

# ビジネス・イベント処理のためのモデリング手法

石井 旬 蓮見 竜太 榎本 聡

## Event Modeling Method for Business Event Processing

Jun Ishii, Ryuta Hasumi and Satoshi Enomoto

イベント処理の企業システムへの適用において、イベントを見いだす作業は、想定が容易なイベントは導出できても、ビジネス全体の価値向上を目的として網羅的に、かつ効果的なイベントを導出することは、これまで体系的な手法がなく困難であった。本論文では、IBM が提唱するビジネス・イベント処理 (BEP) に焦点を当て、BEP ソリューション構築においてビジネス・イベントを導出し定義する手法を提案する。この手法はビジネス・イベントと密接な関係があるビジネス・プロセスとビジネス・エンティティの状態遷移に着目してビジネス・イベントを導出し定義するところに特徴がある。この手法によりイベント処理を活用しようとする企業が、単なるイベント処理ミドルウェアの導入と局所的なイベント活用に陥ることなく、網羅的かつ効果的なイベント識別を基礎にしたビジネスの価値向上を実現する BEP ソリューションを構築することが可能になる。

Leveraging Event Processing to Enterprise systems which extracts comprehensive business events to increase whole business value was hard to execute, and tended to extract events which are imaged easily. This paper proposes Business Event Modeling Methodology based on Business Event Processing (BEP) which is a proposed by IBM. This methodology has unique features that extract business events with business process and state transition which are close to business events. This new methodology enables enterprises who intend to leverage event processing to develop BEP Solutions which increase business value based on comprehensive business events extraction.

**Key Words & Phrases** : イベント処理, イベント駆動型アーキテクチャー (EDA), 複合イベント処理 (CEP), ビジネス・イベント処理 (BEP), モデリング手法  
Event, Event Processing, Event Driven Architecture (EDA), Complex Event Processing (CEP), Business Event Processing (BEP), Modeling

### 1. はじめに

Event Driven Architecture (以下, EDA) やその方式の1つである Complex Event Processing (以下, CEP) などのイベント処理がビジネスに新たな価値をもたらす IT アーキテクチャー・スタイルとして注目されている。あらゆる企業でイベント量が急速に増加中で、この急増するイベントを活用してビジネスに価値をもたらすことができると考えられ始めているからである。大規模企業では毎秒  $10^4 \sim 10^7$  回のビジネス・イベントが発生しているといわれている [1]。

ガートナーによれば、CEP は、①意思決定の質の向上、②対応の迅速化、③情報過負荷の削減、④コストの削減の4通りによってビジネスに価値をもたらすとされており、客観的な視座からもその価値が認識されている [2]。

IBM のコーポレート・ビジョンである Smarter Planet を実現するための4つの課題の1つである Smart Work は、変化を続けるビジネス・モデルやビジネス環境に合わせたプロセスやオペレーションの変更を、より迅速にかつ柔軟に行うことを目標としている。

この Smart Work の実現のソリューションには SOA や Business Process Management (以下, BPM) などが挙げられている。企業における変化を迅速にとらえ、その変化に柔軟に対応していくという意味で、イベント処理を BPM などと組み合わせて利用することで、より価値の高い Smart Work を実現することが可能になると考えられる。

イベント処理の分野では、CEP や Event Stream Processing (以下, ESP) [3] などのイベント処理の方式や基盤構造などが検討されている。The Power of Events [4] は CEP に関する著名な書籍であるが、早くから Event Processing Agent などのイベント処理の基

提出日:2009年5月11日 再提出日:2009年12月10日

盤構造について論じており、今日論じられているイベント処理の基盤構造はこれを基礎にしているといつてよいであろう。ベンダーによるイベント処理関連製品も主要なITベンダーは提供をすでに始めており、CEP専門ベンダーなども含めると製品の選択肢はかなりの数となる状況になってきている。IBMもビジネス・イベント処理（Business Event Processing：以下、BEP）を改めて定義し、それを実現する製品であるWebSphere Business Eventsをすでに提供している [5]。

一方、企業におけるイベント処理の活用は発展途上段階で、個別の事例は幾つもあるものの、統計的に傾向を見極めベスト・プラクティスやメソッドロジーとして集大成する段階にあるとは言い難い。ガートナーは先のレポートでも「イベント処理の導入に当たって最大の障壁は使うべき場所が分からないことである」とし、イベント指向分析・設計の必要性について言及している [1]。本論文では、IBMが提唱するBEPに焦点を当て、BEPソリューション構築においてビジネス・イベント（以下、イベント）を導出し、定義する手法を提案する。

## 2. イベント処理におけるモデリングの必要性

イベント処理の分野では先に述べたようにCEPやESPという方式や基盤構造の在り方が研究・検討されており、企業システムにおける活用事例がすでに幾つも出現している。一方、これから活用しようとする企業にとって最初に行う作業はどの業務領域がイベント処理を利用すべきかという課題領域を明確にすることである。そして、次の段階で行うべき作業は「何をイベントとして見いだすのか、そのイベント発生後にどのようなビジネス・アクティビティーを実行すべきか、そのイベントをどうITでどう捕らえるのか」をビジネス価値向上の視点から網羅的に洗い出し、それに関連するイベントの種類を定義することである。言い換えればイベントに関する体系的なモデリング手法とそれに基づく作業が必要だといえる。CEPやEDAのコミュニティにはすでに著名なものが幾つか存在するが、このビジネス・イベントのモデリングに関する研究や文献は現時点ではほとんど見られない [6] [7]。活用事例を見ても、単なるイベント処理ミドルウェアの導入事例と考えられるものや、課題としてこれまで捕捉できなかったがすでに想定されていたイベントを捕捉した局所的なイベント処理活用事例と判断されるものがかなり存在する [8] [9]。

イベント処理を活用し、継続的にビジネスの価値を向上していくためには、網羅的かつ効果的なイベントの識

別とその段階的実装が不可欠である。このためには活用ビジネス領域のビジネス・プロセスやビジネス・エンティティーの状態遷移からイベントを網羅的に導出し定義する必要がある、これを考慮した体系的で再現可能な手法が必要である。

## 3. モデリングの定義とBEPの本質的なモデル

ビジネス・イベント処理の適用に際して、必要となるモデリングは以下の作業を行うことと定義する。

- (1) ビジネス・イベント処理を適用するビジネス領域について、ビジネス価値を向上するために網羅的かつ効果的なイベントを導出する。
- (2) そのイベントが発生した際に、実行すべきビジネス・アクティビティーを明確にする。
- (3) そのイベントをIT的に捕捉が可能か、可能であればどのように捕捉するのかを定義する。

このBEPのモデリング手法を検討するに先立って、BEPの本質的なモデルとは何なのかをもう一度、押さえておくことにしたい。

図1はBEPのビジネス視点とIT視点のモデルの対比を示している。まず、ビジネスの視点で、何らかの（ビジネスの）イベントが発生したら、そのイベントを受けて適切なタイミングで、ビジネス・アクティビティーを実行する、ということがシンプルだが本質的なモデルであると考えられる。

一方IT視点では、イベント・プロデューサーとコンシューマー、その間にあるイベント処理エージェントというものが一般的には議論されている。ただしこれはあくまでもITの視点である。BEPのモデリングにおいては、まずビジネスの視点でのイベントやビジネス・アクティビティーの識別を行うこととする。

実際にIBMが定義しているビジネス・イベント処理の定義を参照すると以下の通りである。

「何らかの意思決定が必要なビジネス状況が生じたこ

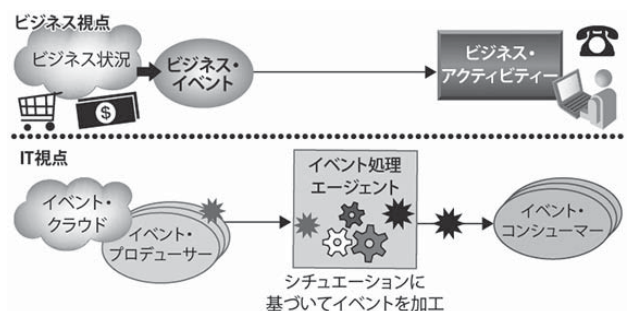


図1. BEPのビジネス視点とIT視点のモデル

とを示す電子的なメッセージの発生、あるいはメッセージが欠けていること（イベント）を検知して、適切なアクションを適切なタイミングで実施することにより処理を行うこと」

これは前述の内容と同様であり、より詳細には「あるいはメッセージが欠けていること（イベント）を検知」ということもあらためてイベントとして定義されている。このイベントの欠損も1つのイベントとして捕捉する必要があることに注意を払う必要がある。

### 3.1 ビジネス・イベントを識別するための根底にある概念

#### 3.1.1 ビジネス・コンテキスト

BEPにおいて最も重要な概念は（ビジネス）イベントであることは自明である。イベントはビジネス・プロセスの中に発生する。このビジネス・プロセスは、例えば顧客との契約、サプライチェーン、製造工程などが考えられるが、このようなビジネス・プロセスの中にはさまざまなイベントが存在する。

BEPのソリューションを構築するに当たり、まず考慮する必要があるのがビジネス・コンテキストである。図2はビジネス・コンテキストとイベントの関係を示したものである。

例えば契約というビジネス・プロセスであれば、それは売り込み局面なのか、契約手続きなのか、解約局面なのか、サプライチェーンであれば、それは製造局面なのか、物流局面なのかなどのビジネス・コンテキストが考えられる。

このビジネス・コンテキストによって、捕捉すべきイベントは変わってくることになる。

#### 3.1.2 コア・エンティティー

もう1つの重要な根底にある概念がコア・エンティティーである。BEPでイベント・モデリングを行う対象となるビジネスには、そのビジネスで取り扱う中心的な対象＝コア・エンティティーが存在する。

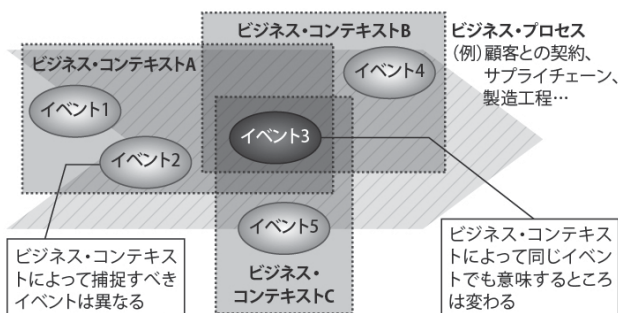


図2. ビジネス・コンテキストとイベントの関係

例えば「契約」というコア・エンティティーがあるとした場合、さまざまな契約にかかわるビジネス・プロセスが存在する。また、このビジネス・プロセスやビジネス・オペレーションの背景には「コア・エンティティーの状態遷移」が存在する。例えば、契約受付、受付完了、審査完了、契約完了という遷移が具体例となる。

BEPのモデリングにおいては、まずBEP適用のビジネス領域に関するビジネス・コンテキストとコア・エンティティーの状態遷移を理解する必要がある。

### 4. ビジネス・イベントを定義するためのモデリング手順

表1は本論文で提案するBEPソリューションのためのモデリング手順を示している。

表1. BEPソリューションのためのモデリング手順

#	手順	内容	アプローチやツール
①	コア・エンティティーの識別と状態遷移の作成	当該ビジネスのコア・エンティティーを定め、そのビジネス・スコープ全体のライフサイクルの視点で状態遷移をハイレベルに記述する	・ERD ・ステートマシン図
②	ビジネス・コンテキストに応じた状態遷移領域の切り取り	全体のライフサイクルの中で、検討の焦点をビジネス・コンテキストとして領域を定義	・ビジネス・コンテキスト・ダイアグラム
③	状態遷移の詳細化	当該コンテキストに着目して、コア・エンティティーの状態遷移をブレイクダウンする	・ステートマシン図
④	ビジネス・アクティビティーの調査	状態の周辺のビジネス・アクティビティーの有無や内容を調査する 状態遷移とアクティビティーのクロスレファレンス	・ステートマシン図 ・アクティビティー図、BPMN、プロセス・モデリング・ツール
⑤	ビジネス・イベント候補の抽出	Desired State遷移するためのイベント候補を網羅的に導出	・状態遷移とアクティビティーのクロスレファレンス
⑥	ビジネス・イベントの定義	ビジネス・イベント候補をとらえるための手段を定義	・現行システム調査

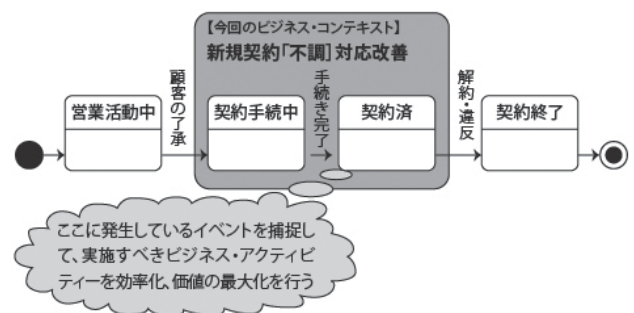


図3. ビジネス・コンテキストに応じた状態遷移領域の切り取り



以降に、この各ステップについてその内容と具体例を解説する。

#### 4.1 手順①：コア・エンティティー識別と状態遷移の作成

このステップでは、当該ビジネスのコア・エンティティーを定め、そのビジネス・スコープ全体のライフサイクルの視点で状態遷移をハイレベルに記述する。

図3の4つの状態からなる状態遷移図は、具体的な現実の案件を抽象化して書いたものであるが、コア・エンティティーは契約である。例題では状態マシン図で書いているが、現実の案件ではハイレベルであっても主要な状態を書くだけでも長く複雑な状態マシン図になる。

ここでの注意事項はまだイベントを洗い出す領域を定めていないので、あくまでもハイレベルで書くことが重要である。ビジネス全体の流れをコア・エンティティーの状態遷移で書き表し、これからその全体の流れのどの領域をBEPでビジネス・パフォーマンスを向上するのかを検討する地図になる。

#### 4.2 手順②：ビジネス・コンテキストに応じた状態遷移領域の切り取り

次にその地図に対して、ビジネス・コンテキストに応じてどの部分をBEPで取り組むのか対象を絞っていく作

業がこのステップである。前述の当該ビジネス全体ライフサイクルの状態遷移の中から、今回与えられているビジネス・コンテキストに応じて領域を限定していく。

このステップの目的はモデリングする領域を作業効率の観点から狭めているという観点もあるが、前述の通りビジネス・コンテキストによってとらえるべきイベントを明確化し、イベントの持つ意味を明確に定義するためでもある。

図3のハイライト部分は、ビジネス・コンテキストに応じて状態遷移領域を切り取ることを例題とともに示している。例題で見ると、設定するビジネス・コンテキストは、「新規契約における不調対応の改善」となる。これはユーザー部門から与えられた課題、もしくは全社もしくは部門のイニシアチブで、BEPで解決できると想定している案件、ということになる。当該業務アプリケーションのIT的な課題の場合もあると想定される。

このビジネス・コンテキストで限定した状態遷移の領域について、イベントを捕捉して適切なビジネス・アクティビティーを適切なタイミングで行うことで、ビジネスを効率化し、ビジネスの価値を最大化するということになる。

ビジネス・コンテキストを捕捉する方法については、モデリングの手法においてはコンテキスト・ダイアグラムも有効である。コンテキスト・ダイアグラム(図4)を描くことで、そのビジネスにかかわる登場人物や関連する組織やシステムにはどんなものがあるか、またそのビジネスやシ

ステムが持っているべき機能の範囲というものも、明確にできるメリットがある。これをBEPのソリューション構築でも領域を限定する段階で描き、ビジネス・コンテキストを明確にし、その領域についてイベントをモデリングすることにより、BEPソリューションの品質が向上し、作業の効率化が可能となる。

前述のコンテキスト・ダイアグラムを利用して、ビジネス・コンテキストに応じた状態遷移領域を切り取る具体的な手法について以下に述べる。

BEPにより課題を解決できる、もしくはビジネス価値を向上するため

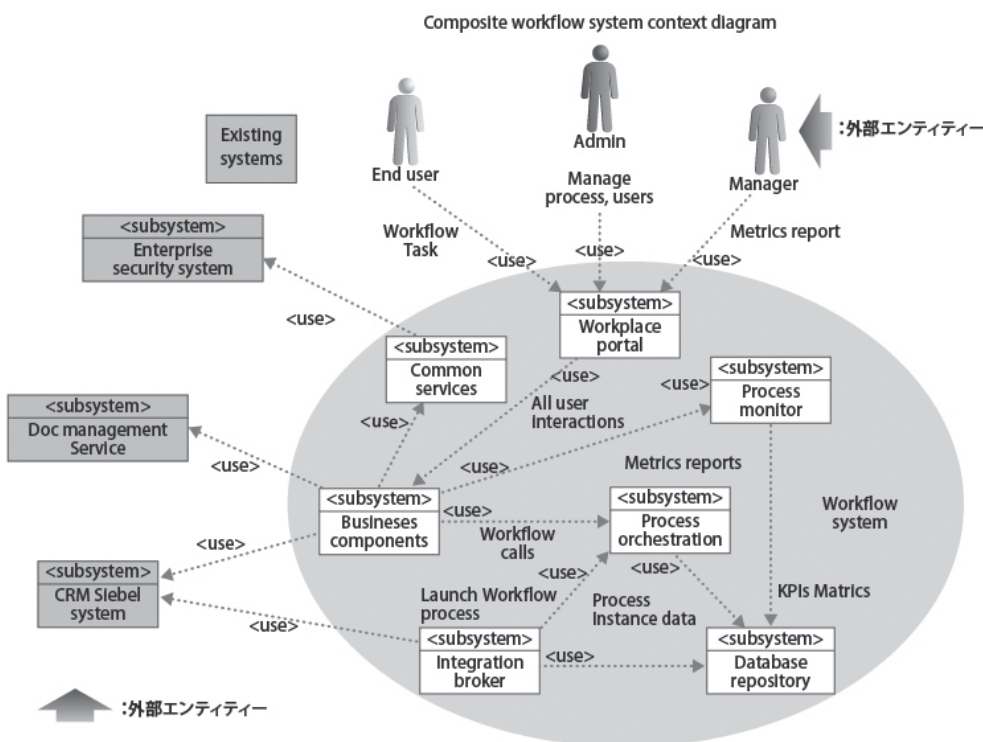


図4. コンテキスト・ダイアグラムの例示

の IT ソリューションのアウトラインを、コンテキスト・ダイアグラムを記述することで先にイメージし、ビジネス全体から BEP により解決する課題領域を切り取る考え方であり、具体的には以下のステップとなる。

- 手順①によりコア・エンティティ識別とハイレベルな状態遷移図を作成する。
- 今回想定する BEP を活用した IT ソリューションについて、外部エンティティを洗い出すべて明らかにする。外部エンティティとは、コンテキスト・ダイアグラムの円の中のシステムがインターフェースをとるべき外部要素である (図 4)。具体的には顧客、ユーザー、またはビジネス・エキスパート、関連他システムなどが想定される (この情報は業務フローやビジネス・プロセス・モデルから入手する)。
- 想定する BEP を活用するシステムの名称を記述したオブジェクト (箱または円) を描き、最初のコンテキスト・ダイアグラムを作成する。このシステム名称を記述したオブジェクトの周辺に、外部エンティティの名前を書いた箱を記述する。
- 外部エンティティとシステムの間にもどのような関係があるかを、線や線上の補記で記述する。
- システムの中にも含めるべき機能やデータと外部エンティティについて、また内部にも含める範囲について、最初のコンテキスト・ダイアグラムを元に推敲する。(ここまでがコンテキスト・ダイアグラムの作成)
- 作成したコンテキスト・ダイアグラムにおける外部エンティティは、BEP での対象領域となるビジネス・コンテキストに応じた状態遷移領域の外となる。手順①で作成したハイレベルな状態遷移図と当手順で作成したコンテキスト・ダイアグラムを相互参照し、状態遷移図における状態のうち、コンテキスト・ダイアグラムのシステムの中に該当する状態と、外部エンティティに該当する状態をおのおの識別していく。
- 識別した外部エンティティに該当しない状態群が、今回の課題領域となる。

### 4.3 手順③：状態遷移の詳細化

このステップでは、イベントを導出する前段として、ビジネス・コンテキストの領域のコア・エンティティの状態遷移をブレイクダウンすることを行う。まず前述の通り、ビジネス・コンテキストやその領域が目標としている課題解決やビジネス価値向上から見て、状態遷移で定義された個々の状態の粒度や遷移は十分かを検証する。図 5 では、ビジネス・コンテキストやその領域における改善・

ビジネス価値向上の観点から新たな状態を識別していることを示している。

例題では、ビジネス・コンテキストが新規契約における「契約不調」対応改善なので、まず対応すべき「契約不調」状態が正しく定義されているかという視点で検証すると、状態の定義が不十分であることが確認できる。その「契約不調」の具体的な内容やパターンは別途、正確に調査し捕捉する必要があるが、まず企業として「契約不調」という正式な状態を定義する必要がある。この不足していた状態を“Desired State”と呼ぶことにする。

### 4.4 手順④：ビジネス・アクティビティの調査

このステップでは、この Desired State 周辺のイベントを洗い出すべく、まず Desired State 周辺のビジネス・アクティビティを調べていくことになるが、ここで状態マシン図の規約を利用して、アクティビティを整理する。UML2.0 の状態マシン図では、ある状態に遷移した時点で実行する振る舞いを「入状時振る舞い」(entry)、ある状態にある間継続して実行し続ける振る舞いを「実行時振る舞い」(doActivity)、状態から離れる直前に実行する振る舞いを「退状時振る舞い」(exit) として状態記述内に定義できる [10]。

図 6 は状態遷移や状態マシン図の規約を利用したアクティビティ調査を示している。

主に以下のアクティビティ (図 6 の※部分) の有無を調査する。

- 遷移元状態の exit (退状時) に実行するアクティビティに該当するものはないか
- Desired State の Entry (入状時) に実行するアクティビティに該当するものはないか
- Desired State の Do (実行時) に実行するアクティビティ

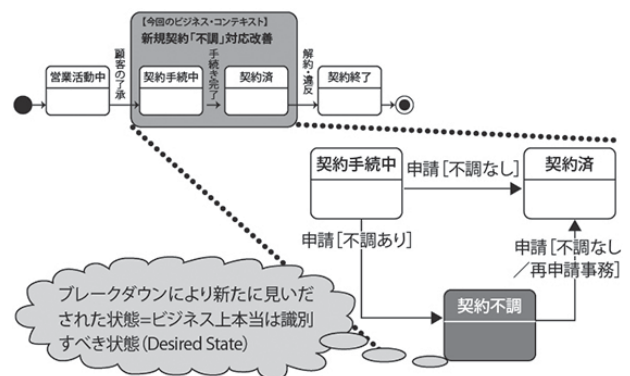


図5. 状態遷移の詳細化

ビティに該当するものはないか

- Desired State の Exit (退状時) に実行するアクティビティに該当するものはないか

また、状態マシン図では遷移上で実行するアクティビティも定義できるので、以下も調査する。

- Desired State から遷移先状態への遷移上のアクティビティに該当するものがないか

さらに、状態とアクティビティをより精緻に識別するためには、状態遷移とアクティビティをクロスレファレンスする形で検証を行う。状態遷移は状態マシン図で、アクティビティはアクティビティ図やプロセス・モデルを利用し、クロスレファレンスすることで、現実のビジネスで実行されている、もしくはされるべき状態遷移やアクティビティを識別できる。図7は例題におけるクロスレファレンスで精査される主なポイントを示している。

このクロスレファレンスについては AsIs と ToBe の両方について分析することで、AsIs で Desired State の周辺で何ができていないか、ToBe では何をすべきかが判明してくることになる。

#### 4.5 手順⑤: ビジネス・イベント候補の抽出

前のステップまでで、ビジネスとしてどうしたいかは、Desired State ですでに判明している。Desired State に遷移するためのイベント候補を網羅的に導出するために、以下の視点からイベント候補を導出する。

- 遷移元状態の exit のアクティビティおよびそのアウトプットやトリガー
- 遷移先状態の entry アクティビティおよびそのインプットやトリガー
- 遷移上のアクティビティおよびそのイン・アウトやトリガー

図8は上記の視点からイベント候補を洗い出す作業とその例題における結果を示している。また、前述の状態遷移とアクティビティのクロスレファレンスの中からもイベント候補の抽出が可能である。イベント候補に成り得るかは、Desired State に入状する条件を定義し、その条件達成をイベントにより効率化などの改善が期待できるかであり、具体的には以下のようなケースが想定される。

- Desired State への遷移のアクティビティが人手による処理であり、イベント捕捉により遷移の自動化が見込める。

- Desired State への遷移のアクティビティが、バッチ処理であり、本来リアルタイムに処理することが望ましいにもかかわらずリアルタイムにイベントが識別できない。

クロスレファレンスから状態を遷移するためのアクティビティが、人手による作業で

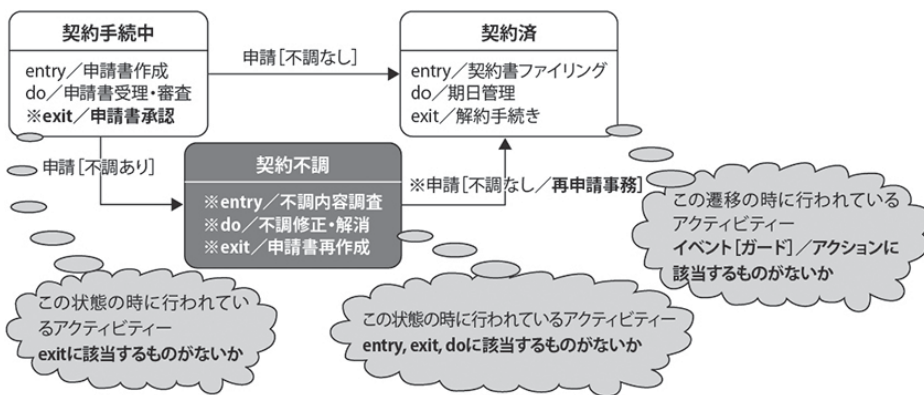


図6. ビジネス・アクティビティの調査

- AsIsとして存在するアクティビティ
- ToBeとして、今はないが、存在すべきアクティビティ

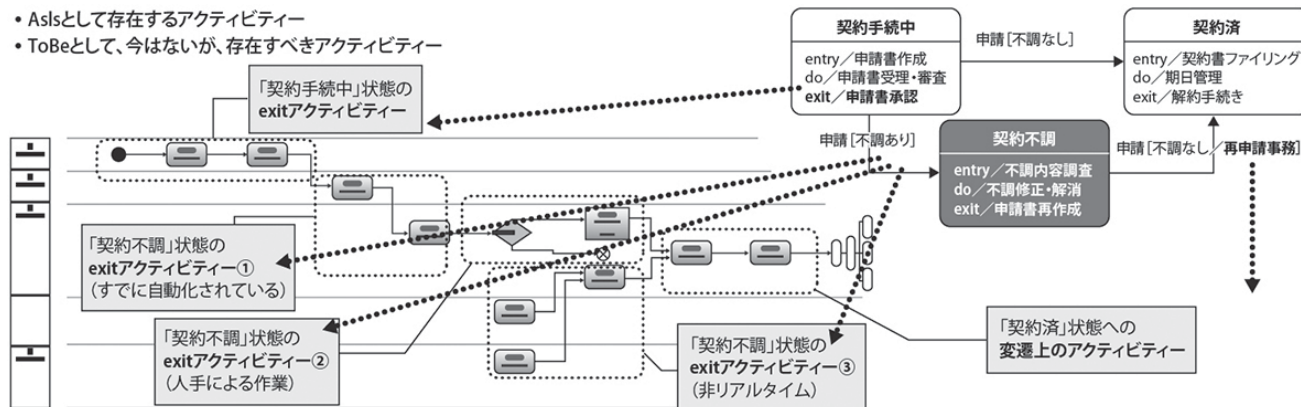


図7. 状態遷移とアクティビティのクロスレファレンス



あったり、非リアルタイムであったり、アクションがない、イベントが存在しないことによってとらえられていない状態だったりするときに、イベントによる改善が期待できる。

#### 4.6 手順⑥：ビジネス・イベントの定義

抽出したイベント候補を一覧にして、イベント・ポートフォリオ表を作る。

候補の意味は、導出されたイベントで重要と思われるものでも、取得が困難と考えられるものが存在することを示唆している。しかしながら取得が困難でも、この導出の段階では候補として削除せずに残しておく。この考え方はSOAのサービス・モデリング手法SOMA (Service Oriented Modeling and Architecture) のサービス・ポートフォリオ [11] の考え方を参考にした。候補として残しておくことで、その時点で取得が困難と思えるイベントも、ビジネス・イベントよりもさらに粒度の細かいイベントをCEPの活用により捕捉できる可能性もあるし、段階的に取得を実装することも可能であるというメリットがある。

イベント・ポートフォリオ表にはこれまで導出したすべてのイベント候補とその属性について以下の項目を記述する。

- ビジネス・イベント候補名
- ディスクリプション
- 当該イベントの発生元状態 (From State)
- 当該イベント発生時の遷移先状態 (To State)
- 当該イベント発生時の実行するビジネス・アクティビティー
- 当該イベントを捕捉する方法

このイベント候補の中からBEPソリューション要件や優先順位、ITでのイベント捕捉や後続ビジネス・アクティビティーの実行実現性などから、実装するイベントを決定していく。抽出されたイベントが多数になった場合は、実装するイベントを決定するプロセスに非属人的な手法が必要と思われる。当論文ではそこまでの十分な手

法の提示には至らなかったが、今後の継続検討課題としたい。

#### 4.7 本モデリング手法の効果

本モデリング手法の利用により、例題に挙げた「新規契約における不調対応改善」領域におけるイベント候補として、Desired Stateである「契約不調」状態に至るイベント群や、「契約不調」状態から「契約済み」に遷移するトリガーとなるイベントを網羅的に導出することができた。イベント・ポートフォリオ表では、各イベントが発生した後に実行するビジネス・アクティビティーや各イベントの捕捉方法が定義され、これらの情報はイベント処理ミドルウェアの定義の入力情報につながる。従来の、課題としてすでに認識されていたが捕捉されていなかったイベントのみを対応する局所的なイベント処理活用から、当該ビジネス領域をより広範囲でビジネス価値を向上するためのイベント群を識別できた。また、イベント処理ミドルウェアの定義情報を導出したことにより、ミドルウェアによるBEPソリューション構築の手順がよりスムーズになったといえる。より広義の効果として、イベント処理をどこで使うべきかが分からない企業や組織においては、この手法によりどの分野がイベントによるビジネス価値向上が可能であるかを想定することができる。また、昨今の企業における急激に増加するイベントのうち、何がビジネス価値向上に貢献するかを知ることが可能になる。

#### 5. おわりに

本論文では、BEPにおけるイベント・モデリング手法を提案した。

この手法を適用した結果、BEPソリューション構築におけるイベント導出の作業において、従来の属人的なモデリングや想定しやすいビジネス・イベントのみ導出のみにとどまらず、体系的で再現可能な手法にのっとりてビ

ジネス・イベントを定義できるようになり、BEPソリューションにおける設計作業の品質向上が図れると考えられる。また、イベント処理をビジネスの価値向上に活用しよ

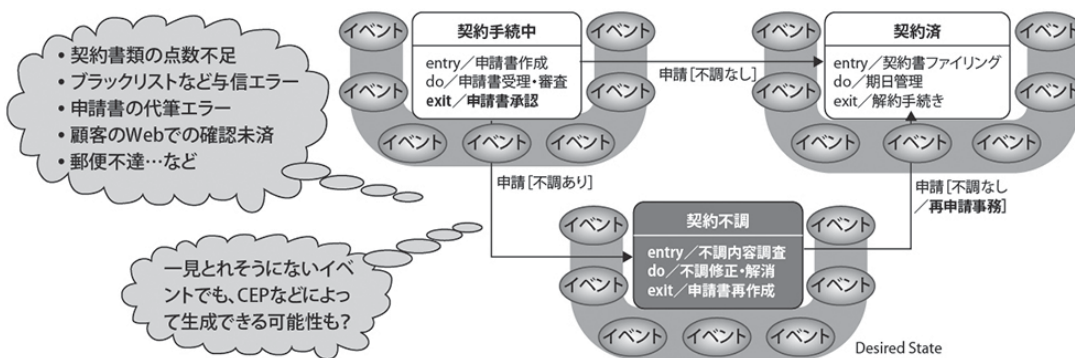


図8. ビジネス・イベント候補の抽出

うとする企業にとって、単なるイベント処理ミドルウェアの導入や局所的なイベント活用に陥ることなく、ビジネスに価値をもたらす網羅的なイベント識別を基礎にしたビジネスの価値向上を実現するBEPソリューションを構築することが可能になる。BEPソリューション構築においては本手法によるイベント・モデリング作業を実行することを検討してほしい。

今後の課題としては、以下の点が考えられ、価値ある現実のソリューションを構築するには、これらの課題についても取り組んでいく必要がある。

- 当手法では課題領域の絞り込みを行っているが、この考え方をより進めて課題領域における状況 (Situation) を識別する手法の導入
- 上記の状況 (Situation) において実行するビジネス・アクティビティー (Action) のマッチング (Situation-Action Matching) のモデル化や Action 実行後の Situation の変化の捕捉などの Situation Management や Action Management に関する手法の強化
- 多数のイベント候補が導出された場合に、実装するイベントを決定する手法の検討
- 導出されたイベントを実装するための実装面のデザイン・パターン
- 実際の事例における当モデリング手法の検証

### 謝辞

本論文の作成にあたっては、日本アイ・ビー・エムのテクニカル・コミュニティーである Technical Experts Council of Japan (TEC-J) のBEP (Business Event Processing) SIG (Special Interest Group) のメンバーの研究活動から多くの示唆をいただきました。あらためて深謝いたします。

### 参考文献

[1] Schulte, Roy Gartner Research: "Event Processing: Enhancing Business Applications Through EDA" (Gartner SOA Summit 2007).

[2] Schulte, Roy Gartner Research: "Event Processing Scenario: Two Ways to Capture the Value of Events" (Gartner SOA Summit 2008).

[3] @ITリアルタイムなイベント駆動型ビジネスを実現する「イベントストリーム処理」, <http://www.atmarkit.co.jp/ad/sonic/apama0611/apama.html>, (参照 2009年5月8日).

[4] David Luckham: "The Power of Events", Addison Wesley (Boston), ISBN-13:978-0-201-72789-0 (2002).

[5] 星島 洋一: SOAを加速するイベント処理技術の価値! - CEP (複合イベント処理) as "SOA's next big thing" -, 渋谷テクニカルナイト講演 (2008).

[6] Complex Event Processing, <http://complexevents.com/>, (参照 2009年5月8日).

[7] Event Processing Technical Society, <http://www.ep-ts.com/index.php>, (参照 2009年5月8日).

[8] Yao, W., Li, Z., Chu, C-H, and Mullen, T.: (November 2008) "Leveraging Complex Event Processing for RFID Applications: A Case Study in Hospitals", In Decision Science Institute (DSI) Annual Conference, Baltimore, MD, (November 2008).

[9] Hugh Taylor et al: "EVENT-DRIVEN ARCHITECTURE – How SOA Enables the Real-Time Enterprise" Chapter7-9 Case Study より, Addison Wesley (Boston), ISBN-13:978-0-321-32211-1 (2009).

[10] 株式会社オージス総研: その場で使えるしっかり学べるUML2.0, 秀和システム, ISBN4-7980-1239-4 (2006).

[11] Ali Arsanjani, Ph.D.: Service-oriented modeling and architecture, <http://www.ibm.com/developerworks/webservices/library/ws-soa-design1/>, (参照 2009年11月28日).



日本アイ・ビー・エム株式会社  
アドバンスド・テクノロジー・センター  
デザインセンター  
シニア IT アーキテクト

石井 旬 Jun Ishii

#### [プロフィール]

The Open Group Master Certified IT Architect, IBM Certified SOA Solution Designer. SOA, クラウド, モデリング, EA, EDA など, 先進技術のエンタープライズ・システムでの活用や設計に従事。技術研修講師や外部講演も努める。



日本アイ・ビー・エム株式会社  
ソフトウェア事業  
ソフトウェア・テクニカル・セールス&サービス  
シニア IT アーキテクト

蓮見 竜太 Ryuta Hasumi

#### [プロフィール]

先進ソフトウェア技術に関する社内戦略の立案と推進, お客様向けソリューションへの適用を担当。



日本アイ・ビー・エム株式会社  
ソフトウェア研究所  
WebSphere サービス  
テクニカル・ソリューション・アーキテクト

榎本 聡 Satoshi Enomoto

#### [プロフィール]

ビジネス・インテグレーションおよびビジネス・アクティビティー・モニタリングを中心とした先進技術領域におけるミドルウェア製品ベースのソリューション開発, パートナー様向けのソリューション展開, お客様向けソリューションへの適用および技術支援を担当。