Empowered by Innovation NEC

JaSST '11 hokkaido 基調講演

「テスト開発レトロスペクティブズ」
ーみなさん、そのテスト楽になってますか？ー

日本電気株式会社　品質推進本部
吉澤智美

今日お話すること

- 私自身のふりかえり
- ここ10年の振り返り
- 最近のテスト
- おわりに

自己紹介

＜略歴＞

- 日本電気株式会社 入社、マイクロプロセッサ開発環境の開発に従事
- IPA/SECへ出向、組込みソフトウェア開発向けガイドなどの執筆・編纂
- 帰社後は全社スタッフとして、標準化、品質保証活動に従事

＜社(内)外活動＞

- ソフトウェアテスト、品質に関する社内外の活動に参画
- NPO法人ソフトウェアテスト技術振興協会(ASTER)理事
- ISO/IEC-JTC1/SC7/WG26(ソフトウェアテスト)国内委員会技術委員
- JSTQB技術委員

私自身のふりかえり

- 自己紹介
- 業務を通して得られたもの
- 社外活動で得られたもの

業務を通して得られたもの

開発環境開発

- ドキュメントの作り方、開発者/サポートとのI/F、不具合票の適切な書き方
- エミュレータ、デバッガの仕組み
- GUIの使い勝手

開発環境統合化テスト、マネージャ

- 自動化、統合化、統合テスト
- チーム編成
- 構成管理、インストールテスト

IPA研究員、本社スタッフ

- 各社、各部門の状況、今までの整理
- ソフトウェアエンジニアリング、品質保証活動

業務を通して得られたもの：開発環境開発

ドキュメントの作り方、開発者/サポートとのI/F、不具合票の適切な書き方

- 「相手のレベル、背景を意識して」文章を書く
- 不具合票の構成の検討←「後へ活かす」しくみ造り

エミュレータ、デバッガの仕組み

- 確認するための仕組みを知っておく

GUIの使い勝手

- 要求仕様を理解する
- 上流へフィードバックをかける

業務を通して得られたもの：開発環境統合化テスト、マネージャ

自動化、統合化、統合テスト

- 段階的にテストを進める
- I/Fテストの重要性と全体を俯瞰すること
- 自動化でできること、できないこと

チーム編成

- 開発チームとテストチームの関係

構成管理、インストールテスト

- 周辺技術・情報の重要性

業務を通して得られたもの：IPA研究員、本社スタッフ

各社、各部門の状況、今までの整理

- 自分がやってきたエンジニアリング活動＝物を作る活動を客観的、総括的に眺める、振り返ること

ソフトウェアエンジニアリング、品質保証活動

- メトリクス、品質保証活動の意味

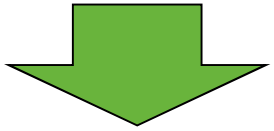
社外活動で得られたもの

社外コミュニティへの参加

- QuaSTom(高品質ソフトウェア技術交流会)
- Tef(ソフトウェアテスト技術者交流会)
- SESSAME(組込み管理者・技術者育成研究会)

イベント、団体への参画

- JaSST(ソフトウェアテストシンポジウム)
- JSTQB(Japan Software Testing Qualifications Board)
- ASTER(ソフトウェアテスト技術振興協会)



社内で得た知見を社外へ、社外で得た知見を社内へ
双方向での知見の交流

いったん総括

- 何を作っているのか、テストしているのかを知る
- どうやったらテストできるのか、どうやってテストしたらよいかを考える
- 技術の限界は知っておく
- チームつくりも大切
- 技術をいかに取り込むかを考える
- 人脈は最大限に生かす～社内も社外も～
  - キーパーソンは必ずいる

ここ10年の振り返り　～セミナ、書籍で語られたもの～

昔々のテキストから

- 基本的な技術＝不変の技術
- 古くなってしまった技術はないが、進化した技術はある

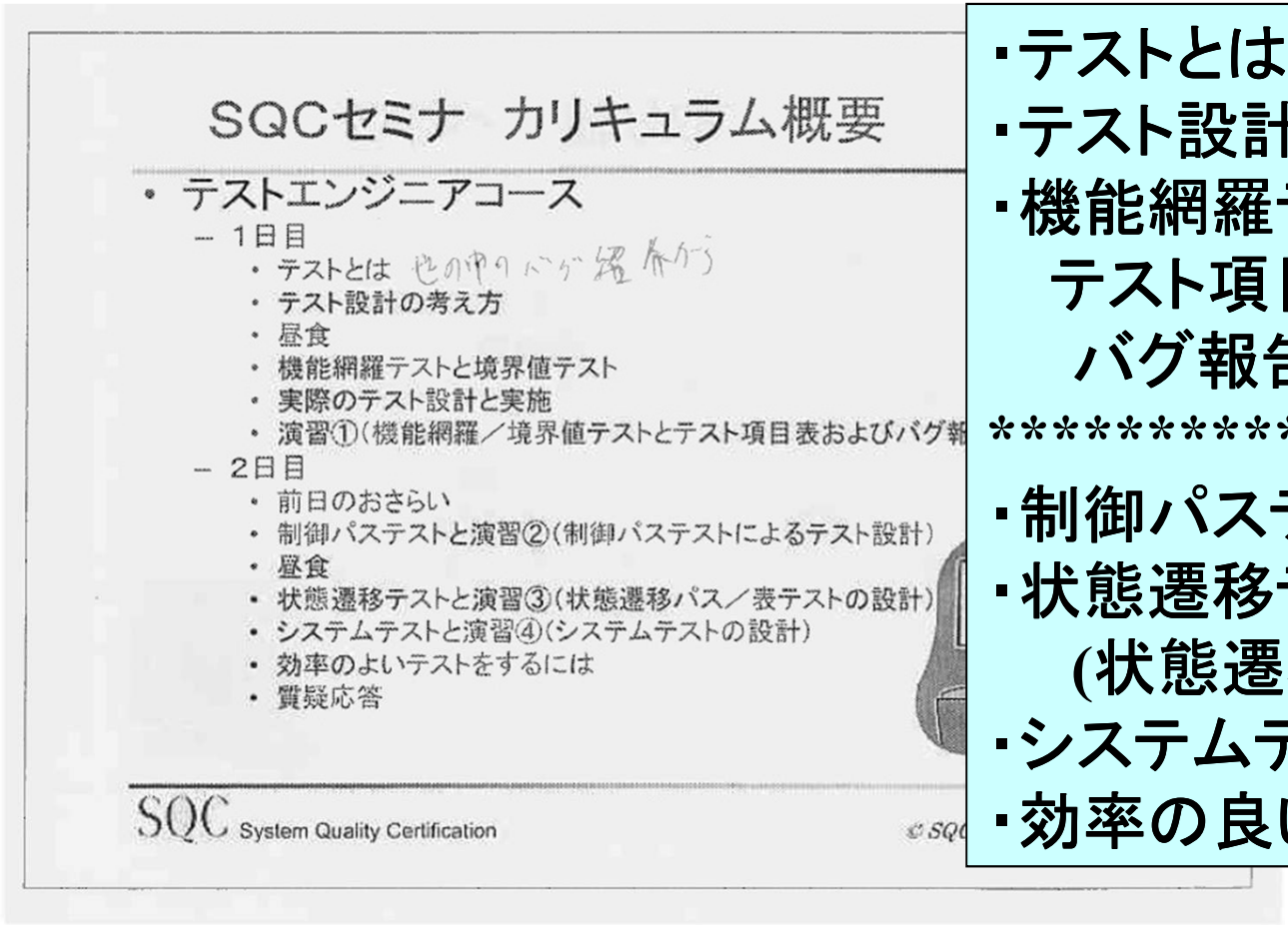
書籍の数は10倍に

- 10年前はMyersしかなかった。今は、選択に困るくらい
  でも、、、ちゃんと読んでます？

ここ10年の振り返り　～セミナ、書籍で語られたもの～

大西さんのカリキュラム

(出典：‘03.10.24 大西さんの講演資料)



・テストとは
・テスト設計の考え方
・機能網羅テストと境界値テスト
　テスト項目表および
　バグ報告書の作成

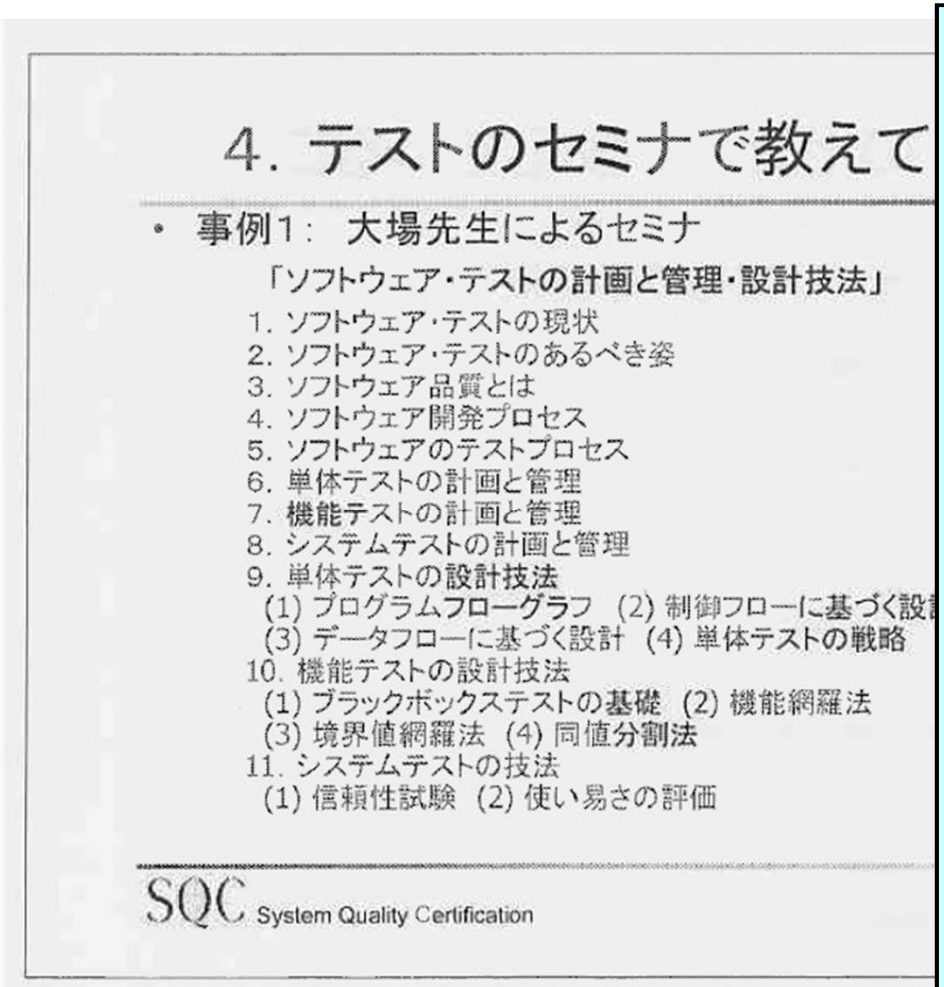
****

・制御パステスト
・状態遷移テスト
　(状態遷移パス／表テスト)
・システムテスト
・効率の良いテストをするには

ここ10年の振り返り　～セミナ、書籍で語られたもの～

大場先生(広島市立大)のテストセミナメニュー

(出典：‘03.10.24 大西さんの講演資料)

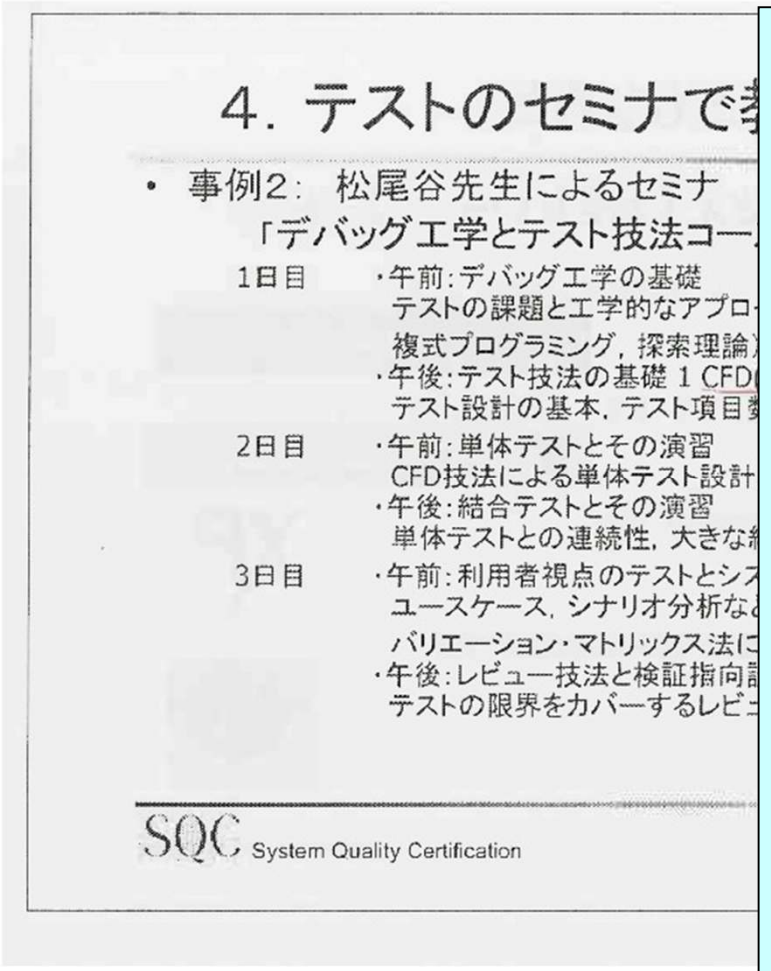


9. 単体テストの設計技法
(1) プログラムフローグラフ
(2) 制御フローに基づく設計
(3) データフローに基づく設計
(4) 単体テストの戦略
10. 機能テストの設計技法
(1) ブラックボックステストの基礎
(2) 機能網羅法
(3) 境界値網羅法
(4) 同値分割法
11. システムテスト技法
(1) 信頼性試験
(2) 使い易さの評価

ここ10年の振り返り　～セミナ、書籍で語られたもの～

松尾谷先生(デバッグ工学研究所)のセミナメニュー

(出典：‘03.10.24 大西さんの講演資料)



単体テストとその演習
　CFD技法による単体テスト設計、
　テスト設計の演習
・結合テストとその演習
　単体テストとの連続性、
　大きな結合テストの分割方法、
　その演習
・利用者視点のテストとシステムテスト
　ユースケース、シナリオ分析など
　　利用者視点のテスト
　バリエーション・マトリックス法による
　　システムテスト、その演習
・レビュー技法と検証指向設計

ここ10年の振り返り　～セミナ、書籍で語られたもの～

松尾谷さん(デバッグ工学研究所)の社内セミナメニュー

(出典：‘94.2.24 松尾谷さんの社内セミナ資料)

目 次

1 誤りと検証技術
1.1 誤りの定義 11
1.2 設計のエラーと検証のエラー 11
1.3 設計と検証の関係 14

2 検証の方法と分類
2.1 検証の構成要素 11
2.2 開発フェーズと検証活動 14
2.3 レビューの方法と分類 17
2.4 テストの方法と分類 20

3 テストの難しさと技法
3.1 場合の数の問題 11
3.2 同値分割と限界値分割 14
3.3 項目間の組み合わせ 17
3.4 原因結果グラフ技法 20
3.5 CFD 技法 23
3.6 状態遷移の問題 26

4 検証指向設計
4.1 VOP の概要 11
4.2 構造制約 14
4.3 制御制約 17

検証技術

H6/2/24

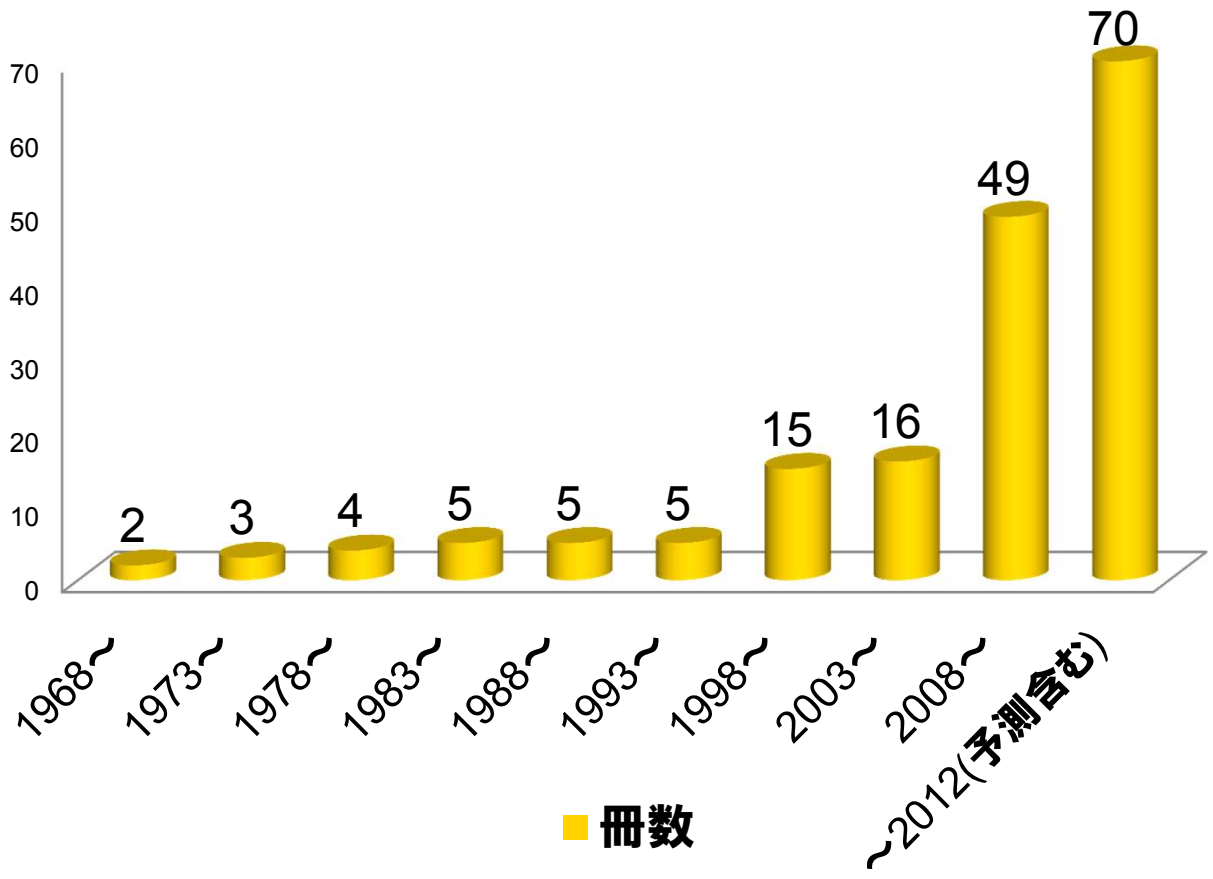
松尾谷　徹

3. テストの難しさと技法
3.1　場合の数の問題
3.2　同値分割と境界値分割
3.3　項目間の組み合わせ
3.4　原因結果グラフ技法
3.5　CFD技法
3.6　状態遷移の問題

ここ10年の振り返り　～セミナ、書籍で語られたもの～

書籍刊行状況（既刊147冊）

出典：JaSST'11 Tokyo　テスト初心者向けセッション



ここ10年の振り返り　～JaSSTで語られたもの～

JaSSTの歴史

講演内容の傾向の変遷

- 話題は、技術→プロセス志向へ？
- 基調・招待講演、一般講演、スポンサー講演、イベント類の傾向分析

ここ10年の振り返り ～JaSSTの歴史～

JaSSTの歴史 2003年に東京にて開催、その後全国へ

開催年	開催地
2003	東京
2004	東京
2005	東京、関西
2006	東京、関西、北海道
2007	東京、関西、北海道、九州(福岡)
2008	東京、関西、四国、北海道、九州(大分)
2009	東京、関西、四国、北海道、東海、九州(福岡)
2010	東京、関西、四国、北海道、東海、九州(熊本)
2011	東京、新潟、関西、四国、北海道、東海、九州(福岡)
2012	東京、新潟、、、

ここ10年の振り返り ～JaSST講演内容の傾向の変遷～

- 10年間で530のセッションがありました
- 参加者は延べで13800人！

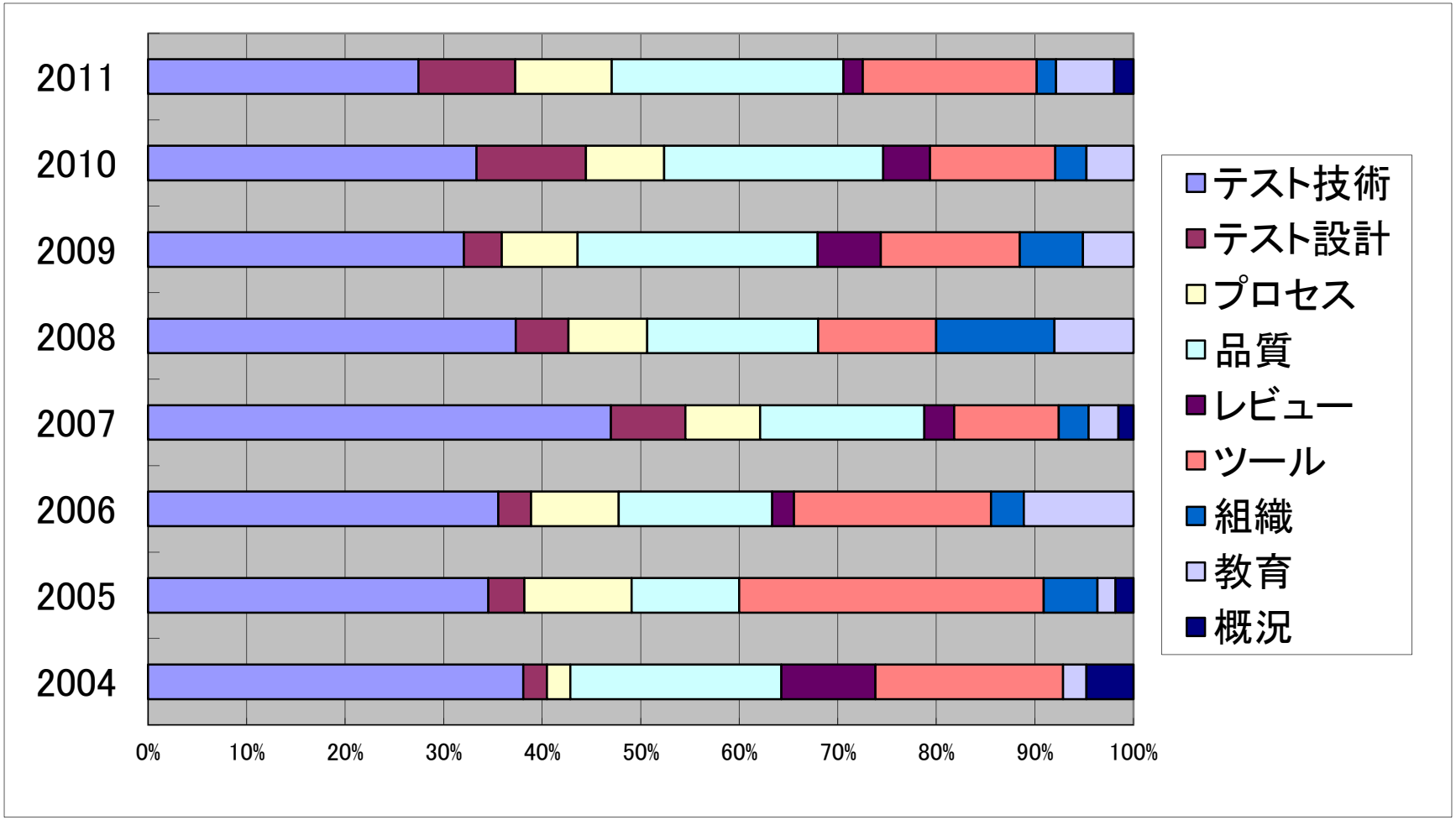
(基調・招待講演、チュートリアル、研究発表、ワークショップ、
パネルディスカッション、スポンサーセミナ、イベント系などなど)

地域	件数
北海道(2006-2010)	39
新潟(2011)	3
東京(2003-2011)	358
東海(2009-2010)	9
関西(2005-2011)	92
四国(2008-2011)	11
九州(2007-2010)	18
合計	530

ここ10年の振り返り　～JaSST講演内容の傾向の変遷～

JaSSTセッションの傾向は技術中心、内容の変化は年別ではそれほどない

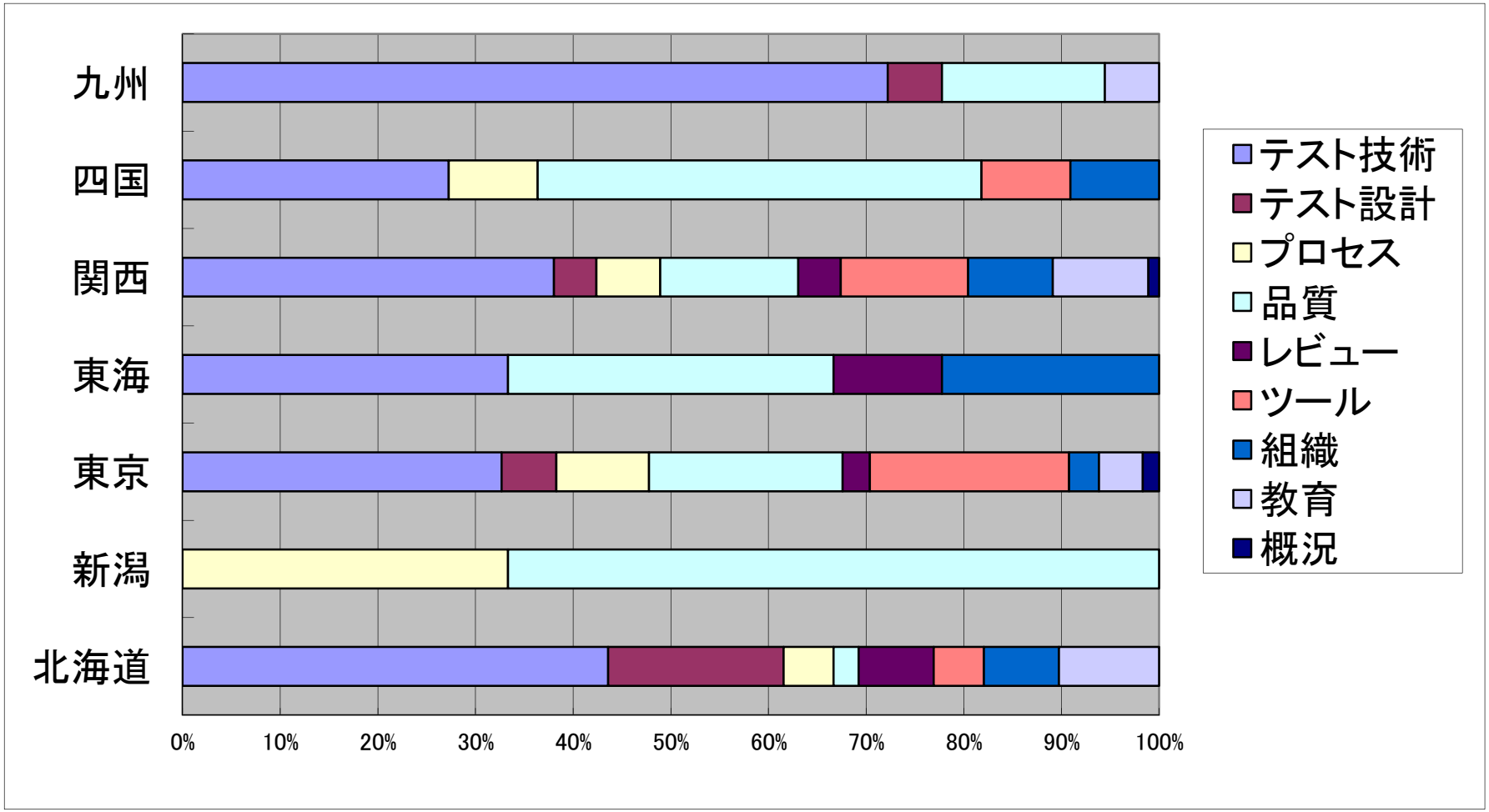
＜年別比較＞



ここ10年の振り返り ～JaSST講演内容の傾向の変遷～

各地域の興味は様々

＜地域別比較＞



ここ10年の振り返り ～JaSST講演内容の傾向の変遷～

各地域のキーワード、特色

地域	キーワード
北海道(2006-2010)	ユーザビリティテスト、スープカレー
新潟(2011)	組織力、派生開発＆DEBFM
東京(2003-2011)	何でもやってます。イベントも多数
東海(2009-2010)	独立検証、AUTOSER
関西(2005-2011)	テスト技術、教育、レビュー
四国(2008-2011)	テスト技術、メトリクス
九州(2007-2010)	品質保証、クリティカルデバイス

いったん総括その2

- 色々な技法、技術が提案されている、進化している
- では、現場は。。。？
  - テスト対象(＝ソフトウェア)の進化が早い？

  → オブジェクト指向設計、大規模化、開発期間の短縮、
  何よりテスト期間の短縮が急務
  - それほどやってることは変わらない？＝技術を取りこめているか？
    - 自動化をやってみたけど、、、

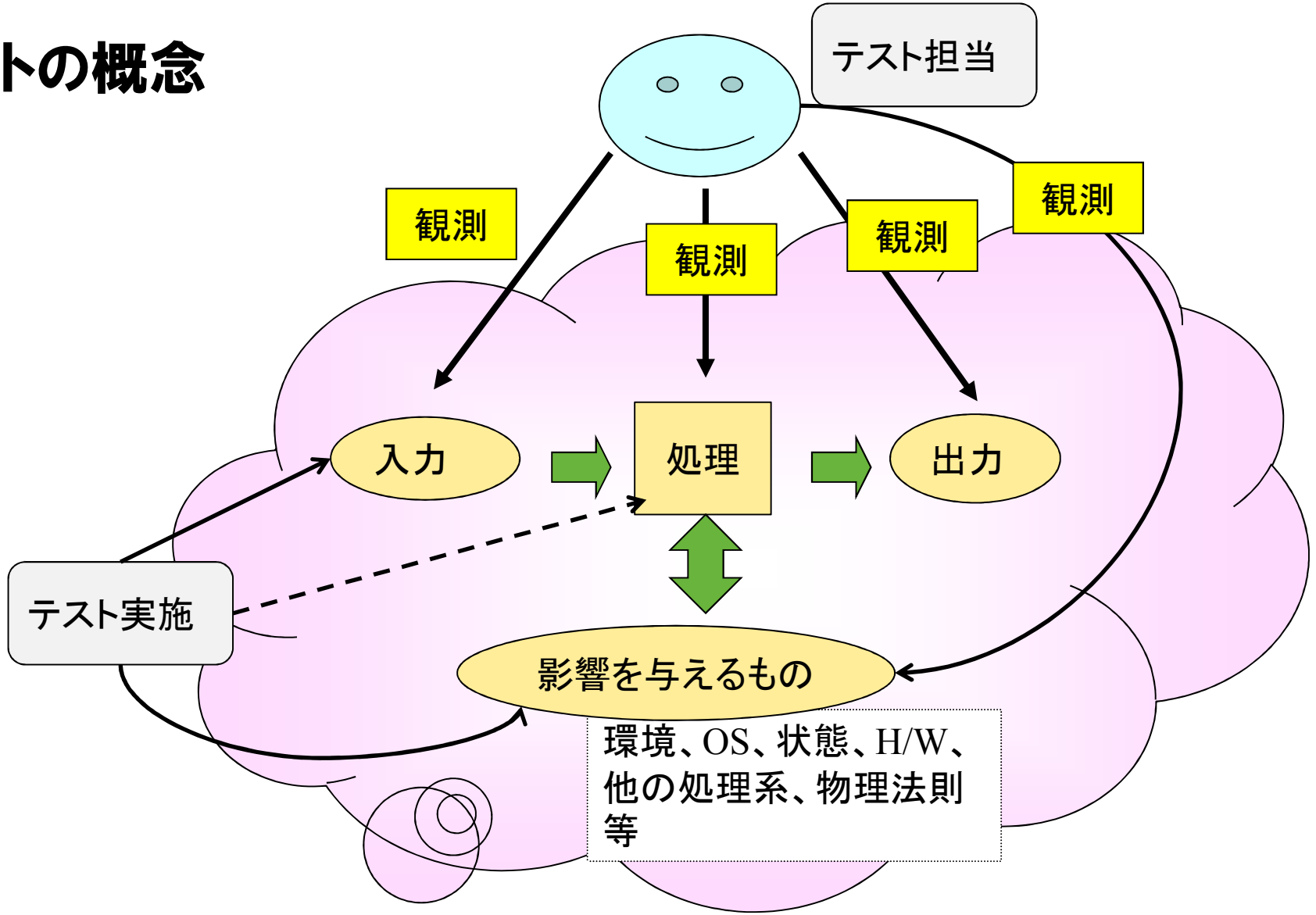
    → テストが深まるばかり？

身近なバグ事例からわかること

- 知らないことはチェックできない
  - →仕様を理解している必要がある
- ノイズがある、細かいものはわかりにくい
  - →ソフトウェアも複雑だとそれだけバグを見つけにくい
- チェックポイントがわかっているとバグが見つけやすい
  - →テストの観点を持っていると狙ってテストができる
- 思い込みがあると見つけられない
  - →ここは動くはず、、、と思い込んでいるとバグを見逃しやすい。
    - なんにでも疑ってかかるのがテスト屋魂
- 他に見るべきことが多すぎると細かいところに気が回らなくなる
  - →バグは少ない方がいいですね

テスト、、、バグをたたき出すためのアプローチ

テストの概念



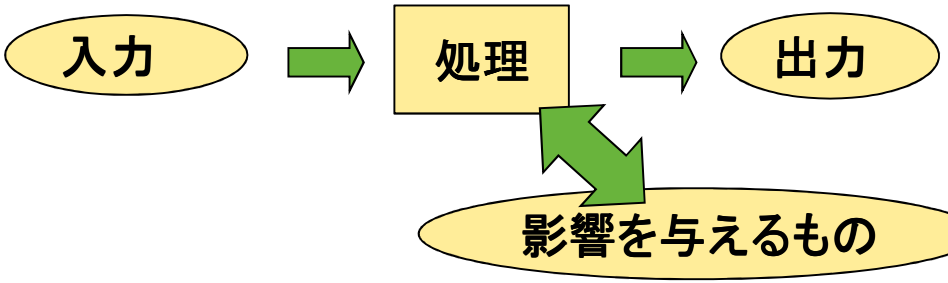
テストするということ

テストはどのように考えていく？

- 入力→出力
- 入力＋環境→出力
- 入力＋環境＋処理→出力
- 入力の異常系→出力
- 入力＋環境の異常系→出力
- 入力＋環境→出力、環境

この辺までは何となくテストできる

この辺から先は
よく考えない
とテストできない



環境、OS、状態、H/W、
他の処理系、物理法則
等

テストするということ

「ちゃんとテストする」＝テストの量を増やすこと、か？

→2つの考え方がある

- テスト項目を増す方向：網羅
- テスト項目を減らす方向：効率

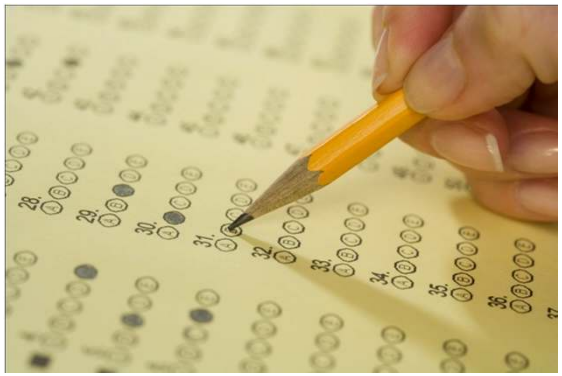
ここで改めて、テストの目的とは？ JSTQB FLシラバス 1.2. テストとは何か？より

- 欠陥を摘出する→バグをたたき出す
- 対象ソフトウェアの品質レベルが十分であることを確認する

  →使い物になることを保証する
- 出荷してよいかどうかの判断をする

  →きちんと「テストした」ことを見せる
- 欠陥の作り込みを防ぐ

  →開発にフィードバックする

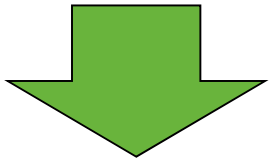


テストするということ

「ちゃんとテストする」＝テストの量を増やすこと、か？

→2つの考え方がある

- テスト項目を増す方向：網羅
- テスト項目を減らす方向：効率



- そのために様々なテスト技術がある

最近のテスト技術

- 昔からの技術、新しい技術
- 変わらなかった？技術、進化した？技術を取り上げます

テスト技術ってどんなものがある？

- ソフトウェア・テスト技法(マイヤーズ)
- ソフトウェアテスト技法ドリル(秋山さん)、
- テスト技法擬人化計画(JaSST'11 Tokyo 企画)
- 挙げられているテスト技法を比較してみました。

秋山さん＋擬人化＋マイヤーズ
エラー推測
原因結果グラフ
同値分割
境界値分析

秋山さん＋擬人化
負荷テスト
シナリオテスト
いじわるテスト
オールペア/ペアワイズ
探索的テスト
デシジョンテーブルテスト
統計的テスト
HAYST法、直交表

秋山さん
「間、対称、類推、外側」の観点
テストの観点
受け入れテスト
CFD法
Nスイッチカバレッジ
三色ボールペン
サンプリングテスト
状態遷移図
状態遷移表
スライド法
探針テスト
ドメイン分析テスト
不具合モード
並列処理テスト
例外シナリオ

擬人化
α・βテスト
業務フローテスト
自動テスト
スモークテスト
ユースケーステスト
アドホックテスト
異常値・特異値分析テスト
運用プロフィール
終わらないテスト
期末テスト
スライド法
制御パステスト(C0,C1)
セキュリティテスト
ソープオペラテスト
タイミングテスト
突入テスト
ピンポイントテスト
ファジング
ファズテスト
ペア構成テスト
並列テスト
変異テスト(ミューテーションテスト)
モデルチェッキング
モンキーテスト
ランダムテスト
ループ境界テスト

マイヤーズ＋擬人化
形式手法

マイヤーズ
ウオークスルー
机上チェック
ピアレビュー
複雑さ
システムテスト
導入テスト
回復テスト
機能テスト
機密保護テスト
構成テスト
効率テスト
互換性/変換テスト
サービス性テスト
信頼性テスト
ストレステスト
設置テスト
増加テスト
大容量テスト
手続きテスト
認可テスト
ブラックボックステスト
文書テスト
ボトムアップテスト
ホワイトボックステスト
有用度テスト
記憶域テスト
トップダウンテスト
予測モデル
漏洩回路分析
限界値分析＝境界値分析

主なテスト技法

古くからの手法

秋山さん＋擬人化＋マイヤーズ
エラー推測
原因結果グラフ
同値分割
境界値分析

同値分割、境界値分析は
基本中の基本

最近(かもしれない)の手法

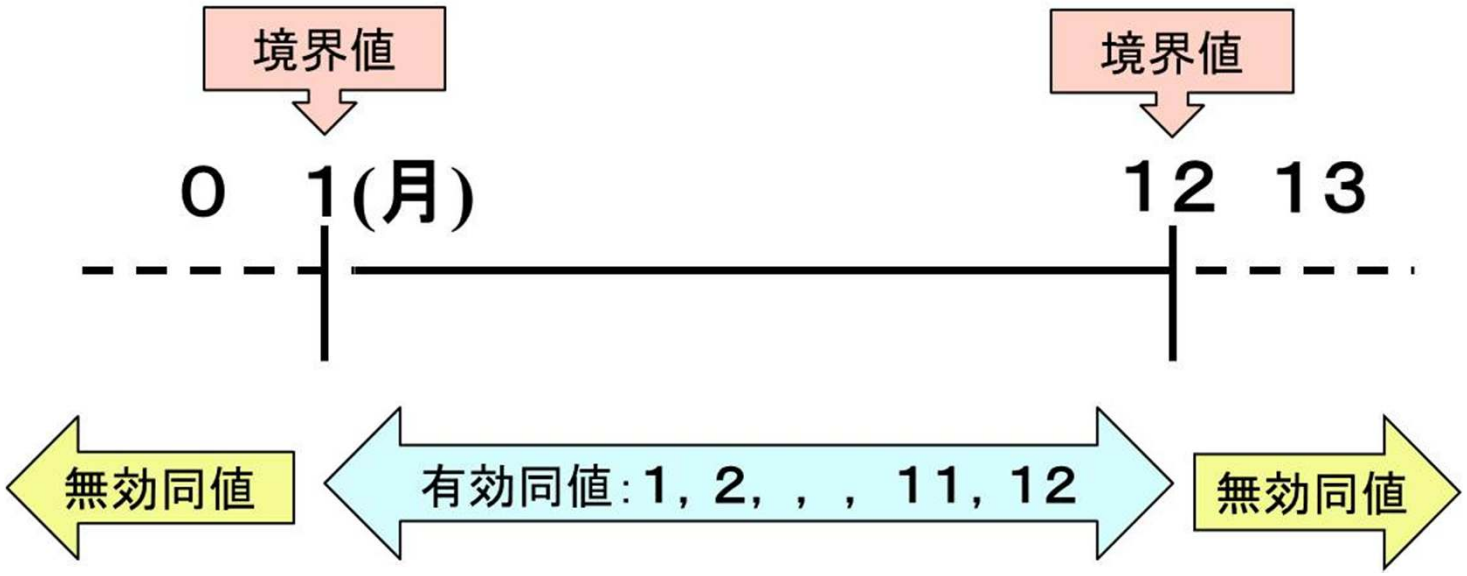
秋山さん＋擬人化
負荷テスト
シナリオテスト
いじわるテスト
オールペア/ペアワイズ
探索的テスト
デシジョンテーブルテスト
統計的テスト
HAYST法、直交表

大規模ソフトに対しての
テスト技術(負荷テスト、シナリオ
テスト、ペアワイズなど)が
出てきている

マイヤーズ本　2版で追加になったもの
Webプログラミング
電子商取引 （インターネットアプリケーション）
エクストリームプログラミング（XP）

同値分割、境界値分析

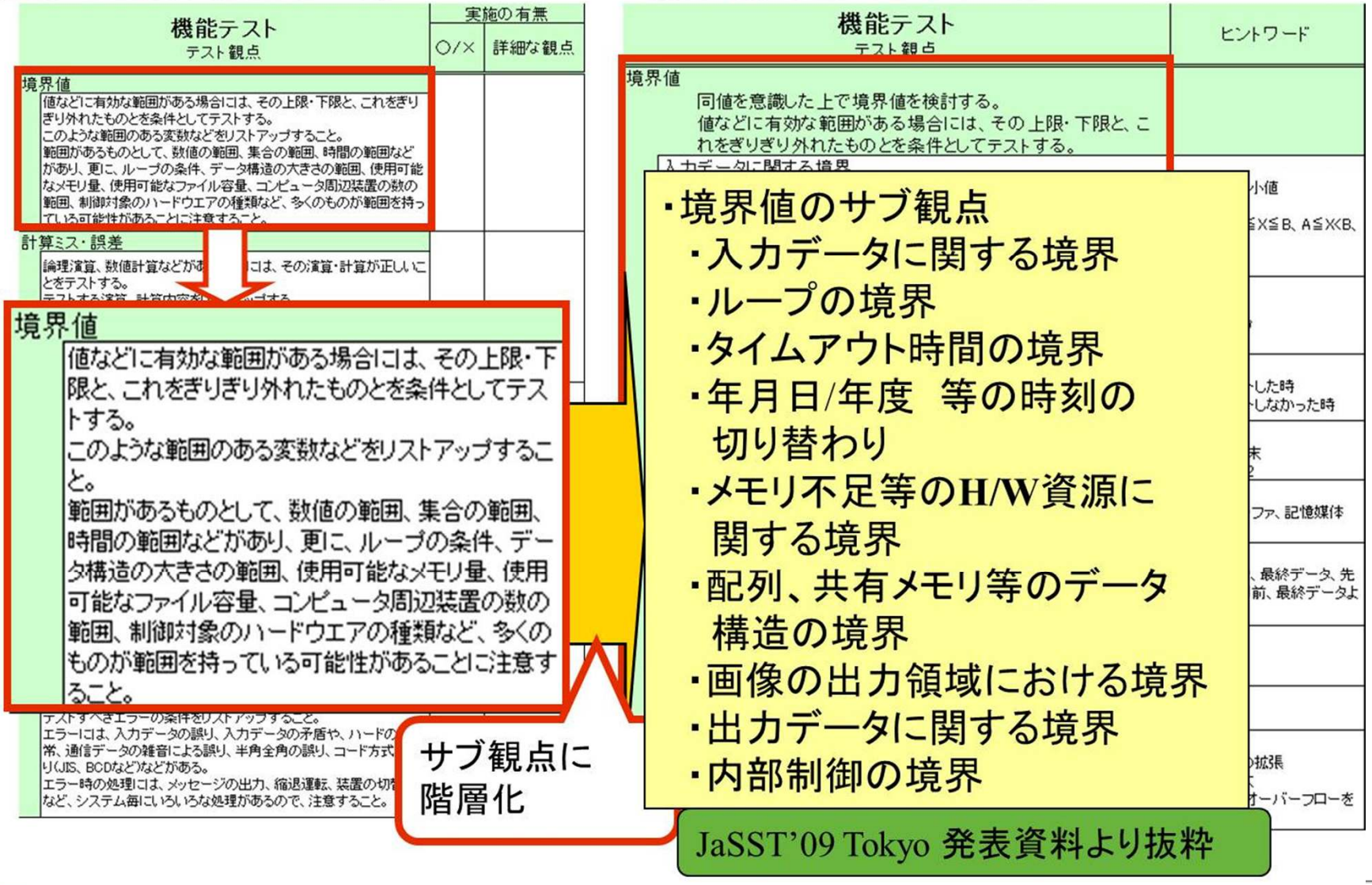
- 古くて新しい技術、実はみんなやっている
- JaSST東京の初心者チュートリアルでは必ず取り上げられている
- 連続したデータ入力の有効、無効を表す
- 例:月の入力を整数で行う場合



テスト観点：NECでの例：テスト設計テンプレート

旧テスト設計テンプレート

新テスト設計テンプレート



ソフトウェアの変遷とテストの変遷

ソフトウェア技術の変遷：課題は増していくばかり

- 大規模化・短納期化
- ソフトの重要性の高まり

テスト技術の悩み：ソフトに追従していかないといけないが、、、
やることは増えるのに時間は相変わらず十分でない

- 大規模化 → 人海戦術？
- 重要性 → 人海戦術？とにかくテストする？

テストの変遷

- とにかく全部テストする ← カバレッジ

- テスト数を減らそうと頑張る ← 同値分割、直交表、、、

- 全部の範囲がカバーできるよう、合理的にテストする
← テスト設計、自動化、リスクベースドテスト、、、

ゆもつよメソッド

(出典：‘11.5.13 ESEC 西、吉澤の講演資料)

機能一覧			テストカテゴリ					
機能分類	機能項目	仕様書頁	ボタン操作	計算	登録・更新・削除	入力チェック	反映	表示
システム管理	起動/終了	電源ON/OFF						
	リセット	リセット						
	電源管理	充電						
		電源管理						
撮影	(1枚づつ)撮影	ズーム撮影	○	○	○	○	○	○
		フラッシュ撮影			○			
	連続撮影	連続撮影	○					
		ズーム撮影	△		○			
再生	サムネイル表示	一覧再生						
	(1枚づつ)再生	通常再生				○		
		再生画像拡大表示						
	スライドショー再生	スライドショー再生						
設定	撮影設定	撮影モード設定						
		撮影設定						
		画質設定						
		ホワイトバランス						
		オートフォーカス				△		
		画像サイズ変更						
	再生設定	再生モード設定		○				
		表示レイアウト				○		
		サムネイル設定						
	日時設定	日時設定	◎	◎	◎	◎	◎	◎
		海外日時設定	◎	◎	◎	◎	◎	◎
		タイムスタンプ			○			

何処でテスト条件を列挙するかを◎や○で記述

・テスト分析マトリクス

参考：

「ソフトウェア・テスト PRESS Vol.10 」

特集1　今こそ聞きたい テストの上流設計

湯本 剛　　ISBN-10: 4774143413

HAYST法-FV表

(出典：‘11.5.13 ESEC 西、吉澤の講演資料)

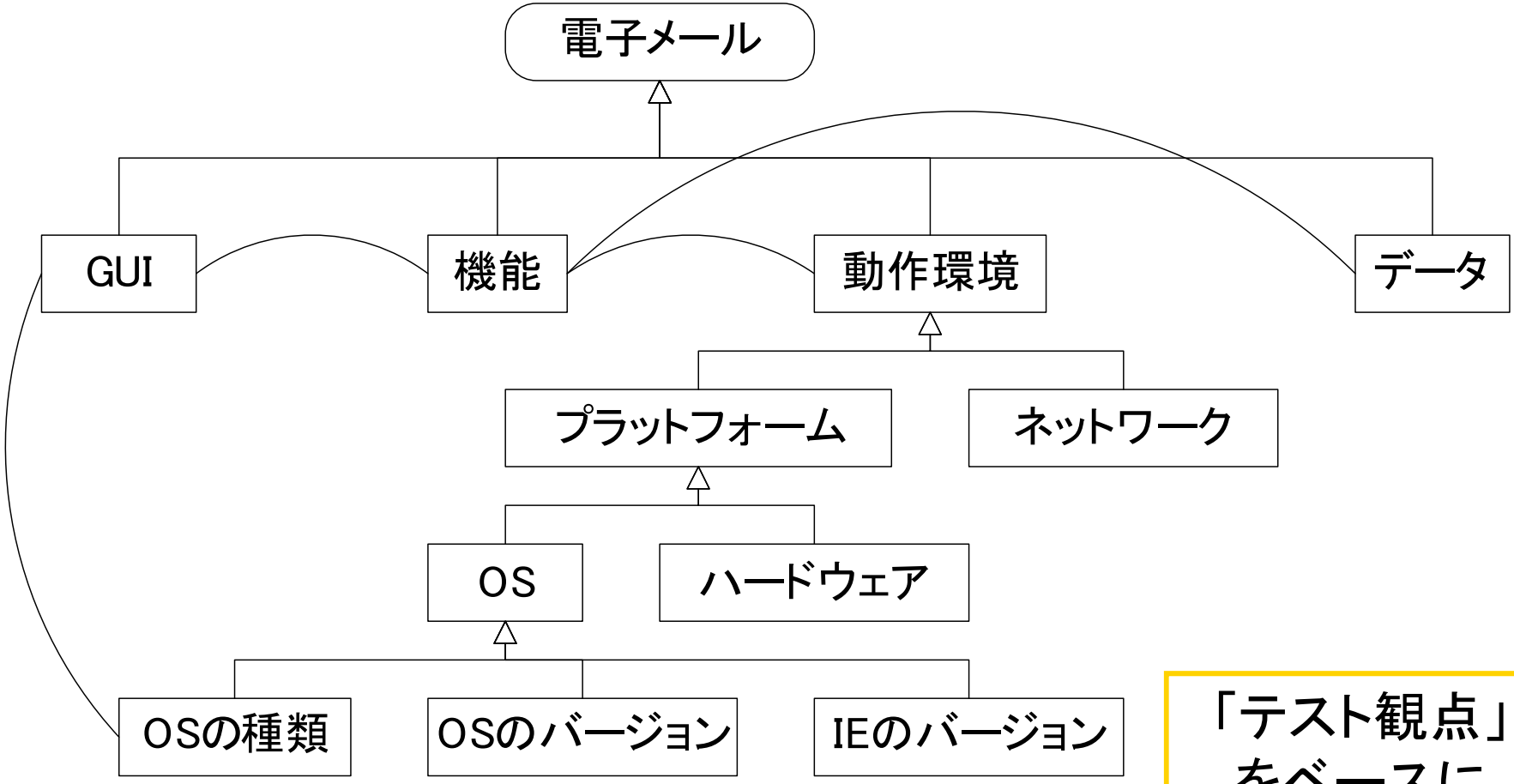
FV表：Function Verification Table

- 「ソフトウェアテストHAYST法入門」より
  - 吉澤 正孝/秋山 浩一/仙石 太郎, 日科技連出版社, ISBN4-8171-9228-3

番号	機能	検証内容	使用するテスト技法
1	商品を検索する	・希望する商品が速やかに検索できること	
1.1	フリーワードで検索	・存在する商品名で検索できること ・存在しない商品名を入力したときにエラーとなること	・商品名をHAYST法で設計
1.2	商品カテゴリから検索	・リンク切れがないこと	・総探索を自動化処理で実施
2	数量を選択し購入する	・数量を入力し計算ボタンを押すことで価格表示が変わること ・ラッピングの指示が反映されること ・購入ボタンで画面が遷移すること ・途中でブラウザが閉じられても購入が継続できることを確認すること	・HAYST法

NGT

(出典：‘11.5.13 ESEC 西、吉澤の講演資料)



「テスト観点」
をベースに
モデリングする

最近のテスト技術　まとめ

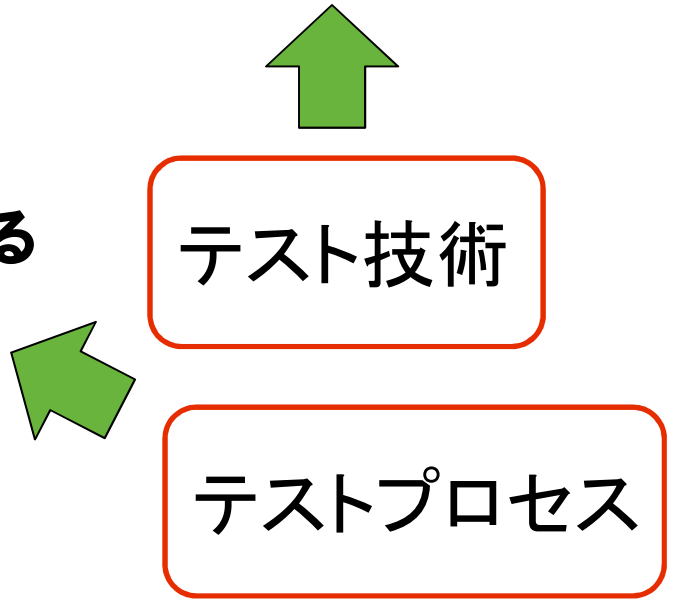
- テスト方法論＝何をどのようにテストしていくか、をまず考え、その中の各場面でそれぞれに用いることのできる技術を決める
- どの技術をどのように使うかは自分で考えること
- 自分に必要な技術は何か、目的を持って取り込むことが大切
- テストの観点を上流にも生かす

テストするということ

ここで改めて、テストの目的とは？

JSTQB FLシラバス 1.2. テストとは何か？より

- 欠陥を摘出する→バグをたたき出す　　テスト技術
- 対象ソフトウェアの品質レベルが十分であることを確認する
  →使い物になることを保証する
- 出荷してよいかどうかの判断をする
  →きちんと「テストした」ことを見せる
- 欠陥の作り込みを防ぐ
  →開発にフィードバックする



テスト開発プロセス

- ちゃんと作る、ちゃんとテストする、技術
- 良いマネージャ、良い組織は実は頭の中でやっている
- 例えば、JSTQBでは以下のように定義している
  - 1.4.1. テスト計画作業とコントロール
  - 1.4.2. テストの分析と設計
  - 1.4.3. テストの実装と実行
  - 1.4.4. 終了基準の評価とレポート
  - 1.4.5. 終了作業

テスト開発プロセス

ちゃんと作る、ちゃんとテストする、技術

- 1.4.1. テスト計画作業とコントロール
  - 計画を立てる
  - 計画の状況を測り、測った結果をフィードバックする
- 1.4.2. テストの分析と設計
  - 対象をちゃんと知る
  - どのようにテストするか考える(俯瞰と網羅)
- 1.4.3. テストの実装と実行
  - 詳細化、手順化
- 1.4.4. 終了基準の評価とレポート
  - どこまでテストしたらよいか、どのように報告するか
- 1.4.5. 終了作業
  - まとめ：保守、次の開発へのフィードバック

どのくらいテストしたらよいの？

- テスト対象によって変化する
- テスト対象を見極める
- ESQRでは項目数について提案がある

ESQR:品質作り込みガイド　2008年12月 IPA/SECが提唱

求められる品質レベルに応じて、品質を保証する行為(レビュー、テスト)に関する作業の十分性を示す指標を提示

1.製品＝テスト対象に求められる品質レベルを分析→品質レベルを知る

2.製品を開発する上での環境を分析→品質レベルを補正

3.プロセス、プロダクトの要素に応じて品質を保証するための作業の十分性の指標を提示、コントロール

(例)

- テスト作業実施率／充当率
- テスト密度

ESQR:品質作り込みガイド

1. 製品＝テスト対象に求められる品質レベルを分析→品質レベルを知る

ユーザがどの程度の品質を求めているのか？

うまく動かなかった場合に(誤動作、動作停止)

・人や物にどの程度の危害を加えるか

・どの程度の損害があるか？

⇒ テストの量に影響

2. 製品を開発する上での環境を分析→品質レベルを補正

自分たちはどのような環境で物を作っているか？

・難しさは？環境は？

・プロジェクトの状況は？

⇒ テストの量をコントロール

ESQR:品質作り込みガイド

3. プロセス、プロダクトの要素に応じて品質を保証するための作業の十分性の指標を提示、コントロール

(例)

注意！
テスト項目、テスト結果のレビュを
きちんとやってこその指標です

- テスト作業実施率

  開発作業全体に対してどのくらいテストするのか？

- テスト作業充当率

  開発規模に対してどのくらいテストするのか？

- テスト密度

  開発規模に対してどのくらいのテスト項目が必要か？

おわりに

テストで「楽」になるのか？

今日が初めの一歩です。全国に仲間がいます。
会社を飛び出して、腕を、頭を磨きましょう。
でも、会社への持ち帰りも忘れないくださいね。

- 北海道：Tef道(聡美塾、HOTATE)
- 新潟：SWANii
- 東京：テスト技法ドリル勉強会、智美塾
- 東海；Tef東海
- 関西；てふかん
- 福岡；Q魂

、、、皆さんの周りにはありますか？

おわりに

テストで「楽」になるのか？

●早く帰れる。。。が、一番ですが、テストの環境はますます厳しくなっています。

- 大規模化＝テストの作業量は増すばかり、複雑化＝テストは難しくなるばかり
- 短納期＝テストの量は増えるのに、期間は短くなるばかり

●せめて、枕を高くして眠れるように

- テストを進めていくプロセスをきちんと決める、開発とタッグを組んで早め早めにバグを追い出す
- テスト技術の腕を磨いて効果的なテストをする
- リリース後のバグを減らして心配ばかりの夜をなくす
- リリース後のバグがなくなれば前製品のテストに追われることなく、新製品の開発・テストに注力できる

テスト技術を工夫してうまく取り込み、楽になりましょう

おわりに

- 今日が初めの一歩です。全国に仲間がいます。
- 会社を飛び出して、腕を、頭を磨きましょう。
- でも、会社への持ち帰りも忘れないくださいね。

- 北海道：Tef道(聡美塾、HOTATE)
- 新潟：SWANii
- 東京：テスト技法ドリル勉強会、智美塾
- 東海；Tef東海
- 関西；てふかん
- 福岡；Q魂

、、、皆さんの周りにはありますか？

次の10年をつくるのは私たちです！

ご清聴ありがとうございました

NECグループビジョン2017

人と地球にやさしい情報社会を
イノベーションで実現する
グローバルリーディングカンパニー

Empowered by Innovation