

## COSEプロジェクトにおけるEASE研究の適用事例 障害修正工数の分析

奈良先端科学技術大学院大学  
門田 暁人

出典 松村 知子, 門田 暁人, 森崎 修司, 松本 健一, ``マルチベンダ情報システム開発における障害修正工数の要因分析,`` 情報処理学会論文誌, Vol.48, No.5, pp.1926-1935, May 2007.



### 背景

- よく知られる仮説を実データにより検証したい。
  - 仮説1: 障害の発見が遅れるほど, 修正工数が増大する.
    - どのくらい増大するのか?
  - 仮説2: 障害の滞留時間が長いほど, 修正工数が高くなる.
    - どのくらい高くなるのか?
- これらの仮説が正しいことを実データで示せば,
  - 上流工程でのレビュー, 品質の作りこみの重要性を説明しやすくなる.  
(実データを示さずに, 品質を作りこめと言っても説得力がない)
- 他にも修正工数に増大する要因があれば明らかにしたい.  
(残業時間を減らす一つの方法は, 修正工数を減らすこと)

2



# 対象プロジェクト・収集データ項目

- **対象プロジェクト:プローブ情報システム開発**
  - ベンダ5社7拠点による分散開発
  - 2005年4月～2005年10月第1フェーズ開発のデータが対象
- **目的変数:**
  - **修正工数(人時)**
- **説明変数:**
  - (拠点名), 起票分類, 発見箇所(新規, 改造, 再利用), **発見工程**, 発見作業, 問題処理機能, 再現度, 重要度, 優先度, **混入工程**, 問題原因, 修正工程, 問題を発見できなかった要因 (=発見遅延の原因)

3

Copyright © 2007 Nara Institute of Science and Technology



項目名	選択項目(カテゴリ)	障害件数	項目説明	項目名	選択項目(カテゴリ)	障害件数	項目説明		
発見箇所	新規	963	障害を発見したモジュールの再利用の状況	優先度	高	320	障害解決の優先度合い		
	改造	12			中	611			
	再利用(未改造)	2			低	46			
発見工程	未入力	13	障害を発見した工程	障害原因	未入力	47	障害混入原因の種類		
	コーディング/単体試験	650			論理エラー	288			
	結合試験	279			演算エラー	52			
	総合試験	33			インタフェース・タイミング問題	56			
発見作業	分析・解析中	108	障害を発見した作業	データ処理問題	361				
	システム動作中(テスト運用)	827		データ範囲不正	30				
	レビュー	42		データ問題	17				
	入力データチェック機能	105		仕様書(設計書)不正	27				
機能	演算機能	111	障害を発見したシステムの機能	仕様書(設計書)品質問題	2				
	データ編集機能	254		機能強化・拡張	20				
	ファイル更新機能	15		性能問題	8				
	データ出力機能	176		相互操作性問題	3				
	連動(組み合わせ)処理	17		標準一致問題	1				
	限界処理	4		ハードウェア障害	0				
	外理条件異常検知機能	2		操作ミス	24				
	その他	293		指摘ミス(仕様どおり)	7				
	再現度	再現頻度大		704	故障発生時の再現頻度		原因不明	3	
		再現度小		94			その他	31	
再現なし		35	発見遅延の原因	未入力		331			
不明(未確認)		144		レビュー未実施		3			
重要度	重大	340	障害の重要さの度合い	レビュー指摘もれ	35	障害を混入工程もしくは本来発見すべき工程で発見できなかった理由			
	中度	503		再レビュー及び修正確認もれ	12				
	軽微	132		工程間引継ぎ	1				
混入工程	未入力	41	障害混入の原因となった工程	コミュニケーション不足	1				
	基本設計	1		試験項目抽出もれ	26				
	詳細設計	86		テストそのものもれ	6				
	コーディング/単体試験	789		環境上の問題で	11				
	結合試験	57		後工程にもっていった	5				
	総合試験	3		結果確認ミス	5				
				その他	547				

4



## 収集データ項目の決定・データ収集準備

- **記録する障害情報の決定**
  - 各社で用いられている障害票やIEEE Standard.1044 Classification for Software Anomaliesをベースに、隔週のミーティングで議論(2005/3～2005/6)
- **修正工数入力依頼**
  - 工数の入力・公開に調整が必要(社間機密情報)
  - 従来、修正工数を収集していたのは2社のみ
  - 手入力→入力者による多少のばらつきなどは許容する
  - 人時入力
- **障害追跡システム(GNATS)のプロトタイプ作成・配布**

5

Copyright © 2007 Nara Institute of Science and Technology



## 収集データ概要

- **障害データ数** : 1182件
- **修正工数統計値**
  - 修正工数入力データ 1063件
  - 修正工数欠損データ 119件
  - 最小値=0が1件→欠損値扱い
  - 0を除くと、0.08人時(約5分)が最小値
  - 最大値は92人時(3.8人日)
  - 欠損値は、分析対象から除いた。

6

Copyright © 2007 Nara Institute of Science and Technology



## 分析データ説明

- 工程分類

- 基本設計, 詳細設計, コーディング/単体試験, (拠点内)結合試験, 統合環境結合試験, 総合試験

- データ前処理

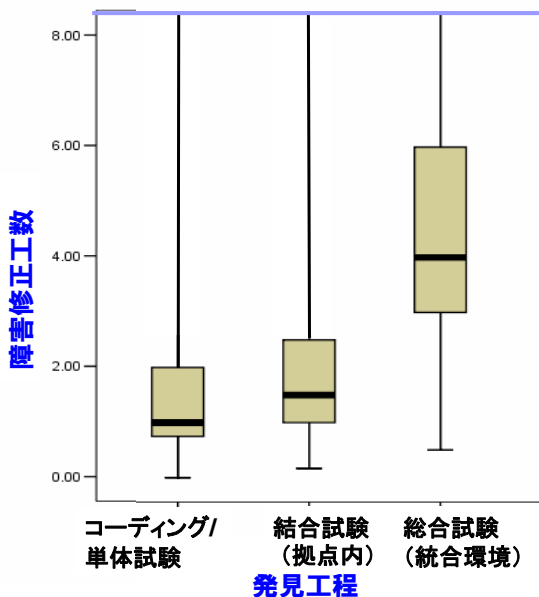
- 統合環境結合試験, 総合試験はデータ数が少ないので, 「総合試験」に統合
- 「発見遅延」は, 「発見工程」と「混入工程」が同じ場合0, その他は工程差ごとに±1する
- 「発見遅延の原因」は, 「発見遅延」が1以上の場合にのみ有効とする

7

Copyright © 2007 Nara Institute of Science and Technology

**EASE**  
EASE PROJECT

## 発見工程の分析



障害の発見が遅れるほど,  
修正工数は増大する!

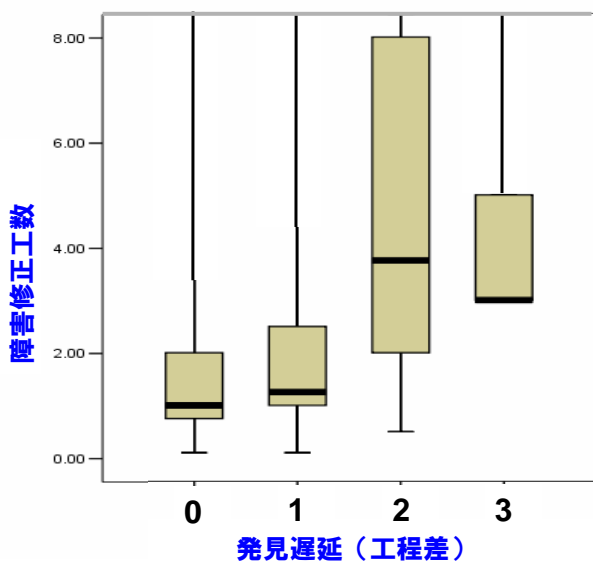
コーディング/単体試験  
↓4.88倍  
総合試験

8

Nara Institute of Science and Technology

**EASE**  
EASE PROJECT

## 発見遅延の分析



障害の滞留時間が長いほど、修正工数が高くなる！

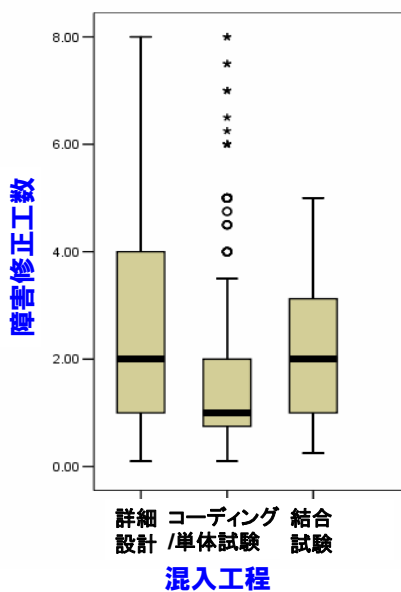
すぐに発見  
↓ 4.44倍  
2工程後に発見

9

Nara Institute of Science and Technology

EASE  
EASE PROJECT

## 混入工程の分析



- 上流工程(設計)のバグは修正工数大
- 結合試験で混入したバグは修正工数大  
→ 回帰テストの重要性

10

Copyright © 2007 Nara Institute of Science and Technology

EASE  
EASE PROJECT

## さらなる分析から・・・

- **設計バグ**は、遅くとも単体試験で発見することが望ましい。結合試験まで発見が持ち越されると修正工数が増大した。
- **コーディングバグ**は、単体試験で発見することが望ましいが、結合試験まで持ち越しても致命的ではなかった。しかし、結合試験まで持ち越すと修正工数が著しく増大した。
- どの工程で混入したバグも、結合試験まで持ち越すと、修正工数が増大した。



- 設計レビューは重要である。
- 結合試験でバグをしっかり取ることが重要である。  
(結合試験でのデグレード回避(回帰テスト)が重要)

11

Copyright © 2007 Nara Institute of Science and Technology



## その他の結果

説明変数	カテゴリ	障害数	平均修正工数
発見遅延の原因	試験項目抽出漏れ	26	5.84
	環境上の問題で後工程に持っていった	11	7.55
	結果確認ミス	5	4.00
障害原因	指摘ミス(仕様どおり)	7	0.75
	その他	31	1.15
再現度	小	94	3.70
	再現なし	35	0.44
重要度	軽微	132	0.88
発見作業	レビュー	42	0.99
(参考) 全障害		978	2.13

詳細は、論文を御参照ください。

松村 知子, 門田 暁人, 森崎 修司, 松本 健一, ``マルチベンダ情報システム開発における障害修正工数の要因分析,`` 情報処理学会論文誌, Vol.48, No.5, pp.1926-1935, May 2007.

12

Copyright © 2007 Nara Institute of Science and Technology



## まとめ

- よく知られる仮説を実データにより検証した。
  - 仮説1:障害の発見が遅れるほど、修正工数が増大する。
    - コーディング/単体試験... 平均1.70人時
    - 総合試験... 平均8.29人時 (4.88倍)
  - 仮説2:障害の滞留時間が長いほど、修正工数が高くなる。
    - すぐ発見... 平均1.70人時
    - 2工程以上遅れる... 平均7.86人時 (4.44倍)
- 上流工程でのレビューの重要性を確認できた。
- 結合試験の重要性(総合試験までバグを持ち越さない)を確認できた。

13

Copyright © 2007 Nara Institute of Science and Technology



## その後の活動

- 修正工数の細分化(第2フェーズでの活動)
  - 分析・解析工数
  - 修正工数
  - 確認工数に分類して、より詳細なデータ計測・分析を目指した。
  - しかし、工数データ取得に失敗(入力が粗くなった)
- 修正工数が大きくなる条件のルール化
  - NEEDLEによる分析を実施 → より詳細なデータの特徴分析が可能になり、プロセス改善など具体的な対策指針を抽出できた\*

### \*参考文献:

> 森崎 修司: "相関ルール分析 NEEDLEのバグ票とプロジェクトデータへの適用", 第14回EASE研究会, 2007/10, [http://www.empirical.jp/download/past/publicdata/14th\\_kenkyukai/3.pdf](http://www.empirical.jp/download/past/publicdata/14th_kenkyukai/3.pdf)

> 森崎, 門田, 玉田, 松村, 松本: "Defect Data Analysis Based on Extended Association Rule Mining", Proceedings of International Workshop on Mining Software Repository (MSR2007), pp.17-24, 2007.

14

Copyright © 2007 Nara Institute of Science and Technology

