

# データフローダイアグラムを利用した 要求仕様の品質向上技術

## Quality Improvement Technology for Requirement Specifications Using Data Flow Diagrams

村田 由香里

弓倉 陽介

和田 大輝

■ MURATA Yukari

■ YUMIKURA Yosuke

■ WADA Taiki

システムの開発を成功させるためには、そのシステムが何を実現するべきかを記述する、要求仕様の品質を高めることが非常に重要である。特に、曖昧性及び抜けや漏れのない厳密な要求仕様の記述することが強く求められる。一方、数理的な表現を用いて要求仕様を形式的に記述する形式手法が注目を集めている。しかし、大規模なシステムの要求仕様を全て形式的に記述する作業は、非常に膨大となる。

東芝は、計算機を用いて自然言語で記述された要求仕様を解析し、データの流れ（データフロー）に着目した図式表現に自動変換することで、要求仕様の品質を向上させる技術を開発した。生成された図式表現を機械的に評価することで、要求仕様の曖昧性及び抜けや漏れを検出することが可能になる。

To successfully perform system development, the quality of the system requirement specifications is of critical importance. In particular, it is essential to completely eliminate ambiguities, omissions, and leaks in the requirement specifications. Formal methods using mathematical notation have recently been attracting attention as a means of describing requirement specifications. However, this approach requires a great deal of time and effort when applied to large-scale systems.

As a solution to this issue, Toshiba has developed a technology for improvement of the quality of requirement specifications based on natural language processing that analyzes the phrase structures of requirement specifications and automatically generates graphic expressions representing data flow diagrams (DFDs). Mechanical evaluation of the graphic expressions generated by this technology makes it possible to detect ambiguities, omissions, and leaks in the requirement specifications.

## 1 まえがき

システムの開発を成功させるためには、そのシステムが何を実現するべきかを記述する要求仕様の品質を高めることが非常に重要である。システムの大規模化や複雑化に伴い、曖昧性及び抜けや漏れのない厳密な要求仕様の記述は、産業界において共通の重要課題となっている<sup>(1)</sup>。

一方、集合論や述語論理などの数理的な表現を用いて要求仕様を記述し検証する、形式手法<sup>(2)</sup>が注目を集めている。形式手法を用いて厳密に要求仕様を記述する過程で、要求仕様の曖昧性及び抜けや漏れを発見する効果も報告されている<sup>(3)</sup>。しかし、大規模なシステムの要求仕様を全て形式的に記述する作業は、非常に膨大である。

そこで、東芝は、自然言語で記述された要求仕様を計算機を用いて解析し、データフローに着目した図式表現に自動変換することで、要求仕様の品質を向上させる技術を開発した。生成された図式表現を機械的に評価することで、要求仕様の曖昧性及び抜けや漏れを検出することが可能になる。

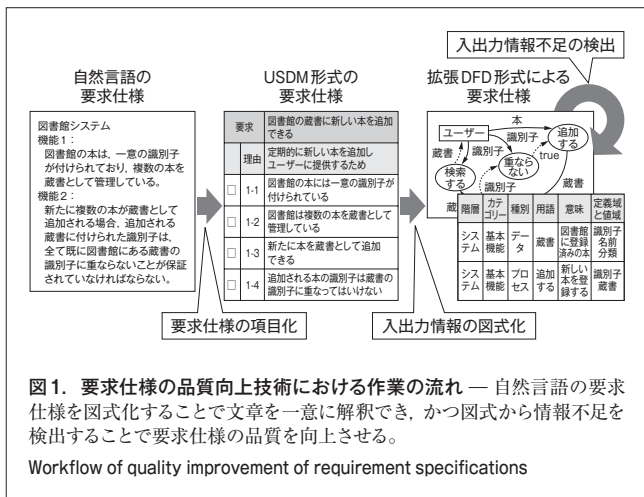
ここでは、この技術の概要と特長、作業の流れ、及び要求仕様の記述に適用し評価した結果について述べる。

## 2 要求仕様の品質向上技術の概要と特長

要求仕様の記述においては、まず、システムが持つべき機能を明確にする。システムが持つべき機能として、システム外部から受け取る入力データ、入力データに対する処理、及び処理の結果としてシステムが外部に提供する出力データを明確にする必要がある。

システムの機能と関連するデータを表す表記法として、データフローダイアグラム (DFD: Data Flow Diagram) がある。当社は今回、DFDの表記法を利用して、自然言語で記述した要求仕様の用語の関係性を図式表現する拡張DFDを考案し、要求仕様を明確化する方法を開発した。開発した要求仕様の品質向上技術における作業の流れを図1に示す。

この技術では、要求仕様の図式表現は文単位で行う。そのため、自然言語の要求仕様を文単位で項目化したUSDM (Universal Specification Description Manner) 形式<sup>(4)</sup>を図式表現の対象とする。USDM形式の要求仕様の文中に現れる主語や、述語、目的語などの関係を自然言語解析により抽出し、入出力情報と処理で表される拡張DFDとして自動的に図式表現する。拡張DFDで表すことによって、システムに必要なデータと処理の関係性を可視化し、例えば入出力情報の不



足を見つけることで、要求仕様の曖昧性及び抜けや漏れを検出できる。

### 3 拡張DFDによる要求仕様の厳密化

ここでは、要求仕様から拡張DFDへの変換方法と、変換した拡張DFDにより要求仕様の曖昧性及び抜けや漏れをどのように検出するのか、について述べる。3章及び4章では、NPO法人 組込みソフトウェア管理者・技術者育成研究会 (SESSAME) が作成した、代表的なUSDM形式の電気ポット要求仕様<sup>5)</sup>から引用したいくつかの文例を使う。

#### 3.1 拡張DFDへの変換

この技術では、自然言語の要求仕様を拡張DFDに変換するために、次の四つのステップを踏む。

##### ステップ1 自然言語解析による要求仕様情報の抽出

自然言語解析を用い、要求仕様情報として出現語情報、文構造情報、及びモダリティ情報の三つの情報を抽出する(図2)。出現語情報は、要求仕様に現れる単語の情報を抽出したものである。文構造情報は、文における主語と述語、あるいは形容詞と名詞など係りと受けの関係を解析した結果である。モダリティ情報<sup>6)</sup>は、文に特定の意味を付加する単語を定めたモダリティ定義表(表1)に従い、出現語情報から該当する単語を抽出したものである。

この技術では、要求仕様においてデータの流れを記述するために、重要な単語をモダリティ情報として定義した。例えば、“水量が異常な場合”における単語の区切りは“水量|が|異常|な|場合”となるが、“場合”という単語は、“水量が異常だ”という単文に対して条件を表す意味を付け加える。このように、モダリティ情報によって単文に特定の意味を付加できる。

##### ステップ2 要求仕様情報から拡張DFDの生成 ステッ

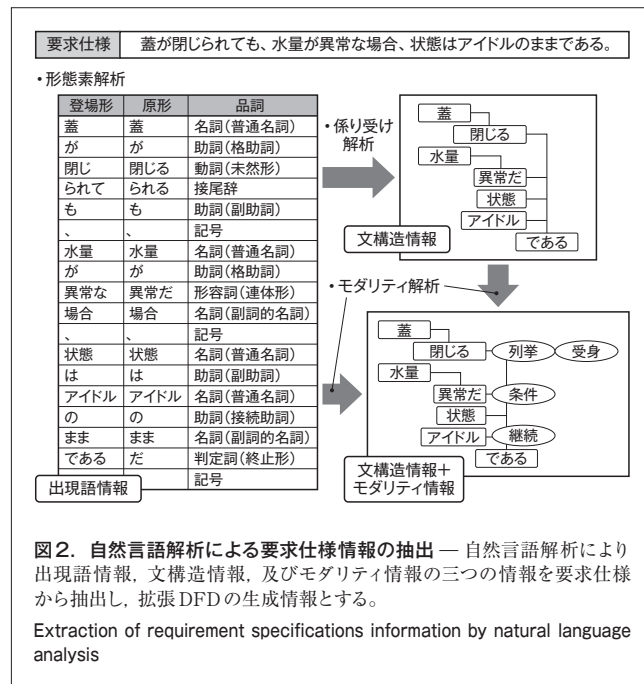


表1. モダリティ定義表

Modality definition table

モダリティ情報	品詞: 原形
列 挙	副助詞: も
条 件	名詞: 場合
	動詞 (仮定形)
限 定	係助詞: のみ
	係助詞: しか
.....	.....

プ1で取り出した出現語情報と文構造情報から拡張DFDを生成する。拡張DFDの生成では、述語を一つだけ含む単文の範囲を文構造情報から抽出し、述語と他の単語との格関係により単文単位で拡張DFDを生成する。

図3に生成ルールの例を示す。例えば“タイマから経過時間を取得する”という場合、述語の“取得する”が処理(以下、プロセスと呼ぶ)に、カラ格としてつながる“タイマ”がデータの発生元(以下、データストアと呼ぶ)に、及びラ格としてつながる“経過時間”がデータフローに相当する拡張DFDに変換する。

##### ステップ3 モダリティを利用した拡張DFD構造の変形

ステップ2で生成した拡張DFDに対して、モダリティ情報の持つ意味を付加するために、プロセスの追加やデータフローの変形を行う。変形ルールの一部を図4に示す。

図4の条件モダリティの例では、“蓋センサが3 sec以上onになる場合に”という単文について、変更前は“場合に”を考慮していないため、拡張DFDでは“蓋センサ”というデータストアに“3 sec以上on”というデータが入力さ

述語と格関係		拡張DFD	拡張DFD変換例
単文構造	拡張DFD		
述語+単語1 "カラ格"+単語2 "ヲ格"			タイムから経過時間を取得する
単語1 カラ格 単語2 ヲ格 述語	単語2 単語1 → 単語1を述語	経過時間 タイム → 経過時間を取得する	
述語+単語1 "ヲ格"+単語2 "ニ格"			表示窓に経過時間を表示する
単語1 ヲ格 単語2 ニ格 表示する	単語2を述語 → 単語1 単語2	経過時間 表示窓に経過時間を表示する → 表示窓	

図3. 拡張DFD生成ルールの例 — 要求仕様情報のうち主に文構造情報を基に、述語と用語の格関係から拡張DFDを生成するための、変換ルールの一部である。

Examples of rules for generating extended DFD

モダリティ種別	拡張DFD構造変形例	
	変形前	変形後
列挙	蓋を閉じて"も"	
条件	蓋センサが3 sec以上onになる"場合に"	

図4. モダリティ情報による変形ルールの例 — 図3で作成した拡張DFDを図2で抽出したモダリティ情報を基に変形することで、データの流れを正しく表すようにできるルールの一部である。

Examples of modification rules using modality information

れるという構造になる。表1の条件モダリティ“名詞：場合”を考慮すると、条件を表す単文という解釈が加わり、“蓋センサ”というデータストアから“3 sec以上on”という条件を表すデータが流れるという形になる。要求仕様としては、蓋センサの条件を意図している単文のため、この変形により適切な拡張DFDを得ることができる。

**ステップ4 拡張DFD構造の統合** ステップ1～ステップ3で作成した単文構造の拡張DFDを結合して、複文を表す拡張DFDにする。述語を複数持つ複文や重文では、文構造情報で述語間に係りと受けの関係が現れる。このような場合、単文の拡張DFDにおける述語に対応するプロセス間をデータフローでつなぐ。更に、同じデータストアやプロセスの要素どうしを統合する。

### 3.2 拡張DFDの評価

ここでは、自然言語の要求仕様から生成した拡張DFDを次の三つの観点から評価することで、要求仕様の曖昧性及び抜けや漏れを検出できることを述べる。

(1) 拡張DFDにおける無名の要素の存在 拡張DFDは、述語に対してガ格やヲ格にあたる単語が不足してい

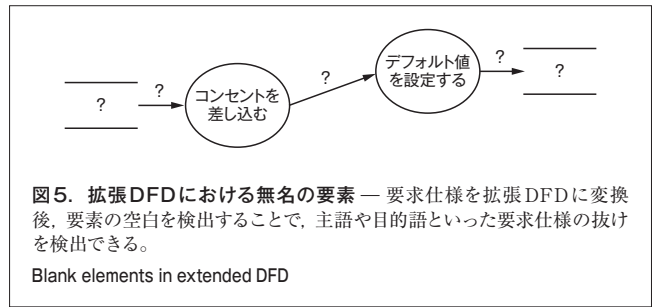


図5. 拡張DFDにおける無名の要素 — 要求仕様を拡張DFDに変換後、要素の空白を検出することで、主語や目的語といった要求仕様の抜けを検出できる。

Blank elements in extended DFD

る場合に、その部分を無名の要素として生成している。したがって、無名の要素がある場合は、述語に対して主語や目的語が不足していることを表す。図5では、“コンセントを差し込む”というプロセスに対してどのようなデータが入力されるのか、どのようなデータを“デフォルト値として設定する”のか、が記述されていないため、無名の要素となっている。

(2) 拡張DFD間の断裂 要求仕様から拡張DFDを生成するとき、データや処理に関係があれば、拡張DFDの要素間はデータフローによって接続される。拡張DFDの断裂は、拡張DFDの要素間にまったく関係性がない場合に限られる。図6に示すように、一部分が断裂した拡張DFDが生成された場合は、要求仕様に曖昧性及び抜けや漏れがあり断裂している可能性が高い。断裂した要素を結合するための情報を検討することで、曖昧性及び抜けや漏れを修正できる。

(3) 条件の組合せ抜け 要求仕様を拡張DFDに変換する際に、条件モダリティに着目することで、データストアとプロセスに流れるデータフローから、プロセスを実行するための条件を抽出し、条件の組合せ漏れがないかを検出する。図7の拡張DFDにおいて、AとBのデータストア、CとEのプロセスに流れるデータフローの条件の組合せから、“Aが100℃以上であり、Bがonである場合”、また“Aが10℃以上であり、Bがoffである場合”の条件の組合せに対する要求仕様が記述されていないことがわかる。記述されていない条件に対して要求仕様が必要かどうかは、人が判断して記述する。この場合は、“Aが100℃以

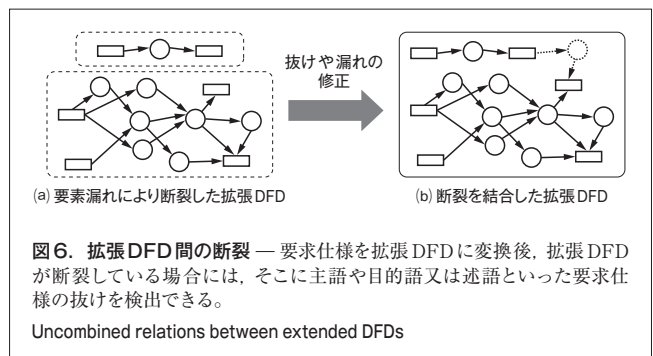
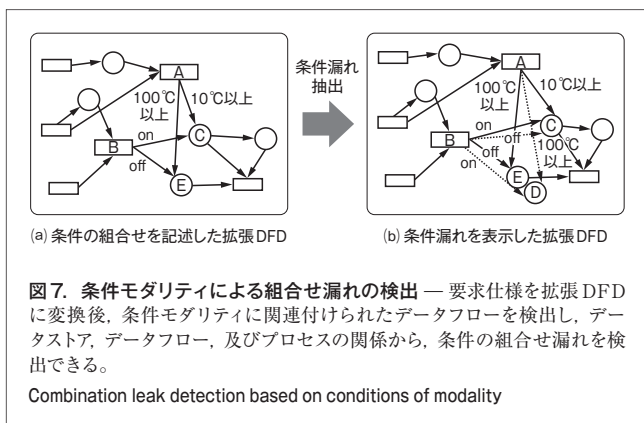


図6. 拡張DFD間の断裂 — 要求仕様を拡張DFDに変換後、拡張DFDが断裂している場合には、そこに主語や目的語又は述語といった要求仕様の抜けを検出できる。

Uncombined relations between extended DFDs



上であり、Bがonである場合”にDというプロセスが抜け

#### 4 要求仕様記述への適用評価

文献(5)に述べられた47項目の要求仕様に対して、この技術を適用した結果を表2に示す。無名の要素や拡張DFDの断裂による仕様の不備を検出できた。また、条件漏れも検出しているが、条件漏れに対して実際に仕様を修正するか否かは要求仕様記述者の判断となる。

検出例として、モダリティ情報に基づいた無名の要素の検出について以下に述べる。

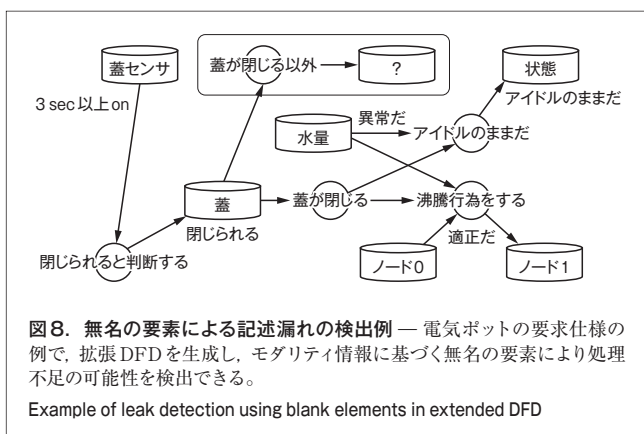
図8は、“蓋が閉じられても、水量が異常な場合、状態はアイドルのままである”という仕様に対して、蓋が閉じられて“も”

表2. 要求仕様の評価結果

Results of evaluation of requirement specifications using extended DFD

計測数	自動生成の数	仕様不備の数
無名の要素数	32	19
DFDの断裂	4	4
条件漏れ	225	-*

\*要求仕様記述者の判断によるため不定



の列挙モダリティから、“蓋が閉じる以外”のプロセス及びそれに付随する無名のデータストアを暗黙知として新たに追加している。このように、モダリティ情報により追加されたプロセスやデータストアを検出することで、要求仕様の抜けや漏れを検出できる。

#### 5 あとがき

計算機を用いて自然言語で記述された要求仕様を解析し、データの流りに着目した図式表現に自動変換することで、要求仕様の品質を向上させる技術について述べた。生成された図式表現を機械的に評価することで、要求仕様の曖昧性及び抜けや漏れを検出することが可能になる。

今後は、拡張DFDを生成する前の、USDM形式による要求仕様の記述段階で機械的に曖昧性及び抜けや漏れを排除し、単文単位での要求仕様の品質を向上させる技術を検討する。更に、そのうえで拡張DFDを生成し、人によるレビューでは検出しにくい、文章全体を通した曖昧性及び抜けや漏れの検出精度を向上させていく。

#### 文献

- 情報処理推進機構. “2010年版組込みソフトウェア産業実態調査報告書”. <<https://www.ipa.go.jp/sec/softwareengineering/reports/20100607.html>>, (参照 2013-10-18).
- フィッツジェラルド, J. 他. ソフトウェア開発のモデル化技法. 東京, 岩波書店, 2003, 306p.
- 情報処理推進機構. “情報系の実稼働システムを対象とした形式手法適用実験報告書”. <<http://www.ipa.go.jp/sec/softwareengineering/reports/20120420.html>>, (参照 2013-10-18).
- 清水吉男. 要求を仕様化する技術 表現する技術 改訂第2版. 東京, 技術評論社, 2010, 384p.
- SESSAME. “話題沸騰ポット (GOMA-1015型) 要求仕様書 第7版”. <[http://www.sesame.jp/workinggroup/WorkingGroup2/POT\\_Specification\\_v7.PDF](http://www.sesame.jp/workinggroup/WorkingGroup2/POT_Specification_v7.PDF)>, (参照 2013-10-18).
- 宮崎和人 他. モダリティ (新日本語文法選書). 東京, くろしお出版, 2002, 325p.



村田 由香里 MURATA Yukari

ソフトウェア技術センター ソフトウェア設計技術開発担当参事。ソフトウェアの要求分析・要求管理技術とソフトウェア上流設計技術の開発、及びソフトウェア開発プロジェクトの支援業務に従事。Corporate Software Engineering Center



弓倉 陽介 YUMIKURA Yosuke

ソフトウェア技術センター ソフトウェア設計技術開発担当。自然言語解析を利用した仕様品質向上とモデルベース設計に関する技術開発に従事。Corporate Software Engineering Center



和田 大輝 WADA Taiki

ソフトウェア技術センター ソフトウェア設計技術開発担当。形式仕様記述とモデル検査を用いた形式手法による仕様品質向上のための技術開発に従事。日本ソフトウェア科学会会員。Corporate Software Engineering Center