

クオリティ ワン  
*Quality One*

Vol.4 2008 年 11 月号

Software Quality Profession

財団法人 日本科学技術連盟

## 1. 品質

### ■ プロセス監査の強化～やりきり感の見える化～

ボッシュ株式会社  
パワートレイン ECU 事業室  
玉井 治久

#### 1. はじめに

私の所属する部署は、部品サプライヤー(以下B社)としてお客様の完成車メーカー(以下A社)へ納めるエンジン制御ユニットのソフトウェア(以下SW)を開発しています。SWの品質を保つため、B社内でのValidation後、A社内でも、完成車としてValidationを行います。また、B社内では、Projectへの開発中のプロセス履行監査(以下監査)を行っています。

ある時、そのA社のSW品質監査室の方から「B社さん、A社へ開示できないノウハウを含め詳細な内容に立ち入った監査をB社で完結することにより、不具合の防止をして下さい。については、B社さんの監査の仕組みのアセスメントをします」と言われました。

そこで私どもB社の監査のアセスメントをA社さんに行っていたいただきました。その際頂いたご指摘を基に、現在仕組みの強化中です。本編では今までB社の構築してきた監査の仕組みと、現在の強化の取り組みについてご紹介しします。

#### 2. アセスメント結果と指摘事項

A社のアセスメント結果は、「監査の仕組みがあり、必要なことの大部分はカバーされている。しかしプロジェクト全体としての『やりきり感』が見えず、そのやりきりの監査が出来ていないので、やりきりを監査する仕組みを構築して欲しい」とのことでした。やりきり確認の監査の方法はB社へ任されました。

B社の監査の方法はインタビュー形式で、作業成果物の確認を行っています。そのため、今までのインタビュー形式では例えば、「コードレビューを行っていますか？」に対し、プロジェクトリーダー(以下 PL)は数件のレビュー結果(エビデンス)を示して、「はい」と答えればよいことになっていました。つまり、レビューを行った事実の一部を見せればよく、プロジェクトが設計工程でやるべき全てのレビューとテストをやりきっているか確認できていませんでした。

#### 3. やりきり確認の仕組み作り

上記の『プロジェクトのやりきりの監査』構築のために、当初は監査担当者が直接エビデンスの確認を行うつもりでした。しかし、エビデンスが入っているフォルダーを監査担当者が見ても、現在のプロジェクトの進捗に対し、全てエビデンスが格納されているのか判断できませんでした。

そこで、先ずプロジェクト自身で『やりきり』を『見える化』し、その姿を確認出来るようにしました。これによって、(1)今までは、担当者のレビューなどの工程スキップにPLが気付かなかった、または気付いても放置していたことによる不具合の流出への歯止めになると期待しています。また、(2)エビデンスチェックをインタビューの前に行っておくことにより、忙しいPL達とのインタビューの時間を短く出来ました。【図 1】『やりきり』に関しては後述しします。

上記をまとめますと下記のような方法になります。  
『「プロジェクトの開発者が工程を遵守していること」を  
「PLが確認(検図)していること」を  
「監査部が監査する」』

また、監査部門として、開発プロジェクトへお願いした新たな品質目標は、以下の2点です。

- (1)『工程不遵守による不具合ゼロ！』
- (2)『工程未定義による不具合ゼロ！』

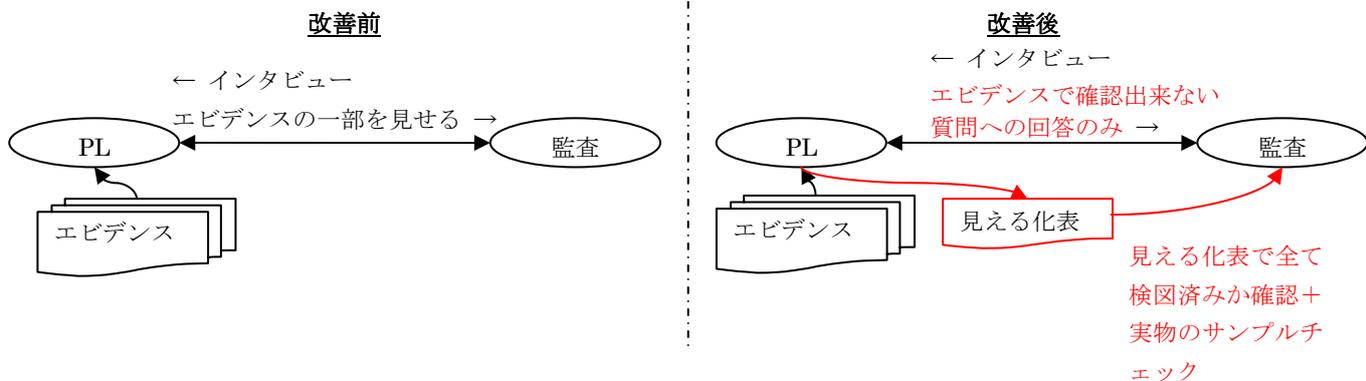


図1 完成度見える化表を使用した、やりきり度を確認する監査の仕組み

以下で、『やりきり』と『見える化』の具体例を示します。

『やりきり』とは

行動としてまず、例えば、全て(10件)の顧客要件に対して、それぞれ(10件)テスト結果があるということです。設計工程の例で言えば『要件分析ー設計ーコーディングー単体テストー結合テスト』の工程があり、それぞれの工程にレビューがあり、テストにはテスト報告書があります。これらの全てのレビューとテストが行われていること。これが『やりきり』の第1ステップです。当たり前『愚直』に行うことですね。

『見える化』とは、

各顧客要件に対して設計工程での作業成果物(エビデンス)が全てあることを確認したことを『見える化』します。【図2】の例では、3件の顧客要件に対し、PLは各設計工程でのレビューとテストの作業成果物を検閲した後に●をつけます。

『見える化』のポイントは、ステータスの良し悪しが一目でわかることです。【図2】の例では、顧客仕様 No.0002の『結合テスト RM(レビュー議事録)』の欄が空白(検閲されていない and/or エビデンスがない)になっているのが一目でわかります。

顧客要件 No.	要件分析結果 RM	設計 RM	コード RM	単体テストケース RM	単体テスト報告書	結合テストケース RM	結合テスト報告書
0001	●	●	●	●	●	●	●
0002	●	●	●	●	●		●
0003	●	●	●	●	●	●	●

RM: レビュー議事録 (Review Minutes)

PLは各作業成果物の中身を検閲してから●を付ける

図2 工程やりきりの見える化表の例

監査部署では、PLへ空白の理由を確認後、不適合項目として監査報告します。PLは報告に対して原因に対する是正処置をとります。

上記『見える化』例で考慮すべきことは 5S(整理、整頓、清潔、清掃、躰)の精神だと思います。いつもきれい(全て●)になっていないと、空白があっても気にしなくなってしまうからです。

#### 4. レビュー自体の『やりきり(品質)』は？

これはきりの無い質問です。しかし**まずは** RM(レビュー議事録)を見て、「(1)レビューアは間違いを指摘できるレベルの経験者である\*1 (2)添付されているレビューチェックリスト(ノウハウとしての観点集)を全て確認(✓)している。(3)レビュー時間がレビュー対象の量に対して妥当である。\*2 (4)全ての指摘事項に対し対策が打たれている。」

上記4点でレビューの『やりきり』を確認します(上記4点が検図の観点であり、レビューの『良品条件』です)。

\*1:『指摘できるレベルの経験者』の判断は現状、検図者である PL に任せています。勿論「対象領域での経験年数3年以上」など、定義できなくは無いです、運用レベルに達していません。

\*2:レビューの時間も検図者である PL の経験値で判断しています。

#### 5. 今後

対策中で、まだまだ改善の余地があります。大きなポイントは以下の2点です。

- 当事者のリーダー自身がどれだけ**真剣**に確認しているか  
プロジェクトの当事者ではない監査担当は、所詮エビデンスチェックで重箱の隅を突つことしか出来ません。どれだけ、当事者のリーダー自身が**真剣**に確認しているかが最終品質に影響することには変わりはありません。この活動のキックオフに際し、「監査の仕組みづくりは、開発者のプロセスへの理解と意識の向上とを並行して行う」と宣言して活動しています。
- 監査担当者のスキル確保  
エビデンスへのアクセスを確保し、いざとなったら監査担当者もエビデンスの中身の良し悪しを確認出来る技術レベルの確保が必要です。運用後の監査担当者の人選と教育計画も必要です。

#### 6. 最後に

『やりきり』という言葉は、無限∞なイメージがあります。きっと決まったゴールは無く、自分で限界を決めてしまっ  
はいけないという心があるのかもしれませんが。また『やりきり』とは、行動(努力)を示す一方、『やりきった』ことの判断は結果ですものかもしれません。今回の対策例に対し、これで十分『やりきっている』のかという問いは、『工程不遵守による不具合ゼロ!』を維持できていますかという問いに代わるのかもしれませんが。ですから逆説的ですが、手段としての『見える化』だけをやり過ぎることにより工数ばかり掛かって結果の出ない『見せる化』にならないよう注意が必要だと感じています。

##### プロフィール

玉井 治久

ポッシュ株式会社 パワートレイン ECU 事業室

自動車部品メーカー勤務

SEPG、ソフトウェアプロジェクト監査体制構築プロジェクト

高品質ソフトウェア技術交流会(QuaSTom)会員

ソフトウェア開発及びマネジメントの後8年前からプロセス改善を推進してきました。今回監査の強化の担当となりました。監査やその上位にあるソフトウェア品質保証体系についても勉強が必要です。交流会等で色々な方々と意見交換をさせて頂きたいと考えています。

## 2. 人材育成

### ■ トップエスイー:ソフトウェア開発におけるトップレベル技術者の育成

早稲田大学／国立情報学研究所 鷲崎 弘宣  
国立情報学研究所 田口 研治  
国立情報学研究所 吉岡 信和  
三菱総合研究所／国立情報学研究所 桑野 文洋  
電気通信大学 田原 康之  
国立情報学研究所／東京大学 本位田 真一

#### 1. トップエスイーとは

社会で求められるソフトウェアは、量・質・種類の全てについて増大の一途にあります[1]。一方で、品質と納期に対する要求は厳しさを増しています。このような厳しい状況において、大規模・高品質・多品種・高生産を確実に追求するためには、理論や経験に裏づけされたソフトウェア工学技術の体系を適切に用いることが不可欠です。しかしソフトウェア工学の分野では、「産業—アカデミア・ギャップ」[2]の存在が指摘されています。アカデミアには良い技術や方法論が多数ありますが、その活用のを欠き、実践性を追及できていません。一方、産業界ではときにソフトウェア工学の習得と導入が欠けています。本稿ではこのギャップを克服する方策の1つとして、国立情報学研究所が実施している産学連携に基づいた人材育成プログラム「トップエスイー」(TopSE: Top Software Engineers <http://topse.jp/>)[3][4][5]をご紹介します。

国立情報学研究所は2004年度から2008年度まで、文部科学省/科学技術振興調整費の支援を受け、トップレベルのソフトウェア技術者の育成を目指した「トップエスイー:サイエンスによる知的ものづくり教育」を実施してきました。これまでに、1期生6名(12名が2006年度に修了)、2期生25名(企業から18名と大学院生7名)、3期生31名(企業から24名と大学院生7名)が受講しています。2009年度からは、より内容を充実させて新しいトップエスイーの教育プログラムが実施される予定です。

#### 2. トップエスイーの教育目標と体制

トップエスイーの教育目標は、受講生が「ソフトウェアシステムの背後にある本質を把握し、モデルとして具体的に表現し、理論的基盤に基づいて体系的に分析・洗練化することで、高品質なソフトウェアシステムを高効率に開発するスキル」を習得することです。

その実現のために、産業界、大学、国立研究所が密に連携し、ネットワーク家電を主な対象ドメインとして教材開発と教育、普及に努めています。具体的には、講座ごとに大学関係者と企業とで作業部会を設立し、約1年かけて教材を開発しています。特に教材の題材として、企業からは、数年先に顕在化することが予想される実問題を持ち込んでもらっています。作業部会では、様々な最先端のツールを実問題に適用し、試行錯誤を繰り返しながら、有用なツールを見極めます。ツールを実問題に適用する際には、様々なノウハウが必要となりますので、それらを顕在化し教材としてまとめあげています。

例えば講座の1つである「設計モデル検証(基礎編)」では、HDレコーダ・DVDレコーダ向けのソフトウェアの設計と組込ボード上での実装を実問題として設定し、実問題に対してモデル検査ツールの活用により設計誤りの検出・修正作業を行うことで、モデル検査技術の有効性と難しさを体得させるものとなっています。

教育を実施する講師陣として、企業側から10名以上の技術者・研究者が参画し、大学側からも国立情報学研究所をはじめとして多数の大学・研究所より教員・研究者が参画し、総勢で約30名に及びます。

受講生としては、情報系大学院相当の計算機科学の知識を有していることを前提として、企業の若手エンジニアを中心に毎年20~30名ほどが、出願書類審査、筆記試験ならびに口頭試問を経て受講が認められ、1~2年間をかけて受講しています。さらに、全国の大学や企業においても同等な教育を実施できるように、普及のための教材作成も合わせて実施しています。

#### 3. ソフトウェア技術者が持つべきスキル

ソフトウェア開発において、技術者が有すべきスキルは多岐に渡ります。プログラマならばプログラミングスキル、アーキテクトならばアーキテクチャの設計スキル、プロジェクトマネージャならばプロジェクト管理スキルといった具合です。しかしそれぞれの職種におけるトップレベルの技術者育成を考えると、職種ごとに特化したスキルを修得する上での土台となる共通なスキルを識別することができます。大学院相当レベルの教育において、特定の職種に特化したスキル教育の準備と実施は有効とはいえないため、トップエスイーでは共通スキルの教育に注力しています。

その共通スキルは、実際のソフトウェア開発に着目して識別します。ソフトウェア開発プロセスを、システム要求分析・方式設計、ソフトウェア要求分析、方式設計、詳細設計、実装、テストという流れとして捉えるとしましょう。ここで、各フェーズでは中間成果物として何らかのモデルが構築されるため、ソフトウェア開発プロセスとは本質的には多段階のモデル変換工程と捉えることができます。その流れを図1に示します。各フェーズで構築される要求モデル、分析モデル、設計モデルといったモデル間の変換を経てソフトウェアシステムが出来上がるということです。

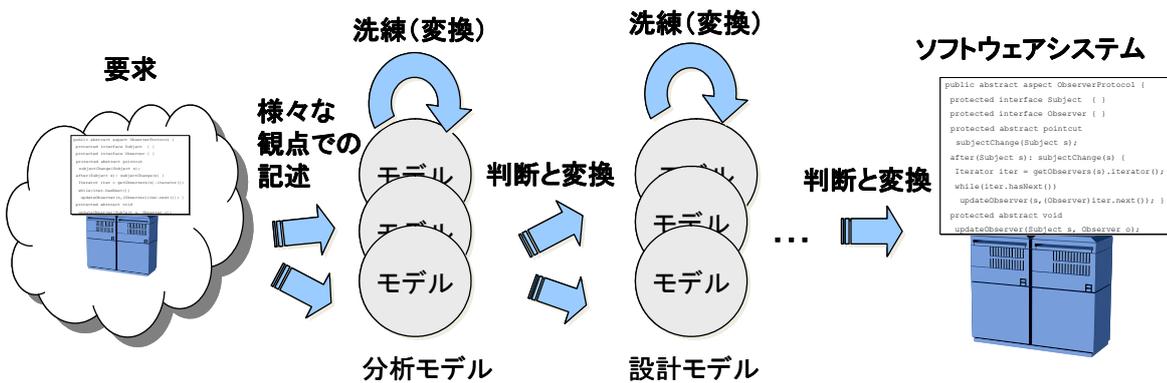


図1 多段階モデル変換としての開発プロセス

マネージャやアーキテクト、プログラマは、それぞれの立場でこれらのモデルに関わることになります。このように、ソフトウェア開発に関わる様々な職種において、モデリング能力を共通スキルの一つとして捉えることができます。ここでモデリング能力とは、良いモデルの構築・分析・検証・操作の能力を意味します。トップエスイーでは、その良いモデルを、「抽象化により事柄の本質を表わしている、最小かつ完備である、解釈の一意性が実現されているという性質を有した表現」と定義しています。

モデリングとは、狭義にはそのようなモデルを作成する行為であり、「表現のための道具の特性を踏まえた手順(モデリングプロセス)に従って対象を道具で表現していく」行為を意味しています。そのうえでトップエスイーでは、下記を備えることをモデリング能力として期待しています。

- ・抽象化や具象化の操作を含むモデリングプロセスが身に付いている
- ・厳格にモデル構築できる
- ・モデル間の関連性を把握できる
- ・保守性、再利用性、非機能性を踏まえたモデル構築できる

ソフトウェアシステムには、様々な視点での切り口が存在し、その切り口に基づいたモデリングの道具も多種多様です。さらに、それぞれのモデリングの道具には固有のモデリングプロセスがあります。トップエスイーでは、モデリングプロセスを問題解決と位置付けて、様々なモデリングの道具とプロセスを習得することが、高度なモデリング能力を身につける際の有効な手法と考えています。

ここで、最先端のモデリングツール(道具)の多くは、計算機科学に基づいています。例えばモデル検査ツールである SPIN [6][7]の習得には、前提知識としてオートマトン、時相論理、抽象解釈などが必要です。トップエスイーでは、形式的な意味論を背景として持ち、その意味論を元に開発されている約 20 種類のツールを導入し、これらのソフトウェアツールをシャワーの如く、繰り返し学習させることとしています。その学習の過程で、計算機科学に関する知識を深耕させることができます。さらに、ツールを実問題に適用する過程で、モデリング能力が洗練化されることとなります。このプロセスを繰り返すことで、モデリング能力の深耕と計算機科学の知識の深耕の両者を

同時に達成でき、その結果としてソフトウェアツールを習得しつつ新たな問題への応用力も身につくこととなります。

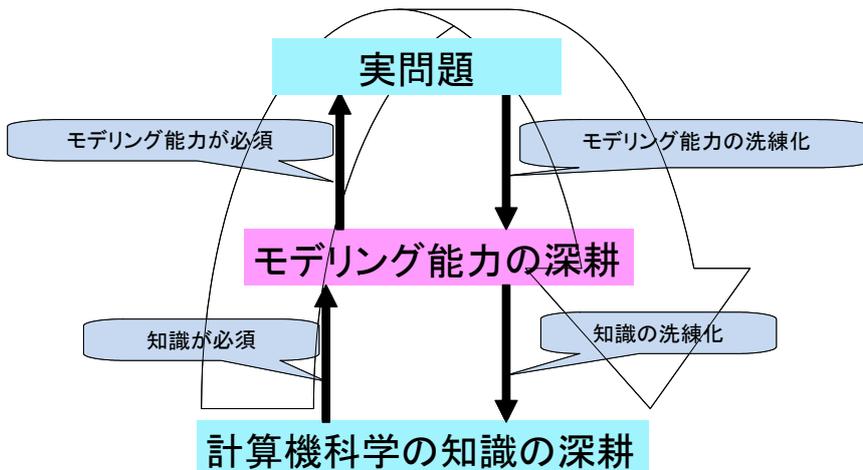


図2 ソフトウェアツール教育の効果

#### 4. トップエスイーのカリキュラム

トップエスイーでは、将来のエンタープライズソフトウェアシステム開発および組込みソフトウェアシステム開発において、特に信頼性、効率性、変更容易性、セキュリティの達成が重要であると捉えています。そこで、それらの各品質特性の達成に有効であり、かつ、計算機科学に基づく評価の高い複数のソフトウェアツールが存在していることを講座開発の前提としました。その結果、形式手法関係の講座が半分近く占めることになりました。トップエスイーは必ずしも形式手法の教育を標榜していませんが、形式手法が有する厳格なモデリングプロセスを学ぶことで、表現上は自然言語やUMLを用いるとしても「厳格にモデリングする」という癖を身につけさせることを意図しています。また、複数のツールを扱うことで、1つのツールに依存せず、対象分野の基本的な概念を習得させて新たなツールへの適用能力を身に付けさせることができます。

2009年度に開講する講座は、下記のとおりです。各講座の詳細は <http://topse.jp/> よりご参照ください。

##### ● 基礎系

・基礎理論: 受講生が他の講座を受講し理解するために必要な計算機科学に関する知識の総復習

##### ● 要求分析系

・ゴール指向要求分析: 個別の要求の背後にある上位目標といった多様な抽象度から要求を分析する技術の習得

・要求獲得・定義: 要求を獲得するための基本的な技術と、それをモデル化する技術、さらに、獲得された知識や要求を仕様化し、要求仕様書にまとめあげ、評価するための技術の習得

・セキュリティ要求分析: システムの脆弱性、利用者の悪意、システムへの攻撃といったセキュリティ上の問題を予測・発見し克服する手法、および、演習を通して実際の要求モデルをセキュリティという視点から分析する方法の習得

・超上流要求工学: 問題を解決するための論理的な思考体系や発想法、および、多様性と変動性を視野に入れた情報システムを構築するための前提としてのビジネスモデリング法の習得

##### ● 形式仕様系

・形式仕様記述(基礎編): 形式仕様言語に基づく段階的な詳細化と呼ぶ技術と、仕様を直接実行する技術の2つの方向から、妥当性の確認を行う方法の習得

・形式仕様記述(応用編): 形式仕様記述(基礎編)に引き続いて静的な情報構造にまとを絞り、上流工程からインプリメンテーションまでをつなぐ形式手法を用いた系統的な開発の方法の習得

・形式仕様記述(セキュリティ編): 形式仕様言語を使ったポリシ・モデルの理解、および、段階的な詳細化、定理証

明、モデル検査というシステム検証技術における主要な三技術を使ったセキュア・システムの開発技術の習得  
 ・並行システムのモデル化と検証：信頼性の高い並行システムを構築することを目的としたモデル化、検証、実装方法の習得

● モデル検査系

- ・設計モデル検証(基礎編)：ネットワーク家電の制御ソフトウェアを題材とした産業ソフトウェアの設計モデル検証のプロセスとノウハウの習得
- ・設計モデル検証(応用編)：設計モデルの特徴に合わせたモデル検査ツールの使い分けおよびノウハウの習得
- ・実装モデル検証：ネットワークを介したメッセージ通信を通して動作する分散ソフトウェアの振る舞いを網羅的に検証する技術、および計算機上で扱える程度の規模に検証問題を変換する抽象化技術の習得
- ・性能モデル検証：設計モデルの性能面に関する動作の正しさを自動的に検証する性能モデル検査ツールを使用し、実際のシステム開発に適用する方法を習得
- ・テストング：組込みシステムからネットワークまで対応可能な、ソフトウェア・テストングの技術の習得
- ・プログラム解析：形式仕様記述に基づくプログラムの静的検証や単体テストの手法の習得

● アーキテクチャ系

- ・コンポーネントベース開発：早期の分割統治とインタフェース中心の分析設計を系統立てて実施するためのコンポーネントベース開発手法とプロセスの習得
- ・ソフトウェアパターン：情報機器系のシステム開発において代表的な既存ソフトウェアパターンの適用ノウハウ、設計原則や非機能的特性からのパターンの適用範囲と取捨選択、支援環境の活用ノウハウの習得
- ・アスペクト指向開発：アスペクト指向に基づく関心事の分離と合成の技術、アスペクト指向プログラミングおよび分析設計の実践的ノウハウの習得

● マネジメント系

- ・ソフトウェアメトリクス：ソフトウェア開発に「数字による管理」を取り入れ、品質の定量的評価、品質予測、手法の有効性評価、プロジェクトの状況判断といった局面で定量的な観点から議論するための基礎知識とスキルの習得

トップエスイーでは、スキルレベルとして4段階を識別し、各講座について表1に示すように該当レベルを明示しています(ただし表1は抜粋)。受講生は、1講座毎に1単位を取得し、修了制作に合格してレベル4を達成しカリキュラムを修了するためには、4単位以上を取得する必要があります。カリキュラム修了には最低1年間が必要です。成績は、頻繁に講義で出題される演習問題に対するレポートを主に考慮して評価しています。各講座は、2009年度は毎週2コマ連続で開講される15コマの講義から構成され、1コマの時間は90分です。講座は2ヶ月続き、これで1学期を構成します。1学年は5学期です。

表1 講座とスキルレベルの対応(抜粋)

技術スキルレベル		要求分析	設計	実装
4 (トップエスイーのゴールレベル)	現場の開発に近い規模・固さのシステム開発課題に対し、独力で最先端のモデリング技術・ツールを複数組み合わせることで開発業務を行い、高い品質の成果を出すとともに、そのノウハウを方法論として一般化したり、新たなツールを開発できる。			修了制作
3	開発工程の各フェーズにおけるモデリング技術やツールに関して、従来のものから最先端のものまで、複数の技術・ツールを理解・習得し、指導者が与えた課題に対し、それらの技術・ツールを使い分けることができる。	セキュリティ要求分析	形式的仕様(セキュリティ)   性能モデル検証   実装モデル検証   並行システムのモデル化と検証 形式的仕様(応用編) 設計モデル検証(応用編) ソフトウェアパターン アスペクト指向開発	
2	従来のモデリング技術やツールの原理の基礎を理解し、指導者が与えた開発工程の各フェーズにおける例題に対し、これらの技術を適用することができる。	要求分析	形式的仕様(基礎編) 設計モデル検証(基礎編) コンポーネント開発	
1 (トップエスイーのエントリーレベル)	ソフトウェアエンジニアリングの原理・原則および最先端のモデリング技術やツールの原理を理解する基礎知識とソフトウェア開発経験を持つ。			基礎理論

表 2 に、各講座で使用している方法論とツールの一例を示します。受講生が、カリキュラムを修了するためには、8 講座の習得により、約 10-20 点の異なるツールが有するモデリング技術の習熟が求められることとなります。

表 2 利用ツール(一例)

講座名	方法論	開発支援ツール
コンポーネントベース開発	ICONIX, KorbA, UML Component, Catalysis	J2SE/JavaBeans, Jude
形式仕様記述(基礎編)	B-Method, VDM-SL	VDM-SL Toolbox, B4Free, Click' n' Prove
設計モデル検証	設計の検査プロセスと検証方法	SPIN, SMV, LTSA
要求分析	KAOS, i*	K-Tool(自家製ツール), ST-Tool
ソフトウェアパターン	アナリシスパターン、アーキテクチャパターン、デザインパターン、オブジェクト指向設計原則	Jude, PatternWeaver, SMC, DoJA, Eclipse
形式仕様記述(応用編)	B-Method, VDM++	VDM++ Toolbox, B4Free, Click' n' Prove
実装モデル検証	プログラムの欠陥の発見的方法論	JPF (Java Path Finder), Centralizer (自家製ツール)
性能モデル検証	設計の検査プロセスと検証方法	UML Profile for SPT, Uppaal
セキュリティ要求分析	KAOS, i*, Misusecase, AbuseCase, Security Usecase	K-Tool(自家製ツール), ST-Tool, Jude

## 5. 修了生の実績と教育効果

トップエスイーでは、受講生の修了を認定するために修了制作を課しています。修了制作とは、受講生が教員の指導を受けながら主体的に約 2 ヶ月間をかけて、職場で抱える実問題あるいは実問題に近い課題を設定し、トップエスイーで習得した 2 種類以上のモデリング技術を駆使して課題を解決し、その成果および過程において得られた知見を報告書にまとめて発表するという一連の実習活動です。

トップエスイーにおいて受講生に求められる最終目標は、表 1 における技術スキルレベル 4 に到達することであり、その到達の可否は、修了制作の発表内容を教員が評価して判定されます。従って表 1 に示すように、高品質な成果を生み出すにとどまらず、修了後の受講生自身や所属組織および広く一般における開発・技術活動について習得モデリング技術を活用可能となることを念頭において、(A) 解決の過程で得られたモデリング上の知見を再利用可能な方法論として一般化することや、(B) 一定の汎用性を備えてモデリング作業を支援する新ツールを生み出すことが求められます。さらに 2 種類以上の技術使用を義務付けることで、状況に応じた技術の使い分けや、工程・側面を超えた複数技術の組み合わせといった実践上のノウハウが習得されていることを確認しています。ただし、(B)を主目的とする場合は、作業負荷を考慮して 1 種類のみ技術使用も可としました。

修了制作の具体例として、平成 18 年度に修了した受講生の事例を以下に 2 つ取り上げます。

(1)「UTM アプライアンスのログ情報解析ツールの解析」: セキュリティ機能を統合した UTM(Unified Threat Management)製品の開発にあたり、同製品が出力するログの解析ツールを合わせて開発する必要がありました。開発にあたり受講生は、機能要求の不明確さや時間効率性・拡張性に関する品質要求などを問題として識別しました。

そこで受講生は、トップエスイーで習得した技術群から、要求分析において KAOS によるゴール指向分析、設計において品質特性駆動型設計を選択・併用し、問題を解決する形でツールを実装しました。前者は多種の目標群を達成する観点に基づいて要求を詳細に識別・分析する手法であり、後者は各種パターン(定石や手本)を再利用して品質要求を満足するアーキテクチャを導出する手法です。

さらに、これらの技術の組み合わせ方法には、これまで明らかでなかった実用上のノウハウが含まれていたため、受講生は全体を品質要求駆動型プロセスとして一般化しました。これは、開発過程で得られた知見を方法論としてまとめるという(A)の好事例といえます。得られた解析ツールの実用化が検討されており、高品質であることから、背景理論と実用上のノウハウを共に重視して複数のモデリング技術を習得させるトップエスイーの高い教育効果が結実した形と捉えられます。

(2)「Web アプリケーションのページ間遷移仕様検証ツール」: Web アプリケーション開発の問題として、大規模・複雑な Web ページ遷移の信頼性や安全性を検証することの難しさがあります。例えば、個人情報ページに到達する前に認証ページを必ず辿っていることや、行き止まりのページが存在しないことを人手で確認しようとすると膨大な時間がかかり、さらには見落としの危険性もあります。

このような状態遷移上の性質の検証には、有限の状態空間を網羅的・自動的に探索するモデル検査が有効です。しかし、対象アプリケーションの振る舞いを有限状態モデルとしてモデリングするにあたって、必要な遷移情報が各ページ定義やフレームワークの構成定義に分散し、結果として適用コストが大きく誤りも混入しやすいという問題を受講生は識別しました。

そこで受講生は、対象フレームワークを Struts に設定し、ページ定義および構成定義から遷移情報を抽出し、モデル検査ツール SPIN が扱う有限状態モデルへと自動変換する実用的ツール `struts2spin` を開発しました。つまり同ツールは、モデル検査を Web アプリケーションという実問題へ適用する際に必要なモデリング上の新ノウハウと手順をカプセル化し、誰でも一定品質のモデリング結果を得られるように自動化します。受講生は実際の Web アプリケーションに同ツールを適用し、識別した問題を解決しつつ、遷移上の信頼性・安全性検証に有効なことを確認しました。これは汎用性を備えた新ツールを生み出すという(B)の好事例といえます。同成果には学术界より高い評価が与えられており、国際会議に論文が採択されたことから[8]、モデリング能力の洗練化を目指すトップエスイーの高い教育効果が結実した形と捉えられます。

## 6. 今後の展開

2008 年度現在のトップエスイーの 3 期生の数は 31 名です。30 名程度が、一つの教室で良質な教育ができる上限値と考えられます。そこでトップエスイーでは、同等の教育を広く大学や企業に普及させて多数の優秀なソフトウェア技術者育成に貢献するために、教材の洗練と普及を進めていく方針です。その一環として、各講座の講義ノートサイエンスによる知的ものづくりシリーズとして順次刊行しています[9]。また、文科省の先導的 IT スペシャリスト育成推進プログラムなど他の教育プログラムとも積極的に連携を進めています。たとえば、東大、東工大の学生を対象にした「情報理工実践プログラム」においてトップエスイーの 6 教材を使用しています。また、2009 年度は講座の一部について北陸先端科学技術大学院大学が開設する社会人向け博士課程「先端ソフトウェア工学コース」の講座と共通化される予定です。

トップエスイーでは 2008 年度 11 月現在、2009 年度 4 月に始まる 4 期生が募集されています。ご興味を持たれましたらぜひ <http://topse.jp/> をご参照ください。個々の講義シラバスや協賛企業名などが掲載されています。

### 謝辞

2004 年度から 2008 年度までの本人材養成ユニットは文部科学省科学技術振興調整費新興分野人材養成の支援により運営されていることをここに記すとともに感謝いたします。また、本記事は文献[4][5]に基づいて、一部改変する形で執筆されました。

### 参考文献

- [1] 鷲崎弘宜, “情報学探求 - 大規模ソフトウェアの効率的開発技術の追求”, 情報通信ジャーナル 5 月号, 2007.
- [2] K. Beckman, N. Coulter, S. Khajenoori, and N. R. Mead, “Collaborations: Closing the industry-academia gap,” IEEE Software, Vol.14, No.6, pp.49-57, 1997.
- [3] Shinichi Honiden, Yasuyuki Tahara, Nobukazu Yoshioka, Kenji Taguchi, Hironori Washizaki, “Top SE: Education Program of Japan to Produce Superarchitects Who Can Apply Software Engineering Tools to Practical Development”, Proceedings of the 29th International Conference on Software Engineering (ICSE 2007), 20-26 May 2007.
- [4] 本位田真一、田口研治、吉岡信和、田原康之、鷲崎弘宜, “トップエスイー「サイエンスによる知的ものづくり教育」”, 映像情報メディア学会誌, Vol. 61, No. 9, 2007
- [5] 本位田真一、桑野文洋、田原康之、鷲崎弘宜, “トップエスイー: サイエンスによる知的ものづくり教育”, 情報処理, Vol.48, No.11, pp1264-1272, 2007.
- [6] G.J. Holzmann, “The SPIN Model Checker: Primer and Reference Manual”, Addison Wesley, 2004.
- [7] 萩谷昌己 監修, 吉岡信和, 青木利晃, 田原康之 著, “SPIN による設計モデル検証”, 近代科学社, 2008.
- [8] Atsuto Kubo, Hironori Washizaki, Yoshiaki Fukazawa, “Automatic Extraction and Verification of Page

Transitions in a Web Application”, Proceedings of the 14th Asia-Pacific Software Engineering Conference (APSEC 2007), IEEE Computer Society, pp.350-357, 2007.

[9] 本位田真一監修:サイエンスによる知的ものづくりシリーズ、近代科学社

## プロフィール

### 鷺崎 弘宜

早稲田大学理工学術院准教授および国立情報学研究所客員准教授。2003年早稲田大学大学院博士課程修了、博士(情報科学)。早稲田大学助手、国立情報学研究所助手・助教を経て現職。アーキテクチャ、再利用、品質保証を中心として、ソフトウェア工学の研究と教育に従事。他の活動に日科技連 SQiP 研究会運営小委員会副委員長、同研究会演習コース主査、情報処理学会論文誌編集委員、同学会ソフトウェア工学研究会幹事など。2004年日本ソフトウェア科学会高橋奨励賞、2006年情報処理学会 SES 優秀論文賞、2007年同学会山下記念研究賞、2008年船井情報科学奨励賞。

### 田口 研治

国立情報学研究所アーキテクチャ科学研究系特任教授。2001年 Uppsala 大学にて PhD in Computer Science 取得。主として、プロセス代数と状態ベース形式仕様言語のように異なる形式手法の統合の研究に従事。1999年国際会議シリーズ IFM (Integrated Formal Methods)を共同設立。2005年より国立情報学研究所に勤務し、教育プログラム・トップエスイーに従事。

### 吉岡 信和

1971年生。1998年北陸先端科学技術大学院大学情報科学研究科博士後期課程修了。博士(情報科学)。同年(株)東芝入社。2002年より国立情報学研究所に勤務、2004年より同研究所。現在、同研究所准教授。エージェント技術の研究、ソフトウェア工学の研究に従事。現在に至る。日本ソフトウェア科学会、情報処理学会、各会員。

### 糸野 文洋

1990年早稲田大学大学院理工学研究科数学専攻修士課程修了。同年(株)三菱総合研究所に入社、現在に至る。自動推論、エージェント指向言語、移動エージェントの研究開発に従事。2007年より国立情報学研究所特任准教授、現在に至る。工学博士(早稲田大学)

### 田原 康之

1991年東京大学大学院理学系研究科数学専攻修士課程修了。同年(株)東芝入社。1993~1996年情報処理振興事業協会に出向。1996~1997年英国 City 大学客員研究員。1997~1998年英国 Imperial College 客員研究員。2003年より国立情報学研究所に勤務、2007年同研究所特任准教授。2008年より電気通信大学准教授。博士(情報科学)(早稲田大学)。エージェント技術、およびソフトウェア工学などの研究に従事。日本ソフトウェア科学会会員。

### 本位田 真一

1978年早稲田大学大学院理工学研究科修士課程修了。(株)東芝を経て2000年より国立情報学研究所教授、2004年より同研究所アーキテクチャ科学研究系研究主幹を併任、現在に至る。2001年より東京大学大学院情報理工学系研究科教授を兼任、現在に至る。早稲田大学客員教授、英国UCL客員教授、現在に至る。2005年度パリ第6大学招聘教授。工学博士(早稲田大学)。1986年度情報処理学会論文賞。DICOMO2007 最優秀論文賞。日本ソフトウェア科学会理事、情報処理学会理事(教育担当)を歴任。IEEE Computer Society Japan Chapter Chair、ACM 日本支部会計幹事。日本学術会議連携会員。情報処理学会フェロー。

## 3. SQuBOK®

### ■ ソフトウェア品質知識体系ガイドーSQuBOK® Guide について 第 2 回

SQuBOK® 編集チーム

株式会社 NTT データ 町田 欣史

富士通株式会社 辰巳 敬三

日立情報通信エンジニアリング株式会社 池田 暁

#### はじめに

QualityOne をご覧になっている方々はすでにご存じかもしれませんが、本連載で取り上げている SQuBOK®ガイドが、2008 年度の日経品質管理文献賞を受賞しました。このような荣誉ある賞をいただき、策定部会メンバも、よりいっそう気を引き締めて取り組んでいく決意を新たにしております。今後とも皆様のご指導ご鞭撻をいただければ幸甚でございます。

(参考: 2008 年度 デミング賞本賞・実施賞・日経品質管理文献賞 受賞者一覧

[http://www.juse.or.jp/prize/deming\\_12\\_y2008.html](http://www.juse.or.jp/prize/deming_12_y2008.html))

さて、第 1 回では SQuBOK®および SQuBOK®ガイドがどういうものであるかを解説しました。今回からは、いよいよ実際の SQuBOK®ガイドの中身に入っていきたいと思えます。

第 2 回は、SQuBOK®ガイドの 1 つめのカテゴリである「ソフトウェア品質の基本概念」について解説します。と言っても、カテゴリ内のすべてをとりあげるわけにもいきませんので、読み進める上で、意識すべきことや重要な部分を中心に解説します。

#### 「ソフトウェア品質の基本概念」カテゴリの概要

このカテゴリは、基本概念というだけあって、「品質」「品質管理」といった、普段何気なく使っている用語の定義や考え方について整理をしています。

そもそも、「品質」という言葉自体が漠然としたものです。皆さんも普段から「このシステムの品質はボロボロだ」とか「ちゃんと品質を満たしているのか」などという会話をするところがあるでしょう。こんなとき、その「品質」という言葉が何を指しているのか、会話の相手と認識が合っているでしょうか。この意識が組織でブレてしまうと、目指すべき品質のベクトルが決まらないどころか、気がついてみたら全く別のところを目指していたという事態にもなりかねません。また、顧客やユーザとの間でも、求められる品質について早い段階で意識合わせをしておくことが非常に大切です。

まずは、このカテゴリに書かれた品質の概念をしっかりと理解し、それを踏まえた上で共通の認識をもつこと(決めていくこと)が重要となります。

#### 「品質の定義」が 10 個もある！

さて、「ソフトウェア品質の基本概念」カテゴリの樹形図を見てみると、「品質の定義」というトピックスがなんと 10 個もあります。先ほど、「品質」の認識を合わせましょうという話をしましたが、これでは、「じゃあどの定義に従えばいいの」と逆に混乱してしまうかもしれません。

SQuBOK®ガイドは、読者を混乱させようとして 10 個もの定義を載せているわけではありませんし、どれかの定義を選ばなければいけないということでもありません。

では何故それだけの定義を載せているのでしょうか？

その答えの一つは「様々な視点により品質の考え方は異なる」ということです。また、言葉は時代によって変わると言われますが、品質についても、その時代の社会環境や技術の進展にもなって考え方や概念は変わります。

先人たちが考え、そして実践してきた定義は皆さんにとってきっと品質に関するベースを作ってくれることでしょう。

そして、これらの定義を皆さんの組織で、かみ砕いて、よりわかりやすく、より具体的に表現することで、品質に対する認識の共有を図ることができるでしょう。

これら 10 個の品質の定義をよく読んでみると、共通して言えるのは「顧客やユーザの要求・ニーズを満たしているかどうか」であると言えます。ソフトウェア開発に携わる人は、どうしても作り手の視点になってしまいがちです。使う側の視点を常に意識することで、品質の高いソフトウェアを開発することができるでしょう。

## 「品質」をもっと具体的に(品質特性)

さて、品質の定義に共通した概念はユーザのニーズに対する満足度だということが見えてきましたが、一口にユーザのニーズといっても様々なものがあります。また、「品質の定義」知識領域の解説には次のように書かれています。『例えば、1950 年代は正しく動くことがユーザにとって重要であったが、その後、信頼性や処理時間に焦点が移り、1980 年代には信頼性は当たり前品質と言われてユーザビリティに目が向けられた。そして現在はセキュリティがユーザの大きな関心事の一つになっている。』このように、求められる品質は時代とともに変化していることが伺えます。

この「品質」を、より具体的な「品質特性」として表したものにソフトウェア品質モデルがあります。有名なものに ISO/IEC 9126 シリーズがあり、SQuBOK<sup>®</sup>でもトピックスとして取り上げています。このモデルではソフトウェアの品質を「内部品質」「外部品質」「利用時の品質」の 3 つに分け、開発の途中段階、完成段階、および利用状況下での品質評価に対応させています。また、外部及び内部品質を「機能性」「信頼性」「使用性」「効率性」「保守性」「移植性」の 6 つの品質特性、利用時の品質を「有効性」「生産性」「安全性」「満足性」という 4 つの品質特性で表しています。

このように「〇〇性」という言葉まで具体化されると、品質というものがよりイメージしやすくなったのではないのでしょうか。顧客やユーザによって、どの品質特性を重視するかは異なることが多いものです。これらの具体化された品質特性を基に、顧客と合意を得ることで、より高い品質、すなわち顧客満足度の向上につながるでしょう。

## マネジメントとコントロールの違いが分かりますか？

SQuBOK<sup>®</sup>ガイドでは「品質マネジメント」「品質コントロール」という用語が使われています。なぜ「品質管理」と言わないのだろうかと思議に思った人がいるかもしれませんね。

日本では、control と management の訳語としてどちらも「管理」が使われてきましたが、別の単語であることから明らかのように欧米では異なる概念です。SQuBOK<sup>®</sup>ガイドではこの違いを明確に意識できるように「管理」とは訳さず、「マネジメント」「コントロール」のままとしています。

では、それぞれどういう意味なのでしょう。

ISO9000 では、「品質マネジメント」は『品質に関して組織を指揮し、管理するための調整された活動』と定義されています。また、「品質コントロール」は『品質要求事項を満たすことに焦点を合わせた品質マネジメントの一部』と定義され、品質マネジメントの下位の概念になっています。

## 「品質コントロール」と「品質保証」の役割

品質管理(品質コントロール)や品質保証という言葉、特に違いを意識せずに使っている方も多いのではないのでしょうか。

SQuBOK<sup>®</sup>では「ソフトウェア品質の基本概念」カテゴリの 2 つ目の知識領域「品質のマネジメント」で、それぞれ ISO9000 の定義を引用して解説を加えています。「品質コントロール」の定義は前述したとおりです。一方、「品質保証」は『品質要求事項が満たされるという確信を与えることに焦点を合わせた品質マネジメントの一部』と定義されています。

これらの定義を読むと、まず「品質マネジメント」の中に「品質コントロール」と「品質保証」があることが分かりますね。ただ、両者の違いを、品質要求事項を「満たすこと」か「満たされるという確信を与えること」という表現の違いだけで理解するのは難しいかもしれません。

「品質保証」の具体的な方法には、副知識領域にもなっている「検査」や「V&V」「ソフトウェア測定」などがあります。これらの方法が、「確信を与える」ための手段ということになります。一方、これらの方法を使って品質を保証するために、求められる品質の基準が満たされるように活動を進めていくことが「品質コントロール」と言えます。

皆さんの組織では、「品質マネジメント」つまり品質に関する一連の活動を指して「品質管理」と言っていることが多いのではないのでしょうか。それ自体は悪いことではありませんが、その中で「品質保証」と「品質コントロール」を正しく行うことが大切です。

## 「ディペンダビリティ」って？

最後に、このカテゴリの中で重要なキーワードを1つとりあげましょう。それは「ディペンダビリティ」です。同じ知識領域にある「セキュリティ」や「ユーザビリティ」はご存じの方も多いと思いますが、この「ディペンダビリティ」という言葉はなじみのない方が多いのではないのでしょうか。

「ディペンダビリティ」については副知識領域の説明が最もわかりやすいと思います。ディペンダビリティは信頼性と訳されることもありますが、同じく信頼性と訳される Reliability や Availability(可用性)、Maintainability(保守性)の上位に位置付けられ、時間的な経過や様々な利用状況の下での品質までを含めた包括的な概念です。

私たちの身の回りは、その大部分が IT 化され、ソフトウェアなしでは生活できないのが現状です。そういった中で利用しているソフトウェアには、作られてから何十年も経ってしまい、それに伴う品質問題が起こりうる可能性があることを意識する必要がでてきました。

この考えは、ソフトウェアが一部の人のためのものだった時代にはなかったもので、現代ならではの新しいものです。これも、品質の考え方が時代によって変わることの例と言えるでしょう。今後は、ソフトウェアの作り手として、長期的な利用も意識した品質保証が必要になるということに注意しましょう。

## 終わりに

このカテゴリは、「そもそも品質とは」というかなりお堅いテーマなので、理解に時間がかかるかもしれません。SQuBOK®ガイドでは、実際の定義が書かれている規格を引用しつつ、それらをよりわかりやすく解説していますので、皆さんの理解の手助けになると思います。

次回は残り 2 つのカテゴリのうち、開発者の方にとってより身近な「ソフトウェア品質技術」を取り上げる予定です。

### プロフィール

町田 欣史(SQuBOK® 編集チーム)

株式会社 NTT データ 技術開発本部 ソフトウェア工学推進センタ

JSTQB テスト技術者資格認定 技術委員会

社内にてテストプロセス改善の研究、テスト支援ツールの開発、プロジェクト支援を行う。

辰巳 敬三(SQuBOK® 編集チーム)

富士通株式会社 富士通株式会社 ソフトウェア事業本部・事業計画統括部

日科技連 SQuBOK®策定小委員会・副委員長

ライフワークとして、ソフトウェア品質、テスト技術に取り組む。

池田 暁(SQuBOK® 編集チーム)

日立情報通信エンジニアリング株式会社 技術・IT 戦略本部 ソフトウェア技術部

NPO 法人ソフトウェアテスト技術振興協会 理事

ソフトウェアテストワークショップ 実行委員長

社内では設計/テストツールの普及活動やプロセス改善業務に従事。また社外においてもコミュニティを中心に品質/テスト技術の普及に取り組む。最近主に取り組んでいるテーマはテスト設計技術で、自らも日々勉強中。

## 4. トピックス

### ■ 第4回世界ソフトウェア品質会議参加ルポ

日本アイ・ビー・エム株式会社  
岡崎 靖子

2008年9月15日から9月18日まで、アメリカ・メリーランド州ベセスダのハイアットリージェンシーホテルで第4回世界ソフトウェア品質会議(World Congress for Software Quality、以降WCSQと表記)が開催されました。その模様を報告いたします。うれしいことに、日本からの論文2件がベスト・ペーパーとして表彰されました。

#### 1. 背景

WCSQは、(財)日本科学技術連盟・ソフトウェア生産品質管理委員会(2007年9月にSQiPソフトウェア品質委員会と改名)、ヨーロッパの品質管理団体であるEOQ(European Organization for Software Quality)のソフトウェア部門、アメリカのASQ(American Society for Quality)のソフトウェア部門の3者が、アジア・アメリカ・ヨーロッパの各地域を代表し、5年ごとに各地域をもちまわりで開催しているソフトウェア品質の国際会議です。第1回は1995年6月にアメリカ・サンフランシスコのフェアモントホテルにて22カ国約300名が参加し、第2回は2000年9月にパシフィコ横浜にて29カ国720名が参加し、第3回は2005年9月にドイツ・ミュンヘンの工科大学機械学部にて30カ国約680名が参加して開催されました。第3回の時に、ソフトウェアをめぐる環境変化の速さから、今後は3年ごとに開催することとなり、本年第4回が開催されました。

#### 2. 4WCSQ 概要

初日は12件のチュートリアルがあり、2日目から4日目に論文発表が行われました。

##### (1) 開会

4WCSQ エグゼクティブディレクターのTaz Daughtrey氏(ASQ)の司会により、James Madison 大学学長のLinwood H. Rose氏が開会の挨拶を行いました。

##### (2) 基調講演

論文発表の3日間で7つの基調講演が行われました。講演者(敬称略)は、Capers Jones(アメリカ)、Larry Constantine(アメリカ)、Watts Humphery(アメリカ)、Bernie Gauf(アメリカ)、勝丸泰志(日本・富士ゼロックス)、Gary McGraw(アメリカ)、Kurt Shneider(ドイツ)です。

##### (3) 参加者

16カ国約250名が参加しました。アメリカが最多で155名、次いで日本が31名、第3位がドイツで7名。以下、カナダ、オーストリア、スイス、デンマーク、オーストラリア、中国、台湾、イタリア、ウルグアイ、ポーランド、フィンランド、イギリス、ハンガリーからの参加がありました。中国はビザの発給が遅れた関係で論文発表者・聴衆ともにほとんど全員がキャンセルしており、聴衆が1名参加しているだけでした。インドからの参加者はいませんでした。

##### (4) 論文発表

論文発表は63件で(キャンセル分は除く)、アメリカが最多の30件、次いで日本が19件、第3位がドイツ5件でした。オーストリア、イタリア、イギリス、フィンランド、スイス、ハンガリー、ポーランド、台湾、オーストラリアからも各1件ずつ発表がありました。

日本からは従来のように実務に基づいた内容が多かったですが、研究論文もいくつかありました。アメリカやヨーロッパの論文にはコンセプトの整理や提示に留まる発表がある一方で、チュートリアル講師を兼任している発表者やコンサルタントの発表者が多いためか、ソフトウェア工学の歴史や基礎をきちんと認識した上での発表も多いと感じました。4WCSQ全体としては研究論文の数は少なかったです。

発表テーマは、テスト、メトリクス・計測、プロセス改善、アジャイルという常連が多いものの、統合的品質管理、改善、ユーザ満足、教育・人、エンジニアの認定、レビュー、セキュリティ、CMMI、新成熟度モデル、品質保証組織、アウトソーシング、オフショア、マネージメント、ソフトウェア品質工学、プロセス、要求分析、フォルトのメカニズム、DSM、IT 業界への警鐘(アメリカ医療現場の実態、アメリカ大学の実態)をテーマにしたものまで、とても幅広いと感じました。個人的には、IT 業界への警鐘をテーマとした 2 件の発表が、あらためて IT と社会との深い関わりを再考するきっかけとなり、印象に残っています。

#### (5)表彰

ベスト・ペーパーは、藤井拓氏(オージス総研)の“Design Quality with Different Design Strategies in Agile Software”と菅田直美氏(日本電気)の“Beyond CMMI Level5: Comparative Analysis of Two CMM Level 5 Organizations“が受賞されました。ベスト・プレゼンターは Timothy C. Kasse 氏(アメリカ・Kasse Initiatives、LLC) の“Twenty Points for Quality and Process Improvement – An Update Look at the Principles from Deming and Crosby“が受賞されました。

また、4WCSQ 開催準備への協力者に感謝の表彰が行われ、日本からは飯塚悦功氏(東京大学)と福井信二氏(オムロン)が表彰されました。

#### (6)閉会

飯塚悦功氏(東京大学)から、WCSQ の歴史と今回の 4WCSQ の総括が紹介され、次回 5WCSQ は 2011 年 9 月から 11 月のいずれかの時期に中国・上海にて開催することが発表されました。論文募集案内は 2010 年 3 月、アブストラクトの締め切りは 2010 年 9 月の予定とのことです。最後に、残念ながら 4WCSQ に出席できなかった居 徳華氏(中国・上海華東理工大学)から届いたパワーポイントの資料とボイスレターを使って上海の紹介が行われ、和やかなムードのうちに閉会となりました。



(Company、 product or service names may be trademarks or service marks of their respective holders.)

#### プロフィール

製品保証、ISO9000 内部監査員、各種能力水準アセスメント、品質教育講師、サービス品質保証などを歴任。ISO/IEC 15504 co-editor、IPSJ 情報規格調査会 SC7/WG10 幹事、同 SC7/WG20 委員、INSATC/WG2 委員、日科技連 SQiP ステアリング委員会委員、日科技連 SQiP 資格運営小委員会委員、IPA/SEC エンタプライズ系ソフトウェア開発力強化推進委員会委員、SQuBOK 策定部会初代リーダー。

## 5. 憩いの広場「体力を科学する」

### ■ 第2回 その場でできる心身評価～1秒でできる体力評価①～

清泉クリニック整形外科スポーツ医学センター  
施設長 脇元 幸一

パソコンに向かってデスクワークしている皆さん、肩こり・腰痛・頭痛などに悩まされていませんか？

さて第2回となる今回は「体力」についてお話します。体力とは大きく2つに分けることができます。それは、「筋力」と「柔軟性」です。この2つを同時に持ちえる方が、体力がある、といっても過言ではありません。そのなかで、今回は「簡単にできる筋力検査」についてお話しますので、自分自身の筋力を把握してみてください。

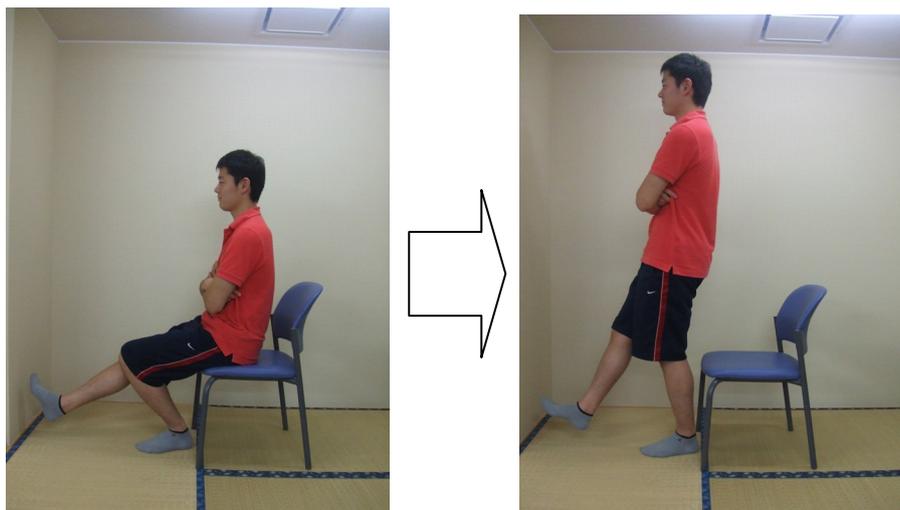
#### あなたの筋力はどれぐらいか？

皆さんがやらなければ、と気付いていることなのに、なかなか実行できないのが運動習慣ですね。この地球上に運動を行う中で絶対に無視できないのが重力です。地球上の人は動作を行うとき、必ず重力に抵抗できなければ動くことができません。この重力にどれだけ勝てるかが筋力なのです。今回はその場ですぐできる筋力テストをお教えします。

それは、下の図に示すように『何センチの高さから片脚で立てるか』をやるだけで筋力が点数化できるのです!! この筋力は『体重支持指数』といって医学的にも根拠ある筋力指数です。



では、普通のパイプ椅子を用意してみてください。パイプ椅子の高さは約 40cm です。

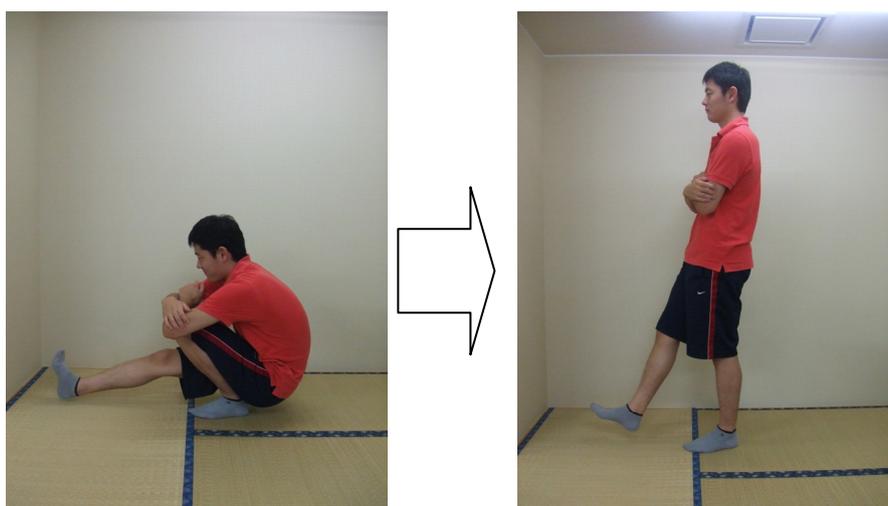


左のように椅子に座り、片脚を宙に浮かし膝を伸ばした状態で座ります。次に反対の床についでいる足で反動なくゆっくり立ち上がってみましょう。このとき、痛みなどがある場合は無理のないようにしてください。

この約 40cm の高さから問題なく立てた方は、以下のように台や椅子を低くして何 cm から立てるかを試してみてください。(週刊誌などを重ねて簡単な台にしてください)

さて結果は…？

### 0cm(床)から立てた人



130 点！これが出来た方は筋力が十分です。どんなことをやっても筋肉は悲鳴をあげない人です。また少々のストレスでは体に変調をきたしません。

### 15cm から立てた人

筋力は問題ないと言える人です。

### 30cm から立てた人

日常生活はほぼ大丈夫でしょう。しかし、ちょっとした油断、生活ストレスなどが蓄積すると肩こり・腰痛・頭痛などの症状が出現する恐れがあります。

### 40cm から立つのが限界な人

この高さから立てた人は体力の低下がみられます。

このような方はその場の対処療法(マッサージ、電気、薬)でも症状がなかなか改善できない体力となります。

### 40cm から立てなかった人

この高さから立てなかった人は、黄色信号から赤信号です。日常生活をしているだけでも体にストレスがかかる状態です。つまり、生きていることがストレスとなります。

この状態の方はいきなり運動を始めても筋力が上昇するどころか、症状が悪化する可能性もありますので、プール歩行から運動を開始することが大切です。

私は若いからといっても体力が必ずしも伴うものではありません。また、今は症状がなくても今後症状が出現する予備軍かもしれません。まずは己を知ることからはじめましょう。ぜひお試しください！

ここでもう1つ筋力について情報を示します。大切なのは、筋力とはただ動くことだけではないのです。実は…、整形外科的疾患である肩こり・腰痛・膝痛などは筋力が無くなっていることが原因で生じるのです。

筋力が落ちている方はレントゲンでも特徴的なものがみられます。



<正常な S 字>



<S 字消失>

写真を見てすぐわかるように、正常な背骨はS字状になっていますが、筋力が低下している背骨はまっすぐな背骨になってしまっています。筋力がある人は筋肉で体を支えられるので背骨をS字状に保つことができますが、筋力が低下している人は筋肉で自分の体を支えることができないので背骨の骨同士で体を支えるようになってしまい、結果としてまっすぐな背骨になって体を支えようとしてしまうのです。つまり、背骨のS字がなくなってきている人は筋力が落ちていると考えられるのです。

このレントゲン像はどこかで見たことがありますか？そうです！第1回で伝えた頭の疲れ・内臓の疲れと同様なのです。このような様々なことが背骨のS字を消失させる原因となるのです。

第1回で頭と内臓のストレス検査、第2回で体力のうち「筋力検査」についてお伝えしてきました。次回は「その場でできる心身評価～1秒でできる体力評価②～」で、もう1つの体力の要素、「柔軟性」について紹介いたしますのでお楽しみに！（文責：清泉クリニック整形外科 スポーツ医学センター 理学診療部 嵩下敏文）

### プロフィール

脇元 幸一

医療法人 SEISEN 専務理事、清泉クリニック整形外科スポーツ医学センター 施設長

日本体操協会 アンチドーピング委員会 常任委員、日本クレー射撃協会 JOC トレーニングドクター、女子体操競技 JOC トレーニングドクター、新体操医科学サポート委員、JOC トレーナー、他

主な著書：『スポーツ選手のための心身調律プログラム』（大修館書店）、他数

近年の国際A級大会サポート活動：2000'シドニーオリンピック、2001'世界水泳福岡2000、2001'ユニバシアード世界大会、2002'パンパシフィック水泳、2003'ユニバシアード世界大会、他多数

本誌の全部、あるいは一部を無断で複写複製（コピー）ならびに転載することは、著作権法上の例外を除き、禁じられています。