

ソフトウェア開発プロセスの評価と改善

Assessment and Improvement for Software Development Processes

小笠原 秀人
OGASAWARA Hideto

中野 一男
NAKANO Kazuo

田中 史朗
TANAKA Fumiaki

効率的、効果的なソフトウェアの開発を行うためには、適切な開発プロセスとそのプロセスを継続的に改善する仕組みが必要である。開発プロセスをあるべき姿に近づけるためには、対象部門の開発プロセスの適切な評価と、改善のための手法や技術の適切な利用がたいせつである。

当社では、ソフトウェアプロセス改善技術を積極的に活用し、ソフトウェアの効率的な開発と品質向上を推進している。

An appropriate development process and continuous improvement of the process are necessary for the development of high-quality software with high efficiency. To establish an ideal development process, assessment of the existing development departments and the application of methods and techniques for process improvement are important.

Software process improvement technology is actively utilized for quality and productivity improvement in Toshiba's software development departments.

ソフトウェアプロセス改善の効果的な進め方

Effective Promotion Method for Software Process Improvement

1 まえがき

より良いソフトウェアの効率的な開発には、適切なソフトウェアプロセスが必要となる。どのような作業をどのような手順で行うか、あるいはどのような組織形態でどのように役割を分担するか、などを定義して実践する活動はソフトウェアプロセスと呼ばれる。近年はソフトウェアの規模や複雑さが増大して開発そのものも非常に難しくなり、様々な問題が顕在化している。これらの問題を解決し、高品質ソフトウェアを効率的に開発するためには、より良いソフトウェアプロセスが不可欠であり、ソフトウェアプロセスを改善し最適化して、より良い製品を作り出していこうとする活動が重視されてきている。

ここでは、ソフトウェアプロセスの評価と改善を中心に、当社の取組みを、事例を交えて述べる。

2 ソフトウェアプロセスに関する問題

ソフトウェアプロセス改善の基本は、現状の姿(AS-IS)とあるべき姿(TO-BE)のギャップを正確に把握し、適切な施策を実行することである、と言われている。ソフトウェアプロセスでは、ソフトウェア開発に携わる人や組織、作業など

が複雑に関係するため、改善策の実行に際して以下に示すような問題が起きやすく、障害となるケースが少なくない。

- (1) 理想的なプロセス像の評価が難しい ソフトウェアプロセスは、製品分野やそれぞれの開発部門によって様々な形をとる。更に、技術的な難しさや厳しい納期などの制約が重なることも多い。各開発部門では、将来あるべきプロセス像を描き、改善活動を推進しているが、そのあるべき姿の十分性を評価することが難しい。これは、現状プロセスの課題を明確にするための手法や方法論が必ずしも十分に整備されていないこととも関連する。
- (2) 課題解決のための改善策の具体化実行が難しい
現状の課題解決のために、どのような施策や改善策が有効であるかを検討し、具体化して実行する場合、人や組織などの問題から、必ずしも円滑に進まない。また、あるべき姿が適切に設定できていない場合には、改善効果が十分に得られないことが多い。

3 ソフトウェアプロセスに関する当社の取組み

3.1 ソフトウェアプロセスの評価・改善のための方法論
ソフトウェアプロセスの最適化については、米国カーネギーメロン大学ソフトウェア工学研究所(CMU/SEI: Carnegie Mellon University / Software Engineering Institute)で開発されたIDEAL手法が著名である。IDEAL手法は、I(Initiating: 開始フェーズ) D(Diagnosing: 診断フェーズ)、E(Establishing: 確立フェーズ)、A(Acting: 実践フェーズ)、

L (Learning: 学習フェーズ) の五つのフェーズから構成され、これら五つのフェーズをサイクリックに実行することでプロセスの改善を推進し、プロセスの最適化を実現する。

当社ではこのIDEALの考え方を更に発展させ、社内で実践してきたプロセス改善の事例や教訓を組み込み、ソフトウェアプロセスを最適化する手法を開発した。

ソフトウェアプロセス改善活動は、ロングスパンの活動であり、その目的や活動の手順などがあいまいになりがちである。当社で開発したソフトウェアプロセスの最適化手法では、次のような情報を明確にして、より効率的なプロセス改善の実施を可能としている。

- (1) IDEALの各フェーズの活動目的を明文化
- (2) IDEALのフェーズごとの開始・完了基準を設定
- (3) 各フェーズの活動手順を抽象化して整理し、活動テンプレートとして提供
- (4) 過去のプロセス改善における具体的な活動事例やノウハウをデータベース化して提供

当社では、これらを“IDEALガイド～プロセス改善活動の手引き～”としてガイドの形で整理し、プロセス改善活動を実行する際に利用している。

3.2 ソフトウェアプロセスの評価と改善手法

IDEALサイクルにおけるDの作業では、現状のプロセスがどのような状態にあるかがあるべき姿と対比して評価する。ソフトウェアプロセスの評価手法としては、CMU/SEIによって開発されたSW-CMM(SoftWare-Capability Maturity Model)をベストプラクティスとして、これと比較して組織の能力成熟度を評価する手法が広く認知されている。

3.2.1 プロセスの評価基準と改善ロードマップ

一般にSW-CMMは、レベル取得や認証を目的とした活用が耳目を集めている。しかし、当社はSW-CMMをレベル認証の道具としてではなく、ソフトウェア開発組織内のプロセス改善の道具として利用している。つまり、このSW-CMMで規定された各レベルのソフトウェアプロセス像をソフトウェアプロセスのあるべき姿としてとらえ、これらと現状プロセスの相違を評価して、改善策を立案し実行している。また、改善活動のロードマップ(数年後のあるべき姿を示す)としても活用している。SW-CMMの各レベルで要求されているプロセス領域を図1に示す。

CMU/SEIからは、公式アセスメントの結果が定期的に報告されている。2001年8月に公開された報告書では、1997年から2001年6月までの間に約1,000の組織が公式アセスメントを実施し、そのうちの約70%がレベル1とレベル2であった。また、レベルを一つ上げるためには、平均で18～30か月掛かることが報告されている。この結果は、ソフトウェア開発における開発管理の難しさと、改善活動にはある程度の時間が掛かることを示している⁽¹⁾。

3.2.2 ソフトウェアプロセス評価手法 当社では、開

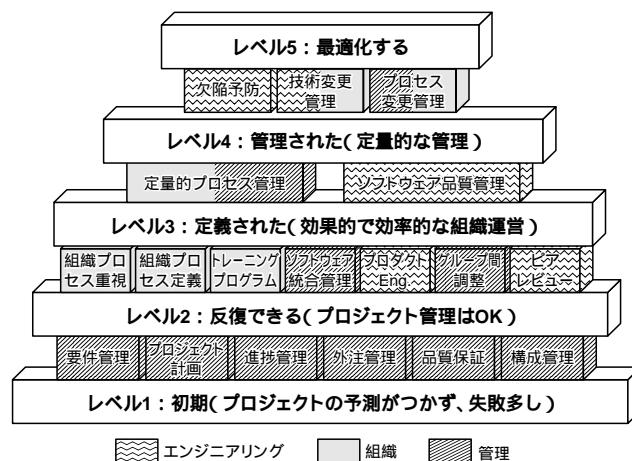


図1 . SW-CMMの能力成熟度と各レベルのプロセス領域 ソフトウェアプロセス成熟度を評価するための参照モデルであり、ソフトウェア組織が場当たりの・混沌(こんとん)的なプロセスから成熟した規律あるソフトウェアプロセスへと進化するためのロードマップを示したモデルでもある。

Capability maturity and process areas of Software-Capability Maturity Model (SW-CMM)

発部門の状況や改善の進み具合に応じて利用できるように、複数のソフトウェアプロセスの評価手法を提供している。CMU/SEIの認定を受けた社内のリードアセッサによる公式アセスメント(CBA-IP: CMM Based Appraisal - Internal Process Improvement)は、現状のソフトウェアプロセスをより正確に把握し、課題を明確にするために活用している。また、改善活動の節目で、目標とする成熟度レベルに達しているかどうかの判断にも利用する。

改善活動を行いたいのだが、どこから着手すべきか悩んでいるような部門に対しては、SW-CMMミニアセスメント手法やアンケートツールを利用して現状の評価を行い、課題を明確化している。更に、その課題に対する改善計画を提供し、改善活動が加速できるように支援している。

3.2.3 ソフトウェアプロセス改善手法 IDEALサイクルにおけるEとAの作業では、Dの作業結果に基づいて改善計画を立案し、具体的な改善策をたて実践する。

改善活動の効率的・効果的な推進のために、SW-CMMのプロセス領域の改善に役立つ技法やツールを提供している。規模の見積りと成果物の品質確保のための技法として、ファンクションポイント法やピアレビュー / インスペクションを活用している。また、ドキュメントやプログラムの構成管理を確実に実施するための構成管理ツールや、プログラムの品質を保証するためのプログラム静的解析ツールなどを組織に定着させるための支援も行っている。

3.3 プロセス改善に対する組織的取組み

ソフトウェアプロセスの評価・改善では、これらの活動を推進するためのグループの組織化やメンバーに対する役割

と責任の付与が重要である。当社におけるプロセス改善でも、IDEALのIの作業で、SEPG(Software Engineering Process Group : プロセス改善を推進するグループ)を組織内に作り、戦略的にプロセス改善に取り組んでいる。また、プロセス改善を実践するには、SEPGリーダーの役割が非常に大きいため、SEPGリーダー育成のためのトレーニングコースの充実も図っている。

4 あとがき

ソフトウェア開発のプロセス改善では、組織としての成熟度の継続的な成長が求められる。ソフトウェアエンジニアリングの分野でこれまで研究されてきたプロセス改善技術を積極的に活用し、プロセス改善のための手法の確立と推進に努めていきたい。(小笠原)

CMMに基づくプロセス改善

Software Process Improvement Activities Based on CMM

1 まえがき

ソフトウェアシステムの大規模化や複雑化が加速しており、ソフトウェアプロセス自身も複雑になる傾向にある。また、これらの開発は比較的大きなプロジェクトで進められることが多く、開発管理プロセスの重要度が相対的に増してきている。

開発管理プロセスは、ソフトウェア開発を円滑に進めるために開発アイテムごとの進捗(しんちよく)状況を把握したり、システム構成を確認し、品質保証を行うといった作業プロセスであり、SW-CMMのレベル2で達成すべきプロセスとして位置づけられている。開発管理プロセスに問題があると、作業の手戻りが増え、開発コストの増大を招く。

ここでは、大規模システム開発における、SW-CMMレベル2に示された開発管理プロセスの改善事例について述べる。

2 大規模プロジェクト管理に関する課題

ビジネス電話のシステムは、現代のビジネスニーズに合致させるための最新技術を積極的に導入しており、ますます大規模・複雑化している。従来から要件管理や進捗管理などの開発管理を行っていたが、大規模・複雑化するに従って、次のような問題が明らかになってきた。

(1) ユーザー要求の仕様変更の制御が難しく、各開発グ

ープ間で要求仕様に対する取扱いに不整合が生ずる。

(2) 何に基づいて進捗管理をしているのかが不明確であり、開発状況が見えづらい。

(3) プロジェクトで規定した手順を徹底することが難しい。

これらの課題を解決し、最適な開発管理プロセスを実現するため、SW-CMMを改善のロードマップとして活用し、プロセス改善を実施した。

3 プロセス改善の手順

プロセス改善は、IDEALの考え方に沿って以下の手順で実施した。

3.1 開始(Initiating)フェーズ

プロセス改善の開始フェーズでは推進体制としてSEPGとSQAG(Software Quality Assurance Group)の組織化を行った。SEPGは標準プロセスの作成と改善に責任を持つ。また、SQAGはSW-CMMに基づいて定義されたプロセスの実施状況を検証するという役割を持つ。

3.2 診断(Diagnosing)フェーズ

アンケートシステムを利用して、組織内の各階層に対するアンケートを定期的実施し、現行の開発プロセスを評価した。この結果に基づき、SEPGを中心として既存のソフトウェアプロセスとSW-CMMレベル2の要求事項の差異を分析し、実践できているところ、不足しているところを明確にした。ある時点での評価結果例を表1に示す。

表1. SW-CMMレベル2のプロセス領域の評価結果
Result of SW-CMM level 2 process area evaluation

項番	プロセス領域	顕在化した内容
1	要件管理	ユーザー要求の仕様変更の管理が不徹底
		仕様のレビュー不足
2	計画・進捗管理	計画の詳細化が不足
		進捗管理活動が各グループ間で不統一
		各グループ間で連携をとった計画と進捗管理活動の不足
3	外注管理	協力会社の行う設計、開発に関する計画の詳細化が不足
4	品質保証	プロジェクトで規定したプロセスを検証する仕組みが徹底できていない
		品質の定量評価の基準が不明確
5	構成管理	成果物の一貫性が確保しきれていない

3.3 確立(Establishing)フェーズ

確立フェーズでは、Dフェーズで得られたソフトウェアプロセスの強み、弱みを基に、実行可能性と現状の開発フェーズを考慮して改善の優先度付けを行い、具体的な改善計画を

立案した。

3.4 実践(Acting)フェーズ

定期的な診断結果に基づいて改善項目を明確化し、プロセス改善を推進した。具体的には、個々の問題点を解決するためのプロセス定義や、開発支援ツールの導入などを行ってきた。定期的に実施したSW-CMMのアンケート結果を図2に示す。継続的に改善を積み重ねることで、着実にSW-CMMレベル2の状態に近づいていった。

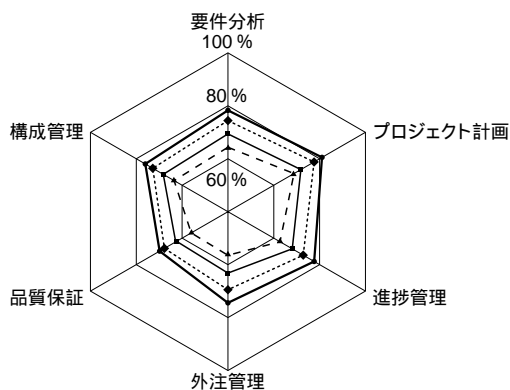


図2. 定期的なプロセスの診断結果 定期的に実施したSW-CMMのアンケート結果から、プロセス領域ごとの達成度を表している。内側の線から順に新しいアンケート結果になっている。
Result of periodic process assessment

そして組織内での改善活動が定着化してきた段階で、SW-CMMレベル2に適合する標準プロセスをSEPGで開発した。標準プロセスは実際のフィールドに適用しやすくするため、特に次のような点を配慮して開発した。

- (1) 標準プロセスにおける組織の理想形を実際の組織にマッピングする。
- (2) SW-CMMで使われている用語を、組織で使われている用語に統一する。
- (3) 各担当者の行動パターンや各プロセス領域の管理活動の状況を把握するための計測指標などを明確に定義する。

更に、開発した標準プロセスを実際の開発フィールドで適用するために、標準プロセスの教育を実施した。また、定義した標準プロセスが遵守されているかどうかを、定期的にSQAGがレビューした。

2000年9月にはCMU/SEI公認のリードアセッサによる公式のアセスメント(CBA-IPI)を実施した。アセスメントは、CMMレベル2のプロセスが満たされていること、及びCMMレベル3の領域における改善点の明確化を目的に実施した。その結果、CMMレベル2の状態に達していることを確認できた。

4 具体的な改善施策とその効果

今回のプロセス改善ではSW-CMMレベル2の管理プロセスの実現を目標に改善を進めた。この中で特に品質管理、進捗管理の各プロセス領域では次のような効果が得られた。

- (1) 品質管理 開発過程でのレビュー計画の立案と実施、また、その際の品質指標の設定やチェック項目の整備などを行った。この結果、定量的な品質管理が実践でき、品質上問題がありそうな機能を早めにチェックできるようになった。
- (2) 進捗管理 具体的な改善策として、大・中・小日程による日程管理方式を導入した。この方式では中日程は週単位で見直し、3か月に一度システム全体の見積りと実績を分析し、大日程計画へのフィードバックをシステムティックに行っている。これにより進捗管理の管理精度が向上し、適切な見積りと、タイミングのよい日程計画の見直しができるようになった。

5 あとがき

効率的なソフトウェア開発管理を実現するため、SW-CMMレベル2に示された開発管理プロセスをあるべき姿とした改善活動を紹介した。SW-CMMレベル2のプロセスを実装したことで、開発管理の基盤が強固になり、プロセスとプロダクトに関するデータがタイミングよく収集できるようになった。今後、データに基づく工程・品質管理を精度よく、タイムリーに実現できるようにしていきたい。(中野)

ピアレビューによるプロセス改善

Software Process Improvement Based on Peer Review Process

1 まえがき

ソフトウェアの設計や実装の誤りの早期除去は、品質向上に大きな効果がある。当社では、設計や実装段階での誤り除去を目的として、デザインレビューなどが行われている。ここではレビュープロセスを評価し、改善した事例について述べる。

2 レビュープロセスに関する課題

発電した電気を安定的に各需要家に送る電力系統監視制御システムを例にとると、その規模は年々大規模・複雑化しており、品質に関する課題が顕在化している。レビュープロ

セスを評価した結果、次のような課題があることがわかった。

- (1) マイルストーンでのチェックを中心としたレビュープロセスとなっており、設計過程での内容面のチェックが担当者のスキルに依存している場合が多い。
- (2) 開発量が変化しているのに、レビューに関する定量的な尺度が更新されず、効果的に利用できていない。また、レビューに関するデータを組織的に蓄積する仕組みが弱い。
- (3) ドキュメント形式などが統一されていない場合があり、レビューの質や効率を妨げている。

このプロセス評価結果を基に、レビュープロセスに関する改善施策の検討を行い、ピアレビューを取り入れることとした。

3 ピアレビュー手法の概要

ピアレビューは、SW-CMMにもレベル3のKPA(Key Process Area)の技術として要求されている。ピアレビューの目的は、早期に効率よくソフトウェア作業成果物から欠陥を取り除くことである。前項の課題解決を目標に、以下のような工夫を施したピアレビュー手法を導入した。

特徴 - 1 レビュー実施を特定のマイルストーンに限定せず、設計書などの中間作業成果物もレビュー対象とし、設計の内容面のチェックを重視する。

特徴 - 2 担当者のスキル依存を防止するため、対象システムが提供する機能ごとに、チェックのタイミングと内容を整理したチェックシートを利用する。

特徴 - 3 レビューの実施状況とレビュー対象成果物の品質を正確に把握するために、定量的な二つのメトリクス(指標)であるピアレビュー比率(各工程におけるピアレビューの実施工数)及びピアレビュー効率(ピアレビュー会議1時間あたりに検出される欠陥の数)を定義し、これらのデータを分析しデータベースに保持して活用する。

特徴 - 4 専用のピアレビューシートを利用して、出席者、対象成果物、現在の工程などの情報、指摘された欠陥などを記録する。

ピアレビューは対象作業成果物のレビューに適しているメンバー間で、1～2時間の会議を開き、欠陥の検出を目的として実施している。このため、問題の解決や、コンセンサスを得ることに時間を掛けないように、レビューリーダーが制御する方式を採用した。

4 ピアレビュー手法の導入

改善施策としてのピアレビュー手法を円滑に導入するため、SEPGを中心に次のような改善施策推進活動を行った。

4.1 ピアレビュー教育・コンサルティングの実施

管理者層及びソフトウェア開発担当者層のそれぞれ全員に対して、階層別にピアレビュー教育を実施した。また、SEPGメンバーが中心になってプロジェクトごとにピアレビュー実施のコンサルテーションを展開した。

4.2 ピアレビュー実施プロセスの規定

ピアレビューの実施手順と記録の取り方を規定化して、ピアレビューの実施手順とピアレビュー実績のデータにばらつきがでないようにした。

5 ピアレビューによるレビュープロセスの改善効果

5.1 総合試験工程における品質問題の減少

ピアレビュー実施前後における開発プロジェクト18件のエラー密度(指摘件数/k step)のばらつき具合を表2に示す。ここで、mは過去の実績データから算出した総合試験時のエラー密度の平均、σは標準偏差である。この平均値と標準偏差は、品質評価のための基準として活用されている。

表2. ピアレビュー実施前後の総合試験時のエラー密度
Comparison of error density in system test phase before and after peer review activities

期間	プロジェクト数	エラー密度		
		m以下	m~m+ の間	m+ 以上
実施前	13	6	3	4
実施後	5	4	0	1

表2から明らかなように、ピアレビュープロセス導入後は、総合試験時のエラー密度が減少していることがわかる。これは、上流工程からのピアレビューの実施によって、要求定義、基本設計で早めに欠陥を検出できたこと、及び技術的な課題がピアレビューで指摘されていたことが大きな要因である。

5.2 定量的尺度とデータ蓄積の仕組みとドキュメント統一

プロセスデータベースを利用して、ピアレビューの実施に関するデータを蓄積する仕組みを確立し、ピアレビュー実施に関する基準を設定した。これによって、レビュー状況の客観的な分析が可能となった(図3)。

例えば、図3(a)では、欠陥をどの工程で作り込んでいるか、(b)ではどの工程でピアレビューが効率よく実施できているか、(c)ではピアレビューごとの欠陥検出数の妥当性、といった内容を客観的に判断できる。また、ドキュメントの書式やスタイルなどについても、チェックが行き届き、統一することができた。

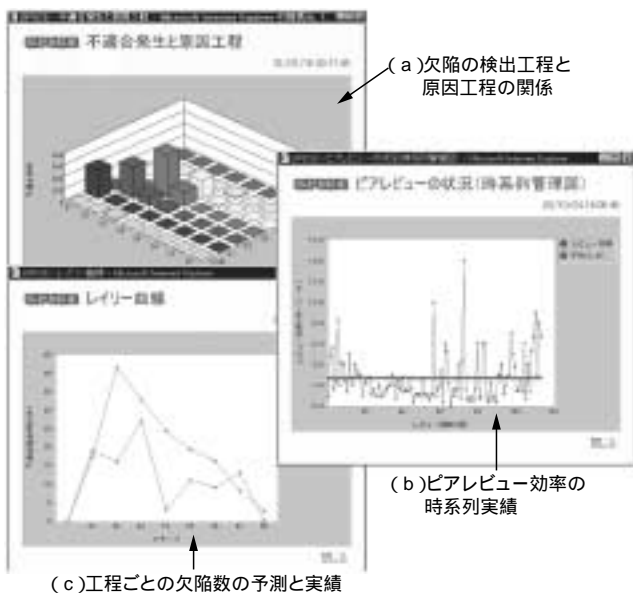


図3. 代表的なデータ分析結果の例 ピアレビューの実施にともなって収集・蓄積されたデータを、三つの観点から分析した例を示す。ピアレビューの進捗状況や対象成果物の品質レベル、欠陥の検出工程と原因工程の関係を確認できる。

Typical peer review data analysis graph

6 あとがき

レビュープロセスの改善に焦点を当てた活動を紹介した。従来から、レビューは品質向上に大きな影響があると言われていた。この活動を通して、品質に与えるインパクトの大きさを実感できた。また、従来からのレビューに関する問題を解決することができた。

今後、ピアレビューの実施記録データを中心とした可視化の仕組みを活用して、定量的な工程フォローや品質確認を実施するための活動を推進していきたい。(田中)

文献

- (1) CMU/SEI . Process Maturity Profile of the Software Community 2001 Mid-Year Update . CMU/SEI ,2001.
(http://www.sei.cmu.edu/sema/pdf/2001_aug.pdf)
- (2) J.Herbsleb ,et al . Software Quality and the Capability Maturity Model . CACM ,40 ,6 ,1997 ,p.30 - 40.
- (3) H.Ogasawara, et al . Process Improvement Activities by High Quality Software Creation Support Virtual Center . The Second World Congress for Software Quality ,2000.



小笠原 秀人 OGASAWARA Hideto

研究開発センター システム技術ラボラトリー研究主務。ソフトウェアのプロセス改善の研究・開発に従事。電子情報通信学会, 情報処理学会, ソフトウェア技術者協会会員。System Engineering Lab.



中野 一男 NAKANO Kazuo

e-ソリューション社 日野工場 ビジネスコミュニケーションシステム部主務。電話交換システムのソフトウェア設計に従事。Hino Operations



田中 史朗 TANAKA Fumiaki

電力システム社 府中電力システム工場 電力計算機システム部経営変革エキスパート。電力系統用計算機システムの設計・開発に従事。情報処理学会, 電気学会会員。Fuchu Operations - Power Systems & Services