

# 製造業向け FTA 支援ソフトウェアの開発と実務への適用

## Development of Computer-Aided Fault Tree Analysis Software for Manufacturing Industry and its Application to Practical Business

古川 慈之<sup>1\*</sup> 澤田 浩之<sup>1</sup> 徳永 仁史<sup>1</sup>

Yoshiyuki Furukawa<sup>1</sup>, Hiroyuki Sawada<sup>1</sup>, and Hitoshi Tokunaga<sup>1</sup>

<sup>1</sup>産業技術総合研究所

<sup>1</sup>National Institute of Advanced Industrial Science and Technology

**Abstract:** This paper describes a case study of software development of computer-aided fault tree analysis system for reliability design in manufacturing industry and its application to practical business. This software development aims at realizing a practical product for the corporate business based on academic research products. This kind of software development usually faces two problems, development cost and usability of the software. We overcome those problems by introducing a software development method based on our software development platform "MZ Platform" and utilizing the iteration process of prototyping and reviewing in a short period.

### 1 はじめに

企業での実務において利用される高度な専用ソフトウェアの開発には、次のような二つの課題が存在する。一つは、ソフトウェアの価格が高額なことである。ソフトウェアの実現する処理や機能が高度であれば、開発に要する人件費や期間、利用する外部ライブラリのライセンス料などの開発コストは高くなる上に、特定の用途に特化していることからそのソフトウェアの市場が小さいことがその理由である。もう一つは、ソフトウェアの操作性である。新たなソフトウェアを導入することは業務プロセスに影響を与え、特にそのソフトウェアを人が操作する場合には、低い操作性が作業者の負担になる。

これらの課題は、企業において業務の知識や技術を継承していくために開発するソフトウェアでも同様である。業務の知識や技術を蓄積して利用するプロセスは企業ごとに異なり、蓄積した知識を活用するために例えば学術的な研究成果をソフトウェア化して導入しようとするれば高度な処理を実装することになる。

本稿では、以上二つの課題を考慮した上で大学での研究成果を企業で利用するためにソフトウェア開発を実施して実務に適用した事例を報告する。対象としたソフトウェアは、製造業における工業製品の高信頼性設計を実現するための Fault Tree Analysis (FTA) 支援ソフトウェアである。ここで扱う FTA とは、設計者が製品に生じる可能性のある故障を網羅的に洗い出し、その原因にあらかじめ対処することで製品の信頼性を確保することを目的に実施される手法である。このような FTA はその実施に知識や経験のみならず客観的な論理性や網羅性が必要なため、その技術の継承と後進の育成および実施のための計算機支援が企業にとって重要な課題である。また、FTA の実施によって作成される故障と原因の関係は、製品に関連するツリー構造の知識として企業の資産となるため、これを有効活用することも望まれている。

なお、本稿の内容はジャトコ株式会社、東京大学、産業技術総合研究所の三者による共同開発[1][2][3]の一部として実施されたものであり、以降ではソフトウェア開発手法と開発したソフトウェアの内容を中心に述べる。

---

\*連絡先：産業技術総合研究所  
〒305-8564 茨城県つくば市並木 1-2-1  
E-mail: y-furukawa@aist.go.jp

## 2 ソフトウェア開発手法

### 2.1 MZ Platform の利用

FTA 支援ソフトウェアの開発には、著者らが開発したソフトウェア基盤の MZ Platform[4][5]を利用した。一般的なソフトウェア開発では、プログラミング言語で記述された動作指示（ソースコード）を書くことでソフトウェアを構築するのに対し、MZ Platform によるソフトウェア開発では、ソフトウェアの部品（コンポーネント）を主にマウス操作によって画面上で組み合わせて動作を定義し、ソフトウェアを構築する（図 1 参照）。MZ Platform を用いるとソースコードを書かずにソフトウェアを作成することができ、ソフトウェアの動作が視覚的に把握できるため、一般的なソフトウェア開発の未経験者でも習得が容易であるという特徴を持つ。この特徴によって、MZ Platform はソフトウェアの使用者（エンドユーザ）が自ら開発を実施する「エンドユーザ開発[6]」の支援を目的に開発されてきた。特に、MZ Platform を開発したプロジェクトは中小製造業の IT 化推進を全体の目的としていたため、ソフトウェア構築の実例としては、データベースと接続して受注や工程情報、進捗情報を共有し、内容をグラフやガントチャートで視覚的に表現して帳票を出力するシステムの例（図 1 右参照）が多い[7]。

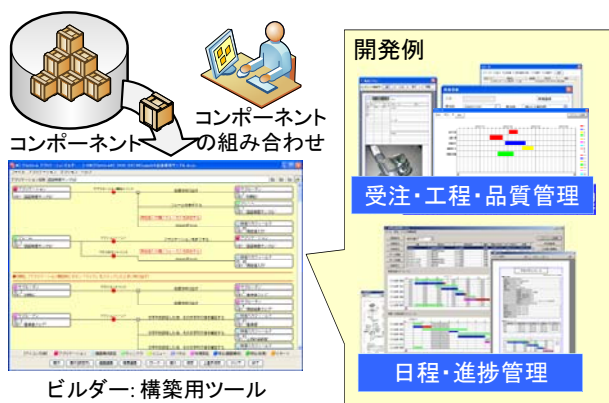


図 1：MZ Platform による開発の概要

本稿で対象とするような大学等の研究成果である何らかの工学的手法をソフトウェア化する場合には、MZ Platform 上で標準的なコンポーネントを組み合わせるだけでは実現が難しい場合が多い。その場合、MZ Platform にはプログラミング言語の一つ Java でテンプレートに基づくソースコードを書くことで新たなコンポーネントを容易に追加できる機能があるため、そのようなソフトウェア開発にも適用するこ

とが可能である（図 2 参照）。一般的なソフトウェア開発の方法と比較すると、ソースコードを書く作業は対象とする工学的手法の実装（専用コンポーネントの作成）のみに限定され、その他の部分は MZ Platform 上で標準的なコンポーネントとの組み合わせで実現できるため、最小限の開発作業で高度な専用システムを構築できるという利点がある。

本稿で報告する事例は、エンドユーザ開発ではなく、MZ Platform によるソフトウェア構築と MZ Platform のコンポーネント作成方法を熟知した著者らが、外部の研究成果について論文を基に工学的手法を実装することで、高度な専用システムの開発を実現したものである。

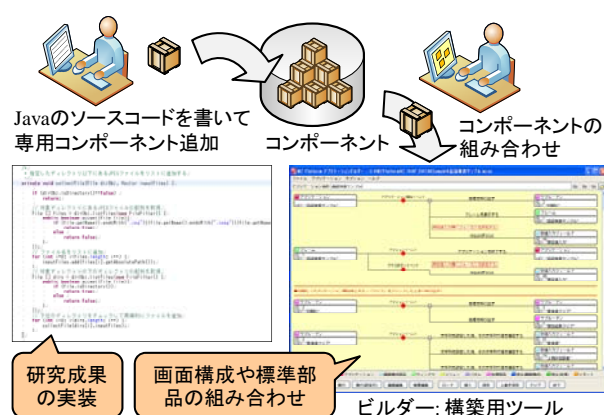


図 2：専用コンポーネント追加による実装

### 2.2 開発プロセスとスケジュール

年	2010			2011		
	4-6	7-9	10-12	1-3	4-6	7-9
現場ヒアリング	→		→			
設計・開発	プロトタイプ作成 → 試用版作成					
				試用版の改善 → リリース版作成		
試用とリリース		試用・フィードバック				リリース

図 3：開発スケジュール

実施した開発スケジュールの概略を図 3 に示す。2010 年 4 月に開発プロジェクトが開始され、約 3 ヶ月で業務内容のヒアリングとプロトタイプ作成による方向性確認、さらに 3 ヶ月後の 2010 年 10 月に試用版をリリース、以降約 3 ヶ月単位で機能追加とフィードバックを反映した修正版をリリースし、2011

年7月に正式版としてリリースした。

図3に示すように、この開発プロジェクトでは比較的短期間でのプロトタイプ作成をはじめとして、試用版の作成とフィードバックのサイクルを複数回繰り返している。このような開発プロセスは、古典的なソフトウェア開発プロセスのウォーターフォール型と対比して、繰り返し型または進化型のプロセスと呼ばれる[8]。この開発プロセスの利点は、企業側の担当者や現場作業員（ユーザ）が、実際に動作するソフトウェアのイメージを早い段階で把握することができ、それが導入された場合の効果や影響について具体的に検討して早期のフィードバックを返せることである。このプロセスを適用することで、高度な専用ソフトウェアの開発を実現しつつ、ユーザにとっての操作性を向上させることを試みた。このプロセスを適用するためには、ソフトウェアの構成を柔軟に修正して短時間で試行と評価を繰り返すための機能がソフトウェア基盤や開発環境に必要なが、MZ Platformはコンポーネントの組み合わせによってその機能を提供しているため、親和性が高いと言える。

### 3 開発したFTA支援ソフトウェア

#### 3.1 FTA支援ソフトウェアの概要

開発したFTA支援ソフトウェア「FTAid」の概要と画面イメージを図4に示す。FTAidは、FTAで用いられるツリー（FT図）の作成を支援するソフトウェアである。FT図作成を支援することで、あり得る故障とその原因を設計段階で事前に洗い出し、対策を施すことで高信頼性設計に貢献することを目的としている。

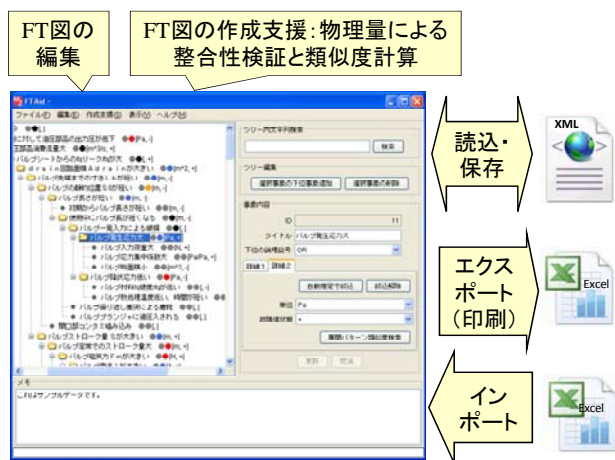


図4：FTAidの概要

導入先の企業では、一般的に用いられるFTA手法と異なるP-FTA[9]と呼ばれる手法を用いており、従来はExcelファイルに文字列と罫線でFT図を記述していた。一般的なFTA手法（例えば[10]）との違いとして、図5に示すように頂上事象を左に配置して右側に下位の事象を配置し、論理記号としてORとANDのみを用いてORは表記を省略することが挙げられる。このFT図の形式に基づき、開発したFTAidでは画面上にツリーを表示してそれを編集する機能と、FT図のデータをXML形式で入出力する機能、後述する物理量に基づく整合性検証と類似度計算を実行する機能、Excelファイルを入出力する機能を持つ。

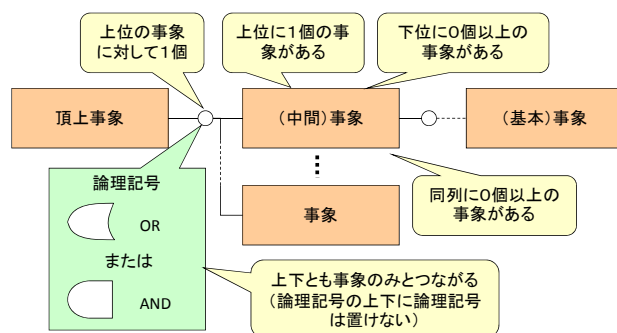


図5：対象とするFT図の構成

#### 3.2 FT図の編集と表示

FTAidの基本機能はFT図の編集と表示である。従来はExcelで作成していたFT図を新しいソフトウェアのFTAidで作成するという業務プロセスの変化が生じるので、現場作業員への配慮から基本的な編集機能に関して少なくともExcelと同等以上の操作性を達成することが必要と考えられた。そのため、FTAidを利用することで作業員にとって有益となる機能を中心に実装を進めた。FT図の表示に関しては、ツリーを直接表示することでExcelでの表示と異なり、必要に応じて部分的に下位の事象を折りたたんで非表示にすることができるため、利便性が向上していると言える。また、FT図の編集に関しては、事象の追加・削除・移動という操作がExcelでは罫線の設定やセルの削除など複数の操作の組み合わせが必要だったが、FTAidでは直観的で必要最小限の操作で実現できる。特に、部分ツリーの移動やコピーペースト操作は、格段に操作性が向上している。一方、この種の専用ソフトウェア構築でありがちなこととして、基本的な操作が煩雑になることや必要手順の多さという問題が入り込みやすいため、それについては十分に配慮して画面配置や操作インターフェースの作りこみを実施した。

### 3.3 物理量次元に基づく FT 図作成支援

FTAid における FTA 支援機能の中心は、FT 図に対する物理量次元に基づく整合性検証と類似度計算である。この手法の概要と実装の概略について述べる。

村上ら[11]は、ある事象が物理量の状態を表現している場合、物理量に対応する単位が必ず存在し、この単位は SI 単位系の 7 基本単位の組み合わせとして分解でき、7 基本単位のべき数からなる 7 次元のベクトルとして表現できることから、これを物理量次元ベクトルと呼んでいる。また、FT 図内の各事象とその直下の下位事象の集合（各階層）に関して物理量次元ベクトルが与えられた場合、上位側の事象（親事象）の物理量次元ベクトルが下位側の各事象（子事象）の物理量次元ベクトルの加減乗除で表現できるかどうかを検証することで、各階層の整合性を検証できることを示した。つまり、この物理量次元ベクトルの導入によって、FT 図の階層ごとに物理量に関する展開の整合性を検証することができ、ユーザによる FT 図の作成を支援できる。さらに、各階層の物理量次元ベクトルに故障値の状態を表す文字列（故障値状態文字列）を加えた情報に対して、類似度を定義している。この類似度計算によって、FT 図における故障と原因の展開パターンや代表的な FT 図の事例データを準備しておくことで、作成中の FT 図の展開が妥当かどうか判断するための類似度検索が可能となり、ユーザによる FT 図の作成を支援できる。

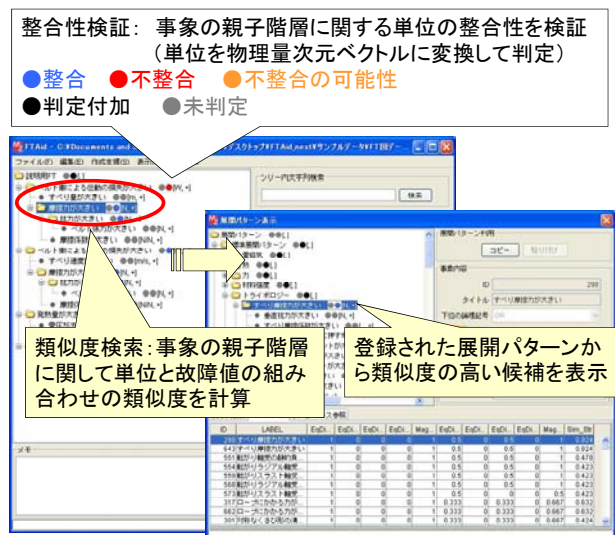


図 6：整合性検証と類似度検索

図 6 に FTAid における整合性検証と類似度検索の画面イメージを示す。これらの機能を実現するため、FT 図の各事象には単位を示す文字列（単位文字列）

と故障値状態文字列の属性を持たせた。単位文字列から物理量次元ベクトルに変換するための単位定義ファイルを外付ファイルとしてソフトウェアに含め、変換は整合性検証と類似度検索の前に実行する。整合性検証では、対象となる階層の物理量次元ベクトルの集合に対し、文献[11]に記載された手法の計算を直接実行する。類似度検索では、展開パターンを外付ファイルとして持たせ、対象となる階層の物理量次元ベクトルと故障値状態文字列の組の集合に関して、展開パターン内の全階層との類似度計算を実行し、類似度の高い順に整理して表示する。ただし、本稿の開発事例では、文献[11]の記述には含まれていない親事象同士の文字列の類似度を整理条件に加えている。これは、物理量次元ベクトルと故障値状態文字列がすべて一致する階層が展開パターン内に複数存在した場合に、文字列まで一致したものを最上位に出現させるための条件追加である。

### 3.4 Excel ファイルの入出力

FTAid の開発では、従来の業務プロセスとの適合性から Excel ファイルの入出力は不可欠な機能であった。過去のデータが Excel ファイルとして多数存在しているため、その資産を有効に活用する必要があったからである。また、FTA を実施する作業者の環境移行の問題からしても、Excel で作成して FTAid で整合性を検証するなどの段階的な移行を可能とする必要があった。さらに、FTA に関連する他の業務プロセスとの関係からすると、Excel で作成した FT 図を前提として作業を実施するものがあるため、従来の書式で Excel ファイルに書き出せる必要もあった。

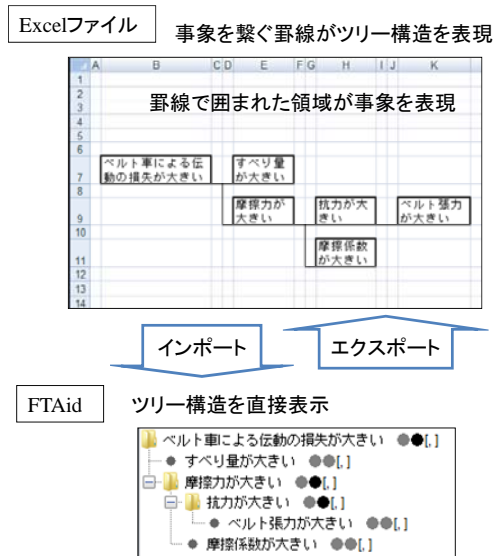


図 7：Excel ファイル入出力



図7に Excel ファイル入出力の概略を示す。Excel ファイルに記述された FT 図は、罫線で囲まれた領域が事象を表現し、事象間を繋ぐ罫線がツリー構造を表現している。FT 図の中で左側が上位で右側が下位を示している。また、前述のように OR 記号は省略されるため、この例では論理記号がすべて省略された状態になっている。

### 3.5 物理量と故障値の自動推定

FTAid では、物理量次元ベクトルに基づく整合性検証と類似度計算を導入したことで、従来の Excel ファイルで表現されていた FT 図のデータに対して、前述の通り事象ごとに新たな属性である単位文字列と故障値状態文字列が追加されている。FTAid を用いて手作業で最初から FT 図を作成する場合、同一のデータを Excel で作成するときと比較して作業者の労力が軽減されたと考えてよいことは既に述べたが、整合性検証と類似度計算という新しい機能を使うために、新たに単位文字列と故障値状態文字列をすべて手作業で入力することは、作業者の労力を増加させることになる。また、Excel ファイルから FT 図を FTAid に入力する場合も、単位文字列と故障値状態文字列の情報は直接含まれていないため、整合性検証と類似度計算の機能を利用するためには、別途入力が必要である。当然ながら、その労力が整合性検証と類似度計算で得られる効果で見合えばそれでもよいと判断される場合もあるが、その負担が原因で整合性検証と類似度計算を使わないという選択を作業者がしないとは限らない。さらには、それらの機能が不要であればそもそも FTAid 自体を使う必要がないと判断される事態がないとは言えない。

このような事態を避けるために、FTAid は事象を表現する文字列から物理量と故障値を表現する文字列を探索し、単位文字列と故障値状態文字列を自動推定する機能を有している。自動推定に必要なデータとして、物理量を表現する用語（物理量用語）と単位文字列の対応を記述したデータと、故障値を表現する用語（故障値用語）と故障値状態文字列の対応を記述したデータをそれぞれ外部ファイル（用語辞書）として作成した。自動推定処理の概略を図8に示す。事象を表現する文字列中に物理量用語と故障値用語が含まれているとして、先に故障値用語を探索して次に物理量用語を探索する。故障値用語を発見した際には、対象となる文字列から故障値用語の箇所を削除し、それ以外の文字列から物理量用語の探索を実施する。両者とも、対象文字列内に用語辞書に登録された複数の用語が発見された場合には、各用語の末尾の位置が文字列内で最も後ろにあるもののうち、用語の長さが最長のものを採用する。こ

の処理は日本語の特徴に依存したもので、最も重要な単語は後に来る傾向があることと、前から探索した場合に一致した単語が後の単語の修飾語の一部である可能性が高いという知見を根拠としている。

1)と2)の推定:  
文字列に含まれる用語のうち末尾に近い最長の用語を採用

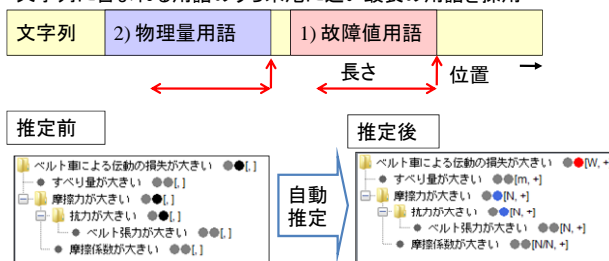


図8：文字列からの物理量自動推定

この自動推定により、手作業で最初から FT 図を作成する場合や Excel ファイルの入力後に、自動的に単位文字列と故障値状態文字列が設定され、物理量に基づく整合性検証と類似度計算の恩恵を受けられることになる。当然ながら、誤変換の可能性があるためその場合の修正操作は必要になるが、用語辞書を充実させてかつ事象を表現する文字列の書き方を標準化していけば、かなりの精度向上が見込まれる。

## 4 ソフトウェアの実務導入と考察

前述のように、開発した FTAid は 2011 年 7 月から企業での実務に導入され、社内での FTA 教育活動でも活用されている。実務での利用による検証とフィードバックについては今後の作業となるが、試用版の導入による検証と評価について既報[3]の考察をまとめると、FT 図の作成については平均 54%の工数削減の効果があり、作成された FT 図の品質が社内基準で合格レベルに達するまでの時間も約 50%削減できているとのことである。また、その要因として、(1) FTAid の機能によるユーザの思考支援の効果、(2) FTAid の操作性の良さ、(3) FTAid による支援手法についてユーザが学習したことが FTA に対する習熟度向上につながったこと、の三点が挙げられている。つまり、開発において重視したソフトウェアの操作性については評価されていて、またソフトウェア導入の目的である FTA の計算機支援については効果が確認され、さらには FTAid による計算機支援を導入したことで FTA に関する教育効果も見込まれると言える。

一方、ソフトウェア開発手法についての考察としては、前述の高度な専用ソフトウェアの開発の課題

を解決して、実務への導入に成功していると言える。一つ目の課題である開発コスト低減は MZ Platform を用いることで達成し、もう一つの課題である操作性向上については繰返し型の開発プロセスを適用することでユーザからの意見を速やかに反映することで解決した。このような開発は他のソフトウェア基盤を用いても可能と考えられるが、今回のように短期間で実務への導入まで実現できた理由としては次のような要因が考えられる。すなわち、(1) この種の開発に適しているかつ開発主体となる著者らが精通するソフトウェア基盤を用いたこと、(2) ソフトウェアのユーザとなる企業側でも試用版の利用や検証などの工数を充分かけたこと、(3) 実装対象となる支援手法がすでに学術論文としてまとめられていたこと、の三点である。

## 5 むすび

本稿は、製造業における高信頼性設計の実現に向けて、大学での研究成果を企業の実務で利用するためのソフトウェア開発を実施した事例として、FTA 支援ソフトウェアの開発について報告した。本報告では、高度な専用ソフトウェア開発における課題の一つである開発コストを低減するために、著者らが開発したソフトウェア基盤を用いて開発を実施し、実際に短期間での開発を実現した。また、ソフトウェア開発におけるもう一つの課題である操作性の問題を回避するために、プロトタイプ作成と評価を短期間に繰り返す開発手法を適用することで、企業の担当者や現場作業者の意見を速やかに反映させることを試み、その効果を確認した。本稿で対象とした範囲のソフトウェア開発では、企業が求める高信頼性設計支援システムの全体像の実現にはまだ道半ばであるが、このような開発手法を引き続きかつより広く適用していくことで、より高度な支援を迅速に企業の実務に導入していけるように取り組んでいきたい。

## 謝辞

本論文の内容は、ジャトコ株式会社および東京大学と共同で実施したソフトウェア開発の事例に基づくものである。開発を実施するにあたり、ジャトコ株式会社の平岡洋二氏と山本克成氏をはじめとする多くの関係者からご意見とご協力をいただいた。また、東京大学の村上存教授と同大学院卒業生の吉永遼氏には物理量次元ベクトルを用いた手法とその実装方法について多くの助言とご協力をいただいた。最後に、MZ Platform は NEDO プロジェクト「もの

づくり・IT 融合化推進技術の研究開発」および「中小企業基盤技術継承支援事業」の一部として開発されたものである。ここに感謝の意を表す。

## 参考文献

- [1] 村上存, 吉永遼: 高信頼性設計のための Fault Tree Analysis 支援システムの開発 (物理量次元インデクシングによる FTA の知識マネジメント), 日本機械学会第 21 回設計工学・システム部門講演会講演論文集, 2209, (2011).
- [2] 古川慈之, 澤田浩之, 徳永仁史: 高信頼性設計のための Fault Tree Analysis 支援システムの開発 (MZ Platform を用いた研究成果活用基盤としての実装), 日本機械学会第 21 回設計工学・システム部門講演会講演論文集, 2210, (2011).
- [3] 平岡洋二, 山本克成: 高信頼性設計のための Fault Tree Analysis 支援システムの開発 (設計現場での Fault Tree Analysis 支援システムの効果検証と実用性評価), 日本機械学会第 21 回設計工学・システム部門講演会講演論文集, 2211, (2011).
- [4] Sawada, H., Matsuki, N., Tokunaga, H., Furukawa, Y.: A Manufacturing Software Development and Operation Framework "MZ Platform" and its Applications in Industry, In Proc. of Advanced Engineering Design, (2004).
- [5] 古川慈之, 澤田浩之, 富澤拓志, 松木則夫: MZ Platform: イベント駆動型コンポーネント指向開発環境を用いたエンドユーザ開発への試み, 情報処理学会第 68 回全国大会講演論文集(1), pp.199-200, (2006).
- [6] Sutcliffe, A., Mehandjiev, N.: End-User Development, Communications of The ACM, 47(9), pp.31-32, (2004).
- [7] 古川慈之, 澤田浩之, 徳永仁史, 手嶋吉法, 松木則夫: エンドユーザ開発による企業内情報化への取り組み, 人工知能学会第 9 回知識・技術・技能の伝承支援研究会, SIG-KST-2009-03-02, (2010).
- [8] ロジャー S. プレスマン: 実践ソフトウェアエンジニアリング, 日科技連, (2005).
- [9] 大城義孝, 村上裕一, 藤原敏: P-FTA-D-QFD を用いた新製品開発の短期化と品質向上, 日産技報, 44, pp.50-53, (1999).
- [10] 小野寺勝重: 実践 FTA 手法, 日科技連, (2000).
- [11] 村上存, 菊地洋輔: 物理量次元インデクシングに基づく故障木解析の知識マネジメント, 日本機械学会論文集 C 編, 75(756), pp.2171-2180, (2009).