

## 「意図」の位置

漆原 憲博  
株式会社ジェーエフピー  
japanfp@jfp.co.jp

佐々木 千春  
株式会社ジェーエフピー  
sasaki1000@jfp.co.jp

### 要旨

納品したシステムが、発注者の最終の目的あるいは端的に「意図」とでもいうべきものを満たしていないがゆえに、受入の承認を受けられない場合がある。自社製品の最終の妥当性確認が得られなかった場合も含む。本論では「意図」の解釈の随意なところがその原因であると考え。また「意図」を、システムに多大な改変を与える大きな要求と考え、まずは本稿を始める。その後、「意図」を解釈する上で基本的な考えがあるということ、それを踏まえて、意図が開発の最後に問題を起こさないようにするための意図の扱い、すなわち開発プロセスでの管理法を提案する。挿話的に意図の前向きな効果も入れた。最後に「意図」とシステムの関わりを述べ、意図の研究を開発論の中に取り入れることを提案する。

### 1. はじめに

システムが、要求文書(要求仕様書)を過不足なく満たしているにもかかわらず、そのシステムが(受入検査において)承認されない場合がある。ここで、要件文書は利害関係者に承認された(権威のある)ものとする。

システムは要求文書を満たしているならば、承認されるのが当然である。しかし、当然ではない事態がまれに起こる。そしてこの「ご無理(な要求)」が土壇場で通ったりもする。この変転の因子は何か。「システムは X を満たしていないならば、X は承認されない」という X が存在するといわざるを得ない。

X の候補は何か。「意図」と思われる。次のような問答は、各位多少覚えがあるかもしれない。

「タブレットの立ち上がりが十分速く、と言った筈なのにこれでは遅すぎる」

「十分速いと思いますが、他社より速いですよ。」

「一番？」  
「調べたかぎりでは」  
「そう？ でもホントに一番でないとダメだ。他社も全部調べて。」

この会話では、「十分速く」は相互に承認されているようである。が最後は、発注者の発言は、「一番速く」に変わっている。

「十分速くとは、もっとも速く、とほぼ同義なはずだ。もっと速くとは一番速くという意味だ。」

かなり無茶である。そして最後は、「別途請求は無理だよ。」(と、いったとか、いわないとか。)

「意図」の意味は辞書にはこうある<sup>1</sup>。

① 何かをしようと考えること。「一した半分もできない」

② こうしようと考えていること。めざしていること。「敵のーを見抜く」

この①と②の例にならえば、発注者からいえば、「意図した肝心の部分ができていない」となり、受注者(以下、「開発者」と呼ぶ)からいえば、「発注者の意図を見抜けなかった」ことになる。

このような開発をめぐる行き違いは海外への委託でも頻繁にある。ちなみに「意図」は「intention」というが、この用語は論理学用語の内包「intension」の原義をなしている<sup>2</sup>。内包は外延「extension」と対をなす。先例においては、内包の抽象的な意味を外延化し、具体的な意味に起こし直せばよかったのであろうか。そこでこう言うしてみる。

開発者:「では、何秒だとよかったですか？」

発注者:「あなた方プロでしょうが、いずれ客先でうちの営業マンがタブレットを開けたとき、なんと思う？ こんなに待たせるなんて、使い物にならん。」

どうも、「一番」ということもなさそうであるが、いかんせん話がこじれてしまった。ただし、事がおさまって後日聞けば、営業マンに「世界一速いと思える」専用タブレットを持たせ、自信を持って営業をしてもらいたかったそうだ。

<sup>1</sup> 三省堂 大辞林 第三版

<sup>2</sup> <https://www.lexico.com/>

このような相手に少し寄り添ったやり取りが開発の当初からあったならば、と思うが、後悔先に立たず。人は実物を見ないと実感にはいたらない。そんな人間特性も開発の一部かもしれない。

現実には取りに足らぬ意図もあるわけであるが、他方傾注に値する意図を、この例を参考に定義すると(微妙に異なる事例はたくさんあるとして)、意図とは感性的な要求で、かつ開発工程の最終あたりに登場し、納入リスクをはらむ要求といえる。ここで感性的な要求とは、数値等では決めがたいもの、さらにいえば、決めようとする開発者(受託者)が、発注者に押し切られてしまいそうなものともいえる。逆に押し切るくらいに、発注者は強い動機を内在している。内在とはここでは本人も言葉に出して明確には言えないもの、というぐらいの意味である。

ところで、このような話は社内にもある。むしろ、要求文書を明確にやり取りしない社内の製品の開発にこそ多いかもしれない。本論は、社外であれ、社内であれ、意図により、無駄が発生しないために、あるいは振り回されないようにするためにどうすべきかを論じる。

## 2. 意図と要求の関係、また登場人物たち

意図が開発サイクルの中で、要求にどう関わってくるのかを確認する。このことは意図を扱う上での基本である。

先の発注者と開発者の関係を模式化するとすると、意図と要求、また発注者と開発者の関係が下図のように描ける。

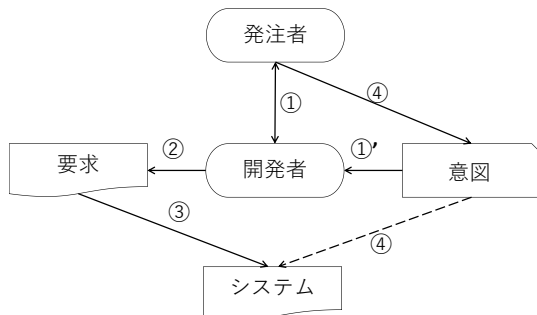


図1 要求と意図の関係

図1では、発注者と開発者が協議し合意の上で①、要求文書が作成され②、その要求文書に基づいて、システムが開発されること③を物語っている。④では出来上がったシステムの最終的な妥当性の確認を示している。そしてこのとき、妥当性の基準は「意図」であることを、発注者からの矢印④で示している。

またこの意図は、「①'」でも表現されている。これは①

と同じ初期の工程でも、発注者の意図を開発者は「汲むべき」ということを表現している。②は、①で両者が合議したものを開発側で作成していることを表わしている。また、加えて「①'」で意図を開発者につなげることで、②での要求文書に開発者は意図も反映させることを示している。

さて、図がこうであるとするならば、④において、妥当性の承認が下されないのは、おかしいということになる。なぜなら、工程①において、また①'において、意図は要求文書に十分盛り込まれているはずだからである。

事例ではしかし、騒ぎが起っていた。仔細は後に論じるところとして、このような騒ぎに巻き込まれるであろう関係者を挙げてみる。後に描く図の各工程に、彼らの多大な工数とストレスが発生することを、ご想像ねがうためである。

開発者は、要求開発者、設計者、実装者、検証者に分かれる。検証者は開発者に含まれるが、他方開発されたシステムの妥当性を確認する者は発注者に含まれる。また、発注を責任をもって行う者がいる。発注責任者である。この発注責任者が通常は妥当性確認の最終的な判断(承認か不承認)をする。

他方、開発者側にも責任者がいる。この開発責任者が、要求文書の開発責任者であり、要求文書の承認、さらに妥当性確認の承認か不承認の通知を受ける人、またその通知に対して同意か否かの諾否を下す人とする。図がこれら登場人物一覧である(図2)。

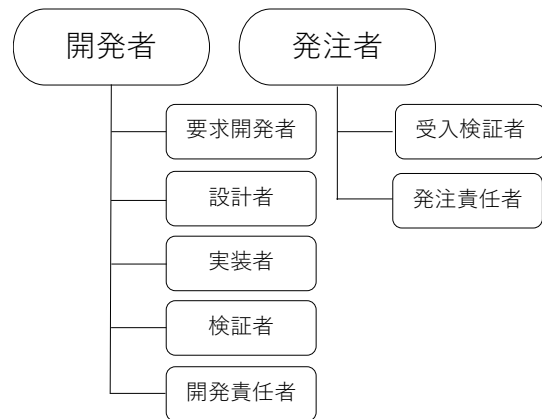


図2 登場人物

## 3. 課題の精査

図1の通りに開発工程が進めば問題ないはずのものが、問題が起きた。工程の各々を疑ってみる。「あやしさ」や「ずれ」また「反映の程度」などを「不確かさ」と呼び、図

3に表す。図3では⑤が新たに加わっている。

- (1) 発注者と開発者の意図と要求の合意のあやしさ。(①, ①')
- (2) 合意を開発者は要求文書に完全に反映できているか(②)。また開発者は発注者の意図を要求事項として作成(要求化)し、要求文書に完全に反映できているか(①')
- (3) 要求文書が完全であったにしても、開発したシステムにその要求が完全に反映されていない可能性がある。そのとき意図はどこに? 要求の中に在る?(③)
- (4) 意図が開発の当初と終わりごろでは変わっていないか(④)。
- (5) システムは承認か(⑤)。
- (6) 不承認の原因は意図か(⑥)

図3では“No”が不合格を示し、工程を遡るべきことを

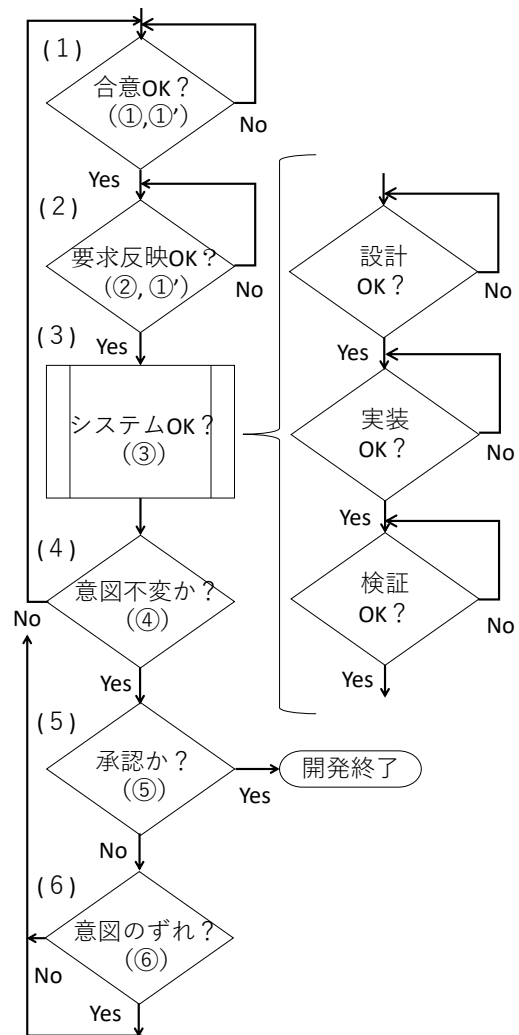


図3 発注者と開発者の不確かさの発生可能箇所

示している。“Yes”になるまで行う。

図3では(5)でシステムの承認の可否が問われ、不承認であった場合に、(6)で意図がその原因かと問うている。このパスは(4)では意図は最初と変わっていないと言っている。しかし、最後に突然意図が変わっている。

このようなことがありうるのだろうか。(4)の意図と、最後の(6)の意図が、言葉は「意図」で同じだが、中身は別物という他ない。何かが現れたのである。これが本論の結論だが、果たしてそのようなことがあるのだろうか、一体なにが現れたというのか。

### 3.1. 「意図」が明らかに関係する工程の点検

工程を遡ると、まず最初の工程の「合意 OK?」の中に

意図に関する「①」がある。「要求反映 OK?」の中にも「①」がある。途中で意図が変わっていないか(「意図不変か?」(④))も意図に関する。よって、意図はこれらの工程で変わっているのはと思わせるが、本論では(4)での意図は最初の意図と変わらなかったとしているので、それに従う。

他方、意図ではなく、要求の変更はこの意図の変化に影響をおよぼさないであろうか。要求が変更され、開発が先頭にまで戻るような場合は、要求の大きな変化である。要求も軽微な変更であれば、意図まで影響しないと経験的にいえるが、大きな変更の場合はどうであろうか。

大きな要求の変更は意図の変更に等しいと考えられる。というのも、意図は要求の上位の概念に属し、意図は要求を含んでいるからである。その小さいはずの要求が大きく変わるといふのであれば意図に影響しないはずはない。「より速く」が「一番に」に変わったとすれば、大きな要求の変更である。志向することが変わったからである。

事例では、営業マンへの動機づけは、相対的なもの(「より速く」)から、唯一絶対(「一番に」)に変わった。これは、システムの技術的な変更におさまらない、大きな経済的な価値の変化ではないか。なにしろ「一番」なのだから。

もちろん、発注者は不変と言うかもしれないが、最終のシステムの出来栄を見て、これなら「一番を」と趣旨変えをしたのかもしれない。意図が(当初の要求の)技術的機能の境界を超えて、約束外のところに向かうところに、意図のおさまりきれないところがある。

ところで、要求に理由を付した要求の記述法がある<sup>3</sup>。次の例がある。事例 B とする(先を A)。

要求:省電力設定をするかしないかを選択できるようにする。

理由:省電力よりもマルチメディアコンポの性能を引き出したい場合があるため。

一般に理由は意図の代替と考えてよい。またこの例⑧では、要求と意図(理由)は意味が近い。他方先の例(④とおく)では、「より速く」とか「一番速く」という意図は、実現させるとすれば、意味のかけ離れた事項を実現させることによって叶うことになる(メモリ増強、その他諸々の機能の実現によって)。

<sup>3</sup> 例:<https://www.jfp.co.jp/slp/exercise.html> 理由の記述は、渡辺滋氏(サイドクロス社)。 <https://www.side->

[cross.com/](https://www.cross.com/)

この例⑥のように要求と理由が意味的に近い場合は、変更があったにしても開発プロセスの中で十分吸収可能

遡及する場合もあろう。設計を再検討する際には、要求が問われ、その場合には先の(例⑥)でのような理由も問

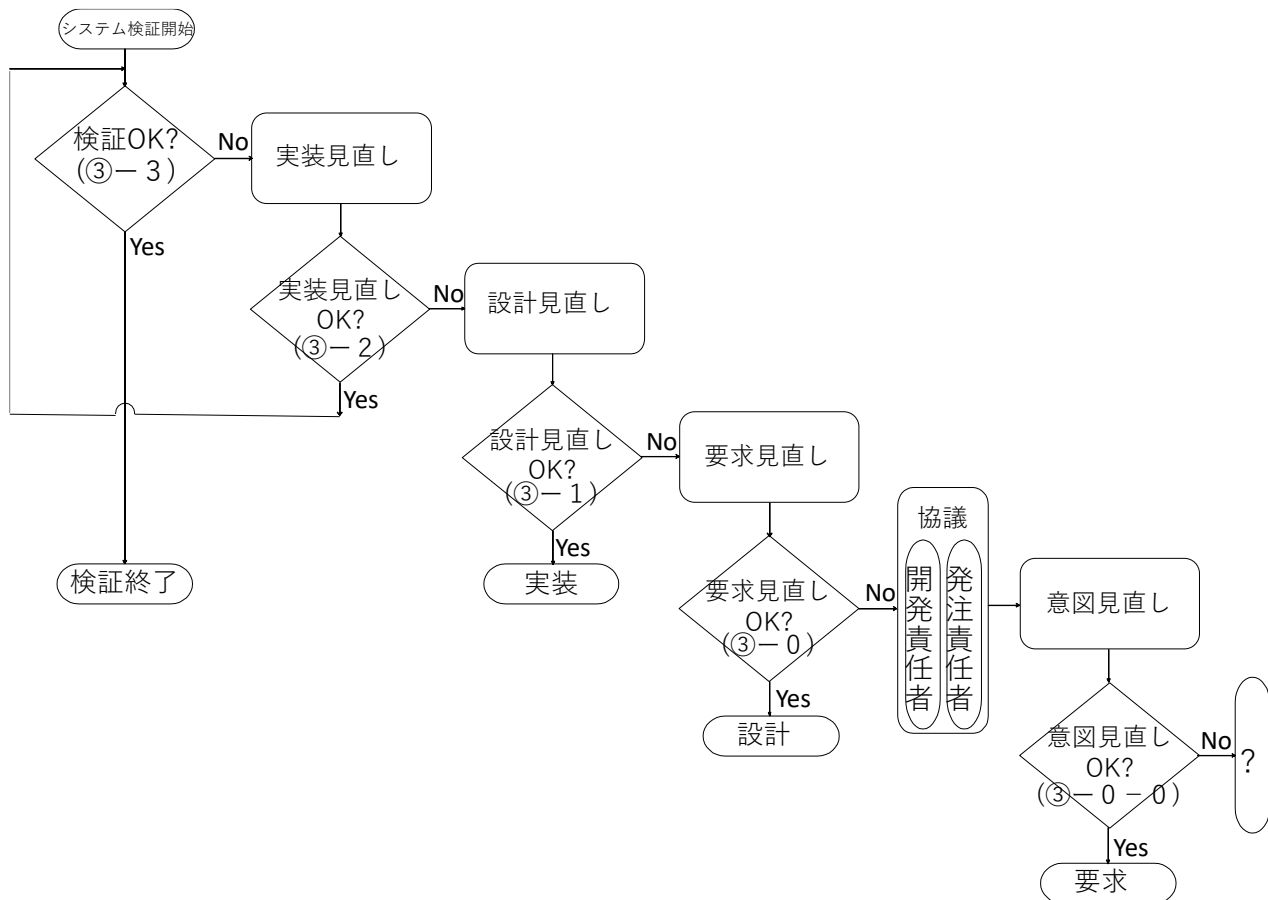


図4 検証工程の前の工程への遡及

であろう。

### 3.2. 意図はもう関係しないと思われる工程の確認

先の図3には設計や実装の工程もサブシステムとして書かれている。図3の段階(4)で意図は不変としたので、このサブシステムには言及しなくてもよさそうだが、意図や要求がどう扱われているのか念のため確認しておく。

設計や実装は、要求の指示通り、であるのが原則であるので、これらの工程で問題が起きたなら、検討は要求工程に遡り、そこで吸収されると考えられる。しかし、検証はどうか。

図4では、先の図3の③「システム OK?」の検証工程を拡大し、工程の遡及を描いている。

単体テストやその上の結合テストをするあたりでは(③-2~③-1)、実装から出発し、設計の見直しあたりまで

われるであろうが、要求を大きく変更する「意図」問題は出てこない。

そして、設計、実装がいずれフィックスし、総合テストに行き、多少の小さな課題は出たにしても、最終的には妥当性の確認で OK がでる。これで本来なら開発は完了である。発注者も納得である。

しかし、図4では要求の見直しにまで遡り(③-0)、発注と開発の両責任者が協議することとなっている。そしてさらにこの図4は、発注者がそもそもこのシステムは、意図を満足していないのではないかと「意図」を持ち出している(③-0-0)。

しかし、そもそも検証とはどういうものであったか。検証の基本は、要求文書に基づいて検証すべき事項を検証仕様として書き、それに基づいて検査を行うことである。

そして、「意図」がここで万が一登場するとしても、意図

は抽象的(内包的)なものであるため、検証者は意図から具体的な検証仕様をつくることはできない。「より速く」とか、「十分速く」といわれても、定量的な数字の提示は(通常の)検証者の責務の範囲外である。

#### 4. 新たな方策

結論からいえば、意図の内包の危うさ、また随意さに対する対応策は、検証工程で量化が求められたように、内包の外延化である。「十分速く」と「一番速く」は、内包の意味は明確に異なる。一番速ければ、十分速いといえようが、十分は速くとも、一番速いとはいえない。しかし、本論の冒頭で発注者は、十分速いという意味は、一番速いことを求めていることだと言う。しかも後には「一番」は「世界一」に変わった。無理を言うが「内包」の融通無碍さである。

このように発注者には内包は自由であるが、それゆえ開発者にはきつい話である、しかし、両者の均衡を取るには、月並みであるが、内包の意味を具体的に合意するしかなかろう。開発者は、内包の融通無碍(発注者には)を、具体性すなわち外延で縛ると強く肝に銘ずべきである。

例えば、「いまだここの会社(A社)のタブレットの起動がもっとも速いのでそれを越えること、しかし、どの会社もしのぎを削っているので、現時点ではA社を越えることを、当該意図の内容とする」。

このように発注者と開発者とは、具体的に取り決めをすべきである。このことは社内においても、である。

しかし、内包には先の事例で述べたように、同じものはずが(「意図」)が変わる。そして、変わった意図の説明が、今までの文脈とは異なる対価などという経済的理由であったりする。意図が要求の上位概念などという場合、論理的な意味ではなく、世俗的な優先度だったりもする。

技術論としては興味の薄れる点かもしれないが、多くの現実のシステムは、技術以外の因子、多くは経済的合理性の中で動いている。従って、意図や内包、すなわち“intention”や“intension”の融通無碍さを開発プロジェクト論として論じてもいいのかもしれない。これらがプロジェクトに様々に影響するから、である。

ときに現場の技術者はこのような経済合理性の議論に無力で、結果的に不合理とも思える時間を強いられる。システム開発が決してスマートなものでないことは世に流布して久しいが、意図や内包の概念を、プロジェクト論としてうまく論ずるものは皆無かもしれない。ときに無理を強いる意図を事前に見抜き、開発の効率を制御したい。

#### 4.1. 入念だが、実の乏しい合意形成かも

表1では、意図を制御できないかと考えている。表1では①‘として意図を要求とは別に取り上げている。

表1 開発プロセスに応じた当事者の合意形成

工程	①			①'			②		③		④			
	依頼	協議	合意	依頼	協議	合意	受け	合意			納品物検査	納品物判定	協議	
発注者	↓	↓	↓	↓	↓	?	○	↓	↓		○	↓	↓	
資料	要求素案		要求原案	意図素案		意図原案	要求文書		正式要求文書		システム	納品結果書		
開発者	○	↑	↑	○	↑	↑	□	↑	○	↑	□	↑	○	
	受け	協議	合意	受け	協議	合意	要求文書作成	提出	協議	合意	システム開発完了	搬入	同意 or 不同意	協議

しかし、先の図3では意図は何も変わっていないながら(発注者も開発者も)、なお最後に発注者は出来上がったシステムは「意図」とは違うと言った。

表1では意図は①‘の工程で明示的に扱われている。意図原案となっている。そして、その意図原案は、次の工程で要求文書に組み込まれているはずである。

発注者と開発者は「依頼」とか「協議」をし、矢印↓は資料等を提示する方向を示し、○は受けとめ、?は不確かだが、本来確かにしなければならぬことを示し、また四角印(□)は自陣での作業を示している。

表1は入念な作業の様子を描き、要求①と意図①‘の工程の後には、工程は要求承認②、開発③、納入検査④と進むことを示している。

表1の作業により、先の図3の(4)では意図は当初(=原案)と変わらないと言ったわけだが、しかし、そう言ったにもかかわらず、図3の(6)ではシステムは意図とは違うと発注者は言ったのであった。

何が変わったか?システムは変わっていないのは事実なので、変わったのは発注者の気持ちである。すなわち、寸前まで(段階(4))変わっていなかった意図が変わったのである。

入念に作業したはずの表1の中に「?」があった。「?」は「不確かだが、本来確かにしなければならぬ」はずのものであるが、確実化していなかったと疑われる。

しかし、意図を確かにすると、そもそもどういうことで

あろうか。それを明確にしなくては、担当者に問いやがらない。

そこがなくては、表1は見かけだけの、合意形成フローに過ぎないかもしれない。

#### 4.2. 「意図」に他の視点を入れる

すでに見た通り、意図は最後に突然意趣変えをするなど「我が儘」になるが、これを少し黙らせる方法はないであろうか。要求は意図に近いので、要求文書の品質を向上させる国際規格は役に立たないだろうか。国際規格 IEEE Std.830 に関する研究資料に考えを借りる<sup>4</sup>。同規格には、要求文書のための8つの品質特性があるとされる(表2参照)。

表2では、表1の要求、意図から開発へと行くプロセスに、これらの要求品質特性を当てはめ、表1の文書がどのような品質特性に気をつけなければならないかを確認している。その際、品質特性を分類し直し、対象とする文書にだけに当てはまる品質特性と、当該文書以外にも当てはまる品質特性とに分けた。Traceable(追跡可能)は他の文書と関係する品質特性である。

また文書内だけのものは、さらに「要求自体が正しいか否か」を問うものと、「要求を第三者的に見る」ものとに分けた<sup>5</sup>。要求の正しさには、要求は Correct(妥当)、Unambiguous(非曖昧)、Complete(完全)、Consistent(無矛盾)が求められ、また第三者的視点では、Ranked for importance and/or stability(重要度/安定度のランク付け)、Verifiable(検証可能)、Modifiable(変更可能)が求められる。

他の文書と関係する「追跡可能」性は、要求事項の追跡関係ばかりではなく、記載履歴、メモのようなものも含んだものである(脚注、4の文書「要求定義で、,,」より)。

よって、意図に戻れば、意図に関する事柄は、記載履歴、メモなど追跡的に管理すればよいと思われる。これを表2では、「◎」の記号で表現している。

先の表1では、「意図原案」、「合意」まではいいとしても、実際に合意形成をしたか否かが「？」となっている。

そこで、ここは国際規格 IEEE Std.830 を援用し、意図

<sup>4</sup> ただし本稿では、「要求定義で困ってませんか？要求仕様書の品質に関する研究成果報告」(要求工学ワーキンググループ 2007年1月24日)より引用。

<sup>5</sup> 特性の数8つは多いので、よりシンプルなクレーピングを試みた。

<sup>6</sup> 「はじめての STAMP/STPA」等。IPA サイト参照。

<sup>7</sup> 事務業務に対する STAMP/STPA の適用は、シス

についても文書化を義務とし、意図が納入物になるとしてもよいであろう。表ではそれも「◎」で表現している。

また、表2では、STAMP/STPA の考え方を取り入れている。すでに見た通り、意図は当初の合意にもかかわらず、最後に当初とは反することになりかねないからである。制止をきかず、制御の輪を潜り抜け、一種のハザードたりうる<sup>6</sup>。

システム開発全体も制御的な業務フローにもととることができる<sup>7</sup>。任意の作業をよく確認しないで次に進むとハザードの原因となることがある。意図はそのなかで、つい「わかった」ぐらいの確認で前に進みがちである。

また表2では STAMP を模し、「与えられた場合のハザード」を「文書が与えられた場合」としている。対比的に「文書が与えられない場合のハザード」も存在するわけである。この「与えられない」場合は、要求仕様書が無くても現場が四苦八苦する際のハザードを思い浮かべただけでよい。今回は議論を割愛した(斜線)。

表2では、「与えられた場合のハザード」に関して、文書の品質特性ごとに「・」で示し、登場人物が品質特性の確認をなすべきことを示している。STAMP では、さらに「早すぎ、遅すぎ、誤順序」、また「早すぎる停止、長すぎる適用」の視点がある。この時間や順序性の視点を表2に適用すれば、任意の品質特性の検討の遅れや検討順序のミスが他の品質特性に悪影響を及ぼすなどはすぐに理解できる。ただし、表が複雑になるのでこれらの視点は割愛した。

以上のように要求の品質特性や、業務制御の考え方を適用することで、開発業務がより周到に管理されるであろうことが理解できる。

テム開発に対する適用の可能性を示唆する。

「STAMP/STPA～業務系システムへの応用検討～」,  
(IPA/SEC「システム安全性・信頼性分析手法 WG」委員、日本電気株式会社、向山輝)

chrome-  
extension://oemmnndcbldboiebnladdacbfmadadm/https://  
www.ipa.go.jp/files/000056599.pdf

## 5. おわりに

本論では、冒頭で厄介者の意図を取り上げ、通常の開発プロセスをたどり、開発に関わる登場人物たちを思い浮かべながら、意図の様相を探った。そこでは意図は時折顔をのぞかせ、意図は変わっていないと言ってみた

したが、しかし他方、もし万が一意図した以上のシステムを作ったとしたならば、中には誰かはが素晴らしいと賞賛するであろう。

前もって持っていた内包概念(intension)の中に、これあれというような具体的なイメージがあったとして、その具体的なイメージにそぐわないときは否定で、具体的なイメ

表2 要求文書のリスク管理と「意図」の位置づけ

工程	順番	登場人物	制御行為	文書が与えられない場合のハザード	文書が与えられた場合のハザード								
					要求を当該文書内だけで扱う特性							要求を当該文書外(成果物,関連文書)と関係づける特性	
					要求自体が正しいか否かの特性				要求を第三者的に見る特性				
					Correct 妥当性	Un-ambiguos 非曖昧	Complete 完全	Consistent 無矛盾	Ranked for importance and/or stability 重要度/安定度のランク付け	Verifiable 検証可能	Modifiable 変更可能		Traceable 追跡可能
①	1	発注者	発注依頼	.	.	.	.	.	.	.	.	.	
		開発者	依頼受け		.	.	.	.	.	.	.		
	2	両者	協議検討		.	.	.	.	.	.	.	.	
			要求概要作成		.	.	.	.	.	.	.	.	
3		合意	.	.	.	.	.	.	.	.			
①'	1	発注者	意図提示	.	.	.	.	.	.	.	.	.	
		開発者	意図検討	.	.	.	.	.	.	.	.		
	2	両者	協議検討	.	.	.	.	.	.	.	.	.	
			<b>意図文書成</b>	.	.	.	.	.	.	.	◎		
3		合意	.	.	.	.	.	.	.	.	.		
②	1	開発者	要求文書作成	.	.	.	.	.	.	.	.	.	
	2	開発者	要求文書提出	.	.	.	.	.	.	.	.	.	
	3	発注者	要求文書受け	.	.	.	.	.	.	.	.	.	
	4	両者	協議検討	.	.	.	.	.	.	.	.	.	.
			要求文書合意	.	.	.	.	.	.	.	.	.	
③	1	開発者	システム開発	.	.	.	.	.	.	.	.	.	
	2		開発完了	.	.	.	.	.	.	.	.	.	
	3		搬入	.	.	.	.	.	.	.	.	.	
④	1	発注者	納入物検査	.	.	.	.	.	.	.	.	.	
	2	発注者	納品物判定	.	.	.	.	.	.	.	.	◎	
	3	両者	同意or不同意	.	.	.	.	.	.	.	.	◎	

り、最後にはシステムは意図とは違っていると言ったりした。意図は、対価にも言及し、払う、払わないと、機能以外の話を持ち出したりする。

そこで要求の品質特性や、業務制御の考え方を適用し、開発プロセスをより周到に回すことを考えた。しかし、このようなことで、意図は本当に矛をおさめてくれるだろうか。

答えはノーである。理由は、意図は要求の上位に位置する概念だからである。あるいは、意図を要求では御しきれないからである。

出来上がったシステムを見て、くだんの人は違うと否定

一をまったく超えた、しかしそれが同じ内包概念の範疇のものであることを気づかされたとき、それが求めているもの、賞賛へとつながる。

この賞賛とは逆に、くだんの人が、意図が同じと言ったそのあとに、実際のシステムを見て意図と違うと言ったのは、好意に理解すれば、イメージにそぐわなかったに過ぎない。氏は何ら矛盾してない。

意図が要求の上位概念という意味は、このように自由な裁量権を要求以上に多く持っているからといえる。また意図は意味を膨らませることができる。あるいは、要求を増やすことができる。意図が内包だとすれば、要求は具体



的な外延といえる。内包と外延には上下関係があるわけではないが、内包は定性的に物事を把握するゆえ、少ない言葉でより多くの領域(世界)に言及できる。本論の文脈で、意図を上位とする理由である。

繰り返すが、この自由な上位の概念を制御する方法は、対としての外延で合意を取り付ける他はなからう。内包(意図)が大きく膨らんでもそれは自由であり、しかし契約は外延的な具体的なもので行うことしか、内包を納得させることはできない。表1の意図原案の合意が「？」であったのは、意図を外延化することが合意であることを知らなかったからである。よって、問うことも問われることもなく、したがって、どうしたらよいかわからなかったのである。

くだんの氏の、「より速く」とか、「世界一」とか、この定性的な用語も内包の用語である。この内包的用語のどれを選ぶか、それこそ意図を問わないといけない。「営業マンが云々」とあるが、どうもこの辺に真意がありそうだ。さてそれで、外延化は？「顧客が大きな、身の丈に合わない意図を持っている」などと否定的なことを言うてはならない。真剣に耳をそばだて、寄り添い、そうしなければ、意図の本当の声は聞こえない。そして、外延化できない。

こうすれば、先に「入念だが、実の乏しい合意形成かも」と、その効果を疑った開発プロセスも生きてくる。意図を制御し、開発プロジェクトが魅力を取り戻す。

以上、意図について試論した。「意図したシステム」などというように、意図はシステム全体をくくる。「要求したシステム」というより、そのくくりが強い。システムをくくる「意図」は、システムの発注者の意思と強く関係する。スポンサーであったりする。あるいは、ああでもない、こうでもないで試行錯誤するアーティストかもしれない。ゲームやプロジェクトマッピングなどヴィジュアル系の世界は、要求文により作られる世界ではなく、表現者の意思、意図で作られる。そのような世界というべきであろう。そして、アートではない自動運転なども、その走行の評価・調整は、感覚的なものであり、最後はなんらかの「意図」で決まる。もしかしたら、決めた本人たちもその実、なぜそうなのか説明ができないかもしれない。もちろん、安全規格はクリアしてるが、例えばデジタル的な処理が幾重にも重なった結果生まれた乗り心地感。この決定は「意図」ではあるまいか。

システムの周辺には「意図」がひしめき、システムと相互作用をしている。システムと意図の関係をひもとき、納得感を得たいものである。

## 参考文献

- [1] 『人月の神話』、フレデリック・ブルックス、1975、20周年記念版;1995

「意図」は氏の説く、システムのコンセプトの完全性と通じているかもしれない。つまり、「意図」はシステム全体の整合性を無意識のうちに求めているというような擬人的特性を持つ。

本論では本書の引用はしていないが、システム開発者の意思、「意図」を本書は随所に感じさせ、本論のヒントになったので、参考文献とした。

- [2] *Integration of Two Kinds of Syntax for Requirements Description and Its Future Development*, Norihiro Urushibara ; Chiharu Sasaki, 2018, IEEE This

当該論文では、システムのための要求記述が整合性を求め、かつこの要求は、設計、実装の各工程の成果物に概念の「連続性」とでもいうべきものを求めると主張する。

この求めは、システムという人工物を作る開発者たちの「意図」の居場所を求めていることではないかと本論を書きつつ、ここでも意図の擬人化を連想した。当該論文は本論では引用していないが、本論執筆の誘導をしてくれたので、参考文献とする。