

---

# 第1章 ビジネスのモデル化

本章では、ビジネスのモデル化を図る際の前提事項、及びビジネスモデリングの方法論が備えるべき要件について整理する。

## 1.1 ビジネスモデリングの前提

### 1.1.1 静的モデル・動的モデル

およそモデル化を図る全ての対象物には静的側面と動的側面があり、そのモデル化を図る場合も、対象物の静的側面に着目した静的モデルと、動的側面に着目した動的モデルが存在しうる。

建築模型、自動車のクレイモデルなどは静的モデルであり、天体運行を表す数式モデルや各種のシミュレーターなどは対象物の動的側面を表す動的モデルである。

ビジネスを静的側面、動的側面に考え分けるなら、静的側面として組織・体制、機能的構造、期末財務計数などが該当し、動的側面としては業務プロセス、作業フロー、処理手順などが該当し、それぞれの観点にフォーカスした静的モデル、動的モデルが存在する。組織図、財務諸表、業務フローチャートなどがその例である。

しかしながら、もしビジネスを総合的な視点で捉えようとするなら、静的側面、動的側面のいずれかのみに着目するのではなく、その両面を表現する方法論が必要になる。フローチャートやBPMN（後述「2.4 BPMN」参照）など、ビジネスをモデル化する手法の多くが、プロセスやフローのみをモデル化する手法に止まっているが、これらビジネスのプロセスのみに着目したモデルはビジネスの一側面を表現しているにすぎない。このようなアプローチは、対象物

の構造を考慮せず、その動きのみに着目しているという意味で、「車の構造を知らずに車を運転している」「海図を持たずに航海している」ような危うさがある点に留意する必要がある。

### 1.1.2 見える化

ビジネスの特徴は「ビジネスは目に見えない」という点にある。ビジネスプロセスの「見える化」「可視化」という言葉が使われるが、この言葉が示すようにビジネスは目で見えにくいものだ。

会社の中で従業員が動くさまを見てみると、ビジネスも目に見えるような気もするが、「今、何をしているのか」「目的は何なのか」「それをやらないと何が起こるのか」などは、ただ仕事の流れを眺めているだけでは殆ど分からない。この傾向は特にホワイトカラーの業務の場合顕著である。このような従業員の業務内容を明らかにするためには、業務処理マニュアルを見たり、インタビューなどをしないと分からないのである。

したがって、ビジネスのモデル化を行う際には、見えているものをミニチュア化するというアプローチではなく、目に見えないものを視覚化する方法論が必要になる。

このように目に見えない抽象的な対象物をモデル化する場合は、何らかの明確な視点・軸足が必要になる。物体を可視光線で見た映像、赤外線で捉えた映像、X線で造影した映像がそれぞれ大きく異なるように、この軸足の置き方によってモデルの形も大きく異なってくる。

ビジネスをモデル化する視点の候補としては、売上、収益、コストなどの財務計数、商品、顧客、組織、機能、リスク、扱っている各種のモノなどがありうるが、このようなモデルの軸足はモデル化の目的に応じて決定される。財務的な評価を主目的とする場合は財務計数が軸足になり、組織戦略を考える場合は組織体系が、商品戦略を考える場合は商品ラインアップがモデル全体の軸足になる。これらの目的ごとに個別最適化されたビジネスモデリングの手法は、既述のように財務諸表、組織図、業務フローチャートのような形で現時点においても存在しているが、ビジネスの評価・分析のために広く用いられている総合的・汎用的で定番的な方法論と言えるものは存在していないのが現状である。

### 1.1.3 モデルとIT

近時、新たに提案されるモデルの多くは、何らかITの存在を前提としたものであり、ITなくしては有用性を持たないモデルが多い。バーチャルリアリティや、衝突解析シミュレーターなどがその例である。

この場合「ITを前提とする」とは、単にコンピュータの演算能力に依存するという定量的な側面だけでなく、質的な意味で、コンピュータにしかできない機能をモデルの前提としている場合がある。例えば、シミュレーションモデルの出力値を3次元グラフで表して、これを上下左右に回転させて視覚に訴える方法などである。このような視覚化は、無機的な数値の羅列を人間が直感的に理解しやすいように情報を加工しているという意味で、このグラフ化のプロセスもひとつのモデル化である。

このようにモデルとITの境界線が曖昧になり、ITなくしてはモデルが成立せず、また、コンピュータのアウトプットを見ないとモデルの有用性が理解できないケースが増えてきている。これは「コンピュータシステム＝モデルそのもの」「モデルの説明＝コンピュータシステムの機能説明」となっていることを意味する。

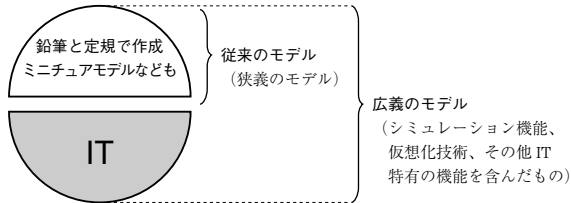


図1 モデル記法とITの活用

一方、モデルというものは鉛筆と定規で表記されるべきという暗黙的な先入観が未だにあり、鉛筆と定規で書かれることを前提にした固有の記号が用いられることが多い。電気回路記号、地図記号、フローチャート記号などがその典型例である。

UML (後述「2.5 UML」参照) などをはじめとするシステム開発用のモデルも、様々な記号に意味を持たせる「紙と鉛筆」型の表記から脱していない。そろそろ、このような従来型モデルの先入観から開放されて、ITの利用を前提

とした新しいモデリングのあり方を考えるべき時期がきているように思われる。

### 1.1.4 システム開発におけるビジネスのモデル化

システム開発作業においても、ビジネスのモデル化は必須の作業となっている。このようなビジネスのモデル化は、システム開発プロジェクトの中でも最初の工程である要求定義・要件定義の工程で実施される。

一方で、システム開発プロジェクトの失敗原因の多くが要件定義工程にあるというのは多くの分析資料によって示されている\*1。システム開発における要件定義工程では、システム導入後の新しいビジネス運営形態が策定され、同時に新システムの機能要件が定義されるが、この際の理解不足、コミュニケーションギャップ、詰めの甘さ、思い込み、見切り発車などが主因となってプロジェクトが破綻する。

このような事態を回避するためにはシステムオーナー、或いはシステムユーザーであるビジネス関係者に、新しいビジネスの形と新システムに対する要件を十分に語らせ、システム開発者がこれを的確に理解することが重要となる。このようなユーザーのイメージとロジックを可視化して、関係者の相互理解と合意を図ることが要件定義工程の目的であって、このような作業の繰り返しと集積があって初めて要件定義が確実なものになる。

このような要件定義工程においてシステム開発者側が陥りやすいミスは、システム開発者が「自分の言語」でシステムの要件を定義し、ビジネス関係者がその内容を十分に理解しないままに実開発作業が始まってしまうような事態である。

ビジネスの現場を一番知っているのはビジネス関係者であり、新システムに対するイメージを明確に持っている者は、システム発注者であるプロジェクトオーナーをはじめとするビジネス関係者以外に存在しない。一方で、ビジネス関係者の多くはシステム開発に関する知識を持たず、システム開発で用いるダイアグラムの見方を知らない（通例、覚えようという動機を持たない場合も多

---

\*1 『日経コンピュータ』2003年11月17日号「プロジェクト実態調査」によると、失敗プロジェクトの35.9%が不十分な要件定義に原因があったとされており（第1位）、同誌2008年12月1日号「第2回プロジェクト実態調査」では36.7%（第2位）となっている。

い) のが普通である。このようなビジネス関係者に対して、「システム開発者の流儀」でドキュメントを書きおこしても、ビジネス関係者は辟易するばかりで、或いは見方を理解するだけで精一杯の状態となり、このようなドキュメントでは自部署に持ち帰って周囲の人間と内容をすり合わせることもできない。これでは、いくら膨大な量の要件定義文書を作成し、これにビジネス関係者の承認印が押されていたとしても、本来探り出すべき潜在的な要求や微妙なニュアンスを文書化することができず、要件定義の作業全体が上すべりなものになってしまう。このようにして出来上がったシステムはユーザーのニーズや現場の実態を反映しないものになり、「話が違う」「聞いていない」ということになる。

近時、システム開発作業はオブジェクト指向の方法論（次ページコラム「オブジェクト指向」参照）をベースになされることが多く、このオブジェクト指向の記法を標準化したものがUMLである。一方で、UMLはビジネス関係者の視点に立ったものではなく、あくまでシステム開発者用の記法である。UMLの中にも、一部「ビジネスモデリングや要件定義フェーズで活用できる」とされているダイアグラムも存在するが、UMLが提供するダイアグラムの多くはビジネス関係者が一見して理解できるものではなく、また、ビジネス関係者の立場に立った付加価値が追求されたものでもない\*2。

要件定義は本来ビジネス関係者が主体となって進めるべき工程であって、ビジネス関係者の意向・イメージを最大限引き出すために、要件定義工程におけるシステム関係者は、助言者ないしは調整役に止まるというのが本来の姿である。また、このような工程における作業成果物の書式・体系は、ビジネス関係者が自身の考えやイメージを表現するという目的に最もフィットしたものを選択すべきであって、ビジネス関係者にとって書き方や読み方が分からないような表記法は採用すべきでない。

---

\*2 システム開発者が、システム開発者のインタレストから、対象業務の概要を理解したり、システム開発のための材料を収集するためのダイアグラムを作成するという事と、ビジネス関係者がビジネスを強化したり、新たなビジネスを創出するためにビジネスをモデル化するという事とは、その意味合いが異なる。「UMLを用いたビジネスモデリング」という言い方も一部でなされるが、これらの殆どは前者の意味合いで使われている。

プロジェクト全体の成功という目的のためには、最も失敗しやすい要件定義工程において「最適な手段」を選ばねばならず、この場合の「最適な手段」とは、ユーザー側の担当者、或いはプロジェクトオーナーや管理者が、特段の事前知識がない状態でも理解可能であって、ビジネス表現という観点から高い付加価値を持つ手法を意味する。

**【コラム】 オブジェクト指向**

オブジェクト指向とは、対象領域をオブジェクト（モノ、概念）に分解してモデル化する方法論であり、特にソフトウェア開発の世界では最新かつ標準的な方法論となっている。

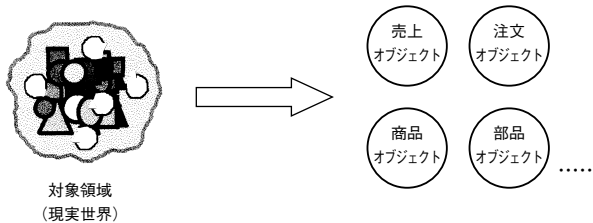


図2 オブジェクトによるモデル化の例

オブジェクトは、目に見える（tangible）具体的なモノでも構わないし、目に見えない（intangible）抽象的な概念でも構わない。目に見えるモノ系のオブジェクトとして商品、製品、部品、原材料、各種の機器類、顧客、取引先、社員、各種帳簿などが挙げられ、目に見えにくい概念的なオブジェクトとして注文、契約、納品、検収、売上などが挙げられるが、これらはいずれもオブジェクトとみなすことができる。

オブジェクト指向の考え方は1970年代前半に開発されたプログラミング言語Smalltalkを起源とするものとされ、その後ソフトウェア開発の世界を中心に発展してきた。

オブジェクト指向におけるオブジェクトは、それぞれに固有のメソッド（操作、機能）と属性（プロパティ、データ項目）を持つが、原則としてこれらの内部構造や内部処理は他のオブジェクトは知らな