

## 4

上流工程における発注者視点から  
の品質向上への取り組み<sup>☆1</sup>

清田 辰巳 東京証券取引所

現在、大規模システムの開発で一般的に採用されているウォーターフォールモデルでの開発では、発注者は要件を開発者に伝え、システムが納品される検収テストまで、成果物の品質を確認する機会が少ない。

東京証券取引所では、システムの信頼性向上を図るには上流工程でのプロセス改善が重要との認識の下、発注企業として要件定義、外部設計の品質を強化することはもちろんのこと、開発ベンダで作成される設計書の品質管理を開発者側と一体となって実施できるようさまざまな取り組みを進めている。なお、本稿での上流工程とは要件定義から基本・詳細設計までをいう。

本稿では、それらのうち、要件定義書等上流工程での成果物の品質確保の取り組みや、要件から設計に至るトレーサビリティの確保による品質向上の取り組みについて紹介する。

## ▶ 東証システム開発標準

東京証券取引所(以下「東証」)では、ウォーターフォールモデルを基本とした発注者の立場からのシステム開発の標準プロセスを定めている(図-1)。

この東証のシステム開発標準では、開発案件の発生から開発ベンダとのシステム開発契約の締結までの上流工程を「ソリューション設計フェーズ」と呼んでいる。

ソリューション設計フェーズでは、東証は発注者の立場から、開発ベンダ選定のための提案依頼書(RFP)の作成と要件定義書の作成を標準成果物として定めている。

ベンダ選定が終了すると、開発ベンダと「ソリューション設計業務委託契約」を締結し、要件定義書や「プロジェクト計画書(プロジェクト運営ルール等を明文化する

ドキュメント)」の精査を行うこととなる。特に要件定義書は、以降の工程におけるインプットとして、システム品質に大きな影響を与えることから、東証では、要件定義書の精度向上が大きな課題となっている。

## ▶ 要件定義の課題

## ■ 要件定義書に係る発注者と開発ベンダとの溝

従前の東証では、要件定義書を作成するにあたっては、開発するシステムの利用者の立場から、当該システムを「どう使うか」という観点で主に業務プロセスを中心に記述してきた。

一方、開発ベンダは、東証が作成した要件定義書に基づき、発注されるシステムを「どう作るか」という観点で、設計書を作成することとなる。換言すれば、これまでの東証での要件定義書では開発ベンダが作成する設計書の直接的なインプット情報となれない大きな溝があり、このことがシステム開発における上流工程での大きなリスクとなっていたといえる。なお、要件定義のとりまとめをベンダに委託するケースもあるが、この場合、ベンダは、「どう作るか」という観点で発注者要件を整理する恐れがあり、発注者が望む要件との齟齬が発生するリスクを意識する必要があった。

また、ウォーターフォールモデルのシステム開発では、要件定義工程の期日までに要件定義を完了させることが前提ではあるものの、この段階での要件確定は現実的には不可能といえる。むしろ、開発ベンダを交えた検討の中で要件定義書の完成度が向上していくことになる。このため、東証では「開発フェーズ」での要件定義書の完成度向上プロセスも品質向上の観点から重要と考えている。特に、非機能要件に関しては、ベンダでの一定程度のシステム設計の検討が進まないと確定できない要件もあり、

.....  
☆1 本稿の著作権は、東京証券取引所に帰属します。

## 4. 上流工程における発注者視点からの品質向上への取り組み

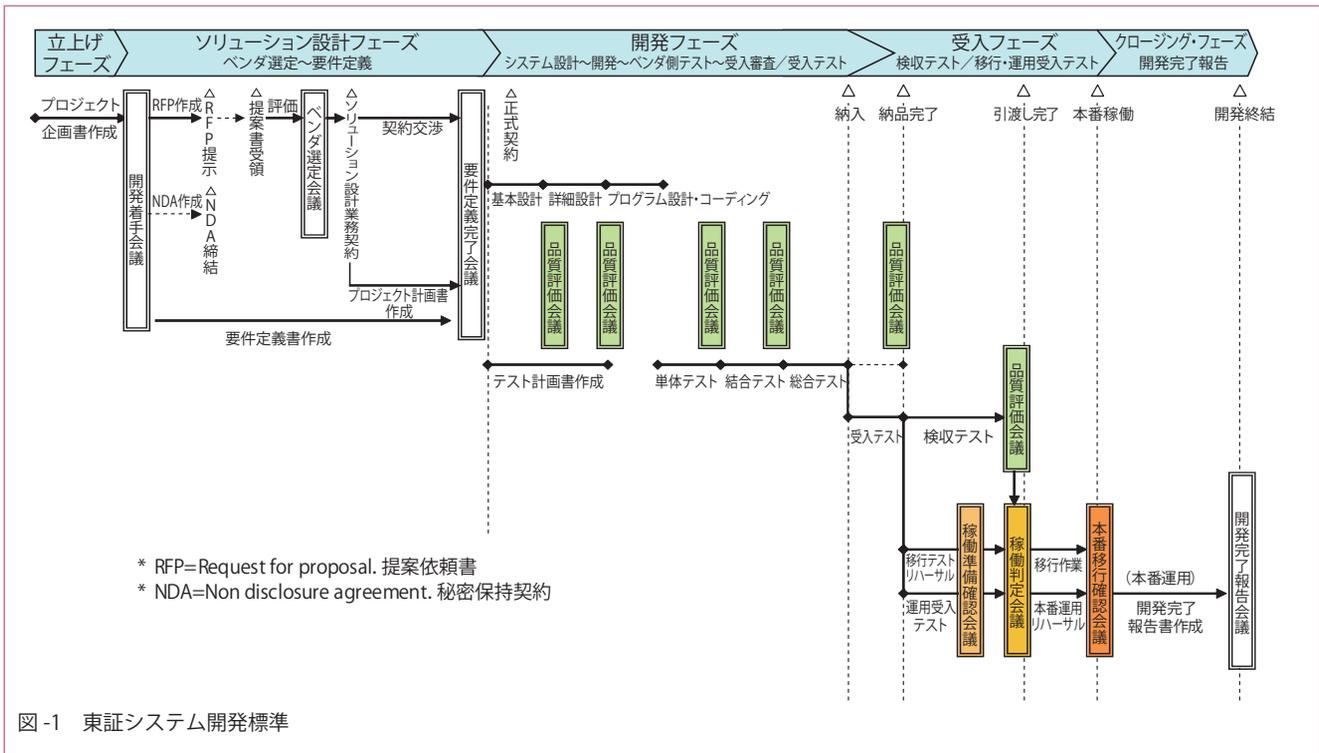


図-1 東証システム開発標準

こうした要件については、ソリューション設計フェーズ完了時において、未確定要件の内容とその確定時期を明確にしておく必要がある。

### ▶ 開発フェーズ(設計工程)での課題

#### ■ 要件トレースの必要性

システム開発工程におけるV字モデルの品質管理プロセスでは、要件定義の品質は検取テスト段階で確認されることになる。

しかしながら、この段階で要件定義内容と開発ベンダが実装した機能とに齟齬が発覚した場合、要件定義まで遡っての手戻りが発生するため、その影響度はきわめて大きいといえる。

したがって、東証では開発フェーズ(設計工程)において、要件定義書の内容が開発ベンダにて適切に各設計書に反映されているか、設計段階においてトレース(要件定義書と設計書の紐付け)できるようにしておくことが発注者としての重要な確認ポイントと考えている。

この要件トレースの実施により、発注者の要件が適切に開発ベンダに齟齬なく伝わっているのか、といった分析・評価に加えて、発注者の要件がきちっと漏れなく要件定義書に反映できているのか、といった要件定義書自体の精度確認も可能となる。

#### ■ 設計工程での要件定義書の変更管理

設計工程において、開発ベンダでの検討が進む中で未確定であった要件の確定や、発注者側の要件の見直しに

よる要件変更が発生してくる。

一方、システム設計の最適化の観点から、発注者から提示された要件の改善提案が開発ベンダから出てくるとも予想される。設計段階での要件定義書の変更管理においては、こうした要件変更起因による分析も重要と考えている。

#### ■ すり抜け管理

ウォーターフォールモデルのシステム開発では、当該工程で作り込まれたエラー(たとえば要件定義書や設計書の不備や製造段階でのコーディングミス等)の抽出が後工程になればなるほどその影響は大きくなる。当該工程で作り込まれたエラーは当該工程でつぶしておくのが鉄則ではあるものの、一定割合で次工程に「すり抜け」が発生する。このすり抜けたエラーを次工程でいかに抽出できるかが効率的品質向上の鍵となる。また、このすり抜け分析が前工程での成果物の品質評価にもつながる。

### ▶ 上流工程における品質の作り込み

東証では、前述の上流工程での課題解決に向けて、以下のような取り組みを行っている(図-2)。

#### ■ 要件定義書記載内容の改善

##### • 3層構造の要件定義

東証で2010年1月の稼働を目指して開発を進めている次世代の取引システム(以下「arrowhead」)では、機能要件に関して3階層に整理したドキュメントを作成して

～上流工程での品質作り込み～

- ① 要件定義書記載内容の改善
  - ◆ 3層構造の要件定義
  - ◆ 要件定義書雛形の改訂
- ② 要件トレース
  - ◆ 要件トレーサビリティの確保
  - ◆ 要件充足度のチェック
- ③ 要件定義書の評価
  - ◆ 検収テスト項目の抽出
  - ◆ 変更管理からの品質分析
- ④ 製造工程での詳細設計書の評価
  - ◆ コーディング中に抽出された詳細設計書のエラー管理 (コードレビューで抽出されるエラーと同等の管理)

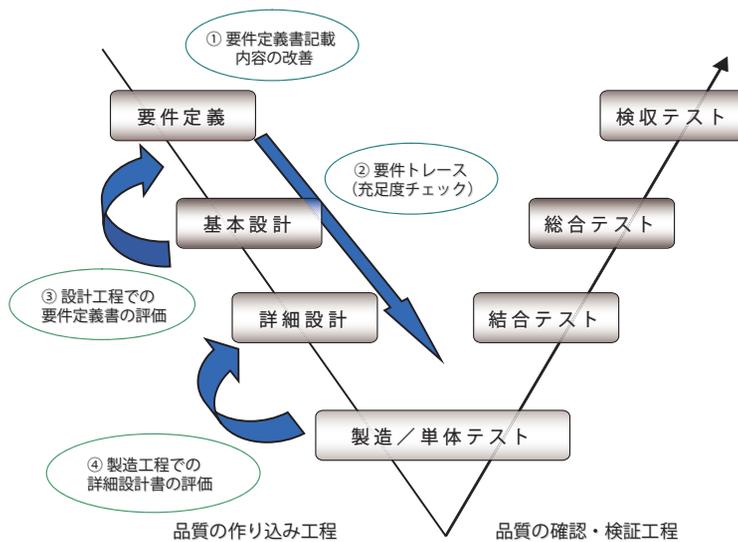


図-2 上流工程での品質作り込み

いる。

最上位層では、「アクター（外部接続先システムや利用者等）」に着目した「全体業務フロー図」を作成し、1枚の資料でシステム全体の機能群を俯瞰できるようにしている。

第2層目では、「注文処理」等の機能群で送受信するメッセージを記述した「業務フロー図」を作成している。さらに「注文処理」等の機能群を細分化した業務要件を第3層目の「要件定義書」に記載している。

こうしたドキュメント体系とすることで、機能要件の漏れや矛盾を予防し、要件定義の精度向上を図っている。

・要件定義書雛形の改訂

東証のシステム開発標準で用意している要件定義書雛形についても、「共通フレーム 2007<sup>1)</sup>」などの一般的な標準に対応させ、他社事例も参考に見直しを行うことで要件の網羅性・正確性の向上に努めている。

特に、求められる要件の背景やその重要度について明示することを推奨しており、これにより要件定義関係者のより深い検討が期待でき、また、開発ベンダでのシステム設計にも役立つものと期待している。

■ 要件トレース

東証では、ベンダが作成する基本設計書等のレビューに参加し、要件定義書に示された要求事項が設計書に適切に盛り込まれているかを確認することで、設計品質の向上を図ることとしている。

東証としての基本設計書等のレビューの観点には、要件定義書に記述されていることが間違いなく実現できるか、ということにある。しかし、基本設計書等はあくまでも

システムを構築していく観点から記述されているため、要件定義書との対応関係は必ずしも明確ではない。このため、東証側としては、レビューに先立って、要件定義書から確認すべきチェックリストを作成し、それに基づいてレビューを行うことで、レビュー漏れを防ぐようにしている。

・要件網羅性のチェック

東証では、設計書レビューにあたって、設計書に要件がすべて網羅されているかという要件網羅性の確認にチェックシートを使うことを推奨している(図-3)。

この要件網羅性チェックシートは、要件が設計に反映されているかどうかを確かめるものであるが、要件が変更になった場合、設計書のどの部分に影響があるかを追跡するためにも使うことができる(「順方向のトレーサビリティ」がある)。逆に、設計に何らかの変更が必要となった場合に、それがどの要件に影響するかを確認することができる(「逆方向のトレーサビリティ」がある)。このように、要件網羅性チェックシートにより順逆両方向のトレーサビリティが確保できることは、発注者にとって非常に有用であると考えている。

arrowheadでは、要件トレーサビリティのチェックの観点から、各要件の要素が設計に盛り込まれているかどうかのチェックやテスト工程においても漏れなくテストが行われているかのチェックのために利用している。

・業務適合性のチェック

設計が詳細化されていくにつれて、要件定義書に記述された項目がシステム上でどのように実現されるかが具体的に明らかになってくる。そこで明らかになった内容が、果たして業務として整合性のある適切なものである



ずれかとなる (図-4 中の橙色部分)。この種の変更が多い場合は、ソリューション設計フェーズでの要件定義書のレビューにおいて見逃しが多いと判断し、要件定義書の再レビュー等のアクションを検討することとしている。

一方、要件改善やエンドユーザからの要望や法制度変更などに基づく要件の追加変更 (現象コードが6か7に分類されるもので、原因コードはB1～B4となるもの。図-4 中水色部分) が多い場合も、その要件変更が他の要件に影響を及ぼしていないか確認する必要がある。したがって、見逃し率<sup>☆2</sup>があらかじめ定めた許容値を超えた場合は、変更が必要となった理由を分析し、アクションとして要件定義書の再レビューを検討することとしている。

・設計の「深さ」の評価

原因コードがB3に分類されたものに着目する必要がある。

B3に分類された指摘事項は、要件定義工程段階で存在したはずの「気づかれない不確かさ」が、設計を進めていくことにより明確になり、要件を変更すべきと判断されたものである。このようなものは要件定義には必ずあるはずで、逆説的ではあるが、この種の問題がどれだけ指摘されているかが設計の深さを示す指標となる。このような着眼点で要件変更の状況を見ることも重要と考えている。

■ 製造工程での詳細設計書の評価

・設計書エラーの管理強化

設計書品質の向上への取り組みとして、東証では、開発ベンダに対して、コーディング時に発見した設計書エラーについての管理強化を要請している。具体的にはコーディング中に検出した設計書のエラーに対し修正票 (開発ベンダ帳票) を発行し、設計書レビューにより修正を行う場合と同じ管理を徹底するものである。

これにより正確なエラー数の把握に基づく設計書品質の管理や設計書の修正が確実に行われるための管理が可能となる。

これは、設計書の修正漏れを防ぐという側面に加えて、「後工程から前工程の品質を評価する」という、包括的な品質管理の実践を目指している。また、この裏返しとして、「前工程から後工程の品質を評価する」、すなわち、後工程で前工程のエラーがある程度摘出されなければ、後工程に問題ありと考えるべきという品質管理の観点がある。

☆2 見逃し率=開発フェーズで抽出したエラー件数÷(開発フェーズで抽出したエラー件数+ソリューション設計フェーズで抽出したエラー件数)。

☆3 Quality, Cost, Delivery.

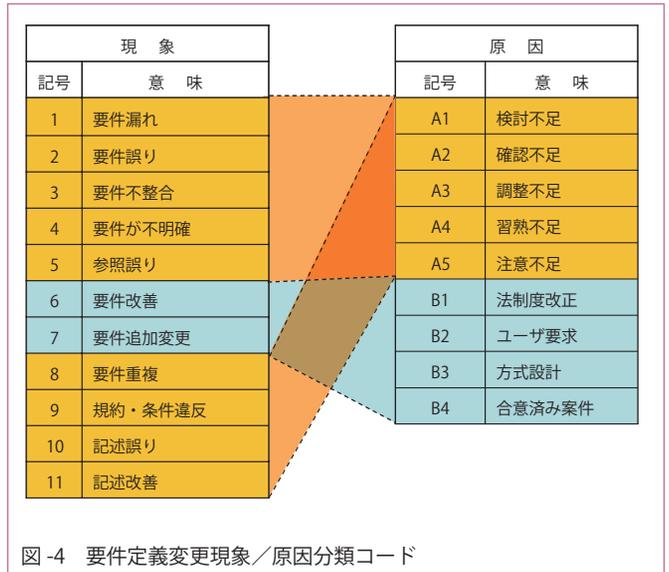


図-4 要件定義変更現象／原因分類コード

▶ 超上流工程からのプロセス整備・改善

発注者にとって、開発されたシステムが提示した要件どおりに、かつ、安定的に動作することだけでは、本来の目的が達成されたとは必ずしもいえない。

すなわち、発注者は、システム化計画を決定する際、たとえば、業務の効率化とその効果としてのコスト削減やエンドユーザでの利便性向上とその効果としての売り上げ増加など、経営レベルの目標達成が求められる。

発注者の立場からすれば、開発したシステムの稼働によって、これらの目標が実現して初めて、すべての目的 (QCD<sup>☆3</sup>+ 経営目標の実現) が達成できたといえる。

システム開発にあたって経営レベルの目標が実現できるよう、発注者 (発注企業のシステム開発関係者) は経営層や社内業務部門からの要求や外部のエンドユーザからの要求も取り込んで要件定義を進めるとともに、当該要件が経営目標を実現するためのものになっているか分析・評価したうえで要件を確定させていかなければならない。このような超上流工程<sup>2)</sup>でのプロセスの整備・改善やその評価プロセスの構築についても、発注者視点からの残された大きな課題である。

参考文献

1) (独)情報処理推進機構ソフトウェア・エンジニアリング・センター編：共通フレーム 2007, オーム社 (2007)。  
 2) (独)情報処理推進機構ソフトウェア・エンジニアリング・センター編：経営者が参画する要求品質の確保第2版, オーム社 (2006)。  
 (平成 21 年 3 月 3 日受付)

清田 辰巳 t-kiyota@tse.or.jp

(株) 東京証券取引所品質管理部長、(独) 情報処理推進機構ソフトウェアエンジニアリングセンタープロセス共有化 WG 委員。