2014年度 SQiP研究会　第7分科会(DBDPチーム)

欠陥モデルに基づいた ソフトウェア欠陥流出防止アプローチの提案

－　処理に着目し欠陥の特徴を捉える　－

A Proposal of approach using **Defect Based Differ Prevention** in software development

主査　　細川　宣啓　（日本アイ・ビー・エム株式会社）
副主査　　永田　敦　　（ソニー株式会社）

2015年２月27日
発表　　加藤　賀久　　（オムロン株式会社）

1. はじめに

世の中の多くの製品はソフトウェアで動いている。
ソフトウェア**欠陥**を市場に**流出**させてしまうと**社会に大きな影響**を与えてしまう。
研究チームが所属する部門の製品についても同じである。

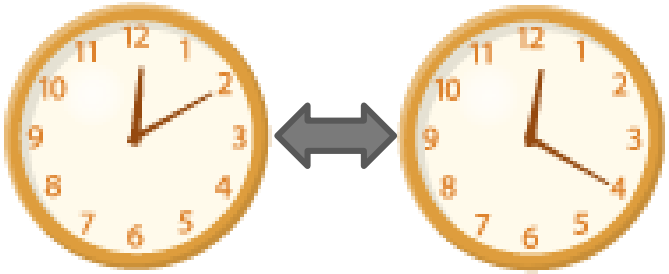
研究チーム：DBDP（Defect Based Differ Prevention）チーム
研究員　加藤　賀久　（オムロン株式会社）
福田　伊津子　（株式会社東芝　社会インフラシステム社）
齋藤　幸裕　（ビジネスキューブ・アンド・パートナーズ株式会社）

2．欠陥流出の現状

流出した欠陥が**以前と似ている**なんて**感じたこと**ありませんか？

- **また、**データにゴミが入っていた　とか？

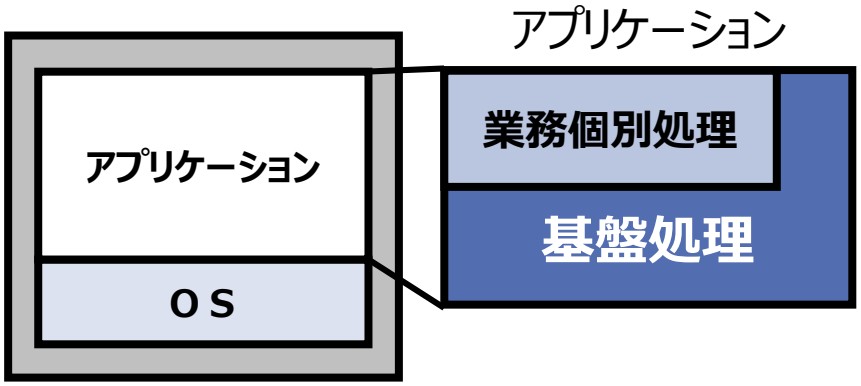
- **この間も**タイミングがずれて・・・　とか？

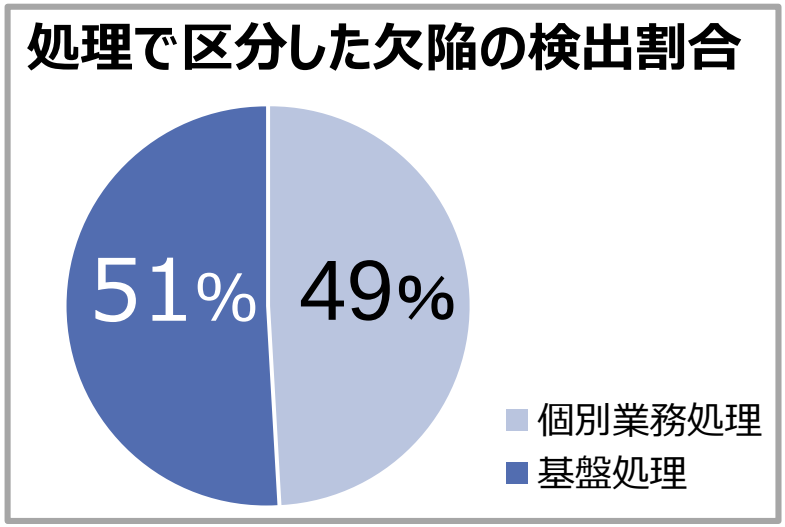


アンケートの結果でも**92**%の人が**以前と似ている欠陥**を検出したことがあると**感じていました！**

3．課題は欠陥流出

再発防止、未然防止は実施しているのに**欠陥流出**は続く。
なぜ・・・？　それも同じような・・・　そこで、
我々研究チームは、**欠陥そのもの**に着目してみた。どのシステムにも実装される**共通的な処理**を**「基盤処理」**と定義し、「基盤処理」ごとに欠陥流出を防止するアプローチを探った。



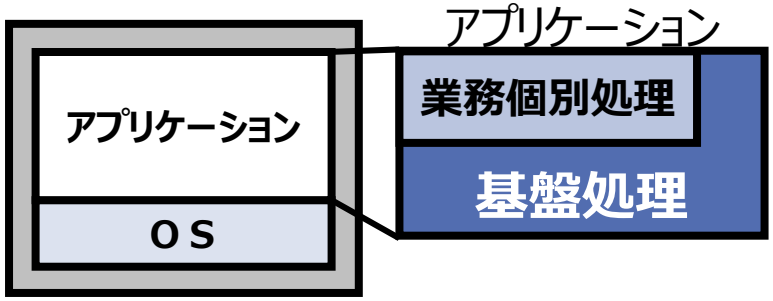


＊調査対象：研究チームの開発製品

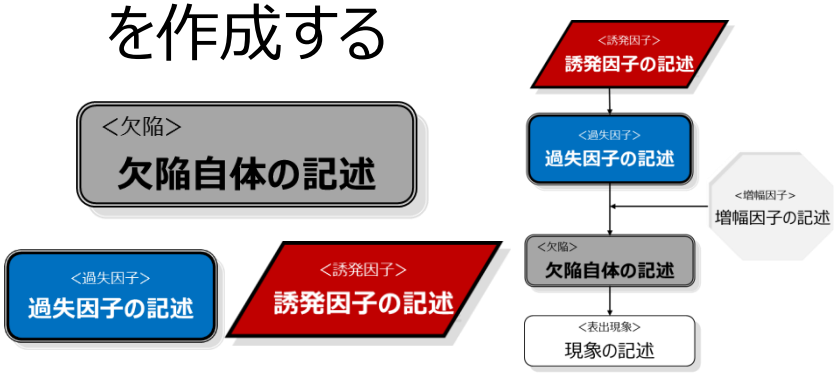
4. 欠陥流出防止アプローチの提案

欠陥そのものに着目し、より上流の開発工程で防止できるDBDP（Defect Based Differ Prevention）アプローチを提案する。

① 過去の欠陥を基盤処理で分ける



② 基盤処理単位で欠陥モデルを作成する



③ 誘発因子の特徴を抽出する

What

４Ｈ１Ｒ３Ｃ

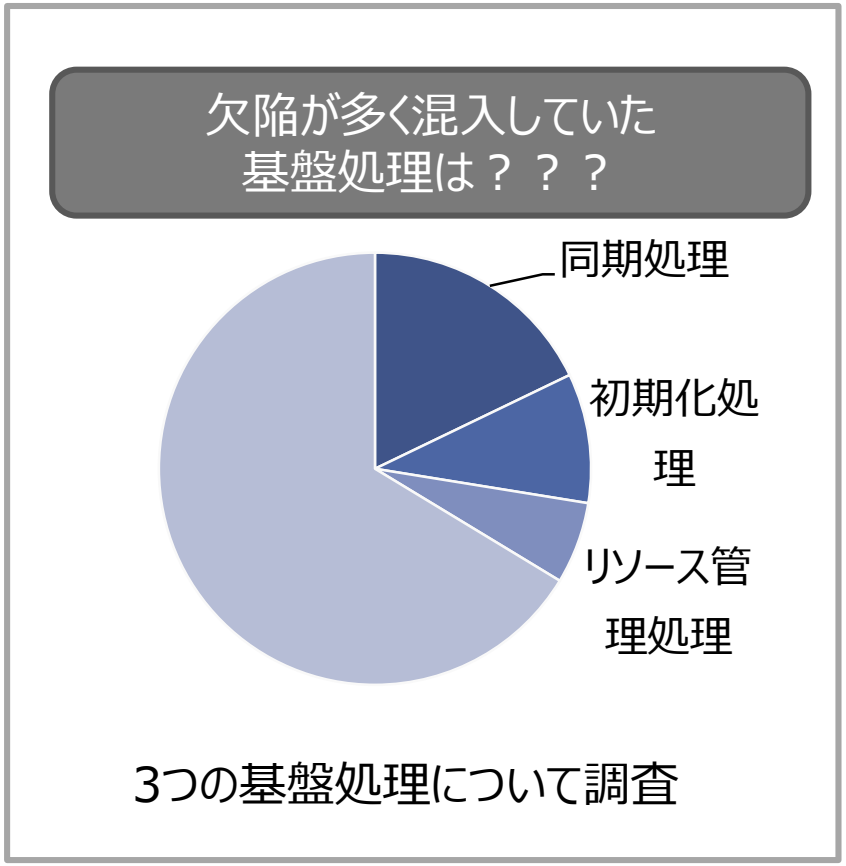
④ 設計／実装前に処理毎に確認する

① 過去の欠陥を基盤処理で分ける

欠陥が混入されていた処理が基盤処理であるか確認する。
基盤処理であれば、処理毎に欠陥を分ける。

基盤処理：システムに依らない共通に実装される処理

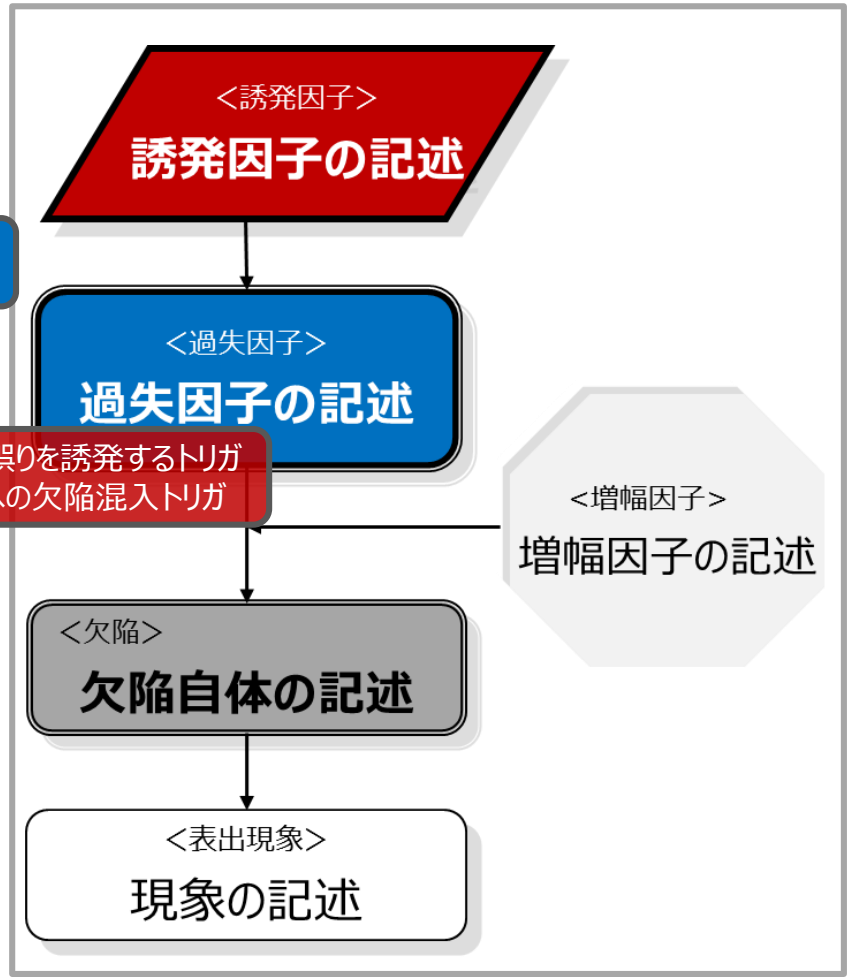
連接処理	**異常判定処理**	**・・・**
初期化処理	**同期処理**	**リソース管理処理**
通信処理		



② 基盤処理単位で欠陥モデルを作成する

Project Fabreの欠陥モデリングに従い、欠陥モデルを作成する。

- 表出した事象から配置する
- 欠陥モデルにつき、**過失因子**は**１つ**のみとする
- **過失を招く要因**となった**誘発因子**は複数あってもよい
- **レビュー、テスト**は**増幅因子**とする
- **欠陥**及び**過失因子**から対象とした**基盤処理**を**整理**する



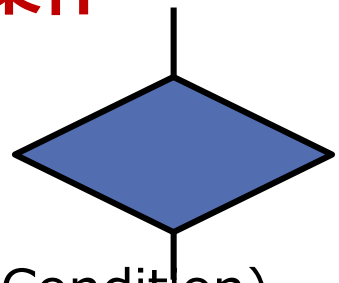
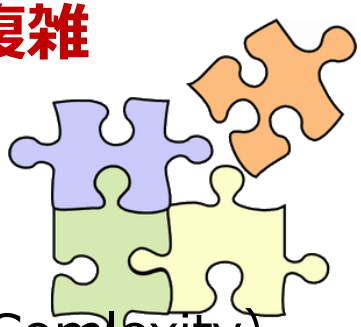
③ 誘発因子の特徴を抽出する

欠陥モデルの誘発因子から設計として押さえるべき項目を抽出する。

初期化処理	同期処理	リソース管理処理
・初期化条件 ・初期化タイミング ・初期値 ・初期化範囲 （容量）	・タイミング ・対象処理 ・処理シーケンス	・メモリマップ ・メモリサイズ ・アクセス要領 （出力要領） ・制約／条件

③ 誘発因子の特徴を抽出する

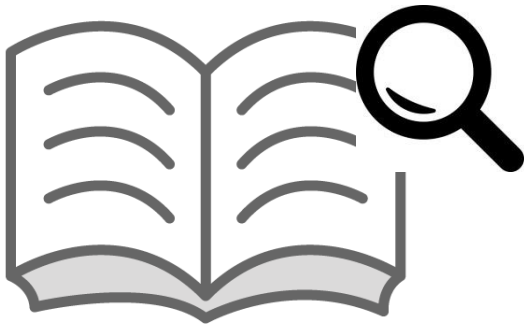
誘発因子のキーワードを定義。思い当たるものがあれば要注意。

4H			
初めて  (Hajimete)	変更  (Henkou)	ひとりで  (Hitoride)	急いで  (Hurry)
1R	**3C**		
流用  (Ryuyo)	条件  (Condition)	複雑  (Comlexity)	Communication 

④ 設計／実装前に処理毎に確認する

次の２つの観点で**設計内容**の**事前チェック**、**設計結果**の**確認**を実施する。

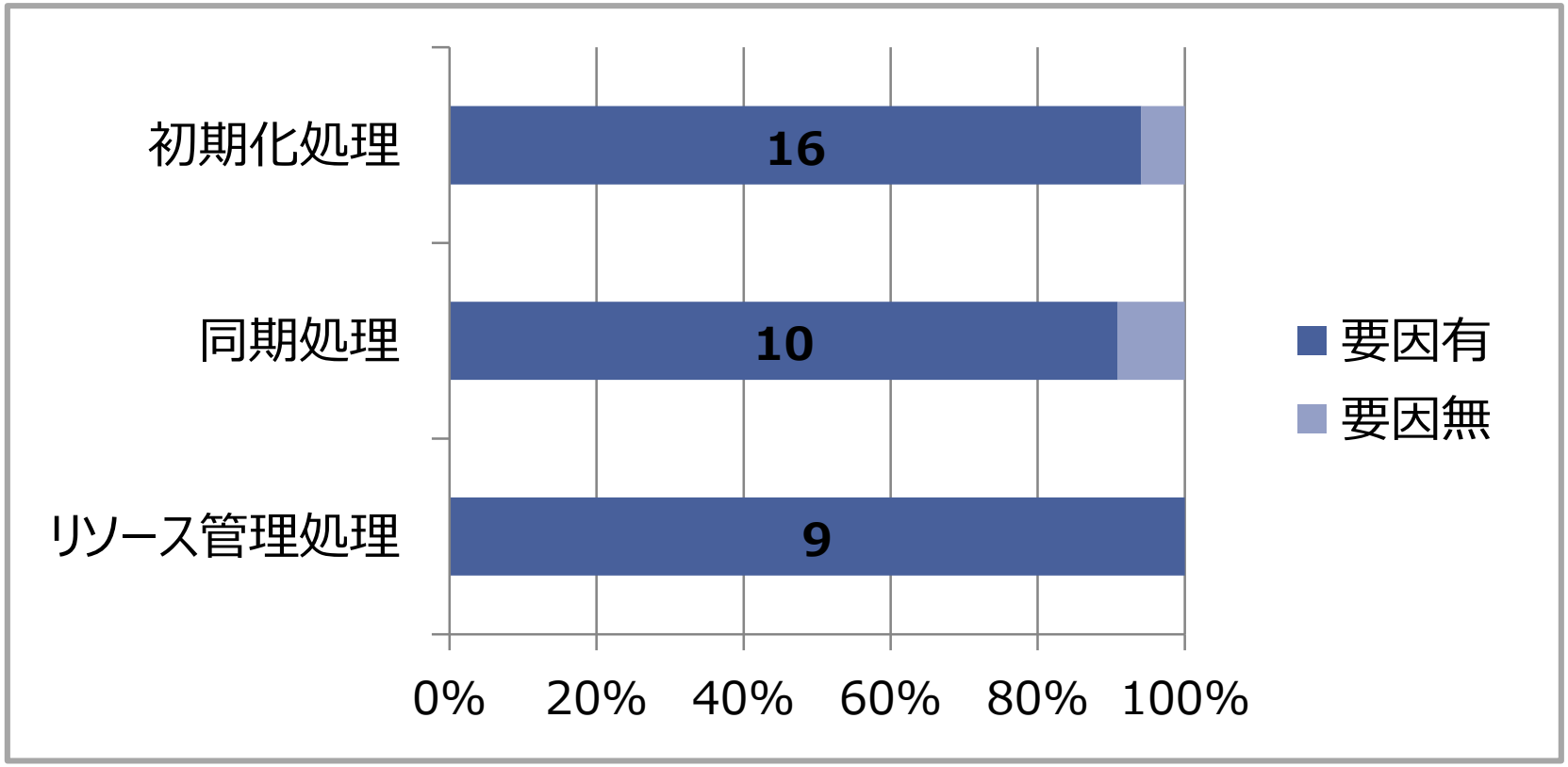
- **基盤処理**
- **欠陥流出防止抽出項目**



欠陥流出防止リスト（欠陥流出防止抽出項目）	
同期処理	・ハードウェア，担当者等変更はないか ・初めて担当する人，器材はないか ・流用することでタイミング，同期対象処理等に問題はないか ・条件による同期，複雑な設計になっていないか
【What】 ・タイミング ・対象処理 ・処理シーケンス	

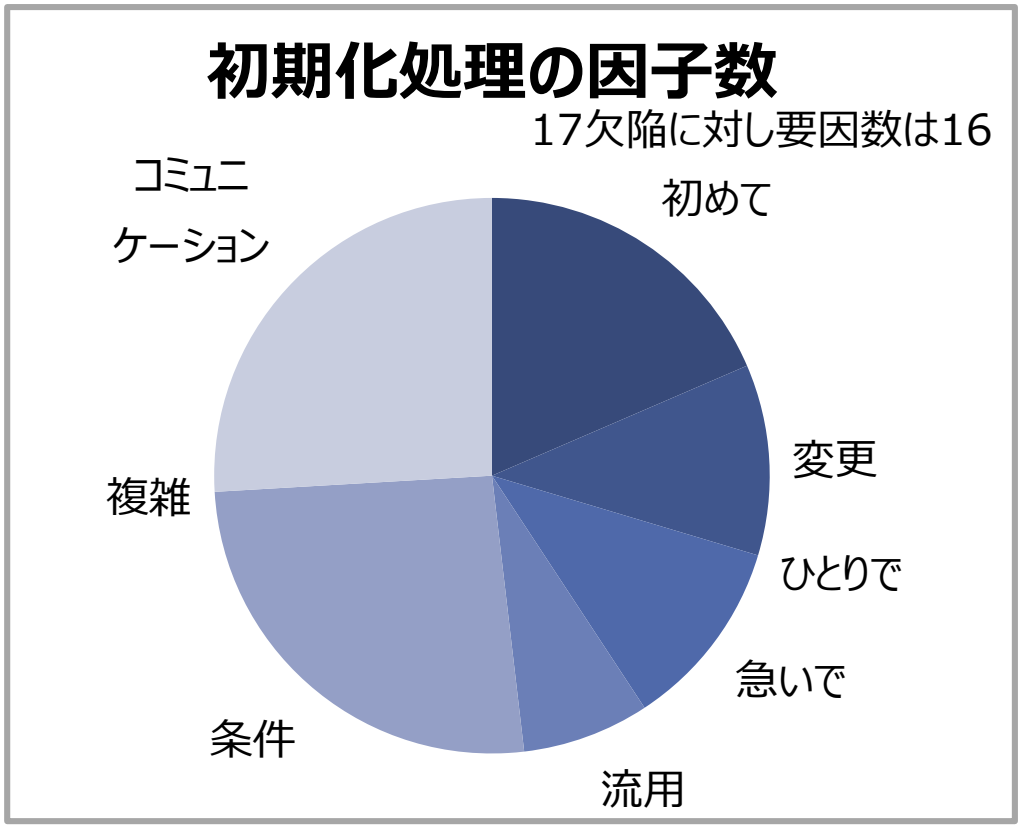
5. 実験(1/3)

実験で収集した欠陥では、４H1R3Cが多くの欠陥の誘発因子となっていた。



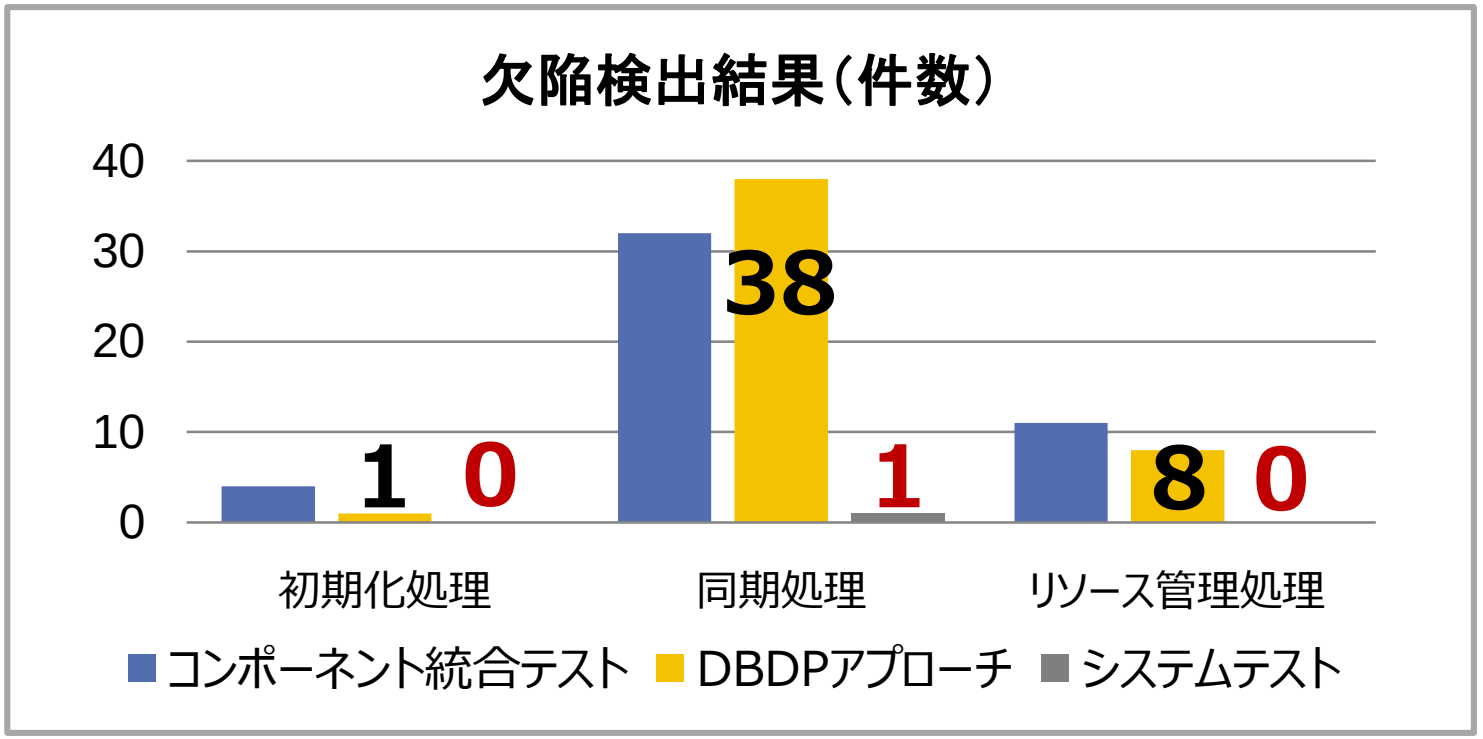
5. 実験(2/3)

初期化処理では、「ひとりで」、「複雑」の因子はなかった。
アンケート(27名）では、「流用」が誘発因子になり得ると回答した人が85%いた。



5. 実験(3/3)

欠陥流出防止リストを作成しシステムテスト前に適用してみた。



DBDP(Defect Based Differ Prevention）アプローチ：欠陥流出防止アプローチ

6. 今後の課題

- 基盤処理を増やす
- 処理ごとの欠陥モデルを増やし、誘発因子を分析する
- 上流工程での活用を試みる

DBDP（Defect Based Differ Prevention）アプローチのプロジェクトへの適用を広げ、有用性・活用容易性を評価していく。

欠陥流出防止リストはまだ途上、多くの処理と多くの欠陥モデルで組織・企業のノウハウが確立される。

7. まとめ

基盤処理ごとに**欠陥モデル**から作成した**欠陥流出防止リスト**は
明日からでもすぐに実施できる！

設計者も試験技術者も管理者も
品質保証担当者も
そしてプロジェクト支援者も

みんなで簡単に活用できる！
みんなが共有できる！

みなさんもこの**DBDPアプローチ**を活用し
欠陥流出を**防止**してください。

ご清聴ありがとうございました

Thank you for listening.