

ソフトウェアテストの基礎

ソフトウェアテストの教科書―品質を決定づけるテスト工程の基本と実践

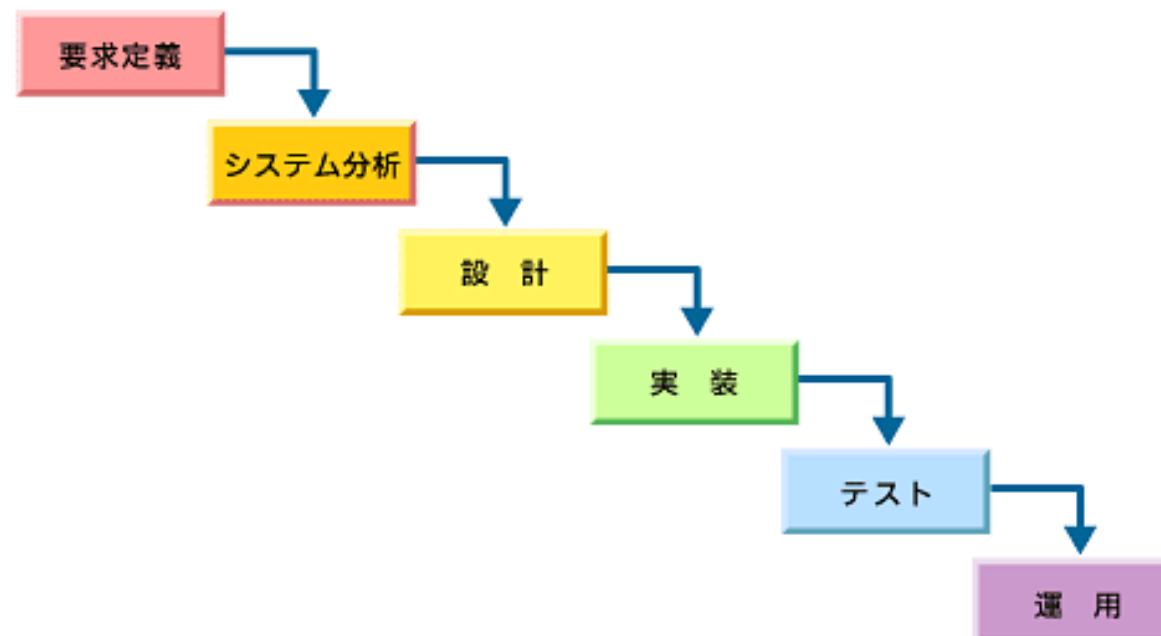
石原 一宏 (著), 田中 英和 (著), 田中 真史 (監修)

Part 1 (p1~p62)

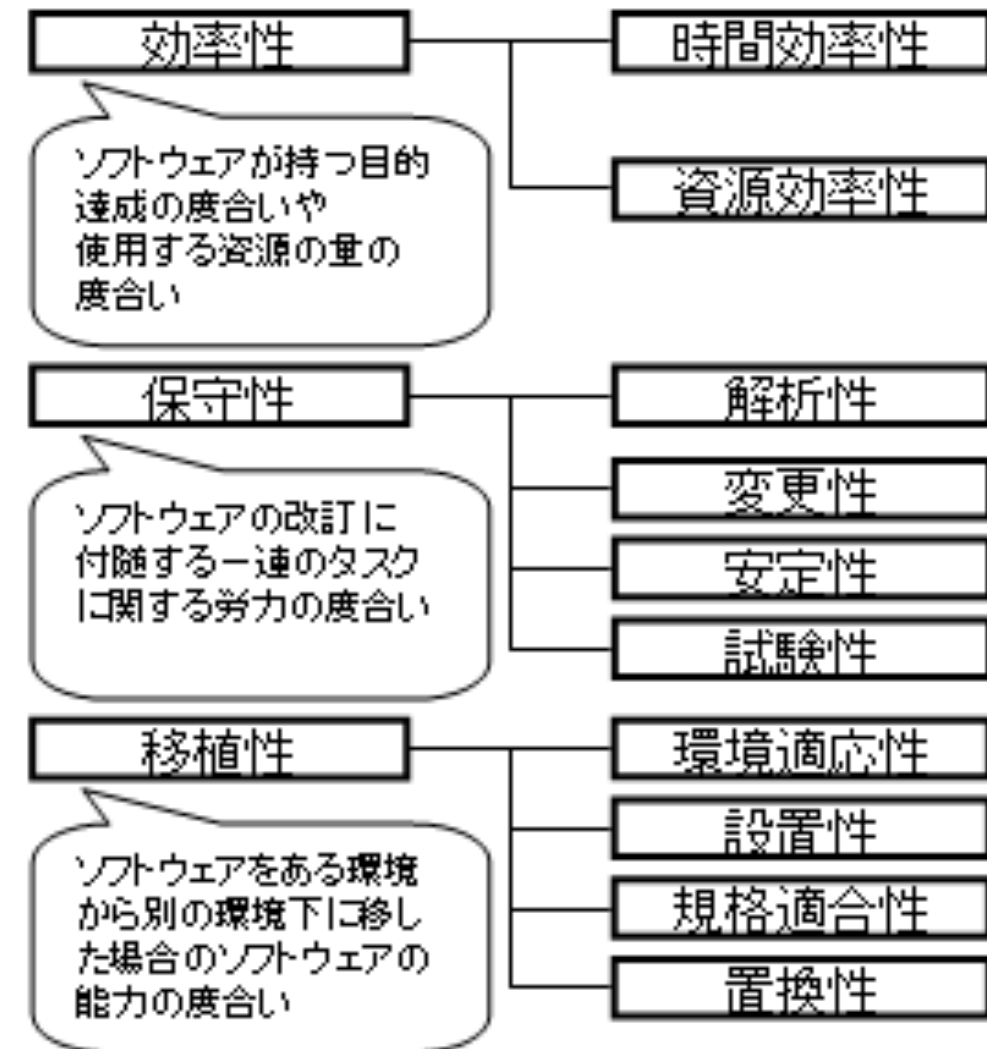
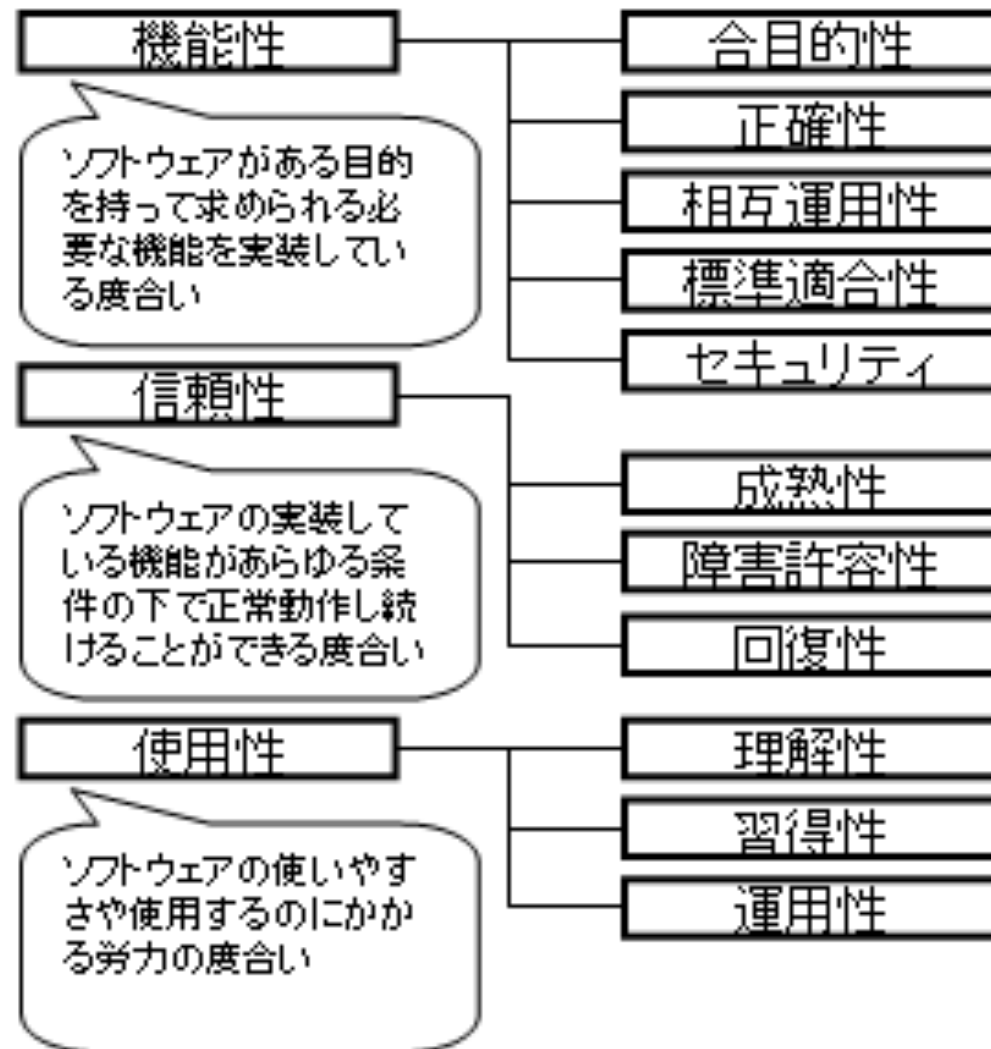
ソフトウェアテストとは

ソフトウェアテストとは、欠陥を取り除いて、ユーザーの要求を満たす、品質のよいソフトウェアを作ること

- ◎ バグ・欠陥を見つけ出す
- ◎ ソフトウェアの属性・品質を測定する
- ◎ ソフトウェアの品質を改善する



品質の特性(ISO/IEC 9126)



ソフトウェアの品質

◎ Verification and Validation（検証と妥当性の確認）

▶ Verification

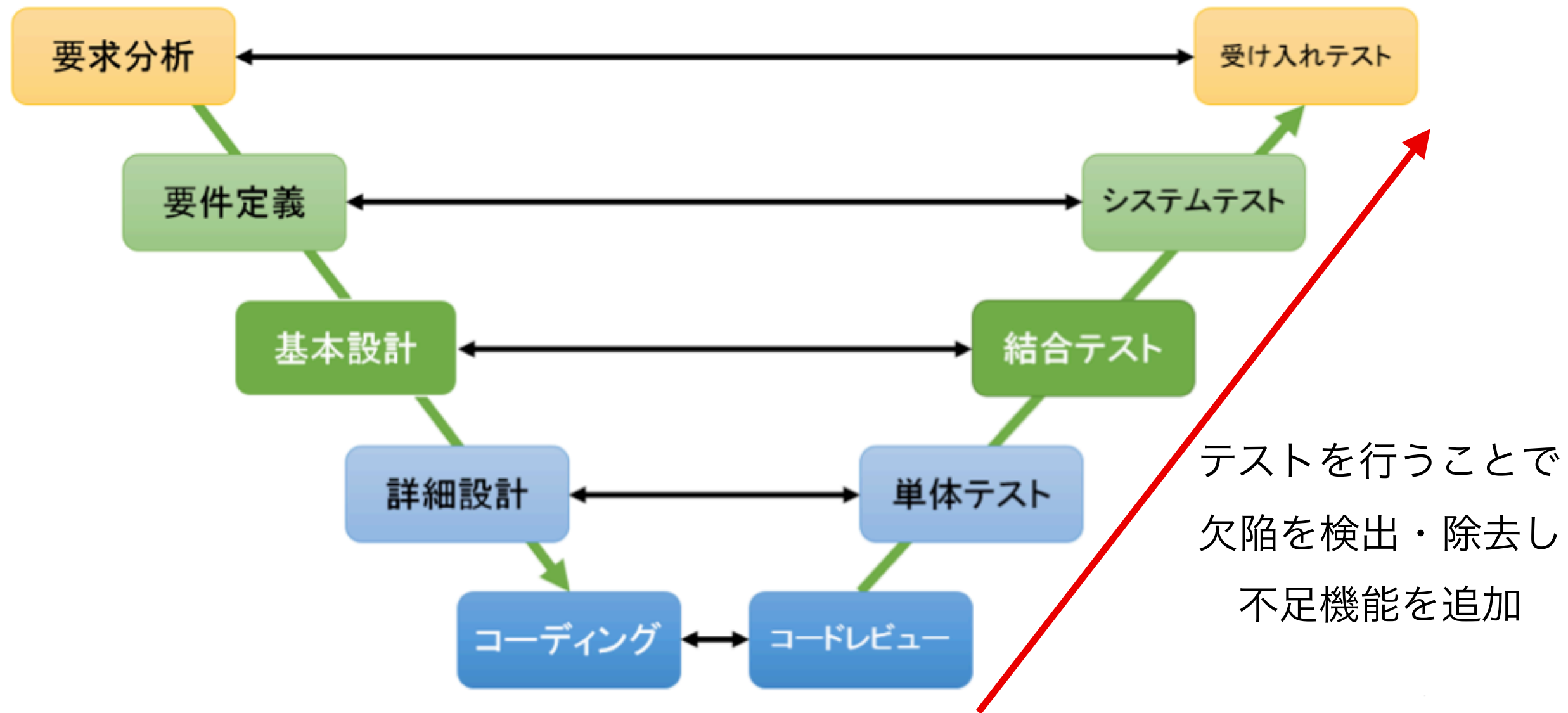
- 開発仕様書の通りソフトウェアが作成されているかどうか

▶ Validation

- ユーザーの要求通りにソフトウェアが動作するかどうか

テスト工程

- ◎ ウォータフォール型のテストでは下流から上流



コーディング完了時には、
いくつかの欠陥が含まれている

単体テスト

- ◎ ソフトウェアの最小単位モジュールごとにテストを行う
 - ▶ 詳細設計書通りに動くかどうか
 - ▶ ソフトウェアの論理構造が適切かどうか
 - ▶ 多くの場合、コーディングを行なったプログラマ自身が担当
- ◎ テスト内容
 - ▶ 機能確認テスト・制御フローテスト・データフローテスト

結合テスト・機能テスト

◎ 結合テスト

- ▶ 複数のモジュールを組み合わせてテストを行う
 - 詳細設計で定義したモジュール間の連携が取れているかどうか

◎ 機能テスト

- ▶ 組み合わせたモジュールを1つの機能としてテスト

◎ テスト内容

- ▶ 状態遷移テスト・機能確認テスト

システムテスト

- モジュールや機能を統合した状態で、要求定義の内容が実現しているかどうかのテスト
 - ▶ 出荷する状態・サービスとして提供する状態に基づいて行う
- テスト内容
 - ▶ 確認テスト・評価テスト・負荷テスト・環境テスト・機能テスト

代表的なテストの分類

静的テスト

動的テスト

ホワイトボックステスト グラックボックステスト

	静的テスト			動的テスト		
	要件定義	設計	実装	単体テスト	結合テスト	システムテスト
ホワイトボックステスト	インスペクション テクニカルレビュー	テクニカルレビュー 非公式レビュー	ウォークスルー 非公式のレビュー 机上デバッグ	機能確認テスト	状態遷移テスト 機能確認テスト	確認テスト 評価テスト 負荷テスト 環境テスト 機能確認テスト
				制御フローテスト データフローテスト		確認テスト 運用テスト アルファテスト ベータテスト
				機能テスト	テスト	テスト

ホワイトボックステスト

- ソフトウェアの最小単位であるモジュールの一つを対象とした単体テストで用いられる
 - ▶ 制御フローテスト
 - 命令文の処理順序を確認する
 - ▶ データフローテスト
 - データの流れを確認する

制御フローテスト

命令文・分岐した経路・条件に着目する

◎ ステートメントカバレッジ

- ▶ 全ての命令文を最低1度は通るようにテストする

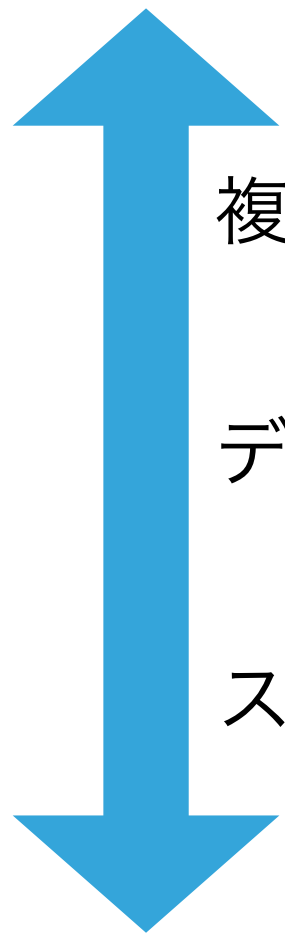
◎ デシジョンカバレッジ

- ▶ 判定が真・偽となる組み合わせを最低1回ずつ通るようにテスト

◎ 複合条件カバレッジ

- ▶ 命令文の全ての条件をテストする

テスト回数が多い



複合条件カバレッジ

デシジョンカバレッジ

ステートメントカバレッジ

テスト回数が少ない

データフローテスト

- ◎ ソフトウェアの中で使われているデータや変数が正しく処理されているか確認するテスト
 - ▶ 定義→使用→消滅の順に処理されているか確認
 - 未定義のまま使用されているデータ
 - 定義されて使用されていないデータ

ブラックボックステスト

◎ ソフトウェアの内部構造を意識しないテスト

- ▶ 仕様を満たしているかどうかのみをテストする
- ▶ 機能テスト・システムテストで用いられる

テスト計画・設計の流れ

1. テスト目的の確認
2. 機能一覧の確認
3. テスト観点の抽出
4. テスト観点と機能への割り当て
5. テスト技法の検討と適用
6. リソースの見積もり・スケジュールの作成