特集 要求工学

5 システムの非正常系の要求分析

橋本 正明
九州工業大学

非正常系の要求分析の品質

計算機システムを組み込んだ家電製品などは，その計算機システムの存在も意識しないようなさまざまなユーザによって，さまざまな環境の下で使用されている．しかも，長期に渡って，安全に使用できることも期待されている．そのような優れた製品の品質を得るため，製品に組み込まれているシステムのソフトウェア，すなわち組込みソフトウェアは，その規模の 70% 強が，製品の装置の誤作動や，利用者の誤操作，動作環境の悪影響などの例外条件へ対処するために費やされている．

ところで，組込みソフトウェアは大規模化し複雑化する一方，その開発期間は逆に短くなっている．そのため，要求分析や設計の工程において，例外条件を漏れなく想定することは，ますます困難となっている．しかも，組込みソフトウェアの例外条件を想定するには，ソフトウェアの知識だけでなく，ソフトウェア以外の知識，すなわち，装置や，利用者，動作環境などの知識も必要である．そのため，ソフトウェア技術者が，例外条件を，ソフトウェア仕様から漏らしてしまうことも起きる．現実に，そのような仕様漏れが，設計やり直しや，テスト工数の増加を引き起し，時には製品に障害を発生させることもある．経済産業省の実態調査においても，組込みソフトウェア仕様の品質は，製品の重要な品質問題の 1 つに挙がっている [1]．

そこで，本稿においては，組込みソフトウェアの例外条件のうち，特に仕様から漏れやすいものに着目して，それをまとめて非正常系と呼び，その漏れを防ぐための要求分析について解説する．なお，正常系および非正常系の用語は，以下のように定義しておく．正常系は，ソフトウェア使用マニュアルに記載されるような，ソフトウェア設計工程開始時にすでに定義されているソフトウェアの振舞いを指す．一方，非正常系は，正常系から逸脱した例外条件による振舞い，たとえば装置の軽微な故障や，過負荷，利用者の誤操作，環境の影響などがいくつか重なって起きる障害を指す．

本稿では，以下，従来の障害分析手法の主要なものを概説し，その考察も含めて非正常系要求分析手法の要件を解説する．次に，その要件を満たすような研究アプローチについて，筆者らの研究を例にして解説し，最後に今後の課題に触れる．

従来の障害分析手法

従来から，ハードウェア機器やプラントなどの設計に，種々の障害分析手法が適用されてきた．そのうち，典型的な例として，FMEA と FTA と HAZOP を以下に概説する．この分析技術は，ハードウェア機器やプラントなどに適用するために発展してきたが，ソフトウェアへ適用することも研究されている [2) ～4)]．

■ FMEA（Failure Mode and Effect Analysis）

FMEA は，システムを構成する部品や装置に故障が発生した場合，その故障が，システムへどのような障害をどの程度与えるかを分析するための手法である．システムの設計工程において，FMEA によってシステムが受ける障害を分析することによって，故障モード（本稿では，システムの障害の原因となる故障の種類を指す）の重要なものを見つけ出す．次に，図 -1 の表に示すように，その故障モードごとに，その影響度や，発生率などによって致命度を算出して，対策をとるべき優先順位をつける．このように，FMEA は，どの部品や装置の

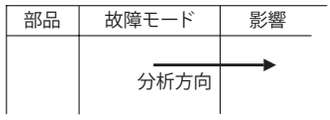

図 -1 FMEA

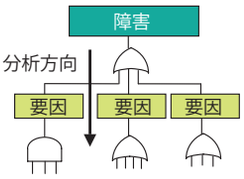

図 -2　FTA

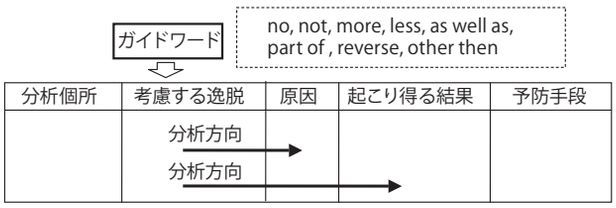

図 -3　HAZOP

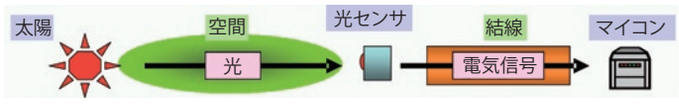

図 -4　街灯システムの動作環境

故障が，システムに大きな障害を発生させるかを見つけ出すための手法である．そのため，ボトムアップ手法と呼ばれている．

■ FTA（Fault Tree Analysis）

FTA は，システムにとって望ましくない障害を最初に定義し，その障害を発生させる原因となる部品や装置の故障を見つけ出すための手法である．この手法は，システムの障害から逆算して部品や装置の故障を見つけ出すため，前述の FMEA とは分析の方向が逆であり，トップダウン手法と呼ばれている．FTA を用いた分析においては，**図 -2** に示すように，システムの障害と部品や装置の故障との因果関係を，木構造（システムの障害が根，部品や装置の故障が葉）と論理記号（AND や OR など）を使用して表した故障木（Fault Tree）の図を作る．さらに，その部品や装置の故障に発生率を割り当てることによって，システムの障害の発生率を得ることもできる．

■ HAZOP（the HAZard and OPerability）

HAZOP はもともと，化学プラントの設計などにおいて，故障や障害などの危険要因を分析するための手法である．具体的には，化学プラントの特定の個所を流れている化学物質に対して，たとえば圧力上昇や流れ停止など，正常な流れから逸脱する現象を想定するためのガイドワードを適用する．そのガイドワード適用によって想定された逸脱現象をもとにして，**図 -3** に示すように，その現象を起こす原因となる部品や装置などの故障と，その現象から起こり得る結果となるシステムの障害を分析する．このように，逸脱現象を中心にして，その原因となる部品や装置の故障と，結果となるシステムの障害という両方向へ分析が進むので，HAZOP はトップダウンとボトムアップの両方の性質を備えた手法である．

非正常系要求分析手法の要件

組込みソフトウェアを例にして，その非正常系の要求分析手法の要件として，重要と思われるものを以下に解説する．

■ 動作環境

業務系システムは，CPU や，メモリ，ハードディスク等の限定された種類の装置を用いて動作する．一方，組込みシステムは，センサやコントローラなどのさまざまな種類の装置を用いて動作する．しかも，その装置の例外条件が，非正常系の振舞いを起こす大きな原因となっている．また，組込みシステムは，利用者による操作のほか，**図 -4** の街灯システムの例に示すように，動作環境からくる光などの情報もトリガとなって動作する．しかも，その利用者や動作環境の例外条件も，非正常系の振舞いを起こす大きな原因となっている．そのため，システムの中の装置に着目するとともに，利用者や動作環境にも着目することが求められる．

■ 正常系のシステム・アーキテクチャ設計

システムに障害が起きるには，その原因が存在し，その原因から障害へ至る因果関係がある．しかも，原因を起こした装置と，障害が認知された装置が異なる場合が多い．その異なる装置の間を流れる情報が，正常な内容から逸脱することによって，前述の因果関係が伝達される．そのため，組込みシステムの中の情報の流れと，その情報の流れを作り出す情報処理プロセスの処理内容に着目することも求められる．

前述のように，システムの中の装置が判明しており，

さらにシステムの中の情報の流れと，その情報の流れを作り出す情報処理プロセスの処理内容が判明してから，非正常系の要求分析が可能となる．すなわち，ソフトウェアも含めた正常系のシステム・アーキテクチャの設計が終われば，非正常系の要求分析に着手できる．

■ 障害シナリオ

装置故障などの重大な例外条件が唯一の原因となって発生する障害は，想定が困難ではない．一方，装置の劣化や，過負荷，利用者の操作タイミングのずれ，環境の影響などは，個々に見れば重大ではない．しかし，それがいくつか重なって起きる障害は，想定が困難である．この例外条件の重なりは，前述の情報処理プロセスが，軽微に逸脱した情報をいくつか入力して処理した結果，重度に逸脱した情報を出力することによって起きる．実際，組込みソフトウェアの熟練技術者は，軽微なものも含めた例外条件の重なりを障害シナリオとしてとらえ，それを丹念に追うことによって，見落としやすい障害も想定できる．そのため，例外条件の重なりを表すための障害シナリオに着目することが求められる．

■ 障害分析手法

組込みソフトウェアの熟練技術者がFMEAの考え方を適用する際は，装置の種別ごとに分類された故障に最初に着目して，分析を進める．その故障の発生は，障害シナリオの始点となる．また，FTAの考え方を適用する際は，製品の品質仕様に反する障害に最初に着目して，分析を進める．その障害の発生は，障害シナリオの終点となる．また，HAZOPの考え方を適用する際は，組込みシステムの中を流れる情報の逸脱に最初に着目して，分析を進める．その情報逸脱の発生は，障害シナリオの中間点となる．このように，従来の3つの障害分析手法をまとめて見ると，最初に着目する障害シナリオの中の個所が相互に異なるという利点を持つ．しかし，その3手法を個別に適用すると，その3手法にまたがって判明する例外条件の重なりを見逃しやすいという欠点がある．この欠点を補完して例外条件の重なりの分析漏れを防止するには，その3手法の考え方を統合する分析手法が望まれる．

研究アプローチ例

前述の要件を満たそうとする研究アプローチについて，筆者らの研究[5]を例にして解説する．

■ IFD（Information Flow Diagram）

前述の要件に解説したように，非正常系の要求分析は，正常系のシステム・アーキテクチャ設計において決められた装置と情報処理プロセスに着目することが求められている．そのため，IFDは，以下に解説するデバイス・ダイアグラムDDとプロセス・ダイアグラムPDを組み合わせて構成する．なお，街灯システムを例にしたIFDを，図-5に示す．

■ DD（Device Diagram）

DDは，設計において決められた装置と，その装置の接続を表す．そのほかに，製品の利用者を含めた動作環境内の対象物も表す．たとえば，図-5に示すように，光センサを用いて昼夜判断を行う街灯システムにおいては，光を発する太陽も，装置として表す．DD中に装置を表すには，同図に示すように四角形を用い，その四角形を三分して，順に装置と属性と故障モードの名前を書く．故障モードは，その装置において発生し得るノイズや，劣化，断線などの故障の種類を示す．属性は，非正常系の分析に必要なもの，たとえば故障と認識されるノイズのレベルや，装置劣化が起きる経過時間などを表す．

ところで，情報は，装置の間を，電気や，光，音などの物理媒体によって伝えられる．しかも，その物理媒体に減衰や断絶などの故障が起きれば，情報の流れが阻害され，障害に至ることがある．そのため，非正常系の要求分析に当たっては，情報を伝えるための物理媒体に着目することも重要である．そこで，その物理媒体をキャリアと呼び，キャリアが装置を接続するものと見なす．キャリアは，図-5に示すように，菱形と矢印を用いて表し，その矢印が2つの装置を接続する．矢印の方向は，情報が流れる方向を示す．キャリアと属性と故障モードのそれぞれの名前は，菱形と重ね合わせて書く．

■ PD（Process Diagram）

PDには，プロセスの機能を静的に表すための手法IDEF0を適用する．その手法においては，プロセスをアクティビティ単位に分割し，そのアクティビティを，図-5に示すように四角形を用いて表し，その中にアクティビティの名前を書く．そのアクティビティには，入力情報と，出力情報，制御情報，メカニズムのそれぞれを，必要に応じて矢印によって指定し，その名前も書く．アクティビティは，それらの情報の流れによって，相互に接続する．アクティビティは入力情報をメカニズムによって処理し，出力情報を作り出す．制御情報は，その処理の条件を決める．

PDは，DDに示した各装置が処理するアクティビ

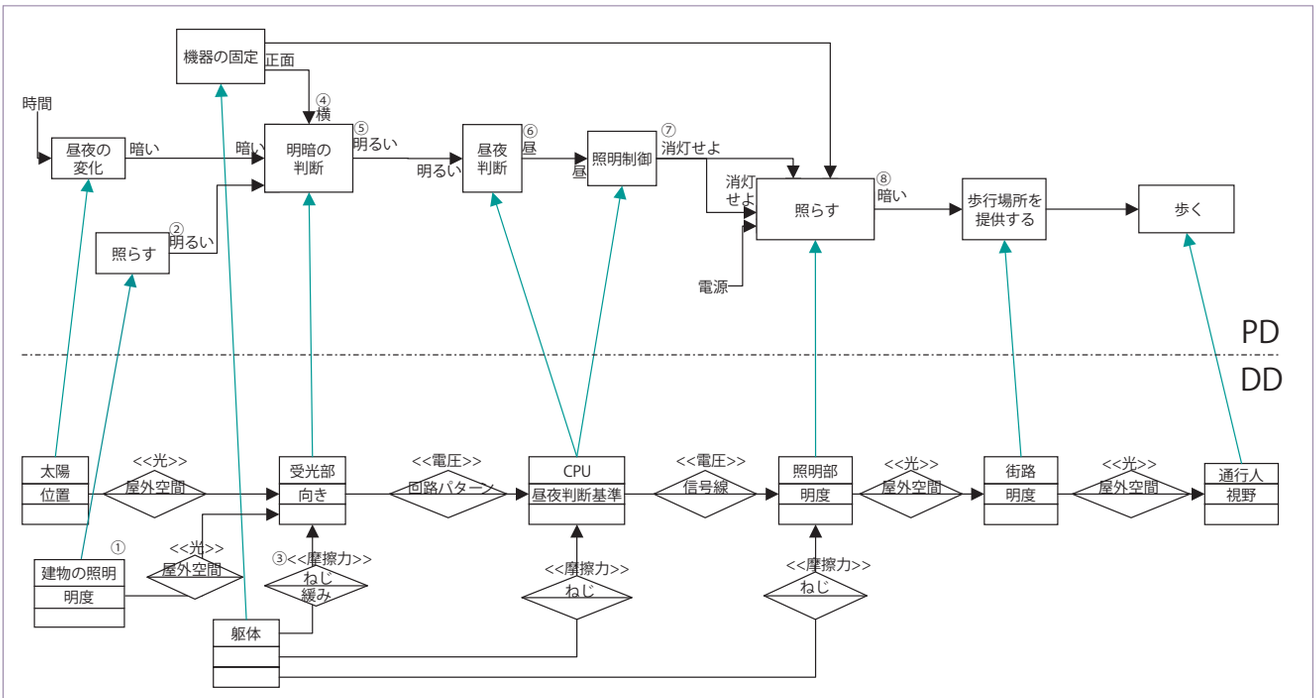

図 -5　街灯システムの IFD 例

ティと，そのアクティビティ相互の情報の流れを表す．DD と PD の関係は，装置と，その装置が処理するアクティビティをメカニズムの矢印によって接続して表す．

■ 分析の手順

非正常系の要求分析の手順を以下に述べる．

(1) 正常系 IFD の作成

最初に，システム・アーキテクチャ設計の内容に基づいて，正常系の IFD を作成する．

(2) 回避すべき障害現象の追記

製品のシステム要求仕様において，回避すべき障害現象が，安全基準として与えられている．その障害現象を，正常系の IFD に追記する．具体的には，その障害現象をプロセスに追記し，その障害によって周辺に及ぼす悪影響も，そのプロセスが出力する情報フローの障害現象として PD に追記する．また，PD 上の各プロセスの間の情報フローに対して，ガイドワードを適用することによって，情報フローが例外になる現象も想定する．この例外現象も PD に記述する．**表 -1** にガイドワードの例を示す．

影響種別	ガイドワード
挙動	停止，不安定，固定，…
負荷	過大，過少，…
意味	範囲外，未定義，…
手順	順序違い，タイミング違い，…
頻度	継続的，反復的，一時的，…

表 -1　ガイドワードの例

(3) IFD を活用した例外条件の検討

障害現象も含めた例外現象が発生するための条件を，例外条件と呼ぶ．前述の (2) 項に述べた例外現象に対して，その例外条件を，IFD を用いて検討する． PD 上において例外現象が想定された情報フローは，その情報を出力するプロセスに結びついている．そのプロセスを当該プロセスと呼ぶ．

当該プロセスは，入力情報と制御情報，およびメカニズムとなる装置のみに結びついている．そのため，当該プロセスから出力される情報の例外現象を起こす例外条件は，当該プロセスの入力情報と制御情報，およびメカニズムとなる装置の例外条件のいずれか，あるいはそれらの組合せのみによって表される．さらに，その入力情報と制御情報の例外条件は，入力情報と制御情報自身が情報フローパス上に存在するため，その情報の送信プロセスの例外条件か，その情報フローパス自体の例外条件のいずれか，あるいはそれらの組合せのみによって表される．一方，メカニズムとなる装置における例外条件も，装置の実現手段がハードウェアとソフトウェアからなるため，ハードウェアの例外条件か，ソフトウェアの例外条件のいずれか，あるいはそれらの組合せのみによって表される．

そのため，例外現象を起こす例外条件は，以下のよう

に分類できる．

(a) 当該プロセスのメカニズムとなる装置の例外条件．これは以下の 2 つに分類される．

①装置の物理的な故障は，部品の故障に関する設計ドキュメントから得られる．

②装置の論理的な故障は，装置のソフトウェア設計の不備を想定することに等しい．

(b) 当該プロセスの入力情報と制御情報の例外条件．これは，以下の 2 つに分類される．

①入力情報と制御情報の送信プロセスの例外条件．

②入力情報と制御情報が伝達される情報フローパス自体の例外条件は，その情報フローパスの故障によって生じる．また，入力情報と制御情報を運ぶキャリアと同じキャリアを発信する，なりすまし装置から情報を受信しても生じる．

(c) 上記に述べた 2 つ以上の例外条件の複合．

上記の分類に従って，着目している例外現象を起こす例外条件を検討する．その例外条件が存在する場合，これを採用する．その例外条件が存在しない場合，その例外現象は起こり得ないので，検討は中止する．上記の (b) と (c) については，例外条件自身が情報フローの例外現象となっている．そのため，その例外現象に対して，上記分類による検討を再度繰り返す．

上記のなりすましが発生し得ると判断された場合，そのなりすまし装置とキャリアを DD に追記する．さらに，なりすまし装置の動作をプロセスとして捉え，そのプロセスを PD に追記する．また，そのプロセスから，なりすまし情報の送信先プロセスへ，情報フローを追記する．

以上に解説した検討において，想定された例外現象から，故障またはなりすましの装置に至る分析の中で判明した情報フローのつながりを障害シナリオ断片と呼び，IFD に記載しておく．なお，例外現象から，それが発生するための原因となる例外条件を検討するのは，因果関係を逆にたどるトップダウン分析である．

(4) 障害シナリオの構築

前述の (3) 項において想定された故障を出発点として，因果関係に沿ってボトムアップに障害シナリオを構築する．具体的には，個々の故障がプロセスの出力情報に与える影響を想定する．次に，その出力情報が，PD 上の情報フローに沿って伝達される方向に情報を追跡し，伝達された情報が，障害を表す情報へ到達するまで追跡する．故障を出発点として，障害を表す情報まで到達する一連の情報フローのつながりが，1 つの障害シナリオとなる．

障害シナリオの追跡において，故障のすべての組合せを考慮しなければならない．しかし，その組合せは情報フローの合流によって発生するので，障害シナリオの追跡は，以下の手順によって行う．

(a) 追跡している情報を入力するプロセスに着目する．そのプロセスのすべての入力と制御の情報，および，メカニズムについて，その可能な例外現象のすべての組合せを考慮する．その組合せに対して，着目しているプロセスに，例外現象を持つ出力情報を出す可能性があれば，その出力情報に対して，追跡を繰り返す．

(b) 上記の (a) の追跡において，着目しているプロセスが，例外現象を持つ出力情報を出す可能性がなければ，その追跡を中止する．

(c) 上記の (a) の追跡において，着目しているプロセスとその出力情報が，前述の (3) 項において述べた障害シナリオ断片の中に現れている場合，その断片を，それ以後の追跡において再利用する．

以上に述べたように，すべての故障について追跡し，その都度得られた障害シナリオが本手法の成果物となる．**図 -6** に，街灯システムから少し離れた所にある建物の照明と，街灯システムの躯体に受光部を取り付けているネジの緩みが，原因の重なりとなって起きる障害シナリオを示す．なお，図 -6 の中の①，②，③，... と，図 -5 の中の①，②，③，... は，相互に対応している．また，図 -6 の中の［　］に，その障害シナリオを構築する際に必要な IFD の追跡内容を説明する．

ところで，前述の (3) 項においては，先に故障やなりすましの確実な存在を確認して，その後，(4) 項において障害シナリオを構築した．論理的には，その逆の手法，すなわち先に障害の可能性を確認する手法も考えられる．しかし，熟練技術者は実在性を重視するので，本稿に述べた手法を好む傾向が高い．

今後の課題

システムの非正常系の要求分析手法については，まだ多くの研究が望まれる．そこで，今後の課題を，以下にあげる．

• 動的な分析手法

本稿の研究アプローチ例として，IFD を用いた静的な分析手法を解説したが，筆者らは状態遷移の考え方を用いた動的な分析手法 [6] も研究中である．特に，組込みシステムにおいては，タイミングなどの動的な側面が，非正常系の要求分析へ大きく影響している．UML (Unified Modeling Language) などの分析手法においても，ユースケース図やクラス図のような静的なものと，

①街灯システムから少し離れた所に，建物の照明がある．
［図 -5 の DD 内の①によって示す「建物の照明」は，この障害シナリオの原因の 1 つである．この原因が PD 内の「照らす」へ伝わる］
②その建物の照明が，街灯システムの方を照らしている．
［建物の照明の明るさが，街灯システムの受光部による「明暗の判断」へ伝わって，判断へ影響を及ぼす］
③街灯システムの躯体に，受光部を取り付けているネジが緩んだ．
［DD 内の③によって示す「ネジ」の「緩み」も，この障害シナリオのもう 1 つの原因である］
④ネジが緩んだため，本来は正面を向いているはずの受光部が横を向いた．
［ネジの緩みが，DD 内に示す「躯体」から，PD 内に示す「機器の固定」へ伝わり，本来は正面を向いている「受光部」が横を向く．その横を向いたことが，明暗の判断へ伝わり，判断へ影響を及ぼす］
⑤横を向いた受光部に建物の照明があたり，本来は夜のため暗いはずなのに，受光部は明るいと判断した．
［受光部による誤った明暗の判断が，「昼夜判断」へ伝わり，判断へ影響を及ぼす］
⑥CPU は，受光部から明るいという情報を得たので，本来は夜であるのに，昼と判断した．
［誤った昼夜判断が「照明制御」へ伝わり，制御へ影響を及ぼす］
⑦照明制御は，CPU から昼という情報を得たので，本来は点灯しておかなければならない夜なのに，消灯の指示を出した．
［誤った消灯指示が照明部の「照らす」へ伝わり，影響を及ぼす］
⑧夜なのに消灯したため，街灯システムの下が暗くなり，街灯システムの本来の機能を果たせない．
［街灯システムの下の街路が暗く，歩行に困難を来す］

図 -6　障害シナリオの例

シーケンス図や状態遷移図のような動的なものが，相補的に混在している．非正常系の要求分析手法においても，静的なものと動的なものの相補的な関係について研究が望まれる．

• 関係性に着目した分析手法

本稿でも解説したが，従来の障害分析手法の中にも，FTA のようなトップダウン手法のほかに，FMEA のようなボトムアップ手法もある．故障や障害は，システムに求められる機能から見れば，本質的ではない実装技術によって発生するものが多い．たとえば，情報を伝えるのに電流を通すワイヤーを用いたため，そのワイヤーが誘導電流を拾ってその情報が破壊され，障害に至ることもある．このように，故障や障害は目的なしに発生するため，非正常系の要求分析は，トップダウンの分析手法が必ずしも上手くは働かない領域と考えられる．そのため，本稿に解説した研究アプローチ例においても，情報フローによって生じる関係性に着目した分析手法が採られている．このような関係性に着目した効率的な分析手法の研究も望まれる．

• システム・アーキテクチャ

前述の関係性が複雑になるに従って，分析漏れも生じやすい．そこで，非正常系の分析手法のみではなく，非正常系の対策処理技術も研究が重要である．その研究の中で前述の関係性を制限することによって，非正常系の要求分析が容易となるようなシステム・アーキテクチャの研究も必要である．

• プロジェクト・マネジメント手法

本稿に解説したように，非正常系の要求分析は，正常系のシステム・アーキテクチャの設計後に可能となる．このように，設計と要求分析が時間的に逆転するので，その 2 つの工程のプロジェクト・マネジメントには注意を要する．このマネジメント手法も，今後の課題である．

• 専門知識の分析手法

非正常系の要求分析の研究には，上手く適用できる理論体系がなかった．そのため，シーズ指向の研究方法は採ることができない．そこで，筆者らは，非正常系要求分析の実務熟練者が持っている専門知識を分析することによって，研究を進めている．その専門知識は，深い暗黙的な知識を含んでいるので，専門知識の分析自体が大きな研究課題である．情報システムがさらに高度化するに従って，専門知識を分析する機会はますます増えてくる．そのため，専門知識の分析手法に関する研究も重要である．

参考文献

1) 経済産業省：2007 年版組込みソフトウェア産業実態調査報告書 (June 2007).
2) Pentti, H. and Atte, H. : Failure Mode and Effects Analysis of Software-based Automation Systems, STUK-YTO-TR 190, p.35 (2002).
3) Dehlinger, J. and Lutz, R. R. : Software Fault Tree Analysis for Product Lines, Proceedings of the Eighth IEEE International Symposium on High Assurance Systems Engineering, pp.12-21 (2004).
4) Redmill, F., Chudleigh, M. and Catmur, J. : System Safety : Hazop and Software Hazop, John Wiley & Sons Ltd, p.248 (1999).
5) 新屋敷泰史，三瀬敏朗，橋本正明，片峯恵一，鵜林尚靖，中谷多哉子：情報フロー・ダイアグラムによる組み込みソフトウェア非正常系の要求分析の一手法，情報処理学会論文誌，Vol.48, No.9, pp.2894-2903 (Sep. 2007).
6) Mise, T., Shinyashiki, Y., Hashimoto, M., Ubayashi, N., Katamine, K. and Nakatani, T. : An Analysis Method with Failure Scenario Matrix for Specifying Unexpected Obstacles in Embedded System, The proceeding of the 12TH Asia-Pacific Software Engineering Conference, pp.447-454 (2005).

(平成 20 年 2 月 3 日受付)

橋本　正明（正会員）　hasimoto@ai.kyutech.ac.jp

九州工業大学大学院情報工学研究科教授．ソフトウェア工学やプロジェクトマネジメント，TOC，物流などの研究に従事．工学博士．