

Danger, Will Robinson! 5 Anti-Patterns of Agile Adoption

危険！アジャイル導入の5つのNGパターン

著者: **Robert Galen** ロバート・ギャレン



ボブ・ギャレンはアジャイルメソッドのコーチングとトレーニングを行う RGCG 社の社長兼プリンシパルコンサルタントです。彼は、アジャイルのコミュニティで活躍し、アジャイルメソッドに関連する執筆活動や教鞭を頻繁に行っているアジャイル & 認定スクラムコーチ(CSC)です。ボブは *Scrum Product Ownership, a definitive work surrounding agile product ownership* と *Agile Reflections* という成熟したアジャイルパターンにフォーカスした2つのアジャイルに関する本を出版しています。連絡先は bob@rgalen.com で <http://eepurl.com/IAxTD> にてメーリングリストに登録できます。

翻訳: 重白 幸作, PMP

関連トピック: **スクラム**

“アジャイルをやりたいと断固に、情熱的かつ真剣に願うチームをほとんど毎日のように出くわします。彼らはアジャイルを人生を変えるイベントとして見ています。よくあるのが、以下のような考えです。

- リーダーやマネージャにとって: より多くのものを売るための戦力増強要因
- チームにとって: マネージメントの負担をかけずに楽しみながらソフトウェアを作ること
- プロジェクトマネージャにとって: 破壊的な変更で、死ぬほど怖いもの
- その他の人: 成り行きに任せている

ある意味、上述の全ては真実味が溢れています。アジャイルが人生の友となるような面は、一理あると思います。ただし、これらの考えと感情は疑問にもなります。ベテランのアジャイルコーチとして、よくアジャイルの方法と実践に関して質問を受けるのは、何がうまくいって、何がうまくいかないかという点です。

多くの人々が、特効薬的な正解を求めています。例えば、プランニングポーカー(Planning Poker)を用いて見積もる方法は何とかか、スプリントの正しい長さとはという質問です。アジャイルの複雑性や文脈の特性から、ピッタリくるような明確な一つだけの答えはありません。各チームの工程は異なり、自分自身で大部分は解決していく必要があります。この曖昧さが私のクライアントや生徒が腑に落ちない点です。ほとんどの人が、自

分で解決策を模索するのではなく、他の人から解決策を提供されるのを欲しているからです。

*注釈) Planning Poker: アジャイルな見積りと計画の技法 スプリント:開発作業

そうは言っても、アジャイルをうまくやるための生成的な行動やルールがあります。この記事で、貴方のアジャイルにおける“やってはいけない”典型的な NG パターンについて解説したいと思います。願わくば、アジャイルをうまくスタートし、ベストプラクティスや戦術の基準となることを期待しています。(但し、ご自身の頭で考え、ご自身の事情に応用して頂く必要があります)

初期のアジャイル導入: 5つの NG パターン

1.開発者主導のアジャイル: チーム全員参加のアプローチではなく、開発者が導入のほとんどを主導するケースです。120 のスクラムチームがいた、あるクライアントを数年前にコーチしたことがあります。組織の中に 300 人を超えるテスターがいるにも関わらず、全てのスクラムマスターは開発者でした。このことから、見積りと計画にフォーカスしていることがわかります。開発者は重要ですが、アジャイルはチームから尊敬や高く評価されるチームスポーツであるべきです。チームや導入部隊のあらゆる面においてにすべての機能を含めるべきです。

このパターンのサブセットは機能の進捗を計測します。例えば、開発者がユーザーシナリオを作成(コーディング)し、チームの他のメンバーに渡して完成させ、また次のコーディングに取り掛かります。つまり、機能的なやり方です。(フロントエンド開発、バックエンド開発、テスト、分析など)

*注釈)フロントエンド: ユーザーと直接やりとりするソフトウェアシステムの部分を指す。

バックエンド: フロントエンドへの出力を生成する部分を指す。

これはチームの成果創出を部分最適化している方法です。代わりに、関係者全員がそれぞれのユーザーシナリオをなるべく早く完成するために一斉にフォーカスするほうが良いです。

正しいパターン: アジャイルはすべてのメンバーが参画し、透明性があり協力的でオープンなチームプレーであること

2. 分散型チームの言い訳: この質問は週に最低 2、3 回は聞かれます。分散されたチームなのでアジャイルの協力的な特徴のほとんどは難しい。つまり、ペアリング、コードレビュー、立ちミーティング、共同計画、スプリントレビューなどは不可能なので、難しい箇所はスキップできますか? という質問です。皆さんは受け入れ難いかもしれませんが、私の回答はいつも NO です。

分散型チームにとって、強いアジャイルチームのコラボレーションの原理に沿って取り組まなければいけません。組織の上層部(リーダー)の協力を得てコラボレーションを促進する方法(ツール、ビデオ、座席、配置替えなど)を揃え活用し、分散型アジャイルをコラボレーションが少なくなる言い訳にしないことです。

正しいパターン: 分散型チームはコラボレーションし、同じ場所で作業しているチームに負けないくらい革新的に能力を高めること

3. 妨害された作業: 1997年に出版したトム・デマーコとの共著「[Peopleware](#)」でフォーカスの力の事例を取り上げました。皆さんはマルチタスク（並行作業）は作業を早くすることができると勘違いしていますが、結果的にはやり終えた作業は少ないことが多いのです。それでもなお、至る所でバタバタすることがあります。マルチタスクにかかるコストを最小化し、封入し、影響のあるデータを共有することは、良いアジャイルチームの太鼓判です。あるべき姿は、あなたの組織の外側に向きつつ、チームの内側にも目を向ける必要があります。ヒーロー的な火消し役は非常にやみつきになりますが、このような攪拌行為から自立すべきです。

正しいパターン: もっと多くの作業を上手にこなすためにフォーカスはもっとも重要である。コアなアジャイルの特性を活用して、作業妨害とタスクの入れ替えを最小化し行うこと

4. リーダーが主導権を失ったチーム: よくある NG パターンは組織がチームを好き勝手に自己主導型にしていることです。アジャイルチームは組織的なルールや規律なく、自分で判断し行動するという理論に基づいているために起こることです。私がかつてコーチしたあるチームでは、リーダーが何らかのガイドを行うわけでもなく、ガイドを求められた際でも自己判断をしていましたので、論外でした。

リーダーシップは強固なアジャイルチームの形成には必要です。作業が完了したという明確な定義ができる状態、作業バックログや目標設定、計画において、制約条件があるのは正しい状態です。チームは、基本的な定義されたアジャイルの役割を尊重しながら、強固な行動原理を支持する必要があります。

正しいパターン: 自己主導型チームは文字通り主導されることを認識すべきで、効果的な制約条件やガイドラインを適用すること

5. 表面的なアジャイル: スクラムの逆効果(Scrum-But)を指します。これは、チームがアジャイル実践の手軽さを過信して、アジャイルを表面的に採用することで、全体の価値や方法の投資収益率(ROI)を見逃すからです。例えば、ほとんどのチームがスクラムに移る際に、立ミーティング、バックログ、反復を行っています。しかしながら、より強い意志が必要な難しい箇所（スクラムマスターにフォーカス、作業完了の定義など）はよく見過ごしています。

典型的な言い訳は、メンバーがもう少し成熟した後にやるというものです。もしくは、組織が追加費用をかけられないという内容です。人生と同じように、努力と投資を惜しまないことによって変化と影響をもたらすことができます。投資に見合う成果はあるはずだと、よく私はチームに言います。

正しいパターン: 多少難しい場合でもしっかりとしたアジャイル方法論とやり方を適用すること。変更を行う前に、枠にとらわれない手本に投資を行い、成熟度と経験を養うこと

まとめ

筆者が上記のパターンを挙げたのは、文句を言うためでもなく、このようなパターンがみられる組織やマネージャやチームを批判するためでもありません。単に避けてほしい事柄に敏感になってもらうために列挙しました。加えて、あなたの組織で“よいアジャイル”は何かを見極める手助けになれば幸いです。過去を振り返り、自分自身の NG パターンを発見し、よいパターンへと引き寄せてほしいと願っています。

つまり、一般的に知られている考え方に反して、アジャイルは簡単ではありません。訓練や学習、柔軟性、正しいことを貫く精神が必要です。最終目的地にたどり着くというよりも、むしろ発見と継続的改善と言えます。

宇宙家族ロビンの例はやめておきます。なぜなら、彼らは地球へ帰ることが出来ませんでした。あなたには正しいアジャイルを導入し現実的な結果をもたらしてほしいと願っています。

平均評価: 6.67