

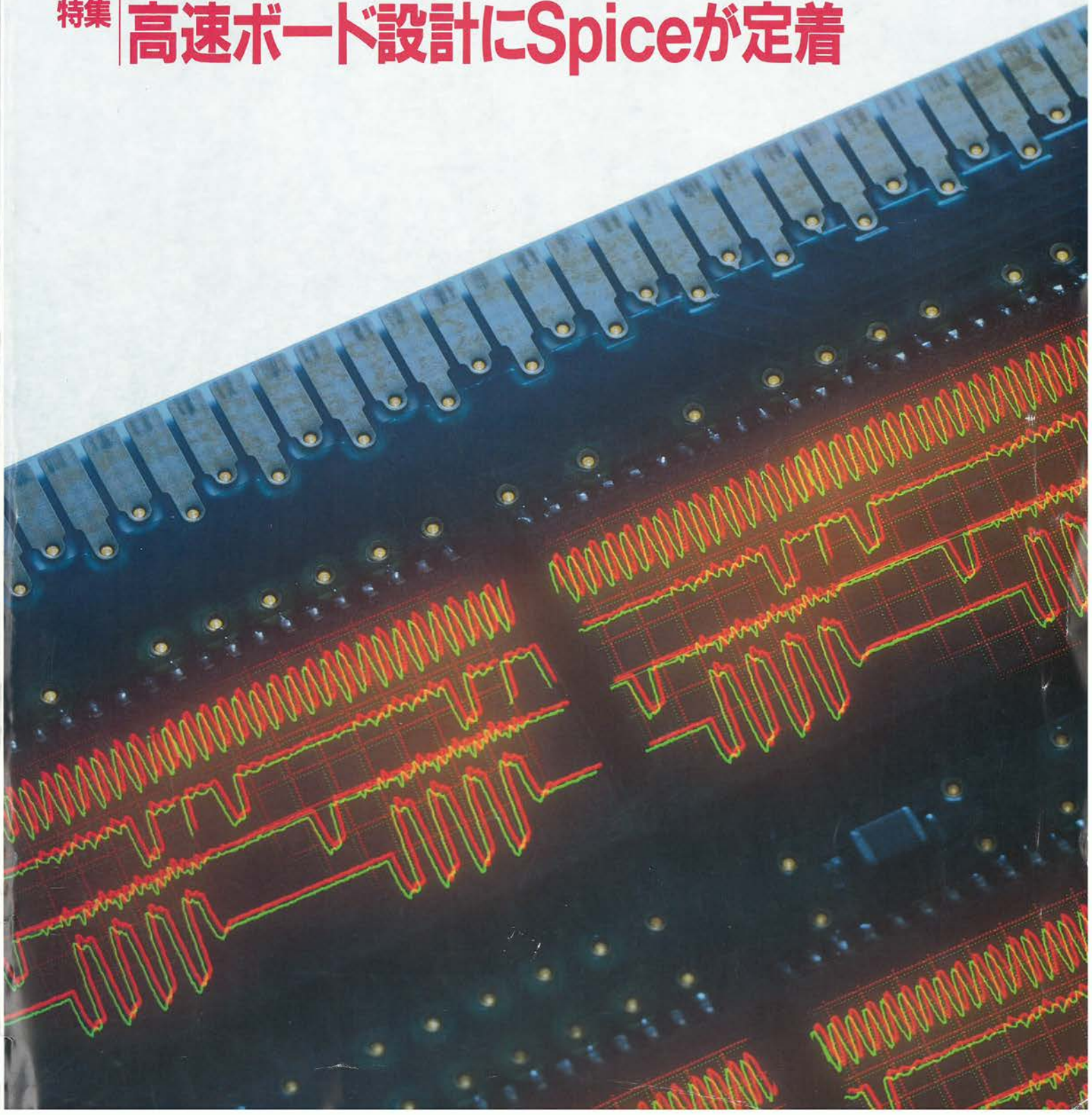
日経エレクトロニクス

NIKKEI ELECTRONICS

1995 7-31

特集 **標準DRAMをいつまで使うか**

特集 **高速ボード設計にSpiceが定着**



講座

ソフトウェア
開発

SEIの発展段階モデルCMM, 賛否両論あるなか普及へ

Hossein Saiedian

米 University of Nebraska at Omaha

Richard Kuzara

米 Sterling Software 社

米 Software Engineering Institute (SEI) が開発した「CMM (Capability Maturity Model)」が普及に向けて動き出した。ソフトウェア開発組織に求められる「品質の高いソフトウェアを日程通りに開発する能力」を、どの発展段階に到達しているかによって定量的に評価するモデルである。評価結果をもとに開発工程を改善する。大きな成果を上げた事例が報告される一方で、根強い批判もある。とはいえ、米国防総省という強い後ろ盾があるため業界標準になるのは確実とみられ、採用企業が増えている。(本誌)

米国の多くの政府機関は、自らのソフトウェア開発部門や委託先である民間のソフトウェア開発業者のソフトウェア開発体制を改善したいと強く望んでいる。先頭に立っているのは米国防総省(DoD: Department of Defense)だ。

そうした動きのなかで最も注目すべきは、米 Carnegie Mellon University (CMU) の Software Engineering Institute (SEI) が政府機関向けに開発した5段階の発展段階モデル「CMM (Capability Maturity Model: 能力成熟モデル)」である。このモデルには「査定 (assessment)」の手続きと、なにかと物議をかもしている「評価 (evaluation)」の手続きがある。

入札業者の絞り込みに使う

Griffiss 空軍基地 Rome Labora-

tory の空軍部門は1991年9月25日付の書簡で、米政府機関との契約に基づいて入札し業務を請け負うコンピュータ・ソフトウェア開発業者に対して、以下のように通知した。

「しかるべき近い将来、すべてのソフトウェア開発業者はソフトウェア開発工程の成熟度(発展段階)がレベル3であることを示さないと、ESD/RL (Electronic Systems Division/Rome Laboratory) が発注する主要なソフトウェア開発プロジェクトに入札できなくなることを指摘しておきたい。…いまこそ万一のために準備を始めるべきときである」。

賛否両論が渦巻く

ソフトウェア業界の反応は賛否両論だった。この書簡をきっかけに、ソフトウェア開発業者は重要で新しい試みを始めた。少なくともその動

Translated and reprinted, with permission, from the original English version which appeared in *COMPUTER*, vol.28, no.1, pp.16-26, Jan. 1995. © 1995 IEEE.

ソフトウェア産業の開発部門向けに Software Process Maturity Model (ソフトウェア開発工程成熟モデル)を開発した。空軍は米 Mitre Corp. に、この開発作業に参加するよう要請した。SEI と Mitre 社の共同チームは、ソフトウェア開発工程の成熟度という観点から開発部門を評価するための質問票と評価の枠組みを作成した。これは、すでに業界で知られていた W. Edward Deming の原理と Walter A. Shewhart の工程管理の概念 (Deming の著書 *Out of Crisis* に記述がある) を組み合わせたものだった。

続いて SEI は、1991 年に CMM (能力成熟モデル)を開発した。CMM は SEI が用意した質問票を引き続き発展させ改善していくための枠組みとなる。SEI の研究成果を見直し、その内容に対する変更の提案が適切かどうかを判断するために、Questionnaire Advisory Board (質問票諮問委員会)を設置した。最も影響を受ける組織の要求と関心のバランスをとるため、同委員会には米国のソフトウェア産業と政府機関の両方が委員を出している。

査定と評価に使う

CMM は 5 段階からなる発展段階モデルである (図 1)¹⁾。低い段階の能力を土台にして、より進んだ高い段階に能力を積み上げていくように設計されている。どの発展段階 (成熟度レベル) にあてはまるかによって、開発部門のもつソフトウェア開発工程の能力をランク付けする。

CMM とそれに関連した質問票は、主に査定と評価という 2 通りの使

い方がある¹⁾。査定では、開発部門が発展段階モデルを使って自分の部門の運用体制を検討し、優先度の最も高い改善項目を洗い出す。これは、その部門の自己改善活動の実行計画の基礎となる。一方評価では、購買部門がこの発展段階モデルを使って入札資格のある業者を選定し、進行中の契約について監視していく。こうすることによって、最も頼りになる有能な業者を選び出すのに使ってきた従来の評価基準を改善して、リスク・プロファイルを作成できる。

成熟度を 5 段階に分類

話を簡単にするため、CMM で定義された 5 段階を簡略化して、① 初期段階、② 再現可能段階、③ 定義段階、④ 管理段階、⑤ 最適化段階と表現する。SEI がこのように 5 段階にした理由は以下の通り¹⁾。

▷徐々に改善を進めていく過程を表現しやすい。

▷達成可能な改善手順を無理のない順序で示せる。

▷中間的な改善目標を提示し、進行度合いを把握できる。

▷この枠組みで開発部門の成熟度を調べれば、すぐに改善すべき事項の優先度を示せる。

一般に CMM の 5 段階は、それぞれ次のように説明される。

① 初期段階：開発部門の管理にはいろいろな程度があるが、最初のステップは日程と費用を大まかに見積もることが目標である。この段階は「アドホック (気まぐれな)」とか「カオティック (混沌とした)」といった、受けをねらった呼び方をされている。Bollinger と McGowan は「実

際のところ段階と言えるものではなく、論理的には F、つまり落第点と同じこと」と言っている¹⁾。開発工程が管理できる状態にならないうちは、開発工程の改善活動はまともに進まない。

② 再現可能段階：委託、費用、日程、変更を厳格なプロジェクト管理の対象とする。これによって再現性のある管理段階となる。開発部門は安定した開発工程に到達している。

③ 定義段階：開発部門は開発工程を定義している。これによって、システムの導入は一貫性があるものになり、開発工程が理解しやすくなる。この段階までくると、先進技術を導入するときのリスクが大幅に軽減される。

④ 管理段階：開発部門は開発工程に関する総合的な測定と分析の活動を開始している。この段階から、最も重要な品質改善が始まる。

⑤ 最適化段階：開発部門は開発工程を継続的に改善し最適化していくための基盤を備えている。

段階ごとに重要な要件を示す

初期段階を除く 4 段階には、ソフトウェア開発工程をその段階に引き上げるために開発部門がどこに力を入れるべきかを特定するための KPA (Key Process Area: カギとなる開発工程の要件) が示されている (KPA はそれぞれの段階の成熟度を達成するのに必要な要件なので、当然ながら初期段階を達成するための KPA は定義されていない)。

開発部門は KPA で定義された活動をすべて実行すれば、開発工程の能力を高めるうえで重要な目標を達

きを加速させた。だが恐怖と混乱ももたらした。CMMの利用については、すなわち以前より少ない費用でソフトウェア製品を改善できることを実証するということである。批判もある。モデルが適切かどうか疑問がある、TQM (Total Quality Management) 技法の要件が欠けている、評価チームが民間企業の担当領域にまで口を出すようになる、などである。とはいいいながらも、多くの開発業者はこの政府機関の方針に従おうとしている。

われわれは30年以上にわたってソフトウェア開発に携わってきた。契約する際にソフトウェア工学をうんぬんし、特にCMMに従っているかどうかを重視するようになったことは、ソフトウェア開発工程を改善しようという考え方が浸透してきたことを如実に示すものである。この取り組みが大きな成果をあげるのか、いまは目立っているが結局しりすばみになってしまうのか、時間がたたなければわからない。

しかし、政府機関がソフトウェア

請負産業を変革できる立場にあることに疑問の余地はない。そして、さきほど引用した書簡の精神と意図を翻さないかぎり、おそらく変革する気になっているのだ。

開発工程の成熟度を5段階で示すCMM

Software Process Maturity (ソフトウェア開発工程成熟) は、SEIが開発した発展段階モデルだ。高品質のソフトウェア製品を矛盾なく日程通りに作り出すために必要なソフトウェア開発部門の能力を、定量化することが目的である。

歴史的にみると、ソフトウェアに関する研究開発は、OSや新しい言語などの製品や、トランザクション処理などの技術に重点を置いていた。今日では社会生活のあらゆる面でソフトウェアが幅を利かせている。だが、ソフトウェアの開発方法は一般にきわめてお粗末であるというのが「専門家」の一致した見解である。ソフトウェア工学の登場によって、

これまで芸術や職人芸とされていた領域に工学的な原理や統制の考え方を持ち込んできてはいるが。

DoDの合同研究が発端

米国政府、もっと明確にいえばDoDは、常にソフトウェアを大量に購入してきた。そして、お粗末なソフトウェア、日程遅れや高い費用という問題と格闘し続けてきた。DoDが結んだ17件の主要なソフトウェアの契約について非公開調査をしたところ、平均28カ月の開発期間が20カ月も延びていた。4年間の開発プロジェクトが7年たっても完了しなかった例もあった。当初の日程通りにいったプロジェクトは1件もなかった¹⁾。

DoDは1982年に、ソフトウェアに関する問題を分析するために、合同の特別研究班を設置した。計画にはSEIの設立と、有名なAdaプログラムの開発が含まれていた。しかし、近年ではソフトウェアの規模と複雑さが1桁はねあがっている。このため、開発工程を根本的に変更しないかぎり、現状のソフトウェア開発技法をよりよいものにするには不可能になっている。

SEIが1991年にCMMを開発

SEIは1984年に設立された。ソフトウェアに関する問題を改善したいというDoDの要求に対応するためである。SEIが集めたデータによって、米国内のソフトウェア開発部門のほとんどは共通の開発モデルをもっていないか、利用していないことが明らかになった²⁾。

SEIは成果物として、DoDとソフ

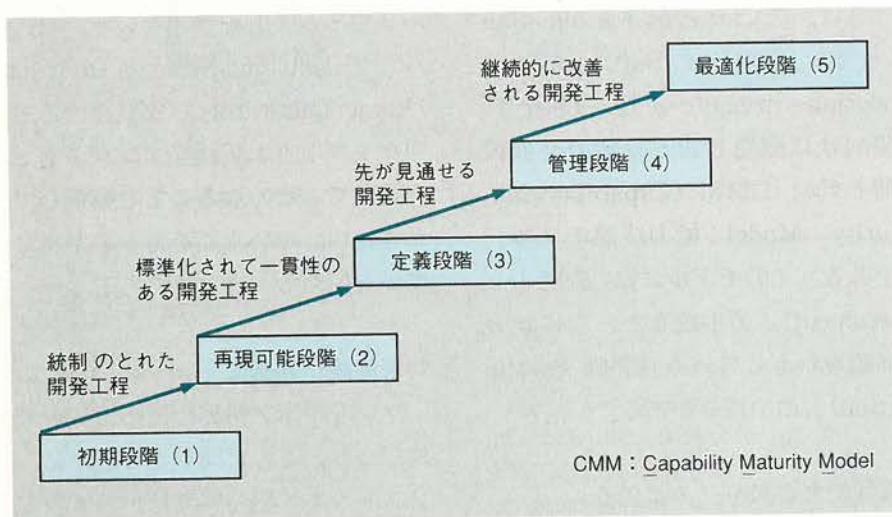


図1 CMMで規定された5段階の成熟度レベル(発展段階)

成できる。すべてのKPAは、プロジェクト単位で取り組む項目と開発部門として取り組む項目の両方からなる。本来プロジェクトは多数のKPAに取り組むべきである。表1に各段階のKPAを示す。個々のKPAはさらに①目標、②承認、③能力、④活動、⑤測定と分析、⑥実施後の検証、に細分されている（さらに詳細はPaulkほかの文献³⁾を参照されたい）。

これらの内容は、①目標を除いて、個々のKPAに適用できるように詳細に定義されている。こうした定義を使って、特定の契約あるいはプロジェクトが評価基準に適合しているかどうかを判定する。判定するのは、実務レベルの担当者として、契約に基づく活動を取り仕切れる第一線の管理

表1 それぞれの成熟度レベルでカギとなる開発工程の要件 (KPA)

レベル5	最適化段階	欠陥の予防
		技術の変更を管理
		開発工程の変更を管理
レベル4	管理段階	定量的な開発工程管理
		ソフトウェア品質管理
レベル3	定義段階	開発部門として開発工程を重視
		開発部門として開発工程を定義
		訓練計画
		統合されたソフトウェア管理
		ソフトウェア製品開発
		グループ間の調整
レベル2	再現可能段階	内部レビュー
		要件管理
		ソフトウェア開発プロジェクトの立案
		ソフトウェア開発プロジェクトの記録と監視
		ソフトウェアの部分委託の管理
		ソフトウェア品質保証
		ソフトウェア構成管理

KPA : Key Process Area

職である。基準を満たしていると考えられる事項については、それを実証する「確たる証拠」を用意しなければならない。

基準を満たすには証拠が必要

基本的に「確たる証拠」とは、何をどうするかを原理を示す文章と、その原理が矛盾なく実行されていることを示す証拠のことである。特定の企業あるいは契約のCMMレベルを判定するときは、評価チームは確たる証拠を示さなくてはならない。ある人には確たる証拠と見えても、ほかの人にはそう見えないこともある。

たとえば、あるプロジェクトではプログラムのノートを使って、コードの検査をどのように実施すべきかを厳密に指定するかもしれない。だが、これは確たる証拠としては不十分である。個々のコード・モジュールの検査報告書には、コードの検査ステップごとに検査あるいは検証したブロックを明記すべきであり、その検査ステップを実施した日付と検証担当者のサインを記すべきである。

こうした手続きによって、個々のKPAで定義されている事項のうち、その時点で完了している項目と完了していない項目がはっきりする。完了している項目については、確たる証拠を見つけて集める。これは開発工程の次の改善段階につながり、開発工程の欠陥に対処するための活動計画を作るのに役立つ。

開発部門の成熟度を査定

ソフトウェア開発工程の査定は、開発部門主導で始める。ソフトウェ

ア開発業務の改善を支援することが目的である。査定をするときは一般に、開発部門から上級のソフトウェア開発のプロフェッショナルを6~8人と、SEIあるいはSEIがライセンスしている査定業者から指導者を1~2人集める。査定は通常、次のような6段階で進める¹⁾。

① 選択：査定の対象とする開発部門の候補を特定する。査定対象として選定された開発部門は役員レベルの簡潔な状況説明をする。

② 承認：上級役員が査定契約にサインする。この時点で、開発部門は査定の手続きを完全に受け入れる。

③ 準備：開発部門の査定チームが訓練を受ける。現場での査定手続きの計画を十分に練る。査定の参加者をすべて集めて顔合わせをし、説明をする。この時点で、成熟度を判定するための質問票に記入する。

④ 査定：約1週間にわたり現場で査定をする。その後で査定チームが集まって勧告書を作る。

⑤ 報告：査定チーム全員で最終報告書を作る。査定参加者と上級管理職に対して、その内容を発表する。報告書には、査定チームが気付いたことと、改善活動のための勧告を盛り込む。

⑥ フォロー・アップ：査定対象となった開発部門の査定チームは、査定を実施した業者の指導のもとに活動計画を立てる。開発部門は約18カ月後に再度査定を実施して進歩の度合いを評価し、ソフトウェア開発工程の改善サイクルを持続させるべきである。

査定を実施するために開発部門は多くの資源を投入する必要がある。

上級管理職が真剣に取り組まなければうまくいかない。査定を実施する部門は、査定の方法とその結果を社外秘の情報として扱う。査定対象となった開発部門が査定情報を管理し、公開するかどうかを決める。

入札業者の成熟度を評価

(上述の)自発的で秘密の査定手続きとは対照的に、SCE (Software Capability Evaluation: ソフトウェア能力評価) は通常、政府機関やソフトウェア請負業者などの外部の業者が実施する。入札業者が使用する管理手法や開発手法を、購買部門が理解するのを助けるようにできている。契約を取りたい入札業者は SCE の手続きに参加する必要がある。

SCE の候補となる開発業者は、最初に成熟度に関する質問票に回答する。評価チームは開発業者を訪れ、成熟度に関する質問票を使って詳細な調査をするのに適した代表的な業務を選び出す。この調査では一般に、開発業者の人員に面接をし、その開発業者のもっているソフトウェア開発に関する文書を調べる。その時点で開発業者がプロジェクトに使用している開発工程を調査することにより、評価チームはレビューの対象とすべき記録を明らかにして、潜在的なリスクのある領域を素早く特定できる。潜在的なリスクがある領域¹⁾には次のような類似点がある。

▷提示された開発工程は購買部門の要求を満たす。

▷開発業者は提示された開発工程を実際に導入する。

▷開発業者は提示された開発工程を効果的に実施する。

評価される側の負担も大きい

SCE は高度な判断が必要な手続きであり、1 件の契約に関係するすべての開発業者が一貫した評価を受ける必要がある。SCE の手法によって、必要とされる一貫した評価基準と手法を提供できるものと SEI は考えている。

評価には、かなり多くの時間、出張、その他の資源を必要とする。購買部門とその評価チームは何週間もかけて準備し評価を実施する。評価を受ける開発業者も、適切な評価の結果興味をもたれたら、かなりの影響を受けるだろう。開発業者はまた、自社の強みを強調するのに多大な努力を費やす。開発業者の人員は予告のちに説明を受ける。文書をまとめ、ファイルを使えるように用意する。

評価のあいだじゅう、面接を受ける人員には精神的にかなり重圧がかかる。正直に、しかし言葉を選んで返事をすることになる。不正確な返事をしてしまったとき、そのせいで契約を失うのではないかと感じるものもある。開発業者は意図的に評価チームを欺こうとしているわけではない。できるだけ好印象を与えるように努めているのである。

欠陥もある

CMM が前提としている考え方は、ソフトウェア開発に重大な問題が起きたりソフトウェア開発プロジェクトが失敗する原因は技術的なものではなく管理に問題があるからだ、というものである。CMM では、開発工程管理と品質改善の技法をソフトウェアの開発保守に適用する。組織的な開発工程改善のモデルである。

しかし、CMM は網羅的なモデルでもなければ、「銀の弾丸」でもない。プロジェクトを成功させるうえで重要なソフトウェアの管理や開発の実践手法のいくつかを、CMM は扱っていない。たとえば、CMM はいまのところ特定のアプリケーション領域の専門的知識を直接には扱わない。特定のツール、手法、ソフトウェア技術を推奨しない。(有能な人材を選別したり、雇ったり、動機付けしたり、引きとめるといった)人的資源に関する問題も扱わない。コンカレント・エンジニアリングやチームワークや変更管理やシステム工学に関連した問題も扱わない。CMM バージョン 1.1 の作成者たちは Paulk らの文献²⁾で、このような欠陥を認めている。

事務処理分野にそぐわない点も

CMM 以外にも発展段階モデルは存在する。注目すべき例としては、Capers Jones のモデルがある。人によっては CMM と肩を並べるか、それ以上であると評価している。事務処理分野で広く使われることを目的としている。ISO 9001 はソフトウェア・システムの設計、開発、導入、サービスのための品質保証ガイドラインを定めている。

CMM のルーツは政府機関のシステムと国防関連のソフトウェア分野にある。このため、事務処理分野には当てはまらないようなことが前提となっていたりする。そこで米 Digital Equipment Corp. などの企業は、CMM を拡張して自社なりの開発工程の改善活動に適用できるモデルを作成した。

的であり、より多くの問題をカバーするので、開発工程を改善するには有用である、と言う業界代表者も多数いる。

評価と査定は別物

こうした議論は、SCEとSPAが認めなければならない共通の思い違いを明らかにしてくれる。実際のところ、SCEとSPAのランク付けはいつも同一である必要はない。SCEは、契約を遂行するのに必要な特定の作業を実行するための開発業者の能力を評価するものだ。これに対してSPAは、開発部門の一般的な成熟度を査定するものである。もしも開発業者が通常の業務範囲を外れるような契約案件に入札するのであれば、SCEの結果がSPAの結果と違っていても問題はない。

表2(詳細はPaulk³⁾の文献を参照)にまとめたように、SPAとSCEは動機付け、目的、結果の所有者が異なっている。この違いは、収集する情報と定式化する結果の違いにつながってくる。たとえば、SPAの作業がオープンで協力的な環境で実施されるのに対して、SCEはより監査指向の強い環境で実施されるし、

評価チームの推薦が請負業者の選定に反映されるので目的が金銭面の考慮と結びついている⁹⁾。

Pyzdek⁹⁾はSEIを相手に、どこかしら厳格なSEIのCMMに関して議論をしかけている。この記事でPyzdekは、ソフトウェアの品質を改善するための「正しい」方法はないと主張している。すべての開発部門は自分自身のアプローチを考え出さねばならない、とも言っている。このことはつまり、CMMはかなり良いけれど、そこで与えられたアプローチはどれをとっても正しいとは言えないということを意味している。さらに、外部から持ち込まれた品質改善策は、その定義からして解とはなりえないと述べている。CMMは明らかに外部からのものである。

これに対して反論もある。CMMのレベルを達成するための「方法」は開発部門が選択できるものであり、外部から規定されるのは「基準」だけである、という意見である。

TQMとの関係があいまい

さらに大きな混乱を招いているのは、CMMとTQM(Total Quality Management)の関係である。図2

はCMMとTQMの関係を示している。この図によると、TQMの原理は開発部門のすべてのプロジェクトに影響を与える。だが、CMMが影響を与えるのはソフトウェア開発のプロジェクトだけだ。Silver¹⁰⁾によると、CMMには次のようないくつかの欠点がある。

▷TQMと開発工程の文化的な側面を無視している。

▷開発工程の基盤と作業の範囲を混同している。

▷品質保証と開発工程グループを制度化している。

▷統計的な開発工程制御をうまく実施できない。

▷有益な開発工程の改善活動を遅延させる。

▷すべてのKPA活動が並行して独立して連続して改善されていくことを説明していない。

▷開発工程の能力を定量的に測定する基準を用意していない。

▷ソフトウェアのサポートを無視している。

Silver¹⁰⁾は欠点ごとに詳説している。CMMの大きな欠点は、TQMを認識できていないことだと指摘している。この批判は、TQMは品質管理の基本として機能すべきであるという信念から発生している。

別の論文の結論部分では「SEIは開発工程の査定をするために、真に優れた手法を開発した。だが、査定とSCEの手法は両方とも、証明されていないSEIの開発工程成熟モデルに依存しているという重大な欠点がある。さらに、開発部門の開発工程の成熟度指数を決定するSCEの手法は、統計的あるいは方法論的

表2 SPAとSCEの比較

SPA (ソフトウェア開発工程の査定)	SCE (ソフトウェア能力の評価)
<ul style="list-style-type: none"> ▷開発部門がソフトウェア開発工程を改善するために使用する ▷結果は開発部門だけに影響する ▷現在実践していることを査定する ▷開発工程を改善するための触媒となる ▷改善策と行動計画のもとになる情報を提供する ▷協力的: 開発部門のメンバーで査定チームを組織 ▷個人ではなく開発部門全体を査定する 	<ul style="list-style-type: none"> ▷購買部門が発注業者を選定し契約について監視するために使用する ▷結果は購買部門と発注した開発業者に影響する ▷現在実践していることを立証する ▷改善するという約束を判定する ▷契約業者としての潜在能力を分析する ▷第三者として評価: 開発業者のメンバーは評価チームに参加しない ▷個別の契約に適した能力があるか評価する

SCE : Software Capability Evaluation SPA : Software Process Assessments

にみると疑問だらけである。このようなランク付けが、高品質のソフトウェアを日程通りに予算範囲内で生産する開発部門の実力と、意味ある相関関係をもつものなのか疑わしい」としている⁴⁾。

質問票に対する批判

Bollinger と McGowan も、さまざまなソフトウェア開発工程の問題に関して作られた、101 項目からなるイエス/ノー式の開発工程成熟度の質問票を批判している。成熟度の各レベルを達成するのに必要なイエスの最少数がそれぞれ決められていて、されにそれを文書で証明する必要がある。質問票に反映されるのは収集する情報の 10~20% にすぎない。残りの情報を集めるには、組織的に面接をしたり、別の手段を使う必要がある。

ほとんどの企業（査定したうちの約 80%）は、現在レベル 1 の段階にある。結果的にレベル 1 は、非常に広い範囲の開発部門を含むことになった。ソフトウェアの開発などまったくできない開発部門から、優秀だがギリギリの利益しか出せない開発部門まで含んでいる。

レベル 2 の重要な質問に回答できないために、レベル 1 に甘んじている企業もある。上位レベルに到達するのに必要な質問のうち、かなりの数あるいはすべての質問に肯定的に答えられるかもしれないのだが、それは意味がない。レベル 2 に上がるのに必要な 12 個の重要な質問に 1 個でも失敗したら、その開発部門はほかの質問にいかにもうまく回答できたとしてもテスト全体としては失敗

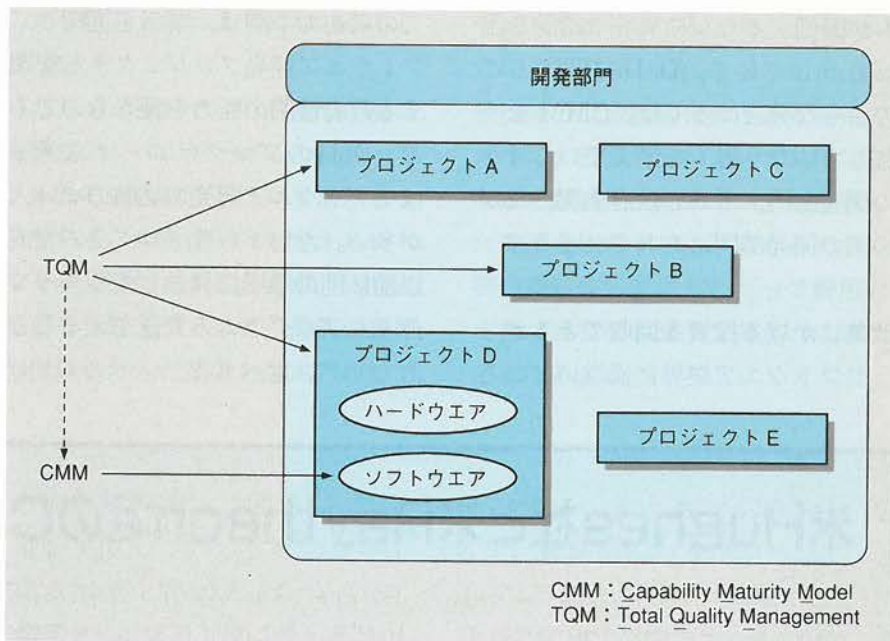


図2 CMMとTQMの関係

となる。したがって、このランク付けはたいへん恣意的であるとみられている。

「CMMが標準になるのは必至」

Humphrey と Curtis¹¹⁾は、Bollinger と McGowan が指摘したいくつかの欠点について反論している⁴⁾。この反論には興味深い文章が登場する。「実際のところ、SEI は SCE の監査人に対して、成熟度をもとにしてソフトウェア・ベンダの契約裁定の推薦をしないように指示している」というのだ。これは政府機関が計画している契約裁定手続きと矛盾している。

しかし本稿の最初で述べたように、ある政府機関は入札業者の条件をレベル 3 に設定しようとしている。この意図がどうであれ、賢明なソフトウェア・ベンダは、入札業者の選択に参加するにはあるレベルに達していることが必須条件あるいは事実

上の基準となることを覚悟しなければならないだろう。

この推測は、米 Signal 誌(空軍の Communications and Electronics Association(通信電子協会)の正式な機関誌)で裏付けられている。同誌は、次のように述べている。「取材の結果得られた一般的なコンセンサスは、SEI のモデルは国防機関が推進しているので、いずれはほかのモデルを差し置いて開発工程の標準となるだろう¹²⁾。記事に出てくるこのコンセンサスやそのほかの情報は、さまざまな規模の民間企業、防衛関連企業、政府関連企業に対する取材から得たものだ。

これらの取材によって、評価の手続きに対する業界の不満も引き出している。評価チームによる評価結果は、契約ごとによって一貫性がない。開発工程の成熟度は評価するたびに異なる。同時に複数のチームが異なる結果を出したり、同じチー

ムが評価するたびに異なる結論を出したりしている (DoD に関するこのような発言の多くは、CMM を実施していたり導入を考えているほかの開発部門、ほかの政府機関、ほかの国の開発部門にもあてはまる)。

改善にける投資を回収できるか ソフトウェア業界に渦巻いている

このほかの不満は、業者を選択しソフトウェア開発プロジェクトを管理する政府機関の能力を疑うものである。DoD のプログラム・マネージャはソフトウェア開発の経験のない人が多い。たいていはプログラム完成以前に別の職場に異動してしまっ、作ったプログラムの責任をとることもない¹²⁾。レベル 2、レベル 3、レ

ベル 5 といった開発部門が、「レベル 0」の政府官僚に管理されている場合もある。しかし、DoD も CMM についてノウハウを蓄積してきているので、改善される日も近いという楽観的な意見もある。

CMM と SCE に関するこれ以外の不満は、評価チームが所属する企業もほかの企業と同様に、政府機関

米 Hughes 社と米 Raytheon 社の CMM 導入事例

ソフトウェアの世界は、いよいよソフトウェア開発工程とこれを改善する最も効率のよい手法を正式に定義すべき段階に到達した。これまで、新しい発展段階モデルがいくつか提案されてきた。だが、定量的に評価されなかったため、それらの効果を正確に査定するのは難しかった。

SEI の CMM も例外ではなかった。実際に CMM の能力を評価し、ソフトウェア開発の品質と予測性を向上させ、費用対効果と投資対収益のバランスを改善することについての、定量的な研究はあまりなされなかった。

以下に示すのは 2 件の事例研究の要約である。米 Hughes 社と米 Raytheon 社が、CMM を利用したときの具体的な統計数字を示してくれている。

年間 200 万ドルの収益

Hughes Aircraft 社は、Software Engineering Division の CMM レベルを 2 年計画でレベル

2 からレベル 3 に引き上げた。1987 年から 1990 年にかけて約 40 万ドル (75 人月) を費やした。開発部門のオーバヘッドは 2% 増えた。この費用は次の 6 項目に配分した^{A-1)}。

- ▷ 開発工程グループのリーダー (8%)
- ▷ 開発工程の定義 (6%)
- ▷ 技術開発 (28%)
- ▷ 定量的な開発工程管理 (41%)
- ▷ 訓練 (16%)
- ▷ レビューと開発工程の標準化 (1%)

Hughes 社はこの投資に対して、年間 200 万ドルの初期収益があると計算した。費用対効果指標 (予算/実績費用) を (0.94% から 0.97% に) 50% 削減できることによる^{A-1), A-2)}。この投資の事業価値は 4.5 対 1 だった^{A-2)}。Hughes 社の費用対効果指標は 1992 年のあいだじゅう、改善し続けた。0.97% から 1.02% に上がった。開発プロジェクト全体が予算内に収まるまでになった^{A-2)}。

Hughes 社は、このように経費を節約できたのは、新しく追加した開発工程で欠陥を早期に検出でき、再作業の費用を実質的に削減できたためであるとしている。Herbsleb ら^{A-2)}によれば、再作業にかかる費用の削減を開発工程別に見ると、ほぼ次のようになる^{A-2)}。

- ▷ 初期設計で 44%
- ▷ 詳細設計で 96%
- ▷ コーディングで 83%
- ▷ 単体テストで 60%
- ▷ 統合テストで 58%

5 年間で 1580 万ドルを節減

Raytheon 社の数字はさらにすばらしい。同社は開発工程の改善に年間約 100 万ドルを投資した。その結果 7.7 対 1 (58 万ドルの投資に対して 448 万ドルの収入) の投資回収と、2 対 1 の生産性向上 (図 A-1) を得た^{A-3)}。

資金の割当方法はニーズに応じて変化させた。1992 年はどのように割り当てた。方針と手続きに 18%、訓練に 23%、ツールと手

のソフトウェア開発プロジェクト(評価対象のプロジェクトとはまったく関係のないプロジェクトであることを願うしかない)に入札する業者になり得ることである。したがって評価を受ける企業は、競合するかもしれない企業からやってくる評価チームが機密事項の約束を覆すのではないかと心配して、評価チームを

あまり歓迎しないようになるかもしれない。

だれもが抱えている共通の不満は、成熟度のレベルを上げるのに数十万ドルあるいは数百万ドルもかかることである。政府機関は指定するレベルを必須要件とする。だが、同時に重要な判断基準として、最低の費用を提示した入札業者を選ぶ。ここで、

ジレンマに陥ってしまう。さらに、この予算削減の時代にあっては、政府機関の財政援助はほとんど期待できない。投資対収益はいずれ黒字になるとはいえ、効率の向上が投資した費用を相殺するようになるまで、開発業者は何年にもわたって費用の支払いと利益の減少を背負い込むことになる。

法に30%、工程管理データベースに29%。人員はパート・タイム中心であり、年間合計で15人。1~2人のフルタイムの人員を置いた。Raytheon社は1988~1992年の間に記録をとった15件の開発プロジェクトで、再作業(11%~41%)の費用1580万ドルを浮かしたと言っている(図A-2)。

導入はトップ・ダウンで

これら2社の事例研究は、CMMバージョン1.1ではわざと規定しなかった開発工程改善のタイプを明らかにするものである。個々のビジネス環境に合わせて変更されているものの、2社の開発工程改善計画はCMMに基づいた類似性を示している。

CMMというモデルは、上級管理職の協力が必要になる。このため、導入しようとする企業はトップ・ダウン・アプローチを採用して、一般にSEPG(Software Engineering Process Group)と呼ばれる組織を設置する。SEPGが開発工程の改善計画を立て、調整し、結果を記録する。SEPGが方針と

手続きを標準化し、開発工程の改善を実践している各種のTWG(Technical Working Group)を監督する。開発工程に関するデータを分析するために、全社規模の

集中型データベースを用意する^{A-1)}。資金割当にも反映されているように、ここでとりあげた2社の開発部門は三つの主要分野に焦点を絞っている。



図A-1 米Raytheon社が分析したCMMの導入効果 18件の開発プロジェクトを対象に調べた。開発生産性の平均値が2倍以上に向上している。1人月当たり開発したソース・コードの行数と同等の指標で計測した。図の出典はIEEE Software。

迷いはあるが普及に向かう

CMMの影響について何社かと議論した。しかし、詳細な資料を提出しなかった企業も多い。情報を提供してくれた企業の利益を守るために、会社名そのほかの詳細がわかる部分を削除した。さらに付け加えれ

ば、われわれは各企業のよく知っているマネージャ1人だけと話をしたので、結論は必ずしも各企業の公式見解を反映していない。

ソフトウェア開発業者が本気でソフトウェア開発を改善するためにCMMを使っているのか、たんに政府機関と契約するための競争力をつけるためだけにCMMを使っている

のか、を判定するのは難しい。前者は製品の品質が活動を支える原動力となり、後者は近々発生する可能性があるビジネスを考慮することが原動力となる。同一の企業でも、両方の動機付けが同時に有効となり得る。

CMM重視の流れは避けられない
政府機関との契約で、入札時に特

定量的な管理が必要

Hughes社は複数のプロジェクトにまたがる統一的なデータ定義を標準化した。これを、見積もり費用、実績費用、誤り、日程の進行状況を記録するのに使用した。情報を編集して、上級管理職向けの月報を作った。これにはプロジェクトの成果、問題点、プログラムの障害報告、品質指標、対象範

囲の変更、資源に対する要求、学んだ教訓を掲載した。同時に提示するものとしては、時間経過に沿って実績値と計画値をプロットした図、プロジェクトの日程、マイルストーン、レート図、獲得した価値、財政/労働状況、対象システムにおける資源の使用状況がある^{A-1)}。

同様に、Raytheon社が測定基準を扱うために設置したTWGは

「開発工程のシステム面での問題領域」を特定するために、Mitre社の管理基準 (Management Metric)を採用した。「提案書の作成、季単位のレビュー、ソフトウェア開発力の評価、レベル4の成熟度を達成するのに必要な予想モデルなど、特定の研究を支援するPDC (Process Data Center)を設けた」^{A-3)}。TWGは標準化したスプレッドシートのテンプレートを提供して、個々のプロジェクトのメンバが測定基準に準拠したデータを収集しやすくしている。

技術担当を設置

Hughes社の技術運営委員会は、技術管理にかかわる業務と手続きを明確に定め、技術移転担当と呼ぶ職務を設定した。技術移転担当者は、とくに開発工程の成熟度を監視し、「各プロジェクトで使用している技術のデータベースを維持し、各プロジェクトが必要としている技術はなにかを把握し」、技術開発、開発工程の成熟度、訓練に関する全社レベルのさまざまな計画に参加した。

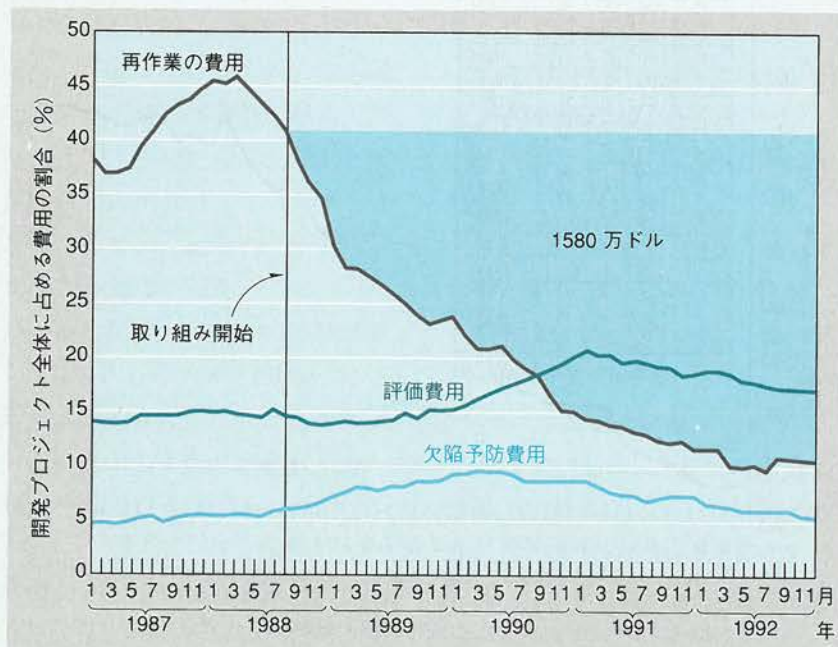


図 A-2 再作業にかかる費用を大幅に削減 Raytheon社は15件の開発プロジェクトで再作業の費用を合計1580万ドル削減した。図の出典はIEEE Software。

定のCMMレベルが必要条件となるのか、そしてそれはいつからなのか。この点についてははっきりしない。特定のCMM適合レベルを必要条件とするのはCMMが開発されたときの原則に反していて、適合しているかどうかは入札業者を選定する際の基準の一つとして使えるにすぎない、という意見がある。しか

し、この仮定に頼ってしまうのは危険である。成熟度レベルが非常に比重の大きい基準になるかもしれないからだ。

入札業者の選定に関与する人たちは、CMMレベルが高いことは「リスク」をやわらげる能力が高いことを意味するとみなしている。入札業者を選定する際に、入札業者のラン

ク付けに影響する重要な要因となる可能性もある。

いろいろな企業の報告によれば、高いCMMレベルを達成するような開発工程は非常に複雑である。しかし、ほとんどの企業はレベル1である。そこで、ここではレベル1からレベル2に昇格した企業が経験したことについて議論していく。

Raytheon社もよく似ている。ツール、環境の評価、開発工程の自動化に焦点を絞った、ツールと手法の行動部隊を設置した。この計画では、代替品となるCASE製品の評価、購入を正当化するために使用する費用対効果の分析、開発者に現状を教え込む訓練、個々のツールを統合してシームレスな開発環境を提供すること、特定のプロジェクトに合わせた調整、いろいろなタイプのユーザに対してマニュアルを生成することに、資金を提供した^{A-3)}。

訓練を重視

Hughes社とRaytheon社はともに、訓練に力点を置いた。Hughes社は訓練を昇進要件ではなく、職務要件としていることを強調している。SEIの勧告と従業員調査に対応するため、Hughes社はプログラミングの実践、プログラミング言語、CASEツールだけだった講習科目に、プロジェクト管理、レビュー、要件記述、要件レベルの検査、単体レベルの検査、品質保証の科目を追加した^{A-1)}。

Hughes社はSoftware Engineering Division以外の開発部門に対して、これらの講習を開放した。Software Engineering Division以外の参加者は20%という、まずまずの数字を報告している。Hughes社はこの強化訓練計画がレベル3を達成した「首尾一貫した開発部門の文化」に大きく貢献したと信じている^{A-1)}。

Raytheon社は勤務時間中に訓練するという総合的な訓練計画に資金を出している(1992年に564コース)。概要を示すコースを設定して、技術分野あるいは管理分野に関する一般的知識を提供している。これは定期的に開催している。詳細コースは特定のプロジェクトに合わせて、適宜実施している。Hughes社と同様に、Raytheon社は「訓練の対象領域に関して、はっきりと文化的シフトが見られる」と報告している^{A-3)}。

上記(ほかのところでも)報告されている投資回収性からすれば、CMMの導入は元を取っていると見える。だが、失敗した結果よりもうまくいった結果を報告したが

る傾向があることも事実だ。CMMの効果をもっと徹底的に理解するには、定量的、統計的、そして比較論的に結果を報告する必要がある。それには、もっとたくさんのプロジェクト(たとえば50プロジェクトとか)が必要である。CMMの発展過程において、この種の研究をするにはまだ時期尚早なのかもしれない。

(Hossein Saiedian=

米 University of Nebraska,

Scott Hamilton=

米 Computer 誌 Staff)

参考文献

- A-1) Humphries, W., Snyder, T. R., and Willis, R. R., "Software Process Improvement at Hughes Aircraft," *IEEE Software*, vol.8, no.4, pp.11-23, July 1991.
- A-2) Herbsleb, J. ほか, "Benefits of CMM-Based Software Process Improvement: Initial Results," *Tech. Report SEI-94-TR-13*, Software Engineering Institute, Carnegie Mellon University, Aug. 1994.
- A-3) Dion, R., "Process Improvement and the Corporate Balance Sheet," *IEEE Software*, vol.10, no.4, pp.28-35, July 1993.

開発工程の改善は、本質的に成熟度がどのレベルであるかは独立している。だから、ここで示すような経験から得られる考察は、ほかのレベルに上がっていく際にも適用可能であるはずだ。しかし、われわれはレベルを上げていく際に、相対的にみてどの程度の努力が必要か、なにか一つわかっていない。このことを強調しておきたい。レベル2からレベル3, 4, 5を達成する際に、必要になる努力がどれほど大きくなるものなのか。レベル2を達成するのに必要な努力の量をもとに、関数で示してくれるようなガイドラインは存在しない。

いきなりレベル2に挑む企業も

会議に次ぐ会議を重ねることは、CMMのレベルを上げるための最初の条件のように思える。われわれが取材した企業は、CMMの原理を導入する作業の準備段階にあった。担当者は(迅速であってほしい)行動と結果が企画や会議を置き換えてくれることを期待していた。しかし、われわれが調査した企業は行動や導入の段階を経験していなかったため、進捗を把握するための会議はまだ続きそうだ。

CMMのソフトウェア開発原理は、よくあるソフトウェア開発原理と似ていて、初期の分析と設計を重視する。導入した企業の報告によると、CMMに基づく開発工程は、初期分析と設計を重くみて工程を分けると成功する可能性が高くなるという。

大きな新規契約を獲得しようとしている企業は、この契約に必須というわけではないのに、CMMをレベ

ル2の原理から始めようとしている。将来は特定のCMMレベルが必要条件となるプロジェクトが出てきて、査定や評価を受けることが求められると考えているからだ。CMMで基礎を固めたうえで新規契約を結べば、企業全体の姿勢を改善できるだろうとも考えている。並行して努力すれば、現在進んでいるいくつかの重要な契約を確認しながら、レベル2に適合させていくことができる。

ソフトウェア開発業者が自己査定をする場合は、利害関係の衝突が起きる可能性が高くなりそうだ。しかし、正直者は報われるものである。自己査定の後に、外部機関による公平な評価(SCE)が実施されることが多い。査定に関係する人員(現場の人員と直接の管理職)は、技術的かつ実際的な問題に話題が集中しがちだ。結果的に、製品の品質を向上させるというCMM本来の目標は、近い将来の契約裁定に向けて取り繕っておきたいという目先の誘惑と、利害が一致する。

事例を見る限り経費削減は確実

CMMを適用したソフトウェア開発の改善事例で公開されているものを調べると、かなりの経費削減または収益増が見込めることがわかる。ソフトウェアの検査と保守にかかる費用を削減できるからだ。ソフトウェアは照合(verification)と確認(validation)がしやすいことによる。

CMMに適合するためにかかる費用について分析したのを見ると、どの企業も①CMMのレベルを上げるためにかけた一時的な費用と、

②高くなったレベルでソフトウェアを開発するために継続的に必要になる費用とを区別していた。②の費用を、CMMレベルが低いときにかかったソフトウェア開発費用と比較すると、実際に経費削減が見込めるかもしれない。ある事例では、CMMレベルが高くなったおかげでソフトウェアの開発費用をかなり削減できた。このため、高いレベルを目指すためにかけた一時的費用をすぐに取り戻せた。

報告によると、余分な努力と費用がかかることを従業員と経営者の両方が納得してはじめて、開発工程の改善活動がうまく機能するようになる。やり方はいろいろあるが、経営陣が食事を用意して昼休みに会議や訓練をするのも一つの方法である。この変形として、従業員と経営者が「時間を共有する」ものがある。この場合は、訓練にかかる時間の負担を両方で折半する。

CMMの経験は売り物になる

CMMを導入することによって、ソフトウェア開発会社の経営者あるいは企業自身が恩恵を受けることについては多くのことが語られている。だが、その従業員が恩恵を受けることは、ほとんど触れられていない。身につけた技法は、役に立つ専門的な技能となる。従業員は高いCMMレベルで働くほど、コンピュータ業界にとって貴重な人材になる。この種の経験は非常に市場性が高い。さらに従業員の誇りや管理者の尊敬といった面も、従業員の利益、報酬、動機付けの点で見逃さない。

われわれが企業に質問した範囲で

は、顧客の評判は基本的に製品の品質と、顧客と業者の関係によるという点で、各社の見解は一致している。CMMのレベルを高くすればソフトウェアの品質は向上し、企業の評判も高まる。この点では、ほとんど異論はなかった。

しかし、CMMに適合することによって、ソフトウェア開発業者と顧客の関係が変化するかもしれない。たとえば、CMMのレベルが高くなると形式主義的になり、気まぐれな顧客の要求にいちいち対応することが難しくなるだろう。だがCMMのレベルを高くすることは、請負業者と顧客の関係をより信頼度が高く、相互に利益のあるものにするのにも役立つだろう。

幸運なことに、最も引き込まれる議論は同時に非常に単純なものでもある。すなわち、安い費用で高品質のソフトウェアを作り企業の評判を高めることは、契約を勝ち取って維持するための非常に有力な手段となる。

肯定的な事例が圧倒的に多い

だれもがあらゆる面で納得できる改善推進策は存在しない。結局のところCMMは、特にTQMの原理と組み合わせたときに非常にうまくいくモデルであるようだ。

いまひとつはつきりしないのは、ソフトウェア開発費用の削減とソフトウェア開発作業がより効率的になったことによって、CMMのレベルを達成し維持する費用を回収できるかどうかである。公開されている研究報告（一部は本稿でも引用している）では、CMMに基づく開発工程の改善は元を取る以上のものがある

という。

しかし、否定的な結果よりも肯定的な結果のほうが圧倒的に多いという程度に考えておくほうが安全である。開発工程の改善で失敗したことを進んで公開しようという企業は少ないからだ。それでも、DoDが強力に支援を続けるので、ソフトウェア開発工程の改善努力をCMMに基づいて実施しようとする企業の数は今後も増えていくだろう。

謝辞

Computer誌の匿名の調停者から得た意見と助言がたいへん役に立った。Software Productivity Consortiumのsenior technical staffであるRichard Bechtoldの訂正と助言が、本稿全体の内容と独自性を十分に高めてくれた。また改訂版をレビューしてくれたLaurie Werthにも感謝する。彼女は優れた教育用のモジュールを制作していて、表2はそこから採った。作家のMark Paulkは親切にも多くの質問に答え、本稿のレビューに時間を割いてくれた。

参考文献

- 1) Humphrey, W. S., "Introduction to Software Process Improvement," *Tech. Report CMU/SEI-92-TR-7*, Software Engineering Institute, Carnegie Mellon University, June 1993.
- 2) Krasner, H. ほか, "Lessons Learned from a Software Process Modeling System," *Communication ACM*, vol.35, no.9, pp.91-111, Sep. 1992.
- 3) Paulk, M. ほか, "Capability Maturity Model for Software, Version 1.1," *Tech. Report CMU/SEI-93-TR-24*, Software Engineering Institute, Carnegie Mellon University, Feb. 1993.
- 4) Bollinger, T. B. and McGowan, C., "A Critical Look at Software Capability Evaluations," *IEEE Software*, vol.8, no.4, pp.25-41, July 1991.
- 5) Humphrey, W. S., Snyder, T. R., and Willis, R. R., "Software Process Improvement at Hughes Aircraft," *IEEE Software*, vol.8, no.4, pp.11-23, July 1991.
- 6) Dion, R., "Elements of a Process Improvement Program," *IEEE Software*, vol.9, no.4, pp.83-85, July 1992.
- 7) Lipke, W. H., and Butler, K. L., "Software Process Improvement: A Success Story," *CrossTalk*, pp.29-39, Nov. 1992.
- 8) Card, D., "Capability Evaluations Rated Highly Variable," *IEEE Software*, vol.9, no.5, pp.105-106, Sep. 1992.
- 9) Pyzdek, T., "To Improve Your Process: Keep it Simple," *IEEE Software*, vol.9, no.5, pp.112-113, Sep. 1992.
- 10) Silver, B., "TQM VS. the SEI Capability Maturity Model," *Software Quality World*, vol.4, no.2, Dec. 1992.
- 11) Humphrey, W. S., and Curtis, B., "Comments on 'A Critical Look,'" *IEEE Software*, vol.8, no.4, pp.42-46, July 1991.
- 12) Johnson, D. L., and Brodman, J., "Software Process Rigors Yield Stress, Efficiency," *Signal*, vol.46, no.12, pp.55-57, Aug. 1992. ●

略語一覧(ABC順)

CASE: computer aided software engineering

CMM: Capability Maturity Model

ISO: International Organization for Standardization

KPA: Key Process Area

OS: operating system

SCE: Software Capability Evaluation

SPA: Software Process Assessments

TQM: Total Quality Management