研究コース５（Team KuKuRu）

ソフトウェア欠陥多属性表現（MARS）モデル

－ ソフトウェア故障モードの抽出方法と利用について －

“Multi-Attribute Representation of Software defect”（MARS）Model

研究員（五十音順）：久野 倫義（三菱電機株式会社）
仁藤 千博（矢崎総業株式会社）
牟田 香奈（日本 ATM ヒューマン・ソリューション株式会社）
主査　：細川 宣啓（日本アイ・ビー・エム株式会社）
副主査：永田 敦　（株式会社日新システムズ）

研究概要

ハードウェア製品の故障を未然防止するため，多くの組織や企業で故障モード影響解析手法（FMEA）が用いられている． JIS C5750 規格ではソフトウェアも FMEA の対象としているが，物理的に存在しないソフトウェアに対し，ハードウェアと同じ方法では故障モードを列挙できない．本稿は従来の FMEA と欠陥モデルの対比から，物理法則にあたるものを誘発因子と過失因子の一連の連鎖と仮定し，ソフトウェア故障モードを定義する．また，誘発因子・過失因子から経年の有無などの排他的属性，および自然言語で自由に表現された欠陥の特徴を属性として選び出し，これらの多様な属性を用いて欠陥を表現した“ソフトウェア欠陥多属性表現（MARS）モデル”を提案する．このモデルを用いて欠陥モデル，ソフトウェア故障モード及び MARS モデルの関係性を示し，誘発される過失の存在を明らかにする．

Abstract

Many companies use Failure Mode and Effect Analysis (FMEA) to prevent hardware failure. In JIS C 5750, software is also targeted by FMEA, but it is not used at the site of software development. Because, they cannot enumerate failure modes of software that do not physically exist in the same way as hardware.

In this paper, from the comparison between the FMEA and the defect model, the Induction Trigger/ Negligence factor is listed as the physical law such as aging degradation, and the failure mode extraction method of software is defined. Also, from the Induction Trigger/ Negligence factor, exclusive attributes such as existence of aging and characteristics of defects freely expressed in natural language are extracted, and defects are expressed using these attributes "Multi-Attribute Representation of Software defect (MARS) model" is proposed. Furthermore, by showing the relationship between the defect model and the MARS model, the diversity of defects is clarified.

1. はじめに

1.1 研究の背景

ソフトウェア開発者が欠陥を欠陥と識別することができなければ，除去することもできない．しかし，一般に組織標準として設定されているチェックリストが開発者に外因欠陥（環境変化や集団認知バイアス等の影響で混入した欠陥）[1] を気付かせる効果については疑問がある．また，インシデントの再発防止として，バグ票などに欠陥の混入や未検出

の原因を教訓的に記録する試みも組織毎に行われているが，収集した欠陥情報をそのまま利用できるプロジェクト（例えば保守開発や同一製品の派生開発）以外での再利用はほとんど進んでいないのが実情である[2]．いずれの方法もこれらを利用する側に相応の知識と経験を要求する，属人性の高さに一因があると考えられる．

ODC 分析[3]は分析用語や分類属性を統一することで，企業や組織の枠組みを超えた議論が可能となる．しかし，属性の選択肢から属人性を排除し排他的選択を行う方法について，実務適用レベルでは解決しきれていない．

1.2 問題解決のアプローチ

ハードウェア製品を製造する多くの組織では故障の未然防止のために，その製品で考えられる故障モード（Failure Mode：FM）を列挙し，対策を行っている．故障モードは組織や企業の枠組みを超えて知識として共有・蓄積され，それぞれの企業や組織で製品への影響解析に利用されていることから，属人性の排除と適度な抽象化がなされていると考えられる．JIS C5750 規格[4]では故障モードの抽出手順を以下の様に説明している．

ⅰ）アイテム（部品，構成品，デバイス，装置，機能ユニット，機器，サブシステム，システムなどの総称又はいずれか）を特定する．

ⅱ）アイテムの故障の様子（故障状態の形式による分類．例えば，断線，短絡，折損，摩耗，特性の劣化など）を列挙する．例えば，構造が異なる機器でも電気回路を内蔵している限り「断線」が起こる可能性がある．

故障を引き起こす故障モードは類型的に分類され，故障モードから製品故障にいたるメカニズムを手繰っていくことで故障の質的予想を系統的に統一的に行うことが可能になる[5]．

ソフトウェアにおいても故障モード影響解析（Failure Mode and Effect Analysis：FMEA）適用の試みがなされている．それらの研究における故障モードの抽出は，従来のハードウェアにおける考え方をベースにしている[6][7]．本研究では，ソフトウェア固有の故障モードとは何かを欠陥エンジニアリングの観点から考え，ソフトウェア故障モードを示すモデルの提案を試みた．

1.3 本稿が答えるべき課題

ハードウェア故障は物理法則に依存することから，物理的な事象を表現する言葉で定義され，影響解析において有効な故障モードは有限である．一方，実体のないソフトウェアの不具合には依存する物理法則がないため，故障モードが単に“発生しうる全ての不具合”を指すのであれば，無限に列挙できてしまう．或いは有効な故障モードに限定するために適用範囲を保守開発や派生開発に絞れば，汎用性が損なわれる．

以上から，ソフトウェア固有の故障モードの存在と意義に関する議論を行い,その結果を踏まえて本稿は以下の課題に答える．

- RQ1：著者らの考える汎用的なソフトウェア故障モードを利用することで，属人性の高い手法では検出が難しい外因の欠陥検出が可能か．
- RQ2：ソフトウェア開発時にソフトウェア故障モードを使うと，なぜ欠陥を未然防止できるのか．

2. 提案

2.1 先行研究

欠陥モデルはソフトウェアの欠陥発生メカニズムに着目した，他に類のない欠陥表現モデルである[8]．著者らは，従来の故障モードが組織を超えてハードウェア製品不具合の予測に利用できる理由を，利用者が故障発生メカニズム（物理法則）を理解していることにあると考えた．そこで，FMEA と欠陥モデルの構造を対比させることで（表 2-1-1），ソフトウェアの欠陥の様子は誘発因子と過失因子の一連の連鎖に従い，影響解析において有効な故障モードの抽出が可能なソフトウェア欠陥は外因欠陥であると仮定した．

表 2-1-1. 故障モード影響解析手法と欠陥モデルとの対比

故障モード影響解析手法	欠陥モデル（付録 1.参照）
時間や重力など（暗黙）	誘発因子
アイテムの故障は物理法則に従う（暗黙）	過失因子（外因）
アイテム	欠陥（対象・位置）
故障モード	欠陥（種類）
頻度・検出可能性	増幅因子
故障の影響	インシデント

以上から，ソフトウェアの故障モードを論ずる場合，ハードウェアにおける暗黙の物理法則に該当する支配法則として「誘発因子」「過失因子（外因）」を明示する必要があり，これを本稿におけるソフトウェア故障モード（SW 故障モード）と定義した．

SW 故障モード ＝ 誘発因子 ＋ 過失因子（外因） ＋ 欠陥/種類 … 式①

一方，欠陥モデルは作成者の属人性（注目している観点，経験やドメイン知識）に影響されやすい．このため，欠陥モデル表現による SW 故障モードを他の組織へ移転することは一定の困難さがあると考えた．

2.2 提案内容：ソフトウェア欠陥多属性表現（MARS）モデル

本稿は欠陥表現の別のアプローチとして，客観的に観測可能な欠陥情報の特徴のみを具象化して属性とし，それらの属性のみで表現した“ソフトウェア欠陥多属性表現モデル”（Multi-Attribute Representation of Software defect Model：MARS モデル）を提案する．本モデルは，前節で定義した SW 故障モードの抽出と利用，および欠陥モデリングにおける属人性の排除を狙いとする．

本モデルが扱う属性について，以下に記述する．前述 1.2 項の従来の故障モードの類型的分類とは，即ち背景にある物理法則を分類したものである．著者らは欠陥の SW 故障モードでも類型的分類を行うため，ソフトウェア欠陥の支配法則である「誘発因子」「過失因子（外因）」から属人性の影響を受けにくい特徴を抽出し，排他的選択が可能な属性として定義した．

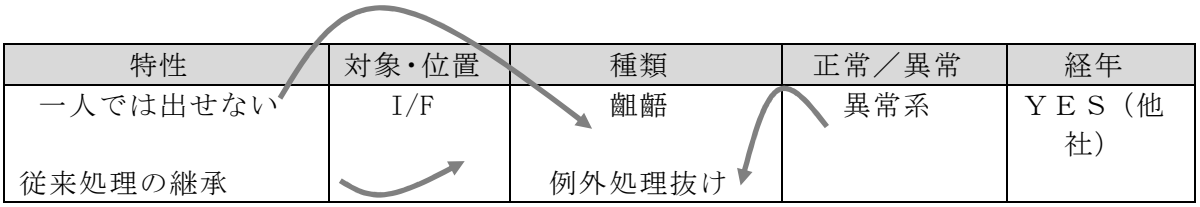
本稿では欠陥の属性例として「特性」「位置」「種類」「正常/異常」「経年」を列挙する（表 2-2-1 参照）．「正常/異常」「経年」は従来の故障モードの考え方から着想し，著者らが収集した欠陥情報の「誘発因子」に見出すことができることから，MARS モデルで扱う

属性として提案するものである.

表 2-2-1. MARS モデルで扱う属性の例–

属性	詳細
特性	欠陥を誘発した特徴的な性質（例：単独では起こらない）
対象・位置	欠陥の含まれるアイテム（例：I/F）
種類	欠陥の種類（例：齟齬）
正常/異常	欠陥の排他的性質（正常/異常）
経年	初期開発からの時間経過の有無（Yes/No）（例：保守性の低下）

MARS モデルは欠陥情報より属性毎に情報を入力しつつ，思考の流れを矢印にて表現する（図 2-2-1 参照）.

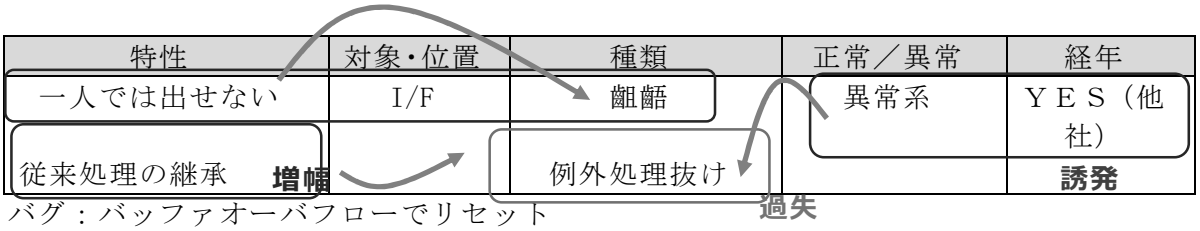


バグ：バッファオーバフローでリセット
結果：無現ループ

図 2-2-1. MARS モデルの作成例

2.3 欠陥モデルとの関連

MARS モデルには欠陥モデルの各因子が含まれている．例えば，図 2-3-1 の各属性で抽出された要素が誘発因子に対応し，矢印で示される思考の流れが過失因子に対応する.



バグ：バッファオーバフローでリセット
結果：無現ループ

図 2-3-1. 欠陥モデルの各因子との関係

3. 実験

3.1 実験の目的

MARS モデルで表現した既知の欠陥を利用すれば，誰でも別のソフトウェア上に同じ或いは類似した欠陥を発見することができることを実証する.
また，欠陥モデルを用いて同様の実験を行い，MARS モデルによる欠陥情報の伝達性に有意差がみられるかを検証する.

3.2 実験手順

8 パターン（付録 6.参照）の MARS モデルを被験者に提供し，過去に経験した欠陥を未然防止できたかを評価する．評価観点および実施手順は以下の通り．

(1) 経験差異
　被験者としてベテラン６人/中堅７人/若手３人を選定し，経験差異による結果の違いを評価する．また，欠陥モデルでも同様の評価を行う．
(2) 欠陥モデル参照時間
　被験者としてベテランを選定，欠陥モデル参照時間による結果の違いを評価する．
(3) 評価対象モデルの利用実感
　評点{1:かなり効果的，2:効果的，3:効果的でない，4:全く効果的でない}，および自由記述形式にて収集する．

【グループＡ】① MARS モデルを 15 分間参照した場合の件数と内容を列挙する．
　　② 欠陥モデルを 15 分間参照した場合の件数と内容を列挙する．
【グループＢ】③ 欠陥モデルを 30 分間参照した場合の件数と内容を列挙する．

3.3 実験結果

実験の結果を下表に示す．

表 3-3-1. 実験結果

	A（各モデルを 15 分づつ参照）			B（30 分参照）
	若手	中堅	ベテラン	ベテラン
MARS モデル利用	0.7 件/人	3.6 件/人	4.8 件/人	－
欠陥モデル利用	1.0 件/人	2.4 件/人	4.5 件/人	15.5 件/人
評価対象モデルの利用実感	3.0 (MARS モデル)	2.3 (MARS モデル)	3.3 (MARS モデル)	2.3 (欠陥モデル)
t 検定 (有意水準 0.05)	0.18 有意な差なし	0.02 有意な差あり	0.36 有意な差なし	－

3.4 結果に対する評価

前節の表 3-3-1 より，中堅では MARS モデルの方が想起される欠陥数が欠陥モデルのものと比べ有意な差があるという期待通りの結果が得られた．一方，若手・ベテランでは MARS モデルを用いても想起される欠陥数にほとんど差はなかった．ベテランにおいては，欠陥モデルを見れば未然防止できる欠陥を自己の経験の中から想起できるため，あえて MARS モデルのようなコンパクトな表現である必要がなかったと想定される.また若手では，経験している欠陥が少なく抽象化されたモデルでは,想起できる欠陥がなかったと言える．中堅メンバからは，「MARS モデルでは欠陥を可視化して分析できるため未然防止策立案の効率化に効果ありと感じました.」という意見があり，MARS モデルの有効性を示す結果を得られた．

次章では抽出した SW 故障モードの正当性の確認方法について考察する．

4. 考察

4.1 SW 故障モードの表現の深化

今回の実験で利用した MARS モデルでは，欠陥を複数軸で表現し，人間の思考を矢印でつなぎ，それら全体を FM であるとした．前章の実験でのアンケートにて，矢印の向きや本数が複数あり，その理解が困難であるという意見が数多く見られた．そこで，MARS モデルの表現方法をより理解容易化し，且つ，抽出した SW 故障モードの正当性を追跡できる「Z-Chart」を考案した．

Z-Chart は，MARS モデルで表現した内容を表形式に纏め，MARS モデルでは矢印で表現していた「思考（過失)」を列で表現するものである．1 つの欠陥をトリガーから開始し，誘発因子，思考（過失)，現象(欠陥)という順序で記載する．この際，現象が次の FM トリガーとなることがあるため，複数の FM の連鎖で 1 つの欠陥が発生することを表現している（表 4-1-1 の矢印）．

表 4-1-1. Z-Chart の様式

	トリガー	誘発因子	思考（過失）	現象（欠陥）
FM1-1	トリガー１	誘発因子１	過失１	現象１
FM1-2	トリガー２（現象１）	誘発因子２	過失２	欠陥

表 4-1-2,4-1-3 に欠陥を Z-Chart で表現した例を示す．各行は個々に 2.1 式①で示した SW 故障モードを表している．

表 4-1-2. Z-Chart による FM 表現例①

	トリガー	誘発因子	思考（過失）	現象（欠陥）
FM1-1	市場の要求変化	受信対向部の変更	既存ロジックの修正を行う（しかない）	I/F 修正
FM1-2	I/F 修正	相手が存在する	双方，I/F 仕様に対する認識に相違ないため，齟齬はない（はず）	齟齬（欠陥混入）
FM1-3	齟齬	本来必要な確認時間が確保されない	I/F 仕様の確認は十分実施（齟齬は未確認）	欠陥流出

表 4-1-3. Z-Chart による FM 表現例②

	トリガー	誘発因子	思考（過失）	現象（欠陥）
FM2-1	システム試験による欠陥報告	既存ボタンへの改修が必要	既存設定は触りたくない．（触る必要はない）	1 つのボタンに 2 つのイベントを設定
FM2-2	1 つのボタンに 2 つのイベントを設定	システム試験の不具合の改修を単体試験で確認しＯＫ．	指摘された欠陥の修正が完了した（だから大丈夫）	2 つの設定が競合するタイミングがある
FM2-3	イベントの競合	システム試験へのリリースを急がされている	既存設定を残しているが動作している（これ以上試験しなくても大丈夫）	欠陥流出

Z-Chart を用いることで，SW 故障モードの作成者は，SW 故障モードに論理的な矛盾のな

いことを確認できるようになると考える.

例えば，下表に示すように，開発者自身の無知（内因）が現象を引き起こすケースでは，誘発因子と現象の間をつなぐ思考は記述されず，１本の矢印として描かれない.

表 4-1-4. Z-Chart による FM 表現（内因欠陥の場合）

	トリガー	誘発因子	思考（過失）	現象（欠陥）
FM1-2	I/F 修正	相手が存在する 無知 教育環境	仕様書通り修正を行っている（はず）	?（なんらかの欠陥混入）

つまり，内因欠陥については Z-Chart での表現ができず，このようなケースでは本稿が提案する MARS モデルのみならず，どのような方法でも具体的な欠陥の予測は困難であることを示唆している.（なんらか欠陥の発生率が高いことは間違いないだろう.）

このことは，2.1 式①の過失因子は外因であるという著者らの仮説の正当性をよく表していると考える.

4.2　ソフトウェア故障モードの副次的効果

今回の実験で欠陥モデルのみを利用したグループＢの被験者の中に，20 件の欠陥を想起した被験者がいた．暗黙知の形式知化は従前からの課題とされているが，8 パターンの欠陥モデルから 2.5 倍（20 件）もの欠陥を想起できる被験者がいたことは特筆に値する．対応の遅れは組織にとって致命的な機会損失となる可能性がある.

5.　まとめ

5.1　本稿の課題への回答

RQ1: 著者らの考えるソフトウェア故障モードを利用することで，属人性の高い手法では検出が難しい外因の欠陥検出が可能となるか.

- 今回の実験によって，経験の浅い中堅･若手も MARS モデルにより未然防止できる欠陥が増えることが判った．アンケートでも実感として有効であると回答している.

つまり，欠陥に関するベテランの知見を同組織内の中堅・若手や知見のない他組織へ移転する方法として，MARS モデルは属人性の高い欠陥モデルよりも適しているといえよう.

RQ2: ソフトウェア開発時にソフトウェア故障モードを使うと，なぜ欠陥を未然防止できるのか.

- SW 故障モードに着目して作成された MARS モデルや Z-Chart には人の過ちを誘発するメカニズムが記録されている．欠陥の発生メカニズムを移転・共有することで，従来の故障モードを利用した不具合の未然防止と同様・同程度の効果が得られる.

今回の実験でも，被験者らは提示された８パターンの MARS モデルから欠陥を複数想起することができた．その理由の一つに，SW 故障モードに含まれる誘発因子と過失因子の連なりから，経験年数や保有スキルによらず欠陥を追体験することにより，SW 故障モードを意識し，欠陥の混入を予測した回避行動を取るようになるため，欠陥を未然予防することができる.

今後の研究で，さらにそのメカニズムを明らかにしたい.

5.2　本研究の意義と今後の展望

企業間での欠陥情報の共有は，欠陥を語る上で欠かせないと考えられているインシデントの公開が障壁となっており，欠陥モデルの研究等では抽象度を上げることで解決を図る方法などが提唱されている．本研究によって，実は SW 故障モード(誘発因子＋過失因子＋欠陥）のみを共有することで容易に移転されることが示された．

著者らは，「ソフトウェア欠陥標本」というものがあれば，これと同じ特徴を持つソフトウェア部品を探索することで，ソフトウェア製品から特定の欠陥を見逃すことなく効率的に識別できると考えている．著者らの考えるソフトウェア欠陥標本は，以下の特徴を持つものである．

① 属人性が低い（欠陥標本の作成手順が示され，誰でも利用できる）
② 抽象度は多層構造（欠陥標本が組織の枠組みを超えて共有・蓄積され，利用できる）

これを踏まえ，MARS モデルの作成手順を以下に示す．

① 欠陥情報保管のため，欠陥モデルを作成する
② 欠陥モデルから MARS モデルを作成する
③ Z-Chart を作成し，SW 故障モード層毎に誘発因子/思考/現象の正当性を評価する

MARS モデル自身に持たせる情報量は少ないが，必要に応じ，元の欠陥モデルと紐づけることもできることから，著者らの考えるソフトウェア欠陥標本の特徴を有するものとなった．また，Z-Chart は欠陥モデリングにおける属人性回避の有効な手段の一つとなるであろう．

今後，外因欠陥の識別が容易となり，蓄積された SW 故障モードを用いた欠陥の未然防止が行われるようになれば，人はもう，誘発因子に惑わされることはなくなるのである．

6.　参考文献

[1] 2016 年度ソフトウェア品質管理研究会 第 7 分科会，“数理科学アプローチを用いた客観的欠陥弁別法～ 外因欠陥の弁別方法とその効果・意義 ～” 2016.
[2] 2015 年度ソフトウェア品質管理研究会 第 7 分科会，“ソフトウェア開発における欠陥情報移転法の提案” 2015.
[3] IBM:Orthogonal Defect Classification v 5.2 for Software Design and Code, IBM, 2013.
[4] 日本工業規格 JIS C5750-4-3：2011，ディペンダビリティ マネジメント-第 4-3 部：システム信頼性のための解析技法-故障モード・影響解析（FMEA）の手順，2011.
[5] 久米均，設計開発の品質マネジメント，日科技連出版社，1999，pp. 141-143.
[6] 山科隆伸，森崎修司，“大規模ソフトウェアの保守開発を対象とした故障モード影響解析(FMEA) 適用の試み”，技報 UNISYS TECHNOLOGY REVIEW，28(4)，pp. 107-121 2009 年 2 月.
[7] 余宮尚志，“観点を用いたソフトウェアにおける FMEA の効率的・効果的な実施方法とその効果”，先進的な設計・検証技術の適用事例報告書 2016 年版，独立行政法人情報処理推進機構（IPA），2016.
[8] 細川宣啓，西康晴，嬉野綾，野中誠，原佑貴子，“過失に着目した欠陥のモデリング ーバグ分析はなぜうまくいかないのか？,” JaSST，13 Tokyo，2013.