

## 1. 品質

### ■ ソフトウェア・プロダクトライン ～「製品開発」から「製品群開発」へ

株式会社エクスマーシオン  
シニアコンサルタント 山内 和幸

#### 1. 従来の派生開発の問題点

製品ファミリに搭載されるソフトウェアの開発は、過去に開発したものをベースに新製品を開発する、すなわち派生開発が通常です。特に製品の種類が多く、同時期に多くの異なった製品をリリースするような製品ファミリでは、各製品の開発が並行して実施されることも少なくありません。これを、ソフトウェアの構成管理の視点で表現すると、下図のようになります。

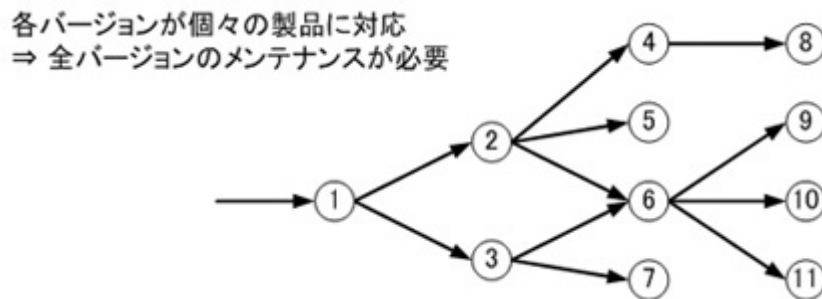


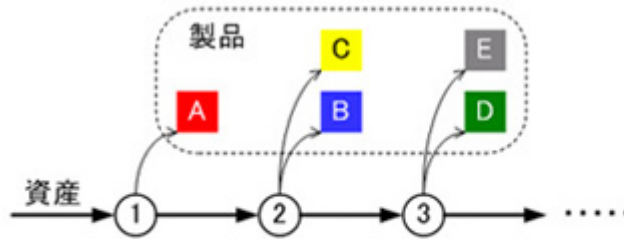
図 1：従来の派生開発における構成管理

この場合、次のような問題が起こります。

- 複数の製品に同じ機能を追加する場合、各バージョンに対して作業が必要
- ある製品で障害が発生した場合、そのベースになっている全製品に対して修正が必要

つまり、複数のバージョンに対して同一の作業をしなければならなくなり、「ムダ」が発生します。特に、バージョンが上がるにつれてソフトウェアの構造（アーキテクチャ）が無秩序に変わっているほど、ムダは大きくなります。派生の数が増えるにつれてメンテナンス対象も増加し、対応漏れ等による品質低下のリスクも増大します。

このような問題を回避するために、ソフトウェアの再利用を前提とした製品開発を実施しているケースも多くあります。これは、製品間で共通となる資産をソフトウェア部品として開発し、再利用するというものです。その考え方自体は良いのですが、実情は下図のようになっています。



各製品は、特定のバージョンの資産からしか作れない  
⇒ 全バージョンのメンテナンスが必要

図 2：従来の再利用開発における構成管理

つまり、ソフトウェアは部品として資産化されているものの、バージョン間での互換性はなく、ある製品を構築するには特定のバージョンが必要になります。この場合も、基本的に先の例と同様の問題が発生します（単に派生の数が少ないだけと言えます）。

## 2. ソフトウェア・プロダクトラインとは？

前述の問題を解決するための方法として、ソフトウェア・プロダクトライン（Software Product Line：SPL）があります。SPL では、製品ファミリに属する各製品で再利用される資産を開発するドメイン・エンジニアリング（domain engineering）と、開発された資産を利用して個々の製品を開発するアプリケーション・エンジニアリング（application engineering）という、2層型の開発を行います。ドメイン・エンジニアリングで開発される資産は、コア資産（core asset）と呼ばれます。また、アプリケーション・エンジニアリング実施時に発生したコア資産の改善要求は、ドメイン・エンジニアリングへフィードバックされ、これを受けてコア資産の更新が行われます。

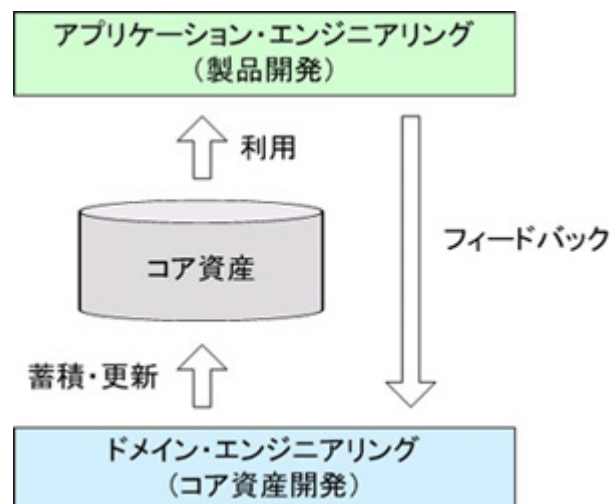


図 3：SPL 開発の概観

SPLの基本的な枠組みは上記のとおりですが、この時点で従来型の再利用と同じではないか、単に専門用語を並べただけと感じる方も多いでしょう。実際、従来型の再利用と全く異なるわけではなく、多くの点で似通った部分があります。それに加えてSPLでは、従来型の再利用では解決できなかった問題を解決するために、様々な要素が加えられています。本稿では、SPLの象徴とも言える2つの要素「スコーピング」と「可変性管理」を中心に、SPLが従来の派生開発／再利用開発とどう違うのか、そして前述の問題をどのように解決するのかについて説明します。

### 3. SPLのキー要素(1): スコーピング

製品ファミリの開発は、追加／変更／修正の連続です。その寿命を終えて開発終了となるまで、延々と続いていきます。このような状況の中で、開発した資産を長期に渡って再利用できるようにするには、「将来の変化を予測して、その変化に対応しやすいように設計する」必要があります。この変化にどこまで対応するか、その範囲を決めることをスコーピング(scoping)と言い、その範囲をスコープ(scope)と言います。

SPLの適用において、スコープを適切に定めることは重要です。一般的に、スコープが狭いと、その資産を利用して開発できる製品数が少なくなり、再利用機会が減ってしまいます。一方、製品数が少ないということは製品間の違いも多くないため、より製品に適した使いやすい資産が揃えられます。逆にスコープが広いと、製品ファミリに含まれる製品が増えるため、同一資産の再利用機会は増えます。しかし、製品間の差異も多様になり、資産が抽象的なものになってくるため、再利用が難しくなる傾向があります。コア資産の再利用による効果、すなわち製品ファミリの開発がどれだけ最適化できるかは、このコア資産の再利用のしやすさと機会によって決まってきます。よって、このバランスが最適となるようにスコープを定義することが、SPL開発の成否を大きく左右します。

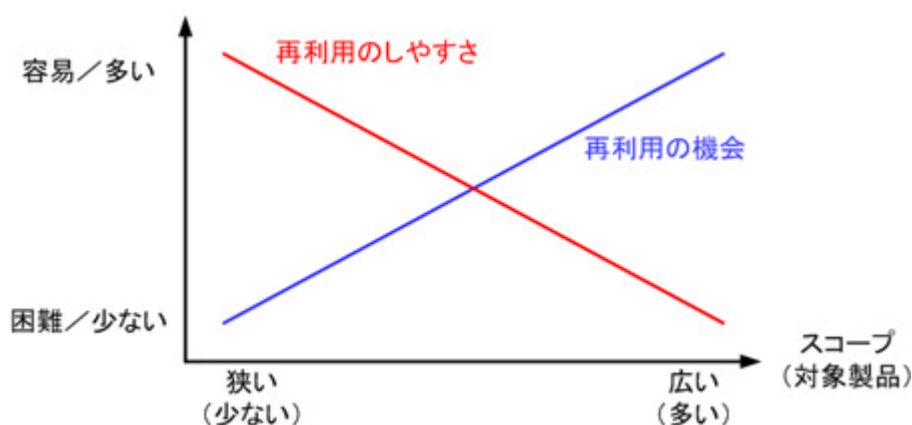


図4: コア資産の再利用の効果

#### 4. SPL のキー要素（2）：可変性管理

スコープに含まれる製品の間には、共通なもの異なるものがあります。下図を見て下さい。ここで、3つの円がそれぞれ製品 A/B/C の資産だとします。この時、全ての製品で共通となる部分を共通部（common part）、それ以外を可変部（variable part）と呼びます。可変部の内、特定の製品でしか使わない資産は、製品固有部（product-specific part）と呼ぶこともあります。SPL 開発では、この共通部と可変部を管理することで、製品ファミリの開発の最適化を図ります。これを、可変性管理（variability management）、または共通性・可変性管理（commonality-variability management）と言います。

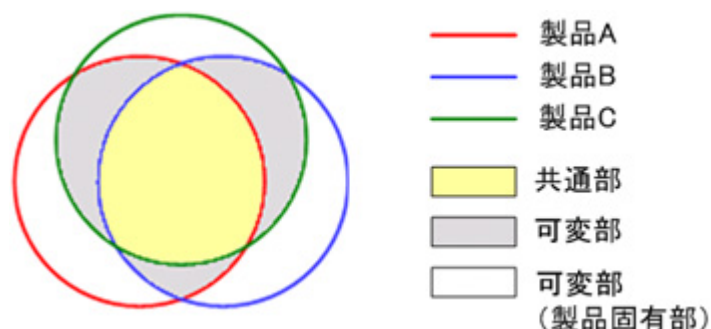


図 5：共通部と可変部

可変部には「なぜその違いが発生するのか」の理由があります。これを可変性（variability）と呼びます。裏を返せば、製品ファミリに可変性があるということは、その違いを生み出すことができることを示しています。そして、可変性には変化の種類が存在します。代表的なものはオプション（optional）と代替（alternative）です。前者は製品に搭載する／しないを選択できるもので、後者は幾つかの選択肢の1つを選ぶものです。簡単な例として、デジタル腕時計ファミリを考えます。このファミリには、以下の4つの製品が含まれるものとします。

- 12 時間表示デジタル腕時計
- 24 時間表示デジタル腕時計
- アラーム機能付き 12 時間表示デジタル腕時計
- アラーム機能付き 24 時間表示デジタル腕時計

この時、このデジタル腕時計ファミリには、時間表示機能（12 時間と 24 時間の代替）とアラーム機能（オプション）の2つの可変性が存在します。このようにして特定された可変性を管理していく際には可変性モデルを利用するのが一般的です。可変性モデルの内、最もよく知られているものはフィーチャモデル（feature model）でしょう。これは、製品ファミリを特徴づける要素（フィーチャ）をツリー上に表現したものです。デジタル腕時計ファミリの例では、以下のようにモデル化できます。

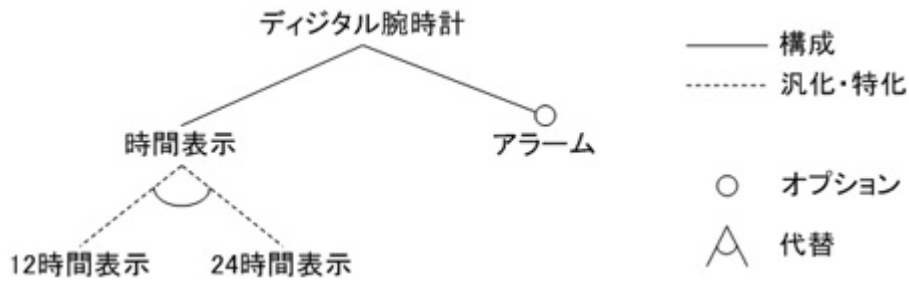


図 6：デジタル腕時計ファミリのフィーチャモデル

したがって、各製品を効率良く開発するためには、コア資産は以下の要件を満たす必要があります。

- 時間表示機能を（製品に載せる前に）12 時間／24 時間に切り替えられること
- アラーム機能の搭載／非搭載が切り替えられること

可変性を分析・管理することによって、開発すべきソフトウェアにどのような柔軟性が求められるのか、それを実現するためにどんな技術を利用すべきかが明確になります。これは、「保守性」という品質特性に関して、その要件を明確にすることと捉えることができます。

## 5. 可変性管理に基づく製品の導出

製品を開発する場合、可変性モデルに記述されている可変性をどのように扱うかを決定します。前述のデジタル腕時計の例だと、アラーム機能を搭載するのか／しないのか、時間表示は 12 時間／24 時間のどちらにするのかを決定します。この決定内容に従って、コア資産の変動点（variation point）をコンフィグレーションしていきます。変動点とは、コア資産中の可変性を実現している場所のことです。例えば、アーキテクチャ上でアラーム機能を実現するコンポーネントが定義されていて、この搭載／非搭載が切り替わるとしたら、このコンポーネント自体が変動点になります。このようにして、コア資産が現在開発中の製品に適した形に変換されます。この一連の過程を導出（derivation）と言います。

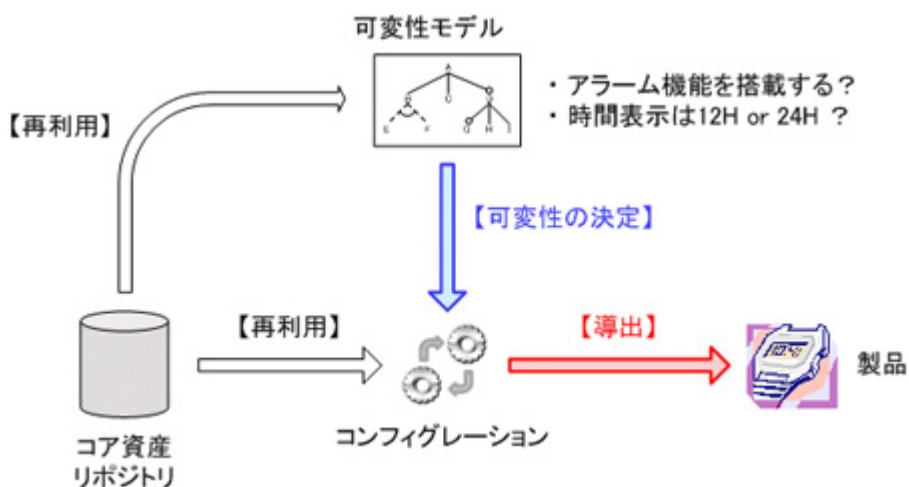


図 7：可変性の決定による製品の導出

コア資産から製品用の資産が導出されるということは、コア資産には共通部だけでなく、可変部も含まれるということがわかります。製品固有部でない可変部は幾つかの製品間で共有されるため、コア資産に含めることができます。もちろん、新製品開発時にはコア資産に含まれない新機能や、製品固有部の開発なども含まれます。前者は通常ドメイン・エンジニアリングで、後者はアプリケーション・エンジニアリングで開発を行います。

## 6. SPLによる従来型の問題の解決

ここで、本稿の冒頭で述べた保守の問題について振り返ってみます。SPL開発では、この問題を本当に解決できるのでしょうか。答えはYESです。SPLにおけるコア資産は、共通部だけでなく可変部も含まれます。そして、管理されている可変性に対して、各開発成果物（要求仕様、内部設計、コード、テスト仕様、etc.）が正しく製品を導出できるように開発されます。このため、コア資産の最新バージョンは、その時点でスコープに含まれている製品を全てカバーできる状態になっているのです。従来の開発と比較してこれを図示すると、以下ようになります。つまり、コア資産のバージョンが上がっても互換性が維持されるように、開発を行っているのです。この結果、コア資産の最新バージョンを保守するだけで、全ての製品の保守を行うことができます。

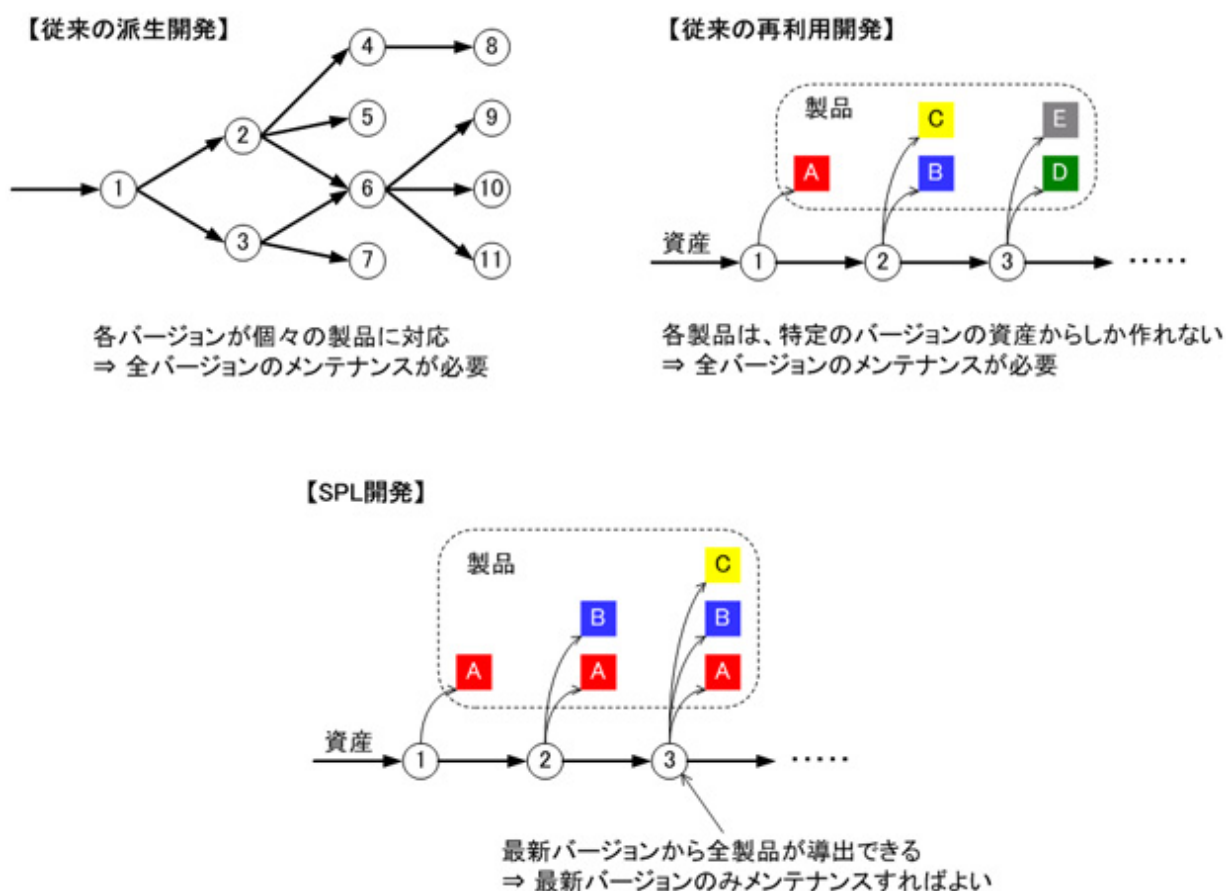


図 8 : SPL 開発と従来型の開発の比較

従来型の開発と SPL を比べてみるとわかるとおり、従来型では各バージョンが「特定の製品」に対応しているのに対し、SPL では「製品群」に対応しています。言い換えると、前者は個別最適な開発を、後者は全体最適な開発を行っていると言えるでしょう。この、「全体最適の視点で製品群全体を開発していく」という姿勢こそが、SPL の根幹を成す考え方なのです。

SPL は単純な開発技術ではなく、ビジネス戦略や開発プロセス・組織構成の最適化等も含んだ包括的なパラダイムであるため、非常に大きな工学体系になっています。このため、そのポイントを上手く掴んで、実開発に役立てることが重要です。本稿では、SPL というパラダイムの一部を、スコーピングと可変性管理を中心に要点だけを説明してきましたが、多くの方の改善の一助となれば幸いです。

#### プロフィール

山内 和幸（やまうち かずゆき）

株式会社エクスマーション シニアコンサルタント

ソフトウェア開発の生産性・品質向上に関して長年取り組んでおり、アーキテクチャ設計や可変性の管理といった技術的観点から、SPL の普及・展開の支援を行っています。特に近年は、これまで開発してきたソフトウェア資産を活用した SPL 開発への移行の支援に力を入れています。



## 2. 人材育成

### ■ 人材育成…よりも、自己育成

日本電気(株)  
IT ソフトウェア事業本部  
統括マネージャー 菅田 直美

「人材育成」というテーマで執筆を依頼されました。正直なところ、人材育成は得意領域ではありません。企業で働く人間として「人材育成」はとても重要なテーマなのですが、本音では、「人材育成なんて実は関係ない、育つ人は育つ・育たない人は育たない！」と強く思っているからです。私には、「自分で自分を育成する」ことに優る人材育成は思いあたりません。そんなわけで、自分自身を振りかえって、自分の成長に良かったと思う自己育成のコツについて幾つかご紹介したいと思います。

#### 1. 本を読む

何か問題にぶつかったとき、何をしますか？ 先輩や上司に聞く？ インターネットで調べる？ それもいいですが、私は、こんなときこそぜひ専門書を読んで調査してほしいと思います。自分の抱える問題が、有史始まって以来、初めて発生した問題だなんてことはほとんどあり得ません。間違いなく先人に同じ問題を抱えて、解決した人がいるのです。それを利用しない手はありません。

私が自己育成で最も効果があるのは何かと問われたら、本を読んで自分の頭で考えることを挙げます。読むだけでなく、自分の頭で考えて消化し、自分の意見として説明できるようになることが重要です。私はソフトウェアを専門に勉強したわけではないので、会社に入ってソフトウェアの仕事について、本を読んで勉強するしかありませんでした。私が入社したころは、まだ日本語のソフトウェア工学の専門書は少なかったので、専門書を読むようになって数年もすると、たいてい見たことがある本ばかりになりました。ソフトウェアの本はこれくらいしかないのかとわかって驚いたことを覚えています。今は、ソフトウェアの本がずいぶん増えて、むしろ良い本を選別する能力の方が重要かもしれません。いずれにしても、本を読んで自分の頭で考えることくらい、自分の成長に役立つことはないと思います。なにより、知らないことを知るのはとてもわくわくすることだし、なるほどと納得することはとても楽しいですから。そして、それを単に読んで楽しむだけでなく、自分の頭で考えて、自分のものにすることがとても重要です。心から納得すると表現すればいいでしょうか、自分の頭で考えるというのはそういう感覚です。

こんな当たり前のことをわざわざ書くのは、ソフトウェア領域でこのアプローチをしている人がそんなに多くないと思うからです。たいていの場合、人海戦術や腕力で切り抜けようとしていると思います。「がんばります！」と体力で勝負する方法ですね。なぜでしょうか？ 専門書を調査して問題を解決するという経験がないのかもしれませんが。いつも徹夜してがんばり通してなんとかしてきたから、それが当たり前になっているのでしょうか。これは、業界として危惧すべき事項だと思います。私達がそんなことをしているから、ソフトウェア技術者は5Kとか10Kとか言われてしまうのです。



先日、JaSST 東京の基調講演で、リー・コーブランド氏が、「最近読んだ専門書を教えてくださいと聞いて、専門書は読んでいませんと答える外科医に、手術してもらいたいと思いますか？」という表現で、本を読むことの重要性を話しておられました。確かにそうです。専門書を読まないエンジニアといっしょにソフトウェア開発の仕事をしたとは思いません。みなさん、本を読みましょう！

## 2.自分の考えを文章にまとめる

NECには、その昔、SWQCと呼ぶソフトウェア開発の小集団活動がありました。SWQCでは成果の論文発表が義務付けられていて、それが形骸化の一因となり、いったん活動を休止した経緯があります（現在は、コミュニティという形式で残っています）。それでも、私はSWQC活動で学んだことは多かったと思います。

その理由は、論文を書いて発表することが義務付けられていたので、仕事の区切りがつくたびに、自然と自分の考えを文章にまとめ、人前で発表する経験ができたからです。文章にして自分の考えをまとめると、取り組んでいたときには正しいと思っていても、論理的な矛盾や不足があることに気がつくことができます。文献調査の必要性を実感したのもこのころです。自分一人で考える工夫には限度があって、先人の経験を取り込んでそれに自分の工夫を重ねる方がより良い成果が得られるのだと実感しました。また、人前で発表することによって、論理的に説明するだけでなく、わかってもらえるように話すことの難しさも学びました。

私が論文を書くことを勧めるのは、自分のこのような経験に基づきます。でも、実際に書いている人は少ないですね。時間がないのはわかりますが、時間の有無は関係ありません。要はやる気です。ただし、文章にまとめることが未経験だと、何をどうしたらいいのか、最初のきっかけがつかみにくいかもしれません。そういうときは、論文募集している団体の論文執筆サポートを利用しましょう。ちなみにSQiPソフトウェア品質シンポジウムにも論文執筆サポートがあります。ぜひ利用してみてください。

実際に書いてみれば、書くことの難しさと重要性に気がつくと思います。ちなみにパワーポイントでは同じ効果は得られません。きれいな図でごまかされて、できたような気がするだけです。口数の多い人は、プレゼンでごまかしてしまうかもしれません。プレゼンの能力と、文章で論理的に記述する能力は異なります。自分で説明してまわれる範囲は限られるのですから、読んでわかるように書くことができなければ、広く自分の意見を理解してもらえません。読んで理解できるように文章を書くことが出来ない人は、技術者とは言えないと私は思っています。

### 3.社外に出る

#### 3.1. 参加者からみた WACATE の特徴

SQiP にソフトウェア品質管理研究会というのがあります。入社したてのころ、私はこの SQiP 研究会へ1年参加させてもらいました。SQiP 研究会では、1年に7~8回の終日かけた研究会が開催され、ソフトウェア品質に関連するテーマについてグループに分かれて研究し、最後にはグループで論文にまとめて発表します。さまざまな会社の方が参加するだけでなく、その領域の有識者がグループを指導してくれます。私にとって、社外に出るという経験はこの SQiP 研究会が初めてでした。たしか、メトリクスをテーマに選んだ記憶があります。そのときの指導役は忘れもしない、保田先生（日立製作所（当時）、つくば国際大学（現））と堀内さん（日立製作所（当時））でした。SQiP 研究会には合宿もあるのですが、当時は箱根の小涌園が合宿会場でした。その小涌園での議論の光景は、今でも目に焼きついていません。保田先生と堀内さんには、今でもお付き合いいただいています。1年かけた研究はとても良い経験になりました。なにより、社外の方と話げできたことが良かったと思います。自分の困っていることや考えていることが他社の人に理解してもらえるのか、他社も同じようなことを悩んだり検討したりしているのか、自社にない新しい試みをしている会社があるのか…等々、多くの刺激を受けました。なにより、社外の人に社内の用語は通じないというあたりまえのことを、実感したことが忘れられません。自分の仕事さえうまく説明できなかつたときのことは、今も恥ずかしさとともに思い出します。相手にわかるように説明できる能力というのは、実はとても高度な能力なのだと思います。社外での交流は、その練習の場にもなります。当時は、ソフトウェア関係のシンポジウムもほとんどなく、他社の方と交流できるような機会はとても少なかったので、私にとって SQiP 研究会は貴重な経験でした。手前みそになりますが、自分自身の体験から、SQiP 研究会はとてもお勧めの教育コースの一つです。

業務が忙しくて、なかなか社外に出る機会が少ないかもしれませんが、会社を休んでも社外の集まりに参加し、社外との交流を持つことは、間違いなく自分を磨く貴重な場になります。ソフトウェア関係のシンポジウムなどは、何年か続けて通うといいと思います。各専門領域の主要なプレーヤーがわかるようになるはずで、主要な人物はそれほど多くないことがわかると思います。また、そういう集まりに参加する人達とも顔見知りになると思います。そういう人たちとの関係が重要なのです。困った時に助けてくれる良き友になると思います。私自身は、社外のお友達にどれだけ助けられたかわかりません。会社と会社はどこでつながっているかわからないし、お互いお客様であり同時にお客でもあるわけで、ビジネスの場面で顔を合わせることもしばしばあります。そんなとき、事前に必要な情報を入手できるかどうかは、実質的にはともかく、精神的には大違いです。特にトラブル対応のときは背景も含めた情報量がものを言います。私は、社外のネットワークが業務で直接役立つようになるまでに、20年くらいかかりました。役立てようと思って顔見知りになったわけではないですが、最近ひしひしと社外ネットワークの重要性を感じています。20年我慢（！）してお友達で良かったと（おそらくお互いに）思うこのごろです。

#### 4.宴会に出る

社内でも社外でも、人的ネットワークを作るには、宴会が一番です。最近では、社内のイベントでも、学会やシンポジウムなどのイベントでも、併設してお酒が出る交流会が開かれることがしばしばあると思います。それにはぜひ参加すべきです。そこには、ネットワークを作りたいと思っている人がたくさん参加します。皆、交流を求めて参加するのですから、人的ネットワークを作るのに最適です。

私は、まわりの人に必ず社外の宴会には参加するように言っているのですが、なかなか参加してくれませんが、なぜでしょうか。業務が忙しいから？ イベントの日は残業がないのだからそんな日くらい早く帰りたい？ おもしろくなさそう？…。理由はいろいろあると思いますが、ここは騙されたと思って、設定されている宴会には必ず参加してみてください。お酒の席でしか聞けない話が必ず出てきます。私は、自分の人的ネットワークは、ほとんど宴会の席で作ったと思います。なんというか、お酒の席は、本音が出ますよね。きっとそれがいいのだと思います。私もきつとずいぶん好き勝手なことを言ったと思いますが、それで信頼関係を作ったことも多いと思いますよ。

また、お酒の席で重要なことが決まったりします。人事とかね。私は、自分の転機になる大きい仕事が宴会の席で決まった経験があります。どうも、指示する側も、宴会の席でもない私が引き受けなさそうだと考えたようなのですが、私も、宴会の席でなければ、エイヤで引き受けることはなかったと思います。それは、2年もかかる大きなプロジェクトでした。その時のことを思い出すたびに、言う側も受ける側も勢いが重要なときがあるなあと思います。こんなふうに、宴会は様々な変化を生む場なのです。

#### 5.自分の意見を一言で言う

私が人前で話すときに事前に自己チェックする点は、言いたいことを一言で言えるかどうかです。その講演で伝えたいことを一言で説明できるか、各スライドで言いたいことを一言で説明できるかを確認します。一言で説明できないときは、自分の中でその内容を整理できていません。たとえば、スライドで何行にも渡って説明しているようなときは、ほとんどNGです。ああでもない、こうでもないという自分の頭の中でぐるぐる回っている混乱状態が、そのまま書かれているだけです。文章を書く場合でも同じです。各章での主張を一言で説明できないときは、結局何を言いたいのかわからない文章になってしまいます。

私が他の人のスライドや文章をレビューするときも、そういう観点でまずチェックします。ですから、スライドの場合は一目見ればすぐに混乱しているかどうかわかりますね。タイトルが一言で表現されているか、中身がぱっと見れば目に飛び込むように表現されているかということです（これは決して、デザインとか色調の問題ではありません）。文章もしかりです。技術系の文章はそれほど難しくなくて、書くべき条件が揃っていれば、たいてい主張はわかります。

皆さんにお勧めしたいのは、常に自分が言いたいことを一言で言えるかを確認することです。それができれば、言いたいことが伝わらないということはなくなるはずで、自分の頭の中が混乱しているのか、整理された状態にあるのかも自覚できると思います。整理された状態であれば、議論の焦点は、お互いの主張を認められるかどうか絞られるので、議論の時間は最小限で済みます。議論が長時間に及ぶのは、当事者が言いたいことをわかっていない場合が多いのです、お互いに何を言いたいのかを知るまでに時間がかかってしまいます。言いたいことを知るために、根掘り葉掘り聞くので時間がかかり、結局わかってみると、時間をかけて聞くほどの価値はなかったりするので、自分が整理された状態であれば、反論されても素直に受け止められるはずで、主張の違いが明確にわかるので、冷静に対処できるようになります。逆に相手のほうが混乱していると気がつくこともあるでしょう。実に生産的な時間を過ごすことができるようになります。

## 6.さいごに

私が自己育成として重要と思うことをまとめました。参考になれば幸いです。私たちは、出来上がってしまったら、もう店じまいするしかありません。まだ若いのに、一丁上がり状態になっている人を見かけます。一丁上がり状態というのは、自分はあるレベルに達したと勝手に思いこんでいる状態と表現すればいいでしょうか。勝手に偉い先生の立場になってしまい、回りの指導ばかりしている人です。評論家のような発言は並べ立てるものの、自分では何一つ実行できません。とても残念ですし、回りにとっては迷惑です。これは本人の意識の問題です。もっと知りたい、もっと自分の好奇心を満足させたいと思う限り、私たちは成長するものだと思っています。いつまでもそういう自分でありたいと、心から願う次第です。

### プロフィール

誉田 直美 (ほんだ なおみ)

日本電気(株) コンピュータソフトウェア事業本部 統括マネージャー

上席ソフトウェアプロセス&品質プロフェッショナル

日科技連 SQiP ソフトウェア品質委員会 副委員長

サーバ/メインフレーム/ストレージ等の基本ソフト/ミドルソフトウェアの品質保証および

CS 向上に従事。2007 年より現職。

### 3. SQuBOK®

#### ■ SQuBOK ガイドの改訂に向けた活動について（参考文献の活用方法）

(株)NTT データ 技術開発本部  
プロジェクトマネジメント・イノベーションセンタ  
町田 欣史

##### ■はじめに

SQuBOK ガイドには、ソフトウェア品質に関する非常に多くの情報が盛り込まれていますが、皆さんは、これらの情報をどのようにお使いでしょうか。今回は、SQuBOK ガイドの使い方について、再度考えてみることにしましょう。

SQuBOK は、その名の通り、ソフトウェア品質に関する知識を体系化したものです。その体系を表したものが、SQuBOK ガイドの序章にもある樹形図です。数多くあるソフトウェア開発の技術要素の中から、ソフトウェア品質に関連するものを選び出し、それらを分類、体系化したものをツリー状に表現しています。

この樹形図は、SQuBOK（ガイド）のオリジナルですが、樹形図以外の内容は、基本的には他の文献から知ることができる情報がほとんどです。こう書いてしまうと、SQuBOK ガイドの価値が下がってしまうような気がするかもしれませんが、そんなことはありません。SQuBOK の目的の一つは「ソフトウェア品質に関する日本の暗黙知を形式知化する」ということです（ソフトウェア品質知識体系ガイド－SQuBOK Guide－ vii ページ参照）。つまり、この「体系化したこと」と「他の文献の情報を集約したこと」こそが、SQuBOK ガイドの大きな価値なのです。

##### ■SQuBOK ガイドから知識を広げる

SQuBOK ガイドを読めば、ソフトウェア品質に関することが“ある程度は”理解できますが、さらに専門的な知識を得るためには、それだけでは足りません。SQuBOK ガイドを入口として、さらに知識を広げていくことが必要になります。

身近な例で考えてみましょう。例えば、先ごろ行われたサッカーのアジア杯で日本代表が優勝したことでサッカーに興味を持った方が、より深くサッカーを知ろうと思ったら、どのようなことができるでしょう。

- ルールや戦術を知る（なぜ 4-2-3-1 のフォーメーションにしたの？）
- 過去の歴史を知る（過去のアジア杯の激闘、アジア杯の開催地ドーハの悲劇、など）
- 代表選手が所属するクラブチームを知る（あの選手はどのチーム？活躍してる？）

など、知識の幅は様々な方向にあります。

これを SQuBOK ガイドに置き換えてみると（やや強引ですが）以下のようになります。

- 各知識領域やトピックスの定義、理論などを知る。  
→基本的な考え方を知って、応用に活かす。
- 各知識領域やトピックスのルーツを知る。  
→現在の技術が確立するまでの裏付けを知る。
- 各知識領域やトピックスの活用事例を知る。  
→本当に有効な技術かどうかを確認する。

これらを知る上で助けになるのが、SQuBOK ガイドに記載されている参考文献、関連文献や、付録にある推奨書籍、規格、論文などです。SQuBOK ガイドを、これらの文献へのポイントとして活用することは、SQuBOK ガイドの有効な活用法の一つです。

それでは、文献を「書籍」「論文」「規格」の3つに分けて、それぞれの入手方法や特徴などについて見ていきましょう。

## ■書籍を読んでみよう

書籍は一般の方が最も入手しやすい文献でしょう。大型書店のコンピュータ書籍コーナーに行けば、ソフトウェア開発関連の多くの書籍（和書）が陳列されています。SQuBOK ガイドの第1版が出版されてから約3年が経ちましたが、その間にも数多くの書籍が新たに出版されました。改訂中の SQuBOK ガイドでは、新しい書籍を関連・参考文献に追加する際には、複数の SQuBOK 策定部会メンバが読んだ上で、追加の可否を判断しています。従って、単に情報が新しくなったというだけではなく、読者の方にとって有益な書籍のみが掲載されます。

また、SQuBOK ガイドに記載されている書籍の中には、絶版本や洋書など、入手が難しいものもあります。洋書はちょっと頑張ればオンライン書店などで手に入りますが、絶版本は古書店などでないと入手は難しいかもしれません。そのようなときは、図書館を利用するのもお勧めです。絶版になるような古い本にこそ、技術の真髄が書かれていることもありますので、より深い知識を得たい方は諦めずに探してみてください。



## ■論文を読んでみよう

いわゆる学術論文は、研究をしている方でないとあまり接する機会はないかもしれませんが、最新の技術を知る、あるいは現在の技術のルーツを知る上では非常に有用な情報源です。

SQuBOK ガイドの付録 D には、ソフトウェア品質に関係が深い論文として、以下の 3 つのシンポジウム、ジャーナルの表彰論文を掲載しています。

- ソフトウェア品質シンポジウム（日本科学技術連盟）
- 世界ソフトウェア品質会議（European Organization for Quality, American Society for Quality, 日本科学技術連盟）
- ジャーナル「品質」（品質管理学会）

これらの表彰論文は、発表、掲載された論文の中でも特に優れたものということで、代表として掲載していますが、実際には表彰論文以外にも多くの論文があります。また、上記以外にも、ソフトウェア開発やソフトウェア品質に関する論文は、情報処理学会や電子情報通信学会、IEEE などにも多く投稿されています。これらの学会の論文は、学会に入会すれば（会員種別にもよりますが）、無料で入手できる場合が多いです。また、入会しない場合でも、有料になりますが、論文を購入することもできます。

論文には、最後に必ず参考文献が掲載されています。論文は、ページ数が限られているために、参考文献で述べられていることは既知の事実として、根拠や詳細などは省略していることも多くあります。そのため、より正しい理解のためには、参考文献をさかのぼって見ていくことが必要になりますし、その中で新たな気づきが生まれることもあります。この参考文献の活用は書籍でも同様ですので、ぜひお試しください。

## ■規格を読んでみよう

ISO や JIS などの規格は、名称は聞いたことはあっても、実際に現物を見たことがあるという方は少ないかもしれません。組織の品質保証部門などの方であれば、国際規格標準や業界標準に準拠するために参照することはあると思いますが、通常の開発に携わっている方にはなじみがないかもしれません。

例えば、ご自身の会社における開発標準があるとします。それが国際規格に準拠しているものとして提供されているのであれば、その開発標準を作った方は、国際規格を読んで、それを会社の特性に合わせて標準化したのだと思います。この場合、開発標準を見た方は、間接的に国際規格を参照していることになります。

しかし、国際規格に準拠しているとは言っても、加工されてしまったものを見ているわけです。好奇心の旺盛な方であれば、オリジナルの定義を知りたいと思うことでしょう。

ただ、この規格は非常に高額なので、個人で購入するのは難しいかもしれません。業務で必要ならば、その必要性を訴え、会社の予算を工面して購入を検討していただきたいのですが、それが無理であれば、国立国会図書館などで閲覧することもできます。いずれにせよ、ご興味のある方は、規格とは実際どんなものなのか、一度ご覧になることをお勧めします。

## ■さいごに

今回は、改訂の話から少しはずれて、SQuBOK ガイドを元に、ソフトウェア品質に関する理解をより深めるためのテクニックについて紹介しました。今回紹介した書籍、論文、規格といったものは、どんどんと更新、追加されています。新たな SQuBOK ガイドでは、それらの最新情報を取り込んでブラッシュアップすると同時に、「温故知新」の言葉通り、古典的かつ現在の技術の基礎となる情報も大切にしながら改訂を進めています。

※2011年10月31日～11月4日に第5回世界ソフトウェア品質会議（5WCSQ）が上海で開催されます。

ソフトウェア品質シンポジウムの発表論文は、日本科学技術連盟の Web サイトで公開されています。  
<http://www.juse.or.jp/software/29/>

### プロフィール

町田 欣史（まちだ よしのぶ）

(株)NTT データ 技術開発本部 プロジェクトマネジメント・イノベーションセンター

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、テスト自動化支援ツールの開発・導入支援、ソフトウェア品質向上のための社内サービス運用を行っています。

## 4. トピックス

### ■ 2010 年度（第 26 年度）ソフトウェア品質管理研究会の報告

鷺崎 弘宜(早稲田大学 / 国立情報学研究所)

執筆協力: 阪本 太志(東芝デジタルメディアエンジニアリング(株))

ソフトウェア品質管理研究会（通称：SQiP 研究会）は、主に品質の観点からソフトウェアの開発技術とマネジメント技術を研究、調査、学習、さらには交流 する場として日本科学技術連盟のもとに設置され、2010 年度で 26 年目を迎えました。4 月からテーマごとに分科会に分かれてほぼ毎月のペースで定例会を 持ち、約一年間にわたり活動する形態をとります。2010 年度も 2011 年 2 月 25 日の成果発表会をもって活動を締めくくります。本稿では 2010 年度に おける一年間の研究会活動を振り返り、さらには 2011 年度の活動予定を紹介します。

#### ■2010 年度のテーマは「ソフトウェアと品質を考えぬく一年間」

ソフトウェア品質技術の実践にあたり、書籍やセミナーで学び後は現場の経験頼みということでは、狭い世界の自己流の域を出ません。組織標準があるといっても、それが最適と確信できることは稀でしょう。まして近年は、ネットワークや社会環境など変革の速度を増しています。一つとして同じことのないそれぞれの状況で技術を活用するには、基礎理論を習得し、エキスパートから助言を得て、同じフィールドで悩む仲間と多様な事例や視点を共有しつつ共に考え抜き、問題や解決の本質を見抜くことが重要です。

そこで 2010 年度の研究会では、「ソフトウェアと品質を考えぬく一年間」をテーマに掲げて、特別講義を含む様々な機会を通じた理論習得と事例把握、主査・副主査の丁寧な指導、小規模な分科会単位における密な議論と気づきの共有を通じて、「考え抜く」ことに取り組みました。「考え抜く」とはつまり、安易に最初から既存技術や書籍に答えを求めるのではなく、徹底的に「問い」や実践を重ねて問題を掘り下げたうえで、同じく問いや実践を重ねて妥当で有効な解決を個人や集団で検討することです。それらの取り組みを通じて最終的には、各参加者の個人ならびに所属先において理論や経験に裏打ちされた品質活動の実践と人材育成を目標としました。

ソフトウェア品質管理研究会は、それらの全てを提供します。研究会は 25 年の歴史を持ち、次の 25 年に向けてレビュー分科会の新設など絶えず改善し続けています。企業における品質活動の導入、発展、さらには人材育成に是非お役立てください。

## ■ニーズにあった研究のかたち：調査、実践、研究

研究会には 66 の企業や団体から 68 名が参加しました。年間を通じて 8 回の例会を行い、さらに 2010 年度は 2009 年度に引き続き研究調査の一環として第 29 回 SQiP シンポジウム（ソフトウェア品質シンポジウム）[1]についても参加しました。各例会では、午前中は参加者全員で特別講義を聴講し、午後はテーマ別の分科会・コースにわかれて学習や研究を行いました。特別講義は様々な知識や事例に触れて視野を広げることを目的とし、分科会・コースは特定の問題や技術について深掘りすることを目的としています。それぞれについて年間テーマに沿った事前の計画や、参加研究員の悩みやニーズに沿った柔軟な対応、さらには主査・副主査・アドバイザー・外部招聘講師による丁寧な指導により、各回のアンケート回答における満足度が高く充実した良い内容であったと評価しています。それぞれの詳細を以下に報告します。

### ◇特別講義：ソフトウェア品質の広がり

ソフトウェア開発の難しさの要因として、ソフトウェアが質量を有さずしばしば抽象的で自由度が高いことや、それに関連してハードウェアやネットワーク、人間・ビジネス・社会系、自然界・実世界などがソフトウェアに複雑に絡み合うことが挙げられます。そのため、従来の工業製品における開発や品質管理の考え方がそのままでは当てはまらないことがしばしばあります。そこで、人的側面、プロセスやマネジメントの側面、設計や検証などエンジニアリングの側面、さらには経済学や統計学などの広がりを知り、様々な気づきを得ていくことが重要です。

そこで 2010 年度の特別講義は、各分野で活躍されている方々より下記の講演をいただきました（図 1 参照）。ソフトウェア品質の本質について考えることの意義に始まり、メトリクスや品質管理、アジャイル開発といったマネジメント系から、プロダクトの開発におけるレビューやテストといったエンジニアリング系に至るまで幅広く、ソフトウェア工学と品質管理に関係する形で各話題が豊富な事例と共に取り上げられ、各研究員はそれぞれ多くの気づきを得ることができました。各特別講義の概要や感想は [2]より参照できます。

- [「今こそ考えよう！ソフトウェア品質」](#)、菅田 直美（日本電気株式会社）
- [「ソフトウェアメトリクスの基礎 ～欠陥に学ぶ～」](#)、野中 誠（東洋大学）
- [「アジャイル開発の方法論～イテレーション体験を通して～」](#)、鷲崎 弘宜（早稲田大学/国立情報学研究所）
- [「新聞紙タワー演習」](#)、足立 久美（株式会社デンソー）、堀田 文明（デバッグ工学研究所）
- [「Usability 2.0～ユーザビリティの現在・過去・未来～」](#)、樽本 徹也（利用品質ラボ）
- [「レビューのすすめと進め方～レビューってなんだろう？ プロジェクト・業務を乗り切るための私流レビューの進め方～」](#)、安達 賢二（HBA）
- [「ソフトウェア開発の工業化と品質管理」](#)、渡辺 純（富士通アプリケーションズ）
- [「W モデルによる全体最適化」](#)、西 康晴（電気通信大学）



図 1: 特別講義の様子

#### ◇分科会・コース: ソフトウェア品質の深堀

研究会への参加目的として、2009 年度に引き続き下記の 3 種を主要なものとして識別しました。研究会は本来「研究」をする場として、新たな技術や考え方の発明を追求する研究型、ならびに、既存技術や知識の適用ノウハウの顕在化や体系化を追求する実践型が主要な活動といえます。しかしながら、それらの挑戦にあたっては基本的な既存技術や知識の把握が不可欠であり、ソフトウェア品質への取り組み要請の増大も相まって、調査型についても研究会の主要な活動の一つと位置付けています。詳しくは、下記記事や講演資料に掲載されていますのでぜひ見てみてください。

- 研究型（分科会）：新技術の発明
- 実践型（分科会）：実問題への既存技術の適用ノウハウ
- 調査型（コース）：既存技術の整理と習得

具体的には、ソフトウェア品質に関する幅広い要請に応えられるように、以下に示す研究・調査型の 6 分科会、ならびに、学習型の 3 コースを設けました（鉤括弧内は研究テーマ）。各分科会・コースが主に扱う領域を Software Engineering Body of Knowledge (SWEBOK、ソフトウェアエンジニアリング基礎知識体系) [3]上で整理した結果を図 2 に示します。いずれもソフトウェア品質を核としながら、分科会全体としてマネジメントからエンジニアリングまでの領域を広くカバーしていることを見取れます。

- 第1分科会 ソフトウェアプロセス評価・改善
  - 主査：三浦 邦彦(矢崎総業(株))、副主査：阪本 太志(東芝デジタルメディアエンジニアリング(株))
  - 「プロセスは定着していますか Part3 ～不具合事象に基づく標準プロセスへのフィードバック手法の提案～」
- 第2分科会 プロジェクトマネジメント
  - 主査：早川 勲((株)山武)、副主査：板倉 稔((株)ビズモ)
  - 「外乱への適切な対応がプロジェクトを救う」
- 第3分科会 ソフトウェアレビュー
  - 主査：細川 宣啓(日本アイ・ピー・エム(株))、副主査：永田 敦(ソニー(株))、  
アドバイザー：森崎 修司(奈良先端科学技術大学院大学)
  - 「間接的メトリクスを用いて欠陥予測を行うレビュー方法の提案 ー欠陥の位置と種類の特定によりレビューの効率と効果を向上ー」
  - 「ソフトウェア品質不安に対する心理的側面に着目した、レビュー計画作成技法の提案」
- 第4分科会 ソフトウェア・ユーザビリティ ーエンドユーザ視点でのソフトウェア開発ー
  - 主査：金山 豊浩(UPA Japan 設立準備事務局)、副主査：福山 朋子((株)インテック)、  
三井 英樹((株)ビジネス・アーキテクト)
  - 「「体験」を伴ったソフトウェア開発における課題: エンジニアのフューチャービジョンとは？」
- 第5分科会 ソフトウェアテスト
  - 主査：秋山 浩一(富士ゼロックス(株))、副主査：奥村 有紀子(デバッグ工学研究所)
  - 「Web アプリケーション開発における画面モックアップを利用した画面仕様書およびテスト仕様書の自動生成と設計作業の効率化の提案」
- 第6分科会 派生開発
  - 主査：足立 久美((株)デンソー)、  
副主査：飯泉 紀子((株)日立ハイテクノロジーズ)、清水 吉男(システムクリエイツ)
  - 「派生開発で成功するための施策 ー部分的に XDDP の仕組みを取り入れた設計書の提案ー」
  - 「XDDP によるデグレード防止効果の検証とその効果を高めるための方法」
- 演習コースⅠ ソフトウェア工学の基礎
  - 主査：鷺崎 弘宜(早稲田大学)、  
副主査：猪塚 修(横河ソリューションズ(株))、アドバイザー：野中 誠(東洋大学)
- 演習コースⅡ ソフトウェアテスト演習コース
  - 主査：堀田 文明(デバッグ工学研究所)、副主査：小池 利和(ヤマハ株式会社)
- 特別コース ソフトウェア品質保証の基礎
  - 主査：池田 浩明((株)インテック)、  
副主査：真野 俊樹(SQA 総合研究所)、藤原 雅明(東芝ソリューション(株))





図 2: SWEBOK2004 における知識領域と分科会・コースの主な対応

具体的には、2009 年度に引き続いて、ニーズがあり社会要請の強い領域について研究・実践型の 5 分科会、ならびに、調査型の 2 コースを設けました。さらに、研究・実践型について、開発のあらゆる段階で有効であり、近年あらためてその重要性和難しさが再認識されつつあるレビューの分科会を新規に設けました。さらに調査型については、ソフトウェアテストの重要性が増大し学習のニーズが高まっているため、2009 年度においてテスト分科会内に設けていたテスト演習コースを独立したコースとして設置し、その指導体制を強化しました。

各分科会では、参加者が抱える問題や取り組みたい技術を研究テーマとして設定し、そのテーマについて 1 年間かけて調査、研究、議論を重ねて最終的に成果論文の形へとまとめていきます。典型的には、各分科会で同様の問題意識を持った数名のグループが作られ、そのグループ単位で主査や副主査・アドバイザーの指導・助言のもと、背景や関連技術の調査、問題の本質の把握、解決に有効な技術や考え方の考案、実験などを通じた検証、論文の執筆、成果発表会における発表とフィードバック、という具合に進めていきます。2010 年度は、各分科会で上記リスト中のテーマを取り上げました。2004 年度から現在までの各分科会の成果論文の内容は、研究会の Web ページ[4]に公開されていますのでご一読ください (2010 年度の論文は 2011 年 3 月以降公開予定)。

## ◇SQiP シンポジウム: 最新状況調査、交流、発信へ

ソフトウェア品質関係の技術や知識体系の最新研究・実践動向の把握、関連研究の効率的な把握、さらには研究会を超えた品質分野における幅広い交流促進のため、日本科学技術連盟が毎年主催するソフトウェア品質に関する大規模な会議「SQiP シンポジウム」への参加を研究会活動の 1 つと位置付けて、その参加と調査・交流を促進しました。また、研究や実践・調査の成果が翌年の同シンポジウムにおいて発表されることで、フィードバックを得て成果をさらに高められること、ならびに、良い成果を広く共有し業界・社会貢献へと繋がるのが期待されています。実際に、2010 年の SQiP シンポジウムでは 2009 年度の研究会成果の報告が複数ありました。

さらに 2010 年度の SQiP シンポジウムにおいては、各分科会の主査や副主査あるいはアドバイザーが外部の方々と協力して、各分科会に関係したテーマについて少人数で議論する 5 つの SIG (Special Interest Group: 意見交換会) を設けました。具体的なテーマはプロセス改善、レビュー、テスト、派生開発であり、それぞれ多数の参加者を集めて盛況でした。この SIG における議論や参加がきっかけとなり、今後の研究会における分科会活動がより広がり深まることが期待されます。

## ■分科会・コース活動の一コマ

研究・実践型の分科会の様子として (図 3 参照)、2010 年度の第 1 分科会「ソフトウェアプロセス評価・改善」における活動を以下に紹介します。

第 1 分科会ではプロセス評価・改善に関する研究を行っています。毎年継続して参加されるメンバーも多く本質的で実践的な議論が活発に行われています。プロセス改善というとレベル達成に目が行きがちですが、2010 年度の研究は「プロセス品質と製品品質の相関」という議論から始まりました。どういったプロセス改善を実施する(実施しない)と製品品質にどのような影響を及ぼすのかという研究です。

メンバーのみなさまは、この研究成果を自社に持ち帰り、「プロセス改善を始めるきっかけとする」、「重要性を現場や経営層に説明する」、「不具合分析結果から開発プロセスのさらなる最適化を行う」など、それぞれの職場の課題にあわせて活用されています。このような研究成果と共に、社外の方々や専門家と議論と研究を重ねた「経験や人脈」も、参加メンバーそして所属する企業の品質向上に活かせるよう心がけています。

さらに調査型のコースの様子として、2010 年度の演習コース「ソフトウェア工学の基礎」の活動を紹介します。同コースは、主に演習と議論を通じてソフトウェア工学技術群を習得する場として 2005 年度より継続して設置されています。産学両面に通じた講師をお招きし、全 10 回にわたり代表的なソフトウェア工学技術に関する講義と演習を実施しました。結果として、2010 年度において既に多数の技術について活用が始められており、単なる習得にとどまらない実践と将来の実施・改善基盤の形成について一定の達成をみえています。例えば参加者は、レビューの演習において習得した観点・品質特性に着目したレビューチェック項目の洗い出しを実践し、その効果を確認しつつあります。



図 3: 分科会・コース活動の様子

## ■おわりに

2010 年度の参加者からは 1 年間の活動を通じて、特別講義について「実例を用いており分かり易く社内でも広めたい」「別業種の話聞いて有益であり自社の品質管理を考えなおす必要を感じた」といった感想、分科会・コースの活動について「グループワークの中で自ら考え、かつ他者の意見を聞くこともでき良かった」「皆似た課題を抱えていることを知り交流・情報交換できた」「現場で使える内容が多く有意義」「今後も積極的に参加し成果が上がるようにしたい」といった感想が数多く寄せられています。ソフトウェア品質について基礎理論を習得し、第一線のエキスパートから助言を得て、同じフィールドで悩む仲間と多様な事例や視点を共有しつつ共に考え抜き、問題や解決の本質を見極める場として機能したと評価しています。

2011 年度は秋山浩一新委員長のもとで、特別講義を含む様々な機会を通じた理論習得と事例接触、主査・副主査の丁寧な指導、小規模な分科会単位での密な議論と気づきの共有を通じて、考えて実践する 1 年間をご提供します。分科会・コースの構成として、参加者の要求や社会要請の変化に応えるべく、新たに形式手法の演習コースを設けるなど充実を図っています。ソフトウェア品質にお悩みの方、さらなる高みを追求したい方、組織へと戦略的に展開したい方、まったく初めてという方、ぜひご参加ください。

また、研究会の活動紹介、および、ソフトウェア品質における最先端の技術動向解説や人材育成に関する提言を兼ねて、2010 年度の優れた研究成果事例の解説や指導陣の講演・パネル討論からなる SQiP ミニシンポジウムを 2011 年 3 月に初めて開催予定です[5]。こちらもぜひ合わせてご参加ください。

**謝辞** 研究員（参加者）、各分科会・コースの主査、副主査、アドバイザー、特別講義の講師、各分科会・コース内の講師、SQiP シンポジウム委員会、SQiP ステアリング委員会、ならびに、SQiP 研究会の事務局各位の熱心かつ的確なご参画・運営に感謝します。

## 参考文献

- [1] 第 29 回 SQiP シンポジウム(ソフトウェア品質シンポジウム)2010,  
<http://www.juse.or.jp/software/173/>
- [2] 2010 年度ソフトウェア品質管理研究会 特別講義,  
<http://www.juse.or.jp/software/205/>
- [3] ISO/IEC/JTC1/SC7: ISO/IEC TR 19759:2005, Software Engineering  
- Guide to the Software Engineering Body of Knowledge (SWEBOK), ANSI , 2007.  
(最新版は <http://www.swebok.org/> より取得可能)  
(邦訳) 松本吉弘 監訳, ソフトウェアエンジニアリング基礎知識体系  
—SWEBOK 2004—, オーム社, 2005.
- [4] ソフトウェア品質管理研究会,  
<http://www.juse.or.jp/software/study.html>
- [5] SQiP 研究会ミニシンポジウム: 特別企画! ~これからのソフトウェア品質技術者の育成~,  
<http://www.juse.or.jp/software/301/>

## プロフィール

鷲崎 弘宜 (わしざき ひろのり)

早稲田大学理工学術院准教授、国立情報学研究所客員准教授。博士 (情報科学)。

再利用と品質保証を中心としたソフトウェア工学の研究と教育に従事。

他の活動に情報処理学会代表会員、日科技連 SQiP 研究会運営小委員会委員長など。

著書に『ソフトウェアパターン』(近代科学社)、『ソフトウェア品質知識体系ガイド SQuBOK Guide』  
(オーム社、分担) など。訳書に『演習で学ぶソフトウェアメトリクスの基礎』(日経 BP) など。

04 年 JSSST 高橋奨励賞、06 年 SES 優秀論文賞、08 年 IPSJ 山下記念研究賞、船井情報科学奨励賞、  
日経品質管理文献賞、09 年 ASTER 善吾賞、FIT ベストペーパー賞。

## 5. 健康

### ■ 困難に立ち向かっているけれど幸せ？

#### ～「ポジティブ心理学」の幸せ観と働き方～ (1/4)

一般社団法人ポジティブイノベーションセンター

代表理事 渡辺 誠

我々の幸せってなんだろう？

我々働く人の幸せってなんだろう？

楽に働けて、いい給料をもらって、与えられた仕事はさっさと済ませて、自分のプライベートの時間を楽しむ。

プライベートでは、素敵なお子さんや旦那さんと、景色のよい明るく大きな家に住み、ゆったりと時間を過ごす。二人でいい車に乗って温泉に出かけ、ゆったりとした時間の流れを味わう。スポーツをして、汗をながし、心地よい疲れに身をまかす。お金はある。時には二人一緒にビジネスクラスで海外旅行に出かけ、高級ホテルに泊まり、大きな自然の風景やその土地の歴史的な建物に囲まれて、おいしいものをたべ、見たこともない文化や芸術を楽しむ。

優雅な時間だ。何も困らずに、何も悩まずに生活していく。そして、何も努力しなくても、何も苦労しなくてもいい。苦しいことなんかどこにもない。なんとなく憧れているそんな生活。幸せってこれ以外にないのだろうか？このような生活をずーっとしていられたら本当に幸せなのだろうか？

人の生活は仕事が順調な時もあり、難度の高い仕事に悩むときもある。人間関係でもお互いが分かりあえる時もあり、うまくかみ合わないで悩むときもある。生活のもろもろの事がうまく行くときもあり、うまくいかないときもある。これが我々の毎日の生活や仕事ではないだろうか？この毎日をイキイキ生きていくためにはどんなことを考えて、行動したら良いのだろうか？

そんな、人の幸せを研究する学問が生まれている。「ポジティブ心理学」と言われている心理学だ。ここには多くの学者が集まり、普通の人より良い幸せな生活をするための研究や、人の能力を最大限に発揮するための科学的な研究がおこなわれている。

ここでは、その研究の一端を紹介し、我々が幸せになるために何をしたらよいのかを考えてみよう。

「ポジティブ心理学」はマーティン・セリグマンが、1998年元旦に心理学の新しい方向性を示したことから始まる。アメリカ心理学会の会長に就任した時の事である。セリグマンは、「私の心理学としての仕事はまだ半分しか終わっていない。(心の病気で)マイナスになった人をゼロに戻すだけでは十分でない。我々は人がイキイキと生きるために心理学で何が出来るかを問いかけなければならない。どうしたら(人を)ゼロからプラス5にできるか？人が満たされて、夢中になれて、意味がある、充実した生活をもたらすためにはどんな活動がよいか？そんなことを問いかけていきたい。」



こんなことから、幸せの研究であるポジティブ心理学が始まった。  
そこにはイリノイ大学のエド・ディエナー教授も加わった。ディエナー教授は主観的満足度を幸せとしている。自分が自分自身の判断で満足できると思えることが幸せなのだ。人はどうしたら自分の人生に満足できるかを 20 年以上も探求してきた。科学的な実証をして多くの研究成果を出してきた。幸せ博士と言われて いる人だ。



写真 1. エド・ディエナー教授と筆者

エド・ディエナー教授はこんな話をしている。

「シンデレラの話を知っていますか？ 継母にいじめられ、貧しい服しか着られなかったシンデレラが、魔法使いのおかげで、お城のパーティに参加することが出来た。王子と巡り合い、プリンセスになった。さて、その後のシンデレラはどう生きるのでしょうか。ストーリーを作ってみてください。その後、シンデレラは、一生幸せで生きていけるのでしょうか？」

ディエナー教授はいう。「我々が生きていくうえでバラ色の生活だけが満足感をもたらすのではない。基本的なニーズが満たされるまではお金や富は大切だが、ある程度のニーズが満たされるとお金や富では人の満足度を増やすことができない。頭のよい人が幸せをつかむように思えるが、IQ が高いことが幸せと言い切れるほど関係性が無い。教育程度も同様だ。大学院卒が低い学歴の人に比べて幸せかというと調べてみると大きな違いが見つからない。年を取った人から見ると若さはうらやましいが、若い人と年をとった人の幸せ度を統計的に比べると、むしろ、年を取った人のほうが幸せ度は高い。」

どうやら、バラ色でなければ幸せでないという考えを変えてみる必要がありそうだ。幸せ博士はいう。「幸せは、アップダウンのある毎日の生活にある。幸せは結果ではなく、プロセスにある。業務課題に取り組み、一所懸命考え、解決しようとしているときのときのプロセスの中に幸せがある。難しい課題を何とか克服しようとして夢中でやっている。その時点では無我夢中かもしれないが、そんなプロセスにも充実感を感じたことがないだろうか。人生はつづく。何かを達成した止まった状態でバラ色になっているときだけではなく、ダイナミックに動いている日々の活動の中に幸せや満足感や充実感があるのだ。」

そんなことを聞いて私は考えた。「え?? バラ色じゃなくてもいいんだ。私だって、アップダウンしているよ。難しい課題を与えられて、ヒーヒー言いながら頑張っているときもある。好きな仕事を夢中ですることもある。じゃあ、私にも幸せがあるかもしれない。そういえば、活動中に充実感や、やりが



いを感じていることもある。確かに自分では満足しているな…。これって、幸せなの？」

心地よいだけでなく、充実感がある、達成感がある、和やかな気分になれるなど、ポジティブな感情が増えた時に満足度が向上する。それを幸せと言ってもいいだろう。

ディエナーは数々の研究調査から人生の満足度を向上する4つのポイントを教えてくれている。それが幸せのダイヤモンドだ。

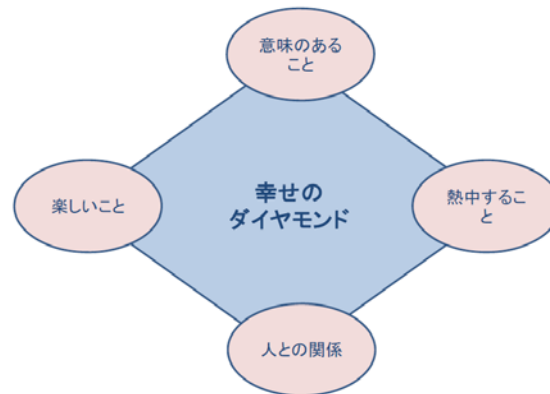


図1. 幸せのダイヤモンド

このダイヤモンドを仕事の場面に当てはめて考えてみよう。質問の形で提供するので是非ノートと筆記具を用意して回答してみしてほしい。

## 1、楽しいこと

(ア) 仕事をしていて楽しいことは何ですか？

どんな喜びがありますか？

その喜びの原点はどこにありますか？

(イ) 楽しいことを掴み取るために、今、どんな努力していますか？

そのチャレンジの中にあえて喜びを見つけるとしたらどんな喜びがありますか？

## 2、熱中すること

(ア) どんな業務をしているときに熱中しますか？

その業務は簡単な業務ですか？

それとも努力や工夫が必要な業務ですか？

(イ) チャレンジをしているときに自分はどう感じていますか？

ネガティブな感情を持っていますか？

ポジティブな感情でしょうか？

(ウ) その業務をしているときにはどんな才能を使っていますか？

その才能は強みと言えますか？

その「強み」を使っているときにエネルギーを感じていますか？

その強みはほかに使える場面はありませんか？

## 3、意味のあること

(ア) 今やっている業務やプロジェクトは、社会に、顧客に、社内にどのような貢献ができますか？

そして、あなたはその業務をすることによりどのような意味を見いだせますか？

(イ) これから先に意味のある仕事をするとしたらどんな業務ですか？

どのような未来の仕事をしたら、先ほど考えた強みを活用できそうですか？

## 4、人との関係

(ア) 業務で関わる人の人脈マップを作ってみてください。

その中で、自分が大切にしたい人はだれでしょうか？

その人と良い関係を作り上げるために何が出来るのでしょうか？

(イ) 業務以外で関わっている人の人脈マップも作ってみてください。

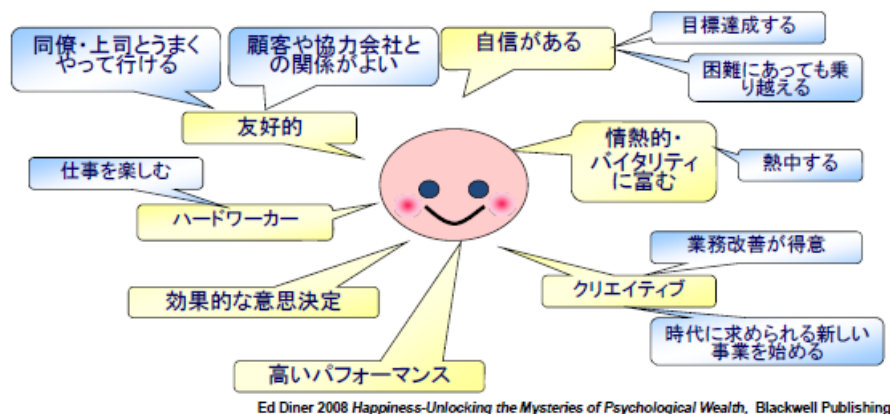
これからの仕事を考えると誰との関係を強化していけたらよいのでしょうか？

その人たちと未来の話をするとどんなことが起きるのでしょうか？

さあ、質問に答えてみると今どんな気持ちになれているだろうか？今の自分自身の満足度を確認してみよう。そして、幸せに向けて努力を始めよう。自分の在りたい姿に向かって、一生懸命努力や工夫を重ねるのだ。目標に基づき歩む、その努力そのものに幸せが宿ってくる。

最後に、ディエナーによるとポジティブで満足して仕事をする人たちは次のような特徴を持っている。

## ポジティブで仕事満足度が高い社員の特徴



### 2. ポジティブで仕事満足度が高い社員の特徴

ポジティブで仕事の満足度が高い人は、自信があり、情熱的でバイタリティに富み、クリエイティブな人である。人のことを大事にする友好的な人でありながらハードワークをする。仕事を楽しんでいるので、精神的に充実している。そして、最後に高いパフォーマンスをあげる。そんな人がポジティブでハッピーな人なのだ。

この絵を見てどう思えるだろうか？一言で言うと、仕事ができる人ではないだろうか？会社に貢献をしている人だ。あなたも私もそんな人になっていける。自分の仕事で満足度を上げることで会社に貢献できる。そんなできる人の仲間入りだ。

## プロフィール



渡辺 誠 (わたなべまこと) (Max 渡辺)

一般社団法人ポジティブイノベーションセンター代表理事

サクセスポイント(株) 代表取締役

富士ゼロックス(株)・富士ゼロックス総合教育研究所にて営業教育・人事教育を企画・開発・実施。人の成長に貢献する事を夢にサクセスポイント(株)と一般社団法人ポジティブイノベーションセンターを設立。ポジティブ心理学を応用し、働く人々がイキイキと最大限の機能を発揮できるようになり、組織に高いパフォーマンスをもたらす事を使命に活動している。国内外のポジティブ心理学者のネットワークも広く、国際的なネットワークの中で新しい手法の研究や開発に取り組んでいる。

一般社団法人ポジティブイノベーションセンターの詳細は以下を参照

<http://positiveinnovation.org/>