

連携するシステムを構築するための開発方式の提案と検証

大澤 貴彦[†] 片岡 信弘[†]

東海大学工学研究科電子工学専攻[†]

1. はじめに

情報システムの開発では、複雑になった組織の構造を体系的に理解して改善することが重要である。しかし、近年の多くの組織は業務の分散化により、各業務がバラバラに管理され運営されていることにより、組織の全体業務を把握できている人はほとんどいない。そのため、他の業務と連携するシステム開発は困難とされており、情報システムそのものが業務のネックになりかねない。

本研究でビジネスプロセス・モデリングを行い、組織全体のビジネスプロセスモデルを大まかに記述して組織全体業務の把握と改善を行い、それぞれの業務を詳細なプロセスモデルへと分割して各業務内容を把握する。それから、ビジネスプロセスとシステムとのギャップを埋めるためのプロセスモデルを記述して、それを UML へと変換する流れの開発方式の提案を行う。これにより、それぞれ分散された業務を集中しての運営が行え、業務間の連携も容易になる。

2. 研究背景

2.1 システム開発の現状

近年の情報システムの開発では、プロジェクトの 8 割近くが失敗しているといった調査報告も出ている。また、業務提携に伴うシステム統合によるシステム障害の発生といった連携による問題事例も多く報告されている。

そこで本研究では、システム開発に関わる、認識の違いや不目的な開発などの問題が起こる『要求定義の曖昧さ』とビジネスプロセスやデータがシステム内で完結していることによる『システム連携の困難さ』と組織の分散化により組織全体を把握できていないことによる『業務改善の困難さ』の 3 つの問題点を挙げ、この問題を解決する方法としてビジネスプロセス・モデリングの活用に着目した。

2.2 ビジネスプロセス・モデリング

ビジネスプロセス・モデリング (BPM) は従来曖昧とされてきたビジネスプロセスを可視化して、その業務の改善を行うことを目的とされた方法である。さらに近年では、BPM によって出されるビジネスプロセスが組織で利用される情報システムの目的であることにより、システム開発に活用することが注目されている。

ビジネスプロセスモデルの記述は、ひとつ以上の目的を達成するためのある情報にかかわる業務や活動を現すファンクション (機能) と機能を起動させるためのトリガーであり、また機能の結果であるイベントによって組み合わせる。

3. 提案内容

3.1 開発手順

連携するシステムを開発するために、以下の手順で開発する。

- (1) 組織レベル : 組織全体を大まかに把握する
- (2) 管理レベル : 業務内容を詳細に把握する
- (3) 技術レベル : ビジネスプロセスとシステムとのギャップを埋める
- (4) UML : 技術レベルから UML に変換する
- (5) 実装システム : UML からシステム開発
- (6) 運営 : 開発したシステムを経営レベルに従い導入・運営を行う

3.1.1 経営レベル

このレベルでは、分散化された組織をまとめることで、組織全体の業務を把握して、それぞれの業務間での連携をスムーズに行えるようにすることを目的とする。

記述手順は、組織を外から見たときの外面多岐な業務をすべて書き出し、その業務の開始イベントと終了イベントを決める。次に、業務に関わる部署とその役割を決め、各役割に対し開始と終了のイベントを記述する。後は、業務全体の開始イベントから組織の運営が効率よく行えるように組み合わせることで終了イベントになるようにし、業務全体の把握と改善を行う。

A proposal of development method to construct cooperated system

[†] Takahiko OHSAWA, Nobuhiro KATAOKA
Tokai University

3.1.2 管理レベル

このレベルでは、各業務内容と扱うデータの属性を詳しく記述し他の業務との無駄や矛盾を解決させ、専門業務の改善を行うことを目的とする。

記述手順は、経営レベルの業務の一つを最初と最後のイベントをあわせて、その業務に関わる人や権限までも詳細に記述する。また扱うデータに関しては、属性まで詳しく記述する。

各プロセスモデルの記述が完了したら、それぞれのプロセス間の連携方法やプロセスの重複や不足・不要データがないかをチェックして、業務上の無駄や矛盾を解消させる。

3.1.3 技術レベル

このレベルは、これまでに記述したプロセスモデルとシステムとのギャップを埋めて、システムを実装させることを目的とする

管理レベルのモデルをシステム実装につなげるために、1 つのシステムになるプロセスをまとめる。また DB や XML ファイルといったシステム内で扱うものなどのアクセスやトランザクションの開始と終了などを詳細に記述する。さらに、人だけが行うようなシステムと関係の薄いことはここでは省略し、そのサポートとなるような機能を追記する。

3.1.4 UML 変換

従来、別々に扱われていた BPM と UML を結びつける必要がある。技術レベルで記述したモデルから UML へと以下の手順で変換する。

- (1) システム化する業務部分ごとに、どんなクラスや属性が必要かを決定して関連させる。
- (2) プロセスモデルからクラスを抜き出す。
- (3) クラス図のクラス間を接続する。
- (4) クラス図に属性、ロール名、多重度、関連名を記述する。

3.2 販売プロセスによる試作

提案方式を、検証するために Fig.1 に示す流れで販売業務にかかわるシステムを試作した。

今回の開発では、『受注システム』『在庫管理システム』『出庫管理システム』の販売業務の3つシステムを開発した。開発したシステムは、経営レベルのビジネスプロセスモデルに従い連携させた。また、連携させる技術として、今後のシステム変更に柔軟に対応できるように XML ファイルによるデータ交換と Web サービスを利用してシステム間の連携を行った。

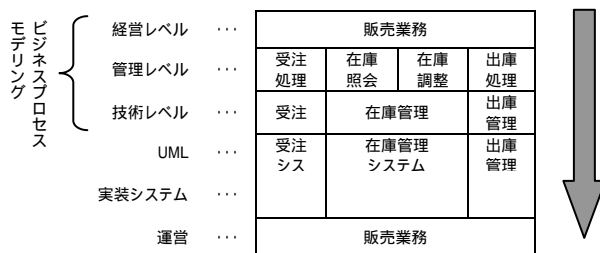


Fig.1 開発の流れ

4. 評価方法

今回提案した、システムに連携させるための開発方式の検証するための方法として、提案手順にのっとったシステムを構築して、以下の項目を評価した。

- (1) 全体業務の把握のしやすさ
- (2) 要求定義の明確さ
- (3) 連携に関する設定の容易さ

5. まとめ

本稿では、ビジネスプロセス・モデリングを行うことで、今日のシステム開発の問題とされている『要求定義』『システム連携』『業務改善』の解決方法を考えた。その解決には、業務全体を把握することを第一に考え、それからシステムへ実装する必要があると考えたが、それを行うことは非常に困難である。そのため、この流れを容易に実行させるために全体業務のプロセスモデルを記述して全体業務の把握と改善を管理する経営レベルから、そのモデルを詳細なプロセスモデルへと分割して記述を行い各担当者が行う業務の把握と改善を管理するレベル、管理レベルのモデルを UML へと変換可能なプロセスモデルを記述する技術レベルの 3 つのレベルに分けてプロセスモデルを記述し開発する方法を提案した。

今後の展開として、本提案による開発方式の手順書の作成を行う。また、本提案は BPM ツールの ARIS を用いたので、その他の BPM ツールでも利用可能にする。

参考文献

- [1] 矢口竜太郎, 栗原雅, “俊敏な情報システム,” 日経コンピュータ, 2005.4.4, no.623, pp.50-63, April.2005.
- [2] 小泉寿男, 辻秀一, 吉田幸二, 中島毅, ソフトウェア開発, オーム社, 2003
- [3] 戸田保一, 飯島淳一, ビジネスプロセス・モデリング, 日科技連, 2000