

テスト設計 A4-1

ソフトウェアテストとマインドマップ そして Agile Inspection

ソニー株式会社 永田 教

2008年1月28日

SONY

ソフトウェアテストシンポジウム JaSST '09 Tokyo copyright (c) 永田 教

目次

- マインドマップによるテスト分析・設計
- リビューのモチベーション
- リビューの目的と効果
- Agile Inspection
- Workshopの結果
- Agile Inspection Policy
- 位置づけ
- これから

SONY

ソフトウェアテストシンポジウム JaSST '09 Tokyo copyright (c) 永田 教

謝辞

今回、JaSST'09 Tokyoで発表させていただくことにつきましては、
さまざまの方にご支援をいただきました。

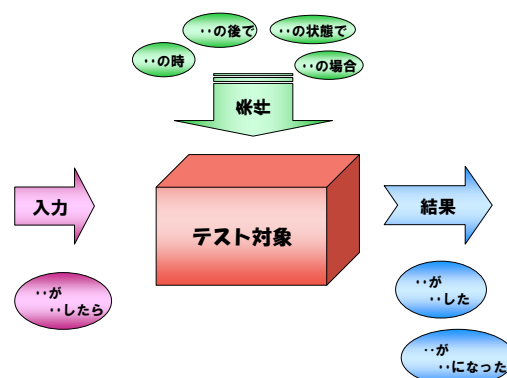
HAYST法を教えていただいた、吉澤 正孝様、秋山 浩一様
前回の発表のきっかけを作ってくださいました、池田 暁 様
本物のマインドマップってこうなのよと教えてくださった 須田様
TestLink分科会の皆さん
マインドマップを教えてくださいました、マインドマップ公認インストラクタの
松岡 克政 様
Agile Inspectionを教えてくださいました、Tom Gilb 様、Kai Gilb様
Agile Inspection Workshopに参加していただいたTEFの皆さん
JaSST'08 Tokyoで発表を聞いていただいた皆さん
そして、ご意見ご感想をくださった皆さん

この場をお借りして、深くお礼を申し上げます。

SONY

ソフトウェアテストシンポジウム JaSST '09 Tokyo copyright (c) 永田 教

TEST System



SONY

ソフトウェアテストシンポジウム JaSST '09 Tokyo copyright (c) 永田 教

テスト設計

分析対象

仕様書

- 要求仕様書
- 基本設計仕様書
-

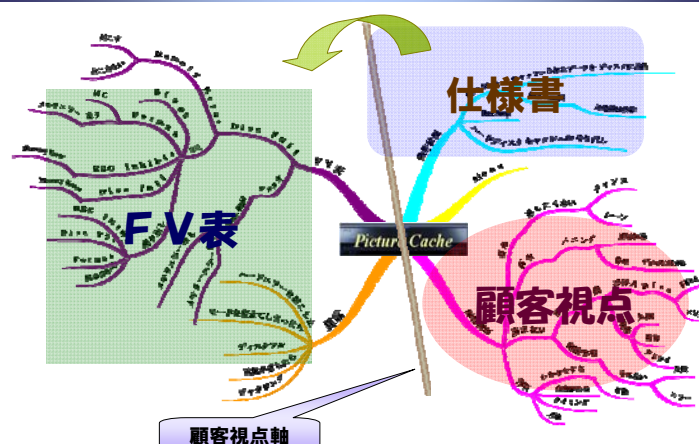
分析

テスト設計書

SONY

ソフトウェアテストシンポジウム JaSST '09 Tokyo copyright (c) 永田 教

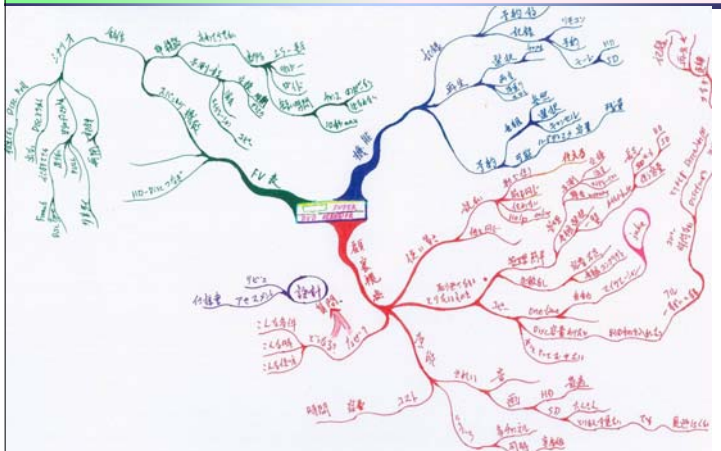
マインドマップによる転写



SONY

ソフトウェアテストシンポジウム JaSST '09 Tokyo copyright (c) 永田 教

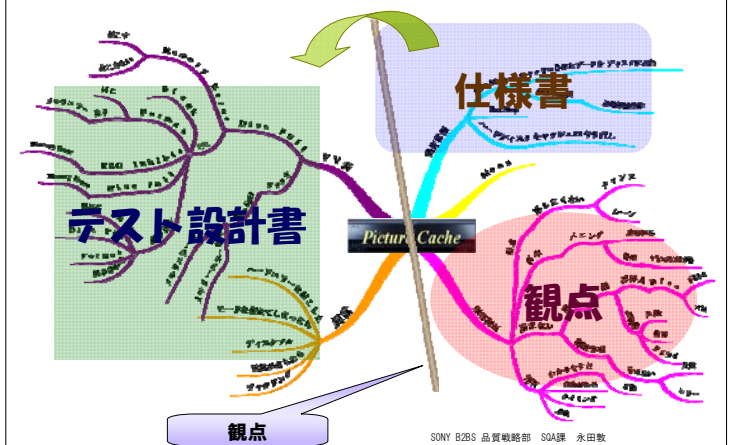
手書きのマインドマップ



SONY

ソフトウェアテストシンポジウム JaSST '09 Tokyo copyright (c) 永田 教

マインドマップによる転写



SONY

ソフトウェアテストシンポジウム JaSST '09 Tokyo copyright (c) 永田 教

観点

- 状態
 - ソフトウェアの内部状態
 - ◆ 初期化処理中か安定動作中か
 - ハードウェアの状態
 - ◆ ヘッドの位置
- タイミング
 - 機能同士のタイミング
 - 機能とハードウェアのタイミング
- 組み合わせ
 - 同じ機能をいくつかカバーするか
 - 異なる機能を何種類組み合わせるか
- 性能
 - 最も遅そうな条件は何か
- 信頼性
 - 要求連続稼働時間
- GUI・操作性
 - 操作パス、ショートカット
 - 操作が禁止される状況は何か
 - ユーザシナリオ、10モード
 - 操作ミス、初心者操作、子供
- 出荷先
 - 電源電圧、気温、ユーザの使い方
 - 言語、規格、法規
- 障害対応性
 - 対応すべき障害の種類
 - ◆ 水没
 - 対応動作の種類
- セキュリティ
 - 扱う情報の種類や重要度
 - 守るべきセキュリティ要件

電気通信大学 西 康晴

SONY

ソフトウェアテストシンポジウム JaSST '09 Tokyo copyright (c) 永田 教

テスト戦略

テスト戦略立案というフェーズは暗黙的に行われることが多い

テスト観点のモデリングを行ってテスト戦略を立案すべきである

テストで考慮すべき観点を一覧でき、俯瞰的に整理できる

電気通信大学 西 康晴

SONY

ソフトウェアテストシンポジウム JaSST '09 Tokyo copyright (c) 永田 教

ソフトウェアテストコミュニティ

ソフトウェアテストとマインドマップ



池田 暁

テストイングリタラシー
in 厚木

松岡 克政

メンタルリタラシー

SONY

ソフトウェアテストシンポジウム JaSST '09 Tokyo copyright (c) 永田 教

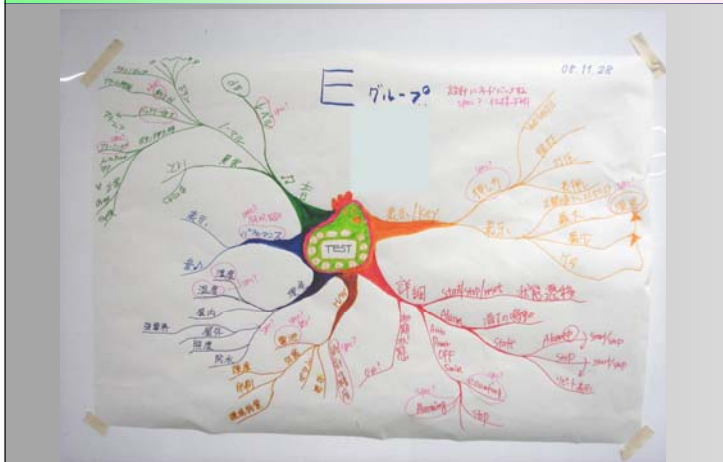
テスト設計+マインドマップ



SONY

ソフトウェアテストシンポジウム JaSST '09 Tokyo copyright (c) 永田 教

テスト設計+マインドマップ



SONY

ソフトウェアテストシンポジウム JaSST '09 Tokyo copyright (c) 永田 敦

テスターは
読まねば
ならない

テスト
ベースを

SONY

ソフトウェアテストシンポジウム JaSST '09 Tokyo copyright (c) 永田 敦

レビューのモチベーション

テストベース
不明確
曖昧
矛盾

レビュー
質問
欠陥発見
修正
上流で
品質向上!

メジャーな欠陥

テストケース書かんでバグ直した!!

SONY

ソフトウェアテストシンポジウム JaSST '09 Tokyo copyright (c) 永田 敦

では

では

SONY

ソフトウェアテストシンポジウム JaSST '09 Tokyo copyright (c) 永田 敦

レビュー対象

■ドキュメント

- 要求書
- 要求仕様書
- 要求定義書
- 品質構想書
- 基本設計書
- 外部仕様書
- テスト計画書
- テスト設計書
- テストケース



SONY

ソフトウェアテストシンポジウム JaSST '09 Tokyo copyright (c) 永田 敦

目的

■ テストを設計する

- 仕様書分析

■ ドキュメントの品質を測る

- ドキュメントの品質を測定、推定する
 - ◆ ドキュメントにどのくらいの欠陥が存在するか
 - ◆ 欠陥 : トラブルになりうるリスク
- ドキュメントの質そのものを上げる施策を打つ



■ ドキュメントの欠陥を発見する

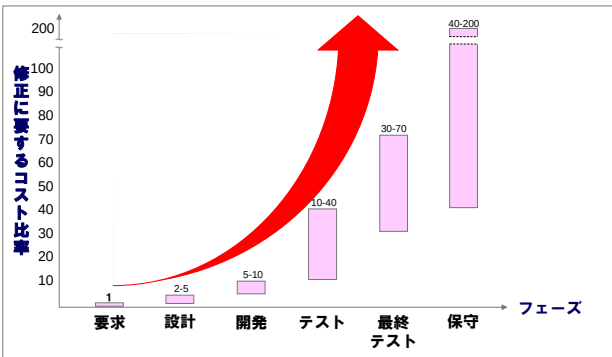
- 少しでも欠陥の侵入を防ぐ
- バグゼロの法則
 - ◆ 欠陥を完全にゼロにすることはできない。

でも待てよ

SONY

ソフトウェアテストシンポジウム JaSST '09 Tokyo copyright (c) 永田 敦

効果：修正のコスト



出典：Grady 1999

SONY

ソフトウェアテストシナリオ JaSST '09 Tokyo copyright (c) 永田 敦

欠陥除去効果の比較

Defect Removal Effectiveness Inspections and Tests

Table 9: Software Defect Removal Effectiveness Ranges

Capers Jones

Defect Removal Activity

Ranges of Defect Removal Effectiveness

Informal design reviews	25% to 40%
Formal design inspections	45% to 65%
Informal code reviews	20% to 35%
Formal code inspections	45% to 70%
Unit test	15% to 50%
New function test	20% to 35%
Regression test	15% to 30%
Integration test	25% to 40%
Performance test	20% to 40%
System test	25% to 55%
Acceptance test (1 client)	25% to 35%
Low-volume Beta test (< 10 clients)	25% to 40%
High-volume Beta test (> 1000 clients)	60% to 85%

SONY

ソフトウェアテストシナリオ JaSST '09 Tokyo copyright (c) 永田 敦

JSTQBでは

- 非公式レビュー
 - 金や時間をかけずにある程度の効果を出すこと
- ウォークスルー
 - ソフトウェアの内容を学習、理解し、欠陥を見つけること
- テクニカルレビュー
 - ディスカッションすること、判断を下すこと、他の方法を評価すること、欠陥を見つけること、技術的な問題を解決すること、仕様や規則に準拠していることをチェックすること
- インспекション
 - 欠陥の発見

JSTQB Foundation Level シラバス V1.0.1 より

SONY

ソフトウェアテストシナリオ JaSST '09 Tokyo copyright (c) 永田 敦

方法

- サンプルング
 - 1ページだけをインスペクトして品質を推定
 - Agile Inspection
- 局所レビュー
 - 観点を持って狙いを付けてレビューする
- 全レビュー
 - ドキュメントすべてをレビューする



SONY

ソフトウェアテストシナリオ JaSST '09 Tokyo copyright (c) 永田 敦

Inspection Facts:

to help you make good decisions to do
Software Inspections and Reviews properly



Tom Gilb

■ JUSE Tokyo September 2008

■ Tutorial by Tom Gilb

- Copyright: © Gilb 2008
 - Sources: Gilb, Competitive Engineering (2005),
- Gilb and Graham, Software Inspection (1993):
- Japanese Edition
- Tom Gilb : Inspection_for_Managers_JUSE_08" より



kyoritsu-pub.co.jp

SONY

ソフトウェアテストシナリオ JaSST '09 Tokyo copyright (c) 永田 敦

サンプリング

■ Agile Inspection Workshop

やってみましょう

どのくらい、欠陥があるか 体験できます

● ルール 対象 要求仕様書

- あいまいでないこと
- 明解であること
- 矛盾のないこと
- 設計要素がないこと

■ 1ページだけ15分かけてチェックしましょう



SONY

ソフトウェアテストシナリオ JaSST '09 Tokyo copyright (c) 永田 敦

Agile Inspection Workshop in TEF

Checker	Major	minor	Design	Time min	Words	Major /LP
1	25	0	2	15	273	27.5
2	14	0	4	15	200	21.0
3	14	0	4	15	300	14.0
4	11		0	15	160	20.6
5	10		2	15	320	9.4

5	Number of Checkers
18.5	Number of Unique Majors found per 300 words:
33%	% effectiveness
56.1	Actual Major per 300 words
11	Pages (300 words) in document
617	Total Majors in Document
204	1/3 actually occur / hit / go-wrong
9	Average Cost in HOURS per Major if let through to the next stage
1832	Estimated delay in WORK HOURS caused by Majors

2009/10/10

SONY

ソフトウェアテストシンポジウム JaSST '09 Tokyo copyright (c) 永田 敦

結果は？

たった1ページで

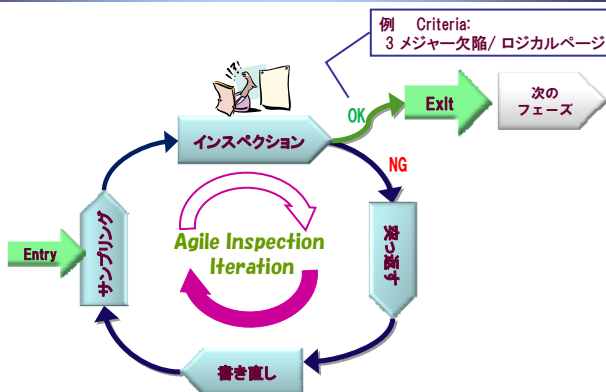
いくつかのメジャーな Defectが
見つかったでしょうか？

まだ、**2/3**は見つかっていません
すべての欠陥の**3割**が
インプリメントされ**バグ**となります。
バグひとつあたりを直すのに
9時間以上はかかります。

SONY

ソフトウェアテストシンポジウム JaSST '09 Tokyo copyright (c) 永田 敦

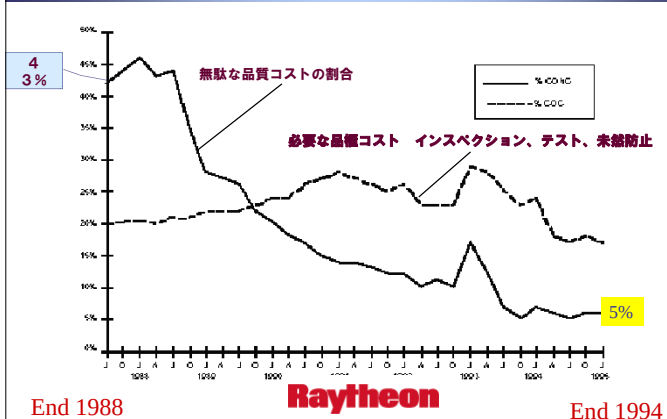
Agile Inspection Process



SONY

ソフトウェアテストシンポジウム JaSST '09 Tokyo copyright (c) 永田 敦

品質改善コスト：RAYTHEONの場合



SONY

ソフトウェアテストシンポジウム JaSST '09 Tokyo copyright (c) 永田 敦

Agile Inspection Policy 1

REVIEW EARLY

- レビューは早いうちにできているところからやる。



全部そろうまで
待つ

Engineer Your Review Process : Tom Glib 2008

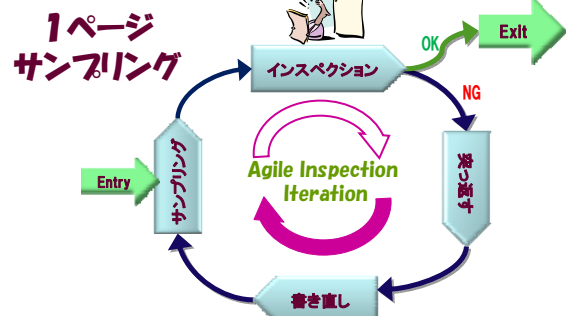
SONY

ソフトウェアテストシンポジウム JaSST '09 Tokyo copyright (c) 永田 敦

Agile Inspection Policy 1

REVIEW EARLY

- レビューは早いうちに“できているところ”からやる。



SONY

ソフトウェアテストシンポジウム JaSST '09 Tokyo copyright (c) 永田 敦

Agile Inspection Policy 2

■ AVOID CLEANUP

- リビューの目的を、
欠陥を取り除き修正することとしては
いけない

- ◆ バグゼロの法則
- ◆ 優秀なレビューアーがどんなに頑張っても
2割から6割程度しか見つからない

やり直して
“元から”ドキュメントの品質を上げてもらう

Engineer Your Review Process : Tom Gilb 2008

SONY

ソフトウェアテストシンポジウム JaSST '09 Tokyo copyright (c) 永田 敦

コスト：Agile Inspection

- 対象
 - 1ページ(300文字程度)
 - 1時間(フルのルールを使って)
- 人
 - インспекタ 数名
 - モデレータ 一人
- 例
 - インспекタ
 - ◆ 人数×時間×賃率=コスト

SONY

ソフトウェアテストシンポジウム JaSST '09 Tokyo copyright (c) 永田 敦

コスト：インспекション

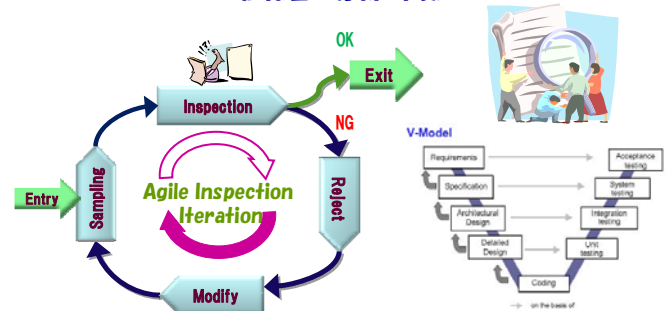
- 開発予算の10～15% Tom Gilb: “ソフトウェアインспекション”より
- 例
 - 時間
 - ◆ ?時間/回
 - ◆ ?分/件
 - 人
 - ◆ リーダー 1名
 - ◆ モデレータ 1名
 - ◆ インспекタ 6名
 - ◆ 提案者 2名
 - 工数
 - ◆ インспекション
 - 人数×時間×賃率
 - ◆ 読み

SONY

ソフトウェアテストシンポジウム JaSST '09 Tokyo copyright (c) 永田 敦

位置づけ

テストプロセスを始めるかを決める指標 仕様書の分析の開始



SONY

ソフトウェアテストシンポジウム JaSST '09 Tokyo copyright (c) 永田 敦

さてそれでは

- 従来との折り合い
 - モチベーション
 - プロセス改善
導入できるか?
 - ◆ Exit Criteria
 - ◆ テーラリング
 - ルール
 - 標準
 - 独自のルール
 - メトリクス
 - メトリクスのテーラリング
 - 表現
 - 書き方
 - Planguage
 - スキル
 - トレーニング
- Agile Inspection
ワークショップで
いっしょに
考えてみませんか?

SONY

ソフトウェアテストシンポジウム JaSST '09 Tokyo copyright (c) 永田 敦

ご清聴
ありがとうございます
ございます

SONY

ソフトウェアテストシンポジウム JaSST '09 Tokyo copyright (c) 永田 敦