

そのレビュー、大丈夫ですか？

～現状レビューの問題発見・解決

株式会社HBA 安達 賢二

[http://www.software-quasol.com/
adachi@hba.co.jp](http://www.software-quasol.com/adachi@hba.co.jp)

安達 賢二（あだち けんじ） adachi@hba.co.jp

株式会社HBA 経営管理本部 共創推進グループ

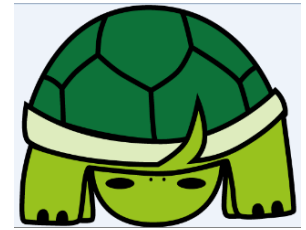
<http://www.software-quasol.com/>

【経歴】

1987年北海道ビジネスオートメーション（現HBA）入社
システム保守・運用・開発業務、プロジェクトマネージャなどを経験後、部門品質保証担当、システム監査委員、全社品質保証担当、全社品質・セキュリティ・環境管理統括責任者、全社生産革新活動SLIM（スリム）技術統括コーディネータなどを担当。
2012年社内イントレプレナー第一号事業者として品質向上支援事業を立ち上げ。
現在、自律運営チーム構築・変革メソッドSaPIDをベースに、関係者と一緒に価値あるコトを創る共創ファシリテータとして活動中。

【社外活動】

NPO法人 ソフトウェアテスト技術振興協会（ASTER）理事
JSTQB（テスト技術者資格認定）技術委員
JaSST（ソフトウェアテストシンポジウム）北海道実行委員
JaSST-Review（ソフトウェアレビューシンポジウム）実行委員
SEA（ソフトウェア技術者協会）北海道支部
SS（ソフトウェア・シンポジウム）プログラム委員
第33-35期SQiP研究会レビュー分科会アドバイザー
SQuBOK_Ver3プロセス改善研究Grリーダー（プロセス改善の黒歴史研究）
TEF（Test Engineer's Forum）北海道テスト勉強会（TEF道）お世話係
IPA（独立行政法人 情報処理推進機構）連携委員
TOCFE北海道幽霊メンバー など



ものごとの見かたを変える



きたのしろくま
@kitanosirokuma

本講演の概要

- レビューは、欠陥検出・予防、生産性向上などへの有効な活動として多くの方たちに認識されています。しかし、レビューをする時間がない、責められるのでイヤだ、時間ばかりかかる、効果が実感できない、などの声が多いのも事実です。
- 本講演では、受講者のみなさんにも一部参加してもらい、現状のレビューの問題点と解決の方向性を持ち帰ってもらえるよう、相互に対話しながら進めるような内容にします。

この講演で目指すゴール

受講者の皆さんが

- 周囲も同じようにレビューに問題を感じていると気づく。
- 自ら捉えるべきレビューの問題点を認識する。
- 改善を進める際の注意事項を理解している。
- これなら自分にもできそう！と思っている。
- 直近レビューで実際に〇〇を試してみよう！と思っている。

コンテンツ

- (1)レビューの現状
 - レビューってどうなってるの？
- (2)レビューで効果を高めるために
 - ではどうすればいいの？
- (3)改善を効果的に進めるために
 - カイゼンってどうやればうまくいく？
- (4)講演のふりかえりとまとめ
 - 今日は何の話だったかな？

最後に 当講演のふりかえり があります！

これは大事・試してみたいと感じたこと	具体的にどのようにするとよいか？ (どのタイミングで何をするか？何で結果・成果を把握すればよいか？など)
大事・試したいと思ったコトに関する現状	

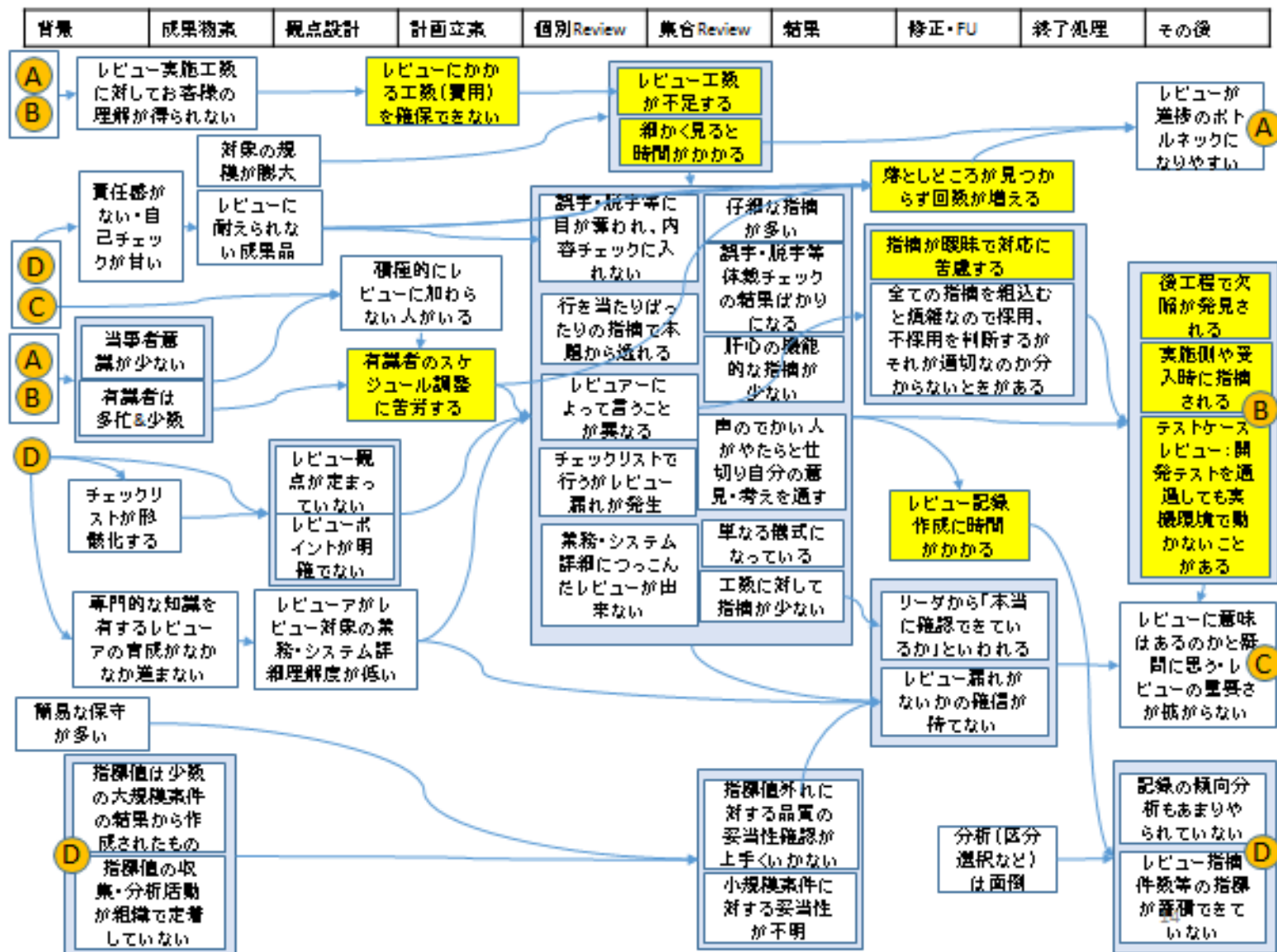
(1)レビューの現状

レビューってどうなってるの？

事前依頼事項：アンケート

- 受講者（予定者）にアンケートを実施。
- 現在のレビューの問題点や困り事を教えてください。
- 何を対象にしたレビューのことなのかを付記してもらう。
- 例：設計書レビューで軽微な指摘しか出ない→あとになってから重大な設計上の問題が発覚する
- 一人：選りすぐり3件まで。
- 開催までに結果を収集して事前にお送りいただく。

レビューの困り事と問題構造



ダメな

レビュー運営シナリオ



- レビューが依頼されたら実施する。
- 時間や余裕がなくなると必要なレビューを端折ってしまう。
- レビュー実施時に対象成果物(のみ)が配付される。
- レビューアがその場で思い通りの観点でレビューを実施する。
- レビューアの経験則や暗黙知、スキルなどにより指摘内容がばらつく。

例: 全体的には表面的なもの(主に記載内容の不備や誤り、不明点など)が多く、腕の良いレビューアが担当する場合などのレアケースで鋭い指摘が混在する。

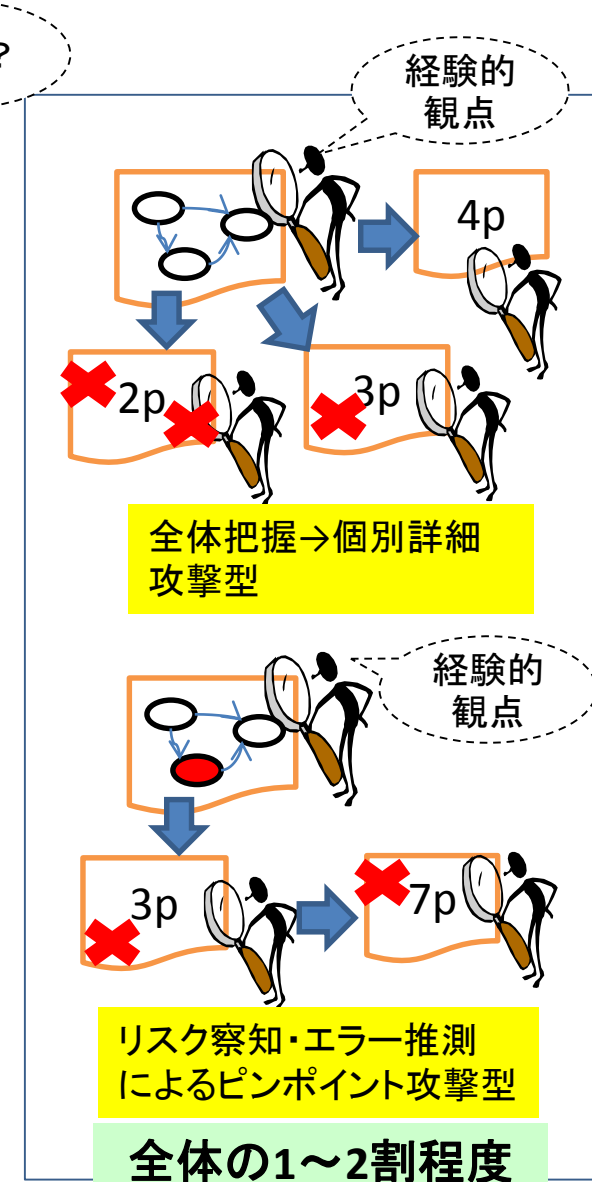
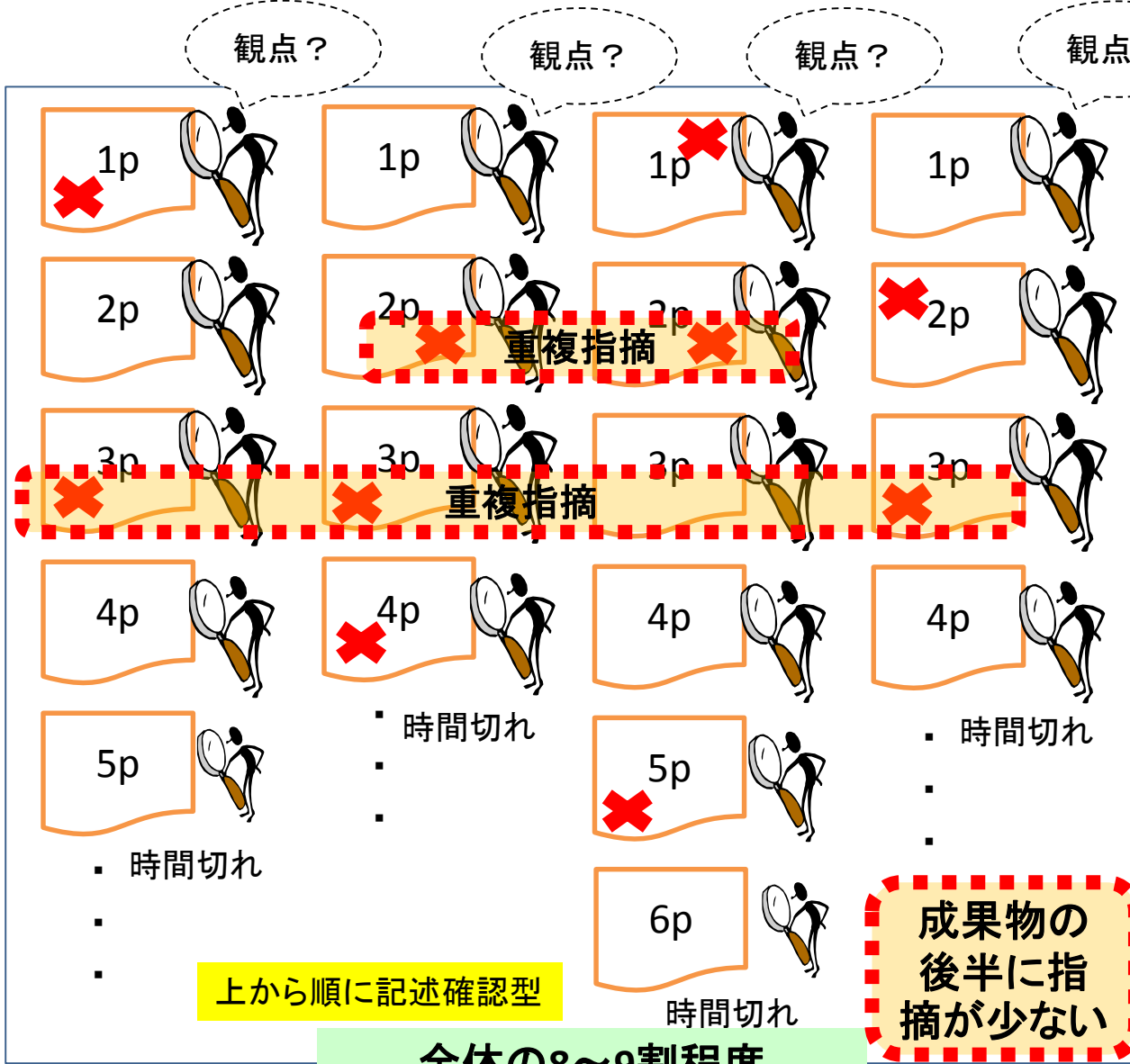
- レビューに使用できる時間により、指摘数や指摘内容が変化する。

例: 時間が少ないほど指摘数や重要な指摘事項が減少し、記述の先頭から順に確認する傾向が高いため、成果物の後半に欠陥が残存する可能性が高まる。

- かけた時間・工数に対して有効な指摘事項が少なくなる傾向が高い。
- 役割分担などを決めずにレビューを行う(ことが多い)ため、探索領域、および観点・指摘内容の重複、抜けが発生する。
- レビューの結果(記録)が残らない。
- フォローアップは実施されない。
- レビューに関わる要員はレビューの効果が実感できず、以降のレビューへのモチベーションが低くなる。



アドホックレビューのアプローチパターン



レビュー結果分析表と記載例

△●:指摘事項			見逃した場合の発見可能Phase(想定)				計	
			実装・UT	IT	ST・OT	C/O後		
			1	3	5	7		
検出効果	効果大 例: 要件抜け・誤り	5					35 ●	
	効果中	3						
	効果小 例: 誤字・脱字・衍字	1	1 △					
計								

UT=Unit Test
IT=Integration Test
ST=System Test
OT=Operation Test
C/O=Cutover

UT=Unit Test
IT=Integration Test
ST=System Test
OT=Operation Test
C/O=Cutover

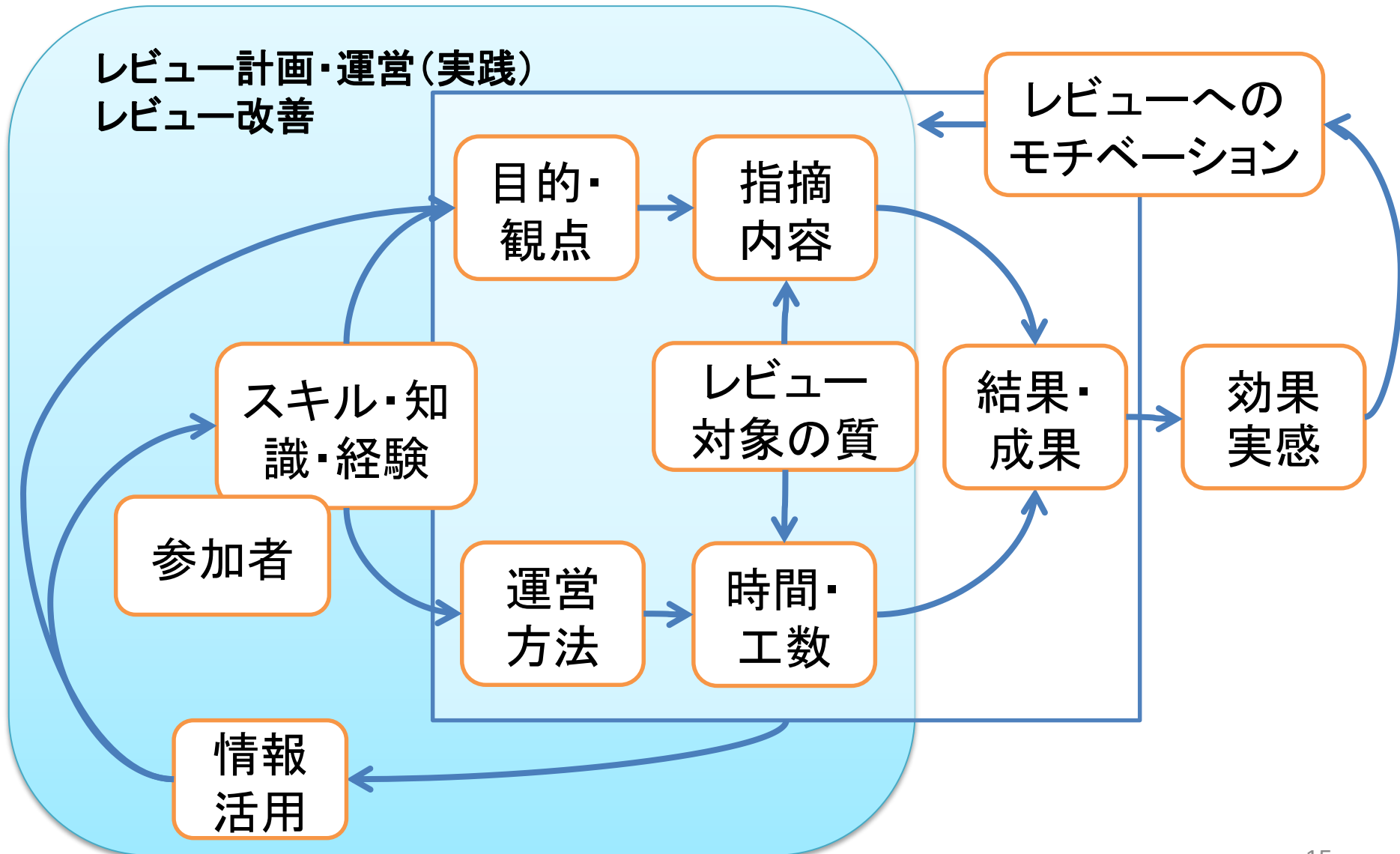
ぼんやりレビューするようになる(例)

△指摘事項 23件			発見可能Phase(想定)				計
			実装・UT	IT	ST・OT	C/O後	
			1	3	5	7	
検出効果	効果大 主対象: 要件抜け・誤り	5	△		△		30
	効果中 主対象: 機能上のバグ (誤植による)	3	△△△△△ △△△△△	△△△△△	△	△	75
	効果小 主対象: 誤字・脱字・衍字 規約違反	1	△△△△△ △△△△△ △△△△△ △			△	16
計			35	18	40	28	121

レビューの視野や深みを小さくする要因の例

- セルフチェックがない粗雑な対象をいきなりレビューする。
 - 表面的な指摘に終始してしまう
- レビュー対象だけを見てレビューする。
 - 書かれていることだけに反応してしまう
- 既成の汎用チェックリストに沿うだけのレビューを行う。
 - 重要度の高低を無視して載せられた項目を平坦に確認する
- 一度に大量の冊子をレビューする。何時間(例:6h)もかけてレビューを行う。
 - “こなす”レビューに。集中できた箇所以外はスッカスカになりうる
- 大勢で同じ対象を(ぼんやり)チェックする。
 - 誰かが見るから私くらい見なくてもダイジョウブ症候群に陥り、見逃しが多くなる／探索領域と指摘事項の重複(ムダ)も多くなる

レビューパフォーマンス影響要因と構造



(2)レビューで効果を高めるために
ではどうすればいいの？

レビューで効果を高めるために

【対応方針1】 できるだけ(無駄な)レビューをしない

- － レビューすべきものだけをレビューする
 - ・ レビュー以前の問題は作成者の責任で解決
 - ・ 限られたリソースで最大の効果を発揮するレビュー対象を特定する
- － 集合形式のレビュー時間を最小化する
 - ・ 個別レビューで8割方の決着をつける
- － 一度のレビュー規模を小さくする
 - ・ 対象作成作業とレビューを連携させる
 - ・ 事前すり合わせ、試作→レビュー&早めのフィードバックを繰り返して完成させる

【対応方針2】 ぼんやりレビューを行わない

- － 目的をより効果的に達成するレビュータイプ等を選択する
- － 必要なレビュー観点等を明確化＝狙いを定めてレビューを行う
- － 作成状況や成果物の特徴等から兆候を察知し、仮説を立ててアプローチを決める
- － 人間特性を考慮して運営を設計する

【対応方針3】 レビューをやりっぱなしにしない

- － 完了判定とフォローアップによりリスクを識別し、修正を確実に完了させる
- － ふりかえりにより改善事項を明確にする

レビューの有効性を支える要因

業務に
よらない
レビュー
技術

業務固有
ノウハウ

運営方法

場／相互理解・協調

レビューで効果を高めるために

【対応方針1】 できるだけ(無駄な)レビューをしない

- － レビューすべきもののだけをレビューする
 - ・ レビュー以前の問題は作成者の責任で解決
 - ・ 限られたリソースで最大の効果を発揮するレビュー対象を特定する
- － 集合形式のレビュー時間を最小化する
 - ・ 個別レビューで8割方の決着をつける
- － 一度のレビュー規模を小さくする
 - ・ 対象作成作業とレビューを連携させる
 - ・ 事前すり合わせ、試作→レビュー&早めのフィードバックを繰り返して完成させる

【対応方針2】 ぼんやりレビューを行わない

- － 目的をより効果的に達成するレビュータイプ等を選択する
- － 必要なレビュー観点等を明確化＝狙いを定めてレビューを行う
- － 作成状況や成果物の特徴等から兆候を察知し、仮説を立ててアプローチを決める
- － 人間特性を考慮して運営を設計する

【対応方針3】 レビューをやりっぱなしにしない

- － 完了判定とフォローアップによりリスクを識別し、修正を確実に完了させる
- － ふりかえりにより改善事項を明確にする

対象の質がレビューに与える影響

間違い探しレビュー実験結果
入力情報は等価(同一間違い10件混入)
【複雑・ランダムな情報羅列】

【A: 複雑】VS【B: 構造化】

A
情報

※記載順、文字・写真の位置、文字の大小は間違いに含めません

A

間違いの数	時間
個	分 秒

TEF (Test Engineer's Forum) 北海道テスト勉強会世話係 など 【研究論文や書籍】
の他 SPI Japan2007/2011/2012 (最優秀賞) /2013 (実行委員長賞) SPES2012 (Best Presentation
賞) /2013、SQIP2011-SIG7・2013-SIG選考テスト設計コンテスト2012・2013 (準決勝)、派生開
発カンファレンス2013、SS2013 (最優秀発表賞) SEC BOOKS『プロセス改善ナビゲーションガイド』
へなげなご(2007.3) 〜プロセス刷新活用編〜 (2007.4) 〜虎の巻編〜 (2009.2) 〜自衛改善編〜
(2013.3) 以上、株式会社 情報処理推進機構 ソフトウェア・エンジニアリング・センター 共著
ソフトウェアプロセス改善手法SAPID入門 日科技術出版社 (2014.3) 1987年HBA入社システム保守、
運用・開発業務を経験後、部門品質保証担当、プロセス監査委員、全社品質保証担当、全社品質、情報セ
キュリティ、環境管理統括責任者、全社生産革新活動スリム技術リーダーなどを担当。2012年社内アント
レプレーター第一号専任者として品質向上支援コンサル事業を立ち上げ【その他社外活動】NPO法人 ソフト
ウェアテスト技術振興協会 (ASTAR) 理事、【経歴】安達 真二 (あたちけんじ) JSTQB (テスト技術者資
格認定) 技術委員、JaSST北海道実行委員、JCT1/SC7/WG24 (Very Small Entities) エキスパート、ソ
フトウェア・シンポジウム (SS) プログラム委員、SPINA3CH User Group運営メンバー、「レビュープ
ロセスの現実的な改善手段の提案」：ソフトウェアテストシンポジウム2006札幌 株式会社HBA Quality
Solution Service SWCSQAアジア地域プログラム委員、派生開発協議会正会員、エキスパート (部長職)
日本科学技術連盟 SQIP/ソフトウェア品質委員会 委員、

Copyright © Kenji Adachi/HBA Co-creation & collaboration Promotion Group, All Rights Reserved

間違い探し
レビュー

平均検出率
(%)

平均時間
(分)

73

8.0

<レビュールール>
各自すべての情報の確
認を終了したら経過時
間と検出した間違い数
を記録する

【B: 構造化】の方が

- ・「71%の時間」で
- ・「検出率が18point 高い」

【構造化・意味ある分類】

B
情報

※記載順、文字・写真の位置、文字の大小は間違いに含めません

B

間違いの数	時間
個	分 秒

安達真二 (あたちけんじ) 株式会社HBA Quality Solution Service エキスパート (部長職)
【経歴】1987年HBA入社
システム保守・運用・開発業務を経験後、部門品質保証担当、プロセス監査委員、
全社品質保証担当、全社品質、情報セキュリティ、環境管理統括責任者、
全社生産革新活動スリム技術リーダーなどを担当。
2012年社内アントレプレーター第一号専任者として品質向上支援コンサル事業を立ち上げ

【研究論文や書籍】
「レビュープロセスの現実的な改善手段の提案」：ソフトウェアテストシンポジウム2006札幌 の他
SPI Japan2007/2011/2012 (最優秀賞) /2013 (実行委員長賞) SPES2012 (Best Presentation
賞) /2013、SQIP2011-SIG7・2013-SIG選考
テスト設計コンテスト2012・2013 (準決勝)、派生開発カンファレンス2013、SS2013 (最優秀発表
賞) SEC BOOKS『プロセス改善ナビゲーションガイド』へなげなご(2007.3) 〜自衛改善編〜
(2013.3) 以上、株式会社 情報処理推進機構 ソフトウェア・エンジニアリング・センター 共著
ソフトウェアプロセス改善手法SAPID入門 日科技術出版社 (2014.3)

【その他社外活動】
NPO法人 ソフトウェアテスト技術振興協会 (ASTAR) 理事、JSTQB (テスト技術者資格認定) 技術委員、
JaSST北海道実行委員、日本科学技術連盟 SQIP/ソフトウェア品質委員会 委員、
JCT1/SC7/WG24 (Very Small Entities) エキスパート、ソフトウェア・シンポジウム (SS) プログラム
委員、SPINA3CH User Group運営メンバー、SWCSQAアジア地域プログラム委員、派生開発協議会正会員、
TEF (Test Engineer's Forum) 北海道テスト勉強会世話係 など

Copyright © Kenji Adachi/HBA Co-creation & collaboration Promotion Group, All Rights Reserved

間違い探し
レビュー

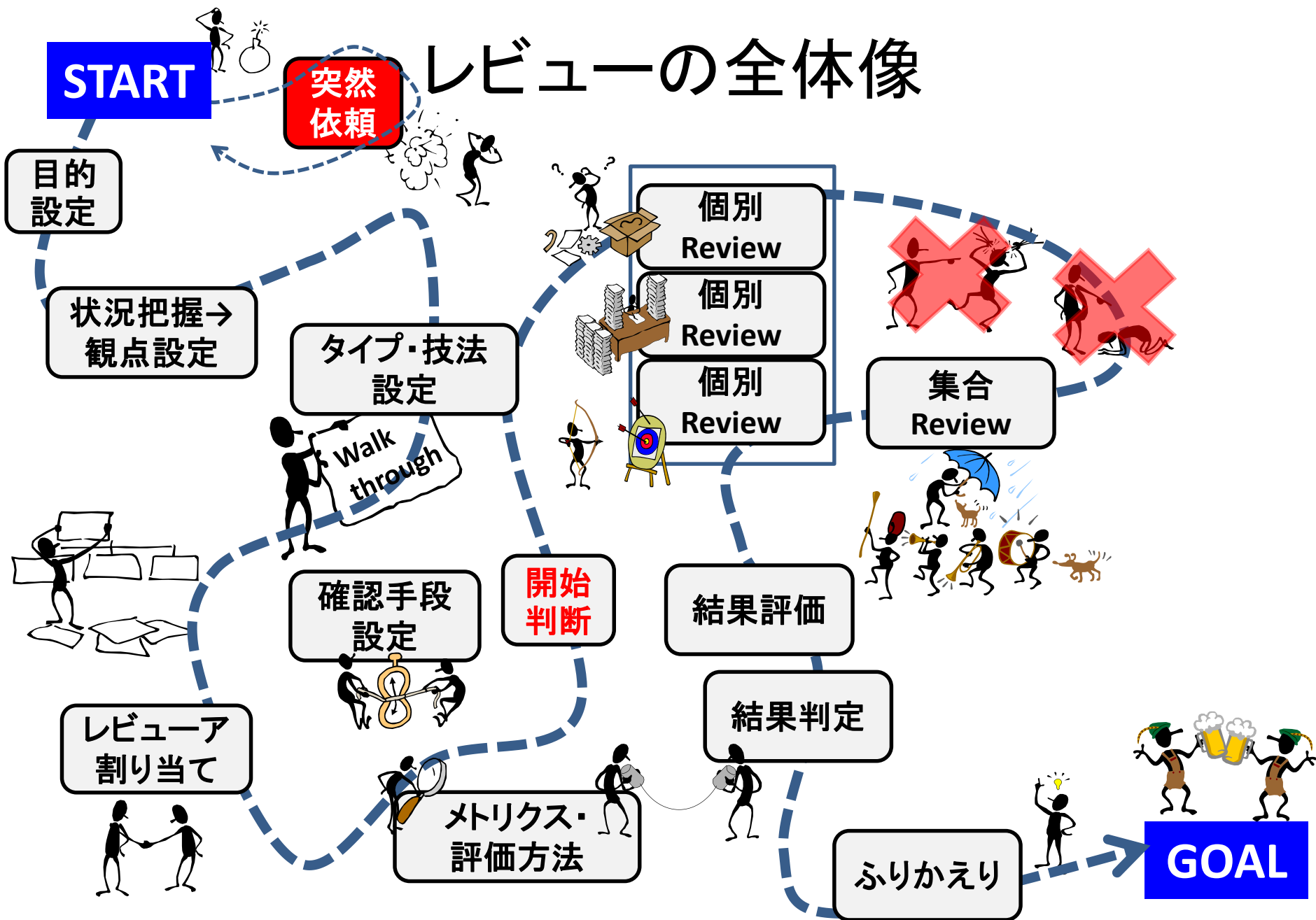
平均検出率
(%)

平均時間
(分)

91

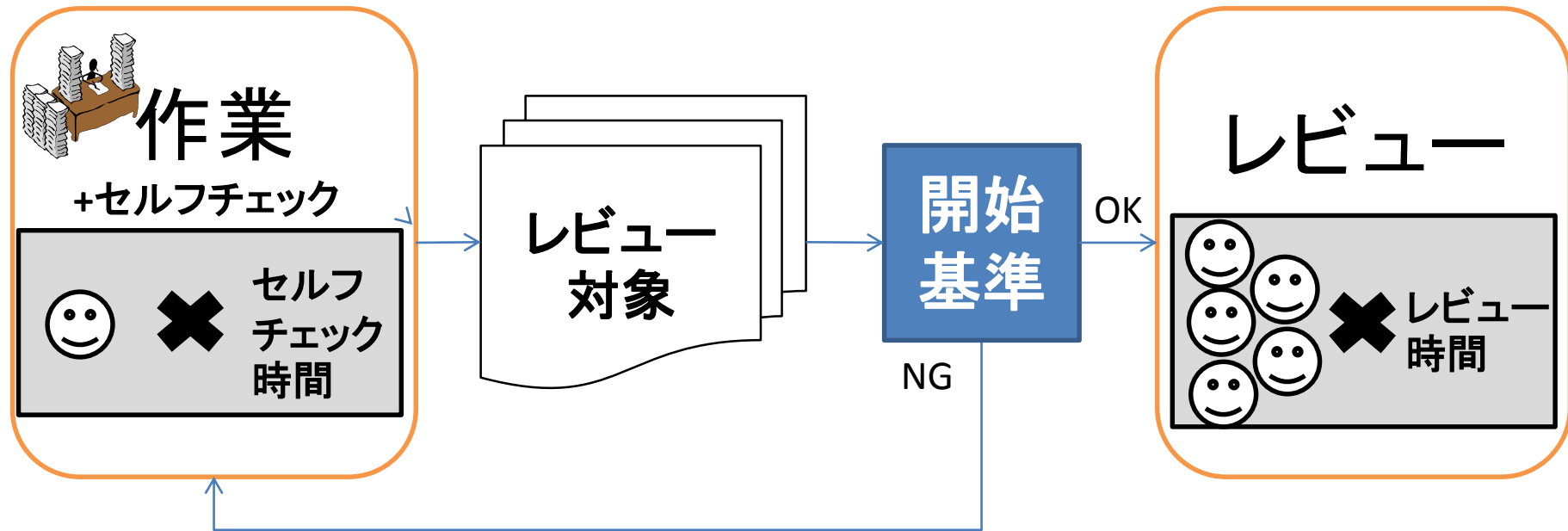
5.6

レビューの全体像



レビューすべきものだけをレビューする

レビューするに値する対象をレビューする



◎効果1 作業者としての最低限の責任（例：セルフチェックの励行）や質の良い作業を動機づける。

◎効果2 効果の低い欠陥や不備の確認→検出などに時間・工数を浪費することを防ぐ。

レビュー開始基準・判定

よくある開始基準の構成要素例

- セルフチェック済みである。
- バージョン識別子が付与されている。
- パッと見で重大な欠陥や軽微なバグ群が存在しない。
- 未解決課題と解決の目途がすべて明示されている。
- 入力、参照情報がすべて参照可能である。
- 先行作業成果物の品質レベルが確認されている。
- レビュー目的が理解され、共有されている。
- 作成者が自ら確認して欲しい観点を提示している。

**開始基準とは、レビュー実施に必要な条件や
準備しておいて欲しい事項**

限られたリソースで最大の効果を発揮するレビュー対象を特定する

どれが重要（レビュー必須）？

成果物	A	B	C	D	E
機能重要度低1-5高	1	1	3	1	5
難易度	低	高	中	低	超高
規模（Page）	5P	13P	25P	3P	7P
作成日数	1.5日	4日	8日	1日	7日
担当者開発経験	10年	3年	3年	1年	8年
作業中変更受付数	0件	5件	1件	2件	1件

集合形式のレビュー時間を最小化する

とある集合レビューの10分間

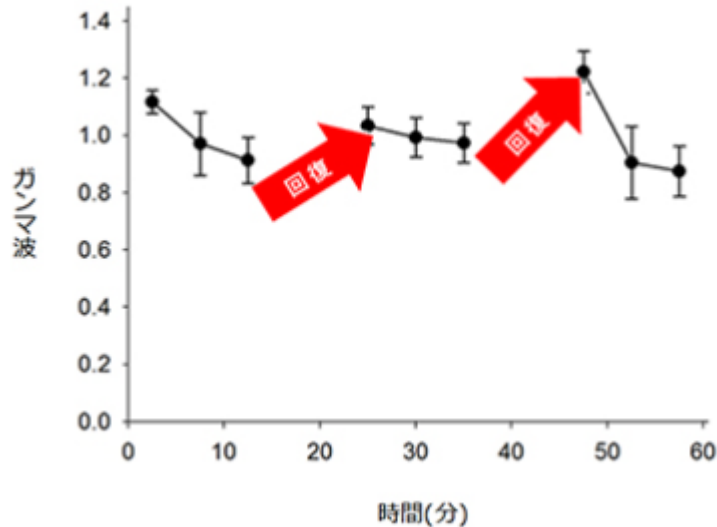
		1分	1分	1分	1分	1分	1分	1分	1分	1分		
レビューアA 		質問・コメント				コメン ト				質問・コメント		5人分
レビューアB 			コメン ト				質問・ コメン ト					2人分
レビューアC 						コメン ト						1人分
レビューアD 				質問				コメン ト				2人分

総工数: 4人 × 10分 = 40人分 実稼働工数: 1分稼働 × 10 = 10人分

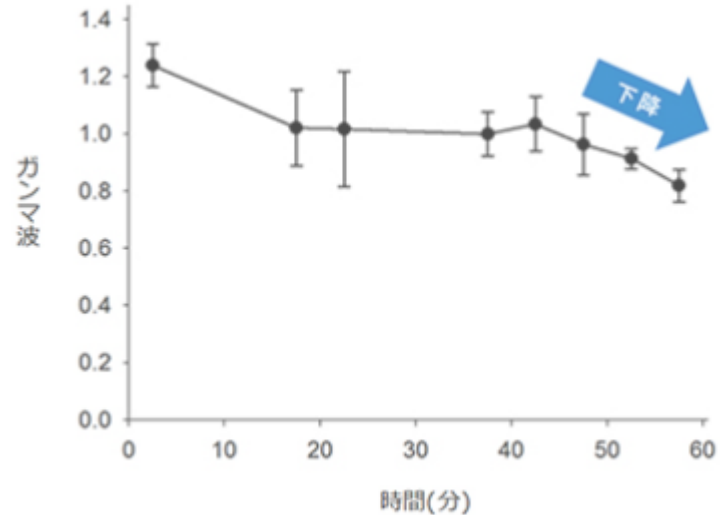
稼働効率: 実稼働工数 / 総工数 × 100 = **25%**

集合形式のレビュー時間を最小化する／一度のレビュー規模を小さくする

人間の集中力継続時間



15分×3(計45分)学習
グループのガンマ波波形



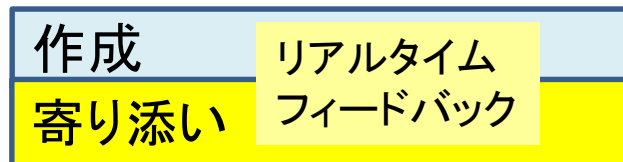
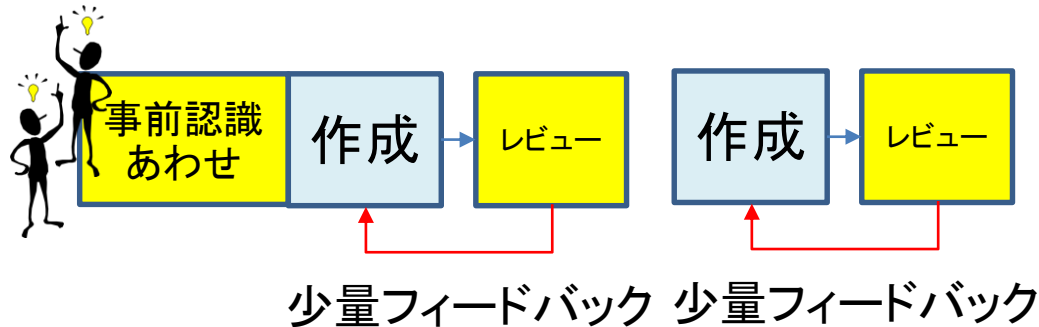
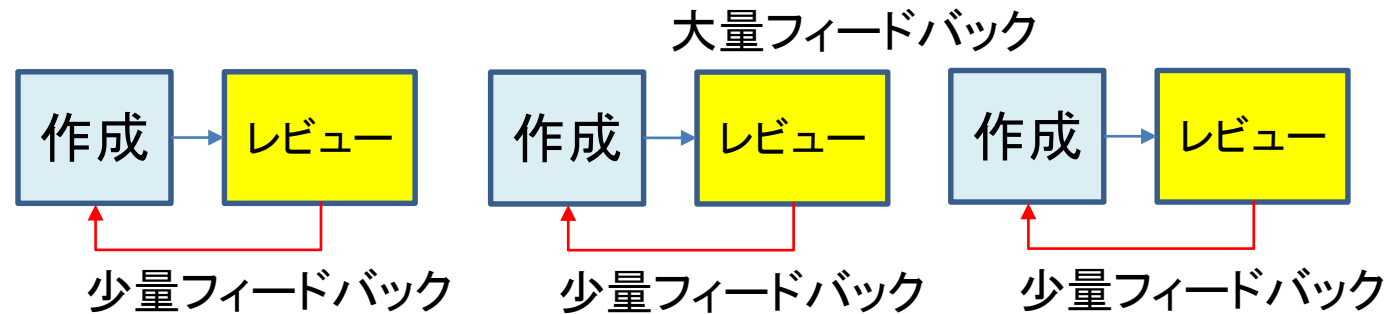
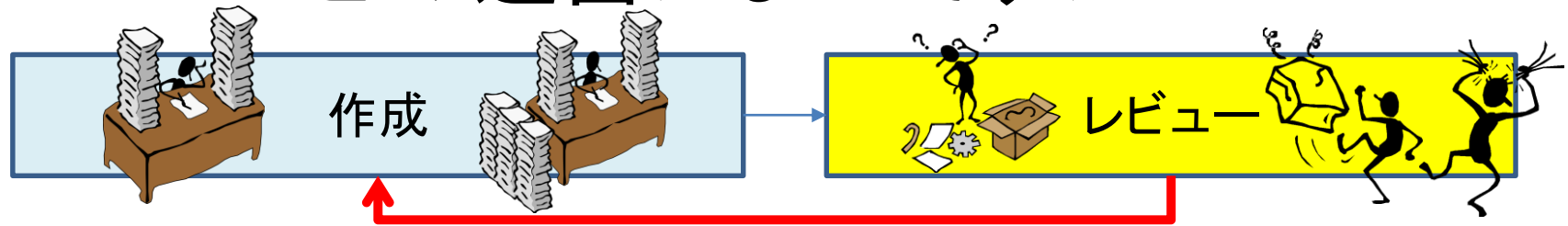
60分学習グループの
ガンマ波波形

※グラフ1と2のガンマ波の絶対値の大小は関係なし

集中力の維持と長期的な学習効果につながる方法
(東京大学・池谷裕二教授の見解)より

http://www.asahi.com/ad/15minutes/article_02.html

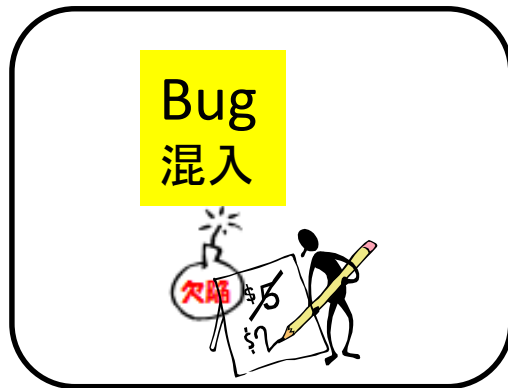
一度のレビュー規模を小さくする どの運営がよいですか？



精神的・身体的・もの
づくり的に最も負荷
が高い/低いもの、効
果的なものはどれで
しょうか？

上流工程でバグを検出するのがよい？

リードタイムが長くなるほど、手戻り工数が膨らみ、生産性が低下し、品質確保が困難に



時間の流れ

このリードタイムを最小にすることが品質を高め、手戻りを最小にし、生産性を高める秘訣

※時間が経過しても手戻りが変わらないものもあるので注意

レビューで効果を高めるために

【対応方針1】 できるだけ(無駄な)レビューをしない

- － レビューすべきものだけをレビューする
 - ・ レビュー以前の問題は作成者の責任で解決
 - ・ 限られたリソースで最大の効果を発揮するレビュー対象を特定する
- － 集合形式のレビュー時間を最小化する
 - ・ 個別レビューで8割方の決着をつける
- － 一度のレビュー規模を小さくする
 - ・ 対象作成作業とレビューを連携させる
 - ・ 事前すり合わせ、試作→レビュー&早めのフィードバックを繰り返して完成させる

【対応方針2】 ぼんやりレビューを行わない

- － 目的をより効果的に達成するレビュータイプ等を選択する
- － 必要なレビュー観点等を明確化＝狙いを定めてレビューを行う
- － 作成状況や成果物の特徴等から兆候を察知し、仮説を立ててアプローチを決める
- － 人間特性を考慮して運営を設計する

【対応方針3】 レビューをやりっぱなしにしない

- － 完了判定とフォローアップによりリスクを識別し、修正を確実に完了させる
- － ふりかえりにより改善事項を明確にする

目的をより効果的に達成するレビュータイプ等を選択する レビュータイプと期待効果例

レビュータイプ	概要	期待効果(欠陥除去)
インスペクション	最も厳格で、体系的、公式に実施されるレビュー。	1000行当たり16～17件という報告有り
チームレビュー	軽量化されたインスペクション。 構造化されているがインスペクションほど公式、厳格でない。	1時間当たりの欠陥検出数はインスペクションの2/3という報告有り
ウォークスルー	作成者が主導して実施されるレビュー。 作成者自らが対象成果物内容を説明することで実施する。 記録が取られないことが多いため、バグの検出にそれほど効果的ではない。	1000行当たり8件程度(インスペクションの半分)という報告有り
ピアデスクチェック	ペアレビューとも呼ばれ、作業員以外にはただ一人だけが作業成果物を調べる。最も安価なレビュー方法。	(少ない工数で欠陥検出する手法の一つ)
パスアラウンド	作成者が作業成果物のコピーを複数人に配付し、複数のコメント・フィードバックを獲得する方法。	(比較的時間がある場合や分散開発で有効)
アドホックレビュー	場当たりの、思いつきで実施するレビュー。ある特定の切り口に対する意見・コメントを即興で求める場合にのみ有効。	欠陥除去にはあまり寄与しない

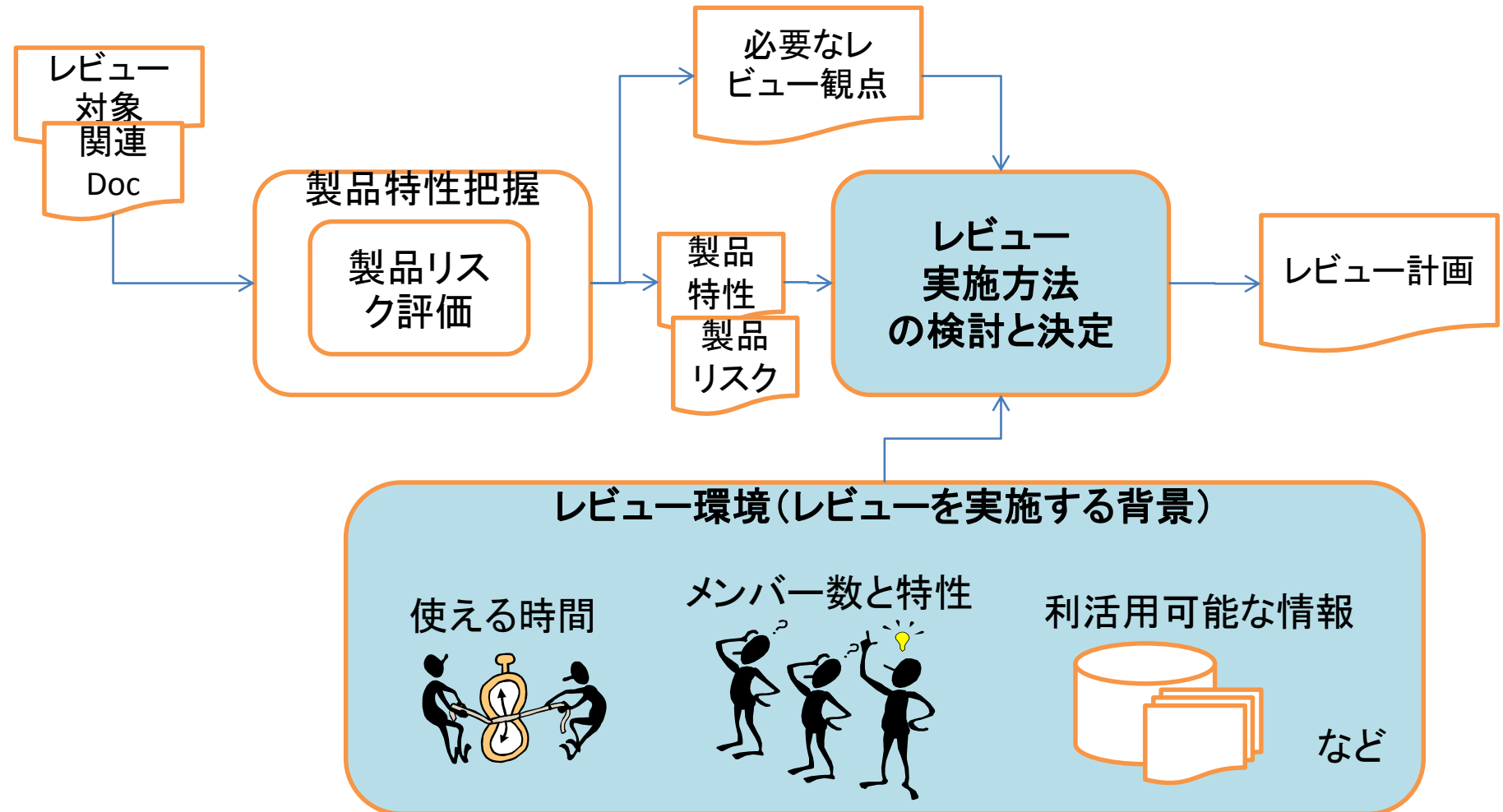
レビュータイプ	特徴(長所・短所)	どのような場合に活用すると効果的か どのような場合に使わないようにすべきか
インスペクション	厳格で、体系的、公式に実施されるレビュー。最も欠陥検出効果が高く、プロセス改善促進も期待できるが、不適切な運営では費用対効果が悪化する。	成功要因:トレーニングされた要員と組織的運営(特にモデレータは重要)、各種実績データに基づく判断等。トレーニング不足なら採用しない方がよい。
チームレビュー	厳格さ抜きでも欠陥検出を期待する場合に効果がある。適切な運営により効果が期待できる。運営ノウハウがないとムダが増え、効果が得られない場合がある。	必要な有識者が参加できる場合、効果が期待できる。力量あるモデレータがいるとスキル不足が補える場合があるが、スキル不足要員ばかりになる場合は避けた方がよい。
ウォークスルー	作成者が主導して対象成果物内容を説明しながら実施する。説明が上手な人が対応する場合、レビュー対象の内容不備が検出されない可能性がある。	力量のある要員が自らの作業結果を確認する場合や、実作業前に作業者の理解度を確認する場合、また関係者に内容を把握しておいてもらいたい場合等に効果が期待できる。
ピアデスクチェック	机上で担当者とレビューア(通常1名)で行う確認。2名の都合が合えばすぐ実施できる。結果がレビューアの力量に大きく左右される。	有識者が育成過程要員の教育を兼ねてレビューア対応を行うのが最も効果的。レビューアの力量が伴わない場合はこの手段を選択しない方がよい。
パスアラウンド	集合せずに複数名のレビューアからコメントを取得できる。リードタイム制約が高い場合や、作業者と回覧先要員の力量が伴わない場合等では効果が少ない。	比較的時間・期間(リードタイム)をかけても構わず、作業者が一定レベル以上のスキルがあつてコメント処理を適切に対応できる場合に活用可能。時間がない場合は避ける。
アドホックレビュー	ピンポイントに活用すると費用対効果が最大になる可能性もある手法。 思いつき確認なので、網羅的な確認に向かない。	特定の確認したい事項を有識者に依頼するなどピンポイント確認での使用を推奨。 網羅的な確認、有識者ではないレビューでは採用しない方がよい。

目的をより効果的に達成するレビュータイプ等を選択する リーディング技法の特徴(実験結果からの考察)

項目/技法	アドホック(Ad-hoc)	チェックリスト(CBR)	シナリオ(SBR)
特徴	・無計画、思いつき対応 ・未確認、見逃しが多くなる可能性大 ・新発想獲得の可能性	・網羅に向くため見逃しを防止できる可能性大 ・項目毎に個別確認することが多く時間がかかる	・利用者、設計者などの視点で、流れで確認する ・上流工程で使用されることが多い
効果決定要因	担当者のスキル、経験に結果が大きく左右される	チェックリストの質と活用方法に結果が左右される	設定するシナリオの質に結果が左右される
コミュニケーションの影響	結果・効果に大きな影響を与える	結果・効果に影響を与える場合がある	結果・効果への影響を受けにくい
強み	担当者が持つノウハウ・直感による確認が可能・時間がかからない	・チェック項目を網羅しやすい ・スキルのばらつき影響を受けにくい	利用者、設計者等様々な視点で、目的達成の可能性に対する確認が可能
弱み	・確認漏れ、重複、表面的確認に終始しがちになる	時間がかかる／形式的確認となる可能性あり／チェック項目以外に目がいなくなる可能性あり／チェックリストの作成、維持が大変	表現形式、スタイルの良し悪しなどの基本的事項の確認が抜けてしまう可能性あり／想定利用者のコンテキストと合わない可能性あり／不整合を見逃す可能性あり
事前準備等考慮事項	スキル・経験ある担当者が対応する	事前にレビュー対象に必要なチェック項目の選定や追加・カスタマイズを行うことが必要	想定利用者とその利用状況を具体化し、利用者の立場に立つて行う
効果的活用法	別途網羅的確認を行いつつ、ピンポイントでこの技法を活用するのが効果的	チェック項目の選定・追加・カスタマイズした上で網羅的確認に活用する	上流工程を中心に、チェックリストと併用することで効果が期待できる

立場で見る「Perspective Based Reading」やユースケースで見る「Usage Based Reading」などがある

目的をより効果的に達成するレビュータイプ等を選択する 製品特性・リスク、レビュー環境等によるレビュー実施方法の採用

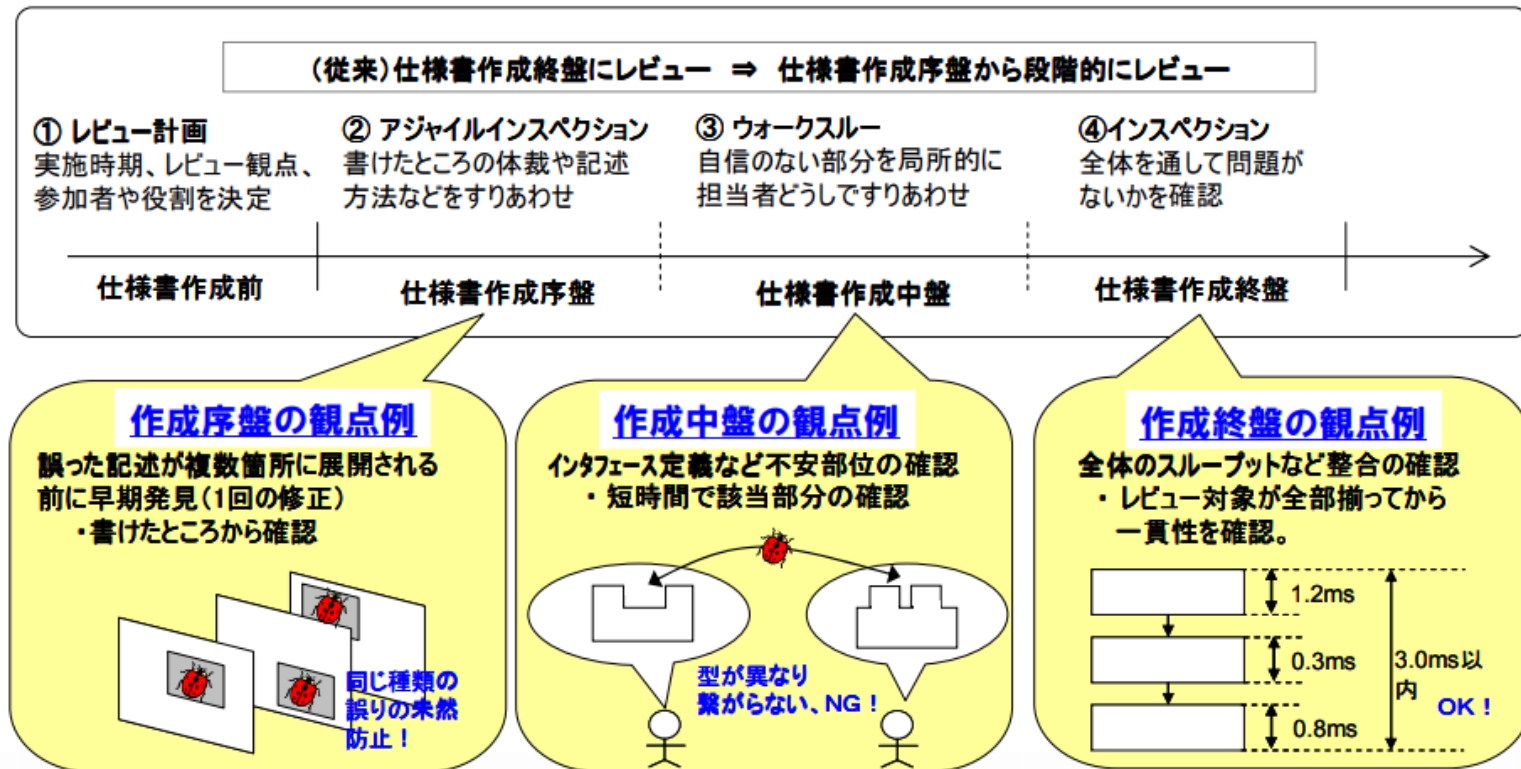


目的をより効果的に達成するレビュータイプ等を選択する 産学連携によるデザインレビュー改善事例

三菱電機 細谷さん他

http://www.jaspic.org/event/2011/SPIJapan/session1B/1B3_ID003.pdf

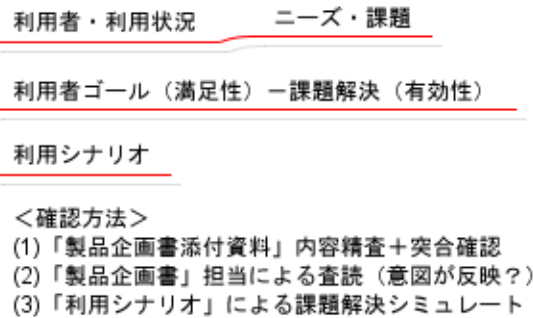
レビュー技法の使い分け



必要なレビュー観点等を明確化＝狙いを定めてレビューを行う マインドマップによるレビュー観点設計事例

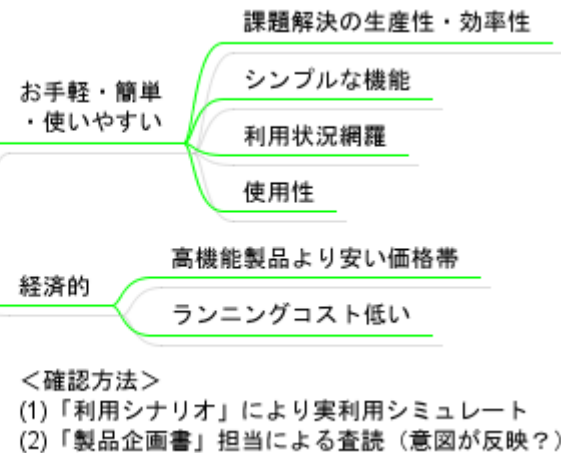
「見どころ」＝レビュー目的
ウリを備えた利用者の課題解決製品として成立する
要件とSPECを備えているかを確認する

【観点①】 利用者ニーズへの適応 課題解決の可能性

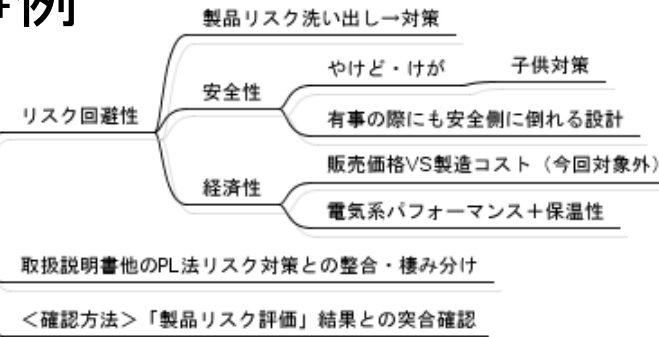


売れるのか？ 役立つのか？

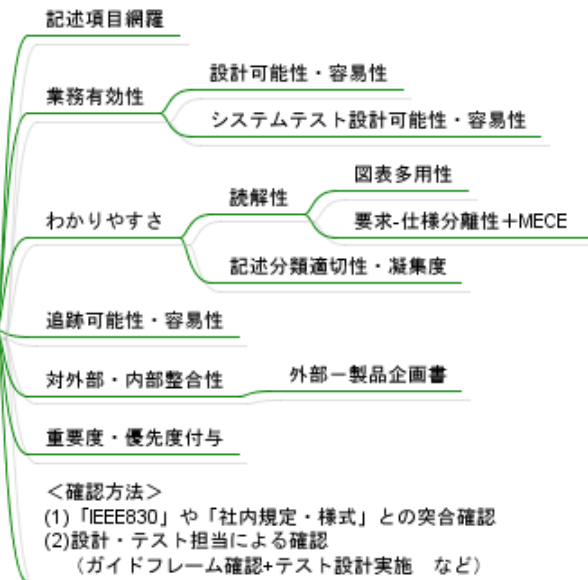
【観点②】 ウリの実現と ターゲットへのヒット



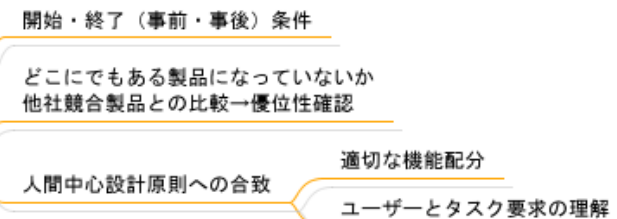
【観点③】 製品リスク対策



【観点④】 要求仕様書適切性



【観点⑤】 弱み・脆弱性



必要なレビュー観点等を明確化＝狙いを定めてレビューを行う

レビュー観点設定

ワークショップ実施組織

- 組込み系メーカ2社 6チーム+6チーム
- Sier 1社 5チーム
- 独立系システムソリューション企業 1社 2チーム
- 組込み系メーカに勤務する協力会社 6社 5チーム

組織ロケーション：東京・横浜・札幌・大分・名古屋・仙台など

分析可能なデータが存在した

10社24チーム

必要なレビュー観点等を明確化＝狙いを定めてレビューを行う

実施結果例

20140618 **A**グループ

④レビュー計画・観点設定票

優先度: ☐

レビュー目的:
ポットの安全性と使いやすさの確認

レビュー観点	確認方法
機能の操作性 正常系・異常系 商品の目的(背景) 仕様の分かりやすさ input/output 誤字・脱字 安全性	要求仕様の項目の照合チェック ハードウェアとソフトウェアの要求仕様の整合性 要求に対するOutput内容チェック あいまいな表現(目的が明確か)対象 機能遷移 エラー検知の確認

(目的)

(観点)

A

① ユーザー視点、使いやすさ	⑥ 仕様の分かりやすさ
② 安全(フェールーフ)	⑦ input/output 内容
③ 何をどう使うのか(商品性)	⑧ 機能遷移
④ 正常系・異常系	⑨ 結果(商品性)
⑤ 商品性	

計画立案過程

⑥レビュー結果分析表

	改善前: Before ↓ 改善後: After	発見可能Phase(想定)				計
		実装・UT	IT	ST・OT	C/O後	
		1	3	5	7	
効果大 主対象: 安全性、機能性	5	●●●●● ○●●●● ●●●●● ●●●●●	●●●●● ○●●●● ●●●●● ●●●●●	●●●●● ○●●●● ●●●●● ●●●●●	●●●●● ○●●●● ●●●●● ●●●●●	
効果中 主対象: 数値の妥当性	3	●●●●● ○●●●● ●●●●● ●●●●●	●●●●● ○●●●● ●●●●● ●●●●●	●●●●● ○●●●● ●●●●● ●●●●●	●●●●● ○●●●● ●●●●● ●●●●●	
効果小 主対象: 誤字・脱字	1	●●●●● ○●●●● ●●●●● ●●●●●	●●●●● ○●●●● ●●●●● ●●●●●	●●●●● ○●●●● ●●●●● ●●●●●	●●●●● ○●●●● ●●●●● ●●●●●	
計						12/19

Copyright © Kenji Adachi@HBA Quasol, All Rights Reserved

必要なレビュー観点等を明確化＝狙いを定めてレビューを行う

実施結果例

20140618 **B**グループ

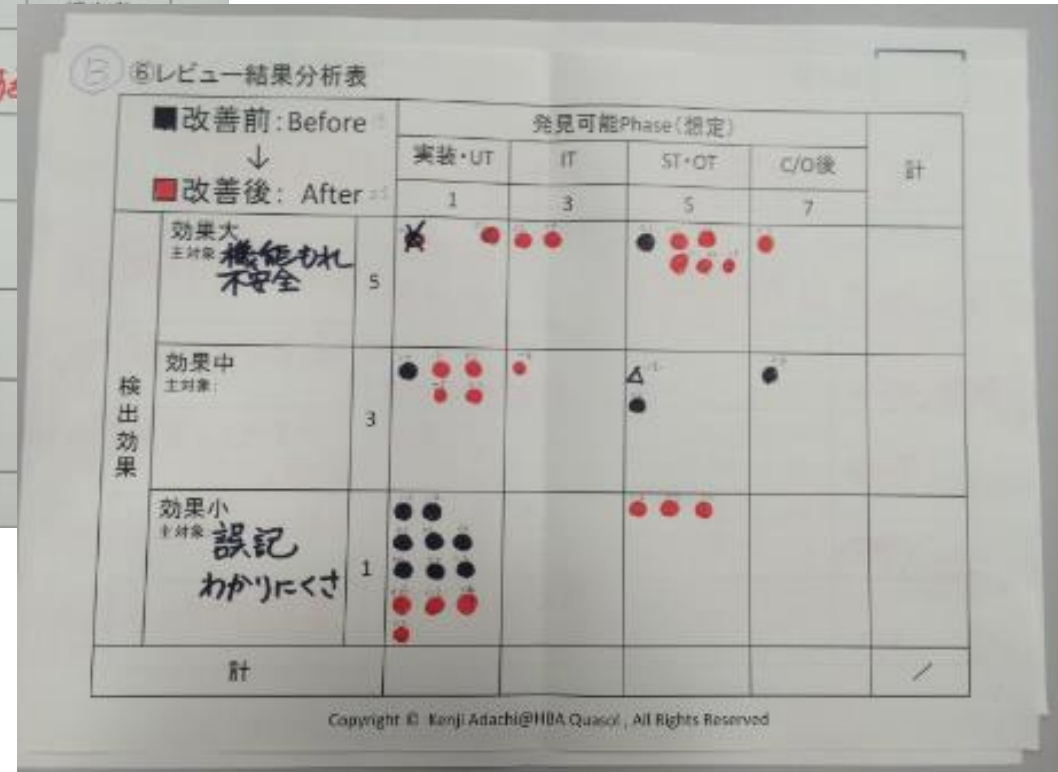
④レビュー計画・観点設定票

優先度:

レビュー目的:
設計お上での必要十分かを確認

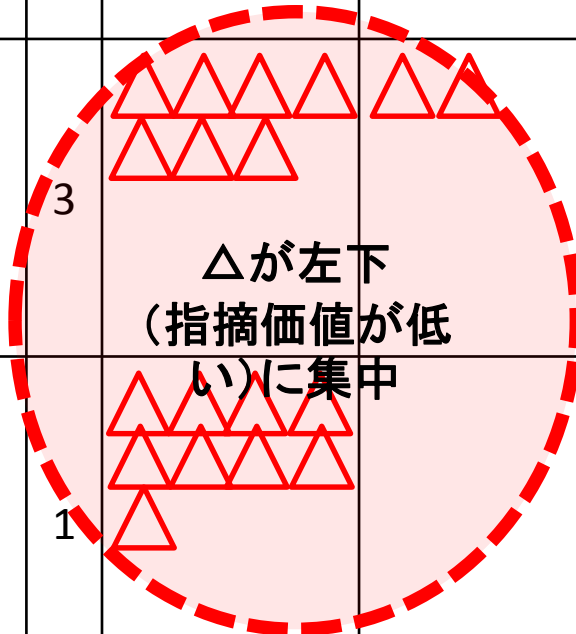
レビュー観点	確認方法
⑥ 設計情報の記述	網羅性 詳細度 の確認
④ 解析性の考慮	トレーサビリティ 故障のわかりやすさ
④ 検査のしやすさ の記述	基準
④ 故障しにくい構造	複雑(使い易) 耐久性
④ わかりやすい文章	誤解のない文章

Copyright © Kenji Adachi@HBA Quasol, All Rights Reserved



ぼんやりレビューするとこうなる(例)

△指摘事項 23件			発見可能Phase(想定)				計
			実装・UT	IT	ST・OT	C/O後	
			1	3	5	7	
検出効果	効果大 主対象: 要件抜け・誤り	5	△		△		30
	効果中 主対象: 機能上のバグ (誤植による)	3	△△△△△ △△△△△	△△△△△	△	△	75
	効果小 主対象: 誤字・脱字・衍字 規約違反	1	△△△△△ △△△△△ △			△	16
計			35	18	40	28	121



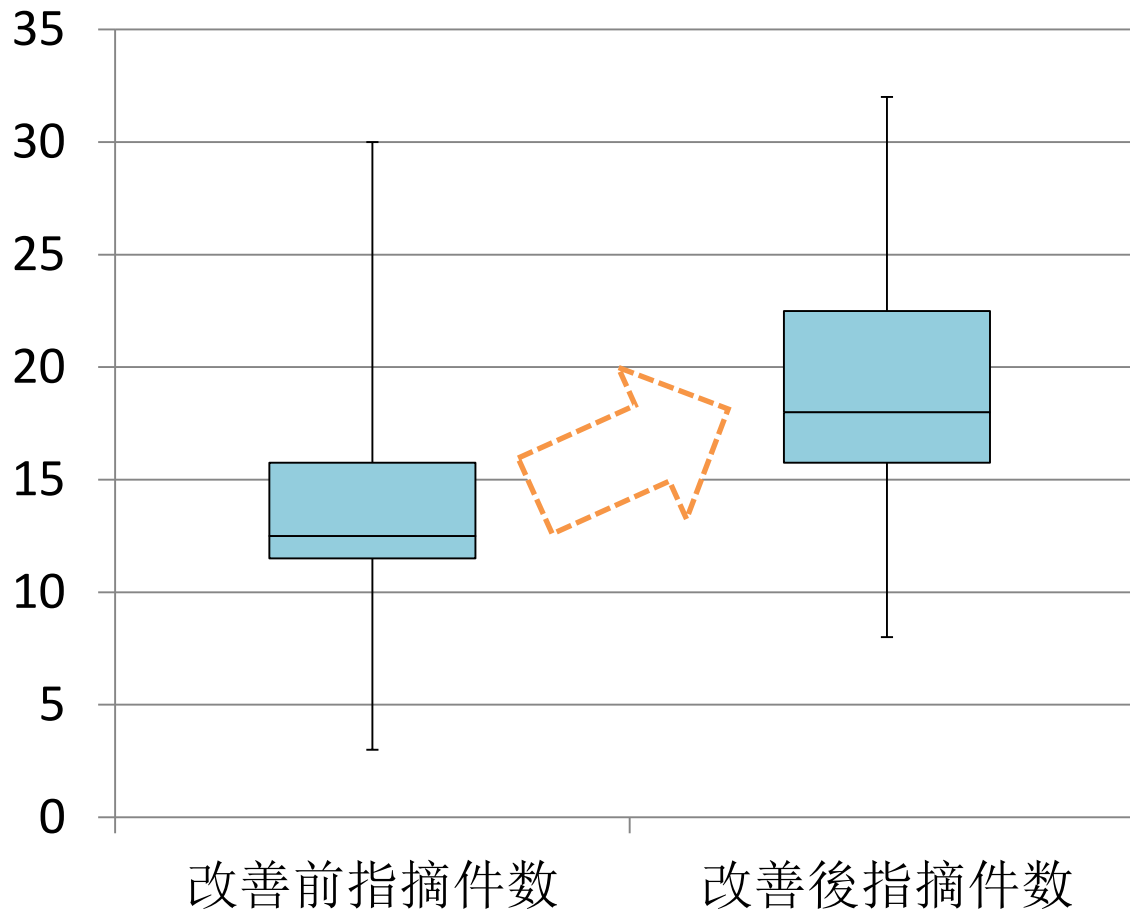
狙い撃ちレビューではこうなる(例)

●指摘事項 24件			発見可能Phase(想定)				計
			実装・UT	IT	ST・OT	C/O後	
			1	3	5	7	
検出効果	効果大 主対象: 要件抜け・誤り	5	●●●				335
	効果中 主対象: 機能上のバグ (誤植による)	3	●				93
	効果小 主対象: 誤字・脱字・衍字 規約違反	1	●			●	8
計			19	0	340	77	436

指摘件数: $\triangle 23 \rightarrow \bullet 24$

Copyright © Kenji Adachi / HBA Co-creation & collaboration Promotion Group , All Rights Reserved 41

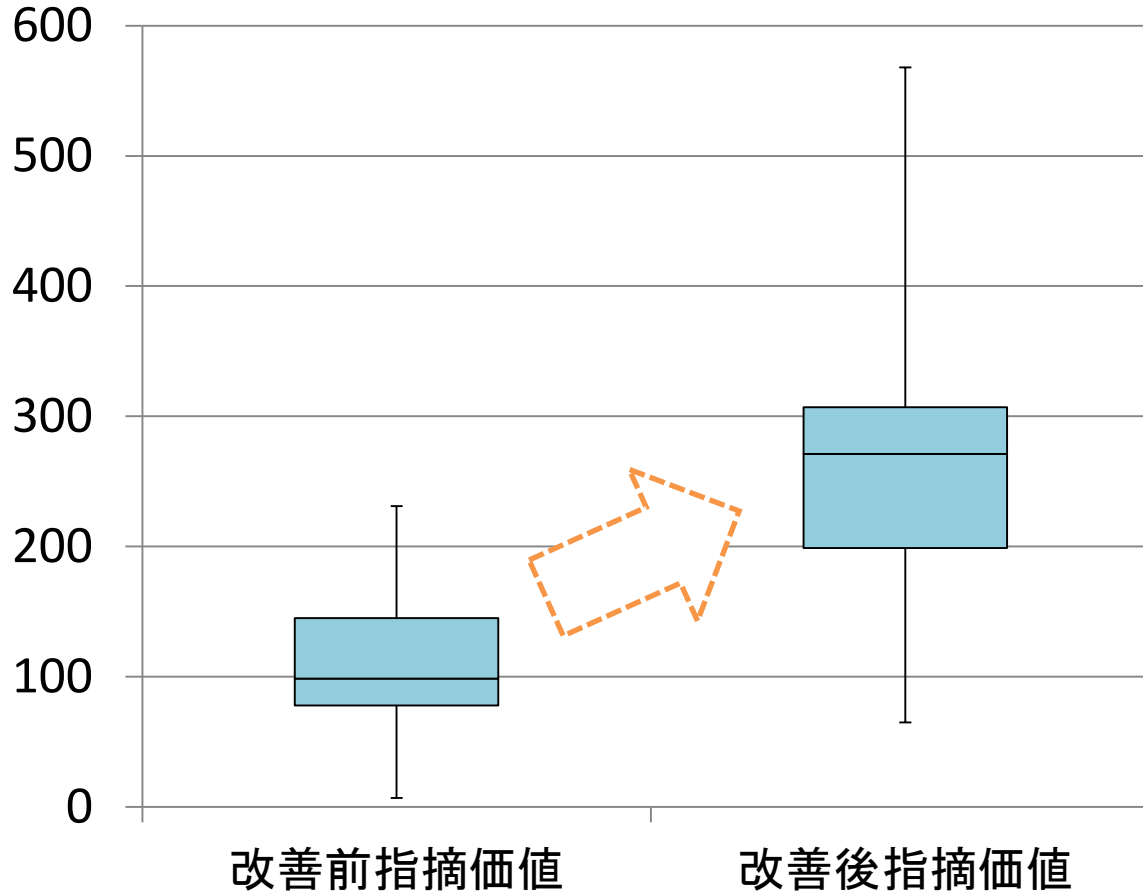
指摘件数の変化



- 指摘件数増: 18チーム (Max20件増)
- 指摘件数変化なし: 2チーム
- 指摘件数減: 4チーム (Max13件減)
- 指摘件数変化平均: 4.8件増／チーム

平均4.8件増
中央値5.5件増

指摘内容(価値)の変化



- ・指摘価値増: 21チーム (Max424p増)
- ・指摘価値減: 3チーム (Max58p減)
- ・指摘価値変化平均: 151.5p増／チーム

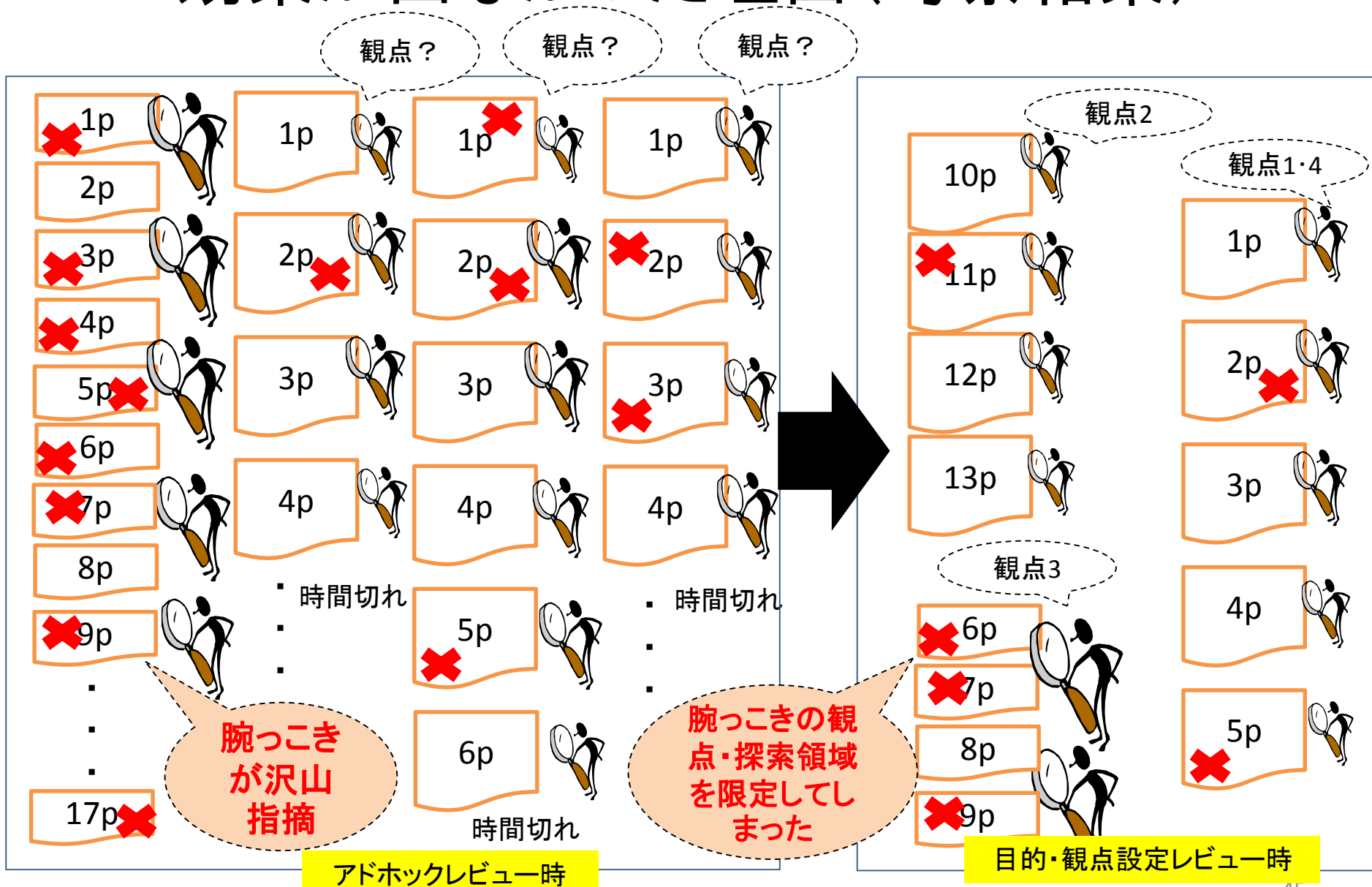
平均151.5P増
中央値172.5P増

結果が出なかったチームの例

指摘件数: △14 → ●11

△改善前 (Before) ●改善後 (After)			発見可能Phase (想定)				計
			実装・UT	IT	ST・OT	C/O後	
			1	3	5	7	
検出効果	効果大 ハード変更必要 サーミスタ壊れたら止められない ミルク温度検出仕様なし	5		●	●		0 ↓ 40
	効果中 沸点記述 整合性 デフォルト エラー後復帰×	3		△ ●●	△△△△		69 ↓ 18
	効果小 誤記 仕向け表記なし ロック押下仕様 長押し仕様	1	△△△△△ △△△△ ●●●●● ●●				9 ↓ 7
計			9→7	9→33	60→25	0→0	78→65

効果が出なかった理由(考察結果)



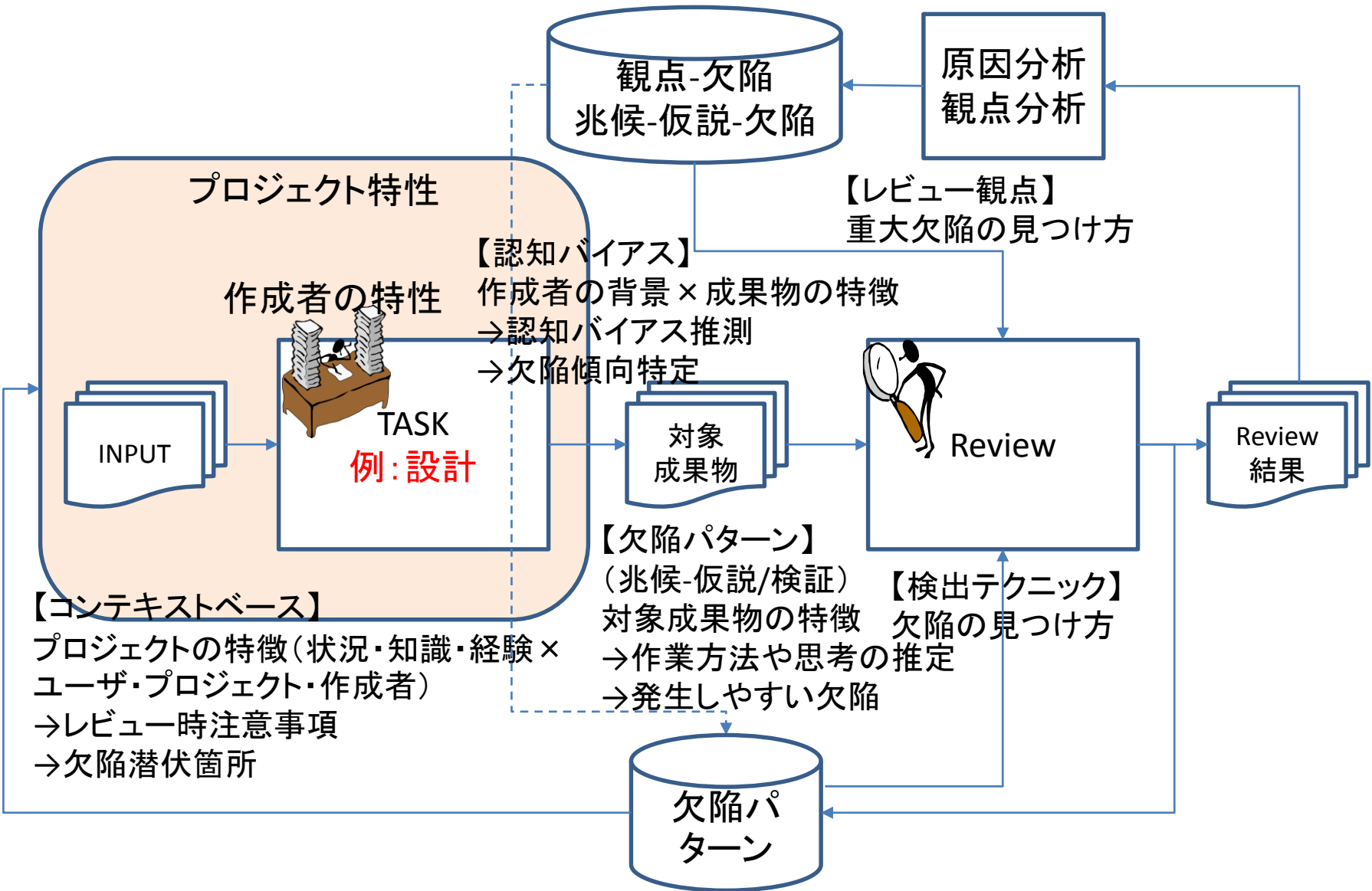
書かれていないことを見つける方法の例

レビューで最も期待されていること

- 過去の記載漏れ事例から確認する
- 表現方法を変えてみる
 - 自分の表現で説明してみる
 - 別の切り口で再構成してみる
 - 図表やモデル化してみる
 - 記述フレームにはめてみる
 - テンプレートとガイドライン
 - テストケース構成要素に割り付けてみる
- 利用者の背景やシーンから読み解く
- 前フェーズ成果物、関連成果物との整合を見る
- 要素間の連携部分を狙う
- 特定の言葉で引っ掛ける など

人間特性を考慮して運営を設計する

これまでのSQiP研究会レビュー関連研究対象(全体像)



人間特性を考慮して運営を設計する

次の状況から想定されることは何か？

どこをどのような視点で確認すると効果的か？

- (1)レビュー対象の成果物は25ページ。ざっと目を通したところ誤字、脱字、衍字が大量に見つかった。
- (2)作成者がレビュー対象の成果物の作成に着手したのは2日前とのこと。
- (3)作成者は先月からこれまで継続して残業、休出が突出していた。過去に参加したプロジェクトで構築したシステムに問題が見つかり、その対応に借り出されているとのコト。
- (4)MS-Wordで作成されたレビュー対象成果物の最終書き込み日時は今日のAM3:35になっていた。また、当初書き込み日時は、5年前の3/28AM2:58だった。

兆候察知 → 仮説設定 → 行動選択

人間特性を考慮して運営を設計する

レビューアの割り当て

人数は3～7名程度

【理想】観点がレビュー出来るメンバーを割り当てる。

【現実】基本的に得意なところを見てもらう。(第一優先)そのうえで、残りを各メンバーで分担。

重点事項・見逃しリスクがある観点は、重点・軽減対策を施す。

複数担当配置・観点詳細化と事前摺り合わせ など

有識者が忙しい場合は、アドホックレビューでピンポイント確認依頼なども考慮。

これらの方法の強み・弱みは何？

	担当a	担当b	担当c	担当d
観点1	○	○	○	○
観点2	○	○	○	○
観点3	○	○	○	○
観点4	○	○	○	○

	重要	担当a	担当b	担当c	担当d
観点1		◎			
観点2	☆			○	◎
観点3			◎		
観点4	★	○	○	◎	○

※何となくアドホックレビューをやると依頼内容はこの状態に近くなる(実施はバラバラ)

人間特性を考慮して運営を設計する レビュー・テストの心理

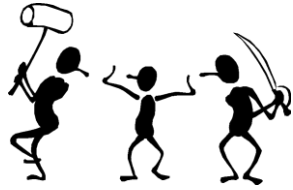
- 見つけたエラー、欠陥、故障を建設的に取り扱えれば、テスト担当者と設計者、開発担当者との間の敵対感情を避けられる。
- テスト担当者やテストリーダーは、欠陥、テストの進捗、リスクの情報をやり取りする場合、建設的に作業が進むよう優れたコミュニケーション能力が必要となる。
- テストで欠陥を見つけて修正するとリリース後に検出して修正するより時間と金の節約になるし、リスクも減る。欠陥の情報はソフトウェアやドキュメントの作成者のスキルを高めることにつながる。
- テスト担当者が「悪いニュースを伝える使者」との扱いを受けるとコミュニケーションの問題が起きる。

コミュニケーション／関係改善方法

“分離”ではなく“協調と連携”で対応する

- 対決ではなく、協調姿勢で開始する。全員のゴールは、高品質のシステムであることと再認識するとよい。
- プロダクトに対する指摘は、中立で、事実を中心にして実施し、欠陥を作りこんだ担当者を非難する方向に行かないようにする。例えば、見つけたことを客観的かつ事実に基づいたインシデントレポートを書いたり、見つけたことをレビューしたりする。
- 他人の気持ちや反応を理解するように努力する。
- 自分の言ったことを他人が理解し、他人の言ったことを自分が理解していることを確認する。

出典: ISTQBテスト技術者資格制度 Foundation Levelシラバス日本語版 Version 2011.J02 より



チームの文化が重要

レビューで想定される悪影響

- **自己責任の放棄・他者責任への転嫁(甘え)**

誰かが誤りを見つけてくれるからそれなりにやっておけばよい。指摘されたら直せばよい。(指摘されない限り一切直さない) 指摘しない奴らの責任にできる。

- **過剰な自己防衛**

恥をかきたくない。指摘されたくない。だから成果物案を見せるのを拒む。指摘されても拒否する など。

**どちらも結局は生産性を悪化させ、要求品質未達成
やコスト超過・納期遅延の原因となる**

人間特性を考慮して運営を設計する プロジェクト計画書指摘と反応の例

レビュー指摘

例1: 計画書でG社との連絡チャンネル・担当者が書かれていない

WIN

例2: 計画書ではG社との連絡チャンネル・担当者が不明である

WIN

例3: 【確認】
G社との連絡・調整はどのように行う予定ですか？
【リスク】

納入済製品の動作に問題がある、納期が遅延すると〇〇工程で△△など危機的状況になりえるように思いますがいかがですか？

LOSE

WIN

Risk
Management
Policy

作成者が適切
に修正し終わる
までがレビュー

相手の反応

LOSE

例1: 計画書に書けばいいのね？それであなたの気が済むの？（形式対応）

LOSE

WIN

例2: 担当Sが窓口でG社と調整しながら進めることになっていますが、、、まだ周知してないですね。追記して周知しなくちゃ。

WIN

LOSE

例3: いちいちうるさいなあ。そんなの知るか！

→手を変えるなど何度か働きかけても対応しない場合はエスカレーションへ

受け取りやすい指摘の伝達

- レビューアが行うのはよりよい成果物を作成するための作成者向け“支援”
- (可能な限り)相手が受け取りやすい伝達により、作成者が自ら進んで行動することが重要(やらせる／やらされる→自ら進んでやる)
- 主語を“あなた”ではなく“私”・“われわれ”
例: 私はこう受け取ったのですが合ってますか？
- 断定的な物言いをしない
例: もしかすると〇〇かもしれないと(私が)感じたので確認してもらえますか？
- 手段よりも事実や目的にフォーカスする

人間特性を考慮して運営を設計する コミュニケーションスタイル

どのような物言いが好き
／嫌いなのか

拠点が
離れて
いる

組織が
異なる

お互い
面識す
らない

お互いを
知らない

幸せでないことがあるときがある

指摘される側
を気遣うあまり
表現が曖昧
になることも

指摘された側
が傷ついたり腹
が立ったりする
ことがある

指摘者も傷つく
ことがある

本来伝えたい
ことがうまく
伝わらない

うまく修正を
ナビゲートで
きなくなる

2019/2/22 SQiP研究会2018年度
成果発表会で研究発表があります！

あなたは何タイプ？コミュニケーション上手になるため
の4タイプ診断法

<https://next.rikunabi.com/journal/20151203/>

「コミュニケーションスタイル」 猪塚修さん資料を編集

Driving【現実派】 自己主張：強 感情表出：弱 指図されるのは嫌い ／思い通りにやらせて よタイプ	Expressive【感覚派】 自己主張：強 感情表出：強 楽しく盛り上がって行 こう！タイプ
Amiable【協調派】 自己主張：弱 感情表出：強 みんなのため頑張れ る／お役に立ちたいタ イプ	Analytical【思考派】 自己主張：弱 感情表出：弱 やるべきことは正確に、 計画通り進めましよう タイプ

論理派 &スピード重視型	感覚派 &スピード重視型
論理派 &マイペース型	感覚派 &マイペース型

レビューで効果を高めるために

【対応方針1】 できるだけ(無駄な)レビューをしない

- － レビューすべきものだけをレビューする
 - ・ レビュー以前の問題は作成者の責任で解決
 - ・ 限られたリソースで最大の効果を発揮するレビュー対象を特定する
- － 集合形式のレビュー時間を最小化する
 - ・ 個別レビューで8割方の決着をつける
- － 一度のレビュー規模を小さくする
 - ・ 対象作成作業とレビューを連携させる
 - ・ 事前すり合わせ、試作→レビュー&早めのフィードバックを繰り返して完成させる

【対応方針2】 ぼんやりレビューを行わない

- － 目的をより効果的に達成するレビュータイプ等を選択する
- － 必要なレビュー観点等を明確化＝狙いを定めてレビューを行う
- － 作成状況や成果物の特徴等から兆候を察知し、仮説を立ててアプローチを決める
- － 人間特性を考慮して運営を設計する

【対応方針3】 レビューをやりっぱなしにしない

- － 完了判定とフォローアップによりリスクを識別し、修正を確実に完了させる
- － ふりかえりにより改善事項を明確にする

レビュー結果の判定

結果判定

☐ 合格

☐ 条件付き合格

☐ 不合格



- ☐ 指摘事項の修正後、再度レビューを実施する。
- ☐ 指摘事項の修正状況と修正結果をモデレータが個別にフォローアップする。
- ☐ 指摘事項の修正結果は、担当者自らが確認し、終了したことをモデレータに報告する。
- ☐ 指摘事項の修正結果は、自ら確認し、問題なければそのまま次のフェーズに進む。

結果記録(例)→修正・フォローアップ／指摘事項の評価

要求仕様レビュー結果を「レビュー効果分析表」にプロットする

No.	指摘箇所	指摘内容	指摘分類	要修正	確認
①	筐体	筐体と構造が“手軽”でないのでは？	ウリ・使用性	☒	☒
②	蓋	閉めると沸騰モードになるのはわかりにくい。	ウリ・使用性・安全性	☒	☐
③	タイマ	単独タイマ機能でコストが高いなら必要ないのでは？(100均ショップで分秒表示タイマ購入可能)	有効性・効率性	☒	☒
④	カルキ抜き	蒸気を逃がす機構がない。カルキが抜けない？	安全性・機能性	☒	☒
⑤	解除ボタン	100msec押下で解除可能。子供のいたずらや偶然の押下でも解除できてしまう可能性あり。	安全性	☒	☐
⑥	解除ボタン	「解除」だけでは何の解除かわかりにくい。	ウリ・使用性	☒	☒
⑦	アイドル移行	pot-330-12・13では蓋センサoffが条件となっているが、pot-221-11では蓋センサが1sec以上offとなったら温度制御行為中止となっている。不整合では？	安全性・内部不整合	☒	☒
⑧	ミルクモード	対象ユーザのニーズなし。必要ないのでは？	有効性	☒	☐
⑨	ミルクモード	蒸気を逃がす機構もないため、沸騰・カルキ抜き後から60℃になるまで時間がかかりすぎないか？(電源onのままだと60℃に到達しない？)	効率性・有効性	☒	☒
⑩	コンセント	筐体との接続部分は安全対策がなされている？	安全性	☒	☒

レビュー結果からリスクを特定

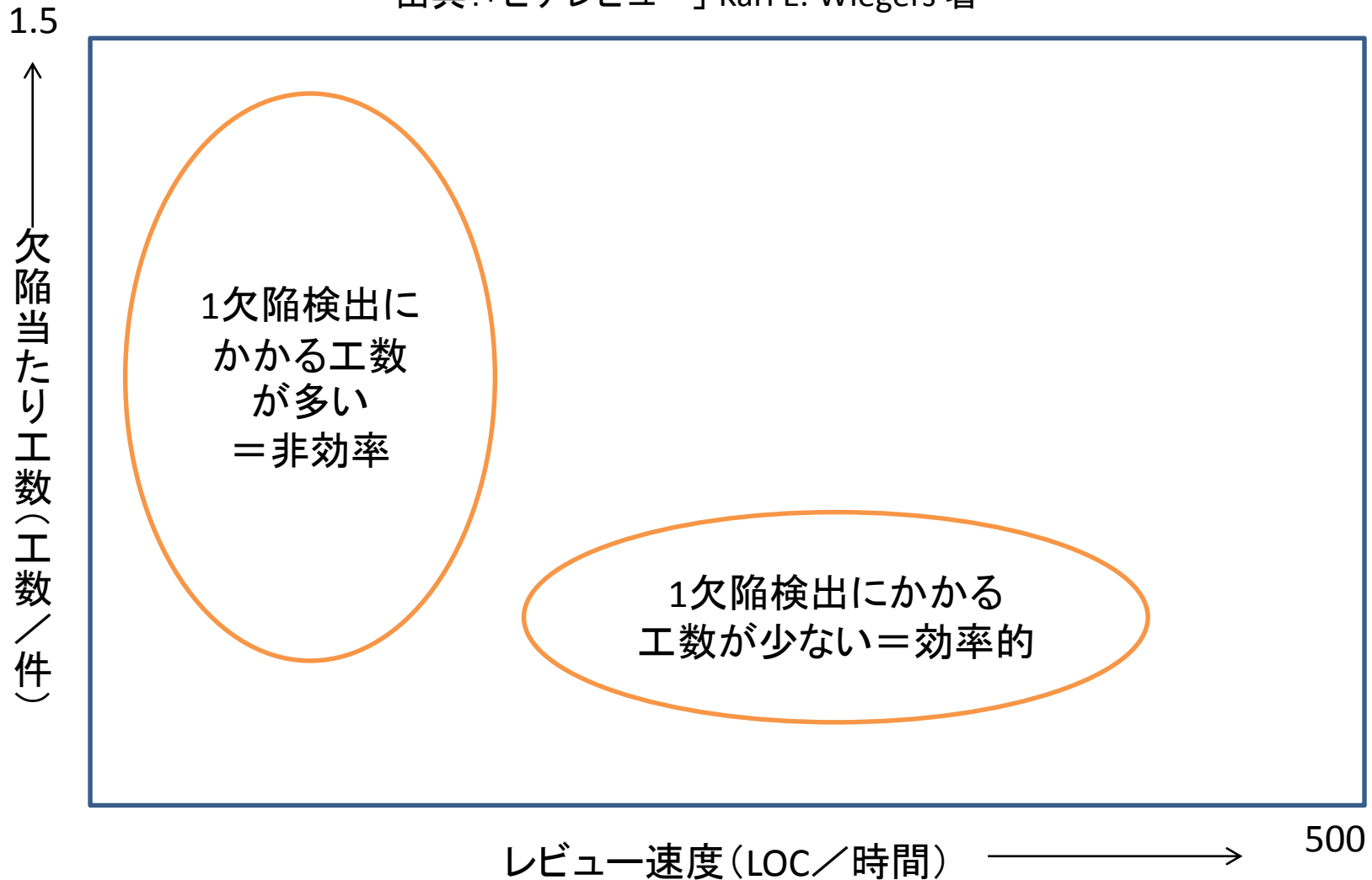
		今回見逃した場合の発見可能Phase(想定)				
		設計	実装・UT	IT	ST・OT	C/O後
		1	3	5	7	9
観点効果／見逃し後影響度	【効果大】 検出欠陥概要 余計な要件・機能 要件・機能漏れ 安全性不備				●●●●● ●●●●● ●	●
	【効果中】 検出欠陥概要 機能上のバグ 設計に影響有の 不整合 重要度小の非機能不備	●	この領域は 大丈夫？			
	【効果小】 検出欠陥概要 設計に影響小の 不整合 規約違反					

		今回見逃した場合の発見可能Phase(想定)				
		設計	実装・UT	IT	ST・OT	C/O後
		1	3	5	7	9
観点効果／見逃し後影響度	【効果大】 検出欠陥概要 余計な要件・機能 要件・機能漏れ 安全性不備					●
	【効果中】 検出欠陥概要 機能上のバグ 設計に影響有の 不整合 重要度小の非機能不備	●●●	この領域は 大丈夫？			
	【効果小】 検出欠陥概要 設計に影響小の 不整合 規約違反	●●●●● ●●●●● ●●●●●				

レビューメトリクス例

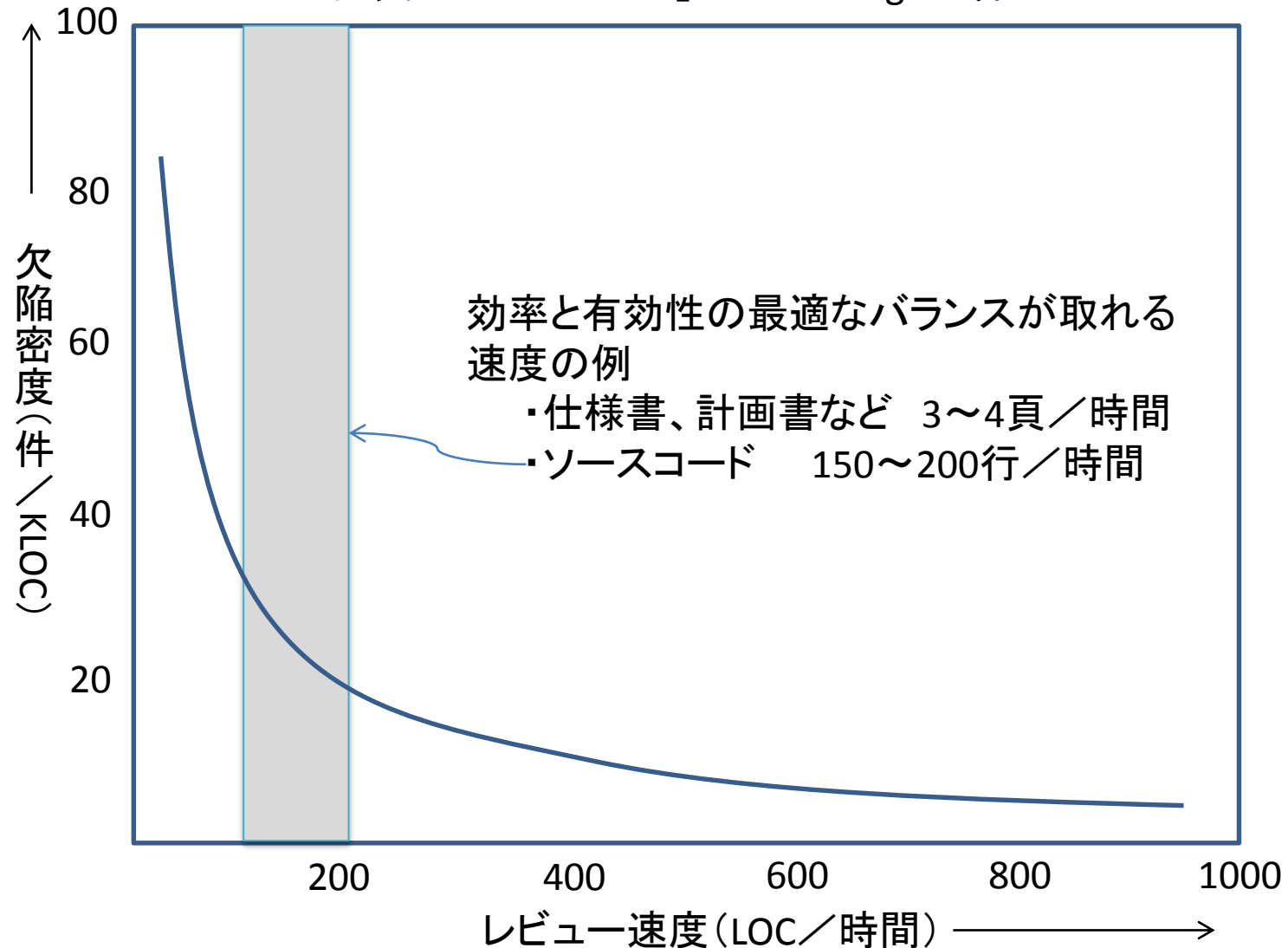
レビュー速度と欠陥当たり工数の関係

出典:「ピアレビュー」Karl E. Wiegers 著



欠陥密度とレビュー速度

出典:「ピアレビュー」Karl E. Wiegers 著



レビューメトリクス例

レビュー速度と欠陥指摘密度の関係

出典:「データ指向のソフトウェア品質マネジメント」野中誠・小池利和・小室睦 著



欠陥指摘密度(件／行)

一般的に「良い状態」

※異常値がある場合は、レビュープロセスと指摘内容の吟味が必要

一般的に「品質不良」

※再レビュー検討

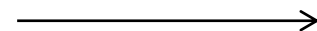
一般的に「良い状態」

※異常値がある場合は、レビュープロセスに問題がないかを確認する

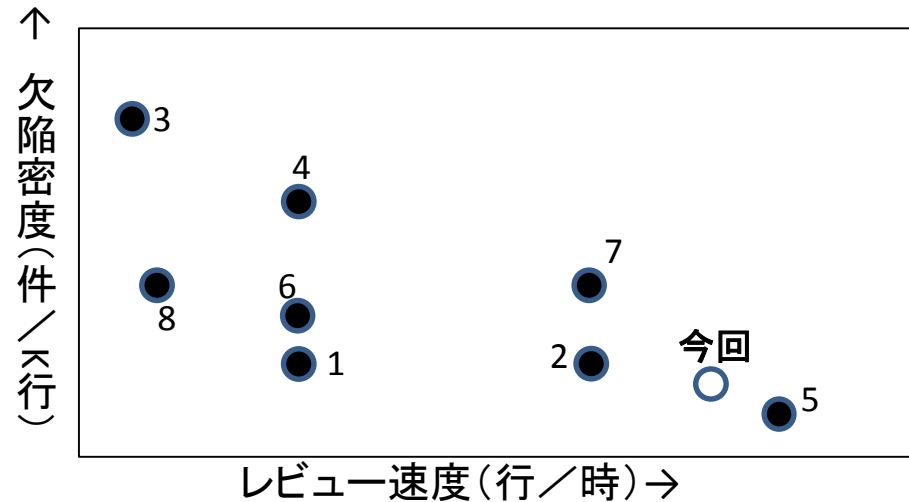
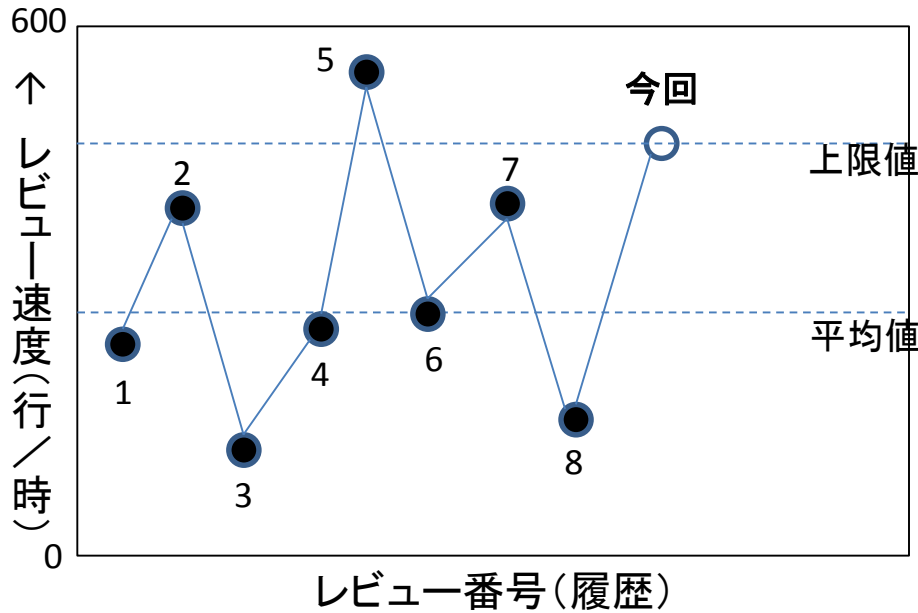
一般的に「急ぎすぎor
見落としの懸念」

※再レビュー検討

レビュー速度(行／分)



レビューメトリクス例



		今回見逃した場合の発見可能Phase(想定)				
		設計	実装・UT	IT	ST・OT	C/O後
		1	3	5	7	9
観点効果／見逃し後影響度	【効果大】 検出欠陥影響 余計な要件・機能 要件・機能漏れ 安全性不備	5			①②③ ④⑤⑨ ⑩	⑧
	【効果中】 検出欠陥影響 機能上のバグ 設計に影響有の 不整合 重要度小の非機能不備	⑦	3			
	【効果小】 検出欠陥影響 設計に影響小の 不整合 規約違反	1			⑥	

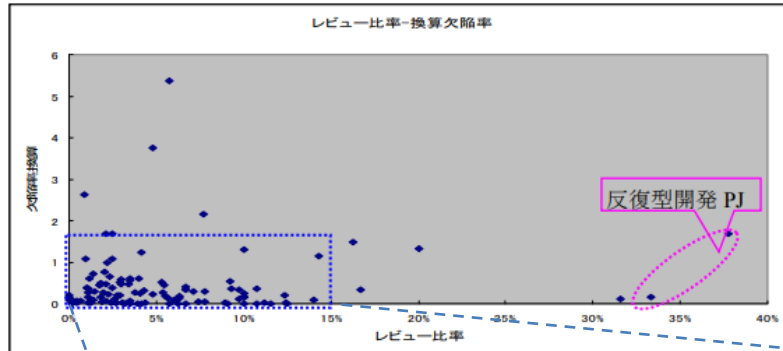
検出欠陥分布

出典:「ピアレビュー」Karl E. Wiegers 著

レビュー比率-換算欠陥数

ユーザー企業 社団法人 日本情報システム・ユーザー協会 2008年6月 2008年版【調査報告書】

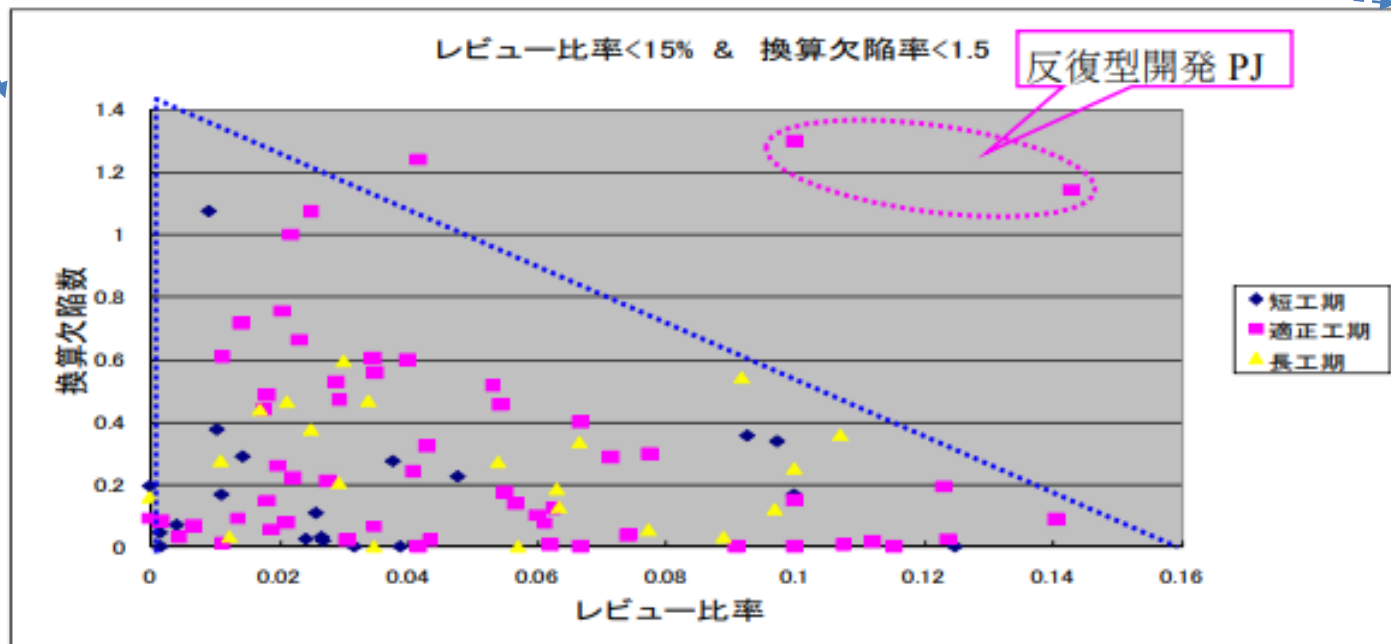
レビュー比率-換算欠陥数



レビュー比率

>10%: 大きな換算欠陥率の点が無い

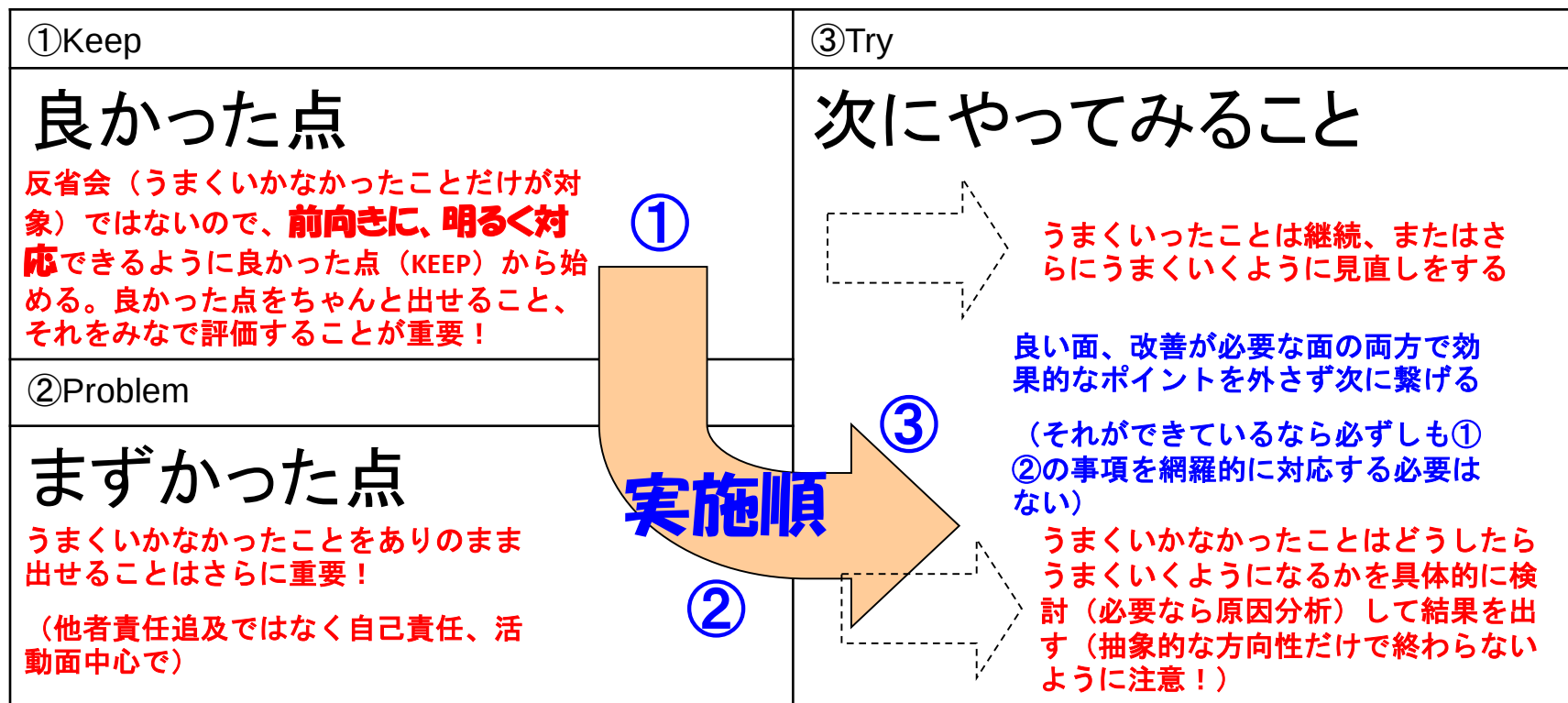
<5%: 大きな換算欠陥率の割合が高い



ふりかえりの流れ

- (実施計画立案～対象活動or期間の終了)
- ふりかえり観点設定
- 実施体制、情報共有・活用方法の明確化
- ふりかえりフレームワークの選択
 - KPT: Keep/Problem/Try
 - YWT: やったこと/わかったこと/次にやること
 - ORID: Objective/Reflective/Interpretative/Decision
- ふりかえり実施
- ふりかえり結果と認識の共有
- ふりかえり結果の活用(以降の活動で改善実践)

KPTによるふりかえり概要

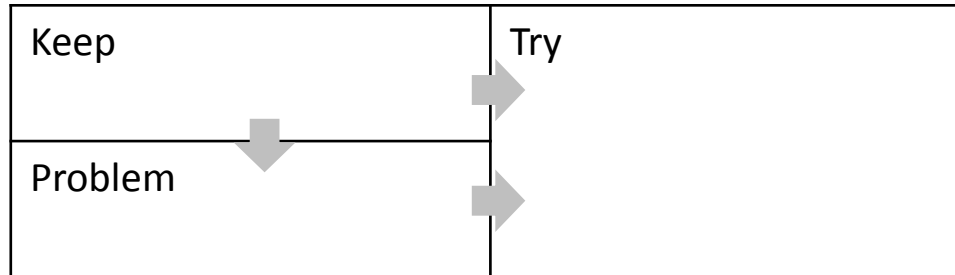


作業の節目（週末・フェーズ終了時・業務完了時など）に関係者全員で“良かった点（Keep）・まずかった点（Problem）・次にやってみること（Try）”を出し合い、情報と認識を共有したうえで、次の業務や作業に確実に繋げる。

ホワイトボードに手書きで対応する、ワープロ・プレゼン用アプリケーションで対応するなど対応方法はいろいろありえる。

ふりかえりフレームワークの例

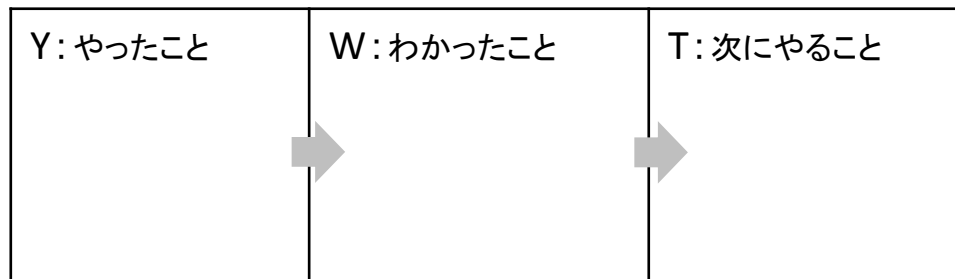
KPT



最も普及しているフレームワーク。

シンプルなので使いやすいが、結果が個人の主観・記憶に左右されやすい側面もある。そのため初期導入時ほど、ふりかえり対象期間を短くする、腕のよいモデレータが進行するなどの工夫が必要になる。

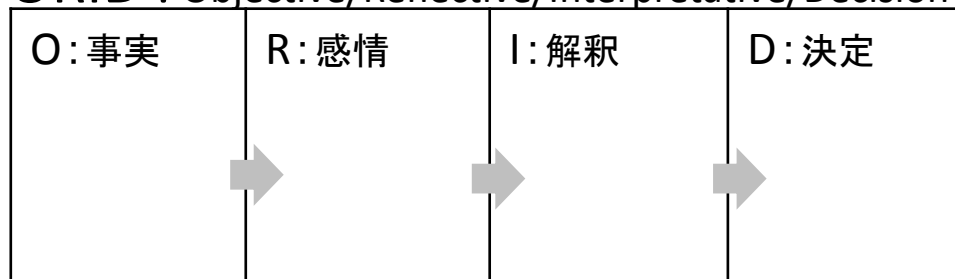
YWT



KPTの次に普及しているフレームワーク。シンプルなので使いやすく「Y: やったこと」からはじめるため過去の記憶を想起しやすい特徴を持つ。一方で「W: わかったこと」が「Y: やったこと」に引っ張られて一面的になるケースも。

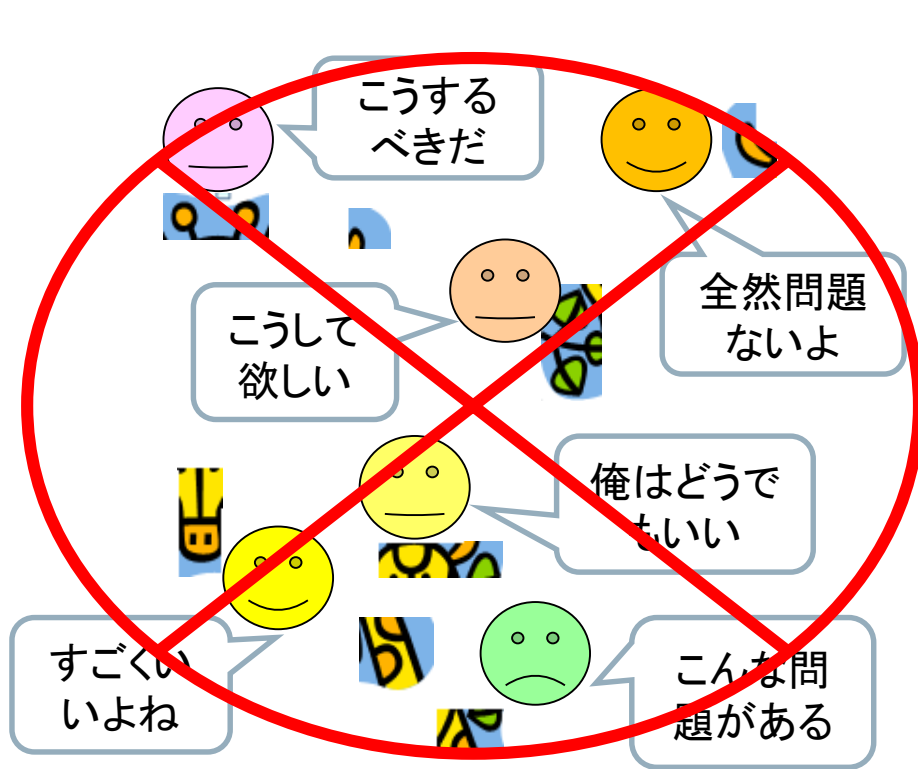
ふりかえり対象のリードタイムが長い場合などに有効。

ORID : Objective/Reflective/Interpretative/Decision

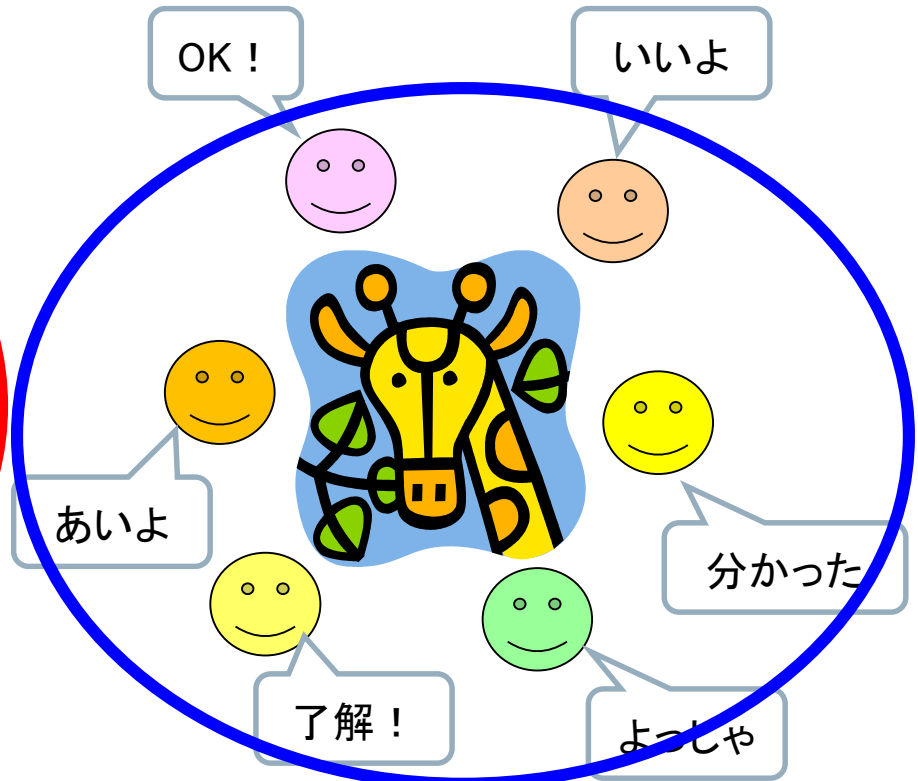


客観的事実だけでなく、それをどう感じたのかを含めることにより、メンバーがより気持ちよく能力を発揮してチームパフォーマンスを上げることを目指す。KPTで運営できるようになったあと、より深くふりかえりを実践する場合などに使う。

ふりかえりのゴール(目指すこと)

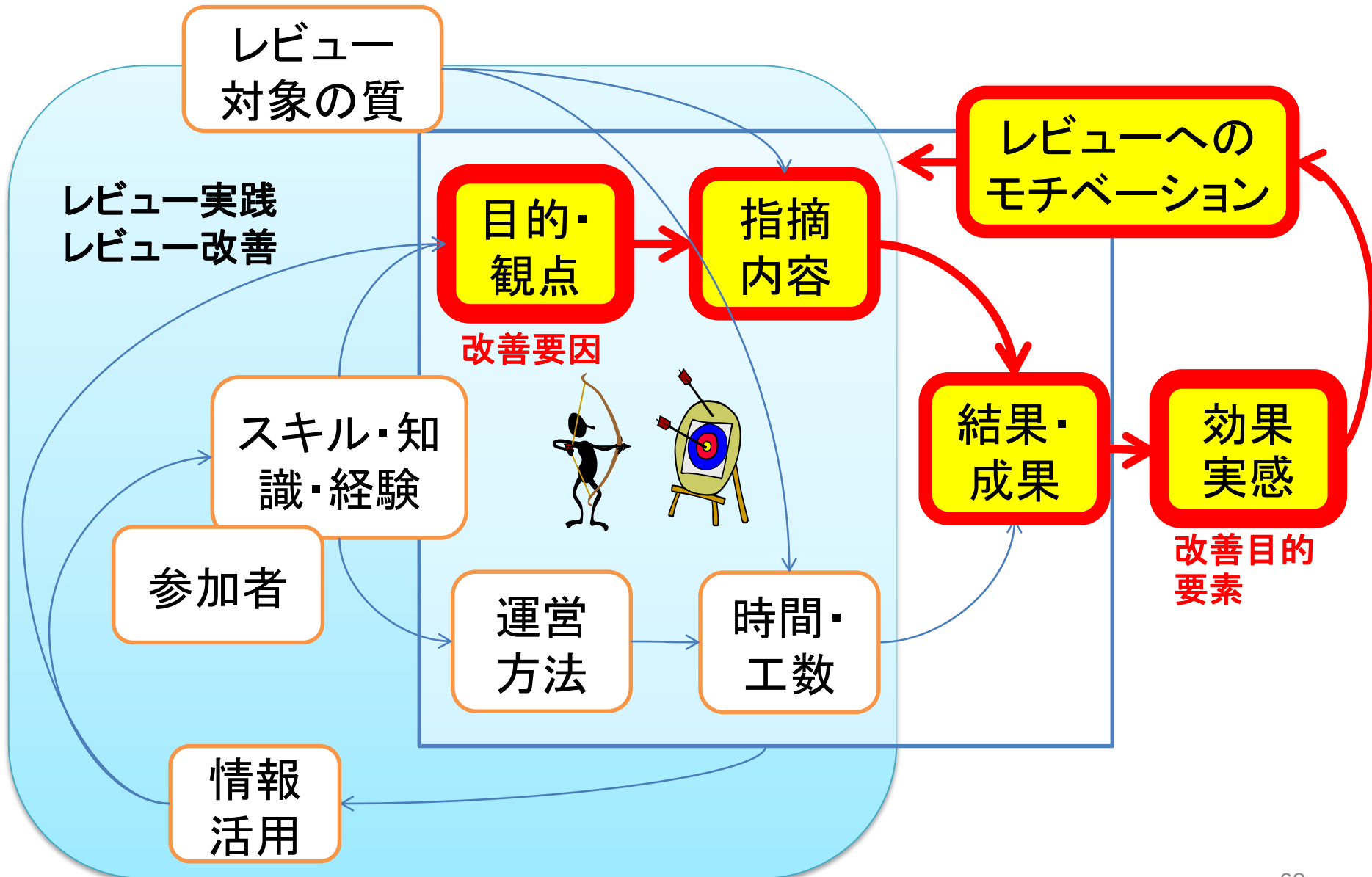


✕ 組織、チーム内で目的・目標が共有されておらず、個々のメンバーが自ら見えている領域で様々な意見や不満を持っている。今後の方向性を含めて認識が合っていない。

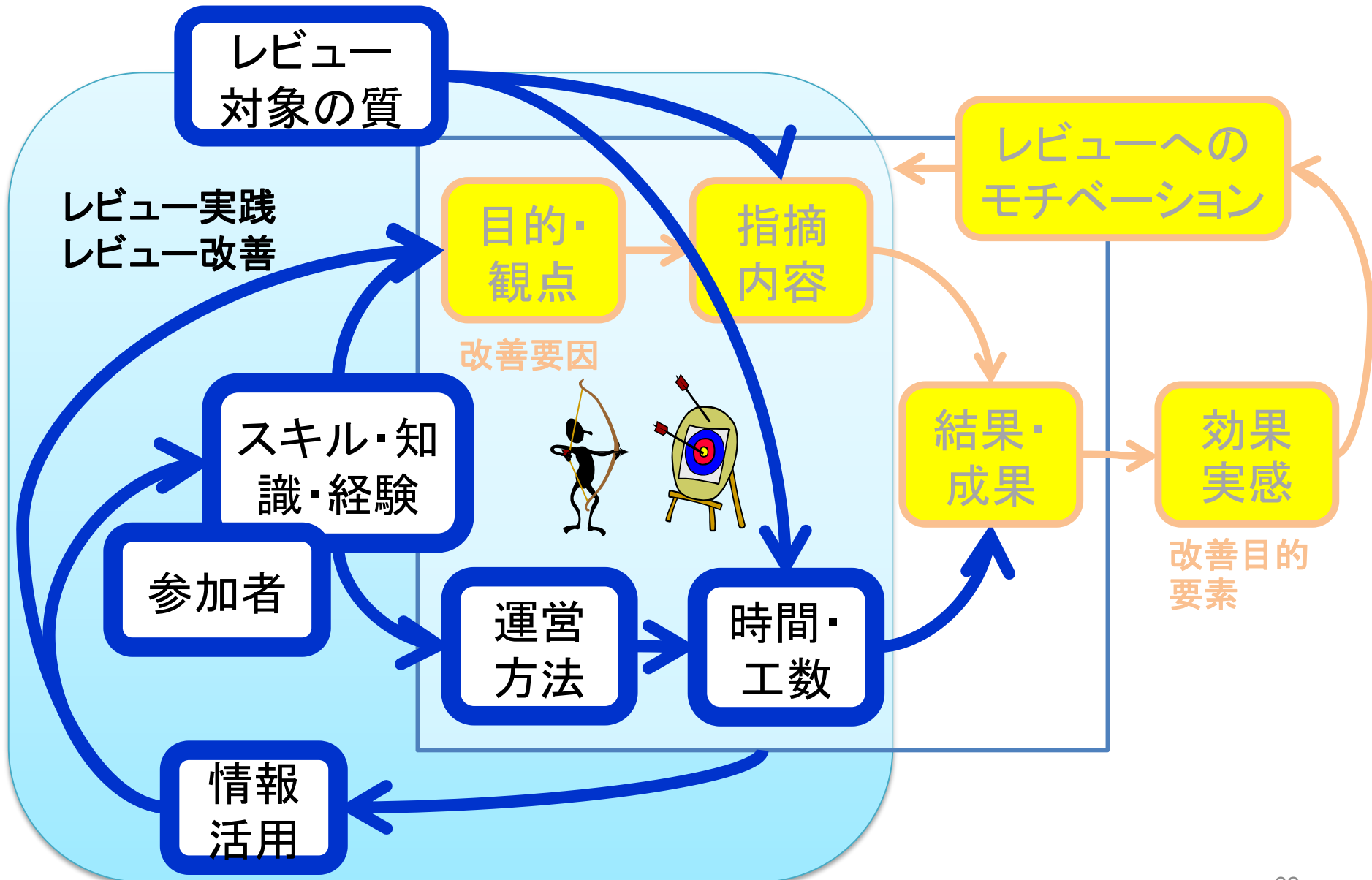


◎ 組織、チームとして目指す目的・目標に対して、現在の状況や今後の方向性に対応、各自のやるべきことについて関係者全員が同じ認識を持ち、同意している。

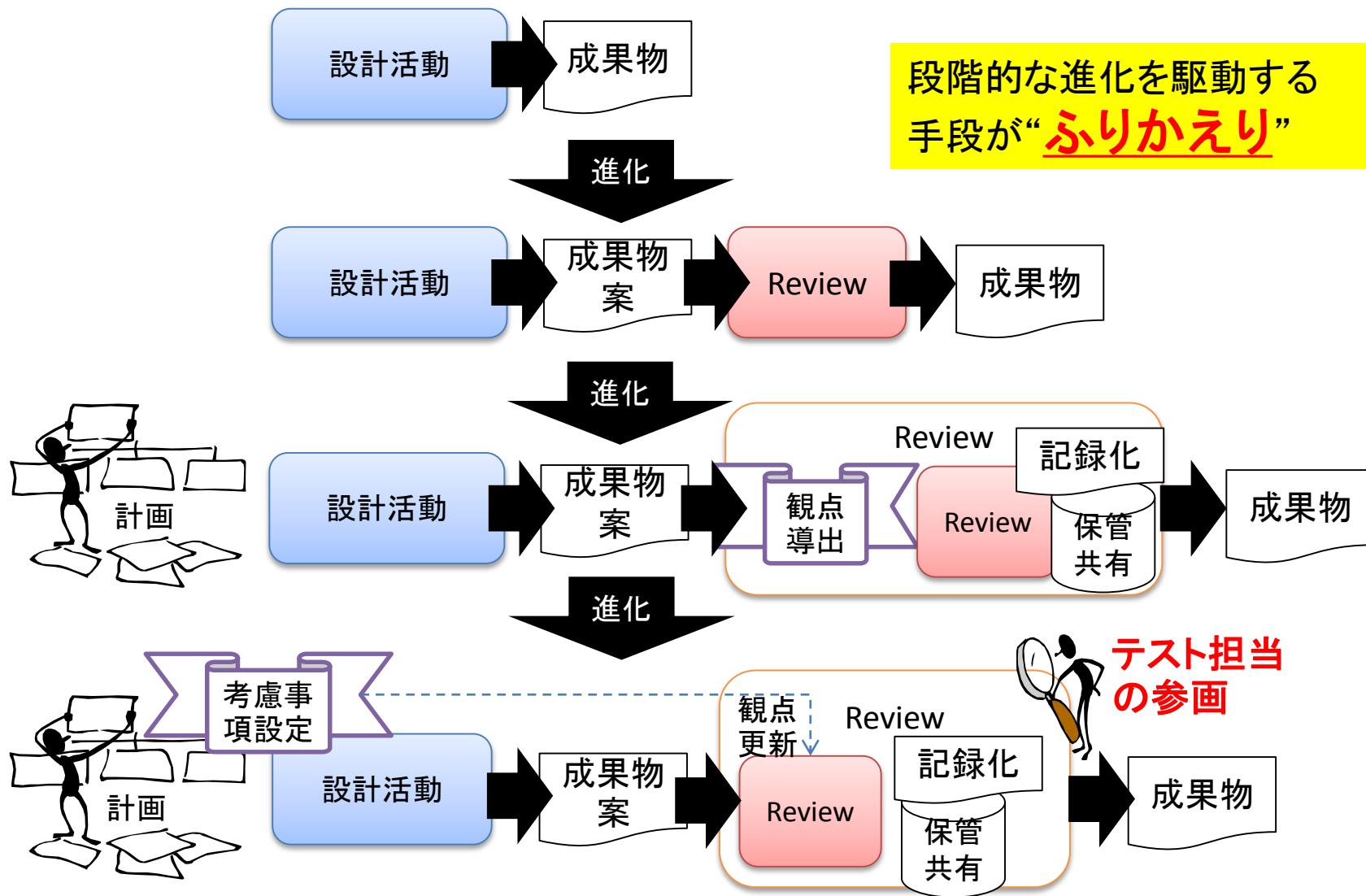
しっかり狙って指摘できるようにする



指摘しやすい環境をつくり出す



レビュー改善による設計プロセスの進化例



ちょっとまともなレビューワークシナリオ

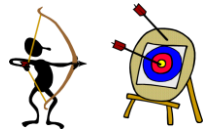
- 1)レビューが依頼される。
- 2)レビュー対象の内容や背景を事前に把握する。レビュー開始できるかを判定し、基準を満足しない場合は作成者にその旨を伝えて返却する。
- 3)レビュー対象の特徴、該当フェーズ、入力情報、製品リスク、利害関係者の要望などを考慮してレビュー目的・観点を導出し、体系化する。
- 4)観点の重要度などを参考にして優先順位を付与し、それぞれのレビューアが担当するレビュー観点を調整し、確定する。
- 5)役割分担に従い、割り当てられた観点でレビューを行う。各レビューアは時間制約の中で、最も重要な観点から確認する。
- 6)観点が特定されているため、探索領域を絞り込みやすく、欠陥検出がしやすくなる。また、**観点重複、探索領域の偏り、指摘重複が減少**する。
- 7)以上の結果、**かけた時間・工数に対して有効な指摘事項が多くなる**。レビュー結果がレビューアのスキルに依存する度合いが減少する。
- 8)ふりかえりによりレビューの成果、よい点、問題点、以降のレビュー改善内容が明確になる。
- 9) **費用対効果が高くなる**ため、レビューに関わる要員はレビューの**効果を実感**し、以降の**レビュー実施やその改善に対するモチベーションが高まりやすい**。

(3)改善を効果的に進めるために
カイゼンってどうやればうまくいく？

改善の制約条件

- 組織には改善に使える余裕はほとんどない。
- 実務は問題だらけ。すべてを一網打尽にはできない。
- 関係者はそれぞれの立ち位置で見えている問題を、自ら見えているものを見たい(歪んだ)見方、で捉えている。
- 関係者それぞれの思いは違っている可能性が高い。
- 人に命令されたコトは後ろ向きに、自分で決めたことは前向きに実施する可能性が高い。
- 人間は感情の動物、慣習を好む＝変化を避ける傾向がある。論理より感情が勝る。
- 自らが思う問題を相手が理解するのは意外と難しい。
- 理解・納得していないことにチカラを注ぐ人はいない。
- 過去と他人は変えられない＝誰かのせいにしても解決しない。
- 変えられるのは自分と未来だけ。
- 撃ち落せる問題領域の幅や奥行きは、担当者～組織が持つ能力に依存する。

これら制約条件をベースに進められる改善プロセスでなければならない



改善を進める上で大事なこと

- 多くの問題、困り事から、取り組む価値のある改善ターゲットを特定する
 - 使えるリソースは非常に限られている！⇒一網打尽はムリ
 - 複数の要素が相互に作用して問題や困り事が発生している
 - 問題、困り事発生メカニズムから取り組むべき改善ターゲットを特定する
- 関係者全員が状況を理解・納得し、当事者（自分事）として取り組む
 - 納得していないことに労力を注ぐ人はいない
 - 当事者として関わらなければ成功の実感も得られない
 - 他者責任追及はあとで／自分たちの役割の範疇で改善すべきことを優先にする
- 改善ターゲットが大きければ、価値ある小さい領域に絞り込む
 - たくさんのことを、長い期間で改善する方法は結局うまくいかなくなる
 - 小さい領域をできるだけ短い期間、少ない工数でちゃんと打ち落とす
- 実践可能で、効果・効率を兼ね備えた手段でターゲットを撃ち落とす
 - 現状能力レベルの110～120%程度のストレッチ目標がベター
 - できなかったことができるようになる、自ら狙って撃ち落せたら“うれしい”
- 小さな成功を積み重ね、改善力と成果を雪だるま式に高める
 - 100m全力走の連続実施や20km全力走は困難
 - 継続できるレベルのFun Runを継続しながらレベルを上げていくのが理想

SaPID(サピッド)

Systems analysis / Systems approach based Process Improvement methoD

(**業界・職種を問わない**)組織やチーム、管理者やエンジニア(または実務担当者)が期待する成果を得る、そして成長するために、当事者自らが**システムズアプローチ**を実践しながら段階的にゴールを目指すプロセス改善手法。

※システム化検討(現状分析～企画立案)時や、さまざまな業種の改善にも適用可能な手法

システムズアプローチ

対象領域に存在するさまざまな要素の関係性、相互作用全体を一つのシステムと捉え、システム的な思考と各種手法を駆使しながら分析・評価・最適化などの段階(フェーズ)を経て複雑な問題の解決を探る方法論。

問題はさまざまな要素の相互作用・関係性が引き起こしている、という立場で解決策を見出す。

関係者からさまざまな情報(構成要素)を収集して「問題構造図」に表し、関係者間で共有することで自らの背景や現状に対する適切な解決策を探る。

SaPIDによる改善事例（一部）

今回のアプローチもSaPIDを活用した事例です

◆情報処理推進機構（IPA）主催セミナーでの事例解説

「ソフトウェア開発現場の"気づき"から始めるプロセス改善 実施事例紹介その1」

https://www.youtube.com/watch?v=n4aRBB_zv2c

※SaPIDを活用してシステム保守を行うチームが自ら改善した事例の解説ビデオ。
表題はSPINA3CH（スピナッチキューブ）ですが、SaPIDでアプローチした事例です。

◆ソフトウェアテストシンポジウム2016東京（JaSST'16東京）

「レビューの目的・観点設定の効果と課題」

<http://www.slideshare.net/AdachiKenji/ss-59510938>

レビューの問題点をSaPIDシステムズアプローチで分析し、攻略ポイントとその解決手段を導出、実践結果を考察したもの。

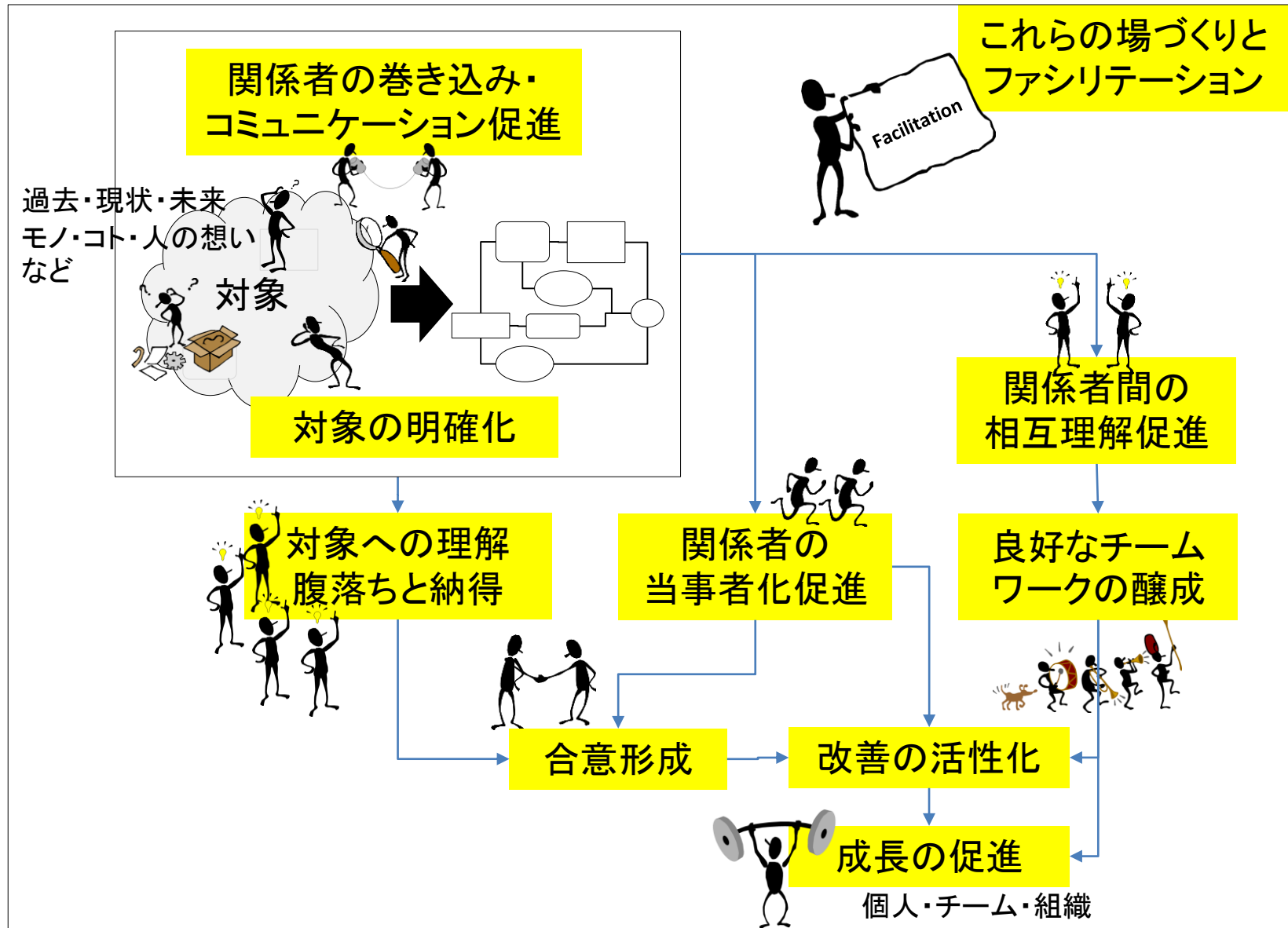
◆ソフトウェアテストシンポジウム2018東京（JaSST2018東京）

「TPI Nextを活用したチームメンバーの問題意識から始めるテストプロセス改善」

<http://bit.ly/2FLtpHU>

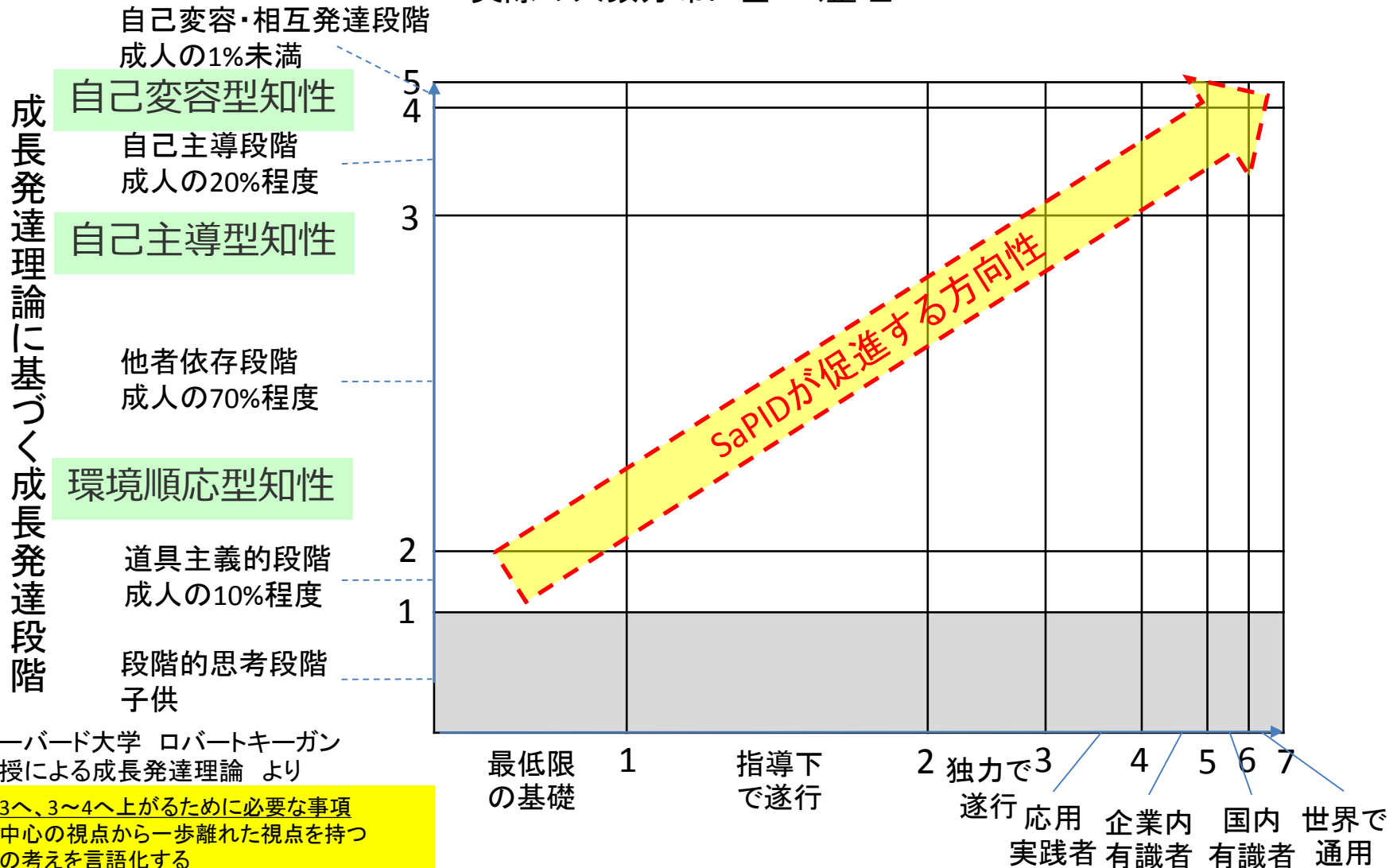
テストの問題点をSaPIDシステムズアプローチで分析し、TPI NEXTによる評価結果を併用しながら自律化を目指すアプローチの事例です。

SaPIDでできること・やっていること



SaPIDが促進する成長発達段階とスキルレベル

実際の人数分布に基づく整理



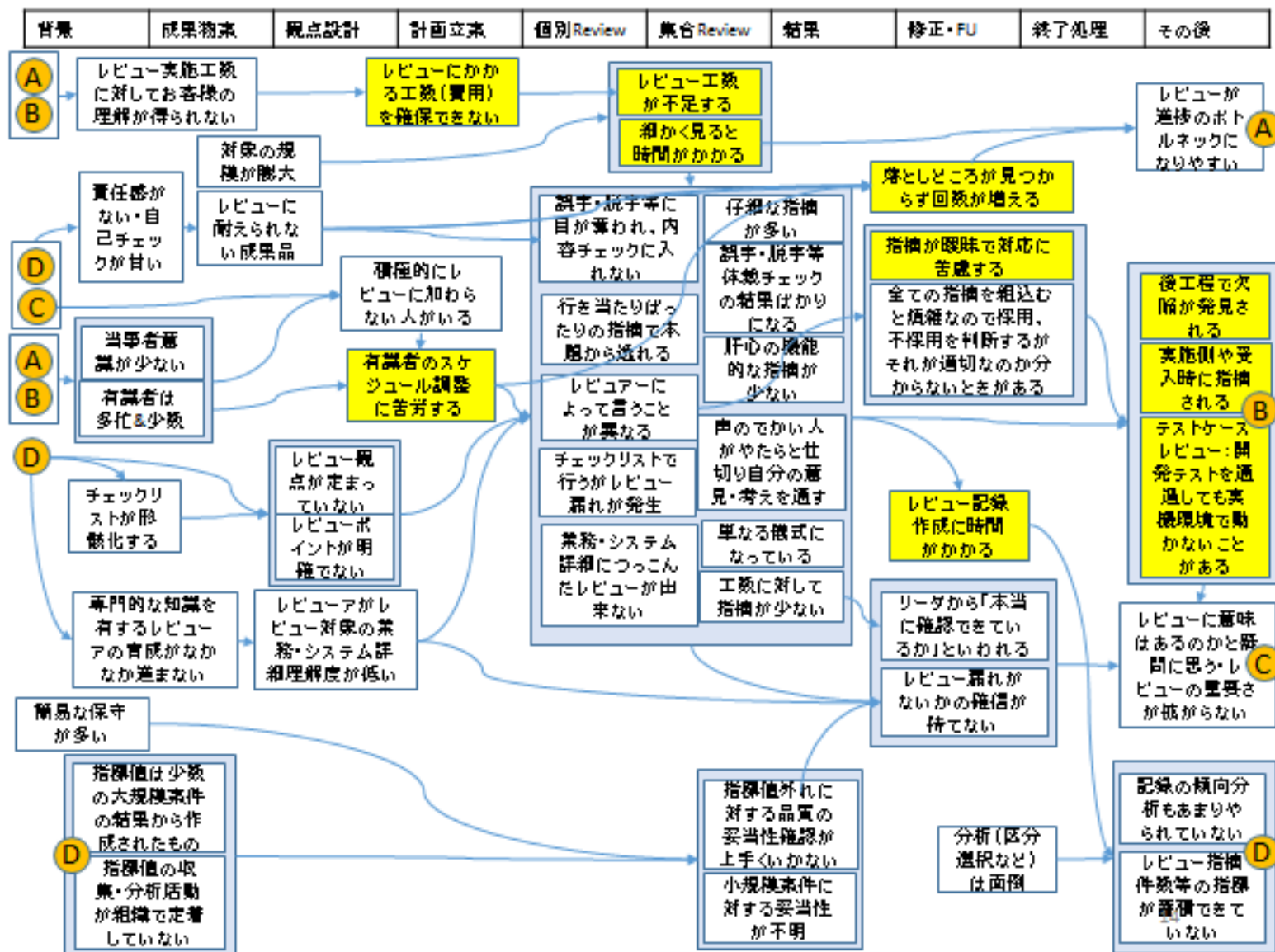
ITSS・ETSS等に基づくスキルレベル(分野毎)

ITスキル標準とは？ <https://www.ipa.go.jp/jinzai/itss/itss7.html>

(4)講演のふりかえりとまとめ

今日は何の話だったかな？

レビューの困り事と問題構造



レビューで効果を高めるために

【対応方針1】 できるだけ(無駄な)レビューをしない

- レビューすべきものだけをレビューする
 - レビュー以前の問題は作成者の責任で解決
 - 限られたリソースで最大の効果を発揮するレビュー対象を特定する
- 集合形式のレビュー時間を最小化する
 - 個別レビューで8割方の決着をつける
- 一度のレビュー規模を小さくする
 - 対象作成作業とレビューを連携させる
 - 事前すり合わせ、試作→レビュー&早めのフィードバックを繰り返して完成させる

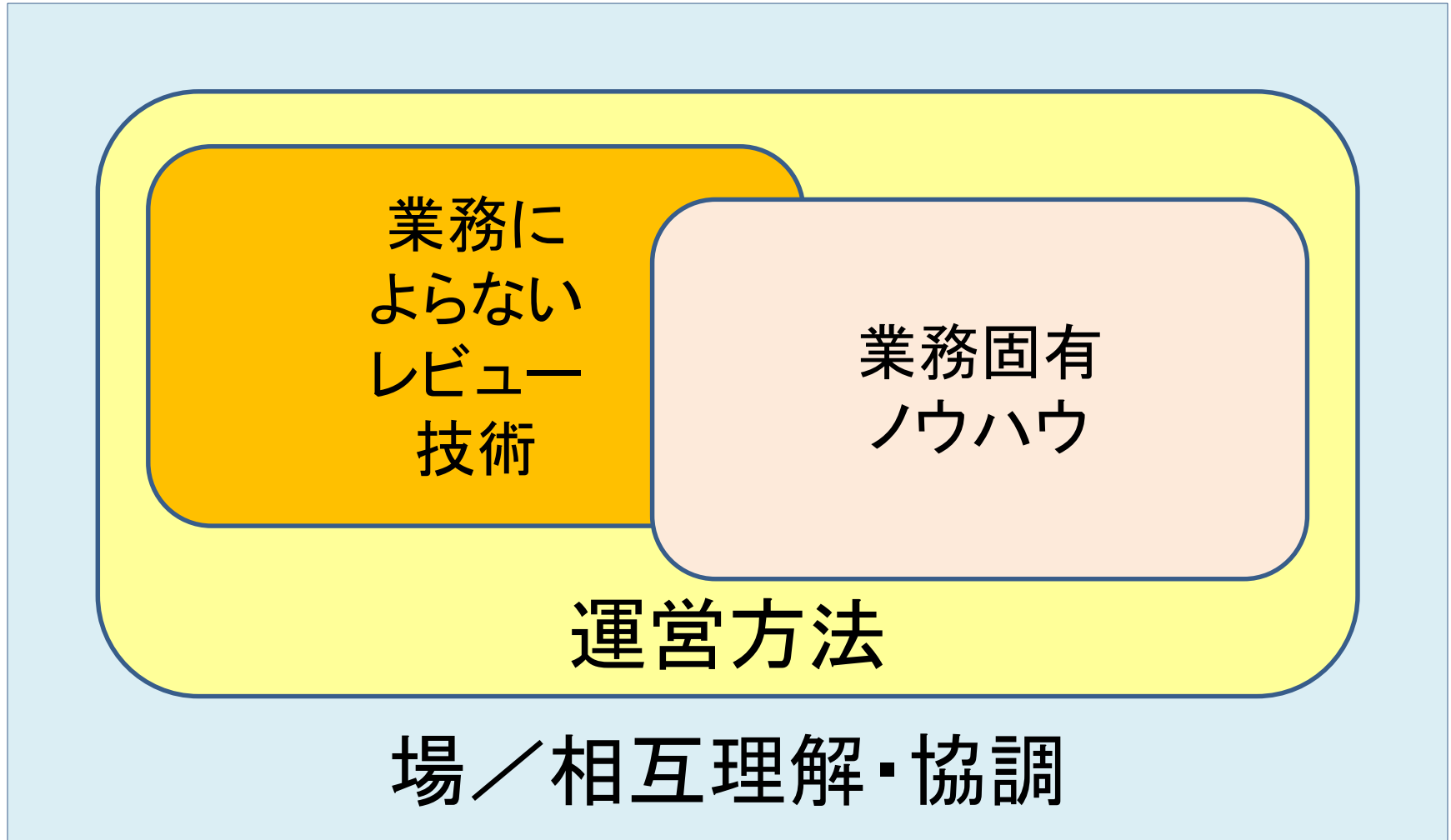
【対応方針2】 ぼんやりレビューを行わない

- 目的をより効果的に達成するレビュータイプ等を選択する
- 必要なレビュー観点等を明確化＝狙いを定めてレビューを行う
- 作成状況や成果物の特徴等から兆候を察知し、仮説を立ててアプローチを決める
- 人間特性を考慮して運営を設計する

【対応方針3】 レビューをやりっぱなしにしない

- 完了判定とフォローアップによりリスクを識別し、修正を確実に完了させる
- ふりかえりにより改善事項を明確にする

レビューの有効性を支える要因



当講演のふりかえり

何か実務に役立ちそうな内容がありましたか？

これは大事・試してみたいと感じたこと	具体的にどのようにするとよいか？ (どのタイミングで何をするか？何で結果・成果を把握すればよいか？など)
大事・試したいと思ったコトに関する現状	

相互理解&協調によるチームプレーで
楽しく仕事をして、「いいね！」と言わ
れるシステム・ソフトウェアをどんどん
提供していきましょう！

ありがとうございました。

参考文献

- 「ピアレビュー」 Karl E.Wiegers著, 大久保 雅一監訳
- JaSST'16東京事例発表: レビュー目的・観点設定の効果と課題
<http://jasst.jp/symposium/jasst16tokyo/pdf/A2-1.pdf>
- JaSST'15東北基調講演: 現在のレビューに必要な次の一手を把握するレビュー実践ウォークスルー
<http://jasst.jp/symposium/jasst15tohoku/pdf/S1.pdf>
- 集中力の維持と長期的な学習効果につながる方法(東京大学・池谷裕二教授の見解)より
http://www.asahi.com/ad/15minutes/article_02.html
- 産学連携によるデザインレビュー改善事例 細谷泰夫、吉岡克浩、森崎修司
http://www.jaspic.org/event/2011/SPIJapan/session1B/1B3_ID003.pdf
- SPI Japan 2011: ふりかえり実践方法の変遷による業務運営プロセスと成果の改善
http://www.jaspic.org/event/2011/SPIJapan/session3B/3B4_ID008.pdf
- 「コミュニケーションスタイル」 猪塚修さんの資料
- あなたは何タイプ? コミュニケーション上手になるための4タイプ診断法
<https://next.rikunabi.com/journal/20151203/>
- 20180413開催: SaPID3.0ガイダンス資料
- ユーザー企業 社団法人 日本情報システム・ユーザー協会 2008年6月 2008年版【調査報告書】
<http://www.juas.or.jp/cms/media/2017/02/08swm.pdf>
- 「なぜ弱さを見せあえる組織が強いのか - すべての人が自己変革に取り組む「発達指向型組織」をつくる」 ロバート・キーガン (著), リサ・ラスコウ・レイヒー (著), 中土井僚 (著)
- ITスキル標準とは? <https://www.ipa.go.jp/jinzai/itss/itss7.html>
- 「データ指向のソフトウェア品質マネジメント」 野中誠・小池利和・小室睦 著

参考文献：SQiPライブラリー掲載論文

<http://www.juse.jp/sqip/library/>

- 欠陥パターンと検出テクニックを用いたレビュー手法の提案とそのトレーニング教材の有効性評価の報告
- HDR法：仮説駆動型レビュー手法の提案
- 検出難易度の高い欠陥を検出するレビュー観点の提案
- SBR（ステルスベースどれビュー手法）の提案
- 欠陥連鎖チャートを用いたレビュー方法の提案
- 作成者の認知バイアスに着目したレビュー手法の提案
- 要求仕様書の特性に着目した個人レビュー手法の実験的評価
- 設計者自身による設計品質の作り込み～テストエンジニア視点の活かし方
- 間接的メトリクスを用いて欠陥予測を行うレビュー方法の提案
- ソフトウェア品質不安に対する心理的側面に着目した、レビュー計画作成技法の提案
- 要求仕様書におけるテストエンジニアの視点を活かした欠陥検出方法の提案
- 効率的・効果的なレビュー実施のための新規役割「ハーベスタ」の提案
- 重点欠陥を早期是正するレビュー手法 3分割レビューの提案
- 重点欠陥を効率よく検出するレビュー手法の提案と有効性の実験報告
- 重大欠陥検出に集中するための問診に基づくレビュー法（IBR法）の提案

ソフトウェアレビュー関連書籍



ソフトウェアインスペクション
Tom Gilb、Dorothy Graham著



ピアレビュー
Karl E.Wieggers著
大久保 雅一 著



間違いだらけの設計レビュー
森崎修司著



今年も開催予定！

①設計レビューで問題を叩き出せ！～QAの役割～
白水 俊治氏(日立製作所)

②レビュー再定義 及部 敬雄氏(楽天株式会社)

③アーキテクチャのレビューについて 鈴木 雄介氏
(グロース・アーキテクチャ&チームス)

