

製造業に必要なアジャイル開発についての考察

山本 佳和[†] 山本 修一郎^{††}

[†]日本アイ・ビー・エム株式会社, GBS
東京都中央区日本橋箱崎町 19-21

^{††}名古屋大学 名誉教授
愛知県名古屋市千種区不老町

Consideration about "Agile development" necessary for the manufacturing industry.

Yoshikazu YAMAMOTO[†] and Shuichiro YAMAMOTO^{††}

[†]IBM Japan, GBS
19-21, Hakozaki-cho, Nihonbashi, Chuo-ku, Tokyo

^{††}Nagoya University Professor Emeritus
Furo-cho, Chikusa-ku, Nagoya Aichi Japan

概要

様々な不確実性の中で効率的・確実に目的を達成するために「アジャイルのプラクティス」は有効とされているが、製造業の開発現場では、具体的なガイドラインがなかった。本稿では、「仮説」を迅速に「検証」して「定説」化することが製造業の Agile 開発では重要であることを提案する。

Abstract

Generally said that "Agile" practices are effective in achieving the objectives efficiently and reliably in various uncertainties, but there were no specific guidelines at the development site of the manufacturing industry. In this paper, we propose that it is important for Agile development in the manufacturing industry to quickly "Verify" the "Hypothesis" to make it "Experiments".

1 はじめに

「コネクティッド」、「IoT」などの多くのキーワードを背景に、製造業では、様々な開発対象が大規模、複雑化している。これらの多くは、従来の製造業者の開発スコープだけでなく、クラウド環境、それを通じた AI 技術等の様々なサービスとの連携など、新たな開発スコープを対象とすることになった。

このような背景の中、各製造業では、大規模、複雑化し、さらに未経験の開発スコープへの取組みのた

めにソフトウェア開発をアジャイル開発にすべきとの声が高まっている[1]。これは、不慣れな開発スコープに対して、先が見通しきれないため、従来のように最初にすべての要件を定義できず、開発の途中でも要求変更が頻繁に発生することが大きな理由とされている。しかし、この背景から、アジャイル開発の適用に至るにはかなり論理が飛躍している。要件を定義する能力が不足している場合には、従来の開発でも同様の事象が発生することは容易に推察でき

る。しかし、このような組織にアジャイル開発を適用しても問題は解決しない。開発全体として対象とすべき問題が何で、それを解決するための施策が本当にアジャイル開発の適用であることを論理的に明示する必要がある。本稿では、これらの背景に対してアジャイル開発を適用する理由を考察するとともに、それらの状況を整理しながらアジャイル開発を効率的に進める方法を提案する。

2 アジャイル開発適用時の誤解

アジャイル開発を最初に適用した開発メンバからよく聞く話として以下のようなものがある。

1. アジャイル開発はドキュメントを書かなくてよい
2. まずソフトウェアを作ることが重要。作って、動かしてみて、それからどうするかを考える
3. ウォーターフォールは悪いやり方であり、開発期間が長い。すべて、生産性が高いアジャイル開発に切り替えるべき

筆者としては、上記の意見には賛成できない。これは、従来の開発でうまくできていなかったことを解決できないことからのいいわけともとらえられるためである。1.の意見については、2001年に公開された「アジャイルソフトウェア開発宣言」[2]を誤解している。もちろん、不必要なドキュメントを書く必要はなく、そのための時間をソフトウェア開発に使うべきではある。しかし、製造業が携わる大規模な開発では、多くの技術者の関与、システムの保守期間の長さと考えた場合、ドキュメントによる意思の伝達、共有は不可欠である。また、2に関しては、なんの段取りもなくモノづくりを始めることは、あるタイミングでの目的に合わせて確実に価値を提供するソフトウェアを作りあげるという「アジャイルソフトウェア開発宣言」の基本の考え方にも反している。2.を実践した結果、いつまでも目的のソフトウェアを完成できず、「アジャイル開発は自分たちの開発には馴染まない」というさらなる誤解を生むことになる。3.については、従来の開発での間違っただけの開発・管理の仕方を指す場合が多く、本来のウォータ

ーフォール型開発で有効であった「段階的に確実に品質を確保する」という理念のひとつを否定するものではないと考える。

3 アジャイル開発の各プラクティスが有効なケースの考察

アジャイル開発が有効といえるケースを考察するために、まず製造業のソフトウェア開発に起きた変化点を明確に定義し、アジャイル開発のプラクティスがそれらにどのように影響を与えるのかを分析する必要があった。この分析結果を説明するためにいくつかの用語を表 1 に定義する。

表 1 用語の定義

用語	定義
確実性	先行きの見通しが立てられ、問題が発生する要因や、その発生確率が明確であるため、事前に対策をうつことが容易な傾向
不確実性	先行きの見通しが不透明であり、問題が発生する可能性はあるが、何が問題の要因となるのか、また、問題の発生確率なども不明な傾向
無説	ある目的を達成するための方法や考え方が何もない
仮説	ある目的を達成するための方法や考え方の仮定
前提	仮説、定説を定義する際に、その説が妥当であることを保証するための情報。この情報が妥当でない場合、この情報を元にしていう説も妥当ではない。 また、この情報の正しさはあるひとまとまりの組織として共有されるものであるため、組織が異なる場合、その正しさは変わる可能性が高い
検証	仮説にもとづいて見通しが正しいことを証明するための活動
定説	ある目的を達成するための、正しいと

立証された方法や考え方

開発成果物は、その目的の確実な達成のため、正しい論理によって開発される必要がある。また、昨今の開発成果物にはほとんどの場合、説明責任が求められる。これらを果たす開発成果物をつくるためには、その開発成果物によって目的を達成できると判断した定説の積み上げが必須となる。

様々な考えが定説となるまでの流れを図 1 に示す。

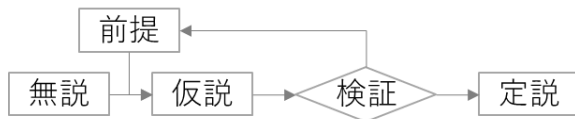


図 1 考えの定説化の流れ

まず、多くの製造業では、新たな開発スコープを対象とすることにより、開発活動そのものの不確実性が高まったと言える。従来の多くの開発は既存技術の改良など、ほぼ同じ開発スコープ内の活動であり、従来の開発で培ってきた知見が定説として組織内では有効活用されていた。このため、新たな開発を開始するにあたり、その開発で達成すべき目標を明確にした後で、社内にある様々な知見を定説として目標を達成する目処をたてて開発を実施してきた。ここで、残る不確実性は主に人為的なミスがほとんどであるため、従来のウォーターフォール型開発で、成果物の具体的詳細化の段階に合わせて確実にレビューを実施し、計画通りに開発が目標に向かって進んでいることを確認しながら進めてきた。従来の開発のイメージを図 2 に示す。

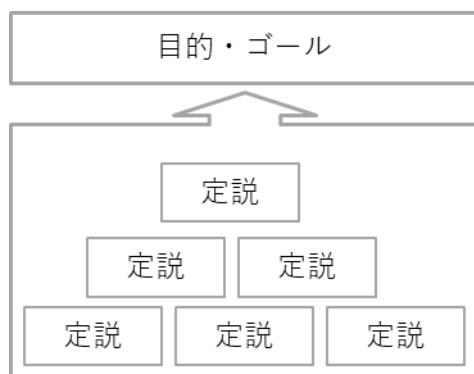


図 2 従来の開発のイメージ

これに対し、新たな開発スコープを取り組むことになった多くの製造業の開発では、様々な不確実性が増していると言える。この開発のイメージを図 3 に示す。

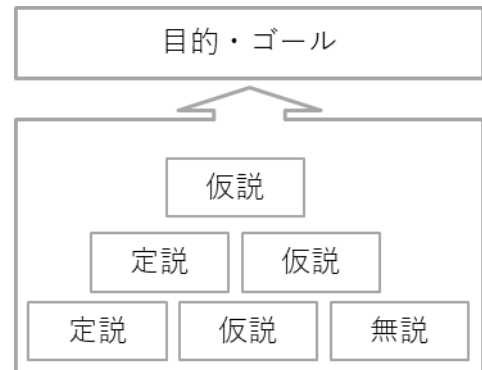


図 3 新たなスコープに挑む開発のイメージ

従来の定説の積み上げでは目的を達成し難く、特に世の中としても新しい取り組みとなる AI の活用などは無説に近いものも数多く存在する。このような状況では、目的達成のために、自組織の定説や世の中の定説を取り入れて目的達成のための仮説を定義することが重要となる。その上で、この仮説が目的達成に貢献することを確認するために仮説の検証を実施する。この際に、以下のプロセスが必要となる。

1. 開発の目的・ゴールを明確にする
2. 目的・ゴールを達成する上で、組織に蓄積された定説で解決可能な「確実なこと」と、無説や仮説で成り立つ「不確実なこと」を明確にする
3. 必要とする仮説と、それに伴う前提を明確にする
4. 仮説を検証して定説にする

ここで、2~4の活動は、従うべきプロセスや決まりきったやり方が存在しない。このため、従来のようなウォーターフォール型開発では、活動がうまくいかない、後戻りが多いなどにより、多くの開発期間を要してしまう危険性がある。この状況では、技術進化の速さが増す中で、製品の市場への投入リードタイムの短縮を迫られる製造業では開発活動が成り立たない。このため、各開発メンバがあいまいな状況から解決の糸口を見出し、試行錯誤しながら動

くもので状況を確認して解を見つけ出すようなアジャイル開発のスタイルが有効である。

5 効果的にアジャイル開発を進めるアプローチの考察

ウォーターフォール型開発にしても、アジャイル開発にしても、開発の目的を明確に定義することは重要である。「何を」、「何のために」、「どうしたいのか？」を明確にしなければ、それを達成するために必要な個々の具体的な活動に詳細化することができないためである。

目的が明確に定義され、その目的の達成のために必要な具体的な活動が詳細化できれば、個々の具体的な活動が既存の定説で解決できるものなのか、できないものかを見極める。場合によっては、個々の具体的な活動をさらに詳細化して、定説で解決できるものとできないものにわけの必要がある。この活動は、明確に定説のみで解決できるものとできないものが明確に区別できるまで繰り返す必要がある。

次に、定説で解決できないものは、仮説を立てる必要がある。この仮説が正しいことを検証により確認できれば、それは定説として定義できる。ただし、仮説には、何らかの前提が必要になる場合がほとんどである。「○○ば××であれば仮説は正しい」というように、その正しさがある条件を満たした場合のみ、その正しさは保証される。これは、その前提の対象により、一般的に定説と言えるものか、ある開発組織内だけで定説と言えるものかを明確にできる。例えば以下のような例が挙げられる。

- 例1) 会社目標が○○であるため、××は正しい
- 例2) 世の中のニーズが△△であるため、××は正しい

前提とした内容は、技術動向の変化、世の中の価値観の変化などにより容易に変化するため、この前提内容のトラッキングは重要な活動となる。この前提が変化した場合、それに基づいて決定した仮説、ならびにその検証により導き出した定説は目標達成のために妥当なものではなくなるためである。

これらのアプローチを実施しながら様々なアジャイル開発の各プラクティスを実施することが重要で

ある。

6 保証ケースの活用

開発目的に対して、必要とする定説や、定説化のために必要な仮説への分解を実施する際に保証ケースを活用できる。

保証ケースとは、ある主張をその主張を成り立たせる様々なサブゴールに分解し、ここのサブゴールでの主張が正しいと認識できるエビデンスを関連づけることにより、全体の主張が正しいということを可視化する手法である。保証ケースの参考モデルを図4に示す。

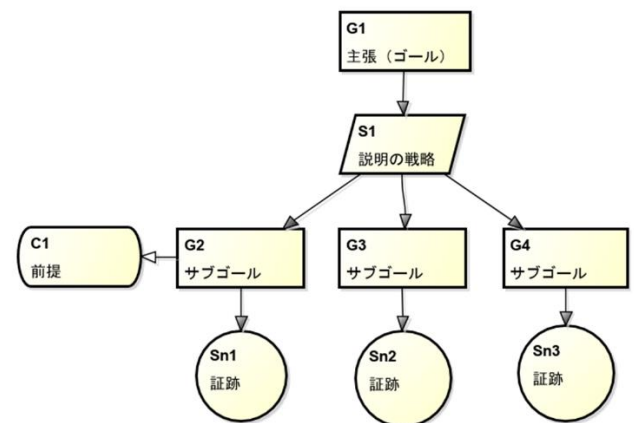


図4 保証ケースの参考モデル

図4に示すモデルのG1には、開発全体の目的・ゴールを記入する。この目的を達成するために必要な定説・仮説をG2~G4に記入する。定説については、その定説の正しさを裏付ける組織内での実績や、文書をSnに記入する。また、仮説のために必要な前提はC1に記入する。

このように保証ケースを使って、問題事象をモデル化しすることにより、開発全体の状況を可視化できる。

また、G2~G3のサブゴールで、仮説となるものは、アジャイル開発のためのバックログとなり、サブゴールごとにイテレーション開発を実施する。そして、仮説を定説化するための証跡をサブゴールに関連づけるSnに記入することで開発全体の確実性を高めることができる。

7 関連研究

安全性についての議論をゴール指向で文書化する手法として、保証ケースが提案された[3]。

小林ら[4]がサービス提供判断に保証ケースを利用する手法を提案し、実際のクラウドサービスの提供判断に適用することによって、有効性を確認している。また、竹内ら[5]は保証ケースを用いて、AI 開発プロジェクトの成功要因を分析する手法を提案している。プロジェクト開始時に提案手法を用いて保証ケースを作成し、成功要因に関連した議論の有無を分析することで、プロジェクトの成功を予測できる可能性が高いことから、AI 実践プロジェクトの導入評価指標としての保証ケースの有効性を明らかにした。

山本[6]は、デジタル変革のためのバランススコアカードを提案している。デジタルバランススコアカードでは、ゴール指向で作成された戦略マップでデジタル変革要求を明らかにしておき、デジタル変革要求のビジネス価値を明確化できる。ビジネス価値と迅速性に基づいてデジタル変革要求に対するアジャイル開発対象の優先順位を定義することにより、アジャイル開発ロードマップを策定することができる。

Traverso ら[7]は、デジタル変革をアジャイル開発によって展開するために、顧客体験指向のエンタープライズ・アーキテクチャを提案している。彼らによる反復型アプリケーション開発方式には、戦略策定、変革計画、アジャイル開発、運用管理からなる段階がある。アジャイル開発で構築されたアプリケーションは、運用管理段階でビジネスに展開される。アプリケーションは顧客からのフィードバックに基づく継続的改善サイクルによって、戦略策定段階で新たなサービス企画となる。

Wedeniowski[8]がエンタープライズ・アーキテクチャによる自動車業界のビジネス変革について紹介している。この中で、自動車業界では「IT 開発と IT 文書化」のアジャイル化が進んでいるとしているが IT 開発はサプライヤによって遂行されるとして、詳細については説明していない。

8 保証ケース利用効果の考察

アジャイル開発は、本稿で述べたように、不確実性の高い開発に適用されるべきである。このような不確実性の高い開発では、様々な側面での状況変化が容易に想像できる。このような状況変化に対して最も注力すべきことは、前提に関わる変化である。いくつかの定説が、ある条件を前提としているため、その前提が満たされなくなる状況変化が発生した場合、その前提に関連づく定説は見直すべきである。対象となる定説に対して、仮説からの見直しを適切に実施しなければ、最終的な目標が達成できなくなる。従来の開発では、この前提が明確に管理できておらず、状況変化に対する対応が遅れ、開発上の大きな問題につながっていた。

本稿で提案した保証ケースを活用した場合、各ゴールに対する前提は明記されるため、この前提を図5に示すような仕組みで管理することにより、状況変化が発生した場合に、関連する前提および、ゴールを迅速に見つけ対応することができる。

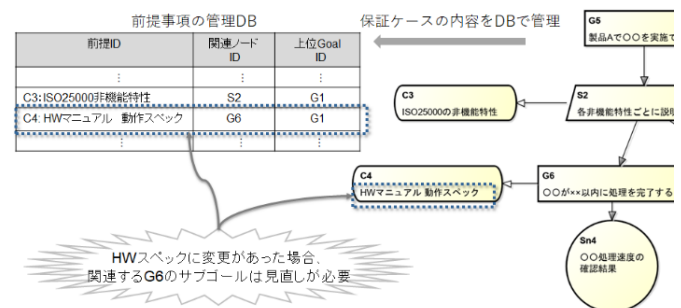


図5 前提管理のしくみ

9 まとめと今後の課題

本稿では、大きな事業変革期を迎えている製造業で適切にアジャイル開発を適用するアプローチを提案した。アジャイル開発で提唱される様々なプラクティスは非常に有効なものであるが、これらを使う意味を考え、適切に適用しなければ開発はうまくいかず、アジャイル開発が否定されることになりかねない。

また、アジャイル開発ではチームメンバー各自が対象となる問題事象に対して、それぞれが考えて行動

する自己組織化されたチームが求められる。このためには、チームメンバが問題事象への取組みに注力できる環境が整えられるべきである。大規模な開発では問題事象のスコープも大きくなるため、様々な管理活動も必要となってくる。これららの管理活動にツール等を利用することで開発チームにとっての無駄な活動を排除することは重要である。

以上のことから、今後の課題としては、開発活動の中での様々な情報の流れに適切にツールを利用して連携させることが考えられる。

まず、保証ケースをアジャイル開発に活用した場合、仮定レベルのサブゴールを要件管理ツールで適切に管理し、アジャイル開発でのバックログとしてシームレスに活用できる仕組みが必要である。さらにこれらの仕組みを前提管理の仕組みと連動させることにより、状況に応じたバックログをアジャイル開発チームに提示することができる。また、この状況に応じた優先度の判断などにAI技術を導入することも課題である。

そのほか、製造業の多くは、その業界ごとに定められた標準プロセスモデルや、安心・安全のために準拠すべき標準プロセスモデルがある。しかし、初期の仮説を検証するフェーズでは、安全を考慮した厳密なテストは必ずしも必要とされない。アジャイル開発を実施しながら、これら標準プロセスに準拠するためには、標準で定義された活動項目から、開発フェーズや、対象にあわせて必須となる活動項目を適切に選択し、アジャイル開発チームに提示できる仕組みが必要である。これにより、標準プロセス準拠のために必要な活動は自動的に提示されるため、アジャイル開発チームは問題の解決に注力する。この状況に応じた必要最低限の活動の抽出にもAI技術の適用の可能性はある。

アジャイル開発を適切に多くの製造業で活用していただけるように今後も研究を進めたい。

参考文献

- [1] 平井直樹, アジャイルの導入と本質, 立教 DBA ジャーナル第 10 号, doi/10.14992/00018685, 2019
- [2] W.Cunningharn, “アジャイルソフトウェア開発

宣言”, <http://agilemanifesto.org/iso/ja/>

- [3] T. Kelly and J. A. McDermid, Safety case construction and reuse using patterns, 16th International Conference on Computer Safety, Reliability and Security, pp. 55–69, 1997.
- [4] 小林茂憲, 山本修一郎, 保証ケースを用いたサービス提供判断方法の提案, 信学技法 KBSE 研究会 111(489), KBSE2011-70, pp. 7–12, 2012.3.8
- [5] 竹内広宜, 秋原史記, 山本修一郎, 保証ケースを用いた AI 実践プロジェクトの成功要因分析, 電子情報通信学会 KBSE 研究会 117(214), pp.7-12, 2017.9.19
- [6] 山本修一郎, デジタル変革に向けたデジタルバランススコアカード DBSC の提案, 電子情報通信学会 KBSE 研究会, 119 (392) , pp.19–24, 2020.1.25
- [7] Giovanni Traverso, Loh WoeiMin, Brian Ng, Customer Experience-Driven Enterprise Architecture: How to Revitalize your DSP Business, W166, The Open Group, 2016
- [8] セバスチャン・ヴェデニフスキー著, 宮下潤子/町村直義/シュタルフ洋子/泉博之[共訳], モビリティ革命:自動車ビジネスを変革するエンタープライズ・アーキテクチャ, 森北出版, 2016