

2013年度 第8回特別講義 レポート

日時	2014年1月17日(金) 10:00~12:00
会場	(財)日本科学技術連盟・東高円寺ビル 2階講堂
テーマ	品質知識体系とソフトウェア開発技術 - SQuBOK を中心に -
講師名・所属	鷲崎 弘宜氏(早稲田大学)
司会	演習Ⅲ主査 小池 利和氏(ヤマハ(株))
アジェンダ	<ul style="list-style-type: none"> <li>・エンジニアリングとは: 科学的基盤、妥当な知識、コミュニティ、社会への価値</li> <li>・品質の知識体系: SWEBOK、SQuBOK</li> <li>・知識体系の活用: ポータル、辞書、文献ガイド</li> <li>・SQuBOKによるソフトウェア開発技術と品質の俯瞰</li> <li>・SQuBOKにみる主要な品質組み入れ技術: 設計、実装、運用ほか</li> </ul>
アブストラクト	<p>あなたのソフトウェア品質活動は「正統なエンジニアリング」を成していますか？ 妥当と認められた知識に基づいていますか？ 概念や技術の理解は、組織内で共有されていますか？</p> <p>理論により裏打ちされた品質の知識体系があってはじめて、品質に関する技術的活動の範囲が定義づけられ、それを共通の規律および言葉として同業者コミュニティを形成でき、社会発信を通じてようやく正当なエンジニアリングと認められます。</p> <p>ソフトウェア品質の領域でまさしくその基盤を得るべく、SQuBOK 策定委員会では品質の概念、品質マネジメント、品質技術について妥当と認められる知識群を整理し『ソフトウェア品質知識体系ガイド - SQuBOK Guide』第1版を2007年に公開しました。</p> <p>ただし第1版で扱う品質技術は、品質保証の技術に限られていました。高品質を確保し続けるためには、ソフトウェア品質に関わるあらゆる立場の人々が、品質の組み入れと保証の両方について妥当な知識を把握共有することが重要です。そこでSQuBOK 策定委員会では、品質の組み入れに関わる開発技術やマネジメントを中心に知識を拡充・整理した第2版をまとめつつあります。</p> <p>本特別講義では、これらのエンジニアリングや知識体系の定義と必要性を説明したうえで、SQuBOK・SWEBOKを中心に品質に関わる種々の知識体系および活用法を解説します。さらにSQuBOK 第2版を用いて、設計に代表される品質の組み</p>

入れ技術群を俯瞰および整理し、幾つかの主要な技術を解説します。

## <講義の要約>

### ◆エンジニアリングとは

エンジニアリングとは科学に裏打ちされた技術活動、学問体系である。我々のソフトウェア品質技術活動は正統なエンジニアリングであろうか。正統なエンジニアリングであるためには、まずはコミュニティ(SQiP 研究会や SQuBOK 策定委員会など)が不可欠である。そしてコミュニティには知識の妥当性・有効性・適用領域などを判断できる環境が存在し、コミュニティにおいて知識が共有され、それらが科学的基盤の上に成り立っていることが必要となる。その上で、ひとり一人の職業人が社会の中で果たす判断、行為、助言などが社会にとって意味のある価値を提供していることが必要となる。

知識体系として SQuBOK があると何が嬉しいのか。我々は知識を得るために、書籍を読んだり講習会に参加したり様々な場面を通じて、知識を高めたり深めたり広げたりしているが、これだけでプロフェッショナルとしての価値を形成することは難しい。なぜならベストプラクティスに裏打ちされたプロフェッショナリズムを高めるためには、知識だけではうまくいかず、経験やガイドが必要となるからである。

知識が整理されずに孤島のように散々していると、知識の紐づけを考えることは困難となり、プロフェッショナリズムの達成は遅くリスクが高い。知識が整理され体系として存在し、その上にパターン、プラクティス、手法を持ち、それを活用することができれば、我々はより良く早くプロフェッショナリズムを達成することができる。

### ◆品質の知識体系の活用

#### ○知識体系 Body Of Knowledge

知識体系 BOK とは、妥当と認められた知識群を整理・構造化した全体、専門領域の定義づけである。多くのものは以下のように構造化されている。

BOK ガイド(知識へのガイド)

カテゴリ(知識領域の大分類)

知識領域(知識の技術・プロセス上の分類)

トピックまたは知識項目(技術・プロセス知識)

文献(知識の詳細記述・実体)

ソフトウェアを取り巻く世界を、組織、ビジネス、システム、ソフトウェアという階層で考えると、以下のものがあり、相互に関係している。

SQuBOK:ソフトウェア品質知識体系ガイド→ソフトウェア

SWEBOK:ソフトウェア工学知識体系ガイド→ソフトウェア、システム

SEBOK:システム工学知識体系ガイド→システム、ビジネス  
REBOK:要求工学知識体系ガイド→システム、ビジネス  
PMBOK:プロジェクト管理知識体系ガイド→システム、ビジネス、組織  
BABOK:ビジネスアナリシス知識体系ガイド→ビジネス、組織

今後 BOK は増えていくことが予想される。大事なことは、BOK が扱っている範囲を知って、自分達が高めたい、深めたい、広げたいと思っている知識が、何であるのかを BOK から見つけ出すことである。

## OSWEBOK について

SWEBOK はソフトウェア工学に関する知識体系である。

知識領域 (KA:Knowledge Area) は、要求、設計、構築、テスト、保守、構成管理、マネジメント、プロセス、ツールおよび手法、品質で構成される。

SWEBOK におけるソフトウェア品質はソフトウェアエンジニアリングのいたる所に関わる問題のことで、知識領域にはソフトウェア品質 KA だけでなく他のすべての KA も関係する。ソフトウェア品質の基礎概念、ソフトウェア品質マネジメントプロセス、実践上の考慮事項で構成される。

KA 間にはリンクがあり、双方向に行き来することで、品質のきっかけを得られるようになっている。特に重視しているのは、ソフトウェア品質保証のプロセスと V&V の設計・検証で、それをサポートする計測やマネジメントが含まれる。

SWEBOK では倫理的な側面を明確にしており、海外では重視する傾向にある。

SWEBOK は v3 として 2014 年に改訂予定である。トピックは以下の通りである。

### 全体

新設 KA:プロフェッショナル実践、経済、計算基礎、数学基礎、一般基礎

※昔ながらの狭いソフトウェアではなく、広く知っておくべき KA が追加される

※社会的な要請からも知っておく必要がある

既存の KA での拡充・新設

#### テスト KA

拡充:テスト目的(ユーザビリティ・インタラクションテスト、テスト駆動開発)

拡充:入力ドメインに基づく手法(ペアワイズテスト)

新設:モデルベーステスト手法

新設:テストツール

#### 品質 KA

拡充:安全性(セーフティ):セーフティハザード分析技法など

※社会的な要請からも知っておく必要がある

拡充:品質ツール

SWEBOK はソフトウェア工学を広く扱っており、ソフトウェア品質については浅い部分もある。ソフト

ウェア品質に絞って深掘されたものが必要で、そのために SQuBOK がある

## OSQuBOK について

SQuBOK はソフトウェア品質に関する知識体系である。

SQuBOK はソフトウェアの重要性が大きくなるという位置づけの変化、品質事故が発生するという実情の中、日本のソフトウェア品質の暗黙知を形式化、最新のテーマを整理・体系化、品質技術の認知度向上、ソフトウェア品質技術者資格制度、人材育成・組織支援を目的に、品質保証に携わるエンジニア、ソフトウェア技術者・管理者を対象に、2007 年に第 1 版が出版された。

その後、業界変化に伴う品質技術拡充や概念変化への対応が必要となった。スマートフォン、iPad などの新デバイスの隆盛、Twitter 等の新玉コミュニケーション手段の流行、クラウドコンピューティングによるビジネスモデルの変化などがこれに当たる。2014 年に第 2 版を出版予定である。ポイントは以下の通りである。

最新化：国際規格、IT 業界変化への対応

※早く出して随時お客さんと直していくといった新しい領域への対応が必要になっている。

樹形図や品質用語を見直し、一貫性と網羅性を確保

※不具合・欠陥は文脈によるが基本的に Fault、Failure を包括している。

※バグはプログラム記述中の Fault である。

### 3. (第 1 版では着手することのできなかつた)開発領域の拡充

すべての人(V&V 開発の前半に携わる人)に品質の意識を広げる

- ・要求分析、基本設計、詳細設計、実装に対する品質の検討・組み入れ技術を追加
  - ・個々のプロセスについての管理サイクル(計画→実践→評価)を追加
- 要求分析のマネージメント、設計のマネージメント、実装のマネージメントなど

た、具体的な追加・改訂は以下の通りである。

ソフトウェア品質の基本概念

品質のマネージメントの概念【改】

ソフトウェアの品質マネージメントの特徴【追】

ソフトウェア品質マネージメント

ライフサイクルプロセスのマネージメント【改】

品質管理、トレーサビリティ管理、情報文書管理【追】

プロジェクトマネージメント【改】

要求分析のマネージメント、設計のマネージメント、実装のマネージメント、リリース可否判定【追】

運用のマネージメント、保守のマネージメント【改】

ソフトウェア品質技術

モデル化の技法、形式手法、設計の技法、実装の技法【追】

ユーザビリティ、セーフティの技法、セキュリティの技法【追】

要求分析の技法、レビューの技法、テストの技法、運用の技法、保守の技法【改】

記述内容に注目すると、知識の【定義】以外に【目的】【方法】【効果】【参考文献】【関連トピック】を併記している。ハンドブックのイメージで活用できる。

#### ◆知識体系の活用

知識体系を活用するシーンは以下の3点である。

- ・ソフトウェア品質の全体を俯瞰するポータルとして活用する。
- ・ソフトウェア品質の辞書として活用する。
- ・厳選された文献ガイドとして活用する。

別の見方をすると、自分が携わっているものから入って、それが全体の何処にあり、それが合致しているのかを判断することができる。そして、それが関連している技術を見つけ、別の領域へと知識を広げることに活用できる。

過去10年分のSQiP研究会の報告書を、SQuBOK第2版の知識領域に沿って整理し、SQiPライブラリとして公開している。

また、SQuBOK活用ノウハウの共有、SQuBOKユーザの声をSQuBOKガイドへフィードバックを目的に、SQuBOKユーザ会が活動している。

#### ◆SQuBOKにみる開発技術と品質

##### ○工程に共通の品質技術

メトリクスに加え、モデル化の技法、形式手法を追加している。モデル化の技術では、離散系のモデル化技法(UML, SysML, 構造化アート PAD)、連続系のモデル化技法(Matlab, Simulink)、ドメイン特化言語を取り上げている。形式手法には、形式仕様の技法、形式検証の技法を取り上げている。現実世界の何かを直接的に扱うことが経済的・安全的に効率よくない場合、それを代替するものがモデルである。ソフトウェアシステム開発においても、様々なモデルが活用されている。

アプリケーションにおいては、離散系のモデリングが使われ、ソフトウェアモデリング言語やUMLを使って設計、検証・シミュレーション、コード生成を実施している。制御系においては、連続系のモデリングが使われ、ブロック線図や Simulink を使って設計、検証・シミュレーション、コード生成を実施している。

また、機械や電気を含めたシステム全体としての検討が必要な場合、システムモデリング言語の SysML やアーキテクチャ記述言語の ADL で設計、シミュレーションが可能である。

##### 活用事例: JAVA スクリプトのアプリケーションに内在するバグ検出

コードを見ながらアプリケーションのバグを見つけることは難しいが、コードが現実世界、状態遷移がモデルと考えて、コードから状態遷移図を自動生成し、これを見ながらバグを見つけると、網羅的に状態空間を形式的に検証することが可能となるので、バグを見つけやすくなる。

##### ○工程に個別な品質技術(前半)

工程に個別な品質技術(前半)には以下のものがある。

## 要求分析の技法

要求抽出:ステークホルダ識別、要求開発・Openthology(\*)

要求分析:機能要求分析、非機能要求分析、品質機能展開、要求可変性分析

要求定義:USDM

要求の妥当検証確認

## 設計の技法

方式設計の技法:部品化の技法、アーキテクチャパターン、

アーキテクチャ方法論(品質駆動設計・評価ほか)、DSM、フレームワーク

詳細設計の技法:テスト駆動開発、デザインパターン、設計原則

(\*)Openthology(コタツモデル):ステークホルダが(コタツに入って)膝を突き合わし一緒に仕様を定義していく開発スタイルのこと

## 活用事例:リダイレクトシステム開発

要求の段階では、品質要求を明確化するために、品質シナリオ、ゴール指向分析を活用した。設計段階では、品質を着実に組み入れるために、パターン、品質駆動設計・評価を活用した。

品質シナリオとは品質に特化した分岐のない物語のことである。ある刺激源から刺激が成果物に到着したら、成果物から応答が出て、その応答を何らかの方法で測定する。この時、成果物がどういう環境(状態)なのかを記述する。例えば「(環境)通常稼働時に、(刺激)リダイレクト情報に変更があった場合、(応答)運用を止めずに反映できる」「(環境)通常稼働時に(刺激)高負荷をかけた場合(応答)HTTP リクエストを秒間 1000 トランザクション処理できる」などである。これを活用すれば、品質要求を明確にできる。

ゴール思考分析は、実現すべき機能のゴールと品質のゴールを据え、そのためには何が必要で、そのためには何が必要というように、単純に必要なものを上げていく。これを活用すれば、定義したモジュールがゴールに対してムダ・モレなく貢献できていることを確認できる。

アーキテクチャ設計時やコンポーネントの詳細設計時に、品質を組み込む方法として品質駆動設計がある。品質駆動設計では、品質要求に対して、成果物の中の形を設計する時に、実績のある、実現手法、アーキテクチャパターンを活用する。これらを活用するに当たって配慮すべきことはトレードオフである。ある品質に注力すると他の品質に影響が出る。例えばセキュリティを重視するとパフォーマンスは落ちる。また、これらの過程を残しておくことに価値がある。将来の機能追加や修正の時に影響範囲を特定する資料として活用できる。

## ○工程に個別な品質技術(後半)

工程に個別な品質技術には以下のものがある。

### 実装の技法

コーディング規約、リファクタリング、防御的プログラミング・契約による設計、

統合開発環境、継続的統合 CI

運用の技法

データ品質、仮想化、クラウド、ソフトウェア若化

保守の技法

プログラム理解、リエンジニアリング、リバースエンジニアリング、コードクローン分析

活用事例: 開発と運用に見える次の“ABCD”

「A:アジャイルとD.DevOps」は「継続的統合 CI」と「タスク・チケット開発」、「B.ビッグデータとC.クラウド」は「モニタリング」と「データ解析」と密接に関わっていると整理できる。

また「B.ビッグデータとC.クラウド」は「A:アジャイルとD.DevOps」を支える「プラットフォーム」や「フィードバック」の役割を負っていると整理できる。

### ○専門的品質特性の品質技術

専門的品質特性の品質技術には以下のものがある。

ユーザビリティの技法

ユーザビリティテスト

脆弱性(安全性)の技法

リスクアセスメント技法(ISO/IEC 31010)、リスク低減法

セキュリティの技法

セキュリティ要求分析、セキュリティ設計(SDL、脅威分析・モデリングなど)、セキュリティパターン、セキュアコーディング、セキュリティテスト

活用事例: セキュリティパターン

要求では脅威・対策はどうするか、設計では構造はどうするか、実装ではコードはどうするか、セキュリティを専門としていない人が判断することは難しい。世の中で知られている認可パターンを活用することで、早い段階からの作り込が可能となる。

### ◆まとめと期待

- ・本日取り上げた事例を SQuBOK 上にマップすると、何と何がつながっているのかを認識できる。
- ・SQuBOK には品質技術と品質特性と結び付ける表はないが、表を起こすための情報は入っている。
- ・開発計画の立案時に SQuBOK を参照すれば、今回の開発はどの段階でどの技術を活用するのがよいのかを検討できる。
- ・SQuBOK/SQuBOK の中から、パターン、プラクティス、手法を選び、組み合わせて活用できるようになった現在、重要となるのは、活用した結果をこれらにフィードバックし、スパイラルに発展させていくことである。
- ・異なる知識体系群を結びつけ活用するために SEMAT メソッドアーキテクチャがある。エッセンス(言語、カーネル)といった共通の基盤の上で、プラクティス、手法・プロセス(UP やアジャイル)を

比較することができる。

・ISO/IEC 24773(2008)では、様々なコミュニティの BOK を SWEBOK ベースでマッピングし、技術資格者の共通化、フレームワーク化している。

・現状の BOK はクローズド(オーナーはあくまで単一コミュニティ)である。今後は BOK を相互横断的に活用できるようにオープン(知識体系の接続)であることが必要となる。

#### ◆質疑応答(コメント含む)

Q:モデルを作る理由は、経済性や安全性の視点で、実態を取り扱うことができないものを対象とする場合であることが腑に落ちた。コードを見て確認するのと、状態遷移図を見て確認するのとは、どちらが優位なのか、モデルを作ったことのない人にとってモデルは余分な成果物かもしれないが、どちらが経済的なのかを考えるとモデルの価値は高いことが認識できた。

Q1:個人的に技術者の倫理に触れる機会があり、世の中には、エレベーターの話、自動車の話などがあるが、ソフトウェアは見えてこない。今後これらが SWEBOK に含まれる可能性はあるのか教えてほしい。

A1:海外では BOK を据えた上で同業者コミュニティをソサエティとして扱っている。昔のギルドみたいなイメージでそれが BOK に反映されている。情報処理学会などでも倫理の話は出ている。今後のソフトウェア品質認定試験に入ってくるかもしれない。

Q2:倫理は重要なので、他の人がどのように考えているかを知りたい人もいる。そのような情報があるのか教えてほしい。

A2:SQuBOK には入ってはいない。いずれは考える必要がある。現状のソフトウェア認定試験に倫理規定は入っていないが、いずれは倫理規定を遵守する成約をした上で資格認定することが必要となる。

全体のハードウェアを含めた品質管理の世界では倫理規定があって、QC 検定には取り入れられている。そこには普遍的なことが書いてある。

鷲崎氏から補足:

SQuBOK の一番身近なのは SQiP ライブラリである。以前は参照しづらかったが、新しい体系で公開するようにしたことで参照しやすくなった。

また、SqIP 研究会の論文は実際に取り組んだものである。SQuBOK の参考文献となるような論文が出てきてほしい。

#### <講義の感想>

SWEBOK や SQuBOK の知識体系は「辞書」としての認識が強かったのですが、具体的に業務で活用するポイントとして「自分が携わっているものが全体の何処にあり、それが自分達の目的と合致しているのかを判断すること」や「それが関連している技術を見つけ、別の領域へと知識を広げる

こと」を解説して頂いたことで、より身近なレベルで利用できる「ハンドブック」であることを認識できました。目の前にあるプロジェクトの活動を確認してみようと思います。

司会者であります小池様、貴重な講演をしてくださいました鷲崎様に深く感謝いたします。