

職種を超えた連携におけるチームビルディング適用とその効果評価

山川 紘明

デンソーテクノ(株)

yamakawa@densotechno.co.jp

要旨

ソフトウェア開発は人的資源の影響を強く受ける。スキル面だけでなく現場力に当たるチームの状態が注目されている。この経験論文は、デザイナーとソフトウェア技術者による職種を越えたチームの現場力向上について実践した報告である。技術、人の両側面からチームビルディングに取り組み、その効果評価を行った。評価には、先行研究で使われた計測方法を用いた。社内、グループ会社、IT 業界において計測結果の統計的な分析を行い、統計値にてチームビルディングによって非常に大きな差が出ていることを確かめた。チームビルディングが現場力に与える影響を考察し、技術、人に基づくチームビルディングが職種を超えた連携に有効であることを論ずる。

1. はじめに

「現場力」をテーマとする製造業における研究では、経験的な現場力の定義として、チームの自立的な問題解決能力、OJT(On-the-Job Training) など技能の伝承力、チームメンバが共有する強いモチベーションなどが上げられている。IT 業界における現場力についても近年研究が行われており、「仲間意識」や「役割意識」などが該当することが分かってきている。また、この現場力がプロジェクトの成否に大きく影響することが分かっている。しかし、製造業の現場力と比較すると IT 業界の現場力は 30 点と言われている[1], [2]。

我々は、カーナビゲーションシステム(ナビゲーション)の開発を行っている。ナビゲーションは、多機能化、高機能化と価格競争が激しく、車の開発期間短縮に伴い開発期間も短くなっているばかりか、車種、仕向地(国や地域)など多くのバリエーションにも対応しなくてはならない。こうした背景から、ナビゲーション開発においては、従来の品質を保ち、かつコスト低減と短納期厳守でリリースしていく必要がある。本稿では、この課題に対し、新たなリソースの投入ではなく、チームの現場力を向上させるア

プローチを行ったので報告する。

ナビゲーションの特徴の一つは、高度の GUI と多機能の連携にあり、その仕様を決めるデザイナーとソフトウェア技術者の連携が、生産性や納期に大きな影響を与える。これまでは、デザイナーとソフトウェア技術者の仕事は独立したチームで役割分担し、開発を進めていた。この方式では、高度の GUI と多機能の連携を達成できないことから、デザインを意識した UX(User eXperience)指向を実現するためデザイナーとソフトウェア技術者が一体となって開発するスタイルを目指した。具体的には、チームを 1 つにし、両者の価値を高めることが現場力の向上に繋がるのではという着想のもと、職種を超えた連携に取り組んでいる。

本稿は、以下の構成になっている。次の 2 章では、職種を超えた連携の必要性とその実現における問題点を論じる。3 章では、チームビルディングの取組を紹介する。4 章では、取り組んだ活動に対する結果を示し、5 章で考察、効果を議論する。最後の 6 章でまとめる。

2. 職種を超えた連携の必要性と問題点

2.1. 連携の必要性

iPhone やそのほかのタッチ・デバイス(指でタッチすることで操作できる端末)に代表されるように、UX および UI を起点とした情報技術(IT)の革命が起きている。「システムが優れた UX を提供する UI か否か」、によって、情報システム自体の資産価値も変わってしまう時代になっている[7]。ナビゲーションも例外ではない。

我々が担当するソフトウェア開発は、デザイン情報をソフトに取り込むなど、デザイン要素が大きく関係する開発となっている。これまでは、デザイナーとソフトウェア技術者の仕事は独立したチームで役割分担し、開発を進めていた。しかし、上述したように、与えられた仕様からプログラムを作るという考え方から、デザインを意識した UX 指向で作る考え方に変化させる必要がある。そのために

は、両者が独立したチーム構成で、それぞれ開発し、デザインとソフトウェアの結合時に初めて UX を意識するのでは遅い。開発当初からデザイナーとソフトウェア技術者が連携し、UX を意識して開発を進めることで、生産性や納期に大きな効果をもたらす。

また、デザイナーとソフトウェア技術者がお互いを知ることで、技術の幅が広がり、両者の価値が向上し、結果として現場力の向上にもつながる。

これらのことから、デザイナーとソフトウェア技術者の連携を開始した。

2.2. 連携での問題点

職種を超えたチーム連携を開始し、ソフトウェア技術者、デザイナーそれぞれにいたマネージャーも一人とし、組織としても1つとした。また、作業スペースも席を隣接させ、物理的な距離を縮めた。

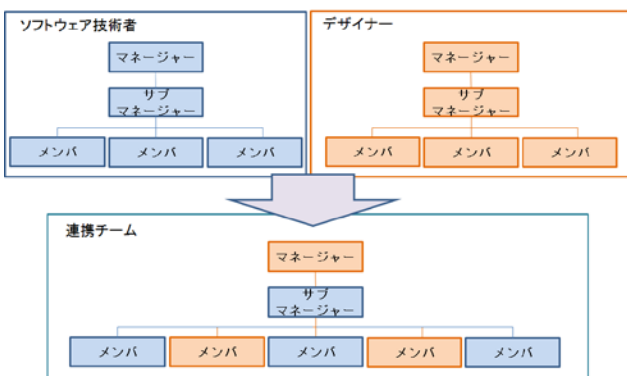


図1. チーム連携前後の組織構成

しかし、組織編成が変わり、物理的な距離が縮まっても容易に連携が行えるわけではない。次に示す、連携を阻む大きな3つの壁が考えられる。

(1) 技術の壁

ソフトウェア技術者とデザイナーでは、今まで作業の進め方や、ツール、成果物等、大きく異なっている。また、コミュニケーションを図ろうとしても、互いの専門用語が理解出来ない。双方、相手が何を知らないのかも把握できない状態であった。このような状態でお互いの技術を習得することは困難である。

(2) 機会の壁

これまで、それぞれ独立したチームで開発を進めてお

り、話す機会は少なかった。技術の壁とも関連するが、組織を1つにしたところで、話すテーマ、議題がなければ両者の話す機会は増えない。

(3) 心の壁

今までのやり方を変えるということは、少なからず負担が発生する。今までのやり方で開発を行ってきた実績があるデザイナー、ソフトウェア技術者にとっては「やりたくない」「なぜそこまでしないといけないのか」という感情が生まれると想定された。

3. チームビルディングの取り組み

チームで新しいことに挑戦する場合、「技術の壁」「機会の壁」にのみ着目して活動しがちである。世間で良いとされている手法や技法を取り入れ、アプローチを行うことが多い。もちろん、このアプローチも有用ではあるが、やりたくない、知りたくないという「心の壁」が残っている状態での活動は「やらされ」となり、活動に身が入らない、活動の本質を理解しないといった状況に陥ることが想定される。そのような状況で連携を行っても、無意識のうちに自身の担当領域に線を引き、いずれ「技術の壁」「機会の壁」が再び現れることになる。1章でも記載したが、「仲間意識」や「役割意識」のような、人の心理がプロジェクトの成否に大きく影響することが先行研究でも証明されている。そのため、デザイナー、ソフトウェア技術者の連携は技術、機会に着目した活動だけではなく、人に着目した活動も必要であると考えた。

我々は適応的開発手法であるアジャイル(スクラム)開発に挑戦することとした。それは、スクラム開発の「チームを単位に物事を考える」「チームメンバ全員がリーダー」という考え方が、技術、機会、心の壁全てを乗り越えるのに適しているのではと考えたからである。しかし、スクラム開発はあくまで仕組みであり、この仕組みを使ってどう自分たちの活動に落とすかが大事である。そのため、我々のチームビルディングでは、スクラム開発の良さを活かしつつ、デザイナー、ソフトウェア技術者の連携を行うためにはどうすればいいかを考えて活動を行った。

3.1. 技術的側面からのアプローチ

3.1.1. 相互の技術習得

一般的に、数(量・規模)が増えると指数関数的に複雑度が増すが、数が少ないうちは比較的容易に理解で

きるという原理がある。そこで、「スプリント」という考え方に着目した。スプリントは小さい範囲を短い期間で繰り返すという考え方である。小さい範囲であれば、ソフトウェア技術者はデザイナーの仕事、デザイナーはソフトウェア技術者の仕事に挑戦することが出来るのではないかと考えた。お互いの仕事を体験すれば、成功、失敗体験から学びを得て、他職種の仕事を理解できるようになる。また、両者が互いに教える、教えられる立場になることで会話も増えると共に、体験を共有することで共通の話題も生まれると考えた。

ただし、両者は他職種の技術が何も分かっていない状態であったため、デイリースクラムにて、問題や困っていることを積極的にヒアリングし、メンバ全員で共有することでサポートできる体制も充実させた。これにより、相手が何を知らないのかを理解にし、必要な時に即座に意思疎通ができる機会を増やし、両者の知識共有を図ることを行った。

3.2. 人的側面からのアプローチ

心の壁はなぜ生まれるのかをチーム全員で話し合った。そこでは下記のような意見が聞かれた。大きく分類すると3つに分けられる。1つ目は、「なぜ連携しないといけないのか」「メリットがよく分からない」という目的が不明確であるという点である。2つ目は、「どこまで考えればいいのか分からない」「相手が何をしてほしいのかが分からない」という役割が不明確という点である。ここについては、相手の技術を理解できていないという技術の壁も関連している。3つ目は、「仕事が増えるから嫌だ」「面倒くさい」といったモチベーションによるものである。

そのため、目的、役割、モチベーションの3つに着目した活動が必要だと考えた。

組織行動論でも、チームとして活動するためには、目的の共有、役割の共有、モチベーションの3つが必要であるとされている[4][5]。社会科学の知見からも、3つに着目することは有用だと考えることが出来る。

3.2.1. 目的の共有

チームの目的の共有は、チームが形成されて行く「チーム立ち上げ期」と、本格的にチームが機能する「チーム活動期」では大きく異なると考え、取り組みは分けて行った。価値観や文化の異なる両者の連携では、意見が食い違うことが容易に想像できた。そのため、チーム立ち上げ期に正しくチームの目的を理解し、意見が食い違った

際に自分たち自身が立ち戻る先を明確にしておく必要があると考えた。ただし、最初に話し合いを設けただけではなかなか定着はしないことが考えられた。そのため、チームで仕事をしていく中で、目的を腹落ちさせる取り組みが必要であった。

取り組みの着眼点は、「当事者意識」である。チームの目的をより深く理解するためには、上司から目的を伝えられるよりも、何故1つのチームになったのかを自分たち自身で考えることが有用だと考えた。人から言われたことは、他人のせいにはできるため受け入れ難いこともある。しかし、自分のこととして、自ら考えたことは当事者意識が生まれ、行動にも変わりやすいと考えた。

(1) チーム立ち上げ期の目的共有

チーム立ち上げ期の取り組みとして、インセプションデッキの作成を行った。チームの目的や役割を考え、我々はどうのようなチームになりたいのかを全員で話し合った。我々の目指すチームは、「ナビゲーション開発での吉野家になる」と定めた。吉野家のコンセプトといえば「旨い、安い、早い」ということで知られているが、我々の開発も、デザイナー、ソフトウェア技術者の連携によってうま味、価値を出す(旨い)、連携により無駄なものを排除しコストを抑え(安い)、かつ開発スピードを上げる(早い)という考えから上述の目標とした。また、目標を達成するための、職場風土についてもこの場で議題として挙げ、どのように仕事をしたいかを話し合った。ここでは、チームの成長を第一とすることを全員の共通認識とし、今までの個人で仕事をするスタイルからチームで仕事をするスタイルに変えようということ合意した。

(2) チーム活動期の目的共有

チーム活動期の取り組みとして、スプリント計画を活用した。スプリント計画は、ゴールを正しく捉えているか、出来そうな道筋なのか、なぜやるのか等を確認し、仕事の目的を共有する場である。この場で今回のスプリントでは何を旨みにするのかコストやスピードはどうアプローチするかを考え、チームの目的、方向性を小さな仕事単位で繰り返し感じることができるようにした。

3.2.2. 役割の共有

役割についても「目的の共有」と着眼点は同一である。自ら考え、それを小さい仕事単位で共有することが必要であると考えた。理由は目的の共有でも述べたが、当事者意識を持つことと腹落ちさせることである。そのため、役割の共有には、仕事の入り口の活動としてスプリント計

画、日々の活動としてデイリースクラムの場合やタスクかんばんを活用した。

(1) 仕事の入口での役割共有

スプリント計画時に、仕事の課題、リスク、懸念点、心配点を考え、それぞれの役割で何をしないといけないかの作戦会議を行った。連携によって解決できる箇所を、仕事の入り口でイメージできるようにし、お互いの役割を共有する場とした。

(2) 日々の仕事での役割共有

仕事の入口で、どのような箇所が注意点なのかを共有できているため、日々のデイリースクラムの場合やタスクかんばんを見て、お互いにサポートをでき、ゴールへのアプローチが問題ないことをそれぞれの役割で考え、共有できるようになった。

3.2.3. モチベーション

モチベーションアップのためには、成功体験を持つことが必要だと考えた。自分たちで考え、工夫した行動が成功に繋がれば、喜びに変わり愉しくなる。この着眼点に基づき、2つの活動に取り組んだ。我々の取り組んだ2つの活動は、先行研究の観点からも有効であることが判断できる。IT 業界における仕事満足の実証研究ではモチベーションに影響を及ぼす7つの要因[6]が報告されている。我々の活動は、①自己実現・スキルアップの可能性、②自分への評価、④コミュニケーションの状態、⑤プロジェクト運営の体制に該当する。

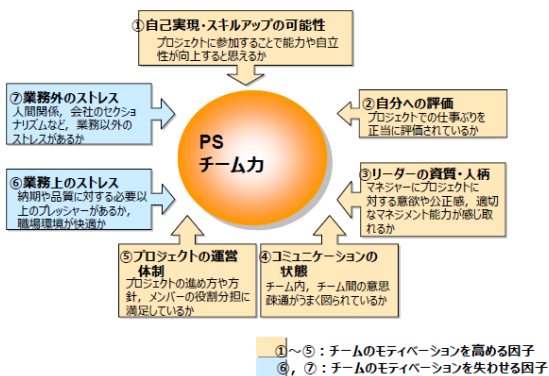


図 2.モチベーションに影響を及ぼす7つの要因

(1) KPT 法を用いた振り返り

振り返りは、スプリント単位で継続的に行う活動であり、

ここに着目した。連携活動を行っていく中で、自分たちの活動が上手くいったと感じ、それを他者にも認められることが重要である。継続的に振り返りを実施することで、自分たちが出来ていなかった部分が改善出来ていることが実感できるようになる。技術的な部分はもちろんであるが、コミュニケーションや、プロジェクト運営についても振り返り、良いことは「良い」と認め合うことで個人の成長とチームの成長を確認できるようにした。

(2) チーム状態の見える化

自分たちの状態を定量的に表す指標を持つことである。チームの状態を定量的に示すことが出来れば、客観的に自分たちを判断することができるようになる。見えることで、改善意識や向上心も生まれてくるのではないかと考えた。指標は、先行研究「IT に現場力は存在するか」で示されている心理尺度を利用した質問紙法を用いたマトリクス[1]を使用した。チームの状態を「仲間意識」、「役割意識」、「規範意識」、「成果意識」、「仕事の難易度」の5つに分類し、定量的に示した。結果については4章で詳細に述べる。

4. 効果検証

4.1. 検証方法

職場におけるチームビルディングの効果を検証する方法として考えられるのは、以下の2つがある。

- (1) 職場において、実施前の生産性や納期と比較する。あるいは、同種の他のチームと比較する。
- (2) チーム状態を何らかの方法で測り、社内外のチームの状態と統計的な比較を行う。ただし、チームの状態を測るためのマトリクスが必要である。

具体的には、図 3 に示す。

- ① 成果比較 A-B:
当該チームの対策前と後でチームの成果、例えばトラブルの件数や納期を比較する。
- ② 成果比較 B-C:
当該チームと同様なチームを探し、その成果を比較する。同じ仕事を行っているチームを探すのが困難であり、非現実的である。
- ③ チームの状態の比較:A-B:
時系列的に計測し比較するが、マトリクスで使用

する同じ質問紙を使用すると、時間的なバイアスが掛かるので、測定が困難である。

- ④ チームの状態の比較:B-C:
同じメトリクスを使えば、比較が可能である。

以降の図、表では、チームビルディングをTBと記載する。

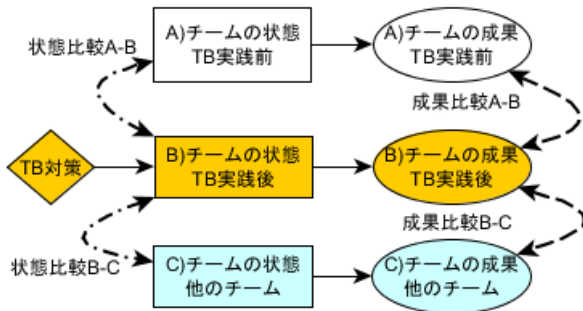


図 3. チームビルディング効果の調べ方

結果として、(1)-①, (2)-④を使って成果、状態の両観点から検証を行うことが出来る。(2)-④の状態比較については、3章でも述べた「IT に現場力は存在するのか」で心理尺度を利用した質問紙法を用いたメトリクスがある[1]。このメトリクスを使って、先行研究で計測されたデータを(a)社外との比較に用い、それ以外の比較として(b)グループ会社との比較、(c)社内(事業部内)の計測を行い、本稿の検証とする。

4.2. チームの成果比較

4.2.1. 比較検証の対象

成果比較では、一般的に不具合件数の増減や、生産性を比較することが多い。しかし、本活動はプロジェクトの初期フェーズ(作り込み時期)での活動であり、不具合件数の比較には適していない。また、他職種の業務を知るために他職種を体験するなど、生産性を目的とした活動ではないため、生産性の比較も適さない。

連携という観点では、デザイナーとソフトウェア技術者がどれだけチームとして活動できているかを検証することが有用であると考える。

デザイナーとソフトウェア技術者は、従来、不明点がある場合は、組織で定められた質問回答手段である QA 票を用いてやり取りをしていた。この QA 件数がどのように推

移したかを確認することで、デザイナーとソフトウェア技術者がチームとして活動できているかが分かる。

また、連携したチームが自立的な改善ができているかは、KPT 法であがった「Try」が「Keep」へ移行できているかを検証することで判断できる。この2点で検証を行った。

4.2.2. 比較結果

連携前はプロジェクト立ち上げ時であれば 100 件近く QA が発生し、回答を得るまでに2~3日を要していた。これが連携後は、QA での確認ではなく即座に認識合わせをするようになり、待ち時間が発生することはなくなった。QA 件数だけでは判断が断定すること難しいが、両者が他職種の業務領域を理解できるようになったことも要因の1つとして挙げられる。

また、KPT 法であがった「Try」が「Keep」へ移行している件数については、連携直後では数は多くなかったが、連携を進めるうちに改善件数が増えてきていることが確認できた。最初は何を改善すべきかが分かっていなかったチームが活動を続ける中で、改善点を見つけ、改善に向けて活動を行っていくことができるようになったと判断することができる。

4.3. チームの状態比較

4.3.1. 比較検証の対象

現場力の比較は、(a)社外、(b)グループ会社内、(c)社内(事業部内)の3つと本稿での活動を行ったチームとの比較を行った。

社外については、「IT に現場力は存在するのか」の実証研究にて、デバッグ工学研究所の松尾谷徹氏が行った、特定非営利活動法人日本プロジェクトマネジメント協会が主催する PM シンポジウム 2013 に集まったプロジェクトマネージャーを対象として行った調査結果を借用した。このデータは、様々なプロジェクトのデータであり、日本の IT 業界の平均的な値と想定できる。

グループ会社内は、デンソー技術研修のハイタレント研修を受講する中堅リーダーを対象とした。ハイタレント研修を受講する人材は各部門のキーマンであり、グループ会社の平均的な値よりは高いと想定できる。

社内は、チームビルディングを実施したチームの他 5 チームに対し計測を行った。

4.3.2. 比較結果

職種を超えた連携とチームビルディングを始めて 6 ヶ月経過したタイミングで、現場力の計測を行ったもの(d)と、前述の(a)(b)(c)とを比較した。

(1) 現場カメトリクスの構成要素による比較

現場力のメトリクスは、質問項目を5つの構成要素に分け、それらについて主成分分析を行い、正規化した得点を算出している。この手法に従い、次の手順で比較を行った。

- i. 比較する質問紙の回答群を(d)の回答に対して、構成要素別に主成分分析を行い、正規化した得点を求めた。
- ii. 構成要素別に、分布と平均値が異なっているか、箱ひげ図によって比較した。

統計処理は、統計ソフト R を用い行い、その結果を以下の図 4, 5, 6 に示す。

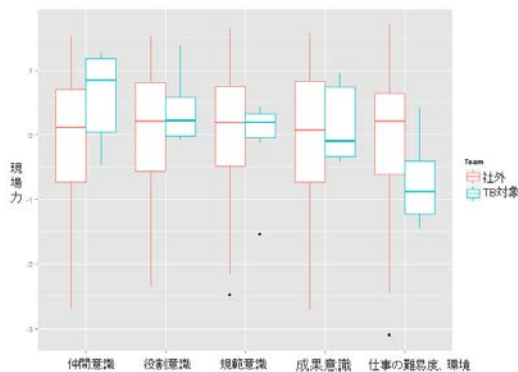


図 4.社外との比較

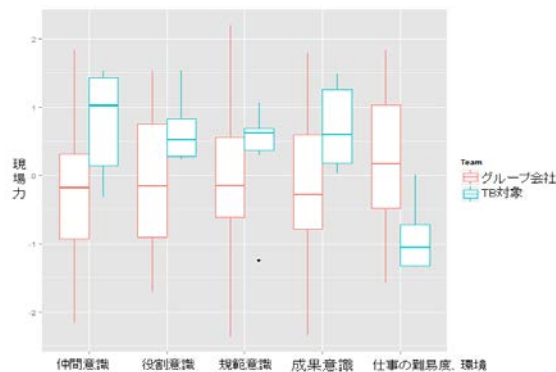


図 5.グループ会社内との比較

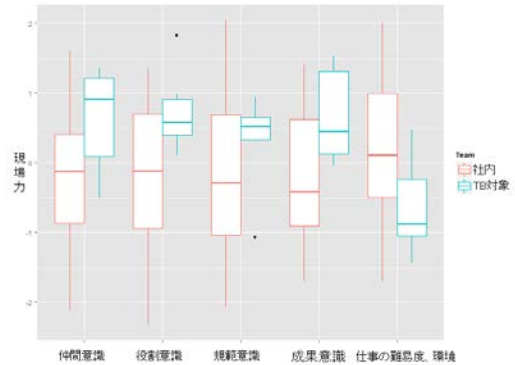


図 6. 社内(事業部内)との比較

現場カメトリクスの構成要素の値の「高い/低い」が、能力の「高い/低い」を表すとは限らない。統計的な主成分分析によって抽出された、違い(分散)を説明するための値である。検証の目的も、「高い/低い」の比較では無く、平均値や分散に差があることを明らかにするためである。

比較結果として、構成要素の「仲間意識」「仕事の難易度意識」については、同じような差が観測された。「役割意識」「規範意識」「成果意識」については、社外との比較では差が認められない結果となった。

(2) 線形判別分析

構成要素別に比較する方法では、多くの構成要素で差があることが明らかになったが、構成要素間の違いや、その違いの差が、全体の差にどのように影響しているのか明らかではない。そこで、現場カメトリクスの項目すべてを使って、判別分析の1つである線形判別分析による比較を行った。判別分析とは、統計的な手法の一つで、与えられたデータ群が、その構成要素である群に分けることができるかを判別する。線形判別分析は、その判別式に線形式を用いたもので、それぞれの観測値に対して判別得点と呼ばれる値を計算する。

比較する対象(a)(b)(c)それぞれに(d)を加え、判別は(d)か、それ以外か、で行った。その結果を表と、箱ひげ図で順に示す。

表 1.全データの線形判別結果

実測グループ	予測グループ			合計
	結果	TB 対象	その他	
	度数	TB 対象 5	1	101
%	TB 対象	83.3	16.7	100
	その他	1.0	99.0	100

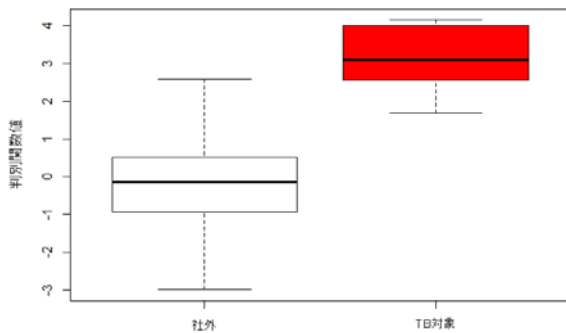


図 7. 全データでの線形判別結果

表 3.社内の線形判別結果

実測グループ	予測グループ			合計
	結果	TB 対象	その他	
	度数	TB 対象 6	0	20
%	TB 対象	100	0	100
	その他	0	100	100

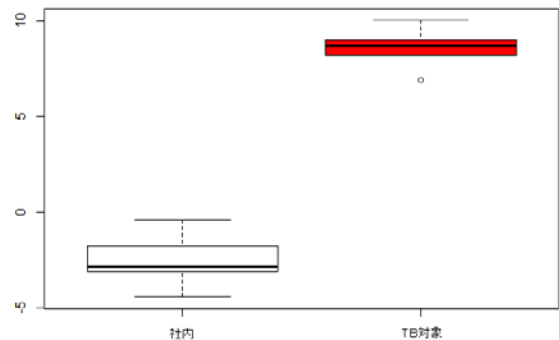


図 9. 社内の線形判別結果

表 2.グループ会社内の線形判別結果

実測グループ	予測グループ			合計
	結果	TB 対象	その他	
	度数	TB 対象 6	0	46
%	TB 対象	100	0	100
	その他	0	100	100

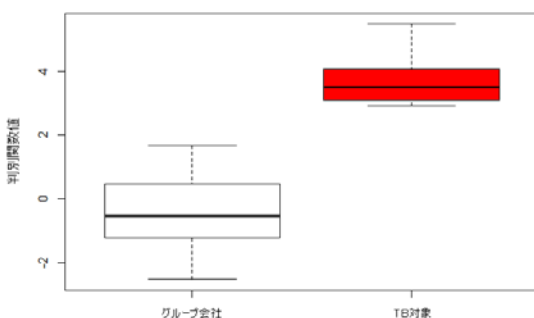


図 8.グループ会社内の線形判別結果

表の判別結果は、実測された2つのグループ、「TB 対象」と「その他」に対し質問紙の回答値からグループの区別を計算で予測した結果とを比較している。表1は、実測された TB 対象が 6、その他が 101 であり、これに対し、質問紙の回答から計算した判別では、TB 対象で1件、その他でも1件、判別が誤ることを示している。正解率は、TB で 83.3% その他で 99%であり、明らかに TB 対象とその他を区別することができる。図7から図9は、判別得点の分布を箱ひげ図で示した。この箱ひげ図から、いずれの場合も、TB 対象とその他では、中央値と分散が異なっていることが明らかである。

(3) 開発現場での定性的成果

チームの状態が異なっていることを(1)(2)で示した。どのように異なっているのかについて、定性的ではあるが調べた。分析結果を裏付けるように、チーム内の振り返りでは以下の声が上がっている。

○最良の選択が採れるようになった。

・デザイナーが楽な方法、ソフトウェア技術者が楽な方法を主張せず、お互いの観点で最適解を検討す

るようになった。

- ・目的とゴールを明確にし、共有することで問題を個人任せでは無く、チームとして扱えるようになった。
- モチベーションが上がる。
 - ・3週間(スプリント)単位で達成感が感じられる。
 - ・定期的な振り返りにより改善意識が高まる。
 - ・技術の幅が広がり、自身の成長を感じられる。(他人にも認められる)
- 問題点の打上げ、補正が早い
 - ・スクラム会議にて、こまめに情報共有が可能となり、問題に対する対応が早くなった。一人で悩まなくなった。
- 開発効率が上がった
 - ・何が本当に必要なかを考えるようになり、不要な成果物作成、無駄な会議をしなくなった。

5. 考察

検証結果より、デザイナー、ソフトウェア技術者の連携を、人、技術の両面からのアプローチしたことは有用であったと考えることができる。特に「仲間意識」は高いレベルにあり、個人ではなくチームとして力を合わせ、共に1つの目標を目指すというチームになっている。現場の声からも、目標に向かって本当に必要なことが何なのかを自主的に考え、チームでコミュニケーションを図ることができるようになったことがわかる。

これはチームとしてのメリットがあるのみでなく、個人にも大きな効果をもたらした。チームのメンバーで互いに支えあうことにより、個人の技術の幅が広がり、成長につながる。それを他人に認めてもらい、チームに貢献していると認識することでモチベーションの向上にもつながった。モチベーションが向上したことによりメンバーの意識が変わり、改善活動のような非生産業務を始め、プロジェクトの様々な活動に挑戦するようになった。その結果、デザイナーとソフトウェア技術者が共に成長し、互いに支えあう領域が拡大していくというプラスのサイクルにすることが出来た。これは、チーム、個人の両方でパフォーマンスの向上に繋がっていくと考える。

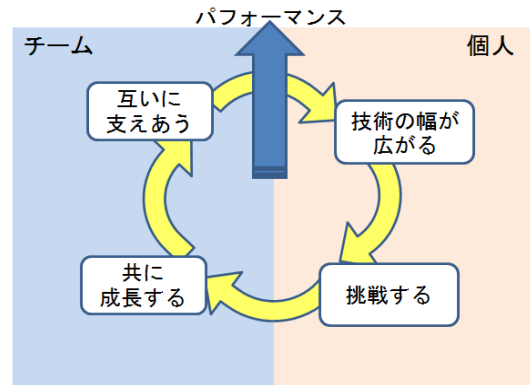


図 10. チームと個人の相互成長サイクル

IT 業界において、組織行動論のような社会科学の知見を適用することは難しいと言われている。その理由は、IT 業界におけるチームの挙動が、一般的な組織(企業)やチームと異なっていることにある[1]。しかし、今回のお互いをよく知らないメンバーが共通のゴールを目指し、自主自律したチームを作るという観点では、社会科学の適用は十分に可能であると判断することができる。

チームビルディングは、成長を続ける「練習の場」(長期的)とすることも重要である[3]。本稿では連携を始めて、6ヶ月のタイミングでの現場力について纏めたが、この状態を維持していくために今後もチームの発展のため活動を行っていく必要があると感じている。そこには今回と同様に、「人間の理論」に着目したアプローチも必要であろう。

6. おわりに

デザイナーとソフトウェア技術者の職種を超えた連携において、技術的なアプローチのみでなく、「人間の理論」に着目したアプローチが十分適用できることが分かった。社会科学の観点からすれば、当然の結果であるが、IT 業界での適用研究はあまり報告されていない。今回の適用事例が同様の取り組みをする方々の参考になれば幸いである。

謝辞

本研究の調査実施、及び論文執筆に当たり、デバッグ工学研究所 松尾谷 徹氏、堀田 文明氏、デンソー電子 PF 開発部 足立久美担当次長、デンソー技研センター 技術研修部 古畑 慶次担当課長、上杉 卓司担当

課長に丁寧かつ熱心な指導を賜りました。また調査にご協力いただいたハイタレント研修受講生の皆様、及び、デンソーテクノ 情報通信 5 部の皆様に感謝いたします。

参考文献

- [1] 松尾谷徹 IT に現場力は存在するのか:その計測と評価の試み,ソフトウェア・シンポジウム 2014.
- [2] 谷古宇浩司 IT 業界の「現場力」はたった 30 点,
<http://www.atmarket.co.jp/news/200607/20/mieru.html>
- [3] 増田礼子 チームビルディングから組織文化へ,SQiP2014.
- [4] 上田泰 組織行動研究の展開,白桃書房,2003
- [5] 関本浩矢 入門組織行動論,中央経済社,2007
- [6] 松尾谷徹 7つの要因を知りモチベーションを管理,
日経 IT プロフェッショナル,2003 年 8 月号
- [7] 橋本圭一, UX とは何ぞや? UX を高める武器を手に入れよう,
<http://www.atmarket.co.jp/ait/articles/1008/31/news096.html>