

## JEITA 組込み系開発スピードアップ・ワークショップ2008

---

# 組込み製品のテスト効率化・合理化への取組み

2008年08月27日

日立製作所 組込みシステム基盤研究所  
主任研究員 島袋 潤

# 1 背景 ～ “組込み系”の多様性と難しさ

---

## 1.1 我々のミッション：組込みソフト開発の課題の解決

### 直面する問題

増大する開発量・  
短縮する開発期間

複雑・曖昧な  
要求や仕様

多彩な実行  
プラットフォーム

### 解決の方向性

開発プロセスやプロジェクト  
マネージメントの改革

開発手法の改善・  
開発環境の向上

ソフトウェアの資産化・  
再利用

**ソフトウェア開発視点での解決**

(プロセス改善, 開発環境改善,  
品質マネージメント, 教育など)

**ソフトウェア要素視点での解決**

(OSやミドルウェアの開発・利用,  
ソフト設計技法, 再利用技法)

『ソフトウェア工学の成果をソフトウェア開発の現場へ』

# 1.2 日立グループにおける“組込み系”とは？

## 幅広い事業フィールドと製品

製品をイメージしにくい分野

部品 (自動車)	交通・情報 システム	コンサル ティング
-------------	---------------	--------------

新幹線

医療機器

ストレージ

発電設備

テレビ

携帯電話

BDカム

洗濯機

製品をイメージしやすい分野

“エンプラ系システム”と  
その基盤ソフト以外は、  
“組込み系システム”？



“組込み”という用語  
自体が期限切れかも

## 1.3 例えば, 「情報系組込みシステム」の例をとってみても...

	車載端末	A/V機器	携帯電話	医療機器
販売方法	<ul style="list-style-type: none"> <li>自動車メーカーOEM</li> <li>量販店販売</li> </ul>	<ul style="list-style-type: none"> <li>量販店販売</li> </ul>	<ul style="list-style-type: none"> <li>携帯電話キャリアを介した量販店販売</li> </ul>	<ul style="list-style-type: none"> <li>医療機器販売会社OEM</li> <li>特定顧客納入</li> </ul>
メーカーの直接顧客	<ul style="list-style-type: none"> <li>自動車メーカー</li> <li>不特定ユーザ</li> </ul>	<ul style="list-style-type: none"> <li>量販店</li> <li>不特定ユーザ</li> </ul>	<ul style="list-style-type: none"> <li>携帯電話キャリア</li> </ul>	<ul style="list-style-type: none"> <li>販売会社</li> <li>特定顧客</li> </ul>
最終ユーザ	<ul style="list-style-type: none"> <li>不特定ユーザ</li> </ul>	<ul style="list-style-type: none"> <li>不特定ユーザ</li> </ul>	<ul style="list-style-type: none"> <li>不特定ユーザ</li> </ul>	<ul style="list-style-type: none"> <li>特定ユーザ(医師, 検査技師など)</li> </ul>
要求の発生元	<ul style="list-style-type: none"> <li>カーメーカー要求あり</li> <li>ユーザは将来何をしたいかを知らない</li> </ul>	<ul style="list-style-type: none"> <li>誰も何をしたいか不明「サイレント・カスタマ」「クチコミ」</li> </ul>	<ul style="list-style-type: none"> <li>キャリアの要求あり</li> <li>ユーザは将来何をしたいかを知らない</li> </ul>	<ul style="list-style-type: none"> <li>ユーザが運用プロセスを知っている.</li> <li>厳しい法規制</li> </ul>
開発の特徴	<ul style="list-style-type: none"> <li>数十種類のバリエーションを並行開発</li> <li>バリエーション毎のHMI</li> </ul>	<ul style="list-style-type: none"> <li>年に2~3回の商機</li> <li>開発期間半年~1年</li> <li>複数の製品計画とソフト開発が並行</li> </ul>	<ul style="list-style-type: none"> <li>年に2~4回の商機</li> <li>共通プラットフォームによる統一</li> <li>機能優先度が不明</li> </ul>	<ul style="list-style-type: none"> <li>比較的長い開発期間</li> </ul>
要求の特徴	<ul style="list-style-type: none"> <li>車搭載時の適合性</li> <li>新機能性</li> <li>特定局面の使い勝手</li> </ul>	<ul style="list-style-type: none"> <li>不特定多数の使い勝手</li> <li>感性面での品質</li> <li>新機能性</li> </ul>	<ul style="list-style-type: none"> <li>ブランド性</li> <li>新機能性</li> <li>ターゲットユーザの使い勝手</li> </ul>	<ul style="list-style-type: none"> <li>正確性</li> <li>作業効率性</li> <li>特定ユーザの使い勝手</li> </ul>
ソフト開発の主な関心	<ul style="list-style-type: none"> <li>カスタマイズ性</li> </ul>	<ul style="list-style-type: none"> <li>開発期間短縮</li> <li>出荷後不具合削減</li> </ul>	<ul style="list-style-type: none"> <li>開発期間短縮</li> <li>出荷後不具合削減</li> </ul>	<ul style="list-style-type: none"> <li>超高信頼性</li> </ul>

## 1.4 改めて、「組み込み系ソフト」開発の課題とは？

- そもそも『組み込みソフト』のくりに意味があるのか？
  - あまりに多種多様な製品
  - あまりに多種多様なビジネス形態
  - あまりに多種多様な開発プロセス
  - あまりに多種多様なステークホルダー
  - あまりに多種多様な課題
  - あまりに多種多様な・・・
- 『組み込みソフトの課題』とされてきたことが本当に問題か？
  - 問題になることが多いのは事実
  - しかし、問題にならない場合も多い

- 共通的な課題を解決すれば良いのか？

- 共通課題の解決そのものは重要
- しかし、固有の課題こそが解決したい難しい課題
  - 固有課題であるがゆえに、解決しても他のドメインに展開が難しい
  - 固有課題ほど解決にコストがかかるが、固有課題ほどコストがかけづらい

「テスト・検証技術」はその代表例

1. 背景：“組込み”の多様性

2. テストプロセス課題と改善

3. 日立での取組み事例

4. テスト技術の最新動向

5. まとめ

# 2 テストプロセスの課題と、改善への提案

## 2.1 「テスト・検証プロセス」の実情

テスト・検証に関して実施すべき活動内容は、共通に理解されているか？

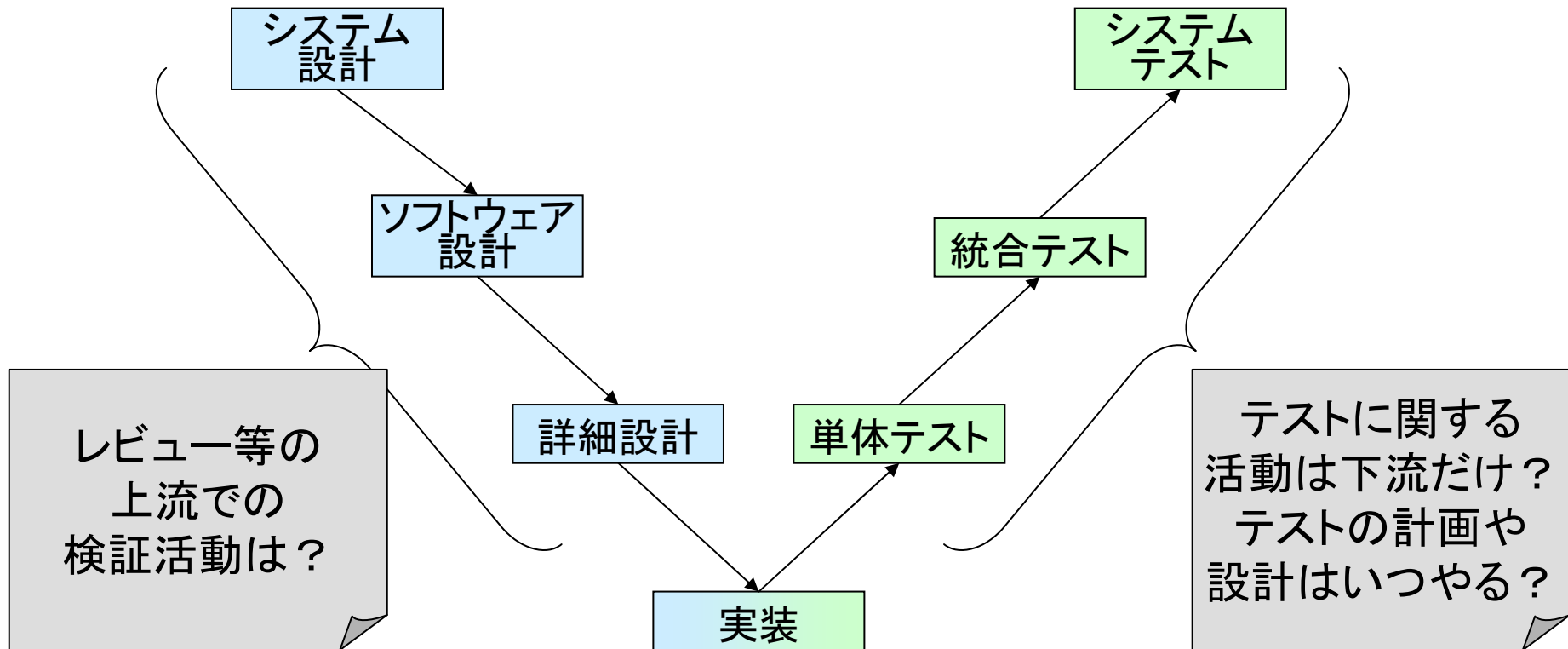
残念ながら、日立グループ内では“No”です。

・ 事業部／会社の成り立ち、対象製品毎によって、様々な観点があるようです。

	回答の視点	例
1	テスト対象の粒度	関数単位→コンポーネント単位→
2	テスト技法	ブラックボックス→ホワイトボックス
3	テスト環境・道具	非実機環境→実機環境→運用環境→
4	テスト実施組織	ソフト開発部署→品質保証部署→
5	テスト作業手順	テスト計画→テスト項目設計→テスト実施
6	管理対象ドキュメント	テスト仕様書→テスト手順書→
7	管理ツール	バグトラッキング→進捗管理→
8	テスト分析手法	限界値・境界値テスト→パステスト→
...	...	...

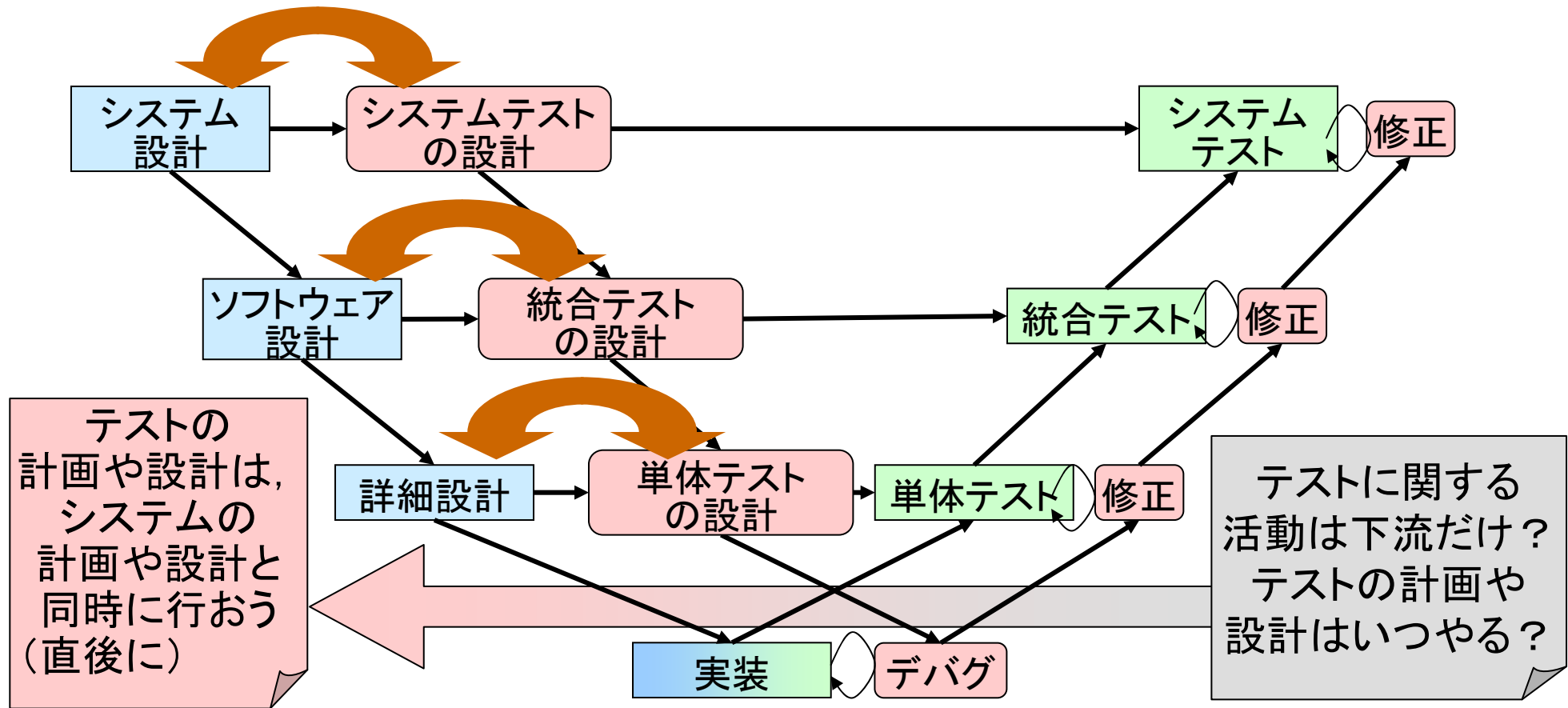


## 2.2 従来のプロセスモデル：V字モデル



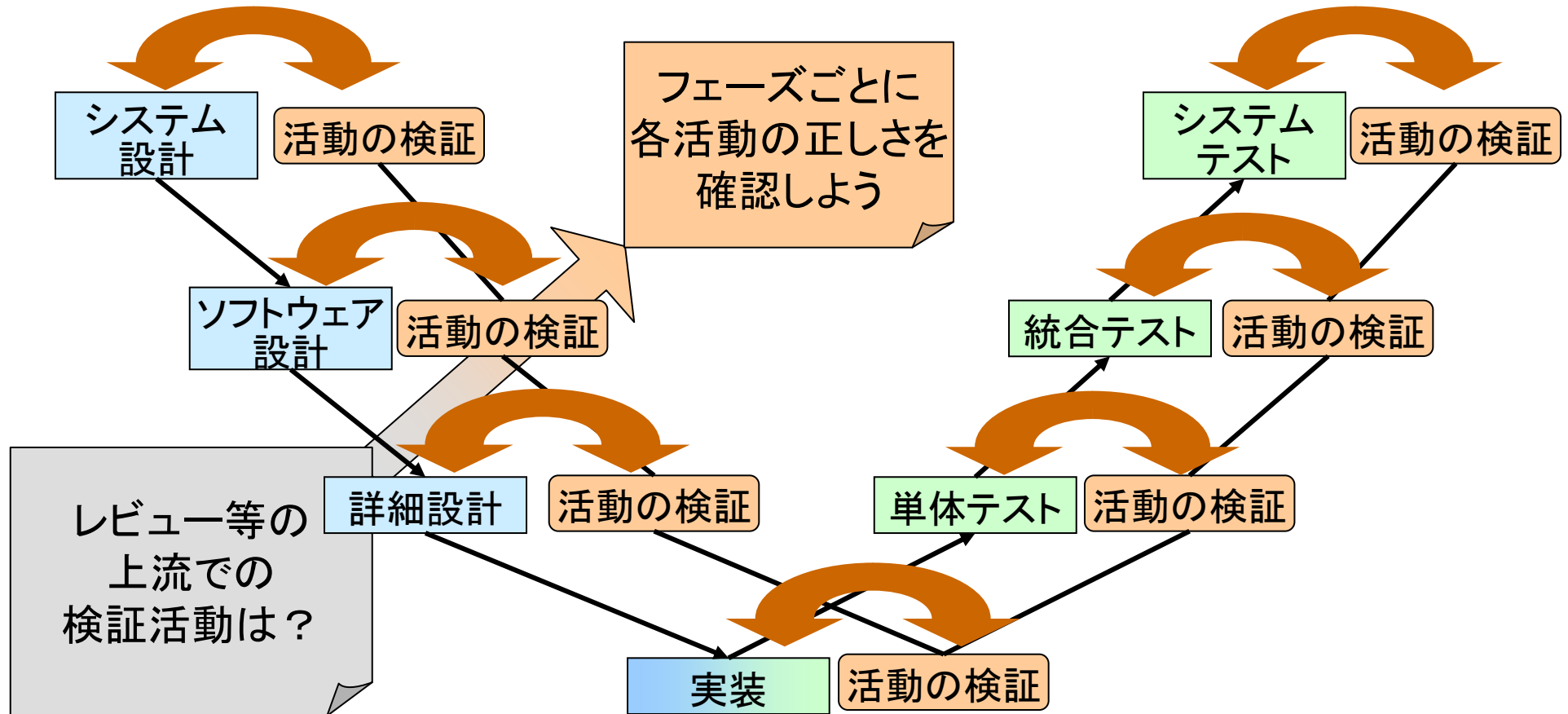
## 2.3 近年のプロセスモデル：W字プロセス（タイプI）

「テスト設計の上流化」を指向



## 2.4 近年のプロセスモデル：W字プロセス（タイプⅡ）

「開発プロセスの各活動の検証」を指向



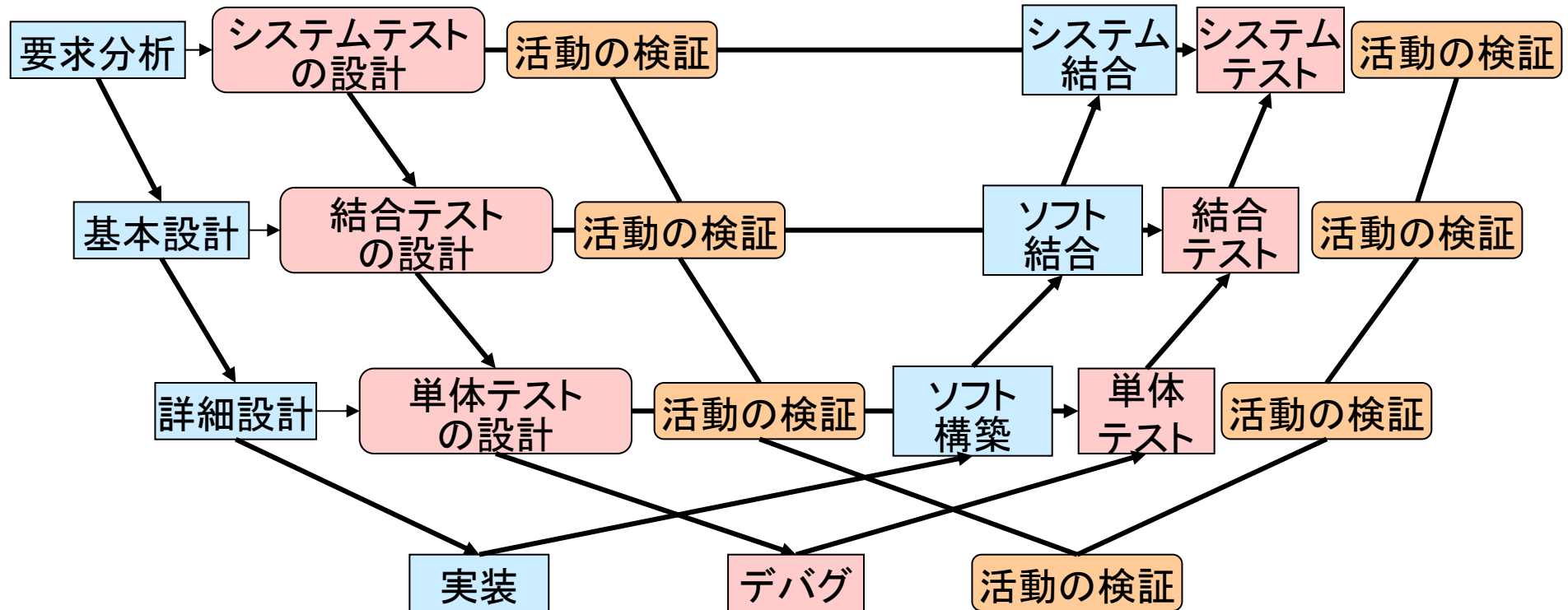
## 2.5 提案モデル(その1):V3モデル

### 設計構築・テスト・活動検証の各プロセスを融合

(V1) 設計構築プロセス

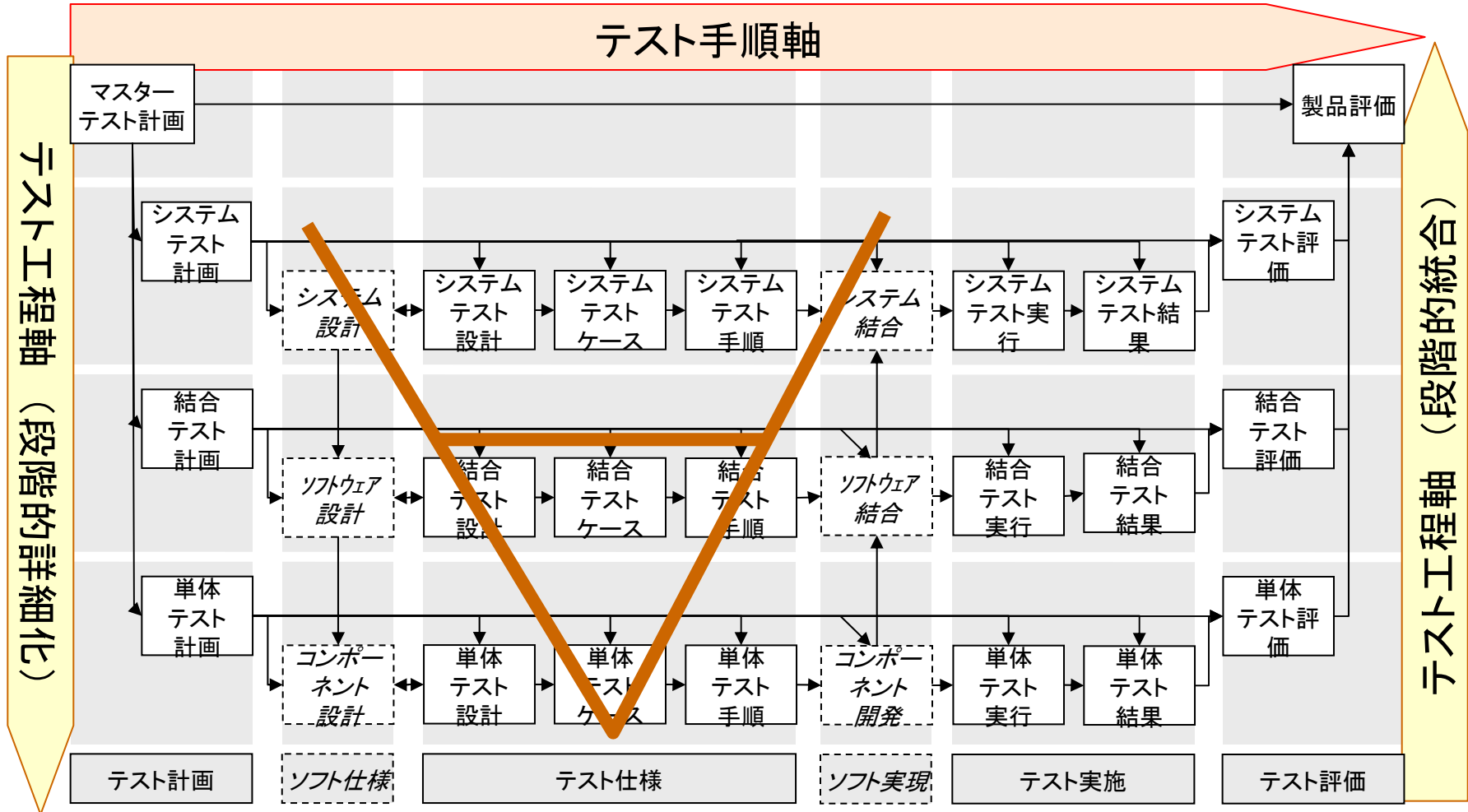
(V2) テストプロセス

(V3) 検証プロセス



## 2.6 提案モデル(その2)：Vモデル

従来のテスト工程軸に加え，テスト手順軸が重要



## 2.7 テスト指向開発プロセス

### テストの改善が、全ての改善への道

#### ■ IEEE829 “Standard for Software Test Documentation”

- |                |             |
|----------------|-------------|
| »テスト計画ID       | »テストの作業     |
| »テスト項目         | »環境         |
| »テスト対象の機能      | »責任者        |
| »テスト対象としない機能   | »人員とトレーニング  |
| »手法            | »スケジュール     |
| »合否基準          | »リスクとアクシデント |
| »テストの中断基準と再開要件 | »承認         |
| »成果物           |             |

#### ■ なんと呼んで戴いても良いのですが…

- テスト指向開発
- テスト指向アーキテクチャ
- テストファースト
- テストプロセス改善
- …

} 心は同じだと思います

# 3

## 日立グループでの取組み事例 ～ Pair-Wise法の活用

---

## 3.1 デジタル家電分野の特徴と課題

- 製品仕様の組合せ爆発
  - 製品採用規格の増加（データ形式、メディア種別等）
  - 入力イベントの増加（ユーザ操作、連動機器、ネットワーク等）
  - 同時動作の増加（自動起動機能、バックグラウンド動作等）
  
- 組合せ条件のテスト漏れが多い



## 3.2 組合せ条件テスト技法の適用

- システムテスト項目の設計に、Pair-Wise法を適用
  - N個のテスト条件について、値の組合せを網羅的・効果的に抽出

テスト条件				条件	A	B	C	条件	A	B	C
条件	A	B	C	#1	a1	b1	c1	#7	a2	b2	c1
テスト 値	a1	b1	c1	#2	a1	b1	c2	#8	a2	b2	c2
	a2	b2	c2	#3	a1	b2	c1	#9	a3	b1	c1
	a3			#4	a1	b2	c2	#10	a3	b1	c2
				#5	a2	b1	c1	#11	a3	b2	c1
				#6	a2	b1	c2	#12	a3	b2	c2

全組合せ  
 $3 \times 2 \times 2 = 12$ 通り



Pair-Wise法  
 N=2のとき: 6通りに削減可能

- N=2のときを、特にAll-Pair法とよぶ

- 限られたリソース(工数, 期間, 機材, ...)の中で, 組合せ条件を無駄なくテスト
- 優れたテストの“重点テスト”と併用し, 絨毯爆撃／ピンポイントの両面で攻める

### 3.3 他の組合せ条件テスト技法との比較

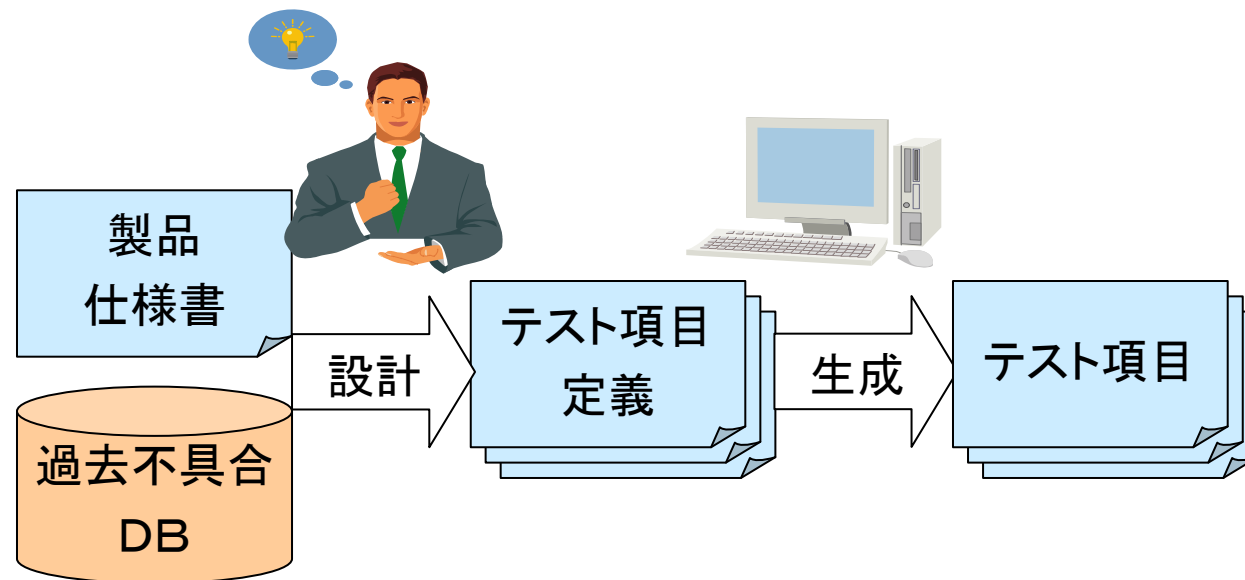
	直交表	Pair-Wise法
テスト条件 (因子)	条件・条件値の大きさに応じて表を選択し、条件・条件値を割り付け	可変
テスト条件値 (水準値)		可変
条件値組合せ の除外	対応方法有り テクニック必要	対応方法有り 容易
直交性	有り 但し、条件値の除外を行うと直交性が無くなる	無し
網羅性	2因子間網羅 (最近ではN因子網羅に拡張してきた)	任意の因子間で網羅

技法としては、「優劣の問題ではなく目的・状況によつての使い分け」であるが...  
**Pair-Wise法のほうが(厳密ではないが)導入ハードルが低いと考える**

【参考】秋山, 組合せテストの設計, 情報処理, Vol.49, No.2, pp.140-146(2008).

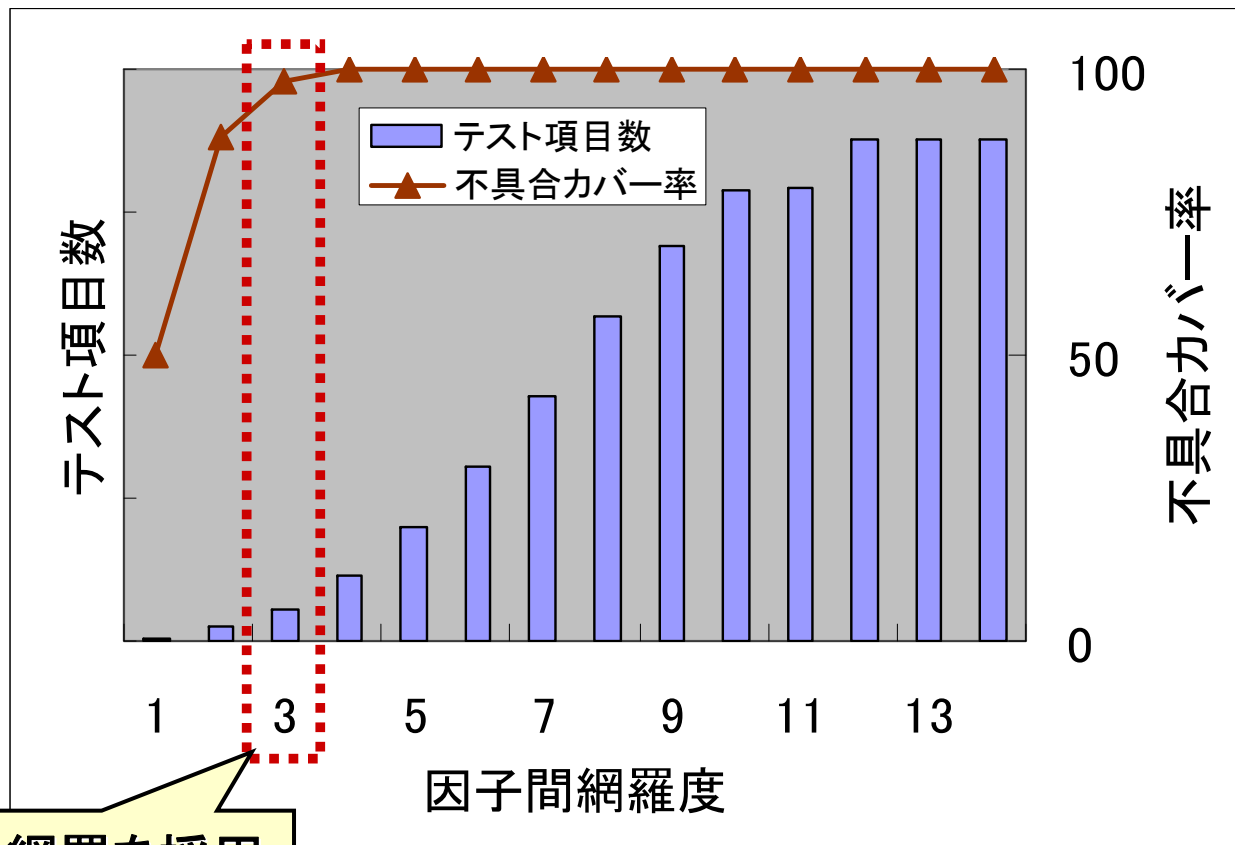
## 3.4 製品開発での運用

- 製品仕様で決めた動作仕様から、テスト条件(因子)を抽出
- 特に、過去の不具合データベースから、不具合になりがちな条件・条件の組合せを抽出してテスト項目を設計



## 3.5 因子網羅度の決定

- 設計したテスト項目を任意の網羅度で生成し，過去の不具合事例のどのぐらいをカバーできているかを確認



3因子網羅を採用

## 3.6 テスト技法としての、今後の課題

- テスト項目洗い出し方法のノウハウ蓄積
- 効率的なテスト消化方法
- 探索型テストの確立
  
- 他製品分野への展開
  - デジタル家電以外の数分野で、適用を試行中

1. 背景：“組み”の多様性

2. テストプロセス課題と改善

3. 日立での取組み事例

4. テスト技術の最新動向

5. まとめ

# 4

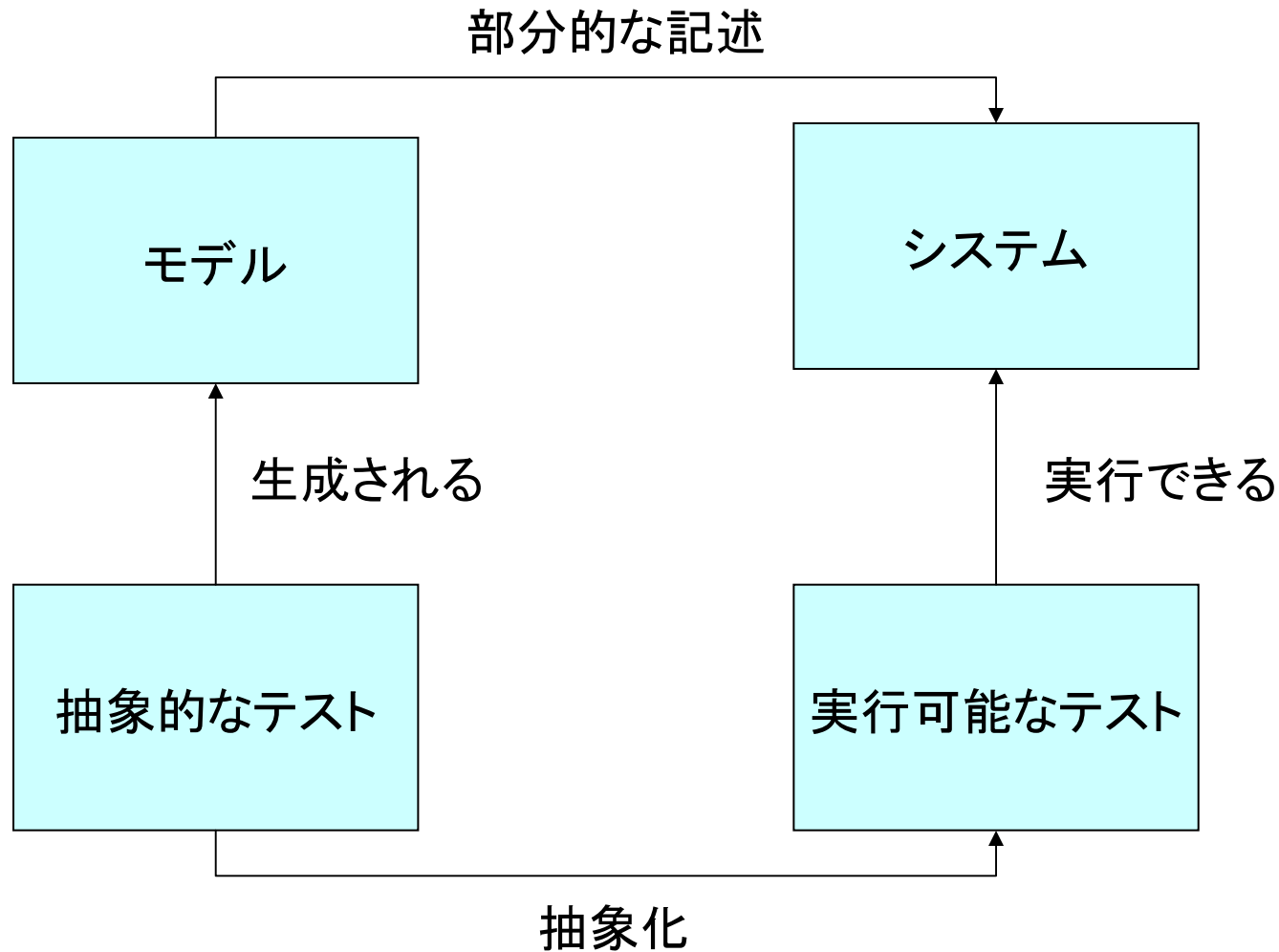
## テスト技術の最新動向

---

## 4.1 「モデルベースドテスト」のご紹介

- Model-based testing is
  - Software testing in which test cases are derived in whole or in part from a model that describes some (usually functional) aspects of system under test (SUT) [Wikipedia]
- モデルベースドテストとは、
  - テスト対象システム(SUT)の性質を記述したモデルから
  - 全部または一部のテストケースが生成される
  - ソフトウェアテスト
- 例えば, テストモデリング言語
  - TTCN-3
    - “Testing and Test Control Notation Version 3”
    - TTCN-1以来, 通信系を中心に, 標準化や産業系での利用実績
  - UT2P
    - “UML 2.0 Testing Profile”
    - UML 2.0に, テストの概念を導入

## 4.2 「モデルベーステスト」の構成要素(1)

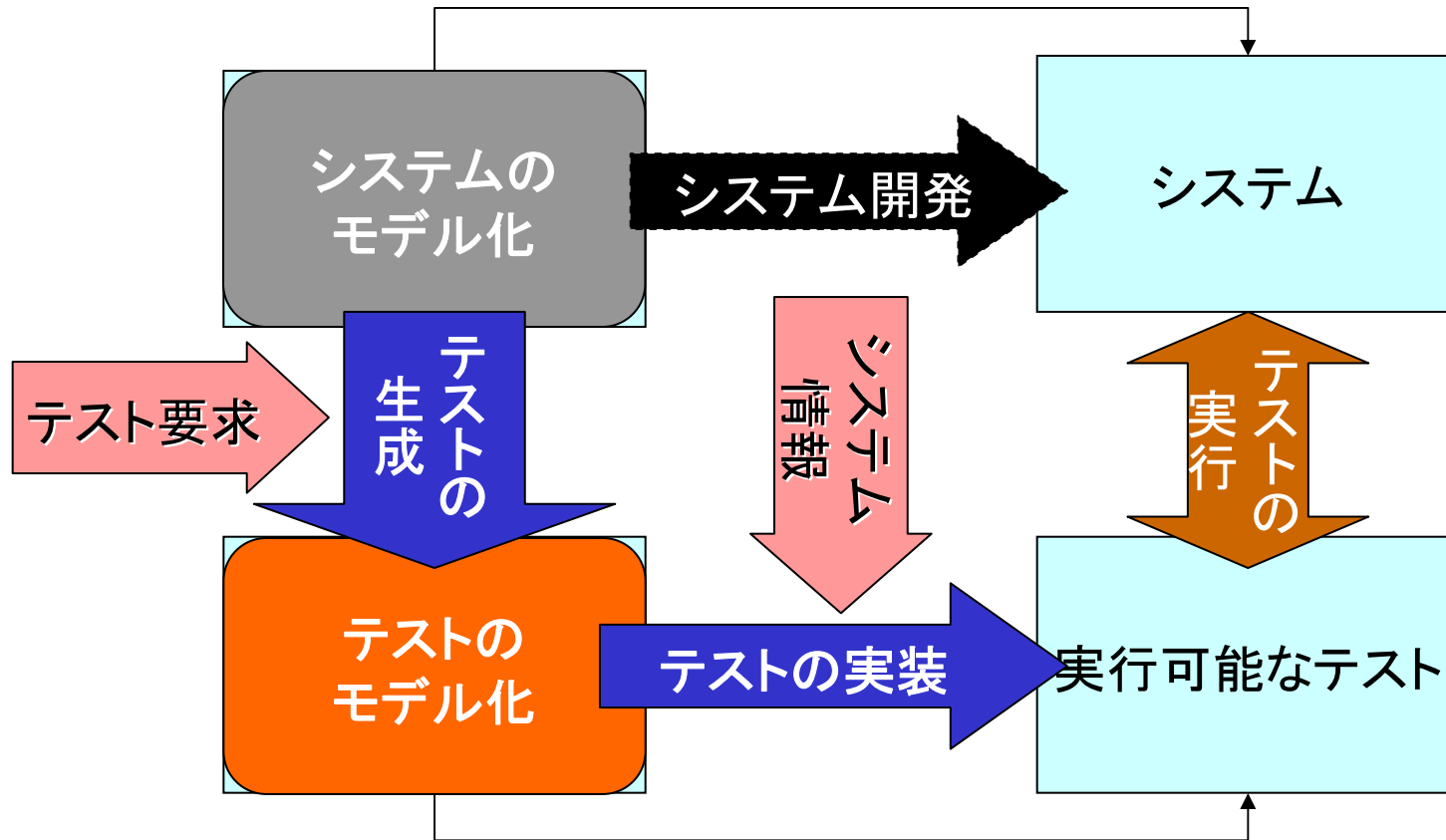


[Wikipedia]より和訳



# 4. 3 「モデルベーステスト」の構成要素(2)

## モデルベース開発



- ①テストのモデル化
- ②システムモデルからのテスト生成・実装

## モデルベーステスト

[Wikipedia]に加筆

## 4.4 詳しくは...

- “特集 ソフトウェアテストの最新動向”，情報処理, Vol.49, No.2(2008)
  - 西・古川, ソフトウェアテスト総論, pp.127-132.
  - 小野塚, UML2.0 Testing Profile/Eclipse Test & Performance Tools Platformの解説, pp.147-153.
  
- ソフトウェアテストシンポジウム(JaSST)2007 in Tokyo  
<http://jasst.jp/archives/jasst07e.html#research>
  - JaSST東京実行委員会, モデルベースドテスト入門
  - 小川, モデルベースドテストの技術動向と研究事例

1. 背景：“組込み”の多様性

2. テストプロセス課題と改善

3. 日立での取組み事例

4. テスト技術の最新動向

5. まとめ

# 5

## まとめ ～ 組込み開発のスピードアップに向けて、テストは...

---

## 5.1 テスト・検証プロセスの方向性

### ポイント：2つの上流化

#### 1. 検証の上流化

- 設計の上流で検証する
  - 検証可能な設計
  - 成果物の検証
  - 設計プロセスの評価

#### 2. テストの上流化

- テストの上流プロセスを整備する
  - テストの計画
  - 設計上流でのテスト設計

## 5.2 ソフトウェア技術者の間では...

- 「モデルベース開発」が注目されています
  - 仕様をきちんと定義すること
  - 仕様のレビューをしっかりとすること(できれば実行して...)
    - ▶ 「組込み」の世界では, “非実機”/“仕様検証”とも言う(?)
- 「アジャイル型開発プロセス」が注目されています
  - Water Fall型プロセスでは, 手戻りが大きすぎること
    - ▶ 「組込み」の世界では, そんなことは始めからやってなかったかも...
  - 顧客(あるいはステークホルダ)の確認を, 随所で取ること
- 「ソフトウェア・プロダクトライン」が注目されています
  - できるだけソフトウェアを作らないこと
  - 再利用によって, 短期開発・品質確保の両面を達成すること
  - そのためには, ソフトウェアだけでは解決できないこと

すべて, 「テスト・検証」から道は開けると思います

## 5. 3 JEITA(および皆様)への期待

- 「(組込み)ソフト」だけではなく、「システム」としての議論
  - “システム＝ハード(メカ, エレキ)＋ソフト”の単純な構図ではない
    - ▶ 例えば, システムレベルの機能記述に, LSIの動作仕様は詳細すぎる
    - ▶ 例えば, 部品の購入も, サブシステム単位になっている
    - ▶ 例えば, ソフトのオフショア開発では, ハード部品の輸出・調達が障害
    - ▶ 例えば, …
- 最後に一言…
  - ソフト技術者へ : 被害者意識はもうやめましょう
  - ハード技術者へ : でも, ソフトの都合もわかってください
  - 経営責任者へ : あいだを取り持ってください

“全員参加型100人ワークショップ”での活発な議論をお願いします

## ワークショップご参加のみなさま

---

**ご清聴ありがとうございました。  
日本の「組込み」の競争力強化の一助となれば幸いです。**

2008年08月27日

日立製作所 組込みシステム基盤研究所  
主任研究員 島袋 潤