ソフトウェア欠陥予測アルゴリズム

- 欠陥混入メカニズムのモデリング手法を利用した欠陥予測方法の提案 -

Software defect prediction algorithm

- Proposed method of predicting defects using "Defect Injection mechanism" modeling -

2014年度 SQiP 研究会第7分科会

The 7rd Section Meeting Team of SQIP Study Group in 2014.

〇柏原 一雄 岡本 晃 1) 鈴木 裕一郎 2) 田村 光義 3)

OKashiwabara Kazuo Okamoto Akira Suzuki Yuichiro Tamura Mitsuyoshi

東久保 理江子⁴⁾ 保栖 真輝⁵⁾ 細川 宣啓⁶⁾ 永田 敦⁷⁾

Tokubo Rieko Hozumi Masaki Hosokawa Nobuhiro Nagata Atsushi

Abstract

Software defect information that harvested from several software development projects is one of the assets to share. Software defect information is indispensable information for the software defect prediction / prevention activity. In this research activity, we set the focus onto that to understand and to predict the defect injection, we have to establish the way to represent the "Defect Injection mechanism", furthermore, defect management process (modeling, store and share) have to be defined. In this paper, we propose "Defect Injection Mechanism" modeling and the method for establish the defect management database for the software defect prediction.

1 はじめに

ソフトウェア開発において、各組織では膨大な欠陥情報が蓄積され、ビッグデータ化している.同じメカニズムで繰り返し発生する欠陥の混入・流出を防止し、品質向上に繋げるためには、この蓄積した欠陥情報をいかに有効活用するかが鍵となる.

研究員が所属している組織でも、欠陥情報は蓄積されている.しかし、情報を蓄積するだけで、欠陥の混入・流出を防止するために、有効活用はできていない.我々は、欠陥の混入・流出を防止するために、蓄積した欠陥情報を利用する方法を研究の対象とした.

欠陥の混入・流出を防止するためには、まず対策の対象とする欠陥を特定しなければならない。また、過去の欠陥情報をもとに、同一条件下で混入し得る欠陥を予測する方法が必要となる。混入し得る欠陥を予測するためには、混入した欠陥とそれによって引き起こされる不具合の情報だけでなく、欠陥混入の要因・背景等も含めた欠陥混入に至るメカニズムを示す情報が必要となる。

本研究では、混入し得る欠陥を予測するためには、欠陥混入メカニズムが理解できるように表現され、その情報を資産として蓄積し、利用できるようにする必要があると考え、 手法を検討した.

本稿では、ソフトウェア開発において、混入し得る欠陥を予測することを目的に、欠陥

株式会社デンソークリエイト DENSO CREATE INC.

愛知県名古屋市中区栄 3-1-1 Tel: 052-238-0460 e-mail:kashiwabara@dcinc.co.jp

3-1-1, Sakae, Naka-ku, Nagoya-shi, Aichi, Japan

1) 農中情報システム(株) NOCHU INFORMATION SYSTEM CO.,LTD

2) (株)日立製作所 Hitachi, Ltd.

3) サイバートラスト(株)4) アンリツ(株)Cybertrust Japan Co.,Ltd.ANRITSU CORPORATION

5) 日本電子(株) JEOL Ltd

6) 日本アイ・ビー・エム(株) IBM Japan, Ltd

7) ソニー(株) SONY Co, Ltd

混入メカニズムのモデリング手法と欠陥データベースの構築・利用手法を提案する. 提案 手法をまとめ、欠陥予測アルゴリズムと呼ぶ.

提案手法の有効性を確認するために,実験では以下を確認した.

- ケ陥混入メカニズムが理解・納得できるように表現される
- ・欠陥混入メカニズムの情報を、発生し得る過失・欠陥の予測に利用できる本研究により、以下の貢献が期待できる.
 - ・テストやレビューの観点抽出のために利用することで、限られた工数で混入の可能性のある欠陥をピンポイントで検出できる。
 - ・欠陥混入メカニズムを理解し、知識として利用することで、プロジェクト開始前に、欠陥を混入させる要因を除去することが可能となり、混入する欠陥数を減らし品質向上とコスト削減に貢献できる。
 - ・欠陥情報を欠陥混入メカニズムという抽象化された知識にすることで、ソフトウェア業界全体で資産として共有できる。

本稿の以降の構成は次の通りである.

以降,2章で解決する課題を示し,3章で課題解決のアイデアを得た関連研究を示す.4章で提案手法の詳細を説明し,5章で提案手法の有効性を確認した実験の結果と考察を示す.6章で,まとめと今後の展望を示す.

本稿では,不具合,欠陥,誤りという言葉を表1のように定義し,使い分ける.

A = 1					
用語	意味				
不具合	ソフトウェアが期待結果を果たせていない状態.				
欠陥	仕様書やソースコードなどのプロダクトに含まれる不正確な記述箇所.				
誤り	不正確な結果を生み出した人間の行為.				

表 1. 用語と意味

2 背景

2.1 欠陥の予測方法

本研究における欠陥予測は、「プロジェクトマネージャや SQA が、プロジェクトの入力となる情報(要求、入力成果物の特徴、ベースソフトの特徴、プロジェクトメンバの特徴等)をもとに、過去の欠陥情報から、同一条件下で発生する可能性のある誤りと混入する可能性のある欠陥を特定すること」と定義する.

2.2 欠陥の予測に必要なこと

欠陥は人間の誤り(過失因子)によって生み出される.人間が誤りを犯したときには, その誤りを引き起こした要因(誘発因子)が存在する.この誘発因子の存在を検知するこ とで,誘発因子により発生する過失因子と,過失因子により混入する欠陥を予測すること が可能となる.

欠陥を予測するためには、不具合と欠陥を示すだけでなく、欠陥を混入させる過失因子、 人間の誤りを引き起こす誘発因子も示す必要があると考えた.この不具合・欠陥・過失因 子・誘発因子の関係を欠陥混入メカニズムと呼ぶ.

本研究では、欠陥混入メカニズムをどのように表現、蓄積するかを検討した.情報を表現しなければ、蓄積もできないため、特に重要となるのが、表現の方法である.

『失敗知識データベース構築の試み』^[1]でも、潜在的な失敗の予測などに失敗知識を活用するためには、最低限、失敗のメカニズムを示す制約・要因・特性・結果の要素は必要であると考えられている.

2.3 解決する問題

研究員が所属している組織では、不具合を解決まで管理し、教訓として共有するために、不具合の一覧(不具合 DB)が存在している. しかし、不具合 DB に蓄積している情報をもとに、欠陥の予測をすることはできていない.

不具合 DB が以下の状態になっていることが多く、欠陥混入メカニズムを表現する情報が理解できない、また欠陥混入予測のために利用しにくい.

ケ、ケンスをはいる。</l

表現が一般化されていないため,不具合の未体験者は,内容を理解できない.また,内容が理解できないため,「自分には関係ない」と判断してしまい,予測のために情報を利用しない.また,同一条件下で発生した欠陥を特定するときに必要となる欠陥混入の条件(原因・背景)が記録されていないことが多い.具体的には,欠陥を混入させた過失因子,人間の過失を引き起こす誘発因子が記録されていない.

・欠陥 DB が欠陥混入予測のために利用しにくい

不具合,欠陥,過失因子,誘発因子が区別し表現されていない.そのため,誘発因子が存在しているかを判断することや,過失因子,欠陥を抽出することが難しい.また,欠陥を予測する手順は確立されていない.欠陥情報は増加する一方であり,多数の欠陥情報から効率よく必要な情報を抽出できるデータベースとその利用手順が必要である.

この問題の主な原因は、欠陥情報を利用し欠陥を予測するという用途を考慮していないことである。想定していない用途で必要となる情報は、表現・蓄積されない。

不具合 DB に表現・蓄積されている情報は、主に以下を目的に、必要な情報のみ記載されていることが多く、不具合の対策が終わった後は、ほとんど利用されない.

- プロジェクトマネージャによる不具合対応の進捗監視
- •経営者,プロジェクトマネージャへの対策結果の報告

目的や利用方法が変われば、表現・蓄積する必要のある情報も変わる.過去の欠陥情報を利用し、欠陥の予測を可能とするためには、欠陥を予測する方法を明確にした上で、必要となる情報を明らかにする必要がある.

2.4 研究課題

本研究では、欠陥の混入予測を目的に、欠陥混入メカニズムモデリング手法と欠陥データベースの構築・利用手法を開発し、以下の2点の問題が解決されることを確認する.

- 欠陥混入メカニズムが理解できない
- ・不具合 DB が欠陥混入予測のために利用しにくい

3 欠陥混入メカニズムの表現手法

本章では、欠陥混入メカニズムの表現手法に関してアイデアを得た関連研究を示す.

3.1 欠陥モデリング

Project Fabre^[2]は、欠陥の情報はモデリングによって欠陥の可視化・抽象化が可能となり、共有・移転が容易になると提唱している。

また, 欠陥のモデリング手法が提案されている. 提案された欠陥のモデリング手法では, 過失(人間の思考や判断の誤り)に着目し, 欠陥情報を汎化し構造化し表現することを提案している.

欠陥混入メカニズムに必要な要素を漏れなく表現し,理解性を向上させるために,提案では,このモデリング手法をもとに,表記ルールを明確化し,記述ステップを定義した.

3.2 特性要因図と 5M1E

ソフトウェア開発において過失を誘発する要因は複数あり、欠陥混入メカニズムを表現 する場合、要因を系統的に(漏れなく、重複なく、矛盾なく)示す必要がある.

提案する欠陥混入メカニズムのモデリング手法では、特定の結果と要因系との関係を系統的に示す特性要因図 $^{[3]}$ の表現方法と要因の分類方法を参考にしている。要因の分類方法としては、5M1E (Man(人),Method(方法),Measurement(測定),Material(原料・材料),Machine(機械・設備),Environment(環境))を利用している。

4 提案

本章では欠陥予測アルゴリズムを説明する. 欠陥予測アルゴリズムの技術的要素は「欠陥混入メカニズムモデリングの手法」と「欠陥予測 DB の構造」,「過失・欠陥予測手順」の3つである.

4.1 欠陥混入メカニズムのモデリング手法

ソフトウェア欠陥混入メカニズム表記ルールとソフトウェア欠陥混入メカニズムの記述ステップを説明する.

4.1.1 欠陥混入メカニズム表記ルール

欠陥混入メカニズム表記ルールとして, ノードの種類とノードの繋ぎ方を示す.

(1) ノードの種類

ソフトウェア欠陥混入メカニズム表記法では、発生し得る「過失」と混入し得る「欠陥」を予測するために、表2のノードにより、「不具合」「欠陥」「過失因子」「誘発因子」の関係を示す。特に欠陥の作り込みに着目するため、「過失因子」は欠陥を見逃し流出させたレビューやテストの過失は除き、欠陥を混入させた過失のみ示すことにした。

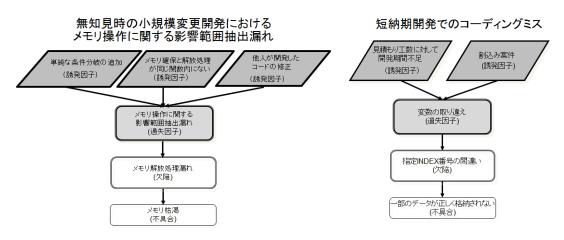
名称	ノード	解説			
不具合	不具合 (Incident)	ソフトウェアの振舞いが正しくないという現象のこと. 欠			
	(mcident)	陥によって引き起こされる.			
欠陥	7787	不具合をもたらした原因である成果物の記述のこと. 人間			
	欠陷 (Defect)	の誤りによって混入、流出する.			
過失因子		人間の思考や判断の誤りそのもののこと. 欠陥を混入させ			
	過失因子 (Negligence Factor)	た誤り. 欠陥を見逃し流出させたレビューやテストにおけ			
		る誤りではない.			
誘発因子		欠陥を混入させた誤りを誘発する"トリガー"となる要素			
	誘発因子	のこと. 誘発因子は, Man(人), Method(方法),			
	(Induction Trigger)	Measurement(測定), Material(原料・材料), Machine(機			
		械・設備), Environment(環境)に分類できる.			

表 2. ノードの種類

(2) ノードの繋ぎ方

ノードの繋ぎ方のルールを示す.欠陥混入メカニズムを人が理解しやすくすることを目的に、モデルにはメカニズムを構成する主要な要因のみを示し、モデルを複雑化させない表記法としている.図1に上記ルールで記載したモデルの例を示す.

- 「欠陥」と「不具合」は1対1で繋げる。
- •「欠陥」と「過失因子」は1対1で繋げる.
- •1つの「過失因子」と1つ以上の「誘発因子」を繋げる.



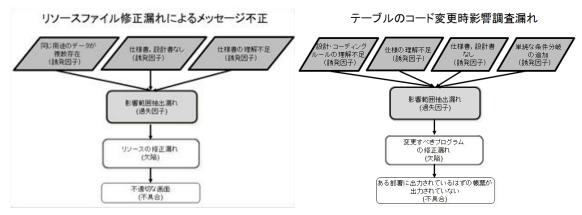


図1. 欠陥混入メカニズムをモデリングした例

4.1.2 欠陥混入メカニズムの記述ステップ

(1)モデルの記述ステップ

- 1. 欠陥情報をもとに、欠陥混入メカニズム表記ルールに従い、「不具合」「欠陥」「過 失因子」「誘発因子」の関係をモデリングする.
- 2. 誘発因子を, 5M1E の観点で系統的に (漏れなく, 重複なく, 矛盾なく) 示されて いるか確認する.
- 3. 欠陥混入メカニズムを共有するため、業務や製品固有の言葉を避け、一般化して表現する。
- 4. モデリングした欠陥混入メカニズムをレビューする.

(2) 欠陥混入メカニズムのレビュー観点

モデリングした欠陥混入メカニズムは、レビューによる妥当性確認を行う.レビューの 観点を表3に示す.

特性	観点	説明
理解容易性	欠陥の未体験者が	欠陥混入メカニズムの表現が一般化されていること
	理解できるか	を確認する.
納得性	欠陥混入メカニズ	過失因子に対して必要十分な誘発因子が挙げられて
	ムに納得できるか	いることを確認する. 欠陥混入メカニズムが納得でき
		ない場合には、客観的分析が不足し、誘発因子が不十
		分または不適切である可能性が高い.
利用可能性	誘発因子が欠陥予	開発開始時等に,誘発因子の該当/非該当の判断がで
	測 DB の検索条件と	きるかを確認する. 判断が難しい場合は, 抽象的な表
	して利用可能か	現になり過ぎている可能性がある. または, そもそも
		誘発因子としている要因が不適切な可能性もある.

表 3. レビュー観点

4.2 欠陥予測 DB の構造

欠陥予測 DB の構造を定義する. 欠陥予測 DB には, 第 4.1 章で述べた 4 つの要素「誘発因子」,「過失因子」,「欠陥」,「不具合」の関係を蓄積する. 図 2 に, 構築した欠陥予測 DB の例を示す.

欠陥予測 DB 構築のポイントは 2 つある. 1 つ目は, 既存の不具合 DB からの移行・拡張を容易にするために, 予測のために必要最低限のデータのみを蓄積することである. 2 つ目は, 検索を容易にするために,「誘発因子」をグルーピングすることである.

4.3 過失・欠陥の予測手順

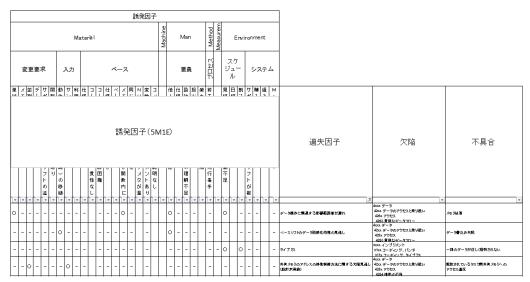
欠陥予測 DB を構築し、現在のプロジェクトの状況(現状と一致する誘発因子)を入力に、欠陥混入メカニズムを検索することで、起こり得る過失・欠陥を予測可能とする.

DBでは過去の誘発因子と過失および欠陥の関係性が蓄積されている. 同一の誘発因子に

よって発生した過失・欠陥が検索可能であり、同一状況下で発生し得る過失・欠陥が予測できる.

以下に現在の状況下で混入し得る欠陥を予測する手順を述べる.

- 1. プロジェクトの状態および環境などを調査する.
- 2. 調査結果をもとに、欠陥予測 DB のすべての「誘発因子」に対して該当/非該当を判断する.
- 3. 該当しない誘発因子が含まれている欠陥混入メカニズムを発生候補から外す.
- 4. 発生候補として残った欠陥混入メカニズムから,「過失因子」と「欠陥」を抽出する.
- 5. 抽出された「過失因子」と「欠陥」が、発生する可能性がある.



※○が付いている「誘発因子」により引き起こされる「過失因子」「欠陥」「不具合」を示す.

図 2. 欠陥予測 DB の例

5 実験

5.1 実験内容

研究員が所属する組織で実際に発生した欠陥情報を用いて、4章で提案した手法の有効性を以下の2つの実験で確認した.

【実験1】欠陥混入メカニズムが理解・納得できるように表現されることを確認する.

【実験2】「誘発因子」を「過失因子」の予測に利用できることを確認する.

【実験 1】では、4.1 節で述べた欠陥混入メカニズムの表記法を用いて記述したモデルを作成し、初めてモデルを見た人が事前説明なしに理解・納得を得られるか以下の選択肢を用いたアンケート方式で確認を行った.

○:全部理解できる・全部納得できる

△:一部の内容が理解できない・一部論理に納得できない

×:全部理解できない・全部納得できない

なお,2.1 節で定義している通り,欠陥予測の対象者は「プロジェクトマネージャや SQA」であるため,アンケートの対象はプロジェクトマネージャと設定した.また,「○」と回答しない場合はその理由をコメントしてもらった.

【実験 2】では、4.2節で述べた欠陥予測 DB を構築し、4.3節で提案した欠陥予測のアルゴリズムを用いて、プロジェクトで発生する「過失因子」を予測できるか確認した.本実験は、研究員が所属する異なる6組織で実施し、組織が変わっても提案手法により「過失因子」の予測が可能であることを確認した. 具体的には、各組織の終了プロジェクトで発生した「過失因子」を予測対象とし、プロジェクトマネージャがその当時のプロジェクトの状態に該当する「誘発因子」を選択し、それを入力として欠陥予測 DB から導き出した

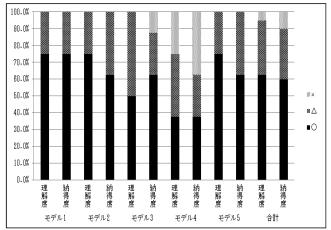
「過失因子」と実際にプロジェクトで発生した「過失因子」の一致率を確認した.

本来「過失因子」だけでなく「欠陥」の予測に利用可能であることを確認すべきであるが、今回は、「過失因子」と「欠陥」の関係を十分に網羅している欠陥予測 DB を構築できなかったため、「過失因子」から「欠陥」を予測する実験は見送った。また、実験に用いた欠陥予測 DB に登録しているデータ数が十分ではなかったため、予測精度の評価も見送った。

5.2 実験結果

【実験 1】の確認結果を図 3 に記載する. 5 つのモデルの平均値は理解度を「〇」と回答した割合は 62.5%, 納得度を「〇」と回答した割合は 60.0%であった. 2 つのモデルは理解度・納得度が 50%を下回った. 理解度・納得度を下げる理由として以下の 2 つが多く上がっていた.

- ・「過失因子」の不適切な表現・・・「過失因子」に記載している内容が理解できない
- ・「誘発因子」の過不足・・・「誘発因子」と「過失因子」の関係性が納得できない



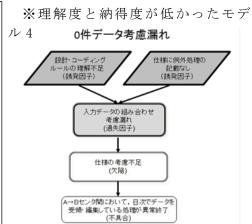


図 3. 【実験 1】の結果

【実験 2】の結果を表 4 に記載する. 欠陥予測 DB に登録したモデル 44 件を使って, 開発が終了した 10 プロジェクトの不具合 DB に対して実験をした結果である. 発生した欠陥のうち, 平均約 55%, 最大 70%の確率で「過失因子」の候補を特定できることを確認した. また, 一致した「過失因子」のうち件数が多かったものは「影響範囲抽出漏れ」,「コーディング漏れ」,「タイプミス」であった.

プロジェクト	発生した欠陥数	予測で一致した過失因子数	一致率
Α	28	11	39.3%
В	24	13	54.2%
С	52	36	69.2%
D	24	15	62.5%
E	43	30	69.8%
F	6	2	33.3%
G	6	4	66.7%
Н	19	5	26.3%
I	16	6	37.5%
J	19	10	52.6%
合計	237	132	55.7%

表 4. 【実験 2】の結果

5.3 考察

【実験 1】の結果から、4.1 節で述べた欠陥混入メカニズムの表記法を用いることで、 平均で6割以上の人の理解・納得を得られるようになったことを確認した.本手法は、「欠 陥混入メカニズムが理解できない」という問題を解決可能な手法であると判断できる. 従来は口頭または文章によって伝えられていた欠陥の情報を,本手法では,混入の経緯を含め図式化して伝えるようした.これにより,初見でも欠陥混入メカニズムを理解・納得することが可能となったと考えられる.

理解を得られなかったモデルについては、各因子に記載している表記方法が理解されていないこと,「欠陥」と「過失因子」の区別が明確でないことが理由として考えられる。納得を得られないモデルについては,「誘発因子」と「過失因子」の関係性に対する納得が得られていないことに起因している。「誘発因子」と「過失因子」の関係性について実験のために用意した個々のモデルに対しての検証が不十分であると考えられ、モデル自体が欠陥混入メカニズムを適切に表現できていない可能性がある。

欠陥混入メカニズムのモデリング手法を用いる際に,万人がいかなる欠陥を表現する場合でも同程度の質でモデリング可能とするため,表記ルールとモデルの検証手法を改善・確立することが,今後の課題である.これらを改善することにより,理解度・納得度の向上が見込まれる.

【実験 2】の結果から、異なる複数の組織で、「誘発因子」から「過失因子」を予測する行為が可能になったことを確認した。本手法は、「不具合 DB が欠陥混入予測のために利用しにくい」という問題を解決可能な手法であると判断できる。また、完全な予測ができることは確認できなかったが、プロジェクトに潜む「誘発因子」から発生するであろう「過失因子」の候補を示すことが可能となり、プロジェクトメンバ間での共通のリスク認識、設計・開発時の注意喚起およびレビュー、テストの観点抽出として活用することは可能であると考えられる。

欠陥の予測可能性,予測の精度,品質向上における有効性を確認することが,今後の課題である.

6 おわりに

我々は、過失は同じメカニズムで繰り返し発生すると考え、欠陥混入メカニズムを表現・蓄積し、その情報をもとにして過失・欠陥を予測する「欠陥予測アルゴリズム」を提案した。実験により、本手法で欠陥混入メカニズムが理解・納得されるように表現できることを確認した。また、欠陥予測アルゴリズムが、プロジェクトで発生する「過失因子」の候補を特定できることを確認した。本研究により、過去の欠陥情報をもとに、発生し得る過失を予測する手法を新たに確立できた。

欠陥を知識として伝承し、理解しやすくなったことは、有効な改善活動へ繋がる大きな 進歩であるといえる。また、実験は異なる複数の企業で実施し、本手法はソフトウェア業 界全体に適用可能と確認できた。将来、ソフトウェア業界全体で欠陥混入メカニズムの情報を蓄積し、各組織で欠陥予測に利用することも可能と言える。

今後は、欠陥の予測可能性の確認、欠陥予測 DB の拡充、今後開発を開始するプロジェクトへの適用実験、過失因子・欠陥と関連する混入・流出防止策(分析技法、設計技法、検証技法等)の整理等を実施し、より有益な手法となるよう改善していきたい.

参考文献

- [1]畑村洋太郎・中尾政之・飯野謙次「失敗知識データベース構築の試み」,2003年
- [2]細川宣啓,野中誠,西康晴,原佑貴子,嬉野綾「過失に着目した欠陥のモデリング ~バグ分析はなぜうまく行かないのか?~」,JaSST2013 Tokyo
 - [3]石川馨「品質管理入門」日科技連出版社,1956年
- [4]野中誠,小池利和,小室睦「データ指向のソフトウェア品質マネジメント」日科技連出版社,2012年