

5

ことばを知る・使い方を知る・可能性を知る

ソフトウェア開発における「ことば」の重要性



高藤 忍
SAITO Shinobu

日本電信電話株式会社/コンピュータ&データサイエンス研究所/特別研究員

ソフトウェアを作るには、それを利用するユーザや顧客の要求を正しく獲得することが重要である。しかしながら、ヒトが話す「ことば」には曖昧さが伴う。ソフトウェア開発者は「ことば」からどのような影響を受け、どのように乗り越えようとしているのだろうか。

IT業界の2つのビジネスモデル

様々な企業や個人等が参画しているIT業界であるが、主要なビジネスモデルは「システム開発」と「企画型サービス」の2つがある。

・システム開発

顧客の情報システムの 企画、設計、開発等を受託・納品し、顧客からの支払い金額を売上に計上する。システム納品後には、顧客のシステムの運用・保守を受託する場合もある。

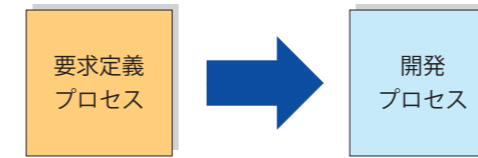
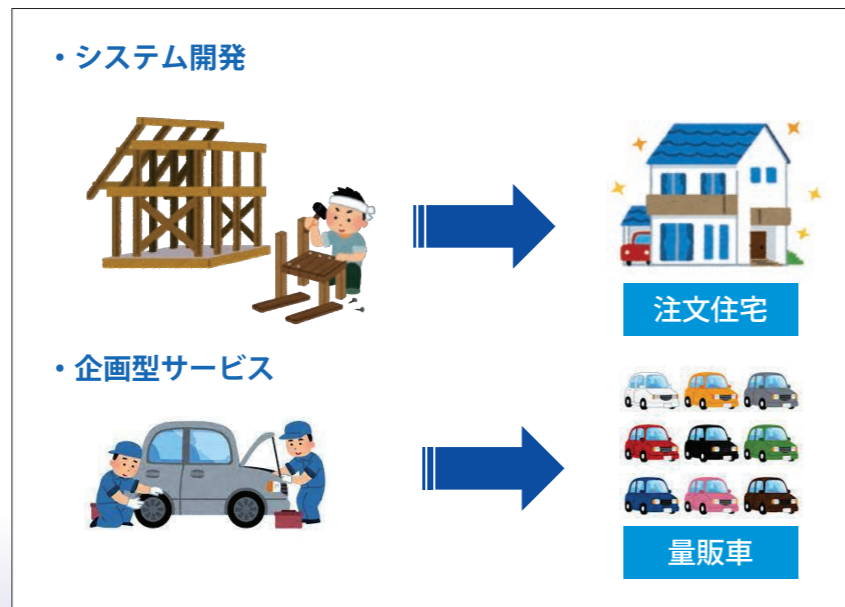
・企画型サービス

複数の顧客の利用を見越して設備投資を行い情報システムを開発し、資産として保有する。顧客のシステム利用時に支払うサービス利用料を売上に計上する。

システム開発を担う企業は「System Integrator(システムインテグレーター)」や、省略して「SIer(エスアイヤー)」と呼ばれる。NTTデータや富士通が国内の代表的な企業である。一方、企画型サービスには様々な種類が存在する。例えば、音楽や映像の配信サービスを担う代表的な企業では、SpotifyやNetflixが挙げられる。

こういった企業のビジネスモデルは、最新、且つ複雑な印象を持

たれるかもしれないが、決してそんなことはない。しかしながら、システム開発は「注文住宅」を建築するハウスメーカーと同じビジネスモデルであり、企画型サービスは「量産車」を生産するクルマメーカーと同じビジネスモデルであると考えられる。前者は、家が欲しい人(注文者)からの注文を受けた後、彼らの要求(外観や内装の好み等)に基づいて家を建築し、引き渡しを行う。後者は、車を買って欲しい人(購入者)を想定し、彼らの要求(車の色や形等)を推測して車を生産し、販売を行う。このように考えればIT業界のビジネスモデルも分かり易くなるのではないだろうか。



システムを作り上げる2つのプロセス

話をITの世界に戻す。一般にシステムは2つのプロセスで作られる¹⁾。最初の「要求定義プロセス」で、対象のシステムやサービスに顧客が求める要求を定義し、次の「開発プロセス」で定義された要求に基づき、文字通りシステムを開発する。なお、「Requirements」に対する訳語として、国内では「要求」と「要件」の2つが用いられているが、本稿では「要求」で統一する。

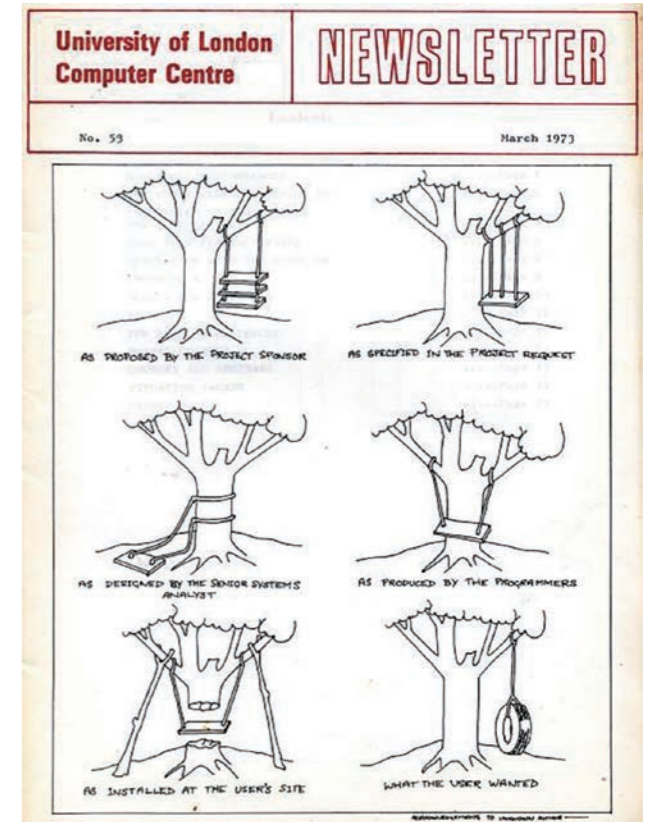
要求定義プロセスは、前述の「注文住宅」や「量産車」でも当然存在する。例えば「量産車」を作るクルマメーカーであれば、売れそうな(想定した購入者が好む)クルマをデザインすることである。アウトプットは、例えば、クルマの模型や3次元のCG画像を作ることだろう。ハウスメーカーも同様である。これらの業界における要求定義プロセスでは、どのようなことをするのか、何をアウトプットするのかのイメージが湧きやすい。

しかしながら、IT業界における要求定義プロセスのアウトプットをイメージすることは難しい。特にシステムやサービスの主要な構成要素であるソフトウェアは、開発者がプログラムを実装(プログラミング)することで作り上げていく。最終的な完成品も住宅やクルマと異なり、予め見たり触ったりすることができない。IT業界のビジネスの現場では、ソフトウェアに対するユーザの要求は文章で記述することが多くなる。

開発の専門家ではないユーザは、日々の生活で特に意識せずに使っている(話している)「ことば」を用いて、自らの要求を伝達(口伝)しがちである。伝達された要求は、開発者の誤った理解や理解不足を誘発する。結果として、ユーザの要求に即さないシステムやサービスが作り出されてしまう。

ソフトウェア開発は究極の伝言ゲーム

「ことば」を紡いでいながらソフトウェアをつくる難しさをユニークに表したものが、通称「ブランコの木の話(Swing-Tree Story)」と呼ばれる絵で



ブランコの木の話 (Swing-Tree Story)³⁾

ある。ソフトウェア工学では古くから伝わる非常に有名な絵であり、1973年にロンドン大学コンピュータセンターのニュースレターに掲載されたものである²⁾。

この絵が示唆することは、「木にぶら下がってスイング(揺れること)ができる遊具が欲しい」という(ことばで記された)最初の顧客の要求に対して、複数の人(開発者)の解釈が加わり、最終的には当初のイメージとかけ離れたものが作り出されてしまうことである。

以降では、筆者の解釈も加えながら、この絵が示唆するストーリーを述べていく。はじめに、左上の図は、顧客が当初欲しいと思っていた遊具のイメージである。このイメージを「ことば」で一人目の開発者に伝え、その開発者は「ロープは2本より3本の方が板の安定性は増すだろう」と考えて、右上の図のようにイメージをする。次の開発者は「安定性を増すのであれば、ロープを枝ではなく幹に巻き付けた方が良さだろう」と考え、左中の図のイメージを抱く。次の開発者は「双方の枝にロープを括り付けた方が板は安定するだろう」と考えて、右中の図の

イメージに至る。更に、次の開発者は「いくら安定していても、板は揺れないとだめだろう。幹の一部に空間をつくり、板が揺れるようにしよう。その場合、双方の枝に支えがあった方が良さだろう」と考え、左下の図のようなイメージとなる。いよいよここまでくると、人が乗れるものでもなくなる。実際には2番目の開発者の左中のイメージの時点で、顧客の最初のイメージとは全く異なるものができてしまっている。ソフトウェア開発は究極の伝言ゲームであり、このゲームの攻略（顧客の要求を開発者に正しく伝達すること）がいかに難しいことであるかを物語っている。

ちなみに、この絵にはもう一つの示唆がある。それが右下の絵である。結局のところ顧客の要求は、木にぶら下がってスイングすることである。それを満たすのであれば、ロープを木の枝から一本だけ垂らしてタイヤに巻き付ければ良いのである。たとえ顧客の最初のイメージ（左上の図）と違っていても、顧客の要求に応えることはできる。この絵は「顧客の言われるままに（イメージのまま）作るだけでなく、もっと良い方法（実現手段）を開発者は考えるべきである」ことを物語っている。この絵を初めて見た時から、非常に味わい深い絵であると筆者は思っている。

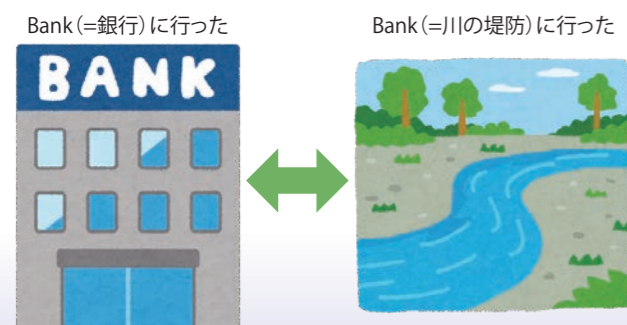
要求の曖昧さ

「ことば」による要求が、開発者の誤った理解や理解不足を誘発するのは、「ことば」には曖昧さが伴うからである。自然言語で記述された要求の曖昧さには5つの分類がある⁴⁾。以降では、各分類について例文を示しながら解説していく。

①語彙の曖昧さ

例文：She walked to the bank.

この文章の訳としては「彼女は銀行に行った」、も



曖昧さの5分類

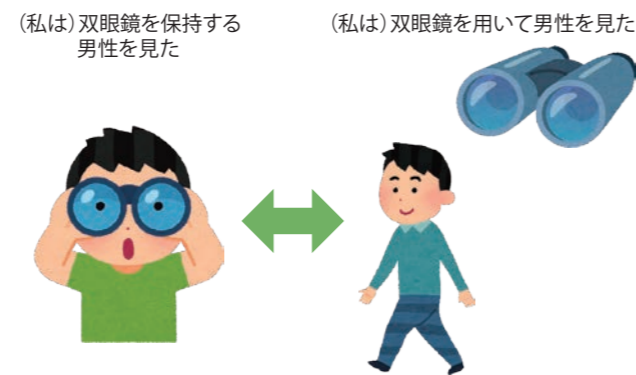
曖昧さの分類	説明
①語彙の曖昧さ (Lexical)	ある単語に異なる別の意味が存在する
②統語的な曖昧さ (Syntactic)	文章に複数の構文解釈 (妥当な統語関係) が存在する
③意味的な曖昧さ (Semantic)	文章の意味の解釈が複数存在する
④参照先の曖昧さ (Referential)	文章中の参照先の解釈が複数存在する
⑤不明瞭な表現の曖昧さ (Vagueness of terms)	文章中に外延的な定義が、一般的に不明か定義が困難な「表現」が含まれる

しくは「彼女は川の堤防に行った」の2つが考えられる。単語「bank」が複数の意味を持つためである。本例では、「銀行」と「川の堤防」のどちらの意味であっても解釈可能であるため、曖昧な文章となっている。

②統語的な曖昧さ

例文：I saw the man with binoculars.

この文章の訳としては「私は双眼鏡 (binoculars) を保持する男性を見た」、もしくは「私は双眼鏡を用いて男性を見た」の2つが考えられる。前者は、男性が双眼鏡を保持しており、私はその男性を見たことを意味する。後者は、私が双眼鏡を保持しており、その双眼鏡を使って男性を見たことを意味する。この例は、英語特有の文法に起因するものである。日本語でも構文関係が起因 (例：係り受けの違い) で複数の解釈となる例は数多く存在する。

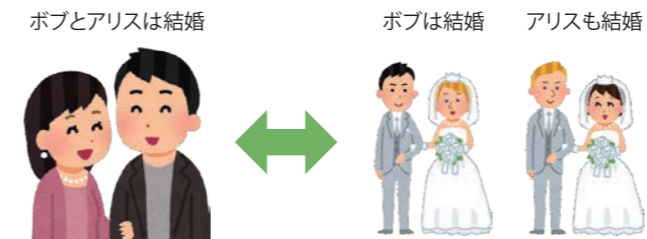


③意味的な曖昧さ

例文：Bob and Alice are married.

この文章の訳としては「ボブとアリスは結婚している」、もしくは「ボブは (誰かと) 結婚しているし、アリスも (誰かと) 結婚している」の2つが考えられ

る。前者は、既婚カップルは1組 (ボブとアリス) だけである。後者は、婚姻関係にあるカップルが2組となる。本例では、文章から想定される状況が複数の可能性があるため、曖昧な文章となっている。

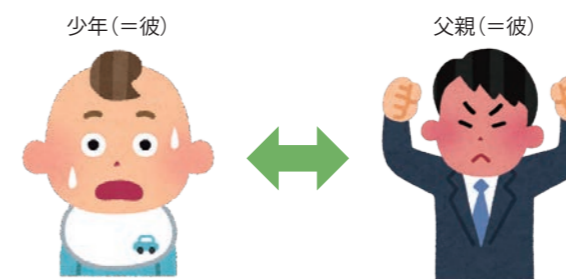


④参照先の曖昧さ

例文：The boy told his father about the damage.

He was very upset.

この文章の訳としては「少年は彼の父親に損傷について伝えた」に続き、「少年 (= 彼) はとても動揺していた」、もしくは「父親 (= 彼) はとても動揺していた」の2つが考えられる。前者は、損傷を受けた少年自身が動揺していたことを意味する。後者は、息子の損傷を聞いて父親が動揺していたことを意味する。代名詞 (He) が指し示す意味が複数の可能性があるため、曖昧な文章となっている。



⑤不明瞭な表現の曖昧さ

例文：George is tall.

この文章の訳としては「ジョージは背が高い」となる。これまでと異なり、複数の訳文は存在しない。一方、「高い」は不明瞭な表現である。日常的に広く使われており、各人のそれぞれの想定のもとに解釈できるからである。例えば、アメリカ合衆国初代大統領のジョージ・ワシントンは6ft (約182.9cm) であったと言われている。これは当時のアメリカ男性の平均身長からすればとても高い身長である。一方、現代のNBA (アメリカプロバス



ケットボール) の男子選手の平均身長は約2mである。身長6ftのジョージ選手がいたとすると、彼の身長は決して高い身長といえない。

曖昧さ避ける・見直すための取り組み

ソフトウェア開発の世界では、これまで見てきたような曖昧さを避けるための様々な取り組みが行われている。例えば、①語彙の曖昧さを避けるには、文章中の語を限定する用語集の利用が有用である。②統語的・③意味的な曖昧さを避けるには、要求文章の構文パターンを予め限定することが効果的である。能動態で書く、「そして」「また」などの複数の文をつなげる接続詞を避ける記述スタイルも有用である。また、④参照先の曖昧さを避けるためには、指示語・代名詞ではなく名詞を再記述する方法が一般的である。

しかしながら、⑤不明瞭な表現の曖昧さは、見つけることも見直すことも難しい。これらの表現は、外延的な定義が一般的に不明であるからである。外延とは、論理学の用語で「集合における概念に対する具体的な物や事」を意味する。即ち、外延的な定義が不明とは、具体的にどんなものがあるか (実体) を、多くの人に受け入れられる内容で示すことができないことであると言える。これまでに筆者らは、要求文章における不明瞭な表現の曖昧さの発見・見直しを支援する研究に取り組んでいる^{5,6)}。興味のある読者はぜひ参照してもらいたい。

<参考文献>

- 1) 共通フレーム2013、独立行政法人情報処理推進機構 (IPA) ソフトウェア高信頼化センター (SEC)、2013。
- 2) 鶴林尚靖、レクチャーソフトウェア工学、数理工学社、2021。
- 3) University of London Computer Center Newsletter, No.53, March 1973。
- 4) Klaus Pohl, Requirements Engineering: Fundamentals, Principles, and Techniques, Springer Publishing Company, Incorporated, 2014。
- 5) 齋藤忍、飯村結香子、山田節夫、曖昧表現の見直しノウハウの形式知化、情報処理学会論文誌, vol.62, No.4, pp.982-994, 2021。
- 6) 齋藤忍、山田節夫、飯村結香子、ビジネスルール可視化の効果検証—理解容易性の観点からの自然言語記述と図表の比較実験—、情報処理学会論文誌, vol.57, No.12, pp.2784-2794, 2016。