

サービス連携の「見える化」のための ESB 機能コンポーネントに関する一考察

上野 雄司[†] 武内 惇^{††} 金子 正人^{††} 泉 隆^{†††} 関根 好文^{†††} 菌田 孝造^{††††}

[†]日本大学大学院工学研究科 ^{††}日本大学工学部 ^{†††}日本大学理工学部 ^{††††}マイクロテクノ株式会社

1. はじめに

システムをサービスとして考え、既存のシステムをうまくサービス単位で分割し、サービス同士を ESB(Enterprise Service Bus)を用いて連携(以下 サービス連携)させてシステムを構成することにより、早期にシステムを実現する手法である SOA(Service Oriented Architecture)に基づくシステム開発が進められている。

しかし、問題として、サービス連携では数多くのサービスが1つのサービスと連携するとき、その1つのサービスを変更するに当たって、数多くのメディアエーション(サービス同士の連携を行う機能)の修正を行わなければならない。修正するメディアエーションが多くなるため、ESB のシステム管理者は、ESB で発生した問題を1つずつ検出し修正することが困難になり、サービスの変更要求に迅速に対応することができないという問題が発生する。

この問題を解決するために、どのメディアエーションにどんな異常が発生し、どのメディアエータファンクション(メディアエーションを構成するプログラム、以下 メディアエータ)をどのように修正すればよいかを迅速に管理者に連絡する「見える化」を実現することにより、より迅速なメディアエータの修正を可能とする。本研究では、ESB に「見える化」する仕組みを組み込むことにより、SOA に基づき構成するシステムの「見える化」を実現する方法を提案する。

本稿では、ESB として Interstage^[1]を用いることを前提に、「見える化」を組み込んだ OSCAR^[2]モデルに基づき ESB コンポーネントの実装方法を提案する。また、クラウド環境上の Google Apps を使用した安否システムのサービス連携を事例に、実装方法の実現可能性について述べる。

2. 「見える化」の対象となる異常

ESB で発生するどのような異常に対し、「見える化」するのか特定する必要がある。そのため、ESB で発生すると考えられる異常を列举し、OSCAR モデルにおいて、サービス連携を実現する ESB コンポーネント^{[3][4]}は Order, Coordinate, Action であるため、その3つの ESB コンポーネントにて発生する異常は表1のものが挙げられる。

表1. ESB コンポーネントで発生する異常

ESB コンポーネント	起こりえる異常
Order	<ul style="list-style-type: none"> 許可しないサービスリクエストの要求 送信する XML の構成誤り 許可しない XML のデータ
Coordinate	<ul style="list-style-type: none"> Web サービスの返り値の誤り Coordinate での URL の設定誤り 長時間の実行時間 実行エラーによる例外処理の発生
Action	<ul style="list-style-type: none"> XML の構造誤り 許可しない XML のデータ 長時間の実行時間 実行エラーによる例外処理の発生

3. OSCAR モデルに基づく ESB コンポーネント

表1の異常を「見える化」する OSCAR モデルに基づく ESB コンポーネントを考える。サービスリクエストからの要求により、ESB 上に登録したメディアエーションのシーケンスのコンポーネントを順番に実行し、Web サービスとの連携を実現する。OSCAR モデルに基づくコンポーネントを用いたサービス連携の仕組みを図1、ESB コンポーネントの概要を表2に示す。

表2. ESB コンポーネントの概要

ESB コンポーネント	概要
Order	サービスリクエストからの要求より ESB に XML 文書を送信しシーケンスを実行する。
Stream (start)	Order から取得した XML 文書が正しい構成であるか、IP アドレスが許可した IP アドレスであるかをチェックする。
Stream	コンポーネント間で処理するデータが正しいものであるかを XMLDB の判断材料を用いて判定を行い、異常があると判定した場合は、XMLDB のエラーログにその異常についての詳細を記述する。
Stream (end)	Stream にて記録した各 ESB コンポーネントの実行時刻から各 ESB コンポーネントの実行時間を計測し、その評価を行う。
Stream (exception)	シーケンスで実行する ESB コンポーネントに例外処理が発生した場合にその詳細をエラーログに蓄積する。
Action	サービスリクエストやサービスプロバイダが扱えるようにデータの編集を行う。
Coordinate	サービスプロバイダにメッセージを送信する。サービスプロバイダの環境に依存しないように、SOAP メッセージを送信する。
Report	一定時間ごとに XMLDB のエラーログをチェックし、異常について新しい記述をした場合に管理者にその異常についての詳細を記述したメールを送信する。

A Study about the Function Component Composition of ESB for "Visualization" of the Service Cooperation.

[†]Yuji Ueno, ^{††}Atsushi Takeuchi, ^{††}Masato Kaneko, ^{†††}Takashi Izumi, ^{†††}Yosifumi Sekine ^{††††}Kouzou Sonoda.

[†]Graduate School of Engineering, Nihon University,

^{††}College of Engineering, Nihon University,

^{†††}College of Sciences and Technology, Nihon University,

^{††††}Microtechno Corp.

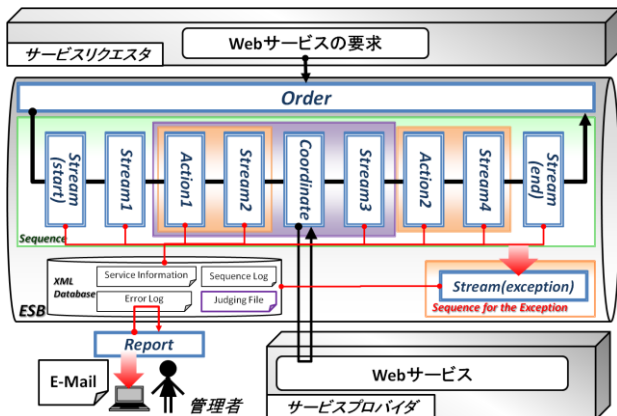


図1. 「見える化」を考慮したサービス連携の仕組み

以上の ESB コンポーネントを用いて、サービス連携を構成する。

XMLDB に異常かどうか判定する際の判断基準となるファイル(Judging File)を作成する。Judging File は、異常を判定するための XML 文書と修正方法を記述するための XML 文書の 2 種類のファイルに分けられ、その例をそれぞれ図2と図3に示す。Judging File を用いることにより、許可する IP アドレスを変更する場合や新たに修正方法を追加する場合など Stream を変更する必要がないため柔軟性の向上を図る。また、異常を検出する Stream をより容易に作成することを可能にする。

```
<?xml version="1.0" encoding="UTF-8" ?>
- <judgingFile_ipAddress>
- <permitted_ipAddress>
  <ipAddress>133.43.8.45</ipAddress>
  <ipAddress>133.43.8.46</ipAddress>
</permitted_ipAddress>
</judgingFile_ipAddress>
```

図2. 許可する IP アドレスを記述した Judging File

```
<?xml version="1.0" encoding="UTF-8" ?>
- <judgingFile_faultCode>
- <coordinate>
  - <fault>
    <id>VersionMismatch</id>
    <detail>処理中のパーティによりSOAPエンベロープ要素に無効なネームスペースが検出されました。</detail>
  - <correct>
    <id>Action_CorrectSOAPMessage</id>
  - </correct>
  - </fault>
  - <fault>
    <id>MustUnderstand</id>
    <detail>処理中のパーティが認識または理解しなかったSOAPヘッダー要素の直下の子要素に、値1を持つSOAP MustUnderstand属性が</detail>
  - <correct>
    <id>TargetService_CorrectWebService</id>
  - </correct>
  - </fault>
</coordinate>
</fault>
```

図3. 修正方法を記述した Judging File

これにより、ESB で発生する異常に対して「見える化」を実現し、シーケンスで異常が発生した場合、迅速に管理者に伝えることができると考えた。

4. 「見える化」の実現可能性の検討

「見える化」を組み込んだ OSCAR コンポーネントにより、サービス連携のときに異常を検出できるか検討する。クラウド環境上の Google Apps を使用した安否システムのサービス連携を事例とする。このサービス連携では、サービスリクエスタから要求し、学生サービスシステムからの要求により、Google Spreadsheets に蓄積されている安否システムの安否情報をファイル出力する。この事例における、OSCAR モデルに基づく ESB コンポーネントを用いたサービス連携の仕組みを図4に示す。サービス連携のときに異常を検出できるかどうか検討する。検出する異常の例として、サービスリクエスタの IP アドレスを変更した場合について、「見える化」がどのように機能するかについて述べる。

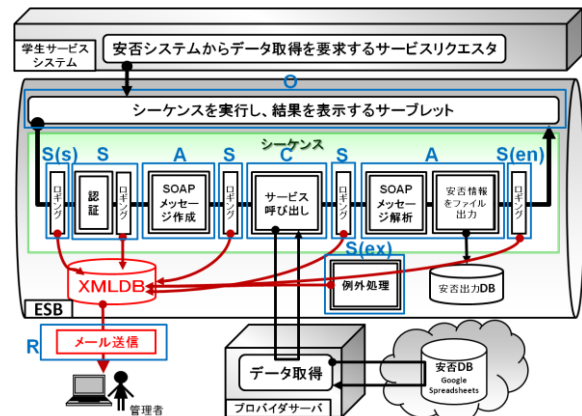


図4. OSCAR モデルを用いたサービス連携の仕組み

ESB コンポーネントの Order により、サービスリクエスタである学生サービスシステムの IP アドレスを取得し、IP アドレスを付加した XML 文書を ESB のシーケンスに送信する。XML 文書を受け取ったシーケンスは、Stream(start)を実行し、XMLDB の判断材料を用いて IP アドレスのチェックを行う。XMLDB の判断材料に登録した IP アドレスのみがサービス連携を実行できるため、Stream(start)にて異常を検出し、その異常の詳細を XMLDB のエラーログに蓄積する。その後、Report により、管理者にメールを送信するため、管理者は異常が発生したと理解できる。この場合は、許可しない IP アドレスからのアクセスがあったという異常が発生し、「IP アドレスを登録する」、または、「IP アドレスのクライアントに警告をする」といった対処法も理解できる。

以上の結果から、「見える化」を組み込んだ OSCAR コンポーネントを用いることにより、管理者は、どの機能コンポーネントにどんな異常が発生したのか、また、その対処法も知ることができる。

5. おわりに

本稿では、「見える化」を組み込んだ OSCAR モデルに基づく ESB コンポーネントの実装方法について述べ、実現可能である見込みを得た。

今後は、管理者がより迅速に問題に対処できるようにするため、より具体的な原因や対処法の内容を提供する仕組みを考える。

謝辞

学生サービスシステムの開発にご助言いただいた、富士通ネットワークソリューションズ(株)、広野真吾氏、藤吉秀明氏、また、Interstage の使用にあたりご協力いただいた、富士通(株)に深謝します。なお、本研究は日本大学学術研究助成を受けて行った。

【参考文献】

- [1] 富士通株式会社「Interstage Service Integrator」：
<http://interstage.fujitsu.com/jp/serviceintegrator/>.
- [2] 武内ほか：“合意形成マルチエージェント指向企業通信システムの構築法の提案”，情報処理学会論文誌第 41 号 第 4 号別冊(2000-4) pp.1212-1224
- [3] 泉(奈)ほか：“OSCAR モデルに基づく ESB の機能構成法に関する一考察—サービス連携をコンポーネントに分割する方式—”，情報処理学会第 72 回全国大会(2010-3)
- [4] 泉(貴)ほか：“OSCAR モデルに基づく ESB の機能構成法に関する一考察—学生サービスシステムへの適用—”，情報処理学会第 72 回全国大会(2010-3)