

ソフトウェア開発における品質管理技法 SQVS

3M-1

注) Software Quality Visual System

澤田 三郎*
*日立製作所

山神 敏行*

伊藤 進**

林 英治**

**日立コンピュータエンジニアリング

1 はじめに

ソフトウェアの開発期間の短縮を狙いとして、品質管理方法を改善した。バグは、机上デバッグ、単体テスト等で前倒し抽出するように諸施策を実施しているが、それでも複数プログラム、複数システムとの結合テストの段階でバグが抽出される。

本報告では、結合テスト/検査工程において、品質をビジュアル化(可視化)し適切な対応策を打つことによりソフトウェアの品質を早期に高める管理技法について述べる。

2 ソフトウェア品質管理の考え方(図2.1参照)

結合テスト/検査工程の品質管理において、できる限り短時間で品質を上げるためには、①不良をできる限り前倒しに抽出し、②抽出した不良は、きめ細かくフォローして短時間に対策することが重要である。

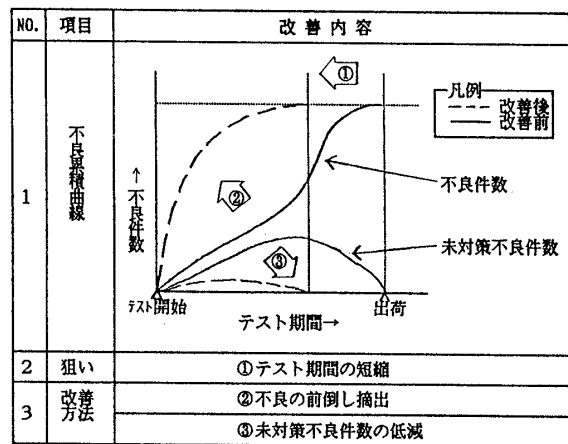


図2.1 改善方針

3 ソフトウェア品質管理のしくみ

3.1 品質管理体制(図3.1参照)

品質管理体制として、①設計部署は品質向上②品質管理部署は品質管理③検査部署は品質評価とした。これは第三者(品質管理部署)が品質管理を業務支援することにより、設計部署にはソフト開発業務に専念してもらうためである。品質管理の運用フローは次の通りとした。

バグ票を基に未対策不良一覧表を作成し、不良対策状況をきめ細かくフォロー・アップする。なお、バグ票は、不良が発生した都度、即時にSQVSに入力する。また、不良を対策するまでの進捗度(ステータス)を更新して行く。

次に、SQVSにより品質分析資料を作成する。これらの資料は、短いターンアラウンド(数時間以内)で作成する。この資料を基に設計取り纏め者は、スピーディーに適切な対応策を検討する。

3.2 品質管理項目の設定

品質管理項目として、①未対策不良件数②不良抽出件数③モジュール別不良密度を設定し、各々の管理項目について①未対策不良一覧表②不良抽出件数推移図③モジュール別不良件数推移表の各品質管理表を作成した。

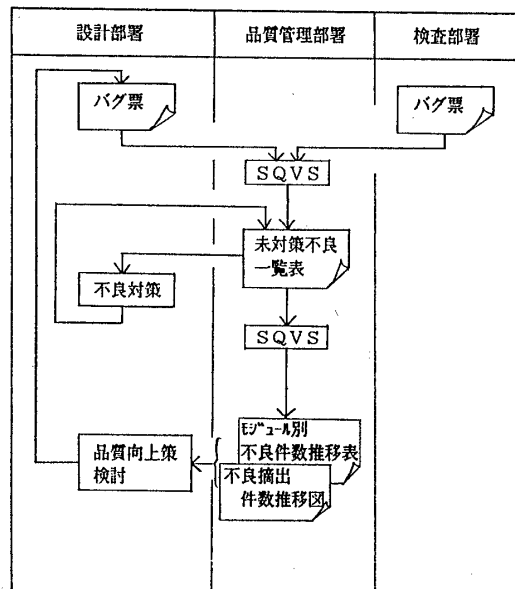


図3.1 品質管理体制

Software Quality Visual System

Saburou Sawada, Toshiyuki Yamagami, Susumu Itoo, Eiji Hayashi
Hitachi Ltd, Hitachi Computer Engineering Ltd.

3.3 品質管理方法

SQVSで出力する主な各品質管理表および品質管理方法の概要を述べる。

(1) 未対策不良一覧表 (図3.2参照)

主な管理項目は、①不良件名②不良区分③重要度④モジュール名⑤担当者⑥不良対策進捗度(ステータス)である。特にステータスは、(A) 不良発生 (B) 原因判明 (C) 対策済 (D) 設計確認済 (E) 検査確認済迄を5段階に分け、きめ細かく管理した。

全不良のうち、未対策分のみを抽出して一覧表を作成する。品質管理部署ではバグ票毎に、ステータス(A)～(E)の完了日を毎日フォロー・アップし記入して行く。

NO	バグ票NO.	不良件名	不良区分	重要度	モジュール名	担当者	ステータス				
							(A)	(B)	(C)	(D)	(E)
1	A-001	A321の終了時の遅延	S	A	SW1	鈴木	1001	1002	1005		
2	B-002	正常処理で13-NO.1発生	S	B	SW2	山田	1010	1012			

図3.2 未対策不良一覧表の出力例

(2) 不良摘出件数推移図 (図3.3参照)

ソフトウェア不良と不良件数合計(ドキュメント不良他を含む)について、不良累積曲線を作成する。不良累積曲線から不良の収束傾向を見る。

図3.3では、ソフトウェア不良が収束傾向にあることを示している。また、ソフトウェア不良のうち未対策不良件数を表示し、不良対策状況を分かりやすくした。

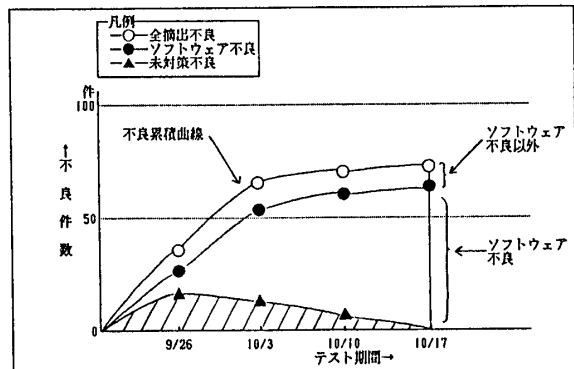


図3.3 不良摘出件数推移図の出力例

(3) モジュール別不良件数推移表 (図3.4参照)

主な管理項目は、モジュールごとの①週別不良摘出件数の推移②不良件数の合計③不良密度である。例えば不良密度が大きく、最近数週間で、不良を多く摘出しているモジュールは、品質向上が必要となる。

品質向上が必要なモジュールについては、最初にテスト項目、不良摘出件数の目標値を設定して品質見直しを実施して行く。

不良件数が多いと予想したモジュールについては、実機テスト工程に進むより前工程である机上見直し工程に戻った方が、品質向上には有効である。

NO	モジュール名	規模(キロバイト)	62/9				62/10				合計	不良密度(件/キロバイト)	不良多発モジュール
			9/26	10/3	10/10	10/17	9/26	10/3	10/10	10/17			
1	SW1	1.5	1	2	1	3	2	0	1	2	13	●	
2	SW2	0.5	0	1	0	1	2	0	1	2	4		

図3.4 モジュール別不良件数推移表の出力例

(4) 効果

本技法をあるプロジェクトについて適用したが、短時間で不良対策が進み、その結果テスト期間を20%短縮できた。

4 SQVSの概要

図4.1にSQVSの構成を示す。

バグ票を入力し、ワークステーション上にデータベース化する。入力後、未対策不良を抽出し、①未対策不良一覧表を出力する。

また、不良データを集計して、②不良摘出件数推移図③モジュール別不良件数推移表を出力する。

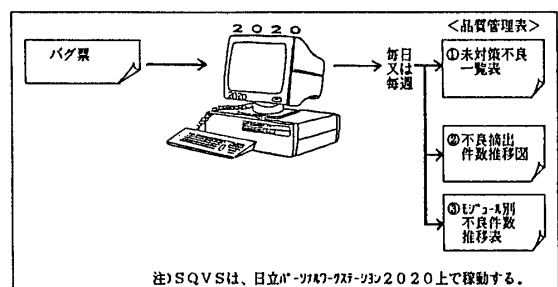
5 おわりに

結合テスト/検査工程において、短期間でソフトウェアの品質を上げる管理技法SQVSを開発した。この技法を複数のプロジェクトに適用し、所期の効果を上げることができた。

なお、本技法を有効活用するには、関連部署の意識合わせと、協力して推進する体制が重要である。今後、不良の内容を含め品質分析を進め、本技法を更に改善して行く予定である。

参考文献

- (1) 石井 「ソフトウェアの検査と品質保証」:日科技連ソフトウェア品質管理シリーズ
- (2) 大場 他 「ソフトウェア品質管理の進め方・施策と各社に見る実際」:応用技術出版



注)SQVSは、日立のワークステーション2020上で稼動する。

図4.1 SQVSの構成