

# 事例に見る先進的な 設計・検証技術の適用分析

---

～高信頼化のための開発技術導入に向けて～

独立行政法人情報処理推進機構（IPA）  
技術本部 ソフトウェア高信頼化センター（SEC）

# 事例に見る先進的な 設計・検証技術の適用分析

---

～高信頼化のための開発技術導入に向けて～

独立行政法人情報処理推進機構（IPA）

技術本部 ソフトウェア高信頼化センター（SEC）

# 刊行にあたって

独立行政法人情報処理推進機構 技術本部  
ソフトウェア高信頼化センター  
所長 松本 隆明



我々の社会活動はもはやITなくしては考えられない時代となってきました。こうしたITを実現するソフトウェアの役割はますます重要となってきており、その信頼性や安全性の確保は、安定した社会活動を実現するうえで必要不可欠となりつつあります。その一方で、IoTの爆発的な進展など、ITシステムを取り巻く環境は急激に変化しつつあります。今後、さらに高度化・複雑化するITシステムの高信頼化を達成するためには、最新の有効な設計技術を適用して、効果的に開発を進めていく必要があります。

独立行政法人情報処理推進機構技術本部ソフトウェア高信頼化センター（IPA/SEC）では、先進的な設計技術の開発現場への導入がスムーズに進むよう、上流工程での先進的な設計方法の効果的な適用事例と信頼性検証技術の活用事例を収集し、それを元に58件の適用事例を報告書として作成して公開しています。

本書は、IPA/SECが中心となって、それらの適用事例を整理、分類、分析したものです。実際の開発現場における先進技術の適用事例を分析し、開発時の課題解決に向けて先進技術を効果的に適用するための道筋を示したものは、他に類を見ない貴重なものです。

本書では、まず、それぞれの適用事例を5つの軸（適用課題・適用領域・適用工程・適用状況・適用技術）で整理、分類し、想定される読者の視点（複数のプロジェクトを統括している者・ある領域に特化している者・技法に特化して研究開発している者）でそれぞれの分析の結果を解説しています。

また、読者の置かれている領域とIT開発の課題項目から、その解決策となり得る技術と、既に公開済みの58件の事例の中の参考になり得るも

のを紐付けることができます。これらによって、現場で多くの課題を抱えている読者に、課題解決のヒントを提供できると期待しています。

IPA/SECでは、本書以外にも、現場の課題を解決するための開発手法（モデルベース開発や形式手法など）の活用ガイドなどをIPA/SECウェブサイトにて公開していますので、ぜひご活用ください。

最後に、本書の成果は、先進的な設計・検証技術をいち早く取り入れ、試行錯誤している団体の方々が、積極的に現場の生データをご提供くださったものです。深く感謝申し上げます。また、収集した適用事例を分析するための分類軸の設定には、2013年度の高信頼設計・検証技術WGと2014年度の開発手法適用のための分析WGに参画して頂きました委員の皆様のご協力の賜物です。この場を借りまして深くお礼申し上げます。また、情報サービス産業協会（JISA）様には「情報サービス産業における技術成熟度調査」に、適用事例で紹介した先進的な設計・検証技術のうち29の技術項目を記載して頂きました。改めて御礼申し上げます。

IPA/SEC ウェブサイト

<http://www.ipa.go.jp/sec/reports/index.html>

# 課題解決のヒント探索を 大幅に効率化できる実践書 — 刊行によせて



名古屋大学 大学院  
情報科学研究科教授  
山本 修一郎

本書のねらいは、「58件の先進的設計・検証技法の適用事例を分析した結果を解説すること」、「IT開発運用保守の課題とその解決策としての技法を関連付けること」の2点である。

本書では、収集した58件の適用事例を分析するために、適用課題、適用効果、適用領域、適用工程に着目したキーワードを用いて、以下のように整理している。

適用課題と適用効果については、1. 品質（製品品質8特性、利用品質5特性）、2. 保証ケース、3. 障害原因分析、4. コスト低減、5. 納期短縮、6. 生産性向上、7. 人材育成、8. プロジェクトマネジメント、9. 見積精度向上、10. 普及促進、11. 組織改革、12. グローバル展開に分類している。58事例で287個の課題が取り上げられており、事例あたり平均約5件の課題が検討されていることがわかる。

適用領域については、1. 設計系と検証系、2. エンタープライズ系、Webフロント系、組込み・制御系に分類している。

適用工程については、1. 企画、2. 要件定義、3. 開発（A：システム・ソフトウェア要件定義、B：基本設計、C：詳細設計、D：製作・単体テスト、E：テスト）、4. 移行・運用準備、5. 運用・保守に分類している。

適用状況については、1. 課題定義、2. 解決策実施、3. 効果測定、4. 継続的課題定義、5. 継続課題への取組計画があるかどうかという5観点で各事例を詳細に分析している。

58件の適用事例で用いられた技法は、開発モ

デル、形式手法、開発検証ツールなど28個である。

また、適用事例の記述では、技法適用の契機、導入判断理由、技法の導入によって達成すべきゴール、適用領域や適用工程など現場組織の環境、適用効果ならびに適用後の継続的取り組みなどについて、具体的にまとめている。

このように本書では、多面的な観点から、先進的設計・検証技法の現場への導入状況を解説している。したがって、本書で用いられたこれらの観点は、各技法の適用状況を客観的に解説するのに役立つだけでなく、適用時に準備すべき事柄を明らかにしている。系統的に整理された観点に基づく、技法の精密な解説は本書の大きな特徴である。

また、本書のもうひとつのねらいである「IT開発運用保守の課題とその解決策としての技法の関連付け」では、ナビゲーションシートに基づき、適用領域、適用工程、期待効果の順で段階的に適用事例を絞り込む手順を提示している。ナビゲーションシートでは、1. 課題、2. 期待効果、3. 適用領域、4. 適用度合、5. 読者属性、6. 測定方法、7. 適用工程を記述する。58件もの適用事例の中から適切な事例を探索するのは容易ではない。このようなナビゲーションシートによる適用事例の絞り込み手順によって、課題解決のヒント探索を大幅に効率化することが期待できる。

## Contents

- ・刊行にあたって ..... 2
- ・課題解決のヒント探索を大幅に効率化できる実践書——刊行によせて ..... 3

### 1章 はじめに ..... 8

- 1.1 本書の構成 ..... 9
- 1.2 背景 ..... 10
- 1.3 本書の目的 ..... 16
- 1.4 想定対象読者 ..... 17
- 1.5 本書の位置づけ ..... 19
- 1.6 本書の活用方法 ..... 20

### 2章 事例の収集・分類・分析・普及の概要 ~取組み紹介~ ..... 22

- 2.1 適用事例収集方針 ..... 23
- 2.2 抽出・整理・分析方針 ..... 34

### 3章 適用事例及びアンケート結果の分析 ..... 56

- 3.1 全適用事例の分析 ..... 57
- 3.2 適用対象分野ごとの分析 ..... 69
- 3.3 適用技術・手法ごとの分析 ..... 110
- 3.4 アンケート ..... 116

### 4章 課題解決のヒントになる 適用事例や適用技術・手法を紐付け ..... 126

- 4.1 読者が置かれている環境をナビゲーションシートに可能な範囲で記入 ..... 128
- 4.2 期待する効果の項目の洗い出し ..... 130
- 4.3 適用事例分類表から適用領域①と適用事例②で抽出 ..... 131
- 4.4 さらに適用工程で抽出 ..... 133
- 4.5 さらに期待する効果項目で抽出 ..... 134
- 4.6 さらに特に期待する効果項目で絞込 ..... 137
- 4.7 先進的な技術・手法の適用 ..... 140

---

**5章** 情報サービス産業における  
情報技術マップに関する調査報告 ..... 144

- 5.1 調査方法と分析の概要 ..... 148
- 5.2 2015年度調査 ..... 149
- 5.3 回答状況 ..... 149
- 5.4 調査結果 ..... 150

---

**6章** 関連情報 ..... 154

- 6.1 国際標準など ..... 154
- 6.2 知識体系 ..... 155
- 6.3 SEC 成果物 ..... 156

---

**7章** 適用事例の概要 ..... 160

- 7.1 設計編 ..... 160
- 7.2 検証編 ..... 172

---

**付 録** ..... 184

## 本書の内容に関して

---

- ・本書を発行するにあたって、内容に誤りのないようできる限りの注意を払いましたが、本書の内容を適用した結果生じたこと、また、適用できなかった結果について、著者、発行人は一切の責任を負いませんので、ご了承ください。
- ・本書の一部あるいは全部について、著者、発行人の許諾を得ずに無断で転載、複写複製、電子データ化することは禁じられています。
- ・乱丁・落丁本はお取り替えいたします。下記の連絡先までお知らせください。
- ・本書に記載した情報に関する正誤や追加情報がある場合は、IPA/SEC のウェブサイトに掲載します。下記の URL をご参照ください。

独立行政法人情報処理推進機構 (IPA)  
技術本部 ソフトウェア高信頼化センター (SEC)  
<http://www.ipa.go.jp/sec/index.html>

## 商 標

---

- ※ Microsoft<sup>®</sup>、Excel<sup>®</sup> は、米国 Microsoft Corporation の米国及びその他の国における登録商標又は商標です。
- ※ Subversion、Apache Subversion および Apache Subversion ロゴは、Apache Software Foundation の米国およびその他の国における登録商標または商標です。
- ※その他、本書に記載する会社名、製品名等は、各社の商標又は登録商標です。
- ※本書の文中においては、これらの表記において商標登録表示、その他の商標表示を省略しています。あらかじめご了承ください。

## はじめに ..... 8

1.1	本書の構成 .....	9
1.2	背景（IT 業界の現状と環境変化） .....	10
1.3	本書の目的 .....	16
1.4	想定対象読者 .....	17
1.5	本書の位置づけ .....	19
1.6	本書の活用方法 .....	20

# 1章 はじめに

本書は、ソフトウェアの高信頼性を確保するため、開発プロセスに先進的な技術・手法を適用した事例を収集し、分析したものである。複雑化・高度化する情報処理システムを実現するソフトウェアについて、その高信頼性を確保するためには、上流設計工程と検証工程を中心に先進的な技術・手法を適用することが望ましい。しかし、対象となる業種・業態をはじめ、サービス・システムは多岐にわたっている。また、適用されている技術・手法も広範囲である。

過去において、情報システム部門の担当者は、社内業務のシステム開発を中心にしていた。また、組込み・制御系のシステム開発の担当者は、今までの開発技術や手法を踏襲して開発していれば、事足りていた。しかし、今後は、新しい異業種と連携するサービスを開発をすることもありえる。

なお、本書は成功事例を中心とした分析なので、そこから得られた分析結果には偏りがある可能性がある。

本書を通して、サービスやシステム・ソフトウェアを開発の現場で奮闘する読者が、ソフトウェアの高信頼化のトレンドを掴みながら、抱えている課題を解決するヒントになる適用事例を見つけ出し、自らが置かれた環境に適したシステム・ソフトウェア開発（先進的な技術を適用したもの）を検討するきっかけになれば幸いである。

今回、分析で対象とした適用事例は、以下のように IPA ウェブサイトで公開済みである。

- ・先進的な設計・検証技術の適用事例報告書 2013 年度版（2014 年 5 月 30 日公開）  
～ソフトウェア開発の高信頼化を目指した先進的な取組み事例を紹介～  
<http://www.ipa.go.jp/sec/reports/20140530.html>



- ・先進的な設計・検証技術の適用事例報告書 2015 年度版（2015 年 11 月 18 日公開）  
～IoT 時代を見据えたソフトウェア開発の信頼性確保を実現したベストプラクティス 34 事例を紹介～  
<http://www.ipa.go.jp/sec/reports/20151118.html>



## 1.1 本書の構成

本書の柱は、「58件の事例を分析して得られた知見を解説すること」と「解決したい課題とそのヒントになる適用事例や適用技術・手法とを紐付けること」の2点である。本書の構成は以下のとおりである。

1章 はじめに ➡ 背景、本書の目的、本書の位置づけ、本書の活用方法をまとめている。

2章 事例の収集・分類・分析・普及の概要～取組み紹介～ ➡ 本書のベースとなる取り組み内容を紹介し、どのようにして本書ができたかを述べる。具体的には、収集した適用事例の分析方法を説明する。さらに、想定している本書の活用方法を示す。

3章 適用事例及びアンケート結果の分析 ➡ 事例と事例提供者へのアンケートを分析して得られた知見を解説する。

4章 課題解決のヒントになる適用事例や適用技術・手法を紐付け ➡ 読者が抱えている課題を解決するヒントになる事例を本章を参考にいろいろな観点で探し、見つけ出す(図1-1の①)。見つけ出した事例の概要説明(7章)を参照する。さらに、既に公開している適用事例報告書の中にある事例の詳細を参照する(図1-1の②)。

5章 情報サービス産業における情報技術マップに関する調査報告 ➡ 情報サービス産業に従事する人へのアンケートを通じて、本書で取り上げた技術・手法への取組み状況(技術を利用した実績や今後の着意向など)を解説する。

6章 関連情報 ➡ システム・ソフトウェアの品質や開発プロセス、機能安全に関わる規格やシステム・ソフトウェア分野の知識体系を示す。また、課題解決のための取り組みを支援するIPAの成果物を紹介する。

7章 適用事例の概要 ➡ 読者の課題解決のヒントになるかどうかの判断を助けるため、収集した適用事例の概要を解説している。

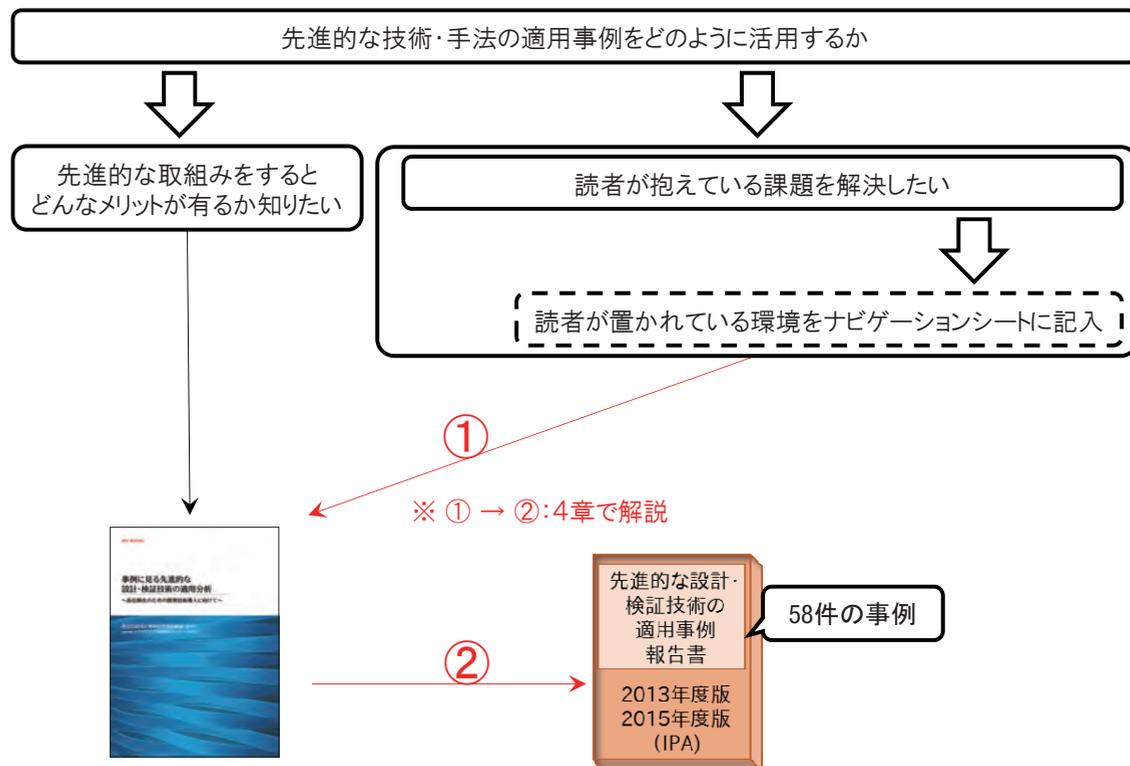


図1-1 本書の活用イメージ

## 1.2 背景 (IT 業界の現状と環境変化)

IT システム (情報システムや組込みシステムなど) は、私達の生活や社会・経済基盤を支える基盤として適用が拡大している。それに伴いソフトウェアの不具合に起因する故障やシステムの停止が社会に与える影響は大きくなっている。

IPA/SEC の調査によれば、多大な影響を与えた IT サービス障害の発生件数は、図 1-2 に示すように 2009 年から 2014 年にかけて、概ね増加傾向にある。

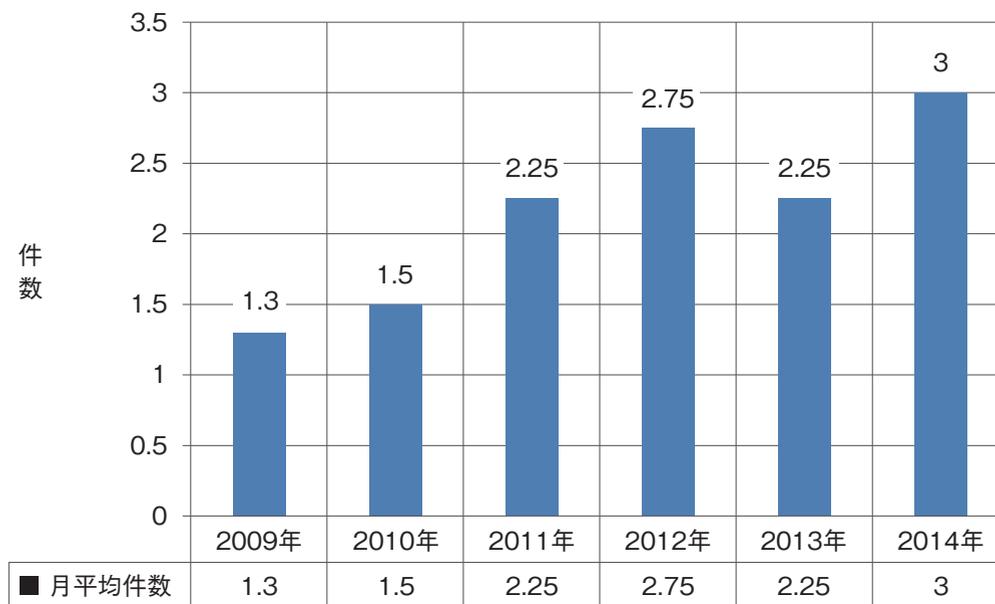


図 1-2 多大な影響を与えた情報システム障害件数 (月平均件数) の推移 (エンタプライズ系)<sup>1</sup>

また、2012 年度の「ソフトウェア産業の実態把握に関する調査」によれば、組込み系では、「なし (不具合発生率 0%)」が 2009 会計年度 (以降「会計」を省略) から大きく減少し、前年度以前の不具合発生割合より高い状態が続いている (図 1-3)。エンタプライズ系ソフトウェア開発企業 (以降、エンタプライズ系ベンダ) では、2010 年度と比べ 2011 年度の結果では「なし (不具合発生率 0%)」が約 5% から約 15% に増加し若干の改善傾向にあった。しかしながら、不具合発生率は決して低い状態ではない (図 1-4)。

<sup>1</sup> 出典：「SECJournal Vol.10 No.6 Mar.2015」(IPA 刊) 情報システムの障害状況 2014 年後半データより作成

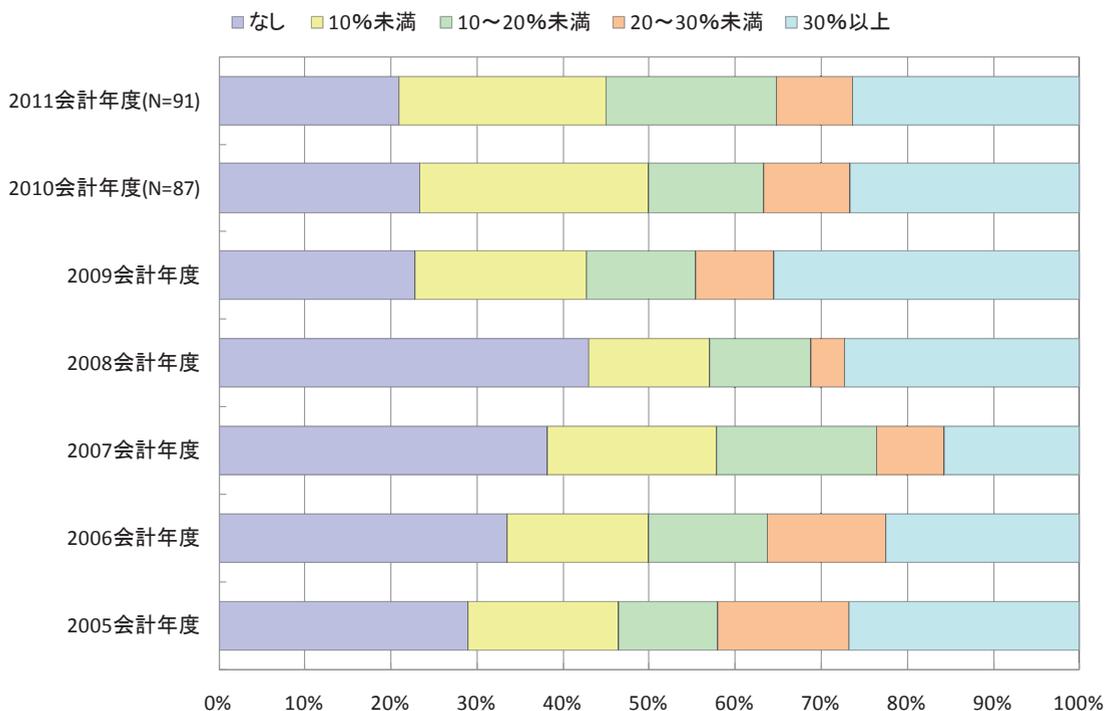


図 1-3 製品出荷後の不具合発生製品率：経年比較 (組込み系)<sup>2</sup>

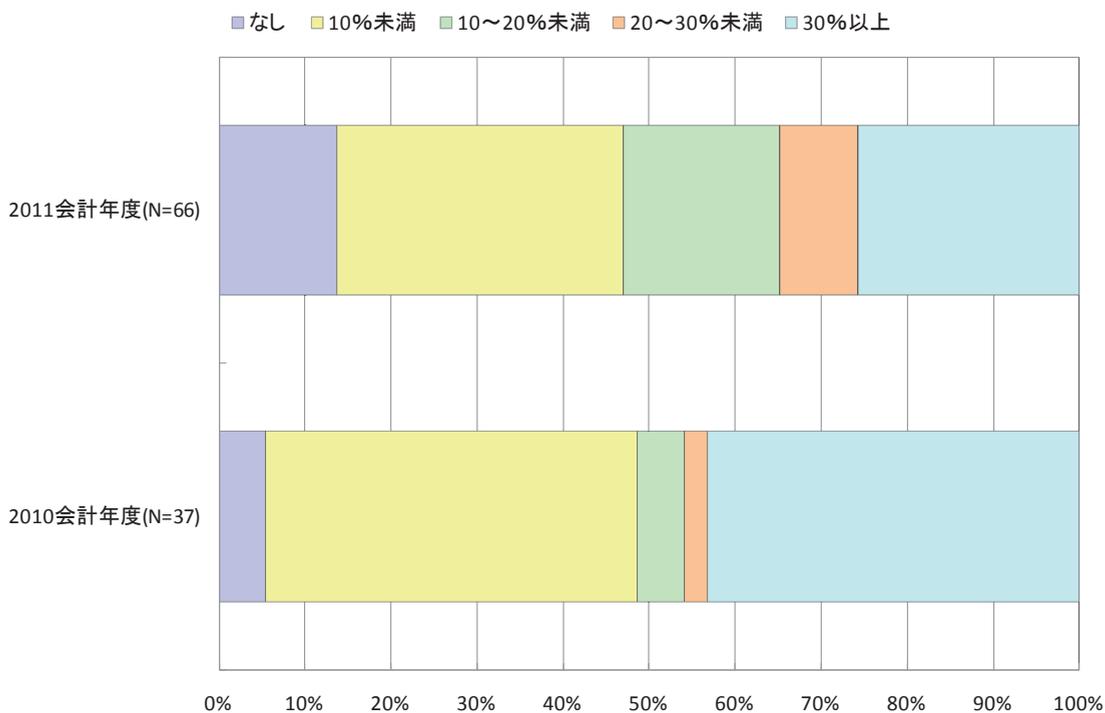


図 1-4 出荷 / カットオーバー後の不具合発生製品 / プロジェクト率：経年比較 (エンタプライズ系ベンダ)<sup>3</sup>

<sup>23</sup> 出典：2012 年度「ソフトウェア産業の実態把握に関する調査」報告書 (IPA/SEC 刊) P14

以上のように、多大な影響を与えた情報システム障害や不具合発生率に大きな改善傾向は見られず、高止まりである。

また、同調査報告書の、「目標 QCD の達成状況」に関する報告内容には、「品質、コスト、納期の全ての目標において、エンタプライズ系ベンダの達成率が組込み系を上回る結果となっている。また、組込み系、エンタプライズ系ベンダともに、品質、コストに比べ納期の達成率が高い」とある（図 1-5）。

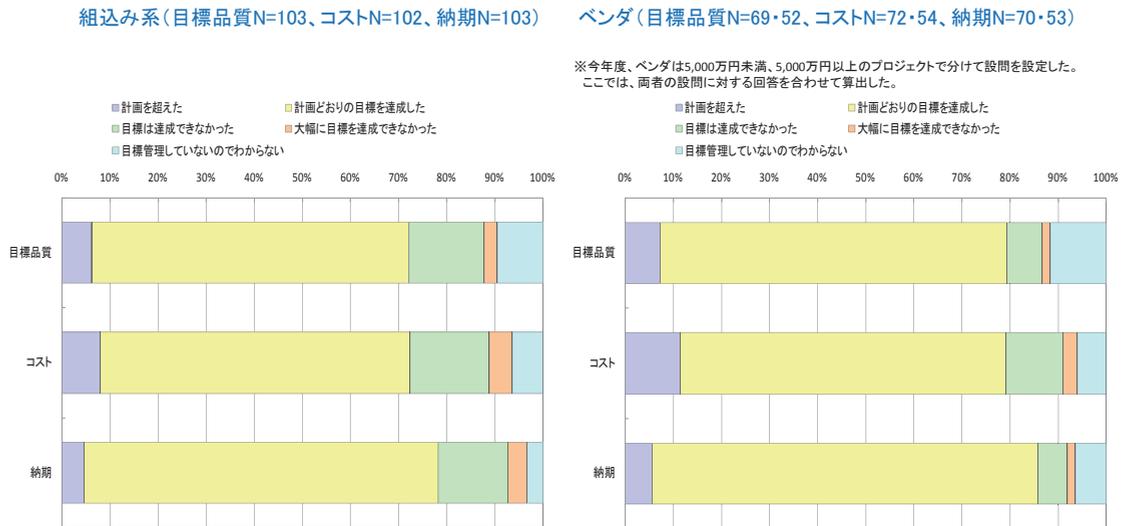


図 1-5 目標 QCD の達成状況 (組込み系とエンタプライズ系ベンダ)

図 1-6 及び図 1-7 は、「目標 QCD の不達成の原因」の理由を調査している。その結果、組込み系では、1 番目の原因として「元々のプロジェクト計画（見積り、目標 QCD 設定等）に無理があった」がトップであり、計画段階に関する課題認識が強いと言える。

## 組込み系(N=79)

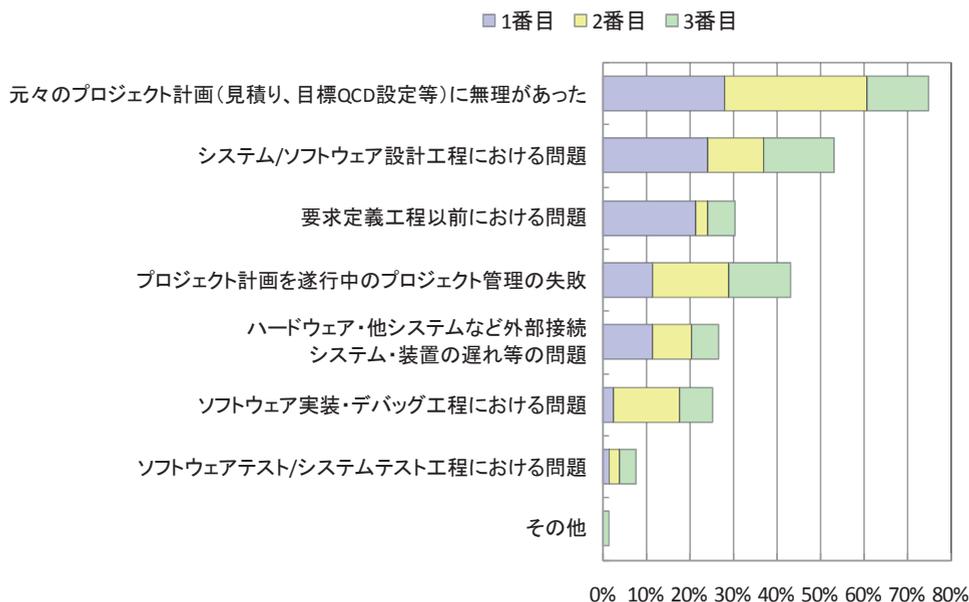


図 1-6 目標 QCD の不達成の原因 (組込み系)

エンタプライズ系ベンダでは、1番目の原因として「元々のプロジェクト計画(見積り、目標 QCD 設定等)に無理があった」がトップで、「システム / ソフトウェア設計工程における問題」、「要求定義工程以前における問題」が上位にある。

## ベンダ(2012年度)(N=30)

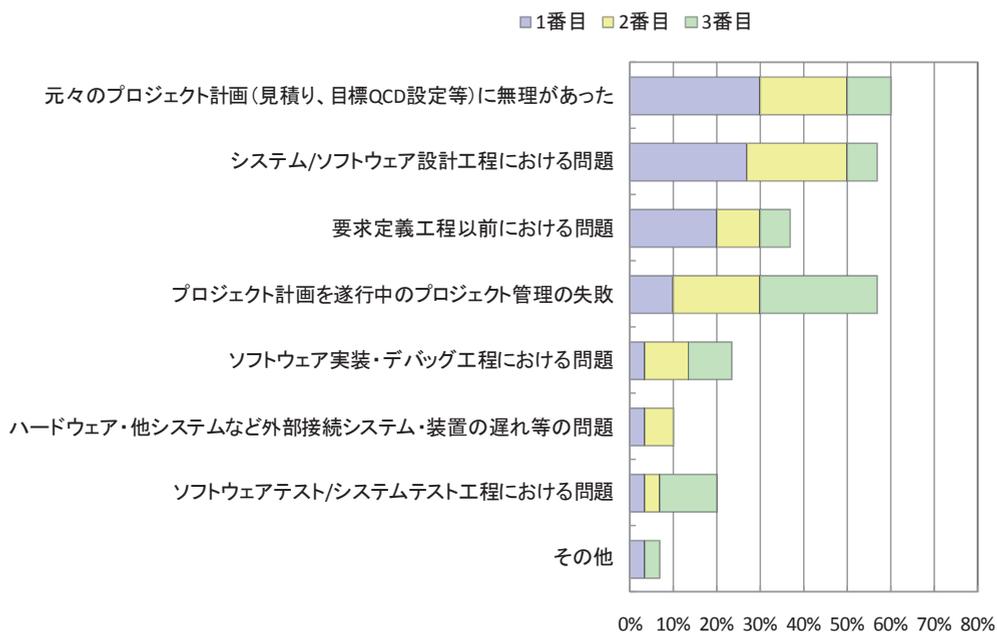


図 1-7 目標 QCD の不達成の原因 (エンタプライズ系ベンダ)

こうした課題に対する解決策について調査したのが、図 1-8、図 1-9、図 1-10 だ。一つ目は、QCD 課題「元々のプロジェクト計画（見積り、目標 QCD 設定等）に無理があった」に対する解決策についてである。組込み系は、1 番目の解決策として、「プロジェクトマネージャのスキル向上」がトップで、人的側面からの取組みを課題解決に効果的と捉えている。また、組込み系もエンタプライズ系ベンダも「管理手法・管理技術（見積り・品質管理等）の向上」という管理的な側面を 1 番目の解決策としてあげている。さらにエンタプライズ系ベンダは「第三者によるプロジェクト・成果物の監査等の体制の強化」が上位にある。

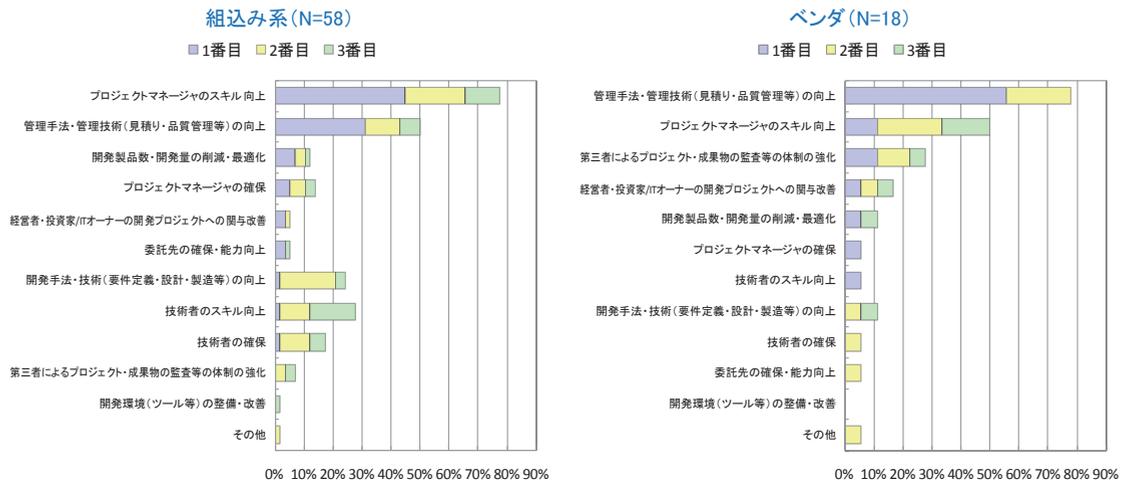


図 1-8 QCD 課題「元々のプロジェクト計画（見積り、目標 QCD 設定等）に無理があった」に対する解決策

二つ目は、「システム / ソフトウェア設計工程における問題」に対する解決策についてである。組込み系・エンタプライズ系ベンダともに、1 番目の解決策として「開発手法・技術（要件定義・設計・製造等）の向上」、「技術者スキルの向上」、「技術者の確保」、「プロジェクトマネージャのスキル向上」の 4 つが上位を占めている。その内の 3 つが人的側面での解決策である。



図 1-9 QCD 課題「システム / ソフトウェア設計工程における問題」に対する解決策

最後は、「要求定義工程以前における問題」に対する解決策についてである。組込み系の1番目の解決策としては、人的側面での解決策が上位を占めている。エンタプライズ系ベンダは「開発手法・技術（要件定義・設計・製造等）の向上」と「経営者・投資家/ITオーナーの開発プロジェクトへの関与改善」が上位にある。

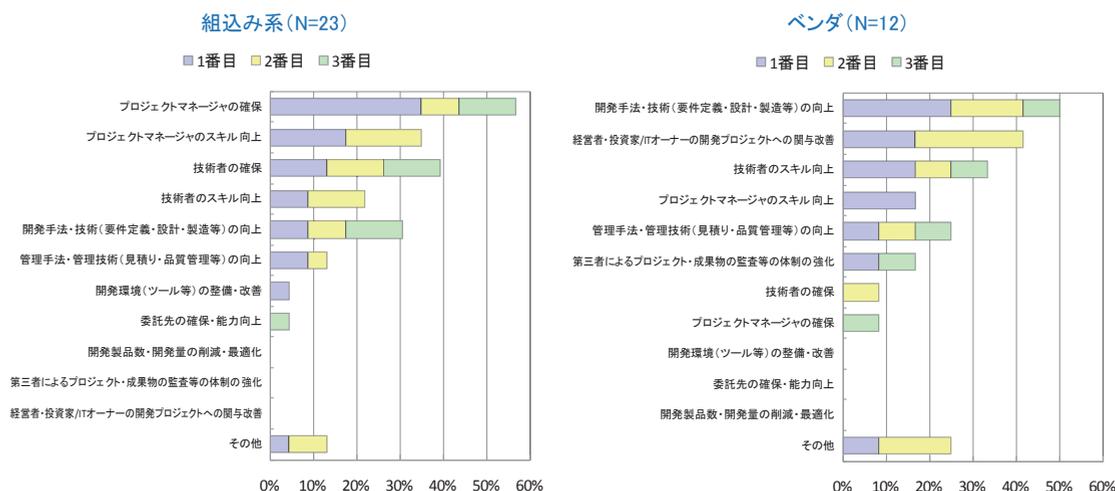


図 1-10 QCD 課題「要求定義工程以前における問題」に対する解決策

近年はコンピュータなどの情報・通信機器だけでなく、世の中に存在する様々なものに通信機能を持たせ、インターネットに接続したり相互に通信したりすることにより、新たなサービスやシステムが構築されている。今まで以上に異業種との連携が密になり、サービスの複雑化、多様化が加速度的に進んでいく。

このような環境の中で、ソフトウェアの高信頼性を確保するためには、その開発プロセスに踏み込んで、見える化や高度化に取り組む必要がある。上述の2012年の調査報告の結果でも示すとおり、目標QCD不達成の原因を解決するための対策として、開発手法・技術（要件定義・設計・製造等）の向上に関する対策が上位に位置している。ソフトウェア開発に関する諸問題に対処するためにソフトウェア・エンジニアリングが1965年に提唱され、以降、色々な開発方法論や手法が国内外で提案され、その有用性も報告されている。しかし、現場レベルでは具体的な導入方法が不明であるとか、効果が見通せないとか、実システム開発における失敗が怖いなどの理由により、広範囲な実用にはつながっていないのが実情である。

とはいえ、そのような中であっても、先進的な技術・手法を導入し、その改善に取組み、実システムの開発にも適用するという実践的な適用事例も多く存在する。

利用者が安心して製品・システムを使えるようにするために、これらの適用事例を数多く収集し、それを共有することにより、我が国のソフトウェア開発の水準を高めていくことは重要だ。本書では、先進的な取組みを行っている企業や団体における適用事例に対し、分析を行った結果を示している。

## 1.3 本書の目的

本書の目的は、読者が抱えている課題を解決するための手引きをすることである。具体的には以下の2点である。

- ① 58件の事例を分析して先進的な技術・手法の適用に取り組むことで、得られた知見を解説する
- ② 解決したい課題とそのヒントになる適用事例や適用技術・手法とを紐付ける

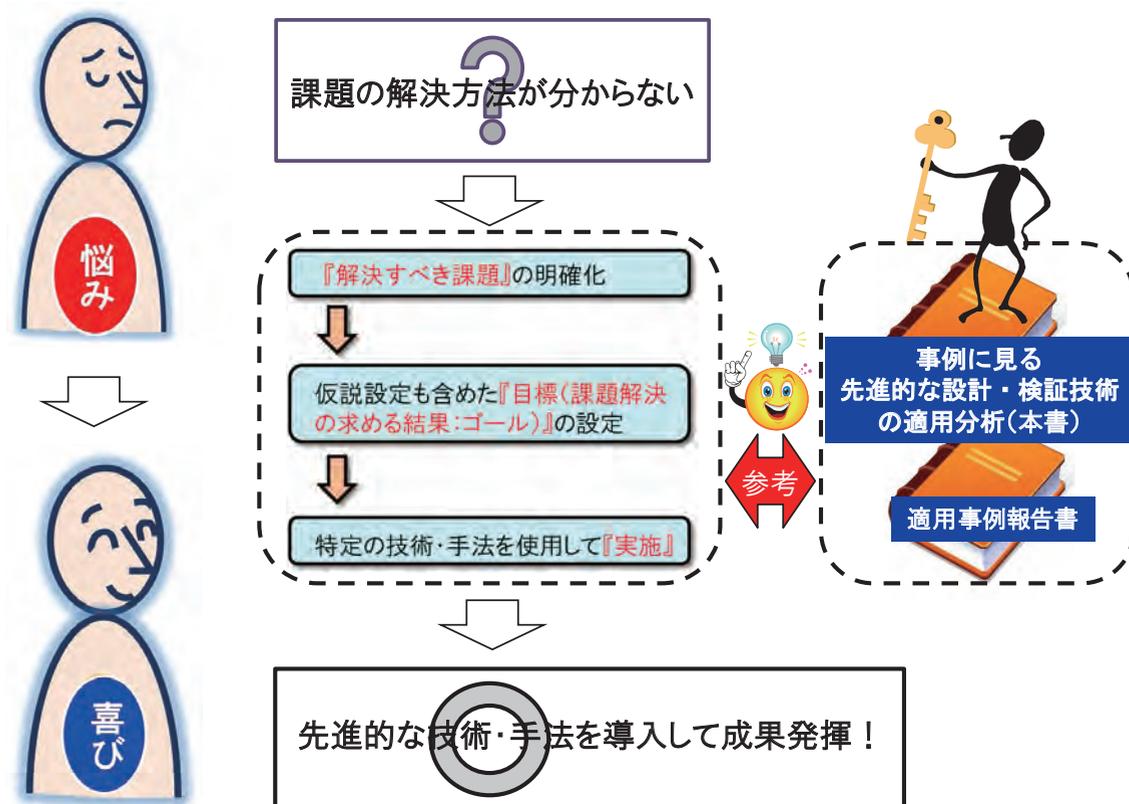


図 1-11 抱えている課題を解決するプロセス

IPA/SECは、本書は図 1-12 のように外部機関等との連携によって先進的な技術・手法の適用事例を収集し、有識者によるワーキンググループ(WG)の活動を介して、適用効果の分析方法を導き出した。今後は、本書と先進的な技法・手法の適用事例を元にセミナーや技術・手法の研修などを行うことで普及促進を図る予定である。

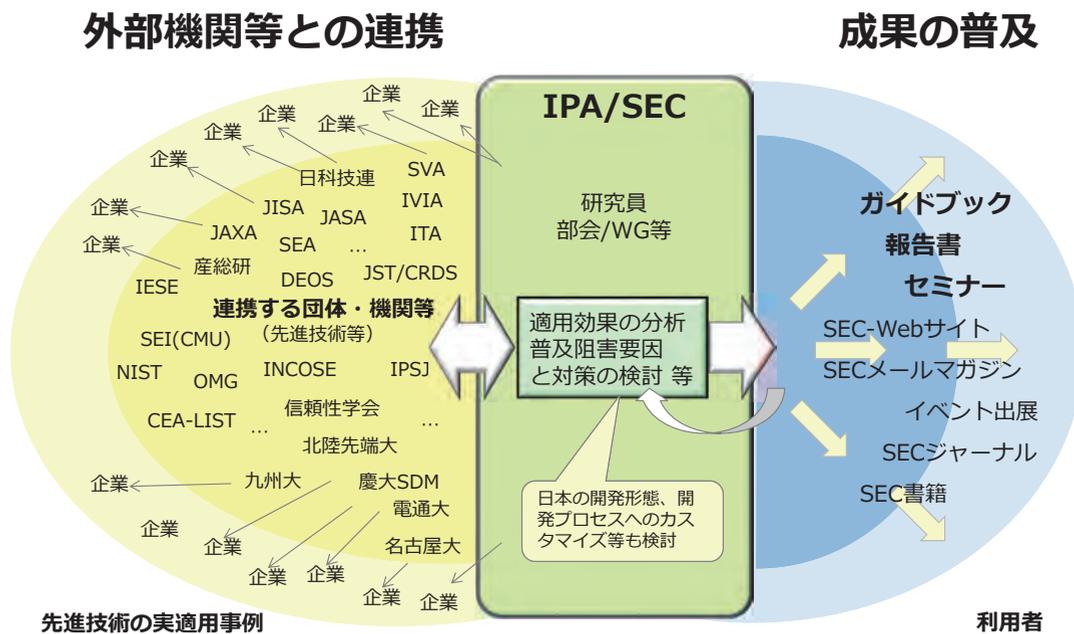


図 1-12 ソフトウェア信頼性の見える化促進のための環境整備

## 1.4 想定対象読者

本書の想定される読者と各読者の活用シナリオを以下のように示す。このように想定したのは、収集した適用事例の提供者が、経営者や統括責任者、プロジェクトマネージャ、プロジェクトリーダー、担当者や大学・研究機関などの研究員と幅広いことと、立場の異なる視点や対応内容の適用事例を元に本書を作成しているからである。想定対象読者と各読者向けコンテンツは表 1-1 のとおり。

### ・ 統括責任者【略称：統括】

今後取り組む予定の適用領域や適用工程でどのような技術・手法があるか、およびその適用状況を知ることができる。既に取り組んでいる技法・手法が本書に取り上げられていれば、比較することができる。また、具体的な適用事例から、対象とする組織の適用時の注意点や工夫する点などを知ることができる。

### ・ プロジェクトマネージャ【略称：PM】

統括責任者の項で記載した内容に加え、現状のプロジェクトで抱えている課題について取り上げられている適用事例の適用領域や適用工程で類似のものを見つけ、分析できる。また、まだ適用していない技術・手法の適用事例を精読し、適用の可能性を見つけ出せる。

### ・ プロジェクトリーダー【略称：PL】

プロジェクトマネージャの項で記載した内容に加え、将来、プロジェクトで適用する可能性のある技術・手法の適用事例をプロジェクトメンバーと精読し、適用可能性の議論をしたり、さらに専門書を参考にしたりすることで、人材育成や意識改革のきっかけにしたりすることができる。特に大学や研究機関などの研究員が取り組んでいる技術・手法は、本人の専門性を高めるきっかけになるであろう。

- **担当者（システム開発）【略称：担当】**

本書で取り上げられている技術・手法がどのような領域や工程で利用されているのかを知るきっかけになる。また、日頃の開発の中で課題に感じていることがあれば、自分と同様の開発領域・開発工程を通して課題の共有化を図り、どのような技法・手法を利用して解決をしようとしているかを知ることができる。さらに興味関心がでてきたら、適用事例を精読し自分のプロジェクトと比較することで、新しい視野が生まれる。

- **品質管理部門や技術統括部門【略称：品管 / 技本】**

会社で対象とする領域の範囲で本書が取り上げている事例を精読し、現場が対応している開発領域や抱えている課題を元に解決方法などを助言する。先進的な技術・手法を導入しづらい現場に他社の成功事例と併せてどのような方法で課題を解決すべきかを示し、新しい取組みの道標を提供する。

- **大学や研究機関などの研究員【略称：R&D】**

ソフトウェアの高信頼化をさらに追求するために、適用している技術・手法の適用度合や課題、効果などから、どのようなアプローチを採用すべきか提言できる。また、海外の研究内容を国内で適用しやすいようにアドバイスするなど、さらに先進的なアプローチのきっかけにできる。

表 1-1 本書の想定読者と対象コンテンツ

本書の構成		想定読者					
		統括	PM	PL	担当者	品管 / 技本	R&D
1章		○	○	○	○	○	○
2章		○	○	○	○	○	○
3章	3.1	○	○			○	
	3.2		○	○	○		
	3.3		○	○		○	○
	3.4	○	○			○	
4章			○	○	○	○	
5章						○	○
6章						○	○
7章			○	○			
付録		○	○	○	○	○	○

## 1.5 本書の位置づけ

本書は2013年度から収集した適用事例を元に分析に必要なキーワードを抽出し、ワーキンググループ(WG)で議論した分類軸を元に整理し、分析を実施した成果物である(図1-13)。

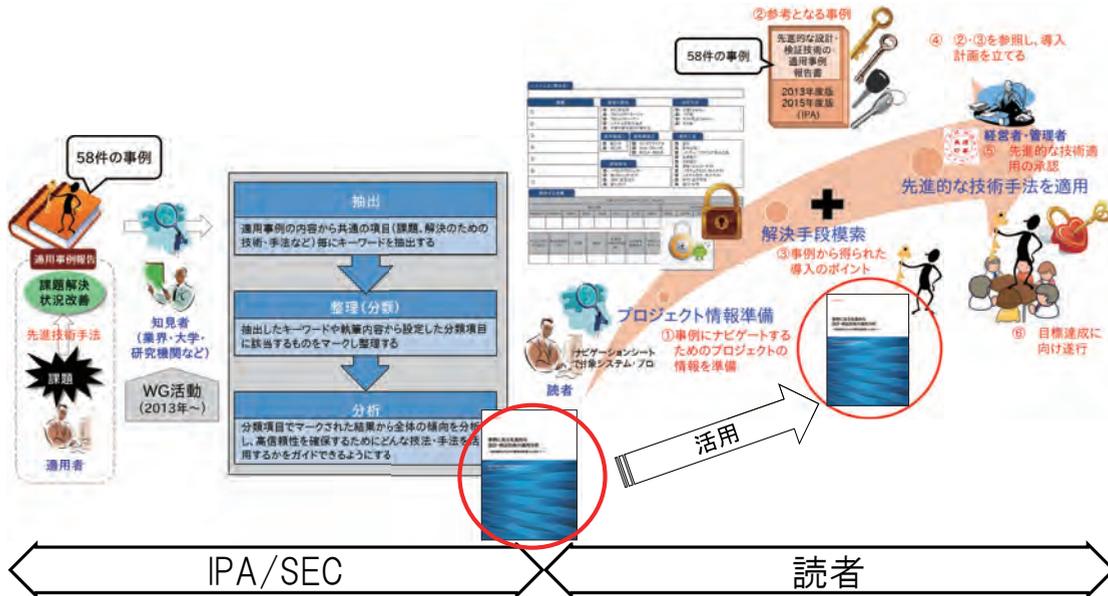


図 1-13 本書の位置づけ

## 1.6 本書の活用方法

本書を活用して、抱えている課題を解決するための先進的な技術・手法を見つけ出し、適用するためのヒントを得る方法について述べる（図 1-14）。その流れは、以下の6つの場面を想定している。①、②、③が本書を活用し参考となる解決手段を探すフェーズであり、④、⑤、⑥は先進的な技術・手法を社内やプロジェクトに実際に適用するフェーズとなる。

- ① 事例にナビゲートするためのプロジェクトの情報を準備する。
- ② ①で準備した情報を元に参考となる事例を見つける。
- ③ 平行して、本書から各技術や手法が効果を発揮するテーマや適用工程を知る（3.2.3.2 章と 3.2.3.3 章に記載）。
- ④ ②と③を参照し、導入のための計画を立てる。
- ⑤ 本書や社内外の基礎データなどから先進的な取組みの重要性の裏付けを用意し、取組みの承認を得る。
- ⑥ 目標達成に向け、先進的な技術や手法を導入したプロジェクトを遂行する。

具体的な本書の活用方法（上記の①、②、③）については、「4 章 課題解決のヒントになる適用事例や適用技術・手法を紐付け」で例を挙げて説明する。

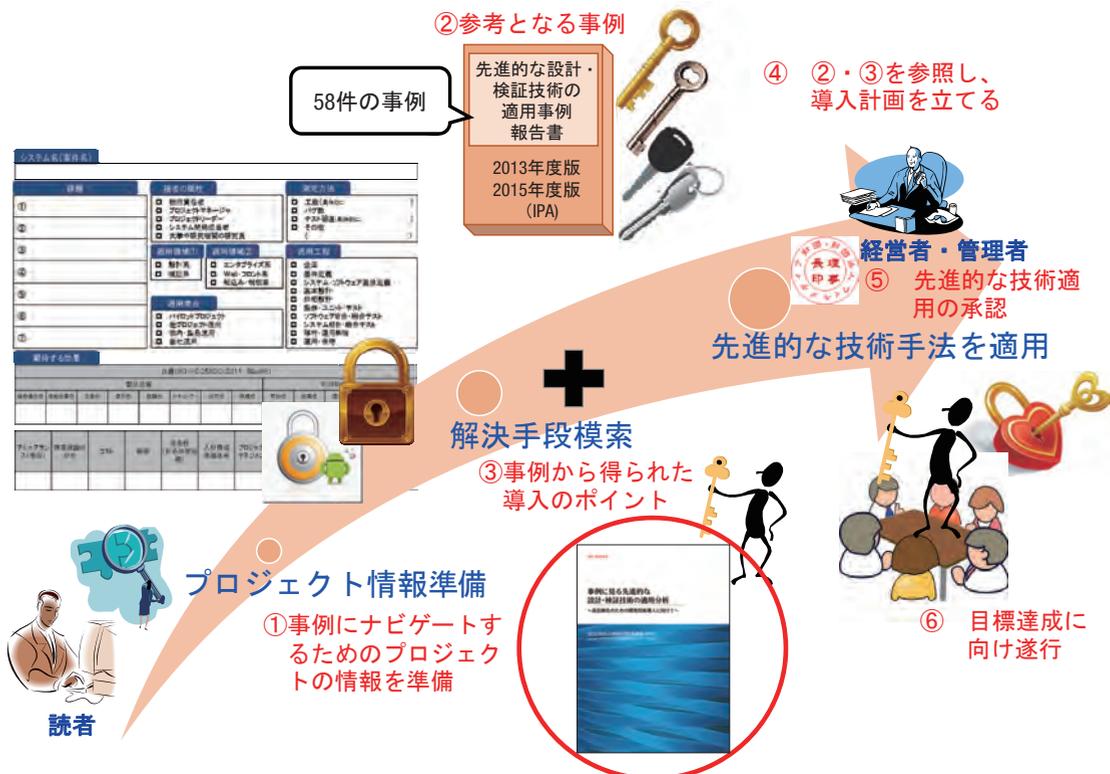


図 1-14 本書の活用方法

## 事例の収集・分類・ 分析・普及の概要 ～取組み紹介～

..... 22

2.1 適用事例収集方針 .....	23
2.1.1 適用事例の内容構成 .....	24
2.1.2 収集する適用事例の主な対象 .....	26
2.1.3 適用事例の収集 .....	27
2.1.4 適用事例収集状況 .....	28
2.1.5 適用事例一覧 .....	30
2.1.5.1 先進的な設計・検証適用事例報告書（2013年度版）...	30
2.1.5.2 先進的な設計・検証適用事例報告書（2015年度版）...	32
2.2 抽出・整理・分析方針 .....	34
2.2.1 共通項目の抽出 .....	36
2.2.2 整理するための分類軸の設定 .....	37
2.2.2.1 適用領域 .....	37
2.2.2.2 適用工程 .....	38
2.2.2.3 技術と手法 .....	39
2.2.2.4 「課題」「効果」「今後の取組」の分類項目 .....	40
2.2.3 整理・分類した結果 .....	49
2.2.4 分析のステップ .....	51

# 2章 事例の収集・分類・分析・普及の概要 ～取組み紹介～

複雑化・高度化する情報処理システムを実現するソフトウェアについて、その高信頼性を確保するためには、従来の開発手法・技術だけでは限界をむかえている。そのため、先進的な手法・技術を積極的に取り入れ、開発の高度化・効率化に取り組む必要がある。開発においては、上流の活動の良し悪しが成果物の良否や生産性に大きく影響する。そこでIPA/SECでは、特に上流設計と検証を中心に、先進的な技術・手法を適用した事例を収集し、適用のためのヒントが得られるよう整理・分類・分析した。

収集した適用事例について、本書に先行して技術・手法ごとに技術・手法のチュートリアルを含む適用事例のセミナーを実施し、普及促進を図っている（図2-1）。

- ① ソフトウェアの高信頼性を確保するために先進的な技術・手法を適用した事例を収集する【適用事例収集】→2.1章を参照
- ② 収集した適応適用事例を整理・分類・分析する【整理・分類・分析】→2.2章を参照
- ③ 本書や適用事例紹介セミナー、関連する技術・手法のセミナーを通して技法・手法を普及促進する【普及促進】

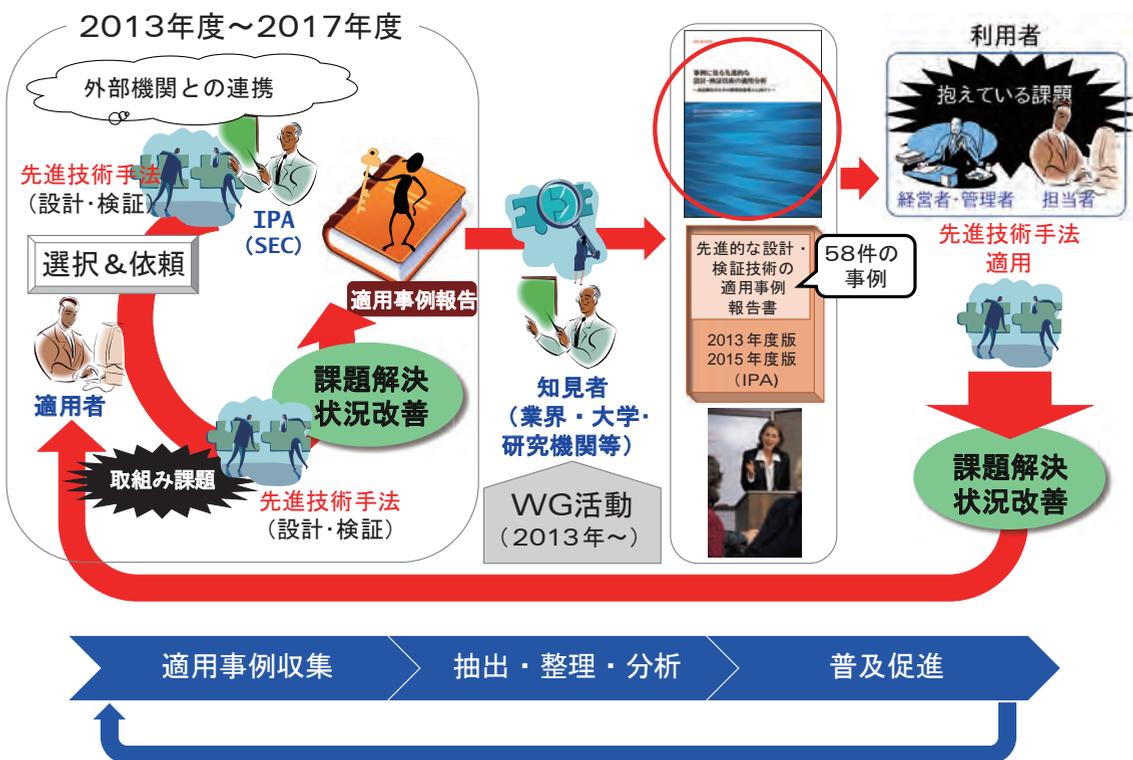


図 2-1 適用事例収集から本書作成と普及活動の流れ

## 2.1 適用事例収集方針

適用事例の収集は、以下の方針で行なった。

- **収集する事例の内容項目の明確化**

技術・手法の解説にならないように、「解決すべき課題」や、達成すべき「目標」を明確にした。

→ 詳細は「2.1.1 章 適用事例の内容構成」を参照

- **収集する適用事例の対象を設定**

社会環境の変化に伴い、システム開発で特に重要な位置づけとなる上流設計と検証に適用したものを対象とした。

→ 詳細は、「2.1.2 章 収集する適用事例の主な対象」を参照

さらに、適用事例をどのように収集したかを「2.1.3 章 適用事例の収集」で、収集状況を「2.1.4 章 適用事例収集状況」で解説する。

## 2.1.1 適用事例の内容構成

適用事例は、基本的に下記の項目を明確にする構成とした。

- ① 『解決すべき課題』
- ② 仮説設定も含めた『目標（課題解決により求める結果：ゴール）』
- ③ 特定技術・手法を使用した具体的な『実施』内容

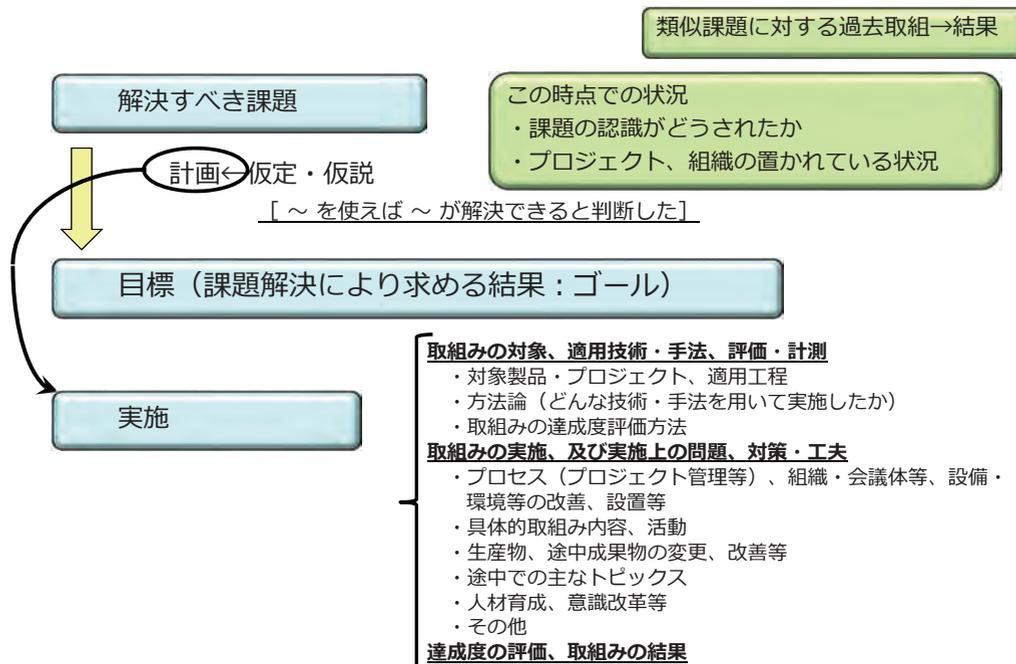


図 2-2 適用事例の内容構成

また、事例の内容が、適用した技術・手法の解説や単なる紹介にならないよう、以下の4つの点を考慮しているか確認して収集した（図 2-3）。

- ① 新たな技術・手法を認知するまでのきっかけや過程
- ② 新たな技術・手法の導入に踏み切る判断をした経緯や判断ポイント
- ③ 新たな技術・手法を導入するための環境（現場状況・既存の開発プロセス・人材育成・他のツールとの関係など）の整備状況
- ④ 実際に技術・手法を適用した結果（効果・残存課題）と今後の取組み予定

本書を手にとった多くの読者は、現状の課題を認識し、改善策に取り組んでいることだろう。しかし、いざ社内でその技術・手法を適用・定着しようとするるとバリア（障壁）が発生したり、限られた期間や予算で効果が出ないと効果を出す前にその適用は途中で終了したりする。このように途中で適用が終了する前に、同様のケースで適用効果が出た先行事例を参考にできれば、適用中に新しい工夫を導入したり、社内のステークホルダに当初計画の見直しを打診したりするなど、効果を出すために色々な方法を考えたり、実施したりできる。できるだけ参考になるアクションをクローズアップするように、適用事例では、各事例の実施者が適用・定着するまでの各節目に取り組んだことを具体的に紹介している。

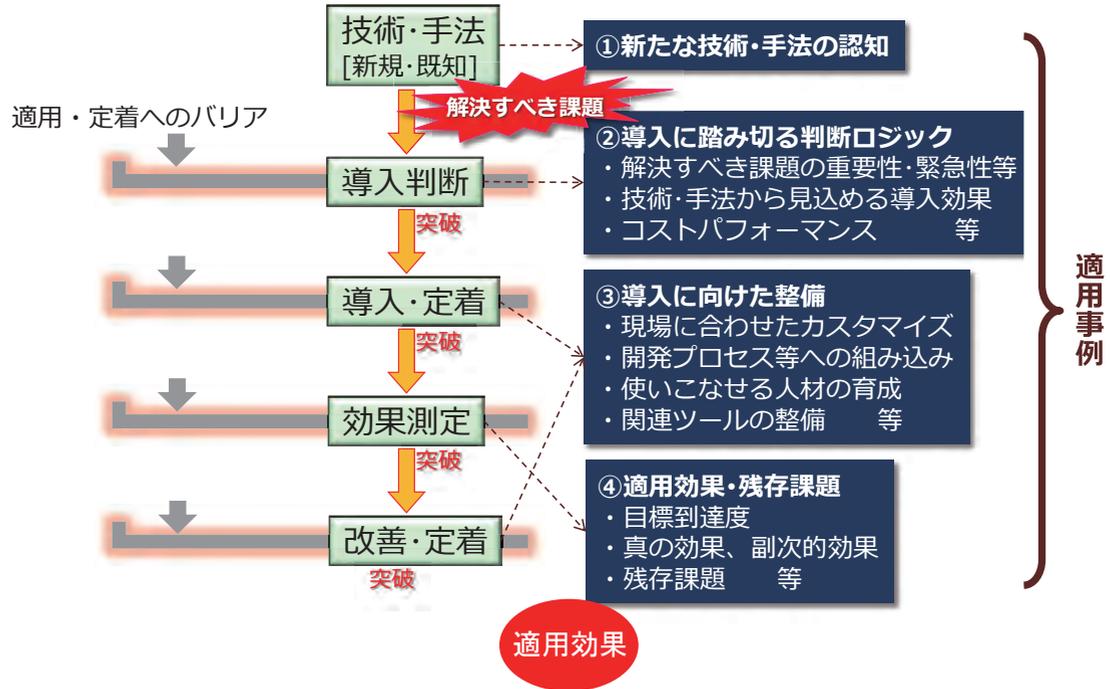


図 2-3 技術・手法の適用・定着までの適用事例の重要性

## 2.1.2 収集する適用事例の主な対象

複雑化・高度化する製品・システムの信頼性を確保するために重要と考えられる、上流設計（システム要求～サブシステム設計）と検証（検証及び妥当性確認）を対象に、先進的な技術や手法を使った効果的な適用事例を収集した。

上流設計（システム要求～サブシステム設計）を収集の対象と位置づけたのは、複雑化する要求に対応するための重要な工程だからだ。「経営環境の変化」、「技術の急激な進歩」、「相互接続・NW（ネットワーク）化」など社会環境の変化をシステムに反映するためには、システム企画を立案したりシステム要件を検査したりする上流工程での取組みが重要である（図2-4）。

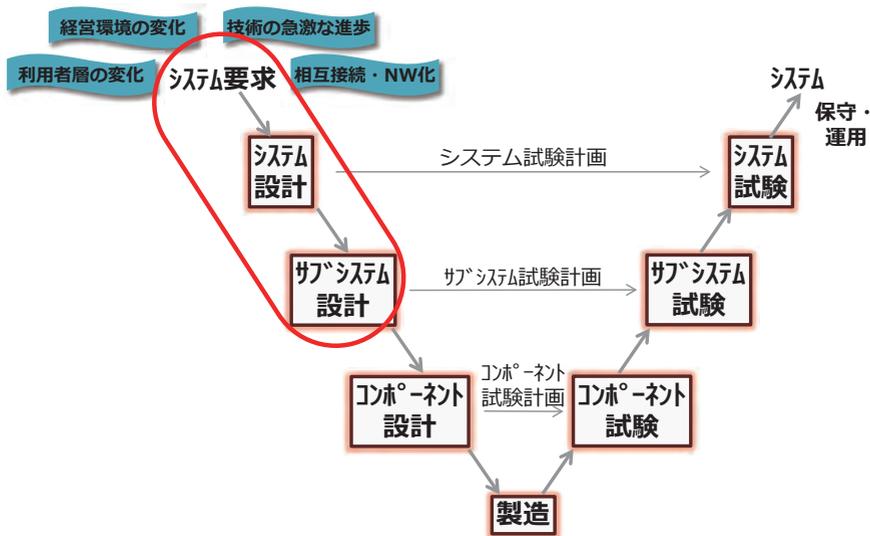


図 2-4 上流工程での先進的な上流設計技術

また、検証を収集の対象と位置づけたのは、上記のような環境変化を上流工程で吸収する取組みの中で、今まで以上に早い段階でのシステム要件やシステム設計の検証、妥当性確認を行なうことが重要であるからだ（図2-5）。

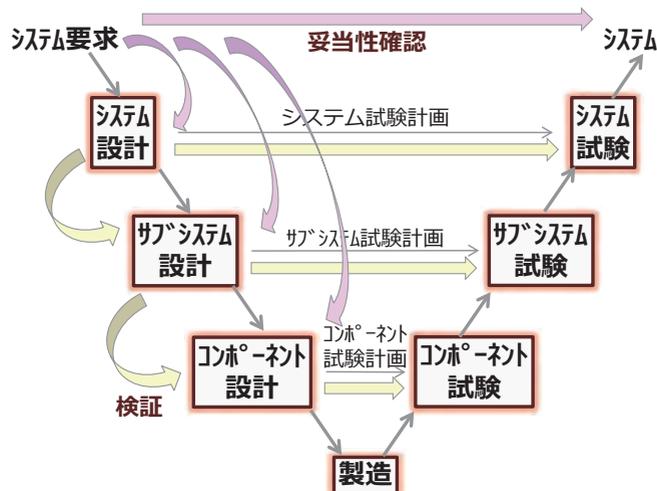


図 2-5 信頼性を確保するための検証技術

### 2.1.3 適用事例の収集

---

適用事例の収集はエンタプライズ系、Web・フロント系、組込み・制御系に大きくカテゴライズしたうえで、有益な情報がバランスよく収集できるよう考慮した。収集候補選定にあたっては、IPA/SECの今までの活動チャンネルを活用することと、新しいチャンネルを発掘することの2面で実施している。

#### IPA/SECの今までの活動チャンネルの活用

- ・ 過去直接的に活動に協力いただいた企業・団体・大学・個人等への協力依頼
- ・ 上記等からの紹介

#### 新規発掘

- ・ 各種文献情報の探索
- ・ 各種展示会、企業・団体セミナー等での情報収集

## 2.1.4 適用事例収集状況

2013年度は24件の適用事例を収集した。各適用領域の対象適用事例数は表2-1に示すとおりである。2013年度に収集できた事例は組込み・制御系が多い。

2014年度については2013年度の収集活動の振り返りおよび収集事例の分析を行い、下記方針を立て実施した。

- ① 組込み・制御系以外の収集数を増やす
- ② 超上流と保守・運用の範囲まで収集する工程を広げる
- ③ 開発者視点だけでなく、利用者（顧客企業）視点での取組みも取り上げる
- ④ 超高速開発などの特定の技術・手法を支援する製品・ツールを用いた取組みも取り上げる

2014年度の収集事例は24件となり、上記収集方針についても一定の成果をみている。

また、2014年度までの事例をもとに、本書記載の整理・分析方法についてワーキンググループでの検討も実施した。

これまでの収集状況の評価および今後の収集方針について明確化し、それらをもとに今後は、以下のような取組み事例も考慮することになった。

- ① IoT<sup>4</sup> や CPS<sup>5</sup> のような新しい環境の中での取組み事例
- ② アシユアランスケースやセキュリティ、リスクアセスメントなどの取組み事例
- ③ 海外での先進的な取組み事例

2015年度前半は、設計系8件、検証系2件の10件を収集した。前年度の適用事例収集の継続で「超上流と保守・運用の範囲まで収集する工程を広げること」と「開発者視点だけでなく、利用者（顧客企業）視点での取組みも取り上げること」について対応できている。

以上の結果、2013年度から2015年度前半現在で合計58件を収集することができた。

<sup>4</sup> IoT (Internet of Things)

<sup>5</sup> CPS (Cyber-Physical Systems)

表 2-1 適用事例収集状況

適用領域 ① <sup>※2</sup>	年度	件数	適用領域 ② <sup>※2</sup>			合計
			エンタプライズ系	Web・フロント系	組込み・制御系	
設計系	2013	13	3	3(1) <sup>※1</sup>	8(1) <sup>※1</sup>	14(1) <sup>※1</sup>
	2014	12	4(2) <sup>※1</sup>	2(1) <sup>※1</sup>	9(1) <sup>※1</sup>	15(3) <sup>※1</sup>
	2015	8	5(1) <sup>※1</sup>	1(1) <sup>※1</sup>	3	9(1) <sup>※1</sup>
	中計	33	12(3) <sup>※1</sup>	6(3) <sup>※1</sup>	20(2) <sup>※1</sup>	38(5) <sup>※1</sup>
検証系	2013	11	1	1	9	11
	2014	12	6(2) <sup>※1</sup>	5(1) <sup>※1</sup>	4(2) <sup>※1</sup>	15(3) <sup>※1</sup>
	2015	2	2	0	0	2
	中計	25	9(2) <sup>※1</sup>	6(1) <sup>※1</sup>	13(2) <sup>※1</sup>	28(3) <sup>※1</sup>
合計		58	21(5) <sup>※1</sup>	12(4) <sup>※1</sup>	33(4) <sup>※1</sup>	66(8) <sup>※1</sup>

※1：適用事例が適用領域②の分類の複数に属しているカッコ内は該当件数  
 ※2：分類軸の定義は、2.2.2章に後述している

適用事例の提供者の立場は、図 2-6 のとおりである。幅広い役職から提供され、特に「統括責任者」と「プロジェクトマネージャ」が多かった。

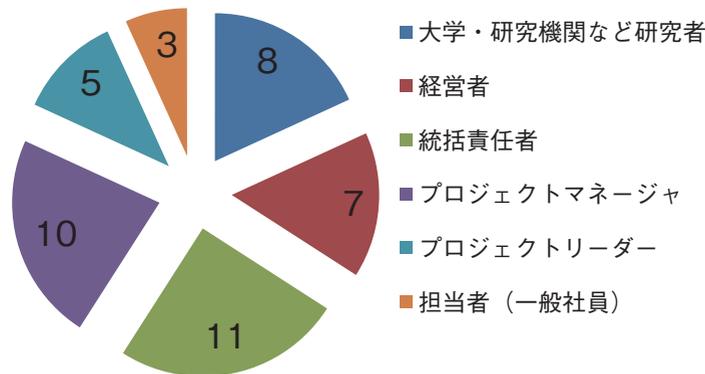


図 2-6 提供者の分布 (全体)

## 2.1.5 適用事例一覧

### 2.1.5.1 先進的な設計・検証適用事例報告書（2013年度版）

2013年度版の報告書に掲載している事例の一覧を表2-2に示す。表中の事例参照番号は「7章 適用事例の概要」の番号と同一である。

表 2-2 2013年度版掲載適用事例一覧

【設計系】

	事例参照番号	標 題	事例提供元 <sup>6</sup>
1	A-1	アシュアランス技術を用いた鉄道信号の革新	東日本旅客鉄道(株)
2	A-2	XDDP 導入による派生開発の品質改善とその効果	(株)日立産業制御ソリューションズ
3	A-3	組込み系の利用品質における「HMI 品質メトリクス」開発と適用事例	(株)U'eyes Design
4	A-4	要件定義段階における信頼性向上の取り組み事例紹介	ビッグローブ(株)
5	A-5	要件定義の品質向上に向けた取り組み	富士通(株)
6	A-6	ジェネレータツールを利用した高信頼開発、高速開発の実践	(株)市進ホールディングス
7	A-7	設計工程における Terasoluna DS の適用	(株)エヌ・ティ・ティ・データ
8	A-8	Grails/Groovy の適用推進	エヌ・ティ・ティ・ソフトウェア(株)
9	A-9	個人依存開発から組織的開発への移行事例～要求モデル定義と開発プロセスの形式化による高生産性/高信頼性化～	三菱電機メカトロニクスソフトウェア(株)
10	A-10	MBSE による双腕作業ロボット動作実行系のコンセプト設計	(独)産業技術総合研究所 (AIST)
11	A-11	仕様記述言語 VDM++ を用いたシステムの仕様の記述	フェリカネットワークス(株)
12	A-12	車載 ECU 開発における上流工程での品質確保	東芝情報システム(株)
13	A-13	独自開発したモデル駆動開発プロダクトラインとエンジニアリングの実践	(株)デンソー

<sup>6</sup> 「先進的な設計・検証技術の適用事例報告書 2013年度版」公開時点（2014年5月30日）での組織名称

【検証系】

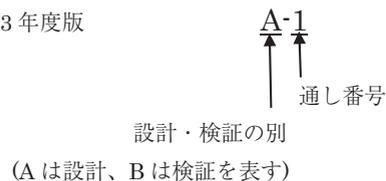
	事例参照番号	標 題	事例提供元 <sup>6</sup>
34	B-1	宇宙システムにおける上流工程仕様の妥当性確認技術	(独)宇宙航空研究開発機構
35	B-2	鉄道の機能安全 (RAMS) 認証支援のためのセーフティケース	(独)産業技術総合研究所 西日本旅客鉄道 (株)
36	B-3	非機能要求グレードの大学ポータルサービスへの適用	名古屋大学
37	B-4	冗長構成システム (クラウド等) の耐故障性に対する検証技術	(株)富士通コンピュータテクノロジー
38	B-5	単体ランダムテスト実行 / 可視化ツール "Jvis" の適用事例	宮崎大学
39	B-6	要求仕様明確化のための仕様記述技術 (USDM) 活用事例	(株)ベリサーブ
40	B-7	形式手法を用いたセキュリティ検証	アーク・システム・ソリューションズ(株)
41	B-8	形式仕様記述手法を用いた高信頼性を達成するテスト手法とその実践	フェリカネットワークス(株)
42	B-9	Orthogonal Defect Classification 分析による欠陥除去と品質の成熟度可視化	オリンパスソフトウェアテクノロジー(株)
43	B-10	モデル検査の適用による上流工程での設計誤りの発見	(株)東芝
44	B-11	上流工程の要求を効率的に閉ループシミュレーションする次世代 SILS の開発	トヨタ自動車(株)

事例参照番号について

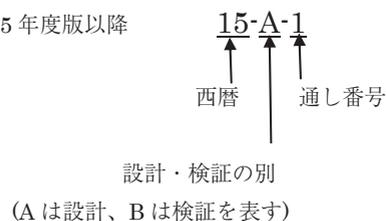
先進的な設計・検証技術適用報告書の各事例ごとに付番した  
 基本構成 : 6 or 7桁、西暦、記号、アルファベット、数字(0~9)  
 ただし、2013年度版掲載分は西暦なしの 3 or 4桁

<例示>

例示 1 : 2013年度版



例示 2 : 2015年度版以降



## 2.1.5.2 先進的な設計・検証適用事例報告書（2015 年度版）

2015 年度版の報告書に掲載している適用事例の一覧を表 2-3 に示す。

表 2-3 2015 年度版掲載適用事例一覧

【設計系】

	事例参照番号	標 題	事例提供元 <sup>7</sup>
14	15-A-1	業務生産性向上や市場環境の変化に対応できる経営・業務を実現するために活用される BPM の紹介	(一社) コラボネット事業推進協会 / (株) BPM 実践企画
15	15-A-2	BPM をベースにした会社統合での新業務プロセス設計の適用	三菱商事 RtM ジャパン(株)
16	15-A-3	人間系プロセスを含む業務を BPM ソフトウェア活用により改善した事例	日本電気(株)
17	15-A-4	受注業務にビジネスアナリシス方法論を適用した業務システムの構築 ～中小企業の特注品業務プロセス改革の提案～	(株) プロセスデザインエンジニアリング
18	15-A-5	ビジネスへの貢献が求められる時代のソフトウェア開発の考え方 ～超高速開発ツールがもたらす方法論のイノベーション～	(一社) ICT 経営パートナーズ協会 / MBC (Method Based Consulting)
19	15-A-6	「フィーチャー」の概念を取り入れたモデルベース開発	三菱スペース・ソフトウェア(株)
20	15-A-7	D-Case 導入によるシミュレーション S / W の期待結果明確化と合意形成	三菱電機(株)
21	15-A-8	大規模システム開発プロジェクトにおけるユーザーエクスペリエンス品質設計プロセスの適用とその効果	NEC ソリューションイノベータ(株)
22	15-A-9	ソニーの電子お薬手帳システム「harmo」に適用したセキュリティ設計分析手法	ソニーデジタルネットワークアプリケーションズ(株)
23	15-A-10	XDDP におけるデグレード防止効果を高めるための手法 ～『気づきナビ』の考案～	アズビル(株) (株) インテック キヤノン IT ソリューションズ(株)
24	15-A-11	モデルベース開発への移行に向けた C 言語ソースコードに対する状態遷移抽出技術の適用	(株) 東芝
25	15-A-12	組込システムのモデルベース開発適用における DI コンテナの活用	ヤマハ(株)
26	15-A-13	大規模システムへのモデルベース開発手法の適用	(株) IHI エアロスペース
27	15-A-14	自動車のパワーバックドアシステム開発のためのモデルベースシステムズエンジニアリングの適用	慶應義塾大学大学院 日産自動車(株)
28	15-A-15	製品開発における SysML 適用の取り組み ～要求の可視化～	(株) リコー
29	15-A-16	D-Case を用いたゴール共有による開発プロセスの適用 ～ ET ロボコンでの試行と成果～	富士ゼロックス(株)

<sup>7</sup> 「先進的な設計検証技術の適用事例報告書 2015 年度版」公開時点（2015 年 11 月 18 日）での組織名称

	事例参照番号	標 題	事例提供元 <sup>7</sup>
30	15-A-17	システム記述言語 (AADL) による複合システム設計 ～航空機前方車輪の回転数から速度を計測・記録・表示するシステムへの適用～	ISAE (Institut Supérieur de l'Aéronautique et de l'Espace)
31	15-A-18	ロケットエンジンにおけるモデルベース信頼性評価技術の構築と試行	国立研究開発法人 宇宙航空研究開発機構
32	15-A-19	デジタル制御電源製品開発に対するモデルベース開発の適用	(株)富士通研究所
33	15-A-20	組込みソフトウェアのアーキテクチャ設計方法の可視化	ピースラッシュ(株)

【検証系】

	事例参照番号	標 題	事例提供元 <sup>7</sup>
45	15-B-1	SysML と CML によるシステムオブシステムの検証	Newcastle University
46	15-B-2	通信制御ソフトウェア開発における状態遷移設計の品質向上への取組み ～状態遷移表へのモデル検査の適用～	富士通(株)
47	15-B-3	国際スタンダード認証に求められる「要件から検証結果までのトレーサビリティ管理」の効率化の取組み	富士設備工業(株)
48	15-B-4	Friendly による内部 API を使ったシステムテスト自動化	(株)Codeer
49	15-B-5	アジャイルプロセスにおける実践的な品質向上施策の適用事例	(株)日立ソリューションズ
50	15-B-6	「コストモデル」を使った開発品質・生産性向上の取組み ～バグ対応コストの見える化と最適化～	(株)HS 情報システムズ
51	15-B-7	メトリクス分析手法を用いた試験品質向上の取組み	(株)東芝
52	15-B-8	ユーザーエクスペリエンスを業務に定着化させるための取り組み事例の紹介	(株)ベリサーブ
53	15-B-9	パッケージソフトウェア開発プロセス改善による品質向上と生産性向上	(株)富士通マーケティング
54	15-B-10	Web システムにおける単体テストの品質向上の取組み	住友電工情報システム(株)
55	15-B-11	安心なサービスの品質改善を実現する為の継続的システムテスト	楽天(株)
56	15-B-12	セキュア開発手法の考察と診断ツールの活用事例の紹介 ～お客様に「安心してご利用ください」と言えるための脆弱性対策～	ビッグロブ(株)
57	15-B-13	モデル検査とテストによる車載オペレーティングシステムの検証	北陸先端科学技術大学院大学
58	15-B-14	モデルベース開発とコード解析を用いた組込みソフトウェアの開発	アルプス電気(株)

## 2.2 抽出・整理・分析方針

収集した適用事例を元に「抽出」「整理(分類)」「分析」の3ステップを実施した(図2-7)。この3つのステップでは、本書の目的である『58件の事例を分析して得られた知見を解説すること』を達成するために、定性的な情報を整理(分類)する中で、共通の軸(分類軸)を使って定量的な情報にすることで、分析をしやすいようにした。また、もうひとつの目的である『解決したい課題とそのヒントになる適用事例や適用技術・手法を紐付けること』を達成するために、全ての適用事例を共通の軸で分類し、読者が抱えるシステム・ソフトウェア開発の課題と類似の適用事例をリンクできるようにした。

それぞれの概要を図2-7に、具体的なやり方を図2-8の例に示す。

### ① 抽出

適用事例の内容から共通の項目(課題、解決のための技術・手法、今後の取組みなど)ごとにキーワードを抽出

### ② 整理(分類)

収集したキーワードをグルーピングし、分類項目を設定。さらに、適用事例ごとに設定した分類項目に該当するものをマークして整理

### ③ 分析

分類項目でマークされた結果から全体の傾向などを分析し、高信頼性を確保するためにどんな技法・手法を活用しているかを明示

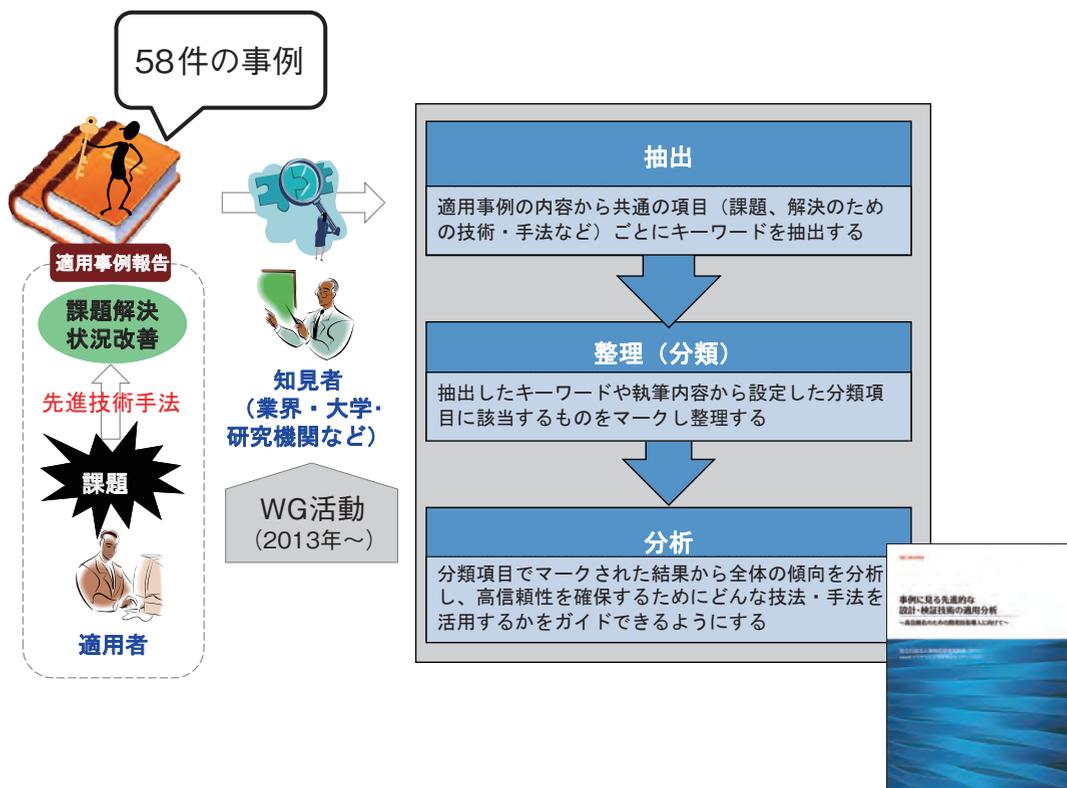


図2-7 WGの知見を元に抽出・整理(分類)・分析

15-A-8  
大規模システム開発プロジェクトにおける  
ユーザーエクスペリエンス品質設計プロセスの  
適用とその効果<sup>1)</sup>

1. 背景  
本報告は、大規模システム開発プロジェクト「E」(Eは「Enterprise」)の事例であり、システム開発の成功は、顧客の満足度の向上に大きく依存している。顧客の満足度の向上は、システム開発の成功に大きく影響している。顧客の満足度の向上は、システム開発の成功に大きく影響している。顧客の満足度の向上は、システム開発の成功に大きく影響している。

2. 背景および課題  
2.1. プロジェクト概要  
2.2. 課題

抽出

団体: TECソリューションパートナー株式会社

標題: 大規模システム開発プロジェクトにおけるユーザーエクスペリエンス品質設計プロセスの適用とその効果

ガイドタイトル(取組み内容を経営者に分かる表現): 「E」の属性である性格や経験の差、組織文化などの主観的な要素が要因で発生する「使いやすさやわかりやすい」操作を企画・設計時に明確にし、遅延や手戻りを削減

技術と手法					定量的な効果	
UX(User Experience)設計	顧客体験型設計	UX共有	ペルソナ	ストーリーボード	カスタマージャーニーマップ・構造シナリオ法	画面・機能上の問題を約40%抑えられる

課題

「E」の属性で発生する遅延や手戻りなどの問題	顧客ニーズの理解・抽出不足	情報共有の不確かさ	大規模プロジェクトでは、開発に際するステークホルダーが増え、その分顧客の要求が多様化	既存のUX品質設計手法(ペルソナ・ストーリーボード)の適用が難しい	顧客側における「E」の属性を踏まえた設計が難しい	手法自体が顧客側や開発現場の経験が乏しいため、受け止められない	顧客側も開発現場も、工数や期間が増える
------------------------	---------------	-----------	--	-----------------------------------	--------------------------	---------------------------------	---------------------

測定方法

標準時・ドキュメントを精査し問題点を抽出	画面デザイン標準の遵守状況を数値化(91%の画面が非標準)	顧客体験型設計の遵守状況を把握し、業務内容や課題を明らかにして対応する工数と効率化
----------------------	-------------------------------	---

今後の課題

開発現場での「E」の属性を踏まえた設計が難しい	「E」の属性を踏まえた設計が難しい	「E」の属性を踏まえた設計が難しい
-------------------------	-------------------	-------------------

適用度合

200大規模プロジェクト(2年以上継続し、開発費が1000万円以上)

定性的な効果

待ち時間短縮	より少ない工数で開発	業務機能設計やサービス設計などの他のプロセスの検討を早期発見	開発や設計で、作業者が多数入れ替わる中、顧客要件を維持するの効果がより効果が増加	信頼立って運用すると定期的な連携確認の効果	マネージャーがメンバーに指示すると効果	顧客体験を定量的に意識させると効果
--------	------------	--------------------------------	--	-----------------------	---------------------	-------------------

ある事例

事例から課題などを抽出した結果

ワーキンググループで  
確定した分類軸

整理

事例の整理の結果をテーマ毎に右図の5つのマス目と5つの記号で明記  
(自動集計をするためにこのような構成を設定)

- 課題設定した場合 : 1段目に□(白抜き)の四角を入力
- 課題未設定した場合 : 2段目に△(下向き)の三角を入力
- 効果確認した場合 : 3段目に■(黒塗り)の四角を入力
- 効果未確認した場合 : 4段目に△(下向き)の三角を入力
- 新規の課題設定の場合 : 5段目に△(白抜き)の三角を入力(△は今後の取り組みを示す)

事例番号	システム及びソフトウェア品質 モデル (JIS X 25010 : 2013)														アシユアランス保証	障害原因の分析	コスト	納期	生産性(対応期間)	人材育成(意識改革)	プロジェクトマネジメント	見積支援	普及促進	体制強化(構築)	グローバル展開
	製品品質					利用時の品質					リスク回避	利用状況													
	機能適合性	性能効率性	互換性	使用性	信頼性	セキュリティ	保守性	移植性	有効性	効率性			満足性												
21	15-A-8																								

抽出した結果を元に整理した結果

分析

図 2-8 抽出・整理・分析の例



## 2.2.1 共通項目の抽出

---

収集した適用事例から、以下の共通項目について抽出・整理した。

- ① 技術と手法  
適用事例中に表現されている『技術や手法』を抽出
- ② 課題  
適用事例中に取り上げられている『課題』を抽出
- ③ 定量的な効果  
適用事例中に表現されている先進的な技術・手法を適用した『効果』で定量的な内容を抽出
- ④ 定性的な効果  
定量的な効果以外の定性的な効果を事例内容の表現で抽出
- ⑤ 測定方法  
適用事例中に今回適用した技術や手法の効果を評価するために測定した方法（メトリクス）を抽出
- ⑥ 今後の課題  
技術や手法などの適用結果を検証した時点以降に発生した課題を抽出
- ⑦ 適用度合  
技術・手法を適用したプロジェクト数や製品開発への適用状態を抽出

## 2.2.2 整理するための分類軸の設定

前述の項目で各適用事例の内容を抽出すると「技術と手法」や「課題」、「効果（定性的なもの）」の中で共通の項目が見出された。各適用事例の中身を共通の項目で整理すれば、読者の関心のある適用事例を容易に見つけ出せる。そこで各適用事例を、以下のような項目で分類した。

- 2.2.2.1 適用領域
- 2.2.2.2 適用工程
- 2.2.2.3 技術と手法
- 2.2.2.4 「課題」「効果」「今後の取組」の分類項目

### 2.2.2.1 適用領域

適用事例の対象の領域を以下の2つに分類した。適用ドメインでの分類は、従来はエンタプライズ系と組込み・制御系の2つに分類していた。しかし、システムやソフトウェア、製品を従来の環境以外（多様化した社会インフラや、その複合的な利用など）で適用する場合もあったので、新たに Web・フロント系を定義した。

#### ① 設計系と検証系（以降、適用領域①）

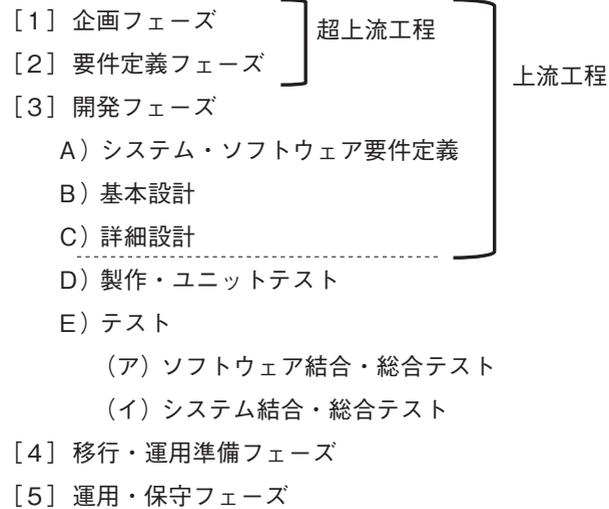
適用事例の対象領域のメインが、V字の開発プロセスの左側であれば設計系、右側であれば検証系とした。事例別で考えると、設計系の取組みと検証系の取組みが混在しているケースがあるが、メインの取組みをしている方にその事例进行分类している。

#### ② 適用ドメイン（以降、適用領域②）

- ・エンタプライズ系
  - 一般的に情報システム部門が対応している基幹・業務アプリケーション系のシステムやソフトウェア群
- ・Web・フロント系
  - モバイル・タブレット端末などを使ってサービスを利用するシステムやソフトウェア群（例えば、事業ラインの対顧客とのやり取りで使われるシステムやソフトウェアやコンシューマ向けのサービス）
- ・組込み・制御系
  - ハードウェア製品に搭載されるシステムやソフトウェア群

### 2.2.2.2 適用工程

適用事例で適用されている開発工程を、以下のように分類した。この分類には、JIS X 0160:2012 (ISO/IEC 12207:2008) に基づく SEC BOOKS「共通フレーム 2013」に掲載している「図 4-25 プロセスの時間軸へのマッピング例<sup>8</sup>」を適用し、作成した。



<sup>8</sup> 共通フレーム 2013：IPA/SEC 巻末折込み参照

### 2.2.2.3 技術と手法

ソフトウェア高信頼化を実現するには、様々な技術と手法がある。しかし、現在、開発技法や開発手法を全体的に体系化したものはない。そこで、今までに収集した 58 件の適用事例で使われている技術・手法を列挙したところ、以下に示す 28 個の技術・手法が存在した。

- ① アジャイルソフトウェア開発モデル
- ② ソフトウェアプロトタイプ開発モデル
- ③ イテレーション開発モデル【反復型開発モデル】
- ④ W（開発）モデル
- ⑤ BPM【ビジネスプロセスマネジメント】
- ⑥ MBSE【モデルベースシステムズエンジニアリング】
- ⑦ 形式仕様記述
- ⑧ HCD【人間中心設計】
- ⑨ UX【ユーザーエクスペリエンス】
- ⑩ MBD【モデルベース開発】
- ⑪ MDD【モデル駆動開発】
- ⑫ 保守・更改・派生開発（XDDP）
- ⑬ MILS/HILS/SILS
- ⑭ RAD【高速アプリケーション開発】
- ⑮ CASE
- ⑯ IDE【統合開発環境】
- ⑰ SPLE【ソフトウェアプロダクトライン開発】
- ⑱ ソフトウェアフレームワーク
- ⑲ テスト自動化
- ⑳ バグ分析（ODC 分析）
- ㉑ 形式検証
- ㉒ モデル検査
- ㉓ IV&V
- ㉔ 耐故障性検証（運用時）
- ㉕ アシユアランスケース（D-Case）
- ㉖ セキュリティ
- ㉗ リスクアセスメント
- ㉘ 独自開発・検証ツール

#### 2.2.2.4 「課題」「効果」「今後の取組」の分類項目

一般的に、ソフトウェア高信頼化に影響を及ぼすのは、QCDM (Quality 品質、Cost コスト、Delivery 納期、Management 管理) が主な要因と言われている。2013年度の事例内容を整理したところ、QCDM以外の課題も浮き彫りになった。適用事例から以下の項目を分類項目とした。

- [1] 品質：JIS X 25010：2013 (ISO/IEC 25010：2011)「システム及びソフトウェア製品の品質要求及び評価 (SQuaRE) - システム及びソフトウェア品質モデル」
  - (ア) 製品品質モデル：8 特性
  - (イ) 利用時の品質モデル：5 特性
- [2] アシユアランス (保証) ケース
- [3] 障害 (バグ) 原因の分析
- [4] コスト
- [5] 納期
- [6] 生産性 (対応時間短縮)
- [7] 人材育成・意識改革
- [8] プロジェクトマネジメント
- [9] 見積支援
- [10] 普及促進
- [11] 体制強化・再構築
- [12] グローバル展開

収集した適用事例から、以下の5つの記号を使って該当する項目 (テーマ) にマークして整理した。使用する記号の意味は、以下のとおりである。

- (白抜きの四角)：『課題設定』したもの (空欄は『課題設定』していないものを意味する)
- (黒塗りの四角)：『効果発揮』したもの (空欄は『効果発揮』していないものを意味する)
- (白抜きの三角)：『今後の取組課題設定』したもの (空欄は『今後の取組課題設定』していないものを意味する)
- ↓ (下矢印)：課題に対して取組みを実施したもの (空欄は取組みをしなかったものを意味する)
- ⇓ (下二本矢印)：今後の取組課題を設定した (△) に対して、取組みを実施する予定のもの (空欄は今後の取組み予定のないものを意味する)

この記号を各適用事例に当てはめて分析をした結果、以下のような4つのパターンに分類された。

- 課題克服**：課題設定した項目（例：品質、コストなど）を解決した（解決途中のものも含む）。
- 副次効果**：当初の課題として認識していなかった項目（例：品質、コストなど）で効果を発揮した。  
例えば、当初「機能適合性」を課題設定せず、先進的な技術・手法を適用したところ、「機能適合性」で効果を発揮した場合は、品質に関する副次効果である。
- さらなる取組**：当初の課題項目と副次効果発揮項目に対して継続的な取り組みをすること。当初の課題に取り組んでいる場合、または、当初から課題を設定していないが効果を発揮した場合が対象となる。  
例えば、「機能適合性」を課題として取り組み、その後に再び「機能適合性」でさらに取り組むを行なう場合は、「機能適合性」に関するさらなる取り組みである。
- 新規取組**：当初の課題が設定されない場合、かつ、効果も発揮していない場合で今後の取組みが発生すること  
例えば、当初「機能適合性」を課題として設定せず、かつ「機能適合性」の効果も発揮しない状態で、新しく「機能適合性」に対し取り組む場合は、新規取組である。

実際に各適用事例にマークした結果の一例を図2-9に示す。事例全体についてマークした表は、本書巻末に付録1として掲載している。また、電子データをIPAウェブサイトから提供する。

事例参照番号	システム及びソフトウェア品質 モデル (JIS X 25010 : 2013)												アシュアランス保証	改善原因の分析	コスト	納期	生産性対応時間短縮	人材育成意識改革	プロジェクトマネジメント	見積支援	普及促進	体制強化(再構築)	グローバル展開
	機能適合性	性能効率性	互換性	使用性	信頼性	セキュリティ	保守性	移植性	有効性	効率性	満足性	リスク回避性											
1	A-1	■	■		■	■	■		■	■	■		■	□									
2	A-2	□	□					□	□											■	■	■	
3	A-3																						
4	A-4	□	□		□															■	■	■	
5	A-5	□	□																				

図 2-9 「課題」、「効果」、「今後の課題」など、マーキングイメージ

上記のマーキングから適用事例ごとに、図2-10のように項目（テーマ）ごとに適用結果が見えてくる。その結果、4つのパターン（適用結果）に分類された。

適用結果	課題克服		副次効果		さらなる取組			新規取組
	□	□			□	□		
記号の組み合わせパターン	□	□	■	■	□	□		
	↓	↓			↓			
	■	■			■		■	
		⇓			⇓		⇓	
		△		△	△	△	△	△

図 2-10 適用結果の状態

## (1) 品質

品質は、ISO/IEC 25000 シリーズ「システム及びソフトウェア製品の品質要求及び評価 (SQuARE)」の JIS X 25010 : 2013 (ISO/IEC 25010 : 2011)「システム及びソフトウェア品質モデル」を元に細分化した。品質モデルは、品質特性と品質副特性で構成されている。JIS X 25010 : 2013 では、以下のような2つの品質モデルを定めている。

- (ア) 製品品質モデル：8 特性
- (イ) 利用時の品質モデル：5 特性

それぞれの品質特性は以下のとおり。

### (ア) 製品品質モデル：8 特性

このモデルは、ソフトウェアの静的特徴及びコンピュータシステムの動的特徴に関係する、8つの特性(これらは、さらに副特性に分けられる)からなる(図2-11)。このモデルは、コンピュータシステムとソフトウェア製品との両者に適用できる。

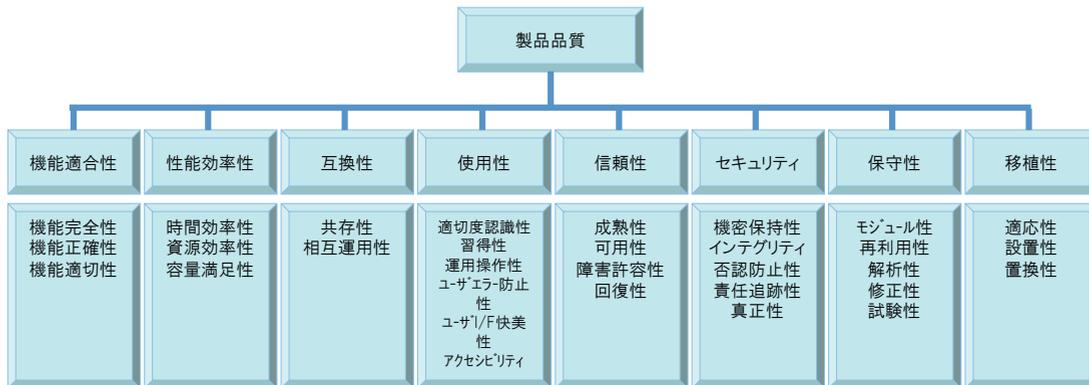


図 2-11 製品品質モデルを構成する特性及び副特性 (JIS X 25010 : 2013)

#### ① 機能適合性 (functional suitability)

明示された状況下で使用するとき、明示的ニーズ及び暗黙のニーズを満足させる機能を、製品又はシステムが提供する度合い。副特性として次がある。

- 機能完全性 (functional completeness)  
機能の集合が明示された作業及び利用者の目的の全てを網羅する度合い。
- 機能正確性 (functional correctness)  
正確さの必要な程度での正しい結果を、製品又はシステムが提供する度合い。
- 機能適切性 (functional appropriateness)  
明示された作業及び目的の達成を、機能が促進する度合い。

#### ② 性能効率性 (performance efficiency)

明記された状態(条件)で使用する資源の量に関係する性能の度合い。副特性として次がある。

- 時間効率性 (time behaviour)  
製品又はシステムの機能を実行するとき、製品又はシステムの応答時間及び処理時間、並びにスループット速度が要求事項を満足する度合い。
- 資源効率性 (resource utilization)  
製品又はシステムの機能を実行するとき、製品又はシステムで使用される資源の量及び種類が要求事項を満足する度合い。

- 容量満足性 (capacity)  
製品又はシステムのパラメータの最大限度が要求事項を満足させる度合い。

### ③ 互換性 (compatibility)

同じハードウェア環境又はソフトウェア環境を共有する間、製品、システム又は構成要素が他の製品、システム又は構成要素の情報を交換することができる度合い、及び／又はその要求された機能を実行することができる度合い。副特性として次がある。

- 共存性 (co-existence)  
その他の製品に有害な影響を与えずに、他の製品と共通の環境及び資源を共有する間、製品が要求された機能を効率的に実行することができる度合い。
- 相互運用性 (interoperability)  
二つ以上のシステム、製品又は構成要素が情報を交換し、既に交換された情報を使用することができる度合い。

### ④ 使用性 (usability)

明示された利用状況において、有効性、効率性及び満足性をもって明示された目標を達成するために、明示された利用者が製品又はシステムを利用することができる度合い。副特性として次がある。

- 適切度認識性 (appropriateness recognizability)  
製品又はシステムが利用者のニーズに適切であるかどうかを利用者が認識できる度合い。
- 習得性 (learnability)  
明示された利用状況において、この場合の学習目標とは、有効性、効率性、リスク回避及び満足性を持って製品、又はシステムを使用できることを指す。
- 運用性 (operability)  
製品又はシステムが、それらを運用操作しやすく、制御しやすくする属性をもっている度合い。
- ユーザエラー防止性 (user error protection)  
利用者が間違いを起こすことをシステムが防止する度合い。
- ユーザインタフェース快美性 (user interface aesthetics)  
ユーザインタフェースが、利用者にとって楽しく、満足のいく対話を可能にする度合い。
- アクセシビリティ (accessibility)  
製品又はシステムが、明示された利用状況において、明示された目標を達成するために、幅広い範囲の心身特性及び能力の人々によって使用できる度合い。

### ⑤ 信頼性 (reliability)

明示された時間帯で、明示された条件下に、システム、製品又は構成要素が明示された機能を実行する度合い。副特性として次がある。

- 成熟性 (maturity)  
通常の運用操作の下で、システム、製品又は構成要素が信頼性に対するニーズに合致している度合い。
- 可用性 (availability)  
使用することを要求されたとき、システム、製品又は構成要素が運用操作可能及びアクセス可能な度合い。
- 障害許容性 (耐故障性) (fault tolerance)  
ハードウェア又はソフトウェア障害にもかかわらず、システム、製品又は構成要素が意図したように運用操作できる度合い。
- 回復性 (recoverability)  
中断時又は故障時に、製品又はシステムが直接的に影響を受けたデータを回復し、システムを希望する状態に復元することができる度合い。

## ⑥ セキュリティ (security)

人間又は他の製品もしくはシステムが、認められた権限の種類及び水準に応じたデータアクセスの度合いをもてるように、製品又はシステムが情報及びデータを保護する度合い。副特性として次がある。

- 機密性 (confidentiality)  
製品又はシステムが、アクセスすることを認められたデータだけにアクセスすることができることを確実にする度合い。
- インテグリティ (integrity)  
コンピュータプログラム又はデータに権限をもたないでアクセスすること又は修正することを、システム、製品又は構成要素が防止する度合い。
- 否認防止性 (non-repudiation)  
事象又は行為が後になって否認されることがないように、行為又は事象が引き起こされたことを証明することができる度合い。
- 責任追跡性 (accountability)  
実体の行為がその実体に一意的に追跡可能である度合い。
- 真正性 (authenticity)  
ある主体又は資源の同一性が主張したとおりであることを証明できる度合い。

## ⑦ 保守性 (maintainability)

意図した保守者によって、製品又はシステムが修正できる有効性及び効率性の度合い。副特性として次がある。

- モジュール性 (modularity)  
一つの構成要素に対する変更が他の構成要素に与える影響が最小になるように、システム又はコンピュータプログラムが別々の構成要素から構成されている度合い。
- 再利用性 (reusability)  
一つ以上のシステムに、又は他の資産作りに、資産を使用することができる度合い。
- 解析性 (analyzability)  
製品もしくはシステムの一つ以上の部分への意図した変更が、製品もしくはシステムに与える影響を総合評価すること。欠陥もしくは故障原因の診断、または修正しなければならない部分を識別することが可能であることについての有効性及び効率性の度合い。
- 修正性 (modifiability)  
欠陥の取込みも既存の製品品質の低下もなく、有効的に、かつ、効率的に製品又はシステムを修正することができる度合い。
- 試験性 (testability)  
システム、製品又は構成要素について試験基準を確立することができ、その基準が満たされているかどうかを決定するために試験を実行することができる有効性及び効率性の度合い。

## ⑧ 移植性 (portability)

一つのハードウェア、ソフトウェア又は他の運用環境もしくは利用環境からその他の環境に、システム、製品又は構成要素を移すことができる有効性及び効率性の度合い。副特性として次がある。

- 順応性 (adaptability)  
異なる又は進化していくハードウェア、ソフトウェア又は他の運用環境もしくは利用環境に、製品又はシステムが適応できる有効性及び効率性の度合い。
- 設置性 (install ability)  
明示された環境において、製品又はシステムをうまく設置及び／又は削除できる有効性及び効率性の度合い。
- 置換性 (replaceability)  
同じ環境において、製品が同じ目的の別の明示された製品と置き換えることができる度合い。

(イ) 利用時の品質モデル：5 特性

このモデルは、特定の使用状況において製品が使用されるときに、対話の成果に関係する5つの特性からなる。このシステムモデルは、使用中のコンピュータシステムとソフトウェア製品との両者を含む完備した人間-コンピュータシステムに適用できる。なお、製品品質モデルの適用範囲は、ソフトウェア及びコンピュータシステムを意図しているが、特性の多くは、より広範囲のシステム及びサービスにも関連する。

標準的な利用時の品質モデルは、JIS X 0129-1:2003 (ISO/IEC 9126-1:2001) で規定されたが、その後、ISO/IEC 25010 に改訂された。ISO/IEC 25010 で標準に規定された利用時の品質モデルを図 2-12 に示す。この品質モデルを参考に、重要な品質特性を選定して対象システム・ソフトウェア製品の利用時の品質モデルを定め、具体的な品質要求を検討することができる。

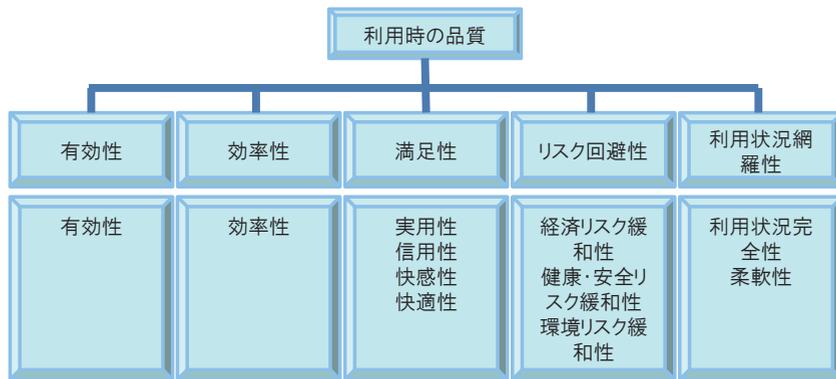


図 2-12 利用時の品質モデルを構成する特性及び副特性 (JIS X 25010 : 2013)

① 有効性 (effectiveness)

明示された目標を利用者が達成する上での正確さ及び完全さの度合い。

② 効率性 (efficiency)

利用者が特定の目標を達成するための正確さ及び完全さに関連して、使用した資源の度合い。

③ 満足性 (satisfaction)

製品又はシステムが明示された利用状況において使用されるとき、利用者ニーズが満足される度合い。副特性として次がある。

● 実用性 (usefulness)

利用の結果及び利用の影響を含め、利用者が把握した目標の達成状況によって得られる利用者の満足の度合い。

● 信用性 (trust)

利用者又は他の利害関係者がもつ、製品又はシステムが意図したとおりに動作するという確信の度合い。

● 快感性 (pleasure)

個人的なニーズを満たすことから利用者が感じる喜びの度合い。

● 快適性 (comfort)

利用者が (システム又はソフトウェアを利用する時の) 快適さに満足する度合い。

#### ④ リスク回避性 (freedom from risk)

製品又はシステムが、経済状況、人間の生活又は環境に対する潜在的なリスクを緩和する度合い。副特性として次がある。

- 経済リスク緩和性 (economic risk mitigation)

意図した利用状況において、財政状況、効率的運用操作、商業資産、評判又は他の資源に対する潜在的なリスクを、製品又はシステムが緩和する度合い。

- 健康・安全リスク緩和性 (health and safety risk mitigation)

意図した利用状況において、製品又はシステムが人々に対する潜在的なリスクを緩和する度合い。

- 環境リスク緩和性 (environmental risk mitigation)

意図した利用状況において、環境に対する潜在的なリスクを製品又はシステムが軽減する度合い。

#### ⑤ 利用状況網羅性 (context coverage)

明示された利用状況及び当初明確に識別されていた状況を超越した状況の両方の状況において、有効性、効率性、リスク回避性及び満足性を伴って製品又はシステムが使用できる度合い。副特性として次がある。

- 利用状況完全性 (context completeness)

明示された全ての利用状況において、有効性、効率性、リスク回避性及び満足性を伴って製品又はシステムが使用できる度合い。

- 柔軟性 (flexibility)

要求事項の中で初めに明示された状況を逸脱した状況において、有効性、効率性、リスク回避性及び満足性を伴って製品又はシステムが使用できる度合い。

品質モデルについての詳細は、「つながる世界のソフトウェア品質ガイド」(2015年5月29日 IPA/SEC 発行)の「SQuaRE 品質モデル活用リファレンス編」が参考になる。

## 【2】アシュアランス (保証) ケース

アシュアランスケースとは、想定している環境下において、システムが正しく動作することを構造化することによって、体系立てて保証する方法をいう<sup>9</sup>。つまり、テスト結果や検証結果をエビデンスとしてそれらを根拠にシステムの安全性を議論し、システム認証者や利用者などに保証する、あるいは確認させるための方法であり、一般的にはドキュメントを作成して相手との合意性を担保する。例えば、安全性のためのセーフティケースやセキュリティのためのセキュリティケース、ディペンダビリティのためのディペンダビリティケースなど総称してアシュアランスケースと呼ぶ。

## 【3】バグ (障害) 原因の分析

前述のとおり、「(1) 機能要件の品質」に挙げたような「製品品質モデルの8特性」や「利用時の品質モデルの5特性」の具体的な個々の品質の属性ごとにバグ (障害) が発生し、課題として取り組んでいる適用事例が多い。この項目は、色々なバグ (障害) を統計データとして収集し、分析して真の原因を追求することを指す。このような分析技術の一つに、ODC 分析 (Orthogonal Defect Classification : 直交欠陥分析) がある。

## 【4】コスト

システム・ソフトウェア開発におけるコストは、ハードウェア開発費、ソフトウェア開発費、運用・保守費、建物・設備・回線費などを指す。ソフトウェア開発費には、人件費・外注費・パッケージ費などが含まれる。

<sup>9</sup> UK Defence Standard 00-56 Issue 4: の定義より

### 〔5〕 納期

「納期」とは、スケジュール上、この日までにこの作業または成果物を完成させ要求された側に納める日時である。ソフトウェア開発は共通フレーム 2013 で示すとおり、複数のプロセスにまたがっている。システムやソフトウェアの完成をいうだけでなく、それぞれのプロセスの成果物についても、目標完成時期について納期という場合もある。

### 〔6〕 生産性（対応時間短縮）

「生産性 (Productivity) とは、投入量と産出量の比率である。」<sup>10</sup> ソフトウェア開発における産出量をソースコード行数やドキュメント枚数などにし、投入量を一人月（または一ヶ月に働く標準時間）でソフトウェア開発の生産性として、見積り用の基準にしている企業もある。しかし、ソフトウェア開発には、開発ツールの費用（購入時・保守時）や開発ツールを使うための教育時間、人材のスキルレベルなど、投入量や産出量ともに色々な要素があり、厳密な生産性として明確にすることは難しい。このような測定しづらい中でも、生産性の向上は企業収益をあげる魔法の言葉として、プロジェクト管理の関心事項として頻繁に使われる。生産性は効率や能率と同じ概念で、効率性を示す指標である。しかし、効率や能率と比べ、生産性は具体的な数値尺度を持っているという特徴がある<sup>11</sup>。

### 〔7〕 人材育成・意識改革

先進的な技術・手法を導入すると、今までの仕事の仕方や考え方など大きく異なる。そのため、プロジェクトや部門、全社規模で導入する場合には、様々な観点から人材育成を考える必要がある。また、従業員に対する丁寧な説明も重要だ。この部分をおろそかにすると、具体的な技術・手法の習得の前に、「なぜ、今までのやり方を変えなければいけないのか」、「この忙しい時に新しい技術を修得する余裕はない」など後ろ向きな意識が芽生え、変化に適応しづらい状況に陥ることがある。

### 〔8〕 プロジェクトマネジメント

プロジェクトマネジメントとは、計画されたプロジェクトを計画されたリソースと納期などを守り成功させる活動である。従来は、品質・コスト・納期 (QCD) の管理活動と考えられ、特に独立した要素と捉えられていなかった。

システム・ソフトウェア開発のプロジェクトでは、品質管理や要員管理、進捗管理など、多くの管理項目がある。先進的な技術・手法を適用するような新しい取組みでは、今まで管理していた項目以外にも管理上の課題が発生する。

プロジェクトマネジメントにも "Project Management Body of Knowledge" (PMBOK) という知識体系が整理され、2012 年には ISO 21500:2012<sup>12</sup> が発行されている。

### 〔9〕 見積支援

見積とは、あらかじめ、作業量や作業工数などを概算することで、見積支援は、そのためのデータやツールを提供することである。

### 〔10〕 普及促進

ソフトウェア開発において、抱えている課題を解決するために試みたことが効果を発揮した時に、その仕組みを広く行き渡らせることを促すことである。

### 〔11〕 体制（強化・再構築）

先進的な技術・手法を導入する時や抱えている課題解決の目処が立った時など、今までのやり方を変更する中で組織として活動を支援する方法として体制の強化や体制の再構築をすることである。

10, 11 公益財団法人 日本生産性本部「生産性の定義」から引用

<http://www.jpc-net.jp/movement/productivity.html>

12 Guidance on project management



## [12] グローバル展開

抱えている課題を解決した先進的な技術・手法を言葉の壁や文化の壁のある国外でも適用できるようにしている状態である。



表 2-5 効果を発揮した項目に整理・分類したもの

	事例参照番号	効果項目																							
		品質										その他													
		機能適合性	性能効率性	互換性	使用性	信頼性	セキュリティ	保守性	移植性	有効性	効率性	満足性	リスク回避性	利用状況網羅性	アシュアランス(保証)	障害原因の分析	コスト	納期	生産性	人材育成	プロジェクト	見積支援	普及促進	体制	グローバル展開
		38	12	7	12	17	7	27	11	8	6	8	4	5	11	4	34	18	36	19	10	2	7	11	1
1	A-1	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	
2	A-2	■														■		■		■					
3	A-3			■															■						
4	A-4	■	■	■												■		■	■	■					
5	A-5	■	■			■										■	■				■				
6	A-6	■		■									■						■					■	
7	A-7	■														■	■		■					■	
8	A-8															■	■	■	■				■	■	
9	A-9	■	■	■	■														■					■	
10	A-10	■																	■	■					
11	A-11	■				■													■	■					
12	A-12	■																	■	■					
13	A-13	■		■															■	■	■			■	
14	15-A-1	■	■	■	■											■		■	■	■				■	
15	15-A-2	■		■															■					■	
16	15-A-3	■	■	■	■														■						
17	15-A-4	■		■															■	■				■	
18	15-A-5	■	■	■	■														■	■	■			■	
19	15-A-6	■												■					■	■					
20	15-A-7	■																	■	■	■				
21	15-A-8																		■	■	■				
22	15-A-9					■								■					■						
23	15-A-10																		■						
24	15-A-11																		■	■					
25	15-A-12	■				■													■	■				■	
26	15-A-13	■																	■	■					
27	15-A-14	■												■					■	■					
28	15-A-15	■																	■						
29	15-A-16	■	■	■	■														■	■				■	
30	15-A-17	■																	■						
31	15-A-18	■				■													■	■					
32	15-A-19					■													■	■					
33	15-A-20					■													■	■					
34	B-1	■																							
35	B-2																								
36	B-3		■	■	■	■	■	■	■															■	
37	B-4	■		■	■	■	■	■	■											■					
38	B-5																			■					
39	B-6	■																						■	
40	B-7					■																			
41	B-8	■																		■	■				
42	B-9	■																		■	■				
43	B-10	■																		■					
44	B-11	■																		■					
45	15-B-1	■				■														■	■				
46	15-B-2	■																		■	■				
47	15-B-3	■																		■	■				
48	15-B-4	■																		■	■				
49	15-B-5	■																		■	■				
50	15-B-6	■																		■					
51	15-B-7	■				■														■					
52	15-B-8																			■	■			■	
53	15-B-9	■				■														■					
54	15-B-10	■																		■	■				
55	15-B-11	■																		■	■				
56	15-B-12	■																		■	■	■			
57	15-B-13	■				■														■	■	■			
58	15-B-14	■																		■	■				

## 2.2.4 分析のステップ

---

分析では、整理・分類された内容を元に、以下の5つのステップを行う。

ステップ1：適用事例を適用領域①や適用領域②、適用技術・手法ごとに抽出

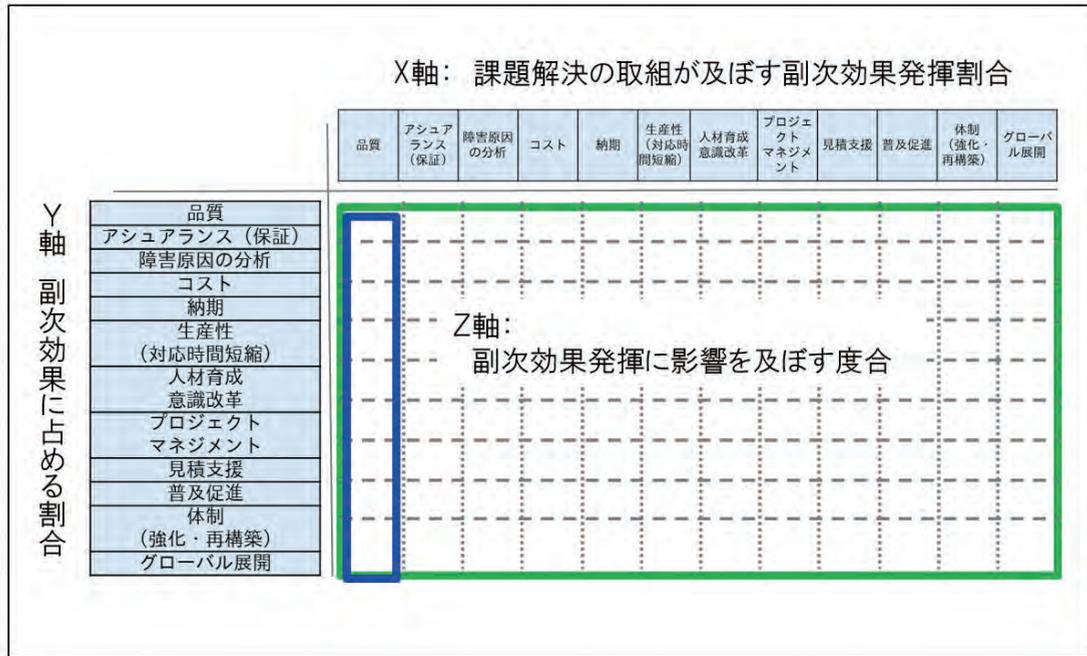
ステップ2：抽出した適用事例をテーマごとに適用結果を集計

ステップ3：集計結果を元に傾向を分析

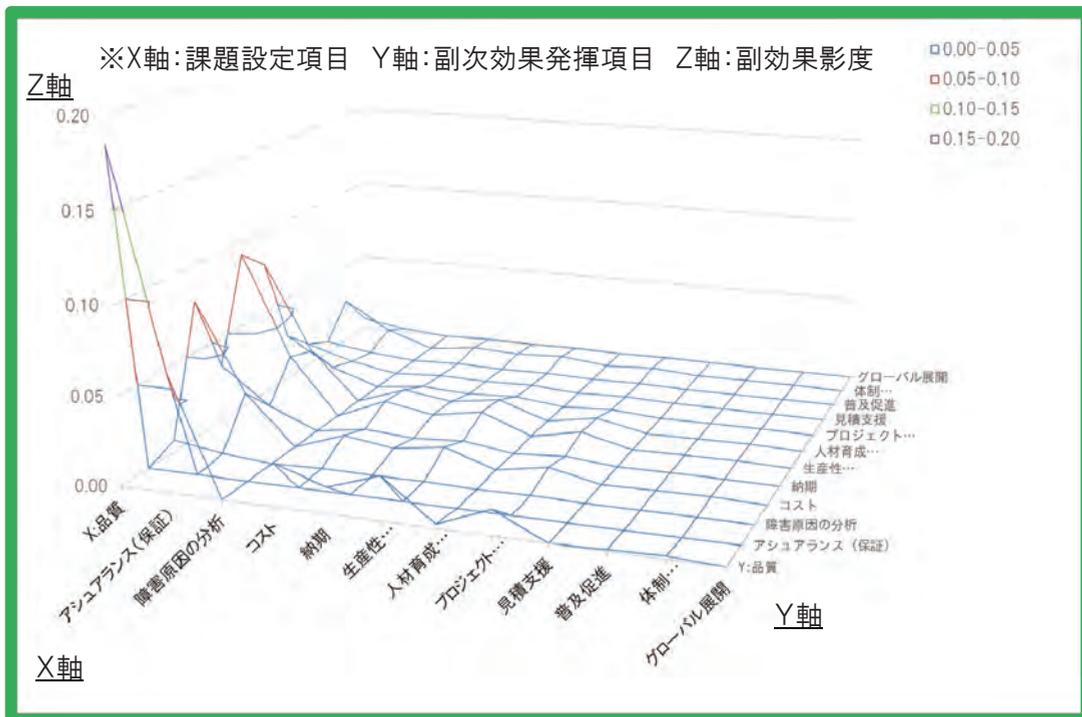
ステップ4：さらに課題への取組みと副次効果発揮の相関関係を分析



ステップ3：集計結果を元に傾向を分析



ステップ4：課題の取組と副次効果発揮の相関関係を分析



# 適用事例及び アンケート結果の分析

.....	56
3.1 全適用事例の分析 .....	57
3.1.1 全適用事例から見る「適用結果」の状況 .....	58
3.1.2 全適用事例から見る「課題」の状況 .....	59
3.1.3 全適用事例から見る「効果」の状況 .....	60
3.1.4 全適用事例から見る「副次効果」の状況 .....	61
3.1.5 全適用事例から見る「副次効果」の結果をさらに分析 .....	62
3.1.6 全適用事例から見る「さらなる取組」の状況 .....	65
3.1.7 全適用事例から見る「新規取組」の状況 .....	66
3.1.8 課題解決の定量的な効果測定状況 .....	67
3.1.9 先進的な技術・手法の適用度合 .....	68
3.2 適用対象分野ごとの分析 .....	69
3.2.1 適用領域①（設計系、検証系）別の適用結果等状況 .....	69
3.2.1.1 設計系 .....	70
3.2.1.2 検証系 .....	76
3.2.1.3 適用領域①（設計系、検証系）で分かったこと .....	82
3.2.2 適用領域②（エンタプライズ系、Web・フロント系、組込み・ 制御系）別の適用結果等状況 .....	83
3.2.2.1 エンタプライズ系 .....	85
3.2.2.2 Web・フロント系 .....	91
3.2.2.3 組込み・制御系 .....	97
3.2.2.4 適用領域②（エンタプライズ系、Web・フロント系、組込み・ 制御系）で分かったこと .....	103
3.2.3 適用技術・手法別の適用結果等状況 .....	104
3.2.3.1 適用結果 .....	104
3.2.3.2 適用技術・手法別の効果を発揮したテーマ .....	105
3.2.3.3 適用工程 .....	107
3.2.3.4 適用領域②別適用工程 .....	108
3.2.3.5 併用されている技術・手法 .....	109
3.3 適用技術・手法ごとの分析 .....	110
3.3.1 技術・手法の説明 .....	110
3.4 アンケート .....	116
3.4.1 アンケート内容 .....	116
3.4.2 アンケート結果 .....	117

# 3章 適用事例及びアンケート結果の分析

2.2.4 章で説明したように5つのステップで分析を行った結果を3つの切り口で解説する。

1. 複数のプロジェクトを総合的に管理している立場 → 「3.1 章 全適用事例の分析」
2. 具体的なプロジェクトに関わっている立場 → 「3.2 章 適用対象分野ごとの分析」  
 ※適用対象分野とは、適用領域①、適用領域②と適用技術・手法を総称した表現とする。
3. 今回適用されている技術・手法を掘り下げて情報収集したい立場  
 → 「3.3 章 適用技術・手法ごとの分析」

分析結果は、適用対象分野別に解説する。それぞれの参照先を表3-1に示す。

さらに、提供者のアンケートを分析した結果を「3.4 章 アンケート」で解説する。

なお、品質関連については、品質項目が多岐にわたるため、JIS X 25010 : 2013 (ISO/IEC 25010 : 2011) のシステム及びソフトウェア品質モデルを構成している品質副特性の13項目に細分化して分析を行った。

表 3-1 分析結果参照先

全体 → 3.1章	適用領域① → 3.2.1章	適用領域② → 3.2.2章	適用技術・手法 → 3.2.3章 → 3.3章	
全部の適用事例	設計系 → 3.2.1.1章	エンタプライズ系 → 3.2.2.1章	アジャイル開発	
		Web・フロント系 → 3.2.2.2章	イテレーション開発	
		組込み・制御系 → 3.2.2.3章	BPM	
	検証系 → 3.2.1.2章	エンタプライズ系 → 3.2.2.1章	MBSE	
		Web・フロント系 → 3.2.2.2章	形式仕様記述	
		組込み・制御系 → 3.2.2.3章	HCD	
				UX
				.....
				アシュアランスケース
			セキュリティ	

### 3.1 全適用事例の分析

この章では、複数のプロジェクトを総合的に管理している立場の人が、開発プロセスに先進的な技術・手法を適用した時の結果を理解し、その結果を元に関係組織に新しい先進的な技術・手法を導入する必要性を認識できるヒントになる事項を説明する。具体的には、全適用事例 58 件の適用結果状況とテーマの状況を説明する。

分析結果の中の数字を以下のように2つの視点で表現している。

- 件数：対象となる事例の数
- 個数：対象となるテーマの数

適用結果とテーマとは、図 3-1 のとおりである。

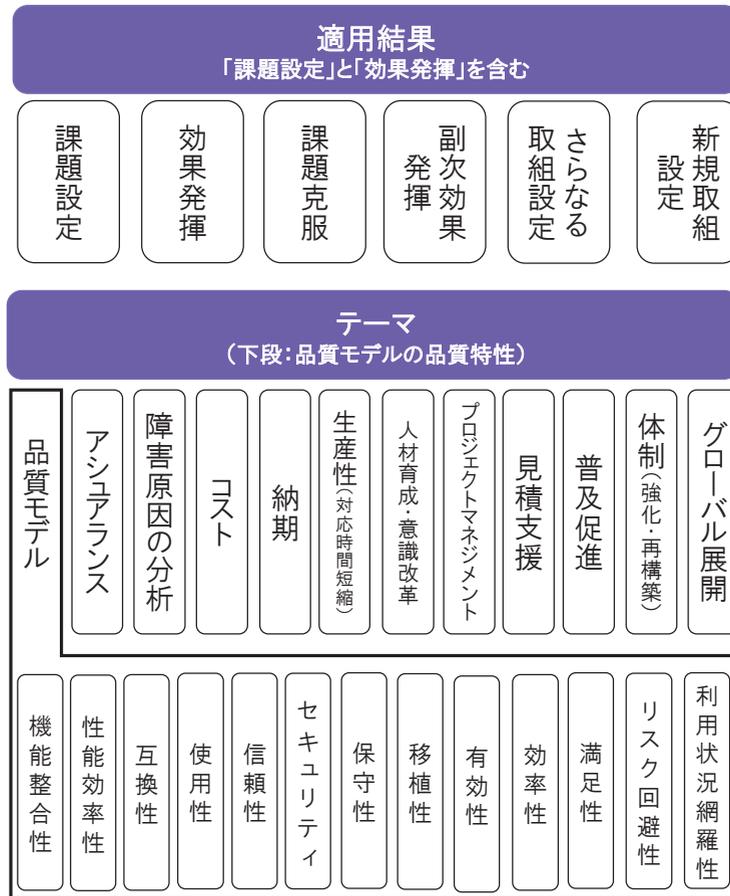


図 3-1 適用事例全般で分析する 2つの視点

### 3.1.1 全適用事例から見る「適用結果」の状況

58件の全適用事例の適用結果は、図3-2のような発生数になった。

また、表3-2のように、課題設定件数に対して効果を発揮した割合は126%で、課題設定以上に多くの効果を発揮できたことを示す。課題克服率は、97%と成功事例の集まりで、この事例集は読者には有益な情報になることを示す。課題設定時に挙げていなかったテーマが効果発揮できた副次効果発生率は29% (84個)であった。このことは、新しい取組みをすることで想定外の効果を誘発できたことを示すもので、この結果が積極的に新しい取組みをするきっかけになるだろう。今後への取組みも「さらなる取組（既に課題設定・効果発揮しているテーマにさらに取り組みむこと）」と「新規の取組（今回の取組みでは課題設定も効果発揮もしていないテーマに新規に取り組みむこと）」で105個（44個+61個）もあり、ソフトウェア開発の最適化に向け好循環を生み出すきっかけになっていることを示す。

なお、副次効果発生率と「さらなる取組」と「新規取組」の項目に対して、全事例数に対するそれぞれの発生事例件数については3.1.4章、3.1.5章、3.1.6章で解説する。

次項以降で適用結果に基づいた分析結果を示すが、「課題克服」に関しては97%を達成し、課題設定の傾向と同じになるため、結果解説を割愛する。

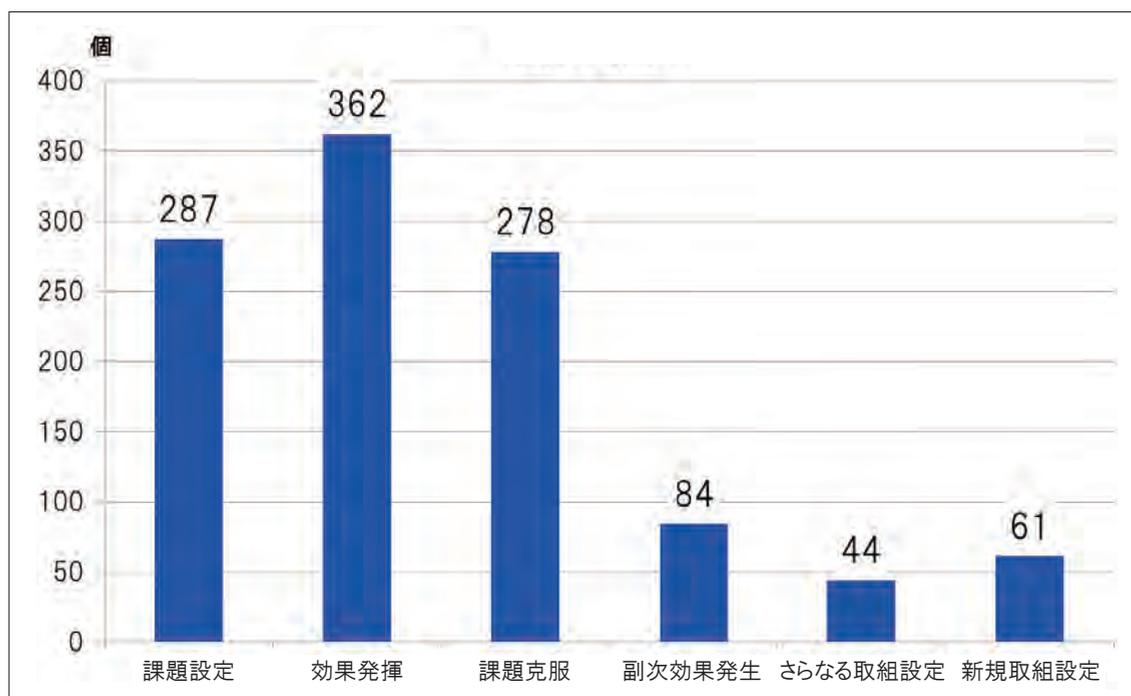


図 3-2 適用事例全体の適用結果ごとの発生数

表 3-2 適用結果の課題設定数に対する発生率（全適用事例）

全体	事例数	課題設定	効果発揮	課題克服	副次効果発生	さらなる取組設定	新規取組設定
個数	58	287	362	278	84	44	61
割合（率）	—	—	126%	97%	29%	15%	21%
割合(率)の算出方法	—	—	(効果発揮数÷課題設定数)	(課題克服数÷課題設定数)	(副次効果発生数÷課題設定数)	(さらなる取組設定数÷課題設定数)	(新規取組設定数÷課題設定数)
発生事例数	—	58	58	58	32	30	34

### 3.1.2 全適用事例から見る「課題」の状況

課題設定の 287 個の内訳は、表 3-3 のように品質モデルの品質特性の課題設定数 164 個と品質モデルを除く 123 個である。

一事例当たりの課題設定数は、4.9 個（算出式：287 個 ÷ 58 件）である。内訳として、品質モデルの品質特性では、2.8 個（算出式：164 個 ÷ 58 件）である。品質以外では、2.1 個（算出式：123 個 ÷ 58 件）である。

また、課題を設定しているテーマ別の課題設定率の上位は、品質（100%）、コスト（45%）、生産性（45%）、納期（26%）、アシュアランス（24%）であった。品質モデルの品質特性では、機能適合性（59%）、保守性（47%）が突出している。

保守性の課題設定率が高かったことから、最近の活発な保守性の向上に向けた各社・団体の活動にも起因している可能性はある。

表 3-3 課題のテーマの内訳と一事例当たりの課題設定率

課題のテーマ：課題設定数（品質を除いて123個）。一事例当たり2.1個

	品質	コスト	生産性 (対応時間 短縮)	納期	アシュア ランス (保証)	人材育成・ 意識改革	プロジェクト マネジメント	普及促進	体制 (強化・再 構築)	障害原因の 分析	見積支援	グローバル 展開
対象事例数	58	26	26	15	14	10	9	9	9	5	0	0
課題設定率	100%	45%	45%	26%	24%	17%	16%	16%	16%	9%	0%	0%

品質モデルの品質特性:品質特性の課題設定数(164個)。一事例当たり 2.8個

	機能適合性	保守性	信頼性	性能効率性	移植性	満足性	使用性	互換性	セキュリティ	有効性	効率性	リスク回避性	利用状況網 羅性
対象事例数	34	27	13	12	12	11	10	9	9	9	7	6	5
課題設定率	59%	47%	22%	21%	21%	19%	17%	16%	16%	16%	12%	10%	9%

### 3.1.3 全適用事例から見る「効果」の状況

効果発揮の362個の内訳は、表3-4のように品質モデルの品質特性の効果発揮数187個と品質モデルを除く175個である。

一事例当たりの効果発揮数は、6.2個である。内訳として、品質モデルの品質特性では、3.2個（算出式：187個÷58件）である。品質以外では、3.0個（算出式：175個÷58件）である。

また、効果が発揮されているテーマ別の効果発揮率の上位は、品質（100%）、生産性（69%）、コスト（64%）、人材育成・意識改革（38%）、納期（33%）であった。品質モデルの品質特性では、機能適合性（69%）、保守性（50%）が突出している。

この結果、人材育成・意識改革に関して、3.1.2章の課題設定では10事例であったにも関わらず、効果発揮で22事例と倍増していることは、新しい取組みを進める上で必然的に人材育成・意識改革を向上させる工夫がされていることが想定される。

表 3-4 効果のテーマの内訳と一事例当たりの効果発揮率

効果のテーマ:効果発揮数(品質を除いて175個)。一事例当たり3.0個

	品質	生産性 (対応時間 短縮)	コスト	人材育成・ 意識改革	納期	アシュア ランス(保証)	体制 (強化・再 構築)	プロジェクト マネジメント	普及促進	障害原因の 分析	見積支援	グローバル 展開
対象事例数	58	40	37	22	19	13	13	12	10	6	2	1
効果発揮率	100%	69%	64%	38%	33%	22%	22%	21%	17%	10%	3%	2%

品質モデルの品質特性:品質特性の効果発揮数(187個)。一事例当たり3.2個

	機能適合性	保守性	信頼性	性能効率性	使用性	移植性	満足性	有効性	セキュリティ	互換性	効率性	リスク回避性	利用状況網 羅性
対象事例数	40	29	17	13	13	13	12	11	10	8	8	7	6
効果発揮率	69%	50%	29%	22%	22%	22%	21%	19%	17%	14%	14%	12%	10%

### 3.1.4 全適用事例から見る「副次効果」の状況

#### 〔1〕 全事例に対する副次効果発揮状況

58 件の事例に対して 32 件の事例（表 3-2 参照）で副次効果が発生している。発揮率は 55% である。（算出式：副次効果発揮事例数 ÷ 事例数 = 32 件 ÷ 58 件）

これは、ひとつの課題に取り組むと、当初想定していない別の課題の解決につながるという点で、先進的な開発技術への取組みは、有益であることを意味する。

#### 〔2〕 事例／テーマ別の副次効果発揮状況

副次効果発揮の 84 個の内訳は、表 3-5 のように品質モデルの品質特性の副次効果発揮数 28 個と品質モデルを除く 56 個である。

一事例当たりの副次効果発揮数は、1.5 個（算出式：84 個 ÷ 58 件）である。内訳として、品質モデルの品質特性では、0.5 個（算出式：28 個 ÷ 58 件）である。品質以外では、1.0 個（算出式：56 個 ÷ 58 件）である。

また、副次効果発揮率のテーマ別上位は、品質（24%）、生産性（24%）、コスト（21%）、人材育成・意識改革（21%）である。品質モデルの品質特性では、機能適合性（12%）、保守性（7%）である。

テーマ別に分析すると、3.1.1 章で解説したように課題設定率の高い品質（100%）、コスト（45%）、生産性（45%）の項目に対して副次効果を発揮している。ただし、品質に関しては前述のとおり、品質特性まで細分化した中で副次効果発生状況を見ている。副次効果の発生状況については、3.1.5 章でさらに分析した結果を解説する。

#### 〔3〕 課題設定数に対する副次効果発揮状況

287 個の課題設定数に対して副次効果発揮数は 84 個で 29%（算出式：84 個 ÷ 287 個）の発生率である。この数字は高く、課題解決の取組みをすることは当初設定の課題以外に別の項目でも良い結果が出る。

表 3-5 副次効果のテーマの内訳と一事例当たりの副次効果発揮率

副次効果のテーマ:副次効果発揮数(品質を除いて 56個)。一事例当たり1.0個

	品質	生産性 (対応時間 短縮)	コスト	人材育成・ 意識改革	納期	プロジェクト マネジメント	体制 (強化・再 構築)	見積支援	障害原因の 分析	普及促進	グローバル 展開	アシユアラ ンス (保証)
対象事例数	14	14	12	12	5	4	4	2	1	1	1	0
副次効果発揮率	24%	24%	21%	21%	9%	7%	7%	3%	2%	2%	2%	0%

品質モデルの品質特性:品質特性の副次効果発揮数(28個)。一事例当たり0.5個

	機能適合性	信頼性	使用性	保守性	性能効率性	有効性	満足性	セキュリティ	移植性	効率性	リスク回避性	利用状況網 羅性	互換性
対象事例数	7	4	3	3	2	2	2	1	1	1	1	1	0
副次効果発揮率	12%	7%	5%	5%	3%	3%	3%	2%	2%	2%	2%	2%	0%

### 3.1.5 全適用事例から見る「副次効果」の結果をさらに分析

副次効果が導き出されるきっかけになる事象を、さらに分析した。課題解決に取り組んだテーマと副次効果を発揮したテーマの2つの相関関係を定量化することで、副次効果発揮に影響を及ぼす割合が分かる。具体的には、図3-3のようにテーマごとにX軸に課題解決の取組みが及ぼす副次効果発揮割合、Y軸に副次効果に占める割合からそれぞれを掛け合わせ（乗算）て、副次効果発揮に影響を及ぼす割合（Z軸）を算出する。

その結果、表3-6のように品質の課題を解決する取組みが副次効果に及ぼす影響が一番大きいことを示す。この相関関係を図3-4のように3次元のグラフで示すとさらにビジュアルに示することができる。一番影響を及ぼす品質の課題解決の取組みの結果は、図3-5のように品質（0.19）、生産性（0.09）、コスト（0.08）、人材育成・意識改革（0.08）が副次的な効果を発揮する影響が高いことが分かった。

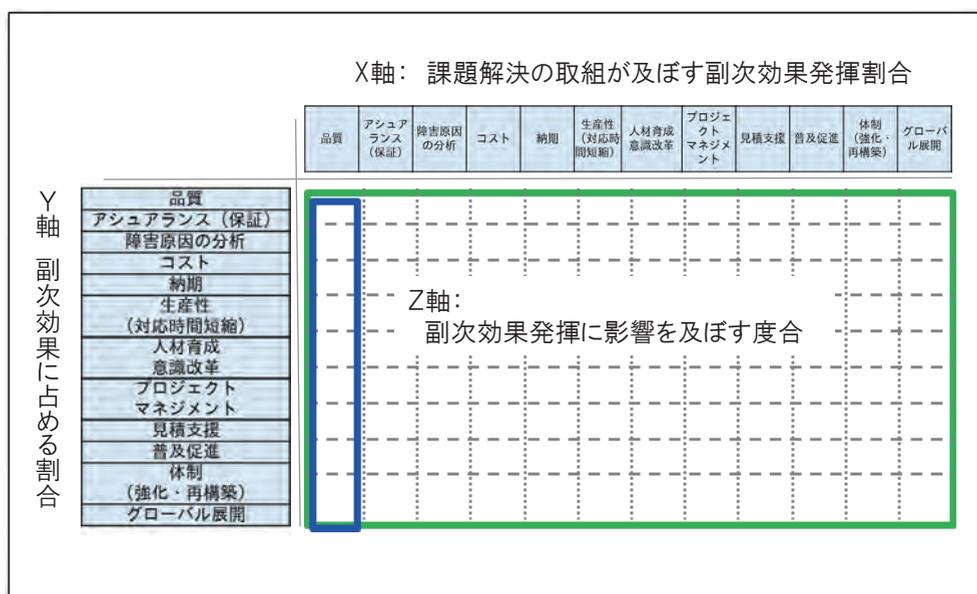


図 3-3 課題解決に取り組んだテーマと副次効果を発揮したテーマの2つの相関関係（イメージ）

表 3-6 課題解決の取組みが及ぼした副次効果発揮割合と副次効果に占める割合の相関関係

副次効果	課題	品質	アシュアランス(保証)	障害原因の分析	コスト	納期	生産性(対応時間短縮)	人材育成意識改革	プロジェクトマネジメント	見積支援	普及促進	体制(強化・再構築)	グローバル展開
品質		0.19	0.07	0.00	0.02	0.01	0.02	0.00	0.01	0.00	0.00	0.00	0.00
アシュアランス(保証)		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
障害原因の分析		0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
コスト		0.08	0.03	0.00	0.01	0.01	0.01	0.00	0.01	0.00	0.00	0.00	0.00
納期		0.03	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
生産性(対応時間短縮)		0.09	0.03	0.00	0.01	0.01	0.01	0.00	0.01	0.00	0.00	0.00	0.00
人材育成意識改革		0.08	0.03	0.00	0.01	0.01	0.01	0.00	0.01	0.00	0.00	0.00	0.00
プロジェクトマネジメント		0.03	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
見積支援		0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
普及促進		0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
体制(強化・再構築)		0.03	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
グローバル展開		0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

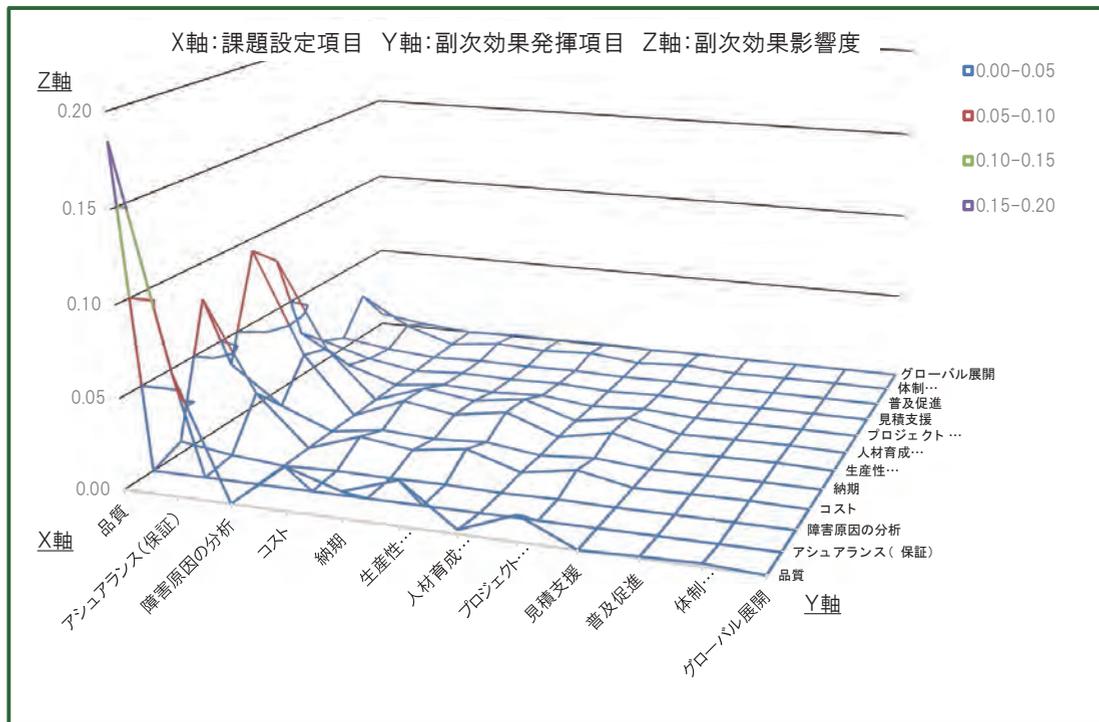


図 3-4 課題解決の取組みが及ぼした副次効果発揮割合と副次効果に占める割合の相関関係

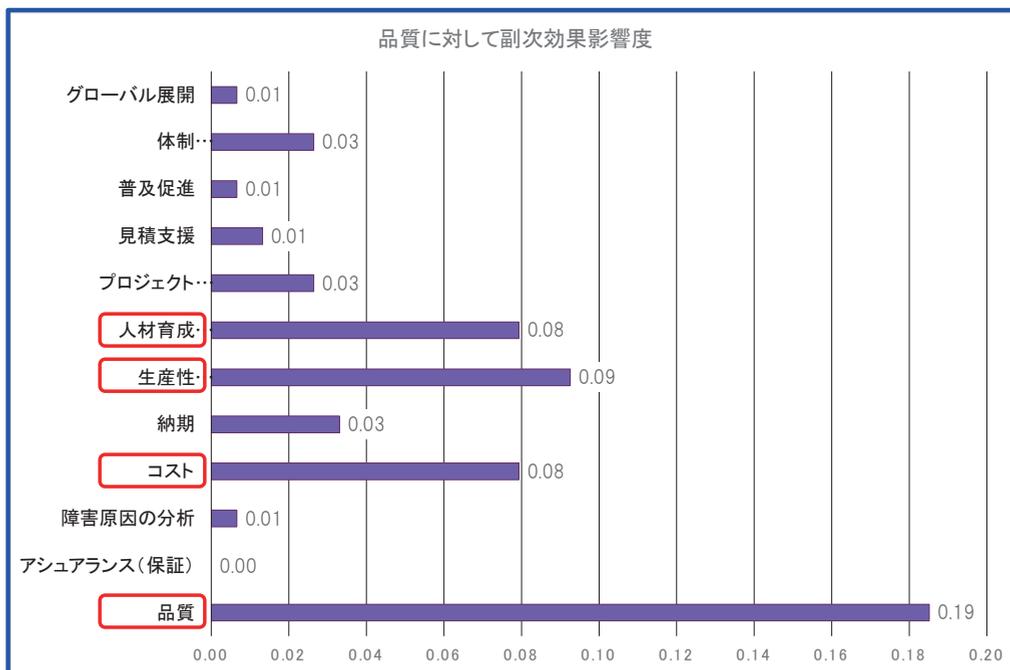


図 3-5 品質の課題解決の取組みに影響を及ぼす割合

### 3.1.6 全適用事例から見る「さらなる取組」の状況

#### [1] 全事例に対するさらなる取組設定状況

58 件の事例に対して 30 件の事例（表 3-2 参照）でさらなる取組が発生している。発生率は 52% である。  
 (算出式：さらなる取組事例数 ÷ 事例数 = 30 件 ÷ 58 件)

この数字は、当初取り組んだ課題を再度取組みの必要がある事例が半数以上あることを示している。当初の課題の解決の効果はでているものの（課題克服率 97% より）、さらに同じ課題テーマでの取組み続ける傾向が伺える。

#### [2] 事例／テーマ別のさらなる取組設定状況

さらなる取組の 44 個の内訳は、表 3-7 のように品質モデルの品質特性のさらなる取組数 20 個と品質モデルを除く 24 個である。

一事例当たりのさらなる取組数は、0.7 個（算出式：44 個 ÷ 58 件）である。内訳として、品質モデルの品質特性のみでは、0.3 個（算出式：20 個 ÷ 58 件）である。品質以外では、0.4 個（算出式：24 個 ÷ 58 件）である。

また、さらなる取組設定率のテーマ別上位は、品質（33%）、コスト（10%）、生産性（10%）、普及促進（9%）である。品質モデルの品質特性では、機能適合性（24%）、保守性（7%）である。

テーマ別に分析すると、普及促進の項目に対して課題設定対象事例数（表 3-5）が 9 個であるのに対して、さらなる取組対象事例数が 5 個であり、その割合は 56%（算出式：5 個 ÷ 9 個）と高い。継続して今後の課題取組みが必要であることは、普及促進の課題は満足の行く効果を発揮するのが大変であることを物語っている。

#### [3] 課題設定数に対するさらなる取組状況

課題設定を元に考えると、287 個の課題設定数に対してさらなる取組設定数は 44 個で 15%（算出式：44 個 ÷ 287 個）の発生率である。

表 3-7 さらなる取組設定のテーマの内訳と全事例当たりのさらなる取組設定率

さらなる取組設定のテーマ:さらなる取組設定数(品質を除いて24個)。一事例当たり0.4個

	品質	コスト	生産性 (対応時間 短縮)	普及促進	体制 (強化・再 構築)	人材育成・ 意識改革	プロジェ クトマネジ メント	アシユアラ ンス (保証)	障害原因 の分析	納期	見積支援	グローバル 展開
対象事例数	19	6	6	5	3	2	2	0	0	0	0	0
さらなる取組設定率	33%	10%	10%	9%	5%	3%	3%	0%	0%	0%	0%	0%

品質モデルの品質特性:品質特性のさらなる取組設定数(20個)。一事例当たり0.3個

	機能適合性	保守性	使用性	信頼性	性能効率性	互換性	セキュリティ	移植性	有効性	効率性	満足性	リスク回避 性	利用状況 網羅性
対象事例数	14	4	1	1	0	0	0	0	0	0	0	0	0
さらなる取組設定率	24%	7%	2%	2%	0%	0%	0%	0%	0%	0%	0%	0%	0%

### 3.1.7 全適用事例から見る「新規取組」の状況

#### (1) 全事例に対する新規取組状況

58件の事例に対して34件の事例（表3-2参照）で新規取組が発生している。発生率は59%である。（算出式：新規取組事例数 ÷ 事例数 = 34件 ÷ 58件）

先進的な開発技術への取組みをすると、半数以上の事例で当初設定していた課題以外に新規取組が発生した。

#### (2) 事例／テーマ別の新規取組設定状況

新規取組の61個の内訳は、表3-8のように品質モデルの品質特性の新規取組数12個と品質モデルを除く49個である。

一事例当たりの新規取組件数は、0.8個（算出式：49個 ÷ 58件）である。また、品質特性のみでは、0.2個（算出式：12個 ÷ 58件）である。

また、新規取組設定率のテーマ別上位は、普及促進（33%）、プロジェクトマネジメント（17%）、人材育成・意識改革（12%）である。品質モデルの品質特性では、機能適合性（10%）、保守性（3%）である。

テーマ別に分析すると、普及促進やプロジェクトマネジメントは、当初の課題設定や効果発揮の項目としては事例対象件数が少なかったが、第二の取組みとして重要視されてきていることが分かる。

#### (3) 課題設定数に対する新規取組状況

287個の課題設定数に対して新規取組設定数は61個で21%（算出式：61個 ÷ 287個）の発生率である。

表 3-8 新規取組設定のテーマの内訳と全事例当たりの新規取組設定率

新規取組設定のテーマ:新規取組設定数(品質を除いて49個)。一事例当たり0.8個

	品質	普及促進	プロジェクトマネジメント	人材育成・意識改革	コスト	生産性(対応時間短縮)	体制(強化・再構築)	障害原因の分析	アシアランス(保証)	納期	見積支援	グローバル展開
対象事例数	12	19	10	7	4	4	4	1	0	0	0	0
新規取組設定率	21%	33%	17%	12%	7%	7%	7%	2%	0%	0%	0%	0%

品質モデルの品質特性:品質特性の新規取組設定数(12個)。一事例当たり0.2個

	機能適合性	保守性	性能効率性	互換性	使用性	移植性	信頼性	セキュリティ	有効性	効率性	満足性	リスク回避性	利用状況網羅性
対象事例数	6	2	1	1	1	1	0	0	0	0	0	0	0
新規取組設定率	10%	3%	2%	2%	2%	2%	0%	0%	0%	0%	0%	0%	0%

### 3.1.8 課題解決の定量的な効果測定状況

適用事例数は 58 件のうち、定量的な効果測定結果が提示されたものは、25 件である。

先進的な取組みは、効果を定量的に可視化しないと、経営者はそれに対する投資の継続判断ができない。また、プロジェクトメンバーなどの関係者にとっても、効果を実感することができない。よって、定量的効果測定方法をプロジェクトの開始前に設定しておくことは有用だと考えられるが、定量的数値の取得方法が難しい、適用していない場合と適用した場合の比較が難しい等の課題があるものと推定される。

表 3-9 に示した測定内容は、定量的な数値が得られているので、今後の定量的な効果測定方法の参考になる。

表 3-9 課題解決のための定量的な効果測定内容

効果測定項目	5	4	3	2	1	0	合計
適用事例数	1	1	3	6	14	33	58

25 件

	不具合数	不具合（バグ）密度（摘出率）	バグ曲線	品質測定	テスト回数	テスト実行件数（日次）	テストケース密度	ラインカバレッジ率	故障モードの識別数	ドキュメント枚数	チェックリスト	欠陥改修日数	ソースコードの流用率	工数割合	開発規模	開発費用	工数（人月・日数・時間）	開発期間	コード数比較	プログラム生産性	プロジェクト数	売上	
品質	○	○	○	○	○	○	○	○	○	○	○	○	○										
コスト												○	○	○	○	○	○	○					
納期																	○	○					
生産性																	○	○	○	○			
普及度																					○		
売上																						○	

全体の件数	4	4	2	1	3	1	1	4	1	1	1	1	1	1	1	2	9	2	1	1	1	1
設計系の件数	3	2	1	1	0	0	1	1	1	0	0	1	1	1	2	8	2	1	1	0	1	1
検証系の件数	1	2	1	0	2	1	1	3	0	0	1	1	0	0	0	0	1	0	0	0	1	0

### 3.1.9 先進的な技術・手法の適用割合

今回の収集した適用事例の普及割合を図3-6のとおり分類した。ただし、それぞれの企業の取組みの母集団のプロジェクト数や規模が明確ではなく、全ての適用事例を同じ物差しで厳密に統計化できないため、あくまで今回の適用事例収集の傾向として、捉えていただきたい。

先進的な技術・手法の適用なので、当初はプロトタイプが多いことが想定されたが、実際には100以上のプロジェクトに適用されていることが多く、かなり実践的な取組みになっていることが分った。

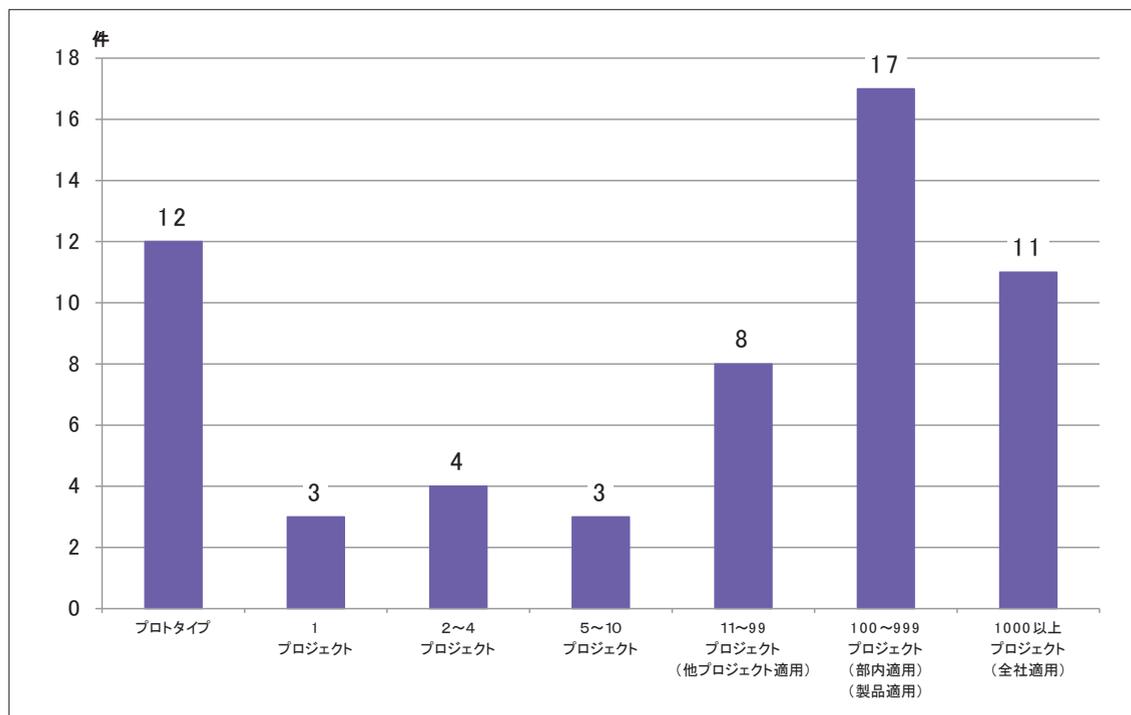


図 3-6 先進的な技術・手法の適用割合

## 3.2 適用対象分野ごとの分析

この章では、具体的なプロジェクトに関わっている立場の人が、自分の置かれている環境（開発領域や開発工程）の視点で、どのような課題を解決するために、どのような先進的な技術・手法を適用すべきかを検討するヒントになる事項を説明する。各適用対象分野（適用領域①と適用領域②と適用技術・手法を指す）における適用結果状況、テーマ、適用工程について様々な角度から詳細を示している。

### 3.2.1 適用領域①（設計系、検証系）別の適用結果等状況

適用領域①の設計系と検証系との分類は、適用事例の対象領域が上流工程主体であれば設計系、品質保証主体であれば検証系とした。

設計系と検証系を比較すると、設計の方が事例件数では1.3倍、一事例当たりの課題設定数では2.0倍多くっており、設計系の方が、より多くの課題認識を抱えていることが推測される（図3-7、表3-10）。

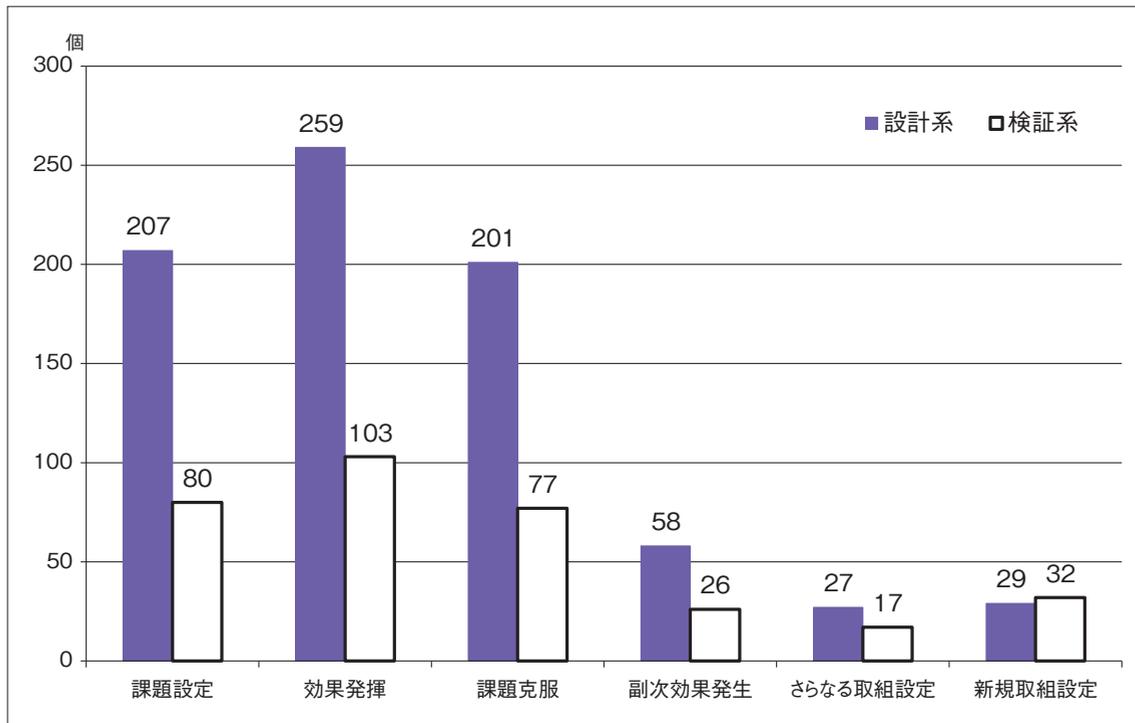


図 3-7 適用領域①別の適用結果

表 3-10 適用結果を設計系と検証系で比較

		事例数	課題設定	効果発揮	課題克服	副次効果発生	さらなる取組設定	新規取組設定
設計系	個数	33	207	259	201	58	27	29
	一事例当たりの発生数 (①)	—	6.3	7.8	6.1	1.8	0.8	0.9
検証系	個数	25	80	103	77	26	17	32
	一事例当たりの発生数 (②)	—	3.2	4.1	3.1	1.0	0.7	1.3
①÷②		1.3	2.0	1.9	2.0	1.7	1.2	0.7

### 3.2.1.1 設計系

#### 3.2.1.1.1 適用結果

全事例の58件の内、設計系は33件であった。対象となる適用事例の適用結果は図3-8のような発生数となった。

また、表3-11のように、課題設定件数に対して効果を発揮した割合は125%で、課題設定以上に多くの効果を発揮できたことを示す。課題克服率は、97%と成功事例の集まりでこの事例集は読者には有益な情報になることを示す。副次効果発生率も28%（58個）であった。このことは、新しい取組みをすることで想定外の効果を誘発できたことを示すもので、この結果が積極的に新しい取組みをするきっかけになるだろう。今後への取組みもさらなる取組と新規の取組で56個（27個 + 29個）もあり、ソフトウェア開発の最適化に向け好循環を生み出すきっかけになっていることを示す。上記の結果は、「表3-2 適用結果の課題設定数に対する発生率（全適用事例）」の結果と同様の結果である。

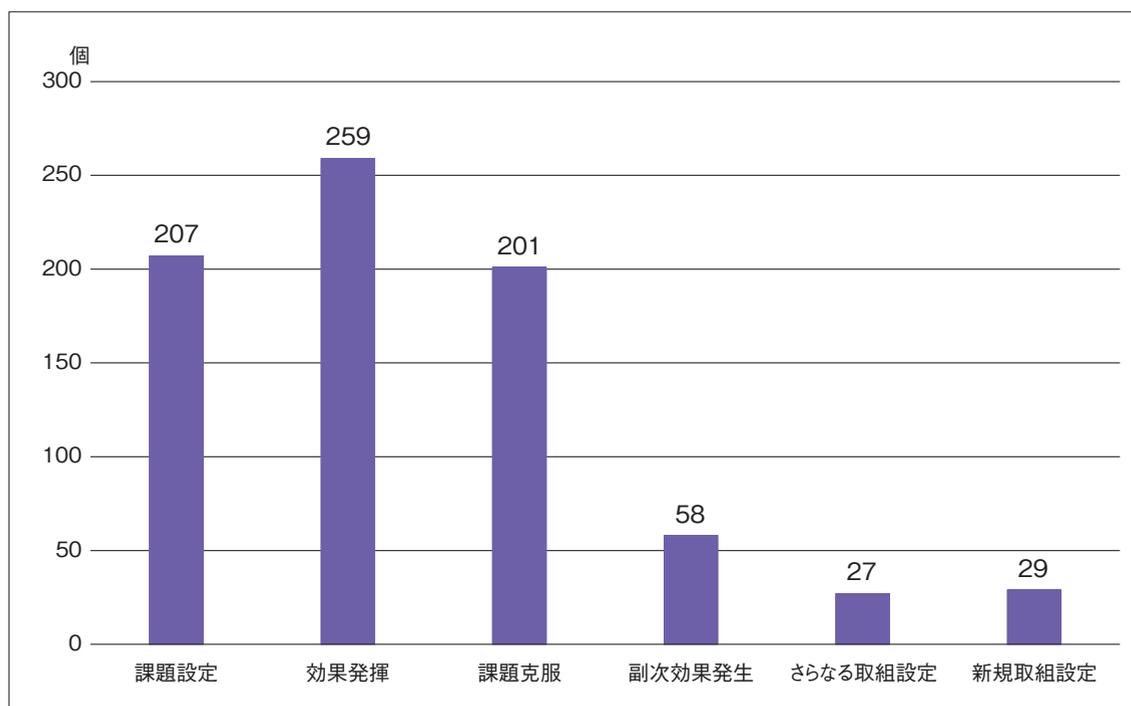


図 3-8 適用領域①の設計系の適用結果

表 3-11 適用結果の課題設定数に対する発生率（設計系）

設計系	事例数	課題設定	効果発揮	課題克服	副次効果発生	さらなる取組設定	新規取組設定
発生数	33	207	259	201	58	27	29
割合（率）	—	—	125%	97%	28%	13%	14%
割合（率）の算出方法	—	—	(効果発揮数 ÷ 課題設定数)	(課題克服数 ÷ 課題設定数)	(副次効果発生数 ÷ 課題設定数)	(さらなる取組設定数 ÷ 課題設定数)	(新規取組設定数 ÷ 課題設定数)

### 3.2.1.1.2 適用結果ごとの状況（テーマ別）

#### 〔1〕対象適用事例から見る「課題」の状況

課題設定の207個の内訳は、表3-12のように品質モデルの品質特性の課題設定数117個と品質を除く90個である。

一事例当たりの課題設定数は、6.3個（算出式：207個 ÷ 33件）である。内訳として、品質モデルの品質特性では、3.5個（算出式：117個 ÷ 33件）である。品質以外では2.7個（算出式：90個 ÷ 33件）である。

また、課題設定率のテーマ別上位は、品質（91%）、コスト（58%）、生産性（58%）、納期（30%）、アシュアランス（27%）、人材育成・意識改革（27%）、プロジェクトマネジメント（27%）であった。品質モデルの品質特性では、機能適合性（58%）、保守性（58%）が突出している。

表 3-12 課題のテーマの内訳と一事例当たりの課題設定率（設計系）

課題のテーマ:課題設定数(品質を除いて90個)。一事例当たり2.7個

	品質	コスト	生産性 (対応時間 短縮)	納期	アシュアランス (保証)	人材育成 意識改革	プロジェクト マネジメント	普及促進	体制 (強化・再構築)	障害原因の 分析	見積支援	グローバル 展開
対象事例数	30	19	19	10	9	9	9	6	6	3	0	0
課題設定率	91%	58%	58%	30%	27%	27%	27%	18%	18%	9%	0%	0%

品質モデルの品質特性:品質特性の課題設定数(117個)。一事例当たり3.5個

	機能適合性	保守性	性能効率性	信頼性	移植性	互換性	使用性	満足性	効率性	有効性	セキュリティ	リスク回避性	利用状況網 羅性
対象事例数	19	19	9	9	9	8	8	8	7	6	5	5	5
課題設定率	58%	58%	27%	27%	27%	24%	24%	24%	21%	18%	15%	15%	15%

#### 〔2〕対象適用事例から見る「効果」の状況

効果発揮の259個の内訳は、表3-13のように品質モデルの品質特性の課題設定数132個と品質を除く127個である。

一事例当たりの効果発揮数は、7.8個（算出式：259個 ÷ 33件）である。内訳として、品質モデルの品質特性では、4.0個（算出式：132個 ÷ 33件）である。品質以外では、3.8個（算出式：127個 ÷ 33件）である。

また効果発揮率のテーマ別上位は、品質（97%）、生産性（85%）、コスト（79%）、人材育成・意識改革（55%）、納期（39%）、プロジェクトマネジメント（36%）であった。品質モデルの品質特性では、機能適合性（67%）、保守性（67%）が突出している。

表 3-13 効果のテーマの内訳と一事例当たりの効果発揮率（設計系）

効果のテーマ:効果発揮数(品質を除いて127個)。一事例当たり3.8個

	品質	生産性 (対応時間 短縮)	コスト	人材育成 意識改革	納期	プロジェクト マネジメント	体制 (強化・再構築)	アシュアランス (保証)	普及促進	障害原因の 分析	見積支援	グローバル 展開
対象事例数	32	28	26	18	13	12	9	8	7	4	1	1
効果発揮率	97%	85%	79%	55%	39%	36%	27%	24%	21%	12%	3%	3%

品質モデルの品質特性:品質特性の効果発揮数(132個)。一事例当たり4.0個

	機能適合性	保守性	性能効率性	使用性	信頼性	移植性	互換性	効率性	満足性	有効性	セキュリティ	利用状況網 羅性	リスク回避性
対象事例数	22	22	10	10	10	10	8	8	8	7	6	6	5
効果発揮率	67%	67%	30%	30%	30%	30%	24%	24%	24%	21%	18%	18%	15%

〔3〕対象適用事例から見る「副次効果」の状況

(ア) 対象事例に対する副次効果発揮状況

33件の事例に対して20件の事例(表3-2参照)で副次効果が発生している。発揮率は61%である。  
(算出式:副次効果発揮事例数÷事例数=20件÷33件)

これは、ひとつの課題に取り組むと、当初想定していない別の課題の解決につながるという点で、先進的な開発技術への取組みは、有益であることを意味する。

(イ) 事例/テーマ別の副次効果発揮状況

副次効果発揮の58個の内訳は、表3-14のように品質モデルの品質特性の課題設定数18個と品質を除く40個である。

一事例当たりの効果発揮数は、1.8個(算出式:58個÷33件)である。内訳として、品質モデルの品質特性では、0.5個(算出式:18個÷33件)である。複数の品質特性を課題設定している場合も課題設定数を1とカウントすると1.2個(算出式:40個÷33件)である。

また、副次効果発揮率のテーマ別上位は、生産性(27%)、人材育成・意識改革(27%)、品質(24%)、コスト(21%)である。品質モデルの品質特性では、機能適合性(12%)、保守性(9%)が突出している。

(ウ) 課題設定数に対する副次効果発揮状況

207個の課題設定数に対して副次効果発生数は58個で28%(算出式:58個÷207個)の発生率である。

表 3-14 副次効果のテーマの内訳と一事例当たりの副次効果発揮率 (設計系)

副次効果のテーマ:副次効果発揮数(品質を除いて40個)。一事例当たり1.2個

	生産性 (対応時間 短縮)	人材育成 意識改革	品質	コスト	納期	プロジェクト マネジメント	体制 (強化・再構 築)	障害原因の 分析	見積支援	普及促進	グローバル 展開	アシユアラン ス(保証)
対象事例数	9	9	8	7	4	4	3	1	1	1	1	0
副次効果発揮率	27%	27%	24%	21%	12%	12%	9%	3%	3%	3%	3%	0%

品質モデルの品質特性:品質特性の副次効果発揮数(18個)。一事例当たり0.5個

	機能適合性	保守性	性能効率性	使用性	信頼性	セキュリティ	移植性	有効性	効率性	満足性	利用状況網 羅性	互換性	リスク回避性
対象事例数	4	3	2	2	1	1	1	1	1	1	1	0	0
副次効果発揮率	12%	9%	6%	6%	3%	3%	3%	3%	3%	3%	3%	0%	0%

[4] 対象適用事例から見る「さらなる取組」の状況

さらなる取組の27個の内訳は、表3-15のように品質モデルの品質特性の課題設定数11個と品質を除く16個である。

一事例当たりの課題設定数は、0.8個（算出式：27個 ÷ 33件）である。内訳として、品質モデルの品質特性では、0.3個（算出式：11個 ÷ 33件）である。品質以外では、0.5個（算出式：16個 ÷ 33件）である。

また、さらなる取組設定率のテーマ別上位は、品質（30%）、生産性（15%）である。品質モデルの品質特性では、機能適合性（18%）、保守性（12%）が突出している。

表 3-15 さらなる取組設定のテーマの内訳と一事例当たりのさらなる取組設定率（設計系）

さらなる取組設定のテーマ:さらなる取組設定数(品質を除いて16個)。一事例当たり0.5個

	品質	生産性 (対応時間 短縮)	コスト	体制 (強化・再構 築)	プロジェクト マネジメント	普及促進	人材育成 意識改革	アシュアラン ス(保証)	障害原因の 分析	納期	見積支援	グローバル 展開
対象事例数	10	5	3	3	2	2	1	0	0	0	0	0
さらなる取組 設定率	30%	15%	9%	9%	6%	6%	3%	0%	0%	0%	0%	0%

品質モデルの品質特性:品質特性のさらなる取組設定数(11個)。一事例当たり0.3個

	機能適合性	保守性	使用性	性能効率性	互換性	信頼性	セキュリティ	移植性	有効性	効率性	満足性	リスク回避性	利用状況網 羅性
対象事例数	6	4	1	0	0	0	0	0	0	0	0	0	0
さらなる取組 設定率	18%	12%	3%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%

[5] 対象適用事例から見る「新規取組」の状況

新規取組の29個の内訳は、表3-16のように品質モデルの品質特性の課題設定数5個と品質を除く24個である。

一事例当たりの課題設定数は、0.9個（算出式：29個 ÷ 33件）である。内訳として、品質モデルの品質特性では、0.2個（算出式：5個 ÷ 33件）である。品質以外では、0.7個（算出式：24個 ÷ 33件）である。

また、新規取組設定率のテーマ別上位は、普及促進（33%）、品質（15%）、人材育成・意識改革（15%）である。品質モデルの品質特性では、機能適合性（9%）が突出している。

表 3-16 新規取組設定のテーマの内訳と一事例当たりの新規取組設定率（設計系）

新規取組設定のテーマ:新規取組設定数(品質を除いて24個)。一事例当たり0.7個

	普及促進	品質	人材育成 意識改革	生産性 (対応時間 短縮)	プロジェクト マネジメント	体制 (強化・再構 築)	アシュアラン ス(保証)	障害原因の 分析	コスト	納期	見積支援	グローバル 展開
対象事例数	11	5	5	3	3	2	0	0	0	0	0	0
新規取組設定率	33%	15%	15%	9%	9%	6%	0%	0%	0%	0%	0%	0%

品質モデルの品質特性:品質特性の新規取組設定数(5個)。一事例当たり0.2個

	機能適合性	互換性	使用性	性能効率性	信頼性	セキュリティ	保守性	移植性	有効性	効率性	満足性	リスク回避性	利用状況網 羅性
対象事例数	3	1	1	0	0	0	0	0	0	0	0	0	0
新規取組設定率	9%	3%	3%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%

### 3.2.1.1.3 事例ごとの適用工程

設計系の適用事例の適用工程は表 3-17 とおりである。適用している工程が上流（企画フェーズから詳細設計）のものは、全事例（100%：33件）である。さらに、超上流（企画フェーズから要件定義フェーズ）のものは、33%（11件 ÷ 33件）である（図 3-9）。超上流を意識した事例が 33%ある背景には、2005 年に SEC BOOKS「経営者が参画する要求品質の確保 ～超上流から攻める IT 化の勘どころ～」で、超上流の重要性とやるべきことが解説され、2013 年に SEC BOOKS「共通フレーム 2013」で超上流のプロセスが追加されているなど、重要性が浸透していることも影響している。

また、移行・運用準備フェーズと運用・保守フェーズに適用している事例は 30%（10件 ÷ 33件）である。この数字は、図 3-10 のように保守運用費が削減されている中で、この工程で削減に向けた取組みがされていることの現れである。

※適用工程の定義は、2.2.2.2 章を参照

表 3-17 適用事例ごとの適用工程分布（設計系）

事例 参照 番号	適用工程 共通フレーム2013									
	企画 フェーズ	要件 定義 フェーズ	システム・ソフトウェア 要件定義	基本設計	詳細設計	製作・ユニットテスト	ソフトウェア結合・総合テスト	システム結合・総合テスト	移行・運用準備フェーズ	運用・保守フェーズ
A-1										
A-2										
A-3		■	■	■	■					
A-4			■				■	■		
A-5		■	■							
A-6			■	■	■	■	■	■	■	■
A-7			■	■	■					■
A-8						■	■			
A-9			■	■	■					
A-10			■	■	■	■	■	■		
A-11			■	■	■	■	■	■	■	■
A-12			■	■	■	■	■	■		
A-13			■	■	■	■	■	■		
15-A-1		■	■	■	■	■	■	■		
15-A-2		■	■	■	■	■	■	■		
15-A-3		■	■	■	■	■	■	■		■
15-A-4		■	■	■	■	■	■	■	■	■
15-A-5			■	■	■	■	■	■	■	■
15-A-6		■	■	■	■					
15-A-7			■	■	■					
15-A-8		■	■	■	■					
15-A-9		■	■	■	■					
15-A-10			■	■	■					
15-A-11										■
15-A-12			■	■	■	■	■	■		■
15-A-13			■	■	■	■	■	■		
15-A-14		■	■	■	■					
15-A-15		■	■	■	■				■	■
15-A-16					■	■				
15-A-17			■	■	■	■	■	■		
15-A-18				■	■			■		
15-A-19				■	■					
15-A-20			■	■	■	■	■	■		

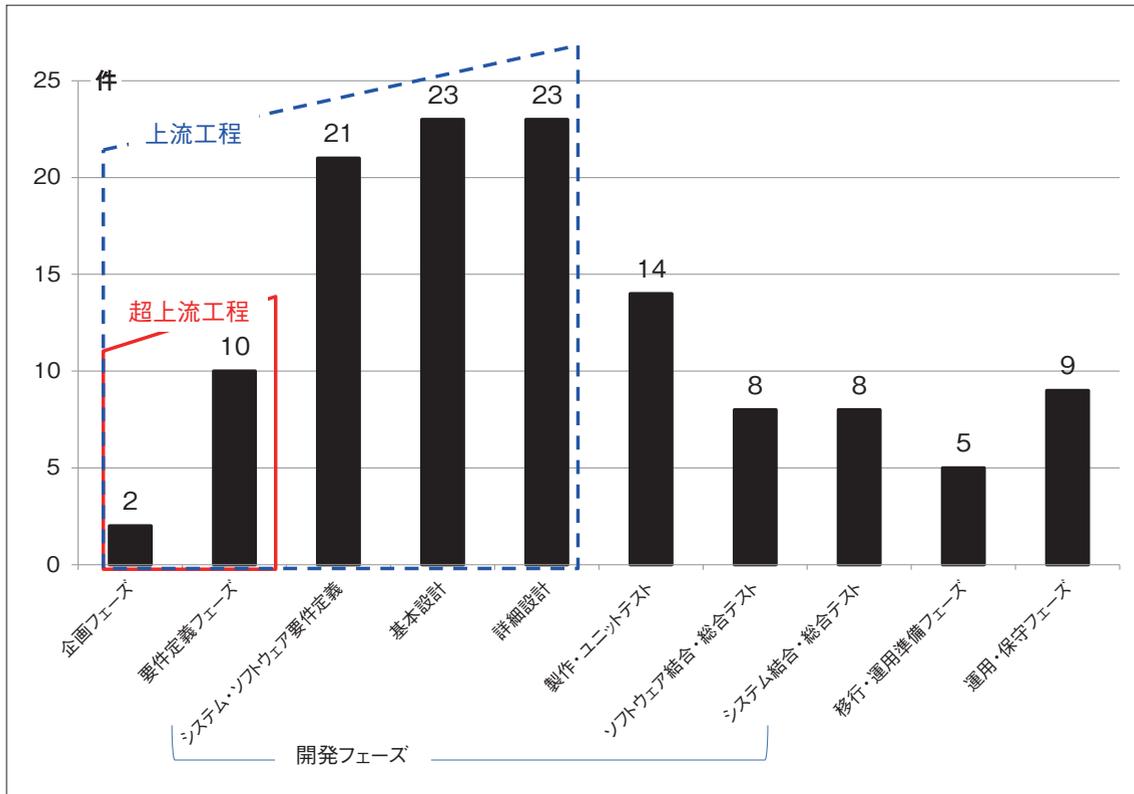


図 3-9 適用工程 (設計系)

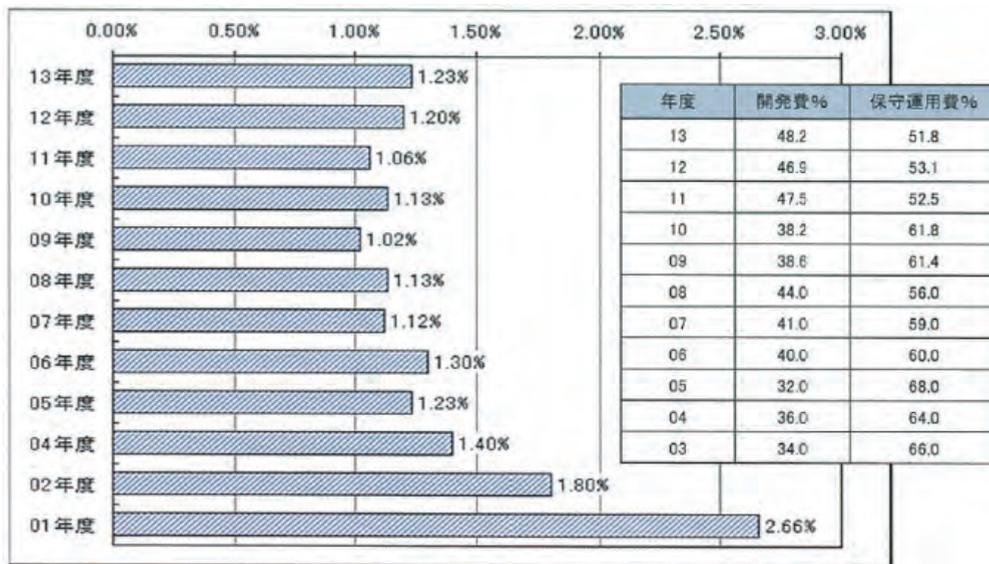


図 3-10 IT 投資費用対売上高比と保守運用割合<sup>13</sup>

<sup>13</sup> 企業 IT 動向調査報告書 2014 一般社団法人 日本情報システム・ユーザー協会 (JUAS) より引用

### 3.2.1.2 検証系

#### 3.2.1.2.1 適用結果

全事例の58件の内、検証系は25件であった。対象となる適用事例の適用結果は図3-11のような発生数になった。

また、表3-18のように、課題設定数に対して効果発揮した割合は129%で、課題設定以上に多くの効果を発揮できたことを示す。課題克服率は、96%と成功事例の集まりでこの事例集は読者には有益な情報になることを示す。副次効果発生率も33%（26）であった。このことは、新しい取組みをすることで想定外の効果を誘発できたことを示すもので、この結果が積極的に新しい取組みをするきっかけになるだろう。今後への取組みも、さらなる取組と新規の取組で49個（17個 + 32個）もあり、ソフトウェア開発の最適化に向けて、好循環を生み出すきっかけになっていることを示す。上記の結果は、表3-2にある適用結果の課題設定数に対する発生率全適用事例の結果と、新規取組設定の割合が2倍ほど多く突出している。これは、設計系の傾向とも異なっている。

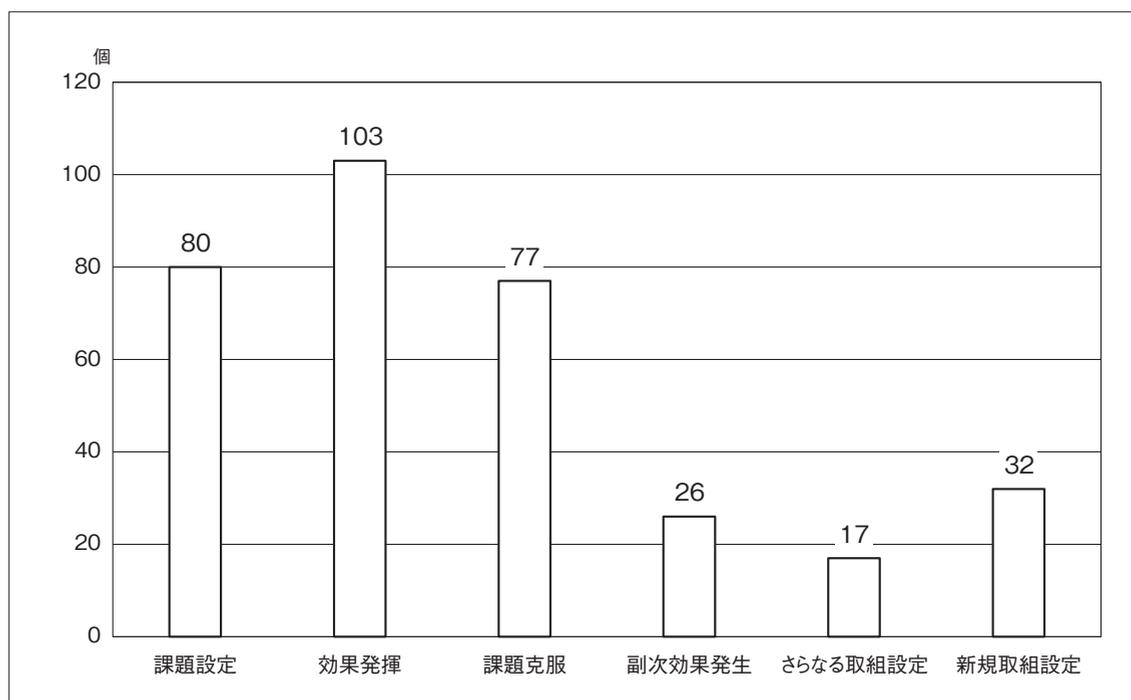


図 3-11 適用領域①の検証系の適用結果

表 3-18 適用結果の課題設定数に対する発生率（検証系）

検証系	事例数	課題設定	効果発揮	課題克服	副次効果発生	さらなる取組設定	新規取組設定
発生数	25	80	103	77	26	17	32
割合（率）	—	—	129%	96%	33%	21%	40%
割合（率）の算出方法	—	—	(効果発揮数 ÷ 課題設定数)	(課題克服数 ÷ 課題設定数)	(副次効果発生数 ÷ 課題設定数)	(さらなる取組設定数 ÷ 課題設定数)	(新規取組設定数 ÷ 課題設定数)

3.2.1.2.2 適用結果ごとの状況（テーマ別）

〔1〕対象適用事例から見る「課題」の状況

課題設定の60個の内訳は、表3-19のように品質モデルの品質特性の課題設定数47個と品質を除く33個である。

一事例当たりの課題設定数は、2.4個（算出式：60個 ÷ 25件）である。内訳として、品質モデルの品質特性では、1.9個（算出式：47個 ÷ 25件）である。品質以外では、1.3個（算出式：33個 ÷ 25件）である。

また、課題設定率のテーマ別上位は、品質（92%）、コスト（28%）、生産性（28%）、アシュアランス（20%）、納期（20%）であった。品質モデルの品質特性では、機能適合性（60%）、保守性（32%）が突出している。

表 3-19 課題のテーマの内訳と一事例当たりの課題設定率（検証系）

課題のテーマ:課題設定数(品質を除いて33個)。一事例当たり1.3個

	品質	コスト	生産性 (対応時間短縮)	アシュアランス(保証)	納期	普及促進	体制 (強化・再構築)	障害原因の 分析	人材育成 意識改革	プロジェクト マネジメント	見積支援	グローバル 展開
対象事例数	23	7	7	5	5	3	3	2	1	0	0	0
課題設定率	92%	28%	28%	20%	20%	12%	12%	8%	4%	0%	0%	0%

品質モデルの品質特性:品質特性の課題設定数(47個)。一事例当たり1.9個

	機能適合性	保守性	信頼性	セキュリティ	性能効率性	移植性	有効性	満足性	使用性	互換性	リスク回避性	効率性	利用状況網 羅性
対象事例数	15	8	4	4	3	3	3	3	2	1	1	0	0
課題設定率	60%	32%	16%	16%	12%	12%	12%	12%	8%	4%	4%	0%	0%

〔2〕対象適用事例から見る「効果」の状況

効果発揮の103個の内訳は、表3-20のように品質モデルの品質特性の課題設定数55個と品質を除く48個である。

一事例当たりの課題設定数は、4.1個である。内訳として、品質モデルの品質特性では、2.2個（算出式：55個 ÷ 25件）である。品質以外では、1.9個（算出式：48個 ÷ 25件）である。

また、効果発揮率のテーマ別上位は、品質（96%）、生産性（48%）、コスト（44%）であった。品質モデルの品質特性では、機能適合性（72%）、信頼性（28%）、保守性（28%）が突出している。

表 3-20 効果のテーマの内訳と一事例当たりの効果発揮率（検証系）

効果のテーマ:効果発揮数(品質を除いて48個)。一事例当たり1.9個

	品質	生産性 (対応時間短縮)	コスト	納期	アシュアランス(保証)	人材育成 意識改革	体制 (強化・再構築)	普及促進	障害原因の 分析	見積支援	プロジェクト マネジメント	グローバル 展開
対象事例数	24	12	11	6	5	4	4	3	2	1	0	0
効果発揮率	96%	48%	44%	24%	20%	16%	16%	12%	8%	4%	0%	0%

品質モデルの品質特性:品質特性の効果発揮数(55個)。一事例当たり2.2個

	機能適合性	信頼性	保守性	セキュリティ	有効性	満足性	性能効率性	使用性	移植性	リスク回避性	互換性	効率性	利用状況網 羅性
対象事例数	18	7	7	4	4	4	3	3	3	2	0	0	0
効果発揮率	72%	28%	28%	16%	16%	16%	12%	12%	12%	8%	0%	0%	0%

〔3〕対象適用事例から見る「副次効果」の状況

(ア) 対象事例に対する副次効果発揮状況

25 件の事例に対して 12 件の事例で副次効果が発生している。発揮率は 48% である。(算出式：副次効果発揮事例数 ÷ 事例数 = 12 件 ÷ 25 件)

これは、ひとつの課題に取り組むと、当初想定していない別の課題の解決につながるという点で、先進的な開発技術への取組みは、有益であることを意味する。

(イ) 事例／テーマ別の副次効果発揮状況

副次効果発揮の 26 個の内訳は、表 3-21 のように品質モデルの品質特性の課題設定数 10 個と品質を除く 16 個である。

一事例当たりの課題設定数は、1.0 個 (26 個 ÷ 25 件) である。内訳として、品質モデルの品質特性のみでは、0.4 個 (算出式：10 個 ÷ 25 件) である。品質以外では、0.6 個 (算出式：16 個 ÷ 25 件) である。

また、副次効果発揮率のテーマ別上位は、品質 (24%)、コスト (21%)、生産性 (20%) である品質モデルの品質特性では、機能適合性 (12%)、信頼性 (12%) が突出している。

(ウ) 課題設定数に対する副次効果発揮状況

80 個の課題設定数に対して副次効果設定数は 26 個で 33% (算出式：26 個 ÷ 80 個) の発生率である。

表 3-21 副次効果のテーマの内訳と全事例当たりの副次効果発揮率 (検証系)

副次効果のテーマ:副次効果発揮数(品質を除いて16個)。一事例当たり0.6個

	品質	コスト	生産性 (対応時間 短縮)	人材育成 意識改革	納期	見積支援	体制 (強化・再構 築)	アシュアラン ス(保証)	障害原因の 分析	プロジェクト マネジメント	普及促進	グローバル 展開
対象事例数	6	5	5	3	1	1	1	0	0	0	0	0
副次効果発揮率	24%	20%	20%	12%	4%	4%	4%	0%	0%	0%	0%	0%

品質モデルの品質特性:品質特性の副次効果発揮数(10個)。一事例当たり0.4個

	機能適合性	信頼性	使用性	有効性	満足性	リスク回避性	性能効率性	互換性	セキュリティ	保守性	移植性	効率性	利用状況網 羅性
対象事例数	3	3	1	1	1	1	0	0	0	0	0	0	0
副次効果発揮率	12%	12%	4%	4%	4%	4%	0%	0%	0%	0%	0%	0%	0%

[4] 対象適用事例から見る「さらなる取組」の状況

さらなる取組の17個の内訳は、表3-22のように品質モデルの品質特性の課題設定数9個と品質を除く8個である。

一事例当たりの課題設定数は、0.7個（算出式：17個 ÷ 25件）である。品質モデルの品質特性では、0.4個（算出式：9個 ÷ 25件）である。品質以外では、0.3個（算出式：8個 ÷ 25件）である。

また、さらなる取組設定率のテーマ別上位は、品質（28%）、コスト（12%）、普及促進（12%）である。品質モデルの品質特性では、機能適合性（32%）が突出している。

表 3-22 さらなる取組設定のテーマの内訳と一事例当たりのさらなる取組設定率（検証系）

さらなる取組設定のテーマ：さらなる取組設定数（品質を除いて8個）。一事例当たり0.3個

	品質	コスト	普及促進	生産性 (対応時間 短縮)	人材育成 意識改革	アシュアランス (保証)	障害原因の 分析	納期	プロジェクト マネジメント	見積支援	体制 (強化・再構 築)	グローバル 展開
対象事例数	7	3	3	1	1	0	0	0	0	0	0	0
さらなる取組 設定率	28%	12%	12%	4%	4%	0%	0%	0%	0%	0%	0%	0%

品質モデルの品質特性：品質特性のさらなる取組設定数（9個）。一事例当たり0.4個

	機能適合性	信頼性	性能効率性	互換性	使用性	セキュリティ	保守性	移植性	有効性	効率性	満足性	リスク回避性	利用状況網 羅性
対象事例数	8	1	0	0	0	0	0	0	0	0	0	0	0
さらなる取組 設定率	32%	4%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%

[5] 対象適用事例から見る「新規取組」の状況

新規取組の32個の内訳は、表3-23のように品質モデルの品質特性の課題設定数7個と品質を除く25個である。

一事例当たりの課題設定数は、1.3個（32個 ÷ 25件）である。内訳として、品質モデルの品質特性では、0.3個（算出式：7個 ÷ 25件）である。品質以外では、1.0個（算出式：25個 ÷ 25件）である。

また、新規取組設定率のテーマ別上位は、普及促進（32%）、プロジェクトマネジメント（28%）、品質（24%）、コスト（16%）である。品質モデルの品質特性では、機能適合性（12%）、保守性（8%）が突出している。

表 3-23 新規取組設定のテーマの内訳と一事例当たりの新規取組設定率（検証系）

新規取組設定のテーマ：新規取組設定数（品質を除いて25個）。一事例当たり1.0個

	普及促進	プロジェクト マネジメント	品質	コスト	人材育成 意識改革	体制 (強化・再構 築)	障害原因の 分析	生産性 (対応時間 短縮)	アシュアランス (保証)	納期	見積支援	グローバル 展開
対象事例数	8	7	6	4	2	2	1	1	0	0	0	0
新規取組設定率	32%	28%	24%	16%	8%	8%	4%	4%	0%	0%	0%	0%

品質モデルの品質特性：品質特性の新規取組設定数（7個）。一事例当たり0.3個

	機能適合性	保守性	性能効率性	移植性	互換性	使用性	信頼性	セキュリティ	有効性	効率性	満足性	リスク回避性	利用状況網 羅性
対象事例数	3	2	1	1	0	0	0	0	0	0	0	0	0
新規取組設定率	12%	8%	4%	4%	0%	0%	0%	0%	0%	0%	0%	0%	0%

### 3.2.1.2.3 適用工程

検証系の適用事例の適用工程は表 3-24 のとおりである。適用している工程が上流（企画フェーズから詳細設計）のものは、36%（9 件 ÷ 25 件）である。さらに、超上流（企画フェーズから要件定義フェーズ）のものは、8%（2 件 ÷ 25 件）である（図 3-12）。

また、移行・運用準備フェーズと運用・保守フェーズに適用している事例は 16%（4 件 ÷ 25 件）である。

表 3-24 適用事例ごとの適用工程分布（検証系）

事例参照番号	適用工程 共通フレーム2013									
	企画フェーズ	要件定義フェーズ	システム・ソフトウェア要件定義	基本設計	詳細設計	製作・ユニットテスト	ソフトウェア結合・総合テスト	システム結合・総合テスト	移行・運用準備フェーズ	運用・保守フェーズ
B-1			■	■			■	■		
B-2										
B-3									■	■
B-4									■	■
B-5						■				
B-6								■		
B-7					■	■	■			
B-8							■			
B-9						■	■	■		
B-10							■	■		
B-11						■	■			
15-B-1	■	■					■	■	■	
15-B-2						■				
15-B-3			■	■	■	■	■	■		
15-B-4							■	■		
15-B-5					■	■	■	■		
15-B-6				■	■	■	■	■		
15-B-7							■			
15-B-8	■									
15-B-9						■				
15-B-10				■	■	■				
15-B-11								■		
15-B-12						■	■	■	■	■
15-B-13						■				
15-B-14				■	■	■	■	■		

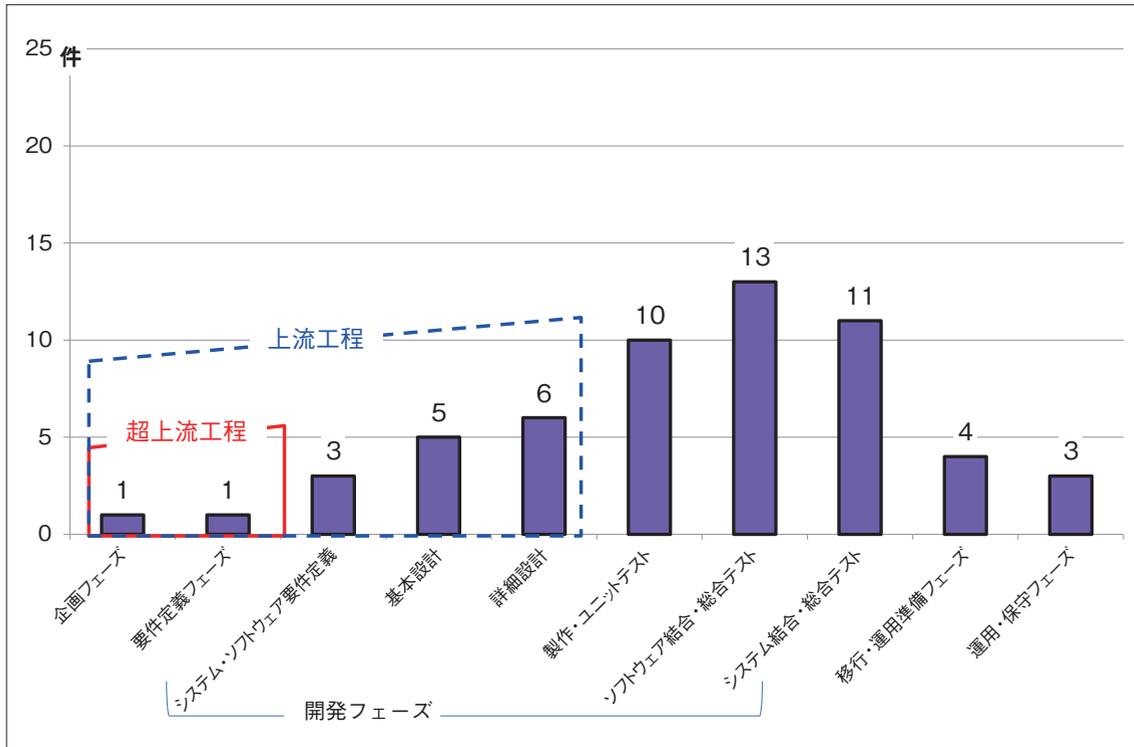


図 3-12 適用工程 (検証系)

### 3.2.1.3 適用領域①（設計系、検証系）で分かったこと

#### 〔1〕対象適用事例から見る「課題」の状況

一事例当たりの課題設定数は、設計系（6.3個）の方が検証系（3.2個）よりも2.0倍多く、設計系の方が多くの課題に直面していることが分かる。これは、設計系の方が検証系よりも対象範囲が広く、複雑化している可能性がある。

課題で上位を占めているテーマは、どちらも同じである。しかし、「コスト」と「生産性」の対象事例に占める割合は、設計系（58%）の方が検証系（28%）のそれよりも高くなっている。また、品質関連では、「機能適合性」が上位であることは同じであるが、「保守性」に関しては、設計系（58%）の方が検証系（32%）のそれよりも高くなっている。

#### 〔2〕対象適用事例から見る「効果」の状況

一事例当たりの効果発揮数も、課題数と同様に設計系（7.8個）の方が、検証系（4.1個）よりも1.9倍多い。

効果で上位を占めているテーマは、どちらもほぼ同じであるが、設計系の「人材育成・意識改革」(55%)は、検証系（16%）と比べて突出している。また、品質関連では、「機能適合性」がトップで、同じであるが、「保守性」に関しては、設計系（67%）の方が検証系（28%）よりもかなり高かった。検証系の他の上位は、「信頼性」（28%）である。

#### 〔3〕対象適用事例から見る「副次効果」の状況

一事例当たりの副次効果発揮数は、課題数と同様に設計系（1.8個）の方が、検証系（1.0個）よりも1.7倍多い。

副次効果で上位を占めているテーマは、「品質」、「コスト」、「生産性」でほぼ同じであるが、設計系の「人材育成・意識改革」（27%）は、検証系（12%）よりも多い。また、品質関連では、「機能適合性」がトップで同じである。それ以外の品質特性では、設計系と検証系で差異はない。

#### 〔4〕対象適用事例から見る「さらなる取組」の状況

一事例当たりのさらなる取組数は、設計系（0.8個）と検証系（0.7個）でほぼ同じである。

さらなる取組で上位を占めているテーマは、「品質」、「コスト」が共通しているが、設計系の「生産性」（15%）、「体制」（9%）は、検証系では、それぞれ4%、0%であり、設計系の特徴のようである。

#### 〔5〕対象適用事例から見る「新規取組」の状況

一事例当たりの新規取組数は、検証系（1.3個）の方が、設計系（0.9個）よりも1.5倍多い。

新規取組で上位を占めているテーマは、「普及促進」、「品質」が共通しているが、検証系の「プロジェクトマネジメント」（28%）は、設計系では9%であり、検証系の特徴のようである。

#### 〔6〕適用工程の状況

設計系では、超上流工程からの取組みが33%（対象事例33件中11件）と多い。一方、検証系では、超上流工程からの取組みが8%（対象事例25件中2件）である。また、保守・運用工程での取組みは、設計系（30%）、検証系（16%）である。

### 3.2.2 適用領域②（エンタプライズ系、Web・フロント系、組込み・制御系）別の適用結果等状況

適用領域②のエンタプライズ系、Web・フロント系、組込み・制御系は、2.2.2.1章の適用領域の定義のとおりである。

それぞれの事例数は、エンタプライズ系が21件、Web・フロント系が12件、組込み・制御系が33件である。

課題設定数に対する効果発揮数の比率は、組込み・制御系（139%）が、エンタプライズ系（117%）、Web・フロント系（120%）よりも若干高い。

また、一事例当たりの課題設定数では、組込み・制御系（3.3件）は、エンタプライズ系（5.7件）、Web・フロント系（4.9件）よりもかなり少ない。

一事例当たりの副次効果発揮数では、組込み・制御系（1.5件）は、エンタプライズ系（1.1件）、Web・フロント系（1.1件）よりも多い。

一事例当たりのさらなる取組設定数では、Web・フロント系（1.1件）は、エンタプライズ系（0.7件）、組込み・制御系（0.5件）よりも多い。

一事例当たりの新規取組設定数では、エンタプライズ系（1.2件）は、Web・フロント系（0.9件）、組込み・制御系（0.8件）よりも若干多い。

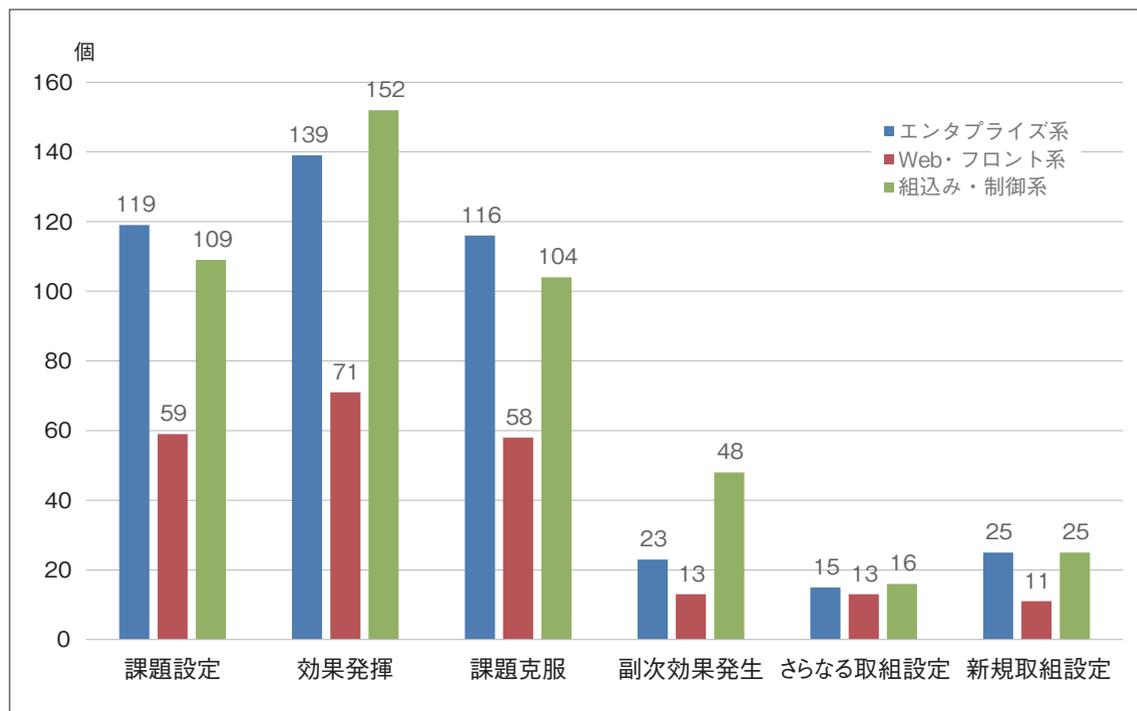


図 3-13 適用領域②別の適用結果

表 3-25 適用結果の比較

		事例数	課題設定	効果発揮	課題克服	副次効果発生	さらなる取組設定	新規取組設定
エンタプライズ系	個数	21	119	139	116	23	15	25
	一事例当たりの発生数	—	5.7	6.6	5.5	1.1	0.7	1.2
	割合 (課題設定数比)	—	—	117%	97%	19%	13%	21%
Web・フロント系	個数	12	59	71	58	13	13	11
	一事例当たりの発生数	—	4.9	5.9	4.8	1.1	1.1	0.9
	割合 (課題設定数比)	—	—	120%	98%	22%	22%	19%
組込み・制御系	個数	33	109	152	104	48	16	25
	一事例当たりの発生数	—	3.3	4.6	3.2	1.5	0.5	0.8
	割合 (課題設定数比)	—	—	139%	95%	44%	15%	23%

### 3.2.2.1 エンタプライズ系

#### 3.2.2.1.1 適用結果

適用領域②の分類では、全事例数 58 件の内、エンタプライズ系は 21 件である。それについて整理したものが以下のとおりである。

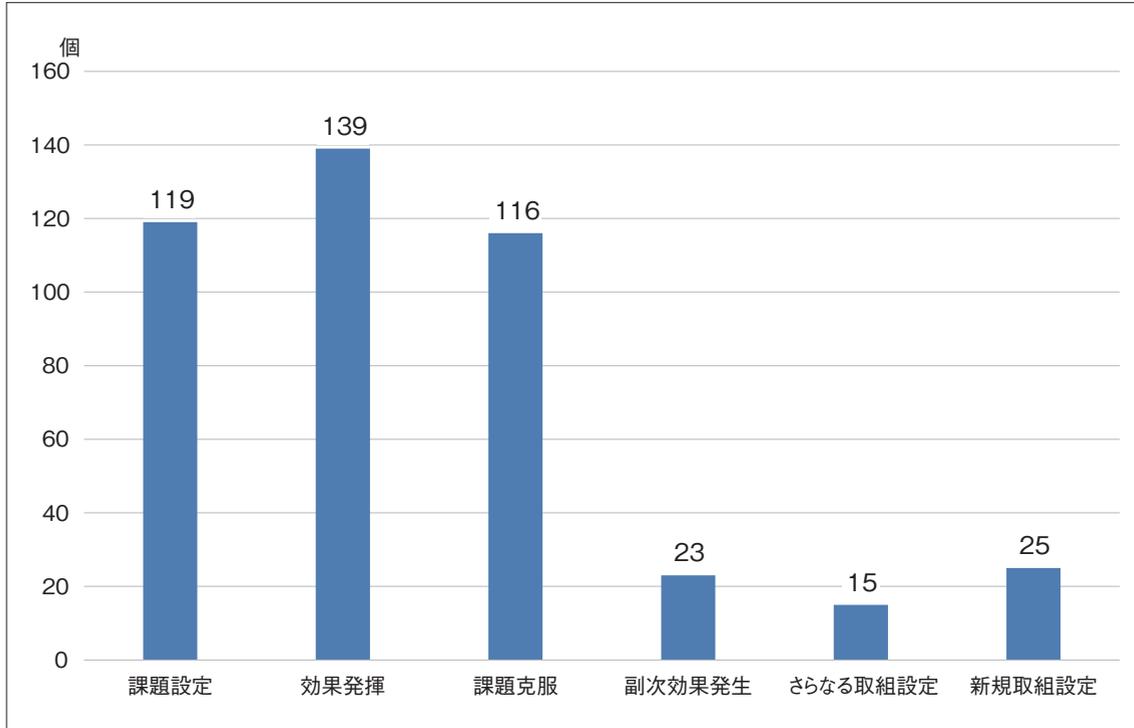


図 3-14 適用領域②のエンタプライズ系の適用結果

表 3-26 適用結果の課題設定数に対する発生率（エンタプライズ系）

	事例数	課題設定	効果発揮	課題克服	副次効果発生	さらなる取組設定	新規取組設定
個数	21	119	139	116	23	15	25
割合 (課題設定数比)	—	—	117%	97%	19%	13%	21%
割合(率)の算出方法	—	—	(効果発揮数÷課題設定数)	(課題克服数÷課題設定数)	(副次効果発生数÷課題設定数)	(さらなる取組設定数÷課題設定数)	(新規取組設定数÷課題設定数)

### 3.2.2.1.2 適用結果ごとの状況（テーマ別）

#### 〔1〕対象適用事例から見る「課題」の状況

課題設定の119個の内訳は、表3-27のように品質モデルの品質特性の課題設定数69個と品質を除く50個である。

一事例当たりの課題設定数は、5.7個（119個 ÷ 21件）である。品質モデルの品質特性では、3.3個（算出式：69個 ÷ 21件）である。品質以外では、2.4個（算出式：50個 ÷ 21件）である。

また、課題設定率のテーマ別上位は、品質（95%）、コスト（57%）、生産性（43%）、納期（33%）、人材育成・意識改革（24%）であった。品質モデルの品質特性では、機能適合性（57%）、保守性（43%）、満足性（33%）が突出している。

表3-27 課題のテーマの内訳と一事例当たりの課題設定率（エンタプライズ系）

課題のテーマ:課題設定数(品質を除いて50個)。一事例当たり2.4個

	品質	コスト	生産性 (対応時間 短縮)	納期	人材育成 意識改革	アシュアランス (保証)	普及促進	体制 (強化・再構築)	プロジェクト マネジメント	障害原因の 分析	見積支援	グローバル 展開
対象事例数	20	12	9	7	5	4	4	4	3	2	0	0
課題設定率	95%	57%	43%	33%	24%	19%	19%	19%	14%	10%	0%	0%

品質モデルの品質特性:品質特性の課題設定数(69個)。一事例当たり3.3個

	機能適合性	保守性	満足性	性能効率性	使用性	信頼性	移植性	互換性	有効性	効率性	セキュリティ	利用状況網 羅性	リスク回避性
対象事例数	12	9	7	6	5	5	5	4	4	4	3	3	2
課題設定率	57%	43%	33%	29%	24%	24%	24%	19%	19%	19%	14%	14%	10%

#### 〔2〕対象適用事例から見る「効果」の状況

効果発揮の139個の内訳は、表3-28のように品質モデルの品質特性の課題設定数73個と品質を除く66個である。

一事例当たりの課題設定数は、6.6個（算出式：139個 ÷ 21件）である。品質モデルの品質特性では、3.5個（算出式：73個 ÷ 21件）である。品質以外では、3.1個（算出式：66個 ÷ 21件）である。

また、効果発揮率のテーマ別上位は、品質（100%）、コスト（76%）、生産性（52%）、納期（43%）、人材育成・意識改革（38%）であった。品質モデルの品質特性では、機能適合性（71%）、保守性（48%）が突出している。

表3-28 効果のテーマの内訳と一事例当たりの効果発揮率（エンタプライズ系）

効果のテーマ:効果発揮数(品質を除いて66個)。一事例当たり3.1個

	品質	コスト	生産性 (対応時間 短縮)	納期	人材育成 意識改革	体制 (強化・再構築)	アシュアランス (保証)	普及促進	プロジェクト マネジメント	障害原因の 分析	見積支援	グローバル 展開
対象事例数	21	16	11	9	8	6	4	4	3	2	2	1
効果発揮率	100%	76%	52%	43%	38%	29%	19%	19%	14%	10%	10%	5%

品質モデルの品質特性:品質特性の効果発揮数(73個)。一事例当たり3.5個

	機能適合性	保守性	信頼性	満足性	性能効率性	使用性	移植性	互換性	有効性	効率性	セキュリティ	リスク回避性	利用状況網 羅性
対象事例数	15	10	6	6	5	5	5	4	4	4	3	3	3
効果発揮率	71%	48%	29%	29%	24%	24%	24%	19%	19%	19%	14%	14%	14%

[3] 対象適用事例から見る「副次効果」の状況

(ア) 対象事例に対する副次効果発揮状況

21 件の事例に対して 11 件の事例で副次効果が発生している。発揮率は 52% である。(算出式：副次効果発揮事例数 ÷ 事例数 = 11 件 ÷ 21 件)

これは、ひとつの課題に取り組むと、当初想定していない別の課題の解決につながるという点で、先進的な開発技術への取組みは、有益であることを意味する。

(イ) 事例／テーマ別の副次効果発揮状況

副次効果発揮の 23 個の内訳は、表 3-29 のように品質モデルの品質特性の課題設定数 6 個と品質を除く 17 個である。

一事例当たりの課題設定数は、1.1 個 (算出式:23 個 ÷ 21 件) である。品質モデルの品質特性では、0.3 個 (算出式：6 個 ÷ 21 件) である。品質以外では 0.8 個 (算出式：17 個 ÷ 21 件) である。

また、副次効果発揮率のテーマ別上位は、品質 (24%)、コスト (19%)、人材育成・意識改革 (14%) である。品質モデルの品質特性では、機能適合性 (14%) が突出している。

(ウ) 課題設定数に対する副次効果発揮状況

119 個の課題設定数に対して副次効果設定数は 23 個で 19% (算出式:23 個 ÷ 119 個) の発生率である。

表 3-29 副次効果のテーマの内訳と一事例当たりの副次効果発揮率 (エンタプライズ系)

副次効果のテーマ:副次効果発揮数(品質を除いて17個)。一事例当たり0.8個

	品質	コスト	人材育成 意識改革	納期	生産性 (対応時間 短縮)	見積支援	体制 (強化・再構 築)	プロジェクト マネジメント	グローバル 展開	アシュアラン ス(保証)	障害原因の 分析	普及促進
対象事例数	5	4	3	2	2	2	2	1	1	0	0	0
副次効果発揮率	24%	19%	14%	10%	10%	10%	10%	5%	5%	0%	0%	0%

品質モデルの品質特性:品質特性の副次効果発揮数(6個)。一事例当たり0.3個

	機能適合性	信頼性	保守性	リスク回避性	性能効率性	互換性	使用性	セキュリティ	移植性	有効性	効率性	満足性	利用状況網 羅性
対象事例数	3	1	1	1	0	0	0	0	0	0	0	0	0
副次効果発揮率	14%	5%	5%	5%	0%	0%	0%	0%	0%	0%	0%	0%	0%

#### [4] 対象適用事例から見る「さらなる取組」の状況

さらなる取組の15個の内訳は、表3-30のように品質モデルの品質特性の課題設定数3個と品質を除く12個である。

一事例当たりの課題設定数は、0.7個（算出式：15個 ÷ 21件）である。品質モデルの品質特性では、0.1個（算出式：3個 ÷ 21件）である。品質以外では、0.6個（算出式：12個 ÷ 21件）である。

また、さらなる取組設定率のテーマ別上位は、コスト（19%）、品質（14%）、体制（14%）である。品質モデルの品質特性では、機能適合性（14%）が突出している。

表3-30 さらなる取組設定のテーマの内訳と一事例当たりのさらなる取組設定率（エンタプライズ系）

さらなる取組設定のテーマ：さらなる取組設定数（品質を除いて12個）。一事例当たり0.6個

	コスト	品質	体制 (強化・再構築)	生産性 (対応時間 短縮)	普及促進	人材育成 意識改革	アシュアランス (保証)	障害原因の 分析	納期	プロジェクト マネジメント	見積支援	グローバル 展開
対象事例数	4	3	3	2	2	1	0	0	0	0	0	0
さらなる取組 設定率	19%	14%	14%	10%	10%	5%	0%	0%	0%	0%	0%	0%

品質モデルの品質特性：品質特性のさらなる取組設定数（3個）。一事例当たり0.1個

	機能適合性	性能効率性	互換性	使用性	信頼性	セキュリティ	保守性	移植性	有効性	効率性	満足性	リスク回避性	利用状況網 羅性
対象事例数	3	0	0	0	0	0	0	0	0	0	0	0	0
さらなる取組 設定率	14%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%

#### [5] 対象適用事例から見る「新規取組」の状況

新規取組の25個の内訳は、表3-31のように品質モデルの品質特性の課題設定数3個と品質を除く22個である。

一事例当たりの課題設定数は、1.2個（算出式：25個 ÷ 21件）である。品質モデルの品質特性では、0.1個（算出式：3個 ÷ 21件）である。品質以外では、1.0個（算出式：22個 ÷ 21件）である。

また、新規取組設定率のテーマ別上位は、プロジェクトマネジメント（33%）、普及促進（24%）、人材育成・意識改革（19%）、品質（14%）、生産性（14%）である。品質モデルの品質特性では、互換性（5%）、保守性（5%）、移植性（5%）である。

表3-31 新規取組設定のテーマの内訳と一事例当たりの新規取組設定率（エンタプライズ系）

新規取組設定のテーマ：新規取組設定数（品質を除いて22個）。一事例当たり1.0個

	プロジェクト マネジメント	普及促進	人材育成 意識改革	品質	生産性 (対応時間 短縮)	体制 (強化・再構築)	障害原因の 分析	アシュアランス (保証)	コスト	納期	見積支援	グローバル 展開
対象事例数	7	5	4	3	3	2	1	0	0	0	0	0
新規取組設定率	33%	24%	19%	14%	14%	10%	5%	0%	0%	0%	0%	0%

品質モデルの品質特性：品質特性の新規取組設定数（3個）。一事例当たり0.1個

	互換性	保守性	移植性	機能適合性	性能効率性	使用性	信頼性	セキュリティ	有効性	効率性	満足性	リスク回避性	利用状況網 羅性
対象事例数	1	1	1	0	0	0	0	0	0	0	0	0	0
新規取組設定率	5%	5%	5%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%

3.2.2.1.3 適用工程

エンタプライズ系では、12件の設計系と9件の検証系の21件の適用事例が対象となった。超上流工程に関わる適用事例は10件、保守・運用工程に関わるそれは6件である。超上流工程と保守・運用工程が、ソフトウェア開発では重要視されていると考えられる。

表 3-32 適用事例ごとの適用工程分布 (エンタプライズ系)

事例 参照 番号	適用 領域 ②		適用工程 共通フレーム2013										
	エン タ プ ラ イ ズ 系	W e b ・ フ ロ ン ト 系	組 込 み ・ 制 御 系	企 画 フ ェ ー ズ	要 件 定 義 フ ェ ー ズ	シ ス テ ム ・ ソ フ ト ウ ェ ア 要 件 定 義	基 本 設 計	詳 細 設 計	製 作 ・ ユ ニ ツ ト テ ス ト	ソ フ ト ウ ェ ア 結 合 ・ 総 合 テ ス ト	シ ス テ ム 結 合 ・ 総 合 テ ス ト	移 行 ・ 運 用 準 備 フ ェ ー ズ	運 用 ・ 保 守 フ ェ ー ズ
A-5	■				■	■							
A-6	■					■	■	■	■	■	■	■	■
A-7	■					■	■	■					■
15-A-1	■				■	■	■	■	■				
15-A-2	■				■	■	■	■	■	■			
15-A-3	■	■			■	■			■				■
15-A-4	■			■	■	■							
15-A-5	■				■	■	■	■	■	■	■	■	■
15-A-6	■				■	■	■	■					
15-A-7	■					■	■						
15-A-8	■	■		■		■	■	■					
15-A-9	■	■	■		■	■	■	■				■	■
B-3	■											■	■
15-B-1	■				■	■				■	■	■	■
15-B-2	■							■					
15-B-3	■					■	■	■	■	■			
15-B-4	■							■	■	■			
15-B-5	■							■	■	■			
15-B-6	■							■	■	■			
15-B-7	■		■							■			
15-B-8	■	■	■	■									

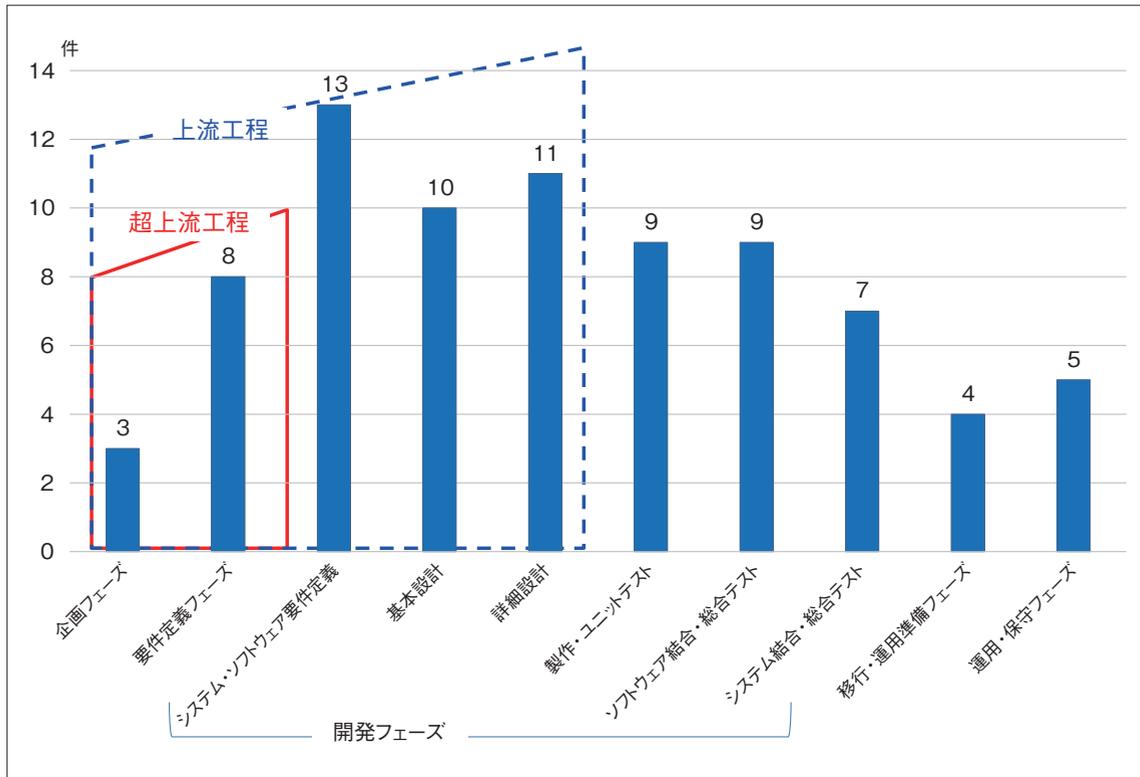


図 3-15 適用工程 (エンタプライズ系)

### 3.2.2.2 Web・フロント系

#### 3.2.2.2.1 適用結果

適用領域②の分類では、全事例数 58 件の内、Web・フロント系は 12 件である。それについて整理したものが以下のとおりである。

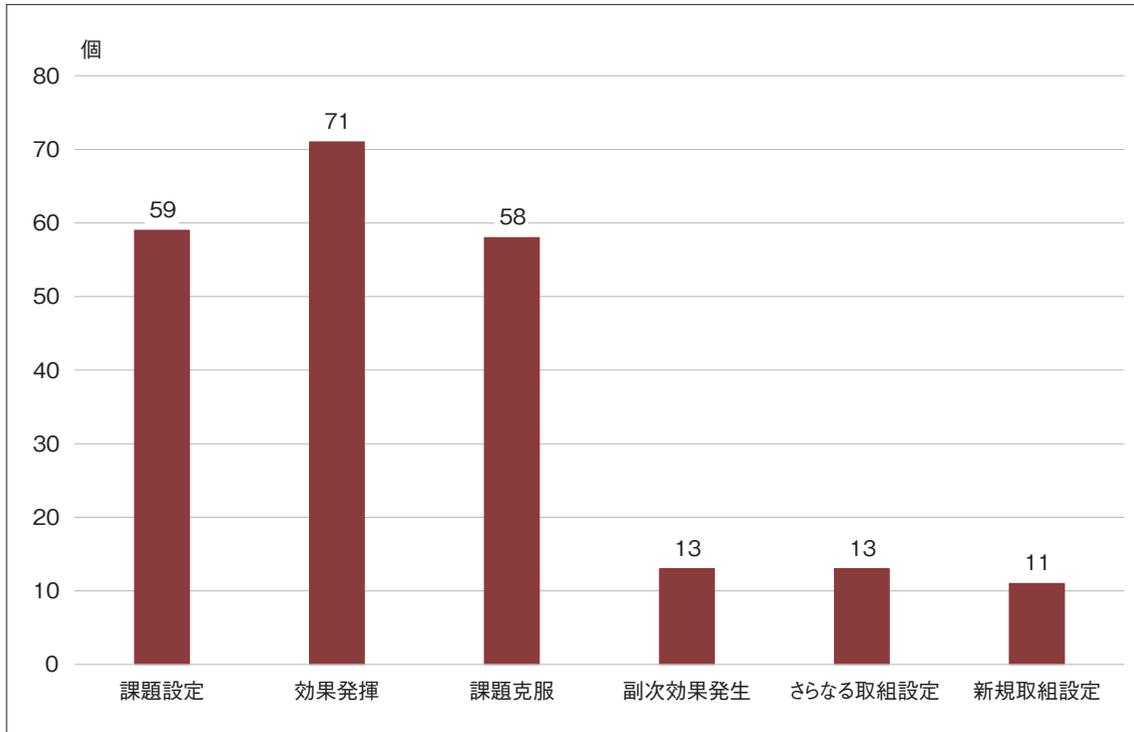


図 3-16 適用領域②の Web・フロント系の適用結果

表 3-33 適用結果の課題設定数に対する発生率 (Web・フロント系)

	事例数	課題設定	効果発揮	課題克服	副次効果発生	さらなる取組設定	新規取組設定
個数	12	59	71	58	13	13	11
割合 (課題設定数比)	—	—	120%	98%	22%	22%	19%
割合(率)の算出方法	—	—	(効果発揮数÷課題設定数)	(課題克服数÷課題設定数)	(副次効果発生数÷課題設定数)	(さらなる取組設定数÷課題設定数)	(新規取組設定数÷課題設定数)

### 3.2.2.2.2 適用結果ごとの状況（テーマ別）

#### 〔1〕対象適用事例から見る「課題」の状況

課題設定の59個の内訳は、表3-34のように品質モデルの品質特性の課題設定数35個と品質を除く24個である。

一事例当たりの課題設定数は、4.9個（算出式：59個 ÷ 12件）である。品質モデルの品質特性では、2.9個（算出式：35個 ÷ 12件）である。品質以外では、2.0個（算出式：24個 ÷ 12件）である。

また、課題設定率のテーマ別上位は、品質（92%）、コスト（42%）、生産性（42%）、納期（33%）、人材育成・意識改革（25%）であった。品質モデルの品質特性では、保守性（50%）、機能適合性（33%）、セキュリティ（33%）が突出している。

表 3-34 課題のテーマの内訳と一事例当たりの課題設定率（Web・フロント系）

課題のテーマ:課題設定数(品質を除いて24個)。一事例当たり2.0個

	品質	コスト	生産性 (対応時間 短縮)	納期	人材育成 意識改革	プロジェクト マネジメント	普及促進	アシュアランス (保証)	障害原因の 分析	体制 (強化・再構 築)	見積支援	グローバル 展開
対象事例数	11	5	5	4	3	2	2	1	1	1	0	0
課題設定率	92%	42%	42%	33%	25%	17%	17%	8%	8%	8%	0%	0%

品質モデルの品質特性:品質特性の課題設定数(35個)。一事例当たり2.9個

	保守性	機能適合性	セキュリティ	性能効率性	使用性	移植性	満足性	有効性	効率性	リスク回避性	互換性	信頼性	利用状況網 羅性
対象事例数	6	4	4	3	3	3	3	2	2	2	1	1	1
課題設定率	50%	33%	33%	25%	25%	25%	25%	17%	17%	17%	8%	8%	8%

#### 〔2〕対象適用事例から見る「効果」の状況

効果発揮の71個の内訳は、表3-35のように品質モデルの品質特性の課題設定数36個と品質を除く35個である。

一事例当たりの課題設定数は、5.9個（算出式：71個 ÷ 12件）である。品質モデルの品質特性では、3.0個（算出式：36個 ÷ 12件）である。品質以外では、2.9個（算出式：35個 ÷ 12件）である。

また、効果発揮率のテーマ別上位は、品質（92%）、生産性（75%）、コスト（58%）、人材育成・意識改革（42%）、納期（33%）、プロジェクトマネジメント（25%）、普及促進（25%）であった。品質モデルの品質特性では、機能適合性（42%）、保守性（42%）、セキュリティ（33%）が突出している。

表 3-35 効果のテーマの内訳と一事例当たりの効果発揮率（Web・フロント系）

効果のテーマ:効果発揮数(品質を除いて35個)。一事例当たり2.9個

	品質	生産性 (対応時間 短縮)	コスト	人材育成 意識改革	納期	プロジェクト マネジメント	普及促進	体制 (強化・再構 築)	アシュアランス (保証)	障害原因の 分析	見積支援	グローバル 展開
対象事例数	11	9	7	5	4	3	3	2	1	1	0	0
効果発揮率	92%	75%	58%	42%	33%	25%	25%	17%	8%	8%	0%	0%

品質モデルの品質特性:品質特性の効果発揮数(36個)。一事例当たり3.0個

	機能適合性	保守性	セキュリティ	性能効率性	使用性	移植性	満足性	信頼性	有効性	効率性	リスク回避性	互換性	利用状況網 羅性
対象事例数	5	5	4	3	3	3	3	2	2	2	2	1	1
効果発揮率	42%	42%	33%	25%	25%	25%	25%	17%	17%	17%	17%	8%	8%

[3] 対象適用事例から見る「副次効果」の状況

(ア) 対象事例に対する副次効果発揮状況

12件の事例に対して5件の事例で副次効果が発生している。発揮率は42%である。(算出式：副次効果発揮事例数 ÷ 事例数 = 5件 ÷ 12件)

これは、ひとつの課題に取り組むと、当初想定していない別の課題の解決につながるという点で、先進的な開発技術への取組みは、有益であることを意味する。

(イ) 事例／テーマ別の副次効果発揮状況

副次効果発揮の13個の内訳は、表3-36のように品質モデルの品質特性の課題設定数2個と品質を除く22個である。

一事例当たりの課題設定数は、1.1個(算出式:13個 ÷ 12件)である。品質モデルの品質特性では、0.2個(算出式:2個 ÷ 12件)である。品質以外では、0.9個(算出式:11個 ÷ 12件)である。

また、副次効果発揮率のテーマ別上位は、生産性(33%)、品質(17%)、コスト(17%)、人材育成・意識改革(17%)である。品質モデルの品質特性では、機能適合性(8%)、保守性(8%)が突出している。

(ウ) 課題設定数に対する副次効果発揮状況

59個の課題設定数に対して副次効果設定数は13個で22%(算出式:13個 ÷ 59個)の発生率である。

表 3-36 副次効果のテーマの内訳と一事例当たりの副次効果発揮率 (Web・フロント系)

副次効果のテーマ:副次効果発揮数(品質を除いて11個)。一事例当たり0.9個

	生産性 (対応時間 短縮)	品質	コスト	人材育成 意識改革	プロジェクト マネジメント	普及促進	体制 (強化・再構築)	アシュアランス (保証)	障害原因の 分析	納期	見積支援	グローバル 展開
対象事例数	4	2	2	2	1	1	1	0	0	0	0	0
副次効果発揮率	33%	17%	17%	17%	8%	8%	8%	0%	0%	0%	0%	0%

品質モデルの品質特性:品質特性の副次効果発揮数(2個)。一事例当たり0.2個

	機能適合性	信頼性	性能効率性	互換性	使用性	セキュリティ	保守性	移植性	有効性	効率性	満足性	リスク回避性	利用状況網 羅性
対象事例数	1	1	0	0	0	0	0	0	0	0	0	0	0
副次効果発揮率	8%	8%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%

#### [4] 対象適用事例から見る「さらなる取組」の状況

さらなる取組の13個の内訳は、表3-37のように品質モデルの品質特性の課題設定数6個と品質を除く7個である。

一事例当たりの課題設定数は、1.1個(13個 ÷ 12件)である。品質モデルの品質特性では、0.5個(算出式: 6個 ÷ 12件)である。品質以外では、0.6個(算出式: 7個 ÷ 12件)である。

また、さらなる取組設定率のテーマ別上位は、品質(42%)、生産性(17%)、普及促進(17%)である。品質モデルの品質特性では、機能適合性(25%)が突出している。

表 3-37 さらなる取組設定のテーマの内訳と一事例当たりのさらなる取組設定率 (Web・フロント系)

さらなる取組設定のテーマ:さらなる取組設定数(品質を除いて7個)。一事例当たり0.6個

	品質	生産性 (対応時間 短縮)	普及促進	コスト	人材育成 意識改革	プロジェクト マネジメント	アシュアラン ス(保証)	障害原因の 分析	納期	見積支援	体制 (強化・再構 築)	グローバル 展開
対象事例数	5	2	2	1	1	1	0	0	0	0	0	0
さらなる取組 設定率	42%	17%	17%	8%	8%	8%	0%	0%	0%	0%	0%	0%

品質モデルの品質特性:品質特性のさらなる取組設定数(6個)。一事例当たり0.5個

	機能適合性	使用性	信頼性	保守性	性能効率性	互換性	セキュリティ	移植性	有効性	効率性	満足性	リスク回避性	利用状況網 羅性
対象事例数	3	1	1	1	0	0	0	0	0	0	0	0	0
さらなる取組 設定率	25%	8%	8%	8%	0%	0%	0%	0%	0%	0%	0%	0%	0%

#### [5] 対象適用事例から見る「新規取組」の状況

新規取組の11個の内訳は、表3-38のように品質モデルの品質特性の課題設定数2個と品質を除く9個である。

一事例当たりの課題設定数は、0.9個(算出式:11個 ÷ 12件)である。品質モデルの品質特性では、0.2個(算出式: 2個 ÷ 12件)である。品質以外では、0.8個(算出式: 9個 ÷ 12件)である。

また、新規取組設定率のテーマ別上位は、普及促進(33%)、品質(17%)、コスト(17%)である。品質モデルの品質特性では、機能適合性(8%)、性能効率性(8%)である。

表 3-38 新規取組設定のテーマの内訳と一事例当たりの新規取組設定率 (Web・フロント系)

新規取組設定のテーマ:新規取組設定数(品質を除いて9件)。一事例当たり0.8個

	普及促進	品質	コスト	生産性 (対応時間 短縮)	人材育成 意識改革	プロジェクト マネジメント	アシュアラン ス(保証)	障害原因の 分析	納期	見積支援	体制 (強化・再構 築)	グローバル 展開
対象事例数	4	2	2	1	1	1	0	0	0	0	0	0
新規取組設定率	33%	17%	17%	8%	8%	8%	0%	0%	0%	0%	0%	0%

品質モデルの品質特性:品質特性の新規取組設定数(2個)。一事例当たり0.2個

	機能適合性	性能効率性	互換性	使用性	信頼性	セキュリティ	保守性	移植性	有効性	効率性	満足性	リスク回避性	利用状況網 羅性
対象事例数	1	1	0	0	0	0	0	0	0	0	0	0	0
新規取組設定率	8%	8%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%

3.2.2.2.3 適用工程

Web・フロント系では、6件の設計系と6件の検証系の12件の適用事例が対象となった。超上流工程に関わる適用事例は4件、保守・運用工程に関わるそれは4件である。超上流工程と保守・運用工程が、ソフトウェア開発では重要視されていると考えられる。

表 3-39 適用事例ごとの適用工程分布 (Web・フロント系)

事例 参照 番号	適用 領域 ②		適用工程 共通フレーム2013										
	Web・ フロント系	組 込み・ 制御系	エン タプ ライ ズ系	企 画 フェ ーズ	要 件 定 義 フェ ーズ	シ ス テ ム・ ソフ トウ ェア 要 件 定 義	基 本 設 計	詳 細 設 計	製 作・ ユニ ット テ ス ト	ソ フ ト ウ ェ ア 結 合・ 総 合 テ ス ト	シ ス テ ム 結 合・ 総 合 テ ス ト	移 行・ 運 用 準 備 フェ ーズ	運 用・ 保 守 フェ ーズ
A-2	■	■											■
A-4	■					■				■	■		
A-8	■							■	■				
15-A-3	■	■		■	■			■					■
15-A-8	■	■		■				■					
15-A-9	■	■	■	■	■	■	■	■					
B-4	■											■	■
15-B-8	■	■		■									
15-B-9	■	■							■				
15-B-10	■	■					■	■	■				
15-B-11	■	■									■		
15-B-12	■	■							■	■		■	■

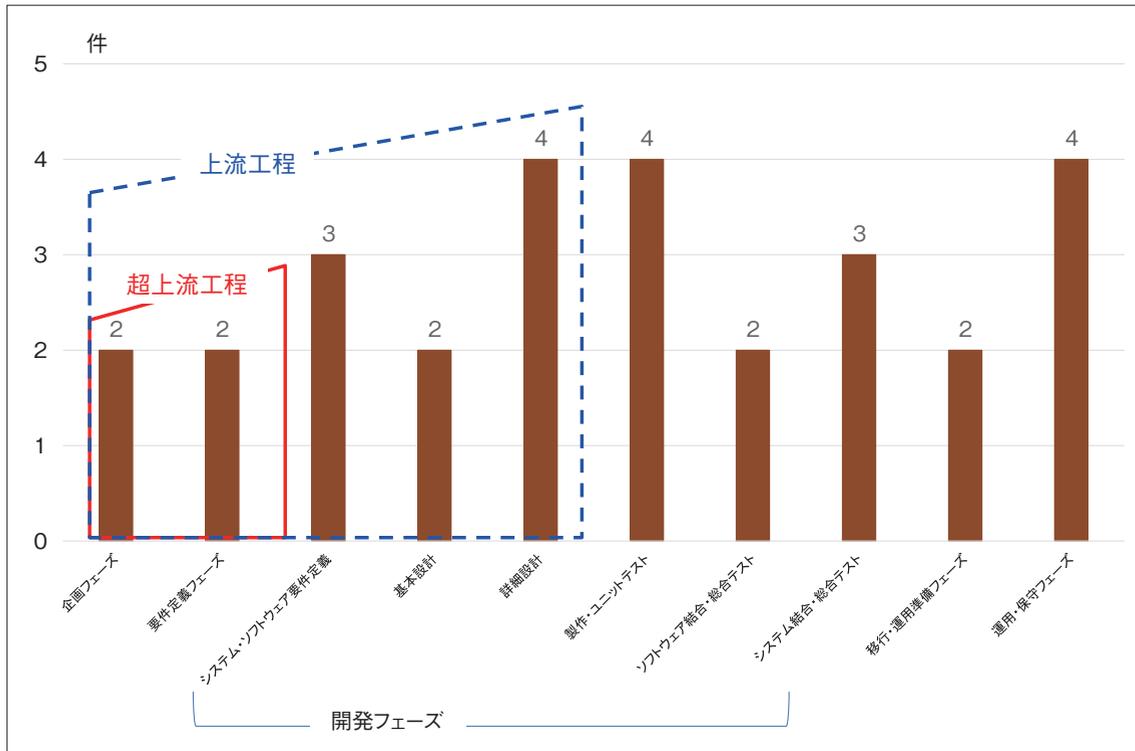


図 3-17 適用工程 (Web・フロント系)

### 3.2.2.3 組込み・制御系

#### 3.2.2.3.1 適用結果

適用領域②の分類では、全事例数 58 件の内、組込み・制御系は 33 件である。それについて整理したのが以下のとおりである。

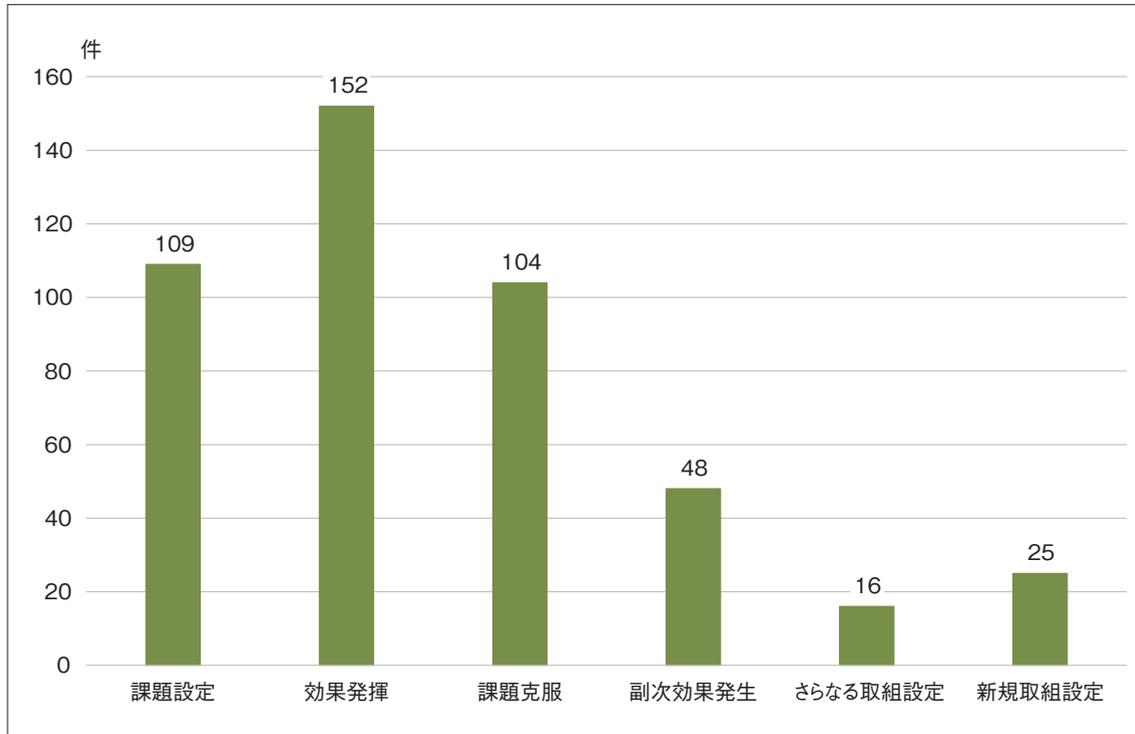


図 3-18 適用領域②の組込み・制御系の適用結果

表 3-40 適用結果の課題設定数に対する発生率（組込み・制御系）

	事例数	課題設定	効果発揮	課題克服	副次効果発生	さらなる取組設定	新規取組設定
個数	33	109	152	104	48	16	25
割合 (課題設定数比)	—	—	139%	95%	44%	15%	23%
割合(率)の算出方法	—	—	(効果発揮数÷課題設定数)	(課題克服数÷課題設定数)	(副次効果発生数÷課題設定数)	(さらなる取組設定数÷課題設定数)	(新規取組設定数÷課題設定数)

### 3.2.2.3.2 適用結果ごとの状況（テーマ別）

#### 〔1〕対象適用事例から見る「課題」の状況

課題設定の109個の内訳は、表3-41のように品質モデルの品質特性の課題設定数60個と品質を除く49個である。

一事例当たりの課題設定数は、3.3個（算出式：109個 ÷ 33件）である。品質モデルの品質特性では、1.8個（算出式：60個 ÷ 33件）である。品質以外では、1.5個（算出式：49個 ÷ 33件）である。

また、課題設定率のテーマ別上位は、品質（91%）、生産性（36%）、アシュアランス（27%）、コスト（27%）であった。品質モデルの品質特性では、機能適合性（55%）、保守性（36%）、信頼性（21%）が突出している。

表 3-41 課題のテーマの内訳と一事例当たりの課題設定率（組込み・制御系）

課題のテーマ:課題設定数(品質を除いて49個)。一事例当たり1.5個

	品質	生産性 (対応時間 短縮)	アシュアランス (保証)	コスト	納期	プロジェクト マネジメント	体制 (強化・再構築)	普及促進	障害原因の 分析	人材育成 意識改革	見積支援	グローバル 展開
対象事例数	30	12	9	9	4	4	4	3	2	2	0	0
課題設定率	91%	36%	27%	27%	12%	12%	12%	9%	6%	6%	0%	0%

品質モデルの品質特性:品質特性の課題設定数(60個)。一事例当たり1.8個

	機能適合性	保守性	信頼性	互換性	移植性	性能効率性	有効性	使用性	セキュリティ	リスク回避性	効率性	満足性	利用状況網 羅性
対象事例数	18	12	7	4	4	3	3	2	2	2	1	1	1
課題設定率	55%	36%	21%	12%	12%	9%	9%	6%	6%	6%	3%	3%	3%

#### 〔2〕対象適用事例から見る「効果」の状況

効果発揮の152個の内訳は、表3-42のように品質モデルの品質特性の課題設定数78個と品質を除く74個である。

一事例当たりの課題設定数は、4.6個（算出式：152個 ÷ 33件）である。品質モデルの品質特性では、2.4個（算出式：78個 ÷ 33件）である。品質以外では、2.2個（算出式：74個 ÷ 33件）である。

また、効果発揮率のテーマ別上位は、品質（97%）で発揮されて、次に生産性（61%）、コスト（42%）、人材育成・意識改革（27%）、アシュアランス（24%）であった。品質特性では、機能適合性（61%）、保守性（42%）、信頼性（27%）が突出している。

表 3-42 効果のテーマの内訳と一事例当たりの効果発揮率（組込み・制御系）

効果のテーマ:効果発揮数(品質を除いて74個)。一事例当たり2.2個

	品質	生産性 (対応時間 短縮)	コスト	人材育成 意識改革	アシュアランス (保証)	納期	プロジェクト マネジメント	体制 (強化・再構築)	障害原因の 分析	普及促進	見積支援	グローバル 展開
対象事例数	32	20	14	9	8	6	6	5	3	3	0	0
効果発揮率	97%	61%	42%	27%	24%	18%	18%	15%	9%	9%	0%	0%

品質モデルの品質特性:品質特性の効果発揮数(78個)。一事例当たり2.4個

	機能適合性	保守性	信頼性	性能効率性	使用性	移植性	有効性	互換性	セキュリティ	満足性	効率性	リスク回避性	利用状況網 羅性
対象事例数	20	14	9	5	5	5	5	3	3	3	2	2	2
効果発揮率	61%	42%	27%	15%	15%	15%	15%	9%	9%	9%	6%	6%	6%

[3] 対象適用事例から見る「副次効果」の状況

(ア) 対象事例に対する副次効果発揮状況

33 件の事例に対して 18 件の事例で副次効果が発生している。発揮率は 55% である。(算出式：副次効果発揮事例数 ÷ 事例数 = 18 件 ÷ 33 件)

これは、ひとつの課題に取り組むと、当初想定していない別の課題の解決につながるという点で、先進的な開発技術への取組みは、有益であることを意味する。

(イ) 事例／テーマ別の副次効果発揮状況

副次効果発揮の 48 個の内訳は、表 3-43 のように品質モデルの品質特性の課題設定数 20 個と品質を除く 28 個である。

一事例当たりの課題設定数は、1.5 個 (算出式:48 個 ÷ 33 件) である。品質モデルの品質特性では、0.6 個 (算出式:20 個 ÷ 33 件) である。品質以外では、0.8 個 (算出式:28 個 ÷ 33 件) である。

また、副次効果発揮率のテーマ別上位は、生産性 (24%)、品質 (21%)、人材育成・意識改革 (21%)、コスト (18%) である。品質モデルの品質特性では、機能適合性 (9%)、保守性 (9%) が突出している。

(ウ) 課題設定数に対する副次効果発揮状況

109 個の課題設定数に対して副次効果設定数は 48 個で 44% (算出式:48 個 ÷ 109 個) の発生率である。

表 3-43 副次効果のテーマの内訳と一事例当たりの副次効果発揮率 (組込み・制御系)

副次効果のテーマ:副次効果発揮数(品質を除いて28個)。一事例当たり0.8個

	生産性 (対応時間 短縮)	品質	人材育成 意識改革	コスト	納期	プロジェクト マネジメント	障害原因の 分析	体制 (強化・再構 築)	アシュアラン ス(保証)	見積支援	普及促進	グローバル 展開
対象事例数	8	7	7	6	3	2	1	1	0	0	0	0
副次効果発揮率	24%	21%	21%	18%	9%	6%	3%	3%	0%	0%	0%	0%

品質モデルの品質特性:品質特性の副次効果発揮数(20個)。一事例当たり0.6個

	機能適合性	使用性	性能効率性	信頼性	保守性	有効性	満足性	セキュリティ	移植性	効率性	利用状況網 羅性	互換性	リスク回避性
対象事例数	3	3	2	2	2	2	2	1	1	1	1	0	0
副次効果発揮率	9%	9%	6%	6%	6%	6%	6%	3%	3%	3%	3%	0%	0%

〔4〕対象適用事例から見る「さらなる取組」の状況

さらなる取組の16個の内訳は、表3-44のように品質モデルの品質特性の課題設定数11個と品質を除く5個である。

一事例当たりの課題設定数は、0.5個（算出式：16個 ÷ 33件）である。品質モデルの品質特性では、0.3個（算出式：11個 ÷ 33件）である。品質以外では、0.2個（算出式：5個 ÷ 33件）である。

また、さらなる取組設定率のテーマ別上位は、品質（33%）、生産性（6%）である。品質モデルの品質特性では、機能適合性（24%）、保守性（9%）が突出している。

表 3-44 さらなる取組設定のテーマの内訳と一事例当たりのさらなる取組設定率（組込み・制御系）

さらなる取組設定のテーマ：さらなる取組設定数（品質を除いて5個）。一事例当たり0.2個

	品質	生産性 (対応時間 短縮)	コスト	プロジェクト マネジメント	普及促進	アジャラン ス(保証)	障害原因の 分析	納期	人材育成 意識改革	見積支援	体制 (強化・再構 築)	グローバル 展開
対象事例数	11	2	1	1	1	0	0	0	0	0	0	0
さらなる取組 設定率	33%	6%	3%	3%	3%	0%	0%	0%	0%	0%	0%	0%

品質モデルの品質特性：品質特性のさらなる取組設定数（11個）。一事例当たり0.3個

	機能適合性	保守性	性能効率性	互換性	使用性	信頼性	セキュリティ	移植性	有効性	効率性	満足性	リスク回避性	利用状況網 羅性
対象事例数	8	3	0	0	0	0	0	0	0	0	0	0	0
さらなる取組 設定率	24%	9%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%

〔5〕対象適用事例から見る「新規取組」の状況

新規取組の25個の内訳は、表3-45のように品質モデルの品質特性の課題設定数7個と品質を除く18個である。

一事例当たりの課題設定数は、0.8個（算出式：25個 ÷ 33件）である。品質モデルの品質特性では、0.2個（算出式：7個 ÷ 33件）である。品質以外では、0.5個（算出式：18個 ÷ 33件）である。

また、新規取組設定率のテーマ別上位は、普及促進（30%）、品質（21%）である。品質モデルの品質特性では、機能適合性（15%）が突出している。

表 3-45 新規取組設定のテーマの内訳と一事例当たりの新規取組設定率（組込み・制御系）

新規取組設定のテーマ：新規取組設定数（品質を除いて18個）。一事例当たり0.5個

	普及促進	品質	コスト	人材育成 意識改革	プロジェクト マネジメント	体制 (強化・再構 築)	アジャラン ス(保証)	障害原因の 分析	納期	生産性 (対応時間 短縮)	見積支援	グローバル 展開
対象事例数	10	7	2	2	2	2	0	0	0	0	0	0
新規取組設定率	30%	21%	6%	6%	6%	6%	0%	0%	0%	0%	0%	0%

品質モデルの品質特性：品質特性の新規取組設定数（7個）。一事例当たり0.2個

	機能適合性	使用性	保守性	性能効率性	互換性	信頼性	セキュリティ	移植性	有効性	効率性	満足性	リスク回避性	利用状況網 羅性
対象事例数	5	1	1	0	0	0	0	0	0	0	0	0	0
新規取組設定率	15%	3%	3%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%

3.2.2.3.3 適用工程

組込み・制御系では、20件の設計系と13件の検証系の33件の適用事例が対象となった。超上流工程に関わる適用事例は5件、保守・運用工程に関わる事例は6件である。

表 3-46 適用事例ごとの適用工程分布（組込み・制御系）

事例 参照 番号	適用 領域 ②		適用工程 共通フレーム2013									
	Web・フロント系 エンタプライズ系	組込み・制御系	企画フェーズ	要件定義フェーズ	システム・ソフトウェア要件定義	基本設計	詳細設計	製作・ユニットテスト	ソフトウェア結合・総合テスト	システム結合・総合テスト	移行・運用準備フェーズ	運用・保守フェーズ
A-1		■			■		■	■				■
A-2	■	■				■	■	■				■
A-3		■		■								
A-9		■				■	■	■				
A-10		■				■	■	■				
A-11		■				■	■	■	■	■	■	■
A-12		■				■	■	■	■	■		
A-13		■				■	■	■	■	■		
15-A-9	■	■		■		■	■	■				
15-A-10		■				■	■					
15-A-11		■										■
15-A-12		■				■	■	■	■	■		■
15-A-13		■				■	■	■				
15-A-14		■		■		■	■	■				
15-A-15		■		■		■	■	■			■	■
15-A-16		■					■	■				
15-A-17		■				■	■	■				
15-A-18		■				■	■	■		■		
15-A-19		■				■	■	■				
15-A-20		■				■	■	■				
B-1		■				■	■		■	■		
B-2		■										
B-5		■						■				
B-6		■							■	■		
B-7		■						■	■	■		
B-8		■							■	■		
B-9		■							■	■		
B-10		■							■	■		
B-11		■							■	■		
15-B-7	■	■							■	■		
15-B-8	■	■		■								
15-B-13		■							■	■		
15-B-14		■						■	■	■		

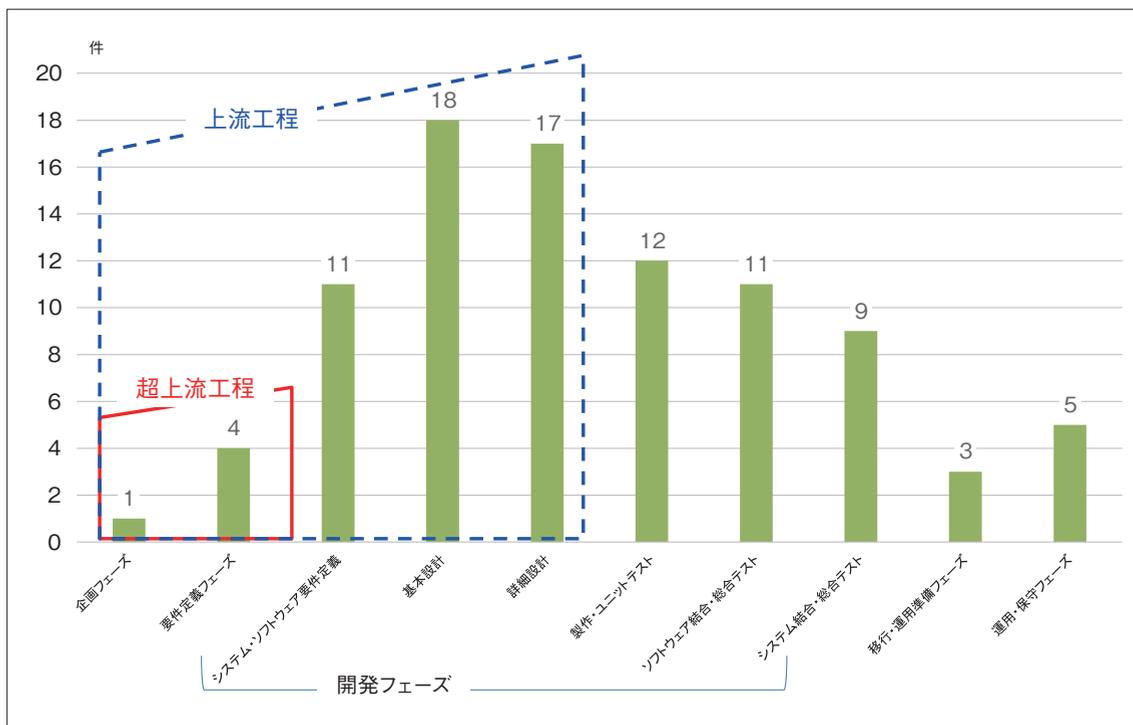


図 3-19 適用工程（組込み・制御系）

### 3.2.2.4 適用領域②(エンタプライズ系、Web・フロント系、組込み・制御系)で分かったこと

#### 〔1〕対象適用事例から見る「課題」の状況

一事例当たりの課題設定数では、組込み・制御系（33件）は、エンタプライズ系（57件）、Web・フロント系（49件）よりもかなり少ない。

課題で上位を占めているテーマは、「品質」、「コスト」、「生産性」でほぼ同じである。しかし、組込み・制御系の「アシュアランス」（27％）は、他の適用領域では、目立っていないもので、組込み・制御系の特徴のようである。また、品質関連では、「機能適合性」、「保守性」が上位であることは、同じであるが、それより下位では、エンタプライズ系では「満足性」、Web・フロント系では「セキュリティ」、組込み・制御系では「信頼性」と、それぞれの領域の中で注目されている。

#### 〔2〕対象適用事例から見る「効果」の状況

一事例当たりの効果発揮数では、組込み・制御系（46件）は、エンタプライズ系（66件）、Web・フロント系（59件）よりも少ない。

効果で上位を占めているテーマは、「品質」、「コスト」、「生産性」、「納期」、「人材育成・意識改革」までほぼ同じである。エンタプライズ系の「体制」（29％）は、Web・フロント系（17％）、組込み・制御系（15％）と比べて多い。「アシュアランス」に関して、組込み・制御系（24％）とエンタプライズ系（19％）は、Web・フロント系（8％）に比べて多い。また、品質関連では、「機能適合性」と「保守性」が上位で、同じであるが、「セキュリティ」に関しては、Web・フロント系（33％）が、エンタプライズ系（14％）と組込み・制御系（9％）に比べて多い。

#### 〔3〕対象適用事例から見る「副次効果」の状況

一事例当たりの副次効果発揮数では、組込み・制御系（15件）は、エンタプライズ系（11件）、Web・フロント系（11件）よりも多い。

副次効果で上位を占めているテーマは、「品質」、「コスト」、「生産性」、「人材育成・意識改革」でほぼ同じである。また、品質関連では、「機能適合性」がトップで同じである。

#### 〔4〕対象適用事例から見る「さらなる取組」の状況

一事例当たりのさらなる取組設定数では、Web・フロント系（1.1件）は、エンタプライズ系（0.7件）、組込み・制御系（0.5件）よりも多い。

さらなる取組で上位を占めているテーマは、「品質」が共通している。また、「コスト」、「体制」については、エンタプライズ系では19％、14％、Web・フロント系では8％、0％、組込み・制御系では3％、0％となり、エンタプライズ系に特徴がみられる。

#### 〔5〕対象適用事例から見る「新規取組」の状況

一事例当たりの新規取組設定数では、エンタプライズ系（1.2件）は、Web・フロント系（0.9件）、組込み・制御系（0.8件）より若干多い。

新規取組で上位を占めているテーマは、「普及促進」、「品質」が共通している。また、「プロジェクトマネジメント」については、エンタプライズ系が33％、Web・フロント系が8％、組込み・制御系が6％と、エンタプライズ系に特徴がみられる。

#### 〔6〕適用工程の状況

エンタプライズ系では、超上流工程からの取組みが48％（対象事例21件中10件）と多い。一方、Web・フロント系は33％（対象事例12件中4件）、組込み・制御系は15％（対象事例33件中5件）で、組込み・制御系の超上流での取組みが少ない。保守・運用工程での取組みでは、エンタプライズ系は29％（対象事例21件中6件）、Web・フロント系は33％（対象事例12件中4件）、組込み・制御系は18％（対象事例33件中6件）である。

### 3.2.3 適用技術・手法別の適用結果等状況

#### 3.2.3.1 適用結果

今回適用されている技術・手法は、28種類ある。その内、適用している事例が3件以上あるものは、21種類であった。7種類は2件以下の事例しかなかったため議論の対象外とした。適用結果を表3-47に示す。この表の並びは、一事例当たりの効果発揮数が多い順に示す（一事例当たりの効果数を赤字で表記）。

表 3-47 適用技術・手法別の適用結果状況

技術・手法		事例(件)	課題設定	効果発揮	課題克服	副次効果発生	さらなる取組設定	新規取組設定
RAD【高速開発】	個数	4	39	46	37	9	5	8
	割合※	—	—	118%	95%	23%	13%	21%
	一事例当り	—	9.8	11.5	9.3	2.3	1.3	2.0
SPLE【ソフトウェアプロダクトライン開発】	個数	3	23	24	23	1	1	2
	割合※	—	—	104%	100%	4%	4%	9%
	一事例当り	—	7.7	8.0	7.7	0.3	0.3	0.7
BPM【ビジネスプロセスマネジメント】	個数	6	45	47	44	3	3	10
	割合※	—	—	104%	98%	7%	7%	22%
	一事例当り	—	7.5	7.8	7.3	0.5	0.5	1.7
MDD【モデル駆動開発】	個数	4	27	30	27	3	3	7
	割合※	—	—	111%	100%	11%	11%	26%
	一事例当り	—	6.8	7.5	6.8	0.8	0.8	1.8
アジャイルソフトウェア開発モデル	個数	6	38	44	38	6	7	10
	割合※	—	—	116%	100%	16%	18%	26%
	一事例当り	—	6.3	7.3	6.3	1.0	1.2	1.7
アシユアランスケース (D-Case)	個数	6	27	44	26	18	7	6
	割合※	—	—	163%	96%	67%	26%	22%
	一事例当り	—	4.5	7.3	4.3	3.0	1.2	1.0
UX【ユーザーエクスペリエンス】	個数	3	17	21	17	4	7	4
	割合※	—	—	124%	100%	24%	41%	24%
	一事例当り	—	5.7	7.0	5.7	1.3	2.3	1.3
HCD【人間中心設計】	個数	3	17	20	17	3	3	3
	割合※	—	—	118%	100%	18%	18%	18%
	一事例当り	—	5.7	6.7	5.7	1.0	1.0	1.0
MBSE【モデルベースシステムズエンジニアリング】	個数	8	38	50	36	14	5	7
	割合※	—	—	132%	95%	37%	13%	18%
	一事例当り	—	4.8	6.3	4.5	1.8	0.6	0.9
保守・更改・派生開発 (XDDP)	個数	11	57	67	56	11	10	16
	割合※	—	—	118%	98%	19%	18%	28%
	一事例当り	—	5.2	6.1	5.1	1.0	0.9	1.5
イテレーション開発モデル【反復型開発モデル】	個数	10	51	60	50	10	10	9
	割合※	—	—	118%	98%	20%	20%	18%
	一事例当り	—	5.1	6.0	5.0	1.0	1.0	0.9
形式仕様記述	個数	7	36	42	34	8	7	4
	割合※	—	—	117%	94%	22%	19%	11%
	一事例当り	—	5.1	6.0	4.9	1.1	1.0	0.6
MBD【モデルベース開発】	個数	13	54	65	50	15	8	11
	割合※	—	—	120%	93%	28%	15%	20%
	一事例当り	—	4.2	5.0	3.8	1.2	0.6	0.8
モデル検査	個数	5	19	24	19	5	5	1
	割合※	—	—	126%	100%	26%	26%	5%
	一事例当り	—	3.8	4.8	3.8	1.0	1.0	0.2
テスト自動化	個数	8	35	38	35	3	14	7
	割合※	—	—	109%	100%	9%	40%	20%
	一事例当り	—	4.4	4.8	4.4	0.4	1.8	0.9
耐故障性検証(運用時)	個数	6	23	28	22	6	6	9
	割合※	—	—	122%	96%	26%	26%	39%
	一事例当り	—	3.8	4.7	3.7	1.0	1.0	1.5
独自開発・検証ツール	個数	17	60	76	58	18	21	16
	割合※	—	—	127%	97%	30%	35%	27%
	一事例当り	—	3.5	4.5	3.4	1.1	1.2	0.9
セキュリティ	個数	4	15	17	15	2	6	4
	割合※	—	—	113%	100%	13%	40%	27%
	一事例当り	—	3.8	4.3	3.8	0.5	1.5	1.0
MILS/HILS/SILS	個数	5	15	20	14	6	4	5
	割合※	—	—	133%	93%	40%	27%	33%
	一事例当り	—	3.0	4.0	2.8	1.2	0.8	1.0
バグ分析(ODC分析)	個数	4	8	14	8	6	9	5
	割合※	—	—	175%	100%	75%	113%	63%
	一事例当り	—	2.0	3.5	2.0	1.5	2.3	1.3
形式検証	個数	3	6	7	5	2	2	0
	割合※	—	—	117%	83%	33%	33%	0%
	一事例当り	—	2.0	2.3	1.7	0.7	0.7	0.0

注：※は、課題設定数との割合

3.2.3.2 適用技術・手法別の効果を発揮したテーマ

事例に適用されている技術・手法が、どのテーマにどの程度効果を発揮しているかを、適用事例数当りのテーマ別効果発揮割合として、以下の表に示す。表 3-48 では、品質以外のテーマについて示し、表 3-49 では、品質特性について示している。

例) RAD【高速開発】の場合

- ・適用事例数は 4 件である。
  - ・「コスト」には 4 件の事例全てにおいて効果が発揮できているので、  
 $4 \text{ (コストに効果を発揮した事例数)} \div 4 \text{ (適用事例数)} = 1.00$  と数値を記載している。
  - ・「納期」には 3 件の事例において効果が発揮できているので、  
 $3 \text{ (納期に効果を発揮した事例数)} \div 4 \text{ (適用事例数)} = 0.75$  と数値を記載している。
- ※数値は、計算結果の小数点第三位を四捨五入し、第二位まで表している。

赤色が濃いほど効果を発揮した事例数の割合が高い。

- ・品質については、全ての技術・手法が 1.00 となっており、効果を発揮している。
- ・効果発揮割合が 0.50 以上のテーマ数が 5 以上あるものに○を、2 以下のものに▲を付与している（表 3-48 右端の欄参照）。○のついた技術・手法は、多くのテーマに効果を発揮するもので、▲のついた技術・手法は、特定のテーマや目的に効果を発揮するものだといえる。

表 3-48 適用技術・手法別の効果を発揮したテーマの割合

技術・手法 \ テーマ	適用事例数	品質	アシユアランス (保証)	障害原因の分析	コスト	納期	(対応時間短縮) 生産性	人材育成改革	プロジェクトマネジメント	見積支援	普及促進	(強化・再構築) 体制	グローバル展開	0.50 未満の種別
RAD【高速開発】	4	1.00	0.00	0.00	1.00	0.75	0.75	1.00	0.25	0.00	0.25	1.00	0.00	○
SPL【ソフトウェアプロダクトライン開発】	3	1.00	0.00	0.00	0.33	0.33	1.00	0.67	0.33	0.00	0.33	1.00	0.00	
BPM【ビジネスプロセスマネジメント】	6	1.00	0.00	0.00	1.00	0.33	0.50	0.33	0.00	0.17	0.33	0.33	0.17	
MDD【モデル駆動開発】	4	1.00	0.25	0.00	0.50	0.50	1.00	0.50	0.50	0.00	0.00	0.50	0.00	○
アジャイルソフトウェア開発モデル	6	1.00	0.33	0.00	0.50	0.67	0.50	0.50	0.17	0.00	0.33	0.33	0.00	○
アシユアランス(D-Case)	6	1.00	0.50	0.00	0.33	0.17	0.33	0.50	0.50	0.17	0.17	0.00	0.00	
UX【ユーザーエクスペリエンス】	3	1.00	0.00	0.33	0.67	0.33	0.33	0.67	0.33	0.00	0.67	0.33	0.00	
HCD【人間中心設計】	3	1.00	0.00	0.33	0.33	0.33	0.00	1.00	0.33	0.00	0.67	0.33	0.00	
MBSE【モデルベースシステムズエンジニアリング】	8	1.00	0.13	0.13	0.75	0.38	0.75	0.50	0.38	0.00	0.00	0.13	0.00	
保守・更改・派生開発(XDDP)	11	1.00	0.18	0.00	0.73	0.27	0.64	0.27	0.27	0.00	0.18	0.36	0.00	
イテレーション開発モデル【反復型開発モデル】	10	1.00	0.10	0.00	0.70	0.60	0.70	0.30	0.10	0.00	0.20	0.40	0.00	
形式仕様記述	7	1.00	0.43	0.14	0.71	0.29	1.00	0.29	0.29	0.00	0.00	0.14	0.00	
MBD【モデルベース開発】	13	1.00	0.15	0.15	0.69	0.31	0.69	0.31	0.23	0.00	0.15	0.08	0.00	
モデル検査	5	1.00	0.40	0.00	0.60	0.40	0.40	0.00	0.00	0.00	0.20	0.20	0.00	▲
テスト自動化	8	0.88	0.13	0.00	0.63	0.63	0.50	0.25	0.00	0.00	0.13	0.13	0.00	
耐故障性検証(運用時)	6	1.00	0.33	0.17	0.17	0.17	0.83	0.00	0.00	0.17	0.00	0.00	0.00	▲
独自開発・検証ツール	17	0.94	0.18	0.00	0.76	0.41	0.71	0.24	0.06	0.00	0.12	0.18	0.00	
セキュリティ	4	1.00	0.25	0.00	0.50	0.25	0.75	0.50	0.25	0.00	0.00	0.00	0.00	
MILS/HILS/SILS	5	1.00	0.00	0.00	0.80	0.20	1.00	0.20	0.00	0.00	0.00	0.00	0.00	
バグ分析(ODG分析)	4	1.00	0.50	0.25	0.25	0.25	0.75	0.00	0.00	0.00	0.00	0.00	0.00	
形式検証	3	1.00	0.33	0.00	0.33	0.00	0.33	0.00	0.00	0.00	0.00	0.00	0.00	▲

表 3-49 は、品質だけを捉えてその効果発揮割合を示したものである。表 3-48 と同様に、赤色が濃いほど効果を発揮した事例数の割合が高い。

- ・効果発揮割合が0.50以上の品質テーマ数が5以上あるものに○を、2以下のものに▲を付与している(表 3-49 右端の欄参照)。○のついた技術・手法は、多くの品質テーマに効果を発揮するもので、また 0.50以上の品質テーマ数が▲のついた技術・手法は、特定の品質テーマに効果を発揮するものだといえる。

表 3-49 適用技術・手法別の効果を発揮したテーマの割合 (品質特性)

技術・手法	テーマ (品質特性)	システム及びソフトウェア品質モデル(JIS X 25010:2013)													0.50 以上の 種別(品質) テーマ数
		適用 事例 数	製品品質						利用時の品質						
			機能 適合 性	性能 効率 性	互 換 性	使用 性	信 頼 性	セ キ ユ リ テ ィ	保 守 性	移 植 性	有 効 性	効 率 性	満 足 性	リ ス ク 回 避 性	
RAD【高速開発】	4	1.00	0.25	0.50	0.50	0.25	0.00	1.00	0.75	0.50	0.50	0.75	0.00	0.50	○
SPL【ソフトウェアプロダクトライン開発】	3	0.67	0.67	0.67	0.33	0.33	0.00	0.67	0.67	0.00	0.00	0.00	0.00	0.00	○
BPM【ビジネスプロセスマネジメント】	6	0.67	0.67	0.33	0.50	0.33	0.17	0.67	0.33	0.17	0.33	0.33	0.00	0.17	
MDD【モデル駆動開発】	4	1.00	0.00	0.50	0.25	0.00	0.00	0.50	0.50	0.25	0.25	0.25	0.00	0.25	
アジャイルソフトウェア開発モデル	6	1.00	0.17	0.33	0.33	0.33	0.00	0.67	0.33	0.17	0.17	0.33	0.00	0.17	▲
アシュアランスケース(D-Case)	6	0.67	0.33	0.17	0.50	0.67	0.33	0.83	0.33	0.17	0.17	0.17	0.17	0.17	
UX【ユーザーエクスペリエンス】	3	0.33	0.00	0.00	0.00	0.33	0.00	0.00	0.00	0.67	0.33	0.67	0.67	0.33	
HCD【人間中心設計】	3	0.00	0.00	0.00	0.33	0.00	0.00	0.00	0.00	1.00	0.67	0.67	0.33	0.33	
MBSE【モデルベースシステムズエンジニアリング】	8	0.75	0.13	0.13	0.13	0.25	0.00	0.88	0.25	0.13	0.13	0.13	0.00	0.25	▲
保守・更改・派生開発(XDDP)	11	0.64	0.09	0.27	0.18	0.27	0.00	0.91	0.36	0.09	0.09	0.18	0.00	0.09	▲
イテレーション開発モデル【反復型開発モデル】	10	0.80	0.20	0.20	0.20	0.20	0.00	0.50	0.30	0.10	0.10	0.20	0.00	0.10	▲
形式仕様記述	7	1.00	0.00	0.14	0.14	0.29	0.00	0.43	0.14	0.14	0.14	0.14	0.00	0.14	▲
MBD【モデルベース開発】	13	0.62	0.23	0.08	0.08	0.31	0.00	0.54	0.23	0.08	0.00	0.00	0.00	0.08	▲
モデル検査	5	0.80	0.20	0.00	0.20	0.60	0.00	0.20	0.20	0.20	0.00	0.20	0.00	0.00	▲
テスト自動化	8	0.75	0.13	0.00	0.13	0.25	0.13	0.38	0.13	0.13	0.13	0.13	0.00	0.13	▲
耐故障性検証(運用時)	6	0.33	0.33	0.00	0.33	0.50	0.33	0.67	0.33	0.00	0.00	0.00	0.00	0.00	▲
独自開発・検証ツール	17	0.65	0.18	0.00	0.12	0.29	0.06	0.35	0.06	0.06	0.00	0.00	0.06	0.00	▲
セキュリティ	4	0.25	0.25	0.00	0.25	0.00	0.75	0.00	0.00	0.00	0.00	0.00	0.25	0.00	▲
MILS/HILS/SILS	5	0.40	0.40	0.00	0.00	0.20	0.00	0.40	0.20	0.20	0.00	0.00	0.00	0.00	
バグ分析(ODC分析)	4	0.75	0.00	0.00	0.00	0.25	0.00	0.25	0.00	0.00	0.00	0.00	0.25	0.00	▲
形式検証	3	0.67	0.00	0.00	0.00	0.33	0.33	0.00	0.00	0.00	0.00	0.00	0.00	0.00	▲

3.2.3.3 適用工程

今回適用されている技術・手法が適用されている工程を表 3-50 で示す。赤色が濃いほど効果を発揮した事例数の割合が高い。

- ・効果発揮割合が0.50以上の工程が4以上の技術・手法に○を、2以下のものに▲を付与している（表 3-50 右端の欄参照）。
- ・○のついた技術・手法は、多くの工程に効果を発揮するもので、▲のついた技術・手法は、特定の工程に効果を発揮するものだけといえる。

表 3-50 適用技術・手法別の効果を発揮した適用工程

開発工程 技術・手法	適用事例数	全体										0・50以上の工程数で種別
		企画フェーズ	要件定義フェーズ	システム・ソフトウェア要件定義	基本設計	詳細設計	製作・ユニットテスト	ソフトウェア結合・総合テスト	システム結合・総合テスト	移行・運用準備フェーズ	運用・保守フェーズ	
RAD【高速開発】	4	0.00	0.25	1.00	1.00	1.00	0.75	0.50	0.50	0.50	0.75	○
SPL【ソフトウェアプロダクトライン開発】	3	0.00	0.00	0.33	1.00	1.00	0.67	0.33	0.33	0.00	0.33	
BPM【ビジネスプロセスマネジメント】	6	0.17	0.83	1.00	0.50	0.50	0.67	0.33	0.33	0.00	0.17	○
MDD【モデル駆動開発】	4	0.00	0.25	0.50	1.00	1.00	0.75	0.50	0.50	0.25	0.25	○
アジャイルソフトウェア開発モデル	6	0.00	0.33	0.50	0.50	0.83	0.67	0.50	0.33	0.17	0.17	○
アシュアランスケース(D-Case)	6	0.00	0.00	0.17	0.50	0.33	0.50	0.17	0.00	0.33	0.17	▲
UX【ユーザーエクスペリエンス】	3	0.67	0.00	0.00	0.33	0.67	0.33	0.33	0.33	0.00	0.00	▲
HCD【人間中心設計】	3	0.67	0.33	0.33	0.33	0.67	0.00	0.00	0.00	0.00	0.00	▲
MBSE【モデルベースシステムズエンジニアリング】	8	0.00	0.38	0.88	0.75	0.75	0.38	0.38	0.38	0.38	0.25	
保守・更改・派生開発(XDDP)	11	0.00	0.00	0.36	0.55	0.64	0.45	0.55	0.55	0.18	0.45	
イテレーション開発モデル【反復型開発モデル】	10	0.10	0.20	0.40	0.50	0.60	0.60	0.80	0.50	0.20	0.30	○
形式仕様記述	7	0.00	0.29	0.71	0.57	0.57	0.57	0.71	0.43	0.57	0.43	○
MBD【モデルベース開発】	13	0.00	0.15	0.38	0.69	0.69	0.54	0.31	0.31	0.08	0.23	
モデル検査	5	0.00	0.00	0.20	0.40	0.20	0.60	1.00	0.60	0.00	0.20	
テスト自動化	8	0.00	0.13	0.13	0.25	0.38	0.75	0.88	0.38	0.13	0.13	▲
耐故障性検証(運用時)	6	0.00	0.33	0.50	0.67	0.67	0.00	0.00	0.00	0.50	0.50	○
独自開発・検証ツール	17	0.06	0.12	0.35	0.53	0.53	0.53	0.71	0.47	0.06	0.18	
セキュリティ	4	0.00	0.25	0.50	0.25	0.50	0.50	0.75	0.50	0.25	0.25	○
MILS/HILS/SILS	5	0.00	0.00	0.40	0.80	0.80	0.60	0.60	0.40	0.00	0.00	
バグ分析(ODC分析)	4	0.00	0.00	0.00	0.50	0.50	0.50	0.75	0.50	0.00	0.00	○
形式検証	3	0.00	0.00	0.00	0.00	0.33	0.67	1.00	0.00	0.00	0.00	▲

### 3.2.3.4 適用領域②別適用工程

ここでは、適用領域②のエンタプライズ系、Web・フロント系、組込み・制御系のそれぞれに分けて表3-51に示す。

赤色の濃さによって、技術・手法の適用領域②のどこに効果を発揮するかを読み取ることができる。

表 3-51 適用技術・手法の適用領域②別適用工程

開発工程	技術・手法	エンタプライズ系										Web・フロント系										組込み・制御系															
		適用事例数	企画フェーズ	要件定義フェーズ	システム・ソフトウェア要件定義	基本設計	詳細設計	製作・ユニットテスト	ソフトウェア結合・総合テスト	システム結合・総合テスト	移行・運用準備フェーズ	運用・保守フェーズ	適用事例数	企画フェーズ	要件定義フェーズ	システム・ソフトウェア要件定義	基本設計	詳細設計	製作・ユニットテスト	ソフトウェア結合・総合テスト	システム結合・総合テスト	移行・運用準備フェーズ	運用・保守フェーズ	適用事例数	企画フェーズ	要件定義フェーズ	システム・ソフトウェア要件定義	基本設計	詳細設計	製作・ユニットテスト	ソフトウェア結合・総合テスト	システム結合・総合テスト	移行・運用準備フェーズ	運用・保守フェーズ			
RAD【高速開発】		4	0.00	0.25	1.00	1.00	1.00	0.75	0.50	0.50	0.50	0.75	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	
SPL【ソフトウェアプロダクトライン開発】		0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.33	
BPM【ビジネスプロセスマネジメント】		6	0.17	0.83	1.00	0.50	0.50	0.67	0.33	0.33	0.33	0.00	1	0.00	1.00	1.00	0.00	0.00	1.00	0.00	0.00	0.00	1.00	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
MDD【モデル駆動開発】		2	0.00	0.50	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	2	0.00	0.00	0.00	1.00	1.00	1.00	0.50	0.50	0.00	0.00	0.00	0.00	0.00
アジャイルソフトウェア開発モデル		4	0.00	0.50	0.75	0.75	1.00	0.75	0.50	0.25	0.25	0.25	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	1	0.00	0.00	0.00	0.00	0.00	1.00	1.00	0.00	0.00	0.00	0.00	0.00
アシユアランスケース(D-Case)		2	0.00	0.00	0.50	0.50	0.00	0.00	0.00	0.00	0.50	0.50	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	4	0.00	0.00	0.00	0.50	0.50	0.75	0.25	0.00	0.25	0.00	0.00	0.00	0.00	
UX【ユーザーエクスペリエンス】		3	0.67	0.00	0.00	0.33	0.67	0.33	0.33	0.33	0.33	0.00	2	1.00	0.00	0.00	0.00	0.50	0.00	0.00	0.00	0.00	1	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
HCD【人間中心設計】		3	0.67	0.00	0.00	0.00	0.33	0.00	0.00	0.00	0.00	0.00	2	1.00	0.00	0.00	0.00	0.50	0.00	0.00	0.00	0.00	1	1.00	1.00	1.00	1.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	
MBSE【モデルベースシステムズエンジニアリング】		2	0.00	0.50	1.00	0.50	0.50	0.50	1.00	1.00	1.00	0.50	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	6	0.00	0.33	0.83	0.83	0.83	0.33	0.17	0.17	0.17	0.17	0.17	0.17	0.17	0.17
保守・更改・派生開発(XDDP)		4	0.00	0.00	0.75	0.75	0.75	0.75	1.00	1.00	0.50	0.50	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	7	0.00	0.00	0.14	0.43	0.57	0.29	0.14	0.29	0.00	0.43	0.00	0.00	0.00	0.00
イテレーション開発モデル【反復型開発モデル】		6	0.17	0.33	0.67	0.50	0.67	0.50	0.67	0.50	0.33	0.33	3	0.00	0.00	0.00	0.33	0.33	0.67	0.00	0.33	0.00	1	0.00	0.00	0.00	1.00	1.00	1.00	0.00	1.00	1.00	0.00	0.00	0.00	0.00	
形式仕様記述		3	0.00	0.33	0.67	0.33	0.33	0.67	0.67	0.67	0.67	0.33	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	4	0.00	0.25	0.75	0.75	0.75	0.50	0.50	0.25	0.50	0.50	0.25	0.50	0.50	
MBD【モデルベース開発】		0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	13	0.00	0.15	0.38	0.69	0.69	0.54	0.31	0.31	0.08	0.23	0.00	0.00	0.00	0.00	
モデル検査		1	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	4	0.00	0.00	0.25	0.50	0.25	0.50	0.75	0.75	0.00	0.25	0.00	0.00	0.00	0.00	
テスト自動化		3	0.00	0.33	0.33	0.33	0.67	0.67	0.67	0.33	0.00	0.00	3	0.00	0.00	0.00	0.33	0.33	0.67	0.33	0.67	0.33	2	0.00	0.00	0.00	0.00	0.00	1.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	
耐故障性検証(運用時)		2	0.00	0.50	0.50	0.50	0.50	0.00	0.00	0.00	0.50	0.50	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	3	0.00	0.33	0.67	1.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.33	0.33	0.00	0.00
独自開発・検証ツール		7	0.14	0.29	0.57	0.57	0.57	0.43	0.43	0.43	0.00	0.14	4	0.00	0.00	0.25	0.25	0.25	0.50	0.50	0.75	0.25	0.25	6	0.00	0.00	0.17	0.67	0.67	0.67	0.33	0.33	0.00	0.17	0.00		
セキュリティ		1	0.00	1.00	1.00	1.00	1.00	0.00	0.00	0.00	0.00	0.00	3	0.00	0.33	0.67	0.33	0.33	0.33	0.67	0.67	0.33	2	0.00	0.50	0.50	0.50	1.00	0.50	0.50	0.00	0.00	0.00	0.00	0.00	0.00	
MILS/HILS/SILS		0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	5	0.00	0.00	0.40	0.80	0.80	0.60	0.60	0.40	0.00	0.00	0.00	0.00	0.00		
バグ分析(ODC分析)		2	0.00	0.00	0.00	0.50	0.50	0.50	1.00	0.50	0.00	0.00	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	3	0.00	0.00	0.00	0.33	0.33	0.33	0.67	0.33	0.00	0.00	0.00	0.00	0.00	0.00	
形式検証		0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	3	0.00	0.00	0.00	0.00	0.00	0.33	0.67	0.67	0.00	0.00	0.00	0.00	0.00		

### 3.2.3.5 併用されている技術・手法

全事例で適用されている 28 種類の技術・手法の中で、併用して使われている技術・手法の併用率を求めた。

併用率とは、ある技術・手法に関して、併用している別の技術・手法の割合（併用している技術・手法を利用している件数÷対象事例数）を意味する。表 3-52 では、併用率が 0.5 以上のものをぬりつぶし白抜き文字で示す。ただし、対象事例数が 1 件の「W（開発）モデル」、「CASE」、「ソフトウェアフレームワーク」、「IV&V」の 4 つの技術・手法の併用率の結果は、対象事例数が少ないので、併用率が 0.5 以上であってもぬりつぶし白抜き文字の表記はしていない。

例) アジャイルソフトウェア開発モデルの場合

- ・適用事例数は 6 件である。
- ・イテレーション開発モデルの併用率は 0.67 である。  
(イテレーション開発モデルを併用していた事例は 4 件なので、4 件 ÷ 対象事例 6 件 = 0.67)
- ・BPM の併用率は 0.17 である。  
(BPM を併用していた事例は 1 件なので、1 件 ÷ 対象事例 6 件 = 0.17)

表 3-52 適用した技術・手法の併用している技術・手法

技術・手法 (併用)	対象事例数	アジャイルソフトウェア開発モデル	ソフトウェアプロトタイプ開発モデル	イテレーション開発モデル【反復型開発モデル】	W (開発) モデル	BPM【ビジネスプロセスマネジメント】	MBSSE【モデルベースシステムズエンジニアリング】	形式仕様記述	HCD【人間中心設計】	UX【ユーザーエクスペリエンス】	MDD【モデル駆動開発】	MILS/HILS/SILS	RAD【高速開発】	CASE	IDE【統合開発環境】	SPL (ソフトウェアプロダクトライン開発)	ソフトウェアフレームワーク	テスト自動化	バグ分析 (ODC分析)	形式検証	モデル検査	非機能要求グレイド	耐故障性検証 (運用時)	アシユアランスケース (D-Case)	セキュリティ	リスクアセスメント	独自開発・検証ツール	
アジャイルソフトウェア開発モデル	6	0	0	0.67	0	0	0	0	0	0.17	0.17	0.33	0	0	0	0	0	0.50	0	0	0	0	0.17	0.17	0	0.17	0.33	
ソフトウェアプロトタイプ開発モデル	2	0	1	0	0.5	0.5	0.5	0	0	0	0	0	0.5	0.5	0.5	0	0	0	0	0	0	0	0	0	0	0	0.5	
イテレーション開発モデル【反復型開発モデル】	10	0.4	0.2	0	0	0.1	0.1	0.1	0	0	0.1	0.2	0.4	0	0.2	0.1	0.1	0.1	0	0.4	0	0.1	0	0.1	0	0	0.6	
W (開発) モデル	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
BPM【ビジネスプロセスマネジメント】	6	0.17	0.17	0.17	0	0	0	0	0	0	0	0.17	0	0	0	0	0	0.17	0	0	0	0	0	0	0	0	0.33	
MBSSE【モデルベースシステムズエンジニアリング】	8	0	0.13	0.13	0	0	0.38	0	0	0	0.63	0.13	0.25	0.25	0.13	0.13	0.13	0	0	0	0	0	0	0.25	0.13	0	0.13	
形式仕様記述	7	0	0.14	0.14	0	0	0.43	0	0	0	0.29	0.14	0.14	0	0.14	0.14	0.14	0	0	0	0.14	0.14	0	0.14	0	0	0.14	
HCD【人間中心設計】	3	0	0	0	0	0	0	0	0	0.67	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
UX【ユーザーエクスペリエンス】	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.33	0	0	0	0	0	0	0.33	
MDD【モデル駆動開発】	13	0.08	0	0.08	0	0	0.38	0.15	0	0	0.08	0.23	0.38	0	0	0	0.08	0	0	0	0.08	0	0.08	0.15	0	0.08	0.23	
MILS/HILS/SILS	4	0.25	0.25	0.50	0	0	0.25	0.25	0	0	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0	0	0	0	0	0.25	0	0	0	0.50	
保守・更改・派生開発(XDDP)	11	0.18	0.09	0.36	0.09	0.09	0.18	0.09	0	0	0.27	0.09	0	0	0.18	0.09	0.09	0.09	0.09	0.09	0.09	0.09	0	0.18	0.09	0.09	0.36	
CASE	5	0	0	0	0	0	0.4	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.4	
RAD【高速開発】	4	0.5	0.25	0.5	0	0.25	0.25	0.25	0	0	0.25	0.5	0	0	0.25	0.25	0	0.25	0.25	0.25	0.25	0	0	0	0	0	0.25	
IDE【統合開発環境】	1	0	1	1	0	0	1	0	0	0	1	1	0	1	1	0	1	0	0	0	0	0	0	0	0	0	0	
SPL (ソフトウェアプロダクトライン開発)	2	0	0.5	0.5	0	0	0.5	0.5	0	0	0.5	0.5	0	0.5	0.5	0	0.5	0	0	0	0	0	0	0	0	0	0.5	
ソフトウェアフレームワーク	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	
テスト自動化	8	0.38	0	0.5	0	0.13	0	0	0	0	0	0	0.13	0	0.13	0	0	0	0	0	0	0	0	0.13	0.13	0	0.75	
バグ分析 (ODC分析)	4	0	0	0	0	0	0	0	0	0.25	0	0	0.25	0	0	0	0	0	0	0	0	0	0	0	0.25	0	0.25	
形式検証	3	0	0	0	0	0	0.33	0	0	0	0	0	0	0	0	0	0	0	0.33	0	0	0	0	0.33	0	0.33		
モデル検査	5	0	0	0.2	0	0	0.2	0	0	0.2	0	0.2	0	0	0.2	0	0.2	0	0.2	0.2	0.2	0	0	0	0	0	0.6	
IV&V	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	
耐故障性検証 (運用時)	6	0.17	0	0.17	0	0	0.33	0.17	0	0	0.17	0.33	0	0	0	0	0	0	0.17	0	0	0.33	0.17	0	0	0.33		
アシユアランスケース(D-Case)	6	0.17	0	0	0	0.17	0	0	0	0	0.33	0.17	0	0	0	0	0	0	0	0	0	0.17	0.17	0	0	0.17		
セキュリティ	4	0	0	0	0	0	0	0	0	0	0	0	0.25	0	0.25	0.25	0	0	0	0	0	0	0	0.5	0	0	0.25	
リスクアセスメント	2	0.5	0	0	0	0	0	0	0	0	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0.5	0.5	0	0.5	
独自開発・検証ツール	17	0.12	0.06	0.35	0.00	0.12	0.06	0.06	0.00	0.06	0.18	0.12	0.24	0.12	0.06	0.00	0.00	0.06	0.00	0.35	0.06	0.06	0.18	0.00	0.12	0.00	0.12	0.06

## 3.3 適用技術・手法ごとの分析

### 3.3.1 技術・手法の説明

この取組みで適用された技術・手法について3つの情報システム関連のインターネット辞書での掲載状況を表3-53に示す(2015年8月12日現在)。

表 3-53 対象技術・手法に関する各種辞書サイトの取扱い状況

技術・手法	IT用語辞典 <sup>14</sup>	情報システム用語辞典 <sup>15</sup>	WIKIPEDIA <sup>16</sup>
アジャイルソフトウェア開発モデル	○	○	○
ソフトウェアプロトタイプ開発モデル	○	—	○
イテレーション開発モデル【反復型開発モデル】	—	—	○
W(開発)モデル	—	○	—
BPM【ビジネスプロセスマネジメント】	○	○	○
MBSE【モデルベースシステムズエンジニアリング】	—	—	●*
形式仕様記述	—	—	○
HCD【人間中心設計】	—	○	○
UX【ユーザーエクスペリエンス】	○	○	○
MBD【モデルベース開発】	—	—	●*
MDD【モデル駆動開発】	—	—	○
保守・更改・派生開発(XDDP)	—	—	—
MILS/HILS/SILS	—	—	●*
RAD【高速アプリケーション開発】	○	—	○
CASE	○	—	○
IDE【統合開発環境】	○	—	○
SPLE【ソフトウェアプロダクトライン開発】	○	—	●*
ソフトウェアフレームワーク	○	—	○
テスト自動化	—	—	○
バグ分析(ODC分析)	—	—	●*
形式検証	○	—	○
モデル検査	—	—	○
IV&V	—	○	●*
耐故障性検証(運用時)	—	—	—
アシュアランスケース(D-Case)	—	—	●*
セキュリティ	○	—	○
リスクアセスメント	○	—	○

※：日本語版にはないが英語版に掲載されていた。

14 株式会社インセプトが運営する事業で別名 e-Words。http://e-words.jp/

15 アイティメディア株式会社が運営する事業である。http://www.itmedia.co.jp/enterprise/subtop/dictionary/

16 ウィキペディア(英: Wikipedia)は、ウィキメディア財団が運営しているインターネット百科事典。

日本語版: https://ja.wikipedia.org 英語版: https://www.wikipedia.org

### ① アジャイルソフトウェア開発モデル

アジャイル開発は、顧客の要求にしたがって、優先度の高い機能から順に、要求・開発・テスト（・リリース）を短い期間で繰り返しながら、システム全体を構築していくというものである。原則として、事前に開発の詳細な計画は作らず、1～4週間という一定の短い周期で要求・開発・テストを繰り返しながら、動作可能なソフトを作り上げる。

「顧客の参画の度合いが強い」、「動くソフトウェアを成長させながら作る」、「反復・漸進型である」、「人と人のコミュニケーション、コラボレーションを重視する」、「開発前の、要求の固定を前提としない」といった特徴を持ち、代表的な手法には、エクストリーム・プログラミング（XP）、スクラム（Scrum）、ユーザー機能駆動開発（FDD：Feature Driven Development）、LSD（Lean Software Development）などがある。

参考：SEC セミナー「アジャイル開発の現状と今後」他、山下博之

### ② ソフトウェアプロトタイプ開発モデル

ソフトウェアプロトタイプ開発とは、ソフトウェアの開発において、早期に「実動する試作品（プロトタイプ）」による評価を行いながら完成品の開発を進める手法を表す。試作品は、評価すべき項目を明確にして、その評価に有効なソフトウェアとして作成される。これを実際に稼働させることで、課題を早期に洗い出し、開発にフィードバックすることで、最終製品の品質を向上させることを目的とする。

「利用者からみた操作性」のような機能要件や、「新しい実行基盤での性能評価」のような非機能要件など、様々な評価対象に利用される手法である。

参考：「高信頼化ソフトウェア開発のための開発ガイドブック - 予防と検証の事例を中心に -」

IPA/SEC 編 <https://www.ipa.go.jp/files/000005144.pdf>

### ③ イテレーション開発モデル【反復型開発モデル】

イテレーション開発【反復型開発】とは、最終的なシステムの開発に向けて、「一部の機能が動作するシステムを作って検証する」ことを繰り返し、段階的に機能を追加していくことで最終的な対象物の開発を行う手法を表す。一般的なウォーターフォール・モデルの開発に比べ、開発過程で頻繁に動作を確認することで、要求事項への適合性を早い段階で確認しながら開発を進められるメリットがある。

### ④ W（開発）モデル

Andreas Spillner 氏などが唱えたモデルで、改善を加えられ利用されている。V 字型の開発プロセスと同じ型のテストプロセスが並列に実施されることから W モデルと呼ばれる。W モデルでは、開発の上流工程でテスト計画やテスト設計を実施することで、早期に問題を検出することが期待できる。

### ⑤ BPM【ビジネスプロセスマネジメント】

「BPM は、業務のプロセス（手順、役割分担、ルール）を、役割分担している関係者で共有することで、日々の業務の成果を向上させる経営手法です。

この手法は、主に以下の3つから構成されています

1. 業務のプロセスを可視化・設計する手法
2. 業務のプロセスを実行・管理する情報システム
3. 上記に基づく業務改革の活動体系

引用：一般社団法人 日本ビジネスプロセス・マネジメント協会 <http://www.bpm-j.org/bpm/>

## ⑥ MBSE【モデルベースシステムズエンジニアリング】

INCOSE(The International Council on Systems Engineering)によると、システムズエンジニアリングは、システムを成功裏に実現するための複数の分野にまたがるアプローチおよび手段のことである。モデルに基づくシステムズエンジニアリングをMBSE (Model-Based Systems Engineering) という。

参考：西村秀和 総監修，藤倉俊幸 企画・監修「モデルに基づくシステムズエンジニアリング」日経BP社，2015、  
INCOSE，INCOSE SYSTEMS ENGINEERING HANDBOOK:A Guide for System Life Cycle Processes and Activities

## ⑦ 形式仕様記述

形式仕様記述とは、形式手法のひとつで、数学的な記法を用いて厳密な仕様を記述する方法である。ソフトウェア開発の上流工程での成果物である仕様を形式仕様言語でモデル化・記述し、それに基づいた検証を行い、仕様の正しさを証明することができる。

## ⑧ HCD【人間中心設計】

HCD (Human-centered design) とは、情報技術に関する人間工学として提唱 (by Shackel,B.) された言葉。同じような言葉としてユーザーエクスペリエンスデザイン (UXD) やユーザー中心設計 (UCD) がある。

HCDは、従来の「機能や性能中心のシステム企画・開発」に対して、「ユーザーの体験価値を重視したトータルデザイン」を行うシステム設計の考え方である。すなわち、利用者にとってより良い体験を実現しようとする設計手法である。ユニバーサルデザイン、ユーザーエクスペリエンスなどを実現する上で重要となるプロセス・手法である。HCDのプロセスに関する規格としてISO 9241-210:2010が発行されている。

参考：特定非営利活動法人 人間中心設計推進機構(HCD-Net) 活動紹介 鱗原 晴彦 <http://www.ipa.go.jp/files/000003979.pdf>

使いやすさ／利用時品質の考え方 ～開発技術者にとって重要なユーザビリティ、UXとは～ 吉武 良治 <http://www.ipa.go.jp/files/000044154.pdf>

## ⑨ UX【ユーザーエクスペリエンス】

UXは、User eXperienceの略であり、「そのシステムを通して利用者が体験するすべて」を表す。システム開発において実現すべき目標や特性のひとつである。

参考：使いやすさ／利用時品質の考え方 ～開発技術者にとって重要なユーザビリティ、UXとは～ 吉武 良治 <http://www.ipa.go.jp/files/000044154.pdf>

## ⑩ MBD【モデルベース開発】

モデルベース開発 (MBD: Model Based Development) は組込みソフトウェア開発で適用されている開発方法である。制御ソフトウェアや制御対象をシミュレーション可能な詳細度でモデル化し、各工程でシミュレーション検証による確認を行いながら開発を進める。

参考：dSPACE Japan 監修「モデルベース開発—モデリング、プラント・モデル、コントロール・モデル—」日経BP社 2013

西村秀和 総監修，藤倉俊幸 企画・監修「モデルに基づくシステムズエンジニアリング」  
日経BP社 2015

## ⑪ MDD【モデル駆動開発】

MDDはソフトウェア開発の上流工程においてUMLなどを使って抽象的なモデルを作成し、徐々

に詳細化してコードへと落とししていく。最近では自動コード生成やモデル検証なども可能なツールが用意されている。

参考：平成 23 年度モデルベース開発技術部会活動報告書 IPA/SEC

<http://www.ipa.go.jp/files/000026871.pdf>

## ⑫ 保守・更改・派生開発 (XDDP)

新規開発に対峙して派生開発という概念がある。既存のシステムやソフトウェアが関わる新しい環境（異機種間を含むネットワークやモバイル機器などとのコミュニケーションや制度改定にともなる変更など）への適応が多い中で改良保守や差分開発、流用開発などを総称した派生開発が多く領域で行われている。特に、組込みシステムやパッケージ開発では、以前から莫大な開発工数をかけ対応していた。近年は派生開発に対する投資額も限られる中、合理的な開発アプローチなどの必要性から色々な開発方法が出始めている。

その一つに XDDP (eXtreme Derivative Development Process) という派生開発に特化した開発アプローチで、「短納期」や「部分理解」といった派生開発特有の問題に合理的に対応する方法である。XDDP が提供する施策を支えているものに「USDM(Universal Specification Describing Manner)」と「PFD (Process Flow Diagram)」という技法がある。

参考：派生開発推進協議会のHPの「派生開発とは」[http://affordd.jp/about\\_xddp2.shtml](http://affordd.jp/about_xddp2.shtml)

## ⑬ MILS/HILS/SILS

いずれも、組込み・制御系の開発において開発対象システムのシミュレーションを行う方法である。MILS (Model In the Loop Simulation) は、実際のハードウェアを使用せず、モデルのみでシミュレーションを行う。

HILS (Hardware In the Loop Simulation) は、実際のハードウェアを使用する。

SILS (Software In the Loop Simulation) は、C 言語などソースコードを用いたソフトウェアで実行するシミュレーションのことである。

## ⑭ RAD【高速アプリケーション開発】

RAD(Rapid Application Development) とは、情報システム開発において開発期間を短縮する手法の 1 つである。エンドユーザーを含む少人数のチームで一貫して開発を進め、プロトタイプを作成しては評価する、という流れを繰り返し、完成品に近づけていく方法である。

## ⑮ CASE

CASE(Computer Aided Software Engineering) とは、ソフトウェアの設計、開発、保守において、品質、効率の向上のためにソフトウェアツールを利用する開発を表す。そのために利用するツールを CASE ツールと呼ぶ。

CASE ツールには、ソフトウェア開発の全工程を統合する統合 CASE ツールと、上流 CASE ツール（設計支援など）や下流 CASE ツール（テスト支援など）などの開発工程ごとに利用するツールがある。

## ⑯ IDE【統合開発環境】

IDE(Integrated Development Environment) は、統合開発環境とも呼ばれる。

ソフトウェアの開発には、ソースコードを書くためのテキストエディタをはじめ、コンパイラ、リンカ、デバッガなど各種のツールが用いられる。その他にも、開発規模に応じて統合的にソフトウェアの資源を管理するための支援ツールなども利用される。これらの機能を統合した開発環境が IDE

である。

#### ⑰ SPLE【ソフトウェアプロダクトライン開発】

ソフトウェアプロダクトラインとは、米国カーネギーメロン大学ソフトウェア研究所(SEI)で開発・体系化された、製品系列としてのソフトウェアを効率的に、効果的に開発する手法である。製品系列全体の共通性と差異を明示的に捉え、再利用技術に基づき、全体最適の考え方をもちて開発全体目的指向で体系付ける先進的な形態である。

参照：SEC journal 17号、ソフトウェアプロダクトライン開発の概要、岸知二

#### ⑱ ソフトウェアフレームワーク

ソフトウェアコンポーネントをレイヤに分けて配置し、アプリケーションを実行するために必要なコンポーネント統合を行うために必要なセキュリティ・運用管理・通信管理機能を合わせそえた枠組み。

#### ⑲ テスト自動化

「ソフトウェアを使って、テストマネジメント、テスト設計、テスト実行、結果チェックなどのテスト活動の実行や支援をすること。」

引用：ISTQB テスト技術者資格制度用語集ソフトウェアテスト標準用語集 日本語版 Version 2.3.J02

#### ⑳ バグ分析 (ODC 分析)

ソフトウェアの品質向上のため、欠陥の発生や混入の原因などを分析することをバグ分析という。その手法のひとつに、ODC(Orthogonal Defect Classification：直交欠陥分類)による分析がある。ODC分析は、1992年 IBM Watson 研究所により提唱された障害の定量的分析手法。検出された障害に直交した複数の属性を付与し、多角的な視点から分析を行う。重複・冗長性のない属性を定義し、その属性のみに注目しながら分類することにより効率的に分析できることが特徴である。

#### ㉑ 形式検証

検証の基準(検証性質)と検証の対象(設計仕様)の両方について形式仕様言語を用いて記述し、それらをツールに入力することにより、検証対象が検証基準を満たしていることを検証する。

#### ㉒ モデル検査

モデル検査は、システムの振舞いを状態遷移システムとみなし、システムの満たすべき性質を、状態空間の探索により検証する技術である。テストでは実現できない網羅的検査に特徴があり、システム構築の上流工程において、その仕様の妥当性を検証するための形式検証技術のひとつとして注目を集めている。実際に開発現場で用いるためには、開発現場での適用シナリオを想定して、検査対象システムのモデルとその検証したい性質を開発者が適切に定義する必要がある。

参考：SEC journal Vol.10 No.2 Jul. 2014、要件定義プロセスと保守プロセスにおけるモデル検査技術の開発現場への適用 松浦佐江子他、<https://www.ipa.go.jp/files/000039865.pdf>

#### ㉓ IV&V

IV & V (ソフトウェア独立検証と有効性確認：Independent Verification and Validation)は、ソフトウェアの開発組織とは独立した組織が、独立した技術及び開発組織に影響を受けない資金によって、ソフトウェアの課題や問題を洗い出し、潜在するリスクを軽減する活動である。

## ②4 耐故障性検証（運用時）

耐故障性（ハードウェア又はソフトウェア障害にもかかわらず、システム、製品または、構成要素が意図したように運用操作できる度合い）を検証すること。特に、運用時に対しては、運用者や利用者の視点を考慮し、運用計画などにもとづき時間軸での検証を行う。

参考：表憲一、「冗長構成システム（クラウド等）の耐故障性に対する検証技術」、SEC セミナー

2014年9月

## ②5 アシユアランスケース（D-Case）

アシユアランスケースとは、想定している環境下において、システムが正しく動作することを構造化されたドキュメントで表現する方法を表す。テスト結果や検証結果をエビデンスとして、それらを根拠にシステムの安全性を議論し、システム認証者や利用者などに保証する、あるいは確認させるための方法である。一般的にはドキュメントを作成して関係者間の合意性を担保する。例えば、安全性のためのセーフティケースやセキュリティのためのセキュリティケース、ディペンダビリティのためのディペンダビリティケース（D-Case）など総称してアシユアランスケースと呼ぶ。

参考：UK Defence Standard 00-56 Issue 4

## ②6 セキュリティ

情報セキュリティとは、企業の情報システムを取り巻くさまざまな脅威から、情報資産を機密性・完全性・可用性（三大要件）の確保を行いつつ、正常に維持すること。

- ・機密性の確保：情報資産を正当な権利を持った人だけが使用できる状態にしておくこと。
- ・完全性の確保：情報資産が正当な権利を持たない人により変更されていないことを確実にしておくこと。
- ・可用性の確保：情報資産を必要なときに使用できること。

参考：経済産業省 中小企業情報セキュリティ対策促進事業 信頼される安全・安心な企業となるために！ <http://www.jnsa.org/ikusei/>

## ②7 リスクアセスメント

「リスクアセスメントとは、リスク特定、リスク分析及びリスク評価の全般的なプロセスである。リスクは組織レベル又は部門レベルで、プロジェクト、個々の活動又は具体的なリスク別にアセスメントを実施できる。状況の違いによって、異なるツール及び技法を適応することがある。リスクアセスメントは、リスク、その原因、結果、及びその発生確率に関する理解を提供する。」

引用：JIS Q 31010 リスクマネジメント－リスクアセスメント技法

## ②8 独自開発・検証ツール

ある課題を解決するために独自に開発した設計系や検証系のツール。

## 3.4 アンケート

### 3.4.1 アンケート内容

適用事例記述内容からは読み取れない適用事例の背景やその他の効果を把握するために、事例提供者にアンケートを実施した。設定した問は以下のとおりである。

- Q1：この取組み<sup>※</sup>は誰の判断で実行できましたか。
- Q2：この取組みを実施するに当たり所属組織で推進できた、または、承認されたポイントは何だと思えますか（複数回答可）。
- Q3：技術・手法を適用することで事業売上に貢献できましたか？また、その他の貢献があれば、記入願います。  
（その他の貢献は任意記述）。
- Q4：この取組みは品質・コスト・納期のどこに貢献しましたか（複数回答可）。
- Q5：技術・手法を導入する時にどの層からどのような懐疑的な意見がありましたか（複数回答可）。
- Q6：この技術・手法を導入したプロジェクトメンバーへの効果は何ですか（選択肢から複数回答可）。
- モチベーション
  - コミュニケーション
  - 達成感
  - 向上心
  - 待遇
  - その他（                    ）
- Q7：今注目している技術・手法がありましたら、その内容と貴社での取組み状況をお教え願います（任意記述）。

※適用事例の内容を指す

### 3.4.2 アンケート結果

回答数は41件で、全体の71%である（アンケート対象件数は、58件）。設問により、複数回答を可としているものや、回答のなかったものもあるので、設問ごとの回答数の合計が41件とならないものもある。アンケート結果は以下のとおりである。

#### Q1：この取組みは誰の判断で実行できましたか。

全体としては、「経営者」、「統括管理者」、「プロジェクトマネージャ」が平均的に同じ件数であった。設計系と検証系を比べると、設計系において「経営者」の割合がかなり多かった。

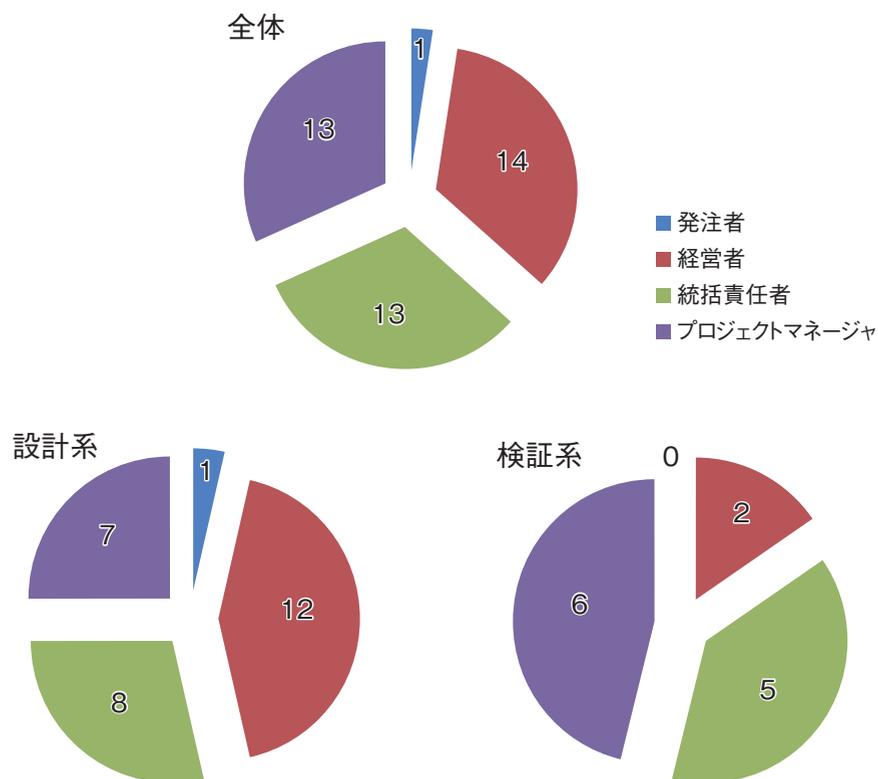


図 3-20 この取組みは誰の判断で実行できたか

Q2：この取組みを実施するに当たり所属組織で推進できた又は、承認されたポイントは何だと思えますか（→どんなポイントで承認されたか）。

「現状の課題解決をしたいから」が28件と圧倒的に多かった。「効果を理解したから」は15件（「定量的に」7件+「定性的に」8件）にとどまっている。効果への見通しは確信できなくても、現状の課題解決を目的にチャレンジしているケースがある。

また、「お客様から要求／期待されたから」が5件あったことも興味ある結果である。

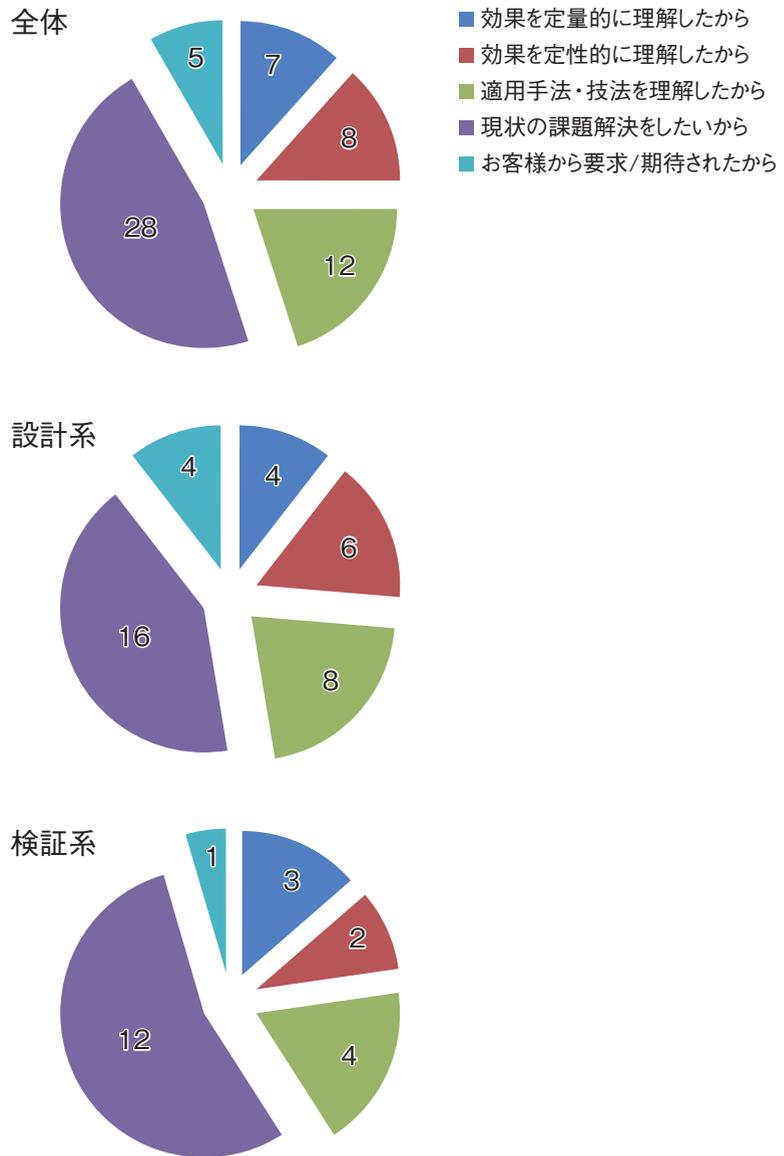


図 3-21 取組みを承認されたポイント

Q3：技術・手法を適用することで事業売上に貢献できましたか？その他の貢献を記入願います（→事業売上への貢献度とその他の貢献度）。

新しい取組みをする場合、費用対効果を定量的に示せると、その後の取組みの継続や普及拡大につなげられる。収集した適用事例の42%（15件 ÷ 38件）が事業売上への貢献ができたと判断していることは、今後の普及促進が進む可能性を示しており、大いに期待できる（図3-22）。また、事業売上だけでなく、「品質向上」や「受注機会拡大」にも貢献している。このことは、コスト削減を主眼とした「守りの投資」ではなく、「攻めの投資」としての効果が出ている証でもある（図3-23）。

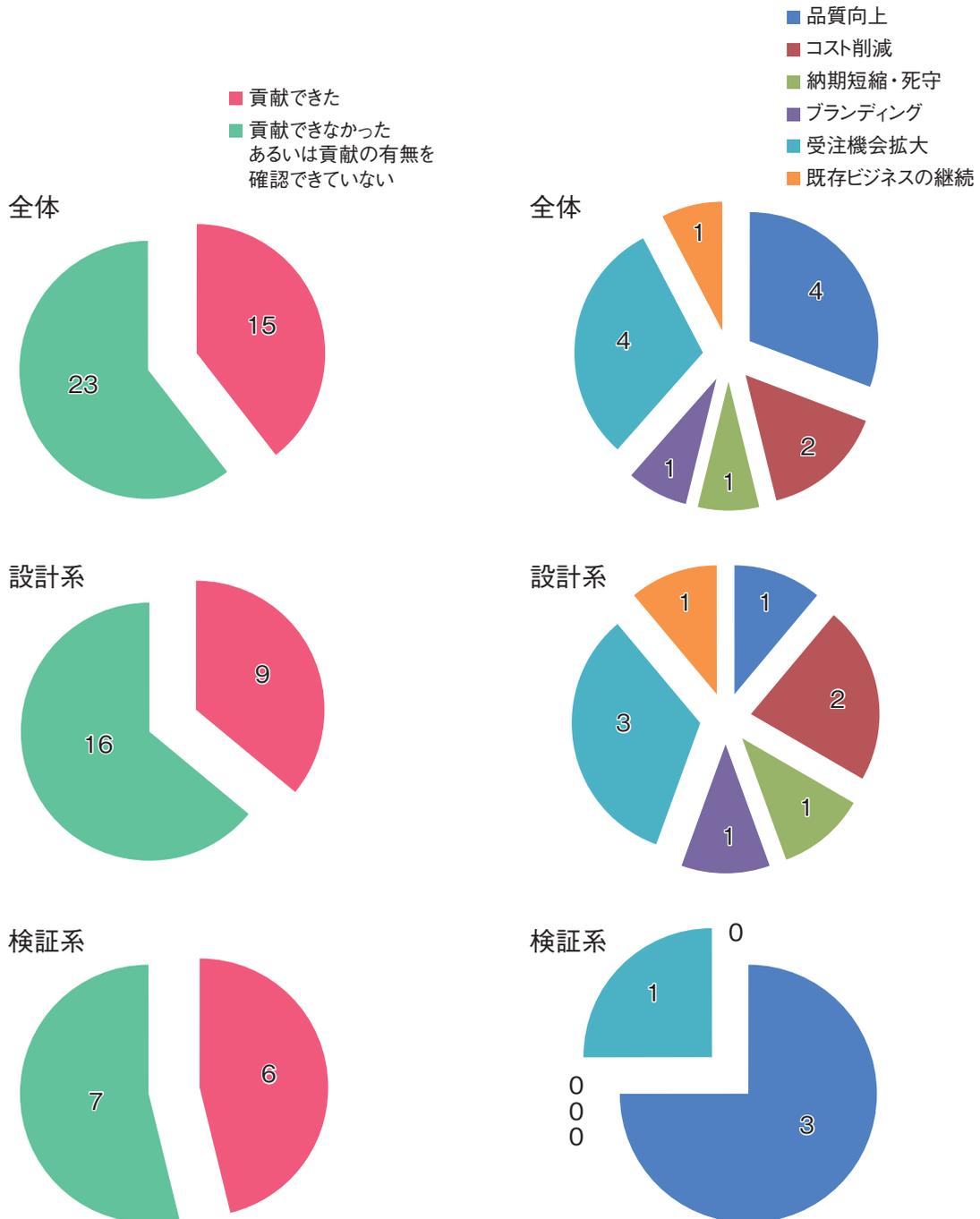


図 3-22 技術・手法の適用で事業売上に貢献できたか

図 3-23 技術・手法の適用での貢献（事業売上以外）

Q4：この取組みは品質・コスト・納期のどこに貢献しましたか（→何に貢献できたか）。

アンケートは適用事例の分析前に実施しており、アンケートの結果と分析結果は、ほぼ同様であった。

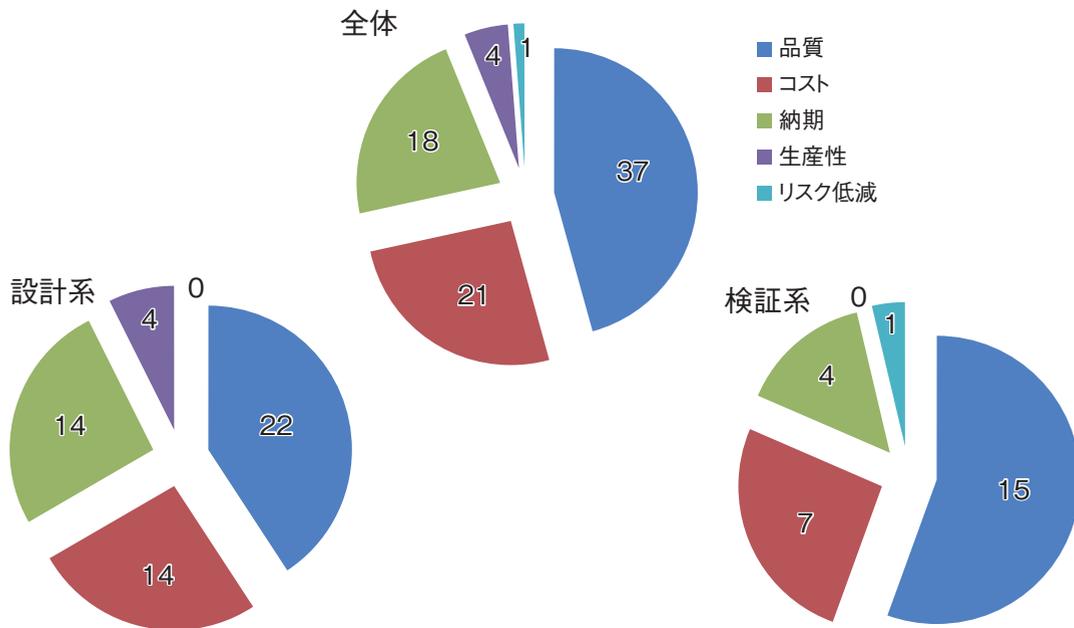


図 3-24 取組み結果として貢献したこと

Q5：技術・手法を導入する時にどの層からどのような懐疑的な意見がありましたか（→この取組みに対して懐疑的であったか）。

(1) 取組に対する懐疑的な意見の有無

収集した適用事例では、多くが先進的な技術・手法のため導入時や普及促進時に懐疑的な意見が社内外から出ていた（53%）。

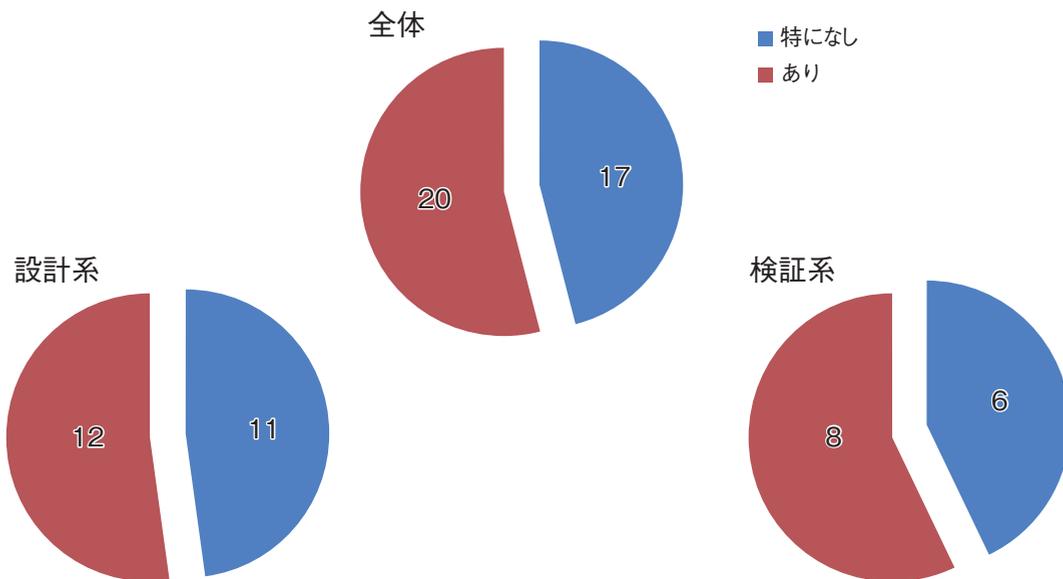


図 3-25 取組みに対して懐疑的な意見があったか

(2) 取組みに懐疑的な人とその理由

取組みに懐疑的な人の上位は、「担当者」、「ベテラン技術者」、「プロジェクトマネージャ」、「プロジェクトリーダー」である。Q1の回答結果（この取組みを判断した人は、「経営者」、「統括責任者」、「プロジェクトマネージャ」）を考えると、企業内における幹部層が取組みに熱心だが、現場サイドに抵抗感があるということが見えてくる。

懐疑的な理由の上位は、「余計な手間がかかる」、「必要性のコンセンサスが得られない」である。取組みを円滑に進めるためには、それぞれの現場でこの様な条件についての配慮も重要であると考えられる。

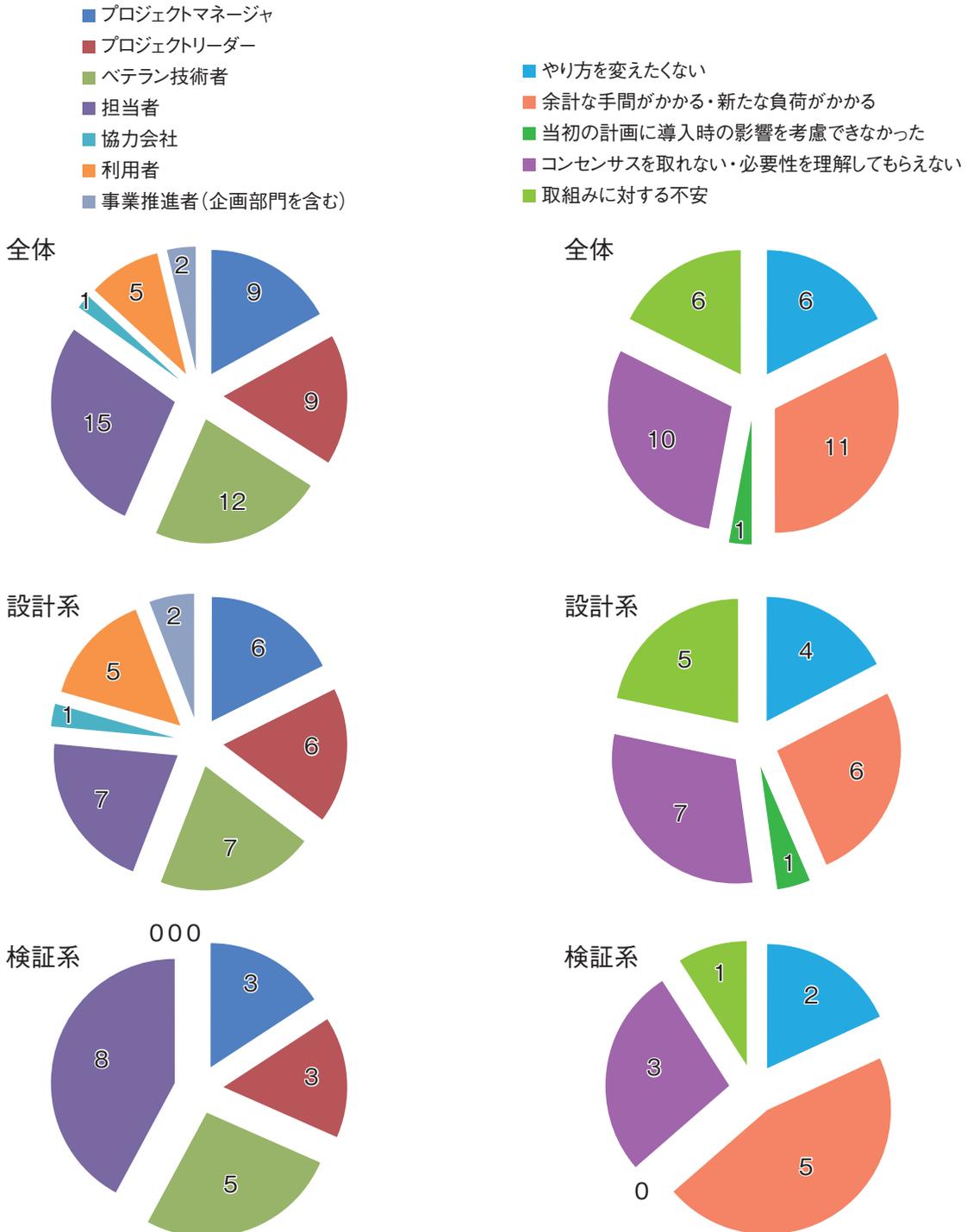


図 3-26 懐疑的な人

図 3-27 懐疑的な理由

Q6：この技術・手法を導入したプロジェクトメンバーへの効果は何ですか（→プロジェクトメンバーへの波及効果は何か）。

設問の5つ選択肢のうち、「待遇」はゼロであった。その他4つの「モチベーション」、「コミュニケーション」、「達成感」、「向上心」は、ほぼ同数の結果が得られた。選択肢以外に「時間的・精神的な余裕の確保」、「人材育成」、「技術力の向上」、「使命感」、「安心感」が追記されたことは、想定以上に色々な波及効果があったことを示している。

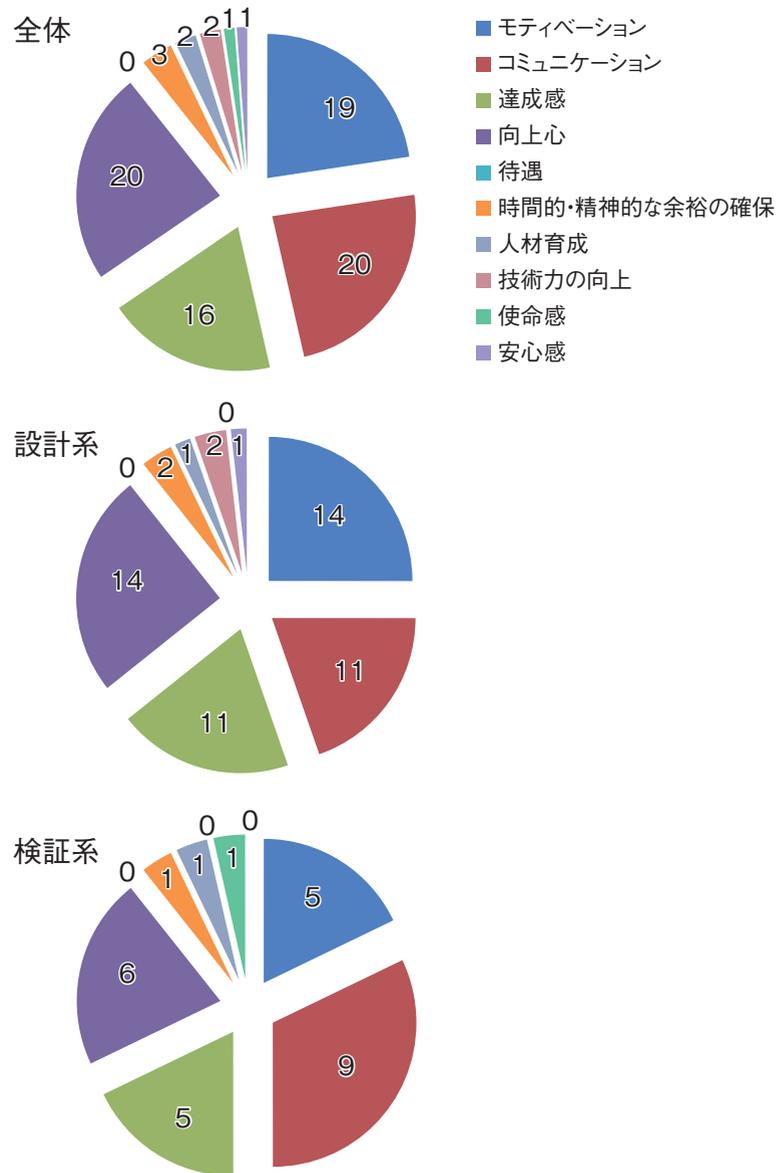


図 3-28 プロジェクトメンバーへの波及効果の内容

**Q7：今後注目する技法・手法は何か。**

この設問は 19 件から回答を得られた。今後注目する技術・手法として列挙されたものは以下のとおりである。

- BPM ツール
- BR・BRMS
- 業務可視化
- デザインシンキング
- アジャイル UX
- UX 定量測定技術
- 新価値創造技術
- 仮想検証技術
- GSN
- セキュリティケース
- デザイン手法
- リファクタリング（大規模ソフトウェア）
- テストスクリプトの自動生成
- バイナリーコード自動解析
- 並列化技術

## 課題解決のヒントになる 適用事例や適用技術・ 手法を紐付け

..... 126

- 4.1 読者が置かれている環境をナビゲーションシートに可能な  
範囲で記入（ステップⅠ） ..... 128
- 4.2 期待する効果の項目の洗い出し ..... 130
- 4.3 適用事例分類表から適用領域①と適用事例②で抽出（ステップⅡ） 131
- 4.4 さらに適用工程で抽出（ステップⅢ） ..... 133
- 4.5 さらに期待する効果項目で抽出（ステップⅣ） ..... 134
- 4.6 さらに特に期待する効果項目で絞込  
（ステップⅣで該当する事例が多すぎた場合ーステップⅤ）..... 137
- 4.7 先進的な技術・手法の適用 ..... 140

# 4

## 課題解決のヒントになる適用事例や適用技術・手法を紐付け

「1.6章 本書の活用方法」で説明した活用方法で「課題解決のヒントになる適用事例や適用技術・手法を探す」ために、以下の2つの方法がある。

### 1 直接、適用事例分類表（付録2）を参照し、各属性からおおよその見当をつけて事例を探す方法

※適用事例分類表は、IPA/SECのウェブサイト（<http://www.ipa.go.jp/sec/publish/tn16-001.html>）からダウンロード可能



### 2 ナビゲーションシート（図4-2参照）を使用して、適用事例分類表（付録2）から事例を探す方法

本章では上記2について詳細を記述する。

以下のステップIからステップVで課題解決のヒントになる適用事例へ紐付けし、適用している技術・手法の導入をヒントに自社の適用の参考にする（図4-1）。

- ステップI： 読者が置かれている環境をナビゲーションシートに可能な範囲で記入
- ステップII： 適用事例分類表（付録2）から「適用領域①」と「適用領域②」で抽出
- ステップIII： さらに「適用工程」で抽出
- ステップIV： さらに期待する「効果項目」で抽出
- ステップV： さらに特に期待する「効果項目」で絞込（ステップIVで該当する事例が多すぎた場合）

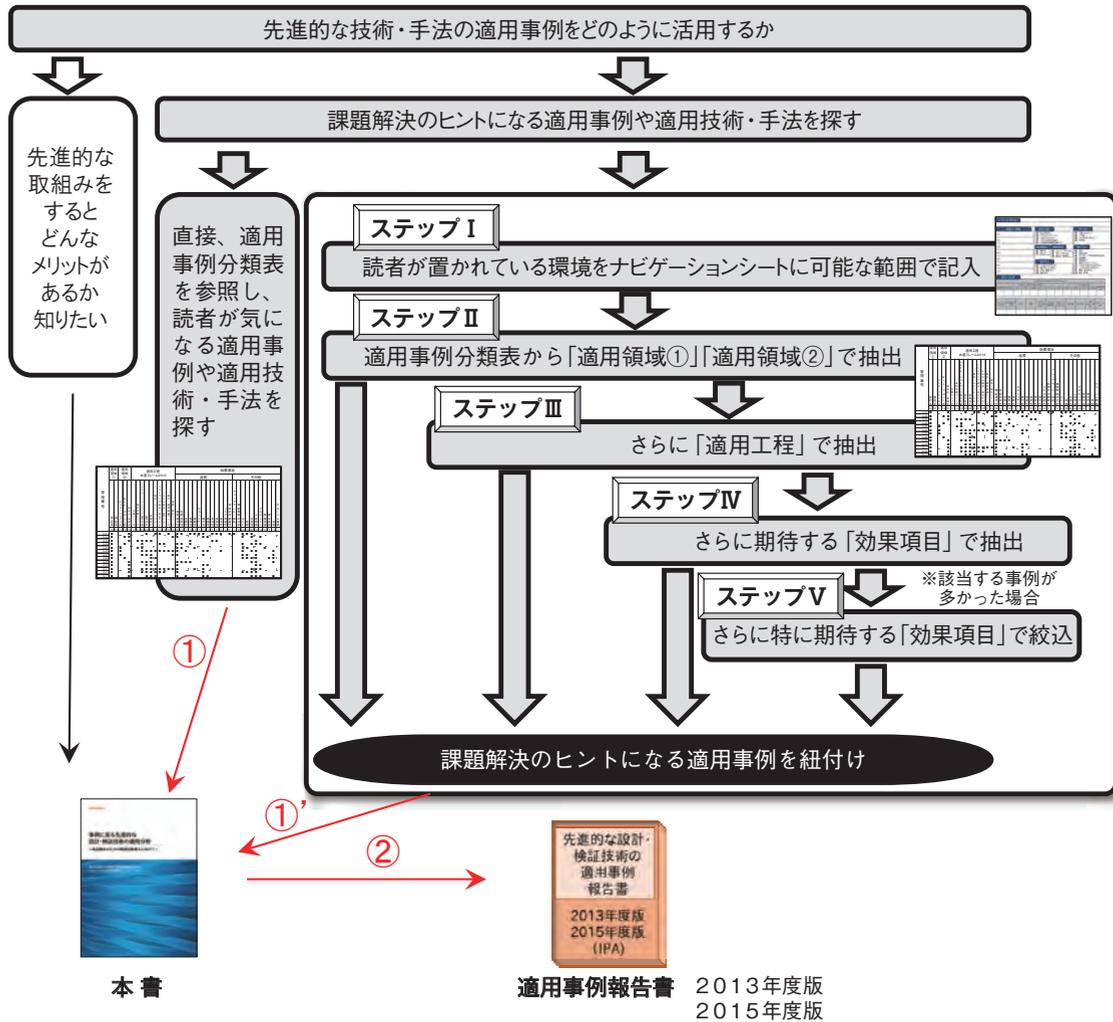


図 4-1 課題解決のヒントになる適用事例の紐付けの仕方

## 4.1 読者が置かれている環境をナビゲーションシートに可能な範囲で記入（ステップⅠ）

図4-2のナビゲーションシートにシステムやプロジェクトの「期待する効果」の項目を除いて情報を入力する。

システム名(案件名)												
課題			読者の属性				測定方法					
①	<input type="checkbox"/> 統括責任者 <input type="checkbox"/> プロジェクトマネージャ <input type="checkbox"/> プロジェクトリーダー <input type="checkbox"/> システム開発担当者 <input type="checkbox"/> 大学や研究機関の研究者				<input type="checkbox"/> 工数(具体的に ) <input type="checkbox"/> バグ数 <input type="checkbox"/> テスト関連(具体的に ) <input type="checkbox"/> その他 ( )							
②												
③	適用領域①		適用領域②		適用工程							
④	<input type="checkbox"/> 設計系 <input type="checkbox"/> 検証系		<input type="checkbox"/> エンタプライズ系 <input type="checkbox"/> Web・フロント系 <input type="checkbox"/> 組込み・制御系		<input type="checkbox"/> 企画 <input type="checkbox"/> 要件定義 <input type="checkbox"/> システム・ソフトウェア要求定義 <input type="checkbox"/> 基本設計 <input type="checkbox"/> 詳細設計 <input type="checkbox"/> 製作・ユニットテスト <input type="checkbox"/> ソフトウェア結合・総合テスト <input type="checkbox"/> システム結合・総合テスト <input type="checkbox"/> 移行・運用準備 <input type="checkbox"/> 運用・保守							
⑤												
⑥												
⑦												
期待する効果												
システム及びソフトウェア品質モデル(JISX25010:2013)												
製品品質							利用時の品質					
機能適合性	性能効率性	互換性	使用性	信頼性	セキュリティ	保守性	移植性	有効性	効率性	満足性	リスク回避性	利用状況網羅性
アシュアランス(保証)	障害原因の分析	コスト	納期	生産性(対応時間短縮)	人材育成意識改革	プロジェクトマネジメント	見積支援	普及促進	体制(強化・再構築)	グローバル展開		

図4-2 ナビゲーションシート

実際に、入力したイメージが図 4-3 である。例として、「自動運転のためのエンジン制御システムの開発」とした。実際に入力した部分は赤色の部分である。

システム名(案件名)					
自動運転のためのエンジン制御システムの開発					
課題	読者の属性	測定方法			
①機能適合漏れの防止	<input type="checkbox"/> 統括責任者 <input checked="" type="checkbox"/> プロジェクトマネージャ <input type="checkbox"/> プロジェクトリーダー <input type="checkbox"/> システム開発担当者 <input type="checkbox"/> 大学や研究機関の研究者	<input checked="" type="checkbox"/> 工数(具体的に ) <input checked="" type="checkbox"/> バグ数 <input type="checkbox"/> テスト関連(具体的に ) <input type="checkbox"/> その他 ( )			
②派生開発に工数がかかっている					
③製品毎に平行して開発している	<div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <th>適用領域①</th> <th>適用領域②</th> </div> </div> <input checked="" type="checkbox"/> 設計系 <input type="checkbox"/> エンタプライズ系 <input type="checkbox"/> 検証系 <input type="checkbox"/> Web・フロント系 <input checked="" type="checkbox"/> 組込み・制御系	適用領域①	適用領域②	<th>適用工程</th> <input type="checkbox"/> 企画 <input type="checkbox"/> 要件定義 <input type="checkbox"/> システム・ソフトウェア要求定義 <input checked="" type="checkbox"/> 基本設計 <input checked="" type="checkbox"/> 詳細設計 <input type="checkbox"/> 製作・ユニットテスト <input type="checkbox"/> ソフトウェア結合・総合テスト <input type="checkbox"/> システム結合・総合テスト <input type="checkbox"/> 移行・運用準備 <input type="checkbox"/> 運用・保守	適用工程
④ISO 26262に適合しているか					
⑤	<th>適用度合</th> <input type="checkbox"/> パイロットプロジェクト <input type="checkbox"/> 他プロジェクト適用 <input type="checkbox"/> 部内・製品適用 <input type="checkbox"/> 全社適用	適用度合			
⑥					
⑦					

図 4-3 ナビゲーションシート (例)

## 4.2 期待する効果の項目の洗い出し

次に、ナビゲーションシートの「期待する効果」の項目を記載する。記載にあたっては、以下の2通りの方法が想定される。

- ① 入力した「課題」の内容を元に、本書で適用している分類軸の「課題」、「効果」、「今後の取組」の分類項目に該当するものに当てはめて、「期待する効果」の対応する項目を埋める。(2.2.2.4章「課題」、「効果」、「今後の取組」の分類項目を参照)

洗い出した課題の品質に関しては、システムとソフトウェアに関する品質定義と品質評価し、枠組みを定める国際規格ISO/IEC 25000シリーズ Systems and software Quality Requirements and Evaluation (SQuaRE) の品質モデルの品質特性及び副特性から紐付ける。詳しくは、IPA発行の「つながる世界のソフトウェア品質ガイド」にある「SQuaRE 品質モデル活用リファレンス編」を参考にすることができる。

ただし、入力した「課題」に該当する項目がない場合もある。その場合は、今回の適用事例に対象とするものはない。

- ② 入力した「課題」の内容を元に、適用している分類軸の「課題」や「効果」の項目に該当するものに当てはめることができない場合、「期待する効果の項目」を優先順に選ぶ。

ステップIで入力した「課題」の内容を元に、本書で適用している分類軸の「課題」、「効果」、「今後の取組」の分類項目に当てはめた。図4-4の丸で囲まれた数字は、ステップIのナビゲーションシートに記載されている「課題」の番号である。

システム及びソフトウェアの品質モデル(JIS X 25010:2013)												
製品品質							利用時の品質					
機能適合性	性能効率性	互換性	使用性	信頼性	セキュリティ	保守性	移植性	有効性	効率性	満足性	リスク回避性	利用状況網羅性
①		③				②				①		
アシュアランス(保証)	障害原因の分析	コスト	納期	生産性 (対応時間短縮)	人材育成 意識改革	プロジェクト マネジメント	見積支援	普及促進	体制 (強化・再構築)	グローバル 展開		
④		②							③			

図 4-4 課題を分類軸の項目にマッピング (例)







## 4.5 さらに期待する効果項目で抽出（ステップⅣ）

ステップⅢの結果を元に以下の条件で更に抽出する。

### 抽出条件

期待する効果が、アシュアランス（保証）、コスト、体制（強化・再構築）、品質（機能整合性、互換性、保守性、満足性）のいずれかを満たすこと（図 4-4 を参照）

その結果が、表 4-4 である。抽出条件を満たす部分は、赤色の四角（■）で、両方を満たす適用事例は、事例参照番号と四角（■）を青色で表現した。該当する適用事例は 13 件である。

### 例) 期待する効果の該当数が 4 つの場合

- ・適用事例は A-13
- ・適用技術・手法は、MDD【モデル駆動開発】と SPLE【ソフトウェアプロダクトライン開発】

### 例) 期待する効果の該当数が 3 つの場合

- ・適用事例は A-9
- ・適用技術・手法は、SPLE【ソフトウェアプロダクトライン開発】
  
- ・適用事例は A-12
- ・適用技術・手法は、MBSE【モデルベースシステムズエンジニアリング】と MBD【モデルベース開発】と MILS/HILS/SILS
  
- ・適用事例は 15-A-12
- ・適用技術・手法は、イテレーション開発モデルと MBD【モデルベース開発】と保守・更改・派生開発と SPLE【ソフトウェアプロダクトライン開発】とモデル検査と独自開発・検証ツール
  
- ・適用事例は 15-A-17
- ・適用技術・手法は、形式仕様記述と MBD【モデルベース開発】









課題解決のヒントになる適用事例は、表 4-7 のステップⅡからステップⅤの手順で対象が絞りこまれている。

表 4-7 抽出結果（ステップⅡからステップⅤの経緯：対象の適用事例の参照番号）

No.	Ⅱの結果	Ⅲの結果	Ⅳの結果	Ⅴの結果
1	A-1	A-3	A-9	A-10
2	A-2	A-9	A-10	A-11
3	A-3	A-10	A-11	A-12
4	A-9	A-11	A-12	A-13
5	A-10	A-12	A-13	
6	A-11	A-13	15-A-9	
7	A-12	15-A-9	15-A-10	
8	A-13	15-A-10	15-A-12	
9	15-A-9	15-A-12	15-A-13	
10	15-A-10	15-A-13	15-A-15	
11	15-A-11	15-A-15	15-A-17	
12	15-A-12	15-A-17	15-A-19	
13	15-A-13	15-A-19	15-A-20	
14	15-A-14	15-A-20		
15	15-A-15			
16	15-A-16			
17	15-A-17			
18	15-A-18			
19	15-A-19			
20	15-A-20			

## 4.7 先進的な技術・手法の適用

前述の結果、置かれている環境から品質特性の「機能適合性」と「保守性」の課題を解決した事例は4つ、技術・手法は7つに紐付けることができた。この結果を元に以下のように、本書と適用事例報告書（2013年度版と2015年度版）を活用する。それぞれの活用時のイメージを一部示す。

### 本書を活用【図4-1の①及び①'】する場合

- 置かれている環境の設計系（適用領域①）と組込み・制御系（適用領域②）での取組み状況を参照する。  
設計系の場合は、3.2.1.1章を参照する。  
組込み・制御系の場合は、3.2.2.3章を参照する。
- 対象の4つの事例の概要は、「7章 適用事例の概要」を参照する。
- 対象の7つの技術・手法の情報は次の情報を参照する。  
適用技術・手法別の適用結果などの状況は、3.2.3章を参照する（表4-8、表4-9参照）。  
適用技術・手法の分析は、3.3章を参照する。

対象の7つの技術・手法

- ① MBSE【モデルベースシステムズエンジニアリング】（図4-6参照）
- ② 形式仕様記述
- ③ MBD【モデルベース開発】
- ④ MDD【モデル駆動開発】
- ⑤ MILS/HILS/SILS
- ⑥ SPLE【ソフトウェアプロダクトライン開発】
- ⑦ アシユアランスケース（D-Case）

### 適用事例報告書を活用【図4-1の②】する場合

- 対象の4つの事例を精読する。

[参考]

表 4-8 対象の事例の適用する技術・手法の効果発揮度合

技術・手法 \ テーマ	適用事例数	アシュアランス(保証)		障害原因の分析		コスト		納期		(対応時間短縮性)		人材育成		意識改革		プロネジメント		見積支援		普及促進		(強化・再構築体制)		グローバル展開		
MBSE【モデルベースシステムズエンジニアリング】	8	0.13	0.13	0.75	0.38	0.75	0.50	0.38	0.00	0.00	0.13	0.00	0.13	0.00	0.00	0.13	0.00	0.00	0.13	0.00	0.13	0.00	0.00	0.13	0.00	0.00
形式仕様記述	7	0.43	0.14	0.71	0.29	1.00	0.29	0.29	0.00	0.00	0.14	0.00	0.14	0.00	0.00	0.14	0.00	0.00	0.14	0.00	0.14	0.00	0.00	0.14	0.00	0.00
MBD【モデルベース開発】	13	0.15	0.15	0.69	0.31	0.69	0.31	0.23	0.00	0.15	0.08	0.00	0.15	0.00	0.08	0.00	0.00	0.15	0.00	0.08	0.00	0.00	0.08	0.00	0.00	0.00
MDD【モデル駆動開発】	4	0.25	0.00	0.50	0.50	1.00	0.50	0.50	0.00	0.00	0.50	0.00	0.00	0.00	0.50	0.00	0.00	0.00	0.50	0.00	0.00	0.00	0.50	0.00	0.00	0.00
MILS/HILS/SILS	5	0.00	0.00	0.80	0.20	1.00	0.20	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
SPLE【ソフトウェアプロダクトライン開発】	3	0.00	0.00	0.33	0.33	1.00	0.67	0.33	0.00	0.33	0.00	0.33	0.00	0.33	0.00	1.00	0.00	0.33	0.00	1.00	0.00	0.00	1.00	0.00	0.00	0.00
アシュアランスケース(D-Case)	6	0.50	0.00	0.33	0.17	0.33	0.50	0.50	0.17	0.17	0.00	0.17	0.00	0.17	0.00	0.50	0.17	0.17	0.00	0.50	0.17	0.00	0.50	0.17	0.00	0.00

表 4-9 対象の事例の適用する技術・手法の効果発揮度合 (品質特性)

技術・手法 \ テーマ (品質特性)	適用事例数	システム及びソフトウェア品質モデル (JIS X 25010:2013)												
		製品品質							利用時の品質					
		機能適合性	性能効率性	互換性	使用性	信頼性	セキュリティ	保守性	移植性	有効性	効率性	満足性	リスク回避性	利用状況網羅性
MBSE【モデルベースシステムズエンジニアリング】	8	0.75	0.13	0.13	0.13	0.25	0.00	0.88	0.25	0.13	0.13	0.13	0.00	0.25
形式仕様記述	7	1.00	0.00	0.14	0.14	0.29	0.00	0.43	0.14	0.14	0.14	0.14	0.00	0.14
MBD【モデルベース開発】	13	0.62	0.23	0.08	0.08	0.31	0.00	0.54	0.23	0.08	0.00	0.00	0.08	
MDD【モデル駆動開発】	4	1.00	0.00	0.50	0.25	0.00	0.00	0.50	0.50	0.25	0.25	0.25	0.00	0.25
MILS/HILS/SILS	5	0.40	0.40	0.00	0.00	0.20	0.00	0.40	0.20	0.20	0.00	0.00	0.00	
SPLE【ソフトウェアプロダクトライン開発】	3	0.67	0.67	0.67	0.33	0.33	0.00	0.67	0.67	0.00	0.00	0.00	0.00	
アシュアランスケース(D-Case)	6	0.67	0.33	0.17	0.50	0.67	0.33	0.83	0.33	0.17	0.17	0.17	0.17	



## 情報サービス産業における 情報技術マップに関する 調査報告

..... 144

5.1 調査方法と分析の概要 .....	148
5.2 2015年度調査 .....	149
5.3 回答状況 .....	149
5.4 調査結果 .....	150

# 5

## 情報サービス産業における情報技術マップに関する調査報告 (JISA発行)

情報サービス産業における要素技術は、一般社団法人 情報サービス産業協会（以降、JISA）の技術強化委員会の情報技術マップ WG で、「情報サービス産業における技術マップに関する調査報告」としてまとめている。JISA では、2004 年（平成 16 年）から「情報サービス産業における情報技術マップに関する調査報告」を、以下のような目的で続けている。

「本技術調査は、技術者へのアンケート調査を通じて、現状の技術への取組み状況、具体的には、技術を利用した実績や今後の着手意向を調査分析することで、情報サービス産業界としての現状と今後の方向性を明らかにすることを目的としている。本調査は、外部専門家による、いわゆる技術トレンドや未来予測といった動向分析とは異なり、現場の技術者の生の声を反映し、情報サービス産業会の実態と今後を明示したものである。」[平成 26 年度 情報サービス産業における情報技術マップに関する調査報告]より引用

この報告書では、IT ディレクトリ（情報サービス産業市場を俯瞰する上で必要と考えられる要素技術をピックアップし、整理・体系化したもの）を 14 のカテゴリに分け、構造化している。この中で黒塗りの「C. システム連携とミドルウェア」、「K. 開発環境・開発ツール」、「L. 開発手法・開発プロセス」の部分が、先進的な設計・検証の適用事例で取り組まれている技術・手法の対象範囲である（図 5-1）。

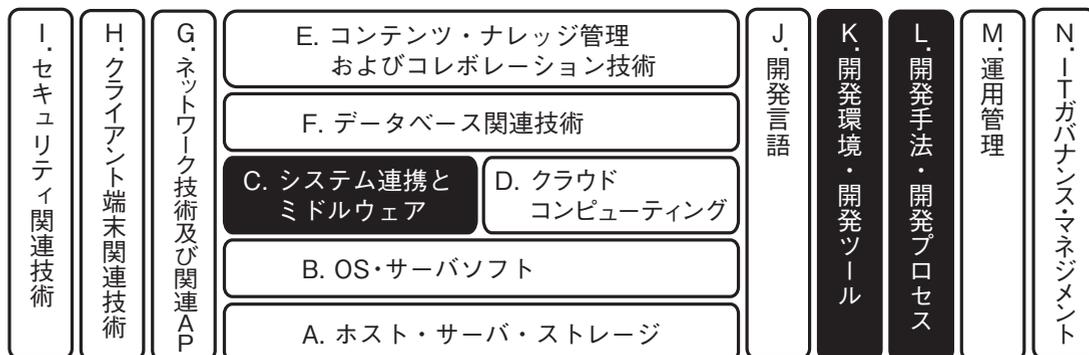


図 5-1 2014 年度版 IT ディレクトリの構造

表 5-1 に上記対象となる 3 つのカテゴリごとに要素技術の項目と内容を示す。また、本書で対象とした技術・手法との重なり具合を○（合致）、△（一部合致）、×（合致せず）で示す。

表 5-1 JISA 情報技術マップと本書の技術・手法の項目比較

C. システム連携とミドルウェア			
定義	異なるシステム／サービス間のデータやプロセスの連携、Web アプリ開発に関するミドルウェア		本書の技術・手法との合致
1.	商用Webアプリケーションサーバ (Oracle Weblogic Server、IBM Websphere Application Server等)	ビジネス・ロジックなどを実装したアプリケーションを実行することを専門とする製品。Oracle WebLogic、IBM Websphere など。	×
2.	オープンソースアプリケーションサーバ (JBoss 等)	Apache Tomcat、JBoss アプリケーションサーバなど、オープンソースのアプリケーションサーバ。	×
3.	Web サービス／メッセージ通信 (Web-API、XML メッセージング、OSGi、MGTT 等)	システム／サービス間の情報を連携する技術。SOAP、REST、XML-RPC といった XML 形式のプロトコルを用いて、ソフトウェア間でメッセージ (情報) を交換する技術。また M2M/IoT 分野等での情報連携、メッセージ通信を行う基盤。最近では IoT も含めて Web サービスとして Web-API で連携することが多い。	×
4.	ESB (Enterprise Service Bus) EAI (Enterprise Application Integration)	企業内の複数のシステムを連携させ、データやプロセスを統合させること、あるいはそのためのミドルウェアの総称。ESB は、データやサービスが仮想的な通信路を流れるバス型の構造になっており、ハブ&スポーク型のように一ヶ所に通信や処理が集中することが無い方式で、SOA に基づくシステムでよく利用される。 EAI はデータベースや業務アプリケーションが持つデータ形式を変換し、相互流通可能とする。ハブ&スポーク構造でハブとなる EAI を介してアプリケーションが接続される。	×
5.	管理系ミドルウェア	アプリケーション連携／統合で必要となるプロセス、マスターデータ、ビジネスルールなどの管理機能。 ・BPM (Business Process Management) : ビジネスプロセスの統合、制御、自動化 ・BAM (Business Activity Monitoring) : ビジネスプロセスの稼働状況の監視、KPI 分析 ・BRMS : ビジネスルールを定義し、それを解釈・自動実行するツール (ILOG Jrules、JBoss Enterprise BRMS、Progress Corticon 等) ・マスターデータマネジメント : 散在するマスターデータの一元管理	△ (BPM のみ対象)

K. 開発環境・開発ツール			
定義	システム・ソフトウェアの開発者が利用する設計技法、ツール、開発環境		本書の技術・手法との合致
1.	Java EE (Java Platform, Enterprise Edition)	Java Servlet, JSP 等のサーバサイドの Java アプリケーションを開発するために必要となる機能を提供する開発・実行環境。	×
2.	.NET Framework	Microsoft 社が開発・提供するアプリケーション開発・実行環境。	×
3.	Web アプリケーションフレームワーク (Struts、Spring、Ruby on Rails 等)	Web アプリケーションを開発する上で一般的に必要な機能を有する基盤ソフトウェア。Struts や Spring Framework, Ruby on Rails 等のオープンソースのフレームワークが多数存在する。	×
4.	クライアント MVC フレームワーク (AngularJS, Backbone.js 等)	JavaScript を利用したリッチなクライアントアプリケーション (Single-page Application) を開発するための基盤ソフトウェア。	×

5.	モバイル開発フレームワーク (PhoneGap/Cordova、Titanium Mobile、Sencha Touch 等)	スマートフォン向けのアプリケーションを開発するための基盤ソフトウェア。Cordova、Titanium Mobile、Sencha Touch 等のライブラリ・フレームワークが存在する。	×
6.	テスト支援/自動化ツール	テストにおける各種作業 (テスト設計、テストコード作成、テスト実行等) を支援・自動化するためのツール。GUI 操作を記録し、自動実行するツールや、テスト設計情報からテストコードを自動生成するツール等がある。	○
7.	コード自動生成ツール (GeneXus、WebPerformer 等)	データ項目や画面、ビジネスルール等の設計情報からソースコードを自動生成するツール。	○
8.	集中型構成管理ツール (Subversion、CVS、Visual Source Safe 等)	ソースコードや設計文書等の管理 (バージョン管理、変更管理等) を行うためのソフトウェアで、単一のリポジトリでプロジェクト成果物を集中管理する方式をとる。CVS、Subversion、Visual Source Safe 等がある。	○
9.	分散型構成管理ツール (Git、GitHub、Mercurial 等)	ソースコードや設計文書等の管理 (バージョン管理、変更管理等) を行うためのソフトウェアで、複数のリポジトリでコピーを分散して保持・管理する方式をとる。Git、GitHub、Mercurial 等がある。	×
10.	課題管理ツール (Redmine、Trac 等)	Redmine、Trac などのプロジェクトの各種課題を追跡・状態管理するためのソフトウェア。Issue Tracking System とも呼ばれる。	○
11.	Continuous Integration ツール (Hudson、Jenkins、CruiseControl 等)	ソフトウェアのビルド・テストを定期的に自動実行するためのソフトウェア。継続的インテグレーションとも呼ばれる。	×
12.	レガシー・マイグレーション	メインフレーム等の旧世代のプラットフォーム上に構築されたアプリケーションを、オープンシステム等の新世代プラットフォームに移行させるためのツールや開発環境。	×

L. 開発手法・開発プロセス			
定義	システム・ソフトウェアの開発プロセス・技法、プロジェクト管理手法	本書の技術・手法との合致	
1.	アジャイル開発/反復型開発 (XP, Scrum 等)	XP(eXtream Programming) や Scrum 等に代表される、反復型で要件変更対応を重視するプロセス・手法。	○
2.	ウォーターフォール開発	プロジェクトを要件定義、設計、実装、テストと工程分割し順次実施する、計画重視のプロセス・手法。	×
3.	派生開発プロセス (XDDP 等)	コアソフトウェアを流用しながら、新しい要件に対応するソフトウェアを開発する (派生させる) ためのプロセス・手法。その1つとして、XDDP (eXtreme Derivative Development Process) がある。	○
4.	形式手法 (VDM++, SPIN 等)	数学的に厳密に意味付けられた言語を利用して、ソフトウェアおよびハードウェアシステムの仕様記述、開発、検証を行うための技術・手法。	○
5.	UML (Unified Modeling Language)	ソフトウェアの仕様、構造、振る舞いなどを表現するための記述言語。OMG により標準化されている。	×
6.	要求開発・要求管理 (ステークホルダ分析、ゴール指向分析 等)	ビジネス目的に適合したシステムの要求を正しく導き出すための手法・方法論およびその管理プロセス。たとえば、ドメイン分析、シナリオ分析、ゴール指向分析などの要求分析手法がある。	○
7.	プロジェクトマネジメント手法 (PMBOK、PMO の採用)	PMO (Project Management Office) の設置や PMBOK の採用など、プロジェクトマネジメントを体系的に行うための手法。	×

本書の技術・手法の中で JISA の 2014 年度調査対象になっているかどうかを○ (対象)、— (非対象) で表 5-2 に示す。

表 5-2 JISA の情報技術マップと本書の調査対象の対比

技術・手法	JISA の調査対象
アジャイルソフトウェア開発モデル	○
スパイラル開発モデル	—
ソフトウェアプロトタイプ開発モデル	—
イテレーション開発モデル【反復型開発モデル】	○
W (開発) モデル	—
BPM【ビジネスプロセスマネジメント】	○
MBSE【モデルベースシステムズエンジニアリング】	—
形式仕様記述	○
HCD【人間中心設計】	—
UX【ユーザーエクスペリエンス】	—
MBD【モデルベース開発】	—
MDD【モデル駆動開発】	—
保守・更改・派生開発 (XDDP)	○
MILS/HILS/SILS	—
RAD【高速アプリケーション開発】	○
CASE	—
IDE【統合開発環境】	—
SPLE【ソフトウェアプロダクトライン開発】	—
ソフトウェアフレームワーク	—
テスト自動化	○
バグ分析 (ODC 分析)	—
形式検証	○
モデル検査	—
IV&V	—
ソフトウェア構成管理	○
耐故障性検証 (運用時)	—
アシュアランスケース (D-Case)	—
セキュリティ	—
リスクアセスメント	—
独自開発・検証ツール	—

## 5.1 調査方法と分析の概要

JISA が実施したアンケート方法と分析の概要を、以下に抜粋する。

### アンケート実施方法

前回調査時同様、WEB アンケートにより実施した。

### アンケート回答選択肢

各技術要素に対するアンケート回答の選択肢は、2015 年度調査においても 2014 年度と同じ選択肢を用いた。選択肢の内容は以下に示すとおりである。

#### 【今年度の選択肢】

1. この技術の利用実績があり、今後も使っていきたい
2. この技術の利用実績があるが、今後は別技術で代替していく予定である
3. この技術の利用実績は無いが、今後は利用すべきである
4. この技術の利用実績は無く、今後も使う予定なし
5. この技術を知らない、もしくは、深く知らない

現在の利用状況や今後の利用意向に関して選択肢を整理したものを表 5-3 に示す。

表 5-3 アンケート回答選択肢

現在の利用状況	今後の利用意向	選択肢
あり	今後も使っていきたい	1
	今後は別の技術が良い	2
なし	今後は使うべき	3
	今後も使う予定はない	4
この技術を知らない、もしくは深く知らない		5

### 分析の概要

本調査におけるアンケートは、14 カテゴリ 121 項目の技術項目に対する実績、着手意向を問うものであり、回答の選択肢は上記同様、以下の 5 つとなっている。

1. この技術の利用実績があり、今後も使っていきたい
2. この技術の利用実績があるが、今後は別技術で代替していく予定である
3. この技術の利用実績は無いが、今後は利用すべきである
4. この技術の利用実績は無く、今後も使う予定なし
5. この技術を知らない、もしくは、深く知らない

各技術項目の分析においては、上記回答をもとに計算された SI 実績指数 ならびに 着手意向指数 が分析の基本となっている。これは上記回答の 1、2 を実績があるとみなし、また 3 を着手意向ありとみなして指標化したものである。各技術項目に対する SI 実績指数ならびに着手意向指数は、それぞれ以下の計算式によって求められる。

$$\begin{aligned} \text{SI 実績指数} &= (\text{選択肢 1 の回答者数} + \text{選択肢 2 の回答者数}) / \text{認知有効回答者数} \\ \text{着手意向指数} &= \text{選択肢 3 の回答者数} / \text{認知有効回答者数} \end{aligned}$$

ここで、「認知有効回答者数」は、無効回答（解答欄が空欄）ならびに選択肢 5（当該技術を認知しない）を除く、選択肢 1～4 までのいずれかの回答を明示的に行った人数の総和である。

また、技術に対する認知度を以下のように定義する。

$$\begin{aligned} \text{認知度 (\%)} &= (\text{選択肢 1 から 4 までの回答者数}) / \text{有効回答者数 (1 から 5 までの回答者数)} \times 100 \end{aligned}$$

認知度は、無回答を除いた有効回答数（選択肢 1～5 の合計）のうち選択肢 5「この技術を知らない、もしくは、深く知らない」を除いた選択肢 1～4 までの回答者数の割合で表している。認知度とは、「ある要素技術について、情報サービス産業における主要技術として一定以上の知識を有する技術者がどの程度存在するか」を数値化したものを指している。

## 5.2 2015 年度調査

前述のとおり、例年の JISA の情報サービス産業における技術成熟度調査（情報技術マップ調査）では、本書の対象の技術・手法のうち、9 個項目しか対象になっていない。そこで、IPA 用に新規に設問を用意することになった。具体的には、前述の過去の情報技術マップ調査項目とは別に「IPA 取組みの先進的な設計・検証技術」の項目が加えられることとなった。

## 5.3 回答状況

会員企業への調査に関しては、Web システムにより回収を行われた。調査の回答状況は以下のとおりであった。

### 【調査実施状況】

- ① 調査対象：JISA 正会員企業のプロジェクトマネージャ、プロジェクトリーダー、チームリーダーの他、システムエンジニア、プログラマ
- ② 調査期間（調査回答期間）：平成 27 年 12 月 7 日～平成 28 年 1 月 29 日
- ③ 回答社数：2015 年度 37 社（前年度：41 社）
- ④ 回答者数：2015 年度 1,332 名（前年度：1,675 名）

## 5.4 調査結果

対象の技術・手法のSI実績指数と着手意向指数と認知度の結果は以下の表5-4のとおりである。

表 5-4 JISA に依頼した調査の結果

IPA が調査依頼した技術・手法	SI 実績指数	着手意向指数	認知度 (%)
アジャイルソフトウェア開発モデル	0.229	0.522	67.177
スパイラル開発モデル	0.258	0.447	67.747
ソフトウェアプロトタイプ開発モデル	0.337	0.450	70.407
イテレーション開発モデル【反復型開発モデル】	0.154	0.469	49.260
W (開発) モデル	0.308	0.346	41.721
BPM【ビジネスプロセスマネジメント】	0.159	0.489	40.771
MBSE【モデルベースシステムズエンジニアリング】	0.054	0.436	28.747
形式仕様記述	0.154	0.371	28.736
HCD【人間中心設計】	0.101	0.528	32.432
UX【ユーザー・エクスペリエンス】	0.136	0.570	39.705
MBD【モデルベース開発】	0.098	0.417	28.003
MDD【モデル駆動開発】	0.087	0.405	26.617
保守・更改・派生開発 (XDDP)	0.162	0.408	28.714
MILS/HILS/SILS	0.034	0.369	22.094
RAD【高速アプリケーション開発】	0.125	0.516	37.154
CASE	0.235	0.321	39.738
IDE【統合開発環境】	0.636	0.209	61.413
SPLE【ソフトウェアプロダクトライン開発】	0.086	0.455	27.609
ソフトウェアフレームワーク	0.541	0.288	58.590
テスト自動化	0.305	0.565	72.263
バグ分析 (ODC 分析)	0.214	0.609	58.096
形式検証	0.139	0.493	34.584
モデル検査	0.075	0.472	29.441
IV&V	0.032	0.404	22.970
ソフトウェア構成管理	0.552	0.327	63.997
耐故障性検証 (運用時)	0.147	0.526	40.786
アシュアランスケース (D-Case)	0.055	0.427	24.076
セキュリティ	0.692	0.240	78.405
リスクアセスメント	0.544	0.355	69.685

それぞれの指標を元に相関関係を分析した結果が図 5-2 に示されている。

SI 実績指数と着手意向指数の相関 (図 5-2) では、JISA の 10 年間の調査結果から中心点 (過去 10 年の SI 実績指数と着手意向指数それぞれの平均値) を基準に 4 つのステージを定義している。そのステージは、図中に示すとおりである。本書の技術・手法は、研究期に 16 項目、普及期に 7 項目プロットされている。残り 5 項目は、安定期に位置しているが、その中の技術・手法も考え方が日々進化し、新たな方法も出てくるものである。



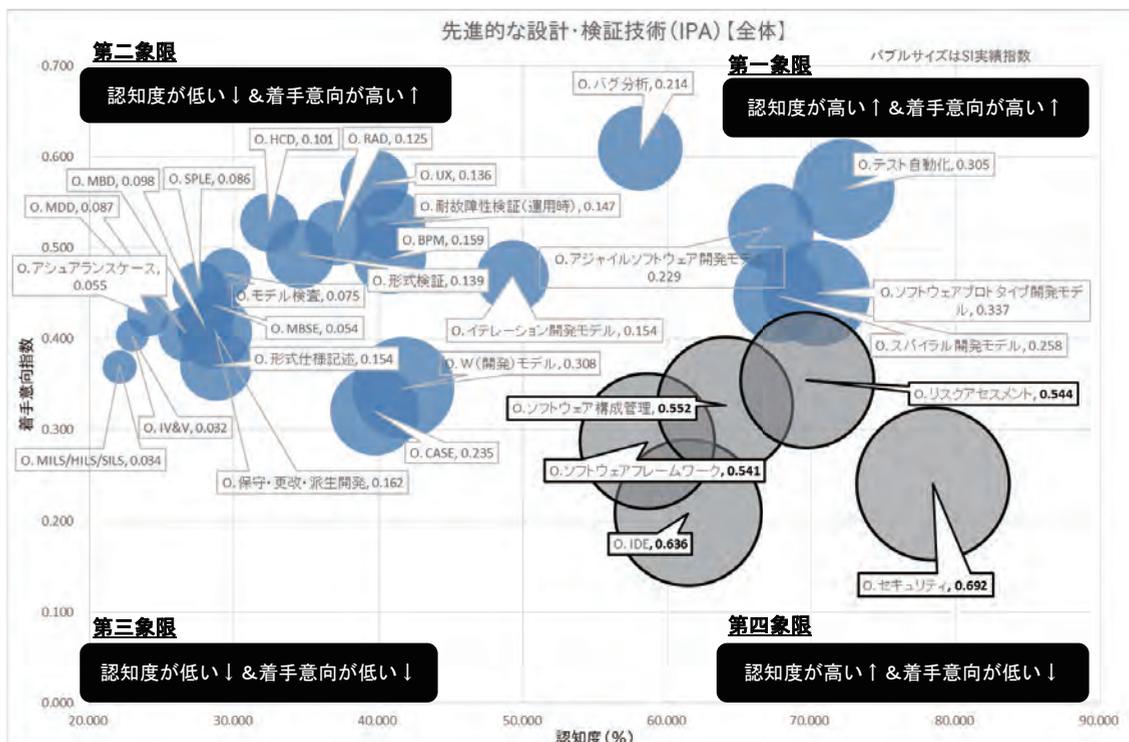


図 5-3 認知度マップ

図 5-3 は、認知度と着手意向指数と SI 実績指数の 3 つの要素の相関関係を示したものだ。

図中の吹き出しの技術項目の右に示した数値は SI 実績指数を示し、それは円の大きさでも示している。また、SI 実績指数が 0.5 よりも大きいものは、円を黒色の縁取りで示す。

JISA では、この認知度を 4 つの象限で表している。

- 第一象限：認知度が高く、着手意向指数が高いもの
- 第二象限：認知度が低く、着手意向指数が高いもの
- 第三象限：認知度が低く、着手意向指数が低いもの
- 第四象限：認知度が高く、着手意向指数が低いもの

本書の技術・手法は、第一象限に 5 項目、第二象限に 19 項目、第四象限に 5 項目がプロットされている。第二象限の中で SI 実績指数が 0.3 以上のものは、「W (開発) モデル (0.308)」のみである。これは、今後さらに SI 実績が増えてくると、この技術を使ったことが世間に公開され、露出度が高くなり、認知度もアップすると想定される。それに伴い、将来的には第二象限から第一象限に進むと考える。第四象限の 5 つの項目は、SI 実績指数も 0.5 以上と高く、現場での実績も高いことを示している。この中の技術・手法は、図 5-2 で示すとおり「安定期」のものである。この技術・手法は、今後も重要なものであるので、「第三象限」、「衰退期」に向かうものではなく、新しい仕組みでこの技術・手法が改良され、発展していくものとする。

## 関連情報 (標準、各種 BOK、 SEC 成果物)

..... 154

6.1 国際標準など .....	154
6.2 知識体系 (Body of Knowledge) .....	155
6.3 SEC 成果物 .....	156

# 6章

## 関連情報（標準、各種 BOK、SEC 成果物）

### 6.1 国際標準など

事例の分析に関し、本書では下記の標準規格などを参考にしている。

- ① 品質モデル
  - ・ JIS X 25010 : 2013 (ISO/IEC 25010 : 2011)  
システム及びソフトウェア製品の品質要求及び評価 (SQuaRE<sup>17</sup>) - システム及びソフトウェア品質モデル
  - ・ つながる世界のソフトウェア品質ガイド
- ② 品質マネジメントシステム
  - ・ ISO 9000 - 基本及び用語
  - ・ ISO 9001 - 要求事項
- ③ ライフサイクルプロセス
  - ・ ISO/IEC 15288 システムライフサイクルプロセス
  - ・ ISO/IEC 12207 : 2008 (JIS X 0160 : 2012) ソフトウェアライフサイクルプロセス
  - ・ 共通フレーム 2013 (JIS X 0160 : 2012 をベースとしている)
- ④ プロセスアセスメント
  - ・ ISO/IEC 15504 シリーズ
  - ・ ISO/IEC 33000 シリーズ
  - ・ CMMI<sup>18</sup> (ISO/IEC 33000 の要件を満たすプロセスアセスメントモデルのひとつ)
  - ・ SPEAK-IPA (ISO/IEC 15504 に準拠したアセスメントモデル)
- ⑤ 機能安全
  - ・ IEC 61508 (JIS C0508) 電気・電子・プログラマブル電子安全関連系の機能安全
  - ・ ISO/IEC 26262 自動車 - 機能安全 -

<sup>17</sup> Systems and software Quality. Requirements and Evaluation

<sup>18</sup> Capability Maturity Model Integration

## 6.2 知識体系 (Body of Knowledge)

事例を読む参考として、ソフトウェアの品質に関連したソフトウェア情報システムの知識体系を紹介する。知識体系は、専門領域を構成する概念、用語、及び活動の完全セットを表す用語に関する知識（概念、用語、技術、ノウハウ、規格・標準など）の総和で、専門領域に関連する団体等によって定義されている。

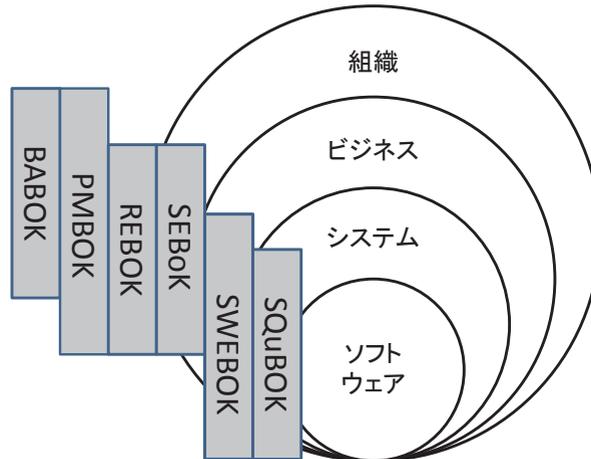


図 6-1 BOK が扱う範囲<sup>19</sup>

表 6-1 : ソフトウェア品質に関連するソフトウェア・情報システムの知識体系<sup>19</sup>

名 称	領域	ソフトウェア品質との関係
BABOK: A Guide to the Business Analysis Body of Knowledge	ビジネス分析	いくつかの知識領域で非機能要求、要求の品質、プロダクト品質、トレーサビリティを含む要求のマネジメントプロセス、技法としてレビューや根本原因分析などの扱い。
PMBOK : A Guide to the Project Management Body of Knowledge	プロジェクトマネ ジメント	独立した知識領域としてまとめてプロジェクト品質マネジメントのプロセスや品質管理ツールの扱い。主にプロダクト品質の観点。
REBOK : Requirements Engineering Body of Knowledge (要求工学知識体系)	要求工学	いくつかの知識領域で品質要求、要求の品質やプロダクト品質、要求の割り当て、管理プロセスの扱い。
SEBoK : Guide to the Systems Engineering Body of Knowledge	システムズエン 지니어リング	いくつかの知識領域で非機能要求モデルの品質、システム要求分析、システム設計における要求の組み入れへの言及。
SWEBOK: Guide to the Software Engineering Body of Knowledge	ソフトウェアエン 지니어リング	独立した知識領域として、プロダクト品質とプロセス品質、品質モデル、品質マネジメントプロセス、レビュー等の静的技法の扱い（動的技法は他知識領域）。
SQuBOK : Guide to the Software Quality Body of Knowledge (ソフトウェア品質知識体系ガイド)	ソフトウェア品質	全知識領域において、主にプロダクトの品質、品質マネジメントプロセス、静的・動的技法の扱い。

19 鷺崎 弘宜, “ソフトウェア品質の知識体系”, 情報処理 2014 年 1 月号 (Vol.55, No.1) を参照

## 6.3 SEC 成果物

IPA/SEC は多くの成果物を公開し、一部、書籍化している。図 6-2 のように、SEC 成果物の適用分野を大きく 3 つに分類している。一つはエンタプライズ系ソフトウェア開発関連で、もう一つは組込み系ソフトウェア開発関連である。さらに、2015 年度から IoT 関連も加わった。それらの成果物を、開発工程に対応させたイメージを図 6-2 に示す。

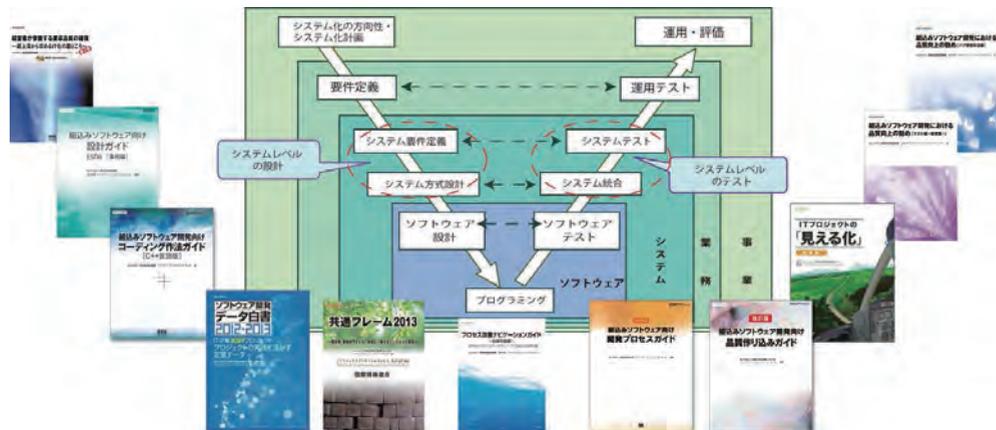
### 適用分野

エンタプライズ系ソフトウェア開発 関連書籍

組込み系ソフトウェア開発 関連書籍

IoT 関連書籍

### 開発工程との関係



【V字モデル】（「共通フレーム2013」より引用）

図 6-2 SEC 成果物

本書は、ソフトウェアの高信頼性を確保するために、開発プロセスに先進的な技術・手法を適用した事例を元に分析したものである。事例の中には一部、SEC 成果物を先進的な技術・手法と併用してソフトウェアの高信頼性を実現しているものもある。

組込み系の書籍は、図 6-3 のように 3 つの視点でガイドを提供している。3 つの視点とは、「プロセスの視点」「品質の視点」「マネジメントの視点」がある。また、経営者、開発者、マネージャの 3 つの立場の読者層も意識しているのが特徴である。これらは、ESxR シリーズで構成されている。

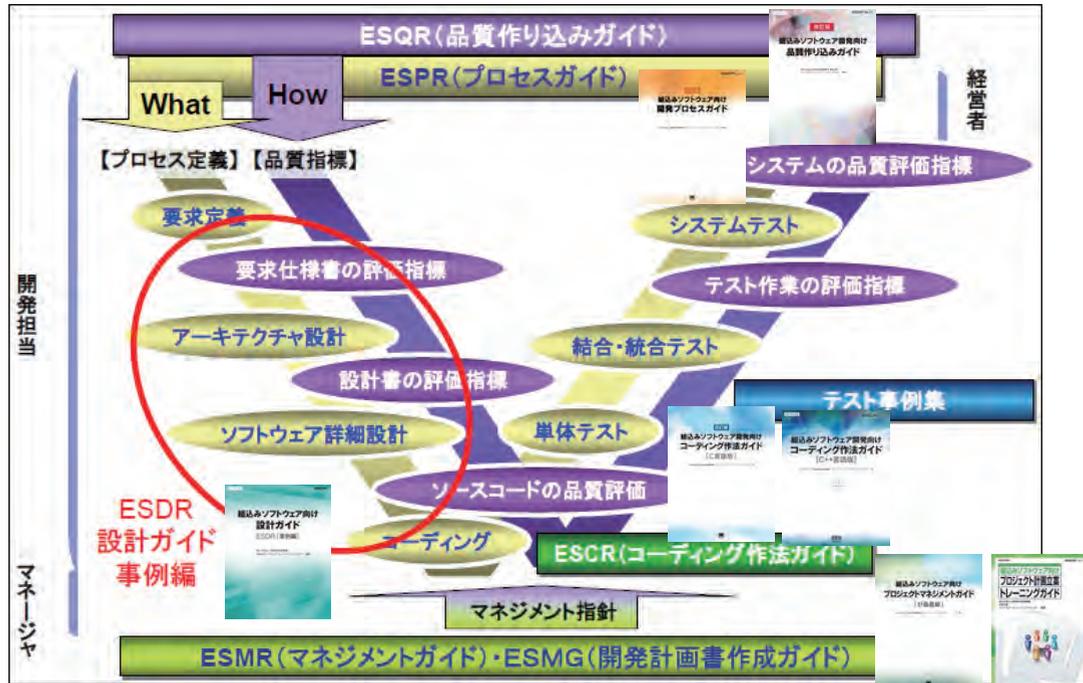


図 6-3 組込み系関連の書籍 (ESxR シリーズ)

ソフトウェアの高信頼性を確保するために、以下の SEC 成果物をそれぞれの主旨 (品質に関するもの、品質のメトリクスに関するものなど) に沿って活用できる。

### ① 品質に関するもの

- ・ 経営者が参画する要求品質の確保 ～超上流から攻める IT 化の勘どころ～ (第 2 版 2006 年 5 月 25 日)
- ・ 【改訂版】組込みソフトウェア開発向け 品質作り込みガイドブック (2012 年 9 月 10 日)
- ・ 組込みソフトウェア開発における品質向上の勧め
  - ・ コーディング編 (初版 2005 年 5 月 20 日)
  - ・ ユーザビリティ編 (初版 2006 年 5 月 31 日)
  - ・ 設計モデリング編 (初版 2006 年 6 月 20 日)
  - ・ テスト編 ～事例集～ (初版 2012 年 11 月 12 日)
  - ・ バグ管理手法編 (初版 2013 年 3 月 8 日)
- ・ つながる世界のソフトウェア品質ガイド (初版 2015 年 5 月 29 日)

### ② 品質のメトリクスに関するもの

- ・ つながる世界のソフトウェア品質ガイド (初版 2015 年 5 月 29 日)

### ③ 機能安全に関するもの

- ・ 組込みシステムの安全性向上の勧め (機能安全編) (初版 2006 年 11 月 10 日)
- ・ つながる世界のセーフティ&セキュリティ設計入門 ～IoT 時代のシステム開発『見える化』～ (初版 2015 年 10 月 7 日)



#### ④ その他

- ・ 共通フレーム 2013（初版 2013 年 3 月 4 日）
- ・ 高信頼化ソフトウェアのための開発手法ガイドブック（初版 2011 年 3 月 28 日）
- ・ 組込みソフトウェア向けプロジェクト計画立案 トレーニングガイド（初版 2011 年 11 月 9 日）
- ・ 組込みソフトウェア向けプロジェクトマネジメントガイド【計画書編】（初版 2006 年 11 月 22 日）

## 適用事例の概要

..... 160

7.1 設計編 .....	160
7.2 検証編 .....	172

# 7章 各適用事例の概要

## 7.1 設計編

### A-1：アシュアランス技術を用いた鉄道信号の革新

東日本旅客鉄道(株)

輸送管理においては情報系と制御系という異なるニーズを持ったシステムの共存、異種性が求められている。これに対応したシステムが、列車の運転を管理するための運行管理システムと呼ばれるシステムで、運転線区ごとに作られている。このシステムは、地理的にも広範囲で大規模であり、システム化も長期間を要するため、必然的に段階的な構築が必要となる。また、稼動したシステムは列車運転の性格から24時間連続運転システムとなり、システム拡張時にも稼動システムの安定稼動を保証した無停止拡

張が前提条件となり、かつ使用者の新しいニーズ（使用実績を踏まえての機能向上等）を吸収した変化・成長を前提とした構築が必要となっている。つまり、適応性が求められている。

これらの課題を克服するため、自律分散システム技術をベースとしたアシュアランス技術を活用して、首都圏「ATOS（Autonomous Decentralized Transport Operation Control System）」と呼ぶシステムを拡大・成長させつつ20線区、182駅に展開してきた。

### A-2：XDDP 導入による派生開発の品質改善とその効果

(株)日立産業制御ソリューションズ

創業以来培ってきた「情報と制御の融合技術」により、電力、交通、上下水、金融などの「社会インフラ」や、鉄鋼、自動車、電子デバイス製造などの「産業基盤」を支える監視制御、運行管理、生産管理、画像検査をはじめとする、さまざまな高信頼システムを提供してきた。その内訳を見ると、過去のソフトウェア資産を基にした改造や機能追加など、いわゆる派生開発が全体の約8割を占めている。

派生開発は新規開発に比べ、短納期・低コストの開発が要求され、他人が書いたソースコードを読まなければならない場面も多いため、「全体を理解」することが難しい。このように全体

を適切に理解できていない状態（部分理解の状態）では、思い込みや勘違いが起りやすくなる「部分理解の罠」に陥りやすい。この派生開発における品質の向上を目指して、ソフトウェアの派生開発のプロセスを強化するために、XDDPを導入してきた。

この適用の結果、品質評価において、設計段階で非適用に比べて154%の不良が発見され、顧客先での発見は84%も減少している。

工数評価においては、設計段階の工数は40%増えたが、総合テスト段階の工数は40%減少しており、総じて若干の減少につながっている。

### A-3：組込み系の利用品質における「HMI 品質メトリクス」開発と適用事例

(株) U'eyes Design

最近では、ユーザが直接ソフトウェアを操作する機会が多く、ソフトウェア開発の総合的な品質特性（ISO9126）の使用性を強化する意味

合いで、利用する人の視点から商品の価値をチェックする「利用品質」の向上が求められている。従来は、ユーザビリティの専門家による

定性的な分析・診断や、人間工学的なガイドラインを参考に属人的ノウハウに頼った品質向上対策が主流であった。しかし、本適用事例は、設計の目標値となり得る、客観的な定量指標の開発を目指し、開発現場への適用を試みたもの

である。こうした定量指標の一群は「利用品質メトリクス」として提唱され、システム開発における利用品質とシステム品質とを繋ぐ役割となることが期待されている。

#### A-4：要件定義段階における信頼性向上の取り組み事例紹介

ビッグローブ(株)

Web サービスやクラウドサービスの開発においては、低予算・短納期でのスモールスタートを前提としてプロジェクトを開始する場合が非常に多い。最短では数日～数週間のサイクルでアプリケーション開発を行うこともしばしばであり、常に数十本単位のプロジェクトが同時並行で動いている。このため、上流工程において性能や機能に関する要件を詳細に詰めるだけの十分な時間を確保できない場合が生じる。

また、顧客からの受託開発案件やカスタマイズの注文においては、顧客企業内でも関係部署

間での調整がなかなか取れず、下流工程での急な要求変更による開発や品質検証への負担が無視できない状態になっていた。

アジャイル開発手法で要件を詰めていこうとした場合でも、業務契約上、ベンダに完成義務がなくなり、最終的なコストが読みにくくなる不安が生じる。また、納期ぎりぎりまで要件が固まらず十分な品質検証が行えないままリリースせざるを得ない状況も発生していた。

上記の問題を軽減するために実践して効果的だった要件分析の効率化手法を紹介する。

#### A-5：要件定義の品質向上に向けた取り組み

富士通(株)

システム開発でのバグを分析してみると、要件定義に起因するものが多い。また、ユーザー企業も要件定義工程を改善しなければならないという思いがある。このような状況を捉えシステム開発における要件定義の品質向上のための取り組みをいくつか行っている。その中で、要件定義手法「Tri-shaping（トライ・シェイピング）」で実施している品質向上策を紹介する。Tri-shaping は、要件定義の「抜け」「漏れ」「曖

昧」だけでなく、その要求が企業や業務、利用者にどれだけ価値を提供できるかなどビジネス価値をも品質の対象としている。本適用事例では、要件定義での品質向上の取り組みを、3つの切り口で整理し対策を紹介していく。

この適用の結果、要件定義工程に対して、期間で2.3倍、工数で1.5倍かかったが、開発全体でみれば、期間は5%短縮、工数は4.5%短縮となっている。

#### A-6：ジェネレーターツールを利用した高信頼開発、高速開発の実践

(株)市進ホールディングス

2010年にグループ企業がホールディングス化され、システムにも統合・一元化を図り、高品質なサービスを迅速に提供することが求められるようになった。ブランドをより生かした事業展開を進めるために、基幹システムの現状課題（他システムとの連携、柔軟なシステム・サービスの顧客満足、運用維持管理の費用対効果）を解決する為に、先進的な技術/手法であるイテレーション（反復）開発ができ、かつプログラムとデータベースを自動生成するジェネレー

ションツール「GeneXus」を利用した。

今回は教育業界が初めてというベンダが新しい開発方法でシステムを構築するという事で懸念はあったが、結果的には、予定の期日通り稼働を開始し、移行後のサービス追加も予想以上に短期間かつ低コストで実現できている。

この適用の結果、要求定義に5カ月、繰り返しのプロトタイプ作成に7カ月を割当てたが、実装は4カ月であった（実装期間には検証も含む）。

## A-7：設計工程における TERASOLUNA DS の適用

(株)エヌ・ティ・ティ・データ

システム開発の上流工程である設計工程で、作るべきシステムの詳細な定義を漏れなく行うことは重要なことである。設計工程の品質向上のためには、下流工程に設計バグがすり抜けのない様に設計工程の中でレビューを行うのが一般的であるが、昨今の開発現場ではそのレビューが適切に機能していないのが現状である。下流工程へのバグのすり抜けの原因は、「設計書量が多く、開発期間が短いために、設計書間整合性を確保することが人間のレビュー作業では不可能」と結論付けた。そして、それを解決するために、設計支援システムを構築している。このシステムは、設計書を検索可能な情報としてDBで管理し、対象の設計書の各設計項目単位

で検索を行う機能(整合性チェック機能)とサービス開始後の保守維持のために利用される影響調査効率化機能で下流工程へのバグのすり抜けを防いでいる。

この適用事例では、当該システムを2012年から累積で50を超える開発プロジェクトに導入し、現在、その効果分析を実施している。

この適用の結果、結合試験時のバグ原因の47%は設計書が原因。「TERASOLUNA DS」の設計工程への導入により抽出されたエラーは、大きな手戻りにつながるものが25%、次工程で気づくと推定されるものが22%であり、開発ライフサイクルの上流の段階でのエラー抽出が可能である。

## A-8：Grails/Groovy の適用推進

エヌ・ティ・ティ・ソフトウェア(株)

従来からJavaを主力言語として開発を行ってきた企業において、主に企業システムを開発対象としたソフトウェア開発に関する種々の課題(開発期間短縮やコスト削減、設計品質の底上げなど)を解決するために、高機能なアプリケーションフレームワークであるGrailsを全社標準フレームワークとして採用している。

一方、インターネットサービス開発で良く使用されるフレームワークRuby On Railsは、そ

の分野では開発速度が死活的に重要であることを背景にして、迅速な開発を可能とする様々な機能と特長を備えている。両者のフレームワークの特徴を比較しながら、Grails/Groovyを社内適用している理由を示している。

実績として、販売ソフトウェアの主力製品を含め、2012～2013年度(2014年1月現在まで)で計42件のソフトウェア開発プロジェクト(継続プロジェクトを含む)に適用している。

## A-9：個人依存開発から組織的開発への移行事例

～要求モデル定義と開発プロセスの形式化による高生産性 / 高信頼性化～

三菱電機メカトロニクスソフトウェア(株)

製品の成熟化に伴う付加価値増加やグローバル対応による製品バリエーション増加に伴い製品に搭載する機能量は増加する一方である。個人依存の開発では機能量の増加に対応しきれず生産性と品質を維持することが困難となる。個人依存の開発から組織的な開発への移行が急務となっている。

ソフトウェア開発で発生する課題は、人の作業によって作り込まれる。課題を作り込まなくするために組込みソフトウェア開発の現場で発生する課題を定義して、課題を抑制する分析・設計・実装・テストのフレーム上にソフトウェアを構築する手順を開発プロセスに定義して組

織的な開発を進めることが大切である。この適用事例では、要求を正確に形式的に定義した分析モデルを設計や実装、テストの双方向に紐付けすることにより徹底した無駄取りによる高生産性、および不具合の混入を極力抑えた高品質化を可能にしたことについて説明する。

この適用の結果、技法適用前の2005年では、分析/設計の工数割合は10%以下で、作成とテストの工数割合が90%をしめていた。技法適用の結果、上流で不具合検出率が97%で、年々テストの工数割合が減少し、分析の工数割合が増加している。現在、分析の工数割合が約53%、テストが約20%である。

## A-10 : MBSE による双腕作業ロボット動作実行系のコンセプト設計

(独)産業技術総合研究所 (AIST)

現在の産業用ロボットの多くは大量生産に最適化したラインを構築するためのもので、人とロボットが分離して運用される。ところがこのようなロボットは技術や用途が飽和してきているのが現状で、今後は多品種少量生産へのロボットの導入が期待されている。

これに対して、近年、多品種少量生産へのロボット導入の有力なアプローチとして、双腕など、人に近い構造を持ったロボットが注目されている。

このような人に近い構造を持つ多品種少量生産のためのロボットは、短い製品サイクルに対応することが求められる一方、導入を容易にして人と空間を共有して作業することが求められる。そのためには要求仕様の頻繁な変更に対応しつつ、センサ等を用いた人との安全な共存、すなわち機能安全が求めるウォーターフォール

型の V 字開発プロセスを実現するという矛盾した要求に応えなければならない。

この適用事例では、MBSE の手法により、このような双腕作業ロボットシステムの設計プロセスを改善し、かつ頻繁な変更と機能安全という矛盾した要求に応えることを目指した。具体的には双腕作業ロボットのコンセプト設計の部分で、SysML モデルを用いて要求の分析およびアーキテクチャの設計を行った適用事例を紹介する。

この適用の結果、目標とした以下の項目はほぼ達成できた。

- SysML を用いた要求の関係の具体的な表現
- 分析した要求を満たすアーキテクチャ設計
- 工程の成果物や参照された文書間のトレーサビリティの実現
- 要求の追加時における効率的なシステムの拡張

## A-11 : 仕様記述言語 VDM++ を用いたシステムの仕様の記述

フェリカネットワークス(株)

おサイフケータイに搭載する「モバイル FeliCa IC チップファームウェア」の開発において、形式仕様記述手法を導入し、開発の成果を上げるのと同時に、手法適用の効果を確認している。この適用事例では、開発プロジェクトにおける形式仕様記述手法適用の成果を、仕様の開発効率、仕様を活用したテストや仕様自身のテストの結果、仕様に関わるコミュニケーションの分析、導入障壁をはじめとする問題点や課題、今後の展望等を交えて紹介する。

この適用の結果、下記の目標を達成できた。

### (a) 厳密な仕様の策定

形式仕様記述言語による 677 ページの外

部仕様書の策定時に、444 件の欠陥を発見した。

- (b) 開発プロセスの策定と導入
- (c) 仕様の多方面からの分析・精査
- (d) 仕様テスト
- (e) コミュニケーションの促進

結果として、上流工程で厳密に仕様を記述、検証するに従って、開発全体の中で、仕様策定工程が占める割合が大きくなる。そして、従来、下流の工程で問題が顕在化していた課題を、上流工程における厳密な仕様の記述と検証と共に解決した。

## A-12 : 車載 ECU 開発における上流工程での品質確保

東芝情報システム(株)

先進運転支援システム、HV 車、EV 車、燃料電池車など、車載 ECU の役割は増大し、複雑化している。ECU の開発現場では品質確保が絶対条件の使命であるが、設計時に予測しなかった問題が検証時に見つかることも多く、後戻りのコスト/期間の問題、市場への不具合流出のリスクなど、多くの課題を抱えている。

この課題を克服するために、上流品質の向上が叫ばれている。上流品質を向上するための手法の一つとして、シミュレーション技術を活用したモデルベース開発がある。モデルベース開発を適用することにより、上流でシミュレーションを行うことで、効率よく設計の妥当性を検証することが可能となる。また、上流で作成

したシミュレーションモデルを活用し、自動コード生成を行うことができるため、開発効率も上げることができる。さらに、検証工程においても、上流で作成したシミュレーションモデルを活用した検証ができるため、品質向上と生産性の向上の両立が可能となる開発手法として注目されている。

この適用事例では、3つのドメインを紹介する。

1つ目は、状態遷移を中心としたシステムで、比較的簡単なシステムであるが、今後他システムとの連携などで複雑化する可能性が大きいいため、モデルベース導入の効果が期待されるドメイン。

2つ目の例は、既に Simulink によるモデルベース開発が導入されているドメイン。

3つ目は、まったく新しい領域に対する上流品質の向上を目指した先進運転支援システム (ADAS)。

これまでの要求分析は、特定のスペシャリストが、頭の中で展開し、設計に落としてきた工程であった。そのため体系的プロセスがなく、検討した経緯を示すドキュメントなどはほとんど残っていないのが現状である。本プロセスに従って、SysML を用いて要求分析を行ってみると、検討した過程／結果が非常にわかりやすかった。設計変更や差分開発時に本プロセスに従って要求分析を行った SysML 準拠のドキュメントがあれば、新しい要求を追加する際の影響分析などに非常に有効なドキュメントになることが分かった。

### A-13：独自開発したモデル駆動開発プロダクトラインとエンジニアリングの実践

(株)デンソー

大規模化、複雑化、多様化する車載電子制御ユニット (ECU) の組込みソフトウェア開発において、従来型の既存ソースコードベースの派生開発では、品質を確保しつつ作業効率を維持・向上することは難しくなっていた。

そこで、モデルを基にしたソフトウェア・アーキテクチャを自動生成する「モデル駆動開発 (MDD: Model Driven Development)」を適用し、コア資産 (Component) 開発の効率向上、製品開発におけるコード自動生成 (コーディングなし) を実現した。また組織をコア資産開発と製品開発に分け、それぞれに適した技術者の育成にも努めた。

さらにソフトウェア・プロダクトライン・エンジニアリング (SPLE: Software Product

Line Engineering) の実効性を高めるため、製品の機能要件を整理するフィーチャー・モデルを定義した。そして、製品開発においてはコア資産を Component 単位ではなくフィーチャー単位で認識することにより、コア資産の再利用方法を適切に統制できるようにした。

2008年以降、延べ30を超える製品を本手法で開発している。本手法の適用の前後でソフトウェアの規模やバリエーションの数自体が変化しているため、単純に計測することは難しいが、生産性の面で2割程度の効率化が認められた。また、コア資産を100%再利用し、手動でのコーディングなしで多様なバリエーションを開発できるので、品質向上にもつながっている。

### 15-A-1：業務生産性向上や市場環境の変化に対応できる経営・業務を実現するために活用されるBPMの紹介

(一社)コラボネット事業推進協会 / (株)BPM 実践企画

国内企業——特に中堅・中小企業——は、国内企業のグローバル化に伴い、事業の国外進出や国外との取引の増加、為替レートの急激な変化対応など、これまでの業務形態では事業が成り立たなくなってきた。このような厳しい市場環境の変化が早くかつダイナミックであることで、中堅・中小企業には、経営と業務の変化の対応力や業務生産性向上が求められる。その

ためには、業務プロセスの可視化、見直し、実行を繰り返すことである。これを実現するのにビジネスプロセスマネジメント (BPM) が国内外で注目され効果を出している。この適用事例では、このBPMの注目される背景とBPMの仕組みと適用事例を取り上げ、日本の経済力の源である中堅・中小企業が国際競争力を持つようになるフレームワークを紹介する。

### 15-A-2 : BPM をベースにした会社統合での新業務プロセス設計の適用

三菱商事 RtM ジャパン(株)

新会社（2013年4月より新会社としてビジネスを行う）は子会社と親会社の金属グループから会社分割された金属資源トレーディング事業部門を統合し、設立された会社である。この

統合準備は2012年7月から8カ月の短期間にて遂行されたが、その際、新会社の業務プロセスをBPMの考え方を適用して設計／構築し、無事に稼働するに至った状況を紹介する。

### 15-A-3 : 人間系プロセスを含む業務をBPMソフトウェア活用により改善した事例

日本電気(株)

この適用事例では、旧来のデータ処理系ITでは弱点であった「人間系プロセスサポート」に着目し、BPMソフトウェア「Metasonic」で構築した業務システムの効果、対象業務プロセスの効率化、及び他の業務プロセスへの適用可能性について述べ、人間系を包含した全体システムの設計、運用の在り方の考察を紹介する。

企業内の業務は「IT化された領域」と「人

間系の処理」が有機的に統合されて初めてその力を発揮する。しかしながら、前者は技術の進展で処理形態を大きく変えてきた一方、後者では大きな進展はなく、人間系を含めた総合的な取り組みが企業力強化の大きな課題になっている。この適用事例では、同課題を解消するための取り組みを紹介する。

### 15-A-4 : 受注業務にビジネスアナリシス方法論を適用した業務システムの構築 ～中小企業の特注品業務プロセス改革の提案～

(株)プロセスデザインエンジニアリング

特注品の業務は従来の情報システムがカバーしてきた定型業務ではなく、非定型な意思決定業務である。そこで、ビジネスアナリシス方法論「GUTSY-4」を適用して、この課題解決を図るプロジェクトが開始された。このプロジェクトでは、約6カ月間で合計13回の打ち合わせによって、戦略をビジネスプロセスと業務ルールに構造化して業務システムを設計し「見える化」した。

結果として、特注品の売上は直後で約2倍、直近では約4倍に伸長した。これは、ベテラン営業の顧客要求の引き出し方法を形式知化して、情報システム上で社内共有した効果が大きい。

また「見える化」は、ベテランから若手へのノウハウ伝承、機能として全社業務を理解したマネージャの誕生、東日本大震災の復旧、海外とのM & A交渉の成果も生み出した。

### 15-A-5 : ビジネスへの貢献が求められる時代のソフトウェア開発の考え方 ～超高速開発ツールがもたらす方法論のイノベーション～

(一社)ICT経営パートナーズ協会 / MBC (Method Based Consulting)

情報システムが単に業務の効率化の道具ではなく、新たな市場や顧客、調達先を獲得することに貢献する道具だとの考えが浸透している。企業だけでなく、あらゆる組織において、情報セキュリティの問題も含めて、「情報をどう扱うのか」、「情報とどう向き合うのか」ということが今の時代以上に、一般人を含めた多くの人から脚光を浴びている時代はない。一方、現在

の多くの情報システムは、「ビジネス環境の変化に迅速に対応できる」状態ではないことが課題となっている。

数年前から、ソフトウェアの設計情報や業務視点で記載された機能仕様をリポジトリ<sup>20</sup>というデータベースに管理できる開発ツールが市場に提供されている。この適用事例では、そのような「リポジトリを持つ開発ツール」が実現す

る価値を示しながら、ソフトウェアの開発と保守のやりかたにどのような変化をもたらすのか

を明らかにすることを狙いとしている。

### 15-A-6：「フィーチャー」の概念を取り入れたモデルベース開発

三菱スペース・ソフトウェア㈱

ソフトウェア開発現場において、若年層や詳細設計書段階から開発に参加したエンジニアは、作ろうとしているプログラムの上位設計書として何を見ればよいのか、どの部分を仕様として捉えればよいのかが分からず苦労していることが多い。これは、システムへの理解不足や設計手法の習得不足が原因ではあるものの、そもそも、当該プログラムに関係する機能方式設計や詳細設計と対応づけるものがないことや、あったとしても、必要な機能方式設計の項目抜けなどの検出が難しいことが原因であることも

多い。

この適用事例では、上記の設計段階の課題を解決できるよう、各設計フェーズの設計粒度を具体化し、それぞれを関係付ける方法を明確にしている。具体的には、アジャイル設計手法の考え方の一つである「フィーチャー（小さな機能のかたまり）」の概念を取り入れ、ソフトウェア要求分析～ソフトウェア詳細設計までシームレスに繋ぐモデルベース開発の仕組みを紹介する。

### 15-A-7：D-Case 導入によるシミュレーション S/W の期待結果明確化と合意形成

三菱電機㈱

ソフトウェア開発において、事前に期待結果を明確化することが困難な場合、システムとしての妥当性確認が適切に実施されない、確認方法についてステークホルダ間で合意が得られない、というリスクがある。このリスク対策として、対象製品に求める特性を定義し、見える化することにより、妥当性確認項目を洗い出すという活動が必要である。

この適用事例では、システムのディペンダビリティについて説明責任を果たし、合意形成するためのツールである「D-Case」をシミュレーションソフトウェア開発に導入することにより、期待結果の明確化とステークホルダ間の合意に成功し、手戻りを防止することができたことについて紹介する。

### 15-A-8：大規模システム開発プロジェクトにおけるユーザーエクスペリエンス品質設計プロセスの適用とその効果

NEC ソリューションイノベータ㈱

この適用事例は、大規模なシステム開発プロジェクトで UX (User eXperience) 品質設計プロセスを実践的に成功させたものである。また、UX 適用効果や、その後の運用について紹介する。

顧客要求の高度化に伴い、システムは複雑になり、使いやすくなりやすい操作が求められている。ここでも、スマートデバイスなどコンシューマ製品の業務浸透による顧客経験値向上

が確認され、SaaS などクラウド利用に伴う業務サービスが多様化し、結果として機能そのものよりもユーザが使いたいと思う心地よさが期待されている。

このようなニーズの変化に対応するため、組織レベルで SI/ ソフトウェア開発プロジェクトの改善活動に努めてきた。特に、システムやサービスを人の視点に立ってユーザの体験価値を高める「ユーザーエクスペリエンス (UX)」の考

<sup>20</sup> 業務の機能やソフトウェアの設計の仕様を保持しているデータベース。詳細は、「超高速開発が企業システムに革命を起こす」[3]を参照のこと。

え方に注目し、それを大規模なプロジェクトに適用させるための新しい手法および枠組みの研

究開発を進めている。

### 15-A-9：ソニーの電子お薬手帳システム「harmo」に適用したセキュリティ設計分析手法

ソニーデジタルネットワークアプリケーションズ(株)

インターネットとコンピュータシステムが重要な社会基盤となった今、セキュアなコンピュータシステムの開発方法の重要性が高まってきた。特に医療系のコンピュータシステムでは患者の医療情報を扱うため、高度なセキュリティが求められる。この適用事例として紹介する「電子お薬手帳システム」には、薬剤師が患者の薬歴情報と処方箋から危険な薬の組み合わせを確認する医療業務支援機能がある。ここで患者の薬歴情報が第三者に改ざんされていると危険な薬の組み合わせを発見できない危険性がある。また患者の薬歴情報が漏えいすると、患者が重病を抱えていることを他人に知られてしまうなどの危険性もある。このような問題を起こさないために、電子お薬手帳システムには高度なセキュリティが求められる。

セキュアなコンピュータシステムを作るためには、まず (a) セキュアな仕様・設計を作り、それに基づき (b) セキュアなソースコードを

作り、(c) セキュリティ観点でシステムを検査する必要がある。(b) および (c) には静的解析ツールや動的検査ツールなどの自動化された解決策やセキュアコーディング教育があり、広く適用され普及している。

一方、(a) については自動化されたものではなく、人手による高度な知識と経験を要する仕様・設計のセキュリティ分析（以下、セキュリティ設計分析）しかなく、セキュアな仕様・設計作りはまったくと言ってよいほど普及していない。しかしながらその重要性は益々高まっている。

セキュリティ設計分析では、暗号技術やOSごとの癖などの個別技術のセキュリティ知識と、それら知識を基礎としシステム全体のセキュリティ状況を可視化する分析技術の2つのスキルが必要となる。この適用事例では後者の可視化技術について従来課題を解決した分析手法を紹介する。

### 15-A-10：XDDPにおけるデグレード防止効果を高めるための手法 ～『気づきナビ』の考案～

アズビル(株) インテック キヤノンITソリューションズ(株)

既存のソフトウェアに変更・追加等を行い、新たにソフトウェアを開発する派生開発では、既存ソフトウェアの仕様書や設計書が更新されない、設計書とソースコードの乖離が次第に大きくなるといった課題がある。そのため、非熟練技術者の場合、派生開発での仕様変更の影響箇所の見極めを誤り、変更箇所の故障や既存箇所の故障（デグレード）が繰り返し発生している。

派生開発におけるこれらの課題を解決するため、XDDP (eXtreme Derivative Development Process) 手法の導入を検討したが、XDDP手法においても、非熟練技術者の場合、デグレードに対してはXDDPの導入効果が少ないこと

が現状分析(シミュレーション)で明らかになった。

この適用事例では、この課題を解決するため、XDDPの導入に伴って、熟練技術者に依存しなくても変更箇所の故障やデグレードを低減するための手法について紹介する。

この適用事例で導入したXDDPは派生開発に特化したプロセスモデルであり、変更を表現する視点から3点セットと呼ばれる3種類の成果物を定義している。具体的には、「何を」「どのように」変更するかを記述する変更要求仕様書、「どこを」変更するかを記述するトレーサビリティ・マトリクス、「どうやって」変更するかを記述する変更設計書の3種類である。

## 15-A-11：モデルベース開発への移行に向けたC言語ソースコードに対する状態遷移抽出技術の適用

(株)東芝

従来型の開発からモデルベース開発に移行する際には、モデルベース開発に必要なモデルを構築する必要がある。これには、二つの方法が考えられる。一つは、既存開発の設計ドキュメントに記載された設計情報に基づいてモデルを構築する方法。もう一つは、ソースコードに基づいて設計情報を抽出しモデルを構築する方法である。

一方、不適合の解消や派生開発の対応などによるソースコードの修正が起因となって、設計ドキュメントに記載された設計情報が、ソースコードから抽出した設計情報と整合しない場合がありえる。そのため、設計ドキュメントではなく、ソフトウェアの実際の挙動を表したソー

スコードに基づいてモデルを構築する方が適切である。ただし、その構築の際には、ソースコードにおいて仕様に基づく記述と実装依存の記述との区別が容易ではないため、工数がかかること。また、人手によりモデルを構築する場合にはモデルの抽象度のばらつきが発生するといった課題がある。

そこで、この課題を解決するために組み込みソフトウェアのC言語ソースコードを対象とし、組込みソフトウェアに不可欠な状態遷移モデルを抽出する技術を提案する。さらに、この技術を実際のソースコードに適用した適用事例を紹介し、本技術の効果と課題を述べる。

## 15-A-12：組込システムのモデルベース開発適用におけるDIコンテナの活用

ヤマハ(株)

組み込みシステム製品における近年の大規模化・複雑化は、多くのメーカーの課題であるが、主力製品である電子楽器もその例に漏れず開発上の課題となっている。その解決策の一つとしてオブジェクト指向技術を基盤とするモデルベース開発の導入適用事例が増加しているが、理想的な設計モデルを実装するには、モデルとして表現されたオブジェクト間の関連をいかにモデルに影響を与えずに実現するかが課題となっている。また組込みシステム製品は、省

リソースのハードウェアで実現する上で、パフォーマンスや資源管理など様々な課題が存在する。

そこでDIコンテナ技術を用いたソフトウェアフレームワークを開発し、電子楽器製品への適用した。現在までに20機種以上の主力製品に適用し、ソフトウェア・プロダクトライン・エンジニアリングを実現し、大幅な開発効率・品質向上を成し遂げている。

## 15-A-13：大規模システムへのモデルベース開発手法の適用

(株)IHIエアロスペース

近年、モデルベース開発の適用は組込みシステムに対して広がりつつある。モデルベース開発は、上流工程でのシステムレベルの不具合検出や、要求トレーサビリティ確保の容易さなどの点において、開発をより効率化できることが期待される。

この適用事例では、大規模システムであるロケット電子制御システムのソフトウェアに対して開発をより効率的、効果的に進めるためのモデルベース開発の適用可能性について検討を行い試行したものを紹介する。

制御システムの設計にモデルベース開発を適用する場合、MILS (Model In the Loop Simulation) を実施時に、制御対象物のモデル化の正確性が不十分だとMILSによる設計効率化はそれほど期待できないという課題があった。しかし、ある工夫の採用でMILSとHILS (Hardware In the Loop Simulation) をほぼ同時に実行させることで、MILSによるモデル化の正確性が不十分な場合にも開発効率をかなり上げることができ、従来と比べて後戻りの少ない設計が可能であるモデルベース開発の有用

性が確認された。

また、組込みシステム開発のさらなる効率化のために導入した、コード自動生成の評価を行った。自動生成されたソフトウェアと、従来どおり手作業で実装されたソフトウェアとを比較した結果、両者の制御性能には有意差がないこと、およびハードウェア性能に影響を与える

使用メモリサイズにも有意差がないことが確認された。この事実により、手作業で実装されたコードと同レベルの品質を確保しながら開発の短縮化などに貢献することが確認できた。

この適用事例では、これらの評価結果を示しながら、ロケット電子制御システムへのモデルベース開発の適用に対する取組みを紹介する。

### 15-A-14：自動車のパワーバックドアシステム開発のためのモデルベースシステムズエンジニアリングの適用

慶應義塾大学大学院 日産自動車(株)

自動車の電気／電子アーキテクチャ（以下、E/Eアーキテクチャ）では、ユーザからの要求の多様化に対応するため、追加開発を余儀なくされている。この追加開発にモデルベースシステムズエンジニアリング（MBSE）を適用することによる効果を明確にするため、この適用事例では、アクセス・盗難防止システムの一つであるパワーバックドアシステムの追加開発を取り上げる。MBSEに基づき SysML (Systems Modeling Language) を利用してパワーバックドアシステムのシステムモデルを記述しアーキテクチャを構築するまでのプロセスを示すとともに、部分として考えるのではなく、システムとして考えることによる技術者の気づきや明らかになった点を示す。特に、SysMLを用いたシステムモデルの記述により、従来の「部品ありき」の設計スタイルでは見いだすことのでき

なった組織の枠にとらわれない、対象とするシステム全体としての最適化の検討が行えたことは大きなメリットとして認識された。また、従来の開発では、成果物そのものを残すことや、その質が個人のスキルに依存していた。しかし、MBSEのプロセスで SysML を用いることにより、文書として残すべき成果物と、関連するシステムモデルを明確にできることの優位性が評価された。ここでは、既存のパワーバックドアシステムの解析から、「操作性のよさ」と「おもてなし」に関するユーザーニーズを引き出し、ユーザの自然な動作によって操作可能な新しいパワーバックドアシステムを設計する。新しい機能のコンポーネントへの割り当てを SysML ダイアグラム上で検討する中で、新・旧パワーバックドアシステム間でのいくつかのコンポーネントの統合を図った設計を試みている。

### 15-A-15：製品開発における SysML 適用の取組み ～要求の可視化～

(株)リコー

最近のネットワークに接続されて使用するプロジェクトなどのシステム製品開発においては、製品単体のみではなく製品間が連携することで新たなユーザ価値を素早く提供することが重要となっている。

また、このような製品の適用領域は、オフィス領域に閉じていた時代からコンシューマ領域にも拡大してきている。コンシューマ領域での特徴は、ユーザの利用シーンがパソコンだけを前提に利用していたオフィス領域とは異なり、スマートフォンやタブレットといったモバイル端末と組み合わせて使用されるなど、多様化してきていることである。また、オフィス領域においてもコンシューマ領域と同じ操作感、同じ機能(普段利用しているモバイル端末での操作)

を求められることが増えてきており、領域間の境目がなくなっている。

その結果、製品開発においては、すべてを自社で開発するのではなく、自社開発オフィス領域製品と既存のコンシューマ領域製品を連携させることで素早く価値を提供することが必要となり、システム製品開発プロセスそのものに対する要求も多様化・複雑化してきている。

このような状況のなかで、お客様、販売担当、商品企画担当など各ステークホルダの要求を正確に捉えるとともに、開発対象となるシステムの複雑化と大規模化に伴い発生するソフトウェアやハードウェア、メカ等各技術領域間にまたがる製品開発における課題（性能や操作性等）を適切に解決し、さらに多様化・複雑化した仕

様を漏れなく仕様化し実現するための仕組みが必要となる。

これらを解決するために、システム全体を記述できるモデリング言語 SysML の要求図の中

心に適用することで、目的や要求を獲得／明確化し、各技術領域にまたがる課題に対して目的や要求に合った解決手段の提案を行い、製品仕様の決定を行った適用事例を紹介する。

## 15-A-16 : D-Case を用いたゴール共有による開発プロセスの適用 ～ ET ロボコンでの試行と成果～

富士ゼロックス(株)

この適用事例では D-Case を用いて開発のゴールを関連者で共有しながらソフトウェアを開発したことを紹介する。ソフトウェアではでき上がったシステムが本来の要求と異なってしまいう事で、作り直しなどの手戻りが発生する事がある。こうした現象を防ぐために、どのようにして要求の意図を関係者間で理解し、共有して開発を進めて行くかが課題である。今回、「ET

ロボコン」の開発で D-Case を活用し、ゴールを共有しながらソフトウェア開発を進める事で手戻りを削減し、円滑にプロジェクトを進められる事を確認した。また、D-Case を使う事で、システムの信頼性や設計根拠を示す事ができた。その結果、ET ロボコンでも競技部門、設計審査部門ともに上位の成績を残すことができた。

## 15-A-17 : システム記述言語 (AADL) による複合システム設計～航空機前方車輪の回転数から速度を計測・記録・表示するシステムへの適用～

ISAE (Institut Supérieur de l'Aéronautique et de l'Espace)

航空機などの複雑なシステムは、複数のコンポーネントで構成される。各コンポーネントは、複数の開発組織で異なる技術を使い、個別に開発や検証される場合が多い。したがって、個別に開発や検証されたコンポーネントを複合システムとして統合する際には、コンポーネントレベルの手直しや再検証が必要になる。

今回のシステムは小規模ではあるが、車輪回転数から速度演算を行うコンポーネント、速度を記録するコンポーネント、速度と記録を表示するコンポーネントなどの複数のコンポーネントで構成される。このような複数コンポーネントで構成される複合システムを、手直しや仕様

の変更をすることなく開発する手法について述べる。

この適用事例では、システムの開発プロセスとしてトップダウンのアプローチを適用した。システムレベル (アーキテクチャレベル) の仕様の定義には AADL (Architecture Analysis & Design Language) を使用した。AADL を採用することで、複数コンポーネントからなる複合システムをシステムレベルで記述可能になる。これにより、システムレベルの不具合 (コンポーネント間のインタフェースの不一致、機能重複や機能欠落など) の確認などができる。

## 15-A-18 : ロケットエンジンにおけるモデルベース 信頼性評価技術の構築と試行 国立研究開発法人宇宙航空研究開発機構

宇宙輸送システムはひとたび点火されて地上を離れたら、そのミッション終了まで機能を停止することが許されない。当然、修理もできない。文字通り一発勝負である。ロケットエンジンはその推進機能を担う中核のシステムであり、非常に高い信頼性が要求される。

これまでのロケットエンジンは開発時に膨大な回数の燃焼試験を行い、その過程で顕在化し

た不具合は徹底的に対処するという実証主義により信頼性を確保してきた。しかし、このような開発プロセスには大きく2つの課題がある。

一つは今後、有人宇宙輸送等にも使用できるような高い信頼性を有するロケットエンジンを実現するためには、その信頼度を評価するために多数の大規模なシステム実証試験 (エンジン燃焼試験) が必要となり費用や時間が莫大にな

ること。もう一つは試験で不具合を洗い出し設計に立ち戻って対処をすることにより、開発コストが増加したり、期間が非常に長くなったりすることである。

このような課題に対しては、上流の設計段階で十分な設計検証を行うことが重要であり、有効な手法の一つとしてシミュレーション技術を

活用したモデルベース開発が自動車をはじめとする民生分野において広く行われている。ロケットエンジンの信頼性設計や検証に関する課題についても同様にモデルベース開発は有効であると考え、モデルベースの信頼性評価技術の構築と試行を実施した。

### 15-A-19：デジタル制御電源製品開発に対するモデルベース開発の適用

(株)富士通研究所

データセンタ省エネルギー化の取り組みの一つとして、電源内の制御をマイコンに組み込まれたソフトウェアによって実現する「デジタル制御電源」の研究開発に取り組んでいる。従来の製品開発での、①回路設計や制御系設計とシミュレーション、②回路製作とプログラミング(並行実施)、③実機テストと3つに分かれていた開発プロセスでは、以下のような課題があった。

- 設計時のシミュレーションと実機動作の結果が異なる場合、その原因が回路(ハードウェア)にあるのか、制御ロジックにあるのか、あるいはコーディング時の人為ミスにあるのか、の切り分けが困難となり、このためデバッグ工数の増加が懸念される。
- 実機テストでは異常ルートの信頼性の評価が難しく、潜在バグの危険性が懸念される。
- 複雑な制御ロジックを使用した場合の原理的な限界性能がどこにあるかを追求する事ができない。

また、製品の高度化に伴って、製品開発でのデバッグ工数の増加や、信頼性の確保が困難、性能限界を追究できない、などの問題が深刻化することも予想された。

この適用事例では、これらの課題に対応するため、モデルベース開発(MBD:Model-Based Development)の手法をデジタル制御電源の研究開発に適用したことを紹介する。ここで用いている主なMBDの手法は、「MILS:Model In the Loop Simulation」「RCP:Rapid Control Prototyping」「PCG:Production Code Generation(量産コード自動生成)」である。なお、デジタル制御電源開発にRCP、PCGを適用する場合、ツールベンダが提供する標準的な環境では十分ではないことが明らかになったことから、これを解決するために必要な要素を独自開発した。

### 15-A-20：組込みソフトウェアのアーキテクチャ設計方法の可視化

ビー斯拉ッシュ(株)

組込みソフトウェアの開発は、デジタルカメラやデジタル家電などに代表されるデジタル機器の普及、進化に伴い、急速に大規模化、複雑化、短納期化、多機種化しているため、品質が安定せず、生産性も悪化するという課題があった。

このような状況を打開するために、既存コードのリファクタリングを進めるボトムアップの取組みに加えて、開発戦略や設計思想をシステム仕様に落とすことで、上流工程の完成度を上げることを重視したトップダウンの取組みが必要であると考えた。

この適用事例では、上流工程の完成度を上げるこ

とを重視したトップダウンの取組みとして、「アーキテクチャ設計を可視化する取組み」について紹介する。このアーキテクチャ設計では、開発戦略の策定、目論見(開発戦略から製品の特徴や売りを端的に表現したもの)の作成、設計方針の策定、構造設計(静的/動的)、可視化という手順で開発を進めて行く。なお、リファクタリングはソースコードの質を高めることでソフトウェアの資産価値を高める施策ではあるが、この適用事例では取り上げていない。

## 7.2 検証編

### B-1：宇宙システムにおける上流工程仕様の妥当性確認技術

(独)宇宙航空研究開発機構

宇宙システムに代表されるセーフティクリティカルシステムでは、要求された機能が諸条件の元で動作するという信頼性の向上のみではなく、安全性の視点でもその健全性の検証が求められる。特に、ソフトウェア開発において、仕様どおり動作することの保証は、従来の品質工学、信頼性工学に基づく手法により強化が可能であるが、要求仕様等の上流工程における仕様定義時の正確性の欠如、不完全性の残留、想定不足がシステムの安全性に脅威をもたらすことが多い。

宇宙分野でも、NASA 等の過去の事故適用事例からの教訓として、上流工程の仕様に関する問題に起因した事象が注目されている。JAXA では、この解決のために、ソフトウェア IV&V（独立検証及び妥当性確認）活動として、「モデル検査」や過去経

験知に基づく「チェックリストベースレビュー」を導入している。

この適用の結果、当初目的として設定した上流工程における仕様定義段階の3つの課題の改善状況は以下の通りである。

- **正確性の欠除**  
チェックリストベースレビューにて多くの問題を早期に抽出できた。仕様書間の一貫性をモデル検査技術により確認した。
- **不完全性の残留**  
モデル検査技術により抽出可能となった。
- **使用方法・環境の想定不足**  
モデル検査である程度抽出できるが、今後の改良が必要。

### B-2：鉄道の機能安全 (RAMS) 認証支援のためのセーフティケース

(独)産業技術総合研究所 西日本旅客鉄道(株)

多くのシステムは、規模の拡大や多機能実現のために複雑化が進む中、航空機、鉄道、自動車など、様々な高信頼システムにおいては、高い安全性も同時に求められている。

特に近年、安全性の中でも機能安全という考え方が注目されており、機能安全の最上位の規格である IEC 61508 をはじめ、各分野において機能安全規格が発行され、その対応が求められている。鉄道も高い安全性を求められる分野であり、機能安全規格が存在する。鉄道分野においては機能安全規格の適合を示す際には、セーフティケースという安全性を示す文書を作成し、それを基にアセサーがアセスメントを行う。この安全性を示すセーフティケースという文書は鉄道分野の機能安全規格の一つである EN

50129 / IEC 62425 に記載内容が記述されている。それによると計画から始まり、検証、妥当性確認と非常に詳細な記述が必要である。そのため、セーフティケース作成には非常に大きなコストが必要となる。また、詳細な記述が必要となるため、その内容に漏れ抜けが発生しやすくなるという問題点もある。一方でセーフティケースを基にアセスメントを行うのだが、その内容が規格の要求事項にどのように従っているのか、記述の根拠理解が困難という問題も存在している。

これらの課題を解決するために GSN を用いたセーフティケース作成支援と、アセスメント支援の方法を開発した。

### B-3：非機能要求グレードの大学ポータルサービスへの適用

名古屋大学

名古屋大学では、大学内の「情報環境マスタープラン 2010」に従って、情報環境整備のための評価指標の策定を進めている。本適用事例では、IPA が提案している非機能要求グレード活

用シートを用いて、大学ポータルサービスの信頼性指標を定義するとともに、D-Case を用いて信頼性評価指標を確認する方法について検討を実施した結果について報告する。

この適用の結果、信頼性評価指標の定義表の列を D-Case の階層に対応付けることで、名古屋

大学のポータルサイトの信頼性を、客観的かつ容易に確認できるようになった。

#### B-4：冗長構成システム（クラウド等）の耐故障性に対する検証技術

（株）富士通コンピュータテクノロジーズ

クラウドを中心とした冗長構成システムで実現される高信頼性システムは、社会インフラの ICT システムとして、なくてはならない存在となっており、トラブルが発生した際の影響度は計り知れない状況となっている。

そのような状況では、トラブル発生に対して即時対応できる運用サービスとする必要がある。しかし、評価や検証手法が十分に確立できていないため、運用に入ってから、重大な障害となるケースがたびたび発生する。

この適用事例では、システムが高度化／複雑化していく中、高信頼性システムについて要件定義から運用や維持までのライフサイクルで、品質を確保して高信頼性を維持していく方法の

一つを紹介する。それは、ハードウェアの故障に対するシステムの耐故障性の確認を、検証項目及び確認内容の網羅性を確保しつつ、ハードウェアの疑似故障ツールを用意し、効率よく検証する効果的な検証手法の適用である。

この適用の結果、「非機能要求グレード」を活用することで、検証項目と確認内容の網羅性を確保しつつ、疑似故障ツールを用意したことにより、効率的に耐故障性の検証が可能になった。これにより、従来は実際の運用に入ってからでなければ見つけることができなかったハードウェアの故障に起因するトラブル要因を未然に検出し、対策を講じることにより、運用品質の向上に貢献できた。

#### B-5：単体ランダムテスト実行／可視化ツール“Jvis”の適用事例

宮崎大学

テスト工程は、ソフトウェア開発全体の工数の中で 50% を占めると言われ、非常に多くのコストを要する。また、ソフトウェアの大規模化、複雑化、短納期化に伴い、ソフトウェアを十分に検証することが難しくなっている。このような背景から、テスト工程のコスト削減やテスト効率の向上を目的とした幅広い研究が行われている。

一方、Web サービスの開発やスマートフォンのアプリケーション開発など、環境の変化がめまぐるしい領域では、短いスパンで実装して

評価することを、何度も繰り返し行うアジャイル開発を採用していることが多い。評価を何度も繰り返す必要があるため、アジャイル開発においてもテスト効率の向上、特に単体テストの効率向上が求められている。

この適用事例では、この問題に対する解として、大学で研究開発された単体テストを自動化し、かつ、テスト実施状況のリアルタイムな可視化を同時に行う単体テスト可視化ツール Jvis (Tool for Java programs to visualize unit testing) を紹介する。

#### B-6：要求仕様明確化のための仕様記述技術（USDM）活用事例

（株）ベリサーブ

自動車メーカーの A 社は、国内サプライヤーとシステム開発を行う場合に、仕様の詳細を打合せ等で決めていく、通称「擦り合わせ開発」が進められていた。しかし、近年の情勢の変化から、海外企業を含んだ複数のサプライヤーと取引するケースが増え、国内サプライヤーと同様の詳細度合の要求仕様のままでは、今までのような開発ができない状況に陥っている。これらに

対応するには、従来の属人的な業務方式からプロセス型の業務方式への移行が必須となっている。当然ながら、業務方式が変化しても品質は維持されなければならない、これらの課題を解決する仮説として、要求（要求仕様）を明確化することを定義した。要求（要求仕様）の明確化とは、高いスキルを保有している技術者間で暗黙的に取り交わされる要件を仕様として文書化

することである。要求が明確化されることで、属人性に依存してきた技術を明らかにでき、既存以外のサプライヤとの取引を可能にするとともに、サプライヤとの業務の分担を明確にすることができる考えた。

この適用事例では、これら要求（要求仕様）の明確化を実現する手段として USDM (Universal Specification Describing Manner) とテスト設計技術を利用する試みを紹介する。

この適用の結果、以下の項目をほぼ達成できた。

- USDM を利用した要求（要求仕様）を記述するためのガイドラインを作成する

ことで、安定した品質で作成が可能になった。

- 要求に対応する受入規準が明確になり、受入試験の項目作成で品質向上が図れた。
- テスト技術を利用して体系的に整理することで、トレーサビリティなどの影響分析に利用可能となった。
- W モデルでの担当する役割分担を明確にすることができた。
- 第三者による実施により、既存の技術者に少ない負担で行えた。

## B-7：形式手法を用いたセキュリティ検証

アーク・システム・ソリューションズ(株)

近年のシステム・ソフトウェア開発における、セキュリティ要件の増大とその保証を確立しなければならない時代において、従来手法であるレビューだけでは十分な保証が難しくなっている。セキュリティ分野においても高品質、高信頼化が求められるようになり、セキュリティが担保されていることを保証するための客観的な方法として、形式手法に注目が集まってきた現状がある。情報セキュリティを評価するための国際規格であるコモンクライテリア (ISO/IEC 15408) 等においても、高レベルのセキュリティ保証として形式手法の適用が求め

られている。

この適用事例では、「平成 24 年度戦略的基盤技術高度化支援事業 一形式手法を活用した組込みセキュリティ技術の確立と、安全で安心な CPS 社会を支える無線通信ミドルウェアの開発一」事業において、組込み機器におけるセキュリティの高品質、高信頼化へ貢献するため、形式手法を用いたセキュリティ機能の有効性の検証を紹介する。選択した形式手法は Event-B (Rodin プラットフォーム) であり、集合論を用いた定理証明やモデル検査が可能となっている。

## B-8：形式仕様記述手法を用いた高信頼性を達成するテスト手法とその実践

フェリカネットワークス(株)

形式手法の利用に関して、3 段階の適用レベルが示されることがある。そのうち、この適用事例は、形式仕様記述を行うレベル 0 の段階 (レベル 0 の段階とは、証明までは行わないが、数学的な記法を用いて厳密な仕様を記述する段階) である。上流工程において記述した形式仕様記述を検証し、その後、仕様に基づいたテスト工程において形式仕様記述を活用して効率的にテストを実施したものである。

この適用事例では、仕様の読みやすさやテスト工程での活用方法を考慮した仕様記述フレームワークの設計方法と、テストにおける形式仕様記述の活用方法 (モバイル FeliCa IC チップ

開発において定めた品質目標を効率的に実施するための形式仕様記述の活用方法) についても述べる。さらに、これらの取り組みの評価として、形式仕様記述を用いた場合と用いなかった場合の開発効率や形式仕様記述を用いることで品質目標を達成したテストの効果についても考察する。

この適用の結果、形式手法記述を用いることによって、定められた品質目標を達成するために開発を行わなければならなかった 2 つの生成器を新たに開発する必要がなくなり、具体的に開発効率を向上させることができた。

## B-9 : Orthogonal Defect Classification 分析による欠陥除去と品質の成熟度可視化

オリンパスソフトウェアテクノロジー(株)

ソフトウェア開発において、その成熟度を評価し、品質を向上させるためには、一般的に2つの手法が用いられている。ソフトウェア成熟度の評価には、信頼度成長曲線のようにマクロ的に品質を分析する「統計的手法」が用いられ、品質の向上のためには、個別の欠陥を分析し除去する「要因分析的手法」が用いられている。「統計的手法」では、ソフトウェア全体の品質を捉えることができる。しかし、開発工程における問題領域を特定し根本的解決を図ることが難しい。「要因分析的手法」では、欠陥の発生原因や工程はわかるが、障害件数に比例して、実施が困難となる。この両者の課題を解決すべく、ソフトウェアの障害分析手法である、「Orthogonal Defect Classification（以下、ODC）分析（直交欠陥分類）」を品質検証に活用した。この手法は、個々の欠陥を直交する複数の属性で分類することにより、効率的に原因・傾向を分析するものだ。開発工程の問題箇所を

特定し集中して品質改善をすると共に、ソフトウェアの成熟度をより多角的に評価できる利点がある。この適用事例では、ODC分析の方法論とその有効性、および導入にあたっての取り組みを紹介する。

この適用の結果、ODC分析による障害原因の特定は、開発設計プロセスにおける改善領域を絞り込み主導的に手を打つ事を可能にし、信頼度成長曲線との組み合わせによって、最終ソフトウェアの品質を担保する説明責任を果たすことに寄与した。

ODC分析では、試験を実施した障害発見者によって決定する属性と、開発サイドの障害発見者により決定する属性の両面から、ソフトウェアの品質を多面的に可視化できる。そのため、開発者とテスターの協業が生まれ、ソフトウェアの品質を高めようとするベクトルを一致できる副次的な効果が大きかった。

## B-10 : モデル検査の適用による上流工程での設計誤りの発見

(株)東芝

ソフトウェアのバグの中には、実行時の稀なタイミングでのみ発生するバグや条件の特殊な組み合わせによってのみ発生するバグ等、テスト工程での発見が困難なものが存在する。こうしたバグは、設計工程における設計誤りが原因である場合も少なくない為、テスト工程の対策だけでは多大なコストが発生する。

こうした問題の対策として、モデル検査の適用が考えられる。モデル検査によって上流工程で設計内容を網羅的に検査し、下流工程への設計誤りの流出を防止することで品質を確保できる。特に、組み込みシステムでは、機能的な品質

に加え、信頼性や安全性といった品質が強くと求められることから、網羅的な検査が可能なモデル検査に対するニーズは高い。

この適用事例では、組み込みシステム開発にモデル検査を適用し、品質を確保した取り組みについて紹介する。

この適用の結果、組み込みシステム開発にモデル検査を適用し、実際に設計誤りを発見することができている。モデル検査の適用によって、1プロジェクトあたり、平均でテスト工程コストの約30%を削減できた。

## B-11 : 上流工程の要求を効率的に閉ループシミュレーションする次世代 SILS の開発

トヨタ自動車(株)

大規模化と複雑化が急速に進む自動車制御系開発は、コンカレント開発へ移行が進んでいる。開発と検証を効率的に行うために、プラントモデルの入出力に関して柔軟な設計ができ、実用

的なシミュレーション速度を実現できる必要があった。

現在広く普及している閉ループシミュレーション技術の「HILS（Hardware-In-the-Loop

Simulation)」は、Electronic Control Unit (ECU) 制御系の下流の工程の開発プロセスで用いられている。より高度な要求に対してより効率的に制御系の開発を進めるためには、さらに上流の工程から閉ループシミュレーションを活用する必要がある。この適用事例では、「MILS (Model-In-the-Loop Simulation)」機能

を内包し、かつ「SILS (Software-In-the-Loop Simulation)」としての主要な機能をほぼ網羅した次世代 SILS を開発し、それによって、正確な割り込み処理実行タイミングと高速なシミュレーション速度の両立を実現したことを紹介する。

## 15-B-1 : SysML と CML によるシステムオブシステムズの検証

Newcastle University

システムオブシステムズ (SoS) は複数のシステムから構成され、各システムは個別に管理・運用される。これらの各構成システムは独立した異なるシステムであり、システムオブシステムズとして開発・運用することは困難であると言われている。

SoS 緊急応答システムは、複数の異なるシステム、すなわち緊急応答ユニットのシステム、消防隊のシステム、山岳救助隊のシステム、電話システム、無線通信網、電話通信網などの複数のシステムを管理する統合緊急コールセンター (CUS : Centrale Unica di Soccorso) の

システムで構成されている。

SoS 緊急応答システムの各構成システムは複数の異なる関係者により独自に開発されており、仕様の記述方法や記述のレベルは異なっている。そのため、「SoS システム全体を仕様として形式検証することが困難である」など複数の課題があった。

これらの課題を解決するために、システム記述言語である SysML を使って各構成システムをモデリングし、SysML を形式言語である CML (Compass Modelling Language) に変換し形式検証した。

## 15-B-2 : 通信制御ソフトウェア開発における状態遷移設計の品質向上への取組み ～状態遷移表へのモデル検査の適用～

富士通(株)

今までの通信制御ソフトウェアの開発では、実績がある大量の既存資産を流用 (派生開発) して、その資産に機能追加や変更を行うことで対応してきた。このソフトウェアでは、複数のノード、通信プロトコル、機能ブロック間の連携増加などにより、その設計がより複雑化し、また、検証 (レビュー / 試験) においても、複雑な遷移ルートに伴う問題をすべて検出するこ

とが困難なことから、問題が下流工程に流出し、品質安定までに多くの時間を要するなどの課題があった。

これらの課題を解決し、より短期間で高品質なソフトウェアの提供を可能とするため、設計工程で作成した状態遷移表に「形式手法モデル検査ツール SPIN : Simple Promela Interpreter」を適用し、網羅的に検査した。

## 15-B-3 : 国際スタンダード認証に求められる「要件から検証結果までのトレーサビリティ管理」の効率化の取組み

富士設備工業(株)

安全性が求められるシステムのソフトウェアに対する規格である、ISO26262 (自動車安全規格)、DO-178B/C (航空システムや装置の安全規格)、IEC61508 (電気 / 電子 / ソフトウェアの機能安全に関わる国際規格) 等の国際スタンダードの認証では、開発ライフサイクル全体に

わたる計画的な作業と、それを証明する文書等の成果物 (被認証物) が証拠として要求される。

特にその中でも DO-178B/C では、検証結果までを含めた要件トレーサビリティの管理が求められるが、この作業を人手に頼ることによる課題は、以下の通りである。

- 要求仕様、設計、ソースコード等の様々な成果物と検証結果とのトレーサビリティを管理する作業は複雑で、作業の正確性が問われる。
- 各開発段階の担当者と検証担当者間の情報伝達や意思疎通を欠く恐れがある。
- 開発工数全体の50～80%を占めるといわれる検証作業の工数が増大する。

これらの課題を解決するために、要件から

検証結果までのトレーサビリティ管理の作業を包括的に支援することができるツールを適用した。このツールは、国際標準で要求されるオブジェクト型（達成すべき課題）と被認証物を関連付け、認証プロセスを支援し、開発ライフサイクル全体にわたる計画的な作業を管理する標準的な環境を提供する。この結果、大幅なコスト削減が図れるとともに、品質の向上を図ることができた。

#### 15-B-4 : Friendly による内部 API を使ったシステムテスト自動化

(株) Codeer

大規模で継続的な開発を続けるアプリケーションでは、システムテスト自動化の必要性が増している。このような開発では、機能追加や不具合修正時の影響範囲が読みづらく、デグレードの危険性は、あらゆるところに潜んでいる。そのため品質を保証するには、いち早く問題を検出し、フィードバックするシステム、すなわち日々実行される自動化された回帰テストの存在が必要となる。しかし、そこには問題がある。実は人間の代わりに、プログラムにアプリケーションを操作させるということは、一般に思われているよりも、はるかに困難なプラクティスなのである。一般的にはアプリケーション操作にはキャプチャリプレイツールやUIオートメーションなどのUI操作専門のライブ

ラリを使うことが多いが、それらでは効果的なテスト自動化を作成することができなかった。

この適用事例では、それらとは異なるアプリケーション操作技術として、内部API (Application Programming Interface) をシステムテスト時にも用いる方法を紹介する。

この適用事例では、内部APIはアプリケーション実装時に使用、作成するインタフェース(フィールド、プロパティ、メソッド)を指す。これらは通常そのアプリケーションを実装するときのみ使用可能なもので、外部からは利用できない。しかし、テスト時にこれを使うことで、一般的なGUIの操作性向上はもちろん、コンポーネントテストで行うようなテストビリティを向上させる手法も使用可能となる。

#### 15-B-5 : アジャイルプロセスにおける実践的な品質向上施策の適用事例

(株) 日立ソリューションズ

アジャイルプロセスでは、開発効率を上げるために作業の適切な省略や簡略化を行うが、品質向上のための作業はしっかりと行う必要がある。作業項目が増えても自動化などで作業スピードを上げることで、開発効率を上げると同時に品質確保も行っていく。

ウォーターフォールモデルでは、ユーザ側のテスト工程以降に仕様齟齬による手戻り作業や、利用者の不満が上がってくる場合がある。その解決策として、アジャイルプロセスを適用

する場合もあるが、アジャイルプロセスで品質確保のための施策が不十分なプロジェクトでは、「テスト漏れ」や「システムと仕様との齟齬」、「システムの変更対応でデグレード」などの問題が発生する。

この適用事例では、これらの問題に対処するためハイブリッドアジャイルの適用や品質向上のための工夫を複数のプロジェクトに適用して得られた効果を紹介する。

## 15-B-6：「コストモデル」を使った開発品質・生産性向上の取組み ～バグ対応コストの見える化と最適化～

(株) HS 情報システムズ

品質の向上と生産性の向上に継続的に取り組んでいる中で、多くの開発は委託先に発注してきているため、品質や生産性を把握するのは難しかった。そのため、委託先の開発の品質と生産性を把握する「ものさし」が必要になった。

この「ものさし」は、バグ1件当たりの工程別バグ対応コストと設定し、業務種別や開発規模別に8種類の「コストモデル」を定義した。

工程別の重み付けは、詳細設計は基本設計時の1件あたりのバグ対応コストの5倍、単体テストは10倍、結合テストは15倍、総合テストは20倍、稼動後は200倍のコスト換算とした。

この適用事例では、このような品質と生産性を関連付けた「ものさし」を導入することにより、生産性が向上するとともに品質の良い開発がどこまで可能になったかを紹介する。

## 15-B-7：メトリクス分析手法を用いた試験品質向上の取組み

(株)東芝

製品開発においてQCD(品質/コスト/納期)の確保は当然であるが、工程が逼迫している受託開発においては、納期(D)が優先される場合が少なくない。これは、顧客と合意した期日に対し、納期遅延が顧客にとっても受託側企業にとっても大きな損失になることが理由であることは言うまでもない。

受託開発において、工程遅れの多くは上流設計の遅れであった。その中で、特に開発期間が1年を超える新規開発において、上流工程に時間を費やしてしまう傾向がある。この結果、実装以降の計画が遅れ、納期に向け残された期間で試験を実施しなければならなくなる。このような状況になると、総合試験の期間を確保するため、ソフトウェア間の段階的な結合試験が十分実施できないまま総合試験に進んでしまうことになる。最終的にはシステム試験工程で、本

来結合試験等で解決されているべき不具合が顕在化する可能性が出てくる。

そこで、決められた納期に対し、逼迫している試験工程の中で品質を如何に確保するかについて検討した。入手したデータから試験に関するメトリクスと品質の関係について、「統計的手法」を用いた分析を行った結果、「結合試験の期間がシステム試験時の不具合発生に影響している」ということが、実績データから分かった。その結果、設計者や管理者等関係者に対しソフトウェアの結合試験の位置づけを再認識してもらい、品質向上に繋がる適正な試験期間の指標を設定することができた。

この適用事例では、試験に関するメトリクス分析のアプローチ及びその有用性と試験品質の向上に向けた取組みを紹介する。

## 15-B-8：ユーザーエクスペリエンスを業務に定着化させるための取組み事例の紹介

(株)ベリサーブ

さまざまな製品やサービスにUX(ユーザーエクスペリエンス: User experience 以降、UX)アプローチを適用した経験から、単純な施策で製品やサービスのUXを向上できるものは少ないと感じている。そこで、UXを更に向上させるために、人や組織および業務プロセスへUXアプローチを適用する方法にフォーカスして改善活動を進めてきた。この適用事例では、その改善活動を導入期から成長期・成熟期へと

継続的にする中で、特に導入期から成長期に移行する時に新たなアプローチをすることでUXが業務に定着する方法を紹介する。新たなアプローチとは、対象の業務を兼任しながらUXの活動を牽引しているUXのジェネラリストを配置することで、UXを業務に定着させるためには、このアプローチが必須と結論づけた。その時に適用した技術は、UXの基本概念のもとである人間中心設計プロセス(HCD)だ。

新しいアプローチを以下の2つの内容で紹介する。

- 製品やサービスの UX アプローチを継続するために、UX アプローチの成果を業務プロセスの改善に用いた適用事例。

- UX アプローチと業務を兼務する UX のジェネラリスト向けの参考になるように、提供した UX のサービスの実施例をあげて、それらのアプローチをベストプラクティスにまとめたもの。

### 15-B-9：パッケージソフトウェア開発プロセス改善による品質向上と生産性向上

(株)富士通マーケティング

この適用事例では、統合ビジネスソリューションおよび業務管理ソフトウェアパッケージのソフトウェアの開発における「プロセス改善、品質強化」について紹介する。

バージョンアップを繰り返すパッケージ開発、かつソフトウェアプロダクトの分業開発において、各社間で設計手法や設計書が不統一で、テスト品質が不安定であった。また、開発規模が大きいにも係らず開発スピードが求められるため、障害が増加する問題があった。さらに、ソースプログラムについては、記述が複雑なため影響範囲が不明で、重複プログラムに起因する修正漏れやインタフェースの不整合などの問題があった。

これらの課題を解決するため、以下に示す3

つの対応を実施した。

- テスト駆動開発に近い手法を採用し、テスト設計をプログラミングの前に行い、通常の開発プロセスの順序を入れ替えて設計手法や設計書を統一した。
- タイムボックスという開発手法を採用し、すべての作業にタイムボックスを設定し繰り返し開発した。タイムボックスの手法は、「時間は動かさない」と宣言することで、開発遅延をなくすやり方である。
- ソースプログラムをリファクタリングすることで、保守をするうえで手に負えないソースプログラムを分かりやすく保守しやすいソースプログラムに洗練させた。

### 15-B-10：Web システムにおける単体テストの品質向上の取組み

住友電工情報システム(株)

長年、ソフトウェア開発の品質改善活動と生産性向上活動に継続的に取り組んでいる。その中で、以下のような品質面の問題があった。

- 現行仕様に更改仕様を反映させる時に仕様の理解不足で適切なテストができず、システムのデグレードが発生した。
- テスト工程にて、妥当なテストデータの用意や適切なテスト手順を実施できず、テ

スの品質を確保できなかった。

この適用事例では、Web システムの開発において、テストファースト手法（テスト設計、コーディング、テストの順に実施すること）の導入、自動テストの導入、および単体テストに用いるテストデータの自動生成システムの構築により、品質が大幅に向上したことを紹介する。

### 15-B-11：安心なサービスの品質改善を実現する為の継続的システムテスト

楽天(株)

ウェブサービスの検索システムの開発においては、検索ランキングを継続的に改善していくための開発とテストプロセスを迅速かつ最小限のリスクで実施していく事が重要である。継続的インテグレーション (CI) の構築等により一定の工数削減等に成功したが、一方で以下のよ

うな問題を抱えていた。

- 手動のシステムテストでは依然、バグ検出時の手戻りが大きい。
- 検索ランキングの評価をプロジェクト終盤まで実施できない。

これらの問題を解決するため、機能性だけでなく運用性や可用性のテスト、また検索結果の評価等を行うシステムテストも自動化し、開発プロセスに組み入れる取り組みを行った。

システムテストの自動化では、アプリケーションのデプロイやテストの事前準備などの手順を自動化し、テストの自動実行ツールを開発した。また派生開発におけるソースコードと同じように発展性が求められるテストスクリプトについて、共有スクリプトの考え方を取り入れ、保守性の高いドメイン固有テスト言語 (DSTL) を定義した。

次に自動システムテストと開発プロセスを相

互作用的に改善していく為の取り組みを実施した。これには欠陥特定にかかる時間の効率化やテストの失敗が機能やテストの実装に与える影響の最小化のための取り組みなどが含まれる。加えて、開発プロセスの改善を継続的に行う為のメトリクスの収集やフィードバックの仕組みを取り入れた。我々はこれらの取り組みを総称して継続的システムテストと呼んでいる。

これら一連の継続的システムテストについての取り組みにより、システムテストで検出された欠陥の修正日数の中央値が5日から2日へと改善された。また、スプリントの初期の段階で多くの欠陥を検出可能になった。

## 15-B-12：セキュア開発手法の考察と診断ツールの活用事例の紹介 ～お客様に「安心してご利用ください」と言えるための脆弱性対策～

ビッグローブ(株)

お客様に安心して安全な Web サービスやクラウドサービスを提供するためには、既知の脆弱性を残さぬようセキュアな開発と検証を行わなければならない。しかし、プロジェクトの予算や開発規模も多種多様であり、検査専門会社に委託するなど一回数十万～100万円以上かかる手法で一律化するのには、事業として現実性を損なってしまう。ツールを用いて診断を内製化するにしても検査作業の工数(コスト)は消費するので、一律一律な検査方法を義務化する事も現実的には難しい。そこで、検証精度の属人化

は抑止しつつも、対象システムの規模感や予算感に合わせた検証法を選択できるようにしなければならない。

このような低コストや短納期な開発計画でも相応の検証と対処ができるようにするために、適切な精度やツールの整備が必要である。この適用事例では、お客様の安心や安全を維持するための最適な開発手法を模索や整備し続けている。その活動を通じて見えてきた脆弱性の仕分け方と診断手法の工夫および診断ツールの活用法を紹介する。

## 15-B-13：モデル検査とテストによる車載オペレーティングシステムの検証

北陸先端科学技術大学院大学

車載ソフトウェアの安全性や信頼性に関する問題は、社会において非常に大きな関心となっておりつつある。車は、従来は機械的に制御されてきたが、近年、コンピュータ制御技術の発展と利便性や性能の追求により、多くの部品の電子化が進んできている。これにより、車載ソフトウェアの規模の急速な増大と複雑化がもたらされ、主に、電子制御部分の安全性と信頼性に関する問題が取り上げられつつある。世界標準においては、機能安全に関する標準が一般の電子システムだけでなく、車載システムに特化されたものが策定されている。また、実社会においては、2010年に発生したトヨタ車の急加速問題において、電子スロットル制御システムの検証が NHTSA (National Highway Traffic Safety

Administration：米国運輸省道路交通安全局)と NASA により実施された。

この適用事例では、このような車載ソフトウェアの安全性や信頼性の問題を背景に、車載オペレーティングシステムの検証手法の研究と実践を紹介する。この取り組みの目的は、車載オペレーティングシステムを次世代の車に組み込むため、形式手法を適用することにより、さらに高い品質を達成することである。一方、形式手法を現実的なシステム開発に適用するためには、様々な工夫が必要であり、一筋縄では形式手法の産業応用ができないことも多々ある。そこで、形式手法の実践課題を解決しながら、車載 OS の検証を行った。

## 15-B-14：モデルベース開発とコード解析を用いた組み込みソフトウェアの開発

アルプス電気(株)

自動車の高機能化や電子制御化に伴い、一台の自動車に搭載される ECU（電子制御ユニット：Electronics Control Unit）の数は年々増加し、また要求される機能も複雑化している。ECU の増加に伴い、実現するソフトウェアにおいては、ソースコード量の増加や ECU 間のネットワーク制御が必要になり、ソフトウェアの品質が全体の性能や信頼性に大きく影響を与えるとともに、設計や検証コストが増大する課題が表面化してきた。

この適用事例では、これらの課題を解決するために、「モデルベース開発」と「コード解析」を適用した独自の手法である MDD（Model

Driven Development）に TDD（Test Driven Development）の思想を取り入れ、モデルベース開発とテスト駆動開発を融合させプロセスの洗練化を図ったものを活用した。その結果、従来開発に比べて開発工程が減ることでコストが削減でき、視覚的に理解しやすいモデルを検証することで、検証効率と品質を向上することができた。また、コード解析では、市販コード解析ツールと自社開発ツールを組み合わせることで、人手で行っていた検証作業をツールに置き換え、スキルによらない検証が行え、検証コストの削減につなげることができた。

# 付 録

..... 184

付録 1：事例ごとの適用結果一覧表.....	184
付録 2：適用事例分類表.....	187

※付録 2 は、IPA/SEC のウェブサイト

<http://www.ipa.go.jp/sec/publish/tn16-001.html> からダウンロードできる。



付録 1：事例ごとの適用結果一覧表

事例 参照 番号	システム及びソフトウェア品質モデル (JIS X 25010:2013)													ア シ ユ ア ラ ン ス ( 保 証 )	障 害 原 因 の 分 析	コ ス ト	納 期	生 産 性 ( 対 応 時 間 短 縮 )	人 材 育 成 改 善	マ ネ ジ メ ン ト プ ロ ジ ェ ク ト	見 積 支 援	普 及 促 進	体 制 ( 強 化 ・ 再 構 築 )	グ ロ ー バ ル 展 開
	製 品 品 質						利 用 時 の 品 質																	
	機 合 能 適 性	性 能 効 性	互 換 性	使 用 性	信 頼 性	セ リ キ ュ イ	保 守 性	移 植 性	有 効 性	効 率 性	満 足 性	リ ス ク	回 避 性											
1	A-1	■	■	■	■	■	■	■	■	■	■	■	■	□	■	■	■	■	■	■	■	■		
2	A-2	□	■	■	■	■	□	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■		
3	A-3	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■		
4	A-4	□	■	■	■	■	□	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■		
5	A-5	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■		
6	A-6	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■		
7	A-7	□	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■		
8	A-8	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■		
9	A-9	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■		
10	A-10	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■		
11	A-11	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■		
12	A-12	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■		
13	A-13	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■		
14	15-A-1	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■		
15	15-A-2	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■		
16	15-A-3	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■		
17	15-A-4	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■		
18	15-A-5	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■		
19	15-A-6	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■		
20	15-A-7	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■		

事例 参照番号	システム及びソフトウェア品質モデル (JIS X 25010:2013)														ア シ ユ ア ラ ン ス ( 保 証 )	障 害 原 因 の 分 析	コ ス ト	納 期	生 産 性 ( 対 応 時 間 短 縮 )	意 材 育 成 革	プ ロ シ ェ ク ト	見 積 支 援	普 及 促 進	体 制 ( 強 化 ・ 再 構 築 )	グ ロ ー バ ル 展 開
	製 品 品 質							利 用 時 の 品 質																	
	機 能 適 性	性 能 効 性	互 換 性	使 用 性	信 頼 性	セ リ キ ュ イ	保 守 性	移 植 性	有 効 性	効 率 性	満 足 性	リ ス ク	回 避 性	利 用 状 況											
21	15-A-8								■	■	■	■	■			■	■	■	■	■	■				
22	15-A-9					□							□			■			■	■	■				
23	15-A-10																		■						
24	15-A-11	□																		■	■				
25	15-A-12		■			□													■	■	■	■	■		
26	15-A-13		■																■						
27	15-A-14	□	■											□						■	■	■	■		
28	15-A-15	□	■											□						■					
29	15-A-16	□			□	□	□													□	□				
30	15-A-17	□	■																	■	■				
31	15-A-18	□	■			□														■	■	■	■		
32	15-A-19					□														■					
33	15-A-20					□														■					
34	B-1		■		■															■					
35	B-2																								
36	B-3		■		□	□	□	□	□	□												■			
37	B-4		■		□	□	□	□	□	□											■	■	■		
38	B-5																								
39	B-6		■																				■		
40	B-7					□																			



事例参照番号	システム及びソフトウェア品質モデル (JIS X 25010:2013)													アシユアランス (保証)	障害原因の分析	コスト	納期	生産性 (対心臓短縮)	人材育成	マネジメント プロジェクト	見積支援	普及促進	体制 (強化・再構築)	グローバル展開
	製品品質						利用時の品質																	
	機能 適合性	性能 効率性	互換 性	使用 性	信頼 性	セキュ リティ	保守 性	移植 性	有効 性	効率 性	満足 性	回避 性	回復 性											
41	B-8	□	■	□																				
42	B-9	■	■											□										
43	B-10	□	■																		△			
44	B-11	■	□											□							△			
45	15-B-1	■	■			□		□						□							△			
46	15-B-2	□	■											□		□					△			
47	15-B-3	■	■					□						□							△			
48	15-B-4	□	■																		△			
49	15-B-5	□	■											□		□					△			
50	15-B-6	□	■			■						■		□		□					△			
51	15-B-7	□	■											□		□					△			
52	15-B-8	□	■											□		□					△			
53	15-B-9	□	■			■															△			
54	15-B-10	■	■					□						□		□					△			
55	15-B-11	□	■					□													△			
56	15-B-12	□	■					□						□		□					△			
57	15-B-13	□	■			■								□		□					△			
58	15-B-14	□	■											□		□					△			



## 著作監修者紹介

### 独立行政法人 情報処理推進機構 (IPA) 技術本部 ソフトウェア高信頼化センター (SEC)

IPA/SEC は、IT の活用による快適なくらしや豊かな情報社会を実現するため、社会全体を支える情報処理システムの信頼性向上に取り組んでいます。

<http://www.ipa.go.jp/sec/index.html>

所在地 〒 113-6591 東京都文京区本駒込 2-28-8 文京グリーンコート センターオフィス  
電話 03-5978-7543 FAX 03-5978-7517

#### 執筆・編集

---

中尾 昌善	IPA/SEC
室 修治	IPA/SEC
春山 浩行	IPA/SEC (TIS 株式会社)
佐々木方規	IPA/SEC
神谷 慎吾	IPA/SEC
齊藤 善治	IPA/SEC
藤原由起子	IPA/SEC
鈴木 恭子	IPA/SEC

#### 協力者 (敬称略) ※所属は当時のもの

---

##### 2013 年度 高信頼設計・検証技術 WG

主査	山本修一郎	国立大学法人名古屋大学
	秋山 浩一	富士ゼロックス株式会社
	大岩 寛	独立行政法人産業技術総合研究所
	佐野 希芳	東芝情報システム株式会社
	新谷 勝利	新谷 IT コンサルティング
	鈴木 尚志	株式会社コギトマキナ
	細川 泰秀	一般社団法人日本情報システム・ユーザー協会
	鷺崎 弘宜	早稲田大学

##### 2014 年度 開発手法適用のための分析 WG

主査	山本修一郎	国立大学法人名古屋大学
	秋山 浩一	富士ゼロックス株式会社
	小笠原秀人	株式会社東芝
	新谷 勝利	新谷 IT コンサルティング
	鈴木 尚志	株式会社コギトマキナ
	鈴木 延保	アイシン・コムクルーズ株式会社
	英 繁雄	株式会社日立ソリューションズ
	細川 泰秀	一般社団法人アドバンスト・ビジネス創造協会
	山田 敏行	株式会社日新システムズ
	鷺崎 弘宜	早稲田大学

SEC BOOKS

## 事例に見る先進的な設計・検証技術の適用分析 ～高信頼化のための開発技術導入に向けて～

---

2016年5月11日 1版1刷発行

監修者 独立行政法人情報処理推進機構 (IPA)  
技術本部 ソフトウェア高信頼化センター (SEC)

発行人 松本 隆明

発行所 独立行政法人情報処理推進機構 (IPA)  
〒113-6591  
東京都文京区本駒込 2-28-8  
文京グリーンコートセンターオフィス 16階  
URL <http://www.ipa.go.jp/sec/index.html>

© 独立行政法人情報処理推進機構 技術本部 ソフトウェア高信頼化センター 2016

---

ISBN978-4-905318-39-2 Printed in Japan

ISBN978-4-905318-39-2  
C3055 ¥926E



9784905318392

定価:926円+税



1923055009264



独立行政法人情報処理推進機構  
技術本部 ソフトウェア高信頼化センター