

## 調査研究論文 SURVEY AND FIELD STUDY

ソフトウェア「要求機能実現管理表」による  
機能欠落低減効果の統計的検証<sup>†</sup>

大森 晃\*/辻田祐純\*\*

Statistical Verification on Effects of Reducing Missing Functions with “Functional Requirements Implementation Management Table” for Software

\*Akira OHMORI/\*\*Hirozumi TSUJITA

A software development project usually has the risk of unimplementing several requirements among the specified software requirements. The existence of the unimplemented software requirements is a problem that cannot be ignored, because it is directly connected to delivery date delay and a cost increase, and then leads to a deterioration of profitability.

As a software requirement, there is a functional one or a non-functional one. With focus on functional requirements, this paper proposes “Functional Requirements Implementation Management Table” (FRIM Table for short) as a management tool for reducing the number of unimplemented software functional requirements within a framework of a software project management. It also describes an experimental application of FRIM Table in a company, and clarifies concrete usage of FRIM Table in the company. Based on the data obtained from the experimental application, this paper statistically analyzes effects of the application of FRIM Table; and it shows that the application of FRIM Table leads to a tendency toward reducing the number of unimplemented software functional requirements.

*Key words* : empirical study, management tool, package, project management, software development

## 1. はじめに

ソフトウェア開発プロジェクト（以下、開発プロジェクトと略称）の目的は、端的にいえばコスト・納期の制約を満たしつつ、ソフトウェア要求を定義し、それを確実に実現したソフトウェアを開発し、市場にリリースあるいは客先に納品することである。こうした目的の達成に導くために、手法の差異や能力の差異はあるものの、通常はソフトウェア開発プロジェクト管理（以下、プロジェクト管理と略称）が行われている

[1],[2],[3],[4]

プロジェクト管理は、開発プロジェクトの目的を達成するための必要条件ではあるが十分条件ではない。したがって、高度な管理能力を備えたプロジェクト管理が行われたとしても、開発プロジェクトはソフトウェア要求の実現漏れというリスクをかかえるのが普通である。ソフトウェア要求の実現漏れは納期遅延、コスト増に直結し、収益悪化をもたらすため無視できない問題であることはいうまでもない。

ソフトウェア要求には機能的要求と非機能的要求がある。機能的要求はソフトウェアが実行すべき機能に関する要求であり、非機能的要求はソフトウェアの外部とのインタフェース、性能、品質などに関する要求である。本論文では、機能的要求（以下、要求機能と呼ぶ）に焦点をあて、プロジェクト管理の枠内で要求機能の実現漏れ、つまり開発したソフトウェアにおける機能欠落を低減させるための管理ツールとして要求機能実現管理表を提案する。また、適用実験と統計的

<sup>†</sup>平成17年3月7日 受付

平成18年3月29日 改訂

平成18年6月16日 採択（論文誌編集委員会）

\*東京理科大学 工学部 経営工学科

\*\*高知工科大学 情報システム工学教室

連絡先：\*〒162-8601 東京都新宿区神楽坂1-3（大森晃）

分析を通して、その有効性を明らかにする。

第2章では、要求機能実現管理表について、その概要、活用上の前提、開発フェーズにおける活用時期について述べる。第3章では、ある企業における要求機能実現管理表の適用実験について述べるとともに、その具体的な適用法を明らかにする。また、通常の方法との差異を明らかにする。第4章では要求機能実現管理表の有効性を分析するために、どのようなデータを収集したかについて述べる。また収集データを基に要求機能実現管理表の適用によって得られる機能欠落低減に関する効果を統計的に分析する。第5章では、第4章での統計的分析結果に対する要求機能実現管理表以外の影響条件について吟味し、統計的分析結果の妥当な解釈を示す。第6章では、要求機能実現管理表の活用が実用性を有するかどうかを検討する。第7章では、関連研究について議論し、本論文の新規性を明確にする。

## 2. 要求機能実現管理表

本章では、開発したソフトウェアにおける機能欠落を低減させるための管理ツールとして、表・1に示すような要求機能実現管理表を提案する。この表における上部左側は、従来からも利用されているマトリックス形式の相互参照表<sup>[5],[6],[7]</sup>の一種であり、どの要求機能がどのプログラム<sup>1)</sup>によって実装されるかをマトリックス形式で記録し、相互参照を確保できるようにしている。例えば、図中では「要求機能1と要求機能3がプログラム1.1によって実装される」ことを「レ」(チェックマーク)によって示してある。表の上部右側と下部は、要求機能の実現に関する管理機能を強化するために、上述した相互参照表に追加した部分であり、要求機能実現管理表の特徴的部分になっている。表の上部右側では要求機能の実現をプログラムレベルで管理できるようにするために、プログラムのコーディング、単体テストの進捗を確認するトリガーを提供し、確認結果を記録できるようにしてある。表の下部では要求機能の実現をシステムレベルで管理できるようにするために、開発プロジェクトにおける最終テスト終了時点で、要求機能が実現されているかどうかを確認するトリガーを提供し、確認結果を記録できるようにしてある。

1. いくつかの機能を実現するソフトウェア構成要素としてのプログラムである。

要求機能の実現は最終的にそれを実装するプログラムのコーディングによって行われる。また、開発プロセスの途中で要求機能の実現を確認できるのは、プログラムレベルでの単体テストによってである。そして、要求機能の実現を最終的に確認できるのは、システムレベルで行われる最終テストによってである。そのためコーディングとこれら2種類のテストに着目して要求機能の実現を管理することは合理的である。さらに、要求機能とプログラムとの相互参照表は、要求機能を各プログラムに関係づけるとともに、要求機能の実装漏れを防止することに役立つ。

なお、プロジェクト管理から離れて要求機能実現管理表を単独で活用することは想定していない。あくまでも、それはプロジェクト管理の枠内で要求機能の実現に関する管理機能を強化するためのものであり、通常のプロジェクト管理に連動させて活用することを前提としている。また、要求機能実現管理表の活用にあたって、納期、品質、コストなどを含めて総合的なプロジェクト管理体制を組めることが理想的な前提であるが、プロジェクト管理の枠内で必要不可欠の前提は、要求機能の変更が常に正しく要求機能実現管理表に反映されるということである。詳しくいえば開発プロジェクトが今現在ターゲットにしている公式の要求機能が明確になっており、それら要求機能が常に正しく要求機能実現管理表に反映されるということである。

ソフトウェア開発のフェーズとして想定しているものは、要求定義、設計、コーディング、テストといったほぼ標準的なものである。要求機能実現管理表の適用時期は、設計作業終了時点からが妥当と考える。その理由は諸種の変更を経ながらも、その時点で要求機能がほぼ固まるであろうと考えられるからである。

当然ながら具体的な適用法は、個別の企業ないしプロジェクトによって変わらざるを得ない。各企業が整備しているプロジェクト管理手法の違いがその大きな理由である。具体的な適用法については、次章で言及することにする。

## 3. 要求機能実現管理表の適用実験

### 3.1 適用会社

要求機能実現管理表の適用実験について、国内の外資系大手システムインテグレータ(以下、A社と呼ぶ)における金融系部門からの協力を得た。A社では、ソフトウェア開発プロジェクトで利用される開発

表・1 要求機能実現管理表

要求機能実現管理表		要求機能								担当者名	コーディング		単体テスト	
		要求機能1	要求機能2	要求機能3	要求機能4	要求機能5	要求機能6	要求機能7	要求機能8		着手日	完了日	着手日	完了日
プロジェクト：C-13														
プロジェクトマネジャー：〇〇														
サブシステム	プログラム													
サブシステム1	プログラム1.1	レ	レ							△△	7/15	7/21	7/21	7/25
	・													
	・													
	・													
	・													
	・													
	・													
	・													
	プログラム1.n													
実現済み機能（レを入れる）		レ									プロジェクト最終テスト結果			
											確認日	MM/DD		
											確認者	〇〇		

作業標準<sup>2</sup>およびプロジェクト管理手法が整備されている。

A社における通常のプロジェクト管理は、かなり洗練されたプロジェクト管理手法に従って行われている。しかしながら、金融系部門においては表面上プロジェクト管理がうまくいっているように見えるが、完成したソフトウェアがすべての要求機能を満たしていないという状況、つまり要求機能の欠落がしばしば発生していた。

こうした状況は開発コストの増大、開発スケジュールの遅延による収益の圧迫をもたらし、収益確保に致命的なダメージを与える危惧があった。そのため要求機能の実現に関する管理機能を強化し、要求機能の欠落を低減することが、A社金融系部門にとって重大な課題となっていた。

### 3.2 対象ソフトウェア

要求機能実現管理表の適用実験において対象としたソフトウェアは、A社金融系部門外部からの受注品ではない。それはA社金融系部門の商品企画担当者がA社金融系ソフトウェア開発部門に対して開発を要求するもので、ミドルウェアを主体としたパッケージ。この用語は、「ソフトウェア開発に関する作業手順・作業方法の規定」という意味で用いている。

ジである。

### 3.3 適用実験における実験群・統制群の設定

要求機能実現管理表の有効性を評価する際、実験用プロジェクトとして同時期に発足するプロジェクト群から、要求機能実現管理表を適用するプロジェクト群（実験群）と要求機能実現管理表を適用しないプロジェクト群（統制群）をそれぞれ適度な数だけ選択し、実験用プロジェクト完了後にプロジェクトデータを収集し、実験群・統制群の差異を適当な統計的手法によって分析するのが理想である。ここで特に問題になることは同時期における実験用プロジェクト群（実験群と統制群）の設定である。

これらを適度な数だけ新たに発足させて実験を行うことは、A社金融系商品企画担当者の日常業務との兼ね合いもあって現実的ではなかった。また、実験群・統制群という実験用プロジェクト群を同時期に発足させた場合には、実験群・統制群に属する実験用プロジェクトの数が少なくなり、両者の差異を分析するための十分なデータ数を確保できない可能性が高かった。そこで実験群、統制群を以下のように設定することとした。

実験群：新たに発足するプロジェクト群に対しては、A社の開発作業標準およびプロジェクト管理手

法を従来どおり用いるとともに、要求機能実現管理表を用いてソフトウェアを開発することとし、これらプロジェクト群を実験群にすることとした。

統制群：過去、A社の開発作業標準およびプロジェクト管理手法を用いてソフトウェアを開発してきたプロジェクト群から、実験群と比較して課せられた要求機能の数が同程度のものを選別して、それらを統制群にすることとした。意味的にわかりやすくするために、A社における統制群を以下、通常群と呼ぶことにする。通常群の詳細な選別手続きについては4.1節で言及する。

### 3.4 要求機能実現管理表の具体的適用法

要求機能実現管理表は、プロジェクトごとにすべての設計作業が終了した時点から、プロジェクト・マネージャーによって、表・2に示すように以下の手順で作成された。

(1) 要求機能およびプログラムの明記：

(1-1) 設計作業終了時点で確定している要求機能を明記する。記載スペースの制約があるので、要求機能を明記する際にその名称ではなく、その識別記号を用いた。

A社で用いられているプロジェクト管理手法では、要求機能名称と要求機能識別記号を一覧にした表「要求機能一覧表」を作成するように定められている。その概要を表・3に示す。要求機能一覧表は、要求定義終了時に作成され、その後、要求変更に関する管理プロセスの中で要求機能の変更・追加・削除を経て更新される。A社では、設計作業終了時点で確定している要求機能一覧表に記載された識別記号である要求区分(大文字アルファベットA, B, C…), 要求機能(数字1, 2, 3…), 要求機能詳細(小文字アルファベットa, b, c…)を要求機能実現管理表に要求機能として明記した。

(1-2) 設計書を参照して、実装すべきプログラム名をサブシステム単位で抽出し、サブシステム名/プログラム名を明記する。

(2) 要求機能とプログラムとの相互参照の明記：

設計書をもとに、以下を行う。

(2-1) どの要求機能が、どのプログラムによって実装されることになっているのかを、チェックマーク(レ)によって明記する。

(2-2) どのプログラムにも対応づけられていない要求機能(つまり、見落とされた機能)がないかをチェックし、見落とされた機能が発見された場合には設

表・2 要求機能実現管理表の適用例

要求機能実現管理表	要求機能												システムテスト結果				
	A			C			D			E			担当者名	コーディング 着手日	コーディング 完了日	単体テスト 着手日	単体テスト 完了日
	1	2	3	4	5	6	7	8	9	10	11	12					
プロジェクト名称：C-01																	
プロジェクトマネージャー：KB																	
サブシステム																	
1. 基本 (業務共通)	レ	レ	レ	レ	レ	レ	レ	レ	レ	レ	レ	レ	レ	レ	レ	レ	レ
2. 個別開始終了																	
3. 付帯など																	
4. プロトコル																	
3. 犯罪カード使用通知																	
1. TCP/IP																	
2. 全銀ハイレベル																	
3. UDP/IP																	
担当者名	NM	NM															
コーディング 着手日	1/13	1/27															
コーディング 完了日	1/19	1/29															
単体テスト 着手日	1/20	1/29															
単体テスト 完了日	1/26	2/3															
担当者名	MG	IT	IT	IT													
コーディング 着手日	2/27	1/13	1/27	2/4													
コーディング 完了日	3/2	1/19	1/29	2/6													
単体テスト 着手日	3/3	1/20	1/29	2/9													
単体テスト 完了日	3/9	1/26	2/3	2/13													
システムテスト結果																	
確認日																	7/1
確認者																	KB

表・3 要求機能一覧表

要求区分	要求機能	要求機能詳細
A. 業務共通	1. 24時間サービス	a. ログ切替
		b. カウンター切替
		c. 精査ファイル更新
	2. 時間延長開始終了 (個別開始終了)	a. ログ出力
		b. カウンター切替
		c. 終了処理受付
		d. 開始処理受付
		e. 電文出力
		f. タイムアウト監視
		g. タイムアウト時延長
		h. 返送電文入力
		i. 終了処理通知
		j. 開始処理通知
∴	∴	
C. 付帯サービス	∴	∴
	∴	∴
	∴	∴
	4. 犯罪カード機能 犯罪カード使用通知 業務	a. ログ出力
		b. カウンター更新
c. 通知内容入力		
d. 通知電文編集		
e. 通知電文出力		
D. プロトコル対応	1. TCP/IP 対応	a. 手順確認
		b. PID 接続
	2. 全銀ハイレベル対応	a. 手順確認
		b. PID 接続
	3. 直結 (UDP/IP) 対応	a. 手順確認
		b. PID 接続

計変更などの適当な処置をとる。

(3) プログラム担当者の明記：

個々のプログラムについて、コーディングと単体テストを担当するスタッフの氏名を明記する。

(4) コーディング着手日/完了日の明記：

プログラム担当者からの報告を受け、プログラムのコーディング着手日/完了日を明記する。完了日については、要求機能を盛り込んだコーディングを行ったことを担当者に確認した上で明記する。

(5) 単体テスト着手日の明記：

プログラム担当者からの報告を受け、プログラムの単体テスト着手日を明記する。

(6) 単体テスト完了日の明記：

プロジェクト・マネジャーがプログラム担当者から、プログラムの単体テスト結果に関する報告書を受け取った後、要求機能の実現を確認し、単体テスト完了を承認した日付を単体テスト完了日として明記する。単体テスト完了日が明記されているということは、プログラムレベルで当該要求機能の実現が確認さ

れていることを意味する。

(7) システムテスト結果確認日・確認者の明記、および実現済み要求機能の明記：

システムテスト結果<sup>3</sup>に関する報告書を受け取った後、要求機能の実現されているかどうかを確認し、実現されている要求機能については対応するマス目にチェックマーク (レ) を明記する。また、システムテスト結果の確認日・確認者 (プロジェクト・マネジャー) を明記する。

上記のようにして要求機能実現管理表を作成し、またそれを管理するにあたって、A社では新たに以下のルールを設定した。

ルール1：要求機能実現管理表の作成・管理は、プロジェクト・マネジャーが責任をもって行う。

ルール2：日付を明記するため、コーディング着手日/完了日、単体テスト着手日については、プログラム担当者からプロジェクト・マネジャーへの報告を義務づける。単体テスト完了日については、単体テスト結果についての報告書を基に判断する。

ルール3：要求機能実現管理表の作成開始以降、プロジェクト・マネジャーは、A社において定期的開催される上位管理者レビュー会議 (プロジェクト・マネジャーの上位管理者、および商品企画担当者がレビューアとして出席するレビュー会議)<sup>4</sup>において、要求機能実現管理表の状況について報告し、管理表の記載内容について、開発ソフトウェアによる収益確保に責任を負う商品企画担当者から承認を得なければならない。

ルール4：要求機能の変更は、確実に要求機能実現管理表へ反映させる。

上記ルール1は責任の所在を明確にし、ルール2はプロジェクト・マネジャーの負担をいくらか軽減するとともに、コーディング以降の進捗を的確に把握するためのものである。ルール3は、要求機能実現管理表の活用を、プロジェクト内に閉じた非公式の活動にとどめるのではなく、上位管理者レビュー会議の中での公式の活動として位置づけるためのものである。ルール4は最も重要であり、要求機能実現管理表の活用が徒労に終わらないようにするためのものである。A

3. A社におけるシステムテストは、開発したソフトウェアのリリース前、擬似運用環境でプロジェクトによって行われる最終テストである。

4. 上位管理者レビュー会議の定期的開催は、A社プロジェクト管理手法において規定されていることであり、今回新たに設定したルールではない。

社にはかなり洗練されたプロジェクト管理手法が整備されており、要求機能の変更に関する管理プロセスによって、ルール4の達成が可能であった。

当然のことであるが、要求機能実現管理表において、実現すべき要求機能の記載漏れが発生しないようにすることが重要である。そのために、まず上記ルール3によって、要求機能実現管理表の記載内容について上位管理者レビュー会議で承認を得なければならないことから、要求機能の記載漏れを防止するプロセスを提供した。また、ルール4によって要求機能の変更への対応漏れがないように注意を促した。

### 3.5 通常の要求管理法

通常群で行われた標準的な要求管理の概要を述べる。なお、開発プロジェクトにおける要求管理は多様な側面を持っているので、以下では特にレビュー、要求変更に関する管理に焦点をあてる。

一つには、開発作業標準に従って開発フェーズの節目（要求定義終了、設計終了など）で開発部門内に閉じた品質レビューが行われる。要求定義については一貫性、完全性、正確性などの品質についてレビューが行われ、要求定義以降についてはエラーに焦点をあてたレビューが行われる。

一方、前節でも述べたように、プロジェクト管理手法に従ってプロジェクトの発足から完了まで定期的に上位管理者レビュー会議が開催される。この会議では財務・進捗、要求などの面からプロジェクトの状況がレビューされる。要求に焦点を当てていけば、これはプロジェクト・マネジャーが、その上位管理者に対して要求の実現状況を報告し、承認を得るための会議である。ただし、要求に変更がある場合には商品企画担当者の承認も必要となる。

要求変更に関する管理は、プロジェクト管理の一環として次のように行われる。まず、要求の変更は表・3に示す要求機能一覧表に対する変更としてとらえられる。そして要求の変更は変更管理ボードによって評価される。変更管理ボードはプロジェクト・マネジャーを議長とし、商品企画担当者、協力会社リーダーから構成される委員会で変更の評価、承認、承認した変更への対応スケジュール作成を主要な役割とする。変更が承認されると、それは要求機能一覧表に反映される。

### 3.6 通常群と実験群における要求管理法の差異

通常群と実験群における要求管理法の差異は、要求機能実現管理表を活用するか活用しないかに尽きる。

しかしながら、詳細に見れば要求機能実現管理表を活用する上で行った管理手続きに関する変更点も差異といえる。変更点については3.4節においてルール1～ルール4としてすでに言及した。また、要求機能実現管理表を活用することによって生じる以下のような管理上のメリットも差異となる。

(1) 要求機能とプログラムとの相互参照表は、要求機能実現管理表の活用によって初めて公式に作成されるようになる。これは、設計終了時点で要求機能の設計漏れが発見しやすくなるというメリットを与える。

(2) 要求機能実現管理表の活用によって初めて公式に、システムレベルで個々の要求機能の実現をチェックするしかけを得ることになる。これは要求機能の実現漏れの全貌が把握しやすくなるというメリットを与える。

## 4. 収集データと統計的分析

### 4.1 収集データ

A社金融系部門の開発プロジェクトにおける開発・管理に伴って作成された各種ドキュメント、各種報告書をもとに実験群、通常群から以下に示すデータを収集した。結果をそれぞれ表・4と表・5に示す。

(a) 要求機能数：実験群については、要求機能実現管理表の活用効果を分析したいので、システムテスト結果確認日において要求機能実現管理表に記載されている要求機能の数である。一方、通常群については、実現されるべき機能として要求機能一覧表に記載されている要求機能の数である。

(b) 実現機能数：実験群については、要求機能数と同様の理由から、システムテスト結果確認日において要求機能実現管理表で実現が確認された要求機能の数である。一方、通常群についてはリリース判定会議<sup>5</sup>の開催時点で開発されたソフトウェアが実現している要求機能の数である。

(c) 機能欠落率： $(\text{要求機能数} - \text{実現機能数}) \div \text{要求機能数} \times 100$ 。

表・4に示す実験群のデータは、要求機能実現管理表の適用を開始した2003年8月以降、2004年12月までの期間内に発足・完了した12プロジェクトすべてから収集したものである。なお、表・4には、要求機能実現管理表の活用がどの程度の負担増をもたらす

5. A社のプロジェクト管理手法では、システムテスト終了時点で、ソフトウェアのリリース可否を判定するための会議を開催するよう定められている。

表・4 実験群からの収集データ

プロジェクト	要求機能数	実現機能数	機能欠落率(%)	開発負担増(人月)	開発工数(人月)	開発負担増/開発工数(%)
C-01	243	243	0.00	0.50	11.10	4.50
C-02	675	653	3.26	0.50	15.60	3.21
C-03	557	557	0.00	0.25	20.10	1.24
C-04	639	639	0.00	0.75	33.00	2.27
C-05	2,385	2,360	1.05	0.50	178.50	0.28
C-06	135	135	0.00	0.30	10.00	3.00
C-07	1,582	1,566	1.01	2.30	25.75	8.93
C-08	135	120	11.11	0.35	15.85	2.21
C-09	438	435	0.68	1.50	17.80	8.43
C-10	965	965	0.00	0.80	17.50	4.57
C-11	746	723	3.08	0.65	9.50	6.84
C-12	1,326	1,228	7.39	1.50	72.18	2.08

表・5 通常群(統制群)からの収集データ

プロジェクト	要求機能数	実現機能数	機能欠落率(%)
B-01	1,523	1,238	18.71
B-02	588	539	8.33
B-03	255	236	7.45
B-04	338	330	2.37
B-05	689	630	8.56
B-06	1,050	992	5.52
B-07	385	298	22.60
B-08	2,465	2,350	4.67
B-09	710	693	2.39
B-10	1,922	1,723	10.35
B-11	2,182	2,026	7.15
B-12	2,385	2,356	1.22

かを吟味するため、開発負担増として要求機能実現管理表の作成・管理・報告などにかかった工数、および開発工数、開発負担増が開発工数に占める比率も示した。

一方、表・5に示す通常群のデータは、以下のような手続きを経て選別されたプロジェクトのデータである。

(1) 要求機能実現管理表を適用していない過去のプロジェクトのうち、データ収集が可能であった16プロジェクトを通常群の候補とし、それらのデータを収集した。ここで対象とした過去のプロジェクトは、1999年4月以降2003年12月までの期間内に完了していたプロジェクトである。

(2) 通常群の候補である各プロジェクトに課された

要求機能数を確認し、実験群のプロジェクトに課された要求機能数135件~2385件の範囲を大きく外れる4プロジェクトを通常群の候補から除外した。そして、要求機能数がほぼ同様の範囲(255件~2465件)に収まる12プロジェクトを、とりあえず通常群として選別した。

(3) 要求機能数に関して、実験群と通常群との間に統計的有意差があるかどうかをウィルコクソンの順位和検定を用いて分析した。その結果、有意確率(両側)は0.236であり、両群に統計的有意差は認められなかったため、上記(2)でとりあえず通常群として選別した12プロジェクトを、そのまま最終的な通常群とした。

## 4.2 統計的分析

機能欠落率という観点から有効性を統計的に分析するために、以下の帰無仮説・対立仮説を設定した。

帰無仮説：機能欠落率について通常群と実験群は、ともに同一母集団分布に従う。つまり、機能欠落率には通常群と実験群による差はない。

対立仮説：実験群のほうが通常群より、機能欠落率が低い。

2つの群のデータを基にウィルコクソンの順位和検定を用いて有意確率(片側)を求めた。その結果、有意確率(片側)は0.00194であった。したがって、1%有意水準で帰無仮説が棄却され、対立仮説が採択される。

## 5. その他の影響条件の吟味と統計的分析結果の妥当な解釈

通常、前節の統計的分析結果から機能欠落率の低減は要求機能実現管理表の活用によるものであると単純に判断することはできない。なぜならば開発プロジェクトには多種多様な条件が絡んでおり、機能欠落率に影響を及ぼす条件が、要求機能実現管理表の活用以外に存在する可能性があるからである。そうした条件をすべて吟味することは不可能であるが、主要なものについて少なくとも理論的に吟味しておくことは重要である。

### 5.1 要求機能数について

要求機能数はプロジェクトが達成しなければならない課題の量を意味し、そのような条件の一つと考えられる。しかしながら、4.1節後半で述べた通常群の選

定手続きによって、実験群と通常群のどちらか一方が要求機能数について極端に多いあるいは少ないという状況を前もって回避している。したがって、統計的分析結果への要求機能数の影響は排除されていると考えてよい。

## 5.2 開発期間とコストの制約について

一般的に機能欠落率に大きく影響を及ぼす可能性のある条件としては、プロジェクト・スタッフへの心理的圧力となり得る開発期間制約およびコスト制約が考えられる。ところが、本論文で対象にしたソフトウェアはA社主導の自主開発的なパッケージであり、A社の規定（開発作業標準）によれば、要求定義終了後にプロジェクトがこれらの制約についての適正な見積もりを確定させることになっている。これは実験群、通常群ともに同様である。各プロジェクトにおける制約の内容に差異はあるが、プロジェクト自らが設定した適正な制約であることから、そうした制約がプロジェクト・スタッフへの心理的圧力となって、実現機能数（つまりは機能欠落率）へ影響を及ぼしたとは考えにくい。

## 5.3 プロジェクト管理手法について

通常群と実験群のプロジェクトが発足・完了した時期に、ずれがあることには注意を要する。両群が利用したプロジェクト管理手法に差異があれば、その差異が機能欠落率に影響を及ぼしたかもしれない。A社では、米国本社で1990年代初期に開発されたプロジェクト管理手法を、90年代半ばから導入し90年代後半（99年4月以前）までには金融系部門の全プロジェクトで活用するに至った。その後、2002年6月経営トップの交代を機に、各種資格取得に熱心となり、金融系部門は経営戦略上CMMIレベル3を2003年2月に、CMMIレベル5を2004年1月に取得している。ただし、資格取得にあたって、A社はプロジェクト管理手法の公式的変更を行っていない。これはA社におけるプロジェクト管理手法の内容が、もともとCMMIレベル5に相当していたためであると考えられる。また、90年代後半にプロジェクト管理手法を本格的に活用しはじめて以降、高い水準のプロジェクト管理を実質的に行っていたためであると考えられる。さらに1999年4月～2004年12月の期間内に、プロジェクト管理手法の公式的変更は一切なく、CMMIレベル3、レベル5の資格取得によってプロジェクト管理の水準が大きく変化したというわけでもない。

ない。

以上のことからA社金融系部門は1999年4月～2004年12月の期間内にCMMIレベル3、レベル5を取得してはいるものの、その期間内でプロジェクト管理手法の公式的変更はなく、またプロジェクト管理の水準に大きな変化もなく、プロジェクト管理手法について両群における差異<sup>6)</sup>はなかったと判断できる。

## 5.4 プロジェクトの開発技術水準について

本適用実験において、プロジェクトの開発技術水準は機能欠落率とは無関係といってよい。なぜならばA社商品企画担当者は実現可能なパッケージを企画し、開発部門にその開発を要求したのであり、開発技術水準の高低が実現機能数（つまりは機能欠落率）を左右する余地はないからである。なお、ここでいう開発技術は設計技術などソフトウェアを実現するための技術である。

## 5.5 個人属性について

プロジェクト・スタッフ（マネジャーおよび一般スタッフ）の個人属性は、機能欠落率に影響を及ぼす非常に重要な条件と考えられる。要求機能を認識する能力、その認識を維持する能力（認識を維持するために何らかの工夫をする能力も含む）、認識した要求機能を実現しようとする能動的姿勢が特に重要であろう。なぜならば、要求機能の実現がスタッフの責務であることから、要求機能をスタッフがきちんと認識し、その認識をきちんと維持し、要求機能を実現しようとする能動的姿勢さえ持てば、論理的には要求機能の実現漏れは発生しないからである<sup>7)</sup>。

上述したようなスタッフの個人属性は、実験者である著者らが統制できないものである。また、測定も困難なものである。したがって、ここで吟味すべきことはスタッフ配置について実験群と通常群で極端な差異があったかどうかである。

そこで、A社金融系のソフトウェア開発部門において、プロジェクトへのスタッフ配置がどのように行われているかを調査した。その結果、プロジェクトのマネジャーと一般スタッフには、適任と思われる者が任意に選ばれていた。いわゆる任意性のある適材適所の原則のもと、プロジェクトへのスタッフ配置が行わ

6. 実験群に加えられた実験操作である要求機能実現管理表の活用、およびそれに伴う手続きの変更は除く。

7. 要求機能実現管理表は、まさにこの論理を利用した管理ツールである。



れていることがわかった。このことからスタッフ配置について実験群と通常群において極端な差異があったとは考えられない。具体的にいえば、通常群に比べて有能なスタッフが、実験群へ意図的に配置されたとは考えられない。結果として、プロジェクト・スタッフの個人属性は、機能欠落率に影響を及ぼしたかもしれないが、実験群と通常群に等しく偏りなく生じたと考えられ、要求機能実現管理表による機能欠落率の低減効果を主張することの妨げにはならないと考えられる。

### 5.6 統計的分析結果の妥当な解釈

以上の吟味から4.2節の統計的分析結果に基づき、要求機能実現管理表を活用することによって要求機能の実現漏れが低減される傾向にあると解釈することは妥当と考える。このように要求機能実現管理表の活用は、要求機能の実現漏れ低減という面で効果的であったといえる。

## 6. 要求機能実現管理表の実用性

前節によって、要求機能実現管理表の有効性は明らかになった。しかしながら、その活用が多量の負荷をプロジェクトに与えるとすれば、また活用効果が小さければ実用性に関する問題が残る。

そこでまず、要求機能実現管理表の作成・管理・報告など要求機能実現管理表を活用することによって新たに発生する工数（表・4における開発負担増）について分析する。表・4から開発負担増が開発工数に占める比率は0.28%~8.93%の範囲内にあり、平均値は3.96%、中央値は3.10%である。平均的に見れば要求機能実現管理表の活用がプロジェクトに多量の負荷を与えるとはいえない。

一方、活用効果について分析する。表・5から通常群における機能欠落率は1.22%~22.60%の範囲内にあり、平均値は8.28%、中央値は7.30%である。実験群においては、表・4から機能欠落率は0.00%~11.11%の範囲内にあり、平均値は2.30%、中央値は0.85%である。これは要求機能実現管理表の活用が通常では1.22%~22.60%の範囲内にあった機能欠落率を、0.00%~11.11%の範囲内に収めたことを示している。また、平均的に見れば平均値は5.98%の減少を示しており、中央値は6.45%の減少を示している。通常群と実験群を合わせて求めた要求機能数の平均値は約1010件であり、中央値は約700件で

あることから平均的に見ても活用効果は決して小さいとはいえない。

以上の分析から、要求機能実現管理表の活用はプロジェクトに多大な負荷を与えるものでもなく、その活用効果は小さいともいえず、要求機能実現管理表は実用的な管理ツールであると判断できる。

## 7. 関連研究

本論文の特徴は、以下の3点である。

(1) プロジェクト管理において要求機能の実現に関する管理機能を強化するための管理ツール（一般的に言えば、プロジェクト管理の改善方法）を提案し、ある企業内で複数の開発プロジェクトに適用していること。

(2) その改善方法の有効性について実証的研究を行い、適正な手続きを経て有効性を明らかにしていること。

(3) その改善方法の実用性をも明らかにしていること。  
残念ながら第3項目に関連する研究報告を見いだすことはできなかったため、以下、第1項目と第2項目の2点に関連する研究報告について議論する。

まず、プロジェクト管理の改善方法、その実施に関してはさまざまな報告を見いだすことができる。例えば、坂井<sup>[1]</sup>はプロジェクトの成功率向上を目的としたプロジェクト管理支援システムについて、福田<sup>[2]</sup>はプロジェクト管理における負荷の軽減、管理情報のタイムリーな入手を目的としたプロジェクト管理支援システムについて、風間<sup>[3]</sup>は検査・検証機能を強化するための手法について、東ら<sup>[4]</sup>はデータ収集を強化する手法について報告している。しかしながら、要求機能の実現に関する管理機能を強化する方法についての報告は見当たらない。

一方、プロジェクト管理の改善方法の有効性を実証的に研究した報告は多くはない。その上、きちんとした実証的研究が行われているものは見当たらない。大槻<sup>[8]</sup>は、稼働中のシステムに対する追加要求機能を実現し数カ月単位でリリースするプロジェクトにおいて実施したプロセス改善策の効果を故障件数、生産性、手戻りの3点から定量的に評価し、改善策の有効性を示している。ただし、これは一つのプロジェクトに限定したもので、実験法に基づく検証<sup>[9]</sup>にまでは至っていない。

田島・川合ら<sup>[10]</sup>は、リスク管理機能を強化する手法を提案し、提案手法について有効性の検証を行ったとしている。実験法に基づく検証という立場をとってはいるが、いわゆる従属変数（多くの場合、有効性を

評価する際の指標)が示されていないために、どうい  
う点での有効性を検証したのかが不明である。さらに  
統計的分析が行われておらず、有効性を主張する根拠  
に客観性が欠けている。

水野・二木ら<sup>[11]</sup>は、ある企業におけるレビュープ  
ロセス改善策の有効性を示したとしている。具体的  
には統計的仮説検定手法を用いて、改善策実施前のプロ  
ジェクトと改善策実施後のプロジェクトとの間に、レ  
ビュー工数、品質(検出フォールト数)について、5  
%有意水準で有意差があったとしている。しかし、仮  
説検定の対象となる統制群(改善策未実施のプロジェ  
クト群)、実験群(改善策実施のプロジェクト群)の  
設定に問題を残している。この報告では、統制群に改  
善策実施のプロジェクトと改善策未実施のプロジェクト  
が混在している。この点で検定手法の適用に適切で  
ない面があり、検定結果に疑問を残している。

以上に述べたことから、本論文の特徴を備えた研究  
報告は、筆者らが知る限りにおいては見当たらないと  
いうのが実情である。

## 8. おわりに

本論文では、要求機能の実現漏れを低減するための  
管理ツールとして、要求機能実現管理表を提案した。  
要求機能実現管理表は、要求機能とプログラムとの相  
互参照表に、要求機能の実現に関する管理機能を強化  
するための表を新たに連結したものである。この種の  
管理ツールは、これまで提示されていない。

また、要求機能実現管理表の適用事例の紹介だけ  
にとどまらず、その有効性を評価するために適用実験を  
ソフトウェア開発現場で行い、適用実験から得られた  
データを機能欠落の低減という点から統計的に分析  
し、その統計的分析結果の妥当な解釈を求めた。その  
結果、要求機能実現管理表の活用によって機能欠落が  
低減する傾向にあることを明らかにした。また、要求  
機能実現管理表の活用負荷と活用効果を分析し、その  
実用性も明らかにした。

これは本論文の大きな成果である。しかしながら、  
本論文で提示した結果がA社金融系部門以外のプロ  
ジェクト群では成立しない可能性も否めない。そのた  
め本論文の成果をより一般化していくために、A社  
金融系部門とは異なる状況での適用実験が望まれると  
ころである。これについては、今後の課題となる。

世界的に見て、技術・管理の両面において近年のソ  
フトウェア工学では、手法の提案だけでなく、手法の

有効性に関する実証的研究が求められている<sup>[12]</sup>。そ  
うした流れにおいて、本論文は国内における実証的研  
究成果を蓄積していくことに貢献するものである。

## 参考文献

- [1] 坂井田幸一(2000):“ソフトウェア開発におけるプ  
ロジェクト管理システム,”「プロジェクトマネジメン  
ト学会2000年度春季研究発表大会予稿集」, 61-64.
- [2] 福田祥久(2000):“グループウェアによる実践的な  
プロジェクト管理技法,”「プロジェクトマネジメン  
ト学会2000年度春季研究発表大会予稿集」, 125-130.
- [3] 風間敬一(2003):“ソフトウェアプロジェクトマネ  
ジメントにおける実践的品质管理,”「プロジェクトマ  
ネジメント学会2003年度春季研究発表大会予稿集」,  
44-49.
- [4] 東 岳人・久米田暁文・角田文広・小室 睦・菅沼  
弘(2003):“ソフトウェア開発における定量的プロジ  
ェクト管理,”「プロジェクトマネジメント学会2003年  
度秋季研究発表大会予稿集」, 136-140.
- [5] 大森 晃(1990):“ソフトウェア品質管理への品質  
展開アプローチ,”「情報処理学会論文誌」, **31**, (10),  
1474-1485.
- [6] 大森 晃(1992):“ソフトウェア品質展開の枠組み  
設計法,”「情報処理学会論文誌」, **33**, (12), 1598-  
1606.
- [7] A. Ohmori(1994):“Software quality deployment  
approach: Framework design, methodology, and  
example,” *Software Quality Journal*, **3**, (4), 209-240.
- [8] 大槻義則(2001):“プロジェクト管理におけるプロ  
セス改善効果の測定事例,”「プロジェクトマネジメン  
ト学会2001年度秋季研究発表大会予稿集」, 49-51.
- [9] 末永俊郎(編)(1994):『社会心理学研究入門』, 59-  
97, 東京大学出版会.
- [10] 田島理史・川合浩司・他8名(2002):“ソフトウェ  
ア開発プロジェクトにおける実践的初期リスク管理手  
法の提案,”「プロジェクトマネジメント学会誌」, **4**,  
(5), 12-18.
- [11] 水野 修・二木俊樹・新原直樹・高木徳生・菊野  
亨(1998):“ある企業におけるソフトウェアプロセス  
改善の効果に対する統計的分析,”「電子情報通信学会  
技術研究報告, ソフトウェアサイエンス」, **98**, (295),  
1-8.
- [12] 井上克郎・松本健一・鶴保征城・鳥居宏次(2004):  
“実証的ソフトウェア工学環境への取り組み,”「情報処  
理」, **45**, (7), 722-728.