

# 「なぜ」を考えるプロセス改善

-プロセス改善の根底に横たわる仮説について

*-Über die Hypothesen, welche der  
Prozessverbesserung zu Grunde liegen*

2018/10/12

(株)プロセス分析ラボ

小室 睦

## ● 自己紹介、背景

- もともとは数学をやっていました ⇒H社に就職。
- 就職当時は第2次AIブーム⇒研究部門でAIシステムの研究  
⇒ソフトウェア工学に転向。(2001年PLoP参加)
- 「日本版CMMI」騒ぎの頃から、CMMIに関わる
- H社、F社でEPGのマネージャー(プロセス改善の全社とりまとめ)
  - ◆ H社では2001年頃からCMMIをプロセス改善に活用
  - ◆ 世界でもかなり早い時期、いくつかの「発見」
- F社では主に高成熟度改善活動に従事
- 現職((株)プロセス分析ラボ)では、改善活動のコンサル、アプライザルなどを実施
  - ◆ レベル3までの改善が多い
- 昨年のSPI Japanでのニコンシステム清水さんの発表[1]
- 昔、いろいろ苦労して見出した「発見」を伝えるのも良さそう
- 当時はよくわからずに実施していたことにもいまから見直すと見直しを与えられるのではないかな？

# すべてのモデルは間違っている でも、役立つものもある。

- All models are wrong, but some are useful. (George P.E. Box)[2]
- CMMI: 開発・管理・改善のためのプロセスモデル
- このようなモデルを活用した改善は「モデルベース改善」と呼ばれることがある
  - モデルって何？
    - ◆ 仮説の複合体: 私の見方
      - 初めてCMMIを理解した時、これで研究のネタがいくらでもできると思った
    - ◆ お手本、規範: 一般の見方
- モデルベース改善
  - 一般的には正しい、有用なことでも自分たちの状況に本当にあっているかどうかは自明ではない
  - モデルそのもの、あるいは解釈が間違っているという可能性もある
  - フォース:  
現場ベースのボトムアップ改善 vs. 組織レベルの改善、経験・知見共有
    - ◆ 特に改善の初期には半ば強制的にプロセスを実施するという戦略もありうるが自分たちで考えるという方向にどこかでスイッチすべき

- 定石とは(局所的に)最善とされる打ち方
- 局所的には最善でも、コンテキスト(文脈・意図)を意識しないと悪手になることもある

## ● (類似例) 数学での公式や定理

- 公式を丸暗記して無理に当てはめて使おうとすると、かえって間違える
- 公式を導いたアイデアやテクニック(「こころ」)を理解して、適用状況にあった導き方をすると、問題の状況にあったうまい解になる
  - ◆ フォースを使え、感じるのだ (ヨーダ)
- すすんだ視点を持っていると、定理や公式も簡明に理解・導出することができて、見通し良くきれいに解くことができる場合もある

## ● プロセス改善では？

- CMMIはモデルであってプロセス定義・記述そのものではない
- CMMIをプロセス定義のお手本のように考えるのは、CMMIの作成意図から見ても間違い
- CMMIの目指すのは「常に自律的に改善し続ける組織」(成熟度レベル5)であり、プロセスは進化し続けるもの。[[常に改善・進化するプロセス]] (別名 変化を受け入れよ)  
何か理想的なプロセスがあり、それを実施すればよいというものではない。
- 自分たちのプロセスを大切に、プロセスの(共)進化

- プロセス改善は科学的にすすめるべき
- 科学的とは？
  - 事実・データに基づいて実証的にすすめる [[事実とデータの重視]]
  - 仮説をたて、それを立証する
  - まだ、確かめていないものは「仮説」の段階であり、結論をそのまま受け入れてはいけない。
  - 確立されたと思われている「理論」でも、状況が違えば確かめられていないかもしれない
  - 既存の理論を絶対視する「科学的な見方」は実は科学的ではない
- 自分たちのプロセスを大事にして、それを改善していくことが重要
  - 標準プロセスで用意されている資源がうまく使えるなら使った方がよい
  - 標準の方が間違っている(改善できる)という可能性もある
  - モデルや標準の想定と異なっている場合、原因があるはず。意図・意味・目的を考えてみるのが重要。別の言い方をすれば、「そのプロセスの価値は何か？」

- 「意味」の取り扱い
  - どんどん深く考えていくと哲学的な問題になってしまう
  - ここでは、一段だけ展開して「原因」「要因」と思われるものをあげることにとどめておく
  - 原因、要因と思われるものに、キャッチフレーズ的なタイトルをつけ  
‘[’ と ‘]’で囲んで示す
  - 例： [[早期発見、早期対策]]
- この発表で示すのは、私個人の理解、解釈の現状に過ぎない。
  - いまは最善と想着いても、しばらくすると「あれは間違っていた」というかもしれない。(あくまで仮説なので)
  - 今後の発展には、他の方の協力が必要

- ピアレビュー(開発のためのCMMI V1.3より)
  - ピアレビューを実施する目的の一つは、早い段階で欠陥を発見し除去することである。ピアレビューは、作業成果物が開発されるにしたがって漸進的に実施される。これらのレビューは構造化されており、管理層レビューではない。
- 意外だったこと
  - レビューといえば「エライ人」に見てもらいたいものだと思っていた(公式レビュー)
  - キビシクチェックすることで品質向上させるものだと思っていた
- 意図・目的
  - [[欠陥(問題)憎んで人を憎まず]]
    - ◆ 欠陥に関する責任追及は厳禁、フランクに話せない雰囲気もNO([[動機付け]])
    - ◆ 作者は最も重要なレビューアー、レビューされるのは作業成果物、作者ではない
  - [[早期発見、早期対策]] (別名 フェイルファースト)
    - ◆ 欠陥をできるだけ早く取り除くことで、品質向上、工数削減する
    - ◆ 工数削減にもなる理由は 欠陥1件にかかる品質コストが工程とともに急上昇するから ([[Quality is Free]])

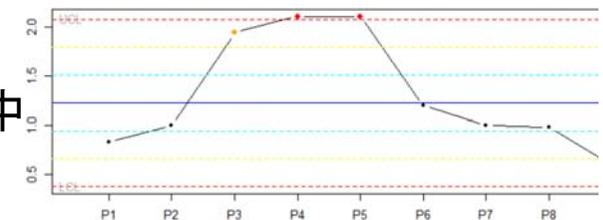


# 改善例(ピアレビュー)

- 誤解されがちなプラクティス:「ピアレビューの準備、実施、および結果についてデータを分析する」
- 意図・目的: [[良い結果は良いプロセスから]]
  - ピアレビュー手法によりパフォーマンスが大きく異なってくる
  - 自分たちの実情にあった手法を工夫(もしくは採用)することが大事



- (事例)レビュー実施データからどんなレビュー(方法)に効果があったかを分析し、改善 [3][4]
  - ある程度の時間をさいてじっくり目を通す(急ぎ過ぎない)
    - ← 一回のレビュー規模を制限する、もしくはレビュー速度を測定して監視制御する
  - 少人数レビュー(4~5人まで)
  - レビュー準備を実施し、観点を明確化、欠陥摘出に集中
    - 原因分析・修正作業はレビュー実施後
  - エキスパートがレビューアとして参加
  - × 悪い例: 工程最後に数日間かけて、システム全体をまとめてレビュー



- [[早期発見、早期対策]]

レビュー前倒し率

= レビューで抽出した欠陥数 / 工程全体にわたる総欠陥数

- プロジェクトおよび組織の品質状況を示す有用な指標

- [[Quality is free]]

- 欠陥除去費用: 欠陥1件を発見・除去するのに  
かかった費用
- 通常、下流工程に行くに従い急激に増加する

- [[良い結果は良いプロセスから]]

効果の高いレビューの仕方を探す

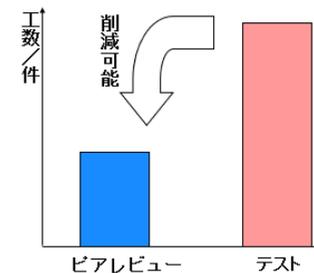
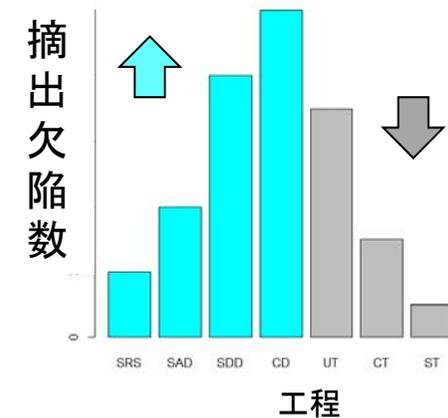
レビュー効率 =

レビューでの指摘数 / レビュー工数

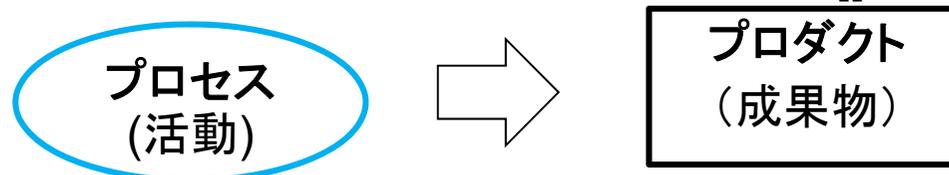
- レビュー実施状況の監視

レビュー速度 = レビュー規模 / レビュー時間

欠陥密度 = 欠陥抽出数 / レビュー規模



- アプレイザルの指摘: プロセスに対するQAの実施(以下、プロセスQA)
  - (プロセス改善活動の初期の事例)H社は品質保証に自信を持っていたのでPPQAで改善機会(弱み)を指摘され非常に驚いた
  - 一方、QA作業の負荷が大きすぎるので何とかならないかという声が品質保証の上級管理者からあがっていた。
  - 改善前アプローチ: プロダクトの徹底的なチェックで品質保証する
- 意味・意図: [[良い結果は良いプロセスから]]
  - プロダクトは「結果」だが、プロセスは「原因」により近く、チェックの効率が高い
  - プロダクトチェックのみで品質を確保しようとする、網羅的なチェックが必要で大変
  - 早いタイミングでチェックでき、手戻りを削減／防止可能[[早期発見、早期対策]]



- 効果の実証データ
  - プロセスQAで品質向上(統計的有意な欠陥密度低下)したという報告[5]
  - プロセスQAによりリスクが軽減されているという報告[6]
  - ただし、プロセスQA以外の改善活動も同時に実施されていたため、これらが本当にプロセスQAのみによる効果かどうかは、はっきりしない

## 仮説1: より効果的なプロセスQAの仕方

- 的を絞ったプロセスチェック: 四六時中張り付く必要はない、かえって非効率
- 「プロセスを見ないとわからないこと、見れば容易にわかること」をチェックする
  - 実施タイミングのチェック: やるべき時にやるべきことをやっているか？  
重要な依存関係(リソース、スケジュール)が適切に監視されているか。
  - 記録にあらわれないプロジェクトの状態: モチベーションが下がっていないか？ チームメンバーの本音は？ 作業負荷のアンバランスはないか？ (但し、「プロセス警察」にならぬよう注意)
  - その他、リスクやプロジェクト状態の客観的チェック: データからの定量的なチェックも有効
  - チェックする活動の例(ピアレビュー)
    - ◆ 有識者が参加しているか？参加者は必要な準備をしてきたか、司会はそれを確かめたか？
    - ◆ 責任追及に走っていないか、発言しない参加者はいないか、特定の人に議論が集中していないか
    - ◆ (叩き出し方式のレビューの場合)欠陥の原因究明などの他の議論に走っていないか
    - ◆ レビュー速度、欠陥密度の定量分析[4]

## 仮説2(提案): 価値向上に結び付くプロセスチェックが有効

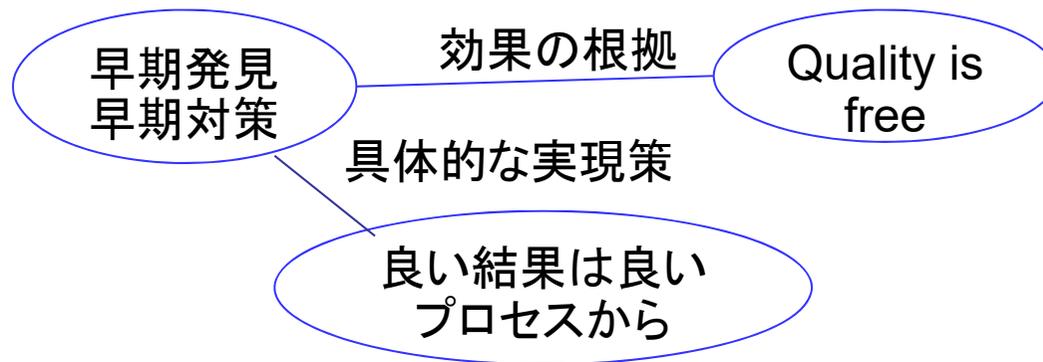
- CMMIでのプロセスQAは定義されたプロセス(標準プロセス)をチェックするものであるが、プロセス自体も進化していくものであり、まだ不完全な点があるかもしれない。
- 不完全なプロセスをもとに遵守状況だけをチェックすると「プロセス警察」に近づく危険がある。
- そのプロセスを実施する意図・目的は何であり、どんな「価値」があるのかを考えて、価値が向上するように、QAがサポートしていくというのがよりよい姿 [[常に改善・進化するプロセス]]  
(別名 変化を受け入れよ)

- GP(共通プラクティス)は軽視されがちだが、プロセスが永続的に実施されるためには非常に重要
- 改善初期によく見られる問題点: プロセス領域の数だけ組織方針がある(GP2.1)
  - プロセス領域はプロセス評価の分類であり、プロセスの実施の分類ではない。  
(例、PP, PMC, IPM, QPM)
- 方針の意図・目的: [[ベクトルを合わせる]] (別名 ビジョンの共有)
  - よい仕事をするには、組織、プロジェクト、チームがビジョン(方針、目的)を共有して活動する必要がある: 組織方針、プロジェクト方針、システム化方針、設計方針、品質方針、プロジェクト管理方針, etc. (注)プロジェクト方針、チームのビジョンはGP2.2, IPM SP1.6のカバー範囲
  - 例えば、システム化方針、設計方針が確立され共有されていないと良いシステムは構築できない
- 問題の多くはGPを通して現れる
  - SPは幹・枝・葉(時には花や実)だが、GPは根にあたるもの。
  - 意図・目的: [[自律的なプロセス実施環境を提供する]]
    - ◆ プロセスをうまく実施するためには、誰がいつどんな形で、誰と協力して実施するのか、必要な環境を提供しなければならない。
    - ◆ 役割を決めて、責任だけ押し付けてもうまくいかない。環境を整えて[[動機づけ]]する必要がある。



# プロセス改善の意図・目的に関する パターン言語

- CMMIのプラクティスはプロセスではない: プロセスのあるべき特徴を述べたもの
  - ソフトウェア開発でいうと「外部仕様」のようなもの
- 本発表ではプラクティスの「意図・目的」を考察することで、一段、展開した特徴を与えた
- 事例としてあげたピアレビューとプロセスQAでは、展開すると[[良い結果は良いプロセスから]]、[[早期発見、早期対策]]といった項目が同じように現れた。つまり、同じ「パターン」になっている
- それぞれの特徴をパターンとして記述して、パターン言語化する試みを実施中



- 協力していただける方を募集中！興味のある方は小室までご連絡ください。

- 本発表でのプロセス改善に対する考え方はアジャイル開発と共通点がある: 仮説・検証ベースで試しながら良くする
- パターン中にアジャイル開発でも用いられる表現が含まれている
  - フェイルファースト、変化を受け入れよ
  - 欠陥(問題)憎んで人を憎まず ⇔ みんなで問題に向き合う、一方向的な報告を求めたり、責任追及をしたりはしない
- アジャイル品質保証の達人である永田さんの実践・提唱している品質保証のやり方[7]は本発表でのプロセスQAに近い
  - QAがプロジェクトに入り込んで、見守り、価値あるサポートを提供する。
  - メトリクスを活用した見える化
- フォースと共にあらんことを！
  - 本発表のように意味・意図に(要素)分解してしまうと、ウォーターフォールでもアジャイルでも違いはなくなる。
  - 違いはフォースのバランスのとり方  
現場ベースのボトムアップ改善 vs. 組織レベルの改善、  
経験・知見共有



- プロセス改善において、意図・目的を考えることの重要性について述べた
- 代表的なプロセス改善施策を分析すると、意図・目的の現れ方にはパターンがありそうなのがわかった
- 意図・目的の観点で見ると、伝統的な開発プロセス（ウォーターフォール）でもアジャイル開発でも大きな違いは見られない
  - 違いはフォース（top down vs. bottom up）のバランスのとり方
- 今後の発展に向け、一緒に考えてくれる人、募集中
  - パターン言語の整備
  - アジャイルの場合の意図・目的の分析

- [1] 清水 崇司、「SEPG活動の立ち上げと、組織定着への取り組み 10年に亘り、「作業者にとって優しいか」を理念に活動した報告」, SPI Japan 2017, [http://www.jaspic.org/events/sj/spi\\_japan\\_2017/](http://www.jaspic.org/events/sj/spi_japan_2017/) (2017).
- [2] Box, G. E. P. "Science and Statistics", Journal of the American Statistical Association, 71, pp.791–799 (1976).
- [3] 小室睦、男澤康、木村好秀、「開発現場の実態に基づいたピアレビュー手法 改善と改善効果の定量的分析」, SEC Journal, Vol.1, No.4, pp.6-15 (2005).
- [4] 野中誠, 小池利和, 小室睦, 「データ指向のソフトウェア品質マネジメント」日科技連出版 (2012).
- [5] 小室睦、高橋一郎、角田文広、「プロセス改善活動とその効果の定量的評価」 PM学会春季研究会, pp.133-138(2004).
- [6] 鈴木新吾、小室睦、「プロセスQAのマネジメント支援への効果」、PM学会春季研究会, pp.152-154(2005).
- [7] 永田敦、「OODAと品質保証、アジャイル開発における品質管理のアプローチ」 第2回enPiT-Proスマートエスイーセミナー資料 (2018).