

SEC

journal

33

巻頭言

和田 成史 一般社団法人コンピュータソフトウェア協会 (CSAJ) 会長

所長対談

加藤 光明 伊藤忠テクノソリューションズ株式会社 取締役 常務執行役員 CIO

真に利用者に求められるソフトウェアの提供と それを実現するエンジニア像について考える

特集

SEC 2012年度活動概要

<統合系>

統合システムの安全・安心に向けた2012年度の取組み
ソフトウェア品質説明力強化の取組み／上流品質技術強化 形式手法普及活動の取組み
上流品質技術強化 コンシューマデバイス機能安全規格化の取組み／上流品質技術強化 MBSE 導入の手引き

<エンタプライズ系>

ビジネス・プロセス改善の推進／超上流及び上流工程の品質向上に向けて
定量的プロジェクト管理の推進／定量的な目標管理手法の普及活動の展開
非ウォーターフォール型開発手法の普及に向けて／IT サービス継続計画の策定を推進

<組込み系>

組込み系ソフトウェアプロジェクトの状況

トピックス

SECにおける第二期中期計画の総括と第三期中期計画の概要について

連載

FP計測手法におけるFP規模と工数の相関の差

論文

Javaプログラムのアクセス修飾子過剰性分析ツール
ModiCheckerの機能拡張とその応用例

トピックス

SEC特別セミナー「IV&V最新動向」について

報告

ソフトウェア工学分野の先導的研究支援事業について

Column

コンピュータは「ソフトウェア危機」を救えるか？

巻頭言 ……49

和田 成史 一般社団法人コンピュータソフトウェア協会 (CSAJ) 会長
パッケージソフトウェア品質認証制度の発足に向けて

所長対談 ……50

加藤 光明 伊藤忠テクノソリューションズ株式会社 取締役 常務執行役員 CIO
真に利用者に求められるソフトウェアの提供とそれを実現するエンジニア像について考える

SEC2012年度活動概要 ……54

統合系 ……56

統合システムの安全・安心に向けた 2012 年度の取組み
ソフトウェア品質説明力強化の取組み
上流品質技術強化 形式手法普及活動の取組み
上流品質技術強化 コンシューマデバイス機能安全規格化の取組み
上流品質技術強化 MBSE 導入の手引き

エンタプライズ系 ……63

ビジネス・プロセス改善の推進
超上流及び上流工程の品質向上に向けて
定量的プロジェクト管理の推進
定量的な目標管理手法の普及活動の展開
非ウォーターフォール型開発手法の普及に向けて
IT サービス継続計画の策定を推進

組込み系 ……73

組込み系ソフトウェアプロジェクトの状況

トピックス ……79

杉原井 康男 SEC 次長 田中 薫 SEC 企画グループ
SEC における第二期中期計画の総括と第三期中期計画の概要について

連載 定量データ分析その4 ……85

古山 恒夫 東海大学理学部 IPA/SEC 専門委員
FP 計測手法における FP 規模と工数の相関の差

論文 ……90

小堀 一雄、石居 達也、松下 誠、井上 克郎
Java プログラムのアクセス修飾子過剰性分析ツール ModiChecker の機能拡張とその応用例

トピックス ……98

香西 省治 SEC 統合系プロジェクト 研究員
SEC 特別セミナー「IV&V 最新動向」について

報告 ……100

小沢 理康 SEC 調査役
ソフトウェア工学分野の先導的研究支援事業について

コラム ……102

コンピュータは「ソフトウェア危機」を救えるか？

書籍紹介 ……103

編集後記 ……104

SECjournal 論文募集 /IT パスポート試験 (iパス) のご案内

パッケージソフトウェア品質 認証制度の発足に向けて

一般社団法人コンピュータソフトウェア協会（CSAJ）会長
株式会社オービックビジネスコンサルタント 代表取締役社長

和田 成史



一般社団法人コンピュータソフトウェア協会（CSAJ）では、今年6月に「パッケージソフトウェア品質認証制度」（略称「PSQ 認証制度」）を開始しました。これは、国際規格であるISO/IEC25051及び日本工業規格JIS X25051「ソフトウェア製品の品質要求及び評価—商用既製（COTS）ソフトウェア製品に対する品質要求事項及び試験に対する指示」に準拠した仕組みで、ソフトウェア製品の品質を認証する制度です。

ソフトウェア製品は、触れることができない財（無形財）であり、かつ経験財であるので、使ってみなければその品質や機能・性能の良さは確認できません。一般的に、ソフトウェア製品を購入しようとするユーザー（企業や個人）は、その製品のパンフレットや製品説明書、あるいはマニュアルを頼りに製品を選択し、購入を決定します。場合によっては、販売員等から直接説明を受けることもあれば、デモを見ることもあるかも知れませんが、製品説明書やマニュアルが重要な役割を持っていることには変わりはありません。したがって、こうしたソフトウェア製品のドキュメントに必要な情報がきちんと記述されていて、かつその情報通りにソフトウェアが動作するかどうか、ユーザーにとっては非常に重要になってきます。

PSQ 認証制度は、こうしたソフトウェア製品という財の性質を考えて、次の二点を第三者として確認します。第一は、製品説明書やマニュアルにユーザーが必要とする情報、特に品質に関する情報がきちんと記述されているかどうかで、第二は、製品説明書やマニュアルに書かれているとおりにソフトウェアが動作するかどうかです。適合性が確認できた製品やその説明書には、PSQ マークを付けることができます。

つまり、そのソフトウェア製品を購入、利用しようとするユーザーが参照するドキュメント類に、そのソフトウェアの機能性、信頼性、保守性、効率性、使用性、移植性などの様々な品質特性がきちんと記載されていて、それが実際にソフトウェアとして実現されているかどうかを確かめて、その証としてPSQ マークというお墨付きを付与するという仕組みがPSQ 認証制度なのです。

この仕組みによって、ユーザーは製品説明書などのドキュメントを信頼し、安心してソフトウェア製品を選択できるようになります。また、国際市場における日本のソフトウェア製品の品質の高さをアピールすることもできるでしょうし、同時に日本のソフトウェア製品の品質底上げにも役立つだろうと思います。そして、国内外のパッケージソフトウェア市場の拡大にも寄与できると考えています。

CSAJでは2010年度にソフトウェア品質基準に関する研究を開始し、それが今回のPSQ 認証制度という成果につながったのですが、ほぼ同じ時期に情報処理推進機構（IPA）のソフトウェア・エンジニアリング・センター（SEC）でも「ソフトウェア品質監査制度（仮称）」の検討が進められてきました。実は、PSQ 認証制度で用いる評価基準はSECの活動の中で作成していただいたものであり、また、PSQ 認証制度の実証実験もSECの事業として実施していただきました。そしてPSQ 認証制度は、6月にIPAから発表された「ソフトウェア品質説明のための制度ガイドライン」に沿った認証制度としてスタートすることができました。今後もSECとの協力関係を維持・強化しつつ、PSQ 認証制度の適正な運用を通じ、日本のソフトウェア品質向上のために貢献していきたいと考えています。

真に利用者に求められるソフトウェアの提供とそれを実現するエンジニア像について考える

伊藤忠テクノソリューションズ株式会社
取締役 常務執行役員 CIO

加藤 光明



SEC 所長

松本 隆明

ITの利用方法が多様化し、同時に新たな技術がすさまじいスピードで登場している今、ITシステムの運用フェーズでは、開発時に想定し得なかった事象が起こりうる。運用も含めたITシステムのライフサイクル全体でソフトウェアの開発を考えていくことがこれまで以上に求められる。利用者が求めるニーズや品質にどう答えていくか、現状と今後の課題・解決策を探っていく。

松本： 昨今、社会生活にとってITが必須かつ身近な存在になると共に、利用者の幅も広がってきました。このため、ひとたびITに障害が起きると、社会生活に非常に大きな影響を与えることとなります。利用形態が多様化する中で、ITシステムの品質、信頼性を担保することが大きな課題となります。ITシステムの場合はソフトウェアが多くをコントロールしているわけですが、御社ではソフトウェアの品質確保についてどのようなお考えをお持ちでしょうか。

加藤： 弊社は、海外ベンダーからお客様のニーズに沿ったハードウェア・ソフトウェア製品、及びその関連技術を探してSIやサービスにおいて活用することを得意としていますが、日本のお客様は非常に高い品質を製品に要求されるため、これまで日本のマーケットは品質の試金石とも言えました。「日本のお客様を満足させられれば他の国へ行っても大丈夫」という海外のベンダーもいたほどです。ところが昨今では、日本における品質の要求は他国では必要としないほどの水準になり、海外ベンダーには日本のお客様をモニターの対象としないケースも出てきています。しかし、弊社のお客様の中にはシステムの信頼性に関

して極めてシビアなお客様もおられ、日本での競争を勝ち抜くためには、やはり日本のお客様の高い要求水準に個別に合わせて、製品やSI、サービスを提供していく必要があります。この観点から、弊社では品質対策として、全社的な品質対策、ビジネスタイプごと、つまり、SI、サービス、製品の提供といったビジネスごとの品質対策、個別顧客ごとの品質対策といった、階層的な対応を取ってきめ細かい品質確保に努めています。端的に言えば個別顧客のニーズに応える品質確保を目指しているということになります。

松本： 御社はSIerでありながら運用・保守サービスに重点を置いているそうですね。運用・保守サービスを重視する狙いは何ですか？

加藤： 従来、弊社は「製品が強み」というイメージが先行していたように思いますが、継続して注力してきた運用・保守サービスも大きな強みです。お客様は利益や効率化といった「効果」に対価を払っています。効果を上げるのは実際にITシステムが運用を開始してからとなり、つまりは運用・保守サービスに直結します。当然ながらここへの品質要求は極めて高くなるわけですが、弊社が運用・保守サービスに注力する理由は2つ。ひとつは、お客様に対して「単なるサプライヤ」や「単なるITの供給者」には留まりたくないという考えからです。ITを通じたビジネスパートナーとして、お客様のビジネスに直接かかわるサービスやソリューションを提供していく、という強い思いがあります。もうひとつは、運用・保守サービスはお客様のパートナーとして継続的に付き合いしていくことになり、長期的に安定した収益を見込めるという点があります。

松本： いずれも重要な観点ですね。企業である以上、収益の基盤を確保しなくてはなりません。また、ソフトウェアの場合、「製品を提供しておしまい」ではお客様の信頼は得られ



加藤 光明 (かとう みつあき)

1975年早稲田大学理工学部電気工学科卒業。同年伊藤忠商事株式会社へ入社。2006年、伊藤忠テクノソリューションズ株式会社へ移籍。2008年に同社執行役員へ就任。2011年取締役常務執行役員、2013年4月より、CIOも兼務している。

ないでしょう。運用・保守サービスに注力する上で、重要視しているポイントは何でしょうか。

加藤：運用・保守サービスにおいては、弊社では顧客視点・品質視点から3つのポイントを重視して対応しています。1つ目は、ITシステムを導入する際に、お客様がどこからでも安心して活用できるようなメニューを作っていることです。開発から運用へとワンストップでご提供する場合もあれば、運用だけ、あるいは保守から運用などと部分的な場合もあります。開発時点や、運用の設計時、運用中など、あらゆる時点で運用に関する基準を設け、受注／設計／開発の各段階で定期的に内部監査が入る仕組みを作っています。2つ目は、様々なISOの資格を取得していることです。ISO27001/ISMS・ISO9001・ISO20000・Pマーク・ISO14001・SAS70・SSAE16などを取得して対応しています。また、“運用・保守で最も大事なものは人”という認識から、3つ目は、ヒューマンエラーを減らすための活動です。HEZ (Human Error Zero) 活動を徹底しており、実際の運用オペレーションの際、システムによっては2人で手を組んで指差し確認を行っています。HEZ活動に関する社内セミナーを実施していますが、実はこの活動はニーズが高いため研修サービスを作ってしまったほどで、競合他社にも提供しているという状況です。

松本：やはり、運用・保守も含めたライフサイクル全体で品質を確保するという考え方は欠かせないものになってきますね。運用・保守のフェーズを考慮すると、特に利用者の視点が大切ではないでしょうか。

加藤：弊社は、コンビニやビール会社など運用の品質がビジネスの根幹をなすような企業にもサービスをご提供しています。品質にシビアなお客様に対しては一緒に基準を決めていきます。アプリごとに決める場合もあります。ただし、効率を上げるためには標準化も必要です。いわゆる標準と個別のバランスが顧客満足度確保の面では非常に難しいですね。

開発フェーズから運用設計に取り組む

松本：運用時に問題が起きてくると、開発にフィードバックをしなくてはなりません。その場合に何が重要となりますか。

加藤：運用と開発を切り離すと品質を担保しづらくなるため、開発の段階から運用のケアをすべきです。弊社では営業グループごとに運用設計要員がおり、開発プロジェクトの初期段階から参画させます。開発の段階で決めるサービス仕様書は、開発要員と運用設計要員の間で情報をシェアします。必要な対応・設計、準備すべき体制などを早い段階でお互いに検討する仕組みがあるわけです。さらに、現在構築中ですが運用・保守業務で蓄積された知識・知見をデータベース化し

ています。標準的な内容よりも、優れた運用者の知見、個人のノウハウを集めるのが目的です。また、運用要員と開発要員のローテーションも定期的に行っています。開発した要員が当面の間は運用も担当し、ある程度うまく運用が回ってスキルトランスファーできた段階で開発に戻るのです。

松本：開発時点で、運用の段階まですべて網羅するのは難しいものです。想定していない使い方を利用者がすることもありあるでしょう。運用で経験を積んだ人が開発に回れば、実際の使い方がある程度想定できますね。

加藤：個人的な考えですが、プログラミングを経験した人間は、良いプログラム設計ができるという認識があります。また、良いプログラム設計の経験があれば、良いシステム設計ができ、良いシステム設計ができると、良いアイデアが生まれる……。お客様の言葉を鵜呑みにせずに、システムでできる部分とできない部分をうまく切り離して考えるなど、優秀なIT人材が育つのではないかと思います。同様に、良い運用を経験すると、良い開発ができるのではないかと思います。

松本：開発時点で想定していた環境と、サービスインしたあとの環境で世の中の情勢が変わっていることも多いですが、将来の環境変化に対するリスクはどの程度見込んでいますか。

加藤：近い将来とはいえ、ビジネス環境の変化への対応をシステム開発中に見込むことは正直言って難しい。本音を言えば超上流工程において、お客様主体で十二分に考慮していただきたいと思います。一方でIT環境の変化に対しては、弊社ビジネスへの影響は常々考えています。弊社のビジネス構造はざっくり言うと「運用・保守サービス」「SI(開発)」「製品販売」に分かれますが、それぞれの商品が依存する環境や技術をカテゴライズしています。「Aの環境が変化したらBの案件/ビジネスに影響がある」「Cの技術が変化したらDの案件/ビジネスに影響がある」と予測ができます。さらに、影響を受けるビジネスの利益や売上の規模がある程度把握できます。結果、新しい環境や新しい技術によるマイナスのインパクトが見えるのです。一方でプラスが



松本 隆明 (まつもと たかあき)

1978年東京工業大学大学院修士課程修了。同年日本電信電話公社(現NTT)に入社、オペレーティング・システムの研究開発、大規模公共システムへの導入SE、キャリア共通調達仕様の開発・標準化、情報セキュリティ技術の研究開発に従事。2002年に株式会社NTTデータに移り、2003年より技術開発本部本部長。2007年NTTデータ先端技術株式会社常務取締役。2012年7月より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)所長。博士(工学)。

読みにくいという課題もあります。どれだけビジネスを伸ばせるのかは分かりにくく、悩みの種でもあります。

松本：御社はテクニカルソリューションセンターという検証センターをお持ちですが、主な役割は、製品の技術評価ですか。

加藤：弊社は海外からより良い製品を調達し、調達した製品を組み合わせてお客様に提供しています。組み合わせた時には、個別のベンダーが品質を保証するわけではありません。組み合わせた時に効果を発揮できるか、品質は十分に担保できるかは、弊社が検証しています。お客様にも総合的な検証施設であるテクニカルソリューションセンターを開放しており、導入を検討している製品で、お客様自身のアプリが動くかどうかを検証する場合があります。また、場合によっては製品単体の技術評価もしています。

松本：お客様に近い環境を提供できるのは素晴らしいですね。昨今、IV&V という言い方もしますが、ソフトウェアの品質を開発者以外の第三者が検証できる仕組みが必要になっています。検証する側は環境を揃えるのが大変ですが、御社は自社に検証環境をお持ちで、お客様に開放しているのは画期的な取組みではないでしょうか。

エンジニアに求められるスキルが変化

松本：先ほど人材が重要だという話がありました。単に開発ができるだけでなく、お客様の真の要望、想定している使い方などを踏まえながらシステム設計や新たな提案をすることが求められていくでしょう。

加藤：お客様のニーズに、より近いところで技術的な対応が要求されますから、それに長けた技術者がが必要です。ただ、一朝一夕に人は育つものではありません。いわゆる PM や IT アーキテクトは対象が明確なので育成しやすいと言えますが、BA（ビジネスアナリスト）は難しい。真のニーズや目的をいかに引き出すかは難しいポイントだと思います。SEC では、導入する IT プロジェクトにビジネスとの整合性を持たせる方法論「GQM + Strategies」を普及促進しておられますが、とても期待しています。お客様は「在庫を減らしたい」と言いますが、そもそもなぜ在庫を減らしたいのかと突き詰めると在庫システムを作ることが本当の解ではないかも知れません。ところが、今までの SIer は「分かりました。在庫システムを作れば在庫がこれだけ減ります」と進めてしまうのです。本質まで突き詰められる技術者が十分にいれば、SIer として大変な強みになるでしょう。

松本：GQM+Strategies を評価していただいてありがとうございます。おっしゃる通り、GQM+Strategies でも、本当の目的を明確にして、ゴール指向で進めていくことがポイント

になります。ただし、エンジニアにそこまで求めるのは難しい部分があるかも知れません。ビジネスベースの議論になると、どうしても引いてしまうでしょう。本来は、お客様が超上流や上流工程に参画して、一緒に考えていく必要があると思います。お客様の情報システム部門が音頭を取って、ビジネスチャンスを拡大するくらいの方向へ持って行ってほしいですね。

加藤：ビジネスチャンスを拡大するシステム、必要となるコストが経費ではなく売上原価として捉えられるようなシステムの企画——それができなければ、CIO の認知がなかなか上がらないとも思っています。

松本：情報システム部門の意識変革も必要でしょう。開発フェーズに入ると丸投げになってしまうケースが多いなか、IT システムの目的をベンダーと一緒に考えなくてはいけないのではないのでしょうか。御社ではエンジニアスキル認定制度、人材の資格制度を導入していますが、どのような制度なのでしょう。

加藤：弊社では、ITSS（IT スキル標準）のように、ある程度定義できているような技術と、ベンダーやコミュニティから発生する先端的な技術の 2 つを分けて考えています。IT の技術認定としては、どちらかというと普遍的な技術である前者に対する技術評価を行おうというもので、時代のニーズに合わせて逐次改訂して実施しています。一方で、後者については対象とすべき技術、技術評価の方法などを現在検討しています。このスキル認定と人事評価をリンクさせ、非役職者でも能力の高い技術者は相応に処遇するようになっていきます。

松本：ビジネススキルは、認定項目に入るのですか？

加藤：競争力の源泉ですから、もちろん入っていますし、さらに充実させようとしています。座学以外のトレーニングも考えています。短期的にはお客様のところへ出向することも検討しています。

松本：それは大事な試みですね。ソフトウェア業界は、たびたび建築業界と対比されます。建造物では、設計者が非常に注目され、後生まで名前が残ります。しかし、IT システムでは聞いたことがありません（笑）。もう少し、設計した人が脚光を浴びても良いのではないかと思います。

加藤：そうなれば、エンジニアもモチベーションが上がりそうですね！

単なるインフォメーションではなく インテリジェンスへ昇格させたい

松本：加藤さんは 4 月から CIO になられました。情報システム部門を引っ張っていることにはなりますが、CIO になって最も苦労している点は何ですか。

加藤：大事なものはCIOの「I」の部分ではないでしょうか。多くの方は「インフォメーション」という認知しかないでしょう。CIOはいかにIT部門のコストを抑えたかという点が評価のポイントになりがちです。しかし、「I」を「インテリジェンス」に昇格させなくてはいけないと思います。ビジネスとして攻めるべき領域、ITを活用する方策、その部分に本来はCIOの職務権限が与えられるべきでしょう。残念ながらそこまで明確な職務権限を持っているCIOは少ないと思います。弊社の場合、私の声が大きいほうなので無理矢理こちらに顔を向かせているところはありますが（笑）。世間一般では難しいかも知れません。

松本：CIOが「チーフインテリジェンスオフィサー」という発想は良いですね。どうしても、CIOや情報システム部門は軽視されているような感があります。

加藤：ただ、最近BIなどが注目されているのは良いことだと思います。CIOがBIに注目して、COOに対して効果を提示していくなど、積み重ねて評価を上げていくしかないでしょうね。

松本：ビッグデータがもてはやされているのは大きなチャンスですね。情報システム部門がデータを処理することで新しいことが見えてくる、と発信できれば、ITが経営やビジネスに役立つと認められ、存在価値が高まります。

加藤：私は学生の頃から数えて40年間この業界におりますが、先ほどのように、縁の下の力持ちから抜けきれないのが実状です。少子化で若い人たちや女性を有効に活用しないといけないときに、今のような位置づけではなかなか良い人材が集まりません。ですから、若い人たちがITに目を向けてくれる方法を常々考えています。今年、私は採用の最終面接をしていますが、世に言われるような「今の若者は意欲がない」という実感は全くありません。むしろ若者は大変な意欲を持っているのです。それなのに、就職して年を重ねるにつれ現実的になりすぎてしまう。正直、若者の大きなビジョンを我々が潰しているのではないかとさえ思います。

松本：確かに。IT業界は徒弟制度から抜けきれない部分があります。先輩の仕事のやり方を見本にして、同じように進めたいというまくい、という考えがあります。若い人ならではの豊かな発想が抑制されてしまうことはあるかも知れません。一方で、トラブルのあったときに「このあたりが怪しそうだ」と勘の働く人もいます。実際に検証してみると、指摘した場所にバグがあって驚くものです。経験に裏打ちされたスキルも決して侮れないところがあります。

加藤：以前、ある方が「データサイエンティスト」が今後のキーワードになる」という話をされていました。私が「お客様のビジネスに対して大事なものを見極めるためには、統計学的なスキルも大切だが、突き詰めるとインスピレーシ

ョンが必要ではないのか」と考えを述べたところ、彼はひとこと「Yes」と答えた。それが印象的でした。インスピレーションを持った人材を潰さないで育成していくことが重要だと、面接をしていてつくづく思いました。この人達を大事にしたい、と心から思います。

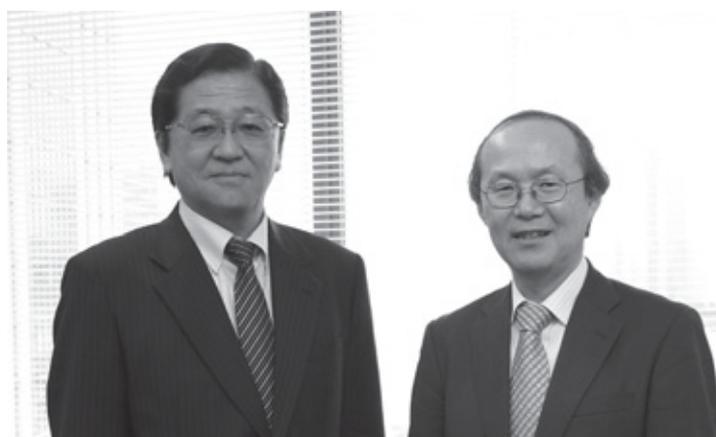
松本：ビッグデータも、データの活用方法を見出すのは発想力ですね。インスピレーションのスキルをどう磨いていくのかは、難しい問題です。

加藤：インスピレーションが湧くためには現場の経験が大事だと思っています。座学で知識を得たり、実際に物作りすることももちろん大事ですが、やはりインスピレーションの源泉は現場、つまりお客様の業界に身を置くのが近道だと思います。何もないところから急に浮かぶわけではなく、様々な過去の経験や実績がシンクロして湧いてくるというケースが多いのではないのでしょうか。

松本：先ほど話されていたように、開発者が運用フェーズに参画したり、お客様の現場へ出向するなどのローテーションをすることで、インスピレーションが浮かぶ資質ができてくるのでしょうか。最後に、SECへの要望や期待などがあればお話しください。

加藤：日本発のものをどんどん出してもらいたいですね。品質管理手法である「シックス・シグマ」ではスペシャリストに「ブラックベルト」という資格を与えるそうです。日本の黒帯が語源となっています。品質に関して日本は過剰なくらいの進んだ考えがあります。それ以外にも、日本のシビアなお客様に対してのITサービスをさらにブラッシュアップして世界に発信することを期待しています。

松本：ぜひ出せるようにしたいですね。日本におけるソフトウェアの品質は、欧米に比べると一桁良いという調査報告もあるくらいです。一方で日本人は、上手にモデル化するところが苦手です。個々のスキルはレベルが高く、高い品質のものを作り出しているのですから、そのやり方をどう普遍化、モデル化するかが課題となりますね。本日はありがとうございました。



SEC2012年度活動概要

2012年度は、IPA第2期中期計画の最終年度となっており、中期計画で掲げた事業の完遂と次期（第3期）中期計画への展開を見据えた活動を進めた。本稿では、2012年度の主要な成果概要を紹介し、本稿以降で詳しい事業内容を紹介する。

また、トピックスとして第2期中期計画の総括と第3期中期計画の概要等を紹介する。

1 情報システムの高信頼化対策

(1) 高信頼ソフトウェアの安全性・信頼性を第三者により検証する制度設計を完了

昨今、ソフトウェアの機能安全に関する国際的な要求の高まりなど、第三者検証によるソフトウェアの信頼性の見える化促進の重要性が高まっている。SECでは、製品やサービスの品質を、独立した第三者の立場で検証・評価し、一般利用者にも理解できる形で提供するための制度設計を終え、様々な分野に展開するための「ソフトウェア品質説明のための制度ガイドライン」を取りまとめた。また、ガイドライン適用の先行分野として、CSAJ^{*1}と共同で実運用に向けた制度の構築、審査基準の策定、評価を行うガイドラインに基づく実証実験を実施した。ここで、制度運用に必要な規程文書等を取りまとめ、規程文書に含まれる制度に対する要求事項をガイドラインに反映した。

(2) 組込みソフトウェア向け開発技術リファレンス (ESxR^{*2}) について、開発プロセス全般を網羅

2012年度は、「組込みソフトウェア向け設計ガイドESDR[事例編]」、「組込みソフトウェア開発における品質向上の勧め[テスト編～事例集～]」、「組込みソフトウェア開発における品質向上の勧め[バグ管理手法編]」の書籍を発行した。これにより、ESxRシリーズは体系的に整理され、組込みソフトウェアの開発プロセス全般が網羅された。

(3) 社会基盤のITインフラの信頼性向上に向けた障害情報の収集・分析への取り組み

重要インフラ10分野及び将来の産業分野を対象に、2000年以降に発生した海外の障害事例を調査し、障害情報収集・公開・調査方法、法律・規定等の整備状況等を整理し、報告書を公開した。また、東日本大震災における被災事例を含めた事例を調査し、報告書を公開した。さらに、ITサービス継続の実践に役立つ、「高回復力システム基盤導入ガイド(概要編)」、「高回復力システム基盤導入ガイド(計画編)」、

「高回復力システム基盤導入ガイド(事例編)」を作成し、公開した。

(4) ソフトウェア開発プロジェクトの実態を分析

民間では収集が困難な企業の開発現場から生のプロジェクトデータを収集し、新規分析結果を加え、「ソフトウェア開発データ白書2012-2013」に取りまとめて発行した。

(5) プロセス改善手法ツール及び研修教材を体系的に整備

中小企業におけるソフトウェア(製品)品質の安定・向上を達成し、コスト削減、納期短縮などを効果的に実現するため、トップダウン型のプロセス改善手法である「SPEAK-IPA^{*3}」とボトムアップ型の「SPINA³CH自律改善メソッド^{*4}」を利用者の視点で整理、体系化し、開発現場における実践的な活用につなげるための「プロセス改善活用ガイド」として公開した。また、2012年度までのプロセス改善活動の成果に基づき幅広い活用を目的とした研修教材(6コース分)を整備し、公開した。

2 地域・中小企業を支援するための活動

(1) 「ESxR」を解説できる人材の育成

ESxRシリーズの教育ができる講師の育成の取組みとして、2012年度はESPR^{*5}、ESMR^{*6}及びESMG^{*7}のトレー

【脚注】

- ※1 CSAJ (Computer Software Association of Japan) : (一社) コンピュータソフトウェア協会。
- ※2 ESxR (Embedded System development exemplar Reference) : 組込みシステム開発技術リファレンスシリーズの総称。
- ※3 SPEAK-IPA : ソフトウェアプロセスの供給者能力判定及びアセスメントキット—IPA版。
http://sec.ipa.go.jp/reports/20130326_2.html
- ※4 SPINA³CH 自律改善メソッド : Software Process Improvement with Navigation, Awareness, Analysis and Autonomy for Challenge 自律改善メソッド。
http://sec.ipa.go.jp/reports/20130326_3.html
- ※5 ESPR : Embedded System development Process Reference
- ※6 ESMR : Embedded System development Management Reference
- ※7 ESMG : Embedded System development Management planning training Guide

ナー養成コースを開発し、指導要領や教育コンテンツ教材として整備し、公開した。自律的な ESCR^{※8} 普及活動の一貫として、既にコース修了者が広島市立大学にて講習会を開催したり、一般向けに JASA^{※9} 会員の教育会社（株）サートプロが ESMR セミナーを開催するなど、SEC 成果が幅広く着実に普及する動きも出てきている。

(2) 「定量的プロジェクト管理ツール」の公開・普及

ソフトウェア開発プロジェクトの定量的管理手法の地域・中小企業への普及を図るために「定量的プロジェクト管理ツール」をオープンソースとして公開した。

モバイル通信会社、メーカー、システムインテグレータなどの5社が導入に至り、15社では試用及び導入の検討が行われている。また、普及・啓発活動を民間の PPMA^{※10} に移行した。さらに、業界団体 ITA^{※11} が同ツールの評価と使用法を盛り込んだガイドを作成しているなど、今後民間主体での普及・導入が期待される。

(3) IPA/SEC 成果を簡単に検索できるデータベース SWE iPedia の提供開始

利用者が SEC 成果の中から必要な資料を容易に探しだせるよう利便性の向上を図るため、ソフトウェア・エンジニアリング情報データベースとして「SWE iPedia」を構築し公開した。SWE iPedia には、2004年10月発足以後の SEC 成果が体系化して保存されている。その過去の SEC 成果物の中から、利用者は多様な検索機能を利用して目的の物を探し出すことができるようになった。

これにより、2012年度の成果物のダウンロード件数は897,089件、前年比135%と3割強の増加となった。

3 SEC 成果の国際的情報発信、国際連携

(1) IESE との共同研究成果である GQM+Strategies[®]※12 の普及展開

独国フラウンホーファ協会実験的ソフトウェア工学研究所 (IESE) との共同研究の成果で、組織ゴール実現のための方法論である GQM+Strategies[®] のワークショップを開催した。また、普及・啓発活動は、ITCA^{※13} に設置された「GQM+Strategies[®] の企業・団体への適用研究会」及び早稲田大学に設置された「GQM+Strategies[®] の教育・普及に向けた研究会（仮称）」に移行した。さらに伊藤忠テクノソリューションズ（株）、（株）クニエ等では社内に導入してコンサルティング事業への展開も計画しているなど、今後、民間を主体とした更なる普及展開が期待される。

(2) 英訳版成果物の活用推進による国内企業のグローバル化対応の支援実施

SEC 成果を海外拠点等でも一層活用してもらうために、2012年度は ESPR、ESCR C++ 言語版を英訳し、公開した。既に英訳済の ESQR、ESCR を含め、2010年8月の公開から英訳版ガイドを22社に提供した。国内企業が海外進出先（先進国、発展途上国含め12ヶ国・地域）で活用するなどの動きにつながっている。また、欧州 MISRA^{※14} の C 言語コーディング規約が改訂されることになり、SEC は事前評価の依頼を受けたため、ESCR の知見を活用して MISRA-C の新旧版の評価を行い、評価レポートを提出した。2013年3月18日に発行された MISRA-C2012 には、ESCR の記述内容の一部が参照されている。

(3) 海外政府関係機関との連携を強化

2010年度から SEC が連携している主要な海外政府機関である米国商務省国立標準技術研究所 (NIST) 及び仏国原子力・代替エネルギー庁 (CEA) システム統合技術研究所 (LIST) との関係強化した。NIST とは、第3回定期協議をワシントンで開催し、ソフトウェアの高信頼性に向けた最先端の技術情報やソフトウェア・エンジニアリング推進の社会的価値等について意見交換を行い、今後の活動のための有用な情報を得た。LIST からは、2012年5月に研究員を招聘し、ソフトウェア品質説明力強化の考え方等に関する意見交換を行った。また、組込みシステム開発技術展 (ESEC) では、LIST 研究員から欧州のソフトウェアの信頼性の取組み状況などを紹介してもらった。さらに、2012年10月には LIST を訪問し、モデリング技術など LIST の先進的なソフトウェア信頼性向上技術に関する最近の取組み内容について議論した。

次項からは、2012年度の事業内容について詳しく紹介する。なお、2012年度の SEC の活動にご協力をいただいた部会委員を始め関係各位にこの場を借りて心よりの感謝と共に、今後の SEC の活動への変わらぬご支援をお願い申し上げます。

【脚注】

- ※8 ESCR : Embedded System development Coding Reference
- ※9 JASA (Japan Embedded Systems Technology Association) : (一社) 組込みシステム技術協会。
- ※10 PPMA (Practical Project Management Association) : (一社) 実践的プロジェクトマネジメント推進協会。
- ※11 ITA (Information Technology Alliance) : 独立系情報サービス会社の各社が相互の事業活性化、競争力アップを図ることを目的に1995年発足した任意の団体。
- ※12 GQM+Strategies[®] (Goal Question Metric + Strategies) : 組織のゴールと結び付けた IT 戦略の実施において、前提とする事実及び過程への考察からゴール成就への影響とリスク評価を行う方法論。
- ※13 ITCA (IT Coordinators Association) : NPO 法人 IT コーディネータ協会。
- ※14 MISRA (Motor Industry Software Reliability Association) : 欧州の自動車業界団体。MISRA が開発した C 言語のためのソフトウェア設計標準規格が MISRA-C。

統合システムの安全・安心に向けた 2012年度の取組み

SEC 統合系プロジェクト リーダー
中村 雄三

SEC 統合系プロジェクト 研究員
三毛 功子

SEC 統合系プロジェクト 研究員
細目 紀子

1 はじめに

現代の社会では、重要インフラをはじめとして、身の回りの製品・システムが高機能・複雑化しており、それを構成するIT技術の重要性が一層高まっている。さらに、このような情報システムや組込み製品・システム等が相互接続されることによる新たなシステム（System of Systems、統合システム）が構成されつつある。経済産業省で推進するIT融合6分野^{*1}は、まさに統合システムであり、例えばスマートコミュニティにおけるスマートハウスでは、従来単独で動作してきた家電、自動車等も相互接続され家庭のエネルギー効率化のため連携して動作するようになる。また、交通システムとしての自動車や都市交通等の連携も今後、進展すると考えられている。

従来、日本の製品・サービスは、その重要なコンポーネントであるソフトウェアも含め、高い品質である、と言われてきた。それぞれの製品・システムに対して、設計・製造工程での品質を意識した開発プロセスの確立や、利用者の利用形態までを含め、様々な観点での品質作りこみが行われてきており、その努力によって安全・安心が確保されてきた。統合システム化等により、想定とは異なった新たな利用形態、異なる動作環境、さらには新たな製品・システムとの接続が行われることで、想定外の新たな障害が生じる可能性がある。同時に、システム間の相互接続により、いったん障害が発生してしまった場合の影響範囲の拡大を防止するなど、利用者の安全・安心に対応する信頼性の確保が重要な課題になっている。

発足後3年となる統合系プロジェクトでは、安全・安心なIT社会の実現を目指し、統合システムに特有、或いは統合システムを構成する情報システムと組込み製品・システムに共通する課題の解決に向けた取組みを進めてきた。ここでは、2012年度の統合系プロジェクトの活動を概説する。

2 2012年度の統合系プロジェクトの活動

2012年度に統合系プロジェクトは、大きく「品質説明力

強化」と「上流品質技術強化」の2つの柱となるテーマに取り組んだ。さらにそれらと関連した「組織的取組み強化」として、信頼性自己診断ツールの提供と海外での障害状況調査等を推進した。ここではその概略を示す。

(1) 1つめの柱：品質説明力強化

「品質説明力強化」では、製品・サービスの品質に関して第三者が検証し、利用者に分かりやすく提示する公正な制度を、それぞれの障害時の影響度やニーズ等に合わせて分野毎に構築できるようにするため、「ソフトウェア品質説明のための制度ガイドライン」を策定した。供給者とは別の第三者が、ある分野の専門家の立場で利用者側の観点から製品・システムを検証することは、供給者のみが品質の確かさを主張することに比べ、利用者の安心のためには大きな前進になると考えられる。また供給者にとっても、品質に関する自らの主張を第三者の検証により確認できること、また場合によっては検証に向けて供給者側で開発プロセスやドキュメント整備等を見直す事によるソフトウェア品質向上の効果も考えられる。詳しくは、この後の記事を参照していただきたい。

(2) 2つめの柱：上流品質技術強化

品質説明力強化の技術的な基盤整備の一環として、「上流品質技術強化」に取り組んだ。上流工程での品質向上を目指して、数学的基盤を持つ形式手法の普及に取り組み、「曖昧性や矛盾のない厳密な仕様を記述すること」、同時に「そのような事のできる人材育成」、のための教材を作成・公開した。また、コンシューマデバイスのディペンダビリティ保証に向けた取組みを進めた。さらに、多様な要求を満足すると同時に、その要求が変化するような統合システムを高信頼で効率的に開発するための手法であるMBSE（Model Based Systems Engineering）の啓発のため、プロジェクト

【脚注】

*1 IT融合6分野 <http://www.meti.go.jp/press/2011/08/20110811002/20110811002.html>

表1 諸外国における政府機関の業種別 IT 障害等に対する対応状況

<諸外国における政府機関の業種別 IT 障害等に対する対応状況について報告書から抜粋>

| 区分 | 業種分野 | 障害情報収集組織 | 障害に関する公開情報 | 公開情報例 | 関連する準拠法、規定 |
|------------|----------------|---|--|---|--|
| 重要 インフラ | 情報通信 ・放送 | シンガポール 情報通信開発庁 (IDA) | シンガポール (IDA) ・通信障害情報や品質 (QoS) 情報を公開 | シンガポール (IDA) ・携帯電話の通信障害に 対し、サービスレジリ エンス法に基づき、事 業者に罰金 (2012 年) | シンガポール (IDA) ・サービスレジリエン シ法、通信法 |
| | 金融 | 米 証券取引委員会 (SEC) 連邦準備制度理事会 (FRB) NY 証券取引所 (NYSE) 金融取引業規制機構 (FINRA) | 米 (SEC) (FRB) ・障害情報、復旧に関す る情報、調査状況、原 因等を公開 ・証券監督者国際機構 (IO SCO)、バーゼル銀行監 督委員会 (BCBS) へ報 告 | 米 (NYSE) (SEC) ・証券自動売買障害に 対し、拡大防止策、リ スク管理規定違反調査 (2012 年) | 米 (NYSE) (SEC) ・米国証券取引法 ・market-access rule (Rule 15c3-5- Risk Management Controls for Brokers or Dealers with Market Access) ・NYSE Euronext 証券シ ステムの障害に関連す る補償 - 規則 18 |
| | 業種横断 (製品安全) | 欧州 EU 緊急警告システム (RAPEX) EU 市場監視制度 (ICSMS) | EU (RAPEX) (ICSMS) ・EU 加盟国と中国の消費 製品の障害情報を公開 ※ Software 起因関連は 自動車系が多い | EU (RAPEX) ・ Software に起因する障 害 18 件報告 (2005 年～) | EU ・ EU 一般製品安全指令 (GPSD) |

の手法選択や予算決定権のある方を対象とした手引きを作成した。詳しくは、このあとの記事を参照していただきたい。

(3) 海外障害事例調査

システムの信頼性向上のためには、様々な障害事例やその対策等の情報を収集し、共有する事が有効と考えられる。そこで SEC は、海外の障害事例に関する状況把握のため、「海外における IT 障害の影響及び対応策に関する事例調査」を実施した。重要インフラ 10 分野（情報通信・放送、金融、航空、鉄道、電力、ガス、政府・行政サービス、医療、水道、物流）における、2000 年以降に発生した海外障害事例約 300 件に関して、障害の概要、原因、事業者の対策、政府機関等における対応策（障害情報収集・公開・調査方法、法律・規定等の整備状況）等について整理し、調査報告書と障害事例集にまとめ、2013 年 4 月に公開した^{※2}。その一例として、「情報通信・放送」、「金融」業種の政府機関等の障害対応状況について、シンガポール、米国の例を抜粋して紹介する。また業種横断の「製品安全」にかかわる対応例として欧州の EU 緊急警告システム (RAPEX) をとりあげた (表 1)。調査結果の詳細は報告書と事例集を参照していただきたい。

(4) 信頼性自己診断ツール

情報システムの信頼性確保のためには、システム自体の信頼性はもとより、そのシステムの開発・運用に関わるユーザ・ベンダの組織的な取組みが重要である。経済産業省が 2009 年 3 月 24 日に発行した「情報システムの信頼性向

上に関する評価指標 (第 1 版)」に準拠して、SEC ではユーザ・ベンダの信頼性・安全性の担保となる作業度合を診断する「信頼性自己診断ツール」を提供してきた。本年度は、中核となる診断機能に特化して選択肢を改善し、診断精度の向上を計り、改変可・営利目的で利用可能な形態^{※3}の「スタンドアロン型信頼性自己診断ツール」として 3 月に公開した^{※4}。本ツールを前述の評価指標と共に活用することにより、情報システムのユーザ・ベンダそれぞれが実施すべき事項、協力すべき事項、合意すべき事項等を明確化して、信頼性向上に取り組むことができる。ぜひ、ご活用いただきたい。

3 おわりに

ここでは 2012 年度の統合系プロジェクトの主な活動に関して概説した。2012 年度をもって IPA の第二期中期計画は終了したが、第三期中期計画でも、国民生活の安全・安心のため、SEC として複雑・多様化する情報処理システムの信頼性の見える化に対する取組みを推進していく。

【脚注】

- ※ 2 「海外における IT 障害の影響及び対応策に関する事例調査」
<http://sec.ipa.go.jp/reports/20130430.html>
- ※ 3 クリエイティブコモンズ表示 - 継承 2.1 日本ライセンス
<http://creativecommons.org/licenses/by-sa/2.1/jp>
- ※ 4 「スタンドアロン型信頼性自己診断ツール」
<http://sec.ipa.go.jp/tool/dependability.html>

ソフトウェア品質説明力強化の取組み

SEC 統合系プロジェクト 研究員 SEC 統合系プロジェクト 研究員 SEC 統合系プロジェクト 研究員

田中 和夫 伊藤 克己 宮崎 義昭

SEC 統合系プロジェクト サブリーダー

田丸 喜一郎

1 背景

今日、ソフトウェアが組み込まれた機器や、ソフトウェアで実現するサービスは日常生活に無くてはならない社会基盤となっている。IT（Information Technology、情報技術）におけるソフトウェアの重要性が増す中で、ソフトウェアの不具合に起因する機器の故障やサービスの停止は社会に大きな影響を与える。

利用者が安心して製品、サービスやシステムを使えるようにするためには、製品・システムの供給者は、高度化・複雑化する製品・システムの品質確保に努めると同時に、信頼性をはじめとする品質について、利用者に対する十分な説明責任を果たす必要がある。

2 ソフトウェア品質説明力の強化

ここで、ソフトウェア品質説明力の考え方を示す。利用者が品質を確認し判断できるように供給者が利用者に対して説明することを「ソフトウェア品質説明」、その説明が持つ力を「ソフトウェア品質説明力」と呼ぶことにする。ソフトウェア品質説明とは、製品・システムについて以下の事項を根拠や事実に基づいて説明することである。

- 1) 製品・システムが想定する利用者、利用目的、利用状況、制約事項
- 2) 製品・システムを利用者が利用する上で必要なソフトウェアの品質とその目標
- 3) 品質目標を達成するための設計・実装・運用及び保守
- 4) 品質目標を達成したことの検証・監査

また、説明が持つ力の強さは、説明が論理的であるか、説明に客観性があるか、他の事業者や製品・サービスと比較可能であるか、など様々な要素が関係する。

したがって、ソフトウェア品質説明力を強化するには、供給者がソフトウェアを開発・運用する過程において、要求される品質を確保する技術的な側面、製品・システムのライフサイクル全般にわたる管理的な側面、それらを評価する制度的な側面があると考えられる。

そして、制度的な側面の1つとして、第三者が品質説明

の適切性を確認する仕組みを構築するアプローチがある。具体的には、第三者が供給者の品質説明の適切性を客観的かつ専門的な立場から利用者の代わりに確認し、結果を利用者に理解できる形で提供する仕組みの導入である。

図1に示すように、供給者が製品や設計書等の書類等を第三者に提出し、第三者が基準に照らして供給者の品質の説明が適切であることを確認（審査・判定）する。そして、その判定結果（適否など）を第三者が利用者に提供し、供給者は判定結果を製品に付与して利用者に提供できる。

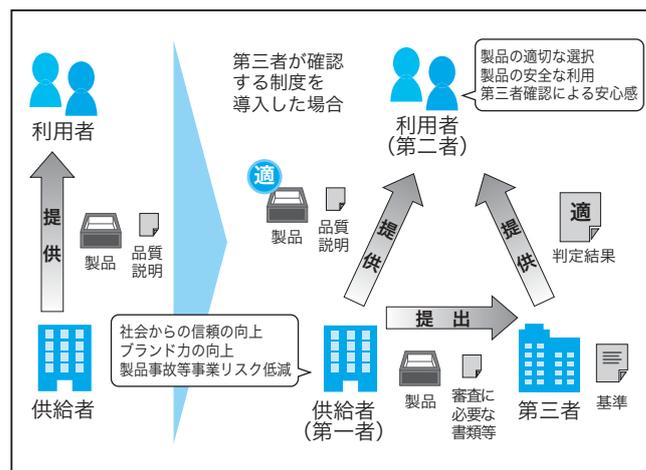


図1 第三者が品質を確認する仕組みの導入

この仕組みにより、利用者には以下のメリットがある。

- ・製品・システムを適切に選択できる
- ・製品・システムの品質を知ることにより、製品・システムが想定している利用方法を理解し、製品・システムを正しく安全に使うことができる
- ・供給者とは独立な第三者が製品・システムを確認していることにより、安心して製品・システムを使うことができる

また、供給者には以下のメリットがある。

- ・製品・システムに対する社会からの信頼が向上する
- ・供給者のブランド力が向上する
- ・製品・システムを利用者が正しく使うことにより、製品事故等のリスクが低減する

- ・国際標準にさきがけて新しい分野での客観性のある品質説明ができる

このような制度は、製品等によって求められる技術が異なることや、対象製品が単体で利用する製品なのか、重要インフラを構成する製品なのかなど社会的な位置付け等が異なるため、分野ごとに構築することになる。しかし、分野ごとに制度を構築すると、制度間の整合性が無いことにより、制度自体の信頼性の低下や利用者のわかりにくさにつながるという課題がある。

そこで、様々な分野に制度を展開するため、制度の要求事項を整理した「製品・システムにおけるソフトウェアの信頼性・安全性等に関する品質説明力強化のための制度構築ガイドライン（通称：「ソフトウェア品質説明のための制度ガイドライン」）（以下、「制度ガイドライン」という）をとりまとめ公開した^{※1}。

3 ソフトウェア品質説明のための制度ガイドライン

制度ガイドラインで示した制度の基本的な考え方を以下に示す。

ソフトウェアが重要な機能の実現に関わる製品・システムにおけるソフトウェアの信頼性や安全性等に関する品質について、供給者が利用者に示す説明が適切であることを、第三者が基準に照らして確認し、第三者並びに供給者がその結果を利用者に分かりやすく提供する制度

制度ガイドラインの構成を以下に示す。

はじめに

1. 用語
2. 制度の基本的な考え方
3. 個別制度の構築
 - 3.1. 制度責任主体と制度構成要素
 - 3.2. 制度責任主体の役割
 - 3.3. 制度責任主体の実施項目
 - 3.4. 制度構成要素の独立性
 - 3.5. 審査基準の策定と審査の実施
4. 個別制度に対する要求事項
 - 4.1. 基本的な要求事項
 - 4.2. 制度責任主体への要求事項
 - 4.3. 制度規定に記述する項目
5. 本ガイドラインへの準拠表示

この制度ガイドラインに基づき、制度設計をすることによって、分野が異なっても同じ考え方を採用した制度の構築が期待できる。

4 実証実験の実施

また、制度ガイドラインの最初の適用分野として、一般

社団法人コンピュータソフトウェア協会（CSAJ^{※2}）が企画していた「パッケージソフトウェア品質認証制度（PSQ 認証制度^{※3}）」について、実証実験 PT（プロジェクトチーム）を設置し、CSAJ と共同で以下の事項に関する実証実験を実施した。

- ・制度ガイドラインへの PSQ 認証制度の適合性の確認
- ・PSQ 認証制度の審査基準の策定
- ・実際に流通している製品を用いた模擬認証（審査と判定）
- ・PSQ 認証制度および PSQ 認証制度の審査基準の検証

その結果、制度ガイドラインの内容の精査ができ、CSAJ からは「実証実験を通じて、各方面の専門家から様々な観点での指摘を受けたことで制度準備の網羅性が広がり、実際の運用面での安心につながった」という評価を得た。

5 今後の取組み

今後は、制度ガイドラインの認知度向上のために各種業界団体等への説明を行い、ニーズのある分野に対しては制度化に向けての支援を実施するなど、利用者が安心して製品・システムを利用できるための制度作りに向けた活動を推進する。図 2 に制度ガイドラインの展開イメージを示す。

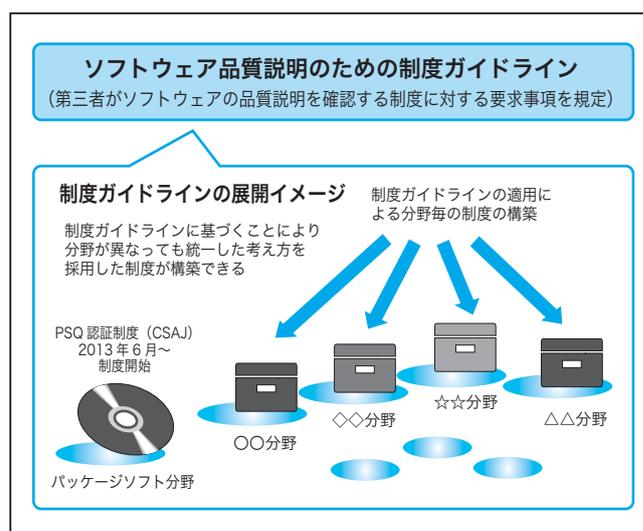


図2 制度ガイドラインの展開イメージ

【脚注】

- ※1 「製品・システムにおけるソフトウェアの信頼性・安全性等に関する品質説明力強化のための制度構築ガイドライン」
<http://www.ipa.go.jp/sec/reports/20130612.html>
- ※2 CSAJ：Computer Software Association of Japan
- ※3 PSQ 認証制度：パッケージソフトウェア製品を対象とした制度であり、JIS X 25051:2011「ソフトウェア製品の品質要求および評価—商用既製（COTS）ソフトウェア製品に対する品質要求事項及び試験に対する指示」（ISO/IEC 25051:2006）の要求事項に基づいた審査基準に照らして、パッケージソフトウェア製品の品質の妥当性に関する認証を行うものである

上流品質技術強化 形式手法普及活動の取組み

SEC 調査役

新谷 勝利

SEC 統合系プロジェクト 研究員

室 修治

SEC 統合系プロジェクト 研究員

神谷 慎吾

SEC 統合系プロジェクト 研究員

藤原 由起子

1 活動概要

IPA/SEC では、システムの信頼性向上のために、上流工程での品質向上への取組みが重要と考え、上流工程における高品質化に向けた技術として、形式手法の普及促進を行ってきた。

形式手法は数学的な厳密さを持つため、形式手法を用いた仕様記述では、ソフトウェアの品質低下や開発コスト増加につながる仕様書の曖昧性や不整合を取り除くなどの効果が期待できる。一方で、形式手法の認知度は高くなく、導入事例や適用効果などについてもあまり知られていなかった。

IPA/SEC では、形式手法についての「難しい」といった先入観の払拭や適用効果の理解を目的とした入門教材を作成し、管理者・リーダー向け、技術者向けセミナーを実施（計7回、310名参加）した。セミナーアンケートでは、以下の様な、形式手法の開発現場導入に向けた前向きな意見や要望などがみられた。

- ・導入に向けて検討することが可能になり、非常に参考になった。
- ・このようなセミナーが草の根的に続けられると、日本でもきっと形式手法が普及すると思う。
- ・形式手法を適用するためには、さらに実習型のセミナーが必要。

セミナーでの意見を反映した「入門教材第二版」、実践法の副読本「対象を如何にモデル化するか?」、さらに、国内外の事例調査結果「厳密な仕様記述における形式手法成功事例調査報告書」^{*1}を元に作成した副読本「厳密な仕様記述入門」を、「実務家のための形式手法シリーズ」として同時公開した^{*2}。

また、形式手法による情報系実稼働システムの基本設計書検証効果を測定した実験結果の公開^{*3}、実験でも活用した「形式手法活用ガイドならびに参考資料」の公開を行った^{*4}。

さらに、形式手法の啓発活動の一環として、形式手法活用調査を国際的に実施しているデンマーク・オーフス大学のピーター・ラーセン教授を招聘し、SEC 特別セミナーを実施した^{*5}。

2 形式手法教材について

IPA/SEC が公開した形式手法教材「実務家のための形式手法シリーズ」は、入門教材セットと副読本2冊からなる。

入門教材には、現場導入で得られた種々の実践的知見を盛り込んだ7つのモジュール（表1 入門教材①～⑦）があり、これを組み合わせることで、管理者向けやエンジニア向けなど目的別の教育コースを構成できるようになっている。

表1 「実務家のための形式手法シリーズ」
教育コース毎の教材の組合せ例

| シリーズ | 表題 | 管理者向け 半日コース | エンジニア向け 一日コース (概説) | エンジニア向け 一日コース (モデル化) | エンジニア向け 二日コース | 自主学习 |
|---------------------|--------------------|----------------|--------------------------|----------------------------|------------------|------|
| 実務家のための形式手法 入門教材 | 「シラバス」 | ○ | ○ | ○ | ○ | |
| | ①「なぜ形式手法か」 | ○ | ○ | | ○ | |
| | ②「形式手法導入に関わるガイダンス」 | ○ | ○ | | ○ | |
| | ③「事例：成功事例」 | | ○ | | ○ | |
| | ④「事例：種々の事例」 | | ○ | | ○ | |
| | ⑤「事例：実証実験」 | | ○ | | ○ | |
| | ⑥「実践法：モデル化の手順と事例」 | | | ○ | ○ | |
| | ⑦「実践法：モデル化の課題例」 | | | ○ | ○ | |
| | 参考資料 | ○ | ○ | ○ | ○ | |
| | 副読本 | | | | | ○ |
| | 対象を如何にモデル化するか? | | | | | ○ |
| | 厳密な仕様記述入門 | | | | | ○ |

「対象を如何にモデル化するか?」は、形式手法を活用するための具体的手順と課題を説明した入門教材「実践法：モデル化の手順と事例」と「実践法：モデル化の課題例」の副読本である。

「厳密な仕様記述入門」は、上流工程における仕様の厳密な定義及び記述に関する副読本である。形式手法をベースとして書かれているが、日本語の文章による仕様書記述においても役立つ内容となっている。

これら副読本は、自主学习にも活用できるので、ぜひご利用いただきたい。

3 おわりに

2013年度は、2012年度末に公開した教材と副読本を活用したセミナーの開催を計画している。過去のセミナーの受講者からのご意見も踏まえ、より効果的なセミナーを行う予定である。

【脚注】

- *1 「厳密な仕様記述における形式手法成功事例調査報告書」
<http://sec.ipa.go.jp/reports/20130125.html>
- *2 実務家のための形式手法シリーズ、教材及び副読本の公開
<http://sec.ipa.go.jp/reports/20130328.html>
- *3 「情報系の実稼働システムを対象とした形式手法適用実験報告書」
<http://sec.ipa.go.jp/reports/20120420.html>
- *4 「形式手法活用ガイドならびに参考資料」
<http://sec.ipa.go.jp/reports/20120928.html>
- *5 「形式手法の98導入事例の調査・分析から見る高信頼性ソフトウェア開発の現状」
<http://sec.ipa.go.jp/seminar/2012/20121023.html>

上流品質技術強化 コンシューマデバイス機能安全規格化の取組み

SEC 統合系プロジェクト 研究員 SEC 統合系プロジェクト 研究員

内田 功志 室 修治

1 はじめに

機能安全の規格は、専門家が使用することを前提としたシステムを対象とする規格が先行して制定されてきた。

一般消費者が使用する製品については、自動車の機能安全規格 ISO 26262 がようやく発行されたところである。

そこで、一般消費者向けのシステムを対象とした製品群を“コンシューマデバイス（消費者機械）”と呼び、分野横断的なコンシューマデバイスにおいて、多様な利用者・利用環境に対応するために必要とされる安全性に可用性や信頼性、保全性を確保するための規格を、自動車の機能安全規格 ISO 26262 をベースとして提案することとした。

2 機能安全規格の概要

コンシューマデバイスは表 1 に示すように、産業機械とは異なり、技術者の手を離れて多様な環境で多くの一般消費者に利用される。そこで、産業機械とは考慮すべき条件が異なる場合も多く、また達成すべき信頼性や安全性にも特別の考慮が必要と考えられる。しかしながら安全性に関する標準化の取組みは前述のように産業機械や工業プラントに対するものが先行しており、コンシューマデバイスの安全性に対する標準化の取組みは遅れている状況である。

表 1 産業機械とコンシューマデバイスの違い

| | 産業機械 | コンシューマデバイス |
|--------|---------------|----------------|
| 生産数 | 少～多 | 多～ |
| 利用者 | 専門家 | 一般消費者 |
| 要求コスト | (高) | 低 |
| メンテナンス | 設置現場 | ユーザ、サービスステーション |
| 環境 | 工場など (限定的) | ユーザ環境 (多様) |

異種のシステムが組み合わされたシステムにおいて個々のシステムについて世代を越えて管理しなければならないことは、今日の社会システムや製品開発に共通の課題である。コンシューマデバイスはそのような複雑なシステムを構成する要素でもあり、些細な不具合が波及して、重大な社会問題を引き起こす可能性を排除できない。すなわち、コンシューマデバイスの安全性、信頼性、セキュリティを含めたディペンダビリティ^{※1}を保証する枠組みを構築する

ことが、今日の重要な課題となっている。

コンシューマデバイスの標準化を検討するに当たり、まずベースとなる ISO 26262 を分析した。ISO 26262 の分析では UML におけるクラス図を利用してメタモデルという形で表現し、検討に当たるメンバー間で共通の理解を得られるようにした。また ISO 26262 における安全についての概念を整理するため、Part1 から Part3 までを分析の対象とした。

ISO 26262 の Part3 は主にハザード分析に関する考え方やセーフティケースの記述に関する考え方、セーフタイムカニズムの必要性、変更に関するインパクト分析、さらには Proven in Use(使用実績による証明)に関する取扱いなど、内容が盛り沢山である。特にセーフティケースを使用して、安全であることを明確にすることは、ISO 26262 では必須要件項目になっている。

コンシューマデバイスに関する現在までの検討結果では、対象モデルの記述においてはセーフティケースからディペンダビリティケース^{※2}へと拡張することを考えている。これはコンシューマデバイスやそれらが複合したシステムの特性を考慮し、単に安全であるだけでなく可用性や信頼性、保全性などのコンシューマデバイスに求められる要素についても配慮すべきとの考えからである。

ディペンダビリティを確保するためには、想定と試行の素早い繰り返しによる改善が有効であり、そのような開発プロセスも規格に盛り込むこととした。

3 おわりに

上記の検討内容も踏まえて、OMG^{※3}に提出した提案要請 (RFP) は、2013 年 3 月に正式発行された。今後は、この RFP に対する具体的な規格化への提案を進めていく予定である。

【脚注】

- ※1 ディペンダビリティ：信頼性性能、保全性性能及び保全支援能力を記述するために用いられる包括的な用語とされている
- ※2 ディペンダビリティケース：ディペンダビリティを保証するために階層的、系統的に構成する論拠の示し方
- ※3 OMG：Object Management Group

上流品質技術強化 MBSE 導入の手引き

SEC 統合系プロジェクト 研究員 内田 功志
SEC 統合系プロジェクト 研究員 室 修治

1 はじめに

SEC では過去 3 カ年にわたり、複雑なシステム開発を高い信頼性をもって実現できる技術の一つとして「モデルベース開発」に注目し、課題の整理から開発現場へどのように導入していくかについて有識者を交え検討を行ってきた。2012 年度はこれまでの検討結果から、複雑なシステムの開発方法論であるシステムズエンジニアリングにモデルベース開発を適応させる技術である MBSE (モデルベースシステムズエンジニアリング) が有効と考え、その普及促進のため、「MBSE 導入の手引き」を作成した。

2 手引きの概要

近年、スマートコミュニティやスマートハウスなどの異なる複数のシステムが互いに複雑な関係を持つ System of Systems (以降は SoS と略す) が拡大し、日常生活に無くてはならない重要インフラになりつつある。国民生活の安全・安心、及び利便性向上のためには、このような SoS を開発するための方法論 (システムズエンジニアリング) が必要とされている。

このようなシステムでは、

- ・法改正や利用者要望ビジネス環境の変化などにより環境条件や要求条件が変化する
- ・例えば、省エネなど一つの機能がメカ、エレキ、ソフトにまたがって実現されるといった複雑な関係性を持つ
- ・開発過程においては、IT 技術だけでなく電力、環境、法律など複数の分野にまたがる関係者が開発に携わる
- ・個々の製品・システムの開発・運用に携わる国内外の複数の会社関係する

といった場合も少なくない。このような重要インフラシステムに対しては、いかに効率よく、要求される品質・コスト・期間に見合った開発を行うかが必要となる。

特に、開発の段階で、多岐にわたる分野の関係者間でさまざまなやりとりを円滑に行うためには、関係者間でお互いに何を考えているのかを共有することが重要となる。人は頭の中でものごとを考える際には、図的 (ビジュアル) な何かを頭の中に思い浮かべて考えている筈である。その図あるいは絵をいかに外に出して、開発関係者間で共有するかが重要なポイントとなる。

複数の分野にまたがるチームが、お互いにそのプロセスを理解しながらシステム開発を推進するためには、従

来の文書に基づくアプローチに代わり、図的表記法である SysML などを用いた MBSE が有効である。

以下の特徴を持つ MBSE の導入により、「製品やサービスなどのシステムの開発を成功に導くこと」が期待できる。

MBSE の特徴

- ・複数の分野にまたがる関係者が、互いに図を見ながらコミュニケーションをとることができる
- ・要求のトレーサビリティが確保され、要求の変更へ適切な対応を行うことができる
- ・開発の初期の段階で要求を図的に明確化でき、開発途中の意図しない手戻りを抑えることができる
- ・図的に表現されたシステムモデルを再利用でき、製品やサービスのシリーズ化、ファミリー化が容易になる

しかし、日本では、SoS に対する方法論は確立されておらず、これに対応できるシステム構築の方法論が必要となっている。欧米では MBSE の評価は高いが、日本での評価はいまだに低い。

導入が進まない要因

- ・従来の日本の開発スタイルは、要件定義からのトップダウン的なアプローチが弱い傾向がある
 - ・MBSE のような新たな手法を導入するには、組織的な取組みが必要であるため、導入に対する敷居が高い
- したがって、MBSE の普及・適用においては、技術的なノウハウだけでなく、趣旨や目的の理解を組織的に進め、導入の敷居を下げる事が重要である。

SEC は、開発方法論としての MBSE 啓発のため、「開発手法や予算等の決定権を持つ管理者」向けに導入の目的や効果を分かり易く解説した「MBSE 導入の手引き」を作成した。

本書は、MBSE 普及促進にむけた課題と考えられる、MBSE の効果や目的の理解に重点を置いて記載されているため、技術職の方でなくても読み進めることができるようになっている。

3 おわりに

今回作成した「MBSE 導入の手引き」の普及をはじめとして、SEC は、重要インフラなど一般利用者に大きな影響のある SoS の信頼性向上を目指し活動している。さらに品質説明力強化のための技術的取組みの一環として効果的な先進設計・検証技術の事例収集や普及活動も推進している。

なお、本書は 2013 年上期公開予定である。

ビジネス・プロセス改善の推進

SEC エンタプライズ系プロジェクト 研究員

倉持 俊之 室谷 隆 森下 哲成 大和田 裕

ビジネス・プロセス改善領域ではプロセス共有化 WG とプロセス改善 WG の 2 つのワーキンググループで活動してきた。プロセス共有化 WG ではソフトウェア開発の標準プロセスを検討し、実際に活動しているプロセスをいかに改善し継続するかを検討を行った。プロセス共有化 WG は「共通フレーム 2013」の発刊に向けて改訂、追記部分の作成、レビューを中心に活動した。プロセス改善 WG では、SPEAK-IPA と SPINA³CH 両方を使用したプロセス改善方法を、実証実験の形で実施した。また 2011 年度に手がけたプロセス改善推進者教育のコンテンツやカリキュラムは、セミナー開催を通じブラッシュアップを図った。これらの成果や改訂された SPEAK-IPA、SPINA³CH を総合的なプロセス改善のパッケージとして提供した。また 2 つの WG は共に、国際標準をベースにした活動を行ってきたが、2012 年度は更にワーキンググループの成果を国際標準に提案する活動も行った。

1 プロセス共有化 WG

プロセス共有化 WG は、既存のソフトウェア開発の標準プロセスである「共通フレーム 2007 第 2 版」を大幅改訂し「共通フレーム 2013」として 3 月 4 日に刊行した。サブタイトルとして「経営者、業務部門とともに取り組む「使える」システムの実現」とあるように、事業（ビジネス）に寄与するシステムを構築するためには何が必要なのかを

検討し、作成してきた。「共通フレーム 2007」からの大きな変更点は以下の 3 点である。

- ・ ベースとなるソフトウェア・ライフサイクル・プロセスの国際規格（JIS 規格）を JIS X 0160:1995（追補 1：2007 を含む）から JIS X 0160:2012 へ変更。
- ・ システム開発のプロセスと、ソフトウェア開発のプロセスを分離し、明確化。
- ・ システム・ソフトウェア開発のプロセスと運用・サー

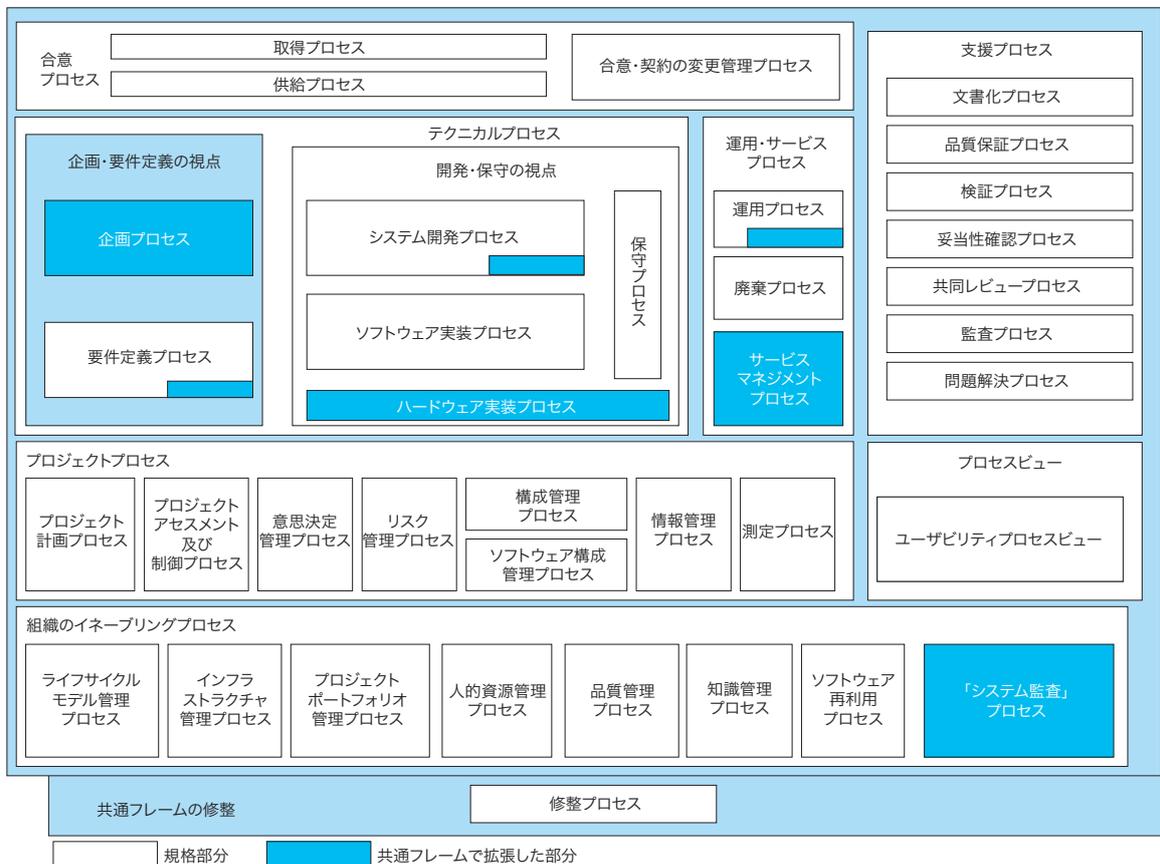


図 1 共通フレーム 2013 体系図

ビスのプロセスを連携。(運用・サービスの視点を要件に取り入れる)

「共通フレーム 2013」の改訂部分の詳細や解説は SEC ジャーナル 32 号に詳しく説明しているの、参照されたい。

2 プロセス改善 WG

(1) 活動概要

プロセス改善 WG は、IT 利活用及びシステム開発に係る組織能力向上をねらいとして、プロセス改善手法について研究し、その成果の普及を促進してきた。2006 年 1 月にプロセス改善部会として発足して以来、プロセス改善ナビゲーションガイド<なぜなに編><プロセス診断活用編><ベストプラクティス編><虎の巻編>の提供とプロセス診断ツール<SPEAK-IPA>、開発技術者自らが改善に取り組むツール<SPINA³CH (スピナッチキューブ) 自律改善メソッド>の公開を行ってきた。

これまでの実証実験での実績を踏まえ、プロセス改善を正しく理解し、ビジネスと改善活動の関係性を明確にして推進するためには、推進できる人材を増やすことが必要と考えた。2012 年度は、プロセス改善活動の推進を担う人向けに、<プロセス改善活用ガイド>を作成し、推進者育成で利用してきた教材と併せて公開した。(図 2 参照)

プロセス改善は、組織の状況によりトップダウンアプローチとボトムアップアプローチを的確に活用することが肝要と考えられる。そこで国際標準 ISO/IEC 15504 (プロセスアセスメント) に準拠したアセスメントモデル SPEAK-IPA と開発現場の課題解決から始める SPINA³CH 自律改善メソッド (図 3 参照) について、公開後の実証実験から得られた知見をもとに両ツールの改良を行い、ツールを活用した教材を用意し、普及・啓発を行った。

(2) SPEAK-IPA の改訂骨子

SPEAK-IPA では、実際にアセスメントモデルを使用して、

| 分類 | 名称 | |
|---------|--|---|
| | プロセス改善活用ガイド | |
| ガイド類 | プロセス改善ナビゲーションガイド ～なぜなに編～ ～プロセス診断活用編～ ～ベストプラクティス編～ ～虎の巻編～ | |
| | SPEAK-IPA アセスメントガイド | プロセス改善ナビゲーション ～自律改善編～ |
| ツール類 | アセスメントモデル SPEAK-IPA | SPINA ³ CH (スピナッチキューブ) 自律改善メソッド |
| | プロセス改善推進者入門 | |
| 教育 (教材) | アセスメントモデル活用コース | スピナッチキューブ活用コース |
| | アセスメント活用コース - 準アネッサ (ベーシック) コース - 準アネッサ (アドバンス) コース - 適格アセッサコース | |
| 事例 | プロセス改善セミナー事例紹介 | |
| | ベストプラクティス事例 | |

□ 今回発表/改訂 □ 既に発表済

図 2 これまでの成果と 2012 年度公開資料

組織 / プロジェクトの診断を行う時に役立つガイド、アセスメントシート、テンプレートなどを充実させた。

- ・ 経験の浅い改善推進者がアセスメントを実施するときの手引きとして「アセスメントガイド」を作成。
- ・ アセスメント実施時に使用するアセスメントシート、インタビュースクリプト例の他、幾つかの場面で使用するプレゼン資料をテンプレートとして提供。

(3) SPINA³CH 自律改善メソッドの改訂骨子

SPINA³CH 自律改善メソッドでは、実証実験で明らかになった改善点を反映し、より自律的な改善に役立つよう内容を充実させた。

- ・ 取り組むべき問題に対する業務 (プロセス) の選定をより分かり易くするため、一覧形式からマップ図に変更。
- ・ ツールの使い方や活用方法を解説した書籍「プロセス改善ナビゲーションガイド～自律改善編～」を刊行。

また、中小組織向けのプロセス改善を検討している国際規格ワーキンググループ (ISO/IEC JTC1/SC7/WG24) へ「SPINA³CH 自律改善メソッド」の考え方をまとめて規格化 (テクニカルレポート) 案を作成し、提案した。

(4) 普及活動

2012 年度は SEC セミナーを始め、SPI Japan 2012 の企画セッション参加など外部団体での普及・啓発も行った。プロセス改善は、組織活動そのものであり継続的に実施してこそ成果が得られるものである。これまでに公開してきた成果物が活用され、IT にかかわる多くの人たちとその組織が発展することを期待している。

今回報告した活動内容は、産学界から参画いただいている委員の皆様により推進されてきた。当領域を取りまとめていただいていた当領域の部会長である元富士通株式会社の村上憲稔領域長、東京海上日動システムズ株式会社の菊島靖弘副領域長をはじめ多くの委員の皆様へ、この場を借りて厚く御礼を申し上げる。

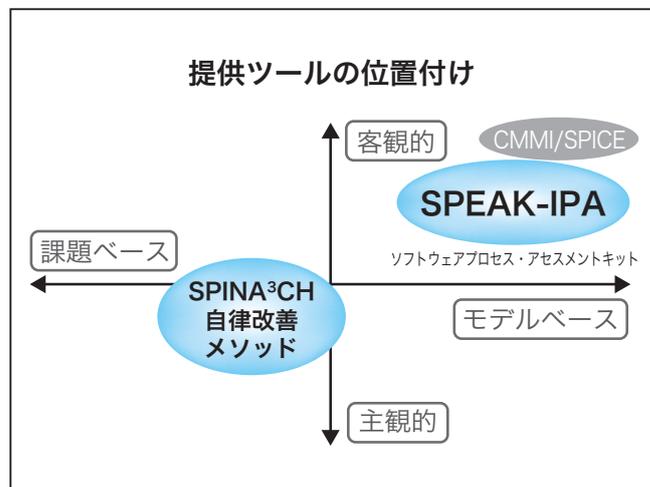


図 3 SPEAK-IPA と SPINA³CH の関係

超上流及び上流工程の 品質向上に向けて

SEC エンタプライズ系プロジェクト 研究員

森下 哲成 三毛 功子 鈴木 三紀夫 平林 大典 柏木 雅之

SEC は、設立以来、超上流や上流工程の重要性を提唱し、書籍やセミナーなどを通じて発信してきた。本稿では、2012 年度における超上流と上流工程の品質向上に向けた WG（ワーキンググループ）等の活動及び取組みに関連した成果物について以下の内容を報告する。

- ① 超上流工程の課題に関する調査・検討
- ② IT プロジェクトのリスク予防の実践的アプローチ（定量的管理基盤 WG）
- ③ 超上流工程の要求定義を革新する環境変化への対応（要求発展型開発 WG）
- ④ 非機能要求グレードの普及展開（非機能要求グレード WG）

なお、上記①～④に関する取組みの成果は、2013 年 3 月にまとめて公開した。^{※1}

1 超上流工程の課題に関する調査・検討

産業界では、超上流工程への意識が高まってきており、超上流工程における課題に対する具体的な取組みを始める企業が多数出てきている一方、依然としてその課題に起因したプロジェクトの失敗例があとを絶たない状況である。

そこで、超上流工程において、各企業がどのような課題認識を持ち、それをどのように解決しているのかを確認するために調査を実施した。課題やその解決策には様々なものがあるが、その内容を共有することで、超上流工程を実践する際のヒントを各社に得てもらうことを最終的な目標としている。

調査は、超上流工程（事業戦略・事業計画、システム化の方向性、システム化計画、要件定義）に限定し、ヒアリングシートにフリーで記入してもらうという方法を使った。シート回収後に内容を精査し、疑問点がある場合には、個別ヒアリングを実施した。ユーザ及びベンダの偏りがないよう、双方を対象とした。

表 1 課題及び解決策の件数

| 工程 | 課題 | 解決策 |
|-----------|-----|-----|
| 事業戦略・事業計画 | 29 | 16 |
| システム化の方向性 | 46 | 31 |
| システム化計画 | 54 | 39 |
| 要件定義 | 60 | 43 |
| 計 | 189 | 129 |

その結果、中堅・大手のユーザ、ベンダ各 10 社の計 22 名の方々から、表 1 にある課題及び解決策が収集できた。全体を眺めると、各社とも色々な策を講じてきたことがうかがえる。また、「ユーザならではの」「ベンダならではの」

という特別な課題は少なく、似たようなものが多い。つまり、どの企業でも同様の問題を抱える可能性があると思定できる。合わせて、情報共有に対するニーズが高いことも確認できた。また、開発標準への取込み、開発プロセスの改善などの活用方法や、報告書・ガイドブック、フォーラム・ワークショップ、HP での公開などの共有方法があげられた。

課題や解決策は、当然ながら各企業、あるいは開発プロジェクトの特性を踏まえたものとなり、第三者的に見た場合は、「この課題にはこの解決策」と一意に言えないものも多いと思われる。直接的な“答え”はないかも知れないが、「うちには合わない」と見切るのではなく、課題の背景も含めて内容を読み解くことで、少しでも役に立ちそうなものを発掘してもらうことを期待している。

2 IT プロジェクトのリスク予防の実践的アプローチ

SEC の設立以来、定量的管理の重要性を提唱し、書籍や SEC セミナーを通じて、多くのベンダの開発管理や計画等の取組み例を発信してきた。新しい取組みとして、2011 年度からはユーザ企業とベンダ企業の間で「プロジェクト目標達成」を阻害するものをリスクと捉えて、リスクの原因

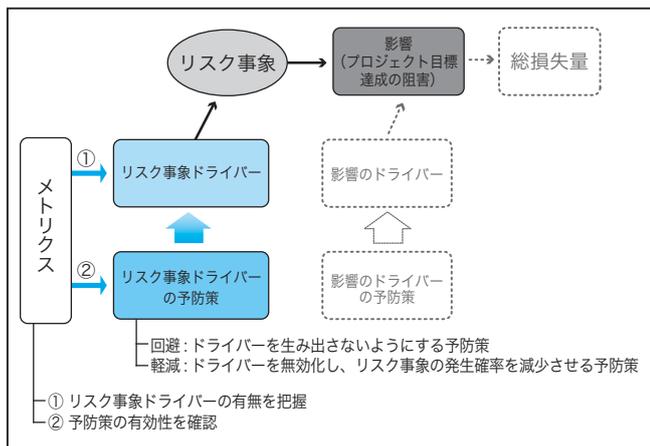
【脚注】

- ※1 システム障害を引き起こすリスク発現の要因と対策等に関する資料 3 点を公開
<http://www.ipa.go.jp/about/press/20130327.html>
 情報システムの非機能要求定義を効率的に行う手法を研修教材等にまとめ公開
<http://www.ipa.go.jp/about/press/20130311.html>

と結果の構造に着目し、リスクを顕在化させないための予防的措置に焦点を当てて検討を行ってきた。

(1) 発注者と受注者が協調してリスクの発現を予防する枠組み

リスクの予防的措置のためには、ITプロジェクトにおいて「どのようなリスクがあるのか」をユーザ企業とベンダ企業が一緒に議論し、重点的に管理すべきリスクを共通認識し、これを誘発する要因（ドライバー）に対して双方が働きかける必要がある。ドライバーに働きかけるためには、まず、その有無を客観的に把握する必要がある。そして、ドライバーを取り去る、またはリスク事象に至らしめないための対策を講じる。さらに、その対策が有効に機能しているかどうかを把握し、リスクを管理下におく。この一連のプロセスにおいて定量指標（メトリクス）を適用し、客観的な把握に努める。その考え方を纏めたものが図1である。また、それに基づいてリスクを整理した具体例を表2に示す。



※参考文献 [1] の標準モデルに対し、「メトリクス」を追加導入

図1 リスク事象の予防モデル

表2 モデル要素と例（抜粋）

| リスク事象（例） | リスク事象ドライバーの内容（例） | |
|-----------------------------|--------------------------------|------------------------------|
| プロジェクトの進行阻害 | ステークホルダ間の力関係がアンバランスである | |
| リスク事象ドライバー有無の把握（例（抜粋）） | | |
| 把握の観点 | | メトリクス |
| 交渉の余地のない契約・取決め事項がある | 存在する / 存在しない | |
| 異なる役割を持つステークホルダ間で指導・統括関係がある | 関係数 / ステークホルダ数 | |
| リスク事象ドライバーの予防策（例（抜粋）） | | |
| 対応区分 | 予防策の内容 | 予防策の有効性を確認するメトリクス |
| 回避 | フェーズ毎に制約事項・スコープ・納期調整のイベントを設置する | 制約事項・コープ・調整イベント数 / プロジェクト全期間 |
| 軽減 | プロジェクト外の目的やその優先順位を可視化する | 目的・優先順位記載数 |

(2) リスク事象ドライバー記述書（テンプレート）

リスク発現の要因と予防策の関係を明らかにし、予防策の検討を容易にできるようにする、リスク事象ドライバー記述書（テンプレート）を作成した。

また、WG委員がそれぞれの立場（ユーザ、ベンダ、プロジェクトマネージャ、情報システム部門、品質管理部門など）で重要なもの、頻出するものを経験に基づく言葉で記述した事例を24件取りまとめた。この事例の整理に当たっては、ユーザ、ベンダの協調作業において多くの接点がある「要件定義」に特に重点を置いた。

3 超上流工程の要求定義を変革する環境変化への対応

社会環境、経済環境、IT環境が相互に関連し不確実性を増しながら大きく激しく変化している。それも先の見えない不連続な変化である。また、社会インフラとなり得る企業の情報システムの規模も大きくなって多くのステークホルダが関与することになり、更に複雑さを増している。

このような状況の中で、情報システムは環境の変化に適応し続けることができなければ、企業の競争力が低下してしまう。従って、ビジネス環境変化に適時適切に対応する情報システムの構築が一層求められるようになってきている。

このような背景から設置した「要求発展型開発WG」では、環境の変化に着目し、2011年度はステークホルダ要求の整理と変化対応技術の取りまとめを行った。2012年度はステークホルダ要求に影響を与える環境の変化に着目し、その対応について議論を進めた（図2）。

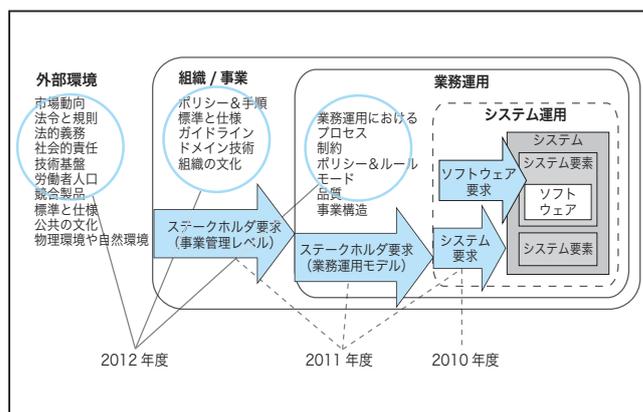


図2 ビジネスコンテキストとWG実施年度^{※2}

環境変化によってステークホルダ要求が変化する過程、つまり、企業や組織がどのように環境変化に対応するのか

【脚注】

※2 [1] ISO/IEC/IEEE29148, Systems and software engineering - Life cycle processes - Requirements engineering Figure 4 - Example of requirements scope in a business context より作成

について整理した。環境変化の確実度合いにより4つのレベル(1: 確実に見渡せる未来、2: 他の可能性のある未来、3: 可能性の範囲が見えている未来、4: まったく読めない未来)に分類し、不確実な変化に立ち向かう姿勢として留保(判断を保留する)、適応(変化に対応する)、形成(新たな未来を作る)の3つを設定した。

4つのレベルに3つの姿勢を掛け合わせると12の組合せが考えられるが、2012年度は変化適応モデル、アシュアランスケース、サービスデザイン、環境変化に対応するシステムに関する課題の4つを取り上げて検討した。(表3)

表3 不確実性レベル、戦略姿勢とWG実施テーマ

| レベル | 留保 | 適応 | 形成 |
|-----------------|---------------------|------------|----------|
| レベル1 | | アシュアランスケース | サービスデザイン |
| レベル2 (2012年度実施) | | 変化適応モデル | |
| レベル3 (2012年度実施) | | | |
| レベル4 | 環境変化に対応するシステムに関する課題 | | |

(1) 変化適応モデル

様々な環境適応モデルを分析した後、変化の特性を解釈し対応方針を決定する情報システム開発に適した変化適応モデルを考案した。(図3)

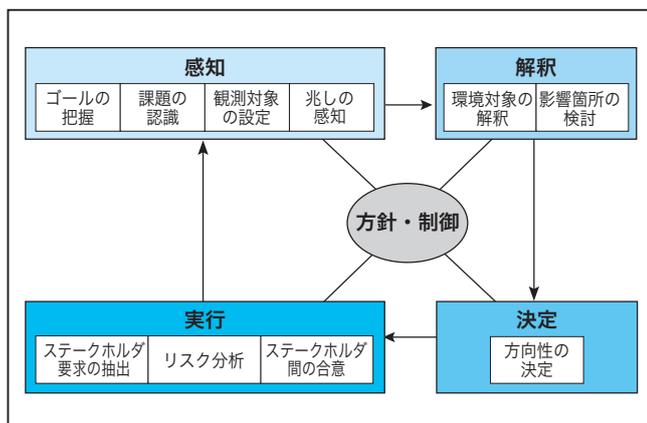


図3 情報システム開発の変化適応モデル

(2) アシュアランスケース

想定外事象を把握することで変化に強いしなやかな情報システム及び運用プロセスを構築するために、アシュアランスケースの活用について議論した。

(3) サービスデザイン

ビジネス環境の変化に対応するためには、「システム」としてだけではなく、「サービス」として捉える必要がある。サービスの設計方法を知ること、どのような要求がくるのか、どんな要求の変更があるのかを想定できるようになる。特にサービスのオープン化などの潮流に対するシステムの位置付けについて議論を進めた。

(4) 課題整理等

環境変化に対するソシオテクニカル観点^{※3}やシステム連携の観点からシステム運用の問題まで、複雑なシステムに残された課題について整理した。また、先駆的な企業5社にヒアリングし、その取り組み事例をまとめた。

4 非機能要求グレードの普及展開

非機能要求グレードの一層の普及を目指し、利用ガイド[活用編]、及び研修教材を作成した。

(1) 利用ガイド[活用編]

非機能要求グレードWGで非機能要求グレードの普及のための施策を検討した結果、非機能要求グレードを活用する局面での分かり易く解説したガイドを作成するという方針を決めた。WG委員が6分野(例: 新規開発の局面、運用の局面など)、合計29件の活用シーン(例: ユーザヒアリングに臨む心構え、稼働後に運用コストを削減するのは難しいなど)について物語風に記述し、それに解説を加えた。

(2) 研修教材

従来からSECでは講義(座学)形式のセミナーを実施してきたが、受講者などから演習を含むセミナーを実施してほしいという要望があがっていた。そこで、講義と5つの演習テーマ(例: モデルシステムを使った演習、発注者と受注者のロールプレイなど)から成る研修教材を作成し、自由に取捨選択や加筆・修正できるような形式にして公開した。また、この教材を用いた研修会を2013年3月13日に開催した。

【脚注】

※3 専門家ではない一般の人がITシステムを使うことによる課題の観点

【参考文献】

[1] プレストン・G・スミス、ガイ・M・メリット(澤田美樹子訳)『実践・リスクマネジメント』生産性出版(2003)

定量的プロジェクト管理の推進

SEC エンタプライズ系プロジェクト 研究員

秋田 君夫 大和田 裕 高橋 光裕 三毛 功子 森下 哲成

1 ソフトウェア開発プロジェクトデータの収集・分析

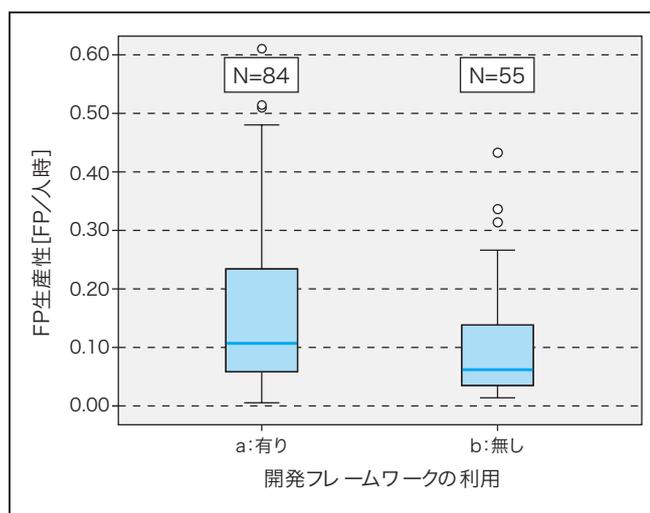
1.1 定量データの収集と分析

ソフトウェア開発データの活用によるシステムの信頼性向上等を目指し、2011年度に作成した、開発フレームワークの利用とソフトウェアの品質との関係(図2)などに関する新規分析項目を加えた素案を基に「ソフトウェア開発データ白書 2012-2013」[データ白書 2012](図1)を2012年10月1日に出版した。

また、白書の発行に伴い、定量データに基づくプロジェクト診断支援ツール^{※1}の保持データを2012-2013版にバージョンアップした。



図1 ソフトウェア開発データ白書 2012-2013



(説明) 開発フレームワーク「有」の方が生産性が高い

図2 新規分析の例 FP・新規開発・生産性

1.2 定量データの活用拡大と推進

より高品質なソフトウェア開発への適用と収集データの

活用拡大をめざして実施中の共同研究において、2件の研究成果が公開された。

① 東海大学との共同研究

「ソフトウェアプロジェクトデータにおける量的変数の予実差分分析」[SEC journal 31]を公表した。本研究において、新規開発においては全体に見積り不足であり、改修・保守における予実比率は、「大きなプロジェクト」では見積り過多、「小さなプロジェクト」では見積り不足の傾向となることなどが判明した。(表1)

表1 プロジェクト規模ごとの予実比率

| 開発種別 | プロジェクトの大きさ | 予実比率 (= 実績値 / 計算値) | | |
|-------|------------|--------------------|----------------|----------------|
| | | FP 規模 | 工数 | 工期 |
| 新規開発 | 「大きい」 | 1.15 (0.73) | 1.09 (0.57) | 1.01 (0.52) |
| | 「典型的」 | 1.09 (0.64) | 1.12 (0.60) | 1.04 (0.59) |
| | 「小さい」 | 1.03 (0.55) | 1.15 (0.62) | 1.08 (0.66) |
| 改修・保守 | 「大きい」 | 0.79 (0.27) | 0.96 (0.47) | 0.96 (0.40) |
| | 「典型的」 | 0.97 (0.47) | 1.08 (0.56) | 1.01 (0.53) |
| | 「小さい」 | 1.21 (0.69) | 1.22 (0.65) | 1.06 (0.65) |

プロジェクト規模: 大きい: $X + 2\sigma$ 、典型的: 計画値の平均値 X 、小さい: $X - 2\sigma$

() 内は実績値が計画値を上回る確率

② 法政大学との共同研究

「情報システム開発プロジェクトの属性データ分析法の考察」[IPSJ SIG122]を公表した。本研究において、コスト、品質、工期、プロジェクト総合を成功裏に終了したプロジェクトでは、要員数は少なく、複数の言語を使用していないという傾向があることを示した。(表2)

【脚注】

※1 定量データに基づくプロジェクト診断支援ツール: 2013年3月に「ソフトウェア開発データ白書 2012-2013」のデータに入れ替えた。

表 2 成功度と他の属性データの相関係数の抜粋

| 品質の成功度 | 相関係数 |
|--------------------|--------|
| 同言語別 _SLOC_2 | -0.404 |
| コストの成功度 | 相関係数 |
| プロジェクト全体のピーク外注要員数 | -0.518 |
| 平均外注要員数（製作） | -0.507 |
| レビュー指摘件数（基本計） | -0.5 |
| プロジェクト全体の平均外注要員数 | -0.472 |
| 平均外注要員数（詳細設計） | -0.465 |
| 平均要員数（基本設計） | -0.449 |
| 平均要員数（製作） | -0.426 |
| プロジェクト全体の実績工数（総計人） | -0.423 |
| プロジェクトの成功度 | 相関係数 |
| 平均要員数（基本設計） | -0.408 |

(説明) 相関係数とは2つの変数間にどの程度の関係があるかを示すもので、-1～1の範囲の値を取る。絶対値が大きくなるほど関係が強いことを表す

1.3 今後の予定

ソフトウェアの品質向上の観点で、品質を評価するための複雑度・難易度等の評価項目について検討し、収集項目の見直しを行う予定である。

2 定量的プロジェクト管理ツールの開発・普及

2.1 定量的プロジェクト管理ツール (EPM-X) の公開

地域・中小企業においてソフトウェア開発プロジェクトの定量的管理手法の普及を図るため、企業に広く普及している開発管理プラットフォームへのプラグイン形態での導入が可能な「定量的プロジェクト管理ツール」の開発を完了し、オープンソースとして公開（平成24年4月）した。公開後、実行形式ファイルが6,948件、文書ファイルが23,294件のダウンロードがあった。

2.2 定量的プロジェクト管理ツールの普及

同ツールの普及展開を図るため、以下の活動を実施した。

①セミナー、イベントでの普及

SECセミナーや共催セミナーにて本ツールの概要、使用例の説明を実施した。また、各イベント（SODEC, ET-WEST, ET等）へ出展し、ブースでの説明とデモンストレーションおよびブース内プレゼンテーションを実施した。（本年度12回）

②外部講演と個別訪問説明での普及

外部講演や外部勉強会での説明を実施した。また、各企業や団体へ訪問し説明を行った。（本年度：10社・6団体・3勉強会、累計：26社・7団体・6勉強会）

普及活動の結果として、約5社で本ツールが使用されて

いる。その中には独自で測定値をふやし新規グラフを作成している企業もでている。また、約15社で試用および試用検討が行われている。

課題としては、稼働環境のバリエーションが多いため動作確認が難しく使用環境によっては動作不安定になること、OSや本ツールで使用しているツール群のバージョンアップに対する動作確認が難しいことなどがある。

2.3 今後の予定

関連団体（実践的プロジェクトマネジメント推進協会：PPMA）や関連コミュニティによる普及展開活動に協力していく。

3 定量的プロジェクト管理手法の普及

昨年から継続して下記のSEC主催セミナーを実施すると共に、SODEC/ESEC2012、ソフトウェアジャパン2013等への出展等、定量的プロジェクト管理の普及に努めた。

表 3 定量的プロジェクト管理手法関連セミナー

| 番号 | 講座名 | 区分 | 実施回数 |
|-----|--------------------------|---------|------|
| I | 定量データ活動等によるITプロジェクトの見える化 | 基本編 | 2 |
| II | ITプロジェクトの見える化 | 応用編 | 2 |
| III | ソフトウェア開発データ白書と定量データの活用方法 | 応用編 | 5 |
| IV | 定量的品質管理とその実践取り組み | 実践編（事例） | 4 |

4 ベンチマーキング関連の国際規格

ソフトウェア開発プロジェクトのデータ収集・分析に関するIPA/SECの取組みが国際規格にも反映されるよう、SEC成果に基づく国際標準化の提案を進めている。それらの国際規格への反映を目指した活動を行った。その結果、日本案に基づくISO/IEC 29155-2（ITプロジェクト性能ベンチマーキング：実施手順）が国際規格最終案として承認され、平成25年中に出版される予定である。また、平成24年度からISO/IEC 29155-3（同：報告様式）、29155-4（同：データ収集と蓄積）の制定に着手し、いずれの日本案も審議文書として受け入れられ審議が進んでいる。

【参考文献】

- [データ白書2012]IPA/SEC：ソフトウェア開発データ白書2012-2013、2012
- [SEC journal 31] 古山恒夫：SEC journal No.31Vol.8 No.4、ソフトウェアプロジェクトデータにおける量的変数の予実差分析、IPA/SEC、2012
- [IPJS SIG122] 野下貴弘、森田貴之、江崎和博：研究報告 情報システムと環境社会、IPJS情報システムと社会環境研究会第122回研究発表会、情報システム開発プロジェクトの属性データ分析法の考察、情報処理学会、PP.1-6、2012

定量的な目標管理手法の普及活動の展開

～組織目標達成と IT 導入の整合性を図る「GQM+Strategies[®]」の活用～

SEC 調査役

新谷 勝利

SEC エンタプライズ系プロジェクト 研究員

平林 大典

「GQM+Strategies[®]」^{*1}は、企業の業種や規模、業務領域にかかわらず、企業や組織の経営レベルにおける「目標」と、その目標を達成するための「戦略」、また実務策定レベルにおける「個別戦術（例：IT化）」の3つの整合性を確認することができ、企業のIT戦略の立案および評価における経営層・業務部門・情報システム部門の関係強化に役立つ手法である。2012年度は、企業からの参加者及び大学教員を中心とする作業グループ（戦略意思決定プロセス強化WG）を設置し、研修教材の作成やワークショップの開催など同手法の普及活動を展開した。

1 実企業への適用と評価

WG委員企業の協力により「GQM+Strategies」を実際に社内で適用し、評価した。

① 【サービス業】年間のシステム開発計画策定に適用

多数あるIT化テーマ候補を、「GQM+Strategies」を使って整理・見える化することで、全社戦略・各事業部門の戦略に関するIT化テーマの選定、IT投資枠を考慮したIT化テーマの優先準備付けがし易くなった。

② 【製造業】中期のシステム投資計画策定に適用

各事業部門の目標とそれを実現するための戦略（施策）を「GQM+Strategies」を使って整理・見える化することで、IT投資が業務の何に貢献するのかが明確になった。また、経営層、業務部門長への説明がし易くなり、役員の理解も得易くなった。

③ 【レンタル業】コスト削減計画策定に適用

コスト削減の目標とそれを実現するための戦略（施策）を「GQM+Strategies」を使って整理・見える化した。

アイデアを論理的に整理・見える化することで、目標と施策の整合性がはっきりし、更に実現性確認のポイントが明確になった。

2 研修教材作成とワークショップ開催

戦略意思決定プロセス強化WGでは、日本での適用事例に基づく事例教材の作成やワークショップの開催（8回、東京、横浜、大阪、北海道など）、講義演習教材の翻訳、さらにはIESE^{*2}研究者を招いて共同でワークショップを開催（東京）するなど、普及活動を展開した。参加者からは「既にBSC^{*3}を導入している企業の場合、現場レベルでの整理に活用するのは有用であると思う」などの意見があり、普及が期待できる。

3 新たな展開

今後、「GQM+Strategies」普及のためのセミナーを引き続き開催すると共に、ITコーディネータ協会内に設置された研究会でのプロセスガイドラインに基づく「GQM+Strategies」の適用（BSC等との関係性・組合せ利用方法等）についての検討に協力する。また、早稲田大学が実施する調査・研究や教育等に協力し、更なる普及を後押しする。

4 謝辞

これまでの普及活動には、参画いただいている委員の皆様にご多大なるご協力をいただいた。実企業への展開に積極的に取り組んでいただいている伊藤忠テクノソリューションズ株式会社の野村典文委員、株式会社クニエの井出昌浩委員をはじめ、ご協力いただいた企業の皆様へこの場を借りて厚く御礼申し上げます。また、今後ともご協力をお願いする次第である。

【脚注】

- *1 GQM+Strategies[®] : Goal Question Metric + Strategies, 組織のゴールと結び付けたIT戦略の実施において、前提とする事実及び仮定への考察からゴール成就への影響とリスク評価を行う方法論。IESEが開発。ドイツにて商標登録。なお、GQMは、Goal Question Metricの省略形で1990年代後半からメリーランド大学のVictor Basili教授を中心にプロセス改善の方策の一つとして推進されてきている。
- *2 IESE : Institute for Experimental Software Engineering, ドイツ・ブラウンホーファー協会実験的ソフトウェア工学研究所。
- *3 BSC: Balanced Scorecard, 4つの視点（財務・顧客・業務プロセス・学習と成長）に基づいて、経営戦略立案や実行評価を行うフレームワークのこと。

【参考文献】

[SEC journal 30] 新谷, 他: SEC journal 30, p121, 企業・組織の目標達成とIT導入計画の整合化を実現するための手法推進、<http://www.ipa.go.jp/files/000024518.pdf>

非ウォーターフォール型開発手法の普及に向けて

SEC エンタプライズ系プロジェクト リーダー
山下 博之

SEC エンタプライズ系プロジェクト 研究員
柏木 雅之

最近、ますます注目度が増している非ウォーターフォール型、特にアジャイル型開発手法について、前年度に実施した普及要因と適用領域の拡大に関する調査の結果を受け、アジャイル型開発プラクティス・リファレンスガイドの発行と共に、同手法の普及促進を行った。

1 アジャイル型開発プラクティス・リファレンスガイド

非ウォーターフォール型、特にアジャイル型開発手法は、ビジネス環境の変化がそのスピード及び質の面で拡大する中で、変化への俊敏な対応が可能なソフトウェア開発手法として一層注目度が高まってきている。

2012年6月に公開した、「非ウォーターフォール型開発の普及要因と適用領域の拡大に関する調査報告書」^{※1}において「現場導入のナレッジ収集と活用するためのTips集づくり」の提言を受け、「アジャイル型開発におけるプラクティス活用事例調査」を実施し、国内でアジャイル型開発の活用が進む先駆的企業（26プロジェクト）の延べ59件のプラクティスの利用状況（図1）を調査・分類した。その結果をもとに、プロジェクト規模やアジャイル型開発の種類などのプロジェクト特性（表1）によるプラクティス適用状況の違い、適用する場合の工夫や留意点などを取りまとめ、調査報告書と共に、「アジャイル型開発におけるプラクティス活用リファレンスガイド」^{※2}（図2に説明例を示す）を公開した。

2 普及促進活動

アジャイル型開発手法に関する過去4年間のIPA/SECの取組み内容を整理した記事を掲載する[SEC journal 31]と共に、同手法の導入事例や人材育成を取り上げたセミナーを多数開催し、普及促進に努めた。

プラクティスの実態—『ふりかえり』

| | |
|--|---|
| <p>状況</p> <p>イテレーション毎に、チームは動くソフトウェアとして成果を出そうとしている。イテレーションを繰り返して、チームはソフトウェアを開発していく。</p> | <p>利用例</p> <p>●事例(25):当初はKPT^{※1}を用いてふりかえりを行っていたが、ファシリテータの技量にふりかえりの質が依存してしまう、声の大きいメンバーに影響を受けてしまうことに気づいた。そのため、意見を集めるやり方として、555(Triple Nickels)^{※2}を用いることにした。</p> |
| <p>問題</p> <p>開発チームは、そこに集まったメンバーにとって最適な開発プロセスを、最初から実践することはできない。</p> | <p>留意点</p> <p>●ふりかえりにチームが慣れていない場合は進行で各人の意見をうまく引出すようにしないとうまくいかない。 ●問題を糾弾する場にしてしまうと、改善すべき点を積極的に話し合えない場になってしまう。 ●改善案を出しても、実際に実行可能なレベルの具体的なアクションになっていないと実施されない。</p> |
| <p>フォース</p> <p>イテレーションでの開発はうまくいくこともあるが、うまくいかないこともある。</p> | |
| <p>解決策</p> <p>反復内で実施したことを、反復の最後にチームでふりかえり、開発プロセス、コミュニケーション、その他様々な活動をよりよくする改善案をチームで考え実施する機会を設ける。</p> | |

※1 メンバ全員で、Kepp(よかったこと・続けたいこと)、Problem(問題・困っていること)、Try(改善したいこと・チャレンジしたいこと)を出し合い、チームの改善を促す手法。
※2 アクションや提案に対するアイデアを出すための手法。5人程度のグループで、各人が5分間ブレインストーミングをしてアイデアを書き出す。5分経過後は紙を隣の人にもらい、新しいアイデアを書き加える。

図2 ガイドにおけるプラクティスの説明例

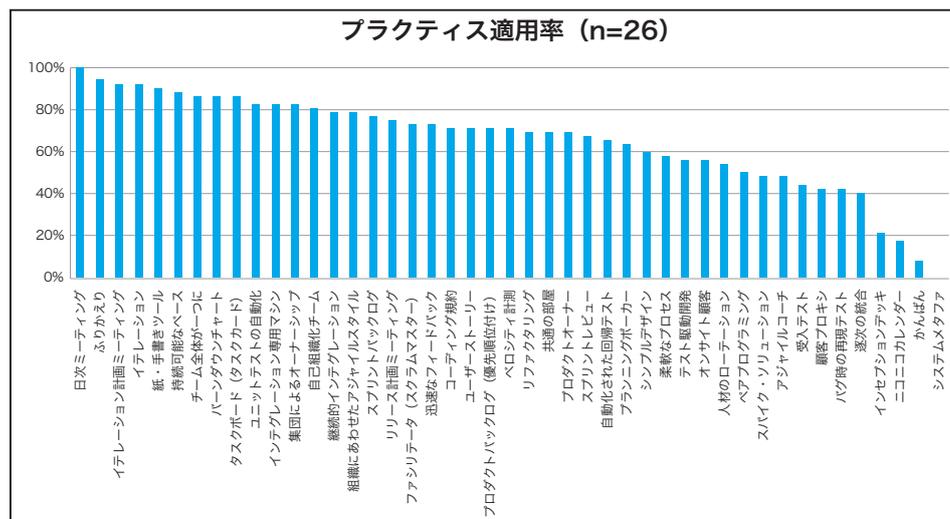


図1 アジャイル型開発プラクティスの利用状況

表1 プロジェクト特性

| | |
|----|-----------------|
| a) | 開発期間が短い |
| b) | 成果物の対象範囲の変動が激しい |
| c) | 求められる品質が高い |
| d) | コストの制約が厳しい |
| e) | 開発者のスキルが未成熟 |
| f) | 初めての技術・業務領域を扱う |
| g) | 新しいチームで開発する |
| h) | 分散開発をする |
| i) | 初めてアジャイル型開発をする |

【脚注】

- ※1 <http://sec.ipa.go.jp/reports/20120611.html>
- ※2 <http://sec.ipa.go.jp/reports/20130319.html>

【参考文献】

[SEC journal 31] 山下、柏木：非ウォーターフォール型（アジャイル）開発の動向と課題, SEC journal No.31 Vol.8 No.4 pp.164-172, 2012

IT サービス継続計画の策定を推進

SEC エンタプライズ系プロジェクト 研究員
柏木 雅之

SEC エンタプライズ系プロジェクト リーダー
山下 博之

IT サービスの回復力（レジリエンス）の底上げを目的に、「情報システム基盤の復旧のための対策調査報告書」のとりまとめを行うと共に、IT サービス継続計画（以下、IT-BCP）策定のための指針「高回復力システム基盤導入ガイド」（以下、ガイド）の事例編を取りまとめた。^{※1}

1 情報システム基盤の復旧のための対策調査

IPA/SEC では、2011年3月の東日本大震災を受け、IT-BCP において、情報システムの高回復力を確保することの重要性を再認識し、企業における IT-BCP 策定の取組み状況や個別対策の実施状況を確認する必要があると考えた。そこで、「情報システム基盤の復旧に関する対策の調査（以下、本調査）」として、文献調査、アンケート調査、ヒアリング調査の3つを実施した。そのうち、アンケート調査結果の概要は次の通りである。

- IT-BCP 策定済企業は調査対象の 1/4 にとどまる。
- 事業規模や IT 依存度が大きいほど取組みは進んでいる。
- IT-BCP 策定済企業であっても、IT サービス継続マネジメントに満遍なく取り組めていない。特に、事業影響度分析（BIA : Business Impact Analysis）、教育訓練計画、維持改善計画等が不十分。
- IT サービス継続戦略と対策との間で不整合が散見。例えば、システムの目標復旧時間（RTO）を6時間未満に設定していると回答した企業のうち約 1/4 が、サーバを冗長化していなかった。これは、設定した時間内にシステムを復旧するために必ずしも適切なシステム構成が採用されていないということである。（図1参照）

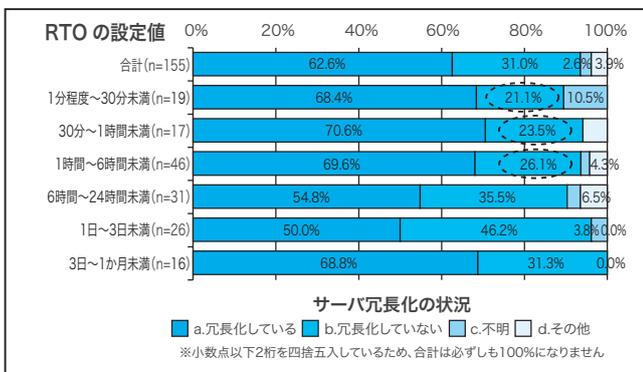


図1 目標復旧時間（RTO）とサーバ冗長化

2 高回復力システム基盤導入ガイド

2011年度に作成した、ガイドの概要編と計画編^{※2}に続き、

前述の報告書を基に 10 件のシステム基盤導入事例をまとめた事例編を作成した（表1参照）。

表1 システム基盤導入事例一覧

| モデル | 業種 | 最重要システム | 対策の特徴 |
|------|---------|--|--|
| モデル1 | 製造業 | 生産管理 / 顧客管理 / 財務・会計管理システム、メール・グループウェア等 | ・バックアップデータをメインサイトとは別の堅牢な建物の施錠キャビネット内に保管。 |
| | 情報サービス業 | メールシステム、顧客用開発システム（ファイルサーバ）等 | ・遠隔地にオンラインバックアップを実施。 |
| | 地方公共団体 | 住民基幹系情報システム（住民記録・税・福祉） | ・東日本大震災で庁舎が津波により被災。遠隔地のバックアップデータで一部のデータを復旧。 |
| モデル2 | サービス業 | Web サービス（情報提供）、ブロガーを管理するプライベート SNS | ・東日本大震災で庁舎が津波により被災。遠隔地のバックアップデータで一部のデータを復旧。 |
| | 地方公共団体 | 住民情報系基幹システム | ・メインサイトは自設データセンターで冗長化、遠隔地バックアップを実施。 |
| モデル3 | その他 | 契約者情報管理システム | ・阪神淡路大震災を契機に新規にシステムを構築し、同時に遠隔地にバックアップサイトを設置。 |
| モデル4 | サービス業 | ERP、メール・グループウェアシステム | ・2箇所の民間データセンターを活用し、バックアップサイトを構築。クラウドサービスも活用。 |
| | サービス業 | 検査・認証情報データベース | ・サーバ仮想化技術採用し、海外にバックアップサイトを設置。 ・同期バックアップを行い、震災時にフェイルオーバーを実施。 |
| | 金融・保険業 | 売買システム | ・システム障害対策のため、メインサイトは三重化を実施。 |
| | 製造業 | 製造・販売・管理システム、メール、CTIシステム等 | ・水害によりサーバの水没の経験 ・仮想化技術採用し、遠隔地サイトに同期バックアップを実施 |

事例編では本調査のヒアリング調査で収集した事例をもとに、「概要編」、「計画編」で示した4つのモデルシステムにおける要件定義の推奨値を、実際にどのように調整し、カスタマイズしたのかを参照できるようにした。

本ガイドの内容については、参考文献の [SEC journal 30] を参照されたい。

3 普及活動

セミナーや各種イベントでの講演等により、IT-BCP 策定の重要性を訴えた。

【脚注】

- ※1 「情報システム基盤の復旧に関する対策の調査」報告書等の公開 <http://sec.ipa.go.jp/reports/20120725.html>
- ※2 <http://sec.ipa.go.jp/reports/20120508.html>

【参考文献】

[SEC journal 30] 柏木、山下：IT サービス継続計画に向けて，SEC journal No.30 Vol.8 No.3 pp.116-120, 2012

組み系ソフトウェアプロジェクトの状況

SEC 組み系ソフトウェアプロジェクト リーダー

三原 幸博

SEC 組み系ソフトウェアプロジェクト 研究員

松田 充弘

SEC 組み系ソフトウェアプロジェクト 研究員

石井 正悟

SEC 組み系ソフトウェアプロジェクト 研究員

濱田 直樹

SEC 組み系ソフトウェアプロジェクト 研究員

石田 茂

SEC 組み系ソフトウェアプロジェクト 研究員

十山 圭介

1 組み系ソフトウェアプロジェクトの概況

SECではこれまで組み系ソフトウェアの高品質化を中心に開発業務の改善を後押しすべく開発・管理レファレンス・ガイド“ESxR¹シリーズ”の整備と普及を進めてきており、多くの企業・技術者の方々に利用していただいている。

2012年度のSEC組み系ソフトウェアプロジェクトでは、一昨年から整備に取り組んできたエンジニアリング工程におけるガイドの「組み系ソフトウェア向け設計ガイドESDR [事例編]」、「組み系ソフトウェア開発における品質向上の勧め [テスト編～事例集～]」(ESTR)、「組み系ソフトウェア開発における品質向上の勧め [バグ管理手法編]」(ESBR)を出版し、解説セミナーを実施した。これによりESxRシリーズは、ほぼ全領域をカバーすることができた。また、これらの成果を民間に移管し、自らで普及・活用ができるようトレーナー養成教育講座の開発と実施・教材提供を昨年度から始めている。ESPR²、ESMR³についてはトレーナー養成教育講座を開発しトライアルを含め一般向けにセミナーを実施すると共にWEBにて教材の提供をしている。

また国際化対応の一環としてESxRシリーズの英訳を進め、2012年度は新たにESPR,ESCR⁴(C),ESCR(C++)を翻訳した英訳版をWEBにて提供した。また、MISRA-Cの2012版への改版に関しては、ドラフト版の評価に協力した。MISRA-C改版に合わせてESCRの改訂も計画している。有識者の皆様の協力をいただきながら改訂作業を進めていきたいと考えている。

2 ESDRの出版

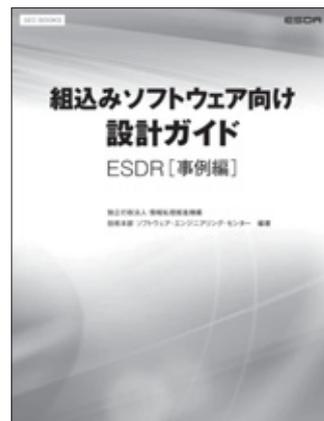
2.1 背景と狙い

組み系ソフトウェアの不具合は、その発見が遅れるほど修正のためのコストが飛躍的に増大することが知られている。したがって、設計段階で十分な作りこみをおこない、不具合の潜在化を未然に防止することは、品質の高いソフ

トウェアを効率良く開発するために極めて効果的である。ところが、こうしたノウハウは、開発組織や会社の中に閉じてしまっており、一般的に利用できるものはほとんどなかった。

一方、ソフトウェアの設計には、異なる設計場面でさまざまな共通の「パターン」があることが知られており、書籍も多数出版されている。そのパターンは特定のドメインや体系、手法などを前提として抽象化されているので、適用可能な場面では強力な道具となりうる。しかし、組み系ソフトウェア設計の現場ではドメインが固定し、既存の様々な道具立てを用いるために体系や手法も先に決まっております、そこで共通に適用できる「パターン」を見つけるのはかならずしも容易ではない。

このような状況をふまえSECは、国内の様々な企業から提出された事例から始めて、実際に現場で活用できる設計のノウハウを提供することを目標としてESDR[事例編]を作成することとした。事例に含まれる経験を活かして具体性を重視しつつ、ノウハウをドメイン・体系・手法によらずに様々な場面で適用できる「作法」として整理した。



2.2 書籍の内容

① 作法の枠組み

すべての事例を以下の9つの項目から成る「作法」に当てはめて整理した。

【脚注】

- ※1 ESxR: Embedded System development exemplar Reference
- ※2 ESPR: Embedded system development Process Reference
- ※3 ESMR: Embedded system development Management Reference
- ※4 ESCR: Embedded system development Coding Reference

- 1) 場面による分類
後述の、設計者の置かれた場面による大分類。
- 2) 作法名称
作法番号と共に作法内容をできるだけ簡潔に表すタイトル。
- 3) 作法概要
作法の具体的な内容を捉えることのできる概説。
- 4) メリット
作法を適用することによって得られる利点。
- 5) 留意点
作法を適用することによってデメリットになり得る点や、その他の注意点。
- 6) 解説
作法内容の解説。
- 7) 例
作法の適用例。「何か問題が起こった際に作法を用いて解決する」といったできるだけストーリー立てた内容を記述。
- 8) 関連する品質特性
JIS X 0129-1 で規定する外部及び内部品質特性と利用時の品質特性から関連が深いものを記述。
- 9) 関連する作法
ESDR に収録した作法の中で関連の深いものを記載。

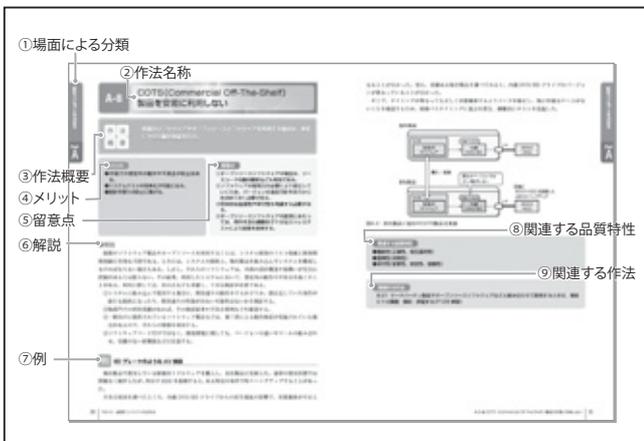


図 1 作法のフォーマット

② パート分け

設計者の置かれた場面・視点により作法を以下の4つのパートに分類した。

Part A: 設計コンセプトを決める

開発や設計の方針など組込みソフトウェア開発の全体にかかわるノウハウ。

Part B: システムレベルの設計の工夫

アーキテクチャなど、組込みソフトウェアを組み上げる枠組みに関するノウハウ。

Part C: ミドルウェア・ネットワークスタック・ライブラリ

レベルでの工夫

様々なサービスを提供する中間の層に関するノウハウ。

Part D: システムで扱う周辺デバイス操作に関する工夫

組込みソフトウェアに特徴的な、周辺デバイスとのインターフェースに関するノウハウ。

③ 作法内容

ESDR[事例集]は、実際の現場の問題意識を反映して、内容が多岐にわたっているのが大きな特徴である。なかでも設計の核心とも言えるソフトウェアの構造上の工夫として、以下のような例を含むさまざまなノウハウを提供している：

- ・実行効率（速度だけでなく資源利用の面も）の向上
- ・大量のデータを取りこぼすことなくハンドリングする
- ・イレギュラーな事象の影響を最小限に抑える

その他、処理速度、資源利用の性能向上と処理の単純化、保守性のバランス、差分開発や改善など既存ソースコードへの対処方法、不具合を作りこまないためのチェックポイントなどを記載している。

2.3 今後の課題

既に IPA が主催・共催するセミナーで数回にわたって ESDR[事例集]の解説をおこなった。「作法」という定型フォーマットに経験を整理することに対してセミナー参加者から賛同と共に「事例を提供してもよい」「現場の匂いのする教材として大学教育で活用したい」といった様々な声もいただいている。

今後は、利用者からのフィードバックをもとに作法の枠組みや既存の記述の精度を高めると共に、事例を増やすことにより体系化やドメインごとに整理する等の展開を計っていきたい。

また、本事例集には設計固有の領域だけでなく、要求や企画といった段階での入念な準備により設計を容易にする作法も含まれている。このように上流の開発プロセスのノウハウを抽出し整理する、といった設計領域だけにとどまらない発展の可能性もあるものと考えている。

3 ESTR の出版

「組込みソフトウェア開発における品質向上の勧め [テスト編～事例集～]」(ESTR)を製本化し、2012年11月12日に発行した。

3.1 背景と狙い

近年、組込みソフトウェアは需要の急拡大に伴い、品質



や信頼性・安全性などが重視される状況になってきている。2011年および2012年版組込みソフトウェア産業実態把握調査報告書によると75%以上もの製品で出荷後に不具合が発生し、そのうち45%以上はソフトウェアの不具合が原因であるという結果であった。組込みソフトウェアを開発する多くの企業は、同報告書の結果を待つまでもなく製品出荷後の不具合発生を削減するためにソフトウェア品質の向上に向けた活動を進めている。特に、テストは市場で製品不具合を発生させないための最後の砦であることから、テストの重要性が強く認識されてきている。しかしながら、実際のソフトウェア開発現場では、様々な理由から十分なテストを実施することが難しく、またソフトウェアはどんなにテストしてもバグがゼロには成り得ないという現実があり、どこまでテストすれば十分なのかの判断に苦慮している、というのが実態である。

そこで、IPA/SECでは2011年4月に「テスト部会」を立ち上げ、ソフトウェア開発におけるV&V (Verification & Validation) の重要性を踏まえ、品質向上のための『実践的』なV&V改善のため、ソフトウェア開発現場が抱える課題を抽出し、先進企業におけるそれら課題に対する考え方や対策事例を収集し、多くの開発現場で改善策検討時の参考となるよう、V&Vの指針や目安をまとめて提供することとした。

本書はベストプラクティス集ではなく、本書に取り上げた事例は、先進的事例というよりもむしろ「当たり前」と考えられているものが多いかもしれない。先進企業の識者達が、そうあるべきと考え、実際そのように実践している事例を示すことで、読者が自社の実態と他社の実態を比較し、自社の取組みの妥当性を議論し、そして他者への説得の際に参照可能な情報となるであろう、という考えに基づいて編纂したものである。

3.2 書籍の特徴

① 3つの観点による整理

本書は次の3つの観点によるパートから構成されており、これらは組込みシステムのソフトウェアテストが直面している技術的、マネジメントの実状に基づく視点であり、利用者の参照しやすさにも配慮し全体で36事例を掲載した。

Part1. テストの役割と限界・・・19事例

Part2. テストへの要求と対応状況・・・8事例

Part3. テストの基本的テクニック・・・9事例

② 実開発現場の成功・有効事例を収集して加工

テストに関わる品質メトリクスや基準値、標準ルールなど実際の開発現場で実践され、その有効性が実証されている内容を、許諾を得た上で可能な限り提示している。また、

自社、自部門における現状の検証活動との比較検討や参考とすることができるよう、図表・グラフなどで事例内容を分かりやすく解説している。

③ 定型フォーマット、Column、用語集付きで記述

全体の統一感を保つため、3.4に示すような構成とした。この際、既発行のESxRシリーズにおける「作法」、「解説」との構成上の統一性も考慮した。

3.3 対象領域

W字モデル開発プロセスにおけるV&Vの対象領域としては、次の領域がある。

①設計レビュー/モデル検証/コードレビュー

②テスト計画

③テスト実施/結果レビュー

SECがソフトウェア開発現場の方々に対して行ったアンケート/インタビューでいただいたコメントから、実際の組込みソフトウェア現場ではとくに②と③に大きな、そして差し迫った課題が潜在しているという理解に基づき、本書では②と③のいわゆる狭義のテスト領域を記述の主な対象とした。

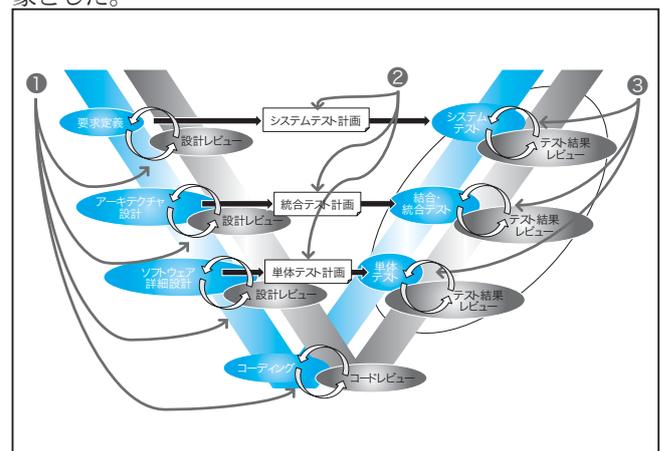


図2 V & Vの対象領域

3.4 事例集構成

本事例集は以下のような構成とした。

【見出し】ESxRシリーズの“作法”、“作法詳細”にならない、各事例での対策の基本的な考え方あるいは重要事項を端的に記述。

【解説】開発現場で実践している事例の内容を図表なども活用して解説。

【留意点】事例の内容に関して留意すべき点、関連するトピックスなどを記述。

【事例】具体例を挙げて要点を示しながら記述。

【Column】事例解説に関連した補足的な事項を記述。

【用語解説】本書で頻繁に用いる用語や勘違いしやすい用語を解説

4 トレーナズトレーニング教材の一般公開

IPA/SECは、組込みソフトウェア開発にかかわる産業界の知見やノウハウを集めて整理した、組込みソフトウェア向け開発リファレンスガイド ESxR シリーズを出版し、多くの人達に利活用してもらうために、SEC 研究員 / 専門委員が講師になり、解説セミナーや演習セミナーを開いて来た。しかしながら、ドメインに応じた解説などの受講者が要望する内容へのカスタマイズという面で制約があった。そこで、受講者側での独自セミナーの開催を可能にし、ESxR シリーズの一層の普及を促進するため、2010 年より講師向けのトレーナズトレーニング教材の開発を開始した。2011 年に完成した「ESCR トレーナズトレーニング教材」は、ソフトウェアの実装品質向上を推進する人向けに C 言語のコーディング規約を制定したものである。これに続き、2012 年度は、標準的な開発プロセスを定義した ESPR に基づいて、組込みソフトウェア開発向けに開発プロセスのテラリングを指導するための教材「ESPR トレーナズトレーニング教材」、および組込みソフトウェア開発のプロジェクト計画書作成を指導するための教材「ESMR / ESMG^{※5} トレーナズトレーニング教材」を開発し、これらの教材を一般公開した。



図3 公開済みのトレーナズトレーニング教材

4.1 教材の利用イメージと構成

これら ESCR、ESPR、ESMR/ESMG の3テーマのトレーナズトレーニング教材は、図1に示す通り、トレーナーが一般受講者向けにセミナーを開催する場合（(1)ESxR 解説セミナー）に利用する教材とトレーナー自身を養成するためのセミナー（(2)ESxR トレーナー養成セミナー）用の教材とから構成される。

(1) ESxR 解説セミナー用教材

ESxR 解説セミナーは、一般受講者向けに、ESxR の概要や ESxR の利活用の仕方を理解してもらうために開催される。その教材は、講義用教材とグループ演習用教材とで構成されており、受講者のスキルレベルに応じて、

- ①基礎編コース
 - ②中級編コース
- の教材2つを用意している。

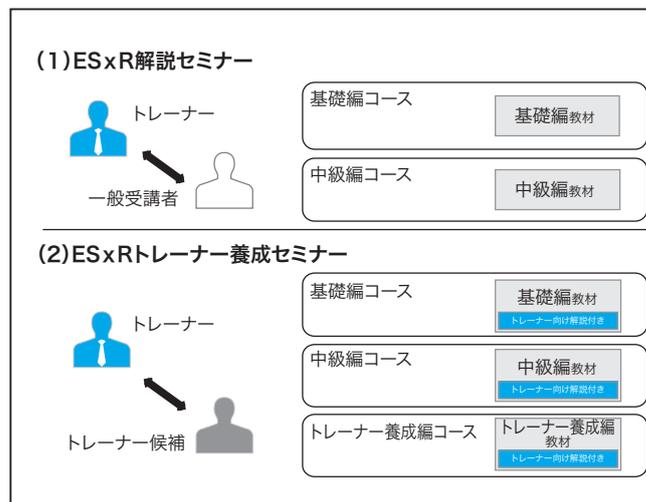


図4 セミナーの種類と教材

(2) ESxR トレーナー養成セミナー用教材

ESxR トレーナー養成セミナーは、ESxR 解説セミナーの「基礎編コース」や「中級編コース」を自組織で開催するための ESxR 解説セミナー講師向けに、その解説の仕方や留意点、教材カスタマイズのポイント等を習得してもらうためのものである。解説セミナー用教材に講師向けの解説を付加した

- ①基礎編コース
 - ②中級編コース
- の教材と、さらには講師が所属する組織内で「基礎編コース教材」や「中級編コース教材」を活用して、仕事のやり方を変えようとする場合のノウハウが修得できる
- ③トレーナー養成編コース
- の教材を用意している。

4.2 教材の内容

教材として昨年度までに下記を開発し、公開している。

- ① ESCR トレーナズトレーニング教材
 - ② ESPR トレーナズトレーニング教材
 - ③ ESMR/ESMG トレーナズトレーニング教材
- (1) ESCR トレーナズトレーニング教材

ESCR トレーナズトレーニング教材は、「コーディング作法ガイド [C 言語版]」(ESCR)を用いて、組織内でコーディング規約を制定し、ソフトウェアの実装品質向上を推進することを目的に開発した。基礎編コースとトレーナー養成編コースで構成される。

【脚注】

※5 ESMG : Embedded Systems development Management planning training Guide

| 教材名 | | 概要 |
|---------------------|---------------------------|--|
| 基礎編 コース | ・講義 ・グループ演習 (半日コース) | 各組織におけるコーディング規約の策定方法と、コーディング規約を浸透させる上での留意点を解説する。 |
| トレーナー 養成編 コース | ・講義 ・グループ演習 (1日コース) | 基礎編コースセミナーを組織内で開催するための一連のプロセスを修得する。 |

(2) ESPR トレーナーズトレーニング教材

ESPR トレーナーズトレーニング教材は、「開発プロセスガイド」ESPR をテーラリングして、組織内に適した開発プロセスを構築する方法を指導することを目的としている。

| 教材名 | | 概要 |
|---------------------|---------------------------|--|
| 基礎編 コース | ・講義 ・グループ演習 (半日コース) | プロセス・テーラリング時における ESPR の活用方法とテーラリングの基本的な進め方を修得する。 |
| 中級編 コース | ・講義 ・グループ演習 (半日コース) | 厳しい開発条件に適用できるような開発プロセスを構築するためのプロセス・テーラリングの実践的なスキルを修得する。 |
| トレーナー 養成編 コース | ・講義 ・グループ演習 (半日コース) | 組込みソフトウェアを開発する組織/部門/プロジェクトにおけるプロセス整備やプロセス改善を促進するノウハウを修得する。 |

(3) ESMR/ESMG トレーナーズトレーニング教材

ESMR/ESMG トレーナーズトレーニング教材は、「プロジェクトマネジメントガイド [計画書編]」(ESMR)、「プロジェクト計画立案トレーニングガイド」(ESMG) を活用して、組織内にプロジェクト計画書に基づいたソフトウェア開発を推進させるための教材である。役に立つプロジェクト計画書の書き方、計画の立案方法を理解させるためのトレーナーを養成することを目的に開発した。

| 教材名 | | 概要 |
|---------------------|---------------------------|---------------------------------|
| 基礎編 コース | ・講義 ・グループ演習 (半日コース) | プロジェクト計画書の記載項目と記述様式を習得する。 |
| 中級編 コース | ・講義 ・グループ演習 (半日コース) | プロジェクト計画立案方法を習得する。 |
| トレーナー 養成編 コース | ・講義 ・グループ演習 (半日コース) | 役に立つプロジェクト計画書作成の仕方を推進する方法を習得する。 |

4.3 教材の特徴と利用許諾

開発した教材は改変を認め、編集可能な電子ファイル (doc/docx, xls/xlsx, ppt/pptx) で提供している。教材の普及活動を民間へ移管することを後押しするために営利目的

での利用も認めている。



図5 教材の利用許諾

4.4 トレーナー養成実績

2012 年度に開催した ESxR トレーナーズトレーニングセミナーの修了者数を以下に示す。

| 実施日 | 場所 | 主催 | 修了者数 | 備考 |
|-----------------------|----|---------|--------|-------|
| ■ ESCR トレーナー | | | | |
| (2011 年度までの修了者数) | | | (42 名) | |
| 2012 年 9 月 27 日～28 日 | 東京 | IPA/SEC | 8 名 | 一般公募 |
| 合計 | | | 50 名 | |
| ■ ESPR トレーナー | | | | |
| 2012 年 7 月 12 日～13 日 | 東京 | IPA/SEC | 14 名 | トライアル |
| 2013 年 3 月 7 日～8 日 | 東京 | IPA/SEC | 10 名 | 一般公募 |
| 合計 | | | 24 名 | |
| ■ ESMR/ESMG トレーナー | | | | |
| 2012 年 12 月 13 日～14 日 | 東京 | IPA/SEC | 14 名 | トライアル |
| 合計 | | | 14 名 | |

4.5 まとめ

トレーナーズトレーニング教材の提供は、2011 年度までは、トレーナーズトレーニングセミナー修了者に対して行って来たが、2012 年度には、セミナーの受講がない場合でも、自習できるように、教材を充実させ、IPA/SEC ホームページから教材のダウンロードも行えるようにした。ESCR トレーナーズトレーニング教材は、2012 年 9 月、ESPR トレーナーズトレーニング教材と ESMR/ESMG トレーナーズトレーニング教材は、2013 年 3 月に一般公開した。

5 普及展開

5.1 バグ管理手法

組込みソフトウェア品質の確保でかつ効率的な確保のために、「組込みソフトウェア開発における品質向上の勧め [バグ管理手法編]」(ESBR) を出版した。

本書は、ソフトウェア開発現場で、“バグ”を期限内に漏れなく分析し、原因個所の特定や修正、正しく修正されたことの確認を困難にしてい



る組込みソフトウェア特有の要因を排除し、“バグ”の発見から担当の割り振り、原因究明、修正、確認、承認など一連の管理プロセスに則った組織活動としての“バグ”管理においてどのような考え・手法・工夫・基準を用いているか、具体的な事例をベースにして解説している。

5.2 MISRA 連携

MISRA は、C 言語の規格書 ISO/IEC 9899:1990 に準拠して MISRA-C を策定していたが、2012 年春には 1999 年に発行された C 言語の規格書 ISO/IEC 9899:1999 に対応した改訂版のドラフトを作成した。IPA/SEC にもこのドラフトに対するコメントを求めてきたため、ESCR を作成した高品質部会にて対応しコメントを返した。2013 年 3 月 18 日には正式版 (MISRA-C2012) が公開されたが、その中で ESCR の一部が参照されている。

5.3 英訳

「組込みソフトウェア開発向け開発プロセスガイド」(ESPR) Ver 2.0、「組込みソフトウェア開発向けコーディング作法ガイド [C++ 言語版]」(ESCR) Ver 1.0 の英訳を実施した。また既訳の「組込みソフトウェア開発向け品質作り込みガイド」(ESQR) Ver 1.0、「組込みソフトウェア開発向けコーディング作法ガイド [C 言語版]」(ESCR) Ver 2.0 とも併せてウェブサイト公開した。これらの英訳版ガイドは、13ヶ国・地域の延べ 22 社に提供されている。(図 6)

| 利用国または地域 | 利用目的 | 利用分野 | E S C R | E S Q R | E S P R | |
|-----------|-----------------|--------------------|------------------|------------------|------------------|---|
| 1 | 中華人民共和国 (4件) | 現地技術者の指導・教育 | | ● | | |
| 2 | | 現地品質担当者の指導・教育 | | ● | | |
| 3 | | 現地技術者の指導・教育 | 組込み製品 | ● | ● | |
| 4 | | 海外関連会社向けコーディング規約作成 | 組込み製品 | ● | | |
| 5 | インド (2件) | 日本市場向け製品の品質向上 | | ● | | |
| 6 | | 現地技術者の指導・教育 | 組込み製品 | ● | ● | |
| 7 | ベトナム (2件) | 現地技術者の指導・教育 | | ● | | |
| 8 | | 現地検証技術者の指導・教育 | 組込み製品 | | ● | |
| 9 | インドネシア | 現地技術者の指導・教育 | ● | ● | | |
| 10 | タイ | テスト業務改善 | | ● | | |
| 11 | マレーシア | 現地技術者の指導・教育 | | ● | | |
| 12 | 英国 (2件) | MISRA-C 改版 | ● | | | |
| 13 | | プログラム品質向上 | 組込み製品 | ● | | |
| 14 | スウェーデン | 海外関連会社向けコーディング規約作成 | ● | | | |
| 15 | ロシア | 現地技術者の指導・教育 | ● | ● | ● | |
| 16 | 韓国 | 現地技術者の指導・教育 | ● | | | |
| 17 | 台湾 | OEM 先の品質向上 | ● | | | |
| 18 | 米国 | コーディングルール適用 | ● | | | |
| 19 | 国内 (4件) | 海外ベンダへの品質説明の語彙 | | ● | | |
| 20 | | 海外子会社従業員向けの教育資料 | ソリューション・ビジネス | | ● | |
| 21 | | 海外関連会社向けコーディング規約作成 | 組込み製品 | ● | | |
| 22 | | 国際学会向け | 学会 | | | ● |
| (合計件数 22) | | (合計件数) | 12 | 13 | 3 | |

図 6 ESxR シリーズ英訳版利用状況

5.4 セミナー実績

年間を通じて ESxR シリーズの普及促進のために解説セミナーを実施している。今年度は以下に示す通り、大学・業界との連携セミナー、企業の導入事例セミナーおよび ESxR トレーナー養成セミナーを中心に実施した。

| 開催日 | セミナー、ワークショップ | 共催 | 開催場所 |
|------------|-----------------------------------|-------------------------------------|------|
| 5/9 ~ 11 | ESCR C++, ESGM 解説 セミナー | ESEC2012 | 東京都 |
| 5/24 | 慶応大学 SDM ESCR 解説 | 慶応大学 SDM | 東京都 |
| 5/31 | 慶応大学 SDM ESQR 解説 | 慶応大学 SDM | 東京都 |
| 6/7 | 慶応大学 SDM ESMR 解説 | 慶応大学 SDM | 東京都 |
| 6/14, 15 | ET-WEST ESGM 解説 | (一社) 組込み システム技術協会 (JASA) | 大阪府 |
| 7/12, 13 | ESPR トレーナーズ トレーニング (トライアル) | - | 東京都 |
| 8/24 | IEEE ワークショップ ESCR 解説 | IEEE SMC Hiroshima Chapter | 広島県 |
| 9/27, 28 | ESCR トレーナーズ トレーニング | - | 東京都 |
| 10/19 | ESMR/ESGM 解説セミナー | - | 東京都 |
| 10/25 | IPA フォーラム SEC セッション | - | 東京都 |
| 11/14 ~ 16 | ET2012 セミナー (ESDR, ESTR, ESBR) | (一社) 組込み システム技術協会 (JASA) | 神奈川県 |
| 12/13, 14 | ESMR トレーナーズ トレーニング (トライアル) | - | 東京都 |
| 1/29 | ESQR 事例付きセミナー | - | 広島県 |
| 2/22 | ESDR, ESTR 解説セミナー | - | 東京都 |
| 3/7, 8 | ESPR トレーナーズ トレーニング | - | 東京都 |
| 3/11, 12 | ESxR, ESDR, ESTR, ESBR 解説セミナー | (一社) 組込み システム技術協会 (JASA) 東北支部 | 宮城県 |
| 3/15 | ESBR 解説セミナー | - | 東京都 |

【参考文献】

[SEC journal 29] 三原、松田、濱田、十山、石井、他：組込みソフトウェアプロジェクトの状況 SEC journal No.29 Vol.8, No.2, pp.63-68, 201

SECにおける第二期中期計画の総括と第三期中期計画の概要について

～ソフトウェア開発の生産性向上からソフトウェア高信頼化へ向けて～

SEC 次長

杉原井 康男

SEC 企画グループ

田中 薫

1. はじめに

情報処理システム（情報システムと組込みシステムの総称）は、あらゆる経済活動や日常生活を支え、社会を効率化し、組織の生産性を高め、新たな付加価値や利便性を生み出す社会基盤となっている。近い将来には、高度な組込みシステムが搭載された自動車や製品機器が、ネットワークを介して、クラウド上に構築された交通渋滞や電力需給等をリアルタイムに監視する大規模情報システムに統合され、全体を最適状態に維持するスマートコミュニティが構想されている。このように、今後益々社会が情報処理システムへの依存を高める中で、SECが取り組む事業は抜本的に見直すことが求められている。本稿では、SECの第二期中期計画（2008～2012年度）の事業成果を総括し、これを踏まえて、第三期中期計画（2013～2017年度）で、SECがどのような現状認識の下で、新たな課題に取り組むのかを解説する。

2. 第二期中期計画（2008～2012年度）の総括

2.1 実施体制

SECは、2004年10月に発足し、第一期中期計画（2004年10月～2007年度）では、我が国の「ソフトウェア開発力の強化」を目指して、活動に着手した。2008年度より開始した第二期中期計画では、事業の方向性を「社会基盤としてのITの安全性・信頼性の向上」にも拡大し、「エンジニアリング手法によるITシステムの信頼性確保」、「地域・中小企業のためのシステム構築手法の提供」を柱として、事業活動を行った。また、実施体制の面では、IT融合化の流れの中で、情報処理システムの複雑化・大規模化に対応するために、SEC発足当初からの活動領域である「エンタプライズ系システム（情報シス

テム）」と「組込み系システム」に加えて、両者が融合し、一体となって機能する「統合系システム」にも着目し、2009年度に「統合系プロジェクト」を新設して、「エンタプライズ系」、「組込み系」、「統合系」の3つのプロジェクト体制で事業活動を実施した。第二期中期計画の期間中における主な事業成果は、以下のとおりである。

2.2 主な事業成果

(1) 「見える化」をはじめとするエンジニアリング手法によるITシステムの信頼性確保

① 重要インフラ分野の情報システムの信頼性向上

システム障害が、社会に大きな損失をもたらす可能性が高い、重要インフラ分野（情報通信、金融、鉄道等）を支える情報システムの信頼性向上に取り組むため、国内外のIT障害に関する事例調査を行い、2011年に信頼性向上に関する取組み方策をまとめた「重要インフラ情報システムの信頼性向上の取組みガイドブック」を公開した。また、自社の信頼性レベルを客観的に自己評価するための「情報システム信頼性自己診断ツール」を開発・提供し、ITサービスを提供する事業者の信頼性向上への取組みを支援した。

② 非機能要求の見える化

機能要求に比較して、システム運用時の可用性等の非機能要求は、受発注者間で十分な合意形成が行われないうまま、システム開発が進められることが多々見られる。二重化・バックアップ等の非機能要求に関する仕様の設計は、システム運用フェーズで信頼性に大きな影響をもたらすことから、受発注者間の非機能要求に対する共通理解を醸成するために、非機能要求の類型化と記述フォーマット、その効果の検証結果等をまとめた「非機能要求記述ガイド」を2008年に公開した。また、大手SI事業者6社が参加した「システム基盤の発注者要求を

見える化する非機能要求グレード検討会^{*1}から譲渡された「非機能要求グレード^{*2}」の活用事例等を踏まえて、内容を拡充し、2013年3月に「非機能要求グレードの研修教材と利用ガイド」として公開するとともに、セミナー・イベントでの普及活動を行い、非機能要求の見える化の重要性への認識を高めた。

③上流工程での高信頼化技術

システム開発における上流工程（システム要件定義、基本設計）での品質向上は、下流工程（コーディング・テスト）での品質改善や手戻り削減の効果が大きい。そこで数学的に厳密な記述をすることで、曖昧さが排除できる形式手法に着目し、先進的な適用事例の調査や、実際に運用中の情報システムの設計書に形式手法を適用し検証を行う実証実験を実施し、形式手法の適用効果を明らかにした。また、形式手法は難解で、導入のハードルが高いという先入観を払拭するために、経営者・実務者向けの入門教材や、日本語による厳密な仕様記述方法の解説書等を産学の専門家を結集して作成し、公開した。さらにセミナー・イベントでの普及活動を行い、形式手法の国内での再評価に結びつけた。

④ソフトウェア信頼性に関する第三者による説明力強化

経済産業省産業構造審議会の場合、「第三者検証によるソフトウェアの信頼性の見える化」の重要性が示された。^{*3} これを受けて、SECは2010年11月に「第三者検証検討部会」を設置し、第三者がソフトウェアの品質等を確認・評価し、利用者へ情報提供する仕組み（図1参照）の提言に向けた活動を開始した。2011年には「ソフトウェアの品質説明力強化のための制度フレームワークに関する提案（中間報告）」を公開し、公募による13件の模擬実験等を行った。さらにソフトウェアの安全性・信頼性を第三者が検証するための制度設計を行い、

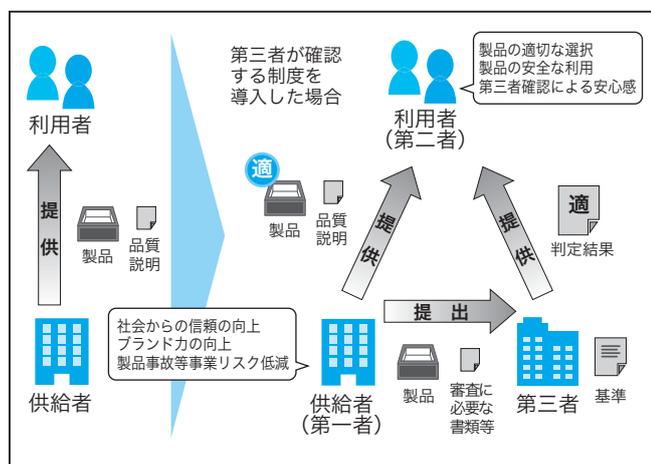


図1 制度の基本的な考え方

2013年3月には「製品・システムにおけるソフトウェアの信頼性・安全性等に関する品質説明力強化のための制度構築ガイドライン案」を公開した。

⑤組み込みシステムの信頼性向上

自動車・家電等の消費者向け製品や産業用途の制御装置等の高機能化とともに、大規模化・複雑化する組み込みシステムの品質向上・高信頼化を確保することが重要となっている。そこでSECは大手企業の開発現場の経験・ノウハウを収集・整理し、中小企業等の開発現場や組み込み技術者の教育に活用できるよう、組み込みシステム開発技術リファレンス「ESxRシリーズ」や手引書の整備を行った。具体的には、第一期中期計画の期間中に発刊したESCR^{*4}、ESPR^{*5}、ESMR^{*6}に加え、ソフトウェアの品質を定量的に測るための指標や事例についてまとめた「ESQR^{*7}」（2008年）、C++言語のコーディングルール等をまとめた「ESCR C++ [C++言語版]」（2010年）、プロジェクト計画立案の具体的な手順をまとめた「ESMG^{*8}」（2011年）、成功・失敗を含む様々な設計事例を基に設計ノウハウをまとめた「ESDR^{*9}」（2012年）を順次発刊した。また、日本企業のグローバル化に対応するニーズにも応えて、ESxRシリーズの英訳版を公開した。

⑥ソフトウェア開発プロセスの改善

組織的なソフトウェア品質向上の取組みとして、ソフトウェア開発のライフサイクル全体に渡って、プロセスを継続的に改善する活動が有効である。特に、人材等の資源が限られる中小企業や小規模組織では、導入コスト

【脚注】

- ※1 株式会社NTTデータ、富士通株式会社、日本電気株式会社、株式会社日立製作所、三菱電機インフォメーションシステムズ株式会社、沖電気工業株式会社の国内SI事業者6社により構成された、非機能要求の確認方法について共同検討を行うための会。2010年2月「非機能要求グレード」の完成に伴い解散。成果の普及活動をIPAに移管した。
- ※2 システム開発において受発注者間での認識の相違や誤解を防ぐため、非機能要求の確認を行うための手法。
<http://sec.ipa.go.jp/std/ent03-b.html>
- ※3 産業構造審議会情報経済分科会 情報サービス・ソフトウェア小委員会（第13回 2010年3月）
<http://www.meti.go.jp/committee/summary/0004664/index13.html>
- ※4 ESCR (Embedded System development Coding Reference)：組み込みソフトウェア開発向けコーディング作法ガイド[C言語版]
- ※5 ESPR (Embedded System development Process Reference)：組み込みソフトウェア向け開発プロセスガイド
- ※6 ESMR (Embedded System development Management Reference)：組み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]
- ※7 ESQR (Embedded System development Quality Reference)：組み込みソフトウェア開発向け品質作り込みガイド
- ※8 ESMG (Embedded Systems development Management planning training Guide)：組み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
- ※9 ESDR (Embedded System development Design Reference)：組み込みソフトウェア向け設計ガイド[事例編]

が低く、段階的に導入できるプロセス改善手法が求められる。そこで、SECが譲渡を受けたプロセス改善手法をベースにして、中小企業の開発現場での導入実験を繰り返し行い、無理なく導入できるプロセス改善手法と、社内の人材をプロセス改善推進者（準アセッサ）として育成するための教材を整備した。具体的には、全社的な取組みを想定しトップダウン型アプローチをする「ソフトウェアプロセスの供給者能力判定及びアセスメントキット IPA 版（SPEAK-IPA^{※10}）」と、プロジェクト単位又は開発現場での自主的な取組みを想定しボトムアップ型アプローチをする「SPINA³CH 自律改善メソッド^{※11}」の2つのプロセス改善手法（図2参照）を開発し、公開した。また、両手法を効果的に使い分けるための解説書「プロセス改善活用ガイド」と「プロセス改善推進者向け教材」を作成し、利用者が自社の環境に合わせて柔軟に導入できるよう、改変・再配布可能な使用条件^{※12}で公開した。さらに、受発注者間でソフトウェア開発プロセスの考え方を共有することを狙った、2007年発刊の「共通フレーム2007」、2009年発刊の「共通フレーム2007第2版」に続き、2013年3月にはその後の国際規格の改定と完全翻訳JISの発行を踏まえて、内容を大幅に拡充した「共通フレーム2013」を発刊した。

⑦ビジネス環境の変化への迅速な対応

ビジネス環境が目まぐるしく変化する中で、環境変化に柔軟に対応できる非ウォーターフォール型開発^{※13}への期待が高まっている。特に、ITサービスをWebアプリケーションとして実装する場合には、システムの枠組みがあらかじめ作り込まれている開発フレームワークを使い、顧客の利便性やニーズに焦点をあてて、小チームで開発を進めるアジャイル型開発が有効とされている。そこで、国内外のアジャイル型開発の適用事例、具体的

な手法（プラクティス）、開発現場で導入を阻害する要因等を調査し、「モデル契約書ひな型」、「非ウォーターフォール型開発の普及要因と適用領域の拡大に関する調査報告書」、「アジャイル型開発におけるプラクティス活用事例調査報告書」、「プラクティス活用リファレンスガイド」を作成して順次公開するとともに、セミナーやイベントでアジャイル型開発についての普及啓発を行った。

(2) 地域・中小企業のためのシステム構築手法の提供

①定量データに基づくプロジェクト管理の促進

大手開発ベンダが受注する大規模なソフトウェア開発では、正確な進捗管理、問題の早期発見等を行い、QCD達成度を向上させるために、定量データ（メトリクス）に基づくプロジェクト管理が普及し定着してきている。一方、人材等の資源に限られる中小企業では、ツールの導入コスト等の壁もあって、なかなか普及が進まない状況にあるため、地方の中小企業が定量データに基づくプロジェクト管理に取り組めるよう、環境整備を行った。具体的には、2007年には「Web版プロジェクト診断支援ツール」、2009年には「スタンドアロン型プロジェクト診断支援ツール」を公開した。そして2012年4月には、プロジェクト管理ツールとして広く普及しているRedmineとTracにプラグイン対応した「定量的プロジェクト管理ツール（EPM-X）」をオープンソースとして公開し、セミナー・イベントでの普及活動とも相俟って、導入企業が増えている。また、「ソフトウェア開発データ白書」は、大手企業27社が提供するプロジェクトデータを集計・分析して2005年から発刊してきたが、ソフトウェア開発のベンチマークとして定着し、広く活用されている。第二期中期計画の期間内においても、2008版、2009版、2010-2011版、2012-2013版と順調に版を重ねた。特に、直近の2012-2013版では、累計3,089件のプロジェクトデータを収録しており、プロジェクトデータ集として、世界有数の規模にまで成長した。なお、2007版と2008版の英訳版を公開したところ、国際標

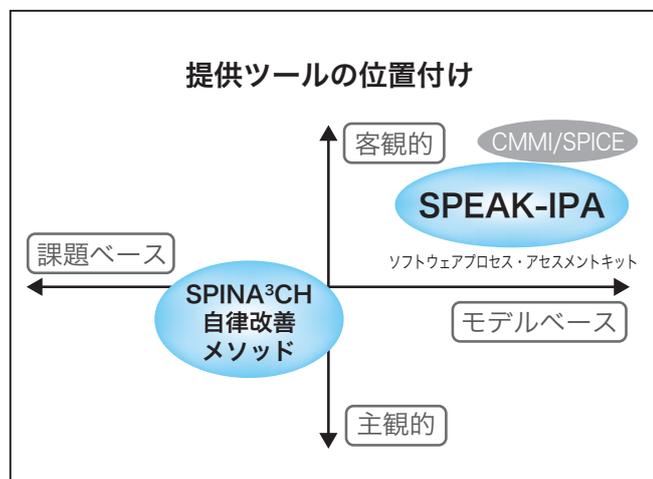


図2 SPEAK-IPA と SPINA³CH の関係

【脚注】

- ※10 SPEAK-IPA（Software Process Evaluation & Assessment Kit IPA）：モデルベースのプロセス改善を推進しようとしたときの活用を目的としたアセスメントモデル。
http://sec.ipa.go.jp/reports/20130326_2.html
- ※11 ソフトウェアの開発プロセスに問題意識を持つ技術者向けに、改善活動に役立つヒント等を盛り込んだツール類をとりまとめたプロセス改善の手法。
http://sec.ipa.go.jp/reports/20130326_3.html
- ※12 クリエイティブコモンズ・ライセンス表示-継承2.1日本ライセンス。
- ※13 アジャイル型開発等をはじめとする、ウォーターフォール型開発以外のソフトウェア開発手法の総称。
<http://sec.ipa.go.jp/std/ent02-c.html>

SECの成果・セミナー等の評価：調査票間比較（組込み系・ベンダ・ユーザ） 概要報告書

総合評価で見ると、組込み系、ベンダは5割程度、ユーザは3割程度が「役に立っている」以上の評価。中でも出版物が最も高く評価されており、組込み、ベンダでは8割弱、ユーザでは5割が「役に立っている」以上の回答となっている。また、後続の項目の評価順位も同様である。

■ たいへん役に立っている
■ 役に立っている
■ あまり役にたっていない
■ まったく役にたっていない
■ わからない

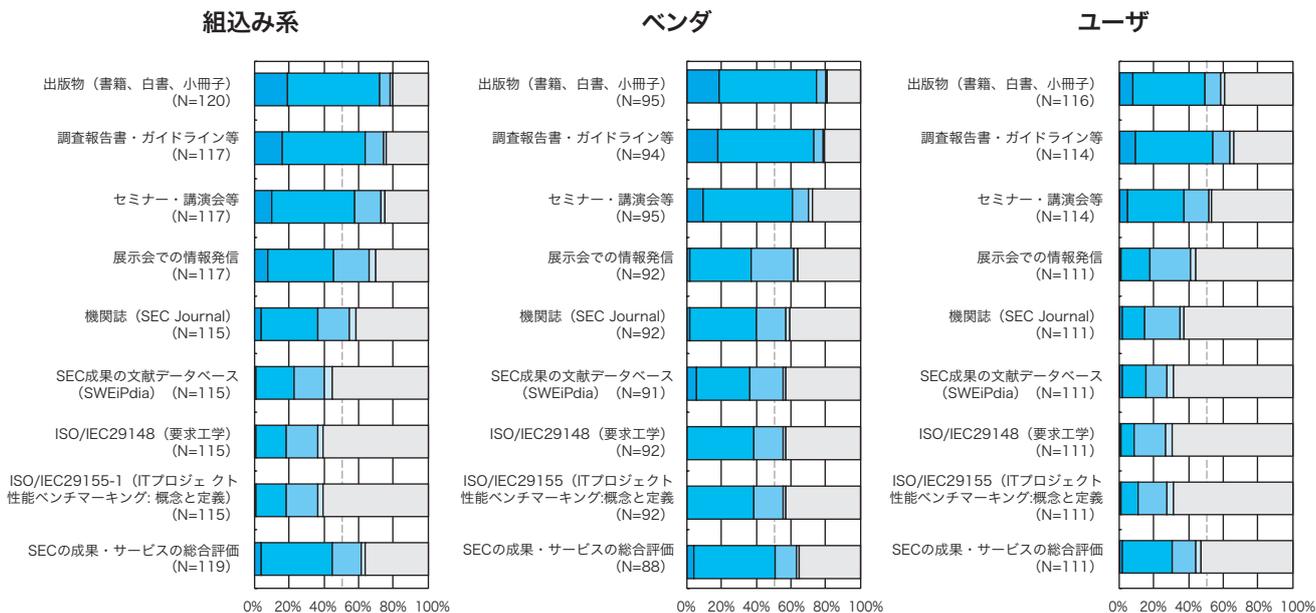


図3 SEC成果の評価

準 ISO/IEC29155 シリーズ^{*14} で参考文献として引用され、国際的な認知も得た。

②地域・中小企業における超上流工程の取組み強化

システム化対象の事業ニーズ把握から要件定義までの超上流工程において、発注者と受注者の間で認識を共通化することは、上流工程以後のシステム開発を成功に導く上で、きわめて重要である。超上流工程での受発注者間の意思疎通を図るために、2008年に「発注者ビューガイドライン^{*15}」を公開した。2010年には、これをユーザ（発注者）視点から強化し、超上流工程での不十分な合意形成が原因で発生するシステム開発での手戻りを防止するためのノウハウをとりまとめて、「機能要件の合意形成ガイド」として公開し、受発注者双方の合理的な役割分担・共通認識に基づく、システム開発のあるべき姿を示した。

2.3 SEC成果の普及状況

「2012年度ソフトウェア産業の実態把握に関する調査^{*16}」では、SEC成果がどれくらい役に立っているかを客観的に評価する設問を加えて、調査を実施した。図3に示すように、組込み系とエンタプライズ系（ベンダ）のソフトウェア開発事業者からの回答企業のほぼ半数が、「SECの成果・サービスの総合評価」として、「たいへん役に立っている」又は「役に立っている」と回答し、SEC成果に

一定の評価を与えている。特に、SECの特徴である部会・WG活動を通じて、産学の有識者の知見を結集してとりまとめた、「出版物」や「調査報告書・ガイドライン等」は、ユーザ企業も含めて回答企業の過半数が高く評価している。また、SEC成果の普及のために力を入れてきたセミナー・講演会等に対しても、約半数の回答企業から「役に立っている」との評価を得た。

3. 第三期中期計画（2013～2017年度）の背景と事業内容

3.1 情報処理システムを取り巻く現状認識と取り組むべき課題

(1) 現状認識

第三期中期計画の策定に当たり、情報処理システムを取り巻く現状認識を、以下の3点に集約した。

①企業活動・個人生活と情報処理システムの一体化

インターネットやクラウド技術等のITイノベーションを活用することにより、情報処理システムは複雑化・高度化しながら、企業活動の生産性や個人生活の利便性

【脚注】

- *14 ITプロジェクト性能ベンチマーキングの国際規格。
- *15 わかりやすい外部設計書を作成する上でのノウハウをまとめたもの。<http://sec.ipa.go.jp/reports/20080710.html>
- *16 IPA「2012年度ソフトウェア産業の実態把握に関する調査」<http://sec.ipa.go.jp/reports/20130426.html>

を飛躍的に高めている。企業活動では、受発注、製品開発、生産管理、在庫管理、販売管理、顧客管理等の個々のビジネスプロセスが、ITにより自動化の範囲を広げるとともに、企業の生産性向上の壁になっていた部門間連携を、情報システムの活用により、低コストで実現できるようになった。「情報システム基盤の復旧に関する対策の調査報告書（SEC、2012年7月）」によると、アンケート調査の回答企業352社のうち、約7割が「ITに依存している事業・業務が多い」と回答し、そのうち約4割は「ほとんどの事業・業務がITに大きく依存している」と回答している。このように、企業活動の情報処理システムへの依存度は予想以上に高くなっている。また、ここ数年で、個人生活も急速に情報処理システムへの依存度を高めており、「平成23年通信利用動向調査（総務省、2012年5月）」によると、我が国では過半数の世帯が、自宅からパソコン等でインターネットに接続している。また、スマートフォンの世帯保有率は、2010年の9.7%から2011年には29.3%へと短期間で急増し、スマートフォン利用者の約8割は「毎日少なくとも1回」は、家庭外でインターネットにアクセスし、SNS等を使った情報収集・共有・発信やオンラインショッピング等を利用している。

②ソフトウェアが支える社会基盤の拡大

どのような複雑・大規模な情報処理システムであっても、単純化すれば、サーバ・モバイル端末・制御機器等の「ハードウェア」と、インターネット等の「情報通信ネットワーク」と、ミドルウェア・アプリケーション等の「ソフトウェア」の3つの要素から構成されている。マイクロプロセッサの省電力化・性能向上に代表されるハードウェアの技術革新や、この十数年で劇的に普及したインターネットやモバイル通信に代表される情報通信ネットワーク基盤は、情報処理システムを構築する上での与件とみなすべきものである。一方、ソフトウェアは、情報処理システムが提供する機能・サービスを実現するための中核部分であり、ミドルウェアの選択やアプリケーションの作り込み（ソフトウェア開発）の仕様次第で、無限のバリエーションが存在する。多様な情報処理システムに依存する社会基盤は、目に見えないソフトウェアに支えられており、大規模化・複雑化するソフトウェアの高信頼化は、これまで以上に、きわめて重要な課題となっている。

③システム障害時の影響範囲の拡大と深刻度の増大

重要インフラ分野の社会基盤が情報処理システムへの

依存を高めるにつれて、機器故障、ソフトウェア不具合、人為的ミスが複合的に絡み合って発生するシステム障害は予想以上に影響範囲を拡大し、深刻な社会損失をもたらす事態が目立っている。2012年に限って見ても、重要インフラ分野の代表である金融分野で、2012年2月（株式取引の一時停止）と2012年8月（デリバティブ取引の一時停止）に発生した東京証券取引所のシステム障害は、情報処理システムに依存する社会にあらためて警鐘を鳴らすとともに、その後に公表された原因分析・再発防止策は、分野・業種を越えて、貴重な教訓をもたらしている。また、この数年で急速に普及しているクラウドサービス分野で、2012年6月に発生したファーストサーバ社のシステム障害は、保守作業中の人為的ミスにより、約5,700社（契約ベース）から預かったデータが消失し、顧客である企業・団体等の事業活動に深刻な支障をもたらした。

(2) 取り組むべき課題

上記の現状認識を踏まえて、第三期中期計画でSECが取り組むべき課題を、以下の2点に集約した。

①システム障害などに起因する社会損失の最小化

大規模なシステム障害であっても、その端緒となった原因は単純な事象であることが多い。しかしながら、機器故障、ソフトウェア不具合、人為的ミスが複合的に重なることによって、結果的に大事に至るケースが見られる。また、システム運用・保守の現場で日常的に起きている「ヒヤリハット」事象に適切に対処していれば、システム障害が未然に防げたのではないかと思われるケースも見られる。そこで、中立の公的機関として、重要インフラ分野のシステム障害情報や生命・財産に危害をもたらす可能性があるソフトウェア不具合に起因する製品機器・制御装置等の障害情報を収集・分析し、そこから得られた再発防止対策・教訓等を公共財（ガイドライン、事例集等）として公開し、企業・業種の壁を越えて情報を共有する仕組みを構築する。さらに、ガイドラインによる提言等を通じて、個々の事業者等に社内のシステム管理体制や運用保守プロセスを継続的に改善することを働きかけることにより、社会全体のシステム障害の発生頻度を抑え、システム障害に伴う社会損失を最小化する。

②利用者が安心してソフトウェアを利用できる社会の実現

目に見えないところで社会基盤を支えているソフトウェアは、数十万～数百万行の設計情報（ソースコード）から作られているが、これまでの経験・知見では不具合

(バグ)を完全に無くすことは不可能とされている。一方、ソフトウェア工学の研究成果等から、上流工程での設計品質を高めることにより、後工程での不具合(バグ)の発生を少なくできることが示されている。近い将来の実現が構想されているスマートコミュニティでは、家庭や施設に設置されたエネルギー管理システムが、系統電源、分散電源、製品機器等を制御して、エネルギー消費効率を高めるとともに、クラウド上の大規模情報システムに統合されて、社会全体の最適制御も行なうようになる。このような大規模かつ複雑なシステムの開発では、個々の製品機器・部品に組み込まれるソフトウェアの高信頼化だけでは不十分であり、相互に接続された複雑なシステム全体を、SoS (System of Systems) として俯瞰できるようにモデル化した上で設計・開発を行う必要がある。また、開発したシステムは第三者がソフトウェアの信頼性と相互関連の安全性を検証し、ソフトウェア品質を利用者に見える化した上で提供する仕組みを構築することにより、ソフトウェアに支えられた社会基盤を国民が安心して利用できる社会を実現する。

3.2 第三期中期計画の事業内容

3.2.1 重要インフラ分野の情報処理システム等に係る障害情報の収集・分析及び対策

(1) 重要インフラシステム等のソフトウェア障害情報の収集・分析

①重要インフラ分野における情報処理システムについて、品質・信頼性確保に関する実証的なデータの収集を継続するとともに、機器故障・ソフトウェア不具合・人為的ミス等により運用時に障害が起きた場合にも、システム全体の停止に波及させることなく、国民生活や経済活動への影響を極小化するIT社会を実現するために、業種を越えて、運用・利用面での障害事例(障害現象・被害状況・原因・復旧対策・再発防止策等)を社会で共有する仕組みを構築する。

②国民生活や経済活動に一定以上の影響を及ぼした障害については、事業者が積極的に情報提供を行えるよう、障害情報を記録する共通様式の設計、機密保持・情報提供の方法のルール化等を推進する。

(2) 重要インフラシステム等のソフトウェア障害の再発防止の導入促進や事例に対する対策支援

①障害の再発防止に向けたシステム開発や運用・管

理の継続的なプロセス評価・改善手法を策定し、ITサービス提供者への導入を促進する。

②障害事例の分析に基づき、ITサービスの利用者の意識改革を促すため、サービスの安全性、信頼性、グレードとコストに関する複数の選択肢を提供する方法等について検討を進める。

3.2.2 利用者視点でのソフトウェア信頼性の見える化の促進

(1) ソフトウェア品質説明力の強化の促進

①製品・サービスを提供する事業者が利用者に対しソフトウェア品質を十分に説明できるよう、ソフトウェアの信頼性に関する表示を行う仕組みを構築し、BtoBを中心とするサプライチェーンにおけるソフトウェア品質のトレーサビリティの確保を図る。

②ソフトウェアの信頼性について、第三者が確認を行うソフトウェア品質説明力強化の取組みを促進する。

(2) 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、その信頼性を確保するため、上流工程での先進的な設計方法の効果的な適用事例を収集し、適用のためのガイドライン等を策定する。

②ソフトウェアの信頼性検証のための先進技術の適用促進、信頼性検証技術の活用手法の提供を行う。

4. むすび

これまで、SECは「ソフトウェア開発の生産性向上」を主たる事業目標として、開発現場の実務者や産学の有識者の経験・知識を結集し、書籍・調査報告書等の実務に役立つ成果を公開し、一定の評価を得てきた。今年度からスタートした第三期中期計画では、「社会全体を支える情報処理システムの信頼性向上」を事業目標として掲げ、組織体制も2013年6月1日付で『ソフトウェア高信頼化センター(SEC: Software Reliability Enhancement Center)』に改組し、中立的な公的機関としての特性を活かして、情報収集の仕組みや産学官の連携関係等を構築しながら、情報化社会の安寧と健全な発展に貢献する事業活動に取り組んでいる。

FP 計測手法における FP 規模と工数の相関の差

東海大学理学部
IPA/SEC 専門委員

古山 恒夫



IPA/SEC で収集したエンタプライズ系ソフトウェアプロジェクトデータから、FP 計測手法ごとに FP 規模と実績工数の相関分析を行った。新規開発プロジェクトでは IFPUG 法の相関が最も高く、簡易な FP 計測手法である SPR 法や NESMA 概算法よりも決定係数でそれぞれ 0.169 と 0.250 大きな値となった。同じ IFPUG 法でも製造業のプロジェクト群の方が金融・保険業よりも相関が高い傾向が見られた。

1. はじめに

FP 計測手法として、IFPUG 法、COSMIC 法、フルファンクションポイント法、フィーチャポイント法、Mark II 法、NESMA 概算法、SPR 法などさまざまな分野のソフトウェアに対してさまざまな計測手法が提案されている [富野 2010]。このうち、エンタプライズ系ソフトウェアに対しては、主に IFPUG (International Function Point User Group) が提案した IFPUG 法が使われているようである。IFPUG 法は FP のカウント方法が詳細に定義されているため、FP 計測の精度は向上するが、逆に設計仕様書が明確にならないと正確な計測ができないという問題がある。また FP 計測の工数もかかる。これらの問題に対処するために、計算を簡略化することにより、開発の早期に比較的少ない工数で FP を求めることができるようにした NESMA 概算法や SPR 法も使われている。ちなみに IPA/SEC で収集したエンタプライズ系ソフトウェアプロジェクトデータにおける FP 計測手法の分布を調べてみると、IFPUG 法が 32%、SPR 法が 21%、NESMA 概算法が 10%、その他の FP 計測手法が 33% となっている (FP 計測手法の記載のないプロジェクトが 4% ある)。

本稿ではエンタプライズ系ソフトウェアプロジェクトで使われている、これら 4 つの FP 計測手法における FP 規模と工数の相関分析を行った結果を報告する。この相関係数の値が大きいということは、FP 計測手法による工数予測の精度が高いということになる。その差を比較することにより、早期見積りにおける見積もり損失の大ま

かな目安を得ることができると考えられる。

2 章で 4 つの FP 計測手法の概要を、3 章で分析結果を示す。3 章では各 FP 計測手法の適用プロジェクトの大きさの比較、相関分析の結果、より詳細な層別による分析結果を示す。4 章で分析結果に対する考察を、5 章でまとめを述べる。

2. FP 計測手法の概要

2.1 IFPUG 法

IFPUG 法では次の手順で機能量 (FP) を計算する [JISX0142], [富野 2010]。

- ① 扱うデータを外部入力 (EI)、外部出力 (EO)、外部照会 (EQ)、内部論理ファイル (ILF)、外部インタフェースファイル (EIF) の 5 つのファンクション型 (以下ではタイプと略称) に分ける。
- ② 扱うデータごとに、「データ項目数」とそのデータに関連する「レコード種類数」を求め、それによってそのデータのファンクションの複雑さを 3 段階 (低、中、高) に分ける。3 段階の分け方は、同じ「データ項目数」及び「レコード種類数」であってもデータのタイプによって異なる。
- ③ 各データに 3 段階のファンクションの複雑さに応じた未調整 FP (重み係数) を掛けて合計し、システム全体の未調整 FP を求める。ファンクションの複雑さのレベルが同じでも、データのタイプによって未調整 FP の値は異なる。
- ④ ①～③までの計算と独立に、対象とするシステムの特性を 14 の観点から 0～5 の 6 段階評価し、合計

する。

⑤ システム特性係数 = 0.65 + ④の合計値 × 0.01 を計算する。

⑥ FP = システム特性係数 × 未調整 FP

JIS 規格として制定されているのは①から③までである。

2.2 NESMA 概算法

NESMA 概算法はオランダの NESMA (Netherlands Software Metrics Association) が提案した手法であり、開発初期段階でも利用できる [NESMA]。NESMA 概算法では、IFPUG 法における②と③を一本化し、各タイプのデータに与える複雑さは 3 段階ではなく、データのタイプごとに固定したものとする。すなわち、外部入力、外部出力、外部照会の 3 つのタイプのデータに対しては、常に「中」の複雑さを与え、内部論理ファイル、外部インターフェースファイルの 2 つタイプのデータに対しては常に「低」の複雑さを与える。これによって、扱うデータの詳細がわからない段階でも未調整 FP を計算することができるようになる。

2.3 SPR 法

SPR 法は SPR 社がファンクションポイントの計算工数削減をするために計算を単純化したものである [富野 2010]。SPR 法では IFPUG 法における②と③を一本化し、データのタイプによらず複雑さは常に「中」とする。

また、システム特性に基づく調整係数の算出は、IFPUG 法の④とは異なり、問題の複雑さ、データの複雑さの 2 つの視点から 5 段階評価し、それらの合計値に応じた調整係数を求める。

2.4 「その他」の FP 計測手法

エンタプライズ系のプロジェクトデータには、FP 計測手法の選択肢として「その他」というものがある。「その他」は主に IFPUG 法をベースに各社が独自に開発した FP 計測手法である。これを本文中では「その他」とカギ括弧付きで表す。

3. 分析対象プロジェクト

3.1 分析対象プロジェクト

分析は、IPA/SEC で収集したエンタプライズ系ソフトウェアプロジェクト 3,089 件のデータ [IPA2012] のうち、次の条件を満たすプロジェクトを対象として分析を行った。

① 開発 5 工程が揃っている。

② FP の実測値 (5001_FP 実測値_調整前) がある。

③ FP 計測手法が明記されている。

④ 層のデータ数が 30 件以上である。

なお、工数のデータはすべてのプロジェクトから報告されている。

3.2 層別プロジェクト件数

開発種別及び FP 計測種別ごとのプロジェクト件数を表 1 に示す。各層とも散布図の目視により若干の外れ値を除いている。なお、以下の表中では、新規開発を「新規」、改修・保守を「改修」と略記する。

改修・保守の NESMA 概算のデータ数が 10 であることを除いて、新規開発及び改修・保守ではすべての層でデータ数が 30 以上であった。したがって、分析はこの 7 つの層を対象として行う。再開発及び拡張を FP 計測種別で層別すると各層のデータ数はすべて 30 件未満となったので、今回は分析対象とはしていない。

表 1 層別プロジェクト件数

| 開発種別 | FP 計測種別 | | | | 合計 |
|------|------------|-----------|-------|------------|------------|
| | IFPUG | SPR | NESMA | その他 | |
| 新規 | 200 (4) | 77 (2) | 68 | 169 (1) | 514 (7) |
| 改修 | 52 (1) | 34 | 10 | 75 | 171 (1) |

(注) () 内は外れ値として除いたもの (別掲)

3.3 変数

FP 規模、工数はそれぞれ次のものを用いる。

- ・ FP 規模 (実績値) : 5001_FP 実測値_調整前
- ・ 工数 (実績値) : 開発 5 工程工数

ただし、分析に際しては上記のデータをすべて常用対数で対数変換したのを用いる。

4. 分析結果

4.1 有意水準

一般に、P 値が 5% あるいは 1% 以下である場合に「(有意水準 5% あるいは有意水準 1% で) 統計的に有意である」という。しかし、プロジェクトデータを分析した結果、P 値が 5% 以下になるようなケースはそれほど多くはない。また、P 値が 5% を超えるような結果であっても、ソフトウェアプロジェクト管理では有用と思われる情報もあると考えられる。そこで、本稿では、P 値が 20% 以下の場合、すなわち 5 回のうち 1 回は誤った情報を与える可能性がある場合も、分析結果は有用な情報であると考えて、「・・・の傾向がある」という表現で表すことにする。

4.2 適用プロジェクトの大きさの比較

3.2で述べた7つの層のプロジェクトの大きさを比較するために、FP規模・工数の各ランクをまとめたものを表2に示す。「ランク」で同じ記号のものは、t検定(有意水準5%)の結果各層の平均値同士で有意差がないことを表している。例えば、表2の「新規・IFPUG法」と「新規・NESMA概算法」は同じ「A」であるので、「新規・IFPUG法」の平均値(常用対数変換後の値で3.048であった)が「新規・NESMA概算法」の平均値(2.964であった)よりも大きいけれど各母集団のFP規模の平均値に統計的に有意な差があるとは認められない、ということを表している。

表2 プロジェクト規模の比較

| 開発種別 | FP計測手法 | データ数 | ランク | |
|------|--------|------|------|----|
| | | | FP規模 | 工数 |
| 新規 | IFPUG | 200 | A | A |
| 新規 | NESMA | 68 | A | A |
| 改修 | IFPUG | 52 | B | B |
| 新規 | その他 | 169 | B | C |
| 新規 | SPR | 77 | B | D |
| 改修 | その他 | 75 | C | D |
| 改修 | SPR | 34 | C | E |

表2から次のことがわかる。

- (1) FP規模と工数の大きさの順序は変わらない。
- (2) 新規開発・IFPUG法と新規開発・NESMA概算法が最も大きなプロジェクト群に適用されている。2つのグループの間にはFP規模・工数のいずれにも差が見られない。
- (3) 改修・保守・IFPUG法、新規開発・「その他」、新規開発・SPR法の順で適用プロジェクト群の大きさが小さくなる。
- (4) 改修・保守の「その他」と改修・保守のSPR法は比較的小さいプロジェクトに適用されている。

4.3 相関分析

各層のFP規模と工数の相関係数を表3に示す。相関係数をz変換すると正規分布に近似できるので、z変換後の値に対して相互にt検定を行って有意差の有無を調べ、ランクを求めた。ランクは新規開発・IFPUG法だけがAで他の層はすべて同一のBである。すなわち、新規開発・IFPUG法の相関だけが他の層と統計的に有意なほど高く、それ以外の層同士では統計的に有意な差があるとは言えないことを示している。従って、例えば改修・保守・IFPUG法の相関係数0.807と新規開発・NESMA概算法の相関係数0.735の間に統計的な有意差はない。

表3 相関係数と決定係数

| 開発種別 | FP計測手法 | データ数 | ランク | 相関係数 | 決定係数(*1) |
|------|--------|------|-----|-------|----------|
| 新規 | IFPUG | 200 | A | 0.886 | 0.784 |
| 改修 | IFPUG | 52 | B | 0.807 | 0.645 |
| 新規 | その他 | 169 | B | 0.800 | 0.638 |
| 新規 | SPR | 77 | B | 0.788 | 0.615 |
| 改修 | その他 | 75 | B | 0.757 | 0.568 |
| 改修 | SPR | 34 | B | 0.752 | 0.553 |
| 新規 | NESMA | 68 | B | 0.735 | 0.534 |

(*1) 自由度調整済のもの

表3から次のことがわかる。

- (1) 新規開発では、IFPUG法の決定係数(自由度調整済)が0.784に対し、NESMA概算法では0.534、SPR法では0.615となっている。すなわちIFPUG法ではFP規模だけで工数の78.4%が決定されるのに対し、NESMA概算法では53.4%、SPR法では61.5%であり、その差はそれぞれ25%と17%ある。これが早期にFP規模を見積もるために失うものと考えられる。
- (2) 同じIFPUG法でも、改修・保守では、新規開発の場合に比べて決定係数は0.139(14%)小さい。

4.4 業種別相関分析

4.2で述べた結果をより詳細に分析するために、開発5工程のデータを業種別に分けて、FP計測種別ごとのFP規模と工数の相関分析を行った。

FP計測種別ごとに業種別のデータ数を調べたところ30件以上データの揃っている層は「製造業」と「金融・保険業」しかなかった。したがって、業種としてはこの2つのみを分析対象とした。

新規開発の業種別の相関係数及び相関係数間の差の有無のP値を表4に示す。表4の「全業種」に示されている各FP計測手法の値は表3からの再掲である。

表4から同一のFP計測手法でも業種間で相関係数に差があることがわかる。すなわち、IFPUG法では、製造業での相関係数が金融・保険業での相関係数よりも大きい傾向にあり(t検定のP値0.168)、「その他」では製造業の相関係数は金融・保険業の相関係数よりも大きい(t検定のP値0.001)。

IFPUG法と「その他」の相関係数は同一業種内でみても差がある。製造業ではその差のP値は0.159であり、IFPUG法の方が「その他」よりも相関係数が高い傾向があると言える。金融・保険業ではその差のP値は0.001であり、有意水準1%でIFPUG法の方が「その他」よりも相関係数が高いと言える。

表4 業種で層別した相関係数とその差の検定

| 業種 FP 計測手法 | 全業種 (参考) | 製造業 | 金融・ 保険業 | P 値 (業種間) |
|--------------------|----------------|---------------|---------------|--------------|
| IFPUG | 0.886 (200) | 0.930 (34) | 0.873 (56) | 0.168 |
| その他 | 0.800 (169) | 0.874 (68) | 0.565 (41) | 0.001 |
| P 値 (FP 計 測手法間) | 0.004 | 0.159 | 0.001 | |

(注) () 内はデータ数

4.5 アーキテクチャ別分析

開発 5 工程のデータをアーキテクチャ別に分けて、FP 計測種別ごとの FP 規模と工数の相関分析を行った。

FP 計測種別ごとにアーキテクチャ別のデータ数を調べたところ 30 件以上データの揃っている層は、新規開発では 2 階層クライアント/サーバ (以下では、2 階層 C / S と略記)、3 階層クライアント/サーバ (3 階層 C / S と略記)、イントラネット/インターネット (ネット系と略記) の 3 つだけであった。

新規開発のアーキテクチャ別の相関係数及び相関係数間の差の有無の P 値を表 5 に示す。表 5 の「全アーキテクチャ」に示されている相関係数とデータ数は表 3 からの再掲である。

表 5 の相関係数を数値だけで比較すると、IFPUG 法では、ネット系 > 3 階層 C / S であり、「その他」ではネット系 > 3 階層 C / S > 2 階層 C / S であるが、t 検定の結果からは、同一の FP 計測手法ではアーキテクチャ間に差あるとは言えない。

同一アーキテクチャ内で FP 計測手法間の差をみると、ネット系における IFPUG 法の相関係数が SPR 法よりも大きい傾向にある (P 値は 0.091) ことがわかる。

表5 アーキテクチャ別で層別した相関係数とその差の検定

| アーキ テク チャ FP 計測手法 | 全アーキ テクチャ (参考) | 2 階層 C/S (A) | 3 階層 C/S (B) | ネット系 (C) | P 値 (アーキ テク チャ間) |
|-------------------------------|---|-----------------|-----------------|--|---------------------------|
| IFPUG (a) | 0.886 (200) | - | 0.859 (37) | 0.905 (128) | >0.2 |
| SPR (b) | 0.788 (77) | - | - | 0.824 (36) | - |
| その他 (c) | 0.800 (169) | 0.797 (55) | 0.827 (70) | 0.873 (32) | >0.2 (*1) |
| P 値 (FP 計測 手法間) | (a) と (b) は 0.013 (b) と (c) は >0.2 | - | >0.2 | (a) と (b) は 0.0091 他の組合 せは >0.2 | |

(注) () 内はデータ数

(*1) (A), (B), (C) の任意の 2 つの組合せに対する結果がすべて >0.2 である。

5. 考察

(1) FP 計測手法による相関係数の違いについて

NESMA 概算法や SPR 法では IFPUG 法で行う各データ種別の複雑さの評価を省略し、評価をデータ種別ごとに固定している (図 1)。それにより、早期に FP を求めることができるとともに、計測工数を削減することもできる。しかし、FP の精度が低下するというデメリットもある。

今回の分析では、新規開発における NESMA 概算法の FP 規模の工数に対する決定係数が新規開発の IFPUG 法よりも 25% 小さいという結果が得られた。これは予想以上に大きな数値であり、その理由が単にデータの複雑さの計算を簡略化しただけなのか、それ以外に大きな要因が隠されているのかは、今後さらに検討していく必要がある。適用プロジェクト群の規模の大きさに差がないことから、少なくともプロジェクト規模が影響を及ぼしているとは言えない。

「その他」は IFPUG 法よりも FP 規模と工数の相関が低い。独自仕様という共通性の減少がばらつきを大きくしているのかもしれない。

(2) 製造業が金融・保険業よりも相関係数が高い傾向にあることについて

未調整 FP はシステム特性を考慮する前の FP 値である。未調整 FP と工数との相関が高いということは、システム特性のバリエーションが小さいということを表していると考えられる。つまり、製造業の方が一般に金融・保険業よりもシステムのバリエーションが小さく、工数を予測しやすいものと思われる。

(3) アーキテクチャに依存しないことについて

2 階層 C / S、3 階層 C / S、ネット系では工数との相関に差が見られなかった。このことはこれらのアーキテクチャの間のシステムバリエーションの差がないと考えることができる。ただし、今回統計的に有意とはならなかったものの、2 階層 C / S、3 階層 C / S、ネット系の順で相関係数は大きくなり、この順でバリエーションがわずかながら少なくなっているのかもしれない。今後のデータ蓄積が待たれる。

(4) IFPUG 法を適用したプロジェクトと NESMA 概算法を適用したプロジェクトの特性の違い

今回の分析では、相関係数は適用業種 (製造業と金融・保険業) によって差が見られ、適用アーキテクチャによっては差が見られないことがわかった。IFPUG 法と NESMA 概算法で適用業種に違いがあれば、相関係数にもその影響を考慮する必要がある。

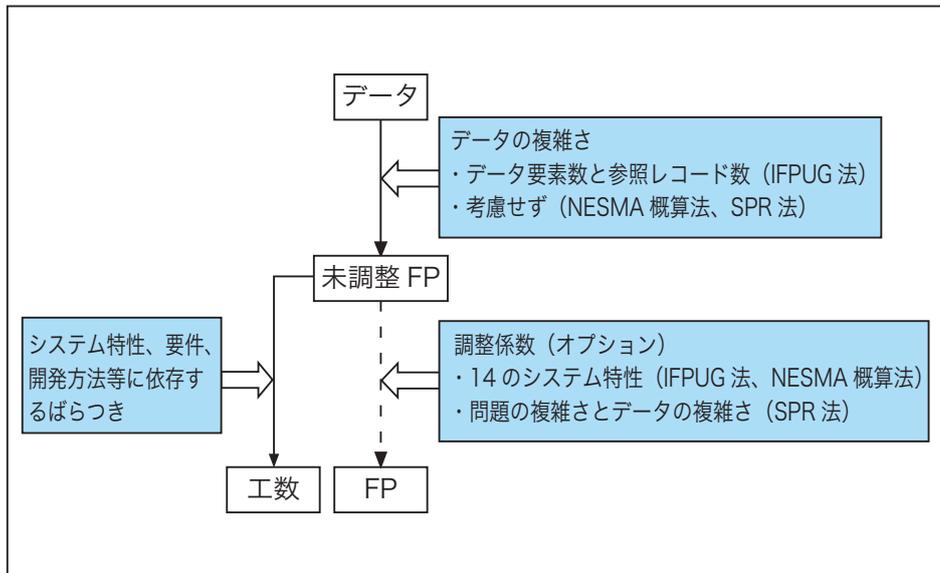


図1 FPの計算方法の概要と工数との関係

2つのFP計測手法と適用業種のクロス集計表を表6に示す。表6からIFPUG法を適用したプロジェクト群とNESMA概算法を適用したプロジェクト群の間で製造業と金融・保険業の差があるかどうかを χ^2 検定で調べたがP値は0.21となり、有意とはならなかった。すなわち、IFPUG法とNESMA概算法の差に業種の影響はないと言える。

表6 適用業種別のクロス集計表

| FP計測手法 \ 業種 | 製造業 | 金融・保険業 | 合計 |
|-------------|-----|--------|-----|
| IFPUG法 | 34 | 56 | 90 |
| NESMA概算 | 10 | 28 | 38 |
| 合計 | 44 | 84 | 128 |

6. まとめ

(1) 層別プロジェクト件数とプロジェクトの大きさ

- 開発5工程のプロジェクトを開発種別及びFP計測種別で層別した場合、30件以上のデータが揃っているのは、新規開発と改修・保守の各4つのFP計測手法の合計8つの層から、改修・保守のNESMA概算法の層を除いた7つの層だけである。
- FP規模と工数それぞれで各層のプロジェクトの大きさを比較すると、FP規模と工数で大きさの順序は変わらない。
- 新規開発・IFPUG法と新規開発・NESMA概算法が適用されるのは最も大きなプロジェクト群で、2つのグループの間にはFP規模・工数のいずれにも有意な差は見られない。

(2) 相関分析

- 新規開発では、IFPUG法とNESMA概算法で適用するプロジェクトの大きさ及び適用業種に差がないにも拘らず、相関係数はIFPUG法の方が大きい。決定係数を比較すると、IFPUG法では0.784である（工数の78%がFP規模で決定される）のに対してNESMA概算法では0.534でその差は0.25（25%）ある。
- 同じIFPUG法を適用しても、改修・保守では新規開発に比べて決定係数が0.139（14%）小さい。

- 新規開発のSPR法は新規開発のIFPUG法よりも決定係数が0.169（17%）小さい。

(3) 業種別・アーキテクチャ別分析

- 製造業と金融・保険業の相関係数を比較すると、新規開発・IFPUG法では、製造業が金融・保険業よりも大きい傾向が見られ、「その他」では、製造業が金融・保険業よりも大きい。
- アーキテクチャ間では相関係数に有意な差は見られない。

今回の分析方法は、FP規模と工数の相関係数及び決定係数から求めたもので、同じソフトウェアを2つの方法で計測して比較したものではない。今後は相関係数以外の指標についても調査し、それらの間の関係や既存の報告などとの整合性を検討していきたい。

今回はデータ数が30以上のものに限定して相関分析を行ったが、業種別・アーキテクチャ別に層別した場合、データ数が25～29までの層が4つ、20～24の層が6つある。今回の分析ではこれらの層を分析対象外としたが、これらの層のデータ数が30に達するとさらに精度の高い分析が可能となると思われる。

【参考文献】

- [IPA2012] 独立行政法人情報処理推進機構 (IPA) ソフトウェアエンジニアリングセンター (SEC) 監修：ソフトウェア開発データ白書 2012-2013, 2012年。
- [JISX0142] JISX0142 ソフトウェア技術—機能規模測定—IFPUG機能規模測定手法 (IFPUG 4.1版未調整ファンクションポイント) 計測マニュアル, 2010年。
- [富野 2010] C. Jones 著、富野 壽、小坂恭一監訳：ソフトウェア開発の定量化手法 (第3版)、共立出版、2010年。
- [NESMA] Early Function Point Counting 開発初期段階でのFPカウント方法 (<http://www.nesma.nl/sectie/home/>)

Java プログラムのアクセス修飾子過剰性分析ツール ModiChecker の機能拡張とその応用例

小堀 一雄[†], 石居 達也[†], 松下 誠[†], 井上 克郎[†]

我々は、Java プログラムの各フィールドやメソッドのアクセス修飾子の宣言とその利用状況との差をアクセス過剰性(AE:Accessibility Excessiveness)と呼び、AEを静的解析するツール ModiChecker を開発してきた。本研究では、検出した AE を容易に解消できるようにするために、AE に対して、開発者の指示に基づいて容易に修正できる機能を ModiChecker へ実装した。さらに応用例として、OSS である Ant の複数のバージョン間における AE の数の変化を調べ、AE の変化量とバージョンアップの種類との関連について調査した。

Correction Support and Version Analysis of Accessibility Excessiveness for Java Programs

Kazuo Kobori[†], Tatsuya Ishizue[†], Makoto Matsushita[†], Katsuro Inoue[†]

Abstract:

We have developed a tool named “ModiChecker”, which statically analyzes the declaration and usage of the fields and methods in Java programs, and reports excessiveness of the declaration as “Accessibility Excessiveness (AE)”. In this paper, we will add a feature to ModiChecker to support developers resolving AE declarations. Also, we will explore the relation between AE and program evolution, by analyzing multiple versions of an OSS program, Ant.

1. はじめに

現在のソフトウェア開発においては、要件の複雑化などに伴い、複数の開発者がチームを組んで設計、プログラミング、テストを実施することが多い。チームに所属する開発者は全員、ソースコードの仕様を完全に把握していることが望ましいが、コストや期間の関係上難しいこともある。その場合、ソースコード上のフィールドやメソッドに対して設計時には意図していない不適切なアクセスの仕方をプログラミング時に行ってしまう可能性がある。

こういった問題を防ぐために、アクセス修飾子を適切に設定することで、意図しないフィールドやメソッドへのアクセスを防止することができる [1][2]。しかし、全

てのフィールドおよびメソッドに関する適切なアクセス範囲を把握するのはコストがかかるため、何らかの支援が必要であると考えた。

我々は、アクセス修飾子過剰性検出ツール ModiChecker を開発した [6]。ModiChecker は、ソースコード群に対して、アクセス修飾子の宣言とフィールドとメソッドの被参照状況を静的解析することにより、過剰に広い範囲に設定されている可能性のあるアクセス修飾子を抽出する。これにより、開発者は意図しないフィールドやメソッドへのアクセスを事前に防止できる。

しかしこの既存研究では、どこからも参照されていないフィールドやメソッドを分析対象としない点や、抽出した後の修正支援をしないなどの課題があった。そのため、本研究では、本ツールの分析対象に無参照のものも

[†] 大阪大学大学院情報科学研究科

含めた。またユーザがアクセス修飾子の修正を効率的に行えるような支援機能を追加した。

さらに、本ツールの応用例としてある特定の時点のソフトウェアにおけるアクセス修飾子の過剰性だけではなく、ソフトウェアのバージョンが上がるとともにアクセス修飾子の過剰性がどのような変遷をするかという視点で、OSSに対する調査を行った。ここでは、Antを調査対象とし、その22個のバージョンに対してModiCheckerを適用して、結果を比較した。その結果、ソフトウェアの大きな変更に伴いアクセス修飾子の過剰性も大きく変化することが分かった。

本論文の構成について説明する。まず2節では、研究の背景となるJavaアクセス修飾子の仕様および過剰なアクセス修飾子の宣言によって引き起こされる問題について述べる。3節では、アクセス修飾子の過剰性の定義や分析するツールModiCheckerについて述べる。4節ではModiCheckerへの機能拡張について述べ、5節では、ModiCheckerの応用例としてソフトウェアのバージョン間におけるアクセス修飾子の過剰性を分析した結果について説明する。最後に6節で本論文のまとめと今後の研究について述べる。

2. アクセス修飾子とそれによって引き起こされる問題点

Javaの言語仕様では、フィールドやメソッドに対して外部からのアクセス範囲を制限できる修飾子を宣言することができる。これをアクセス修飾子と呼ぶ。Javaのアクセス修飾子は表1に示す4種類があり、上にいくほど広い範囲からのアクセスを許容する[3]。

表1 アクセス修飾子の種類

| アクセス修飾子 | アクセスを許容する範囲 |
|----------------|----------------------------------|
| public | 全ての部品 |
| protected | 自身と同じパッケージに所属する部品 および自身のサブクラス |
| default (指定なし) | 自身と同じパッケージに所属する部品 |
| private | 自身と同じクラス |

特に、クラスの外部から直接変更されるとプログラムの動作に異常をきたすようなフィールドは、クラス外部からの直接アクセスを許可しないアクセス修飾子“private”を宣言しておくことで、フィールドの利用方法をクラス設計者の想定内に収めることができる。これをカプセル化と呼び、オブジェクト指向プログラミングの主要な性質の1つとされている[4]。ところが実際には、ソフトウェアを開発する際、各部品の最終的なアクセス

範囲が不透明なままコーディングを開始することがあり、そのような状況では最終的に必要なアクセス範囲以外からのアクセスを許可するアクセス修飾子を設定することがある[5]。この問題を以下のような3つのメソッドを持つクラスXを例に説明する。

```
public class X {
    // フィールド y の初期値は null.
    private String y = null;

    // フィールド y に値を設定する.
    // クラス外から呼ばれることを想定していない.
    private methodA() {
        y = new String("hello");
    }
    // フィールド y の文字列長を返す.
    // クラス外から呼ばれることを想定していない.
    public methodB() {
        y.length();
    }
    // 値の設定されたフィールド y の文字列長を返す.
    // クラス外から呼ばれることを想定している.
    public methodC() {
        this.methodA();
        this.methodB();
    }
}
```

上記のソースコードでは、まずメソッドmethodAを呼び出してyにオブジェクトを代入してからメソッドmethodBを呼び出す必要がある。そこで、そのようなメソッド呼び出し順序を実装したメソッドmethodCを用意している。このメソッドmethodCは外部から呼び出されることを期待して作られているため、アクセス修飾子をpublicに設定している。一方、メソッドmethodBは外部から直接呼び出されてはならないにもかかわらず、アクセス修飾子もpublicに設定してしまっている。これにより、methodAを呼ばずにmethodBを呼ぶことが可能となってしまう。このような呼び出され方をした場合、フィールドyが初期値nullの状態でもメソッドlengthが呼ばれるため、例外NullPointerExceptionが発生する。

このように、アクセス修飾子が過度に広く設定されている場合、意図しないメソッド呼び出しが不具合を産む。また、開発途中における開発者間の設計情報共有不足などにより、想定外の状況下でメソッドが呼ばれることで、論理的なバグ発生の原因が作られる。しかし、Javaの構文上は問題がないため、このような状況をコンパイラ等を用いて機械的に検出することは難しい。また、全てのアクセス修飾子が適切に設定されているかどうかを、レ

ビューによって確認するには高いコストが必要である。

3. ModiChecker

我々は、指定された Java のソースコード群に宣言されたメソッドとフィールドに対して、宣言されているアクセス修飾子と実際に呼び出されている範囲との差異を表現する Accessibility Excessiveness (以下 AE) を定義し、AE をソースコード中から検出するためのツール ModiChecker を開発した [6]。本節では、AE と ModiChecker について説明する。

宣言されているアクセス修飾子と、実際にアクセスされている範囲の組み合わせにより、表 2 の背景色がある位置に示す pub1, pub2, pub3, pro1, pro2, def1 の 6 種類を AE として定義する。例えば、pub1 とは、アクセス修飾子として public が宣言されているフィールドまたはメソッドで、かつ、実際にアクセスされている範囲は

protected と同じである状態を意味する。

一方、pub0, pro0, def0 そして pri0 は、メソッドやフィールドのアクセス修飾子の宣言と実際にアクセスされている範囲が等しい状態、つまり適正な宣言が行われている状態を意味する。また、表 2 で x と表示されている箇所の記述は通常コンパイラによりエラーとして検出される状態を意味する。

表 2 AE の種類

| + | ‡ | public | protected | default | private |
|-----------|---|--------|-----------|---------|---------|
| public | | pub0 | pub1 | pub2 | pub3 |
| protected | | x | pro0 | pro1 | pro2 |
| default | | x | x | def0 | def1 |
| private | | x | x | x | pri0 |

† 列タイトル：宣言されているアクセス修飾子

‡ 行タイトル：実際にアクセスされている範囲

ModiChecker は、与えられたソースコード中の AE であるフィールドやメソッドを探し出し、その結果を図 1

| field name | Declaration ... | Current Modif... | Recommend... | protected | default | file name | line | column |
|-----------------|------------------|------------------|--------------|-----------|------------------|-------------------|------|--------|
| sqlID | jp.terasoluna... | protected | private | | | C:\Users\t-isi... | 29 | 5 |
| bindParams | jp.terasoluna... | protected | private | | | C:\Users\t-isi... | 34 | 5 |
| stream | jp.terasoluna... | protected | private | | | C:\Users\t-isi... | 37 | 5 |
| byteArray | jp.terasoluna... | protected | private | | | C:\Users\t-isi... | 38 | 5 |
| RANDOM_ID... | jp.terasoluna... | public | default | | jp.terasoluna... | C:\Users\t-isi... | 39 | 5 |
| CLASS_FIEL... | jp.terasoluna... | public | private | | | C:\Users\t-isi... | 39 | 5 |
| BEAN POIN... | jp.terasoluna... | public | private | | | C:\Users\t-isi... | 39 | 5 |
| dataRowHan... | jp.terasoluna... | protected | private | | | C:\Users\t-isi... | 40 | 5 |
| DYNAMIC_P... | jp.terasoluna... | public | private | | | C:\Users\t-isi... | 40 | 5 |
| file | jp.terasoluna... | protected | private | | | C:\Users\t-isi... | 41 | 5 |
| code | jp.terasoluna... | protected | default | | jp.terasoluna... | C:\Users\t-isi... | 42 | 5 |
| LINE_SEP | jp.terasoluna... | public | private | | | C:\Users\t-isi... | 43 | 5 |
| value | jp.terasoluna... | protected | private | | | C:\Users\t-isi... | 47 | 5 |
| language | jp.terasoluna... | protected | private | | | C:\Users\t-isi... | 47 | 5 |
| rsCodeColu... | jp.terasoluna... | protected | private | | | C:\Users\t-isi... | 48 | 5 |
| list | jp.terasoluna... | protected | private | | | C:\Users\t-isi... | 50 | 5 |
| country | jp.terasoluna... | protected | private | | | C:\Users\t-isi... | 52 | 5 |
| rsLanguage... | jp.terasoluna... | protected | private | | | C:\Users\t-isi... | 53 | 5 |
| variant | jp.terasoluna... | protected | private | | | C:\Users\t-isi... | 57 | 5 |
| groupList | jp.terasoluna... | protected | private | | | C:\Users\t-isi... | 57 | 5 |
| rsCountryCol... | jp.terasoluna... | protected | private | | | C:\Users\t-isi... | 58 | 5 |
| message | jp.terasoluna... | protected | default | | jp.terasoluna... | C:\Users\t-isi... | 62 | 5 |
| rsVariantCol... | jp.terasoluna... | protected | private | | | C:\Users\t-isi... | 63 | 5 |
| EVIDENCE... | jp.terasoluna... | public | private | | | C:\Users\t-isi... | 66 | 5 |
| rsMessageC... | jp.terasoluna... | protected | private | | | C:\Users\t-isi... | 68 | 5 |
| DEFAULT_P... | jp.terasoluna... | public | private | | | C:\Users\t-isi... | 71 | 5 |
| log | jp.terasoluna... | default | private | | | C:\Users\t-isi... | 72 | 5 |
| ... | jp.terasoluna... | protected | private | | | C:\Users\t-isi... | ... | ... |

図 1 ModiChecker の AE 表示画面

のように表示する。図1で表示される主な分析結果として、3列目 (Current Modifier) に現在宣言されているアクセス修飾子、4列目 (Recommended) に静的解析によって判明した実際にアクセスされている範囲が表示される。

このように、ModiCheckerを用いることで、開発者はソースコード中でAEであるフィールドおよびメソッドの状態を知ることができる。

4. ModiCheckerの機能拡張

4.1. アクセスが無いフィールドやメソッドの分析

既存のModiCheckerでは、宣言されてはいるものの、実際にはどこからもアクセスされていないフィールドやメソッドは分析の対象としていなかった。しかし、実際のプログラムを分析してみると、そのようなどこからもアクセスされないフィールドやメソッドが多数あることに気づいた。

そこで、これらのどこからもアクセスされていないこともAEの一種と考え、表3のように、pub4, pro3, def2, pri1の4種類のAEを新たに定義した。例えば、pub4はフィールドもしくはメソッドのアクセス修飾子としてpublicが宣言されているが、実際にはどこからもアクセスされていない状態を意味している。

また、ModiCheckerを拡張し、これらの新しいAEについても図1に示すAE表示画面で扱えるようにした。

表3 拡張したAEの種類

| + | public | protected | default | private | No Access |
|-----------|--------|-----------|---------|---------|-----------|
| public | pub0 | pub1 | pub2 | pub3 | pub4 |
| protected | x | pro0 | pro1 | pro2 | pro3 |
| default | x | x | def0 | def1 | def2 |
| private | x | x | x | pri0 | pri1 |

†列タイトル：宣言されているアクセス修飾子

‡行タイトル：実際にアクセスされている範囲

4.2. AE修正支援機能

ModiCheckerを用いて分析を行い、AEが多数存在すると分かった場合、そのようなアクセス修飾子をModiCheckerが推奨するアクセス修飾子へ修整するためには、別途、エディタ等を用いて、それぞれのアクセス修飾子を編集しなければならない。AEが大量に検出された場合、この作業は大きな手間が必要であった。

そこで本研究では、AEとなっているフィールドおよびメソッドに対して、推奨するアクセス修飾子へと自動的に修正する機能をModiCheckerに実装した。本機能は、元のソースコードからアクセス修飾子の宣言部のみ

を変換して、推奨するアクセス修飾子に変更した新しいソースコードを自動的に作成する。

ただし、一般的なソフトウェア開発においては、理由があって特定のフィールドやメソッドに過剰なアクセス修飾子を設定している場合も存在する。例えば、以下のような例が考えられる。

- ・ライブラリやAPフレームワークを開発している際に、特定のフィールドやメソッドが開発対象外からの呼び出しを前提として実装されている場合
- ・ソースコードが開発途中である場合、将来的に内部および外部からアクセスされる可能性を残しているため、アクセス修飾子は現状必要な範囲より広いものを選択している場合

上記のような理由を持つフィールドおよびメソッドを開発者自身が選択し、自動修正の対象から外すことのできるユーザインタフェースを実装した。具体的には、図1に示すAE表示画面において、下記の手順による選択的なアクセス修飾子の自動修正に対応した。

1. アクセス修飾子変更したいフィールドやメソッドの行をマウス操作によって選択する。
2. 「ChangeAccessModifier」ボタンを押下する。
3. 選択した部分に対して推奨するアクセス修飾子に変更されたソースコードを自動生成する。

図1に示す出力結果を例に説明する。3行目でフィールド「stream」のアクセス修飾子が現在はprotectedになっている。しかし、実際の呼び出し関係をModiCheckerが分析した結果、アクセス修飾子としてprivateが推奨されている。ここで、前述に示すような特別な理由がないと開発者が判断した場合は、上記に示した手順によりフィールド「stream」のアクセス修飾子はprivateに修正される。

この機能を用いることで、開発者は必要なフィールドおよびメソッドだけに対してAEを効率良く修正することができる。

4.3. AE修正支援機能の検証

前述のAE修正支援機能は、対象プログラム群に対するリファクタリングの一種であるため、修正前と修正後のプログラムの動作が変わってはいけない。そこで、プログラムの動作確認を行うテストスイートを持つAntのバージョン1.8.2を対象として、そのテストスイートを用いて、修正の実施前後でテストを行い、動作が変わっていないことを確認する [7]。

実験の手順は以下の通りである。

1. 実験対象の Ant に対して、動作を確認するテストスイートを実行する。
2. テストスイートが正常に終了したことを確認する。
3. 実験対象のソースコードを ModiChecker に入力し、AE の検出を行う。
4. 実験対象のソースコード群の外部からのアクセスを前提としているなど、開発者が意図的に設定している AE を修正対象から外す。
5. それら以外の全ての AE に対して修正機能を実行する。
6. AE の修正が行われたソースコードが自動生成されたことを確認する。
7. 自動生成されたソースコードに対して手順 1 で使用したテストスイートを実行する。
8. テストスイートが正常に終了したことを確認する。

実験対象のプログラムに上記手順を実施したところ、798 個のフィールドおよび 6076 個のメソッドが AE であると判定された。これらの内、5695 個のメソッドに関しては、ビルド時に Ant 外部からアクセスされる必要があるため、開発者が意図的に AE となるようアクセス修飾子を設定したと考えられるため AE 修正対象から外した。残りの 798 個のフィールドおよび 381 個のメソッドに対して AE の自動修正を実施したところ、修正後のプログラムに対してもテストスイートは正常に終了し、ModiChecker による自動修正によって機能性を損ねていないことが確認できた。

これにより、プログラムの動作を変えることなく、ModiChecker を用いてアクセス修飾子を必要最低限の範囲へ変更できることがわかった。

なお、上記手順 4 で実施した修正対象の選別においては、下記の手順をビルドエラーが出なくなるまで 6 回繰り返し、合計で約 63 分を要した。

1. ビルドエラーのログを確認する
2. ビルドエラーの原因となったメソッドのアクセス修飾子を ModiChecker の AE 修正対象から外す
3. AE 修正前の Ant ソースコードに対して、再度 ModiChecker の AE 修正機能を利用した AE 修正を実行し、AE 修正済みの Ant のソースコードを作成する
4. AE 修正済みの Ant のソースコードに対して再度ビルドを実行する

このうち、ModiChecker の実行時間は約 33 分で、1 回あたり平均 333 秒であった。これは実用に耐えうる性能であると判断した。

5. 機能拡張の応用例としてのバージョン間分析

AE に関する既存研究では、ある時点でのソースコード群を対象とした AE 数について考察を行っていた [6]。しかし、ある時点での AE 数の情報だけでは、不適切なアクセス修飾子もしくは将来的な拡張性を考慮したアクセス修飾子を持つ、未熟な機能の多寡については判断できていなかった。そこで本節では、同じソフトウェアの複数のバージョン間における AE 数の変化量を比較分析することで、どのようなバージョンアップ時に過剰性を残したアクセス修飾子が追加されるのか分析する。

まず、ソフトウェアのバージョンアップを、機能拡張など比較的大きな変化を伴うと予想される「メジャーバージョンアップ（以降、MajorVU と呼ぶ）」と、機能のバグ修正など、比較的小さな変化を伴うと予想される「マイナーバージョンアップ（以降、MinorVU と呼ぶ）」の 2 種類に分類する。機能が新しく追加される際には、将来的な拡張性を考慮して、アクセス修飾子には過剰性を残した設定が行われて AE 数の変化量が増加するが、機能のバグ修正が行われる際には、アクセス修飾子が関連しない限り AE 数は変化しないことが予想される。

上記の予想を検証するために、本論文ではソフトウェアの MajorVU 時の AE 数の変化量と MinorVU 時の AE 数の変化量に有意差があるかどうか実験により調べる。両者に有意差があることが分かれば、バージョンアップ時の AE 数の変化量を調べることで、開発者がソフトウェアの機能が成熟していると判断したかどうかを推測できると考えられる。なお、MajorVU 時および MinorVU 時に実際にどのような修正が行われたかという分析は、本論文の対象外とする。

本節では上記の有意差を確認する実験の対象ソフトウェアとして OSS の Ant を用いる [7]。Ant は Apache ソフトウェア財団が開発を行っているビルドツールであり、多くのバージョンのソースコードが入手可能である。本実験では Ant のバージョン 1.1 からバージョン 1.8.4 までの 22 バージョンを対象とした。

この実験における Ant の MajorVU と MinorVU の定義を以下に示す。

- MajorVU：左から 2 つ目以前のバージョン数が増えるタイミング。例えば 1.2 → 1.3 や、1.6.5 → 1.7.0 など。
- MinorVU：左から 3 つ目以降のバージョン数が増えるタイミング。例えば 1.4 → 1.4.1 や、1.6.4 → 1.6.5

など。

なお、今回の実験対象のバージョンアップ 22 回の内訳は、MajorVU が 7 回、MinorVU が 15 回であった。

また、4.3 節でも述べたように Ant 自身のビルドに必要なパッケージに属するフィールドやメソッドは開発者が意図的に AE であるように設定していることが予想されるため実験の対象から除外した。

表 4、表 5、表 6 に、それぞれ Ant のバージョン 1.3、1.4、1.4.1 におけるフィールドの AE 数を AE の種類ごとに分類して示す。バージョン 1.3 から 1.4 へは MajorVU であり、バージョン 1.3 から 1.4.1 へは MinorVU である。

これらのデータから、例えば pro2 については、MajorVU で 181 個から 314 個と急増し、MinorVU では、1 個の変化もなく、MajorVU と MinorVU で AE 数変化量に大きな差があることがわかる。また、全てのバージョンにおいて AE の約 80% は実際のアクセス範囲が private であり、多くのフィールドが実際にはカプセル化可能であることがわかる。また、NoAccess に関する AE は全体の 2 ~ 2.5% しかなくデッドコードが少ないことが推測できる。

表 4 バージョン 1.3 のフィールドの AE 数

| † ‡ | public | Protected | default | private | No Access |
|-----------|--------|-----------|---------|---------|-----------|
| public | 39 | 0 | 20 | 84 | 2 |
| protected | x | 15 | 37 | 181 | 3 |
| default | x | x | 1 | 43 | 2 |
| private | x | x | x | 952 | 21 |

† 列タイトル：宣言されているアクセス修飾子

‡ 行タイトル：実際にアクセスされている範囲

表 5 バージョン 1.4 のフィールドの AE 数

| † ‡ | public | Protected | default | private | No Access |
|-----------|--------|-----------|---------|---------|-----------|
| public | 49 | 3 | 8 | 82 | 6 |
| protected | x | 16 | 51 | 314 | 10 |
| default | x | x | 2 | 51 | 3 |
| private | x | x | x | 1214 | 27 |

表 6 バージョン 1.4.1 のフィールドの AE 数

| † ‡ | public | Protected | default | private | NoAccess |
|-----------|--------|-----------|---------|---------|----------|
| public | 49 | 3 | 8 | 82 | 6 |
| protected | x | 17 | 51 | 314 | 10 |
| default | x | x | 2 | 49 | 3 |
| private | x | x | x | 1217 | 27 |

各フィールドに対する AE の各要素の数についてバージョンアップ前後の差分を示した棒グラフを図 2 に示す。この図では、MajorVU 時 (図 2 の赤で囲った列) に pro2 と pub3 の変化量が大きく検出されている。pro2 と

pub3 は共に実際のアクセス範囲が private である AE であるため、多くのフィールドが実際にはカプセル化可能であることがわかる。さらに、多くの場合、MajorVU 時には AE 数の差分が大きく、MinorVU 時には小さいことが図 2 から推測できる。

そこで、MajorVU 時の AE 数変化量の群と、MinorVU 時の AE 数変化量の群との間に有意水準 5% における統計的な有意差があるかどうかを検定する実験を行った。

検定の対象としてバージョン 1.1 から 1.8.4 までの MajorVU 時の各フィールドの AE 数変化量の集合と MinorVU 時の各フィールドの AE 数変化量の集合を用いるが、両群のデータが正規分布に従っているかどうか不明であるため、正規分布であることを前提条件としないマン・ホイットニーの U 検定を用いたところ、検定における危険率 p 値および有意水準 5% における有意差の有無は表 7 および表 8 に示す通りであった。

表 7 Ant のフィールドに対する MajorVU と MinorVU 間の有意差の有無

| AE | p 値※ | 有意水準 0.05 における有意差 |
|----------|---------|-------------------|
| pub1 | 0.00080 | 有り |
| pub2 | 0.00114 | 有り |
| pub3 | 0.00113 | 有り |
| pub4 | 0.00032 | 有り |
| pro1 | 0.00002 | 有り |
| pro2 | 0.00001 | 有り |
| pro3 | 0.03715 | 有り |
| def1 | 0.00003 | 有り |
| def2 | 0.00479 | 有り |
| pri1 | 0.00192 | 有り |
| NoAccess | 0.01162 | 有り |

※数値は小数点以下第 6 位を四捨五入している

表 8 Ant のメソッドに対する MajorVU と MinorVU 間の有意差の有無

| AE | p 値※ | 有意水準 0.05 における有意差 |
|----------|---------|-------------------|
| pub1 | 0.00440 | 有り |
| pub2 | 0.00562 | 有り |
| pub3 | 0.07205 | 無し |
| pub4 | 0.00122 | 有り |
| pro1 | 0.00361 | 有り |
| pro2 | 0.00225 | 有り |
| pro3 | 0.02073 | 有り |
| def1 | 0.07656 | 無し |
| def2 | 0.13130 | 無し |
| pri1 | 0.00919 | 有り |
| NoAccess | 0.00012 | 有り |

※数値は小数点以下第 6 位を四捨五入している

以上の実験結果から、今回の実験対象である Ant のフィールドに関しては全ての種類の AE 数に関するバージョンアップ時の差分について、ソフトウェ

アの MajorVU 時と MinorVU 時の間で有意差がみられることがわかった。また、メソッドに関しては、pub3,def1,def2 以外の AE についてはバージョンアップ時の AE 数の差分について、ソフトウェアの MajorVU 時と MinorVU 時の間で有意差がみられ、フィールドとメソッドの両者とも MajorVU 時の方が MinorVU 時よりも AE 差分量が大きくなることがわかった。有意水準 0.05 における有意差が見られなかった pub3, def1 および def2 は他の AE に比べて各バージョン間の AE 変化量の値が小さく、順位がタイとなる値も多かったためマン・ホイットニーの U 検定では誤差が出やすい状況下であった。

本実験の結果を応用すると、ある保守対象のソフトウェアのフィールドおよびメソッドに対してバージョン間の AE 数の差分を分析し、差分が大きく変化したバージョンアップでは本実験の MajorVU に相当するような変化が発生しており、アクセス修飾子に過剰性を残した機能が追加された可能性があることを発見できる。

6. 関連研究

6.1. アクセス修飾子の解析に関する関連研究

アクセス修飾子の解析に関して、我々の研究以前にくつかの研究がなされている。Müller は Java のアクセス修飾子をチェックするためのバイトコード解析手法を提案している [8]。Müller の研究では、我々と似た目的のために Java のバイトコードを解析する AMA (Access Modifier Analyzer) というツールを開発している。しかし、バイトコードに対する解析は、コンパイル時に追加されるフィールドやメソッドの影響で、必ずしもソースコードに対する解析と同じ結果にはならない。さらに、Müller はツールを用いた実践的な実験結果を報告していない。一方、我々の研究では実際に既存のソフトウェアに対して複数の側面から実験したデータを明示し、評価も行った。

Cohen は複数のサンプルメソッドにおける各アクセス修飾子の数の分布を調査した [9]。Evans らは、静的解

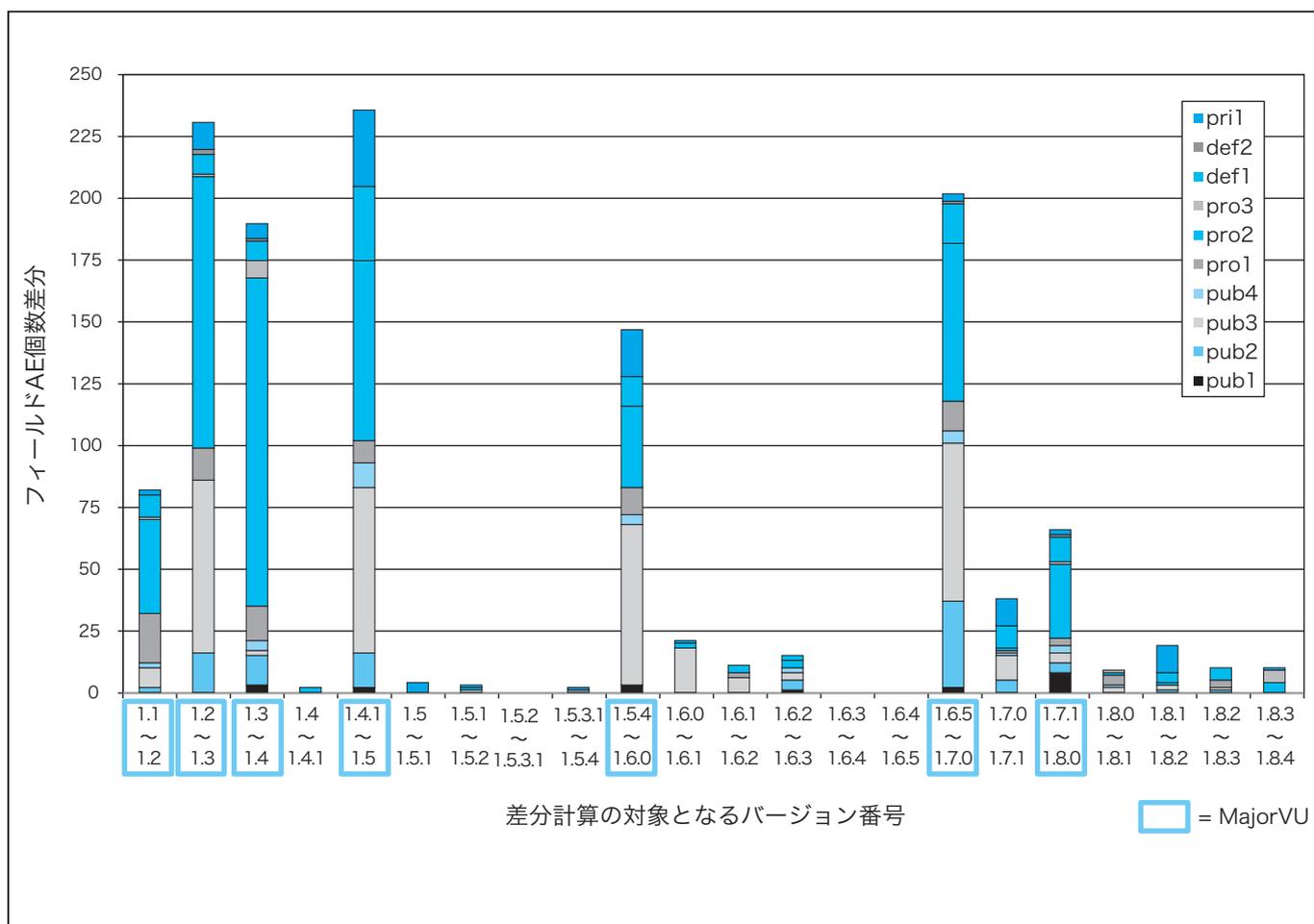


図 2 Ant のバージョン間における、フィールドの AE 数の各要素数の差分

析によるセキュリティ脆弱性の解析を研究した [10]. これらの研究で課題となっているアクセス修飾子の宣言に関しては Viega らによって議論されている [11]. Viega らは, private にすべきだがそのように宣言されていないメソッドやフィールドについて, 警告を出すツール Jslint を開発している. 一方, 我々の開発したツール ModiChecker では, private だけでなく全てのアクセス修飾子に対する警告を出すことができる.

アクセス修飾子の数をメトリクスとしている点については, 我々の過去の研究とも関連がある [12]. この過去の研究では, Java のソースコードの類似性を計算するためのメトリクスの一部として, アクセス修飾子の宣言数が用いられている.

6.2. ソースコードの静的解析に関する関連研究

Java に関するソースコード静的解析ツールは多数存在する [13][14]. これらのツールはデッドロックやオーバーロード, 配列のオーバーフロー等のコーディング上の悪いパターンや, 潜在的なバグを検出するため, 今日の Java プログラム開発では重要なツールである.

Rutar らは, このような機能を持つ 5 つのツールを比較分析した [15]. しかし, これらのツールは, 本論文のようにアクセス修飾子の冗長性を解析する機能は持っていない.

7. まとめと今後の研究

本研究では, Java のアクセス修飾子を分析するツール ModiChecker を利用して, 以下のことを行った.

1. AE の概念を拡張し, アクセスが無いフィールドやメソッドの分析も可能にした.
2. 過剰なアクセス修飾子を持つフィールドおよびメソッドの中から, 開発者が選択したものに対してアクセス修飾子を自動修正する機能を開発し, その妥当性を検証した.
3. 複数のバージョンの AE の分析を通じて, 全ての AE に関して, AE 数の差分とソフトウェアのバージョンアップ種別 (MajorVU / MinorVU) に相関があることを確認した.

本結果は, 特定の OSS の分析に基づいている. 今後, 多様なソフトウェアの分析を通じて, より一般的な関係を導き出す必要があるだろう. さらに, 発展的なテーマとしてソフトウェアの機能が成熟しているかどうかを区別するための AE の閾値の考え方を整理することで, ソフトウェアの現場が判断しやすくするための研究を行うこと

が考えられる.

また, AE とプログラムの品質との関係の調査が挙げられる. 具体的には, 過度に広い範囲のアクセスを許可しているアクセス修飾子を設定している部品と, 最適なアクセス修飾子の設定になっている部品の間で, バグの検出率に有意差がでるかどうかを調べ, 有意差が出るような AE の閾値を見極めることができた後には, AE を用いたプログラムの品質判定方法を提案することで, プログラムのリリース判定を支援することを目指すことが考えられる.

謝辞

本研究を進めるにあたり, 検定手法の選択等において, 株式会社 NTT データの山本英之氏に多くの知識や示唆を頂きましたことを感謝致します.

【参考文献】

- [1] G. Booch, R. Maksimchuk, M. Engel, B. Young, J. Conallen and K. Houston; "Object-Oriented Analysis and Design with Applications", Addison Wesley, 2007
- [2] K. Arnold, J. Goslin and D. Holmes; "The Java Programming Language, 4th Edition", Prentice Hall, 2005.
- [3] J. Gosling, B. Joy, G. Steele, G. Bracha and A. Buckley; The Java Language Specification, Java SE 7 Edition, <http://docs.oracle.com/javase/specs/jls/se7/html/index.html>
- [4] K. Khor, N. Chavis, S. Lovett and D. White; "Welcome to IBM Smalltalk Tutorial", 1995
- [5] Nghi Truong, Paul Roe, and Peter Bancroft; "Static analysis of students' Java programs", In Proc. ACE '04, 317-325, 2004.
- [6] D. Quoc, K. Kobori, N. Yoshida, Y. Higo and K. Inoue; ModiChecker: Accessibility Excessiveness Analysis Tool for Java Program, 日本ソフトウェア科学会大会講演論文集 vol.29, pp.212-218, 2012, コンピュータ・ソフトウェア.
- [7] Apache Ant, <http://ant.apache.org/>
- [8] A. Müller; "Bytecode Analysis for Checking Java Access Modifiers", Work in Progress and Poster Session, 8th Int. Conf. on Principles and Practice of Programming in Java (PPPJ 2010), Vienna, Austria, 2010.
- [9] T. Cohen; "Self-Calibration of Metrics of Java Methods towards the Discovery of the Common Programming Practice", The Senate of the Technion, Israel Institute of Technology, Kislef 5762, Haifa, 2001.
- [10] D. Evans and D. Laroche; "Improving Security Using Extensible Lightweight Static Analysis", IEEE software, vol.19, No.1, pp. 42-51, 2002.
- [11] J. Viega, G. McGraw, T. Mutdosch and E. Felten; "Statically Scanning Java Code: Finding Security Vulnerabilities", IEEE software, Vol.17 No.5 pp. 68-74, 2000.
- [12] K. Kobori, T. Yamamoto, M. Matsushita and K. Inoue; "Java Program Similarity Measurement Method Using Token Structure and Execution Control Structure", Transactions of IEICE, Vol. J90-D No.4, pp. 1158-1160, 2007.
- [13] FindBugs, <http://findbugs.sourceforge.net/>
- [14] Jlint, <http://jlint.sourceforge.net/>
- [15] N. Rutar, C. Almazan, and J. Foster; "A Comparison of Bug Finding Tools for Java", 15th International Symposium on Software Reliability Engineering (ISSRE 04), pp. 245-256, Saint-Malo, France, 2004.

SEC 特別セミナー 「IV&V^{※1} 最新動向」について

SEC 統合系プロジェクト 研究員

香西 省治

1. はじめに

近年日本でも IV&V が注目を集めている。そこで 2013 年 2 月 26 日に SEC 特別セミナーを開催し、日米 IV&V の第一人者である慶應義塾大学の白坂成功准教授とウェストバージニア大学のキューキック教授のお二人に講演して頂いた。ここでは、講演内容から IV&V の背景、実施イメージ、実践に基づく効果について解説する。白坂准教授は宇宙航空研究開発機構 (JAXA) で、またキューキック教授は主としてアメリカ航空宇宙局 (NASA) で、IV&V に深く関わって来られた。

なお、セミナー当日の資料は、SEC ホームページ^{※2} からダウンロード可能になっている。

2. ソフトウェア産業の動向 (IV&V が有効な背景)

現代社会はソフトウェアなしでは成り立たない。ケイパー・ジョーンズ氏は、米国 1 万 3 千件のプロジェクトを調査した結果、ソフトウェア開発で最もコストが発生するのは、障害解析とその対応であると報告している^{※3}。また、IV&V に早くから取り組んできたキューキック教授は、NASA での障害発生事例を 10 年以上にわたって調査・分析した結果、障害原因には以下の分布傾向が見られることを指摘している。

- ・要求、設計工程 (要求, 設計の誤り) 38.25%
- ・要求、設計工程以降 (コーディング、統合試験の誤り) 48.57%

つまり、障害原因はソフトウェアライフサイクルの早期工程だけでなくすべての工程に分布していること。また、発生した障害の危険度の分析結果では、クリティカルなものも全工程で発生していること。さらに NASA の 15 のプロジェクトを障害検出の難易度の観点で詳細分析した結果、50～70% は容易にテストで検出できるものだが、残りはテストで検出が困難なものであったこと等である。テストでは検出が困難なバグが作りこまれる

と、運用開始後に障害が発生した場合には、その対応に大きなコストがかかる。そのため、このようなバグを作りこまないように、開発の早い工程で問題点を抽出することが望ましい。米国スタンディッシュ・グループ社が実施した 1990 年代の米国 8,380 のソフトウェアプロジェクトの調査では、計画通り完了したプロジェクトは 16% に過ぎず、30% は途中で打切られ、残りはリソース追加、納期延長が必要であったと報告されている^{※4}。外部の専門知識を持ったチームが IV&V を実施することによって、①開発プロセスの信頼性が上がる、②目的が達成されているということを定期的に確認できる、③問題の発見がタイムリーに行われる、④途中で大きな意思決定をする場合に、その決定事項が検証される、⑤必要な専門知識を有する人材を開発プロセス中のみ確保できる、といった効果が期待でき、リスク緩和と予算の全体的な縮減を達成できる。

3. IV&V とは

IV&V とは、Verification (検証)：製品を正しく構築できているのかどうかを確認する、と Validation (妥当性確認)：必要とされている製品を作っているのかどうかを確認する、を Independent (独立) に実施することである。例えば、最初の要求仕様で誤りが生じた場合、その後、いくらシステムを正しく開発しているかの検証 (Verification) をしても、最終的に要求にあったシステムかどうかは確認できない。この場合には、要求仕様の妥当性確認 (Validation) が重要になる。独立

【脚注】

- ※1 IV&V は Independent Verification and Validation の略。「独立検証及び妥当性確認」とも言われる。
- ※2 <http://sec.ipa.go.jp/seminar/2013/20130226.html>
IPA チャンネルにてビデオも公開。
- ※3 ケイパー・ジョーンズ (Capers Jones) 氏は米国ソフトウェア工学方法論のスペシャリストで、特に FP 法の権威。米国 DACS の 2012 年 2 月 4 日付け氏の論文で言及している。
- ※4 タイトル名は "THE STANDISH GROUP REPORT CHAOS".
<http://www.projectsmart.co.uk/docs/chaos-report.pdf>

(Independent) については、「IEEE 1012 - Standard for System and Software Verification and Validation」で、①技術面、②管理面、③財務面、で開発者と独立であることが規定されている。これらの独立性により、IV&V 実施組織は、開発組織と独立のプライオリティを持って IV&V を実施し、その結果を委託者に報告することができる。なお、3つの独立性をどこまで徹底するかに関して IEEE 1012 では、Classical (理想) から Embedded (同じチーム内) まで5つのパターンが定義されている。実際の IV&V では、プロジェクトの状況、環境条件等に応じて適切なパターンを選択して実施すると効果的である。

4. IV&Vのプロセス、アプローチ

IV&V はプロジェクトの早期の工程から導入し、早期に Validation (Early Validation) することでより大きな効果を得ることができる。一般的な V モデルにおける Verification と Validation の関係を図 1 に示す。ソフトウェアライフサイクルの全工程にわたって IV&V の活動がある。IV&V の重要なファクタのひとつは、IV & V チームが開発者とは独立に「開発対象のモデル」を構築して妥当性を検証することである。ただし、IV&V の具体的な活動内容に黄金律は存在せず、IV&V チームは対象となるプロジェクトに応じて最適なやり方を検討し、実施する必要がある。一般に IV&V にかかるコストはプロジェクト全体の開発費の 5 ~ 10% 程度までが目安であり、IV&V チームは最初に、IV&V が予算の範囲に収まるように図 1 のどの部分をどこまで実施するかを、顧客や開発者と議論して決定する。

IV&V チームは開発者と独立しているが、両者の関係は競争関係や対立関係ではなく、顧客の期待通りにプロジェクトを完了させるという共通の目的を持って互いに緊密に協力して作業を進める。要求に対するレビュー、

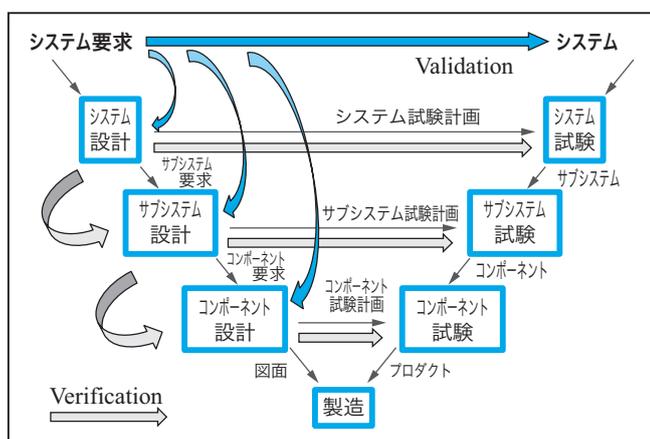


図 1 Independent Verification and Validation

開発に関するレビュー等々重要なレビューは、IV&V チームと開発者のそれぞれが同一のタイミングで実施する必要がある。IV&V チームが問題点を発見すると、開発者に報告し解決方法を協議する。開発者との協議で合意が成立しなかった場合には、IV&V チームが顧客にその問題点を報告する。ここで、IV&V としての作業は問題点の検出までであり、責任分担上から問題点の修正は、開発者が実施することになっている。

5. 実践から得られた IV&V の効果

米国では IV&V サービス提供企業が既に約 10 社あり、NASA や連邦航空局 (FAA) 等の連邦政府機関や地方自治体がこれら企業と契約して IV&V を実施している。

IV&V の効果として、問題点の指摘よりむしろ開発者の意識が高まったことが一番だと白坂准教授は言う。また IV&V サービスを提供した際に、顧客と開発者から、IV&V によってプロジェクトが計画通りに進む可能性が高まった、という良い評価を聞いたとキューキック教授は言っている。顧客と開発者の両者から IV&V の効果について高評価を得ていることは、大変興味深い。

6. おわりに

キューキック教授は IV&V が適するプロジェクトの特性として、開発手法の違いではなく、①期間が長いこと、②リスクが高いこと、③障害発生時の影響が大きいこと、を挙げる。特に信頼性をテストを通じて効果的に実証できないようなシステムでは、IV&V が必要になる。ただ、IV&V の根本的な考え方である、開発者と異なる立場でプロジェクトの全工程において検証と妥当性確認を行うことは、プロジェクト規模の大小や期間の長短に関わらず有効であるが、開発者が V&V をきちんと実施することも、一つのパターンとして試行する意義は十分あるとのことであった。

講演では、開発者の立場で宇宙ステーション補給機「こうのとりのり」の IV&V を受けた経験から、具体的な IV&V の実施内容や効果、更には IV&V に関連する最近の話題を白坂准教授から紹介して頂いた。また、キューキック教授からは、IV&V を実施した立場として、IV&V の手法の有効性に関する分析結果より、高度なツールだけでなく人間の分析や直感の重要性に関する説明、更には IV&V をコストを抑えて実施する方法等についても紹介頂いた。紙面の関係で、いずれもここでは省略しているが、詳しくは講演資料を参照して頂きたい。

ソフトウェア工学分野の 先導的研究支援事業について

SEC 調査役

小沢 理康

IPA/SEC では、大学等におけるソフトウェア工学分野の研究活動の活性化を図る目的で、2011 年度より研究公募の仕組みを創設し、「ソフトウェア工学分野の先導的研究支援事業」として実施している。2012 年度は 9 件の応募があり、そのうち 5 件を採択した。2013 年度も引き続き公募を行ったところ、7 件の応募があり、そのうち 5 件を採択した。本稿ではその事業概要と、2012 年度事業の成果、2013 年度事業の公募における採択状況について報告する。本事業は 5 年間の継続を予定しており、次年度以降も公募を実施する予定である。

1. 研究支援事業の概要

ソフトウェアは、あらゆる産業や市民生活を支える基盤として不可欠な存在となってきており、複雑化・大規模化するソフトウェアの高信頼化や開発プロセスの高度化等を研究領域とするソフトウェア工学の果たす役割はますます増大してきている。このような背景から、SEC は、ソフトウェアの開発現場が持つ様々な課題に対して、実践的な解決策をもたらすソフトウェア工学の一層の振興をねらいとして本事業を創設した。

本事業では、ソフトウェアの開発・利用の現場に密着した研究をとくに重視しており、開発現場と連携した研

究を促進するため、産業界の有識者から成る「ソフトウェア工学研究推進委員会」を設置し、公募する研究分野を決め、研究テーマの選考と研究に対する助言も行いながら実施することとしている。

2012 年度、2013 年度に公募した研究分野及び区分を表 1 に示す。

2. 2012 年度事業の成果

2012 年度に採択した 5 件の研究テーマは 2012 年 6 月から 2013 年 1 月までの期間で実施され、研究成果は成果報告書として取りまとめられた。成果報告書は以下の URL からダウンロードすることができる。

<http://sec.ipa.go.jp/reports/20130422.html>

5 件の研究テーマ、研究分野区分、実施機関名、研究の概要を以下に示す。

実用性が高い形式工学手法と支援ツールの研究開発 (A 区分)：学校法人法政大学

要求分析を形式手法で抽象化する際に非形式仕様、半形式仕様、形式仕様の 3 段階に分け、それぞれの段階の仕様を仮想環境の中で動的に表現 (アニメーション化) するツールを開発することで、形式仕様を顧客にも理解しやすくする。

表 1 公募した研究分野および区分

| 区分 | 分野名 | 分野の概要説明 |
|----|-------------------------------|---|
| A | ソフトウェア工学分野の先導的な研究 | 要求工学、プロセス改善、高信頼性、アジャイル開発、形式手法、モデルベース開発等のソフトウェア工学分野の先導的な研究 |
| B | ソフトウェア開発現場へのソフトウェア工学の適用に関する研究 | ソフトウェア開発現場への適用を目的としたソフトウェア工学の成果・手法を詳細化・具体化・実用化する研究 |
| C | ソフトウェアが経済社会にもたらす革新的効果に関する実証研究 | ソフトウェアが社会や組織経営にもたらす経済価値、生産性向上、競争力強化、イノベーション等の経済効果についての実証 |

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に関する研究 (B 区分)：学校法人芝浦工業大学

システム構築の上流工程において、仕様の妥当性を検証するための形式検証技術（モデル検査技術）を一般的な開発者が利用可能な場面を想定した検証方法とその支援ツールを研究開発する。大規模システムにおける仕様の誤解や実現可能性についての不具合を上流工程において早期発見できるようにする。

コードクローン分析に基づくソフトウェア開発・保守支援に関する研究 (A 区分)：国立大学法人大阪大学

潜在的な不具合の埋込み、一貫した修正に対する悪影響が指摘されているコードクローン（ソースコード上に存在する同一または類似したコード片）の分析方法を開発する。レガシーシステムの理解支援、コードクローンの集約支援、メソッド単位の粒度で再利用に有用なコードの特定等、利用目的に応じて、ソフトウェア開発・保守を技術面、管理面で改善する。

ソフトウェア品質の第三者評価のための基盤技術—ソフトウェアプロジェクトトモグラフィの開発— (A 区分)：国立大学法人奈良先端科学技術大学院大学

開発管理のためにクラウド上に収集・蓄積されているプロジェクトデータを、スナップショットの時系列解析として可視化に都合の良い形式に変換するソフトウェアプロジェクトトモグラフィ技術を開発する。また開発管理+品質の第三者評価をサービスとして提供できるクラウド型開発管理環境を構築・評価し、その有効性を評価することでソフトウェア品質の第三者評価を容易にする技術を確立する。

モデルを含む設計成果物の集積とその活用方法に関する研究 (A 区分)：国立大学法人九州大学

ソフトウェア・システムモデルの編集履歴等を簡単に収集できるツールを開発するとともに、モデルとその編集履歴を継続的に収集できる仕組みを構築し、収集したデータの有用性を評価する。従来ソースコードが対象であった定量的手法を、モデルを使用した開発にも適用可能な仕組みを構築できる。

また 2013 年 3 月 11 日には、支援事業で実施した研究の成果を広く産業界へ普及展開するために、「第 1 回産学連携のためのソフトウェア・シンポジウム」を開催した。このシンポジウムは、前半の各大学による研究成果の講演と、後半の各大学の個別ポスターセッションとで構成して実施した。ポスターセッションでは参加者と各研究責任者との熱心な議論が行われた。同シンポジウムの講演資料のダウンロードと講演動画の閲覧は以下の URL のリンク先からできるようになっている。

<http://sec.ipa.go.jp/seminar/2013/20130311.html>

3 2013 年度公募の状況と採択結果

2013 年度の公募では 7 件の応募があった。研究分野の区分による内訳は A 区分 3 件、B 区分 3 件、C 区分 1 件であった。これらの提案については、ソフトウェア工学研究推進委員会において厳正な審査を行い、5 件の研究提案の採択を決定した（表 2）。昨年度採択された研究提案にはなかった C 区分については、ソフトウェア工学研究推進委員会でも本年度の採択を期待していたが、結果として採択には至らなかった。次回の公募で採択されることを期待したい。

なお、2013 年度から実証実験の実施や仮説に基づくアンケート調査の実施等、相応の期間を有する研究項目を含む場合は、例外として 2 年度にまたがる研究提案も可とした。

表 2 2013 年度採択研究提案一覧

| 区分 | 期間 | 研究テーマ名 | 提案者名 |
|----|-----|---|-----------------------------|
| A | 1 年 | 次世代ソフトウェア信頼性評価技術の開発とその実装 | 国立大学法人 広島大学 |
| B | 1 年 | 抽象化に基づいた UML 設計の検証支援ツールの開発 | 公立大学法人 岡山県立大学 |
| A | 2 年 | ソフトウェア品質の第三者評価のための基盤技術—ソフトウェアプロジェクトトモグラフィ技術の高度化— | 国立大学法人 奈良先端科学技術 大学院大学 |
| A | 2 年 | IPA EPM-X の機能拡張によるプロアクティブ型プロジェクトモニタリング環境の構築—一次世代の定量的プロジェクト管理ツールとリポジトリマイニング研究基盤— | 国立大学法人 和歌山大学 |
| B | 2 年 | 形式仕様とテスト生成の部分的・段階的な活用～探索を通じたコード中心インクリメンタル型開発の支援 | 大学共同利用機関法人 情報・システム研究機構 |

コンピュータは「ソフトウェア危機」を救えるか？

IPA 顧問

松田 晃一

プロ棋士に勝ったコンピュータ

プロ棋士 5 人に将棋のコンピュータソフトが挑戦する電王戦。コンピュータソフトが三勝一敗一分けで勝利しました。一般の新聞などマスコミで度々取上げられ話題になりましたのでご記憶の方も多いと思います。2012 年一月には米長邦雄永世棋聖が、さらに 2010 年の情報処理学会設立 50 周年記念イベントでは、清水市代女流王将（当時）が、コンピュータソフトと対戦しいずれもコンピュータが勝利を取っています。

また米国では、複雑な質問文を理解して適切な回答を見つけて早押しで答えるクイズ番組ジョパディでコンピュータが人間と対戦し、それまでのチャンピオンに完勝する（2011 年）など、特定の狭い領域ではありますが人間のプロの能力に匹敵するようなソフトウェアが生まれてきています。このような状況に、「一線越えたか電脳の浸食」、「コンピュータは人間の領域を侵すか？」など、将来コンピュータが人間の仕事を奪うのではないかと、いったことが話題になる時代にまでなってきました。

今も続く「ソフトウェア危機」

商用コンピュータが次々と世に送り出されるようになった 1950 年頃から、コンピュータの利用領域がどんどん広がり、ソフトウェアに対する需要が拡大していききました。しかし増大するソフトウェアの需要に対して、人海戦術による手作りの域を出ないソフトウェア開発は生産性が低く、とても供給が追いつけない状況となり、いわゆる「ソフトウェア危機」が叫ばれました。「このままでは世界中の人類すべてがプログラマにならないと、ソフトウェアの供給が間に合わない」と言われるほどで、このような危機意識の下でソフトウェア工学が提唱されたのはご存じの通りです。

しかし、幸いなことに世界中の人がプログラマになることなく、今日の IT 時代を迎えています。人類のすべてがプログラマになるどころか、人間の仕事を奪うことを心配するような時代にまで劇的な進歩を遂げたわけです。

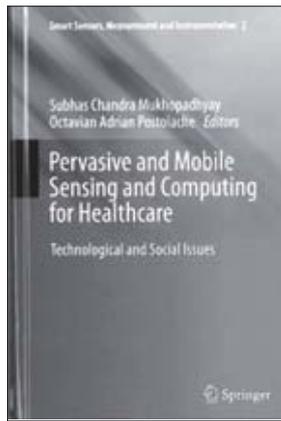
では、当時の「ソフトウェア危機」は解消されたのでしょうか？相変わらずソフトウェアの生産性の低さは大きな問題ですし、ソフトウェアの欠陥によって社会に大きな混乱をもたらす事故は無くなるどころかむしろ大きな問題になっています。残念ながら「ソフトウェア危機」は今もまだ続いているのです。先に述べたように人間の能力を凌駕するコンピュータが現実のものになってきている一方で、数十年前からの課題に対して抜本的な解決がなされていないのは残念なことです。

ゲームには勝っても……

人工知能分野の開拓者であるミンスキーは、「知の逆転」（NHK 出版新書）のインタビューの中で「ゲームには勝っても、人の役に立つ必要な仕事ができない」という趣旨の痛烈な批判をしています。人工知能分野は研究の方向を誤ったおかげで「失われた 30 年」となってしまった、とまで言っています。

確かに、現状でのブレークスルーはゲームの分野など非常に特殊な分野に限られていますが、それはきっと他の分野においても新たな突破口を開く大きな原動力となるはずで、例えば、機械学習の技術や自然言語を理解する技術、人の力では扱えない膨大なデータの中から必要な情報をうまく抽出する技術など、他の領域でも使える一般的な技術に洗練されることによって、多くの分野にブレークスルーをもたらすのではないのでしょうか。そして、ソフトウェアの分野もその一つになるはずで、ソフトウェアに関する膨大な情報、例えば設計書やソースコード、テストデータ、バグ履歴や開発管理のための各種のデータ類など、ソフトウェアに関連するデータはほとんどコンピュータに格納されています。世界中のソフトウェアに関する膨大なデータの蓄積を背景にして、コンピュータの能力を最大限に活用すれば、ソフトウェア開発の在り方に大きな変革をもたらすのではないかと大いに期待しています。

ロボットがロボットを作っている現代です。次の時代には、コンピュータがコンピュータソフトウェアを作ること期待したいものです。



Pervasive and Mobile Sensing and Computing for Healthcare

Subhas Chandra Mukhopadhyay,
Octavian A. Postolache 編集

ISBN : 978-3-642-32537-3

Springer 刊

B5 版・361 頁

定価 24.95 ユーロ (Amazon: 16,699 円)

2012 年 10 月刊

医療分野の IT 融合化に備える技術

医療分野に関する世界の市場成長率は 2001 年以降 8.7% に及び、またアベノミクスの成長戦略でも医療・健康が重点分野のトップとして挙げられている。

とりわけセンサーと IT の技術進歩により、新しい医療サービスの創出は期待が大きい。例えば、これまで医師一人に多数の患者が診断を受ける従来型サービスに対して、医師よりも患者が中心となり、患者は何時でも何処で医療を受けることができる形態のサービス PCMH (Patient-Centered Medical Home) がある。この外、患者の健康状態を記録するサービスである PHR(Personal Health Record) など、IT 融合化によるサービス創出も検討されている。これから人口の高齢化などが後押し、このような医療分野での IT 融合化の流れがますます加速されることが期待される。

本著では、このような新サービスを実現するためのセンサーと IT に関する最新技術を紹介する論文を選定・収録している (Mukhopadhyay 教授と Postolache 教授による編集)。

IT に関しては、その障害がこの種のサービスに多大な影響を与えうることから、「IT に関するサービスのディペンダビリティ(頼れる度合い)」を統制し、これを高めてゆくための技術が本著でも取り上げられた。従来のハードウェア・システムなど IT サービスの個別構成要素の次元で論じられてきたディペンダビリティについて IT サービス全体を俯瞰してとらえ、これをモデル化する技術についても本著で述べている (該当章は SEC と早大共著論文)。

この分野の IT 融合化技術や社会経済活動に係る IT サービスの信頼性技術に興味を持たれる方はご一読頂くよう推薦する。
(大高 浩)

『ついにポンチ絵の教本、見つけました！』



デザイン仕事に必ず役立つ 図解力アップドリル

原田 泰 著

ISBN-10 : 4862670903

ISBN-13 : 978-4862670908

ワークスコーポレーション刊

単行本・184 頁

定価 2,625 円 (税込)

2010 年 9 月刊

ソフトウェア開発に求められるスキルは、プログラミング、マネジメント、論理的思考、コミュニケーションなど多岐にわたる。開発現場に求めるスキルを聞くと『ポンチ絵』という回答があった。最近モデリングツールが普及したことで、ポンチ絵レベルで考え、議論することが欠如しているとの話も聞く。

本書はソフトウェア開発者向けの書籍ではない。デザインの仕事に関わる人に向けた図解の書籍である。近年、デザイン指向などデザインの重要性が認知されているので、ソフトウェア開発との親和性も高い。

ソフトウェア開発においてポンチ絵が求められる局面は多い。ホワイトボードを使った会議、仕様書における図の作成、プレゼンテーションによる説明など、ポンチ絵の有無やレベルによって大きく成果が変わってくる。また海外エンジニアとの調整においても威力を発揮する。

ポンチ絵は情報処理の基本とも言える。伝えたい、伝えるべき多くの情報を、受け手の理解を想定してまとめるのがポンチ絵である。この書評もポンチ絵で描いて発信したいくらいである。

本書はドリルであり、まず絵を描くことを求める。PC を離れ、童心にかえり、大胆に太めのペンで絵を描く。点・線・面を使った伝え方のセオリーも多く紹介されており、図解のセオリーを学ぶこともできる。

最終的にはイラストやプレゼンテーション作成ソフトウェアで、キレイに整形されるが、ポンチ絵スキルが情報の質を左右する。このポンチ絵の質がソフトウェア開発にも大きな影響を与えることを認識してほしい。本書はドリルなので、若手技術者はどんどん問題にチャレンジしてスキルアップして欲しい。

(渡辺 登)

編集後記

今回発刊の SEC journal は、2012 年度 SEC 成果の紹介号となっています。

第二期中計が本年 3 月をもって終了しました。SEC は果たすべきミッションを見直し、第三期以降に新たな取り組みを行なっていくことに対応して 2013 年 6 月 1 日より SEC の組織体制も改めました。それにもなって SEC の呼称は、ソフトウェア・エンジニアリング・センターからソフトウェア高信頼化センターに変わりました。

SEC の第二期までの活動と SEC が今後どういう方向を目指して活動していくかについては、「SEC における第二期中期計画の総括と第三期中期計画の概要について～ソフトウェア開発の生産性向上からソフトウェア高信頼化へ向けて～」の記事で紹介しています。

さて、今回の所長対談では、「真に利用者に求められるソフトウェアの提供とそれを実現するエンジニア像について考える」というテーマで CTC 伊藤忠テクノソリューションズ常務加藤様から、IT システムの開発から運用までを取り組む中での課題、エンジニアに求められるスキルが変化すること、インフォメーションのインテリジェンスへの昇格、インスピレーションの源泉は現場、などについて忌憚のないお話をうかがうことができました。

(h-tanaka)

編集部より

2012 年度の SEC 成果に関連する記事の執筆者の所属・役職などは、2013 年 3 月末時点のもので記載させていただきました。

次世代のソフトウェア・エンジニアリング等に関して、忌憚のない意見をお待ちしております。下記の FAX またはメールにてご連絡ください。

SEC journal 編集部 FAX : 03-5978-7517 e-mail : sec-journal_customer@ipa.go.jp

SEC journal 編集委員会

| | |
|--------------|-------|
| 編集委員長 | 田中秀明 |
| 編集委員 (50 音順) | 石川智 |
| | 遠藤和弥 |
| | 杉浦秀明 |
| | 杉原井康男 |
| | 中川明美 |
| | 中村雄三 |
| | 松田雅幸 |
| | 三原幸博 |
| | 室修治 |
| | 山下博之 |



香川県栗林公園にて

(撮影 : h-tanaka)

SEC journal® 第9巻第2号 (通算 35 号) 2013 年 7 月 31 日発行

© 独立行政法人情報処理推進機構 2013

編集兼発行人 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
〒 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
Tel : 03-5978-7543 Fax : 03-5978-7517
URL : <http://www.ipa.go.jp/sec/>
e-mail : sec-journal_customer@ipa.go.jp

※本誌は「著作権法」によって、著作権等の権利が保護されている著作物です。

※本誌に掲載されている会社名・製品名は、一般に各社の商標または登録商標です。

SEC journal 論文募集

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では、下記の内容で論文を募集しています。

論文テーマ

- ・ソフトウェア開発現場のソフトウェア・エンジニアリングをメインテーマとした実証論文または先導的な論文
- ・ソフトウェアが経済社会にもたらす革新的効果に関する実証論文

論文分野

品質向上・高品質化技術、レビュー・インスペクション手法、コーディング手法、テスト/検証技術、要求獲得・分析技術、ユーザビリティ技術、プロジェクト・マネジメント技術、設計手法・設計言語、支援ツール・開発環境、技術者スキル標準、キャリア開発、技術者教育、人材育成、組織経営、イノベーション

応募要項

締切り : 1月・4月・7月・11月 各月末日

査読結果: 締切り後、約1カ月で通知。「採録」と判定された論文はSEC journalに掲載されます。

応募方法: 投稿は随時受付けております。応募様式など詳しくはHPをご覧ください。

<http://www.ipa.go.jp/sec/secjournal/papers.html>

ITパスポート試験のご案内

ー ビジネスにITを活用する すべての社会人のための「国家試験」ー

- ビジネスにITを活用するためには、情報システム部門に限らず、利用する側の社員一人ひとりにも“IT力”が求められています。
- iパス（ITパスポート試験）は、セキュリティ、ネットワーク等のITに関する基礎知識をはじめ、企業活動、経営戦略、会計や法務、プロジェクトマネジメントなど、幅広い総合的知識を測る国家試験です。
- iパスを通じて、社員一人ひとりに“IT力”が備わることにより、組織全体の“IT力”が向上し、様々なメリットが期待されます。

iパスのメリット

ITを活用した業務効率化とビジネス拡大に！

iパスを通じて習得したITの基礎知識を活かすことで、業務にITを積極的に活用し、業務効率化につながります。また、ITに関する基礎知識は、社内の情報システム部門等との円滑なコミュニケーションにも役立ちます。営業職であれば、顧客に対して製品やサービスを具体的にわかりやすく説明できるようになり、顧客のニーズをより深く把握できるようになり、ビジネスチャンスの拡大にもつながります。

情報セキュリティ対策・コンプライアンス強化に！

社員一人ひとりが、情報セキュリティやモラルに関する正しい知識を身につけ、意識することで、情報セキュリティに関する被害を未然に防ぐことができ、「情報漏えい」などのリスク軽減、企業内のコンプライアンス向上・法令順守に貢献します。

経営全般に関する知識など幅広い知識がバランスよく習得できる！

iパスは、ITに関する知識にとどまらず、企業活動、経営戦略、会計や法令など、ITを活用する上で前提となる幅広い知識がバランスよく習得できます。そうした知識が身につくことにより、業務の課題把握と、ITを活用した課題解決力が備わり、組織全体の業務改善につながります。

詳しくは、iパス Web サイトをご覧ください。<https://www3.jitec.ipa.go.jp/JitesCbt/index.html>
※企業の活用事例、企業の声、合格者の声など魅力的なコンテンツがご覧になれます。

IPA 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

SEC journal No.33
第9巻第2号(通巻35号)
2013年7月31日発行

© 独立行政法人情報処理推進機構

ISSN 1349-8622

