



ほかにはない
アンサーを。

ローコードアプリケーション基盤上での開発における 品質保証の取り組み

オリックス生命保険株式会社

IT品質保証部

藤井 和弘

kazuhiro.fujii.dd@ins.orix.jp

本発表における発表者の発言内容は、必ずしも当社の公式発表・見解をあらわすものではありません。



オリックス生命保険株式会社
IT品質保証部

2016年10月 オリックス生命に中途入社

2019年4月～現在 ソフトウェア品質保証担当

✓本日発表する内容

- ✓ オリックス生命において取り組んでいるローコードアプリケーション基盤上での開発上での品質保証の取り組みについて、背景や工夫した点、今後の展開などをご紹介します。
- ✓ その中でも特に考慮した内容を重点的に発表します。

✓特に聴いてほしい方

- ✓ ローコードアプリケーション基盤上での開発上での品質保証に取り組んでいる方

同様の取り組みをされている皆様との情報交換、意見交換に繋がればと考えています。

1. 取り組みの背景
2. 取り組みの目的
3. 取り組みの内容
4. 品質保証にあたり考えたこと
5. 今後の展開

1. 取り組みの背景

当社では、次の理由からローコードアプリケーション開発基盤（以下LCAP開発基盤）の採用増加を見込んでおり、利用方針や開発プロセスを整理する必要があった。

開発の規模増大とリリースサイクルの高頻度化

既存商品の改定や、新商品開発サイクルの高速化

AIなど新テクノロジーへの対応などによる開発規模の増加

IT人材の不足

長期的な人口減少

先端IT人材の不足などによる人手不足

2. 取り組みの目的

高速開発の利点を生かす

必要なプロセスと成果物を過不足なく定義することで、開発プロセスを効率化する

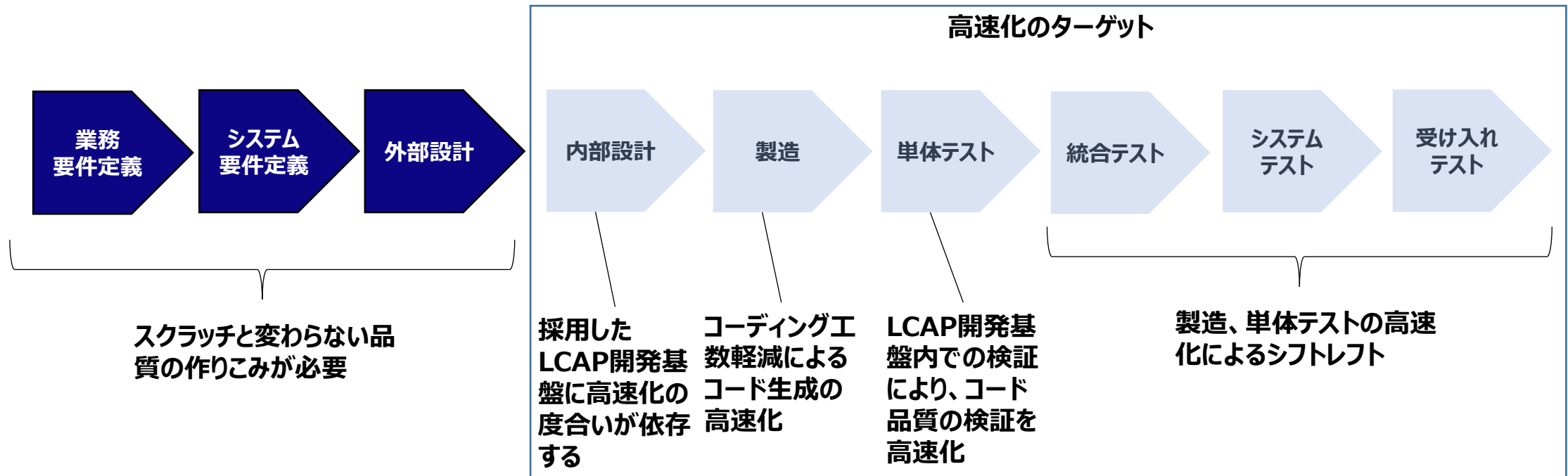
ローコードアプリケーション開発 基盤上での開発においても 品質を保証する

レビューやテストおよび品質メトリクスの違いを明らかにし、スクラッチ開発と変わらずに品質保証できるようにする

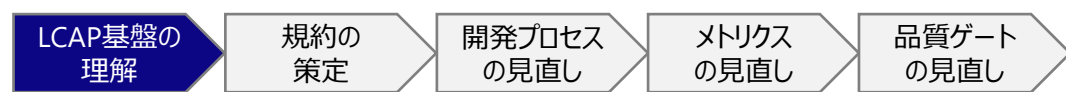
2. 取り組みの目的

■ 高速化の前提

LCAP開発基盤を採用することで、高速化する工程は製造工程以降をターゲットにしており、外部設計までの上流工程はスクラッチ開発と変わらずに十分な作りこみを必要としている。



3. 取り組みの内容

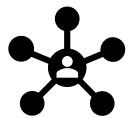


■ 初めにLCAP開発基盤を理解することに取り組んだ

プロダクトであるLCAP基盤の機能が、開発プロセスに影響を与えるプロダクト駆動の点があると考え、まずLCAP基盤の理解を進めた。



外部講習の受講

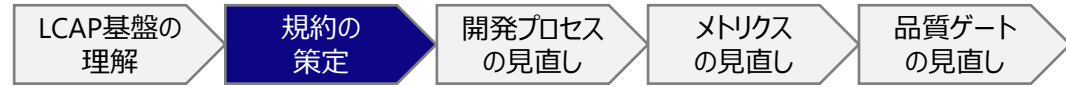


LCAP基盤に関する技術情報の共有



認定資格取得の奨励

3. 取り組みの内容



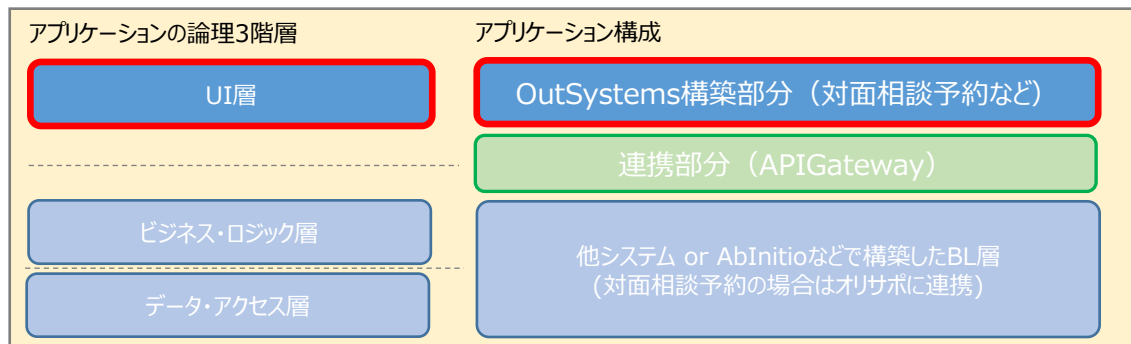
■ 次にLCAP開発基盤をどのように利用する検討した

開発規約に相当する設計者ガイドをLCAP開発基盤の利用者向けに策定した。

設計者ガイドは、開発プロジェクトからのフィードバックを受けて継続更新することとしている。

【OutSystems設計者ガイド_v1.02.アプリケーション構成より】

- ・カスタムコード・・・原則カスタムコードを書かない。
- ・アーキテクチャ・・・システムの肝であるデータ、APIは原則自社システムで実装し、LCAP開発基盤上ではUI、APIリクエスト部分のみ実装する。
- ・キャパシティ管理・・・基盤の性能管理、キャパシティ管理はLCAP基盤側のマネジメント領域とする。
- ・そのほか、コンポーネントの命名規約やライブラリ管理としての共通部品の利用方法を追記予定としている。

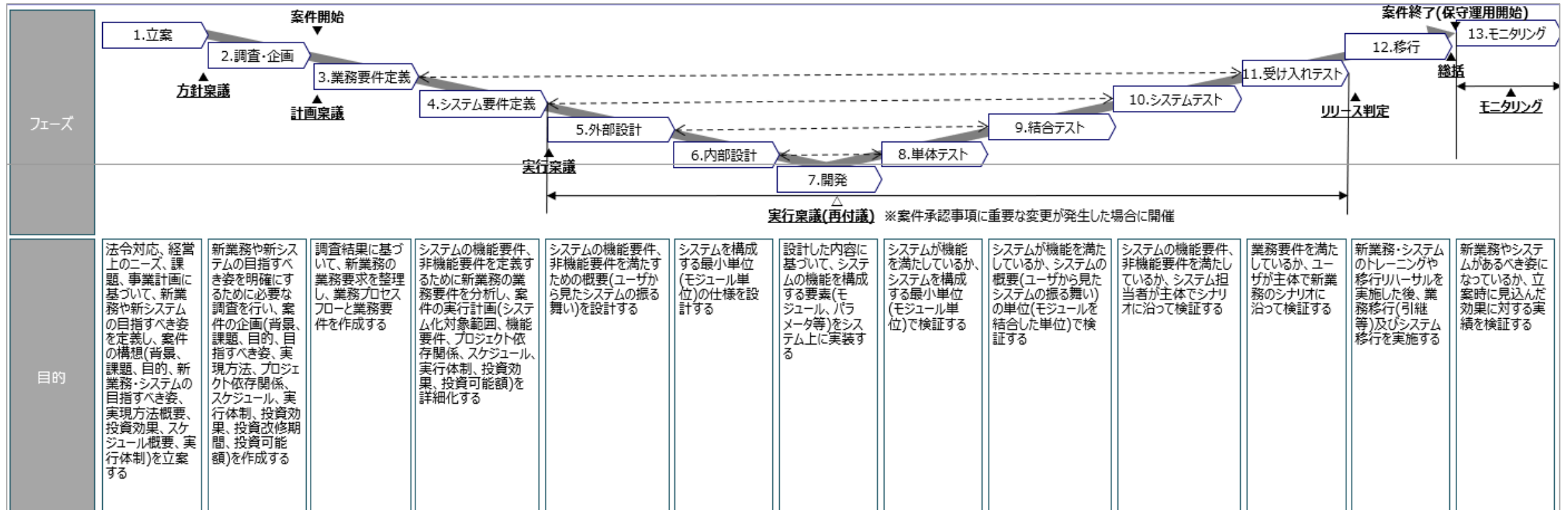


3. 取り組みの内容



■ 当社におけるスクラッチ開発のシステム開発プロセス紹介

ウォーターフォールモデルを前提としたオーソドックスな開発プロセスとなっている。

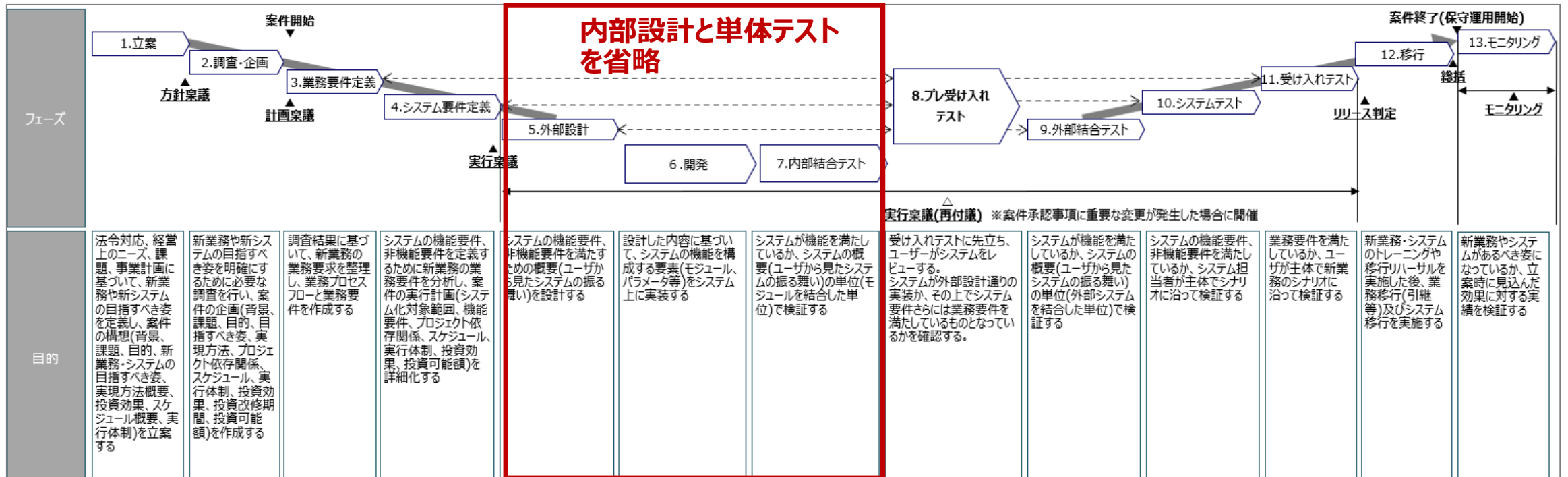


3. 取り組みの内容



■ 内部設計と単体テストを省略した

原則カスタムコードを書かないため、ローコードアプリケーション（以下LCAP）開発基盤では、ソースコードの作りこみや検証を不要として単体テストを省略した。

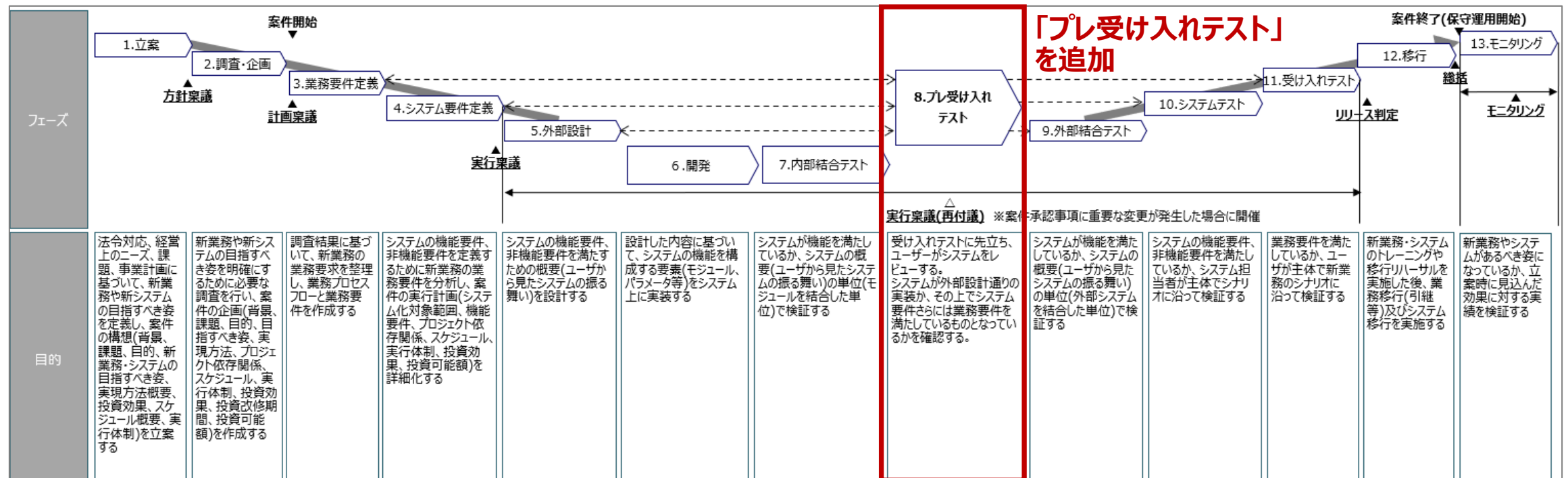


3. 取り組みの内容



■ 受け入れテストの一部を前倒しした

アプリケーションを実装するまでの期間が既存のモデルと比較して短縮されることを利用し、早期に業務側がアプリケーションの実物を確認することで開発の手戻りを防ぐため工程を追加した



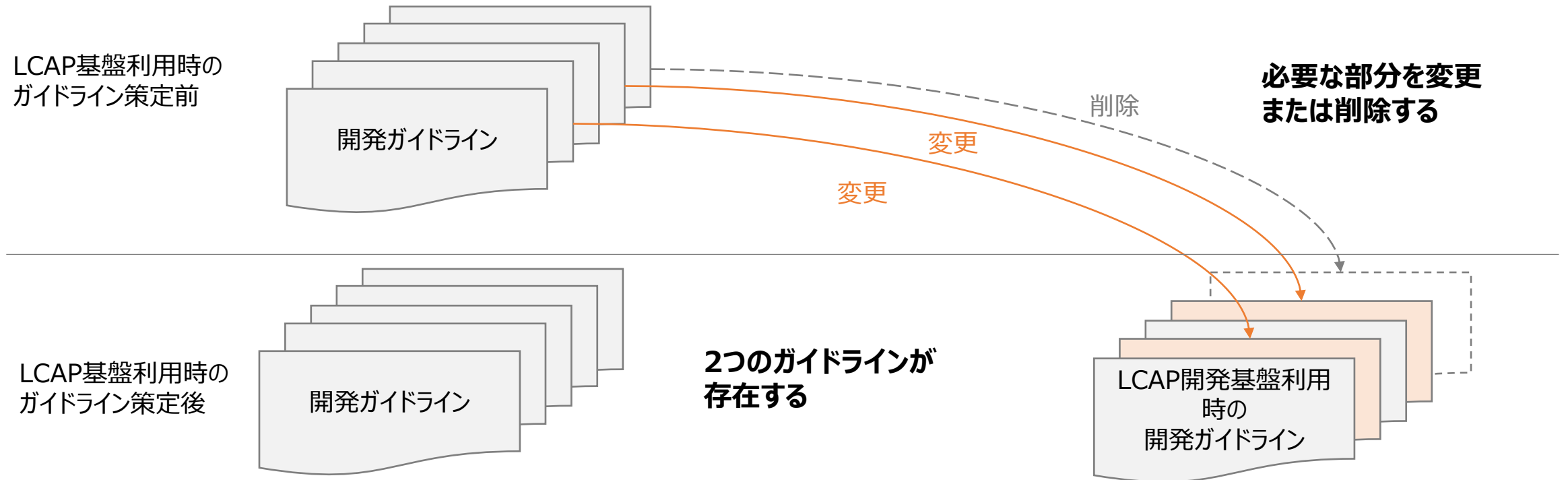
3. 取り組みの内容



■ 標準化ドキュメントについて

標準のシステム開発プロセスや成果物を定義したガイドラインを部分的に変更し、LCAP開発基盤利用時のガイドラインを追加した。

パターン追加のイメージ



3. 取り組みの内容

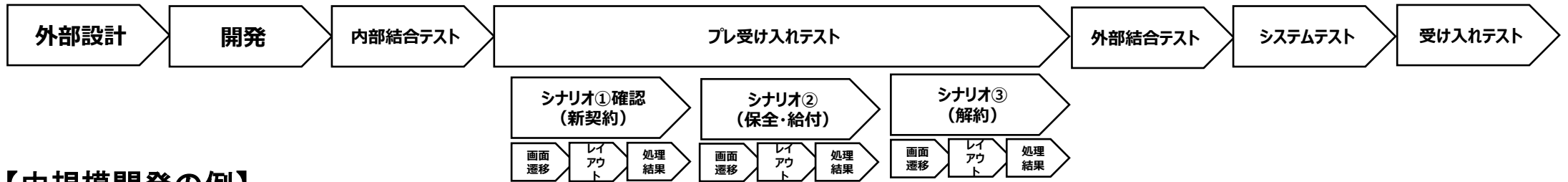


■ プレ受け入れテストは実施有無もテーラリング可能

✓ 「プレ受け入れテスト」を実行するのは開発規模が大きい（＝画面数、画面遷移パターン、インターフェース数、データ項目やバリエーションが多い）ケースを想定しているため、小規模開発では、テーラリングによる省略も可能としている

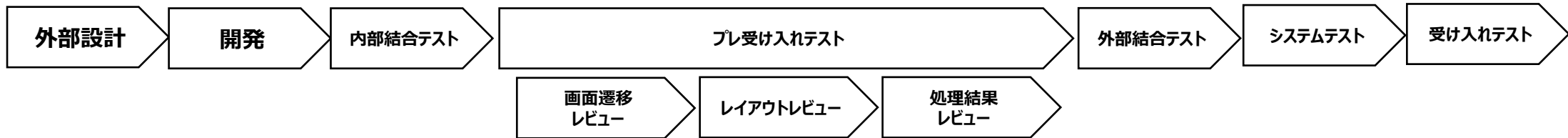
【大規模開発の例】

業務シナリオ順に画面遷移、レイアウト、処理結果をレビュー



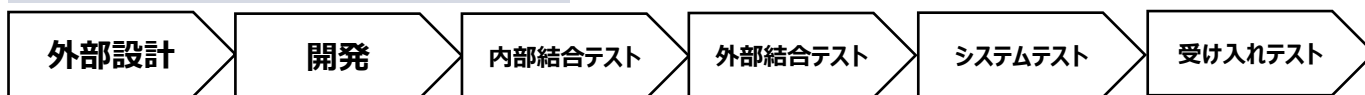
【中規模開発の例】

全体の画面遷移、レイアウト、処理結果を1度でレビュー



【小規模開発の例】

プレ受け入れテストは実施せず、受け入れテストで最終確認



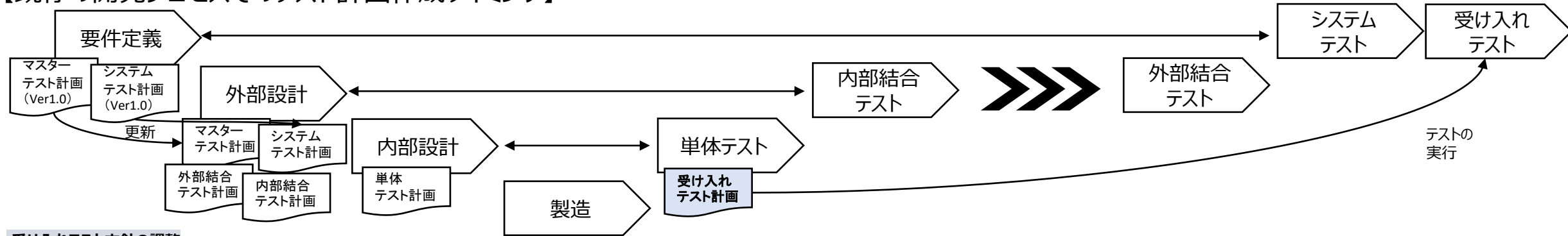
3. 取り組みの内容



■ プレ受け入れテストのテスト計画は、製造工程で作成する

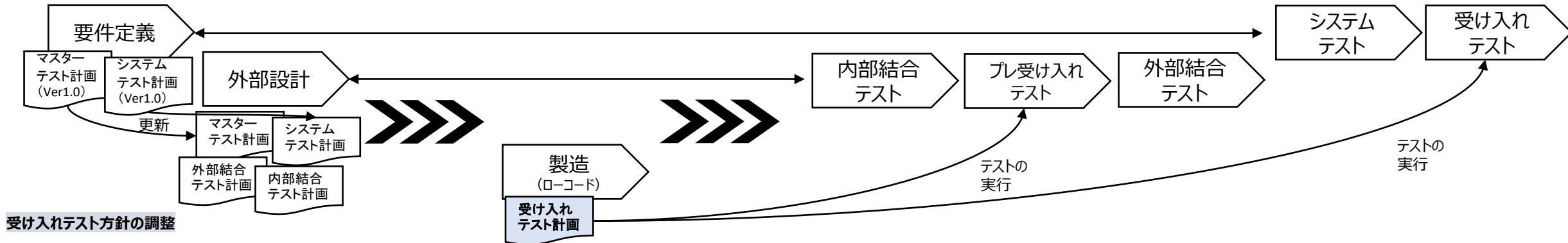
「プレ受け入れテスト」のテスト計画は、製造工程の完了までに作成する。受け入れテストの一部として受け入れテスト計画書に記載する。

【既存の開発プロセスでのテスト計画作成タイミング】



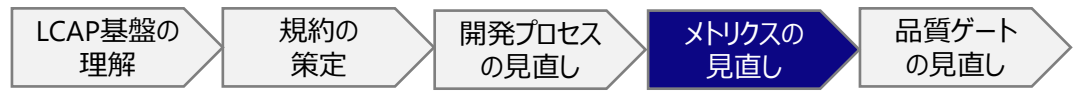
受け入れテスト方針の調整

【LCAP開発基盤上での開発案件におけるテスト計画作成タイミング】



受け入れテスト方針の調整

3. 取り組みの内容



■ メトリクスの見直し

規模見積のインプットであること、ソフトウェア開発データ白書の目標値の基本測定量として使われていることから、ソースコード行数の代わりにファンクションポイントを規模の基本測定量として定義した。

品質計画書テンプレート_v1.0.1.pptx

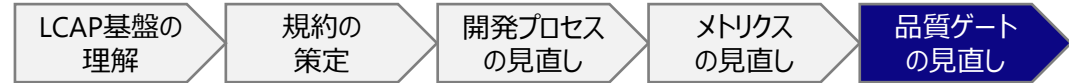
想いを、心に響くカタチに。

品質メトリクスの目標

工程	管理値	目標	目標の根拠
システム要件定義	欠陥検出密度 (件/頁)	××	過去の案件の実績データを蓄積して得たデータを元に設定
外部設計	欠陥検出密度 (件/頁)	××	過去の案件の実績データを蓄積して得たデータを元に設定
内部設計※1	欠陥検出密度 (件/頁)	××	委託先のベンダーが持っている目標値を設定
製造・単体テスト※1	カバレッジ	C1 (100%)	
	テスト密度 (ケース/KLOC)	××	委託先のベンダーが持っている目標値を設定
	欠陥検出密度 (件/KLOC)	××	委託先のベンダーが持っている目標値を設定
内部結合テスト	テスト密度 (ケース/KLOCまたはFP※2)	××	委託先のベンダーが持っている目標値を設定
	欠陥検出密度 (件/KLOCまたはFP※2)	××	委託先のベンダーが持っている目標値を設定
外部結合テスト	テスト密度 (ケース/KLOCまたはFP※2)	××	過去の案件の実績データを蓄積して得たデータを元に設定
	欠陥検出密度 (件/KLOCまたはFP※2)	××	過去の案件の実績データを蓄積して得たデータを元に設定
システムテスト	テスト密度 (ケース/KLOCまたはFP※2)	××	過去の案件の実績データを蓄積して得たデータを元に設定
	欠陥検出密度 (件/KLOCまたはFP※2)	××	過去の案件の実績データを蓄積して得たデータを元に設定

※1 ローコード開発の場合は内部設計・単体テスト工程がないため定義不要 ※2 FP=ファンクションポイント...ローコード開発の場合はKLOCの代わりにFPを使用してください

3. 取り組みの内容

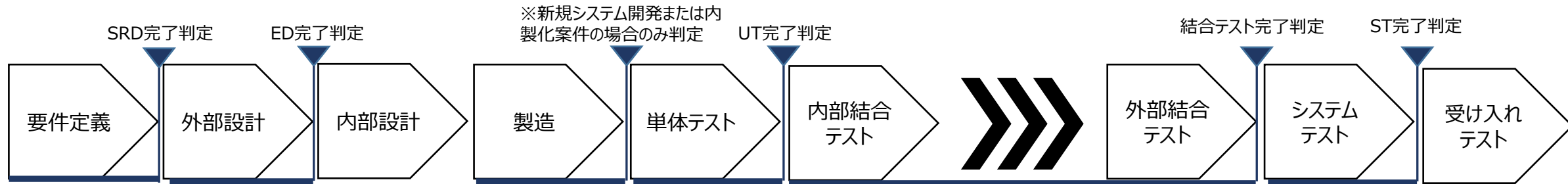


■ 品質ゲートの見直し

LCAP開発基盤上での開発における品質ゲート（工程完了判定プロセス）

- I. 製造工程の完了判定は、ソースコードの品質保証を実施しているかの確認であり、実施しない
- II. 単体テスト工程は省略されるため、完了判定は実施しない
- III. 「プレ受け入れテスト」の完了判定については、結合テストの完了判定に包含する

【既存のスクラッチ開発を前提に実施している工程完了判定タイミング】



【LCAP開発基盤上での開発案件における工程完了判定タイミング】



I. 製造工程の完了判定は実施しない

II. 単体テスト工程の完了判定は実施しない

III. 「プレ受け入れテスト」の完了判定については、結合テストの完了判定に包含する。

4. 品質保証にあたり考えたこと

- LCAP開発基盤を活用する目的を明確にすること
- 目的から考えて、どの種類のLCAP開発基盤を採用するか選択すること
- LCAP基盤をどこで使うのか明確にすること
- 上記をもとに品質保証の取り組みをLCAP基盤向けにカスタムすること

5. 今後の展開



実際に適用した開発案件からのフィードバックによる継続的な改善



FPによるメトリクスの実績蓄積と有効性の確認

ご清聴ありがとうございました

本発表を通して、同様の取り組みをされている皆さまとの情報交換、意見交換に繋がればと考えています。

本発表に関するお問い合わせ先

オリックス生命保険株式会社
IT品質保証部
藤井 和弘
kazuhiko.fujii.dd@ins.orix.jp