



# あなたに捧げる～

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  
【導入時：改善計画立案編】

## リターンズ

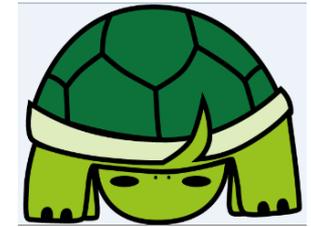
株式会社HBA 安達賢二

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

[adachi@hba.co.jp](mailto:adachi@hba.co.jp)

安達 賢二 (あだち けんじ) [adachi@hba.co.jp](mailto: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/>



ものごとの見かたを変える

### 【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 (JaSST担当)  
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  
SEA (ソフトウェア技術者協会) 幹事・北海道支部メンバー  
SS (ソフトウェア・シンポジウム) プログラム委員  
第33-34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  
テストプロセス改善研究会幽霊メンバー など



きたのしろくま  
@kitanosirokuma

# この発表で行うこと

- プロセス改善モデル、TPI NEXTってどんなもの？

- ✓ プロセス(改善)モデルとは何者なのかを知ろう！

- ✓ TPI NEXTの内容を一緒に眺めてみよう！

- ✓ SaPID流オリジナルクラスタセット構築(我流カスタマイズ)事例(書籍の通りではないのであしからず)



- プロセス改善ってどうやればうまくいく？

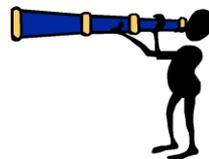
- ✓ このアプローチにはどんな意味や価値があるの？

- ✓ どうしてそんなやり方なの？



- 発表内容のここが？あそこが？

- ✓ 質疑応答(時間がある限り)



# 諸注意



- この発表内容は、TPI NEXTの公式トレーニングも受けていない発表者の**独断と偏見、経験則、バイアス**が思い切り含まれた“**我流**”のものです。

- ✓ 過不足、あるいはみなさんの背景には当てはまらない内容が多分に存在する可能性がありますので、そのまま信用する必要はありません。



- ✓ 信用できそう、活用できる、と思えば使う、わからない、ピンとこない、信用ならないなら捨てる、あるいは自ら組み立て直す、湯本さんなどの有識者に聞く、でOKです！





お願い

わからないことは放置しないでね

- アンケートに「〇〇がわからなかった」と記載することがないように、わからないことは可能な限り質問して疑問を解消してください。私ができる範囲で質問にお答えします。
- 2日目終了時まで会場にはいると思います。当セッション終了後でも構いませんのでいつでも声をかけてください。



# あなたに捧げる～・・・の「あなた」とは？ (裏話)

- 実は前回発表に対するアンケートの中に

「なんだよ、TPI NEXTの内容が  
ぜんぜんわからなかったじゃん」

というコメントが1件ありました。

その方のために当セッションを捧げたいと思います。

※プロセス改善がうまくいっている人、実はうまく  
いっていないけどやったふりで凌げる方、TPI NEXT  
が理解できている方は対象外のセッションです(笑)

# コンテンツ

- プロセス改善とプロセス改善モデル
- TPI NEXTの特徴
- JaSST'18東京発表事例(拡張版)  
別解例を含む



今日の本丸  
はこれ！

# プロセス改善と プロセス改善モデル

# プロセス改善の位置づけ

改善

製品品質改善

製品を直接直して製品の品質をよくなる

プロセス改善

仕事の仕方を見直してその後の製品品質をよくなる

問題、課題ベースの改善

プロセス改善モデルベースの改善

# 相互補完する関係

当発表内容は両方のハイブリッド型アプローチ

|             | 問題・課題<br>ベースの改善  | プロセス改善モデル<br>ベースの改善  |
|-------------|--|--|
| アプローチの原理    | 問題・課題発生の変因(原因)除去   | プロセスのあるべき姿とのギャップ解消   |
| 典型的な手法      | なぜなぜ分析・効果図式など  | CMMI・Automotive SPICE・TPI NEXTなど   |
| アプローチが持つリスク | <ul style="list-style-type: none"><li>・視野が狭くなり、局所対応に終始する</li><li>・いたちごっこになる</li><li>・結果の良し悪しは分析担当者のスキル、管理者の人間性等に依存する</li><li>・個人攻撃や魔女狩りになりやすい</li><li>・個別論になりやすく組織展開が困難になる</li></ul> | <ul style="list-style-type: none"><li>・評価、改善プラン立案の専門性障壁が高い</li><li>・対応工数、期間、費用が大きくなる</li><li>・トップダウンに偏った場合はモチベーション維持が困難になる</li><li>・適合性を目的化すると効果が不明になる、効果が実感できなくなる</li></ul> |



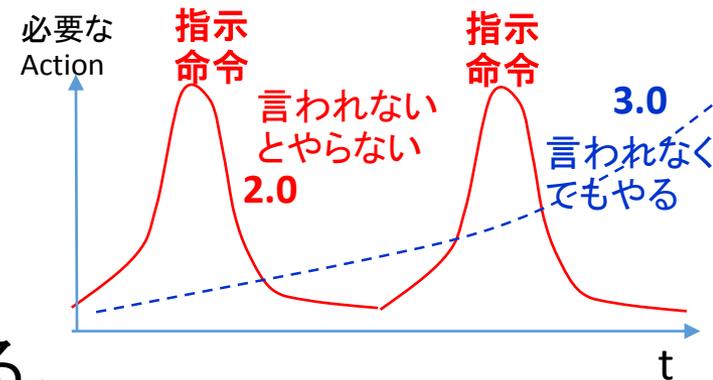
# 「モチベーション3.0」 ダニエル・ピンク

• **モチベーション2.0**(アメとムチなど外発的動機付け)から**モチベーション3.0**(内発的動機付け)という2つの動機付けへ

• モチベーション2.0の管理で報酬を用意すると、管理される側はその報酬のために短期的にやるだけになり、「**自律性(オートノミー)**」を失う。

• **アメとムチの致命的な7つの欠陥**

1. 内発的動機づけを失わせる。
2. かえって成果が上がらなくなる。
3. 創造性をむしばむ。
4. 好ましい言動への意欲を失わせる。
5. ごまかしや近道、倫理に反する行為を助長する。
6. 依存性がある。
7. 短絡的思考を助長する。



# プロセス(改善)モデルって？

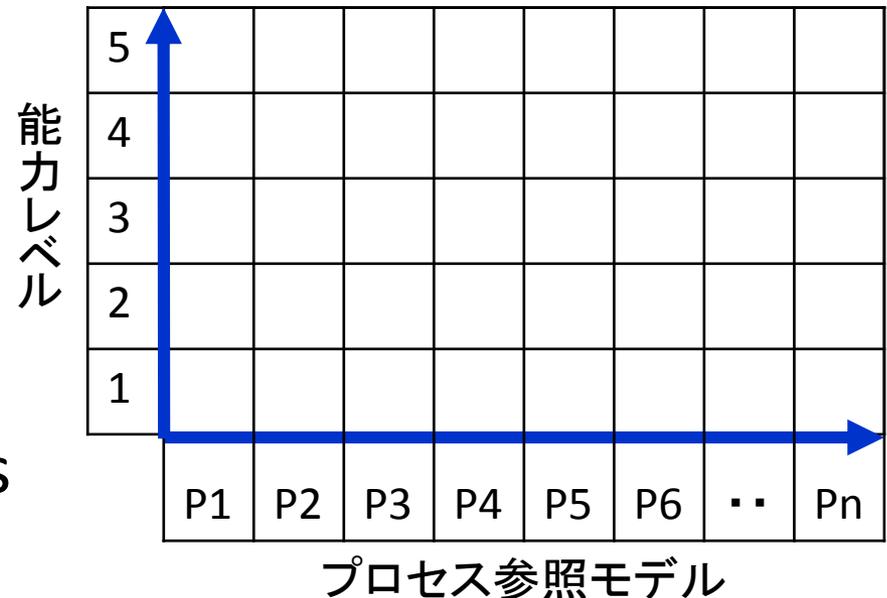
プロセス(改善)モデルが目指すこと

- チーム、組織でProject Managementの原則やSoftware Engineering等を適用、実践して効果と効率の両面を兼ね備えたSoftwareをできるだけ安定的に開発(TPI NEXTの場合はテスト)するために必要な共通(汎用)的なプロセス、プラクティスの実施要件を体系化し、モデル化したもの。

# プロセス(改善)モデルはたくさんある

## 以下は一部の事例

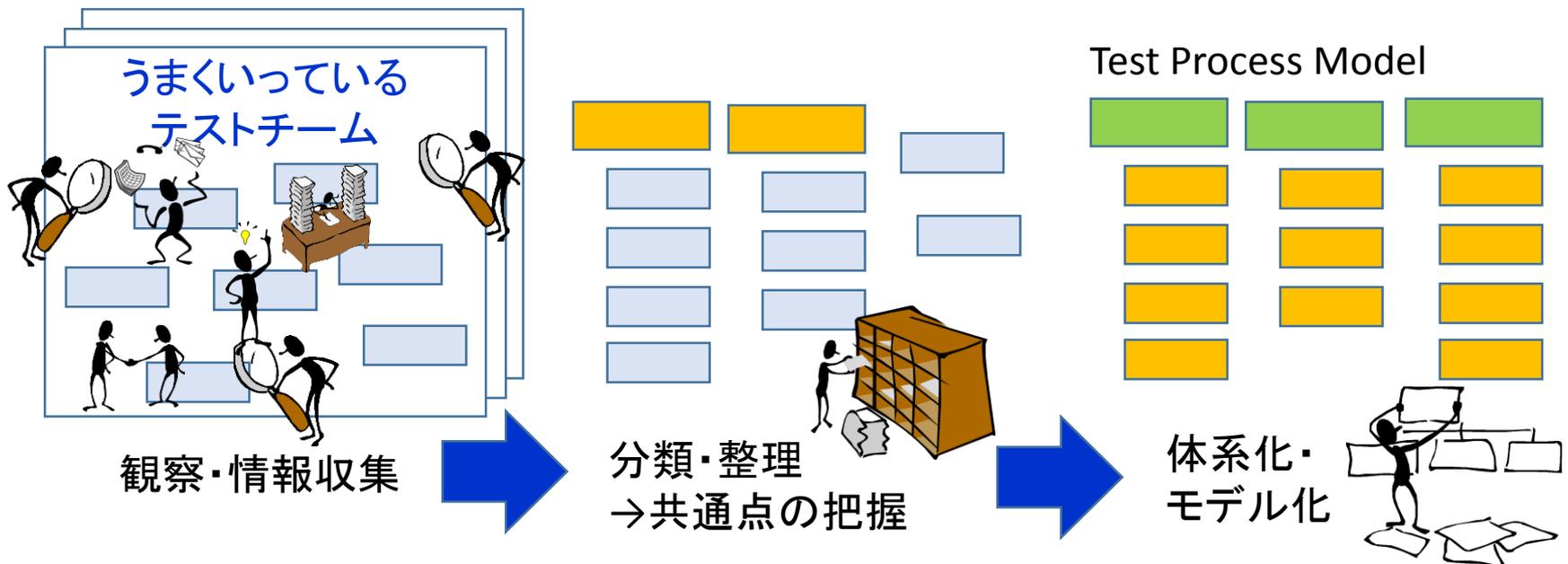
- Software CMM→SPICE→ISO15504→CMMI
- ISO9001→ISO9000-3・TickIT→ISO90003
- (SPICE)→Automotive SPICE・SPICE for Space
- (CMMI)→TMMI
- SPEAK→SPEAK IPA
- SPINACH→SPINA3CH
- TPI→TPI NEXT
- (ISO15504)→ISO33000's



# プロセス(改善)モデルって？

## プロセス(改善)モデルの作り方

- ソフトウェア開発やテストで成果を上げている組織やチームが実践していることを観察し、その共通的な特徴を体系化、モデル化する。



# プロセス(改善)モデルって？

## プロセス(改善)モデルの内容(1)

- ソフトウェア開発やテストの過程、結果、成果に発生しがちな困り事、解決すべき問題点、乗り越えなければならない課題などの発生を緩和、解決、解消するために必要な施策の、または施策要件の集合体

TPI NEXT 11.テストウェア管理

11E2: テストケースと要件のトレーサビリティが確保されている

左記が実践されない場合、何が起きるか？

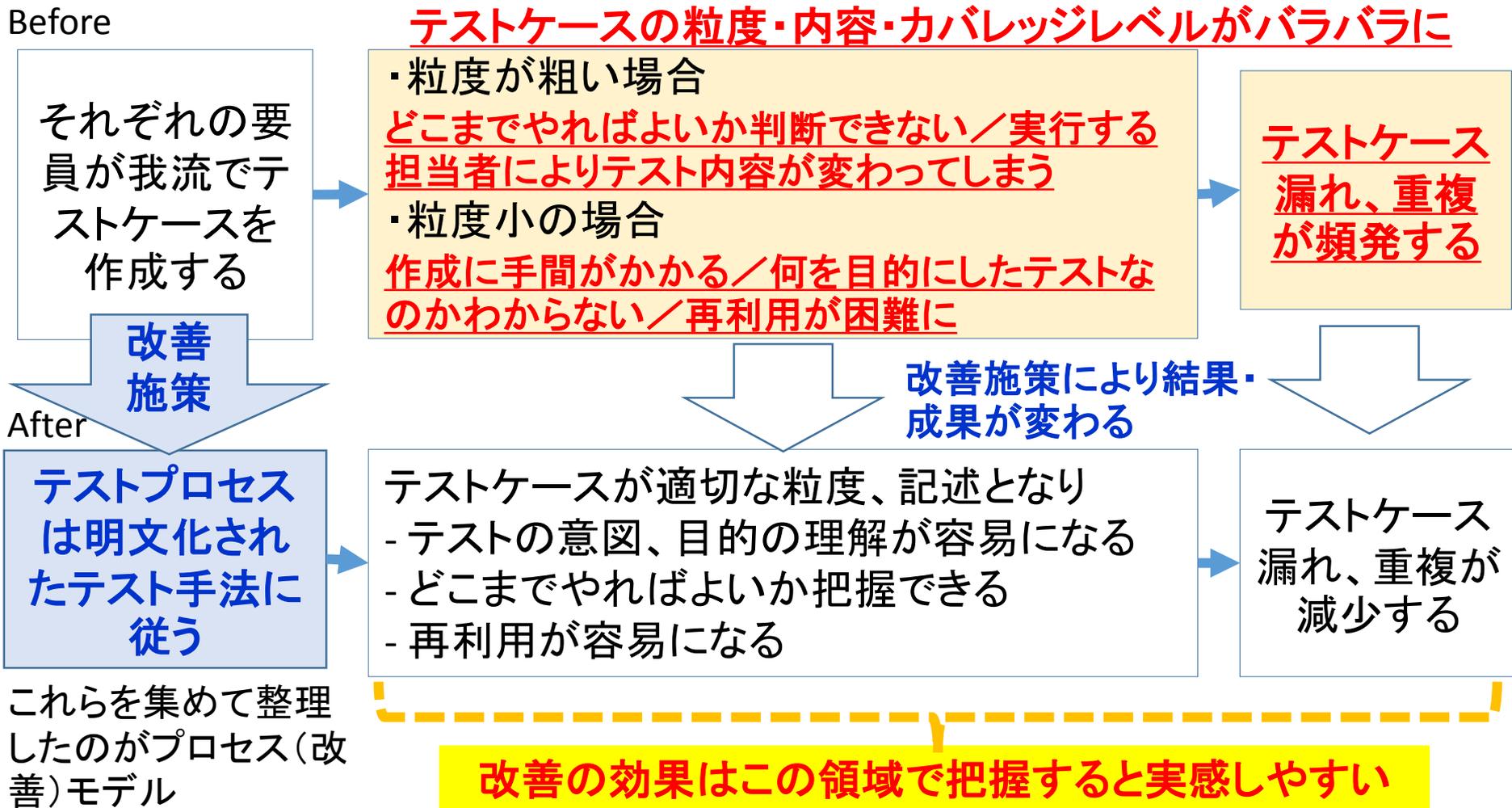


解決したい／解決できる問題



# 困り事・問題・課題と施策の関係

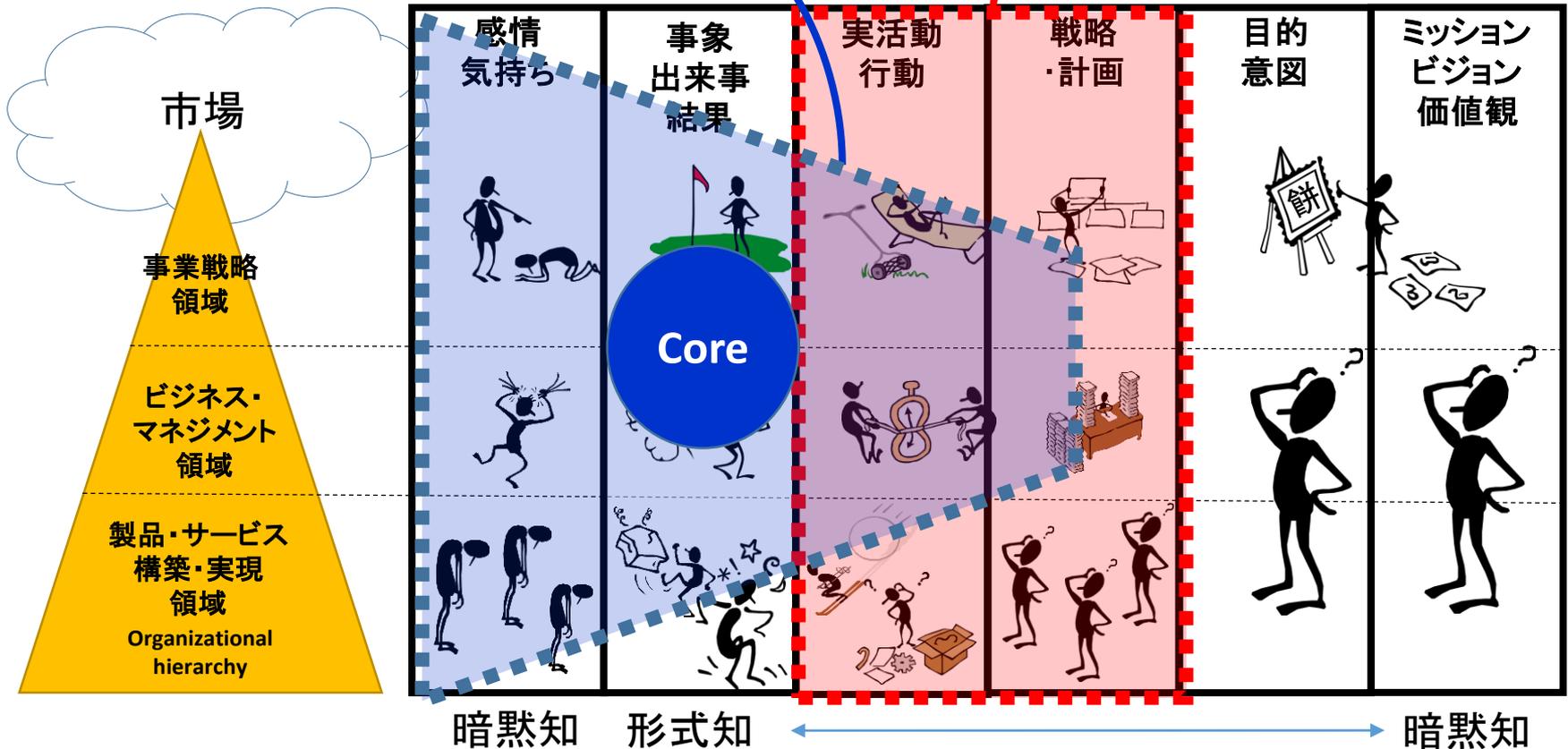
改善の効果をどこ(何)で把握するか？



# SCOPE・ターゲットの違い

SaPIDの当初  
SCOPE

プロセスモデルの  
SCOPE



# プロセス(改善)モデルって？

## プロセス(改善)モデルの内容(2)

- “モデル”なので、一般化している、大事なこと以外は端折っていることに注意！

＝モデルは(可能な限りどこでも使えるように考慮しているが)万能ではない

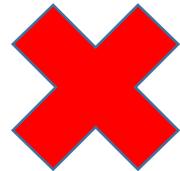
- ✓ 自らのコンテキストにとって大事なことが欠けているなら足し込む、重要ではないことが書かれているなら除外すればよい
- ✓ 適合性認証制度を伴う場合も多い：一律従うこととどの程度の価値があるのかは？？？→自ら決める必要がある

# プロセス(改善)モデルって？

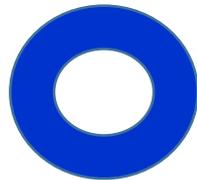
## プロセス(改善)モデルの内容(3)

- わかりやすさを優先しているため、字面だけで適用すると効果が薄い形式対応になる可能性もある。  
= 背後に隠れた前提条件等を読み解いて適用する

14C1: テスト  
ケースを論理レ  
ベルで記録する



テストを実施したあとで記録を作ればいいんだ！(それさえやればあとは何もしなくてよい)

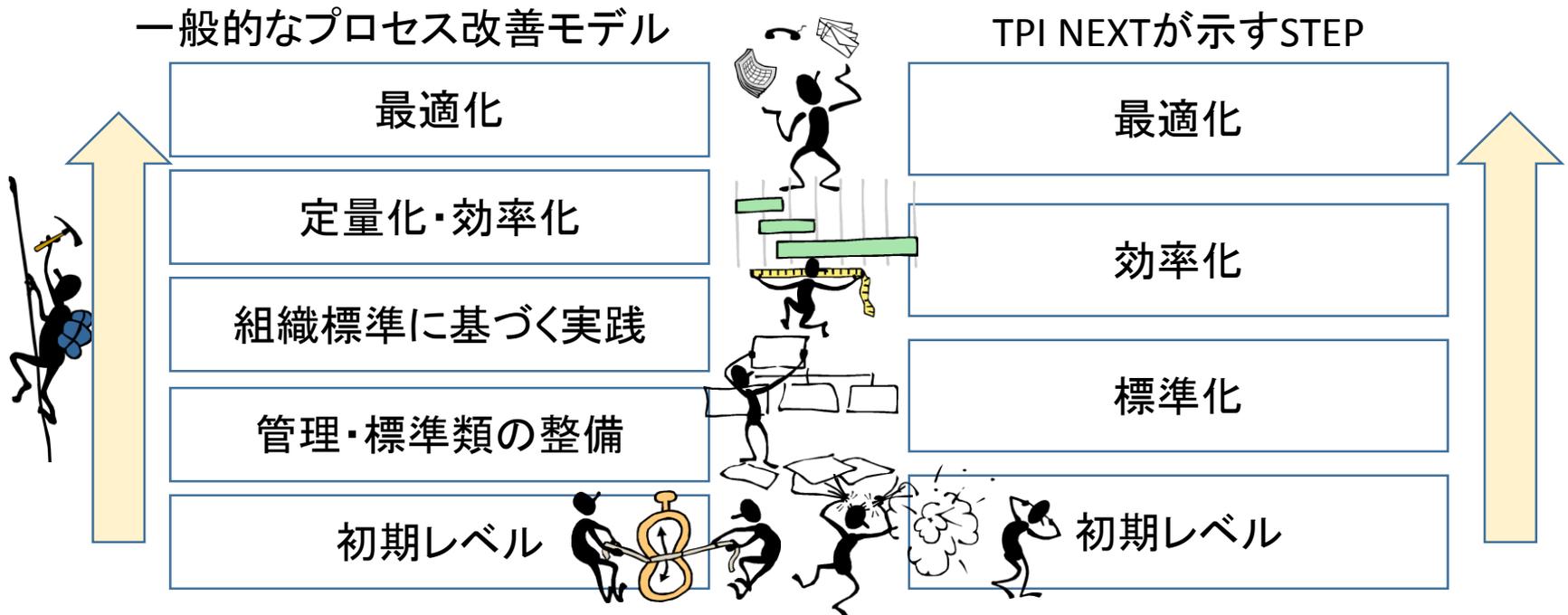


テストケース設計は論理レベルの設計～具体的ケースに落とし込む設計までの幅があり、自らのコンテキストに適切な粒度、内容で設計、記述し(そしてその結果、作業効率と再利用性が高まる)、実践することが必要

# プロセス(改善)モデルって？

## プロセス(改善)モデルを使った階段の登り方(1)

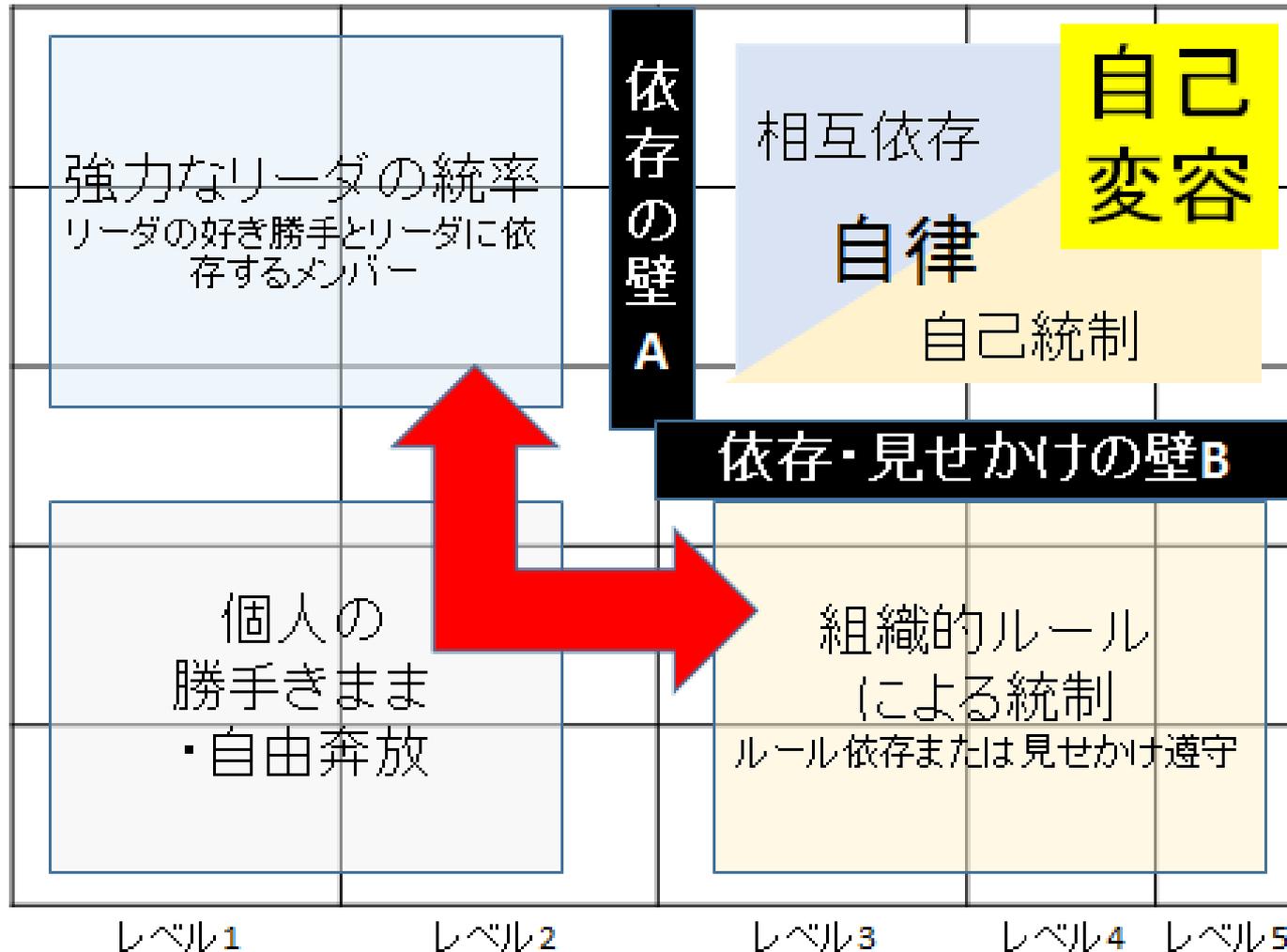
- Software CMMの基本構造とその後の実践検証にて位置づけられた階段の登り方が源流。
- 下位レベルの実践が上位レベルの実践を支える基盤となる構造。



# 山の登り方はいろいろある

高い山に登るのは簡単ではない

有効性・  
継続的成果



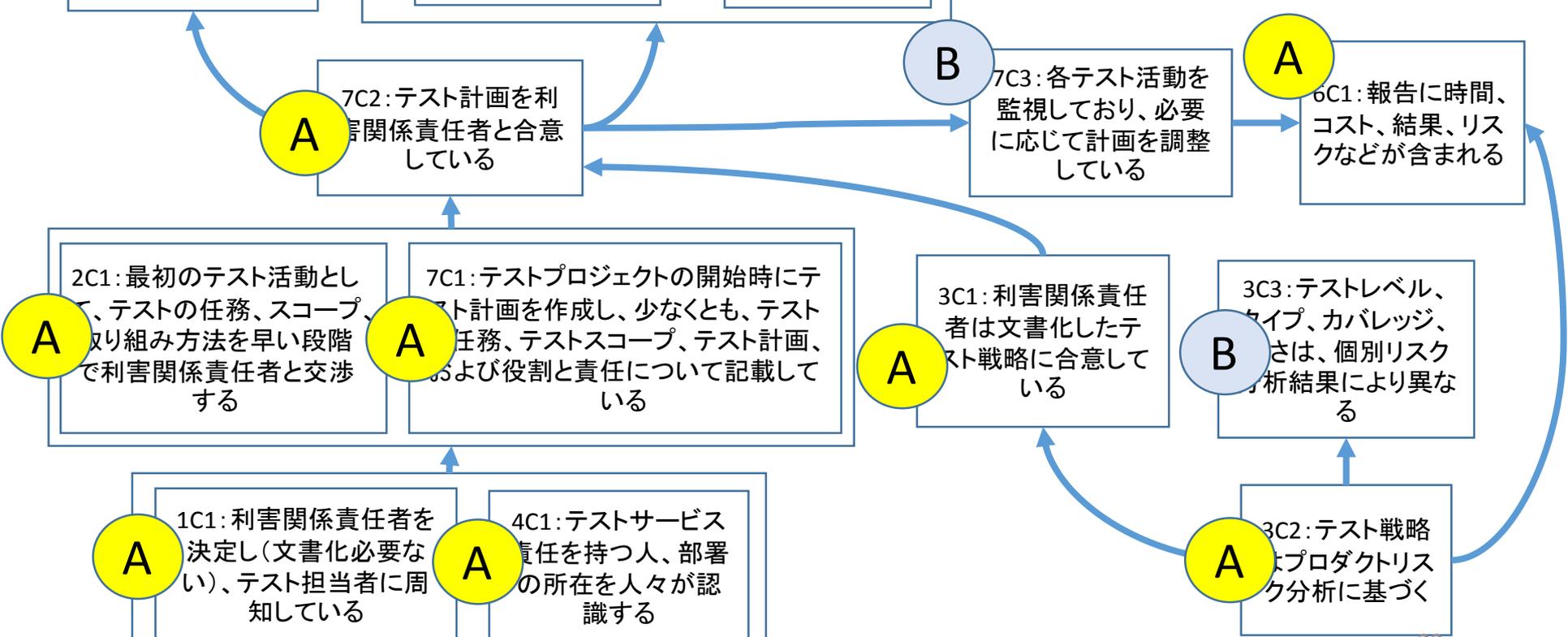
プロセス能力・  
成熟度



# プロセス(改善)モデルって？

## プロセス(改善)モデルを使った階段の登り方(2)

**B** 7 関者  
• 個々のプラクティスには依存関係が存在するがその詳細は書かれていない。→自ら読み解くしかない！



# プロセス(改善)モデルって？

## プロセス(改善)モデルを使った階段の登り方(3)

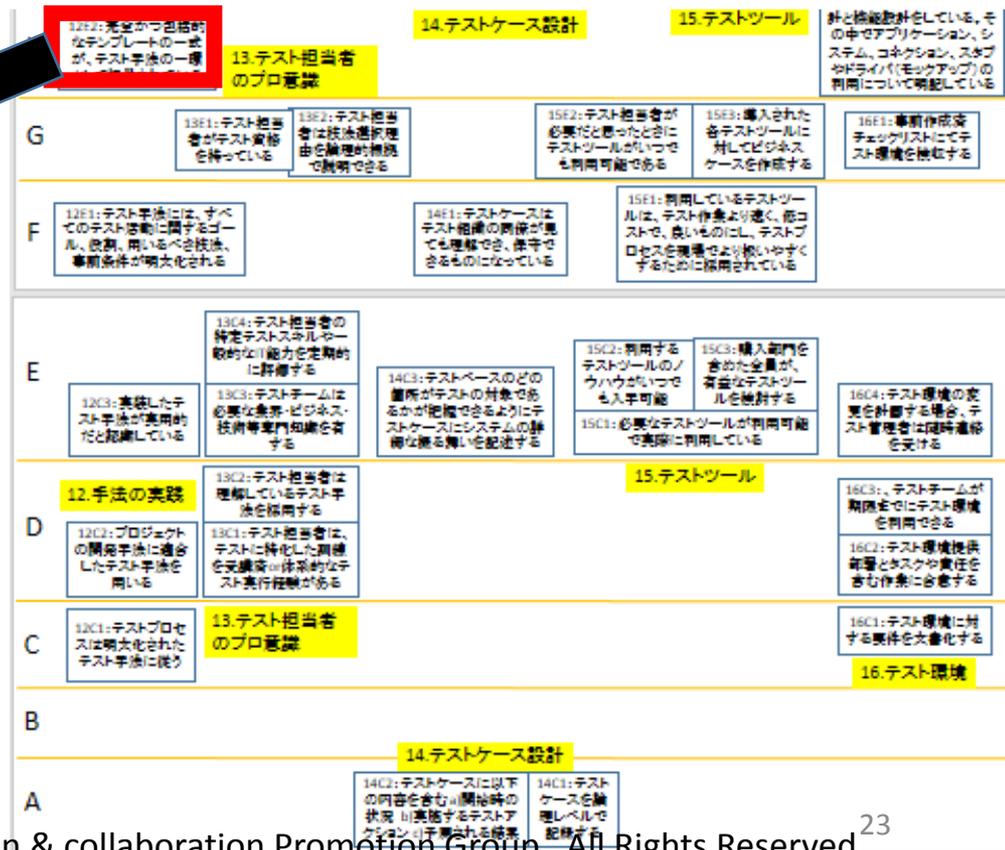
- プラクティスはそのレベル(段階)のスナップショットである。→隠れたプラクティスは自ら導出するしかない。

プロセスモデルには書かれていないが必要な実践事項の例

12E2: 完全かつ包括的なテンプレートの一式が、テスト手法の一環として提供されている

不足しがち、抜けがちなテストタイプ、テストケースを特定して補完する

基本的なテストケースを作成するためのテンプレートが提供されている



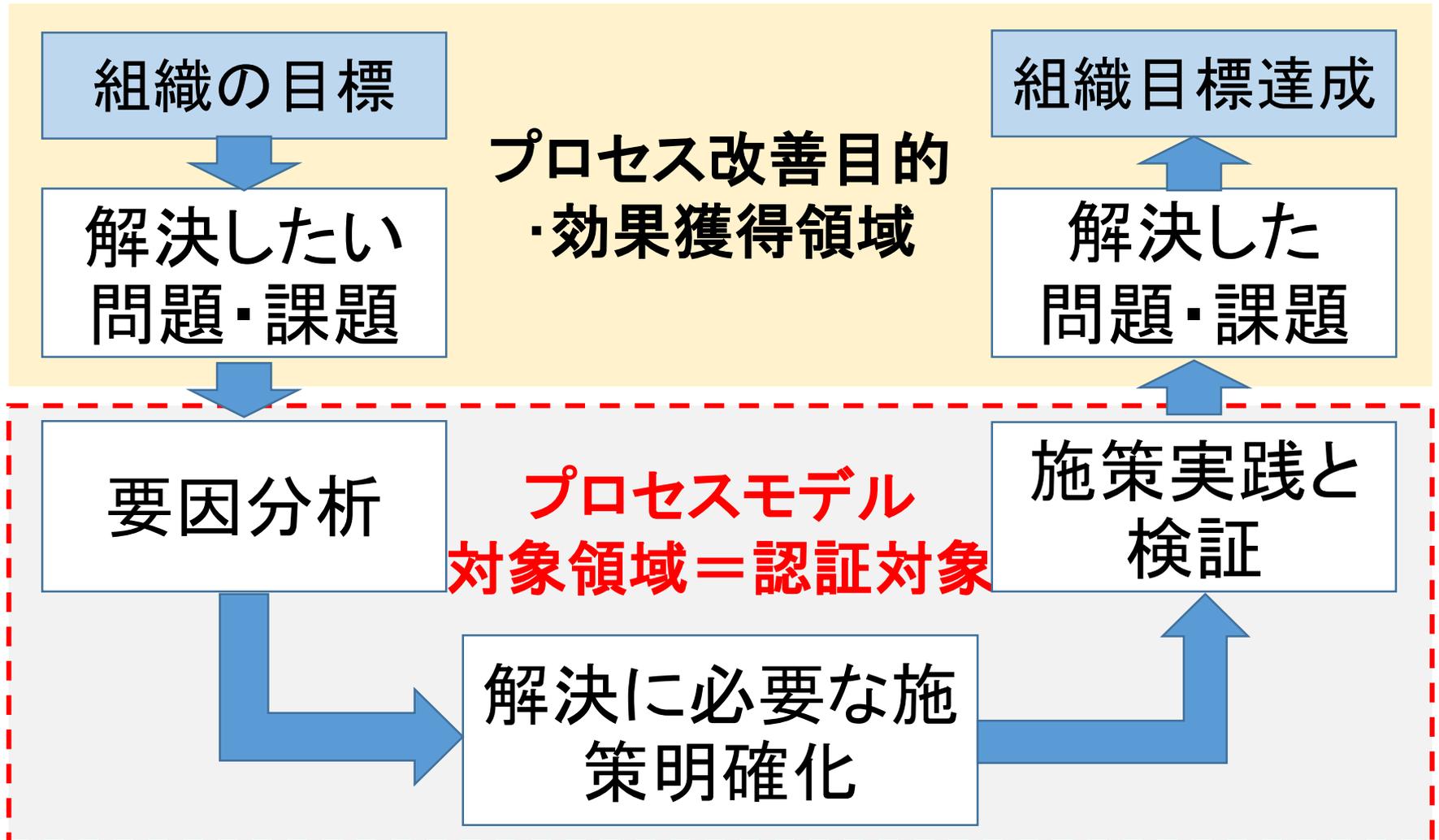
# プロセス(改善)モデルって？

## 適合性認証制度

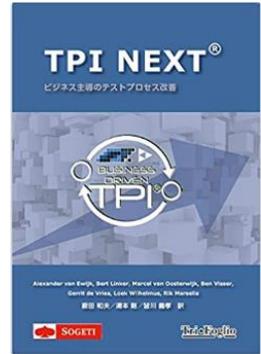
- プロセスに課せられる要求事項を明らかにしたのがプロセスモデルである、という言い方も出来る。
- その特徴を活かしてプロセス要求事項に適合しているかを認証する制度が伴っていることが多い。
- この制度で認証しているのは「プロセス要求事項を満足しているか＝適合しているか」だけであり、その**結果や成果を認めているものではないことに注意**。

効果がほとんどない(むしろ被害が多い)形式的な対応  
であっても認証されてしまう可能性がある

# プロセスモデルの対象領域



# TPI NEXTの特徴



# TPI NEXT

## オランダのSogeti社が提案したテストプロセス改善モデル

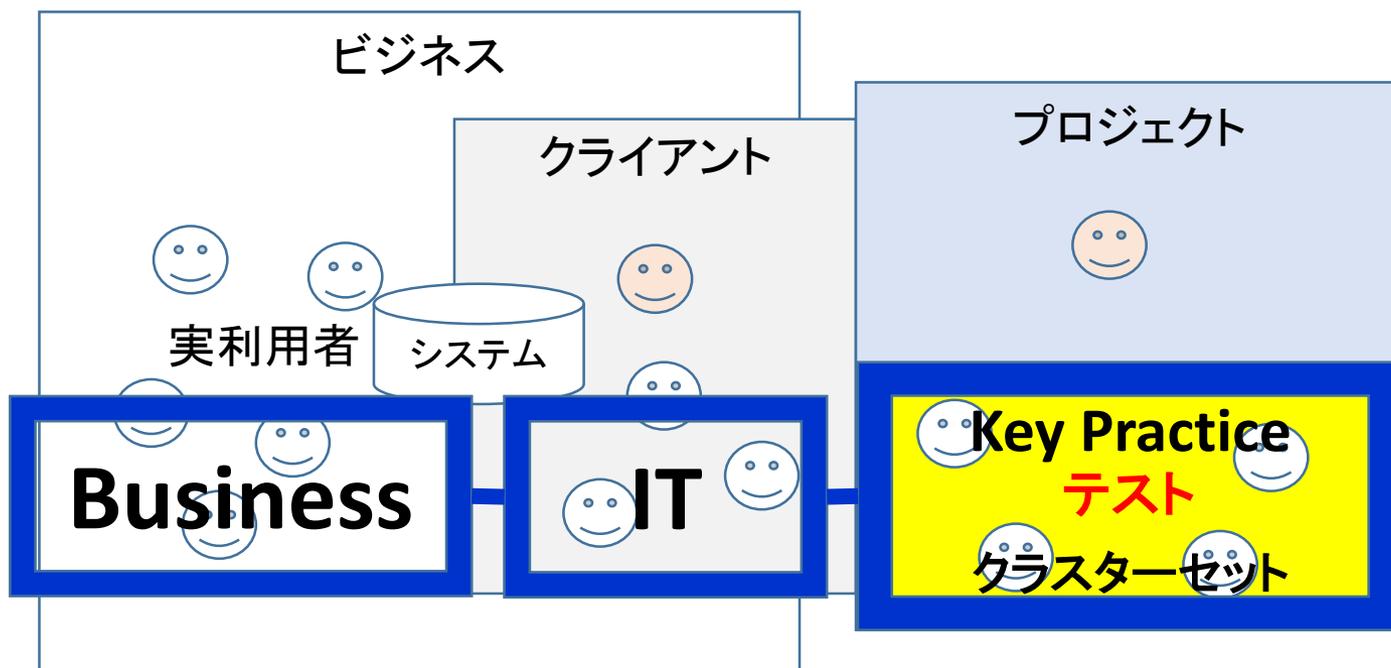
- 3グループ(SR: 利害関係者との関係、TM: テスト管理、TP: テスト業務の専門性) に16キーエリア、146プラクティスがあり、それぞれに初期・コントロール・効率化・最適化の4レベルが定義されている。
- このモデルをベースにアセスメントを実施。成熟度を判定し、改善目標やビジネスゴール達成に寄与するキーエリアを中心に改善を進めることができる。

|    |                 | 初期レベル | コントロール レベル |   |   |   | 効率化 レベル |   |   |   | 最適化 レベル |   |   |
|----|-----------------|-------|------------|---|---|---|---------|---|---|---|---------|---|---|
| SR | 1 利害関係者のコミットメント |       | 1          | 2 | 3 | 4 | 1       | 2 | 3 | 1 | 2       | 3 |   |
|    | 2 関与の度合い        |       | 1          | 2 | 3 | 4 | 1       | 2 | 3 | 1 | 2       | 3 |   |
|    | 3 テスト戦略         |       | 1          | 2 | 3 | 4 | 1       | 2 | 3 | 1 | 2       | 3 |   |
|    | 4 テスト組織         |       | 1          | 2 | 3 | 4 | 1       | 2 | 3 | 4 | 1       | 2 | 3 |
| TM | 5 コミュニケーション     |       | 1          | 2 | 3 | 4 | 1       | 2 | 3 | 1 | 2       | 3 |   |
|    | 6 報告            |       | 1          | 2 | 3 | 4 | 1       | 2 | 3 | 1 | 2       | 3 |   |
|    | 7 テストプロセス管理     |       | 1          | 2 | 3 | 4 | 1       | 2 | 3 | 1 | 2       | 3 |   |
|    | 8 見積もりと計画       |       | 1          | 2 | 3 | 4 | 1       | 2 | 3 | 4 | 1       | 2 | 3 |
| TP | 9 メトリクス         |       | 1          | 2 | 3 | 4 | 1       | 2 | 3 | 4 | 1       | 2 | 3 |
|    | 10 次層管理         |       | 1          | 2 | 3 | 4 | 1       | 2 | 3 | 4 | 1       | 2 | 3 |
|    | 11 テストウェア管理     |       | 1          | 2 | 3 | 4 | 1       | 2 | 3 | 1 | 2       | 3 |   |
| TP | 12 手法の実装        |       | 1          | 2 | 3 | 4 | 1       | 2 | 3 | 4 | 1       | 2 | 3 |
|    | 13 テスト担当者のプロ意識  |       | 1          | 2 | 3 | 4 | 1       | 2 | 3 | 4 | 1       | 2 | 3 |
|    | 14 テストケース設計     |       | 1          | 2 | 3 | 4 | 1       | 2 | 3 | 4 | 1       | 2 | 3 |
|    | 15 テストツール       |       | 1          | 2 | 3 | 4 | 1       | 2 | 3 | 4 | 1       | 2 | 3 |
|    | 16 テスト環境        |       | 1          | 2 | 3 | 4 | 1       | 2 | 3 | 4 | 1       | 2 | 3 |

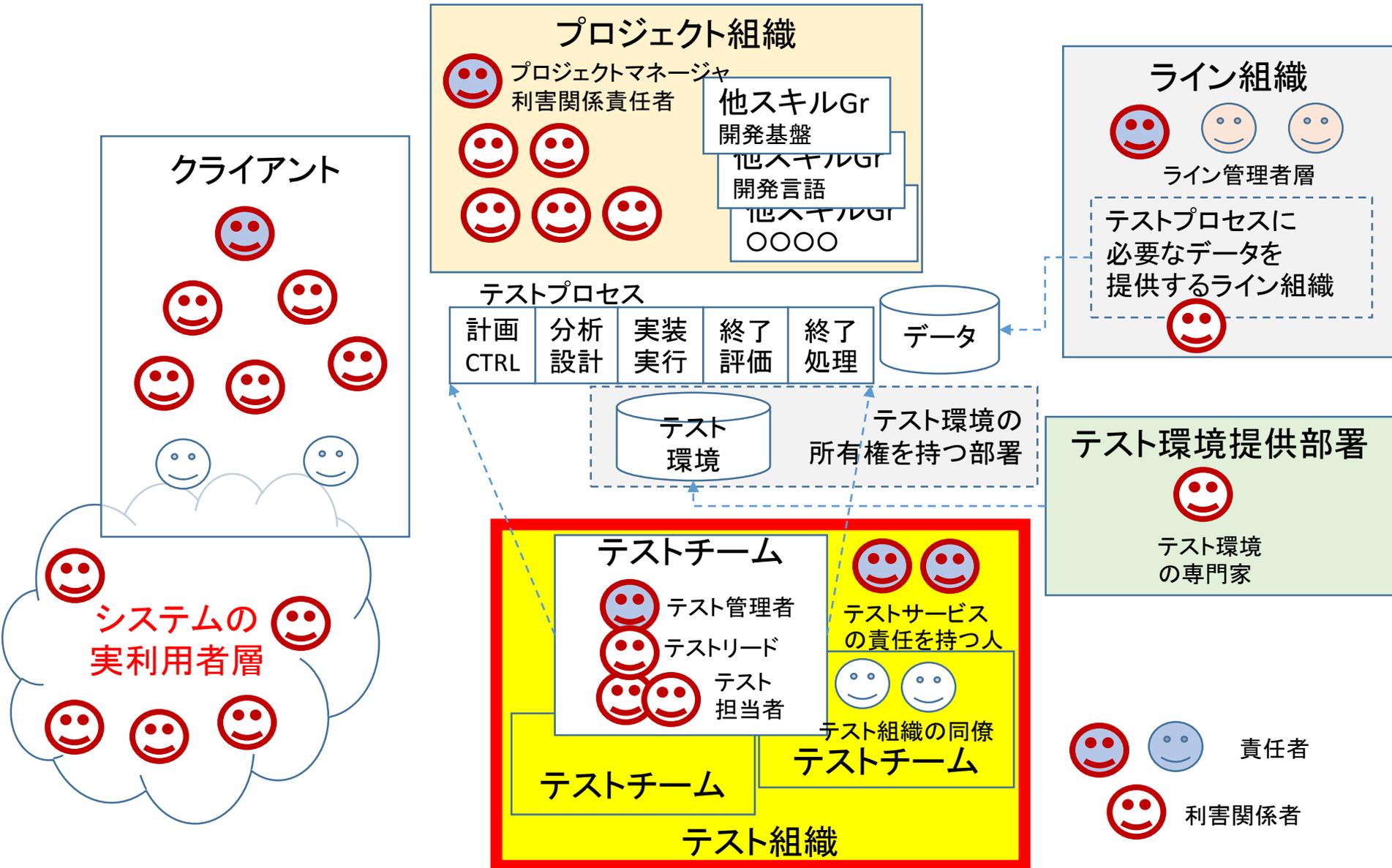
# テストプロセス改善モデル TPI NEXT

## TPI NEXTの特徴(1)

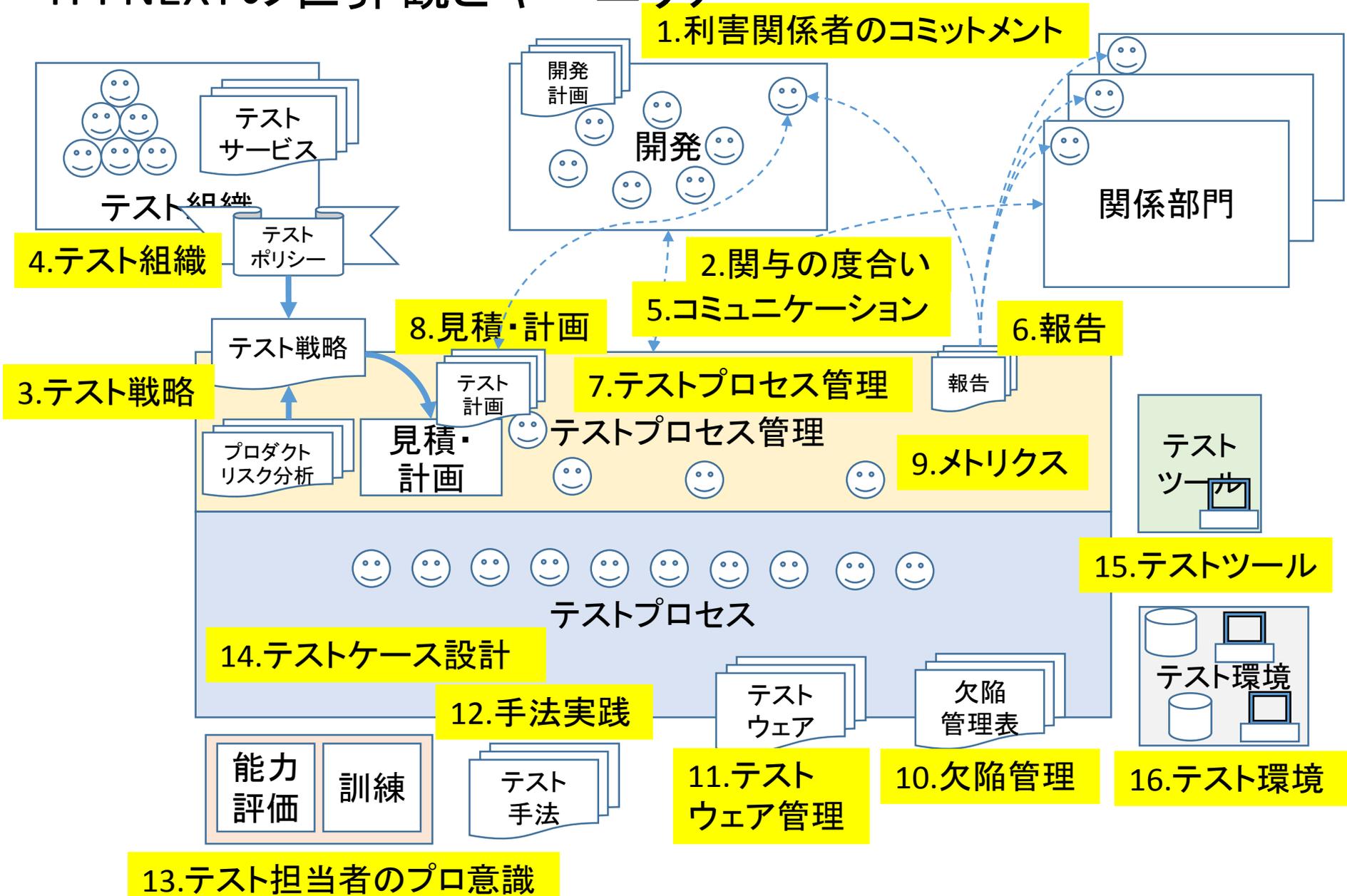
- ビジネス（経営）から見たテストのあり方、実務者から見たテストのあり方など、さまざまな立ち位置からテストをよくするための道具が用意されている。



# TPI NEXTが想定している組織モデル例



# TPI NEXTの世界観とキーエリア







# TP: テスト業務の専門性

M

1202: 実装したテスト手法を継続的に強化し、改善している  
1201: テストチームは、テスト手法について組織的にフィードバックしている

組織的ITプロセスの保守と革新

12. 手法の実践

1303: テスト担当者は、テストの仕事に対する責務と説明責任を持ち、プロセスの継続的改善に努めている

開発-テスト相互理解に基づく連携と品質管理・改善実践

13. テスト担当者のプロ意識

## 14. テストケース設計

1403: テスト設計技法を、将来さらに再利用するために評価し、調整している

本番以降も欠陥・インシデント管理  
→ 分析・評価継続

1503: テストツールの実装とテストツールポリシー実装に関してビジネスケースに記載されたゴール達成について、テストツールを定期的に評価している

1502: 専門知識、実践事例、テストツールプロダクトを収集し、今後のプロジェクトのために公開している

1501: ツールポリシーを策定し、既存のテストポリシーとの同期を保っている

15. テストツール

1603: テスト環境が対象範囲内か範囲外かに関して、サービスカタログに記載している

1602: テスト環境の利用に関して標準契約に定めている

1601: テスト環境の所有権を持つ部署が明確に定められている

16. テスト環境

L

K

1302: テストの職務が、組織の人事管理、もしくは個人のキャリア開発の一部になっている

1301: テスト担当者は、SIGやセミナーの積極的参加、論文読むなどにてスキル保持のための最新情報を常に取り入れる

1402: テストケースそれぞれの妥当性と保守性をチェックし、評価している

1401: 次のテストレベルや本番で発生した欠陥を分析、テストケースの正確性や有効性の向上につなげる

## TP: テスト業務の専門性

12E4: 必須と条件付きの要素について、実践事例がある

12E3: テスト手法の各要素について、必須/条件付き/任意のいずれかが記載されている

16E3: テスト環境の提供部署は、テスト管理者やテスト環境の専門家から正式に検収を受けたテスト環境の技術設計を納品している

16E4: テスト環境の提供部署との合意に、SLAの特性が含まれている

13E3: テストに関わる要員は、プロジェクトの他のスキルグループと良好な関係を築き、自身の仕事を楽しんでいる

13E4: テストのタスクが期待とおりに定義され、割り振られ、実行されている

14E4: テストケースが設計できないような品質特性のテスト作業には、チェックリストを用いている

14E2: テストケースによって達成できるテストベースのカバレッジレベルが明確である

14E3: テストケース設計に正式なテスト設計技法を用いている

15E4: これらテストツールの利用が、テストプロセスに組み込まれている

サービス運営  
管理

### 12. 手法の実践

12E2: 完全かつ包括的なテンプレートの一式が、テスト手法の一環として提供されている

### 13. テスト担当者のプロ意識

より適切な設計実践とその成果物

### 14. テストケース設計

### 15. テストツール

開発全般で利用されているツール

16E2: テスト環境の論理設計と機能設計をしている。その中でアプリケーション、システム、コネクシオン、スタブやドライバ(モックアップ)の利用について明記している

### 組織としての 手法定義

13E1: テスト担当者がテスト資格を持っている

13E2: テスト担当者は技法選択理由を論理的根拠で説明できる

15E2: テスト担当者が必要だと思ったときにテストツールがいつでも利用可能である

15E3: 導入された各テストツールに対してビジネスケースを作成する

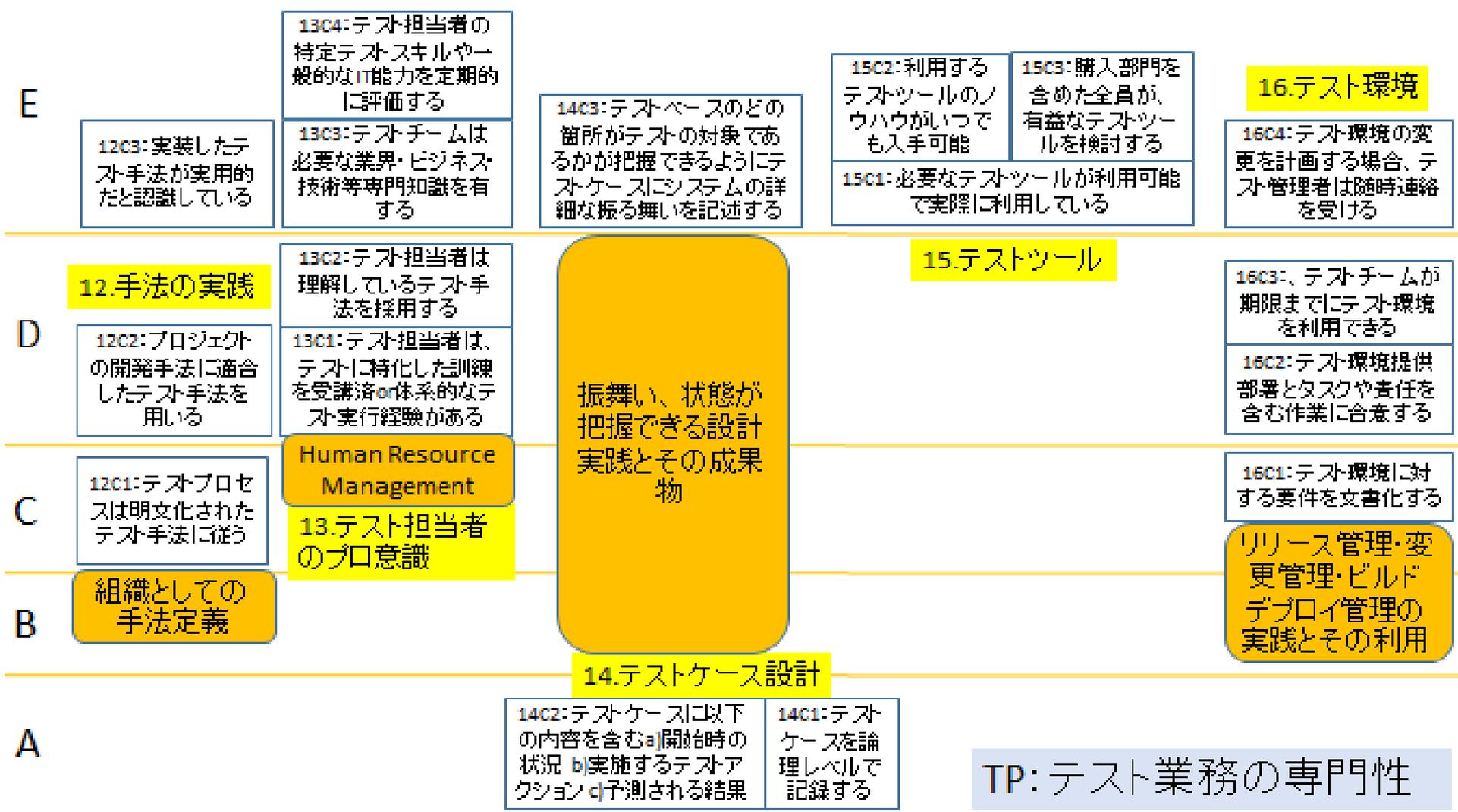
16E1: 事前作成済チェックリストにてテスト環境を検収する

12E1: テスト手法には、すべてのテスト活動に関するゴール、役割、用いるべき技法、事前条件が明文化される

14E1: テストケースはテスト組織の同僚が見ても理解でき、保守できるものになっている

15E1: 利用しているテストツールは、テスト作業より速く、低コストで、良いものにし、テストプロセスを現場でより扱いやすくするために採用されている

### 16. テスト環境



# TM: テスト管理

M

702: 過去のテストプロジェクトから学んだ教訓をテストプロセス管理の改善に利用している

## 7. テストプロセス管理

## 11. テストウェア管理

L

プロジェクト完了  
評価・報告

### 8. 見積もりと計画

|                               |   |
|-------------------------------|---|
| 802: 見積もりの技法や規律を、組織レベルで保守している | 803: 規定された見積もり技法のカギとなる数値やデータが、組織レベルで提供されている |
|-------------------------------|---|

組織全体への工数収集・分析による監視

### 9. メトリクス

### 10. 欠陥管理

|  |
|--|
| 1003: 欠陥が共通の属性のために分析され、今後の欠陥を予防する観点から提案が出されている         |
| 1002: 欠陥管理の責任が、テストプロセスに必要なデータを提供するライン組織あるいはプロジェクト組織にある |

形式張らない分析と解決策

|  |
|--|
| 1103: プロジェクト終了時に保守に引き渡すテストウェアが、保守されないテストウェアと容易に分離できる                   |
| 1102: 再利用に備えたテストウェア保持に関するガイドラインが入手可能な状態にあり、テストウェアの再利用を測定している           |
| 1101: テストプロジェクト終了後にどのテストウェアを保持するかをテストプロジェクト開始時に合意し、テストプロジェクト実施中に見直している |

協調的なプロダクト維持

K

701: テストプロセス管理は、(テスト組織により内部的に、そして利害関係者により定期的に) 評価されている

801: テスト計画には、テストウェアの今後の再利用に向けた保守活動が含まれている

901: メトリクス方法がニーズにどのように貢献するかを監視している

902: 情報ニーズの変化に対応して、メトリクスを新しくしたり最適化している

1001: ライン組織やプロジェクト管理から、欠陥管理の一連のガイドラインが提供されており、各テストプロジェクトで利用されている

7E3:テストロードは、人員の(再)配置を任されている

8.見積もりと計画

11E3:テストウェア管理は、論理的な補完構造と、役割および権限の構造によって支えられている

8E3:見積もりや計画作業を支援するために、メトリクスを用いている

8E4:テスト計画には、テストベースのテスト容易性レビューとテストプロジェクトの評価を含んでいる

9E4:分析したメトリクスから導き出された結論について利害関係者と話し合い、結論に基づいて対処している

10E4:多くの情報を記録し傾向を特定している。記録情報:欠陥、サブシステム、優先度、プログラムとそのバージョン、テストベースとバージョン、根本原因、全ステータス移行と問題解決者など

11E1:テストベースとテスト対象およびすべてのテストウェアを、名前とバージョンで参照できる

11E2:テストケースと要件のトレーサビリティが確保されている

見積・計画技法定義と数値収集・活用

構成管理・要件設計  
トレーサビリティ

11.テストウェア管理

7E2:テスト計画の調整を文書に残している

8E2:正式な技法を用いて、テストフェーズやテスト活動の見積もりや計画作業を行っている

9E2:メトリクスの収集作業が、テストプロセスの進捗や品質を良くすることと矛盾していない

9E3:テストプロセスでは、テストプロセスの効率性を測定するためのメトリクスを定義して活用している

10E3:欠陥管理を広範囲なレポートに役立てている。言い換えると、いろいろな方法でレポートを選択したりソートしたりできる

7E1:(予想される)テスト計画からの逸脱について、利害関係者や他の利害関係者と協議している

8E1:できる限り正確であるよう、最低でも2つ以上の見積もり技法を用いている

9E1:必須データの収集、分析、価値の評価に必要な工数を、そこから得られる利点に対して測定している

10.欠陥管理

7.テストプロセス管理

9.メトリクス

10E1:欠陥の管理ツールが、欠陥の状態遷移に関する承認フローを遵守させる効力を持っている

10E2:欠陥のログを残したり欠陥の状況を追跡するすべての人々は、同じ欠陥管理ツールを利用しているか、あるいは複数の欠陥管理ツールをシームレスに接続させて利用している

TM:テスト管理

# TM: テスト管理

E

9.メトリクス

10.欠陥管理

11C4: テストウェア、テストベース、テスト対象の扱い方が明確に規定され、テストチームに伝えられている

D

8.見積もりと計画

組織が必要なメトリクス情報提供

9C3: メトリクスの入力データは、正確かつ確認可能である

10C4: 欠陥のアクセスメントと解決への関係者は欠陥管理ツールにアクセスできる

11C3: テストチームはテストウェア管理下のすべてのアイテムにアクセスできる

C

7.テストプロセス管理

8C3: 各テストフェーズやテスト活動間の依存関係を、テスト計画に記載している (フェーズと活動にある程度の重複を許容する)

8C4: テスト見積りやテスト計画の作業を利害関係者と協議している

9C1: テストプロセスでは、テストプロジェクト見積りや管理のためにメトリクスを定義し、利用している

9C2: メトリクスに必要なインプットは統一した形式で記録し、定義されたメトリクスは体系的に保管されている

PBS製品分割構成ベースの対象特定

見積・計画手法検討・実施・検証

プロジェクト管理

B

7C3: 各テスト活動を監視しており必要に応じて計画を調整している

7C4: テスト計画は関連する利害関係者の合意を得ている

8C1: テスト作業の見積りには、比率のような単純な技法を用いている

8C2: テスト活動ごとに活動期間、必要なリソース、活動期間に作る成果物の指標がある。特定すべき活動は、テストケースの定義、実行、テスト計画、管理である

10C3: 欠陥を処理するために、責任者を決めている

11C1: テストベース、テスト対象、すべてのテストウェアを名前とバージョンで特定する

11C2: 各テストケースを、テストベース文書に明白な方法で関連付けている

プロセス・プロダクト見識提供

A

7C1: テストプロジェクトの開始時にテスト計画を作成し、少なくとも、テストの任務、テストスコープ、テスト計画、および役割と責任について記載している

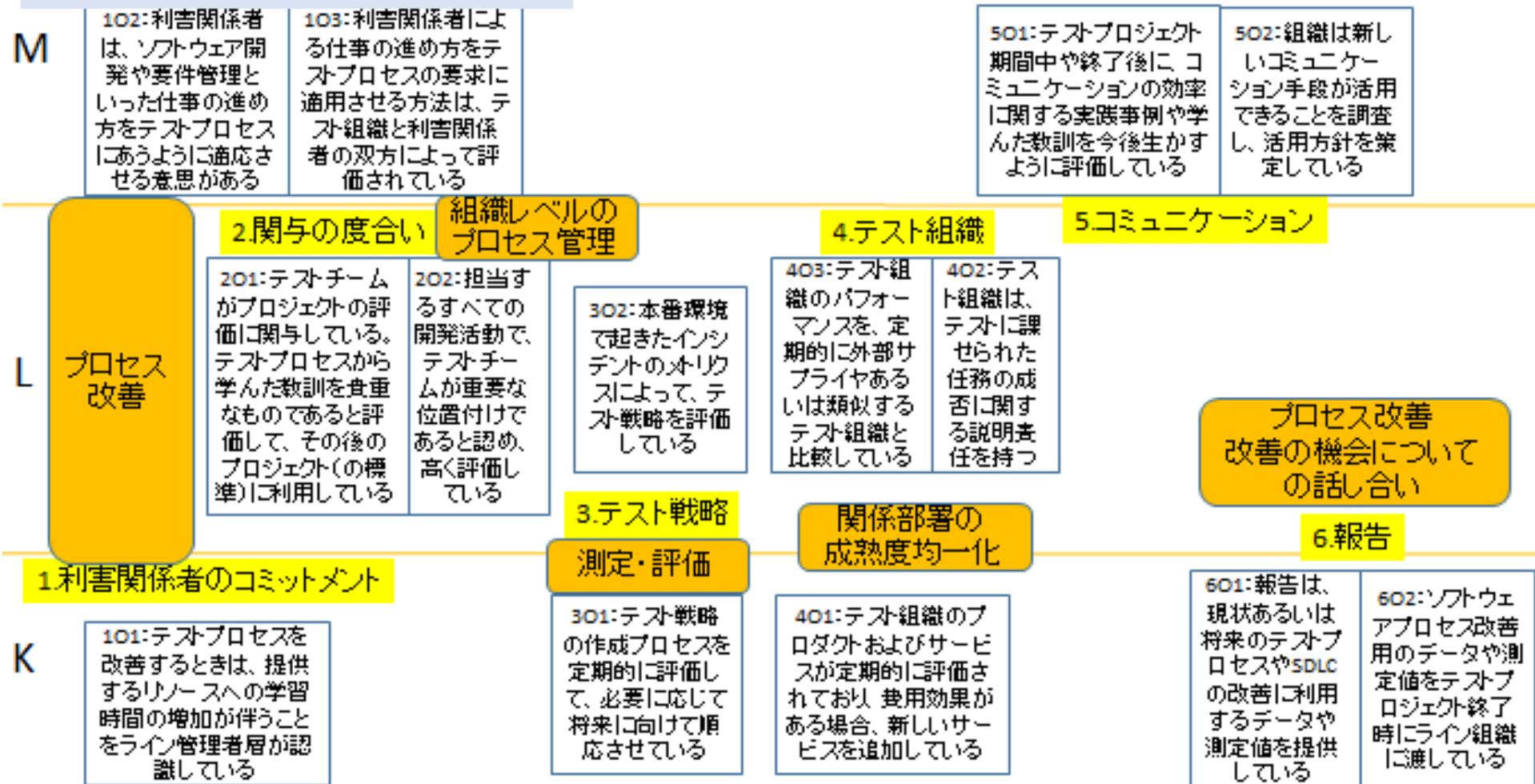
7C2: テスト計画を利害関係者と合意している

10C1: 欠陥ライフサイクルを(再テストを含めて)定義して、適用している

10C2: 各欠陥には、固有のID、関連するテストケースID(あてはまる場合)、欠陥報告者、日付、重要度カテゴリ、説明(欠陥再現手順、期待する結果と実際の結果)、欠陥ステータスを記録する

11.テストウェア管理

# SR: 利害関係者との関係



2E3:テスト担当者は、テスト対象が記述されたテストベースのテスト容易性をレビューし、テストベースの最適化に積極的に関与している

4E3:テスト組織をどこどのように位置づけるかについて、熟慮された選択がなされている

4E4:テストポリシーに則っている

5E3:テストチームは、利害関係者と適切にコミュニケーションできるような様々なコミュニケーション手段を用意する

6E3:報告書には、テストのゴールやプロダクトリスクに関して、テストの傾向、テストプロジェクトからの推奨事項が含まれる

## 2.関与の度合い

# SR:利害関係者との関係

## 欠陥リスク管理

4E2:テスト組織は、合意したテストリソースおよびサービスをプロジェクトに提供している

4E1:テストサービスを提供する異なる人員や部署が、テスト作業について意識合わせを行っている

## 4.テスト組織

1E3:利害関係者は、テストプロセスへの影響側面(テスト対象作業へのリリース順やプロジェクトスコープを含む)について積極的に行動を起こしている

1E2:利害関係者は、積極的にテストプロセスやテスト対象の品質に関する情報を入力している

2E1:テスト担当者は、変更要求やテストベースの変更による影響分析およびリスク分析に積極的に取り組んでいる

2E2:テスト担当者は、欠陥の影響分析に積極的に取り組んでいる

3E3:テスト戦略には適切なテスト設計技法を含めている

# 利害関係者管理

## 6.報告

6E2:報告書には、テストプロセスの進捗やプロジェクトリスクを考慮して、テストの傾向とテストプロジェクトからの推奨事項が含まれている

## 変更管理

## 統合プロジェクト管理

## 1.利害関係者のコミットメント

## 3.テスト戦略

## 5.コミュニケーション

1E1:関連するすべての利害関係者を定義して(必ずしも文書化の必要はない)、テスト担当者に周知している

3E1:関係するすべての利害関係者が、定義された(かつ文書化された)テスト戦略に合意している

3E2:テストカバレッジにおけるテストレベルやテストタイプの重複やギャップについて、十分に検討されている

5E1:テストチームは、どの情報をどの利害関係者と共有すべきかを決定できている

5E2:テストチームは他の利害関係者と共に該当する会議に参加している

6E1:効率的な意思決定に必要な利害関係者の報告要求を満たすことと、提供するための作業量のバランスが取れている

# SR: 利害関係者との関係

E

2C4: テスト担当者は、プロジェクト全体のプロジェクトリスクの分析と軽減策の立案に関与している

3C4: 再テストや回帰テストについて、シンプルな戦略を決定している

4C4: クライアントから見て、テスト組織におけるテスト関連のプロダクトやサービスが明確である

D

## 2 関与の度合い

## テストナレッジ管理システム

テストプロセスの目的・ゴール等明確化

テスト早期関与の重要性強調

4C2: テスト組織において、コントロールおよび説明責任の体制がある

4C3: テストのタスクや責任が定義(および文書化)され、それらは人や組織内の各部署に割り振られている

5C4: テストチームは利害関係者と共に、進捗、プロダクト品質、リスクについて慎重に検討し、潜在的な遅延があれば、先を見越して警告を出す

## 1 利害関係者のコミットメント

C

1C4: 利害関係責任者は、文書化されたプロダクトリスク分析(テスト戦略のインプット)に対する責任を持つ

2C3: プロジェクト計画において、テストプロセスとその他のプロセスの依存関係を考慮できるように、テスト担当者が計画に関与している

## 4. テスト組織

5C2: テストチームは、関連情報を利害関係者から積極的に収集している

5C3: テストチームがコミュニケーションで合意して、決定した時点へさかのぼることができる

6C2: 報告の頻度や内容が、意思決定プロセスの利害関係者の基本要件に合致している

6C3: 報告は書面によって行われている

## 関係者のコミット

## プロダクトリスク管理

## プロジェクト管理 議事録共有

B

1C2: テストリソースに対する予算は、利害関係責任者が認め、交渉可能である

1C3: 利害関係者はコミットしたリソースを実際に手配している

2C2: テスト活動をプロジェクトのクリティカルパスにしないよう、テスト実行より地前の早い時期に開始している

3C3: テストレベル、タイプ、カバレッジ、深さは、個別リスク分析結果により異なる

## 5 コミュニケーション

5C1: 各チームメンバーは、下された決定と内部進捗を把握している

## 6 報告

プロジェクト管理 報告共有

## プロジェクト管理

## 3. テスト戦略

A

1C1: 利害関係責任者を決定し(文書化必要ない)、テスト担当者に周知している

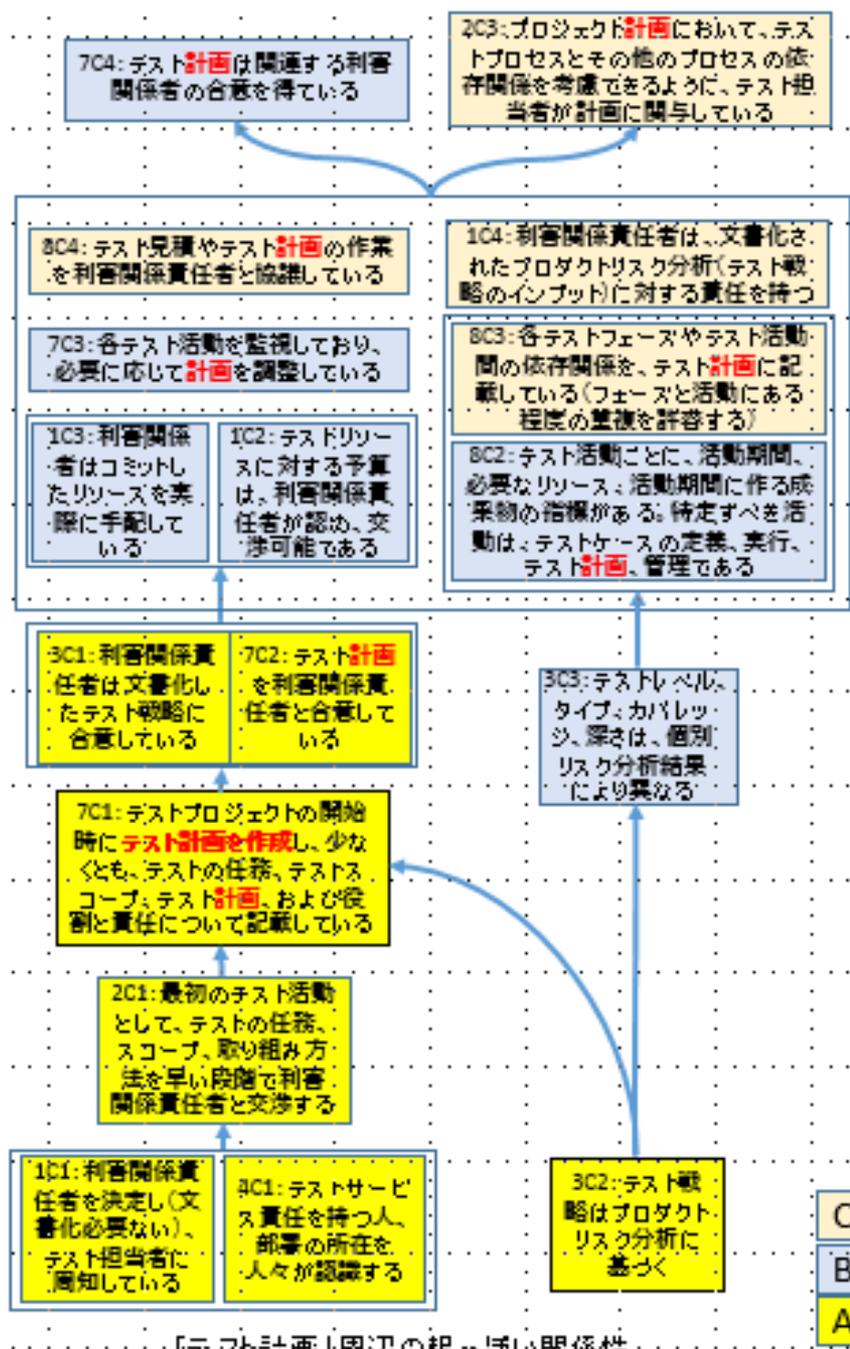
2C1: 最初のテスト活動として、テストの任務、スコープ、取り組み方法を早い段階で利害関係責任者と交渉する

3C1: 利害関係責任者は文書化したテスト戦略に合意している

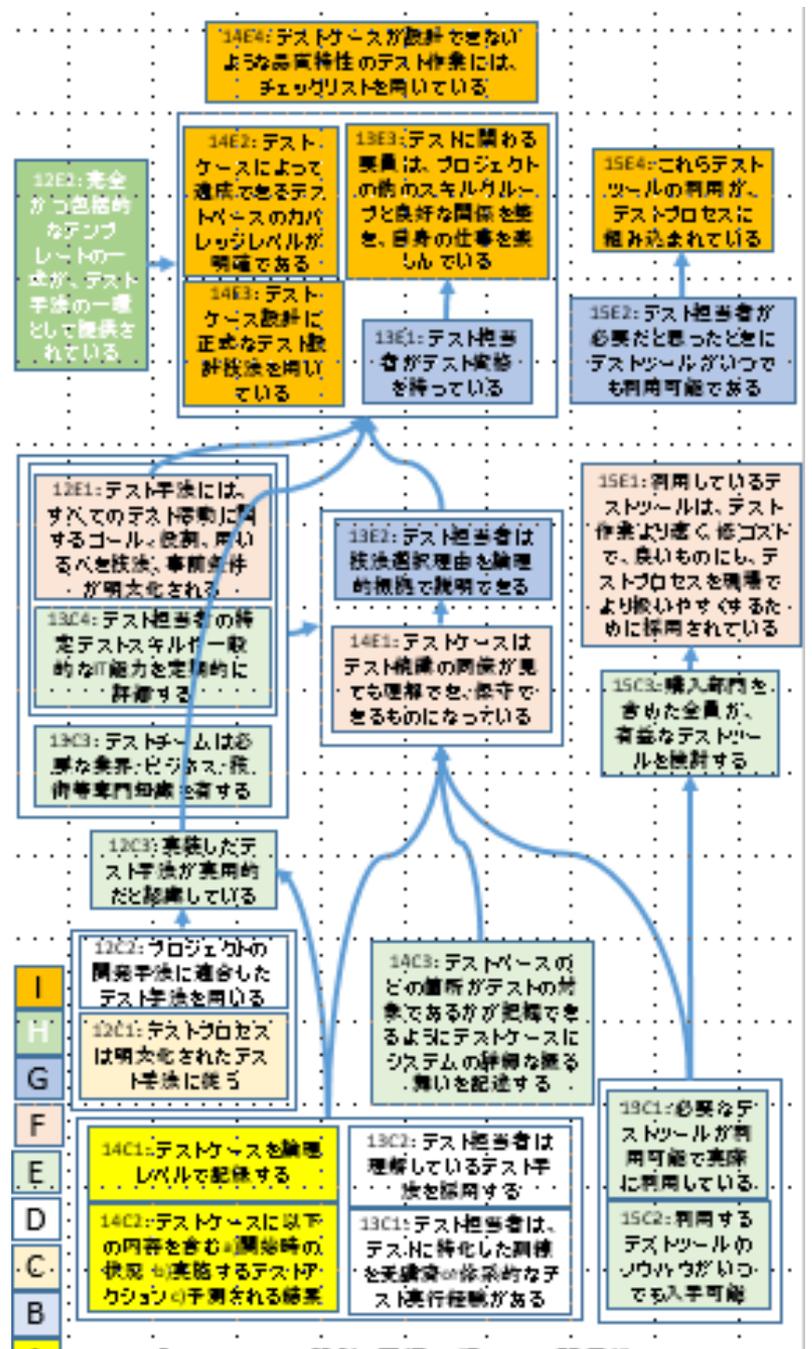
3C2: テスト戦略はプロダクトリスク分析に基づく

4C1: テストサービス責任を持つ人、部署の所在を人々が認識する

6C1: 報告に時間、コスト、結果、リスクなどが含まれる



「テスト計画」周辺の組っばい関係性



「テストケース設計」周辺の組っばい関係性

# テストプロセス改善モデル TPI NEXT

## TPI NEXTの特徴(2)

- テスト技術や管理技術だけではなく、“**人**”に着目した要素にも重きを置いている。(バランスが良い)

### キーエリアの分類

**SR: 利害関係者との調整**

TM: テスト管理

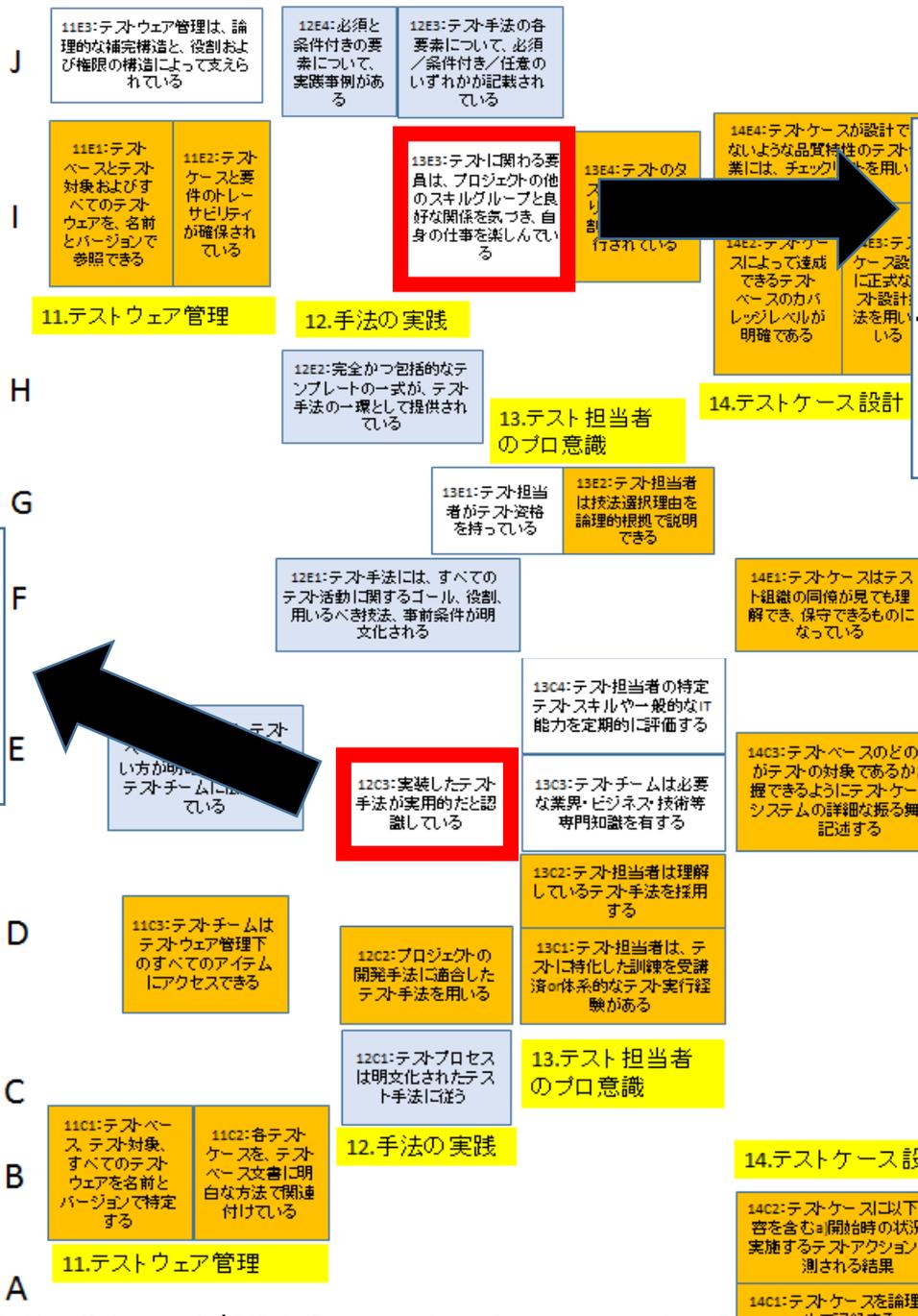
TP: テスト業務の専門性

### 12. 手法の実践

12C3: 実装したテスト手法が  
**実用的だと認識している**

### 13. テスト担当者の**プロ意識**

13E3: テストに関わる要員は、プロジェクトの他のスキルグループと  
**良好な関係を築き、自身の仕事を楽しんでいる**



13E3: テストに関わる要員は、プロジェクトの他のスキルグループと良好な関係を築き、自身の仕事を楽しくしている

12C3: 実装したテスト手法が実用的だと認識している

摘要  
 基礎的な実践事項  
 標準類構築・整備

# テストプロセス改善モデル TPI NEXT

## TPI NEXTの特徴(3)

C213C17C45

- キーエリア名と内容が必ずしも一致していない場合があるので注意。

→個別プラクティス間の依存関係をしっかり理解して活用すること

問:これらのプラクティスはどのキーエリアに存在するものか?

テスト担当者は理解しているテスト手法を採用する

テストプロジェクトの開始時にテスト計画を作成し、少なくとも、テストの任務、テストスコープ、テスト計画、および役割と責任について記載している

テストチームは利害関係者と共に、進捗、プロダクト品質、リスクについて慎重に検討し、潜在的な遅延があれば、先を見越して警告を出す

- 7.テストプロセス管理
- 12.手法の実践
- 13.テスト担当者のプロ意識
- 14.テストケース設計

- 3.テスト戦略
- 4.テスト組織
- 7.テストプロセス管理
- 8.見積と計画

- 1.利害関係者のコミットメント
- 5.コミュニケーション
- 6.報告
- 7.テストプロセス管理

# JaSST'18東京発表事例（拡張版）

理解促進のために、一部分に詳細説明を補記（スライド左上に●）してあります

# 今回の事例

- JaSST'17北海道で発表～テストチームのプロセス改善をTPI Nextを使ってやってみよう！と思ったリーダ中山さんの事例の詳細。
- TPI Next日本語書籍を一読した中山さん。セルフアセスメントをやってみた。

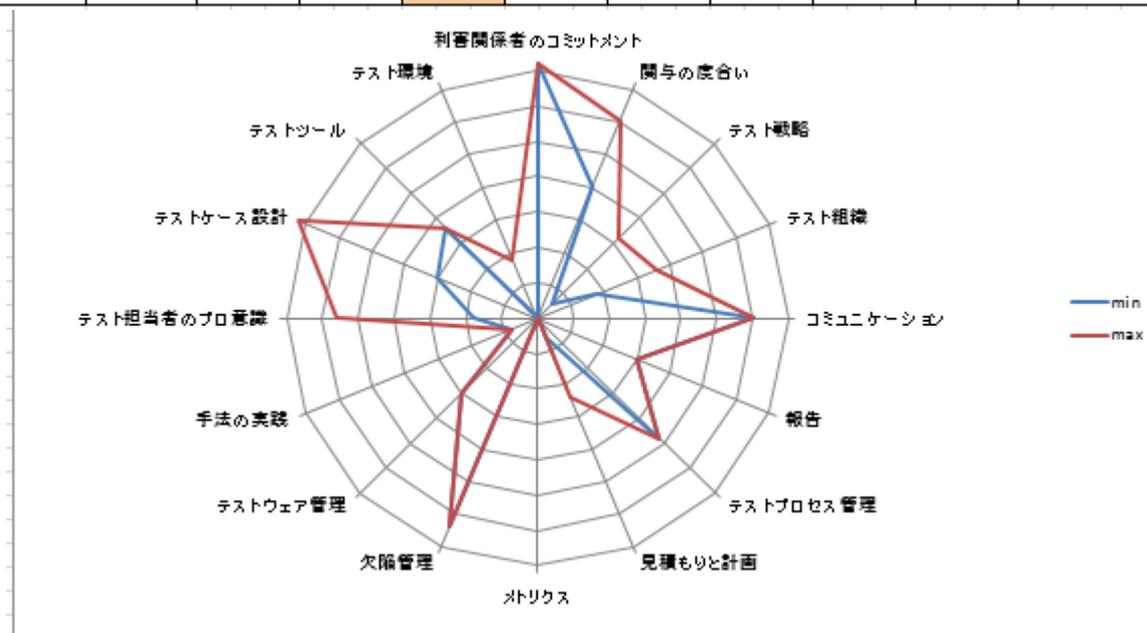
プロセスモデルに精通する有識者は多くはない  
組織やチームの改善活動も積極的ではない  
そのような背景でも「何とかしたい！」と思っているリーダやエンジニアが多いのではないか？

# TPI Nextチェック項目の一部

|   |  |   |    |   |  |
|---|--|---|----|---|--|
| SR  | 1.利害関係者のコミットメント  |   | TM | 11.テストウェア管理   |  |
|   | 初期レベル  |   |    | 初期レベル   |  |
|   | コントロールレベル  |   |    | コントロールレベル   |  |
|   | 1 利害関係責任者を決定し（必ずしも文書化の必要はない）   |   |    | 1 テストベースとテスト対象およびすべてのテストウェアを、名前とバージョンで特定している。   |  |
|   | 2 テストリソースに対する予算は、利害関係責任者が認める   |   |    | 2 各テストケースを、テストベース文書に明白な方法で関連付けている。  |  |
|   | 3 利害関係者はコミットしたリソースを実際に手配している。  |   |    | 3 テストチームは、テストウェアの管理下のすべてのアイテムにアクセスできる。  |  |
|   | 4 利害関係責任者は、文書化されたプロダクトリスク分析  |   |    | 4 テストウェア、テストベース、テスト対象の扱い方が明確に規定され、テストチームに伝えられている。   |  |
|   | 効率化レベル   |   |    | 効率化レベル  |  |
|   | 1 関連するすべての利害関係者を定義して（必ずしも文書化する）。   |   |    | 1 テストベースとテスト対象およびすべてのテストウェアを、名前とバージョンで参照できる。  |  |
|   | 2 利害関係者は、積極的にテストプロセスやテスト対象の品   |   |    | 2 テストケースと要件のトレーサビリティが確保されている。   |  |
| 3 利害関係者は、テストプロセスに影響を与える側面について対象のテスト作業へのリリース順序やプロジェクトスコープが |  | 3 テストウェア管理は、論理的な補充構造と、役割および権限の構造によって支えられている。                        |    |   |  |
| 最適化レベル  |  | 最適化レベル  |    |   |  |
| 1 テストプロセスを改善するときは、提供するリソースへの学習している。                       |  | 1 テストプロジェクト終了後にどのテストウェアを保持するかをテストプロジェクト開始時に合意し、テストプロジェクト実施中に見直している。 |    |   |  |
| 2 利害関係者は、ソフトウェア開発や要件管理といった仕事意思がある。                        |  | 2 再利用に備えたテストウェア保持に関するガイドラインが入手可能な状態にあり、テストウェアの再利用を測定している。           |    |   |  |
| 3 利害関係者による仕事の進め方をテストプロセスの要求に双方によって評価されている。                |  | 3 プロジェクト終了時に保守に引き渡すテストウェアが、保守されないテストウェアと容易に分離できる。                   |    |   |  |
| SR  | 2.関与の度合い   |   | TP | 12.手法の実践  |  |
|   | 初期レベル  |   |    | 初期レベル   |  |
|   | コントロールレベル  |   |    | コントロールレベル   |  |
|   | 1 最初のテスト活動として、テストの任務、スコープ、取り組み交渉している。                                    |   |    | 1 テストプロセスは、明文化されたテスト手法に従っている。テスト手法には、一連のテスト活動、テストプロジェクトの成果となるテストプロダクト、作業の途中で発生する追加要求について記述している。 |  |
|   | 2 テスト活動をプロジェクトのクリティカルパスにしないよう、テストプロジェクト計画において、テストプロセスとその他のプロセス計画に関与している。 |   |    | 2 テスト手法は、プロジェクトが用いている開発手法に適合している。   |  |
|   | 3 テストプロセスは、プロジェクト計画において、テストプロセスとその他のプロセス計画に関与している。                       |   |    | 3 テストプロジェクトにとって、実装したテスト手法が実用的なものであると認識されている。  |  |
|   | 4 テスト担当者、プロジェクト全体のテスト担当者、プロジェクト全体のテスト担当者、プロジェクト全体のテスト担当者                 |   |    | 効率化レベル  |  |
|   | 1 テストプロセスは、すべてのテスト活動に関するゴールと役割、および用いるべき技法と事前条件が明文化されている。                 |   |    | 1 テスト手法には、すべてのテスト活動に関するゴールと役割、および用いるべき技法と事前条件が明文化されている。   |  |
|   | 2 利害関係者は、ソフトウェア開発や要件管理といった仕事意思がある。                                       |   |    | 2 利害関係者は、ソフトウェア開発や要件管理といった仕事意思がある。  |  |
|   | 3 利害関係者による仕事の進め方をテストプロセスの要求に双方によって評価されている。                               |   |    | 3 利害関係者による仕事の進め方をテストプロセスの要求に双方によって評価されている。  |  |

# 当初のセルフアセスメント結果

|                 | 初期レベル | コントロール レベル |   |   |   | 効率化 レベル |   |   |   | 最適化 レベル |   |   |
|-----------------|-------|------------|---|---|---|---------|---|---|---|---------|---|---|
| 1 利害関係者のコミットメント |       | 1          | 2 | 3 | 4 | 1       | 2 | 3 |   | 1       | 2 | 3 |
| 2 関与の度合い        |       | 1          | 2 | 3 | 4 | 1       | 2 | 3 |   | 1       | 2 | 3 |
| 3 テスト戦略         |       | 1          | 2 | 3 | 4 | 1       | 2 | 3 |   | 1       | 2 | 3 |
| 4 テスト組織         |       | 1          | 2 | 3 | 4 | 1       | 2 | 3 | 4 | 1       | 2 | 3 |
| 5 コミュニケーション     |       | 1          | 2 | 3 | 4 | 1       | 2 | 3 |   | 1       | 2 | 3 |
| 6 報告            |       | 1          | 2 | 3 | 4 | 1       | 2 | 3 |   | 1       | 2 | 3 |
| 7 テストプロセス管理     |       | 1          | 2 | 3 | 4 | 1       | 2 | 3 |   | 1       | 2 | 3 |
| 8 見積もりと計画       |       | 1          | 2 | 3 | 4 | 1       | 2 | 3 | 4 | 1       | 2 | 3 |
| 9 メトリクス         |       | 1          | 2 | 3 | 4 | 1       | 2 | 3 | 4 | 1       | 2 | 3 |
| 10 欠陥管理         |       | 1          | 2 | 3 | 4 | 1       | 2 | 3 | 4 | 1       | 2 | 3 |
| 11 テストウェア管理     |       | 1          | 2 | 3 | 4 | 1       | 2 | 3 |   | 1       | 2 | 3 |
| 12 手法の実践        |       | 1          | 2 | 3 | 4 | 1       | 2 | 3 | 4 | 1       | 2 | 3 |
| 13 テスト担当者のプロ意識  |       | 1          | 2 | 3 | 4 | 1       | 2 | 3 | 4 | 1       | 2 | 3 |
| 14 テストケース設計     |       | 1          | 2 | 3 | 4 | 1       | 2 | 3 | 4 | 1       | 2 | 3 |
| 15 テストツール       |       | 1          | 2 | 3 | 4 | 1       | 2 | 3 | 4 | 1       | 2 | 3 |
| 16 テスト環境        |       | 1          | 2 | 3 | 4 | 1       | 2 | 3 | 4 | 1       | 2 | 3 |



# アセスメント結果は出たものの・・・

- 評価結果は正しいのか？実態に合っているのか？不安が・・・。
- 多くのNGが・・・どこから手をつけるべきか??
  - モデルが推奨するクラスタAから順に・・・
    - ✓それだと改善が実感できず、長引いて続かなくなるかもよ
  - 中山さんが普段から問題ではないかと気になっていた「テスト環境」に着目して・・・
    - ✓それがメンバーの人たちが解消したい1番の困り事なのかな？
- **じゃあどうすればいいの???**

# プロセスモデルベース改善の問題点

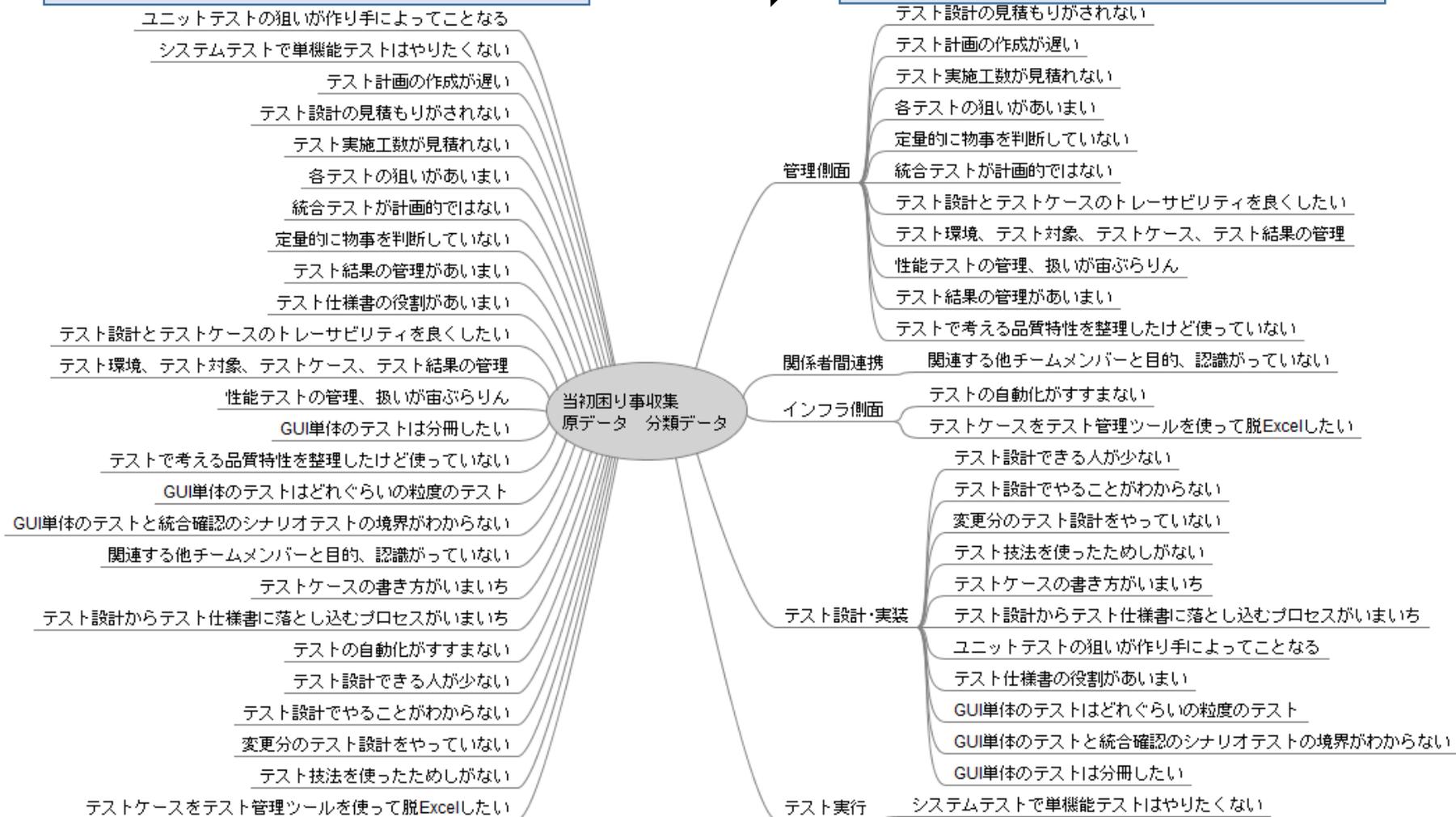
- 重たい
  - 例えばTPI Next: 16プロセスエリア143プラクティス
  - みんなで評価するのは理想だが現実的に無理な場合が多い
  - 第三者や代表者が評価する……他人事化しやすい
- モデルが難解
  - 個別のプラクティスがなかなか読み解けない
  - 対象は何？どこまでやれるとOK？わかるようでわかりにくい
- 評価結果の意味がわからない
  - 大抵は少しのOKとたくさんのNG……モチベーションダウン
  - その意味は？？何から手をつけるべき？判断の拠り所は？
- 取り組んだ結果の効果が実感しにくい
  - 何を指して取り組めばいいのか？
  - 考えるのが面倒になるとモデル適合を目標にしがち……それは誰が、何がうれしいの？

**プロセスモデルをしっかりと理解してから活用しないと  
思わぬ怪我(迷走・頓挫・自然消滅等)をしてしまう**

# メンバーからふりかえりコメントを収集・分類

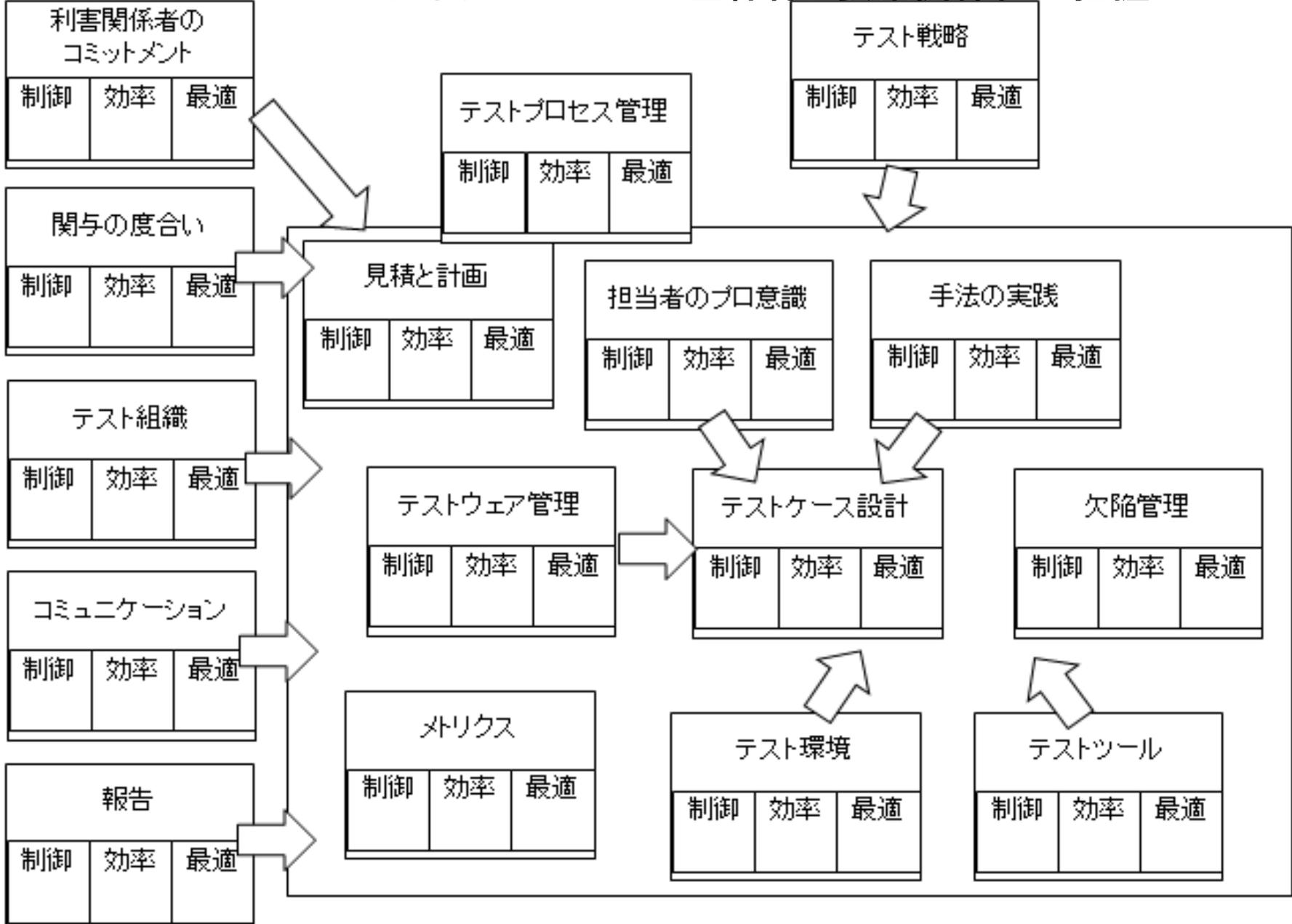
## 実際のふりかえり結果

## 整理・分類結果

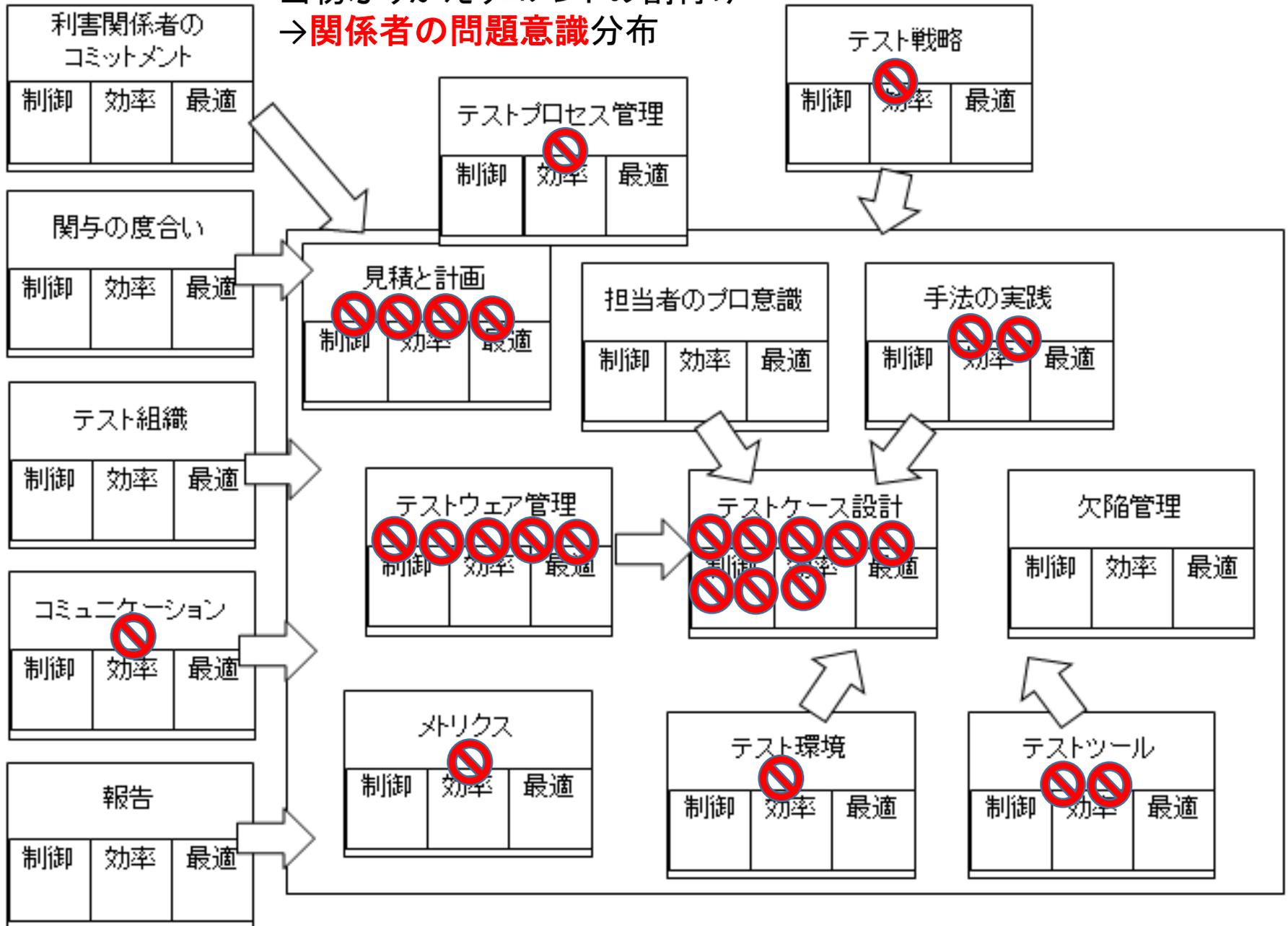


地図に疑問を持ったなら現地を直接調べよう

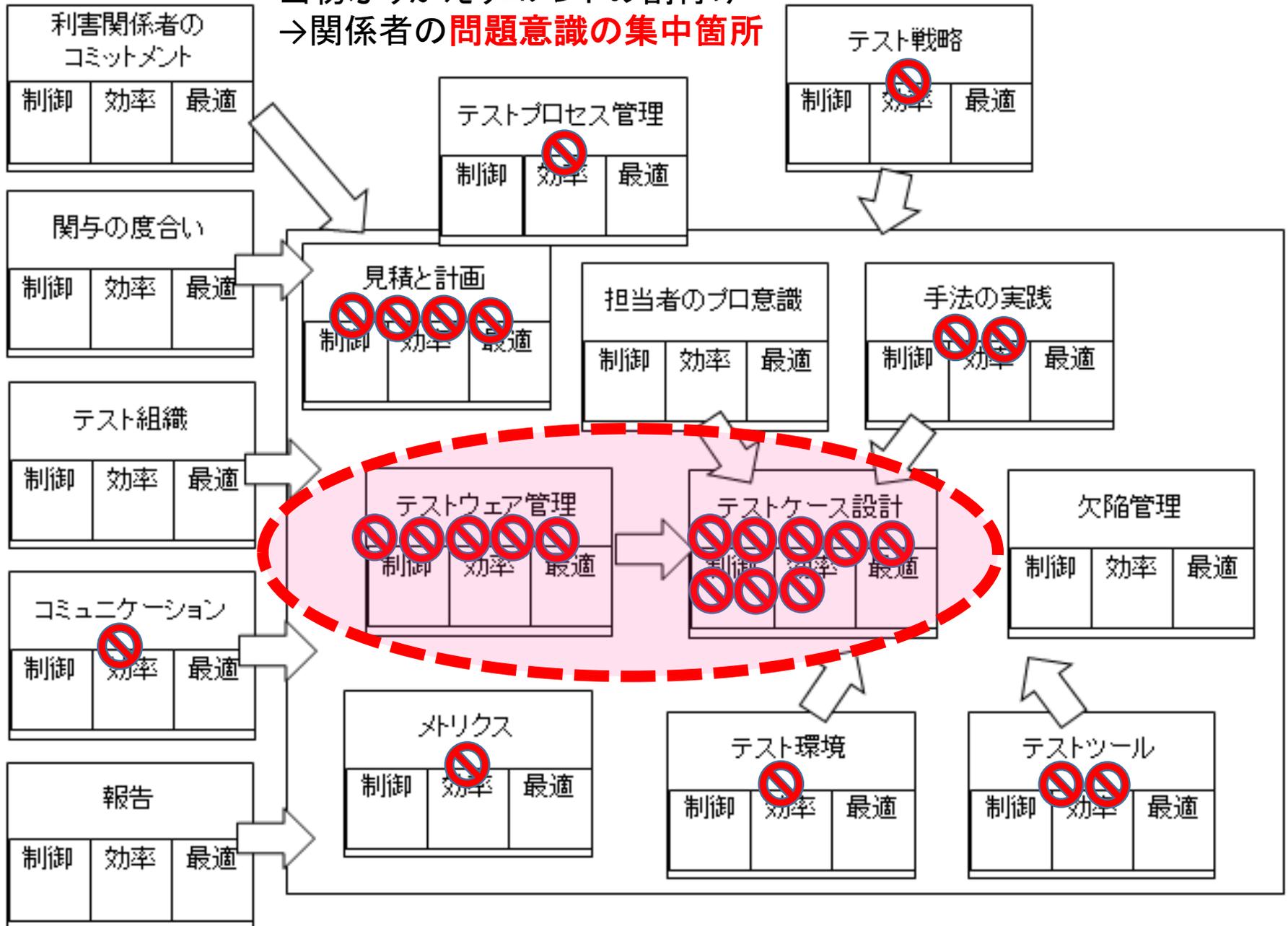
# プロセスモデルTPI Nextによるテストプロセス全体像と要素関係性の把握



当初ふりかえりコメントの割付け  
 → **関係者の問題意識**分布



当初ふりかえりコメントの割付け  
 →関係者の**問題意識の集中箇所**



# 実際に存在する品質・コスト・期間(QCD)問題関連事象・困り事の把握

管理側面

- テスト設計の見積もりがされない  
→見積りされないことでどのような困り事が発生している？
- テスト計画の作成が遅い  
→計画立案スピードのこと？計画が出来上がるタイミングのこと？
- テスト実施工数が見積れない  
→その結果どのような困り事が発生している？
- 定量的に物事を判断していない  
→その結果どのような困り事が発生している？
- 統合テストが計画的ではない  
→具体的にどのような状態？  
状況に適切なテストの組み立てになっていない（戦略的ではない）  
→その結果どのような困り事が発生している？
- 性能テストの管理、扱いが宙ぶらりん  
→管理とは具体的に何？宙ぶらりんとは具体的にどのようなこと？  
→その結果どのような困り事が発生している？
- テスト結果の管理があいまい  
→管理とは具体的に何？あいまいとは具体的に何？  
→その結果どのような困り事が発生している？
- 各テストの狙いがあいまい  
→狙いが曖昧とは具体的にどのような状態？  
システムテストで何を確認したいのか、GUIテストで何を確認したい？  
→その結果どのような困り事が発生している？
- テストの境界に対して人によって判断が異なるため、重複テストやテスト漏れだけは防ぐようにしている（手間がかかっている）
- テスト設計とテストケースのトレーサビリティを良くしたい  
→このトレーサビリティがどのようによくない？  
”テスト仕様書の役割があいまい”に関連する  
元から存在する機能に対するテストケースの粒度がバラバラ  
スパゲッティテスト仕様状態  
→その結果どのような困り事が発生している？
- テスト環境、テスト対象、テストケース、テスト結果の管理  
→具体的にどのような状態？  
”テスト仕様書の役割があいまい”に関連する  
過去のテスト結果が残っているがテスト環境情報が残っていないもの  
定期的に回帰テスト実施前にNG項目を把握して作戦を立てる際に役立つ
- 結果的にすべてをテスト実施することになる  
テストで考える品質特性を整理したけど使っていない  
→使わないことでどのような困り事が発生している？  
狙いがあいまいの背景のために作成したが終了

関係者間連携

- 制御メンバーと目的、認識がっていない  
→その結果どのような困り事が発生している？

インフラ側面

- テストの  
→すす  
テストケ  
→Exce

- 少ないことでどのような困り事が発生している？  
できる人に作業が集中する  
その人しかできない作業（例：レビュー）が止まる
- テスト設計でやることがわからない  
→その結果どのような困り事が発生している？  
できない局面では誰かに聞きに行かないと進まない  
時間がなくなるとできる人に作業が回される  
できる人に作業が集中する
- いつまでもできる人が育たない  
変更分のテスト設計をやっていない  
→どうして？  
やらなければならないと思っていない  
→その結果どのような困り事が発生している？  
小さい案件ほど単純なテスト漏れ（探索的テスト時に）が発生する
- テスト技法を使ったためしがない  
→その結果どのような困り事が発生している？  
技法の勉強会には出ているが実務では使っていない  
結果的に経験則的なテストになる（スッカスカand/orムダにフルフル）
- テストケースの書き方があいまい  
→具体的にどのような書き方になっている？  
人により手順（操作手順）の書き方がバラバラ（粒度や表現が揃わない）  
→その結果どのような困り事が発生している？  
レビューしにくい／細かい修正が多くなる
- テスト設計からテスト仕様書に落とし込むプロセスがあいまい  
→具体的にどのようなプロセスになっている？  
目的と手段の連携を見たいのに、段階的詳細化がめんどくさいと思われる  
→その結果どのような困り事が発生している？  
端折られるとレビューできなくなる／聞かないといいも悪いもわからない  
ユニットテストの狙い（粒度）が作り手によってことなる  
→ユニットテストの狙いの事例（どのように異なる内容なのか？）  
残された情報では以前に作業した方のユニットテストの意図が分からない
- その結果どのような困り事が発生している？  
テストコードの修正箇所特定が難しい  
テストをやってみて、想定と違うテストになっていると気づく場合もある
- テスト仕様書の役割があいまい  
→具体的にどのような状態をあいまいと言っている？  
機能に対する必要なテストケースを記載することになっている  
回帰テストができるようにテストケースを羅列していく  
テスト詳細設計結果レベルで記載する人とテスト実装レベルで記載する人がいる  
→その結果どのような困り事が発生している？  
重複するテストケースが出てきたりする
- GUI単体のテストは分冊したい  
→どうして分冊したいの？分冊していないとどのように困るの？  
他のテストフェーズとの関係性でシステムテスト仕様書には  
GUI、制御、ストレージ、通信などサブシステム単位に必要なテストをすべて記載している  
回帰テストを実施する際にシステムテスト仕様書に記載された  
すべてのテストを実施するような非効率な対応が発生してしまう
- GUI単体のテストはどれぐらいの粒度のテスト  
GUI単体のテストと統合確認のシナリオテストの境界がわからない

**問題意識の背景＝リアルなQCD問題関連事象や困り事を掘り下げて把握する→改善目標設定に役立つ**

# 実存するQCD問題関連事象・困り事の列挙と分類

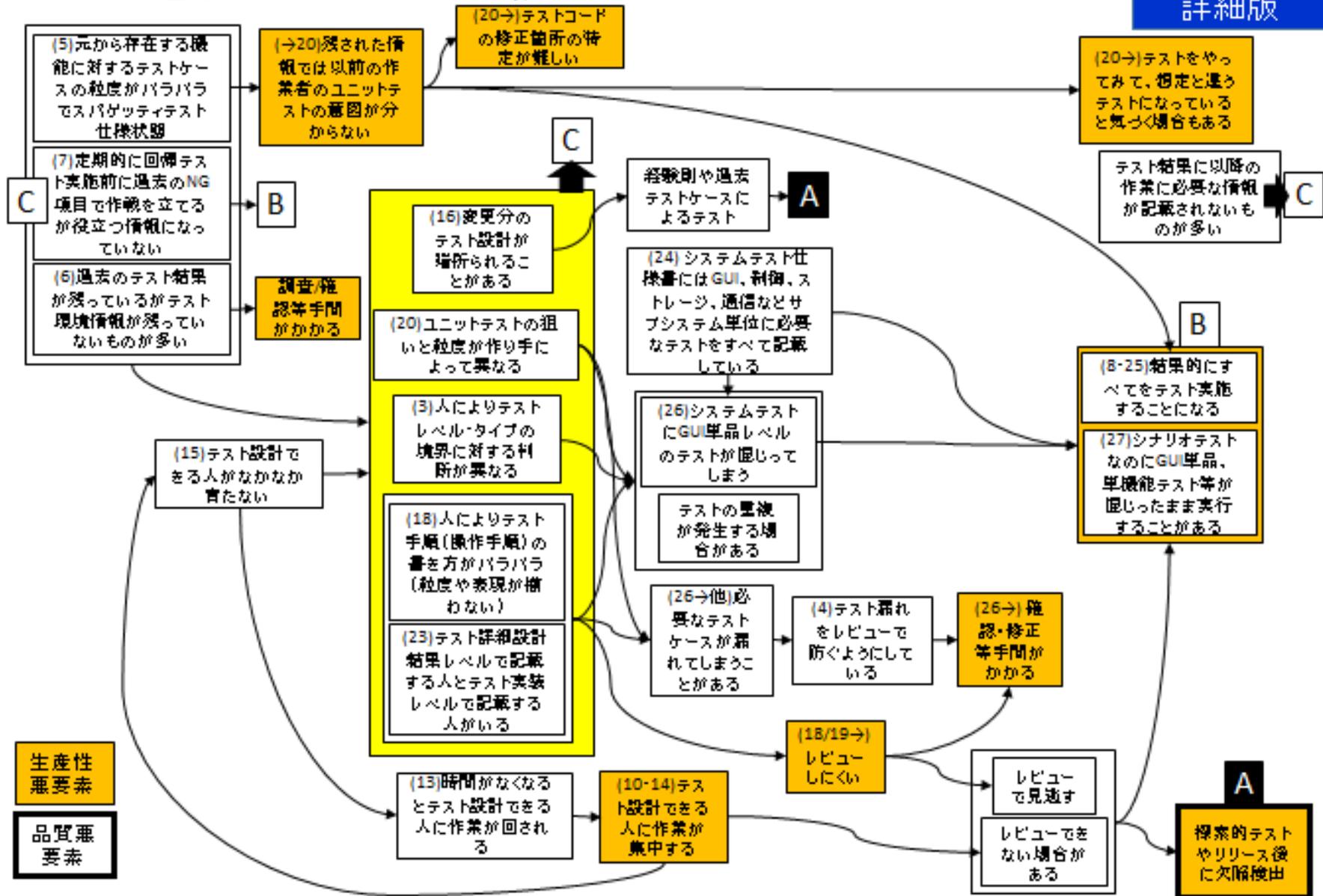
## 実際のふりかえり結果

## 整理・分類結果



# QCD問題関連事象・困り事による構造分析(詳細)

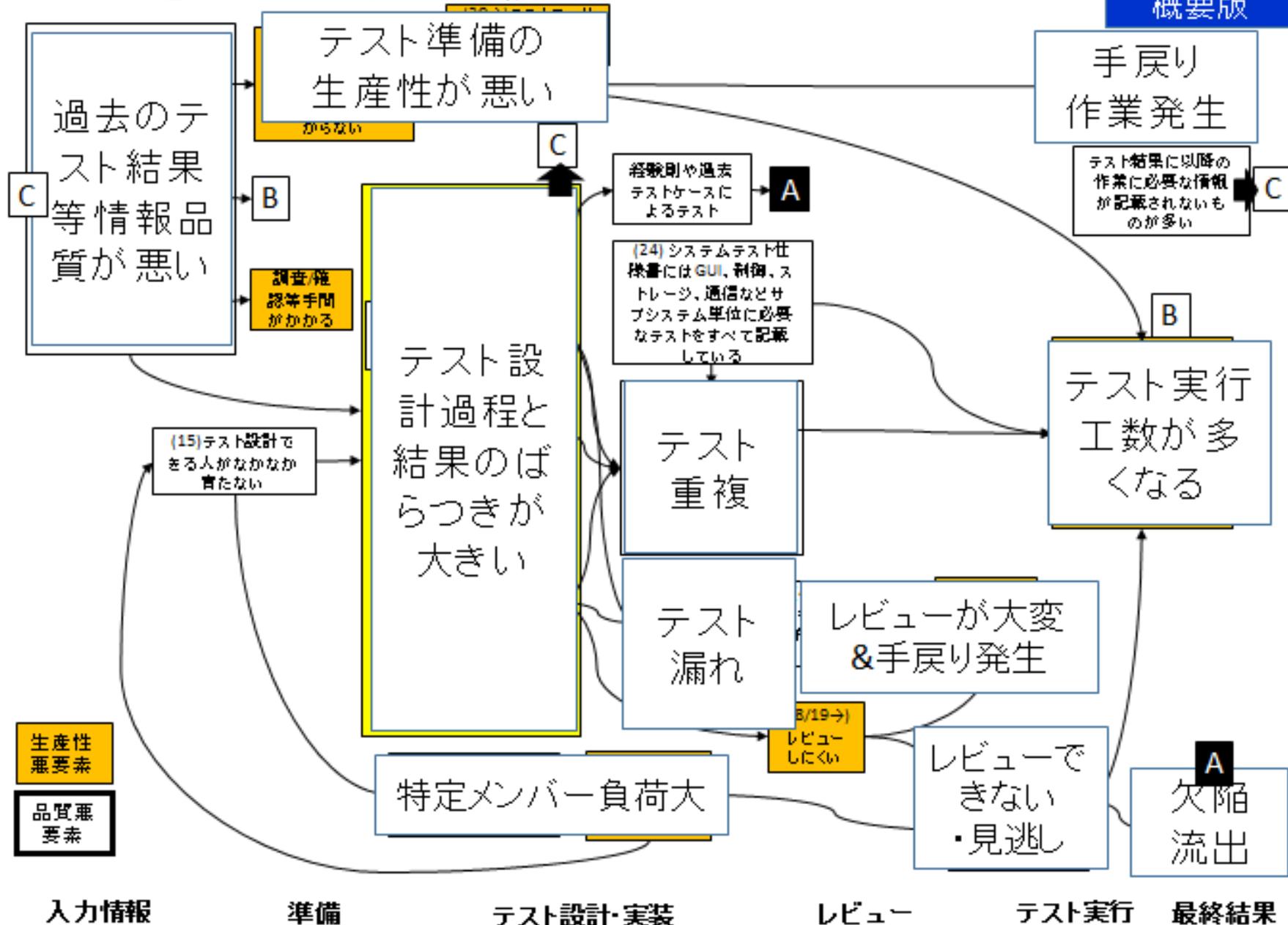
詳細版



入力情報      準備      テスト設計・実装      レビュー      テスト実行      最終結果

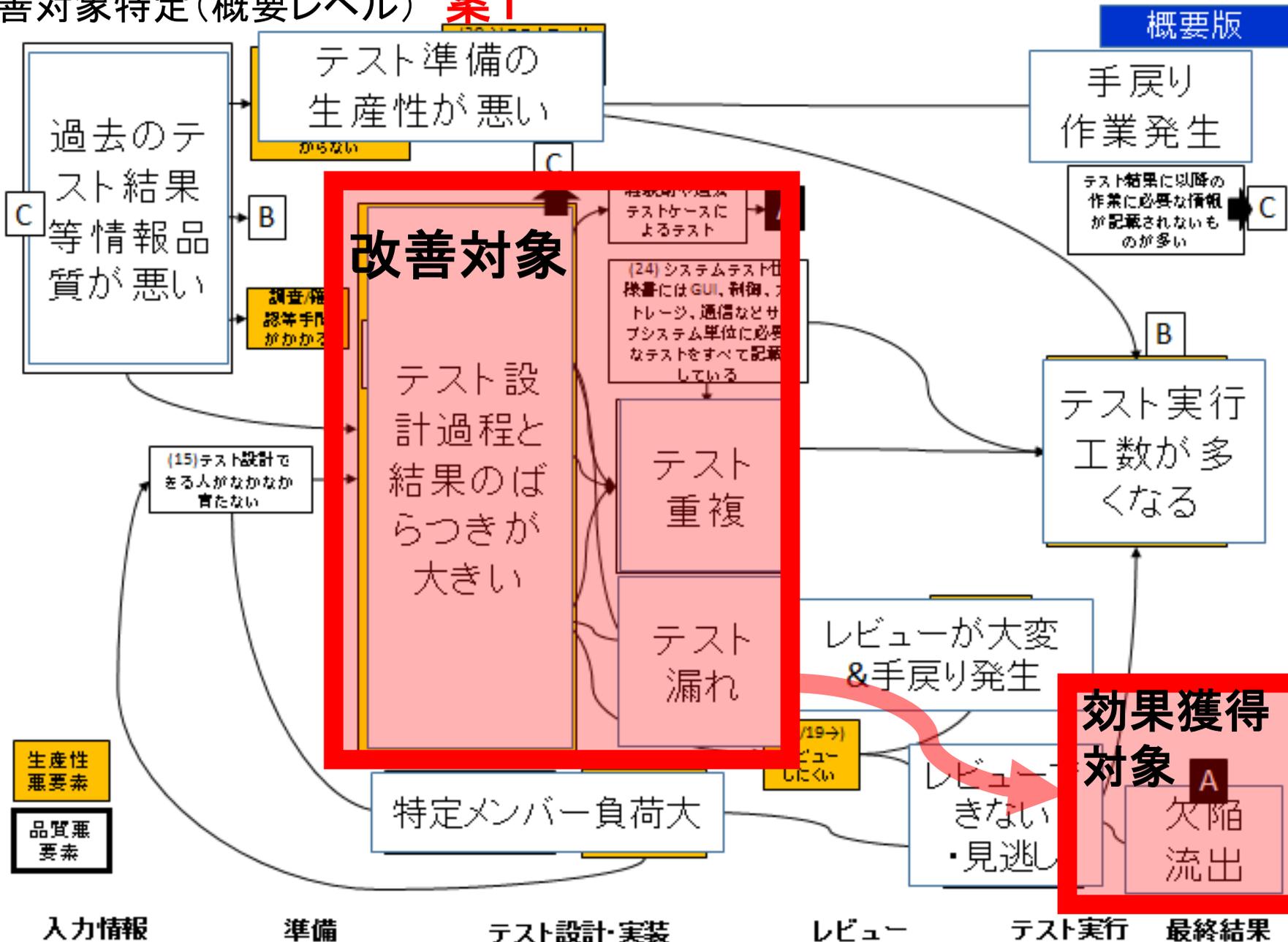
# QCD問題関連事象・困り事による構造分析(概要)

概要版





# 改善対象特定(概要レベル) 案1



入力情報      準備      テスト設計・実装      レビュー      テスト実行      最終結果

# 改善対象特定(概要レベル) 案2

概要版

**改善対象**  
過去のテスト結果等情報品質が悪い

テスト準備の生産性が悪い

手戻り作業発生

テスト結果に以降の作業に必要な情報が記載されないものが多い

経験則や過去のテストケースによるテスト

(24) システムテスト仕様書にはGUI、制御、ストレージ、通信などサブシステム単位に必要なテストをすべて記載している

テスト重複

**効果獲得対象**  
テスト実行工数が多くなる

(15) テスト設計でできる人がなかなか育たない

テスト設計過程と結果のばらつきが大きい

エラーが大変戻り発生

レビューできない・見逃し

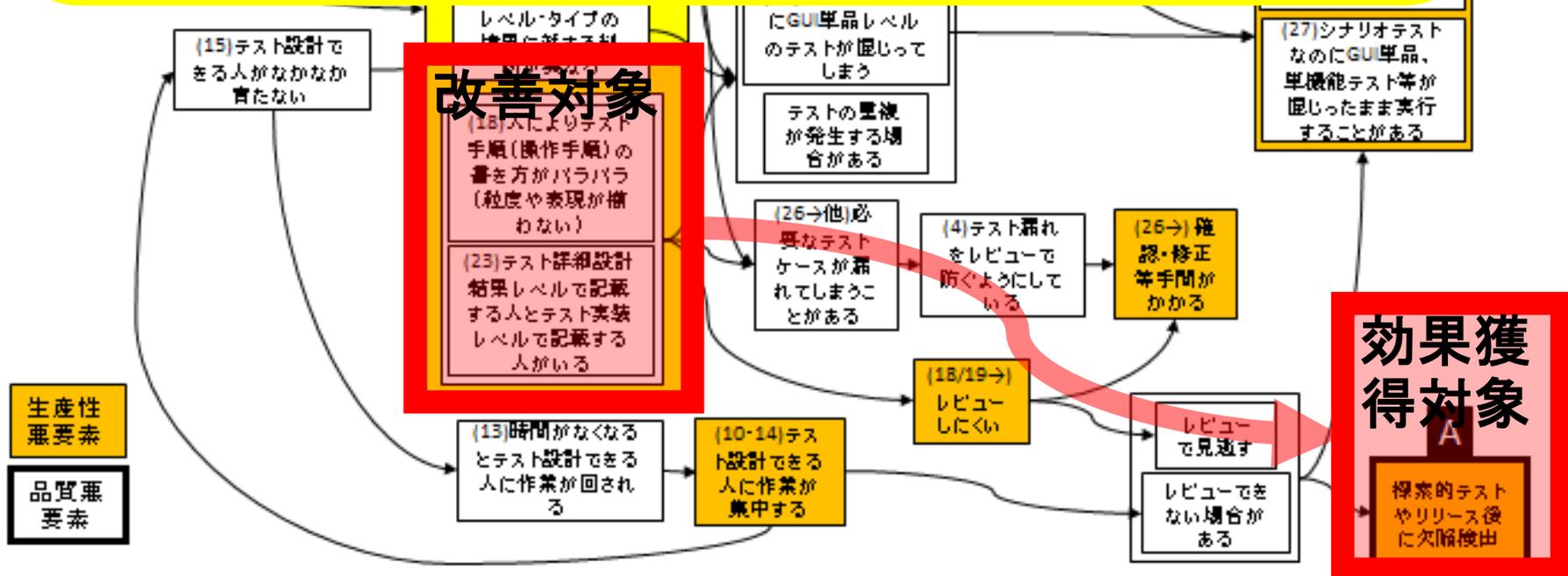
欠陥流出

発生頻度や欲しい効果などから案1を採用することに

- 生産性悪要素
- 品質悪要素

入力情報      準備      テスト設計・実装      レビュー      テスト実行      最終結果

**案1**をベースに詳細レベルで改善対象を掘り下げ、さらに対象を絞り込む  
 →改善規模・工数・期間が小さくなり、短期間で結果が把握できる&効果を実感しやすくなる／失敗した場合の被害も最小に



# 改善施策要件候補洗い出し(TPI Next Practices)

## 11. テストウェア管理

### コントロールレベル

- 1 テストベースとテスト対象およびすべてのテストウェアを、名前とバージョンで特定している。
- 2 各テストケースを、テストベース文書に明白な方法で関連付けている。
- 3 テストチームは、テストウェアの管理下のすべてのアイテムにアクセスできる。
- ④ テストウェア、テストベース、テスト対象の扱い方が明確に規定され、テストチームに伝えられている。

### 効率化レベル

- 1 テストベースとテスト対象およびすべてのテストウェアを、名前とバージョンで参照できる。
- 2 テストケースと要件のトレーサビリティが確保されている。
- ③ テストウェア管理は、論理的な補完構造と、役割および権限の構造によって支えられている。

### 最適化レベル

- 1 テストプロジェクト終了後にどのテストウェアを保持するかをテストプロジェクト開始時に合意し、テストプロジェクト実施中に見直している。
- 2 再利用に備えたテストウェア保持に関するガイドラインが入手可能な状態にあり、テストウェアの再利用を測定している。
- 3 プロジェクト終了時に保守に引き渡すテストウェアが、保守されないテストウェアと容易に分離できる。

## 12. 手法の実践

### コントロールレベル

- ① テストプロセスは、明文化されたテスト手法に従っている。テスト手法には、一連のテスト活動、テストプロジェクトの成果となるテストプロダクト、作業の途中で発生する追加要求について記述している。
- 2 テスト手法は、プロジェクトが用いている開発手法に適合している。
- 3 テストプロジェクトにとって、実装したテスト手法が実用的なものであると認識されている。

### 効率化レベル

- ① テスト手法には、すべてのテスト活動に関するゴールと役割、および用いるべき技法と事前条件が明文化されている。
- ② 完全かつ包括的なテンプレートの一式が、テスト手法の一環として提供されている。
- 3 テスト手法の各要素について、必須／条件付き／任意のいずれかが記載されている。
- 4 必須と条件付きの要素について、実践事例がある。

### 最適化レベル

- 1 テストチームは、テスト手法について組織的にフィードバックしている。
- 2 実装したテスト手法を継続的に強化し、改善している。

## 14. テストケース設計

### コントロールレベル

- ① テストケースを論理レベルで記録している。  
テストケースには、以下の説明項目を含む。
- ② a) 開始時の状況  
b) 変更プロセス=実施するテストアクション  
c) 予測される結果
- ③ テストケースにシステムの詳細な振る舞いを記述することで、テストベースのどの箇所がテストの対象であるかが把握できる。

### 効率化レベル

- ① テストケースは、テスト組織の同僚が見ても理解でき、保守できるものになっている。
- 2 テストケースによって達成できるテストベースのカバレッジレベルが明確である。
- ③ テストケース設計に正式なテスト設計技法を用いている。
- 4 テストケースが設計できないような品質特性のテスト作業には、チェックリストを用いている。

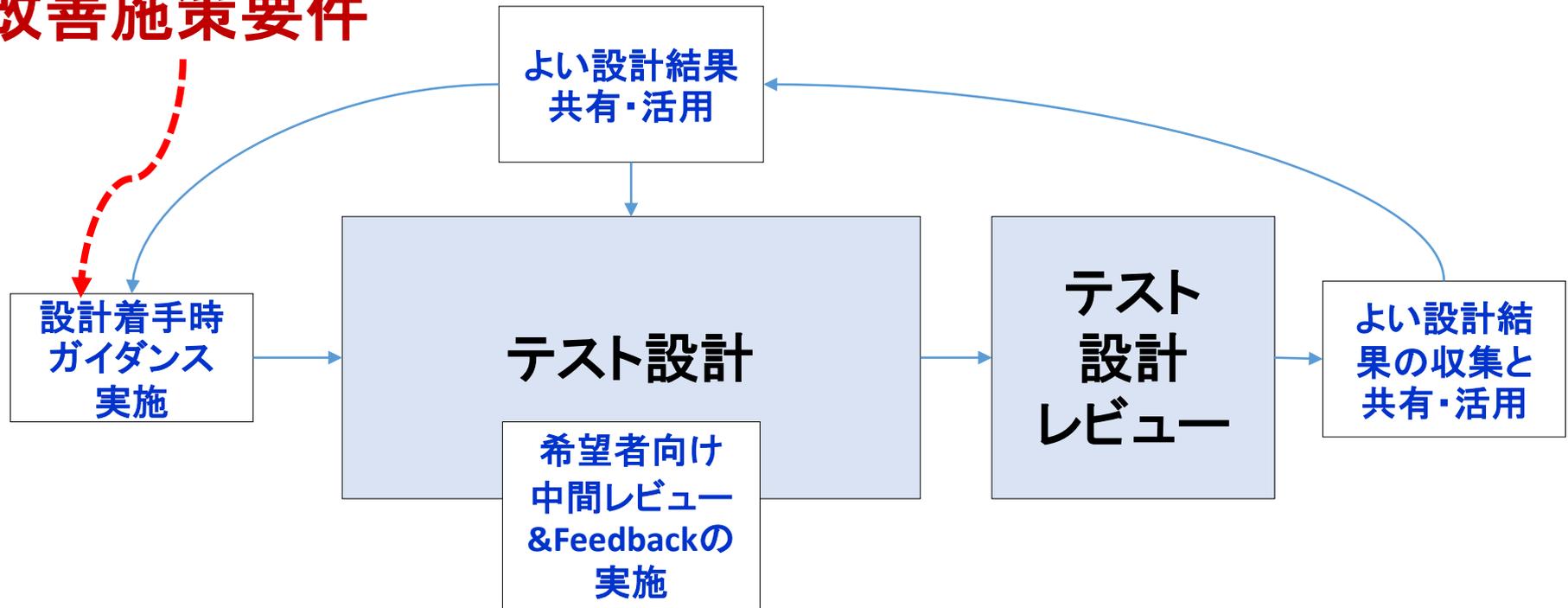
### 最適化レベル

- 1 次のフェーズ(次のテストレベルや本番)で発生した欠陥を分析し、テストケースの正確性や有効性の向上につなげている。
- 2 テストケースそれぞれの妥当性と保守性についてチェックし、評価している。
- 3 テスト設計技法を、将来さらに再利用するために評価し、調整している。

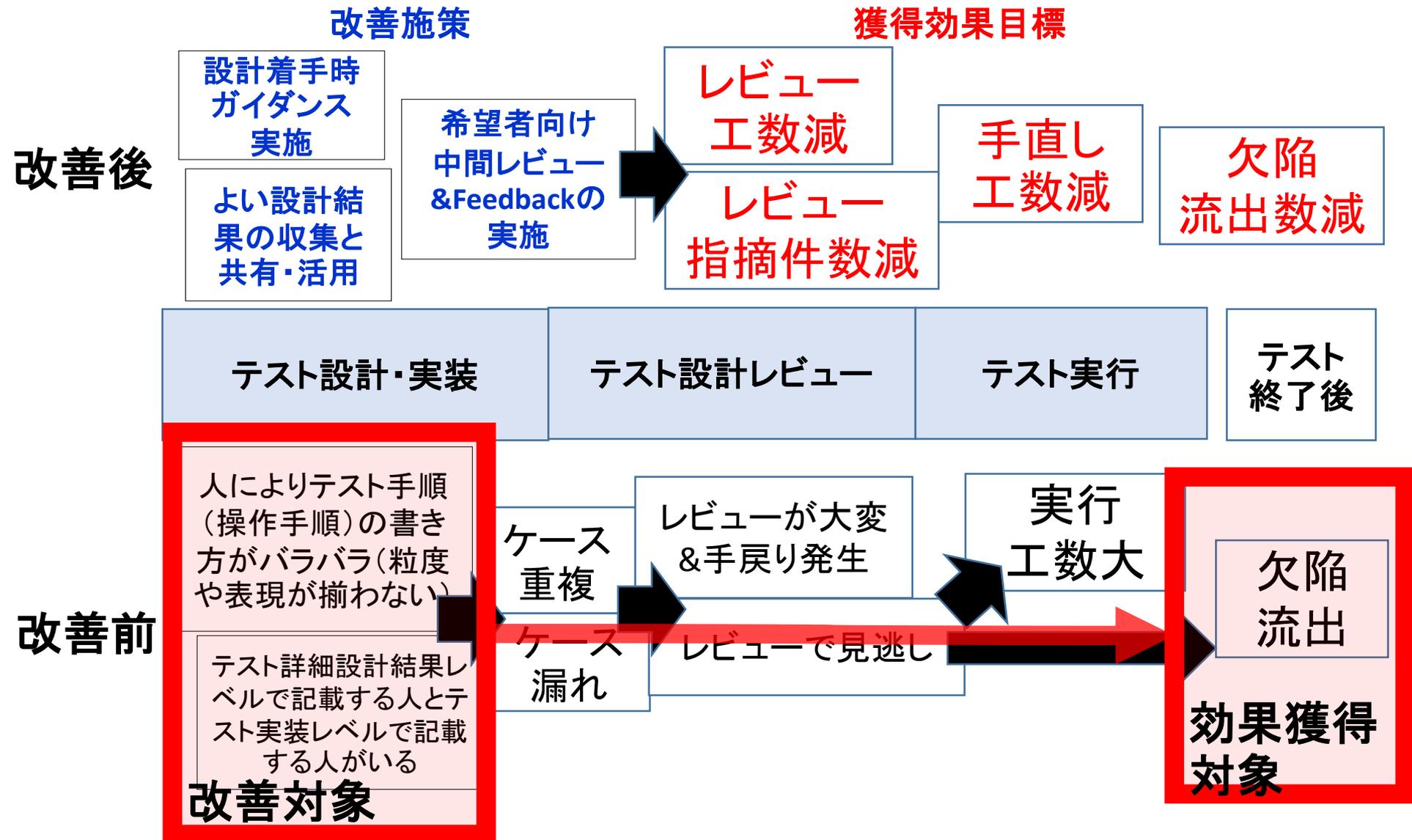
**プロセスモデルのプラクティスは、ものづくりへの効果と効率を高めるための施策要件の集合体**

# 改善施策要件を参考に 改善施策を明確化

## 改善施策要件



# 改善施策と獲得効果目標の設定

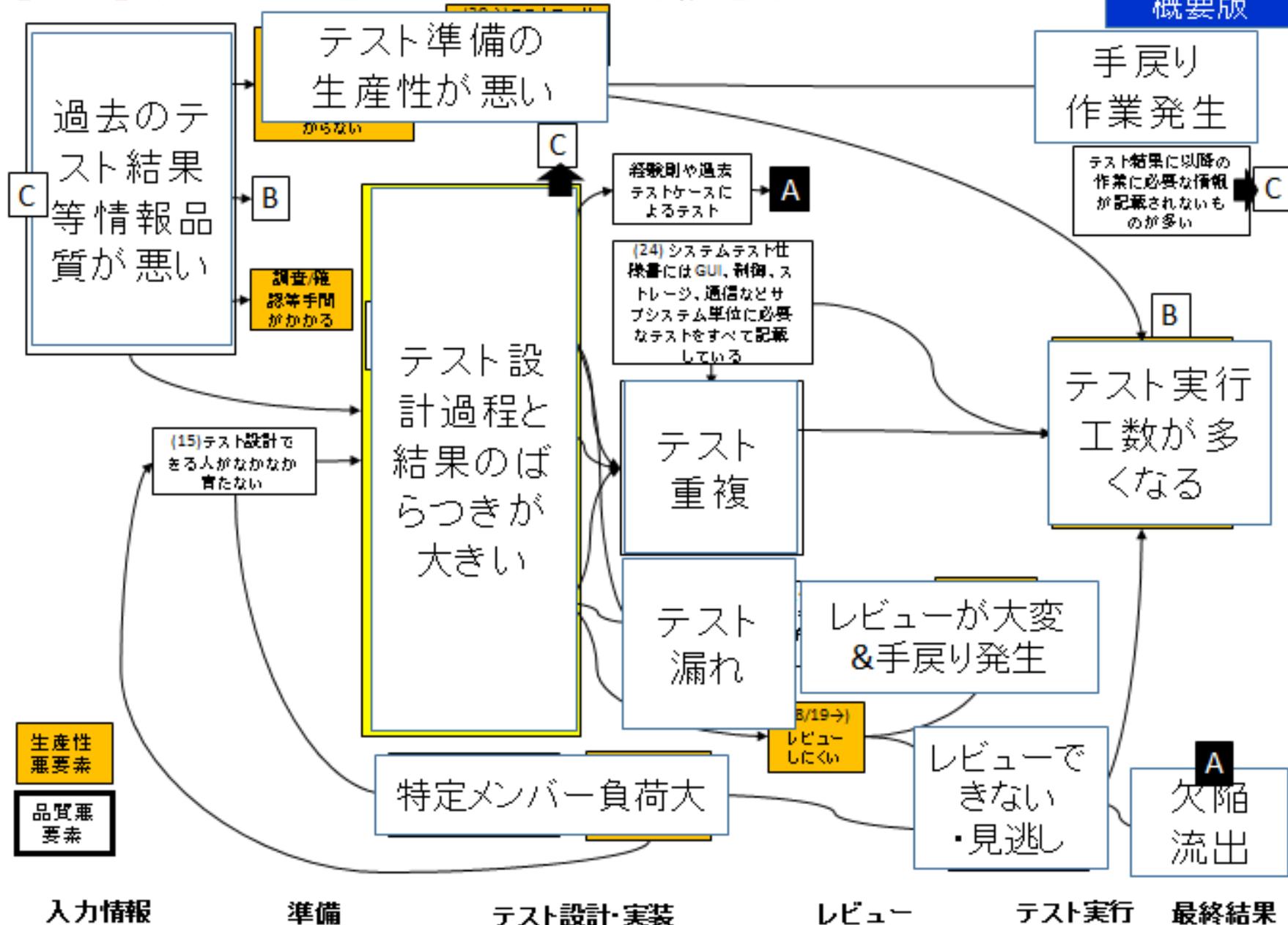




# 別解例

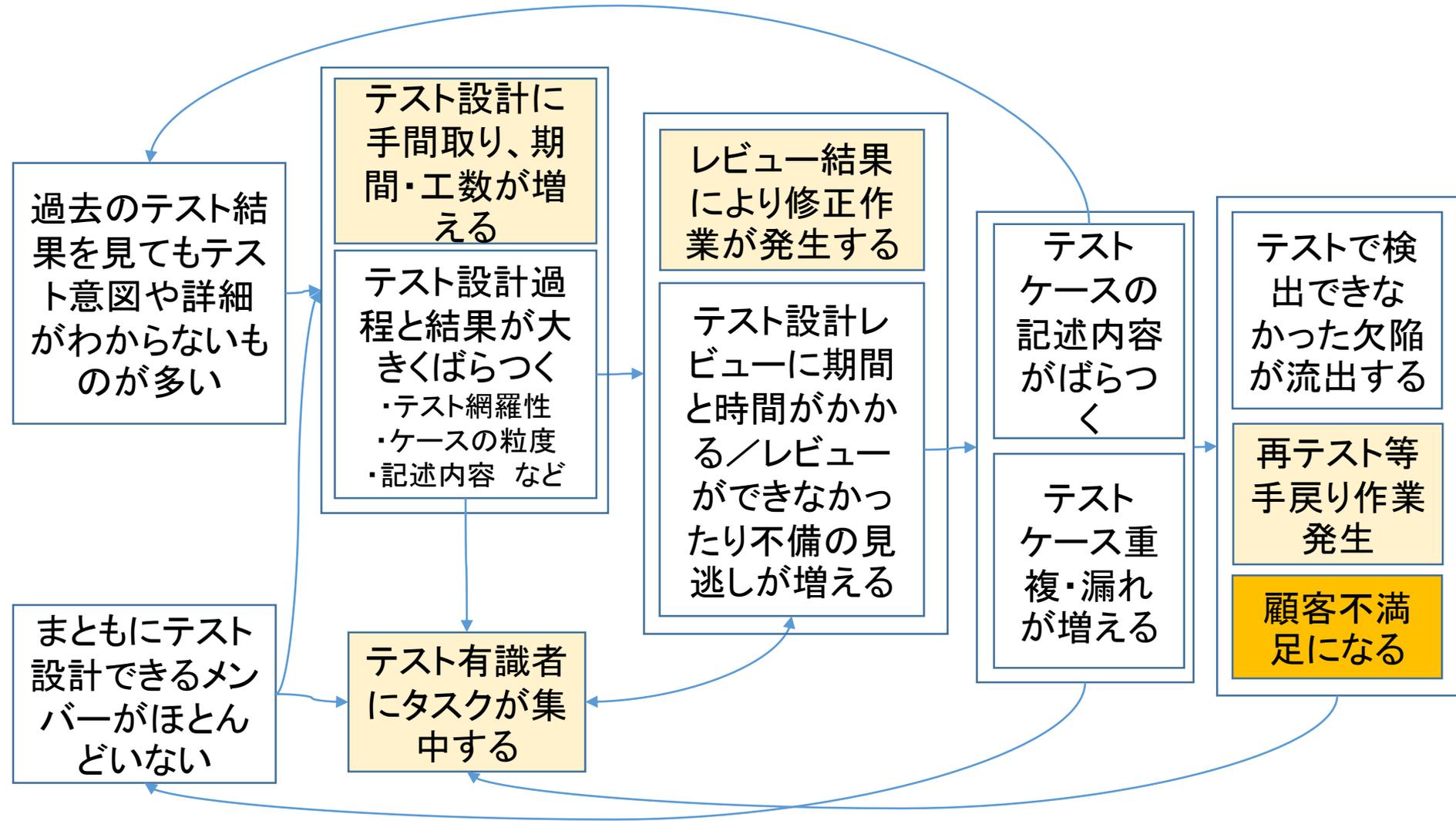
# 【再掲】QCD問題関連事象・困り事による構造分析(概要)

概要版

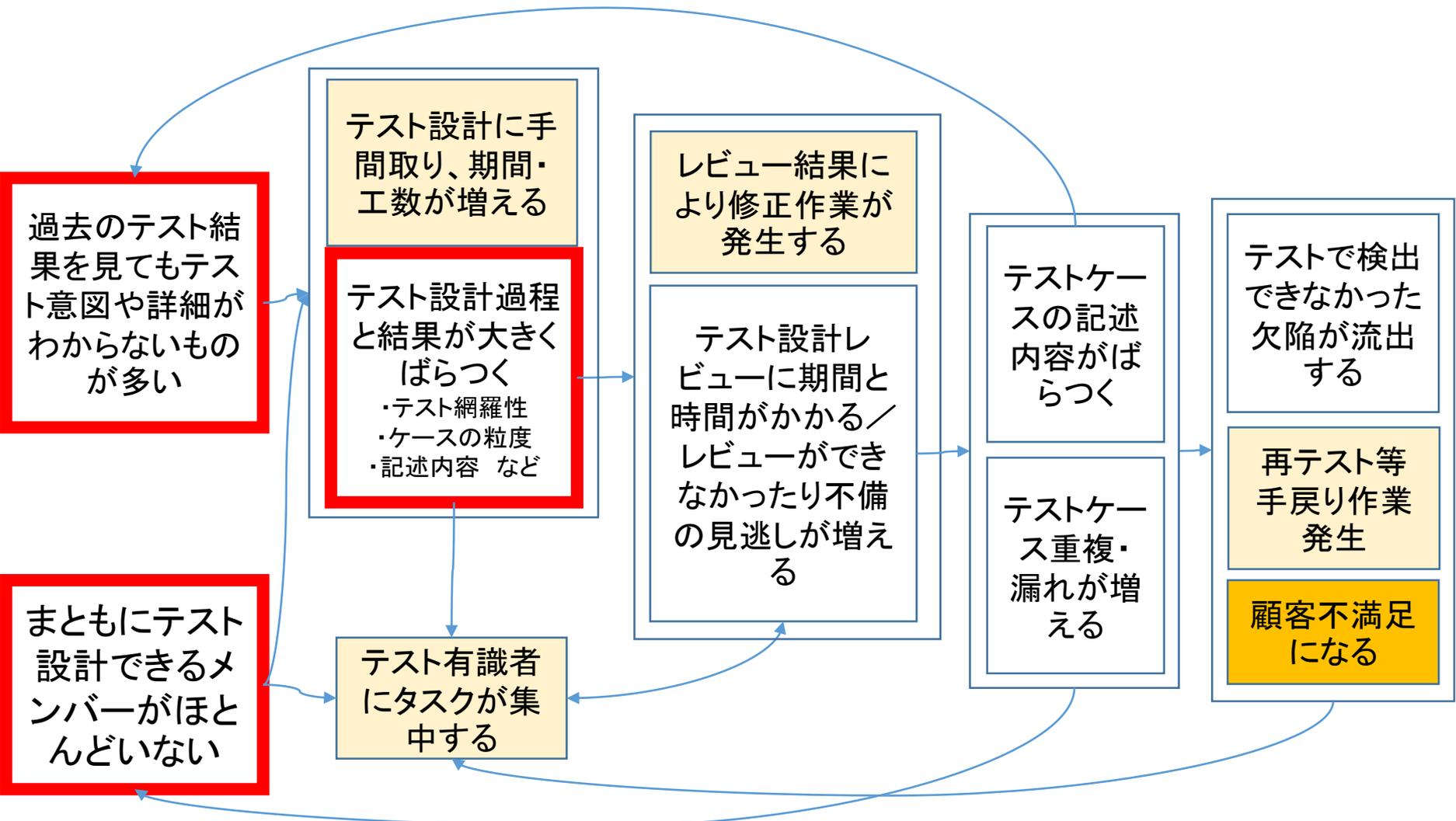


生産性  
悪要素  
品質悪  
要素

# 問題構造図(トラブルモデル)を単純化して再整理



# 改善すべきターゲットの候補を決める



# 効果がありそうな施策の方向性を検討 ～問題解決力があればこれでOK

テストフレーム記述ガイド、テスト設計サンプル提供／相応しいか過去事例を特定し活用するなど

過去のテスト結果を見てもテスト意図や詳細がわからないものが多い

様式や記述ガイド、サンプル、事例の提供など

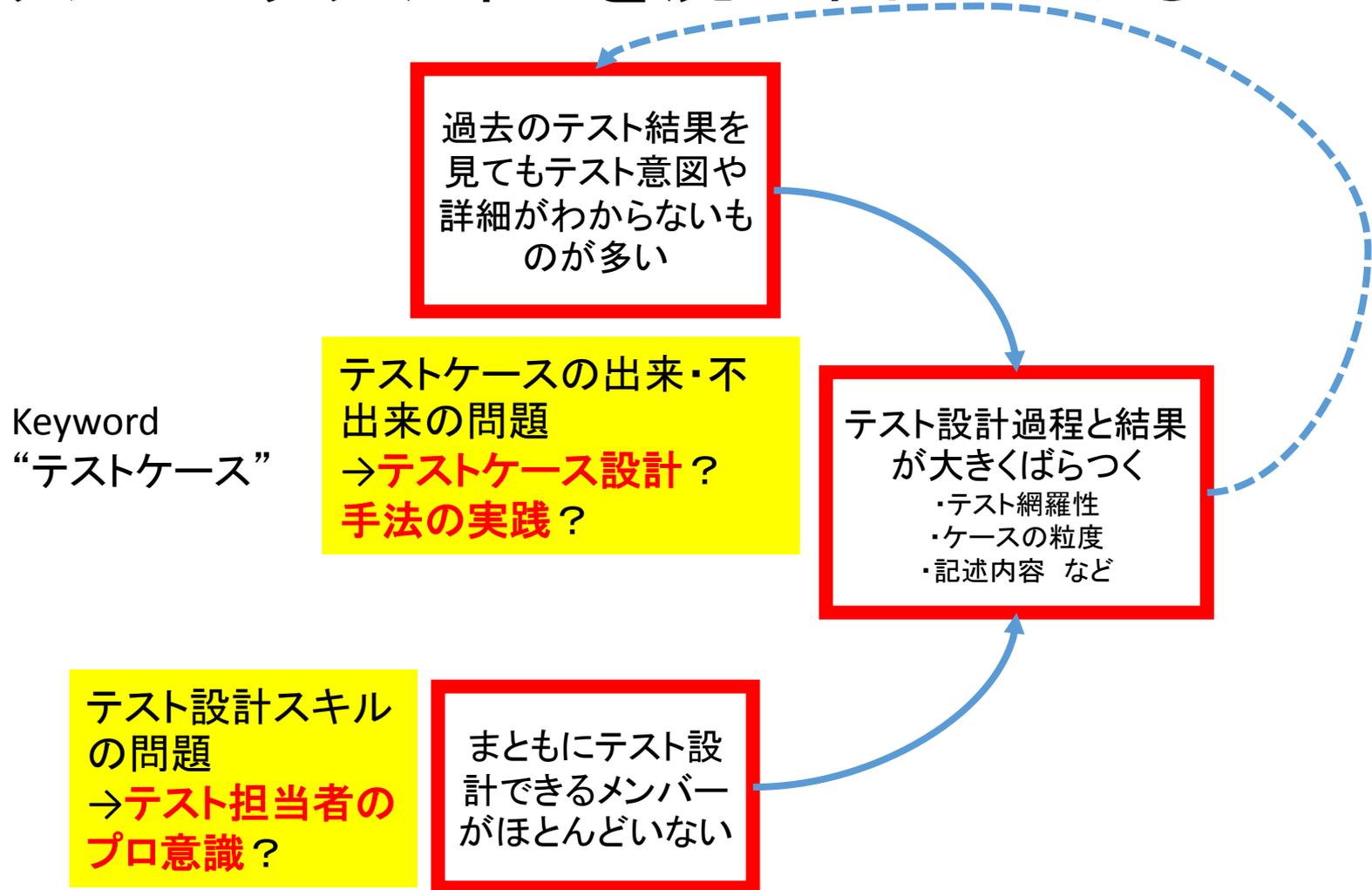
テスト設計過程と結果が大きくばらつく

- ・テスト網羅性
- ・ケースの粒度
- ・記述内容 など

スキルの話……事前説明、オリエンテーション、トレーニング実施など

まともにテスト設計できるメンバーがほとんどいない

# これらに関係しそうなTPI NEXTのキーエリア・プラクティスを洗い出してみる



11C4: テストウェア、テストベース、テスト対象の扱い方が明確に規定され、テストチームに伝えられている

12C3: 実装したテスト手法が実用的だと認識している

13C4: テスト担当者の特定テストスキルや一般的なIT能力を定期的に評価する

14C3: テストベースのどの箇所がテストの対象であるかが把握できるようにテストケースにシステムの詳細な振る舞いを記述する

11C3: テストチームはテストウェア管理下のすべてのアイテムにアクセスできる

12C2: プロジェクトの開発手法に適合したテスト手法を用いる

13C2: テスト担当者は理解しているテスト手法を採用する

13C1: テスト担当者は、テストに特化した訓練を受講済or体系的なテスト実行経験がある

※網羅的に洗い出していないことに注意

12C1: テストプロセスは明文化されたテスト手法に従う

### 13. テスト担当者のプロ意識

11C1: テストベース、テスト対象、すべてのテストウェアを名前とバージョンで特定する

11C2: 各テストケースを、テストベース文書に明白な方法で関連付けている

### 12. 手法の実践

### 14. テストケース設計

14C2: テストケースに以下の内容を含む a) 開始時の状況 b) 実施するテストアクション c) 予測される結果

14C1: テストケースを論理レベルで記録する

### 11. テストウェア管理

※テストウェア: テストプロセス中に生み出される成果物

Keyword “テストケース” でヒット

J  
I  
H  
G  
F

11E3: テストウェア管理は、論理的な補完構造と、役割および権限の構造によって支えられている

12E4: 必須と条件付きの要素について、実践事例がある

12E3: テスト手法の各要素について、必須／条件付き／任意のいずれかが記載されている

11E1: テストベースとテスト対象およびすべてのテストウェアを、名前とバージョンで参照できる

11E2: テストケースと要件のトレーサビリティが確保されている

13E3: テストに関わる要員は、プロジェクトの他のスキルグループと良好な関係を築き、自身の仕事を楽しんでいる

13E4: テストのタスクが期待どおりに定義され、割り振られ、実行されている

14E4: テストケースが設計できないような品質特性のテスト作業には、チェックリストを用いている

14E2: テストケースによって達成できるテストベースのカバレッジレベルが明確である

14E3: テストケース設計に正式なテスト設計技法を用いている

### 11. テストウェア管理

### 12. 手法の実践

12E2: 完全かつ包括的なテンプレートの一式が、テスト手法の一環として提供されている

### 13. テスト担当者のプロ意識

13E1: テスト担当者がテスト資格を持っている

13E2: テスト担当者は技法選択理由を論理的根拠で説明できる

### 14. テストケース設計

12E1: テスト手法には、すべてのテスト活動に関するゴール、役割、用いるべき技法、事前条件が明文化される

14E1: テストケースはテスト組織の同僚が見ても理解でき、保守できるものになっている

※網羅的に洗い出していないことに注意



M

## 11. テストウェア管理

1103: プロジェクト終了時に保守に引き渡すテストウェアが、保守されないテストウェアと容易に分離できる

1102: 再利用に備えたテストウェア保持に関するガイドラインが入手可能な状態にあり、テストウェアの再利用を測定している

1101: テストプロジェクト終了後にどのテストウェアを保持するかをテストプロジェクト開始時に合意し、テストプロジェクト実施中に見直している

1202: 実装したテスト手法を継続的に強化し、改善している

1201: テストチームは、テスト手法について組織的にフィードバックしている

## 12. 手法の実践

1303: テスト担当者は、テストの仕事に対する責務と説明責任を持ち、プロセスの継続的改善に努めている

1403: テスト設計技法を、将来さらに再利用するために評価し、調整している

**※網羅的に洗い出していないことに注意**

## 13. テスト担当者のプロ意識

1302: テストの職務が、組織の人事管理、もしくは個人のキャリア開発の一部になっている

1301: テスト担当者は、SIGやセミナーの積極的参加、論文読むなどにてスキル保持のため最新情報を常に取り入れる

## 14. テストケース設計

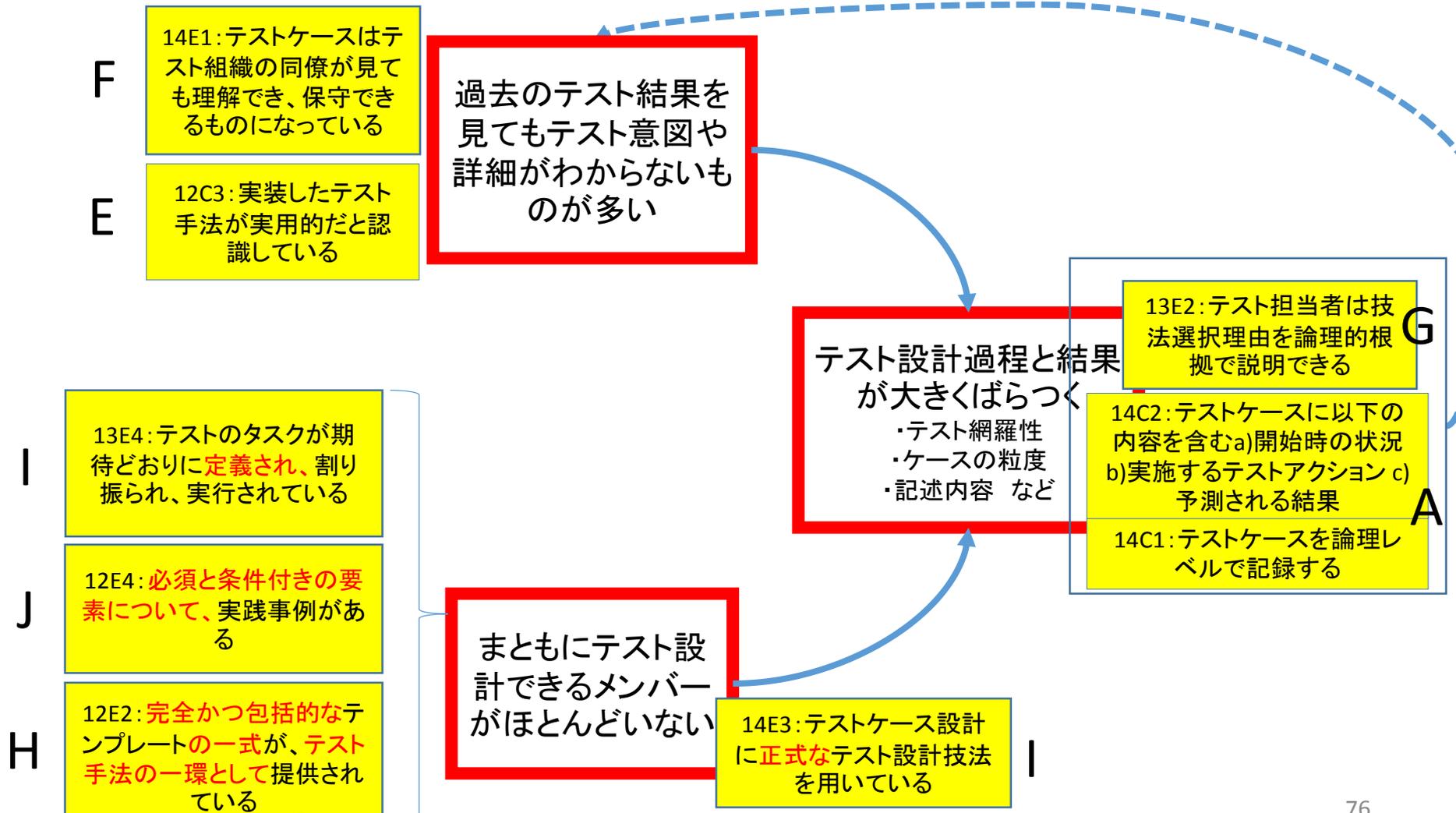
1402: テストケースそれぞれの妥当性と保守性をチェックし、評価している

1401: 次のテストレベルや本番で発生した欠陥を分析、テストケースの正確性や有効性の向上につなげる

L

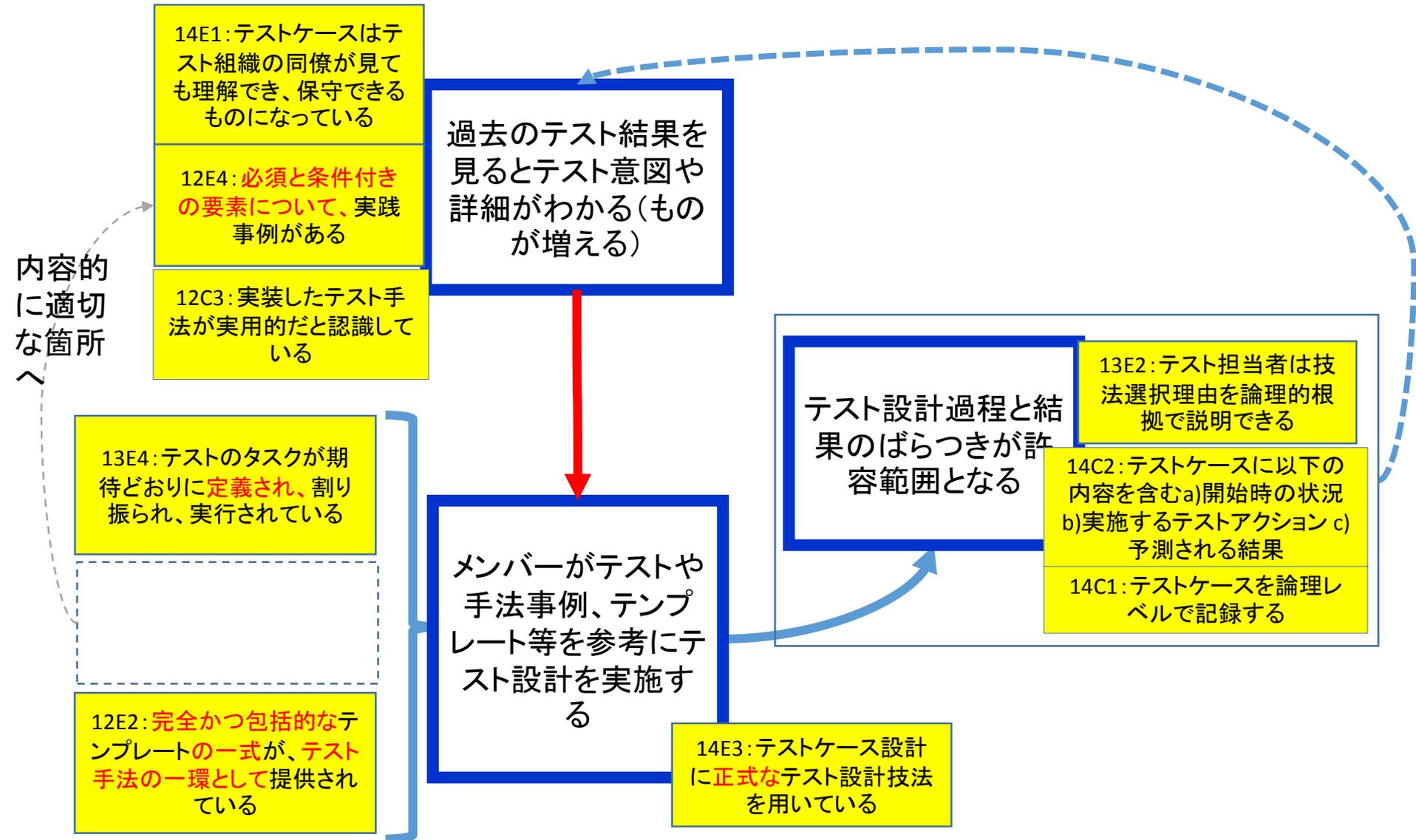
K

# 関連するプラクティスを活用して施策要件を検討してみる

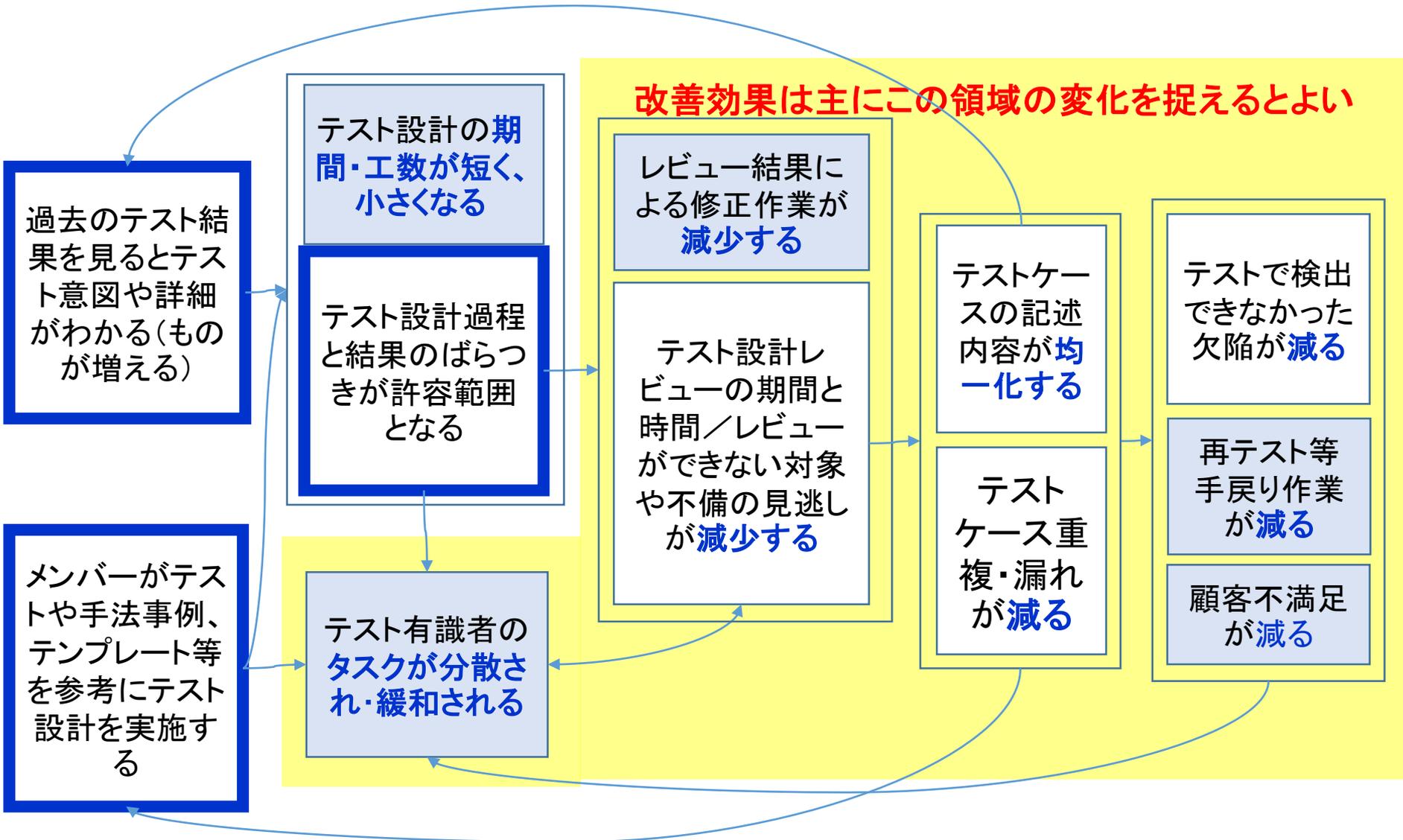




# 最終形～改善シナリオ



# 改善シナリオの全体像



# 当アプローチのSTEP全体像(TPI Next活用箇所と利点・効果)

| セルフアセス終了後の実施STEP         | TPI Next活用                          | 利点や効果   |
|--------------------------|-------------------------------------|---|
| メンバーふりかえりによるテストへの問題意識の共有 |                                     | メンバーの問題意識から始めると当事者意識が高まる+改善効果を実感しやすい  |
| テストプロセス全体像と要素関係性の把握      | <b>テストプロセス全体像モデル(TP全体像M)</b>        | 全プラクティスでの関係性把握は困難／プロセスエリアレベルなら把握しやすい  |
| 関係者の問題意識分析               | <b>TP全体像Mにメンバーの問題意識を割り付け集中箇所を特定</b> | 問題意識が集中しているところに着眼すると関係者を巻き込みやすく、改善効果も高くなる                                     |
| 実在するQCDS上の問題点把握          |                                     | 通常「QCD問題事象の解決手段」を問題として誤認識していることが多い／背後に隠れているQCD問題事象を把握することで実感しやすい改善効果や目標を設定できる |
| 困り事・問題点による構造分析           |                                     | 事象・事実レベルでQCD問題群を構造的に把握すると腹落ちしやすい  |
| 改善対象特定                   |                                     | まずは大枠で改善対象と目指す効果を把握(特定)すると考えやすい   |
| 改善対象絞込み                  |                                     | 絞り込むことで改善リスクを最小化しつつ効果を実感しやすくする  |
| 改善施策要件候補洗い出し             | <b>改善施策要件洗い出し</b>                   | プラクティスは汎用的→施策要件として活用するのが適切  |
| 具体的施策と獲得効果目標の設定          |                                     | 改善をシナリオ化し成功確率を高める   |

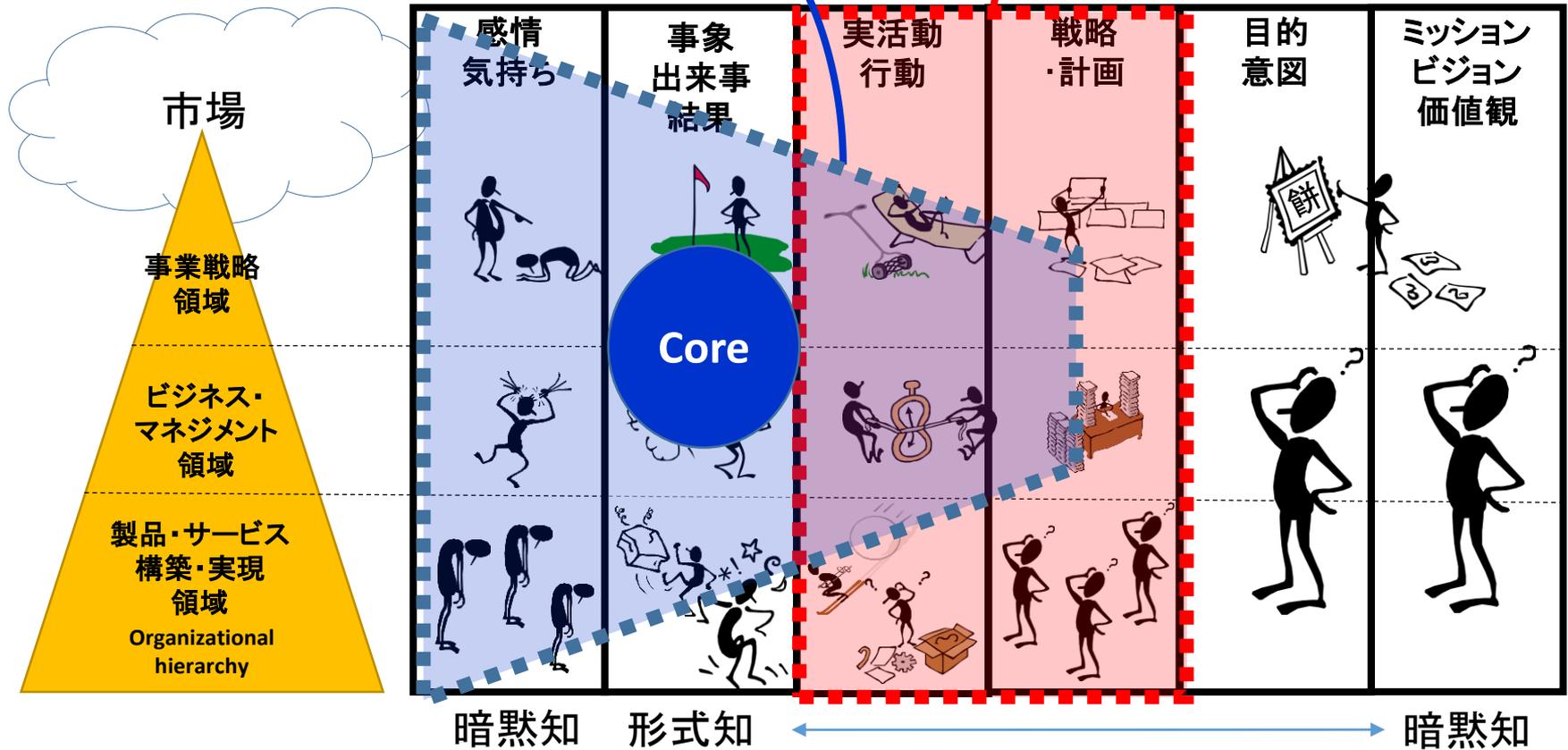
# プロセスモデル適用の典型的失敗例との比較

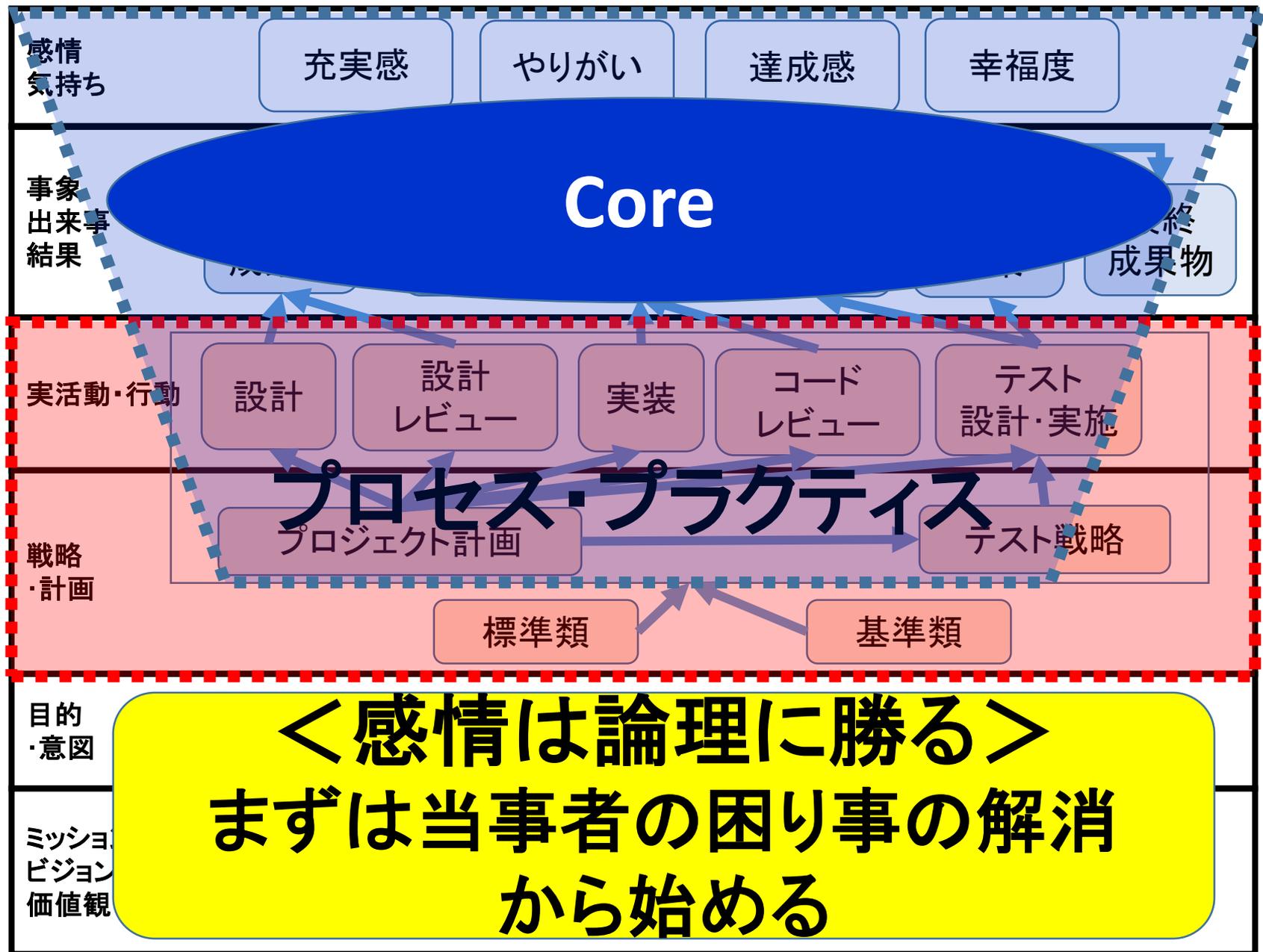
| 主な改善過程          | プロセスモデル適用の典型的失敗例   | 今回のアプローチ   |
|-----------------|--|--|
| 現状把握<br>現状分析    | 16キーエリア146プラクティスフルセルフアセスメント→工数大になるため代表者(例:リーダー)が実施する傾向大=実態を反映しない結果になる可能性が高い  | TP全体像M上の集中箇所に関わるQCD問題関連事象・困り事を洗い出し、構造分析して状況を明確にする／ふりかえり結果などメンバーが感じている問題意識から始められる |
| 改善対象特定          | アセスメント結果から有効な改善対象を特定するのは有識者でなければ困難→代表者の一声で決める、またはモデル推奨に従うが、どちらもうまくいかないケースが多い | QCD問題関連事項・困り事の構造分析結果から改善対象を概要レベルで特定し、さらに詳細レベルで把握する／メンバーの問題意識をベースにした改善対象が特定できる    |
| 改善目標            | アセスメント結果から有効な改善目標を設定するのは簡単ではない→モデル適合を目標に→プロセスが重くなる+効果が実感しにくい                 | QCD問題関連事項・困り事の解消を目標にできる／実務の困り事が減る効果を実感しやすい                                       |
| 改善施策            | プラクティスを直接単品追加しがち→改善するほど重くなる  | 現状のプロセスをベースに改善要件を満たす施策を展開できる   |
| 改善期間・効果<br>実感など | 長期化しやすい・効果実感を得られにくい／失敗と分かるまでに時間がかかる&見直しも大がかりになる                              | 短期間で効果を実感しやすい／失敗した際の見直しも容易   |

# TPI Nextと当アプローチの違い

当アプローチ  
のSCOPE

TPI Next(プロセス  
モデル)のSCOPE





# 当アプローチの特徴

## • 利点

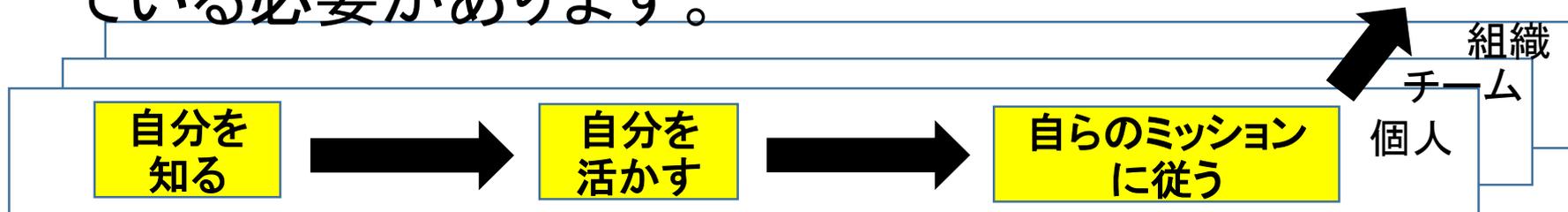
- 146プラクティスのフルアセスメントを行わずとも、16キーエリアの全体像にメンバーの問題意識をマッピングすれば改善対象領域を特定できる。(ふりかえりが実施されていればその情報から始められる)
- メンバーの問題意識をベースに進めることで、チーム全員の当事者意識が高められる&改善効果を実感しやすくなる→メンバーを巻き込んで改善が継続しやすくなる。
- プロセス改善に取り組みながらTPI Nextの理解度向上に応じて活用深度を変化させながら進められる。→より高度な改善活動に段階的にシフトする基盤になる。

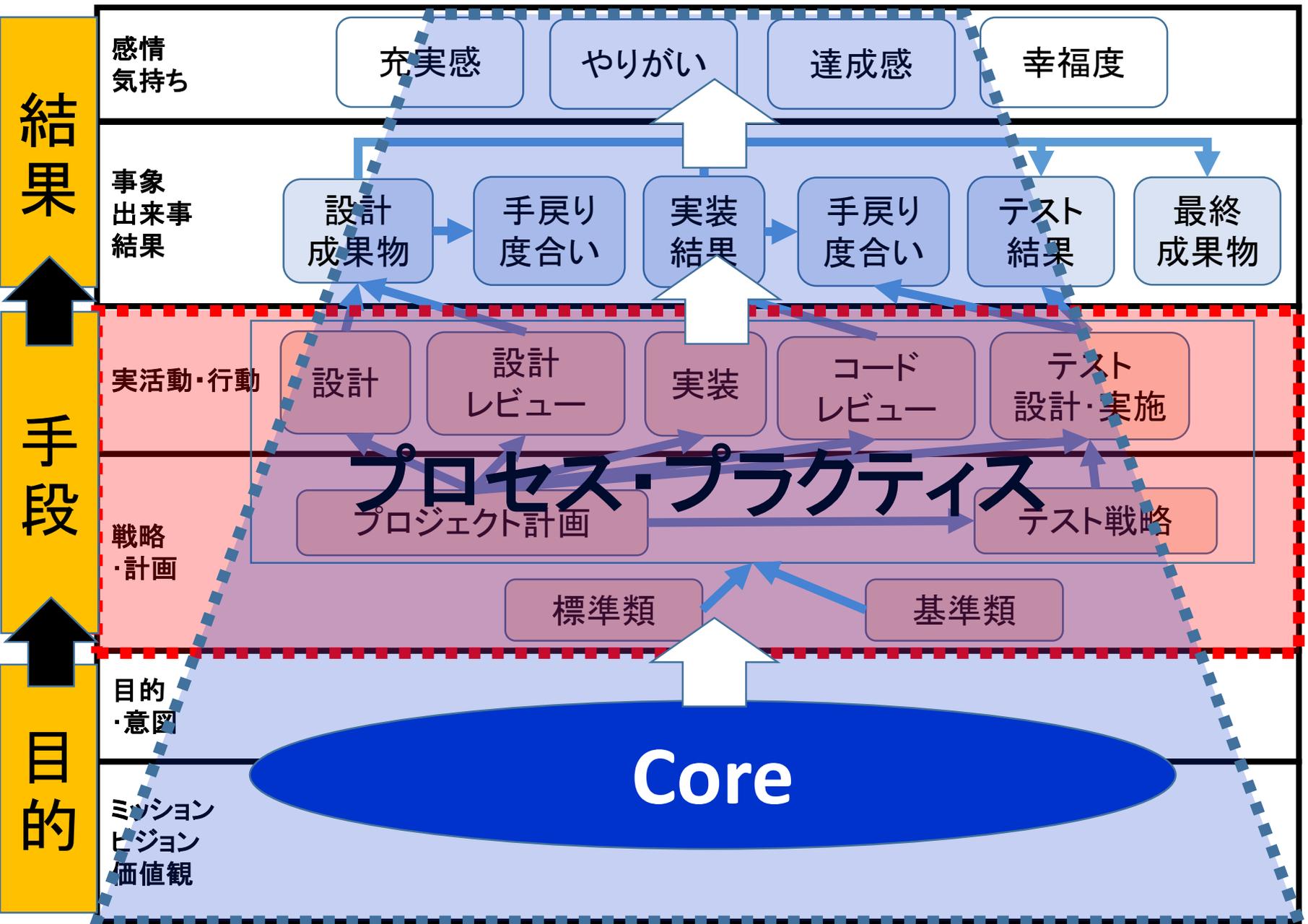
## • 懸念点

- 専門家がアセスメントを行わないため客観性や正確性が低くなる場合もある。
- QCDS問題関連事象・困り事の収集と構造分析実践スキルが必要。当初は有識者の支援を受けるなどの対策は必要。

# そしてその先へ

- 当アプローチは、「自分たちの現在(いま)を知る」ことで実務メンバーの心に火を灯し、自ら走り始めるためのものです。
- 実際に走り始めて徐々に「自らを活かす」ことができるようになったら、その次は「自らのミッションを明確にし、それに従う」ことにチャレンジする必要があります。
- その壁を越えるためには、ミッションや目的達成のために「プロセスモデルを有効活用できる」ようになっている必要があります。





結果

感情  
気持ち

充実感

やりがい

達成感

幸福度

事象  
出来事  
結果

設計  
成果物

手戻り  
度合い

実装  
結果

手戻り  
度合い

テスト  
結果

最終  
成果物

＜自ら明確にしたミッションに従う＞  
混乱を解消できるようになったら、本格的にプロセスモデルを使いこなし、ミッションを成し遂げる

目的  
・意図

Core

ミッション  
ビジョン  
価値観

手段

目的

# 参考文献

- JaSST'18東京 事例発表 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【導入時：改善計画立案編】  
<http://jasst.jp/symposium/jasst18tokyo/pdf/C5-2.pdf>
- TPI NEXT アセスメントツール日本語版  
<http://www.tmap.net/system/files/TPI%20XLS%20versiev2.xls>
- JaSST'17北海道 JaSSTセッション 中山さん、テストプロセスアセスメントやってみたってよ  
<http://jasst.jp/symposium/jasst17hokkaido/pdf/S6-1.pdf>
- 「ソフトウェアプロセス改善手法 SaPID入門」 日科技連出版社  
<http://www.juse-p.co.jp/cgi-bin/html.pl5?i=ISBN978-4-8171-9510-4>  
SaPID: <https://www.software-quasol.com/sapid2-0/>
- 「モチベーション3.0」 ダニエル・ピンク
- ソフトウェア・シンポジウム 2013 in 岐阜 (SS2013)  
プロセスアセスメント結果の現実的・効果的活用方法の提案  
<http://bit.ly/2Fx4IPq> <http://bit.ly/2Dgfx6l>
- Software Testing ManiaX Vol.10寄稿記事  
ソフトウェアテストプロセス評価モデル微考w  
<https://goo.gl/YbgmzG>

# おつかれさまでした。

仲間と一緒に楽しく仕事をして、  
価値あるソフトウェアを提供していきましょう！