

CMMI成熟度レベル別に見た ソフトウェア品質の良否にかかわる 要因の複合的分析

柳田 礼子^{※1}野中 誠^{※2}誉田 直美^{※1}

ソフトウェア品質の良否に影響する要因は、通常単一ではない。要因の一つとしてCMMIで示される開発組織の能力があるが、アセスメント結果のみからキーとなる要因を探り当てることは難しい。ソフトウェア品質を効率的に向上させるためには、各成熟度レベルにおいて、実績データを基に分析した結果を含めて考察する必要がある。本論文では、商用ソフトウェア開発プロジェクト522件のデータを対象に、成熟度レベル別に分類木を構築し、有意差検定と相関分析を組み合わせるソフトウェア品質の良否に影響する要因を複合的に分析した。その結果、成熟度レベル1では開発規模、上工程バグ数/KL、及び上工程バグ摘出率が良否を分ける要因と示された。成熟度レベル2では開発規模とテスト工程バグ数/KLが要因と示され、開発規模は成熟度レベル1の約4倍の境界値となった。成熟度レベル3では有意差検定の結果からテスト工程バグ数/KLが主たる要因と示された。成熟度レベル3では、テスト後半にバグが残存している場合にテスト工数が十分にかげられずに出荷後の品質が悪くなる傾向が確認された。

A Compound Factor Analysis by CMMI Maturity Levels related to Success and Failure on Software Quality

Reiko Yanagida^{※1}, Makoto Nonaka^{※2}, Naomi Honda^{※1}

Organizational capability measured with CMMI is a promising factor to deliver high-quality software. To improve the maturity level of organizational capability, we should identify factors and their relationships affecting software quality by maturity levels. In this paper, the authors analyzed 522 commercial software development projects by using a classification tree method, statistical significance tests and correlational analysis. The result showed that development size, defect detection density in reviews, and defect detection rate before testing were identified as major factors in maturity level 1. In maturity level 2, development size and defect detection density in testing were identified as major factors, but the boundary of development size is four times larger than that in maturity level 1. In maturity level 3, defect detection density in testing was identified with significance tests. In this maturity level, the failed projects had more bugs remained in the latter stages of testing because of lack of testing effort.

※1 日本電気株式会社 ※2 東洋大学

1 はじめに

ソフトウェア品質の良否に影響する要因の一つとしてCMMI(Capability Maturity Model Integration)^[1]で示される開発組織の能力がある。Jones^[2]は成熟度レベルが高いほどリリース後のソフトウェア欠陥密度が低くなることを示している。また、成熟度レベルがソフトウェア品質の良否に影響することが先行研究により示されている^{[3][4][5]}。

CMMIは段階的なプロセス改善を促すモデルだが、成熟度レベルを上げるのには一般に長い期間を要する^[6]。そのため、成熟度の状況に応じたプロセス改善を効率的に進めるには、それぞれの成熟度レベルにおいてソフトウェア品質の良否に影響する要因を把握することが求められる。

そのような要因について成熟度レベル別に分析した研究が幾つかあるが、それらには課題がある。Rainerら^[7]は成熟度レベル別にプロセス改善の成功要因を示している。しかし、それらは定量的なプロジェクト管理に結びつく要因を示していない。Yanagidaら^[8]は開発規模や上工程バグ抽出率などのメトリクスを用いて成熟度レベル別に品質改善の要因を導出している。しかし、メトリクスは「値が大きいほど良い」のように一元的に判断できないことが多く、複合的に分析する必要がある。また、Yanagidaら^[8]は成熟度レベル別に相関分析をしているが、セグメントを細分化した分析はしていない。

本論文では、筆者らの所属部門が管轄する組織から得た商用ソフトウェア開発プロジェクト522件のデータを、成熟度レベル別に見たソフトウェア品質の良否にかかわる要因を複合的に分析した結果を報告する。本論文では、ソフトウェア品質会計^[9]と呼ばれる定量的管理の仕組みを通して収集されたメトリクスのデータを分析対象とし、CARTアルゴリズムに基づいて分類木を構築し、その上でほかのメトリクスも含めた有意差検定及び相関分析を用いる。

2 分析対象データ

2.1 対象組織の背景

筆者らの所属企業ではSW-CMM^[10]及びCMMIに基づくプロセス改善に継続的に取り組んでおり、成熟度レベル1から5までの多様な開発組織がある。また、ソフトウェア

品質会計^[9]と呼ばれる定量的管理の仕組みを30年以上にわたって展開しており、成熟度レベルの低い組織であっても一定の定量的管理が行われている。

開発プロセスはV字モデルに基づいており、要件定義からコードレビューまでを上工程、単体テストからシステムテストまでをテスト工程と呼んでいる。

2.2 対象組織とプロジェクト数

本論文では、筆者らの所属企業内において開発対象や背景などが似ている、システムインテグレーション事業にかかわる組織を分析対象とする。これらの組織は、成熟度レベル1から3の組織がある。ソフトウェア品質の良否は、リリース後バグ密度[b]の実績値をそれぞれの組織があらかじめ定めた品質基準と比較して、「達成」または「未達」に分類した。

本論文では、分析対象の組織において2014年度に開発を完了した522件の商用ソフトウェア開発プロジェクトを対象とした。成熟度レベル別のプロジェクト件数の内訳を表1に示す。表1では、「達成」と「未達」のプロジェクト件数内訳を都合により非掲載としているが、成熟度レベル1の達成率(「達成」件数をプロジェクト合計件数で割った値)を基準としたポイントの差の値を示している。成熟度レベルが上がるにつれてその比率が向上していることが分かる。なお、表1は外れ値を除去した後の件数である。

表1 分析対象のデータ(外れ値除去後)

レベル	プロジェクト数	レベル1との達成率の差
1	317	—
2	138	+13.3pt.
3	67	+21.8pt.

2.3 メトリクス

メトリクスの一覧を表2に示す。これらを選んだ理由は、ソフトウェア品質会計^[9]に基づく品質管理において着目することの多い主要なものであるためと、欠損率が比較的低いためである。欠損率については2.4項で詳しく述べる。

なお、本論文で分析結果を示すときには、実測値ではなく相対値を用いる。相対値は、それぞれの値を対応するメトリクスの全体の平均値で割った値とする。

表2 分析に用いたメトリクス

記号	メトリクス	単位	意味
SIZE	開発規模	KL	新規作成または変更した論理ソースコード行数
UpBD%	上工程バグ抽出率	%	全バグ数に対する上工程抽出バグ数の比率
UpEff	上工程工数/KL	人時/KL	上工程工数を開発規模で割った値
TstEff	テスト工程工数/KL	人時/KL	テスト工程工数を開発規模で割った値
UpRvEff	上工程レビュー工数/KL	人時/KL	上工程レビュー工数を開発規模で割った値
TstItem	テスト項目数/KL	件/KL	すべてのテスト項目数を開発規模で割った値
UpBug	上工程バグ数/KL	件/KL	上工程で抽出したバグ数を開発規模で割った値
TstBug	テスト工程バグ数/KL	件/KL	テスト工程で抽出したバグ数を開発規模で割った値

※KLはKilo Lines of Codeの略

2.4 成熟度レベル別のメトリクスの値

成熟度レベル別のメトリクスの中央値を、「達成」と「未達」に分けて表3に示す。開発規模とテスト工程バグ数/KLは「達成」のほうが一貫して小さい。一方、上工程バグ抽出率は「達成」のほうが一貫して大きい。

成熟度レベル別の各メトリクスの欠損率を表4に示す。成熟度レベル1の欠損率の高さが目立つが、2.1節で述べたようにソフトウェア品質会計の効果もあってバグ数にかかわるメトリクスの欠損率は比較的強く抑えられている。

表3 成熟度レベル別のメトリクスの中央値

メトリクス	レベル1		レベル2		レベル3	
	達成	未達	達成	未達	達成	未達
SIZE	.309	.662	.184	.661	.144	.217
UpBD%	1.071	.996	.986	.950	.983	.942
UpEff	.708	.537	.773	.776	.835	.834
TstEff	.446	.344	.778	.968	.650	.845
UpRvEff	.757	.620	.647	.737	.631	.477
TstItem	.626	.537	.534	.620	.951	.634
UpBug	1.084	.822	.595	.787	.672	1.220
TstBug	.595	1.142	.773	1.142	.744	1.542

表4 成熟度レベル別の各メトリクスの欠損率

メトリクス	レベル1	レベル2	レベル3
SIZE	0.0%	0.0%	0.0%
UpBD%	14.2%	2.9%	1.5%
UpEff	63.1%	0.0%	5.9%
TstEff	65.0%	0.0%	4.4%
UpRvEff	67.8%	0.7%	5.9%
TstItem	9.7%	0.0%	0.0%
UpBug	9.8%	0.0%	1.5%
TstBug	10.1%	0.0%	0.0%

3 分類木による分析結果

3.1 成熟度レベル1

成熟度レベル1の分類木を図1に、各ノードの境界値と達成率を図2に示す。分類木の葉ノードの縦軸に示した値は「達成」と「未達」を示しており、それぞれ1と0である。図2の項番は図1のノード番号と対応しており、*印は葉ノードであることを示している。

最初のノードは開発規模であり、その境界値は平均値の0.101すなわち約1/10となった。図2のノード1と7にある通り、開発規模が全体の平均値の1/10未満だと達成率が91.0%と高いが、これ以上の開発規模では達成率が61.6%に低下している。成熟度レベル1では、小規模開発であることが「達成」に寄与する第一の要因と言える。

続いて、上工程バグ数/KLが平均値の0.349すなわち約1/3未満だと達成率が90.2%だが、これ以上になると達成率が56.0%へと低下している。上工程での抽出バグ数が少ないことは、上工程での混入バグ数が少ない、または、上工程でのバグの抽出が不十分であることが考えられる。しかし、更に上工程バグ数が平均値の0.349以上のノードにおいては、上工程バグ抽出率が平均値の0.922以上だと達成率が62.0%だが、これ未満になると達成率が32.6%へと大きく低下している。これらより、開発規模が小規模でないプロジェクトにおいて、上工程でのバグ抽出が不十分とは考えにくい。上工程でのバグ混入数を低い水準に抑えること、すなわち設計品質の良さが「達成」に寄与する要因と言える。

また、開発規模が小規模でなく、設計品質が悪いプロジェクトでは、上工程バグ抽出率を平均値に近い値以上に維持することが「達成」に寄与すると言える。

以上の分析結果はいずれもソフトウェア品質管理における筆者らの経験則と整合しており、合理的な分析結果と言える。

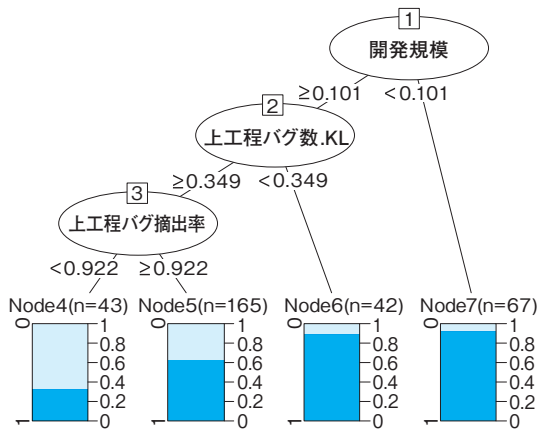


図1 成熟度レベル1：分類木

[1] 開発規模 ≥ 0.101 (61.6%)
[2] 上工程バグ数/KL ≥ 0.349 (56.0%)
[4] 上工程バグ抽出率 < 0.922 (32.6%)*
[5] 上工程バグ抽出率 ≥ 0.922 (62.0%)*
[6] 上工程バグ数/KL < 0.349 (90.2%)*
[7] 開発規模 < 0.101 (91.0%)*

図2 成熟度レベル1：分類木の各ノードの境界値と達成率

3.2 成熟度レベル2

成熟度レベル2の分類木を図3に、各ノードの境界値と達成率を図4に示す。成熟度レベル1と同様、最初のノードに開発規模が示されたが、その境界値は平均値の0.463であり、レベル1の約4倍の値となった。図4のノード1と7にある通り、開発規模がこの境界値未満だと達成率が92.5%と高いが、これ以上になると達成率が57.8%に低下している。成熟度レベル2では、レベル1よりは大きな規模まで対応できるものの、中規模までの開発であることが「達成」に寄与する第一の要因と言える。

続いて、テスト工程バグ数/KLが平均値の0.869未満だと達成率が76.0%だが、これ以上だと達成率が35.0%へと低下している。テストでの抽出バグ数が少ないことは、上工程で品質が確保されてテスト開始時点で残存しているバグ数が少ない、または、テストでのバグの抽出が不十分であることが考えられる。しかし、これらのノードにおけるテスト工数の中央値を比較すると、テスト工程バグ数/KLが平均値の0.869未満のノードの値が大きく、有意差がある。この結果は4.2節で詳細に述べるが、このことから、

開発規模が中規模より大きいプロジェクトにおいて、テスト工程でのバグ抽出が不十分とは考えにくい。テストでの抽出バグ数を平均値のやや下という水準に抑えられていること、すなわち、上工程で一定の品質が確保されていることが「達成」に寄与する要因と言える。

ここまでの分析結果は経験則と整合しており、とくにレベル1の4倍という開発規模の境界値は興味深い。しかしその次には、上工程レビュー工数/KLが平均値の0.403未満だと達成率が62.5%だが、これ以上になると達成率が16.7%へと大きく低下するという、経験則とは逆の関係を示す結果が得られた。ただし、ノード3はデータ件数が20件と少ないため、より多くのデータでの検証が必要である。

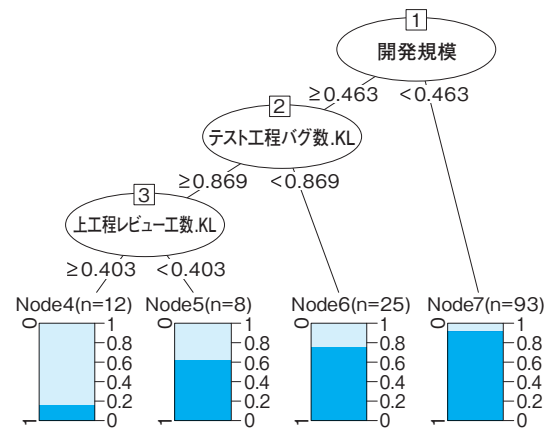


図3 成熟度レベル2：分類木

[1] 開発規模 ≥ 0.463 (57.8%)
[2] テスト工程バグ数/KL ≥ 0.869 (35.0%)
[4] 上工程レビュー工数/KL ≥ 0.403 (16.7%)*
[5] 上工程レビュー工数/KL < 0.403 (62.5%)*
[6] テスト工程バグ数/KL < 0.869 (76.0%)*
[7] 開発規模 < 0.463 (92.5%)*

図4 成熟度レベル2：分類木の各ノードの境界値と達成率

3.3 成熟度レベル3

成熟度レベル3では分類木が構築されなかった。これは、達成率に偏りのあるデータであることと、プロジェクト数が67件とほかに比べて少ないこと、成熟度レベル3であるためプロセスが標準化されていてメトリクス間に大きな差がなかったことによると考えられる。4節において各メトリクスの有意差検定により「達成」の寄与要因を分析する。

4 ノード別の相関分析

3節で得られた分類木の葉ノードのそれぞれについて、メトリクス間の関係を相関分析により詳しく分析する。

4.1 成熟度レベル1

分類木に現れたメトリクスは開発規模、上工程バグ数/KL、及び上工程バグ摘出率であった。ここで、小規模プロジェクトの成功要因を探ることは実務ニーズとして優先度が低いため、図1のノードの2以降に着目する。そして、ノード3と6を分ける要因と、ノード4と5を分ける要因についてそれぞれ分析する。分類木に現れなかったほかのメトリクスは、ウィルコクソンの順位和検定を用いて中央値についての有意差検定を行う。なお、いずれも帰無仮説を「達成」と「未達」の間で各々のメトリクスの中央値に差がないものとし、対立仮説ではそれらに差があるものとしている。

(1) ノード3と6

ノード間の有意差検定の結果と各ノードにおける中央値を表5に示す。両側検定でp値が5%未満のものと、達成率と中央値の変化が妥当と考えられるものを太字で示している。該当するメトリクスとしてテスト工程工数/KL、テスト項目数/KL、及びテスト工程バグ数/KLが得られた。以降、これらのメトリクスを加えて相関分析を行う。

表6と表7に、ノード3と6の「達成」と「未達」に分けたメトリクス間の相関分析の結果をそれぞれ示す。なお、ノード6の「未達」はデータ数が3件と少ないため分析の対象外とした。表6と表7を比べると、ノード6の「達成」ではテスト項目数/KLとテスト工数/KLにやや強い正の相関があるが、ノード3の「達成」では相関がない。3.1節で述べた通り、ノード6は小規模ではないが設計品質が良く、テスト工数がテスト項目数に比例するという安定したテストが実施できていると考えられる。また、ノード6では上工程バグ数/KLとテスト工程バグ数/KLに相関がないが、ノード3では「達成」と「未達」のいずれにも正の相関がある。すなわち、ノード3では「上工程で多くのバグが摘出されるとテストでも多くのバグが摘出される」という状況だが、ノード6ではそうではない。これはノード6における設計品質の良さの現れと考えられ、3.1節の分析結果と整合する。

表5 成熟度レベル1：ノード3と6の中央値と有意差検定

メトリクス	p値	ノード	
		3	6
達成率	—	56.0%	90.2%
UpEff	.1282	.596	.412
TstEff	.0463	.409	.243
UpRvEff	.0841	.691	.424
TstItem	.0232	.569	.475
TstBug	3.29e-13	1.101	.068

表6 成熟度レベル1：ノード3の相関分析

メトリクス	成否	TstEff	TstItem	UpBug	TstBug
UpBD%	達成	-.049	.129	.131	-.723**
	未達	.022	.095	.126	-.732**
TstEff	達成		.168	-.337*	-.220
	未達		.066	-.172	-.111
TstItem	達成			.084	-.122
	未達			.132	.010
UpBug	達成				.422**
	未達				.502**

**p < 0.01 *p < 0.05

表7 成熟度レベル1：ノード6の相関分析(達成)

メトリクス	TstEff	TstItem	UpBug	TstBug
UpBD%	-.229	-.564**	.189	-.538**
TstEff		.692**	.016	.136
TstItem			-.262	.130
UpBug				.246

**p < 0.01 *p < 0.05

(2) ノード4と5

続いて、ノード4と5を分ける要因について分析する。

ノード間の有意差検定の結果と各ノードにおける中央値を表8に示す。有意差があり、達成率と中央値の変化が妥当と考えられるメトリクスとして上工程レビュー工数/KL、テスト項目数/KL、及びテスト工程バグ数/KLが得られた。以降これらのメトリクスを加えて相関分析を行う。

表9と表10に、ノード4と5について「達成」と「未達」に分けたメトリクス間の相関分析の結果をそれぞれ示す。ノード4の「達成」と「未達」の両方において上工程バグ数/KLと上工程バグ摘出率の間に正の相関があるが、ノード5では相関がない。また、ノード4では上工程バグ摘出率とテスト工程バグ数/KLの間に相関がないが、ノード5では強い負の相関があった。ノード4では、設計品質が悪く、レビュー

での抽出が容易なバグを上工程で数多く混入しており、上工程バグ抽出率もそれに応じて一定水準までは高まっていく。しかし、上工程バグ抽出率の割にはテスト工程でバグを十分に抽出できていない状況が考えられる。

ノード5では、上工程バグ数/KLと上工程バグ抽出率の間には相関がなく、「達成」と「未達」の両方において、上工程バグ数/KLとテスト工程バグ数/KLの間に正の相関がある。設計品質が悪い状況において、レビュー及びテストで一定水準まで抽出できていることが、ノード4と比較すると達成率が高くなっている要因と考えられる。

表8 成熟度レベル1：ノード4と5の中央値と有意差検定

メトリクス	p値	ノード	
		4	5
達成率	—	32.6%	62.0%
UpEff	.2018	.437	.599
TstEff	.6778	.393	.430
UpRvEff	.0070	.504	.803
TstItem	.0376	.486	.642
TstBug	2.53e-12	1.942	.721

表9 成熟度レベル1：ノード4の相関分析

メトリクス	成否	UpRvEff	TstItem	UpBug	TstBug
UpBD%	達成	-.125	-.096	.768**	-.160
	未達	-.060	-.269	.459*	-.337
UpRvEff	達成		.887*	.089	.131
	未達		.260	.449	.416
TstItem	達成			-.307	-.432
	未達			.353	.635**
UpBug	達成				.483
	未達				.642**

**p < 0.01 *p < 0.05

表10 成熟度レベル1：ノード5の相関分析

メトリクス	成否	UpRvEff	TstItem	UpBug	TstBug
UpBD%	達成	.065	.111	-.014	-.735**
	未達	-.170	-.085	-.185	-.757**
UpRvEff	達成		.281	-.037	-.031
	未達		.398	.408	.408
TstItem	達成			.084	-.083
	未達			.071	.075
UpBug	達成				.532**
	未達				.721**

**p < 0.01 *p < 0.05

4.2 成熟度レベル2

分類木に現れたメトリクスは開発規模、テスト工程バグ数/KL、及び上工程レビュー工数/KLであった。ここで、中規模以下のプロジェクトの成功要因を探るのではなく図3のノード2以降に着目する。ただし、ノード4と5はデータの件数が少ないため、ノード3と6に着目して分析する。分類木に現れなかったほかのメトリクスについて、中央値についての有意差検定を行った。ノード間の有意差検定の結果と各ノードにおける中央値を表11に示す。

有意差があり、達成率と中央値の変化が妥当と考えられるメトリクスとして上工程バグ抽出率、テスト工程工数/KL、及び上工程バグ数/KLが得られた。以降、これらのメトリクスを加えて相関分析を行う。

表12及び表13に、ノード3と6について、「達成」と「未達」に分けたメトリクス間の相関分析の結果をそれぞれ示す。相関係数の求め方は4.1節で述べた通りである。各セグメントのデータ数は少ないことに留意が必要である。

ノード3の「達成」と「未達」の両方において、上工程バグ数/KLと上工程バグ抽出率の間に正の相関があるが、ノード6では相関がないことが読み取れる。また、ノード3では上工程バグ抽出率とテスト工程バグ数/KLの間に相関がないが、ノード6の「達成」では負の相関があった。これは成熟度レベル1のノード4と5で読み取れたことと同じであり、同様の状況が生じていると考えられる。

また、ノード6の「達成」において、上工程レビュー工数と上工程バグ抽出率との間に正の相関があり、テスト工程バグ数/KLとの間に負の相関がある。これは、一定量のバグが含まれる成果物に対しては、上工程レビューに十分な工数を投入することによってテストで検出されるバグ数が減少する、という品質管理の観点からすれば期待通りの結果が示されている。別の見方をすれば、このような期待通りの傾向が現れるのは、成熟度レベルが2の中でも一部のセグメントのみと言える。

表11 成熟度レベル2：ノード3と6の中央値と有意差検定

メトリクス	p値	ノード	
		3	6
達成率	—	35.0%	76.0%
UpBD%	.000	67.2	77.4
UpEff	.148	.541	.964
TstEff	.013	.379	1.604
TstItem	.293	.601	.508
UpBug	1.37e-06	1.081	.252

表12 成熟度レベル2：ノード3の相関分析

メトリクス	ノード	TstEff	UpRvEff	UpBug	TstBug
UpBD%	達成	.388	.415	.797*	-.116
	未達	.291	.267	.747**	-.297
TstEff	達成		.987**	.700	.288
	未達		.291	.090	-.046
UpRvEff	達成			.144	-.260
	未達			-.358	-.108
UpBug	達成				.761**
	未達				.840**

**p < 0.01 *p < 0.05

表13 成熟度レベル2：ノード6の相関分析

メトリクス	ノード	TstEff	UpRvEff	UpBug	TstBug
UpBD%	達成	.685**	.515*	.034	-.482**
	未達	.246	.481	.255	-.453
TstEff	達成		.536*	-.323	-.611**
	未達		-.138	-.453	-.501
UpRvEff	達成			-.358	-.581*
	未達			-.082	-.438
UpBug	達成				.840**
	未達				.739

**p < 0.01 *p < 0.05

4.3 成熟度レベル3

表3に示したうち、「達成」と「未達」について有意水準5%で有意差のあるメトリクスはなかった。最もp値が小さかったのはテスト工程バグ数/KLであり、0.087であった。

テスト工程の状況を詳細に分析するため、表14に示すメトリクスを用いて、工程別の値を比較する。各メトリクスの「達成」と「未達」の中央値についての有意差検定を行った結果を表15に示す。

工程別テスト工数/KLについて、「未達」は工程が進むにつれて値が一様に増加しているが、「達成」は減少している。一方、工程別テスト項目数/KLについては、全工程にわたり「未達」のほうが「達成」よりも値が低い。とくに「未達」のIT工程テスト項目数/KLは、有意差があるとは言えないが低い値となっている。工程別テスト工程バグ数/KLは、全工程にわたって「達成」よりも高い値である。とくにST工程テストバグ数/KLは有意差があることが示された。

表14 工程別分析に用いたメトリクス

メトリクス	単位	意味
工程別テスト工数/KL	人時/KL	UT,IT,ST工程ごとの、各テスト工程工数を開発規模で割った値
工程別テスト項目数/KL	件/KL	UT,IT,ST工程ごとの、各テスト工程のテスト項目数を開発規模で割った値
工程別テストバグ数/KL	件/KL	UT,IT,ST工程ごとの、各テスト工程で抽出したバグ数を開発規模で割った値

※UT：単体テスト IT：結合テスト ST：システムテスト

表15 工程別の中央値と有意差検定

メトリクス	p値	達成	未達
UT工程テスト工数/KL	.945	.677	.532
IT工程テスト工数/KL	.812	.488	.587
ST工程テスト工数/KL	.338	.456	1.271
UT工程テスト項目数/KL	.702	.584	.392
IT工程テスト項目数/KL	.621	.420	.278
ST工程テスト項目数/KL	.888	.400	.372
UT工程テストバグ数/KL	.134	.578	1.160
IT工程テストバグ数/KL	.494	.583	1.014
ST工程テストバグ数/KL	.006	.000	2.357

5 考察及び関連研究

成熟度レベル1について総じて言えることは、基本的なプロジェクト管理に課題があり、小規模でない開発においてはソフトウェアの品質が悪化する傾向があるということである。小規模でない開発で、設計品質が悪く、上工程バグ抽出率が低い場合には、設計工程での品質状況に応じた十分なレビュー及びバグ抽出が実施されていない。個人の力量によって結果的に品質基準を満たすプロジェクトも存在するが、プロジェクトの結果は予測不能である成熟度レベル1の概念及びアセスメントの結果と合致する。まずは、基本的な開発管理技術の教育、適用が必要である。

成熟度レベル2においては、レベル1と比較すると約4倍の開発規模のプロジェクトまでは品質を制御することが可能である。したがって、基本的なマネジメントが機能していると考えられるが、やはり規模が大きくなるとソフトウェア品質が悪化する傾向が観察されている。ただし、上工程でレビューに十分な工数を投入し、テストでの検出バグを低く抑え、結果的に「達成」となるプロジェクトもある。品質確保を確実なものにするためには、品質の良し悪しにかかわらず上工程でレビューに一定量の工数をかける必要があるという我々の経験則から、レビューに十分な工数を投入することにより一定の品質を確保していると

考えられる。このような傾向を示すプロジェクトのプロセスを標準化して展開し、組織内のプロジェクトの品質底上げを図ることが求められる。ベストプラクティスの展開が不十分、改善成果の組織資産への組み込みが不十分であるといったアセスメントの所見とも整合する。

成熟度レベル3の組織においては開発プロセスの標準化がある程度進んでいる。分類木が構築されなかった理由は、標準化の浸透のために「達成」と「未達」において有意差のあるメトリクスはなかったためと考えられる。一方で、「未達」は、テスト工程全般にわたって「達成」よりもテスト項目数が少ない割にテスト工程バグ数は多く、終盤のST工程においてのバグ数の差が最も大きい。テスト工数については、「未達」は工程が進むにつれて一様に増加しており、徐々に積み残されたバグが終盤のテスト工程で検出され、工数を費やしても検出し切れずに最終的には工数不足となって十分に品質を確保できないまま出荷していると考えられる。すなわち、プロジェクト遂行中のきめ細やかな管理に課題がある場合に「未達」になる傾向がある。プロジェクトの実績をより高い頻度で把握し、実績に応じたタイムリーなマネジメントが重要である。CMMIの成熟度レベル4で求められる、定量的、統計的なプロジェクト管理に関する課題と言える。

CMMまたはCMMI成熟度レベル5の組織に焦点を当てた品質良否の要因分析はこれまでに報告されている。Honda^[11]はCMMI成熟度レベル5に該当する2つの組織について分析しており、同じ成熟度レベル5でもメトリクスの現れ方には大きな違いがあることを示している。また、Agrawalら^[12]

はCMM成熟度レベル5に該当する組織について分析した結果、品質に寄与する統計的有意な要因は開発規模のみであったとしている。成熟度レベル別に分析した研究ではRainerら^[7]の研究があるが、1節でも述べたようにこの研究では組織的プロセスの施策にかかわる要因を挙げており、個別のプロジェクト管理において有用な要因を挙げていない。また、Yanagidaら^[8]の研究では、成熟度レベル別に品質の良否にかかわる要因を分析しているが、メトリクスを複合的に組み合わせた分析という観点において十分でない。本研究はソフトウェア品質の良否にかかわる要因の複合的な分析を成熟度レベル別に行った点が特徴である。

6 おわりに

本論文では、522件のプロジェクトデータからCMMIの成熟度レベル別に分類木を構築した上で、有意差検定及び相関分析を組み合わせることでソフトウェア品質の良否に影響する要因を複合的に分析した。その結果、成熟度レベル1では開発規模、上工程バグ数/KL、及び上工程バグ摘出率が良否を分ける要因であると示された。成熟度レベル2では開発規模とテスト工程バグ数/KLが要因であると示され、このうち開発規模は成熟度レベル1の約4倍の境界値となった。成熟度レベル3では、有意差検定の結果からテスト工程バグ数/KLが主たる要因と示された。

今後の課題は、データ数の少ないセグメントの対処、より上位の成熟度レベルでの分析などである。

【参考文献】

- [1] CMMI Product Team, CMMI for Development, Version 1.3, Carnegie Mellon University, Software Engineering Institute, CMU/SEI-2010-TR-033 (2010).
- [2] Jones, C.: Software Quality in 2013: A Survey of the State of the Art, <http://namcookanalytics.com/software-quality-survey-state-art/> (参照2015-04-15).
- [3] Harter, D.E., Krishnan, M.S. and Slaughter, S.A.: Effects of Process Maturity on Quality, Cycle Time, and Effort in Software Product Development, *Management Science*, Vol. 46, Issue 4, pp.451-466 (2000).
- [4] Harter, D.E. and Slaughter, S.A.: Quality Improvement and Infrastructure Activity Costs in Software Development: A Longitudinal Analysis, *Management Science*, Vol.49, Issue 6, pp.784-800 (2003).
- [5] Gibson, D. L., Goldenson, D R. and Kost, K.: Performance Results of CMMI-Based Process Improvement, Software Engineering Institute, Carnegie Mellon University, CMU/SEI-2006-TR-004 (2006).
- [6] Process Maturity Profile, <http://cmmiinstitute.com/process-maturity-profile> (参照 2015-04-15).
- [7] Rainer, A. and Hall, T.: Key Success Factors for Implementing Software Process Improvement: A Maturity-based Analysis, *Journal of Systems and Software*, Vol. 62, No. 2, pp. 71 - 84 (2002).
- [8] Yanagida, R., Honda, N. and Komiyama, T.: What are the Key Factors of Quality Improvement regarding CMMI Maturity Levels?, *Proceedings of 6th World Congress on Software Quality*, London, England, pp.1-12 (2014).
- [9] 誉田直美: ソフトウェア品質会計, 日科技連出版社 (2010).
- [10] Key Practices of the Capability Maturity Model, Version 1.1, Carnegie Mellon University, Software Engineering Institute, CMU/SEI-93-TR-025 (1993).
- [11] Honda, N.: Success Factors to Achieve Excellent Quality – CMMI Level 5 Organizations Research Report, *Proceedings of 5th World Congress on Software Quality*, Shanghai, China, pp.1-8 (2011).
- [12] Agrawal, M. and Chari, K.: Software Effort, Quality, and Cycle Time: A Study of CMM Level 5 Projects, *IEEE Transactions on Software Engineering*, Vol.33, No.3, pp.145-156 (2007).