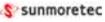




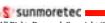

テスト専任チームの事例紹介 ～ サントリーグループにおける機能外テストの実施～



株式会社サンモアテック
 商品サービス開発部 加藤芳彦

1

サントリーグループの組織構造



 All Rights Reserved, Copyright (C) SUNMORETEC Co., Ltd.

2

背景

	～ 2000	2001	2002	2003	2004	2005
Java	研究・調査	パイロット開発	コンポーネント開発			
標準化		Java・オブジェクト指向の標準化作業 (フレームワーク、フレーム、設計手法、プロジェクト管理、品質管理 etc)				
自動テスト		Webアプリケーションに自動テスト採用				
		NOMAD (4GL) のQAのため自動テスト実施				



 All Rights Reserved, Copyright (C) SUNMORETEC Co., Ltd.

3

アプリケーションの品質向上のための取り組み

コンセプト	要求・分析	設計・実装	移行
	デザインレビュー (分析・設計モデル、画面デザイン)	機能受入テスト (統合テスト済みモジュール)	
	アーキテクチャレビュー (アーキテクチャ設計書)	コードインスペクション (最終ソースコード)	受入プログラム評価 (Javaソースコード)
	Check Style + Find bug	IBM Rational Robot	機能外テスト (プロトタイプモジュール、機能テスト済みモジュール)
			メトリクス (Javaソースコード)

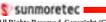


 All Rights Reserved, Copyright (C) SUNMORETEC Co., Ltd.

4

機能外テストの種類

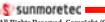
- パフォーマンステスト**
 - 所定の状態においてレスポンスタイムを測定する
 - アウトプット: レスポンスタイム (平均、最長)、スループット
- ストレステスト**
 - 設計処理能力を超えるような多数の処理 (高負荷) を、一定時間行なった場合に、正しく動作することを確認する
 - アウトプット: 発生したエラー、レスポンスタイム (平均、最長)、スループット
- ロングランテスト**
 - 長時間に渡って通常負荷をかけ続けることで、レスポンスの悪化、メモリークの発生が無いかどうかを確認する
 - 通常負荷: 実運用でかかる程度の負荷
 - アウトプット: レスポンスタイム (時間推移)、サーバのリソース情報 (CPU使用率、メモリー使用率)



 All Rights Reserved, Copyright (C) SUNMORETEC Co., Ltd.

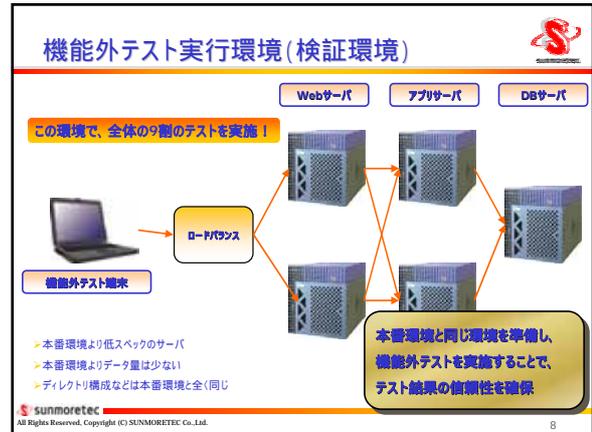
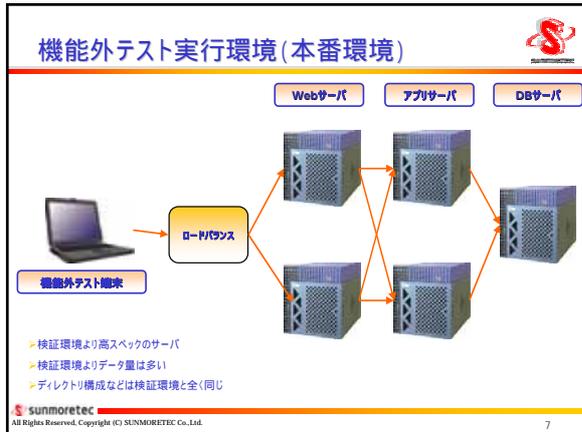
5

機能外テスト実施体制



 All Rights Reserved, Copyright (C) SUNMORETEC Co., Ltd.

6



機能外テストの作成方法

- ※ IBM Rational Robotで作成
 - ☑ テストケース記述書に従い、スクリプトを作成
 - ☑ 実際に対象システムを起動し、操作して、それを記録する形態
- ※ テストケースの内容の不備・提案なども実施
 - ☑ 他システムで発生した不具合を確認できるテストケース
 - ☑ 抜けやすいテストケース
 - ☑ etc

機能外テストのため、対象システムの詳細な仕様を理解する必要がない!

テストパターンのノウハウ蓄積

sunmoretec
All Rights Reserved, Copyright (C) SUNMORETEC Co., Ltd. 9

機能外テストの実行方法

- ※ IBM Rational Robotで実行
 - ☑ テスト準備書に従い、実施
 - ☑ 多重度、実施内容(テストの種類)、負荷の度合い
- ※ パフォーマンステスト
 - ☑ 多重度1、通常負荷、ピーク時の負荷で実施
- ※ ストレステスト
 - ☑ ピーク時の2倍程度の負荷で実施
- ※ ロングランテスト
 - ☑ 通常負荷で、一晩(約15時間)か週末(約63時間)実施
- ※ 共通
 - ☑ 実運用を想定して、記録時と同じ待ち時間で、仮想ユーザー一人一人ずらして実施
 - ☑ 待ち時間0、および仮想ユーザ全員同時実行も可能
 - ☑ 各仮想ユーザーのログインID、および入力データを、CSVファイルを用いて別々に、およびランダムに実施することも可能
 - ☑ 実施中、各ログを監視

**テスト実行の効率化
実運用を意識した実施**

sunmoretec
All Rights Reserved, Copyright (C) SUNMORETEC Co., Ltd. 10

機能外テストの結果報告

- ※ 発見した不具合を「問題点連絡表」にて報告
 - ☑ 許容レスポンスタイムを満たさなかった測定ポイント
 - ☑ 発生した不具合
- ※ 最終結果は「テスト結果報告書」にて報告
 - ☑ 各測定ポイント×各多重度単位で報告
 - ☑ 実施回数、標準偏差、平均レスポンス時間、最大レスポンス時間、スループットなどを報告
- ※ 不具合に対して対策が取られなければ受け入れない

発生頻度の高い不具合の情報蓄積

sunmoretec
All Rights Reserved, Copyright (C) SUNMORETEC Co., Ltd. 11

機能外テストの実績

- ※ 2004年8月末現在
 - ☑ 2004年4月よりサービススタート
- ※ 対象システム数 = 20
 - ☑ ビジネスアプリケーション = 19
 - ☑ 新規案件 = 16
 - ☑ 機能追加 = 3
 - ☑ インフラ = 1 (LDAP認証)
- ※ テストショット数 = 44回
 - ☑ パフォーマンステスト = 23回
 - ☑ ストレステスト = 14回
 - ☑ ロングランテスト = 6回
- ※ 現在も「5システム/月」のペースで実施中

sunmoretec
All Rights Reserved, Copyright (C) SUNMORETEC Co., Ltd. 12

機能外テストの効果



- ☞ 応答性能を保障
 - ☑ 特定日にトランザクションが集中し、かつ、全員が使うようなアプリケーションの応答性能をサービスイン時に保証できた
 - 人事システムの給与明細検索
 - ☑ トランザクションは一定だが、厳しい応答性能を求められるシステムの保証ができた
 - 健康食品通販システムでオペレーターが使う受注機能
 - ☑ どこまで実運用に耐えられるかの計測ができる
 - 生産マスターシステム
 - ・ 最大50ユーザの使用を想定していたが、100ユーザでも実運用可能であることが発覚
- ☞ 機能上のバグを発見
 - ☑ Not Serializable Exceptionの発見
 - Cache容量を超えないと発生しないため、多重度も上げたレスポンステストで発見
 - ・ 少ない
 - ☑ 負分散装置に関する問題の発見
 - 高負荷テストで発見
 - ・ 早い段階で期待していない方(サービ)に処理が割り振られた際の対策がとれた
 - ・ 開発時にはテストできない

機能外テストの波及効果

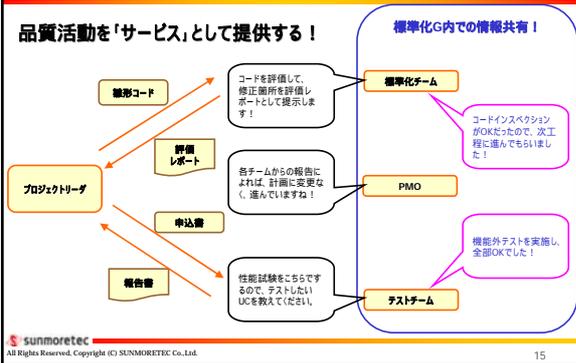


- ☞ PMO機能の活性化
 - ☑ テストサービスの受付をPMOが行うことで、それ以前の評価プロセスである各種レビューやインスベクションといった監査行為を受けないプロジェクトや、進捗を報告しないプロジェクトが激減
- ☞ リスク評価の前倒し
 - ☑ アーキテクチャーが設計された次の段階で、簡単なUCを実装し、機能外テストにけることで、パフォーマンス問題が後に出ないことを確認できる
 - ☑ 機能外テストの専門を作ることで、機能外要求(特に性能部分)に対する設計・対策が以前より早い段階で、さらに「しっかり」実施されるようになってきた
- ☞ テスターのモチベーションアップ
 - ☑ 多くのプロジェクトから頼られることで、それまで以上に誇りを持てるように
- ☞ 人材育成の効果
 - ☑ テストで様々なプロジェクトに抱わり、様々な問題を顕在化させ、解決してゆく過程で、「問題発見・自力解決型」のエンジニアが育ちつつある
- ☞ テストフェーズの確保
 - ☑ テスト専任チームが機能外テストを実施しているため、「スケジュールが押しているので、機能外テストは実施しないやスケジュールが押しているので機能外テストは簡単なものしか実施しない」といったケースを防止できる
 - やっていない「いいわけ」ができない!!

仕組みの定着のために



品質活動を「サービス」として提供する！



更なる挑戦



- ☞ 機能テストの自動化
 - ☑ 機能外テストだけでなく、機能テストも自動化する
- ☞ インフラ環境自体の評価への応用
 - ☑ レスポンステストの結果から、システムの要求を満たすための評価に応用したい
- ☞ 変更時(アプリ・インフラ環境)の影響評価
 - ☑ 機能テスト、機能外テストともに自動化することで、システム変更時、都度、工数を最小限に抑えて、影響範囲を評価していく
- ☞ チューニングノウハウの集積と活用
 - ☑ レスポンステストの結果の対応方法などをナレッジとして蓄積・活用していく