



テスト設計コンテスト '14

NTTデータ流テスト設計プロセスの策定と実践 ～自動販売機のシステムテストを例に～

2014年3月8日

株式会社NTTデータ
技術開発本部 プロアクティブ・テストイングCOE
チーム「らくてす」

NTT DATA

0. チーム「らくてす」の紹介

■ メンバーの所属

- 株式会社NTTデータ 技術開発本部

プロアクティブ・テストニングCOE

- NTTデータグループ向けのソフトウェアテスト（主に機能テスト）に関するR&Dを行う組織
 - テストプロセスの策定・普及展開
 - 上記プロセスを実現したツールの開発・普及展開



「RACTES」
で検索！

■ 出場の目的

- 普段の業務は仕様書／設計書ベースのテスト詳細設計が中心

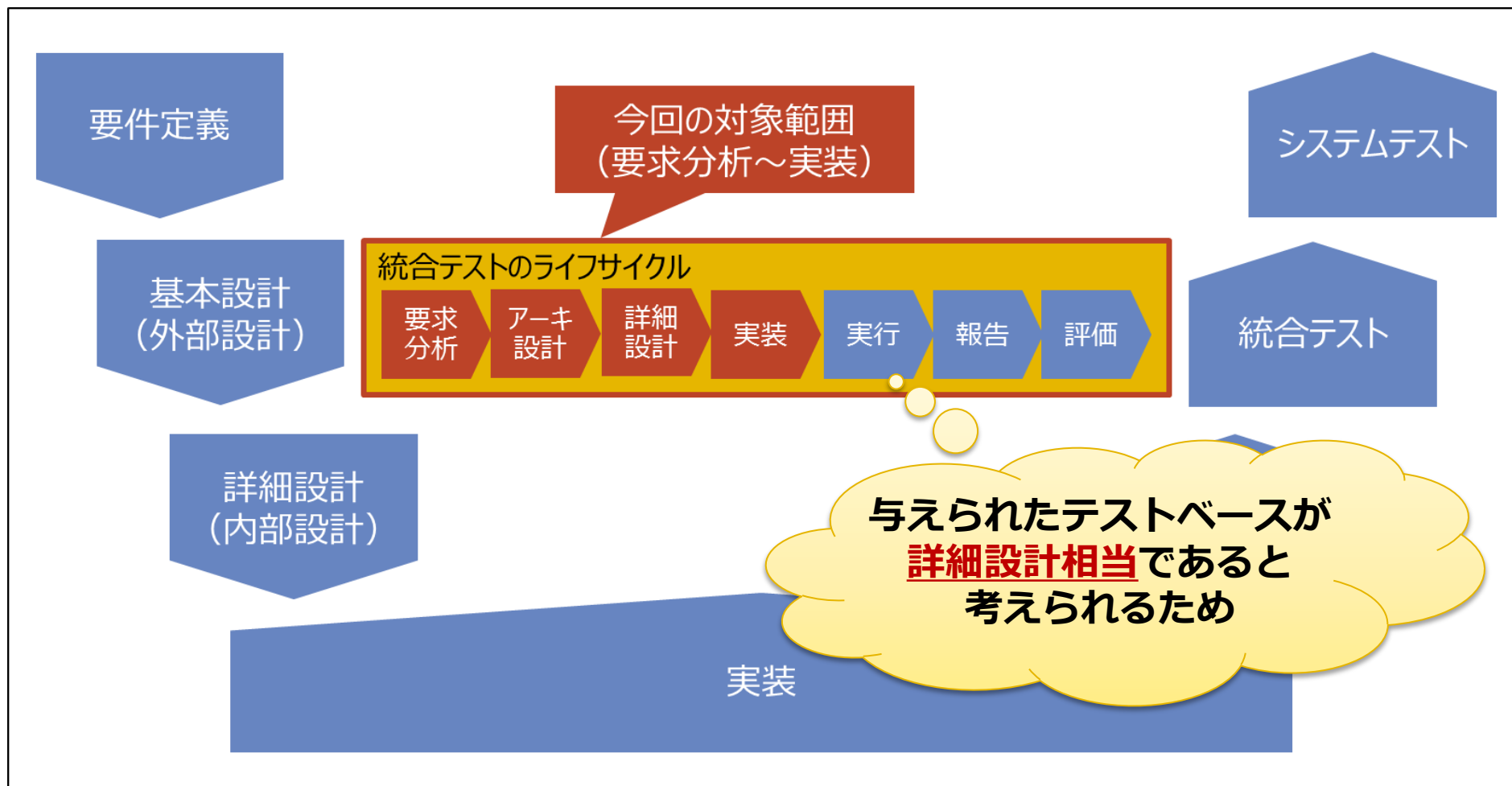
テスト設計コンテストはテスト要求分析／テストアーキテクチャについてじっくり考えて、かつ、手を動かすのに非常にいい機会

1. テスト方針
2. テスト要求分析
3. テストアーキテクチャ設計
4. テスト詳細設計
5. テスト実装
6. まとめ



1. テスト方針

■ Test.SSFにおける統合テストを想定



「SSFに基づくテスト技術フレームワーク」より

■ 以下の4点を全体方針とする

1. 業界標準に準拠したプロセスとする

JIS X 25010をベースにテスト設計を行う

2. 仕様書の情報に閉じない

仕様書に明記されていない情報も極力推測して補完する

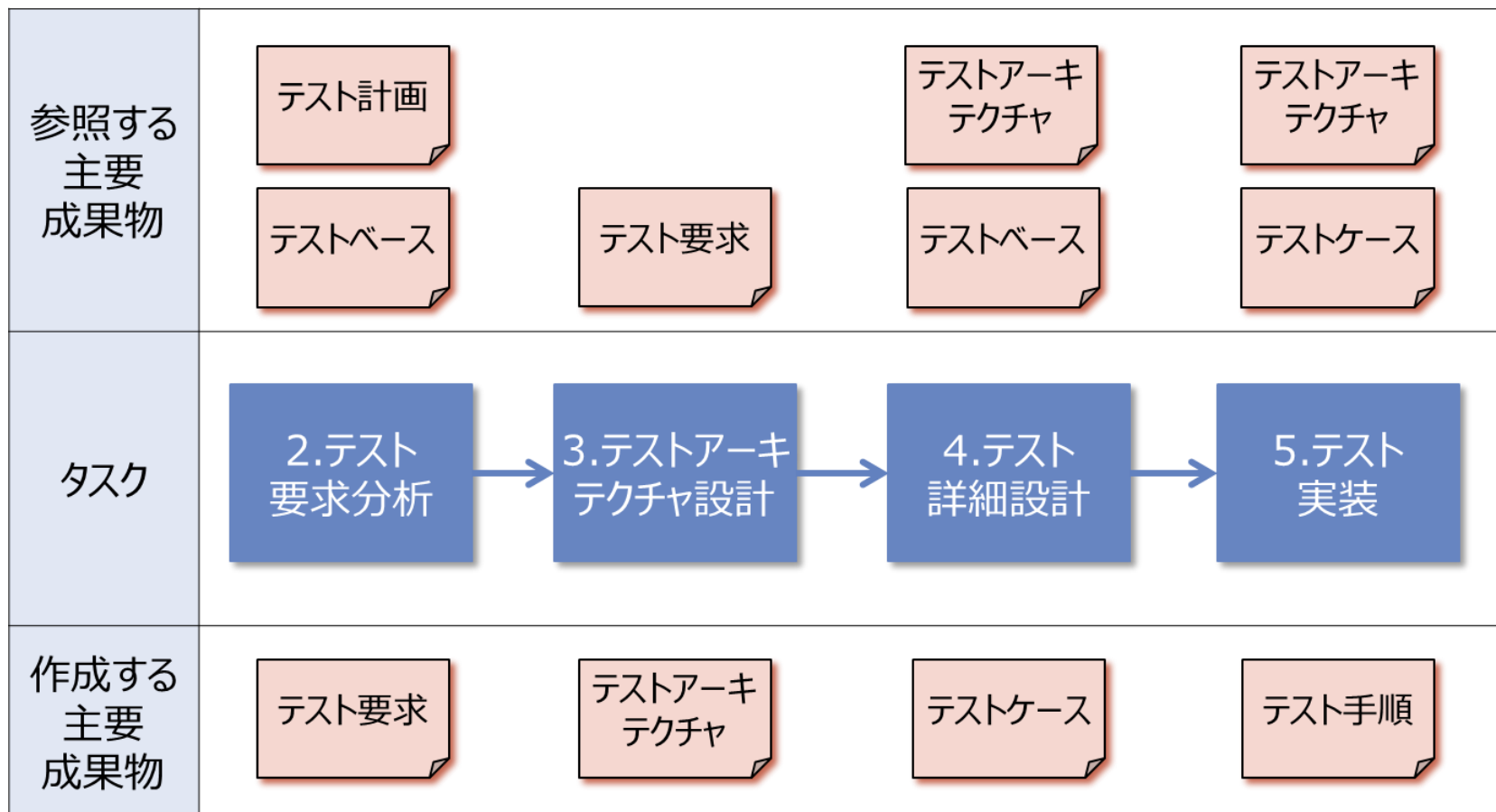
3. 成果物間のトレーサビリティをしっかりとる

成果物を見るだけで設計の意図がわかるようなプロセスにする

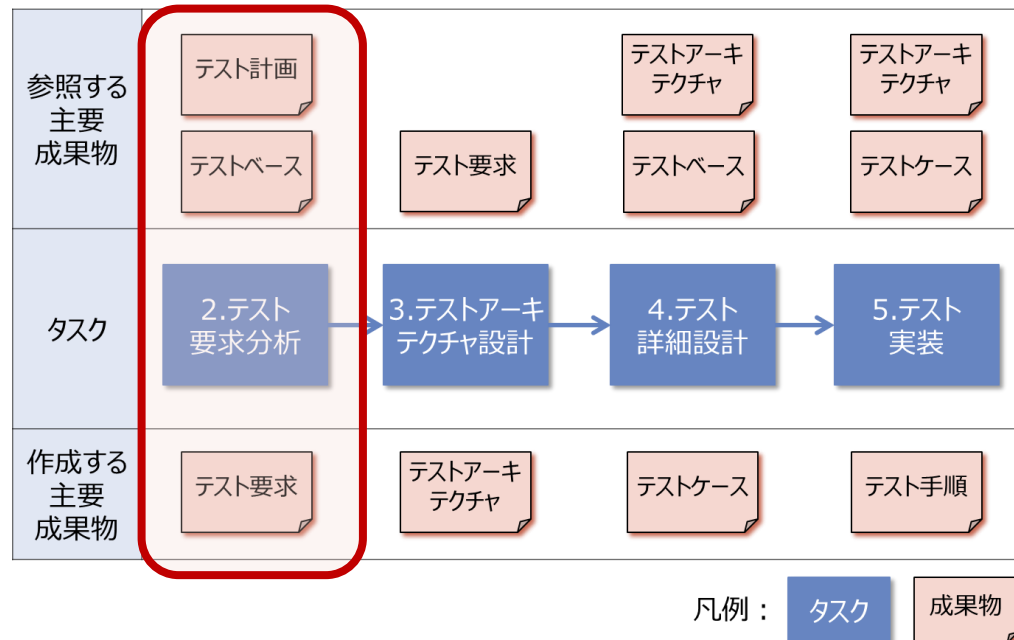
4. 客観的にわかりやすい

成果物間の関連が複雑でないシンプルなプロセスにする

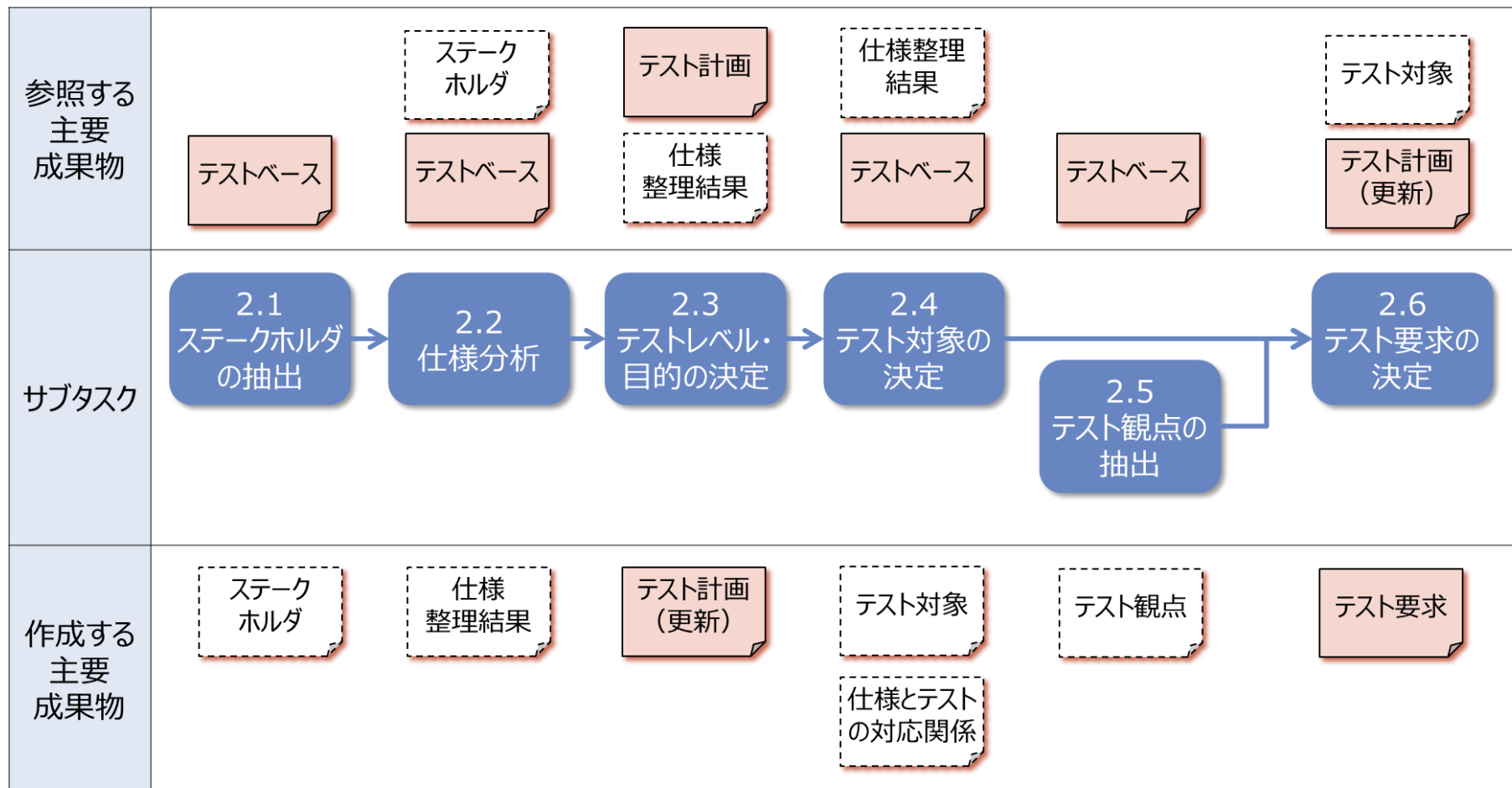
■ テスト設計作業を4つのタスクに分解して実施する



2. テスト要求分析



■ テスト要求分析作業を6つのタスクに分解して実施する



凡例：
サブタスク
成果物
中間成果物

2.1 ステークホルダーの抽出

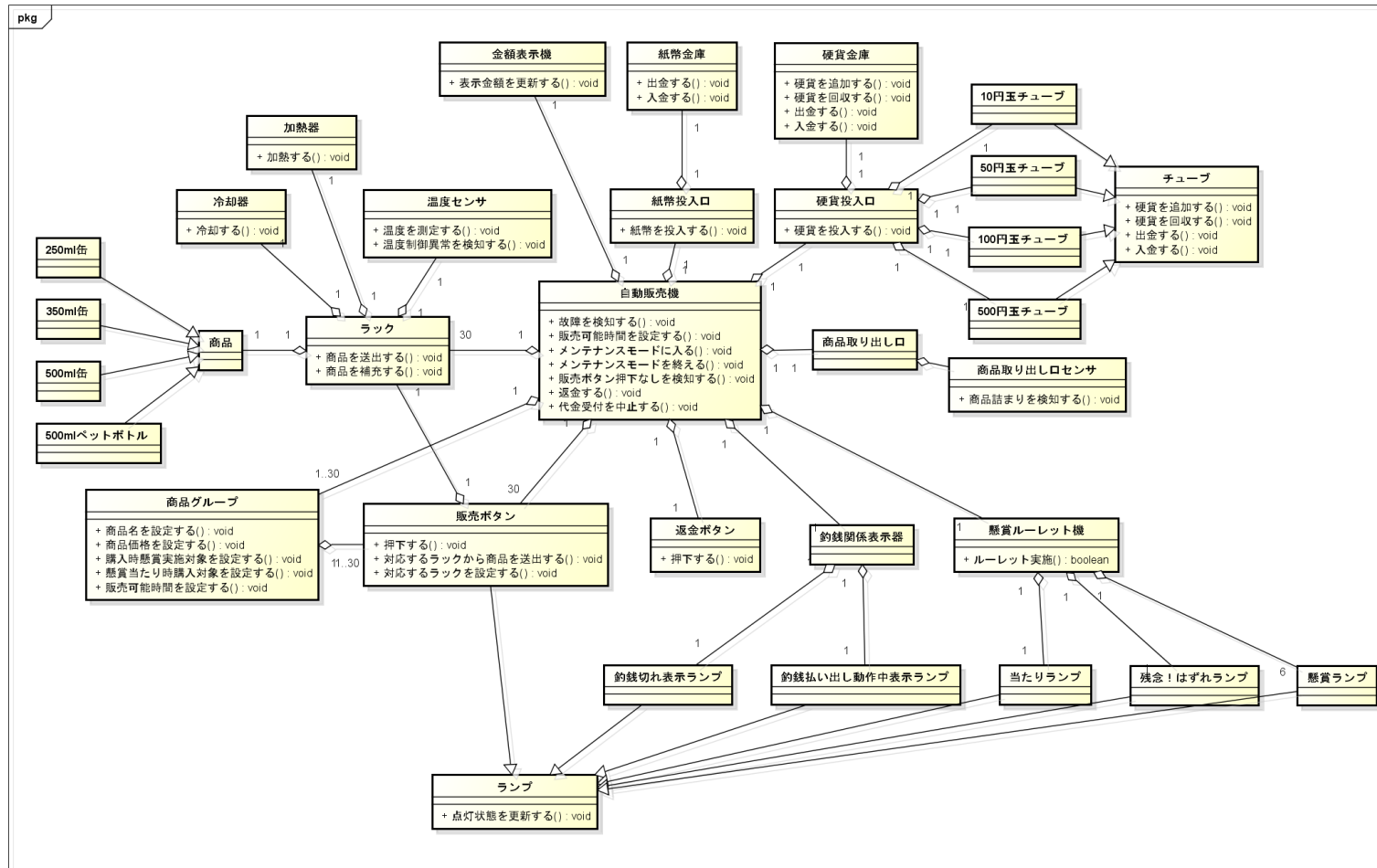
■ 自動販売機のライフサイクルを想定し、ステークホルダーを抽出

ライフサイクル	ステークホルダー	判定	理由
設置	設置業者	×	<u>ソフトウェアは関連しない</u> ため、テスト対象外とする
販売準備	販売者	○	販売者が準備する際のユースケースおよび関係する自動販売機の機能をテスト対象とする。 ただし、 <u>ソフトウェアが関連しない範囲は対象外</u> とする
販売中	商品購入者	○	商品を購入する際のユースケースおよび関係する自動販売機の機能をテスト対象とする
修理中	販売者	○	自動販売機を修理する際のユースケースおよび関係する自動販売機の機能をテスト対象とする
情報活用	情報解析担当者	×	売上データに関連する機能は対象外と明記されている
運用終了	撤去業者	×	<u>ソフトウェアは関連しない</u> ため、テスト対象外とする

2.2 仕様の分析(1/2)

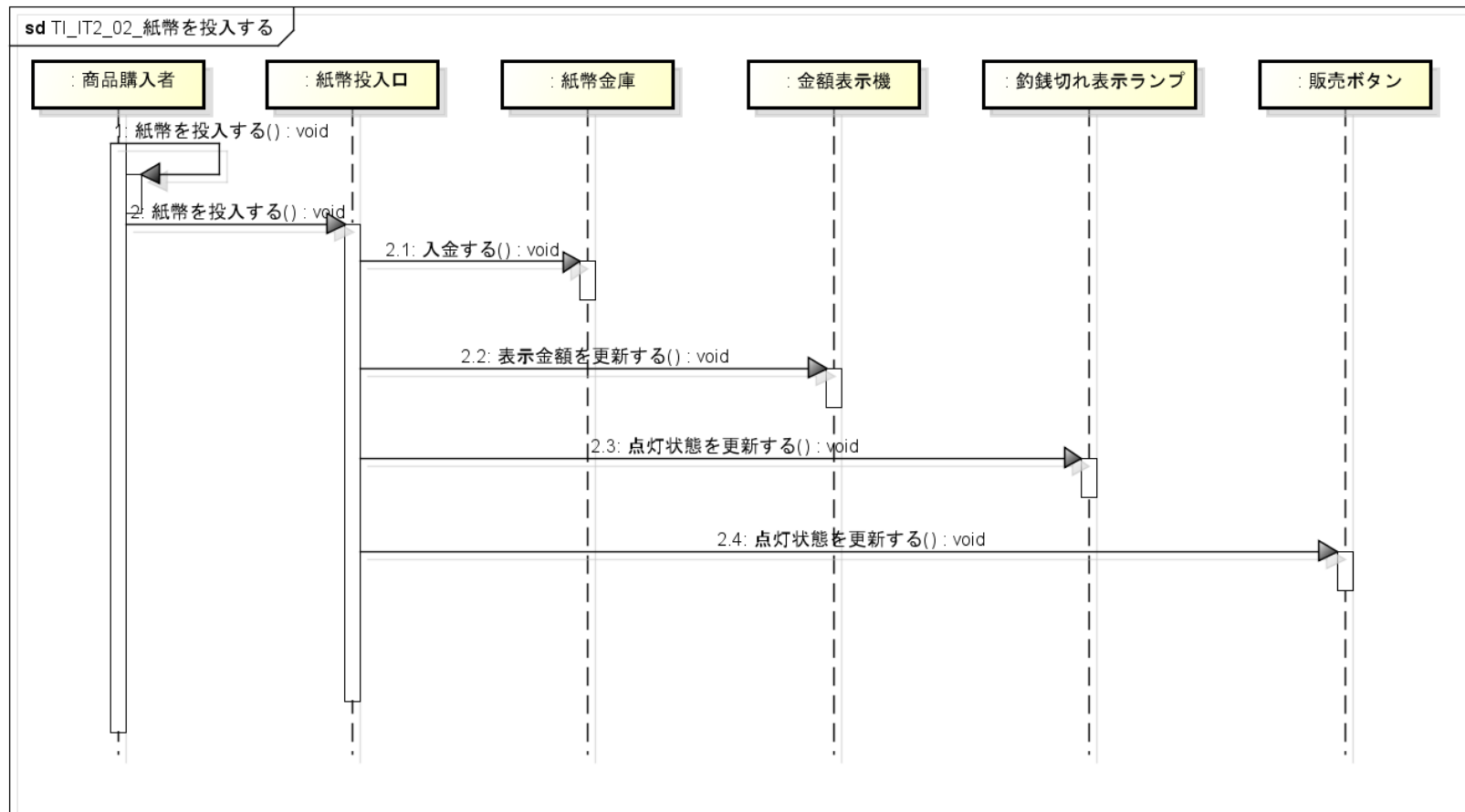
■ 記述された振る舞いをUMLを用いて表現

クラス図：構成要素をクラス、振る舞いをメソッドとして整理



■ 記述された振る舞いをUMLを用いて表現

シーケンス図：振る舞いごとにクラス間の相互作用を整理



■ 仕様分析の結果からテストレベルを決定

統合テストのレベル	テスト対象の単位	定義	当初計画時のテスト対象の単位
レベル1	機能	シーケンス図において <u>自動販売機の構成要素によって開始</u> され、かつその <u>結果が確認できるまで</u> を範囲とする。	ハードウェア構成要素／販売者機能
レベル2	アクション	シーケンス図において <u>アクター（ステークホルダ）の1回の操作</u> によって行われる一連の機能とする。	ユースケース

■ 品質特性をベースにテスト目体を決定

テスト対象の単位	品質特性	テスト目的
機能	機能適合性	機能が期待通りに動作することを確認する。
	性能効率性	機能の実行時間、および実行時の使用リソースが適切であることを確認する。
アクション	機能適合性	想定されるアクターが期待通りにアクションからの一連のユースケースを実行できることを確認する。
	性能効率性	アクションからの一連のユースケースの実行時間、および実行時の使用リソースが適切であることを確認する。
	使用性	想定されるアクターがアクションからの一連のユースケースを理解し、正しく実行できることを確認する。
	信頼性	アクションからの一連のユースケースを実行中に故障が発生した場合に、適切にリカバリができることを確認する。
	セキュリティ	適切なアクター以外ではアクションからの一連のユースケースを実行できないことを確認する。

■ テスト対象を列挙

レベル1：ハードウェアの機能を中心に列挙

テスト対象（機能）	テストベース名	テストベースの該当箇所
販売ボタン点灯	ハードウェア仕様書	1.2販売ボタン
	ユースケース仕様書	3.3販売ボタン
硬貨受理	ハードウェア仕様書	1.3貨幣投入口
紙幣受理	ハードウェア仕様書	1.3貨幣投入口
金額表示	ハードウェア仕様書	1.4金額表示機
商品送出	—	—
返金	ハードウェア仕様書	1.3貨幣投入口 1.8返金ボタン
懸賞	ハードウェア仕様書	1.9懸賞ルーレット機
	ユースケース仕様書	3.2自動販売機 3.3販売ボタン
釣銭切れ表示	ハードウェア仕様書	1.6釣銭取り出し口
		1.7釣銭関係表示器

2.4 テスト対象の決定(2/3)

■ テスト対象を列挙

レベル2 (1/3) : 商品購入者のとれるアクション

テスト対象 (アクション)	テストベース名	テストベースの該当箇所
硬貨を投入する	ユースケース仕様書	2.1代金投入ユースケース
紙幣を投入する	ユースケース仕様書	2.1代金投入ユースケース
販売ボタンを押下する	ユースケース仕様書	2.2商品選択ユースケース 2.4懸賞ユースケース
返金ボタンを押下する	ユースケース仕様書	2.3返金ユースケース

レベル2 (2/3) : センサー類のとれるアクション

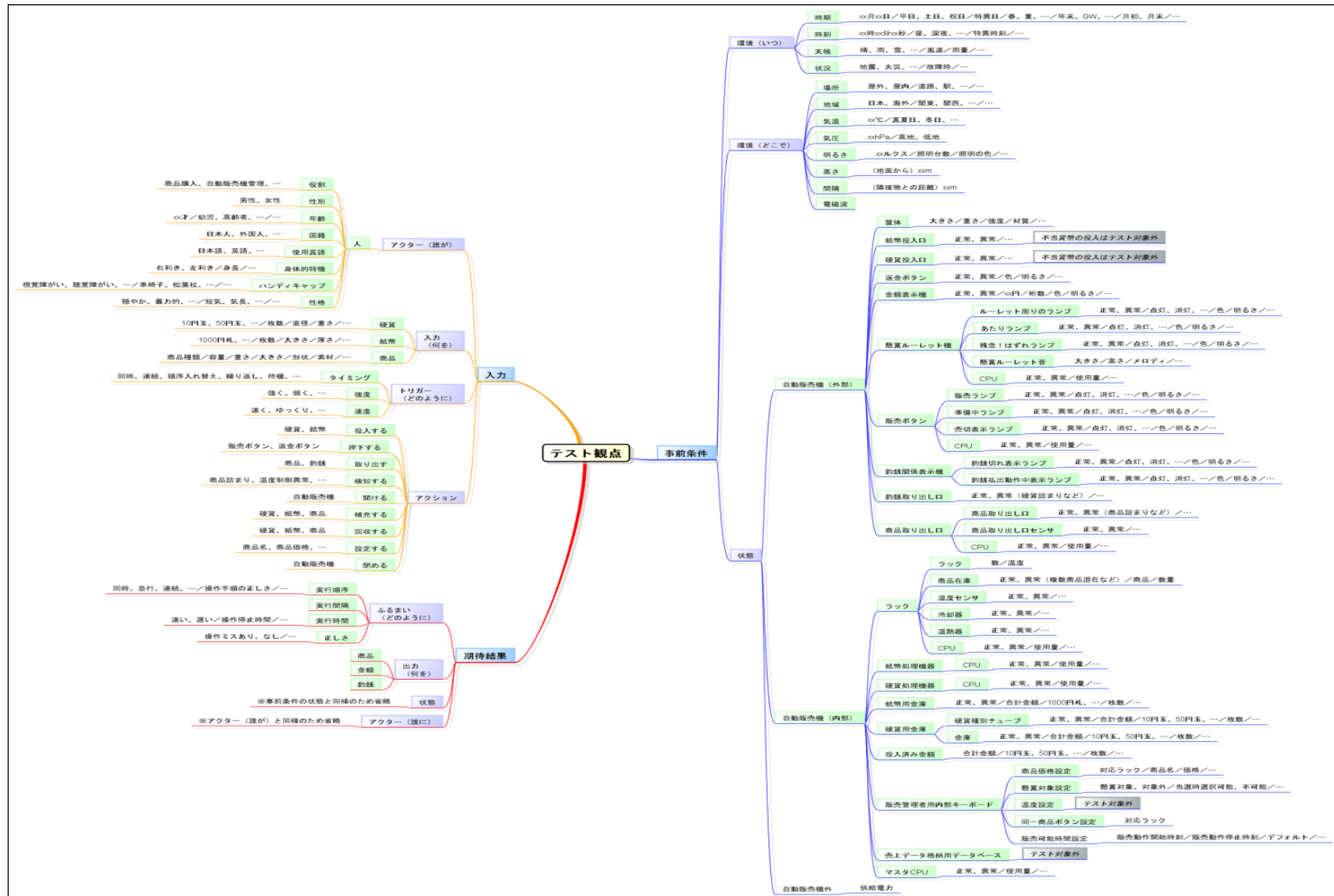
テスト対象 (ユースケース)	抽出元	該当箇所
販売ボタン押下なしを検知する	ハードウェア仕様書	1.3貨幣投入口
商品詰まりを検知する	ハードウェア仕様書	1.5商品取り出し口
温度制御異常を検知する	ハードウェア仕様書	2.1販売管理者用の内部キーボード (各商品の設定)
その他故障を検知する	ハードウェア仕様書 ユースケース仕様書	全体

■ テスト対象を列挙

レベル2 (3/3) : 販売者のとれるアクション

テスト対象 (アクション)	テストベース名	テストベースの該当箇所
メンテナンスを開始する	TR_IT2-41-F01	自動販売機がメンテナンス可能となることを確認する
商品を補充する	TR_IT2-42-F01	商品が正しく補充されることを確認する
商品名を設定する	TR_IT2-44-F01	商品名が正しく設定されることを確認する
商品価格を設定する	TR_IT2-45-F01	商品ごとの価格が正しく設定されることを確認する
以下、省略		

■ 5W1Hをベースに思考を発散



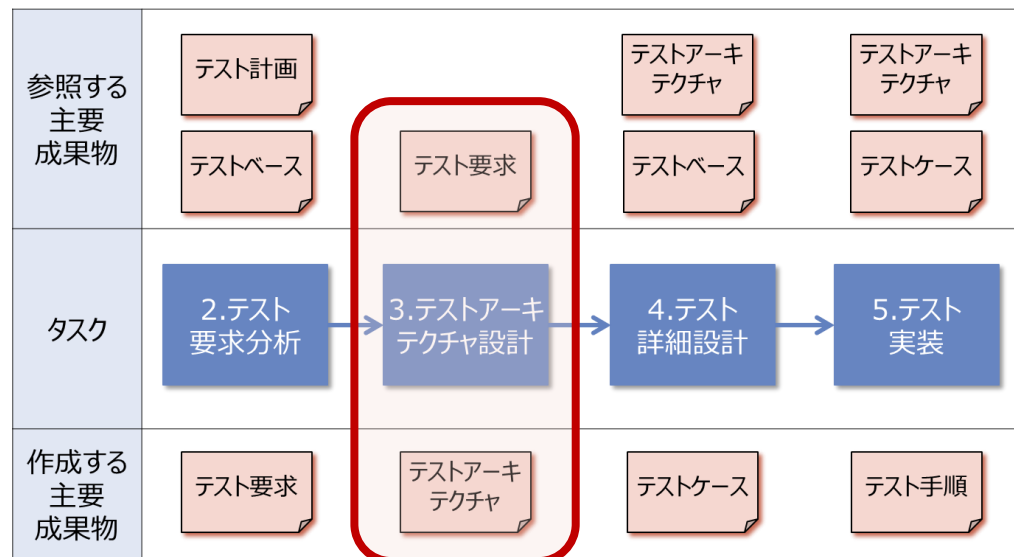
- 2.3のテスト目的と2.4のテスト対象を対応づける
機能に対する機能性のテスト要求の例

テスト対象（機能）	ID	テスト要求
販売ボタン点灯	TR_IT1-01-F01	販売ボタンの点灯が正しく実施されることを確認する
硬貨受理	TR_IT1-02-F01	硬貨の受理が正しく実施されることを確認する
紙幣受理	TR_IT1-03-F01	紙幣の受理／非受理が正しく実施されることを確認する

アクションに対する性能効率性の例

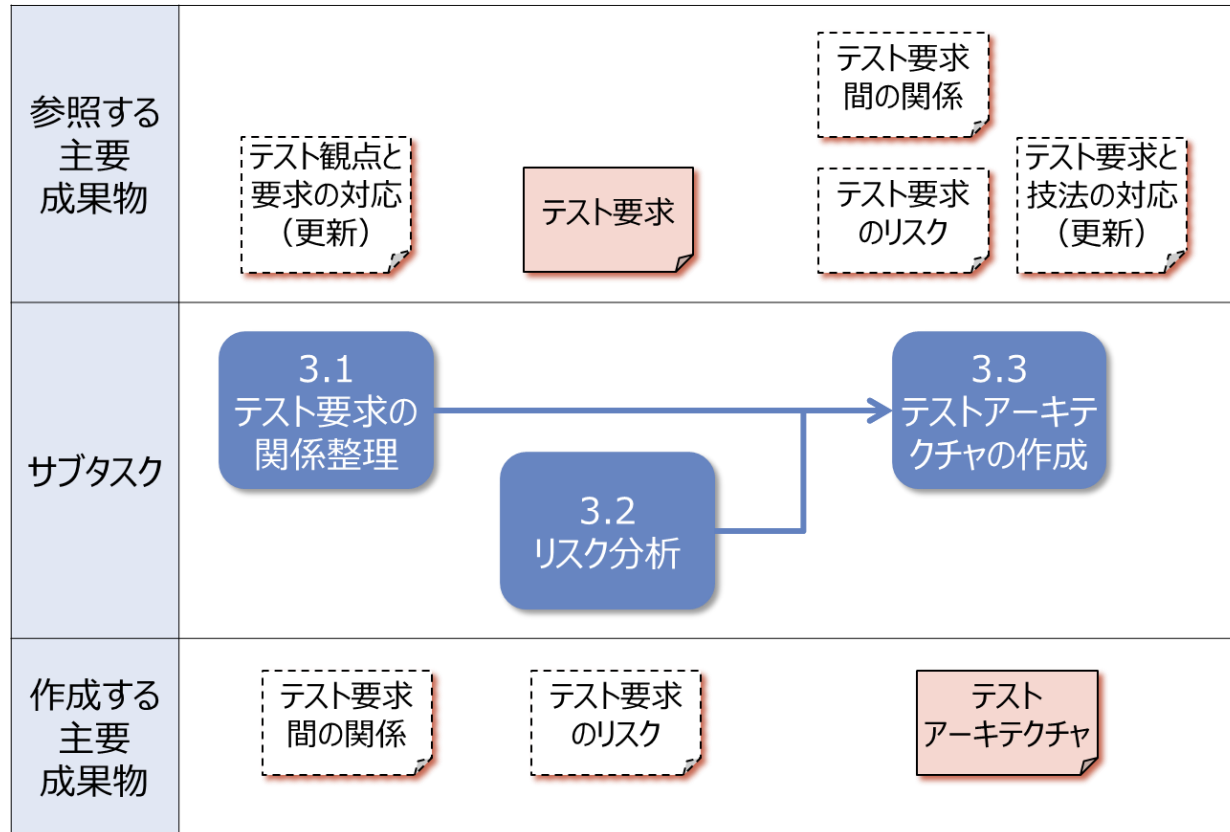
テスト対象（アクション）	ID	テスト要求
硬貨を投入する	TR_IT2-01-P01	硬貨の連続投入間隔が仕様を満たしていることを確認する
紙幣を投入する	TR_IT2-02-P01	紙幣の連続投入間隔が仕様を満たしていることを確認する

3. テストアーキテクチャ設計



凡例：
タスク
成果物

- テストアーキテクチャ設計作業を3つのタスクに分解して実施する



- 重複するテストの発見／テスト順序の決定のために、テスト要求間の依存関係を整理

機能とアクション間の整理

テスト対象 (アクション)	テスト対象 (機能)							
	販売ボタン点灯	硬貨受理	紙幣受理	金額表示	商品送出	返金	懸賞	釣銭切れ表示
硬貨を投入する	○	○		○				○
紙幣を投入する	○		○	○				○
販売ボタンを押下する	○	○		○	○	○	○	○
返金ボタンを押下する	○			○		○		○

- テストの優先度を決定するために、ステークホルダに発生しえるリスクを分析

想定される事象および危害（ステークホルダが“販売者”のテスト対象）

テスト対象	テスト要求	想定される事象	想定される危害
メンテナンスを開始する	自動販売機がメンテナンス可能となることを確認する	メンテナンス状態にならない	メンテナンスを実施できない
商品を補充する	商品が正しく補充されることを確認する	商品が入らない	商品を補充できない
		商品が「販売可能」状態にならない	商品を販売できない
	商品が妥当な時間内で最適温度になる（「販売中」になる）ことを確認する	販売可能になるまで時間がかかる 加熱/冷却の電力効率が悪い	商品を販売できない 電気代が多くかかる 法令・ガイドラインに違反する

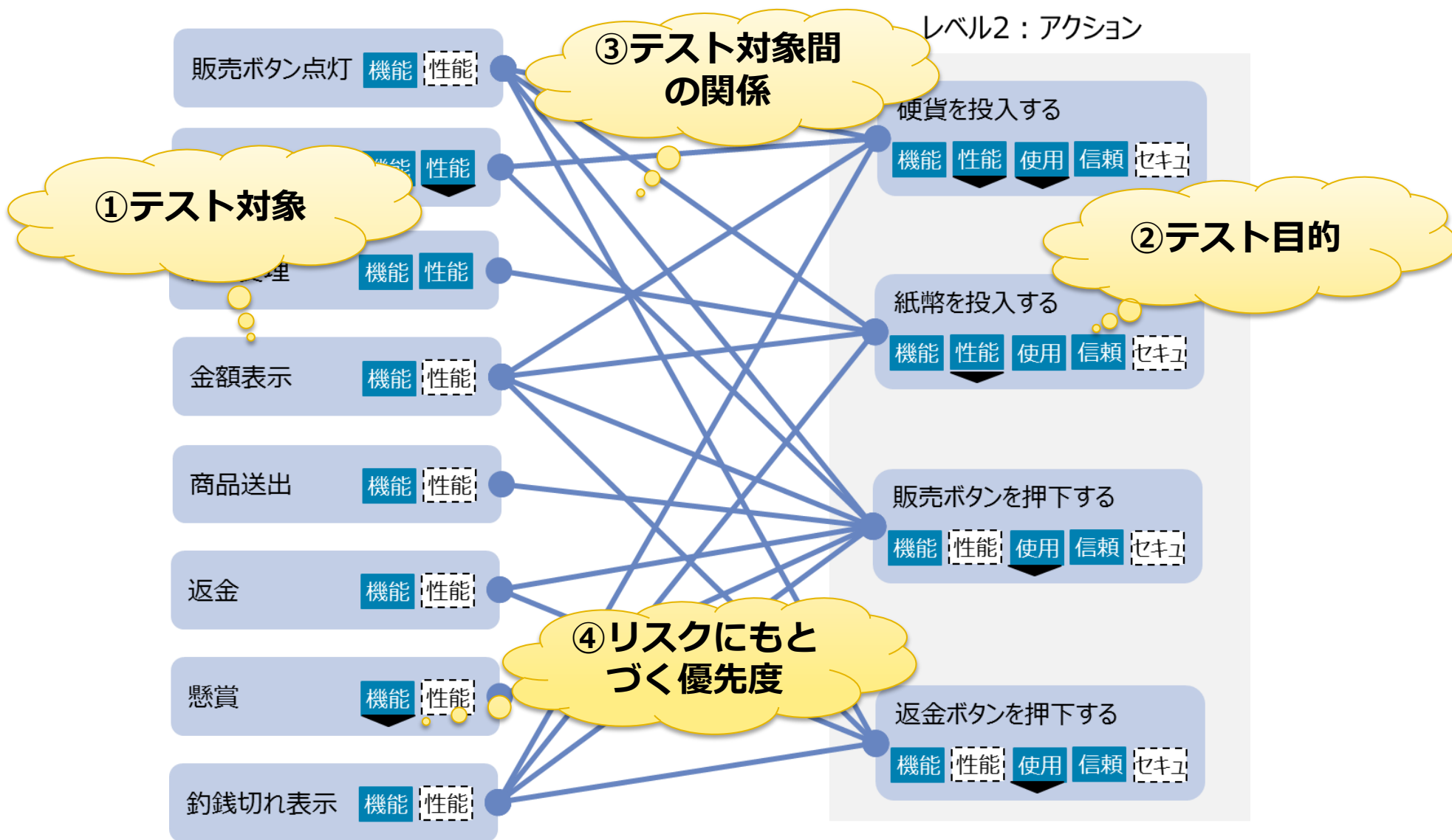
■ 高／中／低で三段階評価しアクションを決定

リスク度	作業方針			
	テスト設計・実装・実行の順序	テスト設計のレビュー方針	テスト設計者のスキル	テスト実行時の証跡保存
高	1番目	レビューでの指摘内容を修正後、再レビューの必要が必要テスト責任者の承認が必要	自動販売機のテスト設計の経験があること	全てのテストケースの証跡を保存する テスト責任者が証跡を確認する
中	2番目	レビューでの指摘内容を修正後、再レビューの必要が必要 テスト責任者の承認が不要		全てのテストケースの証跡を保存する
低	3番目	レビューでの指摘内容を修正後、再レビューの必要なし テスト責任者の承認が不要	特になし	インシデントが発生した場合にのみ証跡を保存する

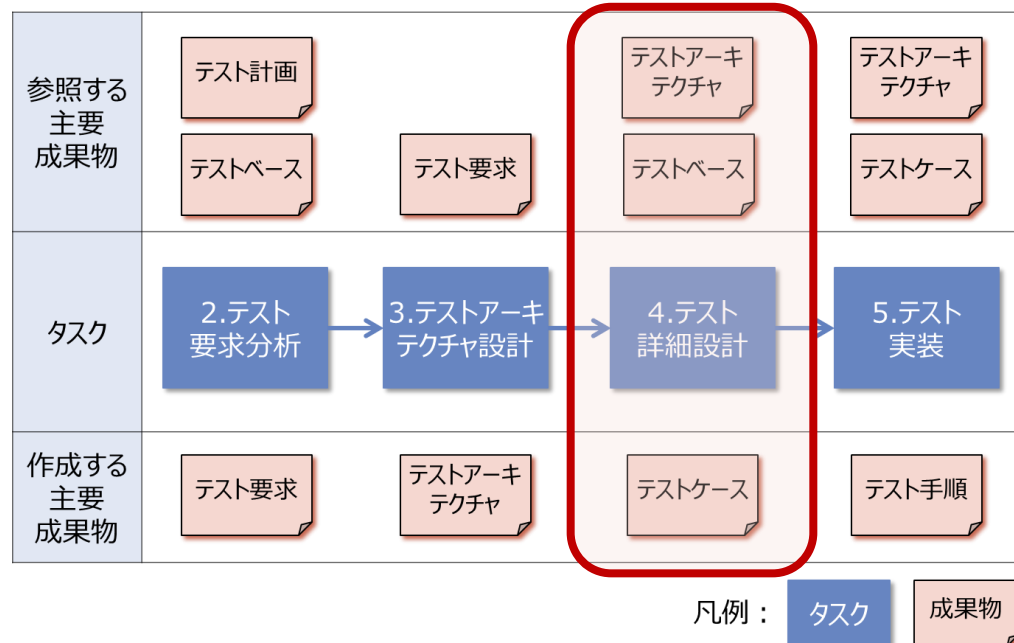
3.3 テストアーキテクチャ

レベル1：機能

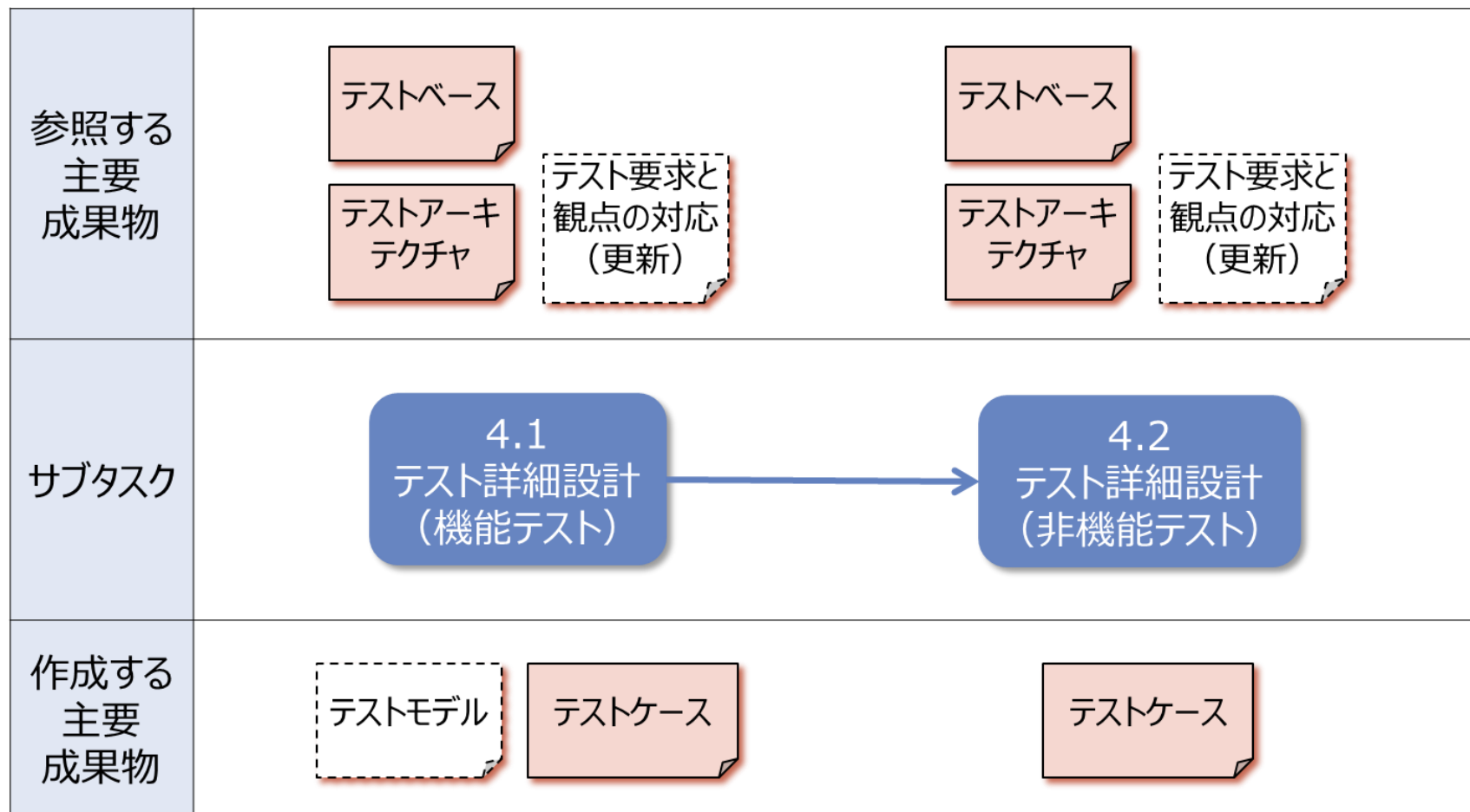
レベル2：アクション



4. テスト詳細設計



■ テスト詳細設計作業を2つのタスクに分解して実施する



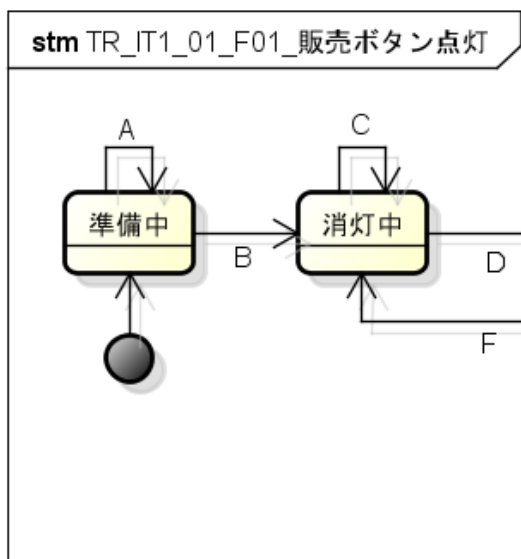
※テストレベル別の実施する



■ 状態遷移テスト + デシジョンテーブルテスト

例：販売ボタン点灯機能

機能のライフサイクルを
状態遷移図でモデリング



因子	水準	1	2	3	4	5	6
事前条件	販売ボタン点灯	準備中	Y	N	N	N	N
		消灯中	N	Y	Y	N	N
		点灯中	N	N	N	Y	Y
		売切れ	N	N	N	N	Y
残高 + 投入硬貨金額	商品の価格以上	-	Y	N	-	Y	-
	商品の価格未満	-	N	Y	-	N	-
懸賞状態	懸賞中である	-	-	-	Y	N	-
	懸賞中でない	-	-	-	N	Y	-
入力値	硬貨を投入する	10円	-	-	-	-	-
		50円	-	-	-	-	-
		100円	-	-	-	-	-
		500円	-	-	-	-	-
販売ボタン点灯	準備中	X	-	-	-	-	-
	消灯中	-	-	X	-	-	-
	点灯中	-	X	-	X	X	-
	売切れ	-	-	-	-	-	X
遷移		A	D	C	E	E	H

遷移ごとの詳細な振る舞いを
デシジョンテーブルで整理

■ テスト要求ごとにテスト観点にもとづいて作成

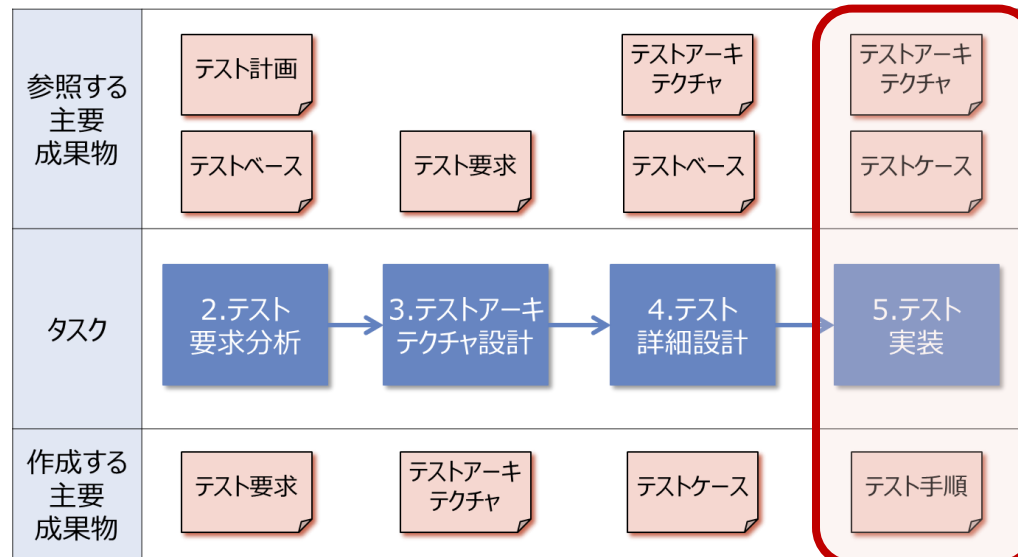
例：商品を補充する（性能テスト）

テストケース		
事前条件	入力値	期待結果
<ul style="list-style-type: none">商品状態が「準備中」であること。商品が「あたたか〜い」対象商品であること。外気温が24度（春・秋を想定）であること。	商品が最適温度（あたたか〜い）になるまで待機する。	商品が妥当な時間内（1時間）で「販売中」になること。

環境（Where）から作成

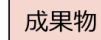
テスト要求に対応づく
テストケースを作成

5. テスト実装

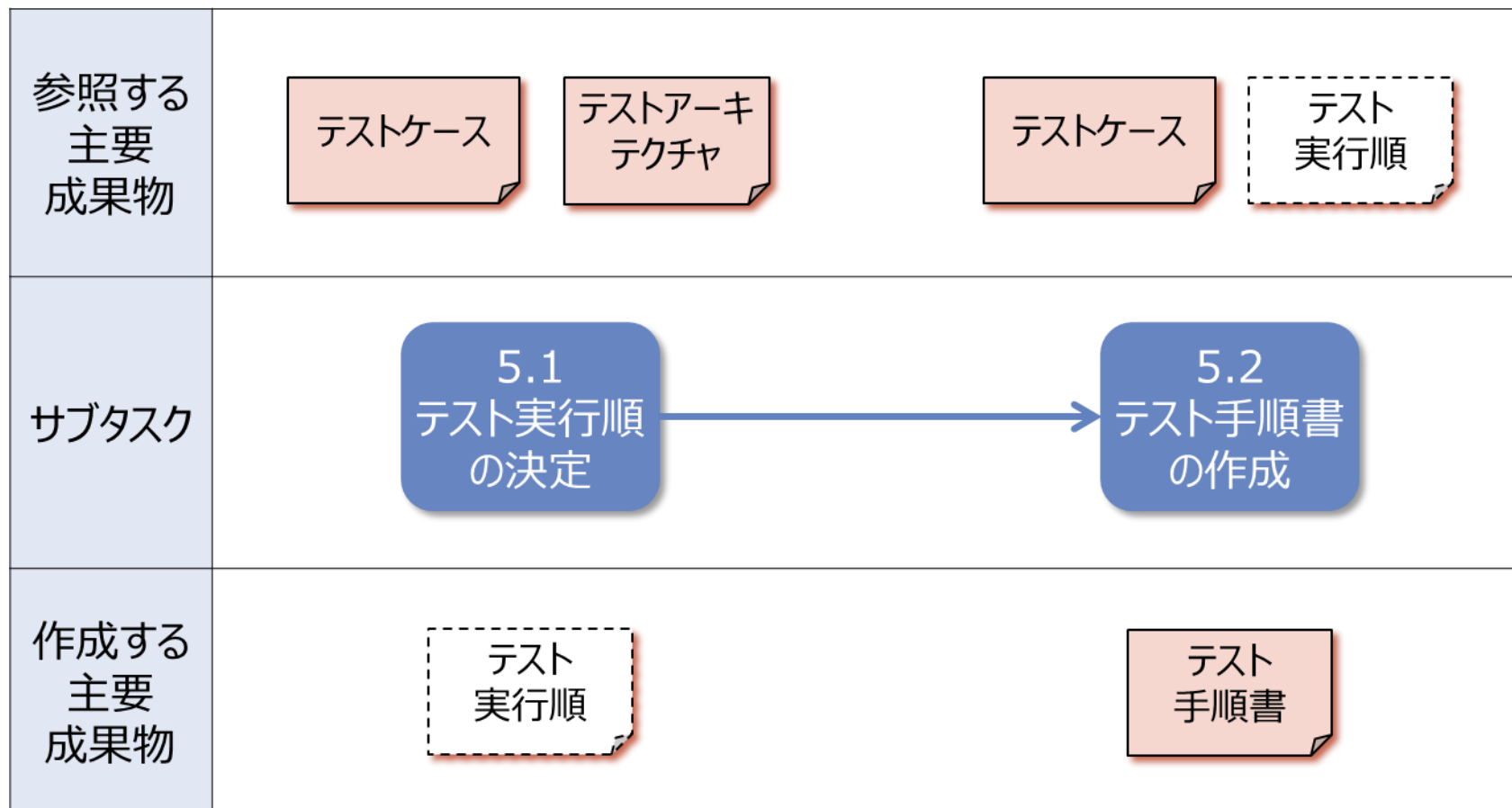


凡例 :

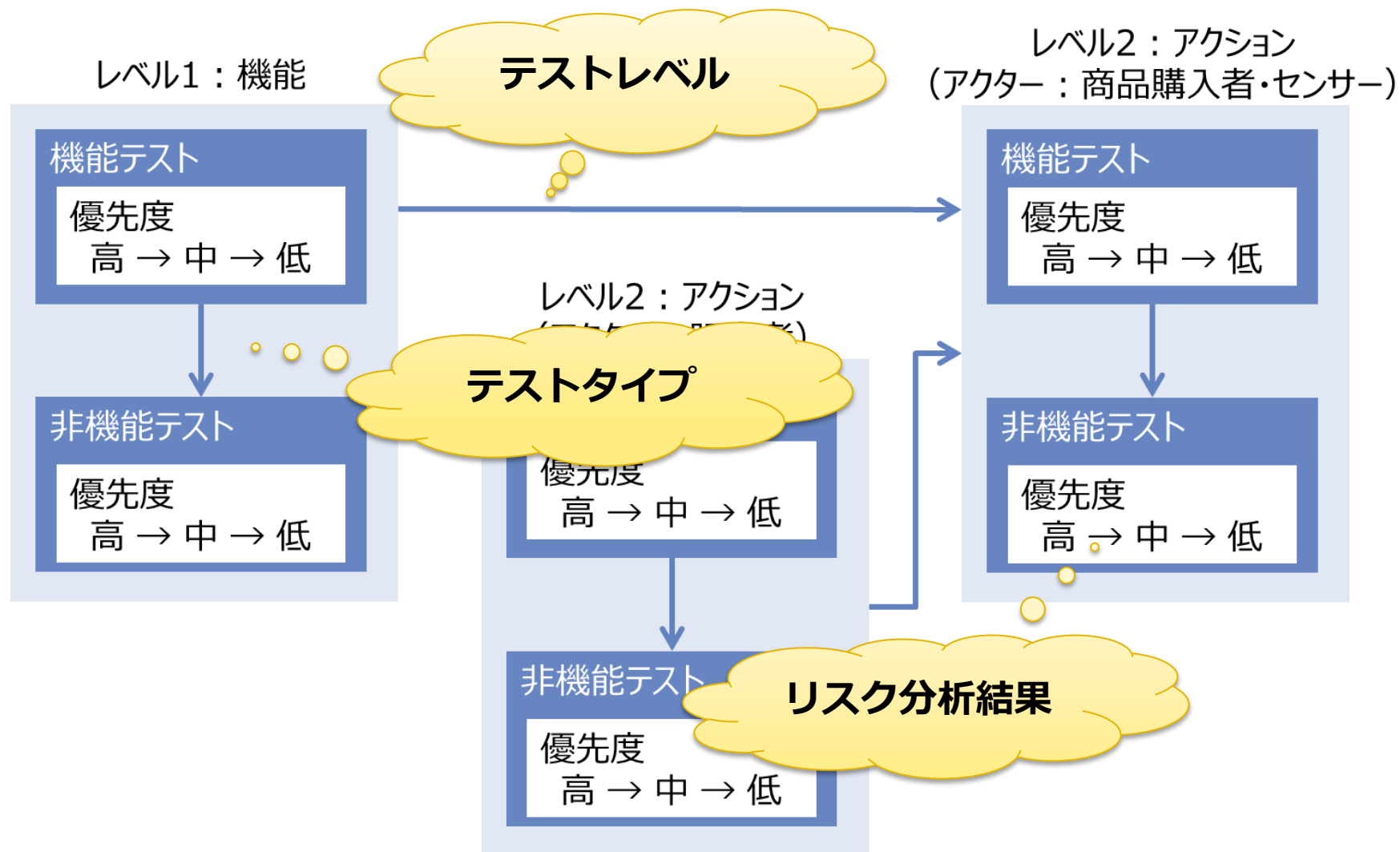
 タスク

 成果物

■ テスト実装作業を2つのタスクに分解して実施する



■ テストアーキテクチャにもとづいて実行順序を整理



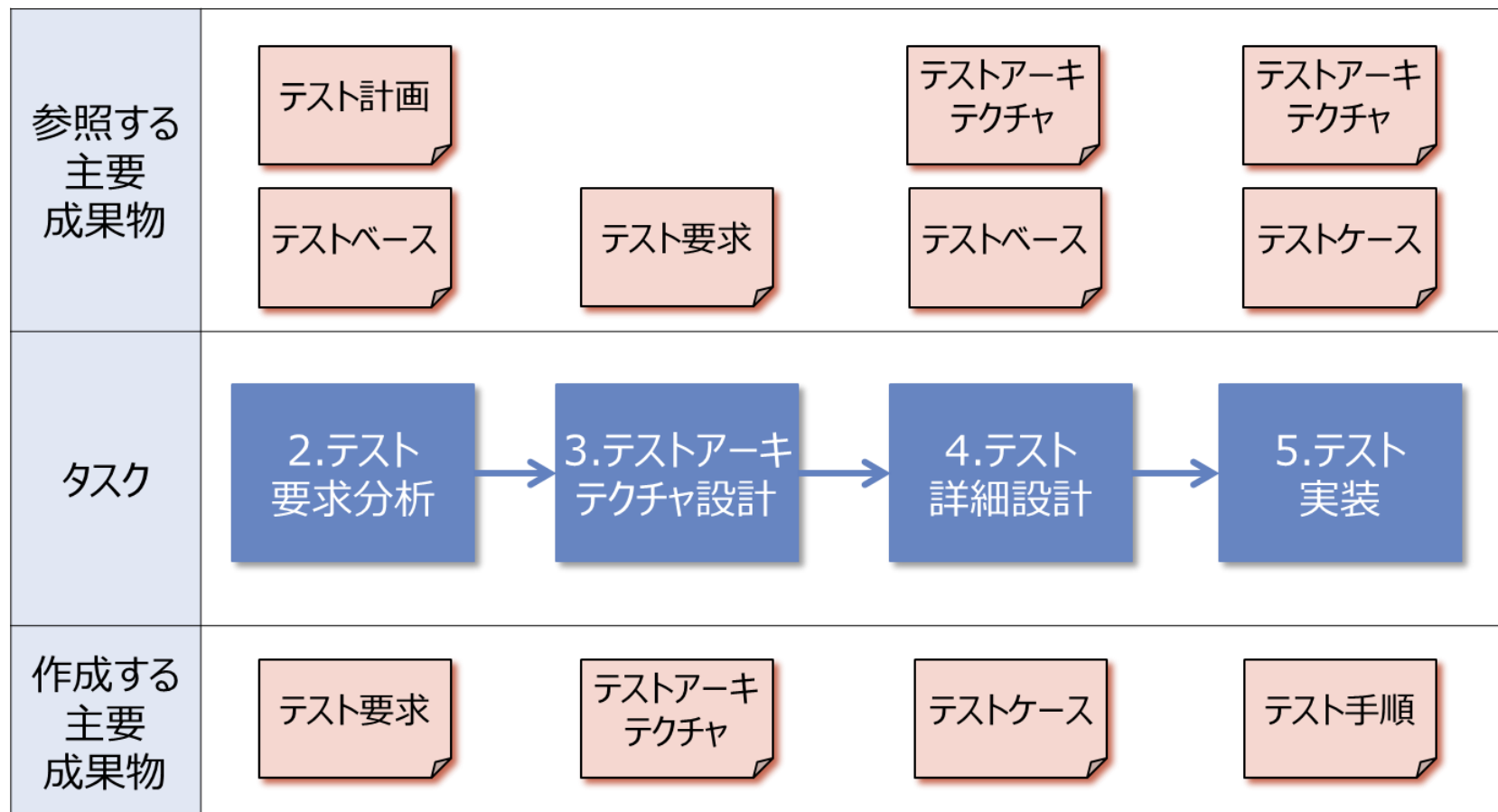
■ テスト手順書には以下の内容を記述

- まとめて実行するテストケースのセットの定義
 - テストスイートとして作成する
- 実行するテストケースの順序
 - テストケースのIDやテストケースのファイル名などにより示す
- すべてのテストケースを実行する前、および実行した後に行う作業
 - テスト環境構築…etc
- 各テストケースを実行する間に行う作業
 - 初期化
 - 事前状態の設定…etc



6. まとめ

■ テスト設計作業を4つのタスクに分解して実施した



■ 以下の4点を全体方針とした

1. 業界標準に準拠したプロセスとする

JIS X 25010の品質特性をベースにテスト設計を行った

2. 仕様書の情報に閉じない

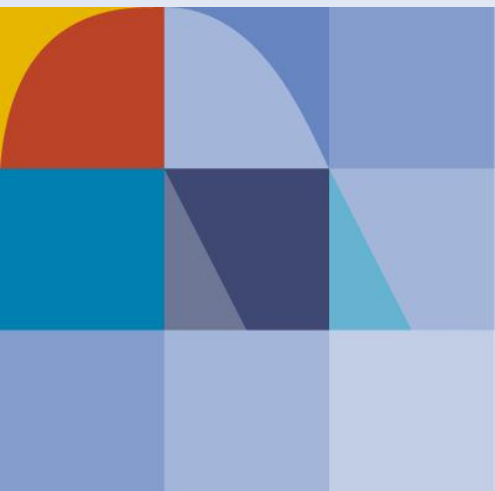
管理者振る舞いを中心に明記のない仕様についても考慮した

3. 成果物間のトレーサビリティをしっかりとる

全ての成果物の入出力関係を明確にした

4. 客観的にわかりやすい

成果物の入出力関係な明快なプロセスとした



NTT DATA

Global IT Innovator