

SEC BOOKS

プロセス改善ナビゲーションガイド

～なぜなに編～

独立行政法人 情報処理推進機構 ソフトウェア・エンジニアリング・センター 編

SEC BOOKS

プロセス改善ナビゲーションガイド

～なぜなに編～

独立行政法人 情報処理推進機構 ソフトウェア・エンジニアリング・センター 編

商標等の取り扱いについて

- ・ CMM[®], CMMI[®] and Capability Maturity Model[®] are registered with the U.S. Patent and Trademark Office.
- ・ CMM IntegrationSM, IDEALSM, PSPSM and TSPSM are service marks of Carnegie Mellon University.
- ・ SPICE は、伝統的に ISO/IEC 15504で Exemplar Model として取り上げられてきたモデル, その初期検討グループ, トライアルアセスメント実施グループ等を総括的に示す愛称です。
- ・ 上記にかかわらず, 本ガイドに掲載されているシステム名, 製品名等は, 一般にその開発元の商標または登録商標です。本ガイドでは, 本ガイドを製作する目的でのみそれら商品名, 団体名を記載しており, 編集者としては, その商標権を侵害する意思, 目的のないことを申し述べておきます。

本書を発行するにあたって, 内容に誤りのないようできる限りの注意を払いましたが, 本書の内容を適用した結果生じたこと, また, 適用できなかった結果について, 著者, 出版社とも一切の責任を負いませんのでご了承下さい。

本書は, 「著作権法」によって, 著作権等の権利が保護されている著作物です。本書の複製権・翻訳権・上映権・譲渡権・公衆送信権 (送信可能化権を含む) は著作権者が保有しています。本書の全部または一部につき, 無断で転載, 複写複製, 電子的装置への入力等をされると, 著作権等の権利侵害となる場合がありますので, ご注意ください。

本書の無断複写は, 著作権法上の制限事項を除き, 禁じられています。本書の複写複製を希望される場合は, そのつと事前に下記へ連絡して許諾を得てください。

(株)日本著作出版権管理システム (電話 03-3817-5670, FAX 03-3815-8199)

JCLS <(株)日本著作出版権管理システム委託出版物>

はじめに

『プロセス改善ナビゲーションガイド（なぜなに編）』（以下本書といいます）は、そもそもプロセス改善とは何を指すのか、どのようなアプローチがあるのか、プロセス改善の基本的な概念を整理してみました。

なお本書では、特に断り書きのない場合には、プロセスを主としてソフトウェアプロセスのこととして解説しています。システムという言葉は、「ソフトウェアを中心としたシステム」の意味で用いています。より一般的なプロセスという言葉は、ビジネスプロセスやプラントのプロセスなどをさすことありもますが、ここではそれらを対象としていません。

『プロセス改善ナビゲーションガイド』は3編でシリーズを構成していますが、それぞれ独立して利用することも可能です。

別冊『虎の巻編』（本書発行時点執筆中）では、プロセス改善を推進するうえで抱く実務的な疑問に答える形で、実務者向けのノウハウをガイドブックとして提供しています。

また別冊『プロセス診断活用編』は、モデルベースのプロセス改善を推進しようとしたときに道具として有効に利用できるプロセスアセスメントモデルの活用法について、国際規格ISO/IEC 15504の考え方をベースにまとめています。

本書の想定する読者はおおよそ次のような方々です。業務の分野や事業の規模などは問いません。

- プロセス改善とは何かを初歩から学びたい方
- 従来からプロセス改善に取り組んでいるが、新しい時代に即してその考え方を整理したい方
- 現場の状況を生かしながらプロセス改善に取り組もうと考える改善推進担当者
- 自社の競争力確保のためにプロセス改善の方向性やあるべき姿を見定めたい経営者の方
- ソフトウェア調達の立場から、あるいはサプライチェーンを含めた総合的な観点からプロセス改善に問題意識のある方

2007年3月 著者らしるす

目次

はじめにI

第1章 なぜプロセス改善か 1

1.1 プロセス改善の意義と狙い1

■ プロセス改善が指すもの1

■ ソフトウェア組織における“プロセス”とは2

1.2 ソフトウェアプロセスの現状4

視点① 「社会」から見た

ソフトウェアプロセスの現状と問題点5

視点② 「ソフトウェアコミュニティ」から見た

ソフトウェアプロセスの現状と問題点6

視点③ 「ソフトウェア企業」から見た

ソフトウェアプロセスの現状と問題点8

視点④ 「ソフトウェアエンジニア」から見た

ソフトウェアプロセスの現状と問題点9

■ 本当は何が問題か9

1.3 仕組みを持つ組織と持たない組織の差11

■ 良い仕組みを持つ組織の特性とは…12

1.4 ソフトウェアプロセスを改善する狙い14

■ 良い結果を定常的にもたらすためには プロセスを良くすることである	14
コラム プロセスの詳細記述の仕方	17
■ ソフトウェアプロセス改善の狙いと視点	18
1.5 プロセス改善のアプローチ	20
■ 失敗を契機にしたプロセス改善の例	20
■ アセスメントモデルをベースにしたプロセス改善の例	22
1.6 プロセス改善の対象プロセスの発見	24
1.7 プロセス改善の効果	25
1.8 まとめ	28

第2章 プロセス改善の姿・形 29

2.1 改善サイクル	29
■ 改善サイクルをまわすのは誰？	32
■ プロセス改善サイクルの国際規格	34
2.2 プロセス管理の手法やツール	34
■ プロセスの現実を理解するツール	35
コラム プロセス理解の主体	36
■ プロセスを意図通りにするための手法	38

2.3	プロセス実施技術	44
■	実施支援ツール	45
■	実施技術・手法の実例	46
■	その他の観点	48
2.4	事業目標とプロセスのニーズ	51
■	開発組織に求められる本質的な事業目標	51
■	事業目標からプロセスに対する要求の筋道	52
■	取得者の業務プロセスとソフトウェアプロセス	53
2.5	サプライチェーンとソフトウェアプロセス	54
■	サプライチェーンの改善アプローチ	55
■	サプライチェーンのプロセスの課題とヒント	56
2.6	モデルベース改善とその他のアプローチ	57
■	モデルベース改善	57
■	その他のアプローチ	58
■	「スマートな改善」	60

第3章 プロセス改善の留意点と実施体制 63

3.1	プロセス改善推進の留意点	63
■	ソフトウェア開発・保守・運用を支える三要素	63
コラム	プロセスを実施する人と プロフェッショナルについて	65
■	プロセス改善のシステム	66

■コミュニケーションと知識活用	67
コラム 会議のもち方について	72
コラム 情報の見える化について	73
3.2 プロセス改善推進の知恵	74
コラム “5ゲン” 主義について	86
3.3 プロセス改善の推進体制	89
コラム 良いアセッサについて	94
3.4 プロセス改善事例の公開	96
3.5 プロセス改善プログラム	103
コラム プロセス改善プログラム (process improvement program) について	105
おわりに	107
付録：用語集	108
索引	110

第1章 なぜプロセス改善か

第1章では、プロセス改善とは何かについて説明します。そして、なぜプロセス改善が必要か、どのようにプロセス改善を進めるのか、プロセス改善によってどのようなメリットがあるのかについて説明します。

1.1 プロセス改善の意義と狙い

本書のテーマであるプロセス改善とは、“良い”と信じる方向に自分たちの仕事のやり方を変えていくことです。

ITとその応用技術を重要な柱とした現代社会において、プロフェッショナルなソフトウェア組織の中での「ソフトウェアプロセス改善」は、ひととき重要な意味を持つものとなります。

その中軸となるソフトウェア開発・保守・運用組織と技術者が、仕事のやり方・スタイルを見直し、“良い”と信じる方向に仕事のやり方を変えていくことで、洗練された業務・仕事の仕組みを形成し、信頼されるソフトウェア組織となること、信頼される職業人となることが求められているのです。

■ プロセス改善が目指すもの

プロセス改善が目指すのは、一般的には“QCD”、つまり品質 (Quality) とコスト (Cost)、そして納期 (Delivery) の3つについての予測精度・制御精度を上げ、さらに有効性を高めることです。

QCDはプロセス改善の実績と効果が上がるにつれて、定量的デー

タで管理・整理され、その精度は向上します。プロセス改善に取り組む中で、組織はひとり一人が常にプロセスを意識して活動することになり、決められた共通のプロセスを踏むことで、プロジェクトの各段階での進め方や深さなどがメンバー間で共有されるからです。

つまり、プロセス改善は、最終的には組織の事業目標の達成に寄与し、そのための有効な手段となるのです。

なお、ソフトウェア組織においてプロセス改善を持続し、効果を高めていくためには、その企業のプログラムの書き方、プロジェクトの進め方などを効率的に学べるトレーニング制度も重要となります。

いっぽう、システムやソフトウェアを発注する側においても、発注先のプロセス改善の成果を適正に診断し、理解したうえで業務発注をおこなうことが、ますます重要になってきています。

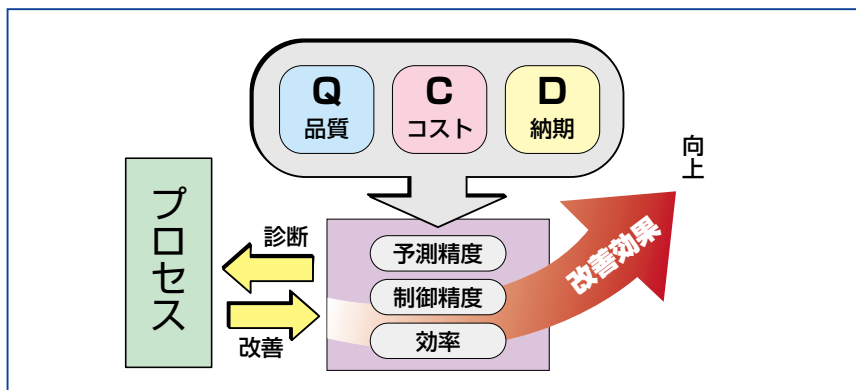


図1-1 プロセス改善の目的

■ ソフトウェア組織における“プロセス”とは

ソフトウェア開発において、プロセス改善をおこなううえで対象

になるのがソフトウェアプロセスです。

ソフトウェアプロセスは、「ある目的を達成するために決めたインプットからアウトプットまでの一連の活動」と考えることができます。具体的には、ソフトウェア要求分析プロセス、ソフトウェアテストプロセス、プロジェクト管理プロセスなどが代表的なものとして挙げられます。

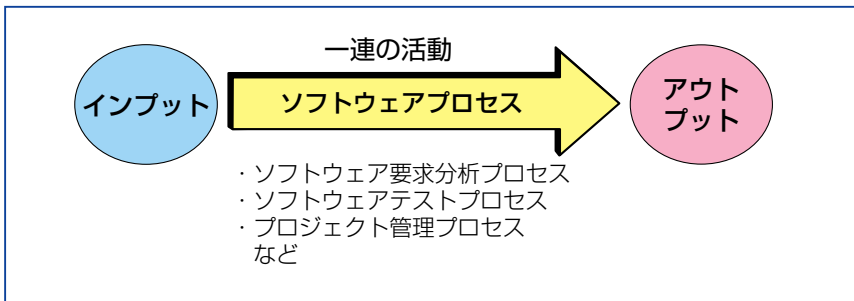


図1-2 ソフトウェアプロセスのインプット、アウトプット

参考までに表1-1に標準等での（ソフトウェア）プロセスの定義をいくつか示します。

ここで重要なのは、プロセスの目的と成果であって、これらを事業目標（QCDなど）と関連づけてプロセスを決めることです。

表1-1 プロセスの定義

規定・標準	プロセスの定義
ISO/IEC 15504-1 及び、ISO/IEC 12207	インプットをアウトプットに変換する、相互に関連する又は相互に作用する一連の活動（JIS Q9000）
IEEE	ある目的のために実行される一連のステップ
Quality Process Management (by Pall, Gabriel A.)	指定された最終結果を生み出すように設計された、作業活動に関わる「人、材料、エネルギー、機器、及び手順」の論理的編成

1.2 ソフトウェアプロセスの現状

ソフトウェアプロセスの改善を考える前に、ソフトウェアプロセスをとりまく現状と問題点を整理しておきます。

ソフトウェアの社会への影響力は増大の一途をたどっています。政府や地方自治体のシステム、交通システム、銀行システムなど、サービス機能のソフトウェア化が進み、人々の生活がシステムに依存する度合いもますます高まっています。また、そうした社会インフラを支えるバックボーンとしてだけでなく、自動車のエンジン制御、携帯電話、デジタルカメラなどの制御のように、身近な工業製品でもソフトウェアへの依存度が高まっています。

ソフトウェアは直接目に見えるものでなく、またシステムや機器が正常に動作しているときには気かけられません。いったんシステムダウンや誤作動などの障害が発生すると多大な影響が出ます。

銀行システム間の統合ミス、航空管制システムの欠陥、証券シス

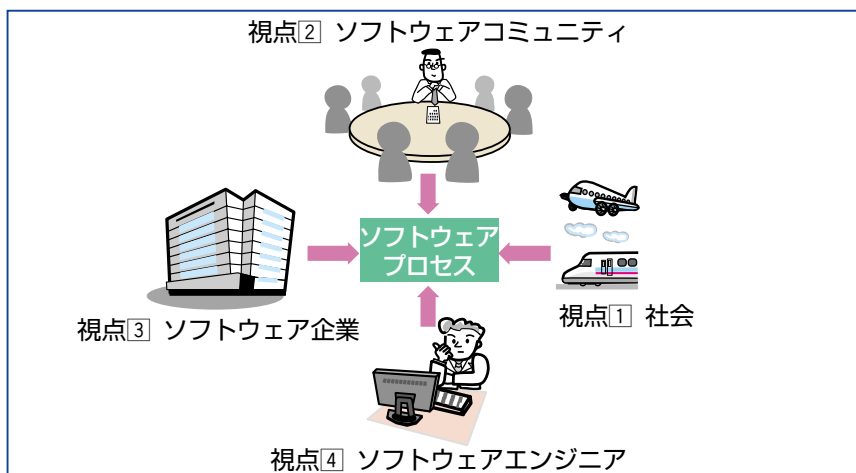


図1-3 ソフトウェアプロセスはどう見られているか

テムの欠陥、携帯電話のソフトの不具合、エレベータの制御プログラムの欠陥など、公表されるソフトウェアの問題は後を絶ちません。

私たちの生活に深くかかわっているソフトウェアですが、その仕事の仕方（ソフトウェアプロセス）について図1-3の4つの視点から現状を見て問題点を洗い出してみましょう。

視点① 「社会」から見た

ソフトウェアプロセスの現状と問題点

「ソフトウェアは工業製品である」といわれるようになり、エンジニアリングやマネジメントの改革に対する取り組みが数多くなされましたが、未だに初歩的なミスから永遠の課題といわれる要件定義に至るまで問題が山積しています。

たとえば、社会から見たソフトウェアプロセスの現状には次のような問題点があります。

- ソフトウェアにも工業製品と同等（以上）の品質が求められているが、実現できていない
- ソフトウェアの品質管理は有形物と同様にはいかない（難しい）
- プロセス改善をエンジニアリングの問題や組織の問題として扱っていない

このような問題が未解決ながら、ソフトウェアは既に生活するうえでの必需品となっています。社会基盤の一部であり、ガス・水道・電気のようにライフライン化しているともいえます。次のことを想像してみてください。ソフトウェアの障害により、銀行の

ATMが使えなくなった、制御システムが故障して電車がストップしてしまった、給与システムがダウンして給料が振り込まれなかった、電話会社のシステムにトラブルがあり携帯電話が使えなくなった、エレベータの制御プログラムに欠陥があり20階まで階段で昇り降りしなければならなくなった、など。工業製品では考えられないような品質管理や成果認識の現状がこのまま放置されれば、ソフトウェア産業の信頼は失墜し、発展の芽は摘み取られる危惧さえあります。



図1-4 「社会」から見たソフトウェアプロセスの現状

視点② 「ソフトウェアコミュニティ^{注1}」から見たソフトウェアプロセスの現状と問題点

今から20～30年前は、ソフトウェアに内在しているバグが時々トラブルを起こすのはある程度許容されていました。しかし現在では、ソフトウェアの社会における重要性や影響力が増大し、バグは許容できなくなってきています。

注1. ソフトウェアの業界団体や研究会などをいう。

現在は再利用や自動化により人手による開発比率が下がり、開発技術・管理技術が従来よりも格段に進歩しています。しかしソフトウェアコミュニティから見ると、未だに次のようなことが起きています。

- 要求する側が要求仕様を明確にできない、また要件の変更が多い
- ソフトウェアが原因でない変更も、ソフトウェア部隊が対応している
- 大規模化・複雑化・短納期化が進み、対応に窮している

これらの課題に対して多くのコミュニティが解決に向けて継続的に取り組んでいます。ソフトウェア・エンジニアリング・センター (IPA/SEC) でも産・学・官などと連携しながら、より適切な仕事の仕方ができ、品質の良いソフトウェアを開発できるように活動しています。

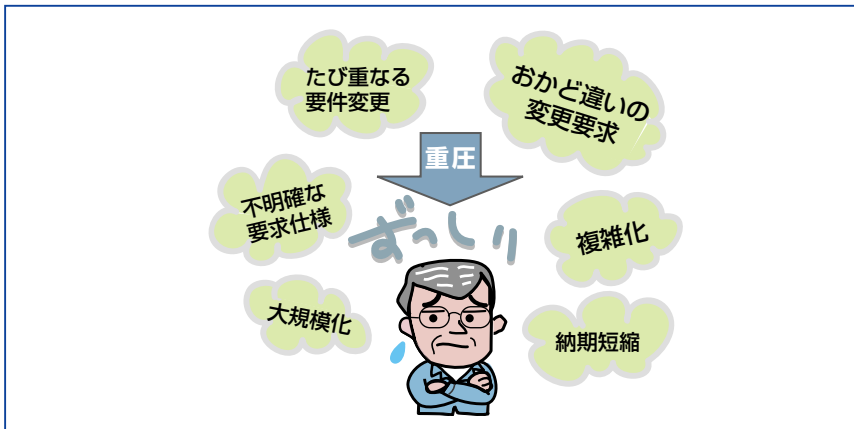


図1-5 「ソフトウェアコミュニティ」から見たソフトウェアプロセスの現状

視点③ 「ソフトウェア企業」から見た ソフトウェアプロセスの現状と問題点

ソフトウェア開発を事業としている企業から見ると、QCDを満足させるソフトウェアを納入することがビジネスの基本です。そのためには、明確な事業戦略を立て、それに則った人材戦略やアウトソーシング戦略を立てて実行します。ソフトウェアの世界では「ドッグイヤー」といわれるほど進歩や競争が激しく、ソフトウェア企業では、以下の2つの事項に焦点が当てられています。

- 人材確保や育成が基本である
- 最新のテクノロジーに対応して他社をリードしていきたい

アウトソーシングという点では、ソフトウェアを世界の各サイトで開発して統合したり、原価の安い国にアウトソース（オフショア開発）したり、他国の企業からアウトソースされたりする場合があります。これらの場合は相互のコミュニケーションが極めて重要であり、共通の開発・管理の基盤が不可欠となります。

特に市場規模の大きい欧米からアウトソースされることを狙って、インドや中国ではエンジニアリング面およびマネジメント面で高度に標準化、効率化を果たした企業が増大しています。

一方、日本では開発コストの安いオフショア開発を推進していますが、コミュニケーションの原点である言語の面で、日本語の特殊性から、要件や仕様の共通理解に課題があります。また、日本的ビジネスの文化や慣習との相違により共通の基盤が確立できず、十分に効果を上げていない状況にあります。

視点④ 「ソフトウェアエンジニア」から見た ソフトウェアプロセスの現状と問題点

実際にソフトウェアを開発している人々はどうかということ、QCDを達成する使命を与えられ、奮闘努力しています。残業や休日出勤も多く、ひとたびトラブルが発生すると休日や深夜でも呼び出されます。このような状況が起こらないように各自で工夫しているのが現状です。仕事の仕方が組織的に確立していないところからは次のような声が聞かれます。

- 開発の現場の実態はひどいものだが、自分の力量でカバーできている
- 現場の混乱を知らない人が口出ししないでほしい
- 仕事の仕方の改善が必要なら、ツールを活用するのが良い

このように人やツールに依存していて、その製品（プロダクト）の品質は重視しますが、仕事の仕方を組織やチームで良くすることは軽視している風潮が見られます。

■ 本当は何が問題か

図1-6はソフトウェアの開発における問題構造を表しています。「ソフトウェアは目に見えないから開発・管理が難しい」を裏付けた氷山の一角モデルです。出荷後に大きな障害が発生すると、

- ・ 誰が欠陥を作ったのか…（人）
- ・ どういう作り方や評価をしたのか…（方法論やツール）
- ・ 何でそのようなやり方をしたのか、もっと適切なやり方はなかったのか…（プロセス）

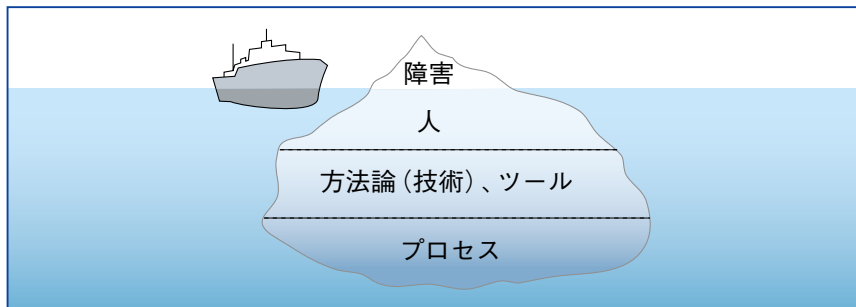


図1-6 ソフトウェアの開発における問題構造（氷山の一角モデル）

と、徐々に問題解析の核心に迫っていきます。そして「障害は現象であり、見かけ上の原因は人の失敗であるが、解決策はプロセスにある」となります。

なお、明らかに「人」が原因であるミスや、プロセスを遵守しないケースがあると思います。このような「人為的なミス」の場合でも、「プロセス」の改善によって、以後の発生を軽減させることができます。

ソフトウェアの開発・管理の課題を一気に解決できるような「銀の弾丸」^{注2}（特効薬）はなく、地道な取り組みが必要です。開発管理の課題に対して、「人」「方法論（技術）、ツール」「プロセス」に対するアプローチはそれぞれ必要であり、効果があります。高品質のソフトウェア製品を効率良く納期どおりに納めることが継続的にできるためには、組織的な能力改善が必要なのです。

氷山の一角モデルに示すような問題構造を把握して、障害を発生させない取り組みを始めましょう。

注2. 「銀の弾丸」とは狼男を唯一倒すことのできる弾丸のことで、転じてソフトウェアエンジニアリングの世界では万能の解決策の意味で使われている。ブルックス（Frederick P. Brooks）が1986年に発表した論文「No Silver Bullet（銀の弾丸などない）」が直接の語源。著書『人月の神話』（The Mythical Man-Month）の中でも引用している。

1.3 仕組みを持つ組織と持たない組織の差

企業には就業規則があり、さらに行動規範やセキュリティ上の役割、報告ルートなどが規定されるようになっていきます。ところがソフトウェア開発はクリエイティブな業務なため、仕事の仕方を個人の裁量に任せるケースを多く見受けられます。しかし、すべてを各人の裁量に委ねてしまってもいいものなのでしょうか。ソフトウェア開発・管理の仕組みは、ソフトウェアに関する仕事の仕方であり、すなわちソフトウェアプロセスを意味します。表1-2にソフトウェア開

表1-2 ソフトウェア開発・管理の仕組みを持たない組織と持つ組織の特徴

開発・管理の仕組みを持たない組織 (改善が必要な組織) の特徴	開発・管理の仕組みを持つ組織 (改善を行っている組織) の特徴
<ul style="list-style-type: none"> ・プロセスは場当たりのであり、プロジェクト進行中に実務者と管理者が必要に応じて作る。 	<ul style="list-style-type: none"> ・定義されたプロセスに関する役割と責務が明確である。 ・プロセスを全員に正確に伝え、作業を効率的に遂行する。
<ul style="list-style-type: none"> ・プロセスの規定があっても厳格に守られず、強制もされない。 ・スケジュール遅れになると、開発ではないレビューやテストなどを縮小または省略する。 ・厳しい納期を課せられると、十分な考慮や調整をせずに機能と品質を落として対応する。 	<ul style="list-style-type: none"> ・必須のプロセス（たとえば、レビューやテストなどの品質強化作業）を省かずに実行する。
<ul style="list-style-type: none"> ・管理者は危機へのその場しのぎに焦点をあてている。 ・プロセスを改善する余裕はなく、いつも事後対応（トラブル対応）をおこなっている。 	<ul style="list-style-type: none"> ・管理者は顧客満足度と成果物の品質をモニタする。 ・リスクが特定され、優先順位により、その低減、解消が実施されている。 ・問題が適切に対策され、再発防止まで実施されている。
<ul style="list-style-type: none"> ・プロジェクトの現状に対する理解は定性的で貧弱であり、成果物の品質も予測できない。 ・スケジュールと費用は現実的な予測に基づかず、慢性的に超過する傾向がある。 	<ul style="list-style-type: none"> ・品質判定とプロセスの問題分析のための定量的な基盤がある。 ・スケジュールと予算は実績に基づいており現実的である。
<ul style="list-style-type: none"> ・実績は、優秀な人がいるかに強く依存している。 	<ul style="list-style-type: none"> ・プロセスは定義され、適用結果の効果分析により、継続的な改善を実施している。

発・管理の仕組みを持たない組織（改善が必要な組織）と持つ組織（改善をおこなっている組織）の特徴を対比してみます。なお、表中では「ソフトウェアプロセス」を単に「プロセス」と表記しています。

仕組み（プロセス）がなくても必ずしもすべての組織が表の左側のように危機的な状況になるとは限りません。最後の項目のように「実績は、たまたま優秀な人がいるかに強く依存している。」ので、場合によっては良い結果を出してくれるかもしれません。しかし組織の永続性（存続性）を考えると、たまたまある特定の優れた人たちだけがプロジェクトを成功させても、それが組織の一部であったり、優秀な人がいずれそのプロジェクトを離れることになるため、組織としてのメリットは小さく、継続的な成功も期待できません。

開発・管理の仕組みの改善をおこなうことによって、表の右側のように組織全体で目に見える形でプロジェクトが運営でき、良い結果も継続的に期待できるようになります。

■ 良い仕組みを持つ組織の特性とは…

仕組みに従って作業を実施すると、ソフトウェア開発は結果のばらつきが小さくなってきます。しかも、同種の作業を繰り返しおこなうと、所要工数や誤りの度合いなどが安定し、それを定量的に表現することができるようになります。QCDに関して現状を定量的に把握するだけでなく、予測もできるようになってきます。

ソフトウェア（プロダクト）品質特性（ISO/IEC 25000シリーズ）^{注3}

注3. ISO/IEC 25000 SQuaRE - Guide to SQuaREはSQuaRE（Software product Quality Requirements and Evaluation）シリーズというソフトウェア製品の品質向上のための測定方法及び評価方法に関する規格体系である。なお、25000シリーズは、ISO/IEC 9126（Software Engineering？ Product Quality）およびISO/IEC 14598（Software Engineering - Product Evaluation）を発展させた次世代規格である。

表1-3 ソフトウェア品質特性の定義

品質特性	ISO/IEC 25000 (ISO/IEC 9126) の定義	品質副特性
機能性	ソフトウェアが、指定された条件の下で利用されるときに、明示的及び暗示的必要性に合致する機能を提供するソフトウェア製品の能力。	合目的性、正確性、相互運用性、セキュリティ、機能性標準適合性
信頼性	指定された条件の下で利用するとき、指定された達成水準を維持するソフトウェア製品の能力。	成熟性、障害許容性、回復性、信頼性標準適合性
使用性	指定された条件の下で利用するとき、理解、習得、利用でき、利用者にとって魅力的であるソフトウェア製品の能力。	理解性、習得性、運用性、魅力性、使用性標準適合性
効率性	明示的な条件の下で、使用する資源の量に対比して適切な性能を提供するソフトウェア製品の能力。	時間効率性、資源効率性、効率性標準適合性
保守性	修正のしやすさに関するソフトウェア製品の能力。修正は、是正若しくは向上、又は環境の変化、要求仕様の変更及び機能仕様の変更にソフトウェアを適応させることを含めてもよい。	解析性、変更性、安定性、試験性、保守性標準適合性
移植性	ある環境から他の環境へ移すためのソフトウェア製品の能力。	環境適応性、設置性、共存性、置換性、可搬性標準適合性

では、信頼性のほかに機能性、使用性、効率性、保守性、移植性など6つの品質特性が定義されています（表1-3）。開発するソフトウェア製品の利用者や社会性を考慮して、目標とする品質特性を決める際に活用できます。

すべての品質特性を同等に重要視して、全部を100%満足させることは困難です。また、仮に実現したとしても高コストになってしまいます。そこで開発する製品の種別や特徴を考慮して、各品質特性に対する優先度付けをして、品質とコストのバランスをとることが必要になります。

たとえば、銀行オンラインシステムなら、相対的に「信頼性」と処理速度（「効率性」）を重視します。また、ATMや窓口業務などで端末を操作するためのソフトウェアは、「使用性」が重要になります。なお、「機能性」はどのソフトウェアにも共通で、常に重要です。

膨大なサービス機能を保有し、銀行間の連携や統合が進んでいる現状では、銀行オンラインシステムは「保守性」や「移植性」も含めたすべての品質特性のバランスが重要であると考えられます。

1.4 ソフトウェアプロセスを改善する狙い

ソフトウェアの開発・管理の仕組み、すなわちソフトウェアプロセスの良し悪しがQCDに影響すると述べてきました。ここでは、もう少し具体的にソフトウェアプロセスを改善する狙いについて考えてみます。

■ 良い結果を定常的にもたらすためには プロセスを良くすることである

プロセス改善の前提となる考え方を図1-7に示します。

プロセスが良くなくても何とか乗り切って、結果オーライという

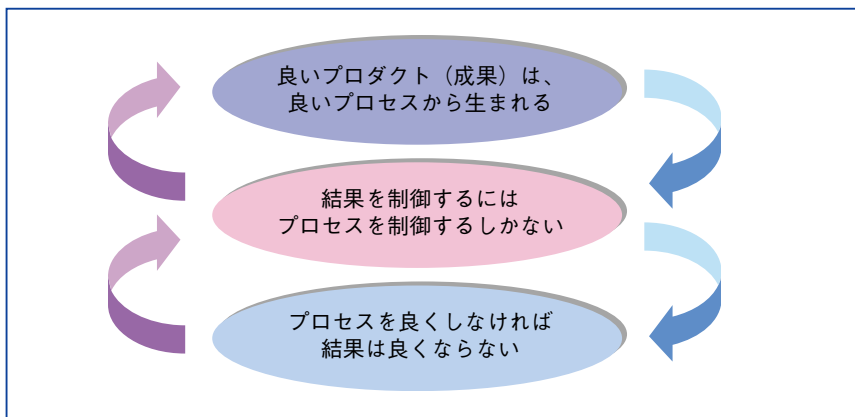


図1-7 プロセス改善の前提

こともあります。しかしながら、「定常的に」良い結果を得るためには、きちんとした仕事の進め方が必要です。この考え方は産業界では広く受け入れられていて、ISO 9001に代表されるようにプロセスを組織で取り決めて、これに従った仕事の進め方が推奨されています。

ソフトウェア開発に関しては「ソフトウェアライフサイクルプロセス (SLCP)」(ISO/IEC 12207:1995/ Amd.1:2002) にまとめられており、組織やプロジェクトのプロセスを構築する際の参考モデルとして活用できます。

図1-8および表1-4がSLCPの概略です。

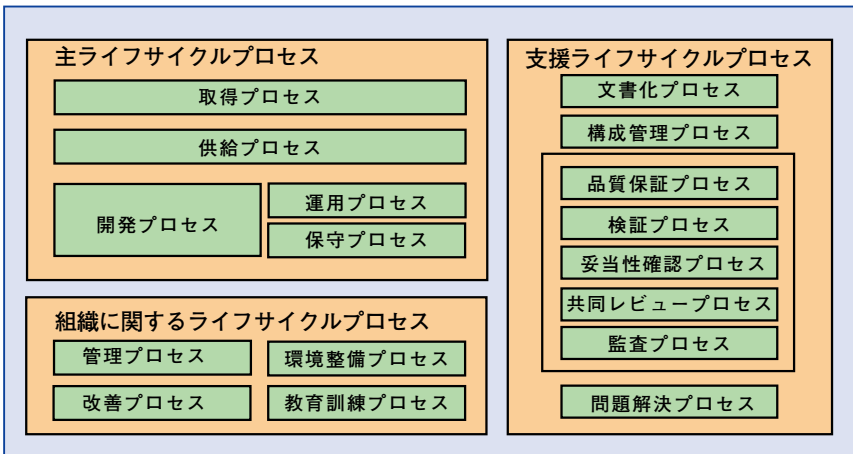


図1-8 ソフトウェアライフサイクルプロセスの体系 (ISO/IEC 12207)

表1-4 ソフトウェアライフサイクルプロセスの説明

プロセス	プロセスの説明
主ライフサイクルプロセス	ソフトウェア製品の開発、保守または運用を実施する人が使用する、次の5つのプロセスです。
①取得プロセス	取得者、すなわち、システム、ソフトウェア製品又はソフトウェアサービスを取得する組織のアクティビティを定める。
②供給プロセス	供給者、すなわち、システム、ソフトウェア製品又はソフトウェアサービスを取得者に提供する組織のアクティビティを定める。
③開発プロセス	開発者、すなわち、ソフトウェア製品を定義しこれを開発する組織のアクティビティを定める。
④運用プロセス	運用者、すなわち、利用者の稼働環境で計算機システムを運用するサービスを提供する組織のアクティビティを定める。
⑤保守プロセス	保守者、すなわち、ソフトウェア製品の保守サービスを提供する組織のアクティビティを定める。
支援ライフサイクルプロセス	それぞれが1つの明確な目的をもち、他のプロセスの活動の不可欠な一部として働く、次の8つのプロセスです。
①文書化プロセス	ライフサイクルプロセスで生産された情報を記録するアクティビティを定める。
②構成管理プロセス	構成管理のアクティビティを定める。
③品質保証プロセス	ソフトウェア製品及びその作成過程が規定要求事項に従い、確定した計画どおりであることを客観的に保証するためのアクティビティを定める。
④検証プロセス	(取得者、供給者又は第三者のために、) ソフトウェアプロジェクトが必要とするレベルに応じて、ソフトウェア製品を検証するアクティビティを定める。
⑤妥当性確認プロセス	(取得者、供給者又は第三者のために、) ソフトウェアプロジェクトが作成したソフトウェア製品の妥当性を確認するアクティビティを定める。
⑥共同レビュープロセス	活動の状況及び生産物を評価するためのアクティビティを定める。
⑦監査プロセス	要求事項、計画及び契約に適合しているかどうかを判定するためのアクティビティを定める。
⑧問題解決プロセス	原因又は性質にかかわらず、開発、運用、保守又は他のプロセスで発見された問題(不適合を含む)を分析し、取り除くためのプロセスを定める。
組織に関する ライフサイクルプロセス	ライフサイクルプロセスおよび要員からなる基本構造を確立して実行し、それらを継続的に改善していくために組織が使用する、次の4つのプロセスです。
①管理プロセス	プロジェクト管理等、ライフサイクルプロセスにおける管理のアクティビティを定める。
②環境整備プロセス	ライフサイクルプロセスのための基礎となる構造を確立するためのアクティビティを定める。
③改善プロセス	ある組織(すなわち、取得者、供給者、開発者、運用者、保守者、又は他のプロセスの管理者)が、自分のライフサイクルプロセスを確立、測定、制御又は改善するためのアクティビティを定める。
④教育訓練プロセス	適切に教育訓練された要員を提供するためのアクティビティを定める。

コラム プロセスの詳細記述の仕方

プロセスの記述は表1-4に示す内容だけでは不十分で、以下の特性を考慮して組織のプロセスを明確にするとよいでしょう。

- 目的は何か？
- 誰が実施するのか？
- 入力は何か？
- 生成される成果物は何か？
- いつ開始すべきか？開始基準は何か？
- いつ完了すべきか？完了基準は何か？
- このプロセスを遂行するために何を実施するか？
- このプロセスを遂行するために実施する3～6個の下位プロセスは何か？
- このプロセスの実績をどう判断あるいは測定するか？
- このプロセスの前後に実施されるプロセスは何か？

プロセスの記述は、普段からよく実施していて定着している活動ほど詳細に書くことができますが、作業品質を保証できる程度に記述してあれば構いません。文書化の労力はむしろこれから改善していきたい活動に投入するべきです。もともと実施していない活動ならば、なおさら詳細な記述が必要です。そして、それに従って実施してみて得られたノウハウや教訓をもとに記述を見直していきます。

■ ソフトウェアプロセス改善の狙いと視点

プロセス改善はその目的・目標を定めておこないます。具体的には「出荷後6カ月間に報告される障害を3件以下にする」とか、「1,000行当たり0.01件以下にする」あるいは「前期の生産性より10%向上させる」などです。これらの事業目標を達成するために、現状の組織能力で足りない部分を強化するという方法をとります。

事業目標としては以下のような項目が考えられます。

- プロセスパフォーマンス（業務遂行能力）の改善
- 自社のプロセスアーキテクチャ（業務標準）の確立・改善
- 自社技術の蓄積
- QCDの改善
- 競争優位性の向上 など

そしてゴールを達成するために、現状の仕組みで不十分な箇所の改善活動を実施します。前述の品質ゴールは「結果系」のゴールです。結果を良くするためには代替手段として「要因系」のゴールを設定して、それを達成することによって結果系のゴールが達成できると考えます。

結果系、要因系と書きましたが、結果系は直接制御できないもので、要因系は直接制御できるものです。たとえば、結果系のゴールである「出荷後6カ月間に報告される障害を1,000行当たり0.01件以下」は、出荷時点で品質が確定しているため、報告される障害件数をカウントするだけで、制御はできません。

一方、要因系のゴールは、結果系のゴールを達成するための中間ゴール（群）であり、ソフトウェア開発中に制御できます。たとえ

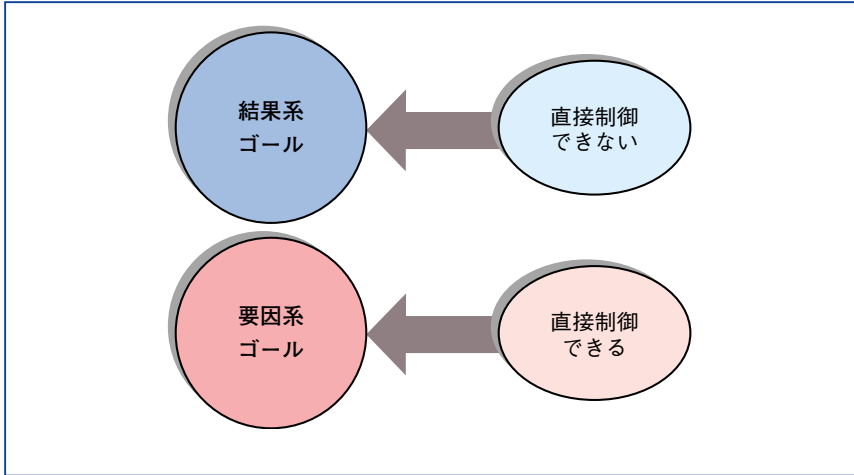


図1-9 結果系ゴールと要因系ゴール

ば、製品品質を良くするためのレビューやテストでの欠陥摘出数、テスト項目数などに対するゴールが該当します。要因系のゴールを達成していくことによって、結果系のゴールが達成できると考えます。そして、結果系のゴールが達成可能であるかについて、要因系のゴールの達成状況を定期的に監視して、必要であれば是正処置をおこないます。

1.5 プロセス改善のアプローチ

プロセス改善は、積極的に改善したい部分、あるいは改善の余地がある部分を特定して変更します。そのために、達成すべき目標に対して、現状で達成可能な水準とのギャップを分析します。そして、そのギャップを埋めるために改善（変更）すべきプロセスを特定し、目標の達成に効果がありそうな施策を検討し、適用します。施策、すなわちプロセスの改善を考える際には、表1-5が参考になります。

多くの企業でおこなっている施策を2つ紹介します。1つは、システム障害が発生してお客様に迷惑をかけたことがきっかけとなり、改善施策を検討するケースです。もう1つは、グローバル標準として認知されつつあるアセスメントモデルを使って診断をおこない、弱みに対して改善施策を検討するケースです。

表1-5 改善施策を選定する際に参考にすべき点

施策検討の糸口		打つべき施策	実施すべきこと
成功実績	有	自己組織で経験済みの施策	ベストプラクティスの導入・展開
	無	自己組織では未経験の施策	動向調査および先行評価
失敗経験	有	失敗経験に基づいた施策	プロセスの欠陥分析・原因分析
	無	アイデア提案型の施策	リスクを考慮
効果の定量化	有	達成状況が監視可能な施策	定期的に測定
	無	達成状況を監視可能にすべき施策	計数化を考慮
アセスメントモデル	有	アセスメント結果に基づいた施策	弱点の克服、長所の強化
	無	自主提案及びプロジェクト完了報告や反省会などからの施策	改善提案

■ 失敗を契機にしたプロセス改善の例

失敗を契機として、再発防止を含めて品質改善の目標を設定します。たとえば、出荷後の障害件数を20%削減するという組織目標を立てたとします。現在のギャップは20%です。問題解決までの手順

は、(a) 現象を分類する、(b) 原因を分析する、(c) プロセス改善施策を策定する、となります。

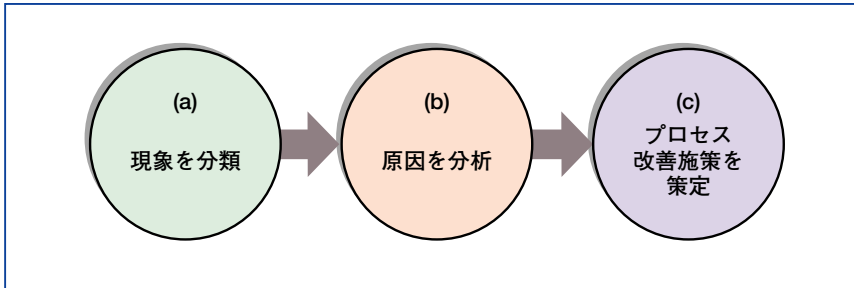


図1-10 失敗を契機にしたプロセス改善の問題解決までの手順例

(a) 現象を分類する

過去の障害の現象や原因を分類します。

- ミスかモレか。ミスであれば、論理ミスか、インタフェース・ミスか、境界値ミスか、文法ミスか、など。
- 欠陥の作り込み工程はどこか。要件定義、基本設計、詳細設計、コーディング、あるいはデグレードか。

(b) 原因を分析する

- なぜ欠陥を作り込んでしまったか。
初歩的なミスか、難易度の高いミスか。どうすれば欠陥の作り込みを回避できたか。
- なぜ出荷前に欠陥を摘出できなかったか。
本来、どの工程で摘出すべきであったか。どうすれば摘出できたか。

(c) プロセス改善施策を策定する

ここで表1-5を参考にしてプロセス改善施策を策定します。

● 欠陥を作り込まない施策

たとえば、障害分析に基づいて失敗要因を抽出し、設計手順のルール化や設計時に欠陥を作り込まないようにするためのチェックリストの作成が考えられます（失敗経験に基づく）。また、品質が保証されている部品の再利用率を向上させるということも挙げられます（成功実績に基づく）。

● 欠陥を摘出するための施策

レビューおよびテストの質を高めることが考えられます。工程内で障害を見逃した原因を分析して、レビューの視点を追加したり、関連のテスト項目を増加させたりなど考えられます。認定レビューアを置くとか、レビューやテストの開始・完了基準を見直すとの施策もあります（プロセス改善提案に基づく）。

改善施策を実施したことによって「出荷後の障害件数を20%削減する」という組織目標を満足できるかを検討します。施策とその影響あるいは効果に関して仮説を立て、施策を実施しながら検証します。

■ アセスメントモデルをベースにしたプロセス改善の例

望ましい活動を記述しているモデルから乖離している活動を特定して、改良、あるいは不足している活動の追加という施策が必要です。問題解決までの手順は、(a) モデルとのギャップを特定する、(b) 改善する活動を選定する、(c) プロセス改善施策を策定する、となります。

ここでも、目標の1つとして「出荷後の障害件数を20%削減する」という組織目標を立てたとします。

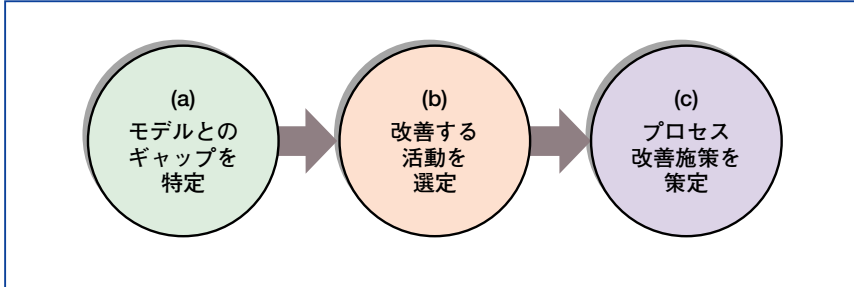


図1-11 アセスメントモデルをベースにしたプロセス改善の手順例

(a) モデルとのギャップを特定する

診断するプロセスを絞り込み、アセスメントモデルを使ってプロセスを診断し、アセスメントモデルとのギャップを特定します。詳しくは別冊『プロセス診断活用編』に詳細をまとめてありますので、そちらを参照してください。

(b) 改善する活動を選定する

ギャップ分析の結果のうち、組織目標に関連する、すなわち品質関連の活動で弱み、あるいは強みとして特定されたものを抽出します。それらに対して改善の対象を絞込むために優先度付けをします。ここで組織目標を達成するために改善すべき活動が選定されます。

(c) プロセス改善施策を策定する

選定された活動を改善する手段として、アセスメントモデルに記載されている補足情報を参考にしたり（詳細は別冊『プロセス診断

活用編』を参照)、先の「失敗を契機にしたプロセス改善の例」のように20ページの表1-5を参考にしたりして改善施策を策定します。

失敗を契機にしたプロセス改善の例とアセスメントモデルをベースにしたプロセス改善の例の2つの改善施策例を紹介しましたが、実際には、両者を組み合わせてプロセス改善をおこなっている組織が多いと思います。

1.6 プロセス改善の対象プロセスの発見

プロセス改善は現在直面している問題を解決したり、現状の強み

表1-6 各ライフサイクルプロセスにおいてよく起こる問題と改善すべき対象プロセスの関係例

問題	改善対象プロセス
主ライフサイクルプロセス	
①コーディング=ソフトウェア開発という黎明期のメンタリティ	? ^{注4}
②スパゲッティプログラム	開発プロセス
③要求をめぐる問題	開発プロセス
④保守をめぐる問題	保守プロセス
⑤供給者からの納入ソフトの品質が悪い問題	供給プロセス
⋮	
支援ライフサイクルプロセス	
①同じトラブルの繰り返しという問題	問題解決プロセス
②巨大開発と混乱	構成管理プロセス
③開発標準が守られない問題	品質保証プロセス
⋮	
組織ライフサイクルプロセス	
①巨大開発と混乱	管理プロセス
②仕様と見積りをめぐる問題	管理プロセス
③工程管理の不備	管理プロセス
④セキュリティ対策	環境整備プロセス
⑤エンジニアトレーニングの不備	教育訓練プロセス
⑥手と勘に頼りエンジニアリングツールを適用しないという問題	環境整備プロセス
⑦プロセス改善推進体制の問題	改善プロセス
⋮	

注4. 表内の ? には、状況に応じて「開発プロセス」「文書化プロセス」「管理プロセス」などが該当する。

をさらに強化したりする活動です。表1-6は改善のテーマをソフトウェアライフサイクルプロセスに関連付けた例です。自社で起きている問題を解決する際に、プロセス改善の対象としてどのライフサイクルプロセスを見直すべきかを考える参考にしてみてください。

プロセス改善の枠組みとして、国際規格15504準拠のアセスメントモデルの活用を推奨します。モデルを活用すると、現状の問題点やプロセス改善のテーマを効率的に見つけることができます。詳しくは、別冊『プロセス診断活用編』を参照してください。

1.7 プロセス改善の効果

プロセス改善による効果は既に多くの組織で実証されています。ここでは米カーネギーメロン大学ソフトウェアエンジニアリング研究所のレポートから、品質、コスト、スケジュール、顧客満足、ROI（投資収益率）について、表1-7にその一部を紹介します。

表1-7 プロセス改善の効果（報告例）

（カーネギーメロン大学ソフトウェアエンジニアリング研究レポートより（CMU/SEI-2006-TR-004: D. L. Gibson, D. R. Goldenson and K. Kost, Performance Results of CMMIR-Based Process Improvement）^{注5}）

（a）効果の概要

カテゴリー	効果（中央値）	効果の範囲	データ数
品質	48%	2～132%	34
コスト	34%	3～87%	29
生産性	61%	11～329%	20
スケジュール	50%	2～95%	22
顧客満足	14%	-4～55%	7
ROI （投資収益率）	4.0 : 1	1.7 : 1～27.7 : 1	22

次ページに続く

注5. レポートは次のURLから入手可能。

<http://www.sei.cmu.edu/publications/documents/06.reports/06tr004.html>

表1-7 プロセス改善の効果（報告例）（つづき）
 (b) 効果の報告例

カテゴリー	効果
品質	①欠陥を44%削減（CMMIレベル3達成組織） ②製品の問題件数を40%削減、うち深刻度1の問題を80%削減（CMMIレベル3からCMMIレベル5への移行で） ③テスト前の欠陥摘出率を50%から70%へ向上させ、出荷後の欠陥を0.35件/KLOCへ（CMMIレベル5からCMMIレベル5への移行で）
コスト	①コストを52%削減（CMMIレベル2、さらにCMMIレベル3への移行で） ②プロジェクト予算どおりの出荷が90%からほぼ100%へ（CMMIレベル3からCMMIレベル5への移行で） ③メモリークの早期検出による再作業工数の削減により、コストを95,000ドル（1億円強）削減（CMMIレベル5達成組織） ④3年で品質コストを45%以上から30%以下へ（CMMIレベル5からCMMIレベル5への移行で）
生産性	①20%以上の生産性向上（CMMIレベル3からCMMIレベル5への移行で） ②生産性を72%向上（CMMIレベル3からCMMIレベル5への移行で） ③生産性を25%向上（CMMIレベル5からCMMIレベル5への3年間で）
スケジュール	①納期遅れ日数を70～80%削減（CMMIレベル2達成組織） ②開発計画からのずれの平均日数を約130日から20日以内に削減（CMMIレベル2の達成後1年で） ③スケジュールのずれを約15～25%削減（CMMIレベル3からCMMIレベル5への移行で） ④オンタイムデリバリが79%から89%に改善（CMMIレベル3からCMMIレベル4への移行で）
顧客満足	①CMMIレベル2の時と比べて、報酬 ^{注6} が55%増加 ②顧客からの可能な報酬 ^{注6} の98%以上を獲得（CMMIレベル5達成組織） ③契約者実績評価調査において、適用可能なすべてのカテゴリーで「極めて優秀」という評定値を獲得（CMMIレベル5達成組織）
ROI （投資収益率）	①リリース後の欠陥削減によりROIが3：1以上（CMMIレベル3からCMMIレベル5への移行で） ②ROIが6：1（CMMIレベル3達成組織）

また、プロセス改善によって組織の能力が向上すると、アセスメントモデルを利用して診断した場合には、組織のソフトウェア開発・管理の能力を示す指標である「成熟度レベル」が向上します。成熟度レベルが向上するにつれ、以下の3つの能力が向上していきます。

注6. 「顧客満足」の効果に記載されている「報酬」とは、欧米流のIncentive Contractにおける成功報酬である。

- 予測能力
- 制御能力
- 効率

図1-12はそれぞれの能力が向上することによるQCDなどの特性値の変化を分布で表したものです。C（コスト）を例にとって説明します。

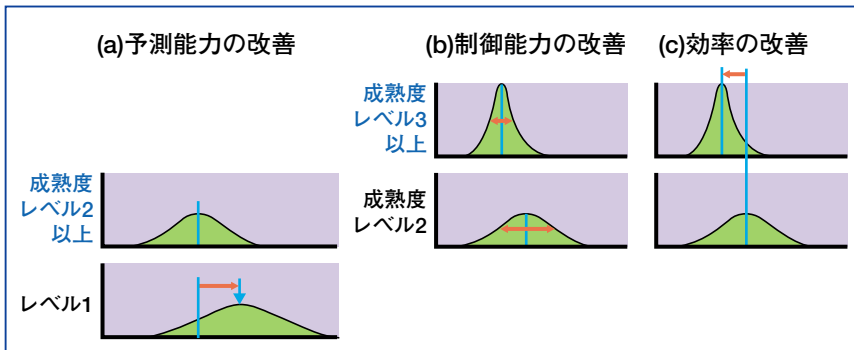


図1-12 成熟度レベルの向上による能力の改善^{注7}

(a) 予測能力の改善

レベル1のときはプロジェクトの能力がわからず、たとえば500時間で開発すると宣言して実際には平均的に700時間位かかっています（図1-12「(a) 予測能力の改善」の下図）。それが成熟度レベルが2になると、500時間と見積もって、結果も平均的に500時間になります（図1-12「(a) 予測能力の改善」の上図）。

(b) 制御能力の改善

レベル2のバラツキ（図1-12「(b) 制御能力の改善」の下図）

注7. カーネギーメロン大学SEIの資料をもとに一部改変した。

をベースとして、高レベルへと改善することによってバラツキが小さくなります（図1-12「(b) 制御能力の改善」の上図）。たとえば、見積りの精度が±20%だったものが、±10%、±5%となっていきます。

(c) 効率の改善

レベル2の生産性（図1-12「(c) 効率の改善」の下図）をベースとして、高レベルへと改善することによって生産性が向上します（図1-12「(c) 効率の改善」の上図）。たとえば、開発時間が500時間だったものが、400時間、350時間となっていきます。

以上はC（コスト）で説明しましたが、Q（品質）なら欠陥数が、D（納期）なら納期遅れの日数が改善されていく過程を同様に説明できます。

1.8 まとめ

これまで述べてきたように、社会基盤となりつつあるソフトウェア製品の開発に際しては、その製品に課せられた責任を果たすため、組織目標を達成するため、自分たちの業務遂行能力を向上させるためなど、同時にいくつものミッションがついてきます。それを解決する手段の1つとしてプロセス改善は有効です。地道ですがスマートな活動であり、改善を継続することによって大きな効果につながります。顧客満足、パートナー満足、社員満足を向上させるために、まずは具体的な目標設定および現状分析から始めてみてください。

第2章 プロセス改善の姿・形

この章では、プロセス改善がどのように具体的な手掛かりを用いてなされるのか、これまでのITの歴史の中で提案された着眼点をいくつか紹介しながら考えていくこととします。

2.1 改善サイクル

プロセス改善を進めるうえで基本となる重要な考え方は、「改善サイクルをまわす」という考え方です。この考え方が重要で基本なのは、その場限りのやりっぱなしや、同じ誤りの繰り返しといった知識や技術の積み重ねのない状態を脱して、着実かつ明瞭で後戻りのないプロセス改善を実現する必要があるからです。あなたの周囲には、言いつぱなしや思い着きの廃墟のような状態がないでしょうか。

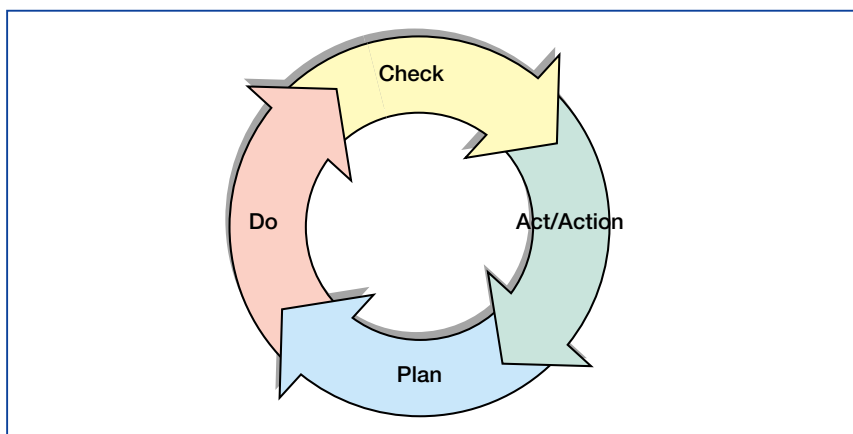


図2-1 CAPDoサイクル

また現状に安住して、新しいやり方は「余計な仕事の増加」としかとらえられない風潮はないでしょうか。

改善サイクルをまわすという考え方は、常に自分たちの仕事の本来の目的に照らして、自分たちの行動がより良い方向に向かっているのかどうかをチェックしていくという基本的な考え方です。詳しくは、別冊『虎の巻編』（本書発行時点執筆中）をご覧ください。

改善サイクルは、一例として前ページの図2-1のようにCAPDo（またはPDCA）という順序を用いることもあります。これらCAPDoの各フェーズをブレイクダウンして見てみましょう^{注8}。

C（Check）

現状の課題を認識して評価することを表します。プロセスの現状がどうなっているのかを調べたり、測定したりします。ソフトウェアプロセス改善では、広い意味でのプロセスアセスメントの実施とみなすこともできます。また、既にプロセス改善の改善サイクル^{注9}をまわしている場合には、改善を実施している現状を評価することも意味します。その場合には、改善が目的実現のために着実に進んでいるかどうかの評価ということになります。

注8. PDCAサイクルの解釈には、いくつかのバリエーションが見られます。改善サイクルでは、最初に現状を知るという考えのもとに、PDCAの代わりにCAPDoという考え方もあります。ソフトウェアプロセス改善ではその詳細論議にこだわる必要はないでしょう。ここで示したものより詳しい解説は、たとえば次のサイトにあります。

<http://www.dartmouth.edu/~ogehome/CQI/PDCA.html>

<http://www.asq.org/learn-about-quality/project-planning-tools/overview/pdca-cycle.html>

注9. 改善サイクル（PDCA）のありかたには、単に業務を回しながらその結果を点検・反省し改善を積み上げていくという視点と、事業目標の達成のために改善計画をたて、組織的な仕組みとして改善サイクルを回していく視点が含まれています。

本節の「プロセス改善の改善サイクル」ということばでは、後者の視点を表しています。

A (Act/Action)

Checkフェーズでおこなった評価に基づいて、ここでは「次に基本的には何をすべきか」を決定し処置します。この意思決定のためには追加的な分析も必要かもしれません。このことはプロセスを改善することに当たります。既にプロセス改善の改善サイクルをまわしている場合には、アセスメント結果の分析と選択岐の意思決定のフェーズだと見ることもできます。

P (Plan)

Act/Actionフェーズでおこなった意思決定に沿って、改善の具体的な考え方や各部署で実施すべき内容を確認し、実施に移します。ここで重要なことは、たとえばプロジェクトの計画であれば、そのプロジェクト実施内容自体の的確さも重要ですが、「改善」の内容がどのような点であり、現実の実施の際にどのような点に着目して今までとは異なるプロジェクトプロセスを実施していくのか、また、プロセス改善は仕事を成し遂げるうえでどのような価値があるのか、などが関係者すべてに明らかとなっていることです。既にプロセス改善の改善サイクルをまわしている場合には、Planフェーズとは改善の具体化計画に当たります。Actフェーズでおこなわれた意思決定、優先順位付けに従って、詳細改善計画を作ることに該当します。

D (Do)

改善されたプロセスの実施・維持管理・データ収集を意味します。もちろん、単に従来どおりのプロセス内容を実施するのではなく、改善されたプロセスを意識しながら実施します。その実施状況を意識的にモニタリング（「監視」という訳語を当てることもあります）す

ることも必要です。既にプロセス改善の改善サイクルをまわしている場合には、Planフェーズで作成した詳細改善計画を実装するフェーズに当たります。正しく実装されたかどうかを次のサイクルCheckフェーズでチェックします。

CMMI^{注10}モデルでよく使われるIDEAL^{注11}モデルも同様の改善サイクルのモデルです。

■ 改善サイクルをまわすのは誰？

このCAPDoサイクル推進のリーダーシップは誰が取るのでしょうか。プロセス改善は、ソフトウェア開発・保守・運用組織での活動としては、組織活動としてプロセス改善を具体的に提唱または認可し、推進の責任を負う人がいることが予想されます。しかしプロセス改善をしなければいけないという問題提起は、開発組織内の多くの立場から提出されます。プログラムを作成している人は、プログラム開発作業の混乱を改善したいと思うでしょう。プロジェクト管理の失敗に苦しめられているプロジェクトリーダーやプロジェクトメンバはプロジェクト管理プロセスの改善を切実に願うでしょう。また、多くのトラブルプロジェクトに悩まされている企業の管理職は、そうした事例の撲滅はできないかと模索するでしょう。このように、プロセス改善は、ソフトウェア開発・保守・運用組織の上から（トップダウン）も下から（ボトムアップ）も、自分の問題として提案されます。このバラエティを尊重することは重要です。

実際にプロセス改善が開始されると、ソフトウェアエンジニアリングに造詣があり、推進意欲もあるエンジニアのグループの創意に

注10. <http://www.sei.cmu.edu/cmami/>

注11. <http://www.sei.cmu.edu/ideal/>

導かれた活動がおこなわれることの有効性も指摘されています。

プロセス改善を実際におこなうのは、このグループではありません。それはプロセスの実際の担い手である現場のメンバと、プロセス実施の責任者であるプロセスオーナーです。これらの実際の担い手がプロセス改善に本気で取り組まなければ意味がありません。

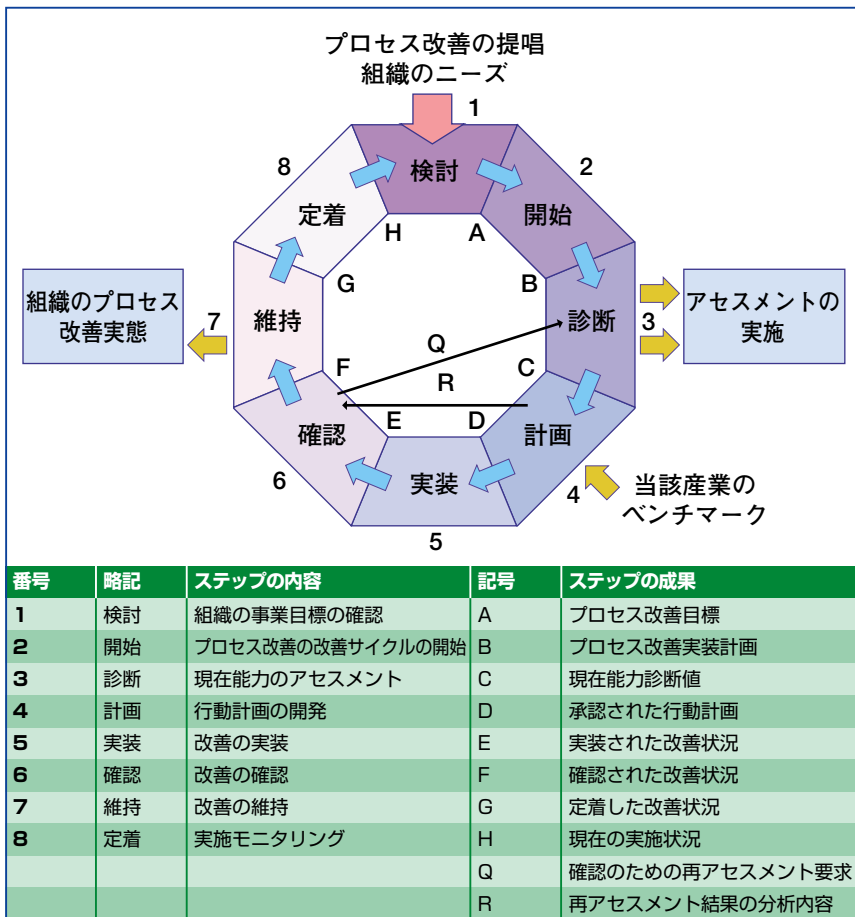


図2-2 15504-4での改善サイクル

ボトムアップの活動がいかに意欲的でも、ソフトウェア開発・保守・運用組織が全体としてその活動に意義を認めていなければ無力に終わる可能性があります。その意味で、企業トップの役割は非常に重要です。プロセス改善活動を組織の事業目標実現の活動の重要な一環として位置付け、プロセス改善のCAPDo活動にかかわっていかねばいけません。プロセス改善活動では、このような役割をコミットメント（確約）という用語で表しています。

■ プロセス改善サイクルの国際規格

プロセス改善の国際規格15504の第4部では、プロセス改善の改善サイクルについて、前ページの図2-2のようなサイクルを想定しています。このサイクルの考え方も基本的にはCAPDoサイクルと変わりません。このサイクルの詳しい説明は、別冊『虎の巻編』（本書発行時点執筆中）で示していますが、評価診断に基づいて改善計画を立て、実施し、実現状況をチェックしながら、プロセス改善を重ねていくという考え方をしています。

2.2 プロセス管理の手法やツール

プロセスを改善するには、プロセスの技術的な側面と、組織・管理的な側面が車の両輪としてうまく噛み合っただけで向上していかねばいけません。プロセス改善というと、とかく、プロセスの手順の遵守のような側面がイメージされがちですが、それは間違いです。実際には、ソフトウェアプロセスをそのすべての側面で能力向上させていくこと、また、技術的な手法を含めプロセスの成果に現れるプロセス実施手法のあらゆる面での向上が重要です。

ここでは、最初にプロセスの管理的な側面を支援する手法やツ

ルにどのようなものがあるか見ていきましょう。

■ プロセスの現実を理解するツール

プロセスは、その関係者から見て、「理解できる」ものである必要があります。プロセスを理解するには、まずそれぞれの現在実施しているプロセスがどのような目的、特性、要件があるのか、あるべきなのかをよく理解しなければいけません。そのための手法やツールを挙げてみましょう。

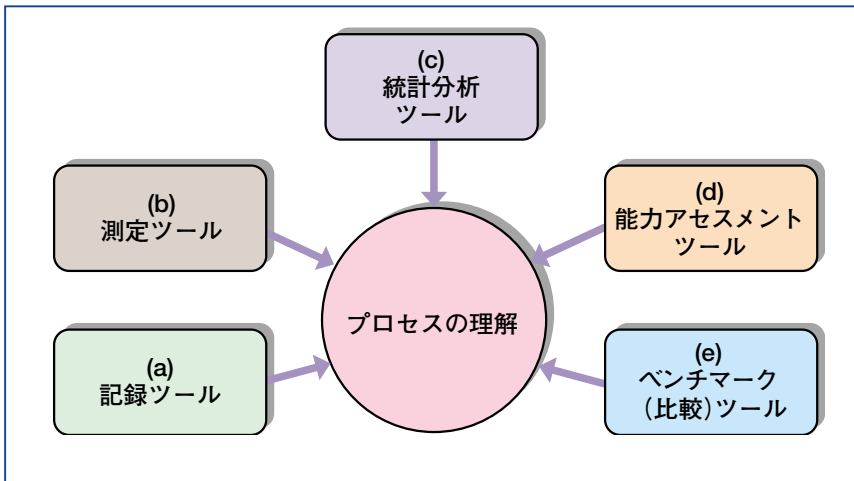


図2-3 プロセスの理解を助けるツール

(a) 記録ツール

プロセスの現状を記録して数字・グラフ・文章・図解などにして表現することが、プロセスを「見える」ものとする基礎的な方法です。単に記録するだけでも、個人の頭の中にあるものを客観化し、また声などによる指示・コミュニケーションを記録してプロセスの

情報が消えてしまわないようにすることができます。

たとえば、ソフトウェア製品出荷後の不具合報告数を記録していくのは初歩的な努力です。

(b) 測定ツール

ある程度目の前に姿が見えているものは、そのまま記録することができますが、多くのプロセスの技術的な要素は、意識的に測定しなければ、記録できる形のものとなりません。

コラム プロセス理解の主体

「プロセスの現実を理解する」というときの、その主体は誰でしょう。それは、プロセスの関係者すべてです。当然、組織の管理者やソフトウェアの調達者は、ソフトウェアプロセスの現状や問題状況を把握したいと思います。しかし、本当に正確にプロセスの現実をしっかりと把握しなければならないのは、現場の当事者であり、プロジェクトリーダー、プログラム作成者、品質管理者、ソフトウェアサプライチェーンの関係者などが含まれます。

プログラム作成者は、自分の状況はわかっているだろうと思う人もいますが、それは間違いです。プロジェクトの状況や、そのプロセスの目標について、しばしばプログラム作成者は自分の客観的状況を理解していません。その際、理解しようという動機や、理解するための情報を欠いている場合もあります。ここで、重要なことは、プロセスの状況が見えにくい原因は、しばしば関係者の多くが「見える化」に対して無関心である、また副次的な関心

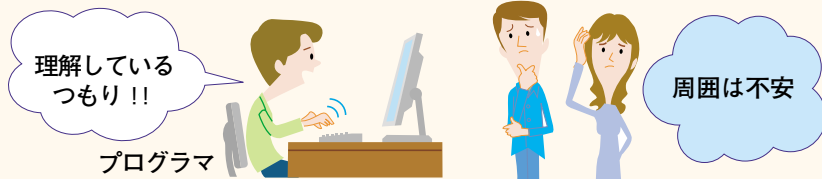
たとえば、ソフトウェアの不具合を修正するのに要した時間を正確に記録することは、測定と見ることもできますし、作成したプログラムの複雑度を複雑度解析ツールにより調べることも測定です。

(c) 統計分析ツール

数値として記録されたものは、その統計を取ることによって、トレンドが明確となり、また統計的品質管理のツールの応用ができるようになります。また、統計的に有意なデータを得る必要性から、

しか持たないということに起因しているということです。

時に、プロセスの「見える化」の議論に対する批判として、「見える」ようにすることによって、実情をよく知らない外部の人が専門家の判断を尊重せず、無用な注文を出すことの弊害が指摘されることもあります。確かに専門家の優れた判断はしばしば、「任せる」ことから生まれます。しかし、実際には情報が流通せず、相互不信や無責任な丸投げが横行する弊害のほうが強いのです。重要なことは、アイデアや情報、あるいは成果物についてオープンな討議と相互向上を促進できるような文化的な環境を準備することでしょう。



新しい測定のニーズが提起されることもあります。

たとえば、開発したソフトウェアの規模と発見された不具合の関係の分析や、それに基づく特定の出荷物の不具合の数が通常か異常かといった判断がおこなわれます。

(d) 能力アセスメントツール

プロセスの状況をパフォーマンスの視点からではなく、プロセス能力の点から診断するツールです。別冊『プロセス診断活用編』を利用してプロセスアセスメントを実施することも、このような能力アセスメントツールの利用に当たります。

(e) ベンチマーク（比較）ツール

いろいろなデータや情報は、ほかの組織や部署での実例や、業界平均などと比較することにより改善努力のうえで説得力のあるものとなります。

たとえば、ソフトウェア製品出荷後の不具合報告数を他部署、他社や業界平均と比較することや、プロセスアセスメント結果を特定条件下で他部署と比較することなどがおこなわれます。ベンチマークには、標準化された質問票を用いることがよくおこなわれます。

■ プロセスを意図通りにするための手法

プロセスの状況を「見える化」することは、それだけでは無力です。「見える」ようにした結果に基づいて行動を起こさなければいけません。プロセスを目的に沿う方向でコントロールしていくといってもよいでしょう。また、自分の作業を、仕事に追われるのではなく、自分が計画した通りに進めていくということだと思ってもよ

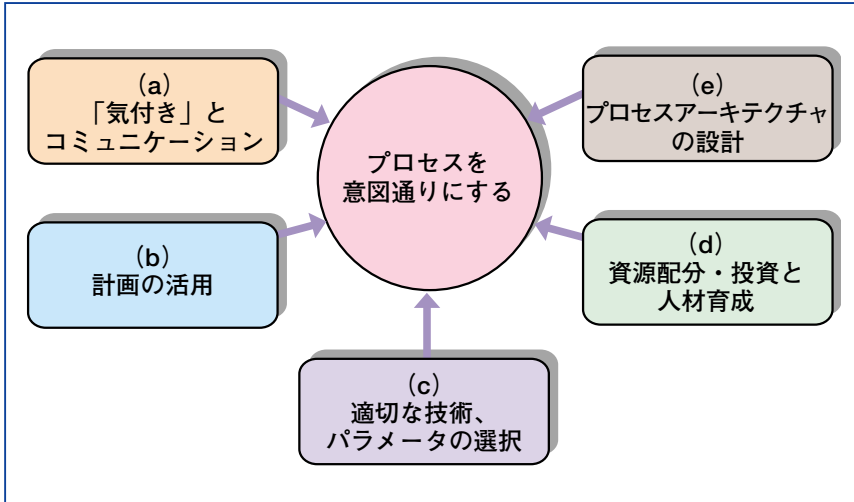


図2-4 プロセスを意図通りにするポイント

いでしょう。

このようなプロセスを意図通りにするためのポイントについて、どのようなものがあるか、以下で逐次見てみることにしましょう。

(a) 「気付き」に基づく対策と、チームのコミュニケーション

プロセスの「見える化」の結果として、プロセスの不具合や、事業目標から見た不足などが明らかとなった場合、特にはっきりとした改善策が設定されなくても、プロセスの関係者が自分のプロセス実施活動を慎重におこなうようになっていたり、努力の水準を高めたりすることがあります。このような場合は、「気付き」そのものが1つの対策となっていると見ることもできます。しかし、このような対策は、気付きの記憶が薄れるとともに、対策の程度も薄れていく可能性があります。

さらに、このプロセスの「見える化」と「気付き」をもととして、

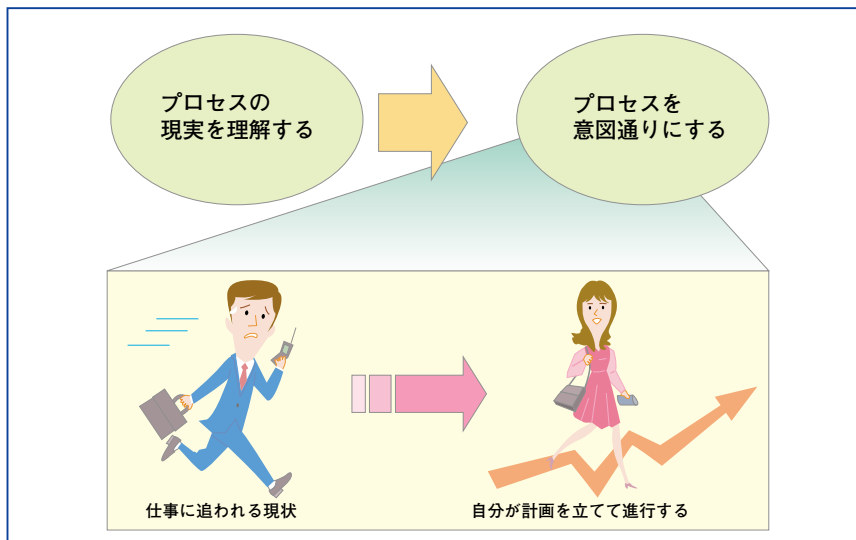


図2-5 プロセスを意図通りにするためのきっかけ

開発・運用のチームの中で、どう改善すれば良いかの討議をおこない、チームメンバーのはっきりとした合意を獲得することは有意義でしょう。また、プロセス実施の中で、プロセスの問題点を意識したコミュニケーションを活発におこなうことがプロセスの不具合を防ぎ、また、より積極的なプロセス技術の採用につながっていくことはよくあることです。

(b) 計画ツールの活用

プロセスの実施は、プロセスの計画から始まります。その計画を着実に立てることを補助する計画ツールを活用することもプロセスの改善につながります。計画ツールの上手な利用ということもあるでしょうし、計画ツールに投入する情報の整備ということも重要でしょう。計画を着実にするということは、計画に盛り込まれる予測

が着実であるということにより裏付けられます。予測を着実にすることと、「見える化」の過程での「測定」とは当然密接に関連しています。

「計画」は、最初にすべてが完璧である必要はありません。ソフトウェアプロセスでは、いろいろ具体的な条件はプロセスの進行とともに深まっていくということがよくあります。このような変化に柔軟に対応すること、また最終的な「完成」へ向けて多くの開発努力を整理・集約・変更、場合によっては取捨選択していくことも計画の重要な側面です。

(c) 技術選択と実施レベルパラメータの設定

プロセスの実施では、どのような技術を採用するのが大きな岐路となります。プロセスの不具合や不足が明らかとなった場合には、選択する技術を変更することは非常に重要なプロセス制御要素です。このことには、プロセスの実施コンセプトや「プロセスパターン」の選択といったごく基本的なものから、特定の技術の適用の仕方や、プロセスで利用するソフトウェアコンポーネントの変更といった現場の判断も含まれます。

よくエンジニアに対していわれることの中に、「ソフトウェアエンジニアは、常に複数のプロセス実施方法を知っている必要があります、具体的な状況や具体的な対象に従って最も良いプロセス実施方法を選択できなければいけない」というアドバイスがありますが、プロセスの技術選択の中にはそのような観点も含まれていると見てよいでしょう。

もちろん、あるプロセスで今までやっていなかったプラクティスを、技術的な知見に基づいて新たに開始する、あるいは今までやっ

ていたことを廃止するという観点もこの対処法に含まれます。

技術選択では、新しい技術の採用・工夫にチャレンジすることも、しばしば必要であり重要な着眼点です。

しかし、プロセスを着実に実施するという観点からは、「新技術だから良いものだろう」と思いこむことには問題があることを知っている必要があります。

プロセスで実施するプラクティスの中には、実施回数や実施の強度などがコントロールできるものがあります。たとえばあるプロセスでの「レビュー実施」というプラクティスがあった場合に、そうしたレビューの実施回数、レビュー詳細度、準備すべき資料などの条件、参加者の役割設定や参加範囲などはいろいろな実施水準があります。そうした実施水準をプロセスのパラメータとして（実施レベルパラメータ）、プロセスのパフォーマンスをコントロールすることができます。

(d) 資源配分計画と人材育成

プロセスを支えるものは人であり、また人が活用できるさまざまなリソース（資源）です。プロセスのパフォーマンスを向上させるには、的確な人材やその他のリソースの投入が的確にできるかどうかが大きな分かれ目となります。プロセスの問題を考えるときに、利用できるリソースを固定して考える必要はありません。たとえば、特定のハードウェアやネットワーク設備、またソフトウェアツールのようなリソースを投入してプロセスパフォーマンスが向上するのなら当然そのようにすべきです。管理的な側面からは、リソースを配分する仕組みやリソース投資の仕組みを合理的かつ柔軟にすることが実現されなければいけません。

資源配分は、多くの場合現場の努力だけでは解決しないでしょう。この点に管理者・経営者のコミットメントの大きな役割があります。改善への投資という観点から見た場合にわかりやすいのはこのような投資であり、組織の管理者は、的確な投資をコミットしなければいけません。

リソースの中で最も重要な人材については、短期的な配置の問題のほかに、より長期的な見通しの中での人材の育成という課題があります。ソフトウェア産業での人材の育成は、従来の産業での「人材の訓練」の側面と、研究開発に従事する「R&D人材の育成」の側面との二重の側面があります。先進的な人材は自らプロセスの技術を開発し、プロセス改善のためにその技術を適用していく可能性があります。いずれにしても、洗練されたプロセスは優れた人材の力により支えられています。

(e) プロセス全体のアーキテクチャの設計

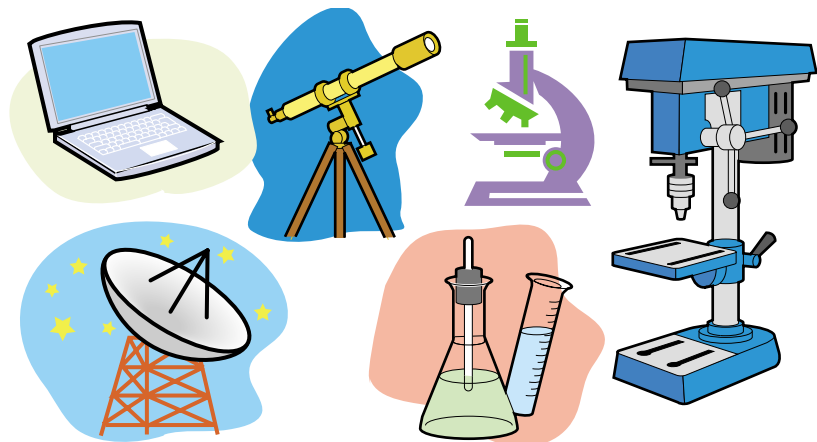
事業目標を的確に達成するためには、それぞれの企業は固有のプロセス群の組み立て、組み合わせ（プロセスシステム、プロセスアーキテクチャ）を持っています。プロセス改善のモデルは、そうした固有のものを平均的なものや理想的なものに近づけようとするものではありません。そうではなく、個別の企業の状況や特性を最大限生かしながら、なおより良いプロセスを求めていくにはどのようにすればよいのかについてのガイドを提供しています。したがって、ソフトウェア開発・保守・運用組織にとって、プロセス改善の最大の課題は自社にとって現在よりさらに有効なプロセスの組み合わせとプロセス群を統合した能力を生み出していくことです。

2.3 プロセス実施技術

プロセスを実施するうえでの技術は、IT技術の総合的な進化とともに発展してきました。それはソフトウェアプロセス能力の成熟の視点から見ると、前進と後退が交互に訪れるような経験であったともいえます。

1980年代にはメインフレーム文化の中でCASEツールが一定の利用の高まりを見せましたが、その後のプラットフォームの変遷の中でこの潮の高まりは1つの引き潮の時期を迎えました。

技術の適切な利用は不可欠ですが、それはマーケット環境に適合したビジネス目標とマネジメントの中でこそ真価を発揮します。また、継続的なプロセス改善努力の中で位置づけられてこそ、継続的な技術としての進化が望めます。



技術を駆使する

■ 実施支援ツール

「外から見てもわかりやすいプロセス実施技術の姿は、個別のプロセスの実施を直接支援するツールの利用です。

(a) 統合開発環境とCASEツール

ソフトウェアプロセス実施のうえで用いられる基本的な技術ツールは、開発プロセスにフォーカスした統合開発環境（IDE）ツールです。これは伝統的にはCASEツールと呼ばれてきたツール群の延長上にあるツールです。エディタ、コンパイラ、デバッガといったごく伝統的なツールや、オブジェクト指向におけるクラス管理ツール、部分実行ツール、UMLエディタとの統合環境などがこのカテゴリに含まれます。

(b) 構成管理ツール

構成管理の初歩的なものは、ソースコードや設計文書の変更管理ツールですが、より一般的には要求管理ツールを含みます。また、CCB（変更管理委員会）といった構成管理の組織的な実施形態の支援も含んでいます。

(c) 要求分析・要求管理ツール

ソフトウェア開発の要求引き出しと要求変更管理は大きな問題です。これらをサポートするツールはいろいろなアプローチのものがあります。

上流・下流のプロセスを貫通した要求変更の管理ツールを利用すると、要求項目・仕様・実装・テストを通じたトレーサビリティを確立することができます。

(d) テストツール

テストツールには、テストを実施するためのデータやコンポーネントを提供する実装技術的なツールと、テストプロセスを管理するためのツールとがあります。テストツールとデバッガを組み合わせることもあるでしょう。テストは、開発のプロセスの上流から下流まで多くのプロセスでそれぞれに実行可能です。特にシステム統合や運用時のテストをあらかじめ上流プロセスにおいて設計・実装しておくことはテストツールの観点からも非常に重要です。

テストプロセスを支援するツールのうち、多くの製品化が見られるものに、不具合点追跡ツールがあります。

(e) その他のツール

その他見積りプロセスに対する見積りツール、プロジェクト管理プロセスに対するプロジェクト管理ツールなどがあります。

■ 実施技術・手法の実例

ツール以外の技術・手法の観点からプロセス実施技術の特色を見てみましょう。

(a) 開発ライフサイクルモデル

ウォーターフォール、スパイラル、インCREMENTALなどの開発ライフサイクルのモデルが提唱されています。ここで重要なのは、モデルの名称の選択ではなく、皆さんの直面している現場実態やビジネス環境では、実は、どのような開発ライフサイクルの考え方が適切なのかを考えきれる柔軟性です。

ライフサイクルモデルではありませんが、ソフトウェアライフサ

イクルの基本的な考え方の国際標準として、ISO/IEC 12207 (SLCP) や、ISO/IEC 15288、また国内の基準案として、「共通フレーム98」などもあり、参考となるでしょう。

(b) オブジェクト指向開発

オブジェクト指向は、現在では、広範な開発基盤が提供されているため、特別な技術選択ではありません。しかし、ソフトウェア開発の歴史の中では、それは特定のプログラミングスタイル、開発スタイルでした。開発におけるオブジェクト指向の採用は、プロセスの姿に独特の色付けを与えます。ソフトウェアアーキテクチャ設計のパターン化、いろいろな開発プロセスでの部品の再利用、テスト手法、分析・設計・実装のシームレスな結合、そしてインCREMENTALな開発方法などです。

オブジェクト指向開発の流れの中で、「統一プロセス」(Unified Process) のような、プロセスの組み立てのパターンが提唱されています。

(c) 形式的仕様記述言語

形式的仕様記述言語は、現在までのところ特定の開発分野でしか実用されていません。しかしその手法は、開発プロセスのいくつかの課題を手法自体の力で解決します。ソフトウェアが与えられた仕様を正確に実現することを保証します。

(d) 要求工学

ソフトウェア開発の出発点ともいえる要件の引き出し、明確化、変更管理などについて多くの手法が提案・実施されています。ソフ

トウェア開発では、要求に係るトラブルが多いため、この技術は重要です。

(e) レビューとテスト技法

レビューとインスペクションは、多くのプロセスで品質向上の仕組みとして利用されます。これらの実施方法には、経験と理論に基づいて、いくつかの手法が提案されています。たとえばレビューの公式の進行役を置くといったことが提案されています。

ソフトウェアプロダクトとプロセスの中間生産物のテスト手法、またIV&V（第三者によるテスト手法）についても多くの手法が提案実施されています。

■ その他の観点

支援ツールや手法だけでなく、人の活動や知識、設備の活用もプロセスを実施するうえでは重要なファクタとなります。

(a) 実施チームの考え方

ソフトウェア開発チームの合意形成に主眼を置いたプロセス改善のアプローチにSCRUM（スクラム）アプローチがあります。これは元をたどれば一橋大学の野中郁次郎・竹内弘高氏^{注12}により提唱された一般的な開発管理に関する考え方ですが、1995年頃、米国のKen Schwaber氏によってソフトウェア開発への適用が提唱されました、Sprint（全力疾走）と呼ぶ短いタイムボックス（一定の時間間隔、典型的には1カ月程度）を数回繰り返し、製品を完成することを目指します。各Sprintの間、発注者は口出しをせず、開発チー

注12. “The New Product Development Game”, Harvard Business Review (1986)

ムが結論を出すのを待っているというスタイルを提唱しています。また開発チームの基本的な合意形成メカニズムとしてSCRUMミーティングというものを提唱しています。

また、CMMのカルチャの中からは、TSP（チームソフトウェアプロセス）というチーム活動を基本とした手法も提唱されています。

(b) 人材のプロセススキルと個人プロセス

プロセスを首尾よく推進するためには、当然プロセスに関するエンジニアリング的な観点をわきまえた人材が必要です。そのためのスキルの育成は、ソフトウェアエンジニアリングの知識と実施スキルが車の両輪となるように系統的に実施する必要があります。

個人のプロセスの実施を統計的なツールも駆使して実施するためのアプローチがCMMのカルチャからはPSP（パーソナルソフトウェアプロセス）として提唱されています。

(c) 協働（CSCW）およびコミュニケーションツール

ソフトウェアプロセスにおける非常に重要な要素として、特に「コミュニケーション」をとりあげるアプローチもあります。そして、コミュニケーションを通じた協働作業を推進することが有効だとするアプローチも提唱されました。これをCSCW（Computer Supported Collaborative Work：協働）と呼びますが、そのためのソフトウェアツールなども開発されました。

この流れは、現在WiKiのようなツールや商用のグループウェアの活用として発展しています。

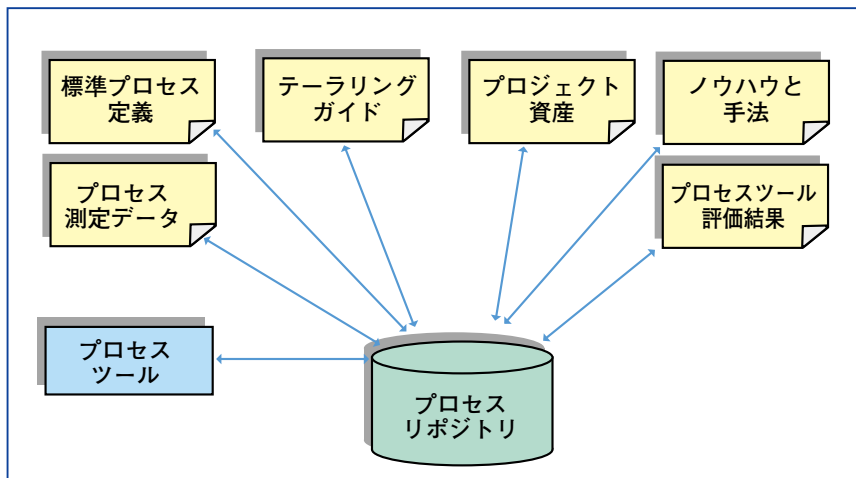


図2-6 プロセスリポジトリ

(d) プロセスリポジトリ

プロセスの重要性を抽象的に述べても精神論に終わるかもしれません。プロセスの活用を実質的に保証するためには、プロセスに関する知識やその活用方法を個人の能力や好みに依存するのではなく、リソースと知識のデータベースとして構築すべきであるということを強調するアプローチもあります。このようなデータベースをプロセスリポジトリといいます。プロセスリポジトリには標準プロセス（複数の選択肢を含む）、プロセス測定データ、プロセス支援ツールとその使用ノウハウ、各種のプロジェクト資産などを含むことがあります。

(e) ハードウェアとネットワーク（諸設備・環境）

プロセスを支援するハードウェアおよびネットワーク設備などの能力の重要性は述べるまでもありません。1980年代には、ハードウェア、CASEツール、ネットワーク設備などで開発環境を強化する

ことを「装備強化」「武装化」といった用語で表現したこともありました。

日本では、1980年代に、シグマ（ Σ ）プロジェクトがそのような装備強化の方向性もはらんでおこなわれました。

2.4 事業目標とプロセスのニーズ

プロセスの改善というキャッチフレーズは、しばしば改善のための特定の仕組みや作業の規範的なモデルの強制と受け取られ、一部のマネジメントの立場の人からも敬遠されてきました。しかしプロセス改善はビジネスとしてソフトウェアを供給するすべての組織にとって、またすべてのプロフェッショナルなソフトウェア開発にとって本来的に必要なものです。なぜなら、プロセス改善の目標は、業務としておこなわれるソフトウェアの開発・保守・運用をより適切なものに変えて事業目標を達成することそのものだからです。

■ 開発組織に求められる本質的な事業目標

ソフトウェア開発・保守・運用組織は、そのビジネスの基本的な存在根拠を、「社会的に有用で信頼にたるプロダクトとしてソフトウェア／システムを供給する」という点に置いています。単に顧客の目の前の要求にこたえるばかりでなく、社会的に有用で信頼性のある成果を求められているのです。ところが、ソフトウェア開発の現場は、最終利用者の運用局面と実質的にも精神的にも離れていることがしばしば見られます。プロフェッショナルなソフトウェアの開発が定着した組織では、常に最終利用者の利用において真価を発揮するような作業を目指します。ソフトウェアプロセスが目指すものは、最終的にはそのようなプロフェッショナルな開発者としての

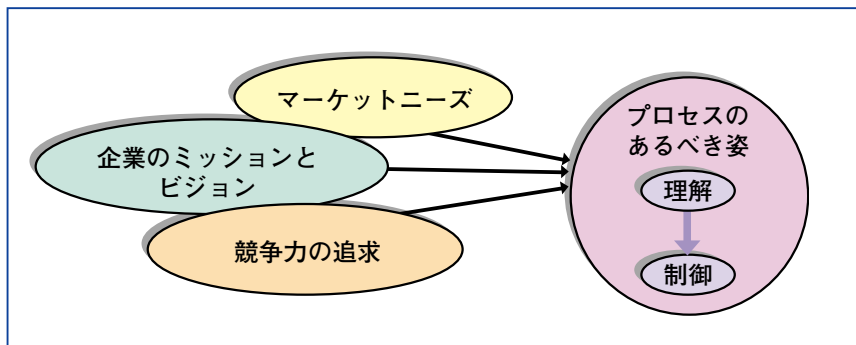


図2-7 事業目標とプロセス

目標を背景として持っています。現場のエンジニアは、エンジニアのプライドとして、そのことを追求します。こうしたことが、ソフトウェア組織が本来的に持っている事業目標です。

また、それぞれの企業はマーケットでの優位性を獲得するためにビジネスプロセスの特色を事業目標として持っています。ソフトウェアプロセス改善は、優位性を導くビジネスプロセスの品質を確保するための手段です。

■ 事業目標からプロセスに対する要求の筋道

企業の戦略に基づく事業目標を実現するには、多くの要素が必要です。ソフトウェア組織では、ソフトウェア開発・運用のいろいろな面でプロセスの能力を向上していくことが企業の能力を高める重要な要素となります。能力を高めることが結果としてプロセスの良好な実績を生み出します。組織は何を改善すれば持続可能な能力向上を果たし、その結果としての安定的なプロダクト供給と業績向上を生み出すのか判断しなければいけません。

それぞれのソフトウェアプロセスは、ソフトウェア組織が持つ歴

史的な過程で、経験的に生み出してきた必要作業のノウハウから成っています。また、これからのグローバルな競争環境での優位性を持つプロセスの保持も重要です。それらを企業業績を高めるため、全体としてどのような方向に進化させながら運営していくのが重要な考慮点であることはいうまでもありません。

組織内に分割されたプロセスがあることを前提とすると、それぞれのプロセスがそれぞれの特性において具体的にどのように事業目標実現に寄与するのか、またそれぞれのプロセスが事業目標から見てなぜ、どのように必要なのか、を明確にしなければいけません。このようにしてプロセスのニーズ、プロセスに求められるものが詳細に理解されていきます。

企業活動の中で、それぞれの企業が出合う自社の強み・弱みに対して、どのように強みを伸ばし、弱みをなくしていくのかを判断しなければいけません。そのためには、プロセスの現実をより深く知ることに努め、またプロセス改善の目標と手段を明確にすることによって事業目標達成に近づくことができます。

■ 取得者の業務プロセスとソフトウェアプロセス

ITがサポートする企業システムでは、ITの開発・運用のプロセスは、本来の業務プロセスと密接に関連しています。ソフトウェアプロセスは、全体としてみれば、このような取得者（発注者・利用者）の業務プロセスとその改善の一環だという性格も持っています。ソフトウェアプロセスの改善は、取得者の業務プロセス全般の改善の一環でもあるわけです。特に取得者の「ソフトウェア／システムを調達するプロセス」の基本的な改善の問題を含んでいるのです。単に供給者（ベンダ）のプロセスの改善の問題だけではないのです。

2.5 サプライチェーンとソフトウェアプロセス

ソフトウェアの供給は、現在、システムインテグレータの業務、個別ソフトウェアの開発業務、共通コンポーネントの開発といったいろいろな特色ある企業の業務の成果の組み合わせでおこなわれることが多いといえるでしょう。また、組込みソフトウェアでは、部品となる装置・機器のソフトウェアが全体システムのコンポーネントとして働き、システムの組み立てではそれらの総合的なQCDを問題とすべき場合もあるでしょう。あるユーザ企業がソフトウェア／システムの開発・運用を外部に委託するという場合にも、その委託先の企業が1社で担当するとは限りません。したがって、ソフトウェアライフサイクルプロセスは1つの企業が各種のプロセス群の全体を実施するものというはまれで、多くの企業の作業や製品の組み合わせとしておこなわれることが多いと見たほうがよいのです。

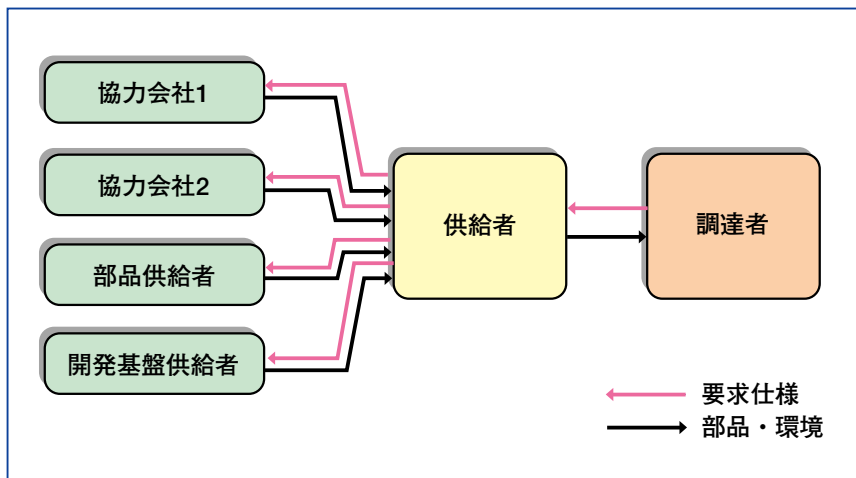


図2-8 サプライチェーンとソフトウェアプロセス

このような組み合わせの問題は、コンソーシアムや元請・下請の問題として述べられることもあります。本書ではソフトウェアコンポーネントやサービスのサプライチェーンにおけるプロセスの問題ととらえて見ましょう。

サプライの上流には「部品」であるソフトウェアコンポーネントがあります。プログラム成果物やそれらの統合されたシステムは、サプライチェーンの上流から下流に受け渡されていきますが、要求仕様や、設計といった要素は部分的には下流から上流に流れ、またそれぞれのチェーンの環で独自に獲得されることもあります。これらの関係を図2-8に表してみました。

■ サプライチェーンの改善アプローチ

このような複数の企業にまたがる複雑なプロセスの改善の問題にはどう対処したらいいでしょう。次の2つのアプローチがあります。

● 協調的アプローチ

全体を統合的なプロセスと見て、統合的なプロセスマネジメントを展開し、その制御下に置きます。このためには、個々の企業が統合されるプロセスの構造に合意するだけでなく、それに習熟する必要もあります。また運用に当たっての緊密な情報交換も必要です。

● 独立的アプローチ

それぞれを、独立で責任ある組織のプロセスとしてみて、組織間のインタフェースとなる調達・供給プロセスの果たすべき役割を強調します。

いずれの場合にも、複雑なプロセスマネジメント問題に直面することには変わりはありません。

■ サプライチェーンのプロセスの課題とヒント

こうしたプロセスの複雑さに対応するためには、実務的には取得側のプロセスが相当しっかりしたものである必要があります。しかし、実際には調達プロセスに基礎的な問題がある場合が多く見られます。よく見られる課題とヒントを列挙してみましょう。

- 主供給者はプロセスの管理をおこなっているが、協力会社ではほとんどプロセス管理がおこなわれていない。しばしば協力会社は小規模であり、品質管理など独自のプロセス管理の仕組みをそもそも持っていない（プロセス管理全般の不完全）。
- 協力会社のソフトウェア開発に従事するエンジニアが派遣的な環境・体質で業務に従事しているため、所属する企業が固有のプロセスを元来持っていない。また、主供給者も適切な指導や訓練を提供していない（プロセス管理と教育プロセスの不完全）。
- 供給・調達のプロセスが不明確で、供給者・取得者間のいろいろなインタフェースが不明確となり、品質上の問題とともに非効率である（供給・取得プロセスの不完全）。
- サプライチェーンで受け渡されるソフトウェアコンポーネントや環境についての品質基準や品質表示がないか非常に不十分である（供給・取得プロセス、品質管理プロセスの不完全）。
- 最終的な用途・性能要求や運用環境が明示されることなくソ

ソフトウェアコンポーネントや環境が調達される（要求関連プロセスの不完全）。

システム運用企業において多くの部門の要求を調整しなければいけない場合も、要求引き出し・確立・管理のプロセスの確立が必要となります。このような場合は、多くの要素を統合するという作業はサプライチェーンの上流から下流に向かってではなく、下流から上流に向かっておこなわれなければいけません。

2.6 モデルベース改善とその他のアプローチ

ここでは、改善を目指すソフトウェア開発・保守・運用組織にとってどのような改善アプローチが存在しているかを見てみましょう。

■ モデルベース改善

このアプローチは、ソフトウェア（システム）開発・運用のプロセスの基本となるプロセスの参照モデルと、それらのプロセスが組織内で運用される際のプロセスの能力や成熟度に関する評価尺度を使います。このことをモデルベースの診断・改善と言います。国際規格15504は、そうした枠組みを標準として定めたものです。国際規格15504のモデルベース診断としての特色は、プロセスの選択に関しての自由度を大幅に認めた点でしょう。改善目標となるプロセスの能力水準は目標プロファイルと呼ばれ、改善の依頼者が自社のビジネスモデル、ビジネス目標に照らして設定することができます。

モデルベース改善で基本となるモデルは、プロセス参照モデルと、プロセスの能力・成熟度の評価尺度とからなるプロセスアセスメン

トモデルですが、それに基づいて具体的にアセスメントを実施する手順や条件は、アセスメント手法と呼ばれます。これには、アセスメントチームの構成、アセッサ（「3.3 プロセス改善の推進体制」（89ページ））の資格要件、アセスメントに用いるフォームやツール、実施手順、アセスメント成果物の品質要求などを含みます。またアセスメント手法の実施をプロセスとして見た場合には、特にアセスメントプロセスと呼びます。

モデルベースのプロセス改善は、アセスメントの実施をCAPDoサイクルにおけるCheckフェーズの具体化とみなして改善サイクルをまわします。その際に、国際規格15504 第4部では、目標プロセス能力とのギャップ分析を有力なアセスメント結果の分析手法として提示しています。

■ その他のアプローチ

プロセス改善の推進において、評価方法の1つである前述のモデルベース改善は特別な位置を占めています。それは、継続的な改善努力そのものを支援するというスタンスがあるからです。しかし、プロセスの改善を正面から目指したアプローチは、必ずしもそうしたアプローチだけではありません。

たとえば、GQM（Goal-Question-Metric）^{注13}というアプローチがあります。GQMは、「2.4 事業目標とプロセスのニーズ」（51ページ）の節で述べたように、ソフトウェア組織の事業目標に照らして、それぞれのプロセスで目指すべきものが何かということをブレイク

注13. GQM: Basili, V., and D. Weiss, "A Methodology for Collecting Valid Software Engineering Data." IEEE Transactions on Software Engineering 198; SE-10 (6) : 728-738. V. Basili, Applying the Goal/Question/Metric Paradigm in the Experience Factory, Presented at the 10th Annual CSR Workshop in Amsterdam, 1993.

ダウンしていきます。ソフトウェア組織が抱える問題状況を、事業目標に照らして分析し、それらを解決するにはどうすればよいかという改善の道筋を見定めます。そして、解決策を実施し評価していく際の評価・測定尺度を定めていきます。

このようなGQM手法は、モデルベース改善における測定の考え方と組み合わせることも可能であり、実際にそのような提案がいくつかおこなわれています。また、GQMのアプローチをより一般化して展開するアプローチとして、エンピリカルソフトウェア工学^{注14}アプローチがあります。エンピリカルソフトウェア工学では実証性という視点を強調してソフトウェア工学を推進しています。

次に、XP (Extreme Programming)^{注15}アプローチに代表される、アジャイルプロセスモデルとして分類されるアプローチがあります。これらの考え方は、多くは反復的な開発ライフサイクルモデルであり、ライフサイクルプロセスの具体的な組み立て方を提示するものです。モデルベース改善の立場から見ると、これらの提案は、具体的なプロセスの組み立て方の提案となっているので、モデルベースの考え方と矛盾なくこれらのアプローチを1つの手法として選択することが可能です。

アジャイルプロセスモデルの1つと言われることもあるRUP (Rational Unified Process)^{注16}のようなアプローチもあります。これもオブジェクト指向の立場に立った、特定のプロセスの組み立て方

注14. V. Basili, "The Experimental Software Engineering Group: A Perspective", ICSE '00 award presentation, Limerick, Ireland, June 5-10, 2000.

注15. XP: Beck, K. 2000, 「XP エクストリーム・プログラミング入門——ソフトウェア開発の究極の手法」長瀬嘉秀監訳、飯塚麻理香・永田渉訳、ピアソンエデュケーション刊

注16. Rational Corporation 2001.

の提案であると見ることができると、XPと同様にモデルベースの考え方と矛盾なく選択することが可能です。

プロセス改善ということを語るときにもう1つ忘れてはならないアプローチに、プロセスプログラミングというアプローチがあります。これは、80年代後半にオスターワイル^{注17}が提唱して話題となって以来の1つの研究の流れであり、ソフトウェアの開発プロセスを形式的な記述言語で表現し、モデル化する試みです。

■ 「スマートな改善」

以上のように、いろいろなアプローチがあることを紹介してきましたが、最初に述べたように、本来、それぞれの企業の特性を生かしたプロセスの改善ということが重要です。改善のモデルはそれに対する考え方のガイドラインとしての役割を持つに過ぎません。本書では、各企業が選択する改善アプローチを「スマートな改善」と呼びます。スマートな改善では、たとえば次のような着眼点が重要です。

● 事業目標の重視

- ・ 自社組織での具体的なプロセス展開の合理的な部分と改善すべき部分の識別を含む

● サプライチェーンの重視

- ・ システムや製品が多く、供給者の連携で構築・開発されることに対応し改善を進める

注17. Osterweil, L., "Software Processes Are Software Too," Proc. 9th International Conference on Software Engineering, IEEE, March 1987, pp.2-13.

● **プロセス改善は独善ではできない**

- ・ 広く社会的水準の視点、他社に対する競争優位の視点、国際的な知見を集約する視点、自社技術として蓄積してゆく視点など
- ・ 実証性と客観性を重視し、先入観にとらわれない前向きな態度を養う
- ・ すべての問題を一拳に解決する特効薬を追い求めず、できることを着実に実行し技術を蓄積する
- ・ 技術革新への継続的なチャレンジを忘れない

● **エンジニアの仕事を快適・円滑にする**

- ・ 規律の押し付けではなく、現場での創意工夫と現実を見る目

● **プロダクトへのこだわりを忘れない**

- ・ 技術者モラルと最終消費者の目線を見据えた、プロダクトに対するプロフェッショナルな愛着

第3章 プロセス改善の留意点と実施体制

第1章、第2章ではプロセスを改善することについてのポイントを説明しました。そこで、本章では、具体的にプロセス改善をするうえで留意すべき点およびプロセス改善を実施するときの体制について説明します。

3.1 プロセス改善推進の留意点

プロセス改善推進者は、プロセスをより良いものにしていくために、次の3つのことをよく認識する必要があります。

■ ソフトウェア開発・保守・運用を支える三要素

まず、大切なのは、ソフトウェア開発・保守・運用を支える三要素について十分に理解をすることです。

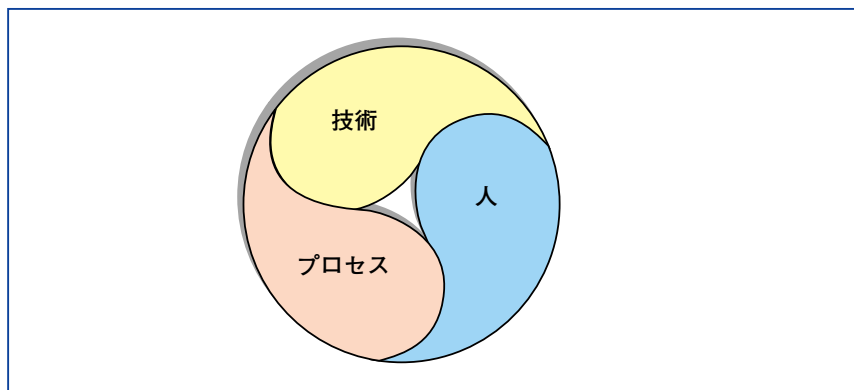


図3-1 ソフトウェア開発・保守・運用を支える三要素

ソフトウェアプロセスが、その目的を達成するためには、以下の3つの要素が、揃っていることが必要です。

● プロセス要素

仕事の組み立て方

● 技術要素

開発支援環境などのプロセスで使用する技術やツール

● 人的要素

そのプロセスを実施する人のスキル、経験、態度

したがって、プロセス改善推進者は、第一に上記の3つの要素をバランス良く考え、プロセスを構築し、改善活動を推進していくことが大切です（図3-1）。

以下、健康診断の採血を例にしてわかりやすく説明します。
健康診断の採血では次のようになります。

● プロセス要素

採血の方法、手順（バンドをする→親指を中に入れて握らせる→針を刺すところを探す→消毒をする、…）。採血の目的を果たすためのさまざまな配慮。

● 技術要素

真空採血管、採血ホルダー、区血帯（バンド）などの採血に使う器具。

● 人的要素

採血の手順・方法を理解し、訓練を積み、採血に必要な能力

(技能) を有する人。

(注：採血をする人には所定の資格が必要)

ここで、もし採血をする人の能力が不足していれば、予定されている人数をこなせませんし、医療事故やトラブルの発生につながります。また、採血の手順や方法が明確に規定されていなければ、人によって手順のバラツキの発生による医療事故の発生につながります。さらに、採血を支援する技術が低ければ、採血の質（安全）を高水準に維持し、高効率なものにすることはできません。

コラム プロセスを実施する人とプロフェッショナルについて

「プロセスを実施する人は、ただそのプロセスを実施するスキルと経験を有している人だけでいいのだろうか？ もし、淡々とノルマを達成するために、無機質的に、ロボットのように作業をするような人だったらどうであろうか？」技能的に採血が非常に上手であることは勿論だが、採血といえども一歩間違えると大変なことになるといったリスクをしっかりと認識し、採血される人の気持ちを考え、適切な指示／声かけとの絶妙のタイミングで採血が進行するといったように、安心感を与えられる人こそが、プロフェッショナルといえるのではないのでしょうか。このことは、我々エンジニアにも通じるものがあるのではないのでしょうか。

つまり、高水準（安全）、高効率な良い採血を実現するには、良いプロセスとそれを支援する技術、そして、それを確実に実施できる人の3つの要素が揃っていて可能となり、1つも欠けてはいけません。

■ プロセス改善のシステム

プロセスをより良いものにしていくために大切なことの第2番目は、プロセス改善システムです。プロセス改善推進者は、ソフトウェアプロセスの維持・改善に当たり、図3-2 (b) に示すようなフィードバック系のプロセス改善の仕組みを構築する必要があります。

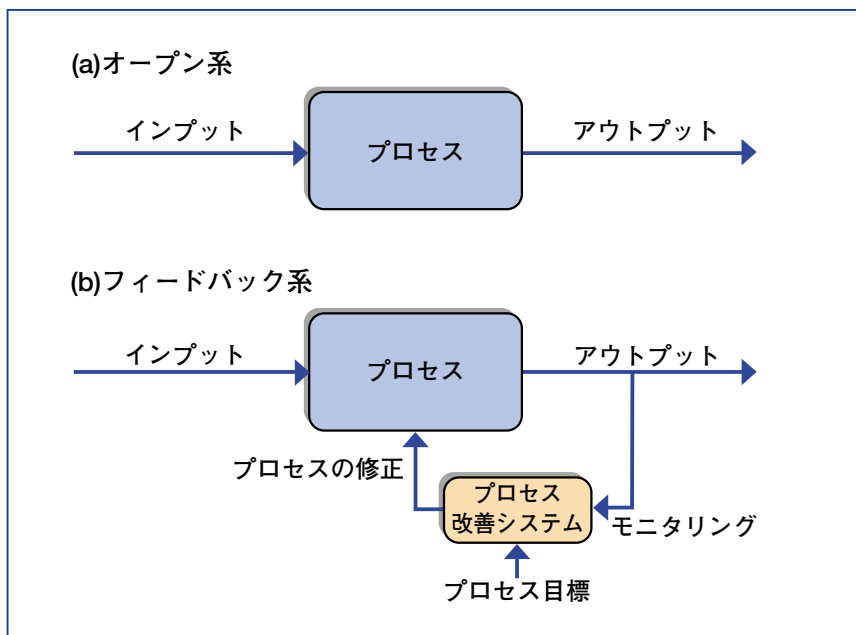


図3-2 プロセス改善システムの基本的な仕組み

図3-2 (a) に示すようなオープン系のプロセス改善の仕組みでは、プロセスはプロセスの性能不足や環境変化などにより、当初期待していた機能が発揮できなくなります（プロセス目標とプロセスのアウトプットとのオフセット（差）が生じる）。このままでは、プロセスは当初の目的を十分に発揮することはできません。そこで、図3-2 (b) にあるように、プロセスのアウトプットが所定のプロセス目標を達成しているかについてのモニタリングをおこない、プロセスの目的が未達であれば、プロセスの修正をしていくためのフィードバック系の仕組みを作り、タイムリーかつ継続的にプロセスを修正（チューニング）していくことが大切です。

■ コミュニケーションと知識活用

システムの高度複雑化、大規模化により、多くの部署（ライン／スタッフ）が協力しあって製品を開発しなければならない状況においては、関係者の相互理解と認識の共有が大切です。そのためには、第3番目としてコミュニケーションを円滑にする仕組みを構築する必要があると考えます。

改善の第一歩は、プロセスにあれこれと手を加えることなく、コミュニケーションの円滑化を図ることです。それは、改善は組織や仕組みを変えるだけでなく、人の意識も変える必要があるからです。

プロセス改善を実施していくうえで、コミュニケーションが円滑で、組織やチームがうまく機能していることが重要です。なぜならば、コミュニケーションができている職場とは、「問題を共有できる職場であること。そして、知識を確認し積極的に活用し改善を推進することができる職場である」と言えるからです。

さて、コミュニケーションが円滑な組織とするための基本は何でしょうか。これは、一般的に“報・連・相^{注18}（ハウレンソウと呼ぶ：“報告”、“連絡”、“相談”の頭をとったもの）”が確実にできることです。

プロジェクトはチームで動きます。チーム内には縦の関係（上司一部下）と横の関係（同僚や関連組織間）の2つの関係があります。“報・連・相”により、この縦と横のコミュニケーションを円滑にし、情報共有（知識と経験の交流・確認・蓄積）ができるチーム作り、風土作りをしましょう。

そして、もう1つ大切なことは、双方向コミュニケーションであるということです（図3-3）。“報・連・相”を実施しても、話し手、

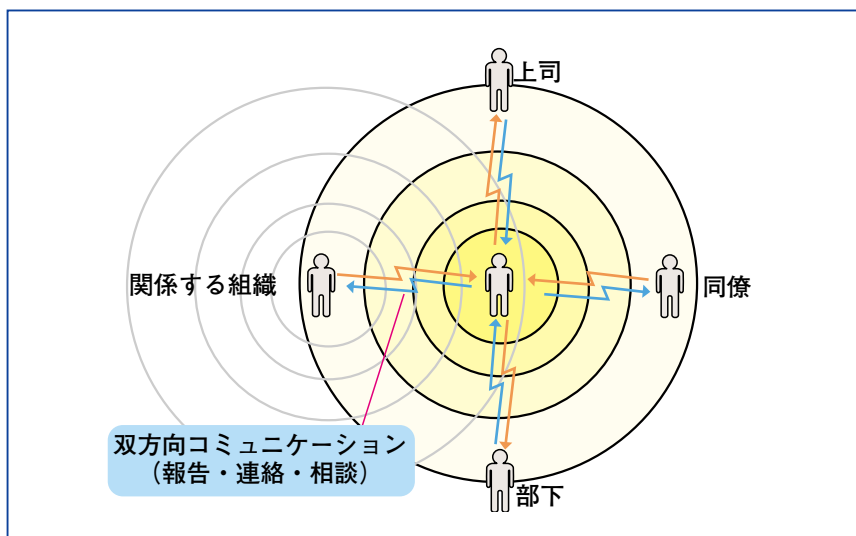


図3-3 報告・連絡・相談による双方向コミュニケーション

注18. 参考文献：山崎富次，“ほうれんそうが会社を強くする”，株式会社ごま書房，1986.

聞き手が、“聞いた”、“話した”というだけでは、コミュニケーションは一方的なものになってしまいます。上司、部下、関係者のすべての人が、闊達にお互いの意見を言いあい、聞きあい、確認しあう、双方向コミュニケーションにより情報を共有し、有効に活用しましょう。

また、業務を進めていくうえでの最重要な関係者は、顧客とエンドユーザであることをよく認識しておきましょう。仕事は社内関係者だけでは進められません。社内関係者だけでなく、顧客を含めた社外関係者とのコミュニケーションを円滑にし、正確な意思の疎通を図り、顧客と協調して課題解決を図っていく姿勢を持つことは大切なことです。

この“報・連・相”がしっかりできる組織風土作りをするためには、以下の(a)～(d)の4つの例にあるような仕組み(形)を作ると良いでしょう。これらは決して新しいことではなく、昔から言われている当たり前のことです。この当たり前のことが、当たり前にできるようになることが大切です。

(a) チーム内および関係者間の協働意識の醸成

現在、製品に要求される機能の多機能化、高度複雑化、大容量化が進み、ソフトウェアを1人で完結するということがあまり見られなくなり、チーム内外の関係者との分業、連携があって成り立つソフト開発が大勢を占めるようになっていきます。このような状況の中では、次のような意識をチームメンバ全員が持ち、情報をオープンにするカルチャを作ることが大切です。

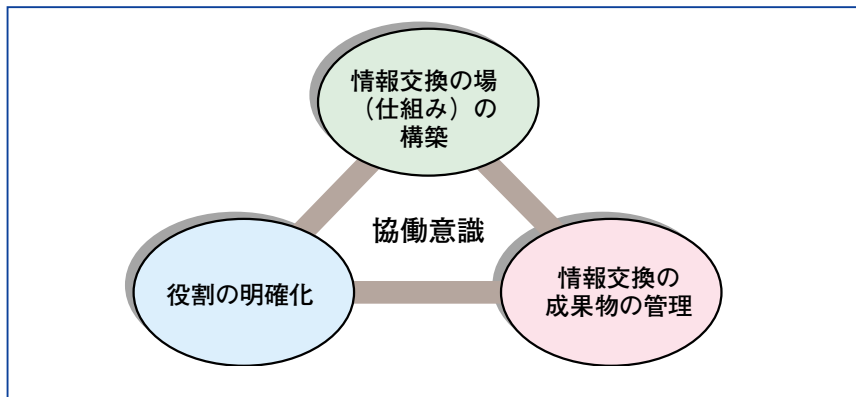
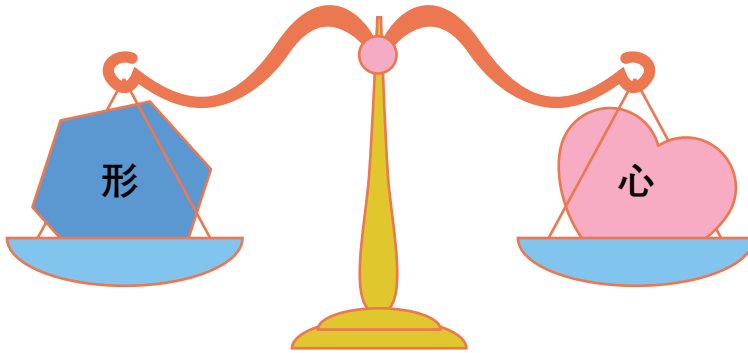


図3-4 協働意識の醸成

- チームメンバ、関係者が協力し合って仕事をするという協働意識を持つこと
- 情報発信を活発にし、情報をオープンにすること
- コミュニケーションは、未来の自分自身に対する情報伝達でもあると意識すること

(b) チーム内および関係者間の役割の明確化

協働意識という“心”がしっかりといても、うまく協働するためには、必要最小限の仕組みという“形”が必要です。この仕組みというのが、役割の明確化です。役割の明確化は、チームおよび関係者が目的達成に向けて円滑に協働するための基本です。チーム内および関係者間の役割が不明確であれば、業務の重複や抜けが発生し、必ずしも効率の良いものにはなりません。責任の所在もあいまいになり、製品品質にも大きく影響が出るでしょう。



“心”と“形”のバランスが大切

(c) チーム内および関係者間の情報交換方法の定義（ルール化）

チームや関係者が同じ目的を持ち目標を達成するのに、もう1つ重要な仕組み（形）があります。それは、チームや関係者間の情報交換の場（仕組み）の構築です。協働意識という“心”がしっかりしていたとしても、業務多忙またはトラブル処理にかかりっきりになってしまうと、目先がそれに集中してしまい、関係者とのコミュニケーションがおろそかになりがちです。そこで、定期的な進捗会議や異常発生時の会議の開催方法などを規定し、確実に運用することが大切になります。このように最初の短時間の面倒を乗り越えて、コミュニケーションをルーチンワーク化し、「報・連・相」を挨拶と同じような当たり前の作業にすることが大切です。

コラム 会議の持ち方について

会議の開催方法で大切なことは、会議の目的をしっかりと明記して会議開催案内を出すことです。突然招集した会議でも、会議の目的を文書化したものを用意するか、ホワイトボードに書くことです。会議の目的が明示されていない会議は、出席者全員のベクトルがズレたり、何を議論すればいいのかわからなくなったりして、無駄な会議になりやすいのです。また、会議の議事録はすぐに出しましょう。後回しにしたとたん、たまっていきます。また、議事録の確認遅れや関係者への情報共有の遅れが、仕事の進捗に影響することにもつながります。また、会議の結論が何なのかははっきりさせましょう。そしてその結論を議事録に明記します。

(d) チーム内および関係者間のコミュニケーション成果物の管理（ミーティング管理、議事録管理、アクションアイテム管理）

情報が口頭で連絡されるなどして、各自の頭の中に存在するだけの状態や、メールや個人メモが残っているだけの状態では、展開された情報は一過性のものになってしまい、情報に含まれている知識やノウハウが蓄積されません。また、有効に再利用されることはありません。

つまり、いくら良いコミュニケーションの意識（心）と仕組み

(形)があっても、コミュニケーションによって共有された情報を、見える化し、整理し、体系的に残しておくことが大切です。そのためには、文書を管理するための手順・方法を明確にしておくことが必要となります。

コラム 情報の見える化について

やりとりする情報の見える化（やりとりした情報を文章化、図表化）により、情報を相手に伝えることの確実性を増すことが大切です。頭の中にあるだけでは、確実な情報の共有は難しいのです。見える化しながら（図表で表現しながら）会議を進めるのも有効（理解しやすい、リアルタイムに確認しやすい、誤りを見つけやすい）です。記述することによる気付きの効果も期待できます。言葉だけのコミュニケーションにおける、思い込みや勘違い、“言った／言わない”といった問題の発生をなくすことが期待できます。

3.2 プロセス改善推進の知恵

組織や仕組みにあれこれと手を加えるだけの改善をしていませんか？

プロセス改善は、組織や仕組みを変えることだけではなく、そこで働く人の意識を変えることも大切です。

そこで、プロセス改善を推進するうえでの知恵はたくさんありますが、代表的な8つについて解説をします。皆さんも自分の組織に当てはめてチェックしてみてください。

プロセス改善推進のための知恵

- すべての改善関係者が“誰が主役のプロセス改善なのか”を理解していますか？
- プロセス改善活動を実施する現場のメンバ全員の合意がとれていますか？
- 現場で実際に実施しているプロセスを明確にしていますか？
- 性急または大規模にプロセス改善を進めないようにしていますか？
- 危機的なプロジェクトを抱える現場を、いきなりプロセス改善で救うのではなく、別の手を打っていますか？
- プロセス特性が違う組織に対し、十把一絡げに標準プロセスを適用しないようにしていますか？
- プロセスオーナーやプロセス改善支援スタッフが、積極的に現場に足を運んでいますか？
- 現場、関係者にスペシャリストとしての自覚を持たせていますか？

(a) すべての改善関係者が“誰が主役のプロセス改善なのか”を理解していますか？

プロセス改善推進者や改善活動を実施する現場のメンバが、「プロセス改善は誰のために実施する改善活動なのか」を忘れてしまったら、改善活動はうまくいきません。改善活動を実施する現場のメンバは、「自分たちの手で、自分たちが主体となって実施する活動である」と認識し、プロセス改善推進者や改善支援スタッフは、「現場の人たち自身が実施する改善活動を支援する」ことであると認識することが大切です。

以下に、改善活動がうまく進展しない顕著なケースを例示しました。

例1 プロセス改善推進者／プロセス改善支援スタッフへの一方的依存

現場の人たちが「プロセス改善推進者／プロセス改善支援スタッフが改善活動をやってくれるので、その指示に従えばいい」と改善活動を受動的にとらえてしまっていると、プロセス改善活動は思うように進展しません。また、プロセス改善支援スタッフに十分な対応力がない場合、プロセス改善活動自体が破たんする可能性があります。

このような状況下では、現場の人たちに対して、「うまくプロセス改善支援スタッフの協力・支援を受けながら、自分たちが主体となってプロセス改善活動を推進していくこと」を理解してもらう必要があります。つまり、現場の人たちが「プロセス改善は自分たちが主導して進める、自分たちのための活動である」と認識してあげることが、プロセス改善活動の成功の鍵となります。

例2 プロセス改善推進者／プロセス改善支援スタッフによる一方的推進

プロセス改善推進者／プロセス改善支援スタッフが、「私たちの言うとおりプロセス改善をすればいいんだ」といったように、現場の状況や現場の意見を十分に把握しないで、プロセス改善活動を推進してしまうと、現場が言うことを聞かなくなるか、前ページの「例1」で例示したようなプロセス改善推進者／プロセス改善支援スタッフへの一方的依存状態に陥ってしまいます。

こうなると、プロセス改善活動がうまく進まなくなってしまうので、プロセス改善支援スタッフは、現場の意見や要望をよく引き出し、現場の主体性を維持しつつ、適切な支援をしていくことが必要となります。

(b) プロセス改善活動を実施する現場のメンバ全員の合意がとれていますか？

現場のことを一番よく知っているのは現場です。現場が抱えている問題点を洗い出すなど、プロセス改善を成功させるためには、メンバ全員の協力が必要となります。

そこで、プロセス改善推進者は、現場に対して、「プロセス改善は現場が主体的に、自分たちのために実施するものである」ことを、メンバ全員に理解・納得してもらえるように働きかけましょう。必ずしも全員の納得が得られるとは限りませんが、現場の人たちの意見をよく聞き、根気よく話し合いをすることが大切です。プロセス推進改善者自身が真剣に現場のことを思っているのなら、だんだん賛同者は増えていくはずです。

図3-5は、個人ベクトルと組織目標ベクトルとの関係を示してい

ます。個人ベクトルの大きさは個人の保有能力を、方向は個人の志向を示しています。また、個人ベクトルの組織目標ベクトルへの分解成分が、組織目標に対しての発揮能力を示します。この図から、個人の志向を組織目標に一致させることが、組織目標に対するメンバの発揮能力を最大化することがわかります。（個人ごとに、個人ベクトルと組織目標ベクトル間の角度の開きが違います。組織目標に対するモチベーションが低下しても、角度の開きは広くなります。ひどい場合は開き方が 90° を超え、発揮能力が負になる領域の人も出るかもしれません。）

つまり、メンバの理解・納得を得ることが、メンバの発揮能力の最大化につながるのです。プロセス改善推進者は以下の2項目について十分留意して、現場主導のプロセス改善活動を展開していただく。

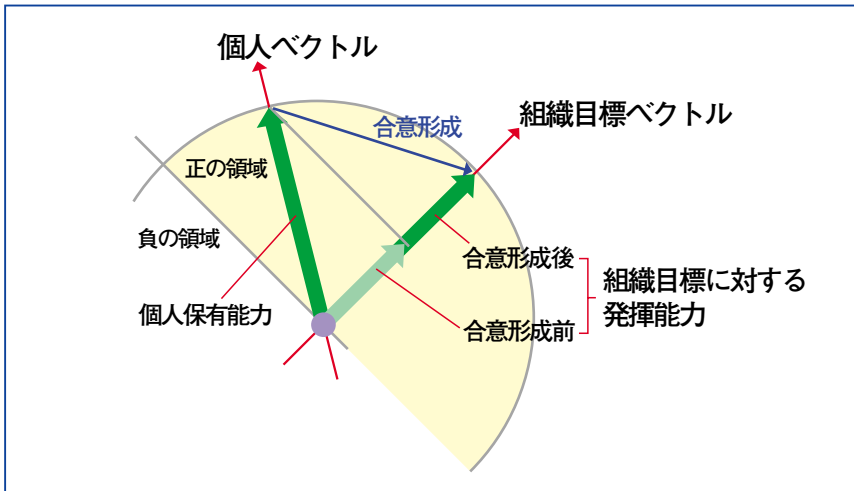


図3-5 合意形成による発揮能力の引き出し

- ソフトウェア開発現場の人たちが楽にならないプロセス改善は長続きしない。
(例：プロセス改善をしたら、逆に仕事が辛くなった)
- プロセス改善活動を一方的に押しつけない。
(一方的な押しつけは、押しつけられた側のモチベーションを下げるにつながります)

また、プロセス改善を成功させるためには、現場とプロセスオーナーとの間で合意を形成し、管理者と現場のメンバの両者の改善意識を高くすることが必要です。最終的には管理者にプロセス改善活動のよきスポンサになってもらうことが大切です。この合意形成を図るためには、ボトムアップアプローチとトップダウンアプローチの2つのアプローチがあります (図3-6)。

- (1) ボトムアップアプローチ (意欲のある人が現場から出て、改善活動を引っ張ってスタートする場合)

上司の理解活動を展開し、プロセス改善を組織的な活動として認知してもらうものです。前者のケースについては組織的な活動として認知されていないので、手弁当的な活動になりがちで、活動の初期は良いとしても、大きな改善効果を得るためには、ある段階で、組織的な活動として認知してもらう必要があります。

- (2) トップダウンアプローチ (トップあるいはプロセスオーナーが改善の必要性を認識して、組織的活動として現場にプロセス改善活動を指示することによってスタートする場合)
このケースはメンバへの理解活動を展開し、プロセス改善を組織

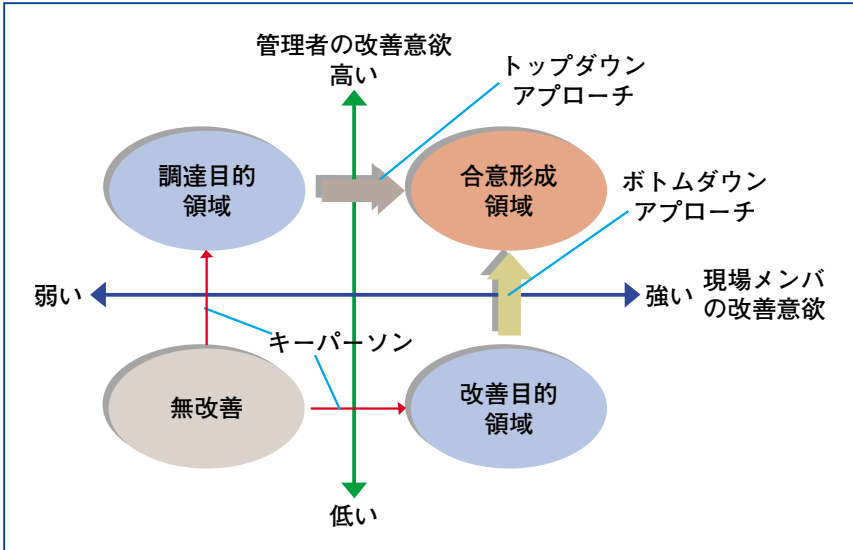


図3-6 トップダウンアプローチとボトムアップアプローチ

的な活動として推進するものです。組織的な活動と認知されているので、プロセスオーナーから必要なリソースを得られる可能性は高いのですが、プロセス改善活動に対する現場の理解が得られなければ、プロセス改善活動はうまく進展しません。管理者の改善意欲は、調達目的を動機として高くなることが多いので、これをリソースを確保するための合意形成のチャンスとすることも1つの方法です。

(c) 現場で実際に実施しているプロセスを明確にしていますか？

プロセス改善活動の対象となる現場が実際に実施している手順／方法をプロセスとして見える化することが大切です。プロセスの見える化は、図3-7に示すように、以下のステップでおこなわれます。

- 不明確な状態になっている一連の仕事のある適切な粒度のプロセス（処理・作業）に区切る
- 各プロセスの目的とプロセス（処理・作業）のインプットとアウトプットを定義する。
- プロセス間の作業生産物の流れをつかみ、プロセス間のつながり（インタフェース）を明確化する。

この見える化が十分できていると、次のようなメリットが得られます。

- 管理しやすい（見積り、進捗がわかりやすくなる）
- 改善しやすい（無駄がわかる、良い点／悪い点がわかる）
- 間違いが減る（確実な作業が繰り返し可能となる）
- ルールを守りやすい、教えやすい

逆に、見える化が不十分であると、次のようなことになります。

- 管理者が仕事を見通せない。管理／掌握ができない
- 問題が発生しても、原因箇所の特定ができない、速やかな対策が打てない。
- 仕事の内容が作業者によって、ばらつきやすい

なお、プロセスの見える化作業中に、プロセスの強みや弱みなどが見える（気づきがある）ことがよくありますが、これをすぐに改善しようと思わずに、まずは、現行プロセスの現状をしっかりと見える化することに注力しましょう。ここで、見える化作業と問題の対

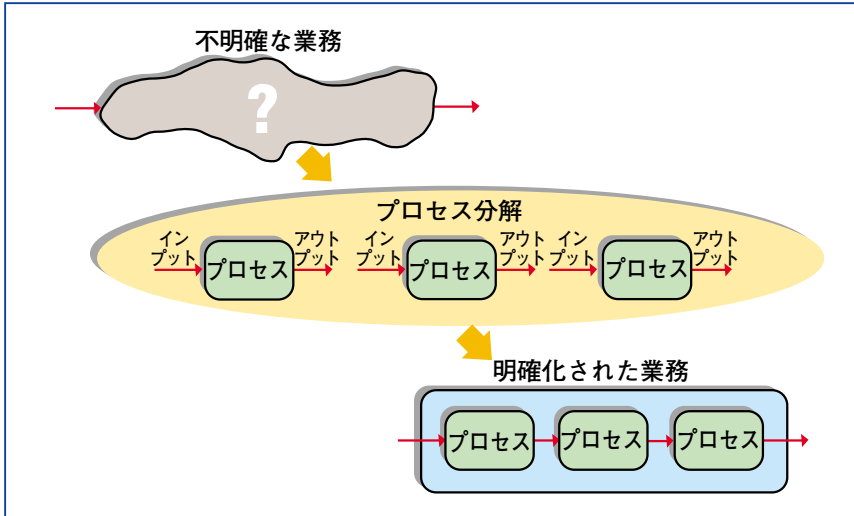


図3-7 プロセスの見える化とは？

策活動を同時に実施すると現場が混乱しやすくなるので、この見える化作業中に見つけた課題は忘れないようにしっかりメモして、後で生かせるようにしておきましょう。プロセスの見える化によってプロセスの認識のバラツキをなくし、その後の改善活動にプロセスの目的とのズレを補正するとよいでしょう。プロセスの見える化をただけでも、見える化のメリットによって、プロセス改善の効果が得られます。

また、プロセスの見える化の際には、以下の2つのポイントについてよく確認しておくことも大切です。

(1) プロセスの目的がしっかり明確化されているかどうか

もしプロセスの目的が不明確であれば、改善自体が意味を持たなくなり、改善の効果が期待できません。

(2) 関連する複数のプロセス間のつながりが明確になっているか
関連する前後のプロセス間の目的（機能・役割）の分割が不適切であれば、単一のプロセスのみを改善しても意味がなく、関連プロセス全体を改善しなくてはなりません。

プロセス改善推進者は、見える化によるメリットをよく認識して、プロセスの見える化の終了後に、ベストプラクティス事例や研究成果の参照により、プロセスへのプラクティスの追加や明確化されたプロセスの煩雑な作業の整理といった改善を実施していくなどの、プロセス改善活動を推進していきましょう。

(d) 性急または大規模にプロセス改善を進めないようにしていますか？

性急または大規模にプロセス改善を進めようとする、プロセスの混乱を招くリスクが高くなります。そうならないための2つのポイントを以下に示します。

(1) 一度に多くのプロセスを変更しない

この際だからと、「あれもこれもと」プロセスの変更や機能追加を一度に多く実施するなど、やりきれない活動にしないことが大切です。

どうしても大きくプロセスを変える必要がある場合は、十分なりソースを確保し、用意周到な事前の準備が必要となります。しかしながら、上司の組織的なプロセス改善の了解が得られたとしても、十分なプロセス改善活動を推進していくための工数が与えられるという保証はありません。改善の規模（レベル）に応じたリソースが確保できなければ、大上段に構えず、簡単な仕組みから着手し、計



性急・大規模なプロセス改善はおこなわない

画的にできることからコツコツと小さな改善の積み重ねを実施していくべきです。

(2) 自分たちの今実施しているプロセスをいきなり別のものに置き換えない

自分たちが今まで実施してきたプロセスは、必ずしも最適ではないかもしれないが、過去からの経験からいろいろと工夫されてきたもので、業務が継続できているのであれば、それなりに妥当なものであるといえるのではないのでしょうか。この今までやってきた仕事のやり方を捨て、ただ良いという理由だけでいきなりほかの仕事のやり方に大きく変更することは、いたずらに混乱を招くだけで、簡単にやれるものではありません。

(e) 危機的なプロジェクトを抱える現場を、いきなりプロセス改善で救うのではなく、別の手を打っていますか？

業務過多や問題が多発し、改善まで手が回らないような非常に厳

しい状況に追い込まれている現場に対して、いきなりプロセス改善を適用して、現場を沈静化させようとする、現場がさらに混乱してしまうことになりかねません。

特に、プロセスアセスメントモデルをそのまま持ってきてこれに置き換えようとしたりすると、現場の混乱はさらに加速してしまうことになるでしょう。

一般的に、混乱している現場を立ち直らせるためには、漢方のような東洋医学的なプロセス改善を適用するのではなく、一部の仕様の実現をあきらめるなどの外科手術のような西洋医学的な手段を用いて、現場を立ち直らせるための一時的な余裕を持たせ、現行のプロジェクトの收拾の目処を立てる（火消しをする）ことが急務となります（アセスメントモデルにある現行プロセスに有効なベストプラクティスを使って事態の收拾をはかる場合もある）。そして、次



まずは消火活動、そして再発防止！
危機的なプロジェクトには、まず外科手術

のプロジェクトでの改善に向けた準備をすることになります。プロセス改善推進者は、このことをよく踏まえて、現場に対してプロセス改善を導入するタイミングをしっかりと見極める目を持つことが大切です。

(f) プロセス特性が違う組織に対し、十把一絡げに標準プロセスを適用しないようにしていますか？

プロセス改善でよく実施される活動に標準化があります。標準化には、人によるバラツキをなくし、無駄な開発を抑制するなどの非常に大きなメリットがありますが、個々の組織またはプロセスの特質を無視し、十把一絡げに同じプロセスを強制すれば、ある現場に対して適切であっても、別の現場では不必要ないしは無駄なものになってしまうことは十分あり得ます。

そこで、組織間で共通に使えるものと、各現場固有のものに区分するといった、柔軟性、多様性を認める必要があります。ただし、何も制約をかけずに、柔軟性、多様性を認めると、固有なものが多くなり、結果的に区分をした意味がなくなってしまうので、適切な区分のルールが必要です。さらに、違いが出る妥当な理由（固有としなければならない理由）を明確にし、固有なものを認めるかどうかを判断する仕組みも必要です。

(g) プロセスオーナーやプロセス改善支援スタッフが、積極的に現場に足を運んでいますか？

管理者や改善支援スタッフが、「現場からの報告が上がってこない！現場はコミュニケーションができていない！」なんて言っていないでしょうか？

これは、管理者や改善支援スタッフが、自席にいることが多く、現場にはあまり行くことがないケースによくあり、コミュニケーションがない原因が現場にあるというよりは、管理者や改善支援スタッフの姿勢に原因があるといえます。問題は上司の席や会議室で起きているのではなく、現場で起きているのです。プロセス改善を牽引していく立場の人は、次のことに留意して、迅速な課題形成と課題解決に努めることが大切です。

- 積極的に現場に入り、現場の状況を自分の目で見て、自分の肌で感じとること
- 積極的に、現場とのよい双方向コミュニケーションの働きかけをすること
- 困っていること、変えたいことを明確にする

コラム “5ゲン” 主義について

積極的に現場に入り、現物、現実を自分の目で見るといった行動は、“5ゲン”に則して行動することともいえます。“5ゲン”とは、「現地・現物・現実」の3現と、「原理・原則」の2原を合わせたものです。現地に行き、現物を見て、現実を知ること（3現）が、管理上、技術上の問題の発見に寄与し、原理・原則（2原）にかなった対策を取ることが、問題解決の妥当性を高めるといふ知見です。

（参考文献：古畑友三，“5ゲン主義入門”，日科技連出版社，1996.）

(h) 現場、関係者にスペシャリストとしての自覚を持たせていますか？

いくら良い仕組み（プロセス）があったとしても、そこで働く人たちがプロフェッショナルとしてのこだわりと自覚を持っていないと、仕事はうまくいかないものです。

職場が「単に要求通りのものをプロセスに従って作りさえすれば良い」と考えて仕事をする風土になっていませんか？

このような状況下では、いつか大きな問題に巻き込まれるリスクが高いでしょう。プロセス改善推進者は、このことをよく認識して、エンジニアの意識や姿勢面での改善活動を推進していく必要があります。

以下、“完成品メーカーから出てくる要求仕様を、部品メーカーがプロフェッショナルとして把握する姿勢”について完成品メーカーと完成品メーカーに部品を納入する部品メーカーとの関係を例に説明します。

もし、部品メーカーのエンジニアが、「発注者側の仕様に問題（誤りや不十分な点など）があっても、発注者側の仕様通りに作り納入すれば良い」という姿勢で部品（製品）開発をすれば、その部品（製品）が搭載された完成品（最終製品）を使用するユーザにとって不利益が発生するリスクを負います。さらに、それが安全に関わるものなら大きな社会問題に発展することもあり得ます。

ここで、よく考えなければならないことは、市場での問題の発生によりユーザ（社会）が混乱すれば、完成品メーカーだけでなく部品メーカーにも、責任問題やしわ寄せが押し寄せてくるということです。

部品メーカーの基本的な責務は、完成品メーカーの要求に合致する部品を開発し、納入することではあります。しかしながら、部品メー

カは「完成品メーカから、いつも完璧な要求が部品メーカに出てくるという保証はない」と考え、要求に問題があればそれを提案し、変更してもらうための行動を起こすことが大切です。

つまり、「問題のあるものは作らない、出さない。うるさいかもしれないが、言うべきことは言う。それが完成品メーカさんのためにもなる、最終ユーザーのためにもなる、そして自分たちのためにもなる」という、“こだわり”を持って、問題の重要性に応じた適切な提案をおこなうことは、プロフェッショナルとしての使命ではな

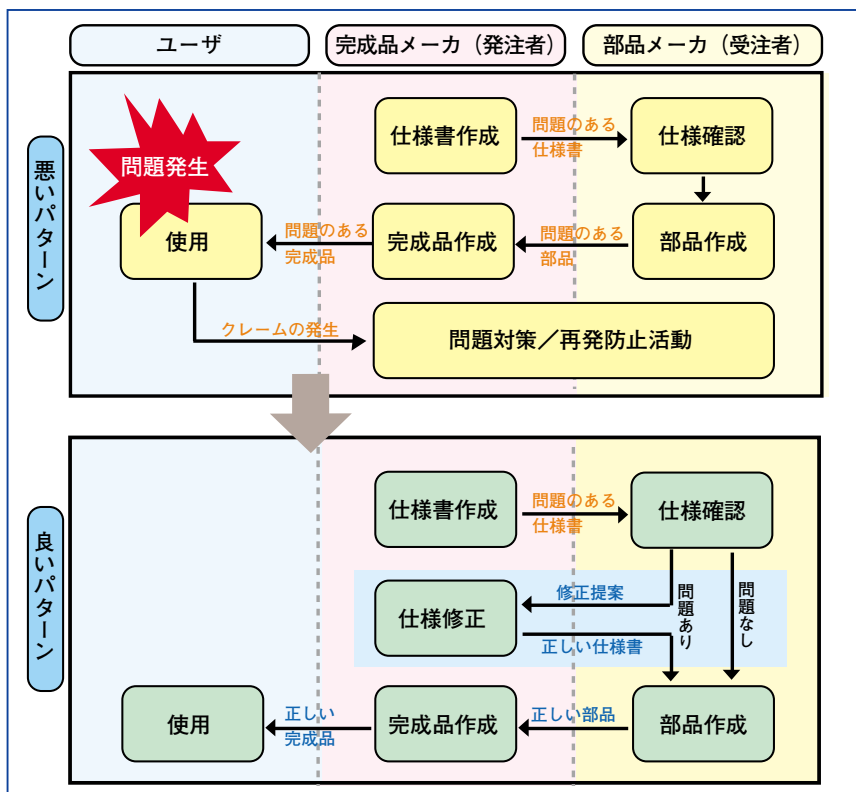


図3-8 現場、関係者がプロフェッショナルな自覚を持つということ

いでしょうか。

また、部品メーカーは、問題を検出し指摘できるだけの技術力を持つことも大切です。技術力がなければ問題点を指摘することすらできません。さらに部品メーカーは、完成品メーカーと共に、最終ユーザーにも視点を置き、一致協力して良い製品を作りあげ、市場に提供していくことが大切です。

3.3 プロセス改善の推進体制

プロセス改善を推進していく体制として、図3-9に図示したようなプロセス改善に関係する各関係者の立場ごとに、どのような役割を持って、どのような点に留意して、プロセス改善活動を推進していくことが大切かについて、以下 (a) ~ (h) の項目をキーワードに説明していきます。

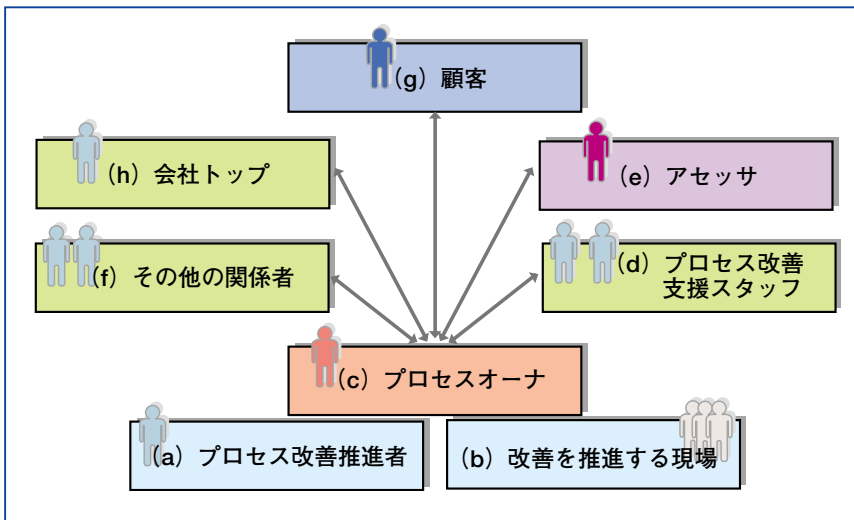


図3-9 改善の推進体制

(a) プロセス改善推進者の役割について

プロセス改善をある現場において推進するためには、中心となって推進していく役割を持った人が必要です。この人をプロセス改善推進者と呼び、現場の中から自発的に買って出た人や、上級管理者によって任命された人がなります。プロセス改善推進者が、プロセスオーナー自身である場合や、プロセス改善支援スタッフの一員である場合も考えられますが、どのポジションであってもその役割は同じです。

また、プロセス改善推進者は、プロセスオーナーの改善に対する理解活動を通じて必要なリソースの獲得や、現場に対して改善活動の理解とモチベーション維持／向上を図ることなど、管理者と現場のメンバの間をつなぐ役目も果たします。

プロセス改善推進者には、現場経験が豊富で、現場の実態を肌で感じてよく知っており、チームを引っ張っていく実行力があり、さらにプロセス改善に対する高い専門性を持っている人がふさわしいと考えます。

また、プロセス改善推進者は、いろいろなシンポジウムなどに参加し、情報を集め、見識、人脈を広げておくことも大切です。良いアセッサ（プロセスを診断する人）や外部コンサルタントに出会えたりするなど、プロセス改善活動の推進に多いに役立つでしょう。

(b) 現場の役割について

現場は、限られた工数、納期の中で、高い生産性を維持し、高品質な製品を作り続けることが要求されます。現場のことを一番よく知っている現場の人たちが、現状の問題や意見を関係者に提起し、そして、自分たちの手で自分たちの現場を改善していくことが現場

の人の役割であると考えます。良い環境は待っているだけで与えられるものではありません。自分たちの手で勝ち取るものです。

たとえば、プロセスオーナーがプロセス改善に理解がない場合、現場からプロセス改善推進者を出し、プロセスオーナーへの理解活動を展開し、プロセス改善支援スタッフによる支援が受けられるようにするなど、プロセス改善のためのリソース（資源）を確保する行動を現場自らが起こすことが大切です。

また、現場の人は「現場のことを一番知っているのは自分たちだ！」とあって現場に閉じこもりがちです。このような状態が続けば、日常化（慣れ、マンネリ化）しすぎてしまい、客観的に問題構造を発見できなくなる可能性があります。こういったことを回避するためには、次の2つの行動を起こすと良いでしょう。

- 社外のコミュニティ（研究会やシンポジウムなど）に積極的に参加し、多くの人と意見交換をするなど、広く外に出て世の中の動向などを把握する
- 第三者による客観的なチェックやアセスメントを受けてみる

(c) プロセスオーナーの役割

プロセスオーナーは、「組織のプロセスを決めて、そのプロセスの実施に責任を持つ人」で、担当しているプロセスの目標を達成するために、環境などのプロセス改善に必要なリソースを確保し、割り当てる役割を担っています。プロセス改善に当たっては、利害関係者として改善活動に関与することが必要です。

複数の組織がプロセスを分担している場合には、複数のプロセスオーナーがいることになります。

たとえば、ソフトウェアの開発プロセスを実施する組織のマネージャや、ソフトウェアの品質保証プロセスを実施する（品質保証活動を担う）組織のマネージャがこれに当たります。

プロセス改善は、関係するプロセスオーナーが協力し、同期してプロセス改善を推進していくことが望ましいでしょう。

また、プロセスオーナーには、積極的に、適切な業務の維持、人材育成、コミュニケーションの円滑化の組織風土を醸成する活動を、部下に丸投げするのではなく、自らが陣頭指揮して推進していける人がふさわしいでしょう（「3.2 プロセス改善推進の知恵（g）項」（85ページ））。

（d）プロセス改善支援スタッフの役割

プロセス改善支援スタッフの役割は、プロセス改善推進者の要請により、プロセスオーナーや担当者の意見を聞き、現場のプロセス改善活動を支援することです。

したがって、プロセス改善支援スタッフには、現場経験が豊富で、現場の実態を肌で感じてよく知っており、さらにプロセス改善に対する高い専門性を持っている人が好ましいですが、現場経験がなくても、現場のことをよく聞き、現場のわがままに流されずに、現場に対して適切な支援や相談に応ずることのできる器量がある人が適任です。仕事ができない人や余った人を集めて推進しようとはしないことです。

一般的に、ソフトウェア開発企業では（有効な）「スタッフ部門」が全く存在しないことも多く、このような場合は、必要に応じて現場や関係者が、兼任でプロセス改善スタッフとして活動をすることもあります。

プロセス改善支援スタッフがプロセス改善を支援するに当たり、よく認識しておくべきことが2つあります。

1つ目は、プロセス改善支援スタッフにより実施される支援で現場にとって一番うれしくないのは、プロセス改善支援スタッフによって、現場のためにならない活動が押しつけられることです。これは、スタッフに現場経験がない人が選任されるとか、スタッフの組織目標が現場を離れて一人歩きしたとき（スタッフの成果を出すためだけの活動になったときなど）によく発生します。

2つ目は、プロセス改善支援スタッフが、現場の内部アセスメントを担当する際に、「監査」と「アセスメント」は違う」ということをよく認識しておくことです。監査とアセスメントをよく混同されている方がいますが、監査は主に「ルールや規則に対する違反事項の摘出（決められたことが確実に実施されているか）」をするのに対して、アセスメントは「自分たちの仕事のやり方がどの程度の能力があるかを診断し、仕事のやり方の強みや弱みを把握する」ものです。現場に対して「プロセス監査だ！」といったような姿勢で接すると、現場のプロセス改善に水を差してしまうことがあります。監査とアセスメントの関係は、ちょうどイソップ物語の北風と太陽の関係とよく似ています。プロセス改善支援スタッフの役割はアセスメントをうまく活用し、現場が自立した改善活動ができるように、現場のモチベーションの維持・向上に気を配った活動による支援をすることが大切です。

(e) アセッサの役割

プロセスを診断するアセッサは、アセスメントを実施し、アセスメント対象プロセスをアセスメントモデルというアセスメント指標

を用いて客観的に評定し、事業目標達成に向けたプロセスの弱み、強み、およびリスクを抽出することにより、プロセスをより良いものに改善するための支援をする役割を担います。

したがって、アセッサには、アセスメント対象の組織の現場をよく理解し、客観的にものごとをとらえて診断する能力とコミュニケーション能力が必要となります。

コラム 良いアセッサについて

プロセスを診断される立場であるアセッシにとっては、単にアセスメントモデルに従ってアセスメントをおこない、単に評定と強み／弱みを出すだけでなく、現場のことをよく理解し、弱みを強みにしていくため、また強みをさらに伸ばすために、熱心に、親身にアドバイスや指導ができる能力と適性を持ったアセッサが良いアセッサと思えるでしょう。

しかしながら、プロセス改善の実施の主体はアセッシ側であることを忘れてはいけません。アセッシ自身がアセスメント結果を踏まえて、自分たちの手で改善を推進していくのであって、アセッサが改善を推進していくのではないということをよく認識する必要があります。アセスメントの中ではプロセス改善の推進役として、また、コンサルタントとしての改善支援行為（作業）まで、アセッサに期待してはいけません。

アセッサの役割はあくまでアセスメント対象の組織の現場をよく理解し、客観的にものごとをとらえて診断することです。

(f) その他の関係者の役割

関係者は、プロセス改善活動を実施する現場と連携して一連の業務を担当している、システム担当部署、ハード設計部署、品質保証部門、サービス部門、営業、製造部署などの組織およびそのメンバーのことをいいます。このように担当製品の設計・製造部署などの、製品開発に多くの部署が関わっている場合、プロセス改善推進者は、対象とする現場のみで改善活動をするだけでは不十分で、関係者との役割の明確化や関係者との情報インタフェースを明確化する必要があります（特に、発注・運用のプロセスの改善が伴う）。したがって、関係者を巻き込んだでの共同の改善活動をすることが望ましいでしょう。

(g) 顧客の役割

供給者にとって、発注者である顧客は最大の関係者であるといえます。したがって、顧客とのコミュニケーションを円滑にし、良好な関係を築いておくことが大切になります。この顧客との良好な関係は、顧客と供給者間の要求の受け渡しにおいてたいへん役に立つものです。

あえていうまでもないことですが、発注者である顧客の役割は、供給者に対して明確な要求をできるだけ早い段階で提示することです。しかしながら、実際はそうはいかない場合が多く発生しています。

また、供給者の役割は、顧客の要求に合った製品を開発し納入することです。開発をできる限り順調に進め、この役割を果たすためには、早期に顧客の要求仕様をまとめ、確定させる必要があります。しかし、顧客の要求を確実に把握することはそう簡単ではありません

ん。

そこで、この要求管理や変更管理といったプロセスを少しでも良い方向に改善していくためには、供給者側だけで解決できない課題が多く存在することから、自社の改善活動だけでなく、顧客の要求管理や発注のプロセスを含めた改善活動を、顧客の理解を得て、顧客と一緒に連携して実施することが大切です。

(h) 会社のトップの役割

トップは、組織が競争力を持ち事業目標を達成するためには、プロセス改善が必須であると認識し、プロセス改善に対して経営資源の配分を適切に実施することが役割です。

しかしながら、関係する現場（関係者）との情報インタフェースや、合同活動を改善しようとしても、プロセス改善についての理解のない部署であれば、改善活動は思うように進められません。したがって、このような状況において、会社のトップは全社レベルでの改善意識の醸成（雰囲気作り）をするなどして、プロセス改善推進者を支援することが大切です。

また、プロセス改善推進者は、このことにトップが気付くのを待っているだけでなく、トップの支援を得るべく、トップに対する理解活動を積極的にしていくことが望ましいことです。

3.4 プロセス改善事例の公開

プロセス改善の結果、ノウハウ、経験は、プロセス改善活動を実施した個々の現場内に埋もらせることなく、関係者間で共有する仕組み（しかけ）を構築し、お互いのプロセス改善活動に積極的に役立て、プロセス改善の効果を最大限に生かすことを考えましょう。

ここで、プロセス改善の結果、ノウハウ、経験を関係者間で共有する仕組み（しかけ）が作れると、データベース（DB）に事例を公開する側および事例を参照する側に、次のようなメリットが得られるでしょう。

事例を参照する側のメリットとして、

- DBに公開された事例から、同じ問題を抱えている人たちとそのプロセス改善事例を知ることができます。
- DBに公開された事例から、プロセス改善のヒントが得られます。

他のプロセス改善活動の考え方、実施方法からヒントを得て、自分たちに応用することができます。

- DBに公開された事例を見て、自分たちのプロセス改善レベルを知ることができます。

自分たちのプロセス改善活動が他に対して、遅れているのか進んでいるのかベンチマーキングすることができます。「もっと良い活動をするぞ!」といった刺激になることもあるでしょう。

事例を公開する側のメリットとして、

- DBの参照者からの問い合わせにより、公開した事例に対しての有益な意見、コメントを期待できます。そして、これをもとにさらなるプロセス改善につなげていくことが可能となります。
- DBの参照者からの問い合わせにより、同様なプロセス改善を実施している部署との意見交換活動や連携活動の可能性が期待できます（仲間の輪を広げられる）。

プロセス改善推進者は、このノウハウや経験の共有の仕組み構築に際して、次の (a) ~ (c) をしっかり認識しておくといでしょう。

(a) ノウハウや経験の共有の仕組みを、単なる改善事例を収集し公表するだけの場で終わらせないようにしよう。

プロセス改善のノウハウや経験の公開の仕組みを作ったものの、なかなか事例が集まらない。公開してもうまく活用されていないということがあります。うまくいかない理由には、検索性が悪いなどのユーザビリティ上の問題など、さまざまな要因が考えられますが、一番の要因は、改善のノウハウや経験を公開する目的が、「ノウハウや経験を集めて公開し、広く活用してもらおう」というだけで、単に事例を集め、DBで公開するだけで終わっていることにあると考えます。

このように、一方通行的な公開になると、事例をDBに公開した側にとって、先に述べた“事例参照側のメリット”を得ることが難しくなります。公開しても享受できるメリットが参照した側と変わらないのなら、あえて事例を公開しようとする気にはなりにくいのではないのでしょうか。これでは、“なかなか事例が集まらない”ということになってもおかしくありません。

また、DB公開事例を参照する側にとって、活用したい事例があっても、DBに公開されている内容だけでなく、実際に事例を公開した部署からもさらに詳しい情報を要求されるのではないのでしょうか。DBまたはこのDBを運用している人がこれに対しての十分なサポートをしていなければ、“公開してもうまく活用されない”ということになってもおかしくありません。

したがって、プロセス改善推進者はノウハウや経験の共有の仕組みを、ノウハウや経験を集め公開するだけでなく、上記のメリットを公開した側も、参照する側も享受できるような工夫（仕掛け）を作っておく必要があります。

そのためには、公開された事例情報だけの情報共有に終わらせないこと。つまり、事例を公開した側とその事例を参照した側との双方向コミュニケーションができる仕組みや、事例の公開を加速させる仕組みが必要であると考えます。たとえば、以下のような仕組みが有効であると考えます。

(1) 事例を公開した側とその事例を参照した側との“お見合い機能”の充実

- 事例を参照した側が参照した事例に対しての意見を述べ、事例を公開した側やほかの人たちが、DB上で参照できる機能を付ける。
- 事例を参照した側が、公開した側に連絡が取れる機能を付ける。

たとえば、公開した側の問い合わせ先をDBの事例に明記しておく。

(2) プロセス改善事例DBをプロセス改善活動結果のアピールと相互研鑽の場とすること。

- プロセス改善の進捗とプロセス改善で利用した技術の宣伝の場とする。

「自分たちの活動結果を積極的に宣伝し、意見を広く求め、次のプロセス改善活動に反映させる」といった意識の醸成活動を展開しましょう。DBにも掲示して主張しま

しょう。

- 広く意見を求める意見交換の場とする。

たとえば、プロセス改善事例の報告会や検討会を開催し、プロセス改善事例の議論の機会を多く作ること（きっかけ作り）がよいでしょう。

- (3) 改善事例DB自体が、一方的な支援の弊害になってしまわないようにすること（「3.2 プロセス改善推進の知恵（a）項」（75ページ））。

- ユーザへの説明も了解もなしに、スタッフであるプロセス改善推進者がよかれと思ってDBを作っても、それがユーザにとってうれしいものであるとは限りません。もし、実際うれしいものであったとしても、ユーザがそれを知らなければその効果は発揮されません。

- プロセス改善事例DBの企画、開発段階から、利用者（関係者）の全員の参画をしてもらい、自分たちの改善事例DBであると認識してもらうことが大切です。

- (4) 事例を公開した人や部署にインセンティブを与える

- 事例を公開した人や部署に報償を与え、事例を積極的に公開する意識を醸成する。

- 公開された事例を活用した人や部署に報償を与え、事例を積極的に活用する意識を醸成する（有効に活用された事例を多く公開した部署に対しても報償を与えることもよい）。

- 成果のみで仕事の評価査定を実施している組織では、ノウハウが個人所有となって表に出てこない、また失敗が隠されるといった風土になりやすいと考えられます。こ

のような風土を持っている組織では、仕事の成果だけでなく、仕事をどのように考えて実施したのか、その過程も評価するといったことにより、行き過ぎた成果主義査定を是正することも必要となります。

(b) 「ほかの良いノウハウや経験を、そのまま自分のプロセスに持ってきて、必ずしもうまくいくとは限らない」ということをよく認識しよう。

ほかの人に効いた薬が必ずしも自分にも効くという保証はありません。人によって体質が違えば、用法・用量も変えなくてはなりません。プロセスも同様に、プロセスごとにプロセスの目的、取り巻く環境や組織風土などのプロセス特性が違っていれば、ほかのプロセスでうまくいったノウハウや経験を、そのまま自分たちのプロジェクトに実装しただけでは、必ずしもうまくいくとは限らないのです。

つまり、ほかの良いノウハウや経験を自分たちのプロセスに適用する場合は、自他両者の事業環境、組織文化、業務プロセス（仕事の手順・方法）などの違いをよく識別し、自分たちのプロセスにうまくフィットするように加工（工夫）する必要も出てくることをよく認識しておくことが大切です。

たとえば、公開されているプロセス改善事例を額面通りに理解するだけでなく、その部署でプロセス改善に実際に携わってきた人への聞き込みをしたり、実際の活動を見させてもらったりするなどして、現地に行き、現物を見て、現実を把握するといった行動により、プロセス改善事例に掲載されていることだけではわからないノウハウや経験を獲得することはたいへん有効です。

(c) 改善ノウハウの公開範囲の拡大への期待と機密管理との
折り合いをつけておくことが必要

プロセス改善ノウハウや経験などのプロセス改善情報の共有の仕組みが、関連する組織間から、会社内の全部署へ広がり、さらに日本国内の企業間で実現されることにより、さらに多くのプロセス改善情報が共有され、有効に活用されることが期待されます(図3-10)。

このようにプロセス改善情報は、企業内/企業間でも共有すべきですが、企業によって機密管理のスタンスがかなり違うため、情報をオープンにし、共有することに対して、機密管理との折り合いをつけておく必要があります。

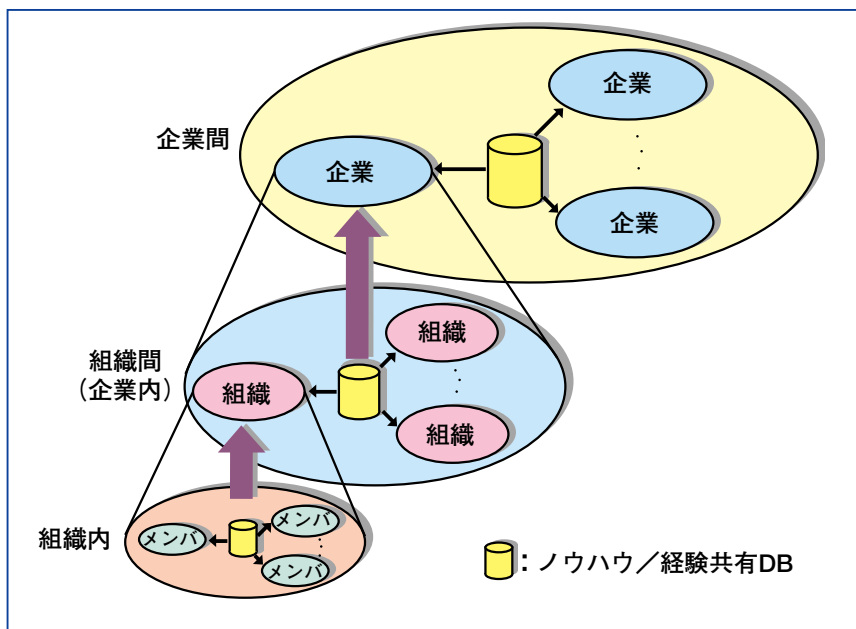


図3-10 ノウハウ・経験の共有データベースの共有範囲

3.5 プロセス改善プログラム

プロセス改善は、プロセス改善プログラム（またはプロセス改善プロジェクト）として、目標を立て、その実現計画をスケジュールを含めて立案し、実施責任者（改善依頼者）や役割分担を明確にし、実施のための資源を割り当てるなどにより、組織的に取り組むと有効であることが多くあります。

そこで、改善プログラムが有効な成果を出すための望ましい要件を記述します。

(a) 組織が抱える課題と改善の目標および対象範囲の明確化をする

改善の対象範囲を決め、その現状とあるべき姿とのギャップを識別し、具体的な活動につなげていくことが大切です。

(b) 経営者・責任者がコミットメントを出す

経営者・責任者が改善プログラムの実施をコミットメントし、組織的活動であることを明確に示すことは、改善活動を実施する関係者全員の意識づけのために必要です。

(c) 関係者の幅広い合意形成をする

関係者の幅広い合意形成を取ることにより、関係者全員の総合力を発揮させることが必要です（「3.4 プロセス改善事例の公開 (b) 項」(101ページ))。

(d) 具体的な活動・役割・手順と資源を用意する

活動を確実に推進するためには、誰が、いつまでに、何をするの

かといった具体的な活動計画が必要です。

**(e) アプローチに関する学習とプログラム進捗の
情報交換の仕組みを構築する**

関連する情報を関係者全員が共有することにより、連携活動が可能となります。また、改善活動において実施したアプローチの評価を実施し、その特質（長所、短所）をよく把握しておくことは、新たな改善活動に対して有効活用することが可能となります。

**(f) 必要な場合、先行的に実施するパイロットプロジェクトの
実施計画を立てる**

プロセス改善では、現在実施中のプロジェクト（事業）に中断などの影響を与えてはいけません。したがって、必要に応じて、パイロットプロジェクトを決め、改善プログラムを先行的に実施し、改善プログラムに無理がないか、導入プロジェクトに支障が出ていないかなどを確認したうえで、全面的に導入するといったアプローチを試みることも有効です。

**(g) プロセス改善の組織文化を育成するための諸施策を
実施する**

プロジェクト情報、業務関連情報のメンバ全員の共有、改善活動計画立案への関係者全員の参画、プロセス改善の必要性や品質意識の醸成教育や改善手法の勉強会の実施、マネージャ・リーダ層のコミュニケーション能力／コーチングスキル／モチベーション管理能力の向上教育の実施など、自分たちに役立つ施策を自分たちで考え企画していくことが大切です。

コラム プロセス改善プログラム（process improvement program）について

- ・国際規格15504 第1部にて、プロセス改善プログラムは、用語として次のように定義されています。

「特定の改善ゴールの達成に関する、全ての戦略、指針、ゴール、責任、及び活動」

- ・国際規格15504 第4部にて、プロセス改善プログラムは、計画を作成し、進捗管理をすることが望ましく、計画には、次のことが含まれていることが望ましいと記述されています。
 - －組織的なプロセス改善活動の背景、歴史、及び現在の状況
 - －組織の事業目標から得た改善目標
 - －組織の範囲 改善プログラムのための組織範囲
 - －プロセスの範囲 改善対象プロセス
 - －プロセス改善のライフサイクル
 - －役割と責任
 - －資源
 - －適切なマイルストーン、レビューポイント、報告の仕組み
 - －プログラムに関係するリスク、選択したリスク管理プロセス
 - －プロセス改善プログラムによって影響される全ての関係者に、進捗を常に知らせる活動

おわりに

プロセス改善は、現場の実態とはかけ離れた理論や知識で、高度なことを議論し、導入すればうまくいくというものではありません。プロセス改善は、現実の開発・運用の実態・現実をいかに変えていくかという、自分たちの周りに転がっている技術的な課題を、日々の活動の積み重ねによって地道に解決し、良い成果を得られるようにしていくものです。このことをよく理解してプロセス改善にチャレンジしていきましょう。

付録 用語集

■ 注意すべき用語

assessment/appraisal

共にプロセスを診断し、プロセス能力あるいは組織のプロセス成熟度を評定する行為を意味する。CMMIカルチャではアプレイザル (appraisal) という用語を、また国際規格ではアセスメント (assessment) という用語を用いる。

assessor/appraiser

共にプロセスのアセスメントをおこなう専門家を意味するが、CMMIカルチャではアプレイザ (appraiser) という用語を、また国際規格ではアセッサ (assessor) という用語を用いる。

Continuous/Staged

歴史的に考案されたいろいろなアセスメントモデルのタイプを分類するときの用語で、連続モデル、段階モデルという分類が使われる。また、CMMIモデルの「2つの表現形式」を示す用語でもある。簡単な説明が別冊『プロセス診断活用編』にある。

process/process area (PA)

本書では、プロセス (process) とは主としてソフトウェアプロセスのことを指す。場合によりシステムエンジニアリングプロセスを指すこともある。プロセスとは何かについては、本書の本文を参

照のこと。なお、CMMIカルチャでは、同様の意味でプロセスエリア (process area) という用語を用いることもある。

■ 略語

15504	……………	ISO/IEC 15504 Information Technology-Process Assessment
CAPDo	……………	Check-Act-Plan-Do
CMM,CMMI	………	Capability Maturity Model (Integration)
GQM	……………	Goal-Question-Metric
IEC	……………	International Electrotechnical Commission
ISO	……………	International Organization for Standard
PDCA	……………	Plan-Do -Check-Act
QCD	……………	Quality-Cost-Delivery
ROI	……………	Return on Investment
SEI	……………	Software Engineering Institute
SLCP	……………	Software Life Cycle Processes
SPA	……………	Software Process Assessment
SPI	……………	Software Process Improvement
SPICE	……………	Software Process Improvement and Capability Determination
SW—CMM	………	Software CMM
SWOR	……………	Strength, Weakness, Opportunity and Risk
TQM	……………	Total Quality Management
TR	……………	Technical Report
XP	……………	Extreme Programming

索引

アルファベット

CAPDo	30
CASEツール	45
CMMI	32
CSCW	49
GQM	59
IDEAL	32
PDCA	30
QCD	1
RUP	59
SCRUM	48
SLCP	15
XP	59

あ行

アセスメント手法	58
アセスメントモデル	22
アセッサ	93
エンピリカル ソフトウェア工学	59

か行

改善サイクル	29
開発ライフサイクルモデル	46
技術要素	64
協働意識	69
記録ツール	35
計画ツール	40
形式的仕様記述言語	47
結果系ゴール	19

構成管理ツール	45
効率	28
国際規格15504	57
国際規格15504 第1部	105
国際規格15504 第4部	34
コミットメント	103
コミュニケーション	72

さ行

サプライチェーン	54
事業目標	51
資源配分計画	42
実施支援ツール	45
人的要素	64
成熟度レベル	26
測定ツール	36
組織目標	76
ソフトウェア ～組織	2
～ライフサイクルプロセス	15
～コミュニティ	6

た行

テスト技法	48
テストツール	46
統計分析ツール	37
トップダウンアプローチ	78

な行

能力アセスメントツール	38
-------------	----

は行

品質特性	12
プロセス	
～オーナー	78,85,91
～改善支援スタッフ	75,85,92
～改善推進者	75,90
～改善の効果	25
～改善施策	22,23
～要素	64
～リポジトリ	50
ベンチマーク（比較）ツール	38
報・連・相	68
ボトムアップアプローチ	78

ま行

見える化	36,73,79
モデルベース	57

や行

要因系ゴール	19
要求管理ツール	45
要求工学	47

ら行

レビュー	48
------	----

執筆者（敬称略）

経済産業省 ソフトウェアプロセス改善研究部WG1

安達 賢二	株式会社HBA
足立 久美	株式会社デンソー
江崎 美保	株式会社日新システムズ
北野 敏明	独立行政法人 情報処理推進機構/新日鉄ソリューションズ 株式会社
近藤 聖久	三菱電機株式会社
砂塚 利彦	独立行政法人 情報処理推進機構/砂塚コンサルティングサ ービス株式会社
伏見 諭	株式会社情報数理研究所
堀田 勝美	株式会社コンピータジャパン

経済産業省 ソフトウェアプロセス改善研究部メンバ

主査 菊島 靖弘	東京海上日動システムズ株式会社/株式会社アイネス
副主査 新谷 勝利	独立行政法人 情報処理推進機構
赤坂 幸彦	株式会社NTTデータ
穴田 直也	株式会社大和コンピューター
岩佐 洋司	住友電工情報システム株式会社
小川 清	名古屋市工業研究所
片平 真史	宇宙航空研究開発機構
北島 義弘	株式会社PM Academy
小泉 浩	マイクロソフト株式会社
河野 文昭	株式会社アドヴィックス
込山 俊博	日本電気株式会社
佐藤 健哉	同志社大学
田中 一夫	株式会社CSKシステムズ
谷川 浩	トヨタ自動車株式会社
徳永 享	富士ゼロックス株式会社
中村 伸裕	住友電気工業株式会社
服部 祐二	ブラザー工業株式会社
松原 友夫	松原コンサルティング
石谷 靖	独立行政法人 情報処理推進機構/株式会社三菱総合研究所
吉田 尚志	独立行政法人 情報処理推進機構/株式会社NTTデータ

編 者 紹 介

独立行政法人 情報処理推進機構 ソフトウェア・エンジニアリング・センター
2004年10月に独立行政法人 情報処理推進機構（IPA）内に設立されたソフトウェア・エンジニアリング・センター（SEC）は、エンタプライズ系ソフトウェアと組み込みソフトウェアの開発力強化に取り組むとともに、その成果を実践・検証するための実践ソフトウェア開発プロジェクトを産学官の枠組みを越えて展開している。


[所在地] 〒113-6591 東京都文京区本駒込2-28-8

文京グリーンコート センターオフィス

電話 03-5978-7543, FAX 03-5978-7517

<http://sec.ipa.go.jp/index.php>

- 本書の内容に関する質問は、オーム社雑誌部「(書名を明記)」係宛、書状またはFAX (03-3293-6889)にてお願いします。お受けできる質問は本書で紹介した内容に限らせていただきます。なお、電話での質問にはお答えできませんので、あらかじめご了承ください。
- 万一、落丁・乱丁の場合は、送料当社負担でお取替えいたします。当社販売管理部宛お送りください。
- 本書の一部の複写複製を希望される場合は、本書扉裏を参照してください。

 <(株)日本著作出版権管理システム委託出版物>

SEC BOOKS

プロセス改善ナビゲーションガイド

～なぜなにに編～

平成 19 年 3 月 30 日 第 1 版第 1 刷発行

編 者 独立行政法人 情報処理推進機構
ソフトウェア・エンジニアリング・センター

発 行 者 佐 藤 政 次

発 行 所 株式会社 オーム社

郵便番号 101-8460

東京都千代田区神田錦町 3-1

電 話 03 (3233) 0641(代表)

URL <http://www.ohmsha.co.jp/>

© 独立行政法人 情報処理推進機構 ソフトウェア・エンジニアリング・センター 2007

印刷・製本 報光社

ISBN 978-4-274-50131-9 Printed in Japan

オーム社/雑誌局

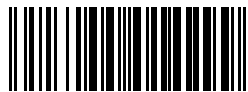
ISBN978-4-274-50131-9

C3055 ¥1429E



9784274501319

定価(本体 1429円【税別】)



1923055014299

IPA[®] 独立行政法人 情報処理推進機構
ソフトウェア・エンジニアリング・センター

SEC-TN06-002



R100

100%リサイクル用紙を使用しています