

## ラショナル統一プロセスによる実践的SOA構築手法

永田 葉子 中島 千穂子 向山 奈美子 亀田 綾 三上 徹也

## Practical Methods for Building SOA Using the Rational Unified Process

Yohko Nagata Chihoko Nakajima Namiko Mukouyama Aya Kameda Tetsuya Mikami

SOA( Service Oriented Architecture )はビジネス環境の変化に迅速に対応するためのITアーキテクチャーとして業界全体の注目を浴びている。しかし、構築に踏み切る企業はまだ少なく、調査機関によればその主たる原因は開発手法が確立されていないことであると報告されている[ 1 ]。筆者らはRUP( Rational Unified Process® )をSOA構築のための現実解の一つとして適用するために、「プロセス」「トレーサビリティ」「再利用」の三つの観点に注力して実践的なSOA開発プロセスにカスタマイズした。さらにプラクティショナーに展開した上で、アンケート調査によって有効性を確認した。

Service-Oriented Architecture (SOA) has been spotlighted throughout the industry as the architecture for responding promptly to changing business environments. However, few enterprises have yet embarked on building SOA. Research firms report that the main reason for this is that methods for developing SOA have not yet been established [1]. In order that the Rational Unified Process® (RUP) be able to be applied as a realistic solution, we customized it into a "practical SOA development process" by focusing on the three aspects of "process," "traceability," "reusability." We have rolled out this process to practitioners, and their evaluation has authenticated its effectiveness.

Key Words & Phrases : SOA ,RUP ,ビジネス・モデリング ,ソフトウェア再利用 ,トレーサビリティ  
SOA, RUP, business modeling, software reuse, traceability

## 1 .はじめに

ビジネス・トランスフォーメーションや企業の吸収・合併など、ビジネス環境の変化に迅速に対応するためのIT基盤としてSOA( Service Oriented Architecture )が注目されている。SOAは「柔軟性」「再利用性」「拡張性」「統合性」「管理性」を劇的に向上させる普遍的なシステム・アーキテクチャーとなる可能性があり、業界やベンダーを問わずSOAへの関心が急速に高まっている。

一方、SOA構築の要となる開発手法については、徐々に整備されつつあるものの混沌としているのが現状である。ビジネス・モデリングの領域ではCBM( Component Business Modeling [ 2 ] )がビジネスをコンポーネントとしてモデリングし、問題領域やコアコンピタンスの発見を目的としているのに対し、SOMA( Service-Oriented Modeling and Architecture [ 3 ] )はシステム化対象となったビジネス・プロセスからサービスの候補を識別しデザインすることを目的としてい

る。CBMからSOMAへの連携はビジネス・モデリングの一つのパターンではあるが、両者共通の表記法やメタモデルは存在しない。その他代表的なものとしては構造化分析・設計から発展したIDEF( Integrated DEFinition methods [ 4 ] )、オブジェクト指向分析・設計から発展したEriksson-Penkerらの手法[ 5 ]がある。ビジネス・モデリングの成果物はシステム開発プロセスの入力となるのが理想である。しかしIBMの標準的なシステム開発プロセスであるADSG( Application Development Standardization Guide )、ADSG for e-Businessなどを見ても、ビジネス・モデリングの成果物とは連携していない。

このような混沌とした状況を解決するには以下の整備が必要である。

- ・ビジネスからITまでをカバーする一貫したプロセス
- ・ビジネスからITまでのシームレスな連携をサポートするモデル

そこで筆者らはビジネスとITのモデルをUML( Unified Modeling Language )という共通の言語で表現し、一貫した開発プロセスを定義しているRational Unified Process®( 以下RUP )をベースにSOA構築の

提出日 : 2005年8月16日 再提出日 : 2006年9月19日

ためのカスタマイズを行って、実践的な手法を定義した。本論文ではRUPをSOA構築プロセスとして適用するために注力した点を「プロセス」「トレーサビリティ」「再利用」の観点から論じる。

以下、2章でSOA開発プロセスとしてのRUPの親和性とカスタマイズについて述べた上で、3章でそのアプローチによるビジネスモデルからITモデルまでのトレーサビリティの実現、4章でサービスの再利用を目指したアセットベースのSOA構築を述べる。さらに5章で本手法の評価・考察を行い、6章でまとめと関連ツールのバージョンアップの影響について補足する。

## 2. SOA開発プロセスとしてのRUP

### 2.1 SOA開発とRUPの親和性

筆者らは、以下の4点においてSOA開発とRUPの高い親和性を見出し、SOA開発手法としてRUPを採用した。

#### (1) ユースケース駆動

SOAは「サービス」の組み合わせによってアプリケーションを構成するシステム構築の考え方である。その粒度の定義には議論のあるところであるが、「サービス」とはビジネス上の価値を持つ単位といえる。ユースケースもアクターへの価値の提供であり、「サービス」をユースケース・モデリングすることは極めて有効である。

#### (2) UMLによるオブジェクト指向モデリング

RUPはオブジェクト指向モデル(UML)でビジネスの構造(サービスの構造)とITの構造の双方を記述する。したがって、ビジネスの抽象化モデルとITの抽象化モデルのセマンティックギャップが小さくなり、それぞれのモデルに登場するオブジェクトの関連を定義するメタモデルを作成して、両モデルの変換機能を定義することが可能になる。

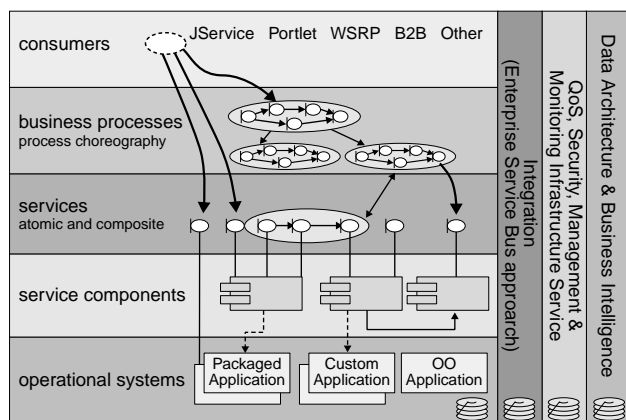


図1. SOAレイヤー概念図 [ 3 ]

#### (3) 再利用

SOAの利点としてサービスの再利用が考えられているが、サービスは再利用対象アセットの一つに過ぎない。RUPはサービスのみならずSOAのアーキテクチャーを再利用し、生産性や品質を向上させる再利用の仕組みを備えている。図1はIBMが定義したSOAレイヤー概念図であるが、この各SOAレイヤーを再利用アセットの分類として位置付け、レイヤーごとにパターンを適用できる。詳細は4章にて述べる。

#### (4) ソフトウェア開発環境

ビジネス要求である「サービス」をITシステムとして実装するためには、ビジネスからITを包括的にカバーするツールのサポートが不可欠である。Rational®の提供するSoftware Development Platformは、ビジネス駆動のソフトウェア開発プロジェクトを支える総合的な開発プラットフォームであり、開発プロジェクトにおいて、Rationalブランドのツールだけでなく、WebSphere®ブランドのモデリング製品やTivoli®ブランドの運用管理製品と連携し、開発ライフサイクルの各役割に対応したラインナップを形成している。さらに、RUPはこれらのソフトウェアを有効に活用するためのガイドとしてツールメンターを提供しており、ソフトウェア開発環境の観点でSOAに大きく貢献する。

### 2.2 RUPのカスタマイズ

RUPではプロセス・ワークフローとして六つ、補助ワークフローとして三つ、そしてRUPプラグインとして「再利用」のワークフローを提供している(図2)(各ワークフローの詳細に関しては [ 13 ] を参照のこと)

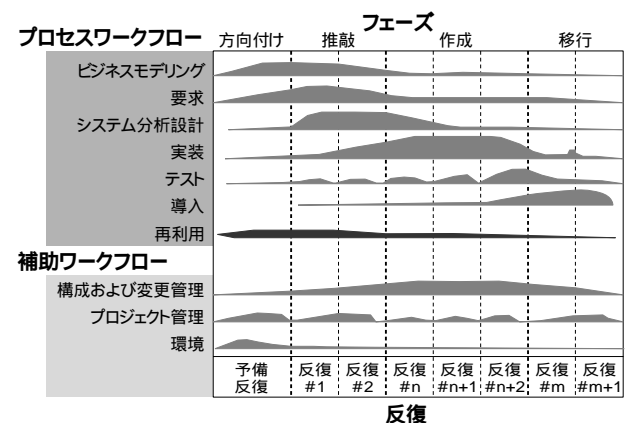


図2. RUPのフェーズとワークフロー [ 13 ]

RUPの構造はメタモデルとして定義されている。したがって、RUPのワークフローはカスタマイズが容易であり、目的に応じた種々のプラグインが提供されている。

筆者らはSOA開発プロセスとして、RUPの、特にビジネス・モデリングとシステム分析/設計のワーク

ローに注力してカスタマイズを加え、実践的なSOA開発手法を定義した。カスタマイズの内容は以下の通りである。

- (1) ビジネス・モデリングでのWBI Modelerの採用
- (2) アーキテクチャー・パターン[7]としてのSOA  
 それぞれの内容の概要については以下で、詳細については次章以降で解説する。

(1) ビジネス・モデリングでのWBI Modelerの採用  
 ビジネス上の価値の単位であるSOAの「サービス」開発のためには、その源流となるビジネス・モデリングが重要である。本手法では、RUP plug-in for WBI Modeler V1.0 (Beta)[6]をベースとして、WebSphere Business Integration Modeler (WBI Modeler)を組み込み、SOA構築手法としてRUPをカスタマイズした。

WBI Modelerのシミュレーション機能によるビジネス・プロセスのパフォーマンス評価(コスト、時間、リソース配置など)を作業として追加し、システム化への根拠を提示することとした。さらに、作成されたビジネス・プロセス・モデルをトレーサブルにITモデルへ連携するための手順を定義した。

- (2) アーキテクチャー・パターンとしてのSOA  
 SOAレイヤー概念図(図1)は、特定のセマンティクスに基づいたモデルではないため、ITモデルへ変換できない。そこで、本手法では、SOAレイヤー概念図をUMLのパッケージ図として表記し(図3)、アーキテクチャーを表現する「4 + 1ビュー[11]」の一つである論理ビューとして使用した。

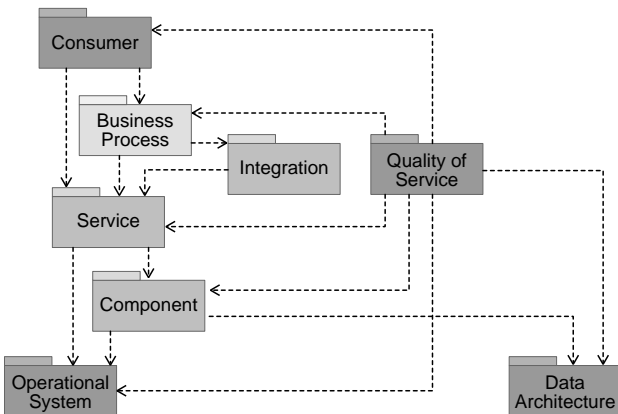


図3. SOAのアーキテクチャー論理ビュー

ビジネス・モデリングにより定義されたビジネス・ユースケースから、ビジネス・アーキテクチャーを構成するビジネス・プロセスが作成され、SOAアーキテクチャーのBusiness Process層に配置される。そして、ビジネス・プロセスからシステム・ユースケースが識別され、分析モデルを経由して設計モデルとしてService層、Component層、Data Architecture層などへ配置される。アーキテクチャー・パターンとしてのSOA

レイヤー概念図を利用し、このSOAレイヤーごとに、製品を使用するか、責務を実現するクラスを分析設計ワークフローにより開発することでSOA開発を実現する。

### 3. ビジネス・モデルからトレーサブルなITモデル

現在、モデリングや開発を行うツールが複数提供されており、ツール間の連携機能も備えているが、各ツールを使用するフェーズと作成する成果物については体系化されていない。そこで本手法では、SOAと親和性の高いRUPによる一貫したアプローチの中で、ツールの役割と成果物を明確に定義した。本章ではRUPのプロセス・ワークフロー(図2)の内の、ビジネス・モデリング、要求、システム分析/設計、実装の四つについてその内容を述べる。

#### 3.1 ビジネス・モデリング

ビジネス・モデリングでは、ビジネス構造を可視化することで課題を明確にし、改善を支援するためのシステム化要求を引き出していく。

この作業分野では通常、ビジネス・ユースケースを作成し、そのフローをアクティビティ図で記述する。本手法では、アクティビティ図の代わりに「ビジネス・プロセス・モデル」をWBI Modelerで作成することとした。WBI Modelerでの組織や人、コスト、時間などを含めたエンタープライズ・モデリングにより、ビジネス構造がより明確に可視化されることとなる。

ビジネス・プロセス・モデリングの手順は以下の通りである。この段階でのWBI Modelerのモデリングは、「実装テクノロジーを意識しない「オペレーショナル・モード」にて行う。

- (1) 「ビジネス・ユースケース」を入力とし、イベント・フローを「ビジネス・プロセス」として、プロセス内の各タスクの担当者を「リソース」として記述
- (2) 「ビジネス・エンティティ・モデル」をRSA (Rational Software Architect)で作成し、「ビジネス・アイテム」へ反映  
 「ビジネス・アイテム」はデータの構造やプロセス内のデータ遷移を表現するために使用するが、データ同士の関連を表現するモデルではない。そのため、RSAで作成した「ビジネス・エンティティ・モデル」をWBI Modelerの「ビジネス・アイテム」へ反映することとした。

ビジネス・プロセス・モデルは「As-Is(現状)」から「To-Be(あるべき姿)」へ、さらに「システム化を考慮したTo-Be」へと洗練を重ねる。この間、WBI Modelerの「シミュレーション機能」を活用し、「ビジネス・プロセス・モデル」をコストや時間の観点などから定量的

に分析していく。この段階で使用したパラメーターから、ビジネス・パフォーマンスを最適化するための機能外要求を導出できる。さらには、これらのシミュレーションや分析の結果をビジネス・アーキテクチャー説明書に記述することで、システム化の判断根拠として後続の作業で参照することが可能である。

WBI Modelerの「ビジネス・プロセス・モデル」は「システム化を考慮したビジネスTo-Beモデル」として、「ビジネス・エンティティ・モデル」「ビジネス・ユースケース・モデル」とともに次工程に引き継がれる。

### 3.2 要求

要求の大きな目的はシステム化の範囲を決定し、利害関係者間での合意を得ることにある。本手法では、ビジネス・モデリング作業により「システム化を考慮したTo-Beモデル」まで作成したことにより、問題分析などの要求の初期作業を軽減することができた。システムの境界線を明確にするためにアクターの洗い出しが最初の作業となるが、ここではビジネス・アクターをアクターとして再設定し、「ビジネス・プロセス・モデル」の「リソース」をアクター候補として検討する。そのリソースが果たす役割が自動化される場合は、アクターとして抽出してはならない。

アクターの洗い出しの次に、ユースケースを獲得する。その際、本手法ではビジネス・モデリングで作成した「システム化を考慮したビジネスTo-Beモデル」を入力とした。このモデルは自動化を前提としているため、各タスクは一つのシステム・ユースケース候補として抽出することができる。また、ビジネス・プロセス全体を一つのユースケースとして抽出した。これはプロセス全体を自動化対象とした結果であり、プロセスの実行を制御するユースケースとして獲得は必須である(図4)。

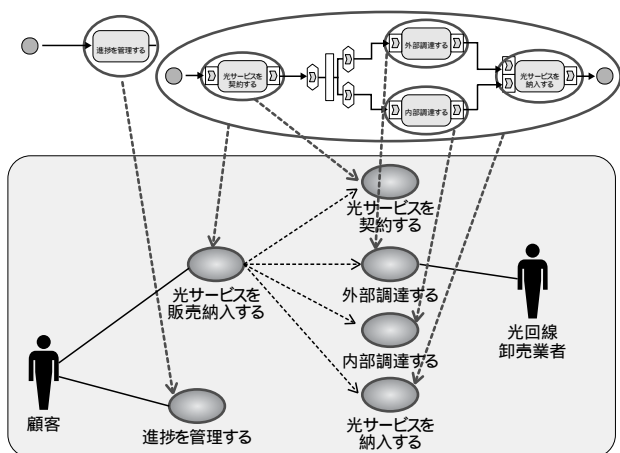


図4. ユースケースの抽出

### 3.3 システム / 分析設計

システム分析/設計では要求を変換し、システムのアーキテクチャーを洗練していく。この作業分野で使用するツールは主にRSAである

本手法では2.2(2)で述べた通り、SOAをアーキテクチャー・パターンとして採用し、論理ビューを作成した(図3)。ここで作成したパッケージの名称は、各レイヤーの名称と一致させた。これにより、新たなSOAのステレオタイプを定義したことになる。

さらにユースケースの振る舞いから分析クラスを抽出し、振る舞いを振り分け、分析モデルを完成させていく。続いて、分析モデルから設計モデルへの変換を行う。本手法ではSOAのレイヤーを基に設計メカニズムを定義し、各レイヤーの内部を構成する責務を表すステレオタイプとして定義した(表1)。

表1. SOAレイヤーのステレオタイプ

レイヤー	レイヤーに配置される設計ステレオタイプ
<<Consumer>>	<<Internet>> <<Intranet>> <<B2B>>
<<Business Process>>	<<ProcessDefinition>> <<ProcessEngine>>
<<Service>>	<<ServiceSpecification>>
<<Component>>	<<Facade>>
<<Integration>>	<<ExposedGateway>> <<MessageBroker>> <<Router>>
<<Data Architecture>>	-
<<QoS>>	-
<<Operational Systems>>	<<PackagedApplication>> <<CustomerApplication>> <<OOApplication>>

分析によってステレオタイプ化されたクラスは、これらの設計ステレオタイプで型付けし、各レイヤーに配置していくことで設計クラスを完成させていく。設計クラスの実装方法は、ステレオタイプごとに製品を使用するか、開発を進めることになる。

Service層では「Webサービス」が、Business Process層では「BPEL(Business Process Execution Language for Web Services)[12]」が、それぞれの責務を実現するテクノロジーとして上げられる。Webサービスは「明確に定義されて自立している」「オープンな標準技術を使ってアクセスされる」「内部の実装を隠蔽しており、実装方法は問わない」など、SOAにおける「サービス」の技術的な特徴[8]をXML(Extensible Markup Language)、SOAP(Simple Object Access Protocol)といった標準技術で実現したものであり、BPELはWebサービスをビジネス・プロセスとして実行していくためのXMLベースの記述言語である。BPELはOASIS(Organization for the Advancement of Structured Information Standards)で標準化が進められており、ビジネ

ス・プロセス実行言語の業界標準としての地位を確立しつつある。

Business Process層の実行形態としてBPELを選択する場合は、WBI Modelerで作成した「システム化を考慮したビジネスTo-Beモデル」を基に、テクノロジー・モードを「BPELモード」に変更して洗練し、設計モデルを作成していく。その際には、RSAでの分析設計の結果により発生しうるインターフェースの変更を反映することも忘れてはならない設計者の責任である。

### 3.4 実装

RSAではトランスフォーメーション機能により、設計モデルからJava™やEJB(Enterprise Java Beans)のスケルトン・コードを生成することが可能である。また、WBI Modelerのエクスポート機能により、BPEL/XSD/WSDLといった定義体を自動生成することが可能であり、これらの定義体をWSAD-IE( WebSphere Application Developer Integration Edition )にインポートすることにより、実装が進められる。このようにツールの機能を活用することによって、これまでに作成したモデルから実装のためのコード・テンプレートを作成できる。

この作業分野ではRSAで作成したスケルトンに実装コードを追加し、デプロイメント・コードを生成していく。さらに、このデプロイメント・コードをWebサービス化するという作業を経て、WSAD-IE上で実行時に必要となる情報( WSDLのBinding情報など )を追加する。

上記の手順でSOAの各レイヤーの実装を完成させる。図5では本手法で使用したツールと作成したモデルを示しており、ビジネス・モデルからITモデルまでのトレーサビリティが確認できる。

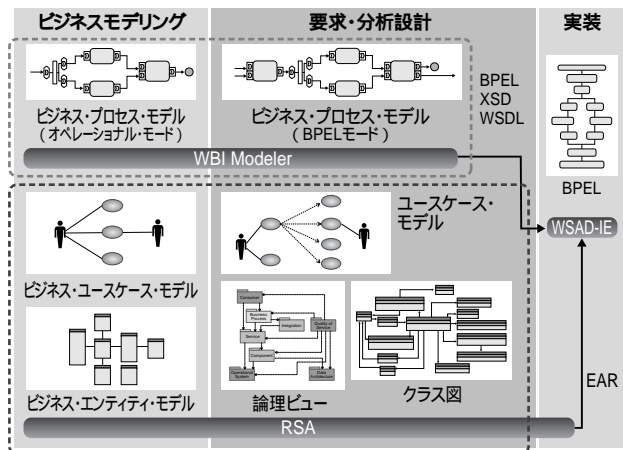


図5. モデルのトレーサビリティ

なお、今回使用したツールでは、設計から実装への自動化は可能であったが、それ以外のツール間の

連携は全て手法に基づき手作業で行った。他の作業分野でのツール間の連携については、今後のツールの発展に期待するところである。また今回定義した設計メカニズムとそこで定義したステレオタイプの洗練、サービス実装手順の精緻化も、今後のプロジェクトへの適用やツールの高機能化を反映しつつ反復的に進めるべきと考える。

## 4. アセットベースのSOA構築

SOAは「サービス」の再利用を目指しているが、さらに本手法ではSOA構築時のアーキテチャーやプロセスを再利用することにより生産性の向上と品質の安定化を目指した。

### 4.1 SOAとアセットベース開発(ABD) [10]の親和性

筆者らは、SOAのシステム構築時に用いた手法や作成した成果物をアセット化し、再利用するプロセスとしてABDが最適と判断した。ABDはRUPとの親和性が非常に高く(2.2参照)、非常に柔軟で幅広い要素をアセットとして登録し、RAS(再利用可能アセット仕様)化することが可能である。さらに、ABDはRSAとの親和性も高く、RSAでは作成した成果物をRAS化し、再利用していくことが可能である。ABDの適用により、前述してきたSOAの様々な要素をアセット化することが可能となる。その中で最も価値のあるアセットを以下で論じる。

#### (1) パターン・アセット

SOAの適用を再利用の観点から見ると「アーキテチャー・パターン [7]」の適用に他ならない。さらに、各層の分析設計時に適用可能な既存のパターンが多々ある。例えば、Data Architecture層にはアナリシス・パターン [9] とデザイン・パターン、Service層およびComponent層にはアーキテチャー・パターンとGoFデザイン・パターンの適用が可能である。RSAには、パターン適用機能が組み込まれ、開発中の分析設計モデルへパターンを適用することができる。そして、再利用可能な分析や設計をパターン化し、登録することもでき、組み込み済みのパターンと同様の手順で適用することができる。分析設計をパターン化し、再利用していくことで、SOAシステム構築のための分析設計標準を築き、品質の安定化へと繋げていくことが可能となる。

#### (2) UML拡張仕様アセット

前述の通り、分析モデルから設計モデルへ変換させるための設計メカニズムを「ステレオタイピング」という形で構築したが、ステレオタイプとは、UML拡張仕様のことである。SOAの各層にあたるサブシステムへの配置用に作成されたステレオタイプをまとめ

て、UML拡張仕様としてアセット化することができる。筆者らが作成したステレオタイプは、RSAでステレオタイプを定義作成し、Eclipseプラグインとして発行することで、RSAでアセットとして再利用できる。これはステレオタイプの再利用にとどまるのではなく、SOAのための設計メカニズムの再利用に値する。そして、モデル・トランスフォーメーション機能を持つRSAでは、分析モデルからステレオタイプ化された設計クラスへの変換を作成することが可能である。これによりRSAの機能で分析モデルから設計モデルへの変換が可能となり、安定的にSOAアーキテクチャーを構築していくことが可能となる。また、プラグインの導入や自動変換により安定した生産性向上が望める。

### (3) プロセス・アセットと標準化

筆者らは、SOA構築において、RUPに追加しなくてはならないプロセスの差分を構築し、ワークフローとガイドラインという形でアセットを残した。このプロセスは今後SOA構築を行う開発者の参考とし、開発の標準化のきっかけとすることができる。SOA構築の開発手法が混沌としている今、こうしたきっかけとなるプロセス・アセットの持つ意味は大きい。

## 4.2 再利用の現状と今後の課題

3種類の有力アセットについて論じてきたが、これらのアセットを再利用し、今後の開発で反復的にアセットが追加されることにより、SOAシステム構築のための材料が揃い、標準化へとつながると考えられる。特にパターン、UML仕様、プロセスは、その傾向の強い強力なアセットである。さらに、アセット化やアセットの再利用機能を持つツールにより、生産性を安定的に向上していくことが可能となる。

現状での課題はツールの機能が限定されている点だと考えられる。RSAはパターン、UML、プロセス機能を備えるものの、限定された内容で、多くを新たなアセットとして新規作成しなくてはならない。WBI Modelerは、作成したビジネス・プロセス・モデルをRAS化する機能は備えていない。統合された環境の中で標準的にアセットを構築可能なツールの提供が望まれる。

## 5. 考察

開発プロセスは、プロジェクトに適用した結果において評価すべきではあるが、第一段階としては、システム構築に携わるプラクティショナーの理解を広く得られるか否かが重要である。そこで筆者らはまず、今回の手法を実際のモデルなどの成果物とともに解説する研修を実施した。表2はその出席人数と理解度評価である。

表2. 研修アンケート結果

対象人数(出席者延べ人数)	275人
理解度平均(全く理解できない:0 よく理解できた:100)	80
有用性平均(有用でない:0 非常に有用である:100)	84

表2から筆者らのアプローチに対する十分な理解が得られるとともに、プロジェクト適用への有用性に対する評価も得られたと判断できる。一方で、ビジネス・プロセスの例外処理への考慮や具体的な工数見積手法が必要などの指摘もいただいた。現在、本手法を基にSOAの開発標準を作成し、開発に取り入れたいというお客様からのご要望を複数いただいている。今後はこれらのプロジェクト適用を積み重ね、手法としての成熟度を高めていく必要がある。

さらにSOA構築のための組織変革が必要とのコメントもいただいたが、それに対して筆者らはSOA自体のビジネス・モデルの定義が必要と考える。その中でビジネスに参画する役割と責務、およびビジネスの構造とそれを実現するソリューションの構造を明確にする。本手法は、そのためのプロセス・アセットとして活用することができる。今回はSOA構築のシステム開発部分を中心とした手法を定義したが、SOAのビジネス・モデルはシステム構築のみにとどまるものではない。SOAをベースとした新たなビジネス・モデルを創出することにより、新規分野のビジネス発掘や既存ビジネスの拡大・発展に寄与することができる。

## 6. おわりに

本論文では、「プロセス」、「トレーサビリティ」、「再利用性」の観点からRUPのカスタマイズを行い、実践的なSOA開発プロセスとして提案した。

本論文執筆後、RUPおよび論文中で扱った製品の新バージョンがリリースされるなどいくつかの変更があった。主たる変更は以下の通りである(詳細は付録参照)

- Rational Method Composer付属のプロセス・コンテンツ(RUP)にSOA開発用のplug-inが製品として組み込まれた(RUP for SOA)
- WebSphereツールは、2005年9月にWebSphereプロセス統合(WebSphere Process Integration)製品群として、新バージョンが発表された。

本手法で確立したツールの連携にツールのバージョンアップによる大きな影響はないが、WPI製品群が提供する新たなプログラミング・モデルであるSCA(Service Component Architecture)の登場が言語やプラットフォームの違いを超えた新しいアプリケーション開発のスタイルを提供していることを付け加えておく。

## 謝辞

当論文の執筆にあたりオブジェクト指向開発およびRUPに関する豊富な実践経験をお持ちの杉本 宣男氏より多くの助言をいただきました。あらためて感謝いたします。

## 参考文献

- [ 1 ] 読者調査結果, @IT, <http://www.atmarkit.co.jp/news/survey/2004/02arcmg/arcmg.html>
- [ 2 ] Component Business Model, <http://www.ibm.com/services/bcs/jp/different/cbm.html> ( 2005.2.8 )
- [ 3 ] SOMA, <http://www.ibm.com/developerworks/webservices/library/ws-soa-design1/> ( 2004.9 )
- [ 4 ] 小林 隆: ビジネスプロセスのモデリングと設計, コロナ社, ISBN-4339026298 ( 2005.3 )
- [ 5 ] ハンス=エリク エリクソン他: UMLによるビジネスモデリング, ソフトバンクパブリッシング, ISBN-479731382X ( 2002.5 )
- [ 6 ] RUP Plug-In for WBI Modeler V1.0( beta ) [http://www.ibm.com/developerworks/rational/library/05/531\\_wbi/](http://www.ibm.com/developerworks/rational/library/05/531_wbi/) ( 2005.5.31 )
- [ 7 ] F. ブッシュマン他: ソフトウェアアーキテクチャ ソフトウェア開発のためのパターン体系, 近代科学社, ISBN-4764902834 ( 2000.12 )
- [ 8 ] IBMが考えるSOA, [http://www.ibm.com/jp/software/isw/handouts/pdf/j/\\_9.pdf](http://www.ibm.com/jp/software/isw/handouts/pdf/j/_9.pdf)
- [ 9 ] マーチン ファウラー他: アナリシスパターン 再利用可能なオブジェクトモデル Object Technology Series, ピアソンエデュケーション, ISBN-4894716933 ( 2002.4 )
- [ 10 ] RUP for Asset-Based Development Plug-In v1.0, <http://www.ibm.com/developerworks/rational/library/4145.html> ( 2004.5.12 )
- [ 11 ] ソフトウェア開発4つの課題( 4 ), @IT, <http://www.atmarkit.co.jp/ad/ibm/virtualguild0406/04/ibm04.html> ( 2004.12 )
- [ 12 ] IBM WSDD Business Process Execution Language for Web Services version 1.1, <http://www.ibm.com/developerworks/library/specification/ws-bpel/> ( 2005.2.1 )
- [ 13 ] パー・クロール他: ラショナル統一プロセス RUP ガイドブック RUP実践者を成功に導く, SB Access, ISBN-4434052349 ( 2004.11 )

## 付録

RUP for SOAはRUPの分析・設計部分をカスタマイズするものであり, 本論文でとりあげたトップダウンビジネス・プロセス主導の他に, 以下のようなサービスの抽出方法が定義されている。

- ・ トップダウン ユースケース主導
- ・ ルール主導
- ・ データ主導
- ・ ボトムアップ
- ・ 既存の資産の公開

またSOAの中心要素をモデル化するためのサービス・モデルが定義されている ( プレスリリース: <http://www-06.ibm.com/jp/press/20050914001.html> )

WBI Modeler V5.1 WebSphere Business Modeler V6  
 WSAD-IE V5.1 WebSphere Integration Developer V6  
 WBISF V5.1 WebSphere Process Server V6  
 WPI製品群は, IBMが提唱するSOAプログラミング・モデル( 下記 )を実現する製品群である。

- ・ サービスを組み合わせる「BPEL」
- ・ サービス呼び出しの単一のインターフェースを提供する「Service Component Architecture( SCA )」
- ・ サービスに対して単一のデータ形式を提供するSDOベースの「Business Object( BO )」

SCAは, 本手法でService層の責務を実現するテクノロジーとしてとりあげた「Webサービス」のみならず, EJBなど各種分散コンピューティング技術を使用する側に単一のインターフェースを提供する, より高度なプログラミング・モデルである。

WSAD-IEの後継であるWebSphere Integration Developerは, このSOAのコンポーネントを開発することが可能なツールである。

また, 同時に発表されたWebSphere Business Modeler V6( 以下, Modeler )では, 新たにモデリング・モードとして「WebSphere Process Serverモード」が追加され, 「テクニカル仕様」「テクニカル属性」というビューが提供された。

分析設計でModelerを使用して最終的に作成する, 設計レベルのビジネス・プロセス・モデルには, このビューを使用してSCAやBPELに関する情報を入力することが可能である。



日本アイ・ピー・エム  
システムズ・エンジニアリング株式会社  
ビジネス・インテグレーション  
主任ITスペシャリスト

**永田 葉子** Yohko Nagata

**[プロフィール]**

Rational製品を担当するグループのグループ・リーダー .RUP ( Rational Unified Process )のスペシャリストとして様々なお客様のプロジェクトを支援すると同時に ,社内でのRational製品スキルの向上にも力を注ぎ ,社内外でのRationalソリューションの普及に努めている .

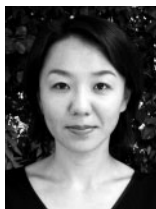


日本アイ・ピー・エム  
システムズ・エンジニアリング株式会社  
ビジネス・インテグレーション  
マネージャーICP-コンサルティングITスペシャリスト

**中島 千穂子** Chihoko Nakajima

**[プロフィール]**

日本におけるBusiness Process Management普及を推進するICP・コンサルティングITスペシャリストであり ,RationalとWebSphere Process Integrationの技術支援を担当するライン・マネージャーを兼務する .RationalとWPIを連携させたサービス開発の実践を目指し ,様々なお客様プロジェクトの支援を通して ,BPMの推進活動を行っている .



日本アイ・ピー・エム  
システムズ・エンジニアリング株式会社  
ビジネス・インテグレーション  
主任ITスペシャリスト

**向山 奈美子** Namiko Mukoyama

**[プロフィール]**

WebSphere Process Integration製品を担当するグループのグループ・リーダー .ワークフロー製品の技術支援を経て ,現在はWPI製品群を担当し ,SOAやBPM( Business Process Management )の普及に努めている .



日本アイ・ピー・エム  
システムズ・エンジニアリング株式会社  
ビジネス・インテグレーション  
ITスペシャリスト

**亀田 綾** Aya Kameda

**[プロフィール]**

WebSphere Process Integration( WPI )製品群を担当するITスペシャリスト .WPI製品群の研修開発やプロジェクト参画を通じ ,SOAやBPM( Business Process Management )の推進活動を行っている .



日本アイ・ピー・エム  
システムズ・エンジニアリング株式会社  
ビジネス・インテグレーション  
副主任ITスペシャリスト

**三上 徹也** Tetsuya Mikami

**[プロフィール]**

Rational開発製品群 ,オブジェクト指向担当のITスペシャリスト .様々なお客様のプロジェクトにて ,オブジェクト指向開発 ,RUP ,プロセス標準化の支援を行っている .アセット再利用にも力を入れ ,組織的再利用アプローチの普及に向けた活動も行っている .