

学生サービスシステムを事例とする SOA に基づくシステム構成に関する研究 —異常の兆候の「見える化」を実現する仕組み—

山崎雄司[†], 大山勝徳^{††}, 金子正人^{††}, 武内 惇^{††}, 泉 隆^{†††}, 関根 好文^{†††}, 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

A Study about the System Construction Based on SOA with Student Service System
—Signs of abnormal "Visualization" Mechanism to achieve—

[†] Yuji Yamazaki, ^{††} Katsunori Ooyama, ^{††} Masato Kaneko, ^{††} Atsushi Takeuchi, ^{†††} Takashi Izumi,
^{††††} Yoshifumi Sekine, ^{††††} Kouzou Sonoda

[†] Graduate school of Engineering, Nihon University, ^{††} College of Engineering, Nihon University
^{†††} College of Science and Technology, Nihon University, ^{††††} Microtechno Corp.

1. はじめに

近年, 企業におけるビジネス業務の変革が求められている. この要求に応えるシステムを早期に実現する SOA(Service Oriented Architecture)に基づくシステム開発法に対する期待が大きい.

しかし, SOA に基づきサービス連携を行うにあたり, 複数のシステム間でどのようなデータのやり取りが行われているか, 正しい方法でユーザが使用しているか, 業務変更が行われたときに, サービス連携に起こる異常を ESB(Enterprise Service Bus)の管理者 (以下管理者) が把握することが難しく, サービス連携に起こる異常に対してどのように対応するべきか分からないことが問題となっている. これらの問題を解決する仕組みとして「見える化」^[1]の仕組みを ESB に装備する.

本研究では学生サービスシステム (以下学生サービス) を事例とし, SOA を用いてシステムを構築した場合に起きる問題を見る化する仕組みを実現する方式の検討を進めている. 本稿では ESB における見える化の仕組み, 「状況の見える化」の重要性, 及び, ESB で発生する問題の原因のカテゴリ分けについて述べた後, ESB に発生する異常を未然に防ぐための異常の兆候について検討を行い, 学生サービスの事例について述べる.

3. 「見える化」の仕組み

3. 1 ESB における「見える化」の仕組み

参考文献^[1]によると見える化とは, 本来見えないもの, 見ようとしなくても見せるようにするものである. 見える化は, 問題, 状況, 知恵, 顧客, 経営の 5 つに分類する. また, 問題, 状況, 知恵, 顧客のカテゴリは対象項目として細分化する. 全ての見える化における出発点は, 問題の見える化における異常の見える化とギャップの見える化である. シグナルの見える化は異常の見える化から得た異常を見る化を行うことである. 基準の見える化とステータスの見える化を行うことによりギャップの見える化が可能となる. 異常の見える化とギャップの見える化から問題が起きた原因を真因の見える化で見えるようにする. 真因の見える化と経験の見える化, ヒントの見える化から問題解決を図る. 問題解決の効果は効果の見える化により見えるようにする.

企業の業務は上司が部下へ作業を依頼し, 作業を終えた後, 部下は上司に結果を報告する流れで進められる.

筆者らは, サービス使用者が ESB に作業を依頼し作業が終了後, サービス使用者に結果を報告する流れで機能すると考え, 企業と ESB で行う仕事の流れは同じであると考える. ESB における見える化の体系を図 1 に示す.

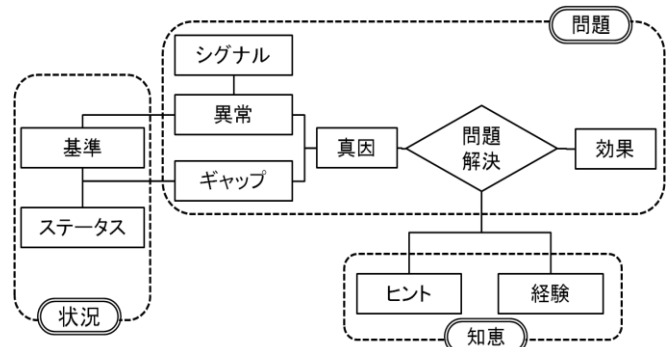


図 1. ESB の見える化の体系

3. 2 ESB における「状況の見える化」の重要性

ESB における見える化では, 「いかに早く問題を見つけるか」以上に「いかに問題発生を抑えるか」が重要になる. 先に述べた問題の見える化により, 問題を発見することはできるが, 問題の見える化はすでに起こっている問題をいち早く見つけようとするものであり, 問題発生を抑えるためには機能しない. また, 問題をいち早く見つけることができたとしても, その問題を必ず対処できるとも限らないため ESB を安全に運用しているとは言えない. この問題を解決するために状況の見える化のステータスの見える化を使用し, 今現在の状況から, 将来発生しようとしている問題の原因を特定し, 問題の発生を未然に防ぐ.

3. 3 ESB で発生する問題の原因のカテゴリ分け

ESB で発生する問題の原因はケースごと異なり多岐にわたるため, 非常に乱雑である. このままでは仮に異常の兆候を特定できた場合においても, その異常に対してのみ使用することでき, 他のケースについては使用することができないため汎用性に欠ける. また, 乱雑なままでは異常の兆候を特定することが困難である.

この問題を解決するために, ESB で発生する問題の原因に対してカテゴリ分けを行い, 類似する問題の原因を 1 つのカテゴリとして扱う. これにより汎用性を高め, また, 異常の兆候を特定する際に使用するデータマイニングの効率の向上をも狙う. 表 1 に問題の原因をカテゴリ分けしたものを示す.

4. 異常の兆候

異常が起こる前には必ず異常の兆候があり, 異常を予測するためには常にステータスが見えていなければならない. ステータスを見ることができなければ, 異常の兆候を未然に発見できないため, 異常が発生した後に対処しなければならず, 業務に支障をきたす恐れがある. しかし, ステータスを見ただけではその見えている情報が

表 1. ESB で発生する問題の原因のカテゴリ

カテゴリ名	原因
セキュリティの不備	セキュリティホール (バグ、不具合、あるいはシステム上の盲点) を起因とする原因
ハードエラー	ハードウェアの故障などに起因する原因
ヒューマンエラー	人間による意図的ではない操作ミス, 設定ミスなどに起因する原因
不正行為	人間による意図的な不正操作、不正な設定などに起因する原因
想定外問題	急なトラフィックの増加やソフト・ハードウェアの不具合、環境などに起因する原因

正常な状態なのか異常の兆候なのかを判別することはできない。判別できるようにするためにはどのような異常の兆候が発生しているのかを知る必要がある。そのため、事例を用いて異常の兆候を知るために必要な情報について検討をする。

事例として学生サービスを事例に、ESB 上に発生する異常の兆候がどのような異常に繋がるのかを管理者の視点で考察した結果を以下で述べる。

4. 1 学生サービスを事例とする異常の兆候

システム上で発生する異常の中で管理者が予想しにくい問題として表 1 の想定外問題が挙げられ、一般に起りえる確率が高い異常としてアクセス集中と実行時間の遅れが挙げられる。学生サービスを事例とした場合にもアクセス集中と実行時間の遅れによる異常が発生することが考えられるため、これらの異常の兆候を見つけることが重要である。アクセス集中と実行時間の遅れの兆候は次の通りである。

(1) アクセス集中の兆候

履修期間に入り、学生が履修のために学生サービスシステムから履修システムにデータ追加の依頼をした。このとき多くの学生が一斉に履修システムにデータ追加の依頼をしたためサービスの呼び出しが頻繁に起こり、ESB の処理時間に遅れが発生する。

(2) 実行時間の遅れの兆候

サービスシステムから web サービスの要求があり、タブレットからシーケンスを実行し、結果を表示するまでの実行時間と基準時間を比べ、基準時間以上に ESB の処理を行っていた場合、処理時間の遅れという異常が発生する。

以上の異常の兆候を知るためにデータマイニングで使用するデータを表 2 に示す。

学生サービスによる履修システムにデータを追加するサービス連携を事例とした場合を以下で述べる。

OSCAR プロセスモデル^{[2][3][4]}に基づいたシーケンスの構成を図 2 に示す。

シーケンスの最初に実行する Stream(start)により、要求元の情報を取得し Sequence Log に記録する。さらに、Stream(start), Action, 及び、Coordinate の前後に実行する Stream により、実行時刻を取得しこれらも同様に Sequence Log に記録する。シーケンスの最後に実行する Stream(end)により、XMLDB の Sequence Log に蓄積された各 ESB コンポーネントの実行時刻を基に、Action と Coordinate の実行時間(ミリ秒)を計算

表 2. データマイニングで使用するデータ

項目	理由
要求元の情報	要求元の情報が許可しない同一サービスリクエストの場合、一定時間内にどれくらいの頻度でアクセスをしているのかを見ることで不正アクセスの判断をする。
日付・時刻	処理の実行時間を見ることで基準時間内に処理を完了しているかを判断する。
要求先の情報	どのサービスで異常が発生しているのかを判断する。

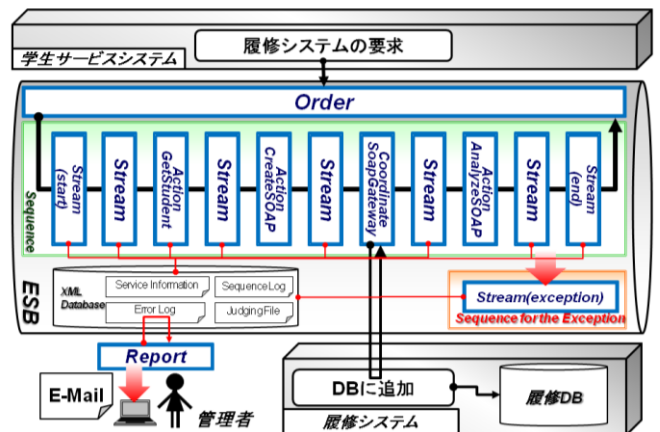


図 2. 履修システムにデータを追加する処理の流れ

する。

これらから取得したデータを元にデータマイニングを行い、異常の兆候の特定を行う。

5. おわりに

本稿では学生サービスを事例とし、ESB における「見える化」の仕組み、ESB に発生する異常を未然に防ぐための異常の兆候について検討を行い、異常の兆候について述べた。

今後は、得られたデータからデータマイニングによる分析を行い、ESB 上で「異常の兆候」を確実に特定する仕組みの実現を目指す。

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われた。

【参考文献】

- [1]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)。
- [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 pp.1212-1224 (2000-4)。
- [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)。