
「見積情報伝達シート」による 調査ノウハウの伝達

2012年度ソフトウェア品質管理研究会
(28SQiP)

第6分科会Aグループ

目次

1. 研究動機
2. 現状分析
3. 解決策
4. 解決策の検証
5. まとめ

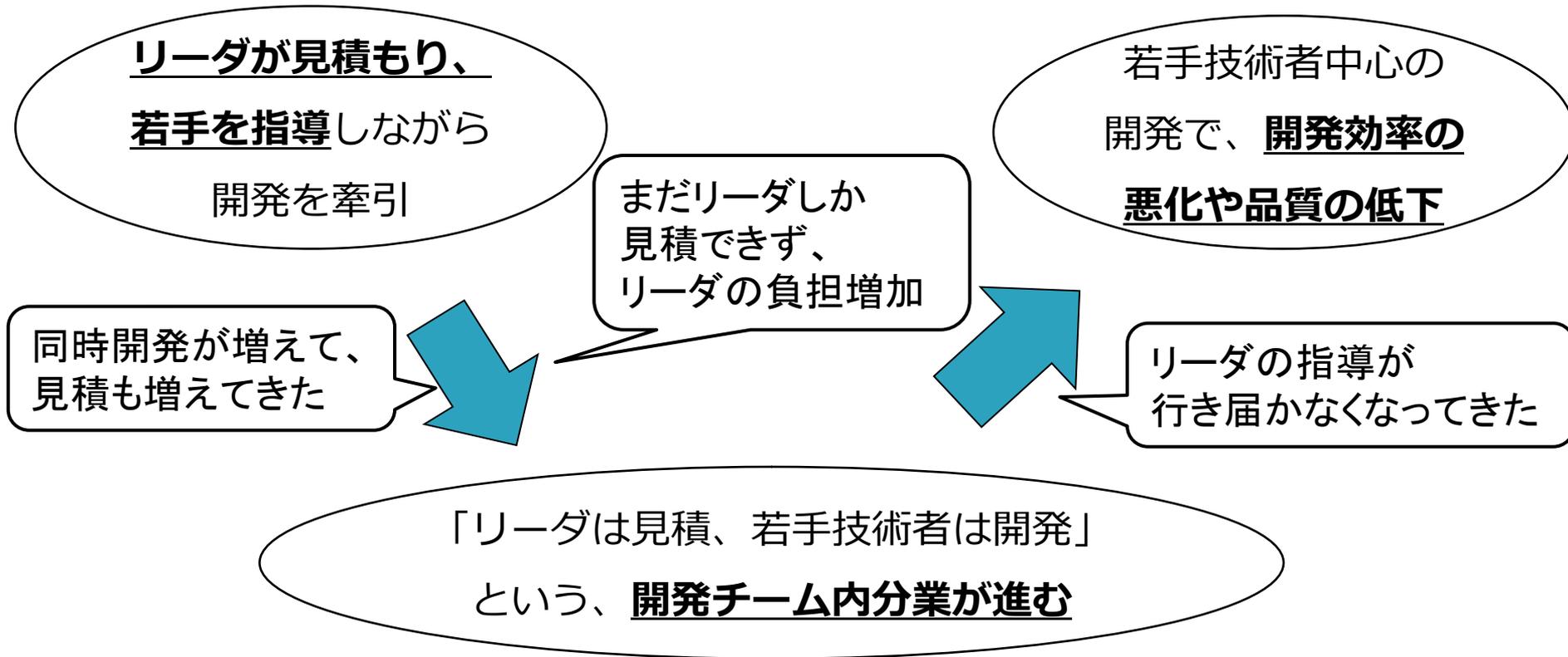
メンバー紹介

主査	清水	吉男	(株式会社システムクリエイツ)
副主査	飯泉	紀子	(株式会社日立ハイテクノロジーズ)
アドバイザー	足立	久美	(株式会社デンソー)

研究員

村嶋	宏太	(株式会社日立ハイテクソリューションズ)
秋山	桂樹	(株式会社リンクレア)
千田	哲義	(NECソフト株式会社)
富島	鉄矢	(AJS株式会社)
南部	妙水	(アンリツエンジニアリング株式会社)
森成	勇一	(アンリツエンジニアリング株式会社)

1. 研究動機：開発チームの移り変わり



**リーダーと若手技術者のコミュニケーション不足により、
開発効率が悪化し、品質が低下している**

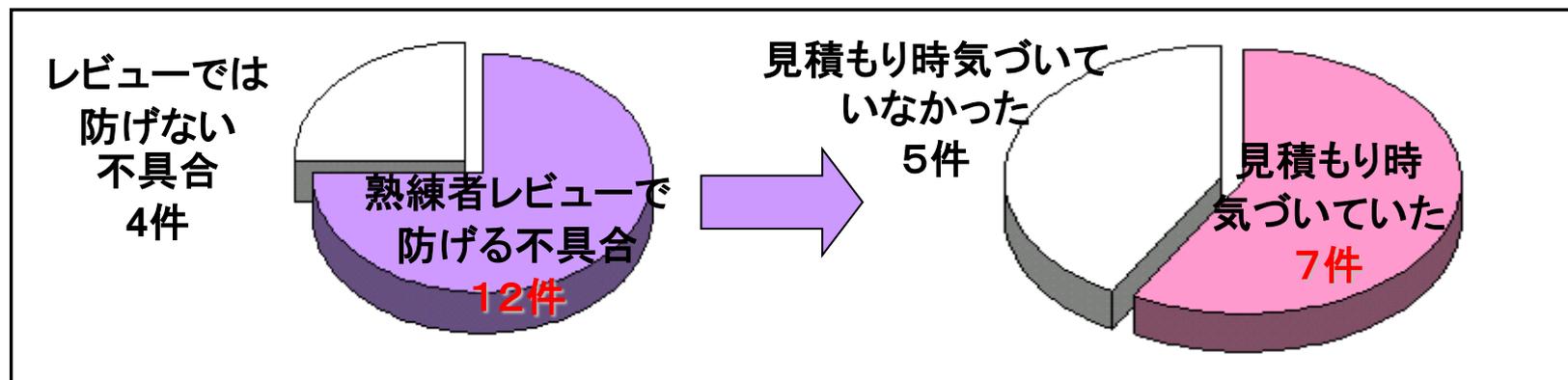
2. 現状分析：不具合分析

熟練者

- ・ 開発者はわかっているだろうという**思い込み**
- ・ ここまで書いておけばやってくれるであろうという**期待**
- ・ どこまで記述すれば良いかわからないまま**記述**

開発者

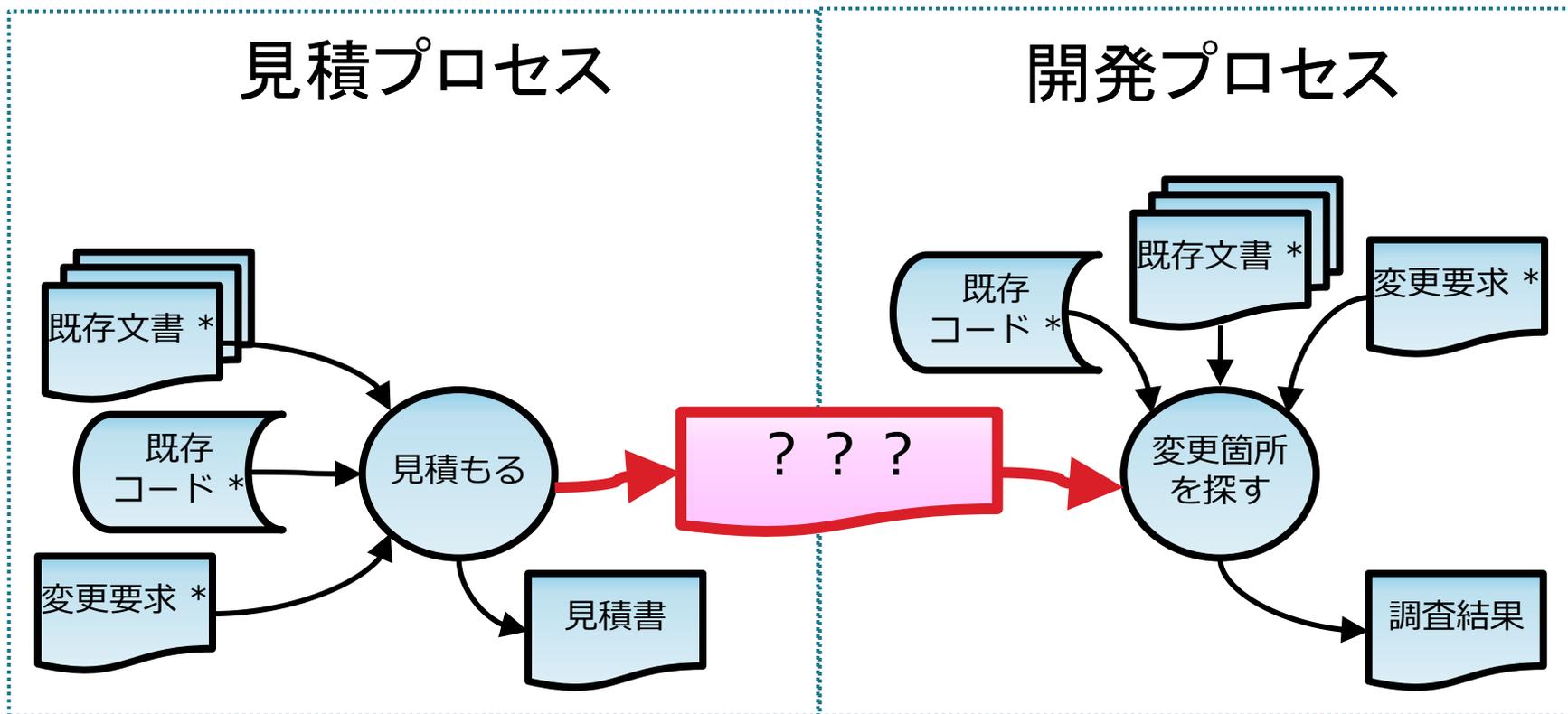
表面的にしか理解していないが、
これで良いだろうという**思い込み**で開発



影響箇所の変更漏れ分析結果

不具合を分析した結果「思い込み」や「期待」により
見積り者から開発者に伝える情報が漏れていた

3. 解決策：見積プロセスと開発プロセスの接続



見積者から開発者に漏れなく伝えてもらうために、何かを書いてもらうことにしたが…

3. 解決策：プロセスをつなぐための課題

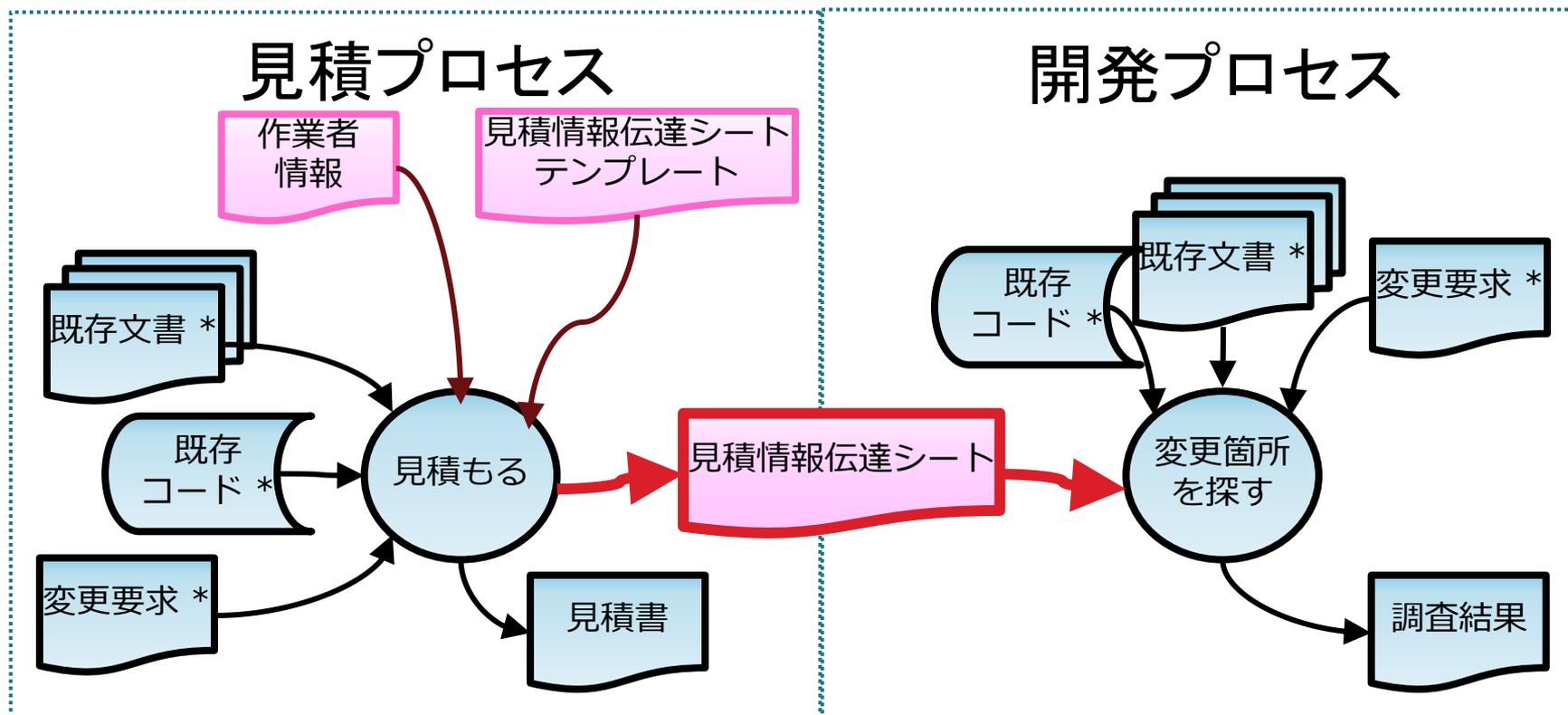
何を、どう書いたら、漏れなく伝わるか？
忙しい見積者に、どうやったら、書いてもらえるか？

開発担当者を意識して
書いてもらおう

表形式のほうが、
漏れがなく書けるし、
見やすいかな

要求の背景などの情報が抜けて、
暗黙知にならないように

3. 解決策：見積情報伝達シートを考案



記載項目を決めて、テンプレートを用意することにした
「見積情報伝達シート」と命名！

3. 解決策：「見積情報伝達シート」の提案

見積情報伝達シート
テンプレート

見積者氏名	
開発者氏名	
変更要求の背景・経緯	

		要求1	要求2	要求3	要求4		
見積情報	規模						
	難易度						
	工数						
各要求に対する 引継情報(気づき、考慮点)							
① 見積ベース・前提	メド仕 ンキ様 トユ・						
	ル仕暗 ー様黙 ル・の						
		要求1	要求2	要求3	要求4	規模	引継情報(気づき、考慮点)
② 作業・変更箇所	作業						
	(大項目) 変更箇所						
③ 変更箇所 詳細	モジュール・ ファイル・ 変更箇所						

表形式にすることにより
記載しやすく、見やすい構成に

- ・ 開発者に変更箇所・影響箇所が伝わる
- ・ 忙しい熟練者でも見積もりながら作成できる
- ・ 熟練者の暗黙知が可視化できる

3. 解決策：「見積情報伝達シート」の提案

見積者氏名	
開発者氏名	

開発担当者を意識して書いてもらうために
見積者と開発者の氏名を記載する欄

変更要求の背景・経緯

		要求1	要求2	要求3	要求4
見積情報	規模				
	難易度				
	工数				
各要求に対する 引継情報(気づき、考慮点)					
① 前 提 ベ ー ス ・	メド仕 ンキ様 トユ・				
	ル仕暗 ー様黙 ル・の				
		要求1	要求2	要求3	要求4
		規模	引継情報(気づき、考慮点)		

要求の背景などの情報が抜けないように
経緯や引継情報を記載する欄

4. 解決策の検証：シミュレーションの実施

実際のプロジェクトで、試してみようとしたが…

うちは機能単位で、
見積もりするから、
この欄はいらないんだけど…

そもそも、
見積りの粒度が違うんですね

うちは、末端のモジュールから、
見積もるから、
上下反対のほうが嬉しいんだけど

4. 解決策の検証：シミュレーションの実施

テーラリングルールを決め、ルールを逸脱しない限りプロジェクトにあわせて自由にカスタマイズ！

★ 開発者氏名は必ず明記すること

★ 「暗黙の」項目が洗い出せる見出しにすること

★ 「引継ぎ情報」欄に気づきを分りやすく書くこと

★ 変更要求を横軸に追加していくこと

★ 推測などミスリーディングを招く記述をしないこと

4. 解決策の検証：シミュレーションの実施

	プロジェクト1	プロジェクト2	プロジェクト3
規模	約 1人月	約0.75人月	約16人月
熟練者の割合	0 %	33 %	0 %
開発者の調査規模	約 1人日	約3人日	約80人日
情報の伝達方法	機能別工数一覧表 口頭の作業指示	機能別工数一覧表	要求と外部規格の対応表
伝達方法の問題点	影響箇所の情報が 欠落	開発者のスキルを考慮 していない	伝達しにくい書き方
発生していた問題	変更漏れによる不具合	変更漏れによる不具合	影響箇所の特定漏れによる作業手戻り

3件のプロジェクトで、以下を検証

- ・ シートの使い勝手
- ・ 開発者の変更漏れを防止できるか
- ・ 熟練者に大きな負担がかからないか

4. 解決策の検証：実際に記入したシート

プロジェクト2での記入した「見積情報伝達シート」(抜粋)

変更要求の背景・経緯

現状では、1ユーザにつき1つの権限しか保持できないが、業務的には、1ユーザが2つ以上の権限を有するため、権限設定時のチェックロジックを変更する必要がある。

要求の背景・経緯を丁寧に記入

変更の内容を記入

変更する際に注意することを記入

見積もリベース・前提		要件1	規模	引継ぎ情報(気づき、考慮点)
変更箇所・作業				
③ 変更箇所詳細	ファイル・モジュール 変更箇所	マスタメンテナンス__A権限登録ソース	ロジック修正	ツールの特性上、共通で実装できないため、各画面ごとに確認
		マスタメンテナンス__B権限登録ソース	ロジック修正	ツールの特性上、共通で実装できないため、各画面ごとに確認
		マスタメンテナンス__C権限登録ソース	ロジック修正	ツールの特性上、共通で実装できないため、各画面ごとに確認
		マスタメンテナンス__D権限登録ソース	ロジック修正	ツールの特性上、共通で実装できないため、各画面ごとに確認

4. 解決策の検証：「見積情報伝達シート」の効果

「見積情報伝達シート」効果についての聞き取り調査結果

	プロジェクト1		プロジェクト2		プロジェクト3	
	見積者	開発者	見積者	開発者	見積者	開発者
変更箇所・影響箇所の書きやすさ/読みやすさ	○	○	○	○	○	○
上記以外の情報の書きやすさ/読みやすさ	○	○	○	○	-	-
1件あたりの書やすさ/読みやすさ	-	-	○	○	×	-
全体の書きやすさ/読みやすさ	○	○	○	○	○	○
発生していた問題を防止できたか	○	○	○	○	○	○

4. 解決策の検証：「見積情報伝達シート」の効果

「見積情報伝達シート」効果についての聞き取り調査結果

	プロジェクト1		プロジェクト2		プロジェクト3	
	見積者	開発者	見積者	開発者	見積者	開発者
変更箇所・影響箇所の書きやすさ/読みやすさ	○	○	○	○	○	○
<ul style="list-style-type: none">・ 開発者の変更漏れを防止することができる・ 文書で伝達できる情報量を大きく増やし、コミュニケーションの不足を補えた・ 見積者は記述方法に、開発者は読み方に悩む必要がない						
全体の書きやすさ/読みやすさ	○	○	○	○	○	○
発生していた問題を防止できたか	○	○	○	○	○	○

4. 解決策の検証：検証結果の考察

開発者の変更漏れを防止できるか

- 見積者と開発者の間で開発に必要な情報をより多く伝達することで、変更漏れを防ぐことができる。

シートの使い勝手はよいか

- 定型フォーマットにより書きやすく、読みやすい。

熟練者に負担はないか

- 熟練者のノウハウがアウトプットされることで問い合わせが減り負担が軽減される。

副次的効果として

- 見積根拠が残り、再見積で利用できる
- 見積の妥当性検証にも活用できる

5.まとめ：研究の成果と今後の課題

研究の成果

「見積情報伝達シート」で

- ① コミュニケーションの不足を補うことができた
- ② 見積時に発見できた影響箇所の変更漏れを防ぐことができた

今後の課題

見積時に発見できない影響箇所の変更漏れ

「見積情報伝達シート」を使用して、
熟練者のノウハウ・経験を若手開発者に伝承し、若手開発者の
スキルアップを促す