能要求の実現に大きく影響する。本研究ではルーティングアルゴリズムのバリエーションごとに ISO9126 の Quality Model を適用し、それぞれのアルゴリズムについて非機能特性との対応関係の整理をおこなう。整理した結果をもとに、非機能特性を考慮したルーティングアルゴリズムの選択手法を提案する。提案した手法を用いることで、非機能要求に最適なルーティングアルゴリズムの選択が可能になると考えた。

3.1 ルーティングアルゴリズムが持つ品質副特性の整理

ルーティングアルゴリズムの性質を品質副特性として整理をおこなう。AHP を利用することによって品質副特性ごとにルーティングアルゴリズムの一対比較をおこない、評価値を求める。評価値を基に、非機能要求に最も適したルーティングアルゴリズムの選択手法を提案する。AHP による評価方法は、目標、評価基準、代替案の階層に整理し、

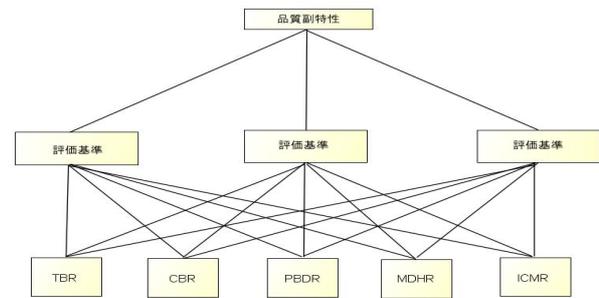


図 2 AHP のモデル図

各階層の要素同士を相対的に評価をおこなう。各階層の要素は、品質副特性、評価基準、ルーティングアルゴリズムとする。AHP のモデルを図 2 に示す。表 1 に各品質副特性における一対比較の評価基準を示し、表 2 に評価値を示す。ただし、本研究においては suitability, interoperability, security, compliance, maturity, usability, changeability, testability, portability を考慮しないものとする。

表 1 評価基準の設定

品質副特性	評価基準		
accuracy	リソースの特定	より良い品質のサービスを提供	品質をリクエスト
fault tolerance	リソースとデータの多重化	動的にルーティングパスを決定	
recoverability	ルーティングパスの復元	メタデータの生成が容易	
time behavior	データストアの数が少ない	リソースが固定	データの解析プロセスが単純
resource behavior	リポジトリを活用している	データの再利用	
analyzability	コンポーネントが簡潔	コンポーネントが疎結合	
stability	責務が明白に分かれている	コンポーネントが疎結合	

表 2 ルーティングアルゴリズムが持つ品質副特性の評価

品質特性	品質副特性	Static	CBR	PBDR	MDHR	ICMR
functionality	accuracy	0.058	0.174	0.072	0.197	0.208
	fault tolerance	0.015	0.084	0.327	0.335	0.143
reliability	recoverability	0.036	0.216	0.234	0.214	0.21
	time behavior	0.637	0.031	0.024	0.021	0.092
efficiency	resource behavior	0.03	0.028	0.215	0.215	0.134
	analyzability	0.437	0.066	0.107	0.04	0.168
maintainability	stability	0.313	0.063	0.063	0.056	0.075

3.2 ルーティングアルゴリズム選択手法の定義

表 2 をもとに、非機能要求に適したルーティングアルゴリズムの選択手法を定義する。本研究で提案するルーティングアルゴリズムの選択手法では、非機能要求に対応した品質副特性の評価の点数が高いルーティングアルゴリズムが非機能要求に最も適している。また、非機能要求の重要度を設定する手順も選択方法に含めることで、非機能要求が複数ある場合にも柔軟に対応することができる。本研究で提案するルーティングアルゴリズムの選択手法の手順を以下に定義する。

1. システムへの非機能要求が Quality Model のどの品質副特性にカテゴリ化されるかを考察
2. 手順 1 でカテゴリ化された品質副特性と表 2 を照らし合わせ、評価値が高いルーティングアルゴリズムを導出
3. 非機能要求が 2 つ以上で尚且つそれら対立関係の場合、AHP にもとづいてそれぞれの非機能要求の重要度を導出
4. 非機能要求の重要度とルーティングアルゴリズムの評価値を乗算
5. 手順 4 で得られた値が最も高いルーティングアルゴリズムを導出

上記の手順を踏むことで、システムへの非機能要求の実現に最適なルーティングアルゴリズムを選択できると考えた。手順 3 以降では非機能要求が多様化した場合を考慮している。複数の非機能要求の重要度を考慮することで、多様化する要求にも対応可能になると考える。

4 事例検証

本研究で提案した手法を実際の事例にあてはめて検証をおこなう。非機能要求を設定し、その要求に最適なルーティングアルゴリズムを示す。事例には、出発駅と到着駅と希望出発時刻を入力すると入力された希望出発時刻以降で最も近い列車の出発時刻を検索し、表示するシステム(以下、時刻検索システム)を用いる。

4.1 要求の設定と要求に対立する品質副特性の整理

時刻検索システムに与える要求として、以下の非機能要求を設定する。

- 実行効率: サービスヘリクエストしてからレスポンスが返ってくるまでの応答時間
- 耐故障性: サービスに障害をきたさない、もしくは障害が起こった場合でもサービスを提供し続ける能力

これらの要求に Quality Model を適用すると、実行効率は time behavior, 耐故障性は maturity と fault tolerance にカテゴリ化されると考えた。time behavior は処理時間を表し、maturity と fault tolerance は信頼性を表す品質副特性なので、このようにカテゴリ化されると考えた。

4.2 提案した手法を用いたルーティングアルゴリズムの選択

本研究で提案した手法にしたがって、非機能要求に対応したルーティングアルゴリズムの選択をおこなう。非機能要求は実行効率、耐故障性、実行効率と耐故障性の 3 通りに設定し、それぞれの要求に適したルーティングアルゴリズムの選択をおこなう。time behavior と maturity, fault tolerance は対立関係となるので、これらの要求に適したルーティングアルゴリズムを選択することで多様化する要求の実現ができたかと考える。その結果、ルーティングアルゴリズムと非機能特性の対応関係を明確に整理することが

可能となり、本研究の目的である SOA アプリケーションの開発支援ができたかと考える。

単一の非機能要求に適したルーティングアルゴリズムの選択手順を示す。手順 1 にしたがって得た結果は、4.1 節で示したとおりである。続いて手順 2 にしたがうと、実行効率に適切なルーティングアルゴリズムは Static Routing となる。耐故障性においては MDHR となる。

複数の非機能要求に適したルーティングアルゴリズムの選択手順を示す。実行効率と耐故障性は対立関係なので、手順 3 にしたがって非機能要求の重要度を求める。図 2 で示した AHP のモデルと同様に目標、評価基準、代替案を定める。非機能要求の重要度を求める AHP のモデル図を図 3 に示す。

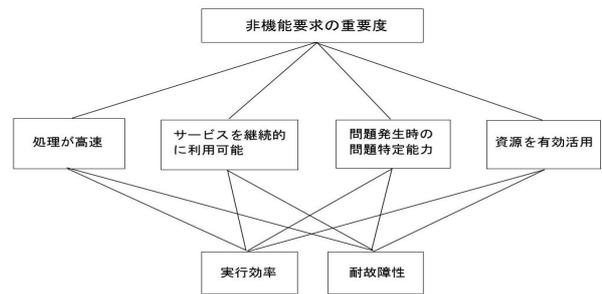


図 3 非機能要求の重要度評価

表 3 にもとづいて実行効率と耐故障性の 2 つの非機能要求の重要度を求める。求めた重要度を基に手順 4 にしたがって、非機能要求の重要度とルーティングアルゴリズムの評価値を乗算する。表 3 に非機能要求の重要度の計算結果を示す。また、非機能要求の重要度とルーティングアルゴリズムの評価値との乗算結果と総合評価を表 4 に示す。

表 3 非機能要求の重要度の計算結果

非機能要求	重要度
実行効率	0.213
耐故障性	0.787

表 4 ルーティングアルゴリズムの評価結果

ルーティングアルゴリズム	time behavior	fault tolerance	総合評価値
Static	0.136	0.012	0.148
CBR	0.007	0.066	0.073
PBDR	0.005	0.257	0.262
MDHR	0.04	0.264	0.268
ICMR	0.02	0.112	0.132

表 4 で導き出した結果より、非機能要求が実行効率と耐故障性の場合に最も適したルーティングアルゴリズムは MDHR とされた。今回の検証では、非機能要求の重要度を主観的に評価して導き出されている。本研究で提案した手法を他の事例で用いる場合には、定義した手順を開発者

が同様に進めていくことで、最適なルーティングアルゴリズムの選択が可能となる。

4.3 ベンチマークテスト

それぞれのルーティングアルゴリズムの実行効率を計測するためにベンチマークテストをおこなった。ルーティングアルゴリズムのフレームワークを基に実装し、実行効率を測定した。ベンチマークテストの条件は以下のとおりである。

- 1つのコンテナに格納されている情報量は同じ
- メッセージを入力してから、宛先が決定するまでの時間を測定
- 宛先が決定してから、経路決定をおこなう処理時間は同じ
- 5回実行し、それぞれの平均時間を比較

ルーティングアルゴリズムのフレームワークを基に実装し、前述した条件にしたがって実行効率を測定した。表5に結果を示す。

表5 ベンチマークテストの結果

Routing Algorithm	CBR	PBDR	MDHR
計測結果	3.317msec	4.219msec	5.091msec

この結果から、今回計測できた Dynamic Routing の中では、CBR がもっとも実行効率が良いということがわかった。

5 考察

5.1 提案したルーティングアルゴリズムの選択手法の妥当性

本研究で提案したルーティングアルゴリズムの選択手法の妥当性について考察する。非機能要求の実現という観点からルーティングアルゴリズムの選択手法を提案したことで、非機能特性を考慮した手法であるといえる。4節で行った事例検証の結果、非機能要求を与えると1つのルーティングアルゴリズムが提示できたので、非機能要求に適したルーティングアルゴリズムの選択が可能であることが確認できた。したがって、提案したルーティングアルゴリズムの選択手法が妥当であると判断した。

5.2 選択されたルーティングアルゴリズムの妥当性

事例検証で選択されたルーティングアルゴリズムについて考察する。実行効率という要求から Static Routing が選択された。Static Routing は静的なルーティングで、Dynamic Routing より単純な構造となり、柔軟性に欠けるが処理が高速という特徴がある。この特徴は実行効率を考慮することに適している。これより、実行効率という要求から Static Routing が選択されたことは妥当であると言える。耐故障性という要求から MDHR が選択された。

MDHR は、Dynamic Routing に属するアルゴリズムで、変更柔軟に対応できるという特徴がある。この特徴は耐故障性を考慮することに適している。これより、耐故障性という要求から MDHR が選択されたことは妥当であると言える。実行効率と耐故障性という要求から MDHR が選択された。MDHR は、変更柔軟に対応できるという特徴と、Dynamic Routing の中で最もメタデータを有効活用しているという特徴がある。この特徴は、実行効率と耐故障性を考慮することに適している。これより、実行効率と耐故障性という要求から MDHR が選択されたことは妥当であると言える。したがって、本研究で提案したルーティングアルゴリズムの選択手法は妥当であると言える。

6 おわりに

本研究は ESB を用いた SOA アプリケーション開発を非機能特性を考慮して支援することを目的として、非機能要求を実現できるルーティングアルゴリズムの選択手法を提案した。提案した手法を用いて、事例検証をおこなった。事例検証の結果、本研究で提案したルーティングアルゴリズムの選択手法の妥当性を確認できた。したがって、本研究で提案したルーティングアルゴリズムの選択手法は、ESB を用いた SOA アプリケーション開発を非機能特性を考慮して支援できたといえる。

今後の課題として、さらに非機能要求が増えた場合の非機能特性のトレードオフを整理する必要がある。非機能要求が増えるたびに非機能特性間の関係が複雑になるが、対応関係を整理することで多様化する非機能特性にさらに柔軟に対応することが可能であると考えられる。また、ESB の Routing 以外のコンポーネントについても非機能特性との対応関係を整理する必要がある。本研究では ESB の Routing に着目して非機能特性との関係を整理したので、ルーティング以外のコンポーネントが持つ非機能特性を整理する必要がある。その結果と本研究で整理した結果を照らし合わせると、非機能要求に応じて ESB の各コンポーネントの組み換えが可能になると考える。

参考文献

- [1] K. Vollmer, M. Gilpin, and S. Rose, "The Forrester Wave : Enterprise Service Bus Q2 2011," Forrester Research, 2011.
- [2] A. Massoud, and P. Nabhani, "Intelligent Conceptual Message Routing in Enterprise Service Bus(ESB)," PROCEEDINGS OF THE 2011 INTERNATIONAL CONFERENCE ON SOFTWARE ENGINEERING RESEARCH PRACTICE, vol. 1, pp. 57-62, 2011.
- [3] ISO/IEC 9126, *Software engineering Product quality -Part 1: Quality model*, 2001.