

マインドマップを適用したソフトウェアテストプロセスの構築

Construction of software test design process utilizing Mind map

主査 : 奥村 有紀子 (有限会社デバッグ工学研究所)
副主査 : 秋山 浩一 (富士ゼロックス株式会社)
副主査 : 堀田 文明 (有限会社デバッグ工学研究所)
研究員 : 来海 大輔 (株式会社インダ)
田中 桂三 (オムロン株式会社)
中嶋 良秀 (株式会社ノーリツ)
平岡 岳 (株式会社リンクレア)
村上 仁 (株式会社ニコン)

研究概要

ソフトウェアのブラックボックステストでは、テスト対象の仕様の明確化が重要な要素である。そのためには機能とテスト観点の抽出が不具合を検出する上での重要な要素である。我々は、SQiP 研究会の分科会活動で実習し、ソフトウェアテストにおける機能とテスト観点の抽出方法の場面で、手軽に利用できかつ観点抽出等に効果があるマインドマップを採用した。ただし、マインドマップで抽出した結果を有効利用するためには、余計な発散を抑え、かつ抽出結果をテストアーキテクチャ設計に容易に利用できるように整理することが重要である。そこで我々は研究活動の中で、マインドマップを使用した機能とテスト観点の抽出方法と、抽出結果をテストアーキテクチャ設計以降に利用できるようにするための整理方法を定義し、さらにこれらの手順を取り込んだソフトウェアテストプロセスを構築した。構築したソフトウェアテストプロセスを実際のソフトウェア開発プロジェクトに適用した結果、マインドマップで新たなテスト観点を抽出することができた。

Abstract: It is an important element to clarify specifications of test target for Software Black box text. To clarify them, it is an important element to extract a function and test point of view to detect software defects. Therefore we selected Mindmap which we took a practical training by subcommittee activity of SQiP study meeting and it was effective to extract function and test point of view easily. However it is important to suppress the unnecessary emission and to arrange the extraction result so that we can use the result of Mindmap effectively. Therefore we defined the rearranging method of output from the Mindmap so that we can make use of them in the post-process after Test Architecture design process of test, and created these procedures as a software test process in our study. As a result of using the software test process that we built for an actual software development project, we were able to extract new test points of view in Mindmap.

1. 本研究の目標と研究課題の設定

1.1. はじめに

ソフトウェアのブラックボックステストにおいて、「入力に対する出力の正しさ」の根拠となるのが、テスト対象の「仕様」である[1]。テスト対象の仕様を明確化するために、機能の抽出が重要な要素である。また、テスト対象の不具合を検出する上で、機能に対するテスト観点[2][6]の抽出も重要な要素である。我々は、SQiP 研究会第 5 分科会の活動でマインドマップ[3]をテスト設計に利用する方法を学習し、全員が機能とテスト観点の抽出を実現できた。よってソフトウェアテストにおける機能とテスト観点の抽出方法としてマインドマップを採用した。その際、機能とテスト観点を抽出した後の整理方法を定義した。そして、これらの手順をソフトウェアテストプロセスとして構築した。また、某組織のソフトウェア開発プロジェクトを対象に、本ソフトウェアテストプロセスを実践し、結果を検証した。本論文では、これらの内容、成果、課題と今後の展開について記載する。

1.2. ソフトウェアテスト設計における問題

我々は、テスト対象の機能とテスト観点を抽出するための手段として、思考・発想法の一つであるマインドマップが適切なツールであると考えた。1.3 に記載した特徴があると知り、また、SQiP 研究会第 5 分科会の活動で、マインドマップをテスト設計に利用する方法を学習[4]し、容易にマインドマップを利用でき、かつメンバーの意思疎通を取りやすいと実感したからである。

一方で、テスト対象の機能とテスト観点を抽出した後、これらの抽出結果が発散したままの状態では、テストアーキテクチャ設計[9]以降へ利用する際にも情報が発散したままとなる。そのため、各機能にテスト観点を組み合わせを実施すると、膨大なテストケースになってしまう。その結果、テストプロセスの中でマインドマップをテストケースの項目出し程度にしか適用できないという問題がある。我々は本問題を解決するために、以下の 2 点が必要と考えた。

- ① テストケースの発散を抑えるために、マインドマップの作成目的を機能とテスト観点ごとに絞り、効率よくアイデアを抽出する。
- ② マインドマップで抽出した結果を、テストアーキテクチャ設計以降への入力として利用できるよう整理する。

上記①②について過去の論文を調査したが、本問題を解決するための方法はなかった。①については、複数の観点到に分けてマインドマップを作成する事例[5] [7]があるが、機能とテスト観点毎に分けていないため、テストケースの発散を抑えるには十分ではない。②については、マインドマップで抽出した結果を FV 表に整理する事例[8]があるが、テストアーキテクチャ設計以降への入力として利用できる整理方法を言及していない。

1.3. マインドマップとは

マインドマップは、1960 年代にトニー・ブザン氏が提唱した、脳の仕組みを取り入れた「考え」や「アイデア」を思考に沿って描いていく図解表現技法である[3][4]。

具体的には、表現したい概念の中心となるキーワードやイメージを図の中央に描き、そこから放射状にキーワードやイメージを繋げていく方法で、下記のような特徴がある。

- (1) 発想力が刺激される
描いているうちに他のブランチとの関連などから新たな発想が生まれやすい。
- (2) 思考経路が残る
思考に沿った記述を行うことにより、「なぜそう考えたか」「何を描いたか」を思い出しやすく、他人にもその考え方やアイデアが容易に理解してもらえる。
- (3) 可視性・可読性に優れている
1 枚の図となっているため、全体を容易に俯瞰でき、思考の漏れや重複を把握しやすい。また、色やイラストを使い、キーワードやイメージなどで構成されているため、一つ一つの繋がりが分かり易い。
- (4) 学習が容易
基本的なルールは単純で、紙とペンがあれば描き始められる。

1.4. 本研究の目標

1.2 の問題を解決するために、マインドマップで抽出したテスト対象の機能とテスト観点を整理する方法を研究した。我々は本研究の目標を、以下のとおり設定した。

「目標：マインドマップを適用したソフトウェアテストプロセスを構築する。」

また、本研究で構築したテストプロセスが、本目標の達成につながるのかを検証するために、某組織のソフトウェア開発プロジェクトに適用した。さらに適用した結果、以下の問題が解決できたのかを検証した。

- 問題解決 1) マインドマップで抽出したテスト観点でテスト設計すれば、テストケースの発散を抑制できるか
- 問題解決 2) マインドマップの抽出内容を整理した一覧表が、テストアーキテクチャ設計以降の入力として十分利用できるか

2. 研究内容

本章では 1.4 に挙げた目標に対する研究内容を記載する。

2.1. 目的別マインドマップの作成

漠然とした視点で作成したマインドマップを付録 3. に掲載する。

この結果から、マインドマップの作成目的を絞らないと、漠然とした視点で進めてしまい、余計な発散が生じるので、効率よくアイデアを抽出できず、有効な結果を得られないと考えた。

そこで我々は、テストプロセス（テストプロセスについては 2.3 で説明する）の「テスト要件定義・要求分析」プロセスの中で、目的別に複数のマインドマップを作成し、機能とテスト観点を抽出する手順を定義した。

(1)機能マインドマップ, (2)利用者要求マインドマップ

テスト対象の機能要件と非機能要件を、対極的な「機能体系重視」と「利用者視点重視」の観点で抽出する。それぞれ抽出観点が異なるので、2つの「機能マインドマップ」と「利用者要求マインドマップ」を作成する。

(3)リスク観点マインドマップ

利用者に多大な損害をもたらす不具合流出防止の為、テスト対象の特徴を考慮し、リスク観点を抽出する。利用者の視点で抽出するが、抽出観点が異なるため、利用者要求マインドマップとは別に作成する。抽出結果は利用者要求マインドマップに含める。

(4)テスト観点マインドマップ

テスト対象が、機能単体だけでなく、製品・システムとして品質上のリスクを低減するため、テストに必要な観点を洗い出す。

各マインドマップの作成目的、説明、作成時のポイントを表 1. に記載する。

表 1. 目的別に作成するマインドマップ

作成目的	マインドマップ	説明	作成時のポイント
機能の抽出	(1)機能マインドマップ *付録 4. に掲載	機能仕様書をベースに、未記載の仕様も含めて、機能を抽出する	【解決すべき課題】 ①機能仕様書の構成は、テストを考慮して作成されていない場合があるので、そのまま転記するとテスト設計の機能分類に適合しない ②後の作業に備えて抽出結果を整理する 【作成時のポイント】 ② テスト設計に最適な分類で機能を抽出する。 ③ 機能分析一覧表(表 2. で説明)と整合性を取る
要求, 非機能要求の抽出	(2)利用者要求マインドマップ *付録 5. に掲載	利用者の視点として、実際にテスト対象物を利用する立場で、必要な要求事項を抽出する	【解決すべき課題】 テスト観点の漏れを防止する 【作成時のポイント】 仕様書ベースで抽出した観点に対し、利用者の潜在要求までを考慮して、要求を抽出する。
リスク観 点の抽出	(3)リスク観点マインドマップ *付録 6. に掲載	利用者に多大な損害をもたらす不具合防止の観点で、必要な要求事項を抽出する	【解決すべき課題】 利用者に多大な損害をもたらす不具合を防止する 【作成時のポイント】 製品・システムの特徴により、リスクとして挙げる。たとえば、不具合発生時の動作・救済動作が不十分などによる。身体への危険、財産損失、機会損失などが生じないか、などの観点で挙げる。
テスト観 点の抽出	(4)テスト観点マインドマップ *付録 7. に掲載	製品・システムの妥当性確認を実施するため、テストに必要な観点を洗い出す。	【解決すべき課題】 機能単体だけでなく、製品・システムの観点でもテスト観点を抽出する。観点の偏り防止する。 【作成時のポイント】 ISO/IEC25010 の品質特性を参考にする。

2.2. マインドマップで抽出した要素の整理

マインドマップの図の性質上、テーマを中心として、抽出した観点をノードとブランチで外側に伸ばした図自体が成果物である。そのため、ノード間の親子関係は分かるものの、どの単位でテストを実施すべきかのイメージがつかめず、テストアーキテクチャ設計以降ではそのまま使えないという問題がある。解決策として、マインドマップで抽出した要素を一覧表で整理する。この際、テストアーキテクチャ設計以降で、テストの実施単位となる機能群と要求群の分類を変更せずに利用できるように整理する。その整理方法のポイントを表 2. に記載する。また、マインドマップと一覧表の関係を付録 12. に掲載する。

表 2. マインドマップで抽出した要素の整理に用いる一覧表

一覧表名称	整理対象	説明	整理方法のポイント
機能分析一覧表 *付録 8. に掲載	機能マインドマップ	<ul style="list-style-type: none"> 機能マインドマップで抽出した要素を整理して、後工程のテストアーキテクチャ設計でその分類のまま利用できるようにまとめる。(この後の作業を分担した場合の齟齬の発生防止) 機能毎に適用するテストタイプを整理するためのインプットに使用する。テストタイプについては、2.3.1.(2)で説明する。 	【解決すべき課題】 ①テストを実施する単位に合わせて分類する。 ②各機能が満足すべき要求を理解する。 【作成時のポイント】 ①複数(3階層程度)に階層化する。その際、テストの実施手順、共通のテスト観点、機能間の関連が深い機能群、または共通の特徴を持った機能群に分類する。 ②各機能の役割を6W2Hで整理し、各機能が満足すべき要求を抽出し、該当機能と要求の関係を表中に記載する。
要求-機能トレーサビリティマトリクス *付録 9. に掲載	利用者要求マインドマップ リスク観点マインドマップ	利用者要求のシナリオとそれを実現している機能の関係をトレースできるようにする。	【解決すべき課題】 利用者要求を過不足なくテストする 【作成時のポイント】 ・利用者要求マインドマップで抽出した要求と、機能マインドマップで抽出した機能の関係を整理するため、縦軸に要求、横軸に機能のマトリクスを作成する。その際テスト実施単位で要求群を分類する。 ・QFDを参考にする。
テストマップ *付録 10. に掲載	テスト観点マインドマップ	各機能とテストタイプを整理したものである テスト観点マインドマップで抽出した要素を整理して、後工程のテストアーキテクチャ設計でその分類のまま利用できるようにまとめる。	【解決すべき課題】 抽出した機能に対して必要なテスト観点を整理する。 【作成時のポイント】 ・機能分析一覧表で整理した機能と、テスト観点の関係を整理するため、縦軸に機能、横軸にテスト観定のマトリクスを作成する。

2.3. テストプロセスの構築

2.1 と 2.2 で挙げた各マインドマップの作成方法と要素の整理方法をテストプロセスとして構築した。テストプロセスのポイントを以下に記載する。

2.3.1. テストプロセスの分類と構築

本研究では、テストプロセスを以下のように分類し、研究対象のプロセスについてアウトプットを定義した。それぞれについて説明する。

(1) 「テスト要件定義・要求分析」プロセス

開発プロジェクトで作成される設計仕様書群やテスト担当者の知識・経験をインプットに、テスト対象の機能要件を整理し、かつテスト対象の利用者要求を分析する。主なアウトプットは表 1. の各種マインドマップと表 2. の一覧表である。

(2) 「テストアーキテクチャ設計」プロセス

テスト要件定義・要求分析で生成された、表 1. の各種マインドマップと表 2. の一覧表をインプットに、テストアーキテクチャを設計する。

アウトプットを表 3. に記載する。

表 3. テストアーキテクチャ設計プロセスのアウトプット

一覧表名称	説明	目的
要求-機能トレーサビリティマトリクス *付録 9. に掲載	<ul style="list-style-type: none"> ・ 2.2. の後に、テストタイプとテスト概要仕様書名を追記する。 ※テスト概要仕様書は、各テストケースの実施概要、テスト技法を記載する。 	利用者要求のシナリオで出てくる機能を整理する。これによりどの利用者要求のシナリオにどの機能を組み合わせてテストすればよいかわかる。 また、登場しない機能がある場合はシナリオが足りないことがわかる。
テストマップ *付録 10. に掲載	<ul style="list-style-type: none"> ・ 2.2. の後に、テスト概要仕様書名を追記する。 	<ul style="list-style-type: none"> ・ 機能ごとに適用するテストタイプを決定し、テスト全体としてやるべき内容の網羅性がわかるようにする。 ・ 抽出したテスト観点をテストタイプに分類して整理する。
テストタイプ一覧表 *付録 11. に掲載	<ul style="list-style-type: none"> ・ ブラックボックステストのうち、テスト実施対象となるテストタイプの一覧表である。 ・ テストタイプは、JSTQB Foundation Level シラバス Version2011[10]では、ソフトウェアの機能や、信頼性、使用性等の非機能的な特性、回帰テストなど、特定のテストの目的に焦点をあてたものであると定義している。これを参考に製品・システムに該当するテストタイプを抽出する。 	<ul style="list-style-type: none"> ・ 同じ目的を持ったテスト観点をまとめて、テストタイプとしてテストをパターン化できるように整理する。 ・ 一緒にテストするものやテストレベルを検討する ・ テスト観点からテストタイプを整理するところがテストアーキテクチャの肝である。

(3) 「テスト詳細設計・実装」プロセス

テストアーキテクチャ設計プロセスのアウトプットである、表 3. の各一覧表をインプットに、テストとして実施しなければならない内容を洗い出し、インプットとアウトプットのイメージをチーム内で共有する。また、選択したテストタイプを元にテストケースを作成する。

(4) 「テスト実施」プロセス

作成したテストケースを使用して、テストを実施する。

2.3.2. プロセス毎の文書化

テスト担当者へのテストプロセスの理解促進のため、プロセス定義を文書化した。特徴は以下のとおりである。

- ・ 大分類したプロセス群の中に、各プロセスの内容と関係を定義する。
- ・ 各プロセスでの必要成果物を定義し、それぞれのプロセスで実施すべき内容を明確にする。
- ・ プロセス毎に、目的、実施内容、インプットとアウトプット、注意点を記載する。

プロセス定義書を付録 1. に掲載する。

2.3.3. プロセスフローダイアグラムの構築

プロセス毎のインプットドキュメント、アウトプットドキュメントを定義し、かつプロセス間の順序と、参照・生成するドキュメントの関係を容易に理解できるように、テストプロセスを 1 枚のプロセスフローダイアグラム（以降 PFD と記載）で表現する。

PFD を付録 2. に掲載する。

3. 研究内容の検証

本章では、「2. の研究内容が、本当に 1.2. の問題解決につながるのか」を確認するために、某組織のソフトウェア開発プロジェクトに対して、本研究で構築したテストプロセスを適用し、以下の問題を解決したかを検証した。

問題解決 1) マインドマップで抽出したテスト観点でテスト設計すれば、テストケースの発散を抑制できるか

問題解決 2) マインドマップの抽出内容を整理した一覧表が、テストアーキテクチャ設計以降の入力として十分利用できるか

3.1. 問題解決 1)の検証

3.1.1. 効果

2.3.2.に記載した通り、某組織のソフトウェア開発プロジェクトに対して、マインドマップ作成を含めたテストプロセスを構築し、文書化した。その中で、それぞれの場面に適合した目的で、複数のマインドマップを作成し、それらを整理するために用いる一覧表の関係を明文化した。その結果、以下に挙げる効果が得られた。

- ・本テストプロセスの中で、目的別に複数のマインドマップに分けて作成することにより、目的以外の観点を含むことがなくなった。また、テストケースの発散の抑制につながる、表3.に挙げた一覧表を作成できた。
- ・テストマップにて、機能に対して必要なテスト観点を整理した。その際、各機能に対して、明らかに該当するテスト観点以外に、「他のテスト観点も当てはまるのではないか。」という視点で横展開ができた。その結果、従来の開発プロジェクトでは抜けていた機能とテスト観点を組み合わせを発見することができた。

本効果事例を付録13.に掲載する。

3.1.2. 課題

マインドマップを作成するにあたり、トップノードを適切に選定しないと発想量が少なくなるリスクがあり抽出の結果に影響を及ぼす。テスト観点マインドマップについては、本プロセスで定義しているとおり ISO/IEC25010 の品質特性をトップノードに置くのが適切と考える。しかし、その他のマインドマップについては適切なトップノードの選定方法が未確立である。

3.2. 問題解決 2)の検証

某組織のソフトウェア開発プロジェクトに対して、マインドマップの抽出結果を整理し、テストアーキテクチャ設計以降への入力として、十分かつ容易に利用できたのかを検証した。結果は表4.のとおり、仮説どおり問題を解決することができた。

表4.マインドマップの検証結果

名称	検証ポイント	対象	検証結果 ○：解決できた △：一部解決できた ×：できなかった
機能分析一覧表	<ul style="list-style-type: none"> ・各マインドマップで抽出した内容を整理できたか ・後工程のテストアーキテクチャ設計で使いやすくまとめることができたか 	・機能マインドマップ	△ <ul style="list-style-type: none"> ・機能マインドマップで抽出した内容を機能分析一覧表に整理した。これにより、機能間のダブリや漏れがないという確認ができた。 ・後工程のテストアーキテクチャ設計時に、機能分類方法をそのまま適用することができ、後戻りなく、テストアーキテクチャ設計に集中することができた。 ・機能マインドマップのトップノードの選定方法が未確立である。トップノードの選定次第では機能の抽出が不十分となる。このため本一覧表で機能を整理しても機能の抽出が漏れる可能性がある。
要求-機能トレーサビリティマトリクス	利用者マインドマップで抽出した結果を、後工程のテストアーキテクチャ設計で利用できたか	・利用者要求マインドマップ	○ <ul style="list-style-type: none"> ・利用者の要求を満足するために関係する機能を整理することによって網羅性の判断が容易になった。例えば、利用者要求をベースとしたシナリオテストで機能間連携を網羅しているかを、本マトリクスを使えば一目で確認可能となった。 ・「要求に対して該当する機能がない。機能に対して合致しない要求がある。」など、要求・機能それぞれの抜け漏れがないかを確認できた。
テストマップ	テスト観点マインドマップで抽出した結果を、後工程のテストアーキテクチャ設計で利用できたか	テスト観点マインドマップ	○ <ul style="list-style-type: none"> ・各機能に対してどのテスト観点でテストすべきかが容易に理解できた。複数テストで作業分担する場合にも効果があった。 ・テストアーキテクチャ設計プロセスで、抽出したテスト観点をテストタイプに分類して整理した。これにより、各機能で実施すべきテストタイプを表現でき、テスト詳細設計・実装の明確なインプットとして利用できた。

4. 目標の到達度と、今後の展開

4.1. 目標の到達度

目標のとおり、マインドマップを適用して、テスト要件定義・要求分析からテストケース作成に至るまでのテストプロセスを構築することができた。それを第三者でも適用できるように、プロセスの文書化、PFDとして図式化することができた。

さらに、マインドマップの作成手順や整理の方法を定義することで、マインドマップをテスト要件定義・要求分析プロセスに利用することができた。マインドマップをテストプロセスに導入した効果を以下に記載する。

4.1.1. 本プロセス適用によるテスト観点抽出効果

以下の場面で、新たなテスト観点を発見できた。

- ・マインドマップと一覧表の照らし合わせにより、ある機能に存在するテスト観点が他の機能のテスト観点にも当てはまるケースを見つけ、テスト観点を抽出漏れを発見できた。
- ・ISO/IEC25010の品質特性をトップノードに置いたことで、製品の潜在要求である障害許容性に関するテスト観点を発見できた。

4.1.2. 本プロセスを適用して効果があるケース

(1)以下の組織には、テスト実施に利用できるもので、テスト対象物の品質向上に効果が期待できる。

- ・テスト要件定義・要求分析、テストアーキテクチャ設計を行っていなかった組織
- ・マインドマップが発散しすぎて、テストケース数が膨大になり、全てのテストケース実施を断念せざるを得なかった組織

(2)本プロセスの適用により、テストタイプを整理し、テストケースの発散を抑えることで、テスト実施の効率化が期待できる。

4.2. 今後の展開

3.1.2.に挙げた通り、マインドマップを作成する上でのトップノードの選定方法が未確立である。今後の展開として、テスト対象による最適なトップノードの選定方法のパターン化を試行し、機能とテスト観点を抽出結果を用いて、効果の検証を行う。

5. 参考文献

- [1] 情報システム用語事典：ブラックボックステスト，URL <http://www.itmedia.co.jp/im/articles/1111/07/news139.html>
- [2] 西康晴，テスト観点に基づくテスト開発方法論 VSTeP の概要，電気通信大学 大学院理工学研究科 総合情報学専攻 経営情報学コース 西研究室，2013
- [3] 池田暁，鈴木三紀夫，「マインドマップから始めるテスト設計」，ソフトウェアテスト PRESS Vol. 3, Vol. 4, Vol. 5，技術評論社
- [4] 鈴木三紀夫，池田暁，「三色ボールペン&マインドマップを用いたテスト設計演習」，2013
- [5] 第23年度ソフトウェア品質管理研究会 第5分科会（マインドマップグループ）直交表とマインドマップを使った効果的なテスト設計」，（財）日本科学技術連盟，2007
- [6] テスト観点とは：Qbook（ソフトウェアに関する品質向上のポータルサイト），URL <http://www.qbook.jp/qptestview/description#anc01>
- [7] 坂本 正，JaSST '09 「マインドマップ活用によるシステムテストの改善」，NPO 法人 ソフトウェアテスト技術振興協会 (ASTER)，2009
- [8] 永田 敦，JaSST '08 東京 「ソフトウェアテスト分析の方法」，NPO 法人 ソフトウェアテスト技術振興協会 (ASTER)，2008
- [9] 吉澤 智美，JaSST '11 東京 「魁!!智美塾 テストアーキテクチャという考え方の提案」，NPO 法人 ソフトウェアテスト技術振興協会 (ASTER)，2011

[10]JSTQB(Japan Software Testing Qualifications Board), テスト技術者資格制度
Foundation Level シラバス 日本語版, Version 2011.J01