

SWEBOK V3.0 日本語訳版^{※1}の 連続紹介 —3の1

SC7/WG20^{※2} エキスパート、新谷 IT コンサルティング
新谷 勝利

1. はじめに

SWEBOK は、Software Engineering Body of Knowledge の省略形で、それが広範に用いられることを目的とし、そのオリジナルの英語版はすべて無料で脚注に示す URL からダウンロード可能になっている。2001 年に SWEBOK Trial Version^{※3}、2004 年に SWEBOK 2004^{※4}、2013 年に SWEBOK V3.0^{※5} がそれぞれ発行されている。

SWEBOK V3.0 を紹介するにあたり、連続紹介の第一回の今回は、そもそもソフトウェア・エンジニアリングはどのように世界に紹介されたか、その後今回の SWEBOK V3.0 が発行されるまでの歴史を振り返って見ることにする。対象とするのは、1) 1968 年の NATO 会議の Software Engineering^{※6} というタイトルの報告書、2) 2001 年 5 月発行の SWEBOK Trial Version、3) 2005 年 6 月発行のソフトウェア・エンジニアリング基礎知識体系—SWEBOK 2004—日本語訳版であり、それぞれを抄訳にて紹介する。第二回及び第三回は、SWEBOK V3.0 の知識領域を 2 回に分けて説明する。

なお、今回用いる技術用語は、SWEBOK V3.0 日本語版の訳者により「SWEBOK V3.0 原語対訳表」が追加されているので、基本的にそれに従うものとする。場合により、ISO/IEC/IEEE Software and Systems Engineering Vocabulary (SEVOCAB^{※7}) を参照する。

2. 1968 年の NATO 会議の Software Engineering というタイトルの報告書

オリジナル版の報告書は 1969 年 1 月に発行されており、その後報告書をスキャンし OCR 技術を用いてテキスト版が作成され、編集されたものが、2001 年にニューキャッスル大学のサーバーから広く入手できるように

なっている。本紹介は、ダウンロード版に依り、抄訳は前半のソフトウェア・エンジニアリング全般に関する議論の部分である。ぜひ報告書をダウンロードの上、前半部分のみならず後半部分も読まれることをお勧めする。今日に通じるものが既に 1969 年初めには広く報告されていることが分かる。後続の SWEBOK の進化の紹介でも明らかになるが、学問的基礎及び実践的な規律の上に成立するエンジニアリングが「身に付く」ためには 50 年という年数では足りないのではないかと思わざるを得ない。

2.1 NATO 会議の背景

1967 年秋に、NATO 科学委員会はコンピューター科学に関する作業部会を発足させている。その年の暮れには、ソフトウェア開発者がその開発にあたり、他のエンジニアリング分野と同様学問的基礎に立脚すると共に実践的な規律に基づくことが必要であり、それを表す用語として、「ソフトウェア・エンジニアリング」という新しい用語を生み出している。この作業部会は、ソフトウェ

【脚注】

- ※1 松本吉弘訳、ソフトウェアエンジニアリング基礎知識体系—SWEBOK V3.0—、オーム社、2014 年 11 月 25 日、ISBN978-4-274-50521-8
- ※2 SC7/WG20 は、ソフトウェア及びシステムの知識体系とプロフェッショナルの形成にかかわる国際標準を審議する委員会、SWEBOK を審議。
- ※3 以下から入手可能、
http://cisas.unipd.it/didactics/STS_school/Software_development/Trial_Version1_00_SWEBOK_Guide.pdf
- ※4 以下から入手可能、
<http://www.computer.org/portal/web/swebok/2004guide;jsessid=dba3ca44da150499851e19bd752a>
- ※5 以下から入手可能、
<http://www.computer.org/portal/web/swebok/swebokv3>
- ※6 Software Engineering, Report of a conference sponsored by the NATO Science Committee, Garmisch, Germany, 7th to 11th Octo., 1968、本報告書は以下から入手可能、
<http://homepages.cs.ncl.ac.uk/brian.randell/NATO/>
- ※7 IEEE と ISO/IEC/SC7 の共同プロジェクトで以下から用語が検索可能、http://pascal.computer.org/sev_display/index.action

ア的设计、ソフトウェアの生産、及びソフトウェアのサービスの観点で議論を進めることを決めた。

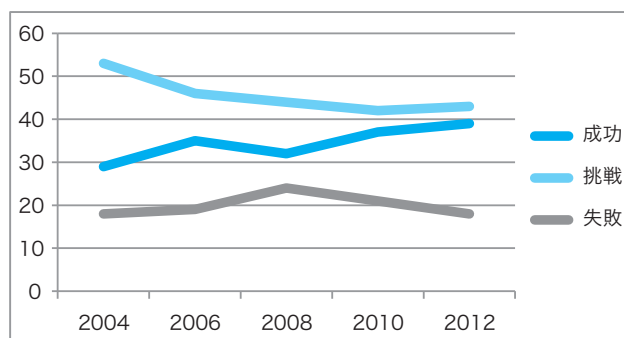
2.2 ソフトウェア・エンジニアリングと社会

前述作業部会は、コンファレンスを組織化し、後日発行される報告書を幅広い読み手に読んでもらうために技術論のみならず、幅広い観点での議論を進めた。この議論の中には以下のようなものがあった。

- コンピューターの導入は年率 25 ~ 50% で増えており、これらの増加に対応するソフトウェアのために 25 万人にのぼる分析者及びプログラマに何らかの影響を与える。
- ほとんどのプログラムは正常に動くであろうが、そうでないプログラムも出てくる。
- 危機とまでは言えないかも知れないが、とくに大規模システムにおいては、危惧がある。
- 大規模システムにおいては、ソフトウェアの不具合の発生をなくすことはできない。
- 他のエンジニアリング分野と比べ、ソフトウェア・エンジニアリングはまだその初期段階にある。
- プログラミングのコスト、スケジュール管理は、依然として低い評判のままである。
- ソフトウェア開発の工程管理の難しさは、進捗をどう測定するのがよいのかわかっていないことにある。
- ソフトウェア不具合は指数関数的に増加している。
- ソフトウェア開発への要望は現場の能力を超えてなされている。

これらは、驚くことに 2015 年の今における発言と異なるものはほとんどない。NATO の会議でソフトウェア・エンジニアリングの基礎及び方向性が見定められたのだとすると、この間の進歩は何であったのであろう？ベンチマーキングを数千のプロジェクトデータで長年実施し Chaos Manifesto 2013^{**8} として報告しているものに、次のデータがある。ここでは、「成功」とは、品質、コスト及び納期が予定した通り及び凌駕したものを示し、「挑戦」とは 3 項目の内どれかが予定に収まらなかったもの、「失敗」とは出荷出来なかったものを示す。

残念ながら 1969 年以降 2003 年に至るデータはないが、最近の 8 年間のデータで見ると、その間、「成功」と「挑戦」において約 10% の改善は見られるものの、ほとんどフラットと言ってもよいであろう。成功が



50% に満たないエンジニアリング成果は他の分野では見られないものではないか。これは、主として欧米のデータであり、日本はこれらより良いと言われている。IPA/SEC の調査ではこれほどの期間のものはないが、スナップショットで、成功が大凡 80% となっている。ただ、日本のデータは母集団が小さい上にプロジェクトの大きさが Chaos Manifesto では小規模のものに相当し、その場合約 70% の成功になっている。NATO 会議においても、ソフトウェアが大規模化していることが問題の出発点と指摘している。

2.3 ソフトウェア・エンジニアリング

コンファレンスを通し、ソフトウェア・エンジニアリングの様々な側面が議論された。

2.3.1 ソフトウェア・エンジニアリングの性質

- ソフトウェア開発はフィードバックループであり、それを通して改善がなされる。
- 設計と実装のペアが繰り返され、再構築を経て最終の成果物になる。
- 設計の段階でユーザが使用時のオプションに関して関与できない。
- どの段階においても、外部仕様はユーザに入手可能なものを示し、内部仕様は外部仕様を実現するプログラム構造を示す。
- プロジェクトの初期の段階から最終段階まで外部仕様にかかわるものと内部仕様にかかわるもの間にはフィードバック系ができていなければならない。
- ソフトウェアシステムの構造として、一番上にアプリケーション、次いでミドルウェア、そして、サービスルーチン及び制御プログラムとなっている。よ

【脚注】

※ 8 The Standish Group International, Chaos Manifesto 2013

り下位のものを変えることはより上位のものに影響を与えるので、維持に経費がかかる。

2.3.2 ソフトウェア・エンジニアリングの管理と方法論

- ソフトウェア設計の方法論は、プログラムとは何かの理解の下に、それが開発される手法、手順及び技術から構成されている。このことは、設計の方法論はソフトウェア・エンジニアリングの管理と密接な関係にあることを示す。
- プログラミングには依然としてアートの部分があるが故に、技術の側面を更に教育し、それを実践するように動機付けなければならない。
- ハードウェア及びオペレーティングシステムのリリースから独立したパッケージというものは存在し得ない。これは、システムエンジニアリングの範疇に入る。(既に、この時点でシステムズエンジニアリングではなく、システムエンジニアリングという用語が用いられている。)
- プログラム設計及び生産の管理のために、以下の項目を考察する必要がある。
 - 設計プロセス
 - 設計プロセスを実施するための組織
 - プログラムの文書化
 - 汎用コンピューター用のツールの開発

2.3.3 ソフトウェア・エンジニアリングにおける設計と実装

- 設計と実装が分けられているのは、担当する人が異なることによる。パフォーマンスは実装の影響が大きく、設計がパフォーマンスに責任を持つのであれば、実装者が何も判断せずに実装できるような設計がなされている必要がある。
- 製品の品質は設計による。
- 設計プロセスは繰り返されるものであり、とくに大規模システムの場合、他の繰り返して問題が分かったとしても自分の仕様を変更するのは締め切りの関係で困難な場合があり、0版、1版、N版というものが作られてしまうことになる。
- 一般的に実践されているものとして、先ず仕様が決められ、それに基づいて設計がなされ、更に実装と進む。設計者が仕様の対象に関して無知の場合、何が起る？

3. 2001年5月発行の SWEBOK Trial Version

1993年5月にIEEEコンピューターソサイエティは、ソフトウェア・エンジニアリングを専門分野として確立するための実行委員会を設立することを決め、1993年8月にACMも同様な決定をした。同年9月から12月まで両者は合同会議を開き、1994年1月に両者は合同実行委員会を開くことで合意し、以下の分野において適切な基準を作ることにした。

- 要求される基礎知識体系と推奨されるプラクティスを定義する。
- 倫理と専門職として充足すべき基準を定義する。
- 学部、大学院及び生涯教育のためのカリキュラムを定義する。

1994年から1996年にかけて、上記の最初の項目のためのタスクフォースが設置された。1996年までに、タスクフォースは基礎知識体系を定義するにはとてつもない経費が必要とされることを認識し、過去40年にわたり開発され、展開されている基礎知識体系の一覧及びガイドとすることを決めた。この基礎知識体系そのものは静的なものではなく動的なものである。1993年から2000年にかけて、IEEEとACMは、Software Engineering Coordinating Committee (SWECC) を通じて共同してソフトウェア・エンジニアリングを専門職分野とすることに協働してきた。

SWEBOK Trial Version は、以下の目次にてカバーされている。

- 第1章 ガイドへの序説
- 第2章 ソフトウェア要求
- 第3章 ソフトウェア設計
- 第4章 ソフトウェア構築
- 第5章 ソフトウェアテスト
- 第6章 ソフトウェア保守
- 第7章 ソフトウェア構成管理
- 第8章 ソフトウェアエンジニアリング・マネジメント
- 第9章 ソフトウェアエンジニアリングプロセス
- 第10章 ソフトウェアエンジニアリングのためのツールおよび方法
- 第11章 ソフトウェア品質

- 付録 A ソフトウェアエンジニアリング基礎知識体系トライアル版のための知識領域記述のための仕様
- 付録 B ソフトウェアエンジニアリング基礎知識体系トライアル版のための関連する規律のリスト
- 付録 C ブルームの教育評価分類法によるトピックスのクラス分け
- 付録 D コンポーネント統合知識分野のための構造の提案

第 1 章において、ソフトウェア・エンジニアリングの定義として IEEE コンピューターサイエティ用語集^{※9}から以下を採用している。

- 1) ソフトウェアの開発、運用及び保守に対して、システムチェックでよく訓練された定量化可能なアプローチを適用すること、すなわち、エンジニアリングをソフトウェアに適用すること。
- 2) 上記アプローチに関する研究。

本 Trial Version は、以下の大規模な査読及びコメントの産物である。

- 3 回の査読工程
- 42 カ国 500 人の査読者
- 9,000 件のコメント

4. 2005 年 6 月発行のソフトウェアエンジニアリング基礎知識体系 – SWEBOK 2004 – 日本語訳版

本バージョンでは、以下の査読及びコメントが寄与した。

- 21 カ国から 124 人の査読者
- 1,024 件のコメント

以上は個人の Web 登録を経由するものであるが、加えて以下の関連する組織における査読及びコメントの提出があった。

- IEEE CS/ACM Computing Curricula Software Engineering
- IEEE CS Certified Software Development Professional プロジェクト
- IEEE Software Engineering Standards Committee
- American Society for Quality Software Division
- Canadian Council of Professional Engineers

結果として、Trial Version と 2004 年版は、内容において改善されたものはあるものの第 1 章から第 11 章までは同じ知識領域をカバーし、以降以下のように修正されている。

- 第 12 章 ソフトウェアエンジニアリングに関連するディシプリン
- 付録 A SWEBOK の Ironman バージョンにおける知識領域記述のための仕様
- 付録 B SWEBOK へのガイドの進化
- 付録 C IEEE および ISO ソフトウェアエンジニアリング標準の SWEBOK 知識領域への割りつけ
- 付録 D ブルームの教育評価分類法によるトピックスのクラス分け

5. 終わりに

インダストリー 4.0、Internet of Things (IoT)、System of Systems (SoS)、Cyber Physical System (CPS) という用語が広く語られるようになってきている。これらのベースはソフトウェアにあり、その社会における重要性は非常に高いものになっている。今回 SWEBOK V3.0 を紹介する機会が与えられ、ソフトウェア・エンジニアリングという用語そのものがいかに世の中に出現したのか、そのオリジンにかかわる報告書を調査すると共に、SWEBOK そのものの進化もトレースした。1968 年にコンピューターにかかわった人々の問題意識は今日現在と同様なことに驚くと共に、ソフトウェア・エンジニアリングという専門分野を担当する専門職として切磋琢磨する必要性を感じている。SWEBOK の策定及びその維持プロジェクトにおいては、実践への考慮をしている。学問的基礎の下の実践が幅広く身につくためには 50 年という年数は短いのかも知れないが、ソフトウェアの社会への影響を考えた時にまだまだ時間が必要とするのではなく、積極的に日々のプロジェクトにソフトウェア・エンジニアリングの新しいプラクティスを導入するようになりたいものである。次回から、SWEBOK V3.0 の具体的な内容を紹介してゆく。

【脚注】

※ 9 IEEE Standard Glossary and Software Engineering Terminology, IEEE, 1990