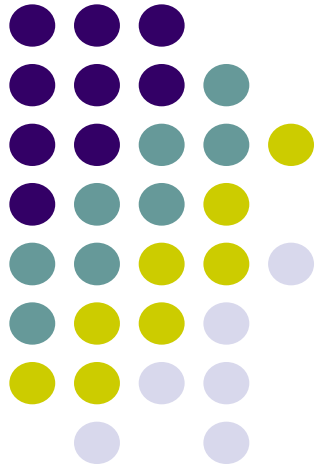
ゆもつよメソッドのテスト要求分析とテストアーキテクチャ設計

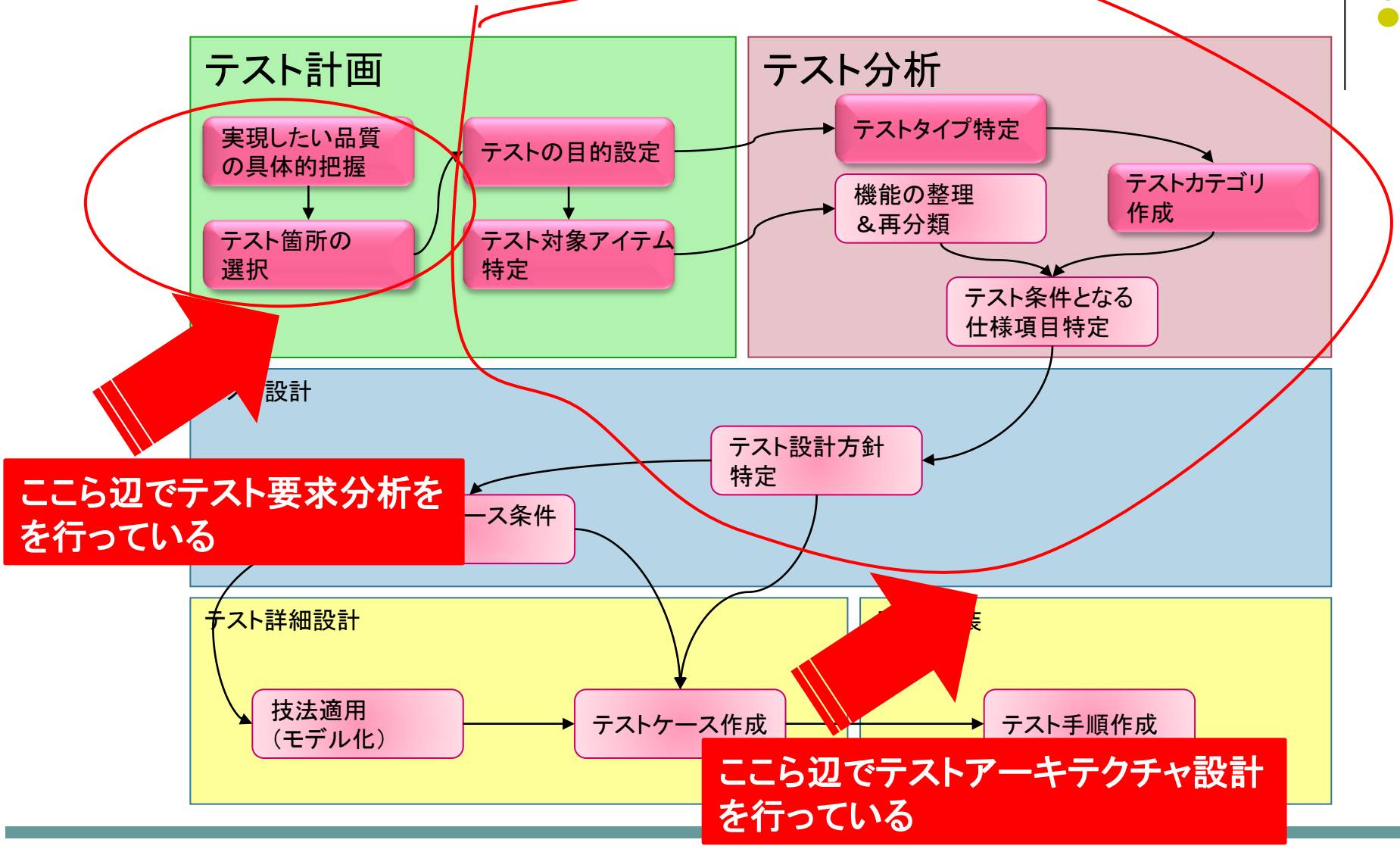
JaSST13東京　智美塾

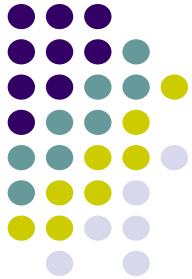
2013年1月30日



湯本剛(日本HP)
tsuyoshi.yumoto@hp.com

ゆもつよ風テスト開発プロセス

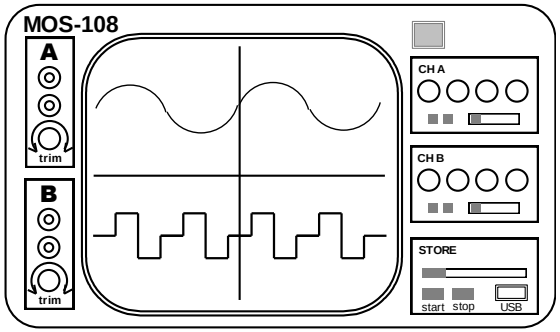




テスト要求分析

要求分析で行うこと(要求の源泉？？)

【前提1】 対象製品の機能の整理



主力製品であるオシロスコープの機能アップ

【前提2】 新機能と開発範囲

- 新機能
  - ①計測周期の引き上げ(最大10GHZまで)
  - ②計測精度の向上(最大32bitまで)
  - ③計測結果の保存(最大1分まで)
    - →データを転送して使う(オシロでは保存結果の再生はしない)
  - ④計測結果の外部転送(USB I/F追加)
  - ⑤外部トリガ対応(外部PCからの操作可能に)
  - ⑥チャンネル数が2つになる
    - ディスプレイ表示にて2チャンネルの同時表示
    - 保存と表示のマルチタスク
  - ⑦4グレード同一ソフト対応
- 開発範囲(ソフトウェア)
  - 新A/Dコンバータ用デバイスドライバ
  - 新保存部用デバイスドライバ
  - 新I/F用デバイスドライバ
  - ディスプレイドライバ
    - (既存の見直し)
  - オシロスコープ制御ソフト
    - (既存の見直し)
    - アーキテクチャ部分含む

--別で開発している想定にする------

- PC側ソフトウェア
- PC側I/F用デバイスドライバ

【前提3】 リスクとの関連

- 開発範囲(ソフトウェア)
  - 新A/Dコンバータ用デバイスドライバ
  - 新保存部用デバイスドライバ
  - 新I/F用デバイスドライバ
  - ディスプレイドライバ
    - (既存の見直し)
  - オシロスコープ制御ソフト
    - (既存の見直し)
    - アーキテクチャ部分含む

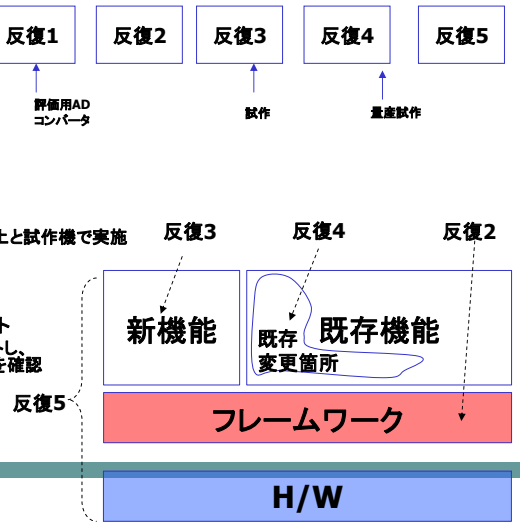
-別で開発している想定にする--

- PC側ソフトウェア
- PC側I/F用デバイスドライバ

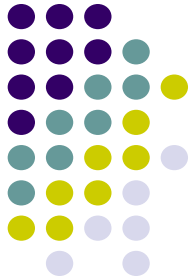
- 技術課題
  - ①A/Dコンバータ用のデバイスドライバにて計測周期と精度への対応
  - ②データトラフィックに対するメカニズム
  - ③チャンネル数が2つに対応するメカニズム
- リスク
  - ①新しいADコンバータ開発に時間がかかり、スケジュールを逼迫する恐れ

【前提4】 開発プロセス

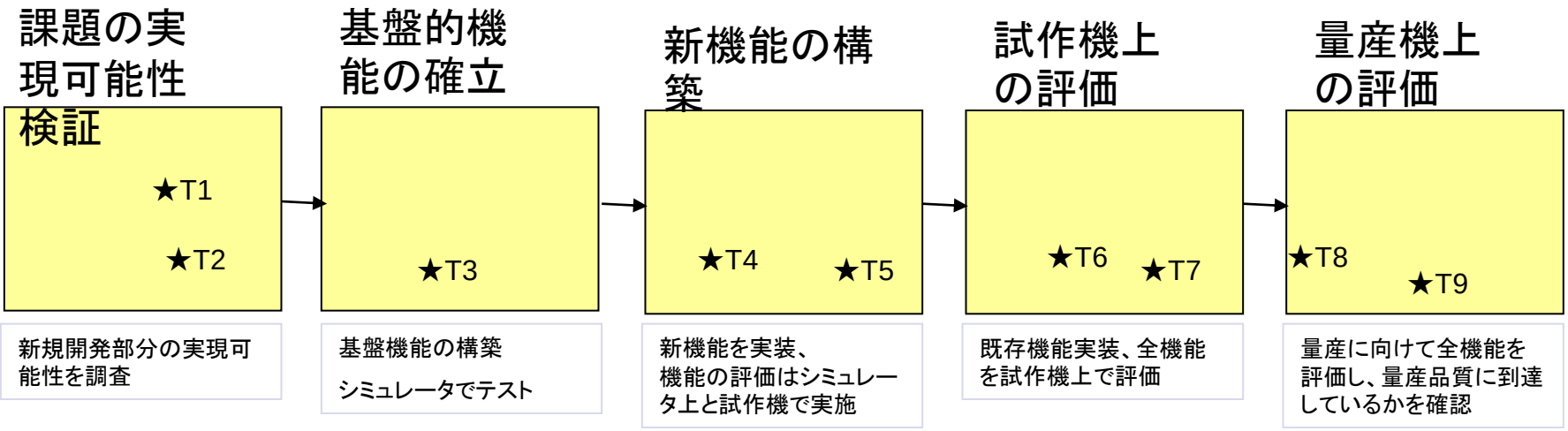
- 方向付け
  - 反復1
    - 技術課題に対する、実現可能性を調査
- 推敲
  - 反復2
    - フレームワークの構築
    - シミュレータでテスト
- 構築
  - 反復3、4(インクリメンタル)
    - 3:新機能を実装、機能のテストはシミュレータ上と試作機で実施
    - 4:既存機能実装、全機能を試作機上でテスト
- 移行
  - 反復5 量産試作上でのテスト
    - 量産に向けて全機能をテストし、量産品質に到達しているかを確認



テスト要求分析結果の構造



開発プロセス



テストの目的

T1 計測周期・精度の性能確保
T2 データ量に対するパフォーマンス評価

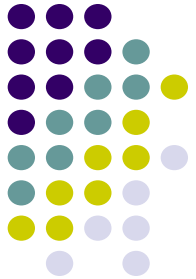
T3 アーキテクチャの機能性・信頼性確保

T4 新機能の機能性確保
T5 試作機上での動作・リリース判定

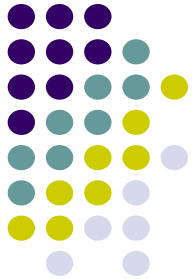
T6全機能の機能性確保
T7システムパフォーマンス評価・リリース判定

T8量産試作上での完成度
T9システムの信頼性確保・リリース判定

テスト要求分析のアウトプット



リリース	テストの目的		テスト対象項目	テストレベル	テストの種類
反復1： 課題検証	T1	・計測周期・精度の性能確認	・新A/Dコンバータ用デバイスドライバ（計測周期・精度の性能）	統合	・構造テスト ・機能テスト
	T2	・データ量に対するパフォーマンス確認	・新A/Dコンバータ用デバイスドライバ・新保存用デバイスドライバでのデータトラフィック処理	統合	・構造テスト ・パフォーマンステスト ・ボリュームテスト
反復2	T3	・アーキテクチャの機能性・信頼性確保	・データトラフィック制御 ・チャンネル制御 ・外部トリガ対応 ・外部転送制御 ・グレード毎の制御	単体 統合 システム	・構造テスト ・機能テスト ・ストレステスト ・ストレージテスト
反復3： 新機能	T4	・新機能の機能性確保 （シミュレータ上）	・新規機能	単体 統合	・構造テスト ・機能テスト
	T5	・試作機上での動作 ・外部PCとの連携・外部PC構成変更時の機能性確保 ・リリース判定	・表示部/操作部/計測部/保存部/I/F部との連携動作 ・USB経由での操作、データ転送 ・外部PCの様々な構成（OS、CPU、RAM容量など）上の動作	統合 システム	・機能テスト ・ボリュームテスト ・構成テスト
反復4： 既存	T6	・全機能の機能性確保	・機能	統合	・機能テスト
	T7	・システムパフォーマンス評価 ・リリース判定	・試作機環境での精度/データ量が多いときの応答時間	システム	・パフォーマンステスト ・ボリュームテスト
反復5： 量産試作	T8	・量産試作上での完成度	・機能 ・グレード毎のシステム構成上での動作、動作保証する転送先USB毎のデータ転送	システム	・機能テスト ・構成テスト
	T9	・システムの信頼性確保 ・リリース判定	実環境でのロバストネス、エラー発生時の復旧処理	システム	・障害対応性テスト ・ストレステスト ・ストレージテスト ・ボリュームテスト ・ロングランテスト



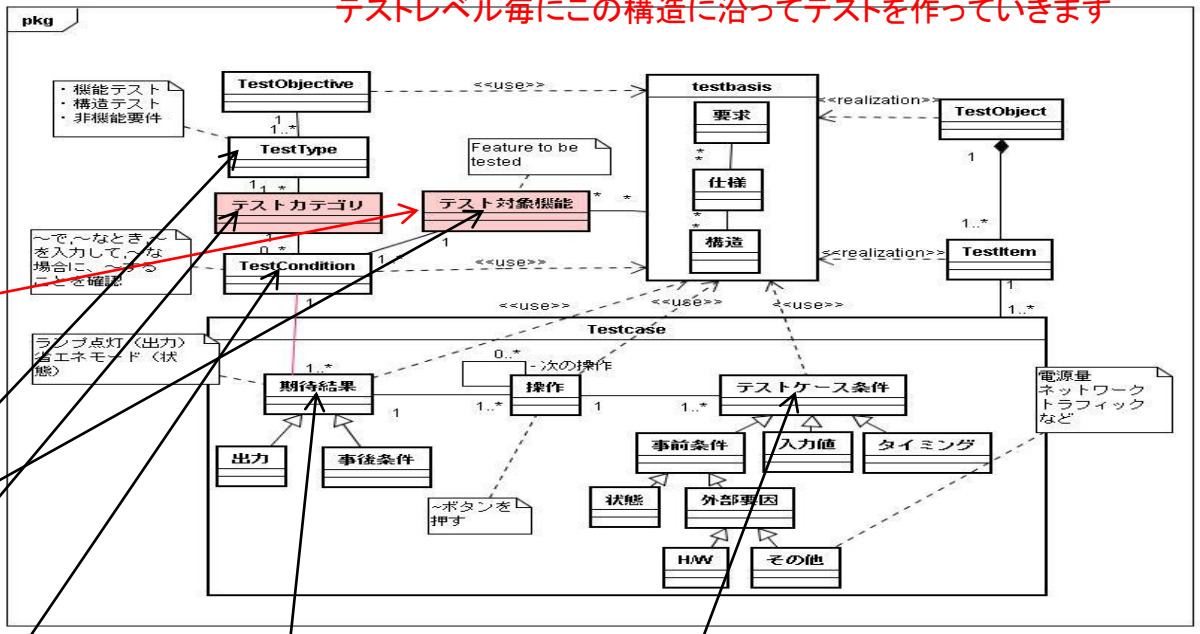
テストアーキテクチャ設計

テストアーキテクチャのメタ構造

計画時に作る一覧（ここでテストレベルは分離される）

	テストの目的	テスト対象アイテム	テストレベル	テストタイプ	役割
MT1	・自身で書いたコードが想定どおり動くことを確認	モジュール	単体テスト	機能テスト 構造テスト	プログラマ
MT2	・既存モジュール含め、コンポーネント単位でビルドし機能が正しく動くことを確認	コンポーネント	コンポーネントテスト	機能テスト	開発チーム
MT3	・コンポーネント間シーケンスの確認 ・S/Wシステムテストへのリリース判定	複数ブロック （評価用試作機）	コンポーネント統合テスト	機能テスト 構造テスト	システム 検証チーム
MT4	・ソフトウェア全体の機能要件・非機能要件の確認 ・システムテストへのリリース判定	S/Wシステム （量産試作機）	S/W		
MT5	・H/W含めた製品としての機能要件・非機能要件の確認 ・出荷判定	システム	システ		

テストレベル毎にこの構造に沿ってテストを作っていきます

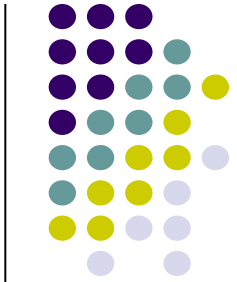


テスト対象機能の一覧が
ゆもつよで作る機能一覧です

下記一覧とクラス図の関係

機能名／機能小分類	テストタイプ	テストカテゴリ	テスト条件とすべき仕様項目	期待する結果／確認内容	テスト設計方針	主なテストケース条件

テストアーキテクチャ設計のアウトプット



テストタイプ、テストカテゴリ		仕様項目（テスト条件）	テスト設計方針	主なテストケース条件
機能テスト	入力	・設定のチェック（自動ON/OFF,更新頻度設定） ・インストーラ	・チェックON/OFF、ラジオボタンチェックの設定順序組み合わせ ・入力文字数の境界値	チェックボックス、ラジオボタン、入力欄
	画面表示	・FWアップデート画面4画面、アプリケーション2画面、インストーラ	画面表示内容を1度確認	
	ボタンと画面遷移	FWアップデート、アプリアップデート、インストーラ	画面遷移の状態遷移テスト　SOカバレッジ	画面、ボタン（更新確認ボタン）
	設定	・自動更新設定（自動/手動） ・自動更新頻度（3パターン）	・デシジョンテーブルで確認 ・更新タイミングの境界値テスト	チェックボックス、ラジオボタン
	処理組み合わせ	・自動更新設定の切り替えパターン（切替前4×切替後4=16パターン） ・処理結果組合せ（FW→アプリの際のUpdate有り無し）4パターン ・上記処理結果組合せパターンの組み合わせ（4×4＝16パターン）	・組合せ/状態遷移テスト＆AllPair ・組合せ/デシジョンテーブル＆実施順の組合せ ・全組合せ/前の組合せの同値分割（あり、なしの2パターンにする）	状態=設定と頻度、イベント=切り替え、アクション

たぶんここがメカニズムを考えている部分になると思うのだが、とっても体系的でない原始的なやりかた