

用語知識再利用による要求分析支援手法 (II)

2R-2

田中 誠一郎 安田 剛 山田 淳 増沢 香
株式会社 東芝 システム・ソフトウェア技術研究所

1. はじめに

要求分析作業において、ノウハウを蓄積、再利用しながら客先からの要求を分析し、定義していく要求分析手法を提案する。このとき、より誤りが少なく高品質な要求仕様を得るため、得られた要求仕様を形式的要求モデルに変換し、そのモデル上で検証・評価を行い、その結果を要求分析にフィードバックしている。この手法のため要求分析支援ツール JOKER (Japanese-Oriented Knowledge Engineering/Environment for software Reuse) を試作した^[1]。

本稿では、形式的日本語で記述された要求仕様を形式的要求モデルに変換する過程と得られたモデル上での検証・評価手法について述べる。

2. 要求仕様の高品質化

ソフトウェア開発の最初の工程である要求分析とは、SE (System Engineer) が客先の真の要求を明らかにすることである。客先からの要求は、論理的にあいまいであったり、誤りがあることが多い。そこで要求を分析し、定義するためにはSEと客先との間にコミュニケーションを可能とする共通の表現形式を用意する必要がある。多くの場合、日常使用する自然言語で要求仕様を記述することになる。

得られた要求仕様に誤りがあったり、また、記述が詳細でなかった場合、後続の開発工程でこれらのことが問題となり、早期に発見されないと修正に膨大な時間と経費を費やすことになる。このため、誤りのない要求を定義することが重要となってくる。本手法では、得られた要求仕様を検証・評価することにより高品質化することを目的の1つとしている。

本手法では、SEはまず過去に用語知識として蓄積したノウハウを再利用しながら客先とのコミュニケーションをとることにより要求を分析する。得られた要求を形式的日本語で定義すると、要求はJOKERによりビジュアルな形式的要求モデルに変換される。そのモデル上で検証、評価を行い、その結果を要求分析にフィードバックし、段階的に要求を詳細化していく。この過程を繰り返すことにより、高品質な要求仕様を得ることができる。

3. 要求仕様から形式的要求モデル (DFD) への変換

日本語で書かれた要求仕様は客先が理解しやすいが、あ

いまい・冗長であるなどの問題がある。また、仕様に抜けや誤りがないかを検証するのにもあまり適しているとは言えない。そこでJOKERでは、形式的日本語で記述された要求仕様を形式的要求モデルに変換することにより明確化し、モデル上での検証・評価を可能としている。

要求仕様には機能に関する機能要求仕様と、それ以外の属性要求仕様の2つに分類することができるが、ここでは機能要求仕様を分析する。これは機能要求がこれから開発するシステムの主要な骨格となるからである。

形式的要求モデルとしては、データフローモデルを採用し、視覚的に理解しやすいようにデータフローダイアグラム (以後、DFDと略す) として表示する。DFDは、最近普及してきているCASEツールにおいて要求定義の手法として広く採用されているので、より下流の開発工程へのインターフェイスとして有効である。

形式的日本語で記述された要求文からDFDへ変換するルールを図1に示す簡単な例を用いて以下に述べる。

- 1) 各文の動詞と名詞を判断し、動詞をプロセス、名詞をデータフロー、データストアおよび外部エンティティとする。名詞におけるこれらの分類は、名詞辞書にある情報により区別する。
- 2) 動詞辞書が持っている格助詞の情報と文章中の格助詞からデータが入力か出力かを判断する。
- 3) 日本語-DFD変換ルールにより、各文を個別にDFDに変換する。
- 4) DFD結合ルールにより、各プロセス名、データ名で同一なものを結合して全体を合成する。

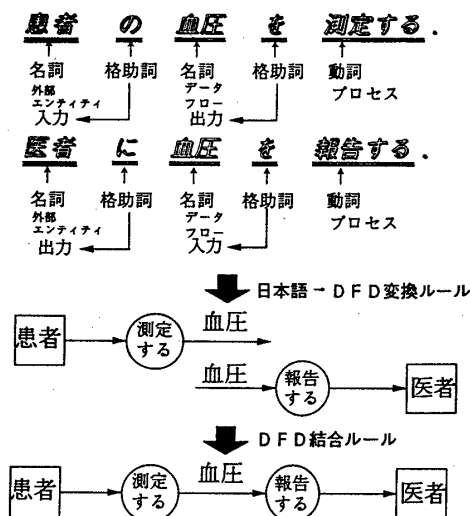


図1. 変換過程

"Requirements Analysis Method by Reusing Word Specification Knowledge (II)" by S. Tanaka, T. Yasuda, A. Yamada, K. Masuzawa, K. Matsumura (Systems & Software Engineering Laboratory, TOSHIBA Corporation)

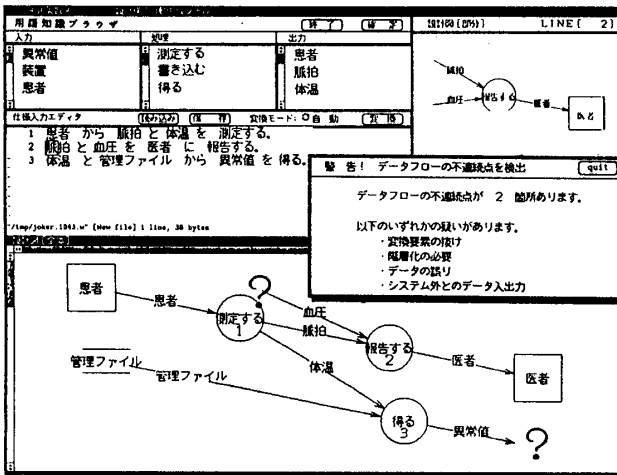


図2. 不連続点の検出

4. 検証および評価手法

本手法では、以下の項目について要求仕様を検証・評価することにより、信頼性、機能性、保守性の向上を図る。

- (1)完全度 : 要求仕様の記述に誤りや抜けのない度合い
- (2)複雑度 : 処理およびデータの流れの複雑さの度合い
- (3)再利用度 : 過去に蓄積した知識を再利用した度合い

4.1 検証手法 データフローモデル上での検証手法はいくつか紹介されている^[2]。ここでは上記(1)完全度についての検証を考える。これは次のような検証法が含まれている。

①不連続点の検出 形式的日本語で記述された要求仕様から変換されたDFD上の不連続点を探すことによって要求仕様の抜けを検出する(図2)。ここで不連続点とは次の2つである。

- ・出力もとのないデータフロー
- ・出力先のないデータフロー

この2つのいずれかが存在したとき、要求仕様に記述されている処理の流れに抜けが存在することになる。

②データアクセス手順の矛盾の検出 SEが自分の頭の中に持っているイメージと生成されたDFDとの間に矛盾がないかを調べるために、DFD上の指定した区間内のデータの生成される様をビジュアルに表示し、それをSEが確認する。また、予めデータの生成順序をアサーションとして記述しておくことにより、これを機械的に調べることも可能となる。生成順序は、「データAは、データBの後に生成される。」のように形式的日本語で記述する。

③制約条件の成立の確認 要求仕様に記述されている制約条件が成立するか確認しながらDFD上のデータの流れを追っていくことにより、プロセスの順序の誤りおよびプロセスそのものの抜けがないか調べる。制約条件は、要求文中で次の下線部のように記述される。「条件CがDのとき、プロセスを実行する。」

4.2 評価手法 ここでは上記(2)複雑度および(3)再利用度についての評価を考える。評価するための尺度(メ

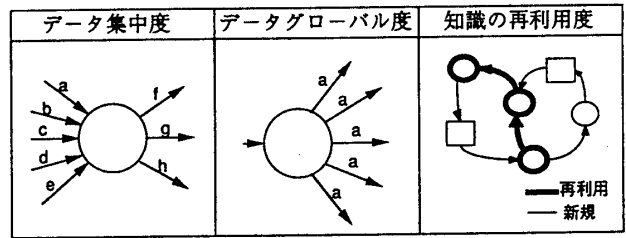


図3. 仕様メトリクス

リクス)としては、データ集中度、データグローバル度および知識の再利用度などの仕様メトリクスを設定した(図3)。

①データ集中度の計測 各プロセスにおける入力データ数と出力データ数を計測し、データの集中の度合いを調べる。これによりデータの集中度の高いプロセスについてはより単純なプロセスの集合と考え、機能を分割することをSEに促す。このとき、入出力データの関係を蓄積されている知識と比較し、より詳細なプロセスの候補を提示する。

②データグローバル度の計測 各データの出力先プロセス数を調べることにより、同一のデータがシステム全体においてどのように使われているかを把握し、システム全体への影響を計測する。また、この情報を機能の階層化の参考とする。

③知識再利用度の計測 新たに定義した要求仕様に過去に蓄積した知識がどの程度利用されているかを計測する。今、蓄積されている知識が十分な経験者によって設計されたものから得られたとしたとき、再利用率の高い要求仕様ほど信頼性が高いと考えられる。また、要求分析以降の開発過程において過去のプロジェクトの成果物の再利用の可能性も高いと考えられる。

以上のように検証、評価を行った結果を要求分析にフィードバックすることにより、詳細で誤りの少ない要求仕様を得ることができる。

5. おわりに

本稿では、要求分析段階において要求仕様を形式的要求モデル(DFD)に変換する方法とモデル上での検証・評価方法について述べた。これにより、詳細で誤りの少ない要求定義が可能となる。また、要求分析以後の開発工程からの後戻りも減少するものと考えられる。

現在、本ツールのプロトタイプが完成したので、今後これを使って評価を行い、改良を行っていく予定である。

参考文献

- [1]安田ほか:「用語知識再利用による要求分析支援手法(1)」, 情報処理学会第42回全国大会予稿集
- [2]情報処理学会:情報処理ハンドブック, pp972-973, 1989