

## マンモスの牙

## ～ Monetize Metrics Vs. Engineering Metrics ～

松尾谷 徹

有限会社 デバッグ工学研究所

matsuodani@biglobe.jp

増田 礼子

フェリカネットワークス株式会社

Ayako.Masuda@FeliCaNetworks.co.jp

## 要旨

この資料はソフトウェア・シンポジウム 2021 の *FP (Future Presentation)* の資料として作成したものです。提案の主旨は、ソフトウェア工学 (SW 工学) が産業界の「働き方」に及ぼした意図しない副作用 (牙) を示し、その対策について検討することです。

意図しない副作用とは、技術者に対し生産性向上の責務を語る労働強化です。産業界が SW 工学に期待したのは、*QCD* 改善や属人性排除など、*Monetize* (金銭化) 側面での生産性向上であり、SW 工学は、この期待に対して様々な技法や技術を研究し、政策と共にソフトウェア工場やソフトウェア *CAD* を推進してきました。しかし、ソフトウェア関連活動の実態は、雇用されている技術者の属人性に依存しており、本質的な特性は変わりません。掲げられた生産性向上の目標指標 (*Monetize Metrics*) は、工学的な手段ではなく、雇用された人々への労働強化的な副作用を生み、職種としての評価すら下げてしまいました。

ここでは、ソフトウェア産業における雇用者の「働き方」の何が特殊なのか？、SW 工学は「働き方」を守って技法や技術を提案できるのか？など、この問題について考え方や方向性を示し、その評価を行う実証的な *Engineering Metrics* の必要性について提案します。

## 1. はじめに

ソフトウェアに係る技術や産業は、他の産業と比べても飛躍的に拡大し、GAFA [1] に代表される世界最大級

のビジネスに成長しました。この産業を支える一つの分野に、ソフトウェア工学 (SW 工学) があります。SW 工学は、産業の黎明期であった、1980 年代から盛んになり、それまでの研究から実用に向けた多様なアプローチが行われました。

SW 工学を活用する産業界から眺めると、過度な期待を持ち導入を進めた時代が終わると、系統的な技術が未成熟であることに気づき、個々の課題について、技術を適切に選択する活用へと変化しました。新人研修で学ぶ、常識的な用語や知識としての SW 工学と、個々の課題に対応する個別の技術 (たとえば、セキュリティ課題で *Fuzzing* をどのように使うか? など) は、活用側で学習コストが必要です。専門家は、この状況を理解していますが、ビジネス競争下では、根拠の無い生産性 (コスト) 競争が存在し、その結末が、雇用されている技術者への労働強化として作用し「働き方」の状況を悪化させています。

SW 工学は、産業界に弊害を与える意図は全くないのですが、ここで議論する「牙」問題は、工学界の意図と、産業界の意図が本質的に異なることと、ソフトウェア産業における労働の特殊性 (属人性) から生じています。歴史的な経緯を含め、この問題について 2 章で説明します。2 章では、Brooks の著書 "The Mythical ManMonth" [2] (日本語版「人月の神話」[3-6] 俗称「マンモスの神話」) で指摘された、*Monetize Metrics* の代表である「人月指標」を例に説明します。その後、産業界は急激に変化しており、ここ 10 年ほどを見ると Brooks の時代とは全く異なる状況になっており、人月問題とは別の構造で「牙」が生じています。3 章では、この状況における「働き方」の問題について説明します。この状況は、

日本ではまだ一部での変化であり、今後、企業の存亡と共に拡大すると思われます。4章では、サラリーマン型雇用下にある日本の技術者を対象として、ソフトウェア関連の仕事環境がどのように影響しているのかについて論じます。「働き方」の問題は、性能や不具合の数などで計測できる特性とは全く異なり、研究や計測は心理学の知見を利用することになります。紹介するのは、著者らが行った、仕事満足度に関連する質問紙を用いた調査(1,700名)など、いくつかの実証型の研究です。

これらを通して明らかになったことは、「働き方」問題は「人」を対象とする課題であり、課題構造が多様であること、局面で変化することです。構造的には、集団(チームや組織)と職種(技術者)の特性が、一般的な労働者の特性と異なっています。現在の「働き方」に対する評価は、仕事満足度では自己成長が若干強いですが、一般と変わりません。一方、仕事不満足は高く、仕事の成果との相関が報告されています。

## 2. マンモスの歴史

本章では、産業界においてソフトウェアの価値を Monetize の観点から「人月」を用いた歴史的な経緯を説明します。それを支援した SW 工学の考え方は「属人性の排除 = 再現性を高める = 工学」です。ソフトウェア関連活動の付加価値生産性は、現実として属人性依存ですが、属人性要因を切り離しています。SW 工学が「牙」を生んだ直接の原因ではありませんが、現実生じた問題として、属人性に依存する技法や技術を工学的に応用する場合に考慮すべき事例です。

### 2.1. 黎明期

半世紀前、コンピュータ産業の黎明期におけるソフトウェア開発は限られた職場でのみ行われる特別な仕事でした。主に次の3種類の職場です。

1. 汎用機メーカー: OS やコンパイラなどハードウェア販売の付属として提供するソフトウェアの開発
2. 研究・開発: コンピュータを使った研究や商品開発のためのソフトウェア開発
3. コンピュータ導入組織: コンピュータ導入による組織内事務作業の省力化のためのソフトウェア開発

この中で、ソフトウェア開発の仕事が急拡大したのは3番目の、新たにコンピュータを導入する組織(企業や公共団体)でした。業務の内容も、前の2者とは異なり、組織固有のさまざまな伝票に基づく事務処理の電算化です。コンピュータ導入時の一時的な作業なので、人的資源を外部から調達し、プロジェクト(一時的な活動)で進める方法が使われました。その活動内容は主に次の2種類が中心で、技術面はコンピュータを受注したメーカーが行いました。

- データ入力、コンピュータ操作 など
- 伝票や帳票処理のプログラム作成 など

組織外にデータを持ち出せない事情から、プロジェクトの要員は派遣の形態となり、その取引に「人月」が自然に使われました。日本における初期のソフトウェア企業は、この種のビジネスから始まっています。監査法人や会計事務所など事務処理の専門集団から始まった海外の事務所系ソフトウェア企業の多くとは違った流れです。

この時代における産業分類では、サービス業に分類されており、その後、電算機センターと呼ばれたコンピュータの共同利用ビジネス業界と統合され、1984年にJISA(情報サービス産業協会)[7]が生まれました。それ以降の歴史はJISAの資料[8]に詳しく書かれています。

### 2.2. ビジネス環境の変化

黎明期における派遣型のビジネスは、絶対的な供給量不足から、当時の一般的な派遣単金と比べても高額を維持しており、ビジネスは繁栄しました。その後、大型の新規導入が減少し、中小組織への導入になると、コスト競争が始まりました。それでも、コンピュータメーカーのパーク維持競争から、数年ごとに機種を変え、マイグレーションを行うことから、市場規模は維持されました。

事務処理系のプロジェクト購入主であるユーザ組織では、プロジェクトの投資効果が低いことから、人月単金の引き下げ、工数削減、納期確保など、生産性向上の要求が強くなり、結果、価格競争が始まります。供給側は、人件費の安い海外での開発など為替差益的な手法で対応したため、さらに人月単金は低下しました。

行政上の変化としては、1986年に施行された労働者派遣法が、1996年、1999年に改定されました。これにより、プロジェクトでの指揮的マネジメント機能が弱体化し、プロジェクトで生じる問題の対応策が遅れ、さら

に生産性が低下しました。政策としては、情報処理技術者試験によるソフトウェア技術者を認定し、人的資源のスキル価値を区別して市場に供給する方策を続けていますが、資格と生産性の実証が無く、市場価格への影響は強くありません。

### 2.3. 牙の形成

ビジネスの世界においては、ソフトウェアに限らず生産性が評価されます。一人一人の評価ではなく、組織や業界としての生産性について考えます。一般的な生産性は次の2要素から構成されます。

1. 付加価値生産性 = 付加価値額 ÷ 労働量
2. 量的生産性 = 生産量 ÷ 労働量

黎明期における生産性は、量理的生産性で表して、顧客からは評価されました。この時の生産量はプログラムの量や作成した帳票の数などで定義でき、労働量は人月です。定型的な活動、入力と出力が明確な活動であれば、この方法が使えます。他の業界でも、たとえば、タイル張りであれば、時間あたり幾らの面積などの定義が業界で決まっています。

付加価値生産性は、付加価値額を測らないと分かりません。パッケージビジネスなら、パッケージの売価と販売数で測ることができます。プロジェクトの場合は、プロジェクトの売価になります。現実の問題として、黎明期のプロジェクト売価構造は「プロジェクト原価 - コンピュータメーカーの補助 + 利益」であり、顧客は、運用期間中のハードウェアレンタル料金をコンピュータメーカーに支払い、コンピュータメーカーは補助を回収する仕組みでした。

初期の携帯電話ビジネスのように、初期費用は安く見せて、月額で回収するビジネスモデルです。顧客にとってもこのモデルは価値があり、電算機導入後の事務処理コストを合理化した後で、レンタル経費を経理処理できます。

このビジネスモデルが崩壊したのは、汎用機のレンタルビジネスが破綻し、コンピュータメーカーの補助が無くなり、プロジェクト売価が上昇したにも関わらず、プロジェクトの中身は変わらない事態になったためです。このギャップも従事する技術者の労働強化となって作用し、「牙」が形成されていきました。

### 2.4. 人的資源の特性

SW 工学においても、属人的な特性を計測し、データを用いた実証型の研究は存在します。代表的な研究として、Boehm が 1981 年にまとめた著書「Software engineering economics」[9] では、属人的な生産性の差をチーム単位で計測し、それをモデル化した「COCOMO (COnstructive COst MOdel)」を示しました。このモデルは、一般的な線形回帰式 (1) で表すことができます。

$$Y = b_b X + A_E \quad (1)$$

過去の統計データから、見積もりや改善を行う方法は統計的品質管理の手法が使えろと考え、産業界は大きな期待を持ちました。しかし、結果は残念な結末となりました。式の目的変数  $Y$  は、 $\log$  (人月)、説明変数  $X$  は  $\log$  (規模) です。式は線形ですが「人月」と「規模」の関係は対数正規分布であり、実軸に変換するとべき分布に分類されることから、一般的な統計手法が役立ちません。

対数に変換する前のリアルな値では「人月」と「規模」の関係は弱く、一般的な統計手法でモデルを作ることができません。その後、国内で行われた大規模な実証研究においても、同様の結果が観測されています [10,11]。一般的に調査範囲を広げ、データ量を増やせば、真実に近づきモデルの精度は上がるとされていますが、対象の特性が「対数正規分布」の場合、そのような特性は見られません。「対数正規分布」は、対数の世界では統計モデルが成立しますが、リアル軸に戻した場合には、産業界が求める精度には達しませんでした。

この研究は、結論として「ソフトウェア開発は再現性が低い」こと、「規模が大きくなると生産性が低下すること」などを明らかにしています。市場取引にとっては不都合な事実であり、ビジネス分野で積極的に使われることはありませんでした。

### 3. プロダクトビジネスの衰退

GAFA に代表される現代のビジネスは、サービスとそのインフラの提供です。その基盤はインターネット関連のインフラ提供であり、インターネットとスマートフォンやタブレットの普及とともに、世界中で無償で使えるサービスの提供を行い、覇権を制した上でさまざまな有償サービスを展開しています。

20 世紀末に生じた、プロダクトの権利問題や訴訟問題は、インターネット上で稼働するプロダクトの著作権を主張することが困難であることを示しました。Linux [12] や Android [13] などインターネットを利用するプロダクトは、独自の著作権主張するのが難しくなっています。結果的に、2010 年頃から、一斉にオープンソースへの移行が行われました。

プロダクトを納入して利益を得るビジネスと、サービスによって利益を得るビジネスの違いは何でしょうか。その仕事に従事する技術者の働き方と共に考えます。

### 3.1. 実質的な生産性の向上

ソフトウェアの特性として、流用の容易性、流用コストが小さい特性があります。流用の効果は、ソフトウェアに限らず、広く使われているナレッジであり、確かめられた先行事例の活用や蓄積が製品開発の基本的なプロセスとされています。

プログラミングを例にすると、ライブラリを活用する / しなないによる生産性の差は大きいです。たとえば、WEB スクレイピング (WEB サイトからデータを取得し、情報の抽出・加工を行う) を行う場合において、任意のプログラム言語で、すべてを開発することは可能です。別の方法としては、Python の Beautiful Soup [14] など専用のライブラリを使って開発することができます。

例に挙げたライブラリ Beautiful Soup は OSS なので使用権に対するコストはゼロであり、圧倒的に後者の実質生産性が高くなります。実質とは、目的達成に必要なコストで生産性を評価した場合で、人月指標だと、逆に前者が高くなります。

現実的な事例としては、2020 年 2 月末に始まった東京都 新型コロナウイルス感染症対策サイト [15,16] があります。この事例では、約 60 ほどの地域で Fork (ソースコードを複製して派生すること) され、サイトが開発されています。このような規模での再利用は、プロプライエタリ・ソフトウェアの世界 (人月の世界と重複) では考えられません。

SI ビジネスにおける企業規模が大きくなれば、人月指標の世界でも再利用による実質生産性が向上すると考えられます。しかし、個々の開発案件は顧客との契約があり、開発企業がすべての著作権を持っていないため、流用は限られ、流用による生産性向上は進んでいません。

顧客側にとっても、自社のシステムのソースコード以外を流用することはできず、購入したパッケージも使用権に制約があるため、実質生産性の向上にはつながりません。

まとめると、流用はソフトウェア生産性向上の基本的な手段ですが、プロプライエタリ・ソフトウェアの場合、または、量的生産性評価の場合においては、阻害され、結果として付加価値生産性が向上しません。

### 3.2. ワークフロー標準化

オープンソースの慣習による開発は、開発の進め方やテストなどワークフローに関する情報の公開につながりました。その結果、実務に従事する技術者が体験した効率的な開発環境やワークフローが、デファクトスタンダードとして広がり、かつ改善が続いています。例を次に示します。

- 他の OSS の利用と依存関係  
OSS が使用権、利用権を開放していることから「使えるものは作らない」が徹底しており、外部の機能を多用しています。その状況は GitHub [17] であれば Dependency Graph で表示されます。先に紹介した東京都 新型コロナウイルス感染症対策サイトの例でも、利用している外部機能は 1,200 件を超えています。OSS のプロダクトやサービスは氷山のようなもので、海面下では当該リポジトリには含まれない他の OSS に支えられ、その成果の上でさらに機能を開花させています。
- マイクロサービスの概念導入  
マイクロサービスとは、機能ごとに独立したサービスに分割し、相互に影響しない作り方です。この設計概念により、同時に複数の機能について変更を加えても、相互影響は小さくなります。特別な実装方法を使わなくても、多くの機能が既存の OSS を利用する呼び出しや接続であることが貢献しています。
- 自動テストなどの自動化  
機能の独立性が高いことから、1 行のコード修正であっても当該機能の統合テストでデグレードを確認する習慣が定着しています。それを支える自動テストも機構間で独立である特性から、統合は

システムレベルですがテスト内容は機能テストで済みます。

また、膨大な依存関係から生じる依存先のバージョン問題については、Bot による自動検出を利用するなど、最新の技術が共有されています。

#### ● 内部ドキュメントの廃止

従来の品質概念の中には内部ドキュメントの重要性があります。受発注の納入物にも含まれ、膨大なドキュメントにリソースが投入されていました。これらのドキュメントは、OSS 開発では全く見当たりませんが、大勢の技術者が外部から参加し、短期間で成果を上げています。マネジメントと同様、無い方が効率的だという実証結果です。

これには、さまざまな原因が考えられます。たとえば、コードは公開され、必ず Review を受けるので、非効率でトリッキーな処理は排除されるなど作成者の技術習熟度が高くなっており、ドキュメント自動生成ツールなどから得る他との関係情報に重点が置かれています。

### 3.3. 技術者のモチベーション

商用に利用されている OSS 開発においても、雇用関係にないメンバが多数参加しています。このような環境下で、技術者はどのようにチームを作り協働を行っているのか、どのようなモチベーションを持っているのかについて、次の 3 つの現象から考えます。

#### 1. 働き方

雇用や契約による労働提供下では観測されない現象として、活動の分布が指数分布かべき分布であること [18,19]、また、活動時間帯が広い範囲に及んでいる [18] ということが挙げられます。これは、自由意志での参加であり、その背景に社会貢献としての善行の意識があると考えられます。雇用されている一部のメンバも、この意識の影響を受け、行動変容が生じているように推察されます。

#### 2. 協働と役割

3.2 節において組織内対立の課題を述べましたが、OSS 開発の中では「やらされ感」がなく、仕事に対する「オーナー感」が生じていると考えます。図 1 は、東京都 新型コロナウイルス感染症対策サイ

トにおける Issue / Pull Request を提起 (Create) あるいは Review したメンバ (2 件以下の貢献を除く) の重複をベン図で示したものです。図 1 から、「オーナー感の範囲」も広く、役割の枠 (提起者 / Reviewer および Issue / Pull Request) を超えて協働が生じていることが分かります。

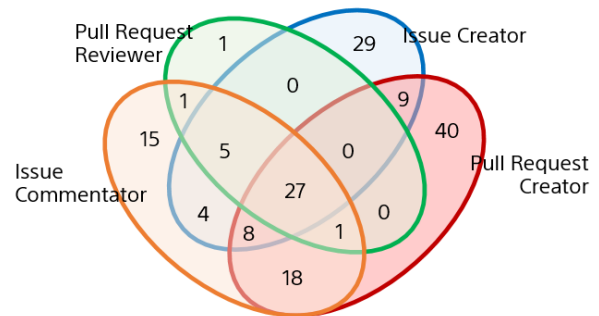


図 1: Create / Review における役割重複を示すベン図

#### 3. チームや仲間意識

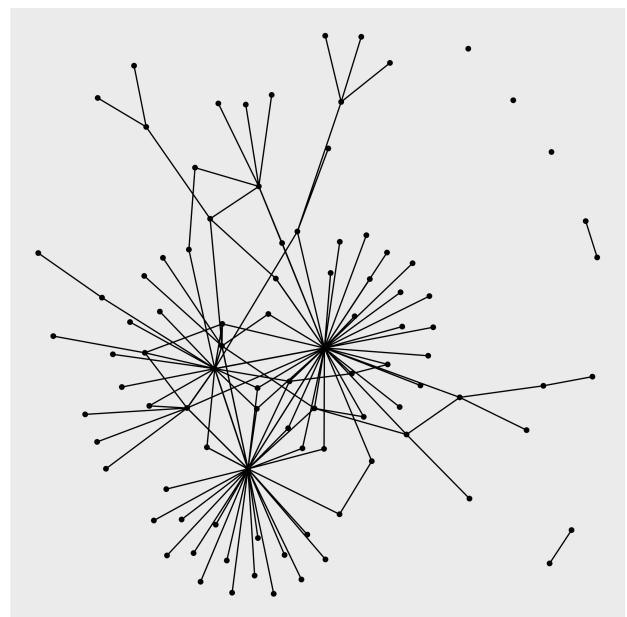


図 2: Pull Request と Review におけるチーム形成

連携活動のほぼすべてがサイバー空間で行われ、個々の活動はローカル環境での個人作業です。この環境下において、仲間意識や小集団としてのチー

ムが生まれるのかについて分析を行いました。図 2 は、東京都 新型コロナウイルス感染症対策サイトにおいて、開始後 1 ヶ月間の Pull Request と Review の協働活動の一部をネットワーク図で表したものです。図 2 のネットワーク分析においてクラスタと呼ばれる集団が確認できることから、チーム形成が行われていることが分かります。

従来型の仕事の進め方、あるいは技術者の働き方は Monetize の影響を強く受けています。この影響が強く同一化した価値観として「牙」が出現し、さまざまな弊害を生んでいます。比較分析として、直球勝負な Monetize を絶ち、繁栄した OSS を調べることで、今まで気づけなかった問題が見えてきたと感じています。

### 3.4. 新たな「牙」

サービス・ビジネスの世界では「人月」問題とは異なる新たな「牙」が生じています。それは「成果主義」と「育成問題」です。

#### 1. 成果主義

チームや組織に対する評価が明らかになりフィードバックされるようになると、その構成メンバーである技術者個人の成果が識別されます。成果物に対するメンバーの貢献分布は、多くの場合「べき分布」となり、少数の「Head」と多数の「Tail」に分かれています。べき分布が現実の中で、多様性や平等性をチームの規範として取り入れるかという問題があります。

#### 2. 育成問題

技術者としてスキルを磨き成長するには、経験を積む必要があります。しかし、経験が無いと経験を積むポジションに就くことができないジレンマがあります。GitHub や一部の企業では、インターンシップを拡大して運用し、成長支援を行っています。

### 4. 働き方の測り方

属人的な特性を持つ活動の価値、すなわち生産性をどのように捉えるのでしょうか。ビジネスの世界における評価の流れは次の通りです、

- 供給不足の時代：頭数を揃える「人月」
- 量的生産性の時代：伝票数や規模の量と「人月」
- 終着段階：付加価値など、さまざまな要素

付加価値評価の終着段階においても、ビジネス観点からの評価は技術者の「働き方」にとって、多少ルールは異なりますが、労働強化として作用することになります。そのルール下の競争により、よりビジネスに貢献できる技術者を選別する仕組みです。強者にとっては、良い仕組みかもしれません。

「働き方」問題は、この競争に自ら志願して参加する強者の働き方だけではなく、サラリーマン型雇用下の技術者も対象とします。結論的に、ビジネスルールでの計測指標では「働き方」の良否に強いバイアスがかかり、この計測だけでは正しく表現できない、と言えます。

働いている技術者から見た「働き方」の状態を測る必要があります。この種の計測は、物理的な計測方法ではなく、人の行動を引き起こす意識を測り、統計処理を行って特性を明らかにする方法です。SW 工学においても GUI や要求の評価に用いられています [20]。

心理学では心理尺度と呼ばれ、その手法が確立しています。心理尺度の作成には質問紙法と呼ばれる手法が用いられています [21, 22]。質問紙法は、計測対象を構造的にとらえ、下位尺度と呼ばれる要素に分解し、下位尺度ごとに質問項目を作成して計測を行います。プロジェクトマネジメントにおける先行研究では、プロジェクト・メンバーのモチベーションを「職務満足感」「仕事意欲」「精神健康・ストレス反応」の 3 つの下位尺度に分け質問紙を作成し、そこから主成分を抽出して尺度化を行っています [23–25]。

ここでは、ソフトウェア分野における「働き方」について、実際に計測した 2 つの先行研究 [25, 26] について、主に計測方法と分析方法について示します。

#### 4.1. パートナー満足度の計測と分析

この研究は、従業員満足度のプロジェクト版であるパートナー満足 (PS: Partner Satisfaction) 調査として 2000 年に始まりました。質問紙による調査であり、サーベイの結果から、試作した質問紙を使って内部で評価を行い、その後、実際のプロジェクトで予備調査と分析を行いました。

最初の予備調査は、調査票の評価を兼ね、調査時期は2001年10月で、回収された件数は約200件でした。本調査は、情報サービス産業3社を主体とするプロジェクトに従事する技術者を対象とし、2002年1月から3ヶ月間で行い、調査件数は回収ベースで約1,700件でした。

予備調査と本調査から、データの主成分分析を行った結果、目的変数を総合的な満足度とし、7つの説明変数を抽出しました。7つの説明変数は、次の構成です。

- リーダーシップ：プロジェクト（チーム）リーダーのリーダーシップ行動の影響
- プロジェクトの運営方針：プロジェクト運営の方針、方針に関する説明、運営状況の影響
- コミュニケーション：プロジェクト間、プロジェクト内チーム間、プロジェクト内のコミュニケーションの影響
- スキルアップ：プロジェクトの業務によるパートナーの市場価値の向上、仕事の挑戦性などの影響
- 業務上の問題：業務上、調整や対応が必要となる問題（不満足）の影響
- 業務外の問題：業務上の直接の障害ではないが、プロジェクト内の力関係や雰囲気等、業務に影響を与える問題（不満足）の影響
- 評価：プロジェクトにおける評価の公正性、給与への反映などの影響
- 総合的な満足感：プロジェクト及び自身の業務への全般的な満足感

この分析により、因子の尺度化を行い、プロジェクト間の差、職種による差、年齢層による差があることが明らかになりました。ここで得られた尺度は、その後のPS調査の基礎データとして使われています。

#### 4.2. 現場力とプロジェクト成果：計測と評価の試み

この調査は、2つの異なったプロジェクト群（調査事例1：20プロジェクト、調査事例2：55プロジェクト）に対して、現場力の要素とプロジェクト成否の関係を質問紙法を用いて計測し、その影響について判別分析により評価を行っています。相対的な評価ですが、2事例に

おけるプロジェクト成否の判別分析では、95%（20プロジェクト：4段階評価）と82%（55プロジェクト：5段階評価）の高い判別率を得ることができています。

計測に用いた質問紙の構成は次の通りです。

表1: 調査事例1:下位尺度の質問数と因子数

下位尺度	質問数	因子数
顧客との関係性	7	2
仲間意識	5	1
役割意識	4	2
規範意識	4	2
合計	20	7

表2: 調査事例2:下位尺度の質問数と因子数

下位尺度	質問数	因子数
仲間意識	4	1
役割意識	7	2
規範意識	5	2
成果意識	4	2
環境意識	4	1
合計	24	8

質問紙を使った調査と分析について2つの先行研究を示しました。この種の調査方法は、主観的な評価に対して一般的に用いられており、対象が存在する「働き方」の計測としては有望と思われます。「働き方」の構成因子は多様なため、質問紙の質問項目で抜けると計測から漏れてしまう点や、同じ質問項目を繰り返して使うとバイアスが生じるなど、課題もある。

## 5. おわりに

ここでは、ソフトウェアに係る技術者の「働き方」をテーマとしました。産業界の求める生産性向上の期待をそのままSW工学の課題として捉えると、想定外の副作用として「働き方」に影響が及ぶことを示しました。想定外を想定するには「働き方」の主人公である技術者に対して、客観的な計測：Engineering Metricsとして質問紙など心理調査の方法を取り入れ普及することを提案します。

## 参考文献

- [1] Wikipedia, 「ビッグ・テック」,  
<https://ja.wikipedia.org/wiki/ビッグ・テック> (accessed:2021/05/14)
- [2] Brooks, Frederick P, "The Mythical Man-Month : essays on software engineering", Addison Wesley, 1975
- [3] フレデリック・フィリップス・ブルックス, 山内正弥 (訳), 「ソフトウェア開発の神話」, 企画センター, 1977
- [4] フレデリック・P・ブルックス Jr., 滝沢徹 (訳), 牧野祐子 (訳), 宮澤昇 (訳), 「人月の神話 狼人間を撃つ銀の弾はない」(原著発行 20 周年記念増訂版), 星雲社, アジソン・ウェスレイ・パブリッシャーズ・ジャパン, 1996
- [5] フレデリック・P・ブルックス Jr., 滝沢徹 (訳), 牧野祐子 (訳), 宮澤昇 (訳), 「人月の神話 狼人間を撃つ銀の弾はない」(原著発行 20 周年記念増訂版, 新装版), ピアソン・エデュケーション, 2002
- [6] フレデリック・P・ブルックス Jr., 滝沢徹 (訳), 牧野祐子 (訳), 宮澤昇 (訳), 「『人月の神話 狼人間を撃つ銀の弾はない(新装版)』, 丸善出版, 2014
- [7] 一般社団法人情報サービス産業協会,  
「JISA とは」,  
[https://www.jisa.or.jp/about\\_jisa/admission/admission/tabid/923/Default.aspx](https://www.jisa.or.jp/about_jisa/admission/admission/tabid/923/Default.aspx)  
(accessed:2021/03/20)
- [8] 一般社団法人情報サービス産業協会,  
「情報サービス産業とは」,  
<https://www.jisa.or.jp/explain/tabid/754/Default.aspx> (accessed:2021/03/20)
- [9] Barry W Boehm, "Software engineering economics", IEEE transactions on Software Engineering, No.1, pp. 4-21, 1984
- [10] 寺本雅則, 松尾谷徹, 上村松男「ソフトウェア開発過程を定量的に解析して生産性と品質を向上させるソフトウェアメトリクス」, 日経エレクトロニクス 6 月 4 日号, 1984
- [11] 松尾谷徹「開発コストのメトリクスの現状と課題」, 昭和 63 年電気・情報関連学会連合大会予稿集, 1988
- [12] Linux ,  
<https://www.linux.com> (accessed:2021/05/14)
- [13] Android ,  
<https://www.android.com> (accessed:2021/05/14)
- [14] Leonard Richardson, "Beautiful Soup",  
<https://www.crummy.com/software/BeautifulSoup/>  
(accessed:2021/03/20)
- [15] 東京都 新型コロナウイルス感染症対策サイト,  
<https://stopcovid19.metro.tokyo.lg.jp/>  
(accessed:2021/03/21)
- [16] 東京都 新型コロナウイルス感染症対策サイト (GitHub) (Tokyo COVID-19 Task Force website),  
<https://github.com/tokyo-metropolitan-gov/covid19>  
(accessed:2021/03/20)
- [17] GitHub, <https://github.com/>  
(accessed:2021/03/20)
- [18] 松尾谷徹, 増田礼子「コミュニティ型チーム活動の進化 ~ COVID-19 対策サイト開発から学ぶ ~ 」, ソフトウェア・シンポジウム 2020 論文集, pp. 117-123, 2020
- [19] 増田礼子, 森本千佳子, 松尾谷徹, 津田和彦, 「大規模オープンソース・ソフトウェアプロジェクトにおける開発効率の計測」, 電気学会論文誌 C (電子・情報・システム部門誌), Vol.138, No.8, pp.1011-1019, 一般社団法人 電気学会, 2018
- [20] 中尾仁, 津田和彦, 中谷多哉子「ERP 導入の為にカスタマージャーニーマップを用いた顧客満足分析」, 電子情報通信学会技術研究報告・KBSE, 知能ソフトウェア工学, Vol.112, No.314, pp.31-36, 一般社団法人電子情報通信学会, 2012
- [21] 小塩真司, 西口利文「質問紙調査の手順」, ナカニシヤ出版, 2007



- [22] 吉田富二雄 (編) , 「人間と社会のつながりをとらえる「対人関係・価値観」: 心理測定尺度集 2」, サイエンス社, 2001
- [23] 松尾谷徹, 「パートナー満足によるソフトウェア生産性の向上」, 第 20 回ソフトウェア生産における品質管理シンポジウム, 日本科学技術連盟, 2001
- [24] 松尾谷徹, 「パートナー満足 (PS) と人的リソースのパフォーマンス」, プロジェクトマネジメント学会誌, Vol.4, No.1, pp.3-8, 一般社団法人プロジェクトマネジメント学会, 2002
- [25] 榎田由紀子, 松尾谷徹, 「Happiness & Active チームを構築する実践的アプローチ: チームビルディングスキルの開発 (i 特集i コミュニケーション・マネジメント)」, プロジェクトマネジメント学会誌, Vol.7, No.1, pp.15-20, 一般社団法人プロジェクトマネジメント学会, 2005
- [26] 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 in 秋田 論文集, pp.1-8, ソフトウェア技術者協会, 2014