

無断使用をお断りします。日科技連出版社

# ソフトウェア テスト技法ドリル

第2版

テスト設計の考え方と実際

秋山浩一[著]

日科技連

## 改訂にあたって

本書が出版されてから、早いもので10年以上も経ちました。その間に思いもかけぬほどの多くの方々に「ドリル本で勉強会をしました」と声をかけていただいたことを大変うれしく思います。

大変うれしいと思う一方で、本書で取り扱った例題やキャプチャー画面が古くなっていたり、リンク先のウェブページが既になくなっていたりすることを読者に申し訳ないと思っていました。

また、JSTQB(ソフトウェアテスト技術者資格認定)では2016年からAdvanced Level テストアナリスト(ALTA)の資格の認定を始めています。その試験ではソフトウェアテスト技法のスキルを評価しています。本書を用いてテスト技法の勉強をされる方もいらっしゃる聞き、ますます改訂の必要性を感じていました。

そのようななか、本書を一緒につくった編集の鈴木兄宏氏から「時代に合わせて改訂するなり、新しい本にするなりしていかないとならないのでは？」と、お声がけいただきました。

改訂版をつくるのなら、章末の練習問題を新しくし、クラシフィケーションツリー技法の説明を追加し、GIHOZの紹介をしようと欲張った関係で、改訂に1年近くもかかってしまいました。

改訂に際して、大変有益なレビューをしていただいた井芹洋輝氏に感謝します。また、改訂のきっかけをくださったのみならず、今回も編集をしてくださった日科技連出版社の鈴木兄宏氏に感謝します。本書がソフトウェアテストを行う人の役に立ちますように願います。

2022年9月

秋山浩一

## まえがき

本書は、配属されて間もない、テスト技術者の卵から、実際にテスト設計を何度かしたことがあり本も数冊読んで勉強した中級テスト技術者で「もっとテストが上手になりたい!」と思っている人を対象としています。

また、TDD(Test-driven Development: テスト駆動開発)を行い、テスト用のプログラムを先に書くという習慣がついているソフトウェア開発者の人で、さらに効果的なテストケースを書いて、TDDを開発のためだけでなくその後の保守フェーズにおいても活用していきたいと考えている人にとっても参考になるようにとの想いで執筆しました。

筆者は、2007年7月に『ソフトウェアテスト HAYST 法 入門』という本を書きました。それは、直交表を使ったソフトウェア組合せテストについて解説した本です。同書は、広く読まれたばかりではなく、私がかつ知っているだけでもJaSST'08九州における安部田章氏の発表をはじめとし、5つのHAYST法を自動化するためのツールが開発され、2010年度の品質工学研究発表大会では坂本秀樹氏から適用効果も出ているとのうれしい報告がありました。

しかし、その一方で各社にお邪魔し、現場の人の声に耳を傾けると「HAYST法をしたらバグがなくなるのですか?」という誤解や、「実は、HAYST法以前の話で困っています。同値分割法と境界値分析くらいは使っていますがほかのテスト技法になると勉強しただけで使っていないのです」という話をよく聞きました。

残念ながら、HAYST法は“銀の弾丸”ではありません。たしかに強力な技法ですが、ほかのテスト技法がHAYST法によって不要になったわけではありません。むしろ、HAYST法のテスト技法としての位置づけを明確にして、他のテスト技法と効果的に組み合わせて使っていく必要があります。

そこで、誰が読んでも理解でき、また、すぐに実践できることを目標にソフトウェアテスト技法を一連の流れで説明した本を書きたくになりました。一連の流れとは、テストの視点・観点といった「点に注意を向ける」といった話から、テストの目的の一つである品質に関する情報を提供する「多次元の品質」までです。少しずつ難しいテスト技法を紹介することで気がついたらテストがうまくなっていたという姿を目指しています。

本書で述べたことは、例えば Boris Beizer の『ソフトウェアテスト技法』を読めば書いてあることが大半です。しかし、同書を読むにはかなり専門的な知識が必要で初級者には敷居の高いものでした。それから、欧米のテスト技術だけではなく、日本人が考えたテストの技法についても、紹介したいという気持ちもありました。

また、テストの効率を上げるために開発されたツールが自由に使えるようにと、インターネット上に公開されているのですが、それらについても紹介し現場ですぐに活用ができるように丁寧に解説したつもりです。

本書がソフトウェアテスト設計を実施する必要がある多くのテストエンジニアや開発者の一助になれば幸いです。

本書を書くにあたり、多くの人に助けていただきました。松尾谷徹氏には、同氏が開発し続けている CFD 法(Cause Flow Diagram)の解説を書くことを許可していただきました。そしてそればかりではなく、書いているうちに疑問となった点について質問させていただき、私の知らなかったポイントについてご教授いただきました。第3章の CFD 法の内容が濃くなっているのはそのためです。

また、鈴木三紀夫氏には、同氏が考案した三色ボールペンによるテスト分析について記載した箇所についてレビューしていただきました。ここでも単なるレビューにとどまらず「私ならこう分析しますが……」と具体的にテスト分析をしてくださいました。その結果は本書に反映されています。

テストツールについては、CEGTestを開発された加瀬正樹氏、PictMasterを開発された鶴巻敏郎氏、stateMatrixを開発された判谷貞彦氏からその紹介について快諾いただきました。これらのツールはいずれもテストの効率を上げるためになくはないものと考えています。

特に、加瀬正樹氏には第3章の原稿をレビューしていただいた際に「XXという動作になるので注意しましょう」と私が書いた箇所について、原稿を直すのではなくCEGTestツール自体を直してしまうという、思ってもみなかった対応をいただき感動しました。

それから、テストの考え方全般について、西康晴氏のアイデアが多く盛り込まれていることも述べておきたいと思います。テストを網羅性とピンポイントの視点で捉えることも西氏から教わりました。また、原稿全体をレビューしてくださった湯本剛氏にも感謝します。

そして、現在、大学院でご指導いただいている古川善吾先生にもお礼を述べたいと思います。特に統計的テストの考え方と活用方法について、ご教授いただきました。

日科技連出版社の鈴木兄宏氏にも感謝します。大幅に遅れている執筆を責めることなくフォローしてくださいました。本書が読みやすくなったのは鈴木氏のおかげです。

最後に、本書の執筆中、さまざまな面で理解し、励まし、協力してくれた友人とそして家族に感謝します。

2010年8月

秋 山 浩 一

## 本書の読み方

本書は、第1章の「点に注意を向ける」から第6章の「多次元の品質」に向けて、点、線、面、立体、四次元、多次元と次元が一つずつ増えるように、徐々に複雑なテスト設計を行うためのテスト技法の解説がされています。

したがって、基本的に順番に読み進めていただくと理解が進むように構成されています。しかし、特定の技法を参照したいというケースもあると思います。そこで、表1に各章とそこで取り扱うテスト技法についてまとめました。

表1の技法に括弧がついているものは、ウェブからダウンロードし、使用できるツールの名前です(ライセンス条項については各ツールを参照ください)。また、ツールの使用方法については、本文に記載しましたので参考にしてください。いずれも、実際のテストの現場で有効にご使用いただけるものと考えています。

それから、本書は例題を解きながら技法を学ぶ構成になっています。本文中の「例題」は、初めはそのまま読み進めていただいてもかまいません。各章末にはその章で学習したテスト技法を使って解く演習問題がついています。こちらは、巻末の解答を見ずにじっくりと取り組んでください。簡単と思える問題にも思わぬ落とし穴があるかもしれません。

なお、ソフトウェアテストは数学のように一つしか答えがないというわけではありません。そのソフトウェアの開発状況やお客様が求める品質レベルによってテストの内容も加減されます。したがって、本書の解答と自分が導いた答えが異なってもがっかりしないでください。解答例の一つと考え参考にし、何か新たな気づきが得られればそれで十分です。

最後に、本書を持ち寄って勉強会を開いてワイワイやるのも楽しいと思いま

表 1 各章で解説しているテスト技法

章	章タイトル／テストタイプ	使用している技法
1	《点に注意を向ける》 ピンポイントテスト	三色ボールペン 「間、対称、類推、外側」の視点 意地悪テスト 不具合モード エラー推測 探索的テスト
2	《線を意識する》 単機能テスト	同値分割法 境界値分析 負荷テスト
3	《面で逃さない》 論理組合せテスト	ドメイン分析テスト クラシフィケーションツリー技法 デシジョンテーブル 原因結果グラフ (CEGTest) CFD 法
4	《立体で捉える》 無則組合せテスト	HAYST 法 (OA.xls) スライド法 ペアワイズ (PICT、PictMaster)
5	《時間を網羅する》 状態遷移テスト	状態遷移図 (GIHOZ) 状態遷移表 N スイッチカバレッジ (stateMatrix) 並列処理テスト
6	《多次元の品質》 シナリオテスト ユーザー受け入れテスト 品質保証のテスト	テストの観点 シナリオテスト 例外シナリオ 受け入れテスト サンプリングテスト 探針テスト (QP.xls) 統計的テスト

す。いろいろな人の考えを聞くと自分が気づかなかった視点や観点が得られ、それだけでもテストが上手になります。

それでは、頑張ってください。

## ソフトウェアテスト技法ドリル第2版 目次

改訂にあたって .....	iii
まえがき .....	v
本書の読み方 .....	ix
<b>第1章 点に注意を向ける</b>	
1.1▶ピンポイントテスト .....	4
1.2▶過去の経験を活かす .....	10
1.3▶本章のまとめ .....	14
演習問題 .....	15
<b>第2章 線を意識する</b>	
2.1▶同値分割法と境界値分析の基本 .....	19
2.2▶同値分割法と境界値分析の応用 .....	23
2.3▶ループ境界 .....	29
2.4▶負荷テスト .....	29
2.5▶本章のまとめ .....	31
演習問題 .....	32
<b>第3章 面で逃さない</b>	
3.1▶ドメイン分析テスト .....	35



<b>3.2</b> ▶クラシフィケーションツリー技法 .....	39
<b>3.3</b> ▶デシジョンテーブル .....	42
<b>3.4</b> ▶原因結果グラフ .....	45
<b>3.5</b> ▶CFD法 .....	79
<b>3.6</b> ▶本章のまとめ .....	88
演習問題 .....	89

## 第4章 立体で捉える

<b>4.1</b> ▶HAYST法 .....	94
<b>4.2</b> ▶ペアワイズ .....	122
<b>4.3</b> ▶本章のまとめ .....	140
演習問題 .....	141

## 第5章 時間を網羅する

<b>5.1</b> ▶状態遷移テスト .....	145
<b>5.2</b> ▶並列処理テスト .....	158
<b>5.3</b> ▶本章のまとめ .....	159
演習問題 .....	160

## 第6章 多次元の品質

<b>6.1</b> ▶人間に対するテストとソフトウェアテスト .....	165
<b>6.2</b> ▶ソフトウェアテストとは何か .....	170
<b>6.3</b> ▶シナリオテスト .....	172
<b>6.4</b> ▶受け入れテスト .....	175
<b>6.5</b> ▶品質保証のテスト .....	176

6.6▶ 本章のまとめ .....	183
演習問題 .....	184
参考文献 .....	185
索引 .....	187



### 演習問題の解答について

演習問題の解答は、日科技連出版社のウェブサイト(<https://www.juse-p.co.jp/>)からダウンロードできます。トップページ上部のタブ [ダウンロード] をクリックすると、検索画面が表示されますので、書名もしくは ISBN を入力してください。

#### 注意事項

1. 演習問題の解答の著作権は、著者にあります。本資料を無断で使用することを禁じます。
2. 著者および出版社のいずれも、本資料をダウンロードしたことに伴い生じた損害について、責任を負うものではありません。

のテストをするときに基本となるテクニックです。

## 3.2▶クラシフィケーションツリー技法

クラシフィケーションツリー技法は、複数の変数(とその値)を木構造で分類して図示することで、テスト対象の変数と値を視覚的に認識するために使用します。また、識別した変数間の組合せテストをするかどうか、単独で確認するだけで良いか、組み合わせる場合のカバレッジは2変数間か、3変数間かといった組合せの方法(組合せ強度)について検討するためにも使用します。

### 3.2.1 クラシフィケーションツリー技法の概要

図3.2は、クラシフィケーションツリーの例です。この図の上部にあるテスト対象の変数を木構造で表現したものをクラシフィケーションツリーと呼びます。木構造で表現することによって、変数(パラメータ)と、その変数がとりう

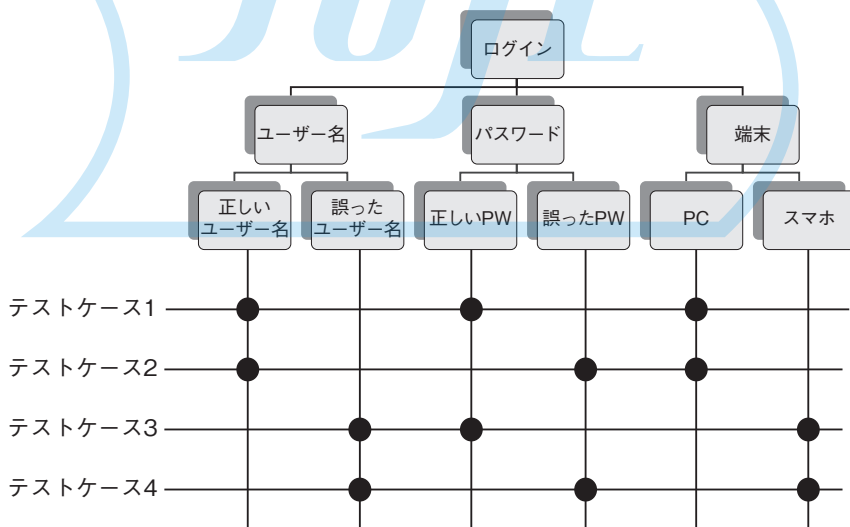


図3.2 ログイン機能のクラシフィケーションツリー

る値(バリュー)の抜け漏れをチェックすることができます。抜け漏れのチェックには、区別の法則を使用します。

### (1) 区別の法則

区別の法則には「一貫性の原則」「相互排除の原則」「一致の原則」「漸進の原則」という4つの原則があります。この4つの原則を木構造の要素に当てはめることによって、要素の抜け漏れを防ぎます。それぞれの原則については図3.3を参照してください。

### (2) クラシフィケーションツリーの要素

前掲の図3.2の木構造部分は3段階になっています。最上位のノードのことを「トップノード(Root)」と呼びます。2段目の「ユーザー名」「パスワード」「端末」は変数ですが、「クラシフィケーション(Classification)」と呼びます。そして最下層の「正しいユーザー名」「誤ったユーザー名」などを「クラス

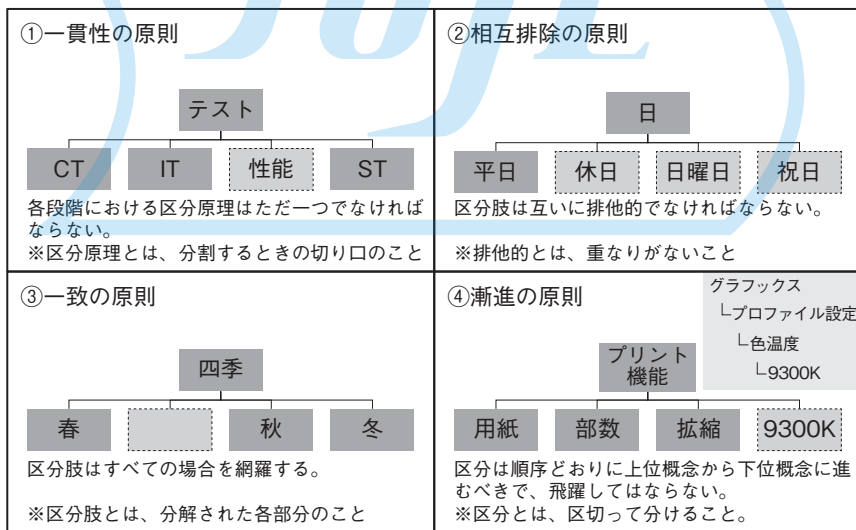


図 3.3 区別の法則

ストするという考え方もあります。しかし、その方針では、すぐにテスト項目数は爆発し、テストしきれなくなります。

そこで、本章では、ソフトウェア自体は第3章と同様に二次元で考えて、そこに組み合わせるべき「要因を選ぶ目線」を追加して立体(三次元)で考えます。「要因を選ぶ目線」はソフトウェアの利用者である「顧客の視点(6W2H)」となります。

## 4.1 ▶ HAYST 法

HAYST 法は、直交表を用いた組合せテスト技法です。第3章のドメイン分析テスト、クラシフィケーションツリー技法、デシジョンテーブル、原因結果グラフ、CFD 法は、システムへの入力や条件に対して複雑な論理関係や順序関係がある場合にテストする手法でした。

一方、HAYST 法は、仕様上は機能と機能の間には関連する規則がないことが前提になっているテストです。機能と機能の間に関連がないことを「直交している」あるいは「無則」といいます。そういった直交している機能を組み合わせたときに「本当に問題が起こらないこと」を効率よく確認する方法が直交表をベースとした HAYST 法です。

本章の冒頭で述べたとおり、ソフトウェアは、既存機能と新規追加機能の間には関係がなく、組み合わせさせて使ってもまったく問題がないように設計されず。ところが、“Nobody’s Perfect. Everybody makes mistakes.” といわれるように、完全な人間は誰一人としていませんから、どこかで意図しない組合せの問題が作り込まれてしまいます。そしてリリース後に「まさか、こんな組合せで問題が起こるとは想定外でした」という言い訳が繰り返されるのです。HAYST 法はこの言い訳をしないために富士ゼロックス(株)で筆者らが考案し、改良を重ねた技法です。

### 4.1.1 HAYST 法を使う意義

表 4.1 は、HAYST 法による簡単なテストです。「画面解像度」「ブラウザ拡大率」「ブラウザの文字サイズ」といった機能を因子、それぞれの機能に含まれている「1024 と 1920」「100% と 125%」「中 と 大」といった選択肢を水準と呼びます。

表 4.1 のテストでは 2 水準の大きさの因子を 3 つ組み合わせたものになっています。1 回目のテストで、画面解像度「1024」と、ブラウザ拡大率「100%」、ブラウザの文字サイズ「中」の組合せをテストし、4 回目のテストで画面解像度「1920」と、ブラウザ拡大率「125%」、ブラウザの文字サイズ「中」の組合せをテストするといったように使います。つまり、行がテストケース、列名が因子、表中のセルが水準にあたります。

ディスプレイの画面解像度と、ウェブブラウザの拡大率と、ブラウザのデフォルトの文字サイズは、それぞれ独立して自由に設定できます。つまり、この 3 つの機能は直交しています。独立しているので本来、組合せテストは不要です。しかし、「本当に独立しているの?」「絶対に?」「命賭けてそう言える!？」と聞かれたら、自信をもって命を賭けられる人はいないと思います。

一般にソフトウェアは各機能に対して、それぞれを独立してつくろうと努力し開発されるものです。しかし、完璧に独立させることは難しく、どこかで意図しない関連性が生まれてしまうものです。そこで、表 4.1 のように(直交表をベースとした)HAYST 法を用いてテストをすることが重要となります。

表 4.1 HAYST 法による簡単なテスト

No.	画面解像度	ブラウザ拡大率	ブラウザの文字サイズ
1	1024	100%	中
2	1024	125%	大
3	1920	100%	大
4	1920	125%	中

◆著者紹介

秋山 浩一(あきやま こういち) 博士(工学)

1962年生まれ。1985年青山学院大学理工学部物理科卒業。同年富士ゼロックス(株)入社。現在、(株)日本ウィルテックソリューション IT コンサルタント

NPO法人ソフトウェアテスト技術振興協会理事、日本ソフトウェアテスト技術者資格認定委員会(JSTQB)ステアリング委員

品質工学会正会員、日本品質管理学会正会員、情報処理学会正会員

【主な著書】

『ソフトウェアテスト HAYST 法 入門』(共著、日科技連出版社)

『事例とツールで学ぶ HAYST 法』(日科技連出版社)

『ソフトウェアテスト講義ノート』(日科技連出版社)

『ソフトウェアテスト入門』(共著、技術評論社)

『基本から学ぶソフトウェアテスト』(共訳、日経 BP 出版)

『ソフトウェアテストの基礎』(共訳、センゲージラーニング)

---

## ソフトウェアテスト技法ドリル【第2版】

テスト設計の考え方と実際

---

2010年10月5日 第1版 第1刷発行

2021年5月10日 第1版 第14刷発行

2022年10月26日 第2版 第1刷発行

著者 秋山 浩一

発行人 戸羽 節文

発行所 株式会社日科技連出版社

〒151-0051 東京都渋谷区千駄ヶ谷5-15-5  
DSビル

電話 出版 03-5379-1244

営業 03-5379-1238

検印  
省略

Printed in Japan

印刷・製本 港北メディアサービス

© Kouichi Akiyama 2010, 2022

ISBN978-4-8171-9766-5

URL <https://www.juse-p.co.jp/>

本書の全部または一部を無断でコピー、スキャン、デジタル化などの複製をすることは著作権法上での例外を除き禁じられています。本書を代行業者等の第三者に依頼してスキャンやデジタル化することは、たとえ個人や家庭内での利用でも著作権法違反です。