

プロジェクトファシリテーション

価値と原則編

(株)永和システムマネジメント
オブジェクト倶楽部 平鍋 健児 / 天野 勝

Ver1.5 2005年3月2日

Ver1.6 2005年4月13日

Ver1.7 2010年2月24日

Ver1.8 2011年3月29日

オリジナル: <http://ObjectClub.jp/>



このドキュメントは、クリエイティブ・コモンズ・ライセンス（帰属 2.0）の下で提供しています。このライセンスについて詳しくは、

<http://creativecommons.org/licenses/by/2.0/>

を参照してください。原著作者のクレジットさえ示して頂ければ、コピー、修正、配布してかまいません。みなさんのプロジェクトでこれをカスタマイズして使用することを歓迎します。

また、ご意見などを以下のアドレスへ頂けるとさらに嬉しく思います。

平鍋健児
editors2007@ObjectClub.jp

はじめに

PF(プロジェクトファシリテーション)とは

プロジェクトを成功させるために、大切なものはなんでしょう？

私自身、ソフトウェア開発のプロジェクトに多く携わってきましたので、その例でお話します。まず、プロジェクトに参加する個人個人に、プロジェクトが扱う問題に関する「ドメインスキル」が必要です。ソフトウェア開発であれば、最新のハードウェアやソフトウェアプラットフォームの知識、プログラミング言語、デザインパターンやユースケースなどの開発手法の知識とそれを利用した分析、設計、実装のスキルまで、幅広い技術知識とスキルが求められます。さらに、扱うビジネス分野ごとの業務知識も求められます。銀行のシステム開発であれば金融業務、病院のシステム開発であれば医療業務の知識です。これらのスキルは、おもに勉強や実習によって、本、現場、先輩から学んでいくものです。

次に「ヒューマンスキル」が必要です。対人コミュニケーション能力、交渉能力、プレゼンテーション能力などがこれにあたります。ヒューマンスキルがないと、ドメインスキルを持っていてもそれがうまく活かされません。これらのスキルは、本のみで勉強することは難しく、個性や資質と、実際の経験、参加型のセミナーなどによって成熟していくものです。

さて、こういったドメインスキルとヒューマンスキルをもった「個人」が集まって「チーム」を組んだプロジェクトでは、もう1つ大切なことがあります。それは、個人と個人をつなぐ場作りです。個人の能力を100%発揮させ、さらに、個人と個人を繋いで100%以上を導く場。このようにチームを機能させ、活性化することにフォーカスした技術・手法を、「ファシリテーション」と言います。ファシリテーションには、机の配置やホワイトボードの使い方、さらに会議やブレインストーミングの運営方法まで幅広い内容が含まれます。

ちなみに、ソフトウェア開発に限らず、「ファシリテーション」およびそのスキルをもった実践者である「ファシリテーター」はさまざまな分野で現在注目を集めています。街づくりにおいて、住民たちを巻き込んで合意形成をしていくプロセスや、企業の変革活動において、社員を巻き込みながら組織を変えていく手段¹としても、ファシリテーションは有効です。どの場面でも、ファシリテーターは参加者の能力をフルに引き出すことに集中し、自身は触媒のように働きながらチームとしての価値を最大にします。

ファシリテーションとは、スキルを持った個々人の能力を最大限発揮させるチームの場作りです。

¹近年米国では、企業内の会議のあり方がみなおされ、会議体のデザインとそこから価値を生み出す責任者として、CMO (Chief Meeting Officer)なる経営幹部の肩書きも現れました。

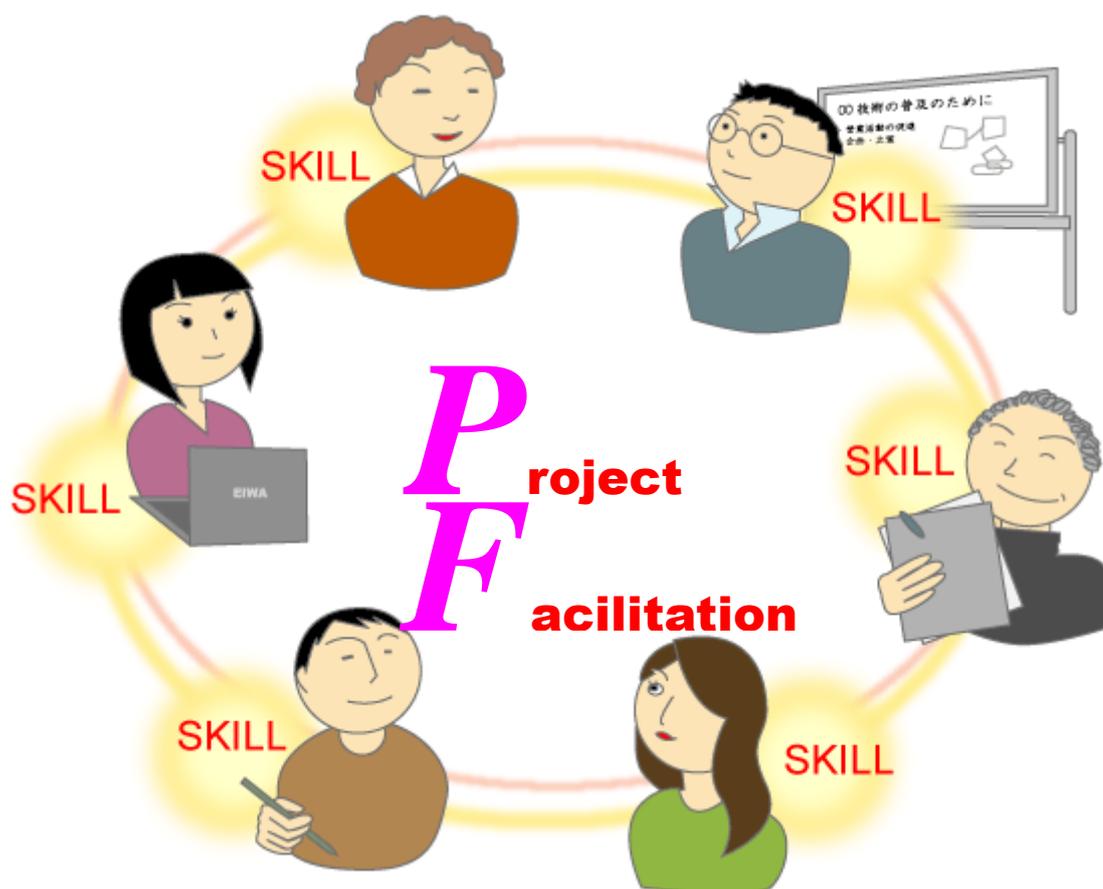


図 1 個々人のスキルを最大限発揮させる場作りが、ファシリテーション

PMとPF

プロジェクトにおけるファシリテーション技術は、プロジェクトマネジメント(PM)の一部と見ることもできます。しかし、PMが「計画達成のマネジメント」に重点を置くのに対してPFは「協調の場作り」に重点を置いています。PMは、計画の立案と実行、差異に注目した管理に重点が置かれ、どちらかと言うと「コマンド・コントロール型」のマネジメントスタイルが背後にあります。これに対してPFは、その場その場の変化に対応し、チームが協力し合って創発的に成果を出していく、「リーダーシップ・コラボレーション型」の新しいチーム作りの形なのです。

PMの重要性は、ここでは強調しておきたいことです。PMは成果を計画的に出していくために必要な技術です。これに対して、PFはプロジェクトの現場を活性化し、モチベートし、協調関係を作るために必要な技術です。PMをプロジェクトの「動脈」とするとPFはプロジェクトの「静脈」といえるかもしれません。静脈が詰まってしまえば、動脈がうまく働きません。



図 2 静脈としての PF

近年、プロジェクトの成功が、PM のみならず PF にも掛かっているのではないかと気づいている人たちがいます。ソフトウェア開発プロセスの分野で例を挙げましょう。

ソフトウェア開発と PF

これまで、ソフトウェア開発プロセスといえば分析や設計、実装、テストの手法とそれらの手順や、成果物を定義したもの、という管理的で味気ないものでした。しかし、2000年に発表された Kent Beck による XP および 2004 年の XP 第 2 版[Beck04]では、「コミュニケーション」、「フィードバック」、「シンプルさ」、「勇気」、「尊敬」といった人間の持っている本質に価値を置き、ソフトウェア開発プロセスをそこに参加する生身の人間の側からの視点で解体・再構築してみせました。また、Alistair Cockburn も Crystal Clear[Cockburn04]において、そこに参加する人々(People)を中心にすえ、プロジェクトルールの作り方からコミュニケーション手法まで、ファシリテーションに大きな力点を置いた方法論を打ち立てています。さらに、Jim Highsmith[Highsmith04]は、「計画—実行」型のプロジェクトではなく「発想—探索」型のプロジェクトにおいて必要とされる、まったく新しいマネジメントスタイルを体系化し、アジャイルプロジェクトマネジメント(APM)という領域を開拓しています。

そして、これらの方法論者たちによる 2002 年の「アジャイル開発宣言」[AgileAlliance02]では、「プロセスとツールよりも個人と対話を重視する」という宣言で、ファシリテーションの重要性が唱えられています。

ここでは、ソフトウェア開発の中で、特にプロジェクトにおけるファシリテーションにフォーカスしてその手法を体系化してみたいと思います。アジャイル開発や XP そのものを取り入れることは難しいプロジェクトや組織もありますが、その場合でも、ファシリテーション技術を取り入れることは可能です。また、ソフトウェア開発以外のプロジェクトでも、プロジェクトファシリテーションを取り入れることができるのです。

21 世紀のソフトウェア開発プロセスでは、ファシリテーションの重要性がクローズアップされています。

QoEL(Quality of Engineering Life)

プロジェクトを成功させる, という視点とは別に, もう 1 つファシリテーションの目的があります。それは, 「みなさんが質の高いエンジニアとしての人生の時間を過ごすこと」です。このものを, QoEL(Quality of Engineering Life)と呼び, エンジニアとして生きる時間の質を表します。この質は, 自身のスキルを向上したい, 高い技術力でよいものづくりをしたい, 顧客から感謝の言葉をもらいたい, 尊敬できる先輩と意見を交換したい, 家庭や趣味と仕事をバランスさせたい, というモチベーションを充足させることで生まれます。逆に, 辛だけの仕事, 労働環境の悪い仕事, 人に成果が認められない仕事, 同僚と喜び合えない仕事は QoEL を落とします。

仕事を楽しく有意義にやっていますか?

この問いに Yes と答えることが, プロジェクトファシリテーションのもう 1 つの目的なのです。

PF の目的

さて, まとめると, 以下のようになります。

プロジェクトファシリテーションの目的は,

1. 個々人の能力をうまく発揮させることで, プロジェクトを成功させること
2. エンジニアとしての人生の時間の質(Quality of Engineering Life)を向上させること

この 2 つの両立が, PF の目的なのです。

価値

まずは, プロジェクトファシリテーションが大切に「価値(value)」から話を始めましょう。プロジェクトファシリテーションでは, 以下の 5 つに価値を置きます。

- 対話
- 行動
- 気づき
- 信頼関係
- 笑顔

チームで 1 つのことに取り組むには, 価値を共有することが重要です。何に価値を置くのか, という「価値観」を合意しておかないと, 実践の場面で判断に迷った場合や, 意見が対立した場合などに, よりどころがなく混乱を招き入れてしまいます。混乱がおこりそうな場合は, 「価値に戻って考えてみよう」, と唱えてチームで話しあうことができます。

では、以下に5つの価値の1つ1つを解説していきます。

対話

1+1を2以上にするには、まず、1と1を足すこと、すなわち個人と個人が話をすることから始めなければなりません。チームとは何か。チームは単なる個人の集合ではありません。それは、一人ではできないことを協力しあって達成する集団なのです。このためには、コミュニケーション路を「リッチ」で「短い」ものにすることが重要です。リッチとは、「メール」や「電話」よりも対面の話し合いを重視する、ということです。「短い」とは、できるだけ近くで仕事をし、話したいときにすぐに相手にたどりつけるということです。

プロジェクトファシリテーションでは、対話を価値とします。**必要な人と、必要を感じたときすぐ、対面で話をしていますか？**

チームとして個人の総和を超える成果を上げるために、「対話」を価値とします。

行動

人は、知識を重視するあまり、行動を起こさずに判断をすることがあります。やってみないで物事を批判したり否定したりすることもあります。しかし、多くの知識は、実際にやってみてはじめて意味や効果に気づくことが多いものです。逆に、知っているだけで実践しなければ、まったく価値のないものです。本に書かれていること、机上の理論、他人から聞いたうまくいった例、などはそのままでは価値をもちません。行動に移して、体をうごかしてみることが大切なのです。例えば、プロジェクトでふりかえりをして、その後行動が伴わなければ意味がありません。問題の解決には、行動が必須です。具体的に行動を起こすこと。とにかくやってみること。そこで気づきを得て、前進できるのです。

プロジェクトファシリテーションの1つ1つのプラクティスも同じです。やってみないと、その本当の効果は感じることができません。道元禅師の教えの中に、「修さざるもの現れず(Without Practice, No Emergence)」という言葉があります。身体性のない知識は智慧にはならないのです。

プロジェクトファシリテーションでは、行動を価値とします。**あなたの言葉に、行動はともなっていますか？**

価値を現実のものとするために、そして気づきを得るために、「行動」を価値とします。

気づき

「気づき」とは、今まで知らなかったものが見えることです。それは、個人の成長、あるいはチームの成長に欠かせないものです。1つのプロジェクトを成し遂げることは、チームの大きなミッションですが、このプロジェクトの中で人は多くのことを学びます。自分が知らなかったこと、誰かから教えてもらったこと、悩んで作り出した新しいやり方、やってみて体で感じた違い…。「ああ、そうなんだ！」と気づくこと、これは人間が自分を変えていく原動力になるのです。

プロジェクトファシリテーションでは、気づきを価値とします。**今日、何かに気づきましたか？気づきを、毎日誰かに話していますか？**

個人そしてチームが成長するために、「気づき」を価値とします。

信頼関係

「信頼関係」は、個人と個人の間で作られた見えない絆です。信頼関係は、すぐにできるものではなく、1 つの困難をともに乗り越えたり、助け合ったりすることで徐々に作られる貴重な関係です。決して裏切らない仲間、誰かがサポートしてくれる、という感覚をチームメンバに対して持つことで、問題解決にあたる場面で行動を起こす勇気が得られます。この安全ネットが、チームの力を作り出すのです。

プロジェクトファシリテーションでは、信頼関係を価値とします。**あなたはチームのメンバーを信頼していますか？チームのメンバーはあなたを信頼していますか？**

行動を起こす勇気の源として、「信頼関係」を価値とします。

笑顔

仕事を楽しんでますか？人によっては、仕事は辛いものであり、その見返りとして報酬をもらっているのだ、という考え方もあります。プロジェクトファシリテーションでは、この立場はとりません。むしろ、人生の貴重な時間を仕事に使うのであれば、仕事そのものが楽しく、やりがいがあるものでなければならぬ、と積極的に主張します。あなたのエンジニアとしての時間の質(Quality of Engineering Life)を高めることが、ファシリテーションの目的の 1 つでした。この楽しさのパロメータといえるものが「笑顔」です。

プロジェクトファシリテーションでは、笑顔を価値とします。**人からの非難をおそれてびくびくしていませんか？冗談を言える雰囲気はありますか？今日、みんなの笑顔は見えますか？**

人生の貴重な時間を楽しく過ごすために、「笑顔」を価値とします。

原則

次に、5 つの価値を実現するための「原則(principle)」を示します。この原則は、後で述べるプラクティス全体に渡る基礎となる考え方を提供します。「価値」だけでは具体的な行動ができません。行動はプラクティスに対応しますが、プラクティスは現場ごとに変化します。この「原則」が、価値とプラクティスを繋ぐ架け橋です。原則は不変です。別の現場のプラクティスを生み出すには、この原則が重要になります。ここでは、プロジェクトファシリテーションの「原則」を紹介します。

- 見える化
- 名前づけ

- 初めよければすべてよし
- リズム
- 問題対私たち (Problem vs. us)
- カイゼン

見える化(Management by Sight)



図 3 見える化の好例

チームがプロジェクトを進めていくには、現在の進捗状況や成果の状況を知らなければなりません。進捗状況を共有するために、エクセルシートをメールで配信する、なんてやり方もあります。しかし、これでは「リッチ」で「短い」コミュニケーション路とは逆方向です。チームが共有すべき情報は、どう扱えばよいでしょうか？

* * *

チームの現在の状態を、目に見えるようにして、壁に貼り出しましょう。進捗であれ、共有すべき情報であれ、壁に貼り出すことによってその情報が語りだします²。

図 3 は、野球の表示板です。この表示板には、現在のゲームの状況のみならず、誰がどんな役割をしているか、審判は誰か、なども示しており、観客、審判、プレーヤーに情報を見える化しています。これは、最もよい見える化の例といえるでしょう。

XP のプラクティスの1つに、「Informative Workspace」(語る作業場)があります。以前は「Open Workspace」(オープンな作業場)と呼ばれていたものをより具体化したものです。Alistair Cockburn は、壁に貼り出したものを「Information Radiator」(情報発信器)と呼びましたし、Ron Jeffries は「Visible Big Chart」と呼んでいます。どれも、自ら情報を発信する「貼り物」の重要性を指摘しているのです。

トヨタ生産方式でも、「目で見える管理」(management by sight/visual management)および「見える化」という言葉でこの重要性を表現しています。目に見えることが、現場の参加型マネジメントの第一ステップです。

² 「アフォーダンス」の用語では、「掲示板が(みずから)情報をアフォードする」となります。

名前づけ(Name and Conquer)



図 4 名前は大切(写真協力:平塚市博物館)

チームが改善をかさねて行く過程で、ノウハウ、知識、ナレッジと呼ばれるものが「気づき」として蓄積されます。最初、それは個人の中にあり、それを口に出すことでチームとして集団化されます。問題は、これをチームとしてどう扱うか、あるいは、チームを超えてどう伝えるか、です。

気づきによって得た知識(ナレッジ)を、チームに定着させたり、チーム外の人に伝えたりするにはどうしたらよいでしょうか？

* * *

気づきによって得た知識に名前をつけましょう。 知識を、書き出したり、他人に伝えたりしていくと、その輪郭が浮かんでいきます。それに、名前をつけるのです。名前を付けることによって、その気づきは「ラベル」を得て、次の気づきへと進む足場となることができます。他人へ強い印象を与えることもできます。名前を付けることで、人間はその存在を認識することができるようになるのです(この事実には、「ヨシュアツリーの原則(Joshua Tree's Principle)」³という名前が付いています)。

問題を解決する方法には、ブレイクダウンによる方法(Divide and Conquer)と、もう1つ、名前づけによる方法(Name and Conquer)があります。名前を付けることでその存在の輪郭をくっきりと周囲から切り取って示すことができるのです。ある人は、アイデアを得たときに「名前をつけておかないと、どっかへいっっちゃう！」という表現をしています。

初めよければすべてよし

チームでミーティングを行う場合、議題の順序に迷っています。また、チームが開発する機能の順序に

³私の好きな Web デザイナーである Robin Williams(映画スターではありません)が、著書 Non-Designers' Design Book の前書きで、「ヨシュアツリーの原則(Joshua Tree's Principle)」という原則を書いています。著者は、ヨシュアツリーという木があることを図書館の本で知り、「こんな木は見たことないな」と思います。ところが、図書館から自宅への帰り道、この木がたくさん生えていることを発見します。人間は、名前を得ることで、その存在を発見する、というのがこの原則です。

迷っています。どのようにチームが扱う問題の順序を選んだらよいでしょうか。

* * *

良いこと、心地よいこと、成功すること、を先に行ないましょう。そして、次にハードな課題を扱きましょう。ミーティングにおいては、よい場の雰囲気が存在し、個々人がこの雰囲気にうまく溶け込むことで、よい会議を生み出します。会議を「安全」で「生産的」に行なうために、アイスブレイカー、よいニュース、褒めること、を議題の先頭に持っていきましょう。

たとえば、Keep/Problem/Try (KPT:けぷと) による「ふりかえり」を行なう際には、まず Keep を出します。これは、「良かったこと」を先に口に出すことを意味しています。自分たちの成功を自分たちで認識することからはじめ、次に問題点を口にします。順序が逆では、全員が問題解決に素直に向えない、という状況が起こりえるのです。

また、開発がはじまったばかりのチームで機能に選択の余地がある場合、まずやさしい問題を1つ片付けてから、次にハードな問題に取り掛かりましょう。これを Alistair Cockburn は”Early Victory”(早期の成功体験)と呼んでいます。最初に小さな勝利を味わうことで、チームは一体感を増し、次のハードな問題の成功率が上がります。

私はかつて週に一度、月曜のチーム会議のときに、「ハイ・アンド・ロー」と呼ぶゲームを行なっていました⁴。これは、週末にメンバーが遭遇した「ハイ＝良かったこと」と「ロー＝悪かったこと」を一言ずつ発表するものです。このアクティビティは、プロジェクトにおいて仕事以外の話題で個々人の価値観の多様性を受け入れて信頼関係を築くという目的があります。ここでも、まず「ハイ」を先に、次に「ロー」を発表するようにしていました。

さらに、人に対してアドバイスをするときもこの原則が使えます。ハードな批判を最初に行なうのではなく、まず、具体的によい点に言及してから、次に改善点に関するアドバイスを行なう。これによって聞き手の受け入れ方が全く違ってくるのがよくあります。コーチング技法にも、「まず褒める」があります。

リズム(Rhythm)



図 5 リズムにのって(サンバのリズム)

チームは、ある一定の頻度で集まったり、何かをアウトプットしたりします。例えば、それは会議であったり、ソースコードチェックインであったり、顧客へのリリースであったり、ふりかえりの時間であったり。**チームがある同期をとるイベントを定期的に行なうタイミングを、どのように設計したらよいでしょうか？**

* * *

⁴ 最初は「グッド・アンド・バッド」という名前でしたが、あるアメリカ映画で家族がこのゲームをやっているのを見て、「ハイ・アンド・ロー」という名前を知りました。(この映画名をご存知の方、教えて！)

週次、日次でイベントを予定しましょう。 人間のリズム感覚は、カレンダーと時計で作られます。例えば、朝会であれば、「毎朝 10 時から」という具合に決めてしまいます。進捗会議や計画会議は、「毎週月曜日」と決めてしまいます。

プロジェクトが始まったら、ファシリテーターの重要な役割は、プロジェクトチームのリズム刻む会議体のタイミングと目的を設計してしまうことです。プロジェクトが軌道にのると、この会議がハートビートとなってプロジェクトをドライブします。リズムが行動の搬送波となり、プロジェクトは前にすすみます。人間が行動を起こすには、リズムが必要なのです。

問題 対 私たち(Problem vs. Us)

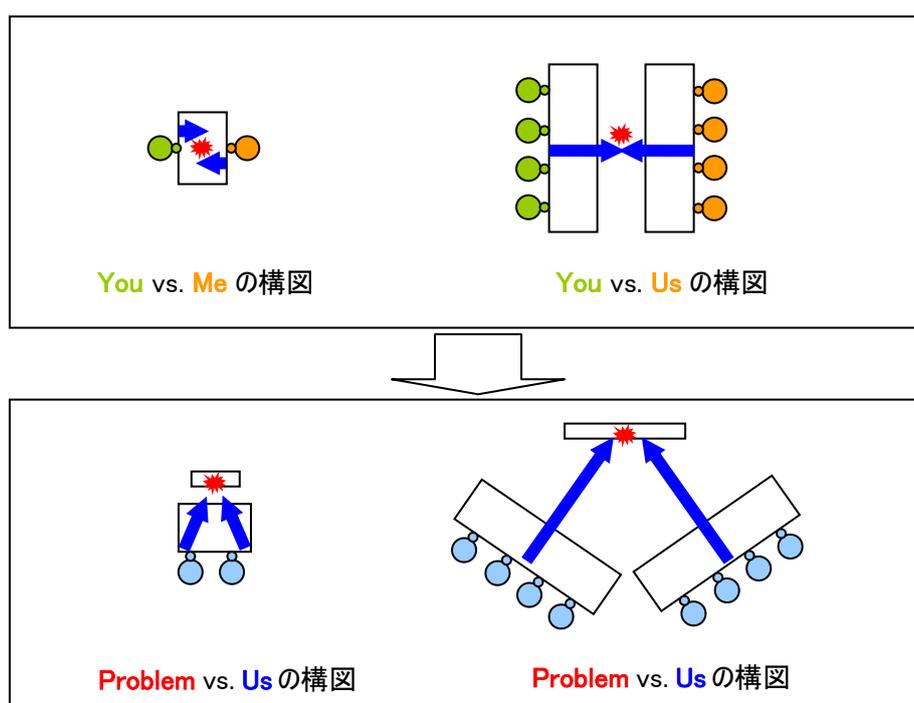


図 6 視線の先に問題を置く

2人で問題についての話をしたり、あるいは複数人で話をしたりするときに、本来は敵対関係ではないのに敵対してしまうことがあります。共通の問題を解決しようとして集まったにもかかわらず、お互いのあら探しになってしまった経験はありませんか？ **複数の人が話をするとき、敵対関係にならずに問題解決という本質的な議論にフォーカスするにはどうしたらよいでしょうか？**

* * *

席の並べ方を工夫し、全員が同じ方向に向くようにしましょう。そして、その視線の先に、ホワイトボードやコンピュータ、プロジェクトのスクリーンなど、いま解決しようとしている問題を置くようにしましょう。

相対で座ると、どうしても「You vs. Me」や「You vs. Us」という対立関係を象徴する構図が生まれます。

座席を工夫し、同じ方向を向くようにします。そして、問題点(Problem)をホワイトボードなどの「第三者」に書き出し、それに向かってみんなで話し合いをします。こうすることによって、「問題 対 私たち」という構図を作り出し、全員で同じ問題を解決しようという姿勢を作り出すのです。

この原則を使った実践には、ペアプログラミング、ペアボード、ソフトウェアかんばん、などさまざまなものがあります。また、この手法は、席に限ったことではありません。議論のテーマ(議題)の書き方 1 つをとっても、工夫してこの「問題 対 私たち」の構図を作ることができます。

カイゼン



図 7 一歩ずつの積み重ねが、大きな結果となる

現状に満足はしていない。問題があるのは分かっている。問題をそのまま放っておいたらもっとひどい状態になってしまうだろうが、その問題を解決するには時間も取れないし、解決しようとして現状よりも問題を深刻にしてしまうかもしれない。現状の問題に立ち向かっていくにはどうしたらよいでしょうか？

* * *

現状をちょっとでも良くできる、今の自分達できることから問題を解決しましょう。そして改善を継続していきましょう。

大きな問題に対して、大きな解決策を考えて行動しようとしても、なかなか行動が起こせなかったり、行

⁵ 改善を継続していくこと(Continual Improvement)をカタカタで「カイゼン」と呼びます。

動できても問題がなかなか小さくならなかつたりします。まずは、「行動」の原則に則り、小さな問題に対して少しでも良くなるような改善策を考え、行動に移していきましょう。小さな問題は解決しやすく、成果を数値化することは難しくても、現場のメンバーはその成果を肌で感じ取ることができ、成功体験を得ることができます。成功体験の積み重ねが、前進する原動力となります。小さな問題を解決していくなかで、自分達も成長しますから、昔は大きいと感じていた問題も、気づけば自分達で解決できる程度の問題になっていることでしょう。

問題は思わぬところから忍び寄ってきます。問題を解決しても、新たな問題に遭遇するでしょう。問題が尽きることはありません。継続的に改善をすることで、やっと現状を維持できるのだ、というぐらいの想いが必要かもしれません。問題に対して自分達で取り組んでいくにはふりかえりが強力な武器になります

実践

さて、「価値(value)」と「原則(principle)」を解説したことで、プロジェクトファシリテーションの目指すところが見えてきたと思います。しかし、具体的に目的を達成するには、何を行動として起こせばよいでしょうか。ここでは、プロジェクトファシリテーションの「実践(practice)」を紹介します。具体的な実践として、2つ簡単に紹介しましょう。

- 朝会
- ふりかえり
- 偏愛マップ
- 進捗(バーンダウン, かんばん)

ここに書かれていない他の実践もたくさんあります。また、これらの実践は、「ふりかえり」によるフィードバックで、日々変化します。ここでは、簡単な紹介にとどめ、詳しくは、『プロジェクトファシリテーション実践編』で解説することにします。

朝会

毎朝、全員で集まって 15 分以下でミーティングを行ないます。目的は、チーム内の情報共有と、仕事を始めるにあたって、朝のスタートを気持ちよく切ることです。

チームリーダーなどが司会を行い、全体的な進捗状況や伝達事項を共有した後、各人が以下の 3 つについて話します。各人の話題をこれ以上増やすと、15 分では終了できなくなります。もし、問題点の討議などが必要であれば、司会者は別のミーティングをアレンジします。

このミーティング会議は、立ったままで行なうことが多い⁶ため、スタンドアップミーティングとも呼ばれます。また、このミーティングは、チーム全体の進捗が貼り出されている貼り物の前で行なうのが良いでしょう。



図 8 朝会の様子(進捗を見ながら)

朝会のやりかたについては、別途『プロジェクトファシリテーション—実践編:朝会ガイド』に記述します。

ふりかえり

チームは、定期的に、その期間のふりかえり(反省、回顧)を行ないます。繰り返し型開発を採用していれば、イテレーションの終了時に行なうのが自然です。そうでない場合でも、1週間～1ヶ月程度の期間で定期的に行なうのがよいでしょう。ふりかえりの目的は、その期間で行なったことから「気づき」を得て、よりよい仕事ができるように自己改善を行うことです。個人個人が思っていることを、言葉に出して言ってみる。個人のなんとなく思っていることが形になり、さらにチームで共有されて「気づき」になります。単に悪かった点を上げる、という「反省」のみではなく、その期間のよかった点、次の期間で試してみたい改善策なども話し合います。

もっともポピュラーなふりかえりのフォーマットは、Keep/Problem/Try (KPT:けぷと)です。この手法では、ホワイトボードや模造紙を3区画に区切って、それぞれの区画に Keep(良かった点＝継続したいこと)、Problem(問題点)、Try(試してみたいこと)を書き出します。

⁶ 時間を短くする効果がある。

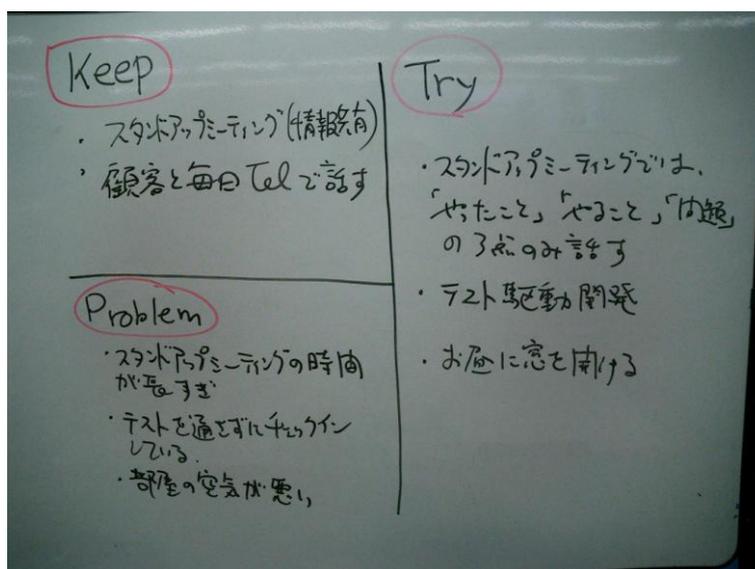


図 9 KPT(Keep/Problem/Try)の例

その他にも、時系列に沿って、個々人の感じたことを書き出す「タイムライン」などの手法があります。

「ふりかえり」のやりかたについては、別途『プロジェクトファシリテーション—実践編：ふりかえりガイド』に記述します。

まとめ

この小冊子では、「プロジェクトファシリテーション」の「価値」と「原則」をまとめ、2つの「実践」として「朝会」と「ふりかえり」を紹介しました。

これらの手法を使って、あなたのプロジェクト・現場がより活性化し、プロジェクトファシリテーションの目的である、「プロジェクトの成功」と「エンジニアとしての時間の質の向上」の両方が得られることを望んでいます。うまく行った、行かない、などのフィードバックをぜひ頂きたいと思います。

実践の詳細については、『プロジェクトファシリテーション実践編』を参照ください。また、実践するにあたって必要なツール(小道具)については、『プロジェクトファシリテーションツール編』を参照してください。

以上で、PF の価値と原則編はおしまいです。(editors2007@ObjectClub.jp)まで、コメントおまちしています！(平鍋)

謝辞

構想段階からアイデアを頂いた、懸田剛さん、安井力さん、有用なコメントをくれた岡島幸男さん、角谷信太郎さん、ありがとうございました。編集、Web への掲載を推進してくれた田中里実さん、イラストを描いてくれた、伊登左和さんに感謝しています。バージョン 1.0 公開後にコメントを頂いた、安井昌男さんはじめ、多くの方々にも感謝します。

コメントつき参考文献

[大野 78] 大野耐一、『トヨタ生産方式—脱規模の経営をめざして』、ダイヤモンド社、1978
トヨタ生産方式の根源には、現場のモチベーションがある。トップダウンでなく現場から改善がおこる環境づくり、人づくりの大切さ。

[佐々木 94] 佐々木正人、『アフォーダンス:新しい知の理論』、岩波科学ライブラリー、1994
「アフォーダンス」という新しい概念についての入門書。多くの気づきが得られる。

[平木 93] 平木 典子、『アサーショントレーニング—さわやかな「自己表現」のために』、日本・精神技術研究所、1993
相手を思いやる気持ちと、自分を主張する気持ち、そのバランスの取り方について。相手を優先しすぎてぐちっぽくなったり、逆に自己主張がすぎて煙たがられたりしないよう、「さわやかな自己表現」をめざしましょう。

[本間 04] 本間 正人/椿 景子、『[入門]キャプテンシップ・マネジメント』、PHP 研究所、2004
マネジャーとプレーヤー、この中間領域でもっともいきいきと活躍すべき「リーダー」に求められる資質、キャプテンシップについて。

[平鍋 05] 平鍋健児、『プロジェクトファシリテーション実践編:朝会ガイド』
([http:// ObjectClub.jp/](http://ObjectClub.jp/)), 2005
ここで書いた価値と原則編の、最初の実践ガイド。「朝会」の実践的な回し方について。

[平鍋 05] 平鍋健児、『リーンソフトウェア開発と「見える化」』
<http://ObjectClub.jp/event/2004christmas/detail#keynote>, オブジェクト倶楽部クリスマスイベント 2004

[Beck04] Kent Beck, *Extreme Programming Explained: Embrace Change(2nd)*, Addison-Wesley, 2004

2000年にソフトウェア開発方法論のパラダイムシフトをやったのけた「事件」、XPの原点。2004年に第2版が出版された。方法論の体系は、「価値」、「原則」、「実践」の三階層。

[Highsmith04] Jim Highsmith, *Agile Project Management*, Addison-Wesley, 2004
「Plan-Do型」のプロジェクトマネジメントではなく、「Envision-Explore」型のプロジェクトマネジメントの新しい枠組み。この本でも、方法論全体を、「価値」、「原則」、「実践」の三階層にまとめている。

[Cockburn04] Alistair Cockburn, *Crystal Clear: A Human-Powered Methodology for Small Teams*, Addison-Wesley, 2004

ソフトウェア業界の「文化人類学」。どういったチームが高い生産性を挙げ、創造的に共同作業するか、を観察、そこで発見される特性についてまとめている。ほとんどのチームが「ホワイトボードを使っている」という発見。チームを「安全地帯」に導く、というコンセプト。本書では、方法論を「特性」、「戦略」、「テクニック」の3階層でまとめている。

[AgileAlliance02] Agile Alliance, *Agile Manifesto*, <http://www.agilemanifesto.org/>, 2002

4つの宣言文は、現在のソフトウェア開発に大切なことを、強く胸にきざませてくれる。

[Poppendieck03] Tom/Mary Poppendiek, *Lean Software Development*, Addison-Wesley, 2003, 『リーンソフトウェア開発』, 日経 BP 社

ソフトウェア開発に「トヨタ生産方式」を取り入れる方法。7つの原則と22のツールキット。トップダウンでなく、現場から、チームからのカイゼン活動。

[Kaner96] Sam Kaner, *Facilitator's Guide to Participatory Decision-Making*, New Society Publishers, 1993

ファシリテーションを使った「参加型意思決定」への導き方。議論の方向コントロールの手法。

[Kerth01] Norman L. Kerth, *Project Retrospectives*, Dorset House Publishing, 2003

プロジェクトのふりかえり(Retrospective)の方法。グラドルールの設定やタイムラインなどの手法について詳しい。ワインバーグが序文を書いている。