

# SOAライフサイクルを支える開発方法論とツール

## — ビジネスの迅速性を実現するための実践的な開発方法論とツール —

近年、WSDL や SOAP などの実装技術や標準規約が整ったことにより、多くのお客様で SOA への取り組みが始まり、基幹システムにおいても SOA を実行技術として適用できることが実証されつつあります。

その中で現在関心を集めているのは、SOA の最大のメリットといわれるビジネスの迅速性を実現するためにビジネスとサービスを整合する手順、および SOA 化された基幹システムの確立手法です。そして SOA における開発方法論や成熟したツール、それらを利用できるスキルを持った技術者が今日求められています。

IBM では SOA を推進し始めた当初より、方法論、ツール、ならびにそれらを効率的に利用するためのアセットの整備に取り組んできました。そして多くの SOA プロジェクトでこれらを実際に利用し、ノウハウをフィードバックすることにより、ビジネスから IT までのすべての局面において利用可能な方法論とツールを確立してきました。当解説記事ではこれらを概観し、その活用のポイントについて解説します。

### ① SOA ライフサイクルを実現するモデルと開発方法論

SOA を活用してビジネスの迅速性を実現するためには、ビジネスから IT までを「モデル」として表現し、トレーサビリティを確立することがポイントになります。「モデル」は SOA ライフサイクルである「モデル-アセンブル-配置-管理」の最初の局面でもあり、ビジネスと IT の整合（アライメント）を確立する際、最も重要な成果物となります。

SOA ライフサイクルの主要なモデルには、企業全体をビジネス機能別のビジネス・コンポーネントとして表現した「コンポーネント・ビジネス・モデル（以下、CBM）」、特定のビジネス領域をロール、プロセス、アクティビティ、タスク、エンティティなどによりビジネスの動的な側面を表現した「ビジネス・プロセス・モデル（BPM）」、ビジネスと IT が提供するサービスをモデリングした「サービス・モデル」、そして、それらのサービスを IT として実現するための「コンポーネント・モデル」の 4 つがあります。このように、ビジネスのモデル

Article 3

### SOA Methodologies and Tools for SOA Lifecycle

#### - Practical Methods and Tools for Business Agility -

Since SOA implementation technology and standards are mature, recently many customers have started applying SOA as runtime technology for enterprise systems, and SOA technologies are gradually becoming applicable.

Therefore, the current concern is how to align business with IT, and how to design and build SOA enterprise systems in order to establish business agility. Customers especially want SOA methods and tools, and skilled specialists.

IBM has been establishing SOA methodologies, tools, and assets to use effectively in the early SOA phase, and IBM is providing practical methods and tools for all the SOA lifecycle phases. This article is an overview of SOA methods and tools, and describes how to use them.

を起点としてシステム開発につなげていく手法を総称して「ビジネス駆動開発」といいます。

各モデルには、それぞれ関連するさまざまな成果物が定義され、その開発手順であるプロセスを定義した開発方法論があります。図 1 は、抽象度の異なる 4 つのモデルとそ

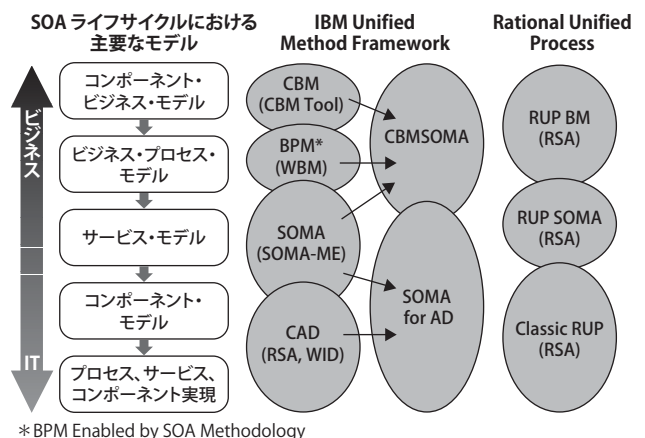


図1. モデル、開発方法論、ツールの概要

の実現を作成する際に活用する開発方法論をマッピングしたもので、カッコ内はその開発方法論で使用される主要なモデリング・ツールです。

SOA 適用プロジェクトでは、1つのモデルだけを作成することは少なく、いくつかのモデル間でトレーサビリティを確保しながらモデリングするため、個別の開発方法論を組み合わせ利用しやすいようにパッケージ化（テラリング）した方法論を用意しています。例えば、CBM、BPM Enabled by SOA Methodology、Service Oriented Modeling and Architecture（以下、SOMA）をテラリングし、企業全体のモデリングからサービスを定義するところまでをカバーする CBMSOMA、また SOMA とカスタム開発用の CAD（Custom Application Development）をテラリングした SOMA for AD（Application Development）が代表的なものです。さらに、パッケージの利用やアジャイル開発プロセスに合わせたサービス指向開発方法論も準備が進んでいます。

また、すでに Rational® Unified Process（以下、RUP）をご利用のお客様、または今後 RUP を開発プロセスとして採用されるお客様においても SOA をご活用していただけるよう、ビジネス・モデリングおよびサービス・モデリングのプロセスが RUP に追加され、それぞれ開発方法論（RUP BM および RUP SOMA）として統合されています。

このように IBM では、お客様の状況や SOA の取り組み方に応じて選択可能な開発方法論、ツール、アセットを準備するとともに、これらのスキルを習得し経験を積んだ技術者を多く育成しています。次のセクションからは、ビジネス、サービス、コンポーネントの各モデリング・フェーズにおけるメソッド、ツール、アセットなどを解説します。

## ② ビジネス・モデリング

ビジネスの迅速性を実現するためには、ビジネスを可視化し、構造、機能、関係を把握することが重要です。しかしながら、企業には複雑なビジネス構造があり、モデルとして表現することは容易ではありません。この複雑さを簡素化する手法として「コンポーネント化」を利用して企業全体のモデリングを行います。これが CBM です。CBM では、組織構造ではなく、ビジネス機能に着目してビジネスをコンポーネント化します。さらに、ビジネス・コンポーネント間の依存性を明らかにし、コンポーネントごとの評価やベンチマーキングを行い、どのコンポーネントがてこ入れすべき「ホット」なものであるかを評価します。ホットなビジネス・コンポーネントではビジネス・ケースが作成され、そのビジネス・ケースを基に企業変革をスタートさせます。

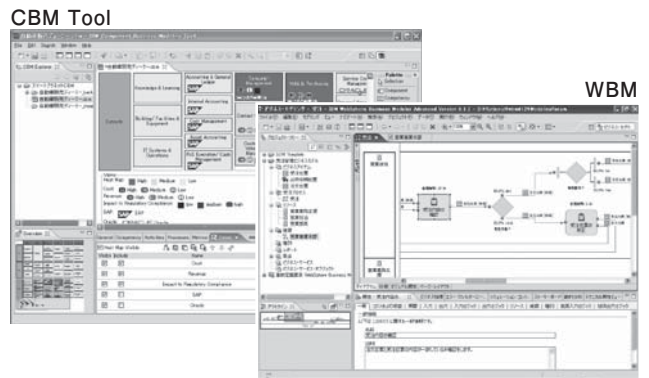


図2. CBM ToolとWBM

IBM ではこの CBM をモデリングするツールとして CBM Tool を提供しています。CBM Tool には、70 種類以上の業界別 CBM テンプレートが用意されており、フィット&ギャップによりモデルを定義できます。また、CBM の全体マップにはビジネス評価を記入し、ホットなコンポーネントをビジュアルに表示することができます（図 2）。

しかし、整然と並んだビジネス・コンポーネントを眺めていても、実際にビジネスがどのように動いていて、どこに改善ポイントがあるかを把握することはできません。そこで、ビジネスの動的側面をモデリングする手法として、ビジネス・プロセス・モデルがあります。また、それを記述するツールとして WebSphere® Business Modeler（WBM）を、さらにビジネス・プロセスを早く記述するためのテンプレートとして、APQC Process Classification Framework<sup>SM</sup>（以下、APQC PCF<sup>SM</sup>）に準拠した業種別プロセス・テンプレートを提供しています。このテンプレートを参照してビジネス・プロセスを記述することで、APQC PCF 準拠のベンチマーキングを活用し、お客様独自の KPI、そして迅速性を評価する KAI（Key Agility Indicator）を算出することができます。ベンチマーキングを、WBM の持つシミュレーション機能と併せて利用することで、企業変革の定量化評価を実施することができます。

ビジネス・モデリングの発展形には、ビジネス変革の結果を再度ビジネス・モデルにフィードバックするサイクルを確立する、すなわちビジネス・プロセスを継続的に管理する活動であるビジネス・プロセス・マネジメント（BPM）があります。そして IBM では、SOA ベースのビジネス・プロセス・マネジメントを実現する開発方法論として、CBMSOMA を提供しています。CBMSOMA を活用したプロセス管理サイクルを企業内に確立することで、継続的なビジネス変革が実現でき、ビジネス環境の変化を予測し、迅速に対応できる企業体質に改善していくことが可能になります。

### ③ サービス・モデリング

ビジネスの迅速性を高めるためには、さまざまなモデル要素から、企業において再利用性が高く、変化に対応可能なサービスを適切に探し出し、定義する必要があります。サービス・モデリングのための開発方法論である SOMA は、独特かつ実践的な開発方法論とアーキテクチャーを提供しています。

SOMA は5つのフェーズで構成されています (表 1)。

表 1. SOMA の5つのフェーズ

| フェーズ                     | 概要                                    |
|--------------------------|---------------------------------------|
| サービス識別<br>Identification | 再利用可能なサービスを判別してサービス一覧を作成します。          |
| サービス仕様<br>Specification  | 公開サービスの仕様 (メッセージやオペレーションなど) を定義します。   |
| サービス実現<br>Realization    | 公開サービスをどのようなプロセスやコンポーネントで実現するかを決定します。 |
| サービス実装<br>Implementation | 実行可能なサービスをプロセスやサービス・コンポーネントとして実装します。  |
| サービス配置<br>Deployment     | 実装されたサービスを SOA 稼働環境へ配置して利用できるようにします。  |

これらのフェーズでは、実践に基づいたサービス指向特有の設計開発手法が定義されています。例えば、サービスは単にビジネス・プロセスやユース・ケースから識別されるだけではなく、ビジネス・ゴールやビジネス・ルール、そして既存システムもサービスを識別するための分析対象となります (図 3)。その上で、企業にとって再利用性と変化対応を兼ね備えたサービスをサービス・モデルとして定義し、

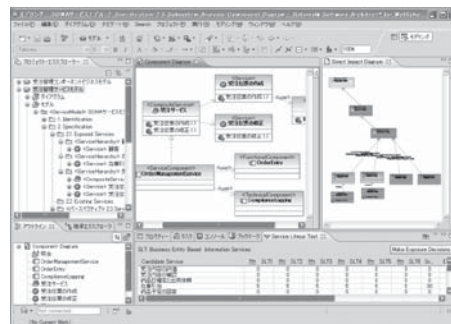


図4. SOMA-ME

その情報ソースである各種上位モデルやサービス・リポジトリとトレーサビリティを確立します。

サービス・モデリングを実施するモデリング・ツールとして SOMA-ME (Modeling Environment) (図 4) を提供しています。SOMA-ME は UML2.1 ベースのサービス・モデリング環境であり、Rational Software Architect (以下、RSA) のプラグインとして提供されています。SOMA-ME は各種モデルやリポジトリから情報を取り込んでサービス化するだけでなく、それらの基となる情報との間にトレーサビリティを確立し、ビジネスの各要素とサービスの関係を維持します (図 3)。このトレーサビリティにより、ビジネスの変化に対して IT 化されたサービスをどのように変更すればよいか、またどの IT サービスがどのようにビジネスに寄与するかを可視化することができます。

SOMA-ME により作成されたサービス・モデルは、Word や PDF などのドキュメントに出力可能であり、モデルとドキュメントとの二重メンテナンスを行う必要がなく、モデルを設計の中心に据えたモデル駆動開発が可能です。後続のサー

ビスを実現するためには、サービス仕様 (以下、WSDL) やサービス・メッセージ (XSD)、SCA モジュールなどを出力し、連携します。これらの情報は WebSphere Integration Developer (WID) に取り込まれ、BPEL、SCA、SDO などの SOA 技術を利用してサービスが実装されます。

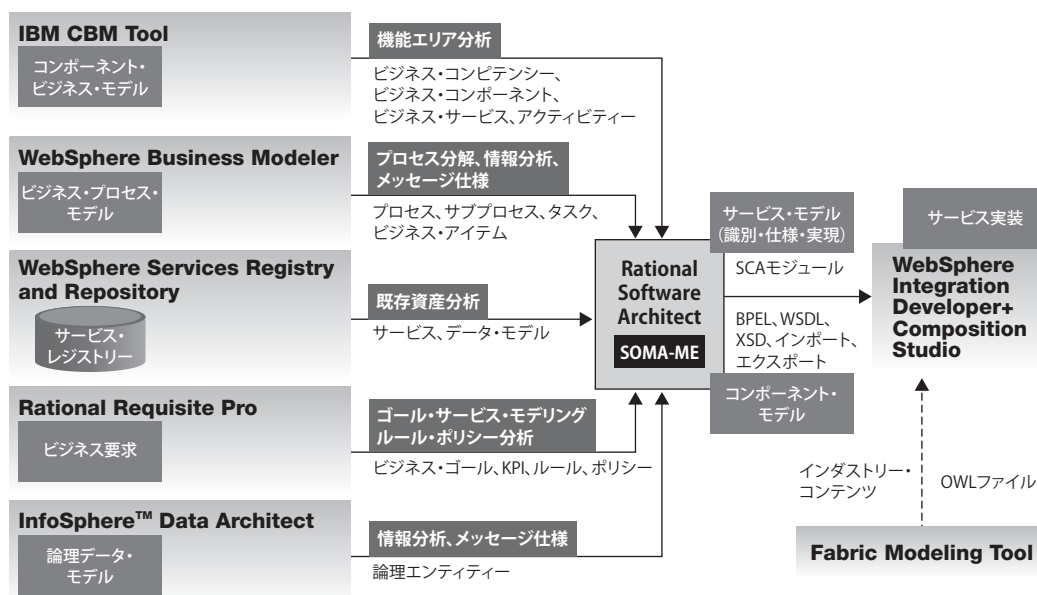


図3. サービス・モデルに関連するモデルとツール

#### ④ コンポーネント・モデリング

SOMA の上流工程において識別されたサービス・コンポーネント、およびそれらに含まれる機能コンポーネントや技術コンポーネントをカスタム開発する場合には、従来からあるコンポーネント設計 (Component-Based Design: CBD) やオブジェクト指向分析設計 (Object-Oriented Analysis and Design: 以下、OOAD) といった手法を用いてコンポーネント・モデリングを行います。コンポーネント・モデリングは、RSA を利用して SOMA-ME のモデルからシームレスに行うことができます。

コンポーネント・モデリングは、従来の OOAD と同様にユース・ケース駆動で行います。ここでのユース・ケースとは、サービス・モデリングと並行して、ビジネス・プロセス、またはビジネス・ユース・ケースからシステム化対象として抽出されてきたシステム・ユース・ケースです。このユース・ケースを分析しながら、サービス・コンポーネント、機能コンポーネント、技術コンポーネント間の振る舞いを示すシーケンス図を作成し、これらのコンポーネントとすでに識別済みのサービス・オペレーションでユース・ケースを実現できることを確認します。さらに、ユース・ケースに現れるデータをサービス・モデル、論理データ・モデルと合わせて検討し、オペレーションの入出力パラメーターも含めて各コンポーネントのインターフェース仕様を確定します。

仕様が確定したコンポーネントは、それぞれ独立して内部の詳細な設計を行うことができます。ここで、アーキテクチャや実装技術、コンポーネントのタイプに合わせてデザイン・パターンを適用し、OOAD の手法によりコンポーネント内部のクラスの設計を行います。

このようなコンポーネント・モデリングの手法を取ることで、最初から細かい粒度のクラスを大量にモデリングして全体が複雑化し、把握できなくなるといった、従来の OOAD でよく見られた問題を避けることができ、モデル全体のふかかさと詳細性を両立することができます。また、必然的に各コンポーネントの独立性も高まり、ビジネス・プロセスからサービス、ユース・ケース、コンポーネントへとトレーサビリティも確保されるため、ビジネスの変化に対して迅速かつ柔軟にコンポーネントの組み換えを行うことが可能になります。

コンポーネント・モデルは、詳細化が進むにつれて対象要素が格段に増加し、モデリング作業の負荷も高くなるため、作業の効率化が鍵となります。そこで、IBM では、モデルの変換やパターンの適用、クラス図の生成、モデルの品質検証、コード生成などを支援する RSA のプラグインや、モデリングのガイドをソフトウェア資産 (アセット) として提供することにより、モデリング作業の効率化を図っています。

#### ⑤ ビジネス・プロセスの実現

前述のコンポーネント・モデリングで、必要なビジネス機能を提供するサービスがモデリングされます。一方、目的とするビジネス・プロセスを実現させるためには、それらのサービスをどのようなルールの下でどのように呼び出すかというフロー (図 5) の実現も必要です。

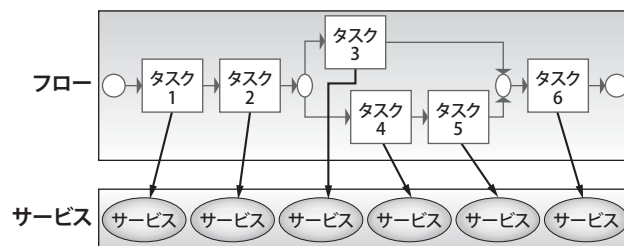


図5. ビジネス・プロセスにおけるサービスとフロー

SOA においては、ビジネス機能を提供するサービスと、それらのサービスを呼び出すフローを、別々にモデリングして実現するという手法を採用することにより、開發生産性を高めるとともに、高品質かつ変更に対して柔軟なビジネス・プロセスが実装されます。

フローのモデリングにおいては下記のような事項を考慮することが重要となります。

- サービスを呼び出すタスクの順番
- WSDL などのサービスを呼び出すための方法
- 次のタスクに遷移するためのルールやイベント
- フローの分岐やループの条件
- 画面で確認などを行うヒューマン・タスク
- フロー内で取得すべき KPI の種類と設定箇所

また、実際の業務を遂行する上では、ERP パッケージ、レガシー・システム、J2EE™ アプリケーションなどの多くの既存アプリケーションが利用されます。SOA を現実のものとするためには、これら既存アプリケーションをサービス化して再利用することも重要となってきます。WID (図 6) は、アダプターを介しての既存アプリケーションのサービス化の実現や、サービスとフローの連携を構成する開発ツールです。

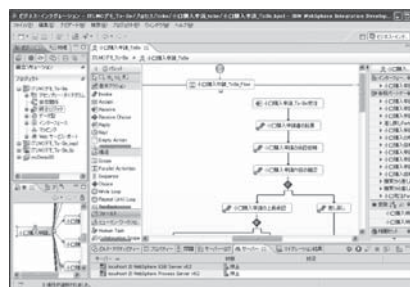


図6. WebSphere Integration Developer (WID)

WID は、ビジネス・プロセスを実現するためのサービス呼び出しのフローを BPEL の形で具体化する役割と、ESB におけるメディエーションを作成する役割を担います。BPEL としてフローを具体化するには、WBM で設定した KPI も取り込まれ、稼働中にビジネス・メトリックスとして収集することができるようになります。また、アダプターを介した既存アプリケーションのサービス化の設定も行います。

このようにして作成された BPEL フローと、コンポーネント・モデリングを経て実現されたサービスを合体してコンポジット・アプリケーションを構成し、稼働環境に配置されることになります。

稼働中に収集されたビジネス・メトリックスはビジネス・モデルにフィードバックされ、ビジネス・プロセスを改善するための基礎データとなります。このサイクルを繰り返すことにより、ビジネス変化に柔軟かつ効果的に対応可能なシステムを実現していきます。

## 6 変化対応のためのビジネス駆動開発

ビジネス環境の変化が激しい今日、多くのパラメーターが存在する外部環境の変化を察知することは極めて難しくなっています。それに比べて、自らのビジネス状況を戦略的な観点から評価し、変化を察知することは比較的容易に実施可能ではないでしょうか。ビジネスと IT をモデリングし、それを通して外部環境の変化に気付くようになるものと期待されます。そのようなビジネス基盤を築くためにも SOA を活用していただくことを期待しています。

しかしながら、広範囲にわたって詳細にモデリングしてサービス化していくことは、それ自体に膨大な時間を要し、本末転倒になりかねません。そこで、ビジネス迅速性を実現するための基盤を効果的に確立するには、最上位の CBM から「ホット」なポイントを識別し、その限られた範囲において SOA 技術の特長を生かした IT アーキテクチャー・トランスフォーメーションを行います。さらに、SOA ライフサイクルの 1 サイクルの期間を短くし、サイクル・スピードを速めることで柔軟性もスピードも向上します。

サイクル・スピードの速い SOA ライフサイクルを実現するためには、その企業に最適な実践的な開発方法論、ツールを活用することがポイントです。IBM では、内外の変化を察知し、迅速に変化し続けるビジネス& IT 基盤構築のお手伝いをさせていただいており、実績のある開発方法論とツールをご提供しています。さらに、それらを効率よく活用していただくためのノウハウをアセットとしてご提供しています。



日本アイ・ビー・エム株式会社  
GBS エンタープライズ・アーキテクチャー&  
テクノロジー  
シニア IT アーキテクト

藤田 一郎 Ichiro Fujita

### 【プロフィール】

1993 年、日本 IBM 入社。製造業のお客様を担当するサービスデリバリー部門において、システム開発プロジェクトに多数参画。1997 年頃よりプロジェクト活動の傍ら、講演・Web/雑誌記事・書籍・コミュニティ活動などを通じて Java™/J2EE 展開を推進。2002 年頃よりモデル駆動開発 (MDD)・モデル駆動アーキテクチャー (MDA)・サービス指向アーキテクチャー (SOA) を活用した業務システム開発を推進している。



日本アイ・ビー・エム株式会社  
GBS BV 推進 ソリューション&アセット  
シニア IT アーキテクト

五十嵐 正裕 Masahiro Ikarashi

### 【プロフィール】

1987 年、日本 IBM 入社。開発部門にて大型から PC までのシステム評価、特許調査に従事。1995 年より営業/サービス部門に異動し、C/S 型、Notes®、WAS などのアプリケーション・フレームワークの開発とお客様への展開を担当。現在は、お客様プロジェクトでアーキテクトとして活動するとともに、開発メソッド、モデリング関連のガイド、ツールなどのアセットの開発とプロジェクトへの普及を推進している。



日本アイ・ビー・エム株式会社  
ソフトウェア事業 SOA ビジネス開発  
IT アーキテクト/ソフトウェア SOA エバンジェリスト

上沢 健 Ken Uesawa

### 【プロフィール】

1989 年に日本 IBM に入社。ホスト担当 SE を皮切りに Windows® 系/Java 系の開発プロジェクトを経て、大小さまざまなシステム構築プロジェクトにおいて IT アーキテクトとして活動。2004 年より SOA に携わり、現在は IT アーキテクトとしての職務とともに、ソフトウェア事業における SOA エバンジェリストとして社内外における SOA 関連の研修やセミナーの講師も務める。