

## システム更改時に現行システムと新システムとの要求仕様の差分を聞き出す質問の誘導方法の提案

谷藤史門<sup>†</sup> 埜口元<sup>†</sup> 山中隆敏<sup>†,††</sup> 古宮誠一<sup>†</sup>

ソフトウェア開発は顧客の要求をまとめた要求仕様書を基に進められていく。このため、要求分析工程において顧客要求の漏れや誤りが発生した場合、開発されるソフトウェアも漏れや誤りを含んだものとなる。従って、ソフトウェア開発において、要求分析工程は重要な作業である。しかし、ソフトウェアに求められる顧客の要求を漏れなく抽出し、これを正しく要求仕様書に記述することは容易ではない。このため、これらの作業の支援を行う技術の開発が望まれる。著者はソフトウェア要求抽出作業をインタビュー技術であると捉えた。本稿では、顧客が運用している現行システムを更改するときに、SE が顧客の要求を漏れや誤りなく抽出できるように、インタビューによる要求抽出作業を誘導する仕組みを提案することを目的としている。

### Proposal of a method to navigate question that finds out difference of software requirements specification of present system and new system when system is replaced

Shimon Yato<sup>†</sup> Hajime Noguchi<sup>†</sup> Takatoshi Yamanaka<sup>†,††</sup>  
Seiichi Komiya<sup>†</sup>

Software is developed on the basis of requirements specification that described users' requirements. Therefore, when the requirements analysis process includes some errors and omissions of users' requirements, it becomes the one that the developed software contained the leakage and the mistake. Therefore, the requirements analysis process is important work in the software development. However, it is easy not to extract the requirements of the customer requested from software without omission not to describe this in the requirements specification correctly. It is required to develop the technology to support requirements elicitation process. We regard software requirements elicitation technology as an interview technique. This paper has proposed a framework to guide the extraction process by an interview request.

#### 1. はじめに

ソフトウェア開発におけるライフサイクルモデルには、「ウォーターフォールモデル」や「スパイラルモデル」などのように様々なモデルがある。しかし、どの様なライフサイクルモデルを用いた場合にも、顧客の要求をまとめた要求仕様書を基に開発が進められて行く。このため、要求仕様書に顧客の要求の漏れや誤りが含まれている場合、開発されるソフトウェアの機能にも漏れや誤りを含んだものとなる。従って、ソフトウェア開発において、顧客の要求を獲得し要求仕様書にまとめる作業を行う要求分析工程は重要な作業であると言える。しかし、ソフトウェアに求められる顧客の要求を遺漏なく抽出し、抽出された要求を正しく分析し、要求仕様書を記述することは容易な作業ではない。そこで、これらの作業の支援を行うシステムの開発が望まれる。

著者は、この問題を解決するために、ソフトウェア要求抽出作業をインタビュー技術であると捉え、SE が行うインタビューによる要求抽出作業を誘導するシステムの開発を行う。さらに、顧客の要求を獲得するだけでなく、獲得した顧客の要求を基に要求仕様書を自動生成するための仕組みの開発を行う。そして、これら2つのシステムの開発を行うことで、ソフトウェア開発プロセスにおける要求分析工程の作業全てを支援することを目指している。本稿は次のような構成になっている。2章では要求抽出工程の重要性と課題について説明する。3章では本研究の目的を述べる。4章では熟練SEの特徴を述べ、5章では、インタビューにおける話題の誘導方法について述べる。最後にまとめと今後の課題について述べる。

#### 2. 要求抽出工程の重要性と課題

ソフトウェア開発は、顧客が求めているソフトウェアへの要求をまとめた要求仕様書を基に進められる。このため、もし要求仕様書に顧客要求の漏れや誤りがあるまま開発が進められると、出来上がったソフトウェアは顧客の意図に合わないものとなる。その結果、開発工程に手戻りが発生し、ソフトウェア開発は致命的なダメージを受ける。このため、ソフトウェア開発工程における要求抽出作業は、大変重要な工程であることがわかる。

しかし実際には2つの理由により要求抽出作業を困難になっている。一つ目は、顧客側に存在する理由(原因)である。顧客は、ソフトウェア開発技術について詳しくないため、SEにどのような情報を与えれば、ソフトウェアを作ってもらえるかわから

<sup>†</sup> 芝浦工業大学院  
Graduate School of Engineering, Shibaura Institute of Technology

<sup>††</sup> 富士通株式会社  
FUJITSU LIMITED

ない。つまり、SEの質問に対して何を答えてよいか分からなくなり顧客の求める要求をSEに伝えることが出来なくなる。2つ目は、経験の浅いSE(初心者SE)側に存在する理由(原因)である。初心者SEは、顧客の業務内容について詳しくないために、初心者SEは顧客要求の抽出漏れや矛盾点があっても判らなくなってしまう結果として仕様書に要求の漏れが発生する。

### 3. 本研究の目的

2章で示したように、顧客と初心者SEの間に共通な知識が存在しないため、互いによくコミュニケーションが取れず、要求を抽出する際に顧客要求の漏れや誤りが発生してしまう。その結果、要求仕様書にも顧客要求の漏れや誤りが発生しソフトウェア開発に多大なダメージを与えてしまう。そこで、本研究ではこの問題に対して、ソフトウェアに対する顧客要求を抽出する技術をインタビュー技術であると捉え、初心者SEのインタビューによる要求抽出作業を誘導することで解決を試みる。本稿では、特に、システム更改時に現行システムと新システムとの要求の差分を聞き出す質問の誘導について述べていく。

### 4. 熟練SEの特徴

初心者SEと実務経験の豊富なSE(熟練SE)では要求抽出作業においてインタビューの内容や進め方、要求仕様書の記述内容についてどのような相違があるか明らかにするために、文献[1]の実験によって、以下のことが判明している。

#### 4.1 話題の分類カテゴリ

熟練SEが行ったインタビューを分類すると、表1の様にインタビューに取りあげられるべき話題が9つのカテゴリに分類されることが判明している。

表1 話題の分類カテゴリ

What	開発すべきソフトウェアの機能
Why	開発背景・理由
Example	機能についての具体例
Current system	現行システムの質疑
Budget	開発予算
schedule	開発期間
Constraints	他システムとの関連
Polices	開発方針
Conditions	各種インタフェース

#### 4.2 話題の遷移パターン

前節で示した熟練SEのインタビューにおける話題のカテゴリには次のような明確な3つパターンが見出すことが出来た。

- (1)パターン1(疎遠な関係の顧客の場合)  
 {What | Example | Why | Current System}  
 > {Constraints, Policies, Conditions}  
 > {Budget | Schedule}
- (2)パターン2(標準)  
 {What | Example | Why | Current System}  
 > {Budget | Schedule}  
 > {Constraints, Policies, Conditions}
- (3)パターン3(非常に親しい顧客の場合)  
 {Budget | Schedule}  
 > {What | Example | Why | Current System}  
 > {Constraints, Policies, Conditions}

{A, B, C}は話題をA, B, Cの順序で採り上げるが、1つのカテゴリの話題が出尽くしてからでない次のカテゴリの話題には移れない。このような話題の遷移方法をシリアルタイプと呼ぶ。

{A | B | C}は1つのカテゴリの話題が終わる前に他のカテゴリの話題に移ることが出来る。このような話題の遷移方法をパラレルタイプと呼ぶ。

本研究では、この話題の遷移パターンを用いて初心者SEの誘導を行う。話題の遷移パターンを用いた要求抽出作業の誘導方法は次の章で述べる。

また、採用される遷移パターンは、上記で示した顧客との親密度の相違によって異なる3つのパターンがある。パターン1の疎遠な関係にある顧客(例えば、これまで、顧客のシステム開発を請け負ったことがなく、これから新たにビジネスの関係を構築しようとしている顧客)に対しては、最後に開発予算や開発期間に関する話題を採り上げる。このため、開発予算や開発期間に合わせるためにソフトウェアの機能や作り方に関する話題に戻る必要が生じる。パターン2のどちらとも言えない間柄にある顧客(パターン1またはパターン3のどちらにも当てはまらない顧客)の場合には、作成するソフトウェアに求められる機能を聞き出した直後に開発予算や開発期間に関する話題を採り上げ、開発予算や開発期間に合うようにソフトウェアの作り方を調節する必要がある。

パターン3の非常に親しい顧客(例えば、何度もシステム開発を請け負ったことがある顧客で、ビジネスパートナーの関係を継続している顧客)に対しては、最初に開発

予算や開発期間に関する話題を採り上げ、それに合うようにソフトウェアの機能や作り方を調節する。

## 5. インタビュー誘導方法

本研究では、初心者SEが熟練SEと同じように要求抽出できるようにするために、熟練SEが採用している話題の遷移パターンに沿ってインタビューできるように支援(誘導)する。しかし、話題のカテゴリの遷移パターンを熟練SEと同じにするだけではインタビュー作業を誘導することはできない。何故なら、各カテゴリ内での質問から質問への遷移も同様に誘導する必要があるからである。そのために、話題のカテゴリの遷移パターンが熟練SEと同じになるように誘導する層と、1つのカテゴリの中で質問する作業を誘導する層からなる、図1のような話題と質問の二階層によりインタビューを誘導する。

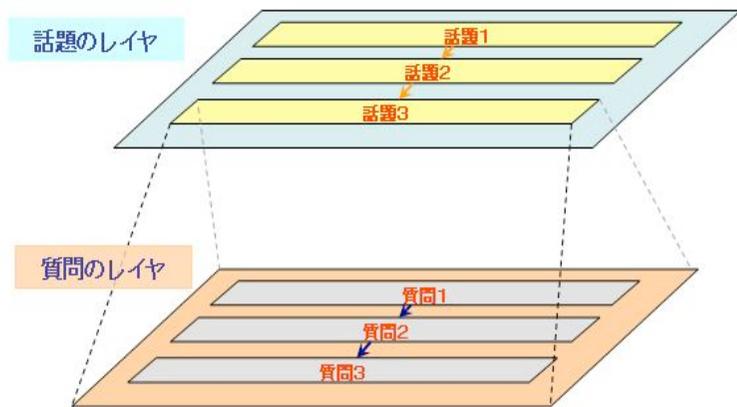


図1 話題と質問の二階層モデル

### 5.1 話題レイヤの誘導方法

話題レイヤでは、話題の遷移パターンをインタビューのシナリオに用いることで、インタビューを誘導する。図2は標準の場合のパターンである。話題レイヤにおいて、話題のカテゴリの遷移パターンが熟練SEと同じになるように誘導するために、進捗管理テーブルを用いる。インタビューによる要求抽出において、シナリオに沿ってインタビューを進めると共に、カテゴリごとにその進捗を管理すれば、シナリ

オに沿ってどこまでインタビューが進んだかを管理することができる。このため、シナリオに対応させた進捗管理テーブルを支援ツールが作成してインタビューによる要求抽出作業の進捗を管理する。進捗管理の例として表2に示す。

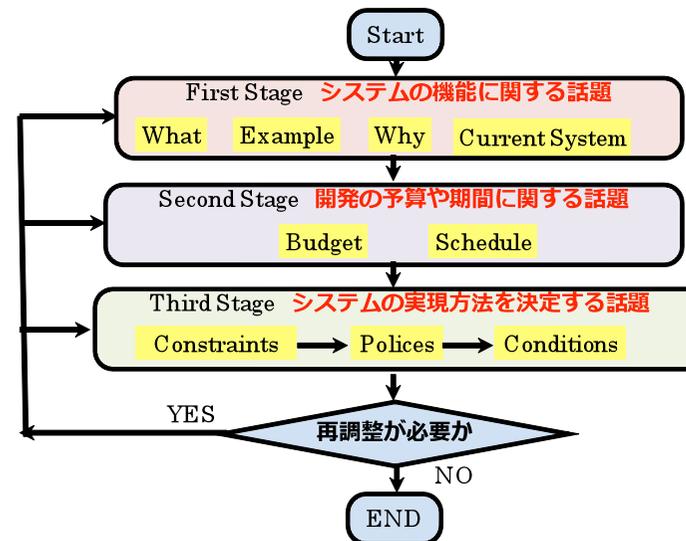


図2 話題の遷移パターン(顧客との関係が標準場合)

表2 進捗管理テーブル

カテゴリ	順序	タイプ	進捗
What	1st	Parallel	2
Examples	1st	Parallel	2
Why	1st	Parallel	2
Current system	1st	Parallel	2
Budget	2nd	Parallel	1
Schedule	2nd	Parallel	1
Constraints	3rd	Serial	0
Policies	3rd	Serial	0
Conditions	3rd	Serial	0

このテーブルの進捗の欄は以下のことを意味している。

- 0 であれば、その話題は未だ採り上げていない。
- 1 であれば、まさにその話題を採り上げている最中である。
- 2 であれば、その話題については終了した。
- 3 であれば、そもそもその話題は採り上げる必要は無い。例えば、新規導入の場合には、現行システムについての質問(current system)は採り上げる必要はない。

尚、各カテゴリにおける進捗状況は SE (SE が複数の人数で対応していればそのリーダー) が判断し、進捗管理テーブルに情報をセットする。従って、表 1 のような進捗管理テーブルを用いれば、要求抽出作業の進捗を管理することが出来る。

### 5.2 各ステージでのインタビューの誘導方法

話題の遷移パターンは、熟練 SE が行うインタビューにおける話題の進め方をパターン化したものであり、3つのステージから成り立っている。ここでは、1st ステージでのインタビューを誘導する方法について明らかにする。

#### 5.2.1 1st ステージでの質問を誘導する方法

要求を効率よく抽出するために、まずは What で開発しようとするシステムの機能についてインタビューすることから始める。このとき、顧客の提示するシステムの機能がどのようなものかわからなかった場合は、顧客に対して具体例を提示してもらうために Examples (システム機能の具体例) に関する質問を行う。顧客から具体的なシステム機能の例を提示してもらうことによって、顧客の提示する機能が理解できる。次に、顧客の提示するシステムの機能が何故必要なのか、その機能が必要な理由について理解するために、Why でシステム機能の開発背景やシステム機能の目的・理由について質問し理解する。次に、現在稼働しているシステムが存在する場合には、Current system で現在稼働しているシステムについての質問を行う。これから新規開発するシステム機能に対してどのようなアプローチをとれば良いのかを決定する非常に重要な情報となるからである。図 3 に 1st ステージでの誘導の流れを示す。

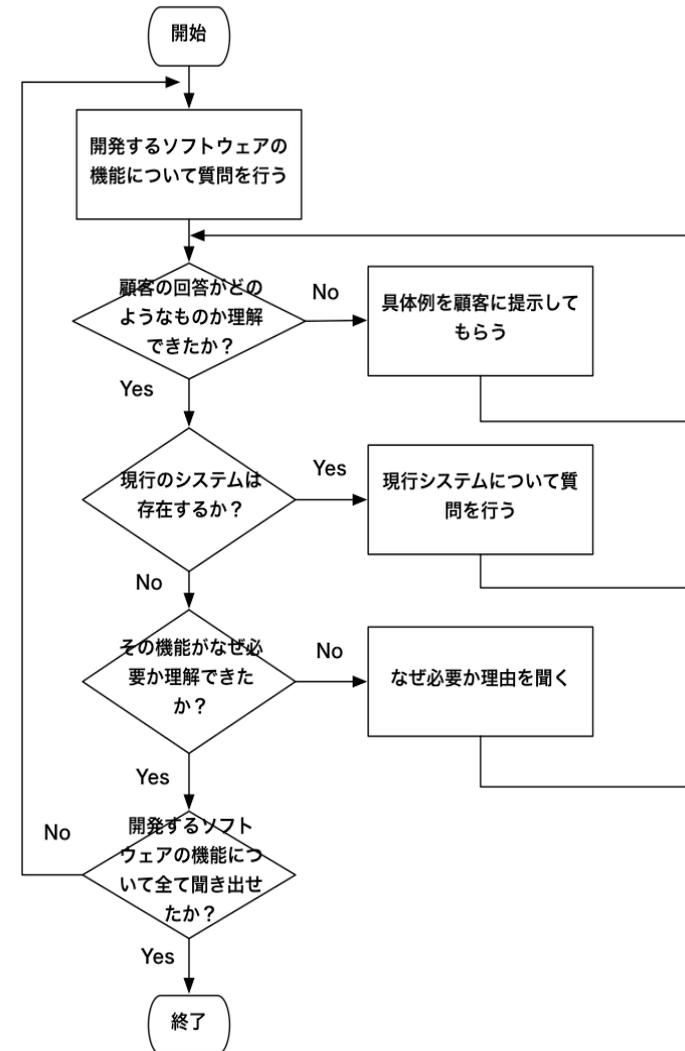


図 3 1st ステージでの誘導

### 5.3 現行システムの差分を聞く質問方法について

新規システムの機能要求を獲得するために必要な質問の遷移方法としては2つのパターンがある。

- a. 現行のシステムの機能を洗い出し、そこから問題点を見つけ出し、新システムの機能要求とする。
- b. すべての機能を新規システムの機能として質問を行い、その機能は現行のシステムの機能ではどのようにになっているのかを質問する。

本稿ではbの方法を対象とし、顧客から現行システムの仕様書をもらう事が出来ない状況でもSEを誘導できるような質問の誘導方法の提案を行った。

#### 5.3.1 現行システムに関する質問に用いられる話題のカテゴリ

現行システムに関する質問に用いられる話題のカテゴリを表3のように定義した。Current System カテゴリ内では3つの話題がある。”Example”では現行システムの機能の具体例を挙げってもらうための質問を行う。”Why”では開発される機能についての理由が聞かれる。”What about”では現行システムの機能をどのように変更を行い新システムで利用するのかを聞くための質問が行われる。

表 3 現行システムに関する質問で用いられる話題のカテゴリ

What		システムの機能に関する話題
Current System	Example	具体例に関する話題
	Why	開発背景についての話題
	What about	機能の修正場所に関する話題

#### 5.3.2 現行システムと新システムの差分を聞き出す質問の流れ

現行システムと新システムの差分を聞き出す質問の誘導の流れは、4つのパターンを考慮する必要がある。

- ① 現行システムの機能を、新システムでもそのまま用いる
- ② 現行システムの機能に修正を加えて新システムに用いる
- ③ 現行システムには無い機能を新システムの機能として新たに加える
- ④ 現行システムにある機能の一部を新システムでは削除する

差分を聞き出す質問の流れについての説明を行う。SEは、まず顧客に対してどのような機能を作りたいのか質問を行う。この時、顧客の提示するシステムの機能がどのようなかわからなかった場合は、顧客に対して具体例を提示してもらう。次に、現行システムが存在する場合、SEは顧客に対してその機能が先程挙げた①～③のパターンのどれにあたるのかを認識するための質問を行う。どのパターンか決定できたら、①～③のパターンで誘導を行う。これを繰り返し、新システムに必要な機能を全て洗い出す。そして、最後に現行システムにある機能で新システムでは削除される機能についての質問④を行う。

次に、①～④のパターンについて詳細を示す。

#### ① 現行システムに必要な機能を新システムでもそのまま用いる場合

まず、開発するシステムの機能が現行のシステムの機能でどのように実現されているかどうかのインタビューを行う。このとき、顧客の提示するシステムの機能がどのような物かわからない時は、具体例を示してもらうために質問(Example)を行う。そして、なぜ今回その機能をそのまま利用するのかの理由(Why)を聞き、その機能がシステムに本当に必要であるかの検討を行う。

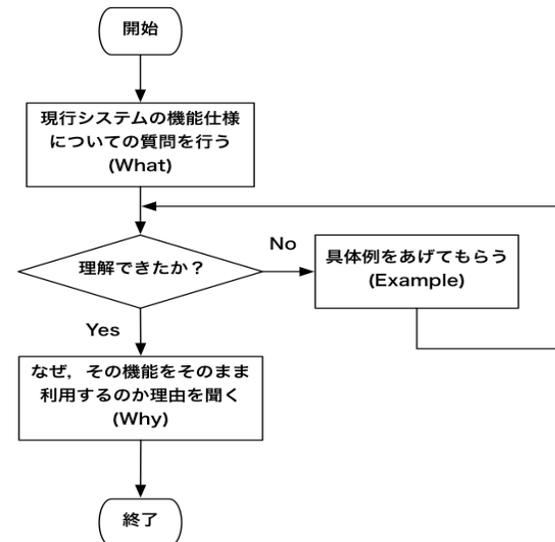


図 4 現行システムの機能を、新システムでもそのまま用いる場合

② 現行システムの機能に修正を加えて新システムに用いる場合

まず、開発するシステムの機能が現行のシステムの機能でどのように実現されているかどうかのインタビューを行う。ここでは、システムの機能に必要な前節で述べた項目についての質問が含まれている。このとき、顧客の提示するシステムの機能がどのような物かわからない時は、具体例を示してもらうための質問(Example)を行う。

次に、現行のシステムで機能の変更（機能修正，機能縮小，機能強化など）に関する質問(What about)を行い、その経緯に至った理由(Why)について質問を行う。

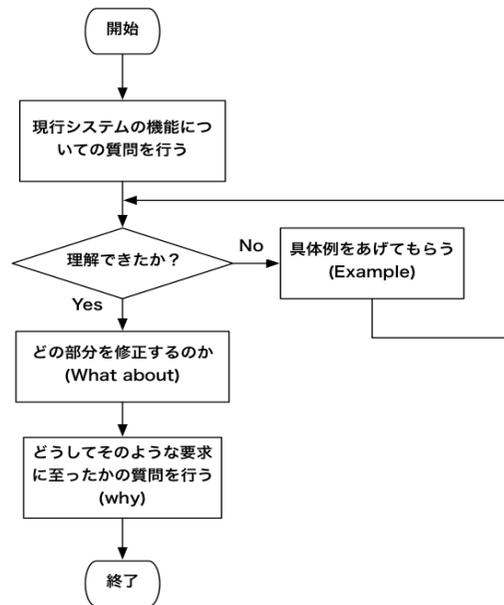


図 5 現行システムの機能に修正を加えて新システムに用いる場合

③ 現行システムには無い機能を新システムの機能として新たに加える場合

現行システムに無い機能を新システムに導入する場合は、その機能についての質問を行う。このとき、顧客の回答が理解できなかつたら、具体例を挙げてもらう。最後に新システムで、この機能が必要になったのか質問を行う。

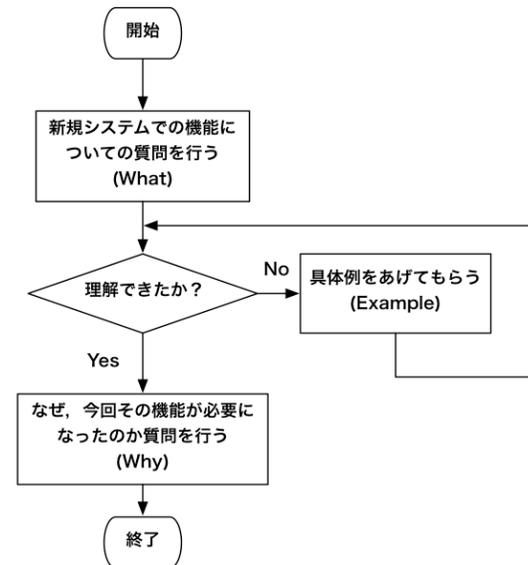


図 6 現行システムには無い機能を新システムの機能として新たに加える場合

④ 現行システムにある機能の一部を新システムでは削除する場合

①～③のパターンを用いて、全ての機能を聞き出した後に、最後に削除したい機能に関する質問を行う（図 7）。まず、最初に現行システムにある機能で新規システムでは削除される機能についての質問を行う。そして、なぜ今回その機能がなくなったのか理由を聞き出すための質問を行う。

5.4 質問レイヤの誘導方法

質問レイヤの誘導方法として、what カテゴリの質問の誘導方法について述べる。ここでの質問は開発すべきソフトウェアの機能に関する質問であり、必要となる機能はアプリケーション領域ごとに異なる。そのため、どのようなアプリケーション領域であっても利用できる汎用的な質問の誘導をする仕組みを考案しなければならない。

アプリケーション領域を絞ると、そこで求められる機能の選択肢も絞られてくる。そこで、顧客が求める機能のバリエーションのすべてを選択肢の形で表現できるようになるまでアプリケーション領域を絞る。このようにしたとき、機能の選択肢のそれぞれを、どのような機能が欲しいかという顧客への質問に対する顧客の予想回答と見なすことができる。その仕組みが図 8 である。

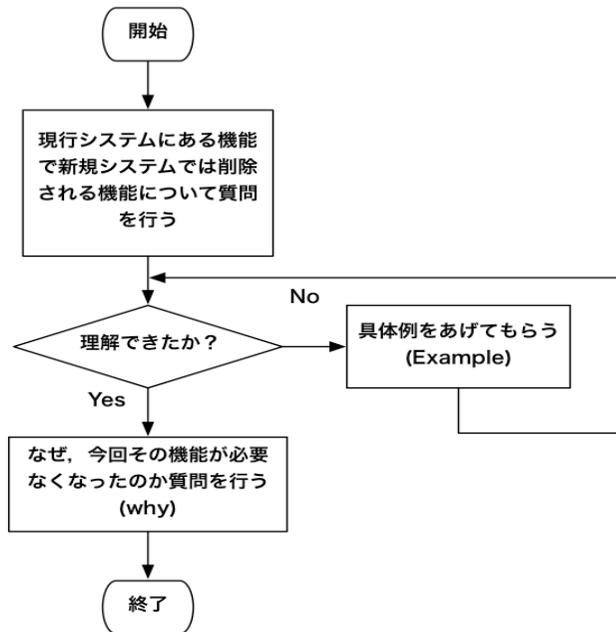


図 7 現行システムにある機能の一部を新システムでは削除する場合

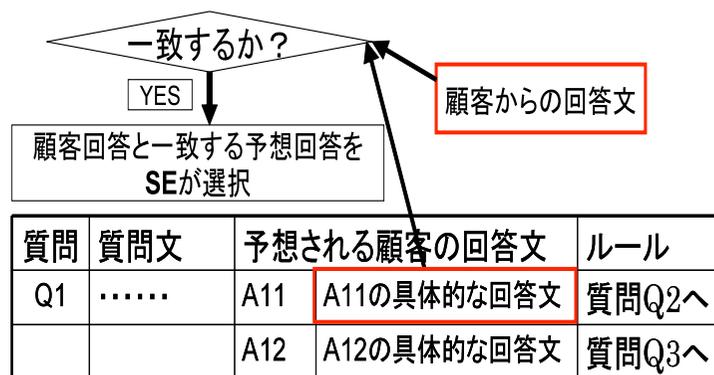


図 8 質問とそれに対する予想回答の選択から、次の質問を自動的に決定する仕組み

まず、本手法では顧客に聞くべき質問、質問から予想される顧客の回答の候補、顧客からの回答に対する次の質問がある。図 8 をシステムに用いたインタビューの流れは以下ようになる。

- ① システムが SE に対して質問文を提示し、SE がその質問を顧客に対して行う。
- ② 顧客が SE に対して回答を行う。
- ③ SE が、システムが提示する予想回答の中から、顧客回答に一致と思われるものを選択する。予想回答と顧客回答が一致しない場合は
  - A) SE は顧客に対して、顧客回答が予想回答に一致するか否か判断するための質問を行う。
  - B) 顧客から得た回答と予想回答が一致するか否か判断する。
  - C) 予想回答と一致する場合には、その予想回答を選択する。予想回答と一致しなかった場合は、予想回答と一致するまで A^B を繰り返す。
- ④ 選ばれた予想回答に対する次の質問文を自動決定する。システムが SE に対して次の質問文を提示する。

質問事項がなくなるまで①～⑤を繰り返す。

このように顧客回答を基に次の質問文を決定している。質問項目を動的に変化させることが出来るので、適切でかつ効率的なインタビューの誘導を行うことが出来る。

## 6. まとめと今後の課題

顧客が運用している現行システムを更改するときに、SE が顧客の要求を漏れや誤りなく抽出できるように、インタビューによる要求抽出作業を誘導するための仕組みを提案した。今後は、提案した誘導方法の有効性の検証を行うために、実際に使われている業務を用いて熟練 SE と初心者 SE を対象に、インタビューを誘導した場合と誘導しなかった場合で、インタビューにより抽出された顧客要求について適合率 (precision ratio)[2]と再現率(recall ratio)[2]を用いて評価を行う予定である。

## 参考文献

- 1) 古宮誠一, 加藤潤三, 永田守男, 大西淳, 佐伯元司, 山本修一郎, 蓬萊尚幸, “インタビューによる要求抽出作業を誘導するシステムの実現方法” 第 19 回 IPA 技術発表会論文集, pp.37-48, Oct.11-12, 2000.
- 2) 長島武生, “改良 Kepner-tregoe 法に基づくリスク識別法の提案と有効性の評価”, 電子情報通信学会学会論文誌, vol.J91-A, no.9, pp.883-897, 2008