

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 —

泉 貴志[†], 武内 惇^{††}, 金子 正人^{††}, 泉 隆^{†††}, 藺田 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

企業におけるビジネス業務の変革が急速に進み、その変革に応えるシステムを迅速に構築することが難しい。そのため、システム上のサービスを ESB(Enterprise Service Bus)を用いて連携して、業務変革に柔軟に対応できる SOA(Service Oriented Architecture)に基づくシステム構成法への期待が大きい。しかし、問題として、1つのサービスがいろいろな業務で使用されるため、サービスを変更する場合、ESB 上でサービスの連携部分のどの機能をどのように変更すべきかの判断が難しく、サービスの変更要求に迅速に対応することができない。

変更すべき機能を発見するためには、どの連携部分にどんな異常が発生したのかを見える化(異常の見える化)する必要がある。「異常の見える化」を組み込んだ OSCAR モデルに基づき、ESB の機能コンポーネントを構成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、ESBの機能コンポーネントの実装方法を提案する。また、その実装方法の実現可能性を、事例を用いて検討する。

2. サービス連携処理のモデル

ESB を用いたサービスの連携の処理について説明する。ESB を用いたサービス連携についての仕組みを、図 1 に示す。

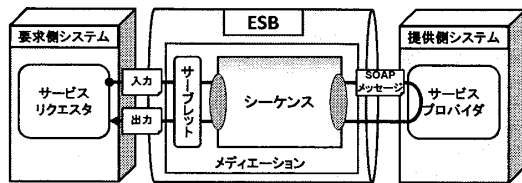


図 1. サービス連携の仕組み

要求側システム上に存在するサービスリクエスタが、ESB 上のサブレットに入力値を送信する。入力値を受け取ったサブレットは、登録したシーケンスを実行する。シーケンスでは、提供側のシステムのサービスプロバイダのインターフェースに合うように、データを編集し、SOAP メッセージにして送信する。サービスプロバイダは、受け取った SOAP メッセージに基づいてサービスを実行する。実行結果を受け取ったシーケンスは、SOAP メッセージを解析し、出力値を取り出しサービスリクエスタに返す。このような処理を実行して、サービスの連携を実現する。

A Study about the Function Composition of ESB based on OSCAR model, Application to Student Service System.

[†] Takashi Izumi, ^{††} Atsushi Takeuchi, ^{††} Masato Kaneko,

^{†††} Takashi Izumi, ^{††††} Kouzou Sonoda.

[†] Graduate School of Engineering, Nihon University,

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

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

^{††††} Microtechno Corp.

3. OSCAR モデルに基づく ESB の機能コンポーネントを用いたサービス連携処理

OSCAR モデルを用いた ESB の機能コンポーネントをどのような順番で実行するのかについて、学生サービスシステムが出席システムのデータベースにデータを追加することを事例として述べる。その事例において、OSCAR モデルでの ESB の機能コンポーネントを表 1 に示す。また、そのコンポーネントを用いたサービス連携の仕組みを図 2 に示す。

表 1. ESB の機能コンポーネント

OSCAR 機能コンポーネント	機能コンポーネントの役割
Order	・データの認証
Stream	・ロギング
Coordinate	・サービスの呼び出し
Action	・XMLDB からのデータの取得 ・SOAP メッセージの作成 ・SOAP メッセージの解析
Report	・管理者への報告

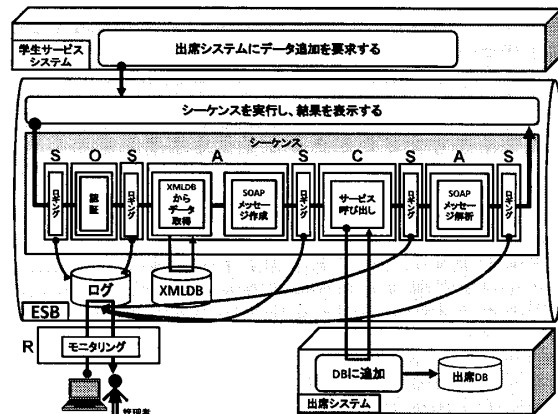


図 2. OSCAR モデルを用いたサービス連携の仕組み

学生サービスシステムからの要求により、ESB 上に登録したシーケンスを実行し、登録した機能コンポーネントを順番に実行し、出席システムのサービスの連携を実現する。このシーケンスの役割は、まず送信するユーザーとデータの認証を行い、送信するメッセージを編集して対象のサービスに送信し、返ったメッセージから実行結果を解析することである。シーケンスで実行する機能コンポーネントの順番は、Order, Action, Coordinate, Action の順となる。そして、それらの機能コンポーネントの実行後に、Stream を実行することにより、直前の機能コンポーネントの実行結果をログに蓄積して、どの機能コンポーネントでどんな異常が発生したのか明確にする。すなわち、シーケンスで実行する機能コンポーネントの順番は、① Stream, ② Order, ③ Stream, ④

Action, ⑤ Stream, ⑥ Coordinate, ⑦ Stream, ⑧ Action, ⑨ Stream である。Stream で蓄積したログにより, 管理者は, Report を実行し, どの機能コンポーネントにどんな異常があるのかを発見することができる。

4. Stream を用いる見える化の実現法

OSCARモデルの機能コンポーネントにおいて, サービス連携に関する問題の早期発見と問題解決を実現するために, 「見える化」の考え方を採用し, すなわち, 「見える化」の仕組みとして, サービスの実行制御上の問題を発見するための「異常の見える化」の仕組みの実現を図る[2]。 「異常の見える化」は, 誰もが異常を認知できるようにすることが重要であることから, ESBにどんな異常が発生したのか見えるようにする仕組みが重要である[2]。 OSCARモデルでは, Streamに機能コンポーネント間で処理する情報を基にどんな異常であるかをログに蓄積することにより, 「異常の見える化」を実現する。Streamで行う処理の流れを図3に示す。

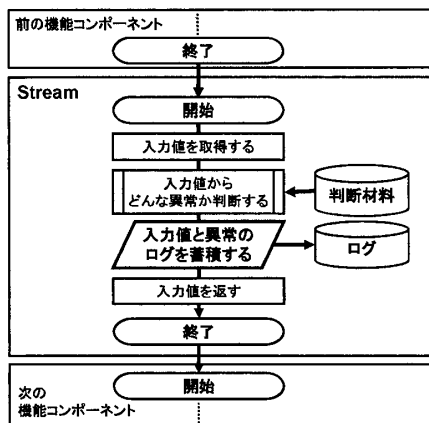


図3. Streamの処理の構成

Streamの処理では, 直前のコンポーネントの実行結果を入力値として取得し, それが異常であるか判断する。次に, 入力値と異常についてのログを蓄積し, 入力値をそのまま返す。図3において重要となるのが, 異常の判断である。入力値だけでは, どんな異常であるのか判断することは難しいため, あらかじめどんな異常が発生するか考える必要がある。図2の事例では, ESB上で発生すると予想した異常は表2の通りである。

表2. コンポーネントで予想される異常

OSCAR コンポーネント	予想される異常
②Order	・不正アクセス ・不正な入力値
④Action	・XMLDBでのエラー ・SOAPメッセージの作成の失敗
⑥Coordinate	・対象のサービスが存在しない ・対象のサービスに適切な入力値
⑧Action	・SOAPメッセージの解析の失敗 ・不正な出力値

これらの異常を判断するような処理を OSCAR コンポーネント別に設定することにより, それぞれに対応した「異常の見える化」を実現する。

5. 学生サービスシステムによる実現可能性の検討

ESBの機能コンポーネントが, サービス連携での異常

の検出が可能であるか検討する。事例として, 表2の⑥Coordinateでの「異常の見える化」について検討する。予想される各異常を検討するために, (1)サービスのアドレスの変更と(2)サービスの入力値の追加という業務変化が発生した場合について(図4), 「異常の見える化」とそれを用いた解決法について述べる。

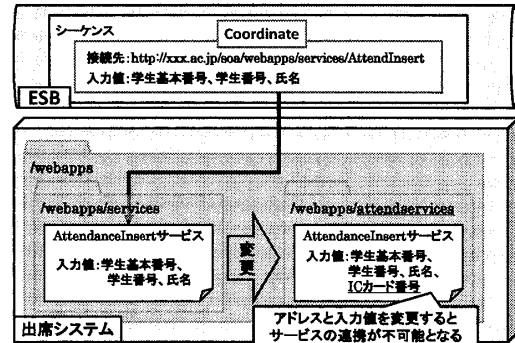


図4. 対象のサービスの変化

OSCARモデルのStreamで蓄積するログにより, (1)「⑥Coordinate」において「対象サービスが存在しない」という異常が見えるようになる。これにより, サービスの選定で設定したアドレスが適切ではないと判断でき, 対象のサービスのURLを取得し, Coordinateで登録するURLをhttp://~/webapps/attendservices/~に修正することで異常を解決することができる。

(2)Streamで蓄積したログには, 「対象のサービスに適切な入力値」という異常も見えるようになる。そして, 対象のサービスの入力値の情報を取得することにより, 送信する入力値にICカード番号が必要であると判断できる。入力値の修正は, Actionで修正する。この場合「XMLDBからのデータの取得」の機能に, 学生基本番号, 学生番号, 氏名以外にICカード番号を取得することを追加すれば, 適切な入力値とすることができる。

以上の結果から, OSCARモデルを用いたコンポーネント分割法を用いることにより, どの機能コンポーネントにどんな異常が発生したのか容易に判断することができる。

6. おわりに

本稿では, OSCARモデルを用いたコンポーネント分割法を, 学生サービスシステムの事例へ適用し, サービス連携の不具合の検出が可能となる見込みを得た。

今後は, 「異常の見える化」だけではなく, 状況の見える化や会社全体としての見える化の仕組みも実現し, 内部統制の仕組みの組込み法も明らかにしたい。

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)今田和雄氏に深謝します。なお, 本研究は日本大学学術研究助成金を受けて行った。

【参考文献】

- [1]富士通株式会社「Interstage Service Integrator」: <http://interstage.fujitsu.com/jp/serviceintegrator/>
- [2]遠藤功: 見える化 強い企業をつくる「見える」仕組み, 東洋経済新報社(2005-10)。