

企業システム間連携を見越した業務分析指針についての一提案

A Proposal of Business Modeling Guidelines which foresaw cooperation between Enterprise Systems

井田明男[†]

Akio Ida[†]

[†] 株式会社オーガス総研

[†] Osaka Gas Information System Research Institute.

要旨

近年、顧客価値の洗練を図るべく、企業を跨るシステム間の連携への要求は高まるばかりである。その実現のためには、企業間のビジネス連携のメリットの追求や SOA の考え方に基づいた独立性の高いサービスの切り出しなど、業務の専門家にとっての論理的な側面を重視した業務分析の指針が必要となる。しかし、業務分析におけるモデル化やその実現のための要素技術仕様は年々肥大化・複雑化し、業務専門家のみならず情報システムの専門家にとっても仕様を理解し使いこなすのが難しくなっている。そのため業務フローの作成法も統一されていない。そこで、今回は将来の企業システム間連携にスムーズに繋がるシンプルで実践しやすい業務プロセスモデリングのガイドラインを提案し、ガイドラインに副って作成されたモデルの実現に適したアーキテクチャを検討する。

1. はじめに

ビジネスにおける企業間の連携、いわゆる B to B (Business to Business) の取引は、受発注取引をはじめとして、古くから行われてきた(図 1) が、近年は顧客価値の洗練や多様化を図るために、より緊密・迅速・柔軟な連携が求められるようになってきた[1]。この傾向は、複数の企業からの関係者による企業間のビジネス連携のメリットの追求や SOA の考え方に基づいた独立性の高いサービスの切り出しなど、論理的な側面をより重視した業務分析の実施が求められるようになってきたことを意味する。

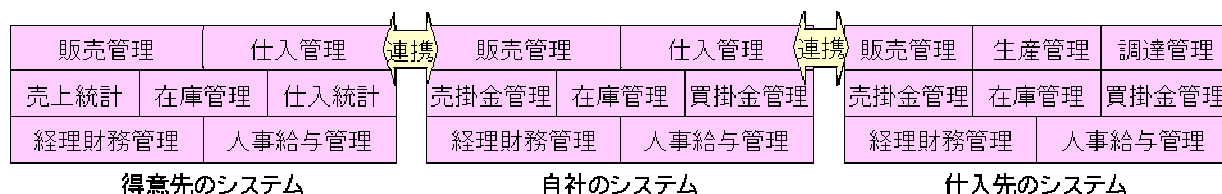


図 1 企業を跨るシステム間連携のイメージ

複数の企業におけるそれぞれの業務プロセスを可視化したものを持ち寄って、よりよい連携のための業務を検討する必要がある。また、検討結果についても複数企業からの経営者や業務分析者の間で合意形成が行われなければならない。けれども、例えば、同じ企業の同じ業務(例えば、受注業務)を対象として 10 人の業務分析者に業務フローを描かせたとすると、10 種の異なった業務フローが作成される可能性がある。まず、ビジネスのどこからどこまでを 1 枚の業務フローに描くのかの範囲が分析者によって異なる。ある者は受注から出荷までを対象に描くが、ある者は受注から売上計上までを対象にするかも知れない。さらに、業務フロー上に配置される個々の作業の粒度が異なる。一般的に、手続き型のプログラミングに精通した者ほど詳細な作業を切り出す傾向がある。これは UML や BPMN 等で表記法が規定されていたとしても、仕様が複雑すぎて利用しにくかったり、仕様を理解したとしても表記すべき内容までは規定されていないことが原因である。このような状況では複数企業からの関係者間で業務プロセス間の連携について議論し、合意形成を行うことが難しくなる。

そこで、本稿では、まず、経営者や業務専門家にとっても理解しやすいと考えられるシンプルな業務分析のためのガイドラインを提案し、つぎに、そのガイドラインに従って作成された成果物を実現するのに適したアーキテクチャを検討する。

2. ガイドラインとアーキテクチャ

企業を跨る情報システム間の連携を実現しようとするならば、業務分析段階からの準備が必要である。本稿では、ビジネスプロセスモデリングを対象としたガイドラインについての提案を行う。

2.1. ビジネスプロセス識別のガイドライン

企業の活動はすべて、企業の外部要素である、ビジネスアクター（典型的には「お客様」）からのイベントを契機として（典型的には、「注文」）によって、受動的に起動され、ビジネスワーカ（典型的には、「営業担当者」）によって遂行される1つ以上の「作業」の集合と捉えることができる。これを「ビジネスプロセス」と呼ぶことにする[2]。本ガイドラインでは、1つのビジネスプロセスは、当該プロセス起動の契機となったイベントに対する一定の応答を、イベントの発信元であるビジネスアクターに返却することによって完了すること[3]を要求する。なお、イベントはビジネスワーカによる先行する行為なしで認識可能な状況でなければならない[4]。

2.2. 作業識別のガイドライン

また、「作業」を、単一のビジネスワーカによって実施される、インプットとアウトプット（典型的には、「伝票」の記入や「台帳」への追記・更新）が明確な仕事と定義する[4]。ビジネスワーカとはビジネスアクターと異なり企業の内側の存在である。1つの作業は、当該作業開始の契機となったインプットの受け取りを契機として、所定のアウトプットを後続の作業に提出することによって完結するものとする。先行する作業のアウトプットが、後続する作業のインプットなり、それは取りも直さず後続する作業が起動されるためのイベントでもある。図2はこれらの概念間の関係をUMLクラス図で表現したものである。

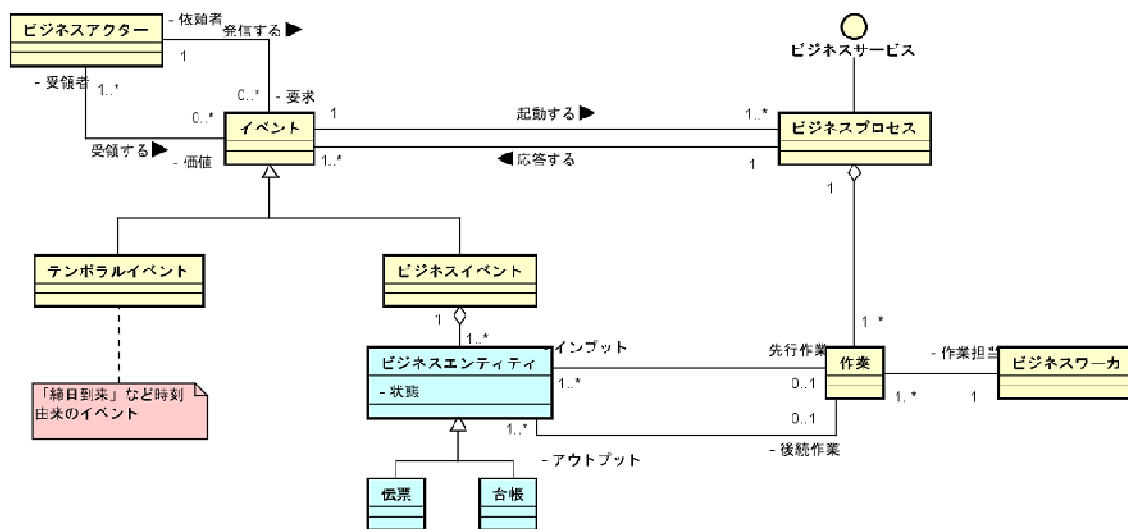


図2 本ガイドラインにおけるイベント、ビジネスプロセス、作業の関係

2.3. 業務フロー作成のガイドライン

本ガイドラインではビジネスプロセスと1対1で業務フローを作成するものとし、その際の留意点は次の2点である。一点目は、作業の前後には、必ず当該作業のインプットおよびアウトプットとなるエンティティ（具体的には伝票や台帳）をオブジェクトノードとして表記することである。ただし、エンティティであっても当該作業にとって参照のみに用いるエンティティは表記しない。なぜならば、当該プロセスにおいて参照のみに用いるエンティティのライフサイクル責任は他のビジネスプロセスにあると考えられるためである。二点目は、条件分岐や繰り返しを表記しないことである。それらを表記しな

ければならない事態が生じるのは大抵は作業の粒度が細かすぎることが原因である。これらは、業務フローを過度に複雑にせず、作業の粒度と完結性を適切に保つために重要なポイントであると考える。

2.4. ガイドライン適用の具体例

簡単な例として、ある卸売業者A社を考える。A社はお客様からの注文を受けて、お客様に商品を出荷している。注文が舞い込むことが「ビジネスイベント」で、商品を出荷することが、それに対応する応答である（図3）。

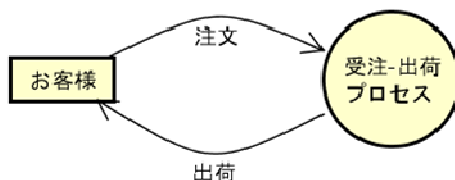


図3 卸売業者A社のビジネスプロセスの1例

注文を受けることが契機で、出荷が応答であるため、この業務プロセスを「受注-出荷プロセス」と命名することにする。次に、「受注-出荷プロセス」の中味を考える。このプロセスは、ビジネスワーカーである営業担当者と在庫管理者の作業の連携で実現されるものとする。図4はこの様子をUMLアクティビティ図で表現したものである。

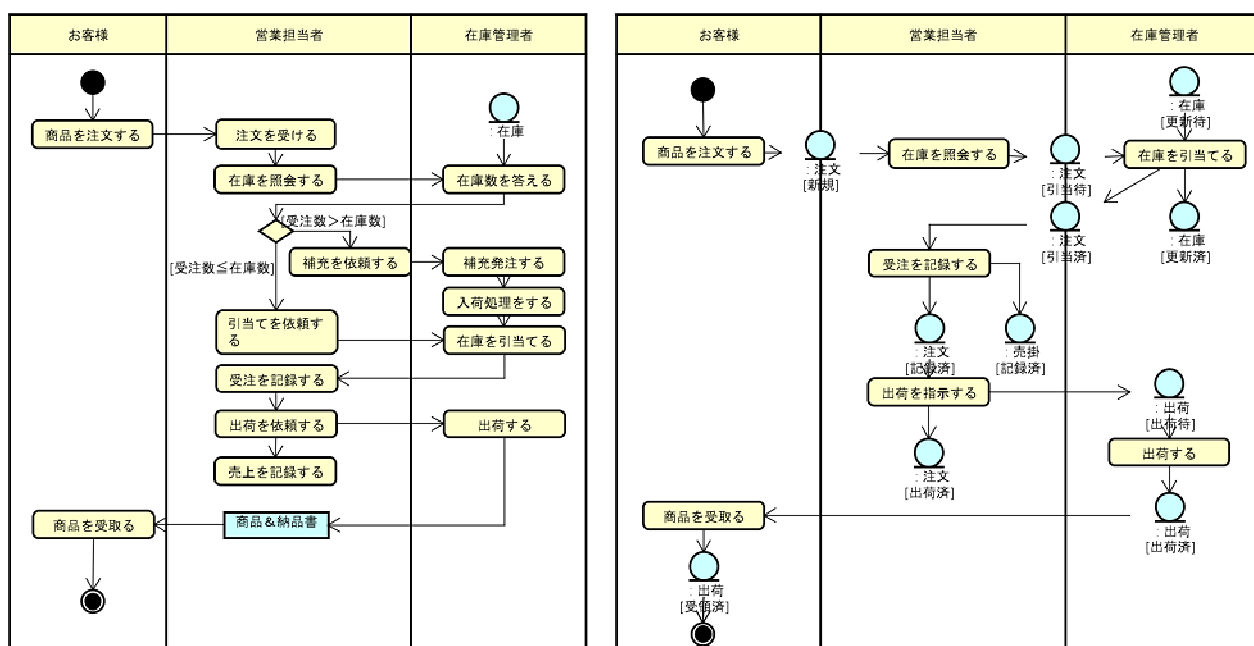


図4 業務フローの例（左側のフローは作業の切り出しが明確でない）

図4の左側のフローは本ガイドラインに従っていない例で、作業の入出力が明確になっていない。右側は本ガイドラインに則った例で作業の入出力が明確な例である。上述の通り、作業の前後に当該作業のインプットとアウトプットを必ず表記しつつ、オブジェクトノードには、その状態をかぎ括弧の中に表記している。このことによって作業には必ずインプットとアウトプットがあり、それはインプットをアウトプットの状態に進捗させるためのものであることが明確になる[5]上、インプット-作業-アウトプットから成るセットを作業ユニットと定義すれば、どの作業ユニットも簡単に業務フローから取り出したり、組込んだりを柔軟に行うことができるようになる。

2.5. 実現アーキテクチャとの関係

以上見てきたように本提案のガイドラインに従えば、業務分析の段階ですでに完結した独立性の高いビジネスプロセスおよびそれを構成する作業ユニットを自然に切り出すことができる。

さて、ビジネスプロセス全体およびそれを実現する業務フロー上の作業については、コンピュータ処理に適したものと、手作業によらざるを得ないものが混在している。コンピュータタスク化することに適したものについては、さらにタスクの公開を検討する必要がある。この場合、連携先の企業に提供すべきビジネスプロセスや公開すべき作業については、メールボックスなどに代表されるメッセージキューを介した非同期のプロセス間通信を行うタスクとして統合的に実装するのがよい。なぜならば、メッセージキューにイベントに付随するデータが到着したことを契機として、タスクの実行が開始され、必要な処理が、連携先のタスクが監視しているメッセージキューにデータを書き出すことで完了する形に統一することで、再利用性と独立性の高い完結したサービスを実装できるからである（図5）。

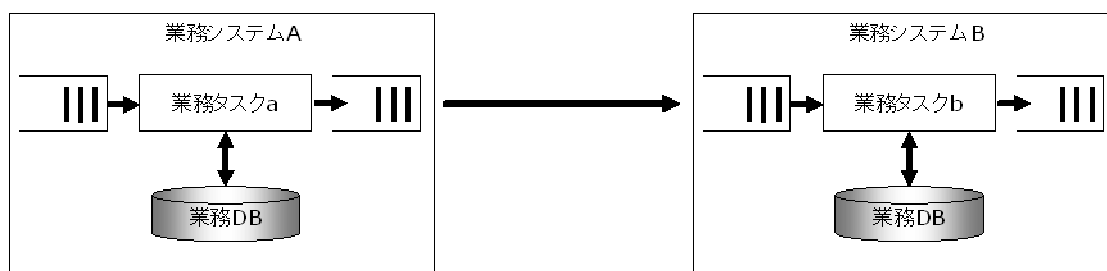


図5 メッセージキューを介した業務システム間連携のイメージ

3. 効果と展望

3.1. 本提案により期待される効果

業務プロセスをモデリングする際に本提案のガイドラインに従うと、大きく3つのメリットが考えられる。1つは、ビジネスモデルの検討がやりやすくなることである。ビジネスアクターのタイプを考えることにより、その企業にとっての顧客セグメントが見えてくる。また、顧客セグメントが明らかになると、そこから発せられるイベントが想定しやすい。結果、その企業が提供すべきサービスが明らかになる。2つ目は、ビジネスプロセスや作業の粒度の問題から属人性を排除することができることである。ビジネスプロセスの境界が明確になり、1枚の業務フローの範囲が確定するため、複数の分析者による分担作業も計画しやすくなる。3つ目は、個々のビジネスプロセスもさることながら、業務プロセス上に配置される作業が完結したものとなり、後にWebサービス化を行う場合にも大抵はそのままの粒度でサービス化を検討できることである。

3.2. 今後の課題と展望

業務分析の手法として本稿が提案するガイドラインを普及させるためには、具体的なサンプルを豊富に掲載した手引書が必要であると同時に、イベントに付随し、プロセス間で授受されるエンティティのモデリングガイドラインも必要となる。これらのテーマに対して今後も継続的に取り組んでいきたい。

参考文献

- [1] 天野敦之, 「宇宙とつながる働き方 経済を回復させるたった一つの方法」, 綜合法令出版, 2009.
- [2] Scott W. Ambler, John Nalbone, Michael J. Vizdos, "The Enterprise Unified Process," 2005. 株式会社オーグス総研監訳, 「エンタープライズ統一プロセス」, 翔泳社, 2006.
- [3] Ellen Gottesdiener, "The Software Requirements Memory Jogger," 2005. 平山輝, 藤井拓監訳, オーグス総研訳, 「実践ソフトウェア要求ハンドブック」, 翔泳社, 2009.
- [4] 渡辺幸三, 「業務システムのための上流工程入門」, 日本実業出版社, 2003.
- [5] 中村善太郎, 「もの・こと分析で成功するシンプルな仕事の構想法」, 日刊工業新聞社, 2003.