

SEC journal

50

巻頭言

渡邊 昇治 経済産業省 商務情報政策局 総務課長

所長対談

**IoT時代のソフトウェア・
エンジニアリングについて考える**

片山 卓也 北陸先端科学技術大学院大学/東京工業大学 名誉教授

特集

創刊50号記念

情報システムの本格利活用時代の到来

産学官連携の基盤となるソフトウェア工学雑誌SEC journal
— 50号記念に寄せて —

企業から見たSEC journal、SEC journal論文

SEC journal (論文)の見え方

SEC journal 50号の歩み

論文

定性的信頼性/安全性解析支援ツールの開発

土肥 正 広島大学/岡村 寛之 広島大学

NE比を活用した、 つながるシステムにおける利用時の品質向上の提案

神田 周一 株式会社U'eyes Design / 諸熊 浩人 株式会社U'eyes Design

入江 哲 有限会社 エム・エス・エス / 伊藤 潤 UX測研

オープンシステム・ディペンダビリティのための 形式アシュランスケース・フレームワーク(FFO)

奥野 康二 神奈川大学プログラミング科学研究所 / 木下 修司 神奈川大学プログラミング科学研究所

木下 佳樹 神奈川大学プログラミング科学研究所 / 神奈川大学理学部情報科学科

武山 誠 神奈川大学プログラミング科学研究所 / 神奈川大学理学部情報科学科

中原 早生 神奈川大学プログラミング科学研究所 / 神奈川大学理学部情報科学科

連載

情報システムの障害状況2017年前半データ

松田 晃一 IPA顧問 / 目黒 達生 SEC研究員

1

巻頭言

創刊50号に寄せて

渡邊 昇治 経済産業省 商務情報政策局 総務課長

2

所長対談

IoT時代のソフトウェア・エンジニアリングについて考える

片山 卓也 北陸先端科学技術大学院大学/東京工業大学 名誉教授

8

特集：創刊50号記念

情報システムの本格利活用時代の到来

有賀 貞一 AITコンサルティング株式会社 代表取締役/経済産業省産業構造審議会情報経済小委員会 委員/日本イノベーション融合学会 会長

産学官連携の基盤となるソフトウェア工学雑誌SEC journal

— 50号記念に寄せて —

井上 克郎 大阪大学 大学院情報科学研究科/産業技術総合研究所 情報技術研究部門

企業から見たSEC journal、SEC journal論文

久野 倫義 三菱電機株式会社 設計システム技術センター ソフトウェア技術推進部

SEC journal (論文) の見え方

斎藤 忍 NTT ソフトウェアイノベーションセンター

SEC journal 50号の歩み

SEC journal編集部

28

論文

定性的信頼性/安全性解析支援ツールの開発

土肥 正 広島大学/岡村 寛之 広島大学

NE比を活用した、つながるシステムにおける利用時の品質向上の提案

神田 周一 株式会社U'eyes Design / 諸熊 浩人 株式会社U'eyes Design

入江 哲 有限会社 エム・エス・エス/伊藤 潤 UX測研

オープンシステム・ディペンダビリティのための 形式アシュランスケース・フレームワーク(FFO)

奥野 康二 神奈川大学プログラミング科学研究所/木下 修司 神奈川大学プログラミング科学研究所

木下 佳樹 神奈川大学プログラミング科学研究所/神奈川大学理学部情報科学科

武山 誠 神奈川大学プログラミング科学研究所/神奈川大学理学部情報科学科

中原 早生 神奈川大学プログラミング科学研究所/神奈川大学理学部情報科学科

52

連載

情報システムの障害状況2017年前半データ

松田 晃一 IPA顧問/目黒 達生 SEC研究員

59

書籍紹介

60

編集後記

創刊50号に寄せて



渡邊 昇治

経済産業省 商務情報政策局 総務課長

創刊50号、お慶び申し上げます。ITは技術進歩やビジネスモデルの変化が激しく、ともすると継続の大切さを忘れてしまいそうですが、多くの物事に関しては継続が重要だと思います。

せっかくの機会ですので、私が発見した「経産省20年の法則」を紹介します。経産省(当時は通産省)がサンシャイン計画を開始したのは1974年、住宅用太陽光発電システムが発売されたのは1992年でした。ムーンライト計画は1978年にスタートし、日本の企業が燃料電池車を発表したのは1997年、家庭用燃料電池を産業化したのは2008年でした。IT分野では、1973年に自動車総合管制システムの研究が始まり、VICISは1996年にサービスインしました。ITS・自動走行には1995年頃から本格的に取り組み、2015年頃から自動ブレーキなどの技術が次々と製品化されています。このように、経産省が研究に着手してから産業化が始まるまでには20年くらいの期間があります。あきらめずに研究開発を続けることが重要です。

「経産省20年の法則」は、経産省は目を付けるのが早過ぎるという意味も含みますが、これは、言い方を変えると、私達の先人が未来を正確に読んでいたということであり、あらためて先輩方を尊敬します。

以上まとめると、大切なことは「先見の明」と「継続」です。近年多くの製品・システムにおいて、ソフトウェアへの依存度は高まる一方で、ソフトウェアの信頼性はますます重要になっています。また、多くの製品がネットワークに接続し、かつAI(人工知能)を搭載するようになるため、ネットワーク、セキュリティ、AIなど、ソフトウェア技術者が習得すべき事項は増えると予想されます。このような中、IPAによる各種ガイドラインの策定などの活動は有意義です。2004年にSEC(現ソフトウェア高信頼化センター)を設立した先見の明と継続努力には敬意を表します。

継続という点では、近々50周年を迎える国家試験「情報処理技術者試験」、間もなく20年の歴史を刻む「ITスキル標準」「未踏事業」など、IPAが実施してきた人材関連の事業には、どれも実績の重みがあります。

一方で、地域、人材、サイバーセキュリティなどの新たな課題に対するIPAの事業にも期待しています。「J-CSIP(サイバー情報共有イニシアティブ)」「J-CRAT(サイバーレスキュー隊)」、サイバーセキュリティに関する高度な国家資格「情報処理安全確保支援士(RISS)」、ハイレベルな実践的人材を育成する「産業サイバーセキュリティセンター」などには先見の明があり、前述の事業同様、10年、20年続いていくと確信しています。

経産省は2017年7月から商務情報政策局を、IT、クールジャパン、サービス、ヘルスケアなどの政策を扱う局から、IT政策に特化した局へと装いを変えました。コンパクトになることで、より強力に重点的に政策を進める考えです。

IoTは日本にとってチャンスです。エッジ(端末)側に情報処理機能を持たせる「分散戦略」は日本のチャンスを拡大します。経産省はIoT推進ラボなどを通じて、先導的な実証試験や先進的な企業を支援していますが、この成果を実用化につなげたいと考えています。

これからはデータの利活用が競争力を左右します。人材育成、研究開発、標準化、サイバーセキュリティなどの基盤的な取り組みを進めつつ、様々なデータを結び付けて活用し、予防医療・予防介護、ロボット・AIの導入、技能伝承、交通安全、物流効率化など、社会課題の解決と産業競争力の強化を図っていききたいという思いで、経産省は産学官連携により「Connected Industries」を推進していきます。

IoT時代のソフトウェア・エンジニアリングについて考える

北陸先端科学技術大学院大学／東京工業大学
名誉教授

片山 卓也

IPA/SEC所長

松本 隆明

ソフトウェア開発力の強化のためソフトウェア・エンジニアリングの啓発と、実践に資するべく発行してきたSEC journalも、本号で50号を迎えた。これまで本誌が果たしてきた役割や今後の方向性、更にIoTの本格的展開が期待される現在にふさわしい先進的なソフトウェア・エンジニアリングのあり方について、SEC journal論文賞表彰委員長でもある片山卓也先生にお話を伺った。

SEC journalが果たしてきた役割

松本 50号を迎えた本誌ですが、昨年度までの論文の採録数は68件となっています。数は決して多いとは言えないかもしれませんが、SEC journalが果たしてきた役割は小さくないと思っ

ています。片山先生はどう見られていますか。

片山 大きな意味があったと思っています。実践的な論文を載せるアカデミックな場所というのは決して多くありません。どうしても新規性や独創性が評価の基準になってしまうので、原理は分かっているがプラクティスに意味があるというような論文を載せる場所は限られています。業界誌がたくさんあった頃はそれなりに投稿場所もあったけれど、今はその数も減りました。国のソフトウェアの政策に責任を持つSECのような機関が場所を提供していることは意義深いと思います。

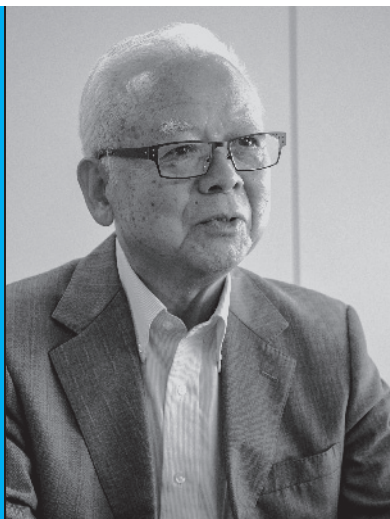
松本 ありがとうございます。確かに、ソフトウェア・エンジニアリングに範囲を絞ったプラクティス論文誌は、SEC journal以外にはないように思います。論文の

質という点では、どう評価されていますか？

片山 それなりにきちんとしたものであると思います。注文も付けましたが、それは質が良い悪いということではなくて、どの辺りの論文を採るかということなのです。私としては、試しにやってみたというものが載っていて、業界における技術導入に道を拓くような論文がたくさんあって欲しいと思っていたのですが、実際掲載されている論文は、ある程度評価が決まっているものを取り上げたものが多かったように思います。逆に言えば、新しいソフトウェアの作り方を巡って、試しにこんな新しいツールを使ってみたらできました、というようなものが少なかったのではないのでしょうか。これは論文の集め方というか、テクニックの問題もあるのではないかと思います。応募を待って審査をするというスタイルでは、発掘できない。業界の人は基本的に忙しく、しかも新しいことに取り組んでいるような人は、ほかにもたくさん仕事を担っていることが多く、論文を書く時間がないという状況にあります。新しい試みをSEC journalで紹介するためには、こちらから依頼をして書いてもらうなど、論文を集めるための努力も必要になるでしょう。いわば“枯れた技術”であれば書く手法も決まっていて、書きやすいし、集まりやすい。しかし、例えば以前クラウドが出てきた当初は、まだ細部では分からないことがたくさんあった時期で、産業界でも色々なことを試しているグループがありました。しかし、そういう話がSEC journalにはあまり出てこなかった。それは残念だと思います。

松本 産業界からの応募をもっと増やしていく必要があるということでしょうか？

片山 産学連携を一生懸命にやっている大学から、というのもあると思いますが、大学の人が産業界のことを想像して書いたものは、その中には将来的に非常に面白いものになるアイデアが含まれている可能性はありますが、しかしそれはアカデミックなジャーナルで拾えば良いわけです。やはり産業界に根ざして、産業界の難しい問題を新しい手法で解いたものが欲しいですね。



片山 卓也 (かたやま たくや)

東工大電気工学科卒業(1962)、同情報工学科助教授(1974)、教授(1985)、北陸先端大情報科学研究科教授(1991)、学長(2008)、中央大研究開発機構教授(2014)、日本ソフトウェア科学会理事長(1991)、名誉会員、情報処理学会理事(1985)、名誉会員、専門：形式的ソフトウェア開発方法論、高信頼組込みシステム、ソフトウェア発展・進化機構、法令工学

それを書けるのは産業界の人か、産業界と密な連携をしている大学や研究機関の人だと思います。SEC journalは、そういう人の論文を集めよう、題材は産業界にある、というところからスタートしました。その方針は正しかったと思います。

松本 ただ産業界の場合だと、先生もおっしゃったようにみなさん忙しいですね。

片山 忙しいことに加えて、あまり価値のないものまで秘密にする傾向がありますね。

松本 新しい技術を製品に適用しました、ということになると企業として守秘上それを表に出しにくいという面もあるのでしょうか。

片山 それを出したからといって企業が困るということはないと思います。かえって、あの会社はこんな素晴らしいテクノロジーを使っているということで、評価は上がると思います。秘密にしておく価値はあまりないでしょう。

プラクティスにかかわるホットな報告が欲しい

松本 企業の方から研究成果を発表していただくとする場合、論文の書き方に慣れていないという問題もあります。

片山 そこは査読する人が何か言えばそれで解決するでしょうし、多少書き方が悪くても構わないと思います。いわゆるアカデミックな論文の書き方に則って、まず背景を述べ、その中で自分はこれをこう解決した、そのどこが新しく、ほかとの比較ではどういう意義があるのか、といったオーソドックスな書き方は難しい。ほかとの比較といっても、そもそも他社がどうやっているかなんて知らない場合があるでしょう。とくに産業界にいる場合は、他社と比較して変なことを言ったら失礼になるかもしれません。ですからそこは緩やかにして、何をやったかということを中心にすれば良いのではないのでしょうか。論文の価値の最も重要なところは、アカデミックなジャーナルの場合は明らかに独創性であり、何かを新発見したということでしょう。一方業界に関する話というのは、新しい発想というよりは、既存の技術や知見を新たに適用したということにあると思います。場合によっては二番煎じでも良く、追試を行ったらうまくいったというのでも良い。独創性を評価の主眼に置くことは少し問題だと思います。

松本 なるほど。ただ、もう少し最低限のことは書いて欲しいと感じる論文もあることは事実です。単にやったことを書いただけでは、ほかの人にとっては役に立たない。どのような課題があってどれだけ苦労したか、実行の過程において何を工夫したのかということは明確にして欲しいですね。

片山 確かにそこは要求すべきでしょう。しかし論文としての形の素晴らしさではなく、適用した中身の素晴らしさを第一に見ていきたい。外見は悪くても中身のおいしいものを採るということをするれば、論文を読んだ人が同じようなことをやってみようとしたときの参考になると思います。

松本 プラクティスなのだから、「やってみよう」というところが重要ですね。そうでないと産業界に広がっていかないわけですから。

片山 産業界からのデータを分析してモデルを作ったとか、産業界でよくやっていることを厳密に適用して評価してみたとい

うものなど、最初から結果がある程度分かっているような論文が多かったような気がします。もう少しホットで今話題になっている手法を適用してみるとか、そういうほうが読んでいても面白いですよ。

松本 両方あるのかもしれないですね。既存技術に近いところだけけれど、例えば人材育成に関する論文がそうですが、それほど目新しいことをやっているわけではない。しかし、企業ではソフトウェア技術者をどう育成していくかということについて非常に苦労されていて、その事例は大いに参考になります。

片山 確かに人材育成については、優秀論文に残ったものもありましたね。

松本 それと同時に、全く新しいことに取り組んでみてこうでした、という論文もあって欲しいですね。

片山 そうですね。いずれにしても、SEC journalのような雑誌は世界的に見てもそうあるわけではない。今後もぜひ活用していただきたいと思います。

来るものを待つのではなく書いてもらう

松本 論文の投稿を増やすという意味では、SEC journalに論文を載せることが学位取得のポイントになるという大学も幾つかあるので、そういう大学をもっと増やしていったらどうかとも考えています。

片山 なるほど。しかしそれは難しそうですね。学位の基準は大学によって違うし、その基準は大学のポリシーを表しています。ですから、そういう論文が載るようなジャーナルになって欲しいとは思いますが、それなりに色々なことをやらないといけません。例えば、ソフトウェア・エンジニアリングで言うとInternational Conference on Software Engineering (ICSE)でもインダストリペーパーはたくさん採ろうとしていて、レベルもなかなか高い。ですから、覚悟を決めてそれと競うような内容にしておく必要があります。そのためにも開発現場に声をかけて良い論文を集め、相乗効果で全体のレベルも上げていく必要があります。

松本 ただ投稿されたものだけでやってもたぶんレベルは上がらないということですね。

片山 論文の査読をする人の意識や取り組み方という問題もありますね。実際そ



松本 隆明 (まつもと たかあき)

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

のことに不安を抱かせるような論文も幾つかあったように記憶しています。良いカンファレンスの査読者になるということはそれ自体名誉なことであって、名前も公表されます。そこに入れるかどうかは自分の実力の証になる。ですから、査読をするときもかなりしっかり調べて、この論文が良いか悪いかを委員会で論じることが求められます。全員で議論しながらカンファレンスを作るので、査読者もしっかり取り組みます。いずれにしても、評価が高まれば論文も査読も相乗効果でレベルが上がっていくと思います。

もともと日本の業界は一生懸命やっても表に出さないですから、お願いして書いてもらう。招待論文でも構わないから、これはすごいなというものが毎号1本は載っているというようになれば、みるみる良くなっていくと思います。例えば、SECで新しい先進技術の適用事例の「事例集」を作っていますよね。ああいうところに出てくるようなものを、こちらからお願いして長めのものを書いてもらうのはいかがでしょうか。学者であれば良いジャーナルに掲載されることが学位や評価の基準になりますし、名誉になる。しかし業界の人にとってはそれほど名誉ではない。だから待っていても論文は投稿されない。良いことをやっている人を見つけてお願いして書いてもらう、それをやれば良い論文が投稿されるようになり、執筆者の社内評価も高まり、良い循環になっていくのではないのでしょうか。今でもサーベイなどについてはみなさんある程度参考にしていただろう。更に先進的な事例などについても誰かに書いてもらう。SEC journalに招待されて論文を掲載することが名誉なんだということになれば、投稿論文のクオリティも上がっていくでしょう。

再び注目を集めるプログラムの世界

松本 ソフトウェア・エンジニアリングという言葉が出てきたのは1968年のNATOの会議だと言われ、もう50年になるわけですが、この50年間のソフトウェア・エンジニアリングの進歩ということについて、どうお考えですか？

片山 明らかに色々なことが進歩して、本当にエンジニアリングの分野になりましたね。NATOの会議の頃は、大規模プログラムの時代です。私も興味を持って当時の論文を読んでみましたが、大規模プログラム開発の失敗が相次ぎどうしようかということで、当時のコンピューターに関するそうそうたる研究者が会議を開いて、今で言うソフトウェア・エンジニアリングからはもう少し広い、プログラミングとか言語なども含めたカンファレンスだったようです。エドガー・ダイクストラ(Edsger Dijkstra)の「Go To文は有害である」(“GoTo Statement Considered Harmful”)という論文などが発表された時期で、構造化プログラミングや制御構造のハイラーキーの話とか、検証に関する不変表明法などが議論された時代でした。ICSEもその頃にできました。ICSEの論文で見ていくと、その後、今から30年くらい前からソフトウェアの開発プロセスやプロジェクト管理の話が盛り上がった時期が長く続きましたが、その後またプロダクトに関する話が多くなり、プログラムやソフトウェアの解析や構造化に関する論文が多くなったという印象です。それはなぜかと言うと、大きなものを間違いなく作るにはプログラムが分析できなければいけ

ないからです。それからメンテナンスの問題があります。メンテナンスの問題は泥臭い問題だと誰もが考えていると思いますが、メンテナンスに関する国際会議の論文集などを見ると、プログラムの解析の話が随分多いという印象です。ツールを使っているに危ないところを見つかるかとか、波及分析をどうするかという話で、色々な技術が進んでプログラムやソフトウェアを分析できるツールが出てきたことによって、そのような技術的な話が増えているのではないのでしょうか。

松本 なるほど。いったんプロセス寄りになりマネジメントのほうに注目が集まっていたものが、またプログラミングのほうに戻ってきたということですね。

片山 要するに仕様を書くことも含めて、物事をちゃんと書かない限り分析のしようがないということで、そちらに揺り戻しているということだと思います。一時私もプロセスの研究をしましたが、研究の中心が管理系の話になり、レベル幾つだからこの会社は良いとか悪いとかいう話になった頃プロセスから離れました。最近ではアジャイルやDevOpsなど面白い話が色々あるようですね。

松本 プロセスに関しては最近IoTの大きな流れもあり、きちり上流から固めていくというスタイルではソフトウェアが作りにくくなっています。何がつながるか分からないというように要件そのものもあいまいになっている。それならアジャイル的にとりあえずプロトタイプングでやってみて、それをブラッシュアップしていくという開発手法になっていくことが考えられるのではないのでしょうか。今までのような、がっちりした手順では立ちゆかなくなっているケースが増えているように思います。

片山 全部を決めてから作り始めるということがやりにくくなっている、それは事実ですね。もう一つは、保守の問題にかかわるけれど、何か付け加えようという話になったときに、どうやって付け加えるか、中身を全部知っている人がいるわけではないので、できているものを分析しない限りものが作れなくなってしまうわけです。従って、きちり分析できるように物事を作るとか、人間はいい加減だから機械に分析させて、おかしなところがないことをチェックするという技術がこれからもっと必要になってくると思います。IoTの環境で、作ったもの同士が、相手が分からないままつながるわけではないんです。この世界にマジックはない。論理と論理のぶつかり合いですから、きちんと分析してつなげないといけない。分析できるように物事ができていること、分析できるようなフォーマリズムで物事が書いてあるということが必要です。アジャイルやDevOpsでも同じことだと思います。作って動かしてみないと人間の想像力は働きませんから、まず動かしてみても色々なことが分かるということでしょう。そしてその次のサイクルに入るときに必要な分析を行えることが重要です。あらかじめ色々なことが分かっているものとは作り方が違うということだと思いますね。

今求められる システムズエンジニアリングとは

松本 そういう意味で、私たちは最近システムズエンジニアリングという考え方に注目しています。日本語で表現してシステ

ム工学と言ってしまうと制御系のイメージを持たれがちですが、システムの開発をなるべく抽象化したところから全体像を捉えて、それから徐々に具体化して物に落とししていくという考え方でやっついていかないと、下から個別に物を積み上げて作っていくという考え方でやったのでは、今の世の中では全体の設計ができなくなってくる。

片山 その通りですね。しかしそれがなかなかうまくいかない。

松本 なぜなのでしょう。

片山 物を作り始めると作っているところに張り付いてしまうんです。本来、そこでもう一回全体を見渡し、常に抽象的な世界を更新しながら下のほうも作って、上から下まで矛盾のない世界で物事を完結させていくことが求められるのですが、みなさん忙しすぎるから、上のほうを作ったら下に行ってしまう、後は下だけで物事をやっちゃっているということではないですか。

松本 下のほうはイメージしやすいですね。部品などは目に見えるものになっている。そこで、これとこれを組み合わせればこうなるよねと下から積み上げるビルディングブロック的に考えていくと分かりやすいですね。上から抽象的に俯瞰して考えるというのはなかなか難しい。この辺りの発想のし方から変えていかなければならないと思います。

片山 目の前の問題の解決だけを考え、そのシステムを作ることだけを考えたならあまり問題にならないのに、抽象的な上のほうのこともきっちり考えておこうということですから、無駄なことと感ずるかもしれません。しかし、そうやっておけば後は確実に楽になるのです。

松本 幅広く業界を横断して色々な物が連携し、サービスなどが展開されるようになると、一つのサービスだけに限って考えていけば良いとはいかなくなりますね。うまく連携させることができるように考えていかなければなりませんし、だんだん難しくなっています。

片山 そこで必要だと思うのは、ある程度動かせるというか、上のほうも推論ができるように作り上げておく、ということです。人間は物事を忘れてたり、そもそも全体を見通すことは不可能に近いので多くの見落としをします。見落としをしないのは機械です。機械は疲れを知らずにとことん調べてくれますから、機械に分析させれば良い。もちろんそのためには色々なことを決めなくてはなりません。スペックをロジックで書き、それをだんだん具象化して最終レベルに達したときに下にコードができる、上には論理式があるというように階層的にでき上がっている、そういう発想でパリの地下鉄のシステムを作り上げたという話があります。これくらい徹底してやれば、何かを変えるというときにも、計算機の力を借りて推論をやり直すとか、新しい要求を付け加えると、ここがバッティングするよ、というようなことを機械が調べてくれるようになる。でも、残念ながら日本にはそういう文化はないですね。ここまで徹底することが通常の商業的ソフトウェア開発で可能かどうかは分かりませんが、それでも開発や進化・変更プロセスにもっと計算機による推論能力を使っても良いのではないかと思います。計算パワー的には十分可能な時代になったと考えます。

プログラムは人間の創造性の産物

松本 正にそこはAIですよ。AIを使って自動的に学習して、ここが問題になるとか。

片山 そうですね。今で言うディープラーニングがすぐに使えるかどうかは分かりませんが。

松本 どうでしょうか。ある程度人間が抽象化したものを具現化していく作業は機械学習によってできるようになってくるのではないかと思います。

片山 ワトソンの技術なんかはそれに向いているのではないかと思います。今話題のディープラーニングは、規則としては特徴付けにくいデータ集合をニューラルネットに学習させることに成功し、碁のチャンピオンに勝ってしまうような性能を発揮しました。しかし、記述された規則や論理としてその集合を特徴付けているわけではありません。一方、プログラムやソフトウェアはそうではなくて、規則を記号として明確に書いたものであり、その学習にディープラーニングの考え方が上手く使えるかどうかは今後の研究によるところが大きいと思います。具体的な話では、プログラムエラーの膨大なデータベースから学習して、デバッグやテストを効率的に行うことが可能になると良いのですが。いずれにしろ、プログラミングやソフトウェア開発へのAI技術の適用は今後の大きなテーマだと思います。

松本 私もAIの進出でプログラマーの仕事がなくなるとは思っていない。プログラムというのはある意味では創造性の産物です。芸術に近いようなところがある。人間が創造性を発揮してプログラムを作るところに発見があるような気がします。機械で厳格にすればできるというようなものではないと思います。

片山 そうですね。

松本 私の経験でも、プログラムと言ってもある機能を実現するためだけのコードであれば、誰でも作れる可能性があると思います。その中でも本当にきれいにできているプログラムと言ったら良いのか、メンテナンスも含めて誰が見ても分かりやすいプログラムというのはすごく良いプログラムであって、それは機械では作れない。

片山 メンテナンスのしやすさとは何かということを経験が理解しなければいけないわけで、そのためには今流のやり方例えば、メンテナンスのしやすいプログラムを膨大な数を集めて機械に食べさせてみて、ということになると思うけれど、そんな単純なことではうまくいくとはとても思えません。そもそも出来の良いプログラムを大量に集めること自身が非常に困難です。もっと本質的な原理的な飛躍が必要だと思います。

松本 そういう意味では、ソフトウェア・エンジニアリングの方向性が、プログラミングのほうにシフトしつつあるということはどういうことかだと思います。ソフトウェアがこれから非常に大きな役割を果たしていく中で、プログラミングはますます重要になるのではないかと考えています。

片山 そうですよ。私は一時、JAVAが出てきた頃ですが、新しいプログラミング言語なんてもう生まれないのではと思いましたが、大きな間違いでしたね。

松本 PythonやRubyなどが出てきた。

片山 最近は百花繚乱で、日本発のRubyなどが世界的に使われ

ているのは嬉しいことです。また、関数型言語のような硬い言語が商用開発でも使われるようになりました。ただ、プログラミング言語の研究開発を継続的に行っている人が欧米に比べて少ないのは残念です。我々日本人は流行のことに流れやすいようです。

松本 ソフトウェア・エンジニアリングを専門にされている先生の数もあまり多くありませんし、昔からプログラムやプログラミングに一貫して取り組まれている先生は少ない。そこに日が当たるようなことを考えていかなければいけませんね。

片山 アメリカとの比較で言うと、もともと人口は日本の倍以上ありますが、世界中からエンジニアを吸い付けていますよね。それは世界的な企業が優秀な人を集めて仕事をしていることに加えて教育制度も大きいですね。実は近代的な大学院教育を始めたのはアメリカなんです。大学制度はヨーロッパで作られましたが、大学院はある意味でアメリカが確立したと言って良い。そこに海外から優秀な学生をたくさん引き入れて、その人たちが新しい学問を作り上げ、ITインダストリを作り上げたわけです。その点、アメリカはすごいと思いますね。

価値ベースの評価にしないと やりがいが見えてこない

松本 話は変わりますが、ソニー生命保険が今年の春インターネットで行った「日本の中高生のなりたい職業」の調査で、男子中高生のトップが「ITエンジニア・プログラマー」でした。

片山 それは素晴らしい。

松本 すごく良いことです。せっかく今の若年層がそういうことに興味を持ち始めているのだから、教育もきちんとしていかなければいけないと思います。

片山 ではもう「3K」ではなくなりましたか。

松本 中高生ですから、職場の実態をどう見ているかそれは分かりません。しかしいずれにしてもプログラミングに興味を持っているのは事実でしょう。

片山 きちんとした処遇をしていかなければなりませんね。

松本 同志社大学の中田喜文先生が行ったITエンジニアの処遇の各国比較がありますが、そこでは日本は先進国の中で最低ラインです。給与が低いだけでなく勤務時間も長いし、満足度も低い。

片山 そこをなんとかしないと中高生が育たないでしょう。

松本 せっかく興味を持っていても、実際に職場に入ってからしてしまいそうです。

片山 なぜそうなのでしょうね。

松本 産業構造そのものにも大きな問題があると思います。日本のソフトウェア業界は、プログラムの価値にお金を払うのではなくて、その作成にどの程度の工数がかかったか。何人が何日働いたかで計算している。そういう契約形態では価値ベースにはならない。そこが大きいような気がします。

片山 フランスなどもあまり高くないのではないですか？

松本 高いのはアメリカとドイツですね。日本は中国と同じ程度です。

片山 アメリカが成功した大きな理由の一つは、新しいビジネスと結び付けてITを進歩させたことにあるのではないかと思います。ああいうし方で新しい技術を作りマネジメントのやり方

も変えて会社を発展させていく。日本では難しそうですね。もっと硬くかっちりとしていて、新しいことはできないけれど安定しているというのが日本のITのシステムなのかなと思います。それはそれで社会的な価値は十分にあると思いますが。

松本 そうですね。高品質で売っていくというのも一つの方向ですし、新しいビジネス、新しいサービスで売っていくのも一つのやり方で、両方あるかなと思います。ただ、今後ダイナミックな社会になっていくと、高品質だけで売っていくのは難しくなっていくのではないかなと思います。

IoTの普及で再認識される 「形式手法」の価値

松本 これまで先生は、数学的な手法を用いて正しいソフトウェアを開発する形式手法(フォーマルメソッド)で、リジットにプログラムを作ることに取り組まれてきましたが、残念ながら形式手法はなかなか産業界に一般的な方法として普及していかないのですが、それについてはどうお考えでしょうか。

片山 本当にクリティカルな、故障したら困るところでは使わざるを得ない技術だと思います。同じものを今までと同じ方法で作れるかと言えばそうではない。必要なテストの数が膨大になり、経済的に引き合わないということになってしまう。どれだけ注意深く作るべきか、ということにかかわると思いますが、それほど注意深く作る必要がないものは今まで通りの方法で作れば良い。私は形式手法の一番の使い道は、長く使うもののメンテナンスだと思っています。ずっと使い回すために何か新しい機能を付け足すというとき、テストする人もいない、中身も分からないということになると、そのソフトウェア自体を分析できるかどうか最後の手段になります。

松本 おっしゃる通りですね。検証に近いというか。それが仕様通りになっているかということがちゃんと証明できるという仕組みを入れておかないと、後から追加したときにこの辺りに影響があったということが分からなくなってしまいます。

片山 IoTの世界に新しいものをつなぐときには、いい加減につなげば良いということではない。新しいものが矛盾なく接続されなければならない。規格も重要で、いい加減な規格のものをつなげと言われてもできない。無理は通らないんです。そのためには分析可能な形で設計書とかシステムのコードなどが書かれていなければならない。そこに形式手法の使い道があるのではないかなと思っています。間違いのないものを一から作るという意味では、人間はそれなりに注意力があり、またテストでもある程度のことは分かるから、そこそこのものは作れてしまう。しかし、作ってしまって巨大になったものがどう振る舞うかというこの細部は人間は分からないので、そこは機械にやってもらうしか手がないのです。しかし、これはなかなか理解してもらえません。

松本 SECとしてもそういう考え方を広めていかなければならないと考えています。これからますますそういう時代になってくるわけですね。あいまいな要求条件の下で作ることになるし、一度作ったものを環境が変わっても使えるようにしていかなければいけません。それをどうやって保証していくのか、ある程

度形式的に、きちんと均一化されていないとできません。

片山 やりようがないんです。

松本 試験のケースもべらぼうに増えてくるから、いちいち実地試験でやっていたら大変です。全部つながなければいけない。

片山 しかもそれぞれのシステムも、作っているときだったら、担当者があるから、その人に聞けば分かる。しかし人が去ってしまった後でつながなくてはいけないのだから、残っていて信頼できるのはコードだけです。それを分析するしか手がない。それは形式手法的なやり方でしかできません。

松本 実はSECでも昨年「システム再構築を成功に導くユーザガイド」というものを出していて、今、内容の改訂を進めようとしているところです。産業界は膨大なレガシーシステムを抱えていて、それをIoTやビッグデータなどの新たな要件にどう対応させていくかが課題になっている。一から新しく作り直すわけにはいかないので、既存のシステムをどうテラリング、あるいはチューニングしていくかということで悩んでいるわけです。まだ形式手法の適用というところまでは踏み込めていないのですが、いずれそのようなツールや手法も考えていくことになると思います。

片山 できあがった物を分析するのは非常に難しいことで、ドキュメントなども完全なものであるかどうか分からない。しかしそういうところにサイエンスやエンジニアリングを作ることが重要だと思っています。

巨大化するシステムのメンテナンスをどうするか

松本 企業から見ると、そういうところはコストに見えて投資に見えないんですね。既存のものをなんとかうまくもたせようという“メンテナンス”になってしまって、それならできるだけコストをかけずにやろうと。

片山 結局人をずっと張り付けておくだけになる。

松本 そういうケースが多いですね。しかしそういう人もいずれはリタイアします。

片山 そういう場合、企業はシステムを捨てているのでしょうか？

松本 いやそう簡単には捨てられません。最近の調査でも20年、30年と同じシステムを使っている企業が多いですね。できるだけコストを抑えて、現在あるものをだましだまし使っていくというようなケースがほとんどです。

片山 そういうことに関するエンジニアリングやサイエンスを作るという点では、SECに大きな役割がありますね。方法論も含めて展開するというのは民間企業ではできませんからね。

松本 “巨大システムのメンテナンスのためのエンジニアリングをどうするか”ということですね。そこに最新鋭のコンピューターサイエンスを使いたい。学のところでは研究は行われているのでしょうか？

片山 興味を持っている人はいると思いますが、具体的にはよく分かりません。実用的な意味合いだけでなく、学術的にも面白い問題ですので、具体的な研究プロジェクトでも始まれば、多くの研究者が集まると思います。学の世界でも研究費がえら

れるようなテーマを設定しないと博士学生や研究員などのスタッフが集められないので、どうしても時代に流されます。

今はAI関連でないと大きな研究費が取れないので、本当に必要なことを頑固にやり続けることが困難です。しかしヨーロッパやアメリカにはそのような頑固さがありますね。例えばコンパイラの研究者がアメリカには、少数ですが、まだいるでしょう。日本でコンパイラの研究をやっている人はもう産業界にも大学にもいないと思います。OSの研究者もそうではないでしょうか。

松本 確かに以前はOSの研究者がいましたが、現在はそう多くはないのではないかと思います。

片山 国民性でもありますが、重要なことは投資だと思います。メンテナンスの問題は地味ですが、国や産業界にとって大変重要ですから、それに見合った投資をして欲しいと思います。

松本 企業が投資をしにくい分野ですし、そういうことこそ産学連携で官も巻き込んでやっていかなければいけないのじゃないでしょうか。

片山 SECが中心となってやってくれると良いですね。最前線のコンピューターサイエンスやAIを駆使してメンテナンスの科学や技術体系を作るということをやればインパクトは大きいと思います。

松本 重要なテーマですね。どんどんシステムは肥大化しているし、人の力でメンテナンスしていくのは難しくなっていく一方だと思います。

今日は、メンテナンスの話をはじめ、今後考えていかなければならない貴重な問題提起をいただいたと思います。最後にSECに期待されることをぜひお伺いしたいと思います。

片山 SECは日本に一つしかないソフトウェアに関する研究開発機関であり、実用・普及に貢献する機関です。ソフトウェアが良くならなければ国も良くならない。SECの役割は非常に大きいと思います。長年内部にいらっしやると、価値が見えにくくなるかもしれませんが、国の重要なファンクションを担っていらっしやるわけですから、ぜひともがんばっていただきたいと思います。

松本 はい、しっかり進めていきたいと思っています。本日は貴重なお話をありがとうございました。



情報システムの本格利活用時代の到来

AITコンサルティング株式会社 代表取締役
経済産業省産業構造審議会情報経済小委員会 委員
日本イノベーション融合学会 会長

有賀 貞一

SEC journalが創刊されて50号になるという。筆者の、ITシステム／サービスの開発・提供側、利用者側双方の実務を経験しているバックグラウンドから、産業界の視点でSECやSEC journalに関して歴史的視点も含めて述べてみたい。要旨としては、SEC誕生の経緯、その後の事業仕分けに絡む曲折、ユーティリティ化の実現による情報システム構築の変化、クラウド時代の到来、本格IT利活用時代の到来と必要人材の変化などである。

1 SECの誕生とその後の曲折

SEC journal発行の母体となったのは、IPAのソフトウェア・エンジニアリング・センター(SEC)である。SECは2004年10月に誕生した。ちょうどこの原稿を書いている間に新たに経済産業省次官に就任された嶋田隆氏が、情報処理振興課長を務めていた2003年頃、ソフトウェアの構築をより科学的にできないかという要請が、情報サービス産業協会(JISA)をはじめとする関連業界にあった。当時JISAの理事であり、産業構造審議会情報サービス・ソフトウェア小委員会人材育成WGの主査をも務めていたため、筆者にも協力要請があった。その時点では、株式会社CSK(現株式会社SCSK)の代表取締役副社長であり、システム開発・運用に関する品質・生産性委員会の委員長として、ソフトウェアの品質や生産性の可視化、計測化を推進していた。

当時、CSKにおいて、ソフトウェア開発プロジェクトの科学的な管理の一環として、構築計画時と終了時にファンクションポイント(FP)法によるシステム規模の計測を実施させていた。その時点までに、内容は未熟な点もあったが、約1,000のプロジェクトに関する計測データベースができていた。内容を精度の高いものにするため、より有益なケースをデータベースに残し、更にデータを活かしたシステム構築規模の予測モデルなどの分析にかかる方針としたばかりの時期であった。

私は嶋田課長に、もしIPAの新しい組織がFP法によるシステム規模の計測を行うならば、収集したデータを提供する用意があると伝えた。

SEC設立の準備が進んで、最終段階(2004年3月頃)において、新しい組織が当初取り組む、システム規模計測方法をどの方法論とするかが議論となった。当時まだそれほど有力なSI業者として認知されていなかったCSKが計測したデータなどには信頼

がおけない、という意見もあった。またFP法自体が持つ限界などがあったことも事実である。

しかし具体的にシステム規模の計測に取り組んでいる企業はほとんどなく、事例も少なかった。データベースの存在が評価されたようで、SECが取り組む計測方法論はまずはFP法でいくという方向となった。

SECはその後、ソフトウェア開発における諸側面の科学的な研究、実績の収集、分析と結果のフィードバックを行ってきた。エンタープライズ系のシステム開発における定量的管理、開発プロセスの定義、要件定義における留意点の指摘、高信頼性システム構築方法などに関して活発な活動を行った。また、組込み系システム開発においては、一連の高品質な組込みソフトウェア開発方法論の構築、組込みエンジニアのスキル標準などの研究と公表を行ってきた。システム構築の見える化に大きく寄与してきた。その事業成果はIPAの下記のWebサイト(<http://www.ipa.go.jp/sec/about/info.html>)に詳しく記載されている。

残念ながら、2009年7月に成立した民主党政権の下、事業「仕分け」の結果、2013年6月SECとしての活動は終了し、現在のソフトウェア高信頼化センターとして活動を継続している。継続はしたものの、ここに述べたような諸活動のかかなりの部分は中断を余儀なくされた。システム構築に関するルールや規格は、官民の協力のもとに、技術の進歩を先取りし過ぎず、かつ遅れないように制定されていくのが望ましい。

そのような活動のHUB的機能を担っていたのがSECであったと考える。ソフトウェア高信頼化センターの下で幾つかの機能は残ったが、2010年以降の数年間に急速に進展したIT利活用による新しいビジネス構築時代、すなわちデジタル・トランスフォーメーション時代への先導的、誘導的活動がそがれている感を禁じ得ない。

後述するが、インターネット、モバイル、クラウドをキーワードとする新しい時代に即した、セキュリティを十分考慮した開発方法論や、IoT時代にふさわしい組込みソフトウェア開発のあり方、ひいてはそれらを包括する新しいシステムライフサイクルプロセスなどの検討や標準、ルールの設定などに遅れが出ていると思う。今後機会があれば、現状組織の拡充などで、あらためて上記のような活動領域の拡大を図ってもらいたいものである。

これに関して、昨年産業構造審議会情報経済小委員会の議論において、経済産業省側から提示された資料に「ドイツにおけるインダストリー4.0の動向を主導しているフラウンホーファー研究機構のような機能がないことが、我が国が遅れをとっている原因、云々」というくだりがあった。委員の一人として、2010年から2013年に起きたことをあまりにも理解不足な経済産業省に大いに苦言を呈したものであった。

幸い、最近の情報セキュリティ問題の多発と対応策の重要性に鑑み、IPAの情報セキュリティ関連機能が充実され、政府の内閣サイバーセキュリティセンター(NISC)との連携(情報処理推進機構(IPA)及び国立研究開発法人情報通信研究機構(NICT))をはじめ、大規模なサイバー攻撃への対処などに対する知見を有する者との積極的な連携)も深まっているように、もっとIPAの持つ機能、知見を活用すべく動いて欲しいものである。

2

ユーティリティ化の実現による 情報システム構築の変化

SEC設立の2004年当時は、AmazonによるクラウドサービスAWSの開始が2002年、マーク・ザッカーバーグによるFacebook設立が2004年、Googleのナスダック上場が2004年8月、YouTube(インターネットでの動画共有サービス)創業が2005年、といった時期である。インターネットによるIT関連企業としてその後巨大化する者たちの黎明期と言えよう。

しかしながら、日本においては、とくに大企業の企業情報システムは依然として従来型のウォーターフォール型での構築が続いており、構築体制も多重下請け構造から抜け切れていなかった。2005年6月にJISA副会長を拝命した直後から、このままでは情報サービス産業の将来は危ういものとなると警告を発し、改革組織も作成して業界変革を展開した。

時代は業界の変革を待ってはくれない。先ほど述べたAWSサービス及び、マイクロソフト社のAZUREサービスにより、情報システムの世界は様変わりしていくのである。同時に、2007年発表のアップルiPhoneによりスマートフォンの時代が始まり、インターネット、モバイル、クラウドは、重要な3つのキーワードとなって、IT利活用の時代を変革してきている。

2.1 ユーティリティ化=クラウドの始まり

1961年、MIT(米国、マサチューセッツ工科大学)100周年記念式典で、ジョン・マッカーシー教授が、「タイムシェアリングシステムの技術によって(水道や電力のように)コンピューターの能力や特定のアプリケーションを販売するビジネスモデルを生み出すかもしれない」と述べたのは、AWSサービス開始の約50年前である。電力やガスといった各種の基礎的ユーティリティが30年~50年かかって普及していった軌跡を、コンピューティング・パワーもまたたどることとなったのは、非常に興味深いものである。約50年間の技術進歩により、ジョン・マッカーシーの予言は現実のものとなってきているのだ。

ジョン・マッカーシーはLISP言語の発明者であり、後にスタンフォード大学教授となる人工知能研究の第一人者であった。彼のスピーチのベースにあったのは、当時MITで構築されていたMulticsと呼ばれる、コンピューティング・パワーを多数(Multi)ユーザが同時に利用可能な高性能コンピューターの開発プロジェクトであった。余談であるが、プロジェクトは失敗に終わったが、Multicsを実現するために構築されたOSの構造を、シングル(Uni)ユーザ用のミニコンピューターに搭載したOS、すなわちUnixが遺産として残った。

この『コンピューター・ユーティリティ』という考え方は1960年代後半には非常に人気となったが、当時はハードウェア、ソフトウェア、通信技術すべて未熟であったために1970年代中頃には消えていった。1967年10月に行われた日本生産性本部の「訪米MIS使節団」の報告書(アメリカのMIS、日本生産性本部、日本電子計算開発協会共編、ペリカン社)に、これからの電子計算機利用は「ユーティリティ化」していくということが述べられていた。我々の先達は、当時の米国で最先端概念「コンピューター・ユーティリティ」を聞いて、電力、ガス、水道のようにコンピューティング・パワーを自由に使える日が来ることを夢見たのである。

2.2 ユーティリティ化とは

今日においては、コンピューティング・パワーのみならず、メモリ、ストレージ、ネットワークといったほとんどすべてのITリソースがユーティリティ化される状況が到来している。これによって、電力やガスの利用と同様に、基本的には利用量による従量課金が可能となり、利用者は設備投資や設備増強、設備運用、安定供給、バックアップなどに関して関与しないで済むようになった。また、リソース提供業者がかなりのボリュームのリソースを運用可能にしていると、24時間365日ノンストップ運用も可能である。

ユーティリティ化の世界では、これまでのコンピューター機器類を入手し、インフラシステムからアプリケーションシステ

ムに至るまで、すべてを注文生産する構築手法と異なり、表1のような各点があらかじめ規定され、原則的には利用者は「使った分だけ払う」環境が整備される。

表1

- コンピューティング・パワー、ストレージ、ネットワークに関する抽象化された課金単位の設定
- 契約約款と料金規定、SLAの標準化
- 少額な基本料金と、使った分だけ払う従量課金による価格体系
- 開発環境・ステージング環境の短期間での提供
- OS、基本ソフトの異なる環境が混在可能
- 使用量増加への柔軟な対応（通告、非通告型双方のサービスと課金体系）

2.3 ユーティリティ化を可能とする仮想化

このようなITリソースのユーティリティ化が可能になったのは、仮想化技術の進展によるものである。基本的には、コンピュータOSは、CPU、メモリ、HDDといったハードウェアへ全面的に依存しており、アクセス競合が起きることから、複数のOSで一つのハードウェアを制御することは通常できない。競合を防ぎ、複数のOSの並列的な同居を可能にするのが、仮想化ソフトと呼ばれる一連のソフトウェアである。仮想化ソフトが行うのは、CPU・メモリ・HDDなどハードウェア群をソフトウェア的に模倣・再現し、OSによって立つハードウェアリソースをいわば見せかけ的に用意してやることである。見せかけ的に再現されたハードウェアを「仮想マシン」、仮想マシンの上にインストールされ稼働するそれぞれのOSを「ゲストOS」と呼ぶ。

仮想ソフトをインストールするサーバーハードウェアは「物理マシン」あるいは「物理サーバー」と呼ばれ、仮想マシンと区別される。仮想化システムは、1台の物理サーバー上に複数の仮想マシンを立て、それぞれの仮想マシンにゲストOSを稼働させることによって成立している。

仮想化によって実現できる代表的な機能にライブマイグレーションがある。仮想化された複数の物理サーバーを起動している状態で、任意の物理サーバー上で動作する仮想サーバーを、ストレージを共有する別の物理サーバーへ、サービスは継続したまま瞬時に移動することができる。ライブマイグレーションは、物理サーバーのメモリ上にある仮想サーバーの情報を、別の物理サーバーのメモリへコピーすることで実現できる。これらの機能が、最新のクラウドサービスで活用され、従来のオンプレミス・サーバーでは容易にやり遂げられなかった柔軟性のある高品質運用が、クラウドでは提供可能になる。

2.4 ユーティリティ化のインパクト

ユーティリティ化の実現により、現実にはクラウドの世界が実現した。AWSやAZUREが常識の世界になってきた。CPUパワー（サーバー）はもちろんのこと、ストレージから、本来ユーティリティであるはずのネットワーク利用までもが、SDxx (Software Defined xx) として更なる高度な仮想化が行われるところまできた。

仮想化のもたらす効果としては、次のような点が挙げられる（表2）。更に重要な点は、これら各項目とも、規模の利益を享受できることを示している。個別のユーティリティ・センター、すなわちクラウドセンターの規模が大きく、かつ世界各国に展開されていけばいいだけ、利用者はより柔軟で品質の高いサービスを低廉に享受できることになる。

大規模なセンターにリソースが集中されることが有利になれば、サーバーや、ストレージ装置、更にはネットワーク機器、というこれまでのIT（ハードウェア）産業を支えてきた「箱もの」の「台数減少」をもたらす。近年、かつてサーバーやネットワーク機器を生産していた大手IT企業（IBM、富士通、日立、HPなど）が、全く元気を失ってきた最大の原因がここにある。EMCの仮想化ソフトVMwareやセキュリティ関係企業群が欲しかったDELLは、その買収によって見事に立ち直ってきたことが、この業界構造の大変革と生き延びる方向の一例を示している。

表2

- **リソースの有効利用**
従来では低負荷時に大量に余りがちだったCPU処理能力やメモリといったサーバー・リソースを、複数のOSで分配し有効に活用できる
- **省コスト・省エネルギー**
物理サーバー運用台数の大幅減少により、消費電力・設置スペース・管理リソースといった様々な側面でコスト削減を実現
- **高い柔軟性**
サーバーが物理的な制約から逃れ、仮想マシン上で稼働することで、多台数の一元管理、サーバー立ち上げの高速化、ダウンタイムの最小化
- **高い耐障害性**
仮想マシンはそれぞれが完全に隔離されており、いずれかがクラッシュしてもほかの仮想マシンはそのまま稼働するため、仮想マシン同士のサーバークラスターで可用性向上・耐障害性の向上が可能

3 クラウド時代の到来

3.1 利活用者主導の時代

AWS及び、AZUREに加えて、ここ数年、多数のIT関連企業からクラウドサービスが提供され始めた。しかしながら、2.4で述べたように、クラウドサービスの決め手は規模にあり、先行する2社に対して、日本勢はもとより米国IBMと言えども十分な応戦ができていない。

また、クラウド上に様々なアプリケーションが提供されることにより、従来型のシステム構築は大きく変化してきた。従来から運営されてきた受発注や在庫管理、生産管理といった大型定型事務システムのクラウドへの移行は始まったばかりであるが、セールス・マーケティング、IT技術を活用した新サービス、大量データを分析活用するシステム、中堅・中小企業の新しいシステム化などには、積極的に活用されている。

クラウドの活用で特徴的なことは、ユーザ(利活用者)側が積極的に活用していることである。従来、情報システム部門に依頼しないとできなかった、機器の手配から購買、契約締結、ソフトウェア作成などの作業が、クラウド上に展開されているありもののソフトウェアやオープンソースなどの活用で、容易に短期間でプロトタイプ構築でき、かつ低廉に実現できることをユーザが知ってしまったのである。

3.2 ビジネスモデル構築のアジャイル化とイノベーションの創出

ユーザが自らビジネスモデルを考え、クラウドを活用して実験を行い、うまくいけばすぐに「ビジネス」を実施する、といういわば「ビジネスモデル構築のアジャイル化」とも言うべき時代がやってきている。従来の情報システム部にはインフラの整合性やセキュリティルールなどは支援してもらおうにしても、ユーザに多少のシステム知識があれば、クラウド上に、業務を分かっている者が、クラウド上の道具を活用して構築するほうが早い。

Webを作り、Eコマースサイトを立ち上げ、顧客管理CRMを構築し、上がってくるデータはWeb最適化やデータ分析で活用…といったことが容易に、スピーディにできる。従来型ウォーターフォール開発ではスピードが出ない、コストがかかる。更に重要なことは、コストの低減、スピードの向上により、繰り返し実験可能なことからくる「失敗コストの軽減と、成功確率向上」である。

結果として、ビジネス・イノベーション創出の加速化ができる。このような点に関してはまだ十分に認識されておらず、クラウド利用のコストと従来型手作りコストの比較云々といった議論が盛んである。しかし、ビジネスモデルの創出スピードの違いは、単なるコスト比較では測れない重要な点である。

4 Amazonに見られる新しいIT利活用の姿

第2、第3のAWS登場か？

最近のAmazonの先端技術への取り組みが素晴らしい。日本においても、昨年来矢継ぎ早に色々と公表された。Amazon Robotics、Amazon Go、Amazon Echo、Amazon Dash Button (Amazonダッシュボタン) などである(図1参照)。

中でも注目したいのが、Just Walk outをうたい文句にしている、最新鋭テクノロジーの塊のような新しい概念の小売店舗、Amazon Goと、Amazon配送(物流)センターを自動化するための社内チーム「Amazon Robotics」である。2012年3月に約7億7500万ドルかけて買収したKIVAシステムズがベースになっている、配送センター用ロボットが導入されたセンターは米国に20カ所以上となり、欧州などでも導入が進んでいる。2016年12月には日本でもロボットが導入されたセンターが、川崎にオープンした。ロボット導入によって、配送センター内の効率化が40%以上向上するという。

「Amazon Robotics」は2003年に設立され、Amazon.comの配送センターの効率化が目的である。配送センターを様々なロボット工学技術で自動化することを目的としており、その研究分野は自律移動ロボット、精巧なコントロール・ソフトウェア、言語認識、パワーマネジメント、コンピュータービジョン、深度センシング、機械学習、物体認識、コマンドの意味論的理解など、様々な分野に及ぶという。同様の技術はAmazon Goにも活用されている。

Amazonは既によく知られているように、自社の膨大な販売、流通を支える情報システムで培われた経験、ノウハウをAmazon Web Services (AWS) としてクラウドサービスの世界に乗り出し(というよりはクラウドサービスの世界を創出し)、現在世界シェアNo.1を誇るまでになっている。仮想化を最大限活かしたコンピューティング・パワー提供ビジネスは、既存の大手ITベンダのビジネスモデルを危機に陥れたことは周知の事実である。しかも、現状社内での一番の収益源はAWSなのである。

上記の配送センター効率化の諸技術をもってすれば、他企業の配送(物流)センターを自動化するための新しいシステムサービスが考えられるだろう。システムサービスだけではなく、運営ノウハウを含む物理的な倉庫運営を効率良くやってのけるであろう。現状で3PLサービスを標榜する物流専門各社には、残念ながらこのような最先端諸技術を手がけるだけのノウハウも、人もいるまい。もちろん、既存のITベンダにもそのような技術蓄積はない。

またも世界は新しいAWSすなわちAmazon Warehouse Servicesに席卷されることになるのであろう。そして、これがAmazon

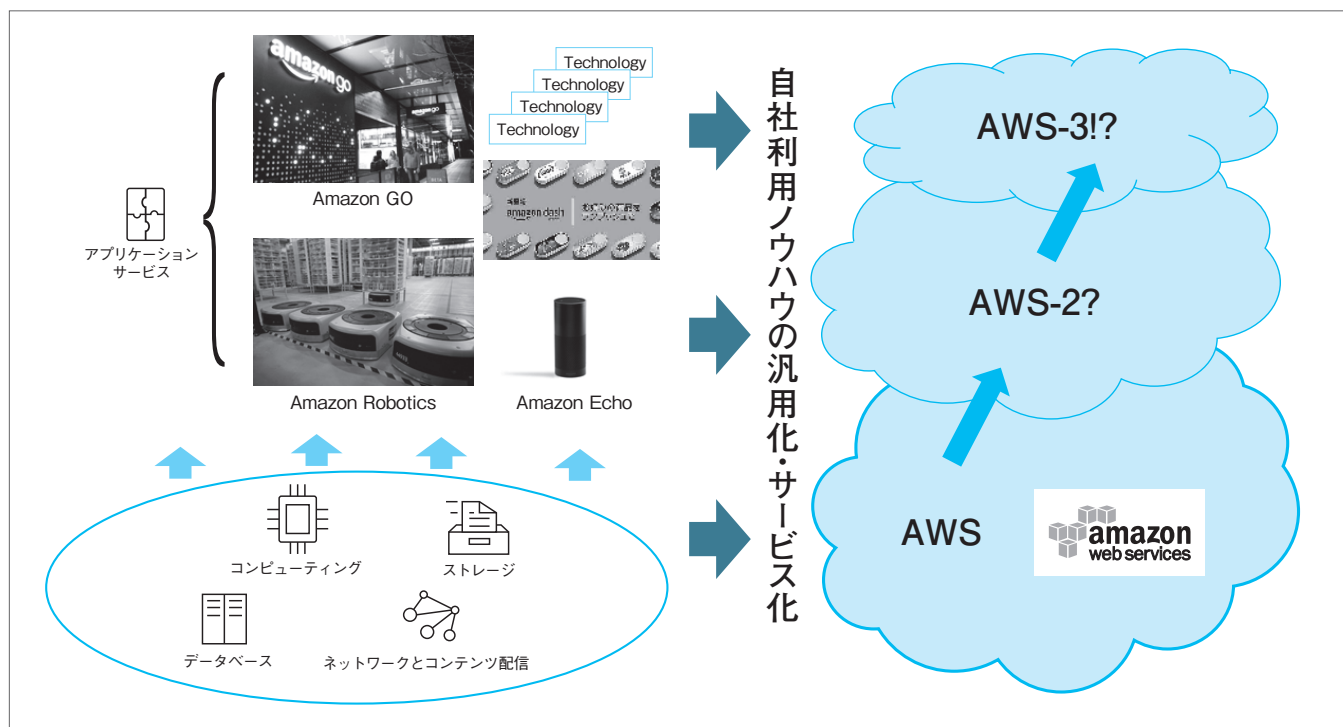


図1 Amazon: IT活用新しい企業スタイルへ

GOと結び付くと、流通、物流から小売りに至るまでのすべてのプロセス (Whole-Sales Activity) が、更なるサービスとして提供され始めるだろう。これもまたAWS: Amazon Whole-Sales Servicesとなる。

第2、第3のAWSを生み続けるAmazonは、従来型のIT企業ではない。しかしITをフルに活かしたビジネスモデルを生み出し続けている。IT活用によるイノベーションとは、こういう動向を言うのだと思う。

5 これからのIT活用と人材

一昨年、産業構造審議会情報経済小委員会で、日米IT技術者の分布状況について話題になった図がある (図2参照)。

IT人材が、ユーザ側企業に所属しているのか、供給者側 (外部情報サービス企業) に所属しているのかの日米比較図である。米国では、その比率が71.5%対28.5%、奇しくも日本の比率はほぼ逆の24.8%対75.2%であった。結論を言えば、日本においては利用企業内若しくは組織内IT人材の厚みが形成されておらず、主要な構築業務は外部情報サービス企業に委託されているのである。

もちろん優秀なSI業者がいることを否定するものではないが、やはり自社の業務に精通している人材が、経営課題を踏まえてシステムの構築を行うのが正しいのではないか。その中で、とくに技術的に専門性がある分野、例えばデータベースであったり、ネットワーク、もしくは大量データ (ビッグデータ) 分析やAIと

いった分野に関して、外部業者から専門的な支援をもらうのが通常であろう。自社における技術者が少ないと、定常的なシステム運用、外部業者との契約、購買、社内調整などに手間を取られると、実質的に社内のユーザ部門への対応が十分できず、結果外部SI業者などに丸投げとなる。このような状況で真に経営の課題を受け止めて、根本的に解決を目指すシステムが構築できるとは思えない。

米国の場合は、人材の流動性が高いので、自社人材71.5%の中で業務専門人材が企業間を異動できる。日本の場合、労働慣行と25%の比率ではあまりそれも望めない。となると、外部にいる75%の中から、ユーザ側へ人材流動を起こすことと、新たに人材育成、発掘が必要である。例えばここ数年で、比率を50%対50%にしようとする、50万人程度の人材異動が必要だ。年10万人。大変な数である。もちろん企業内における情報システム部門のあり方も見直しが必須だ。

既に多くのベンチャー企業においては、自社内でシステムを構築する、若しくは最低でも、自社でやりたいこと (要求仕様) は自社内で詳細化する、ということが普通に行われている。むしろ伝統的大手企業や中堅、中小企業において、依然として丸投げ傾向が残っている。

このような状況を、ユーザ企業の経営者はもっと真剣に捉え、対応策を打つべきである。とくに中堅、中小企業のIT化は待ったなしであり、大企業に比較して低い生産性の向上、更には新しいビジネスモデルによる企業変革を目指すならば、優秀な人材の確保は必須であろう。

日米のIT技術者の分布状況

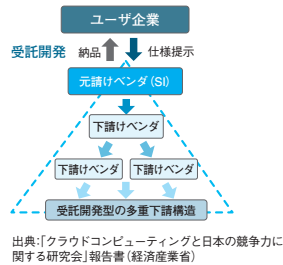
経済産業省産業構造審議会商務流通情報分科会情報経済小委員会（平成27年2月9日）

資料7:IoT時代に対応したデータ経営2.0の促進のための論点について(事務局説明資料)28頁より

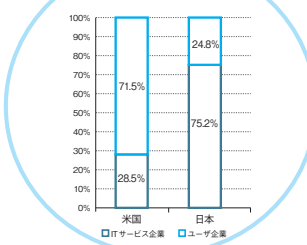
③成長エンジンとなる企業のデータ活用に係る体制の見直し

ユーザ企業のIT経営やデータ活用促進のため、IT子会社やITベンダへ外注している業務の内製化（IT業務の下請発注体制の見直し）や横断的組織など組織の見直しなどどのような施策が考えられるか。

<多重下請け構造の概念図>

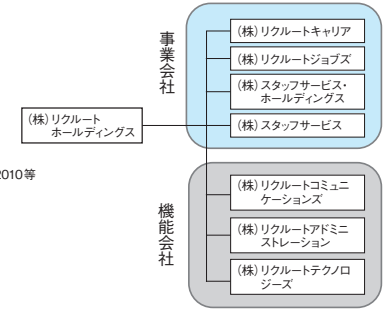


<日米のIT技術者の分布状況>



<データ活用のための組織体制強化の事例>

- ・リクルートは2012年に(株)リクルートテクノロジーを設立し、ビッグデータの専門部隊によるビジネス部門の事業支援を本格化。グループ各社に対して競合優位性の高いIT・ネットマーケティング基盤の開拓、ビジネス実装の支援を行うことにより、リクルートグループの競争優位の構築を目指す。
- ・コンサル型や、技術型といった専門性を有するデータサイエンティストが事業部門のマーケット精通者と一体となって取り組むことで各種事業分野で実績を上げている。



○部分が、あまりにも対極的であり、かつ日本のIT活用が米国に対して遅れていることへの一つの大きな原因であろうという議論が行われた。

→ユーザ企業のIT経営を促進するためには、多重下請け構造の中で能力を発揮していないIT人材の流動化を進め、ユーザ企業が必要とするIT人材を獲得できる環境を整備するべきではないか。

図2 日米のIT技術者の分布状況

6 人材像も変えなければ

経済産業省の情報関連部門改革が、7月より実施された。情報通信機器課と情報処理振興課(情振課)を、ハードウェア産業とソフトウェア産業に対する政策を一体的に実施する「情報産業課」と、IT活用の促進などの利用者向けの政策を適確かつ効率的に進める「情報技術利用促進課」に再編するものである。IT活用の新しい動向に即しての、対応体制の整備である。ここ10年間に起きているIT適用範囲の急拡大と、それに伴う新しい産業の興隆への対応には、やや遅きに失した感もある。

これまで情振課の施策の一部でしかなかった「利用」に関して、専門の部署を設けることは非常に重要だ。もちろん、基礎的技術や新しい機器の開発を軽視して良いわけではない。が、世界的に見ても後れを取っているIT活用による新しい産業の振興、それを可能とする人材の育成、社会環境変化の促進・対応などのほうが、はるかに重要である。新しい組織分担の中で、5節で述べたように新しい時代に適合する人材の育成は、喫緊の課題である。最近、IPAが第4次産業革命に対応した新スキル標準の検討にあたり、専門分野の更なる具体化が求められる「セキュリティ領域」と、足元でとくに必要性が高まっているものの、現状のITスキル標準(ITSS)には十分に含まれていない「データサイエンス領域」の2つについて、新スキル標準の策定に先行し「ITSS+」(プラス)として公開をした。これは、主に従来ITSSが対象としている情報サービスの提供やユーザ企業の情報システム

部門にいる既存の人材が、2つの領域のそれぞれに向けたスキル強化を図るための「学び直し」指針として活用されることを想定している。「データサイエンス領域」では、ビジネス、データサイエンス(含むAIなど)、データエンジニアリングの3つのスキルカテゴリを定義している。

ITに関連する技術やスキルは大きな広がりを見せており、供給者側人材に加えてユーザ側人材の育成が急務である。更には、AI、ビッグデータ、IoTといった広がりカバーするには、もはや単なるIT人材、という一くくりの概念では表せないものとなっている。せっかく官の統括部門構成が変わるのであるから、この際IT人材という、あいまいで焦点の合わない表現をやめ、より具体的な人材像が浮かぶような言葉を使ったほうが良い。もちろんそれに対応して、既存のITSSと、情報処理技術者試験制度の根本的な見直しが必要だ。現状の制度は利活用者と言うよりは、供給者側の試験制度であった。時代は変化しユーザによる新しい利活用時代が始まっている。ユーザ側において利活用を促進、推進する資格付与を含む諸制度は、これまで本格的に検討されたことがない。小学校からのコンピュータープログラミング教育は緒に就いたばかりだが、ここでIT利活用を理解している人材が育ってくるまでには、10数年はかかるであろう。高校生や大学生のITリテラシーを高め、経営の課題を、ITを利活用していかを実現するかを考えられる人材に育成していく、といった観点の教育や訓練体系は、あるようで少ない。この面でも、IPAの果たすべき役割は大きい。

産学官連携の基盤となる ソフトウェア工学雑誌SEC journal

— 50号記念に寄せて —

大阪大学 大学院情報科学研究科 / 産業技術総合研究所 情報技術研究部門 井上 克郎

SEC journal誌が50号を迎えるにあたり、創刊当時の振り返り、論文としての魅力、産学官連携の場としての重要性、ソフトウェア技術者への教材としての役割などについて述べる。

1 SEC journalの創刊

SEC journalが本号で50号となることはソフトウェア工学の研究、実践にかかわるものとしては、大変素晴らしい出来事であり、永年の関係者各位のご努力、ご支援に対し、心から敬意を表する。

今、2005年1月のSEC journalの創刊号を見てみると、SECの設立当時の色々な活動が蘇ってくる。我々は奈良先端科学技術大学院大学の鳥居宏次先生のもと、文科省の委託を受けたe-Societyプロジェクトの一環として、EASE (Empirical Approach to Software Engineering) プロジェクトという実証的なソフトウェア工学手法を開発、展開する活動を活発に行っていた。北陸先端科学技術大学院大学の片山卓也先生は、同じe-Societyプロジェクトの一環として、組込みソフトウェアの信頼性を高めるための技術開発を推進していた。これらの活動の記事が大きく掲載され、当時の様子が目に浮かぶ。またドイツのフラウンホーファー研究機構 実験的ソフトウェア工学研究所 (IESE) 所長のDieter Rombach先生と鶴保証城SEC所長との産学官連携に関する対談、東京大学の藤本隆宏先生の「ものづくり経営研究センター」の紹介、組込みソフトウェアスキル標準の議論など、当時大きな話題になっていた記事が掲載されていた。

これらの記事のもととなる活動は、十数年後の現在でも引き続き活発に活動が続けられているものばかりであり、あらためてSEC journalの記事の確かさを感じると共に、ソフトウェア工学における諸活動の息の長さ、普遍性というものを再認識している。

SEC journal創刊時、関係者からこういうものを作ると聞き、大変面白くて有用なものができる、と強く期待した。当時、

EASEプロジェクトなど実証的ソフトウェア工学の普及活動に携わっており、それらの成果を広く伝える手段がなかなか見つからなかった。学会誌の論文として投稿、採録、掲載されるためには、研究としての新規性が必須であるが、普及のためには、逆に少し古くても確実で安定した技術が望まれる場合も多く、活動成果を学会論文として掲載することは困難であった。また、商業雑誌が記事とするには、我々の活動はいささか地味であり、取り上げにくいものであった。SEC journalは、学会誌や商業誌にはない視点、すなわちソフトウェア工学の技術を広く日本に普及させ、日本のソフトウェアに関する技術力を向上させる、というものを持っており、我々の活動にとって非常にありがたかった。我々が行ったEASEプロジェクトや、その後に行ったソフトウェアの開発保証技術に関するSTAGE (Software Traceability and Accountability for Global software Engineering) プロジェクトの活動や成果に関しても何度も掲載していただいた。

一方、この雑誌を持続させることの難しさも感じた。ソフトウェア工学という限られた分野、それも日本国内だけの読者を対象とした雑誌に対して、どれだけ高い品質の記事や論文を継続的に集めて掲載できるか、これは今も昔も、大変チャレンジングなテーマである。SECという力強いバックアップがあるとしても、何年も継続して出版していくことはなかなか容易なことではない。実際、編集を担当している関係者から、面白い記事はないか、実践に即した役立つ論文はないか、など何度も問われ、その先細りを心配したが、杞憂に終わったことを喜ぶたい。

SEC journalの発刊後、実践的な知見に関しても論文として集めて掲載しようとする動きが学会でも起こっている。これは、学会においても、実践的な論文やシステム論文などの重要性が認識された結果ではないかと思っている。

2 SEC journalの論文

SEC journalは広く論文の投稿を受け付けており、企業や大学の技術者や研究者が自分の成果を広く公表できる場所になっている。投稿された論文は複数のレビューによって査読されており、品質も担保されている。また、十数年間の積み重ねにより、掲載論文に対しては研究者や技術者から一定水準の評価が得られており、博士号取得のための条件の論文として認められている大学もある。これは、ソフトウェア工学を学ぶ社会人の博士学生にとっては、非常に重要なことであり、実務に近い研究内容について論文としてまとめ、発表でき、学位取得を可能にすることを意味する。

前述のように、学会の論文誌は、一般には高い新規性を求めると共に、有効性の厳密な証拠を提示することを要求しており、それを示すためには多大な時間、コストが発生する。SEC journalは、実践的にどのような価値があるか(あったか)が重要であり、実務家にとっては業務の目的に沿った活動の一環として論文をまとめればよく、負担が軽減できる。これからも多くの実務家や社会人の博士課程学生が、論文として投稿し、採録、掲載されることを期待する。

3 ソフトウェア工学における産学官連携の架け橋

ソフトウェア工学という分野は、工学・理学の色々な研究分野の中でも、かなり実学的要素の強い分野である。ソフトウェアを開発し、利用する産業界やユーザの要求、要望に応えるべく新たな技術、新たな方法を開発し、社会に還元することが究極の目的であろう。実際の利用現場の要求や評価がないソフトウェア技術は普及が困難である。学問的な新規発想だけでは、社会的なインパクトを与えるものを作ることは難しい。そういう意味で、ソフトウェア工学研究には、産学官交流が必須である。SEC journalは、それをを行う場を提供している。産学官のそれぞれの現状が分かる記事がタイムリーに掲載されており、自分の今後の方向性を考えるために貴重な情報源となっている。とくに、特集記事は、その時々重要なテーマの記事を集約しており、研究者にとっては産業界の動向を知る上で大変役立っている。

今後もソフトウェア工学の実践のために必要な普遍的な知識、例えば、実証的なバグデータ分析、開発プロセス、スキル標準などを継続的に扱うと共に、これから重要になってくるとされる技術、例えば、深層学習、IoT、巨大データ分析などの記事も継

続的に取り扱い、産学官のソフトウェア工学の活動にどのようなインパクトがあるかを検証し、それぞれに刺激を与え続けるような魅力的な紙面作りを続けていってほしい。

自分自身の活動を振り返ると、産学官の連携活動というのは、個人的な関係に依存した属人的なものに陥りやすいが、SEC journalが提供するような色々な立場での活動や成果の情報を見ることによって、より大きな視点で連携を考えることができる。

4 ソフトウェア工学教育のための情報源

ソフトウェア工学では、ソフトウェアの生産性、信頼性の向上を目指し、色々な活動を技術的、管理的な側面で捉えて議論している。その重要な側面の一つとして、技術者の知識や技術を向上させための教育の議論がある。ソフトウェア工学における銀の弾丸を探し出すことは困難であるが、確実な投資効果の得られるものは技術者のソフトウェア工学教育ではなかるうか。

SEC journalは、日本のソフトウェア技術者が、実際のソフトウェア開発の現場の問題、解決の方向性、今後の技術動向などを探る上での最良の情報源となっている。今後も引き続き、技術者にとって有益な記事を提供し続けることによって、この分野の大きな指針となっていくことを強く望む。

5 SEC journalの今後

論文や雑誌の評価は、そのコンテンツが与えるインパクトの大きさで決まるが、良い論文、良い記事を集めるには、多くの人々がそこに投稿したい、記事を書きたいという論文や雑誌である必要がある。しかし、そのようになる即効性の高い方法は思い付かない。一般には、長年同一分野で安定した品質の出版を繰り返し、評判を少しずつ上げていくしかないのではないかと。SEC journal創刊から10余年経過し、その価値も安定してきており、今後も地道にその評価をじわりと上げていって欲しい。また、掲載される論文や記事が話題になって、日本のみならず世界のソフトウェア工学の技術者、研究者に大きなインパクトを与えることを期待する。

企業から見た SEC journal、SEC journal論文

三菱電機株式会社 設計システム技術センター ソフトウェア技術推進部 久野 倫義

SEC journal、SEC journal論文には、開発現場の問題点を解決した事例や現場ですぐに利用できる情報が掲載されており、企業における利用価値が高い。また情報源としての価値に加え、情報発信者としての達成感や高揚感を得られる貴重な場であり、多くの方々に論文を投稿することをお勧めしたい。

1 はじめに

開発現場では様々な問題が発生し、それを解決しながら開発を行っている。解決策も様々あり、先端的な取り組みのみではなく、過去の改善事例を参考にしたものも多い。そのような開発現場における活動をjournal論文として発表しようとする場合、発表に対しハードルが高いという意識が働く。進歩性と有用性の両面を持つ活動でなければ発表できないのではないかと、現場の改善活動には進歩性は少ないのではないかと、思ってしまうことがある。

上記現状から、企業の開発現場で実践した活動をjournal論文として発表を促進するには、意識的なハードルを下げるのが重要である。この意識的なハードルを下げる活動がなければ、開発現場で実践された様々な活動を広く世の中に広めることは困難であると言える。

2 企業から見た SEC journal、SEC journal論文

2.1 情報源としての有効性

SEC journal、SEC journal論文には、開発現場の問題点を解決した事例、現場ですぐに利用できる情報、IoTや情報セキュリティなどの最新情報が掲載されており、企業における利用価値が高い。更に、開発現場で実践された活動がjournal論文として認められていることを知ることで、開発現場にいる読者が発表してみようという意識的なハードルを下げることに貢献していると言える。

2.2 情報発信先としての有効性

企業で実践した活動に汎用的な価値があるのかを、企業内で評価することは困難である。査読付き論文として評価されるSEC journal論文は、企業が実践した活動の価値を客観的に判断いただけるため、腕試しとして貴重な場となっていると考える。企業で働きながら、自分の実践した活動を広く世の中に広めたいという情報発信のモチベーションに対し、企業や学会で広く購読されているSEC journalは、それを実現する手段を提供していると言える。

3 今後への期待

3.1 論文賞受賞時に感じたこと

私は、これまでSEC journal やIPA/SEC出版の本を購読し、他部門の活動を参考に改善活動を行ってきた。そして、SEC journal論文を投稿することにより、情報源として利用してきた立場から、広く世の中に情報を発信し、開発現場の問題解決に貢献するという立場となり、高い達成感を得ることができた。更に、多くの論文の中から、自分たちが行ってきた活動が高く評価され、論文賞をいただいた際には、あらためて高揚感を感じた。SEC journalは、情報源としての価値に加え、情報発信者としての達成感や高揚感を得られる貴重な場であり、多くの方々に論文を投稿することをお勧めしたい。

3.2 論文賞受賞時の企業の反応

論文賞受賞時には、事業所内報に受賞したことが掲載され、事業所内であらためて活動を周知することを行った。また社内の読者から祝福のメールがあり、SEC journalが社内でも広く読まれていることを再認識できた。更に、社内の講座において、受賞時のSEC journal論文を教室に置き、受講者へ周知する活動を行うことで、より広く社内へ情報展開することに活用できた。このような活動により、論文を書いてみようという社員が増え、自己の仕事を論文としてまとめるという技術力の向上と、新しい視点による仕事への取り組みに貢献できると考えている。

3.3 今後への期待

SEC journal、SEC journal論文は、これまで企業における問題点や改善事例を取り上げ、産業界への貢献、学会における研究テーマへのインプットとしての二つの重要な役割を担ってきたと考える。企業における問題は、AIの進展やIoTなどの外部環境の変化により、従来とは異なった形で現れ、これまでの解決策では対応できないケースが増加してきている。このような、変化する問題を解決するために、企業における努力と、学会における新たな取り組みを融合させた対応がますます重要となってくる。ぜひこのような視点で、今後この貴重な場を維持していただければ幸いである。私も論文という形で貢献できるように努力していきたいと考えている。

SEC journal(論文)の見える方

NTT ソフトウェアイノベーションセンタ 齋藤 忍

SEC journal 創刊50号、おめでとうございます。本特集への寄稿をSEC journalの編集の方からご依頼があったとき、「(SEC journalに)自分が投稿したのはいつだったか?」とすぐに思い出せませんでした。調べたところ、私が投稿したのは2008年の夏で(第15号に掲載)、9年も経っていました。この間、企業の研究開発部門に所属していることもあり、ほかの論文誌や会議にかかわる機会もありました。上述の編集の方からも「企業にいる立場から、他誌・他会議の論文と比べたSEC journalの論文の見える方を述べて欲しい」とのリクエストでした。そこでSEC journalの中で「論文」に的を絞り、私の見解を述べたいと思います。

1 インプット(情報収集)としての論文の価値

仕事をする上でも世の中のソフトウェアエンジニアリングの動向は気になります。情報収集の一環として、私は論文誌や会議の論文も活用しています。

私にとって、論文による情報収集の価値は主に二つあります。一つ目は、萌芽的研究(Cutting-edge Study)の動向を把握することです。「まだ実際の開発現場への適用には時間がかかりそうだけど、提案している手法やツールの有用性はとても高そうだな」といったことが分かります。二つ目は、(とくに他社の)開発現場で実践した結果(After-the-fact Study)を知ることです。「この企業はこんな手法やツールを使っているのか、そしてこれぐらいの効果があるのか(若しくはないか)」といったことが分かります。

プレスリリース(報道発表)やホワイトペーパー(製品解説)も重要な情報源です。これらは速報性が高く、内容も分かりやすいです。ただし、紙面の都合などで採用技術の中身や、適用現

場の詳細までは書かれませんが、

一方、ソフトウェアエンジニアリングの取り組みに関する論文は、一定の形式(例:背景、目的、提案内容、適用結果、評価・考察、将来の課題)に則って記載されます。速報性は必ずしも高くはない場合もありますが、ぴったりの研究テーマであれば、他社の取り組みを把握する上でとても有益でリッチな情報源になります。企業の研究開発部門に属する立場として私が論文を活用する理由の一つであります。

2 アウトプット(投稿・発表)の機会

ソフトウェアエンジニアリングの取り組み、とりわけ企業の実際の開発現場に適用・実践した論文を積極的に募集する論文誌・会議は、SECジャーナル以外にもあります。以下に記す私が過去にかかわった二つを含めて、三つの論文誌・会議の概要を表1に示します。

表1 3つの論文誌・会議の概要

	投稿・発表形式	刊行・開催時期	投稿締切	査読プロセス	査読基準	編集委員の構成
SEC journal	A4タテ・2段組 (8ページ以内)	年4回刊行	1月・4月・7月・11月の各月末	2名以上の 査読委員による 審査	実用性、可読性、有効性、 信頼性、利用性、募集 テーマとの関係	産学所属の委員が ほぼ同数
デジタルプラクティス	A4タテ・2段組 (4~8ページ) ※通常原稿	年4回刊行	常時受付	1次審査 2次審査 共同推敲	社会的有用性	企業所属の委員が 多数占める
JDMF経験報告 (旧:SPES)	MS-PowerPoint PDF(発表15分)	年1回開催 (2017年は10/22)	開催日の約2カ月前 (2017年は8/22)	企画WG全委員 による審査	実践の観点から新規性、 独創性、有効性	企業所属の委員が 多数占める

デジタルプラクティス (DP) ※1

情報処理学会から年4回刊行されている論文誌です。企業・組織などの実務における具体的な課題と情報技術の実践を扱う論文を募集しています(情報技術の対象はソフトウェアエンジニアリングに限定はしていません)。

JISA Digital Masters Forum (JDMF) 経験報告※2

情報サービス産業協会(JISA)が開催する、ソフトウェアエンジニアリングの実践や効果に関する経験報告や実践事例を発表するイベントです。以前はSPES(Software Process Engineering Symposium)と呼ばれていました。

表に記していますように、SEC journalを含めたこれら3つは投稿・発表の形態や査読のプロセスなどに細かな違いはあります。ただし、ソフトウェアエンジニアリングに関する手法やツールの“実践”や“実証”に関する論文を扱っているという点では同じであると言えます。

3 SEC journalの見え方

3つの論文誌・会議に様々な立場(投稿者・編集者・査読者・一般読者)でかかわってきました。本稿を執筆にするにあたり、一般読者としてSEC journalに過去に掲載された論文(2012年～2016年までに表彰対象となった合計10編)を見てみました。SEC journalの論文は以下のような特徴があることを再認識しました。

- ソフトウェアエンジニアリングの手法やツールの企業内での適用報告の論文が圧倒的に多い。
- ほぼすべての論文で、適用した手法やツールの有用性や効果を定量的に評価している。

SEC journalの論文からは、各企業の等身大のソフトウェアエンジニアリングの取り組みが把握できます。更に、定量的なデータは、自らが実践する上でのベンチマークとして大変参考になります。

企業に属する立場から見ると、(とくに他社の)開発現場で実践した結果(After-the-fact Study)を知りたいものです。そして、私自身も他社の実践の取り組みの論文を見つけると、職場の同僚に共有(メーリングリストでの周知やミーティングでの紹介)をしていました。DPやJDMF経験報告に集まる論文は、ほとんどが開発現場で実践した結果が占めます。これは論文の査読基準からも必然的なものです。

一方、私にとってSEC journalにはもう一つの見え方がありま

した。SEC journalには、将来の開発現場への適用を目的としたソフトウェアエンジニアリングの手法やツール(Cutting-edge Study)に関する論文が一定数の割合であります。私はこれらの論文を「手法やツールの有用性は高そうだけどまだ共有することはないかな。適用はまだ先かな」と思い同僚には共有せず、自分で興味深く読んでいました。また「後で読もう」と思い、該当論文をコピーしていました(今回、それらが“積読”状態として発見されました…)

以上のように、一般読者として見た場合、私にとってSEC journalの論文は、数多くの開発現場での実践結果(After-the-fact Study)を把握でき、ときおり興味深い萌芽的な研究(Cutting-edge Study)の動向を調査できるという二つの価値があり、私の情報感度を高めてくれる読み物でした。ちなみに同じような価値を感じる場としては、研究論文と実践論文の二つのカテゴリを募集するソフトウェアエンジニアリングシンポジウム(SES)※3があります。

最後にSEC journalにお願いがあります。常々、SEC journal内の特集と連動した論文を掲載して欲しいと思っていました。例えば、昨年1年間でもSEC journalはIoT、人工知能、組込みソフトウェア、セキュリティ・高信頼化というテーマで特集を組んでいます。読者としては、特集の内容に関連した論文が同じ号にあると大変興味をそそられます。特集と論文が同じテーマのSEC journalは冊子としての価値も高まります(きっと保管しておこうと思います。また積読になるかもですが…)

4 おわりに

色々勝手なことを書いてきましたが、SEC journalは、ソフトウェアエンジニアリングに関する萌芽的な研究と現場の実践結果の双方の知識・経験を共有化(掲載・発表)できる、ちょうど良いバランスが取れた場としてこれからも続いていって欲しいと思っています。

今回は一般読者の(インプットをする)立場からの見え方を書きました。ただ、いつも私が言っているのですが、アウトプット(投稿・発表)をすること、及びそのための準備をすることで得られるインプット(知識・気づき)もたくさんあります。本稿で取り上げた論文誌・会議以外にもアウトプットの場合はたくさんあります。ぜひ投稿・発表を検討してみませんか？

脚注

※1 デジタルプラクティス : <http://www.ipsj.or.jp/dp/dp-index.html>

※2 JDMF 経験報告 : <http://www.jisa.or.jp/event/jdmf/tabid/2259/Default.aspx>

※3 SES2017 : <http://ses.sigse.jp/2017/>

SEC journal 50号の歩み

SEC journal編集部

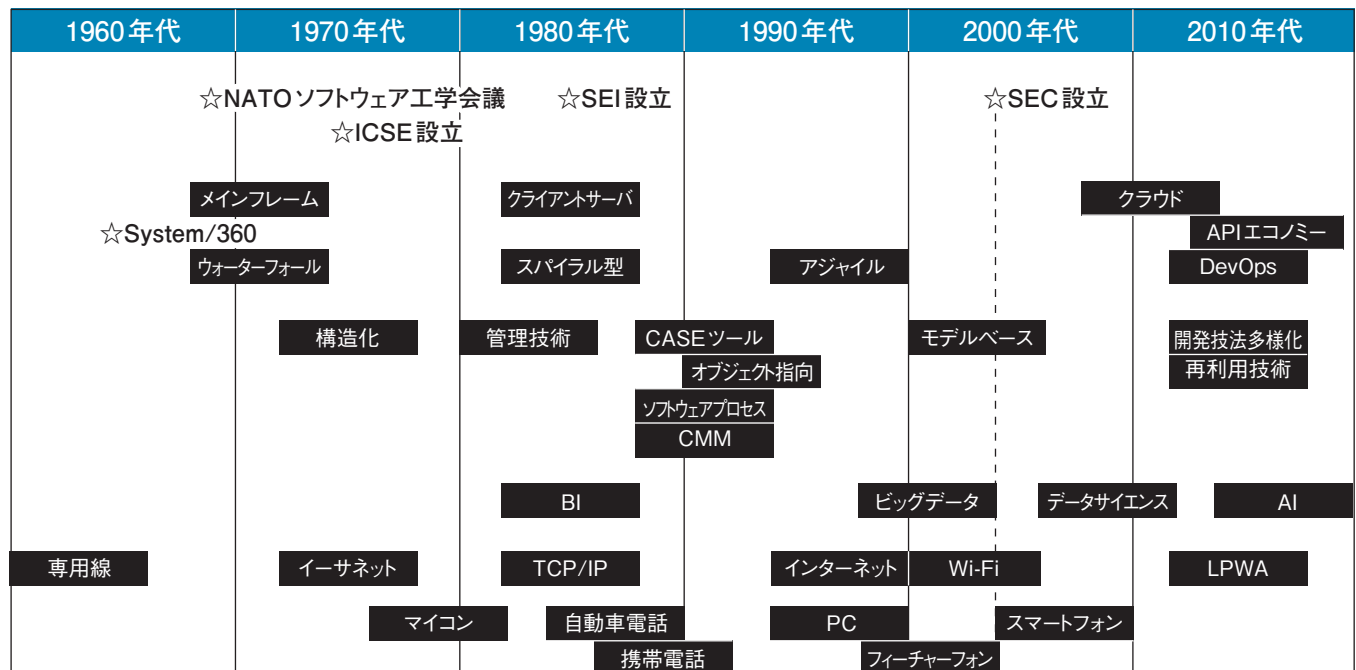
SEC journalは本号で50号を迎えた。ソフトウェア・エンジニアリング・センター（SEC）がIPA内に誕生したのが2004年10月、SEC journal創刊号が発行されたのが2005年1月であるので、SEC設立と同時に誌の作成に取り組み始めたと言える。

SEC journalはソフトウェア・エンジニアリングの啓発と共に産業界での国際競争力強化や技術力の向上を目指して、主にソフトウェア開発の現場に従事する技術者へ向けて、実証的な論文や関連する技術・手法・事例の紹介、また、SECの活動成果などを掲載してきた。50号を迎え、本稿では歩んできた時代背景、誌面の振り返り、論文採録状況について記してみたい。

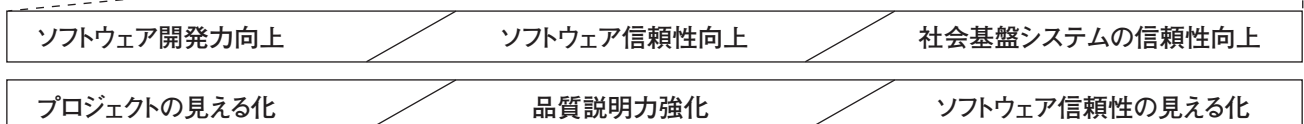
1 背景

SEC journalが歩んできた時代背景について、ソフトウェア工学やIT技術関連の変遷について振り返ってみたい(図1参照)。ソフトウェア工学は1968年にドイツのガーミッシュで開催さ

れたNATOの会議でSoftware Engineeringという名を冠したことから始まると言われている。ここではソフトウェア工学の歴史を振り返り次章のSEC journalの歩みを投じてみたい。



SECの取り組み



注：帯部分が開始終了の時期を特定するものではありません

図1 関連技術の変遷とSECの取り組み

2005年

創刊号

(2005年1月25日)

はじまりは2005年



「創刊にあたって」より抜粋

SEC(ソフトウェア・エンジニアリング・センター)は2004年10月1日、IPA(独立行政法人情報処理推進機構)の中に設置されました。産学官連携の拠点として、「高品質のソフトウェアを効率よく開発する手法を確立し、普及させる」ことを旗印に活動いたしますので、読者各位のご支援とご協力をよろしくお願いたします。活動の一環として「SEC journal」を発行することにしました。ソフトウェア・エンジニアリングに的を絞り、学術論文のみならず、プラクティカルでエンピリカルな情報を幅広く発信したいと考えています。

第2号

(2005年4月25日)

投稿論文の掲載開始

第3号

(2005年8月5日)

1	巻頭言 経済産業省 情報処理政策 情報処理部長 菅野 昌之
2	所長対談：西澤地二 ローランド・ベルガー取締役共同会長 経営とIT
6	論文 プロジェクトデータ分析の指針と分析事例 東海大学 古山 恒夫
14	品質マネジメントシステムの再構築：競争優位性の獲得に向けて 野中 誠 (東洋大学)
22	ものづくり戦略論とアーキテクチャ—ソフトウェア・アーキテクチャの測定と分析— 立本 博文 (東京大学 ものづくり経営研究センター)
30	技術解説 ソフトウェア・アーキテクチャに関する技術動向 森田 謙二
36	コンプライアンス/ソフトウェア開発における要求品質の確保 小原 謙二 (東洋大学)
42	組込みスキル標準解説 ECSS 大塚 隆之
50	組込みターゲットアーキテクチャ 大野 隆雄
56	組織紹介 情報処理学会 組込みシステム研究グループ 高田 幸生
58	社団法人 日本情報システム・ユーザー協会 JIAS 佐藤 隆

3号にて投稿論文を初めて掲載した。(敬称略)

- プロジェクトデータ分析の指針と分析事例 東海大学 古山 恒夫
- 品質マネジメントシステムの再構築：競争優位性の獲得に向けて 東洋大学 野中 誠
- ものづくり戦略論とアーキテクチャ—ソフトウェア・アーキテクチャの測定と分析— 東京大学 ものづくり経営研究センター 立本 博文

の3本であった。以降、本50号までに63本が採録、掲載となった。SEC journal論文に関しての振り返りは本稿の後半も参照していただきたい。

第4号

(2005年11月4日)

2006年

第5号

(2006年1月31日)

複雑化・短期開発が要求される組込みソフトウェア

1	巻頭言 経済産業省 情報処理政策 情報処理部長 菅野 昌之
2	所長対談：David Ward MISRAプロジェクトマネージャ 複雑化と短期開発要求が進展する組込みソフトウェアの課題を解決する道を示す
6	論文 第4世代のテストプロセス 山崎 隆夫
16	企業横断的収集データに基づくソフトウェア開発プロジェクトの工数見積り 大和 隆雄、高田 幸生、門田 隆人、松村 祥子、松本 隆一、梶原 隆夫
26	通信ソフトウェア開発におけるプロセス改善のためのフェール品質に注目の主要な活動要因の抽出 梶原 隆夫、安藤 淳、本野 泰、梶原 隆
36	技術解説 ソフトウェア確率信頼測定法の最新動向 ISO/IEC 9126の観点からフェールモデルを構築する 高田 幸生
44	Project Report 先進ソフトウェア開発プロジェクト Part II 松澤 謙、神谷 芳樹、磯口 史

複雑化と短期開発要求が進展する組込みソフトウェアの課題を解決する道を示す

組込みソフトウェアの開発は、その特性上、ハードウェアの設計・開発と並行して、ソフトウェアの開発も同時並進で行われる。その中で、開発者の負担軽減とソフトウェアの品質向上を両立させることが、本誌の目的である。本誌は、開発者の負担軽減とソフトウェアの品質向上を両立させるための主要な活動要因の抽出と、その解決策について、開発者の経験に基づき、実践的な情報を提供する。本誌の発行は、開発者の負担軽減とソフトウェアの品質向上を両立させるための重要な活動である。

当時、組込みソフトウェアは複雑に、そして大規模なものとなってきていた。その一方で産業界からは短期開発要求の度合いが増していた。そのような時代に、欧州自動車産業で生まれ、高信頼ソフトウェア開発のガイドラインとして知られるC言語のコーディング規約「MISRA C」を作成したMISRAプロジェクトのマネージャであり、MISRA Cステアリングメンバー委員長のDavid Ward氏に組込みソフトウェアをめぐる課題と、その解決策について伺った。



この時期SECでは、組込み系のソフトウェア・エンジニアリングや組込み系の人材育成に関する取り組みを立ち上げ活動を進めていた。

その後SECでは、ソースコード品質向上を目的としてコーディング作法ガイド(ESCR)を策定、公開しているが、ESCRとMISRA Cとで相互引用や、改定時のレビューを行うなど、MISRAと連携した活動は現在も続いている。

第6号

(2006年4月28日)

第7号

(2006年9月11日)

第8号

(2006年11月30日)

2007年

第9号

(2007年2月26日)

第10号

(2007年5月28日)

第11号

(2007年9月28日)

2008年

第12号

(2008年1月15日)

第13号

(2008年2月29日)

第14号

(2008年9月30日)

第15号

(2008年12月26日)

2009年

第16号

(2009年3月31日)

ETSS
特集号

(2009年4月30日)

見積り手法

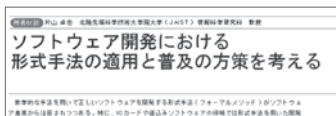
1	巻頭言
2	巻頭対談 (社団法人 情報サービス産業協会 会長)
3	巻頭対談: 長谷川英一 (社団法人 電子情報技術産業協会 常務理事)
4	「失敬学」の視点から
6	ソフトウェア開発の課題と解決策を考える
16	目次
17	目次
18	目次
19	目次
20	目次
21	目次
22	目次
23	目次
24	目次
25	目次
26	目次
27	目次
28	目次
29	目次
30	目次
31	目次
32	目次
33	目次
34	目次
35	目次
36	目次
37	目次
38	目次
39	目次
40	目次
41	目次
42	目次
43	目次
44	目次



SEC開設当初の主要取り組みの一つが見積り手法の確立であった。プロジェクトの規模、工数・コスト、工期の見積りの妥当性確保は、ユーザ企業とベンダ企業の双方にとって、必要なシステムを適切な価格で必要な時期に入手したり、適正な利益を確保する点で、プロジェクトの成否に直接つながる。この点で見積り活動はプロジェクトや組織全体にかかわる重要なものであるが、当時は、勘と度胸で見積りがなされ実績が予測から大きく外れることも少なくなかった。その結果ユーザとベンダ間で納得が得られていないという状況があった。この問題意識に基づいて規模、工数・コスト、工期を対象に、精度の高い(または、妥当な)見積りとは何か、その実現方法は何かをSECでは追求してきた。「ITユーザとベンダのための定量的見積りの勧め」(2005年発行)をはじめとし、「ソフトウェア開発見積りガイドブック」(2006年発行)などによる企業での見積り手法の事例紹介やCoBRA法やEASE協調フィルタリング法といった、先進的な手法に関してSECが共同研究の一環として実証実験した手法についての解説などにより、自社のデータに基づいた見積りモデル構築、定着を図った。

形式手法

1	巻頭言
2	巻頭対談: 長谷川英一 (社団法人 電子情報技術産業協会 常務理事)
3	巻頭対談: 片山卓也 (北陸先端科学技術大学院大学 JAIST) 情報科学研究所 教授
4	ソフトウェア開発における形式手法の適用と普及の方策を考える
6	論文
6	メモリリーク検出システム trace
16	SPICE Days 2007に参加して
20	Project Report
20	先進ソフトウェア開発プロジェクト Part III



数学的な手法を用いて正しいソフトウェアを開発する形式手法(フォーマルメソッド)がソフトウェア産業から注目されつつあった。とくに、ICカードや組み込みソフトウェアの領域では形式手法を用いた開発実績が出現していた。そうした背景を踏まえ、形式手法の研究に取り組んできた北陸先端科学技術大学院大学の片山卓也教授に産業界における形式手法の適用及び普及の方策について伺った。当時形式手法はまだ産業界に広く普及している段階ではなかった。ソフトウェア開発現場には、日本語で作成する仕様書の記述内容があいまいである事に起因する製品品質の低下やプログラム不具合対応などの手戻り作業に伴う開発コストの増加といった課題が存在していた。SECでは、こうした仕様記述の内容からあいまいさを排除し、誤解を招かない仕様書を作成する手法の一つとして形式手法に注目し、形式手法に関する調査や普及活動を推進することとなった。

ETSS特集号



2003年10月、経済産業省は組み込みソフトの開発力を強化すべく、産学官の有識者を集めて「組み込みソフトウェア開発力強化推進委員会準備会」を設置した。この委員会の活動の結果、組み込みソフト開発における人材育成を目的とした組み込みスキル標準、つまりETSS (ETSS : Embedded Technology Skill Standards) を設定する必要性とその緊急性が確認された。2004年10月からはSECに引き継がれ「組み込みソフトウェア開発力強化推進委員会」として本格的な標準化活動に入り、ETSSの正式版を公開した。その後ETSSの完成度向上と企業での導入実証実験、実際の導入に向けた支援などを行ってきた。これら活動成果をまとめ、ETSSの普及・促進を目指して、ETSSの概要説明、ETSS導入推進企業の実例紹介などを掲載した別冊をETSS特集号と題して発行した。

第17号
(2009年6月30日)

安心安全なIT社会を目指して



148	特集 安心・安全なIT社会の実現に向けてその課題と解決策を考える
152	特集 SEC2008年度活動概要
156	◆ソフトウェア系
160	ヒソホメプロジェクト改善案
164	実用・実践ソフトウェア系
168	◆セキュリティ系
171	組込みソフトウェアエンジニアリング系
175	◆基礎系

インターネットが普及し、自動車や家電などの製品に組み込みソフトウェアが搭載される中、安心・安全なIT社会を実現するソフトウェア技術開発が重要な課題となっていた。そこで、ディペンダブル(頼れる)なOSを開発しようという挑戦的なプロジェクトの研究総括を務める株式会社ソニーコンピュータサイエンス研究所代表取締役社長の所真理雄氏を所長対談にお迎えし、安心・安全なIT社会を実現するためのソフトウェアや新しい科学的な方法のあり方について伺った。

IoTの時代を迎えた現在、あらゆる装置や仕組みというものはソフトウェアで実現されており、安心・安全なIT社会の実現に向けてはソフトウェアの品質確保はますます重要になってきている。そのため「安心・安全なIT社会の実現」へのアプローチに関してはSECの各種取り組みのベースとなるものであり、SEC journal誌上でも特集や活動報告などで何度も登場する。

第18号
(2009年9月30日)

定量的アプローチ

222	特集 実践活用へ向けて活発化するSECの『定量的アプローチ』
226	専科解説 発注者視点からの工程エラー管理指標の導入
234	読者調査 業界調査員向けユーザーベンダ層での開発データ共有
244	論文 AQUAMarine: 定量的管理計画立案システム
252	読者調査 組込み人材の教育プログラム開発

特集
実践活用へ向けて活発化するSECの『定量的アプローチ』

SECでは発足以来ソフトウェア開発における定量データの重要性に着目し、その計測・収集・蓄積・分析、そしてその結果の可視化、活用に焦点を当てた活動をしてきた。世界的にも定量データに基づいたソフトウェア・エンジニアリングの有効性への認識が高まってきた時期でもあり、「実践活用へ向けて活発化するSECの定量的アプローチ」と題した特集を組み、現場でのプロジェクトマネジメントへの活用を推進した。「ソフトウェア開発データ白書」や計測ツール、診断ツールなどを紹介しSEC成果の自社での活用を後押しした。

2005年に初めて出版したデータ白書は、白書としての継続性、連続性を確保しながら、最新版は「ソフトウェア開発データ白書2016-2017」(2016.10.1発行)として継続発行しており、4067件のプロジェクトデータを使用した分析を実施している。また、データ件数の多い金融・保険業、情報通信業、製造業に関しては業種ごとの分析も実施し、それぞれ別冊化するに至った。組込み系に関しては「組込みソフトウェア開発データ白書」を発行している。

**ESxR
特集号**
(2009年11月16日)

ESxR特集号

第19号
(2009年12月28日)



品質の高いソフトウェアを作るにはどのような作業をどのように進めたら良いのか。ソフトウェア開発の作業や品質を見えるようにするにはどのようにしたら良いのか。

SECでは、2004年の設立以来、産学官連携の立場を活かし、業界の協力を得て開発現場のベストプラクティスを収集・整理して、組込み開発リファレンス集として取りまとめた。これまでにまとめた組込み開発リファレンス(ESxR: Embedded System development exemplar Reference)はこんな疑問に答えるものである。

当時SEC主催のセミナーなどを通じて普及を図ろうとしたが、ご希望される方が多く、全員をお招きすることができない状況であった。

また、セミナーでは時間の関係から、内容は技術的なポイントが主となり、導入の考え方や基本的なコンセプト、背景などを説明できていなかった。

こうした点を踏まえ、ESxRを背景からご理解いただき、実際に現場に導入していただくことを目指してESxR特集号を企画・発行した。

2010年

第20号
(2010年3月31日)

第21号
(2010年6月30日)

第22号
(2010年10月25日)

2011年

第23号

(2011年1月14日)

ユーザビリティ



162	所長対談 藤原 晴彦 株式会社U'eyes Design 代表取締役/ NPO法人 人間中心設計推進機構 事務部長
	ユーザビリティを基軸にして ITシステムの安心・安全を実現する手法を考える
166	特別記事 SECにおける国際学術活動展開 Part1 「ITプロジェクトの可視化」を中心とした開設以来6年間の軌跡
174	Part2 「ITプロジェクトの可視化」を中心とした国際学術活動の6年間の軌跡

ITシステムが社会に広く浸透し、ITシステムの安心・安全を担保することの重要性が増している。そうした中、従来は、信頼性の高いシステムが安心・安全なシステムである、という考え方でソフトウェアが開発されてきたが、信頼性の高さを追求することに加えて、ユーザビリティを考慮したユーザ中心設計が、ITシステムの安心・安全を実現する上で重要な鍵になると考えられるようになった。また、ユーザは製品やサービスの仕様だけでなくワクワク感などといったエクスペリエンスを期待するなど要求も多様化し、このような対応も必要になっていた。

23号では、株式会社U'eyes Design 代表取締役/NPO法人 人間中心設計推進機構の鱗原晴彦氏にユーザ中心設計の方法とITシステムへの適用可能性についてお話を伺った。今振り返ると、対談当時IoTというワードこそ登場していないが、「モノとモノがつながる時代に求められる設計とは」と、正にIoT時代のシステム開発の難しさと何をすべきなのかをお話いただいていることがわかる。

信頼性説明責任



ソフトウェアの信頼性に関する説明責任と品質監査のあり方について考える
平野 晋 × 松田 晃一

2009年米国で日本車急加速問題が起き、電子制御装置の不具合が疑われた。米高速道路交通安全局 (NHTSA) と米航空宇宙局 (NASA) による10カ月の調査結果で、電子制御装置ではいかなる問題点も見つからなかったという結論に至ったが、製品・システムの供給者は、高度化・複雑化する製品・システムの品質確保に努めると同時に、信頼性をはじめとする品質に関して利用者に十分な説明を行う責任を果たす必要があることが認識された。

SECにおいても、この時期「ソフトウェアの信頼性向上」「品質説明力の強化」を中期計画に掲げ活動を進めていた。24号の所長対談ではソフトウェアの信頼性に関する説明責任と品質監査のあり方について中央大学総合政策学部の平野晋教授にお話を伺った。

第24号

(2011年3月31日)

災害に強いシステム



50	所長対談 藤原 晴彦 株式会社U'eyes Design 代表取締役/IT企画部長
	災害に強い情報システムのあり方を考える
54	特別記事 SEC2010年度活動概要
56	◆統合系 統合系プロジェクト設置の狙いと取り組みについて

日本は、この年3月11日に未曾有の大震災に見舞われた。災害が起きたとき、インフラや業務を素早く復旧させるためには、重要な基盤である情報システムが確実に機能することが求められるが、東日本大震災に際して、迅速かつ的確な業務の遂行を続けることができた東京海上日動火災保険株式会社 執行役員・IT企画部長 澁谷裕以氏に被災地域で素早く業務を立ち上げることを可能にした情報システムと、その強さの根幹にある同社の「抜本改革」と「アプリケーションオーナー制度」について、お話を伺った。

SECでは、IT-BCPにおいて情報システムの高回復力を確保することの重要性を再認識し、ITサービスの回復力(レジリエンス)の底上げを目的に「情報システム基盤の復旧のための対策調査報告書」のとりまとめを行うと共に、ITサービス継続計画策定のための指針「高回復力システム基盤導入ガイド」の事例編などを取りまとめた。

第25号

(2011年6月30日)

第26号
(2011年10月13日)

2012年

第27号
(2012年1月12日)

第28号
(2012年3月30日)

第29号
(2012年6月29日)

第30号
(2012年9月28日)

第31号
(2012年12月14日)

2013年

第32号
(2013年3月1日)

第33号
(2013年7月31日)

第34号
(2013年9月30日)

2014年

第35号
(2014年1月31日)

第36号
(2014年3月31日)

第37号
(2014年7月1日)

第38号
(2014年9月30日)

別冊
10周年
特別号
(2014年11月1日)

2015年

第39号
(2015年1月31日)

プロセス改善



日本発アセスメントモデルのSPEAK-IPA

SPEAK-IPAは、プロセス改善及びプロセス能力判定のためのアセスメント体系を規定する国際規格 (ISO/IEC 15504) に準拠した日本発のアセスメントモデルである。新日鉄ソリューションズのアセスメントツールSPEAKと情報サービス産業協会 (JISA) のアセスメントモデルSPINACHをベースに2007年に初版を発表した。

その後、SPEAK-IPAの有効性を評価する実証実験とプロセス改善推進者教育のコンテンツやカリキュラムを確認する実証実験の実施、SPINA³CH (スピナッチ・キューブ) 自律改善メソッドの公開、セミナーなどを通じた普及活動を精力的に実施してきた。プロセス改善推進者、アセッサの育成と認証に活動はつながっていった。

共通フレーム



共通フレームは、ソフトウェア、システム、サービスへの従事者 (利害関係者) が、言葉の意味 (範囲) の解釈の違いによるトラブルを予防する (誤解を招かぬようにする) ため「同じ言葉話す」ことができるよう共通の枠組みを提供し、システム及びソフトウェアの構想から開発、運用、保守、廃棄に至るまでのライフサイクルを通して必要な作業内容、役割等を包括的に規定したものである。開発方法論には依存しない。発表当時、とくに二者間契約時のお互いの役割や分担を明確にしたことは特筆すべきで、画期的なものであった。このソフトウェアを中心としたシステムの取引に関する共通フレーム (共通フレーム94) はソフトウェア・ライフサイクル・プロセス (SLCP) の国際規格であるISO/IEC 12207に先駆け発表されたものであった。歴史的には、1994年に「共通フレーム94」が発表され、1998年の「共通フレーム98」、2007年10月に「共通フレーム2007」を経て、システム監査プロセスなどを追加した「共通フレーム2013」が2013年に刊行された。

創刊10周年特別号



SEC設立10周年の節目にあたり、これまでの10年にわたるSECの活動を振り返ると共に、今後の展望について取りまとめた特集号である。回顧だけではなく、この間の技術の進歩に伴い世の中の仕組みに浸透し、産業や私たちの生活に変革を起しつつあるIoTの大きなムーブメントについて各界有識者から寄稿をいただいた。

SEC成果報告の一覧やSEC journal掲載論文一覧、SEC journalバックナンバー一覧なども取りまとめてあり検索手段としても活用いただきたい。

システムズエンジニアリング

第40号
(2015年3月1日)



利用者からの要請や期待も高度化し、製品やサービスは複雑化、高度化の一途をたどっている。更に近年、従来の業種をまたぐような新たな領域のビジネスが目立っており、多様な専門領域にまたがった複雑な取り組みが増加しており、システムの企画や開発を進める上での影響範囲が広がり、困難度が高まっている。

このような状況においては、従来までのアプローチでは解決できない、期待される成果を十分に得られない、思わぬ不具合を発生させるなどの問題が起こり得る。これらは今求められているシステムと従来までのモノづくりの間に何らかのギャップがあるためと考えられ、そのギャップがどこにあるかを見極め対策していく必要がある。SECではこれらを考えていくベースとして「システムを全体として捉える」ことによるメリットに注目し、そのプラクティスの集積であるシステムズエンジニアリングの理解を進め新しい時代のシステム開発に対する考え方、適用のための有益な情報を発信すべく活動を開始した。

40号ではシステムズエンジニアリングに関する国際組織であるINCOSEの中心メンバ、ジェームズ・マーティン氏との対談を、48号では巻頭言、所長対談を含めた特集号として紹介しシステムズエンジニアリングの理解促進を図った。

第41号
(2015年7月1日)

システムズエンジニアリングの最前線
—技術・人・社会を含めて考えるシステム
開発と運用—
The Nanque Corporation
ジェームズ・マーティン X 松本 隆明



第42号
(2015年9月1日)

つながる世界

第43号
(2015年12月1日)



IoT時代が到来し、ソフトウェアが組み込まれた製品・システムは日常生活に無くてはならない社会基盤となってきていた。更に、そのような製品・システム同士がネットワークを介してつながって、新しいサービスを生み出す世の中へと変遷する。これをSECは「つながる世界」と呼んだ。

このつながる世界では、色々な製品・システムがつながって新しい製品・サービスが生み出され、我々の生活のますますの利便性向上や新しいビジネスの創出に寄与するものと期待されている。一方で、色々な品質の製品・システムが氾濫し、それが勝手につながってしまうと、安全上あるいはセキュリティ上の問題を引き起こす危険がある。そこで、SECは本格的なつながる世界の到来に備えて、つながる世界に潜むリスクを防止する取り組みを進めている。SECの取り組みや各産業界の期待や課題を取りまとめ特集とした。

つながる世界に向けた取り組み
SECソフトウェアグループリーダー 中尾 昌善 原 真次

2016年

第44号
(2016年3月1日)

第45号
(2016年7月1日)

第46号
(2016年9月1日)

第47号
(2016年12月1日)

2017年

第48号
(2017年3月1日)

第49号
(2017年7月1日)

上記でご紹介した記事を含め、創刊号からすべての誌面がダウンロード可能になっている。ぜひ活用いただきたい。

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

3 SEC journal論文

SEC journal発行の目的の一つが、論文を募集し誌面に掲載、普及を図ることである。ソフトウェア・エンジニアリングの効果的な活用による開発力強化、産業競争力強化を目指し、アカデミックな取り組みと開発現場での取り組みの両方を対象に、開発現場での実務で有効な情報共有を図っている。

SEC journal論文テーマとしては

- 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

などであるが、各種学会が扱う論文とは重要視する観点が異なる。(図2参照)

	目的	特徴	重視ポイント
学会論文	研究推進	先進的研究成果	先進性
IT技術誌	ビジネス 知的満足	実証不要	トレンド
SEC論文	産業競争力向上	実証論文	実用性 有効性

図2 SEC journal論文の位置付け

開発現場での実務への適用に有効な情報共有を図るため、実用性、有効性を優先し、必ずしも新規性は求めない。事例研究や replication studiesなども対象となり、自らの業務に関連した社会人ドクターの投稿先としても積極的に活用していただきたい。

SEC journalには産学連携の場としての機能も求められるところではあるが、この点はまだ十分ではないと感じている。新たな技術や手法の開発への適用が誌面上で提案され、それを見て、自社への導入の試みが生まれ、現場での改善や根付かせるための工夫などが更に誌面で紹介されるといった循環を作っていきたい。

論文採録数の推移を図3に示す(2017年度の数値は年度途中のため参考値)。創刊から50号までに63本の論文を掲載した(招待論文などを除き投稿論文のみをカウントした)。2010-2011年度頃は採録数が少なく査読付き論文の掲載がない号が3号連続となった時期もあった。近年は採録数が増加している。筆頭著者の所属で産学を分けてみた結果は累計ではほぼ同数、近年は学界からの論文の採録が産業界より若干多い結果となっている。近年の傾向の原因は、産業界からの投稿が減少傾向にあることと、論文執筆経験がない企業の執筆者の最低限の要件を満たしていない論文などは受付や査読を突破できないことなどが要因と考える。

SEC journal論文は、必ずしも独創性新規性を問わない。自らの業務での replication studiesなどは、産業界からならでの論文であると考え。企業の中で、論文作成の時間が取りづらいといった理由もあると思うが、その研究成果を世に還元することは産業界発展のためにも大切なことである。横展開可能な論文投稿を引き続きお願いしたい。

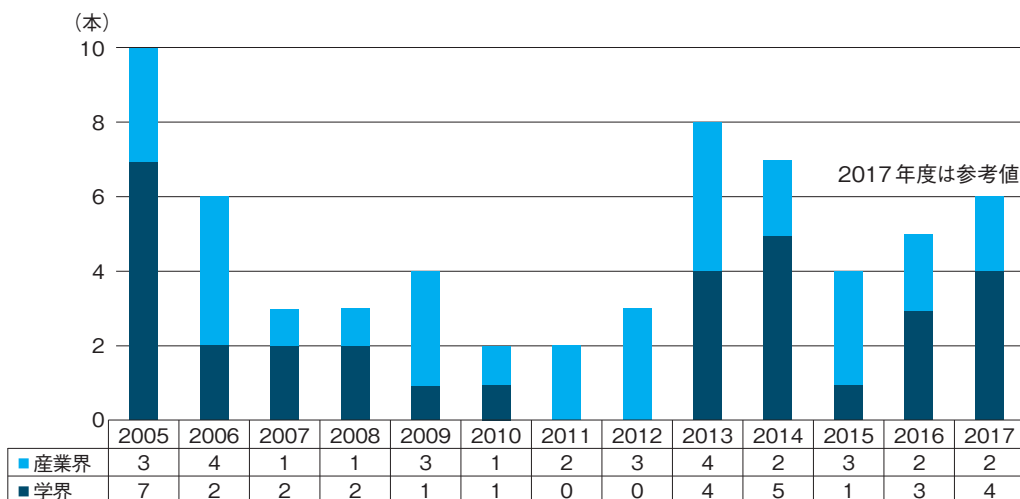


図3 年度別採録数

【参考文献】

P.Naur and B.Randell, editors. Software Engineering – Report on a conference sponsored by the NATO Science Committee, Garimisch, Germany, Oct. 1968., <http://homepages.cs.ncl.ac.uk/brian.randell/NATO/nato1968.PDF>

玉井哲雄 ソフトウェア工学の40年, 情報処理, Vol.49, No.7 (2008), pp.777-784

斎藤亨 ソフトウェアエンジニアリング講座1, 日経BP, 2007.

ガートナー 先進テクノロジーのハイブ・サイクル:2016年, <https://www.gartner.co.jp/press/html/pr20160825-01.html>

IEEE Software January/February 2016 (Vol. 33, No. 1), <https://www.computer.org/csdl/mags/so/2016/01/index.html>

SEC journal 創刊号~49号, http://www.ipa.go.jp/sec/SEC_journal/index.html#L1

定性的信頼性／ 安全性解析支援ツールの開発

土肥 正[※]岡村 寛之[※]

本研究は、FTA (Fault Tree Analysis), FMEA (Failure Mode and Effect Analysis), HAZOP (Hazard and Operability Study) などの定性的信頼性/安全性分析を支援するためのツール開発を行う。具体的に、設計や障害事例などに関する過去の情報を非構造型データとしてデータベースに蓄積し、FTA, FMEA, HAZOPに現れる故障モードやガイドワードなどのキーワードと、対象とするシステムの設計情報 (UML / SysML) を手がかりに、関連する過去の事例を抽出する枠組みを提案する。更に、それらを重要度に従ってランキングする。これは安全性分析者が設計段階における工数削減に寄与する。

Development of a Support Tool for Qualitative Reliability / Safety Analysis

Tadashi Dohi[※], and Hiroyuki Okamura[※]

In this article, we develop a support tool for qualitative reliability/safety analysis such as FTA (Fault Tree Analysis), FMEA (Failure Mode and Effect Analysis) and HAZOP (Hazard and Operability Study). The developed tool contains the database for unstructured data; UML/SysML diagrams and prior incidents, and picks up the related prior events from given failure modes of FTA and FMEA, guidewords of HAZOP and UML/SysML diagrams. Based on these events, our tool also provides the ranking of events with their severities. This would be helpful for safety analysts to reduce their efforts in the phase of system design.

1 はじめに

ソフトウェア内の欠陥がシステムの安全性を脅かす事例は枚挙に暇がない。ソフトウェアの欠陥が発生する要因は幾つか存在するが、設計時における「考慮漏れによる欠陥」は、フィールドにおいて最も深刻な障害を引き起こす可能性のある欠陥であることが多い。同時に、この種の欠陥は最も取り除くのが困難な欠陥の一つとして知られている。一般的なシステム設計では、システムが配置される外部環境において発生し得る事象を網羅的に列挙し、その事象が起きた場合にシステムがどのようなレスポンスを示すかを分析する。これは、単に外部環境要因とシ

ステム間で相互に関連する事象を列挙するだけでなく、システム内における特定のコンポーネントが故障して初めて外部環境に伝搬されるような、システム内部のコンポーネント間における交互作用効果も注意深く分析する必要がある。このように、考慮漏れによる欠陥は常にヒューマンエラーによって引き起こされる欠陥であり、現実の開発期間内で完全に防ぐことは極めて困難である。

考慮漏れによる欠陥を防止するためには、すべての事象を網羅的に分析することが重要となる。とくに、安全性を阻害する故障はこれまでに未経験であることが多く、事象の発生をあらかじめ想定することが難しい。安全性分析は「想定外を想定する」

※ 広島大学

という難しい課題に挑戦しなければならない。具体的な安全性分析手法として、FTA (Fault Tree Analysis), FMEA (Failure Mode and Effect Analysis), HAZOP (Hazard and Operability Studies) などがある。FTAでは、ある発生してはならない障害(例えば、人命にかかわる事故など)が発生する要因や事象をトップダウン的に洗い出す手法であり、最終的には、故障木と呼ばれる事象と原因の関係を記述するグラフモデルを構築する。FMEAは、FTAとは逆にボトムアップ的な手法であり、あるコンポーネントで典型的な故障(故障モード)が発生した場合に起こる影響を積み重ねることにより、システム全体にどのような障害が起きるかを分析する。HAZOPはシナリオに基づいた分析手法である。その考え方は故障モードを用いたFMEAによく似ており、故障モードの代わりにガイドワードと呼ばれるプロセスに対する異常な操作や状態を用いて、プロセス異常がシステムに与える影響を分析する。これらは、信頼性や安全性を数値的に定量化するのではなく、信頼性や安全性を損なう要因を定性的に分析することから、定性的信頼性/安全性分析手法と呼ばれる。

実務レベルで、これらの定性的信頼性/安全性分析を行う場合、二つの大きな問題がある。一つ目は、分析結果が分析者の経験に大きく依存するという問題である。FTA, FMEA, HAZOPは、本来、経験したことのない事象であっても考慮漏れによる欠陥が入らないような設計を行うための手法であるが、システムに関する知識が豊富な分析者と経験が少ない分析者では、結果として作成された分析結果が大きく異なる。FTA, FMEA, HAZOPは本質的に、事象の原因、故障モードとその影響、ガイドワード

などをきっかけとして、分析者への連想を促す手法と言える。つまり、分析者の知識や想像力などの個人的な能力による所が非常に大きいため、最終的な分析結果の品質に対するばらつきが大きく、「経験の少ない分析者が不十分な分析を行う」というリスクが依然として残る。二つ目は分析に要するコストの問題である。FMEAやHAZOPはボトムアップ的な分析手法であり、すべてのコンポーネントに対象となるすべての故障モードを適用する、あるいは、すべてのプロセスに対してすべてのガイドワードを適用することが推奨される。しかしながら、その組み合わせはコンポーネント数の増加やプロセスの複雑化によって指数関数的に増加する。先にも述べたように、定性的信頼性/安全性分析は、そもそも分析者の連想に依存した試行錯誤的なマニュアル分析であるため、爆発的に増加する組み合わせをすべて網羅するためには膨大な人的コストがかかる。そのため実務では、分析者の経験則から「最も危険なコンポーネントやプロセス」を経験的に特定し、その部分の信頼性・安全性を集中的に分析することが多い。この点においても、分析者の経験不足による考慮漏れのリスクが常に存在する。

本論文では、自動車制御のような高い信頼性/安全性が必要なシステム開発において、このような問題点を部分的に解決若しくは緩和する目的で、定性的信頼性/安全性分析を支援する汎用的な技術の開発と定性的信頼性/安全性分析を支援するツールの開発を行う。具体的に、過去の情報(設計やHAZOPによる安全性分析結果など)を蓄積し、対象とするシステムの設計情報(UML / SysML)で注意すべきコンポーネント(プロセス)

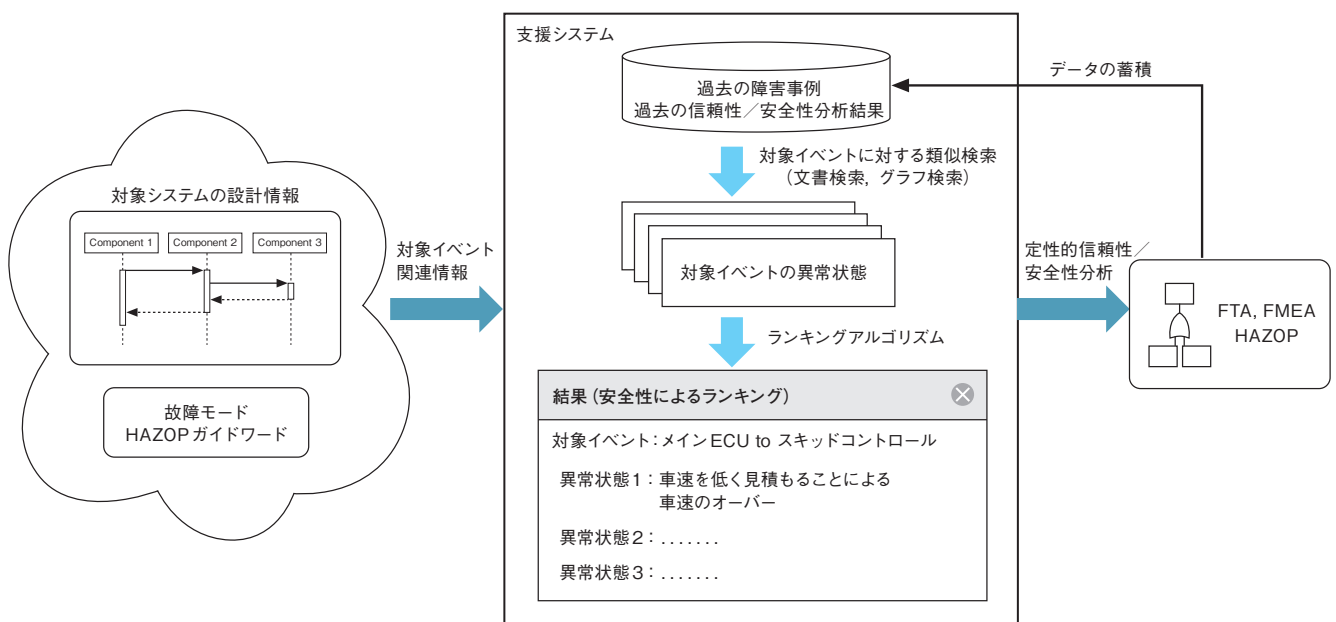


図1 ツールの概要

を重要度に従ってランキングするための学習アルゴリズムを開発する。とくに、定性的信頼性/安全性分析はシステム設計の早い段階で行う必要があるため、ソフトウェア開発の上位レベルでの定性的信頼性/安全性分析を支援することを目的とする。

2 ツールの概要

本論文では、対象システムの設計情報と過去の情報(安全性分析結果)から対象システムの設計上で重要なコンポーネント(プロセス)を推定し、優先順位を付けて分析者に提示する支援ツールの開発を目標とする。支援ツールは過去のHAZOP分析結果と現在の設計の類似度を評価し、その類似度に基づいた重要

度スコアリングを行う(図1参照)。そのため、過去の分析結果を学習するフェーズと現在の設計を評価(スコアリング)するフェーズに分けられる。図2及び図3は作成する支援ツールのユーザースペースを示している。図2は過去の分析結果を学習するフェーズであり、(1)一般的な文章からの単語(タグ)の抽出及びベクトル化、(2)定性的信頼性/安全性分析に関連した文章からのタグの抽出及びベクトル化、(3)過去の設計書及び分析例の分解と登録されている単語による特徴付け(タグ付け)を行う。図3は現在の設計書の入力により評価を行うフェーズを表しており、(1)設計書の分解及びタグ付け、(2)タグに基づいた過去の分析例との類似検索、(3)類似度と重要度のスコアリングを行う。

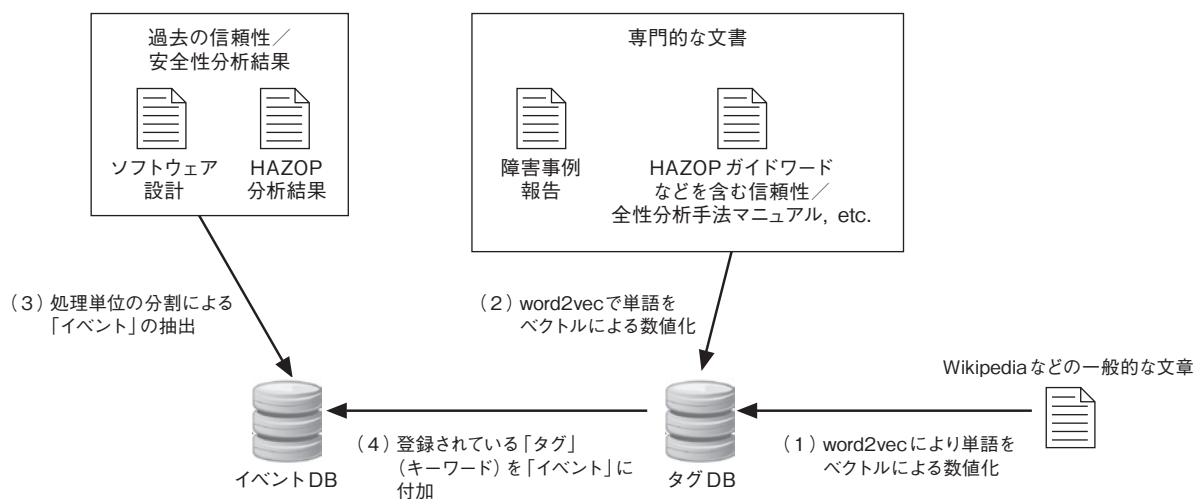


図2 学習フェーズ

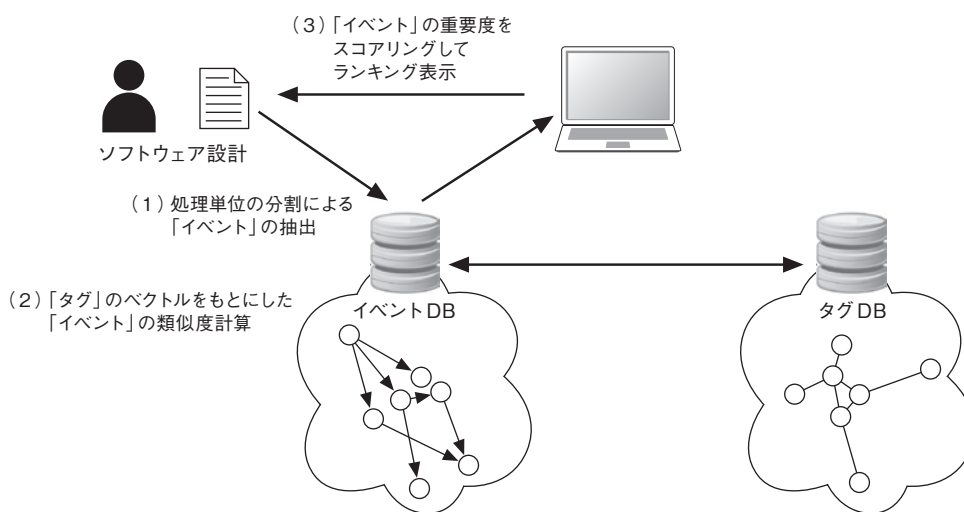


図3 評価フェーズ

3 類似度検索手法

検索手法の調査・開発では文章データとUMLによる図で記述された設計書を统一的に扱う必要があるため、データ構造の定義が重要になる。安全性分析結果とソフトウェアの設計情報という、一見異なるものの類似性(関連性)を評価する必要がある。つまり、一般のドキュメント間での類似検索技術を開発する必要がある。

データ構造については、設計書や安全性分析を構成する「文章」と、その「文章」を構成する「単語」に着目したデータ構造を定義することで、すべてのデータを统一的に扱う仕組みを構築する。また、類似性の評価についても「単語」に着目した類似性の定義を行う。

3.1 データ構造の定義

テキスト文書及びUMLシーケンス図を格納するデータ構造として図4に示すクラスを用いる。テキスト文書及びUMLシーケンスは表1の単位の「文章」に分解されEventクラスを生成する。

Eventクラスは、識別子(id)、名前(name)、対応するテキスト文書(description)、重要度(score)の属性を持つ。また、イベント間の関連を表すためのリンク(link)属性も持つ。更に、descriptionをもとに「単語」の抽出が行われTagクラスを生成する。Eventクラスに関連するTagクラスはそれぞれ文書内の単語の出現頻度に応じた重み(weight)を持つ。Tagクラスは、識別子(id)、キーワード(keyword)と後述するword2vec^[1]によって算出されたベクトル(vector)の属性を有する。descriptionから日本語の単語を抽出するためにMeCab^[2]による「わかち書き」を適用した。

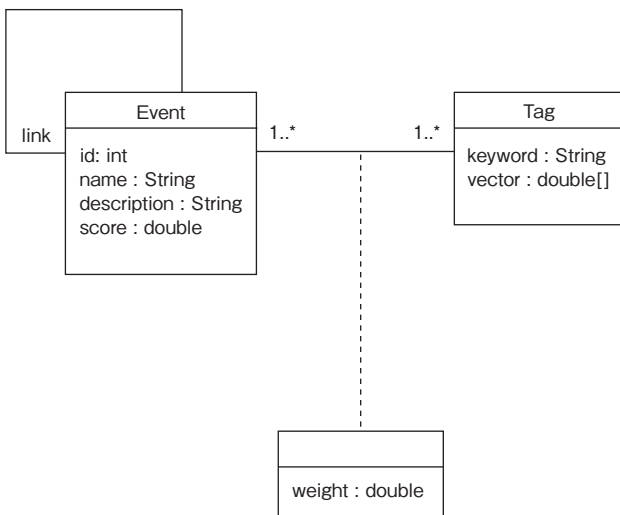


図4 データ構造のクラス図

表1 Eventクラスの単位

ドキュメント	イベント単位
テキスト文章	パラグラフ
UMLシーケンス図	メッセージ、オブジェクト、コメントなどUMLの構成要素単位
HAZOP分析	一つのガイドワードに対する分析結果

3.2 イベント間類似度の定義

Eventクラス間の類似度を算出するために、本論文ではタグの数値ベクトル化を行う。j番目のEventクラスの数値ベクトル v_j は、Tagクラスの数値ベクトル w_1, w_2, \dots (word2vecによって得られる)とEventクラスに対するTagクラスのTF-IDF値をもとに以下の式で得られる。

$$v_j = \sum_i \text{tfidf}_{ij} w_i.$$

ここで、 tfidf_{ij} はTagクラスiのEventクラスjにおけるTF-IDF値である。一般に、ある単語に対するTF-IDF値は文書中出现する単語の頻度(TF:Term Frequency)と、その単語を含む文書の頻度(DF:Document Frequency)から算出され、DFについては逆数の対数を取ったIDF(Inverse Document Frequency)を利用する^[3]。また、 w_i はTagクラスiのベクトルを表す。数値ベクトル化されたイベントの類似度に対しては、以下の式で定義されるコサイン類似度^[4]を用いる。

$$\text{sim}(A, B) = \frac{(\sum_{a \in S_A^n} a) \cdot (\sum_{b \in S_B^n} b)}{\|(\sum_{a \in S_A^n} a)\| \|(\sum_{b \in S_B^n} b)\|}$$

ここで、 S_A^n, S_B^n はEventクラスA及びBを中心にnホップ先までのイベントを含めた集合(a, bはそれらの要素(イベント)に対する数値ベクトルを表す)を表す。これはグラフ構造のデータに対する類似度(カーネル)定義^[5]を利用している。

3.3 Tagクラスのベクトル化

Tagクラスに対してはword2vecによるベクトル化を行う。word2vecはニューラルネットワークによって単語をベクトル化する手法を実装したツールである。具体的にはCBOW(Continuous Bag-of-Words)と呼ばれる図5に示すアーキテクチャを持つニューラルネットワークによって、自然言語で記述された個々の単語のベクトル化を実現する。

CBOWでは周辺の単語から適切な単語を予測する。図5の $w(t-2), w(t-1), w(t+1), w(t+2)$ は周辺の単語を表し、それらから目標とする単語 $w(t)$ を予測するようにINPUTとPROJECTION

間の重みを調整(学習)する. word2vecツールでは, わかち書きされた日本語を入力(コーパス)として, 単語をベクトル化する.

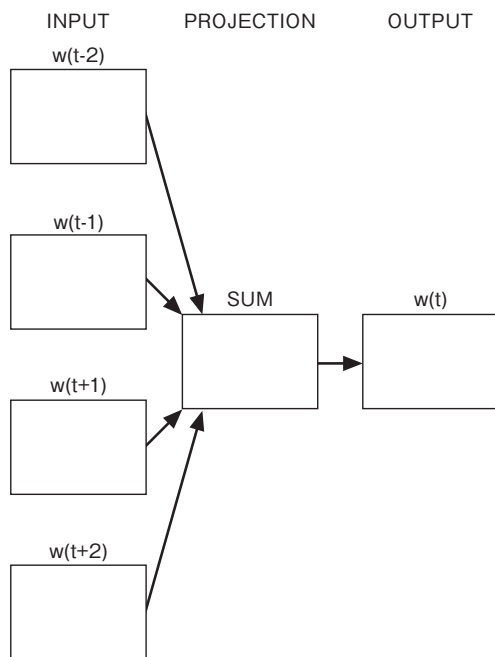


図5 CBOWのアーキテクチャ

3.4 重要度スコアリング

重要度の算出は, 文書(イベント)間の類似度と過去のHAZOP分析結果にあらかじめ付加されている重要度(HAZOPスコア)から行われる.

以下の記号を定義する.

$score(i)$: UMLシーケンスのイベント i の重要度

$sim(i, k)$: UMLシーケンスのイベント i とHAZOP分析結果 k の類似度

$score(k)$: HAZOP分析結果 k の重要度 (HAZOPスコア)

このとき, UMLシーケンスのイベント i に対して次の重要度スコアを算出する.

類似度が閾値 θ 以上のHAZOPスコアの平均

$$score(i) = \frac{\sum_k I(sim(i, k) > \theta) score(k)}{\sum_k I(sim(i, k) > \theta)}$$

ここで $I(\cdot)$ は指示関数を示す.

また, 上記を報酬としたマルコフ決定過程(MDP: Markov decision process)を用いたスコアリングアルゴリズム^[6]を用いる. MDPとは, 状態で行うアクションによって, マルコフ連鎖の推移確率が変化する確率過程を表し, 通常はマルコフ連鎖上で報酬構造を定義し, 報酬を最大にするアクションを行う.

今, ユーザがリンクをたどってノードを巡回するモデルを考える. Ω をノード全体の集合, ノード i のリンク先を Ω_i とする. MDPに基づいた枠組みでは, ユーザは各ノードのコンテンツの評価値に依存した最適な行動(移動)を行うものと仮定する. すなわち, ユーザがノード i を訪問しているときに選択できる行動を

$$A_i := \{\Omega_i, rnd\}$$

とする. ここで, Ω_i はリンクをたどって移動するノード, rnd はランダムなジャンプで別のノードへ移動することを意味する. ユーザは, これらの中で最も高い評価値が得られる行動を選択するものと仮定する.

ノード i を訪問することによって得られる報酬(重要度スコア)を r_i とする. 更に, 割引率 $0 < \beta < 1$ を定義して, n 回目の移動によって得られる報酬に β^n を乗じて価値を減じるものとする. この場合, より少ないノード移動でより多くの報酬を得ることが最適な行動となる. このMDPに基づいた枠組みを利用することにより, 単純にHAZOP分析結果との類似度だけで評価する場合よりも, UMLシーケンス上のリンク情報を加味した上での評価が可能となる.

具体的にMDPで各ノードをスコアリングするためには, 最適な行動を求める必要がある. 今, 次の記号を定義する.

$W^a(i)$: ノード i においてノード a への移動を選択し, 以後, 最適な移動を選択し続けた場合の総期待割引報酬.

$V(i)$: ページ i における最大総期待割引報酬.

これらを用いると, 最適性の原理^[7]から, 以下の最適性方程式を得る.

$$V(i) = \max_{a \in A_i} W^a(i),$$

$$W^a(i) = \begin{cases} r_i + \beta V(a), & a \in \Omega_i, \\ r_i + \frac{\beta}{|\Omega|} \sum_{i \in \Omega} V(i), & a = rnd. \end{cases}$$

上記の最適性方程式を満足する $V(\cdot)$ 及び $W(\cdot)$ を求めることで、総期待割引報酬を最大にするユーザの行動が決定される。最終的に、最適な行動が決定された後に、各ノードのスコアとして各ページに対する最大総期待割引報酬 $V(i)$ を用いる。またこれは、ページランクアルゴリズム [8, 9] の理論的な拡張にもなっている。

4 ツールの実装

開発したツールはGUIによるフロントエンド(図6参照)とデータベースから成る。GUIの作成にはJavaを用いた。開発したツールは、word2vec ツールによる出力をツール内のDBに登録する「タグDB登録」、過去の設計、過去のHAZOP分析結果、現在のUMLシーケンス図をDBに登録する「イベントDB登録」、登録されたイベントと類似するイベントを検索する「類似検索」、過去のHAZOP分析結果をもとに現在のシーケンス図のスコアリングを行う「重要度スコアリング」などの機能を有する。

「タグDB登録」ではword2vecから出力されたファイル(テキ

ストファイル)を読み込むことで、ツール独自の形式(バイナリ形式)に変換してタグを登録する。

「イベントDB登録」ではXML化されたドキュメントまたはドローツールから出力されるUMLシーケンス図のXMLファイルから、イベントのDB登録を行う。HAZOP分析結果をはじめとするドキュメント類は表計算ソフトウェアからXMLを作成して登録することができる。

「類似度検索」では、登録してあるイベント間の類似度に基づいた検索を行う。ここでは、コサイン類似度(COS similarity)及び関連するイベントを考慮したコサイン類似度(COS similarity with tags of one-hop neighbors)の2種類のイベント間類似度を用いた類似検索が実現できる。

「重要度スコアリング」は開発したツールの主要な機能であり、対象となるUMLシーケンス図とHAZOP分析結果を指定することで、シーケンス図上の各イベントに対する重要度を算出する。スコアリング手法には類似度を閾値として用いるものとMDPによるものをそれぞれ適用することが可能である。

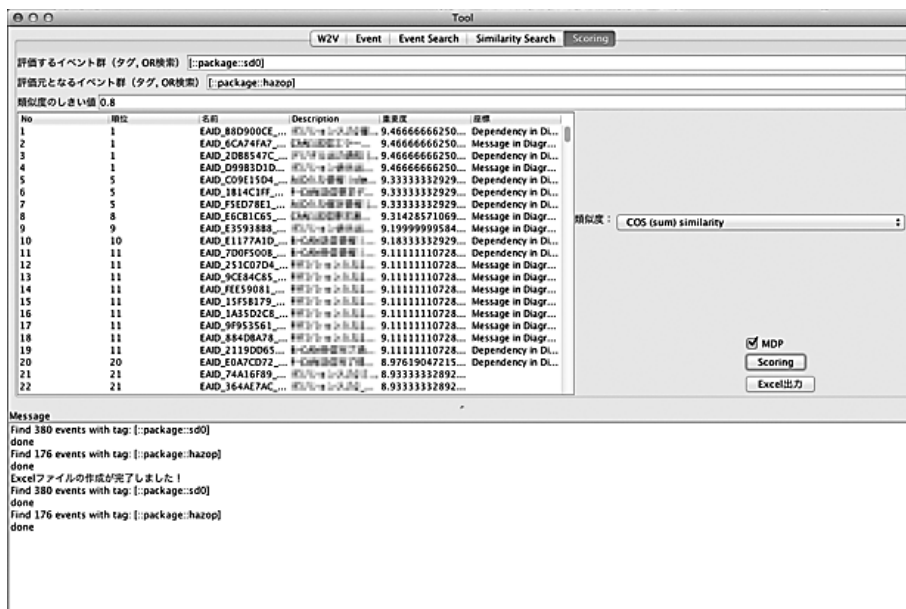


図6 ツールの実行画面(重要度スコアリング)

5 有効性検証

実際の企業(A社, B社)における設計者及び安全性分析者を対象に, 作成したツールの評価を行った. 実験においてはTagのデータを生成するため, Wikipedia, IPAが発行する電子書籍及び国際会議において収集した単語や文節データのテキストを利用し, 667,644単語に対するベクトル化を行った.

5.1 実証分析(A社)

A社で利用した設計データは, 自動車のシフト制御に関するシーケンス図であり, 2通りの異なる設計に基づいたシーケンス図(SD0, SD1)に対する重要度評価を行ったものである. 重要度スコアリングに必要となる過去のHAZOP分析結果は, A社で実際に同じアプリケーションドメイン(シフト制御)に対してHAZOP分析を行った結果(HAZOP1)と, JasPar(Japan Automotive Software Platform and Architecture)によって提供された自動車のトルク制御に関する安全性分析結果(HAZOP2)を用いて比較を行った. HAZOP1はシフト制御について安全性分析した結果であり, ポジション入力, CAN制御, デジタル出力における各機能, データ, ポートなどに対してHAZOPガイドワードを適用した障害シナリオを作成し, 安全性の検証を行っている. この分析結果に対して, 安全性分析者の手であらかじめ各障害シナリオの重要度を5段階(最も危険な障害を5)で評価してもらっていた. HAZOP2はJasParによる安全性分析例におけるHAZOP分析結果である. JasParではトルクセンサにおける安全性分析を行っており, 同様に上記のHAZOPガイドワードを適用して障害シナリオを生成している. 重要度についてはASIL(Automotive Safety Integrity Level)に応じた5段階の評価を行った. ASILでは, リスク分析を行った結果に応じて高いレベルの安全方策が求められる機能からD, C, B, A, QMの段階評価を付ける. 通常, これらの評価結果は過去の類似プロジェクトで既に分析したものを利用することが想定される.

HAZOP1を使った評価では, シーケンスSD0, SD1共に類似度の閾値を0.8と設定して評価を行った. SD0, SD1共にシーケンス中に全380イベントが存在したが, それぞれのシーケンスで100件程度のイベントを分析対象から外することができるという結果が得られた.

一方, JasParの結果(HAZOP2)を利用した場合, アプリケーションドメインが異なるため, 出現する用語が大きく異なり, 全体的に類似度が低下した. そのため, HAZOP1と比較すると有効なスコアリング結果が得られず, もととなるHAZOP分析と

シーケンスが同じアプリケーションドメインあるいは非常に近いアプリケーションドメインである必要があることが示された.

5.2 実証分析(B社)

B社で利用した設計データは, 空調制御に関するシーケンス図であり, このシーケンスに対する重要度評価を行った. 重要度スコアリングのもととなるHAZOP分析結果は同一ドメインのものが得られなかったため, JasParによって提供された自動車のトルク制御に関する安全性分析結果(HAZOP2)を用いて比較を行った. シーケンス図をイベントに分解して登録した場合のイベント数は24イベントであった. HAZOP2はASILによって重要度が与えられている.

HAZOP2を使った評価では, 先に述べたA社の分析と同様にアプリケーションドメインが異なるため, 閾値0.65と低く設定した. ツールによる評価では「EEPROM」, 「指定ブロック書込」, 「EEPROMミラーバッファ」及びその周辺のプロセスである「データ読出」, 「Verifyコマンド送信」, 「Disableコマンド送信」, 「Writeコマンド送信」, 「Enableコマンド送信」が比較的高い重要度として評価された. しかしながら, 実際のヒアリングでは, 「Backup Manager」などの項目が重要であるという指摘があり, ランキング精度についての問題が指摘された. その主な原因として, A社の場合と同様に異なるアプリケーションドメインのHAZOPスコアを用いたことが挙げられる.

6 まとめ

本論文では, 安全性分析において重要なコンポーネントを過去の安全性分析結果(HAZOP分析結果)から推定することによって, 安全性分析の工数削減を行うための支援ツールを開発した. とくに, 分析結果を構成する単語に注目して, 単語の頻度から文章間の類似度を算出する手法について考察し, それを用いた重要度分析手法について検討を行った.

過去において同様の研究はこれまでになされておらず本論文で行う試みは非常に挑戦的なものであった. 一方で, 通常の文書間における類似度分析とは異なり, ソフトウェア設計に使われる文章では, その文書量が圧倒的に少ない. 例えば, シーケンス図上のメッセージなどは短い文書で説明されていることが多く, その点を考慮した類似度評価を行う必要があったため, 本論文では単語そのものをベクトル化する word2vecと呼ばれる技術に注目し, それを利用した類似度評価と類似検索アルゴリズムを実装した.

類似検索についてはある程度の精度で実行できるが、実際的な問題としては、専門的な単語の登録など、辞書の整備をすることでより推定精度が高まることが期待される。一方で、現在の実装では、対象となるイベントの組み合わせを総当たりで類似度計算しているため、より多くの文書を扱う場合には計算を高速化するための工夫が必要になると考えられる。

重要度のスコアリングについては、分析対象のシーケンス図におけるすべてのイベントと過去のすべてのHAZOP分析結果の類似度を算出し、それらがある一定の類似度以上であれば、そのHAZOP分析結果にあらかじめ付加されている重要度を算術平均することでシーケンス上のイベントに対する重要度を評価している。実際、重要度を付けることによって、ある程度、分析に必要なイベントを特定することができ、安全性分析の工数削減が行える可能性があることは検証できた。しかしながら、実用にあたっては推定精度が十分ではなく、辞書の精緻化などによって類似度を高めることや重要度の評価精度を高める必要があることが分かった。また、過去のHAZOP分析結果とシーケンス図が同一のアプリケーションドメインである必要があるため「同一のドメイン」かどうかを判断するための指標を算出することも必要であることが分かった。

本論文では、HAZOP分析結果を前提として、シーケンス図とHAZOP分析結果の類似度を算出することで重要度を評価したが、今後の可能性としては、過去のシーケンス図上で重要な部

分があらかじめスコアリングされている場合には、対象シーケンスと過去のシーケンスの類似度を算出することで、対象シーケンスの重要度を評価する可能性も考えられる。

ツールそのものについては、プロトタイプであるため、入力及び出力がまだユーザにとって十分使いやすいものにはなっていない。とくに、出力については重要なコンポーネントがシーケンス図上のどこにあるかという情報を視覚的に明示することで、重要部分の特定を支援するための仕組みが必要である。また、入力については、HAZOP分析結果などの文書からの入力が特定のXMLによって与えられており、これを生成するための労力が必要である点が指摘できる。これらの問題については、統合開発環境との融合により解決すべき課題であると考えられる。更に、word2vecによるデータの管理や、イベント、タグの管理も現状はあまりスケーラブルな設計にはなっておらず、多くのデータから分析を行う場合など、より現実的開発状況においては分散処理などを行うことを検討する必要がある。

謝辞

本研究は独立行政法人情報処理推進機構技術本部ソフトウェア高信頼化センター（SEC：Software Reliability Enhancement Center）が実施した「2015年度ソフトウェア工学分野の先導的研究支援事業」の支援のもと行われたものである。

【参考文献】

- [1] T. Mikolov, K. Chen, G. Corrado and J. Dean, Efficient estimation of word representations in vector space, arXiv preprint arXiv: 1301.3781, 2013.
- [2] T. Kudo, K. Yamamoto, Y. Matsumoto, Applying conditional random fields to Japanese morphological analysis, Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing, pp230-237, 2004.
- [3] G. Salton, A. Wong and C.S. Yang, A vector space model for automatic indexing, Communications of the ACM, 18, pp613-620, 1975.
- [4] P.N. Tan, M. Steinbach, V. Kumar, Introduction to Data Mining: Pearson New International Edition, Pearson Education, 2014.
- [5] H. Kashima, K. Tsuda and A. Inokuchi, Marginalized kernels for labeled graphs, Proceedings of the 20th International Conference on Machine Learning, pp321-328, 2003.
- [6] 岡村 寛之, 宮内 聡, 土肥 正, マルコフ決定過程によるWebページランキングアルゴリズム, 電子情報通信学会論文誌 D-I, J89-D(2), pp210-219, 2006.
- [7] M. Puterman, Markov Decision Processes, John Wiley & Sons, New York, 1994.
- [8] S. Brin and L. Page, The anatomy of a large-scale hypertextual Web search engine, Computer Networks and ISDN Systems, 30, pp107-117, 1998.
- [9] L. Page, S. Brin, R. Motwani and T. Winograd, The PageRank citation ranking: bringing order to the Web, Stanford Digital Library Technologies, Working Paper 1999-0120, 1998.

NE比を活用した, つながるシステムにおける利用時の品質向上の提案



神田 周一^{※1, ※4}



諸熊 浩人^{※1}



入江 哲^{※2}



伊藤 潤^{※3}

従来のシステムでは, 利用時の品質が低いために, 利用者が使えなくて困るという問題が発生してきた. この問題は, 多様なステークホルダがかかわる「つながるシステム」において, 更に多発することが心配されるが, 原因はシステムの品質要求に利用時の品質が考慮されていなかったことにある.

本論文では, 品質要求の基準として, 仕様に熟知した開発者 (Expert) が操作する時間と, 利用者 (Novice) が操作する時間との比率であるNE比を用いると, 利用時の品質を考慮できることを説明する. また, つながるシステムにおけるNE比測定の困難さという課題を解決する手法を提案する.

Proposal of NE ratio of quality requirements for improving quality of use in “connected system”

Shuichi Kanda^{※1, ※4}, Hiroto Morokuma^{※1}, Satoshi Irie^{※2}, Jun Ito^{※3}

Conventional systems often cause usability problems due to low quality of use. It is concerned that such problems may occur more often in "connected systems", which various stakeholders are involved with. The reason lies in the fact that quality of use is not taken into consideration when system quality requirements are defined. In this paper, it is explained that the NE ratio, i.e., the ratio of operating times consumed by Expert with knowledge of how to operate and by Novice user of the system, can be applied as a reference value of quality requirements in order to have quality of use taken into consideration. The method that solves the difficulty in measuring NE ratio of a connected system is proposed as well.

1 背景

一般利用者が購入し, 自ら家庭内のネットワークに設置する見守りカメラや環境センサ, スマートキーなど多様ないわゆるIoT機器が登場し注目を集め出している.

このように, つながるシステムを構成するセンサ, デバイス, アプリケーションなどがシステムとしてつなげられたときの標準的な品質指針としてISO/IEC 25000 SQuaRE

(Systems and software Quality Requirements and Evaluation) シリーズが制定されている [1:IPA2015].

1.1 SQuaREシリーズにおける品質の定義

SQuaREシリーズにおける品質モデルを規定したISO/IEC 25010では, システム/ソフトウェア製品品質に加えて, 利用時の品質を定義しているのが特徴である (図1).

利用時の品質とは「特定の利用者が特定の利用状況にお

※1 株式会社U'eyes Design ※2 有限会社 エム・エス・エス ※3 UX測研 ※4 現在, エスディーテック株式会社

いて、有効性、効率性、リスク回避性及び満足性に関して特定の目標を達成するためのニーズを満たすために、製品またはシステムを利用できる度合い」と定義されている。[2: JIS2013]

しかし、多様な利用状況が想定されるシステムを開発する場合、品質目標値や尺度の特定に困難を伴う。このため残念ながら開発時や発注時の品質要求に利用時の品質の観点を含めないことが日常的である。

この結果として、一般利用者にとって使い方が分からないシステムや、そもそも満足できないシステムが世の中に回っているのが現状である。

更に、多様なステークホルダがかかわるつながるシステムにおいては、よりこの傾向が強まり、悪化するものと思われる。

1.2 高まる人間中心設計へのニーズ

他方、認定人間中心設計専門家や認定人間中心設計スペシャリストなど利用時の品質を重視した開発・設計ができる専門家が毎年百名近く増えている[3:HCD-shikaku]。利用時の品質重視の意識が広まりつつあることは間違いない。

しかしながら、つながるシステムの世界では、単体のデバイスだけががんばっても、つながるシステム全体におい

て質の高い利用時の品質を達成することはできない。また、従来は一般利用者と直接の接点がなかったことにより、利用時の品質に対する理解が高くない企業の新規参入も増えていると想定できる。

よって、つながるシステムが適切な利用時の品質を確実に達成できる仕組み作りは重要なテーマである。

1.3 利用時の品質確保における課題

課題は以下2点である。

- 1) 開発時や発注時の品質要求に利用時の品質を含める必要性が認識されていないこと。
- 2) 利用時の品質を品質要求として定義する手法が認識されていないこと。

課題1)について利用時の品質を適切に管理することはこれまで困難であったが、2015年3月に制定された『JIS X 25041 システム及びソフトウェア製品の品質要求及び評価(SQuaRE)ー開発者、取得者及び独立した評価者のための評価手引』[15: JIS2015]に基づき、利用時の品質の報告を発注時に要求することで、高度な専門知識を要せずに、利用時の品質を管理できるようになる。更にシステムやソフトウェア商品の調達時に利用時の品質を調達先に分か

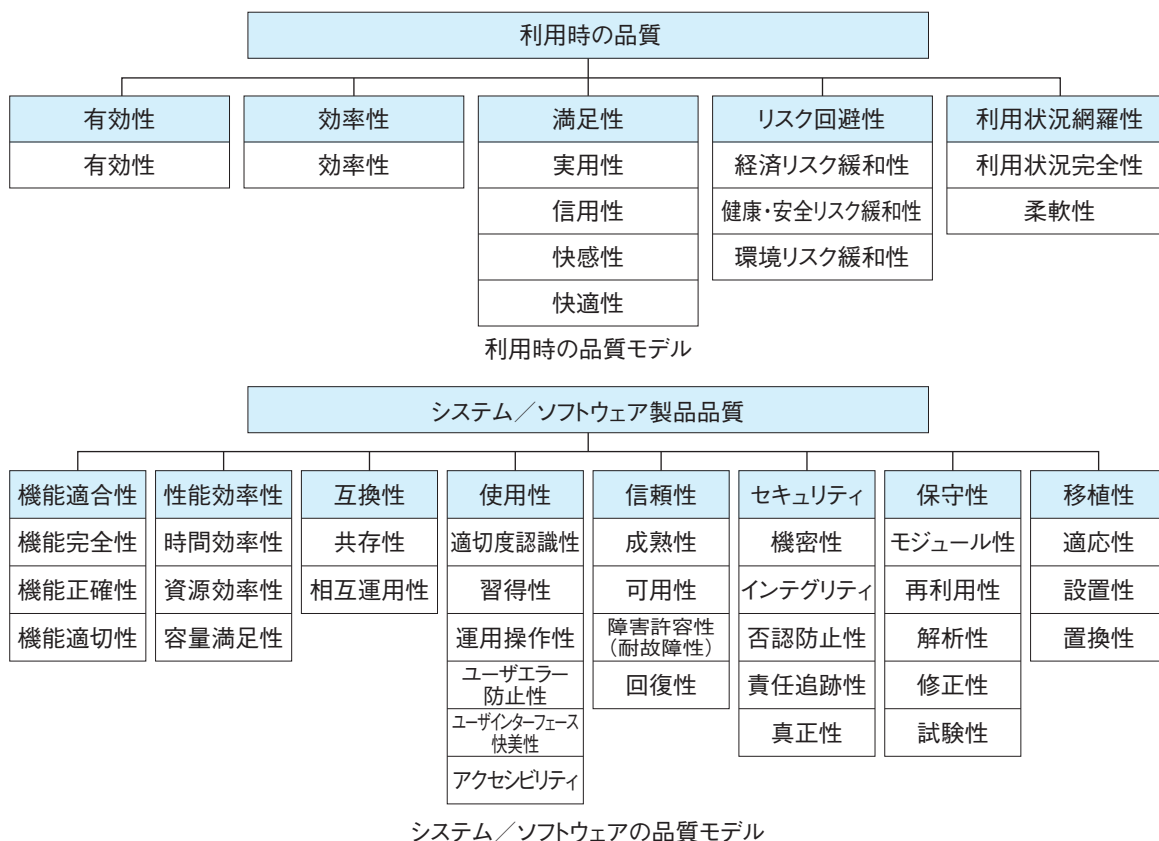


図1 ISO/IEC 25010品質モデル

りやすく報告するための報告書書式がISO/IEC 25062にて規定されており、現在JIS化が進んでいる[4:ISO2006].

今後はつながるシステムの発注者や品質責任者に対してJIS X 25041やISO/IEC 25062の順守を促進することで、利用時の品質要求記述の認識促進が期待できる。

一方、課題2)について、現状では利用時の品質要求を定義する手法は存在するものの、まだあまり認知されていない。そこで本論では、有効な手法を解説すると共に、より「つながるシステム」に適した手法を提案することで、この課題の解決を狙う。

2 利用時の品質の定量化について

2.1 定量化可能な品質特性

利用時の品質は、5つの品質特性(図1)が規定されており、この中で有効性、効率性、満足性については、以下の手法をユーザテストなどに取り入れることで定量化が可能となる。

- 有効性：タスク達成率, エラー回数
- 効率性：タスク完了時間, NEM (2.2節参照)
- 満足性：利用率, System Usability Scale (SUS)

[5:Brooke1986] など

実施例として、複合機 (Multi-Function Printer) の事例がある [6:Nakano2006].

本事例のように、タスク達成率や達成時間の計測を開発目標として定義すれば、競合製品や過去製品との比較が可能になり、開発部署と評価部署が、同じ認識のもと利用時の品質を管理できるようになる。

2.2 Novice Expert ratio Method (NEM) の活用

NEMとは、有効性・効率性に関する利用時の品質の程度を定量的に把握する手法である [7:Urokohara1999]. 目的達成に必要なタスクについて、仕様に熟知した開発者 (Expert) が操作する時間と、特定の利用者 (Novice) が操作する時間との比率 (NE比) をステップごとに算出する手法である。NE比は開発者と一般利用者とのギャップに相当し、NE比の大きさで、利用時の品質が低いことを検出できる (図2).

2.3 NEMの品質目標としての活用

NEMは、問題検知の目安として活用可能で、分かりにくく試行錯誤を要するような、有効性の課題や、慣れないと操作が難しい、といった効率性の課題が検知可能である。また、様々な事例から、NE比が4.5倍以上であれば重篤な使いやすさの課題が存在するという結論が導かれている [7:

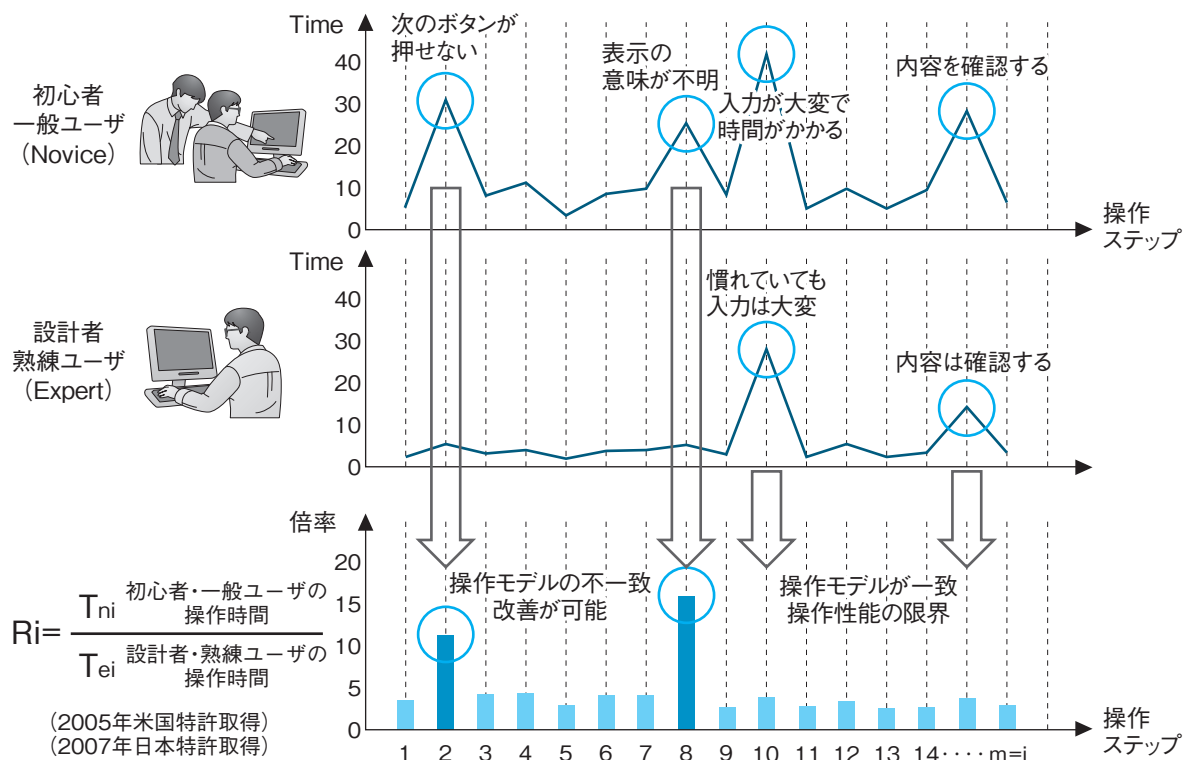


図2 NEMの算出イメージ

Urokohara1999].

「電子政府ガイドライン基本調査」では、6.67という高い数値が示されている。この数値は、実際に重篤な問題が存在することを表している [9:Denshi2009].

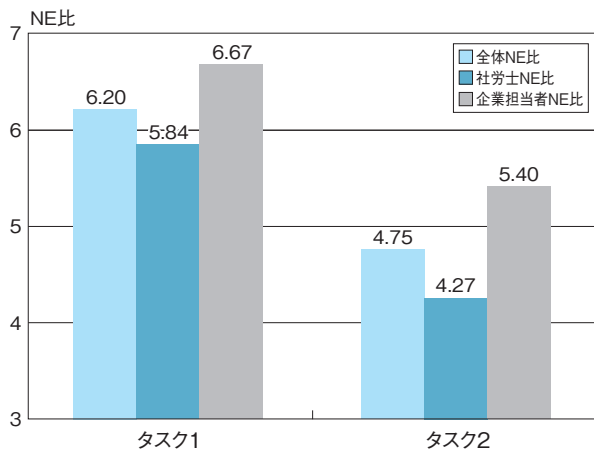


図3 労務担当者向け電子政府サイトにおけるNE比

一方、自動車運転中に操作するエアコンなど多重課題を伴う場合は2以下を目標にするなど利用状況に応じて品質目標を調整可能である [1:IPA2015].

加えて、改善前後のNE比の変化から、改善効果を定量的に把握することや、過去のNE比の知見参照による品質レベルの把握も可能である。このように、NEMは、タスクや機器の違いによらない共通指標として使える可能性もある。

表1 使われない電子申請システム

省庁名	申請件数	利用率 ^{※1}	開発コスト(概算)	年間運用コスト(概算)	対象手続き数
外務省	7件	不明	1億6,800万円	1億1,300万円	73種
環境省	30件	1%未満	単独では不明 ^{※2}	単独では不明 ^{※2}	345種
金融庁	70,000件	不明	15億5,000万円	5,600万円	1,550種
警察庁	13件	1.3%	4,300万円	2,600万円	259種
経済産業省	3,000件	1%未満	34億5,000万円	4,000万円	4,400種
公正取引委員会	3,000件	4%	8,600万円	750万円	23種
厚生労働省	29,000件	不明	11億円	5,000万円	1,800種
国土交通省	7,000件	不明	23億円	2億円	2,500種
財務省	60件	1%未満	7,000万円	6,000万円	300種
総務省	87件	0.03%	公開しない	公開しない	916種
農林水産省	3件	0.002%	110万円	不明 ^{※3}	1,103種
法務省	2,000件	0.0005%	2億円	7,000万円	214種
防衛庁	38件	不明	4,500万円	単独では不明 ^{※4}	36種
文部科学省	8件	0.04% ^{※5}	7億6,000万円	1億2,000万円	1,400種

※1 電子申請件数を全申請件数で割ったもの

※2 認証局システム、行政文書管理システム、許認可進行管理システムなどと合わせて算出しているため。総開発費は7億3,000万円、運用費は9,000万円

※3 省内LANシステムを合わせた契約のため

※4 防衛庁認証局システムと合わせて算出しているため

※5 国が行う手続きのうち、申請届出などの手続きを計上

3 利用時の品質を品質目標にする重要性

3.1 使われない電子申請システム

システムとして完成しながら、著しく利用率が低く問題視された「使われない電子申請システム」の一部を表1に示す [10:Nikkei2005].

「電子政府ガイドライン基本調査」では、次のように考察されている。「オンライン申請・届出システムは、利用者にとってはその利用を強制されるものではなく、書類申請などをはじめとする代替手段の中から「選んで利用する」選択肢の一つである。そのため、オンライン申請・届出システムには、単に「対象手続きが正しく処理できる」信頼性や安全性だけでなく、「利用者に選んで使ってもらえる」利便性や有効性、効率性が求められる。

また、厚生労働省より公表されている、電子申請システムに対する利用者アンケートの集計結果では、電子申請の利用がうまくいかなかった原因に「申請書作成用のソフトウェアを操作すること」、電子申請に対して重視・希望することに「データ作成の画面から直接電子申請の画面に連動していること」などが挙げられている [16:Denshi2008].

このように、利用時の品質の低さが、システムの利用率が低かった原因の一つとなっていることが示されている。

操作者が存在するシステムの開発は、仕様通りに動作す

るだけでは不十分であり、利用時の品質についても適切な品質確保が必要である。

利用時の品質を確保するためには、システム品質の目標に「利用時の品質」を設定すると共に、達成度の確認をすること、すなわち「利用時の品質マネジメント」の導入が有効である。

3.2 利用時の品質マネジメント採用への流れ

動作するが使われないシステムを防ぐため、開発段階から利用時の品質の目標を設定し、運用する事例も報告されている [11:HCD-award2015]。

この事例では、利用者が特定タスク達成に要する時間に着目し、品質目標設定を行っている。トライアル導入の結果、使い方に関する問い合わせ7割減など効果が確認され、全社的に導入された。

4 つながるシステムへ向けた NEM応用の提案

4.1 つながるシステムにおけるNE比測定

2.3節で示したように、NE比は利用時の品質目標として利用可能である。しかし、つながるシステムにおいては、多様なステークホルダがかかわり開発が進むため、すべての仕様を熟知した開発者 (Expert) が存在しないという課題が存在する [1:IPA2015]。このため、測定を担当した人により結果が変わってしまうといった、再現性や信頼性の欠如が発生する。

そこで、Expertに相当する操作時間を、一定のモデルを用いて機械的に推定する手法を提案する。これによって、Expertの招集が不要となり、再現性や信頼性を確保することが可能になる。

4.2 Expert値の推定の仕方

筆者らは、基準となるExpertの操作時間の推定に、利用者の操作時間予測を行う古典的な手法Keystroke Level Model (KLM) [12:Card1980] の考え方を応用することとした。

KLMはインタラクティブシステムにおける個々の操作に要する作業時間Tを以下に示す単位要素に該当する定数の和として算出する考え方である。例えば画面が切り替わった後、特定ターゲットまでマウスでカーソルを移動後クリックしそのターゲットに特定文字をキー入力する操作に要す

る時間は以下の式で算出できる。このときの各定数の説明と推奨されている定数(以降「標準定数」と記す)を紹介する。

$$T = M + P + C + H + K$$

M：精神的な準備時間 1.35sec

P：画面内のポインティングに要する時間 1.1sec

H：キーボードとマウス間で手を移動する時間 0.4sec

C：マウスによるクリックの所要時間 0.2sec

K：キー入力の所要時間 0.2sec

NEMにおけるExpertは、その製品の仕様を熟知した開発者である。このため、迷いのない一貫した操作が可能であり、標準定数の操作時間に近似することが可能と考える。

13thWOCS²において4節の提案を発表した時点では、4.2節に記載した標準定数を用いて、Expert値を推定することを主張した [14:Kanda2016]。

しかし、その後、この方法で算出した操作時間と、実際のExpertが操作した時間(以降「実測値」と記す)とを比較すると、標準定数による推定の精度が不足していることが分かった。そこで、よりExpertに近似する操作時間の得られる定数(以降「提案定数」と記す)を新たに導出した。

4.3 Expert向けのKLM定数を算出した結果

推定対象タスクとして自治体Webサイトのユーザビリティ評価に使われた引越しタスク [13:Hikkoshi2009] を、構造の異なる4つの市を対象とし、Expertと見なせる操作者に操作を依頼した。

Expertと見なせる対象者として、Webサイト設計の経験が豊富な人に、事前に対象サイトの構造を十分に学習させ、迷いのない安定した操作ができるようにした。実際KLM定数の算出に用いた操作において、操作手順に試行錯誤は生じていない。

Expert操作の様子をビデオで記録し、その記録を解析することで単位要素に相当する時間を導出した。例えば、画面が切替わった後目標に向けてマウスを動かし始めるまでの時間をMとし、操作全体の平均値を提案定数とした。

今回導出した提案定数を表2に示す。表2の対象数とはビデオで観測できた単位要素の数であり、観測のバラつき状況を示すために分散も示している。

表2 KLM単位要素の説明と定数

単位要素	単位要素の説明	標準定数	提案定数の導出		
			定数	対象数	分散
M	精神的な準備時間	1.35	0.39	43	0.15
K	キー入力の所要時間	0.20	0.64	32	0.15
P	画面内のポインティングに要する時間	1.10	0.75	72	0.13
H	キーボードとマウス間で手を移動する時間	0.40	0.53	26	0.20

また、今回導出した提案定数を用いて、13thWOCS²発表時[14:Kanda2016]に行った操作を推定した結果を表3に示す。なお、比較のために標準定数による推定結果も記している。

表3 引越タスク全体の操作時間

	Expert 実測値(秒)	標準定数		提案定数	
		推測値(秒)	実測との比	推測値(秒)	実測との比
熊本市	36.43	39.79	1.09	32.75	0.90
神戸市	34.22	42.94	1.25	34.54	1.01
札幌市	27.17	34.60	1.27	26.78	0.99
藤沢市	34.33	43.39	1.26	33.10	0.96

4.4 考察

表3に示したように、本実験で新たに導出した提案定数による推測値は、従来のKLM標準定数による推測値よりも実測値に近い値となっている。このことから、今回導出した提案定数を用いることで、よりExpertに近い操作時間を算出できることが分かった。表2に示されるような、今回の提案定数と標準定数との違いの原因としては、標準定数を算出した当時と、今の操作環境の違いによる影響が考えられる。具体的には、標準定数は、いずれも1980年代に提唱された値であるため、現在の測定環境とはハードウェアの性能や画面解像度などの点で違いが見られる。

加えて、スクロール操作や、タッチ操作などは標準定数として既定されていない。このため、それらの操作を必要とするタスクを対象にNE比を測定するためには、新たな推測ロジックを導出する必要がある。

次節では、マウスホイールを用いてコンテンツを探す場合のスクロール操作を対象として、新しいロジックを導出した事例を示す。

5 マウスホイールによるスクロール操作時間の検討

パソコンにおけるスクロール操作には、主に以下の3通りの方法が存在する。

- ① キーボードによる操作
- ② マウスによるスクロールバー操作
- ③ マウスホイールやトラックパッドによる操作

操作方法①及び②は、既存の標準定数や提案定数を用いてExpert操作時間を推測可能だが、操作方法③については有効な推測方法が定義されていない。

NEMではNoviceとExpertを同一条件で比較することを前提とするため、Noviceが使用する可能性のある操作方法③についても、Expert操作時間を推測できる必要がある。

そこで以下の実験を行い、Expertがマウスホイールを用いてコンテンツを探す過程でのスクロール操作時間の推測ロジックを導出した。

5.1 実験

まず、日常的にマウスホイールを利用する20代~40代の男女30名(著者が所属する会社の従業員)に、一般のWebサイトから所定の目標を見つけるタスクを与え、スクロール操作の速い上位8名を被験者として抽出した。

次に、図4の検証用画面を用いて、所定の目標をマウスホイールによるスクロール操作で探す実験を行った。被験者には、必要なスクロール量の異なる9種類の目標をタスクとしてランダムに提示した。

なお、被験者の抽出を含め、実験には表4に示すような標準的なWindows7ノートパソコン及び、USBレーザーマウスを使用した。

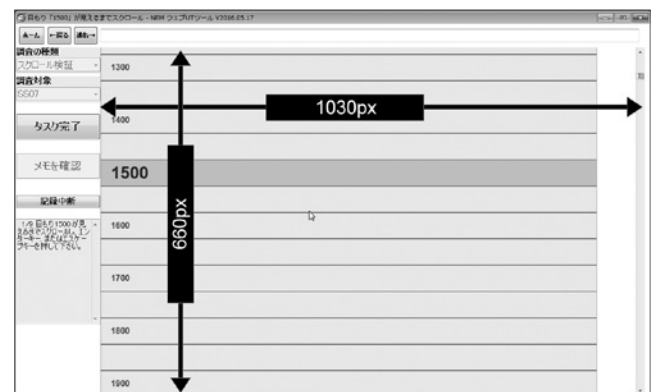


図4 検証用画面

表4 スクロール実験環境

PC	Windowsノートパソコン
機種	Lenovo ThinkPad X220i 4286-3MJ
OS	Windows7 32bit
ディスプレイ	横 1366px, 縦 768px (メーカー推奨解像度)
ブラウザ	Internet Explorer 11 (横 1030px, 縦 660px)
測定精度	平均 20fps (0.05秒間隔)
マウス	HID準拠 USBレーザーマウス
機種	ELECOM M-TG03UL
解像度	1200dpi
重量	85g
ホイール	垂直方向のみ ● 1回転当たり7目盛り ● 1目盛り当たり3行 (99px) スクロール

5.2 実験結果

タスクの遂行に必要なスクロール量に対する平均スクロール速度の関係を図5に示す。

一般的に、現在地から目標地点が遠いことが明らかな場合、行き過ぎることを気にせず速くスクロール操作をすることができる。また、人間による操作であることから、スクロール速度には一定の上限が想定される。

以上のことから、必要なスクロール量 (x) に対する平均スクロール速度 (y) は対数近似が妥当と考えられる。近似式を求めた結果、以下の式が得られた。

$$y = 335.15 \ln(x) - 1076$$

従って、マウスホイールを用いたExpertによるスクロー

ル操作時間 (t) は以下の式で推測可能となる。

$$t = x / (335.15 \ln(x) - 1076)$$

5.3 考察

マウスホイールを用いてコンテンツを探す操作について、現在のExpertが操作した場合の操作時間を提案定数として導出した。

なお、測定に使用するマウスやディスプレイ環境により、結果は変化する可能性はある。このような課題はほかの標準定数、提案定数にも当てはまるため、今後もその時代の標準的な環境での測定を継続することで、測定結果をアップデートすることが望まれる。

6 まとめ

本提案では、まずつながるシステムにおける利用時の品質を定量化する手法としてNE比が活用できることを示した。

更に、NE比算出における「つながるシステムでは全体に熟練した開発者 (Expert) が存在しない」という課題に対して、KLMを応用してExpert操作時間を推定する手法を提案し、現在のシステム利用環境に最適化した新しい定数を導出した。また、標準定数に存在しない操作についても、マウスホイールによるスクロール操作を例に操作時間が推測できることを示した。

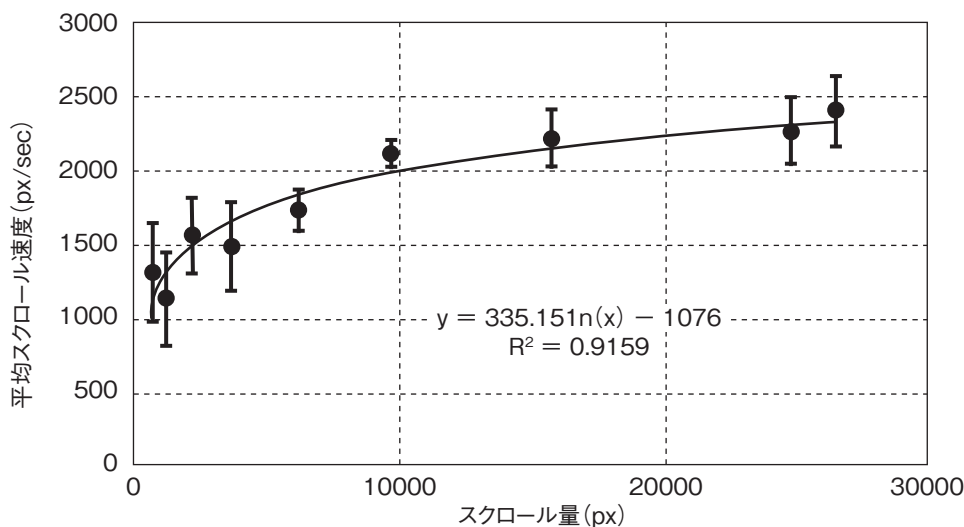


図5 スクロール量に対する平均スクロール速度

システム利用時の品質をNE比によって定量化できるということは、利用時の品質を品質要求の一つとして定義可能であることを示している。

利用時の品質を適切に管理することはこれまで困難であったが、1.3節で記述したようにJIS X 25041やJIS化予定のISO/IEC 25062の適用で今後は高度な専門知識を要さずに、利用時の品質を管理できるようになる。

利用時の品質向上は、利用者が使い方を理解できずに困る、更にはシステムが使われなくなるという事態を減らし、システム本来の利便性を享受できる暮らしやすい世界の実現に寄与するものと考えられる。

7 今後に向けて

KLMモデルによる操作時間の理論値の算出は古くから存在するテーマで、簡単で良い方法である。またKLM手法でExpert操作時間を推定し、容易に利用時の品質基準設定ができる。

ただし、ベースとなる操作モデルに加え、マウスホイール、タッチパネルなど、時代に応じて進化するユーザインタフェースに適合させることが必要であることが分かった。

今回、一例としてマウスホイールを取り上げ、新しいユーザインタフェースにKLMを拡張して適用できることを示した。今後、タッチパネルのフリック操作などへの拡張を行っていくことで、適用範囲を広げていきたい。

新しい操作モデルへの適用にあたっては、産学連携や、様々な業種での取り組みを実施することで、KLMモデルとNE比の利用を促し、標準手法としていきたい。

8 用語・文献

略号一覧

NEM : Novice Expert ratio Method

SQuaRE : Systems and software Quality Requirements and Evaluation

KLM : Keystroke-Level Model

【参考文献】

- [1:IPA2015] IPA/SEC, つながる世界のソフトウェア品質ガイド, P30, May 29, 2015
- [2:JIS2013] 財団法人 日本規格協会, JIS X 25010:2013システム及びソフトウェア製品の品質要求及び評価 (SQuaRE)-システム及びソフトウェア品質モデル, Jun 20, 2013, <<http://www.webstore.jsa.or.jp/webstore/Com/FlowControl.jsp?bunsyold=JIS+X+25010%3A2013&dantaiCd=JIS&status=1&pageNo=0&lang=jp>>
- [3:HCD-shikaku] 特定非営利活動法人 人間中心設計推進機構, 2014年度「人間中心設計専門家・スペシャリスト」資格認定の審査結果報告, May, 2015, <http://www.hcdnet.org/certified/news_certified/2014_1.php>
- [4:ISO2006] ISO/IEC 25062:2006 “Software engineering — Software product Quality Requirements and Evaluation (SQuaRE) — Common Industry Format (CIF) for usability test reports”, Apr 01, 2006, <http://www.iso.org/iso/catalogue_detail.htm?csnumber=43046>
- [5:Brooke1986] J. Brooke, P. W. Jordan & B. Thomas & B. A. Weerdmeester & A. L. McClelland., “SUS - A quick and dirty usability scale”, Usability Evaluation in Industry. London: Taylor and Francis., 1986
- [6:Nakano2006] 中野幹志 & 岩田弘子 & 長崎正道, MFPユーザビリティ品質の定量評価, Human Interface Symposium 2006, Sep 25, 2006
- [7:Urokohara1999] 鱗原晴彦 & 古田一義 & 田中健一 & 黒須正明, 設計者と初心者ユーザの操作時間比較によるユーザビリティ評価手法, Human Interface Symposium 1999, Oct 04, 1999
- [8:Urokohara2015] 鱗原晴彦, UI/UXを測る —いま注目したい品質の視点「利用時の品質」実践事例, 情報処理学会 デジタルプラクティス Vol6 (4), P271-P279, Oct 15, 2015
- [9:Denshi2009] 各府省情報化統括責任者 (CIO) 連絡会議, 電子政府ユーザビリティガイドライン 付属文書, P45, Jul 01, 2009, <http://www.kantei.go.jp/jp/singi/it2/guide/security/kaisai_h21/dai37/h210701gl_f.pdf>
- [10:Nikkei2005] 日経コンピュータ, 普及せぬ電子申請, 平均利用率は1%未満 本誌調査で明らかに, 利用率「不明」のシステムも, 日経コンピュータ2005年11月14日号, P17, Nov 14, 2005, <<http://itpro.nikkeibp.co.jp/article/COLUMN/20051111/224405/>>
- [11:HCD-award2015] 特定非営利活動法人 人間中心設計推進機構, HCDベストプラクティスアワード2015「使いやすさの品質目標値を定義した全社的な品質管理」, Oct 29, 2015, <http://www.hcdnet.org/practice/award/first_awasrd/hcd.html>
- [12:Card1980] Stuart K. Card & Allen Newell & Thomas P. Moran, “The keystroke-level model for user performance time with interactive systems”, Communications of the ACM 23 (7), P396, 1980
- [13:Hikkoshi2009] 特定非営利活動法人 人間中心設計推進機構, 地方自治体Webサイトのユーザビリティ評価 (引越部門) 結果報告書, Apr 23, 2009, <http://www.hcdnet.org/pdf/2008_Web_Evaluation_Report.pdf>
- [14:Kanda2016] 神田周一 & 諸熊浩人 & 入江哲 & 伊藤潤, つながるシステムにおける利用時の品質向上にむけた品質要求事項定量化の提案, 13th WOCS2, Jan 20, 2016 <<https://www.ipa.go.jp/files/000050242.pdf>>
- [15:JIS2015] 財団法人 日本規格協会, JIS X 25041:2015 “システム及びソフトウェア製品の品質要求及び評価 (SQuaRE) — 開発者, 取得者及び独立した評価者のための評価手引”, Mar 20, 2015, <<http://www.webstore.jsa.or.jp/webstore/Com/FlowControl.jsp?lang=jp&bunsyold=JIS+X+25041%3A2015&dantaiCd=JIS&status=1&pageNo=0>>
- [16:Denshi2008] 大臣官房統計情報部 企画課情報企画室, “厚生労働省：電子申請に関するアンケート調査結果 (平成20年度)”, Mar 30, 2008, <<http://www.mhlw.go.jp/sinsei/torikumi/11/>>

オープンシステム・ディペンダビリティのための形式アシュランスケース・フレームワーク (FFO)

奥野 康二^{*1} 木下 修司^{*1} 木下 佳樹^{*1,*2} 武山 誠^{*1,*2} 中原 早生^{*1,*2}

オープンシステムのディペンダビリティを主張する形式アシュランスケースのためのフレームワークを構築し、オープンシステム・ディペンダビリティの要件を再検討すると共に、車載システム及び防災システムにおける事例研究を行った。

Formal assurance case Framework for Open systems dependability (FFO)

Koji Okuno^{*1}, Shuji Kinoshita^{*1}, Yoshiki Kinoshita^{*1,*2}, Makoto Takeyama^{*1,*2}
and Hayao Nakahara^{*1,*2}

A framework for formal assurance cases that claim dependability of open systems is developed. Requirements for open systems dependability are reexamined and case studies in automotive systems and disaster management systems are given.

1 はじめに

本稿では、IPAによる「ソフトウェア工学分野の先導的研究支援事業 (RISE)」の委託研究「オープンシステム・ディペンダビリティのための形式アシュランスケース・フレームワーク」の研究成果^[1]を紹介する。

この研究では、システムがオープンシステム・ディペンダビリティ (Open systems dependability, 以下OSD) を達成していることを議論する形式アシュランスケースを書くためのフレームワーク FFO (Formal assurance case Framework for Open systems dependability) を作成した。また、その作成過程で得られた知見をもとに、システムがオープンシステム・ディペンダビリティを達成するためのシステムライフサイクルプロセスへの要件を考察し、国際標準案 IEC 62853 Open systems dependability^[7] 策定の技術的根拠を与えた。

本論文の構成は以下の通りである。第2節では、アシュランスケースをはじめとする研究背景を述べる。第3節では、フレームワーク FFO 自体の研究成果を述べる。第4節と第5節は事例研究の紹介で、それぞれ FFO を車載システム、防災システムに適用した成果を述べる。第6節で総括する。

2 背景

2.1 アシュランスケース

アシュランスケース (assurance case) は、具体的なシステムの安心・安全に関する議論 (アシュランス議論) の記録文書である。アシュランスケースはシステムの利害関係者間の合意事項の記録、契約文書や認証における提出文書、あるいは事故調査委員会の資料として、近年急速に注目され始めた。

アシュランスケースの利用は、プラントや軍事技術など、高度な安全性が要求される、いわゆる safety critical system に関する安全性議論に用いられることから始まった。現在では車載、鉄道、航空、医療システムの認証に関してアシュランスケースの提出が要請される国際標準があるなど、その需要は増加傾向にある。

膨大なアシュランスケースの構造的な理解を助けるため、図式を用いた構造化アシュランスケースの記法が幾つか提案され、それぞれ普及が図られている (GSN^[9], CAE^[2], D-Case^{[10][11]} など)。アシュランスケースの国際標準 ISO/IEC 15026-2^[17] が発行されている。

我が国でも 2010 年代から D-Case など、アシュランスケース

*1 神奈川大学プログラミング科学研究所 *2 神奈川大学理学部情報科学科

に関連する研究が盛んになり、独立行政法人情報処理推進機構 (IPA) や一般社団法人ディペンダビリティ技術推進協会 (DEOS 協会) において、普及活動が展開されている。

2.2 形式アシュランスケース

形式アシュランスケースとは、整合性検査を自動化するため形式言語によって記述されたアシュランスケースである。筆者らは、関数型プログラミング言語Agdaによる記法、D-Case in Agda^[10](図1)を研究開発している。Agdaは構成的型理論に基づく関数型プログラミング言語であり、高い論理的表現力を持つ。

D-Case in Agdaにより、アシュランスケース記述における整合性の問題が解決された。アシュランスケースは、システムの安心・安全に関する議論の記録文書であるため、対象システムが大きくなり、その議論が重ねられることで、膨大な文書となる。そのような文書の細部にまでわたって整合性を取るのには困難だが、一方で、ほんの少しの不整合も、システムの大きなトラブルに直結し得る。

GSNなどの図式による現行のアシュランスケース記述では、アシュランス議論が何に基づくのか、が明示されないことが問題である。D-Case in Agdaでは、それを形式言語Agdaで定義された概念体系(オントロジー)として明示した上で議論を形式記述することにより、機械的な整合性検査をAgda処理系によ

て行う。これにより、アシュランスケースの本質的な部分の検討に人間の注意を集中することを可能にした。

2.3 OSD

オープンシステムは、システムの境界が定義できず、その機能が時間と共に変化するようなシステムである。オープンシステム・ディペンダビリティ(OSD)とは、オープンシステムが目的、目標、環境及び性能の変化に適応し、説明責任を絶え間なく達成して、想定されるサービスを、要求されたときに要求通り提供する能力のことである^[10]。OSDでは、ある一時点のシステムの機能や状態がディペンダブルであるからと言って、システムがディペンダブルであるとは言えない点に注目する。そのため、システムライフサイクルの観点から長期的にディペンダビリティを達成させなければならない。その際、単にシステムが障害や環境変化に対応できるということだけでなく、説明責任や合意形成などのhuman factorをも考慮することが必要である。

SoS (System of Systems) やIoT (Internet of Things) に関するディペンダビリティ達成のためにはOSD達成が必要である。また、OSDはシステムのレジリエンス(resilience)^[5]とも関連する。防災計画やセキュリティ対策では想定外の災害や攻撃への対処が重要であるが、OSD達成のためには、どこまでを想定するかを明確にすることが求められる。

```
module 安全要求仕様
  (PB : 前提条件[基本機能]-type)
  (PS : 前提条件[安全機能を含む]-type PB)
  (C : 追加の支援情報-type)
  (D : システム設計[基本機能]-type PB C)
  where
  record 安全分析-対象ハザードの発生原因分析[FTA]-項目-type : Set where
    field
      対象安全目標ID : ID-type 安全目標
      安全分析-対象ハザードの発生原因分析[FTA] : FT図-type
  安全分析-対象ハザードの発生原因分析[FTA]-type : Set
  安全分析-対象ハザードの発生原因分析[FTA]-type =
    List (安全分析-対象ハザードの発生原因分析[FTA]-項目-type)
  module 技術安全要求仕様 (SA : 安全分析-対象ハザードの発生原因分析[FTA]-type) where
    record 技術安全要求仕様-項目-type : Set where
      field
        システムブロック名 : システムブロック名-type
        基本事象 : 基本事象-type
        安全要求 : 安全要求-type
        ASIL : ASIL-type
        時間間隔 : Time-type
```

図1 Agdaによる形式アシュランスケースの記述例

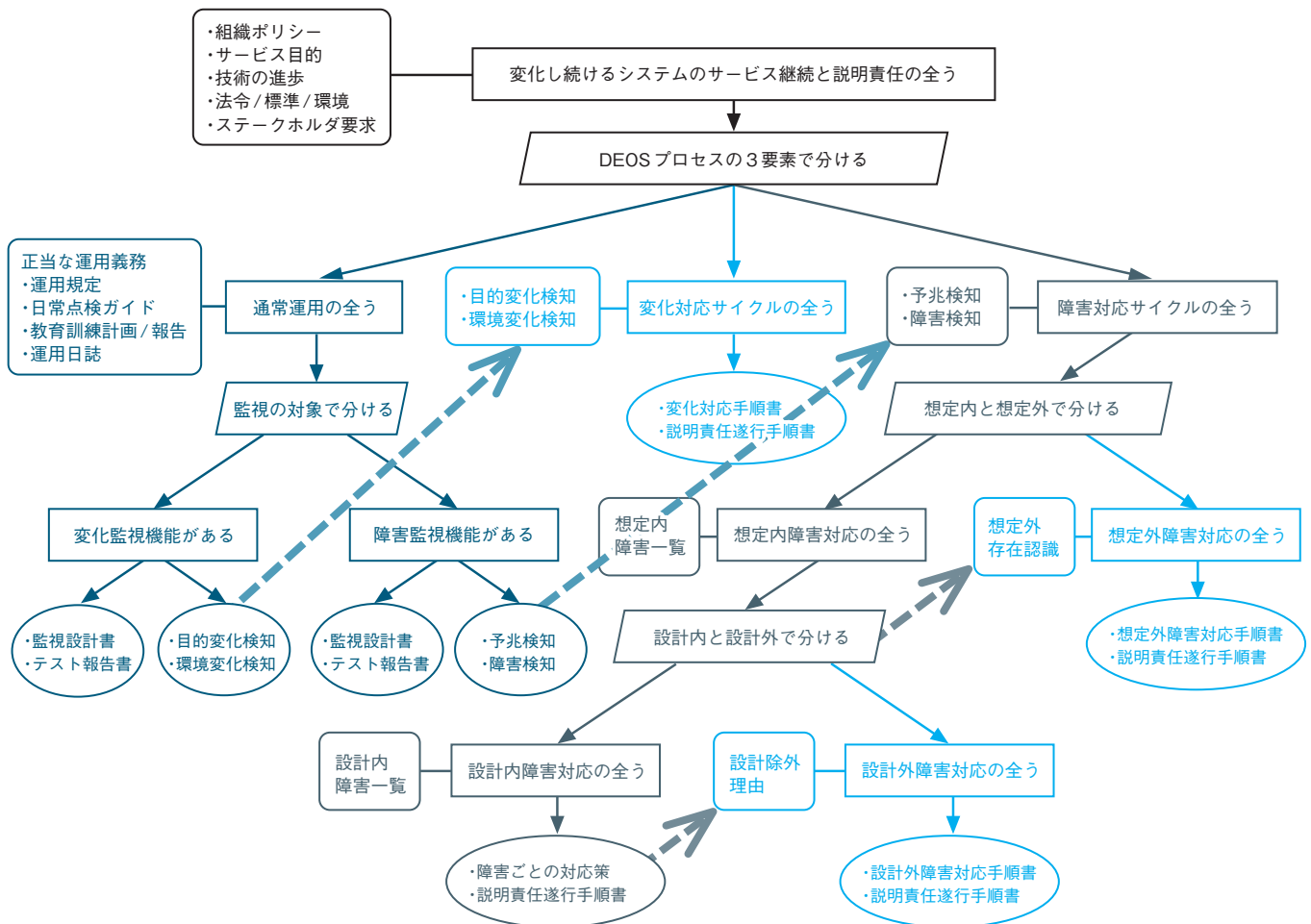


図2 DEOS基本構造のGSNによる表現

3 FFO

3.1 DEOS基本構造

FFOは、OSD達成を議論する形式アシュランスケースのフレームワークである。これは、D-Case in AgdaをOSD達成の議論向けに詳細化したものである。OSD達成に必要な議論のモデルを、先行研究の成果であるDEOS基本構造^[10] Chapter 3. 4. 1に従い作成した。

図2はDEOS基本構造をGSN記法によるアシュランスケースで示したものである。議論の最上位ゴールは、「変化し続けるシステムのサービス継続と説明責任の全う」である。DEOS基本構造に基づき、これを以下の3つのサブゴールに分けた。

- **通常運用**：変化監視と障害監視が正しく機能する
- **変化対応**：システムの目的や環境の変化が検知されたときに正しく対応する
- **障害対応**：システム障害に対して、正しく応じる

また、形式アシュランスケースのAgda言語による実装として、以下のような工夫を考案した。

- 形式アシュランスケースの中で用いる語彙を定義する部分 (Context) と、形式アシュランスケースが提示する主張を議論する部分 (Argument) を、別のモジュールとして明確に分離した。D-Case in Agdaではこのような明確な分離がなされていなかった。
- 証拠の吟味、手順の遂行などのように、形式アシュランスケースの議論部分で繰り返して使われる議論の類型を幾つか見出す、Agdaの関数として実装した。例えば、証拠の吟味は、変化の検知、変化対応への移行、障害検知、障害対応への移行などで議論される。これらの議論では、監査結果、開発証憑、運用証憑の3つが必ず用いられる。この共通部分を1カ所にまとめて機械的に具体化することにより、形式アシュランスケース記述、理解、保守が容易になることが期待される。

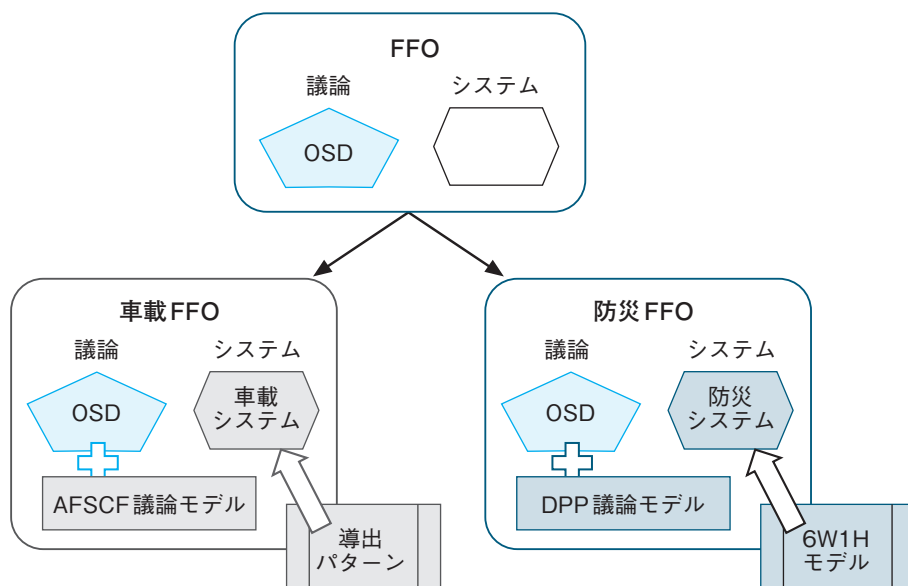


図3 FFOとその事例研究の関係

3.2 FFOを利用したアシュランスケース記述

FFOは、オープンシステム一般に適用可能なフレームワークである。FFOを具体的なシステムの議論に適用するには、以下の手順を踏む。

1. FFOの適用対象となるシステムを明確に定義する(記述する)。
2. FFOが提供するOSD一般の議論モデル(DEOS基本構造)に合わせて、技術領域個別に議論モデルを構築する。
3. 1.2.及びFFOに従ってアシュランスケースを作成する。

1.や2.は必ずしも自明な作業ではない。そこで本研究では、車載システム(第4節)と防災システム(第5節)を対象とした事例研究において、このようなシステム定義の枠組みや議論モデルの構築も研究対象とした。

図3は本研究の各成果の関連を示したものである。まず、FFOはシステムの技術領域を限定しない、オープンシステム一般に適用可能なフレームワークである。また、2つの具体的な技術領域において、FFOに基づくディペンダビリティ議論を考察した。

車載システムのOSD達成に必要な議論を検討し、議論モデルとしてAFSCF議論モデルを考案した。また、システム定義の枠組みとして、導出パターンを考案した。

防災システムのOSD達成に必要な議論を検討し、議論モデルとしてDPP議論モデルを考案した。また、システム定義の枠組みとして、6W1Hモデルを考案した。

4 事例：車載システムへの応用

車載システムの機能安全を規定するISO26262^[3]は、システムライフサイクルのテクニカルプロセスのうち開発設計プロセスに対する要求を主に定める。しかしこれだけでは、機能安全を超えた、開発設計時に作り込むことができないような安全要求を十分に盛り込むことが困難である。

機能安全に加えて、OSDを要求することにより、開発設計時に想定できなかった安全要求を、運用、保守、合意プロセス、説明責任遂行などに対して盛り込むことができる。これらを達成することにより、機能安全の、より現代的な実現が可能になる。そこでFFOを車載システムの機能安全に具体化したAFSCF(Automotive Functional Safety Case Framework)議論モデルを考案した。また、車載システムの定義を明確にするため、導出パターンを考案した。

4.1 導出パターン

導出パターンとは、下位の仕様から上位の仕様を導く導出のパターンである。これを利用することにより、仕様書の段階的な詳細化が容易になり、開発者の負担が軽減される。

システム開発における仕様書は一般に、基本設計書と詳細設計書のように、上位の仕様から下位の仕様へと段階的に詳細化される。その際、それら仕様書間に要請される性質には、「下位の仕様が満たされれば、上位の仕様が満たされる」ことがある。例えば、詳細設計書を記述する場合、その仕様がすべて適切に実装されたときに、基本設計書の仕様が満たされなければいけない。

図4にその例を示す。左側では機能Aが機能A1/A2/A3に適切に分割され、詳細設計書が記述されたのに対して、右側では機能Bを詳細化した際に、本来記述すべき機能B3の詳細設計書が欠落している。そのため、詳細設計書B1/B2の仕様が適切に実装されたとしても、機能Bは満たされない。

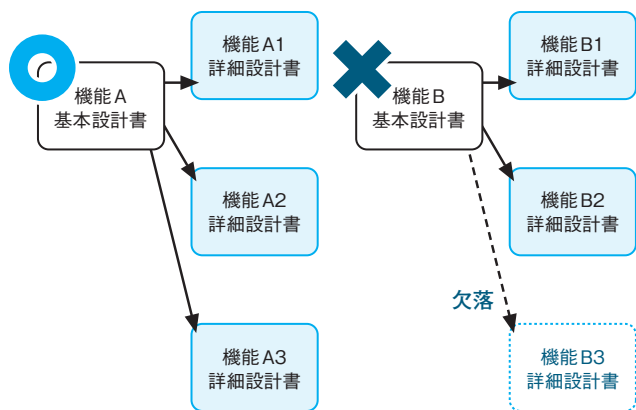


図4 下位仕様と上位仕様との関係

そこで「下位の仕様が満たされれば、上位の仕様が満たされる」ことの議論を明示する。それによって、このような仕様の不備を事前に発見することが可能になる。

これを示したのが図5である。左の例では、機能Aの基本設計書、機能A1、A2、A3の詳細設計書を記述したのちに、「機能A1、A2、A3の詳細設計書の仕様が満たされれば、機能Aの基本設計書の仕様が満たされる」ことが示されている。つまり、欠落がないことの検討の根拠を提供している。一方、右の例では、機能Bの基本設計書、機能B1と機能B2の詳細設計書を記述したのちに、「機能B1とB2の詳細設計書の仕様が満たされれば、機能Bの基本設計書の仕様が満たされる」ことが示されていない。つまり、B1、B2以外にB3の詳細設計書が必要であることが判明する。

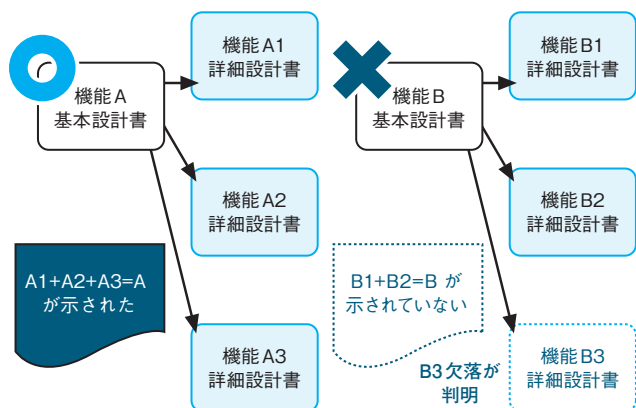


図5 仕様の不備を事前に発見

一般に、上位仕様と下位仕様の対応を示すには、それらの項目間の対応を追跡する必要がある。更に、妥当性の議論のためには、単に対応させるだけでなく、対応することの理由まで明示することが必要である。しかし、この作業は必ずしも自明ではない（どのような記述であれば「対応している」と言えるのかは高度な判断を要する）上に、コスト面の負担が大きい。また、度重なる仕様修正が起こると、そのコストは増大する。

そこで、「下位の仕様が満たされれば、上位の仕様が満たされる」ことが示された仕様書のテンプレートを用意しておく。そのようなテンプレートがあれば、開発者はそのテンプレートに沿って具体的な仕様の埋めていくだけで、適切な上位の仕様書と下位の仕様書が完成する。このように、対応関係だけでなくその妥当性の議論まで記すためのテンプレートが導出パターンである。

4.2 AFSCF議論モデル

車載システム開発がISO 26262に準拠するためには、安全性を示す議論を安全ケース（安全性のアシュランスケース）として提出することが要求される。それらを各社が個別に実施することは容易ではない。そのため、(社)JASPARによって、ISO 26262に準拠した開発を行うための機能安全テンプレート^[4]が提供されている。

そこで、IEC 62853 国際標準案のアシュランスケースモデル^{[7]Annex B}とISO 26262に準拠した安全ケースのレイヤーモデル^[6]を統合した、AFSCF (Automotive Functional Safety Case Framework) 議論モデルを考案し、Agdaで記述した。

ISO 26262は安全ケースの提出を要求しているが、その詳細は規定されていない。AFSCF議論モデルは、安全ケース執筆の指針を提供する。

図6はAFSCF議論モデルをGSNで示したものである。トップゴールは「OSD-機能安全の達成」(システムの正常でない振る舞いによるハザードに起因する不適当なリスクは回避される)である。機能安全の達成は次の6つのサブゴールに分割できると考えられる。6つのうち、1、2は機能安全の観点からの分割であり、3-6はそれらに加え、OSD達成のために必要な要件をDEOS基本構造に従って追加したものである。

1. 「機能安全の達成」(システムの正常でない振る舞いによるハザードに起因する不適当なリスクは回避される)
2. 「環境」(適切な環境のもとで機能安全を達成している)
3. 「合意形成」(システム、システムの目的、目標、環境、性能、ライフサイクル、及びこれらの変化に関する共通理解と明示的合意を確立し、維持する)

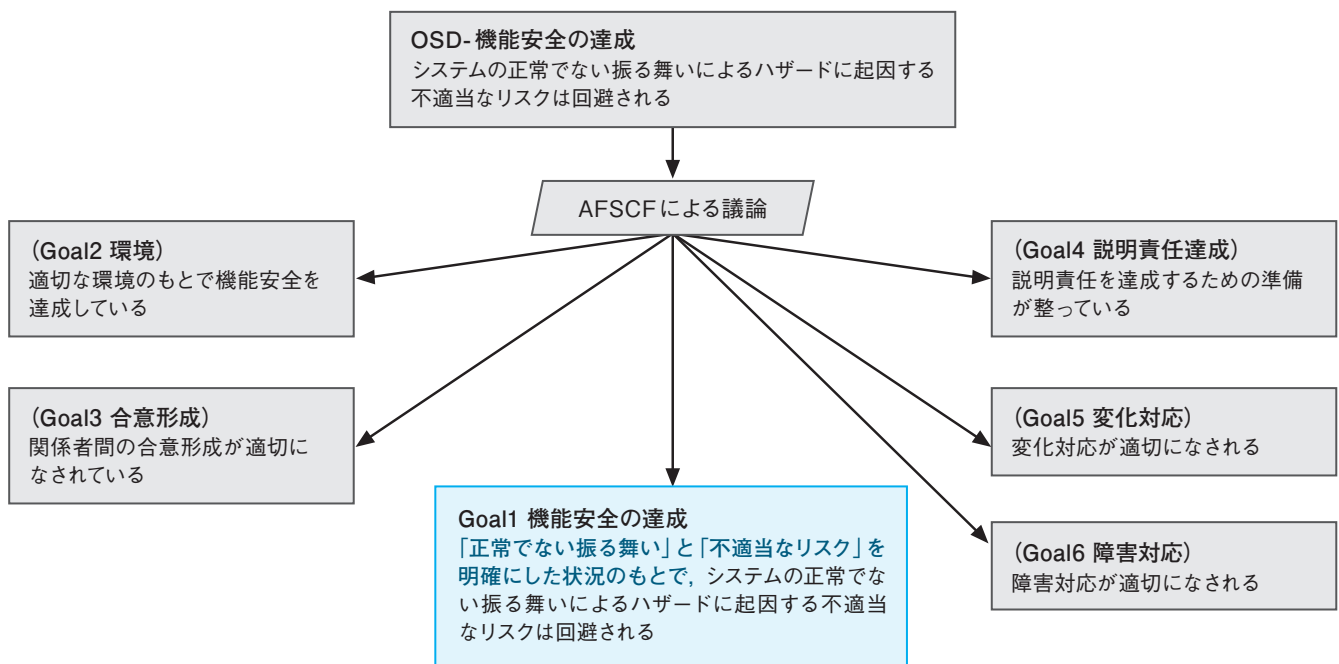


図6 AFSCF議論モデル

4. 「説明責任達成」(システムに関する合意事項の不履行がステークホルダや一般社会に及ぼす影響を同定し、合意事項の遂行を改善して、システムに関する確信と信用を保ち、潜在的な被害に対する補償を確実にする)
5. 「変化対応」(要求、環境、目標及び目的が変化しても、システムの「目的にかなった (fit-for-purpose)」状態を維持する)
6. 「障害対応」(障害に際してもサービス中断と損害を最小にとどめ、その状況のもとで最も適切なやり方で、可能な限りサービス提供を続ける)

更に、このうち1.「機能安全の達成」について、ISO 26262に基づく開発を4つのフェーズに分け、そのフェーズごとに議論することと定めた。[14]にその詳細がある。

5 事例：防災システムへの応用

もう一つのFFOの適用事例として、防災システムを選んだ。ここで言う防災システムとは、地方自治体の防災業務のことである。地方自治体に警察・消防・公共インフラ企業などが協力して実施する防災関係業務は「地域防災計画」という行政文書によって規定される。これは、1961年に施行された災害対策基本法^[13]に基づき、地方自治体で作成する文書である。

平塚市との共同研究により、平塚市地域防災計画^[12]を対象に防災FFOを作成し、防災システムのオープンシステム・ディペンダビリティを主張する形式アシュランスケースの記述を試みた。その際、システム記述の枠組みとして6W1Hモデル^[16]を、議論の枠組みとしてDPP議論モデルを考案した。

5.1 6W1Hモデル

6W1Hモデルとは、6つのW (Who, What, Whom, Where, When, Why)を持つアクションが構成するツリーである。6W1Hの1HとはHowのことである。一般に、「ある1つの作業の記述」は、更に細かい複数の作業の記述によって表現できる。この関係のことをHowと呼ぶ。これが、システムのアシュランス議論に必要なシステムの明確な記述を提供する。

防災業務をシステムライフサイクルとして定式化することにより、様々な局面でやるべき業務を明確にすることができると考えられる。システムライフサイクルについては、国際規格ISO/IEC/IEEE 15288^[8]があるので、これに基づいた防災業務の定義を試みた。

そのためにはまず、システム自体が明確に記述されていないといけない。しかし、それは地域防災計画をもとに自明に構築できるものではないことが明らかになったため、記述の枠組みとして6W1Hモデルを考案した。

図7は、6W1Hモデルを利用したシステムの記述例である。

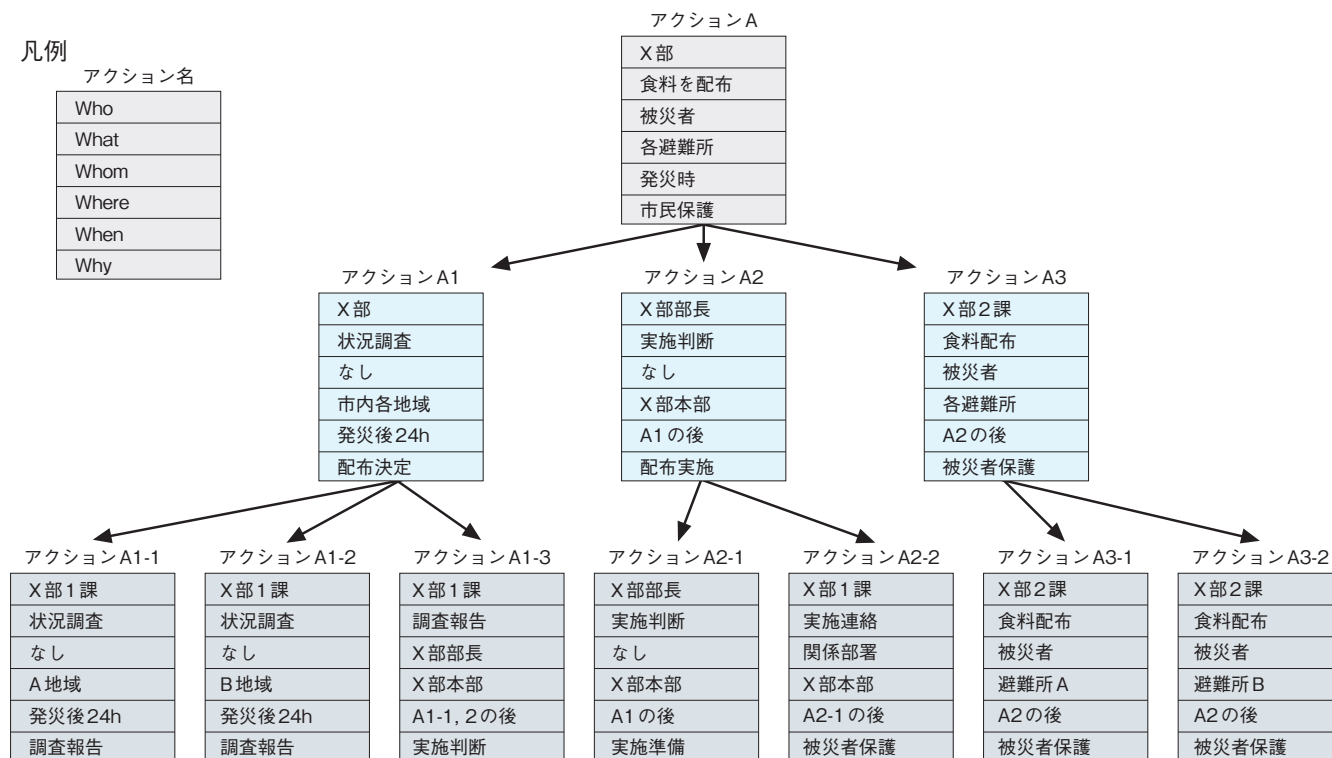


図7 6W1Hモデルの記述例

発災時の食料供給業務を例に取り、Who、What、Whom、Where、When、Whyの6つのラベルを1つのアクションに付け、それを段階的に詳細化した。例えばアクションA「X部は市民保護のため、発災時に、被災者に対して、各避難所で食料を配布する」は、以下の3つのアクションによって詳細化される。

- **A1**：X部は食料配布決定のため、発災後24時間以内に市内各地域の状況を調査する
- **A2**：X部部長は、食料配布実施のため、アクションA1の後にX部本部において実施を判断する
- **A3**：X部2課は、被災者保護のため、アクションA2の後に各避難所において被災者に食料を配布する

これらのアクションは、アシュランス議論に必要な単位まで更に詳細化することができる。地域防災計画に記述が不足しがちな主語(Who)が明示されるとして、平塚市の防災担当者から肯定的評価を得た。発災時の給水業務を事例とした記述実験では、6W1Hモデルの49個のアクションを構成した。そのうち、主語を明確にしなければならなかったアクションは19個にのぼった。主語を明確にすることにより、地域防災計画をシステムとして定義可能にし、それについてのアシュランス議論が可能になった。

5.2 DPP議論モデル

防災システムのオープンシステム・ディペンダビリティに関する議論モデルの一例がDPP議論モデルである。「DPP」とは、Decision(決定)、Preparation(準備)、Provision(実施)の頭文字を取ったものである。多くの業務は以下の3つに分類されるという考えに基づく。

- **決定 (Decision)** ある業務を実施するために必要な情報収集と開始の判断及び、実施した業務の確認と継続・終了の判断
- **準備 (Preparation)** ある業務を実施するために必要な人的、物的リソースの準備、実施計画の作成
- **実施 (Provision)** 準備された業務の実施

図8はDPP議論モデルを、発災時の給水業務に対して適用した例である。このような決定、準備、実施による議論の分割は、給水だけでなく、物資配給全般や応援要員派遣などの様々な防災業務に適用可能である。

6W1Hモデル及びDPP議論モデルに基づき記述した給水業務のアシュランスケースは、DEOS基本構造に基づく防災業務全体のアシュランスケースの一部として位置付けられる。すなわち、発災時の給水業務とは、DEOS基本構造における障害対応のうち、設計内障害対応の一つに当たる^[1]。

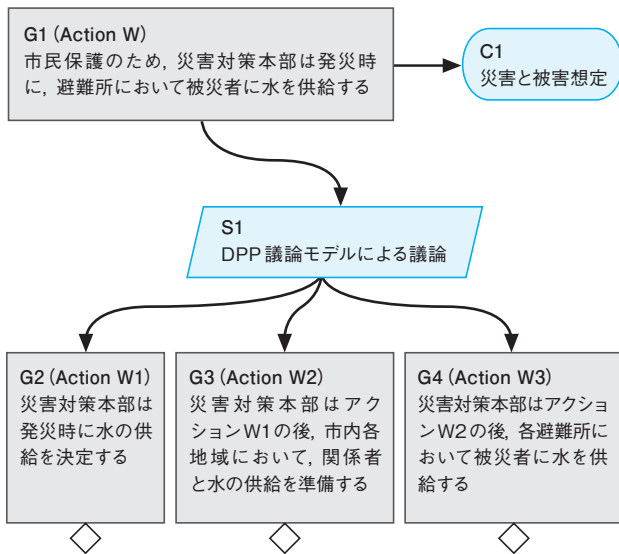


図8 DPP議論モデルの給水業務への適用

6 考察

本研究は、以下の2つの内容をアシュランスケース開発フレームワークとして明示した。

- 形式アシュランスケースの様式とその処理技術
- オープンシステム・ディペンダビリティの概念体系

フレームワークの試作により、オープンシステム・ディペンダビリティの要件について、より洗練された理解を得ることが

できた。それらは、策定中の国際標準案^[7]に反映されている。車載システム及び防災システムにおける事例研究によって更にこの理解を深めることができた。

しかし、ソフトウェア開発現場へのFFO導入手法、ほかの技術領域への適用手法などの構築は、今後の課題である。本研究においては、車載システムにおいてAFSCF議論モデルを、防災システムにおいてDPP議論モデルを構築した。現場の技術者によって、これらのモデルを具体的なシステムに適用し、その工数を計測するなど、更なる実証評価が必要である。

導出パターンが用意する仕様書のテンプレートが適切であるとはどういうことかの明示も、今後の課題として残されている。

本研究の後、プロセスを組み合わせたライフサイクルモデルへの要件を規定するなどの後継活動の必要がある。このような要件は、社会における一定のコンセンサスを得て初めて利用可能なものとなっていく。そのためにもオープンシステム・ディペンダビリティ要件の国際標準化は重要であると考えられる。

謝辞

本研究は、IPA RISE事業の一テーマとして行われた。AFSCF議論モデルと導出パターンの研究にあたって(株)デンソーのご協力をいただいた。また、6W1HモデルとDPP議論モデルの研究にあたって平塚市防災危機管理部災害対策課のご協力をいただいた。ここに記して深甚なる感謝の意を表する。

【参考文献】

- [1] 2014年度ソフトウェア工学分野の先導的研究支援事業「オープンシステム・ディペンダビリティのための形式アシュランスケース・フレームワーク」成果報告書, 神奈川大学. URL: <http://www.ipa.go.jp/sec/raise/#03-1>, 2017-08-05閲覧.
- [2] Adelard, Safety Case Structuring: Claims Arguments and Evidence. URL: <https://www.adelard.com/asce/choosing-asce/cae.html>, 2017-08-05閲覧.
- [3] ISO 26262 Road Vehicles — Functional Safety. ISO Standard (2011)
- [4] (社)JASPAR, 機能安全テンプレート, 2013. URL: <https://www.jaspar.jp/disclosureDocument/#anchor-8>, 2017-08-05閲覧.
- [5] J.-C. Laprie, From Dependability to Resilience, The 38th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), 2008.
- [6] I. Habli, et. al., A Layered Model for Structuring Automotive Safety Arguments, 10th European Dependable Computing Conference (EDCC 2014), Newcastle upon Tyne, UK, May 2014.
- [7] IEC 62853 Open systems dependability (unpublished Committee Draft for Vote).
- [8] ISO/IEC/IEEE 15288:2015 Systems and software engineering — System life cycle processes
- [9] Origin Consulting Limited, GSN Community Standard, Version 1, York, 2011.
- [10] M. Tokoro (ed.), Open systems dependability 2nd edition, CRC Press, 2015.
- [11] 所眞理雄編著, DEOS — 変化し続けるシステムのためのディペンダビリティ工学, 近代科学社, 2014.
- [12] 平塚市地域防災計画, 平塚市防災会議, 2015. URL: http://www.city.hiratsuka.kanagawa.jp/bosai/page-c_01661.html, 2017-07-20閲覧.
- [13] 災害対策基本法. URL: <http://law.e-gov.go.jp/htmlldata/S36/S36H0223.html>, 2017-08-05閲覧.
- [14] 中原早生, 木下佳樹, 自動車機能安全・OSDアシュランスケースの為のAFSCF議論モデル, Science Journal of Kanagawa University Vol.27, 2016.
- [15] 木下佳樹, DEOS関連国際標準の動向, 第2回DEOS協会オープンシンポジウム, 2015. URL: http://deos.or.jp/link/obj/pdf/hyoujunka_bukai.pdf, 2017-08-05閲覧.
- [16] Shuji Kinoshita and Yoshiki Kinoshita, The 6W1H Model as a Basis for Systems Assurance Argument, 4th International Workshop on Assurance Cases for Software-intensive Systems, LNCS, vol. 9923, pp.63-74, Springer, 2016.
- [17] ISO/IEC 15026-2:2011 Systems and software engineering — Systems and software assurance — Part 2: Assurance case

情報システムの事故データ

情報システムの障害状況 2017年前半データ

IPA顧問 松田 晃一
SEC研究員 目黒 達生

2017年1月から6月までの間に、情報システムの障害は23件報道されており、相変わらず障害の発生は高い水準にある。その中で、システム障害がセキュリティ問題を引き起こした事例が2件発生している。また、業務処理の誤りが見逃されたまま運用されていたが、その誤りが偶然発覚した事例が4件報告されている。

1. はじめに

2017年1月から6月までの6カ月間に報道された情報システムの障害23件の概況を次節で述べる。3節では、システム障害が原因で個人情報が漏えいするセキュリティ問題を引き起こした事例を紹介する。更に4節では、業務プログラムの誤りに気がつかないまま運用されていたシステムにおいて、偶然その誤りが発覚した事例について述べる。

2. 2017年前半の概況

2017年の前半に報道された障害23件(表1)のうち4件(事例1703、1704、1705、1706)は、いずれも2016年から誤りが発生していたにもかかわらずそれに気づかず、2017年になって偶然発覚し、報道されたものである。これまでの例に倣って報道された年の障害としてカウントすると、今期の発生件数は月平均3.8件となり、かなり高い水準である(図1)。仮にこの4件を差し引いたとしても月平均件数が3.2件となり高い水準であることには変わらない。なお、この4件については第4節で具体的に取り上げる。

また、システム障害が原因で個人情報が流出する事態を招いた事例が今期2件報道されている(事例1716、別表4)。システム障害がセキュリティ問題を引き起こした事例として次節で取り上げる。

更に、本連載でも度々取り上げてきた[松田 2017]システムへのアクセスの集中をきっかけとする障害が今期も2

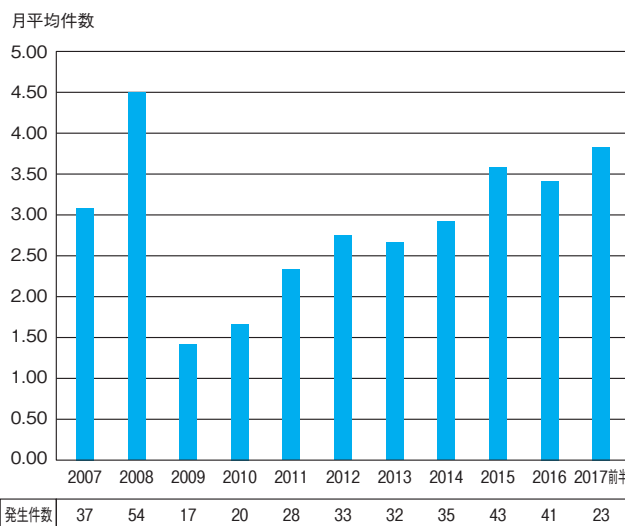


図1 報道された情報システムの障害件数の推移

件(事例1707、1710)報告されている。

2015年に運用開始後、障害が多数発生し問題となっていたマイナンバー関連システムは、今期2件の障害が報道されている(事例1709、1717)。前者はマイナンバーカードを使ってコンビニで住民票などを交付するサービスにおいて、ネットワーク障害が発生した事例であり、後者はマイナンバーカードの交付に使うシステムの端末ソフトウェアの更新に伴う障害である。なお、このほかに自治体の行政サービスに使われているシステムの障害が5件報道されている。これら障害の影響はそれぞれの自治体に閉じており、全国的に大きな影響を与えたわけではないが、各自治体での安定的な運用への注意を促すために別表に別枠として取り上げた。

表1 2017年前半の情報システム障害データ(報道に基づきSECが整理)

No.	システム名	発生日時(上段) 回復日時(下段)				影響	現象と原因	直接原因	情報源
		年	月	日	時				
1701	りそなHD ATM	2017	1	10	8時45分	ATM利用手数料の誤徴収。過大徴収は、約1万9,000件、計205万円。過小徴収は、約3万9,000件、420万円。	10日8時45分から12時59分までに、りそなHD系銀行、コンビニ大手などのATMで、りそな以外のキャッシュカード使用者に、本来108円の手数料を、誤って216円徴収。原因は設定ミス。	設定ミス	<ul style="list-style-type: none"> 朝日新聞朝刊(2017.1.12) 日本経済新聞朝刊(2017.1.12) りそなホールディングスニュースリリース(2017.1.11)
		2017	1	10	12時59分				
1702	Z会運用システム	2017	1	11		通信教育講座の一部申し込み不可、教材の印刷や製本が不可など発生。また、最大約10万人に教材を発送できなくなる可能性。	新システムへの移行作業を進めていたところ、障害が発生。受け付けを3月20日に再開。	システム移行による障害	<ul style="list-style-type: none"> Z会プレスリリース(2017.1.30) Z会お客様へのご案内HP 朝日新聞朝刊(2017.1.31) 日本経済新聞朝刊(2017.1.31)
		2017	3	20					
1703	北海道電力託送業務システム	2017	1	12		インバランス料金の不具合のため、発電・小売電気事業者などと一般送配電事業者との間の取引に影響が生じた。	電力需要の計画と実績の過不足量(インバランス)を算定する際、本来計算に加える必要のある値が一部欠落。原因は送料金制度の変更における情報収集不足と、算定プログラムの作成に際して、仕様確認が不十分だったこと。2017年3月末までにプログラムの修正を行う。	プログラムの不具合	<ul style="list-style-type: none"> 日本経済新聞朝刊(2017.1.19) 北海道電力プレスリリース(2017.1.18) ※障害発生は2016年4月であるが、それが判明した日時に基づき掲載。
1704	中部電力料金請求システム	2017	1	15		<ul style="list-style-type: none"> 振込用紙の重複送付[約7,500件] 請求書記載の電気使用量等の表示誤り[約1,000件] 口座再振替のお知らせ時の金額誤り[約3,000件] 請求書等発行遅延[約11万件] 高圧受電(6,000V)のお客さまの電気料金を請求書を届けられないまま、口座から引き落してしまった。 	1月4日～6日に検針したスマートメータ設置顧客に、振込用紙を重複送付。4日～6日に検針した複数契約顧客に、請求書誤記載。12月分の残高不足顧客で、複数契約で次回振替日が1月11日～13日の顧客に、金額誤通知。電気料金請求書等の発送、最大3営業日遅れ。高圧受電(6,000V)の顧客に請求書を届けず、いきなり口座引き落としを実施。原因/対策は、①開発時の仕様漏れ、設計漏れ、テスト項目漏れ、検出漏れ→組織間の責任、役割分担の明確化。体制、マネジメントの強化。②運用に伴う誤認、認識相違→事業者と委託会社の役割の明確化と情報共有。	プログラムの不具合 運用ミス	<ul style="list-style-type: none"> 朝日新聞電子版(2017.1.15) 日本経済新聞朝刊(2017.1.16) 中部電力プレスリリース(2017.1.15、1.19、1.21、1.27) ※障害発生は2016年12月であるが、それが判明した日時に基づき掲載。
1705	日本臓器移植ネットワーク患者検索システム	2017	1	27		移植患者を選ぶ新しい検索システムに不具合があり、2016年10月のシステム導入後にあった脳死臓器提供20例のうち、3例の心臓移植で選定ミスがあった。提供を受けるはずだった2人が移植を受けられず、1,000日以上待機となった。	病院から指摘があり、患者の治療状況の情報修正時、待機日数が誤って長く計算されるプログラムミスが発覚。対策は、①CIOとPMOを設置し、情報システムの計画、保守などを行う。②熟知したコーディネーターを配置する。③新システムは、旧システムとの比較検証を行った後、コーディネーターによる確認後再稼働する。④課題の共有や安全管理室の機能を強化する。	プログラムの不具合	<ul style="list-style-type: none"> 朝日新聞朝刊(2017.1.28、3.30) 読売新聞朝刊(2017.1.28) 日本経済新聞朝刊(2017.1.28、3.30) 日本臓器移植ネットワーク第三者調査チーム報告書(2017.3.29) ※障害発生は2016年10月であるが、それが判明した日時に基づき掲載。
1706	東京電力パワーグリッド託送業務システム	2017	1	27		顧客の引越しや契約変更の情報を、契約切替えシステムに反映するのが遅れた。原因は、①4営業日以内に通知すべき需要データ(新規検針分)の未通知の発生、②電気使用量の誤通知による未確定電気使用量の発生。対策は、①体制・役割の見直し、②開発プロセスにおけるチェック強化、③適切な業務設計とリスク想定・リスク対策、④トラブル発生時の基本対応の確立、⑤経営層をはじめとする全職員の意識改革を実施。	顧客の引越しや契約変更の情報を、契約切替えシステムに反映するのが遅れた。原因は、①4営業日以内に通知すべき需要データ(新規検針分)の未通知の発生、②電気使用量の誤通知による未確定電気使用量の発生。対策は、①体制・役割の見直し、②開発プロセスにおけるチェック強化、③適切な業務設計とリスク想定・リスク対策、④トラブル発生時の基本対応の確立、⑤経営層をはじめとする全職員の意識改革を実施。	ソフトウェア障害	<ul style="list-style-type: none"> 日本経済新聞朝刊(2017.1.28) 日経コンピュータ(2017.6.19) 東京電力パワーグリッドプレスリリース(2017.1.27、6.9) ※障害発生は2016年4月であるが、それが判明した日時に基づき掲載。

No.	システム名	発生日時(上段) 回復日時(下段)				影響	現象と原因	直接原因	情報源
		年	月	日	時				
1707	地方税電子化 協議会 電子申告・納税 システム(eLTAX (エルタックス))	2017	1	27		インターネットを利用した地方 税の電子申告で、当初はつな がりにくい状態が発生したが、 企業などが送信したはずの申 告データが自治体に届かない 事態が発生した。	原因は、①予想以上のアクセ スが集中し、システムの負荷上 限超過、②一部の通信機器の 再起動繰り返しによるレスポ ンス遅延、③この期間中、受付が 未完了なのに「送信完了」と表 示され、正常終了でないにも かかわらず、手続きを終える ケースが多発。 対策は、①負荷上限を拡大、② 通信機器のソフトウェア修正、 ③利用者への注意喚起。	アクセス 集中 プログラ ムの不 具合	<ul style="list-style-type: none"> 東京新聞朝刊(2017.3.30) 地方税電子化協議会からの お知らせ(2017.2.10)
		2017	2	1	午後				
1708	JR九州 運行管理 システム	2017	1	27	14時00分	鹿児島線で27日14時頃、最 大1時間にわたって全線が運 転を見合わせ、その間56本が 運休した。全線で40分間から 1時間運転を見合わせて4県 で約2万5,000人に影響 した。	博多総合指令で、列車の運行 状況を表示する管理システム の画面が突然動かなくなった。 原因は、外部業者が指令所内 の電源装置のバッテリーの交 換をしていた際、予備電源に切 替える際に不具合が発生し、運 行管理システムに電気が一時 的に供給されなくなった。	作業ミス	<ul style="list-style-type: none"> 毎日新聞地方版朝刊 (2017.1.28) 朝日新聞地方版朝刊 (2017.1.28) 日本経済新聞地方版朝刊 (2017.1.28) 読売新聞地方版朝刊 (2017.1.28)
		2017	1	27	15時00分				
1709	J-LIS マイナンバー コンビニ交付	2017	2	13	4時11分	ネットワーク(LGWAN)の障 害により、コンビニでの住民票 などの交付サービスを実施し ている全国の団体(360団体) のうち19市町で、13日朝から 169件の交付に支障が発生 した。	コンビニ交付で使用している ネットワークの全国センターの ルータで障害が発生。8時13 分に再起動を実施した後に、 ネットワークが不安定になった。 11時に接続エラーが解消され、 復旧した。原因は調査中。	ネットワー ク障 害	<ul style="list-style-type: none"> 産経新聞朝刊(2017.2.14) 地方公共団体情報システム 機構(J-LIS)報道発表 (2017.2.13)
		2017	2	13	11時00分				
1710	DAZN 動画配信 サービス	2017	2	26	16時30分	Jリーグ(日本プロサッカーリ グ)の試合をインターネット配 信する「DAZN(ダ・ゾーン)」で、 26日のライブ動画が見られな いトラブルが起きた。Jリーグ の公式戦2試合のライブ中継 が視聴できなくなり、見逃し映 像の配信もストップした。	26日16時頃に終了したJリ グ7試合を配信直後、アクセス が集中。原因は、ログが急増し、 リソース不足により処理が滞留 したことによる。作業ログ消去 機能や、データ蓄積領域の上 限監視機能を備えていたが不 具合があり、作動せず、更に2 系統の映像処理分散を行って いたが、1系統だけで処理す るミスがあった。対策として、 ソフトウェアの修正、バックアップ を2系統から3系統に増やす など実施。	アクセス 集中 ソフトウェ ア障 害	<ul style="list-style-type: none"> 朝日新聞朝刊 (2017.2.27, 28, 3.3) 日本経済新聞朝刊 (2017.2.27, 28, 3.3) 日経コンピュータ (2017.3.30)
1711	警視庁 免許管理 システム	2017	3	15	11時05分	運転免許試験場などで、シス テム障害により、約1時間にわ たり免許更新などができず、府 中、鮫洲、江東の各試験場のほ か、12の警察署と、神田と新 宿の運転免許更新センターを 訪れた人約210人に影響。	運転免許の更新や住所変更を 担うシステムで障害が発生。 12日施行の改正道路交通法 に対応するため、プログラムを 修正し12日から運用していた が、15日の免許更新業務で、 プログラムの不具合が判明 した。	プログラ ムの不 具合	<ul style="list-style-type: none"> 朝日新聞東京版朝刊 (2017.3.16) 東京新聞朝刊(2017.3.16)
		2017	3	15	11時55分				
1712	大分県警 免許証システム	2017	3	26	10時00分	大分県運転免許センター(大 分市)のシステム障害により、 免許更新に訪れた557人のう ち287人に新しい免許証を発 行できなかった。来訪者に対 する講習や、顔写真の撮影は 実施できた。	26日10時頃、大分県警の運 転免許発行や更新などの業務 を担う県運転免許センターの システムに障害が発生。当初 は通常通りに発行できていた が、作業途中から新たな免許 証が出てこなくなった。原因は 調査中。	不明	<ul style="list-style-type: none"> 産経新聞ネット(2017.3.26) 日本経済新聞地方版朝刊 (2017.3.27)
1713	りそな銀行 振込システム	2017	3	24	未明	一時、約4万6,000件の振り 込みができなくなった。完了で きなかった振り込みについては、 順次手続きを続け、27日朝ま でにほぼ解消した。	24日未明、りそな銀行の現金 自動出入機(ATM)やネットバ ンキングで他行の口座に金を 振り込むシステムに障害が発 生。8時24分過ぎに復旧。他 行と接続する複数の回線のう ち一つで障害が発生した。原 因は調査中。	不明	<ul style="list-style-type: none"> 朝日新聞デジタル (2017.3.24) 読売新聞朝刊(2017.3.25) ニッキン(2017.3.31)
		2017	3	27	朝				

No.	システム名	発生日時(上段) 回復日時(下段)				影響	現象と原因	直接原因	情報源
		年	月	日	時				
1714	コンテナ 車両待機場 入退場システム	2017	3	28	20時00分	入場待ちの車両が4~5kmも延び、更に周知不足で、道路が混雑。5つある入場レーンのうち、1つしか開けられず。誘導指示がドライバーへのメール通知であるといった危険性も指摘された。	28日夜から、東京港大井コンテナふ頭待機場が入退場システムを導入したが、直後にシステム障害が発生。専用の「ドライバーズカード」をかざして整理券を受け取る発券機が作動せず。	不明	<ul style="list-style-type: none"> カーゴニュース(2017.4.4) 物流ニッポン(2017.4.4)
1715	日本カード ネットワーク	2017	4	15	11時08分	JCBや三菱UFJニコスなど複数のクレジットカード会社で、15日11時頃から、カードの決済ができなくなった。一方、JR東日本が運営するインターネット予約システム「えきねっと」でも同日、クレジットカードを使った乗車券の新規予約や変更、払い戻しなどのサービスが利用できなくなった。	15日11時8分から、日本カードネットワークが運営する「CARDNETセンター」でカードの決済ができなくなるトラブルが発生。複数ある同センターの1拠点で、二重化してあるL3スイッチの一方が故障。もう一方のスイッチで処理を続行したが、1系統にトラフィックが集中したことで輻輳が起きた。故障のスイッチを交換しシステムを再起動。17時18分復旧。	ハードウェア 障害	<ul style="list-style-type: none"> 読売オンライン(2017.4.15) 産経ニュース(2017.4.15) 日本カードネットワークニュースリリース(2017.4.15) 日経コンピュータ(2017.4.17)
		2017	4	15	17時18分				
1716	日本気象 予報士会 個人情報	2017	4	23		日本気象予報士会に所属する気象予報士のうち389人の氏名や連絡先といった個人情報が、インターネット上で閲覧できる状態になっていた。約5年間で500件超の情報が流出した。	会員が「名刺発注サービス」を利用する際、氏名やメールアドレス、電話番号などが、一部の検索サイトからアクセスできていた。4月23日に発覚。原因は、システムの一部にパスワードを設定していなかったため。パスワードをかけ、対応。	作業ミス	<ul style="list-style-type: none"> 日本経済新聞電子版(2017.5.30) 産経ニュース(2017.5.30) 日本経済新聞朝刊(2017.5.31)
1717	地方公共団体 情報システム 機構(J-LIS) マイナンバー カード発行 システム	2017	4	28		4月28日、自治体がマイナンバーカードの交付に使うシステムのソフトウェアをバージョンアップした端末で不具合が発生した。バージョンアップを適用していない端末は正常に稼働していたため、未適用の端末を確保して業務を継続した自治体もあった。	カード管理業務で利用する端末の機能向上のため、人口規模に応じて自治体に複数設置される「市町村統合端末」と、自治体ごとに設置する「住民ネットCS(住民基本台帳ネットワークシステム コミュニケーション・サーバー)」について、28日早朝からソフトをバージョンアップしていた。ところが、バージョンアップすると「カード発行状況照会」が利用できなくなった。	設定ミス	<ul style="list-style-type: none"> 日経コンピュータ(2017.4.28)
1718	日本テレビHD 動画配信Hulu	2017	5	17		一部の機種で、「動画が表示されない」「再生中に止まる」「画質が荒くなる」「アプリが動かない」「字幕が表示されない」といった視聴できない事象が発生。	動画配信サービス「Hulu」が、17日にリニューアルしたところ、一部のテレビ受信機やモバイル端末などで動画再生できないトラブルが発生。クラウドサービス障害、コンテンツ保護機能の設定の不具合、プログラムの仕様のミスなど複数の問題が生じた。機種ごとにプログラムの修正作業を進めており、徐々に復旧中。	システム 移行による 障害	<ul style="list-style-type: none"> 日本経済新聞電子版(2017.5.23) 日経産業新聞(2017.5.24)
1719	新生銀行 送金システム	2017	5	25	9時00分	約3万7,000件の送金取引が遅延。	他行から自前の個人口座宛の送金取引を処理するシステムで、送金情報のあるサーバーから別のサーバーへ伝達する際、データの漏れがあり、エラーが発生。エラーを引き起こした送金取引を特定し、データを再送したところ解消。	ソフトウェア 障害	<ul style="list-style-type: none"> 日本経済新聞電子版(2017.5.25) 新生銀行ホームページお知らせ(2017.5.25)
		2017	5	25	11時55分				
1720	気象庁 羽田空港 気象観測 システム	2017	5	26	11時05分	羽田空港の気象観測システムの故障により、同空港を出発する航空機に遅延が生じた。全日空は、10便で離着陸が30分以上遅れた。	26日11時5分頃から約1時間にわたって、羽田空港の気象観測システム(同空港で観測した風速や気温、気圧などのデータを管制官や航空会社に自動配信)のサーバーに障害が発生。同庁職員がデータを口頭で読み上げて対応した。	ハードウェア 障害	<ul style="list-style-type: none"> 読売オンライン(2017.5.26)
		2017	5	26	12時05分				

No.	システム名	発生日時(上段) 回復日時(下段)				影響	現象と原因	直接原因	情報源
		年	月	日	時				
1721	JR東日本 メカトロニクス、 三菱UFJニコス クラウド決済 システム	2017	5	29		イオングループの「ミニストップ」 と「まいばすけっと」の店舗で、 電子マネーの決済ができない などの状況が発生した。	5月29日に、クラウド決済シ ステム「J-Mups」が、使えなくな る障害が発生。6月5日までに 解消した。原因は、5月29日 にJ-Mupsの運営会社を実施 したシステム更改による。 「J-Mups」とは、インターネッ トからクレジットカード決済、各 種交通系電子マネーなどの利 用まで、1台の端末で取り扱い 可能にしたシステム。	ソフトウェア 障害	• 日経コンピュータ (2017.6.28)
		2017	6	5					
1722	ヤフー ネット オークション	2017	6	19		出品者は、落札者情報が確認 できない、商品が出品できな いなどの不具合、落札者は、 決済サービスが利用できない などの不具合が発生。	システム障害は19日に発生し た。一部で、商品の出品や発 送などができない事態が2日 間続いた。障害は、21日18 時頃に復旧予定。原因は調 査中。	不明	• 日経コンピュータ (2017.6.21) • ヤフーお知らせHP (2017.6.19)
		2017	6	21	18時00分				
1723	日本郵便 e発送サービス	2017	6	28	16時00分	ローソンの店舗端末「Loppi」 で送り状の印刷ができなくなり、 荷物の発送ができない状態が 発生。影響は、約1万2,300 店舗。	28日16時頃、Loppiと日本郵 便のシステム連携に問題が発 生。システム連携を休止し復 旧を実施。原因は不明。e発送 サービスは、フリーマーケット やネットオークションなど個人 間取引での配送サービス。	不明	• 日経コンピュータ (2017.6.30) • 日本郵便HPお知らせ (2017.6.29)

別表 2017年前半の情報システム障害データ(報道に基づきSECが整理)

No.	システム名	発生日時(上段) 回復日時(下段)				影響	現象と原因	直接原因	情報源
		年	月	日	時				
1	柳井市 基幹システム	2017	2	20	11時30分	住民票の発行に使う基幹系ネッ トワークに障害が起き、証明書 類が一時的に発行できなくな るなどの影響が55件発生した。	20日11時頃から基幹系ネッ トワークの接続端末でエラーが 発生。翌21日6時30分に仮 復旧。原因は庁舎内の一部接 続器の故障。取り換え作業を 実施中。	ハードウェア 障害	• 朝日新聞地方版 (2017.2.22) • 毎日新聞地方版 (2017.2.22)
		2017	2	21	6時30分				
2	真庭市 業務システム	2017	2	27	8時00分	住民票や印鑑証明、所得証明 が発行できなくなるなど、519 件の処理ができなくなった。	27日8時頃業務システムを起 動したが、正常に動かず。応急 処置を行い、17時30分に仮 復旧。原因は庁内情報ネッ トワークの接続障害とみられ、調 査中。	ネットワーク 障害	• 朝日新聞地方版(2017.3.1) • 山陽新聞さんデジ (2017.2.27)
		2017	2	27	17時30分				
3	大阪市 住基システム	2017	4	2	9時00分	5区役所で11人が住民票の 発行や転出入の届け出、印鑑 登録ができなかった。	住民基本台帳システムのサー バーが業務開始から10分間、 全24区役所の端末と接続で きない障害発生。	不明	• 産経新聞ネット(2017.4.2)
		2017	4	2	9時10分				
4	堺市 メール誤送信 防止システム	2017	4	10	8時45分	219件のメールアドレスが流 出した。4通の電子メールが本 来「BCC」に記入して送信され るところ、「TO」に記入され、受 信者にすべてのメールアドレス が表示される形で送信された。	10日8時45分から運用を開 始した電子メールで「BCC」と 記入されたメールアドレスを 誤って「TO」に自動的変換して いた。誤りに気づき、10時59 分に変換機能を停止した。	設定ミス	• 朝日新聞(2017.4.12) • 堺市報道発表(2017.4.11)
		2017	4	10	10時59分				
5	大阪市 ホームページ	2017	5	15	10時00分	平成29年5月15日10時14 分に大阪市ホームページが閲 覧できなくなる障害が発生 した。	15日10時14分、大阪市ホー ムページに関する警報を検出。 ホームページの閲覧及びホー ムページ管理システム(CMS) による編集ができない状態が 判明。16時40分に仮復旧を 行い、ホームページを公開した。 原因は、業務委託事業者の設 計不備と対応誤りによるネッ トワークサービスの機能停止。	設計ミス 作業ミス	• 日本経済新聞夕刊 (2017.5.15) • 大阪市報道発表 (2017.5.17)
		2017	5	15	16時40分				

3. システム障害に起因するセキュリティ問題

セキュリティ問題は、システム障害と同様に利用者の安全・安心を脅かす重大な脅威である点は共通であるが、それを引き起す要因は異なり、前者は悪意を持った攻撃者による意図的なものであるのに対し、後者はシステムに潜在的に存在していた欠陥が、意図しない何らかの理由によって顕在化する偶発的なものである。本連載ではセキュリティ問題は対象外とし、専らシステム障害に限ってレポートをしてきた。この方針は今後も維持するが、今期の障害の中で、システムの障害によって、個人情報にかかわるセキュリティ問題が発生した事例が2件発生しているため本節で紹介する。

まず、別表に示す事例4は市役所のメールシステムに誤送信防止システムを導入したところ、設定の誤りによって200件余りのメールアドレスが流出した事例である。TO及びCCに設定されたメールアドレスをBCCに自動的に移し、他人のメールアドレスを受信者から見られないように保護するためのシステムであったが、設定の誤りによって逆にBCCのアドレスをTOに転写する結果となりすべての受信者のメールアドレスが読み取れるようになってしまった。個人情報を保護するための措置が逆効果になるという皮肉な結果になってしまった。

また、事例1716は日本気象予報士会に所属する気象予報士のうち400名弱の氏名や連絡先などの個人情報がインターネット上で閲覧可能となっていた事例である。会員がオンラインで名刺発注ができるサービスで、外注業者へ提供する情報のデータファイルにパスワードなどによる保護がなかったためとのことである。しかも、この状態に気づかず約5年間にわたってインターネット上で閲覧可能な状態にあった。

以上が今期に発覚した2件の事例であるが、同様の事例をさかのぼって調べてみると2011年以降6件の事例を見ることができる(事例1127、1216、1329、1423、1530、1621)。事例1127では携帯メールシステムの障害によって返信用メールアドレスに別人のアドレスが誤って設定され、意図しないメールが他人に送られた[松田2012]。事例1216は携帯電話サービスにおいてソフトウェア保守のミスによって、顧客の各種設定情報が他人によって参照・更新できる状態に置かれた[大高2013]。事例1329、1530は誤った信用情報が登録され流通した事故である[松田1 2014]、[松田1 2016]。事例1423では、顧客管理システムの更改をしたところ、新しいソフトウェアに欠陥があり、顧客の個人情報が数日間閲覧可能な状態に置かれた[松

田2 2014]。事例1621では、オンラインショッピングシステムにおいて店舗ごとの日次売上速報メールが、ソフトウェアの不備により他店に誤送信され機微な情報が漏れた。更に、誤送信メールでは送り先のメールアドレスがすべてTO欄に羅列されており、受信者から見られる状態になっていた[松田2017]。

このように、システム障害による個人情報の漏えいなどのセキュリティ事故が過去にも発生している。また、漏えいには至らないがそれに至る可能性のある事故も発生している(事例1533、1534)[松田1 2016]。セキュリティ事故は、システム外部の悪意ある攻撃者によって引き起こされるだけでなく、これら事例に見られるように、意図しないシステムの障害によっても発生する可能性があることに留意が必要である。

4. 業務処理の誤りの長期間見逃し

事例1705は、日本臓器移植ネットワークの患者選定システムに不具合があり、心臓移植を待つ患者の選定に誤りが発生した障害事例である。しかも、このシステムを導入した2016年からこの不具合を抱えたまま運用されており、この間に優先順位下位の患者が誤って選ばれ、本来対象ではない患者に移植が行われるという重大な患者選定ミスが3件発生していた。本件については、2017年3月に第三者調査チームによる原因検証結果と再発防止策が報告書としてまとめられ公表されているので[報告書2017]、詳細はそちらに譲るが主要な点を以下に示す。

本件の問題発覚の契機となったのは、2017年1月26日の心臓移植患者候補の選定であった。システムによって第1候補者と第2候補者が選定され移植実施施設に通知された。偶然、両候補者とも同じ施設を希望していたため、施設の担当者が順位の逆転に気づいた。もし、候補の患者がそれぞれ異なる移植施設を希望していたら問題の発覚は更に遅れた可能性が高い。連絡を受けて人手でチェックしたところ、患者の待機日数の計算に誤りが発見され、患者選定順位に誤りがあることが確認された。待機日数を計算するプログラムのコーディングに際し、変数名の取り違えという初歩的なミスがあり、このミスが後続のテスト工程でも発見されずそのまま運用に至った。プログラムミスが起こった機能は、当初の要件定義や設計段階では考慮されておらず、詳細設計工程を過ぎた後にユーザから追加・変更が求められた。このため、当初のテストシナリオには存在せず、機能の追加後もテストシナリオの追加がされなかったため、テスト工程でも発見されなかった。また、続く受

け入れ試験においても一連の業務フローを広くカバーするテストが行われず、更に旧システムとの並行稼働においてシステム出力の比較確認が行われたが、期間が1カ月と短くその間に今回の誤りに該当するケースが発生せず、誤りが検出されなかった。

要件の追加・変更が頻繁に行われることになったのは、システム利用者／発注者側での業務の整理ができていなかったことが主な要因である。このため当初の要件定義が実際の業務とはかけ離れた形で終わり、後続の工程での要件の修正が多数発生した。そのための業務内容の確認・調整に時間がかかり、工程が遅延しテスト工程の時間不足となった。また、同様に利用者側での業務の整理ができていなかったため、テストケースを網羅的に設計できず、受け入れ試験でケース漏れが生じた。

このように報告書では、システム利用者／発注者側での業務の整理の不十分さが、要件定義など上流工程の品質低下や手戻り、テストの不備となり今回の事故につながった大きな背景要因として指摘している。このことは本件に限った特別な事情ではなく、多くのシステム開発に共通した問題である。IPA/SECが公開している「ユーザのための要件定義ガイド」では、これらの点についての勘どころを記載しており参考にさせていただきたい[SEC1 2017]。

このほかに、業務処理の誤りが長期間見逃された同様の事例が3件(事例1703、1704、1706)報道されている。い

ずれも電力会社の料金計算にかかわるシステムの問題であり、電力小売自由化やスマートメータ導入など、新しい制度の導入を契機に発生した。紙面の関係で詳細は別の機会に譲ることにしたいが、この種の問題はこれまでも繰り返し発生し、本連載でも何度か取り上げているが[松田2 2016など]、注意が必要である。

5. むすび

2017年前半の情報システムの障害について、報道などをもとに整理し報告した。IPA/SECでは、失敗の経験を社会の共通の財産として共有し、少しでも事故を防ぎ、安心・安全なIT社会を目標に、これらの障害事例を分析し、参考にすべき教訓をくみ取る活動を進めている。そして、教訓がまとまるごとに逐次Webサイトで公開しているので参照していただきたい。

URL :<http://www.ipa.go.jp/sec/system/lesson.html>

また、教訓集活用メールマガジンの配信も行っているので、興味のある方は上記IPA/SECのWebサイト「情報処理システム高信頼化教訓のリンク集」のページからメール配信の登録をしていただきたい。更に、教訓をまとめた教訓集が公開されているので併せて参考にさせていただきたい[SEC2 2017]。

【参考文献】

[松田2012] 松田晃一・金沢成恭：情報システムの障害状況 2011年後半データ、SEC journal No.28、Vol. 8、No1、pp.6-8、Mar.2012

[大高2013] 大高 浩・松田晃一：情報システムの障害状況 2012年後半データ、SEC journal No.32、Vol. 9、No1、pp.37-41、Mar.2013

[松田1 2014] 松田晃一・目黒達生他：情報システムの障害状況 2013年後半データ、SEC journal No.36、Vol. 10、No1、pp.32-35、Mar.2014

[松田2 2014] 松田晃一・八嶋俊介他：情報システムの障害状況 2014年前半データ、SEC journal No.38、Vol. 10、No3、pp.42-47、Sep.2014

[松田1 2016] 松田晃一・八嶋俊介：情報システムの障害状況 2015年後半データ、SEC journal No.44、Vol. 11、No4、pp.48-53、Mar.2016

[松田2 2016] 松田晃一・八嶋俊介：情報システムの障害状況 2016年前半データ、SEC journal No.46、Vol. 12、No2、pp.43-49、Sep.2016

[松田 2017] 松田晃一・八嶋俊介：情報システムの障害状況 2016年後半データ、SEC journal No.48、Vol. 12、No4、pp.62-67、Mar.2017

[SEC1 2017] 情報処理推進機構 SEC：ユーザのための要件定義ガイド、2017年3月

[SEC2 2017] 情報処理推進機構 SEC：情報処理システム高信頼化教訓集(2016年度版)(ITサービス編)、2017年3月

[報告書2017] あっせん誤りに関する第三者調査チーム：報告書～心臓あっせん誤りの原因検証結果と、国民からの信頼を回復するための再発防止策の提言～、http://www.jotrnw.or.jp/file_lib/pc/etc/2017-0329-1.pdf、2017年3月29日



神谷 芳樹 著

ISBN : 978-4908520082
サイバー出版センター刊
A5判・302頁
定価3,200円(税抜)
2017年4月14日刊
(Kindle版は2016年7月)

コンピュータを「電気・ガス・水道」の ように身近にしたイノベーション

コンドラチェフの次の大波に乗れ

SEC journalは本号で50号を迎えたが、ここで紹介する書籍は、電気情報通信の50年を鳥瞰した『コンピュータを「電気・ガス・水道」のように身近にしたイノベーション』である。私を感じた本書の特徴は二つあり、一つ目が著者が実際に身を置いたプロジェクトをトピックとして取り上げていること、もう一つは産業史を景気循環の波と併せて大きな視野から捉えていることである。

筆者が取り上げているトピックとしては、国家間のメインフレーム開発競争からソフトウェアビジネスまで、日本のソフトウェア産業史のダイナミズムを語る上で抜けのないもので、それが実体験なのだからすごい経験の持ち主である。それらトピックは経験談だけでなく、そのプロジェクトの意義が解説され、その後の動きも含めて考察されているため、歴史本として懐かしむにとどまらず示唆を与えてくれる。

また、景気循環との組み合わせはコンドラチェフの波などを用いている。コンドラチェフの波(コンドラチェフ循環)は、約50年の周期を持つ景気循環経験則で、技術革新に起因すると考えられている。筆者は正に産業革命後の第4番目のサイクルをエキサイティングな産業活動の中で経験し、この経験を次の波の立ち上がりを担う若い人たちに伝えたいとしている。その想いは、サブタイトルの「コンドラチェフの次の大波に乗れ」にも表れている。私はその若い人たちの分類には当てはまらないが、ソフトウェア産業、情報サービス産業に携わってきた私たちにとっては、産業界を振り返る書として手元に置きたい一冊となるであろう。

(遠藤 秀則)



独立行政法人
情報処理推進機構
AI白書編集委員会 編

ISBN : 978-4-04-899607-5
株式会社角川アスキー総合
研究所刊
A4判・360頁
定価3,300円(税抜)
2017年7月刊

AI白書2017

～人工知能がもたらす技術の革新と社会の変貌～

IPAは、ディープラーニングの進展などにより急速に進化を続けるAI(人工知能)について、正しい技術の理解と利用促進を促すため、現時点でのAIを巡る動きを総合し、分かりやすく解説した「AI白書2017」を刊行した。

ディープラーニングを起爆剤として、AIはその技術や利用が急速に進展しつつあり、産業や社会の様々な場面に大きな影響を与えることが予想されている。そこで、情報処理技術の利活用を促進するIPAでは、正しい技術の理解と利用を促すため、AIにおける産学の第一線の有識者7名で構成される編集委員会や、多くの執筆委員の協力を得て、「AI白書2017」を7月20日に刊行した。本書では、AIの技術動向、利用動向にとどまらずAI関連の制度的課題とその対応動向、AI研究開発の発展と産業への適用を促進する国内外の政策動向など、AIに関連した多様な動向を総合的に解説している。また、企業経営における第一人者による寄稿や情報系教育機関におけるAI分野の教育動向調査なども掲載している。

本書がAIへの正しい理解と適切な利活用の進展、及び日本発のイノベーション、新たなビジネス創出などによる、産業の活性化に寄与することを期待している。

(遠藤 秀則)

編集後記

創刊50号記念号をお届けいたします。創刊号の発行は、SEC設立3カ月後の2005年1月、干支が一周してしまいましたが、年4回のペースでの刊行を続け本誌で50号となりました。今回、SEC及びSEC journalを様々な形で支えてくださった方々より寄稿いただき本記念号を作成しています。また、50号の歩みも振り返ってみました。興味を引くもの、懐かしいものがございましたら、IPA/SECのWebサイトからすべて元記事が参照できますので、ぜひご覧いただければと思います。

最後に、50号を迎え、SEC journalを支えてくださった読者の方々、論文投稿者、執筆者の方々、編集に携わった方々すべてにあらためてお礼を申し上げます。お役に立つ誌面作りに取り組んでまいりますので、引き続きよろしくお願いたします。

(編集長)

編集部より

次世代のソフトウェア・エンジニアリングに関してなど、忌憚のないご意見をお待ちしております。下記のFAX またはメールにてお気軽にお寄せください。

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

SEC journal 編集委員会

編集委員長	遠藤 秀則
編集委員 (50音順)	荒川 明夫 石橋 正行 江野村 亮輔 日下 保裕 佐藤 康彦 中尾 昌善 長谷川 佳奈子 三原 幸博 室 修治 山下 博之 和田 恭



京都・南禅寺水路閣

撮影：Y.Kusaka

SEC journal 第13巻 第2号(通巻53号) 2017年9月1日発行

©独立行政法人情報処理推進機構 2017

編集兼発行人 独立行政法人情報処理推進機構
技術本部 ソフトウェア高信頼化センター
所長 松本 隆明
〒113-6591 東京都文京区本駒込2-28-8 文京グリーンコート センターオフィス16階
Tel : 03-5978-7543 Fax : 03-5978-7517
URL : <http://www.ipa.go.jp/sec/> e-mail : sec-journal_customer@ipa.go.jp

※本誌は「著作権法」によって、著作権等の権利が保護されている著作物です。

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

SEC journal 論文募集

独立行政法人情報処理推進機構（IPA） 技術本部 ソフトウェア高信頼化センターでは、下記の内容で論文を募集しています。

論文テーマ

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

論文分野

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

応募要項

締切り：1月・4月・7月・11月 各月末日

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

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

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

SEC journal 論文賞

毎年「採録」された論文を対象に審査し、優秀論文にはSECjournal論文賞として最優秀賞、優秀賞、所長賞を副賞と併せて贈呈します。

IoT時代に活躍する【組み込みシステムの腕利きエンジニア】を目指す！

国家試験 エンベデッドシステムスペシャリスト試験

高度な実践能力の証明に！

- ▶ 身近な場面を想定した出題を通して、最適な組み込みシステム実現のために必要となる高度な実践能力（レベル4）を問います。

レベル4の定義：専門分野において、自らのスキルの活用によって、独力で業務上の課題の発見と解決をリードするレベル。

技術要素

プロセッサ、メモリ、バス、計測・制御、リアルタイムOS、プラットフォーム、電気・電子回路、ネットワーク、セキュリティ

開発技術

- ・要求分析の実行とレビュー
- ・設計の実行とレビュー
- ・テストの実行とレビュー

管理技術

- ・開発環境マネジメント
- ・知財マネジメント
- ・構成管理、変更管理

- ▶ 近年の試験では、「無線通信ネットワークを使用した安全運転支援システム」、「3次元複写機」、「通信機能をもつ電子血圧計を用いた健康管理システム」、「非接触型ICカードを使用した入退場ゲートシステム」などのテーマを出題しました。
- ▶ 自動車、家電、モバイル機器などに搭載する組み込みシステムや重要インフラの制御システムを、ハードウェアとソフトウェアを適切に組み合わせて構築し、求められる機能・性能・品質・セキュリティなどを実現できる組み込みエンジニアを目指す方に最適です。

試験概要

【試験区分】 エンベデッドシステムスペシャリスト試験（情報処理技術者試験 高度試験の1区分として実施）

【日 時】 年1回の実施（毎年4月第3日曜日）

【申込受付】 毎年1月中旬から2月下旬（予定）までWEB・郵送で申込み受付

詳しくは、Webページをご覧ください。<http://www.jitec.ipa.go.jp/index.html>

試験概要の最新情報、過去問題、活用事例などをご紹介します。

IPA Better Life with IT

SEC journal No.50
第13巻第2号(通巻53号)
2017年9月1日発行

©独立行政法人情報処理推進機構

ISSN 1349-8622

