

SOA に基づくアプリケーションシステムの課題と提案

牧野 友紀†, 栗山 勝宏†, 羽田 昭裕†, 福地 修一†, 妻木 俊彦†

我々は、部門間ビジネスプロセス統合のシステム化事案において、分散アプリケーションの統合概念である SOA (サービス指向アーキテクチャ) の適用を試みた。SOA の分析・設計に適すると広く考えられているビジネスプロセス分析のモデリング・アプローチを用いたが、要求分析段階で実現性の観点で問題に直面した。我々は、新たにロールの概念を導入しモデルの実現性を考慮した独自のモデリング・アプローチを適用することで問題を回避した。本稿では、そのシステム化事案を取り上げ、ビジネスプロセス分析のモデリング・アプローチの問題点と新たなモデリング・アプローチによる解決方法を述べる。

An approach for modeling of application systems based on Service-Oriented Architecture

Tomonori Makino†, Katsuhiko Kuriyama†, Akihiro Hada†, Shuichi Fukuchi†, Toshihiko Tsumaki†

We tried to develop an information system using Service-Oriented Architecture because the system was constructed of some applications cross over several business organizations owning common activities. However, our model was refused by the client on account of some reasons. We thought the refusal are caused by vagueness of concept of service and defectiveness of current methods and tried to develop a new modeling method introducing the concept of the role. In this paper we report the new method and our case study using the method.

1. はじめに

インターネットの登場や規制緩和といったビジネス環境に変化により企業のビジネス活動は急激な拡張や変化にさらされている。一方、そうしたビジネス活動を支援するアプリケーションシステムは、ビジネス活動に変化がおきるたびに該当するプログラムの改造を繰り返し、ひいては、プログラム全体の書き換えを余儀なくされることになる。こうした中で、既存ソフトウェアの再利用を通してアプリケーションシステムを構築するための枠組みとしてサービス指向アーキテクチャ (SOA) [1] が提案された。

SOA は、ネットワーク上に分散配置されているアプリケーションの機能をソフトウェア部品として公開し、他のアプリケーションとの間での共有を可能とするようなソフトウェア構造を構築するための指針である。そこで公開された機能はサービスと呼ばれる。既存のサービスを組み合わせることによって複雑なアプリケーションを作成したり、ビジネスプロセスを形成するためにアプリケーションを連携させることができる [2]。このように、SOA では必要な機能を個別にプログラム化する必要が無い場合、頻繁なビジネスプロセスの変化を伴うシ

ステム開発に適している。

われわれは、実際のアプリケーションに SOA を適用することによって、その有効性を検証しようとした。その結果、新規プログラムの作成コストの低減などの便益を確認できたが、一方で、幾つかの課題に直面することになった。

本稿では、それらの課題の中から、サービスの不安定性と多重価値観という問題についての原因を解析し、それを解決するための手法について提案する。

2. SOA の適用と考察

本節では、われわれが SOA を適用したアプリケーションシステム構築の概要と、そこで発生した課題について述べる。

2.1. システムの概要

われわれが SOA によるシステム開発を試行したのは、カタログ通信販売業における物流システムである。本システムの開発に SOA を適用しようとした理由は、現行システムへの変更を極力抑制する中で、他社に比べて高い物流費用率の低減という課題を実現しなければならない、というシステム課題が与えられたからである。この課題を実現するためには、既存システムが持つ機能の再利用が不可欠であるとわれわれは判断し、SOA による既存システムの改造を試みた。具体的には、不良品対応の早期化において重要な役割を担うコールセンターのアプリケーションと商品管理部のアプリケーションを SOA の適用対象にすることにした。

† 日本ユニシス株式会社
Nihon Unisys, Ltd.
<http://www.unisys.co.jp/>

図1は、不良品の取り扱いに関する、ステークホルダー間の関係を描いたものである。コールセンターは、返品連絡を顧客から受けると、物流センターに該当商品の引き取り輸送を指示する。物流センターは、引き取り輸送があった場合には、そのことを商品管理部に報告することになっている。商品管理部は、取り扱い商品の品質に対して責任を持っており、同一商品の返品率が高い場合には、該当商品が不良品であるという判断を下し、その商品の販売停止措置を実行する。すなわち、生産管理部門に生産停止を依頼し、物流センターに対して出荷停止を指示し、コールセンターにはその旨を連絡している。

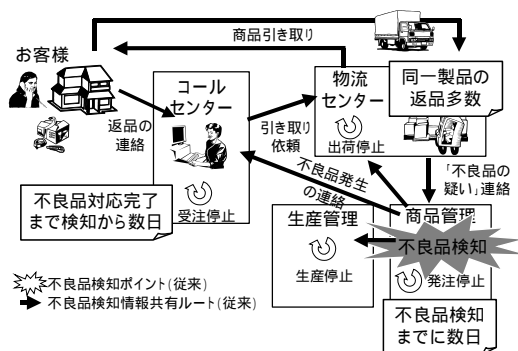


図1. 不良品の取り扱いに関する現行プロセス

2.2. SOA のモデリング・アプローチ

SOA に基づくシステム開発の基本は、共有機能を発見し、それをサービス化することにある。新たに付加されたサービスは、ソフトウェアモジュールとして実装されるが、既存システムの中に組み込まれたサービスは、インタフェースを通してアクセスされることになる。本システム開発では、新たな機能である返品統計情報処理は前者に、統計情報の収集は既存のコールセンター・システムが持つ返品管理機能を、新たなインタフェースを通して共有する後者の形式になる。

このように、SOA は、サービスという概念をもとに、実現されるソフトウェアの構造を具体的に示してくれるが、一方、サービスの分析・設計の手法に関しては未だに確立した状況にあるとは言えない。

L.Keith らは、企業活動の中からのソフトウェアコンポーネントの抽出に、ゴールという概念の導入を提案している[3]。“ゴール-サービス-グラフ”を用いてサービスの導出を試みているが、これは一種のゴール分解であり、どのレベルのサブゴールがサービスとして妥当かの判断が難しい。

実際の SOA 開発の現場では、IT ベンダー各社が様々な手法を提案しているが、Business Process Management Initiative (BPMI) によって制定されたビジネスプロセスの記述言語 Business Process

Management Notation(BPMN) [4]などを使って記述したワークフロー図からサービスを抽出するという考え方は共通している。しかし、何をサービスとするかについては、様々な考え方が提案されている。IBM は、top-down, bottom-up, middle-out という3つの視点からのサービスの識別し、それらを適度な粒度にまとめることを提案している[5]。その視点は一般的かつ汎用的であるが、その手順や基準の具体性に欠けている。

われわれが行った基本的な分析手順は、次の3つのステップからなる。

(1) ビジネスゴールの設定

最初のステップは、ビジネス活動の中の課題を解決するための目的を設定することである。これは、新たなシステム構築の目標設定を意味する。

多くのSOA構築手法が、現行のビジネスプロセスから、サービスを識別しようとしているが、本事例では、“物流費用率の低減”というビジネス課題を解決する必要があったため、現行のビジネスシステムではなく、問題が解決された後のビジネスシステムをシステム化の対象としなければならなかった。“物流費用率の低減”をゴールとしたゴール分解により、“届け配送数と引き取り輸送の数の低減”というサブゴールを通して、“不良品の配送の早期停止”という本来不要な輸送の削減につながる実行可能なサブゴールが導出された(図2)。

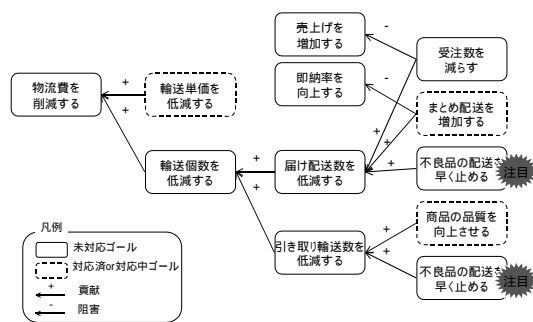


図2. ビジネス・ゴール

(2) ビジネスプロセス・モデルの作成

次のステップでは、導出されたビジネスゴールを実現するためのビジネスプロセスを設計し、それを詳細化する。現行のビジネスプロセスから、導出されたサブゴールを実現するための将来のビジネスプロセスを設計する。

現状のビジネスプロセスを解析した結果、不良品の決定は、商品管理部が、コールセンターからの返品受け取り指示により、引き取り作業を行った物流センターからの報告に基づいて行っているため、判定までに時

間がかかってしまい、そのことが、引き取り輸送の増加に繋がっていることが判明した。このことから、新たなビジネスプロセスでは、商品管理部が、最初に顧客からの返品依頼を受けたコールセンターからの報告に基づき、返品率の統計を取り、返品率が一定の閾値を超えた商品に関し、不良品候補とすることになった。

この処理をワークフローモデルによって詳細化したのが図3である。記法はBPMNに基づいている。

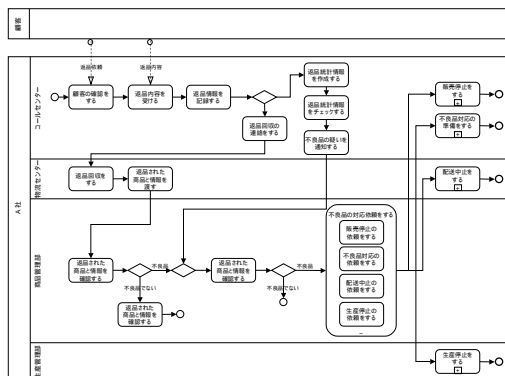


図3. 不良品判定に関するワークフローモデル

(3) サービスモデルの作成

最後のステップでは、ビジネスプロセスの中からシステム化すべきサービスを識別し、それらを整理する。

上記ワークフローモデルでは、それぞれの組織で担当するビジネスプロセスが、ビジネスアクティビティという形に詳細化されている。この詳細化は、前述したサービス識別の top-down アプローチに該当する。top-down アプローチでは、適切な粒度のサービスを識別するためにビジネスユースケースの導出を試みる方法もあるが、本事例ではワークフローモデルの詳細化によって得られたアクティビティが、適切な粒度のサービスとして同定できると判断されたので、これを最小単位の基本サービスとした。

次に、これらの基本サービスを、システム化という観点から、適度な粒度のサービスに纏め上げ、上位のサービスを構成してゆく。ここで言うシステム化の観点とは、プログラムモジュール化に適した粒度という程度の意味で、具体的にはモジュールの独立性や実行効率などを基準としているが、本事例では筆者らの経験則によって判断することにした。結果を図4に示す。ここでは、“問い合わせ受付”、“返品統計”、“不良品発生後対応”という3つの上位サービスが識別され、更にそれらを纏め上げた“コールセンター不良品対応”というサービスが識別された。

この纏め上げは、サービス識別の bottom-up アプローチに該当する。本事例では Key-Process-Indicators (KPI) が設定されていなかったため middle-out アプロ

ーチは採用しなかった。middle-out アプローチによってサービスを識別するには、SOA 設計の前に、KPI を設定するためのビジネス分析が十分に行われている必要がある。

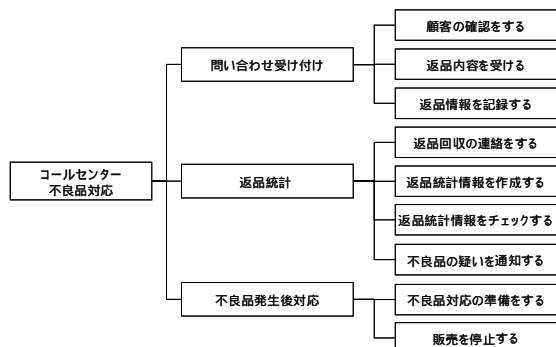


図4. サービスの構造

2.3. SOA 適応における問題

本事例では、上記サービスモデルは、コールセンターでの作業量の増加は受け入れられないという理由によって、クライアントから実行許可を得ることが出来なかった。すなわち、現在のコールセンターの作業量と現要員数のバランスは適正であり、作業量を増加することは困難である。本システムの解決提案には、多くの不確実性が含まれているため、それを元にコールセンターの定員増加を認めることは出来ないということであった。こうした問題が発生する原因としては、SOA の構造的な問題と概念上の問題が考えられるが、SOA の構造は、基本的にクライアントサーバーモデルの発展形であり、それ自体に問題が存在するとは考えられなかった。したがって、問題はSOAの構成要素の概念上の問題と考えられ、それは、システム構築手法の問題として認識されることになる。

検討の結果、われわれは、これらの問題の原因を次の3点に帰着させた。

- ・ 部門固有の活動条件が考慮されていない
- ・ サービスの効果が提示できていない
- ・ サービスの単位が曖昧である

まず、部門固有の活動条件を考慮するためには、各部門が持つ活動条件、特に、制約事項や部門固有の課題や価値観などを、モデル化プロセスの中で取り扱える必要がある。本事例では、業務を遂行するうえでの各コールセンターの予算、リソース、権限などを考慮したうえで、不良品の早期対応策が既存業務に与えるインパクトを考察することが必要であった。企業内における組織の活動条件をサービスに反映させるための枠組みを設定する必要がある。

また、効果の根拠を提示するためには、それぞれのサービスとゴールの関係を明確にする必要がある。事

例では、それぞれのサービスのゴールが、不良品の早期対応という上位ゴールにどのように寄与するかを具体的に示す必要があった。実際には、ビジネスゴールを設定し、そこから新たなビジネスプロセスを設計したが、サービスの元となる個々のアクティビティとゴールの関係に断絶があり、そのため、新規アクティビティのゴールが、既存アクティビティにどのように作用するのかも不明であった。アクティビティのゴールとビジネスゴールを関連付ける新たな機構が必要である。

こうした問題の原因の一端は、サービスという概念が曖昧であるためである。この曖昧性のために、ビジネスにおける多様な価値観とサービスとの安定的な関係をモデル化することが困難であった。多くの SOA 構築手法では、プロセスモデルの一種であるワークフローモデルからサービスというソフトウェア・モジュールを識別しようとするが、それは構造化手法におけるモジュール化と同様、極めて抽象的な概念に基づくものであり、組織の活動条件やサービスの効果といった現実世界の価値観へのフィードバックは困難である。オブジェクト指向分析設計におけるような具体的なサービスの写像対象を設定することが重要である。

3. ロールに基づくモデリング・アプローチ

上述したように、現行の SOA 構築手法の課題を、われわれは次の3点に絞った。

- ・ 企業内における組織の活動条件をサービスに反映させるための枠組み
- ・ アクティビティのゴールとビジネスゴールを関連付ける新たな機構
- ・ 具体的なサービスの写像対象の設定

この3つの課題を解決するために、われわれは組織の役割というものに注目した。組織が持つ活動条件は、企業風土のようなものにも影響を受けるが、それは組織の上位概念としての企業の特性であり、むしろ、その企業における組織の役割によって規定される部分が多い。組織の活動条件は、組織の役割の属性と考えることができる。組織は単独では存在せず、それゆえ組織の役割は他の組織の役割との協力によって目的を達成することになる。このように、組織の役割は目的という概念を誘導することが出来る。しかし、企業における現実の組織が、その企業における役割に対応しているとは限らない。むしろ、いくつかの役割を実現するために構成されたのが組織であると考えた方が現実に近いだろう。こうした考察から、われわれは役割と言う概念をサービスに同定させることとした。役割をモデル化することによって、それぞれの組織が与えられたゴールを効率的に実現するための活動条件が明示化される。これが組織の活動条件である。

役割(ロール)という概念は、もともとオブジェクト指

向分析設計におけるオブジェクトの識別のための手段として提案されたものである。オブジェクトは、仕事を行う場合、他のオブジェクトとの協調関係(コラボレーション)の中で互いの責務が定義され、それらの責務の集合をロールという概念で表していた[6]。オブジェクトモデルの中では、互いに関連するオブジェクト同士の特定の関係を表すのにロールという用語が用いられた[7]。その後、エージェントモデリングの中などでも役割という概念が使われるようになり、そこではトップダウン的に役割が定義され、その詳細化をととして責務が定義されている[8]。

われわれは、ビジネスゴールを実現するための活動主体という意味で役割を位置づける。すなわち、役割とは組織や作業者の抽象であり、主体的なものであるという意味で、以後、ロールと呼ぶことにする。ロールは他のロールとの協調関係を通して与えられたゴールを実現する。ロールは具体的なアクターに割り当てられる。ゴールを実現するために、アクターは、異なった相手ロールとの間で異なった協調関係を持つ。すなわち一人のアクターは複数のロールを持つことができる。ロールはゴールを実現するための責務を持ち、アクターは責務を具体化したアクティビティを持つ。順番付けられたアクティビティの集合をプロセスと呼ぶことにする。(図5)

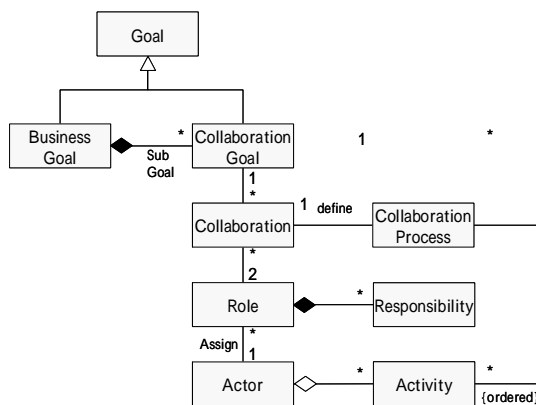


図5. ロールの概念

3.1. 新しいモデリング・アプローチ

SOA に基づいたシステムを構築する上で、ロールという概念を導入することの目的は、次の通りである。

- ・ ビジネスゴールの視点で捉えたアクターの抽象概念であるロールを、サービスの写像対象とする。
- ・ 個別のアクターの活動条件を、ロール間の協調関係を通して捉えモデル化する。
- ・ ロール間の協調関係のゴールを企業のビジネスゴールに連携させる。

協調関係とゴールを元にしたモデリング技法として i^* が提案されている [9]。 i^* は、アクター同士の関係にゴールを定義し、目的手段分解を通してタスクを導出しようとする手法であるが、アクター同士の協調関係に基づいた分析手法であることと、その分析過程でソフトゴールと呼ばれる活動条件を明示的にモデル化するところに特徴がある。初期の i^* は比較的小規模な問題を扱っていたために、アクターとロールは同一視されていたが、より大きな問題を扱うために、最近ではロールという概念を取り込み初めている [10]。 i^* の概念は、われわれのモデリング条件に合致しているところから、新たなモデリング手法を、 i^* をベースに構築することにし、 i^* に欠けている以下の点を修正、補強した。

- ・ ロールの識別とサービスへの写像
- ・ アクター間のゴールと企業のビジネスゴールの間の連携

新たなモデリング手法は、前章の3つのステップにロールの識別というステップを加えた4つのステップから構成される。

(1) ビジネスゴールの設定

ビジネスゴールを定義すると共に、ソフトゴール識別の情報を収集するために、リッチピクチャー [11] を使って、現状の活動条件を収集する。不良品対応の早期化を実現する上で各アクターの課題を分析したリッチピクチャーを図6に示す。

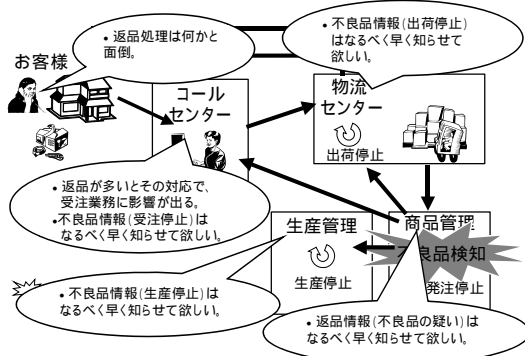


図6. リッチピクチャーによる活動条件の抽出

(2) ロールモデルの作成

特定のビジネスゴールに関わるアクター同士の協調関係を識別する。この協調関係は、一対一の関係であり、その協調関係の中で達成すべきゴールを定義することによってアクターはロールとなる。何故なら、アクターは、異なったビジネスゴールに対し、異なった協調関係を結ぶことができるからである。

リッチピクチャーに描かれた活動条件から、それぞれのロールのソフトゴールを定義する。図7は、ロール同士の関係を i^* の Dependency モデルを拡張して描い

たロール依存関係モデルである。

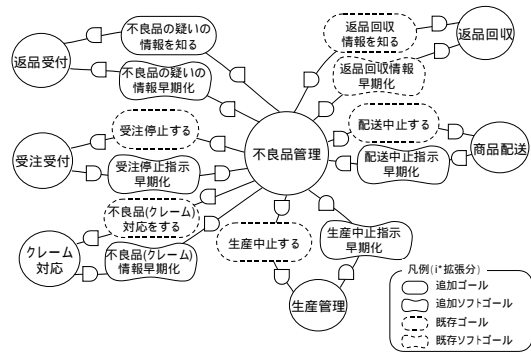


図7. ロール依存関係モデル

また、ロール間の協調関係のゴールがビジネスゴールの達成に、どのように寄与するか関連を考察し、課題が解決するか検証する。図8は、それぞれの協調関係ゴールとビジネスゴールの関係を描いたゴール寄与度関係図である。このグラフから、返品受付と商品管理部間のゴールである“不良品の疑いの情報を知る”は、“受注停止指示早期化”、“配送中止指示早期化”、“生産中止指示早期化”に効果があり、結果として不良品対応の早期化に貢献することが分かる。また、新規ゴールが既存ゴールを阻害しないことも分かる。

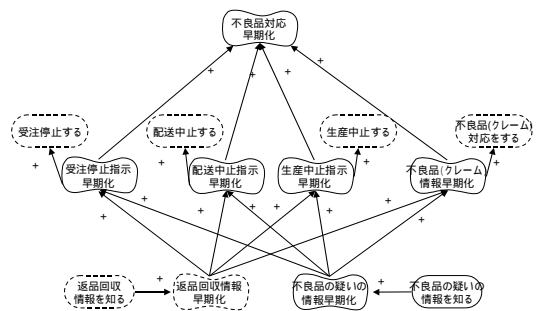


図8. ゴール寄与関係図

(3) ビジネスプロセス・モデルの作成

識別されたそれぞれのロールのアクティビティを抽出し、それを元にビジネスプロセス・モデルを描き、各アクティビティが達成するゴールがアクター内の他のロールのゴールと矛盾しないか評価する。ビジネスプロセス・モデルはロール依存関係モデルを詳細化したモデルであり、 i^* の Rationale モデルを拡張して描くことに

した。

図9は、不良品の早期対応策の鍵となる返品受付と商品管理部間のゴールに絞り、関連するアクティビティの導出を行ったビジネスプロセス・モデルである。コールセンターの返品受付では、“不良品の疑い情報を知る”ゴールのため新たなロールとして“不良品疑い監視”ロールが追加されている。そのロールでは、“不良品の疑い情報を知らせる”ゴールと関連する二つのソフトゴール、“返品情報を監視する”アクティビティと“返品情報を分析する”アクティビティを導出した。しかし、新規の二つのアクティビティは既存ロールの“低労力”ソフトゴールを阻害し、既存ロールの二つのアクティビティは新規ロールの“リアルタイム”ソフトゴールを阻害し、返品受付アクター内で内部矛盾が生ずることが分かる。このため、返品受付で“不良品疑い監視”ロールを担うことは現実性が無いと評価し、図10のように新規にアクターを設け新規ロール割り当ててことを検討した。

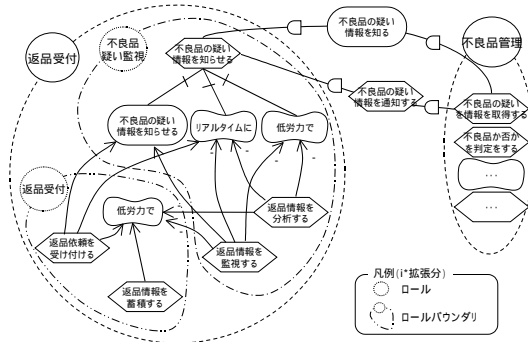


図9. 返品受付に関するビジネスプロセス・モデルの一部

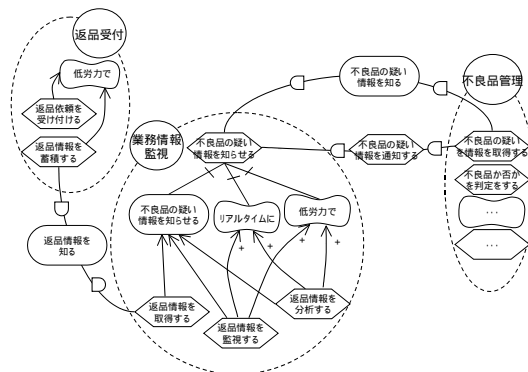


図10. 更新された返品受付のビジネスプロセス・モデルの一部

(4) サービスモデルの作成

ビジネスプロセス・モデルで導出された各アクティビティをサービスに同定し、それらのサービスを纏め上げ、サービスの構造を作成する。(図11)

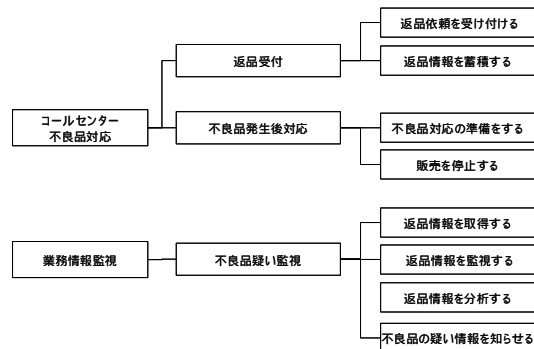


図11. ロールに基づくサービスモデル

3.2. 考察

本稿で取り上げた不良品対応の早期化では、ロールの概念を導入したモデリング・アプローチによりこれまでのSOA構築手法が抱えていた課題の一部を克服することができた。

この事例に見られるように、企業では、通常、コールセンター、商品管理部、物流センター、生産管理部のように、各部門は組織的に互いに独立して運営されている。各部門は独立した予算を持ち、独自の事業戦略により資源やビジネスプロセスの運用管理を行っている。部門間では相手部門の内部的な方針や運用に干渉することは無く、強制力は及ばないのが普通である。セキュリティなどCSR関連業務を除き、主たる業務活動において部門を横断するビジネスプロセスを統合的に管理する主体が存在することは稀である。したがって、部門間の協調の多くは、二者間の調整を関連する部門で積み上げることで実現することが多い。そこでは、ビジネスゴールや役割は部門間で認識されるが、それぞれの役割を実行するためのアクティビティに、他の部門が関与することは無い。部門をアクターに同定し、その部門が持つべき役割をロールとすることによって、複数のロールを担うアクターをモデル化したり、ロール単位に設計されたアクティビティをアクターに割り当てることによって組織変更に容易に対応することができる。このように、ロールベースのモデリング・アプローチは、部門間の協調的な業務活動の実態に適合しており、部門間の緩やかな協調をシステム化するための分析・設計に適していると考えられる。

SOAが目指すソフトウェアのモジュール化には、二つの目的がある。一つは、部門間や企業間のソフトウェアをモジュール単位に動的に連携すること、もう一つは、複数のアプリケーションで共通に利用するソフトウ

ェア・モジュールを準備しアプリケーション構築の生産性と品質を向上させることである。前者のソフトウェア・モジュールをロール単位で利用するソフトウェアと考え、後者のソフトウェア・モジュールを詳細なレベルのアクティビティを支援するソフトウェア・モジュールと考えることで、アーキテクチャとモデリング手法に一貫性を持たせることができる。ロール単位のソフトウェア・モジュールにおけるサービスは、他のアクターに提供するインタフェースとなり、詳細なアクティビティを支援するソフトウェア・モジュールにおけるサービスは部門で共通に利用する機能になる。

4. 今後の課題

本稿で述べたロール・ベースのモデリング・アプローチは、未だ、特定の事例での適用を試みた段階であり、今後、より多くの適用経験を通して手法として確定してゆく必要がある。

本稿では、ロールやアクティビティをモデル化するのに i*法の記法を流用したが、ロールとアクターの記述には難があり、独自の記法を考える必要がある。

また、クライアントとの合意を得るには、問題に対する解を定量的に示す必要があり、ゴール寄与度の定量化も課題の一つである。

5. まとめ

本稿では、SOA に基づいた現状のアプリケーションシステム構築の問題点を、実際のケーススタディを通して分析し、ロール・ベースのモデリング・アプローチという手法によって解決を図り、その有効性を示した。今後、本モデリング・アプローチに更なる洗練を加えて行くつもりである。

参考文献

- [1] W3C Web Services Addressing Working Group, Web Services Architecture, <http://www.w3.org/TR/ws-arch/>
- [2] Dirk Krafzig, Karl Banke, Dirk Slama, Enterprise SOA: Service-Oriented Architecture Best Practices, Prentice Hall, 2004.
- [3] Keith, L. and Ali, A. "A goal-driven approach to enterprise component identification and specification", *Comm. ACM* 45(10), 2002, pp45-52.
- [4] BPMI.org, Business Process Modeling Notation (BPMN) Version 1.0, [http://www.bpmi.org/downloads/BPMN-V1.0.pdf\(2004\)](http://www.bpmi.org/downloads/BPMN-V1.0.pdf(2004))
- [5] Ali Arsanjani, Service-oriented modeling and architecture, [http://www-128.ibm.com/developerworks/webservices/library/ws-soa-design1/\(2004\)](http://www-128.ibm.com/developerworks/webservices/library/ws-soa-design1/(2004))
- [6] Wirfs-Brock, R. et al. *Designing Object-Oriented Software*, Prentice Hall, 1990.
- [7] Rumbaugh, J. et al. *Object-oriented modeling and design*,

Prentice Hall, 1991.

- [8] Wooldridge, M., Jennings, N.R. and Kenny, D. "The Gaia Methodology for Agent-Oriented Analysis and Design" *Autonomous Agent and Multi-Agent Systems* 3, 2000, pp.285-312.
- [9] Yu, E. "Towards Modelling and Reasoning Support for Early-Phase Requirements Engineering", *Proc. of 3rd IEEE International Symposium on Requirements Engineering*, 1997, pp.226-235.
- [10] van der Raadt, B., Gordijn, J. and Yu, E. "Exploring web services from a business value perspective", *Proc of 13th Requirements engineering conference*, 2005, pp.53-62.
- [11] Checkland, P. *Systems Thinking, Systems Practice*. John Wiley & Sons, 1981.