

JaSST2006
一ヶタ高い品質のための
ソフトウェアテスト

2006. 1. 31
つくば国際大学
保田 勝通

K.YASUDA Copyright(c) 2005 TIU

1

目次

1. はじめに
2. 普遍的な品質保証の考え方
3. 品質検証の技術
4. 品質保証の仕組みと方式
5. 人と組織

K.YASUDA Copyright(c) 2005 TIU

2

1. はじめに

- 一ヶタ高い品質を獲得するためには
- 問題意識と講義の要点
- 品質管理・ソフトウェア工学の基本を忘れていないか？
- テスト工程の重大な問題点と提言

K.YASUDA Copyright(c) 2005 TIU

3

一ヶタ高い品質を獲得するためには

- 開発の重大な問題点の解決なしに一ヶタ高い品質の実現は無理
- 自らの組織の弱点の認識とフィードバックが重要
- テストだけでは、一ヶタ高い品質の実現は無理
- 設計品質の向上が必須
- 技術と管理は、車の両輪
- 開発全般を通じて、品質管理・品質保証のトータルなアプローチが必要
- 人材(外注先を含む)の育成
- モチベーションの高いチーム作り

K.YASUDA Copyright(c) 2005 TIU

4

問題意識と対策方針

- なぜ品質の良いソフトウェアを期限内に作れないのか？
 - それは開発に対する考え方、仕事への取り組み方が間違っているからである。
 - 設計を手抜きし、テスト工程という名のデバッグ作業にどんなに汗水を流しても、うまくいくはずがない。デバッグとは手戻り作業！
 - 慢性的な長時間残業とモラルの低い職場から良い製品は生まれない。
- それではどうすべきか？
 - ソフトウェア工学に基づく正しい開発方法論により、品質を作りこむこと。
 - 品質管理の原点に戻り、PDCAを回し、改善し続けること。
 - 気力を充実させ、作業に集中できる職場環境を作ること。
- 正しい方針に基づく、充実した、やりがいのある、達成感を味わえる、チームワークのよい楽しい職場から、良いソフトウェアは生まれる。

K.YASUDA Copyright(c) 2005 TIU

5

品質管理・ソフトウェア工学の基本を忘れていないだろうか？

- 1980年代に習得した大規模開発のためのプロジェクト管理、高信頼性達成のための品質管理やソフトウェア工学の適用経験は、伝承されているか？
- 開発環境が変化し、大規模開発が激減した1990年代の空白をどうやって取り戻すか？

K.YASUDA Copyright(c) 2005 TIU

6

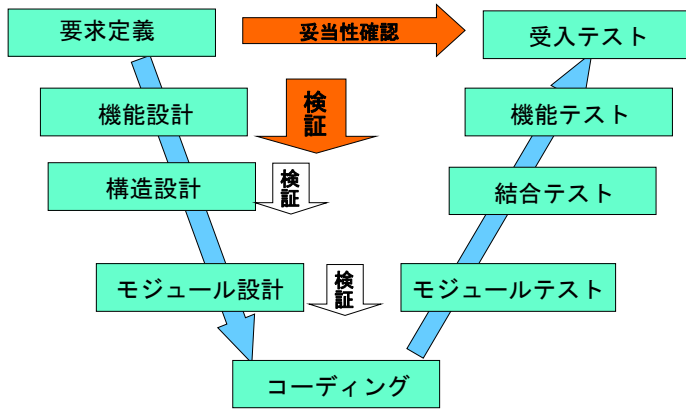
テスト工程の重大な問題点と提言

- テスト工程の実体は、テスト・**デバッグ**工程である
- **デバッグ作業は手戻り作業**
- テスト工程は全工程の40-50%を占めるが、その過半数はデバッグ作業
- この実績を元に、標準日程や標準工数が設定される
 ➡ 現状は、設計の検証(レビュー)を不当に短縮し、テスト時のデバッグ作業を正当化
- **設計とその検証(レビュー)日程・工数を増加し、テストのデバッグ作業を短縮すべき。**
- もう一つの事実:不良修正のミスは20-30%!

2. 普遍的な品質保証の考え方

- ソフト工学の基本原則
 - **V&V(検証と妥当性確認)** ⇨ 2. 1
 - 計測なくして制御なし
- 品質管理の基本原則
 - **不良低減の原則:品質はプロセスで作り込め** ⇨ 2. 2
 - 不良を作り込まない:インスペクション
 - 不良の早期抽出:テスト戦略
 - **品質管理の進め方:PDCA(フィードバック)** ⇨ 2. 3
 - 再発防止策-原因除去策
- **品質保証部門のあり方** ⇨ 4. 4
- ここで重要なことは、知っていることではなく、どこまで実践しているかである。

2.1 V&V(検証と妥当性確認)



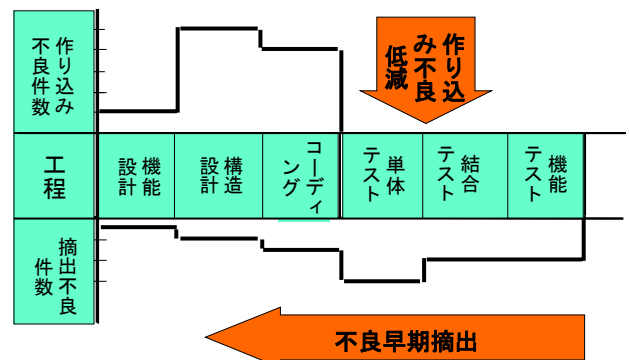
検証の要点

- 当該工程が定義されていること。入力と出力が定義されていること。
- 入力仕様と出力仕様が、明確かつ必要十分に適切な形式や手段で記述されていること。(例)状態遷移図、DFD、UML、擬似コード
- 入力仕様から出力仕様への変換ルールがソフトウェア工学に則った技術と手続きに従っていること。(例)DFDで記述され、段階的詳細化のプロセスが明示されている、チェックリストが整備されている
- 合理的な手段(ソフトウェアインスペクション等)で、タイムリーにレビューが実施されること。

V&V実施手段

- 検証
 - レビュー(デザイン/コード・レビュー、ウォークスルー、インスペクション)
 - ドキュメント検査
 - テスト(テスト項目の作成手法、テスト項目レビュー)
- 妥当性確認
 - 製品検査
 - システム検査(SST)

2.2 不良低減の方針



品質の作りこみとは

- 品質はレビューによってではなく、**設計作業自体によって確保することが基本**。すなわち、成果物(各プロセスの出力である仕様書)作成の工程で「品質を作りこむ」と。
- これは、ソフトウェアエンジニアリングの考え方と適切な技法を駆使して、しっかりした設計を行うということ。
- レビューが重要ではないということではなく、物作りがしっかりしていないとだめだということ
- 素人は犬小屋が作れても、高層ビルは作れないが、強度計算すらできないソフト開発の現場はどうか？デバグと言う名の補強工事の連続！

なぜ品質に焦点をあてるのか？

- 多くのプロジェクトにおいて、計画性のない修正作業は最大コスト要因である(全コストの40%—80%)
- 経済性のために品質に焦点をあてる(上記)
- 品質のために品質に焦点をあてる
- 品質はプロジェクトのなかに計画されていなければならない；最後に付け加えるのではだめ
- 変更は不安定さを増大させる、従って欠陥は早く除去するのがよい。

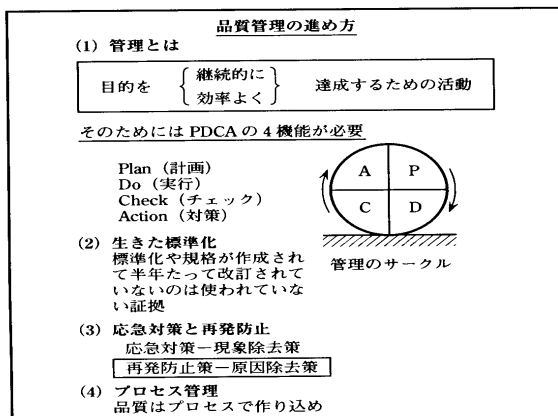
欠陥コストは、欠陥がプロセス内に長く留まれば留まる程、増大する

- 欠陥の作りこみ工程が上流であればあるほど、また欠陥の摘出が下流工程であればあるほど、言い換えれば欠陥の生存期間が長ければ長いほど、欠陥修正コストは増大する
- 要求定義工程で作りこまれた欠陥を要求定義工程内で摘出した場合を1としたとき、要求定義工程で作りこまれた欠陥を出荷後修正すると50から200倍の費用が発生する

2.3 品質管理の進め方

- 管理とは
 - 目的を継続的に効率よく達成するための活動
- 管理のサークル—PDCA
 - Plan—Do—Check—Action
- 生きた標準化
 - 改訂されない規格は使われていない証拠
- 応急対策と再発防止
 - 応急対策—現象除去策
 - 再発防止策—原因除去策
- プロセス管理
 - 品質はプロセスで作り込め

品質管理の進め方(2)



日本的品質管理の特徴—その本質

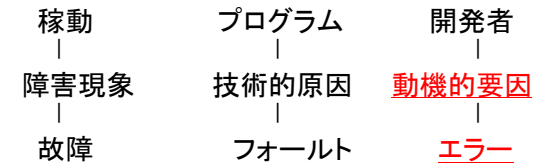
- フィードバック
 - 発生問題点の分析から、再発防止策の実施
 - P-D-C-Aではなく、C-A-P-D
- プロセスの改善
 - 製品改善は対症療法—CMMLレベル3(定義)
 - 舵とり組織(ワインバーグの組織文化パターン3)
 - プロセス改善は体質改善—CMMLレベル4(管理)
 - 予知組織(ワインバーグの組織文化パターン4)

温故知新: PDCAに学ぼう

- 1980年代世界から驚異の眼で見られた日本の発展は、製造業の品質管理の賜物である。
- その原点は、戦後の混乱のなかで製造した製品の不良の山のなかで、一つ一つ不良の原因を掘り下げて改善を積み上げてきた成果である。品質管理の原点であるPDCA、正確にはCAPDをしっかりと回した成果である。

不良分類の考え方

Iレベル → IIレベル → IIIレベル



I 故障(現象)コード: 障害の種類と影響の把握(ユーザ視点)

II フォールトコード: 不良の技術的原因の把握(開発者視点)

III エラー(動機的要因)コード:

人間の知的活動に起因する原因の把握(再発防止・改善の視点)

3. 品質検証の技術

- 不良の作りこみの低減
 - デザインレビュー
 - ソフトウェア・インスペクション
- 不良の抽出
 - テスト戦略

3.1 不良の作り込みの低減

- 適当に設計を済まして、その不良除去を後ろのテスト工程に託すのは、正道ではない。
- クリーンルーム手法
 - 1ステップずつ全員で正しいことを確認して進める。
 - テスト以前に時間をかけ、単体テスト工程は設けない
- 効果的なレビューの徹底
 - クリーンルーム手法を採用しないでも、テスト以前に全不良の80%をレビューで除去することは可能。
 - ソフトウェア・インスペクションの採用

「レビューの形骸化」という現実に対して

- レビューの効果がでない
 - 不良を分析することにより、どういうレビューをすれば、不良が見つかるかが分かる。
 - レビューで不良を絶対見つけるという強い意志をもつこと。
- レビュー時間の確保が困難
 - 設計・レビューと、テスト・デバグの時間比率がおかしい。もっと設計・レビューに時間をかけて、不良を少なくすれば、デバグと再テストの時間は少なくてすむ。

ソフトウェア・インスペクションー概要ー

- IBM社のFaganが提唱、レビュー方式をウォークスルーよりも明確に規定。
- ここでは、その後の改善を盛り込んだ次の書籍をベースに紹介
 - T.Gilb他著、伊土他訳:ソフトウェアインスペクション、共立出版、1999
- インスペクションは、以下の2つで構成される
 - 成果物インスペクション
 - プロセス改善

ソフトウェア・インスペクションの特徴

- インスペクション実施責任者のリーダー(モデレータ)の役割大
- 方法
 - リーダーは、**準備不足と判断した場合開始せず**
 - キックオフミーティング: チームの訓練と動機付け
 - 個人チェック: **参加者は事前に問題点を抽出**
 - チェック方法: **レビュー対象仕様書と元の入力資料を比較・レビュー**
 - ロギングミーティング: 抽出問題点を報告記録し、それをもとにレビュー。**相乗効果で新たな指摘を誘導。**

なぜいつも時間が足りないか？

- 上流工程で時間をかけず、手抜きをするから、後でツケが回ってくる。設計に納得のいくまで時間をかければ、不良が減り、下流は工数も少なくてスムーズに終了する。

3.2 不良の抽出—テスト戦略

- 「KKDに頼ったテスト」という現実
- テスト戦略

テスト戦略

- 各テスト工程ごとに不良抽出目標値を設定
- コード・レビューの重視(70-80%の不良抽出)
- 系統的なテストの実施
 - ブラックボックス・テストとホワイトボックス・テストの併用
 - 適切なテスト基準を持つテスト項目設定技法の採用
 - テスト・カバレッジ尺度の使用
- テスト資産・出力の再利用とテスト支援ツールでの省力化
- 顧客環境を擬似するシステムテストの実施

系統的なテストの実施

- ブラックボックス・テストとホワイトボックス・テストの併用が必須
- ブラックボックス・テストの特徴
 - 長所: ユーザの立場からのテストが可能
 - 短所: 内部仕様の特異な処理のテストが困難
- ホワイトボックス・テストの特徴
 - 長所: 内部仕様に基づく網羅的テストが可能
 - 短所: 外部仕様に規定されていないながら、実現されていない部分のテストが漏れる

テスト項目設定の問題点

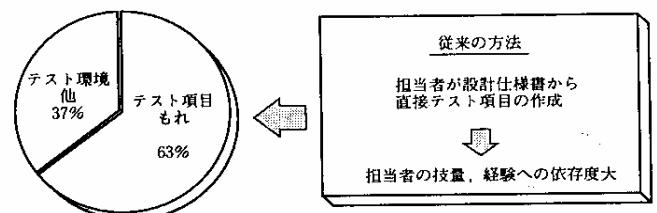


図 5.5 テスト項目設定の問題点[B1]

テスト項目作成の重要性

- 社外事故のテスト不良要因の第一はテスト項目漏れ
- これは、テスト項目作成作業の軽視・認識不足が原因
- テスト項目作成には高度な技術と豊富な経験が必要
- 適切なテスト項目設定技法の採用
- 社外事故の再発防止策に基づくテスト項目ノウハウの伝承
- 業務知識やユーザの使用方法の吸収と反映

ブラックボックス・テスト項目設定技法

- 第一の問題は、外部仕様が厳密・網羅的に記述されていないこと
- 特に正常仕様の記述中心で、限界・境界および異常時の仕様が不明確
- この対策は仕様記述が文章表現の場合は決定表、状態遷移図、状態遷移表/ステータスマトリックス等で整理すること(根本対策は仕様書記述を正確な図表記述にすること)

ブラックボックス・テスト項目設定技法(続き)

- 第二の問題は、すべての組み合わせ項目をテストするのは非効率というよりも現実的ではない。
- この対策は不具合を検出できる最小組み合わせ項目セット導出技法の採用→原因結果グラフ技法(組み合わせ論理に有効)
- 現実的対策は、組み合わせの基準を設けて項目を設定する方法
- 要因分析法では実験計画法の直交配列表を応用し、「任意の二つの因子間で全状態の組み合わせが一つ以上存在する」基準に基づく項目を設定

ホワイトボックス・テスト項目設定技法

- 制御構造に着目しテストカバレッジ尺度の判定条件(分岐)網羅(C1)基準に基づき、項目を設定する方法が一般的
- データ構造やデータフローに着目し、データ階層の親子関係の組み合わせに対して、データ要素の設定・参照を網羅する方法もある

テスト項目作成基準

表 5.3 テスト項目作成基準 [B3]

区分	構造モデル	テスト項目作成基準の例	
		仕様記法例	テスト項目作成基準例
外部仕様	空間モデル	原因-結果グラフ	原因-結果グラフ技法
		決定表	実験計画法による技法
	時間モデル	状態遷移図	AGENT 技法
内部仕様	制御構造モデル	テストカバレッジ尺度の適用 C0: すべての命令文を実行 C1: 命令の全分岐方向を実行 M0: すべてのモジュールを実行 M1: モジュールの全コール関係を実行	
	データ構造モデル	D0: すべてのデータ要素を設定, 参照 D1: データ階層の親子関係の組合せですべてのデータ要素を設定, 参照	

4. 品質保証の仕組みと方式

- 品質保証の構成要因
- 品質第一主義の原則
- 検査の定義
- 品質保証部門のあり方
- 開発工程と品質保証部門の役割
- ドキュメント検査
- テスト工程の品質目標値管理
- 製品検査
- 稼動品質管理と品質向上活動

4.1 品質保証の構成要因

- 品質第一主義(CS)の理念
- 独立した品質保証部門と検査
- 標準開発手順
- 生産技術とCASEツール
- 開発環境・設備・ピープルウェア
- 生産管理(品質管理・工程管理・プロジェクト管理)
- 躰と教育・訓練

4.2 品質第一主義の原則

- (品質重視の)企業理念
- (品質に対する)経営者の積極的指導
- (設計部門から)独立した品質保証部門による製品検査
- 小集団活動(QCサークル)
 - 従業員の自主的な改善活動

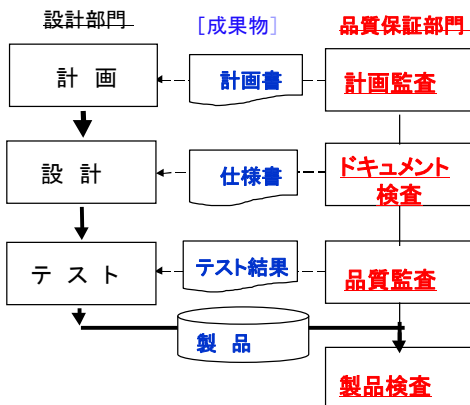
4.3 検査の定義

- ソフトウェアをなんらかの方法で試験した結果を品質判定基準と比較して、ソフトウェアの良品・不良品の判定を下すこと(JIS)

4.4 品質保証部門のあり方

- ◆ 品質保証部門の組織: 製品の合否判定を公正にするため、**設計製造部門から独立した組織**にする
- ◆ 品質保証部門の職務:
 - **顧客の立場に立った検査を行い、所定の品質に適合しない製品を合格、納入させないこと**
 - 製品納入後も、顧客の立場にたつて、品質を保証するための活動を行うこと
- ◆ 品質保証部門の活動範囲
 - 社内: ・工程別検査
・工程全般を通じて品質管理の総まとめ
 - 社外: フィールド保証活動・顧客ニーズ、クレームの窓口

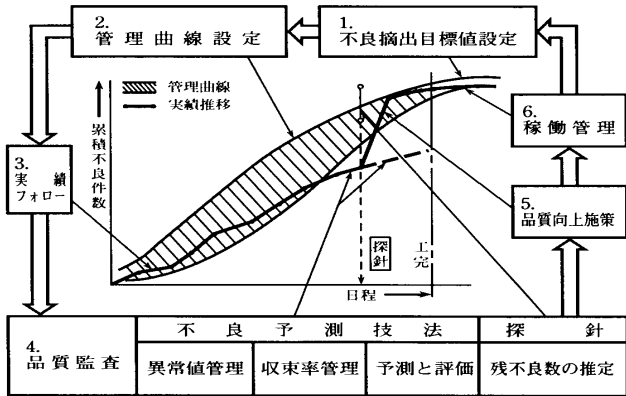
4.5 開発工程と品質保証部門の役割



4.6ドキュメント検査

- 開発用ドキュメントの検査
 - ◆ 対象ドキュメント: 機能仕様書/設計仕様書
 - ◆ 検査の狙い
 - ・良いドキュメントが良いプログラムを作る
 - ・設計工程の品質管理 (中間工程検査)
 - ・設計工程進捗管理精度向上
- 顧客提供ドキュメントの検査
 - ◆ 対象ドキュメント: マニュアル類
 - ◆ 検査の狙い
 - ・マニュアル製品品質の保証
 - ・開発ドキュメントと整合性
 - ・「理解性」や「使用性」の評価

4.7 テスト工程の品質目標値管理



K.YASUDA Copyright(c) 2005 TIU

43

探針

- 目的:
テスト工程(設計部門)に対する、品質保証部門による品質の計測と予測
- 方法:
(1) **抜き取りテスト**(検査項目の10-20%)
(2) 品質保証部門が実施
(3) 結果分析に基づく:
 - ・ **未摘出不良件数の予測** [量的]
 - ・ **設計テストの弱点指摘** [質的]

K.YASUDA Copyright(c) 2005 TIU

44

探針の手順

- 探針とは、設計・開発部門のテスト工程中のソフトに対して、検査(品質保証)部門が検査項目をサンプリングテストし、その結果から残存不良を推定する手法である。
- 探針項目として、検査項目の10~20%を層別抽出法でサンプリングする。
- 探針項目からでた不良に対して標本不良率を求める。
標本不良率 = 不良件数 / 探針項目数
- 二項確率紙または近似式により、母不良率の信頼限界の上限値Puと下限値Plを求める
- 母不良率から残不良件数を求める
上限推定不良件数 = 総検査項目数 × Pu
下限推定不良件数 = 総検査項目数 × Pl
- 詳細は参考文献[保田勝通: ソフトウェア品質保証の考え方と実際, 日科技連, 1995] P.85-87を参照。

K.YASUDA Copyright(c) 2005 TIU

45

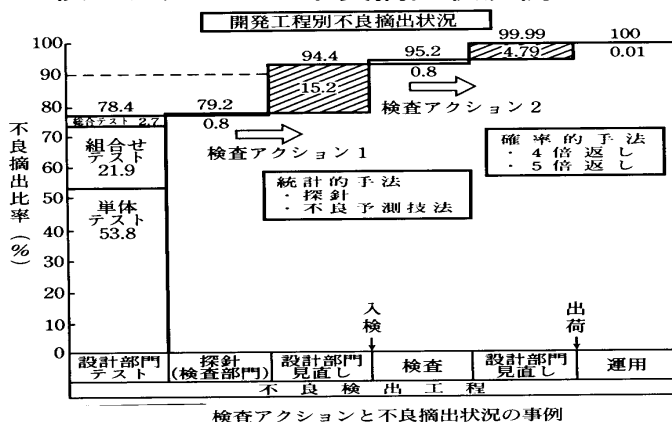
4.8 製品検査

- 目的: 製品がユーザ使用に耐えうるか否かの観点で出荷可否を判定
- 方法: (1) ユーザ視点に立った検査の項目と環境で検査
(2) 品質保証部門が自らマシンを使用して設計部門とは独立に実施
(3) **検査で不良を1件でも摘出すれば、不合格で設計に差戻し**
- 要点: (1) **製品検査の開始時に、検査対象製品が十分な品質に達しているかどうかを見極めること**
(2) **検査で不合格差戻しの場合、品質評価に基づき、設計部門に類似不良の摘出を義務づけること**

K.YASUDA Copyright(c) 2005 TIU

46

検査アクションと不良摘出状況例



K.YASUDA Copyright(c) 2005 TIU

47

なぜ検査と出荷停止権限は必要か

- 納期・コスト・品質のなかで、納期とコストは誰でもオーバーか否かは数値で分かる。
- 品質は、現在検出された不良以外は予測である。検出済不良を直せば判明不良はゼロ!
- 独立した検査(品質保証)部門の予測に基づく、出荷停止権限がなければ、不良内在のまま、出荷される可能性が極めて高い。

K.YASUDA Copyright(c) 2005 TIU

48

なぜ独立した品質保証部門は必要か

- 納期が近づくと、経理部門、工程管理部門、営業部門、さらにプロジェクトマネージャや場合によっては経営者も、コストと納期厳守を迫ってくる。
- 上記の通り、QCD(品質、コスト、納期)のなかで、品質だけが予測であって実績数値ではない。
- 組織力学のせめぎ合いに抗していくには、出荷停止権限をもつ独立した品質保証部門は必要不可欠

システム テスト(SST)

SST: システム・シミュレーション・テスト

- SSTの目的と効果
- SSTのテスト方式

SSTの目的と効果

■ SSTの目的

- ◆ ユーザ環境を工場内に構築してシステムをテストすることにより、システム品質を出荷前に保証する
- ◆ システムの妥当性確認
 - ユーザの運用シナリオを実施し、業務遂行が可能であることを確認する
- ◆ 実ユーザ環境ではシミュレート困難な下記の条件をテストすること
 - ・将来、拡張され得るユーザ環境 下での過負荷や性能テスト
 - ・各種障害シミュレータ(入出力、 回線障害)による障害テスト

■ SSTの効果

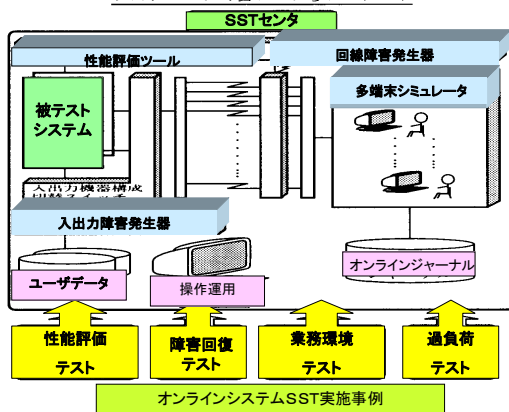
- ◆ 出荷直後の初期不良を大幅削減
- ◆ 顧客の安心感大

SSTのテスト方式

—大規模オンラインテスト事例

- 過負荷テスト
 - 実回線使用のCPU対向多端末シミュレータ
- 障害回復テスト
 - 各構成機器・ネットワークに対する障害シミュレータ
- 性能評価テスト
 - 過負荷テストの環境に性能評価ツール適用
- 業務環境テスト
 - ユーザデータ、オンラインジャーナル、業務運用操作

S S T の 概 念 図
システム シミュレーション テスト



4.9 稼動品質管理と品質向上活動(1)

■ 障害対策 —当該品対策と既納品対策—

- 当該品対策: 顧客の影響を最小限にすることを第一に、迅速かつ確実に
- 既納品対策: 同件障害の発生防止のため、迅速、安全かつ簡単な手段で
- 再発防止策
発生障害の作りこみ要因及び見逃し要因を、技術的要因に留まらず、開発者側の動機的要因まで掘り下げ、同一要因に起因する類似不良の根本原因を除去
 - 類似不良防止策 = 作りこみ・抽出プロセスの改善

稼働品質管理と品質向上活動(2)

- 再発防止検討会議の定期的開催
 - 信頼性に対する幹部の関心と姿勢が重要
 - 幹部自らも学ぶ場
 - 再発防止策を追究する教育の場
- 稼働実績に基づく製品及び開発プロセスの改善
 - 顧客の立場に立った稼働品質指標の活用
 - 稼働実績データベースの構築と活用
 - 製品開発プロセス／手順、製品構造の要改善箇所の解析
 - 障害顕在化以前の予防保守アプローチ
- ユーザミス、ユーザ要望から学ぶ
 - 仕様や構造の改善に結び付ける

5. 人と組織

- 人材育成が最も大切
- 外注先の育成:「あゆみ」活動の紹介
- ソフト開発組織の文化水準

人材育成が最も大切

- 躰教育
 - 安全確認の「指差称呼」の重要性(身体で覚えて確認する習慣)。(例)キーインしたら、確認。
- 日常業務教育
- 技術教育
 - 大学教育が不十分、あるいは受講していない
 - 企業内教育はきわめて不十分
 - ソフトウェアエンジニアリング教育
 - プロジェクトマネジメント教育
 - 対象領域教育等
 - 教育投資が必要
- 自己研鑽も必要

外注先の育成:「あゆみ」活動の紹介

- 下記の名著をぜひ熟読されたい。
 - 富士通通信ソフトウェア開発部編:富士通における「あゆみ」活動—高品質ソフトウェア開発への挑戦—,日科技連,1992
- 自分の組織で、ここに書かれていることを実行することが先決。自らが出来ないことを外注先に要求することはできない。
- 「あゆみ」活動は、ソフトウェア開発のプロセス改善活動を協賛会社と共に推進していく仕組みである。この仕組みは受け入れ検査、品質監視、品質連絡会、品質管理技術開発の4つの核からなる。
- 「あゆみ」活動では、現場にノウハウを求めて、データでものを見て、現場にフィードバックさせることを体系化して、開発現場の実態に合わせながらレベルアップを続けている。QC手法がソフトウェア分野でも適用できることを見事に証明してみせた。

ソフト開発組織の文化水準

- システム開発を長期的に制御できるのは、ワインバグの組織文化パターンの「舵取り文化」以上(CMMのレベル3相当—フィードバック制御可能)
- 文化水準向上には、以下の性格を持つ組織風土であることが必要。
 - 開示性(Openness)
 - 誠実性(Honesty)
 - 進取性(Learning)
 - 出典:大野尙郎氏語録

おわりに

一銀の弾丸はない、しかし道はある

- 技術にせよ管理手法にせよ、単独で10年間にソフトウェアの生産性や品質を飛躍的に改善するものはない。
- しかし、道はある。
- 人間がすべて(そう、だいたいのところは)
 - 成功のためには、人々の質およびその組織形態と管理こそがツールや技術的アプローチよりもはるかに重要な要因—F.ブルックス「人月の神話」
- 生産性は品質に「ついてくる」
 - C.ジョーンズ「ソフトウェアクライシスの評価と制御」

参考文献-講演内容対応

- 保田勝通:ソフトウェア品質保証の考え方と実際,日科技連,1995
- 石川馨:日本の品質管理,日科技連,1981
- F.P.ブルックス Jr.:人月の神話,アジソンウェスレイ,1996
- T.Gilb他著,伊土他訳:ソフトウェアインスペクション,共立出版,1999
- 富士通通信ソフトウェア開発部編:富士通における「あゆみ」活動—高品質ソフトウェア開発への挑戦—,日科技連,1992
- G.M.ワインバーグ:ワインバーグのシステム思考法,共立出版,1994
- W.S.Humphrey著,藤野監訳:ソフトウェアプロセス成熟度の改善,日科技連,1991
- C.Jones著,富野監訳:ソフトウェア品質のガイドライン,共立出版,1999
- G.J.Myers著,長尾訳:ソフトウェア・テスト技法,近代科学社1980
- T.デマルコ/T.リスター,松原友夫/山浦恒央訳:ピープルウェア第2版,日経BP社,2001

K.YASUDA Copyright(c) 2005 TIU

61

参考文献-設計法対応

- 野口健一郎:ソフトウェアの論理的設計法,共立出版,1990
- 片岡雅憲:ソフトウェアモデリング,日科技連,1988
- 片岡雅憲:ソフトウェア再利用技術,日科技連,1992
- 二村良彦:プログラム技法—PADによる構造化プログラミング—,オーム社,1984

K.YASUDA Copyright(c) 2005 TIU

62

参考文献-テスト法対応

- B.Beizer著,小野間他訳:ソフトウェアテスト技法,日経BP,1994
- C.Kaner他著,テスト技術者交流会訳:基本から学ぶソフトウェアテスト,日経BP,2001
- リー・コーブランド著,宗雅彦訳:はじめて学ぶソフトウェアのテスト技法,日経BP,2003
- C.Kaner他著,テスト技術者交流会訳:ソフトウェアテスト293の鉄則,日経BP,2003

K.YASUDA Copyright(c) 2005 TIU

63

付録

- ワインバーグの組織文化パターン
- 銀の弾丸はない,しかし道はある
- DOD推奨「ベストプラクティス」項目
- DOD推奨「ワーストプラクティス」警告

K.YASUDA Copyright(c) 2005 TIU

64

ワインバーグの組織文化パターン

- 組織文化パターンの定義
- 組織文化パターンの見分けかた

K.YASUDA Copyright(c) 2005 TIU

65

ワインバーグの組織文化パターン(1) —定義—

- 0. 無意識:プロセスを実行していることに気付いていない
- 1. 可変:いままさにやりたいことは何でもやる
- 2. 慣習:(パニックのとき以外は)慣習に従う
- 3. 舵とり:結果によって慣習を選別する
- 4. 予知:過去の経験に基づいて慣習を確立する
- 5. 適合:だれもがいつもあらゆる改善に没頭している

K.YASUDA Copyright(c) 2005 TIU

66

ワインバーグの組織文化パターン(2) ー見分けかたー

- 0. 無意識:組織が病んでいるの知らない(無自覚)
- 1. 可変:病気を避けがたい人生の現実として扱う(宿命)
- 2. 慣習:病気を罪悪のように扱うために症状を隠したがる(隠蔽)
- 3. 舵とり:組織が健康であること(プロセス)よりも病気を治すこと(製品)に注力する(治療)
- 4. 予知:時に病気のプロジェクトを抱えてもすぐ治る(予防)
- 5. 適合:病気に対して免疫がある(免疫)

ー銀の弾丸はない、しかし道はあるー

- 技術にせよ管理手法にせよ、単独で10年間にソフトウェアの生産性や品質を飛躍的に改善するものはない。
- しかし、道はある。
 - 購入して構築する(パッケージソフトの活用)。
 - コンセプトの完全性とアーキテクト(偉大なデザイナー)
 - インクリメンタルモデルの採用
 - 要件確立のためにラピッドプロトタイピングの使用
 - 人間がすべて(人の質と組織形態と管理)
 - ーF.ブルックス「人月の神話」

DOD推奨「ベストプラクティス」項目

- 体系的なリスク管理の実施
- インターフェース仕様の確定
- 正式なインスペクション(レビュー技法)の実施
- メトリクスに基づいたスケジューリングと管理
- 細分化した作業単位での二値(終了/未終了)管理
- プロジェクト全体の進捗予実績管理の可視化
- 品質目標の設定と欠陥の追跡
- 構成管理の徹底
- メンバーの士気の向上と知識・成功体験の共有

DOD「ワーストプラクティス」警告(1)

- 日程に余裕がないという理由で新技術を導入するのは不可
- RFP(Request For Proposal)のなかで実装技術を規定するな
- 確かでない「銀の弾丸」の擁護をするな
- 機能を大幅に減らさない限り、大幅な納期短縮は期待できない
- クリティカルパス上の作業項目をプロジェクト制御外に置くな

「ワーストプラクティス」警告(2)

- 過去の実績以上の大きな改善成果の達成を安易に期待するな
- 複雑だからという理由でハードウェア機能・処理を安易にソフトウェアで処理するな
- ハード・ソフトの機能分割は、ソフトウェアの専門的技術なしに実施するな
- フォーマルレビューの参加者が5人を超えると、その有用性は参加者数に反比例して減少する。