

# E-Business 開発モデルの提案と評価

2000MT055 本山高之, 2000MT068 西尾晴仁

指導教員 青山幹雄

## 1. はじめに

複雑な E-Business システム[1]を Web サービス[2]を用いて開発するための E-Business 開発モデルを提案し、図書システムの開発を例として提案した開発モデルを評価した。提案した E-Business 開発モデルはソリューションダイアグラムを使用する方法と UML[3]を使用する方法の二通りで、この開発モデルを用いることでビジネスの内容をあらかじめ用意されているビジネスパターンに分類し、ビジネスプロセスを明確に定義できる。ソリューションダイアグラムを用いたよりも UML を用いた方がビジネスパターンとビジネスプロセスの両方が明確に定義できることが明らかになった。また、これらを組み合わせた方法として、顧客要求の際にソリューションダイアグラムを、ビジネス分析の際に UML を用いることで、複雑な E-Business システムを柔軟に開発する方法を提案した。

## 2. E-Business 開発モデルの提案

### 2.1. E-Business システムを実装する問題点

企業間の柔軟な連携を実現するためには、システムがオープンで動的であることが必要である。「オープン」とは、OS・プラットフォームに依存しなくて相互運用性が高いという意味である。また、「動的」とは、システム開発時だけでなく、運用時にも接続先を柔軟に変更できるという意味である。そこで、ビジネス環境の変化に柔軟に対応でき、動的な業務連携を可能にすることを満たす E-Business システムを開発するために、ビジネスパターンおよび BPEL からの視点で E-Business 開発プロセスを提案する。

### 2.2. E-Business 開発モデルで使用する技術的背景

#### 2.2.1. E-Business のパターン

E-Business のパターン[4]とは、ソフトウェア分野で活用されるパターンを分類し類型化したものである。パターンを用いると、複雑なビジネス問題を理解して分析でき、またそれをより小さな理解しやすい機能にまで分解できる。本研究では次の 2 つの E-Business パターンを用いる。

表1 E-Business パターンについて

パターン名	意味
ビジネスパターン	ビジネス目標を達成するためのもの
統合パターン	ビジネスパターンを実際のビジネス要件に合致するように組み合わせるもの

### 2.2.2. ビジネスプロセス

ビジネスプロセスとは、あるビジネスを実現するためのアクティビティとその間の実行関係である。複数の企業が提供する Web サービスを組み合わせると、どのようなビジネスプロセスにも柔軟に対応できる仕組みが必要になる。

### 2.2.3. BPEL4WS

BPEL4WS(以下 BPEL と略記)[5]は、XML をベースとした Web サービスのフロー記述言語である。BPEL を使用することで、Web サービスの起動、データの操作、障害の通知、プロセスの終了などの様々なアクティビティを記述し、それらを結び合わせて、複雑なプロセスを定義できる。

### 2.3. E-Business 開発モデルの提案

我々は複雑な E-Business システムを柔軟に開発するために 2 通りの E-Business 開発モデルを提案した。E-Business 開発モデルはビジネスの内容をビジネスパターンに分類し、ビジネスプロセスを明確に定義するためのモデルである。

#### 2.3.1. E-Business 開発モデル1

E-Business 開発モデル1は、図1に示すようにソリューションダイアグラムを用いてパターンとプロセスの定義をするものである。

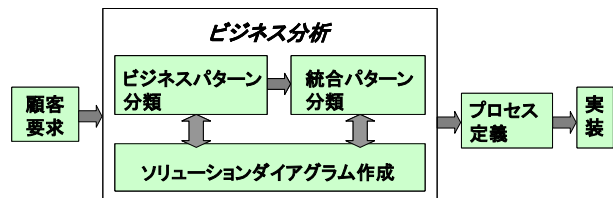


図1 E-Business 開発モデル1

表2 E-Business 開発モデル1の手順

手順	内容	
1. 顧客要求の定義	ビジネス記述を用いて定義	
2. ビジネス分析	ソリューションダイアグラム作成	ビジネス記述からアクターとビジネス機能を導出し、ソリューションダイアグラムを作成する
	ビジネスパターン分類	ソリューションダイアグラムにかかれたビジネス機能をもとにビジネスパターンに分類する
	統合パターン分類	ソリューションダイアグラムにかかれたビジネス機能同士のつながりをもとに統合パターンに分類する
3. ビジネスプロセス定義	システムの処理内容からビジネスプロセスをBPELを用いて定義する	

2.3.2. E-Business 開発モデル2

E-Business 開発モデル 2 は、図2に示すように顧客要求の定義を行い、UML ダイアグラムを作成し、ビジネス分析を行う。

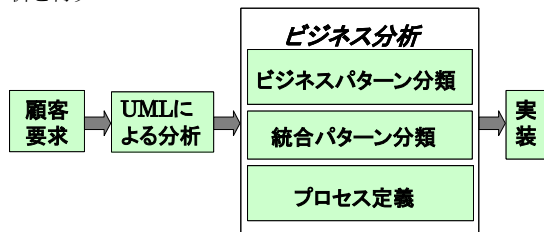


図2 E-Business 開発モデル2

表3 E-Business 開発モデル2の手順

手順	内容	
1. 顧客要求の定義	ビジネス記述を用いて定義	
2. システム分析	UML ダイアグラムを使用してシステム分析	
3. ビジネス分析	ビジネスパターン分類	ユースケース図とクラス図よりビジネスパターンに分類する
	統合パターン分類	クラス図より統合パターンに分類する
	ビジネスプロセス定義	アクティビティ図とステートチャート図よりビジネスプロセスを考え、BPELで定義する

3. 図書システムへの適用

本研究では図書システムの開発を例にして、提案した開発モデルを適用した。図書システムの概要を以下に示す。

- (1) 図書システムを使用するには、利用者は利用者登録をしなければいけない。
- (2) 図書システムの機能として、「図書の貸し出し」、「図書の検索」、「図書の返却」があり、利用者は利用者登録をすることでそれらの機能を利用できる。

3.1. E-Business 開発モデル1の適用結果

E-Business 開発モデル1の手順をもとに作成した図書システムのソリューションダイアグラムを以下に示す。

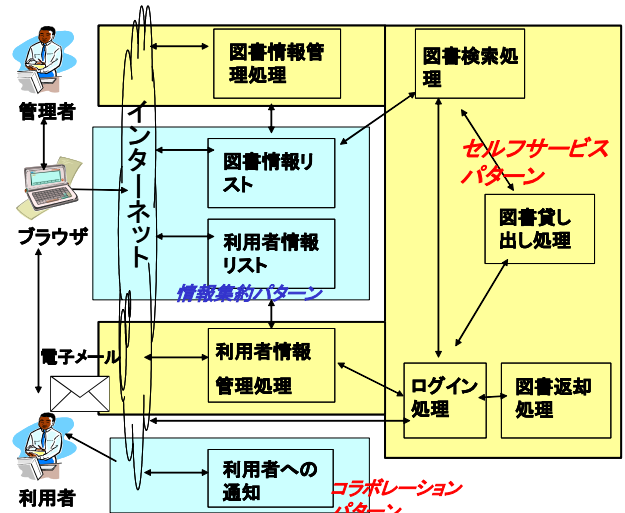


図3 図書システムのソリューションダイアグラム

ソリューションダイアグラムをもとに、表 4 に示すようにビジネス機能をビジネスパターンに分類した。

表4 ビジネスパターン分類

ビジネス機能	ビジネスパターン
図書貸し出し処理	セルフサービス
図書情報管理処理	セルフサービス
図書情報リスト	情報集約
利用者への通知	コラボレーション
利用者情報管理処理	セルフサービス
利用者情報リスト	情報集約

3.2. E-Business 開発モデル2の適用結果

E-Business 開発モデル2の手順をもとにビジネス分析を行った結果を以下に示す。

表5 UMLを用いたビジネスパターン分類

	ユースケース図	クラス図
図書貸し出し処理	セルフサービス	セルフサービス
図書情報登録処理	セルフサービス	/
	情報集約	
図書情報管理処理	/	セルフサービス
図書情報リスト	/	情報集約
ログイン	セルフサービス	セルフサービス
ユーザ認証	コラボレーション	コラボレーション
ユーザ情報変更	セルフサービス	/
	情報集約	

個人情報管理処理		セルフサービス
個人情報リスト		情報集約

#### 4. 評価

図書システムを例に2通りの E-Business 開発モデルの評価を行う。

##### (1) 開発モデルの特徴

図書システムに適用した場合にわかった、それぞれの開発モデルの特徴を挙げる。

##### (2) 有効情報量からみた開発モデルの評価

パターンとプロセスを決定に結びつく図の有効情報量を求める。ここで、図の有効情報量は式(1)のように定義する。式(1)で「図に描かれている全要素数」とは、使用した図に描かれるそれぞれの要素を数え上げたものである。

##### 図の有効情報量(%)

$$\frac{\text{パターン・プロセスの決定に結びつく要素数}}{\text{図に描かれている全要素数}} \times 100 \quad (1)$$

#### 4.1. 2つの開発モデルの特徴

##### 4.1.1. E-Business 開発モデル1の特徴

ビジネスパターン・統合パターンの分類とビジネスプロセスの定義をする上での判断材料は、最初に考えるビジネス記述とソリューションダイアグラムである。ソリューションダイアグラムにより、全体の機能、処理の関連の制約や関係を捉えることができ、ビジネスパターンの分類が容易に行えることが考えられる。しかし、ビジネスプロセスの定義を行うには、ソリューションダイアグラムのみでは処理の流れが明確ではないので、正確な定義が行えない。

##### 4.1.2. E-Business 開発モデル2の特徴

E-Business 開発モデル2の特徴は、UMLを用いることでシステムの分析が明確に行えることである。その中で、ユースケース図とクラス図からビジネスパターンに分類できた。また、シーケンス図、アクティビティ図から処理の流れが容易に理解でき、ビジネスプロセスの定義も容易に行えた。さらに、ステートチャート図により、正確な定義を考えられた。UMLによるダイアグラムの作成は、特にビジネスプロセスの定義に有効であると考えられた。

##### 4.2. 有効情報量からみた開発モデルの評価

2通りの E-Business 開発モデルで使用した図に対する有効情報量を表6に示す。

表6 E-Business 開発モデルの有効情報量

	E-Business 開発モデル1		E-Business 開発モデル2	
	使用した図	有効情報量	使用した図	有効情報量
ビジネスパターン	ソリューションダイアグラム	28.2%	ユースケース図	42.3%
			クラス図	50.0%
統合パターン	ソリューションダイアグラム	28.2%	クラス図	50.0%
ビジネスプロセス	なし	0%	アクティビティ図	100%
			ステートチャート図	100%

E-Business 開発モデル2においてビジネスパターンの分類には曖昧性が生じた。その原因を有効情報量の視点から見て E-Business 開発モデル1でのビジネスパターンの有効情報量と比べてみる。

有効情報量の少ないソリューションダイアグラムでは一つのビジネス機能に対して一つのビジネスパターンに当てはまることができたが、有効情報量の多いユースケース図はそうではなかった。その理由としてユースケース図の場合は、システムの外部機能しか着目しておらず、内部でどのような処理をしているかは判断できないものになっている。

一方、ソリューションダイアグラムの場合は、ビジネス記述を前提にアクターとビジネス機能を決定してそれらを図示していくもので、そのビジネス機能は外部機能と内部機能の両方に着目している。その点から判断すると、仮に数値が少なくても明確な分類ができたのは、情報の質にも関わってくると言える。そう考えると、ユースケース図ではビジネス機能に分類する際に、どのビジネス機能がどのビジネス機能と連携し、どのような処理を行うかという詳しい情報は得られないことが明らかになる。一方、クラス図の場合は内部構造も理解できる。つまり、ビジネスパターンの分類にあたっては、外部機能に着目するほかに、あいまいな分類になった場合には内部機能にも着目する必要があると考える。

##### 4.3. 評価を通じて考えた新しい E-Business 開発モデル

E-Business 開発モデルの特徴は、利用者の視点に立つてシステムを考える上で効果的な開発モデルであるといえる。しかし、システムを構築することに主眼を置くと、利用者だけの視点で考えても柔軟な開発は困難といえる。また、E-Business 開発モデルはシステムの内部構造を明確に理解しにくいといえる。つまり、利用者だけでなく、開発者がシステムを構築する上でビジネスパターンの分類とビジネス

プロセスの定義を明確にできるための手段を考えることが必要である。

E-Business 開発モデル 1, 2 を組み合わせた、さらに有効的な E-Business 開発モデルを提案した。

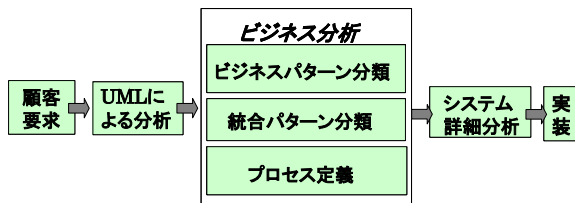


図3 新たに定義した E-Business 開発モデル3

表7 E-Business 開発モデル3の手順

手順	内容	
1. 顧客要求の定義	ビジネス記述を用いて定義し、ソリューションダイアグラムを作成	
2. システム分析	UML ダイアグラムを使用してシステム分析	
3. ビジネス分析	ビジネスパターン分類	ユースケース図とクラス図よりビジネスパターンに分類
	統合パターン分類	クラス図より統合パターンに分類
	ビジネスプロセス定義	アクティビティ図とスタートチャート図よりビジネスプロセスを考え、BPEL で定義
4. システム詳細分析	アプリケーションパターンに分類	

再定義した E-Business 開発モデル3の特徴は、E-Business 開発モデル2をベースに、顧客要求の定義とシステム詳細分析の2点を付加したものである。

(1)顧客要求の定義にソリューションダイアグラムを利用する  
顧客要求の際にビジネス記述に加えてソリューションダイアグラムを用いた。その理由は顧客要求の際に視覚的に理解ができる図を作成したほうが、顧客側にも理解できる。また、システム分析の際に明確な分析ができる。

(2)システム詳細分析

また、システム詳細分析を付加した。開発モデル1, 2 は主に、設計の前段階に絞って考えたものなので、実際に実装を行う上ではさらにシステムの詳細な分析をする必要がある。従ってアプリケーションパターンにも分類することで、ビジネス分析で行ったパターンとプロセスの分類が、実際の機能に生かされるものになる。

## 5. 考察

複雑な E-Business システムを実装するには、ビジネ

スの内容毎にあらかじめ用意されているビジネスパターンに分類し、ビジネスの処理を明確に定義する、という開発モデルを実際の E-Business システムに適用していくことで、複雑な E-Business システムを柔軟に開発できる。

パターンを用いて、ビジネスパターン、統合パターンによって分類していくことにより、細かい部分の制御までビジネスプロセスの定義を行えるので、どのような制約や関係を記述していくのか理解できる。既存のサービスを利用して新規のサービスを開発する際にも、どの部分を組み換えていけばよいか、パターンを用いることで柔軟に行えると考えられる。

また、ビジネスプロセスの定義の際に BPEL を用いることで、明確にシステム、パートナー間の処理の連携が行い理解することができ、順序制約や依存関係などを正確に定義することができた。SOAP, WSDL, UDDI の Web サービス技術だけではフローに関しては正確にできず、BPEL が有効であると考えられる。

しかし、BPEL を用いたビジネスプロセスの定義において、複雑な処理には複雑な相互作用が必要となる。BPEL では内部処理のフローが中心で複雑な相互作用はあまり扱っていない。より複雑な処理になった場合に、複雑な相互作用が正確に定義できる問題がある。パートナーと Web サービス間の相互作用をより便利にするものを提供することが今後の課題ではないかと考えられる。

## 6. まとめ

本研究で提案した E-Business 開発モデルは、柔軟な開発を行う方法を、効率的に、有益に進めるものとして提供している。そして、このモデルを基礎として、さらによりよく改良したモデルを開発し、複雑な E-Business システムを柔軟に開発する方法を提案することを、我々は考えていく。

## 参考文献

- [1] E. Turban (ほか):e コマース電子商取引のすべて、ピアソンエデュケーション (2000).
- [2] 日本アイ・ビー・エム irstart チーム:最新 Web サービスがわかる、技術評論社 (2002).
- [3] 竹政 昭利:はじめて学ぶ UML, ナツメ社 (2003).
- [4] J. Adams (ほか):Web システムのデザインパターン, 翔泳社 (2003).
- [5] T. Andrews, et al. Specification Business Process Execution Language for Web Services Version 1.1: <http://www-106.ibm.com/developerworks/library/ws-bpel/>