

ソフトウェア要求仕様書に基づいた テスト項目作成手法の提案

喜多 義弘^{†1} 鈴木 三紀夫^{†2} 秋山 浩一^{†3}
片山 徹郎^{†1} 西 康晴^{†4}

ソフトウェア開発において、上流工程での欠陥による下流工程での手戻りを防ぐために、要求仕様書の欠陥の早期発見の重要性が求められている。本論文では、要求仕様書からキーワード群を抜き出し、それをテスト観点として分類・整理し、そこからテスト項目を機械的に導き出すテスト項目作成手法を提案する。そして、要求仕様書の例を用いてテスト項目を作成し、テスト要求に対するテスト項目の網羅性について検証した。その結果、本提案手法によるテスト項目の高い網羅性を確認し、要求仕様書の欠陥を発見できた。

Proposal of a Method Generating Test Items from Software Requirements Specification

YOSHIHIRO KITA,^{†1} MIKIO SUZUKI,^{†2} KOICHI AKIYAMA,^{†3}
TETSURO KATAYAMA^{†1} and YASUHARU NISHI^{†4}

It is important to detect the defects of requirements specification early, in order to prevent the return to the lower process by the defect in the upper process. This paper proposes a method for generating test items that draws the test items by the testing viewpoints and extracts keywords by the requirements specification. We generated the test items by using the sample of requirements specification, and verified the comprehensibility of the test items over the test requirements. As a result, we confirmed the high comprehensibility of the test items, and found the defect of the requirements specification by using the proposed method.

1. はじめに

ソフトウェア開発において、上流工程での欠陥は目に見えないため発見されにくく、下流工程であるテスト工程で初めて発見される場合が多い。その都度手戻りによる修正作業が行われるため、多大な時間とコストが費やされる。手戻りの原因として、要求仕様の不備が全体の3割近く占めており、要求仕様書の欠陥の早期発見の重要性が求められている¹⁾。

要求仕様書の欠陥として最も多いのは、要求の記述漏れである¹⁾。そのため、要求仕様書において要求の記述漏れを発見することは重要であり、全ての要求を網羅するためのテストを行うことが必要である。

ソフトウェアテストには、様々な「観点」が必要である。例えば、Ostrandの4つのビュー(ユーザ、仕様、設計、欠陥)²⁾や、ISO/IEC9126の6つの品質特性(機能性、信頼性、使用性、効率性、保守性、移植性)³⁾などが観点の一例である。しかし、これらの観点を実際のテストでそのまま使用するには粒度が粗い。そのため、テスト要求分析のための、粒度がより細かい観点が必要である。この観点を、本論文では「テスト観点」と定義する。

要求仕様書からテスト観点となるキーワードを挙げ、そこからテスト項目を作成することは重要である。しかし、挙げたキーワードからそのままテスト項目を作成するだけでは、要求仕様書の要求を全て網羅できるテスト項目を得られるとは限らない。挙げたキーワードを組み合わせることでテスト項目を作成することにより、より多くの要求を網羅でき、要求の記述漏れなどの要求仕様書の不具合を発見できると考えられる。また、挙げたキーワードの組み合わせによっては網羅できる要求も変わるため、その組み合わせ方も考える必要がある。

そこで本論文では、要求仕様書からテスト観点となるキーワード群を抜き出して4つのタイプに分類・整理し、それらを組み合わせることでテスト項目を作成する、テスト項目作成手法を提案する⁴⁾。

この提案手法の有用性を確かめるために、要求仕様書の例として、組込みソフトウェア管理者・技術者育成研究会 (SESSAME) が公開している「話題沸騰ポット GOMA-1015型」の要求仕様書第7版⁵⁾からテスト観点に沿ったキーワード群を抜き出し、そこからテスト項目を作成する実験を行い、その結果について考察する。

^{†1} 宮崎大学, University of Miyazaki

^{†2} TIS 株式会社, TIS Inc.

^{†3} 富士ゼロックスアドバンスドテクノロジー株式会社, Fuji Xerox Advanced Technology Co., Ltd.

^{†4} 電気通信大学, University of Electro-Communications

2 ソフトウェア要求仕様書に基づいたテスト項目作成手法の提案

2章では、関連研究について述べ、既存の要求分析の手法と本提案手法との違いを示す。3章では、テスト観点の導出について述べ、テスト観点をキーワードを組み合わせることでテスト項目を作成する手法を提案する。4章では、本提案手法の有用性を評価するために、要求仕様書の例を用いてテスト項目を機械的に作成し、テスト要求に対するテスト項目の網羅性の検証実験を行う。5章では、4章での実験の結果を検証し、本提案手法の有用性について考察する。

2. 関連研究

以前より、要求分析のための、要求仕様書の解析手法の研究が行われてきた。システム運用知識抽出手法⁶⁾、要求仕様書における欠陥検出方法⁷⁾、および、要求仕様のテスト方法論 Tiramis⁸⁾ は、要求仕様書の要求をモデルやレビューにより分析する手法である。これらの手法は、要求記述をアクタ、事前状態、イベント、入力、動作、出力、応答、事後状態という構成要素に分解し、各要素の相互関係を分析することにより、要求の完全性を確認する。

要求仕様書から自動的にテストケースを作成する手法として、自然言語から論理的なフォーマットへ自動的に変換して要求を分析し、テストケースを自動的に作成する要求分析ツール RETNA による要求分析手法⁹⁾ や、要求仕様書から各要求に対するソフトウェアの入力および出力の組を導出し、そこからテストケースを自動的に作成する手法¹⁰⁾ などが挙げられる。

しかし、これらの手法は、要求のキーワードを用いてテストケースを作成するが、それらのキーワードを組み合わせることでテスト項目を作成することはない。キーワードからそのままテストケースを作成するだけでは、実行すべきである全てのテスト項目を網羅できるとは限らない。キーワードを組み合わせることによって、作成できるテスト項目数を増やし、より多くのテスト項目を網羅することにより、要求の記述漏れなどの要求仕様書の不具合を発見できることが見込まれる。

そこで、要求仕様書からテスト観点となるキーワードを抜き出して4つのタイプに分類・整理し、それらを組み合わせることでテスト項目を作成するテスト項目作成手法を提案する。この手法は、キーワードの全ての組み合わせを機械的に網羅することで、要求分析の網羅性を高めることを狙う。また、キーワードの組み合わせにより、要求仕様書に記述していない要求をテストするテスト項目を作成できることが見込まれ、これにより、要求仕様書の記述漏れを発見できると考えられる。

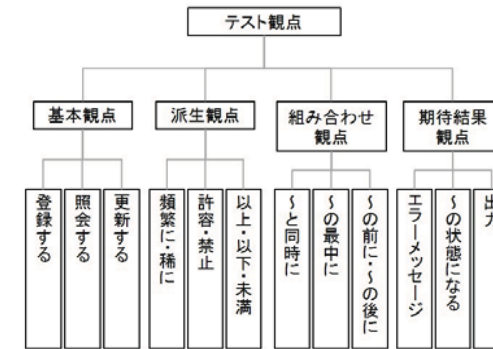


図1 テスト観点ツリー
Fig. 1 Tree of test viewpoints

3. テスト観点およびテストアーキテクチャ設計

ソフトウェアテストでは、様々なテスト観点を組み合わせることでテスト項目を作成する必要がある。思いつきでテスト観点を組み合わせるだけでは、網羅的なテストを行うことはできない。そのため、テスト観点を分類し組み合わせる文法構造を構造化しておくことにより、文章を作るようにテスト条件を作成することができる⁴⁾。

そこで本研究では、動詞や形容詞などの文章の構成要素をテスト観点をキーワードとし、それらを4つのテスト観点到分類し、キーワード群を組み合わせるためのテストアーキテクチャ設計に従って、分類したキーワードの全ての組み合わせを機械的に導出する。

分類したテスト観点、および、それらのテスト観点を基に、文法構造を用いたテストアーキテクチャ設計について、以下に述べる。

3.1 テスト観点

この節では、テスト項目作成のためのテスト観点について提案する。

ツリー状の階層構造で表現したテスト観点を、図1に示す。本提案手法でのテスト観点は、大きく4つの観点がある。

基本観点は、要求の振舞いに対する観点である。基本観点をキーワードとして、動詞を要求から見つけ出す。具体的には、「登録する」、「照会する」、「検索する」などのキーワードを探し出す。

派生観点は、要求の振舞いの程度に対する観点である。これにより、テスト項目のバリ

3 ソフトウェア要求仕様書に基づいたテスト項目作成手法の提案

エーションが増える。派生観点のキーワードとして、基本観点のキーワードを修飾する形容詞や副詞を要求から見つけ出す。具体的には、「頻繁に・稀に」、「許容・禁止」、「以上・以下・未満」などの、ソフトウェアの異常を誘発するためのキーワード(要因)を挙げる。

組み合わせ観点は、基本観点および派生観点を組み合わせに対する観点である。組み合わせ観定のキーワードとして、集合、関係、および、組み合わせを示すキーワードを要求から見つけ出す。具体的には、「～と同時に」、「～の最中に」、「～の前に・～の後に」などのキーワードを探し出す。

期待結果観点は、期待結果の属性に対する観点である。期待結果観定のキーワードとして、「エラーメッセージ」、「～の状態になる」、「出力」などの期待結果を表すキーワードを要求から探し出す。この期待結果観点は、次の点で他の3つの観点とは異なる。基本観点、派生観点、および、組み合わせ観点は、見つけ出したテスト観点自体をさらに整理・分解してテストを細分化することが可能である。しかし、期待結果観点は結果系の観点であるため、例えば、「エラーメッセージ」というキーワードを見つけたとき、それを「(メッセージの)文字の大きさ」、「文字の種類」、「文字の配置」と分解できても、そこから直接にテスト設計をすることはできない。そのため、期待結果観点は単独に使用するよりも、基本観点、派生観点、組み合わせ観点を使用してテスト項目を作成した後に、期待結果の観点で漏れがないか確認するために使用する。

3.2 テストアーキテクチャ設計

4つのテスト観定を使用して、網羅的なテストを設計する手法について説明する。

まず、テスト対象を発見し決定するところから始める。テスト対象は、時間、規模、および、処理方法の3つの軸でとらえることができる⁴⁾。例えば、単体テストのときに規模が小さく、処理方法がバッチ処理であるテスト対象を考えた場合、「モジュール」や「関数」というテスト対象を発見できる。また、同じ単体テストであっても、規模が大きく、処理方法がオンラインであるテスト対象を考えた場合、「画面」というテスト対象を発見できる。このように当該テストフェーズにおいて、規模と処理方法の組み合わせを考えることにより、テスト対象を発見することができる。

次に、発見したテスト対象に対し、基本観点、派生観点、組み合わせ観点、および、期待結果観点を用いてテスト項目を作成する。テスト観定を用いることにより作成できるテスト項目の構造を、以下に4つ述べる。

(1) 基本構造

基本観定(動詞)を使用することにより作成できる、以下のテスト項目の構造を、基

本構造とする。

{テスト対象}に{基本観定}させる。

または

{テスト対象}に(目的語)を{基本観定}させる。

(2) 派生構造

基本構造に派生観定(形容詞・副詞)を追加することにより、

{テスト対象}に{派生観定}の(目的語)を{基本観定}させる。

という派生構造を作成できる。

(3) 組み合わせ構造

組み合わせ観定(関係・組み合わせ)を用いることにより、複数の基本構造や派生構造を組み合わせることができる。例えば、基本構造のテスト項目を2つ組み合わせると、

{テスト対象}に{基本観定}させることと、{テスト対象}に{基本観定}させることを{組み合わせ観定}の組み合わせ方法で実施する。

という組み合わせ構造を作成できる。

(4) 期待結果構造

基本構造、派生構造、および、組み合わせ構造のそれぞれに対し、期待結果観定(期待結果)の期待結果を考えることができる。テスト設計において期待結果を考える理由は、期待結果を網羅するためであり、想定する期待結果を導出できるように基本構造、派生構造、および、組み合わせ構造を考えるためである。例えば、基本構造に対し、期待結果観定を追加することにより、

{テスト対象}に{基本観定}させると{期待結果観定}になる。

という期待結果構造を作成できる。

派生構造や組み合わせ構造において、無作為にテスト観定のキーワードを組み合わせることにより、作成するテスト項目の数が爆発的に増える。そして、爆発的に増えたテスト項目を人手によって全てテストすることは、現実では不可能に近い。そのため、テスト項目の数をテスト実行可能なレベルにまで絞り込む必要があり、そのための手法については今後の課題である。

4. 実 験

テスト観定を用いたテスト構文作成手法の有用性を確認するために、実際の要求仕様書を

4 ソフトウェア要求仕様書に基づいたテスト項目作成手法の提案

表 1 全ての要求項目から抽出した各テスト観点のキーワード数

Table 1 Number of keywords of test viewpoints extracted from all requirements

	キーワード数
基本観点 (動詞)	171
派生観点 (形容詞・副詞)	12
組み合わせ観点 (組み合わせ)	3
期待結果観点 (期待結果)	21

表 2 各構造のテスト項目の適合率および再現率

Table 2 Precision and recall of test items in each structure

	作成したテスト項目数	有効であるテスト項目数	適合率 (%)	再現率 (%)
基本構造	171	68	39.8	22.6
派生構造	4,104	102	2.5	33.9
組み合わせ構造	17,569,566	299	0.002	99.3
(内訳)				
基本 × 基本	29,070	144	0.5	47.8
基本 × 派生	701,784	247	0.04	82.1
派生 × 派生	16,838,712	299	0.002	99.3
期待結果構造	21	21	100.0	7.0

用いて、テスト観点に従いキーワードを抽出し、テスト項目の作成を行った。

4.1 実験準備

要求仕様書の例として、組込みソフトウェア管理者・技術者育成研究会 (SESSAME) が公開している「話題沸騰ポット GOMA-1015 型」の要求仕様書第 7 版⁵⁾を用いた。この要求仕様書には、ソフトウェアに対する要求仕様が 57 項目ある。

実験を行う前に、57 項目の要求仕様を網羅するために必要なテスト項目の総数を計測した。計測方法は、要求仕様書を見ながら人手によって各テスト観点を抜き出し、基本構造、派生構造、および、組み合わせ構造を用いて、実行不可能なものが出ないようにテスト項目を作成し、57 項目の要求仕様を網羅した時点で作成したテスト項目数を、網羅するために必要なテスト項目の総数とした。その結果、301 項目のテスト項目が存在することを確認した。

4.2 実験手順

まず、ソフトウェアに対する要求仕様から、そこに含まれるテスト観点ごとのキーワードを調べる。このとき、重複しているキーワードは 1 つにまとめる。

次に、抽出したテスト観点のキーワードを用いて、基本構造、派生構造、および、組み合わせ構造に従い、テスト項目を機械的に作成する。作成したテスト項目によって、既知である 301 項目のテスト項目のうち、どれだけ網羅することができるかを構造別に調べる。

4.3 実験結果

57 項目の要求項目全てから抽出した各テスト観点のキーワード数を、表 1 に示す。

表 1 より、基本観点のキーワードを最も多く抽出したことが確認できる。また、基本観定の動詞には、ソフトウェアの挙動を示す動詞だけでなく、ユーザの操作も含む全ての動詞が含まれている。

抽出したテスト観点をを用いて各構造のテスト項目を作成し、作成したテスト項目と網羅すべき 301 項目のテスト項目との比較を行い、その適合率および再現率を導出した。ここで、適合率とは、作成したテスト項目のうちどれだけのテスト項目が有効であることを示す割合で

あり、以下の式で求める。

$$\text{適合率} = \frac{\text{有効なテスト項目数}}{\text{作成したテスト項目数}} \times 100(\%)$$

「有効なテスト項目」とは、要求仕様書から抽出でき、かつ、実行可能であるテスト項目である。そして、それ以外のテスト項目は「無効なテスト項目」とする。

再現率とは、有効なテスト項目によって、網羅すべき 301 項目のテスト項目のうちどれだけ網羅できたかを示す割合であり、以下の式で求める。

$$\text{再現率} = \frac{\text{有効な項目数}}{\text{網羅すべき 301 項目のテスト項目}} \times 100(\%)$$

各構造におけるテスト項目の適合率および再現率を、表 2 に示す。表 2 より、基本構造のみの場合、適合率は約 40%であるのに対し、再現率は約 23%程度であることが確認できる。組み合わせ構造の場合、基本構造と比べて作成したテスト項目数が爆発的に増大しているため、適合率が著しく低下しているが、再現率は 99%と高い数値であることが確認できる。

期待結果構文においては、作成できたテスト構文数が少なく、適合率が 100%であるのに対し、再現率が低いことが確認できる。

5. 結果の比較と議論

実験の結果において、自動的に作成したテスト項目の数に着目する。組み合わせ構造に至っては、1700 万以上のテスト項目を作成していることが確認できる。これらのテスト項目を全て実行することは現実的ではない。しかし本論文は、テスト項目の要求仕様書の要求に対する高い網羅性および要求の記述漏れの発見を目的とするため、作成されるテスト項目の数の削減およびテスト項目の実用性については今後の課題とし、以下では、要求仕様書の要求に対する網羅性および要求の記述漏れ発見の可能性について述べる。

5 ソフトウェア要求仕様書に基づいたテスト項目作成手法の提案

表 2 より、組み合わせ構造の再現率が 99%以上であることから、本提案手法によって、要求仕様の項目の大部分をテスト可能であると考えられる。再現率が 100%に到達しなかった理由は、テスト項目 2 つの組み合わせのみを考慮したため、テスト項目を 3 つ以上組み合わせる必要があるテスト項目については網羅できなかったからである。そのため、組み合わせの数を増やすことにより、機械的に作成するテスト項目数は爆発的に増加するが、再現率をより高めることができると考えている。

そして、組み合わせ構造の内訳において、基本 × 派生の組み合わせでも再現率が 80%以上であることから、テスト観点の抽出手法および組み合わせ方次第では、より少ないテスト項目数でもある程度の要求仕様の項目数をテストできることが考えられる。

一方、適合率はテスト観点をより加えた構造になるほど低下し、派生構造においては約 2%、組み合わせ構造においては基本 × 基本の組み合わせでも 1%未満であるため、作成したテスト項目の大部分が無効なテスト項目であると考えられる。

無効なテスト項目である理由として、以下の 2 点が考えられる。

- 実行することが不可能な、あり得ない組み合わせのテスト項目である
- 要求仕様書には書いていない振舞いを行うテスト項目である

前者については、テスト観点の基本観点にはソフトウェアの挙動ではない動詞が含まれているため、それらの動詞を用いて機械的にテスト項目を作成すると、例えば「ソフトウェアがポットの蓋を閉める」や「ソフトウェアが給湯ボタンを押す」などの、実行できないテスト項目を作成してしまう。その結果、無効なテスト項目が増えてしまう。

一方、後者については、網羅すべきテスト項目は、要求仕様書に記述されている要求のみを網羅するように作成したため、網羅すべきテスト項目が足りていない、すなわち、要求仕様書に要求の記述漏れがある場合が考えられる。要求の記入漏れはソフトウェア開発の上流工程のうちに見出すべきである。そのため、要求仕様書が出来上がった段階で、要求仕様書からテスト観点を抜き出し、機械的にテスト項目を作成することにより、要求仕様書の要求の記入漏れを早期に見出せる可能性がある。

そこで、本提案手法で作成し、かつ、実験において無効であったテスト項目によって、要求仕様書の不備を発見できるかの検証を行った。要求仕様書は、4 章の実験で用いた話題沸騰ポットの要求仕様書を再び用いた。まず、無効なテスト項目から明示的に実行不可能なテスト項目を除いてから、残ったテスト項目に沿って要求仕様書を参照した。

その結果、要求仕様書の不備を発見することができた。図 2 は、要求仕様書の一部である。この部分は、給湯ボタンへの要求であり、給湯ボタンを押しながら給湯している最中に

2. 6 給湯ボタン	要求	pot-260	給湯ボタンを押すことによって給湯口から給湯する。
	理由		ポットを持ち上げることなく、また給湯ボタンを無意識に押して期待していないタイミングでお湯が出て怪表（やけど）しないように、給湯したいから。
	説明		
		<ポンプを作動できる場合>	
	□	pot-260-11	以下の条件を全て満たすとき、給湯ボタンを押している間、ポンプを作動する（給湯する）。一つでも条件を満たしていなければ給湯できない。 ・ ロック解除状態であること ・ 水量が適正であること ・ 蓋センサが on であること ・ ヒータ用電源が on であること ・ 温度エラーを検知していないこと ・ 保温行為中であること
		<途中でポンプを停止する場合>	
	□	pot-260-21	給湯中に一つでも以下の条件になったとき、給湯ボタンを押している間でも、ポンプの動作を停止する（給湯を止める）。 ・ 水量が異常になる ・ ヒータ用電源が off になる ・ 温度エラーを検知する

	<停止する場合>
60-21	給湯中に一つでも以下の条件になったとき、給湯を止める。 ・ 水量が異常になる ・ ヒータ用電源が off になる ・ 温度エラーを検知する

条件に「蓋センサが off になる」が抜けている。

図 2 要求仕様書の不備

Fig. 2 A defect of the requirements specification

ポンプを停止するための条件を示している。そして、条件に「蓋センサが off になる」が抜けている。この記入漏れは、従来の手法である要求仕様書に沿ったテスト項目作成手法⁷⁾では発見することができない。本提案手法により、機械的にテスト項目を作成することにより、基本構造「蓋センサを off にする」と基本構造「給湯ボタンを押す」の組み合わせにより、組み合わせ構造「給湯ボタンを押している最中に蓋センサを off にする」というテスト項目を作成することができ、このテスト項目を用いて要求仕様書を参照ことによって、要求の記入漏れを発見することができた。

このことから本提案手法は、無効なテスト項目を機械的に作成しても、それが無効である原因を探ることにより、要求の記入漏れを発見できる可能性があるといえる。

そして、本提案手法は、要求仕様書からテスト観点到属するキーワードを抽出してテスト項目を作成するため、ソフトウェア開発の上流工程の段階で要求の記入漏れを確認すること

6 ソフトウェア要求仕様書に基づいたテスト項目作成手法の提案

ができる。そのため、下流工程での手戻りを防ぐことができ、手戻りの手間やコストを抑えることができると考えられる。

また、今回の実験では、テスト観点到属するキーワードを人手によって要求仕様書から抽出したが、抽出のためのキーワードの選定は機械的に行うことができるため、キーワード抽出の自動化ができると考えられる。自動化により、キーワード抽出の効率が上がるだけでなく、キーワードの見逃しなど人為的なミスを減らすことができるため、テストの効率化および要求仕様書の信頼性の向上に繋がることを見込まれる。しかし、キーワード抽出の自動化を行うためには、曖昧な表現を持つ日本語からテスト観点的キーワードを適切に抽出することが求められるため、キーワードを要求仕様書から抽出するための規則や意味解析の手法の導入などを考察する必要がある。

6. おわりに

本論文では、要求仕様を網羅するテストを行うことを目的として、要求仕様書からテスト観点的キーワードを抜き出して4つのタイプに分類・整理し、それらを組み合わせてテスト項目を作成するテスト項目作成手法を提案した。

テスト項目作成手法とは、要求仕様書から、基本観点、派生観点、組み合わせ観点、期待結果観点のいずれかに属するキーワードを抽出し、そのキーワード群から、派生構造、組み合わせ構造、期待結果構造のテスト項目を機械的に作成する手法である。

キーワードを機械的に組み合わせることでテスト項目を作成することによって、テスト項目数が爆発的に増加するが、テスト項目の再現率が上がることにより、作成したテスト項目によってより多くの要求を網羅することができることを実験によって示した。また、無効なテスト項目も、無効である原因を調べることにより、要求仕様書の要求の記入漏れを発見できる可能性がある。そして、本提案手法により、ソフトウェア開発の上流工程において要求の記入漏れを発見でき、下流工程での手戻りを防ぐことができるため、テスト工程の効率化や要求仕様書の信頼性向上に繋がると考えられる。

今後の課題として、以下を挙げる。

● 膨大なテスト項目数の削減

現在の手法は、テスト観点的キーワードを機械的に組み合わせているため、作成するテスト項目数が爆発的に増加しやすい。特に、基本観点的動詞の抽出は、ソフトウェアの動作以外の動詞も抽出するため、無効なテスト項目も作成してしまう。そこで、テスト観点到属する新たな規則を設けるなどの、作成するテスト項目数の爆発的な増加を抑えつつ、

有効なテスト項目をより多く取得するための手法を考える必要がある。

● テスト項目作成手法の自動化

本提案手法は、機械的にテスト項目を作成することから、手法の自動化が期待できる。そのためには、要求仕様書からテスト観点的に抽出するために、意味解析や日本語読解などの言語理解のための手法を取り入れる必要がある。

参 考 文 献

- 1) “2010年度組込みソフトウェア産業実態調査報告書,” 経済産業省 商務情報政策局, (2010),
http://www.meti.go.jp/policy/mono_info_service/joho/ESIR/index.html
- 2) T.J. Ostrand and E.J. Weyuker, “Using Data Flow Analysis for Regression Testing,” Proc. of the 6th Annual Pacific Northwest Software Quality Conf., pp.233-247 (1988).
- 3) “ISO/IEC 9126:1991, Information technology software product evaluation: Quality Characteristics and guidelines for their use,” International Organization for Standardization / International Electrotechnical Commission, (1991).
- 4) “高信頼化ソフトウェアのための開発手法ガイドブック -予防と検証の事例を中心に-,” 独立行政法人 情報処理推進機構, (2010).
- 5) “話題沸騰ポット要求仕様書 (GOMA-1015 型) 第7版,” 独立行政法人 組込みソフトウェア管理者・技術者育成研究会,
<http://www.sesame.jp/>
- 6) 山本修一郎, “システム運用知識抽出の提案,” 人工知能学会第7回知識流通ネットワーク研究会予稿集 (2010),
<http://www4.atpages.jp/sigksn/conf07/SIG-KSN-007-02.pdf>
- 7) 近江久美子, 永田敦, 伊藤慈朗, 上田克則, 朝永糸子, 森崎一邦, “要求仕様書におけるテストエンジニアの視点を活かした欠陥検出方法の提案,” 日本科学技術連盟 ソフトウェア品質管理研究会 第24年度 (2008年度) 分科会成果報告, (2008).
- 8) 鈴木三紀夫, “テスト方法論 Tiramis,” ソフトウェアテストシンポジウム東京 2009 クローリングパネル講演資料,
<http://www.jasst.jp/archives/jasst09e/pdf/A7-5.pdf>
- 9) R. Bobbu, L. Guo, and S. Mukhopadhyay, “RETNA: From Requirements to Testing in a Natural Way,” Proc. of the 12th IEEE International Requirements Engineering RE'04, pp.262-271 (2004).
- 10) A. Gargantini and C. Heitmeyer, “Using Model Checking to Generate Tests from Requirement Specifications,” Proc. of the 7th European Software Engineering Conf., pp.146-162 (1999).