

SEC[®]

journal

Software Engineering Center

16

巻頭言

松田 晃一 IPA/SEC 所長

所長対談

高度ICT時代における サービス創造と人材育成の方策を考える

宇治 則孝 日本電信電話株式会社 代表取締役 副社長

第1特集

本格的普及フェーズに入ったETSS

- 1 ETSSの普及と運用の状況
- 2 ITベンダによるETSS適用事例報告 ETSSを活用したスキルマネジメント
- 3 富士ゼロックスにおけるETSSの実践的取り組み
- 4 教育効果測定に関するETSSの取り組み

第2特集

ソフトウェア開発プロジェクト 計測プラットフォームEPMの今

各国で活発化するインプロセス計測の動向

技術解説

ソフトウェア定量的品質コントロールの提案
『組込みソフトウェア開発向け品質作り込みガイド ESQR』の紹介

ソフトウェアに関する論文の書き方 主に開発現場に携わる人のために

井上 克郎 大阪大学大学院 教授

論文

生産性要因に基づいて層別されたソフトウェア開発工数見積りモデル

角田 雅照, 門田 暁人, 松本 健一, 高橋 昭彦

特別寄稿

21世紀のパラダイムシフトが、
日本の復活につながるイノベーションではないだろうか

鶴保 征城 IPA 顧問

海外レポート

NASSCOM Quality Summit 2008 参加報告

立石 譲二, 新谷 勝利

組織紹介

ITA

IPA[®]独立行政法人 情報処理推進機構
<http://www.ipa.go.jp/>



SEC journal

Software Engineering Center
No.16目次

- 1 巻頭言
松田 晃一 IPA/SEC所長
- 2 所長対談：宇治 則孝 日本電信電話株式会社 代表取締役 副社長
高度ICT時代における
サービス創造と人材育成の方策を考える
- 第1特集
- 6 本格的普及フェーズに入ったETSS
- 1 1 ETSSの普及と運用の状況
室 修治
- 10 2 ITベンダによるETSS適用事例報告 ETSSを活用したスキルマネジメント
ITA スキルマネジメントワーキンググループ
- 22 3 富士ゼロックスにおけるETSSの実践的取り組み
富士ゼロックス株式会社
- 35 4 教育効果測定に関するETSSの取り組み
渡辺 登
- 第2特集
- 38 ソフトウェア開発プロジェクト
計測プラットフォームEPMの今
各国で活発化するインプロセス計測の動向
神谷 芳樹, 吉川 宏幸, 樋口 登
- 技術解説
- 46 ソフトウェア定量的品質コントロールの提案
『組込みソフトウェア開発向け 品質作り込みガイド ESQR』の紹介
吉澤 智美
- 51 ソフトウェアに関する論文の書き方
主に開発現場に携わる人のために
井上 克郎 大阪大学大学院 教授
- 論文
- 58 生産性要因に基づいて層別された
ソフトウェア開発工数見積りモデル
角田 雅照, 門田 暁人, 松本 健一, 高橋 昭彦
- 特別寄稿
- 68 21世紀のパラダイムシフトが、
日本の復活につながるイノベーションではないだろうか
鶴保 征城 IPA顧問
- 海外レポート
- 70 NASSCOM Quality Summit 2008参加報告
立石 譲二, 新谷 勝利
- 組織紹介
- 72 ITA
Information Technology Alliance
矢野 達也 ITA SE会議 H20年度幹事
- 74 BOOK REVIEW
- 75 用語解説
編集後記
SEC案内

「ハドソン川の奇跡」とディペンダビリティ



IPA(独立行政法人 情報処理推進機構)
SEQ(ソフトウェア・エンジニアリング・センター)
所長

松田 晃一

昨年(2008年)2月より情報処理推進機構のIT人材育成本部長としてIT人材育成の諸施策推進を担当していましたが、今年1月にソフトウェア・エンジニアリング・センター所長に就任致しました。どうぞよろしくお願い致します。

ソフトウェア・エンジニアリング・センターは、創立以来4年余りにわたり多くの成果を生み出し日本のソフトウェア工学の拠点としての役割を果たしてきました。しかし、技術の発展、取り巻く環境の変化は激しく、それらに対して不断の対応が必要です。これまでの成果を一層発展させると共に、技術の進展や社会の変化に対応した新たな課題にも挑戦し、我が国のソフトウェア産業競争力の強化に貢献していきたいと考えています。

ハドソン川の奇跡

ところで、正月早々に飛び込んできた航空機事故のニュースは、まだ記憶に新しいところです。

ハイテクの塊である最先端の航空機が、鳥の一撃で墜落するほどの致命的なダメージを受けたことは近代文明の脆さを改めて実感させられました。一方、機長の冷静沈着な判断と行動によって、1人の犠牲者も出さずハドソン川への不時着水に成功したことは、まさに「奇跡」と呼ぶにふさわしい出来事でした。そして、大きなシステム、複雑なシステムであればあるほど、技術だけを過信するのではなく、人間系も含めたトータルなシステムとして捉えて安全設計することの重要性を思い知らされた事故だったと思います。

SECでは、社会の情報基盤として、ますます重要性を増しているITシステムの安全の確保について、重点課題として取

り組んでいますが、その検討に当たっては技術の側面だけではなく、人間系もシステムを構成する1つの重要な要素と考
えて問題を捉え直す必要があると強く感じた出来事でした。

ディペンダビリティと信頼性

ITシステムの高信頼性に関する議論は、フォールト・トレラント・コンピューティング(FTC)の分野を中心に、過去から続けられてきたことはご承知の通りです。最近では「ディペンダビリティ」という用語が使われるようになってきています(といっても、この用語の提案はかなり前で1985年が最初だったようです)

「ディペンダビリティ」とは、「様々なアクシデントがあったとしても、システムが提供するサービスを、利用者が許容できるレベルで維持すること」を言うようです。様々なアクシデントには、自然災害や機器の故障、設計、製造、運用時における人間のミス、さらには悪意をもった攻撃等も含まれます。これまでの信頼性に比べ「利用者の視点」により重点が置かれ、「人間系」を含んで広範に対象を捉える概念であるということが出来ます。

「頼りになる」ITシステムにむけて

私たちの生活に深く組み込まれたITシステムは、一旦事故を起こすとこれまでとは比べ物にならない大きな影響を生活に及ぼすこととなります。さらに、近年は家電製品や自動車など一般の消費者が直接利用する製品の多くがソフトウェアで制御されるにつれ、その事故の影響は大きくかつ深刻さを増してきます。私たち市民がいつでも、安心してサービスを受けられる頼れるシステム、ディペンダブルなシステムが強く求められる所以です。

SECは、人間系も含めたシステム全体のディペンダビリティ確保のための手法の開発を目指し、社会の安心・安全の確保に貢献していきたいと思っています。

高度ICT時代における サービス創造と人材育成の方策を考える

ブロードバンド・ユビキタス社会の新しいネットワークとして日本電信電話株式会社(NTT)がサービスを開始したNGN(Next Generation Network)に大きな期待が寄せられている。NGNを基盤とした高度ICT時代における多様なサービスを創造していくための方法や人材育成のありかたについてNTTの代表取締役副社長の宇治則孝氏とIPA/SECの松田晃一所長が語り合った。

日本電信電話株式会社 代表取締役 副社長
宇治 則孝



SEC所長
松田 晃一

松田 NGNという言葉は皆さん既にご存じのことだと思のですが、そもそもNGNはどのようなネットワークなのか、とくに今までのネットワークに対比してNGNはどのようなものなのかについて、まずお話しいただけませんか。

宇治 今、急速に新しいネットワークの使い方が広がり、トラフィックも急激に増えている状況です。その状況を踏まえて、インターネット技術、IP技術をベースにしたネットワークを構築することが我々キャリアの側にとっても、利用者にとっても大きなメリットがあると考え、NGNを構想しました。今までは、電話網があり、またIP電話もあるというように個別のネットワークがそれぞれ存在していたのですが、それらを1つに統合し、かつインターネットよりセキュリティが高く信頼性の高いネットワークを作ることが新しい時代のネットワークとしてふさわしいと考え、NGNを構築したのです。世界のキャリアもIP技術をベースにしたネットワークを構築する方向で動いています。

松田 これまでの電話をつなぐネットワークとNGNには大きな違いがあるようですが、既に提供されているインターネットを用いたサービスとNGNの違いは何でしょうか。両方ともIP技術を使ったネットワークのようですが。

宇治 NGNは、これまでよりも安全で信頼性の高いネットワークです。例えば、インターネットでは「なりすまし」というセキュリティ上の問題があります。NGNには、きちんとアクセス回線の識別をする機能があるので、そうした問題を防ぐことが出来ます。また、インターネットはトラフィックが集中すると不安定になることがあります。その点、NGNはインターネットと違って、よく管理されたネットワークなので、非常に高い信頼性を実現しています。ですから、ネットワークを通じて映像を見るということにNGNは非常に有効だと思います。



松田 晃一(まつだ こういち)
1970年京都大学大学院修士課程修了後、日本電信電話公社入社。NTTソフトウェア研究所 ソフトウェア開発技術研究部長、株式会社国際電気通信基礎技術研究所(ATR)取締役企画部長、NTTコミュニケーション科学研究所 所長、NTT先端技術総合研究所所長、NTTアドバンステクノロジー株式会社代表取締役常務、NTT-AT IPシェアリング株式会社代表取締役社長を歴任し、2008年2月IPA(独立行政法人情報処理推進機構)IT人材育成本部長に就任、2009年1月よりSEC(ソフトウェア・エンジニアリング・センター)所長に就任。

松田 インターネットで動画を見ると、ときどき画面がぎくしゃくした動きになりますが、NGNになるとそういう問題が改善される訳ですね。

宇治 インターネットで多くのビジネスが展開されていますが、多くの利用者がセキュリティを気にしています。NGNの大きなポイントは高いセキュリティであり、ビジネスユースであれホームユースであれ、人々の生活や社会を豊かにする多様なサー

ビスが提供されることを大変期待しています。

松田 国際的にもNGNと同じ方向へ向かっているということですが、国際的な動きの中で日本の取り組みははどのようなものですか。

宇治 ITU¹という標準化機構で議論した国際標準を具体的にサービス開始したという意味で、日本は世界で初めての国になると思います。NGNの一部のサービスを提供している国やキャリアはあるのですが、映像まで含めたフルスペックでサービスを開始しているのは日本のNTTだけでしょう。

松田 今までの電話網は徐々にNGNに置き換わっていく、あるいはNTTの地域IP網もNGNに移していくという計画ですか。



宇治 則孝(うじのりたか)
1973年、京都大学大学院工学研究科修士課程修了。
同年、日本電信電話公社入社。総裁室企画室にてNTT
民営化関連業務、新規事業開発業務を担当後、1988
年NTTデータに異動。新世代情報サービス事業本部
長、経営企画部長、法人分野の事業本部長等を経て、
2005年代表取締役常務に就任。2007年より日本電信
電話株式会社の代表取締役副社長となり現在に至る。
『進化する企業のしくみ』(PHPビジネス新書)共著。

宇治 そのように計画しています。地域IP網もIP技術を使ったネットワークですが、2012年度には、NGNへのマイグレーションを完了する予定です。更に電話網もNGNに移行させようと考えていますが、移行時期はもう少しばら検討してから、アナウンスしようと思っています。

松田 固定網と無線網を融合させるFMC²というキーワードを聞いたことがあります。こちらはどのように進んでいるの

ですか。

宇治 FMCについてはいろいろなイメージがあります。NGNが日本中をカバー出来る2010年くらいに、スーパー3Gという、やはり非常に高速なモバイル網が登場します。そうなるとブロードバンドの固定とモバイルという2つのネットワークがシームレスにつながります。そうすると、本当の意味で固定とモバイルのネットワークをうまく使った新しいサービスが登場してくると期待しています。

オープンとコラボレーションを キーワードに多様なサービスを創造

松田 その新しいネットワークを使ってどのようなサービスが提供されるのでしょうか。その点が重要な問題だと思います。

宇治 おっしゃるとおりです。今、非常に高速で大量のデータを低料金で利用出来る環境は、日本が世界のトップクラスになっています。それを使ったサービスとしてどのようなものが考えられるかという、1つは最近話題になっているSaaS³です。SaaSはネットワークを通じてソフトウェアを利用することです。SaaSで大切なことは、ネットワークにどこまで信頼性があるか、高いセキュリティがあるかということです。安全で信頼性が高いという特徴を持つNGNのアプリケーションとしてSaaSは非常に大きいものだと思っています。

また、NGNは非常に品質の高い映像を送れることも大きな特徴です。テレビ電話やテレビ会議、テレプレゼンスといろいろな言い方がありますが、そういう映像を利用したコミュニケーションサービスがすごく伸びるのではないかと考えています。今までも携帯電話でテレビ電話としての使い方が出来るのですが、新しいネットワークでは、より自然で使い勝手が格段に良くなります。屋外の広告もブロードバンド回線で映像や宣伝を流す電子看板(デジタルサイネージ)も伸びるでしょう。ビジネスフィールドではそういうアプリケーションがあると思っています。

松田 NGNにより、家庭で見るテレビの見方も変わってくるでしょう。すべての映像データがデジタルになると、アンテナではなく光ファイバを通じて見るIPTVが広がると期待されますが、NGNらしさということで言うと、自分の見たい番組・コンテンツを好きな時間に見るビデオ・オン・デマンドも今以上に便利になるでしょう。

NGNの普及によって多種多様なアプリケーション、サービスの登場が期待されますが、問題はそれをどう作るかですね。

宇治 NTTだけで作ることは出来ません。我々は、オープンとコラボレーションをキーワードとしてアプリケーションサイドの人

1 ITU : 国際電気通信連合, International Telecommunication Union
2 FMC : Fixed Mobile Convergence

3 SaaS : Software as a Service

たちとディスカッションして進めていこうと考えています。SaaSを例にすれば、いろいろなソフトウェアベンダさんとコラボレーションして新しいサービスを創造していくということです。

松田 確か、Microsoft社と協業されるという報道がありましたね。

宇治 その件もコラボレーションの一環です。他にも、医療や健康、教育等の業界の人とコラボレーションする必要があり、IPTVでは放送局と色々な話をさせてもらっています。また、コンテンツ制作会社がネットワークを通じてみなさんに見てもらおう等、新しい事業機会、新しい産業展開が生まれるのではと思っています。

松田 NTTデータとNTTコミュニケーションズが一緒になってSaaSの基盤を作るという報道がありましたね。かつては考えられないことでした。

宇治 それも1つの時代の流れでしょう。両社共にそういう基盤を作ろうと考えていたことは事実です。その基盤の中で、NTTコミュニケーションズはネットワークに近い部分にノウハウを持っているし、NTTデータはアプリケーションに近い部分にノウハウを持っています。その両方の強いところを併せ持つことによってよりよいサービスを提供出来るプラットフォームを作れるのではないかと考え、2社にプラスして研究所の技術を加味し、NTTデータ、NTTコミュニケーションズ、研究所がプロジェクトを組んで進めているところです。

松田 一方で、NTTだけで独占してしまうのでは、という懸念の声はないでしょうか。

宇治 そういうふうにとらえることはおかしいと思っています。日本企業は世界を見据えた競争をすべきです。現在、GoogleやAmazonは世界中で事業を展開しています。グローバル視野の競争において日本のパワーを強化するためにはいろいろなコラボレーションが必要となります。NTTグループ内のコラボレーションもその一環ですが、場合によっては海外のネット企業や国内の企業等ともコラボレーションするということによって、新しい時代に向けた社会基盤を作っていくことが求められています。

松田 確かにそうですね。国内のコップの中の競争をしている場合ではありませんね。プラットフォームの上のアプリケーションにいろいろなプレイヤーに入ってもらい、新しいサービスを作っていくというアプローチが有効ですね。

宇治 ネットワークがあってその上にプラットフォームがあり、プラットフォームの上にアプリケーションあるという構造の中で、顧客から見ると最も重要なのはアプリケーションです。プラットフォームの上でアプリケーションがどんどん出てきて、多くの選択肢が存在することが利用者から見ると望ましいと思います。

松田 なる程、NTTは中期経営戦略で「サービス創造グループ」を目指す、と言っておられますが、その意図はこういうことだったのですね。ところで、魅力あるサービスを作る上ではソフトウェアの力が重要だと思います。その点についてどのように考えられていますか。

宇治 これからのICT時代においてソフトウェアが非常に重要だと思っています。サービス提供型のソフトウェアを開発するためには、大きな投資が必要となるでしょう。そこで、みんなが使うものだから投資を負担し合ったり、みんなで使い勝手を進化させていくというアプローチが必要ではないでしょうか。また、プラットフォームを作るに当たっても、SaaSのためのプラットフォームやIPTVのプラットフォームを個別に作るのではなく、プラットフォームを作っていくための共通基盤をうまく作れるといいなと研究を推進しています。

松田 サービスをどんどん作っていくと、ソフトウェアの量が無限に近くなりますよね。それでは大変だから、共通化出来るところは出来るだけ共通化していこうという考え方でですね。

宇治 おっしゃるとおりですね。

安全で安心なシステム開発の手法を確立することが急務

松田 世の中のソフトウェア需要はすごい勢いで増えています。コンピュータで動くソフトウェアだけでなく、組込みソフトウェアの量もそうです。携帯電話には昔のバンキングシステム並みの量のソフトが入っています。「ソフトウェアの爆発」ということが起こっているわけです。このように、ソフトウェアが巨大になり複雑になると事故が起きる危険性が高まるという問題があります。

宇治 情報システムが社会インフラを支える時代になっているので、ネットワークの信頼性を含めてトータルなシステムの信頼性、ソフトウェアの信頼性を確保することは極めて重要なテーマとなっています。その点について、SECも議論を重ねていると聞いています。

松田 SECでソフトウェアの信頼性の問題すべてを扱うわけに



はいかないので、まず最も大切な重要インフラから着手して、そこから何らかの答えを出したいと考えています。そこで得られた技術が普及していくと、全体に波及するのではないかと考えています。企業が互いに経験を共有することがシステムの信頼度を上げることに有効なことは間違いありません。既にそういう活動を立ち上げているのですが、今後、さらに加速させたいと考えているところです。

宇治 安全で安心なシステムをきちんと作っていくための手法や考え方は非常に重要です。事例をベースに新しい考え方を展開していくこともいいのでしょう。システムの信頼性については、二重化する等いろいろと工夫されてきていますが、ここまでシステムが巨大になり、かつ人間が作るものなのでまったくどこにも問題がないということはないでしょう。そこで、何かあったときに素早くリカバリする仕方というかコンティジェンシープラン的なものがきちんと作成されることが必要だと思います。

松田 そうですね。そうしたことに加えて、電電会社の時代より「24時間365日サービスを中断しない」という、NITが蓄積した素晴らしい技術やノウハウを教えてくださいなと思っています。

宇治 今、ネットワークを構成している機器が電電会社のときから大幅に変わっています。ネットワークはソフトウェア・ディペンデントになっていますが、電話のネットワークの特性と大規模システムとしてのマネジメントの両方を兼ね備えた展開をしているので、その辺りのノウハウをいろいろと共有出来るでしょうね。

松田 NITデータでは全銀システムをはじめ、大規模なシステムを手がけていますね。そういう大規模な情報システムとNITのネットワークシステムに関して、お持ちの信頼性に関するノウハウの共有が促進されると全体の信頼性も高まると期待しています。

宇治 ソフトウェアは経済活動や社会生活になくてはならない基盤となっており、経済活動の効率化や社会生活の利便性向上に大きな役割を果たしていますが、いったん事故を起こすと大きな影響を社会に及ぼすこととなります。安全で安心なシステムを開発する手法の確立は急務であり、手法のブラッシュアップやプロセス改善に向けて産学官横断的な活動を推進しているSECの果たす役割は大きいと認識しています。

高度ICT時代に重要な人材育成

松田 優れたソフトウェアを効率的に開発するためには開発手

法の確立やプロセス改善と共に、人材育成が大切だと考えています。技術を使うのは人です。最後は人の問題に行き着くと思うのですが、その点についていかがお考えですか。

宇治 情報システムを使いこなす側の技術と、いい情報システムを作る側の技術の両方が求められていると思いますね。その意味で、この4月から始まるITパスポート試験の趣旨は大変よいものです。

松田 以前はITというと専門家だけが知っていればいいという時代でしたが、今や全国民が社会に出るためのパスポートとして、一定の知識を持っていないと生きていけないという感じになり始めていますね。

宇治 実は、私は情報処理技術者試験のシステムアナリストの第1回の試験の合格者なんです。

松田 そうなのですか。それはすごい。

宇治 システムアナリストの資格が出来たときにうれしくて受験して幸運にも合格しました。そういう資格はモチベーションになります。ですから、時代に合わせた形で新しい資格をきちんと作っていくと同時に、うまく展開していくことが高度ICT人材の育成に向けて本当に重要なことですね。

松田 おっしゃるとおりです。つい最近、情報処理学会のパネルディスカッションに出席して、高度ICT技術者の資格制度に関する議論を行ったのです。レベル1から、2、3までは情報処理技術者試験のITパスポート試験等、それぞれ試験によって資格を認定出来るようになります。しかし、レベル4から上の資格は試験だけでカバー出来ません。実績を評価しないとイケません。そういう高度なレベルの人材の資格制度をどうするかをテーマにしたパネルディスカッションです。そこで、皆が元気になるからぜひやろう、学会が中心になってやっていこうじゃないか、しかも国際的に通用する資格にしようじゃないかと議論が盛り上がりました。IFIPという情報処理学会の国際的な連合体があるのですが、情報処理学会が認定するとそのまま国際的な資格になるようにしたいということです。

宇治 そういう活動は本当に重要です。弁護士は尊敬され、待遇もいい。情報処理分野の技術者もそうなりたいですね。今、ICTを活用して社会的意義のある価値を創造するイノベーションが求められています。個人レベルでもICT人材が自発的に新たな領域にチャレンジする意義付けが必要でしょう。グローバルに通用する人材の育成・認定制度を作っていくことにIPA/SECの役割は大きいと期待しています。

松田 IPAとしてSECとしてそういう取り組みを進めていこうと考えています。どうもありがとうございました。

文：小林秀雄 写真：越昭三郎

本格的普及フェーズに入ったETSS

組込みスキル標準ETSS¹は、2005年にスキル基準、2006年にキャリア基準及び教育研修基準のVersion 1.0が策定・発表された。スキル基準が発表され約4年、本特集では、ETSSの普及と運用の状況、活用事例について紹介する。なお、本特集は以下の構成で展開する。

- 1 ETSSの普及と運用の状況
- 2 ITベンダによるETSS適用事例報告 ETSSを活用したスキルマネジメント
- 3 富士ゼロックスにおけるETSSの実践的取り組み
- 4 教育効果測定に関するETSSの取り組み

1 ETSSの普及と運用の状況

SEC 研究員
室 修治

1 普及状況

ETSSの普及状況を紹介するに先立ち、まず「組込み産業関連の政府施策として重要なこと」として経営者及び事業責任者に聞いた結果を図1に示すのでご覧いただきたい。

図1の政府施策の中で重要とされているものは「組込みソフトウェアの開発を担う人材の育成」であり、他を圧倒している。また「人材」をキーワードとする項目が上位5項目中3項目を占め、人材に関わる施策への期待の高さが依然として強いことが伺える。

このような環境の下、組込みソフトウェアの開発力強化のための人材育成と人材活用の実現を目指し策定されたETSSの導入状況は図2の通りである。

従業員5,000名超の、いわゆる大企業においては「導入済み」「導入検討中」で40%に達している。日本の産業構造から考えるとその影響力は大きく、裾野産業をはじめ全体への波及・拡大は確実と見込まれる。

図3はETSSの有効性の評価について年度ごとにまとめたグラフである。人材関連の項目について順調に推移し、一定の評価まで達していることが読み取れる。

以上のように導入が進むETSSであるが、そのフレームワークについてとくにフィードバックによる改訂が必要な状況は発生していない。実際に企業等に導入して効果を上げるためには個別の条件を加味して自分たちのスキル標準を策定する必要があるがSECが認識している範囲でフレームワークに対する不都合は報告されていない。

1 ETSS : Embedded Technology Skill Standards

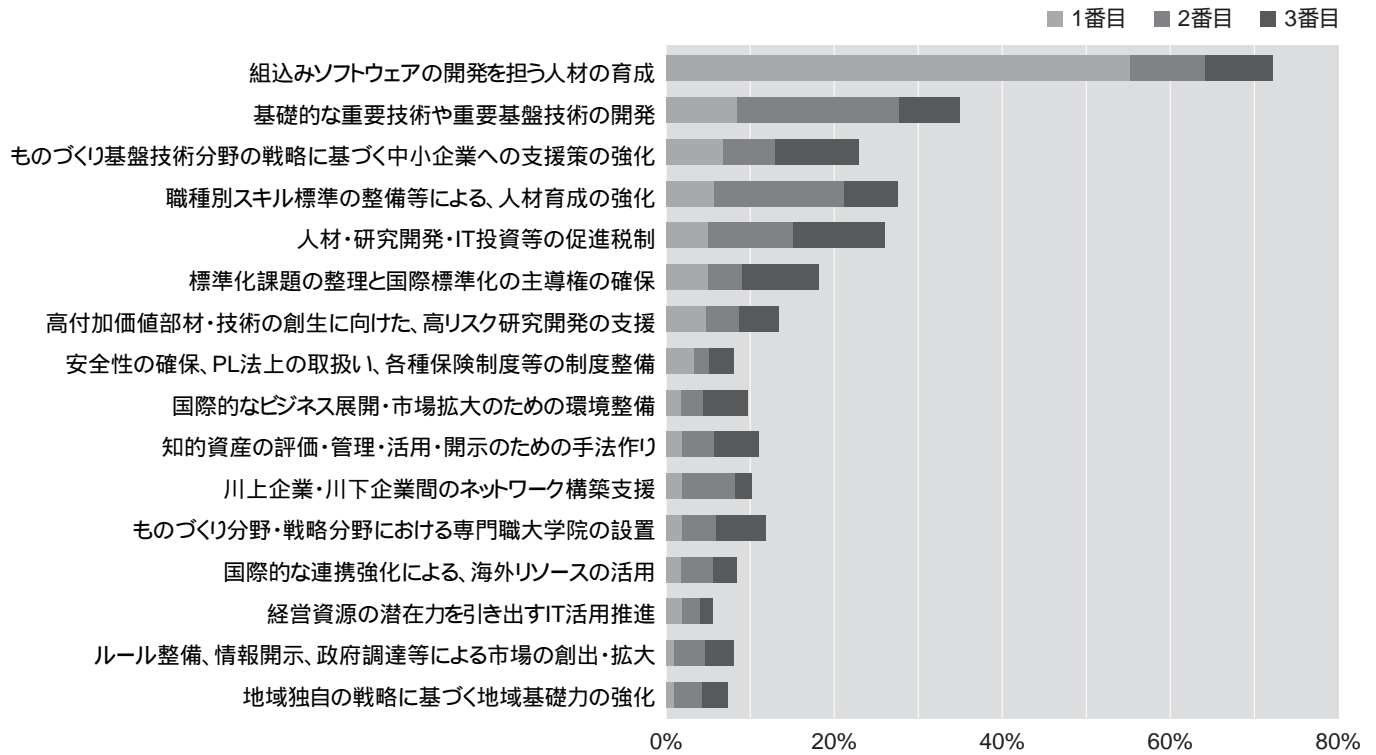


図1 組込み産業関連の政府施策として重要なこと (2008年版組込みソフトウェア産業実態調査より)

2 企業・団体での導入実績

SECはETSSの策定後、その有効性の確認及び企業・組織への導入ノウハウの開発、蓄積を目的に実証実験を行っている。また業界団体等でのスキル標準策定にもETSS的検証の立場から参加させていただいている。

独自に人材施策を実施する中でETSSを参照、比較対象とされている企業もあり図4のようなドメインの企業、団体等での導入実績が確認されている。

3 ETSSを利用した人材マネジメントサイクル

ETSSは組込みソフトウェアの開発力強化のための人材育成と人材活用の実現を目指し策定された。広い意味で使われる「人材マネジメント」の中の「育成」と「活用」の分野に使えるツールである。言い替えると「開発者・組織の保有している(あるいは必要とする)スキルの把握」と「教育」と言うことができる。人材を採用することや外部に求める場合も、可視化されたスキルがあれば適正な人材像を示すことができる。

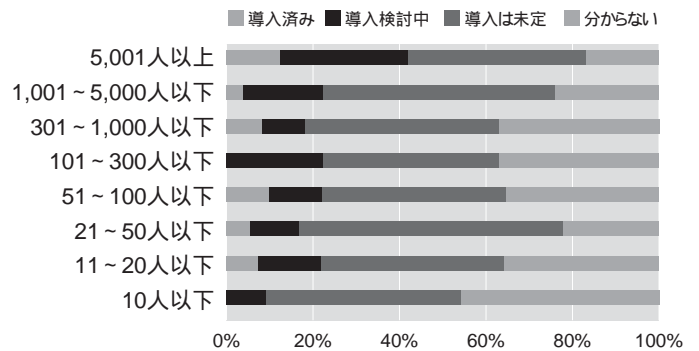


図2 従業員数別ETSS導入状況 (2008年版組込みソフトウェア産業実態調査より)

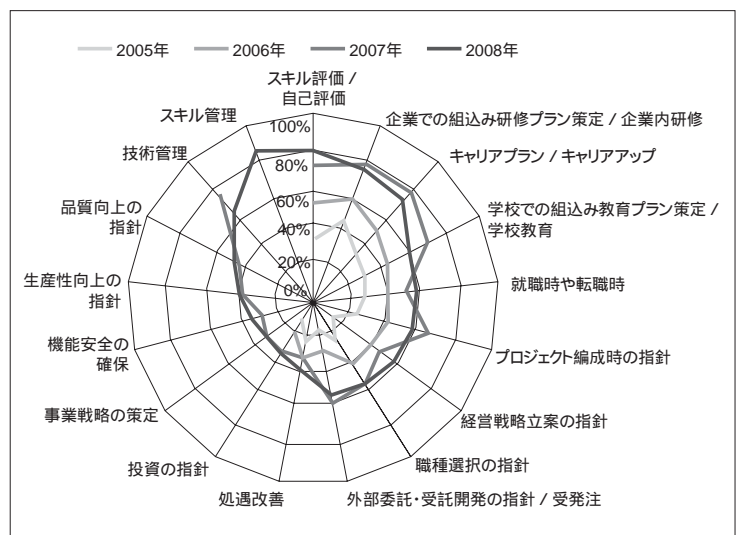


図3 ETSSの有効性について

ETSSの有効性について「有効」「非常に有効」の回答をグラフ化したもの。人材関連においては、順調に評価が上がっていることが分かる。



図4 ETSS導入実績のあるドメイン

4 スキル診断の重要性と課題

ETSSの利用に際して、非常に大きな比重を担うのが「スキル診断」である。スキル診断とはすなわち『自分たちの目的を達成するために必要なスキルを“可視化”することである。これにより 必要なスキルは何か、とそのスキルがどのくらい充足しているか、の2つが見えてくる。

スキル診断の大まかな流れは図6のようになるが、中でも重要なのが「スキルの洗い出し・定義」である。

スキルを洗い出すために、ETSSではスキルカテゴリごとに技術を体系的に整理出来る『スキルフレームワーク』というフレームを用意している。

フレームではスキルの粒度に合わせて階層分類され、更にスキル評価層となる構成になっている。この階層分類は、第1階層から第n階層までで組み込みソフトウェア開発の技術を詳細・具体化、最終階層でスキル評価が可能な具体的技術名称が定義出来る。

ETSSでは、第2階層までのスキルカテゴリを例示するに留め、下位階層を含むスキル項目の具体的定義は行っていない。実際に利用するためには企業や機関での定義が必要である。ETSS公開後ドメインごとの完成されたスキル基準を求める声も多かったがSECでは敢えてそこには踏み込まずにきた。詳細な部分のスキルの定義はドメインの価値そのものであり、絶えず発展・拡大(あるいは縮小・消滅)していくものでもある。この部分の策定はやはり専門性の高いところ、すなわち企業であり、業界であると現在も考えている。本号で事例を紹介いただく富士ゼロックス株式会社様、ITA²様の報告をぜひ参照されたい。

このようなスキルの定義により「必要なスキルの可視化」は実現出来た。では「求められるスキルの充足度」の可視化はどうであろう。図7スキルフレームワークの右側に「スキルレベル」とある。スキル評価すべき項目ごとにスキルレベルを測定することにより可視化が実現出来る。図8はスキル診断結果の例であるが組織が必要として定義されたスキルに対して充足しているか否かが一目で判断出来る。技術者ひとり1人が、開発チームが必要としている全スキルレベルに達している必要は必ずしもないであろう。その場合チーム全体のスキル診断結

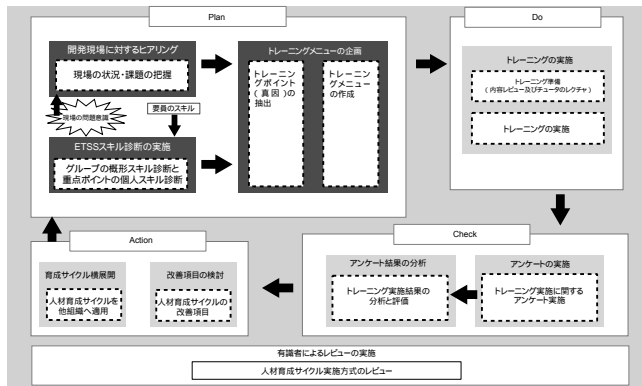


図5 ETSSを利用した人材マネジメントサイクル

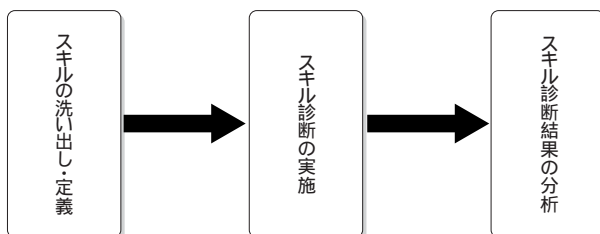


図6 スキル診断の流れ

図5にETSSを利用した人材マネジメントサイクルのイメージを示す。スキルを把握するために「スキル診断」を実施しその結果を基に「教育」計画を立て実施していく。教育の達成度評価のために再度「スキル診断」を行い次の教育サイクルへと進めていく。このようにETSSの利用ということを考えていくと、まず「スキル診断」が非常に大きな比重を持っていることが分かる。

2 ITA : Information Technology Alliance。「情報交換、共同研究活動等を行うと共に、各社が営業面、技術面で協力し合うことにより個々の能力を超えた事業展開を図る」ことを目的として結成。現在、会員会社16社、社員総数8,000名(ITAホームページより抜粋)

果の合計として充足度が評価出来ればよい。

ここで問題となるのがスキルレベル評価の精度となる。

精度のぶれの原因は以下の2つに集約出来る。

スキル定義の際に正しくスキルレベル評価出来るような基準を示しているか

人によって甘辛のばらつきが出る

ETSSでは当初より評価の指針として「評価文言」を例として示しているが、更に次のような工夫が必要である。

- 1) スキル項目内の要素をポイント化し、その合計ポイントで自動的にスキルレベルが決定出来るようにする。
- 2) スキル診断実施時に、スキル診断対象者に対して説明会を開き、診断基準の教育をする。

実施例の報告では、これらを試行することによって、

実施側・対象者側双方の理解・習熟度の向上が得られ、一定の効果があった。

また実際の診断結果をパブリックに活用出来れば、更に精度が向上される。今後はこのような仕組みの構築についても検討を開始している。

スキル定義、診断については人材の採用、調達、教育の実施等において自分たちの組織内だけではなく、対外的なインターフェースも必要であることを考慮し、業界横断的に見ても違和感が無いよう配慮しておくことも付け加えておく。

5 ETSS導入推進者の育成、認定

これまで述べてきたようにETSSの普及が進み、課題も見えてくる中、ETSSをより良く活用出来かつ企業間等での相互利用も促進出来るようETSS導入業務についての人材面の施策として「ETSS導入推進者」についての育成と認定を企画している。詳細はSEC journal 別冊ETSS特別号にて紹介出来る予定である。

6 国際的な広がり

ETSSは海外でも注目されてきている。2009年5月には「ETSS国際シンポジウム」(IPAX 05/26.27 東京)を開催し、海外からも講師を招きETSSへの期待等について講演いただく予定となっていることを最後にお伝えし、本稿を終えることとする。

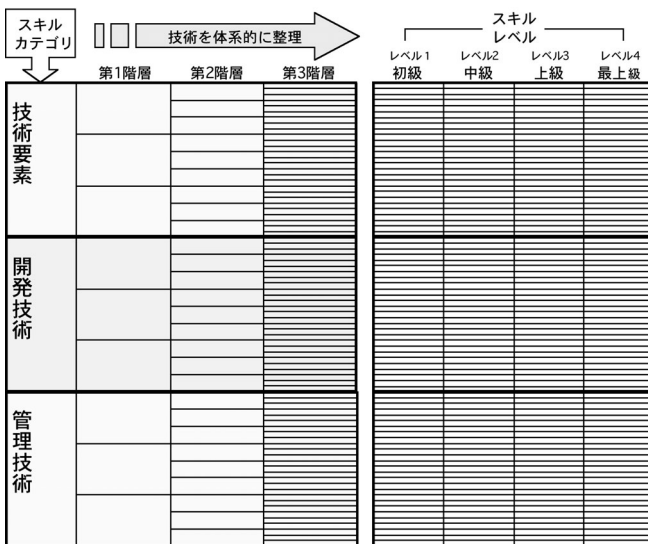


図7 スキルフレームワーク（組込みシステム開発一般例）

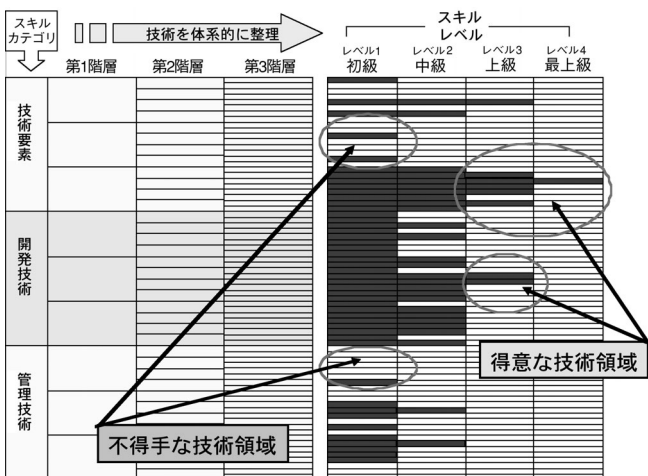


図8 スキル診断結果の例

2 ITベンダによる ETSS適用事例報告

ETSSを活用したスキルマネジメント

ITA スキルマネジメントWG

アイエクスナレッジ株式会社
中谷 則仁

株式会社アイネット
辻 敏尚・高松 茂

株式会社エヌアイディ
戸澤 秀行・若本 稔

情報技術開発株式会社
宮田 哲・田尻 誠・東谷 上

株式会社ソフトウェアコントロール
中安 猛・鈴木 潔孝

ナレッジビーンズ株式会社
土屋 崇

株式会社日本アドバンストシステム
隆宝 桂悟

株式会社ピッツ
押耳 正孝
(社名50音順)

独立系情報サービス企業16社で構成されているITA¹では、IPA/SECをはじめとするスキル標準関連団体とコラボレーションを図り、ETSSを活用したスキルマネジメント・システムの開発を推進している。ここではその活動を紹介します。

1 はじめに

独立系情報サービス企業の連合体であるITAは「1社ではできないことを、複数の企業が集まって実現させよう」という主旨の下、1995年2月に発足し、図1のように現在16社で構成されている。

ITAのソフトウェア・エンジニアリング会議(SE会議)がIPA/SECとのコラボレーション活動の主体であり、スキルマネジメントWG(MOS_WG²)はその下部組織に位置する。

MOS_WGは組込み(ET)系ビジネスを有するITベンダ8社13名から構成され、IPA/SECからは「ETSS推進テクニカルアドバイザー」としてWGに2名参画している。

メンバ構成は図2の通り。

また、スキル標準関連団体である「NPOスキル標準コ

ーザ協会(SSUG)・有限責任事業組合ITスキル研究フォーラム(iSRF)・社団法人組込みシステム技術協会(JASA)」から多数の有識者に参画いただき、現在では図3の組織構成でスキルマネジメント・システムの開発を推進している。

MOS_WGは2008年度に発足したWGであり、前身は2007年度に活動を開始した「ETSS・ETEC導入検討プロジェクト(ET2_PJ)」である。

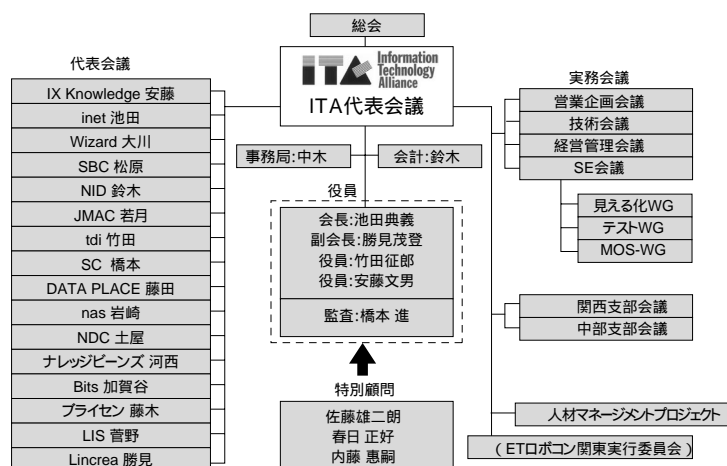


図1 2009年度のITA運営体制

1 ITA : Information Technology Alliance
2 MOS_WG : Management Of Skills Working Group

MOS_WG	
発起人	遠藤 和弥
参加メンバ	
IPA/SEC	渡辺 登 (石川 秀一)
アイエックス・ナレッジ株式会社 株式会社アイネット	中谷 則仁 辻 敏尚 高松 茂
株式会社エヌアイディ	戸澤 秀行 若本 稔
情報技術開発株式会社	宮田 哲 田尻 誠
株式会社ビッツ 株式会社ソフトウェアコントロール	押耳 正孝 中安 猛
ナレッジピーズ株式会社 株式会社日本アドバンスシステム	鈴木 潔孝 土屋 崇 隆宝 桂悟
リーダ	
情報技術開発株式会社	東谷 上

図2 MOS_WGメンバ構成



図3 スキルマネジメント・システム開発推進体制

2 導入検討プロジェクト (ET2_PJ) の背景

ET2_PJの開始は、経済産業省から発表された「高度IT人材の育成をめざして」に起因する。

ここにはグローバル・マーケットにおける各国ITエンジニア数の推移が2015年まで試算されている。2015年における日本のITエンジニア数は、中国やインド等の新興国にはるかに及ばない割合となり、今後は日本のITエンジニアの「数の優位性」は国際社会において重要視されないとの予測が明示されている。

当時のスキル標準は、IT系エンジニア向けであるITSS

が唯一の基準であり、「高度IT人材育成をめざして」でもITSSキャリアレベル4以上の高度IT人材の育成が急務であることを提言している(図4)。

これを受けて、ITAのET系ビジネスを有するITベンダ各社では「IT同様、高度ET人材を育成することは急務である」と考え、何とかETSSを有効活用し「ET系エンジニアを育成したい」との思いを募らせていた。この時点ではETSSに関する情報も少なく、ITSSとETSSの違いすら分からない状態であった。

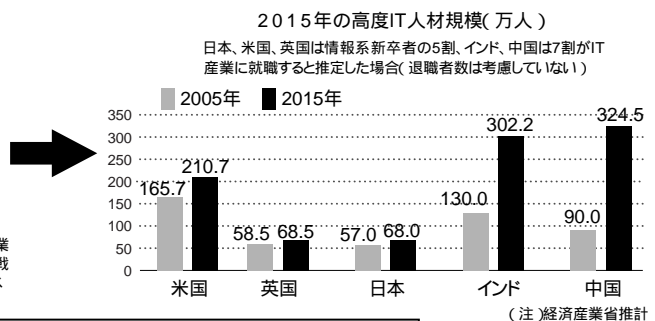
そのような状況の中、IPA/SECとのコラボレーションがSE会議で開始され、ETSSの有効活用を検討するためのプロジェクトが発足されるとの話を聞き、危機感を覚えていたITA各社がET2_PJに集まった。

経済産業省「高度IT人材の育成をめざして」

- ・高度IT人材についても、新興国が主要な供給源となる。とくに、インド、中国は、その大規模な人口プールからメガプレーヤとして登場するのは時間の問題と考えられる。
- ・従って、日本のIT人材をより高付加価値の職種にシフトすると共に、産業競争力維持の観点から世界のIT人材の有効活用を戦略に組み込むことが急務である。

	IT産業の就業者数 (万人)			情報工学系卒業生数(単年)
	2000年	2005年	伸び率	
米国	160.0	165.7	104%	9.0
英国	54.0	58.5	108%	2.0
日本	54.7	57.0	104%	2.2
インド	52.2	130.0	249%	24.6
中国	33.5	90.0	269%	33.5

出所: 米国・英国は株式会社野村総合研究所調べ、日本は経済産業省「特定サービス産業実態調査」、文部科学省「高度情報通信(IT)人材の育成に向けた文部科学省の基本戦略」、インドはNASSCOM「Press information note」、中国は情報サービス産業協会調べ



組込み系エンジニアの競争力(=スキル)強化は急務!

図4 ET2_PJ開始の背景

3 ET2_PJの活動内容

ET2_PJの立ち上げ時は、IPA/SECよりETSSの説明を十分受けた上で、ETSSの正確な理解に努めた。その後、ITAでETSSを有効に活用するため、ET2_PJの活動方針を次のように設定した。

ミッション

「ET系エンジニアのスキル強化」により「ITA企業の組織力」を高め「日本の国際競争力向上」に寄与する。

ビジョン

「ET系エンジニアの効率的なスキル強化」と「企業の組織力強化」を実現するETSS活用システムを構築する。

ストラテジ

「ETSSを有効活用」したITA版のETSS (ITA-ETSS) を策定する。

ET2_PJではこの活動方針を元に、ITA-ETSSを策定するための具体的なモデル化プロセスを図5のように定義した。

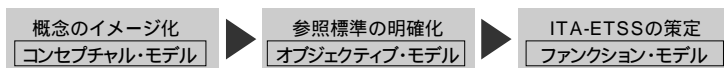


図5 ITA-ETSS策定モデル化プロセス

3.1 コンセプトチャル・モデル

最初に、ITAにおけるETSSの具体的な活用場面を想定し、概念レベルのモデルとして「コンセプトチャル・モデル」を策定した(図6)。

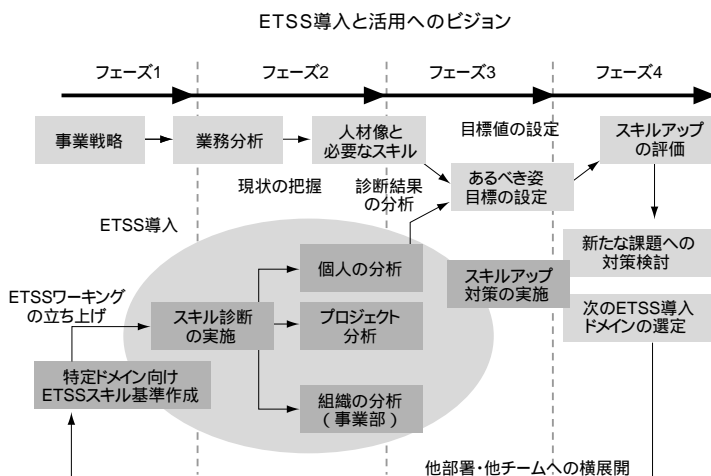


図6 コンセプトチャル・モデル

コンセプトチャル・モデルのスコープは「ETSSを段階的に導入し、エンジニア育成と組織力強化に活用すること」である。

ITA-ETSS策定は、コンセプトチャル・モデル中「フェーズ1 特定ドメイン向けETSSスキル基準作成」に位置付けられる。

3.2 オブジェクトィブ・モデル

次に、ITA-ETSSを実際に策定するに当たり、策定手法と参照標準を明確にしておく必要があった。そこで、客観的なモデルとして「オブジェクトィブ・モデル」を策定した。

(1) ITA-ETSS策定手法

ITA-ETSSを策定するに当たり、まず、策定手法を図7に示すようにモデル化した。ここでは、ETSSとITA-ETSSの関係性を直感的に理解できるように、UMLを利用してITA-ETSSの策定手法をモデル化している。

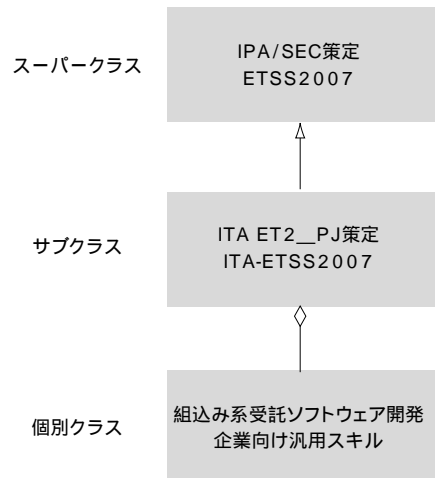


図7 ITA-ETSS策定手法モデル

ITA-ETSS策定手法のスコープは「ETSSのオリジナリティは最大限活用し、ITAとして必要なスキルは個別に追加すること」である。このため、IPA/SEC策定の「ETSS」はスーパークラスとして位置付け、ET2_PJが策定する「ITA-ETSS」はサブクラスとして「ETSS」の機能を継承するよう定義している。また、「ITA-ETSS」は、ITAで必要なスキルを「組込み系受託ソフトウェア開発 企業向け汎用スキル」として個別に追加し集約するモデルとして定義している。

なお、「組込み系受託ソフトウェア開発 企業向け汎用スキル」の意味に関しては、後述する「4 ITA-ETSSのスキル領域」を参照願いたい。

表1 ITA-ETSS参照標準

スキル基準	国際標準	SEC	備考	ITSSの領域
ITA開発技術	ISO/IEC12207(JIS X 0160) Software Life Cycle Process ISO/IEC15288(JIS X 0170) System Life Cycle Process	組込みソフトウェア向け 開発プロセスガイド(ESPR)	—	メソロジー
ITA管理技術	ISO 10006(JIS Q 10006) PMBOKガイド1996年版準拠 ISO・PC236策定中(2010年リリース) PMBOKガイド第3版準拠	組込みソフトウェア向け プロジェクトマネジメントガイド [計画書編] (ESMR)	「プロセスマネジメント」は SEPG等スタッフ系スキルのため 2007年は対象外とする	プロジェクト マネジメント
ITA技術要素	—	組込みスキル標準 (ETSS2007)	—	テクノロジー
ITAパーソナル	—	—	SEC監修の「ETSS標準ガイド ブック(赤本)」を参照ドキュ メントとする	パーソナル
ITAビジネス	—	—	厚生労働省の「ビジネスキャ リア制度」を参照先とする	ビジネス/ インダストリ

(2) ITA-ETSS参照標準

ET2_PJのミッションに「日本の国際競争力向上」のキーワードが設定されている。

このため、ITA-ETSSを策定するに当たっては参照標準となる国際標準を明確にした。次に、IPA/SECの資産を有効活用するため「SECドキュメント」を定義し、SECドキュメントが無い場合には別途参照する標準等を設定している。

ITA-ETSS参照標準を表1に記す。

また、ITA-ETSSはITSSとの整合性を確保しておきたいという観点から、相互のスキル領域の関連性を意識するように努めている。

3.3 ファンクション・モデル

これまでトップダウン・アプローチで「コンセプトチャル・モデル」から「オブジェクティブ・モデル」へと段階的詳細化を行ってきた。ここでは、実際にETSSを活用するため、機能するモデルとして「ファンクション・モ

デル (= ITA-ETSS)」を策定している。

ファンクション・モデルであるITA-ETSS策定に当たっては、ET2_PJメンバの人的リソースを考慮した「ロードマップの確定」から開始した。これはメンバの主業務に影響を与えない範囲でET2_PJを効率的に推進する上で、具体的な成果物とリリース時期をゴールとして設定しておく必要があったためである。

ET2_PJのロードマップを図8に記す。

なお、ロードマップで定義しているPhaseとコンセプトチャル・モデルで定義しているフェーズは、同一でないことに注意願いたい。ロードマップのPhaseは「年度」を意味し、コンセプトチャル・モデルのフェーズは「活動段階」を意味している。

2007年度に策定した「ITA-ETSS」は、前述のロードマップに従い、スキル標準「ITA-ETSS2007」、スキル基準「ITA-ETSS2007 スキル基準 Ver.1.0」の2ドキュメントをITAホームページにリリースしているのぜひアクセスしていただきたい(<http://www.ita.gr.jp/>)

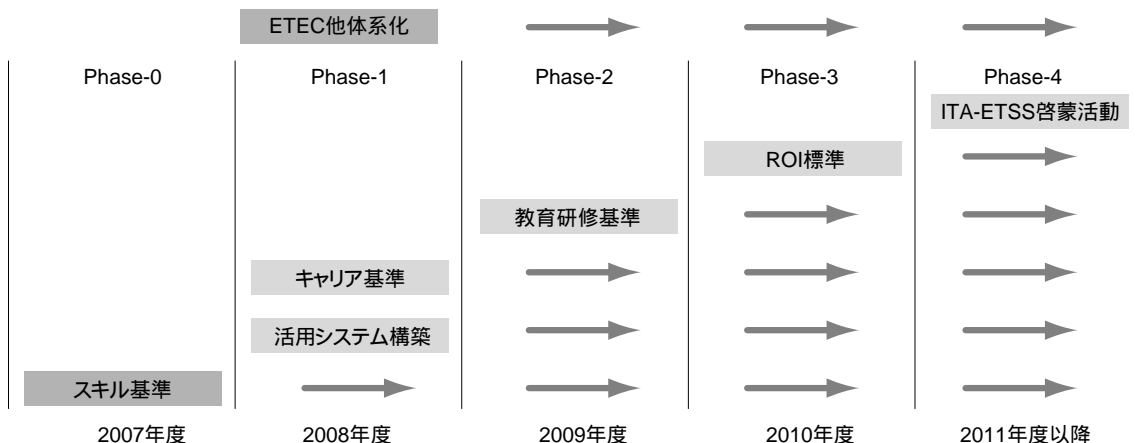


図8 ET2_PJのロードマップ

4 ITA-ETSSのスキル領域

次にITA-ETSSのスキル領域について触れておきたい。
ETSSを活用している団体としては、「JMAAB³」や「JasPar⁴」が代表的である。

ET2_PJではJMAABやJasParの資産を有効に活用しなかったが、前述の2団体はビジネスドメインが「自動車産業」であり「スキル領域」が特定されるため、ET2_PJでは一部しか参考にできなかった。

ET2_PJに参画している各社のビジネスドメインを重ね合わせると(図9) ある特定のドメインに偏った開発を行っていないことが分かる。このようにET2_PJは、「幅広いスキル領域の定義」を必要としていたのである。

ITA-ETSSが「組込み系受託ソフトウェア開発企業向け汎用スキル標準」と命名された由縁は「幅広いスキル領域を定義」しているからである。

5 スキルマネジメントWG (MOS_WG) 発足の背景

2007年度にはITA-ETSSとして策定した「スキル基準」をET2_PJ各社でパイロット入力し、「ET系エンジニアのスキル強化」に関する有効性は検証した。一方、この時点で「企業の組織力強化」に関する有効性は検証できて

いなかった。

ロードマップに従い、2008年度は「キャリア基準」策定と「ETSS活用システム構築」を目標に活動を開始したが、活動当初に根本的な課題に直面した。

スキル基準は策定したものの、「ビジネス戦略に有効なスキルとして定義されているのか?」、ここにキャリア基準を策定し追加すれば「企業の組織力強化が実現できるのか?」という難題である。

また、ET2_PJ各社では既にITSSを導入している企業もあり、今後ETSSを活用するに当たり同じスキル標準として整合性を確保しておきたいとの意向もあった。

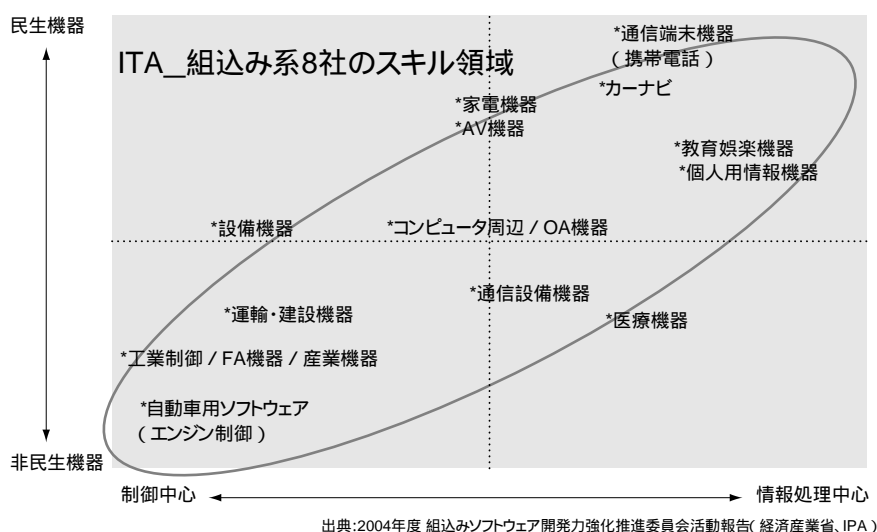
前述の観点を元に「ETSS活用システム構築」をスタートするに当たって、次のスコープを設定した。

企業の組織力強化のために「ビジネス戦略に有効なスキルをマネジメントする」仕組みを作る。

複数のスキル標準を有効利用するために「ETSSとITSSの整合性を確保する」仕組みを作る。

このような背景から、2008年度はET2_PJを発展的に解消し、MOS_WGを新たに発足したのである。本テーマとなっている「ITベンダによるETSS適用事例報告～ETSSを活用したスキルマネジメント～」は、ETSSとITSSの整合性を確保しながら、いかにスキルマネジメントを実現するかという思いが込められている。

また複数のスキル標準の整合性を確保しMOS_WGで活用するために、共通キャリア・スキルフレームワーク



出典:2004年度 組込みソフトウェア開発力強化推進委員会活動報告(経済産業省、IPA)

組込み系受託ソフトウェア開発企業向け汎用組込みスキル標準(ITA-ETSS)

図9 ITA-ETSSのスキル領域

3 JMAAB : Japan MATLAB Automotive Advisory Board

4 JasPar : Japan Automotive Software Platform Architecture

を参照している。

共通キャリア・スキルフレームワークの考え方を図10に記す。

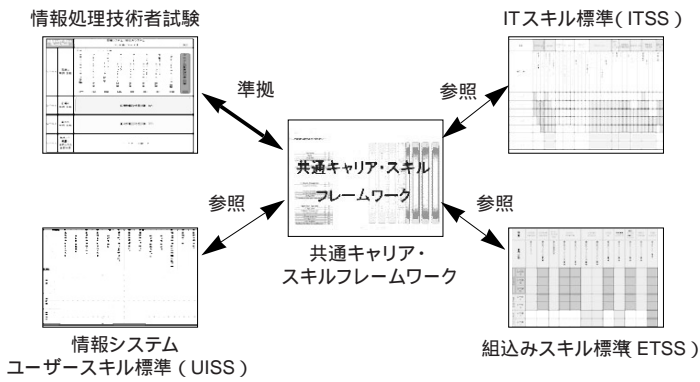


図10 共通キャリア・スキルフレームワークの考え方

6 ETSS実証実験

2008年度は、ITA-ETSSの「キャリア基準」策定と「活用システム構築」を活動の主体とし、スキルマネジメントの有効性を検証するため、IPA/SEC及びSSUGと共同で「ETSS実証実験」を開始した。

ETSS実証実験の体制を図11に記す。

また、ETSS実証実験スケジュールを図12に記す。

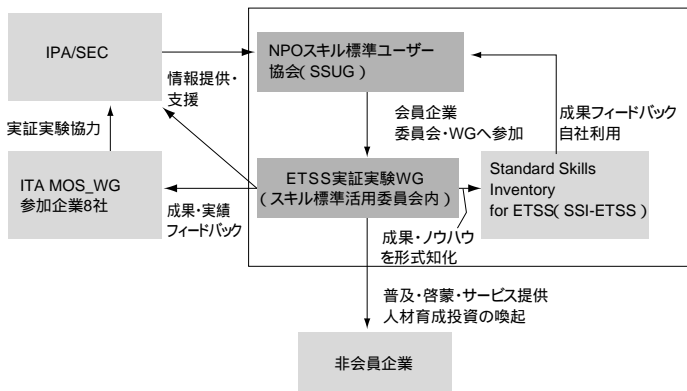


図11 ETSS実証実験体制図

6.1 フレームワーク

スキルマネジメントが「企業の組織力強化」に有効であることを検証するためには、「ビジネス戦略とスキルの関係」及び「組織とスキルの関係」を明らかにする必要がある。これは「ビジネス戦略上何のスキルが重要なのか?」、「戦術を実行する組織にはどのようなスキルが必要なのか?」という観点から、スキルの依存関係を整理し「強化すべきスキル」を明確にするためである。

最初に、ビジネス戦略とスキルの関係を可視化するために「バランスト・スコアカード (BSC⁵)」に準拠したスキルマネジメントのフレームワークを策定した。

策定したスキルマネジメント・フレームワークは、「ビジネス戦略駆動型スキルマネジメント・フレームワーク」と命名し、スキルはビジネス戦略を支える基盤として定義している。

MOS_WGで策定したビジネス戦略駆動型スキルマネジメント・フレームワークを図13に記す。

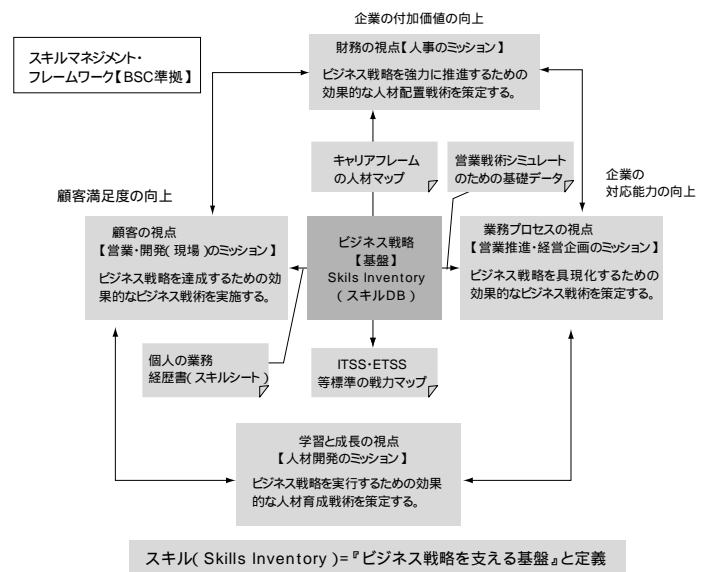


図13 ビジネス戦略駆動型スキルマネジメント・フレームワーク

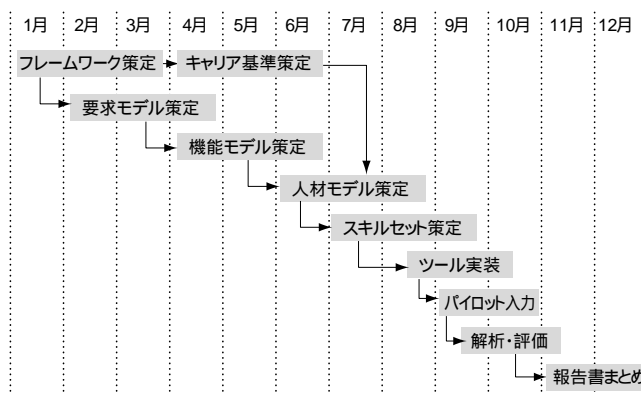


図12 ETSS実証実験スケジュール

次に、組織とスキルの関係を可視化するため、戦略マップを策定した。

戦略マップではビジネス戦略をトリガーとし、組織とスキル (Skills Inventory) の間で必要となる「スキル・フロー」を定義した上で、企業におけるスキルマネジメントを仮説化した。

策定した戦略マップ (スキルマネジメントのイメージ) を図14に記す。

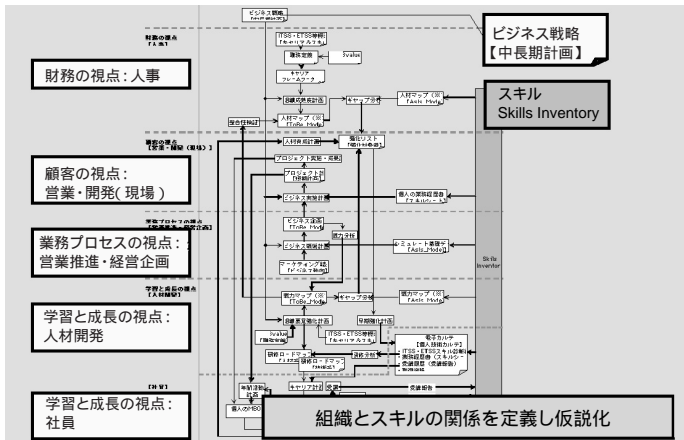


図14 戦略マップ (スキルマネジメントのイメージ)

6.2 要求モデル (オブジェクトブ・モデル)

仮説として設定したスキルマネジメントの有効性を検証するために、モデルとなる企業が必要となる。MOS_WG参加8社では仮定の合同企業である「ITA 仮想企業」を組織し、トリガーとなる「ビジネス戦略」を設定した。

設定したビジネス戦略とスキルの関係を明らかにするため、「ITA 仮想企業における要求モデル」を策定した。策定に当たっては次の手順を実施した。

ビジネス戦略を実現するためのビジネス戦術を設定する。

ビジネス戦術を実現するためのスキルを設定する。

スキルマネジメント・フレームワークの視点でビジネス戦術を整理する。

今回はビジネス戦略として「世界に信頼される企業をめざす」と設定した。これを実現する一例として、ビジネス戦術「顧客の期待する技術力」を設定、「システム開発力の強化 - 新しい技術への対応力強化」をスキルとして設定、最後に、「顧客の視点」にビジネス戦術を整理する手法である。

要求モデルは、このように「ビジネス戦略 - スキルマネジメント領域 - ビジネス戦術 - スキル」の構造化を図っている。

ITA 仮想企業として策定した要求モデルを図15に記す。

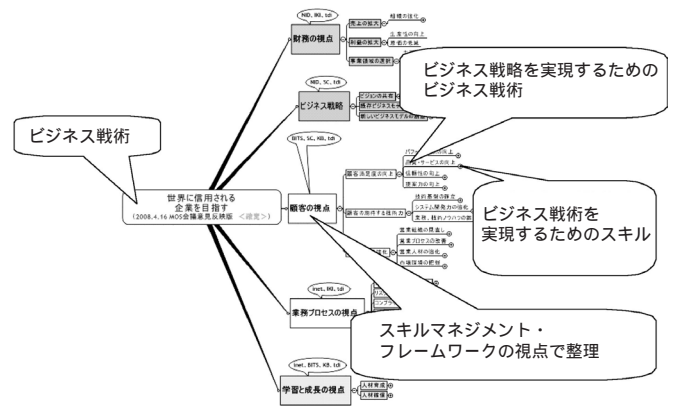


図15 ITA 仮想企業の要求モデル

6.3 機能モデル (ファンクション・モデル)

次に、組織とスキルの関係を明らかにするため、「ITA 仮想企業における機能モデル」を策定した。策定に当たっては次の手順を実施した。

ビジネス戦術を遂行する組織を設定する。

ビジネス戦術を遂行するために必要な組織の機能を設定する。

機能を実現するためのスキルを設定する。

今回はITA 仮想企業の一例として、ビジネス戦術を遂行する組織「システム開発・ソフトウェア製品開発」を設定、ビジネス戦術を遂行する組織に必要な機能「システム分析」を設定、最後に、「要求分析 - 要求モデル策定」をスキルとして設定している。

機能モデルは、このように「(ビジネス戦略を遂行する) 企業 - ビジネス戦術を遂行する組織 - ビジネス戦術を遂行する組織に必要な機能 - 機能を実現するためのスキル」の構造化を図っている。

ITA 仮想企業として策定した機能モデルを図16に記す。

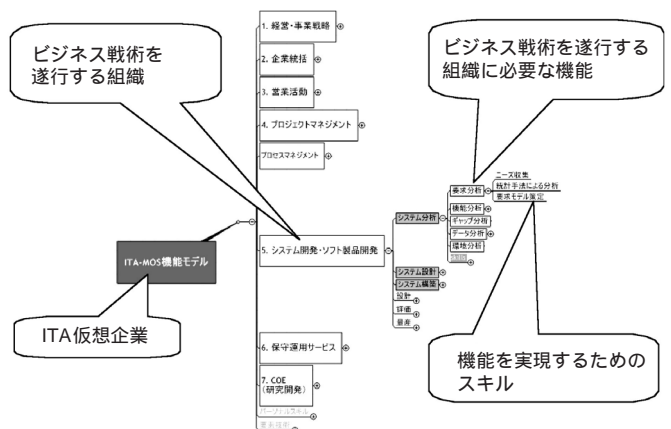


図16 ITA 仮想企業の機能モデル

6.4 要求モデルと機能モデルの検証

ここでは、要求モデルで策定した「ビジネス戦略とスキルの関係」と、機能モデルで策定した「組織とスキルの関係」に対する整合性を検証した。

要求モデルで策定した「ビジネス戦略を実現するためのスキル」を横軸にプロットし、機能モデルで策定した「機能を実現するためのスキル」を縦軸にプロットする。縦軸と横軸のスキルをクロスリファレンスすることにより、クロスした部分で「ビジネス戦略・組織・スキルの依存関係」に関する整合性を検証した(図17)。

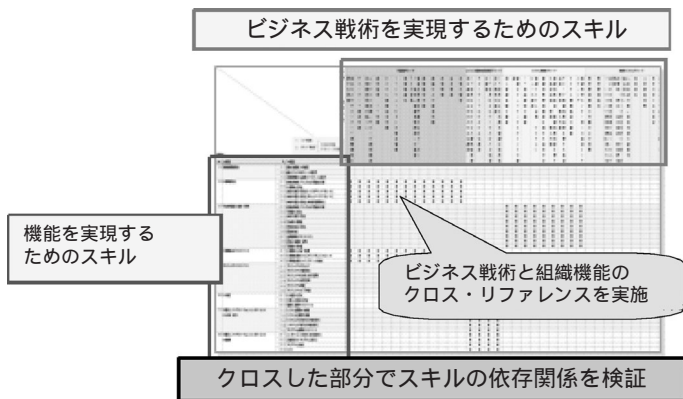


図17 要求モデルと機能モデルの検証結果

6.5 キャリア基準

次に、ビジネス戦略を遂行する組織に必要な「職種と活動領域」を設定する。今回は、ET系エンジニアの効率的なスキル強化の視点から「職種と活動領域」を策定した。このため、ET系エンジニアのスキル強化対象外職種

については、MOS_WGの検討範囲外とた。

MOS_WGで策定した職種と活動領域を表2に記す。

活動領域はETSSで記述されている「技術要素・開発技術・管理技術・パーソナルスキル・ビジネススキル」を活用した。開発技術にはIPA/SECドキュメントである「組み込みソフトウェア向け開発プロセスガイド(ESPRver.2.0)」、管理技術のプロジェクトマネジメントにはプロジェクトマネジメント協会(PMI)標準の「PMBOKガイド第3版」、プロセスマネジメントには開発技術同様「ESPR ver.2.0」を標準として参照している。

職種としては次の12職種を設定した。

- ・営業
- ・プロダクトマネージャ
- ・プロジェクトマネージャ
- ・ソフトウェアエンジニア
- ・ハードウェアエンジニア
- ・QAスペシャリスト
- ・開発環境エンジニア
- ・システムアーキテクト
- ・プロセス改善スペシャリスト
- ・ドメインスペシャリスト
- ・ブリッジSE
- ・テストエンジニア

ETSSで記述されていない「営業/ハードウェアエンジニア」の2職種は、MOS_WGとして必要なエンジニアとして新たに設定している。ただし、ハードウェアエンジニアについては国際的に見てもハードウェア開発に関する標準が無く、カバーするテクノロジーの範囲が広いため、現在「職種(定義)と活動領域」を再検討中である。

表2 職種と活動領域

スキル領域 職種	技術要素	パーソナルスキル	ビジネススキル	開発技術												ESPR Ver.2.0参照		
				システムエンジニアリング			ソフトウェアエンジニアリング/プロセス						システムエンジニアリング			製造検査		
				製品企画	システム要求定義	アーキテクチャ設計	ソフトウェア要求定義	ソフトウェアアーキテクチャ設計	ソフトウェア詳細設計	実装および単体テスト	ソフトウェア統合テスト	ソフトウェア総合テスト	システム統合テスト	システムテスト				
営業																		
プロダクトマネージャ																		
プロジェクトマネージャ																		
ドメインスペシャリスト																		
システムアーキテクト																		
ハードウェアエンジニア																		
ソフトウェアエンジニア																		

職種	PMBOK®ガイド第3版参照												ESPR Ver.2.0参照					
	プロジェクトマネジメント						ソフトウェアエンジニア						プロセスマネジメント					
	統合マネジメント	スコープマネジメント	タイムマネジメント	コストマネジメント	品質マネジメント	人的資源マネジメント	コミュニケーションマネジメント	リスクマネジメント	調達マネジメント	安全性要求定義	安全性テスト	文書化と文書管理	構成管理	問題解決管理	変更管理	共同レビュー	開発委託管理	開発環境整備

MOS_WGで策定した職種とキャリアレベルを図18に記す。

職種	1 新卒	2 システムマネージャ	3 プロジェクトマネージャ	4 ソフトウェアエンジニア	5 ハードウェアエンジニア	6 O/Aシステムリスト	7 情報技術推進エンジニア	8 システムアーキテクト	9 プロセス改善スペシャリスト	10 ドメインスペシャリスト	11 アプリシスマ	12 テクニシアン
5 High	3年											
5 Entory	3年											
4 High	3年											
4 Entory	3年											
3	3年											
2	3年											

ハードウェアエンジニアは参照標準がなく2009年にスライド

図18 職種とキャリアレベル

また、MOS_WGでは企業内でET系エンジニアが育成できる範囲を明確にするために、ETSSのキャリアレベルに補正を加えている。

ETSSのキャリアレベルは共通キャリア・スキルフレームワークのキャリアレベルを参照しているため、キャリアレベル6は「国内を代表するハイエンドプレーヤ」、キャリアレベル7は「国際的なハイエンドプレーヤ」となる。MOS_WGでは企業内でキャリアレベル6～7は育成不可能と判断し、キャリアレベル1～5を代表的な組織構造にマッピングしている。

MOS_WGが策定したキャリアレベルと共通キャリア・スキルフレームワークのキャリアレベルの差異を図19に示す。

共通キャリア・スキルフレームワークのキャリアレベル	
7	業界全体から見ても先進的なサービスの開拓や事業改革、市場化等をリードした経験と実績を有し、世界レベルでも広く認知される。
6	社内だけでなく業界においても、プロフェッショナルとしての経験と実績を有し、社内外で広く認知される。
5	プロフェッショナルとして豊富な経験と実績を有し、社内をリードできる。
4	高度な知識・スキルを有し、プロフェッショナルとして業務を遂行でき、経験や実績に基づいて作業指示ができる。またプロフェッショナルとして求められる経験を形式化し、後進育成に活用できる。
3	応用的知識・スキルを有し、要求された作業についてすべて独力で遂行できる。
2	基本的知識・スキルを有し、一定程度の難易度または要求された作業について、その一部を独力で遂行できる。
1	情報技術に携わる者に必要な最低限の基礎的知識を有し、要求された作業について、指導を受けて遂行できる。

図19 キャリアレベルの差異

ただし、ITA-ETSSのキャリア基準でもキャリアレベル5Highの上位レベルとしてキャリアレベル6と7をそのまま設定している。

詳細については2009年2月にリリースした「ETSS2008キャリア基準ver.1.2」を参照願いたい。

6.6 人材モデル

キャリア基準をより具体的に定義するため、職種共通に必要なキャリア要件・職種ごとに必要なキャリア要件という形で「人材モデル」を策定している。

人材モデルは、次の手順で策定している。

職種を定義する。

職種共通で期待するキャリア要件を定義する。

職種ごとに期待するキャリア要件を定義する。

今回の場合には「期待するキャリア」を定義する上で「期待人物像・対仕事面のスキル・対人面のスキル」の項目を設定した。

策定したソフトウェアエンジニアの人材モデルを図20に記す。

職種	期待人物像	対仕事面のスキル	対人面のスキル	育成上の課題
ソフトウェアエンジニア	ソフトウェア開発に関する技術を持ち、システム要求分析から基本設計、詳細設計、製造、検証試験を実施できる技術者。	ソフトウェア開発に関する技術を持ち、システム要求分析から基本設計、詳細設計、製造、検証試験を実施できる技術者。	ソフトウェア開発に関する技術を持ち、システム要求分析から基本設計、詳細設計、製造、検証試験を実施できる技術者。	ソフトウェア開発に関する技術を持ち、システム要求分析から基本設計、詳細設計、製造、検証試験を実施できる技術者。
キャリアレベル定義	期待人物像	対仕事面のスキル	対人面のスキル	育成上の課題

図20 ソフトウェアエンジニアの人材モデル

MOS_WGでは「実在人物イメージ」と「育成上の課題」に関してはとくに規定していないが、各企業で活用する場合には具体的に設定して欲しい。MOS_WGは8社の集合体であり、個々の企業が有する特徴を表現できないため割愛していることに注意願いたい。

6.7 スキルセット

これまでのプロセスでアウトプットしてきた成果物を元に、職種ごとに必要とするスキルセットを策定した。

スキルセットは、次の手順で策定している。
 機能モデルで策定した「ビジネス戦略を遂行する組織に必要な機能」と「スキル」を縦軸にプロットする。
 キャリア基準策定した「職種」と「キャリアレベル」を横軸にプロットする。
 人材モデルで定義したキャリア要件を元に、キャリアレベルごとに必要とするスキルを特定する。
 このようにして「職種ごとのキャリアレベルに必要なスキルセット」を設定した。

策定した職種ごとのスキルセットを表3に記す。

表3 職種ごとのスキルセット

職種ごとのキャリアレベルを判定するため、次に、職種のキャリアレベルごとに保有していなければいけない「スキルレベル」の割合を定義した。スキルレベルの詳細については「ETSS2008 スキル基準ver.1.2」を参照して欲しい。

今回の場合、ある職種の「キャリアレベル3」と判断するためには「Aスキルがスキルレベル2以上であり、かつBスキルがスキルレベル3以上であること」のように条件を設定している。

キャリアレベル判定条件のイメージを表4に記す。

ここで、MOS_WGが工夫しているところは、ETSSとITSSのスキルレベルの整合性を確保するために、ETSSとITSSのスキルレベル間補正を行っている点である。

スキルレベル間補正のイメージを表5に記す。

表5では、左にETSSのスキルレベル、右にITSSのスキルレベル、中央にITA-ETSSのスキルレベルを配置している。

例えば、ETSSのスキルレベル1「支援の下に作業を遂行できる」とITSSのスキルレベル2「サポートがあれば実施可能」は同等のため、ITA-ETSSではスキルレベル2「他者からの助言やサンプル提示を受け、担当業務を遂行した経験がある。」と定義している。

6.8 ツール実装

SSUGが保有しているツール「SSI-ITSS」に、これまでの成果を実装し、「SSI-ETSS」と称して実験フィールドを提供していただいた。

表4 キャリアレベル判定条件のイメージ

表5 スキルレベル間補正のイメージ

ETSS2008 (スキルレベル)	ITA-ETSS2008 (スキルレベル: SL)	ITSS (スキルランク)
4 新たな技術を開発できる(後進の育成・指導が可能)	5 新たな技術を開発できる(後進の育成・指導が可能)	4 後進の育成・指導が可能
3 作業を分析し改善・改良できる(後進の育成・指導が可能)	4 作業を分析し改善・改良できる(後進の育成・指導が可能)	3 単独で実施可能
2 自律的に作業を遂行できる	3 自律的に作業を遂行できる	2 サポートがあれば実施可能
1 支援のもとに作業を遂行できる	2 支援のもとに作業を遂行できる	1 (ベースとなる)知識あり
0 やったことがない(支援のもとにも作業ができない)	1 やったことがない(知識あり)	0 知識なし
	0 やったことがない(知識なし)	

SSI-ETSSのイメージを図21に記す。



図21 SSI-ETSSのイメージ

6.9 パイロット入力

SSUGから提供いただいた「SSI-ETSS」に、MOS_WG参加8社から代表者を選抜しパイロット入力を実施した(図22)。

パイロット入力の目的は、策定したスキルセットとキャリアレベル判定条件の妥当性評価である。このため、パイロット入力代表者の人選は「前述した妥当性を確認できるモデル人材」という基準で選出した。

NO	ID	キャリアレベル	職種	該当者なし
1	mes0112	ハイ	ソフトウェアエンジニア	
2	mes0113	ミドル	ソフトウェアエンジニア	
3	mes0114	エントリ	ソフトウェアエンジニア	
4	mes0115	ハイ	システムアーキテクト	
5	mes0116	ミドル	システムアーキテクト	
6	mes0117	ハイ	プロジェクトSE	
7	mes0118	ミドル	プロジェクトSE	
8	mes0201	ハイ	営業	
9	mes0211	ミドル	営業	
10	mes0221	エントリ	営業	
11	mes0222	ミドル	営業	
12	mes0241	エントリ	営業	
13	mes0261	ミドル	営業	
14	mes0271	エントリ	営業	
15	mes0281	ミドル	営業	
16	mes0301	ハイ	プロジェクト・マネージャ	
17	mes0311	ミドル	プロジェクト・マネージャ	
18	mes0321	ハイ	ソフトウェアエンジニア	
19	mes0331	ミドル	ソフトウェアエンジニア	
20	mes0351	ミドル	QAスペシャリスト	
21	mes0371	ミドル	開発環境エンジニア	
22	mes0381	ハイ	開発環境エンジニア	
23	mes0401	ミドル	システム・アーキテクト	
24	mes0411	ハイ	プロセス改善スペシャリスト	
25	mes0421	ミドル	プロセス改善スペシャリスト	
26	mes0431	エントリ	ドメイン・スペシャリスト	
27	mes0451	エントリ	プロジェクトSE	
28	mes0461	ミドル	テストエンジニア	
29	mes0471	エントリ	テストエンジニア	
30	mes0481	エントリ	テストエンジニア	
31	mes0491	ミドル	テストエンジニア	
32	mes0501	ハイ	テストエンジニア	

職種	専門分野	キャリアレベル							合計
		エントリ	ミドル	ハイ	合計	エントリ	ミドル	ハイ	
営業		×	×	×	×	×	×	×	9
プロジェクト・マネージャ									0
ソフトウェアエンジニア									23
システムアーキテクト									0
プロジェクトSE									3
開発環境エンジニア									15
システム・アーキテクト									1
プロセス改善スペシャリスト									2
ドメイン・スペシャリスト									1
プロジェクトSE									3
テストエンジニア									11
合計		14	28	6	12	6	0	0	61

図22 パイロット入力結果

6.10 解析・評価

パイロット入力の結果から、予想通りの結果が得られた職種とそうでない職種が明らかになった。

予想通りの結果が得られた職種である「プロセス改善スペシャリスト」を図23に、そうでなかった職種である「ソフトウェア・エンジニア」を図24に記す。いずれもキャリアレベル4の場合を表している。

パイロット入力結果

The table shows skill levels (Level 1 to Level 7) for various job profiles. For 'プロセス改善スペシャリスト', Level 4 is the highest achieved level across all skills.

図23 プロセス改善スペシャリストの例

上段はパイロット入力結果のスキルレベルであり、下段がキャリアレベル判定条件で設定したスキルレベルである。

一見して分かるように上段のスキルレベルは、下段のスキルレベルをいずれも上回っており、想定したキャリアレベルとして判定された。

パイロット入力結果から解析すると、「専用技術者としてスキルを設定したため、ビジネスドメインに左右されない結果が得られた」と評価できる。

次に、ソフトウェアエンジニアを記す。

パイロット入力結果

The table shows skill levels (Level 1 to Level 7) for various job profiles. For 'ソフトウェアエンジニア', Level 4 is the highest achieved level across all skills.

図24 ソフトウェアエンジニアの例

プロセス改善スペシャリストの例に対し、パイロット入力結果のスキルレベルとキャリアレベル判定条件のスキルレベルが逆転していることが分かる。このため、想定したキャリアレベルとして判定されない結果となった。

パイロット入力結果から解析すると、「汎用技術者としてスキルを設定したため、ビジネスドメインに左右され未経験領域が多い結果となった」と評価できる。

以上の観点から、「専門技術者系職種」の妥当性は確認できたが、「汎用技術者系職種」に関しては各企業のドメインを考慮する必要がある課題が抽出された。

課題は残しつつも、今回の目的とした「ET系エンジニアのスキル強化」と「企業の組織力強化」を両立するための仕組みである「ビジネス戦略駆動型スキルマネジメント・フレームワーク」の有効性は担保されたと評価している。

7 今後の展開

2008年度のETSS実証実験成果を基に、2009年度もETSS実証実験を継続する。課題となっている「汎用技術者系職種」のスキルセットとキャリアレベル判定条件の見直しを含め、更にITA各社で導入しやすい仕組み作りを推進していく予定である。また現時点ではITA数社で実際にITA-ETSSを導入し、スキルの棚卸を実施している状況である。

これと歩調を合わせ、「教育研修基準」策定と「ET系資格体系化」を順次開始する予定としている。

この機会を境に多くの方々に「ITA-ETSS」とMOS_WGの考える「ビジネス戦略駆動型スキルマネジメント・フレームワーク」とを利活用していただければ幸いである。

謝辞

ITA-ETSS策定ならびスキルマネジメント・システム開発でMOS_WGをサポートして頂いた関係各位に対し、感謝の意を込め以下に紹介させて頂く。

ETSS推進テクニカルアドバイザー

IPA/SEC 渡辺 登氏

IPA/SEC 石川 秀一氏（～2008年/09、現、東芝情報システム株式会社）

有識者

SSUG 植松 英次氏

SSUG 樽谷 謙二氏

SSUG 井本 貴志氏

SSUG 清水 千博氏

iSRF 堤 裕次郎氏

JASA 近森 満氏

発起人

旧ITA事務局 遠藤和哉氏（現、IPA/SEC）

参考文献

[ETSS2007/ETSS2008] IPA/SEC：組込みスキル標準 ETSS2007/ETSS2008

[ESPR ver.1.0/ESPR ver.2.0] IPA/SEC：組込みソフトウェア向け開発プロセスガイド ESPR ver.1.0/ESPR ver.2.0

[IPA] IPA：共通キャリア・スキルフレームワーク 第一版

[IPA2006] IPA：組込みソフトウェア開発のためのETSS標準ガイドブック,日経BP社,2006

[PMI2004] PMI：プロジェクトマネジメント知識体系ガイド第3版（PMBOK ガイド第3版）,PMI,2004

3 富士ゼロックスにおけるETSSの実践的取り組み

富士ゼロックス株式会社
技術人材教育センター グループ長
遠藤 裕隆

富士ゼロックス株式会社では、ソフトウェア技術者の技術力を測るため、ETSSに準拠するFX-STSS¹を開発し、技術者教育に活かしてきた。ここでは、ETSSの活用事例を紹介する。

1 はじめに

富士ゼロックス株式会社（以下、富士ゼロックス）は、以前からコンピテンシーアセスメントを実施しているが、技術力を定量的に測定し、強みと弱みを把握する活動が不十分であった。2006年度にETSSに準拠したFX-STSSを開発することにより、ソフトウェア技術者の技術力を定量的に測定し、「事業貢献」を目指した技術者教育に取り組んできた。

ETSSベースのコンピテンシーアセスメントは、2007年度と2008年度に実施し、活動の成果も見えてきたところであり、ここではETSSの活用事例として富士ゼロックスの活動状況を述べる。

2 技術人材育成ビジョン

技術人材育成ビジョンの狙いは、「事業戦略の実現」と「自己実現」を具現化するために、キャリアパスをベースに専門能力と人間力をバランスよく強化し、「技術系プロフェッショナル人材」を育成していくことである。同時に「車の両輪」となる、技術力をリードする「プロフェッショナル人材」の育成と共に、技術力の平均値を高めるための「自りつ²人材」育成にも取り組んでいる（図1）

3 人材育成のグランドデザイン

技術人材育成ビジョンを実現するために、人材育成活動全体を2次元の「垂直統合」と「水平統合」で捉え、次に述べる3軸で構成されるグランドデザインを構想した（図2）

会社軸：事業貢献を目指した人材育成プロセスを提供し、技術力強化を図ることにより、事業戦略を実現

個人軸：技術者の「モチベーションを向上」し、技術力強化を図ることにより、「自己実現」を達成

グローバル軸：国内/海外の技術者に、「人材育成を考える場」を提供することにより、とを実現

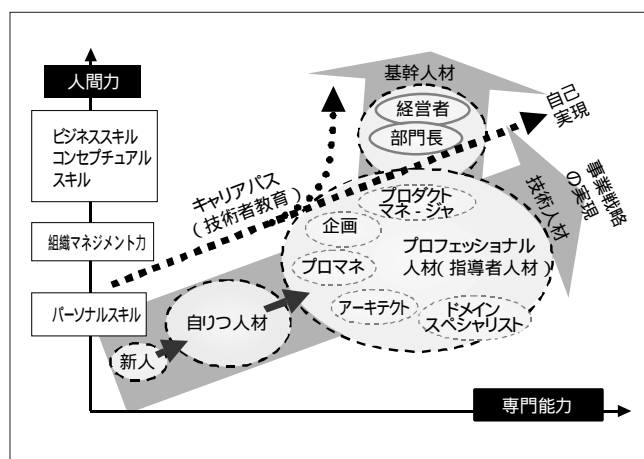


図1 技術人材育成ビジョン

1 FX-STSS : Fuji Xerox Software Technology Skill Standard

2 自りつ: 「自立」と「自律」との両方の意味をかけた造語。自分で立ち立をして(自立) 自らを律し自分で考えて行動する(自律)ことを表わす

(1) 課題解決の「場」

事業貢献に直結した「経営課題」や「部門課題」の解決を図るためには、トップマネジメントや現場のマネージャに働きかけて、現場の活動とバランスよく連携をとっていきながら、「分かる 出来る 使いこなす (指導出来る)」をカバーする技術者教育等の要素で構成される、課題解決の「場」を構築することが重要である(図3)。後述する技術教育推進委員会は、技術者教育のみに焦点を当てるのではなく、これらの3つの要素を融合しながら、課題解決の「場」を構築して提供する責務を担っている。

(2) 技術者教育の「場」

富士ゼロックスでは、スキル開発の中心となる技術者教育の「場」として、技術伝承、触発、変革、の3つの観点で取り組んでいる(図4)。

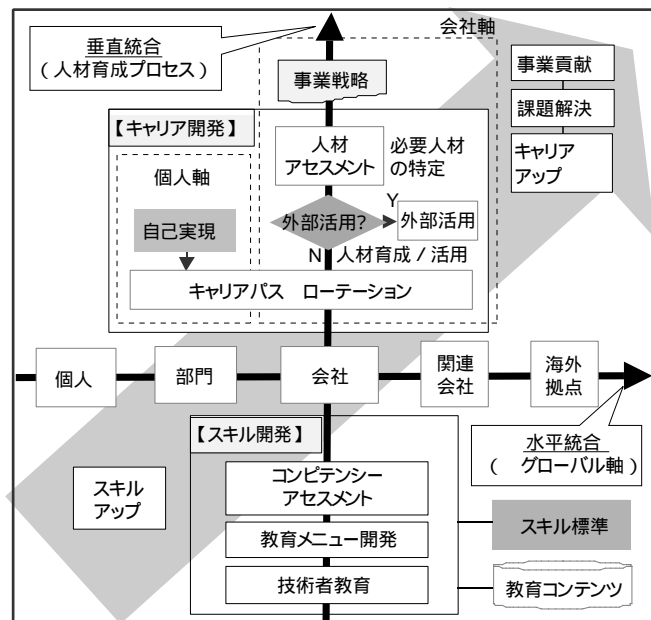


図2 人材育成のグランドデザイン

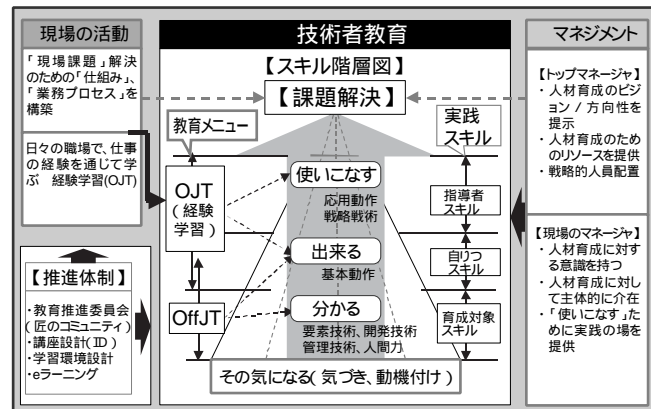


図3 課題解決の「場」

「技術伝承」は、開発・生産現場の「匠」である先輩技術者が後輩に技術を伝承する「場」であり、「触発」は、社内外の「その道の先駆者」から刺激を得て、次の行動のきっかけをつかむ「場」、「変革」は、市場革新をもたらす新たなビジネスモデル、ビジネスプロセスや技術革新を構想し、実現のために行動を起こす「場」である。さらに、これらの「場」を円滑に運用するために、教育推進委員会、人材育成プロセス、教育プロセス、講座設計手法、eラーニング等から構成される推進体制を整備してきた。

(3) 人材育成プロセスの鳥瞰図(図5)

図2でも示した人材育成プロセスを実現する2要素として、各種の人材育成ガイドと、育成手段となる基幹ツールを構築中である。人材育成ガイドは、主にキャリアパスを実現するキャリア標準や、スキル標準等で構成され、育成手段は、キャリア開発システムやスキル開発システム等で構成される。

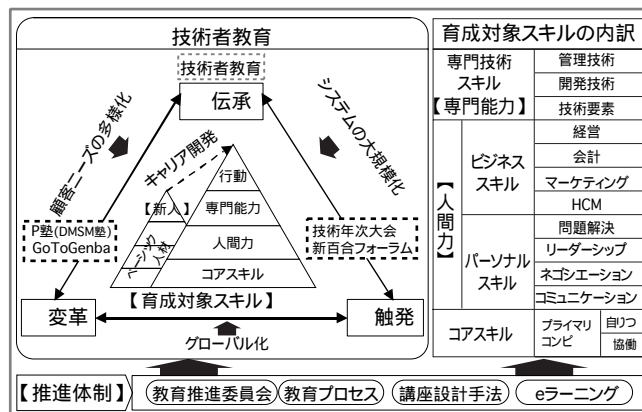


図4 技術者教育の「場」

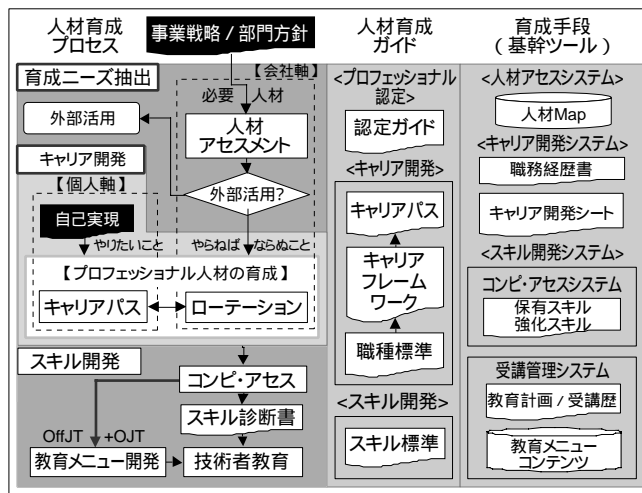


図5 人材育成プロセスの鳥瞰図

4 技術教育活動の歩み

富士ゼロックスの技術者教育は、2000年度から本格的にスタートし、コンピテンシーアセスメントの結果をベースに、技術講座を開発してきた。過去の経緯を表1にまとめる。

講座の提供状況は、2008年度時点で、合計240講座（140集合講座と100WBT³講座）を提供中であり、年10%増のペースでWBT講座を開発している（図6）。

5 匠のコミュニティ

富士ゼロックスの技術教育推進体制は、図7のようにソフトウェア教育推進委員会とハードウェア教育推進委員会で構成される。教育推進体制のエンジンとなる部分は、現場の第一線の技術者から構成される「分野長/講座長体制」であり、「匠のコミュニティ」と呼んでいる。

表1 技術教育活動の過去の経緯

期 間	内 容
第1期 (2000～2002年) 講座数の増加	<ul style="list-style-type: none"> 現場の第一線の技術者で構成される、分野長/講座長体制を構築し、社内の「匠」が講師となり140集合講座を開発 2001年度に、能力開発優秀企業賞特別賞を受賞
第2期 (2003～2005年) 「e化」による 受講量の増加	<ul style="list-style-type: none"> 受講管理システム(新百合カレッジ)を自前で開発 技術者の受講機会を増やすために、90WBT講座を自前で開発 2005年度にeラーニング大賞(経済産業大臣賞)を受賞
第3期 (2006年～) グローバルレベル の技術力強化	<ul style="list-style-type: none"> 技術教育推進委員会をリニューアル ETSS準拠のFX-STSSを開発し、これをベースに「スキル開発システム」を自前で開発 「グローバルレベルの技術力強化」のために、海外拠点に対して、教育コンテンツとスキル開発システムを展開

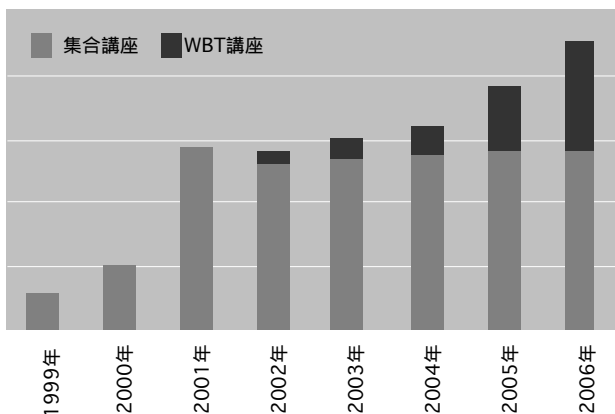


図6 提供講座の推移

(1) 教育推進委員会のリニューアル

2006年度の段階で、「技術教育は充実してきたが、現場の課題解決に役立っているかどうか分からない」という声を受け、教育推進委員会のリニューアルを実施した。

新規教育推進委員会は、グローバルレベルの人材育成ガバナンスを念頭に置き、担当役員を委員長に迎えることにより、トップ自らのリーダーシップの下に「事業貢献」を目指した活動を開始した。

(2) 活動方針

ソフトウェア教育推進委員会の活動方針としては、ソフトウェア技術者の技術力向上のために、現場のマネージャの主体的な介入を促進し、継続的に「人材育成の場」を提供すること、とした。

具体的には、次の3点を実現することである。

- ・人材育成の一部として教育推進を実施すること
- ・「現場の課題」と「経営の期待」の両面に沿うこと（事業貢献に取り組むこと）
- ・成果はマネジメントが認識できること（成果の見える化）

(3) 教育分野体系図

活動の対象となる教育分野を決めるために、既存の教育分野とETSSのスキルカテゴリとのマッピングを実施し、9分野からなる教育分野体系図を再構築した（図8）。各教育分野には、社内の「匠」で構成される正分野長/副分野長がアサインされ、教育メニューの企画/設計を担当している。

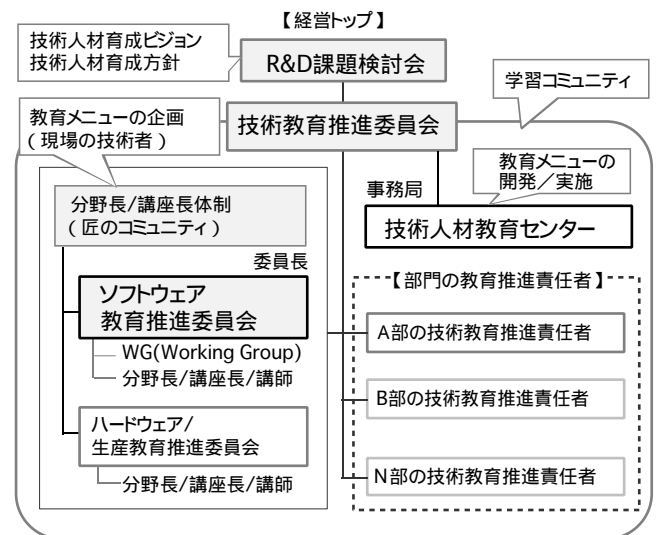


図7 技術教育推進体制

3 WBT : Web Based Training

6 「事業貢献」への取り組み

6.1 現状把握

「技術教育が、現場の課題解決に役立っているかどうか分からない」という問題提起に対して、現状分析/現状把握を実施した。

6.2 要因分析

主な要因として、次の3点を抽出した。

技術力を的確（定量的）に把握出来ていない

コンピテンシー辞書の内容をETSSにマッピングしたところ、とくに開発技術分野の抜け漏れがあり、スキルの網羅性が不十分であることを確認出来た。

現場（部門）の課題を的確に把握出来ていない

課題把握が断片的で、全部門の課題の全体像を網羅的/体系的/定量的に把握出来ていない。

現場（部門）課題を解決するための強化スキルが不明

部門の課題を解決するために必要な強化すべきスキルの全体像を把握出来ていない。

6.3 対応策

(1) 現場のマネージャ自らが人材育成をリードする体制

前述の要因の対応策として、最初に、現場のグループ長をリーダとする4つのWG⁴を発足し、グループ長自ら

スキルカテゴリ		ソフトウェア教育分野(9分野)				
技術要素 (ドメイン技術)	What系 (サービス技術)	G分野(サービス技術) サービスビジネスモデル関連技術				
		O分野(サービスソフトウェア技術) Webアプリ、セキュリティ技術				
How系		N分野 (コントローラコア)		M分野 (ネットワーク)		
		F分野 (画像処理)		L分野 (OS・ドライバ)		
	開発技術	企画	分析	設計	実装	テスト
		Y分野 (企画)		J分野(SW開発基礎) プログラミング、オブジェクト指向		
管理技術	プロジェクトマネジメント		プロセスマネジメント			
	K分野(SW開発管理) プロマネ、CMM、開発プロセスマネジメント					

図8 教育分野体系図

が、教育コンテンツ(教育メニュー)の企画/設計/開発、人材育成プロセスの構築、各種標準の開発をリードするようにした(図9)。現時点で対象となる標準は、スキル標準とキャリア標準の2種類であり、ETSSをベースとしたものを開発している。WG3/WG4は、組み系/IT系/エレキ系のすべての関連部門のマネージャから構成され、活発に活動を実施している。

(2) 現場の課題を解決するプロセスの構築

更に、抽出された3要因への対応策として、スキルのものさしの開発、部門課題調査、コンピテンシーアセスメント、スキルギャップ分析、教育メニュー開発、技術者教育、の6ステップから構成される、「スキル開発システム」を構築することにし、現場の課題解決に役立つ技術教育を提供する活動に取り組み始めた(図10)。

次に、から の順番で実施した概要を述べる。

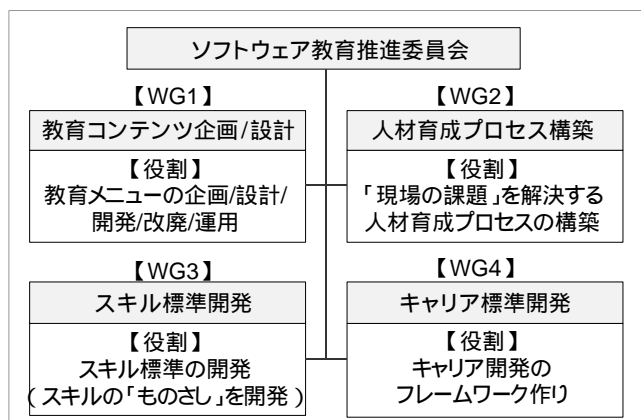


図9 WGの構成

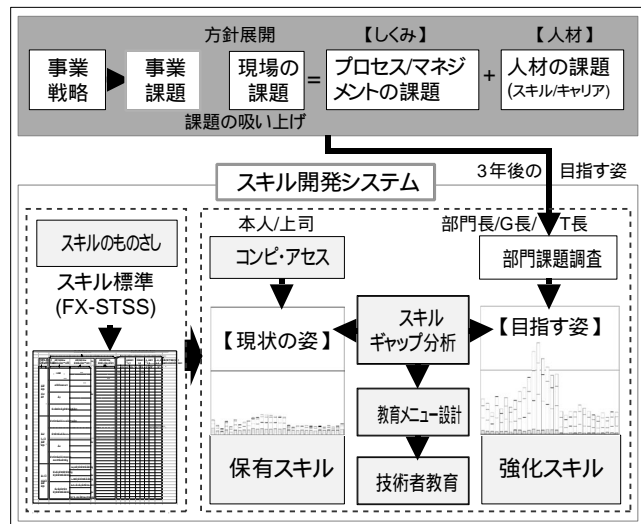


図10 現場の課題を解決するプロセス

4 WG : Working Group

「スキルのもものさし」の開発

・専門コンピテンシー辞書のグレードアップ

技術力を的確（定量的）に把握するために、2006年度に、WG3にてETSSと専門コンピテンシーを融合したFX-STSSを開発した。今回は、組込み系 / SI系技術者を対象にするために、システム運用等のIT系スキルをITSSから流用する等を行い、専門コンピテンシーを、技術要素、開発技術、管理技術の3分野のカテゴリにマッピングし、約400項目で構成されるFX-STSSを開発した。専門コンピテンシー辞書への追加は、約70項目となり、とくに開発技術への追加比率が最も高くなった（図11）。

・組織に必要な必須 / 選択スキルの絞込み

今回は、組織（部門、グループ、チーム等）ごとに重点スキルをフレキシブルに絞り込むために、50項目を目安にした必須スキルと選択スキルを設定可能にした（図12）。必須スキルとは、コンピテンシーアセスメント時に必須で入力するスキルであり、組織の過半数（60%）が保有するスキルである。一方、選択スキルとは、少数でも（10%程度）組織が必要とされるスキルで、選択入力を可能にした。2007年度は、約70組織の必須 / 選択スキルを記述した「アセスパターンファイル」を設定し、アセスメントを実施した。

・スキルの階層構造

各種スキルの関係を述べると、1,000項目で構成される

専門コンピテンシー辞書の中から、400項目のFX-STSSを選択し、更に、FX-STSSから50項目の必須 / 選択スキルを選択するようにしている（図13）。後述する強化スキルは、必須 / 選択スキルの中から指定する。

・スキルレベルのマッピング

同時に、スキルレベルの整合をとるために、ETSSと専門コンピテンシーのスキルレベルのマッピングを行った（表2）。

部門課題調査

現場から問題提起された、「部門課題に直結する教育」を実現するためには、現場（部門）の課題を的確に把握する必要がある。2007年度から、全社のアセスメント実施前に、部門のマネージャが部門課題とそれを解決するために、必要なスキルをイントラネットから入力可能にすることで、現場のマネージャに「部門課題と解決策を考える場」を提供することにした（図14）。

・部門長方針展開のステップ

- 部門長は、人材育成方針と、今後3年のスパンで部門課題とそれを解決する強化スキルを入力
- G長 / T長は、部門長の上位方針を受けて自グループ / 自チームの育成方針、課題と強化スキルを入力
- 上司（部門長 / G長 / T長）は、自分の部下一人当たり3件の強化対象スキルと、強化目標レベルを入力

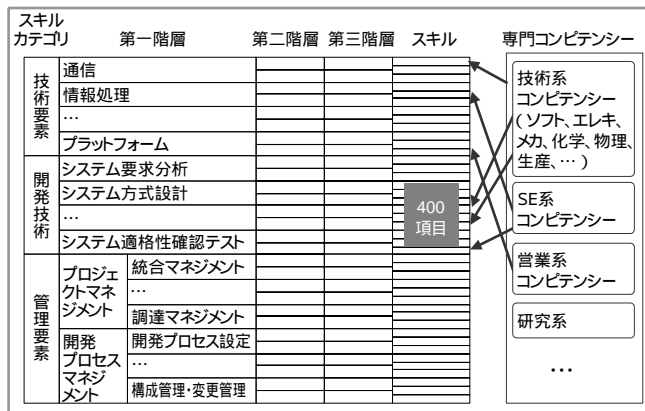


図11 FX-STSSの基本構成

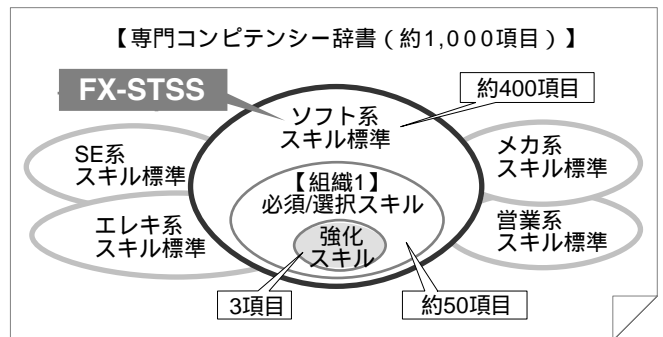


図13 スキルの階層構造

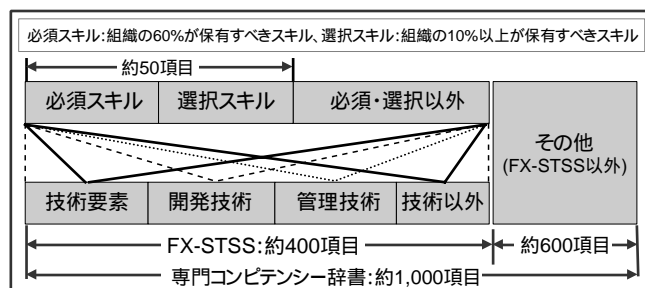


図12 必須 / 選択スキル

表2 スキルレベルのマッピング

ETSS	レベル	レベル	専門コンピテンシー	
		1	関心がある	育成対象
指導を受けてできる	初級	2~2.5	この分野での基本がある	
自分でできる	中級	3~3.5	一般的な課題を解決できる	自りつ
指導できる	上級	4~4.5	複雑な課題を解決できる	指導
新技術を開発できる	最上級	5	難解な課題を解決できる	

(d) 部下は、本人アセスメント時に上司の育成方針、課題認識、3件の強化スキル等を確認し、上司の期待としての自分の「強化ポイント」を把握

これらのステップにより、部門/個人レベルで「3年後の目指す姿」をあげり出すことが出来た。

アセスメントの実施

今回は、組込み系技術者とSI系技術者に対してアセスメントを実施し、「保有スキル」を抽出した。次に、アセスメントの概要を述べる。

・アセスメントワークフローの提供

アセスメントの実施に当たり、Web上でワークフローを提供することにより、全体の中で、どの工程を実施しているのかを明確に把握出来るようにした(図15)

・アセスメント画面の提供

図16にアセスメント画面を示すが、スキルカテゴリ順、種類順(必須スキル、選択スキルの順)第1階層順、第2階層順、スキル番号順に体系的/階層的に表示される。この画面は、強化スキル、本人スキル、上司スキル、確定スキル等の閲覧/入力/編集に活用する。従来、受診者は必要スキルを都度検索していたが、今回はアセスメントの対象となるスキルが必須/選択スキルとして画面に埋め込まれているので、検索の手間が省け、容易に入

力出来る。

また、自分のスキルレベルが部門/会社の中でどの位置にあるかを確認するために、「ポジショニング分析機能」を提供し、他者との相対的な比較を可能にした。この機能は、アセスメント受診者に好評である。

スキルギャップ分析

強化スキルを把握するために、2007年度に部門課題調査分析とスキルギャップ分析を実施し、部門課題と強化スキルの関係を定量的に把握した(図17)。結果として、強化ポイントとなる、何を(強化対象スキル)どの程度まで強化するか(強化目標レベル)を抽出することが出来た(図18)。

No	カテゴリ	種類	第1階層	第2階層	第3階層	コンピュータ系項目	コード	本人	上司	確定	強化
1	技術要素	必須	情報処理	セキュリティ	-	セキュリティ管理技術	A002&0	2	未	未	3.5
2	技術要素	必須	情報処理	データ処理	サーバ系	サーバアプリケーション系技術: GroupWare (ノーツ、Exchange等)	B0020	3	未	未	3.5
3	技術要素	必須	情報処理	データ処理	サーバ系	サーバアプリケーション系技術: DataBase(Oracle, SQL)	B0020	未	未	未	3.5
4	技術要素	必須	情報処理	データ処理	サーバ系	文書管理ソフト系技術 (DocuShare / ArcShare)	B0030	未	未	未	-
5	技術要素	必須	情報処理	データ処理	その他のデータ処理	ドキュメント管理システム: フラットホーム(DMP)系技術: コローサーバ)	B0040	未	未	未	-
6	技術要素	選択	ストレージ	メディア	-	Disk Drive (HDD等)	H0210	未	未	未	-
7	技術要素	選択	ストレージ	メディア	-	記憶媒体	H0220	2	未	未	-
8	技術要素	選択	計測・制御	制御処理	-	マイコン制御回路	H0230	2	未	未	-

図16 本人アセスメント画面の例

図14 部門課題調査の画面

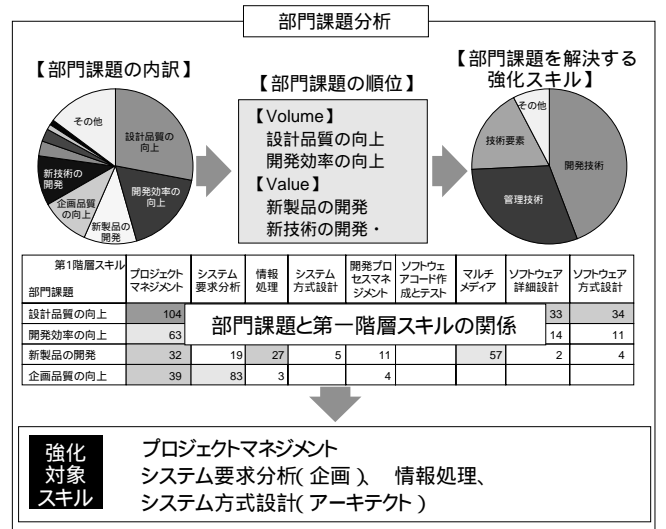


図17 強化対象スキルの抽出

ステップ	項目	担当	実施日	実施内容
0	初期決定	部門	2008/08/21 09:56	上司とアセスメントパターンを設定し...
1	部門課題認識の閲覧 G・T課題認識の閲覧	本人	---	部門長、G長/T長が記入した「部門...
2	強化スキル設定	上司	2009/01/16 08:41	部下の強化スキルを設定(3個まで/...
3	アセスメント	本人	実施日 2009/01/16 08:44 確定日 2008/11/20 00:00	本人アセスメントを実施してください。
4	上司アセスメント	上司	未実施	上司アセスメントの実施
5	面談用結果出力	本人	---	「必須スキル」、「選択スキル」の全...

図15 アセスメントワークフロー

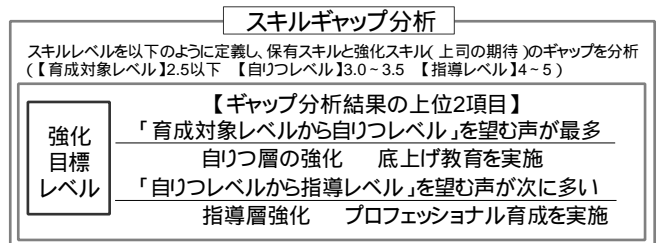


図18 強化目標レベルの抽出

・ 2007～2008年度の部門課題の変化

部門課題と強化スキルの変化分を把握するために、2007～2008年度の変化分を分析した（図19）。その結果、2008年度は「開発効率の向上」と「開発技術のスキル強化」が、より望まれていることを確認出来た。現在、上司が入力した部門課題の言語情報を分析し、教育メニューに反映中である。

以上、ギャップ分析について述べてきたが、現在実施している、分析事例を表3に示す。

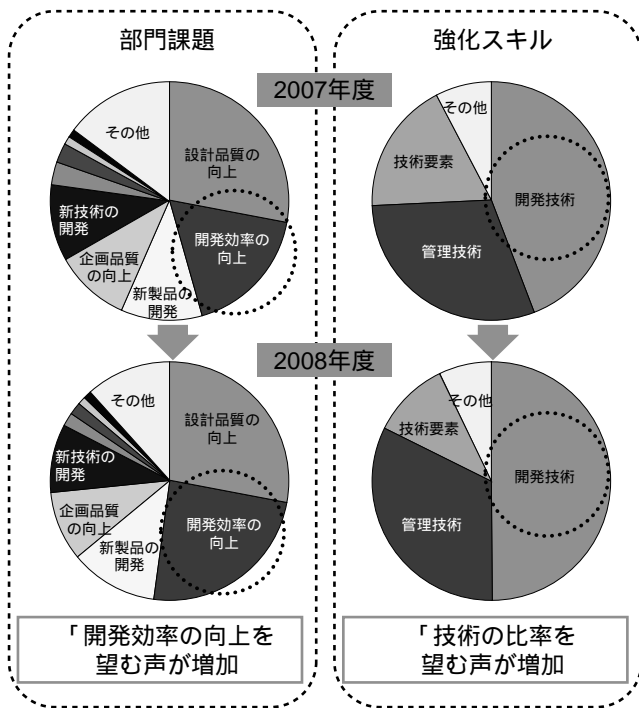


図19 2007～2008年度の変化分析

表3 分析事例

診断内容		概要
1	スキルギャップ分析	技術力の「弱み」分析 技術力の強化ポイント(弱み)を明確にするために、保有スキルと強化スキルのギャップ分析を実施し、強化対象スキルを抽出
2	ポジショニング分析	技術力の「強み」分析 技術力の相対的な、強みと弱みを明確にするために、他社/全社/部門/個人等との比較を行う
3	成長分析	自りつスキル数を年度ごとに「定点観測」することにより、時系列的に「自りつ度の成長率」を測定
4	年代別分析	強化スキル/保有スキルを年代別に分析し、整員全体/部門での整員ポートフォリオ設計に活用できる年代別に技術力の「強み」と「弱み」を把握できる
5	職種別分析	強化スキル/保有スキルを職種別に分析できるので、職種別の特徴の抽出と対応策を検討できる。
6	Webスキル分析Tool	上記の分析を、Web上から実施できるTool

教育メニューの開発

2007年度に実施したスキルギャップ分析で抽出された「強化ポイント」を基に、分野長が中心となり、プロジェクトマネジメント、企画、ソフトウェアエンジニアリングの3分野で約20講座を新規設計し、順次提供中である。

次に教育メニューを開発するための教育プロセスについて述べる。

・教育プロセスの全体像

教育プロセスは、6項目のプロセスと各プロセスを担うプレイヤー（担当者や組織）で構成され、主に「教育メニュー企画プロセス」と、「講座設計/開発プロセス」の2大要素から成る（図20）。

「教育メニュー企画プロセス」の中で、今日的ニーズや明日のニーズからOJT/OffJTの教育メニューの要求仕様を策定し、「講座設計/開発プロセス」の中で、要求仕様を基にOffJTの構成要素である講座を設計/開発する。

富士ゼロックスの主なプレイヤーとしては、SME⁵は分野長/講座長、講座設計者は講座長/講師、という構図である。また、講座開発（集合講座）は講師が担当し、講座開発（WBT講座）は技術人材教育センターが担当している。

技術人材教育センターのメンバは、人材育成/教育の観点で、現場の課題を解決出来るコンサルテーションが可能な、「ラーニング・コンサルタント」を目指している。

・開催中の講座の調査

最初のステップとして、開催中の講座が部門課題を解決する強化スキルを提供しているかどうかを調べたところ

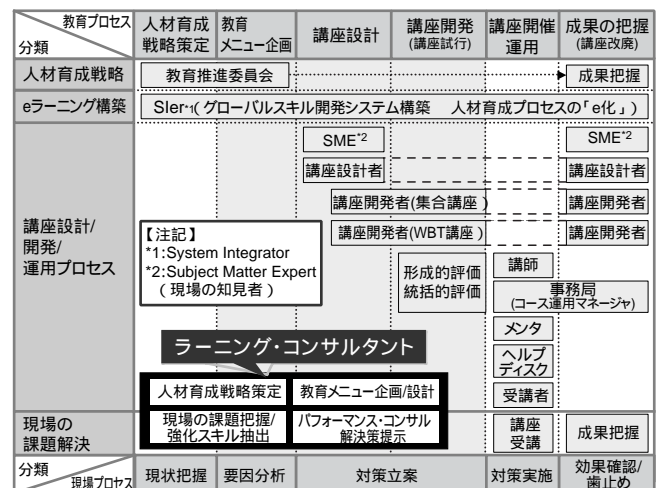


図20 教育プロセス

5 SME : Subject Matter Expert

ろ、とくに「開発技術」と「管理技術」に関する講座を十分に提供出来ていないことが分かった(図21)。

・教育メニュー企画プロセス

教育メニュー企画プロセス(図22)を次に述べる。

(a) 教育推進委員会基本方針作成

教育メニュー企画は、事業戦略、部門課題分析結果、委員長の想い、スキルギャップ分析の結果、等を総合的に判断し、委員長が教育推進委員会の基本方針を策定することから始まる。

次に、分野長/WGリーダーが中心となり、強化スキル群の中から、上位30項目の「強化対象スキル」を抽出した(図23)。

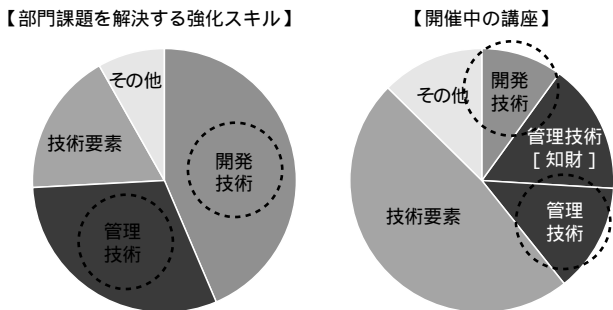


図21 開催中の講座

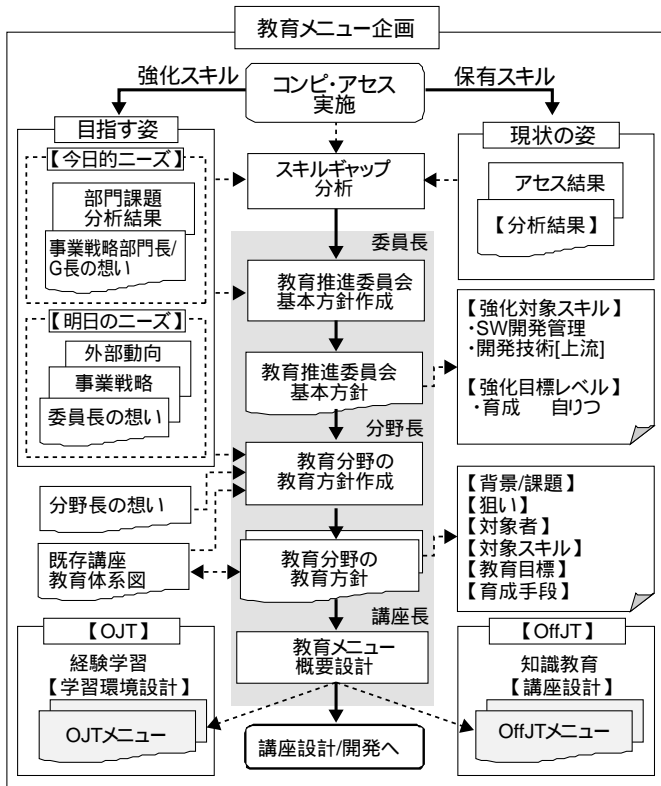


図22 教育メニュー企画プロセス

(b) 教育分野の教育方針作成

分野長は、教育推進委員会の基本方針を受け、担当教育分野のプロフェッショナルとして、強化対象スキルごとに、部門課題分析結果(数値情報)と、部門課題調査シートの内容(言語情報)を読み込んで総合的に分析を実施する(図24)。

分析結果を基に、教育分野ごとに教育方針を作成する(図25)。

(c) 教育メニュー概要設計

講座長は、教育分野の教育方針を受け、教育メニューの概要設計を実施し、経験学習ベースのOJTメニューと、講座(群)を中心としたOffJTメニューを設計する。同時に、新規講座と既存講座の改廃情報を、教育体系図に反映する(図26)。

・講座設計/開発プロセス

講座長/講師は、OffJTメニューを受けて、関連する講座(群)の設計/開発を実施する。

講座設計/開発を進めるに当たり、科学的手法の一種

ID	スキルカテゴリ(大分類)	第1階層スキル(中分類)	対応コード	コピテンシー項目(小分類)
1	開発技術	プロジェクトマネジメント	A00061	プロジェクト維持管理技術能力
2	開発技術	システム方式設計	T01080	システム仕様設計
3	開発技術	プロジェクトマネジメント	P05040	プロジェクトマ
4	開発技術	開発プロセスマネジメント	S08112	CMM ISPD
5	開発技術	ソフトウェア詳細設計	S08040	ソフトウェア
6	開発技術	システム要求分析	S08000	システム分析
7	開発技術	開発プロセスマネジメント	T03316	発明力
8	開発技術	システム要求分析	G00240	顧客情報取
9	開発技術	プロジェクトマネジメント	P05110	プロジェクト
10	開発技術	システム方式設計	S0A020	システムア
11	開発技術	ソフトウェアコード作成とテスト	S08520	C++
12	開発技術	共通スキル	S08111	オブジェクト
13	開発技術	プロジェクトマネジメント	P05080	コミュニケー
14	技術要素	情報処理	S06910	セキュリティ
15	開発技術	ソフトウェア詳細設計	S08045	ソフトウェアの詳細設計のレビュー
16	開発技術	プロジェクトマネジメント	T03120	戦略立案黒箱
17	開発技術	ソフトウェアコード作成とテスト	S08055	コードレビューとプログラムテスト項目のデザインレビュー

図23 強化対象スキル(事例)

<情報源>	強化対象スキル	上位30項目		
		スキル1	スキルn	スキル30
年代(年齢) 職種/グレード				
<部門課題調査分析結果>	部門課題			
	設計品質の向上	126	99	76
	開発効率の向上	10	数値情報	60
<部門課題調査シート>	活動の狙い	xxxx	yyyy	zzzz
強化スキルを 取り上げた理由	対象者	xxxx	yyyy	zzzz
強化対象 スキル	対象スキル	xxx	言語情報	zzzz
育成手段	教育目標 (どうしたいの?)	xxxx	yyyy	zzzz
<スキルギャップ 分析結果>	育成手段	xxxx	yyyy	zzzz

図24 課題認識分析チャート

である、講座設計手法 (ID⁶) を共通言語として活用した (図27)。ID活用により、「魅力的」で「効果的」な講座を「効率的」に設計することが出来るようになった。

ID手法は、しっかり適用するとかなり工数がかかるので、標準設計ガイドラインと、ポイントを押さえた簡易設計ガイドラインを作成し、講座設計時に講座の優先順位を付けて、どちらのガイドラインを適用するかを決めている。

次に、今までの経験から得た、IDの特徴を述べる。

(a) 効率的に講座を設計出来る

- ・ 初心者でも一定レベルの講座を設計することが出来る
- ・ 講座の質を測定出来る
- ・ 設計仕様書を「資産」として残すことが出来るので、他講座設計時に活用出来る
- ・ 多数の関係者で講座を「共同開発」出来る
- ・ 受講者は、教育目標に沿って設計された講座を受講出来るために、効率的に講座を受講出来る

(b) 「効果的」な講座を設計出来る

受講対象者を的確に選定出来るので、「落ちこぼれ」や「吹きこぼれ」を防止出来る。

(c) 「魅力的」な講座により、「受講者のやる気」に火をつける

各種の動機付けモデルを活用することにより、受講者が「主体的に学ぼう」とする意欲を高めることが出来る。

技術者教育の実施

各部門は、(a) スキル診断書を入手することにより、強みと弱み (強化ポイント) を把握する。同時に、(b) 教育体系図と (c) OJT成功事例を入手し、OJTとOffJTの観点から、「弱み」を強化するための対応策を検討する。次に、ISO 9001の力量管理に則り、技術力強化のために教育計画書を策定し、予実管理を実行する。

7 人材育成施策の策定

2007年度の活動の締めくくりとして、人材育成施策を策定した (図28)。人材育成施策は、会社の「経営方針」や技術人材育成ビジョンを部門長が受け、「部門課題」を策定、強化スキルと保有スキルからスキルギャップを分析、強化対象スキル抽出、という一連の方針展開チェーンの結果としてまとめた。現在、自りつ度向上を図るた

項目	概要
1 分野長の想い	・ 「良いISW商品」と「Service Offeringsを支えるSW」を開発する人材育成
2 目指す人材像	・ 基本と時流の技術を兼ね備えた自りつ人材への底上げ、着実な伝承
3 背景/課題	・ 「セキュリティの常態化」「C&SからWeb Applicationへ」「大型案件化」
4 活動の狙い (何のために)	・ 時流の技術を備えた人材の拡大、PF技術と商品技術の着実な伝承担当に留まらない幅広い視野、技術者としての営業現場感覚
5 対象者 (誰のために)	・ SW PF及びSW商品の開発者 ・ Service Offerings向けSWの開発者 ・ SW商品/サービス向け技術開発者 ・ SI開発者 (SE)
6 対象スキル (何を)	・ システムとしてのセキュリティ技術、Web Application技術、可用性・堅牢性等の非機能要件技術、SW商品の外部IF
7 学習目標 (どうしたいのか?)	・ SO6910セキュリティ技術、SO6370Web Applicationのコンピテンシーの自りつ度を向上する
8 育成手段 (どのように教育?)	・ セキュリティ講座の更新、商品開発を題材としたWeb Application講座の新設、商品A/商品Bを題材とした可用性技術の強化、OJT支援
9 教育成果の測定法 (成果をどう測定?)	・ 関連コンピテンシーを計測することで、技術力向上分を測定する。

図25 教育分野の教育方針 (事例)

経験年数目安	- 3年		4 - 7年	8年 -
スキル分類	入門レベル (レベル1 - 1.5)	育成レベル (レベル2 - 2.5)	自りつレベル (レベル3 - 3.5)	指導レベル (レベル4 - 5)
プロジェクトマネジメント		プロジェクトマネジメント基礎 プロジェクト開発/ノウハウ (実践編) プロジェクト開発/ノウハウ (基礎編)	プロジェクトマネジメント CMM ソフトウェア 方法標準 (SSM) CMM (PR) CMM (SCM)	
開発プロセス設定	特許基礎 (特許見学)	特許基礎 特許海外対応 特許出願実務 特許出願演習 知財ライセンス契約基礎	特許応用 特許実践	

図26 管理技術の教育体系図 (事例)

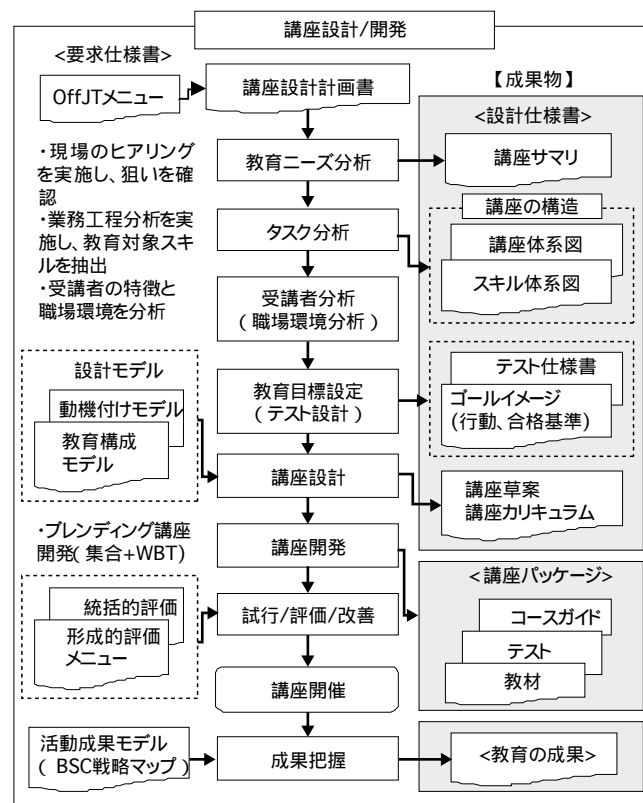


図27 講座設計 / 開発プロセス

めの底上げ教育と、プロフェッショナル人材育成教育を、キャリア開発活動として取り組んでいる最中である。

8 スキル開発システムの全体像

図29に全体像を示すが、スキル開発システムは、コンピテンシーアセスメントシステムと受講管理システムから構成され、当事者である推進委員会の事務局が中心となって、EUC手法をベースに少人数で自前開発したものであるので、使い勝手の良いシステムを、迅速に短期間で開発することが出来た。

今回は、EUC開発の特徴を活かし、両システムの連携機能の強化を図っている。

- 次に、代表的な連携機能を述べる。
- ・アセスメント時に、強化スキル関連の講座を検索し、即申し込むことが出来る
- ・自分のスキルが、全体 / 部門の中でどの位置にあるかを確認することが出来る (ポジショニング分析)

8.1 ETSSベースのスキル開発システムの特徴

今回、ETSSベースのスキル開発システムを構築したことにより、技術者個人と組織の技術力を定量的に把握することが可能になった。次に特徴を述べる。

- ・「部門の課題と解決策」を考える場の提供
マネージャに、人材育成という観点から、部門課題と解決策をじっくりと「考える場」を提供する。
- ・方針展開ツールとして活用
部門長 G長 T長 技術者へ人材育成方針を展開することにより、「上司の期待」の見える化が実現出来、技術者のモチベーション向上につながる。

経営方針	部門課題	スキルギャップ分析結果	人材育成施策
顧客・社会の変化を察知する力 (ニーズ把握)	企画品質の向上	プロジェクトマネジメント力の強化 ・プロジェクト進捗管理技術 ・プロジェクトマネジメント計画	【人材育成の進め方】 底上げ教育 自りつスキル強化のためには、「OJT」と「OffJT」を組み合わせ教育の提供 プロフェッショナル人材育成 指導者スキル強化のため「選抜教育」を実施 キャリア開発 「中長期的」に、底上げ/指導者教育に取り組む
新たな顧客価値を生み出すコア技術の開発/強化 (ベネフィット)	新技術/新製品の開発	システム要求分析力強化 ・システム分析と要求定義 ・顧客情報収集/分析力	
変化に対応するスピード	設計品質の向上	サーバ技術の強化 情報処理技術の強化 ・セキュリティ、Web Application	
(体力強化) 工程能力強化	開発効率の向上	開発プロセスマネジメントの強化 ・CMMI(SPI)、発明力(知財)	
技術人材育成ビジョン	長期的育成指針の策定	システム方式設計の強化 ・システム仕様設計 ・システムアーキテクチャ	
推進体制/学習環境整備	人材育成プロセスの構築/IT化の推進 関連会社(海外/国内)全体の「技術教育」	自りつ/指導者スキル強化 プロフェッショナル人材の育成	【教育メニューの開発】 ・新規:Y分野(SW企画) ・分野:SW開発基礎 ・分野:SW開発管理 ・分野:サービス技術
			【教育環境整備】 eラーニングの強化 グループ企業の技術教育

図28 人材育成施策

【上司】 技術者に「何を求めているのか」を提示
【技術者】 自分が「何を求められているのか」を確認
・スキルポータビリティが向上

共通のスキル標準を採用することにより、部門間で同じスキルセットを活用出来、異動時に「スキルセット」を持ち運び出来るようになった。

- ・グループ企業全体の「技術力の測定」が可能
グローバルレベルで共通のスキル標準を使用することにより、グループ企業全体の「技術力の測定」が可能となり、業務の最適配置を実現出来る。

また、Viewとしての「スキル標準」を入れ替えることにより、エレキ系等の他分野でも活用出来るようになった(図13)。現在、エレキ分野のViewとしてFX-STSSベースのスキル標準を開発中である。

- ・部門課題に直結した教育メニューを提供
部門の課題と、解決するための強化スキルが明確になるので、「課題解決に直結した教育メニュー」を提供出来る。

9 グローバル教育

海外拠点への教育の展開は、グローバルレベルの技術力強化のために、技術力を測定する、強化に必要な

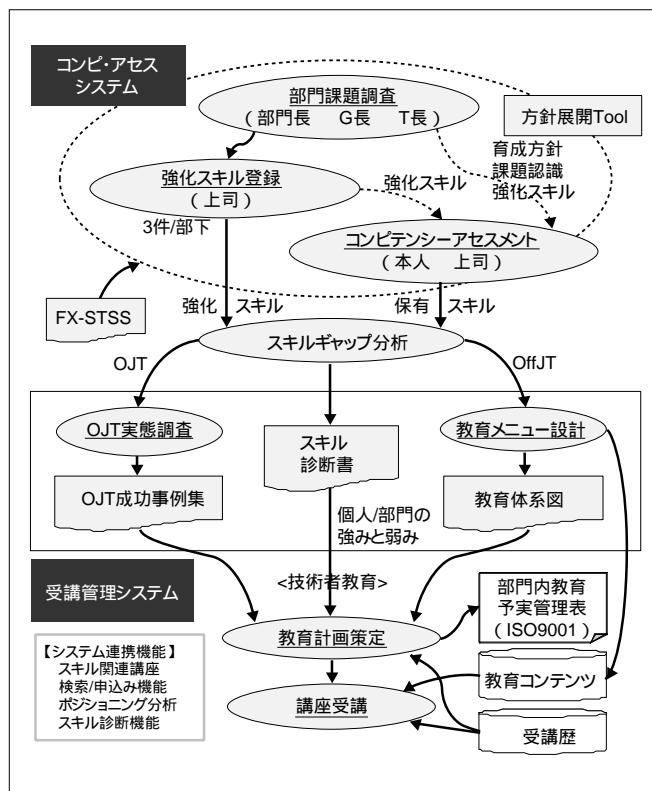


図29 スキル開発システムの全体像

教育メニューを提供する、というステップで実施している（図30）。

グローバル教育に関しては、社内の開発・生産部門のメンバで構成される「海外教育検討タスク」を発足し、G（グループ）活用戦略と同期を取った形で、開発/生産拠点の教育計画を策定している。このタスクで年に一度、各海外拠点の教育計画を調査し、提供する教育パッケージを検討することになっている。調査結果に応じて、日本語の教材を現地の言語に翻訳するサービスも実施している。

技術力の測定

現在、上海のソフトウェアセンターに、FX-STSSとコンピテンシーアセスメントシステムを展開中である。

スキル標準は、FX-STSSを活用し、必須/選択スキルを規定する「パターンファイル」は、パートナーとなるFXの組織のファイルをベースに、拠点のものを開発している。

技術力の測定は、ゆくゆくはエレキ分野やメカ分野にも展開予定である。

教育メニュー提供

現在、中国/韓国の4箇所の開発・生産拠点に対して、集合講座とWBT講座を提供している。WBT講座に関して

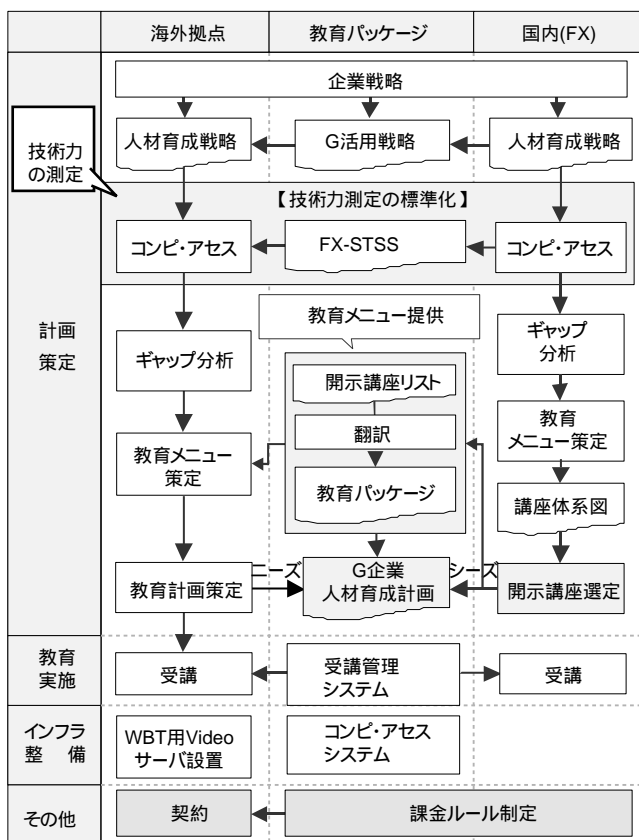


図30 グローバル教育のグランドデザイン

は、拠点間のネットワークのトラフィックを低減するために、Videoサーバを各拠点に分散配置している（図31）。受講管理システムについては、日本側で集中管理するという「集中・分散管理方式」をとっているため、受講履歴やコンピテンシーアセスメントデータ等の一元管理が可能になった。

10 成果把握

(1) 活動成果モデル

活動成果モデル（図32）としては、BSCの戦略マップを活用し、事業戦略、課題解決、キャリア開発、スキル開発をマッピングし、それぞれ成果測定項目と方法を規定している。

ターゲットとなるビジネスモデルには、図2で示した「人材育成のグランドデザイン」を使用した。

今回は、部門課題解決のために抽出した強化スキルの成長度合いを測定するために、「学習（教育）/スキル開発」層で2007年度強化スキルの自りつ度成長率を測定した（自りつ度とは、コンピテンシーレベル3以上のスキル数が全体に占める割合である）。

(2) 2007活動成果の把握

2008年度のアセスメントの終了後に、2007年度に強化スキルとして指定されたスキルと、それ以外のスキルのレベルの成長度合いを調査するために、それぞれの自りつ度成長率を分析した（図33）。結果として、2007年度に指定された強化スキルの成長率が6%となり、それ以外のスキルの2%よりも高いことを定量的に把握出来た。

今回、部門課題解決に直結した強化ポイントを明確にした結果、重点的に強化スキルの自りつ度を向上することが出来たことは、成果の一つと考えている。

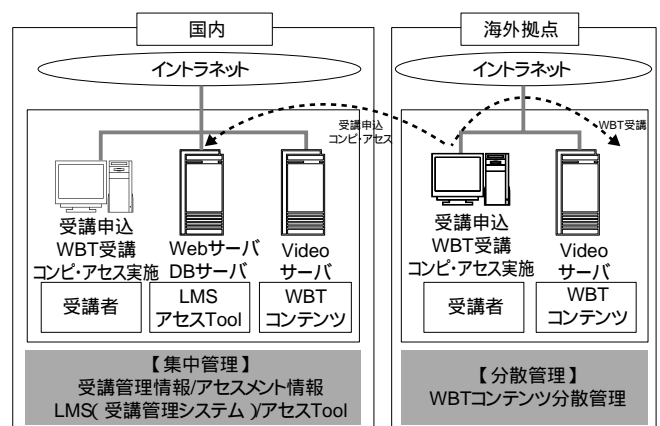


図31 グローバルスキル開発システムの構成

第1階層やスキルレベルでも成長分析を実施した結果、同様に、2007年度強化スキルの自りつ度が確実に向上していることを確認出来た。

(3) アセスメント受診者の反応

アセスメント終了後に、受診者に対してアンケート調査を実施した。集計の結果、約6割の受診者に「今回の活動は有効である」と認めてもらった。次に概要を述べる。

【部下の反応】

- ・マネージャの意思が明示されることは良い。マネージャの考えがより具体的に分かる
- ・自分に何が求められているのかが分かる
- ・自分の得意、不得意分野を定量的に把握出来る

【上司の反応】

- ・ETSSを導入したことにより、今まで漠然と捉えていた部門課題と強化スキルを定量的に捉えることが出来るようになった意義は大きい
- ・部下とのベターコミュニケーションを深めるよい機会

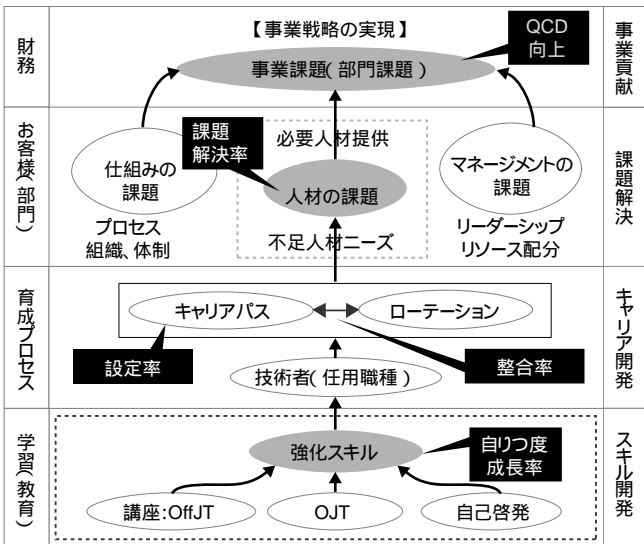


図32 活動成果モデル (BSC戦略マップ版)

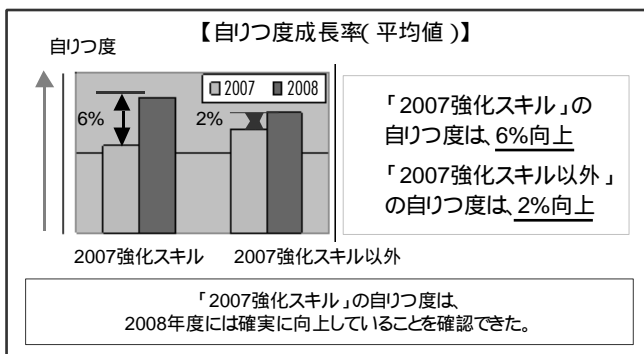


図33 2008年度の自りつ度成長率

になった

- ・現業務以外のスキルについても把握出来、新しい業務をアサインするときの参考になる

【アセスメント受診者の要望】

一方、要望として挙げられた内容を述べる

- ・研修や自己啓発ではスキル向上に限界があるので、業務経験(OJT)との連携が必要
- ・専門的職種(キャリア)の定義と、それに必要なスキルの明確化が必要

11 今後の展開

以上、主にスキル開発システムの概要について述べてきたが、次に今後の計画について述べる(図34)。

・キャリア開発

現在、WG4にて、技術者のキャリア開発フレームワークを構築中である。最初のステップとして、人事部が設定した職種/役割名と、ETSSで設定している職種との親和性を調査した。結果として、人事部が設定した職種/役割名とETSSの職種の親和性は高いということが分かったので、双方の職種のマッピングを取っていく方向で検討を進めている。今後は、エレキ分野の技術者のスキル標準開発と職種設定も実施するので、技術分野をまたがる「基本キャリアフレームワーク」を検討中である。基本キャリアフレームワークは、X軸：職種、Y軸：レベル、Z軸：業務分野

	2006~2007	2008~2009
	スキル開発	キャリア開発
推進会の主な狙い	【教育の方向性の明確化】 部門課題調査 人材育成戦略の策定 【スキル開発】 新規教育メニュー開発 部門教育実施	【キャリア開発】 キャリアパス設定 ローテーション/キャリアパスとの連携 【OJT環境整備】 社内OJT成功事例研究
活動の概要	しくみ(体制) コンピアセス実施のために人事部と連携 分野長主導の下に新規教育メニュー開発	【関連組織との連携】 人事部との連携 HW推進会との連携
主な活動	部門課題調査 スキルギャップ分析 20新規講座開発 OJTベンチマーク調査	職種設定 キャリアフレームワーク開発 キャリアパス設定 OJT社内実態調査
	技術者教育のグローバル展開(中国/韓国の開発・生産拠点)	
	人事の人材育成システムとの整合	

図34 今後の計画

(組込み系、IT系、エレキ系等) で分類することを考えている(図35)。

これにより、業務分野をまたがる移動があった場合に、スキルとキャリアの移動も可能にすることを旨とする(スキル/キャリアポータビリティの向上)。

・OJTへの取り組み

OJTに関するベンチマークを実施したが、自社の文化に合った方法を、意図的(意志を持って)、計画的、継続的に進めていくことが重要であることを改めて認識出来た。

現在、社内のOJTの成功事例とハイパフォーマの行動特性/思考特性を調査するために、OJT実施状況調査を実施している。これにより、社内でベンチマークとなる事例を調査し、「効果的なOJTの進め方」を水平展開する。

・グローバル教育の展開

海外拠点への展開は、今までは主に講座を提供することを主に実施していたが、今後は、グローバル活用戦略との連携を更に図り、必要な技術力を明確にした上で、不足スキル強化のための教育メニューを提供するという一連の流れを各拠点に展開していく。

12 ETSS活用のすすめ

まとめとして、ETSS活用のメリットについて述べる。

今回、ETSSを採用することにより、従来のコンピテンシー辞書の網羅性を高めることが出来、定量的な技術力の測定が可能になった。次に、具体例を述べる。

自社のスキル標準を効率よく開発出来る

・網羅性が高い

ETSSはソフトウェア開発に必要なスキルをまんべんなくカバーしているので、スキルの抜け漏れが少ないスキ

ル標準を開発出来る。今回の事例のように、既存のスキル標準の抜け漏れを確認することも出来る。

・構造が体系的

ソフトウェア開発に必要な要素が体系的に提供されているので、スキル標準を効率的に整理出来、技術力の可視性が高まる。

・構造が階層的

スキルカテゴリ、第1階層、第2階層等、スキル標準が階層的に定義されているので、必要に応じて階層的な分析が可能

スキル比較分析が可能

社内での個人/部門のスキル比較が可能になる。

また、業界標準を採用したことにより、個人/部門/会社のレベルで、社外とのスキル比較も可能になる。

13 おわりに

以上、ETSSの活用事例を述べてきたが、ETSSの採用がきっかけとなり、一年に一度実施している専門コンピテンシーアセスメントを実用的で効果的なものにする事が出来た。今後は、今回培ったノウハウをエレキ等の他分野や、海外拠点にも展開する予定であり、「ETSSの輪」を広げていきたい。

この場をお借りして、ETSSの開発に携わった方々に感謝の意を表すると共に、今後もETSSの普及に努めていただくことをお願いしたい。

参考文献

[SEC2007] IP/ASEC : 組込みスキル標準2007年度版, 翔泳社, 2007

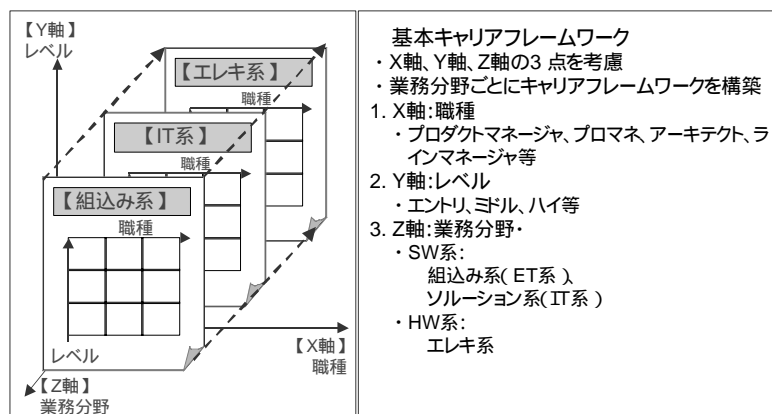


図35 基本キャリアフレームワーク

4 教育効果測定に関するETSSの取り組み

SEC 研究員
渡辺 登

ETSSに基づく人材育成における教育プログラムの活性化を目的とし、教育効果測定に関する取り組みを紹介する。この教育効果測定に関する取り組みは、組込みソフトウェアを対象に組み込みソフトウェアとして行われているが、組込みソフトウェア以外の開発においても適用可能な内容である。

1 教育効果測定の概要と状況

教育効果測定は、教育に対する費用と利益の分析を目的に実施され、企業の場合には、経営戦略と教育の整合性を見る際に利用される。教育に対する投資が必要であることは広く知られている。人的資本[BECKER1976]では、技能形成の教育訓練は投資であり、この投資によって身につけた人材の技能は資本であるとされている。

IPA調査[IPA2008]では人材育成の課題として「育成の費用対効果が分かりにくい」と回答するする企業が多く、これは企業規模が大きければ大きいほど顕著に現れる特徴がある(図1)。

教育効果測定があいまいなままでは、教育投資の先送

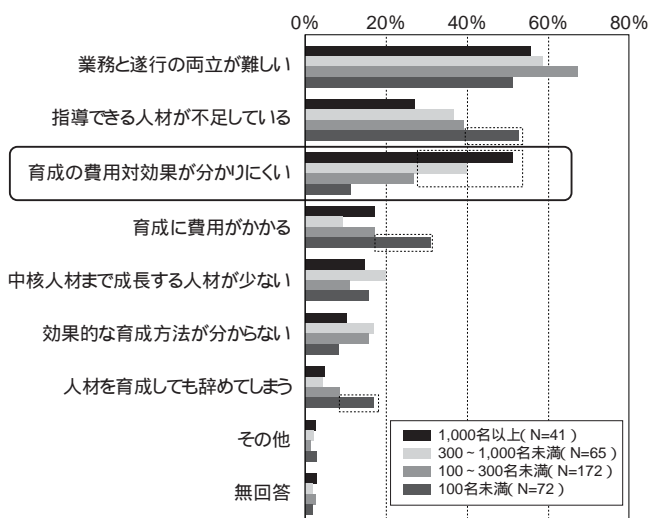


図1 人材育成の課題 (IT人材市場動向予備調査、IPA、2007より)

りを助長することが懸念される。開発現場では、ソフトウェアの大規模・複雑化が進み、これを短期でリリースしなければならない状況が続いている。このような状況下、技術者は忙しく、教育に対する取り組み(教育の企画や参加等)が疎かになる。これにより、次々と適用される技術に対する追従が困難となり、開発の効率化が図れないという負のスパイラルが続く。

これを断ち切るには、投資による積極的な人材育成が求められる。しかし、ソフトウェア技術者に対する教育投資は少なく、諸外国と比較しても少ない。

人材育成担当者は、具体的な教育効果測定の方法や効果が不明であり、経営者や技術者への教育に関する訴求が出来ない。また、開発現場の管理者は、目先の業務に追われ、教育等即効性が見えにくい取り組みには消極的になっていることも影響している。

教育効果測定は必要性を認めるが、実際に行うとなると測定方法や評価の客観性等問題が多い。製造や生産に関する技能教育の場合には、教育前後の量的・質的な変化が容易に測定可能であった。しかし、ソフトウェア開発は、毎回対象が異なることや、ステークホルダによる影響も多く、計測や評価が困難な特徴を持つ。

教育効果測定には、カークパトリック(Kirkpatrick)の評価法が知られている(図2)。この評価法でレベル3以上の教育効果測定を実施している企業は少なく、計測や評価のためのコストが多く必要となることがネックとなっている。2007年度のSECとの共同研究として、NEXCESS¹が行った企業調査でも非常に少ない。また、

1 NEXCESS : 名古屋大学組込みソフトウェア技術者人材育成プログラム

アメリカでも同様にレベル3以上の実施は少ないという [リクルート2004]。

2 ETSSでの教育効果測定に関する取り組み

SECでは、ETSSに基づく人材育成における教育プログラムの活性化を目的とし、教育効果測定に関する次の三つに取り組んでいる。

ETSS教育プログラムデザインガイドブックの発行

ETSS向け教育研修コースを対象とした評価フレームワーク作り

組込みソフトウェア教育効果計測のための調査研究

教育効果測定に関しては、効果測定のアクターとして、次の三者が想定される。それは、受講者（技術者） 管理者（上司） 経営者の三者であり、それぞれモチベーションが異なるため、それぞれに対応する情報提供が求められる（図3）。

2.1 ETSS教育プログラムデザインガイドブック

ETSS教育プログラムデザインガイドブックは、ETSS教育研修基準の教育プログラムフレームワークに準じて、適切に教育プログラムを開発し、教育を実施するための手順、及び注意すべき点を説明している。

内容としては、教育プログラムを開発し、教育を実施、評価するまでの手順がメインとなっている。とくに注目すべきは、経営戦略との整合性と、各段階での評価である。経営戦略との整合性については、教育プログラムデザインの最上流である人材育成計画立案で行われる。経営者の描く事業戦略等から人材育成計画への落とし込みを行う。いくら素晴らしい人材育成を企画・提供しても、組織の戦略から外れている場合には意味が無い。

この上流を確実にやり、各段階での評価を行うことで、的確な教育プログラムの提供が可能となる。これは、システム開発の代表的V字モデルと同じと言える（図4）。

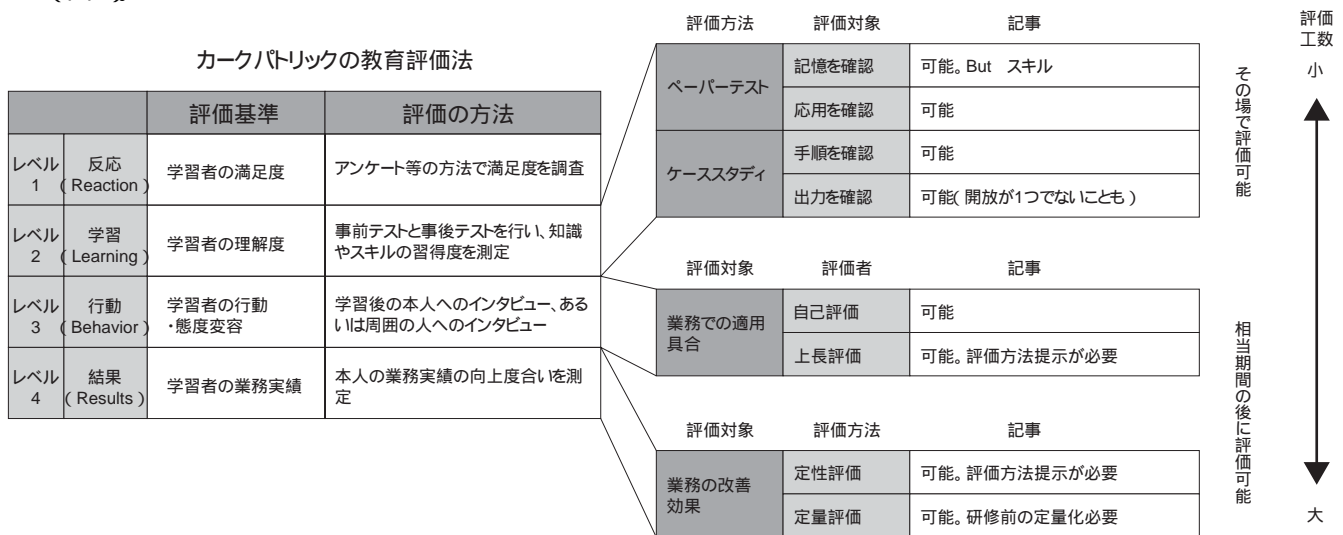


図2 カークパトリック (Kirkpatrick) の評価法

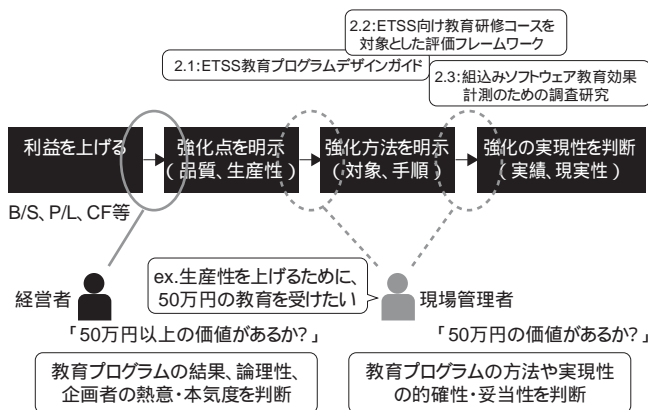
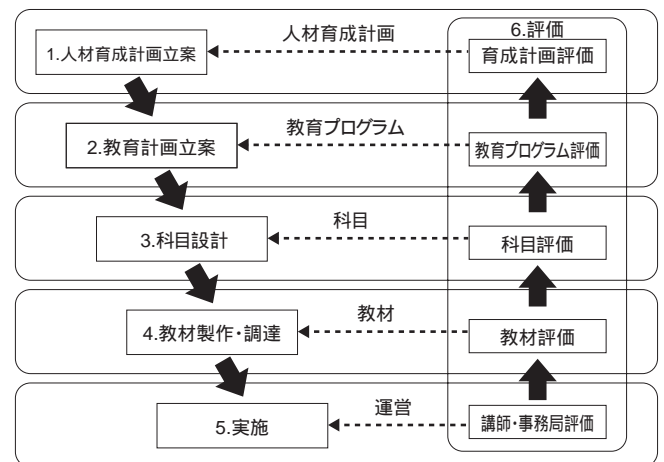


図3 立場による観点の違いと取り組み



出典: ETSS教育プログラムデザインガイド, IPA/SEC, 2007

図4 教育プログラムデザイン: 6つの手順

2.2 ETSS向け教育研修コースを対象とした

評価フレームワーク

この取り組みは、国立情報学研究所との共同研究にて行われたものであり、組込みソフトウェアに関する教育プログラムに対し、受講価値の判断やプログラム設計・改善を支援する評価手法（評価フレームワーク）の策定を目的としている。

受講価値を判断する情報を、受講側に提供しているかどうかを企業インタビュー調査で検証した。企業に提示する受講判断の情報として、教育内容としての技術やスキル以外にも、教育ファシリティ、効果、コスト、資格との関連等の提示が重要であることが分かった（表1）。

これらは、ETSS教育プログラムデザインガイドブックに随時反映していく予定である。

表1 受講判断で重視する項目

ステークホルダ	受講の是非を判断するときに重視する情報
受講者本人	教育対象技術
直属上司	教育対象技術、演習課題（実用レベル）、講師陣（企業人の有無等）、実績、期間や頻度、費用
部門長レベル	受講に必要な知識・能力、習得目標、実績、期間や頻度、費用、修了後の資格

2.3 組込みソフトウェア教育効果計測のための調査研究

この取り組みは、NEXCESSとの共同研究にて行われたものであり、教育の費用対効果の計測に向けた基礎データ取得を目的に、職場における業務遂行能力の評価を用いた教育効果の計測式を検討した。ソフトウェアエンジニア向けコースを受講した受講者と上司に対して、業務遂行能力の評価項目の抽出をしている（表2）。

また、抽出された各評価項目の重要度の評定、そして受講者の各評価項目に対する評定を行っている。教育効

表2 業務遂行能力の評価項目

管理力	E1	品質を維持する能力
	E2	費用を管理する能力
	E3	時間を管理する能力
コミュニケーション力	E4	チームメンバーと共同作業を行う能力
	E5	上司への報告連絡相談の頻度と質
	E6	顧客とコミュニケーションをとる能力
技術力	E7	担当業務を遂行する組込みソフトウェア開発技術力
	E8	担当業務を含む業務全体を捉える能力
	E9	問題発見及び問題解決を行う能力
行動力	E10	業務遂行に好ましい性格
	E11	責任感の強さ
	E12	自己の能力を伸ばす行動力

果測定レベル3以上では、教育以外の要因が多く影響する。よって今後は、研修を行ったチームと研修参加していないチームの比較といった実験群 統制群モデルでグループ分けして評価する等の工夫が必要となる。

3 人材育成の活性化に向けて

これまでの定石や今回の取り組みによって、教育の選択においてはマッチングであり定性的・定量的に評価可能であるが、教育の効果の測定については教育以外の影響要素が多く定量的な評価が困難であることが分かっている。

先進的な企業では、教育効果測定をBSC²とも対応付けしマネジメントしている事例がある[SANRO2006]。例えば、レベル1、2は学習と成長、レベル3はプロセス、レベル4は顧客と収益である等、マネジメントツールと教育効果測定を有機的に結びつけることが、教育効果測定自体の意義を高め、人材育成への投資を含め活性化が可能になる。

教育効果測定において、レベル4として経営に対する効果を導く際、教育対象の技術を、開発現場が組織的に適用することが前提となる。

例えば、UML³のように、教育は充実しているが現場では利用が進んでいない技術も存在する（組込みソフトウェア産業実態調査の結果では約30%）。教育対象技術の現場導入を支援する組織的取り組みが求められる。しかし、開発現場は忙しく、技術者の保有する既存の技能を使うことを好む傾向にある。パイロットプロジェクトによる試行等、教育以外の投資も必要であることを忘れてはならない。

教育学と経営学、そしてソフトウェア工学の三つの知識を駆使することが、教育の活性化による開発力強化には必要不可欠である。今後、SECのETSSに関する活動としては教育プログラムデザインガイドブックの改版等を通じて情報提供をしていく。

参考文献

- [BECKER1976] 人的資本 - 教育を中心とした理論的・経験的分析
 [IPA2008] IT人材市場動向予備調査 報告書,
<http://www.ipa.go.jp/about/press/20080129.htm>
 [IPA/SEC] IPA/SEC : 教育プログラムデザインガイドブック, 2007 (SECBOOKSとして2009年5月に書籍化予定)
 [JACK1999] ジャック J. フィリップス : 教育研修効果測定ハンドブック, 日本能率協会マネジメントセンター, 1999
 [SANRO2006] 産労総合研究所 : 企業と人材 Vol.39 No.883, 2006.6
 [SEC JOURNAL14] IPA/SEC : SEC journal No.14, 2008
 [リクルート2004] リクルート, Works Institute 66, 2004.10-11

2 BSC : Balanced Score Card

3 UML : Unified Modeling Language

ソフトウェア開発プロジェクト計測プラットフォーム EPMの今

各国で活発化するインプロセス計測の動向

SEC 研究員

神谷 芳樹

SEC 研究員

吉川 宏幸

SEC 研究員

樋口 登

SECが積極的に進めてきた進行中のソフトウェア開発プロジェクトを計測し、その状態を可視化してプロジェクト運用やプロセス改善に役立てようという、いわゆる「インプロセス計測」の動きが、世界的にも大きな流れとなってきた。この動きを推し進める仕掛けの面からの鍵が「プロジェクト計測プラットフォーム」である。本特集では、その最新動向を、SECが改良と普及に注力してきた「プロジェクト計測プラットフォーム」EPM¹の今と合わせて紹介する。

なお、本特集は、以下の構成で展開する。

- 1 はじめに
- 2 エンピリカル・ソフトウェア工学研究グループからの発信
- 3 実用レベルへの展開
- 4 EPMの展開
- 5 利用法の進展
- 6 EPMを巡る新しい環境
- 7 インプロセス計測の方向性を巡る論点とEPMの今後

1 | はじめに

ソフトウェア開発の様子は、自然のままでは見えにくい。これを様々な手段で計測の対象とし、計測されたデータを元にプロジェクト推進を運営したい、という考え方は昔からあった。具体的な試みはソフトウェア開発環境の高度化に関心が向いた1980年代に始まったが、計測にはそれなりの手間、すなわちオーバーヘッドがあり、得られる利得が、すぐに直接的には見えにくいため、なかなか発展しなかった。ところがソフトウェア開発環境が進化し、いわゆる機械支援で開発が進められるようにな

ると、この中に計測のための仕掛けを埋め込むことで、ある程度自動計測が可能になり、ここに1つの研究開発領域が生まれた。以下に、その経緯を概観し、次いで活発な現状を示したい。

2 | エンピリカル・ソフトウェア工学 研究グループからの発信

ソフトウェア開発プロジェクトのインプロセス計測に関する研究は、データに基づいた実証的な研究を志向するエンピリカル・ソフトウェア工学の研究グループの中で具体化していった。例えば、この領域の国際会議

1 EPM : Empirical Project Monitor

ISESE²2004 (2004年10月、米国・ロサンゼルス郊外 Redondo Beach)で、2つの発表があった。1つはハワイ大学、Philip Johnson 教授のHakystat、もう1つは日本からのEPMである[EASE][JOHNSON2004][OHIRA ISESE-2004][MITANI2004][鳥居2008]。

Hakystatはソフトウェア開発環境の中の要所にセンサと呼ぶ小さいプログラムを埋め込み、ソフトウェア開発に関する作業を識別出来る一定の種類のAPI³の使用時間を計測し集計、分析し、これをリアルタイムでグラフ等に可視化するものである。例えば、ソース編集の時間、コンパイルの時間、ビルドの時間、その比率等の状況をリアルタイムで把握出来る。複数の人々で刻々と開発作業が進む様子をリアルタイムでモニタ出来、とくにアジャイル開発作業の状況把握等に威力を発揮する。

EPMはソフトウェア開発の中で、開発管理に役立つ道具として最近比較的広く使われるようになった「構成管理システム(版管理システム)」「障害追跡システム(バグ管理システム)」「メール管理システム」の中から、自動的にデータを収集して、これを集計・分析し、グラフ等の可視性の高い形で表示して進行中のプロジェクト運用に直接役立てようという仕組みである。例えば開発したソースコードの行数の推移、検出した障害件数、残留障害件数、障害平均滞留時間、開発グループ内のメール件数等の推移をグラフ化して表示する。進行中のプロジェクトのプロセスとプロダクト、それにコミュニケーションの状況、その推移をリアルタイムに把握し、プロジェクト管理に役立てることが出来る。

いずれの研究もこの時点ではプロトタイプの域を出なかったが、その後大きな展開を見るに至った。

3 | 実用レベルへの展開

ISEMA

Philip Johnson教授は国際会議ISESEがMETRICS⁴と合併してESEM⁵となった第1回の国際会議ESEM2007(2007年9月、スペイン・マドリッド)においてこの領域の包括的な発表を行っている[JOHNSON2007]。そこでは

この領域の研究をISEMA⁶と呼び、HakystatのようなシステムをISEMAシステムと呼んだ。そして、この論文の中でHakystatの現状を報告すると共に、他の5種のISEMAシステムをその公開情報から評価し、こうした方式を考える上で必要な12種のトレードオフについての考察を示した。評価したシステムはEPMの他、6th Sense (米国・6th Sense Analytics社) PROM⁷[PRO] (イタリア・Bolzano Bozen自由大学) ECG⁸ (ドイツ・ベルリン自由大学) SUMS⁹ (米国・Pittsburg Super-computing Center / IBM) である。

この中でHakystatについては、2001年以来研究が続けられ、この時点までにリリースした版が40、アーキテクチャ更新が7版、100プロジェクト以上で適用され、コード量が30,000行になること、また、6th SenseがHakystatのコードの一部を使用して実現され、サービス提供の形で商用化されていることを報告した。また教授は、方式的にHakystatや6th Senseのようにセンサを埋め込んでリアルタイムに計測するタイプを「センサ型」、EPMのようにソフトウェア開発環境のツールの中から情報を取り出す方式を「pull型」と分類した。米国の6th Sense Analytics社がベンチャー資本(Intersouth Partners, Core Capital Partners)の資金を得て、商用サービスを開始したことは画期的であった。

またJohnson教授の12のトレードオフに関する考察も、主に「センサ型」システムに焦点を合わせているが、こうしたプロジェクト計測システムの方式を考える上で示唆に富むものだった。

Software Cockpit

一方、ESEM2007ではドイツからも興味深い発表があった。SECとも共同研究を通して親しいフラウンホーファ協会実験的ソフトウェア工学研究所IESE¹⁰のJürgen Münch氏、Jens Heidrich氏らのグループによるSoftware Cockpitと呼ぶプロジェクトの成果である[CIOLKOWSKI-2007]。この研究はそれまで欧州にあったProject Dashboardの考え方を具体的に発展させたものと考えられる。Project Dashboardは、ソフトウェア開発プロジェクトの状態を表す様々な計測データを一覧出来る形で表示

2 ISESE : International Symposium on Empirical Software Engineering

3 API : Application Interface

4 METRICS : International Software Metrics Symposium

5 ESEM : International Symposium on Empirical Software Engineering and Measurement

6 ISEMA : In-Process Software Engineering Measurement and Analysis

7 PROM : Professional Metrics

8 ECG : ElectroCodeoGrum

9 SUMS : Standardized User Monitoring Suites project

10 IESE : Institute for Experimental Software Engineering

し、プロジェクトマネジメントに役立てていこうという考え方で、そのルーツはVector社のようなドイツの自動車産業の中で活発なソフトウェアツールベンダのようである[DUMKE2007]。

Cockpitプロジェクトでは、ソフトウェア開発組織の中にPCC¹¹と呼ぶ組織を作り、ここにプロジェクト計測に関する道具立てと計測と分析の専門家を集約し、プロジェクト運用のニーズに合致した計測と分析、その結果を提供する実験を行った。

この研究の全貌とその展開は翌年の国際会議ESEM2008(2008年9月、ドイツ・カイザースラウテルン) Mensura¹²2008(2008年11月、ドイツ・ミュンヘン)で明らかにされた[CIOLKOWSKI2008][HEIDRICH2008]。Cockpitプロジェクトは公的な資金を得て7社で開発コンソーシアムを構成し、3年間(2005-10~2008-5)かけてSoftPitというプロジェクト計測/分析システムを開発し、成功裏にプロジェクトを終了した。そしてこのツールを核に先に述べたPCCというサービスコンセプトを固め、研究レベルではあるがコンサルテーション事業を開始した。この方式では、計測ツールやPCCそしてその運用に関するノウハウはコンサルテーション企業内にあり、経験と共に技術やノウハウが蓄積されていく方式となる。背景に、インプロセスのプロジェクト計測と分析、フィードバックという行為には、プロジェクトの計測、計測ツールのハンドリング、計測ニーズの把握と、ニーズに合致したカスタマイズした分析結果の提供等多くの知識、技術を必要とし、単純にソフトウェア開発組織に計測ツールを配布していく方法では普及しない、という考え方がある。

SoftPit

またHeidrich氏らはこの計画と平行してSoftPitと名付けたワークショップを呼びかけ、2007年に第1回会合をドイツで開催した(SOFTPIT¹³2007)[SOFTPIT2007]。この会合では日本を含む各国からこの領域の6編の論文が発表されている(ドイツ・カイザースラウテルン大学/IESE、ドイツ・ブランデンブルグ大学、カリフォルニア

大学/リオデジャネイロ大学、ウーン大学、奈良先端科学技術大学院大学、IBMハイファ研究所)。この会合は2008年は開催されなかったが、2009年には企画中とのことである。

新しい展開、Spagoと6th Sence

このように、インプロセス計測に関して日本のEPMだけでなく、米国や欧州でそれぞれの視点から研究が進められてきたが、昨年2008年秋になって更に新しい展開が見られた。

アジア太平洋のソフトウェア工学の国際会議であるAPSEC¹⁴2008(2008年12月、中国・北京)に併設して、日本の研究グループの呼びかけで開催されたソフトウェア開発プロジェクトの説明性と追跡性に関するワークショップATGSE¹⁵2008において、日本からの発表と合わせて2つの興味深い発表があった。1つはイタリア・ミラノ大学のSpago、もう1つは6th Sense Analytics社の北京支部からの発表である[ATGSE2008][DAMIANI2008]。

Spagoはプロジェクト計測データを集中型のセンタに集め、自動分析したビジュアルな結果をプロジェクトに提供するもので、ソフトウェアはオープンソースで提供されている。SpagoについてはAPSEC2008でも発表されたが、研究グループはWebサイトを活用する等して精力的な普及活動を展開している[COLOMBO2008][SPAGOBI]。ミラノ大学のSESAR Lab.¹⁶が研究の中心となっており、例えばCMMIの各レベルに対応した計測項目を抽出し、その項目と自動計測対象ツールの例との対応をまとめ、発表している[CMMI]。こうした基礎研究を受けて、Spagoが開発・提供されている。1プロジェクトではなく、マルチベンダ、マルチプロジェクトで均質でない開発プロジェクト群全体のモニタと評価を対象としている。

6th Sense Analytics社は商業ベースに乗せたサービス事業を展開している。用意しているデータ収集用のセンサの対象は非常に幅広い。また、サンプルのデータを送れば30分で結果を送り返すというPRも実施している。本拠地は米国・ノースカロライナ州のリサーチ・トライアングルと呼ばれる地域のMorrisvilleであるが、北京にまで事業

11 PCC : Project Control Center

12 Mensura : International Conference on Software Process and Product Measurement

13 SOFTPIT : Workshop on Measurement-based Cockpits for Distributed Software and Systems Engineering Projects

14 APSEC : Asia-Pacific Software Engineering Conference

15 ATGSE : Workshop on Accountability and Traceability in Global Software Engineering

16 SESAR Lab. : Software Engineering and Software Advanced Research of Department of Information Technology - University of Milan

展開している活発さである。この6th Sense Analytics社が本稿執筆中に買収されてしまった（2009年1月）。買収したのは米国・コロラド州ボルダーのRally Software社である。この企業は各種のソフトウェア・プロジェクト管理用のツール提供と、これを用いたSaaS型のサービスを提供している。プロジェクト計測の商用展開もベンチャー起業段階から第2世代に入った感触となった[RALLY]。

4 | EPMの展開

各国でインプロセス計測の研究と商用化を含む実用レベルへの展開が進む中で、EPMは日本の産業構造に適合した発展を見た。

(1) EPMの歩み

EPMは、初めに文部科学省の委託研究である「e-Society基盤ソフトウェアの総合開発計画（2003年4月-2008年3月）」の1つのプロジェクトとして、大阪大学と奈良先端科学技術大学院大学が中心になって進めた産学連携プロジェクト、「データ収集に基づくソフトウェア開発支援システム計画（略称：EASE¹⁷プロジェクト：代表、鳥居宏次 奈良先端科学技術大学院大学 学長（当時）」）で開発された。開発された製品は共同研究等の形で幾つかの実証に供された後、オープンソースで公開された[OHIRA MSR2004]。

SECでは連携して活動していたEASEプロジェクトからの提案を受けて、これをプロジェクト可視化施策の重要ツールと位置付け、他の手法、ツール群と合わせて、「SEC先進ソフトウェア開発プロジェクト」と名付けた実証プロジェクトに適用した。「SEC先進ソフトウェア開発プロジェクト」は経済産業省の委託により実用水準の「プローブ情報システム¹⁸」を開発するプロジェクトで、大手システムインテグレータ、ソフトウェア企業、自動車関連メーカー等7社でソフトウェアエンジニアリング技術研究組合（略称：COSE¹⁹）を構成し、典型的なマルチベンダ体制で進められた。「SEC先進ソフトウェア開発プロジェクト」は規模こそ中規模であるが、大型開発に

準じる開発体制が生まれ、EPMの適用はこうした開発においてプロジェクトの透明性を高め、プロジェクト管理上有効性が高いことが実証された[SEC2007]。

またEPMは、EASEプロジェクト参加企業の1社のNTTソフトウェア株式会社で自社の管理システムに組み込む形で全社適用され、その一端が、EASEプロジェクトの主催する研究会や、SEC設立3周年のフォーラムで発表されて産業界にインパクトを与えた[EPM]。

SECではこれらの成果を背景に、2007年には「ITプロジェクトの『見える化』」と名付けたプロジェクト可視化施策の一環として、EPM及び関連ツール群の配布とその手法の普及を進めることとした。そのために、EPMの品質及び機能の大幅な向上を図り、商用レベルの製品に高めて、「EPM検証プロジェクト」として一般企業への媒体による配布体制を実現した[EPM][SEC IT2007]。

このIPA版EPMはその後版を重ね、現在V2.0版を提供中、「EPM検証プロジェクト」には累計65企業が参加している。IPA/SECでは引き続き機能拡充計画を準備中である。このEPM検証プロジェクトでは、出来るだけ「SEC先進ソフトウェア開発プロジェクト」での環境をどの企業でも再現出来るように、関連ツールセットやドキュメントの提供等の工夫をした。

(2) 新しい展開

SECでは、EPMあるいは、ソフトウェア開発プロジェクトの『見える化』施策について、展示会、説明会、出版等を通して、機会を捉え、あるいは機会を作って普及に努めてきた。こうした中で次のような展開があった。

産業界組織でのEPM検証

情報サービス産業協会（JISA）では、会員企業の中で公募によりワーキンググループを組織し、情報交換しながらそれぞれの企業でEPMを導入して評価を進めた。2007年度は7社が参加し、2008年度は12社に拡大して進行中である（SEC journal No.15号に事例解説を掲載している[田村2008]）。この中には、PMOによる全社展開を狙っている企業もある。

また、同じく産業界組織であるITA²⁰においても、「プ

17 EASE : Empirical Approach to Software Engineering

18 プローブ情報システム：都会で走行中の自動車等の移動体を情報収集用のプローブに見立て、ここから例えばその位置情報等を無線を介してして収集し、交通情報の把握等に役立てるシステム。

19 COSE : Consortium for Software Engineering

20 ITA : Information Technology Alliance

プロジェクト見える化ワーキンググループ」が組織され、この中で2社がEPMの評価に取り組んでいる。

「組込みソフトウェア領域」での実証プロジェクトへの適用

「SEC先進ソフトウェア開発プロジェクト」が「エンタプライズ領域」だったのに対し、「組込みソフトウェア領域」で次の実証プロジェクトが企画された。「高信頼組込みソフトウェア開発プロジェクト」(略称、「国プロ」)で、やはり経済産業省の委託で、自動車産業に関わる業界団体のJasPar²¹が受託して進めている。2007年度から3年計画で進められ、具体的にECU²²と呼ばれる車載コンピュータの標準ミドルソフトウェアを開発している。OEMベンダ、tier1サプライヤ、ソフトウェアベンダと呼ばれる産業界の階層構造の中で、EPMとその他のツールを組み合わせたプロジェクトの可視化が図られている。計測データは主にグラフの形で可視化され、定期的に各リーダーにフィードバックされている。ソフトウェア開発に関して必ずしも透明度の高くない産業構造の中で、大きな成果が期待出来る状況にある。

重要システムでの計測

情報システムの社会的な重要性が高まるのに伴い、これを支えるソフトウェアの信頼性への要求が高まっている。こうした中で、関係方面の協力を得て、ユーザ企業からの調達型で進められた社会的な影響度の高いシステムの開発をEPMによって可視化する機会があった。マルチベンダで進められた開発工程の中で、結合試験、総合

試験工程をモニタし、この工程の状況の把握に役立つだけでなく、1つ前の設計・製造工程、すなわち、中流工程の課題を浮き彫りに出来ることを実証した。ソフトウェア開発は調達側、ベンダ企業側協力して進めることが推奨されているが、ともすると中流工程は開発ベンダ企業に任せきりになり、調達側からの透明度は低くなりがちである。こうした中で、EPMによる結合・総合試験工程の計測はプロジェクト管理に新しい視点を与えることになった。

5 | 利用法の進展

EPMはプロジェクトの開発管理に関わるツールであるために、個人や小グループの判断で扱いやすい単機能のソフトウェアツールと異なり、その本格導入には技術面だけでなく、マネジメントの領域、あるいは企業経営に関わる施策が必要になり、比較的導入障壁が高い。こうした中で進められた幅広い評価により、本格導入に向けて様々な工夫が試みられた。

プロジェクトの観測の高度化

EPMがグラフ等の形で可視化する情報からより深くプロジェクト進捗の状況を読み取ろうとするものである。例えば、EPMの他に、WBS²³を基盤とした進捗管理システムの出力をEPMのレポート機能を用いてEPMのグラフ上に重畳させ、自己申告による進捗報告データと対比させたり、前述のように、結合・総合試験工程の観測から、

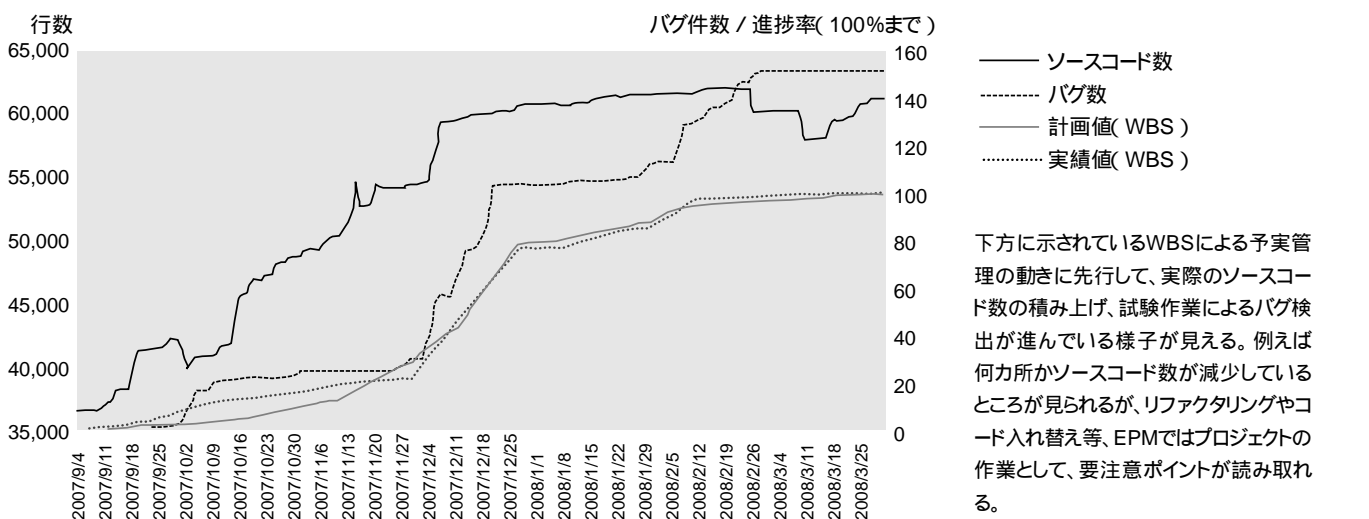


図1 EPMの出力レポート例：ソースコード行数、累積バグ、WBSによる進捗の予実の推移

21 JasPar : Japan Automotive Software Platform Architecture
22 ECU : Electronic Control Unit

23 WBS : Work Breakdown Structure

その前の中流工程の問題点を浮き彫りにしたりする工夫がある。図1に、WBSによる進捗データと重畳したレポート表示例を示す。

全社適用への準備

ソフトウェア企業において、EPMを1プロジェクト、1開発グループへの適用ではなく、多数のプロジェクトが平行して進められている全社に適用し、全体状況の把握と共に、プロジェクト比較による異常値の検出を狙うものである。そのためには、社内の多様なソフトウェア開発環境と整合させ、既存の各種の社内管理システムと接続していく必要がある。EPM利用企業の側で、データ整合用のトランスレータの開発等が試みられている。

コクピット型観測

EPMには収集したデータに対してリレーショナルデータベースを活用した絞込み分析機能がある。この機能を利用して、1プロジェクト全体の観測ではなく、開発個人別、グループ別、あるいは別の視点で、モジュールのようなプログラムの構成上の属性別、あるいはこれらを組み合わせた実現機能別、業務別等の視点からの分析が可能である。そしてこれらを一望出来る形に出力し、言わば大型産業施設の中央制御装置のコクピットのような機能の実現を狙うものである。背景に、一般にプロジェクトの異常は全体に拡散しているよりも、何らかの属性で見て局所化している場合が多く、全体的な視野からこれをいち早く掴もうという考え方がある。

6 | EPMを巡る新しい環境

開発着手から5年余を経て、EPMを巡る環境にも新しい動きが出てきた。

SECのプロジェクト「見える化」施策への組み込みと海外アピールの開始

SECでは進行中のソフトウェア・プロジェクトの運用を直接支援することを目的に、プロジェクトの可視化を図る総合的な手法とツールをまとめ、いわゆる「見える化」施策として普及活動を始めた。この施策は、定性的アプローチ、定量的アプローチ、統合アプローチから構成され、EPMのような計測ツールを用いたインプロセス計測を定量的アプローチの核となる施策に位置付けてい

る。SECではこの施策を4冊の書籍、ダウンロード可能なデータやツール、そしてEPMと関連ツール群のセットとして一般利用可能な形に組み上げ広く提供している[SEC2007 IT][神谷 IT2008][神谷2008]。

また、我が国で使用されるソフトウェアが、他の工業製品と同様に幅広く海外で開発されていることを考えて、この「見える化」施策の海外へのアピールを開始した。これまで2つの国際会議でその考え方を発表し、“*M I E R U K A*”という表現をPRした[OHTAKA2008][MITANI2-008]。将来、既に国際的に知られている“*K A I Z E N*”と並んで認知されることを狙っている。

要求定義工程、設計工程への適用拡大努力

EPMによるモニタリング機能は、当初よりプログラミングやデバッグ作業等、製造工程に焦点を合わせていたために、常にその上流の要求定義工程や設計工程への拡張が要求されてきた。SECでは、この要望に応じて、研究レベルであるが要求定義工程の生産物のインプロセス計測実験や、UML²⁴ダイアグラムの計測調査等を行った。そして、エンタプライズアーキテクチャモデリングの考え方に沿って各種のダイアグラムを大量記述する場合には、その記述量を計測していくことで、製造工程と類似のモニタリングが可能で、かつ一定の有用性のあることを明らかにしてきた[MITANI2007]。その結果をEPMの機能拡充に反映させ、幅広い工程に対して一貫した計測環境を実現出来るよう準備を進めている[MITANI ESEM2007]。

工事進行基準からのニーズ

新年度からソフトウェア開発についても建設工事等と同様に工事進行基準が適用される。工事進行基準では、ソフトウェア開発の進捗度を精度高く、説明性高く示していく必要がある。これと合わせて各企業では内部統制が厳格に運用され、いわゆる遡及処理は出来なくなっている。こうした状況で、WBSに象徴されるような消費稼働や自己申告を基盤とした進捗把握には限界があり、EPMのような開発現場における、恣意的要素の介入余地の少ない計測データに基づいた進捗把握が必須となってきた。

新用途、トレーサビリティの登場

ソフトウェア開発のインプロセス計測に対して、進行

24 UML : Unified Modeling Language

中のプロジェクトの管理やプロジェクト終了後のプロセス改善への反映という用途に加えて、近年、ソフトウェアの信頼性への社会的な要請の高まりで、「説明性、追跡性の確保」という新しいニーズが生まれた。ソフトウェア開発プロセスを正確に記録し、重大事故の原因究明、紛争処理、そして各種の説明に役立てようというものである。また、こうした記録を残すことで、開発プロセスへ一種の牽制効果を与え、その信頼性、生産性向上を図るという狙いもある。

SECでは、このテーマに真正面から取り組んでいるStagE²⁵プロジェクトと連携して、総合的なプロジェクト計測の枠組みについて検討を進めている[MITANI2008]。StagEプロジェクトは文部科学省の委託で奈良先端科学技術大学院大学と大阪大学が中心になって進めている研究プロジェクト（「エンピリカルデータに基づくソフトウェアタグ技術の開発と普及プロジェクト」、代表：松本健一 奈良先端科学技術大学院大学教授）で、ソフトウェア・プロジェクトに関する計測データをソフトウェアタグと呼ぶ媒体に記録して、ソフトウェアと共に流通させようというもので[StagE]、昨年そのタグ規格案を発表している[StagE2008][StagE NAIST2008]。またStagEの研究グループでは自ら前述の国際ワークショップATGSEを主宰し、研究の国際展開を狙い、SECもここに参加している[ATGSE2008]。

7 | インプロセス計測の方向性を巡る 論点とEPMの今後

インプロセス計測の方向性についても、それぞれ異なった視点から幾つかの論点が明らかになってきた。これらと合わせてEPMの今後の課題について述べる。

「見える化」のありがたさ、嬉しさへの濃淡

EPMの実用性、有用性が実証されていくにつれて、その普及を進める者にとって、プロジェクト可視化に対する「ありがたさ」「嬉しさ」への感情的な濃淡が感じられるようになった。もともと「見える化」を生産性や品質向上の武器としてきた製造業、組込みソフトウェアの領域では、ソフトウェアプロセスの「見える化」に抵抗は少なく、透明性を高めることは素直に歓迎された。一方、

多重階層構造を持つエンタプライズ系ソフトウェアの受託開発の分野では、評価は分かれている。「見える化」を戦略的な手法、道具立てと考え積極的に評価、活用法の開拓を進める企業群と、必ずしも歓迎しない企業群が見られる。後者には、現状の産業構造のまま、開発プロセスの可視化だけ進められても困る、という本音が伺える。この課題に対しては、計測ツールや方法論だけでなく、産業構造の革新を含めた総合的な取り組みが必要である。

ツール配布VS.サービス提供

SECではEPMをツールとして配布する方式で普及を進めている。その普及のペースはゆっくりしたものである。これに対して、ドイツのIESEの指摘にもあるように、インプロセス計測とその活用には多くの技術や知識が必要で、これをソフトウェア企業やユーザ企業に直接委ねていくのは現実的でない、という考え方がある。この課題への解が、インプロセス計測とフィードバックをサービスの形で提供し、開発プロジェクトはこうしたサービスを利用しながら、自らのニーズに合致したカスタマイズされたフィードバックを得る方式である。EPMにおいても現実に「4 EPMの展開」で述べた実証プロジェクト等では、この方式に近い形で適用され、その有用性が実証されてきた。EPMの普及策においても今後考えていかなくてはいけない方向である。またツールへの要求条件も配布方式とサービス提供方式では操作性や運用性等で異なるところが多く、この視点からもEPMの今後の展開を考えていく必要がある。

自前分析VS.コンサルティング・サービス

ソフトウェア開発組織が自らデータ分析、可視化を図り活用していくか、こうした活動に精通したコンサルティング・サービスを活用するか、という選択肢である。

でも述べたように、この課題には多くの技術や知識を必要とするので、コンサルティング・サービスの活用も重要な選択肢となる。とくに、導入初期や、ユーザ企業においては現実的な解と言える。そして、そのためのサービス体制を実現することが課題となる。欧州や米国と異なり、コンサルティング・サービスの活用が活発でない日本においては、経済的な面も含めて現実的なコンサルティング・サービスの実現が求められる。

25 StagE : Software Traceability and Accountability for Global Software Engineering

現場分散型VS.集中型

プロジェクト計測とフィードバックの物理的な環境をソフトウェアの開発現場で自己完結的に実現・運用するか、集中型で用意し、データをいったん集めて分析、結果をそれぞれの現場に配布するかという選択肢である。常時多数の開発が併走する大型ユーザ企業や、中規模以上のソフトウェア企業においては、集中型は重要な解である。この場合でもツールへの要求条件に異なるところがあり、EPMの機能拡充で考えていくことが求められる。

以上、4つの視点からの論点を示したが、この視点は直交するものではなく、組み合わせさせて現実の課題となっている。

EPMについては本稿執筆時、次の機能拡充に向けて準備作業と、幾つかの企業での事業化の検討が進められている。SECではここに述べた視点を踏まえて、その機能拡充と手法の普及を進めていく計画である。

謝辞

日頃EPMの開発とサポートに尽力され、EPMの出力画面を提供いただいた、キャッツ株式会社、日立システムアンドサービス株式会社、株式会社SRA先端技術研究所の関係各位に心より謝意を表します。

参考文献

- [ATGSE2008] ATGSE2008 : <http://sdlab.naist.jp/ATGSE2008/>
 [CIOLKOWSKI2007] M. Ciolkowski, J. Heidrich, J. Munch, F. Simon, and M. Radicke : Evaluating Software Project Control Centers in Industrial Environments. (ESEM 2007) , pp.314-323, Madrid, Spain, 2007-9
 [CIOLKOWSKI2008] M. Ciolkowski, J. Heidrich, F. Simon, M. Radicke : Empirical Results from Using Custom-Made Software Project Control Centers in Industrial Environments. (ESEM 2008) , pp.243-252, Kaiserslautern, Germany, 2008-9
 [CMMI] CMMI Metrics Framework : <http://sesar.dti.unimi.it/CMMImetrics/>
 [COLOMBO2008] Alberto Colombo, Ernesto Damiani, Fulvio Frati, Sergio Oltolina, Karl Reed, Gabriele Ruffatti : The Use of a Meta-Model to Support Multi-Project Process Measurement, APSEC 2008, Beijing, pp.503-510, China, 2008-12
 [DAMIANI2008] Ernesto Damiani, Frati Fulvio, Sergio Oltolina, Gabriele Ruffatti : Improving Software Process Accountability with Spago4Q, ATGSE 2008, pp.13-14, Beijing, China, 2008-12
 [DUMKE2007] Reiner Dumke, Christof Ebert : Software Measurement: Establish-Extract-Evaluate-Execute. (Chapter 8.3) Springer-Verlag, 2007-9
 [EASE] EASE プロジェクト : <http://www.empirical.jp/>
 [EPM] EPM検証プロジェクト : <http://sec.ipa.go.jp/tool/epm.html>, NTTソフトウェア株式会社EPM社内適用
 [HEIDRICH2008] Jens Heidrich, Jurgen Munch : Implementing Software Project Control Centers. An Architectural View, IWSM/MetriKon/Mensura 2008, pp.302-315, Munich, Germany, 2008-11
 [JOHNSON2004] P. M. Johnson, H. Kou, J. M. Agustin, Q. Zhang, A. Kagawa, and T. Yamashita : Practical automated process and product metric collection and analysis in a classroom setting: Lessons learned from Hackstat-UH. International Symposium on Empirical Software Engineering (ISESE 2004) , Redondo Beach, pp.136-144, CA, 2004-8
 [JOHNSON2007] Philip M Johnson : Requirement and Design Trade-offs in Hackstat:

- An In-Process Software Engineering Measurement and Analysis System. (ESEM 2007) , pp.81-90, Madrid, Spain, 2007-9
 [MITANI ESEM2007] Yoshiki Mitani, Tomoko Matsumura, Mike Barker, Seishiro Tsuruho, Katsuro Inoue, Ken-Ichi Matsumoto : Proposal of a Complete Life Cycle In-Process Measurement Model Based on Evaluation of an In-Process Measurement Experiment Using a Standardized Requirement Definition Process, International Symposium on Empirical Software Engineering and Measurement (ESEM 2007) , pp.11-20, Madrid, Spain, 2007-9
 [MITANI2004] Yoshiki Mitani, Mike Barker, Seishiro Tsuruho, Koji Torii : An Experimental Framework for Japanese Academic-Industry Collaboration in Empirical Software Engineering Research, International Symposium on Empirical Software Engineering (ISESE 2004) Vol.2 pp.35-36, (Poster session) Redondo Beach CA, 2004-8
 [MITANI2007] Yoshiki Mitani, Tomoko Matsumura, Mike Barker, Seishiro Tsuruho, Katsuro Inoue, Ken-Ichi Matsumoto : An Empirical Study of Process and Product Metrics based on In-process Measurements of a Standardized Requirements Definition Phase, LNCS4895, Software Process and Product Measurement, IWSM-Mensura2007, Revised Papers, (Selected Papers) , pp.46-59, 2007
 [MITANI2008] Yoshiki Mitani, Hiroshi Ohtaka, Ryoza Nagaoka, Hiroyuki Yoshikawa, Seishiro Tsuruho : A Proposal for Integration of In-process Project Visualization and Keeping Post-process Traceability : "MIERUKA" Method and "Software Tag" Framework, Workshop on Accountability and Traceability in Global Software Engineering (ATGSE 2008) , pp.11-12, Beijing, China, 2008
 [OHIRA ISESE2004] Masao Ohira, Reishi Yokomori, Makoto Sakai, Ken-ichi Matsumoto, Katsuro Inoue, Michael Barker, Koji Torii : Empirical Project Monitor:A System for Managing Software Development Projects in Real Time, International Symposium on Empirical Software Engineering (ISESE 2004) Vol.2 pp.37-38 (Poster session) Redondo Beach, CA, 2004-8
 [OHIRA MSR2004] Masao Ohira, Reishi Yokomori, Makoto Sakai, Ken-ichi Matsumoto, Katsuro Inoue, Koji Torii : Empirical Project Monitor : A Tool for Mining Multiple Project Data, International Workshop on Mining Software Repositories (MSR 2004) , pp.42-46, Edinbargh Scotland, UK, 2004
 [OHTAKA2008] Hiroshi Ohtaka, Ryoza Nagaoka : Visualization of IT Project - MIERUKA -, The 4th International Project Management Conference (ProMAC 2008) , Anchorage, USA, 2008
 [PRO] PRO Metrics : <http://www.prom.case.unibz.it/>
 [RALLY] Rally Software : <http://www.rallydev.com/company/>
 [SEC IT2007] IPA/SEC : ITプロジェクトの「見える化」下流工程編, 日経BP社, 2007
 [SEC2007] IPA/SEC : ソフトウェアエンジニアリングの実践 - 先進ソフトウェア開発プロジェクトの記録, 翔泳社, 2007
 [SOFTPIT2007] SOFTPIT2007 : <http://www.soft-pit.de/tiki-index.php?page=SoftPitWorkshop2007>
 [SPAGOB] SpagoBI : <http://www.spagobi.org/ecm/faces/public/guest/>
 [Stage] Stage Project : <http://www.stage-project.jp/>
 [Stage NAI2008] Stage Project, NAIST, Osaka Univ., Software Tag Standard V1.0, Draft English Translation V0.7, 2008-10
 [Stage2008] StageE プロジェクト, 奈良先端科学技術大学院大学, 大阪大学 : ソフトウェアタグ規格 V1.0, 2008年10月14日 : http://www.stage-project.jp/seika_dl.php
 [神谷 IT2008] 神谷芳樹, 樋口登 : オビニオン, ITプロジェクトの「見える化」, 電子情報通信学会誌, Vol.91, No. 6, 2008, pp.510-514, 2008
 [神谷2008] 神谷芳樹, 吉川宏幸, 樋口登 : 進行中のソフトウェア開発プロジェクト計測ツールの提供と普及活動, プロジェクトマネジメント学会誌, Vol. 10, No. 3, 2008, pp.11-16, 2008
 [鳥居2008] 鳥居宏次監修, EASE プロジェクト : ソフトウェア開発におけるエンピリカルアプローチ, アスキー, 2008-2
 [田村2008] 田村一賢他 : 情報サービス産業協会ワーキンググループによるEPM検証プロジェクトの推進, SEC journal 15, pp.48-53, 2008

ソフトウェア定量的品質コントロールの提案

『組み込みソフトウェア開発向け 品質作り込みガイド ESQR』の紹介

SEC研究員

吉澤 智美

SECでは組み込みソフトウェアのより一層の品質向上を目的として昨秋、「組み込みソフトウェア開発向け品質作り込みガイド」を刊行した。このガイドでは定量的品質コントロールを行うための概念、手法、品質指標、参考値や、システム障害の程度を評価するためのスケール（ST-SEISMIC¹ Scale）等の新しい概念も提案している。また、定性的な観点から開発プロジェクトを円滑にするためのヒント集も含んでいる。本稿では品質作り込みガイドのあらましを紹介すると共に、利用する上での注意点等を論じる。

1. ESQR：品質作り込みガイドの目的

近年、組み込みソフトウェアは需要の急拡大に伴い、品質や信頼性・安全性等が重視されるようになってきている。ところが実際のソフトウェア開発現場では、明確な品質目標を定めることが難しく、品質目標に基づく品質の定量的なコントロール（品質制御）という考え方が十分に実践されているとは言えない状況が続いている。その結果として、更なるソフトウェア不具合やシステム障害が発生し続けている。

SECでは組み込みエンジニアリング領域の活動として、ソフトウェア開発における品質制御の重要性を捉え、定量的品質コントロールをシステムティックに導入するための手引きとして「品質作り込みガイド」を編纂した。具体的には「ソフトウェアの成果物・作業の品質を指標を用いて可視化し、その程度に応じた適切な開発手法の利用に道筋をつける」ことを大きな目的としている。

2. ESQR：品質作り込みガイドの5つの特徴と品質定量コントロールループ

ESQRはEmbedded System development Quality Referenceの略で、組み込みシステムを開発していく過程で品質コン

トロールを行っていくためのリファレンス、ガイドであり、次に示すような5つの特徴を備えている。

特徴 ST-SEISMIC Scale システム障害震度

機能安全で言うところの事後安全計画の考え方に則り、実際のフィールドで発生した不具合の影響度を分析し、表すスケール。システムプロファイルの参考となる

特徴 システムプロファイル

ユーザの視点からシステムを分析し、システムとして求められる品質水準を明確にしていく評価方式

特徴 プロジェクトプロファイル

作り手の視点から開発プロジェクトの特徴を分析し、品質水準を考える際に参考とする評価方式

特徴 品質指標、参考値の提示

プロジェクトの作業や成果物を計測評価するための指標群とその参考値を例示

特徴 ヒント集

開発作業を円滑に進める上でのポイントを示し、それらを定性的に評価するためのチェック項目を例示

こうした、ESQRを用いての定量的品質コントロールを行う様子を表したものが図1である。

ST-SEISMIC、システムプロファイリング、プロジェクトプロファイリングを適用してシステム及びプロジェクトの分析を行い、何をどのように造るかを可視化する。

¹ ST-SEISMIC：System Trouble SEISMIC；Originally Invented by M.Hirayama and S.Yoshizawa, S.Yamaguchi

その分析結果を用いて、品質目標の選択と目標値の設定を行う。品質目標値を使用して品質コントロールを行う。その結果を用いて次製品の分析をより詳細に行う、というようにフィードバックをかけながらコントロールループを回していく。

以下に各活動について詳細を記す。

2.1 ST-SEISMIC Scale システム障害震度

市場でシステムに障害が起こった場合にその被害度の分析を行い、8階級のスケールで表す。このスケールを用いることでシステムに起こった障害を客観的かつ冷静に表し議論することが可能となる。具体的にはST-SEISMIC Scaleは表1のように、障害の程度をユーザの操作、健康被害、経済的被害あるいはシステム周辺への環境という4つの観点で評価分析し、8階級に分類している。ST-SEISMICというのは地震の震度になぞらえており、0階級なら無震、7階級なら激震というように表す。

この結果を使って障害発生時の対応を決定出来ると共

に、2.2に述べるシステムのタイプの分類を見直し、次回開発時のシステムプロファイルにフィードバックしていく。

2.2 システムプロファイリング

システム及びシステムに組込むソフトウェアに市場が求める品質について分析を行い、4つのタイプ(レベル)に分類、可視化する。このプロファイルの結果を基に、定量的コントロールを行う際に用いる品質指標の目標値を決定する。ユーザがシステムに求める品質はシステムによって、あるいはシステムを構成する各部分によって異なる。

システムプロファイリング(SCP²)の手順

システムプロファイリングの手順を図2に示す。システムプロファイリングではシステム利用・運用時に発生する可能性のあるシステム障害を想定し、人的側面及び、経済側面の2つの観点によって対象システムを4つのシステムタイプに分類する。

まず、人的側面ではそのシステムが障害を起こすこと

表1 ST-SEISMIC Scale

階級	ユーザ操作への影響	ユーザに対する健康被害の状況	ユーザに対する経済的被害の状況	システム周辺の環境への影響
0	無	不具合に気付かない		
1	微	一部のユーザがシステムの動作に違和感を感じる		
2	軽	ユーザの多くが不具合に気付く		
3	弱	ユーザのほとんどが不具合に気付く	一部のユーザが軽微な健康被害の可能性あり	一部のサービスが停止し、ユーザの一部で軽微な経済損が出る可能性がある
4	中	ユーザの目的が一部達成出来なくなる	一部のユーザに健康被害が出る	一部のサービスが停止し、ユーザの一部で経済損が出る可能性がある
5	強	ユーザの目的が一部達成出来なくなる	一部のユーザに重大な健康被害が出る	一部または全部のサービスが停止し、ユーザの一部で多大な経済損が出る可能性がある
6	烈		多くのユーザに重大な健康被害が出る	一部または全部のサービスが停止し、多くのユーザで多大な経済損が出る可能性がある
7	激		ほとんどのユーザに重大な健康被害が出る	一部または全部のサービスが停止し、ほとんどのユーザで多大な経済損が出る可能性がある

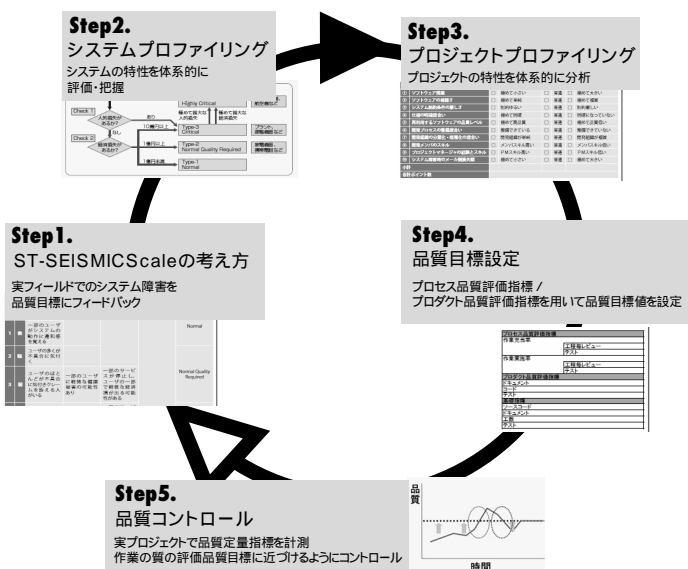


図1 ESQRによる品質定量コントロールループ

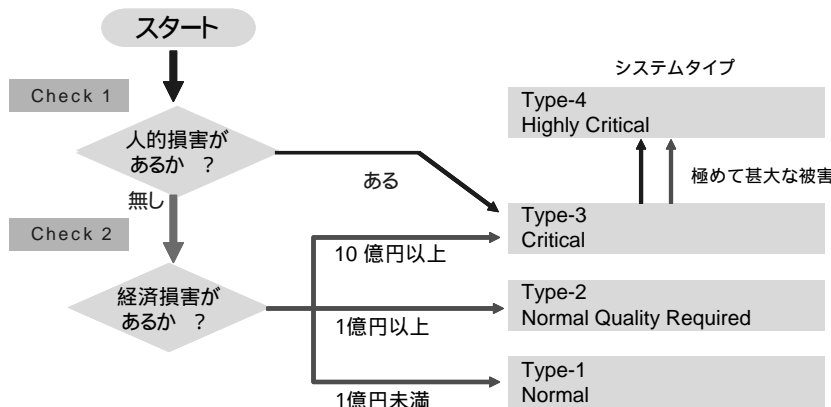


図2 システムプロファイリングの手順

2 SCP : System Characteristic Profiling ; Originally Invented by M.Hirayama and S.Yoshizawa

により、人が亡くなったり大怪我したりすることがあるか、ということを検討する。被害があるシステムであればType-3以上、さらに、人的・経済的に極めて甚大な被害が発生すると想定した場合にはType-4と判断する。

次に人的損害が無い場合は経済側面を考える。経済側面というのは、そのシステムあるいはシステムを制御するソフトウェアに障害が発生した場合にユーザがどの程度の経済損失を被るか、ということである。被害額により、1億円未満 (Type-1) 1億円以上 (Type-2) 10億円以上 (Type-3) という3段階に分類し、更に極めて甚大な被害が発生すると想定した場合にはType-4とする。ここで注意したいのは、経済側面は使い手の立場で考える、ということである。作り手の都合、例えばリコールのための費用はここには含まない。作り手の都合は次のプロジェクトプロファイルに含むので注意されたい。

2.3 プロジェクトプロファイリング

製品を開発するに当たってのプロダクト及びプロジェクトの性質を分析し可視化する。この結果を元に品質目標値に補正をかける。

開発対象となるシステムの性質がどのようなものであるかはプロジェクトを進めていく上で非常に重要なファクターとなる。また、システムを開発するプロジェクトメンバの構成がどのようなものであるかということも大きなファクターとなる。求める品質を実現させるためにシステムやプロジェクトの状況を分析し、開発時にどの程度の品質作り込みが行わなければならないかということのを可視化するのがプロジェクトプロファイリング (PCP³) である。

PCPファクターを使った補正係数の求め方

プロジェクトプロファイリングを用いた補正係数は

表2 プロジェクトプロファイリング算出表

	マイナス補正(-1)	基本	プラス補正(+1)
ソフトウェア規模	極めて小さい	普通	極めて大きい
ソフトウェアの複雑さ	極めて単純	普通	極めて複雑
システム制約条件の厳しさ	制約ゆるい	普通	制約厳しい
仕様の明確化度合い	極めて明確	普通	明確になっていない
再利用するソフトウェアの品質レベル	極めて高品質	普通	極めて品質低い
開発プロセスの整備度合い	整備出来ている	普通	整備出来ていない
開発組織の分業化・階層化の度合い	開発組織が単純	普通	開発組織が複雑
開発メンバのスキル	メンバスキルが高い	普通	メンバスキルが低い
プロジェクトマネージャの経験とスキル	PMスキルが高い	普通	PMスキルが低い
システム障害時のメーカ側損失額	極めて小さい	普通	極めて大きい
小計			
合計ポイント数 (Turning Point)			

表2を使用して算出する。

この表にはプロジェクトの進め方等に影響する10個のファクターが提示されている。これらのファクターはプロジェクトを進める上で加速あるいは制約の条件となり得るものである。例えば開発メンバ全員のスキルが高い場合、作り込み時の品質が相対的に高くなると考えられるので、レビューやテスト等の不具合検出に要する時間を低く抑えることが出来ると考えられる。

逆に、例えば仕様が極めてあいまいな場合等はプロジェクトを進めていく上でレビューを頻繁に行い、かける工数を増加する等の必要がある。このような観点でチェックを行っていき、補正係数を求める。

2.4 品質指標、参考値

定量的品質コントロールを行っていくためには、適切な品質指標を用いて開発作業や開発物を測り、評価していくことが必要となる。ESQRでは品質指標としてプロセス、プロダクトの2つの観点について品質評価指標とその参考値及び、それらを計測するための基礎指標を提示している (図3)。

(1) プロセス品質評価指標

品質を確認する作業にはレビューとテストがある。

プロセス品質評価指標ではこれら2つの作業が要求定義、設計、実装、テストの各プロセス⁴で十分に行われているかどうかということの評価する。さらに評価の観点として、開発全体作業に対して適切な工数が費やされているか (作業充当率) と、開発規模に対して十分な工数が費やされているか (作業実施率) の2つがある。

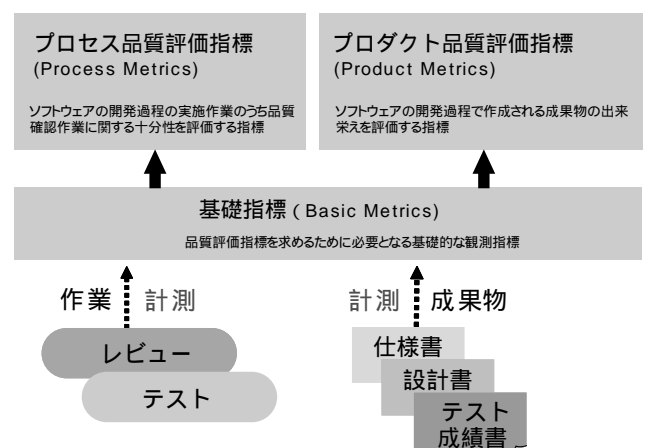


図3 品質指標

3 PCP : Project Characteristic Profiling ; Originally Invented by M.Hirayama and S.Yoshizawa

4 各プロセスで行う作業については『組込みソフトウェア向け開発プロセスガイド : ESPR』(IPA / SEC編) で詳細を記述しているので参考されたい。

(2) プロダクト品質評価指標

開発を行っていく上で、最終成果物であるソフトウェアそのものと、開発過程ではドキュメント等の中間成果物が生成される。プロダクト品質評価指標はこれら開発時の成果物であるドキュメント、ソースコード、オブジェクト（ソフトウェアそのもの）の3つについて評価を行う。

ドキュメント品質評価指標では品質を確保するだけのボリュームがあるか、記述内容のバランスを見ることで質を評価する。ソースコード品質評価指標ではコードのそれぞれ（ファイル、関数）が品質を確保するに足るボリュームに収まっているか、コードの内容が品質を確保するに足る構造になっているかということの評価をする。テスト品質評価指標ではオブジェクトを十分にテスト出来るようになってきているか、検出した不具合がきちんと修正されているか等々を評価する。

(3) 基礎指標

基礎指標は品質評価指標を算出するために計測すべき、ソースコード行数や作業工数等の具体的な計測指標そのものである。これらの基礎指標については、

- ・誰でも手軽に測れる（計測のために高価なツールを必要としない）
- ・計測者によって計測結果がぶれない

ということを念頭にその定義や測り方を規定している。

(4) 品質目標値の算出方法

ESQRでは以上述べたプロセス品質評価指標及びプロダクト品質評価指標のそれぞれに対し、システムレベルごとの参考値を提供している。

この参考値をベースに、下の式を用いることでプロジェクトごとに各品質評価指標の目標値を決定することが出来る。

$$\text{品質目標値} = (\text{システムレベルごとの参考値}) + \{ (\text{補正係数} / 10) \times \text{補正ベース値} \}$$

2.5 品質評価指標を用いた開発作業の制御

例えば、設計レビュー作業充当率で、6.0という目標値を決めたとする。これは設計作業全体の工数のうち、6%をレビューに充てるという意味である。設計レビューを行った際にどのくらいの工数をかけたかを計測し、

目標値との差分を求める。大きな差が無ければ品質を確保するためのレビュー工数が確保されたと判断し、作業を続行する。計測した値が目標値より極端に小さい場合には品質を確保するためのレビュー工数が足りないと判断し、再度レビューを行う、あるいは後工程でフォローアップする等の手段を考える。また、計測した値が大きい場合はレビューに工数がかかりすぎているということも考えられるため、レビュー作業の進め方に問題が無いか、ベースとなるレビュー対象の設計書の品質に問題が無いか見直す等の手立てを行う。このように品質指標の目標値と計測値とを対比することで開発作業の工程ごとに品質を確保出来ているかどうかの見直しをすることが出来る。

3. 品質向上に向けた開発作業の質的向上

定量的品質コントロールを行っていく背景には健全なプロジェクト活動を行っていく必要がある。ソフトウェア開発は人間的な活動を多く含み、その中には定量的に測れないものもある。

ESQRではソフトウェア開発に関する活動のうち、コミュニケーション、ドキュメント、レビュー、テスト及び定量的品質コントロールの5つの活動について、定性的な側面からの評価と改善を行うためのチェック項目を解説すると共に提示している。例として、コミュニケーションに関するチェック項目を表3に示す。

例えば、「2.コミュニケーションの相手を尊重しているか」という項目ではまず、メンバ間で尊重し合うことについて解説し、更に注意すべき点として、(1)相手の立

表3 コミュニケーションチェック項目

NO.	チェック項目	チェック
1	円滑なコミュニケーションを行う土壌があるか	
2	コミュニケーションの相手を尊重しているか	
3	会議の事前準備、環境が整っているか	
4	必要なテーマに対して、必要なだけの時間を割いて情報展開、議論したか	
5	過去の経験や有識者の経験を尊重しているか	
6	議論、話している内容の本質的なところが共有されているか	
7	意思決定の道筋は合意が得られているか	
8	必要な情報が必要なメンバに正しく伝わっているか	
9	メールなどの非同期コミュニケーションに依存しすぎているか	
10	生産性の向上につながるようなコミュニケーションがとられているか	
11	会議やコミュニケーションの結果が開発に反映されているか	
12	会議が終わった後に無常感が漂っていないか	

ち位置（背景）を理解してコミュニケーションする、(2)メンバ全員が理解出来る言葉を心掛ける、を示している。これらのチェック項目、解説は当たり前前のことが多い。ただ、当たり前前を当たり前にする、ということが品質を作り込むのに何より重要である。そのようなことも読み取っていただければ幸いである。

4. ESQRを用いる際の注意点

4.1 品質指標をすべて利用する必要は無い

ESQRでは品質評価指標、基礎指標合わせて約50種類の指標を提示している。実際の利用に際しては、プロジェクトの成熟度及び経験により適切な指標を選択していく。

(1) プロジェクトで計測の経験が無い場合

組織やプロジェクトで品質計測の経験が少ない場合には、まずスタートポイントとして、自分たちが計測出来るような、あるいは自分たちに一番役に立ちそうな指標をいくつか選んでプロジェクトで適用するの1つの方法である。ここで重要なのは、ただ測って値を出すのではなく、その値を開発メンバと共有し、開発の進め方にフィードバックをかけるという行為を行うことで定量的品質コントロールの意義等を十分に理解していくことである。

(2) 既に計測の経験があるプロジェクトの場合

逆に、既に品質計測の経験がある組織やプロジェクトではESQRで示す品質指標の説明を元に、今までの指標を見直してみるのも効果的である。SECではESQRの目標値を算出するに当たり、多くのプロジェクトにヒアリングを行ってきたが、データの計測は行っているが分析はうまく行えておらず、せつかくのデータを開発にフィードバック出来ないというケースが散見された。データを計測するには労力が伴うものであるから、改善のサイクルを回すためにどのようにしたら良いかということ念頭に置いて品質定量化の活動を見直していくのも有効である。

4.2 参考値をそのまま利用しない

ESQRでは、すべての品質評価指標に対し、システムタイプごとの参考値を示している。これらはあくまで参

考値である。このため、2.3節で示したプロジェクトプロファイリングをしっかりと行い、まずは自分たちのシステムとプロジェクトの特性を見極め、関係者全員が納得のいく目標値を定める必要がある。また、何度かプロジェクトを経験する中で適切な目標値を調整していくことも重要である。

4.3 数値に振り回されない

ESQRの本質は適切な品質指標を選択し、適切な目標値を求め、開発時にその目標値を目安として定量的コントロールにより品質作り込みを実施していくところにある。実際の開発の中で計測する品質指標の値を上下させることは簡単であり、例えば、作業工数の水増しを行えば数字上は目標を達成出来てしまう。このような場合、数値の目標は達成出来ても、求める品質が確保出来ないということは明白である。開発側に一方的に値を押し付けるのではなく、関係者間で合意し、自分たちのプロジェクトの身の丈に合った品質コントロールを行うことが必要である。また一方で、定量的な側面だけでなく、定性的な面からもソフトウェアやプロジェクトの健全さを補完しながら考えていくことも重要である。

5. おわりに

以上、ESQRのあらましを紹介した。ESQRを用いることによって開発対象の可視化（システムプロファイル）及び開発側条件の可視化（プロジェクトプロファイル）を行うことが出来る。これにより、定量的品質コントロールを行うに当たり、適切な品質指標を選択し、開発するシステムに見合った目標値を定めることが出来る。

ESQRではこれら基本的な概念を示すと共に、具体的な目標を決めるためのベースとなる参考値を提示している。従来、ソフトウェアの品質コントロールは技法や考え方は示されていたものの、何をどのくらいやったら良いかということについては行間を読み取らねばならないことが多く、組織間で具体的な数値を用いて議論出来ることは少なかった。ESQRで示した品質指標、参考値等はまだまだ精査すべきものと考えているが、今後はソフトウェア業界全体でこれら数値を用いて、現実的な議論を展開し、ソフトウェア業界全体の品質の向上に寄与出来れば幸いと考えている。

ソフトウェアに関する論文の書き方

主に開発現場に携わる人のために

大阪大学大学院情報科学研究科 教授
井上 克郎

ソフトウェア開発の現場に携わっている技術者がソフトウェア工学やソフトウェアシステムに関する論文を書くための要点をまとめる。論文を書いて、それが採録され、論文誌に掲載されるのは簡単ではないが、大きな価値がある。本稿を参考に、多くの論文が書かれ、発表されることを期待する。

1 はじめに

ソフトウェア工学やソフトウェアシステムに関する論文は、非常に書きにくい、という声をよく耳にする[権藤2008]。これは他分野の論文に比べて相対的に、という意味であろうが、定量的な比較は困難である。ただ、筆者の主観としても、工学、理学の他の分野に比べて多少ハンディがあるように感じる。

これは、対象物のソフトウェアの可視化が困難である、絶対的な評価の基準がない等、種々の理由によるものと思われるが、だからと言って論文を書かなければどうなるであろうか。せっかく苦労して開発した技術や方法論を広く伝えることが出来ず、この分野の技術進歩が停滞してしまう。

本稿では、ソフトウェア開発に関わっている技術者を対象として、ソフトウェア関連の論文を書くためのいくつかのポイントを解説する。

自分自身、論文が自由に書けるとは思えず、まだまだ苦労をしているが、より多くのソフトウェア関係の論文が世の中に出て、この分野の技術進歩の一助になることを願って、思いつくままに記す。

2 論文を書く目的

科学技術分野に限らず、いろいろな分野で数多くの論文が執筆され、公表されている。

ソフトウェア分野では、このSEC journalを始め、情報処理学会論文誌、電子情報通信学会論文誌D分冊、コンピュータソフトウェア(ソフトウェア科学会)等の和文論文誌がある(英文でも受付あり)。また、IEEE Transactions on Software Engineering (TSE)、ACM Transactions on Software Engineering and Methodology (TOSEM)、Automated Software Engineering等を代表として、数多くのソフトウェア工学全般やソフトウェアシステム、プロセス、メトリクス等の特定のテーマを扱う英文論文誌がある。また、学会が主催する各種会議や研究会での発表もある(口頭のプレゼンテーションと議事録論文)。自分の研究成果をこれらの論文として掲載することによって以下のような効果があると考えられる。

- ・会社や組織の技術開発力の優位性をPR出来る。
- ・研究開発の1つの業績やマイルストーンとして認識出来る。

・論文として成果をまとめることにより、研究成果を整理し、その位置付けをはっきりさせることができる。

一方、論文を書くためには、仕事を振り返って整理し、時間をかけて文章化して、投稿をし、査読者のコメントに従って何度か修正をし、(論文誌によっては掲載料まで払って) やっと掲載される。査読者のコメントは強烈なことも多く、不採録(リジェクト)の場合もある。精神的にタフでなければやっていけない。

このような苦労が予想される論文について、執筆する価値があるかどうかは十分検討する必要がある。大学や研究機関では、論文が評価対象になっており、Publish or perish(論文出さずか死か)の世界なので、書かざるを得ない。

逆に企業の開発現場では、論文を書くためのコストが予算として計上されていない場合が普通であるが、上記のようなメリットがあるので、トップの意思として積極的に取り組んでもらいたい。日本の開発現場では、先進的な取り組みを行ってきているが、論文がうまく書かれずに普遍化出来ない例が多々ある。単なる社内の一事例として世の中に埋もれては非常に残念である。

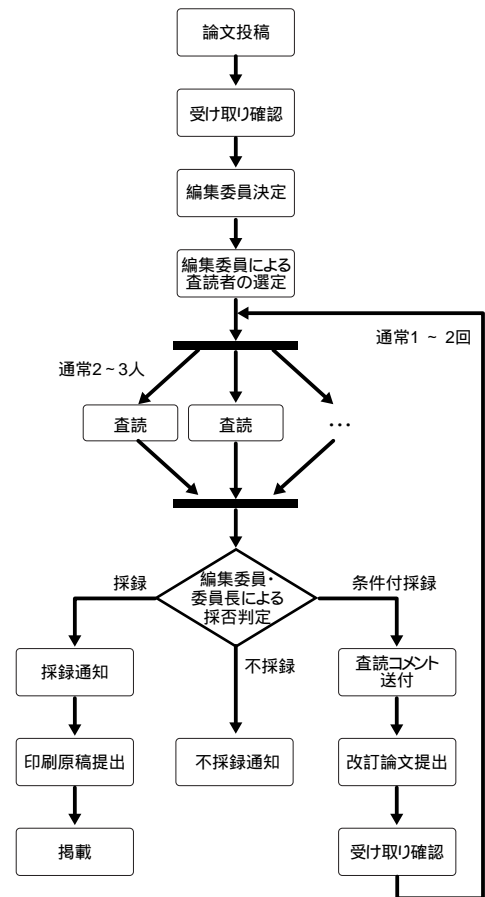


図1 典型的な論文の査読プロセス

3 論文掲載までのプロセス

図1に典型的な論文の投稿から論文誌や雑誌に掲載されるまでのプロセスを示す。

まず、作成した論文を掲載したい雑誌、論文誌の投稿先に送付する。そこで、その論文を担当する編集委員(メタ査読者)が決められ、その編集委員と著者がやり取りをすることになる。

編集委員は、その論文にふさわしい査読者を選び、査読をしてもらう。査読者の数は、雑誌や論文の内容によって決まっており、通常2~3人である。また、1~3カ月程度が査読期間として割り当てられる。

返ってきた査読結果を見て、編集委員及び編集委員長は、「そのまま採録する(無条件採録)」、「不採録(返戻、リジェクト)」、「条件付採録」のいずれかの決定を下し、著者に連絡する。

条件付採録は、査読者のコメントに従って修正すれば採録の可能性があることを示すもので、通常、2~3カ月以内に改訂版を作って送付し、再度査読者によるチェックを受け、判定される。

一般に査読者のコメントは非常に厳しく感じられ、なかなかそれに従って論文の修正をするのは厄介であるが、

専門家による指摘は重要であり、よく検討して、修正を行う(すべて査読者の言いなりというわけではないが)。

また、コメントに従ってどこをどのように修正したかを記した文章を付けるのが通常で、そこでは、ボランティアで査読を行ってくださっている査読者に謝意を示した文言を入れるとよい。

4 論文の主テーマについて

会社の現場で工夫し、努力した開発案件を論文にしようとしたときに、まず考えなければならないのは、どのようなテーマで論文を書くかである。以下は、筆者が考えるポイントである。

自分にとっても読者にとっても面白いテーマを選ぶ

よく“論文の書き方”等で、「新規性」、「有効性」、「信頼性」等とこの価値をブレークダウンして表現されているが、非常に抽象的で分かりづらい[電子情報通信学会]。結局のところ、いかにして査読者や読者に面白い、読んで良かった、と思わせることができるかが重要である。そのためには、まず、書く本人が面白いと思えなければ

伝わるはずがない。開発現場で新たにこういう発見をして驚いた、開発方法を工夫したらびっくりするほど性能が改善した等、まず、自分が興奮し、それが他人にも共感を得るようなテーマでなければならない。

書くべき要素を絞る

ソフトウェア関係の論文として、対象のプロジェクトのライフサイクル全体を満遍なく書くことは、フォーカスがボケて良くない。査読者や読者にとって、例えばプロジェクトの中で使われた新しいアルゴリズムが重要なのか、定量的な評価が興味を引くのか、よく考えて、論文に書くものを厳選する。

一般読者の知識を考える

投稿する論文誌をよく読み、対象とする読者の知識を推定してテーマ選定を行う。SEC journalの読者は、概ねソフトウェア技術者、とくにソフトウェアの信頼性や品質に非常に興味を持っている人々である。余りにも読者の知識とかけ離れた論文は、読者は理解出来ず、受け入れられない。一方、多くの読者が既に知っていたり簡単に想像出来たりするものでは、論文としての価値が無い（これは一般論で、実証の一方として繰返実験等は価値がある）

役に立つかどうかの視点を忘れずに

論文は、他人に読まれてその知識が広く伝わり、いろいろな形で利用されてこそ価値がある。掲載されることが目的になってしまっはいけない（学位取得のため等、大学的な事情はよくある話ではあるが...）。論文のテーマや要素を選ぶ際には、どのように役立つかを常に意識する必要がある。同業他社に知られてはまずい、という考えでは論文は書けない。オープンソースと同様、自分の知見をぜひ使って欲しい、という精神が重要である。とはいえ、所属企業の方針の確認も必要であるが、ソフトウェア関係の手法や技術は、論文に書く程度の情報だけでは簡単には真似出来ないの、どんどん公開すべきである。

5 論文の構成

ソフトウェア工学分野でよく見る論文のパターンは図2のような構造である。本稿では、このパターンに沿い、それぞれについて説明する。

5.1 タイトルについて

タイトルは非常に重要である。それを見て、何をテーマとして主張したい論文か、簡単に分かるようにすべき

タイトル

著者、所属

概要

まえがき

本文1

本文2

...

[本文列は例えば以下のような構成]

モデル

実装

評価

議論

関連研究

まとめ

謝辞

参考文献

(付録)

図2 典型的なソフトウェア関係の論文の構成

である。また、論文の内容に比べて広すぎず、狭すぎず適切な範囲をカバーしているかチェックする必要がある。

タイトルは、過去の論文や関連する報告等のタイトルと同一であってはいけない。内容がほぼ同じであったとしても混同されないために、違いがある必要がある。

タイトルは、1つの日本語の文として、分かりやすくバランスの良いものであって欲しい。また、一文だけでは不十分な場合は、必要に応じて副題を付けて、主題を補足することもよく行う手法である。

タイトルを決めるのは簡単ではない。論文の主たるテーマを何にするか、どういう構成でストーリー展開するかが明確になっていないと、良いタイトルは付けられない。1つのタイトルを巡って、共著者が何時間にもわたって議論する、ということはよくあることで、この議論を通じてテーマやストーリーが整理されていく。

5.2 概要について

概要は、あらましや内容梗概、アブストラクトとも呼ばれ、その論文誌によって大体量が指定されている。ポイントとしては、ここを読めば、背景、手法、結果、評価等の仕事の全体が分かると同時に、何を売りにした論文なのかが理解出来るようなものにする。そして、続く本文を読みたくなるような具体的なセールスポイント

(例えば「効率が30%向上した」等)をしっかりと書く。

まえがきの一部を抜き出した概要がよくあるが、まえがきとの役割が違うので、新たな文章で書くのが美しい(読んでいると同じ文章の使いまわしは簡単に分かり、がっかりする)。

5.3 まえがきの書き方

まえがきは、その後続く本文の導入である。従って、論文を書く背景や動機、目的や必要性、同様な研究技術の現状等を分かりやすく書く。新たな開発法についての論文であれば、なぜ今までの開発法では駄目か、どのような問題がクリアされなければならないか等である。ただし長い背景説明は読む気を削ぐので、出来るだけ簡潔にまとめる。重要な先行研究がある場合は、参照しておく。

一方まえがきは、それ自体で完結した短い読み物である必要がある。通常、査読者や読者が本文を丁寧に読むことは考えにくい。概要とまえがきだけで、評価されてしまう。従って、この論文の価値を分かってもらえるように、提案する手法、やったこと、評価結果、主張等も簡潔に書いておく。概要と重複する部分もあるが、一般にまえがきのほうが分量は多いので、より丁寧に書くことが出来る。また、以降の本文に書くべき内容であっても、セールスポイントに関しては、重複を恐れずに繰り返し書こう。

最後に、各章の簡単な説明を書く場合が多いが、必ず書かなければならないものではない。

5.4 本文について

一般に本文は複数の章から構成される。それにはいろいろなパターンがあるが、ここでは、ソフトウェア関係の論文によく見られるパターン(モデル、実装、評価、議論、関連研究)に沿って説明する。

モデル

ソフトウェア開発に関するプロダクトやプロセスを分かりやすく説明するためには、対象を抽象化して図や式、(擬似)プログラム等で表現した「モデル」を用いることが多い。対象をモデルで表現することをモデル化という。例えば、「開発プロセスを状態遷移図でモデル化する」等が一例である。

論文の主張をうまく表現するために、適当なモデルを設計することは重要である。状態遷移図やフローチャート等はプロセス関係によく用いられるモデルである。既にあるモデルをそのまま利用するか、拡張して用いることもよく行われる。例えばWBSを拡張してプロジェクトを表現する等である。

モデルを用いないこともあるが、言葉での表現を用いるだけでは、問題を厳密に定義することが出来なかったり、誤解を生じさせたりすることがあるので気を付ける必要がある。

実装

モデルで表現したことを実際にシステム化した場合、その実装について書くことが多い。論文のテーマと直接関係ない実装の詳細を書く必要はないが、少なくともどの程度のシステムを作成したかが分かるようなデータを入れると、提案しているテーマの現実味が増し、査読者や読者を説得するのに都合が良い。

もし、システムの機能や性能が主たるテーマの場合は、より詳しく実装について書く必要がある。しかし、システムのすべてについて同じレベルで記述する必要はない。システム中の構成要素ごとに同じような説明を行っている論文があるが、それは良くない。関心事である機能や性能に直接関連する部分をより詳しく書き、関連の無い部分は大幅に省略すべきである。

ソフトウェア開発に関する提案であれば、その提案手法に沿った開発の実行がこれに当たる。

評価

実験やケーススタディ等いろいろな評価方法があるが、ソフトウェア関係の論文としては、評価が無ければあまり価値が無い。論文で著者が提案しようとしていることが本当に有効で有益なものかを、読者や査読者が判断出来るように、出来るだけ、定量的なデータを示す必要がある。これを行うために、実装したシステムの実行データや、提案した手法の種々の実験データの収集が重要である。また、通常、ソフトウェア関連のデータは複雑な要因が絡んで結果を簡単に論じることが出来ない場合が多く、データの処理や評価のためには統計的処理が必要である[豊田1994]。これに関しては、「7.3 ソフトウェア技術の実証と評価」でも詳しく述べる。

議論

得られた結果や成果に関して、主に定性的な議論を記述する。何が分かったか、何が良くなったか、どういう効果があったか等のポジティブな面だけではなく、どういう限界があるか、まだ解決されていない問題は何か等のネガティブな面も書くべきである。そうすることによって、公正な立場で書かれていることが示せ、論文全体の説得力が向上する。

ここで書くことは、選定したテーマに沿っており、読者にとって「なるほど」と思わせる必要がある。反発を招くような独善的な解釈や主張の展開は、他の部分の

論旨の展開も駄目にする。慎重な論理展開が必要である。

関連研究

先行する研究との違いやアドバンテージについて議論する。まえがきのすぐ後の本文の先頭を書くこともある。また、上記「議論」の議論の一部として書かれることも多い。

先行研究すべてを網羅することは容易ではないが、重要な論文は必ず引用し、議論しておくことは、著者らの研究が独善的ではないことを示す意味で大切である。通常、先行研究をいくつかのグループに分け、それらごとに、違い、特徴等を書いていく。ただし、違いを強調するために、不当に先行研究をけなすことは良くない。

5.5 あとがきについて

「あとがき」、「おわりに」、「まとめ」等、いろいろな書き方がある。論文の主張を強調するために、成果や結果を再び書く。それほど時間が無い読者は、まえがきの後、本文を飛ばして、ここだけを読む場合も多い。また、本文で書ききれなかった少し主題とは外れた議論や主張を書く場合もある。

更に、今後の研究や開発の進め方の方向性を示して終わるのが一般的である。

5.6 謝辞について

論文の執筆者として名前を載せなかった関係者に感謝の意を表する記述を行う。また、研究や開発を支援してもらったスポンサーを明記する必要がある場合は、ここに書く。一般に、謝辞に名前を載せることで生じる問題は、載せないで生じる問題よりは少ないので、出来るだけ広く載せる。

5.7 参考文献の書き方

論文中で引用する参考文献は、正確にその著者や著書・論文名、出典、年等を書く。書き方は、その論文誌の決まりがある場合はそれに従う。書き方が統一されていないか、一部の情報が欠落していたりする参考文献のリストは非常に見苦しいので、細心の注意を払う必要がある。昔は、完全なリストを作るのは大変な作業だったが、今はインターネットで容易に情報収集出来、作ることが出来るようになったので、丁寧に取り組みさえすればよい。

査読者は、その論文と他の論文との関連を知るために、参考文献リストをよく見る。重要な参考文献が抜けている場合は、その論文のレベルを疑う。また、査読者は同じような開発、研究を行っている場合が多いが、査読者が行った先行論文が参考文献リストに載っていない場合も、悪印象を与える。

そういう観点で、同様な先行研究の調査は重要で、ある程度網羅的に収集を行い、取捨選択して参考文献のリストを作る。

5.8 付録について

付録はオプションであり、多くの論文では付録は無い。本文で書くと、読者にとって論旨の理解を妨げるような詳細なデータや手法、証明等がある場合は、付録を設け、そこに書くとよい。一部の興味ある読者や査読者が読む。ただ、あまりにも長大な付録は、ページ数が増える要因にもなるため、避けるべきである。その場合、一部の読者にとって必要と思われる情報は、別途、技術レポート等の形で公開し、必要に応じて、付録への参照を誘導するような文言を書く。

6 分かりやすい文章について

論文の文章は、読んで理解出来なければ、査読者は×をつけるし、読者は途中で読むのを止める。出来るかぎり、平易で理解しやすい文章を書くべきである。以下いくつかのポイントを説明する。

用語の定義

著者にとっては当たり前の用語でも、読者にとってはなじみの無いものであったり、理解が異なっていたりする場合もある。当たり前と思っても、予備知識の無い読者に向けて書くつもりで、丁寧な解説が必要である。出来るだけその用語が出現する最初のところで、1~2行の簡単な説明をつけるべきである。新たな用語を使う際には、その説明は必要かどうか、謙虚になって検討すべきである。

自己完結性

論文は、ある程度の背景知識を持った同業者が読めば、内容が理解出来るように、必要な説明を全部含んでおく必要がある。企業特有の知識やツール等を前提にした説明では、他者は理解出来ない。また、先行研究の論文があってそれに基づいて論文を書く場合も、その概要がある程度分かるように、説明を書くべきである。その論文単独で、テーマが理解出来るように工夫する必要がある。

分かりやすい日本語文

日本語として分かりやすい文になるよう努力する。主語、述語がきちんとあるか、論理的な構造になっているか、長すぎないか、用語の意味や使い方にブレは無いのか、等いろいろなチェックポイントがある[谷口2008]。

書くべき事柄が論理的に整理されていないと、良い日

本語にはならない。長い文章は、論理が途中で分からなくなってしまうので、短い文章に分解するとよい。

構造的に書く

長い段落の文章を、最初から最後まで読みきるのは大変である。出来るだけ論旨を整理し、文章を構造化する。箇条書きを利用して、議論のポイントが視覚的に理解出来るように工夫する。

例の利用

難しいアルゴリズムや手法等の説明には苦勞する。擬似プログラムやフローチャートを用いても読者には容易には理解されない。そういう場合は、出来るだけ単純な例を用いて説明すると理解しやすい。ただし論文としては例だけではなく、アルゴリズムや手法の一般的な説明も必要である。

図や表の利用

手順や方式、システム構成等、ソフトウェア関連の論文では多くの図や表が用いられる。これは、文章だけの説明よりずっと理解しやすいので、積極的に活用しよう。

図や表には、本文を読まなくてもその意味が分かるように、丁寧なキャプション(説明文)を書く。また、線や点、スケールが分かるように、単位や凡例を記載する。複雑な図や表は、読者が解読するのが困難になるので、表現する対象を絞るか、または、表現したい部分を強調するとよい。

7 ソフトウェア技術者にとっての論文作成のポイント

企業の実務家を書く論文を読んで感じるのは、開発や研究に非常に大きな努力をしており、大学や研究所では出来ない規模や品質の対象を扱っている割に、研究の位置付けやデータ処理、評価、関連研究との違い等が弱く、論文として非常に損をしていることである。それらを克服するためのいくつかのポイントを示す。

7.1 社内向けの発表と論文の違い

社内発表では、背景知識として社内共通のものが前提となつて、それに基づいて論文や発表を組み立てていく。論文は社外の多数の一般技術者が読者である。まず、テーマとして内部受けするだけのものか、社外の一般読者に耐え得るものか、よく考える必要がある。社内では評価されないものでも、対外的には評価され得るものもあり、よく検討して積極的に論文としてまとめて欲しい。

そのためには、関連研究の文献を探し、世の中のレベルはどうなっているか、それに対して開発したものがど

の程度のものか冷静に分析する必要がある。先行事例に対して、すべて勝っている必要はない。あるポイントで優れているならば、そこだけをテーマに絞り論文を構成するのも良い考えである。

また、論文を書く上で、社内発表では簡単に伝わる内容も、一般読者には理解不能な用語や概念がある。新入社員に説明するようつもりで、慎重な用語や概念の使い方が必要である。一般の読者が読んで分からない、というのは読者の責任ではなく、著者の責任である、ということに肝に銘じて執筆をする必要がある。

7.2 データの表現について

提案する手法やツールの評価に関して、定量的な議論をすることが重要である。そのためには、社内で収集したデータを公表する必要がある。いろいろな制約があり、集めたデータをそのまま論文として公表することは出来ない場合がほとんどであろう。そのためには、生データを直接論文に書くのではなく、加工した形で書く。

例えば、ある手法Mを用いて、バグの検出を効率化する、という論文を書くとする。生データとしては、X月Y日にバグがZ個検出された、というようなものであろう。論文としては、どのように効率化されたか、ということが知りたいわけで、データの絶対値には興味はない。従って、作業時間全体のa%の時点で、全バグ中のb%のバグが検出された、で十分である。作業量も同様に、例えばテスト工程のc%の工数をこのMのために用いた、という表現でよい。

また、プロジェクト自身の定性的な性質を記述する場合でも、顧客が推測されないように心がける。例えば、顧客の業種を示す場合も、金融系、流通系等、抽象的表現を用いればよい。プロジェクトの規模を示すためには、人月やコード量、FP値等を明示的に示すことが出来ればよいが、それが無理な場合は、100人月程度、数MLOC規模等とそれらのおおまかな値を示すことで読者に規模の印象を与えることが出来る。

7.3 ソフトウェア技術の実証と評価

ソフトウェアに関する論文では、提案する手法(アルゴリズム、ツール、システム等)の評価が重要である。ソフトウェア工学の論文は他の工学や科学の分野に比べて、提案する手法の評価が甘い、という指摘もある[ZELKOWITZ1998]。それは、ソフトウェアに関する評価、とくに定量的な評価を行うことが困難である場合が多いからである。理由としては次のようなものがある。

・評価するための手間や時間が大きく、経済的に成り立ちにくい。簡単に計測出来る物理量等に比べて、ソフ

トウェアに関する種々の特性を計るのは容易ではない。とくにソフトウェア開発に関する技法等に関して、その効果を計測するためには、何度も同じような開発を違った環境で繰り返し行う必要があるが、そのような実験の実行は、経済的に非現実的である。

- ・絶対的な計測基準が定めにくい。物理量と異なり、人造物であるソフトウェアプロダクトやプロセスは、定量的な尺度を広く合意の下で決めることは困難である。行数(LOC)1つを取っても、空行やコメント行の扱いをどうするか等、議論は多い。
- ・一般に評価結果に影響を与える環境の要素が多く、提案するものが本当に有効なのか、たまたま外の要因でそうなったのか判別が難しい。そのような問題を排除するためには、人為的に制御された実験環境を整備し、提案手法を使う場合と使わない場合で比較すればよい(新薬の治験と同様な手法)。しかし、そのような制御された実験環境、とくに大規模ソフトウェア開発に関する環境を整備することは、物理的、経済的に困難である。

しかしながら、難しいからといって評価を行わず提案するだけでは、科学的な態度ではない。完璧に制御された環境での対比実験で評価を行うことは出来なくとも、いろいろ工夫の余地はある。

- ・対比実験が出来ない場合、単独の実証実験や適用事例の結果について細かく分析し、他の因子ではなく提案手法の効果によって、有効な結果が得られていることを定性的に議論する。
- ・小規模な対象を厳密に評価し、それが実用的な規模まで拡大出来る(スケールアップ)ことを議論する。
- ・環境は整っていないが、提案する手法を利用した事例と利用しない事例を集め、統計的処理を用いて、有意差を議論する。

これらを随時組み合わせて、効率的に評価を行い、読者に提案手法の有効性を訴える。

7.4 より良い論文を作るために

より良い論文を作成するためには、既出版され、良い論文という評価を受けているものを読み、そのパターンを真似るのが近道である。分かりやすく、説得力のある論文を見つけたら、その論理展開の仕方を分析し、自分のテーマに関して、同様な論理展開が出来るか検討してみよう。手法や評価方法も、良い論文の使える部分を利用してみよう。

ただし、文章のコピーや図のそのままの利用は(参考文献として明示した小規模な場合を除き)してはならな

い。

一度書いた論文は、自分で何度も読み返し、論旨が一貫しているか、読者がつまづくところはないか、用語の統一が出来ているか、日本語としておかしくないか等を慎重に調べる。

さらに、第三者に読んでもらい、コメントをもらうことも重要である。自分1人では気づかない種々の問題点の発見に役立つ。書いた論文の初めての読者が査読者というのは危険すぎる。査読者に誤字脱字を発見されるようでは、論文の採録は程遠い。

8

おわりに

企業の人にとって論文を書くことは容易ではない。論文を書く直接的な利益も見えにくい。しかし、「2 論文を書く目的」にも書いたように、論文を作成することは、個人的にも組織的にも多くのメリットがある。

今回は、国内の論文誌に日本語で論文を書くことを前提として本稿の論を進めてきたが、ぜひ、英文の論文作成も試みて欲しい。国際的な評価を得るためには、英文での国際会議や論文誌、雑誌への発表が非常に有効である。日本語の論文よりさらに手間を要するが、その価値は高い。

最後に、本稿によって、ソフトウェアに関する論文投稿、とくにソフトウェア開発に従事する現場の技術者からの投稿が増えることを期待する。

謝辞

本稿に対して、ご意見をいただいた大阪大学楠本真二教授に感謝する。また、執筆のご支援をいただいたSEC journalの神谷芳樹編集長及び矢野亜希編集委員に深謝する。

参考文献

- [ZELKOWITZ1998] M. Zelkowitz, and D. Wallace, Experimental Models for Validating Technology, IEEE Computer, Vol.31, No.5, pp.23-31, May 1998
- [権藤2008] 権藤克彦: なぜソフトウェア論文を書くのは難しいか?, 第15回ソフトウェア工学基礎ワークショップポスターセッション, 2008-11
- [電子情報通信学会] 電子情報通信学会: 電子情報通信学会和文論文誌投稿のしおり(情報・システムソサエティ), http://www.ieice.org/jpn/shiori/iss_4.html#4.1
- [豊田1994] 豊田秀樹: 違いを見ぬく統計学 実験計画と分散分析入門, プルーパーボックスB1013, 講談社, 1994
- [谷口2008] 谷口健一: 情報科学における論文の書き方, 大阪大学大学院情報科学研究科コンピュータサイエンスセミナー資料, 2008-6

生産性要因に基づいて層別されたソフトウェア開発工数見積りモデル



角田 雅照 †



門田 暁人 †



松本 健一 †



高橋 昭彦 ††††

重回帰分析により工数見積りモデルを構築する場合、何らかの基準に基づいてプロジェクトを層別（分類）したデータセットを作成し、分類ごとに見積りモデルを構築することが望ましい。本論文では、生産性と関連が強い（寄与率の大きな）プロジェクト特性（開発言語、アーキテクチャ等）を用いて層別を行うことを提案する。財団法人 経済調査会が収集した153プロジェクトより抽出した、43プロジェクトの実績データを分析し、規模当たり要員数（平均要員数 / FP）の寄与率が大きい（寄与率38%）ことが分かった。一般に見積りモデル利用時には（規模当たり）要員数は確定していないが、規模当たり要員数を概算値（大、中、小の3段階）で決めることは比較的容易である。規模当たり要員数を用いてプロジェクトを3つに層別した結果、見積り精度が大きく改善する（要員数概算値の相対誤差平均が50%の場合、工数見積り値の相対誤差中央値が56.6%から35.9%になる）ことが分かった。

Software Development Effort Estimation Models Stratified by Productivity Factors

Masateru Tsunoda †, Akito Monden †, Ken-ichi Matsumoto † and Akihiko Takahashi ††††

When building effort estimation models by multiple regression analysis, it is preferable to stratify software projects according to some criteria, and to build models for each stratified projects. In this paper, we propose to stratify projects by project attributes (programming language, and architecture et al.) which have strong relationships to productivity when building effort estimation models. We analyzed 43 projects data selected from 153 projects collected by Economic Research Association, and concluded that team size per project size had strongest relationship to productivity (variance explained was 38%). Although in general size per project size is not fixed when estimating effort, it is not very difficult that size per project is fixed approximately. We stratified projects by team size per project size and built models for each stratified projects. As a result, the accuracy of the models showed better performance than a model built from whole projects; and, the median of relative error of estimated effort was improved from 56.6% to 35.9% when the average of relative error of estimated team size was 50%.

1 はじめに

ソフトウェア開発プロジェクトの初期段階において開発工数を見積もることは、プロジェクト完遂に必要な資源の確保や、スケジュール管理を行う上で重要である。そのために、過去のソフトウェア開発プロジェクトの実

績データを見積りの根拠に用いる定量的見積り手法が数多く提案され、用いられてきた[BOEHM1981][SHEPPERD1997][SRINIVASAN1995][TSUNODA2005]。中でも、プロジェクトのメトリクス（開発規模等）を説明変数として用い、目的変数である開発工数との関係を一次式で表現する重回帰モデルを用いた見積りは、ソフトウェアツールが普及しており適用が容易なことから、広く実施されてきた[BRIAND2000][WALSTON1977]。

† 奈良先端科学技術大学院大学情報科学研究科, Graduate School of Information Science, Nara Institute of Science and Technology
†† 財団法人経済調査会, Economic Research Association
††† 産業技術大学院大学産業技術研究科, Graduate School of Industrial Technology, Advanced Institute of Industrial Technology

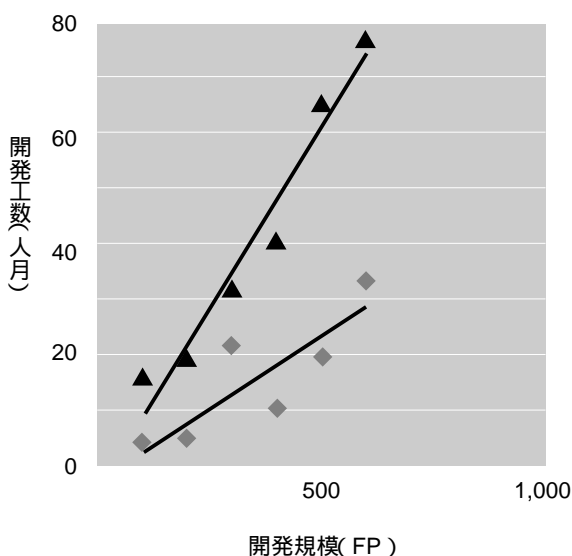


図1 プロジェクトの分類ごとに構築された見積りモデルの例

表1 生産性の統計量 (FP / 開発工数)

平均値	中央値	標準偏差	最小値	最大値
15.5	12.8	10.9	1.3	48.7

一般に、重回帰分析により工数見積りモデルを構築する際には、多様なプロジェクトを含むデータセットをそのまま用いるのではなく、何らかの基準に基づいてプロジェクトを層別（分類）したデータセットを作成し、分類ごとに見積りモデルを構築することが望ましい [HOMMA2004]。図1はプロジェクトの分類ごとに構築された見積りモデルの（単純化した）例である。各点は1つのプロジェクトを表す。この例では、2つのタイプのプロジェクトが存在し、それぞれの開発規模（FP）と開発工数（人月）の関係を直線によりモデル化している。図に示すように、異なるタイプのプロジェクトが存在する場合、1つのモデル（直線）を当てはめて見積りモデルを構築したとしても、見積り誤差の小さなモデルは得られない。このため、プロジェクトを分類するための系統的な方法が求められる。

ここで図1を詳細に観察すると、2つの直線は傾きが大きく異なっており、この傾きの違いは生産性（開発規模（FP） / 開発工数（人月））の違いであり、生産性が大きく異なるプロジェクトが混在していることが分かる。見積り誤差の小さなモデルを得るためには、生産性の違いに基づいてプロジェクトを分類（層別）し、分類ごとに

見積りモデルを作る必要がある。ただし、生産性は開発規模 / 開発工数で表され、開発工数はプロジェクトが完了するまで確定しないため、工数を見積る時点では生産性は未知である。

本論文では、見積りを行う時点において決定可能なプロジェクト特性（開発言語、アーキテクチャ等）のうち、生産性との関連が強い（寄与率の大きな）特性を用いて層別を行うことを提案する。これにより、見積りを行う時点において決定しているプロジェクト特性に基づいて、利用する見積りモデルを決定することが出来る。

本論文では、財団法人 経済調査会が収集した43プロジェクトの実績データを用いて、各プロジェクト特性の（生産性に対する）寄与率を求める。そして、寄与率の大きなプロジェクト特性を用いてプロジェクトを層別し、見積りモデルを構築する。層別の効果を評価するために、層別しない場合の見積りモデルと見積り精度を比較する。

以降、第2節では分析対象のデータについて説明し、第3節では生産性とプロジェクト特性との関連について分析する。第4節では生産性と関連の強いプロジェクト特性に基づいてプロジェクトを層別し、モデルを構築した場合の見積り精度を比較する。第5節では実験結果に対する考察を行い、第6節では関連研究について説明し、第7節でまとめを述べる。

2 分析対象データ

分析に用いたデータは、日本ファンクションポイントユーザ会FP法利用検討会（JFPUG）及び奈良先端科学技術大学院大学との共同調査として、財団法人 経済調査会が平成18年度に110社から収集した、153プロジェクトの実績値である。データには、従業員数4人の企業から従業員数約40,000人の企業のプロジェクトまで含まれており、非常に多様であるという特徴がある。プロジェクトごとに、約140個の変数（プロジェクト特性）が記録されている。ただし、変数によっては欠損値（値が未記録であること）が多く含まれている場合があった。

分析対象のプロジェクトの条件を揃えるため、下記の条件に従ってデータを抽出した。

- ・新規開発のプロジェクト（改造のプロジェクトは分析

表2 分析対象のプロジェクト特性

プロジェクト特性	種別	件数	詳細説明
発注者分類	カテゴリ	42	府省庁 地方自治体 民間企業(親会社) 民間企業(親会社以外)
システム構成	カテゴリ	42	クライアントサーバシステム Web系システム メインフレームシステム
適用分野	カテゴリ	40	事務系 制御系
適用業種	カテゴリ	40	建設業 製造業 電気・ガス・熱供給・水道業 情報通信業 運輸業 卸売・小売業, 金融・保険業 不動産業 飲食店・宿泊業 医療・福祉 教育・学習支援業 公務
サーバOS	カテゴリ	37	UNIX Linux Windows NT 系
データベース	カテゴリ	42	Microsoft SQL Server Oracle その他(オープンソース) その他(非オープンソース)
開発技法	カテゴリ	39	構造化 DOA オブジェクト指向
開発作業標準	カテゴリ	39	使用した 使用しなかった
主開発言語	カテゴリ	32	使用割合が50%以上の開発言語 ASP ASP.NET COBOL Delphi Java SQL VB VB.NET
平均要員数	数値	40	平均要員数 = $\frac{\text{開発工数}}{\text{開発期間}}$
開発期間	数値	40	開発期間の実績値
FP	数値	43	未調整ファンクションポイントの実績値
規模当たり期間	数値	40	規模当たり期間 = $\frac{\text{平均期間}}{\text{FP}}$
規模当たり要員数	数値	38	規模当たり要員数 = $\frac{\text{平均要員数}}{\text{FP}}$
基本設計比率	数値	28	全工程に対する基本設計工程の割合
詳細設計比率	数値	35	全工程に対する詳細設計工程の割合
PG設計製造比率	数値	37	全工程に対するプログラム設計・製造工程の割合
ソフトウェアテスト比率	数値	35	全工程に対するソフトウェアテスト工程の割合
システムテスト比率	数値	31	全工程に対するシステムテスト工程の割合

から除外した)。

- ・基本設計，詳細設計，プログラム設計・製造，ソフトウェアテスト，システムテストの工程比率の合計が100%のプロジェクト。もしくは，各工程の比率は欠損しているが，開発工程の範囲が基本設計からシステムテストまでのプロジェクト（一部の工程しか実施していないプロジェクトは分析から除外した）。
- ・FP（ファンクションポイント），開発工数が計測されているプロジェクト（生産性を計算するためにはFP，開発工数が計測されている必要がある）。

データを抽出した結果，分析対象のプロジェクトは43件となった。分析対象のプロジェクトのうち37件は事務系，4件は制御系，残りはその他のシステム開発であり，組込み系システム開発のプロジェクトは含まれていなかった。1件のプロジェクトを除き，開発方法論はウォーターフォールであった。

本論文では，FPを基本設計からシステムテストにおける開発（総）工数（人月）で割った値を生産性と定義した。表1に抽出した43件のプロジェクトにおける生産性

の統計量を示す。生産性の第1四分位数（生産性を高い順に並べた場合の下位25%）と第3四分位数（生産性を高い順に並べた場合の上位25%）を比較すると，約3.3倍の差があった。

分析を行ったプロジェクト特性を表2に示す。欠損値の多いプロジェクト特性や，ほとんどのプロジェクトで同一の値となっていたプロジェクト特性は分析から除外した。ただし，分析対象の変数にも欠損値が含まれているため，表2に示すように，変数の値が記録されているプロジェクト数はそれぞれ異なる。

データに含まれていない平均要員数，規模当たり要員数，規模当たり期間を新たに定義した。平均要員数は開発工数を開発期間で除したものである。一般に，開発要員数と開発期間は開発規模（FP）に比例して大きくなるため，規模でそれぞれを除すことにより正規化した，規模当たり要員数と規模当たり期間を定義した。

なお，一般に開発要員数は見積り工数から決定することが多く，工数見積りモデルを利用する際には（開発工数が未確定の時点では）開発要員数や規模当たり要員数が確定していないことが多い。ただし，モデルを利用す

際には、開発要員数や規模当たり要員数の正確な値を決定する必要はなく、後述するように概算値（大，中，小の3段階）を決定するだけでモデルを利用することが可能であるため、分析対象に含めた。

3 生産性の分析

本論文では、以下の手順で分析を行う。

手順1 分散分析の寄与率を用いて、生産性と関連の強いプロジェクト特性を明らかにする。

手順2 生産性と関連の強いプロジェクト特性に基づいてプロジェクトを層別し、工数見積りモデルを作成する。

本節では、生産性と関連の強いプロジェクト特性を明らかにし、第4節においてプロジェクトを層別した工数見積りモデルを作成する。

3.1 生産性と関連の強い変数の特定

生産性と関連の強いプロジェクト特性を特定するために、一元配置分散分析の有意確率と（調整済み）寄与率を求め、生産性と各プロジェクト特性との関連の有無と強さを分析した。一元配置分散分析は、カテゴリデータと数値データの変数の関連の有無と強さを分析する手法であり、有意確率が有意水準未満になった場合、そのプロジェクト特性と生産性は統計的に見て関連があることを示す（本論文では有意水準を10%とした）。また、寄与率は関連の強さを表す。寄与率は0%（場合によっては負の値となる場合もある）から100%の値をとり、寄与率が大きいほどプロジェクト特性と生産性の関連が強いことを示す。寄与率は以下の式によって求める[WINER1991]。

$$\text{寄与率} = \frac{SSB - (k - 1)MSE}{SST + MSE} \quad (1)$$

ここで、 SSB は群間平方和、 SST は総平方和、 MSE は平均二乗誤差、 k はグループ数を表す。寄与率はデータ全体のばらつき（ SST ）に対し、各要因のばらつき（ SSB ）がどの程度寄与しているかを示している。 MSE はデータの分布の偏りを補正するために用いられており、 k は各要因のカテゴリ数の違いを補正するために用いられている。

一元配置分散分析に当たり、すべてのプロジェクト特性をカテゴリデータの変数として扱うため、数値データのプロジェクト特性をカテゴリデータに変換した。これ

により、平均要員数等の数値データのプロジェクト特性と、開発言語等のカテゴリデータのプロジェクト特性の、生産性に対する関連の強さが比較可能となる。各特性の第1四分位数以下（値を大きい順に並べた場合の下位25%以下）の値を「低位」カテゴリとした。同様に、各特性の第3四分位数以上（値を大きい順に並べた場合の上位25%以上）の値を「高位」カテゴリとし、残りの値を「中位」カテゴリとした。これらのカテゴリの閾値を表3に示す。なお、カテゴリデータに変換することにより、本来あったはずの関連性が消える危険性があるため、数値データのままで順位相関係数を算出し、離散化を行った場合と同様の傾向が見られるかどうかを確かめる。

表3 各プロジェクト特性のカテゴリと閾値との対応

プロジェクト特性	「低位」カテゴリ	「高位」カテゴリ
平均要員数	2.6	14.1
開発期間	5.7	10.3
FP	262.5	1,494.0
規模当たり期間	0.007	0.018
規模当たり要員数	0.005	0.012
基本設計比率	12.9	20.3
詳細設計比率	14.9	25.0
PG設計製造比率	30.0	40.0
ソフトウェアテスト比率	15.0	25.0
システムテスト比率	10.0	20.0

表4 生産性に対する関連の有無及び関連の強さ

プロジェクト特性	寄与率	有意確率
規模当たり要員数	38%	0%
平均要員数	31%	1%
規模当たり期間	19%	1%
開発期間	9%	6%
適用業種	7%	27%
FP	7%	9%
サーバOS	6%	13%
発注者分類	5%	19%
データベース	3%	27%
適用分野	2%	21%
開発作業標準	-1%	37%
開発技法	-1%	43%
基本設計比率	-3%	56%
主開発言語	-4%	57%
システム構成	-4%	80%
システムテスト比率	-5%	71%
PG設計製造比率	-5%	88%
詳細設計比率	-6%	92%
ソフトウェアテスト比率	-6%	97%

一元配置分散分析の結果を表4に示す。関連があったプロジェクト特性の有意確率を斜体としている。5つのプロジェクト特性（規模当たり要員数，平均要員数，規模当たり期間，開発期間，FP）が生産性と関連があり（有意水準が10%未満となり），残りのプロジェクト特性（適用業種，サーバOS，発注者分類等）は生産性と関連があるとは言えなかった。

生産性と関連があったプロジェクト特性のうち，規模当たり要員数の関連が最も強く（寄与率 38%），FP（開発規模）は弱い関連しか見られなかった（寄与率 7%）。直感的には，規模の大きなシステムは開発が難しい（すなわち生産性が低い）と考えられるが，分析においては，開発規模（FP）は生産性との関連があまり強くなかった。さらに，規模当たり要員数の寄与率が平均要員数の寄与率よりも大きかったことから，プロジェクトの生産性に対しては，単純な開発要員数の多寡よりも，開発規模に比して適切な開発要員数かどうか重要であると言える。

数値データのプロジェクト特性と生産性との順位相関係数を表5に示す。規模当たり要員数，平均要員数，開発期間，FPの順に生産性との関連が強くなり，相関も有意であり，各工程比率は生産性との関連が弱く，相関も有意ではなかった。これは，表4の一元配置分散分析の結果と同様の傾向である。よって，カテゴリデータに変換することにより，本来あったはずの関連性が消えていることはないと考えられる。なお，規模当たり期間と生産性との相関係数が小さくなっていったが，これは規模当たり期間と生産性の関係が，3.2.3節で後述するように単調増加または単調減少でないためである。

3.2 生産性と関連が強い要因の個別分析

3.1節で生産性と統計的に関連が見られた5つのプロジ

表5 数値データのプロジェクト特性と生産性との順位相関係数

プロジェクト特性	相関係数	有意確率
規模当たり要員数	-0.65	0%
平均要員数	-0.59	0%
開発期間	-0.36	2%
FP	-0.28	7%
基本設計比率	-0.19	34%
詳細設計比率	0.09	62%
規模当たり期間	0.08	62%
ソフトウェアテスト比率	0.06	71%
システムソフト比率	-0.01	96%
PG設計製造比率	0.00	98%

ジェクト特性（規模当たり要員数，平均要員数，規模当たり期間，開発期間，FP）について，箱ひげ図を作成し，プロジェクト特性の値と生産性との関係を分析した。以降に詳細を述べる。

3.2.1 規模当たり要員数

規模当たり要員数のカテゴリ別（規模当たり要員数=「低位」，「中位」，「高位」）の生産性の箱ひげ図を図2に示す。それぞれの箱の太線は生産性の中央値を表す。丸印は生産性の外れ値を表す。外れ値は箱の上端（もしくは下端）から，四分位範囲（箱の長さ）の1.5倍以上離れた値である。星印は極値（極値は箱の上端（もしくは下端）から，四分位範囲（箱の長さ）の3倍以上離れた値）を表す。

分析の結果，規模当たり要員数が大きいプロジェクト程，生産性が低くなっていった。低位カテゴリの生産性の中央値は，高位カテゴリの生産性の中央値の約3.7倍であった。各カテゴリの生産性の違いを統計的に確かめると（マン・ホイットニーのU検定を用い，有意水準は10%と

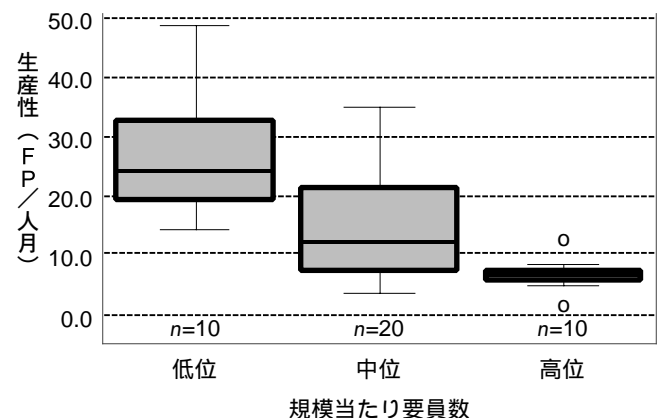


図2 規模当たり要員数と生産性の関係

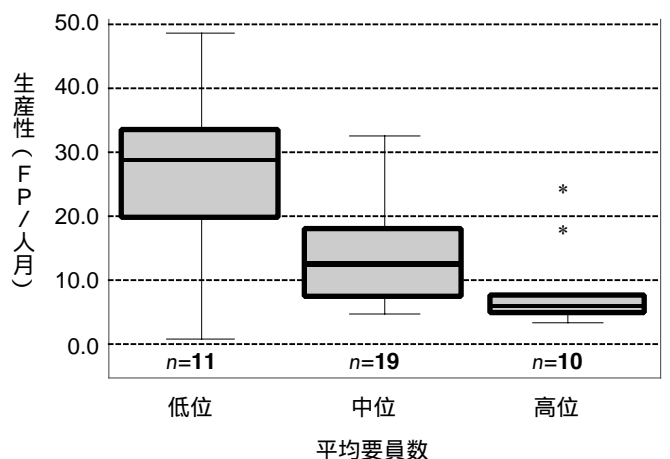


図3 平均要員数と生産性の関係

した),すべてのカテゴリ間で差があると言えた。開発期間が短く制限されている場合等,開発規模に比して開発要員を多く投入する場合(本論文で用いたデータでは1,000FP当たりの平均要員数が12人以上),とくに生産性が低くなると考えられる。規模当たり要員数と生産性との関連が大きかった理由は,規模当たり要員数が増える,開発要員間のコミュニケーションのオーバーヘッドが大きくなるためであると推測される。

3.2.2 平均要員数

図3に平均要員数によって分類したプロジェクトの生産性の箱ひげ図を示す。図より,平均要員数が増加すると,生産性が低下していることが分かる。低位カテゴリのプロジェクトの生産性の中央値は,高位カテゴリのプロジェクトの約4.5倍であった。各カテゴリの生産性の違いを統計的に確かめると,すべてのカテゴリ間で差があると言えた。平均要員数が多い(14人以上)プロジェクトでは,生産性は低くなると考えられる。

3.2.3 規模当たり期間

図4に規模当たり期間によって分類したプロジェクトの生産性の箱ひげ図を示す。規模当たり期間が中位のプロジェクトの生産性が高くなっていた。中位カテゴリの生産性の中央値は,低位カテゴリの生産性の中央値の約2.7倍であった。高位カテゴリの生産性の中央値は,低位カテゴリの生産性の中央値の約1.1倍であり,ほとんど差が無かった。各カテゴリの生産性の違いを統計的に確かめると,中位カテゴリとその他のカテゴリ間で差があると言えた。開発規模に比して開発期間を短くすると(1,000FP当たり開発期間が7ヵ月以下)生産性は低下するが,逆に長くしすぎても(1,000FP当たり開発期間が

18ヵ月以上)生産性は低くなると考えられる。

3.2.4 開発期間

図5に工期に応じて3つに分類した場合の,プロジェクトの生産性の箱ひげ図を示す。開発期間が低位の(開発期間が短い)プロジェクトの生産性は,開発期間が中位や高位のプロジェクトの生産性よりも比較的高くなっていった。低位カテゴリの生産性の中央値は,高位カテゴリの生産性の中央値よりも約3.1倍大きかった。各カテゴリの生産性の違いを統計的に確かめると,低位カテゴリと高位カテゴリの間で差があると言えた。

ただし,各カテゴリの生産性の中央値を比較すると差があるが,四分位範囲(箱の位置)の差は比較的小さかった。すなわち,開発期間が低位の(短い)プロジェクトが,開発期間が高位の(長い)プロジェクトよりも生産性が高いということが比較的発生しやすいということであり,開発期間がプロジェクトの生産性に与える影響が小さいことが分かる。0の結果と合わせると,プロジェクトの生産性に対しては,単純な開発期間の長短より

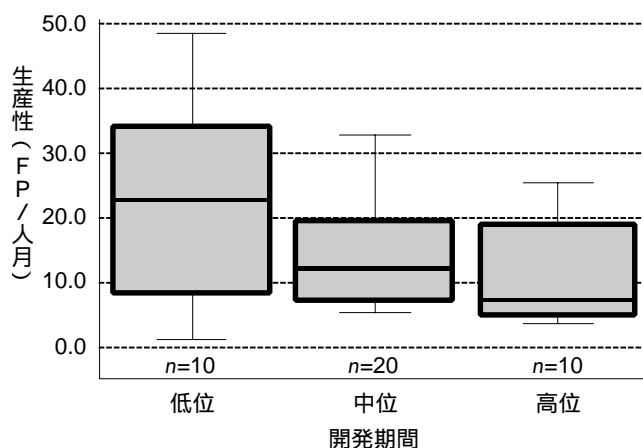


図5 開発期間と生産性の関係

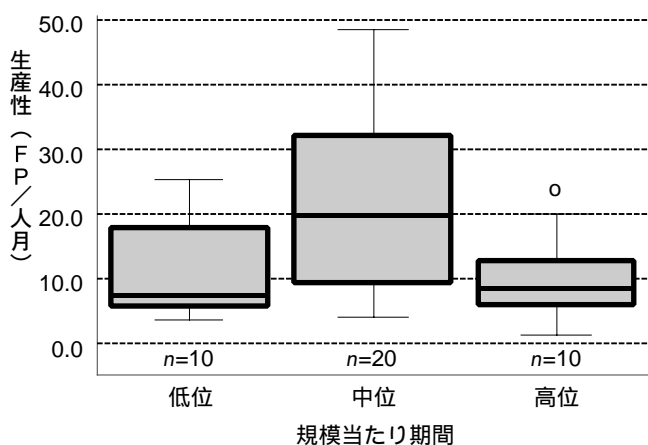


図4 規模当たり期間と生産性の関係

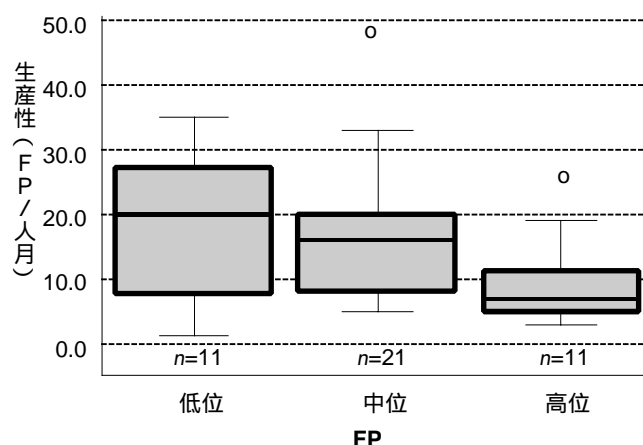


図6 FPと生産性の関係

も、開発規模に比した開発期間の長短のほうが影響していると考えられる。

3.2.5 FP

FP（開発規模）によって分類したプロジェクトの生産性の箱ひげ図を図6に示す。低位カテゴリのプロジェクトの生産性の中央値は、高位カテゴリのプロジェクトの生産性の中央値よりも約3.0倍大きかった。各カテゴリの生産性の違いを統計的に確かめると、高位カテゴリと他のカテゴリとの間で差があると言えた。図からも、FPが高位の（開発規模が大きい）プロジェクトの生産性は比較的低くなっており、FPが中位や低位のプロジェクトでは生産性の差は小さいことが分かる。システムの規模が特に大きい（本論文で用いたデータではFPが1,500以上）プロジェクトの場合は、生産性が低下しやすい傾向があると言える。

4 工数見積りモデルの構築

第3節の分析結果より、規模当たり要員数が最も生産性と関連が強いことが分かった。そこで、規模当たり要員数に基づいてプロジェクトを層別して工数見積りモデルを構築する。そして、従来から生産性と関連が強いことが明らかとなっている平均要員数に基づいてプロジェクトを層別してモデルを構築した場合との工数見積りの精度を比較する。また、開発規模により層別してモデルを構築することが一般に行われているため、この場合との見積り精度の比較を行うと共に、プロジェクトを層別せずにモデルを構築した場合との見積りの精度の比較も行う。

実験では、モデル構築時に用いるプロジェクト（学習データ）を規模当たり要員数が小（規模当たり要員数 0.005）、中（ $0.005 < \text{規模当たり要員数} < 0.012$ ）、大（規模当たり要員数 0.012）に層別し、各層ごとに工数見積りモデルを構築した。そして、例えば見積り対象プロジェクトの規模当たり要員数が大（規模当たり要員数が0.012以上）となるとき、規模当たり要員数が大のプロジェクトを用いて作成したモデルを用いて見積りを行った。平均要員数に基づいてプロジェクトを層別してモデルを構築し、見積る場合も同様の手順で行った。プロジェクトを層別する際の閾値は表3の値を用いた。

なお、開発要員数は見積り工数から決定することが多

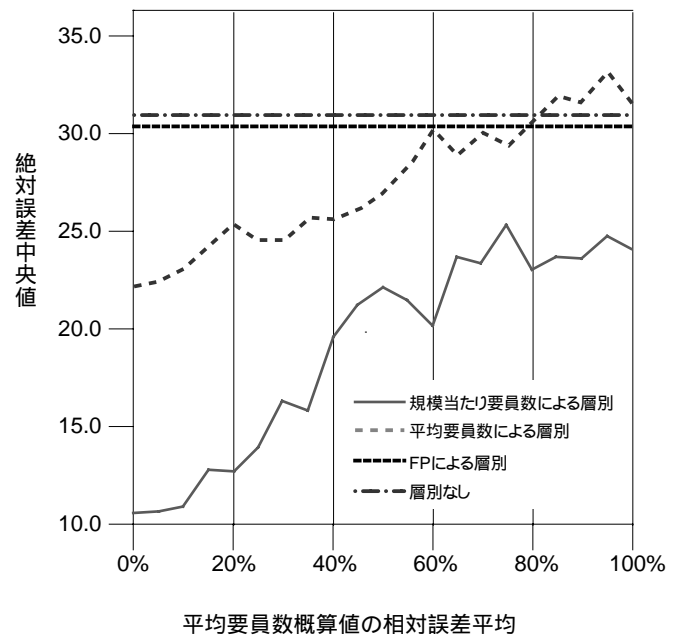


図7 見積り工数の絶対誤差中央値と平均要員数の概算値の相対誤差平均との関係

いため、工数見積りモデルを利用する際には（開発工数が未確定の時点では）開発要員数が確定していないことが多い。この場合、開発要員数を概算で決定し、開発要員数の概算値とFPにより規模当たり要員数を求め、規模当たり要員数に応じて使用するモデルを決定する必要がある。ただし、開発要員数の概算値には誤差があると考えられるため、使用するモデルを誤って選択する可能性がある。

そこで本論文では、開発要員数の概算値に平均 $n\%$ の相対誤差が含まれると仮定し、モデルの選択を誤った場合に、工数見積り精度がどの程度変化するかを実験により確かめた。開発要員数の相対誤差は正規分布に従っていると仮定し、開発要員数の相対誤差平均 n を5%刻みで変化させると共に、プロジェクトごとに誤差を含んだ平均要員数の概算値を20回生成し、工数見積りモデルの精度を確かめた。

工数見積りモデルは回帰分析により作成した。(重)回帰分析は、工数見積りモデルの構築に広く用いられている手法である。モデルの説明変数はFPのみとした。表2のプロジェクト特性のうち、工数見積り時に確定しており、かつ工数見積りに役立つと考えられる適用業種と主開発言語を説明変数に含めて工数見積りモデルを構築す

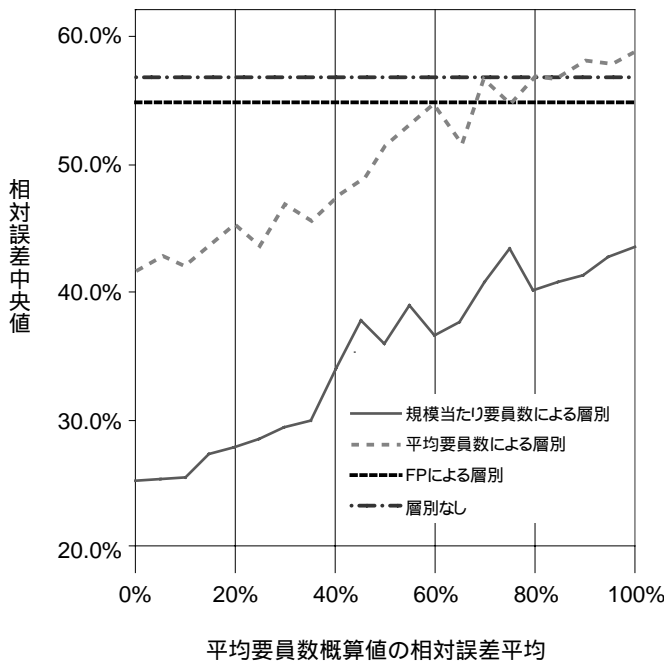


図8 見積り工数の相対誤差中央値と平均要員数の概算値の相対誤差平均との関係

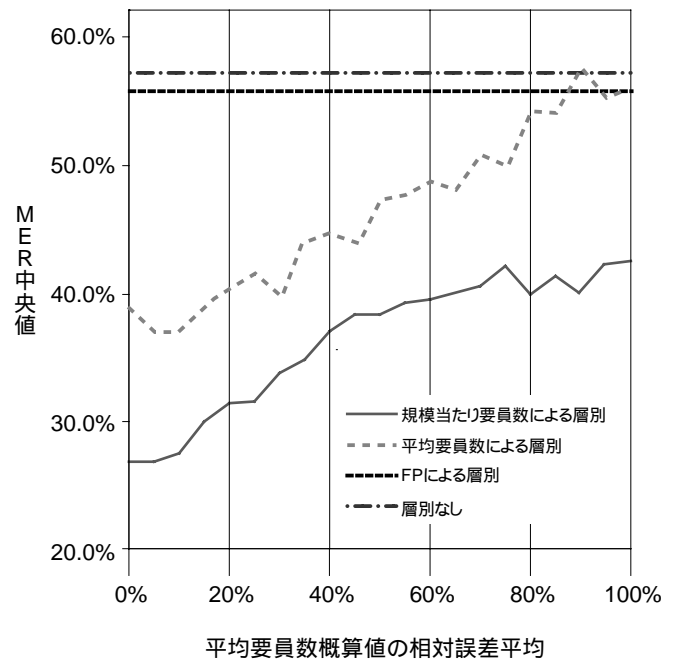


図9 見積り工数のMER中央値と平均要員数の概算値の相対誤差平均との関係

る予備実験を行ったが、工数見積りに役立つとは言えなかった（ステップワイズ重回帰分析を行った結果、説明変数として採用されなかった）ため、これらを説明変数に含めなかった。また、開発期間は工数見積りモデルの利用時（開発工数が未確定のとき）に確定していない場合があるため、開発期間と規模当たり開発期間を説明変数に含めなかった。

正規Q-Qプロット[TANAKA1995]により変数の分布を確かめると、FPと開発工数が正規分布していなかったため、それぞれを対数変換してから回帰分析を行った。また、全プロジェクトに対して回帰分析を行い、CookのDが2.73となったプロジェクト1件を外れプロジェクトとみなし、これを除外して実験を行った。CookのDは外れケース（プロジェクト）を特定する手法であり[TANAKA1995]、一般にCookのDが1を超えているケースは外れケースとみなされる。

実験ではリーブワンアウト法を用いた。リーブワンアウト法とは、全プロジェクトのうち1つのプロジェクトをモデル評価用のテストデータ、残りのプロジェクトをモデル構築用の学習データとすることを、すべてのプロジェクトに対して行うことにより、モデルを評価する方法である[HASTIE2001]。

モデルの精度の評価基準として、一般的に用いられる絶対誤差（ $|\text{見積り値} - \text{実測値}|$ ）、相対誤差（絶対誤差 / 実測値）[CONTE1986]、MER（絶対誤差 / 予測値）[KITCHENHAM2001]を用いた。これらの値が小さいほど、モデルの精度が高いことを示す。

図7から図9に各見積りモデルの精度を示す。層別なしの場合、FPにより層別した場合は各評価指標の中央値である。規模当たり要員数により層別した場合、平均要員数により層別した場合は、各評価指標の中央値20個の平均値（プロジェクトごとに誤差を含んだ平均要員数の概算値を20回生成しているため）である。

規模当たり要員数によってプロジェクトを層別してモデルを構築した場合、平均要員数の概算値の相対誤差平均が100%の場合でも、平均要員数により層別して構築したモデルや、FPにより層別して構築したモデル、層別せずに構築したモデルよりも、絶対誤差、相対誤差、MERの各中央値が小さくなっていた。

平均要員数によってプロジェクトを層別してモデルを構築した場合、平均要員数の概算値の相対誤差平均が60%を超えると、絶対誤差、相対誤差の中央値が、FPにより層別した場合とほとんど変わらないか、大きくなっていた。更に、平均要員数の概算値の相対誤差平均が

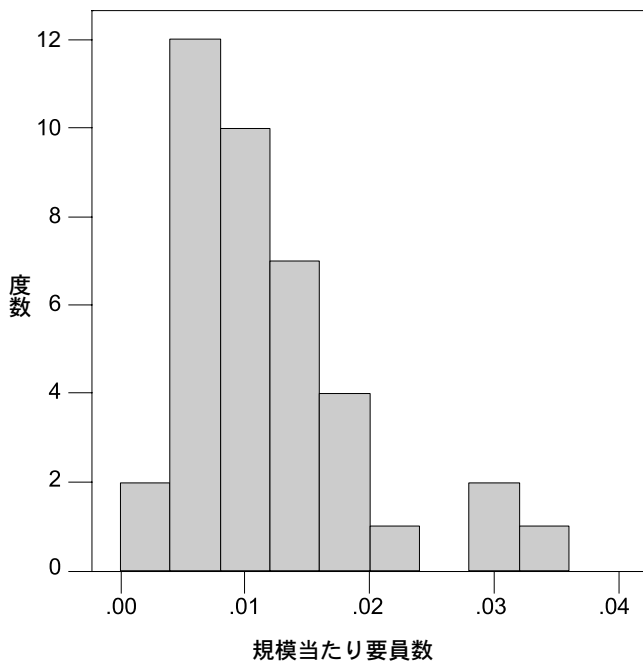


図10 規模当たり要員数の分布

85%を超えると、絶対誤差，相対誤差の中央値が，層別しない場合とほとんど変わらないか，大きくなっていった。

FPに基づいてプロジェクトを層別してモデルを構築した場合と，層別せずにモデルを構築した場合を比較すると，FPにより層別した場合の見積り精度がわずかに高いが，見積り精度に大きな違いは見られなかった。

これらの結果から，生産性に関連の強い規模当たり要員数によってプロジェクトを層別し，モデルを構築した場合，平均要員数の概算値の相対誤差平均が100%の場合でも，平均要員数により層別して構築したモデルや，FPにより層別して構築したモデル，層別せずに構築したモデルよりも見積り精度が高くなると言える。

5 考察

実験結果より，規模当たり要員数に基づいてプロジェクトを層別し，工数見積りモデルを作成することにより，見積り精度が高まることが分かった。

一般に，規模当たり要員数の大小は，工期の厳しさ [MONDEN2007] に影響されると考えられる。非常に短い開発期間で開発を行う場合，納期を守るために多くの要

員を投入せねばならず，規模当たり要員数は大きくなる。一方，開発期間に余裕がある場合にはより少人数で開発出来るため規模当たり要員数は小さくなる。このことから，本実験の結果は，工期の厳しさに応じてモデルを作ることの重要性を示唆している。逆に言えば，工期が厳しいプロジェクトとそうでないプロジェクトを混在させてモデルを作ることの危険性を示しているとも言える。

なお，開発プロジェクトの計画立案に際しては，規模当たり要員数には上限値がある（すなわち，工期を厳しくするにも限度がある）ことに留意されたい。図10に規模当たり要員数の分布を示す。規模当たり要員数が0.02以上のプロジェクトは少なく，また規模当たり要員数が0.035以上のプロジェクトは1件しか存在しなかった。よって，工期が厳しい場合でも，規模当たり要員数を0.035以上にするのは難しいと考えられる。

次に，プロジェクト特性を構成するメトリクス間の関係について考察する。層別に用いたプロジェクト特性と生産性を開発工数=MM，開発期間=DDを用いて表すと，以下ようになる。

$$\text{生産性} = \frac{FP}{MM}$$

$$\text{平均要員数} = \frac{MM}{DD} = \frac{FP}{\text{生産性}}$$

$$\text{規模当たり要員数} = \frac{\text{平均要員数}}{FP} = \frac{MM}{DD \cdot FP} = \frac{1}{\text{生産性}}$$

すなわち，平均要員数，規模当たり要員数とも，定義の分母に生産性を含んでいることになるが，これらのメトリクスが生産性と負の相関を持つことは必ずしも自明ではない。例えば，生産性（FP / MM）の分子はFPであり，分母は開発工数（MM）となっていることから，生産性は分子のFPと正の相関があり，分母の工数とは負の相関があることは一見自明であるように思われる。しかし，実際に順位相関係数を調べると，FPと生産性との相関係数は - 0.28，工数と生産性との相関係数は - 0.60となっており，いずれも生産性と負の相関となっていた。すなわち，分子にあるから正の相関，分母にあるからといって負の相関になるとは一概に言えないことになる。

また，分析対象のプロジェクト特性の多くは，生産性を構成するメトリクスを含んでおり，生産性と何らかの

関連があることは一見自明であるが、生産性に対してプラスに働くのかマイナスに働くのかは、定義からは明らかではない。例えば、FP（開発規模）は生産性に対してマイナスに働くのは一見自明のようであるが、ソフトウェア開発以外の業種（製造業等）では、規模が大きいと量産効果によって生産性が向上し、プラスに働く場合もあり得る。同様の例としてバグ密度が挙げられる。バグ密度はバグ数／開発規模により定義されるが、開発規模がバグ密度に対してプラスに働くのかマイナスに働くのかは、定義からは明らかではない。

6 関連研究

これまでソフトウェア開発の生産性について、様々な国のデータリポジトリを用いた研究が行われており、開発要員数と生産性との関連が強いことを示した研究が幾つか存在する[BLACKBURN1996][JEFFERY1996][MAXWELL1996]。ただし、規模当たりの開発要員数と生産性との関連はこれまで明らかにされていなかった。本論文では、開発要員数よりも規模当たりの開発要員数のほうが、より生産性に強い影響を与えることを示した。

開発要員数を考慮したモデルとして、開発要員数をそのまま説明変数に用いたモデルが幾つか提案されている[JEFFERY2000][MAXWELL1999][MENDES2004]。ただし、工数見積りモデルの利用時（開発工数が未確定のとき）に開発要員数が確定していることは少ないため、モデルの利用可能機会は少ないと考えられる。また、規模当たり要員数を説明変数に用いたモデルや、規模当たり要員数により層別した工数見積りモデルは提案されていない。

7 まとめ

本論文では、生産性との関連が強いプロジェクト特性に基づいてプロジェクトを層別し、工数見積りモデルを構築することを提案した。分析により、規模当たり要員数は生産性との関連が強いことを明らかにした。そして、規模当たり要員数に基づいてプロジェクトを層別して構築したモデルは、平均要員数に基づいて層別して構築したモデルや、FPに基づいて層別して構築したモデル、層別を行わずに構築したモデルよりも、工数見積りの精度

が高いことを確かめた。今後の予定は、様々なデータセットを用いて提案方法の見積り精度を確認し、結果の信頼性をより高めることである。

謝辞

本研究の一部は、文部科学省「次世代IT基盤構築のための研究開発」の委託に基づいて行われた。

参考文献

- [BLACKBURN1996] J. Blackburn, G. Scudder, and L. Wassenhove : Improving speed and productivity of software development: a global survey of software developers, IEEE Trans. on Software Eng., Vol.22, No.12, pp.875-885, 1996
- [BOEHM1981] B.W. Boehm : Software Engineering Economics, Prentice Hall, New Jersey, 1981
- [BRIAND2000] L. Briand, T. Langley, and I. Wiecezorek : A replicated assessment and comparison of common software cost modeling techniques, Proc. 22nd IEEE International Conference on Software Eng., Limerick, pp.377-386, 2000
- [CONTE1986] S. Conte, H. Dunsmore, and V. Shen : Software Engineering Metrics and Models, Benjamin-Cummings, MenloPark, California, 1986
- [HASTIE2001] T. Hastie, R. Tibshirani, and J. Friedman : The Elements of Statistical Learning: Data Mining, Inference and Prediction, Springer-Verlag, New York, 2001
- [HOMMA2004] 本間周二 : CSKに見るFP法の実践 プロジェクト定量化で品質と見積り精度向上, 日経ITプロフェッショナル, pp.53-61, 2004-7
- [JEFFERY1996] R. Jeffery : Time-sensitive cost models in the commercial MIS environment IEEE Trans. on Software Eng., Vol.13, No.7, pp.852-859, 1996
- [JEFFERY2000] R. Jeffery, M. Ruhe, and I. Wiecezorek : A comparative study of two software development cost modeling techniques using multi-organizational and company-specific data Information and Software Technology, Vol.42, No.14, pp.1009-1016, 2000
- [KITCHENHAM2001] B. Kitchenham, L. Pickard, and S. MacDonell : What accuracy statistics really measure, IEE Proceedings - Software, Vol.148, Issue 3, pp.81-85, 2001
- [MAXWELL1996] K. Maxwell, L. Wassenhove, and S. Dutta : Software development productivity of European space, military, and industrial applications, IEEE Trans. on Software Eng., Vol.22, No.10, pp.706-718, 1996
- [MAXWELL1999] K. Maxwell, L. Wassenhove, S. Dutta : Performance evaluation of general and company specific models in software development effort estimation Management Science Vol.45, No.6, pp.787-803, 1999
- [MENDES2004] E. Mendes, and B. Kitchenham : Further comparison of cross-company and within-company effort estimation models for web applications, Proc. 10th International Software Metrics Symposium, pp.348-357, 2004
- [MONDEN2007] 門田暁人, 馬嶋宏, 増田浩, 羽田野尚登, 磯野聖, 内海昭, 菊地奈穂美, 服部昇, 細谷和伸, 森和美 : 工期の厳しさに関連する要因の分析, SEC journal, No.10, pp.48-53, 2007
- [SHEPPERD1997] M. Shepperd, and C. Schofield : Estimating software project effort using analogies, IEEE Trans. on Software Eng., Vol.23, No.12, pp.736-743, 1997
- [SRINIVASAN1995] K. Srinivasan, and D. Fisher : Machine learning approaches to estimating software development effort, IEEE Trans. on Software Eng., Vol.21, No.2, pp.126-137, 1995
- [TANAKA1995] 田中豊, 垂水共之(編) : Windows版 統計解析ハンドブック 多変量解析, 共立出版, 東京, 1995
- [TSUNODA2005] 角田雅照, 大杉直樹, 門田暁人, 松本健一, 佐藤慎一 : 協調フィルタリングを用いたソフトウェア開発工数予測方法, 情報処理学会論文誌, Vol.46, No.5, pp.1156-1164, 2005
- [WALSTON1977] C. Walston, and C. Felix : A Method of programming measurement and estimation, IBM Systems Journal, Vol.16, No.1, pp.54-73, 1977
- [WINER1991] B. Winer, D. Brown, and K. Michels : Statistical Principles in Experimental Design (3rd edition), McGraw-Hill, New York, 1991

SECがスタートして4年が過ぎたが、その間に経済情勢や社会情勢はすっかり一変してしまった。経済については、米国発の金融危機が世界中を覆いつくしており、「100年に一度の危機」といわれている。とくに問題であるのは、金融業だけでなく製造業から小売業にいたるまで、あらゆる実体経済を直撃していることだ。この結果、雇用情勢は半年前には信じられないほど悪化してきた。

われわれは過去に多くの経済危機を経験してきたが、その多くは、在庫の増加による循環的な景気の減退だった。今回も、ここ数年の大量生産で在庫が積みあがっていることは間違いないが、筆者は今回の危機は従来の不景気とは一線を画するものだと思う。どういうことかと言うと、消費者がこれまで付加価値と信じてきたものに対して、それらが本当の価値なのかどうか、大枚のお金を払うに値する価値であるのかを疑い始めたということだ。この消費者心理の変化は大きい。自動車でいうと100にも上るマイコンを搭載した高級化や高機能化、家電では数ヵ月毎のモデルチェンジや極薄で超高精細のディスプレイ、携帯では鏡のような外装や使いきれないほど豊富なアプリケーション、小売では豪華な店舗やうるさいほどの接客等に、一斉にノーを突きつけ始めた。この結果、価格低下と売上減少に歯止めがかからなくなっている。いずれは到来すると思われていた「モノ離れ」という現象がついに現実化したのだ。

ダニエル・ベルが、「今後の社会は脱工業化社会」と喝破したのは40年以上も前のことだ。多くの人が、「20世紀は工業製品の大量生産の時代。合理主義の時代。21世紀は情報・サービス、感性や心の時代」と教えられてきたが、頭が完全に切り替わっている個人や企業はごく少数にすぎない。P. ドラッカーは、すでに起きてしまってもはや元に戻らないが、一般には認識されていない変化

のことを、「すでに起った未来 (The future that has already happened)」と表現した。モノ離れは、典型的な「すでに起こった未来」である。

「100年に一度の危機」ということは、「100年に一度のチャンス」と考えることもできる。モノは既に十分すぎるほど存在し、世の中に、家庭に溢れ返っている。たとえ目の前でモノが飛ぶように売れたとしても、売れているのは「モノ」ではなく、「感性」が売れていると見切らなければならない。iPodに代表される米国のアップル、スウェーデンのH&Mやイケア、フィンランドのノキアなどを観察すると、シンプルな機能をすっきりしたデザインとリーズナブルな価格で実現すれば、今なお顧客の心をつかめるということがよくわかる。顧客ニーズというものに対して、圧縮に圧縮を積み重ねたユニクロとマクドナルドが絶好調ということもこの文脈で理解できる。余談だが、筆者の職場の近くの新宿スバルビルのマクドナルド店は、価格に比して味のよいコーヒーとハンバーガーを厳選して圧倒的な集客を誇っている。ここにチャンスの芽がある。

ソフトウェア業界にもチャンスが到来した。2009年は、多くのユーザ企業が生き残りをかけて、まやかしの付加価値と決別し真の付加価値を模索し始める年だ。苦しいけれども、これができなければ日本の製造業や小売業は立ち行かない。ソフトウェア企業も同じだ。真の価値をユーザと共に追及し、抜群のソフトウェア実現力(価格、品質、納期)を自ら身につけることで、グローバル競争に立ち向かうユーザ企業のよきパートナーにならなければならない。まやかしの付加価値を大量に製造するソフトウェア量産時代は、二度と到来しない。

いま、Googleで「ソフトウェアエンジニアリング」と入れてみると、なんと222万件もヒットする。「software engineering」では3,650万件、「software engineering center」

特別寄稿

21世紀のパラダイムシフトが、 日本の復活につながる イノベーションではないだろうか

IPA顧問
鶴保 征城

では1,060万件とさらに膨大になる。言葉の由来から、研究、技術、応用、書籍、ブログまで、さらには、SECの設立経緯や過去の国家プロジェクトの功罪まで、ソフトウェアエンジニアリングの森羅万象を調べることができる。学の立場は別として、実用的にはソフトウェアエンジニアリング研究というストックはすでに十分すぎるほど存在すると言っても過言ではない。しかも、信じられないほどのスピードで日々増加している。明らかに、問われているのは、ありあまるストックを整理し、それを使用（フロー）に変えることができるかどうかだ。ストックからフローに転じることによって、初めて価値が生ずる。

モノだけでなく研究や技術も世の中に溢れているという認識を持たざるを得ない。このような状況からSECは、ゼロからソフトウェアエンジニアリングの研究を立ち上げるというよりも、情報の収集 整理 分析 仮説化 実証 普及・フィードバック 最新情報収集...というループを重視してきた。これはストックをフローに転化させるということだ。代表例として、プロジェクト定量データの収集・分析があげられる。社内機密という名のもとに各社内にストックされていた状況から、タスクフォースという場に公開され、企業間で横並びの分析がなされ、さらに学の知見や新たなプロジェクトへの適用研究がなされて、見事にフローに転化できた。極論すれば、各社内に死蔵されていたデータが見事によみがえり、プロジェクトの実態を語り、対策を示唆し始めたのだ。この過程で、提供を決断した企業や検討に参加した企業が、多くのスキルを習得できたことはいうまでもない。

別の見方をすると、各社の私有財から多くの企業が共有する公共財に転じたということでもある。20世紀を通じて、工業化社会の超優等生であった日本のストックは世界で群を抜いている。だが、ストックのままでは価値

がなく、それらを整理・分類してフローに転化でき、さらに公共財に転化出来たときに、はじめて大きな価値を生じる。人材も同じだと思う。産学連携の下に新たな人材を育成すること、海外に優秀な人材を求めることはもちろん重要であるが、企業内にストックされている多くの優秀な人材を、フローとして、また社会的な公共財として活用することが先決だ。そもそも人材の公共財化がなければ、研究や技術の公共財化はありえない。SECに参集された人材は、企業の決断によって公共財化された人材の好例だと思う。

ダニエル・ベルに擬えると、「20世紀はストックを量産し私有する時代。21世紀はそれらをフロー化し活用する時代、公共財として共有する時代」ということだ。地道ではあるが、このパラダイムシフトが、日本の復活につながるイノベーションではないだろうか。

大きく括れば、SEC活動のループは2008年度から2巡目に入っている。上述のような活動を通じて、SECはソフトウェアエンジニアリングという膨大な情報の海におけるナビゲータの役割を果たしてきたつもりだ。これからもその役割は変わらないだろう。

4年余にわたり、SECをご支援いただいた産学の皆様には心から感謝いたします。産業界の方々には、何とでも現下の厳しい経済危機を突破され、筋肉質でたくましい業界に再生されることを期待しています。学には、学の論理に捉れることなく、社会が求めるプロフェッショナル育成の原点に立ち返っていただきたいと思います。筆者は昨年未だにSEC所長を退任いたしました。今後もIPA顧問として、また一ソフトウェアエンジニアとして活動してまいりますので、倍旧のご支援をお願い申し上げます。

鶴保 征城(つるほ せいしる)

1966年大阪大学大学院工学研究科電子工学専攻修士課程修了後、日本電信電話公社(現NTT)入社。NTTソフトウェア研究所長、NTTデータ通信株式会社取締役開発本部長、同社常務取締役技術開発本部長、NTTソフトウェア株式会社代表取締役社長を歴任し、2004年10月独立行政法人 情報処理推進機構 ソフトウェア・エンジニアリング・センター所長に就任。2009年1月よりIPA顧問。高知工科大学教授。工学博士。

NASSCOM Quality Summit 2008 参加報告

SEC副所長 立石 謙二
SEC研究員 新谷 勝利

2008年10月15日(水)～16日(木)に、インド・バンガロールで開かれたNASSCOM Quality Summit 2008に筆者は参加した。また、今回の訪印を利用して、主要会員の数社を訪問し、意見交換をした。この出張報告で、インドにおけるソフトウェア開発業界の現状を読者の皆様感じていただければ幸いである。

1 NASSCOM Quality Summitからの所感

最終日の最終セッションは、MindTree²の創立者の1人で、現在ガードナー(庭師)という地位についているSubroto Bagchi³氏の以下の力強いメッセージで締めくくられた。「品質の専門家である皆さんは、もっと経営層に対して発言をするべきである。言いなりになってはいけない。品質は皆さんの活動にかかっていることを認識して欲しい。」

今回のサミットは、Dr. Keeni⁴を含む7名のプログラム委員が準備をしたもので、対象はNASSCOM参加企業のシニアマネージャ以上で、セッションチェアは、NASSCOM参加企業のバイス・プレジデント(日本では、事業部長級)レベルの人であった。品質担当スタッフだけで品質が維持出来るとは筆者は考えないが、最終セッションのアジテーション(あるいは、動機付け)にあるように、職務として品質に責任を持つものをそれなりに処遇しているということであろう。また、トップダウンで品質に関する推進をしているという非常に強いメッセージを発信していることを感じた。そもそも、彼らシニアマネージャがNASSCOM Quality Summitに積極的に関与していることが、NASSCOM参加企業における品質への取り組みへの真摯さを感じさせるものであった。

以降、聴講したセッションの一部を報告する。



会場周辺。交通ルールはあって無いようなもの。理路整然と議論が交わされる一方で、こんな混沌が共存している。

2 セッション報告⁵

セッション1: CEO Speak-Quality Strategies in the boardroom

品質と言えば日本が引き合いに出されるが、セッション1では、ベンチマーキング⁶の重要性を強調。セッション1のチェアは、P-CMM⁷の開発者として知られるBill Curtisであったせいもあり、開発プロセスだけでなく、人材も育成されなければ良い品質のソフトウェアは作れないとのコメントあり。

セッション4A: Practitioner Speak: Why industry is not Agile in adopting Agile

セッション4Aではアジャイル開発を導入する企業が増加する反面、多くの企業がそのメリットを活かしきれていないという問題意識が議論され、以下の状況が発表された。

会員4社に1社が何らかの形で既にアジャイル開発を採用。売上ベースでは前年比53%増。

オンショアにおけるコミュニケーションが幾何級数的に増大していることへの対応と、有効なベンチマ

1 NASSCOM: National Association of Software and Services Companies. インドソフトウェア・サービス協会、1988年に設立。詳しくは、<http://www.nasscom.in/Default.aspx>を参照。

2 Mind Tree: 1999年に当時WIPROの副会長・社長をはじめとする10人により設立され、現在約5,800人。年間売上は約230億円。

3 Subroto Bagchi: 約100人のエグゼクティブのメンター及び地域活動担当。

4 Dr. Keeni: Tata Consultancy Services Limitedのバイス・プレジデント。

5 発表されたものは、以下からダウンロード出来る。<http://www.nasscom.in/Nasscom/Templates/CustomEvents.aspx?id=55066>

6 NASSCOMとして、ここ数年ベンチマーキングを積極的にBangalore SPINという組織を通して推進。

ーク指標を開発することが急務。

アジャイル開発に関するよくある誤解。

- ・大規模システムには不向き プロセス標準とプランニング及びトラッキング・ツールの活用。
- ・分散開発には不向き 開発サーバとテストサーバを分離しリアルタイム連携。米Microsoft社はまさに実践している。
- ・請負契約には不向き 効果的な変更管理プロセスの導入である程度までは対応可能。

ただし、契約面で、価格固定の請負契約では早晚限界を迎え、顧客が容認する固定スコープ契約に移行。

セッション7A : Beyond certification

セッション7Aは一番時間枠の大きなセッションで、筆者の一人である新谷はこのセッションでSECにおけるプロセス改善WGをSECの他の活動とのシナジー効果を考慮しながら実証実験を計画していることを説明した。

Bill Curtisは、CMMIレベル5に対してIEEEサーベイによると必ずしも顧客は満足していないことを示しながら、今後はソフトウェアの第4の波⁸、アーキテクチャに基づくソフトウェア開発に移っていくであろうと言及。

アクセンチュアでは、実に多くの認証をとってきた。その経験から、組織の成長に合わせ、最適の認証の組み合わせプログラムの必要性を強調。

同様に多くの認証をとってきた別の会社の人は、認証はその組織に意味があるときにのみ実施するのが



WIPRO社訪問の様子。TCS (TATA Consultancy Services) キーニ副社長 (最右)、WIPROラビチャンドラ副社長 (右から二番目)

適切と指摘。

モデルとは何かを認識し、モデルに記述されていることを忠実に評価することではないと指摘し、資格取得が目的化することへの警鐘を鳴らした。

セッション9B : Role of Quality in Process Transformation

オフショアに対する顧客の期待は以下のように変化してきている。

- 第1段階 : 経費削減
- 第2段階 : 品質とSLAの向上
- 第3段階 : 継続的価値付加及びプロセス改善

プロセス イノベーションへのシフトは、「価値」により測定されている。このためには、ドメイン・エキスパートになることが必須であり、そのためには、顧客とのコミュニケーション力が極めて重要。

3 訪問企業からの報告

L&Tインフォテック社⁹

1,000人を超える新入社員の教育方法についての説明を受けた。

- ・技術教育そのものは大学で実施しており、企業ではプロセス等の組織的なソフトウェア開発の環境・仕組みを1~3ヶ月で実施。
- ・どこでも、いつでも学習出来るように、Eラーニングを主体。
- ・リーダーシップ開発に注力。
- ・KPO (Knowledge Process Outsourcing) に重点。
- ・5年間で10,000人増員し、毎年40%の売上増を計画。

WIPRO社¹⁰

日本とインドのIT産業の有り様についての意見交換。

- ・多重階層の契約関係はインドには無い。専門分野を持つ複数のITサービス会社がお互い共同して請け負うことはあり、この場合でも2階層まで。
- ・WIPROも10年前は80%がボディショップ (日本の人材派遣に相当する用語) であったが、この方法では、社内に知的財産やノウハウが蓄積出来ないので、10年かけて今は20%程度。
- ・新入社員は、WIPROの組織人として動くことを知らないで、それに関連する教育を半年実施。

7 P-CMM : People CMMの省略形。人材育成をいかに実施しているかを成熟度の考えで評価。日本では未実践されていないようであるが、今回訪問した企業はすべてCMMIに加えてP-CMMを導入。人材育成を重要視していることが分かる。

8 第1の波 : 1980年代までのコードジェネレータ、第2の波 : 1980年度CASE、第3の波 : 1990年代のプロセス、第4の波 : 2005年以降はアーキテクチャ。

9 L&Tインフォテック社 : 約70年前にエンジニアリング会社として設立。現在ソフトウェア事業部門は従業員約13,000人。

10 WIPRO社 : 独立系のITサービス会社。現在従業員94,000人。バンガロールのキャンパスにソフトウェア開発センターを建設中で、そのキャパシティは20,000人。

ITA Information Technology Alliance

URL: <http://www.ita.gr.jp/>

ITA SE会議 H20年度幹事

矢野 達也

ITAは、IT業界の独立系中堅企業の「情報交換、共同研究活動等を行うと共に、各社が営業面、技術面で協力し合うことにより個々の能力を超えた事業展開を図る」ことを目的として結成された。平成21年は15年目を迎え、現在、会員会社15社、社員総数8,000名、総売上1,100億円超の企業集団である。

1 ITAについて

ITAは、平成7年2月にIT業界の独立系中堅企業14社が、「情報交換、共同研究活動等を行うと共に、各社が営業面、技術面で協力し合うことにより個々の能力を超えた事業展開を図る」ことを目的として結成された。平成21年は、15年目を迎え、現在、会員会社15社、社員総数8,000名、総売上1,100億円超の企業集団である。

発展し続ける社会情報システム、企業情報システムにより高度化・複合化の増す中、特定のコンピュータメーカに技術的にも資本的にも依存しない独立系情報サービス企業へのユーザ企業の期待は高まりつつある。しかしながら、ユーザ企業の様々なニーズにお応えしてしていくためには単独の企業のみでは限界がある。そこで、「1社では出来ないことを、複数の企業が集まって実現させよう」とこの考えの下にITAは誕生した。会員各社はそれぞれ専門分野を活かし、相互に補完し合い、会員企業共同による企画提案、システム構築、先端技術の研究開発、

更には次代のIT産業のリーダの育成をも目指している。

志を同じくする各社の強固な連帯の下に、独立系のよさを活かしてユーザ企業と会員企業会社が共通のメリットを創出するため、「協業」と「相互研鑽によるレベルアップ」を目指して活動を展開している。

2 ITA活動の概要

アライアンスは参加15社の代表者（経験者）によって構成される「代表会議」と、その下で活動する6つの「実務会議」と2つの「プロジェクト」（人材マネジメントプロジェクト+ETロボコン東京地区実行委員会）で活動を行っている（図1、図2）。

実務会議

各実務会議の「営業企画」・「経営管理」・「技術」の3会議には各社の役員・幹部が参画してオープンに意見を交換し、会社の運営に活かしている。また、高品質・高生産・高度人材育成と積年のIT業界での課題につ

ITA運営体制(平成21年度)

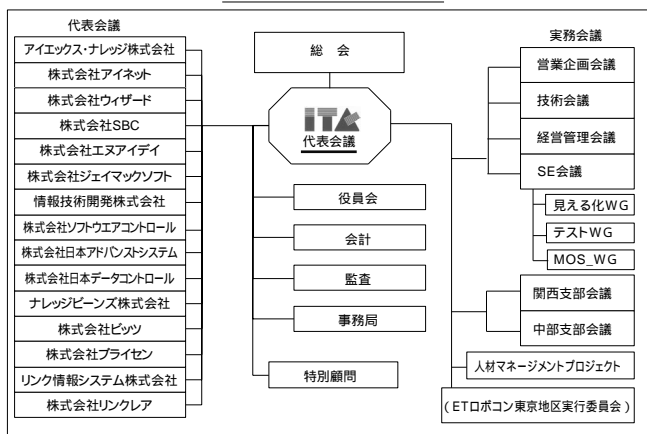


図1 運営体制

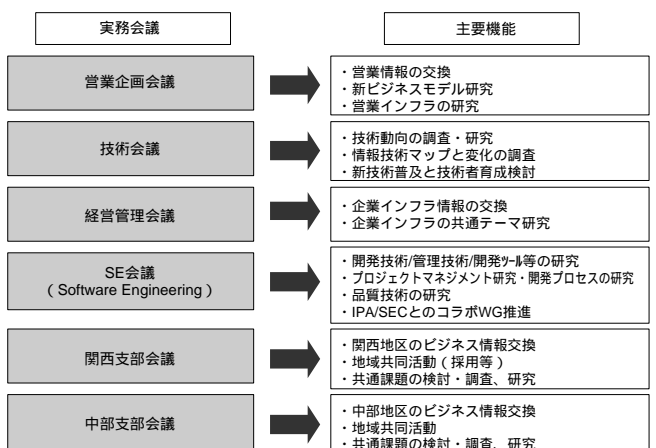


図2 実務会議の機能

いては、ソフトウェアエンジニアリングの重要性を踏まえて各社の管理、開発責任者が中心となり、IPA/SECとのコラボレーションで「SE会議」の下、活動を行っている。

ITA_Forum

毎年11月には、「ITA_Forum」にて各ワーキンググループ等の1年間の活動成果の報告セミナーを開催すると共に、最新の業界・技術動向についてご講演をいただいている（昨年はチェンジビジョン株式会社の平鍋社長にご講演いただいた）。また、各社の若手で実行委員会を組織し、「ITAロボコン」を実施している（昨年は18チーム参戦）。

ETロボットソフトウェアデザインコンテスト

平成19年より、社団法人組込みシステム技術協会（JASA）主催のETソフトウェアデザインロボットコンテスト（ETロボコン）関東大会の実施運営の依頼を請け、ITA内に実行委員会を組織し、昨年は130チームの参戦で9月中旬に西新宿の工学院大学で3日間の熱戦を繰り広げた。平成21年も支援を行う予定である。

3 SE（ソフトウェア・エンジニアリング） 会議（ワーキング活動含む）の活動

SE会議は、平成19年度ITAの活動方針「外部機関連携強化」を受けて、2007年の3月1日に発足した。IPA/SECとの緊密な連携・情報交換を図りつつ、双方にとって有用な成果を出していこうということでテーマ領域の意見交換を重ね「見える化」「見積り」「ETSS（組込みスキル標準）」を設定し、ワーキンググループ活動を開始した。昨年度は各テーマ領域を深堀、あるいは、拡張しつつ、ITA参加者各社にとって必要なもの、実際にすぐ使えフィードバック出来るもの、困っている問題の共有と解決等に焦点を当てた活動内容となった。

エンタープライズ系「見える化」ワーキンググループ

昨年度はSEC見える化チェックリスト活用評価、EPM活用評価の継続に加え、各社の「さまざまな観点での見える化」事例の相互発表・意見交換を実施した。年度の後半からは保守運用の見える化、事例収集に着手。今年度は「見える化」体系を整理しつつ「ITA見える化のプラクティス」をまとめていきたいと企図している。

エンタープライズ系「テスト（見積り）」ワーキンググループ

「CoBRA法」を活用した見積り実証の展開はシミュレ

ーションツールのオフィシャル提供を待つこととしていったん終了。そこまでのITAの取り組みはSEC報告書[SEC2008]で事例として掲出された。昨年度は「テスト（見積り）」を新たなテーマとして設定し、テスト工程の課題や事例を収集。今年度もこれを継続し「ITAテスト工程の事例集」をまとめていきたいと企図している。

組込み系人材開発（MOS：Management Of Skills） ワーキンググループ

ITA-ETSSロードマップに則り、フレームワーク標準～スキルセット策定を経てパイロット試行し、結果の解析評価を実施。これを報告書化した。今年度は、ETSS実証実験は引き続き継続し、更に教育研修基準の策定、ETEC他資格体系化の策定への取り組みを企図している。なお、2007年活動成果として公開した「組込み系受託ソフト開発企業向け汎用組込みスキル標準ITA-ETSS 2007」について内容を充実してVersion1.2を今年1月に再公開した（申し込みはITAホームページより）。

活動規模：昨年度の参加状況

各ワーキンググループ、会議への参加状況を表1に示す。

表1 各ワーキンググループ、会議への参加状況

ワーキンググループ・会議名	参加社数・人数
「見える化」ワーキンググループ	9社14名
「テスト」ワーキンググループ	10社19名
「MOS」ワーキンググループ	8社13名
「SE会議」（連絡会議）	13社23名

4 おわりに

ITAは、IPA/SECとのコラボレーション（テスト、見える化、MOS各ワーキンググループ）をより一層推進すると共に、各ITA活動の成果は、ホームページに掲載し、皆様に提供・報告させていただきます。

百年に一度と言われている非常に厳しい環境の中、今後とも「小さいながらも元気な集団」として、中堅・中小独立系IT企業の将来像を模索してまいります。

どうぞ、皆様のご指導・ご鞭撻のほどよろしくお願い申し上げます。

参考文献

[SEC2008] IPA/SEC：「先進的見積り手法実証と普及展開の調査」報告書、2008

BOOK REVIEW

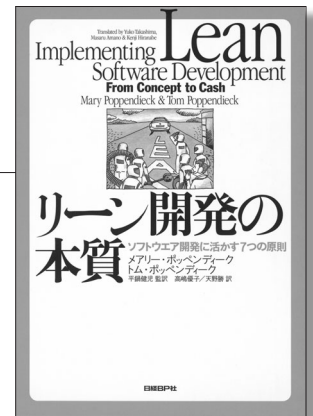
リーン開発の本質：ソフトウェア開発に活かす7つの原則

Mary Poppendieck・Tom Poppendieck 共著 平鍋 健児 監訳 高嶋 優子・天野 勝 訳

ISBN：978-4-8222-8350-6 日経BP社刊

A5判・352頁・定価2,520円（税込）・2008年2月刊

原著名：Implementing Lean Software Development, From Concept to Cash



最強の組織に学ぶプロセス改善の秘訣

米国人の著作でありながら、冒頭から豊田佐吉、豊田喜一郎、そして、大野耐一氏の話で始まる。今や世界一となったトヨタ自動車の「トヨタ生産方式」のソフトウェア開発への応用法の紹介である。原題を直訳すれば、「無駄なしソフトウェア開発の実行、概念の提示から現金化まで」というようになる。

ソフトウェア開発を、その概念の提示から最終的に現金化、言うなれば納入するまでの一連の流れと捉え、この間の無駄をトヨタ式に徹底的に排除する方法を示している。中ほどにデミング博士の話が登場し、ソフトウェア開発においても博士の指導を反映していくべきだと主張している。歴史的背景を示した後、「原則」「価値」と

して考え方を整理し、次いで、「ムダ」「スピード」「人」「知識」「品質」「パートナー」という視点から進め方を経験や事例を交えながら示している。そして、最後に整理されたまとめと、監訳者による充実したあとがきがある。

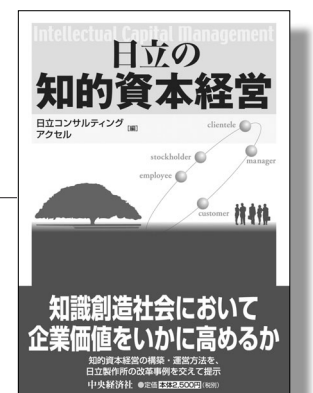
素直に読めば素晴らしく、本書の主張を実行しないことは愚かに思えるが、現実のソフトウェア産業界、とくに多重階層構造化した受託開発の領域に目を向けると、まったくの対極に感じられる。「皆で本書を読んでから議論を始めようよ」と呼びかけたい気持ちだ。人の働き甲斐に関する主張がとくに印象深い。（神谷芳樹）

日立の知的資本経営

日立コンサルティング・アクセル 共編著

ISBN：978-4-502-39430-0 中央経済社刊

A5判・240頁・定価2,625円（税込）・2007年7月刊



知的資本を武器にグローバルで戦える企業への変革のヒント

100年に1度の経済危機と言われ、正社員も含めた人員削減が行われている。本書は1990年代後半と2000年初頭、事業の屋台骨を揺るがすほどの打撃を受けた日立製作所の経営改革を記したものである。現在の経済状況で読むにはちょうどいいと考え、手に取った。

日立製作所の経営改革は、知的資本力を見直し、エクセレントカンパニーを目指すものである。「この木、何の木、気になる木」ではないが、立派な樹木の地中には立派な根が存在する。この樹木の根が、知的資本と言える。

広く知られる財務資本（資産・資本/負債）に加え、知的資本と知的資産の3つが循環する会計が求められる。この循環とは、知的資本が知的資産を生み、知的資本が財務資本を生む。そして、財務資本が知的資本を生むサイクル

である。

この知的資本の中心は人的資本である。これらのマネジメントのために、BSC（Balanced Score Card）やSECI（Socialization, Externalization, Combination, Internalization）モデルを活用し実現する。

人的資本のKPI（Key Performance Indicator）にETSSを適用出来ないかと考えた。可視化するには、指標が必要となるが、ETSSによるスキルやキャリアの可視化は、KPIに利用出来る。この人的資本という考え方をぜひ浸透させたい。そして、人への投資を推奨・推進し、開発力強化を実現したい。そのためにも、知的資本経営を広く認知・活用してもらいたいと願う。（渡辺 登）

アセスメント(assessment)

もともとは、財産評価・査定、賦課(課税額の確定とそれを負担させること)の意。日本では「評価」「査定」の意で用いられることが多い。「開発が環境に与える範囲・影響を予測・評価し、環境に配慮すること」を旨とする1999年施行の環境影響評価法関連の報道等で、「環境アセスメント法」という通称が用いられ、「アセス」という短縮形とともに一般化した。

産業界においてもいろいろな場面で使用されているが、そのいずれもが対象を評価することを中心に実施する活動を指す。SECの活動領域にある「開発プロセス」「スキル」についても、改善、向上のPDCA¹のベースとして現状の把握や活動の評価は必須であり、「プロセスアセスメント」「スキルアセスメント」と呼ばれているものがこれに該当する。アセスメントを行うには評価する項目・内容が明確に定義されている必要があり、それぞれ規格として提唱されている。

アセスメントを実施する(評価を行う)人をアセッサ(assessor)と呼ぶ。アセッサは、それぞれの規格を提唱している団体、あるいは、そこから承認された代行組織において、必要なアセスメント業務の教育がなされ、実施資格の認定を受けることによりその資格を呼称できるのが一般的である。

同様の意味合いを持つ呼称として、アプレイズ(appraise)/アプレイザ(appraiser)やエヴァリュエイト(evaluate)/エヴァリュエータ(evaluator)も使用されている。(室 修治)

エンピリカル(empirical)

“empirical”は、「実証的」と訳されることが多い。「実践的」とも訳されるが、こちらにはpracticalという単語がある。experimental(実験的)、experience(経験)等の仲間の単語。なかなかぴったりの訳語が無い。

近年、Empirical Software Engineering、エンピリカル・ソフトウェア工学という使い方で少しずつ認知度が上がってきた。他の分野では、Empirical Economics、エンピリカル経済学といった使い方を目にする。

ソフトウェア工学の世界では、物理学のような実験が難しく、誰かの発明や発見、主張を別の人が確認するいわゆる「追試」のようなことも現実的でない。また、そのデータ取得にも秘匿性、個別性といったこの領域固有の壁があり、なかなか議論の決め手がない。結果として、議論が延々と続くいわゆる「話百篇のソフトウェア工学」という状況が見られる。

これに対して、実証的なデータでもって終止符を打とう、というのがエンピリカル・ソフトウェア工学に集まった人々の考え方である。例えば、International Symposium on Empirical Software Engineering and Measurement (ESEM) という国際会議が毎年開催され、Empirical software engineering: an international journal という論文誌が発行されている。我が国では、「ソフトウェア開発におけるエンピリカルアプローチ」² という書籍が出版されている。

(神谷 芳樹)

1 PDCA : Plan-Do-Check-Act/Action

2 鳥居宏次 監修, 奈良先端科学技術大学院大学 著: ソフトウェア開発におけるエンピリカルアプローチ, アスキー, 2008

編集後記

本号から新しい編集兼発行人、すなわちソフトウェア・エンジニアリング・センター所長、松田晃一を迎えて、SEC journalも4年目の活動を進めております。創刊以来の鶴保証城(IPA顧問に就任)に変わらず、皆様のご支援、ご指導をよろしくお願いいたします。

本号はお気づきのよう各寄稿者からの大変力の入った記事が揃い、大幅増ページとなりました。ぜひご活用いただきたいと存じます。また本ジャーナルが学術論文誌としても一層活用されることを目指して、本号では「ソフトウェアに関する論文の書き方」を掲載すると共に、ノブルを学術論文誌で一般的に用いられている表現に変更いたしました。次号は恒例の「SEC成果特集号」です。そしてその後も斬新な新企画が進行中です。今後ともSEC journalにご期待ください。

(神)

SEC広報だより

年明けから順次開催している『SEC主催セミナー』ですが、毎回申し込みからすぐ満席になってしまい、ご迷惑をおかけしております。

「どうにか参加できないか」という熱心なご希望をいただけるのは嬉しい反面、お断りせざるを得ないことに、運営スタッフ一同、申し訳なく思っております。

現在、追加開催等、なるべく多くの皆様にご参加いただけるよう検討中です。また、ご参加いただけなかった場合でも、終了後講演資料をSEC Webサイトにアップいたしますので、ぜひアクセスしていただければと思います。

どうぞよろしくお願いいたします。

SEC journal 編集委員会

編集委員長	神谷 芳樹
副編集長	渡辺 登
編集委員(50音順)	市川 里恵
	遠藤 和弥
	矢野 亜希



ビル群を背に咲く梅の花。冬の日差しながら、春近しの感。

(撮影:神谷芳樹)

SEC journal® 第5巻第1号(通巻16号) 2009年3月31日発行

© 独立行政法人 情報処理推進機構 2009

編集兼発行人 〒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

独立行政法人 情報処理推進機構 ソフトウェア・エンジニアリング・センター 所長 松田 晃一

Tel.03-5978-7543 Fax.03-5978-7517

<http://sec.ipa.go.jp/>

編集・制作 〒101-8460 東京都千代田区神田錦町3-1 株式会社オーム社 Tel 03-3233-0641

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

IPA（独立行政法人 情報処理推進機構）
ソフトウェア・エンジニアリング・センターでは、
下記の内容で論文を募集します。

応募様式は、下記のURLをご覧ください。
<http://sec.ipa.go.jp/secjournal/papers.html>

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

開発現場への適用を目的とした手法・技法の詳細化・具体化などの実用化研究の成果に関する論文

開発現場での手法・技法・ツールなどの様々な実践経験とそれに基づく分析・考察、それから得られる知見に関する論文

開発経験とそれに基づく現場実態の調査・分析に基づく解決すべき課題の整理と解決に向けたアプローチの提案に関する論文

論文の評価基準

- 実用性(実フィールドでの実用性)
- 可読性(記述の読みやすさ)
- 有効性(適用した際の効果)
- 信頼性(実データに基づく評価・考察の適切さ)
- 利用性(適用技術が一般化されており参考になるか)
- 募集テーマとの関係

応募要項

投稿締切り

年4回、3ヵ月毎に締切り、締切り後に到着した論文は自動的に次号審査に繰り越されます。

(応募締切:1月・4月・7月・11月各月末日)

締切り後、査読結果は1ヶ月後に通知
詳細スケジュールについては、投稿者に別途ご連絡いたします。

提出先

独立行政法人 情報処理推進機構 ソフトウェア・
エンジニアリング・センター内 SEC journal事務局

eメール: sec-ronbun@ipa.go.jp

その他

論文の著作権は著者に帰属しますが、採択された論文については SEC journalへの採録、ホームページへの格納と再配布、論文審査会での資料配布における実施権を許諾いただきます。
提出いただいた論文は返却いたしません。

論文賞

SEC journalでは、毎年SEC journal論文賞を発表しております(前回は2008年10月28日SECコンファレンス)。受賞対象は、SEC journal掲載論文他投稿をいただいた論文です(論文賞は最優秀賞、優秀賞、SEC所長賞からなり、それぞれ副賞賞金100万円、50万円、20万円)。

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
コーディング作法
テスト/検証技術
要求獲得・分析技術、ユーザビリティ技術
見積り手法、モデリング手法
定量化・エンピリカル手法
開発プロセス技術
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

SEC journal

バックナンバーのご案内

詳しくは<http://sec.ipa.go.jp/secjournal/>をご覧ください。



No. 1 No. 2 No. 3 No. 4 No. 5



No. 6 No. 7 No. 8 No. 9 No. 10



No. 11 No. 12 No. 13 No. 14 No. 15

SEC Journal No.16
第5巻第1号(通巻16号)
2009年3月31日発行 ©独立行政法人 情報処理推進機構

編集兼発行人

〒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
独立行政法人 情報処理推進機構 ソフトウェア・エンジニアリング・センター
所長 松田 晃一

Tel.03-5978-7543 Fax.03-5978-7517
URL:<http://www.ipa.go.jp/>



IPA®

独立行政法人 情報処理推進機構

