

インタビューによる要求抽出作業を誘導するシステム ～顧客回答によって次の質問文を自動的に決定する仕組み～

An Interview-Driven Software Requirement Elicitation System: A Mechanism that Next Question Sentence Automatically Varies Depending on User Reply

山崎健広[†] 埜口元[‡] 谷藤史門[‡] 山中隆敏[‡] 古宮誠一[†]
Takehiro Yamazaki[†] Hajime Noguchi[‡] Shimon Yato[‡] Takatoshi Yamanaka[‡] Seiichi Komiya[†]

[†] 芝浦工業大学 工学部

[‡] 芝浦工業大学大学院 工学研究科

[†] Faculty of Engineering, Shibaura Institute of Technology .

[‡] Graduate School of Engineering, Shibaura Institute of Technology.

要旨

ソフトウェア開発では要求仕様書を基にシステム開発を行う。もし、要求仕様書に漏れや誤りがあれば、その後の成果物にも漏れや誤りが含まれてしまう。そのため要求仕様書を作成するための要求抽出作業は非常に重要な工程である。そこで我々は要求抽出作業をインタビュー技術と捉え、要求抽出作業を正確に行えるようインタビュー作業を誘導するシステムを構築中である。本稿ではインタビューにおける次の質問をシステムが動的に決定する仕組みについて述べている。

1. はじめに

要求抽出作業はソフトウェア開発の最初の工程であり、要求抽出作業から得た顧客の要求を基に要求仕様書を作成する。もしこの要求仕様書に漏れや誤りがあると、これを基に作成する成果物にも漏れや誤りが発生してしまう。そのため要求仕様書を作成するための要求抽出作業は非常に重要な工程である。と同時に要求抽出作業は大変困難な作業でもある。その大きな理由は、顧客とソフトウェア技術者 (SE) との間で共通の知識を持っていないからである。具体的には、顧客は自分たちの業務に関しては深い知識を持っているが、システム開発に関して殆ど知識を持っていないからであり。SE はシステム開発に関しては深い知識を持っているが、顧客の業務に関しては殆ど知識を持っていないからである。そのためお互いにコミュニケーションが上手くとれず、要求抽出作業にも漏れや誤りが発生してしまう。結果的に要求仕様書にも漏れや誤りが発生してしまい、ソフトウェア開発は大きなダメージを受けてしまう。このため、我々はインタビューにおける要求抽出作業を適切に行うために、計算機で SE の要求抽出作業を支援 (誘導) するシステムを構築中である。本稿では SE が顧客に対して行う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。

2. 熟練 SE と初心者 SE との比較実験で判明したこと

表 1 話題の分類

What	開発すべきソフトウェアの機能
Example	例えばどのようなことができるか
Why	開発の理由, 背景
Current System	現行システムについての質疑
Constraints	他システムとのインタフェースなど
Policies	開発方針やこれに基づくアーキテクチャ
Conditions	各種インタフェースなど
Budget	開発予算
Schedule	開発期間

古宮ら [2] が行った、インタビューによる要求抽出に関する熟練SEと初心者SEの比較実験により、種々のことが判明している。

- (1) SEが採り上げる話題は表1のような9つの話題のカテゴリに分類される。
- (2) インタビューによる要求抽出では、話題の遷移方法について、初心者SEには明確な遷移パターンはなかったが、熟練SEには図1のような明確な遷移パターンがある。

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

図1 話題の遷移パターン

ここで、{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と次のカテゴリの話題には移らないことを意味する。このような話題の遷移方法をシリアルタイプと呼ぶ。また、{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あることを意味する。このような話題の遷移方法をパラレルタイプと呼ぶ。

図1に掲げた話題の遷移パターンをよく見ると、これらは次のことで共通している。即ち、{What | Examples | Why | Current System}は、ここで採り上げられる話題がいずれも開発予定のソフトウェアに対して顧客が求める機能に関するものであり、それらがパラレルに展開される。{Constraints, Policies, Conditions}は、ここで採り上げられる話題がいずれもソフトウェアの作り方に影響を与えるものであり、それらがシリアルに展開される。{Budget, Schedule}は、ここで採り上げられる話題が開発予算や開発期間に関するものであり、それらがパラレルに展開される。

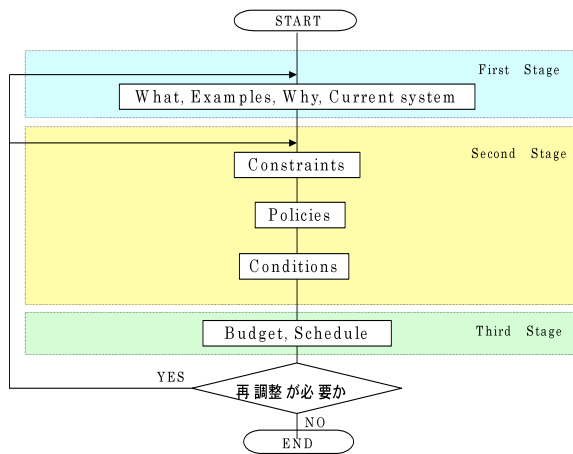


図2 話題の遷移パターン

そして、採用される遷移パターンは、顧客との親密度の相違によって異なる。疎遠な関係にある顧客に対する話題の遷移パターン(パターン1)をフローチャートの形で示す。非常に親しい顧客に対しては、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。疎遠な関係にある顧客に対しては、最後に開発予算や開発期間に関する話題が採り上げる。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に関する話題に戻る必要が生じる。どちらとも言えない間柄にある顧客の場合には、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しようとしていることが分かる。

3. インタビューを誘導する方法

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、SE が顧客に対して行う要求抽出作業をインタビュー作業であると捉え、初心者 SE が行うインタビュー作業を誘導することを考える。

3.1. 話題と質問の2階層モデルについて

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、1つのカテゴリの中で質問する作業を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3のような2階層モデルを採用する。

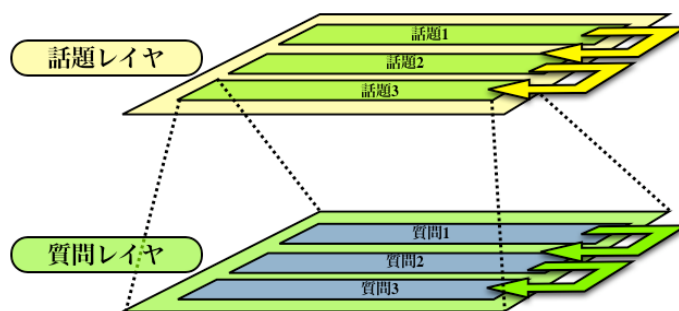


図 3 話題と質問の2階層モデル

3.2 上位層で話題のカテゴリの遷移を誘導する方法

上位層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。進捗管理テーブルの詳細は[2]を参照して下さい。ここでは紙面の都合で掲載を省略する。

3.3 What カテゴリでの質問を誘導する方法

What カテゴリの中での質問(開発予定のソフトウェアに求められる機能についての質問)を誘導する方法を明らかにする。What カテゴリ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。顧客が求める機能の選択肢が絞れるとき、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4である。

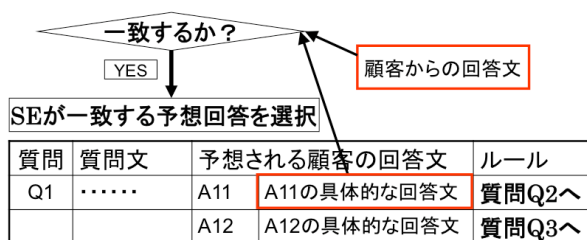


図 4 質問項目と回答内容のモデル

図4のように、SEが顧客にインタビューする質問の文章と、それに対する顧客の予想回答の文章を設定する。予想した回答が返ってきた場合、それに対して次に行う質問がどれであるかを設定する。(つまり、インタビューを誘導するためのルールを設定する。) 質問と予想回答との関係は AND/OR 木で表現できる。AND/OR 木の階層が最も浅くなるように質問の順序を設定することが肝要である。何故なら、AND/OR 木の階層が浅ければ浅いほど、同じことを聞き出すのに必要な質問の回数が少なくて済むからである。なお、顧客の回答がシステムで設定した予想回答のどれに当てはまるのかわからない場合には、SEはそれを識別するための質問を顧客に行えばよい。

顧客が要求した機能(What)がどのようなものであるか理解できなかった場合には、具体例(Example)を提示してもらい、顧客が要求した機能が何故必要なのか理解できなかった場合には、そのような機能が何故(Why)必要か質問する。顧客が現在使用中のシステム(現行システム)がある場合、その機能は現行システム (Current System) ではどうなっているか質問する。顧客が要求した機能がどのようなものであるか理解できなかった場合、このように What→Examples→Why→Current System の順序で質問する。この様子をフローチャートで図5示す。

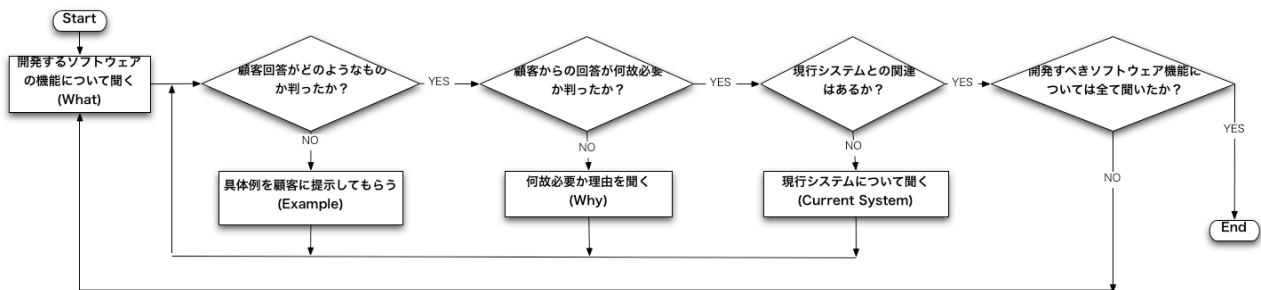


図5 顧客が要求した機能がどのようなものであるか理解できなかった場合における質問の順序

4. おわりに

ソフトウェア開発において要求抽出は非常に重要な工程である。しかし熟練 SE であっても、開発予定のソフトウェアに対する要求を漏れなく、効率よく行うことは容易ではない。このため我々は、SEが顧客に対して行う要求抽出作業を誘導するシステムを構築中である。本稿では、初心者 SE でも熟練 SE と同じように要求抽出できるように、要求抽出作業を SE が顧客に対して行うインタビュー作業であると捉え、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した。

本稿では述べなかったが、IEEE Std. 830-1998 に基づく要求仕様書を自動生成する機能も開発する予定である。

5. 今後の課題

本稿で提案した、インタビューによる要求抽出作業を誘導する仕組みに基づいてシステムを構築中である。本稿で提案した仕組みが有効であることを証明するために、この仕組みを用いたシステムを利用して要求抽出作業を行う場合と、この仕組みを用いたシステムを利用しないで現場の SE が顧客にインタビューする場合とを比較する実験などを検討している。

参考文献

- [1] IEEE Recommended Practice for Software Requirements Specifications, IEEE Std. 830-1998.
- [2] 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回技術発表会論文集, pp.37-48, 2000.