

プロセスメトリクスを用いたシステムテスト検出バグ数の予測

Estimation of the Number of Bugs Detected in System Testing by using

Software Process Metrics

日本電気株式会社 IT ソフトウェア事業本部

NEC IT Software Operations Unit.

○高橋 亮介¹⁾ 守屋 整¹⁾○Ryosuke Takahashi¹⁾ Hitoshi Moriya¹⁾**Abstract**

Development project consume many efforts by correcting bugs in the final stage of test. As a result, development project delayed in shipment time for delivery. In this paper the author introduce a model for estimation of the number of bugs in the final stage of development by using the metrics collected now, and the author considered the factor that bugs in the final stage of development extracted. As a result, it became clear that the number of bugs in the final stage of development was concerned with the design effort and review effort of an upper process.

1. はじめに

開発プロジェクトが出荷納期を遵守できない要因の1つに、開発終盤のテストでバグが発生し、その対応に多くの工数を費やしてしまうことが挙げられる。一般的にソフトウェア開発では、開発終盤になればなるほどバグに対応するための後戻り工数が増大すると言われている。そのため、ソフトウェア製品品質保証の立場からすれば、メトリクス等を活用することで、開発の出来るだけ早い段階に開発状況の異常を検知し、それを是正することによって、開発終盤のテストで発生するバグ数の削減を図りたい。

筆者の組織では開発工程全体に対してのバグ数を見積もるためのモデルは存在するが、開発終盤のテストで検出されるバグ数に焦点を絞った見積りモデルは存在しない。また、メトリクスに関する最近の文献(参考文献[1][2][3]参照)を見ても、開発終盤のテストで検出されるバグ数を見積もるための手法は見当たらない。

本稿では、多変量解析を使用し、システムテスト工程で検出されたバグ数とその他のメトリクスとの関連性を分析し、システムテスト工程でのバグ数を見積もるためのモデル式を構築する。その上で、このモデルに基づいて、開発終盤のバグ数を削減するための有効な手段を考察した結果を示す。

2. 開発プロセスについて

筆者の組織の開発プロセスは基本的に図1のV字モデルを採用している。基本設計工程からコーディング工程までを上工程とし、単体テスト工程からシステムテスト工程までをテスト工程として定義している。筆者の現場で課題となっている開発終盤のテストで検出されるバグとは、システムテスト工程(以下STと省略)で検出されるバグを指す。

また、筆者の組織における開発の特徴は以下の通りである。

- 特定の顧客向けではなく、汎用的なソフトウェアの開発が多い
- 新規製品の開発は少なく、既存製品に対する機能強化(バージョンアップ)を行う開発が多い

1) 日本電気株式会社 IT ソフトウェア事業本部

NEC IT Software Operations Unit

〒211-8666 神奈川県川崎市中原区下沼部 1753 Tel: 044-431-7692

1753, Shimonumabe, Nakahara-Ku, Kawasaki City, Kanagawa Japan Tel: +81-44-431-7692

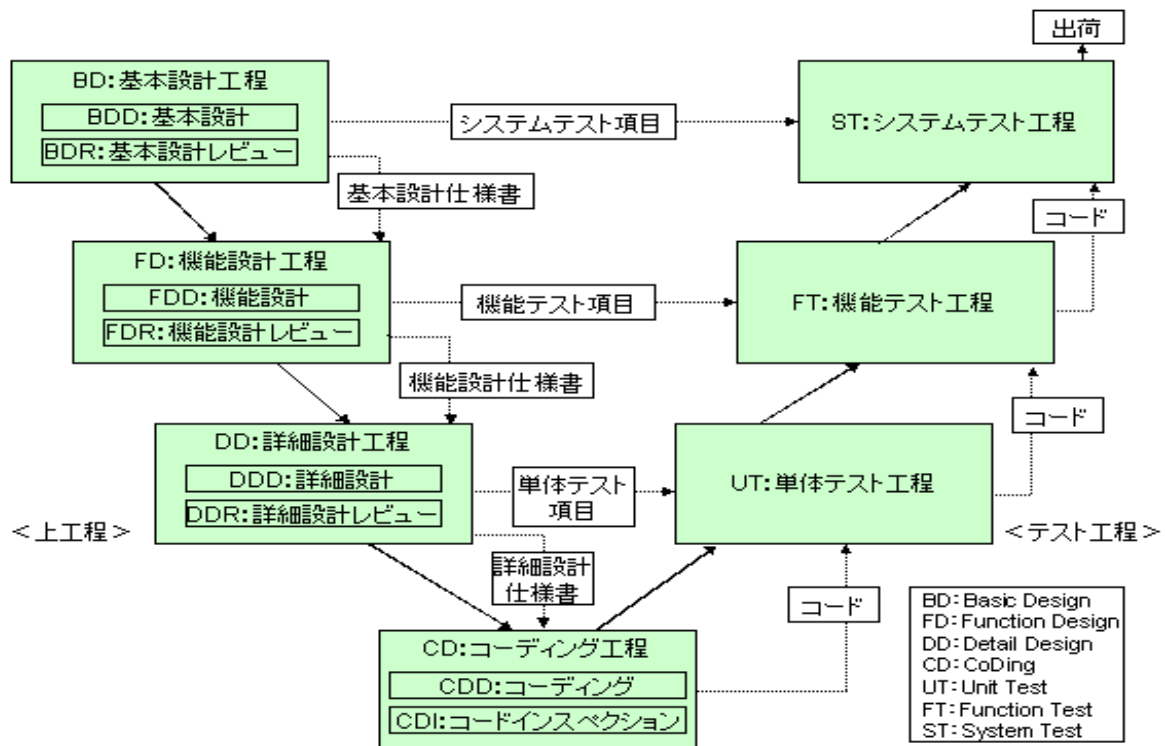


図1 開発プロセス

3. 分析対象データの概要

今回の分析に使用したデータの概要は以下の通りである。

- 開発期間：2010年度～2011年度に実施した開発
- データサンプル数：302個
- 開発規模が10LOC～数万LOCと非常に小規模な開発から大規模な開発のデータ
- データ分析に用いるメトリクス項目一覧

- | | |
|---------------------------|-----------------------------------|
| (1) STバグ数 | (11) 各工程の設計製造工数 (BD～CD) |
| (2) 開発規模 ※自製規模+流用規模 | (12) 各工程のレビュー工数 (BD～CD) † |
| (3) 自製規模 † | (13) 上工程レビュー比率 |
| (4) 上工程バグ数 | (14) 上工程レビュー比率 (CD工程除く) |
| (5) 全体工数 (ST工程除く) | (15) 各工程のレビュー比率 (BD～CD) |
| (6) テスト工数 (ST工程除く) | (16) 作り込み品質 ※上バグ/上レビュー工数 |
| (7) 上工程設計製造工数 | (17) 全体仕様書ページ数 |
| (8) 上工程レビュー工数 † | (18) 各工程の仕様書ページ数 (BD～CD) |
| (9) 上工程設計製造工数 (CD工程除く) | (19) 各工程の新規テスト項目数 (UT, FT, UT+FT) |
| (10) 上工程レビュー工数 (CD工程除く) † | |

このメトリクス項目一覧の内、(1)～(4)と(17)～(19)をプロダクトメトリクス、残りの項目をプロセスメトリクスと呼ぶことにする。また、(1)は目的変数、(2)～(19)は説明変数の候補となる項目とする。

† 筆者の組織では自分達で作り込んだ開発規模を自製規模と呼ぶ。

レビュー工数には開発者個人が実施するセルフレビューの工数は含まない。

4. 目的変数の設定

本稿の見積もりモデルで目的変数となるのは、開発終盤の ST で検出される ST バグ数とする。ただし、一般的にバグ数は自製規模と相関が強いと考えられるため、ST バグ数を自製規模で割った ST バグ密度と、自製規模で割らない ST バグ数の 2 つが目的変数の候補として考えられる。多変量解析を行う上で、どちらの項目を目的変数として設定することが妥当なのかを検討する。

一般的には開発する規模が大きければ大きいほどバグ数も多くなることが考えられる。しかし、それが全体バグ数と ST バグ数のみでは傾向が異なる可能性がある。そこで今回は、自製規模と全体バグ数との相関関係と自製規模と ST バグ数との相関関係を比較することによって、ST バグ数が全体バグ数と同じように自製規模と高い相関関係にあるか分析を行う。

自製規模と全体バグ数との相関関係を分析してみたところ、図 2 に示すように相関係数が 0.912 と非常に強い正の相関が見られた。

同じように自製規模と ST バグ数との相関関係を分析してみたところ、図 3 に示すように相関係数が 0.430 とそれほど強い相関は見られなかった。

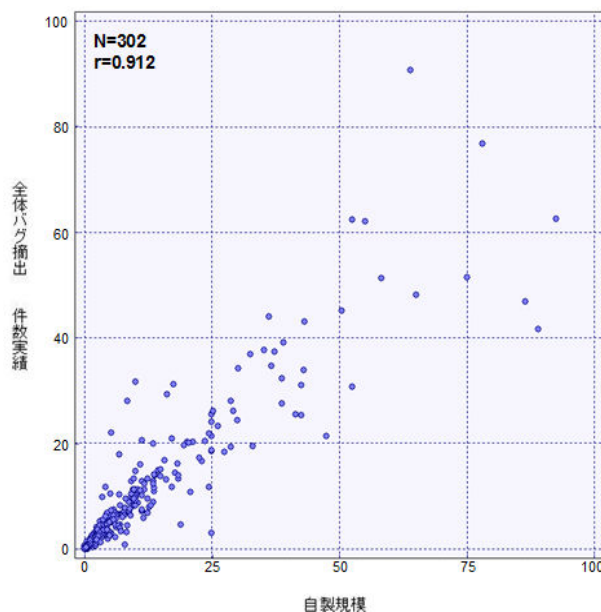


図 2 自製規模と全体バグ数の散布図

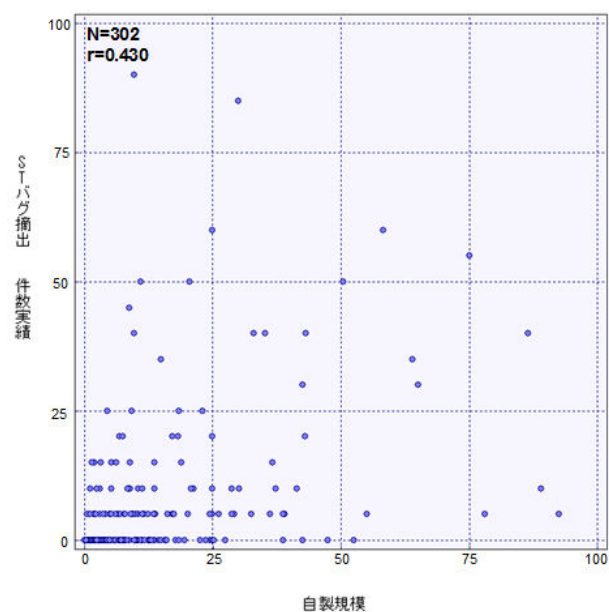


図 3 自製規模と ST バグ数の散布図

以上より、ST バグ数は全体バグ数と異なり自製規模の影響を強く受けないことが分かった。この結果より、ST バグ密度を目的変数として分析すると、分析結果でデータにばらつきがあった場合、ST バグ数の影響でばらついているのか、自製規模の影響でばらついているのかの判別が難しくなる。したがって、分析に用いるデータは ST バグ数を自製規模で割った ST バグ密度を使用せず、ST バグ数をそのまま使用する。

5. 相関係数による評価

5.1 ST バグ数と各変数との相関関係

ST バグ数と各変数との相関分析を実施した結果、特に相関が強かった 5 つの項目を表 1 に示す。表 1 を見ると一番強い相関係数は CD 設計製造工数の 0.520 であり、一般的に強い相関と言える 0.7 以上の相関係数がある項目は見当たらなかった。

†† 図 2～4, 6 で表示している目盛りの値は生データの値ではなく比率(ある数値で割った値を 100)で出している。

表 1 ST バグ数と各変数の相関係数

変数名	ST バグ摘出件数実績
全体工数実績 (S T 除く)	0.488
上工程設計製造工数実績	0.501
FD 設計製造工数実績	0.492
DD 設計製造工数実績	0.467
CD 設計製造工数実績	0.520

5.2 自製規模によるデータの絞込み

一般的に、大規模な開発ほど開発コストがかかり、重点的に開発管理を行なう対象となり得ることから、小規模な開発のデータは除外して、ある程度自製規模の大きいデータに絞って分析することにした。今回は自製規模を以下の3つのセグメントに分けてデータ分析を実施する。

- 自製規模1K LOC より大きい開発(データ数199個、全データ数に対する比率65.9%)
- 自製規模5K LOC より大きい開発(データ数78個、全データ数に対する比率25.8%)
- 自製規模10K LOC より大きい開発(データ数34個、全データ数に対する比率11.3%)

5.3 自製規模別による相関関係

1K LOC, 5K LOC, 10K LOC の3つにセグメントを分けてデータを相関分析してみた結果、ST バグ数と一番強い相関係数はそれぞれ以下の値となった。

- 1K LOC⇒0.477
- 5K LOC⇒0.511
- 10K LOC⇒0.795

1K LOC と 5K LOC より大きい開発データでは、全データと比較して一番強い相関係数の値に大きな差異が見られなかった。しかし、10K LOC より大きい開発データを用いると、表 2(特に相関が強かった5つの項目を表示)に示すとおり、上工程の設計製造工数で0.7以上と比較的強い相関がST バグ数と見られた。

表2 ST バグ数と各変数の相関係数(10K LOC↑)

変数名	ST バグ摘出件数実績
上工程設計製造工数実績	0.725
上工程設計製造工数実績(CD 除く)	0.675
FD 設計製造工数実績	0.666
DD 設計製造工数実績	0.795
CD 設計製造工数実績	0.759

上記の相関分析の結果より、自製規模を3つにセグメント分けしたデータの内、自製規模が10K LOC より大きい開発のデータを対象に、重回帰分析を使用して、目的変数であるST バグ数にどの変数が影響しているかを分析した上でモデル式を作成する。

6. 重回帰分析によるモデル式の作成

ST バグ数を目的変数として各変数間で重回帰分析を実施した。その結果、今回は上工程設計製造工数と上工程レビュー工数の2つを説明変数として選択する。

上工程設計製造工数を選択する理由は、ST バグ数に対して相関が高かった上工程設計製造工数と各工程の設計製造工数の相関係数はそれほど値に差異がなかったが(表2参照)、ST バグの作り込み工程を調べてみるとFD 29%、DD 24%、CD 47%と各工程にばらついており、説明変数として1つの工程に絞った項目ではなく、上工程全体の項目を用いる方が妥当だと判断したためである。

また、上工程レビュー工数を選択する理由は、重回帰分析で上工程設計製造工数を説明変数として選択した場合に、他の項目よりも調整済決定係数が高くなり、かつモデル式に適用した場合

に負の値を取り、レビューを実施すれば作り込まれているバグが排除されるという開発の実体と整合していたためである。以上より、この 2 つの変数を説明変数として選択するのは妥当だと判断する。

上工程設計製造工数と上工程レビュー工数の 2 つを説明変数とした場合、表 3 に示すように重相関係数で 0.747、決定係数で 0.558、調整済決定係数で 0.530 とある程度の精度の高い値が得られた。

表 3 ST バグ数と各変数の重回帰分析結果

	目的変数名	残差平方和	重相関係数	寄与率R ²	R* ²
	STバグ摘出件数実績	191.904	0.747	0.558	0.530
		R** ²	残差自由度	残差標準偏差	
		0.503	31	2.488	
vNo	説明変数名	残差平方和	変化量	分散比	偏回帰係数
0	定数項	196.486	4.581	0.7400	0.938
5	開発規模(実績)	189.099	-2.806	0.4451	-
6	自製規模(実績)	186.365	-5.539	0.8917	-
14	上工程バグ摘出件数実績	182.366	-9.538	1.5691	-
17	全体工数実績(ST除く)	185.194	-6.710	1.0870	-
18	テスト工数実績(ST除く)	185.194	-6.710	1.0870	-
20	上工程設計製造工数実績	434.196	242.292	39.1395	0.001
21	上工程レビュー工数実績	206.339	14.435	2.3318	-0.002

重回帰分析を実施した結果(表 3)から得られた値を使用して作成したモデル式は以下となる。

$$Y = 0.938 + (0.001 * X_1) - (0.002 * X_2)$$

Y: ST 予測バグ数

X₁: 上工程設計製造工数実績

X₂: 上工程レビュー工数実績

7. 作成したモデル式の評価

作成したモデル式を見ると、ST バグ数に対して上工程設計製造工数が正の(ST バグ数が増える)要素を与え、上工程レビュー工数が負の(ST バグ数を減らす)要素を与えていることが言える。これは設計製造を実施すればバグが埋め込まれ、レビューでそのバグを排除するという開発の実体と整合している事が言え、今回作成したモデルが妥当であることを示している。

続いて、今回作成したモデル式の精度について検証する。自製規模が 10K LOC より大きい開発のデータを対象に作成したモデル式によって得られた ST 予測バグ数と ST 実績バグ数との散布図と相関係数を図 4 に示す。

モデル式によって得られた ST 予測バグ数と ST 実績バグ数との相関係数は 0.745 と高い値が出た。以上の結果より、今回作成したモデル式は自製規模が 10K LOC より大きい開発に対しては、充分適用可能な精度だと考える。

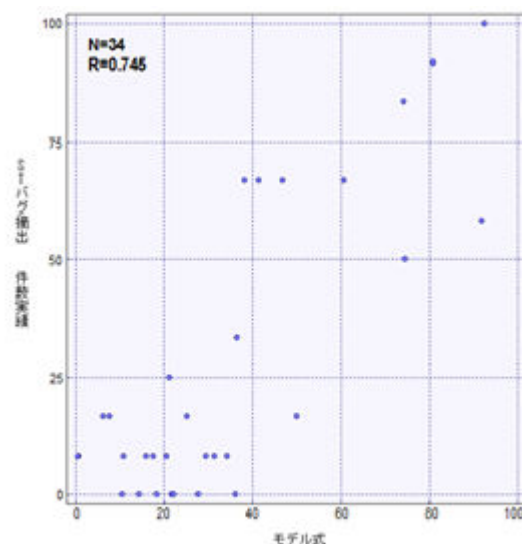


図 4 ST 予測バグ数と実績の散布図と相関係数

8. 考察

今回のモデル式では上工程設計製造工数と上工程レビュー工数の2つを説明変数として選択した。上工程設計製造工数と上工程レビュー工数の2つの説明変数が、一般的にバグ数と相関が強い自製規模より相関が強くなった要因について考察する。

同じ自製規模の開発でも、一般的に難易度の高い開発ほど上工程設計製造工数が多くなると考えられる。同じく全体バグ数についても開発難易度が高いほど多くなることが予想され、全体バグ数が多くなることに比例して ST 工程で検出されるバグ数も多くなることが考えられる(全体バグ数と ST バグ数の相関係数は 0.432)。このため、今回の回帰分析の結果では、自製規模より開発難易度によって値に変動が出てくる上工程設計製造工数の方が ST バグ数と強い相関が出たことが考えられる。これは ST バグ数が自製規模より開発難易度の方が強い影響を受けていることが 1 つの仮説として挙げられる。

一方、上工程設計製造工数単体より、上工程レビュー工数を説明変数として加えた方が、より ST バグ数との相関が強くなった理由は、レビューを実施すれば作り込まれているバグが排除されるという一般的な考えから説明できる。同じ設計製造工数をかけた成果物でも、レビュー工数をかけた方が、よりバグを排除できる可能性が高くなり、結果的に ST バグ数の削減につながる。このことから、バグを作り込む設計製造工数とバグを排除するレビュー工数は密接な関係があり、上工程設計製造工数単体より、上工程レビュー工数を説明変数として組み合わせることで、より ST バグ数に対する相関が強くなったと考えられる。

以上より、上工程設計製造工数と上工程レビュー工数の2つの変数を組み合わせることで、自製規模より ST バグ数に対する相関が強くなる結果が出たことが考えられる。しかし、自製規模が全く ST バグ数に影響がないとは考えていない。上工程設計製造工数と上工程レビュー工数は自製規模と強い相関(相関係数 0.81 程度)があるため、自製規模が大きくなるほど上工程設計製造工数と上工程レビュー工数も多くなる。したがって、図 5 に示す通り自製規模は直接的には ST バグ数と強い相関は見られなかったが、間接的には ST バグ数に影響を及ぼしていると考えられる。

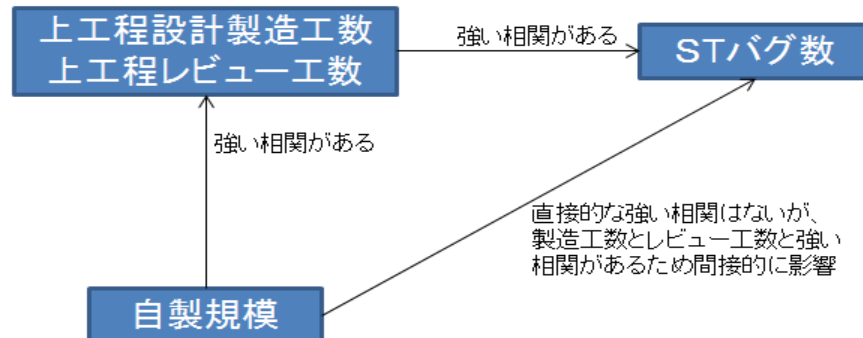


図 5 ST バグ数と各変数との関連図

次に今回のモデル式で説明変数として選択した上工程設計製造工数と上工程レビュー工数に対する標準偏回帰係数を表 4 に示す。標準偏回帰係数は上工程設計製造工数が 0.763、上工程レビュー工数が -0.186 だった。この結果から、上工程レビュー工数より上工程設計製造工数の方が ST バグ数に与える影響度が大きいことを統計的に示していると言える。

表 4 ST バグ数と各説明変数の標準偏回帰係数

目的変数名	残差平方和	重相関係数	寄与率 R^2	R^{*2}	残差自由度	残差標準偏差
ST バグ摘出件数実績	191.904	0.747	0.558	0.503	31	2.488
変数名	偏回帰係数	標準誤差	t 値	標準偏回帰	トレランス	
定数項	0.938	1.090	0.860			
上工程設計製造工数実績	0.001	0.000	6.256	0.763	0.959	
上工程レビュー工数実績	-0.002	0.001	-1.527	-0.186	0.959	

また、今回作成したモデル式で使用している説明変数はテスト工程での測定項目がなく、全て上工程完了時点には実績値が得られる項目となっている。これにより、開発の早い段階で開発状況の異常を検知するという当初の目的を満たしている点も評価できるポイントである。

さらに、モデル式から ST バグ数に対して設計製造工数が正の要素、レビュー工数が負の要素を与えていることから、上工程全体工数(設計製造工数+レビュー工数)に対する上工程レビュー工数の比率が、ST バグ数を削減するための要因になってくると考えられる。図 6 に上工程レビュー比率(上工程レビュー工数/上工程全体工数)と ST バグ数との散布図を示す。上工程レビュー比率のある値を 100 とした場合、100 を切っている開発で ST バグが多発(図 6 の丸囲み箇所)していることが分かる。この結果より、上工程レビュー比率は ST バグ数に影響してくるメトリクスの 1 つであることが言える。

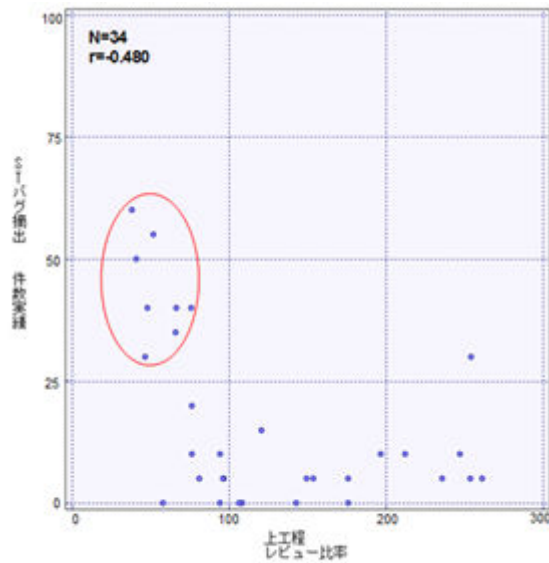


図 6 上工程レビュー比率と ST バグ数の散布図

以上より、上工程完了時に ST バグが多発しやすい状況(上工程レビュー比率がある値以下となる状況)を未然に防止するという観点で、上工程レビュー比率について基準値を設け、開発開始時に適用し、上工程の実績データが確定する上工程完了時に今回作成したモデル式を使用して、ST バグ数を見積もるという適用方法が適切だと考える。

また、上工程完了時に今回作成したモデル式を適用して ST バグが多発するような予測値になった場合は、以下の 2 点の対策を行うことを考えている。

- 開発側に検討不足やレビュー不足がないことを再度確認してもらう
- 品質保証部門が開発成果物(仕様書やレビュー記録票等)を直接確認する対象とする
(品質保証部門が全ての開発成果物を確認するのは工数面から見て不可能なため、ST バグが多発する予測値になったものを重点的に確認する)

以上の対策により、ST 工程でバグが多発するリスクを低減させていきたいと考えている。

9. おわりに

今回の研究により、自製規模が 10K LOC より大きい開発に対しては ST バグ数の予測モデルを構築することに成功した。また、このモデルから ST バグ数に影響してくる要因をプロセスメトリクスから特定することができた。この 2 点が、本研究の大きな成果である。

一方、課題としては以下の 2 点が挙げられる。

- 自製規模が 10K LOC より大きい開発しかモデルが作成できなかった
- 今回作成した予測モデルが実際の開発に対してまだ適用実績がない

今後、これらの課題を解決していくことで、ST バグ予測モデルの精度を向上させ、現在測定しているメトリクスから開発の異常な状態を早い段階に検知して、対策を打つことによって、開発終盤のバグ数の削減を図り、出荷納期遅延のリスクを低減させていきたいと考える。

10. 参考文献

- [1] 誉田直美, ソフトウェア品質会計: NEC の高品質ソフトウェア開発を支える品質保証技術, 日科技連, 2010
- [2] 倉下亮, 吉村博昭, 野中誠, 誉田直美, CK メトリクスの分布に基づくソフトウェア設計の質の定量的評価, ソフトウェア品質シンポジウム 2011 発表報文集, pp. 1-8
- [3] 吉村博昭, 森岳志, 倉下亮, 野中誠, プロセスおよび出荷後バグメトリクスによる残存バグ予測, ソフトウェア品質シンポジウム 2010 発表報文集, pp. 1-8