

"Quality Inspection"の実践事例 —プロジェクト上流フェーズにおける熟視テストの実践と効用について—

細川 宣啓†

† 日本アイ・ビー・エム株式会社 サービス事業・品質技術 〒103-8510 東京都中央区日本橋箱崎町 19-21
E-mail: † carvin@jp.ibm.com,

あらまし 本稿はプロジェクト上流フェーズでの成果物熟視テストを通じたコスト効果の高い欠陥除去・予防活動の実践事例を紹介する。高いレベルの品質が要求されるいわゆるライフ・クリティカル・システムや公共性の高いシステムでは、プロジェクト全フェーズを通じた品質検証活動が必須であると言われている。しかし自動化ツールの充実した下流フェーズに比して分析や設計フェーズでの欠陥除去活動は属人性が高く、また時間とコストのかかる手作業の場合が多くなりがちである。これら課題の解決策として従来からある品質インスペクション技法を最適化した事例を紹介する。現代のIT市場に、より多くの品質検証プロフェッショナルが育成されることを切に願う。

キーワード 欠陥除去, 欠陥予防, 上流工程, 熟視テスト, 品質インスペクション

“Agile Quality Inspection” – Best Practice — Adopting “stare test” in project upper phase —

Nobuhiro HOSOKAWA †

† IBM Japan LTD. Service Quality Assurance 19-21 Hakozaiki-cho, Nihonbashi Chuo-ku, Tokyo, 103-8510 Japan
E-mail: † carvin@jp.ibm.com

Abstract This article represents a best practice of defect detection, prevention activities for software project to improve quality. Generally, in “life critical” system development, Quality Inspection activities are indispensable. But in project upper phase (analysis, design phase), these kind of activities tends to be “Skill depend”, “waste much workload”. To solve these problems, IBM Japan LTD tried to launch the QI (Quality Inspection) activities for IT services project. In aspect from “Speed” and “Cost of Quality”, new inspection mechanism was established. QI with optimized tools and process brings you the way to prevent defects and quality improvement as “Speed to market”.

Keyword Defect Detection, Defect Prevention, Upper phase, Stare test, Quality Inspection

1. はじめに

1.1. 上流成果物品質点検の重要性

昨今のIT市場において、高い品質が要求される—いわゆるライフ・クリティカルな—システムや公共性の高いシステムのトラブルが社会問題として報じられている。欠陥の顕在化やパフォーマンスの低下・停止等のシステム・トラブルが原因となり引き起こされる様々な脅威には、単なる工業製品としての位置づけでは済まされないコンピュータ開発の重要性と難しさを再認識させられる。

ところで、このようなシステムの開発プロジェクトでは、プロジェクト全フェーズを通じた品質検証活動が必須であると言われている。テストフェーズや構成管理作業等に代表される開発フェーズ以降の自動化ツールは市場にも多く出回り一般に広く普及しており、また、テストケースの設定や潜在バグの予測等、数学的手法も下流フェーズにおいては確立されているとい

えよう。

これに比べ、プロジェクト上流フェーズでの品質検証／欠陥除去／欠陥予防の分野では、ソフトウェア工学史上の「インスペクション」[1][2]や「ウォークスルー」[3]はその有用性が再評価されているが、現代のプロジェクトにて望まれるスピード感やコスト効果に対してこれらだけが有効かどうか十分に検討されていない。さらに上流での品質検証活動は、顧客要件の充足度、仕様の整合性、後続フェーズのリスク予測など、活動結果の精度がプロジェクト全体に与える影響が極めて大きく、信頼性が要求される。しかし現実には属人性が完全に排除しにくいという課題が内在する。

そこで本稿では、トラブル・プロジェクト発生の未然防止のために設置されたトラブル兆候検知 (“Smoke Sensor”)の専門組織の活動経験を中心にシステム開発プロジェクトにおける効果的な品質向上策について論じる。

2. IBM における品質インスペクション技法「Quality Inspection」とは

日本アイ・ビー・エムにおける QI(Quality Inspection)とは、一般的な品質インスペクションに対しコスト効果を高める様々な最適化を施した Agile な欠陥検出技法である。QI の目的を以下のように定義している。

- (1)ソリューション・リスク最小化
- (2)プロジェクト上流工程の成果物品質向上

具体的には(1)プロジェクト・マネージャ及び品質保証部門と連動したプロジェクトリスクの最小化、(2)要件定義・外部設計フェーズ成果物の“熟視テスト (Stare Test)”の 2 点に主眼を置いた、広範囲なシステム開発プロジェクト適用を目指した技法である。

QI 活動の主な特徴を表 1 に示す。

2.1. QI の実践

2.1.1. 対象 ー何を検証するかー

QI の対象領域は要求定義・設計フェーズにおける成果物である。Ronald A. Radice[4]の示すとおり、上流フェーズで注入された欠陥がもっとも高価で下流肯定での除去が困難であるということは広く知られている。また欠陥検出活動と欠陥除去活動を分離することも、効率的な作業実施に必須であると言われている。QI ではこれらの観点から対象を上流フェーズ成果物に絞って品質検証活動を行っている。

筆者はこの活動中で、この検証対象をプロジェクト成果物、すなわち仕様書のみ絞っている点を強調したい。後に詳述するが、作業効率の観点と成果物作成者の一切の弁明を受け付けないという観点から、成果物の“欠陥検出”のみに集中して作業を行うのである。

当たり前のことだが、埋め込まれた欠陥の除去方法を検討するより、欠陥を埋め込まないように上流工程から品質検証・予防を行うという方針がコスト効果向

上に貢献するという実験結果もある。

2.1.2. 方法 ーいかに検証するかー

QI 組織では熟視テスト (“Stare Test”) を基礎形態として採用している。熟視テストとは専門教育された担当者がプロジェクトの成果物を一つ一つに対して目視によってくまなく欠陥検出する、IEEE 等で定義されている品質検証活動である

特筆すべきは、QI はあくまで欠陥検出を目的としており検出した欠陥を除去あるいは修正する活動を含んでいないという点である。このことは専門教育された QI 担当者がプロジェクトに所属せず第三者的な中立の立場で検出のみに専心して実施するほうがより多くの欠陥を発見でき、最終的に得られる品質レベルは高くなるという実験結果から、このような方法を採用している。

また別の特徴として、全社的な評価基準に基づき各種成果物の欠陥検出を行っている点も強調したい。評価基準は、主要なメソドロジーに準拠したチェックシート形式のツールとして実装しており、工程別に各種成果物に対する質問項目について品質レベルを 5 段階で評価して評点合計から品質向上率を自動算出するものである。最終的に利得金額 (見込み利益) への変換までをこのツールを用いて自動算出する。

2.1.3. 組織 ーどんなチームで検証するかー

QI 活動はその作業チーム編成に工夫を凝らしている。

一般的なインスペクションでは 1 人の運営者と複数人のインスペクター、及び成果物作成者によって実施する。一定時間内に可能な限り多くの欠陥を検出/記録し、対策を立案するまでを対象範囲とする。この形式の問題点は一般にモデレーターと呼ばれる正規に教育されたミーティングの運営者が確保できないこと、及びプロジェクト内部で主要メンバーを招集し、一定時間「拘束」して会議を運営することが難しい点が挙げられる。

<表 1. Quality Inspection の特徴>

Quality Inspection の特徴

- QI は熟視テストによる欠陥検出活動である。
- プロジェクト上流フェーズの最終成果物、中間成果物を対象とする。
- 一定の評価基準に基づく品質の定量化を目指す
- QA 担当者との連携など一定のプロセスに準拠した活動を実施する
- 少人数短期間でコスト効果の高さを狙う。

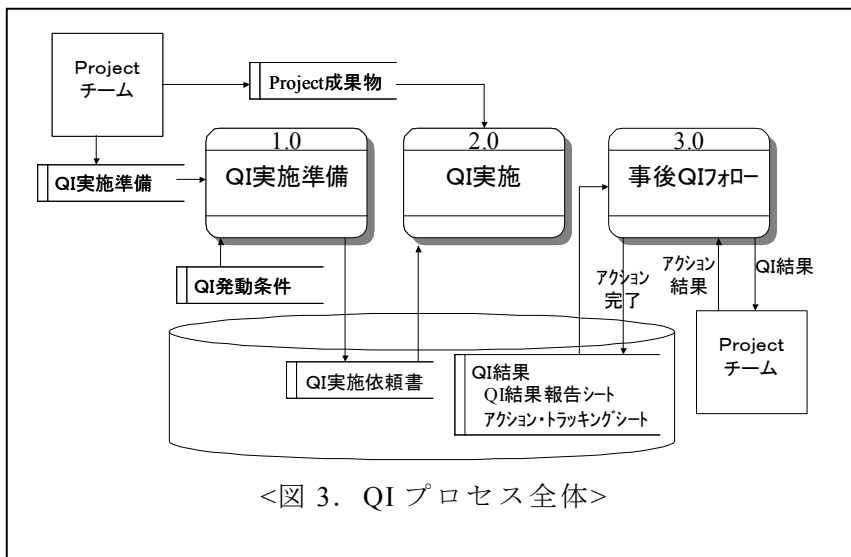
これに対して QI 活動でのチーム編成では、会議の形態を行わない。①リーダーの作業分担指示、②各 QI 担当者が机上で担当範囲を品質検証、③最終的に QI 担当者が成果を持ち寄り報告書形態にまとめるという形式にて活動を行っている。この形式のメリットは並行作業による単位時間当たりの作業効率の向上と、それに伴う 1 プロジェクトあたりの品質検証活動の回数が増加する点にある。

また QI チームでは同時並行で多くのプロジェクトに対して品質検証を行わなくてはならない場合が多い。このような場合には、パスアラウンド形式

(n 人のインスペクターによるベルトコンベア方式の対象成果物をインスペクターで回覧し、課題点や欠陥を検出する形式[4])にて作業を行うように指示している。

組織の最後の特徴として、少人数短期間でコスト効果の高さを狙うフォーメーションを最優先としている点が上げられる。

通常の IBM-QI において対象とする成果物量は 1 件当たり 1000 ページ前後から数ギガバイトまで様々であるが、実施ワークロードとして 1 件あたり平均 8-10 人/日を要している。これら実績数値は一般の品質インスペクション活動と比較すると極めて少ない数値であるが、多くのプロジェクトに対して短期 QI を複数回実施することにより、企業全体の技術レベルの底上げが期待できる、という観点からこのような形態にて運用している。



<図 3. QI プロセス全体>

2.1.4. プロセス

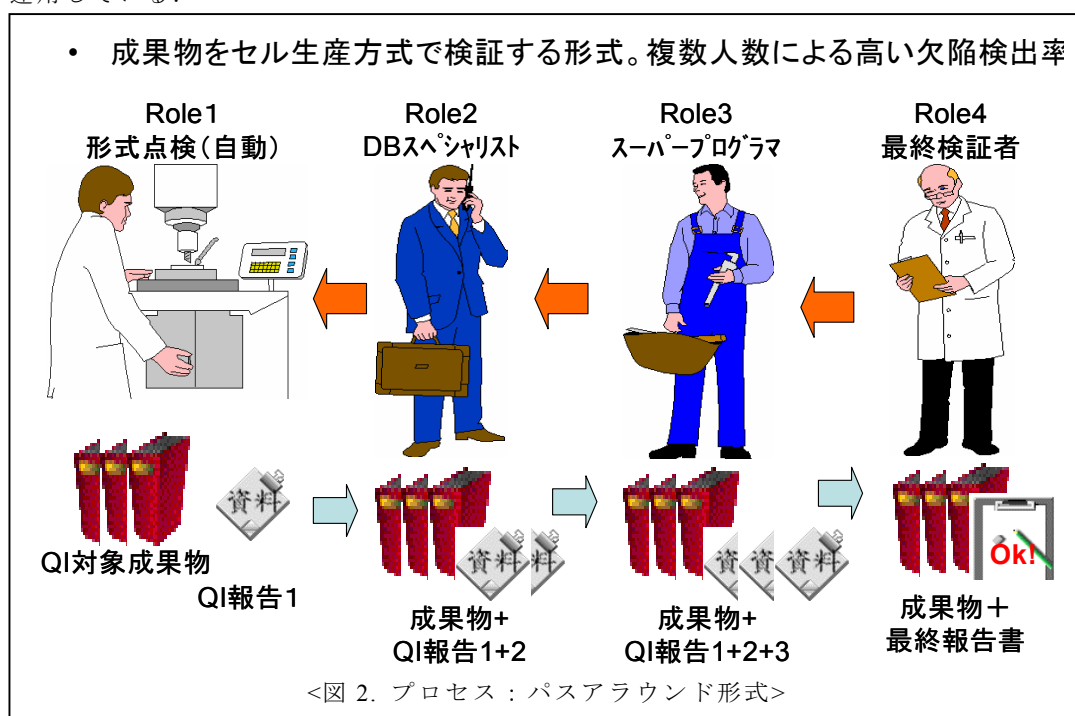
筆者の所属する QI では、プロジェクト独立の第 3 者機関として品質検証を行っている。プロジェクト側にとって負担がかからない方法を採用している。1 人のモデレーターと n 人のインスペクターによって実施する。成果物の作成者（つまりプロジェクト・メンバー）のいない場所で行うこと、また複数人数で実施することにより品質面以外の外因排除、インスペクターの属人性排除や経験・スキル差異の解消も目指す。

はじめにモデレーターとプロジェクト・マネージャや品質保証部門担当者 (QAer) が目的や方針などを確認する。その後、インスペクション実施方針を各インスペクターへ通達。各インスペクターはインスペクション時間内に個人作業を行い、最終的に全員分の QI 結果をまとめる。結果はプロジェクトに対してフィードバックミーティングを行い、事後 QI フォロー（コンサルティング/実施アクションのトラッキング等）が行われる。

更にこれら第 3 者機関による品質検証プロセスを実施するメリットは、高品質成果物の作成だけでなく、品質文化の醸成、若手の育成、短期の技術移転・共有等の効果にも及ぶ。

更にこれら第 3 者機関による品質検証プロセスを実施するメリットは、高品質成果物の作成だけでなく、品質文化の醸成、若手の育成、短期の技術移転・共有等の効果にも及ぶ。

- 成果物をセル生産方式で検証する形式。複数人数による高い欠陥検出率



<図 2. プロセス：パスアラウンド形式>

2.2. QI 実践のコツ

2.2.1. 成果物品質の評価基準

以上、様々な面からインスペクション“方法”の最適化について概観したが、ここではコスト効果の高い QI を実施するための様々な“内容”に関する工夫について解説する。

一般に、インスペクターの教育コストが品質管理を実践する上での障壁となると言われている。QI 組織では前掲のツール適用とあわせ、図 4 に示す 7 つの成果物品質基準に基づくインスペクションを推奨し、属人性の排除と QI 品質の一層の向上を目指している。

これらの品質基準に従えば、例えば、大規模プロジェクトにおいて大人数で大量の成果物を作成する場合には、統一性や整合性の欠陥を中心として、またプロジェクト期間が長くなる場合には追跡可能性上の欠陥を中心として QI を実施するなど、作業の方向付けや方針の重み付けが行いやすくなる。

(1)統一性 (Integrity)

ひとつのドキュメントの中に矛盾がない

(2)整合性 (Consistency)

複数のドキュメント間に矛盾がない

(3)理解の容易性 (Understandability)

ドキュメントの全体の体系が明確である
作成者以外にも一意に理解できる記述である

(4)追跡可能性 (Traceability)

上流での仕様(例:要求事項など)が
後続仕様書に反映されていることが理解できる

(5)保守性(Maintainability)

仕様そのものと仕様書書式,両面の変更容易性

(6)安定性(Stability)

要件の確定度合い, 変更の反映程度

(7)バランス(Balance)

設計のトレードオフの考慮の程度
(使いやすさ, 複雑さ, パフォーマンス, 開発し易さ)

<図 4. 成果物の品質基準>

2.2.2. QI 実施方針と実施パフォーマンス

前述の QI プロセスの全ての作業工程中で最も作業時間に影響を与える工程は、実は QI 作業の“作業依頼”を行う QI 実施準備なのである。各 QI 担当者はリーダーから示される“QI 作業依頼書”に基づいて作業を行うが、この中にはプロジェクト現状やプロジェクト・マネージャからのコメント、品質検証方針と共に、様々な角度から測定された成果物の基礎データが列挙されている。

この基礎データとは、成果物のあらゆる外的因子を測定し、加工して成果物検証作業の入力として利用するための数値である。このような成果物の中身、意味

を熟視テスト (Stare Test) する前工程で測定する外的因子を「品質兆候因子」(以下 QVS:Quality Vital Signs)と呼んでいる。

この QVS は、病院に通院した際の医療行為に置き換えると分かりやすい。診察室にて医師の受診に先立ち、身長、体重や体温・脈拍・血圧を測定した事はないだろうか。俗にバイタルサインと言われるこれらの数値は、治療計画や検査実施に際する参考数値・前提数値として利用される[6]。この考え方をプロジェクト上流工程での成果物検証に適用したものである。

弊社では QVS の測定をプログラムによって自動化している。実際の QVS 測定時間は 1-2 分程度で測定できるよう QVS ツールセットを作成し、プロジェクト成果物品質の定量的把握に貢献している。

これら基礎数値を元に品質の検証方針を決定した後、各担当者へ QI 作業依頼書を作成することにより QI 作業の均質化とスピードアップが同時に図れるのである。

但し、ここには注意すべき点がある。

QVS 測定することは品質兆候因子を測定するのであって“成果物の品質特性そのもの”を検査しているのではないという点である。あくまで品質特性[5]そのものや精緻な欠陥検出にはウォークスルーやフォーマル・インスペクションといった“精密検査”が必要なのである。QVS という“測定結果”や、担当者が目視点検した“診察”結果、さらにはプロジェクト背景等を総合的に見て品質を“診断”しるのであって、その後プロジェクトに対する推奨事項やアクションプランを決定しなくてはならないのである。

2.3. 全体観的品質一検証事例

以下に、実際のプロジェクトにて筆者が経験した中規模サイズのプロジェクトに置ける上流フェーズでの品質検証の事例を紹介する。

今回紹介する事例として、代表的な上流成果物であるユースケース・シナリオの検証例を紹介する。

前出のプロセスにて説明したが、QI チームの最初のアクションである品質兆候因子(QVS)の測定結果は表 3 に示すとおりである。

“階段状”(=作成日群があり、何回かに分けて全ての

表3. ユースケースに対する Quality Vital Signs の例 (丸部分が測定結果)

検査項目	正常値	異常値	Shoti
作成日/最終更新日	フェーズ期間内に均等分散	工程フェーズ終了間際 階段状・バラツキ	WBS 等から達成品質を再検証し品質基準内か確認 時系列不整合を再検査
作成日/最終更新日	9:00~17:00 に集中	夜間に集中	コミュニケーション・エラーを検証
作成者/更新者	作成者の会社名と氏名	代表者名のみ	組織体制を再検証
文書フォーマット	統一されている	不統一	作業環境の再確認(マシン・プリンタ他)/標準化ガイドの有無確認
	統一されている	不統一	ブックフォームの適用・標準化他
使用フォント	統一されている	同一文書内でバラツキ	作成者・更新者が一人かどうか再確認
	統一されている	担当者間バラツキ	成果物間整合性の徹底検証

ユースケースを作成した可能性)”や、“担当者間でフォントのバラツキ (=標準化または他作業者との相互品質確認不測の可能性)”などの兆候因子から次のような品質検証方針を決定した。

- 方針1) 表記ミス・文章ケアレスミス等の可読性を先ず徹底検証する。
- 方針2) ユースケースの意味定義に過不足・ダブルミーニング(二通りに解釈できる意味内容)がないか検証する
- 方針3) 全ユースケース間を通じて意味的不整合・矛盾記述・重複記述がないか検証する
- 方針4) その他検出できた欠陥を列挙する

これら方針から各 QI 担当者から報告された欠陥の集計を表4に示す。総検証ユースケース数65に対して、検出欠陥数:126, 1件あたり平均1.94個の欠陥を含有している計算になる。

これら欠陥から分かることは、ID3-4までに層別されるいわゆる“要件調査不足”について62%を占め、仕様書の完了基準を設定せず作業を行ったか、曖昧であった可能性を示唆しているということである。また別の観点では、書式に関するケアレス・ミスが全体の3割を占めており(日本語表現や曖昧さから生じる欠陥が33%)プロジェクト内部での品質検討・検証の仕組み確立が有効な欠陥予防策であろうと推測した。

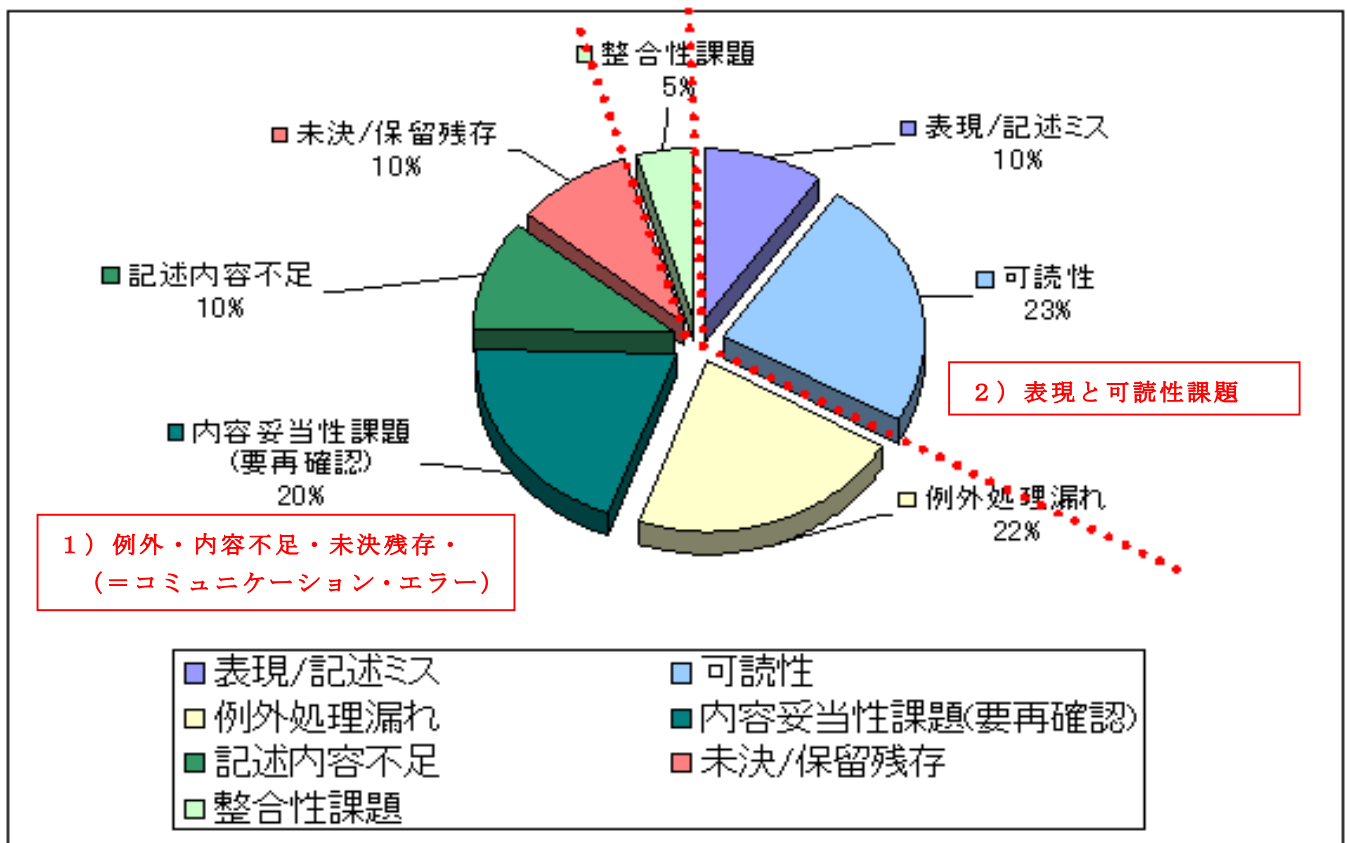
これら層別結果に基づいて、最終のフィードバックコメントとしては、複数作業者が同じ完了品質基準に基づき、均質なユースケースを記述しているか検証するように要請するコメントとなった。

予防対策1)

1 ユースケースに対して再検証/再確認が必要な欠陥が1.94個内在しています。特に例外処理に対する考慮(ID3), 妥当性の面で課題が残るもの(ID4)およびID5とID6で62%を占めます。プロジェクト内で作成者以外の他メンバーと相互レビ

表4. ユースケース上に検出された欠陥の層別

ID	Defects	Descriptions	# of Detected
1	表現/記述ミス	日本語表現の間違いや「てにをは」の間違い, 及びそれらに準ずるミス	12
2	可読性	一意に解釈できない仕様, 曖昧さの残存	30
3	例外処理漏れ	例外処理の定義がない記述	28
4	内容妥当性課題 (要再確認)	定義内容の妥当性について再確認が必要な記述	25
5	記述内容不足	定義内容に不足がある記述	13
6	未決/保留残存	未決/保留事項として明記がある記述	12
7	整合性課題	整合性の観点で課題を内包する記述	6



<図 5 : 品質欠陥のパイチャート>

ューを実施するなど、仕様の目視テスト回数を増やし欠陥予防に注力してください。

予防対策 2)

日本語表現や曖昧さから生じる欠陥が 33% を占めます。これも相互レビューなど多くの人の目視によって比較的簡単に除去できる欠陥です。

実際に上記 2 対策に基づいて、プロジェクト側メンバーでは、「内部相互レビュー」やメンバー自身による目視テストの体制強化と技術移転・均質化が行われていたため、次フェーズ（設計局面）における管理項目（進捗度・欠陥数・スケジュール）は当初の計画値を大幅に上回る数値であった。

当該例で示した欠陥検出に関するプロセス、即ち①予測②欠陥の層別分析③可視化④対策立案の 4 つのうち、②-③については過去にも一般的に行われた内容ではあるが、①の予測については、事例を蓄積することにより予測精度が高まり、多くのプロジェクトに適用可能なレベルへ昇華できるのではないかと筆者は考える。

3. まとめ

本稿で述べたインスペクション自体は、決して目新しい技法ではない。他の技術に比して派手さや斬新さもあまりない。しかし、様々な企業文化、様々なプロセス・方法論・組織体へ適用可能な応用範囲の広い技術ではないかと筆者は考えている。

特に技術変化の激しい IT 業界において 10 年以上将来にもおそらく陳腐化しないであろう技術は極めて少ない“稀有な専門技術”であろうと確信しているのである。

QI 手法の普及には、当初は筆者本人も相当の年月が必要であろうと予想していた。しかし、筆者の所属組織の年齢が 10 年以下、主に 5-6 年であることと、入社以来 QI 作業専門に研鑽を積んだことが功を奏し、わずか 2 年間の間にコスト効果の高い QI 手法が驚くべき速度で普及したことを最後に付け加えておく。

現在の国内 IT 市場において、品質検証に関わる職種（テスト・スペシャリスト、モデレーター、インスペクター、テストツール・スペシャリストなど）がもっと注目されてもいいのではないかと筆者は常々考えている。昔から「コーディングはヒトの技、デバッグは神の技」と呼ばれるほど、広範囲な知識、経験、そして何より習得の難しい「センス・勘どころ」が要求さ

れる職種であろう。これらを兼ね備えた人材の処遇が、今よりも更に重要視されてもよいのではないかと考えている。

他の工業製品と同様に、より多く、早く、均質化した製品・サービスの開發生産を行う真の IT 国家形成に向けての最も注力すべき点はこの品質保証の分野ではないだろうか。

現代の IT 市場により多く品質検証のプロフェッショナルが育成されることを切に願う。

文 献

- [1] Michael E. Fagan, “Design and Code Inspection to Reduce Errors in Program Development”, IBM Systems Journal, 15,3 1976.
- [2] Michael E. Fagan, “Advances in Software Inspections”, IEEE Transactions of Software Engineering, 12, 7 (July 1986), pp.744-751.
- [3] Edward Yourdon 著 国友義久・千田正彦共訳, “ソフトウェアの構造化ウォークスルー[第2版]”, 近代科学社, 1991, ISBN 4-7649-0187-0.
- [4] Ronald A. Radice, “High Quality Low Cost Software Inspections”, Paradoxicon Publishing, 2001, Chapter 1-9, ISBN 0-9645913-1-6.
- [5] Tom Gilb・Dorothy Graham 著 伊土誠一・富野壽監訳, “ソフトウェアインスペクション”, 共立出版株式会社, 1999, ISBN 4-320-09727-0.
- [6] HEART Nursing - “心疾患患者の早期退院への可能性”, メディカ出版, 2001, Vol.14 no.12 (1205) P 76-85.
- [7] Robert Dunn 著 渡部研一訳, “ソフトウェアの欠陥除去技術ーバグを効果的に取り除くにはー”, 日経マグロウヒル社, 1985, ISBN 4-8222-7010-6.
- [8] IEEE “IEEE Standard Glossary of Software Engineering Terminology”, IEEE Computer Society, 1990, ISBN 1-55937-067-X.
- [9] Quality Inspection Team Strategy 2003, 日本アイ・ビー・エム 社内資料, 2002.
- [10] Kokugo Dai Jiten Dictionary. Shinsou-ban (Revised edition) Shogakukan 1988/国語大辞典 (新装版) 小学館 1988.