

ソフトウェア開発プロジェクトの遂行に影響を与える要因の分析

古山恒夫 † 菊地奈穂美 ‡ 安田 守 † 鶴保征城 ‡

† 東海大学理学部情報数理工学科
〒259-1292 平塚市北金目 1117

‡ 独立行政法人 情報処理推進機構 ソフトウェア・エンジニアリング・センター
〒113-6591 東京都文京区本駒込 2-28-8

概要：エンタープライズ系ソフトウェアを開発するプロジェクトが、コスト超過・納期遅延・品質低下を起こす要因を分析した。分析の結果、(1)プロジェクト規模そのものが大きく、特に工期当りの規模が大きいプロジェクトはコスト超過を起こす割合が高く、(2)要求仕様があいまいなプロジェクトは納期遅延や品質低下を起こす割合が高いことがわかった。一方、(3)事前に工期の妥当性を評価したプロジェクトでは納期遅延を起こす割合が低く、(4)業務分野の経験者を揃えたプロジェクトでは納期遅延や品質低下を起こす割合が低く、(5)テスト体制を整備したプロジェクトでは、コスト超過・納期遅延・品質低下を起こす割合が低いことを明らかにした。

Analysis of the Factors that Affect Performance of Software Projects

Tsuneo Furuyama †, Nahomi Kikuchi ‡, Mamoru Yasuda †, and Seishiro Tsuruho ‡
† Tokai University, ‡ IPA SEC

This paper reports the analysis results of clarifying factors that cause cost overruns, excessive time to delivery, and lower quality than planned of enterprise software projects. (1) The possibility of cost overruns depends significantly on the project size, in particular, the size of the software to be developed per development duration. (2) Ambiguous requirement specifications accelerate excessive time to delivery and lower quality than planned. (3) Evaluation for project plans improves the possibility to keep delivery time. (4) The projects that provide developers who have experience of target business areas are less possible to excessive time to delivery and lower quality than planned. (5) The projects that have the test teams of high ability can improve all these performance of the projects.

1. はじめに

ソフトウェア開発プロジェクトで見積りを行う場合、まず開発対象ソフトウェアの開発規模を見積り、それに基づいて工数や工期を見積ることが多い。開発規模の見積り方法として、エンタープライズ系ソフトウェアの場合、ファンクションポイント法によって、仕様が固まると比較的安定した精度で見積ることが知られている[1]。規模が見積られると、COCOMO II [2]などの方法で工数や工期を見積ったり、組織で蓄積した生産性データから工数を見積ったりすることができる。蓄積データを見積りに利用するには重回帰分析を用いるのが一般的であるが、欠損値が多い場合は必ずしも十分ではない。欠損値が多い場合にも適用できる方法として、最近では協調フィルタリングの手法をプロジェクト見積りに応用する研究も行われている[3]。これ以外に過去の類似プロジェクトからの類推や、WBS (Work Breakdown Structure) からの積み上げ方式で見積

る方法も現実には有力である。

このように科学的な見積り方法があるにも拘らず、ソフトウェアプロジェクトでは、当初計画の目標通り遂行できないことが少なくない。その主な理由のひとつに、ソフトウェア開発プロセスが、物理的現象を利用してものを製造する化学プラントなどと異なり、人間の知識・能力・組織力に依存しているため、ソフトウェア開発プロジェクトを取り巻く数多くの不確定要因による影響を受けやすいためと考えられる。このことから、ソフトウェアプロジェクトの遂行に影響を与える要因を把握することができれば、ソフトウェア計画立案の指針ともなり、ソフトウェアプロジェクトのリスクを軽減できると考えられる。

IPA (独立行政法人情報処理推進機構) SEC (ソフトウェア・エンジニアリングセンター) では、エンタープライズ系ソフトウェアプロジェクトのデータとして、規模・工数・工期・バグ数などの最も基

本的なプロジェクトデータ項目(量的データ)に加えて、プロジェクト特性を示すさまざまなデータ項目(質的データ)を定義し収集している。2005年度に収集したSECのデータ定義Ver.2.0[4]では、プロジェクト特性のデータとして、プロジェクトの成否を示すデータ、すなわちコスト・納期・品質(QCD)に関して各プロジェクトが計画通りにプロジェクトを遂行することができたかどうかを示すデータ項目を含んでいる。このようなデータ項目はISBSGのデータ[5]にも見られず、筆者らの調査の限りでは新しいと考える。

プロジェクトの成否については、与えられたデータの数値から類推することもできるが、各プロジェクトの判断結果は有力な情報となる。したがって、プロジェクトの成否を示すデータと各プロジェクトの量的データや質的データの関係を分析することによりプロジェクトの遂行に影響を与える要因を抽出できれば、これまでわれわれが経験的に感じていたことを統計的に裏付けることができる。

第2章で分析対象データの概要を、第3章で分析方法を紹介する。第4章で分析結果すなわちプロジェクトの成否と関連の見られる要因の候補を示す。第5章では要因候補とプロジェクトの成否について考察を加える。

2. 分析対象データの概要

分析対象データは、2005年度にSECで収集したエンタープライズ系ソフトウェアプロジェクトのデータ1,419件である[4]。そのうち、分析結果にばらつきの少ない新規開発データ846件のみを分析対象とした。データ定義は[4]のものを使用している。

2.1 目的変数

目的変数は、プロジェクトの遂行実績に関する①実績の評価(コスト)、②実績の評価(工期)、③実績の評価(品質)の3つである。これらの項目に回答があったプロジェクト数は、①が217件、②が215件、③が98件である。それらのデータセット間には、①②③の関係がある。すなわちそれぞれのプロジェクト群はお互いに真部分集合をなしている。各目的変数に対する回答内容とプロジェクト数を表1に示す。ただし、各目的変数とも5つの選択肢を2つに集約している。

目的変数間の関係の有無についてのP値と量的データにおける相関係数に相当するクラメールのV値を表2に示す。お互いに無相関とは言えないが、V値はいずれも0.2から0.35の間にあり、お互いの相関は必ずしも高くない。したがって、それぞれを別々の目的変数として分析することにする。

2.2 説明変数

(1) 量的データ

最初に、ソフトウェアプロジェクトを評価する際に最も重要な要素である①規模を選ぶ。規模はFPベースのものとする。次に、それぞれの目的変数と直接関連のある②工数、③工期、

④開発時のバグ数を選ぶ。開発時のバグ数(以下では「バグ数」と省略)としては、回答数の多い、「検出バグ現象数結合テスト」と「検出バグ現象数総合テスト(ベンダ確認)」の和を用いる。最後に工数、工期、バグ数とも規模と高い相関を持つことから、⑤規模当りの工数(生産性)、⑥工期当りの規模、⑦規模当りのバグ数をとる。

(2) 質的データ

表3に説明変数としてとりあげた質的データを示す。これらは、SECの収集データのうちから、回答数の少ない評価項目を除いたものである。

3. データの分析方法と有意性の判定基準

3.1 説明変数が量的データの場合の分析方法

2.2(1)で述べた7つの説明変数はいずれも右裾が長く伸びる分布をしていて正規分布とはならない。一方、それを対数変換したものは一般に正規分布の仮説が否定されない。このことから、7つの説明変数に対してはすべて対数変換してから分析を進める。次に説明変数ごとに等分散の検定を行い、等分散の仮説が否定されない場合に一元配置の分散分析を行う。なお、欠損データは補正せず、該当するデータ項目の分析に対してのみ除外する。

分析結果は、P値をもとに、(1)危険率1%で有意、(2)危険率5%で有意の他に、(3)危険率10%で有意の場合も考慮する。これはソフトウェアプロジェクトデータのように、多くの要因の影響を受けるデー

表1 目的変数と該当プロジェクト数

目的変数	内容	プロジェクト数
実績の評価 (コスト)	コスト超過なし	180
	コスト超過あり	37
	合計	217
実績の評価 (工期)	納期遅延なし	171
	納期遅延あり	44
	合計	215
実績の評価 (品質)	品質低下なし	69
	品質低下あり	29
	合計	98

表2 実績の評価の相関

	コスト	工期	品質
コスト	1	1.6E-4	0.036
工期	0.257	1	1.2E-3
品質	0.212	0.328	1

(注)右上はP値、左下はクラメールのV値

表3 説明変数(質的データ)

分類	データ項目	分類	データ項目
開発プロジェクト全般	新技術利用	ユーザ要求管理 (続き)	ユーザ担当者_システム経験
	役割分担_責任所在		ユーザ担当者_業務経験
	達成目標_優先度_明確度合		ユーザとの役割分担_責任所在_明確度合
	作業スペース		要求仕様_ユーザ承認有無
	プロジェクト環境_騒音		ユーザ担当者_設計内容理解度
計画の評価	計画の評価(コスト)		設計_ユーザ承認有無
	計画の評価(品質)		ユーザ担当者_受け入れ試験関与
	計画の評価(工期)		要求レベル_信頼性
利用局面	業種(大分類)		要求レベル_使用性
システム特性	アーキテクチャ		要求レベル_性能・効率性
	主開発言語		要求レベル_保守性
開発の進め方	類似プロジェクト_有無		要求レベル_移植性
	プロジェクト管理ツール_利用		要求レベル_ランニングコスト要求
	構成管理ツール_利用		要求レベル_セキュリティ
	設計支援ツール_利用		法的規制有無
	ドキュメント作成ツール_利用		PMスキル
	デバッグ_テストツール_利用		要員スキル_業務分野経験
	CASE ツール_利用		要員スキル_分析・設計経験
	コードジェネレータ_利用		要員スキル_言語・ツール_利用経験
開発方法論_利用	要員スキル_開発プラットフォーム_使用経験		
ユーザ要求管理	要求仕様_明確度合	要員等スキル	要員スキル_開発プラットフォーム_使用経験
	ユーザ担当者_要求仕様関与	要員等スキル	テスト体制
		要員等スキル	定量的出荷品質基準_有無

タでは、危険率 1%や 5%で有意な結果だけを採用するとほとんどの説明変数が「影響を与えていない」という結果になることが多いためである。P 値が 10%以下のものについては、その説明変数が目的変数に与える影響度合いを示す寄与率を計算する。

3. 2 説明変数が質的データの場合の分析方法

各説明変数に対する回答数は必ずしも多くはないので、目的変数だけでなく説明変数も 2 つのレベルに集約して 2 × 2 のクロス集計表を作成して分析する。クロス集計表は χ^2 分布を仮定して χ^2 検定を行う(文献[4]9.2 参照)。しかし、欄の中に極端に少ない数が含まれている場合は、 χ^2 分布の仮定が成立するとは言えないため、Fisher の直接計算法を用いる[6]。ここでは 4 つのすべての欄が 5 以上であるときは χ^2 検定を行い、少なくとも 1 つの欄が 4 以下のときは Fisher の直接計算法を用いる。質的データの場合も量的データと同様に危険率 1%、5%、10%を有意性の判定基準とする。

4. 分析結果

分析の結果得られた、QCD いずれかの目的変数に対する P 値が 10%以下の説明変数(影響要因の候補)とその P 値および寄与率(量的データの場合)また

は V 値(質的データの場合)を表 4 に示す。また、P 値が 5%以下のもの及び寄与率が 10%以上のものに対しては、計算の根拠となった統計量を表 5~表 8 に示す。

4. 1 量的データの影響要因の候補

分散分析の結果、危険率 5%以下(5%および 1%)で有意な結果が得られたのは、目的変数が実績の評価(コスト)で説明変数が①規模、②工数、③工期、④バグ数、⑥工期当りの規模の場合と、目的変数が実績の評価(工期)で説明変数が⑦規模当りのバグ数の場合だけであった。危険率を 10%にまで広げた場合、実績の評価(品質)と⑦規模当りのバグ数および実績の評価(工期)と④バグ数に関連があるという結果が得られた。

4. 2 質的データの影響要因の候補

44 個の説明変数のうち、開発方法、主開発言語を除いて 21 個が、少なくともいずれかの目的変数と 10%以下(危険率 1%、5%、10%)の P 値を持つことがわかった。ただし、コスト超過、納期遅延、品質低下を引き起こす、あるいはそれらと関連のある説明変数は、次に示すように目的変数ごとに異なる。(1) コスト超過と関連のある説明変数は、危険率 5%

表4 プロジェクトの遂行結果と関連のある説明変数の寄与率(またはV値)とP値

分類	説明変数(要因の候補)		コスト超過	納期遅延	品質低下
外部要件	規模		86(2.3)	-	-
	工期		54(1.3)	-	-
	工期当りの規模		11.1(1.7)	-	-
	要求レベル	要求仕様の明確度合	-	0.27(2.7)	- (9.4)
要求レベル(移植性)		-	- (6.8)	0.6 (67.5)	
要求レベル(性能・効率性)		-	-	0.53(2.5)	
開発側で管理可能な項目	計画の評価	計画の評価(コスト)	-	0.15(4.9)	-
		計画の評価(工期)	-	0.34(0.001)	-
	プロジェクト体制	役割分担_責任所在	- (5.4)	-	-
		達成目標_優先度_明確度合	-	-	- (9.8)
		PMスキル	-	-	- (9.3)
		要員スキル(業務分野経験)	-	0.43(0.008)	0.40(4.2)
		要員スキル(開発プラットフォーム使用経験)	- (8.2)	-	-
		テスト体制(スキル&員数)	0.45(3.8)	0.46(2.6)	0.58(1.2)
		テスト体制(スキルの有無)	-	- (9.3)	0.58(1.2)
	ツールの利用	テスト体制(員数の充足性)	- (6.0)	0.56(1.2)	- (7.0)
		類似プロジェクト_有無	- (6.1)	-	-
		プロジェクト管理ツール_利用	-	-	- (9.8)
		構成管理ツール_利用	-	-	- (7.8)
遂行結果	工数		5.8(0.04)	-	-
	バグ数		8.2(0.3)	2.9(8.0)	-
	規模当りのバグ数		-	14.4(3.9)	27.3(6.3)

(注)量的データの数値は、寄与率[単位:%](P値[単位:%])、質的データの数値は、クramerのV値(P値[単位:%])である。

表5 品質低下に影響を及ぼす要因のクロス集計表

要因	項目	品質低下		合計
		なし	あり	
要求レベル_性能・効率性	極めて高い+高い	8	1	9
	中位(+低い)	4	7	11
	合計	12	8	20
要求レベル_移植性	極めて高い+高い	10	2	12
	中位+低い	1	6	7
	合計	11	8	19
要員スキル_業務分野経験	全員が十分な経験+半数が十分な経験、残り半数(はい)からの経験	22	4	26
	半数が(はい)からの経験、残り半数は経験なし+全員が経験なし	3	4	7
	合計	25	8	33

で有意となるものはテスト体制だけである。危険率 10%で有意となるものとして、役割分担_責任所在(の明確さ)と要員の開発プラットフォーム

利用経験、類似プロジェクトの利用がある。

(2) 納期遅延と関連のある説明変数は、危険率1%で有意となるものとして、計画の評価(工期)、要員スキル(業務分野経験)があり、危険率5%で有意となるものとして、要求仕様の明確度合、計画の評価(コスト)、テスト体制がある。危険率10%で有意となるものとして、要求レベル(移植性)、テスト_デバッグツールの利用がある。

(3) 品質低下と関連のある説明変数は、危険率1%で有意となるものとして、要求レベル(移植性)があり、危険率5%で有意となるものとして、要求レベル(性能・効率性)、要員スキル(業務分野経験)、テスト体制がある。危険率10%で有意となるものとして、要求仕様の明確度合、達成目標_優先度_明確度合、PMスキル、プロジェクト管理ツール_構成管理ツール_テスト_デバッグツールの利用がある。

5. 考察

ここでは4. における結果に基づいて、プロジェ

表 6 納期遅延の有無と要因の候補のクロス集計表

要因の候補	回答	納期遅延		合計
		なし	あり	
要求仕様明確度合	非常に明確+かなり明確	33	6	39
	ややあいまい+非常にあいまい	17	11	28
	合計	50	17	67
計画の評価(コスト)	コスト算定の根拠が明確で実行可能性を検討済み(*1)	114	27	141
	コスト算定の根拠が不明確、または実行可能性を未検討(*2)	13	8	21
	合計	127	35	162
計画の評価(工期)	工期計画の根拠が明確で実行可能性を検討済み(*1)	119	24	143
	工期計画の根拠が不明確、または実行可能性を未検討(*2)	8	12	20
	合計	127	36	163
要員スキル(業務分野経験)	全員が十分な経験+半数が十分な経験、残り半数はいくらかの経験	60	11	71
	半数がいくつかの経験、残り半数は経験なし+全員が経験なし	5	9	14
	合計	65	20	85

(*1)「計画あり」とみなす

(*2)「計画なし」とみなす

クトの遂行に影響を与えそうな要因を最も有力な候補から順に検討を加えることにより、真の影響要因を推定する。特に筆者らの経験に照らして疑わしい候補については、分析対象となったプロジェクトデータを調査し、他の要因の影響がないか再検討する。

5. 1 外部要因の影響

(1) コスト超過の要因

コスト超過と 5%以下の危険率で有意な関連の見られる量的データを調べてみると、コスト超過を起

表 7 実績の評価に対する量的データの平均と分散

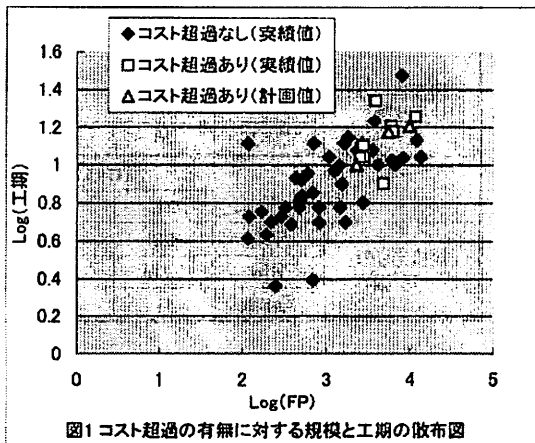
		コスト超過	
		なし	あり
規模	平均	3.00	3.43
	分散	0.289	0.251
	標本数	50	10
工数	平均	4.09	4.48
	分散	0.373	0.253
	標本数	172	37
工期	平均	0.84	1.00
	分散	0.055	0.090
	標本数	97	17
バグ数	平均	2.31	2.74
	分散	0.351	0.288
	標本数	84	21
工期当りの規模	平均	2.11	2.51
	分散	0.205	0.045
	標本数	43	8
		納期遅延	
		なし	あり
規模当りのバグ数	平均	-0.81	-0.37
	分散	0.137	0.141
	標本数	26	4
		品質低下	
		なし	あり
規模当りのバグ数	平均	-0.87	-0.36
	分散	0.119	0.008
	標本数	11	2

表 8 説明変数がテスト体制のクロス集計表

要因	項目	コスト超過			納期遅延			品質低下		
		なし	あり	合計	なし	あり	合計	なし	あり	合計
スキルと員数	スキル、員数ともに十分	23	2	25	21	4	25	17	3	20
	上記以外	3	3	6	2	4	6	1	4	5
	合計	26	5	31	23	8	31	18	7	25
スキル	スキルは十分	23	3	26	21	5	26	17	3	20
	スキルは不十分	3	2	5	2	3	5	1	4	5
	合計	26	5	31	23	8	31	18	7	25
員数	員数は十分	25	3	28	23	5	28	18	5	23
	員数は不十分	1	2	3	0	3	3	0	2	2
	合計	26	5	31	23	8	31	18	7	25

(注 1)「テスト体制(スキル&員数)-品質低下」と「テスト体制(スキルの有無)-品質低下」は同一データである。

こうしたプロジェクトはそうでないプロジェクトよりも、明らかに開発規模と工数が大きく工期が長い。バグ数も多い。すなわちプロジェクト規模が大きい。しかし、開発規模で正規化した量で比較すると、生産性および規模当りのバグ数では差が見られず、工期当りの規模にのみ有意な差が見られる。工期当りの規模の寄与率も 11.1%と量的要因の中で最も高い。このことから、コスト超過は大規模プロジェクトで起きやすく、特にこれを避けるために最も注意すべきことは、「工期当りの規模」を適正にしておくことであると言える。ただし、この結果は実績値に基づいたものであり、計画段階でリスク要因としてあげることができるかどうかは、計画値で分析する必要がある。コスト超過を起こしたプロジェクトで規模と工期の両方の計画値が揃っているものは3件しかなかったが、規模と工期の散布図を描いてみると(図1)、上記の結論を揺るがすものではないことがわかる。なお、1ヶ月当たり200FP以下のプロジェクトではコスト超過は起きていなかった。また、コスト超過を起こしたプロジェクト群で業種・アーキテクチャの偏りは見られなかった。



(2) 納期遅延および品質低下の要因

量的データでは、規模当りのバグ数のみか納期遅延と品質低下のいずれとも統計的に有意な関連が見られる。規模当りのバグ数の増加は品質低下そのものであるが、それにより納期遅延も起きやすくなることがわかる。

質的データに関しては、要求仕様が明確なプロジェクトに比較して要求仕様があいまいなプロジェクトは、納期遅延を起こす可能性が高く(5%有意)、品質低下を起こす傾向(10%有意)も見られる(表4)。これは筆者らの直感に合っている。

一方、移植性に対する要求レベルが高いと品質低下を起こす可能性が低く(1%有意)、納期遅延を「起こさない」傾向が見られる(10%有意)という結果(表4)に対しては直感と合わない。そこで再度該当するプロジェクトデータを見直したところ、移植性に対する要求レベルが低いプロジェクトは、要員スキルも低い傾向があることがわかったので、新たに「プロジェクト能力」という測定量を導入して分析を行った。ここで「プロジェクト能力」とは、「PMスキル」と4つの「要員スキル」のレベルを高い方から4~1までの点数を付けて平均をとることにより、プロジェクトの能力をレベル化したものである。すべてが最高レベルのプロジェクトは4.0、すべてが最低レベルのプロジェクトは1.0と評価される。移植性に対する要求レベルに対して品質低下の影響を調べたデータ18件(1件は要員スキルデータが欠損していたので除外)をプロジェクト能力の観点から分析した結果を表9に示す。表9に示すように、この18件のデータでは「プロジェクト能力」の低さが品質低下を起こしたとも言える。ユーザの要求レベルが低いのでプロジェクト能力の低いプロジェクトチームを構成し、その結果として品質低下を起こしたと解釈するのが妥当と思われる。

移植性に対する要求レベルの場合と同様に、性能・効率性の要求レベルの低いプロジェクトはプロジェクト能力が低いという結果が得られ、性能・効率性に対する要求レベルが低いプロジェクトは品質低下を起こしやすい(5%有意)という結論は採用しない方がよいと思われる。

表9 要求・移植性の分析データのプロジェクト能力

	品質低下	
	なし	あり
平均	3.21	2.65
分散	0.234	0.329
標本数	10	8
P値 = 3.9%、寄与率 = 24.0%		

5. 2 プロジェクトの努力と遂行能力の向上

表4における「開発側で管理可能な項目」によって、プロジェクトの遂行に影響を与える外部要因を緩和できると考えられる。

(1) 計画時のQCDの評価と納期遅延防止

計画の評価には、コスト・工期・品質に関するものがあるが、これらは、「納期遅延」に対して最も顕著な効果を発揮している(表4)。

- ・ 工期の評価をきちんと行ったプロジェクトが納期遅延を起こす割合は約20%、そうでないプロジェクトは60%と顕著な違いが見られる(表6)。
- ・ コストの評価をきちんと行ったプロジェクトは、コスト超過に対するよりも納期遅延に対して効果

を發揮しているように見える。しかし、表6からわかるように計画の評価（コスト）と計画の評価（工期）のクロス集計表は似ているので、改めてその差を分析したところ152件のプロジェクトでコスト評価の有無と工期評価の有無が一致していた（表10）。つまり、90%以上のプロジェクトでは納期遅れがコスト評価の有無の結果によるものか工期評価の有無によるものかわからない。コスト評価の有無と工期評価の有無が異なるプロジェクトを調べてみると、工期を評価したがコストは評価しなかったプロジェクトは6件のみであり、これらはすべて納期遅延を起こしていなかった。また、コストを評価したが工期は評価しなかったプロジェクトは4件あり、そのうち3件は納期遅延を起こしていた。このことから納期遅延を起こさなかったのはあくまで工期の評価を行わなかったためであると言える。

一方、コスト評価がコスト超過に対する効果があるかどうかの統計的な有意性は11%であって、一般の統計的な基準から

表10 コストと工期に対する計画の評価の有無と納期遅れの関係

計画の評価	納期遅れ	
	なし	あり
あり	あり	113
	なし	3
なし	あり	0
	なし	8
未回答	なし	1

すると有意とは言えないが、142件中26件(18%)のコスト超過なしと21件中7件(33%)のコスト超過ありという結果が得られており、これはプロジェクト管理者にとっては無視できない数値と思える。計画の評価がプロジェクトの順調な遂行に重要な役割を果たしていると考えてよい。

(2) プロジェクト体制の影響

プロジェクトを組む際に優秀なプロジェクト管理者・経験豊富な要員を揃えるのは理想であるが、現実にそのようなプロジェクトを常に組織できるとは限らない。ここでは、プロジェクト能力の側面とプロジェクト遂行能力の関係について検討する。

- ・要員の業務分野経験が豊富なプロジェクトは、納期遅延と品質低下をともに起こしにくい（それぞれ1%有意と5%有意）。
- ・役割分担・責任所在が明確なプロジェクトおよび要員の開発プラットフォーム使用経験が豊富なプロジェクトはいずれもコスト超過を起こしにくい傾向が見られる（いずれも10%有意）。

- ・達成目標_優先度_明確度合の高いプロジェクトおよびプロジェクト管理者のスキル（PMスキル）の高いプロジェクトは、いずれも品質低下を起こしにくい傾向が見られる（いずれも10%有意）。

(3) テスト体制の効果

- 優れたテスト体制はコスト・納期・品質いずれの面からもプロジェクトの遂行によい結果をもたらす。
- ・スキル・員数とも十分なプロジェクトはそうでないプロジェクトに比較してコスト超過・納期遅延・品質低下のいずれをも起こしにくい(5%有意)。
- ・テスト要員数が充足していると、品質低下より納期遅延の防止に効果が見られる(5%有意)。
- ・テスト要員にスキルがあると納期遅延より品質低下の防止に効果が見られる(5%有意)。

(4) ツールの利用

- ツールの利用はプロジェクト遂行によい影響を与える傾向が見られる。ただし、利用するツールによって効果の見られるQCDが異なる(表4)
- ・コスト超過防止効果のありそうなものは、「類似プロジェクトの存在」である(10%有意)。
- ・納期遅延防止に効果のありそうなものは、「デバッグ・テストツールの利用」である(10%有意)。
- ・品質低下防止に効果のありそうなものは、「プロジェクト管理ツールの利用」、「構成管理ツールの利用」、「デバッグ・テストツールの利用」である(いずれも10%有意)。これらツール利用が品質低下に効果のありそうなことが伺える。

5. 3 業種・主開発言語の違いによる影響

業種、主開発言語は、分類尺度の選択肢が多く、かつ件数の少ないものが多い。そこで件数の多いもののみを選択して有意性の検証を行った。

表11 業種(大分類)ごとのコスト超過を起こすプロジェクト数

業種(大分類)	コスト超過		
	なし	あり	合計
製造業	31	7	38
情報通信業	20	1	21
卸売・小売業	25	5	30
金融・保険業	42	15	57
公務	23	1	24
合計	141	29	170

(1) 業種(大分類)別

情報通信業に比較して金融・保険業がコスト超過を起こす割合が高い(表11)。この結果は危険率5%で有意であるが、これは金融・保険業が情報通信業よりプロジェクト規模が大きい[4]ためと考えた方がよい。

(2) 開発言語

VB>C言語>>Java>COBOLの順にコスト超過を起こしにくく、COBOL>C言語>Java>>VB

の順に品質低下を起こしにくいという結果が得られた(表 12)。これは言語自体の性質ではなく、その言語を利用したプロジェクトの業種や性格によると考える。例えば、プロジェクト規模の大きい(恐らく品質要求は厳しい)金融・保険業で COBOL が用いられていることに関係していると考えられる。なお、表 4 に示されていないように、ユーザ担当者の関与の程度やアーキテクチャの違いからは有意な結果は得られなかった。

表 12 主開発言語ごとのコスト超過と品質低下を
起こすプロジェクト数

主開発言語	コスト超過			品質低下		
	なし	あり	合計	なし	あり	合計
COBOL	26	8	34	10	2	12
C 言語	28	2	30	14	3	17
VB	26	1	27	4	6	10
Java	40	10	50	14	4	18
合計	120	21	141	42	15	57

5. 4 典型的プロジェクトの比較

プロジェクトを順調に遂行したプロジェクト群と遂行できなかったプロジェクト群それぞれの典型的なプロジェクトの比較結果を表 13 に示す。ここで典型的なプロジェクトとは、各プロジェクト群の(対数変換後の)平均値を持つプロジェクトとした。もとの単位に戻すと、いずれも比較すべき2つの値に大きな差があることがわかる。ただし、説明変数ごとに標本集団が異なることから、説明変数間での厳密な一致性は必ずしも見られない。

6. おわりに

エンタープライズ系ソフトウェア開発におけるプロジェクトの遂行でコスト・納期・品質目標が当初計画通りに達成できなかった要因を分析し、次のことを明らかにした。

- (1) コスト超過は大規模プロジェクトで起きやすく、これを避けるためには、特に「工期当りの規模」を適正にしておくことが効果的であると思われる。
- (2) 要求仕様があいまいなプロジェクトは、納期遅延を起こす可能性が高く(5%有意)、品質低下を起こす傾向(10%有意)も見られる。
- (3) 計画の評価がプロジェクトの順調な遂行に重要な役割を果たしている。特に、工期の評価をきちんと行ったプロジェクトが納期遅延を起こす割合は約 20%、そうでないプロジェクトは 60%と顕著な違いが見られる(1%有意)。
- (4) 要員の業務分野経験が豊富なプロジェクトは、納期遅延と品質低下を起こしにくい(それぞれ 1%

有意と 5%有意)。

- (5) スキル・員数とも十分なテスト要員を確保したプロジェクトはそうでないプロジェクトと比較してコスト超過・納期遅延・品質低下のいずれをも起こしにくい(5%有意)。テスト要員数が充足していると、品質低下に比べて納期遅延の防止に効果が見られ(5%有意)、テスト要員にスキルがあると納期遅延に比べて品質低下の防止に効果が見られる(5%有意)。

表 13 プロジェクト遂行の成否ごとの
典型的プロジェクトの比較

量的データ	単位	コスト超過	
		なし	あり
規模	FP	1,000	2,694
工数	人時	12,407	29,921
生産性	人時/FP	13.0	14.2
工期	月	6.8	10.0
工期当りの規模	FP/月	127	325
バグ数	件	204	550
		納期遅延	
		なし	あり
規模当りのバグ数	件/FP	0.16	0.42
		品質低下	
		なし	あり
規模当りのバグ数	件/FP	0.13	0.44

謝辞 本研究は、経済産業省の支援により、東海大学と IPA ソフトウェア・エンジニアリング・センターが共同で実施したものである。定量データ分析部会委員各位及び SEC 研究員である横山健次、本間周二両氏のご協力に感謝致します。

参考文献

- [1] Kemerer, C. F. and Porter, B. S., "Improving the Reliability of Function point Measurement: An Empirical Study, IEEE Trans. Software Eng., Vol. 18, No. 11, pp. 1011-1024, 1992.
- [2] B. W. Boehm et al., Software Cost Estimation with COCOMO II, Prentice Hall PTR, 2000.
- [3] 角田雅照, 大杉直樹, 門田暁人, 松本健一, 佐藤慎一, "協調フィルタリングを用いたソフトウェア開発工数予測方法", 情報処理学会論文誌, Vol.46, No. 5, pp.1155-1164, 2005.
- [4] 独立行政法人 情報処理推進機構 ソフトウェア・エンジニアリング・センター著作監修: ソフトウェア開発データ白書 2006, 日経 BP 社, 2006.
- [5] ISBSG, "The Benchmark Release 8", ISBSG, 2004.
- [6] 氏家勝巳, 土井誠: 統計数学序論, 東海大学出版会, 2001.