

8. 品質に対応した プロセスデザイン

中島 毅(三菱電機(株)) 山田 淳((株)東芝) 福住伸一(日本電気(株))

組織における品質指向の開発・保守 プロセスをデザイン

システム・ソフトウェアまたは運用支援サービスには、利用者の環境や利用目的によって異なる品質要求がある。たとえば24時間無停止で運用できる程度まで信頼性を高める要求や、高齢の利用者の使いやすさを高める要求が挙げられる。一方、1つの開発組織で、複数分野に関連した数種類のシステムを扱う場合があり、1つのシステムの中でも、サブシステムや機能別に異なる品質が求められることもある。

このためISO/IEC 25010¹⁾にあるような品質(副)特性ごとの要求事項と、それらの重要度や開発のリスクの内容と大きさを分析し、サブシステムや機能別のリスク対策をもとに、システム開発や保守プロ

セスのライフサイクルや詳細なプロセスをデザインすることが、品質を作り込み、管理するために効果的である。

図-1に示すように、開発組織は一般に要求分析、設計、実装、テストという段階を設けた標準開発プロセスを決めて、段階ごとに品質管理およびプロジェクト管理を行う。要求分析段階では、サブシステムや機能別の品質要求分析も行い、それぞれの品質要求の重要度と開発リスクを明確にする。

これらに対応させて、各段階の中で、品質実現対策を設計・実装に作り込む作業や検査活動、品質測定・評価と品質改良を繰り返す機会を増やすようにプロセスを設計することができる。たとえば、幅広い年齢層や地域、身体的特徴を持った利用者とのユニバーサルな対話型インタフェースや、複雑な機構の長時間制御、高速の実時間処理など、品質要求の

開発リスクが大きいサブシステムや機能は、設計・実装・テストを反復し、複数回の測定・評価・改良を繰り返せるとよい。これには、アジャイル手法も含めイテレーションプロセスが適している(図-1②③④)。一方、開発組織が品質要求の実現方法に経験を持っており、リスクが小さいサブシステムや機能は反復せずオータフォールによる順次作業で開発することが可能であろう

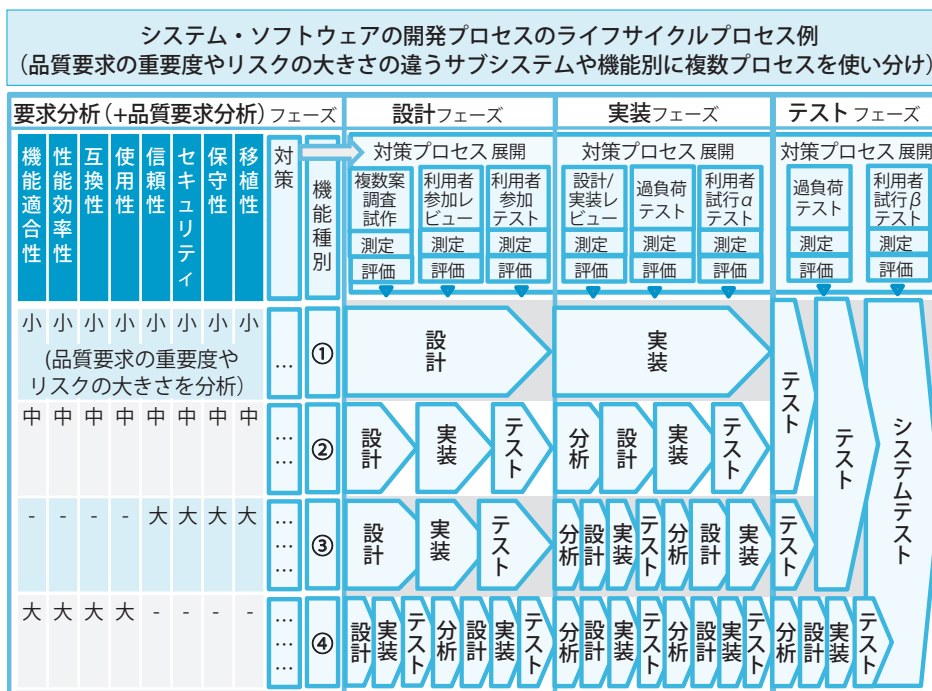


図-1 開発プロセスのライフサイクル例

(図-1 ①).

さらに、品質に関連するリスク対策を、実施する段階やイテレーションに展開して割り当てる。これにより、プロジェクトや開発組織の中で、特徴のある品質要求と開発リスクに対応できる柔軟で選択可能な複数のプロセスを組み合わせ、標準プロセスをデザインすることができる。

品質特性から考えるリスクベースの品質向上施策

● 品質向上施策立案上重要な原則

品質を確保する活動は、期間とコストの壁があり潤沢に実施できないことが多い。そのため有限の期間とコストで最大効果を持つ設計と検査の戦略が求められる。以下の原則は戦略立案上重要である。

原則1：運用で重要な品質の達成を重視する

システム開発は「利用者に情報システムが果たす価値を届ける」ために実施する。したがって、情報システムが利用者の運用環境で持つべきより重要な品質の達成を優先することが望ましい。ISO/IEC 25010¹⁾で定義する利用時品質で考えると以下のようになる。

- ① **リスク緩和性**：重大な問題を起こさないことは最も重要である。情報システムがもたらす可能性のある経済、健康・安全、環境への悪影響、その潜在リスクを考える。
- ② **有効性・効率性**：次に重要なのは情報システムが本来目指す目標、すなわち利用者が情報システムを利用することに対する有効性と効率性に寄与できることである。
- ③ **満足性・利用状況網羅性**：最後に利用者の満足を得ることができる優れた実用性があり、信用、

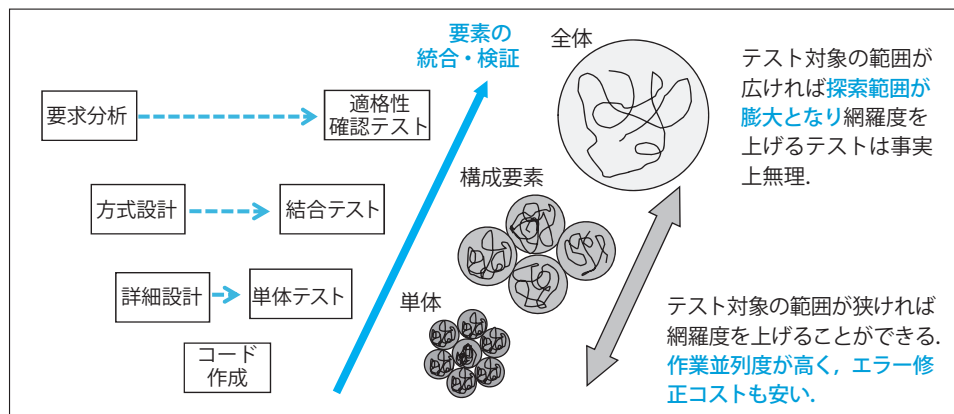


図-2 V字モデルと網羅度の関係

快感、快適を感じられるか、想定外の状況にどれだけ対応できるかである。

原則2：リワークによって開発に大きな影響を与える欠陥を早期に検出する

製品の品質特性、特に性能効率性、信頼性、使用性、セキュリティは、システムを構成する要素個々に分解して割り当てることは困難で、それらが組み合わさって初めて実現できる創発的な特性である。創発的な特性に関する欠陥は、その修正が広範囲に及ぶため、大きなリワークを生じ、開発全体の品質・コスト・期間にインパクトを与えることになる。そのため、開発のなるべく早期に対策を打つことが望ましい。

原則3：対象範囲が狭いところで網羅度を上げる

検査対象の範囲が狭ければ、網羅度を上げやすい。図-2に示すようにV字モデルの右のテストでは、なるべく早い段階で網羅度を上げるとよい。

ある統合状態で実施するテストは、その統合によって可能になる検査に集中することが望ましく、コーディングミスのような欠陥は単体レベルで検出する方がずっと効率がよい。システムを統合した状態でいくら頑張っても、コードカバレッジが50%以下であるという報告もある。

● リスクベース品質向上施策の立案

品質向上と、開発のコスト削減および期間短縮は、開発現場の重大なトレードオフ事項である。決まりきった方法と単一の品質目標のもとで戦略なく品質

8. 品質に対応したプロセスデザイン

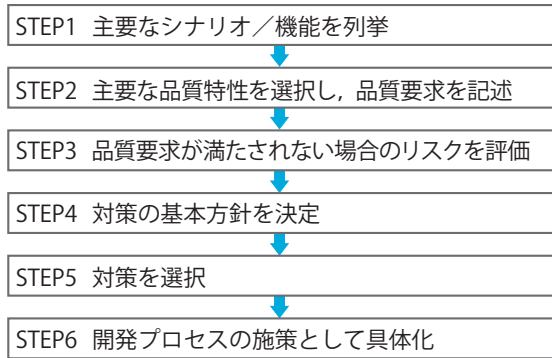


図-3 リスクベースの品質向上アプローチ

向上の活動を実施していたのでは、コストと期間を掛けた割に、重大な不具合が市場に流出しかねず、それによって事業の足をすくわれることさえある。

リスクベース品質向上アプローチは、原則1～3を用い、品質リスクを事業推進上許容できる範囲内に抑えるように開発プロセスを設計する方法である。

図-3にこのアプローチの手順を示すとともに、以下手順を詳述する。図-4はこのアプローチの各STEPを順に実施した例である。

STEP1：主要なシナリオ／機能を列挙

検査対象となる主観点を列挙する。主観点には、システムを利用するシナリオ、主要な機能、システ

ムを構成するサブシステムなどがある。

STEP2：重要な品質特性を選択し品質要求を記述

検査対象で重視すべき品質特性を列挙し、各品質特性に対して測定量とその目標値を明確にする。

STEP3：品質要求が満たされない場合のリスクを評価

以下①と②の2つのリスク観点について品質要求が満たされない場合の影響を評価する。①は原則1、②は原則2に基づいている。

① 重大さ：運用環境で問題が生じた際の重大さ

- 大 重大な事故／社会的な損失に繋がる
- 中 製品の価値が著しく損なわれる
- 小 製品の価値が少なからず損なわれる

② 影響度：リワークの大きさ

- 大 システム全体の設計の見直しに繋がる
- 中 サブシステム／機能ブロック間にまたがる設計変更に関わる
- 小 機能ブロック内にとどまる設計変更である

STEP4：対策の基本方針を決定

STEP3で評価したリスクの対策の基本方針を選択する。対策の基本方針は、重大さと影響度によって決定する。基本方針には後述する3つの方針、①

| シナリオ/機能/サブシステム | 重視すべき品質特性 | 品質要求 | 重大さ① | 影響度② | 対策の基本方針 | 可能な対策 | コスト | 効果 | 採否 | 要求分析 | 方式設計 | 詳細設計 | コーディング | 単体テスト | 結合テスト | 適格性確認テスト | |
|----------------|-----------|-------------------------------------|------|------|-----------|------------------------|-----|----|----|------|------|------|--------|-------|-------|----------|---|
| XX表示機能 | 機能完全性 | 現行機能を網羅していること | 大 | (不明) | 検査網羅度の向上 | 現行機能仕様との追跡性の確保とレビューの徹底 | △ | ○ | 採 | ○ | ○ | ○ | | | | ○ | |
| | | 現行データによる運用が可能であること | 大 | (不明) | 検査網羅度の向上 | 疎通テストの実施 | △ | ○ | 採 | | | | | | | ○ | ○ |
| | 機能正確性 | 位置、数量などの表示データに誤りがないこと | 大 | (不明) | 検査網羅度の向上 | テスト網羅度の向上 | △ | ○ | 採 | | | ○ | | | | ○ | |
| | | | | | | 観点レビューの導入 | ○ | △ | 採 | | | ○ | ○ | | | | |
| | 使用性 | 画面操作と表示内容が、操作員に誤操作を引き起こさないこと | 中 | 大 | 早めのV&V | 仕様調整用プロトタイプ開発 | × | ○ | 採 | ○ | | | | | | | |
| | | | | | 検査網羅度の向上 | システムテスト：現行テストによる確認 | × | ○ | 採 | | | | | | | | ○ |
| | 時間効率性 | すべての表示が3秒以内であること | 中 | 大 | 設計品質の向上 | タイミングチャートの利用 | ○ | △ | 採 | | ○ | | | | | | |
| | | | | | 検査網羅度の向上 | コーディング規約の利用 | ○ | △ | 採 | | | | | ○ | | | |
| 観点レビューの導入 | | | | | 観点レビューの導入 | ○ | △ | 採 | | | | ○ | ○ | ○ | | | |
| YY管理機能 | 機能正確性 | システムの状態の誤った把握がなくかつ遅れなく(5秒以内)把握できること | 大 | 大 | 早めのV&V | 擬似運用環境構築 | △ | ○ | 採 | | | | | | | ○ | |
| | | | | | 検査網羅度の向上 | 観点レビューの導入 | △ | ○ | 採 | | ○ | | | | | ○ | ○ |
| | 障害許容性 | 24時間運用、可用性99.94%以上 | 大 | 大 | 早めのV&V | 複合条件化での網羅度向上 | △ | ○ | 採 | | | | | | | | |
| | | | | | 設計品質の向上 | 擬似運用環境の構築 | × | × | 否 | | | | | | | | |
| | | | | | 設計品質の向上 | サーバ2重系採用と可用性設計 | △ | ○ | 採 | | ○ | | | | | | |

図-4 リスクベースの品質向上施策立案手順の実施例

設計品質の向上, ②検査網羅度の向上, ③早めの V & V (Verification and Validation) があり, その中から以下のように, 相応しい方針を選択する。

「**重大さ**」が大の場合：当該品質に関する欠陥の混入を防止すること, 混入した欠陥を可能な限り検出すること, の両方面から活動する。それぞれ①設計品質の向上と②検査網羅度の向上に対応する。

「**影響度**」が大の場合：当該品質の達成をなるべく早期に確認する。③早めの V & V が対応する。

STEP5：対策を選択

対策の基本方針を実装する可能な対策を列挙する。対策にかかるコストと期待効果のトレードオフで採否を決める（図-4の例では○△×の3段階で評価）。重大さあるいは影響度が大のものを放置することはできないので, 何らかの対策を選択する。以下に①～③の基本方針に対する可能な対策の例を示す。

- ①設計品質の向上：設計をしっかりと実施することで, 品質の向上とバラツキの抑制を図ることができる。対策例として, 設計モデル・技法・パターンの利用, 外部専門家の利用などがある。
- ②検査網羅度の向上：ほかの検査観点に比べて, レビュー密度, およびテストの網羅度（観点, 因子, 水準, 組合せ）を上げる。対策例として, 観点レビューの導入, 静的解析の利用, 境界値テストの徹底, テスト組合せ技法の利用などがある。
- ③早めの V & V：検証と妥当性確認を, なるべく早い段階で実施する。対策例として, 使い捨てプロトタイプの利用, シナリオレビューの導入, 実データを利用したテストの早期化などがある。

STEP6：開発プロセスの施策として具体化

STEP5の対策を展開し, 開発プロセス²⁾ごとの施策として具体化する。施策展開のポイントは以下である。

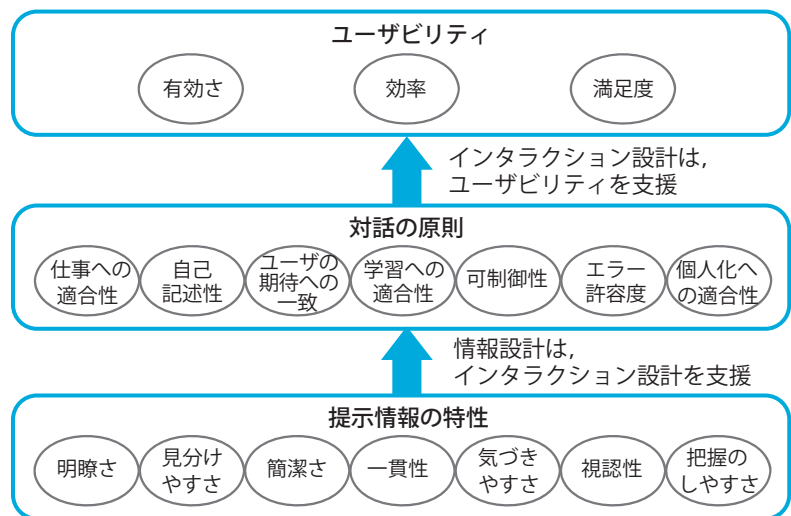


図-5 ユーザビリティ, 対話の原則, 提示情報の特性の関係 (ISO 9241-110⁴⁾ を改版)

- ①設計品質の向上は, 欠陥の混入プロセスに適用
- ②検査網羅度の向上は, 混入欠陥をどのプロセスで除去すれば効率的かを考え展開
- ③早めの V & V は, 早いプロセスで施策を展開

プロセスデザインの詳細

品質に対応したプロセスデザインに関して, 要求分析, アーキテクチャ設計, V&Vの3つの重要なプロセスに焦点を当て, 利用可能なアプローチを紹介する。

●人間中心設計の開発プロセスへの取り込み

いかなるシステム・ソフトウェアにおいても, それらの利用者が存在する以上, 製品品質はもちろん, 利用時の品質を高めることは提供側の責務である。この利用時品質は, 前述のように ISO/IEC 25010 で規定されており, 人間工学規格 ISO 9241-11³⁾ で規定されているユーザビリティの定義を含んでいる。また, ISO/IEC 25010 では, 製品品質モデルを定義しているが, その中で「使用性」が取り上げられている。これは, 同じく人間工学規格 ISO 9241-110⁴⁾ の対話の原則を参照している。さらに, ISO 9241-12⁵⁾ で, 提示情報の特性が規定され, これらの規格は図-5に示すような関係になっている。

ユーザビリティは, 機能や価格と並び, 顧客がシ

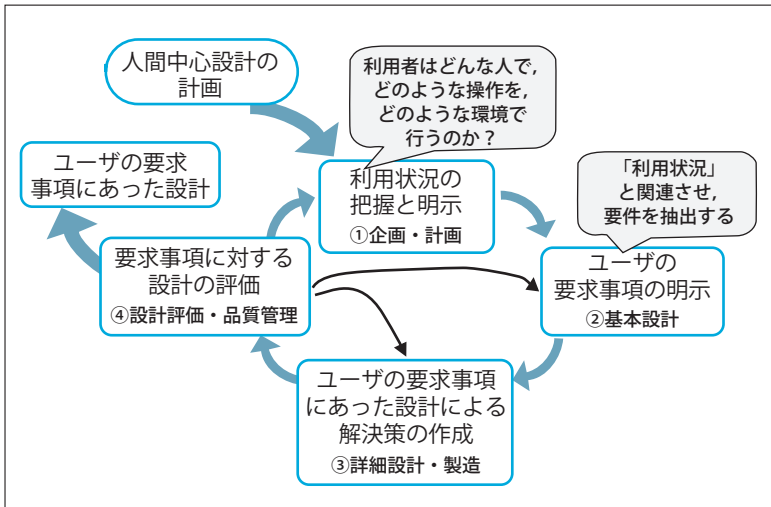


図-6 人間中心設計活動の相互依存性（出典：HIS2012）

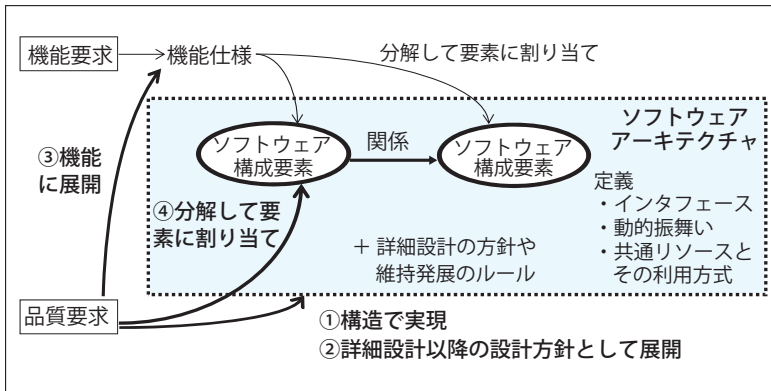


図-7 品質要求のアーキテクチャ設計への展開

システムやソフトウェアに求める重要な要素となってきた。しかし、ほかの品質要求と同様、ユーザビリティに関する要求事項として明確に定義することが難しく、しばしば開発手戻りの発生や利用時品質の低下を招いてしまう。このため、工数やコストを増加させることなくユーザビリティを向上させるために、開発の上流段階でユーザビリティに関する顧客要求を明確にする必要がある。

この課題を解決し、ユーザビリティを高める手段の1つとして、図-5に示すそれぞれの特性や原則を捉える、人間中心設計の考えを開発プロセスに取り込む方法がある。これは、利用者に焦点を当て、利用者が求めるシステム・製品を開発できるようにする考え方／手法である⁶⁾。図-6に人間中心設計の考えに基づいた活動を示す。この図に示すように、人間中心設計の考え方は4つの活動から構成されており、それぞれが開発プロセスに組み込まれて実施

される必要がある。

これらの活動の中で、特に「利用状況の把握と明示」、 「利用者の要求事項の明示」が、開発の上流段階で特に重要である。設計のアウトプットとしては、「代表利用者と主な対象業務の抽出」、「顧客要求明確化」、「要求実現のための要件抽出」、「画面操作・表示イメージの具体化」であり、さらに、設計工数に影響を与える要件を明らかにし、顧客との合意形成に繋げられることも行う⁷⁾。

このように、開発プロセスに人間中心の考え方を取り込むことによって、システム・製品の利用時品質は高められると考えられる。しかし、実際にプロジェクトで適用可能とするためには、開発者がこの考え方をいかに負担なく適用できるようにするか、が重要となってくる⁸⁾。

● 品質に対応したアーキテクチャ設計 (AD) プロセス

品質要求の展開

品質要求の多くの部分はアーキテクチャ設計 (AD) で実装される。アーキテクチャとは、構成要素と、構成要素間および構成要素—環境間の関係、設計と発展をガイドするルールに具体化されたシステムの基本的な構造（組織）である⁹⁾。

機能要求が、機能要求を分解して、構成要素に割り当てることで実現できるのに対して、品質要求の実現はもっと複雑である。品質要求は次の4つの方法で展開される（図-7）。

① 構造で実現

ソフトウェアアーキテクチャで実現する。たとえば呼び出し関係が整理されたレイヤ構造はソフトウェアの保守性を高め、適切な並列タスクの設計は効

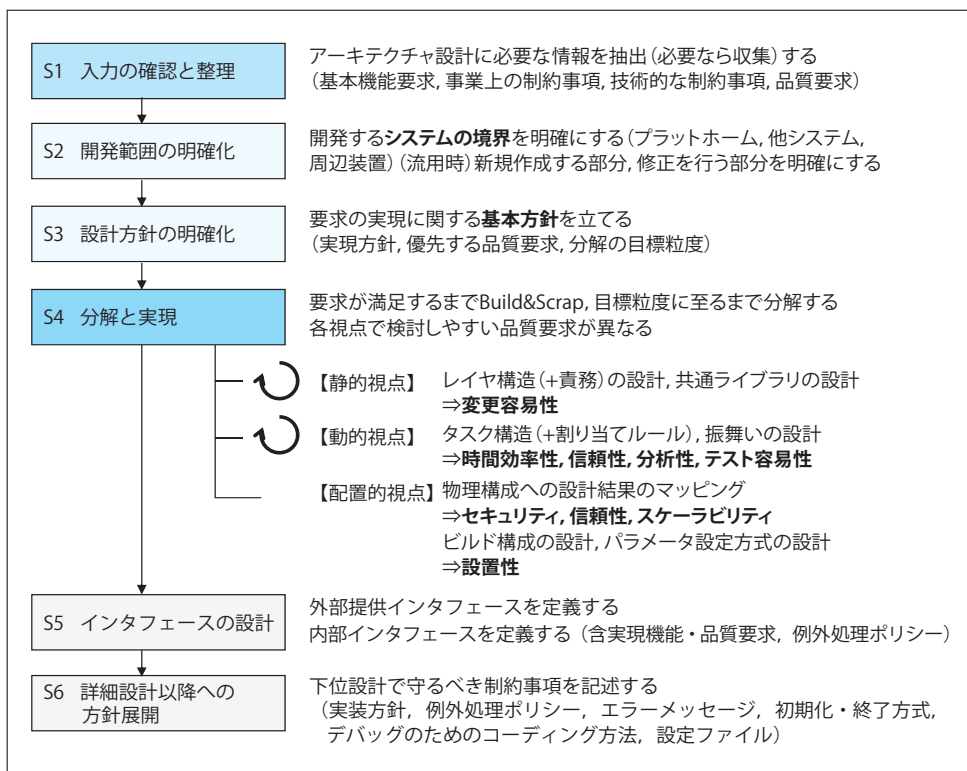


図-8
品質要求を実現
するADのプロセス

率性を高める。

② 詳細設計以降の設計方針として展開

アーキテクチャを壊さないための詳細設計ルール, 例外処理のポリシー, セキュリティ要求を満たすためのコーディングルールなど, 詳細設計以降の設計の進め方に対する方針として展開される。

③ 機能に展開

品質要求の中には機能として実現されるものがある。たとえば, セキュリティ要求がアクセス制御機能, 改ざん防止機能に, 分析性要求がログイン機能に展開されるなどがある。

④ 分解して要素に割り当て

品質要求の中には, 分解され機能の構成要素に割り当ててあるものがある。たとえば, 応答時間要求は要素ごとの処理時間に展開される。

品質要求を実装するプロセス

品質要求には, 絶対満足すべきものからなるべくなら満たしておきたいものまで優先順位がある。それらの適切なバランスが重要である。そのためADは, アーキテクチャ候補の作成と品質要求面からの評価を繰り返す試行錯誤のプロセスとなる。 **図-8**

にADのプロセスの一例を示し, これを用いて説明する。

S1で, 要求分析結果からADに必要な情報を確認・整理し必要なら再獲得する。S2で開発範囲を明確にし, S3でADを進める基本方針を決める。S4がADの核心部である。制約事項を満足し品質要求を最大限実現するためのアーキテクチャを作り上げるために, 静的/動的/配置的の各視点で構成要素への分解を進めていく。この過程は試行錯誤的である。品質要求ごとに, アーキテクチャの表現に適した視点が異なるので, 適切な視点を選択し, 設計と評価を繰り返す⁹⁾。さらに, 要求からのトレーサビリティ確保のため, 設計したアーキテクチャがどの品質要求をどのように満足しているかを設計根拠として残す。複数候補から選択した場合には, 品質要求に対するトレードオフ表を作成し文書化することでその結果を残すことも必要である。

S5とS6は, ADの成果物の利用者が必要とする情報を提供する。

● 検証と妥当性確認のための V&V プロセス

要求分析された品質特性ごとの品質要求事項やその実現への開発リスク対策項目が設計や実装で反映されているかを、レビュー・テストで確認する。結果は問題指摘や不具合として検出・測定し、許容範囲内かどうかを比較検討して検証する。

検査網羅度の向上が特に重要な項目については、

- ① プロジェクト全体より目標基準を高くする。
- ② レビューでは、レビュー密度について高い基準を明確に設定する。
- ③ テストでは、テスト対象の範囲および入力値の選択の仕方、並びに組合せ網羅度について高い基準を明確に設定する。

システム利用中に不具合が発生すると影響が大きい項目については、妥当性確認のレビュー・テストケースの網羅性を高める。このため、過負荷や例外・異常処理、入力エラー処理などが起こり得るシステム利用シナリオを増補する。また、直接操作運用する利用者や保守者に加え、直接操作しないが運用の影響を受ける間接的な利用者も含めて、利用目的・利用状況などを網羅するようなシナリオも補う。層別した利用者群から各代表者を選び、テストに参加してもらう方法もある。

参考文献

- 1) ISO/IEC 25010 System and Software Quality Model (2011).
- 2) ISO/IEC 15288 System Life Cycle Processes (2008).
- 3) ISO 9241-11 : Ergonomic Requirements for Office Work with Visual Display Terminals (VDTs) – Part 11 : Guidance on Usability (1998).

- 4) ISO9241-110 : Ergonomics of Human-system Interaction – Part 110 : Dialogue Principles (2006).
- 5) ISO 9241-12 : Ergonomic Requirements for Office Work with Visual Display Terminals (VDTs) – Part 12 : Presentation of Information (1998).
- 6) ISO 9241-210 : Human-centred Design for Interactive Systems (2010).
- 7) 福住, 谷川, 池上 : ユーザビリティ関連規格の現状と活用方法, ヒューマンインタフェースシンポジウム 2012, pp.147-152 (2012).
- 8) 谷川, 大久保, 福住 : ソフトウェア技術者がユーザビリティ向上に取り組む上での課題の考察～人間中心設計プロセス支援環境の検証実験を通じて～, ヒューマンインタフェースシンポジウム 2013, pp.167-170 (2013).
- 9) Lumsde, A. J. : Architecture-centric Design Method, (邦) アーキテクチャ中心設計手法 (Jan. 2011).

(2013年9月24日受付)

● 中島 毅 (正会員) Nakajima.Tsuyoshi@ap.MitsubishiElectric.co.jp

三菱電機・設計システム技術センター主管技師長。早稲田大学理工学研究科修士課程修了。博士(工学)。ソフトウェア工学専門。ISO/IEC JTC 1/SC 7/WG 6 国内・国際委員。

● 山田 淳 (正会員) atsu.yamada@toshiba.co.jp

(株)東芝ソフトウェア技術センター主幹。大阪大学大学院工学研究科修士課程修了。ソフトウェア工学、プロセス・品質技術に従事。ISO/IEC JTC 1/SC 7/WG 6 国内・国際委員。

● 福住伸一 s-fukuzumi@aj.jp.nec.com

NEC 情報・ナレッジ研究所技術主幹。慶應義塾大学大学院工学研究科修士課程修了。工学博士、認定人間工学専門家。HI学会理事。ISO TC 159/SC 4 (HCI) 国内委員会副主査および ISO/IEC JTC1/SC 7/WG 28 国際セクレタリー。