

科学研究費助成事業（科学研究費補助金）研究成果報告書

平成25年 3月31日現在

機関番号：32612

研究種目：基盤研究(C)

研究期間：2010～2012

課題番号：22500032

研究課題名（和文） サービス指向システムの動的再構成可能な環境に関する研究

研究課題名（英文） An Environment for Dynamically Reconfiguring
Service-oriented Systems

研究代表者

高田 眞吾 (TAKADA SHINGO)

慶應義塾大学・理工学部・准教授

研究者番号：60273843

研究成果の概要（和文）：近年、様々な Web サービスなどのサービスが公開されると共に、そのようなサービスに基づいたソフトウェア（サービス指向システム）が出現している。しかし、サービスは基本的にサービス指向システムとは異なる組織が管理しているため、利用しているサービスが利用不能になったり、変わったりすることがある。本研究では、そのような状況に対応できるように、サービスの状態変化の検出、状態変化時の代替サービスの検索、代替サービスへの切り替えを実現する手法をそれぞれ提案した。

研究成果の概要（英文）：Service-oriented systems, which are software that are based on services (e.g., Web services) are rapidly becoming popular, with the increase in available Web services. But such services are often managed by entities other than the ones managing the service-oriented systems. This means that the services themselves may become unusable and/or change. This research focuses on this issue and proposes approaches that will detect changes in service states, search for services that can replace such changed services, and switch to found services.

交付決定額

(金額単位：円)

	直接経費	間接経費	合計
2010年度	1,300,000	390,000	1,690,000
2011年度	1,200,000	360,000	1,560,000
2012年度	700,000	210,000	910,000
年度			
年度			
総計	3,200,000	960,000	4,160,000

研究分野：ソフトウェア

科研費の分科・細目：情報学・ソフトウェア

キーワード：BPEL サービス指向システム 検索

1. 研究開始当初の背景

近年、Web サービスをはじめとするサービスに基づいたソフトウェア（本研究では「サービス指向システム」と呼ぶ）が注目されている。サービス指向システムは、サービスを連係させることにより動作する。サービス指向システムは、広義では、ソフトウェア部品の再利用に基づいたシステムであり、その見

地から今までに様々な研究が行われてきている。研究代表者の高田も、平成21年までに、再利用したいソフトウェア部品の探索に関する研究や、ソフトウェア部品の統合に関する研究を様々な角度から行ってきた。

サービスは多くの場合第三者が提供しており、利用する側（サービス指向システム本体）とは独立に存在するため、使えない状態に

なったり、バージョンアップしたりすることがある。つまり、利用しているサービスが、サービス指向システム開発時の状態から変わることがある。また、ソフトウェアは要求の変更のため、保守が従来から大きな位置を占めており、サービス指向システムもその限りではない。つまり、サービス指向システムが必要とするサービスが変わることがある。いずれの場合も、サービス指向システムを運用した後でも、状況に応じて構成を変えたり、進化したりする能力が必要であることを意味する。

2. 研究の目的

本研究では、サービス指向システムが利用しているサービスに、その利用に関わる状態（利用不能やバージョンアップなど）に変化があったとき、サービス指向システムとして利用し続けられるような環境を提案し、構築することを目的とする。環境を実現するために、状態変化の検出、従来サービスに代わる代替サービスの検索、代替サービスへの切り替えの三つに着目する。

3. 研究の方法

本研究は、利用しているサービスの状態変化の検出、従来サービスに代わる代替サービスの検索、代替サービスへの切り替えの三つを対象にしているため、それぞれについて研究方法について述べる。

(1) サービスの状態変化の検出

サービスの状態変化を検出するためには、まず、どのようなサービスがあるか列挙し、分類する必要がある。そこで、様々なサービス指向に関わる文献などを調査する。そして、それらの特徴などに基づいて、分類する。

次に、状態変化の分類に基づき、検出方法を提案し、実装する。検出方法は、状態変化の種類によって異なることが考えられるため、その点に注意する必要がある。さらに、各状態変化を検出するためにどのような情報が必要なかを検討する。特にサービス提供者自身が提示しなければいけない情報と、自動的に取得できる情報は区別する。また、必ず再構成しなければならない状態変化なのか、再構成しなくてもよい状態変化なのかも区別する。以上を踏まえて、状態変化の検出機構を、可能な限り一つのフレームワークとしてまとめる。

(2) 従来サービスに代わる代替サービスの検索と選択

状態変化に基づき、使用中のサービスに代わる代替サービスを検索し、選択する手法を提案する。検索手法については、様々な手法が考えられるため、一つの包括的な手法を提

案するよりは、様々な角度から検討する。その際、代替サービスを決定するために、どのような情報が必要となるかに注意する。そして、必ずしも事前に代替サービス候補を用意していなくても、必要になった時点で候補を探し、その中から代替サービスを決定する手法を目指す。

(3) 代替サービスへの切り替え

代替サービスへの切替機構は、基本的に様々な方法を実装し、試行することにより、よりよい手法を提案する。つまり、切替機構の提案と実装は並行して行う。

4. 研究成果

研究成果を、サービスの状態変化、サービスの検索と選択、サービスの切り替えに分けて述べる。

(1) サービスの状態変化

サービスの動的再構成を行う際、いつどのように行くと、どのような影響を与えるかを明らかにする必要がある。そこで、状態変化の分類と検出の二つに分けて述べる。

①状態変化の分類

状態変化は、「静的・動的」と「守る必要あり・なし」の2種類の軸により分類した。

まず、状態を大きく「静的」と「動的」の2種類に分類した。静的な状態は、あまり変化しない状態であり、特にプロバイダが何かしなくとも変わらない状態である。代表例は利用料金である。動的な状態は、刻一刻と変化する状態であり、プロバイダの意図とは必ずしも関係なく利用中変化してしまう状態である。代表例は応答時間、スループット、利用頻度などがある。

次に、状態の種類を守らなければならないか否かで分類できる。守らなければならない状態は、サービス提供者と利用者間で結んだ契約が対象としている状態と、サービス利用者側が独自に設定した状態の2種類である。前者の例として、SLA (Service Level Agreement)の中で規定された状態である。後者は、サービスが有用であるために、サービス利用者が設定したレベルの希望値である。例えば、応答時間は1秒以内でないと、そもそも使えない、とサービス利用者が考えた場合である。

②状態変化の検出

サービスの状態の検出方法は、その種類によって異なる。静的状態は、プロバイダが何かしたときに変化するため、基本的にプロバイダが知らせる必要があると考え、本研究では動的状態のみを対象にした。

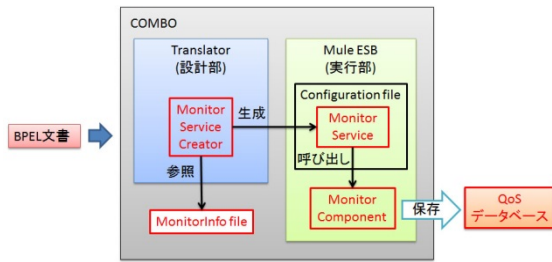


図1 状態変化検出機構

動的状態の変化を検出するために、インフラストラクチャレベルで Quality of Service (QoS) をモニタリングする機構を提案した。インフラストラクチャの実装はオープンソースの Mule Enterprise Service Bus (ESB) および COMBO システムを利用した。COMBO システムは、Jongtaveesataporn および高田が作成した、BPEL 文書を Mule ESB Configuration File に変換するシステムである。そのため、実装 (図 1) は、まず、各サービス指向システムを定義している BPEL 文書を分析し、どの QoS をモニタリングするか判断し、その結果に基づき Monitor Component が実際に様々なデータを取得する。取得したデータは QoS データベースに蓄積し、その傾向を見て、サービスの状態が変化したか否かの判断をする。

上記の分類はすべてを網羅している保障はないため、今後新たな状態変化を見つけた時、それが本分類との関係を精査し、分類を更新する必要がある。さらに、モニタリングは、動的状態のみを対象としたため、今後、静的状態については、プロバイダがどのように変化を知らせるべきかなどを検討する必要がある。

(2) サービスの検索と選択

サービスの検索と選択については、三種類の手法を提案した。

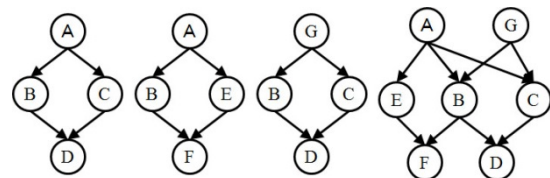
① サービス群に基づいた検索

サービス指向システムは、原理的には各サービスが独立しているため、一つのサービスが利用不能になった場合、同等の機能をもっているサービスに変更できるはずである。しかし、実際は、サービス間に何らかしらの関係があるため、利用不能になったサービスを代えるだけではなく、それと強く関係のあるサービスも変更する必要がある。さらに、従来のサービスの検索は単独のサービス検索に注目していた。しかし、複数のサービス指向システムの間、部分的に同等のサービスの組み合わせが含まれていることがある。

上記の状況を基に、サービスを単独に検索

するのではなく、複数のサービスの組み合わせを検索する手法を提案した。この組み合わせは、BPEL フラグメントと呼び、BPEL (Web Service Business Process Execution Language) に基づいて記述されているものとする。

サービスを単独に検索するのではなく、複数のサービスの組み合わせを検索する意義として次があげられる。まず、組み合わせは既存の BPEL 文書の一部であるため、コンテキスト情報 (どのように使われているか) がある。また、既存の BPEL 文書の一部であるということは、利用可能な組み合わせであることを意味し、その組み合わせは“妥当”であることが考えられる。さらに、使われているということは、低品質である可能性が低いということが考えられる。



(a) BPEL1 (b) BPEL2 (c) BPEL3 (d) BPEL graph

図2 BPEL グラフの統合

検索機構は、BeFraS (BPEL Fragment Search) というツールとして実装した。本ツールはデータベース部と検索部からなり、データベース部は、グラフデータベースである Neo4J を利用して実装した。新しい BPEL 文書は、共通しているサービスをまず統合してからデータベースに登録する。例えば、図 2 では、サービス B が BPEL1, BPEL2, BPEL3 すべてで使われているので、データベースに格納すべき最終的なグラフでは、一つのノードとして表す。検索部では、ユーザがプロセスクエリを作成することにより、データベース部内を検索し、既存の BPEL 文書の一部 (BPEL Fragment) を部分グラフとして表示する。

本研究は、雑誌論文①で発表した。

② クラスタリング

二つ目の手法は、クラスタリングに基づく。クラスタリングによって得られた情報を利用することで、QoS の値が変化した場合に連携する Web サービスの再選択を行う手法を提案する。なお、クラスタリングの情報とは、候補となる似た機能を持つ Web サービスごとの ϵ 近傍と属しているクラスタ、及び選択された Web サービスが属するクラスタのことを指す。

再選択を行うためにクラスタリングの情報を利用する理由は、QoS の値が変化する前に選択された Web サービスを含むクラスタのみに着目することで再選択を行えるため

ある。また、クライアントの要求に最も沿っている Web サービスがうまく関係出来なかった際、同クラスタから素早く代わりとなる Web サービスを選択することが出来るという利点もある。

クラスタリングに基づいた手法は、QoS の値が変化した場合の再選択だけでなく、QoS の値の変化がクラスタに与える影響についても考慮した。QoS の値の変化により Web サービスが以前属していたクラスタとは別のクラスタに属するべき可能性や、クラスタが分裂する可能性があるためである。そこで、Web サービスの再選択と同時に再クラスタリングを行う。再クラスタリングすることにより、再度 QoS の変化が起きた場合でも正しく Web サービスの再選択が行われるようになる。

本研究は、学会発表①で発表した。

③ K-d 木とキャッシュを利用した高速検索

より高速な検索を可能にするため、K-d 木とキャッシュを応用した手法を提案した。提案手法は、まず Service Type という概念を導入し、サービスを機能ごとに分類する。次に機能ごとに K-d 木を作成する。K-d 木の各ノードはサービスであり、次元 K は QoS(Quality of Service)の種類の数である。ノードを K-d 木上に配置するとき、ノードの QoS 値に着目して木に挿入する。

次に、サービスを検索する際、まず、過去の検索結果を保存しているキャッシュの中を探す。過去の各検索結果は、複数のサービスから成り、そのサービスを一つの QoS 特徴ベクトルでまとめてある。この特徴ベクトルは、複数の QoS 値からなり、各 QoS 値はサービスの中の最も悪い値をとる。つまり、検索結果中の各サービスの QoS 値は、特徴ベクトル中の各 QoS 値以上の値を取っている。キャッシュ中を探す際、クエリの QoS 値を満たす特徴ベクトルをもつ過去の検索結果を探す。

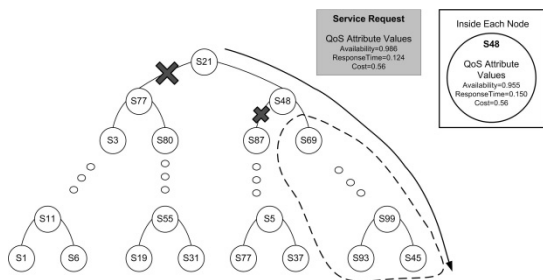


図 3 K-d 木上の K-NN 探索

キャッシュに利用できる過去の検索結果がない場合、K-d 木上からサービス候補を探す。探索は、K-Nearest Neighbor(K-NN)手法を利用している(図3)。

以上の提案手法はシミュレーションを通して、その効果を確認した。本研究の成果は現時点では未発表であるが、今後発表予定である。

上記の三つの手法はそれぞれに特徴があるものの、現状では独立している。そこで、今後は、これら三つの手法を一つのフレームワークに統合する必要がある。

(3) サービスの切り替え

代替サービス群から、実際に利用するサービスを決定した後、切り替えを動的に行えるような基礎的な枠組みを、Mule ESB 上で実装した。

5. 主な発表論文等

(研究代表者、研究分担者及び連携研究者には下線)

[雑誌論文] (計 1 件)

① Shingo Takada: “Finding Web Services via BPEL Fragment Search”, Proc. of SUITE 2011, 査読有, pp.9-12 (2011年).

[学会発表] (計 1 件)

① 中島健, 高田真吾: “QoS に基づいたクラスタリングによる Web サービスの選択”, 情報処理学会第 74 回全国大会, 名古屋工業大学 (2012年3月7日).

6. 研究組織

(1) 研究代表者

高田 真吾 (TAKADA SHINGO)
慶應義塾大学・理工学部・准教授
研究者番号: 60273843

(2) 研究分担者: なし

(3) 連携研究者: なし