

# ソフトウェアプロセス改善を組織的，実証的にすすめるためのデータ分析パターン言語(progress report)

小室 睦<sup>1</sup>, (株)プロセス分析ラボ  
鷺崎弘宣<sup>2</sup>, 早稲田大学

---

ソフトウェア開発プロセス改善におけるデータ分析活動をパターン言語として記述する試みについて紹介する。これはソフトウェア開発企業内でプロセス改善活動を実施する状況を念頭に，改善活動をより効果的にすすめるためのデータ分析活動をパターン言語としてまとめたものである。これは現在作成中のパターン言語であり予備的な記述となっている部分もある。このパターン言語ではパターンを (1)データ分析のための環境・インフラの整備や準備に関わるパターン(2) データ分析そのものに関するパターン，(3) データを活用した改善活動に関するパターンの3つに分類して記述している。それぞれの分類について概要を述べたのち，パターン記述の例を3つ紹介する。

---

## 1. はじめに

プロセス改善の歴史は古く，日本での源流は第2次大戦後の製造業の品質管理運動[1]に求めることができる。しかし，ソフトウェア開発にプロセス改善を適用しようとするとき，人的要因や環境要因の影響が大きいなど，特有の問題がある[2][3]。

品質管理運動では統計的手法を用いた工程管理が重要視されているが，ソフトウェア開発とそのプロセス改善を効果的にすすめるためにもデータ分析は有効である[4][5]。データを通じてプロセスの実施状況を客観的に把握し，改善事項の発見，改善効果の確認・評価，品質の可視化など実施状況の定量的監視・制御を実現でき[6]，さらに進んだ段階では実施結果（品質・工数・納期等）の定量的予測も可能になる[7]。また，最近一般的になりつつあるアジャイル開発においても継続的な改善は中心的な概念であり，そのためのデータを用いた可視化は重要なテーマである[6][9][10]。

しかし，有効である反面，ソフトウェア開発での定量的データの扱いには困難も伴う。上述の通り人的要因や環境要因の影響が大きいため，品質管理運動での統計的手法適用では前提条件とされているデータの安定化（雑音の低減）を実現するのが難しい。さらには，人の活動（プロセス）のデータを扱うため，次のような問題が起り得る：測定活動が形骸化して使用されていないデータが収集され続ける，人やプロジェクトの評価や叱責に測定データが使われるおそれから現場が警戒して測定活動に非協力的になる，極端な場合には虚偽の測定値が報告される[8]。

データ分析に関する文献は数多くあるが，このような適用上の実際的な問題点や対処法に触れたものは少ない。筆頭著者はソフトウェア開発プロセスに関するデータ分析の講義を社会人に対して継続的に実施しているが，講義内容は分析技法，統計ツールの使用法，統計知識の解説などに偏りがちで，適用事例の解説は行うもののデータ分析の実践的なコツや留意点をうまく伝えきれていないというもどかしさを持っていた。そこで，パターン言語の形でこれらのコツや留意点を記述することを考えた。

本論文では，ソフトウェアプロセス改善を組織的，実証的にすすめるためのデータ分析パターン言語の作成について報告する。これは進行中の活動の現状報告である。「ソフトウェアプロセス改善を組織的，実証的にすすめるためのデータ分析パターン言語」という呼称は長すぎるので，以下ではデータ分析パターン言語もしくは単にパターン言語と呼ぶ。

---

<sup>1</sup> Email: komuro@cs.stanford.edu

<sup>2</sup> Email: washizaki@waseda.jp

## 2. パターン言語の構成とその意図

データ分析パターン言語ではパターンを次の3種類に分類して整理している：(1) データ分析のための環境・インフラの整備や準備に関わるパターン、(2) データ分析そのものに関するパターン、(3) データを活用した改善活動に関するパターン。

それぞれの分類の概要は以下の通りである。

### (1) データ分析の準備、環境・インフラの整備に関わるパターン

概要：データ分析では和食のシェフのように素材（データ）を生かして、本来の味を引き出さないといけない。そのためには準備と下ごしらえが肝心である。特に、ソフトウェア開発データでは人的要因、環境要因などから雑音が入りやすいので、雑音を一つ一つ除いていく地道な作業が重要になってくる。ここで、雑音とは分析で着目している（主）要因以外の種々雑多な要因の影響のことを指す。例えば、人やプロジェクトによる測定・報告方法の違い、使用している環境やツールの種類・バージョンやそれらに対する経験度の違い、関係者の関与の仕方や要求・プレッシャーの強さなど。

### (2) データ分析そのものに関するパターン

概要：ソフトウェア開発データの分析という文脈ではデータは（開発・管理）プロセスから生み出される。データを生み出しているプロセスの実態に関心を持ち続け、どのような状態が「良いプロセス」といえるのか、また、それを明らかにする指標は何なのかという理解・洞察を確立する。このような理解・洞察は一朝一夕には確立しないので、何度も試して良くしていくことが重要である。その際、方向感を失って道に迷わないように、地図（先人の結果や現場の専門家の知恵）とコンパス（定性的な結果と定量的な結果のすり合わせ、分析の意味づけ、組織の方針など）が必要である。

### (3) データを活用した改善活動に関するパターン

概要：有効な改善活動につなげるためには人を納得させ動かす戦略が必要である。この戦略の中には、改善のビジョンと価値を共有しながら、目標と現場の要求のバランスを取り改善の方向感を失わないこと、現場の協力を得るため心理的安全性を保障することなども含まれる。スポーツの世界にたとえるとわかりやすいかもしれない：勝利／優勝という共通の目標を目指し各プレーヤーがそれぞれの役割を適切に果たすこと、各自が率直な意見を出し議論し協力していけること、データ分析結果なども活用して戦略を決めパフォーマンスをうまく修正できることなどが強いチームの特徴のように思われる。

各分類に属するパターンの概要を表1に示す。

表1 パターンの分類

分類	パターン名称	問題点（要点）	解決策（要点）
データ分析のための環境・インフラの整備や準備に関わるパターン	ありのままを報告する環境を作る	改善点・問題点があっても現場から相談がなく報告も挙がらない。現場からの測定報告に実際の実績値どおりであるかどうか疑義がある	測定データを人やプロジェクトの評価に使用しないことを明文化し約束する。経営層への報告には集約データを用い、個々のプロジェクト・個人のデータは挙げない
	測定と価値・目的を整合させる	価値を生まない測定活動が実施されている	目的に沿ったメトリクスを使用する(GQM)、測定の価値を見直す
	データを理解し雑音を減らす	データのばらつきが大きすぎて有効な分析ができない	運用定義を明確にする。ばらつきの原因を究明し一つ一つ解決する
データ分析そのものに関するパターン	データ（事実）に基づき考える	勘と主観で判断し有効な解決策がとれていない	データを通して事実を確認する。判断基準を確立していく
	結果を生み出すプロセス（仕組み、過程、要因）に着目する	測定データが結果に関するものに偏っており、アクションにつながりにくい	結果がどのようなプロセスにより生じたのか考察し、改善アクションにつながる仮説を立てる

	一般化と特殊化のバランスを取る	個々の特殊事情を考慮すると深い分析ができるが一般性は失われる	改善戦略（トップダウン／ボトムアップ）にしたがいバランスを取る
	プロセスの実態・実力を理解する	根拠のない目標値設定，形式的なチェックが実施される	プロセスの実力・実態を理解し実績値のベースラインを確立する
	定性的な理解を定量的理解に深化させる	数値のみにこだわり，改善につながりにくい／改善方向が誤った分析をしている。	定性的理解に基づき，測定データからプロセスの実態を理解する．定量分析に定性データを活用する
	キーとなるプロセスを見出す	分析結果が改善アクションにつながりにくい	改善目標への寄与が大きく，制御可能な因子を選んで分析する
	自分たちのデータで確かめる	他部署・他組織の成功事例を形式的に真似して実状に合わない測定分析が実施される． 良い結果がでるといふ先入観がある	自組織のデータを用いて，検証を実施し成功事例がなぜうまくいったのか，自部署でもうまくいくかについて分析する。
データを活用した改善活動に関するパターン	早めに分析・報告し始める	正確性にこだわり，分析作業を開始できない	最初は定性的な分析でもよいので分析作業を開始する．フィードバック結果により分析を改善していく
	成功事例を作り仲間を増やす	（新規の）測定分析の導入には抵抗感を持つ人達（プロジェクト）がいる	協力的なプロジェクトを見つけ効果を実証し，そこから適用を拡げていく
	仮説・検証・学習ループ（試して学ぶ）	結果につながる有効な改善施策が選択・実施されていない	プロセスの理解に基づき仮説を立て，判断基準を決め検証し，結果から学ぶ．この経験に基づき新たな仮説を立て，上記を繰り返す。
	トップダウンとボトムアップの両面作戦	問題を解決する方法が知られているのに現場には届いていない．現場の実態に合わない標準や目標が組織から強制されている。	現場の声に耳を傾けながら，組織レベルの目標に向けて改善策を展開する，トップダウンとボトムアップの両面から攻める。
	改善のビジョン・価値を示し共有する	データ収集を始めとする改善活動に理解・協力が得られない	データ収集がどのような改善につながりどんな価値を生むかの見通し・ビジョンを示す

### 3. パターンの記述形式

データ分析パターン言語では以下のような項目に分けてパターンを記述している：

パターン名， 状況，問題，フォース，解，実施例，関連技術・関連パターン。

ただし，最後の関連技術・関連パターンは選択的であり，記述されないこともある。

また，パターン記述内で他のパターンを参照するときには，<<プロセスの実態・実力を理解する>>のように'<<'と'>>'で囲んで示している。

表 2 にそれぞれの記述項目の意味・意図を示す。

表 2 記述項目とその意味・意図

項目名	意味・意図
パターン名	パターンの名称
状況	パターンの適用される状況，文脈
問題	パターンの解こうとしている問題
フォース	パターンを適用するための前提条件・制約などを生じる，問題の解決に影響を及ぼす要因・力
解	問題に対する解決策
実施例	パターンを実施した例
関連技術・関連パターン	関連する技術・統計技法，パターン

#### 4. パターン例

各分類「データ分析のための環境・インフラの整備や準備に関わるパターン」「データを活用した改善活動に関するパターン」「データ分析そのものに関するパターン」から1つずつのパターンを紹介する。

##### 4.1 パターン名：ありのままを報告する環境を作る

--- Honesty is hardly ever heard. And mostly what I need from you.  
Billy Joel, "Honesty"

###### [状況]

あなたは EPG（組織の改善担当）もしくは QA として、組織や担当プロジェクトのプロセス改善をデータに基づき実施しようと考えており、各プロジェクトからデータを収集・分析し、その結果を各プロジェクトにフィードバックするとともに、組織の改善状況を経営層に報告している

前提として、測定の価値・目的についての組織的な合意が既にある。

###### [問題]

改善点・問題点があっても現場から相談がなく報告も挙がらない、もしくは形式的な報告しか挙がらない。

現場からの測定報告に実際の実績値どおりであるかどうか疑義がある：実績値が良すぎる、同じもしくは類似の報告内容が繰り返されているなど

###### [フォース]

改善をするためには現状の評価をすることが必要となるが、データを提出するプロジェクトや個人の側には、結果を良く見せようとする心理的圧力が働く。特に、測定データが人材評価やプロジェクト評価にも使われる場合にはこの圧力は大きくなる。

また、改善施策を特定のプロジェクトだけにパイロット実施する場合などには、注目を浴びたことが原因で実績が向上することがある。（ホーソン効果）

###### [解]

・データ利用を改善のみに制限することで心理的安全性を確保する。  
測定の目的を明確にする。特に、人やプロジェクトの評価、（失敗した際の）責任追及に使わないことをデータ利用の組織ルールとして明記し、遵守する。

必要なら、さらに具体的な方策をとる。

（例）経営層や上位管理者には集約データや分析結果は開示するが、個人や各プロジェクトの実績値は基本的に開示しない。やむを得ず開示する場合には匿名化処理を実施する。

改善のためには現状把握が必要で、そのためには飾りのない現状のデータ報告が重要であり、無用な責任追及は行われないことを公式・非公式の場で繰り返し伝える。

運用定義を明確にし、測定結果に測定者の主観が混入する可能性を最小化する。データ収集・報告の自動化を可能な限りすすめる。

###### [実施例]

T 君の勤める H 社では最近、社長が交代しそれに伴い管理部門の人事異動が実施された。T 君は EPG（プロセス改善担当）のマネージャーとして開発部門から異動することになった。

前社長はかなりワンマンな性格で、コスト（工数）や生産性に関する報告の数値が悪かった部署やプロジェクトリーダーは激しく叱責されることが続き、結果として実状とかけ離れた報告がまん延する事態になっていた。

このため、開発部署側は EPG のまとめる改善指標の報告に不信感を持っており、測定データの収集にもあまり積極的でないプロジェクトが増えていた。

開発部署出身である T 君はこのような状況をよく心得ており、新任マネージャーとして事態を打開したいと思っていた。旧弊を打破するため、T 君は人やプロジェクトの評価、（失敗した際の）責任追及に使わないこと」をデータ利用の組織ルールとして明文化し、開発部署との会合がある度に明言するよう努めた。実際に報告書では部署レベルの集約データのみを提示し、個別のプロジェクトの情報は示さないようにした。さらに、期初めに行われる全社集会で、新しい組織ルールを説明する機会を作り、周知を図った。

プロジェクトの実態調査から、工数データや開発コード行数の測定について、問題のあることがわかった：測定作業が個人／プロジェクトまかせになっており、負担になっているだけでなく、結果にばらつきが大きい。測定作業にあいまいさがあるため、無意識のうちに生産性が高くなる方向に測定データを報告する傾向が見られた。

#### (1)工数データ

作業コードを基に各人が入力することになっていたが作業コードがプロジェクトの実態にあっておらず、どの作業コードにつければよいのかあいまいであった。週次に一週間分を一度に入力する人が多く、記憶に頼っているためデータ精度も高くない。

#### (2)開発コード行数

メトリクス定義があいまいで、コーディングスタイルによって測定行数が大きくばらついていた。

これらの問題を解決するため、以下のような改善策を実施した：

#### (1) 工数データ

作業コードを実態にあわせて簡素化するとともに、全社で使用していた自社開発のスケジュール管理ツールを拡張して、作業時間を入力できるようにし、作業負荷と入力ミスを低減するようにした。

#### (2)開発コード行数

各プログラミング言語のコーディング基準を全社で統一した。また、そのコーディング基準に基づいたコード行数測定ツールをプロジェクトに配布し、適用した。（この部分は<<データを理解し雑音を減らす>>の実施例にもなっている。）

実施結果：

測定データのばらつきは大きく減少し、各プロジェクトの実態が把握しやすくなった。生産性の算出結果は EPG や QA の定性的な観察とおおむね一致するようになった。

EPG や QA がプロジェクトの会合に出席した際に、改善点の相談を受ける機会が増え、信頼関係が構築しやすくなった。

#### [関連技術／関連パターン]

このパターン適用の前提として<<測定と価値・目的を整合させる>>でどんなデータを何のために測定するかが明確化されている。

このパターンと同時あるいはその後に<<データを理解し雑音を減らす>>が実施され、測定プロセスが最適化されていく。（上記の実施例では同時に実施されている。）

#### 4.2 パターン名：結果を生み出すプロセス（仕組み、過程、要因）に着目する

--- 大切なことは目には見えない  
サン=テグジュペリ「星の王子様」

##### [状況]

あなたはプロジェクトリーダー、または EPG（組織の改善担当）として、自分のプロジェクト／組織の実績値（品質、コスト、納期など）を改善するための（定量的）分析をしたいと思っている。

##### [問題]

最終的な実績値をまとめるだけでは、結果として良かった／悪かったということはいえても、有効な改善アクションにつながりにくい。特に、組織の事業評価、プロジェクト管理の成否の判断を実施する文脈では結果のみが重視されがちである。

（これらの結果データに対して）原因分析等を実施して改善策を立てたととしても、改善されるのは次のプロジェクトであり、現在のプロジェクトへの利点は少ない。

##### [フォース]

因果律によればすべての結果には原因がある。現象として現れた結果だけを分析するよりも、原因を調べれば、より根本的で適切な分析ができる可能性がある

利用可能なデータの種類と量に応じてプロセスの理解の深さは変わってくる

##### [解]

結果を生じる仕組み・メカニズムを考える。ソフトウェア開発の文脈では開発・管理のプロセスを考察することになる。ただし、実施環境等の違いもプロセスの違いととらえる。

改善の目的・目標につながる「良い結果」とは何かを考え、その「良い結果」を生み出す「良いプロセス」は何かを明らかにしていく。

対象となるプロセスの内容が複雑な時、プロセス自体が大きすぎる時には、構造化してプロセスを分割して考察する。

結果に対してプロセスがどう影響しているかについて仮説を立てる。

- ・ 改善活動・分析活動の初期の段階で分析対象のデータがあまりない時  
過去プロジェクトの経験、専門家の知見、他社・他組織の結果などを基に定性的仮説を立てる。その仮説を定量化して検証するためのメトリクス定義についても検討する
- ・ メトリクス定義が既にあり、定量データが利用可能な場合  
データの定性的な意味を吟味しながら相関分析、仮説検定などの定量的分析を実施する。

プロセスが構造化されている場合には、<<キーとなるサブプロセスを見出し>>、結果への影響をモデル化（構造化された仮説を立案）する。

このパターンで見出されたプロセスに対する仮説は<<仮説・検証・学習ループ>>にかけられ、プロセスの理解を深めた上で具体的な改善策へと展開される。検証に成功した場合、実施中のプロセスの実績を良い状態に保つことで、（今後実施されるプロジェクトのみならず）当該プロジェクトが良い結果を得る確率を高めることができる。

##### [実施例]

A 君は、EPG（プロセス改善担当）として自社の炎上プロジェクト（トラブルプロジェクト、問題プロジェクト）を減らしたいと考えている。炎上プロジェクトの定義は組織によって異なるが、A 君の組織では基準以上の赤字を出したプロジェクトのことを「炎上」と呼んでいる。

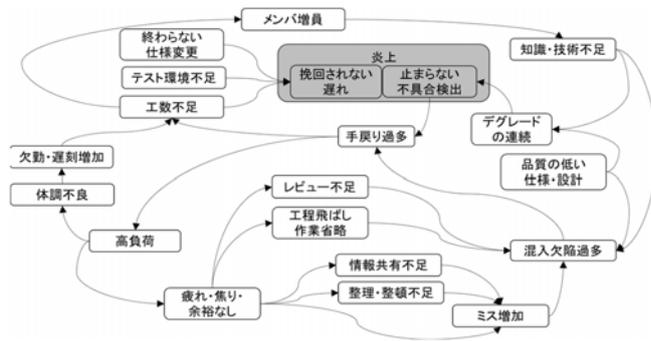


図1 炎上につながる要因の問題構造

最近の炎上プロジェクトを調査し、図1に示す問題構造図を用いて「炎上」につながる要因を洗い出し、関連するプロセスについて考察した。

関係者との話し合いにより、以下の関連プロセスと考えられる問題点を特定した。

- 見積りプロセス：十分な根拠を持たない楽観的な見積りで工数不足が発生している
- 計画プロセス：適切なスキルを持った要員が割り当てられていない
- 要件分析・管理プロセス：要件の引き出し、理解が不十分で要件変更が多発している
- レビュープロセス：十分な（参加者、頻度、時間、欠陥摘出を伴う）レビューが実施されていない
- コミュニケーションプロセス：関係者とのコミュニケーションが十分でない

これらの問題点が炎上につながっているというのはまだ仮説であって、<<仮説・検証・学習ループ>>にのせて改善活動につなげていく。

上記の問題点の記述は定性的なものであるが、<<仮説・検証・学習ループ>>の中でプロセスの理解を深める中で定量化（メトリクス定義）も行い、最終的には改善効果の分析も実施する。

[関連技術・概念，関連パターン]

結果をまとめる段階はおもに記述統計にあたる。

データを生み出すプロセスの考察は推測統計にあたる。

プロセスを構造化・分解して出てきた1つの確率変数のとりうる値の集合は母集団と呼ばれる

このパターンは<<仮説・検証・学習ループ>>と共に実施され、プロセスについての理解を深めていくために用いられる。

4.3 パターン名：仮説・検証・学習ループ（試して学ぶ）

---仮説を立てるといっているのはある意味で勇気のいることである。（中略）  
 初めに立てた仮説は大抵だめだというジンクスがあるからだ。  
 広中平祐「学問の発見」

[状況]

あなたは EPG（組織の改善担当），QA，プロジェクトリーダーとして，組織や担当プロジェクトの改善をしたいと考えており，実績データから改善点を見出したいと思っている。実績データには結果（品質，コスト，納期など）とそれに関係すると考えられるプロセスのデータが含まれている。

### [問題]

何のための改善策なのか目的がわからず、結果として何がよくなったかもあいまい。

改善策を実施しても主観的な評価しか行われず、本当に有効な改善策なのかよくわからない。

改善のために努力したことだけが評価基準となっており、結果が実際によくなっているかわからない。労力の無駄使いになっている懸念がある。

### [フォース]

検証時に他の要因の影響を排除する必要がある

客観的な評価をするためにメトリクス定義が必要

仮説が不成立だったとしてもそこから学べる雰囲気、余計なプレッシャーのない状況が必要

### [解]

現状のプロセスの理解に基づき仮説を立てる。

仮説の評価の仕方を決める。適切なメトリクスで測定されたデータがなければ、新たにメトリクスを定義してデータを収集する。

雑音（他要因の影響）が軽減されるように検証方法を定める。  
軽減の方法として、ベースラインの利用（プロセス安定化）、問題としている要因以外の状況が同じ2つのプロセスデータの比較などがある。

検証作業を実施する。（既存データの分析、パイロットプロジェクト、比較実験の実施など）

結果を分析し、プロセスについて学ぶ。

仮説が成り立たず、期待されていた結果は（最初からは）得られないという場合も多い。

仮説が成立しなかった場合には、検証作業から得られたプロセスに対する知見・経験も加えて新たな仮説を立て、仮説・検証・学習を実施する。

### [実施例]

上流工程レビューの改善

S事業部ではソフトウェアプロセス改善を継続的にすすめている。全社的に実施されている品質向上活動によりテスト工程の手順は確立されているが、テスト工程が長引き、スケジュール超過するプロジェクト、納入後に顧客からクレームがつくプロジェクトもある。

S事業部では開発システムの品質指標としてテスト工程全体でのバグ率（テスト工程での欠陥密度：テスト工程で発見した欠陥数を開発規模で割った指標）を採用していた。

事業部のプロセス改善担当のO君は最近のトラブルプロジェクトの原因分析を行い、要件や設計工程に多くの課題があることを見出した。そこで、上流の品質向上策が最終的な製品品質の向上にも有効であると考え、「上流工程でレビューを徹底すれば、バグ率が低下するであろう」という仮説を立てた。

この仮説を検証するためにレビュープロセスに対する既存の指標である欠陥指摘密度（レビュー規模当たりの指摘件数）、開発規模当たりのレビュー指摘件数、レビュー速度（単位時間当たりのレビュー規模）などの測定データとバグ率との相関分析を実施した。しかし、どの指標についてもバグ率との明確な関係を見出すことは出来ず、この時点では仮説の検証はできなかった。

この検証作業を通して、事業部内のプロジェクトのレビューデータをさまざまな観点から調べたO君はレビューデータそのものにかんがいのばらつきがあり、特に、積極的にレビューを実施しているプロ

プロジェクトとそうでないプロジェクトの間では、レビューの実施回数、各回の指摘件数、レビュー速度などの指標値に相当の開きがあることに気が付いた。

このようなばらつきの大きなデータでは有効な分析もできないと考えた O 君はドキュメントレビュー／コーディングレビュープロセスが安定的に実施されるような改善策を複数のプロジェクトに対してパイロット実施することにした。

具体的には管理図を用いたプロセス実績の分析を定期的に行い、異常値の分析から各回のレビューに対する改善点を見出し、随時改善するようにした。典型的な改善点としては「一度にレビューにかける成果物の規模が大きすぎる」「レビューの事前準備がプロセス通りには実施されていない」「レビューに人数をかけすぎている」などがあつた。また、専門家を招いてレビューをした際には、「レビュー実績が管理限界を外れるほど向上した」という「良い方の異常」が出現することも見出した。

O 君はこのような改善によりパイロットプロジェクトでは製品品質が実際に向上したのではないかと、いう新たな仮説を立てた。そして、この仮説の検証として同時期に実施されたプロジェクトとの品質比較を実施した。図 2 に示すように安定化を実施したプロジェクトの方がテスト工程の欠陥密度は低下している。5%有意水準で検定（ノンパラメトリック検定）を実施したところ、この低下は統計的に有意であることが確かめられた。

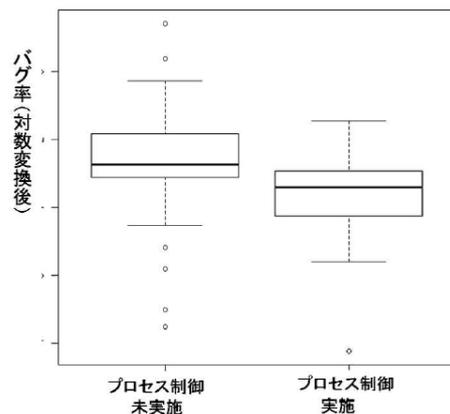


図 2 パイロット実施したプロジェクトと他のプロジェクトのバグ率比較

(注) 「プロセス制御」とは「レビューの安定化」のことを指す

O 君はこの検証結果を受け、改善作業の中で見出したレビュープロセスのベストプラクティス（専門家の招待、レビュー規模・人数の制限、レビュー準備の徹底など）を標準プロセスに反映し、レビューに関する教育も実施することにした。

実施結果：トラブルプロジェクトの原因分析により仮説を立て検証した。この検証は不成功であったが、検証作業から新たな改善策を見出し、その効果検証として、品質向上を確認した。また、次の段階に向かうための新たな改善策（レビューのベストプラクティス）を見出した。

#### [関連技術・概念、関連パターン]

管理図はプロセスの実績データを定量的に分析・制御するのに用いられる標準的な技法である。

このパターンは<<データ（事実）に基づき考える>>、<<結果を生み出すプロセス（仕組み、過程、要因）に着目する>>、<<プロセスの実態・実力を理解する>>、<<定性的な理解を定量的理解に深化させる>>、<<キーとなるプロセスを見出す>>、<<自分たちのデータで確かめる>>とともに実施され、分析に関する仮定を1つ1つ確かめながらプロセスに対する理解を深めていくのに用いられる。

## 5. まとめと今後の展望

ソフトウェア開発プロセス改善のためのデータ分析パターン言語について紹介した。(1)データ分析の準備に関わるパターン, (2)データ分析そのものに関するパターン, (3)データを活用した改善活動に関するパターンの3つに分類して, データ分析に関するコツや留意点を伝えることを目指している。

このパターン言語は, 主に伝統的な計画駆動型開発(いわゆるウォーターフォール型開発)への適用経験を基にまとめたものであるが, 多くのパターンはアジャイル開発等の新しい開発手法の基でも有効であると期待している。

## 6. 謝辞

シェパードをしていただいた平鍋健児さん, WritersWorkshop でレビューしていただいた皆さんから, 多くの励ましと適切なお指摘をいただいたことに感謝いたします。

## 参考文献

- [1] 石川 馨, 「品質管理入門」日科技連出版(1988).
- [2] W. S. Humphrey, “Managing the Software Process” Addison-Wesley (1990).
- [3] M. V. Hilst, E. B. Fernandez, "A Pattern System of Underlying Theories for Process Improvement" PLoP 10 (2010).
- [4] 野中 誠, 小池和彦, 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] J. Choma, E. M. Guerra, T.S. da Silva, "Pattern Language as Support to Software Measurement Planning for Statistical Process Control" PLoP 17 (2017).
- [6] J. Yoder, R. Wirfs-Brock, A. Auiar, H. Washizaki, "QA to AQ Shifting from Quality Assurance to Agile Quality" PLoP 16 (2016)
- [7] M. B. Chrissis, M. Konrad, S. Shrum, “CMMI® Guidelines for Process Integration and Product Improvement”.Addison-Wesley (2006).
- [8] W. S. Humphrey, “Winning with Software” Addison-Wesley (2002).
- [9] C. W. H. Davis, “Agile Metrics in Action” Manning Publications Co. (2015)
- [10] M. V. Hilst, E. B. Fernandez, “A Pattern System of Underlying Theories for Process Improvement” PLoP 10 (2010).

## 付録 A パターン言語の典型的な適用順序

パターン言語の代表的な適用場面を表 A-1 に、各適用場面に沿った典型的な適用順序を図 A-1～A-3 に示す。

表 A-1 パターンと適用場面

分類	パターン名称	データ誤り・ばらつきの軽減	価値・目的に沿った活動	協力・協調	仮説, 見方を深める (モデリング)
準備	ありのままを報告する環境を作る	✓			
	測定と価値・目的を整合させる		✓		
	データを理解し雑音を減らす	✓			
分析	データ (事実) に基づき考える				✓
	結果を生み出すプロセス (仕組み, 過程, 要因) に着目する				✓
	一般化と特殊化のバランスを取る				
	プロセスの実態・実力を理解する	✓			✓
	定性的な理解を定量的理解に深化させる		✓		✓
	キーとなるプロセスを見出す			✓	✓
	自分たちのデータで確かめる				
改善	早めに分析・報告し始める				
	成功事例を作り仲間を増やす			✓	
	仮説・検証・学習ループ (試して学ぶ)				✓
	トップダウンとボトムアップの両面作戦		✓	✓	
	改善のビジョン・価値を示し共有する		✓	✓	

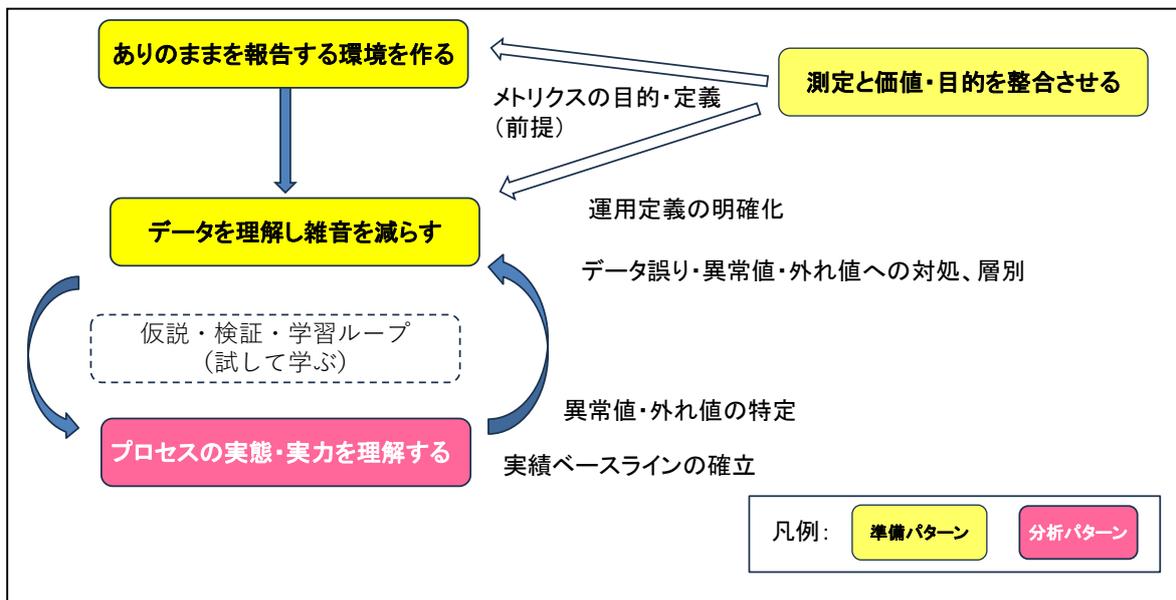


図 A-1 データ誤り・ばらつきの軽減の適用順序

<<測定と価値・目的を整合させる>>パターンにより、メトリクスの目的・定義が明確となる。  
 <<ありのままを報告する環境を作る>>パターンで測定に関する心理的安全性を確保し、<<データを理解し雑音を減らす>>パターンで運用定義を明確化、測定誤り・外れ値・異常値への対処を行う。改善の進行に伴い<<プロセスの実態・実力を理解する>>パターンにより実績ベースラインの確立、異常値・外れ値の特定・分析が行われ、これらへの対処は<<データを理解し雑音を減らす>>パターンで実施される。この2つのパターンの間のやりとりのループは<<仮説・検証・学習ループ>>の適用であるともみなせる。

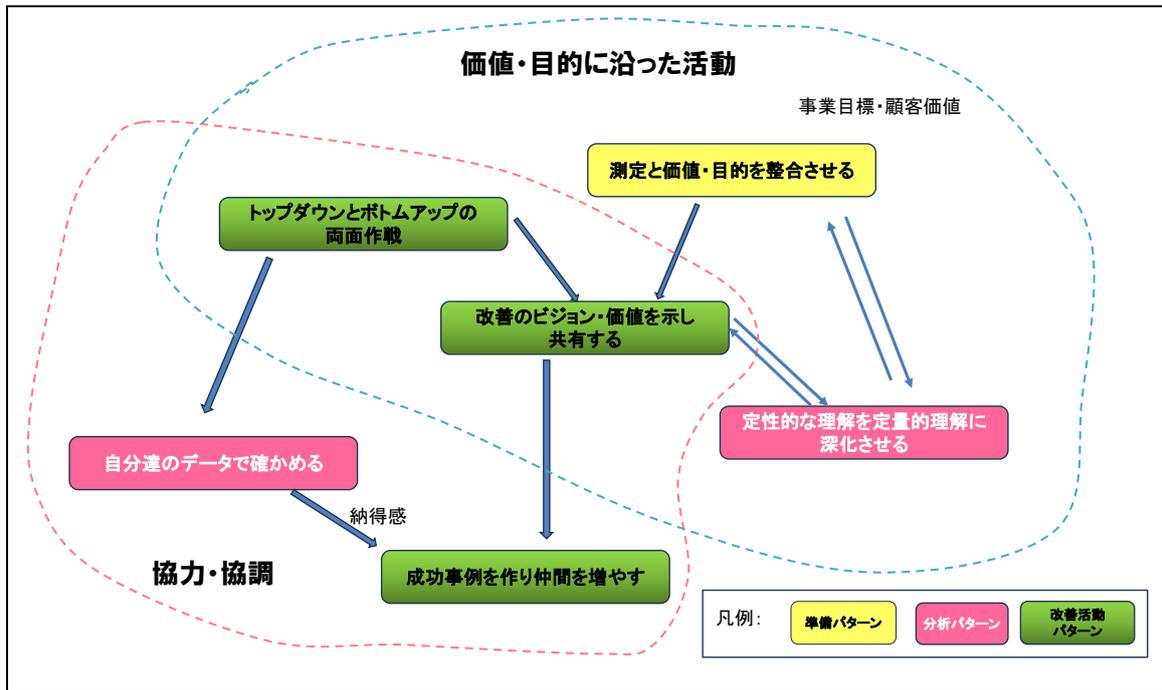


図 A-2 価値目的に沿った活動，協力・協調の適用順序

価値・目的に沿った活動は<<測定と価値・目的を整合させる>>と<<改善のビジョン・価値を示し共有する>>により事業目標・顧客価値などの上位の目的をプロジェクトの活動に反映していく。この際、改善の方向性を確かめながら定量化を進めるために<<定性的な理解を定量的理解に深化させる>>を用いる。

協力・協調は<<改善のビジョン・価値を示し共有する>>により現場に改善の方向性を示しながら<<成功事例を作り仲間を増やす>>で改善の輪を広げていく。この際、現場の納得感を得るために<<自分達のデータで確かめる>>が用いられる。

これら両者の活動をすり合わせるのが<<トップダウンとボトムアップの両面作戦>>パターンとなる。

価値・目的に沿った活動は主にトップダウンにすすめられ、協力・協調はボトムアップの活動を支える。

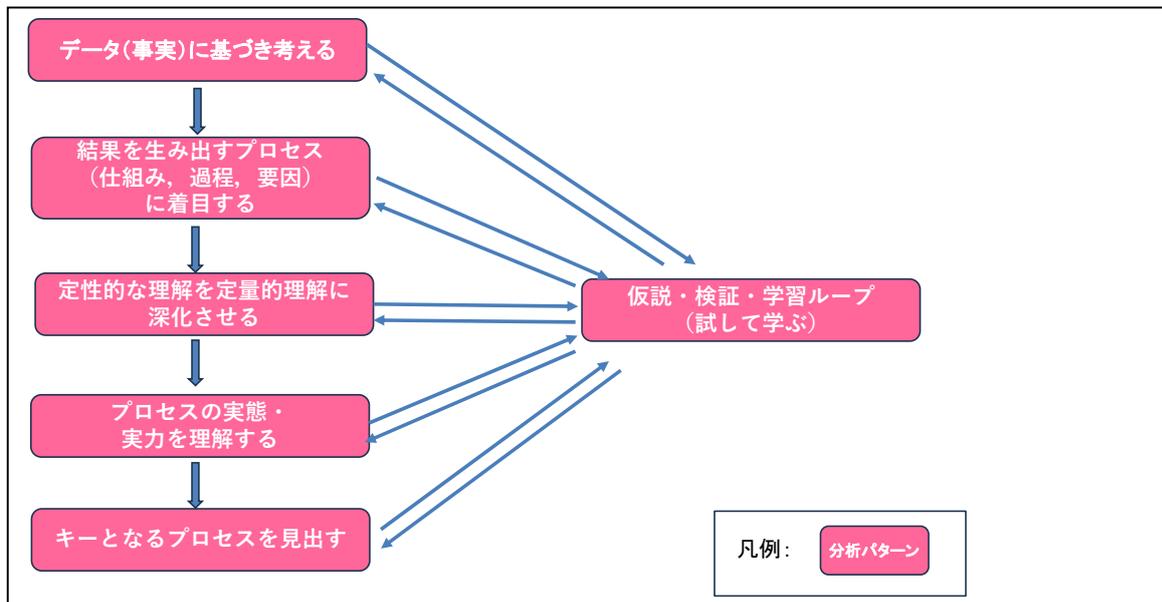


図 A-3 仮説, 見方を深める (モデリング) 活動の適用順序

仮説・検証・学習ループを繰り返しながらデータとプロセスの理解を深める。まずは<<データ(事実)に基づき考える>>によりデータを通して課題・問題点と現状を確認する。次に問題点を生み出している原因となるプロセスを<<結果を生み出すプロセスに着目する>>で調べ、理解していく。この際、改善の方向感を失わないよう<<定性的理解を定量的理解に深化させる>>を用いて分析結果の意味を確認する。プロセスの理解が進んだ段階では<<プロセスの実態・実力を理解する>>により実績データを統計分布(ベースライン)としてとらえることができるようになり、さらに<<キーとなるプロセスを見出す>>ことで、効果的な定量的改善を実施できるようになる。