

## 第7章 ソフトウェアの品質保証

### ソフトウェアの品質保証の内容

「ソフトウェアの品質とは何か」という問題については、既に第5章で述べた。それでは「保証」とは何か。広辞苑第5版によれば、保証とは「大丈夫だ、確かだとうけあうこと」とある。これに従えばソフトウェアの品質保証とは、今対象としているソフトウェアについて、その品質を誰かが使用する人に「大丈夫だ、確かだとうけあうこと」であることを意味している。それではこれは、誰が、どのようにして行うのだろうか。あるいは、こんなことがそもそも可能なのだろうか。この章ではこういった立場から、ソフトウェアの品質保証の問題を考えてみたい。

ソフトウェアを作るための作業の内容を定義した国際規格がある。ISO / IEC 12207 : 1995 と呼ばれるものである。これは JIS 化されて、JIS X 0160-1996 になっているが、この規格がおもしろいのはこの JIS 規格の他に、この国際規格を基に「SLCP-JCF98 (Software Life Cycle Process – Japan Common Frame 98)」と呼ばれる別の基準が作られていることである<sup>1</sup>。SLCP-JCF98 では、単にソフトウェアを作る作業だけでなく、ソフトウェア作りを支援する作業も併せて定義されている。この支援作業の中に、「ソフトウェアの品質保証」が含まれている。これによれば、ソフトウェアの品質保証は次の3つの作業から構成される。

1. 製品の保証
2. プロセスの保証
3. 品質システムの保証

「製品の保証」は、理解できる。それでは[プロセスの保証]や、「品質システムの保証」とは、何を意味するのだろうか。

まず、この段階での議論を最も簡単にすませることができる「品質システムの保証」から検討を始めたい。

### 品質システムの保証

SLCP-JCF98 によれば、「品質システムの保証」とは、「ソフトウェアを開発する組織がその品質管理活動を行う際、ISO 9000 (現時点の規格は ISO 9000 : 2000、該当する JIS 規格は JIS Q 9000) をその基準として適用すること」としている<sup>2</sup>。先でも述べたとおり SLCP-JCF98 の大元は ISO / IEC 12207 : 1995 であるから、ここで同じ ISO 規格である ISO 9000 をその品質活動の基準にあげていることは、当然と言える。

ISO9000 の目的は、これをベースにして企業内に優れた品質マネジメントシステムを構築・維持し、全ての利害関係者のニーズの実現に取り組むこと、特に顧客を重視すること

<sup>1</sup> SLCP-JCF98 が作られた経緯、その内容などについては、第12章で述べる。

<sup>2</sup> ISO 9000 のソフトウェア開発への摘要については、第36章で述べる。

である。この目的から、ISO9000 の適用が品質保証の一部に組み込まれていることは、よく理解できる。

しかし私に言わせれば、ISO 9000 に代わって、ISO 9000 と同じようにソフトウェアの品質システムを作ることができる CMMI ( Capability Maturity Model Integration )<sup>3</sup>や ISO / IEC 15504<sup>4</sup> ( SPICE ) がその組織の規格であっても、良いように思える。

## プロセスの保証

これまで何度かこの原稿の中でも述べたように、現在あらゆる工業製品の品質保証のベースにある考え方は、「製品を作る作り方 ( プロセス ) が良ければ、その結果作られる製品の品質が良い」とするものである。この考え方は自動車やテレビ、工作機械などに広く適用される。この考え方をソフトウェアに摘要したものが、ここで言う「プロセスの保証」である。

何年か前に私がソフトウェアの品質保証について始めて学んだ時、当時ソフトウェアの品質保証はこの「プロセスの保証」が全てだった。少なくとも私は、そう受け止めた。この考え方はその時の私にとって、非常に斬新で、ある意味で衝撃的だった。

ソフトウェア工学の世界で著名な IEEE の用語集 ( Standard Glossary of Software Engineering Terminology ) では、「品質保証」について次のように記述されている [IEE90]。

1. 品種や製品が技術的な要求を満たすことについての適切な自信を持つために必要な、システムティックで、かつ計画された全ての活動に関するパターン
2. 製品が開発され、あるいは製造されるプロセスを評価するためにデザインされた、一連の活動

この規格は 1990 年に作られたものであるが、ここには「製品の品質保証」についての記述がない。このことから、当時の私の受け止め方は適切だったのではないかと考えている。

その時点での「プロセスの保証」とは、「以下の 2 つのことを同時に保証することで達成できる」としていた。

1. 今品質の保証をしようとするソフトウェアを開発する組織 ( プロジェクト、など ) が持っているソフトウェア開発の基準書 / 手順書が、品質の高いソフトウェアを開発する上で十分なものであること。
2. ソフトウェアを開発している期間中、その開発組織はその基準書 / 手順書に記載されている通りに作業を行っていたこと。

この方法でソフトウェアの品質を保証するチームは、ソフトウェアを開発する組織の外側において、開発する人たちの作業を冷静に観察している、ということになる。このようなチームを「品質保証チーム」と呼んでいる。品質保証チームについては、また後で述べる。

私が初めてソフトウェアの品質保証を勉強した 1995 年頃と今とでこの「プロセスの保証」

---

<sup>3</sup> CMMI については、第 37 章で述べる。

<sup>4</sup> ISO / IEC 15504 ( SPICE ) については、第 38 章で述べる。

について変わったところは、基準書／手順書についてである。私が勉強した時は、この基準書／手順書は既に何らかの形で用意されているもの、というところからスタートした。

SLCP-JCF98 では、この基準書／手順書は「契約に従ったものであること」と明記されている。SLCP-JCF98 のベースである ISO / IEC 12207 : 1995 はソフトウェア産業 / 会社の立場を念頭に置いて作られたという経緯があるので、ここでは「契約に基づいた品質保証」、「契約の内容を満足する品質保証」という考え方が明確になっているのは当然と言える。一般の企業の中に位置するソフトウェア部門を対象に考えた場合には、そのソフトウェアのユーザとなる組織との間で、架空の、あるいは明示されていない契約があると考えればよい。いずれにせよ、「ソフトウェアの開発は何らかの契約に従うもの」という考え方がこの場合、開発の根底にある。そしてそこで開発されるソフトウェアは契約ごとに異なる内容と性格のものであるから、そのソフトウェアの内容と性格は契約の中に充分反映されているべきである。そして、「ソフトウェアの内容と性格ごとに、開発の手順（プロセス）が異なるべきである」、あるいは「ある内容と性格のソフトウェアを開発するためには、それに適した開発のプロセスがあるべきである」ということが、この SLCP-JCF98 の考え方の背景にあることになる。

CMMI の中で定義されている品質保証<sup>5</sup>では、「プロセスの保証」で使用する基準書／手順書は、「プロジェクトの開始時点で、ソフトウェアの品質保証チームが参画してそのプロジェクトの手順書を作成し、あるいは別途その開発組織全体が持っている基準書／手順書をそのプロジェクト用に手直しして、そのプロジェクト専用の基準書／手順書を用意し、それに基づいて品質保証活動を行う」と記されている。ここでの手順書／基準書についての考え方も、SLCP-JCF98 と変わらない。繰り返しになるが「ソフトウェアのプロセスは、これから開発しようとするソフトウェアの内容と性格を反映してそれぞれ異なるべき」ということになる。

プロジェクトごとに、あるいは開発するソフトウェアごとに、そこで開発しようとするソフトウェアの内容と性質を見て開発の手順や品質保証の作業などを修正することを、「テラーリング」と呼ぶ。日本語では、「修整」という文字が使われることが多い。テラーリングは、言うのは簡単だが、適切に実行するのはたいへんに難しい。しかしその考え方は、たいへんに良く理解できる。我々はソフトウェア開発のレベルを上げて、必要に応じて適切にテラーリングができる状態に到達しておく必要がある。

もし、手順書／基準書に記載されていないことが行われた場合はどうするか。あるいは、記載されていることが行われなかったらどうするか。この問題は、CMMI の品質保証のプロセスの中で述べられている。CMMI では、この不一致を「非遵守（ひじゅんしゅ、英語では noncompliance という単語が使われている）事項」と呼んでいる。そしてこの事例が発見されると品質保証チームは、まずプロジェクトのメンバーと協力してその解決を図る。

---

<sup>5</sup> CMMI では、ソフトウェアの品質保証は「プロセスと成果物の品質保証」プロセスの中で行う。

それで解決できない場合は、その問題を解決できるレベルの管理者に通告して解決を図るとしている。

なお ISO12207 : 1995 と SLCP-JCF98 では、品質保証の活動に、それ自身の支援ライフサイクルプロセスの中に位置づけされている検証、妥当性確認、共同レビュー、監査、問題解決の各プロセスに依存することが明記されている。妥当な指摘ということができる。

## 製品の保証

製品の品質保証は、例えば ISO / IEC TR 9126-2、-3、-4 で示されているようなソフトウェアの品質に関わる測定項目を定義し、あるいはその趣旨に基づいた項目を選定して、顧客に納品する前にそれを測定して結果をユーザに提示する、あるいはそれに基づいたテストケースを設定してテストを行う、というのが1つの考え方である。

SLCP-JCF98 では、ここでも「契約」という言葉が表面に出ている。開発するソフトウェアごとにその内容と性格が異なるので、その内容と性格に見合った測定項目を定義/選定し、あるいはテストケースを設定して、それを契約の中に明記し、測定やテストを実施するという考え方である。

CMMI では「契約」という言葉は使われていないが、同じような考え方を取っている。つまり CMMI では、「プロジェクトの初期段階で、評価する作業成果物と基準を定め、顧客の納入前にこれら进行评估する」としている。

ただソフトウェアの場合、最終製品が複雑な上、実態が容易に目に見えるものではなく、さらに限られた項目の測定やテストの実施だけで必要とするソフトウェアの幅広い品質全部を保証できるものではないということなどから、「製品の品質保証」より「プロセスの品質保証」の方が、ソフトウェアの品質保証として適切なように、私には思える。

その理由として、CMMI では「プロセスの品質保証」を「製品の品質保証」より先に記述しており、「Encyclopedia of Software Engineering」の品質保証の記述でもプロセスの品質基準の方がより丁寧に記述されていることをあげたい[MAR02]。

## 積極的な品質保証と消極的な品質保証

まだ品質保証チームの活動について記述していないが、品質保証チームが受け身の立場で終始するような品質保証を「消極的な品質保証」、逆に積極的に他のプロジェクトやチームの活動に参画してソフトウェアの品質を向上させる品質保証活動を「積極的な品質保証」と呼んでいる。前述の「プロセスの保証」で述べた2つのことを保証するだけの活動は、消極的な品質保証の典型である。

一般に、積極的な品質保証が消極的な品質保証に勝ることは、言うまでもない。

## 品質保証チームの活動

ソフトウェアの開発組織がソフトウェアの品質保証を行う場合には、この「品質保証チー

ム」を編成し、機能させなければならない<sup>6</sup>。

この品質保証チームは、ソフトウェア開発組織を指揮している管理者より上位の管理者に直属し、報告する形にするべきと、アメリカ生まれの教科書には記述されている[JON96b]。ソフトウェアの開発がぎりぎりの状態に至って、品質を犠牲にしてスケジュールを守るか、スケジュール遅れが発生しても品質を重視するかを選択を迫られたとき、開発に責任を持っている普通の管理者は、品質を犠牲にしてスケジュールの方を取ることが一般的であると言うところに、この記述の背景がある。だから高い品質のソフトウェアを開発するためには、開発に責任を持っている管理者より高いレベルの管理者に品質に関わる問題を報告し、開発に責任を持つ管理者に適切な指示命令をしなければならない。そうしなければ、高い品質のソフトウェアを開発することはできない、と言う考え方である。

品質保証チームの開発組織内での活動は、多岐にわたる。それはこのチームを構成しているメンバーの人数、レベル、組織内での位置づけ、管理者の支援状況などによって異なることになる。

最低限のこのチームの活動は、繰り返しになるが、プロセスの保証のところでは述べたように、以下の2つの事項を保証することである。

1. 今品質の保証をしようとするソフトウェアを開発する組織（プロジェクト、など）が持っているソフトウェア開発の基準書／手順書が、十分な品質のソフトウェアを開発する上で十分なものであること。
2. ソフトウェアを開発している期間中、その開発組織はその基準書／手順書に記載されている通りに作業を行っていたこと。

これが品質保証活動の出発点であり、最低限の品質保証活動である。

これに加えて、品質保証チームは以下のようなことを行うことができる。

- 以下のような事項が開発組織内で、適切に行われていることを保証する。
  - 全てのテストの計画が妥当なものであり、その計画に基づいてテストが実施されていること。
  - 使用されているツールや手法が適切なものであり、正しく使用されていること。
  - エラー処理の手順が妥当であり、問題への対応だけでなく、その原因究明もなされていること。
- 品質について、要員の教育を実施する。
- 品質についてのデータを測定し、結果を分析する（「製品の保証」を含む）。
- 欠陥発生を予測する。

ケイパース・ジョーンズは、品質保証チームの規模は全開発要員の5%程度が必要と述べている[JON96b]。確かにこれだけの数のレベルの高いメンバーがいれば、その活動はかな

---

<sup>6</sup> ソフトウェア技術者の専門職化については、第43章で述べる。

り多岐にわたり、しかも内容が濃いことが期待できて、その組織が作るソフトウェアの品質を高いものにすることができる。

一方チームの人数が少ない、あるいはレベルが低い、などと言うようなことがあれば、このチームの活動は限られたものにならざるを得ず、その結果としてこの開発組織は十分な品質のソフトウェアを作ることができないかも知れない。

しかし問題は、チームの規模やレベルだけにあるのではない。仮に強力なチームがあっても、その活動が制約されるようなことがあれば、結果として高いレベルのソフトウェアを開発することが難しくなる。ポイントは、開発組織の文化がこのような活動を受け入れるかどうか、もっと一般的な言い方をすれば、開発組織が本当に高い品質のソフトウェアを開発する必要性を認識し、それを実践しようとしているのか、と言うことにある。

開発組織の文化がこのようなものでない場合、文化をこのように変え、維持してゆくのは、言うまでもなく管理者の責任である。このことから、ソフトウェアの品質について管理者の責任はたいへんに重い、と言わざるを得ない。

## ソフトウェア品質保証活動の計画

何もソフトウェア作りに限る話ではないが、ソフトウェア作りでは最初に、これから行おうとすることについてしっかりした計画を立て、レビューまで行って、それから実際に行動を起こすことを習慣にするべきとしている。

ソフトウェアの品質保証活動を始めるに当たっての計画について、IEEE に規格がある。この規格名を IEEE Std 730™-2002 というが、この規格の内容は、記述すべき計画の推奨例である [IEE02]。その目次の項目だけを、以下に示す。当然その計画の内容は、現実に実施される品質保証活動の内容を正確に反映したものでなければならない。そのためにここで述べる目次項目の一部の追加や削除があることは、充分にあり得ることと言える。

1. 目的
2. 参照すべきドキュメント
3. マネジメント
  - 組織
  - 作業内容
  - 役割と責任
  - 品質保証に必要なリソース
4. 文書
  - 目的
  - 最低限の文書作成についての要求
  - A) ソフトウェアの要求についての記述
  - B) ソフトウェア設計についての記述
  - C) Verification と Validation についての計画

- D) Verification と Validation の結果についての報告
- E) ユーザへの文書
- F) ソフトウェアの構成管理の計画  
その他の文書
- 5. 標準、プラクティス、習慣、メトリクス
  - 目的
  - 内容
- 6. ソフトウェアのレビュー
  - 目的
  - 最小限の要求
  - A) ソフトウェアの仕様についてのレビュー
  - B) アーキテクチャーの設計についてのレビュー
  - C) 詳細設計についてのレビュー
  - D) Verification と Validation の計画についてのレビュー
  - E) 機能についての監査
  - F) フィジカルな側面の監査
  - G) 進行中の作業についての監査
  - H) 管理面についてのレビュー
  - I) ソフトウェアの構成管理計画についてのレビュー
  - J) 開発終了後のレビュー
  - その他のレビューと監査
- 7. テスト
- 8. 問題についての報告と修正作業
- 9. ツール、テクニック、方法論
- 10. メディア・コントロール
- 11. サプライアのコントロール
- 12. 記録の収集、保守とその維持
- 13. トレーニング
- 14. リスク管理
- 15. 用語集
- 16. ソフトウェア品質保証計画の変更手続きとその経緯

### 再び「ソフトウェアの品質保証」という言葉について

「ソフトウェアの品質保証」という言葉は、3つの意味で使われることがあるという。その3つとは、

1. 「ソフトウェアの品質を保証する」という意味の、素直な概念

2. 開発組織内の、あるチームなどの名称
3. テストやレビューなど、ソフトウェアの品質を高めるための作業

ケイパース・ジョーンズは、3 の立場でこの言葉を使うのは間違いだと指摘している [JON96b]。

私もこの考え方に賛同し、先に述べた「品質保証チームの活動」にこれに関わるものを除外した。テストやレビューなどソフトウェアの品質を高めるための直接の活動は、やはりそのソフトウェアを開発しているプロジェクトの責任である<sup>7</sup>。そのプロジェクトの活動を支援し、結果としてプロジェクトが品質の高いソフトウェアの開発を実現することが品質保証チームの役割であり、責任であると私も考えるからである。

### ソフトウェアの品質保証の位置づけ

ソフトウェアの品質保証の作業は、ISO / IEC 12207 : 1995 の国際規格の中では「支援プロセス」の1つとして位置づけられている。

カーネギー・メロン大学のソフトウェア工学研究所 (SEI : Software Engineering Institute) が設定した CMMI ( Capability Maturity Model Integration : 能力成熟度モデル統合) の段階表現では、ソフトウェアの品質保証はレベル2 (「管理された」) でのプロセスに位置づけられている<sup>8</sup>。このことは、ソフトウェアの品質保証活動はソフトウェア工学全体の中で、非常に基本的なものであることを意味している。

### ソフトウェアの品質保証は可能か

私が「ソフトウェアの開発保証」という言葉を初めて聞いたのは、私が一般企業のソフトウェア開発部門で働いていた時のことだった。その時に私が反射的に考えたことは、「ソフトウェアの品質など、保証できるわけがない」というものだった。私が働いていた開発組織は、ソフトウェア危機のあらゆる症状で苦しんでいた、当時として「普通の」開発組織だった。

今私は、その時の私の考えが間違いであったことを知っている。必ずしも容易ではないけれど、ソフトウェアの品質を保証することは、これまで述べてきたように可能である。今でも以前の私と同じような考えを持っている人がいれば、その考えを是非ここで改めてほしい。そして、ここで述べたようなソフトウェアの品質保証活動を通して、品質の高いソフトウェアを開発してほしい。それを私は、切望している。

### キーワード

ISO / IEC 12207 : 1995、JIS X 0160-1996、SLCP - JCF98、製品の保証、プロセスの

---

<sup>7</sup> ソフトウェアのテストについては第 29 章と第 30 章で、レビューについては第 18 章で、それぞれ述べる。

<sup>8</sup> 既に記したことだが、CMMI については第 37 章で述べる。

保証、品質システムの保証、ISO 9000:2000、CMMI、ISO / IEC 15504、SPICE、品質保証チーム、テラーリング、非遵守事項、Standard Glossary of Software Engineering Terminology、積極的な品質保証、消極的な品質保証、ISO / IEC TR 9126-2 : 2003、ISO / IEC TR 9126-3 : 2003、ISO / IEC TR 9126-4 : 2004、IEEE Std 730™-2002、CMMI

## 人名

ケイパース・ジョーンズ (Capers Jones)

## 参考文献とリンク先

[CMM02] CMMI 成果物チーム、「能力成熟度モデル統合 (CMMI<sup>SM</sup>) , 1.1 版 システムエンジニアリング、ソフトウェアエンジニアリング、統合成果物プロセス開発、および供給者ソーシングのための CMMI<sup>SM</sup> (CMMI-SE/SW/IPPD/SS, V1.1) 段階表現 CMU/SEI-2002-TR-012」, ソフトウェアエンジニアリング研究所、カーネギメロン大学、2002 年 .

この資料は、次の UML からインターネットでダウンロードすることができる。

<http://www.sei.cmu.edu/cmmi/translations/japanese/models/japanese-staged.pdf>

なお、この資料の原書は次のものである。

CMMI Product Team, “Capability Maturity Model Integration (CMMI<sup>SM</sup>), Version 1.1 CMMI<sup>SM</sup> for Software Engineering and System Engineering (CMMI-SE/SW/IPPD/SS, V1.1) Staged Representation CMU/SEI-2002-TR-012,” Software Engineering Institute, Carnegie Mellon University, 2001 .

この英文の資料は、次の UML からインターネットでダウンロードすることができる。

<http://www.sei.cmu.edu/pub/documents/02.reports/pdf/02tr002.pdf>

[IEE90] IEEE Computer Society Standards Coordinating Committee, “IEEE Standard Glossary of Software Engineering Terminology IEEE Std 610.12-1990(R2002),” IEEE, 1990 .

[IEE02] IEEE Computer Society Standards Coordinating Committee, “IEEE Standards for Software Quality Assurance Plans IEEE Std 730™-2002,” IEEE, 2002 .

[ISO95] 日本規格協会、「ソフトウェアライフサイクルプロセス International Standard ISO / IEC 12207 英和対訳版」, 日本規格協会、1995 年 .

[ISO03a] ISO / IEC, “Software Engineering – Product Quality – Part 2 : External metrics,” ISO, 2003 .

[ISO03b] ISO / IEC, “Software Engineering – Product Quality – Part 3 : Internal metrics,” ISO, 2003 .

[ISO04a] ISO / IEC, “Software Engineering – Product Quality – Part 4 : Quality in use

metrics,” ISO, 2004 .

[JIS96] 日本工業標準調査会審議、「ソフトウェアライフサイクルプロセス JIS X 0160-1996」, 日本規格協会、平成 8 年 .

[JON96b] Capers Jones 著、富野壽監訳、「ソフトウェア品質のガイドライン」, (株)構造計画研究所、1999 年 .

[MAR02] John J. Marciniak Ed. “Encyclopedia of Software Engineering Second Edition,” John Wiley & Sons, 2002 .

[SLC98] SLCP-JCF98 委員会編、「共通フレーム 98 SLCP-JCF98 1998 年版」, 通産資料調査会、1998 年 .

(2004 年(平成 16 年)6 月 8 日初稿作成)