

2019/1/25 祝 JaSST'19北陸 初開催！

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【補足資料】

JaSST'19北陸 事前アンケートと分析結果

株式会社HBA 安達 賢二
<http://www.software-quasol.com/>
adachi@hba.co.jp

事前依頼事項：アンケート

みなさんから40件ほどの回答をいただきました。
ご協力ありがとうございました！

- ・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。

JaSST'19北陸 事前アンケート結果

レビューポイントが明確でない

レビューによって言うことが異なる

レビューの指摘が曖昧で対応に苦慮する

レビュー対象が膨大

レビュー工数が不足する

「声のでかい人」がやたらと仕切り、自分の意見・考えを通そうとする

当事者意識が少なく、積極的にレビューに加わらない人がいる

自己チェックが甘い

形式チェック(誤字・脱字など)に目が奪われる

レビューが内容チェックには入れない

自己チェックリストも形骸化する

有効なレビューをするために有識者を集めるが、スケジュールの調整に苦労する

有識者はとにかくビジー

⇒有識者でなくても、それなりの人で出来る様な方法はないの？

レビュー結果のドキュメント化(レビュー記録)に時間がかかる

記録の傾向分析もあまりやられていないのが現状

レビュー記録から区分を選択するなどは面倒)

①業務内容、システム内容の理解

レビュー者がレビューする開発システムの業務詳細、システム詳細に関して理解度が低い

つっこんだレビューが出来ない

②レビューの育成

レビューの育成がなかなか進まない。

専門的な知識を有するメンバを育成することで効果的なレビューを行うことが可能であるが、その場合、レビューに多くのメンバを関与させる必要がある

レビューがレビュー対象の知識があまりない場合のレビュー方法

(背景)テスト設計レビューの段階において、知識がないと漏れがないかの確信が持てない。設計書など他のドキュメントとの整合性確認や有識者をレビュー会に含めるなどにはなると思われますが。

実機環境と開発環境のテスト方法の違いに関するテストケースのレビュー

(背景)社内開発環境でシステムテストがOKであっても、実機環境で動作しないことが多々ある。原因としては、プログラムの不具合ではなく環境不具合であることがほとんど。(例: 実機のADに参加したら動作しない。実機(お客様)のユーザーだと動作しない。実機のパソコンだと動作しないなど)※実機のみのテストをどう選別するか。

①業務視点でのレビューが、なかなかできない(または、弱い)場合がある。

⇒お客様業務を深く理解できない場合に、社内レビューが手薄、浅はかとなる

※お客様業務を知る、業務深耕を図ることが大事ですが

②レビュー指摘件数などの指標が蓄積できていない

⇒ソフトウェア規模に対する指標などと成るべきであるが、ソフトウェア規模が定量化はされていない

パッケージソリューション導入、スクラッチ開発導入、流用導入 等様々である。。。

※ルールを決めて、指標蓄積が大事であるが。。。"

レビューを受ける側もレビューコメントをする側も、どちらもとても善意な姿勢でいるにもかかわらず、幸せでないことが起こることもある。

お互い、成果物がよりよくなることを目指しており、

そのための指摘であることは十分承知しているのだけれども、

どんなに指摘者が気遣っても(そしてそれを指摘される側が察していたとしても)

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し、

感情に触ったことを指摘者が知ると指摘者も傷つくことがある。

指摘される側を気遣うあまりに、表現が曖昧になり、本来伝えたいことが伝わらずにうまく修正をナビゲートできなくなることもある。

毎日近くで顔を合わせ意見交換をするチーム内であれば、チームビルディングを工夫することで対応しやすくなるが、必ずしもそのような密接な関係になれるわけではない。

拠点が離れていたり、組織が異なったり、面識すらないときもある。

そのような、チーム外の相手に対してレビューを行うとき、あるいはチーム外の誰かにレビューを依頼するとき、できるかぎり誰も傷つくことなく(あるいは傷ついてもすぐに立ち直り)、遠慮なく指摘できることでの確に指摘内容が伝わり、関係者全員が目的を達成できたことを喜べるために、どのような工夫をするとよいのかを知りたい。

(特にレビューを受ける側の工夫を聞くことが無いので、受ける側ができる事を知りたい。)

誤字、誤記、罫線の切れなど体裁ばかりの指摘が多く、肝心の機能的な指摘が少ない場合が多い。機能については、レビューではなく実施側や受け入れ時に見つかる。

細かくみると時間がかかる。

行き当たりばったりの指摘で、本題からそれていく。

レビューを繰り返すうちに落としどころが見つからず、ずるずるレビューの回数だけが増えていき、どこでけりをつけていいのかいまいち分からぬときがある。(全ての指摘事項を組み込むと煩雑になり、自分の判断でレビューコメントを採用、不採用としているが、それにレビューの意味はあるのか…?)

WEB上でレビュー用の掲示板のようなもの(Sharepoint)があり、レビューコメントを各自記載していくのだが、コメントがないのは見てくれた結果指摘がないのか、見てくれていないのかが判別付かない。だからといって名指しで催促するのも…。

レビューの重要性は理解しているが、お客様の理解が取れずレビューにかかる工数(費用)を確保できない。

"レビュー工数に対する指摘の件数が少ないということです。簡易的な保守が多く、すべてレビューしていますが、簡易なことから指摘する内容がないといった現状です。指摘がない場合、リーダから「本当に確認できているか」といった声がでてきています。"

・チェックリストを使用して、レビューを行っておりますが、レビュー漏れが発生してしまいます。時間をかける割には、些末な指摘しか出来ず単なる儀式になってしまっている。短時間で効果的なレビュをするためには?

システムの内容に関らず、一般的に押さえておくべきレビュの観点はあるのか?

レビュされる方はテクニカルな指摘をされることを期待しているが、レビュアは利用者目線の指摘をするなど、ギャップがある場合がある。様々な視点でレビュするためには、参加するメンバを増やすなどしかないのか?→限られた時間と要員で現実的には厳しいと思うので…

「現存するエンジニアが急に成長するとは考えられない。」ので、組織、ルール、手順書などで、有効な活動を伺いたい。設計書やテスト仕様書に記載されている項目については指摘できるが、記載されていない（記載漏れ）項目については指摘ができず、後の障害発生原因として「設計漏れ、テスト漏れ」が上げられる。「記載されていない」項目についてレビューで指摘できるようになるにはどのような活動が考えられるか？

システム構築当時のメンバーや、システム改修した前任者が不在の場合、レビューの指摘に対して担当者から「前回同様の記載としています。」という回答が良く出ます。前回と今回ではプロジェクトの内容が違うという事を説明して、やり直しを指示しているが、何か有効な先手はないか？"

適切なレビューのためには、レビューが的確に作業指示をし、レビュー前にレビュー観点・箇所を明確にしていることが必要と考えています。（指示していくてもレビューの理解度・設計・レビュースキルは重要）これができないと、レビューの質が確保できない認識です。この考えは、ありますか？

レビューに費やす工数は、品質管理の一つの指標と思います。開発工程における適切なレビュー工数比率（レビュー工数／計画工数）はどれくらいが適切と安達さんはお考えですか？

指標値に基づいた品質評価を実施しているが、そもそも指標値の収集・分析活動が組織として定着していない。指標値は少数の大規模件名の結果から作成されたものであり、小規模件名に対する妥当性が不明。

指標値から外れるものに対する品質の妥当性確認が上手くいかない。（担当者に確認して、それなりの理由は付けられるが、後工程で欠陥が発見される）

レビューが進捗のボトルネックになりやすい。理由としては、以下の2点を考えている。
・レビューに耐えられない成果品がレビューにかかっている
・レビュー観点が定まっておらず仔細な指摘が多い

○属人生が高い

→本当のレビューができる要員(業務有識者)が少ない

→業務有識者が参加していない形骸化したレビューが散見される(業務有識者は多忙、自身の作業も持っている...時間が取れない)

○レビュー効果が共有されていない

→上手にレビューを実施した結果得られた『美味しいところ』を見る化・共有していない

→実のあるレビュー実施の重要性の認識が広まらない(やらされ感、形骸化)

1. レビュースキルの測定 レビューポイントをチェックリストとして挙げているが、特に業務に入り込んだレビューとなると なかなかレビュースキルの評価を定量的に計測できない。(レビューの適任者の選定が感に頼っている) 定量的な計測はないでしょうか。

2. ソフトウェアのレビュー技法に関するバイブル的な書籍を知りたい (レビューをきちんとできる人を増やすため)

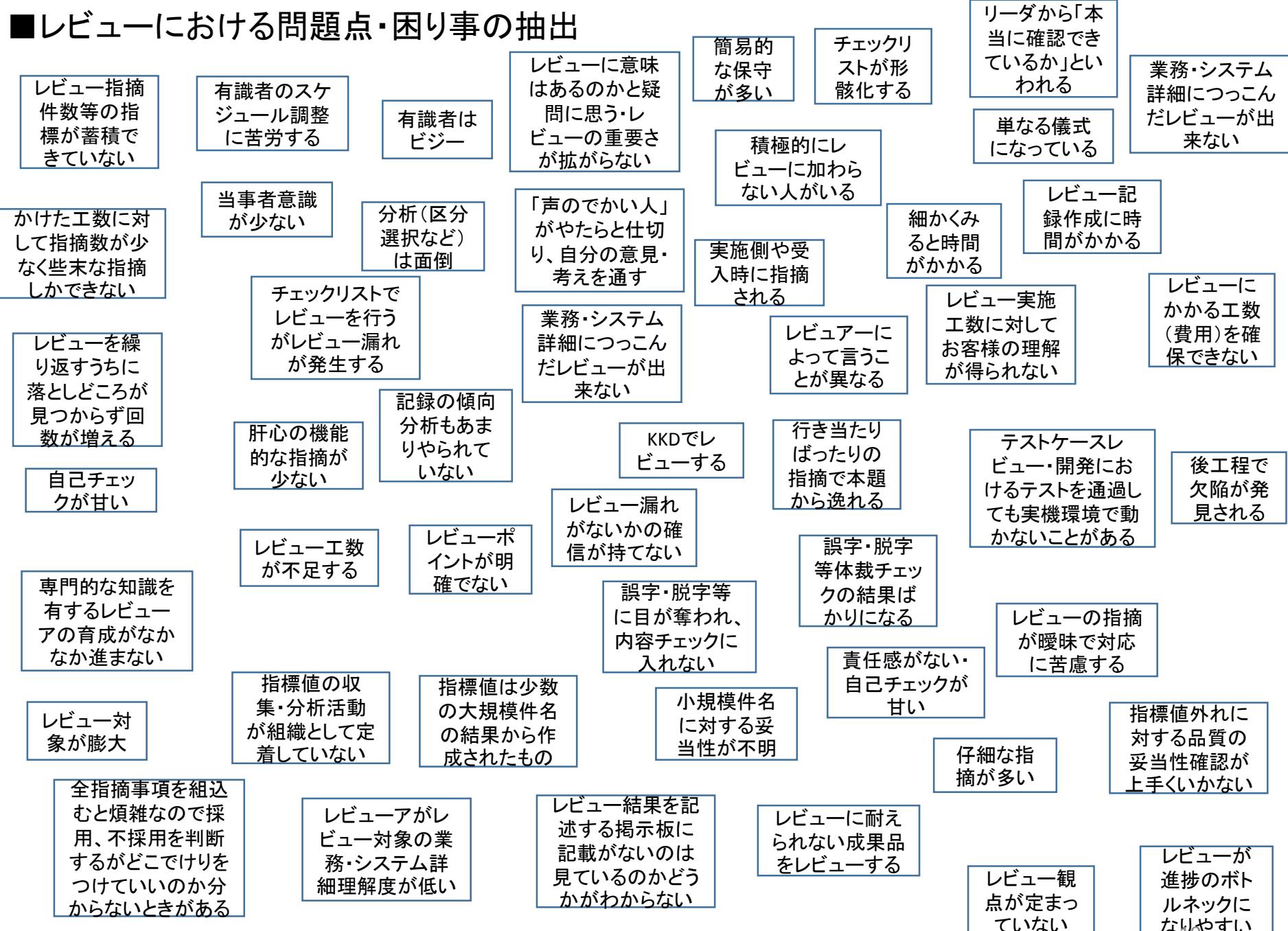
①レビューアが効果的なレビューのやり方を知らないため、経験と勘によるレビューになっている。レビューアの体調、気分、作業状況により、レビューのしかたが変わらるような気がする。わかつていたのに欠陥を見逃していたこともある。

②レビューされる側に成果物に対する責任感が低い場合がある。レビューアに指摘してもらえばよいと思ってわからないところを残したままの成果物を持ってくることがある。

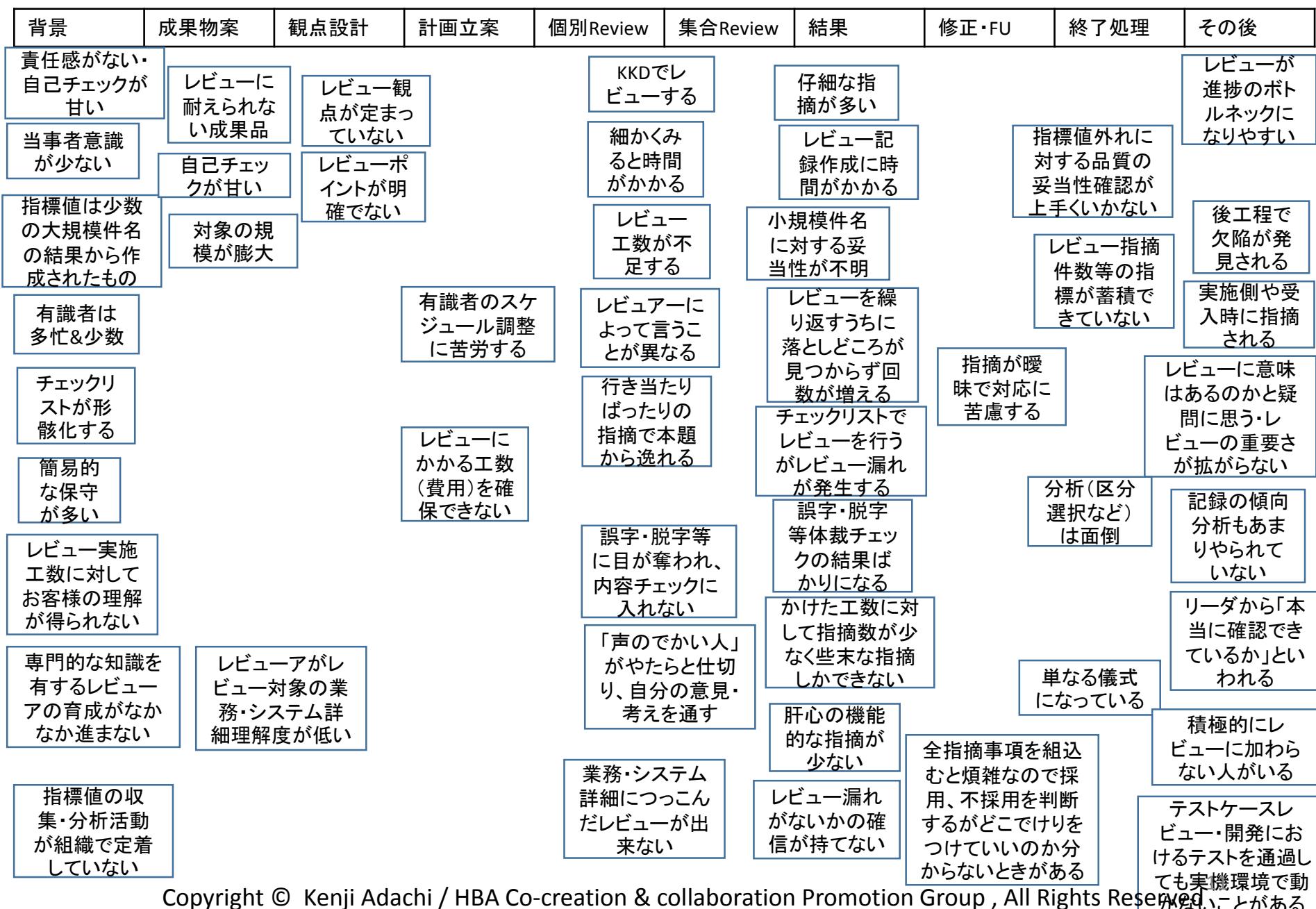
注意:アンケート結果をそのまま貼り付けているため、誤字等があつてもそのままとなっているところがあります。ご了承ください。

システムズアプローチによる 構造分析

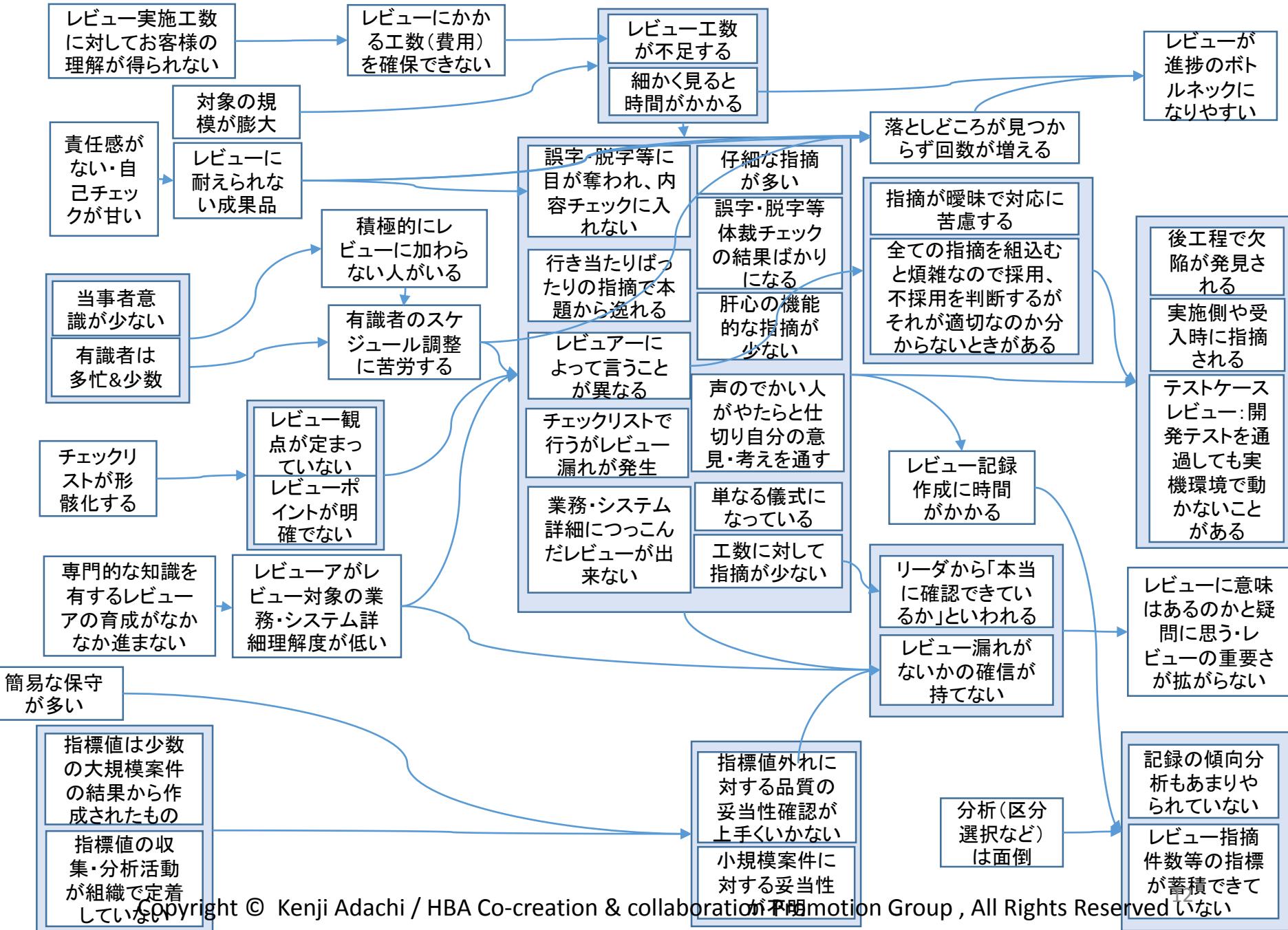
■レビューにおける問題点・困り事の抽出



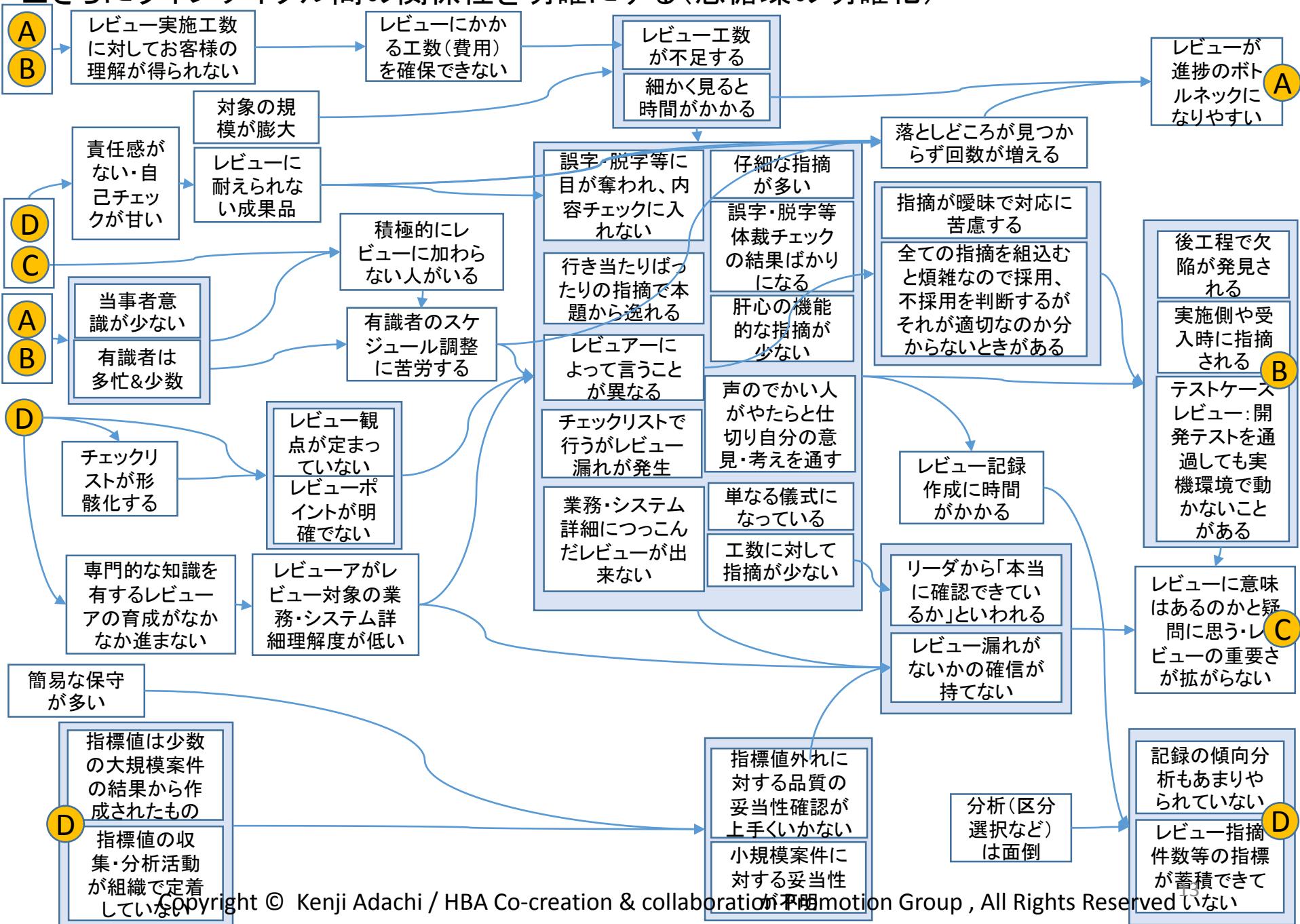
■レビュープロセス上の発生タイミングに割付

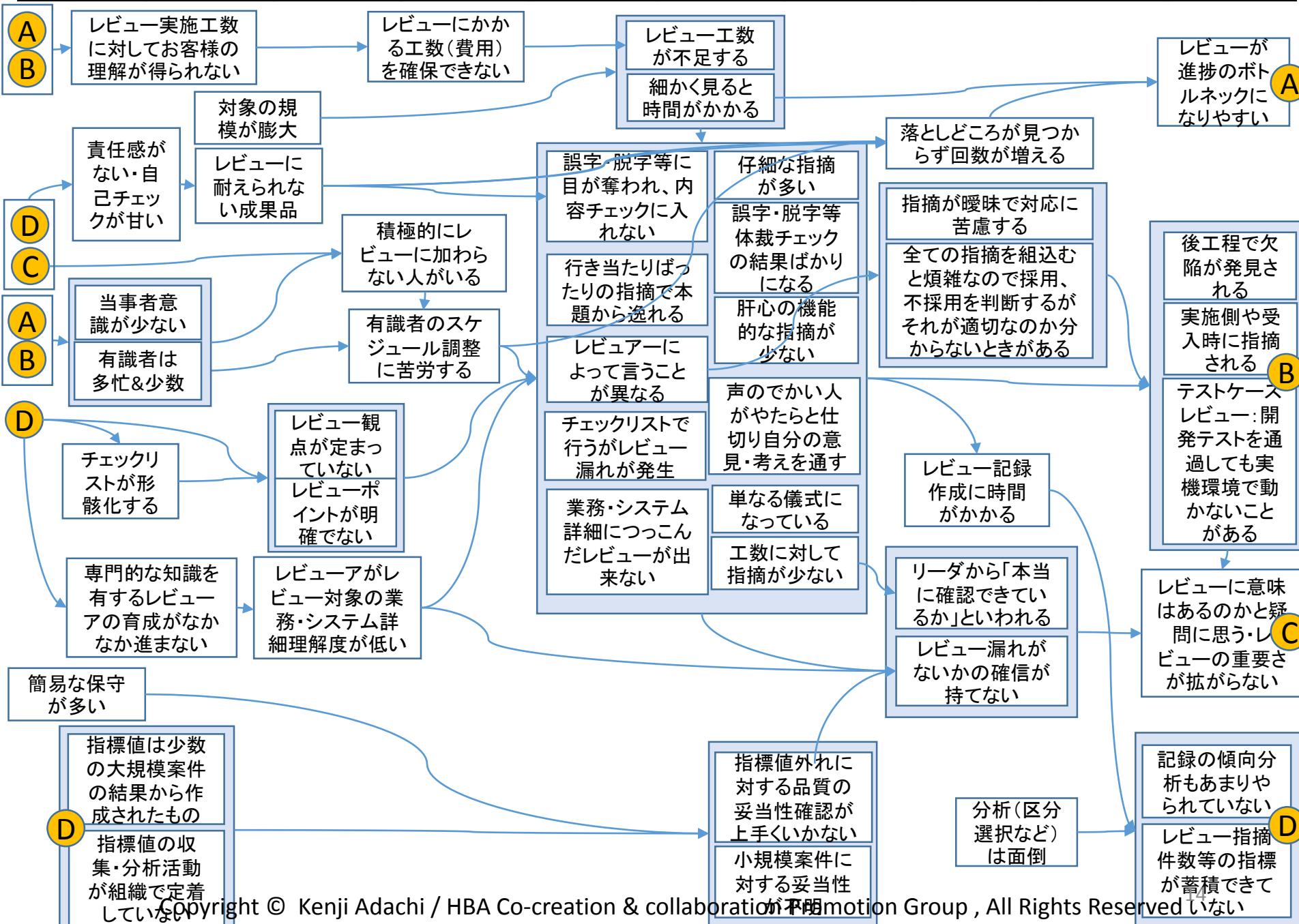


■要素間の関係性を分析(因果関係があれば→で結ぶ)

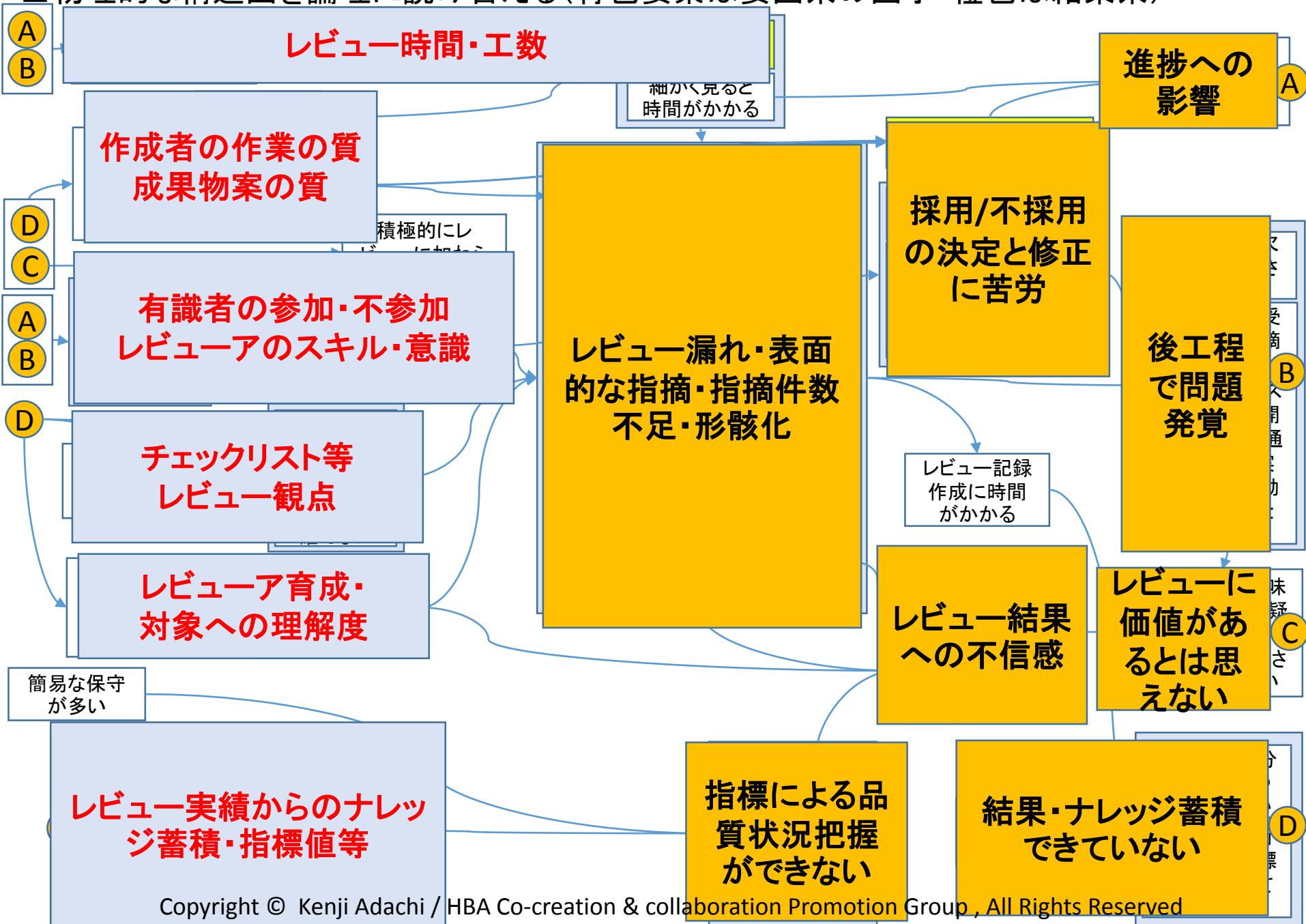


■さらにライフサイクル間の関係性を明確にする(悪循環の明確化)

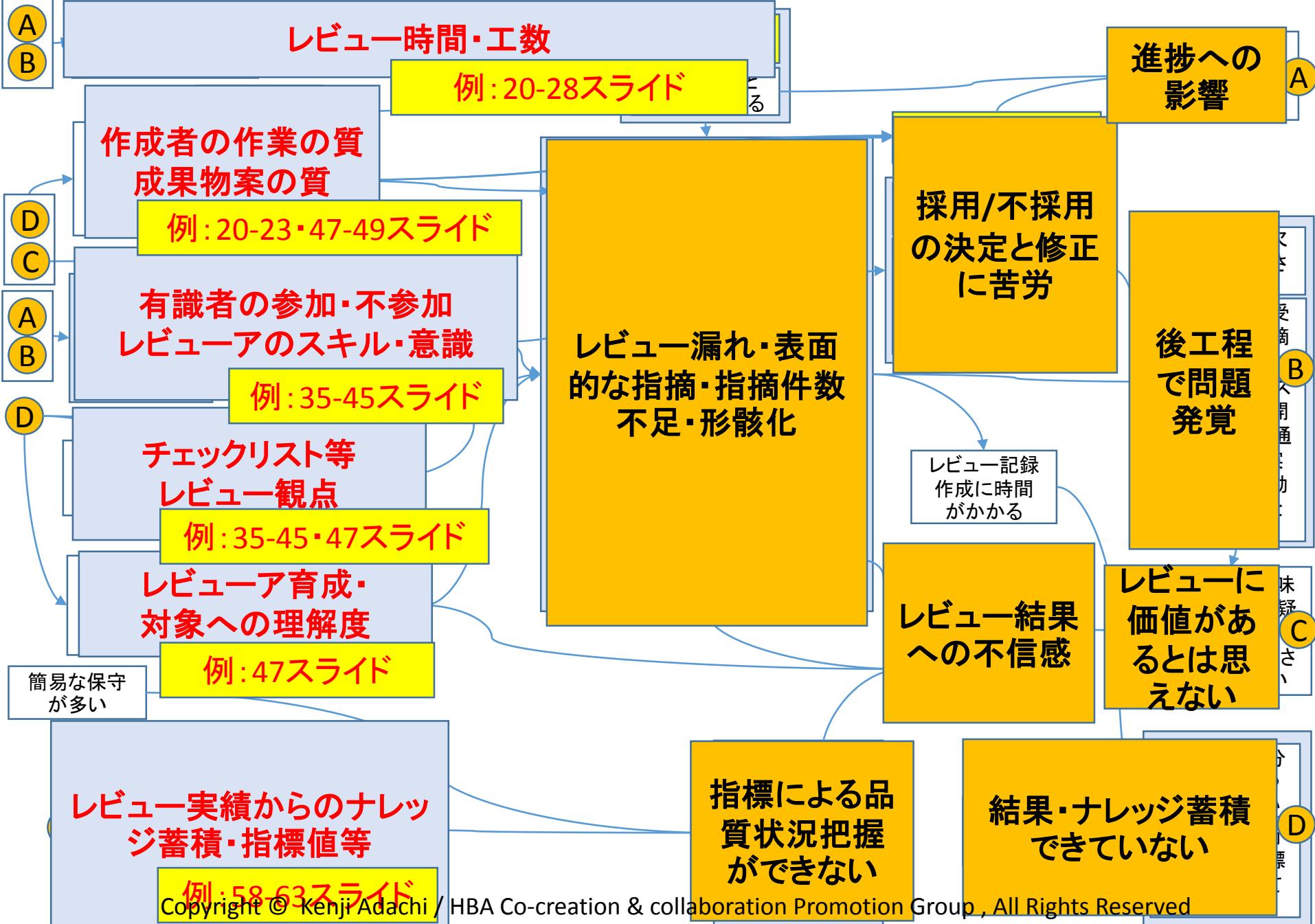




■物理的な構造図を論理に読み替える(青色要素は要因系の因子・橙色は結果系)



■青色要素:要因系の因子への解答例として本編スライドの参照ページ数を付与



アンケート結果から抽出した “知りたいこと”

知りたいこと

黄色オブジェクト=解答例としての本編スライドの参照ページ数

- (1)短時間で効果的なレビューをするためには? 例:24・28スライド
- (2)有識者でなくとも、それなりの人で出来る様な方法はないのか?
- (3)レビュー者がレビュー対象の知識があまりない場合のレビュー方法例:33-49スライド
- (4)社内開発環境でシステムテストがOKであっても、実機環境で動作しないことが多い。プログラムの不具合ではなく環境不具合であることがほとんど。(テストケースレビュー) →湯本さんに聞いてね!
- (5)システムの内容に問らず、一般的に押さえておくべきレビューの観点はあるのか? →あるがそれだけではない&対象により深みは異なる
- (6)「現存するエンジニアが急に成長するとは考えられない。」ので、組織、ルール、手順書などで、有効な活動を伺いたい。設計書やテスト仕様書に記載されている項目については指摘できるが、記載されていない(記載漏れ)項目については指摘ができず、後の障害発生原因として「設計漏れ、テスト漏れ」が上げられる。「記載されていない」項目についてレビューで指摘できるようになるにはどのような活動が考えられるか?例:46スライド

知りたいこと

黄色オブジェクト=解答例としての本編スライドの参照ページ数

(7)レビューされる方はテクニカルな指摘をされることを期待しているが、レビューは利用者目線の指摘をするなど、ギャップがある場合がある。様々な視点でレビューするためには、参加するメンバを増やすなどしかないので？→限られた時間と要員で現実的には厳しいと思うので…

例:33～49スライド

(8)システム構築当時のメンバーや、システム改修した前任者が不在の場合、レビューの指摘に対して担当者から「前回同様の記載としています。」という回答が良く出ます。前回と今回ではプロジェクトの内容が違うという事を説明して、やり直しを指示しているが、何か有効な先手はないか？

例:70スライド

(9)適切なレビューのためには、レビューが的確に作業指示をし、レビュー前にレビュー観点・箇所を明確にしていることが必要と考えています。(指示していくてもレビューの理解度・設計・レビュースキルは重要)これができないと、レビューの質が確保できない認識です。この考えは、ありますか？

例:70スライド(作成者)

(10)レビューに費やす工数は、品質管理の一つの指標だと思います。開発工程における適切なレビュー工数比率(レビュー工数／計画工数)はどれくらいが適切と安達さんはお考えですか？

例:62スライド

知りたいこと

黄色オブジェクト=解答例としての本編スライドの参照ページ数

(11)チーム外の相手に対してレビューを行うとき、あるいはチーム外の誰かにレビューを依頼するとき、できるかぎり誰も傷つくことなく(あるいは傷ついてもすぐに立ち直り)、遠慮なく指摘できることで的確に指摘内容が伝わり、関係者全員が目的を達成できたことを喜べるために、どのような工夫をするとよいのかを知りたい。(特にレビューを受ける側の工夫を聞くことが無いので、受ける側ができるることを知りたい。 **例:50~54スライド**)

(12)レビュースキルの測定:レビューポイントをチェックリストとして挙げているが、特に業務に入り込んだレビューとなるとなかなかレビュースキルの評価を定量的に計測できない。(レビューの適任者の選定が感に頼っている)定量的な計測はないでしょうか。

例:47スライド→仮説力と要約力など

(13)ソフトウェアのレビュー技法に関するバイブル的な書籍を知りたい。(レビューをきちんとできる人を増やすため) **例:87スライド**