

情報システム開発におけるソフトウェア技術者の思考の特徴 に関する考察

Cognitive features of software engineers in information system development

谷川 由紀子^{1,4}, 鈴木 栄幸², 加藤 浩³, 福住 伸一¹, 原田 悦子⁴
Yukiko Tanikawa, Hideyuki Suzuki, Hiroshi Kato, Shin'ichi Fukuzumi, Etsuko Harada

¹ NEC 情報・ナレッジ研究所, ² 茨城大学, ³ 放送大学, ⁴ 筑波大学
NEC Corporation, Ibaraki University, The Open University of Japan, University of Tsukuba
y-tanikawa@cw.jp.nec.com

Abstract

We investigate cognitive features of software engineers who design and develop information systems. The observation experiment and the interview for software engineers were executed, and were analyzed focusing on "How to understand the target system". It was revealed that software engineers are hardly aware of use scenes of systems, unlike usability specialists. It was also found that they recognize work not for being done by users to achieve a task but for using functions. Through analyzing interaction protocols in interviews, it was suggested that work procedure of software engineers and their process for developing skills by participation in development projects influence these cognitive features.

Keywords Information systems, usability, Human centered design, software engineers, thinking, how to catch an object and a problem

1. 背景と問題

近年, 社会全体の情報化が進み, 一般の利用者が情報システムを使う機会が増加した上に, スマート端末の普及によって誰もが日常生活で「直観的な操作」の便利さを体感するようになった。そのため, 仕事で使う情報システム(業務システム)にも, 厳しく使いやすさが求められるようになっている。しかしながら, 情報システムの開発現場において使いやすさ向上への取り組みはあまり浸透していないという実態がある。その原因を, E.Metzer^[1], T.Memmel^[2] は, 情報システムの開発プロセスと使いやすさ向上のプロセスとの違いにあると指摘したのに対して, A.Seffah^[3-4], X.Ferre^[5] は, プロセスを実践する人々のコミュニティの違いにあると捉えている。すなわち, 情

報システムの開発を担うソフトウェア技術者と使いやすさ向上を担うユーザビリティ専門家は, 共有する言葉, ものの見方, 規範, 道具等の異なるコミュニティに属しており, それが開発現場への使いやすさ向上活動浸透の障害になっていると指摘している。

筆者らは, 使いやすさ向上のためのソフトウェア技術者支援の研究に取り組む中で, Metzer^[1] の見地にたち, 既存のシステム開発プロセスに使いやすさ向上のプロセスを統合する試みを行ってきた^[6]。しかし, プロセスを統合しても, 使いやすさ向上の活動にはソフトウェア技術者にとって難しい点が多いことが, 開発した技術の現場試行を通じて明らかになっている^[7]。試行結果の分析から, その原因には, Seffah^[3-4], Ferre^[5] が指摘したように, ソフトウェア技術者のコミュニティに特有の, もの(対象・課題)の見方・捉え方やアプローチの仕方(従う規範や活用する道具)が関係するのではないか, と考えた。

ここで, ソフトウェア技術者とはものを「作る側」の人であり, ユーザビリティ専門家とは, 利用者の視点でものの使いやすさを考える人, すなわち, ものを「使う側」の視点を持つ利用者の代弁者と捉えることができる。海保^[8] は, インタフェースをよりよくするために, 「使う側」と共に, 「作る側」の認知特性を知ることの重要性を指摘している。しかし, A.Seffah^[3-4], X.Ferre^[5] らの研究においても, 両者の違いについての言及はわずかしかない。

そこで, 本稿では, ものを「作る側」のソフト

ウェア技術者のものの見方や捉え方の特徴とその要因について、「使う側」の視点を持つユーザビリティ専門家との比較をもとに分析し、考察する。

2. 方法

2.1 実験

ソフトウェア技術者 10 名(実務経験 10 年以下の若手 7 名とベテラン 3 名)およびユーザビリティ専門家 3 名を対象として、使いやすさ向上に不可欠な「利用者視点での考え方」を代表するタスク実行を課題とする実験を行い、その過程を観察した。

実験課題は、情報システムを使って実施する業務とそのシステムの利用者との関係の把握を目的とする一連のタスクで、以下の3つとした(図1)。

- (a) 構築または改修するシステムの主な対象業務を指定
- (b) 対象業務別に、その達成に必要な作業フローと各作業の担当者(役割)を抽出
- (c) システム利用者を定義して、各利用者が担う役割を記述

実験参加者には、上記タスク用に作成した MS-Excel ベースのシートを電子的に提供し、現在担当中か過去に担当したシステムについて分析・記入してもらった。実験後に、タスク実行に関して、半構造化インタビューを行った。さらに、実験から半年後に、実験参加者が分析対象としたシステムについて、上記をもとに実験者がまとめ

た内容の確認を目的とする半構造化インタビュー(フォローアップインタビュー)を行った。実験の様子は、ビデオデータと PC 画面の動画キャプチャとして記録した。

2.2 インタビュー

ソフトウェア技術者 10 名(実験参加者と共通)およびユーザビリティ専門家 6 名(3 名は実験参加者と共通)に対して、参加者自身の仕事の経験や仕事に関わる環境等について、半構造化インタビューを行った。インタビューは、全てビデオデータとして記録した。

3. 結果

実験参加者 13 名が実験課題で分析したシステムとその対象業務の概要について、機能構成と業務フローを併せた「概要図」にまとめた。図2に「概要図」の具体例を示す。その際、参加者による分析結果(シートへの記入内容(a)主な対象業務、(b)業務毎の作業フローと作業毎の役割、(c)利用者と各利用者の担う役割)と、参加者の実験時の発話、インタビュー時の対象システムについて説明した発話をもとにした。

さらに実験参加者が認識している機能、対象業務、業務毎の作業フロー、各作業の担当者(役割)、そして利用者との関係づけを直感的に把握できるように、対象システムを機能、作業フロー、現場での実践の3つの観点から捉えて、それぞれの繋がりを記述する「3層モデル」を考案した。先の

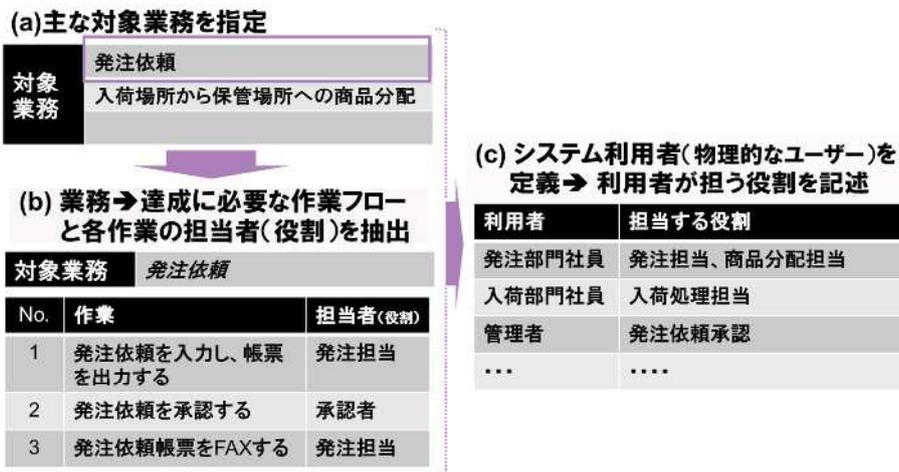


図1 実験課題

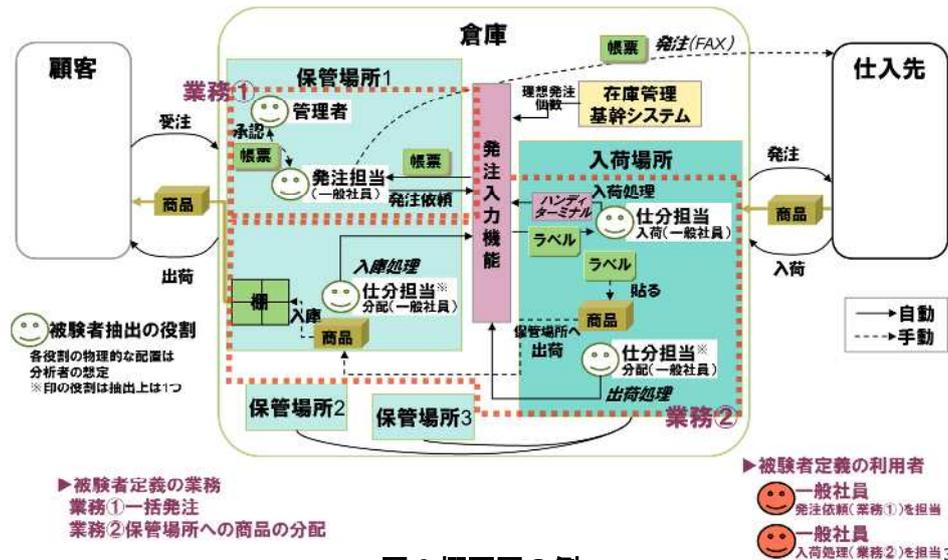


図2 概要図の例

概要図をもとに、分析対象システムの(a)(b)(c)を、ユーザビリティ専門家2名が協力して分析し、3層モデルに対応付けて書き直した。これは、実験参加者からの提供情報をもとにした「ユーザビリティ専門家の対象システムの捉え方」と考えることができる。図3に具体例を示す。次に、この図に参加者が記入した(a)(b)(c)をマッピングした。これは、「実験参加者の普段の対象システムの捉え方」と考えることができる。図4に具体例を示す。

これら2種類の3層モデル図を比較検討するための1つの指標として、両者の一致率を算出し、「実験参加者とユーザビリティ専門家のシステムに対する認識の一致率」として定義した。具体的な考え方と算出方法を、図3と図4の3層モデル図に示した事例をもとに示す。

3層モデル図には、各レイヤー内に要素(例：図3の作業フローレイヤーにおける[発注依頼&帳票出力])と要素間の関連がある。要素間の関連には、順序関係(例：図3の作業フローレイヤーにおける[発注依頼&帳票出力]と[発注依頼承認])と、包含関係(例：図3の作業フローレイヤーにおける[発注依頼&帳票出力]と[発注依頼業務])の2種類がある。また、隣接するレイヤーにある要素間にも関連がある(例：図3の作業フローレイヤーにおける[発注依頼承認]と現場での実践レイヤーにおける[管理者]、作業とその実行者の関係)。これらの要素および要素間の関連の有無に着目して、図3に示す「ユーザビリティ専門家の対象システムの捉え方」と図4に示す「実験参加者の普段の対象システムの捉え方」を比較すると、表

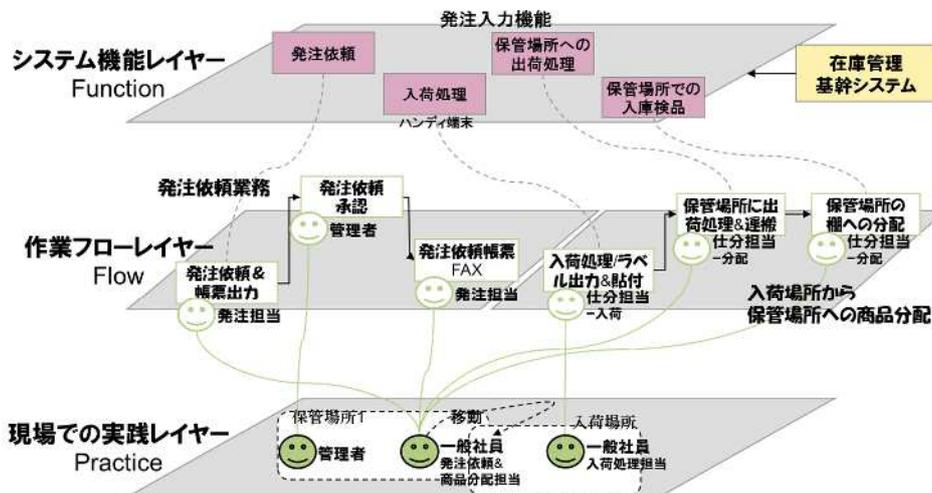


図3 3層モデル図：ユーザビリティ専門家の対象システムの捉え方の例

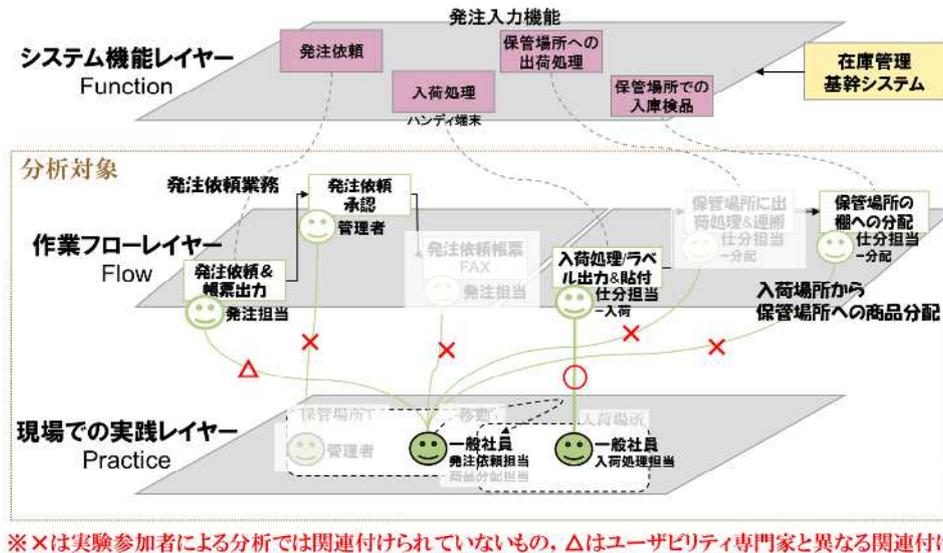


図4 3層モデル図：実験参加者の普段の対象システムの捉え方の例

1, 表2に示すようなパターンに整理できる.

表1 要素に関する比較パターン

		ユーザビリティ専門家の捉え方	
		有	無
実験参加者の捉え方	有	A	C
	無	B	-

表2 要素間の関連に関する比較パターン

		ユーザビリティ専門家の捉え方	
		有	無
実験参加者の捉え方	有	A'	C'
	無	B'	-

各パターンの意味を, 図3, 図4にそった具体例と併せて次に示す.

- パターンAとパターンA'

Aの具体例として, 作業フローレイヤーの要素[発注依頼&帳票出力]がある. これは, 「ユーザビリティ専門家の対象システムの捉え方」と「実験参加者の普段の対象システムの捉え方」が一致するものと考えられる.
- パターンBとパターンB'

B'の具体例として, 作業フローレイヤーの要素[発注依頼承認]と現場での実践レイヤーの要素[管理者]との関連がある. これは, 「ユーザビリティ専門家の対象システムの捉え方」に対する「実験参加者の普段の対象システムの捉え方」の漏れと考えられる.

- パターンCとパターンC'

Cの具体例として, 現場での実践レイヤーの要素[一般社員: 発注依頼担当]がある. これは, 「ユーザビリティ専門家の対象システムの捉え方」に対する「実験参加者の普段の対象システムの捉え方」の違いと考えられる.

表1のA~C, 表2のA'~C'の各パターンに合致する個数をカウントし, ユーザビリティ専門家の捉え方と実験参加者の捉え方が一致したパターン(表1のAと表2のA')の個数を全パターンの合計個数で割って, その比率を算出した. これを「実験参加者とユーザビリティ専門家のシステムに対する認識の一致率」と定義した. 具体的な計算式を以下に示す.

$$\text{認識の一致率} = \frac{A + A'}{A + B + C + A' + B' + C'}$$

この計算式を用いて, 実験参加者のうち, フォローアップインタビューにおいて, 実験者がまとめたシステム機能および対象業務内容の確認を終了した9名(ソフトウェア技術者7名とユーザビリティ専門家2名)の分析事例について, 「実験参加者とユーザビリティ専門家のシステムに対する認識の一致率」を算出した. 算出は, 現場での実践レイヤー, 作業フローレイヤー, そして現場での実践レイヤーと作業フローレイヤーとの関連, の3つの視点から行った. 算出した「認識の一致

率」を表3に一覧にして示す。

表3 実験参加者とユーザビリティ専門家のシステムに対する認識の一致率

参加者	レイヤー内		レイヤー間
	現場での実践 Practice Layer	作業フロー Flow Layer	現場L&作業L Practice & Flow
1	0.21	0.17	0.15
2	0.29	0.50	0.32
3	0.10	0.61	0.13
4	0.25	0.24	0.18
5	0.62	0.42	0.35
6	1.00	1.00	1.00
7	0.92	0.82	0.73
8	0.75	0.53	0.59
9	0.67	0.49	0.55

4. 対象システムの捉え方に関する考察

「対象システムの捉え方」に着目して、インタビュー参加者16名を対象に発話プロトコル分析を行った。さらに、実験参加者毎の2種類の3層モデル図、すなわち「実験参加者の普段の対象システムの捉え方」を表す3層モデル図と「ユーザビリティ専門家の対象システムの捉え方」を表す3層モデル図との相違点の特徴に着目して分析を行った。これらに、表3に示した「実験参加者とユーザビリティ専門家のシステムに対する認識の一致率」を併せて、参加者間で比較し考察した。

その結果、システムの捉え方には、大きく分けて次の2パターンがあることが明らかになった。

- 機能を起点とした捉え方
- 利用者が行う作業を中心にした捉え方

(1) 機能を起点とした捉え方

機能を起点とした捉え方は、情報システムを開発する際に機能から考える、また機能を最も優先する捉え方である。この捉え方をする人は、表4、5に示す実験参加者の発話例にみられるように、3層モデルにおける機能レイヤーを起点として、作業フローレイヤー、現場での実践レイヤー、の順に考えていく。また、表6、7の発話例にみられるように、「現場での実践」レイヤーをあまり意識していない。この「現場での実践レイヤー」に対

する意識の低さは、発話から機能起点の捉え方をしていることが伺える参加者と、表3に示した現場での実践レイヤーに対する「認識の一致率」が0.3以下の低い値を示した参加者が概ね一致することから、機能を起点した捉え方の1つの特徴と考えることができる。

表4 3層モデルの確認順に関する発話例

A: えっとですね、作業、システムフローレイヤーから見ました。なんでかっていうと最初に目があったのは作業フローレイヤーでした。ですが、ちょっと難しくてここから追いかけるのが、また機能から言って、0123って私が言ったと思うんですけど、頭の中で実際の処理の動きごとに持っていったほうが理解がしやすいなと思ってここから見ている。そうすると比較的作業フローで結びつけるので。

表5 3層のどこから考えるかに関する発話例1

W: だいたい、あの、これだけじゃなくて、他にも色々担当されたと思うんですけど、
B: はい
W: 上から見ていくって感じですか？
B: そうですね。機能があって、そこから作っていく感じですね。

表6 日常の仕事で意識することに関する発話例

C: あくまでどっちかっていうと、なんていうですかね。人とか、どういう風に向こうが動いているからこういう機能使ってるとか、全然意識してなくて。基本はそのシステム間のデータのやり取りぐらいしかやっぱこっちは意識してないんで。

表7 現場での実践レイヤーに関する発話例1

D: うん。そうですね。たぶん知っというたほうが、より良いテストができる気はしますが、あまり意識してなかったですね。それでもできますね。

さらに、機能起点の捉え方をする人は、実際のシステム利用者（例：図3の現場での実践レイヤーの[管理者]）を考える場合に、「人」というより「機能の利用権限を付与する対象」として捉えている可能性が高い。例えば、参加者による分析例

の中に、3層モデル図において、一般社員（現場での実践レイヤー）に、必要なスキルレベルが異なるとされる発注作業と発注計画作業（機能フローレイヤー）を一律に紐づけているのに対して、管理者（現場での実践レイヤー）には承認作業（機能フローレイヤー）のみを紐づけているものがあった。これは、「発注機能」の利用権限は一般社員に与えるが「承認機能」の利用権限は管理者のみに与える、という「機能利用の権限」の視点で考えると説明しやすい。ソフトウェア技術者と協業するユーザビリティ専門家も、表 8, 9 に示すように、本考察を支持する発言をしている。

表 8 利用者の捉え方に関する発話例 1

R: 利用者ってあんまり実は考えてないのかなと思います。利用者を考えるときって、権限としか捉えてないんです。これは、この人が使えない画面という考えも権限分けになっちゃうと思います。

表 9 利用者の捉え方に関する発話例 2

S: 普段はここ（機能レイヤーを指しながら）に人がいるんですね。ソフトウェア技術者の方の設計って。ここ（機能）に人がついてるんですよ。これを使ってなんかするというよりも、この機能を使う人はこの人っていうか、設計をする感じなので。

M: 権限っていうやつですね。

(2) 利用者が行う作業を中心にした捉え方

利用者が行う作業を中心にした捉え方は、現場でどのような利用者が、どのような作業を、どのように実施しているかを中心に考える、また優先する捉え方である。この捉え方をする人は、表 10, 11 に示す実験参加者の発話例にみられるように、3層モデルにおける作業フローレイヤー、現場での実践レイヤーを中心と考えて、それをもとに機能レイヤーを考える。従って、表 12 の発話例にもみられるように、「現場での実践」レイヤーを明確に意識している。このことは、発話から利用者/作業中心の捉え方をしていることが伺える参加者と、表 3 に示した現場での実践レイヤーに対する「認識の一致率」が 0.7 近くの高い値を示した

参加者が概ね一致することから、利用者/作業中心の捉え方の 1 つの特徴と考えることができる。

表 10 3層のどこから考えるかに関する発話例 2

W: 今ご自身がお仕事するときって、どういう風に全体のシステムを考えていくものですか。

E: えっと、私は作業フローレイヤーですね。まず作業の流れをやって、それを誰が使うのかっていうのをやっていて、それを決める感じですね。

表 11 3層のどこから考えるかに関する発話例 3

F: ここ（現場での実践レイヤーを指しながら）からみて一斉配信の仕方（機能）がおかしかったんで、やっぱりそれ（機能）は直してもらったっていうことがありましたね。

表 12 現場での実践レイヤーに関する発話例 2

G: 少なくとも実際に（顧客と）接触できないケースもあるんですけども、その場合でもここでキチンとこんな人が利用するんです、どんな人が利用するんですか、どんな人がいますかっていうのを聞くようにしてますね。

(3) 作業フローレイヤーの捉え方の違い

上記にあげた違いに加えて、機能を起点にした捉え方と利用者が行う作業を中心にした捉え方では、作業フローレイヤーの捉え方にも違いがある可能性が見えてきた。作業フローレイヤーは、機能起点の捉え方をする人も、表 13 の発話に示すように把握の必要性を認識している。しかし、その把握対象に対する考え方や見方は、利用者/作業中心の捉え方をする人とは異なる。

表 13 システム開発に必要な層に関する発話例

H: 人にはよりますが、真ん中のところ（作業フローレイヤー）までわかっていけば作れると思います。

作業フローにおける把握対象は、システムを利用して実施する「業務」と業務を構成する作業とその手順、すなわち「作業フロー」である。まず、「作業フロー」について、利用者/作業中心の捉え方をする人は、業務達成のための作業フロー（図 3 の発注依頼業務の例では、発注を完了するためのフロー）と考える。これに対して、機能起点の

捉え方をする人は、機能を使う作業フロー（図3のシステム機能レイヤーにある[発注依頼]機能を使うフロー）と考える。従って、業務達成に必要でも、機能を使わない作業は把握の対象に入らない。実際に、図4（実験参加者の捉え方）をみると、機能を使わずに手作業で行う[発注依頼帳票FAX]は、作業フローから漏れていることがわかる。また、表14に示した実験参加者の発話、また表15,16に示したユーザビリティ専門家の発話からも、この捉え方の違いを伺うことができる。

次に、「業務」について、利用者/作業中心の捉え方をする人は、業務を、現場における実践の視点から定義するのに対して、機能起点での捉え方をする人は、機能に対応するように定義する。

例えば、営業部員向けの顧客情報提供システムを対象にした実験参加者による分析例において、このシステムのメインターゲットとなる「営業部員」がシステムを利用して実施する主要業務（すなわち提供情報を活用した営業活動）が、業務として定義されていないケースがあった。分析者（実験参加者）が定義した業務は、このシステムによる各種提供情報の管理（登録/更新）業務であり、情報の種類別に設けられた機能（情報のライフサイクル管理のための機能）と1対1で対応していた。一方、「営業部員」は、各業務において登録・更新される情報の末端の利用者としてのみ位置付けられていた。すなわち、この例の分析者は、対象システムが管理する情報を現場で利用する最終の利用者ではなく、情報そのもののライフサイクル管理を中心に、業務を捉えたと考えられる。

このように、業務に対する見方が異なることから、利用者/作業中心の捉え方をする人と、機能起点での捉え方をする人では、同じシステムを対象としても、定義する業務にズレ/違いが生じることが多い。この点について指摘したユーザビリティ専門家の発話例を、表17に示す。

表14 作業フローの捉え方に関する発話例1

I: まあ、作業フローレイヤーもたぶん、なんていうんですかね。システム部分だけな気がしますけど。その発注案に承認とか。

表15 作業フローの捉え方に関する発話例2

T: ワークフロー図って言うってお客さんに提出する資料ってほしいAシステムとか機能とか、ABCとかってこうなって、こう行きます、承認ですってこう行きますって。で、例えばここからこの先がFAXだったりすると、なんかこうピッて書いてあって出力とかって書かれてるだけで、最後のデータはここに返ってきてこういう系の画面ですとかこんな感じなんですね。ここに、ここでやってほしいことってこの間に人がいるよねとかそういうのがあってほしいんですけど、こっってバサッといくんですよ。自分たちがつくるものじゃないので。

M: あなたが作るものの、フロー。

表16 作業フローの捉え方に関する発話例3

U: あの、なんていうか、この情報更新しなきゃ、とか緊急時に配信されるんですよみたいなところは、意外に、レベルが何個あるんですよとか、情シスさんとかも詳しいですし、ソフトウェア技術者さんも詳しいような気がします。で、すごくこの話に集中したりとかして、作業フローレイヤーを全然考えてないって訳ではないと思うんですけど、このなんていうかこう対応のパターンがいろいろありますよねとかそういうことは見てない気がします。

M: ふーん。面白いね。

U: なんか、フロント側っていうのはあんまり見ていない。バックヤード側は結構みてる、みたいな感じですね。

表17 作業フローの捉え方に関する発話例4

W: バックヤードの機能とかなり密接度の高い作業フローに関しては見ている。

U: 見ている。あの、バックステージ側というか。ただ、それが、ここ（現場での実践レイヤーを指しながら）と関係あるので、あっているかはちょっと微妙。

上記のような作業フローレイヤーにおける漏れや捉え方の違い/ズレは、表3の現場での実践レイヤーに対する「認識の一致率」が低い参加者（すなわち前記(1)の考察からは機能起点の捉え方をしていると考えられる人）に加えて、「認識の一致率」が比較的高い参加者（前記(2)の考察からは利

用者/作業中心の捉え方をしていると考えられる人)の分析例(3層モデル図)においても確認されている。これは、現場での実践レイヤーが把握できていても、作業フローを考える際に機能起点の捉え方が分析者(実験参加者)の中で優先されるケースがあることを示唆する。今回の実験参加者の中にも、この漏れや違い/ズレのために、作業フローレイヤー、また現場レイヤーと作業レイヤー間の「認識の一致率」が現場での実践レイヤーのものと比較してかなり低くなった人が複数いた。ものを「作る側」の人にとっては、利用者/作業中心の捉え方よりも、機能起点での捉え方が優位に働く傾向が改めて示唆された。

5. 捉え方への影響要因に関する考察

6章で考察したような捉え方・思考の特徴に影響を与える要因を考察するために、インタビューにおける発話プロトコル分析を行い、要因の候補となるものを抽出した。具体的には、技術者としての育成過程における「開発システムの利用現場との接点の多さ」とソフトウェア開発における「仕事の進め方」である。システム利用現場との接点の多さが捉え方に影響する可能性を示唆する若手ソフトウェア技術者による発話例とユーザビリティ専門家による発話例を、各々表18,19に示す。

表18 現場を知る機会に関する発話例

- J: で、やっぱりその、あんまどういった人がどういう動きしてるかというよりも、こういう業種だとかこういう用語が出て、こんなシステムだよっていうとこぐらいですね。社内の共有。実際その私も現場行くのって、別にこういう聞きに行く機会とかもなくて。たまたまサーバーセットアップしに行き行って来いって言われて行った時に、まあちょっと見て、感じたり。あとはその次期の話あるんで、現場見学来ませんかって言われて、見たりした時に感じるぐらいなんで。あんまりこういうの知ろうって活動はないですね。
- M: まあ技術者として仕事をするなかにそういう事はないんですね。そういう。
- J: あんまりそういうの学ぶ、学べる機会はないかなあって気はしますけどね。

表19 現場を知らない影響に関する発話例

- V: この部分って、「だと思えます。」って人が大半だと思うんですね。
- M: やっぱりFAXだと思えますとかね。
- V: 思いますっていう人なので、やっぱり技術者って自信のないこと関わらないので。あの、あやふやなものを出して、問題になるくらいなら出さないの。

さらに、発話分析をもとに、利用現場との接点の多さをみる観点として、要件定義を主とした上流工程への参画経験、開発したシステムの顧客への導入支援経験、また開発プロジェクトにおけるリーダー経験を設定し、先の表3の「認識の一致率」を算出した参加者について、経験状況を整理した。これらの個人としての経験に加えて、参加者が実務経験を積んできた開発プロジェクトの特性の影響についても併せて考察するために、プロジェクト特性として、規模(参画人数)と開発の種類(新規開発、既存システム改修、パッケージシステム利用開発)も整理した。参加者それぞれの職種と実務経験と併せて、表20に結果をまとめる。

表20と表3をあわせてみると、表3の現場での実践レイヤーに対する「認識の一致率」が0.7近くの高い値を示した参加者(参加者6~9)に共通しているのは、上流工程またはシステム導入支援の経験とリーダー経験があることである。これらの経験は、現場での実践レイヤーに対する「認識の一致率」が低い値の参加者にはほとんどないことから、システム利用現場との接点の多さが、現場での実践レイヤーを明確に意識することへの影響要因となっていることが考察される。一方で、現場での実践レイヤーに加えて、作業レイヤー、また作業レイヤーと現場での実践レイヤーの関連も含めた全ての「認識の一致率」を対象として考えると、現場との接点が多い場合でも値が0.5前後とそれほど高くない参加者もいる。従って、現場との接点の多さは、システムに対して利用者/作業中心の捉え方をする事への影響要因とはいきれない。表3において、全ての「認識の一致

表 20 実験参加者の経験

実験参加者		1	2	3	4	5	6	7	8	9
職種	S:ソフトウェア技術者, U:ユーザビリティ専門家	S	S	S	S	S	U	S	S	U
実務経験	N:若手, E:ベテラン	N	N	N	N	N	E	E	N	N
開発経験	上流(企画:提案~要件定義)									
	導入支援									
	リーダー									
プロジェクト (PJ) 経験	PJ規模	大:100人以上								
		中:数十人								
		小:10人以下								
開発種類	新規									
	既存改修									
	パッケージ利用									

率」が 0.7 以上の高い値を示す参加者に共通するのは、実務経験が豊富なベテランだということである。このことから、実務経験の豊富さが、対象システムの捉え方の 1 要因となる可能性が示唆された。

さらに、表 21 に示すような発話例から、ソフトウェア開発における仕事の進め方、すなわち、対象となるシステムを細分化し、部分に閉じて最良の開発をすることでシステム全体を効率よく開発する進め方が、特に機能起点での捉え方の 1 つの影響要因となっている可能性が考察される。

表 21 開発の進め方に関する発話例

- M: ということは、統合テストができるっていう時には、この作業フローレイヤーは、百パーセント理解してないとやっぱできないですかね？それとも七割程度で。
- K: うーん、そのテストする対象のところさえわかればできるので、全体を全部わか、例えば、ここの部分テストするときに、こっこの作業のフローとかは分かってなくてもできるので。

6. おわりに

情報システムの設計開発を担うソフトウェア技術者に対して、使いやすさ向上に不可欠な「利用者視点での考え方」を代表するタスクを実行してもらった実験とインタビューを行い、ソフトウェア技術者の思考の特徴とその要因について分析した。その結果、特に若手のソフトウェア技術者は、システムを実際に使う利用者、また利

用者がどのようにシステムを使うのか(利用状況)といった「現場での実践」の観点を、ほとんど意識していない、また、利用者を、システムを「使う人」というより「機能の利用権限を付与する対象」と捉えている、さらに、システムを使う作業についても、業務達成のために人が担うとの意識は明確でなく、機能を使うものとの視点が強い、といった共通の特徴を持つことがわかった。さらに、インタビューでの発話内容の質的分析と参加者の捉え方の特徴と経験との関係の分析を通じて、上記のような思考の特徴、すなわちシステムの機能を起点とした捉え方には、ソフトウェア技術者の仕事の進め方と開発プロジェクトを通じた育成過程が影響していることが示唆された。

今後さらに多くの参加者事例を集めて分析することで、ソフトウェア技術者の思考の特徴とその影響要因の詳細化を進める。さらに、それらをもとにして、ソフトウェア技術者が使いやすさ向上に自ら取り組むことができるようになるための育成プログラムや支援方法の策定に取り組んでいく。

参考文献

- [1] E Metzker and M Offergeld. (2001): "An Interdisciplinary Approach for Successfully Integrating Human-Centered Design Methods into Development Processes Practiced by Industrial Software Development Organizations." 8th IFIP International Conference, EHCI 2001: 19 ~ 33

- [2] T Memmel, F Gundelsweiler, H Reiterer. (2007): "Agile human-centered software engineering." BCS-HCI '07 Proceedings of the 21st British HCI Group Annual Conference on People and Computers: HCI...but not as we know it - Volume 1: 167-175
- [3] A Seffah, and E Metzker. (2004): "The obstacles and myths of usability and software engineering." Communications of the ACM 47.12 (2004): 71-76.
- [4] A Seffah, J Gulliksen, M C. Desmarais, eds. (2005): Human-Centered Software Engineering-Integrating Usability in the Software Development Lifecycle. Vol.8. Springer, 2005.
- [5] X Ferre. (2003): "Integration of Usability Techniques into the Software Development Process." Bridging the Gaps Between Software Engineering and Human-Computer Interaction (2003): 28.
- [6] Y Tanikawa, R Okubo, S Fukuzumi. (2012): "Proposal of human-centered design process support environment for system design and development", Proceedings of the 4th Applied Human Factors and Ergonomics (AHFE) International Conference, pp.7825-7834
- [7] Y Tanikawa, et al. (2014) : "Problems in Usability Improvement Activity by Software Engineers - Consideration through verification experiments for human-centered design process support environment", HCI International 2014 Proceedings Vol.12 LNCS8521, pp.641-651, 2014.
- [8] 海保博之 (1992) : “ インタフェースの認知科学的諸問題 ” 認知科学の発展 第5巻 特集「インタフェース」, 日本認知科学会, 講談社, 1992 : pp.1 ~ 4