

テストの上手いやり方の実践と テストプロセス

11:00-12:30 4th Nov. 2016 JaSST'16 九州

湯本剛

自己紹介 湯本剛 (@yumotsuyo)

経歴

1991年 工作機器メーカーにて生産管理システムの構築メンバーとして働く

1995年 情報機器メーカー系列ソフトハウスにてテストリーダーとして9年間働き
数多くのアプリケーションの開発に携わる

財務会計のパッケージソフト、プリンタ、スキャナ、製造業受託開発（在庫管理、
生産管理など）、顧客向けプロダクト販売Webサイト構築など

2003年 ソフトウェアテストのコンサルタントになる

テストプロセスの改善コンサル、テストツールの導入支援、テストの教育

※ 転社 5回（多分）

現在（2016年11月）

日本ヒューレット・パッカード プリンシパルテストコンサルタント

・ 自社の受託開発案件で現場のテストリーダーもやってる

NPO法人 ASTER理事

・ JaSST東京実行委員、JSTQB技術委員、テスト設計コンテスト審査委員、テストツールWGお世話
係、智美塾生徒

日科技連 SQiPステアリング委員

ISO/IEC JTC1/SC7 WG26 メンバー（ISO29119テストプロセス標準の策定）

筑波大学 システム情報工学研究科 津田研究室在学中

アジェンダ

開発におけるテストの役割

テストプロセス

テストプロセスの改善

テストプロセスに特化した改善のための道具

改善のロードマップを描く

まとめ

開発におけるテストの役割

上手いテストをするとは？

ソフトウェア開発において
テスト工数はどの程度を
占めているでしょう？

A: 24.5%

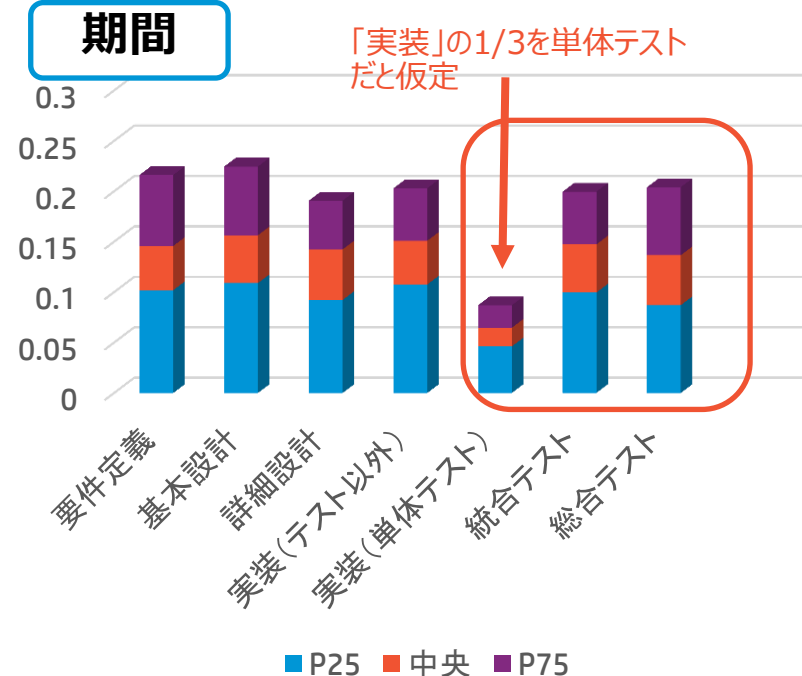
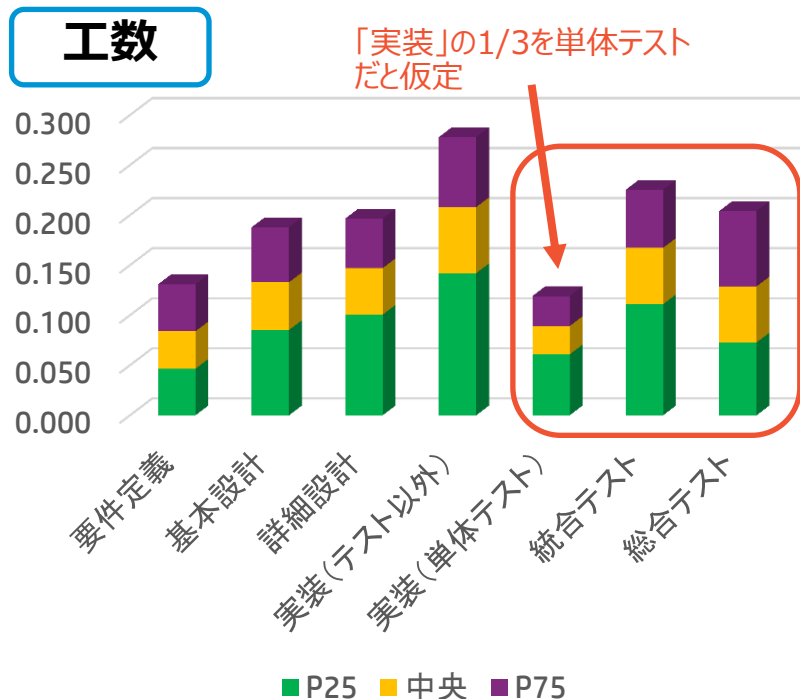
B: 38.5%

C: 40.6%

D: 54.9%

※IPA ソフトウェア開発データ白書2014 - 2015をベースに
改良開発の工数の中央値を要件定義も含めた工程比率で再計算した値

テストは開発工数/開発期間のどの程度を占めるか？



※IPA ソフトウェア開発データ白書2014 - 2015をベースに改良開発データの中央値を要件定義も含めた比率で再計算した値

上記比率を使って見積もってみると

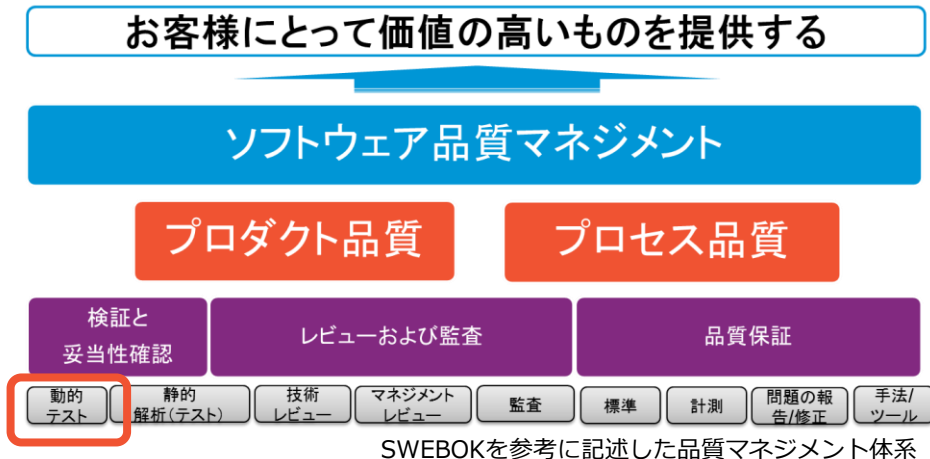
★要件定義を1ヶ月間 3人で行ったとすると

テストは、「 」ヶ月間「 」人の方が行うと見積もれる

★要件定義3ヶ月間 10人で行ったとすると

テストは、「 」ヶ月間「 」人の方が行うと見積もれる

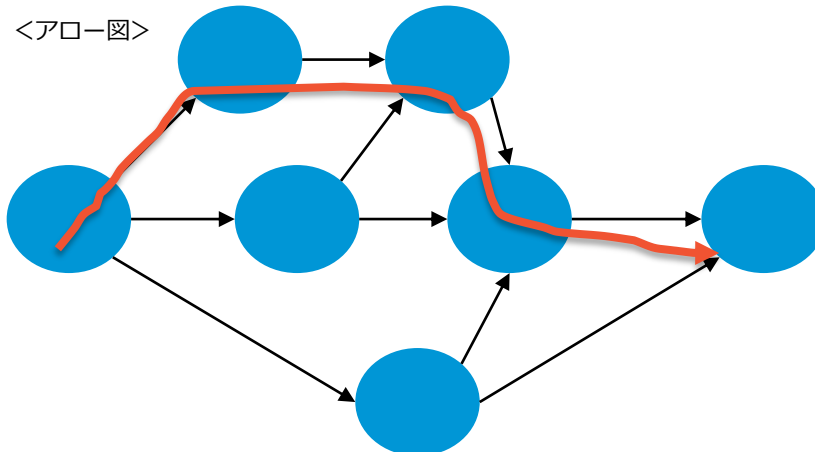
テストには2つの大きな役割



品質の良いものを世の中に 送り出すための大事な手段

・品質計測のインプット情報
として鮮度がよく綺麗な情報
を渡すこと

どれだけ正しく写す鏡か？



開発ライフサイクルの主要 な活動

・技術業として、決められた
QCDを達成できるように仕事
をすること

どれだけ効率的にできるか？

上手いテストをするとは

テストの活動をクリティカルパスにしないこと

テストは品質マネジメントの一手段

- 他の品質マネジメント手段との住み分けをしてスコープを明らかにする
- 関係する人たちへの「迅速に」「正しい」情報提供

テスト技術次第

- テスト対象の確認をより少ない量で確実に行う技術（設計、マネジメント）
- 作業スピードを上げる技術
- テストでのアウトプットを再利用する技術

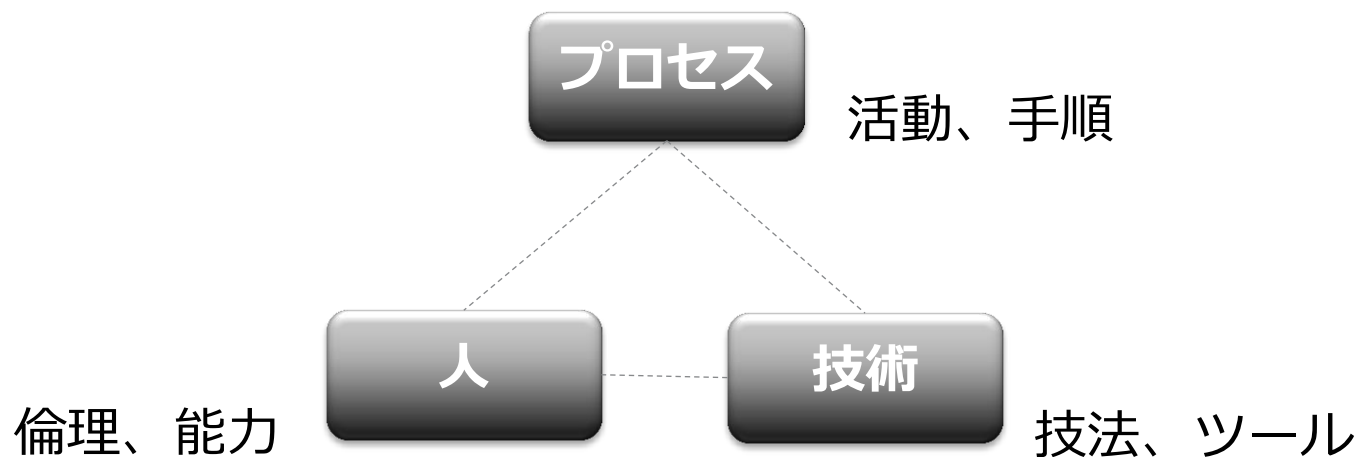
開発ライフサイクル次第

- テストの成熟度を高めようとするほど、開発の状況が前提条件となる

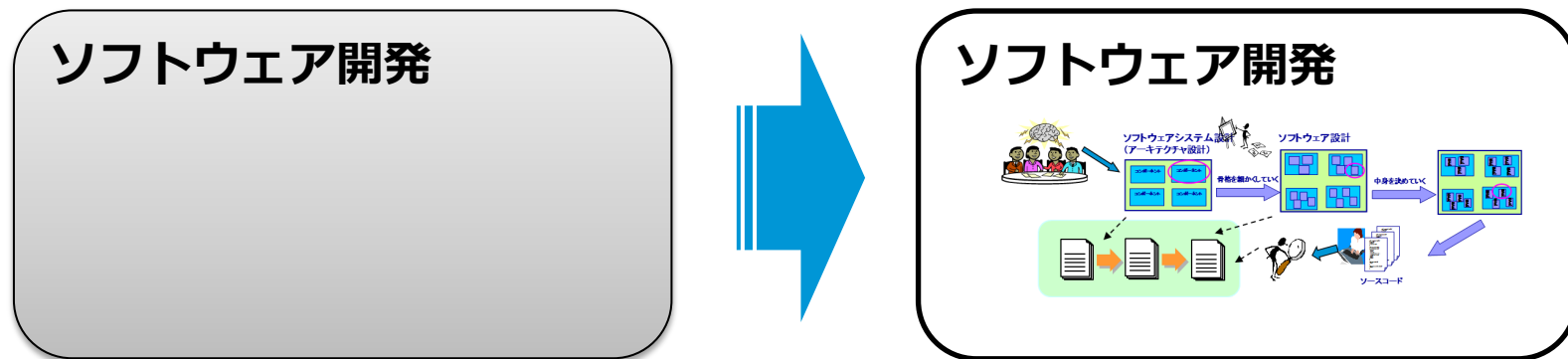
テストプロセス

プロセスとは？

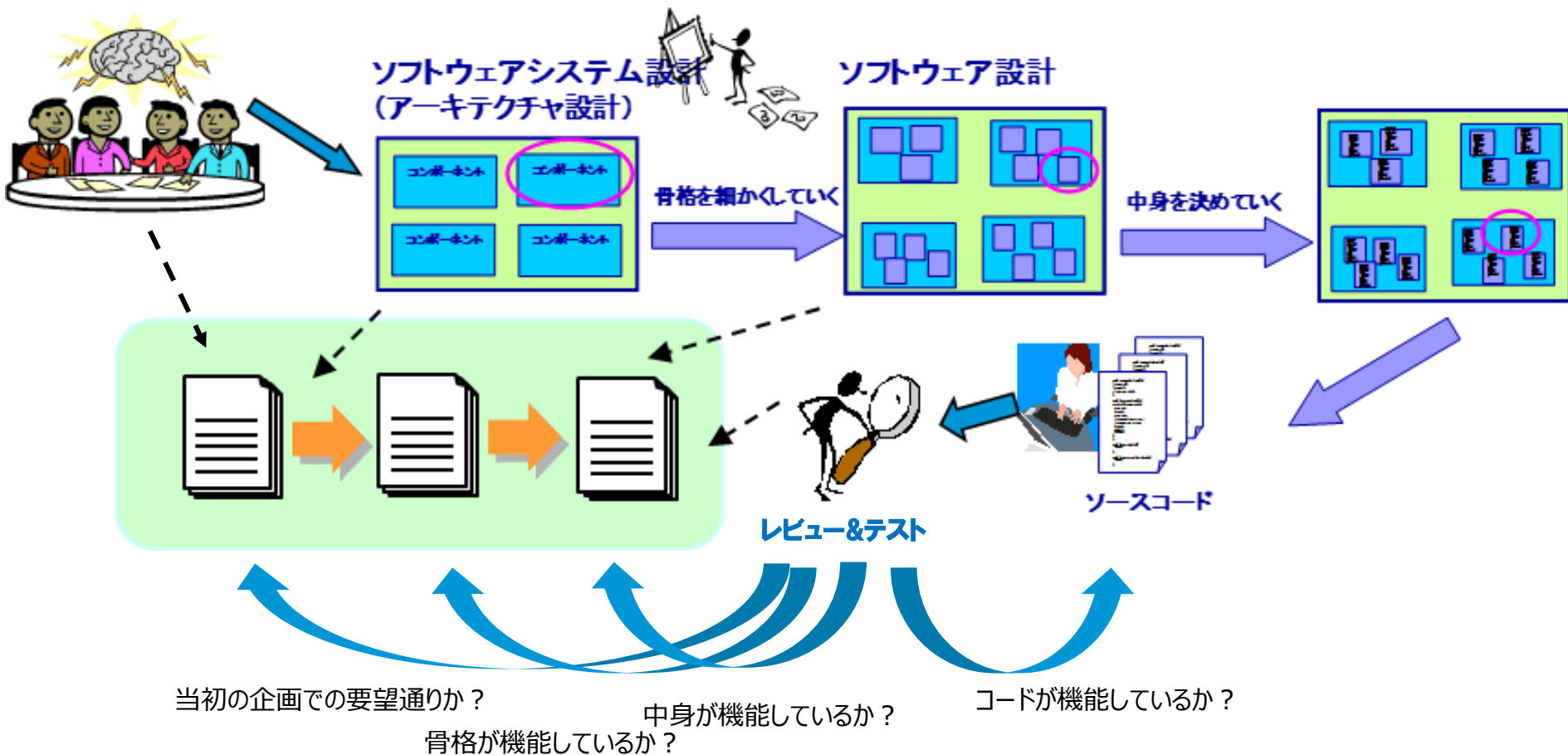
プロセスはエンジニアリングの一要素



開発での活動を具体的にあきらかにして共有可能にするのがプロセス



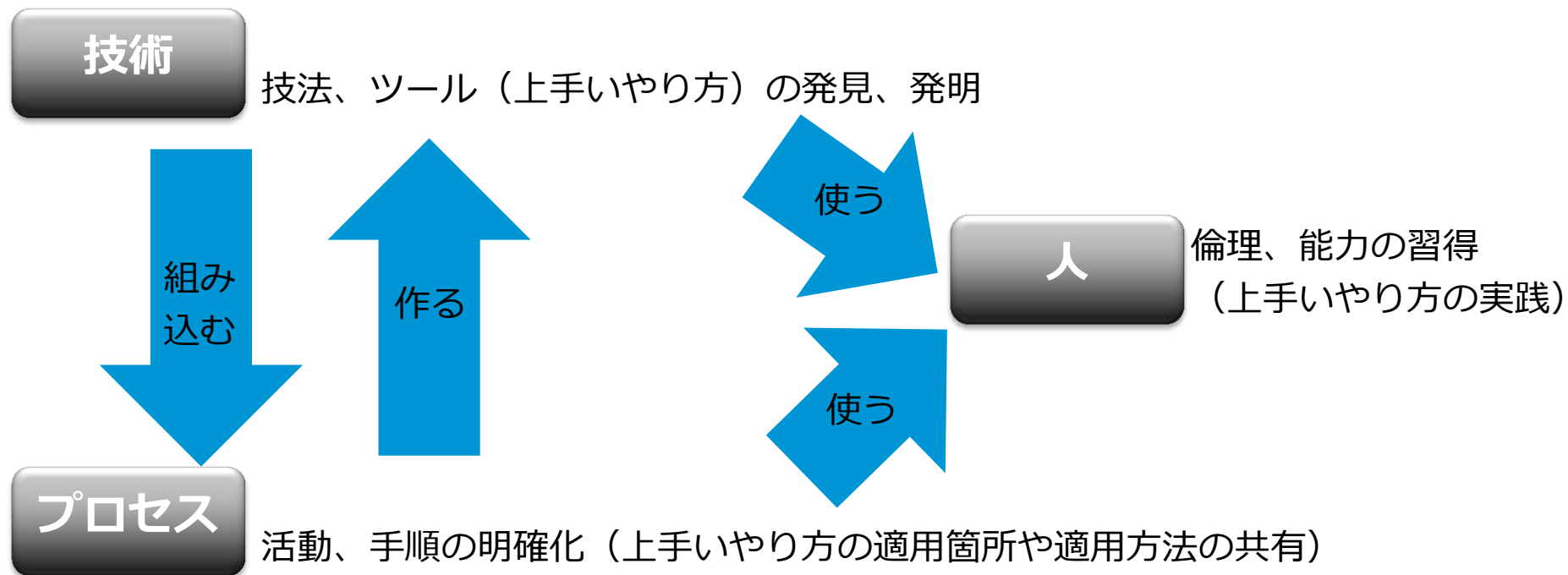
プロセスとは？



プロセスの節目単位で大丈夫か確認

プロセスとは？

エンジニアリングの三要素の関係



三要素があって初めて現場で実践できる

テストプロセス

- ・ テスト実行だけでは、「何故その行為をするのか？」が不明のまま
- ・ それに、仕組みや仕掛けを入れ込んでおかないと、実行するのもままならない

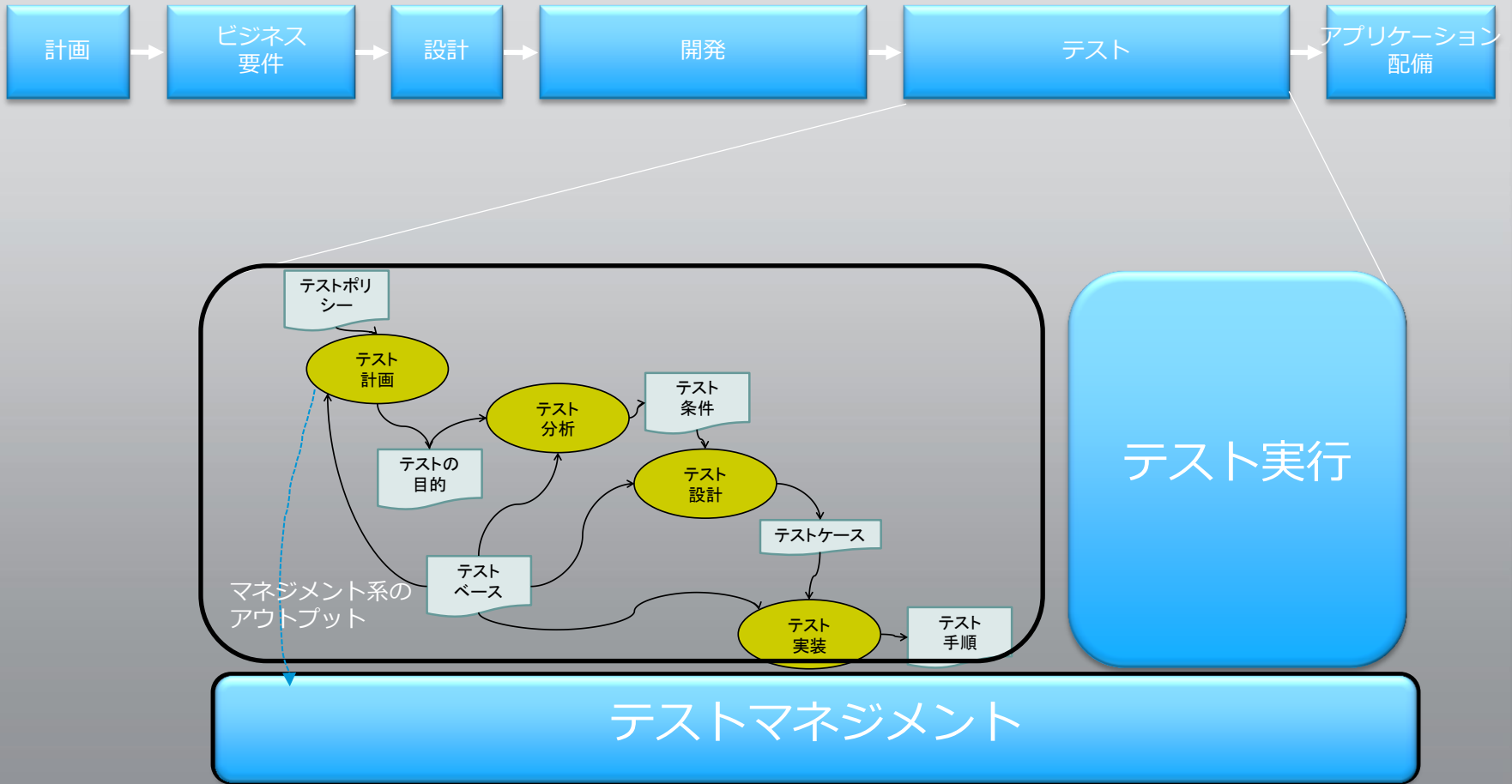
開発ライフサイクル



テストプロセス

- ・ **テスト実行だけでは、「何故その行為をするのか？」が不明のまま**
- ・ それに、仕組みや仕掛けを入れ込んでおかないと、実行するのもままならない

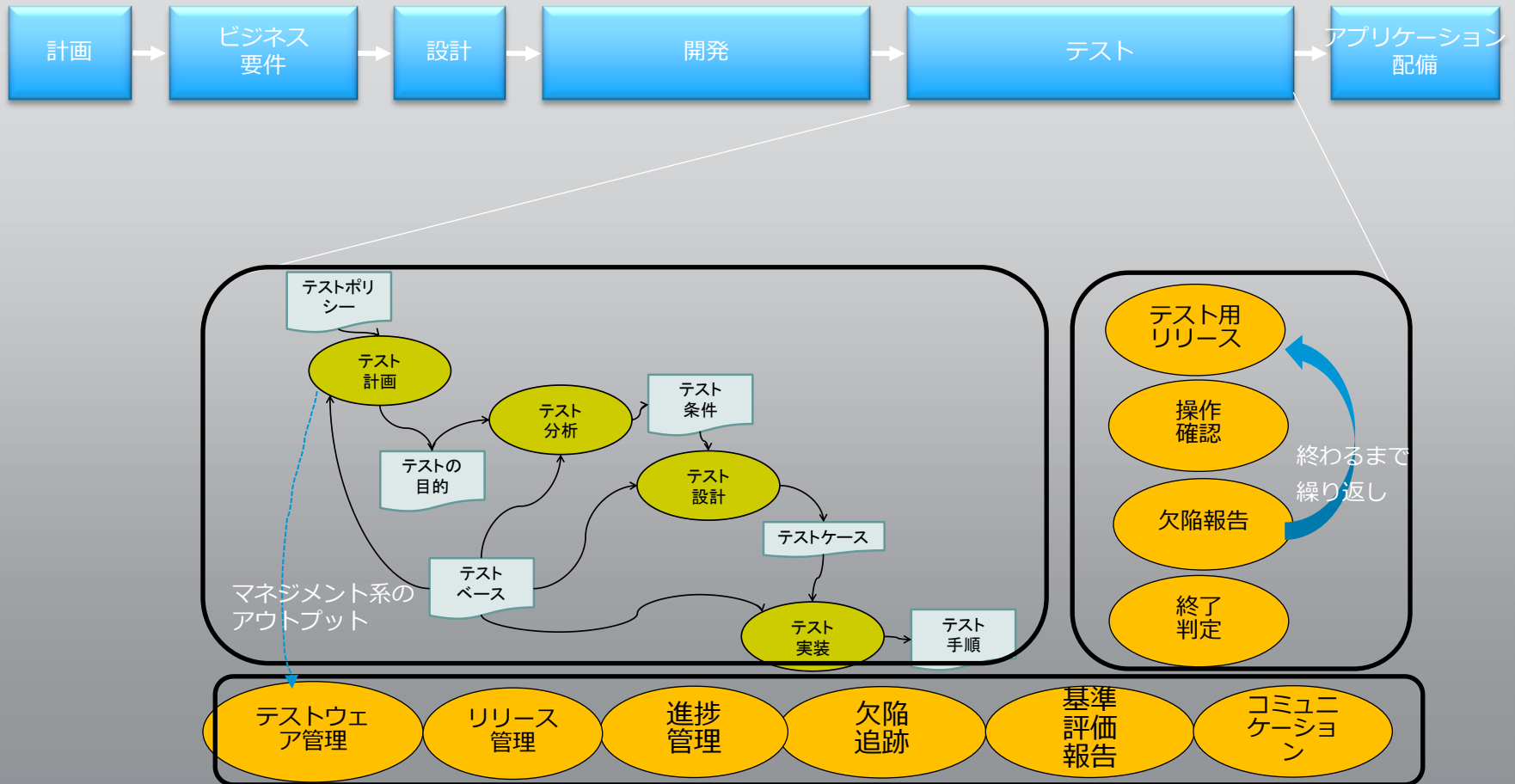
開発ライフサイクル



テストプロセス

- ・ テスト実行だけでは、「何故その行為をするのか？」が不明のまま
- ・ それに、仕組みや仕掛けを入れ込んでおかないと、実行するのもままならない

開発ライフサイクル



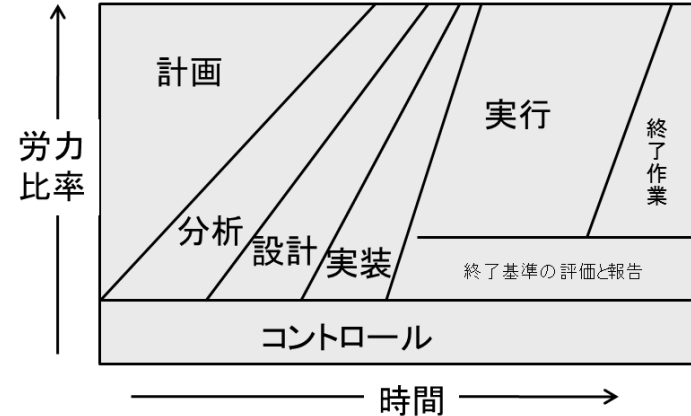
ISTQBが定義するテストプロセス

<http://jstqb.jp/>

ISTQBでは、テストプロセスは、以下の活動に分けて考えるとしている

- 計画
- コントロール
- 分析
- 設計
- 実装
- 実行
- テスト終了基準の評価と報告
- テスト終了作業

テスト開発プロセス



これらの活動はひとつずつ順番に行われるとは限らず、重なったり、並行したり、繰り返されたりする

テスト開発プロセスの三つの活動

(分析する) テストを作れるようテスト対象を理解する

(設計する) 目的に沿ったテストを作る

(実装する) テストを実行できるよう用意する

(手動なら) 手順を考える (自動なら) テストコードを書く
環境構築、テストデータ作成

テスト開発プロセス

本来のテストケースは以下の構造となり、段階的に成果物を作っていく
(つまり、プロセスがある)

テスト分析

テスト対象フィーチャ

フライト予約

テスト条件（仕様項目と期待結果）

購入金額の計算（クラス×人数に消費税をかけた金額になること）

テスト設計
方針

テスト詳細
設計

テストケース

ここで一般的なテスト
設計技法を活用する

テストパラメータ	値	アクション	期待結果
- クラス	エコノミー	1. フライト予約画面起動	・ 15万×3×1.08=48.6万
- 人数	3人	2. 必須項目を入力	・ 金額欄に上記を表示
- フライト	BA 成田-ロンドン 往復		
- 日程	141112/900-141120/721		
- チケット残数	25		

テストプロセスの改善

現場で上手いテストをしていくための方法

テスト設計技法の適用箇所

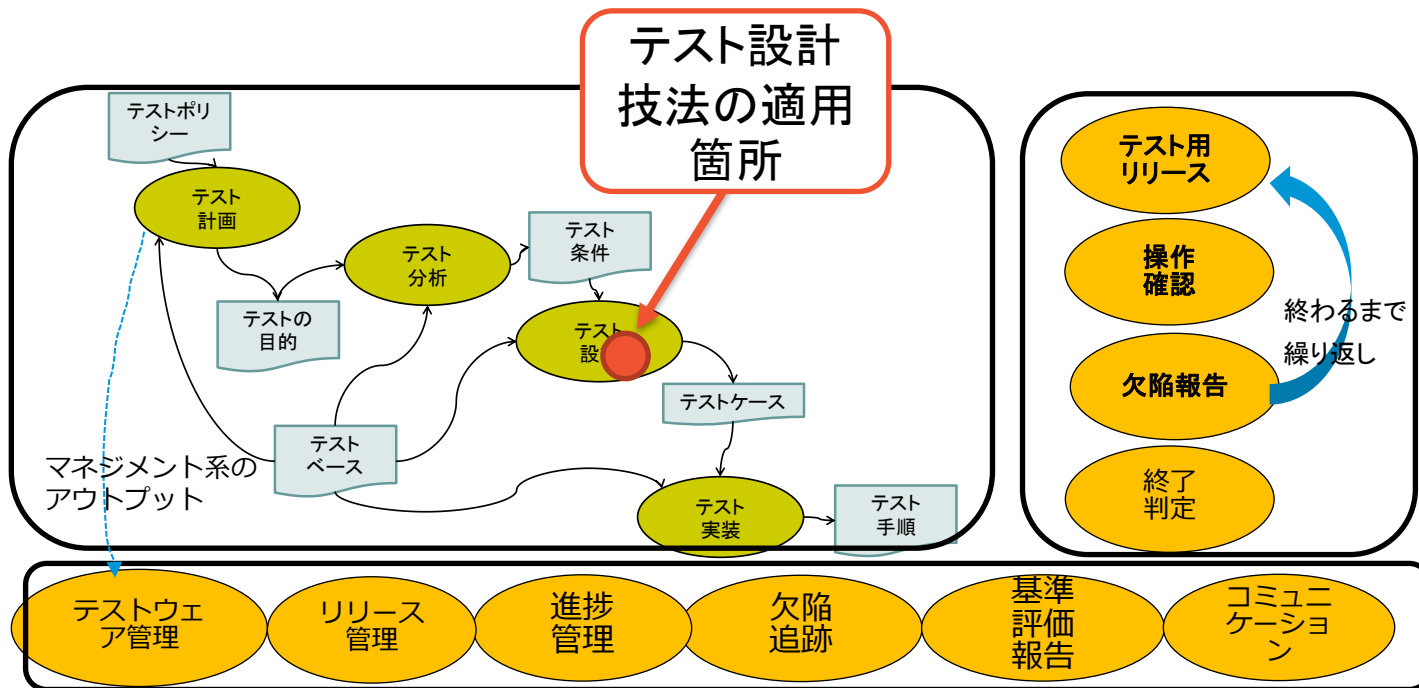
テストプロセスの改善

テストプロセスとして全体を考慮した改善の必要性

テストプロセスは開発プロセスの多くの部分を占めている

テストを良くするためには技法、手法だけで対応できない

- 多くの依存関係、想定外の対応を余儀なくされる



テスト設計技法に課題があるときの影響（1）

テストプロセスの改善

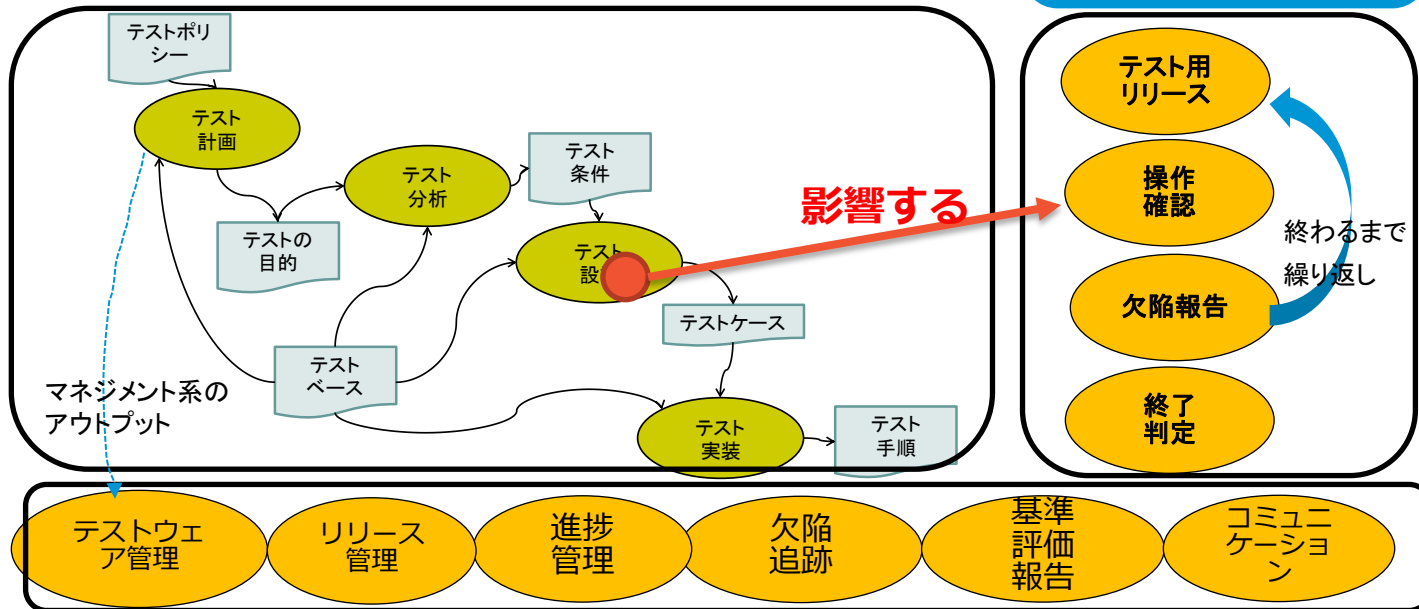
テストプロセスとして全体を考慮した改善の必要性

テストプロセスは開発プロセスの多くの部分を占めている

テストを良くするためには技法、手法だけで対応できない

- 多くの依存関係、想定外の対応を余儀なくされる

テストが無駄に多いと…



テスト設計技法に課題があるときの影響 (2)

テストプロセスの改善

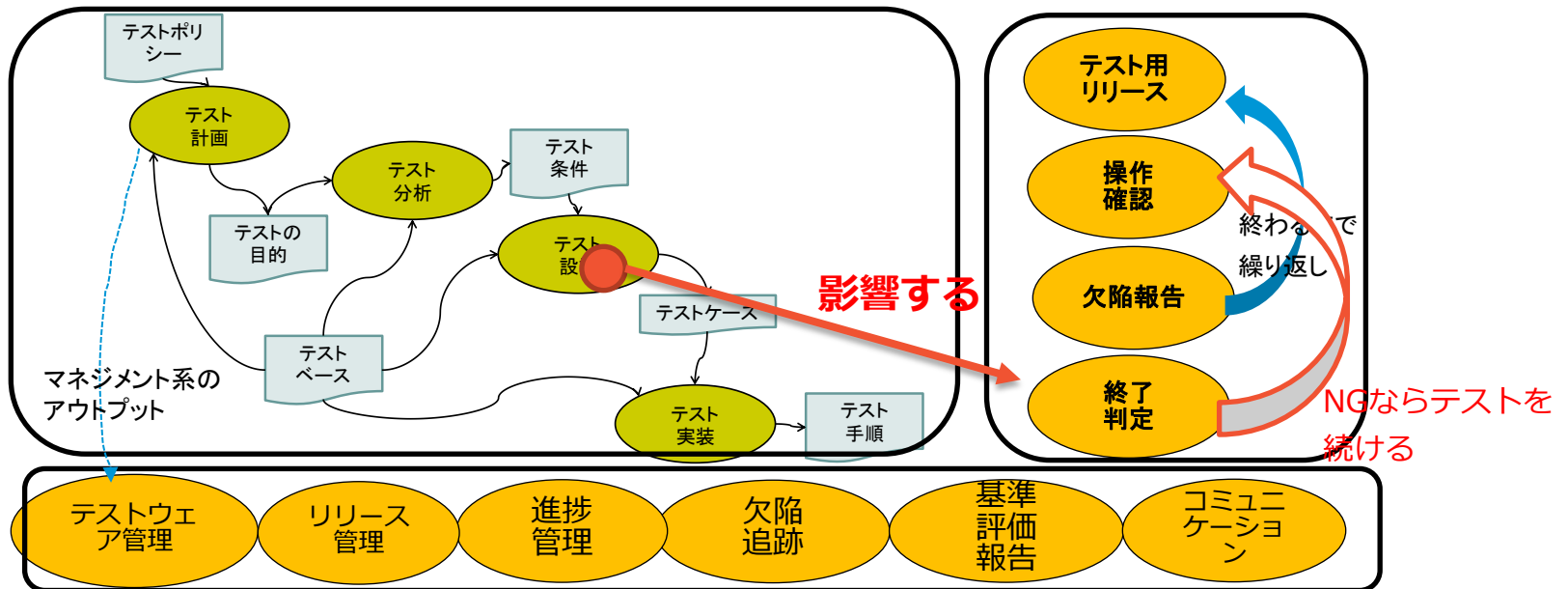
テストプロセスとして全体を考慮した改善の必要性

テストプロセスは開発プロセスの多くの部分を占めている

テストを良くするためには技法、手法だけで対応できない

- 多くの依存関係、想定外の対応を余儀なくされる

テストがあきらかに足りないとき…



テスト設計技法に課題があるときの影響 (3)

テストプロセスの改善

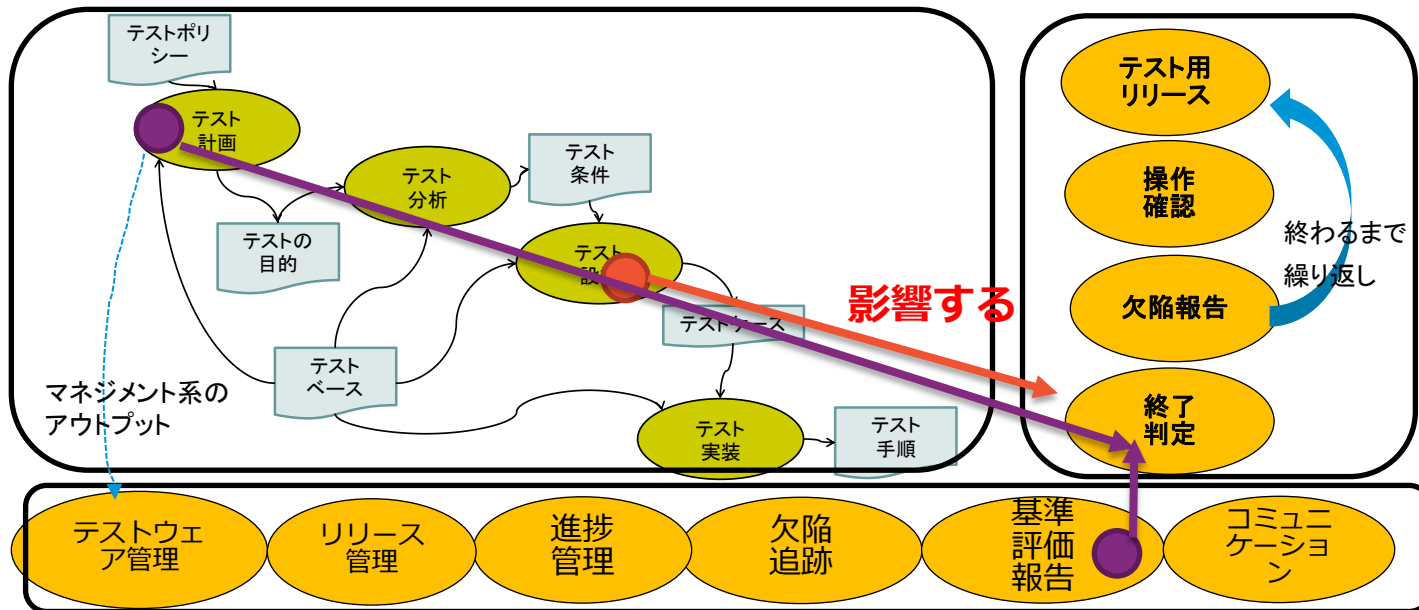
テストプロセスとして全体を考慮した改善の必要性

テストプロセスは開発プロセスの多くの部分を占めている

テストを良くするためには技法、手法だけで対応できない

- 多くの依存関係、想定外の対応を余儀なくされる

更に計画時に判断基準を誤ると…



市場トラブル

テスト設計技法を習得しても解決しない問題（1）

テストプロセスの改善

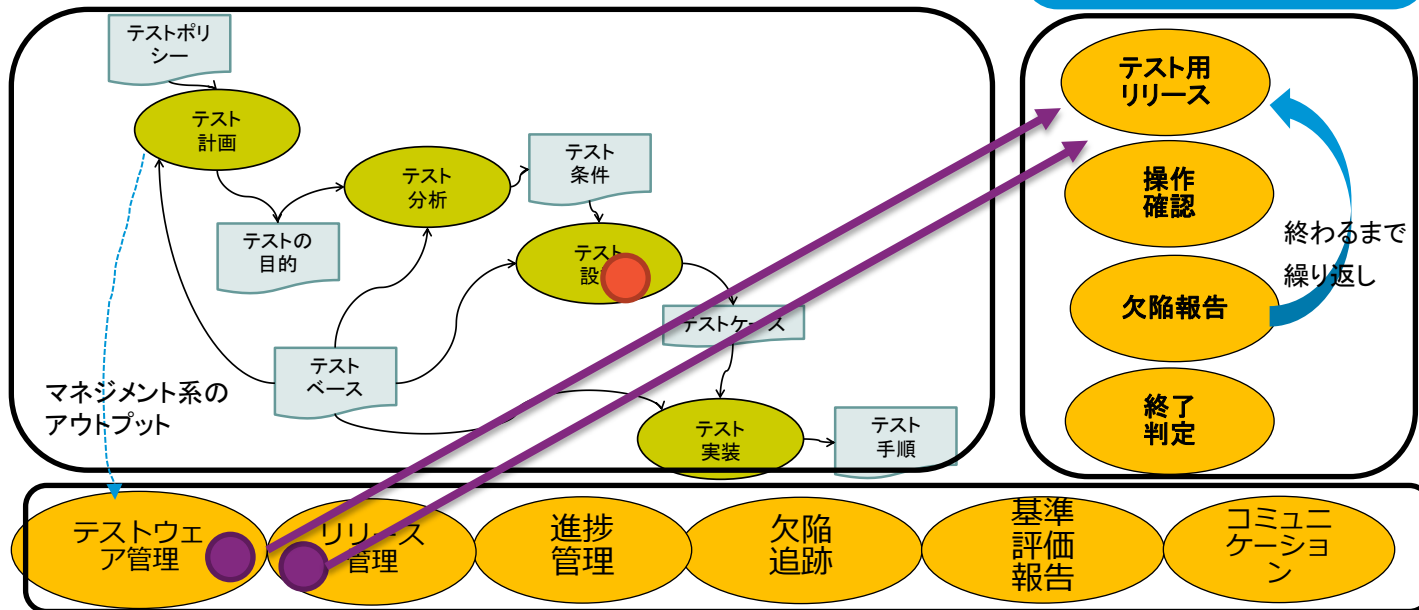
テストプロセスとして全体を考慮した改善の必要性

テストプロセスは開発プロセスの多くの部分を占めている

テストを良くするためには技法、手法だけで対応できない

- 多くの依存関係、想定外の対応を余儀なくされる

リリースにもた
ついてどんどん
遅れる



テストウェア、リリースマネジメントを怠ると

テスト設計技法を習得しても解決しない問題（2）

テストプロセスの改善

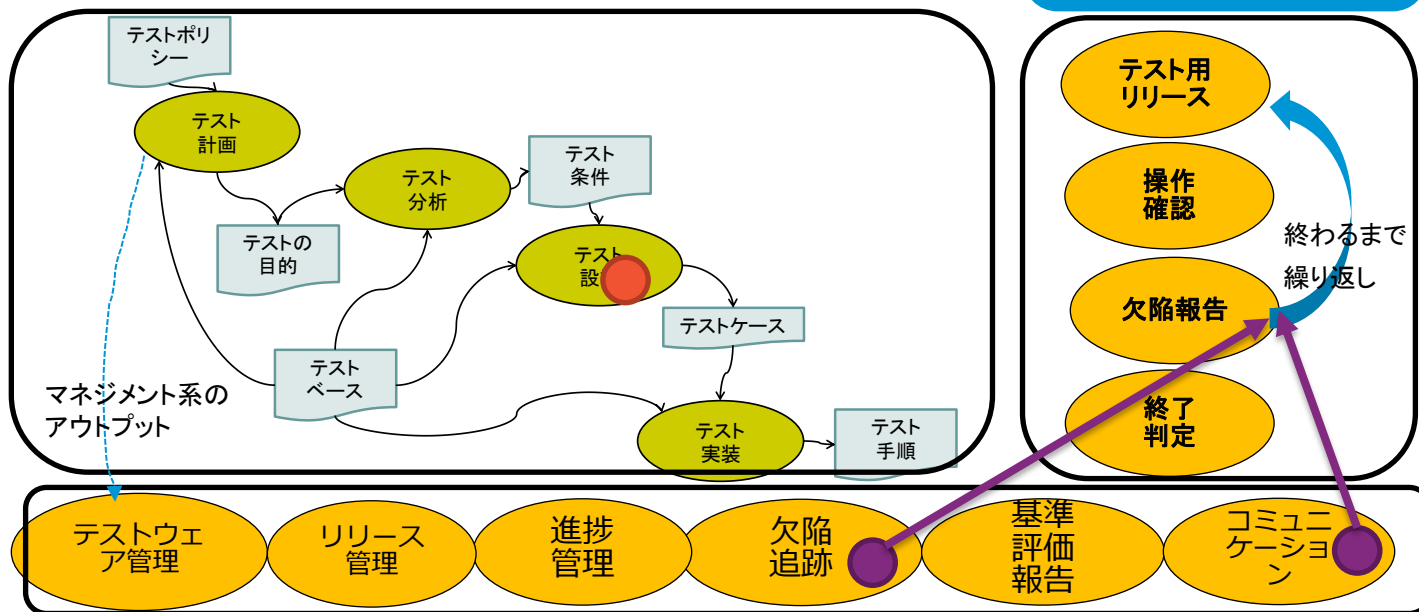
テストプロセスとして全体を考慮した改善の必要性

テストプロセスは開発プロセスの多くの部分を占めている

テストを良くするためには技法、手法だけで対応できない

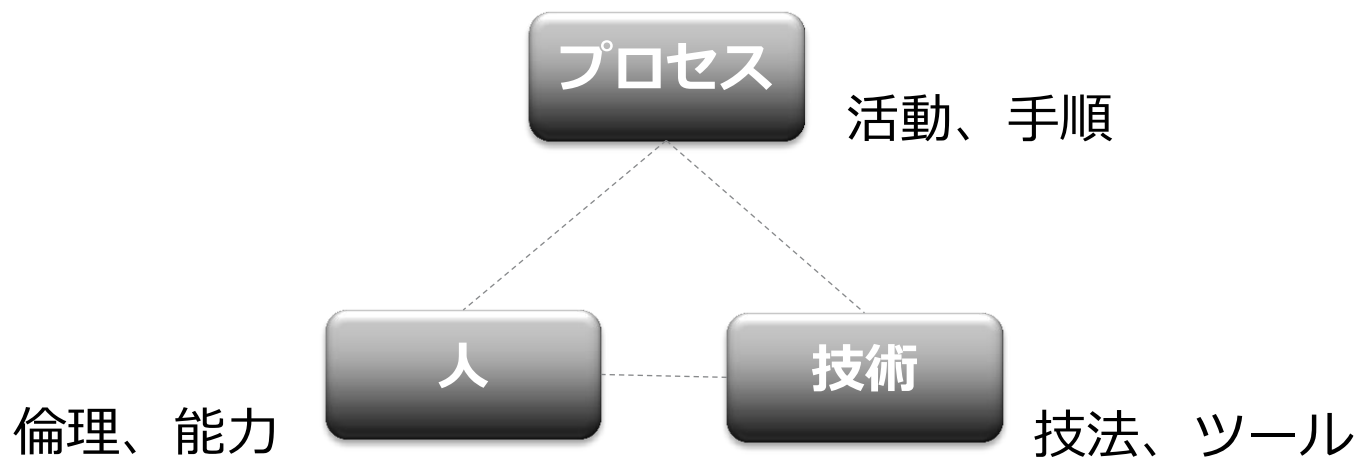
- 多くの依存関係、想定外の対応を余儀なくされる

欠陥が直らない
ので進まない



欠陥追跡を怠ると

テストプロセス全体を見渡して改善するのが必要



三要素のバランスを保つのがコツ

世の中に公開されているテストプロセス改善モデル

段階的モデル

TMMi (テスト成熟度モデル統合...CMMIを基にしたテストプロセス改善モデル)

ISO/IEC/IEEE33063 Process assessment model for software testing

連続的モデル

TPI NEXT (TPIの後継版として2009年に開発された)

CTP (クリティカルテストイングプロセス)

STEP(体系的テストと評価プロセス)

※段階的モデル、連続的モデルの分類はISTQBテスト技術者資格制度
Advanced Level シラバス 日本語版 テストマネージャ Version2012.J03より引用

**テストを改善するにはテストに特化した
参照モデル（改善の道具）利用が近道**

テストプロセスに特化した 改善のための道具

TPI NEXTのエッセンス

TPI NEXTとは

テストプロセス改善に特化した成熟度モデル

“自己評価”とそれに基づく“自己改善”

アセッサにより“認定”したり,“認定”されたりしない

評価軸(通常横軸)は単なるプロセスではない

独自に選定した16個のキーエリア

(人の要素、技術要素など)

ビジネス主導 [BD (ビジネス主導) のTPI]

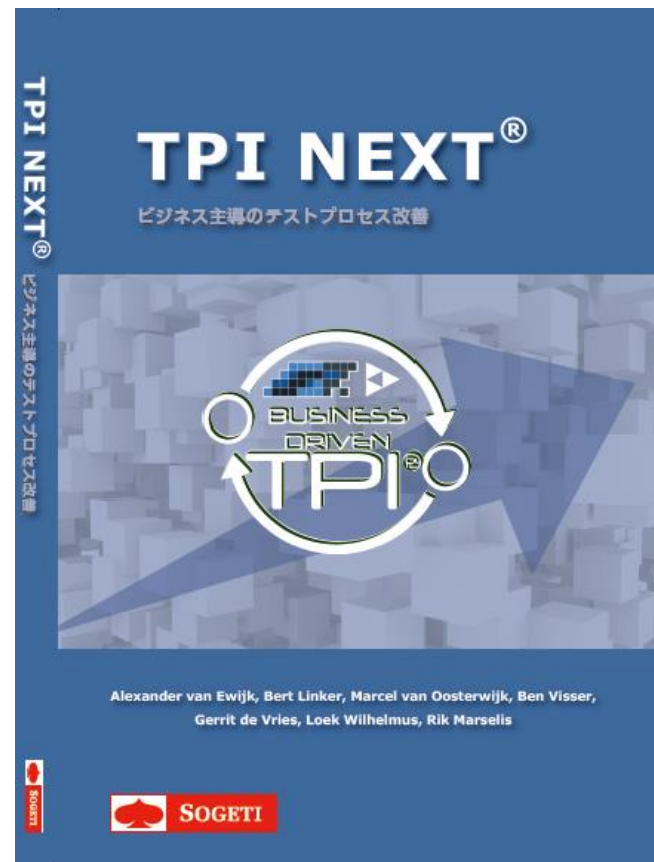
TPI NEXTでいう「ビジネス」とは

「テスト業務が実現すべきこと」を指している

- 適切な市場投入タイミング順守
- コスト削減
- テスト対象の有効性の向上、など

達成度の因果関係を示す「クラスタ」

- ビジネス要因を加味したクラスタのカスタマイズ



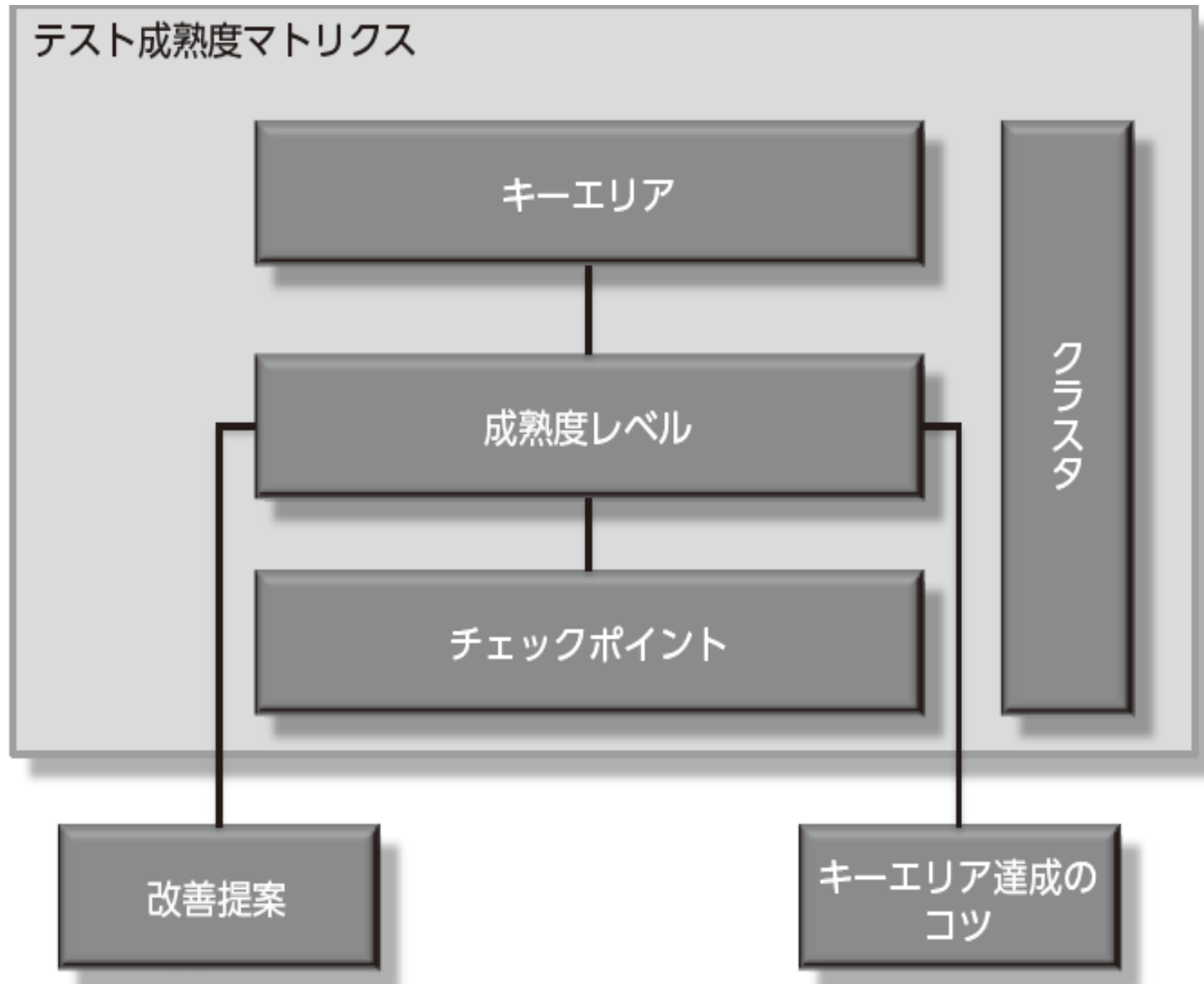
英語版

- 時期:2013年出版
- 出版社:UTN Publishers
- 著者:Alexander van Ewijk, 他6名

日本語版

- 時期:2015年出版
- 出版社:トリフオリオ
- 訳者: 薮田和夫、湯本剛、皆川義孝

TPI NEXTの基本構成要素



16のキーエリアと3つのグループ

3つのグループの特徴

利害関係者のコミットメント
関与の度合い
テスト戦略
テスト組織
コミュニケーション
報告

- ・開発プロジェクト全体から情報収集が必要
- ・改善対象がテストと関連する他組織にひろがる

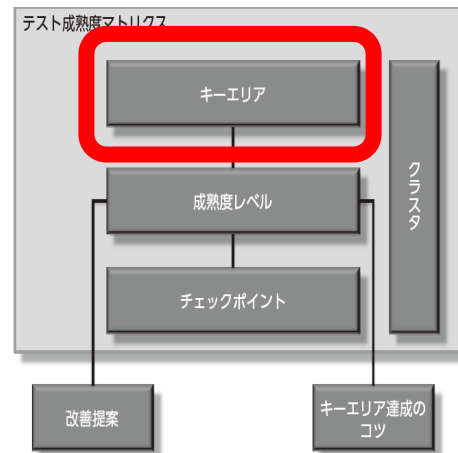
利害関係者との関係

テスト管理

- ・テストの管理をしている人たちからの情報収集が必要
- ・KA全般的に改善対象に関わる人が同じ

テスト業務の専門性

- ・テストを実際に行っている人たちからの情報収集が必要
- ・KA毎に改善対象に関わる人が異なる

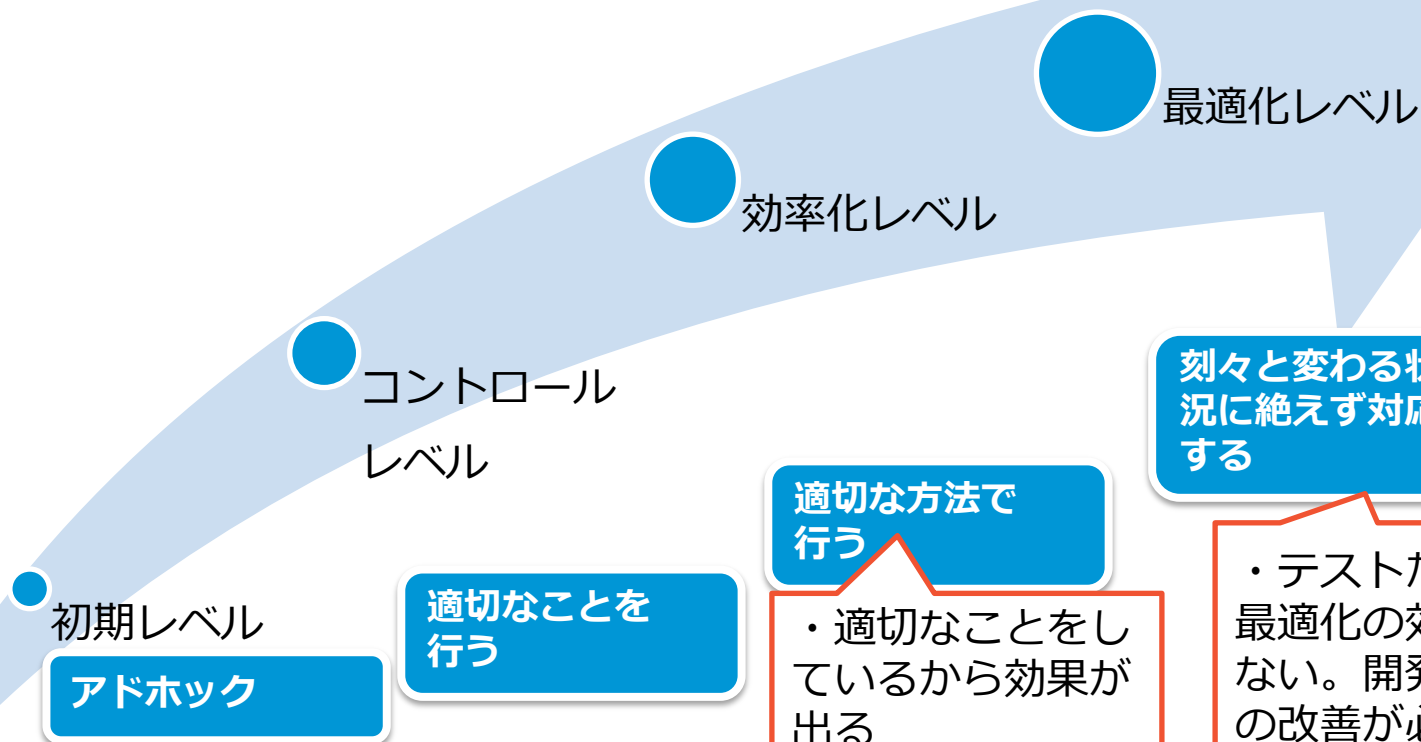
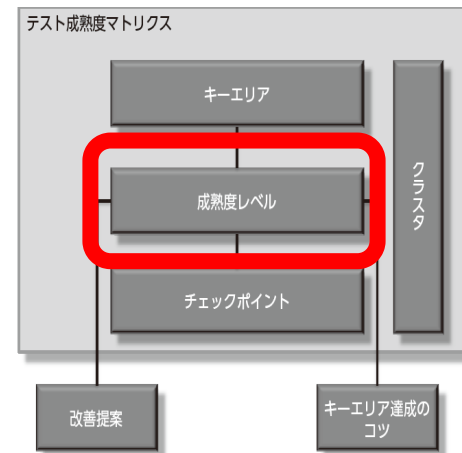


手法の実践
テスト担当者のプロ意識
テストケース設計
テストツール
テスト環境

4段階の成熟度レベル

KA[見積もりと計画]の 成熟度レベル定義

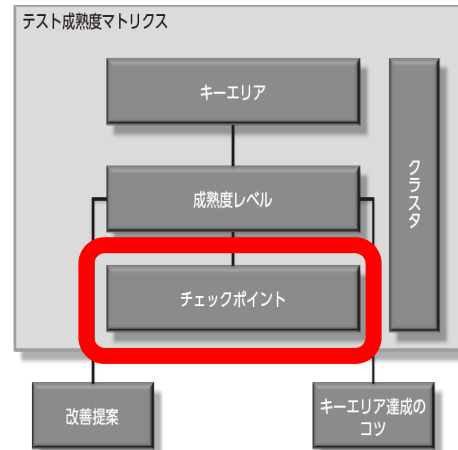
- 初期レベル：必要なリソースや計画された活動が大まかに見積もられてる
- コントロールレベル：各活動に対して必要なリソース量が予測できている
- 効率レベル：正式な技法により見積りや計画を信頼性のあるものになっている
- 最適レベル：見積りは組織の経験データに基づいて行われている



成熟度レベルとチェックポイント

KA[見積もりと計画]の チェックポイント

初期レベル（1）：テスト作業の見積もりには比率のような単純な技法が用いられている [YES/NO]



	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 テスト戦略	1	2	3	4	1	2	3			1		2
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
6 報告	1	2	3		1	2	3			1		2
7 テストプロセス管理	1	2	3	4	1	2	3			1		2
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		1	2	3	4		1		2
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		1	2	3	4		1		2
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		1	2	3	4		1	2	3
15 テストツール	1	2	3		1	2	3	4		1	2	3
16 テスト環境	1	2	3	4	1	2	3	4		1	2	3

※ マトリックスの1~4がチェックポイント

成熟度レベルとチェックポイント

チェックリストに全部答えると何が分かるか？

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

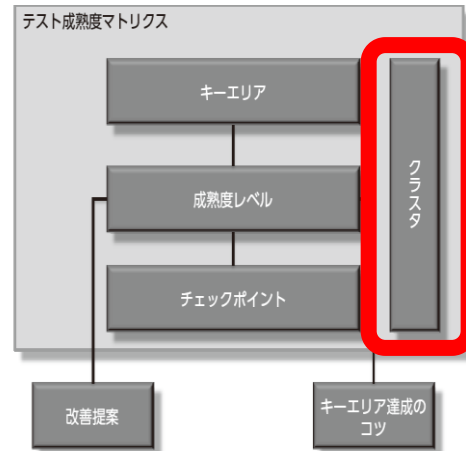
1.利害関係者のコミットメント	1234				123				123		
2.関与の度合い	1234				123				12		
3.テスト戦略	1234				123				12		
4.テスト組織	1234				1234				123		
5.コミュニケーション	1234				123				12		
6.報告	12		3		123				12		
7.テストプロセス管理	12		34		123				12		
8.見積もりと計画	12		34		1234				123		
9.メトリックス	12		3		1234				12		
10.欠陥管理	12		34		1234				123		
11.テストウェア管理	12		34		123				123		
12.手法の実践	12		3		1234				12		
13.テスト担当者のプロ意識	12		34		1234				123		
14.テストケース設計	12		3		1234				123		
15.テストツール	12		3		1234				123		
16.テスト環境	12		34		1234				123		

※マトリックス中の1～4はチェックポイント

基本クラスタセット

何から手をつけるか？

基本クラスタセットは「典型的」な改善の実施順



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

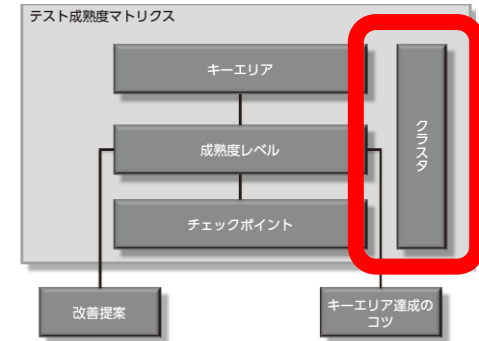
- 1.利害関係者のコミットメント
- 2.関与の度合い
- 3.テスト戦略
- 4.テスト組織
- 5.コミュニケーション
- 6.報告
- 7.テストプロセス管理
- 8.見積もりと計画
- 9.メトリックス
- 10.欠陥管理
- 11.テストウェア管理
- 12.手法の実践
- 13.テスト担当者のプロ意識
- 14.テストケース設計
- 15.テストツール
- 16.テスト環境

A	B	B	C	F	H	H	K	M	M
A	B	C	E	H	H	J	L	L	
A	A	B	E	F	F	H	K	L	
A	D	D	E	I	I	J	K	L	L
B	C	C	D	F	F	J	M	M	
A	C	C		F	G	G	K	K	
A	A	B	B	G	H	J	K	M	
B	B	C	C	G	H	I	K	L	L
C	C	D		G	H	I	K	K	
A	A	B	D	F	F	H	J	K	L
B	B	D	E	I	I	J	L	L	L
C	D	E		F	H	J	M	M	
D	D	E	E	G	G	I	K	K	M
A	A	E		F	I	I	K	K	M
E	E	E		F	G	G	L	M	M
C	D	D	E	G	H	J	L	M	M

※マトリクス中のA～Mはクラスタ

クラスタセットA

クラスタAには12のチェックポイント



	チェックポイント			
利害関係者のコミットメント	63%	25%		
関与の度合い	63%	8%		
テスト戦略	53%	9%	50%	26%
テスト組織	78%	74%		
報告	66%	66%		
テストプロセス管理	56%	56%	59%	68%
欠陥管理	78%	82%	75%	84%
テストケース設計	75%	72%	94%	68%

改善のロードマップを描く

TPI NEXTを活用した改善ステップ

TPI NEXT の改善ステップ

改善の進め方

1. 気付きを与える（意識付け）

- ・なぜテストプロセスを改善すべきかを理解

2. ゴール、スコープ、取り組み方を決定する

- ・ To-Be像と開始ポイントを決定

3. 現状をアセスメントする

- ・ As-IS把握（SWOT分析）

4. 改善を定義する

- ・ **ゴール達成に向けたロードマップ作成**

5. 行動計画を立案する

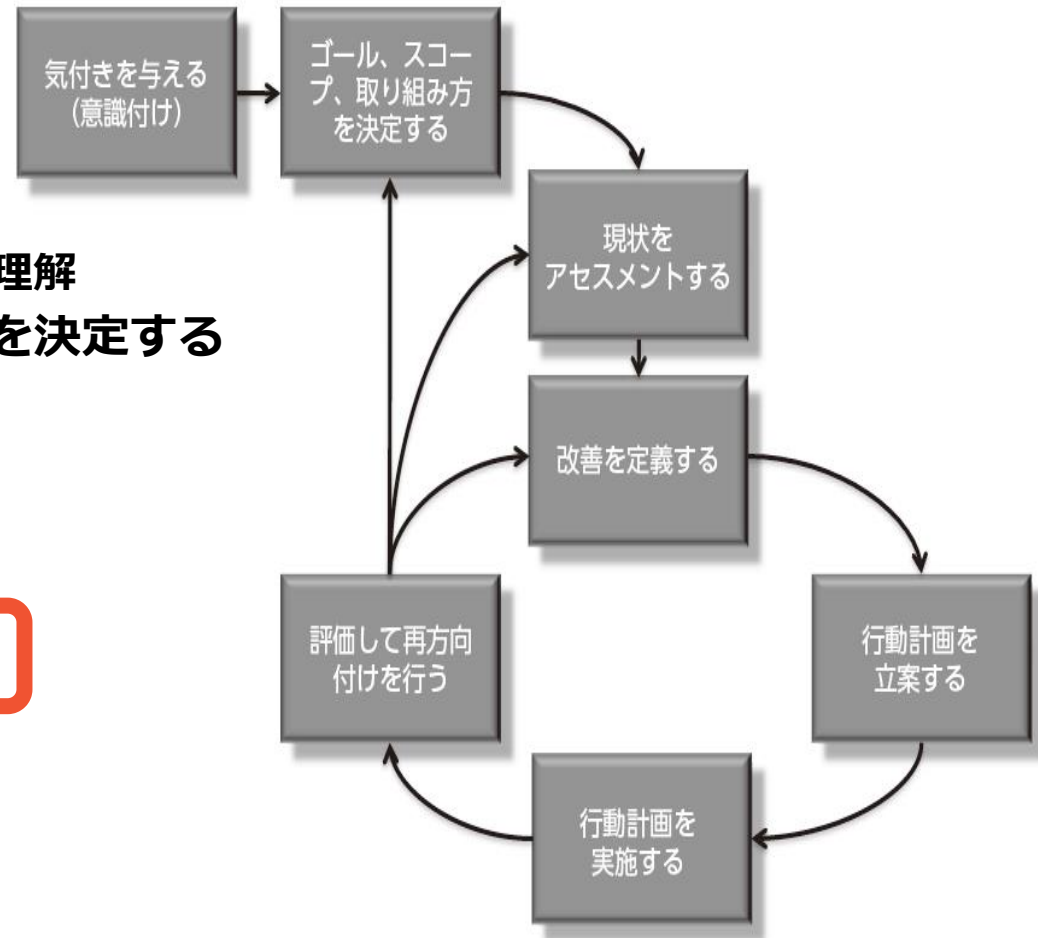
- ・ 人員のアサイン、タイムフレーム

6. 行動計画を実施する

- ・ 新しい手順やテンプレートの適用

7. 評価して再方向付けを行う

- ・ ゴール達成を評価し、評価結果から次のアクションを決定



TPI NEXT の改善ステップ

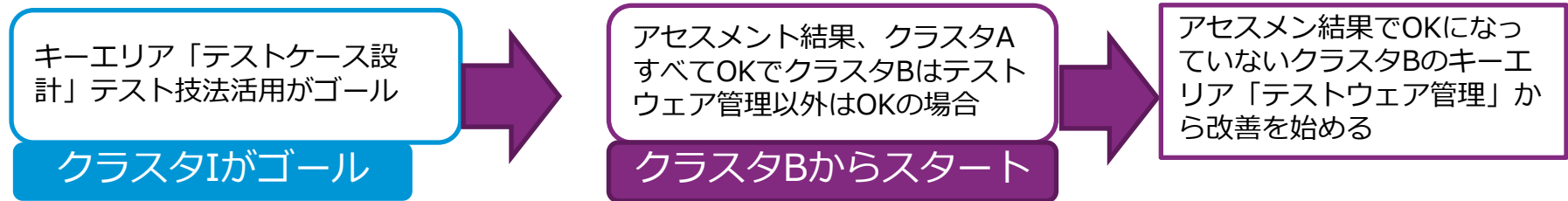
ゴール達成に向けたロードマップ

ロードマップの作成方法（基本編）

基本クラスタセット

- TPINextにて定義した標準的な改善のロードマップ
 - 筆者たちのこれまでの経験ベース

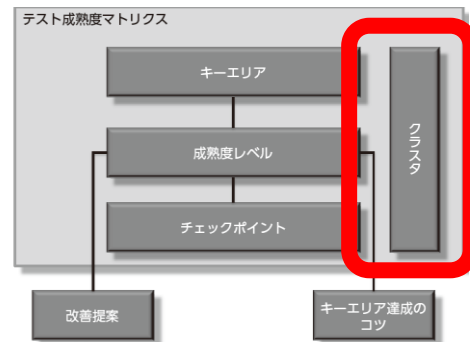
＜組織がテスト設計技法を使いこなしている状態に改善する例＞



コントロールレベル	効率化レベル	最適化レベル
A : テストケースを論理レベルで記述している	F : チームの他の人が見て理解出来て、保守できる	K : 次フェーズで発生した欠陥を分析し、テストケースの正確性、有効性の向上に勤めている
A : テストケースには「事前条件」「操作」「期待結果」がある	I : 達成できるテストベースのカバレッジレベルがわかる	K : テストケースの妥当性と保守性を評価している
E : テストベースのどこがテスト対象になっているかがわかる	I : テスト設計技法を用いている	M : 再利用できるようにテストケースの見直しをしている
	J : 動的テストができないものでもチェックリストを用いて確認している	

基本クラスタセット

組織がテスト設計技法を使いこなしている状態に改善



適切なことを行う

適切な方法で行う

	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	A	B	B	C	F	H	H	K	M	M					
2 関与の度合い	A	B	C	E	H	H	J	L	L						
3 テスト戦略	A	A	B	E	F	F	H	K	L						
4 テスト組織	A	D	D	E	I	I	J	K	L	L					
5 コミュニケーション	B	C	C	D	F	F	J	M	M						
6 報告	A	C	C		F	G	G	K	K						
7 テストプロセス管理	A	A	B	B	G	H	J	K	M						
8 見積もりと計画	B	B	C	C	G	H	I	K	L	L					
9 メトリクス	C	C			H	H	I	K	K						
10 欠陥管理	A	A	E		F	H	J	K	L	L					
11 テストウェア管理	B	B	D		I	J	J	L	L	L					
12 手法の実践	C	D			H	J	J	M	M						
13 テスト担当者のプロ意識	D	D	E	E	G	G	I	K	K	M					
14 テストケース設計	A	A	E		F	G	I	K	K	M					
15 テストツール	E	E	E		F	G	I	L	M	M					
16 テスト環境	C	D	D	E	G	H	J	L	M	M					

道程を自分たちで考えるのが改善の醍醐味

テストウェア管理が未達なため、クラスタBからスタートしてテストケース設計Eを目指す

クラスタIがゴール

自組織に合わせたロードマップの作成例

「適切な方法」で行うその前に、「適切なこと」を行う

テスト業務の専門性

テスト管理

利害関係者との関係

テストケース設計

その他4つのKA

5つのKA

6つのKA

I

- ・テスト設計技法を用いている
- ・達成できるテストベースのカバレッジレベルがわかる

H

G

F

- ・チームの他の人が見て理解出来て、保守できる

E

- ・テストベースのどこがテスト対象になっているかがわかる

D

C

B

A

- ・「事前条件」「操作」「期待結果」が書かれている
- ・テストケースの論理レベルで記述している

OK

＜手法の実践＞

開発スタイルに合わせてテスト手法を見直している

＜プロ意識＞

テストに特化したトレーニング・体系的なテストの現場経験がある

＜手法の実践＞

プロセスがテスト手法にそっている

＜テストウェア管理＞

だれもがすぐにテストウェアにアクセスできる

＜テストウェア管理＞

テストケースにバージョンがついており、テストベースがどれかわかる。

＜コミュニケーション＞

関連する情報を、関係者から収集している。

＜コミットメント＞

コミットしたリソースを実際に手配している。

要は、推進するためにやること

- ・リソース（人、道具）確保
- ・道具の調達、作成（テンプレート、ツール）
- ・作業環境の整備
- ・他部署へ周知、調整（会議体）
- ・技術者教育（勉強会、セミナー、自己学習）

テストをクリティカルパスにしないための三大要素

「適切なこと」を自分たちの状況に合わせて行うのが「上手いやり方」

ちゃんと仕様理解する

- 問題が起きたら必ず仕様書を読み返す。
- テストケースに該当する仕様書がどこにあるかわかるようにしておく。

細かく可視化する

- テストケースを細かく用意して（1期待結果で1ケース）、その消化状況、一日の消化目標数、不具合とのリンクを常に全員が見れるようにする。
- 上記の「共通認識」になる情報を日次確認できる時間を作る。（5分でよい）

迅速に不具合を直す（よう働きかける）

- スタックしている不具合を、関係者達の目の前で転がして動けるようにする。
- 修正後は、できるだけ早く再テストを行いレポートをクローズする。

これらは愚直にやらないといけない。
技法、ツールはこれらを支援しないといけない。

まとめ

開発におけるテスト役割

- テストは開発の2割~5割をしめるが、やり方次第
- テストを開発のクリティカルパスにしないのが上手いテスト（テストが品質に貢献することは大前提！）

テストプロセス

- 活動を具体的にあきらかにして共有可能にするのがプロセス
- テストもプロセスなので、実行前に目的（とその詳細化）をして、仕掛けを入れ込む

テストプロセスの改善

- テストプロセス全体で何を改善すべきか考える

テストプロセスに特化した改善の道具

- TPINEXTを例にエッセンスを紹介

改善のロードマップを描く

- 「適切なこと」を行ってこそ「適切な方法」の効果が出る

参考文献

**IPA/SEC「ソフトウェア開発データ白書2014-2015」
(IPA/SEC)2014**

**IEEE Computer Society「SWEBOOKソフトウェアエンジニアリング知識体系」
(オーム社)2005**

**湯本剛「ソフトウェアテストPressVol.10 特集1 今こそ聞きたいテストの上流設計」
(技術評論社) 2010**

**豆蔵「ソフトウェアエンジニアリング講座 エンジニアリング入門」
(豆蔵)**

**ISTQB「ISTQBテスト技術者資格制度Foundation Levelシラバス日本語版Version2011.J01」
(NPO法人ASTER,JSTQB)2011**

**ISTQB「ISTQBテスト技術者資格制度Advanced Level シラバス日本語版TM Version2012.J03」
(NPO法人ASTER,JSTQB)2013**

**Alexander van Ewijk 他 (Sogeti)「TPINEXT ビジネス主導のテストプロセス改善」
(トリフォリオ) 2015**

ご清聴

ありがとうございました