

ソフトウェア欠陥予測アルゴリズム

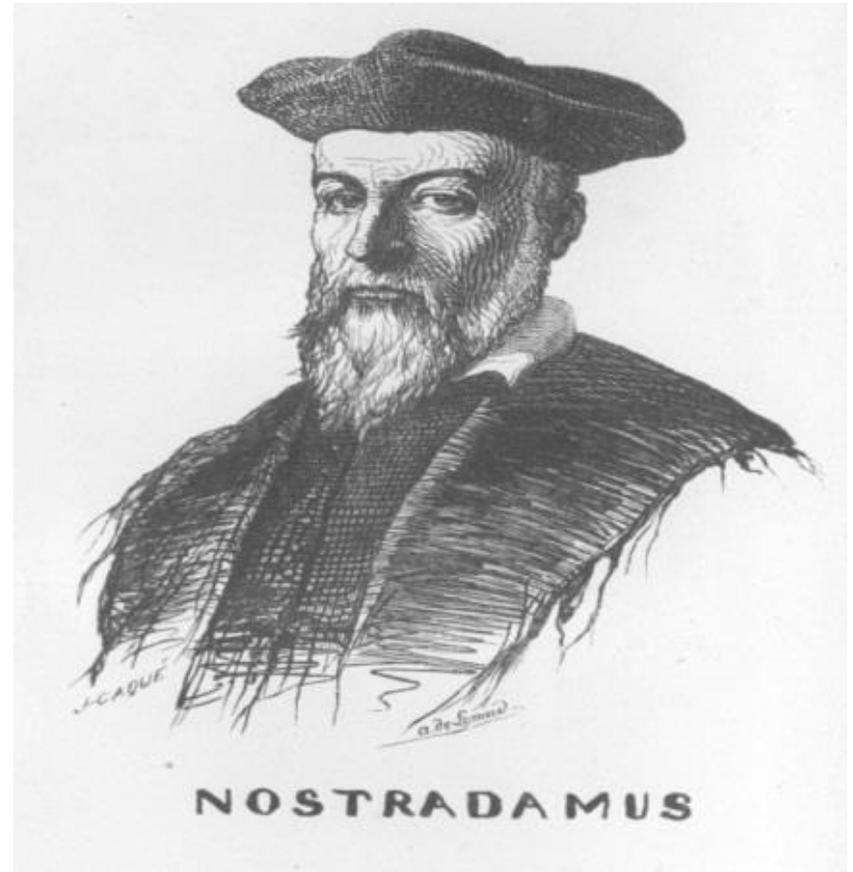
～欠陥混入メカニズムのモデリング手法を
利用した欠陥予測方法の提案～

2014年度第7分科会 Predictorチーム

主査	: 細川 宣啓	日本アイ・ビー・エム(株)
副主査	: 永田 敦	ソニー(株)
研究員	: 柏原 一雄	(株)デンソークリエイト
	岡本 晃	農中情報システム(株)
	鈴木 裕一郎	(株)日立製作所
	田村 光義	サイバートラスト(株)
	東久保 理江子	アンリツ(株)
	保栖 真輝	日本電子(株)

目次

- 目指す姿
- 背景
- 研究課題
- 解決のアイデア
- 提案手法
- 実験
- まとめ



※Predictor・・・予言者、予報者

目指す姿

• 欠陥予測とは

- プロジェクトマネージャやSQAが、プロジェクトの特徴をもとに、蓄積している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欠陥を特定すること



• 欠陥予測の狙い



欠陥を予測し、脱炎上プロジェクト！市場バグ防止！

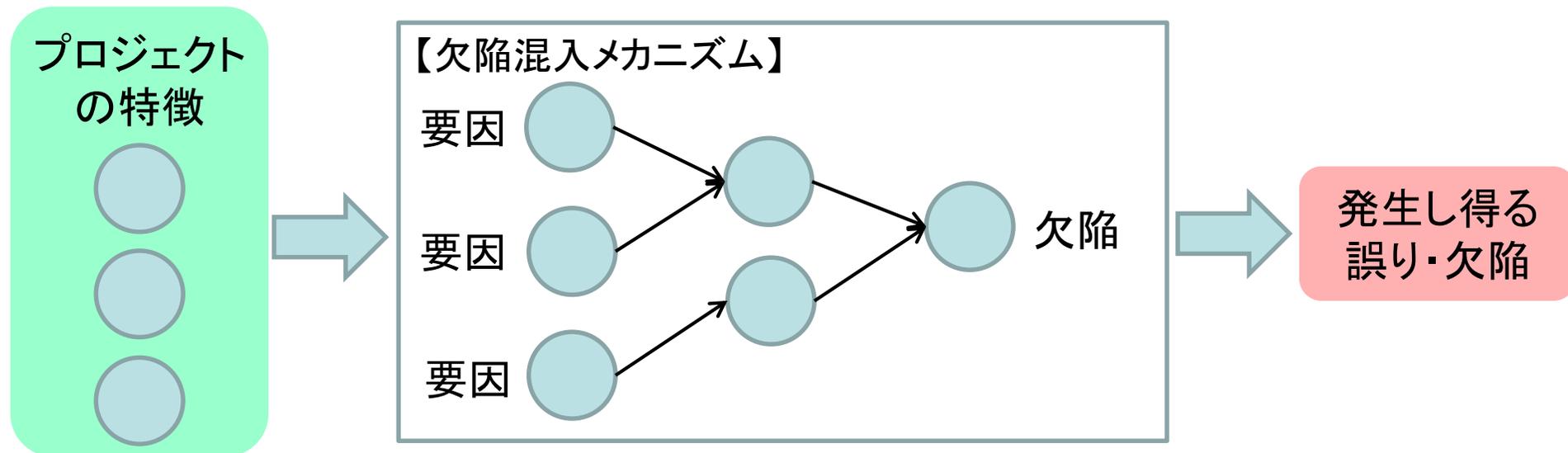
背景：欠陥の予測に必要なこと

- 前提

- 過失・欠陥は同じメカニズムで繰り返し発生する
- メカニズムがわかれば、要因から発生する事象が予測できる

- 欠陥の予測に必要なこと

- 混入した欠陥や発生した不具合の情報だけでなく、
欠陥混入に至るメカニズムを示す情報が必要不可欠



- ・ 既存の不具合DBの問題
 - 欠陥混入メカニズムが理解できない
 - ・ 表現が一般化されていない
 - ・ 欠陥混入の要因(欠陥を混入させた人間の過失、人間の過失を引き起こす誘発因子)が記録されていない

– 不具合DBが欠陥混入予測のために利用しにくい

- ・ 不具合、欠陥とその要因を区別して表現していない

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(過失)	不正確な結果を生み出した人間の行為

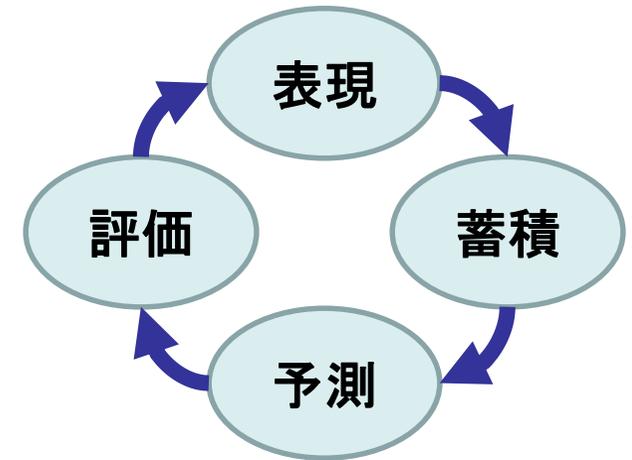
- ・ 不具合DBから欠陥を予測する手順が確立されていない

**そもそも、予測という用途を考慮していない
使われない情報は、残らない・整理されない**

- ・ 過失・欠陥の混入予測を可能にする手法を開発する

【開発手法】

- 欠陥混入メカニズムの**表現**手法
- 欠陥混入メカニズムの**蓄積**手法
- 欠陥混入メカニズムを利用した**予測**手法



- ・ 既存の仕組みの問題が解決できることを評価する

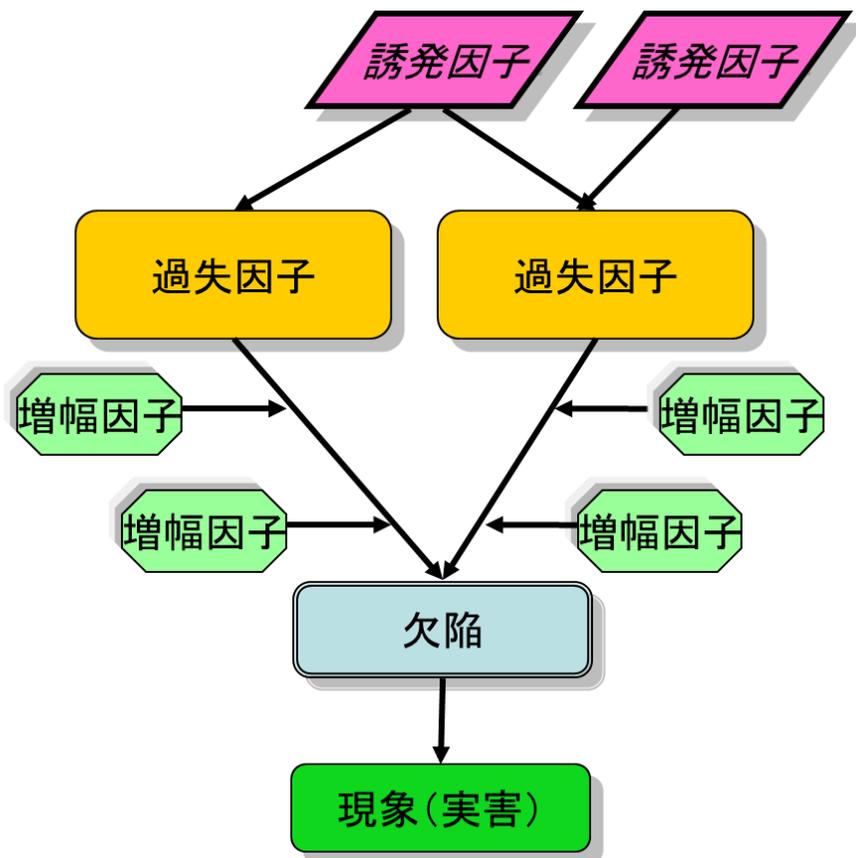
【解決対象の問題】

- 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

欠陥混入メカニズムから、過失・欠陥を予測する

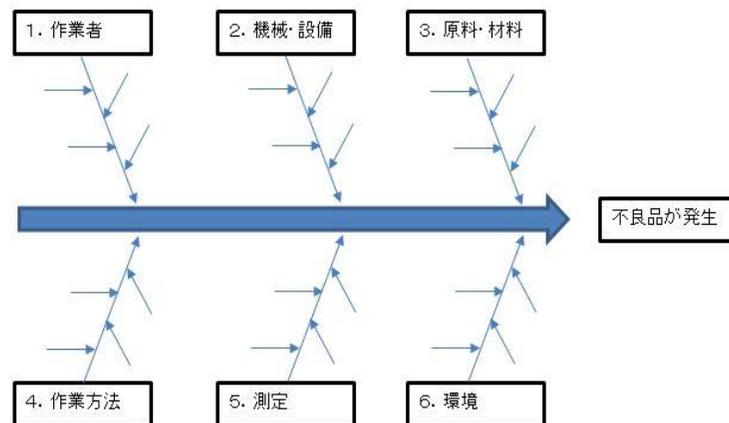
解決のアイデア

欠陥モデリング



特性要因図と5M1E

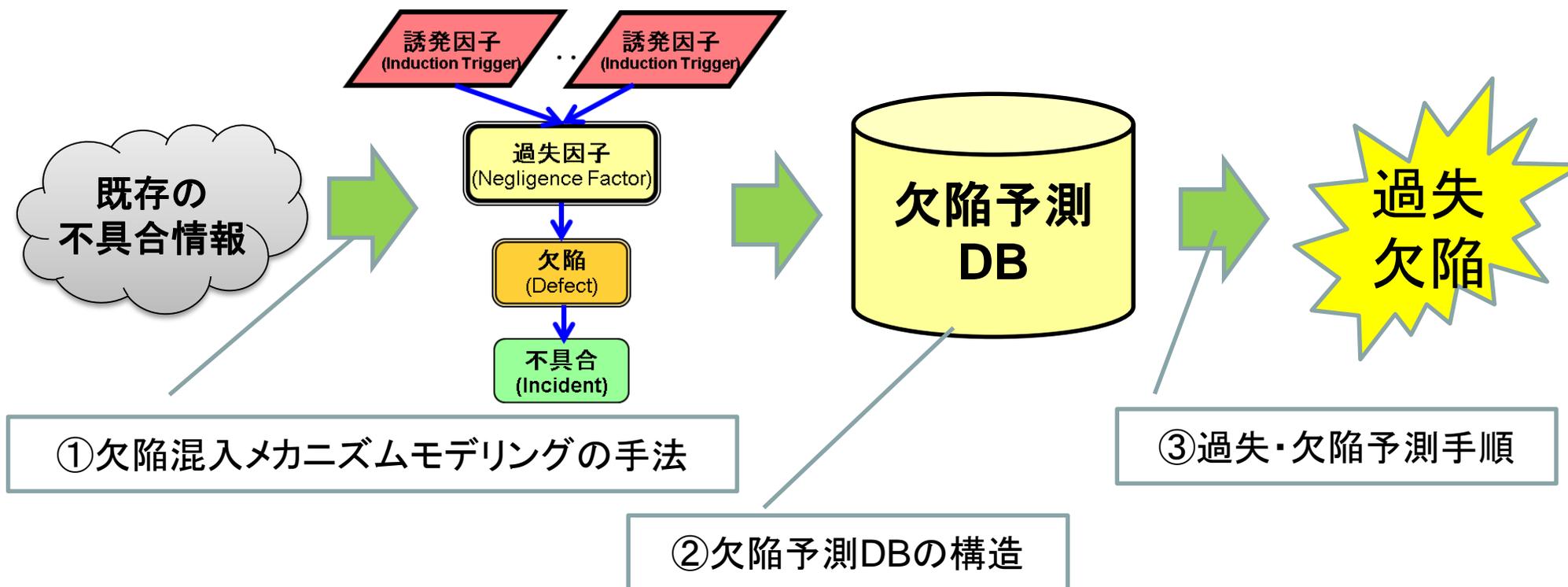
- 人 (Man)
- 機械・設備 (Machine)
- 原料・材料 (Material)
- 方法 (Method)
- 測定 (Measurement)
- 環境 (Environment)



**欠陥の情報を構造化して表現する
表現するのは、欠陥 と 人の過失 と その誘発因子**

提案手法:ソフトウェア欠陥予測アルゴリズム

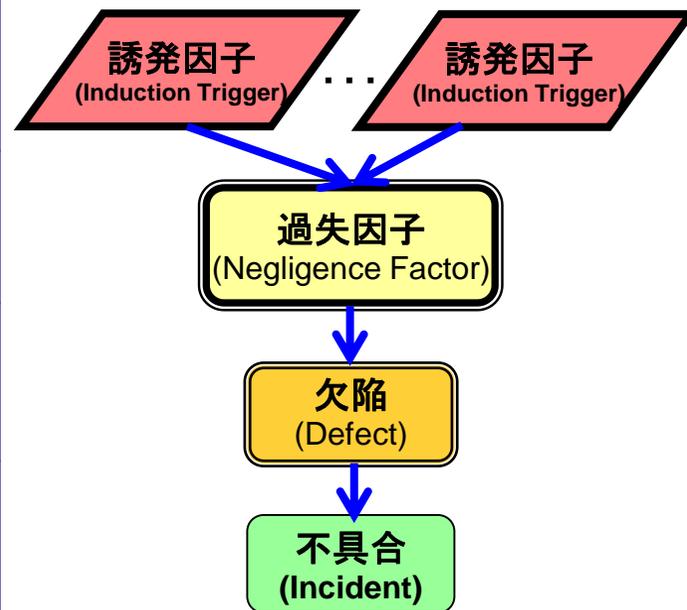
- ソフトウェア欠陥予測アルゴリズムの技術的要素
 - 欠陥混入メカニズムモデリングの手法
 - 欠陥予測DBの構造
 - 過失・欠陥予測手順



提案手法：欠陥混入メカニズムのモデリング手法

表記ルール：ノードの種類

名称	ノード	解説
不具合		ソフトウェアの振舞いが正しくないという現象のこと 欠陥によって引き起こされる
欠陥		不具合をもたらした原因である成果物の記述のこと 人間の誤りによって混入する
過失因子		人間の思考や判断の誤りそのもののこと 欠陥を混入させた誤り
誘発因子		欠陥を混入させた誤りを誘発する“トリガー”となる要素のこと



表記ルール：ノードのつなぎ方

- 1つ以上の「誘発因子」と1つの「過失因子」をつなげる
- 「過失因子」と「欠陥」は1対1でつなげる
- 「欠陥」と「不具合」は1対1でつなげる

理解し易くするため、欠陥混入の主要因の関連のみを示す

提案手法：欠陥混入メカニズムのモデリング手法

• モデリングのステップ

1. 既存の欠陥情報をもとに、「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. 業務や製品固有の言葉を避け、一般化して表現する
3. 一般化された欠陥混入メカニズムを**レビュー**する



• 欠陥混入メカニズムのレビュー観点

理解容易性 表現が一般化されているか

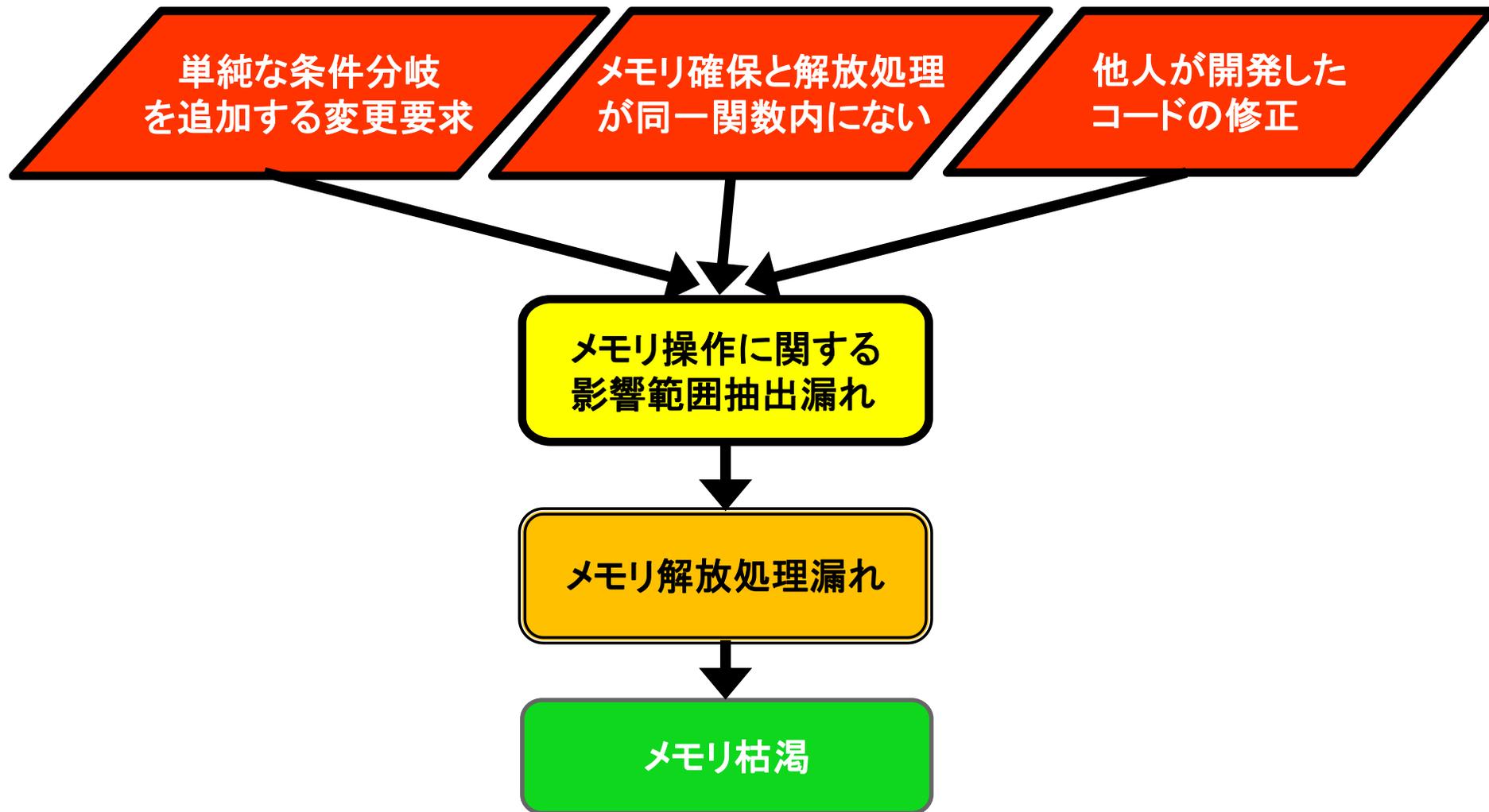
納得性 過失因子に対して必要十分な誘発因子が挙げられているか

利用可能性 開発開始前に、誘発因子の該当/非該当の判断ができるか

**“人は誤る”ことを前提に、メカニズムを明らかにする
「そりゃ、バグも入り込むなあ」と、納得できるか？**

提案手法：欠陥混入メカニズムのモデリング手法

• 例：モデリング



提案手法：過失・欠陥予測手順

プロジェクト現況



【誘発因子のリスト】

誘発因子分類	誘発因子
変更要求	単純な条件分岐の追加
	メモリ配置の変更
	並列処理を可能とする変更
	データ構造の変更
	サポート対象の機器・関連ソフトの追加
	開発途中での一部変更の見送り
入力	動作実績のあるソフト(機能)の移植
	サンプルプログラムの利用
	利用実績のないOSの使用
ベース	仕様書、設計書なし
	コードと仕様書、設計書に一貫性なし
	コードと関連する仕様が追跡困難
	仕様書に例外処理の記載なし
	ベースソフトの仕様が不統一
	メモリ確保と解放処理が同じ関数内にない
	同じ用途のデータが複数存在
	NULLの扱いが異なるパラメタが混在
	変数に誤解釈を誘発するコメントあり
	コメントにパラメタの値域説明なし

現況と一致する
誘発因子



予測した
過失因子・欠陥

誘発因子のリストは、プロジェクトの問診票として利用

・【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か？

- 初めてモデルを見た人から理解・納得を得られるかをアンケート方式で確認
 - ：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

・【実験2】

「誘発因子」を「過失因子」の予測に利用できるか？

- 終了プロジェクトで発生した「過失因子」に対して、欠陥予測DB の情報をもとに予測できたかを確認
- 異なる6組織の欠陥情報：44件を蓄積した欠陥予測DBを実験に使用
- 終了プロジェクトの状態に該当する「誘発因子」を選択し、欠陥予測DBから導き出した「過失因子」と実際に発生した「過失因子」の一致率を算出

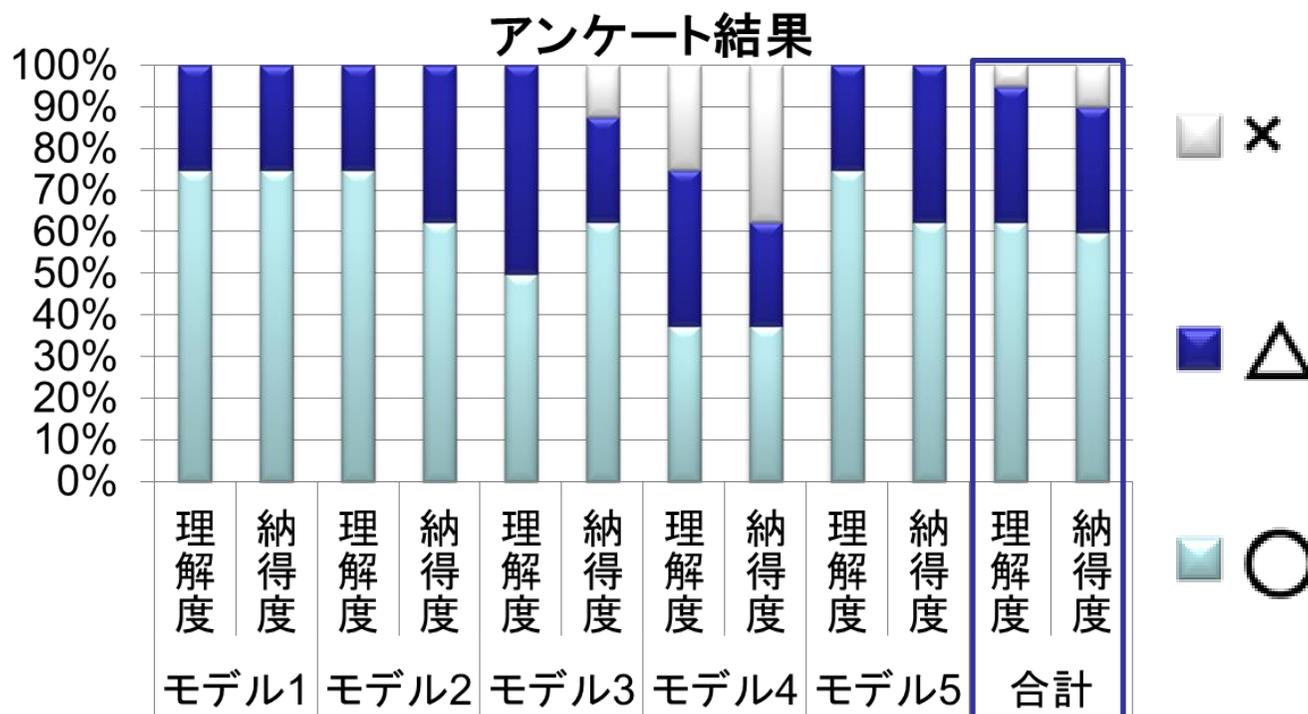
**異なる複数の組織で実験を実施、
組織が変わっても提案手法が有効であることを確認**

実験：実験結果

【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か？

➡ 約6割程の人の理解・納得を得られた。一部のモデルは改善の余地あり。



モデリング手法を用いることで、
欠陥混入メカニズムを伝達し、理解・納得を得られる

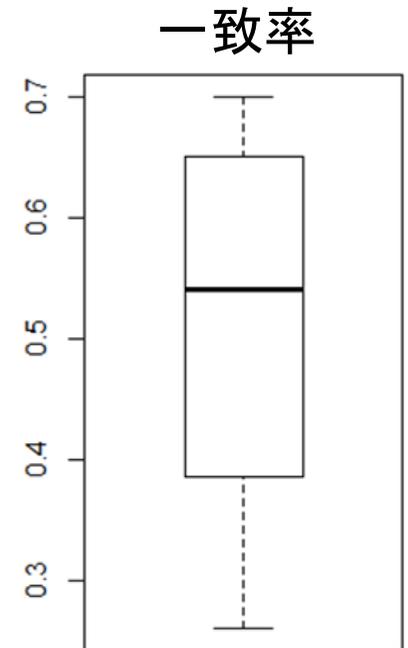
実験：実験結果

• 【実験2】

「誘発因子」を「過失因子」の予測に利用できるか？

➡ どのプロジェクトでも、一致する(予測できた)過失因子が確認できた。

プロジェクト	欠陥数	一致した過失因子数	一致率
A	28	11	39%
B	24	13	54%
C	52	36	69%
D	24	15	63%
E	43	30	70%
F	6	2	33%
G	6	4	67%
H	19	5	26%
I	16	6	38%
J	19	10	53%
合計	237	132	56%



**欠陥予測DBと過失・欠陥予測手順を利用することで、
「誘発因子」から「過失因子」の候補を特定する行為が可能**

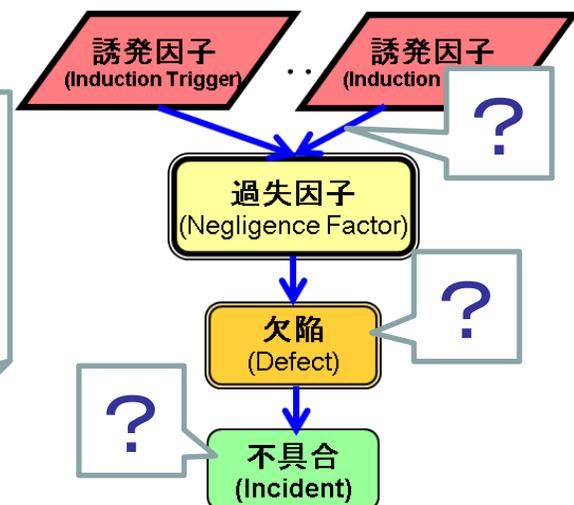
実験：課題

課題①

表記ルールの改善、モデルの検証方法確立

【一部のモデルに対するコメント】

- ・不具合や欠陥の表現が理解しづらい
- ・「誘発因子」と「過失因子」の関係が納得できない
- ・「欠陥」に成果物の問題ではなく人の誤りが示されている



課題②

提案手法の有効性に関する追加確認

【未確認事項】

- ・欠陥の予測可能性
- ・予測精度
- ・品質向上における効果



欠陥予測アルゴリズムの改善のきっかけが得られた

まとめ

・ まとめ

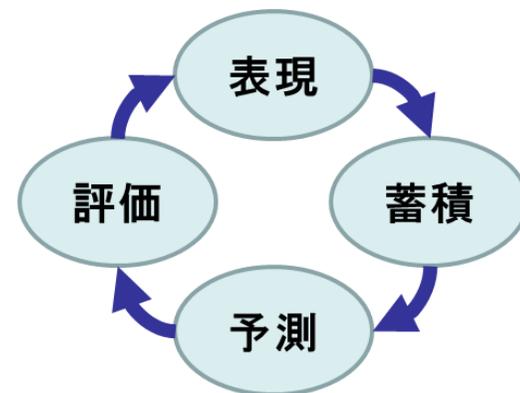
- 欠陥混入メカニズムのモデリング手法で、欠陥混入メカニズムを伝達し、理解・納得を得られるようになった
- 欠陥予測アルゴリズムで、「誘発因子」から「過失因子」を予測する行為が可能となった

・ 期待できる効果

- テストやレビューの観点抽出に利用し、欠陥を効率的にピンポイントで検出
- 過失の要因を開発前に除去し、欠陥の混入を防止
- 欠陥混入メカニズムをソフトウェア業界全体で資産として共有し利用

・ 今後の課題

- 欠陥混入メカニズムのモデリング手法の改善
- 欠陥予測DBの拡充
- 過失・欠陥とそれを防止・除去する技術を繋ぐ
- 欠陥予測アルゴリズムの有効性・効果の確認



ご清聴

ありがとうございました