

## 短期大規模開発における標準化品質確保戦略

### The Strategy for Standardization Quality Assurance of System Development in the Large-scale and Short-term Development

伊藤 慎太郎

**要約** 日本郵便の取引管理システム（後続稼働）プロジェクトは、大規模かつ短期間の開発であった。一般的な手法や日本ユニシスの過去のプロジェクトの知識をベースに品質管理プロセスを策定したが、期間内で開発を完了させるためにシステムの細分化を行った結果、標準化をより緻密に行う必要が発生した。その対策として、設計工程における第三者による2段階の標準化レビュー、製造工程におけるハイスル要員によるコードレビューを追加で実施し、それぞれ効果を上げた。

**Abstract** The development of Japan Post Trade Management System was a large-scale and short-term development. At first, to ensure the quality, we developed a quality management system process based on general methods or the technique used in the past system development by Nihon Unisys, and applied it to the development. As a result of the segmentation of system to complete the development within a time frame, the standardization of system development required extreme precision. To take countermeasures for this problem, we added a two-steps independent review at the design process and a code review by experts.

#### 1. はじめに

本号別稿の「短期大規模開発におけるシステム分割構築戦略」で記述しているとおり、日本郵便株式会社（以降、日本郵便）の取引管理システム（後続稼働）開発<sup>\*1</sup>（以降、本プロジェクト）は「大規模」かつ「短期間」という特性があった。また本番時期についても延期できない状況であり、その対策として、システムの細分化による個々の開発規模の縮小を行った。

本プロジェクトの品質確保策として、まず、一般的な品質管理手法や、これまで日本ユニシスが実施してきたプロジェクトの経験をベースに、「予防対策」「成果物作成時の対策」「品質評価」「品質向上策」の順で従来の品質管理プロセスを策定した。ただし、システムを細分化したことにより、規模や期間に対して、開発チーム数、要員数、インターフェイス数が多くなる結果となった。品質確保のためには、開発手順やコーディング方法等をチームや要員間で統一すること、インターフェイス間の仕様を統一することが重要である。本プロジェクトにおいては、これらをより緻密に実施する必要があった。

工業製品等、機械で標準化を実施する場合は数の増加による影響は小さいが、システム開発の標準化については人手で行うため、影響が非常に大きくなる。そのため、従来の品質管理プロセスに追加の対策を組み込み、新たな品質管理プロセスを構築した。

追加の対策は、設計工程（BD 工程、SD 工程）と製造工程（PG/UT 工程）<sup>\*2</sup>で実施した。本稿では、2章で日本ユニシスの知見に基づく従来の品質管理プロセスの概要、3章で本プロ

プロジェクトの特性を踏まえた従来の品質管理プロセスの課題，4章で標準化を課題とした対策と効果について記述する。

品質管理プロセスの策定は，ベンダの責任工程であるST工程までとした。また，本プロジェクトは要件定義工程の途中で一旦中断した後，半年後に再開したという経緯があるため，RA工程（要件定義工程）も対象外としている。

## 2. 従来の品質管理プロセスの概要

本章では，規模や期間の特性によらず，日本ユニシスの知識と経験に基づいて定めた従来の品質管理プロセスの概要について記述する。

本プロジェクトでは，以下に挙げる予防対策から品質向上策までの四つのサイクルを回すことで品質を確保する計画とした（図1）。以降の各節で説明する。

- 1) 予防対策  
問題を未然に防ぐための事前対策
- 2) 成果物作成時の対策  
セルフチェックやレビュー等，成果物作成時の対策
- 3) 品質評価  
レビュー結果，テスト結果の評価と，品質改善対策の立案
- 4) 品質向上策  
品質評価の結果に基づく改善策の実施

		2013			2014												2015				
		10	11	12	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3		
工程		設計						製造						テスト							
		BD			SD			PG/UT				PT		ST							
品質管理プロセス	予防対策	規約・サンプル作成、説明会・勉強会、パイロット																			
	成果物作成時の対策	セルフチェック、テクニカルレビュー(設計レビュー、コードレビュー、テスト仕様書レビュー)																			
	品質評価	品質レビュー																			
		品質マトリクス分析、不具合分析、個体管理																			
														外部品質特性評価							
	品質向上策	再教育、不具合・障害の横展開、再点検																			
								追加テスト													

図1 品質管理プロセス全体図

### 2.1 予防対策

予防対策は事前教育と結果レビューで成果物の品質を高めることを目的とした施策であり，大きくは次の3種類に分類される。

## 1) 規約，手順書の作成

成果物作成手順書，設計，管理フロー，セルフチェックリスト等を作成する。

## 2) 説明会，勉強会

規約，手順書等に関する説明会，勉強会を実施して関係者へ周知する。

## 3) パイロットの実施

当該工程の成果物の一部を工程開始前に前倒しで作成し，問題の事前検出を図る。各チームから代表者を選定し，スキル習得とチームへのフィードバックも併せて実施する。

## 2.2 成果物作成時の対策

成果物作成時には，まず作成担当者がセルフチェックリストで確認した後，レビューによるテクニカルレビューを実施する（表1）。テクニカルレビューは，設計書やプログラムの主担当者によるレビューとチームリーダークラスによるレビューの2段階で，成果物への要件・規約等が確実に反映されているかをチェックする。レビューについてもチェックリストに基づいて実施する。これにより，個人のスキルへの依存をなくし，レビュー品質の平準化を図る。最後に，出荷前確認として，主に体裁面の点検を実施する。

表1 テクニカルレビュー

No	工程	対象成果物	テクニカルレビュー
1	BD 工程	基本設計書	成果物検証（主担当）
			成果物検証（リーダー）
			出荷前確認
2	SD 工程	詳細設計書	成果物検証（主担当）
			成果物検証（リーダー）
			出荷前確認
3	PG/UT 工程	プログラム	成果物検証（主担当）
		単体テスト仕様書	成果物検証（リーダー）
		単体テスト結果	出荷前確認
4	PT 工程	結合テスト仕様書	成果物検証（主担当）
		結合テスト結果	成果物検証（リーダー）
			出荷前確認
5	ST 工程	システムテスト仕様書	成果物検証（主担当）
		システムテスト結果	成果物検証（リーダー）
			出荷前確認

## 2.3 品質評価

品質評価では，レビューやテストの実績を以下の四つの観点で評価するとともに，問題点を分析した上で品質改善の対策を立案する。

## 1) 品質メトリクス分析

各工程の作業・成果物の特性に応じた品質メトリクス（表2）で，成熟度を判断する。

## 2) 不具合分析

各工程のレビューやテストにおける不具合/障害情報，及びその分類集計結果から，問題傾向の把握と対策の立案を行う。

表2 各工程の品質メトリクス

工 程	内 容	管理指標	算定方法
BD 工程	基本設計書レビュー	レビュー密度	レビュー人時/ページ
	基本設計書レビュー指摘	不良密度	不具合件数/ページ
SD 工程	詳細設計書レビュー	レビュー密度	レビュー人時/ページ
	詳細設計書レビュー指摘	不良密度	不具合件数/ページ
PG/UT 工程	プログラムの複雑度	サイクロマティック複雑度	ツールにより計測
	単体テストのテスト密度	テスト密度	テスト項目数/Ksteps
	異常系テストケース比率	正常系/異常系比率	異常系テスト項目数/全テスト項目数
	単体テストの障害検出率	不良密度	障害件数/Ksteps
PT 工程	結合テストのテスト密度	テスト密度	テスト項目数/Ksteps
	結合テストの障害検出率	不良密度	障害件数/Ksteps
ST 工程	システムテストのテスト密度	テスト密度	テスト項目数/Ksteps
	システムテストの障害検出率	不良密度	障害件数/Ksteps

### 3) 個体管理分析

各工程の作業/成果物について、品質を分析する単位（＝個体管理単位）を設定し、問題傾向のある個体の検出と原因の把握、対策の立案を行う。

### 4) 品質特性評価

品質特性<sup>\*3</sup>（表3）の目標と達成度評価基準を定め、各工程終了時に達成度を評価する。また、品質評価については、プロジェクトの品質管理チームによる評価に加えて、日本ユニシスの品質管理部門による品質レビュー（QAR<sup>\*4</sup>）での評価も併せて実施する。

## 2.4 品質向上策

品質向上策として、品質評価で立案した品質改善の対策を実施する。対策は、以下の4パターンをベースとし、具体的な内容は都度検討する。実施主体は成果物を作成したチームだが、規模が大きい場合や複数チームに跨る場合は、品質向上の専門チームを編成することも検討する。

- 1) 弱点に対する再教育や再周知、ハイススキル要員への担当交代
- 2) 不具合や障害の情報の横展開と、再発防止策の実施
- 3) 設計書やテスト仕様書等の再点検
- 4) テストケースの見直し、再テスト

## 3. 従来の品質管理プロセスの課題

大規模かつ短期間という特性により、2章で説明した品質管理プロセスの十分性やリスクを分析し、当該品質管理プロセスでは解決できない課題を洗い出した。

### 3.1 大規模であるというリスク

一つのチーム（1人のリーダー）で品質や進捗をコントロールできる開発規模には限界がある。そのため、開発の規模が大きくなる場合はシステムを複数のサブシステムに分割した上で、それぞれ別のチームで開発を進めていくという方法が一般的である。ただし、分割した場合、サブシステム単体では規模が縮小し品質リスクが小さくなるが、サブシステム間でのデータ受け渡し規約や画面・帳票の標準作成規約の認識齟齬等、いわゆる標準化に関わる問題が発生する

表3 品質特性

No	品質特性	副特性	特性概要
1	機能性		ソフトウェアが、指定された条件の下で利用されるときに、明示的及び暗示的の必要性に合致する機能を提供するソフトウェア製品の能力
		a) 目的性	指定された作業及び利用者の具体的目標に対して適切な機能の集合を提供するソフトウェア製品の能力
		b) 正確性	必要とされる精度で、正しい結果若しくは正しい効果、又は同意できる結果若しくは同意できる効果をもたらすソフトウェア製品の能力
		c) 相互運用性	一つ以上の指定されたシステムと相互作用するソフトウェア製品の能力
		d) セキュリティ	許可されていない人又はシステムが情報又はデータを読んだり、修正したりすることができないように、及び許可された人又はシステムが情報又はデータへのアクセスを拒否されないように、情報又はデータを保護するソフトウェア製品の能力
		e) 機能性標準適合性	機能性に関連する規格、規約又は法律上及び類似の法規上の規制を遵守するソフトウェア製品の能力
2	信頼性		指定された条件下で利用するとき、指定された達成水準を維持するソフトウェア製品の能力
		a) 成熟性	ソフトウェアに潜在する障害の結果として生じる故障を回避するソフトウェア製品の能力
		b) 障害許容性	ソフトウェアの障害部分を実行した場合、又は仕様化されたインタフェース条件に違反が発生した場合に、指定された達成水準を維持するソフトウェア製品の能力
		c) 回復性	故障時に、指定された達成水準を再確立し、システムを回復するソフトウェア製品の能力
		d) 信頼性標準適合性	信頼性に関連する規格、規約又は規則を遵守するソフトウェア製品の能力
3	使用性		指定された条件の下で利用するとき、理解、習得、利用でき、利用者にとって魅力的であるソフトウェア製品の能力
		a) 理解性	ソフトウェアが特定の作業に特定の利用条件で適用できるかどうか、及びどのように利用できるかを利用者が理解できるソフトウェア製品の能力
		b) 習得性	ソフトウェアの適用を利用者が習得できるソフトウェア製品の能力
		c) 運用性	利用者がソフトウェアの運用及び運用管理を行うことができるソフトウェア製品の能力
		d) 魅力性	利用者にとって魅力的であるためのソフトウェア製品の能力
		e) 使用性標準適合性	使用性に関連する規格、規約、スタイルガイド又は規則を遵守するソフトウェア製品の能力
4	効率性		明示的な条件の下で、使用する資源の量に対比して適切な性能を提供するソフトウェア製品の能力
		a) 時間効率性	明示的な条件の下で、ソフトウェアの機能を実行する際の、適切な応答時間、処理時間及び処理能力を提供するソフトウェア製品の能力
		b) 資源効率性	明示的な条件の下で、ソフトウェアの機能を実行する際の、資源の量及び資源の種類を適切に使用するソフトウェア製品の能力
		c) 効率性標準適合性	効率性に関連する規格又は規約を遵守するソフトウェア製品の能力
5	保守性		修正のし易さに関するソフトウェア製品の能力
		a) 解析性	ソフトウェアにある欠陥の診断又は故障原因の追及、及びソフトウェアの修正箇所の識別を行うためのソフトウェア製品の能力
		b) 変更性	指定された修正を行うことができるソフトウェア製品の能力
		c) 安定性	ソフトウェアの修正による、予期せぬ影響を避けるソフトウェア製品の能力
		d) 試験性	修正したソフトウェアの妥当性確認ができるソフトウェア製品の能力
		e) 保守性標準適合性	保守性に関連する規格又は規約を遵守するソフトウェア製品の能力
6	移植性		ある環境から他の環境に移すためのソフトウェア製品の能力
		a) 環境順応性	ソフトウェアにあらかじめ用意された以外の付加的な作業又は手段なしに、指定された異なる環境にソフトウェアを適用させるためのソフトウェア製品の能力
		b) 設置性	指定された環境に設置するためのソフトウェア製品の能力
		c) 共存性	共通の資源を共有する共通の環境の中で、他の独立したソフトウェアと共存するためのソフトウェア製品の能力
		d) 置換性	同じ環境で、同じ目的のために、他の指定されたソフトウェア製品から置き換えて使用することができるソフトウェア製品の能力
		e) 移植性標準適合性	移植性に関連する規格又は規約を遵守するソフトウェア製品の能力

リスクが高くなる。

本プロジェクトにおいても10のサブシステムに分割されていたが、規模の大きなサブシステムについては、その中でさらにチームを細分化していた。そのため、実質的には22のチームに分割される状態となっており、標準化に関する問題の多発が懸念された。

### 3.2 習熟期間が短いというリスク

取引管理システム開発（後続稼働）は、取引管理システム開発（先行稼働）に比べて開発規模が大きく、多数の要員が新規に参画した。特に、論理設計工程では、プロジェクト開始直後に一斉に成果物作成を開始する。そのため、前章の品質対策は実施するものの、設計書の形式や開発フレームワーク、開発支援ツール等に不慣れな状態である。その中で、多岐にわたる設計規約やコーディング規約を作成担当者が習得できないまま成果物作成を開始することが懸念された。

また、1人の担当者が多くの成果物を作成すれば、習熟によるスキルアップが期待できるが、短期間であったため、1人が作成を担当する成果物は1～2本程度という状況であった。

### 3.3 大規模・短期間という特性における課題

大規模、短期間という点を前節までのように考慮した場合、標準化に関する問題が多発するリスクが大きいと結論に至った。また、リスクとなる標準化の範囲について、以下のように整理した。

#### 1) システム面の標準化リスク

- ・設計規約やコーディング規約が遵守されないことに起因するシステム障害により、テスト工程で手戻りが発生する。
- ・同じ処理に対して異なる実装がされることにより、保守性が低下する。

#### 2) 体裁面の標準化リスク

- ・体裁面の指摘が多発することにより、レビューでの本質的な指摘が不十分となる。
- ・設計書の可読性が悪く、コーディング誤りやテストケースの抽出漏れが発生する。

## 4. 標準化対策の実施内容と効果

標準化はシステム全体に関わる事項であり、致命的な問題が発生した場合の影響が大きく、システム全体の見直しに及ぶ可能性もある。そのため、テスト工程での検出・対策では遅く、設計工程、製造工程で対策をする必要があると判断し、対策を立案した。

本章では、従来の品質管理プロセスの課題である標準化の課題解決のため、設計工程、製造工程において実施した対策の内容と効果について記述する。

### 4.1 設計工程の標準化対策と効果

設計工程では、まず、事前教育の強化による要員のレベルアップという予防対策を検討した。説明会の回数を増やし、実習や確認テストを取り入れることを検討したものの、プロジェクト開始直後から論理設計を行う必要があり、予防対策の実施期間の確保は難しいと判断した。そのため、予防ではなく、成果物のテクニカルレビュー終了後に、標準化に特化したレビューを追加で実施し、問題のすり抜けを防止することを対策とした。



### 4.1.1 レビュー観点の整理

体裁、処理方式に関するレビュー観点を表4のとおり整理した。観点整理については、開発フレームワークの主管部、取引管理システム（先行稼働）での経験者を中心に実施した。

表4 標準化レビューの観点

レビュー名	主な観点	内 容
体裁レビュー	設計書の表記統一	フォント及びサイズ、章立て、設計書内図表の凡例統一、日付の表記方法等
	文章の適正化	文末（ですます調）の統一や句読点の打ち方等の文章表現、半角全角使い分け、あやふやな日本語の訂正等
	設計書への記載内容統一	設計書への記載項目、粒度、設計書の分割ルール、設計書間での記載分担等
処理方式レビュー	機能単体処理	画面処理（単件登録、複数件登録、ファイルアップロード、照会）、バッチ処理方式（常駐、非同期、適時）
	内部連携	画面遷移、画面処理とバッチとの連携、帳票出力
	外部連携	OLTP、SOA、ファイル連携

### 4.1.2 レビューの実施

レビュー体制は、体裁、処理方式の2チーム編成とした。それぞれレビュー観点を定めた要員が中心となったが、追加メンバについては、客観的な視点でチェックできること、成果物作成の要員リソースを減らさないことを目的として、プロジェクト外から調達した。体裁チーム、処理方式チームそれぞれ、設計書作成経験が豊富な要員や、開発フレームワークの有識者を選定した。また、サブシステム単位に担当を固定せず、横断的にチェックさせることにより、事前

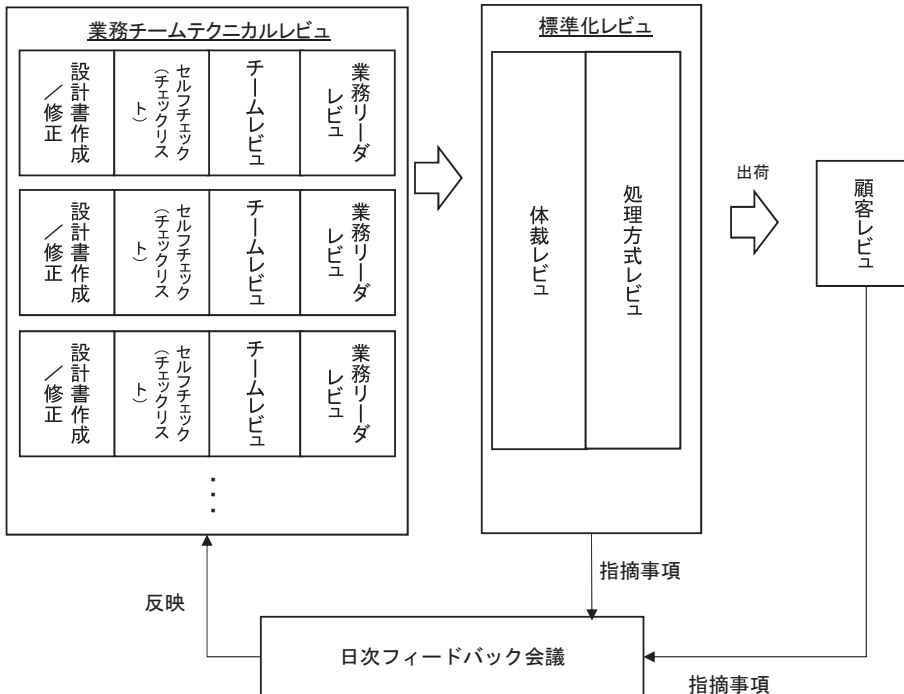


図2 標準化レビューフロー

に定義できなかった新たなレビュー観点を発見し、レビュー品質を向上させることを狙った。

標準化レビューフローは図2のとおりである。業務チームによるテクニカルレビュー終了後に体裁、処理方式のレビューを追加した。また、指摘事項については、日次で全体へフィードバックした。間違えやすい誤りや、追加された標準化規約の周知を行うことで、以降の成果物作成時における品質の向上を図った。

新たな指摘事項については、レビューが終了している設計書に対しても実施する必要がある。そのため、図3のマトリクスを作成し、レビュー済みの設計書にも遡及して確実に指摘事項が反映できることを担保した。

		全ての設計書									
		設計書 1	設計書 2	設計書 3	設計書 4	設計書 5	設計書 6	.	.	.	.
新たな指摘があれば追加	指摘1	済	済	済	済	済	済	済			済
	指摘2	済	済	済	済	済	済	済			済
	指摘3	済	済	済	済	済	済	済			済
	指摘4			済	済	済	済	済	済	済	済
	.				済	済	済	済	済	済	済
.					済	済	済	済	済	済	
.						済	済	済	済	済	

図3 指摘反映マトリクス

#### 4.1.3 標準化レビューの効果

BD工程とSD工程の不具合数を表5に示す。BD工程において、標準化レビューで3,025件（BD工程の不具合の26%）の不具合を検出した。標準化に関する不具合の総数が5,335件であり、半数を超える57%が標準化レビューでの検出であった。追加の対策を実施しなければ、3,025件の問題が後工程にすり抜けていたため、非常に大きな効果があったと評価する。

SD工程では、標準化に関する不具合の割合が33%で、BD工程に比べて減少している。また、標準化レビューでの指摘割合も8%と大幅に減少している。レビュー内容の変更はないため、設計書作成者、レビューが習熟したことにより、標準化の観点での設計書作成時の品質と、テクニカルレビューの精度が向上したと考える。

表5 BD工程、SD工程の不具合数

	BD	SD	計
全不具合数（件）	11,567	24,360	35,927
うち、標準化不具合（件）	5,335	8,026	13,361
全不具合数に対する標準化不具合の割合（%）	46%	33%	37%
標準化レビューでの指摘数（件）	3,025	2,068	5,093
全不具合数に対する標準化レビューでの指摘数の割合（%）	26%	8%	14%
標準化不具合数に対する標準化レビューでの指摘数の割合（%）	57%	26%	38%

#### 4.2 製造工程の標準化対策

製造工程でも事前教育での要員レベルアップを検討したものの、設計工程同様、説明回数の増加と実習以外に効果的な対策はないとの結論に至った。設計担当者とコーディング担当者に分けることで、事前の時間確保は可能であったが、本プロジェクトでは習熟による期間短縮と



品質確保を図り、基本的にコーディングと詳細設計の担当者は同一としていた。そのため、事前の時間確保は難しく、設計工程同様、標準化専門のコードレビューを追加し、成果物作成後に問題のすり抜けを防止することを対策とした。

#### 4.2.1 レビュー観点の整理

標準化コードレビューでの観点を以下1) から4) の通りに定めた。規約の遵守を機械的にチェックするだけでなく、問題のあるコーディングや非効率なコーディングについてもチェックアウトの対象とした。また、標準化の範囲ではないが、レビューにあたり詳細設計書を参照するため、設計書との処理内容不一致もレビュー観点に加えた。

- 1) 標準化規約違反  
標準化規約として定めた事項に対する違反
- 2) コーディングの誤り  
無限ループになる処理や、データベースのクローズ漏れ等の誤り
- 3) 非効率な処理  
無駄な処理や、処理効率の悪いコーディング
- 4) 設計書との処理内容不一致  
設計書の期待結果と異なる結果になるコーディング

#### 4.2.2 レビューの実施

レビューは機械的なチェックだけではないため、実施する要員は、コーディングスキルが高く、開発フレームワークを熟知していることに加えて、業務仕様についても一定の知識が求められた。そのため、規約を作成した要員と、取引管理システム（先行稼働）経験者から選抜したスキルの高い要員でチームを編成した。

また、レビューには時間とパワーが必要なので、全量を対象とせず、ピックアップとし、指摘事項を各サブシステム内で横展開するフローとした。ピックアップは、「規模大若しくは、複雑度高である」、「業務的に使用頻度が高い」という観点で実施したが、処理が類似しているプログラムについては、代表的な1プログラムのみを対象とした。

レビューフローは図4のとおりである。期間短縮のため、業務チーム担当者のコードレビュー終了後、業務チームのリーダーレビューと並行して標準化レビューを実施した。レビューでの指摘事項は、各チームにフィードバックされ、他のプログラムへの横展開を行った。

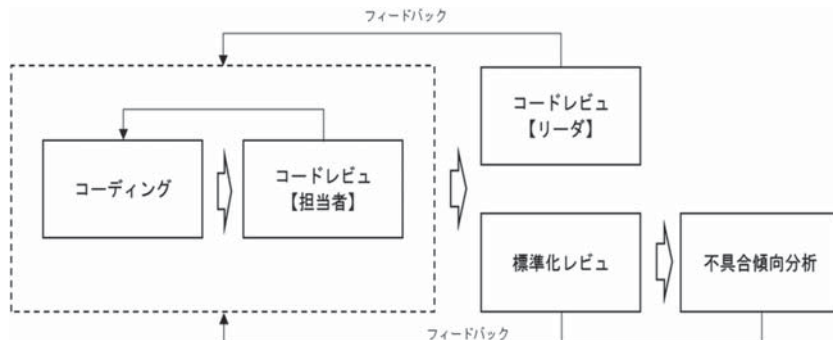


図4 コードレビューフロー

また、工程中間で指摘事項の傾向をサブシステム単位に分析した。間違えやすいポイントを明確にして要員の再教育等の対策を行い、品質の底上げを図った。

#### 4.2.3 標準化レビューの効果

標準化レビューでの指摘数は2,622件で、全指摘数の9%であった(表6)。この指摘が対策されずに、単体テストにおいて同件数の障害が発生すると仮定した場合、単体テストの障害検出率が約4割増となったが、プロジェクトにこれを吸収できるだけの期間・要員リソースはなかった。結果として、この対策を実施したことにより期間を遵守できたため、非常に大きな効果があったと評価する。また、ピックアップした以外のプログラムに対するフィードバック分を含んでいないため、標準化レビューでの実質的な指摘数は、さらに多かったと推定される。

表6 コードレビュー, 単体テスト結果

コードレビュー	全指摘数(件)	30,155
	標準化レビュー指摘数(件)	2,622
	全指摘数に対する標準化レビュー指摘数の割合(%)	9%
単体テスト	障害数(件)	6,384
	不良密度(件/Kstep)	6.99
	障害数+標準化レビュー指摘数(件)	9,006
	障害の増加割合(%)	41%
	不良密度(件/Kstep)	9.86

## 5. おわりに

日本郵便の取引管理システムは2016年2月に本番を迎え、2016年7月現在は保守フェーズに入り安定的に稼働している。ただし、機能追加は継続して実施されており、2017年4月本番予定の追加開発についても既に着手している状況である。以降も一定規模の追加開発が予想され、引き続き品質確保への取り組みは重要なテーマである。

本稿で述べたとおり、効率性やすり抜けリスクを考慮すると、問題を発生させないように予防することが重要である。本プロジェクトでは、対策検討の期間的な制約もあり、成果物作成後のレビューに焦点を絞って実施した。今後の開発においては、標準化以外の観点も含めて、予防により重点をおいた対策を取るよう、品質プロセスを継続して改善していきたい。

- 
- \* 1 日本郵便の取引管理システムでは、先行と後続の2段階に分けて開発を実施した。先行開発は2013年4月に、後続開発は2016年2月にそれぞれ本番稼働した。
  - \* 2 RA工程:要件定義工程, BD工程:基本設計工程, SD工程:詳細設計工程, PG/UT工程:プログラミング/単体テスト工程, PT工程:結合テスト工程, ST工程:システムテスト工程。
  - \* 3 ソフトウェアバグ以外のソフトウェア品質を評価する指標。本プロジェクトではISO/IEC9126(JIS X 0129-1)で定義されている指標を適用した。
  - \* 4 Quality Assurance Review 日本ユニシス品質保証部門による、工程終了判定レビューを指す。工程終了時だけでなく、必要に応じて工程中間でも開催する。

**執筆者紹介** 伊藤 慎太郎 (Shintaro Ito)

1990年日本ユニシス(株)入社。地方金融機関の勘定系担当として一連の金融自由化対応に取り組んだ後、2000年より金融機関向けCRMソリューション及びDWHの開発と導入を担当する。

2013年より日本郵便株式会社における次世代取引管理システムの開発に従事。

