
第3章 ビジネスモデル表記法の研究

— ビジネスモデルを表記する方法の提案 —

第3章 ビジネスモデル表記法の研究

第3章では、本論文における第2の課題である「ビジネスモデルから情報モデル化の間に大きな隔たりがあり、シームレスな情報システム開発に繋がらない。」を解決する目的で、ビジネスモデルを可視化するための表記方法を研究する。システム関係者に限らず、第三者にも理解できるように統一された表記方法に則って、ビジネスモデルを分析・検討でき、且つ、情報モデルの作成にシームレスに繋げる手法について提案を行う。換言すると、ビジネスモデル設計を具現化するためのモデル表記法を研究し、ビジネスモデル設計から情報システム設計へ直接つながるシームレスな表記法のあるべき姿を示す。

まず、従来のビジネスプロセスを表記する代表的ツールである IDEF、ARIS、PDR、UML を取り上げて比較検討する。また、ビジネスモデルとそれを反映した情報モデルの間に大きな隔たりがあり、シームレスな情報システム開発に繋がらないという課題の解決策として、UML に基づくビジネスモデルから情報システム開発の可能性を研究した。ここでは、表記のルールが統一されており、理解しやすい共通方法で必要な項目に漏れのないビジネスモデル表記と情報モデル表記法の提案を行う。UML は最近脚光を浴びている統一的なモデリング言語であり、その可能性が注目されている。本稿では現実のシステム構築を効率的に行うための情報モデルの表記方法として UML に着目し、ビジネスモデルを UML によって表記する試みを行う。ビジネスモデルを描き、表現する記述法として必要十分な条件は何かを明らかにするために、従来の手法を比較検討して、有効なモデル表記を検討する。本研究では、ビジネスモデル研究の範囲としてサプライチェーン (Supply Chain) を対象にしている。そこでサプライチェーンのリファレンスモデルである SCOR モデルを研究した。

本章での研究の新規性は、UML によるビジネスモデル表記方法の提案である。

第3章の構成は、次のとおりである。

第1節では、ビジネスモデル表記について整理し、従来の手法である IDEF、ARIS、PDR、サプライチェーンのリファレンスモデルである SCOR (Supply Chain Operation Reference model) の比較研究を行う。

第2節で UML のシステムモデル表記について従来研究の特徴を述べ、UML によるシステムモデル表記の分析を行う。

第3節において、ビジネスモデル設計に UML を使用するための提案を行い、第4節で事例検証と考察をし、第5節で結論を述べる。

注) ARIS (Architecture of Integrated Information System)、PDR (Process Design Repository 業種別業務プロセス設計リポジトリ)、IDEF (Integrated Computer Aided Manufacturing Definition methodology)、UML (Unified Modeling Language)

第1節 従来のビジネスモデル表記法の比較研究

本章では、ビジネスモデルを理解しやすく表記でき、かつ、その後の情報システム構築にシームレスにつながる表記法を研究した。従来、ビジネスプロセスを表記する方法としてフローチャート、DFD（データフロー・ダイアグラム）、IDEFO、UMLなどがある。これらの目的は、業務を分析し、その内容を分かり易く記述することにより、いずれも有用な情報システムを構築するために採用可能なツールである。この為、情報の流れや商流は、把握できるが、一部の物流と金流は、把握できない場合がある。

この章では、ビジネスプロセスモデルの記述と分析、設計を行う技法とツールについて分析し、問題点を探る。これらの技法の代表的なものとして、本節では、ARIS、PDR、IDEF、SCORを取り上げ、これについての特徴と課題を分析する。

1-1 ARISによるビジネスモデル表記

ARIS（Architecture of Integrated Information Systems）は、ドイツのA. W. シェアー博士が開発したビジネスモデルの設計ツールを中心に、情報システム開発に必要な管理機能を付加したものである。主に、ビジネスプロセスの最適化とアプリケーションシステムの構築に活用される。ARISは、製品としてIDS.Scheer社が販売している。下記の図3-1は、ARISの支援範囲とARISの構造として5つのビューから構成されていることを示している。5つのビューとは、下記である[45]。

- 1) 組織ビュー：業務を執行する1責任を持つ組織単位を定義する。（例：営業部）
- 2) データビュー：重要な情報を定義する（例：顧客、製品、原材料コストなど）。
- 3) 機能ビュー：機能されるべき業務を定義する（例：製造計画作成、受注処理など）。
- 4) プロセスビュー：データ、機能、組織単位の関係性を定義する。
- 5) アウトプットビュー：ビジネスプロセスで作成されたアウトプットの定義をする。

上記のビューで定義された要素は、データベースで一元管理される。従って、異なるビューで一度定義すれば、他のビューでも使用できる。例えば、プロセスビューでの営業部オブジェクトと組織内の営業部オブジェクトは、同一オブジェクトと認識される。データの一元管理することによって、複雑になりがちなオブジェクト管理が容易に行える。つまりビジネスモデル同士が統合され、変更も容易に行うことができる[46]。

図3-1のとおり、ARISは3階層の階層構造になっている。

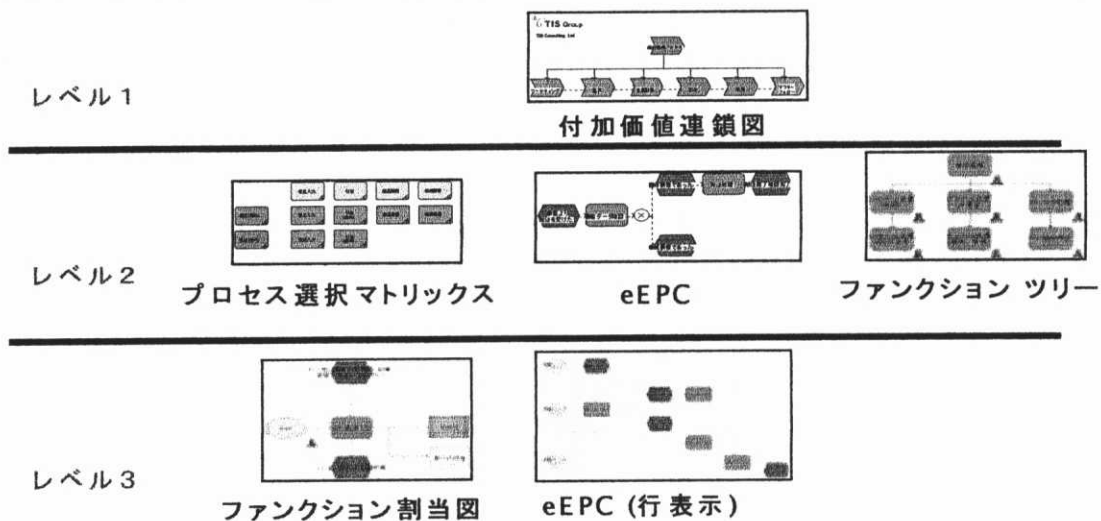


図3-1 ARISの階層（引用：IDSシェア社資料）

ARISで使用している表記法の特徴としては、eEPC (electronic Event Driven Process Chain) を使用している。下記にARISの概要を記述する。

ARISのレベル1階層は、会社全体の主なビジネスプロセスを示し、付加価値の生成を表す。基幹ビジネスプロセスを明確にし、その時間的流れを把握するために使用する。

レベル2階層は、ビジネスプロセスを把握するために使用する。

- ・プロセス選択マトリクスにおいて、ビジネスシナリオに対して使用されるビジネスプロセスを定義する。
- ・eEPCにおいて、ビジネスシナリオごとの業務フローの全体像（各ビジネスプロセスを連携したもの）を定義する。
- ・ファンクションツリーにおいて、連鎖しないビジネスプロセスを定義します（財務会計、管理会計等の管理業務関係）。

レベル3階層は、各ビジネスプロセスの詳細を把握するために使用する。

- ・eEPCにおいて、担当者のタスクにまでブレイクダウンされた業務の詳細を定義する。
- ・ファンクション割当図において、各タスクにおける入出力情報や処理担当者、使用システム等を定義する。

図3-2は、ビジネスプロセスの表示例を表している。

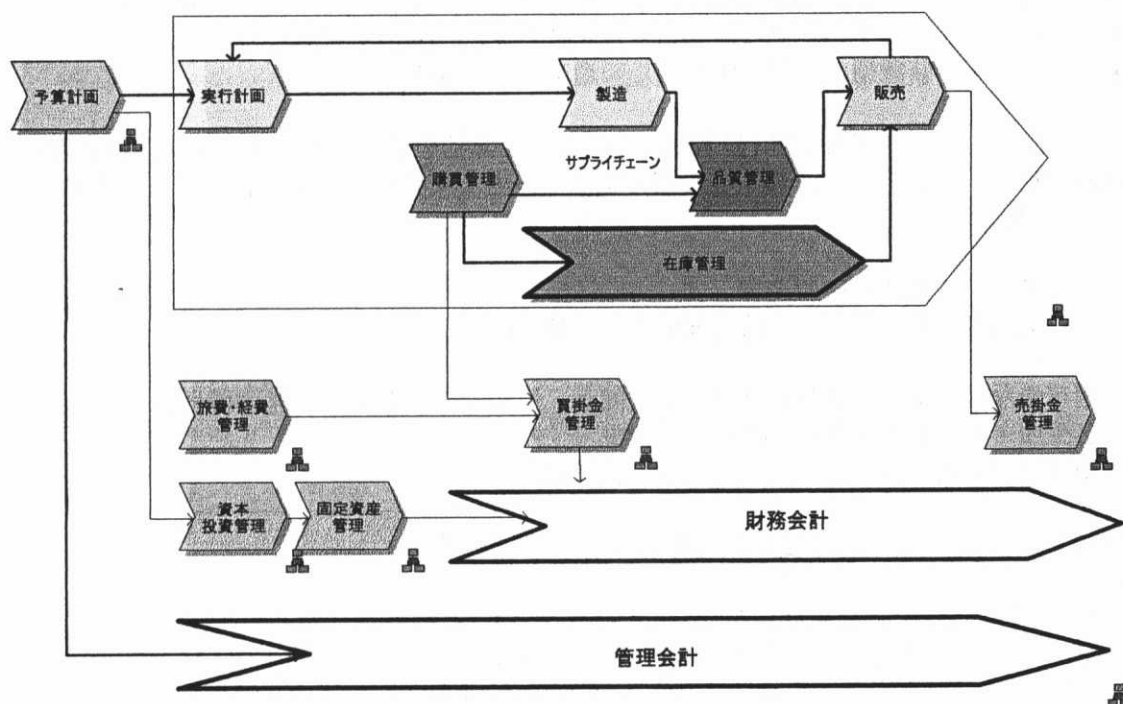


図3-2 ARISのビジネスプロセス表示例（引用：IDSシェア社資料）

ARISを使用する目的は、下記のとおりである。

- ・モデリングして表記する
- ・可視化することによって無駄な業務を洗い出す。
- ・仮説を複数立てAs isモデルと比較する
- ・To beモデルで業務改革後の影響を把握する・業務を様々に分析する

期待できる効果としては、複雑なビジネスのプロセスを構造化でき、プロセスの変更ができることがあげられる。また、再利用可能なテンプレートとして有効活用することも可能である。

1-2 PDRによるビジネスモデル表記

標準的な業務プロセスモデルとして作成されたのが、PDRである。PDR（業種別業務プロセス設計リポジトリ）は、製造業、流通業、情報サービス業など10業種の企業モデル（企業が持つべき業務処理機能を4ないし5階層で表現）とそれを操作するプログラムから構成されている。またいくつかの業務には、業務の説明、改善・改革の留意点および業界専門用語解説が付加されている。

PDRは、三井造船システム技研（株）、（株）Nixシステム研究所、システム企画研修（株）の業務に強い各分野のコンサルタントによって作成され、情報システム化（要件定義）でのシステム分析、ビジネスモデル構築に利用できるだけでなく、業務知識の習得および業務ノウハウの蓄積、業務マニュアルの整備でのマップとしても活用できる[44]。

（1）対象業種

1999年7月のリリース開始時には5業種であったが、その後バージョンアップを重ね、2000年3月にリリースしたVer. 1.2では10業種を対象としている。

- 見込生産・組立型消費財製造業 ・ エンジニアリング業
- 受注生産・部品製造業 ・ 日常生活財卸売業
- 見込生産・装置型消費財（化学製品）製造業 ・ チェーン・ストア（GMS）業
- 見込生産・装置型消費財（発酵食品）製造業 ・ レストラン・チェーン業
- 見込生産・装置型消費財（医薬品）製造業 ・ 情報サービス業

なお、見込生産・組立型消費財製造業、受注生産・部品製造業、見込生産・装置型消費財（化学製品）製造業のPDRは、以下のような製造業モデルとして作成しており、列記した業種等で利用可能である。

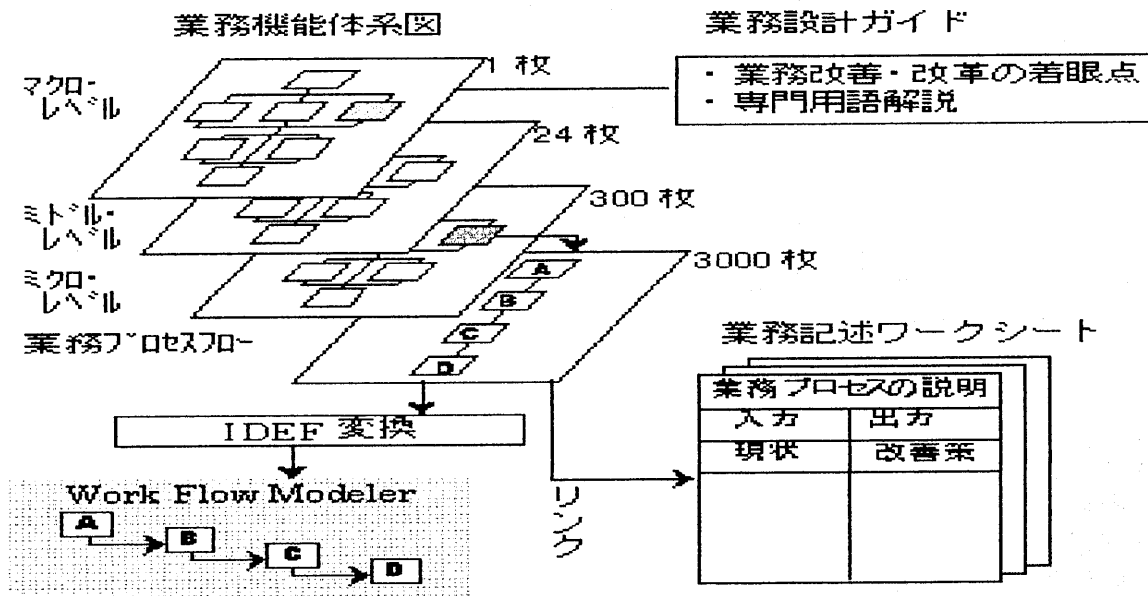
- * 見込生産・組立型消費財製造業
適用可能な製造業は、家電、情報通信機器、精密機器、玩具、自転車、など
- * 受注生産・部品製造業
適用可能な製造業は、自動車部品、家電部品、情報通信機器部品、精密機器部品、など
- * 見込生産・装置型消費財（化学製品）製造業
適用可能な製造業は、トイレタリ、飲料、など

（2）10業種分のコンテンツとそれを動かすソフトウェア

対象とする各業種のコンテンツと、それを加工・修正できるソフトウェアとをご提供している。コンテンツは、「標準業務プロセスモデル」と「業務設計ガイド」とから構成されている。標準業務プロセスモデルは、業務プロセスを階層的かつ論理的に整理し、業務設計ガイドは、標準業務プロセスモデルを使って、新しい業務実施やシステムを検討する際に参考になる情報を提供する。図3-3は、PDRのプロセスモデルの構造を示している。

標準業務プロセスモデル

（枚数は見込生産・組立型消費財製造業の場合です。）



Work Flow Modeler は META software の登録商標です。

図 3-3 PDR の構造 （ 引用：（株）Nix システム研究所資料 ）

ERP パッケージなどを利用する開発、あるいは SCM 開発の場合、新システムの機能を検討するマップとしての利用（現状とパッケージとの機能対比や、それを受けての新システム案検討を行う）が可能である。

- 一般的にパッケージが提供している機能はコア機能であって、検討している対象範囲の全機能を提供しているわけではない。
- SCM システム等の検討の際も、PDR にすべての業務機能の地図（マップ）が用意されているので、関連する業務をもれなく押さえることができる。
- インタフェースをとるべきシステムや機能を確実に押さえることもできる。

PDR は、情報システムを作る前提として、必要な業務プロセスモデルをマクロレベルから詳細レベルへと記述している。PDR で書かれたビジネスモデルに近い業務においては、大変、有効なツールと考えられる。しかしながら、その表記方法には、標準化されたものが少なく、汎用性や転用に対して柔軟に対応は難しいと思われる。この意味で UML 等での記述がされればさらに有効と考える。

1-3 IDEFによるビジネスモデル表記

IDEFとは、Integrated Computer Aided Manufacturing Definition methodologyの略で、Softec社の構造化分析設計手法をベースにしたシステム設計・分析・記述の手法のことである。IDEFには、IDEFファミリーと呼ばれるIDEF0～IDEF14までの手法がある。

IDEFはビジネスプロセスを可視化することを主眼とした手法として、米国でCALCの発展とともに充実・普及していたものであり、近年、日本でも注目を浴びている。理解しやすい簡明な図示表現に特徴があり、ソフトウェアツールも提供されている、構造化分析・ワークフロー分析・ER図記述・オブジェクト思考設計などから成る各種分析手法の総合体系である。

1-3-1 IDEF0によるビジネスモデル表記

ビジネス手法に関連が強いのは、IDEF0 Function Modelingである。IDEF0は、業務や生産のプロセスをすべて箱と矢印で記述するようになっており、視覚的に理解しやすくなっている。国際的に標準化されたオープンな手法であり、アメリカ政府標準であるFederal Information Processing Standardとしても採用されている。

IDEF0はシステムを簡単なシンボルで記述し、業務プロセスを可視化することによって、システムにおける、決定・動作・活動のアクティビティ（機能）をトップダウンで階層的に詳細化してモデル化する手法で、システムがどのようなアクティビティから構成され、それらがどのような関係にあるのかを表現することを主眼としたモデリング手法である。特にIDEF0は、業務分析の初期段階で利用され、他のIDEF手法群を利用するためのベースを供給する。

IDEF0はIDEFの開発の主契約者であったSoften社の構造化分析手法SADT (Structured Analysis and Design Technique)をもとに開発された。SADTはDouglas.T.Ross氏がMITの教授時代に開発した技法で、その後Softec社の社長となり商品化したものである。IDEF0は使用方法も簡単であるため、現場の実務者が業務改善・改革策を検討し、具体的な進め方をイメージ的に示すのに適している。

< IDEF0の目的 >

- ・ 企業の基幹プロセスを明確にし、ドキュメント化し、共通認識を作る
- ・ プロセスの一部として、どのようにファンクションが相互に関連しているのかを理解する
- ・ 不可価値及び非不可価値アクティビティを明確にする
- ・ 改善の対象となる機能とアクティビティを明確にする

IDEF0の主な目的は企業の業務を識別しドキュメント化することであり、この業務だけでなく、各業務が機能する環境を表現することが出来る。品質管理の分野でも、ISO-9000の2000年改訂版でも業務モデリング手法として取り上げられており、業務の担当者や責任を明確化する手段として注目されている。

< IDEF0の特徴 >

- ・ IDEF対象範囲を構造化されたグラフィック表現で表すモデリング言語である。
- ・ アメリカ空軍をはじめとしてアメリカの政府開発プロジェクト、民間会社で長年の使用実績がある。
- ・ IDEF0のモデルはシステム分析・設計、企業分析、BPRのための機能と機能の関係、データやオブジェクトをあらわすことが出来る。
- ・ IDEF0はCASE (=Computer Aided System Engineering)の方法やツールから独立したモデリング言語ではあるが、関連させて使用することは可能である。
- ・ 多種多様なビジネス・製造・企業活動において、どのようなレベルに対しても詳細に表現できる。
- ・ システム分析者・開発者・ユーザ間での情報の共有ができる。
- ・ 標準化のために原稿のアクティビティを記述する。
- ・ 新規のアクティビティ担当者が、業務を短時間で習得するためのガイドラインを提供できる。

1-3-2 IDEF0の表記方法

IDEF0ダイアグラムは図3-4に示すように、「ボックス」と「アロー」から成る。ボックスは

アクティビティを表し、詳細なアクティビティが必要になれば階層的に分解して表現する。アローはアクティビティへの、またはアクティビティからのインタフェースを表し、入力・コントロール・アウトプット・メカニズムの4種類に分類され、それらを総称して ICOM という。

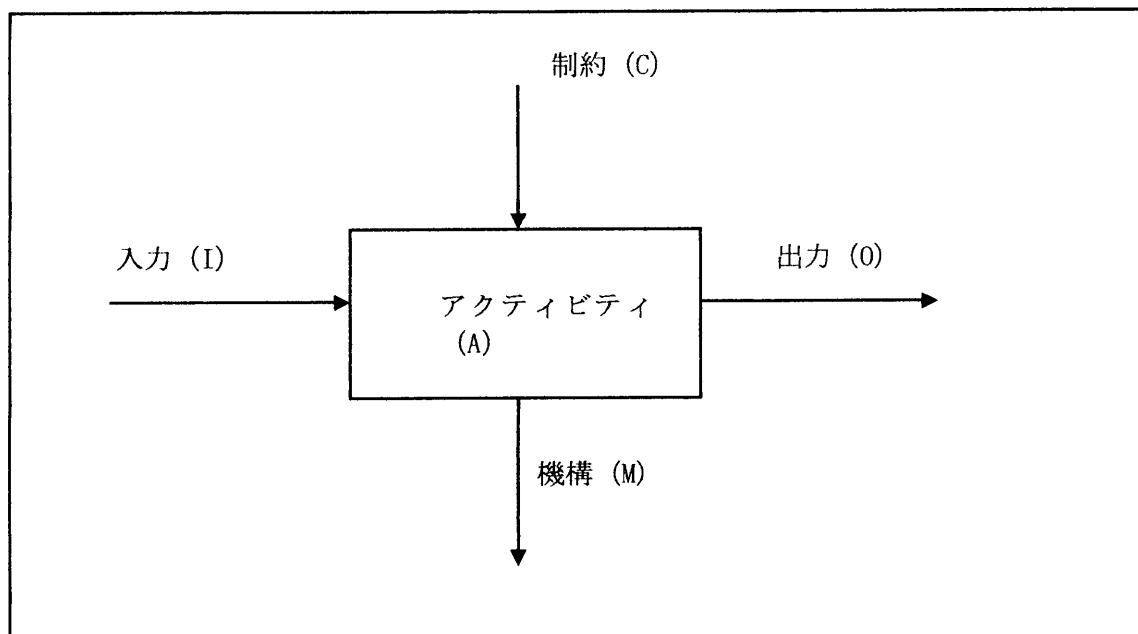


図 3-4 IDEF0 ダイアグラムの基本形

図 3-4 の説明としては、処理の内容を表すボックスを「アクティビティ」と呼び、動詞句で表現される活動・機能を示す。入力を示すのは、「I (インプット)」と呼び、左から挿入されるアロー。処理をされる情報や物（データ・作業内容・投入資源など）を示す。出力を表すのは、「O (アウトプット)」と呼び、右へ出るアロー。アクティビティの産物としての情報や物（データ・商品・サービスなど）を示す。上からの矢印を「C (コントロール)」と呼び、上から挿入されるアロー。アクティビティを制約する情報や物（制約条件・指示など）を示す。

下からの矢印を「M (メカニズム)」と呼び、下から挿入されるアロー。アクティビティを遂行する手段（ヒト・機械・情報システム・ツールなど）を示す。

また、モデルを構築するにあたり、目的 (Purpose : なぜ、このモデルが開発されたのか、どのように使用されるのか)、コンテキスト (Context : 目標は何か)、視点 (View Point : どんな視点から何を見ようとするのか) を明らかにする原則がある。これらが明確でないとアクティビティやモデル自体が定まらず、ICOM の表現が曖昧になるだけでなく、分析者と読者との間で誤った解釈をしてしまう恐れがある。

なお、IDEF0 は NIST の FiPS183 にその文法が詳細に定義されている。

アクティビティは図 3-5 に見られるように階層化が可能である。最上層 A0 をブレイクダウンしたものが A1、A2、A3 を第 1 階層として展開する。さらに、その中の A2 をブレイクダウンすると第 2 階層、その中の A23 をブレイクダウンすると第 3 階層として表現できる。従来の方はノードツリーや WBS (Work Breakdown Structure) を用いて業務を階層的に表現していたが、IDEF0 ではボックスで示されているアクティビティ毎に ICOM の関係をアローとして視覚化できる。さらに、インプットとアウトプットの他にコントロールとメカニズムが加えられているため、他の構造化手法とは異なり、システムが持つ特性を容易に表現できる手法である。

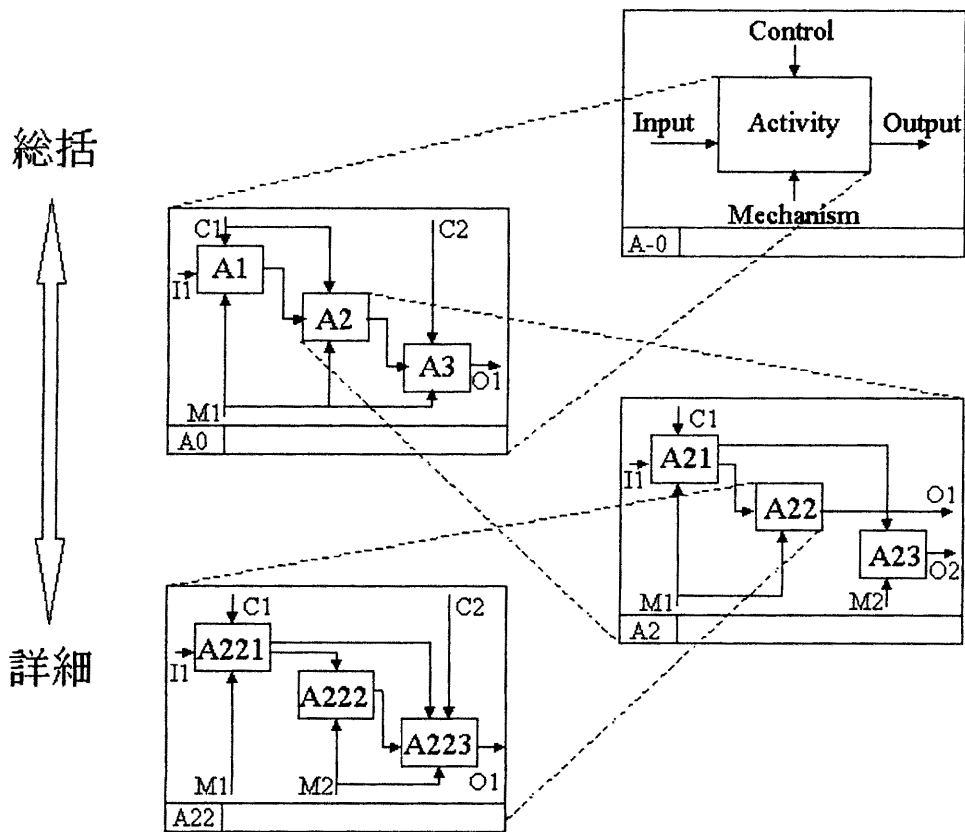


図 3-5 アクティビティの階層化

IDEF0 は別名 “機能モデリング手法” と言われ、企業や組織における業務の仕組みを（責任と権限を伴う）機能的側面からモデル化する手法である。次に IDEF ファミリーの中で、IDEF3 について述べる。

1-3-3 IDEF3 の概要

IDEF3 はプロセス記述獲得手法と呼ばれ IDEF 手法の中では比較的新しく開発された手法である。この IDEF3 は、IDEF0 によるアクティビティ・モデリングを正しく行うための補助手段として開発された。すなわち IDEF0 で業務モデルを作成するには主題全体に関する業務知識を満遍なく知っていることが必要であり、そのような業務知識を収集するための手段として IDEF3 が開発され手法の名前も “プロセス記述獲得手法” と呼ばれている。

1) IDEF3 のモデル表現形式

IDEF3 (プロセス記述獲得手法) は、作業の流れを表現するプロセス・フロー図とそのフローで加工されるオブジェクトの状態を表現するオブジェクト状態遷移図の 2 つの図式表現形式を定めている。プロセス・フロー図はボックスで表されるプロセス (UOB: 振る舞いの単位) を作業の時間的順序関係を表す矢印 (リンク) で結び、流れの分岐や合流はジャンクションと呼ばれる接合子で表す。

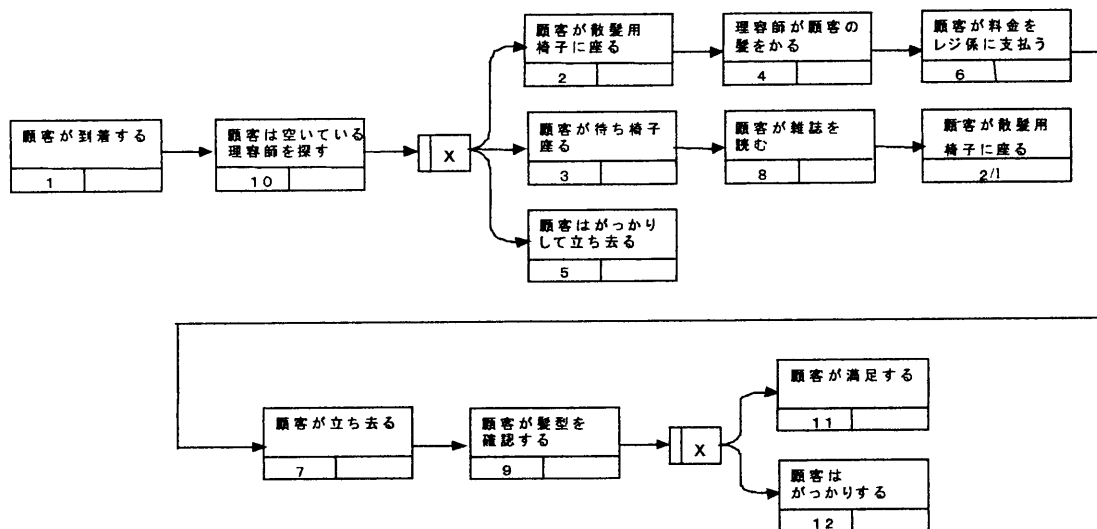


図 3-6 IDEF3 のプロセス・フロー図

IDEF3 手法は厳密なモデル（目的や視点を一つに絞るなど）であることを求めておらず、任意の場面での作業の流れをそのまま表現することを意図してプロセス・フローの単位をシナリオと呼んでいる。（図 3-6 参照）

2) IDEF3 プロセス・フローへの付加情報

各プロセスにはそのプロセスを更に詳細化したプロセス・フロー（複数可）を階層的に持つこと、各プロセスに係るオブジェクトや制約を関連付ける詳述（エラボレーション）といわれる仕組みを持ち関係する情報を付加することが出来る。

3) IDEF0 と IDEF3 の関係

IDEF3 の役割の一つは IDEF0 による業務モデリングの前に業務知識を獲得することであり、もう一つは出来上がった業務モデルを作業の流れとして詳細化するプロセス・モデリングの手段としての利用である。通常、現場作業に熟知している担当者でさえも、最初から IDEF0 の機能モデリングを行うのは難しい。そのため、表記し易いプロセスをモデル化する IDEF3 より、プロセスの全体を可視化してからこれに基づいて IDEF0 モデルを構築する手順の方が実務レベルには、大変有用である。すなわち IDEF0 を挟んで事前の役割と事後の役割がある。

1-4 SCORによるビジネスモデル表記

SCORとはSupply Chain Operation Reference modelの略で、非営利団体のSCCが、部門間・企業間を横断して実行されるSCMの標準として使用するために、策定・開発したサプライチェーンモデル記述仕様であり、サプライチェーンのリファレンスモデルである。

SCMのビジネスモデルを各企業が個別に作成していたのでは、さまざまなモデルの表現方法が混在してしまい、企業間でモデルを検討する際にどのようにして整合性をとるかが問題となる。このため業種、業界などに依存せず標準化する必要があると考え、こうした問題点を解決するために発足したのがSCCである。SCCでは、SCMのモデルの表現以外にも、SCMの効果測定で使用する評価指標の定義の明確化など、さまざまな観点で、SCMの標準化を推進している。

SCORは、サプライチェーン業務プロセスのリファレンスモデルである。各産業での広範囲な利用を目的として、汎用性、共通性、簡潔性を求めているため、設定されているプロセス内容が粗くなっている部分もある。しかし一方で企業を超え、国境を越えてサプライチェーンを形成していこうとするときに、業務のキャッチボールをする相手と協業できる共通のフレームワーク、共通の言語はSCORが唯一のものである。

本論文では、サプライチェーンを研究対象としている。SCMは企業や組織の壁を越え、1つのビジネスモデルとして経営資源や情報を共有し、全体最適を目指してプロセスの無駄を徹底的に削減する経営手法でもある。また、SCMの目指すのは、調達・生産・流通・販売というサプライチェーン全体の最適化を図ることで、顧客満足度の向上、リードタイムの短縮、在庫削減、キャッシュフロー増大などを同時に目指す戦略的な経営手法である。図3-7に一般的なサプライチェーンの範囲を示す。高度成長時代の供給を重視した時代では、生産部門では生産効率を追求し、物流部門は輸送コストを最小にするため大ロット輸送を行い、販売部門も欠品率を最小にする必要があり、これらの個別の部分最適化は共に在庫を増大させた。また、会計帳簿上在庫は資産として評価されるので、在庫を持って生産・販売効率を上げ、製造・販売原価を下げ、売上を確保すれば、見掛け上の利益は大きくなった。しかし、最近のキャッシュフロー会計の視点からは、在庫は悪であり、リスクを増す要因と考えられるようになってきた。これを改善するための経営手法としてSCMが登場してきた背景がある[16]。

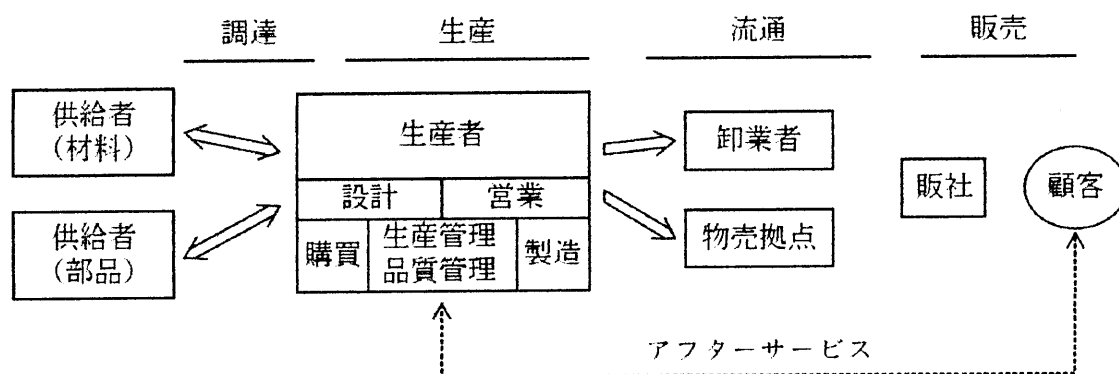


図3-7 サプライチェーンの流れ

1-4-1 SCMの概念とビジネスモデルの変化

本研究では、ビジネスモデルを表記するのに適したツールを研究し、その結果としてUMLを選択した。次に、その選択したツールで何が描けるのかを検証するために、実際の企業または、実際に近いビジネスの形態をUMLで表現してみる必要があると考えた。本節では、サプライチェーンの分野を取り上げ、その実現手段としての情報システム化では、ERP構築方法を取り上げて検

証する。本章では、サプライチェーンにおけるビジネスモデル表記を取り上げる。また、サプライチェーンを理解し、表記するのに役立つファレンスモデルである SCOR についても分析を加えた。

SCM と ERP の関係を図 3-8 に示す。SCM は MRP、ERP を含んだ大きな広がりとして表現することができる。そして、SCM の特徴である複数業者間の連携を含んだ考え方であることが理解できる。ただし、SCM が MRP、ERP を含んでいるので、MRP、ERP は必要ではないという意味ではなく、複数企業にまたがった販売・調達・物流・在庫の計画を管理する SCM、企業内の基幹業務を管理する ERP、企業内で生産に関する資材所要量を管理する MRP と、それぞれ守備範囲が異なる。

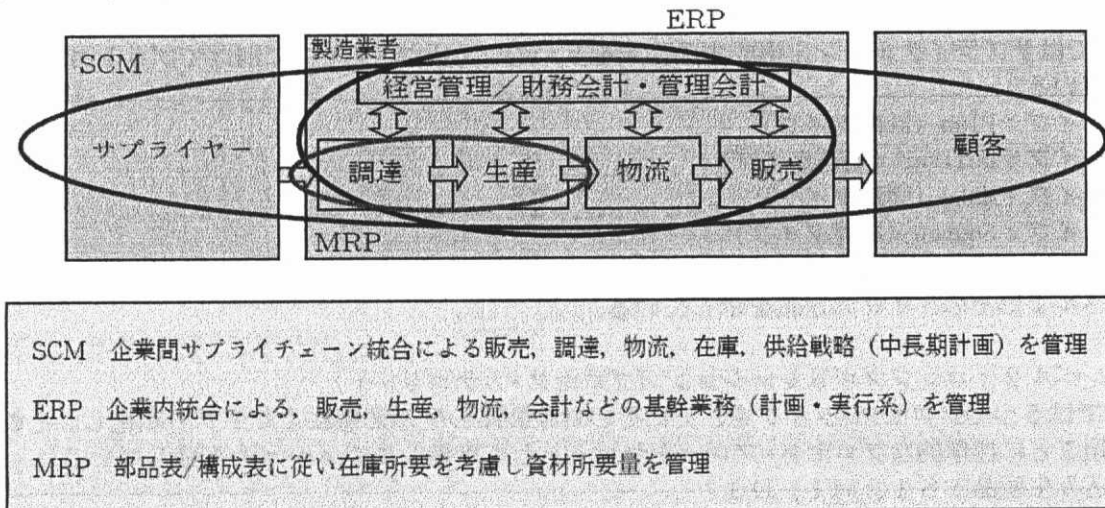


図 3-8 ERP と SCM、MRP の関係図

SCM の目的としては、調達・生産・流通・販売というサプライチェーン全体の最適化を図り、顧客満足度の向上、リードタイム短縮、在庫削減、キャッシュフロー増大などを同時に目指すことである。ここでは、SCM を語る上でのキーワードとして、“WIN-WIN の関係”と“全体最適”について述べることにする。

“WIN-WIN の関係”とは、従来、業者と業者の間には、納入業者とメーカー、メーカーと卸業者、卸業者と小売業者といった業者間の力関係が存在し、強者が利益を独占してしまう“WIN-LOSE の関係”であった。しかし市場も成熟し、メーカーがただモノを作っても売れる時代ではなくなった。悪い言い方をすると、全体で利益が出ないので、強者が奪う利益を生み出すことさえできなくなっている。

小売業者から物流業者へ順に伝わる売れ筋情報では、事実がゆがめられ、情報伝達にも時間が掛かり、メーカーが商品を投入するころには顧客ニーズも変わってしまうなどの問題が起こっていると推測できる。そこで、メーカー、卸業者、小売業者が持っている顧客の購買情報、製品在庫状況の情報を提供・共有化することにより、鮮度の高い、生きた情報を入手する必要性が出てきた[43]。

情報を共有化する企業群、つまり サプライチェーンを構成するメンバーが集まり、新しい利益を共同で作り出し、お互いがパートナーシップの関係になり勝ち組みになること、これを“WIN-WIN の関係”と言い、SCM のポイントといえる。

SCM が目的とするのは、“全体最適”である。ここでの全体最適の“全体”とは、サプライチェーンを構成するプレイヤーすなわち業者全てを指す。従来は、それぞれの業者で個別に最適化していた業務を、サプライチェーンに関係するすべての業者で共有化した情報をもとに、在庫削減、リードタイム短縮に努め、チェーン全体で最適化することをいう。従来の部分最適化から、全体最適化へ考え方を変えることも SCM のポイントとなる。この 2 つをいかに実現していくかが SCM の目的である。

SCM を導入後、その成果を評価する上で「何をもちて良しとするか」、その指標となるべき基準の設定は、困難を要することが予想される。実際に各社で用いている指標を調査したが、

内容的には同じものでも、呼称の違い、単位の違い、評価期間のサイクルの差など、並べて比較するにはかなりの手間と時間が必要とされることがわかった。客観的な視点で、正当に評価を与えてくれる基準が必要となった。

1-4-2 SCORモデルの概要

本節では、SCORモデルの概要について解説する。

SCORはサプライチェーン業務を4つの階層レベルにまとめている。

1) レベル1：プロセス・タイプ

SCORではサプライチェーンを構成する作業を5つのプロセス・タイプに別けてプロセス・フロー化している。

- Pタイプ：Plan（計画タイプ）
 - Dタイプ：Deliver（配送タイプ）
 - Mタイプ：Make（製造タイプ）
 - Sタイプ：Source（調達タイプ）
 - Rタイプ：Return（返品タイプ）
- レベル1はそのタイプの定義を示している。

2) レベル2：コンフィギュレーション（プロセス・カテゴリー）

SCORでは5つのプロセス・タイプを、更なる対象製品の生産形態によりサブ分類化し、そのサブ分類ごとに標準的なプロセス・フローをセットとして準備しモジュールを形成している。

- 見込み生産品：S1、M1、D1
- 受注生産品：S2、M2、D2
- 受注設計生産品：S3、M3、D3

	階層		プロセス構成要素	内容
	#	レベル		
SCORの範囲	1	トップレベル (プロセス・タイプ)	Plan, Source, Make, Deliver, Return	レベル1=SCORモデルをベースとして行う業務範囲と内容の定義。競合、パフォーマンス、目標がセットされる。
	2	コンフィギュレーションレベル (プロセス・カテゴリー)	Sp, Mp, Dp, Rn	レベル2=SCORモデルの持つ26のコア・プロセス・カテゴリーの中から適切な組合せをすることで、目標とするサプライチェーンの姿、形を明確にし、実行戦略策定のベースとする。
	3	プロセスエレメントレベル (プロセスを分析する)	D2:受注生産品の配送 D2.1: 引合と見積, D2.2: オークション, D2.3: 納入日決定, D2.4: オークション	レベル3=選ばれたマーケットにおいて首屈よく競争できる会社の能力を策定する。 ・プロセスエレメント策定 ・インフラ/IT/IT/IT情報策定 ・プロセス・パフォーマンス・メトリクス策定 ・ベストプラクティス分析とベスト・プラクティス構築 ・サプライチェーン実行戦略のファインチューニング
SCORの範囲外	4	詳細実行レベル (プロセス要素を分析する)		レベル4=サプライチェーンにおける特定のマーケット・セグメントの詳細定義を行う。競争優位を達成し、変化するとシナリオ環境に適合するため実践を定義する。

図3-9 SCORの階層構造 (SCOR4.0より引用)

SCORでは図3-9で示すように、これらのモジュールを組み合わせることで需要先から供給元までのサプライチェーン・モデルを組み合わせて構成できるよう工夫されている。図3-10は、D1（見込み生産品の配送）の下位階層レベル3の参照プロセスを示している。

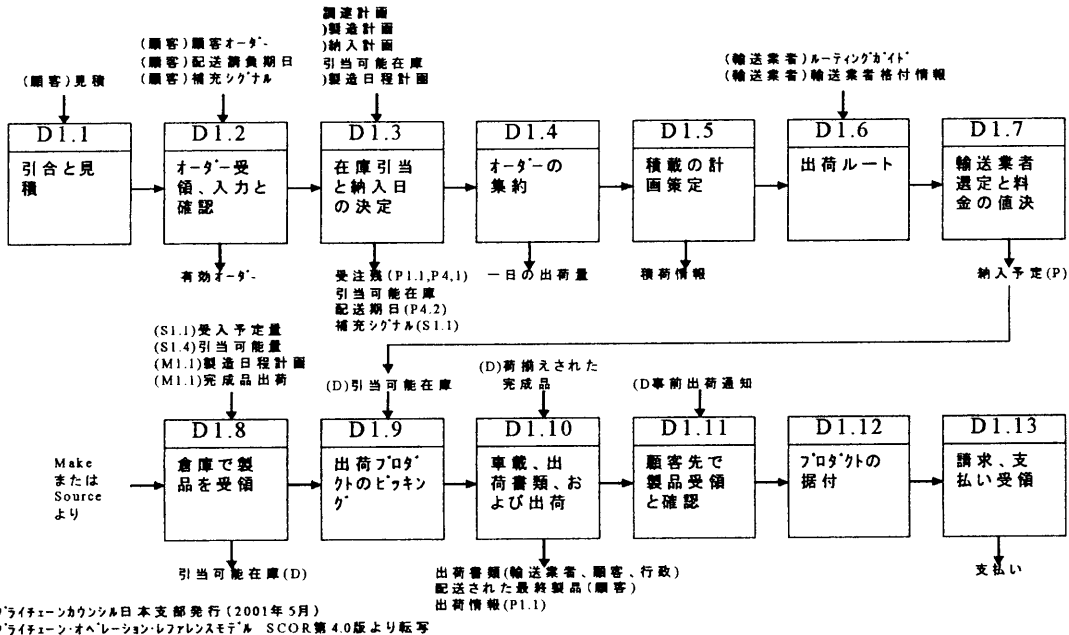


図 3-10 レベル 3 の参照プロセス・モデル (SCOR 第 4.0 版より引用)

3) レベル 3 : プロセス・エレメント

レベル 3 はレベル 2 で分類定義されたモジュールごとに、その作業の一般化された標準の流れ (プロセス・フロー) を定義しており、作業プロセスの参照モデルを形成している。更にレベル 3 ではプロセス・フローを構成する個々のプロセスに一般化された情報定義しており、下記の表 3-1 のとおりである。

プロセス要素	D1.2 オーダーの受領、入力と確認	
パフォーマンス属性	柔軟性/応答性	オーダー受領リードタイム オーダー登録リードタイム
	コスト	顧客オーダー作成コスト オーダー登録維持コスト
	信頼性	特定不可
	資産	特定不可
ベストプラクティス	電子商取引・オンライン在庫引当・確認	EDIアプリケーションと統合オーダー管理
	注残等のリアルタイム参照	特定不可
	補充情報等の遠隔測定	POS使用による顧客動向データ
	遠隔オーダー入力	特定不可
	自動多段階与信チェック	オーダー/財務の統合管理
合理的価格設定	活動基準原価法(ABC)、顧客ごと、品目ごと	
入出力情報	インプット	(顧客)オーダー
		(顧客)配送請負期間
		(顧客)補充シグナル
	アウトプット	有効オーダー 支払い受領(場合による)

表 3-1 各プロセスへの定義情報 (SCOR 第 4.0 版より引用)

表 3-1 における「パフォーマンス属性」とは、そのプロセスが影響する時間、コスト、資産、信頼性に関する評価指標項目をさす。「ベストプラクティス」とは、そのプロセスにおいて考えられる最新の改善要素を示す。「入出力情報」とは、各プロセスに出入りする一般的情報をいう。

4) レベル 4：詳細実行レベル

レベル 4 はレベル 3 で定義されているタイプ別に分類された汎用的なプロセスとその定義情報を特定のビジネス・ユニットに適用させ実践的なサプライチェーン・プロセス・モデルに詳細化する。そしてこのレベルでの詳細化では次のような情報について補う必要がある。

・プロセスの対象物情報

レベル 3 ではそれぞれのプロセスが処理する対象物をオーダーや完成品といった抽象的な表現にとどめている。実際のサプライチェーン・モデルとして完成させるには、その入出力情報を参考として、具体的な対象製品や伝票類を定義する必要がある。

・作業の担当等の資源情報

当然ながら、レベル 3 の参照モデルでは作業を誰が担当するか、あるいはどこで何を使って処理するかなどの情報はほとんど含まれていない。

・制約となる与件条件やルール

サプライチェーン・プロセスの効率性を評価するには、それらのプロセスに課せられている各種の与件条件や守るべきルールなどの制約との係わり合いを明確化しておく必要がある。各プロセスにこれらの制約やルールを関連付けておくこともサプライチェーン・モデルを具現化するには大変重要である [47]。

1-4-4 SCOR によるビジネスモデルの特徴

SCOR モデルの特徴としては、SCM モデル分析における実施手段を明確化している。これによりモデル分析手順の雛形に沿って作業を進めていくことが可能となる [36] [37]。

・ サプライチェーンのモデル分析・設計のための業界標準仕様

モデル分析や設計における仕様を明確化しており、企業ごとの仕様が共通化される。

・ サプライチェーンの確実な検証（流れの可視化と数値評価）

SCM のモデル上をどのように、モノ・カネ・情報が流れるかを規定するとともに、評価指標を定義しており、これによって検証作業の際の基準が企業間で一本化される。

・ ERP/SCM ソフトから独立したモデルの構築・評価

SCM モデルの分析手法、評価指標などは、ERP/SCM の各パッケージベンダーや各コンサルティングファームなどが SCC と同様規定しているが、SCC ではこうしたベンダー依存の指標ではないため独立性が保たれる。

・ 取引先企業や SCM ベンダーへのビジネスモデル仕様提示

SCOR で規定することで、企業間における仕様検討において、公開性が高まり、SCM 全体の検討作業がスムーズになる。また、こうして検討したビジネスモデルを SCM ベンダーへ開示することで、SCM ベンダー側の適合性などの検討作業もスピーディーに行われるといった効果も考えられる。

今後、サプライチェーンを構築する際には、以下の手順が適当であると考えられる。

- 1) SCOR 指標に基づいてベンチマーキングを実施し自社の水準を知る。
- 2) SCOR やベストプラクティスを参照しながらプロセスデザインを行う。
- 3) このプロセスモデルに対して、SCOR 準拠の ERP、SCM を適用する。

SCOR のプロセスは、企業経営中枢機能の 4 つのメインプロセスから構成される。企業毎に組織や業務内容、機能等は異なるため、SCOR は、計画・調達・製造・出荷（受注／納入）の大きく 4 つのプロセスでモデルを構成する。

SCOR では、各プロセスを多段階に定義、分析、構築できる構成となっている。ただし、SCOR で規定するのはレベル 3 までであり、レベル 4 以降は標準化の対象外としている。SCOR で規定しているレベル 1 からレベル 5 までの表現内容を図 3-11 に示す。このレベル 2 の表現によりサプライチェーンの流れの可視化を可能としている。

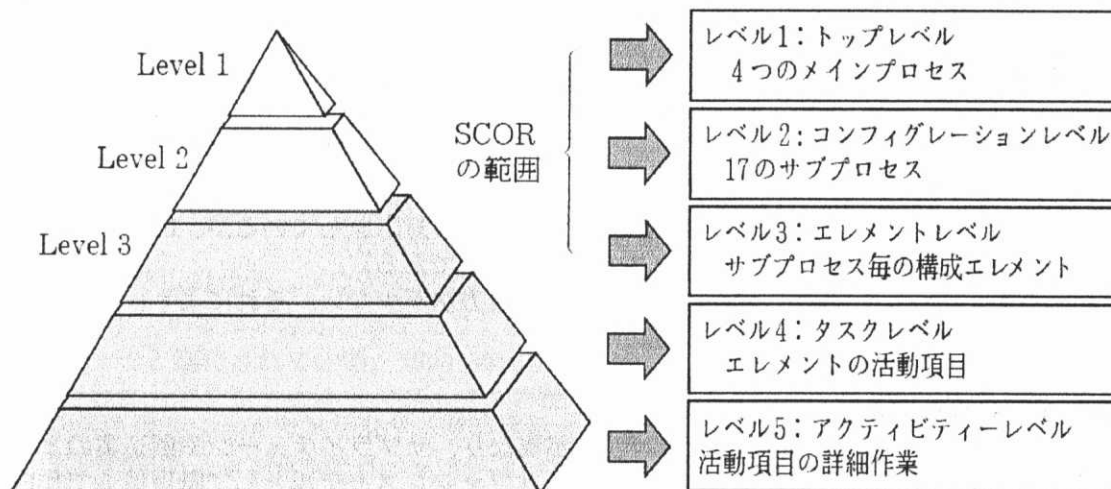


図 3-11 SCORの階層構造 (SCOR4.0 より引用)

1-5 ビジネスモデル表記方法の比較評価

前節で述べた従来のビジネスモデル表記法4つについて評価を述べる。

1-5-1 ARISの評価

ARISは、ERPのSAP R/3にも使用されたこともあり、世界規模で普及しているツールである。ARISは、階層構造で表現ができ、ビジネスプロセスを表示させるには、十分な機能があるといえる。R/3との親和性も高く、ビジネスプロセスからR/3への画面へ結びつけることも可能であり、こうしたことから、R/3導入の際には便利なツールといえる。

また、ERP導入と直接関係なくビジネスモデルを検討するツールとしての機能も充実させており、最近では、ARISで表記したモデルからUML表記に変換も可能になっている。この機能の充実によってARISでビジネスモデルを表記して、UMLモデリングにつなげることも考えられる。その場合、ARISからUMLへの変換機能の充実が求められる。

しかしながら、本研究で提案するようなビジネスモデルの多様な表現(視点と切り口による表記)を行うための機能がない。

1-5-2 PDRの評価

PDRは、10業種による情報システム構築に耐えうる詳細なビジネスプロセスモデルを持っており、その詳細な内容は、十分な業務機能説明がある。長い時間をかけて作成した労作と考えられる。PDRにある標準業務プロセスモデルは、それを使って情報システム構築を可能とするレベルの十分な情報を保持していると考えられる。しかしながら、PDRによってビジネスモデル設計を行う場合には、下記の点が課題となる。

- (1) 使用目的がビジネスモデルを検討する。または、ビジネスモデルを設計し、新たなビジネス形態を模索するといった場合には、表現種類が不足している。
- (2) システム構築をオブジェクト指向で行う場合には、再定義する部分が多くなる。
- (3) ERPなどの導入検討には、不向きである。(手作りシステムに向いている。)

1-5-3 IDEFの評価

IDEFは、国際的に標準化されたオープンな手法であり、広く研究されるツールであると共に多くの実績をもつ。ビジネスプロセスを容易に表現し、インプットとアウトプットを明確に表記する仕組みは、優れているといえる。しかしながら、発表から10数年以上が経ち、広く知られてはいるが、実務において普及しているとはいえない。最近の傾向としては、UMLにスタンダードを奪われたように見える。

IDEFの決定的な欠点は、2つあると考える。

- ・ オブジェクト指向に対応が弱い。(IDEF3により、その対応をしているが、UMLと比較すると、表現形式が少なくビジネスプロセスの表記が中心である)
- ・ 表記の種類が少なく、複数の切り口、視点での表現方法が少ない。それにより、モデルの十分な記述と理解がしづらいことがある。

1-5-4 SCORの評価

SCORは、前述の3つのツールとはカテゴリーが異なり、サプライチェーンや製造業のビジネスプロセスを記述したリファレンスモデルである。リファレンスモデルとして階層構造で作成され、内容も順次充実されており、参照モデルとして有効であると考えられる。しかしながら、SCORは、個々の企業に必要なビジネスプロセスの詳細までは描いておらず、この部分は、各企業の役割として位置付けている。また、ビジネスプロセスを具体化するための情報システムを設計するためのシステムモデリングの機能はない。オブジェクト指向の概念もないため、そのままシステム設計につながらず、システム化には別途多大な労力とコストを要することになる。実用方法としては、SCORをサプライチェーンのリファレンスモデルとして参考にしながら、自社やグループ間のサプライチェーンモデルをUMLで記述することにより、ビジネスモデルの設計・表記からそのビジネスモデルを実現するための情報システム開発へと繋げることが可能である。

1-5-5 比較研究のまとめ

本論文では、ビジネスモデル表記から情報モデル、情報システム開発に連携できる表記方法を求めている。シームレスな連携を行うには、ビジネスプロセスの内容と流れ及びデータ項目、処理機能などを上流のビジネスモデル表記から下流の情報システム設計に正確に伝達するシステムや情報共有の仕組みが必要である。概要レベルから次第に詳細な情報を伝達していくことが、ビジネスモデル表記の必要条件となる。

更に、ビジネスモデルを表記するには、下記の項目を満たす必要がある。

- ・ 多様な表現に対応できる表記が可能なこと
- ・ 情報連携ができること (オブジェクト指向対応)
- ・ 標準化されたもの

また、ビジネスモデルの変更に伴う、情報システムの柔軟な変更が求められる。こうした場合の対応として、オブジェクト指向の表記方法が求められる。前述の5つのビジネスモデル表記方法の評価から、従来の表記法は、上記のすべてを満たすことはできない。

上記の項目を満たす表記法として、本論文では、UML (Unified Modeling Language:統一モデリング言語)を採用し、不足する部分を本論文において提案しUMLによるビジネスモデル表記方法を提案する。次節でUMLに関して記述する。

第2節 UML 表記法の従来研究分析と考察

前節では、従来のビジネスプロセスを中心としたビジネスモデル表記の手法を分析し比較研究を行ったが、本節では、UMLによるシステムモデル表記の従来研究を分析、整理することにより、次節以降のビジネスモデル表記の提案に繋げる。

2-1 UML とオブジェクト指向について

UML (Unified Modeling Language: 統一モデリング言語) はOMG (オブジェクト・マネジメント・グループ) での標準が制定されており、これからの情報システム構築の中心的役割を担っていく技術の一つと期待されている。1990年代初めには、オブジェクト指向の開発方法論の検討が盛んに行われ、米ラショナル・ソフトウェア社(Rational Software)においてジェームス・ランボウ(James Rumbaugh)、グラディ・ブーチ(Grady Booch)、イバー・ヤコブソン(Ivar Jacobson)の3氏が中心となり、1997年UMLコンソーシアムが、UML1.1をリリースし、OMG(Object Management Group)がUMLを標準として採択した。その後、改定作業が続けられ、2001年にバージョン1.4、さらに2004年にバージョン2.0が発表されている。こうした経緯からUMLは、オブジェクト指向のシステム開発方法のなかで統一モデリング言語となった[59]。

現在では、UMLは情報システム構築のためのモデリングのほか、情報システム構築の前提となる環境、すなわちビジネスモデルを表現する手段としての役割も盛んに議論されている。本研究では、ビジネスモデル表記にUMLを適用する際の課題を明確にし、解決策としてUML拡張提案を行う。狭義のUMLに的を絞ってもクラス図を始めユースケース図、シーケンス図、アクティビティ図など様々なモデル表現形式が準備されており、それらを適切に活用することも容易ではない。ここではUMLによるシステム構築の中核はクラス図であるとの認識に立ち、そのクラス図設計にシステムに求められる要求機能を反映するための上流側の手段としてユースケース図やシーケンス図が存在し、それをシステム実装のために下流側に具体化していく手段としてコンポーネント図や配置図があると捉える。そしてSCOR情報をUMLに繋げる接点がユースケースであり、シーケンス図はそれを補強するものと位置付け、ユースケース図作成に繋げることを目的としたツールであると考え[48][50][51][52]。

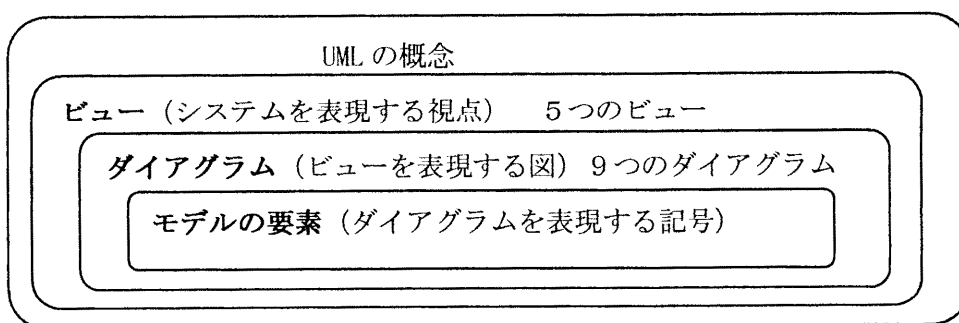


図3-12 UML 3つの概念

UMLの特徴として、表現力豊かで視覚的なモデリングが可能であり、システム、ドメイン、開発プロセスの枠にとらわれない。また、実装、開発方法論に依存しないことから、拡張性が高いと言われる。拡張性が高いという特徴を活かすためにUMLには、3つの概念が用意されている。その概念とは、「ビュー」、「ダイアグラム」、「モデル要素」の3つである。図3-12は、UMLに含まれる3つの概念の関係を図示している。

UMLは、さまざまなオブジェクト指向方法論のモデル記述方法を統一化する形で開発された記述技法であり、そのため、UMLが提唱される以前に利用されていたモデル表記法に比べ、表記で

きる範囲が充実している。その範囲を示す基準としてビューがある。

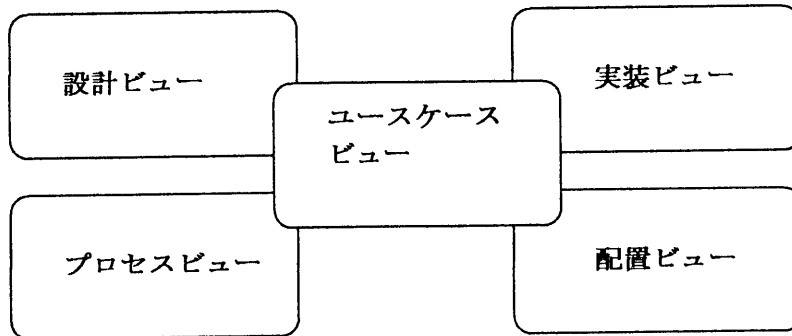


図 3-13 UML の 5 つのビュー

図 3-13 は、UML の 5 つのビュー関係を示しているが、それぞれについて下記に詳細を示す。

- 1) ユースケースビュー
- 2) 設計ビュー
- 3) 実装ビュー
- 4) プロセスビュー
- 5) 配置ビュー

通常、システムの構築には、多くの人がその開発に関わることになる。エンドユーザ、アナリスト、プログラマ、システムインテグレーター、テスター、テクニカルライターなど役割も様々で、役割が変われば見る観点や視点が変わる。これらの人々がそれぞれ欲しい情報を得るため、システムを様々な視点からモデリングして描く必要がある。

ユースケースビューは、エンドユーザやアナリストからみたシステムの振る舞いを記述する。静的には、ユースケース図、動的には、相互作用図、状態遷移図、アクティビティ図を使用する。設計ビューは、システムの機能的要求やエンドユーザへのサービスを記述する。静的には、クラス図、オブジェクト図、動的には、相互作用図、状態遷移図、アクティビティ図を使用する。プロセスビューは、システムの並列性や同期、スレッド、プロセスを記述する。実装ビューは、実際のシステム稼動に関する情報を記述する。静的には、コンポーネント図、動的には、相互作用図、状態遷移図、アクティビティ図を使用する。配置ビューは、システムの分散、配布、インストールを記述する。静的には、配置図、動的には、相互作用図、状態遷移図、アクティビティ図を使用する。

<補足：ビジネスオブジェクトモデル>

ビジネスオブジェクトモデルは、ビジネス構造モデルとビジネス振る舞いモデルの二つより成り立つ。その中心概念がビジネスオブジェクトである。ビジネスオブジェクトは、企業情報システムにおけるアプリケーションプログラムの開發生産性を飛躍的に向上するアプローチがある。

1) ビジネスオブジェクト誕生の背景

企業情報システムは、それぞれの企業の業種、業態、組織、歴史、経営方針などに深く依存しており、それぞれ異なるものである。そのため、業務アプリケーションシステム及びそのプログラムは、企業ごとのカスタムメイドが多かった。こうした中で OMG (Object Management Group) が 1989年に設立された。情報システムのプログラム開發生産性を向上させる目的で企業や研究機関などを会員とする協議会である。「いかにしてそのシステム固有に開発する部分を減らせるか」という観点から要求仕様を分析し、部品化、再利用を図るための基盤を整備し始めた。つ

まり、再利用可能な部品（コンポーネント）を揃えることで、ゼロからの情報システム構築ではなく、部品（コンポーネント）の組み合わせによる情報システム構築を実現することがビジネスオブジェクトの狙いである[49]。

2) ビジネスオブジェクトの定義 OMG 1996年の定義

ビジネスオブジェクトとは、ビジネス分野で活発な物を表現するもので少なくともビジネス上の名前、定義、属性、振る舞い、関連、ルール、ポリシーそして制約を含む。例えば、それは、人、場所、イベント、ビジネスプロセスあるいは、概念を表現する[60]。

今城らの定義は、ビジネスオブジェクトとは、業務上意味ある概念をオブジェクトとして表現したものである。つまり、分析、設計時においてオブジェクトモデルの一要素であり、プログラミング時にはプログラムの一単位であり、実行、運用時には動いているソフトウェアのある単位となるものである。

ビジネスオブジェクトを3つの種類に分類している。それぞれの内容を以下に示す。

(1) エンティティ ビジネスオブジェクト

人、場所、もの、あるいは概念を表すビジネスオブジェクト。これらはビジネスプロセスの遂行に欠かせないビジネス上の基本的な概念である。

具体例：場所（住所、電話番号、電子メールアドレス、地方など）、製品（商品、サービス、部品など）、会計手段（貨幣）

(2) プロセス ビジネスオブジェクト

ビジネスプロセスやワークフローを表すビジネスオブジェクトで、複数のエンティティビジネスオブジェクトとそれらのインスタンス間の相互作用やビジネスイベントのパターンからなる。エンティティビジネスオブジェクトのインスタンスのパターンからなる。

具体例：受注処理、調達処理、請求・入金処理、イベント ビジネスオブジェクト

ビジネスイベントを表すビジネスオブジェクトで、それは時間軸上で表し、ある時間経過することも、また、ある時間的な境界をまたぐこともある。

具体例：タンク内の気圧が限界を超えた。会計年度が終了した。在庫量が発注点を下回った。イベントビジネスオブジェクトは、プロセスビジネスオブジェクト間の相互作用を引き起こし、その結果として別のイベントが起こることもある。

2-2 現行UMLダイアグラムの特徴分析

従来のUMLを使用するビジネスモデリングの対象は、ビジネスプロセスモデルであり、本論文で論じているビジネスモデルではない。この前提に立ってその設計開発の手順を分析する。設計開発フェーズは、オブジェクト指向分析とオブジェクト指向設計に分かれる。それぞれのフェーズにおいて使用するUMLダイアグラムを整理すると下記のようなになる[53][54][55][56]。

オブジェクト指向分析は、形ある物をオブジェクト分析して、実世界のシステムを目に見える形にモデル化することであり、クラスを作成することである。オブジェクト指向設計は、整理したクラスを使用してクラス間の関係を明示し、設計する。下記にその手順と使用するUMLダイアグラムを示す。

- オブジェクト指向分析におけるUMLダイアグラムを使う手順

- 1) ユースケース図の作成 (システムの要求機能とシステム外との境界を明確にする)
- 2) アクティビティ図の作成 (ユースケースの詳細記述図を表す)
- 3) シーケンス図の作成 (オブジェクトどおしの相互作用を見る)
- 4) オブジェクト図・クラス図の作成 (クラスを抽出し、クラス間の関係をみる)
- 5) シーケンス図・クラス図の作成 (モデルを詳細化する)
- 6) ステートチャート図 (オブジェクトの状態変化をみる)

- オブジェクト指向設計におけるUMLダイアグラムを使う手順

- 1) シーケンス図・クラス図を作成 (画面や外部システムとのインターフェースとなるクラスを検討する)
- 2) コラボレーション図を作成 (オブジェクト間の関係と相互作用について見る)
- 3) シーケンス図・クラス図を作成 (属性や操作や関係を詳細化する)
- 4) パッケージ図の作成 (モデルをパッケージに分割する)
- 5) コンポーネント図作成 (ソフトウェア構成を考える)
- 6) 配置図を作成 (ハードウェアも含めたシステム構成を表す)

次に各UMLダイアグラムの概要と表記について記述する。

2-2-1 クラス図

クラス図は、システムとドメインにおける事物、及びそれらが互いにどのように関係しあうかを示す。クラス図は、論理的なビューを表現するために用いられるダイアグラムであり、図に示すように、クラスの属性、振る舞いそしてクラス間の静的な側面を表現している。

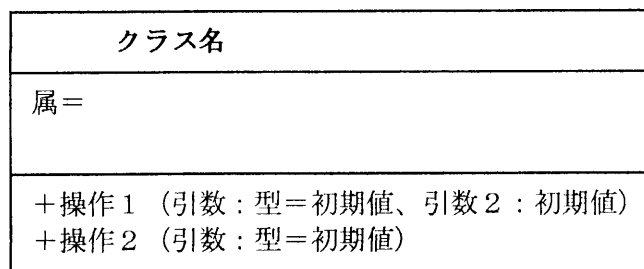


図3-14 クラスの記述

図3-14は、クラス図自体の記述例である。このようにクラス図は、上段にクラス名、中段に

クラスを持つ属性、下段にクラスを持つ操作を記述できるようになっている。その属性には、可視性、型、初期値を設定できる。操作にも、同様のプロパティを設定できる。また、クラス自体の詳細を記述するのではなく、図3-17に示すようにクラス間の関連を記述することもできる。

図3-15にある「ルール1」、「ルール2」、「1」、「*」「限定子」といったクラス間の関係を示す文字や記号が使用される。関連ルールは、クラス間の役割を意味する。クラス1から見たクラス2の役割は、ルール2で示し、クラス2から見たクラス1の役割は、ルール1で示す。また、「1」、「*」は、対応するクラスの数を表す。例えば、クラス2のオブジェクトから見てクラス1のオブジェクトは、1つしかないことを示し、クラス1のオブジェクトから見てクラス2のオブジェクトは、複数対応することを示す。また、「限定子」を使用してクラス間の関連に制約を設定できる。クラス図はオブジェクト指向における論理的最終設計成果とも言える存在で、そのシステムの用途における適用性の寿命を左右する重要な役割と考えている。

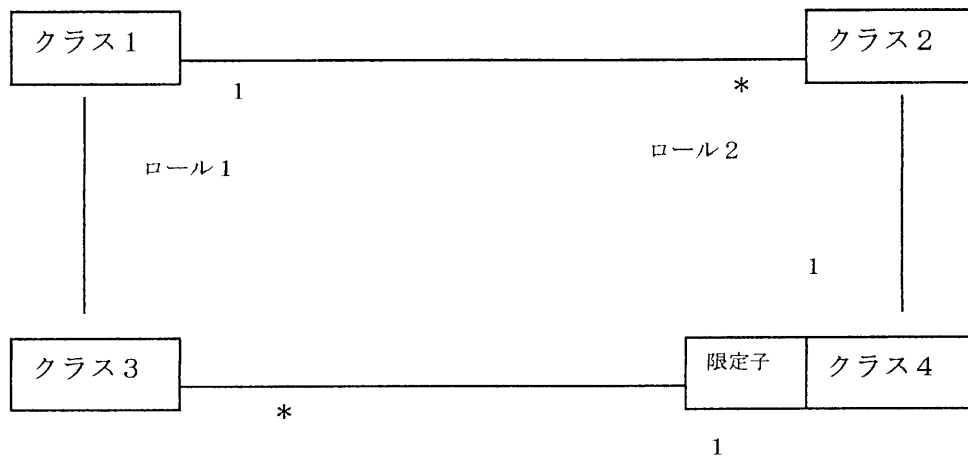


図3-15 クラス間の関連

クラス概念

オブジェクト指向では、抽象的に表現している「もの」をオブジェクト、オブジェクトの型をクラスと呼ぶ。クラスにはクラスの状態を表現する「属性」と、クラスの役割を表現する「振る舞い」を記述する。クラスは、属性と振る舞いを規定することにより、各人の認識を一致させ、共通認識を抽象的な表現で統一できることにある。

クラスは抽象的な表現であり、実体を持っていない。オブジェクト指向では、この抽象的なクラスをもとに実体化したものを、オブジェクトと呼ぶ。そして、クラスからオブジェクトを実体化することをインスタンス化と呼ぶ。言い換えるとクラスをインスタンス化したものがオブジェクトとなる。例として自動車でクラスをインスタンス化して示してみると図3-16のようになる。

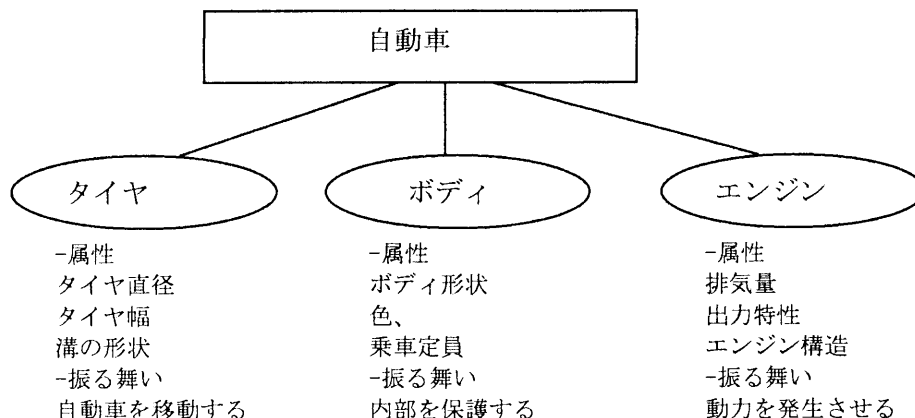


図3-16 インスタンス化したクラス

2-2-2 UML ダイアグラム

UML ダイアグラムは、8つないし9つのダイアグラムがある。その代表的なユースケース図は、開発するシステムとその外部との境界を明確化するためのダイアグラムで、狭義のUMLでは最も上流に位置する。ユースケース図では、開発対象システム名を持ったボックスに楕円形でそのシステムが果たす機能的塊であるユースケースを持っている。ボックスの外にはそのシステムのユースケース（すなわち機能）への起動や結果を利用するアクターを配置し、そのアクターとユースケース間で交信関係を矢印で結んで表現する。ユースケース図を作成するにはそのシステムを利用するユーザやシステムが果たす機能の塊が明確になっていることが求められる。

ユースケース図の構成要素は下図 3-17 に示すようにシステムの提供する機能を表すユースケースとそれらのユースケースを利用する外部の利用者を表すアクター、及びこれらの関連を示す関係表現から成り立つ。

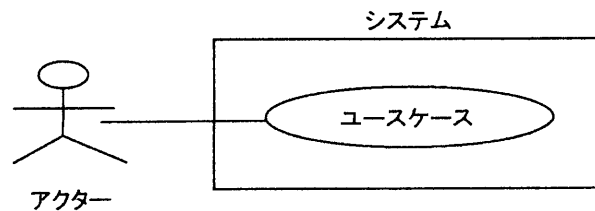


図 3-17 ユースケース図の基本

ユースケース図はそのシステムが外部からどのように使われどのような役割を果たすかを明確に表すことから、UML にとっては最も重要な図式表現の一つであり、階層化も可能である。そして個々のユースケースに対して下記の事項が明確になっていることが重要とされている。

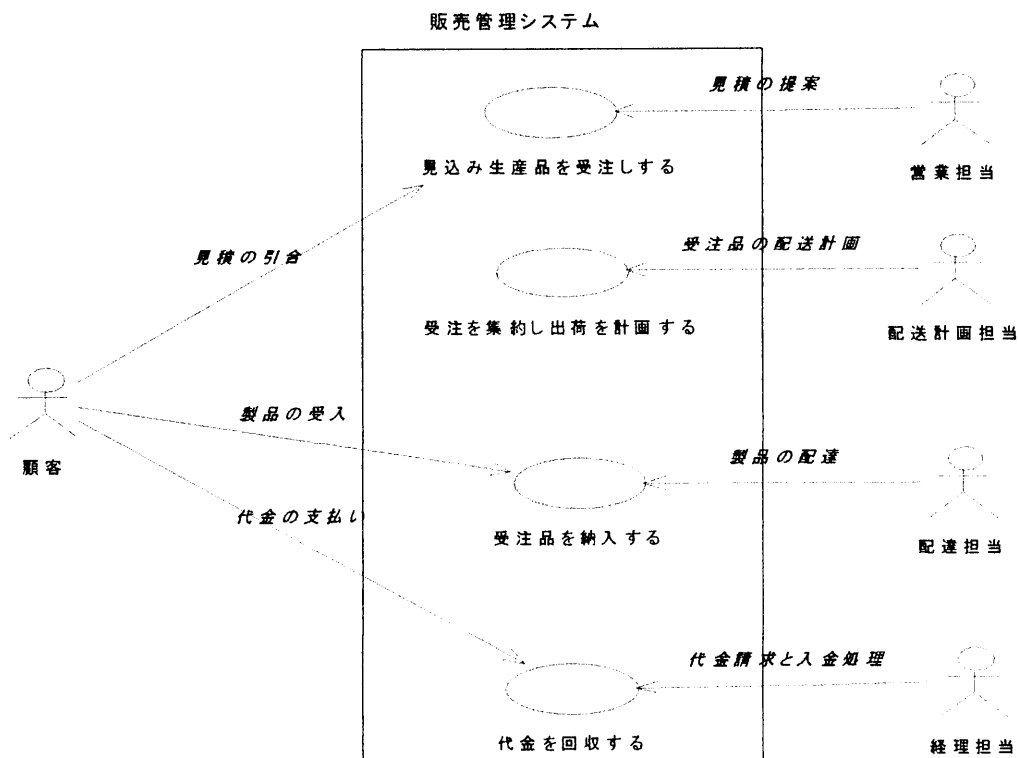


図 3-18 ユースケース図の例

図 3-18 に示すように、ユースケース図の特徴は、アクターによって利用者が明確になり、その機能に求められる目標が明らかになる。また、その機能に必要な入出力情報が明らかになる。その機能で果たすべき役割を明らかに示す。

ユースケースでは、ユースケース相互間で汎化や依存関係等の関係付けをして詳細化を行うことができる。ユースケースシナリオ、またはユースケースドキュメントによって記述が必要である。

従来から使用されている IDEF0 と UML ユースケースの比較すると IDEF0 によるアクティビティ・モデルと UML のユースケース図の対応関係を示している。IDEF0 のアクティビティで開発しようとするシステム名をメカニズムとして持っているアクティビティが UML のユースケースの候補になりえる。前節に UML のダイアグラムを考察したが、大きく分類してビジネスモデルを記述するために使用できる UML のダイアグラムと、情報システム構築に使用する UML ダイアグラムに分類できる。

次に、シーケンス図あるいはコラボレーション図はユースケース図で定義された機能の塊に対して、その動作を分析しクラス図設計へつなげて行く中間的なダイアグラムと位置付けている。シーケンス図はシステムを構成するクラスやアクター間でどのようなメッセージがどのような時間的順序でやり取りされるかを表現したもので、これによってクラスやアクターがそのメッセージにより何をしなければいけないかを明確にする。

シーケンス図では、オブジェクト間のメッセージのやりとりが時系列で表す。一方、コラボレーション図では、メッセージの時系列な流れはわかりにくい、どのオブジェクトとどのオブジェクトに関係があるかが理解しやすい。

コラボレーション図は、オブジェクトの相互作用を表す図の 1 つで、開発工程の「設計」段階で主に用いられる。その特徴を以下に示す。

- ・ シーケンス図と表裏一体
- ・ シーケンス図がオブジェクトの相互作用を時系列的に表現しているのに対し、コラボレーション図はオブジェクトをフラットに配置し、オブジェクト間の接続関係に着目して表現している。
- ・ シーケンス図と比較して表現できない内容は、オブジェクトの活性区間とライフラインである。
- ・ コラボレーション図にはアクターやインタフェースを含むこともできる。
- ・ メッセージのやりとりに相対的な順序番号を付すことで、オブジェクト間に流れるメッセージの順序を把握することができる。

コンポーネント図は、システムを構成する実行可能モジュールやソースコード (=コンポーネント) の物理的な構造 (依存関係) を表現するダイアグラムで、開発工程の「設計」段階で用いられる。

オブジェクト図は他のクラス図やユースケース図といった UML の図面全体の表現形式を規定しているものではない。UML では最終的にオブジェクト・クラスと呼ばれる対象物のタイプとそれらの関係を示すクラス図で対象物 (オブジェクト) の存在関係を表すことで無駄のないシステム記述を図ることを目指している。しかしながら最初から最も無駄のない対象物のタイプが描きだすことは一般に困難であり、その観点から実際の対象物がどのように振る舞いどのように他と関わりあうかを分析し、整理を繰り返すことにより、最もそのシステムの用途に最適なオブジェクト・タイプの定義や関係を練り上げていく必要がある。オブジェクト図とはこのような実際の対象物がどのように振る舞い他と関わりあうかを分析する過程で特定の対象物 (すなわちインスタンス) を例示して分析する際の対象物の表現手段として準備されている表記法である。対象物の一般化されたクラスはボックスの中にクラス名を記載して表されるのに対して、オブジェクト

図はボックスの中にインスタンス名とクラス名を：で区切りアンダーラインを引いて表される [57][58][59]。

配置図は、システムが実行される際の物理的な構成を表現するダイアグラムである。システムは、複数のノードとそれを結ぶ実線で構成される。ノードとは、物理的なハードウェアを表し、直方体のアイコンで記述される。ノードには、ソフトウェア・コンポーネントを実行できるプロセッサ（サーバや PC）と実行できないデバイス（ハブやプリンター）の2種類がある [60][61]。

アクティビティ図は、処理やプロセスでの手順や条件判断、分岐などを表示するためのダイアグラムである。主な役割と目的は、以下のとおりである。

- ・システムの動的な側面をモデリング
- ・業務や処理の流れ（ワークフロー）を表す
- ・フローチャートに類似（手順、条件判断、分岐の図式的表現）

記述方法としては、以下の特徴がある。

これまでのクラス図やオブジェクト図、ユースケース図がシステムの振る舞いやひとまとまりのクラス、オブジェクト、ユースケースをモデル化するのに対し、ステートチャート図は、1つのオブジェクトに着目してその状態の変化をモデル化するものである。したがって、ステートチャート図で描かれた UML モデルは、オブジェクトが時間の経過とともに変化する様子を表現することが可能である [62][63][64][65]。

クラス図やオブジェクト図では、システムの階層構造・関連性・振る舞いの内容など静的に表現することはできても、システムの振る舞いを動的に表現することができなかったのに対し、ステートチャート図は、オブジェクトの振る舞いを動的に表現することができる。オブジェクトがとるべき振る舞いを特に把握しておかなければならない開発者にとって、その振る舞いを明示することができるツールとしてステートチャート図は価値がある。ステートチャート図を利用すれば、開発チームが要求に合致したシステムを完成できる可能性が高まるといえる。このように、ステートチャート図は、分析者・設計者から開発者への意思疎通の手段として、重要な役割を担っている。表記法としては、角に丸みのある長方形のボックスが状態(state)、ボックスとボックスをつなぐ実線の矢印が状態遷移(transition)を表す。矢印の向きは、状態の遷移の方向を表す。内側を塗りつぶした丸型のアイコンは開始状態(starting point)、二重の円の内側だけを塗りつぶした形のアイコンは終了状態(endpoint)を表す [66][67][68][69][70]。

以上 UML ダイアグラムについて調査分析したが、これら結果から次節で評価した。
また、UML ダイアグラムを記述するツールとして、本研究では、下記 2 ツールを使用した。

補足 1：表記利用ソフト：Rational Rose、Rose Modeler Edition Release Version 2003.06.

補足 2：表記利用ソフト Konesa 株式会社 オージス総研

2-3 現行UMLの評価と考察

前節において、UMLでの表記について個別に分析してきたが、これらの使用目的と使用するダイアグラムを下表3-2にまとめた。表3-2では、オブジェクト指向のシステム構築手順に沿ったフェーズごとに、その目的と使用するUMLダイアグラムとその概要を記述する。

表 3-2 開発フェーズとUMLダイアグラムの関係

フェーズ名	目的	使用するUML図	概要
ビジネス分析	ビジネスの理解、 ビジネスの視覚化	アクティビティ図、 クラス図、シーケンス図	ビジネスを視覚化し、ユーザ、開発者間で共有、理解する。そのうえで改善点やシステム化対象を洗い出す
要求分析	要求の把握（ユースケース=システム利用例の視覚化）	ユースケース図、ユースケース記述	システム化対象の要求についてユーザの視点でユースケースモデルを作成することで明確化する
システム分析	要求の概念構造の理解、視覚化	クラス図、オブジェクト図、相互作用図、パッケージ図	要求の中に現れる用語を利用して、要求の意味、概念構造をとらえるためにクラス図を作成する
システム設計	ソフトウェア構造の理解、視覚化	クラス図、オブジェクト図、状態図、相互作用図、パッケージ図、コンポーネント図、配置図	実装されるアーキテクチャを含んだソフトウェア構造をモデル化
実装・テスト	ソフトウェアの実現	なし	設計に合わせて実装を行い、単体、複合テストを行う

※ シーケンス図は、相互作用図の1つである

表3-2は、情報システム開発のフェーズ毎に利用されるUMLダイアログとその概要を記述した。ビジネス分析のフェーズにアクティビティ図やシーケンス図が利用されることが多い。

2-3-1 現行UMLの長所

本研究では、ARIS、PDR、IDEFといったビジネスプロセスを表記するツールを比較研究してきた。その結果、オブジェクト指向に対応しており、標準化された記述が確立されつつあるUMLをビジネスモデル表記方法として選択した。

UMLを選択した理由は、以下の3点である。

- 1) オブジェクト指向対応と後工程にシームレスな繋がりが可能
- 2) 統一ルールにより属人的表記を排除
- 3) 複数のモデリングの表記が可能

上記の理由についての具体的な例を含めて説明する。

1) オブジェクト指向対応による後工程にシームレスな連携の検証

UMLは、オブジェクト指向言語であり、UMLでは、クラス、オブジェクトなどを作成し、UMLダイアグラムとプログラムのソースコード作成まで繋がる仕組みが出来ている。業務分析、設計により、UML表記をしたダイアグラムがそのまま、プログラム作成に繋がる具体的な事例を以下に示す。

図3-19に、チケット予約をするシステム事例を用い、要件からモデル、プログラム実装までの一連の繋がりを示す振る舞い分析モデルを表記する。このチケット予約システムの概要は、下記のaからgまでの手順で行う。

- a. 会員は、メニューから「チケットを予約する」を選択する。
- b. システムは、予約可能なチケットをその在庫情報とともに一覧表示する。
- c. 会員は、一覧の中から予約を希望するチケットを選択し、予約枚数を入力する。
- d. システムは、入力された枚数だけ在庫があるかどうかをチェックし、在庫があれば、そのチケットの予約を受け付け、在庫数を更新する。
- e. その結果、予約受付完了のメッセージが表示される。このメッセージには、予約受付番号が含まれている。
- f. 顧客は、チケット購入時に提示するために、予約受付番号を控える。
- g. 一覧の中に他に予約したいチケットがある場合、c～fを繰り返す。

上記の業務内容をシステム化する手段としてUMLを使用する場合、ユースケースからプログラムコードまでが、順次①から④を使用してシームレスに繋がることを示す。

- ①振る舞いを示す分析モデル (図3-19)
- ②構造を示す分析モデル (図3-20 上)
- ③構造を示す設計モデルが作成される。(図3-21 下)
- ④プログラム実装ができる。(図3-22)

以下にその概要を図示する。

① 振る舞いを示す分析モデル

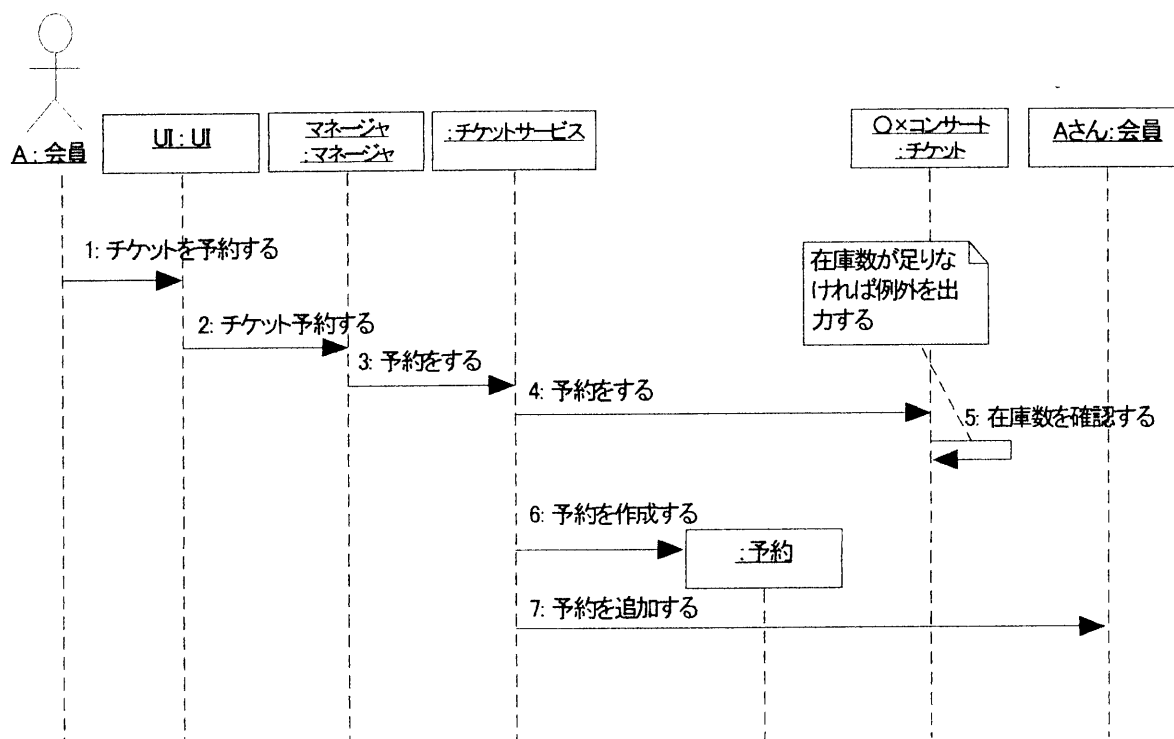
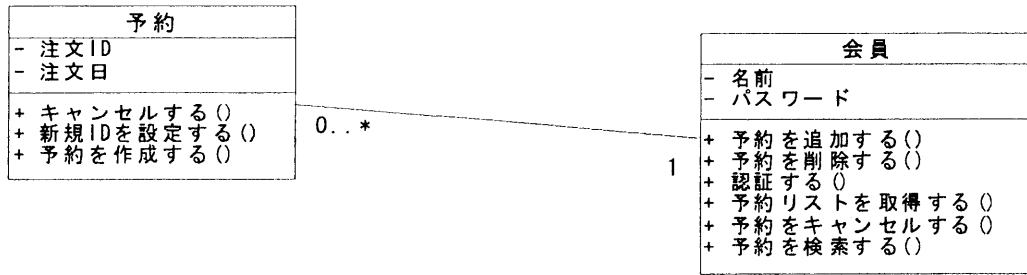


図 3-19 振る舞いを示す分析モデル

図3-19は、シーケンス図を使って会員のチケット予約を行うビジネスプロセスを振る舞い分析モデルとして表記した。

② 構造を示す分析モデル



③ 構造を示す設計モデル

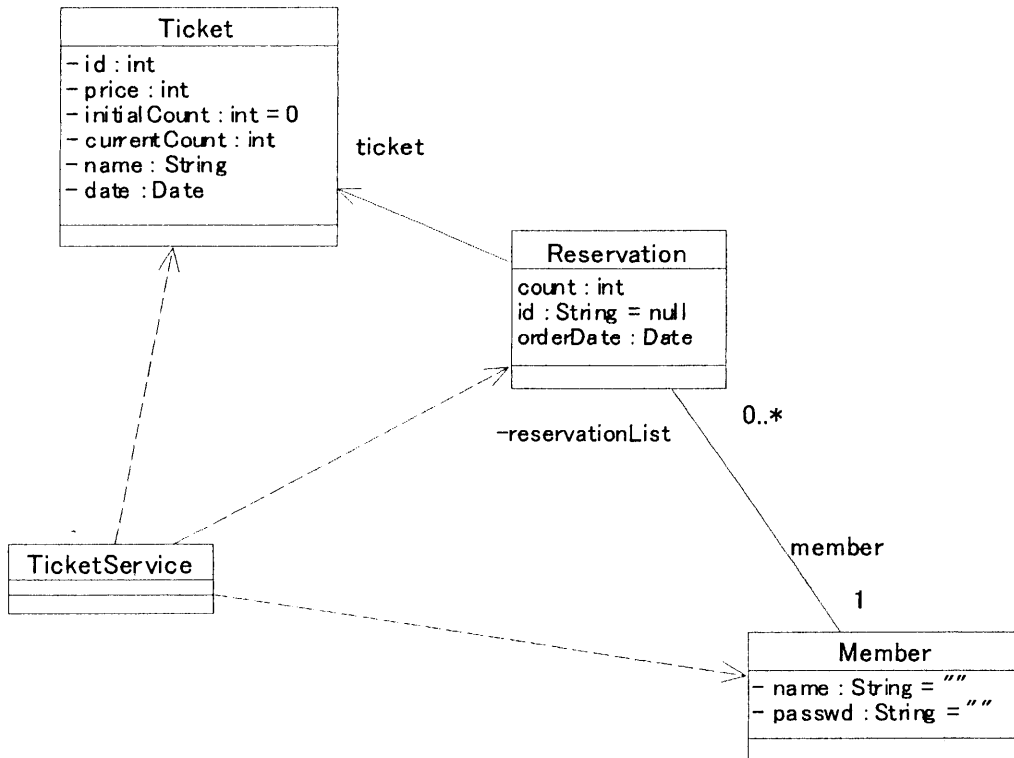


図 3-20 構造を示す分析モデルと構造モデル

図 3-20 は、クラス図を使用して構造を示す分析モデルと構造を示す構造モデルを作成した。クラス図の中段には、そのクラスの属性や初期値を記入する。

④ JAVA言語実装例

構造を示す分析モデルと構造モデル設計モデルからプログラム実装ができることを下記の事例が示している。

```

package ticket.domain;
import java.util.*;
import java.io.*;
import java.text.*;
/**
 * 会員クラス
 */
public class Member implements Serializable {

```



```

    private String name="";
    private String passwd="";
    private List reservationList= new ArrayList();
    /**
     * コンストラクタ
     */
    public Member(String name、 String passwd) {
        this.name=name;
        this.passwd=passwd;
    }
    public String getName() {return name;}
    public String getPasswd() {return passwd;}

    /**
     * パスワードをチェックする
     */
    public boolean verify(String pwd) {
        return passwd.equals(pwd);
    }
    public List getReservationList() {
        return reservationList;
    }

    public Reservation getReservationByID(String id) {
        Iterator it = reservationList.iterator();
        while(it.hasNext()) {
            Reservation anReservation = (Reservation)it.next();
            if(anReservation.getID().equals(id)){return anReservation;}
        }
        return null;
    }
}

/**
 * 予約オブジェクトの追加
 */
public void add(Reservation reservation) {
    this.reservationList.add(reservation);
}

/**
 * 予約オブジェクトの削除
 */
public void remove(Reservation reservation) {
    this.reservationList.remove(reservation);
}

/**
 * 会員オブジェクトの文字列表現
 */
public String toString() {
    StringBuffer buf = new StringBuffer();
    buf.append("名前:" +name +"¥n");
    return new String(buf);
}
}

```

図 3-21 JAVA 言語実装例

以上がUMLによるオブジェクト指向の特徴を生かした一連のシステム構築の流れを示した事例であり、ユースケースからプログラムコードまでが、順次①から④を使用してシームレスに繋がることを示した。図3-19の振る舞いを示す分析モデルから図3-20の構造を示す設計モデルを経て図3-21に示すJAVAのプログラムコードが作成されることを表わしている。このようにシステムの上流である業務分析、設計からUML表記をしたダイアグラムがそのまま、プログラム作成に繋がる。

2) 統一ルールにより属人的表記を排除

従来のシステム開発では、ドキュメントを文章で記述し、組織やプロジェクトごとに表記のルールが異なるなど、仕様書や設計書などのドキュメントが文章や独自のルールにもとづいた記述になっていた。この文章による属人的表記やプロジェクトごとに記述が異なるドキュメントは、開発者間で誤解が生じて要件が正確に伝わらず、書き手の意図を把握するのに時間がかかることになる。担当者によって品質のばらつきが出やすく、網羅性にも欠けやすい。また、開発工程間をまたがるドキュメントの対応付けが明確に定義されていないことが多く、人の理解や記憶に頼って対応付けを判断する必要があるため、開発中に生じた変更を関連するドキュメントに反映することは難しい。この結果、仕様を記述したドキュメントと実際に動いているプログラムの中身が乖離してしまい、ドキュメントの信頼性がなくなり、そのうちメンテナンスもされなくなる。このようなドキュメントの不備はシステムの品質にも影響することになる。ビジネスモデリングでは、ビジネスモデルを継続的に改善していくためにモデル化の作業を行う。独自の表記で作成したビジネスモデルは属人性が高くツールベンダに依存するため、1サイクル回した後にモデルを引き継げなくなり、継続時にはゼロからモデルを再構築する必要が出てくる。

3) 複数のモデリングの表記が可能

モデリングをするのは、システムが大規模になるにしたがって、システムの全体像が見えなかったり、開発者間で共通のシステム認識が持てなかったりするなど、担当部分を理解することも困難になってくるためである。担当の異動や引継ぎによる仕様書や設計書の見直しの工数も大きくなると言える。システム開発とビジネスモデリングにおいて、モデリングの作業自体は、自分の頭の中にある考えを図示することなので直感的に理解しやすく考えを整理する一手段と言える。UMLにより、ユーザの意図や考えを正確に把握するために、複数の図示したモデル（ダイアグラム）をもとに議論することが可能となる。また、モデリングは文章などに比べて曖昧さを残しにくいいため、お互いの考えについてできるだけ誤解を少なくできる有効な手段と言える。

最後に企業を取り巻く環境がグローバル化しており、こうした中で、国際標準の表記法に準拠する重要性が増している。UMLは、1997年にOMGが標準として採択し、システム開発における事実上の標準として急速に普及している。また、経営戦略、ビジネスプロセス、情報システムを一貫して表現するモデリング言語としても国際標準（ISO/IEC 19501）となってきた。UMLを使用すれば、国際標準表記で海外も含めた広範囲なコミュニケーションが可能となり、こうした意味でも、この研究の意義が高まることを望みたい。

2-3-2 UMLによるビジネスモデル表記の従来研究と本研究比較

従来研究では、企業経営の視点からのアプローチとシステムモデリングからのアプローチによる2つの視点からの研究が行われていた。

1) システムモデリングの視点からビジネスモデルの要素

オブジェクト指向のシステム開発方法のなかで統一モデリング言語であるUMLをビジネスモデリングに適用とする試みがなされている。

UMLはシステムモデリングを目的とした言語であるが、特定の領域やプロセスに合わせて特化させるため、UMLプロファイルが準備されている。UMLプロファイルは特定の領域において、UMLのモデルを作成するための拡張された方法を提供する。UMLプロファイルは要素や属性・メソッド・リンクなどに適用され、ステレオタイプとタグ付き値で構成されている。UML仕様書では、ビジネスモデリングのためのUMLプロファイルが定義されているが、ビジネスプロセス（業務フロー）の記述が中心となり、企業の組織階層やビジネスルール、ビジネスゴールなどは含まれていない。

このUMLプロファイルを利用したビジネスモデリングへのUMLの適用方法従来研究として、Erikson&Penkerによる「UMLによるビジネスモデリング」[74]がある。

彼らは、モデル化の対象として、「リソース」「プロセス」「ゴール」「ルール」が必要であると考え、そのモデル化の視点として、次の4つのビューを提唱している。

- ・ビジネスビジョンビュー
そのビジネスのビジョンとゴールを示し、ゴール達成に解決すべき問題を図示している。
- ・ビジネスプロセスビュー
そのプロセスで作られる価値とゴールを達成するためのプロセスとリソース間の相互作用を図示する。
- ・ビジネス構造ビュー
ビジネスにおける組織と生み出される製品の構造、リソース間の構造を示す。
- ・ビジネス振舞いビュー
そのビジネスモデルにおける各リソースやプロセスの個々の振舞いを示す。

また、マーシャルは、「UMLによるビジネス分析と情報システムの設計」[86]において、ビジネスの基本構造を4つのビジネス要素（目的、エンティティ、ビジネスプロセス、オーガニゼーション）で表している。特徴的なものとして、エンティティには、「顧客」、「従業員」、「装置」など活動主体となるものと、「製品」や「サービス」、「資産」などビジネスの主体となるものがあるとし、その中味をモデルとして表記し、ビジネスオブジェクトとしてコンポーネント化している。

2) 企業経営の視点におけるビジネスシステムの研究（加護野、伊丹らの研究）

加護野、伊丹らによる「ゼミナール経営学入門」では、顧客に製品やサービスを届け、価値を提供するしくみ全体を「ビジネスシステム」と呼んでいる[141]。

このビジネスシステムの設計のために必要な要素として次の3つをあげている。

- ・分業関係の構造の決定
- ・情報、モノ、カネの流れの仕組みの設計
- ・システム全体の情報、モノ、カネの流れの調整と規律のメカニズムのくふう

この3つの要素については、「インタンジブル・アセット」という呼ばれ方もされている。このビジネスシステムの概念を、小川[142]は、以下の図に示すように、事業概念、業務プロセス、組織、資源、製品やサービス、顧客との情報作用、ケイパビリティの7つの要素で整理している。また、藤本・武石・青島ら[143]は、著書「ビジネス・アーキテクチャ」において次のような定義を提案している。

ビジネス・プロセスとは、製品やサービスが開発され、生産され、必要とするユーザに販売されユーザの使用をサポートするまでの一連のプロセスをいう。

ビジネスの構造は、ビジネス・プロセスの中のどこで付加価値が生み出されているのかを示す価

値創造活動の配分パターンをいう。

ビジネス・モデルは、ビジネスの構造に対する理解を前提にした、自社独自の付加価値創出のパターンをいう。

ビジネス・アーキテクチャは、ビジネス・プロセスの中にあるさまざまな活動要素間の相互依存性もしくは関係性のあり方を示す。

このビジネスシステム構築のインプットとなるのが、競争ドメインである。すなわち、

- a) どのような製品・市場のセグメントを事業対象とするか
- b) その製品・市場でどのような差別化をするか
- c) その差別化のために、どのようなビジネスシステムを構築するか

というようにビジネスシステムの要件が決定される[14]。以上、経営学の視点とモデリングの視点各々からビジネスモデルの要素をレビューしたが、両者の内容を比較整理すると次の表 3-6 になる。

以上の従来研究を本研究での提案と比較するため、ビジネスモデルの構成要素（ビジネスモデル表記の種類）とビジネスプロセスのUMLによる表記方法の観点で比較表(表 3-3)を作成した。

表 3-3 ビジネスモデル表記の従来研究と本研究の比較

	ビジネスモデルの構成要素			UMLによるビジネスプロセス表記			
	加藤予・伊丹	小川	藤本・武石・青島	本論文	Eriksson & Perker	Marshall	本論文
経営学入門	事業とビジネスシステム	ビジネス・アーキテクチャ	ビジネスモデル	UMLによるビジネスモデリング	UMLによるビジネス分析と構想システムの記述	ビジネスプロセスモデル	
ビジネスモデリング	a) どのような製品・市場のセグメントを事業対象とするか	事業概念					
	b) その製品・市場でどのような差別化をするか		ビジネス・モデル	(3) バリューチェーンモデルによるコア・コンピタンスモデル	ビジネスビジョンビュー		目標ビュー 戦略マップ
	c) その差別化のために、どのようなビジネスシステムを構築するか		ビジネスの構造	(5) 財務モデル (収支別) (1) 企業間ビジネスモデル			
	○ 関係性の構造の決定		ビジネス・アーキテクチャ				
	○ 情報 モノ、カネの流れの仕組みの設計	業務プロセス	ビジネス・プロセス	(2) 企業内ビジネスモデル	ビジネスプロセスビュー ビジネス振舞いビュー	ビジネスプロセス	プロセスビュー ・ビジネスプロセスモデル ・ビジネスユースケースモデル ワークフロー ・ビジネスオブジェクトモデル ビジネス構造モデル ビジネス振舞いモデル
		顧客との相互作用		(9) プロセスレベルの記述	ルール		
		製品やサービス		(8) 販売方法と資金回収モデル	ビジネス振舞いビュー		
		組織		(6) 製品構成と価格モデル	ビジネス構造ビュー	エンティティ 製品 オーガニゼーション	組織ビュー
		資源		(7) 組織と人材モデル		エンティティ (主体)	
		○ システム全体の情報 モノ、カネの流れの調整と規程のメカニズムのくふう	ケイパビリティ	(4) サイクルタイムモデル	リソース ゴール	エンティティ (主体) 目的	ネットワークビュー 目標ビュー スコアカード

表 3-3 には、ビジネスモデルの構成要素（ビジネスモデル表記の種類）とビジネスプロセスのUMLによる表記方法について比較検討した結果をまとめた。本研究では、ビジネスモデルを明確に可視化するには、複数の視点によりビジネスモデルを表記する必要があると考えた。また、ビジネスモデルをUMLで表記するための表記方法の提案を行った。

表 3-6 のとおり、本研究では、ビジネスモデルの構成要素として、ビジネス構想レベルで3つのモデルに区分し、ビジネスモデリングレベルでは、6つのモデルに区分することにより、従来

研究では、不十分であったビジネスモデルを可視化する視点別のモデル化が可能になる。また、ビジネスモデル表記では、ビジネスプロセスの表記が中心であった従来研究に追加して、ビジネス構想レベルを強化し、目標ビューと戦略マップを表記することにより、ビジネスモデルとの強い連携が可能となった。本研究の特色は、従来研究に比較し、ビジネス構想とビジネスモデリングの両領域をカバーしており、ビジネスモデルの構成要素を視点別に考慮したモデルを提案している。また、ビジネスプロセス表記においては、4つのビュー（目標ビュー、プロセスビュー、組織ビュー、ネットワークビュー）とプロセスビューの表記方法を提案する。

2-4 UMLによるビジネスモデリングの課題

UMLはもともとビジネスプロセスを表記とシステムモデルを表記する言語として開発されてきた。従って、UMLによるビジネスモデリングは、システム構築を前提にそれに必要なシステムモデリングをする情報の収集を目的としている場合が多い。したがって、本論文で定義したビジネスモデルを表記する機能は弱いという問題がある。具体的な課題としては、以下がある。

課題1. ビジネスモデルを可視化するために複数の視点、切り口での表記種類がない。

このビジネスモデルを可視化するために複数の視点、切り口についての考え方は、従来研究にはないものであり、本論文の次節以降で詳しく述べる。

本研究では、次節でビジネスの実態をビジネスモデルで表記するためには、ビジネスモデルの内容が理解できる表現が必要と考え10種類のビジネスモデルが必要であることを提案する。また、情報システム開発にシームレスに繋げるためには、UMLによる表記が必要という結論に至った。しかしながら、現行のUML及びUMLビジネスモデリングでは、十分なビジネスモデル設計のツールとはなりえない。本論文では、次章においてUML表記の新提案を行うことにより、ビジネスモデルから情報システム構築へ効率的に繋がる方法を提案する。

課題2. UML表記方法にビジネスモデルを表すために十分な表記機能がない。

現行のUMLでは、特定の領域やプロセスに合わせて特化させるため、UMLプロファイルというものが準備されている。UML仕様書では、UMLをビジネスモデル領域に適用するため、ビジネスモデリングのためのUMLプロファイルが定義されているが、ビジネスプロセス（業務フロー）の記述が中心となり、企業の組織階層やビジネスルール、ビジネスゴールなどは含まれていない。

同じくUMLプロファイルを利用したビジネスモデリングへのUMLの適用方法の提案例として、Erikson and Penkerによる「UMLによるビジネスモデリング」がある。

彼らは、モデル化の対象として、「リソース」「プロセス」「ゴール」「ルール」が必要であると考え、そのモデル化の視点として、次の4つのビューを提唱している。

(1) ビジネスビジョンビュー

そのビジネスのビジョンとゴールを示し、ゴール達成に解決すべき問題を図示している。

(2) ビジネスプロセスビュー

そのプロセスで作られる価値とゴールを達成するためのプロセスとリソース間の相互作用を図示する。

(3) ビジネス構造ビュー

ビジネスにおける組織と生み出される製品の構造、リソース間の構造を示す。

(4) ビジネス振る舞いビュー

そのビジネスモデルにおける各リソースやプロセスの個々の振る舞いを示す。

Erikson and Penkerの4ビューでの表記の欠点として、第1に、ビジネスモデルの認識が不足しており、ビジネスプロセスと区別がないため、ビジネス全体を俯瞰する視点がない。第2に、ビジョンビューはあるが、単に目標を記述するだけで、それが、他のビューやビジネスプロセス

に関連付けができていない。つまり、基本的な構成は、ビジネスプロセスビューとビジネス構造ビューとビジネス振る舞いビューの3つから構成されており、ビジネスプロセスモデリングの域を出ていない。

この他、システム開発方法論として広く採用されつつある RUP (Rational Unified Process) では、開発プロセスの第一フェーズとして、「ビジネスモデリング」のフェーズを位置づけ、そこで明らかにするユースケースを「ビジネスユースケース」として、システムモデリングとは異なるモデル化を提案している。

また、マーシャルは、ビジネスの基本構造を4つのビジネス要素（目的、エンティティ、ビジネスプロセス、オーガニゼーション）で表す。特徴は、企業活動で登場する「製品」や「サービス」をエンティティ（実体）とし、その中味をモデルとして表記し、ビジネスオブジェクトとしてコンポーネント化している。

以上に示したように、エリクソンとパーカー及びマーシャルも、ビジネスプロセス（業務フロー）の記述が中心となり、企業の組織階層やビジネスルール、ビジネスゴールなどは含まれておらず、ビジネスモデル表記には、十分ではない。

本論文では、上記の課題を解決する提案を次節で行う。

第3節 UMLによる新表記法の提案

本節では、前節で取り上げたUMLによるビジネスモデル表記に関する課題の解決策として、UMLの拡張表記法を提案する。UMLは、従来、システムモデルを表記する統一言語として開発されてきた経緯があり、ビジネスモデルを表記するには、十分な内容を示す表現力と機能が不足している。本研究では、大きく分けて2つの提案を行う。第1は、ビジネスモデルを明確に可視化するための複数の視点とその表記法の提案である。第2は、UMLの標準表記に追加すべき、新たな拡張UML表記の提案である。

従来のUMLビジネスモデリングは、ビジネスプロセスモデリングの表記が中心であり、ビジネス構想やビジネスモデルを明確に示すための表記法が不足している。本研究では、従来の手法では、不足しているビジネスモデルを表すための8視点の表記方法を考案し、表記法を提案する。

課題1：ビジネスモデルを可視化するために複数の視点による表記方法がない。

この課題は、ビジネスの実態をビジネスモデルで表記する場合に、複数の視点や切り口が必要ということから起因している。言い換えると、ビジネスの実態が把握できるようなビジネスモデル表記にするには、ビジネスモデルをいくつかの視点、切り口により、表現する必要がある。ビジネスモデルの内容が理解できる表現方法を検討してみると表記法の前に、表記する対象や範囲によってビジネスモデル表記の種類があることに気づき、ビジネスモデルを可視化するための複数視点の表記法を提案する。

3-1 UMLによるビジネスモデル視点別表記法の提案

ビジネスの実態をビジネスモデルで表記するためには、ビジネスモデルの内容が容易に理解できる表記法が必要と考えた。また、検討の結果、表記する対象や範囲によってビジネスモデル表記の種類が複数必要であることに至っている。以下にビジネスモデル表記法の種類をあげる。ビジネスモデルを表記する場合、従来のUML表記だけでは、その表現能力は十分ではない。従来のUML研究は、情報システム構築を前提に業務の分析、設計を行い、ビジネスプロセスモデリングの表記が中心であったため、ビジネス構想やビジネスモデルを明確に示すための表記法が不足している。

第1章でビジネスモデルを可視化するための11種類の表記を提案した。しかしながら、11種類のビジネスモデルの視点のなかで サイクルタイムモデル、財務モデル、リスクモデルについては、UMLによる表記が適当ではないと判断し、対象から外した。この中で、UMLによるビジネスモデルの可視化に適した項目と財務諸表や表グラフの方が理解しやすい項目に分類し、UML表記をする8種類を選定した。

本研究では、従来の手法では不足している、ビジネスモデルを表現するための視点、切り口として第1章で述べた8種類の表記方法をサブモデルとして、表記する研究を行い、その表記法としてUMLによるビジネスモデル表記法を提案する。

ビジネスモデルの表記をUMLで8つのサブモデルに表記することを提案する。

表記の順番として対象範囲の広い順に記述した。

- 1) サプライチェーンビジネスモデル (企業間のビジネス関係を表す)
- 2) ビジネスプロセスモデル (自社全体のビジネスモデル概要と詳細を表す)
- 3) コア・コンピタンスモデル (その企業の核となる付加価値を生むプロセスを示す)
- 4) サイクルタイムモデル (自社のビジネスサイクルタイムを示す)
- 5) 製品構成と価格モデル
- 6) 組織と人材モデル
- 7) 販売と資金回収モデル
- 8) プロセスルールの記述 (ビジネスプロセスの補足)

以上が、ビジネスモデルを明確に示すための視点別モデルの種類である。

8つのサブモデルの関係を図示すると図3-22の通りとなる。

全体を横軸に通るサプライチェーンモデルと縦に組織と人材モデルがあり、中心にビジネスプロセスモデルがあり、その中にコア・コンピタンスモデルがある。

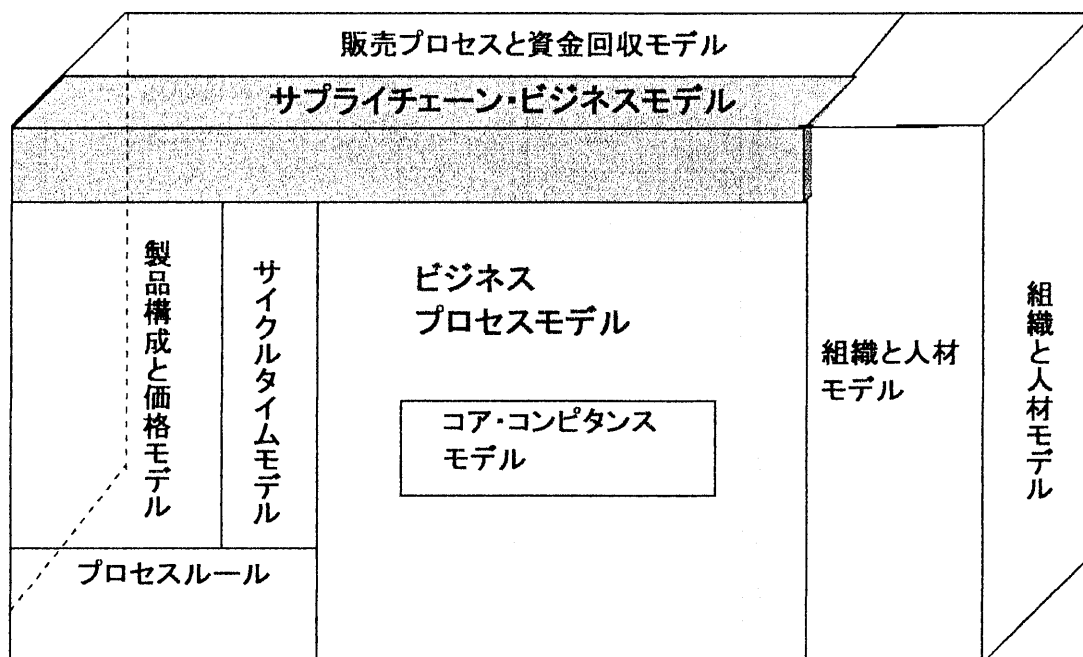


図3-22 ビジネスモデルを表すためのサブモデルの関係

以上 8種類のビジネスモデル表現による可視化を提案した。前述した10種類のビジネスモデルの視点のなかで財務モデル（財務内容）、リスクの分析、表示を除外した。財務モデルは、財務諸表で表すことが正確な数値情報が表示でき、リスクについては、モデル表示よりも、論述や数値表示が適当と判断し、対象から外した。

尚、このビジネスモデルを可視化するための8種類のサブモデル表記については、第3章4節UMLによるビジネスモデル視点別表記法の提案 において具体的な表記方法を提案する。

尚、提案の中でビジネスプロセスモデルは、概要ビジネスプロセスモデルと詳細ビジネスプロセスモデル に区分した。企業の概要を示すビジネスモデルを表すには、まず、概要ビジネスプロセスモデルが理解しやすい。多面的な切り口のビジネスモデル表示により、いくつかの視点から見たビジネスモデルが必要であるが、その主となるのが、ビジネスプロセスモデルである。このビジネスプロセスモデルには、上層から下層に細分化していく方法をとるので、本提案では、第2階層以下を詳細ビジネスプロセスモデルで表記する。

3-1-1 サプライチェーンビジネスモデル

UMLビジネスモデルのビジネスオブジェクトモデルを企業間に拡張するかたちでサプライチェーンを表現する。これは、ビジネスユースケースを、企業間でどのように実現するかを構造を示す構造モデルと振舞の観点から示す振る舞いモデルがある。

提案1 サプライチェーン構造モデル表記の提案

プロセスを実現する企業間の要素と、それらの間の意味的な関係をクラス図で表す。バリューチェーンを構成するプロセス分類でグループ化する。サプライチェーン構造モデルの例を示す

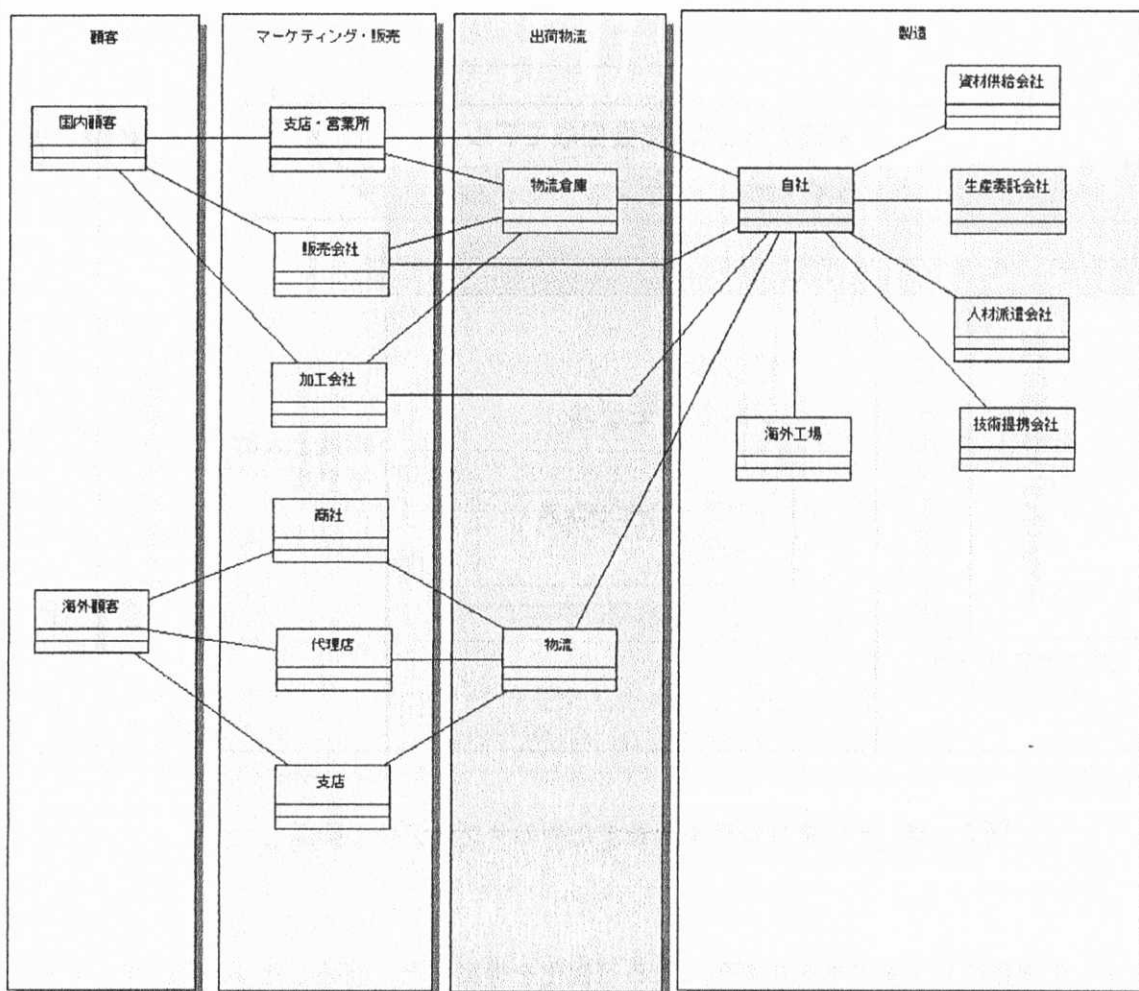


図 3-23 サプライチェーン構造モデル図

図 3-23 では、ビジネスオブジェクトモデルを企業間に拡張する形でサプライチェーン構造モデルを表現している。従来のビジネスオブジェクトモデルでは、企業内の構造モデルを示しており、本研究では、近年のビジネス形態を考慮してサプライチェーンを表現する企業間のサプライチェーン構造モデル図を提案した。

提案1-2 サプライチェーン振る舞いモデルの提案

各要素（企業）が、どのように協調してプロセスを実現するか（商流、金流、物流の表現）を相互作用図（コラボレーション図）で表す。商流は、メッセージのステレオタイプ<<transaction>>で表す。金流は、メッセージのステレオタイプ<<payment>>で表す。物流はメッセージのステレオタイプ<<distribution>>で表す。例を図 3-24 に示す。

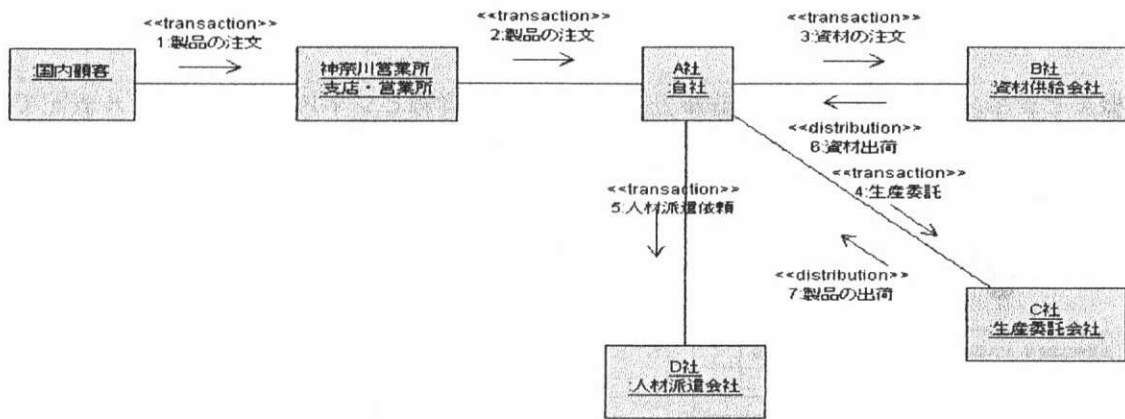


図 3-24 サプライチェーン振舞モデル図

図 3-24 では、振舞モデルでサプライチェーンを示すことにより、そのビジネス活動内容がより明確に表記できることになる。

3-1-2 ビジネスプロセスモデル(提案2 概要ビジネスプロセスモデル表記の提案)

企業の最上位のビジネスプロセスを示し、構成する機能同士の関連をクラス図で表現する。

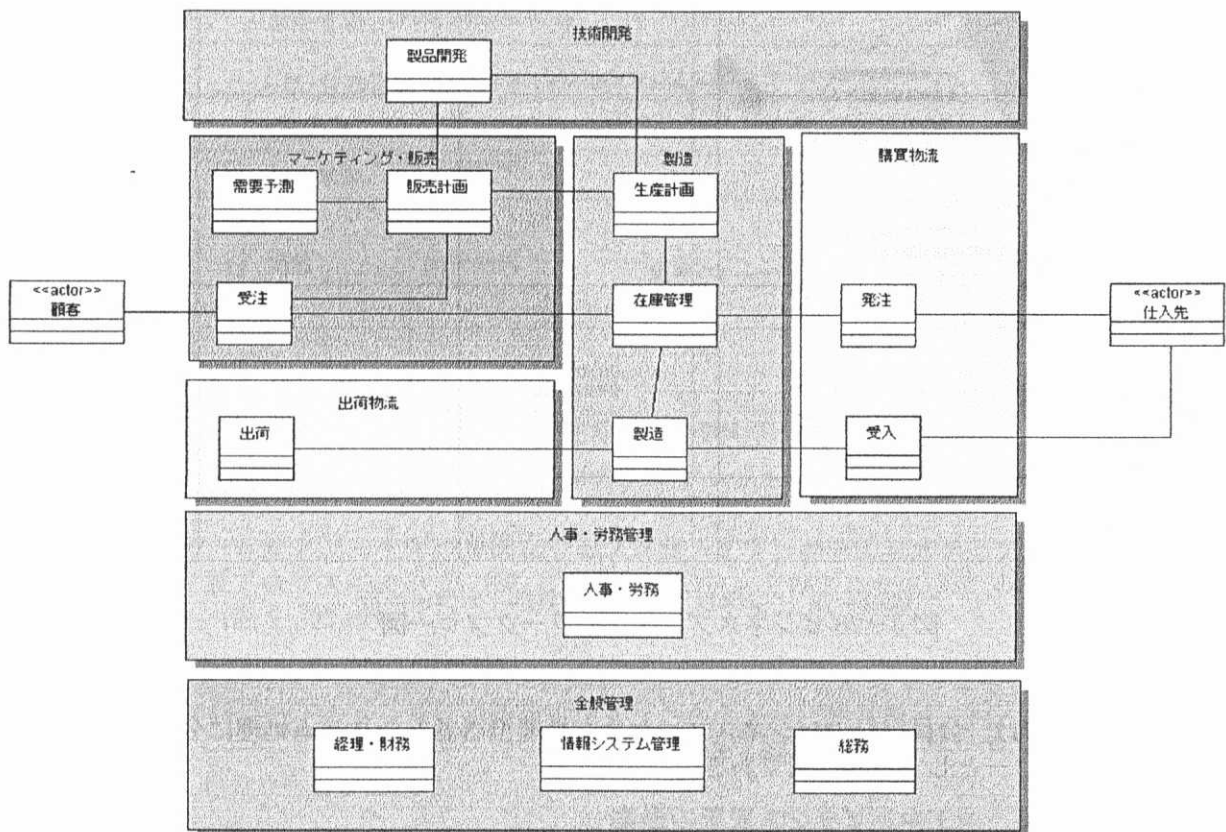


図 3-25 企業ビジネスモデル図

図 3-25 に示す企業ビジネスモデル図によって、企業全体の重要な機能構成が俯瞰される。各機能を、上記バリューチェーンを構成するプロセス分類でグループ化する。

なお、各機能は UML ビジネスモデリングの企業ロール (注1) の単位である。企業ロールを実現するものとしては、組織単位や人的資産、情報資産 (情報システム)、財務資

産、知的資産などの企業資産がある。

提案2-2 詳細ビジネスプロセスモデル表記の提案

概要ビジネスプロセスモデルを詳細表記したものが、詳細ビジネスプロセスモデル表記であるが、ここで使用するUML拡張提案については、次節3-2で詳しく記述する。

ここでは、企業ビジネスプロセスをワークフローモデル図(図3-26)で示す。

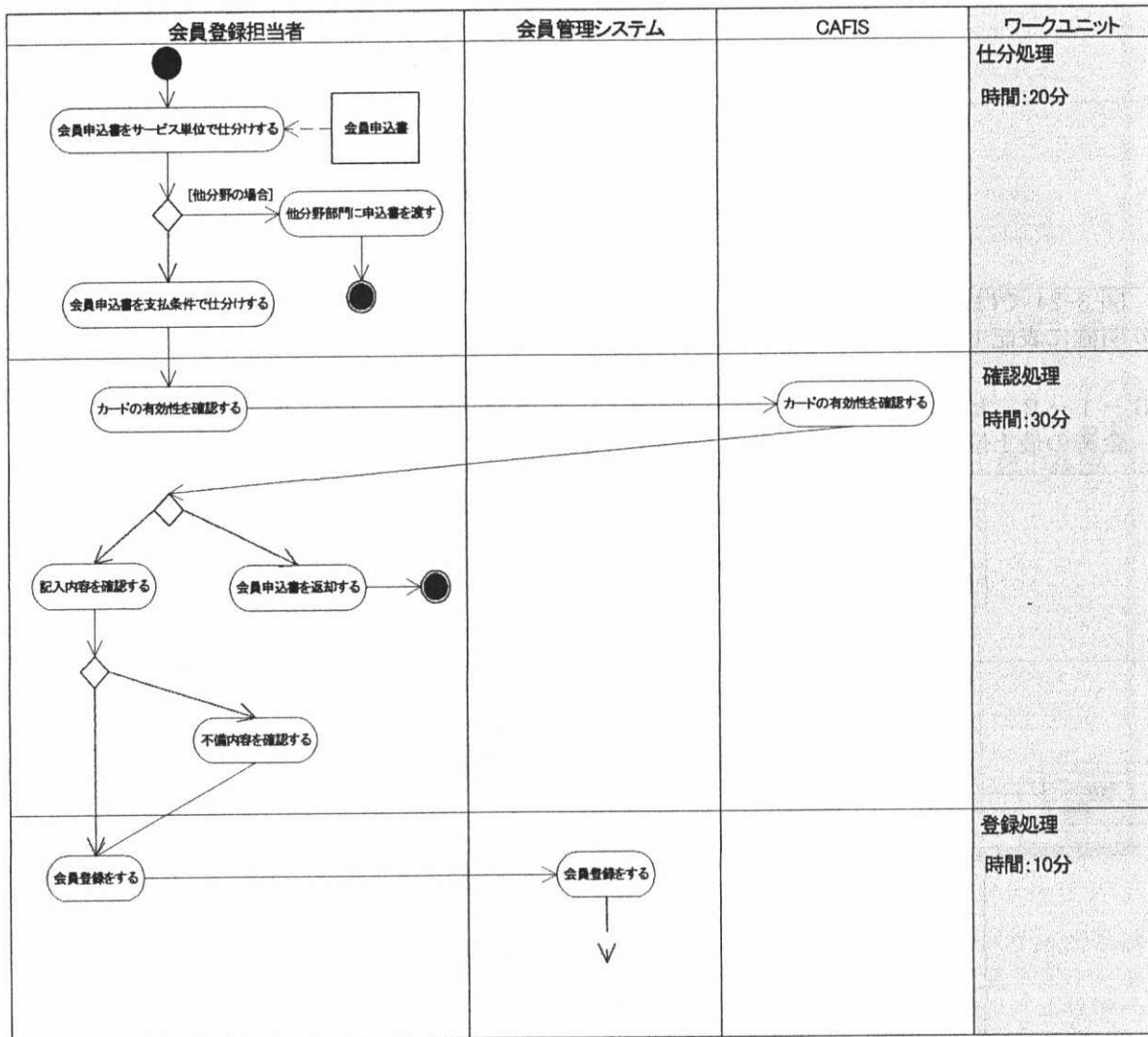


図3-26 ビジネスプロセスのワークフロー図

図3-26では、会員登録のビジネスプロセスを処理担当者とシステム処理に分けて処理の流れをワークフローとして示した図である。

3-1-3 コアコンピタンス表記の提案

バリューチェーンを構成するプロセス分類のうち、その企業の核となる付加価値を生むプロセス分類を選択してプロセスクラスとして構成する。その際、プロセスクラスのステレオタ

イプを設定する。次に、プロセスクラス中で、企業を特徴づけるコアプロセス¹をグループ化して識別する。

1) バリューチェーンのテンプレート

ビジネスモデルのレベルでは、企業間の比較により、優位性や独自性の検討が必要となる。最近の企業合併やM&Aなどでその必要性が増している。企業内に閉じたビジネスプロセス表記であれば自由な設定が許されるが、ビジネスモデルとする場合には、表記基準が必要となる。そこで、図3-27に示すバリューチェーンのテンプレートをビジネスプロセス表記の基準とした。

バリューチェーンを構成する一般的なプロセスの典型的な例としては以下が考えられる。

<主要プロセス分類名>

- ・ Marketing and Sales : マーケティング・販売
- ・ Inbound Logistics : 購買物流
- ・ Operations : 製造
- ・ Outbound Logistics : 出荷物流
- ・ Service : サービス

<支援プロセス分類名>

- ・ Firm Infrastructure : 全般管理 (財務管理、情報管理、経営企画等)
- ・ Human Resource Management : 人事・労務管理
- ・ Technology Development : 技術開発
- ・ Procurement : 調達



出所:「競争優位の戦略(ダイヤモンド社)より作成

図3-27 バリューチェーンのテンプレート

これを、コア・コンピタンスモデルを構成するプロセスクラスのステレオタイプとして用いる。そして、上記主要プロセスによって、UML ビジネスモデリングのプロセス (ビジネスユースケース) を分類する (図3-27 参照)。

図3-28 は、部門または、事業種類ごとにビジネスユースケースを整理することにより、ユースケースによって示される担当者ごとの仕事が整理でき、部門の仕事とその担当者が明確に表すことができる。

(注1) 企業ロール: プロセスを実現するための機能的な役割。

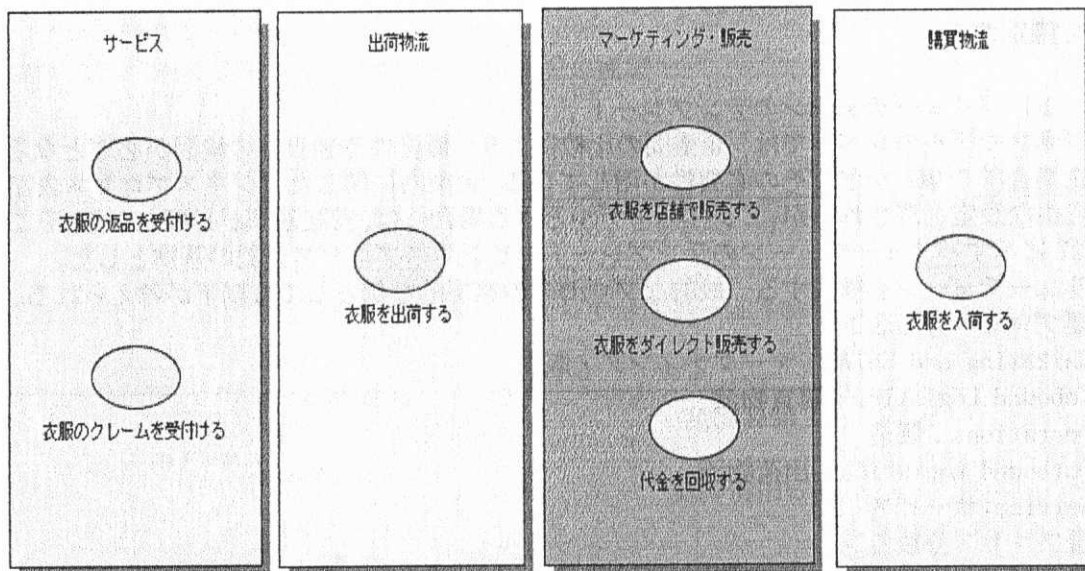


図 3-28 部門別に分類されたビジネスユースケース図

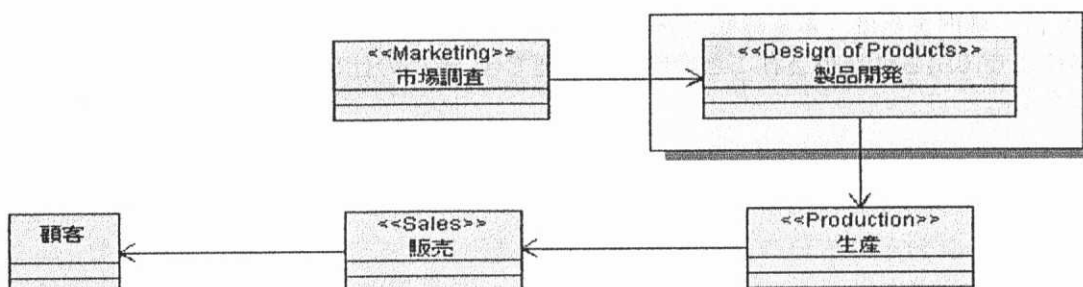
ビジネスユースケースの分類方法としては、以下のものがある。

- ・バリューチェーンを構成するプロセスによるビジネスユースケースの分類
バリューチェーンを構成するプロセス分類のうち、その企業の核となる付加価値を生むプロセス分類を選択してプロセスクラスとして構成する
→コア・コンピタンスモデル図で表す。
- ・企業全体を範囲にをあらわす役割をクラスとしてあらわす。
→サプライチェーン構造モデル図で示す。
- ・ビジネス振る舞いモデルを企業間に拡張してサプライチェーン振る舞いモデルを表現する
→サプライチェーン振る舞いモデル図で示す。

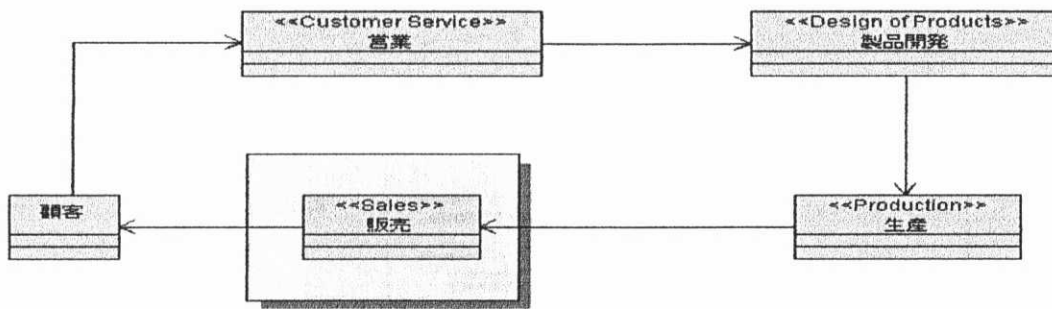
2) コアコンピタンスモデルの提案 (提案3)

バリューチェーンを構成するプロセス分類のうち、その企業の核となる付加価値を生むプロセス分類を選択してプロセスクラスとして構成する。その際、プロセスクラスのステレオタイプを設定する。次に、プロセスクラス中で企業を特徴づけるコアプロセスをグループ化して識別する。コアプロセスとは、その企業の強みや利益を生み出すプロセスを指す。

例 1：独自の商品開発力や独自技術力をコアコンピタンスとして、競争力のある商品を顧客に提供できる会社の例



例2：顧客の要求にきめ細かな対応により、信頼のあるブランドを築き、販売力を維持する。



例3：低価格の商品を供給することができる生産能力と原価管理ができることにより、他社より競争力を持つ。

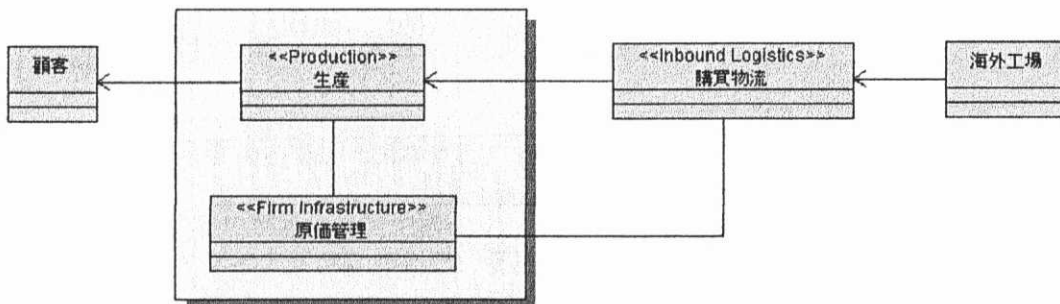


図 3-29 UML によるコアコンピタンス表示 (例1から例3)

図 3-29 は、3種類のコアコンピタンスを示すビジネスモデルを示しており、そのコアコンピタンスの内容は、プロセスクラスの詳細を見ることによって示すことができる。また、コア・コンピタンスモデルのプロセス分類名でステレオタイプ化することにより、コア・コンピタンスモデルを構成するプロセスクラスをプロセス分類名でステレオタイプ化した。

例：<<Production>> <<distribution>>

このステレオタイプ化により、部品の共有化などの使用方法での活用が可能となる。

3-1-4 サイクルタイムモデルの提案 (提案4)

サイクルタイムを示すことは、ビジネス形態を理解する上で重要な要素と考える。サプライチェーン構造モデルをシーケンス図に変換し、情報を流すタイミングや出荷するタイミングをコメントとして追加することにより、サプライチェーン内の情報流、物流の時系列な動きとサイクルタイムを理解できる。

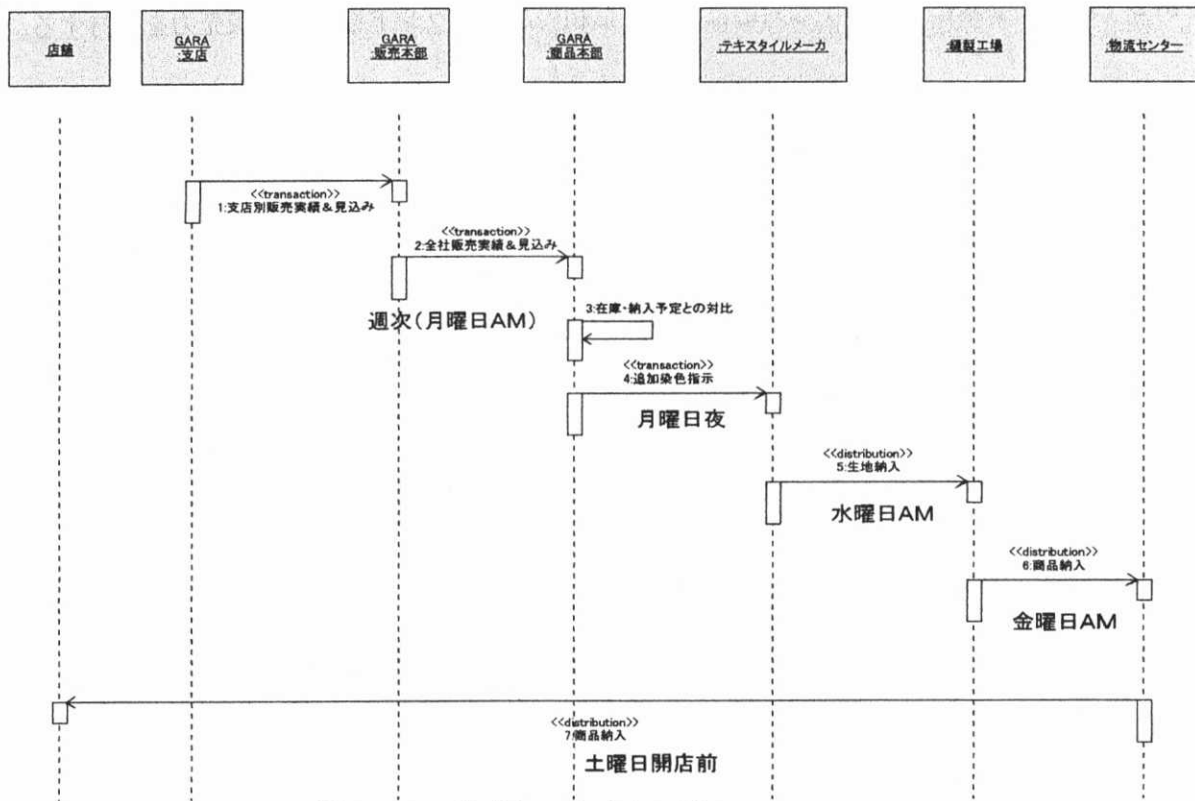


図 3-30 サイクルタイムモデル

図 3-30 では、店舗、本部、工場、物流センター間の 1 週間を単位とした情報流と物流を示している。

3-1-5 組織と人材モデル表記の提案（提案 5）

組織は、UML ビジネスモデルの組織ビューを構成する組織構造である。組織は、図 3-31 のとおり、組織ビューで表記する。

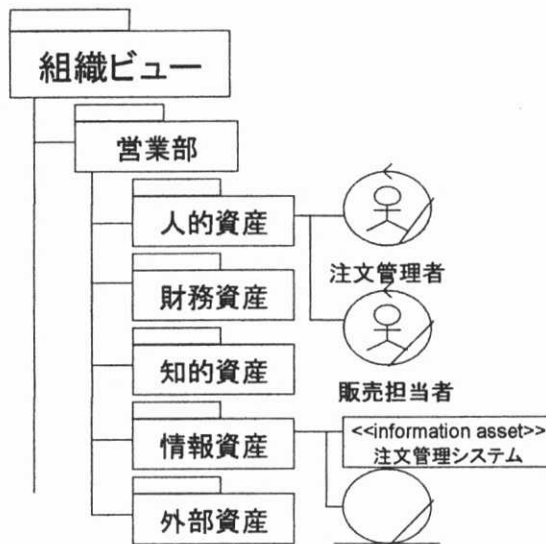


図 3-31 組織ビュー

人材は、各組織を構成する人的資産（ワーカー）で表す。人材の構成は、図 3-32 資産構造で表す。ワーカーの属性としてスキルレベル、年齢などを定義できる。

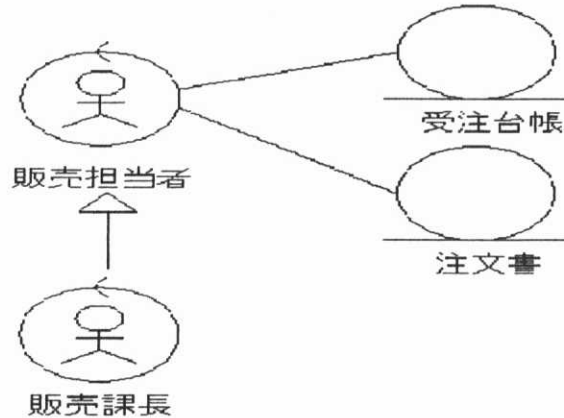


図 3-32 人材構成図（人材資産構造）

3-1-6 製品構成と価格モデル表記の提案（提案 6）

製品は、知的資産の製品クラスで構成する。各製品のプロパティに価格を設ける。図 3-33 は、製品の構成を示すダイアグラムである。価格は、プロパティとして設定できる。

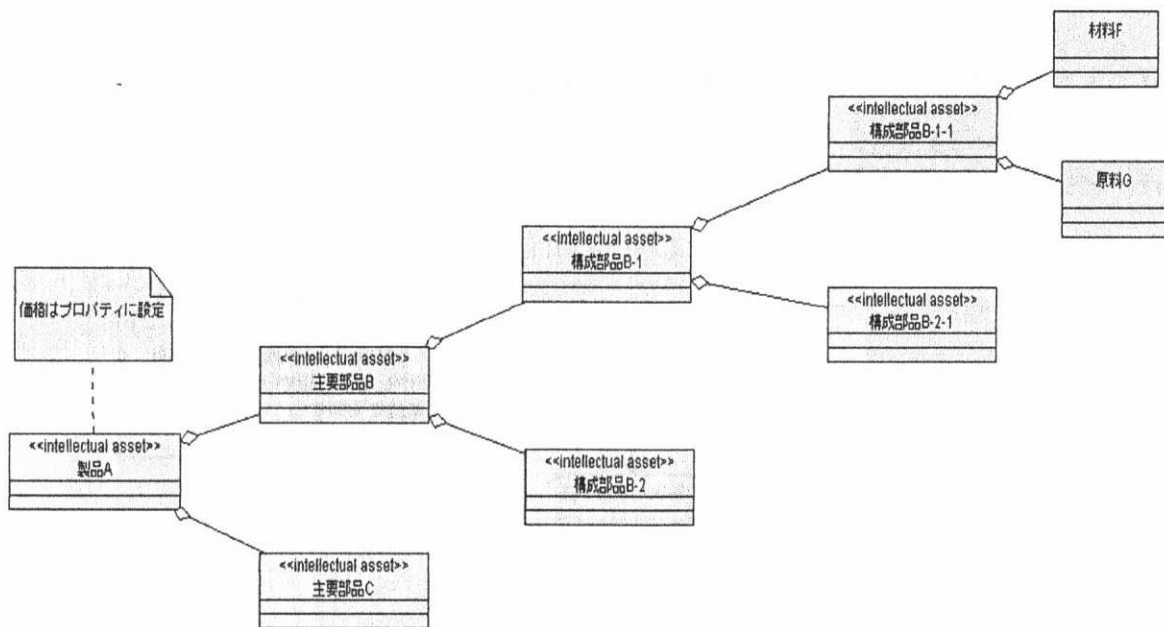


図 3-33 製品構成図

図 3-33 は、クラス図を使用して、製品構成を示した。各構成部品の内容は、構成部品の属性によって詳細に示すことができる。

3-1-7 販売方法と資金回収モデル表記の提案（提案 7）

顧客までの流通経路や販売方法と資金回収のモデルを明確にする。相互作用図を使用する。サ

プライチェーン振る舞いモデルと同様、情報の流れと商流、金流、物流をメッセージのステレオタイプで表す。サプライチェーンビジネスモデルが物流に焦点を当てているのに対して、当モデルは資金回収に焦点を当てる（図 3-34 参照）。ここでは、アクティビティ図、相互作用図を使用する。情報流、商流、金流は、メッセージのステレオタイプで表す。

E コマースにおける資金回収のプロセスの例を次の図に示す。

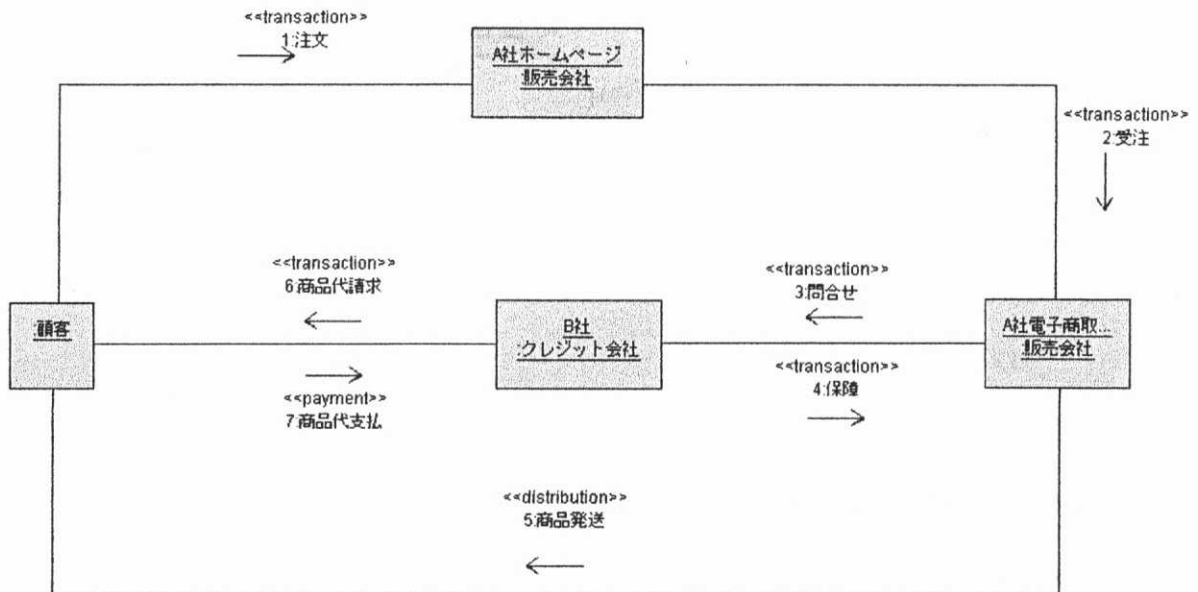


図 3-34 UML による資金回収プロセス図

3-1-8 プロセスルール記述方法の提案（提案 8）

UML では、ビジネスユースケース、アクティビティに対してノートを使用して記述する。目的は、ビジネスプロセスの記述による補足説明が可能となる。（参照図 3-35）

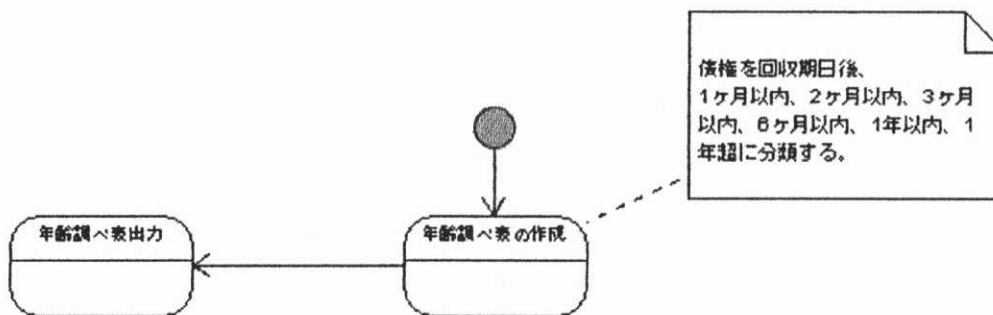


図 3-35 プロセスルールの例

以上 ビジネスモデルを 8 つの視点で UML 表記する提案を行った。

本研究の提案により、ビジネスの実体が 8 つの視点から可視化でき、ビジネスの概要を理解することがより容易に明確に示すことが可能となった。

尚、このビジネスモデル表記に関する本研究の提案は、論文としてビジネスモデル学会に投稿し論文誌 (Vol. 3)、2005 [140] に掲載された。

3-2 UMLによるビジネスプロセスの基本的表記の拡張提案

ビジネスモデルを詳細化していく上でビジネスプロセスの表記方法が重要である。ビジネスプロセスモデルをUMLで表記する場合、従来の先行研究では十分なビジネスモデルが記述できない。その問題点を明確にするため、下記に課題を記述する。

この節では、ビジネスプロセスモデルをUML表記する上の記述方法に関する提案を行う。

課題2. UML表記方法にビジネスモデルを表すために十分な表記機能がない。

この課題に対して、その表記方法として、全体の継続性を考慮して、ビジネスモデル、ビジネスプロセスモデルでのUML拡張を検討した。課題2の対応として下記の拡張を提案する。

3-2-1 基本構造の表記と4つの視点での可視化

課題2に対する解決策として、提案を示す。

拡張提案1：ビジネスプロセス表記として、基本構造表記と4つの視点で可視化する。

企業の戦略と仕組みを考慮した表記法として、基本構造表記と4つの視点で可視化する方法を提案する。

ビジネスプロセスを表記する場合に、統一的な視点で捉えるために、基本構造として、外部的視点からユースケースを作成し、内部的視点から、振る舞いモデリングと構造モデリングを表記する。

1) 基本構造の考え方

前述のように、ビジネスプロセス表記については先行研究があるが、ビジネスプロセス（業務フロー）の記述が中心となり、企業の組織階層やビジネスルール、ビジネスゴールなどは含まれていない。本研究では、以下の観点からビジネスプロセスの基本構造表記法を考察する。

- ・経営戦略の反映（BSCの取込）
- ・AS-ISのみではなく、To-Beのモデル化を行う
- ・システムモデリングとの継続性

<表記の基本構造>

ビジネスプロセスモデリングの基本構造を図3-36に示す。まず、企業目的を遂行するために「ビジネスは何をするのか」（機能）を明らかにする。UMLでは、価値を産出するための機能一つ一つをユースケースで表す。また、価値を提供する相手や価値を産出するために関わっている外部要素をアクターとして表しユースケース図を作成する。

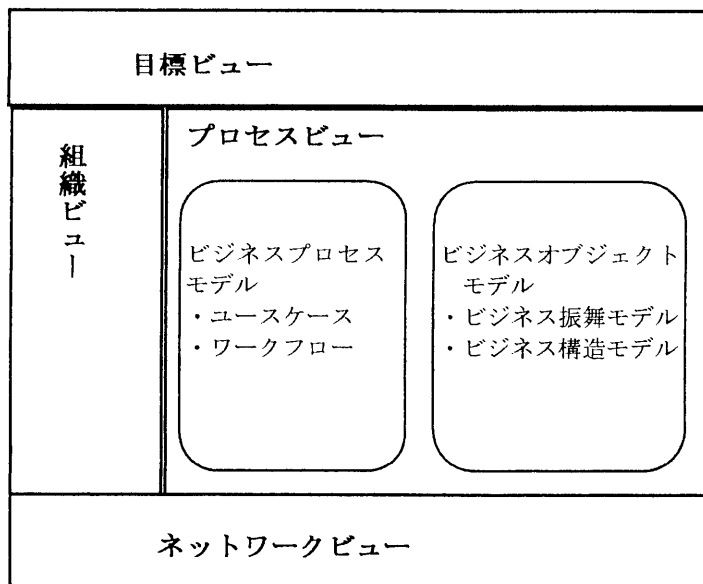


図3-36 ビジネスプロセスモデリング表記の基本構造

次に、ビジネスの機能を「誰が実現するのか」（構造）を明らかにする。ビジネスを構成するものには、人、物、金、情報などがあるが、UMLでは、これらをクラスとして表し、クラス間の関係をクラス図で表現する。最後に、ビジネスを構成する要素がビジネスの機能を「どのように実現するのか」（振る舞い）を明らかにする。具体的には、ユースケース単位に相互作用図やアクティビティ図を作成する。相互作用図は、構成要素がお互いにどのように作用しながら機能を実現するのか（相互作用）を表すのに使用し、アクティビティ図は、構成要素がどのように作業を進めながら機能を実現するか（ワークフロー）を表すために使用する。

2) 4つの視点での可視化

基本的な表記法は、ビジネスプロセスを4つのビューからモデル化を行う。これによって、企業全体の重要な機能構成が俯瞰される。各機能を、上記バリューチェーンを構成するプロセス分類でグループ化する。なお、各機能はUMLビジネスモデリングの企業ロールの単位である。

企業の基本構造を踏まえ、ビジネスプロセスを下記の4つの視点（目標、プロセス、組織、ネットワーク）から可視化する。これにより、ビジネスプロセスの内容が明確に表示できる。

- ・ 目標ビュー
- ・ プロセスビュー
- ・ 組織ビュー
- ・ ネットワークビュー

企業内部のビジネスプロセスを記述するモデル。UMLによる基本的な表記方法の拡張で述べた表記方法に沿ってビジネスプロセスモデルを表記した。

ビジネスプロセスモデリングでは、図 3-37 に示すように、4つのビュー（視点から）モデル化を行う。以下、各ビューの作成内容について説明する。

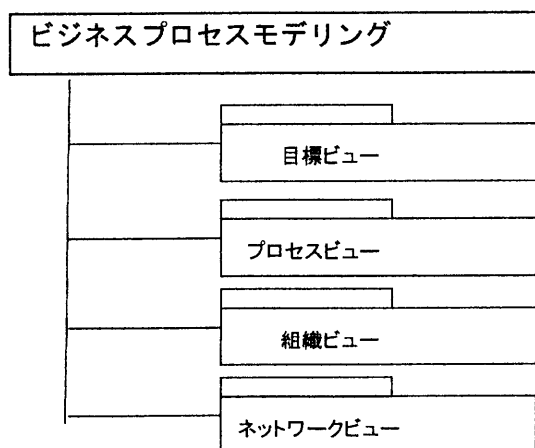


図 3-37 ビジネスプロセスモデリングの4つのビュー

(1) 目標ビュー

目標ビューの目的は、企業の目標および指標とそれらの因果関係を示すことにより、そのビジネスプロセスを明確にする役割がある。目標ビューの表記方法では、Balanced Scorecardの戦略マップを、クラス図を拡張することで表記する。図 3-38 に示した目標ビューは、戦略的目標となるCSF(Critical Success Factor)とそれに対する業績指標、それを實現するアクションプランから構成されUMLのクラスで表現される。CSFや業績指標はバランススコアカード(以下BSC)の4つの視点で分類することができ、それぞれにステレオタイプで示す。それぞれ、<<financial>>、<<customer>>、<<internal business process>>、<<learning and growth>>にステレオタイプ化できる。BSCの戦略マップは、具体的な値を持ったCSFオブジェクト間、業績指標オブジェクト間のリンクでオブジェクト図を作成することによって表す。

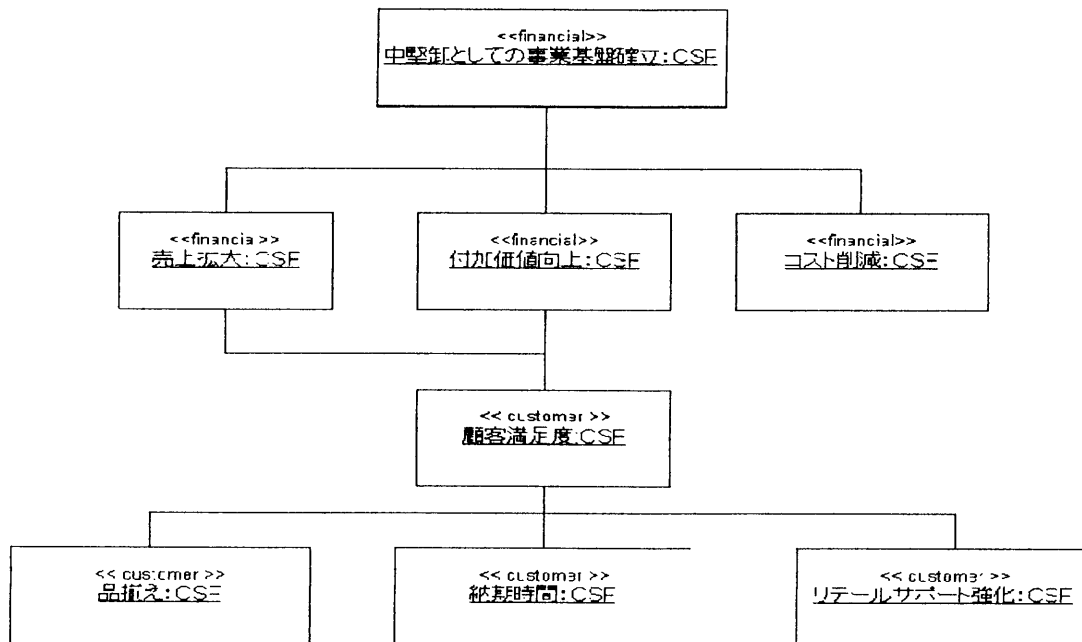


図 3-38 目標ビューの例

(2) プロセスビュー

プロセスビューでは、ビジネスプロセスをユースケースを使用して表記するプロセスモデルとワークフローさらに、ビジネスオブジェクトモデルとしてビジネス構造モデルとビジネス振舞モデルを作成する。

① ビジネスプロセスモデル

ビジネスプロセスモデルは、以下の二つより成る。

a) ビジネスユースケースモデル

ビジネスのプロセスの単位をユースケース（ビジネスユースケースと呼ぶ）で表し、プロセスが価値を産出する相手およびプロセスの実現に関わっている外部要素をアクターとして表したユースケースモデル（図 3-39 参照）。

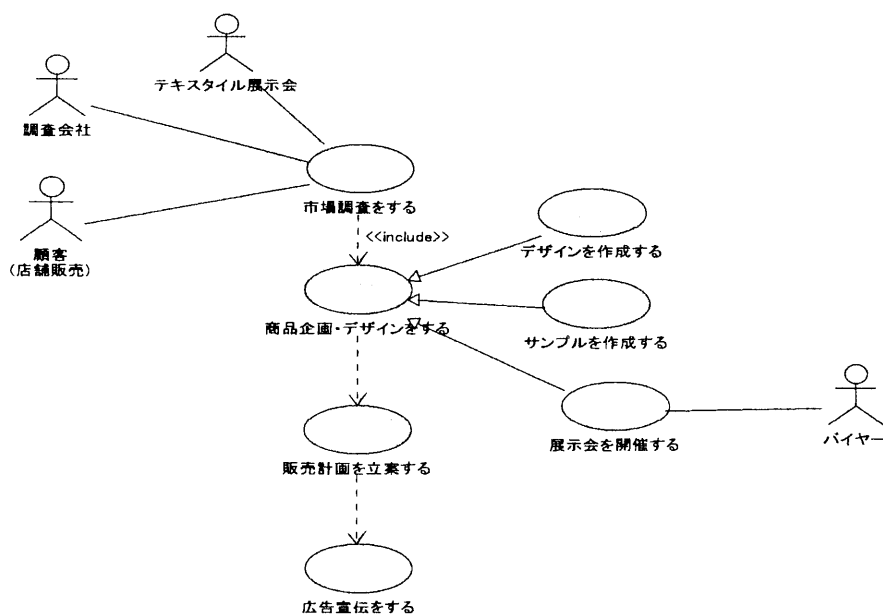


図 3-39 ビジネスユースケースモデルの例

b) ワークフロー

ビジネスオブジェクトモデルで抽出された企業内の要素が、ビジネスユースケースを実現する手順を表したモデル。企業内の要素をレーンとし、作業の流れをアクティビティ図で記述する。(参照 図3-27 ビジネスプロセスのワークフロー図)

プロセスビューでは、企業を構成する事業と、そこで遂行されるプロセスを整理する。プロセスはUMLのユースケースで表され、典型的なものには資本調達プロセス、資産確立プロセス、資産維持プロセス、資本運用プロセス、経営プロセスがあり、それぞれのプロセスにステレオタイプを作成することを提案する。

<<financial process>>、<<investment process>>、<<maintenance process>>、<<business process>>、<<management process >>でステレオタイプ化される。

②ビジネスオブジェクトモデル

ビジネスオブジェクトモデルは、以下の二つより成る。

a) ビジネス構造モデル (図3-40 参照)

ビジネスのプロセスを実現する企業内の要素 (人的資産、財務資産、情報資産、知的資産、外部資産) と、それらの間の意味的な関係をプロセス単位にクラス図で表したものの。

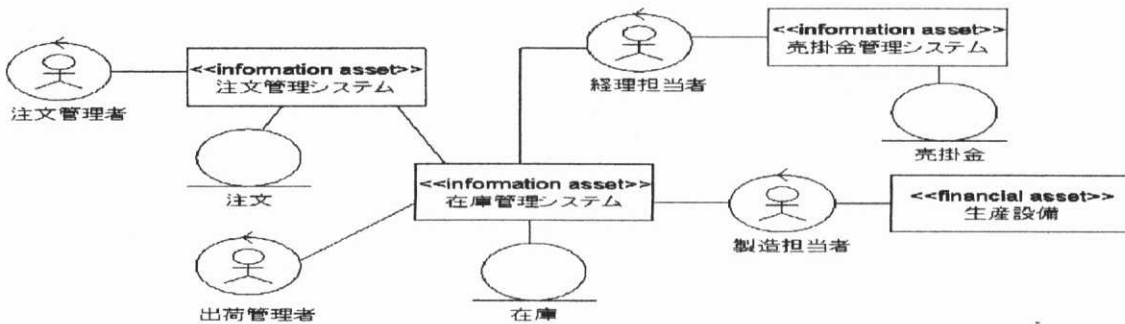


図3-40 ビジネス構造モデルの例

b) ビジネス振る舞いモデル

各内部要素が、どう協調してプロセスを実現するかを相互作用図 (コラボレーション図) で表したものの (図3-41)。

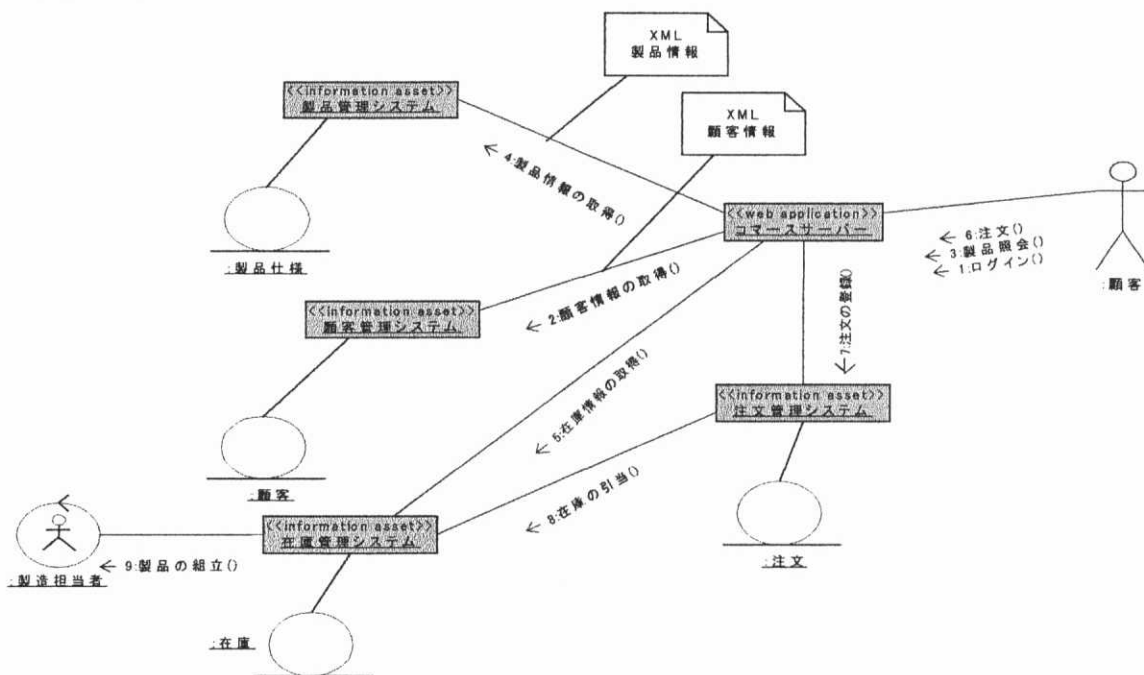


図3-41 ビジネス振る舞いモデル

(3) 組織ビュー

組織ビューでは、ビジネスのプロセスを実現する企業内の要素（人的資産、財務資産、情報資産、知的資産、外部資産）を組織単位に整理したものである（図 3-42 参照）。

組織ビューではプロセスを実現するための組織とそこに配分される資産を整理する。

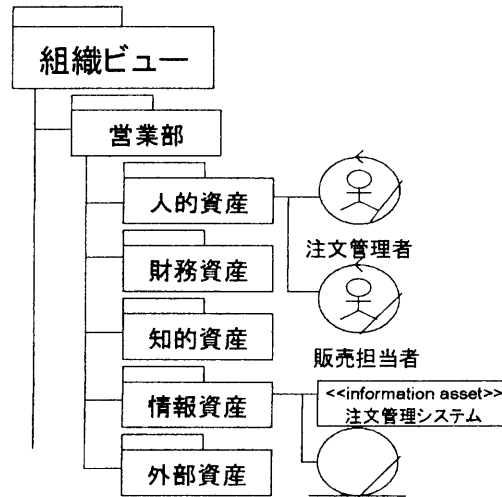


図 3-42 組織ビューの例

組織ビューは、上位階層から下位階層へブレイクダウンができ、最終的に人材ならば担当者迄つながらる仕組みになっている。また、資産間の概念的関係をクラス図（資産構造）として表す。

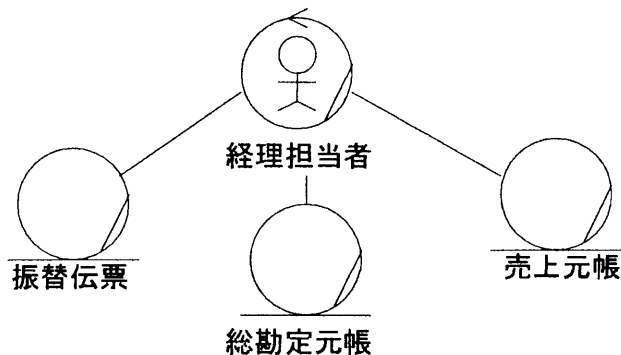


図 3-43 資産構造の例

図 3-43 では、販売に関する資産構造を表しており、この例では、販売担当者が受注台帳と注文ファイルを管理していることを示している。

資産はUMLのクラスで表され、典型的なものには人的資産、財務資産、情報資産、知的資産、外部資産があり、それぞれの資産オブジェクトにステレオタイプを作成することを提案する。下記にステレオタイプの例を表示する。

<<worker>>、<<financial asset>>、<<information asset>>、<<intellectual asset>>、<<actor>>でステレオタイプ化される。

(4) ネットワークビュー

ネットワークビューでは、企業を構成する事業所や工場など物理的な要素の配置を配置図に表す。ネットワークビューでは工場や事業所の物理的な配置を整理する。事業所などの建物をUMLのノードで表し、それらのネットワークを配置図で表す（図 3-44 参照）。

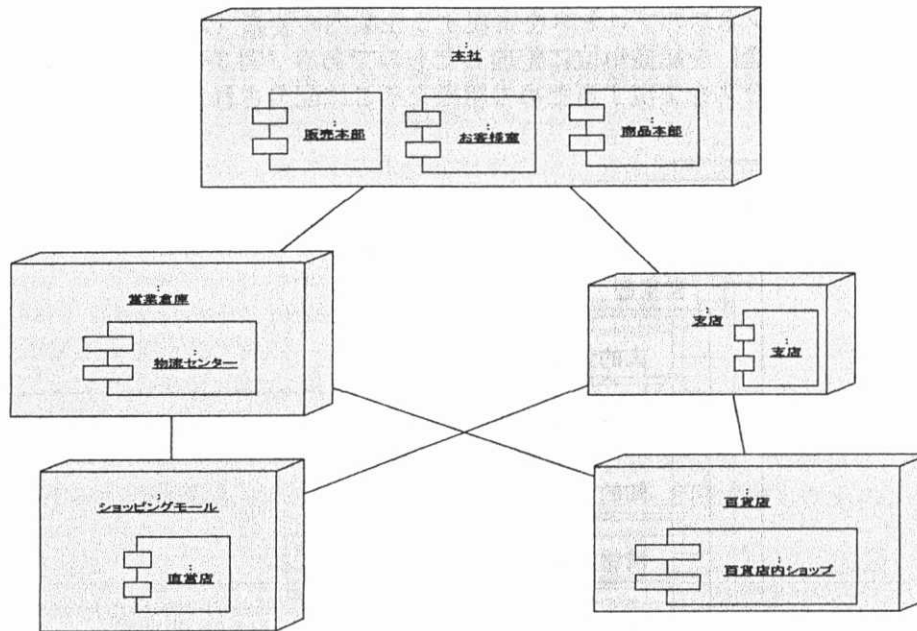


図 3-44 ネットワークビューの例

3-2-2 クラスの要素をステレオタイプで表示する

拡張提案 2 : 4つのビューの構成要素をステレオタイプで表示する。

4つのビューでビジネスモデルを表記する場合、クラスやオブジェクトの構成要素を表示できると理解しやすい。この検討にあたっては、ビジネスモデル設計からシステム開発の継続性確保（シームレス）のために以下のように上方への拡張を採用した。つまり、システムモデルで使用できる表記法をビジネスプロセスモデル、ビジネスモデルで使用できる繋がる表記を採用した。

この拡張は、ステレオタイプ²の設定については、UML プロファイルを設定することとし、それ以外の拡張と区分する。目標ビューの各オブジェクトは、それぞれ、<<financial>>、<<customer>>、<<internal business process>>、<<learning and growth>>にステレオタイプ化できる。BSCの戦略マップは、具体的な値を持った CSF オブジェクト間、業績指標オブジェクト間のリンクでオブジェクト図を作成することによって表す。

プロセスビューでは、企業を構成する事業と、そこで遂行されるプロセスを整理する。プロセスはUMLのユースケースで表され、典型的なものには資本調達プロセス、資産確立プロセス、資産維持プロセス、資本運用プロセス、経営プロセスがあり、それぞれのプロセスにステレオタイプを作成することを提案する。

<<financial process>>、<<investment process>>、<<maintenance process>>、<<business process>>、<<management process >>でステレオタイプ化される。

組織ビューではプロセスを実現するための組織とそこに配分される資産を整理する。資産はUMLのクラスで表され、典型的なものには人的資産、財務資産、情報資産、知的資産、外部資産があり、それぞれの資産オブジェクトにステレオタイプを作成することを提案する。

<<worker>>、<<financial asset>>、<<information asset>>、<<intellectual asset>>、<<actor>>でステレオタイプ化される。

以上 ビジネスモデル表記法に関する提案として、8視点によるビジネスモデル表記の提案と2つのUML表記の拡張提案を行った。

² ステレオタイプ

ステレオタイプとは、UMLのモデル要素（クラスなどモデルを構成する要素）の語彙を拡張するための手法で、モデル要素に<< >>につけて意味を拡張する。

第4節 事例検証と考察

4-1 実企業への適用事例

前節で提案したUMLによるビジネスモデル表記について、実際の企業に適応して検証を行った。C社は、中堅アパレルメーカーである。下記にその対象となったC社の会社概要と主な業務を示す。

業種	アパレル（製造卸）		
本社	東京	支店	大阪、名古屋、福岡、仙台、札幌
工場	ファブレス（全て委託加工）		国内に数社の協力縫製工場
倉庫	物流センター	関東に一箇所	商品 ニット、カジュアル中心 季節変動が大きい、Tシャツなどは季節に関わらずある一定量は売れる
顧客層	メンズ、ウィメンズ、kids		価格帯 数千円のレベル

C社の主な業務処理の概要（現状）は、以下のとおりである。

商品企画	社内デザイナーが、デザインだけでなく、生地を選定、副資材の選定も実施
生産	仕入担当から協力工場に製造を依頼。生地や副資材の調達は協力会社にまかせる。製造オーダーは、商品名と数量、納期にて指示。国内委託先で縫製、マーク入れ、包装してできあがったものを一旦物流センターに保管 値付けは変動要素が多いので物流センターで実施
仕入れ	工場への依頼分は検品後合格分を全品買掛計上。 納期遅れの発生を前提に物流センターに在庫。発注は、店舗から注文状況と今後の販売計画をもとに本部で決定。週次発注。
物流	店舗にはストックスペースがあまりない。（店舗面積を広くするため） 物流センターから店舗には翌々朝には配送（発注締め切りは22時）。 商品入替時の売れ残り品は一旦各店舗から物流センターに返送。新商品を届けにきた物流業者がそのまま売れ残り商品を回収。
売上げ	店舗POSで、商品の売上げ管理。当日の売上げ情報を、営業終了後、本部のサーバに伝送。夜間バッチで前日までの売上げ実績の詳細把握が可能。
商品体系	色違い、サイズ違いは同じ型名コード。 商品コードは 型名コード+色コード+サイズコード
店舗	独立採算。本社から仕入れて販売する。 売変権限は店舗。採用権限はない。店舗設計は全国統一。 ネットショップも店舗扱い。オープン時間 10時から20時 四半期毎の商品入れ替え時のみ、午前中閉店 店長は、店舗PCから当日実績を把握。今後の販売予測を自ら考えて追加発注を行う。勤怠管理の基礎データも店舗PCに入力
販売計画	商品企画段階で年度の販売数量を決定。この段階で協力工場には情報提供。さらに シーズン3ヶ月前に月単位の販売数量を決定。この段階で店舗ごとに販売目標をアサインする。
生産計画	協力工場では、販売数量とデザイン情報をもとに素材や副資材の確保を始める。3ヶ月前の月別販売数量をもとに生産計画を立案する。 なお、店舗ごとの販売目標はトップダウンではなく、店長のコミットによる。
本社	商品企画、生産調整、実績管理、人材育成機能、SP、情報システムなど店舗支援機能に徹する。
システム	売上管理システム（計画系含む）、仕入管理システム（計画系含む）、店舗管理システム、バックオフィスシステム（人事、経理、庶務）

4-1-1 提案によるビジネスモデル表記の適応例

S P A兼用型アパレル企業(直営店を持ち製造小売の形態もとりながら、従来型である問屋経由の製造卸の形態もとっているアパレル企業)の企業モデルである。その企業のビジネスモデルとビジネスプロセスモデルを前節で提案した8つの視点別ビジネスモデル表記とビジネスプロセスの拡張表記を使用して記述を行った。

1) 8つの視点によるビジネスモデル表記適用事例

(1) サプライチェーンビジネスモデル

UML ビジネスモデルのビジネスオブジェクトモデルを企業間に拡張するかたちでサプライチェーンを表現する。これは、ビジネスユースケースを、企業間でどのように実現するかを構造と振舞いの観点からモデル化したものである。

a) サプライチェーン構造モデル

バリューチェーンを実現する各要素(企業・部門)と、それらの間の意味的な関係をクラス図で表し、バリューチェーンを構成するプロセス分類でグループ化する。外部組織は<<actor>>として識別している。(図3-45)

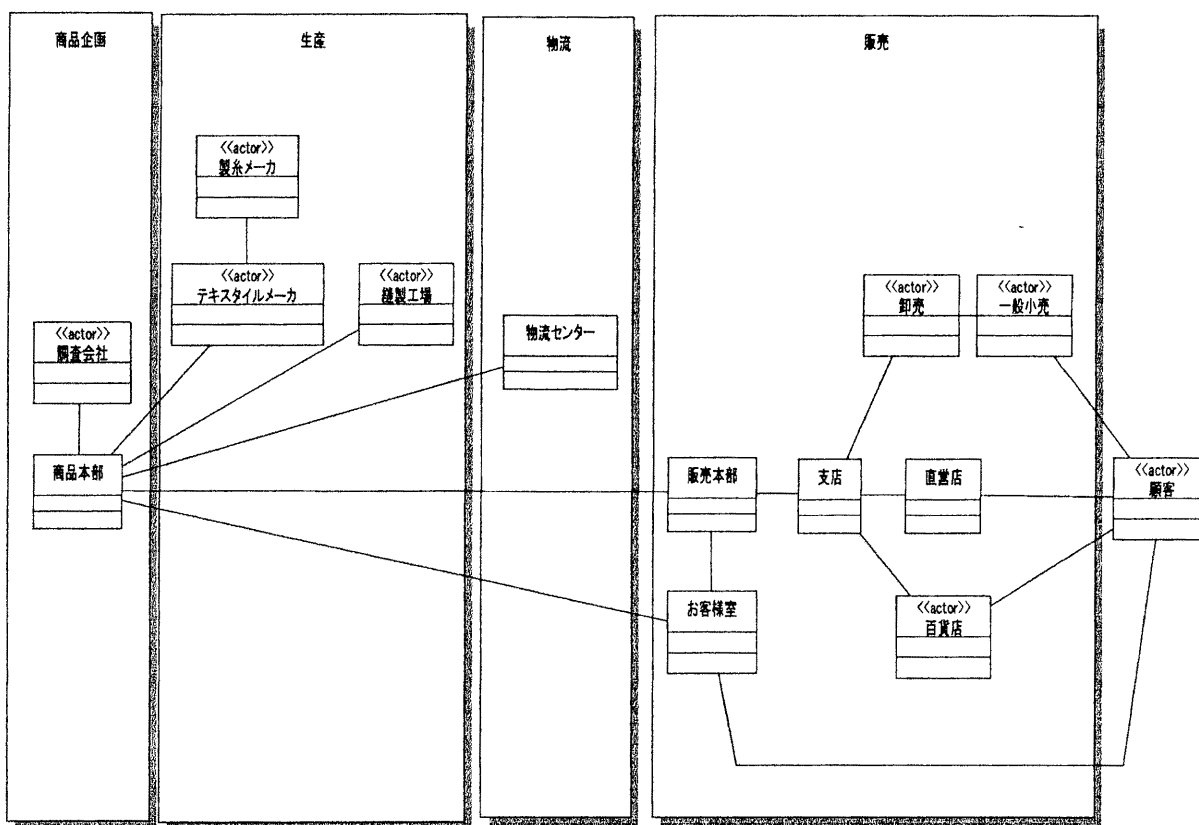


図3-45 サプライチェーン構造モデル図 (C社プロジェクト資料より引用)

b) サプライチェーン振舞いモデル

各要素(企業)が、どのように協調してプロセスを実現するか(商流、金流、物流の表現)を相互作用図(コラボレーション図)で表す。商流は、メッセージのステレオタイプ<<transaction>>で、金流は、メッセージのステレオタイプ<<payment>>で、物流はメッセージのステレオタイプ<<distribution>>で表わしている。(図3-46 参照)

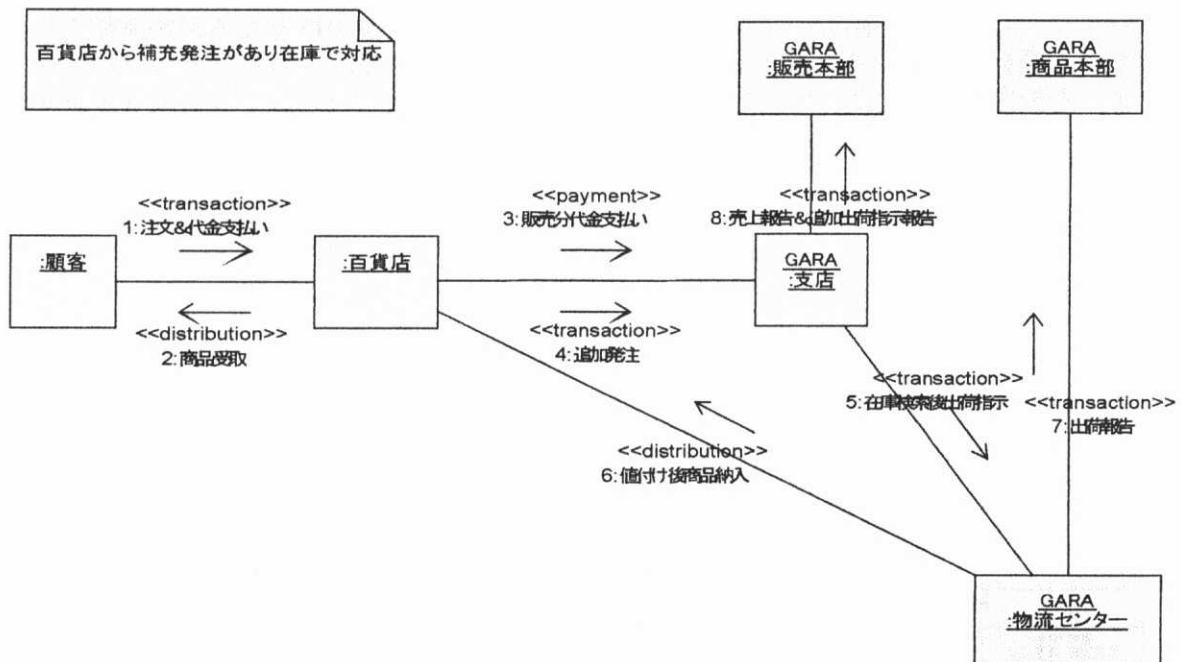


図 3-46 サプライチェーン振舞いモデル図 (C社プロジェクト資料より引用)

(2) 企業内ビジネスモデル (ビジネス機能構造モデル)

プロセスを構成する機能同士の関連をクラス図で表現する。これによって、企業全体の重要な機能構成が俯瞰される。ベースとなるバリューチェーンモデルと対比することで、自社での機能範囲を特定する。なお、各機能はビジネスプロセスモデリングの企業ロール³の単位である。(図 3-47 参照)。

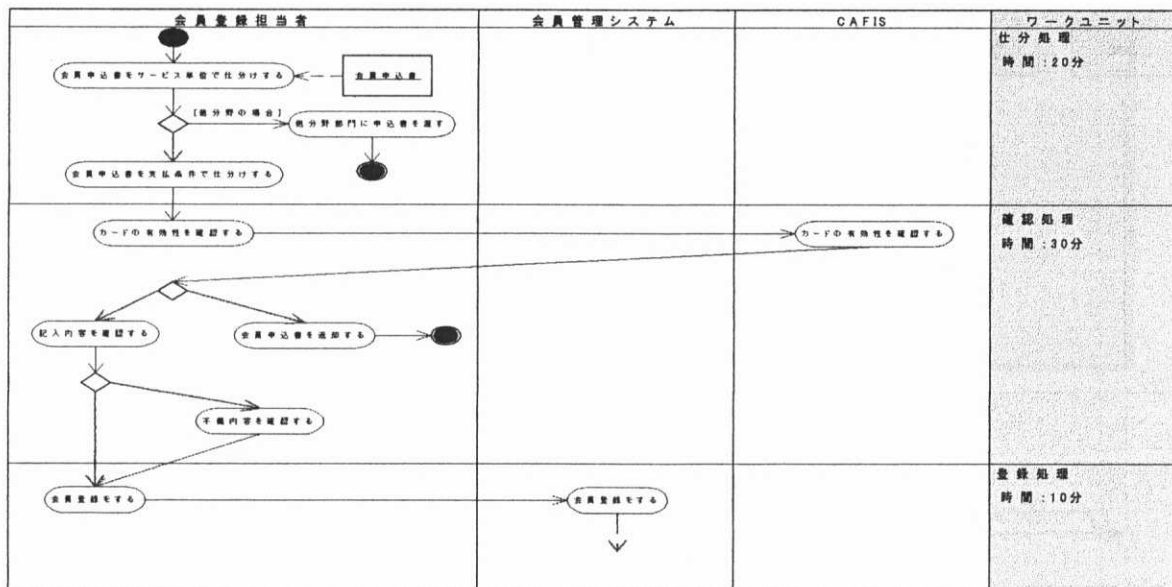


図 3-47 ワークフロー図 (C社プロジェクト資料より引用)

³企業ロール

プロセスを実現するための機能的な役割。企業ロールを実現するものとしては、組織単位や人的資産、情報資産 (情報システム)、財務資産、知的資産などの企業資産がある。

(3) コア・コンピタンスモデル

バリューチェーンを構成するプロセス分類のうち、その企業の核となる付加価値を生むプロセス分類を選択してプロセスクラスとして構成する。その際、プロセスクラスのステレオタイプを設定する。次に、プロセスクラス中で、企業を特徴づけるコアプロセスをグループ化して識別する。(図 3-48 参照)

例 1：商品の取り揃えと店頭対応により他社と差別化する。

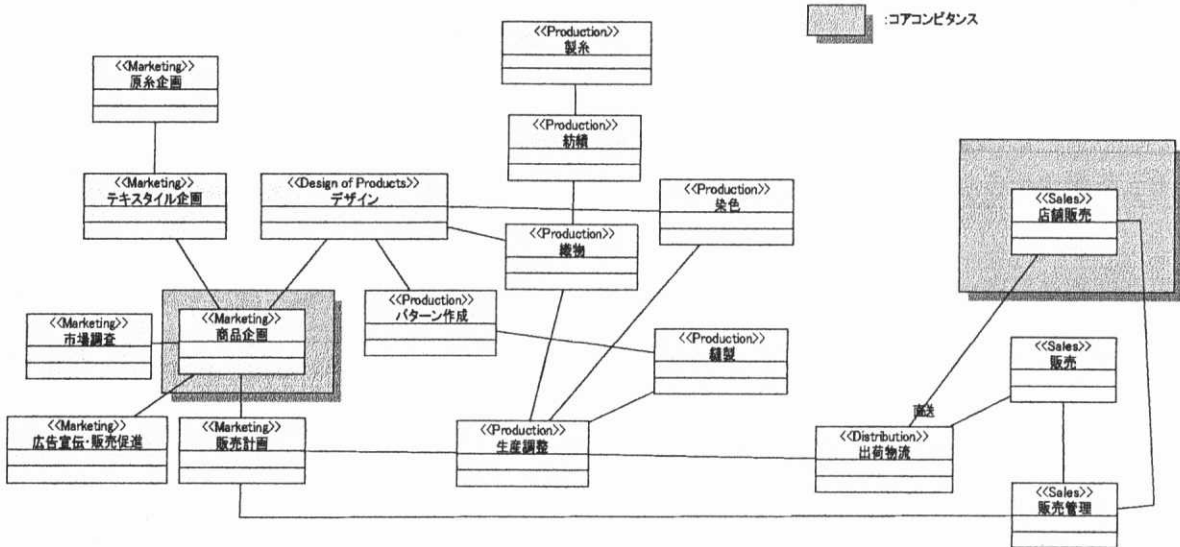


図 3-48 コア・コンピタンスモデル図 (C社プロジェクト資料より引用)

(4) サイクルタイムモデル

サプライチェーン構造モデルをシーケンス図に変換し、情報を流すタイミングや出荷するタイミングをコメントとして追加して示す。(図 3-49 参照)

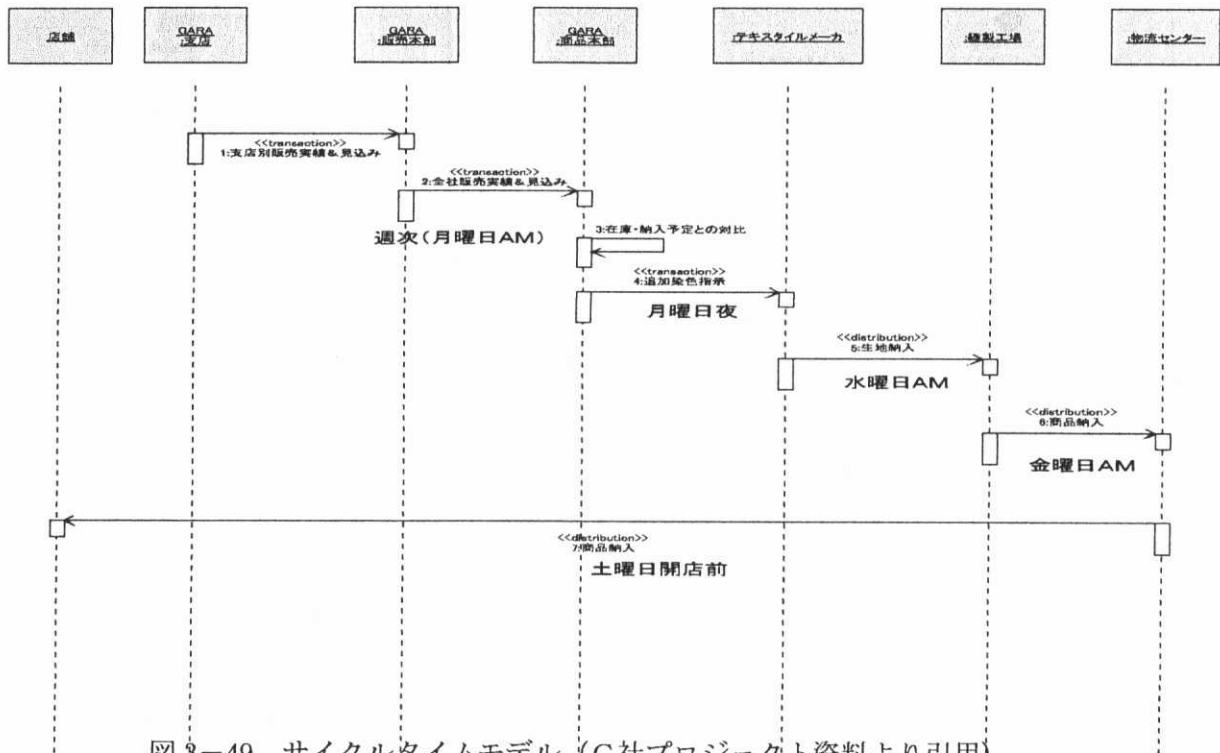


図 3-49 サイクルタイムモデル (C社プロジェクト資料より引用)

(5) 製品構成と価格モデル

製品は、知的資産の製品クラスで構成する。視点のレベルにより商品構成を表すものと商品の原価構成を明らかにするものがある。

a) 商品構成モデル

自社の商品クラスを定義し、その構造をクラス図で表現する。価格帯別にグループ化することでラインナップの可視化をする。(図 3-49 参照)

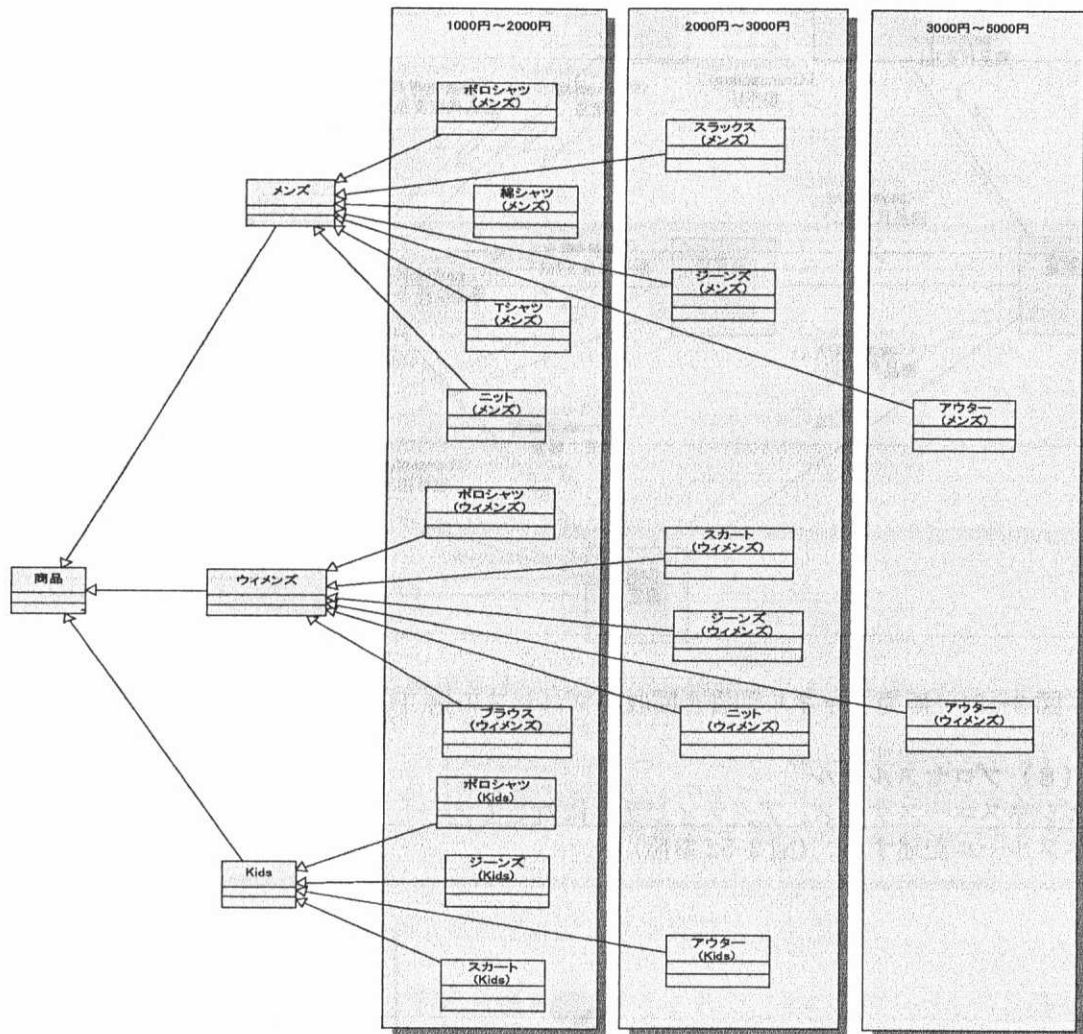


図 3-50 商品構成と価格モデル (C社プロジェクト資料より引用)

図 3-50 の商品構成では、商品は、メンズと女性用、子供用に分類でき、例えばメンズの場合は、ポロシャツ、綿シャツ、Tシャツ、ニット製品に別れ価格帯別に分類されている。

尚、部品構成モデルを作成し、部品のプロパティに価格を設けることができる。

(6) 組織と人材モデル

組織は、UML ビジネスモデルの組織ビューを構成する組織構造である。組織ビューはビジネスプロセスのUML表記を参照。

人材は、各組織を構成する人的資産 (ワーカー) で表す。また、人材のスキルセット、権限、能力、知識レベル、コンピテンシーをワーカーのプロパティとして設定する。人材の構成は、望

ましい人材構成があれば、商品構成と価格モデルのように人材オブジェクトをマッピング化し、現状との差異分析に利用する。

(7) 販売方法と資金回収モデル

顧客までの流通経路や販売方法と資金回収のモデルを明確にする。相互作用図を使用する。サプライチェーン振舞いモデルが物流に焦点を当てているのに対して、当モデルは資金回収に焦点を当てる。(図 3-51 参照)

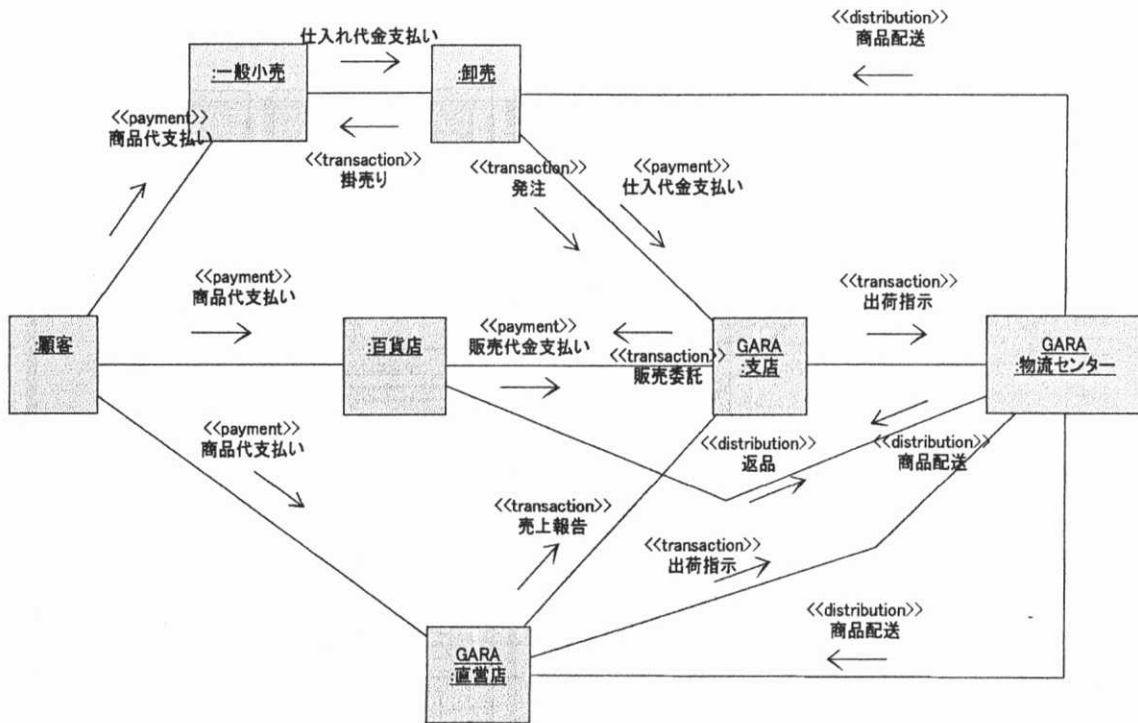


図 3-51 販売チャネル別資金回収のプロセスの例 (C社プロジェクト資料より引用)

(8) プロセスルール

ビジネスユースケース、アクティビティに対してノート (UML 標準に含まれる) を使用してビジネスルール記述する。(図 3-52 参照)

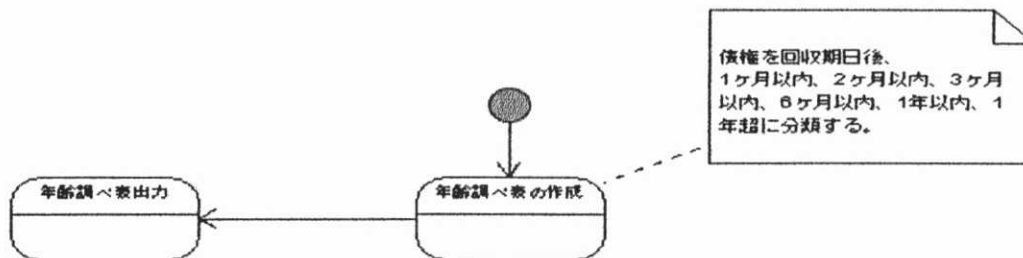


図 3-52 プロセスルールの例 (C社プロジェクト資料より引用)

2) ビジネスプロセスモデル表記の UML 拡張提案適応事例

UML によるビジネスプロセスモデルの基本的表記拡張の事例を示す。

(1) 目標ビュー (戦略マップ)

目標ビューでは、企業の目標および指標とそれらの因果関係、すなわち BSC の戦略マップをクラス図で拡張して記述する。

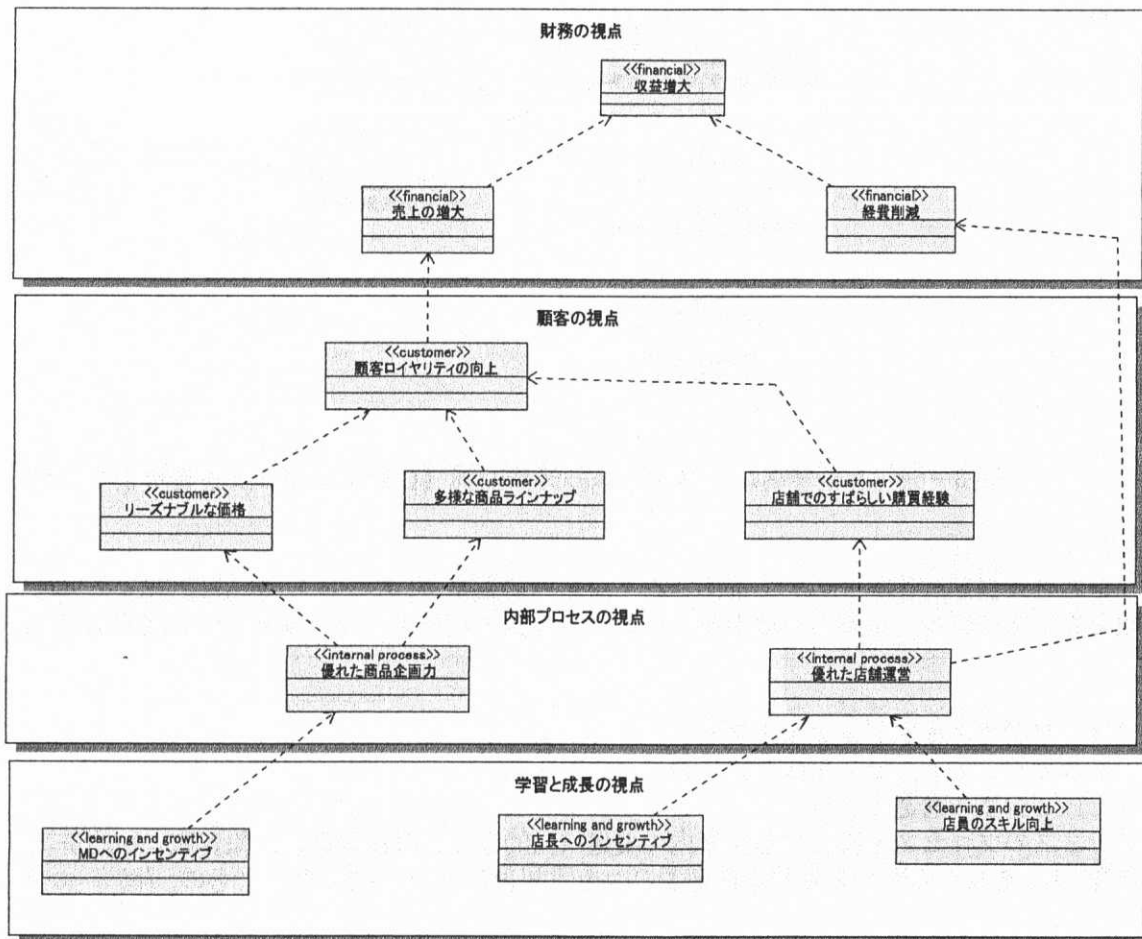


図 3-53 目標ビューの例 (C社プロジェクト資料より引用)

図 3-53 に示したようにクラス図で戦略目標を記述することで、その下位レベルの目標をクラスの属性に含めることで組織の下位への戦略目標の展開ができる効果もある。

(2) プロセスビュー

プロセスビューでは、ビジネスプロセスモデルを以下の 2 つより成る。

i) ビジネスユースケースモデル

ビジネスのプロセスの単位をユースケース (ビジネスユースケースと呼ぶ) で表し、プロセスが価値を産出する相手およびプロセスの実現に関わっている外部要素をアクターとして表したユースケースモデルを使用する。(図 3-54 参照)

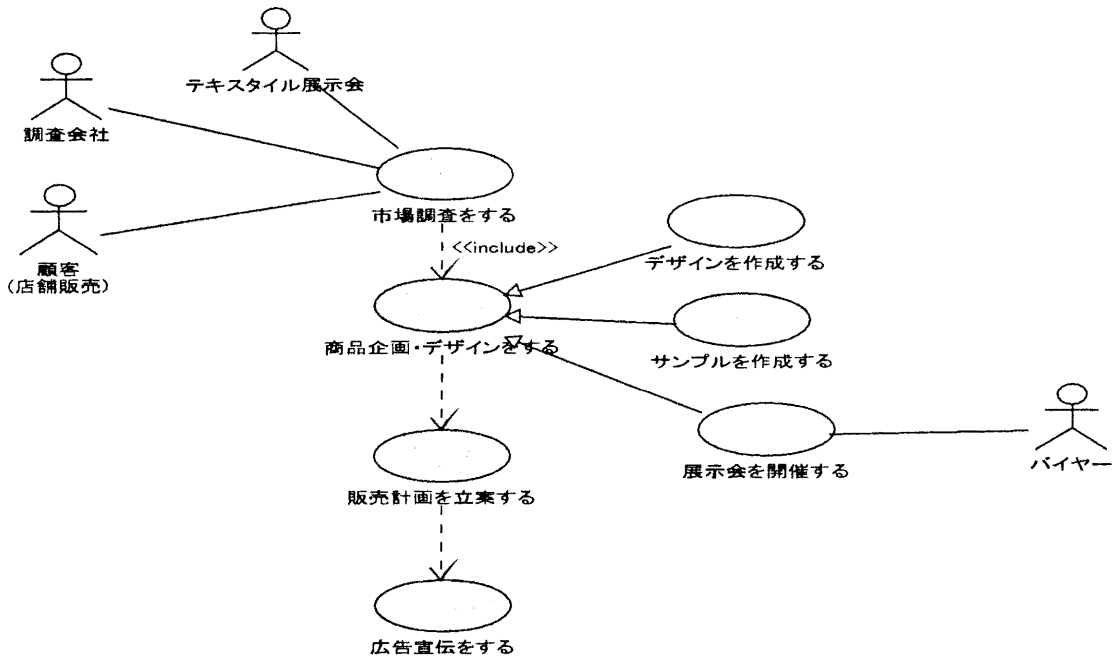


図 3-54 ビジネスユースケースモデルの例 (C社プロジェクト資料より引用)

ii) ワークフロー

ビジネスオブジェクトモデルで抽出された企業内の要素が、ビジネスユースケースを実現する手順を表したモデル。企業内の要素をレーンとし作業の流れをアクティビティ図で記述する。

(3) ビジネスオブジェクトモデル

ビジネスオブジェクトモデルは、以下の二つより成る。

i) ビジネス構造モデル

ビジネスのプロセスを実現する企業内の要素 (人的資産、財務資産、情報資産、知的資産、外部資産) と、それらの間の意味的な関係をプロセス単位にクラス図で表したもの。(図 3-40 参照)

ii) ビジネス振舞いモデル

各内部要素が、どう協調してプロセスを実現するかを相互作用図 (コラボレーション図) で表したもの。(図 3-41 参照)

(4) 組織ビュー

組織ビューでは、ビジネスのプロセスを実現する企業内の要素 (人的資産、財務資産、情報資産、知的資産、外部資産) を組織単位に整理したものである。(図 3-42 参照)

また、資産間の概念的関係をクラス図 (資産構造) として表す。(図 3-43 参照)

(5) ネットワークビュー

ネットワークビューでは、企業を構成する事業所や工場など物理的な要素の配置を配置図に表す。

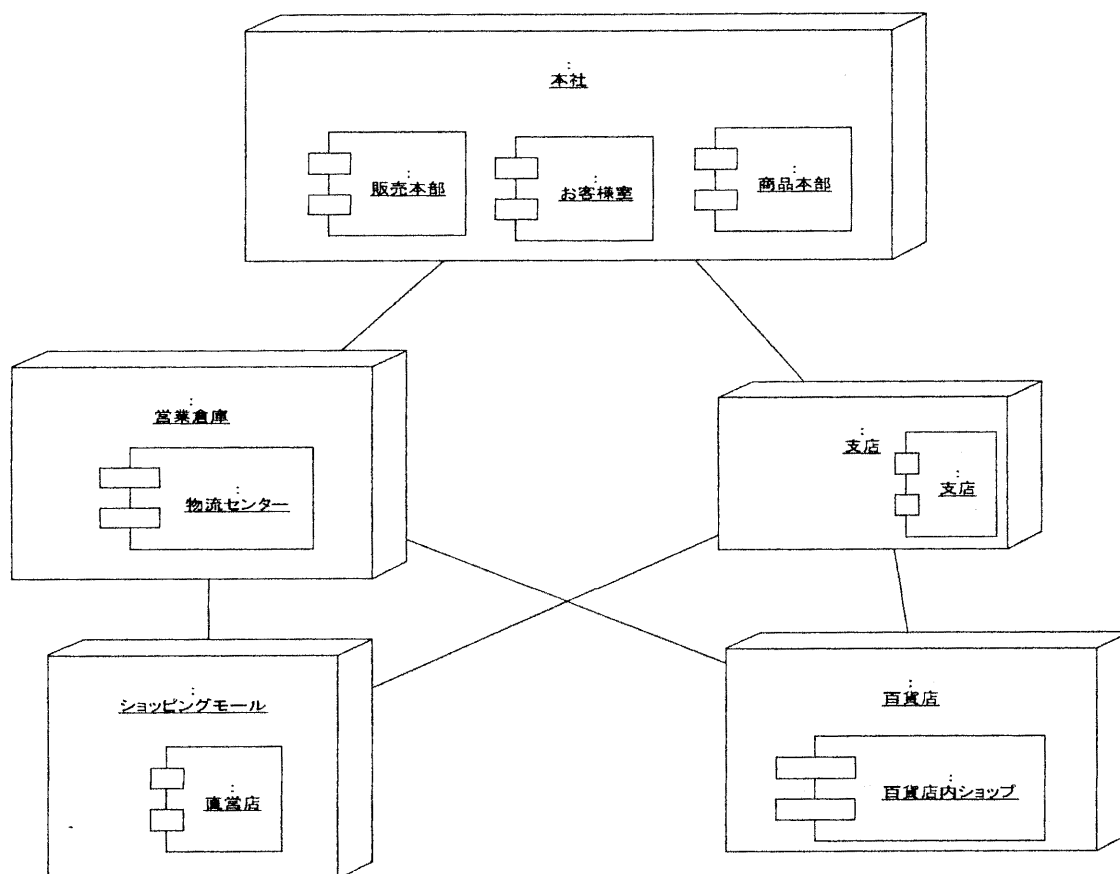


図 3-55 ネットワークビューの例（C社プロジェクト資料より引用）

図 3-55 のネットワークビューは、UML 用語のコンポーネントを使用して、企業の物理的な位置関係や取引関係を示すために使用できる。UML では、物理的なものはすべて、コンポーネントとしてモデル化でき、インターフェースを持つことができる。本研究では、コンポーネントを使用して企業間の本社、支店、倉庫、販売店の関係をネットワークビューで表した。

以上、UML によるビジネスモデル表記の提案の適用した事例を示した。

4-2 新提案の有効性検証基準

本節では、本論文の新提案を評価する方法を検討する。第3章のビジネスモデル表記と第4章のERP導入方法は、いずれも情報システム構築を前提としたプロジェクトの形態で実施されるので、情報システム構築のプロジェクト評価に準じた評価方法が適切と考える。

従来研究として、プロジェクトマネジメントやプロジェクト評価に関する研究を調査・分析した[146][147][148]。その結果、情報システムのプロジェクト評価基準としては、開発費用、開発期間、品質が主な評価基準となっている[149][150]。

本研究の提案に関しても従来研究と同様の評価基準により、下記の3項目を選択した。

- ・ 開発費用の差を比較する。
開発費用 = 投入する人数 × 人材単価 × 期間
尚、人材単価は、その人のスキルによって異なる。
- ・ 開発期間の差を比較する。
予定納期と開発リードタイムは、重要な指標と認識されている。
- ・ 品質及び変更への柔軟な対応性
品質のトラブルがなく、情報伝達の連続性が確保しており、変化への対応が早い。

4-3 新提案の検証結果

前節で提案したビジネスモデルを8つの視点から表記する方法と2つのUML拡張提案(ビジネスプロセス表記)について検証する。実際のアパレル企業C社のプロジェクト事例を基に機密事項などを加味して検証用に企業を設定し、具体的にアパレル業での事例で、本方法論に基づいて、ビジネスモデルをUML表記した。

この10の提案について評価を加える。本研究の評価にあたって、UMLに関する分野で日本では最も先進企業の1つである(株)オービス総研と共同研究を行い、本研究の提案を使用した実際のプロジェクトで、提案内容についての評価を調査した。可能な限り定量的な評価を試みた。また、定性的効果については、提案に関する有効性を情報システム構築の専門家を含め確認を行った。

(プロジェクト資料の詳細は、公開できないが、検証のため別途資料を保管している。)

1) ビジネスモデルの表記方法提案のまとめ

第1の拡張提案：サプライチェーンモデル表記の作成

サプライチェーンを表記することにより、自社の外部環境も表記できる。また、サプライチェーン構造モデルと振る舞いモデルにより、構成と構造といった静的モデルと動きや流れといった動的モデルが表記できた。

第2の拡張提案：概要ビジネスプロセスモデル表記の作成

企業ビジネスモデル表記は、クラス図を使用して主な機能グループの関係と組織との関係を表記できたことにより、企業の主要機能が理解しやすくなった。ビジネスプロセスモデルの表記は、プロセスビューにより、ビジネスユースケースで表記するプロセスモデルとワークフローの表記がされる。これにより、ビジネスユースケースでの主な必要機能が分かり、ワークフローで業務の流れが表記される。さらに、ビジネスオブジェクトモデルとしてビジネス構造モデルとビジネス振る舞いモデルを作成することにより、静的な構造と動的な個別の動きが理解できる。

第3の拡張提案：バリューチェーンモデル表記の提案

バリューチェーンのテンプレートにより、自社のコアプロセスを選択しやすくなった。また、コアコンピタンスをクラス図で関連を示すことにより、自社の強みを強調できるメリットがある。

第4の拡張提案：サイクルタイムモデル

情報の流れや出荷の流れに対して時系列の動きを示す。サプライチェーン構造モデルのクラス図を情報の流れと出荷タイミングによりシーケンスを付けることにより、シーケンス図ができる。

第5の拡張提案：組織と人材モデル表記の提案

組織は、組織ビューの視点で表記され、人材モデルは、組織ビューで表記でき、組織と人材の関係付けもされ、人材のスキル経験などは、ワーカーのプロパティで設定でき、有効である。

第6の拡張提案：製品構成と価格モデル表記の提案

製品は、知的資産の製品クラスで表記され、その構成もクラスで表記される。各製品のプロパティに価格を表記でき有効である。

第7の拡張提案：販売方法と資金回収モデル表記の提案

相互作用図を使用して流通経路や販売方法と資金回収のモデルが明確になる。サプライチェーン振る舞いモデルと同様、情報の流れと商流、金流、物流をメッセージのステレオタイプで表すことにより、詳細な情報も織り込める点は有効である。

第8の拡張提案：プロセスルール記述方法の提案

ビジネスユースケース、アクティビティに対してノートを使用して記述することにより、図示し難い事項を補足表現することができた。

2) UMLの拡張提案

第9の拡張提案：ビジネスモデル表記法として、基本構造表記と4つの視点で可視化する。

従来研究では、モデル化の視点をビジョン、プロセス、構造ビュー、振る舞いビューの4つを挙げており、ビジネスプロセス内の表記には適応できるが、ビジネスモデル表記には使用できない。本論文では、目標、プロセス、組織、ネットワークの4つのビューで俯瞰的にビジネスモデルを描き、その内容として基本構造表記のなかで構造ビューと振る舞いビューを描くことで、ビジネスモデルを明確にかつ不足なく表記できた。

第10の拡張提案：4つのビューの構成要素をステレオタイプで表示する。

各ビューのオブジェクトの構成要素を示す方法がなかったが、ステレオタイプを使うことで、構成要素が各オブジェクトで表示できる。尚、ステレオタイプを使う理由は、国際標準であるUMLの表記ルールあるステレオタイプを使用することで標準を逸脱しないためである。

以上、新提案に対する個別の評価を行った。

3) 定量的効果の検証

この提案を採用したC社のプロジェクトでの結果を従来方法と比較し有効性を確認した。検証の方法は、前節で述べた情報システム開発プロジェクトの基準として広く使用されている費用と期間を基準としている。従来方法と新提案での投入人数を同一人数とした場合、必要日数が費用と期間を示す基準となる。

表3-4は、新提案方法での作業日数と㈱オージス総研での従来方法での8事例平均との比較から検証したものである。この方法論を適用した場合、30%の効果が確認できた。

表3-4 UMLによるビジネスモデル表記方法適用事例による効果検証

フェーズ	作業項目	従来方法 日数	提案方法 日数	削減率 %
ビジネスモデリング	ビジネスモデル定義	5	5	0%
	コアコンピタンス、バリューチェーンの確認	5	4	20%
	ビジネスモデル表記などドキュメント資料作成	10	5	50%
ビジネスプロセスモデリング	ビジネス・プロセスのアセスメント	5	5	0%
	ビジネスプロセスの定義	10	8	20%
	ビジネスプロセス評価基準の定義	5	5	0%
	最適化の可能性のあるプロセスの決定	10	5	50%
	組織構造、ファンクション・モデルとプロセスの作成	10	5	50%
	合計	60	42	30%

次に新提案でのビジネスモデル表記を用いることにより、プロジェクトにおいて以下のような定性的効果を得た。

4) 定性的効果の検証

(1) ビジネスモデルの品質の確保

従来、この領域はコンサルタントの能力に負うところが大きく、個人により成果物やその品質にバラツキがある。本方法論では、ビジネスモデリングにおいて明らかにすべき内容、モデルすべき要素を経営の視点から業務の視点まで定義していることから、平均以上の成果物の品質を確保できるものと考えられる。

(2) システムモデルの品質の確保

システム開発において、業務要件を整理したコンサルタントがRFPを作成し、それをITベンダーが開発するということが一般的であったが、この間の情報伝達にどうしても漏れ、誤解が生じ、当初意図したとおりのシステムができていないということが常態化している。

UMLでモデル化すると正確に意味が伝わるため誤解が生じにくく、意思疎通が容易になり、エンドユーザと開発者間や関係者間の情報共有を強化できる。UMLは国際標準のモデル記述言語であるため、国内だけではなく海外も含めた広範囲なコミュニケーションが可能である。UMLを利用すれば、書き手の意図を読み手に正確に伝えることができ、文章ベースに比べ誤解が生じるリスクはかなり低くなると言える。UMLで記述することで、モデル化のルールが決まっているので、品質にばらつきが出にくい。これらは、ソフトウェアの品質の向上につながる。また、UMLは図示されることによって可読性が向上するため、属人的表記や文章ベースに比べてドキュメントの理解が容易になると言える。過去のシステムも理解しやすくなり、再利用や保守時に作業効率がよくなる。

加えて、UMLはオブジェクト指向開発をベースとしているので、オブジェクト指向言語（Java、C++など）との親和性が高い。UMLでモデル化した設計モデルと実装（プログラムコード）は直結しているため、モデル上でのデバッグも可能である。UMLモデリングツールを使用すれば、UMLモデルから実装の自動生成や実装からUMLモデルの逆生成が可能であるため、システム開発の効率が向上する。

(3) ビジネスモデルの維持管理の向上

従来のシステム開発では、ドキュメントを文章で記述したり、組織やプロジェクトごとに表記のルールが異なるなど、仕様書や設計書などのドキュメントが文章や独自のルールにもとづいた記述になっている。この結果、仕様を記述したドキュメントと実際に動いているプログラムの中身が乖離してしまい、ドキュメントの信頼性がなくなり、そのうちメンテナンスもされなくなる。このようなドキュメントの不備はシステムの品質にも影響することになる。

UMLを利用することで生産性が向上することを述べたが、同じ理由により、維持管理の工数削減になる。加えて、上述のようにビジネスモデルからシステムモデルへのシームレスな連携を担保しているため、ビジネスモデルの維持管理の向上も期待できる。

(4) 情報システムの変更容易性

昨今のビジネススピードが早い社会では、戦略変更やビジネスモデル変更が発生する確率が高く、情報システムの変更を伴うことが多い。こうした場合に情報システムの変更が容易にできる表記方法が求められる。UMLによるビジネスモデル表記法では、オブジェクト指向技術の特徴である「クラス」「ポリモーフィズム」「継承」を活用することにより、プログラム上の重複したロジックを排除し、必要な機能を整理された形になり、プログラムの変更と追加が容易になった。

注1) クラス: データとその操作手順であるメソッドをまとめるオブジェクトの雛形を定義し、同種のオブジェクトをまとめたもの。

注2) ポリモーフィズム(Polymorphism): 類似したクラスに対するメッセージの送り方を共通に

する仕組み。相手のクラスを意識せずに共通の方法でメッセージを送ることができる。

注3) 継承: クラスの共通点と相違点を整理する仕組み。上位のクラスは共通の性質を持ち、固有の性質をサブクラスで持つ仕組み。

4-3 考察

第3節では、ビジネスモデル表記方法を描く表記ツールとして、表記の後工程である情報システム構築の効率性も考慮してUMLを取り上げ、研究を行った。UMLに基づくシステム開発の効率性が高いことは広く認識されている。UMLはオブジェクト指向開発をベースとしているので、オブジェクト指向言語（Java、C++、C#など）との親和性が高い。UMLでモデル化した設計モデルと実装（プログラムコード）は直結しているため、モデル上でのデバッグも可能である。UMLモデリングツールを使用すれば、UMLモデルから実装の自動生成や実装からUMLモデルの逆生成が可能であるため、システム開発の効率が向上する。

本研究において多くのUML表記を行ったが、この作成にあたっては、UML表記用のソフトウェアを使用した。UMLでは、いくつかのモデル要素と図を使って、いろいろな視点からシステムの構造を表現できる利点がある。互いに関連し合っているモデル要素は、どれかを変更すれば関連するモデル要素も変更して整合性を保証する必要がある。UMLモデリングツールを使えば、モデル要素を変更したときも関連するモデル要素を自動的に変更して整合性が取れる。

尚、UMLを使ったシステム開発では、UMLモデリングツールを使用することがほとんどであるが、これは、手書きよりモデル記述の効率がよいことや、モデルから実装の自動生成や逆生成の機能があり、変更などにおいても、モデル間の整合性が取れる機能を持つなどシステム開発の効率化と品質の向上につながるからである。また、UML自体は固有のツール製品に依存しないモデル記述言語なので、ツール製品を変更してもモデルの記述はそのまま使用ができる長所がある。

第5節 第3章の結論

第3章では、ビジネスモデル設計（第2章）からの課題であったビジネスモデル構築、実施の手段である情報システムへ効率よくシームレスな連携を可能にするビジネスモデル表記方法を確立することを研究目的とした。

ビジネスモデルでビジネスを可視化するためのビジネスモデル表記方法に使用するツール、表記形式の検討を行い、代表的なツールである ARIS, IDEF, PDR, UML の4種類を分析検討した結果、UMLを採用した。その理由としては、オブジェクト指向による情報システムへのシームレスな連携が期待できること。及びUMLの持つ複数の表記方法により、ビジネスモデルが複数の視点により可視化できることである。

1) 提案内容

本研究では、ビジネスモデルを表すには、多面的な複数の種類が必要であるとの考えから、ビジネスモデルの内容を明確に書き示すための視点として8項目のビジネスモデルの表記種類を提案し、2項目のUML拡張を提案した。本章での新提案の10提案の項目を下記に示す。

ビジネスモデル表記に関する視点別8つのサブモデル提案を以下に記す。

第1の表記提案：サプライチェーン構造モデルの提案

第2の表記提案：概要ビジネスモデルとビジネスプロセスモデルの提案

第3の表記提案：コア・コンピタンスモデルの提案

第4の表記提案：サイクルタイムモデルの提案

第5の表記提案：組織と人材モデルの提案

第6の表記提案：製品構成と価格モデルの提案

第7の表記提案：販売方法と資金回収モデルの提案

第8の表記提案：プロセスルール記述方法の提案

提案内容の主な項目を示すと以下の表3-8となる。

表3-5 ビジネスモデル表記のため提案とUMLの拡張

提案項目<表記提案>	使用UMLダイアグラム	提案内容
1. サプライチェーン構造モデル	ビジネスオブジェクトモデル図 (クラス図)	バリューチェーンプロセステンプレートに基づきグループ化
サプライチェーン振る舞いモデル	コラボレーション図	商流、金流、物流をあらわすメッセージのステレオタイプを設定
2. 概要ビジネスプロセスモデル	クラス図	バリューチェーンプロセステンプレートに基づきグループ化
3. コア・コンピタンスモデル	クラス図	コアプロセス部分を強調表示
4. サイクルタイムモデル	シーケンス図	情報の流れや出荷の流れを時系列に並べシーケンス図でサイクルタイムを示す
5. 組織と人材モデル	ビジネスプロセスの組織ビューと資産ビュー (人的資産)	人材のスキルセット、権限、能力、知識レベル、コンピテンシーをワーカーのプロパティとして設定
6. 製品構成と価格モデル	クラス図	部品クラスのプロパティに価格を設ける
7. 販売方法と資金回収モデル	コラボレーション図 (金流を中心に記述)	金流をあらわすメッセージのステレオタイプを設定
8. プロセスルールの記述	ビジネスユースケース	アクティビティに対してノートを使用して記述

表 3-5 を補足すると、提案 1 から提案 8 は、新しいコンセプト（視点）による UML を使用したビジネスモデル表記法の提案である。

UML 表記の拡張 2 提案を以下に記す。

第 9 の拡張提案：ビジネスモデル表記法として、基本構造表記と 4 つの視点で可視化する。

ビジネスプロセスをクラス図を使用して 4 つのビュー（目標・プロセス・組織・ネットワーク）で表す。

第 10 の拡張提案：4 つのビューの構成要素をステレオタイプで表示する。

構成要素のクラス図を使用してステレオタイプを定義する。

提案の 9 と提案 10 は、UML の基本的表記に関する UML 拡張提案である。

提案した理由は、以下のとおりである。

- ・ ビジネスモデルを UML により表記する場合、ビジネスプロセスの基本構造表記方法を拡張することにより、明確なビジネスプロセスの表記が可能になる。
- ・ ビジネスモデルを可視化する視点として 8 つの視点から UML による表記方法を提案することにより、ビジネスモデルの構造が複数の視点から理解でき、且つ、表記方法の改良でモデルの具現化ができ理解しやすくなる。

2) 提案による定量的効果

実際の企業でのプロジェクトにおける定量的効果として、この提案を採用した C 社のプロジェクトでの結果を従来方法と比較し有効性を確認した。この方法論を適用した場合、従来方法の工数比較で 30% の効果が確認できた。

その内訳は、以下の通りである。

- | | | | |
|--------------------|-----------|--------|--------|
| ・ ビジネスモデリングにおける効果 | 従来平均 20 日 | → 14 日 | 30% 削減 |
| ・ ビジネスプロセスモデリングの効果 | 従来平均 60 日 | → 42 日 | 30% 削減 |

ビジネスモデル表記の新提案に対する検証は、第 4 節で事例適用により確認できた。評価に関する追記を以下に記述する。

ビジネスモデルを UML で表すことにより、情報システム化時に再度、業務分析作業、システムモデリングの作業の発生が軽減される。これは、工数削減効果だけでなく、業務要件に関する情報ロスを防ぐことになる。ビジネスモデリングとシステムモデリング間の情報ロスが、テスト段階での仕様変更発生の主要原因になっていると考えられている。ビジネスモデルからシステムモデルの連続性を確保することにより、設計の早い段階での仕様確定を実現する。テスト段階で発覚した仕様変更に伴うコストは、要求定義時の 1.5 倍から 3.0 倍になると言われていることから、このコスト削減・工数削減効果は大きい。

3) 定性的効果の検証

(1) ビジネスモデルの品質の確保

ビジネスモデリングにおいて明らかにすべき内容、モデルすべき要素を経営の視点からと業務の視点まで定義し、UML 表記により、成果物の品質を確保できた。

(2) システムモデルの品質の確保

UML でモデル化すると正確に意味が伝わるため誤解が生じにくく、意思疎通が容易になり、エンドユーザと開発者間や関係者間の情報共有を強化できる。UML は国際標準のモデル記述言語であるため、UML で記述することで、品質にばらつきが出にくい。これらは、ソフトウェアの品質の向上につながる。UML モデリングツールを使用することにより、UML モデルから実装の自動生成や実装から UML モデルの逆生成が可能であるため、システム開発の効率が向上する。

(3) ビジネスモデルの維持管理の向上

UMLを利用することで仕様書や設計書などドキュメントの生産性が向上することを述べたが、同じ理由により、維持管理の工数削減になる。加えて、上述のようにビジネスモデルからシステムモデルへのシームレスな連携を担保しているため、ビジネスモデルの維持管理の向上も期待できる。

(4) 情報システムの変更容易性

昨今のビジネススピードが早い社会では、戦略変更やビジネスモデル変更が発生する確率が高く、情報システムの変更を伴うことが多い。こうした場合に情報システムの変更が容易にできる表記方法が求められる。UMLによるビジネスモデル表記法では、オブジェクト指向技術の特徴である「クラス」「ポリモーフィズム」「継承」を活用することにより、プログラム上の重複したロジックを排除し、必要な機能が整理された形になり、プログラムの変更と追加が容易になった。

本研究での結論として、本研究で提案したUMLによるビジネスモデル表記方法は、有効である。従来のIDEFなどのビジネスモデル表記ツールと比較して、UML表記に本論文での10項目の新提案を取り入れたUMLによるビジネスモデル表記方法は、有効であることを実証した。UMLによるビジネスモデル表記方法で可視化したビジネスモデル表記することは、情報システム構築の前段階であるビジネスモデリングの作業を効率化できた。検証方法として、事例検証を行い効率化と品質の向上を確認でき、UMLによるビジネスモデル表記方法は、有効な手段である。

本研究の提案であるビジネスモデルを8項目の可視化方法で表現することにより、ビジネスの形態を明確することができた。従来、システム開発におけるモデリングは、対象となる業務のうちシステムが適用される部分についてのみ行われるのがこれまで通例であった。このため、システム側からはビジネスの全体を把握できないという問題がこれまで指摘されている。一方、経営の立場から様々な視点からビジネスを表現するためのアプローチがなされているが、基本的には、単一目的の表記法であり、ビジネスや会社全体のシステム化を前提に表現しようというものがなかった。今回の試みは、経営の立場からアプローチされてきた表記内容をUMLによって表記の統一化を図り、しかもシステムへの連携を考慮していることに価値がある。この結果、事業要件と実際の企業プロセスとの関連性をモデルとして管理できるようになり、経営の高度化、いわゆるモデルベース経営への移行も可能になる。

第3章での研究の新規性を以下に箇条書きする。

- ・ビジネスモデルを表記するツールや方法論を比較検討し、表現方法の多彩性と後工程の情報システムとの連携の高さから、UMLを使用することを採用し、その有効性を確認した。
- ・ビジネスを可視化するビジネスモデルには、複数の種類が必要であり、表記する方法として、8種類の視点による表記UMLによるビジネスモデルの表記法を考案した。
(8つの表記方法提案)
- ・UMLでの表記についての従来の標準的表記に加え、拡張提案により、ビジネスプロセスモデル表記が可能となった。(2つのUML拡張提案)

このビジネスモデル表記の後工程として、ビジネスモデリング、情報システム構築へと繋がっていくが、UMLの表記により、オブジェクト指向によるビジネスモデルから情報システム開発へのシームレスで効率的な方法が確立されることが期待できる。第4章では、ビジネスモデル表記方法を活用した効率的な情報システム構築の提案を行う。