

# PictMaster ユーザーズマニュアル

2015年5月7日      第2.2版      v6.2対応

岩通ソフトシステム株式会社

## 本マニュアルの読み方

### 1. とにかく早く使ってみたい方へ

詳しいことは後でよいからとにかく早く使ってみたいという方は、少なくとも第0章と第3章を読んでから使ってください。デフォルトの PictMaster の設定は必要最小限の機能しか動作しない設定になっています。PictMaster を有効に使うためには、後でもよいですから本マニュアル全体を読まれることをお勧めします。

### 2. 生成結果の例について

説明の例として示されている組み合わせ生成結果の表および組み合わせ数は、PictMaster の最小テストケース生成の機能を用いて生成したものです。最小テストケース生成の機能は、生成結果の組み合わせ再現性を保証していません。したがってユーザの方が同じモデルで最小テストケース生成を行った場合、本マニュアルで示されている生成結果とは異なる生成結果となる場合があります。

### 3. PictMaster を業務で使う前に

PictMaster は組み合わせテストをはじめ、複数の種類のテストに対応したテストケースを生成するツールです。本ツールを使いこなすには基礎的なテスト技術を一通り習得していることが必須条件です。

現在は、テスト技術を解説した書籍が何種類か出版されています。基礎的なテスト技術が不足していると感じられた方は、まずこうした書籍などを通じて幅広いテスト技術の知識を学んでから PictMaster を業務で使用されることをお勧めします。

組み合わせテスト技法は数あるテスト技法のひとつにすぎません。組み合わせテストよりも他のテスト技法がふさわしい場合がよくあります。テスト対象の性質に合ったテスト技法を使うようにしましょう。

効果的な組み合わせテストを行なううえで、どのパラメータを組み合わせに含めるのかが最も重要です。パラメータの選定は慎重に行なう必要があります。

### 4. PICT と CIT-BACH の記述分けについて

組み合わせ生成エンジンの PICT と CIT-BACH で内容に違いがない記述については原則として PICT と表記しています。

記述内容が PICT に関するものと CIT-BACH に関するものとに分かれる場合は、分かりやすいように記述文の右上に次のマークを表記しています。

**PICT** PICT に関する記述

**CIT** CIT-BACH に関する記述

PICT と CIT-BACH の機能の違いについては[付録 C](#)を参照してください。

## 更新履歴

版数	更新日	対応 Ver.	更新内容
1.0	2011.6.18	4.3	新規作成
1.1	2011.7.19	4.3.2	0. PictMasterのインストールの章に、Excelのバージョンの違いに起因する問題と対処法についての記述を追記した
1.2	2011.11.28	5.0	3.3 環境設定ボタンの章の内容をPictMaster 5.0 の変更に合わせて改定した。 6.5 ショートカットメニューを使って編集する の章を追加した。
1.3	2011.12.19	5.1	3.5.1 拡張サブモデルの章を追記した。
1.4	2012.1.10	5.2	4.7 ワイルドカードの使用の章で全角文字を“?”で指定する方法を追記した。
1.6	2012.4.23	5.5	3.6 拡張サブモデル（パラメータの重み付け）の章を追加した。 3.7 希望したカバレッジを確保したテストケースを生成するの章を追加した。 4.5 ダミーの値のついての章を削除した。 6.4 要因列挙テストでテストケースを削減するの章を削除した。 付録C 拡張サブモデル（パラメータの重み付け）のアルゴリズムの章を追加した。 付録D 希望する t-way カバレッジを確保したテストケースを生成するアルゴリズムの章を追加した。
1.7	2013.3.11	5.6	4.5 制約表の編集方法の章にショートカットメニューからの記入の説明を追記した。 6.4 ショートカットメニューを使って編集するの章に制約表と結果表のセルへの記入について説明を追記した。
1.8	2013.3.25	5.6	0. PictMasterのインストールの章に【PictMasterを実行するとVBAのエラーとなる場合の対策について】を追記した。
1.9	2013.3.25	5.6	軽微な記述誤りを修正した。
1.10	2013.8.1	5.7	6.4 ショートカットメニューを使って編集するの章でCTLキーを押しながら右クリックでExcelのショートカットメニューを表示するを追記した。 付録C、Dを削除した
1.11	2013.11.1	5.7.2	0. PictMasterのインストール【インストール方法】(1)の記述で、PictMasterをネットワークドライブの割り当てを行っていないサーバー上に置いた場合、ログインしているユーザのマイドキュメントフォルダ内に、生成したファイルが出力される旨の記述に変更した。
1.12	2013.12.10	5.7.3	3.5 サブモデルの章でサブモデルの効果的な使用方法を追記した。 6.4 ショートカットメニューを使って編集するの章でサブモデル欄でもショートカットメニューが使用できることを追記した。
2.0	2014.11.11	6.0	生成エンジン CIT-BACH に対応する追記と変更を行なった。
2.1	2015.1.12	6.1	3.6 拡張サブモデルの章を 3.5 サブモデルの章に統合した。
2.2	2015.5.7	6.2	0. PictMaster のインストール【Excel のバージョンについて】の記述を新しいファイル形式に合わせて変更した。 制約表の指定可能数が 50 から 100 に増加したことに合わせて各記述を修正した。 結果表の指定可能数が 50 から 100 に増加したことに合わせて各記述を修正した。

更新履歴の「対応 Ver.」はこのマニュアルが対応している PictMaster の最初のバージョンを表しています。



## PictMaster 使用規定

以下の使用規定にすべて同意される場合のみ PictMaster を使用することを許可します。

1. PictMaster（以後 本ソフトと表記）はフリーソフトで自由に使用することができますが、著作権は岩通ソフトシステム株式会社にあります。
2. 本ソフトは[オープン・ソフトウェア・ライセンス v2.0](#)に基づき、本ソフト（オリジナル成果物）をもとに派生成果物を作成し、配布することができます。ただし、PICT を使用する限り、Microsoft のライセンス条項により、派生成果物を販売することは禁じられています。同様に PICT を同梱しての配布も禁じられています。
3. 本ソフトを販売して収益を得る行為を禁じます。
4. 本ソフトの著作権表示（Copyright (C) 2008-2014 IWATSU System & Software Co., Ltd.）を読めないようにすることを禁じます。ただし、本ソフトをもとに派生成果物を作成し、配布する場合はこの限りではありません。
5. 本ソフトを使用したことによるいかなる損害に対しても著作権所有者は一切の責任を負いません。
6. この「PictMaster 使用規定」は予告なく変更を行なうことがあります。

## 目 次

0. PictMaster のインストール	- 7 -
1. はじめに	- 9 -
2. PictMaster の仕組み	- 9 -
3. PictMaster の使い方	- 10 -
3. 1 「実行」 ボタン	- 13 -
3. 2 「整形」 ボタン	- 13 -
3. 3 「環境設定」 ボタン	- 14 -
3. 3. 1 原型シートの使い方	- 19 -
3. 3. 2 制約式の最適化	- 22 -
3. 4 値の並び欄への記入のしかた	- 24 -
3. 4. 1 1つの値に複数の名称を与えるエイリアス	- 24 -
3. 4. 2 機能が動作しない無効値テストの方法	- 26 -
3. 4. 3 値の重み付け	- 27 -
3. 5 サブモデル（パラメータの重み付け）	- 29 -
3. 6 希望するカバレッジを確保したテストケースを生成する	- 31 -
4. 制約表への記入のしかた	- 32 -
4. 1 制約に関する用語の定義	- 32 -
4. 2 制約表の構成	- 32 -
4. 3 制約条件と制約対象の指定方法	- 33 -
4. 3. 1 条件付き制約	- 33 -
4. 3. 1. 1 パラメータと値との制約	- 33 -
4. 3. 1. 2 パラメータとパラメータとの制約	- 37 -
4. 3. 2 無条件制約	- 39 -
4. 4 使用できる演算子の一覧	- 42 -
4. 5 制約表の編集方法	- 43 -
4. 6 ワイルドカードの使用	- 43 -
5. 結果表への記入のしかた	- 44 -
5. 1 結果表の構成	- 44 -
5. 2 一致条件の指定方法	- 44 -
5. 3 使用できる演算子の一覧	- 46 -
5. 4 記入上の注意事項	- 46 -
5. 5 結果内容の連結	- 47 -
6. より有効な使い方	- 49 -
6. 1 PictMaster のカスタマイズ	- 49 -
6. 2 エラー／警告メッセージが表示された場合	- 49 -
6. 2. 1 矛盾した制約をすばやく見つけるには	- 50 -
6. 3 画面を分割し制約表を記入しやすくする	- 51 -
6. 4 ショートカットメニューを使って編集する	- 52 -
付録A 仕様	- 54 -
付録B 制限事項	- 54 -
付録C 生成エンジンによる機能の相違	- 55 -
付録D PictMaster でデシジョンテーブルのテストケースを自動生成する	- 56 -

## 0. PictMaster のインストール

### 【PictMaster を使う上で用意するもの】

(1) PICT そのものは

<http://download.microsoft.com/download/f/5/5/f55484df-8494-48fa-8dbd-8c6f76cc014b/pict33.msi>

の URL から入手できます。

あらかじめダウンロードし、インストールしておいてください。インストール先は必ず以下のデフォルトのフォルダ内にインストールする必要があります。

C:\Program Files      64bit OS の場合は C:\Program Files (x86)

(2) Excel2000 以降の Excel。

### 【インストール方法】

(1) PictMaster.zip の圧縮ファイルを開き PictMaster.xls を PC 内の任意の場所に置きます。サーバー上に置くこともできます。ネットワークドライブの割り当てを行っていないサーバー上に置いた場合、生成されたテストケースファイル “a.xls”、モデルファイル (PICT への入力となるファイル) “a.txt” などはログインしているユーザの “マイドキュメント” フォルダ内に作成されます。

(2) PictMaster.zip の圧縮ファイルに同梱されている nkf.exe を PICT がインストールされたフォルダ内にコピーします。

(3) CIT-BACH は JAVA ベースの組み合わせ生成エンジンです。使用するにはバージョン 6 (ビルド 1.6.0) 以降の JAVA がインストールされている必要があります。JAVA がインストールされているか、JAVA のバージョンのチェックおよび最新版のダウンロードは次のサイトで行なうことができます。

<http://java.com/ja/>



(4) PictMaster.zip の圧縮ファイルに同梱されている CIT-BACH のファイル cit.jar を PICT がインストールされたフォルダ内にコピーします。

以上でインストール作業は終了です。

### 【PictMaster を実行すると VBA のエラーとなる場合の対策について】

PictMaster を実行すると次のいずれかのエラーメッセージが表示される場合があります。

「オートメーションエラーです。エラーを特定できません」

「無効なオブジェクト ライブラリです。または定義されていないオブジェクトへの参照を含んでいます」

これらのエラーが発生する直接の原因は、VBA が参照しているファイル **MSCOMCTL.OCX** が参照不可となっているためです。この現象はしばしば Windows Update を行なった結果として発生することがあります。エラーが発生する条件には、Windows Update の内容、OS の種類、Excel のバージョンが関係するため、条件を特定することは困難です。

これらのエラーが発生した場合は、次の対策を実施して下さい。

#### [WindowsXP の場合]

[スタート]→[すべてのプログラム]→[アクセサリ]→[コマンドプロンプト]を選び、次のコマンドを入力し、Enter を押す。

```
regsvr32 C:\¥Windows¥System32¥MSCOMCTL.OCX
```

#### [Windows 7 （32bit の場合）]

[スタート]→[すべてのプログラム]→[アクセサリ]→[コマンドプロンプト]を右クリックし、[管理者として実行]を選び、次のコマンドを入力し、Enter を押す。

```
regsvr32 C:\¥Windows¥System32¥MSCOMCTL.OCX
```

#### [Windows 7 （64bit の場合）]

[スタート]→[すべてのプログラム]→[アクセサリ]→[コマンドプロンプト]を右クリックし、[管理者として実行]を選び、次のコマンドを入力し、Enter を押す。

```
regsvr32 C:\¥Windows¥SysWOW64¥MSCOMCTL.OCX
```

これらの対策を行なっても解決しない場合は、Windows Update を行なってみて下さい。解決する場合があります。

### 【重要な注意点】

PictMaster は日本語が使用できますが、スペースについては、必ず半角スペースを使用してください。

生成エンジンが CIT-BACH の場合は名称にスペースを含めることはできません。また次の半角文字は名称に使用することはできません。

( ) { } [ ] , ;

生成エンジンが PICT の場合は次の半角文字は名称に使用することはできません。

( ) { } [ ] | ~ ,

### 【Excel のバージョンについて】

PictMasterのExcelファイルは、Excel2007からExcel2013まで使用できます。ファイル形式は「Excelマクロ有効ブック (\*.xlsm)」です。

Excel2003以前のバージョンでは使用できませんので注意してください。



## 1. はじめに

**PictMaster** は Pairwise 法(All-Pairs 法ともいう)を採用した組み合わせテストケース生成を行なう Microsoft のフリーソフトである **PICT** (Pairwise Independent Combinatorial Testing Tool) と大阪大学の土屋達弘教授が開発した **CIT-BACH** (Combinatorial Interaction Testing tool with a BDD-Assisted Constraint Handler : シットバック) をより使いやすく、より高機能にした Excel ベースのフリーソフトです。

**PictMaster** を公開する目的は、多くの人が **PICT** と **CIT-BACH** という非常に優れた組み合わせテストツールを Excel 上で簡単に使えるようにすることです。**PictMaster** は完全に無償でかつオープンソースのツールとして提供されます。

**PictMaster** は、用意されている数多くの機能をできるだけ簡単な操作で使えることを目標にして開発されました。特に制約表(組み合わせできない組み合わせの除外を指定する表)は簡単に使えるだけでなく、複雑な多くの制約間の関係を視覚的に分かりやすく表現することができます。**PictMaster** がそなえている数多くの機能の中でも、制約表は最もユニークな機能と言えるでしょう。

## 2. PictMaster の仕組み

**PICT** そのものはコマンドプロンプト上で動作する CUI (キャラクタユーザインターフェース) ベースのアプリケーションです。今となってはコマンドプロンプトになじみのない人が大半です。コマンドプロンプト上で動作する **PICT** に抵抗感を感じる方も少なくないと思います。

Excel でテスト仕様書を作成しているのなら、Excel 上で組み合わせテストケースも生成できたらとても便利になります。これを実現したのが Excel の Book である **PictMaster** です。**PictMaster** は、CUI ベースの **PICT** と **CIT-BACH** に Excel の GUI (グラフィカルユーザインターフェース) ベースの殻をかぶせます。イメージ的には図 2-1 のようになります。

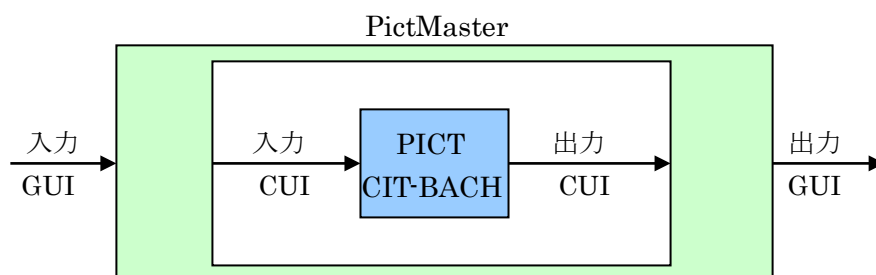


図 2-1 PictMaster のイメージ

図 2-1 に示すように、ユーザからは **PICT**/**CIT-BACH** の存在はまったく見えません。GUI ベースですべての作業を行なうことができます。

**PictMaster** は次に示す 6 つのソフトの連携で動作します。

- (1) Excel の VBA
- (2) コマンドプロンプト
- (3) バッチファイル
- (4) nkf
- (5) **PICT**/**CIT-BACH**

**VBA** (Visual Basic for Application) は、Excel 用のプログラミング言語 (Visual Basic) です。**PictMaster** では **VBA** を使用することで Excel のさまざまな GUI をコントロールします。またモデルファイルの作成、バッチファイルの作成、コマンドプロンプトの起動およびバッチファイルの実行も行ないます。さらにユーザの指定に応じて生成結果の並び替え、罫線を描くなどの処理も行ないます。

コマンドプロンプトのバッチファイルは、nkf の実行と **PICT** または **CIT-BACH** の実行を行ないます。



PictMaster は以下の各部分からなっています。

### 1～7行目 フリーエリア

ユーザが任意にレイアウト可能なエリアです。テストに関する大項目番号、小項目番号、作成日、作成者など、実際にユーザが使いやすいようにレイアウトを決めてください。なお5～7行目は非表示になっているため、そのエリアを使いたい場合は書式メニューから行の再表示を行なってください。

デフォルトのフリーエリアのレイアウトを図3-2に示します。

1	PictMaster				
2	大項目 No.		大項目名		作成日
3	小項目 No.		小項目名		作成者
4					

図3-2 デフォルトのフリーエリアのレイアウト

### 9～58行目 パラメータ欄と値の並び欄

パラメータと、値の並びをカンマ（，）で区切って記入します。パラメータと値の並び欄は50行で固定です。デフォルトでは16行目以降は非表示となっています。値の並び欄には50個までの値を記入することができます。いずれの欄も行を開けずに詰めて記入してください。値にはエイリアス記号（!）、無効値記号（~）および重みづけ指定の（n）を付加することができます。

この欄を編集する際の注意点があります。行の削除、挿入は行なわないで下さい。かわりに行のクリア、コピーと貼り付けで対応してください。コピー元の行と貼り付け先の行が重なると正しく貼り付けされません。コピー先が重ならないようあらかじめ間をあけておくようにしてください。A列の部分を右クリックすると編集専用のショートカットメニューが表示され、行の移動、行の挿入、行の削除および元に戻す、を行なうことができます。パラメータ、値の並び欄の例を図3-3に示します。

パラメータ	値の並び
A	a1, a2, a3
B	b1, b2, b3, b4
C	c1, c2, c3
D	d1, d2, d3, d4
E	e1, e2, e3

図3-3 パラメータ、値の並び欄の例

※パラメータ欄の最初の行（9行目）には文字“ID”で始まる名称は使用しないでください。

### 61～62行目 サブモデル欄

デフォルトの状態では非表示となっており、記入内容は無効です。サブモデル欄は後述する環境設定フォームで「サブモデルを使用」を指定することで表示され、記入内容が有効となります。サブモデル欄の例を図3-4に示します。

サブモデルについては[3.5章](#)で説明します。

サブモデル
A, B, C, 3

図3-4 サブモデル欄の例

## 6 6～9 5 行目 制約表欄

デフォルトの状態では非表示となっており、記入内容は無効です。制約表欄は後述する環境設定フォームで「**制約表を使用**」を指定することで表示され、記入内容が有効となります。制約表は組み合わせできない組み合わせ（制約）を除外するために使用します。制約の内容を表形式で記入します。50 行まで用意されています。デフォルトでは 16 行目以降は非表示となっています。この欄を編集する際は、パラメータ、値の並び欄と同じ注意事項があります。

制約表欄の例を図 3－5 に示します。

制約表			
パラメータ	制約1	制約2	制約3
A	a1	a2, a3	
B	b1, b2	b3, b4	
C			c1
D			d2
E			e2

図 3－5 制約表の例

制約表への記入のしかたは次の[第4章](#)で説明します。

## 1 1 9～2 1 8 行目 結果表欄

デフォルトの状態では非表示となっており、記入内容は無効です。結果表欄は後述する環境設定フォームで「**結果表を使用**」を指定することで表示され、記入内容が有効となります。結果表は組み合わせ内容に応じてあらかじめ期待する結果（結果内容）を記入する表です。100 行まで用意されています。デフォルトでは 16 行目以降は非表示となっています。結果表を使用するとテストケース生成後、自動的にテストケースごとの結果内容欄に期待する結果が設定されます。

結果表の例を図 3－6 に示します。

結果表			
結果内容	A	B	C
aaaとなる	a1	b1	
aaaとなる		b3, b4	c1
bbbとなる	#a1		#c1
cccとなる			

図 3－6 結果表の例

結果表への記入のしかたは[第5章](#)で説明します。

## 2～3行目 「実行」、「整形」、「環境設定」ボタン

デフォルトのレイアウトでは、2～3行目の右端に図3-7に示す3つのボタンがあります。



図3-7 3つのボタン

### 3.1 「実行」ボタン

パラメータ、値の並び欄などに必要な記入を行なった後に、このボタンを押すことでテストケースが“a.xls”という Book 名で作成されます。どのような条件でテストケースを生成するかを「環境設定」ボタンで指定します。

### 3.2 「整形」ボタン

テストケースが生成された後で、値の並び替え、罫線を描く、など指定した条件でテストケースの形を整えることができます。「整形」ボタンをクリックすると、図3-8の例のようなフォームが表示されます。

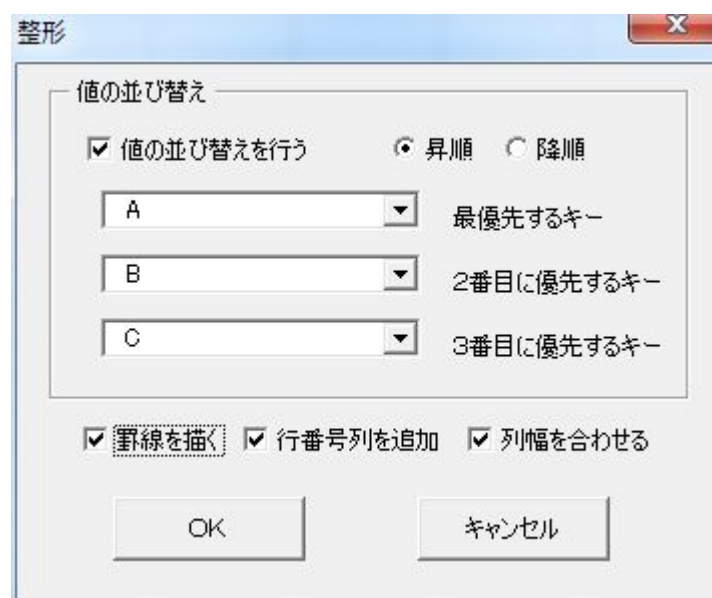


図3-8 「整形」ボタンのクリックで表示されるフォーム

並べ替えのキーには先頭から3つのパラメータが選択され、罫線を描く、行番号を追加、幅を合わせる、のすべてが指定されます。すでに行番号が追加されている場合は「行番号を追加」は指定されません。もちろん、ユーザが好きなように手直することもできます。「OK」ボタンのクリックで処理が行なわれます。

生成結果を整形ボタンで整形した結果の例を表3-1に示します。

表 3－1 整形されたテストケースの例

No.	A	B	C	D	E
1	a1	b1	c2	d1	e2
2	a1	b2	c1	d2	e2
3	a1	b2	c1	d1	e3
4	a1	b3	c2	d1	e1
5	a1	b4	c1	d1	e3
6	a2	b1	c1	d2	e3
7	a2	b2	c2	d1	e2
8	a2	b3	c1	d2	e1
9	a2	b3	c1	d2	e3
10	a2	b4	c2	d2	e2
11	a3	b1	c2	d1	e2
12	a3	b2	c2	d2	e2
13	a3	b3	c2	d2	e2
14	a3	b4	c1	d2	e1
15	a3	b4	c2	d1	e1

### 3. 3 「環境設定」 ボタン

このボタンをクリックすると図 3－9 のフォームが表示されます。

図 3－9 環境設定ボタンのクリックで表示されるフォーム

「自動整形を実行」を指定してテストケースの生成を行なうと、テストケースが生成された後、自動的にテストケースの整形が行なわれます。この際の整形の条件は、図 3－8 と同様な条件で行なわれます。異なる条件でテストケースの整形を行ないたい場合は指定せずに、テストケースが生成されてから「整形」ボタンをクリックして任意の条件を指定してください。

「制約表を使用」を指定すると先頭から 15 行について、100 制約の制約表欄が表示され、記入された制約指定が有効となります。制約表欄は 50 行設けていますので 15 行で不足する場合は残りの行を再表示さ

せてください。詳細は[第4章](#)で説明します。

「**結果表を使用**」を指定すると 15 行の結果表欄が表示され、記入された条件が有効となります。結果表欄は 100 行設けていますので 15 行で不足する場合は残りの行を再表示させてください。詳細は[第5章](#)で説明します。

「**サブモデルを使用**」を指定するとサブモデル欄が表示され、記入されたサブモデルが有効となります。詳細は[3. 5章](#)で説明します。

「**原型シートを使用**」を指定してテストケースの生成を行なうと、すぐ右隣のシートを原型シートとして扱います。詳細は次の[3. 3. 1章](#)で説明します。

すぐ右隣にシートがない、あったとしても原型シートの形式と異なる形式の場合はエラーメッセージが表示されます。原型シートの 1 桁目が**ダラーマーク (\$)** の場合その行は**コメント行**として扱われます。原型シートは空白行を含まず詰めて記入してください。コメント行を除いた最初の行にはすべてのパラメータの名称を記入します。途中の行に空白行があると、それ以降の行は無視されます。

「**モデルファイルを表示**」を指定して、テストケースの生成を行なうと、テストケースが生成された後、PictMaster がパラメータ欄、値の並び欄、制約表などをもとに生成し、PICT に渡したモデルファイル“a.txt”がメモ帳によって表示されます。

「**組み合わせるパラメータ数**」には 1～50 までの数字を指定します。「実行」ボタンのクリックで、指定されたパラメータ数での組み合わせが生成されます。大きな数を指定した場合、テストケース生成に非常に長い時間がかかる場合があることに注意してください。

CIT-BACH の場合に指定できるパラメータ数は 2～5 までです。6 以上の場合は全パラメータ数の指定(全数組み合わせ)のみ可能です。

「**制約式を最適化**」を指定してテストケースの生成を行なうと、PictMasterが制約表から生成する制約式の最適化が行なわれます。パラメータの値の数が多い場合に制約指定が適切でないため、テストケース生成に時間がかかると判断されると、プログラム内部で制約指定の記入形式を最適化し、より短時間で生成が完了するようにします。詳細は[3. 3. 2章](#)で説明します。このチェックボックスは「制約表を使用」が指定されている場合に使用可能となります。この機能はCIT-BACHでは不要であり使われません。

「**ゼロサプレスしない**」を指定してテストケースの生成を行なうと、生成結果のセルの書式が文字列として生成されます。これにより、先頭が 0 で始まる値 (001、010 など) であってもゼロサプレスされることなくそのまま出力されます。ただし、値が文字列として扱われるため、生成結果の並び替えでは値の大小だけでなく、文字の長さも含めて並び替えが行なわれるようになります。



「統計情報を表示」を指定すると、テストケース生成が完了した時点で図 3-10 の例に示す生成回数、生成数およびテストケース生成にかかった経過時間が表示されます。

この指定を行なうと「カバレッジを表示」のチェックボックスが使用可能となります。

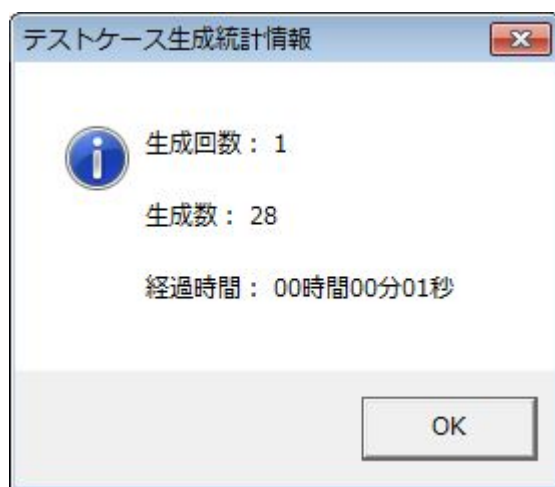


図 3-10 表示される統計情報の例

「カバレッジを表示」を指定すると、テストケース生成時に生成されたテストケースの **n-way** カバレッジ (**n** パラメータ間の組み合わせ網羅率) と **t-way** カバレッジが図 3-11 のように表示されます。**n** は「組み合わせるパラメータ数」で指定した値です。**t** は **n** に +1 した値です。

「組み合わせるパラメータ数」に 5 以上の値が設定されている場合は、カバレッジの表示は行なわれません。

**n-way** カバレッジを算出するために、**n** パラメータ間の組み合わせで 1 回生成を行ないます。また **t-way** カバレッジを算出するために **t** パラメータ間の組み合わせで 1 回生成を行ないます。そのため、大きなモデルでは結果が表示されるまでに時間がかかる場合があることに注意してください。

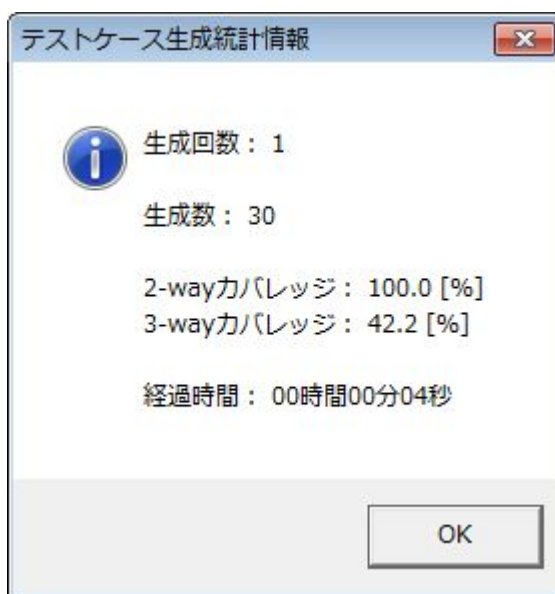


図 3-11 表示されるカバレッジの例

このチェックボックスは「統計情報を表示」を指定すると使用可能となります。



「設定を常時表示」を指定すると環境設定フォームの主な設定内容が、画面右上に常時表示されるようになります。この指定を行なうことで設定内容を確認するためにいちいち環境設定フォームを開かなくて済みます。

「ウインドウ分割ショートカットキー」の入力欄に任意の文字または数字 1 文字を入力しておく、コントロールキーと入力したキーを押すことで PictMaster のウインドウが 2 つ開かれ、上下に整列され、下側のウインドウはパラメータ欄と制約欄との間で分割されます。この機能は多くの制約がある場合に制約表への記入をやりやすくするためのものです。結果表への記入時にも使えます。詳細は [6. 3 章](#) で説明します。

「最小テストケースを生成」を選択すると、テストケース生成の際、ランダムな条件（シード）で PICT を実行し、最もテストケース数の少ないテストケースを生成結果として出力します。「生成回数」で何回テストケース生成を行なうかを指定します。2 から 999 回まで指定できます。デフォルトは 30 回です。

PICT は内部で固有のシード（デフォルトは 0）を使用してテストケースの生成を行なっています。このシードを変えることにより、生成される組み合わせ数が若干違ってきます。

最小テストケース生成が完了した時点で [図 3 - 1 2](#) の例に示す生成回数、最小数、最大数、初期数、最小時シードおよびテストケース生成にかかった経過時間が表示されます（CIT-BACH では最小数、最大数、初期数は表示されない）。

初期数の値は、PICT のデフォルトの生成結果を表します。最小時シードは、最小テストケースを生成したシードを表します。

ランダムな条件で生成した場合、生成されるテストケース数には最大で 10% 程度のバラツキが発生し、多くの場合、最小テストケース生成を行なうことにより 5 % 程度テストケース数を減らすことができます。生成回数を増やすほど、テストケース数を減らせる確率が高くなりますが、ほとんどの場合、30 回程度行なえば充分のようです。1 件でもテストケース数を減らしたい場合は、生成回数に 100 回程度の値を入力して最小テストケース生成を行なってみてください。



図 3 - 1 2 表示される最小テストケース生成統計情報の例

生成エンジンが PICT の場合、最小テストケース生成実行中は、図 3-13 の例に示すプログレスバーが表示されます。生成エンジンが CIT-BACH の場合は、CIT-BACH 内部で最小テストケース生成が行なわれるため、プログレスバーは表示されません。



図 3-13 最小テストケース生成中のプログレスバーの例

「デフォルトのシードで生成」を選択すると、テストケース生成の際、デフォルトのシード（0）で PICT を実行します。

「特定のシードで生成」を選択すると、テストケース生成の際、「シード」欄に記入されたシードで PICT を実行します。0 から 65535 までの値を設定することができます。最小テストケース生成を実行すると、最もテストケース数が少なかったときのシードが「シード」欄に設定されます。

「カバレッジを指定して生成」を選択すると、希望する t-way カバレッジを確保したテストケースを生成結果として出力します。t は「組み合わせるパラメータ数」で指定されている値 n に + 1 した値です。希望する t-way カバレッジを「(t)-way カバレッジ」の欄に記入します。30%～95%の範囲内で指定することができます。希望するカバレッジを確保したテストケースを出力するために、何回かテストケースの生成またはテストケースの変更を繰り返す必要があります。「繰り返し回数」で繰り返しを何回行なうかを指定します。3 回～10 回の範囲内で指定することができます。実行するたびに異なるシードが用いられます。この機能の有効な使い方を 3.6 章で説明しています。

この機能は「組み合わせるパラメータ数」が 1～5 の範囲内である場合に使用することができます。この機能は PictMaster の原型シートの機能を利用しています。環境設定フォームで「原型シートを使用」が指定されている場合、「カバレッジを指定して生成」は指定できません。

原型シートの機能を利用しているため、値の重みづけを行なっている場合は希望するカバレッジが得られないことがあります。

この機能は原型シートの機能を使用するため、Excel のワークシート名の番号が増加します。例えばこの機能を 3 度実行すると、Excel のワークシートは Sheet4 まで使用されます。この状態でユーザが Excel の新規ワークシートを作成するとそのワークシート名は Sheet5 となります。ワークシートの番号を元に戻したい場合は、PictMaster をセーブしたうえで PictMaster を開いたまま再度開いてください。ワークシートの番号がリセットされます。

### 【重要な注意】

希望する t-way カバレッジに低い値を指定した場合、希望する値よりもかなり高いカバレッジとなる場合があります。この現象はデフォルトのシードで生成したテストケースのカバレッジが、希望するカバレッジよりも高い場合に起こります。この場合は希望する低い t-way カバレッジとした場合に n-way カバレッジ 100%を確保できなくなるために、指定した t-way カバレッジは無効となります。

「デフォルトのシードで生成」を指定して生成した場合のカバレッジを確認することで、この現象が起きているかどうか分かります。

値の重み付けを行なっている場合は、希望するカバレッジが得られない場合があります。これは「カバレッジを指定して生成」の機能が原型シートの機能を使用していることからくる制限事項です。

値が無効値を含む場合は正しいテストケースが得られません。これは生成されたテストケースの後半部分に集中して無効値が出現するという PICT の仕様による制限事項です。

繰り返し処理中は、図 3-14 に示すプログレスバーが表示されます。

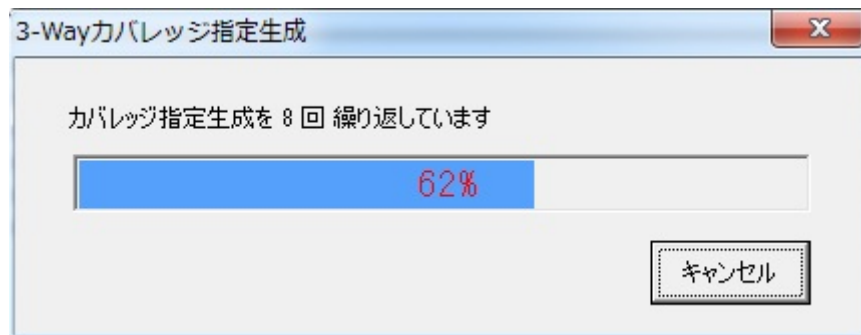


図 3-14 カバレッジを指定して生成中のプログレスバーの例

「PICT」を選択すると生成エンジンに PICT が使用されます。

「CIT-BACH」を選択すると生成エンジンに CIT-BACH が使用されます。

### 3. 3. 1 原型シートの使い方

原型シートには 2 つの使用方法があります。

- (1) 前に使用したモデルを変更する必要がある場合、以前のモデルで作成したテストケースを再利用して、できるだけ少ない変更で新しいテストケースを作成します。
- (2) 生成されるテストケースに必ず含まれるべきである**重要な組み合わせを指定**します。指定された組み合わせで出力が初期化され、次に残りの組み合わせが生成されます。

環境設定フォームで「原型シートを使用」を指定してテストケースの生成を行なうと、**すぐ右隣のワークシートを原型シートとして扱います**。すぐ右隣にワークシートがない、あったとしても原型シートの形式と異なる形式の場合はエラーメッセージが表示されます。

原型シートの 1 桁目が**ダラマーク (\$)** の場合その行は**コメント行**として扱われます。原型シートは空白行を含まず詰めて記入してください。

原型シートには 65534 件までのテストケースを記入できます。原型シートは PICT によって生成されたテストケースと同じフォーマット（行番号は除く）を使用します。コメント行を除いた最初の行にはすべてのパラメータ名を記入します。PictMaster のパラメータ欄と同じパラメータ名がすべて記述されている必要があります。次の行以降は組み合わせられた値の並びを記入します。値が空白のパラメータがあってもかまいません。

#### 【重要な注意】

- (a) 原型シートが現在のモデルにない値を含んでいる場合、エラーとなります。
- (b) 原型シートの値の行が現在の制約指定のどれかに違反する場合、その行は無視される（PICT の場合）か、エラーとなります（CIT-BACH の場合）。
- (c) 原型シートを使用する場合、**値の重み付け（3. 4. 3 章）は正常に動作しません**。値の重み付けを行なって生成した結果を原型シートとして使用する場合は、PictMaster の値の並び欄から値の重み付けを取り除いてテストケースの生成を行なうことを推奨します。
- (d) 原型シートの組み合わせが 1 行ずつ PICT に読み込まれます。読み込んでいる途中ですべての n パラメータ間の組み合わせが網羅された場合、PICT はそれ以上原型シートを読み込もうとはしな

くなります。ただし、現在のモデルが原型シートにない値を含んでいる場合は、原型シートの組み合わせ全体が読み込まれます（PICT の場合のみ）。

(e) この機能は「カバレッジを指定して生成」とは同時には使用できません。

原型シートの使用例を以下に示します。

#### 例 1：新しい値を追加する

	A	B	C
1	A	B	C
2	a1	b2	c2
3	a1	b1	c1
4	a3	b2	c1
5	a2	b1	c2
6	a2	b2	c1
7	a3	b1	c2

図 3－15 原型シート of 例（その 1）

パラメータ	値の並び
A	a1, a2, a3
B	b1, b2, b3
C	c1, c2

図 3－16 新しいモデル of 例（その 1）

新しいモデルでは、パラメータ B に値 b3 が追加されています。

表 3－2 新しい値が追加されたテストケース

No.	A	B	C
1	a1	b2	c2
2	a1	b1	c1
3	a3	b2	c1
4	a2	b1	c2
5	a2	b2	c1
6	a3	b1	c2
7	a3	b3	c1
8	a1	b3	c2
9	a2	b3	c1

新しく生成されたテストケースには、原型シートの内容がそのまま流用され、追加した値 b3 に関する組み合わせが追加されています。これにより既存のテストケースでテストした後で仕様変更などにより、新しい値を追加してテストを行う必要が生じた場合、追加された値に関する組み合わせだけテストすればよい場合もあります。（注：表 3－2 は値の並べ替えを行っていません）

## 例 2：新しいパラメータを追加する

	A	B	C	D
1	A	B	C	D
2	a1	b2	c2	
3	a1	b1	c1	
4	a3	b2	c1	
5	a2	b1	c2	
6	a2	b2	c1	
7	a3	b1	c2	

図 3-17 原型シートの例（その 2）

パラメータ	値の並び
A	a1, a2, a3
B	b1, b2
C	c1, c2
D	d1, d2, d3

図 3-18 新しいモデルの例（その 2）

新しいモデルではパラメータ D が追加されています。この場合、原型シートにも追加するパラメータ D の名称が記述されている必要があります。

表 3-3 新しいパラメータが追加されたテストケース

No.	A	B	C	D
1	a1	b2	c2	d3
2	a1	b1	c1	d2
3	a3	b2	c1	d1
4	a2	b1	c2	d1
5	a2	b2	c1	d2
6	a3	b1	c2	d2
7	a2	b1	c1	d3
8	a3	b1	c2	d3
9	a1	b2	c1	d1

新しく生成されたテストケースには、原型シートの内容がそのまま流用され、追加したパラメータ D に関する組み合わせが追加されています。（注：表 3-3 は値の並べ替えを行っていません）

原型シートに新しく追加したいパラメータの名称を記入しておくだけで新しいパラメータを追加したテストケースを簡単に生成することができます。

## 例 3：必ず含まれなければならない組み合わせを指定する

	A	B	C
1	A	B	C
2	a1	b1	c1
3	a1	b1	c2
4	a1	b2	c1
5	a1	b2	c2

図 3-19 原型シートの例（その 2）

パラメータ	値の並び
A	a1, a2, a3
B	b1, b2
C	c1, c2

図 3-20 モデルの例

原型シートでは値 a1 について、他のパラメータ B と C のすべての値と組み合わせられています。このような、あるパラメータの特定の値についてのみ、ほかのすべてのパラメータが持つ値と組み合わせるような特殊または複雑な組み合わせは原型シートを使って指定します。その他に 3 つ以上の特定の値の組み合わせを含めたい場合も原型シートを使って指定します。

原型シートの値は歯抜けのように空白のセルがあってもかまいません。

表 3-4 指定した組み合わせが含まれたテストケース

No.	A	B	C
1	a1	b1	c1
2	a1	b1	c2
3	a1	b2	c1
4	a1	b2	c2
5	a2	b1	c2
6	a2	b2	c1
7	a3	b1	c1
8	a3	b2	c2

新しく生成されたテストケースには、原型シートの内容がそのまま流用され、その他のパラメータの組み合わせが追加されています。この例のように、制約条件では指定できない複雑で特殊な組み合わせでも、原型シートで指定することにより、簡単に扱うことができます。

### 3.3.2 制約式の最適化

PICT

組み合わせ生成エンジンの PICT は、組み合わせできない組み合わせを指定するのに、IF 文に似たスクリプト言語（制約式）を用います。PictMaster では、制約表への記入を PICT が理解できる制約式に変換します。PICT の特性として、値の個数が多いパラメータに制約の指定を行なうと、制約の指定方法によって組み合わせの生成時間に大きな差が出ます。「制約式の最適化」を指定すると、PictMaster は制約指定が行なわれているパラメータの値の個数を調べ、8 個以上の場合で、制約式の最適化の処理が有効な条件である場合、制約指定をプログラム内部で変更し、最も生成時間が短くなるような制約式を生成します。

制約指定の違いによる生成時間の違いを示すために図 3-21 のモデルを使用することにします。使用した PC 環境 は、CPU が PentiumD 2.8GHz、RAM 1 GB、OS は Windows XP Professional SP3 です。

パラメータ	値の並び
A	1, 2, 3, 4, 5, 6, 7, 8, 9, 10
B	1, 2, 3, 4, 5, 6, 7, 8, 9, 10
C	1, 2, 3, 4, 5, 6, 7, 8, 9, 10
D	1, 2, 3, 4, 5, 6, 7, 8, 9, 10
E	1, 2, 3, 4, 5, 6, 7, 8, 9, 10

図 3-21 生成時間比較用のモデルの例

このモデルに次の制約指定を行なった場合の生成時間を示します。これらの制約指定は異なる記述です



がいずれも等価な制約指定です。生成時間は最小テストケース生成を30回行った場合の経過時間です。

(1) 制約指定その1

制約表		
パラメータ	制約1	制約2
A	1, 2, 3, 4, 5	
B	1, 2, 3, 4, 5	1, 2, 3, 4, 5
C		1, 2, 3, 4, 5
D		
E		

制約式最適化=False 生成時間 = 7 秒

制約式最適化=True 生成時間 = 4 秒

(2) 制約指定その2

制約表		
パラメータ	制約1	制約2
A	1, 2, 3, 4, 5	
B	#6, 7, 8, 9, 10	1, 2, 3, 4, 5
C		#6, 7, 8, 9, 10
D		
E		

制約式最適化=False 生成時間 = 3 秒

制約式最適化=True とはならない。(すでに最適化の形式となっている)

(3) 制約指定その3

制約表		
パラメータ	制約1	制約2
A	#6, 7, 8, 9, 10	
B	1, 2, 3, 4, 5	#6, 7, 8, 9, 10
C		1, 2, 3, 4, 5
D		
E		

制約式最適化=False 生成時間 = 7 分 20 秒

制約式最適化=True 生成時間 = 4 秒

(4) 制約指定その4

制約表		
パラメータ	制約1	制約2
A	#6, 7, 8, 9, 10	
B	#6, 7, 8, 9, 10	#6, 7, 8, 9, 10
C		#6, 7, 8, 9, 10
D		
E		

制約式最適化=False 生成時間 = 7 秒

制約式最適化=True 生成時間 = 3 秒

図 3-22 制約式最適化の有無と生成時間

以上の結果から、制約指定が(3)の場合に制約式最適化の効果が最も顕著であることが分かります。

それ以外では（２）の場合、最適化が行なわれないことがわかります。この制約指定では最適化が不要なためです。制約式最適化の処理が行なわれるのは以下の条件をすべて満たす場合です。

1. 環境設定フォームで「制約式を最適化」が指定されている。
2. 制約条件と制約対象のパラメータに属する値の個数が共に 8 個以上である。
3. 制約対象で指定された値の個数が 2 個以上である。
4. 制約条件が逆制約か、または制約対象が順制約である。

制約指定の意味については[第 4 章](#)を参照してください。

PictMaster は制約式の最適化が指定されると、1から4の条件をすべて満たす制約欄がある場合に、プログラム内部でその制約欄の記述形式を（２）の 制約指定その 2 の形式に変換します。したがって、環境設定フォームで「制約式を最適化」が指定されていても、実際には最適化が行なわれない場合があります。また制約式の最適化が行なわれたとしても、多くの制約指定の一部についてだけ行なわれた場合は、生成時間が短縮されない場合もあります。

パラメータとパラメータの比較を行なう制約指定は最適化の対象外です。

制約式の最適化が行なわれた場合は、制約表の制約指定とは異なる制約式が生成されることになるため、万が一、PICT 内部でエラーを検出した場合、エラーメッセージとして表示される制約式と、記述されている制約指定が対応しないことがあることに注意してください。

「制約式の最適化」を指定した場合、「統計情報を表示」が同時に指定されていると、表示される統計情報に制約表の最適化が実際に行なわれたかどうかが表示されます。

### 3. 4 値の並び欄への記入のしかた

値の並びをカンマ（，）で区切って記入します。値の前に特定の記号を付加することで特殊な処理を行なうことができます。特定の記号にはパイプ（|）とチルダ（~）があります。パイプはひとつの値に複数の名称を与えるエイリアスで使用します。チルダは相互に組み合わせ不可の値を指定する無効値テストで使用します。値の後ろに半角括弧（）で囲まれた数字を付加することで、その値が他の値より多く組み合わせに出現するようになる値の重み付けを行なうことができます。

これらの記号はパラメータおよび値の名称には使用できません。

#### 3. 4. 1 1つの値に複数の名称を与えるエイリアス

エイリアスは、1つの値に複数の異なる名称を与える機能です。まったく同一とは言えないがほぼ同一と考えることのできる複数の値に対して、同値分割の考え方を適用し、エイリアス機能を用いることができます。これにより組み合わせ生成の際に 1 つの値として扱われ、組み合わせ完成後、1つの値はエイリアスで与えた複数の名称に戻されます。

エイリアスを使用することによって生成される組み合わせの数を減らすことが可能です。Pairwise 法で生成される組み合わせの数は最も多くの値を持つパラメータ P1 と、P1 と同じか次に多くの値を持つパラメータ P2 の、それぞれの値の個数を  $v1n$ 、 $v2n$  とすると、これらを積算した値か、それよりもやや多い値になります。生成される組み合わせ（テストケース）のおおよその数  $TCn$  は次式で表されます。

$$TCn = (v1n * v2n + \alpha) * m$$

ここで  $\alpha$  の値は、パラメータ P2 の次に多くの値を持つパラメータ P3 以降が持つ値の個数  $v3n$ 、 $v4n$ ... によって異なります。この数が  $v2n$  と等しいかほとんど同じ場合は、多くの場合  $\alpha$  は比較的大きな値になります。逆に  $v2n$  より小さい場合は、 $\alpha$  は小さな値になります。この場合多くは  $\alpha$  の値は 0 になります。このことから、エイリアスを適用するパラメータは、最も多くの個数の値を持つパラメータを対象にしたほうが、組み合わせ数を削減する効果が最も大きくなります。逆に、比較的少ない個数の値しか持たないパラメータに適用しても効果がありません。mは制約の数が 0 の場合 1 です。制約の数が増えるに従って



徐々に増加しますが、増加の程度は緩やかです。

エイリアスを使用したモデルの例を図3-23に示します。この例では OS 種別と HD 容量および HD インターフェースの組み合わせをテストします。このモデルでは OS 種別の5個の値から、Windows 系3個をエイリアスの記号“|”で1つにまとめ、全体で3個にしています。エイリアスでまとめた場合、先頭の値の名称を使用することで制約などを記述することができます。

パラメータ	値の並び
OS種別	Windows Vista   Windows XP   Windows 2000, Linux, Mac OS X
HD容量	250GB, 500GB, 750GB
HDインターフェース	USB2.0, IEEE1394, eSATA

図3-23 エイリアスを使用したモデルの例

エイリアスを使用した組み合わせ生成結果を表3-5に、エイリアスを使用しなかった場合の組み合わせ生成結果を表3-6に示します。

表3-5 エイリアスを使用した場合

No.	OS 種別	HD 容量	HD インターフェース
1	Linux	250GB	IEEE1394
2	Linux	500GB	USB2.0
3	Linux	750GB	eSATA
4	Mac OS X	250GB	USB2.0
5	Mac OS X	500GB	eSATA
6	Mac OS X	750GB	IEEE1394
7	Windows 2000	250GB	eSATA
8	Windows Vista	750GB	USB2.0
9	Windows XP	500GB	IEEE1394

表3-6 エイリアスを使用しなかった場合

No.	OS 種別	HD 容量	HD インターフェース
1	Linux	250GB	eSATA
2	Linux	500GB	USB2.0
3	Linux	750GB	IEEE1394
4	Mac OS X	250GB	eSATA
5	Mac OS X	500GB	IEEE1394
6	Mac OS X	750GB	USB2.0
7	Windows 2000	250GB	USB2.0
8	Windows 2000	500GB	IEEE1394
9	Windows 2000	750GB	eSATA
10	Windows Vista	250GB	USB2.0
11	Windows Vista	500GB	eSATA
12	Windows Vista	750GB	IEEE1394
13	Windows XP	250GB	IEEE1394
14	Windows XP	500GB	eSATA
15	Windows XP	750GB	USB2.0

この例ではエイリアスを使用した場合、使用しなかった場合に比べてテストケース数が3分の2未満に減少しています。値の数の多いパラメータで同値と考えることのできる値がある場合は、エイリアスを積極的に使ったほうがよいでしょう。

エイリアスで1つにまとめた複数の値のうち、後述する制約表や結果表で指定できるのは先頭の値の名

称のみです。

#### ※ 注意事項

エイリアスとして1つにまとめた値の数が多い場合は、生成結果に一度も現れない値が存在する可能性があります。これはエイリアスを含まないパラメータに属する値が少ない場合に発生します。エイリアスの値すべてを組み合わせに使わなくてもすべての組み合わせが網羅されてしまうからです。本来、エイリアスは同値とみなせるものを1つにまとめたものですから出現しない値があっても直ちに問題とは言えませんが、こうした性質があることは知っておいたほうがよいでしょう。

### 3. 4. 2 機能が動作しない無効値テストの方法

PICT

組み合わせテストでは通常、機能が動作しない無効値を含まないようにする必要があります。テストケースに無効値を含むと機能が動作しなくなり、他のパラメータと組み合わせの意味がなくなります。

ここでは無効値を含むテストの実行を目的としている場合について説明します。1つのテストケースに無効値を2つ以上含むと1つの無効値についてのテストとなり、残りの無効値についてのテストが行われなくなり、不完全なテストとなります。機能が動作しない場合の組み合わせでは、無効値どうしの組み合わせが行われなくする必要があります。

PICTには無効値テストという機能があり、無効値どうしの組み合わせが生成されないようにすることができます。値の並び欄で値の前に半角記号“~”を置くことで無効値を指定します。

図3-24に無効値を含むモデルの例を示します。このモデルではFAXと通信回線の組み合わせをテストします。FAXとは通信できない無効値である電話機が含まれています。さらにFAX使用に制限があるIP外線も無効値としています（実際には多くの場合、支障なく使用できます）。

パラメータ	値の並び
発信端末	FAX, ~電話機
通信回線	アナログ, ISDN, ~IP外線
着信端末	FAX, ~電話機

図3-24 無効値を含むモデルの例

表3-7 無効値テストの生成結果

No.	発信端末	通信回線	着信端末
1	~電話機	ISDN	FAX
2	~電話機	アナログ	FAX
3	FAX	~IP外線	FAX
4	FAX	ISDN	~電話機
5	FAX	ISDN	FAX
6	FAX	アナログ	~電話機
7	FAX	アナログ	FAX

表3-7が生成結果です。この生成結果には無効値どうしの組み合わせがありません。この例では無効値の場合でも無効値以外のすべてのペアを組み合わせています。そこまでの徹底したテストが不要な場合は、制約定義で各無効値1つに1つのテストケースのみが生成されるように指定することもできます。

値が複数の名前を持つエイリアスの場合、最初の名前に記号“~”を付けます。制約の記述で無効値を指定する場合は、記号“~”は省略します。

### 3. 4. 3 値の重み付け

特定の値を重点的にテストしたい場合、重み付けの機能が役に立ちます。重み付けを使用すると指定された値がより多くテストケースに現れるようになり、その値についてのカバレッジが向上します。重み付けは重点的にテストしたい値の右側に半角の括弧（ ）で数値を付加します。

図 3－2 5 に重み付けの例を示します。

パラメータ	値の並び
A	a1(2), a2, a3
B	b1, b2, b3, b4
C	c1, c2, c3(3)

図 3－2 5 重み付けの例

括弧内に記入できる数値は 2～1 0 です。基本的には記入された数値をかけた分だけ他の値より多く組み合わせに出現するようになります。例えば c3(3) を記入した場合、値 c3 は他の値 c1、c2 より 3 倍多く出現するようになります。ただし、生成された組み合わせで重複する組み合わせが存在する場合、重複する組み合わせは 1 つを残し削除されて最終的な生成結果となります。

図 3－2 5 で重複する組み合わせを含んだ生成結果を表 3－8 に示します。

表 3－8 重複する組み合わせを含んだ生成結果

No.	A	B	C
1	a1	b1	c3
2	a1	b1	c3
3	a1	b1	c3
4	a1	b2	c1
5	a1	b2	c2
6	a1	b3	c3
7	a1	b3	c3
8	a1	b4	c1
9	a1	b4	c2
10	a1	b4	c3
11	a2	b1	c1
12	a2	b2	c3
13	a2	b3	c2
14	a2	b3	c3
15	a2	b4	c3
16	a3	b1	c2
17	a3	b2	c3
18	a3	b2	c3
19	a3	b3	c1
20	a3	b4	c3

表 3－8 でパラメータ C の各値の個数は、c1 が 4 個、c2 が 4 個、c3 が 12 個です。この時点で c3 は他の値より 3 倍多く出現しています。これは重み付けの指定どおりです。しかし、網掛けした組み合わせが重複しています。組み合わせの重複が発生するのは、1 つの値を重み付けにより 3 倍多く用いて組み合わせを生成しているため、他のパラメータの値の数が少ないと、同じ組み合わせが生成されるためです。

PictMaster は、重複した組み合わせのうち 1 つを残して他の組み合わせを削除します。そのため重複した組み合わせが存在する場合は、結果的に重み付けで指定した数値より少ない重み付けとなります。

最終的な生成結果を表 3－9 に示します。なお、表 3－8 とは異なる生成実行のため、組み合わせ内容は一部異なります。

表 3－9 重複分を取り除いた最終的な生成結果

No.	A	B	C
1	a1	b1	c1
2	a1	b1	c3
3	a1	b2	c3
4	a1	b3	c2
5	a1	b3	c3
6	a1	b4	c1
7	a1	b4	c2
8	a2	b1	c3
9	a2	b2	c2
10	a2	b3	c1
11	a2	b3	c3
12	a2	b4	c3
13	a3	b1	c2
14	a3	b2	c1
15	a3	b3	c3
16	a3	b4	c3

最終的な生成結果では、重複していた 4 個の組み合わせが削除され、16 個の組み合わせとなりました。この結果では、パラメータ C の各値の個数は、c1 が 4 個、c2 が 4 個、c3 が 8 個です。重み付けを行なって最小テストケース生成を行なうと、環境設定フォームで「統計情報を表示」が指定されていれば、図 3－26 の統計情報が表示され、“重複削除後の数”によって重複した組み合わせがいくつあったかが分ります。

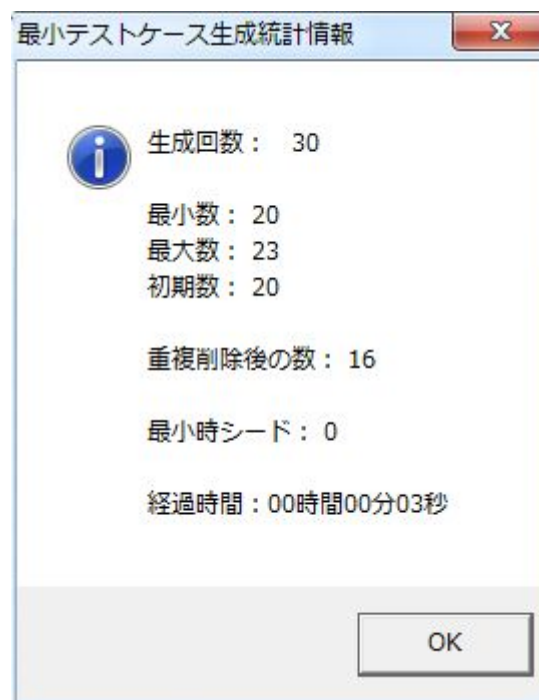


図 3－26 重み付けのあるモデルの生成統計情報の例

重み付けの数値が正確に反映された生成結果が得られるのは、他のパラメータの数が多い場合か、他のパラメータで値の数が多い場合です。この場合は重複が起らず、生成されるテストケース数も重み付けを行なわない場合と比べてそれほど増加することはありません。そうでない場合、重み付けの数値は目安的な意味合いを持ちます。

重み付けを行なうと値の数が実質的に多くなります。入力できる値の個数は 50 個までです。重み付けを行なう場合は実質的な値の個数が 50 個を超えないようにしてください。

値の重み付けと原型シートを同時に使用した場合、値の重み付けが正常に行なわれません。値の重み付けと原型シートは同時に使用しないことを推奨します。

※ PICT 自体が備えている重み付けの機能は使用できません。

### 3. 5 サブモデル（パラメータの重み付け）

この章は PICT と CIT-BACH で内容が異なります。

PICT

すべてのパラメータがすべて同じ重要度を持つというケースはそれほど多くはありません。PICT ではサブモデル定義を使用することで、特に重要と考えられる限定したパラメータについて異なるパラメータ組み合わせ数を指定することができます。サブモデル欄を表示させるには環境設定フォームの「サブモデルを使用する」を指定します。

サブモデルの記入形式を図 3-27 に示します。

<パラメータ 1>, <パラメータ 2>, … , <組み合わせるパラメータ数（2 以上）>

図 3-27 サブモデルの記入形式（PICT）

サブモデルの対象としたい複数のパラメータ名をカンマで区切って記入します。最後に組み合わせるパラメータ数を記入します。この数は 2 から、指定したパラメータの数までの任意の値が記入できます。環境設定の「組み合わせるパラメータ数」が 1 の場合、セミコロン (;) で区切ることにより、1 つの行に複数のサブモデルを記入することができます。環境設定の「組み合わせるパラメータ数」が 2 以上の場合では、指定できるサブモデルの数は 1 つのみです。いずれの場合もサブモデルの組み合わせるパラメータ数は環境設定の「組み合わせるパラメータ数」の値よりも大きくなければなりません。

サブモデルで指定された複数のパラメータは、「組み合わせるパラメータ数」の数のパラメータ間の組み合わせが生成されます。例えば、この数が 3 の場合、指定された複数のパラメータのみ 3 パラメータ間の組み合わせが生成されます。

テストを行なっているとテストケース数を調節したい場合がよくあります。2 パラメータ間の組み合わせを網羅したテストケース数では多すぎる場合がそうです。こうした場合はパラメータの組み合わせは行わず、値を列挙したテストケースとする方法があります。この方法では環境設定フォームの「組み合わせるパラメータ数」に 1 を指定することで値を列挙したテストケースを得ることができます。

値を列挙したテストでは少なすぎるが、2 パラメータ間の組み合わせとしたテストでは多すぎる場合、サブモデルを使用して特定のパラメータのみ、2 パラメータ間の組合せとすることでテストケース数を適切に調節することができます。

CIT-BACH では、サブモデルを指定することで、指定したいいくつかのパラメータのみに限定して全数組み合わせとすることができます。

サブモデルの記入形式を図 3-28 に示します。

**<パラメータ 1>, <パラメータ 2>, ...**

**図 3-28** サブモデルの記入形式 (CIT-BACH)

サブモデルの対象としたい複数 (3 つ以上) のパラメータ名をカンマで区切って記入します。指定されたパラメータのみ全数組み合わせとなります。セミコロン (;) で区切ることで、1 つの行に複数のサブモデルを記入することができます。

PICT の場合と異なり、組み合わせるパラメータ数は記入しません。記入するとエラーとなりますので注意してください。

### 3. 6 希望するカバレッジを確保したテストケースを生成する

PictMaster では、希望するカバレッジを確保したテストケースを生成することができます。3パラメータ間の組み合わせとするとテストケース数が多くなりすぎて問題となる場合、例えば3-way カバレッジを80%としたテストケースでは、生成されるテストケース数を大幅に削減することが可能となります。

例としてパラメータが15個あり、それぞれの値の数が10, 8, 8, 6, 6, 6, 4, 4, 4, 4, 2, 2, 2, 2, 2であるモデルの場合、3-way カバレッジを5%刻みで変化させた場合のテストケース数を次に示します。

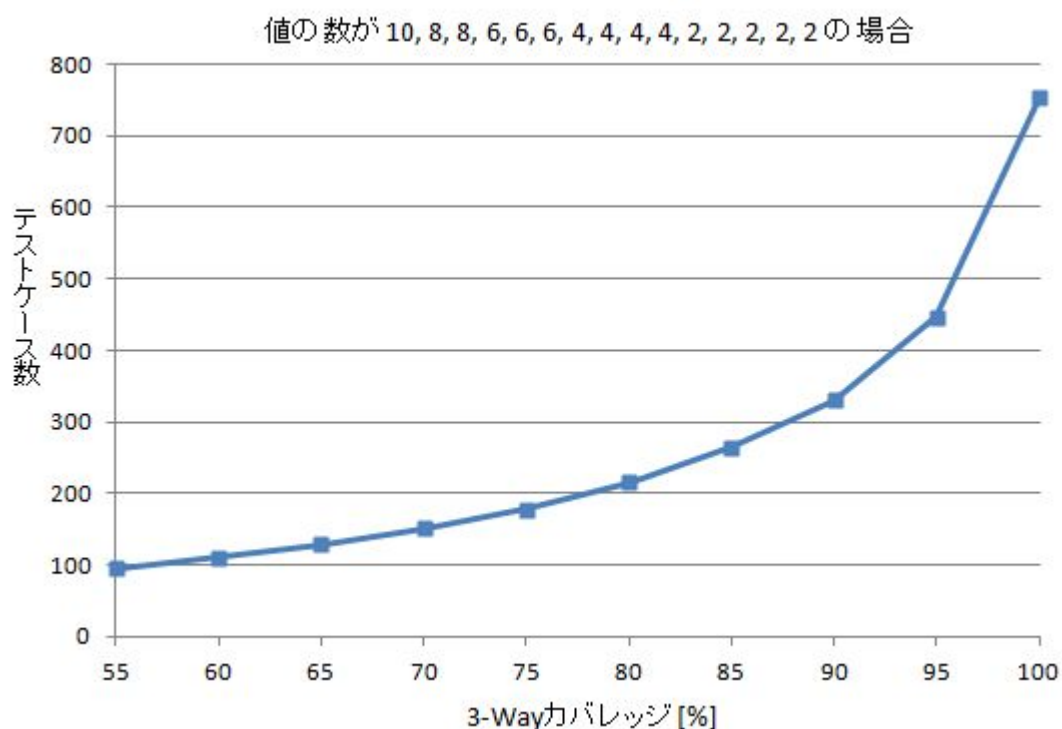


図3. 29 3-way カバレッジを変化させた場合のテストケース数

3-way カバレッジを80%とした場合のテストケース数は216件となりました。これは2パラメータ間の組み合わせを網羅したテストケース数の2.4倍であり、3パラメータ間の組み合わせを網羅したテストケース数の28.6%にあたります。この程度の増加で80%という高い3-way カバレッジを網羅することができるのであれば、特に重要と考えられるテスト対象についてはテストを実施するだけの価値があるといえるでしょう。

希望するカバレッジを確保したテストケースを生成する機能は、2-way から5-way までのカバレッジを指定することができます。環境設定フォームの「組み合わせるパラメータ数」で指定された値  $n$  に+1した  $t$ -way のカバレッジが確保されます。例えば「組み合わせるパラメータ数」に1を指定した場合は、1-way カバレッジが100%確保され、2-way カバレッジが「希望するカバレッジ」で指定された値で確保されます。

## 4. 制約表への記入のしかた

### 4. 1 制約に関する用語の定義

制約表の説明をする前に制約に関する用語の定義を明確にしておきます。

**制約**とは、組み合わせることのできないパラメータと値の組み合わせがあることです。この制約を if 関係式 then 関係式 の形式で記述した式を**制約式**といいます。制約式によって、組み合わせできない組み合わせを除外し、組み合わせできる組み合わせだけを残すことができます

**関係式**は、パラメータと値、またはパラメータとパラメータを比較するために = 、< など**演算子**として用います。PICT はこの制約式を入力として制約に対応した組み合わせを生成します。

if と then との間の関係式を**制約条件**といい、then 以降の関係式を**制約対象**と言います。関係式の演算子が等号 (=) のとき**順制約**といい、関係式の演算子が不等号 (<) のとき**逆制約**といいます。制約条件が成立するとき、制約対象で指定された組み合わせとなります。

以上の用語を使って制約式を記述すると以下のとおりとなります。

if 制約条件 then 制約対象 … 制約条件が真のとき、制約対象の組み合わせとなる。

以上の制約式を**条件付き制約式**といいます。制約条件のある制約式はすべて条件付き制約式です。これに対して、制約条件がなく、制約対象のみからなる制約式を**無条件制約式**といいます。無条件制約式ではパラメータのみが対象となります。基本的な無条件制約式の記述を以下に示します。

パラメータ 演算子 パラメータ

無条件制約はパラメータの値の如何にかかわらず、常に成立する制約です。

### 4. 2 制約表の構成

未記入の制約表を図 4-1 に示します。

制約表			
パラメータ	制約1	制約2	制約3

図 4-1 未記入の制約表

図 4-1 でパラメータの列には9行目からのパラメータ欄の内容をコピーして貼り付けます。環境設定フォームを開いてOK ボタンをクリックすることで自動的に張り付けを行なうこともできます。ただし**制約がない場合はパラメータ欄のコピーは不要です**。制約 1、2、3、～ は、基本的には1つの制約式が1つの制約に対応します。パラメータ欄で指定されたパラメータの値のうち、任意の値を制約欄に記入します。また別のパラメータそのものを記入することもできます。

パラメータは 50 個まで、制約は 100 個まで記入できます。パラメータ欄は間に空白行を置かずに詰めて記入してください。PictMaster はパラメータ欄、値の並び欄および制約表欄の記述内容を、a. txt というモデルファイルに出力し、PICT に入力データとして渡します。この際、PictMaster は制約表の記入内容を PICT が理解できる制約式に変換します。



### 4. 3 制約条件と制約対象の指定方法

制約条件と制約対象で、記述できる形式に差はありません。以降の説明で制約条件と制約対象の例として挙げられている記述は、制約条件と制約対象のどちらでも記述可能です。

#### 4. 3. 1 条件付き制約

**制約条件**とする制約欄は白色以外の**任意の色で塗りつぶ**してください。塗りつぶされた制約欄に記入された値またはパラメータが制約条件となります。

**制約対象**とする制約欄は塗りつぶさないでください。白色で塗りつぶしても塗りつぶしなしとして扱われます。制約欄に記入された値またはパラメータが制約対象となります。

制約欄に複数の値を記入する場合はそれぞれの値をカンマ(,)で区切ります。

先頭に**シャープ(#)**をつけると逆制約となり、記入した値以外の値を意味します。この場合もカンマで区切って複数の値を記入することができます。

先頭に**グレートザン(>)**、**レスザン(<)**をつけることで値の大小比較ができます (PICTのみ使用可)。あるパラメータの値に“>”と“<”をつけることで任意の範囲の値を指定することができます。この場合はカンマ(,)で区切られた二つの値の AND 条件となります。値が数字だけの場合は問題ありませんが、数字と文字が混在する場合は全体が文字と見なされ、文字コードでの大小比較になります。

エイリアスの値を指定する場合は、先頭の値の名称を使用します。

**パラメータそのものを記入する場合は**、順制約の場合は先頭に**イコール(=)**をつけます。逆制約の場合は**エクスクラメーション(!)**を付けます。複数のパラメータを指定する場合はそれぞれのパラメータをカンマ(,)で区切り、それぞれのパラメータにイコール(=)またはエクスクラメーション(!)を付けます。値を記入する場合と異なりますので注意してください。パラメータそのものを記入する場合はパラメータ欄と制約欄のパラメータ双方で少なくとも一部が同じ値を含んでいる必要があります。

制約条件でもなく制約対象でもない場合、制約欄は空白とします。

制約が1つもない場合は制約表のパラメータ欄は空白でもかまいません。

#### 4. 3. 1. 1 パラメータと値との制約

この章では、パラメータの値がある特定の値の場合に発生する制約の指定方法を説明します。

モデルが図4-2で、制約表が図4-3の場合、生成されるモデルファイル“a.txt”の制約式の部分はリスト4-1となります。

リストでは最初に PICT の制約式、次に CIT-BACH の制約式を示します。生成結果は PICT のものです。

パラメータ	値の並び
A	a1, a2, a3
B	b1, b2, b3
C	c1, c2, c3

図4-2 モデルの例 (その1)

制約表		
パラメータ	制約1	制約2
A	a1	#a1
B	b1	#b1
C		

図4-3 制約表の例 (その1)

リスト4－1 制約式の例（その1）

```

if ([A] = "a1" )
    then ([B] = "b1" ) ;
if ([A] <> "a1" )
    then ([B] <> "b1" ) ;

(if (== [A] a1)
    (== [B] b1))
(if (<> [A] a1)
    (<> [B] b1))

```

生成結果の例を表4－1に示します。

表4－1 生成結果の例（その1）

No.	A	B	C
1	a1	b1	c1
2	a1	b1	c2
3	a1	b1	c3
4	a2	b2	c1
5	a2	b2	c3
6	a2	b3	c2
7	a3	b2	c2
8	a3	b3	c1
9	a3	b3	c3

図4－3の制約指定の意味は次の通りです。

**制約1**：パラメータAの値が a1 の場合、パラメータBの値で組み合わせ可能な値は b1 である。

**制約2**：パラメータAの値が a1 以外の場合、パラメータBの値で組み合わせ可能な値は b1 以外である。

表4－1の生成結果を見ると、パラメータAの値が a1 の場合、パラメータBの値は b1 のみであり、パラメータAの値が a1 以外の場合、パラメータBの値は b1 以外となっています。これは図4－3の制約指定の通りです。

図4－2のモデルで制約表が図4－4の場合、生成されるモデルファイル“a.txt”はリスト4－2となります。

制約表		
パラメータ	制約1	制約2
A	a1,a2	a3
B	b1	#b1
C		

図4－4 制約表の例（その2）

リスト 4-2 制約式の例 (その 2)

```

if ([A] = "a1" or [A] = "a2" )
    then ([B] = "b1" ) ;
if ([A] = "a3" )
    then ([B] <> "b1" ) ;

(if (or (== [A] a1) (== [A] a2))
    (== [B] b1))
(if (== [A] a3)
    (<> [B] b1))

```

生成結果の例を表 4-2 に示します。

表 4-2 生成結果の例 (その 2)

No.	A	B	C
1	a1	b1	c1
2	a1	b1	c2
3	a1	b1	c3
4	a2	b1	c1
5	a2	b1	c2
6	a2	b1	c3
7	a3	b2	c1
8	a3	b2	c2
9	a3	b2	c3
10	a3	b3	c1
11	a3	b3	c2
12	a3	b3	c3

図 4-2 のモデルで制約表が図 4-5 の場合、生成されるモデルファイル “a.txt” はリスト 4-3 となります。

制約表		
パラメータ	制約1	制約2
A	a1,a2	a3
B	b1	#b1
C	#c2, c3	c1

図 4-5 制約表の例 (その 3)

リスト 4-3 制約式の例 (その 3)

```

if ([A] = "a1" or [A] = "a2" ) and ([B] = "b1" )
    then ([C] <> "c2" and [C] <> "c3" ) ;
if ([A] = "a3" )
    then ([B] <> "b1" ) and ([C] = "c1" ) ;

```

```
(if (and (or (== [A] a1) (== [A] a2)) (== [B] b1))
    (and (<> [C] c2) (<> [C] c3)))
(if (== [A] a3)
    (and (<> [B] b1) (== [C] c1)))
```

生成結果の例を表 4－3 に示します。

表 4－3 生成結果の例（その 3）

No.	A	B	C
1	a1	b1	c1
2	a1	b2	c3
3	a1	b3	c2
4	a2	b1	c1
5	a2	b2	c2
6	a2	b3	c3
7	a3	b2	c1
8	a3	b3	c1

これまでの例で分かるようにある制約の 1 つの欄に複数の値が記入されている場合は、それぞれの値は基本的には OR 条件になります。これに対してある制約の異なる行に値が記入されている場合は、それぞれの値は AND 条件になります。これはパラメータそのものが記入された場合も同様です。

**制約条件**として異なるパラメータの値を OR 条件で指定したい場合は、異なる制約に制約条件として異なるパラメータの値を記入し、制約対象は同じにします。

**制約対象**として異なるパラメータの値を OR 条件で指定したい場合は、すぐ右側の制約欄に制約条件として同じパラメータの値を記入し、制約対象に異なるパラメータの値を記入します。図 4－6 にこの場合の制約表を示します。このとき生成される制約式をリスト 4－4 に、生成結果の例を表 4－4 に示します。複数の制約条件がある場合はすべての制約条件が同一である必要があります。

制約表			
パラメータ	制約1	制約2	制約3
A			a1
B	b1	b3	b3
C	c1	c2	

図 4－6 制約表の例（その 4）

リスト 4－4 制約式の例（その 4）

```
if ([B] = "b1" )
  then ([C] = "c1" ) ;
if ([B] = "b3" )
  then ([C] = "c2" ) or ([A] = "a1" ) ;

(if (== [B] b1)
    (== [C] c1))
(if (== [B] b3)
    (or (== [C] c2) (== [A] a1)))
```

表 4－4 生成結果の例（その 4）

No.	A	B	C
1	a1	b1	c1
2	a1	b2	c1
3	a1	b2	c2
4	a1	b3	c1
5	a1	b3	c3
6	a2	b1	c1
7	a2	b2	c3
8	a2	b3	c2
9	a3	b1	c1
10	a3	b2	c3
11	a3	b3	c2

この例では制約 2 と制約 3 の制約条件が 1 つの制約式として統合され、制約対象が OR 条件となっています。制約対象がいくつあっても同じ制約では AND 条件となり、異なる制約間では OR 条件となります。ただし隣り合う制約の制約条件の欄を異なる色にすると制約式の統合は行われず、2 つの異なる制約式となります。

#### 4. 3. 1. 2 パラメータとパラメータとの制約

次に制約欄にパラメータを指定する場合を示します。このときのモデルを図 4－7、制約表を図 4－8、生成される制約式をリスト 4－5、生成結果の例を表 4－5 に示します。

パラメータ	値の並び
A	a1, a2, a3
B	1, 2, 3
C	1, 2, 3
D	1, 2, 3

図 4－7 モデルの例（その 2）

制約表		
パラメータ	制約1	制約2
A	a1	a3
B		=C
C	!B	
D		

図 4－8 制約表の例（その 5）

リスト 4－5 制約式の例（その 5）

```

if ([A] = "a1" )
    then ([C] <> [B] ) ;
if ([B] = [C] )
    then ([A] = "a3" ) ;

(if (== [A] a1)
    (<> [C] [B]))
(if (== [B] [C])
    (== [A] a3))

```

表 4－5 生成結果の例（その 5）

No.	A	B	C	D
1	a1	1	3	3
2	a1	2	1	2
3	a1	3	2	1
4	a2	1	2	2
5	a2	2	3	1
6	a2	3	1	3
7	a3	1	1	1
8	a3	2	2	2
9	a3	2	2	3
10	a3	3	3	2

パラメータを指定する場合は、指定する側と指定される側のパラメータの値に一部でも**同じ値が含まれている必要があります**。また、指定する側と指定される側の値の種類（文字列か数値）が一致しなければなりません。1つの欄に複数のパラメータを記入することもできます。この場合はそれぞれのパラメータの前にイコール (=) またはエクスクラメーション (!) を付加し、カンマ (,) で区切ります。各パラメータは OR 条件での指定となります。

1つの欄に AND 条件で複数のパラメータを指定する場合は演算子の前に**アンパサンド(&)**を付加します。このときのモデルを図 4－7、制約表を図 4－9、生成される制約式をリスト 4－6、生成結果の例を表 4－6 に示します。

制約表		
パラメータ	制約1	制約2
A	a1	#a1
B	=C, &=D	!C, !D
C		
D		

図 4－9 制約表の例（その 6）

#### リスト 4－6 制約式の例（その 6）

```

if ([B] = [C] and [B] = [D] )
    then ([A] = "a1" ) ;
if ([B] <> [C] or [B] <> [D] )
    then ([A] <> "a1" ) ;

(if (and (== [B] [C]) (== [B] [D]))
    (== [A] a1))
(if (or (<> [B] [C]) (<> [B] [D]))
    (<> [A] a1))

```

表 4－6 生成結果の例（その 6）

No.	A	B	C	D
1	a1	1	1	1
2	a1	2	2	2
3	a1	3	3	3
4	a2	1	2	3
5	a2	2	3	1
6	a2	3	1	2
7	a3	1	3	2
8	a3	2	1	3
9	a3	3	2	1

#### 4. 3. 2 無条件制約

無条件制約には制約条件がありません。そのため、**無条件制約は常に成立する制約です**。制約表に記入できるのはパラメータのみです。制約欄に演算子付きで記入することで、2つのパラメータ間の制約を指定します。

使用できる演算子には、**イコール (=)**、イコールと逆の意味を表す**エクスクラメーション (!)** があります。1つの制約欄に**カンマ (,)** で区切って複数のパラメータを指定することができます。この場合、それぞれの **OR** 条件となりますが、2つめ以降のパラメータの先頭に**アンパサンド (&)** を置くことで **AND** 条件とすることができます。

モデルを図 4－1 0、AND 条件で逆制約の制約表を図 4－1 1、生成される制約式をリスト 4－7、生成結果の例を表 4－7 に示します。

パラメータ	値の並び
A	a1, a2, a3
B	1, 2, 3
C	1, 2, 3
D	1, 2, 3

図 4－1 0 モデルの例（その 3）

制約表		
パラメータ	制約1	
A		
B	!C	
C	!D	
D	!B	

図 4-11 制約表の例（その 7）

リスト 4-7 制約式の例（その 7）

([B] <> [C] ) and ([C] <> [D] ) and ([D] <> [B] ) ;

(and (<> [B] [C]) (<> [C] [D]) (<> [D] [B]))

表 4-7 生成結果の例（その 7）

No.	A	B	C	D
1	a1	1	3	2
2	a1	2	1	3
3	a1	3	2	1
4	a2	1	2	3
5	a2	2	3	1
6	a2	3	1	2
7	a3	1	3	2
8	a3	2	1	3
9	a3	3	2	1

この例では 1 つの式になっていますが、記入する制約を別にとすると、異なる独立した式になります。その場合でも意味的には同一の制約です。

OR 条件で順制約の制約表を図 4-12、生成される制約式をリスト 4-8、生成結果の例を表 4-8 に示します。OR 条件を指定するには無条件制約の右側の制約で、先頭にプラス (+) を置きます。

制約表			
パラメータ	制約1	制約2	制約3
A			
B	=C		
C		+D	
D			+B

図 4-12 制約表の例（その 8）



リスト4－8 制約式の例（その8）

([B] = [C] ) or ([C] = [D] ) or ([D] = [B] ) ;

(or (== [B] [C]) (== [C] [D]) (== [D] [B]))

表4－8 生成結果の例（その8）

No.	A	B	C	D
1	a1	1	1	2
2	a1	2	2	3
3	a1	3	3	1
4	a2	1	2	1
5	a2	2	3	2
6	a2	3	1	3
7	a3	1	3	3
8	a3	2	1	1
9	a3	3	2	2

1つの制約欄に複数のパラメータを記入した例として、モデルを図4－12、制約表を図4－13、生成される制約式をリスト4－9、生成結果の例を表4－9に示します。

パラメータ	値の並び
A	a1, a2, a3
B	1, 2, 3
C	1, 2, 3
D	1, 2, 3
E	1, 2, 3
F	1, 2, 3

図4－12 モデルの例（その4）

制約表	
パラメータ	制約1
A	
B	=C, !D
C	
D	!E, &=F
E	
F	

図4－13 制約表の例（その8）

リスト4－9 制約式の例（その9）

（[B] = [C] or [B] <> [D] ） and （[D] <> [E] and [D] = [F] ）；

（and （or （== [B] [C]）（<> [B] [D]））（and （<> [D] [E]）（== [D] [F]）））

表4－9 生成結果の例（その9）

No.	A	B	C	D	E	F
1	a1	1	1	3	1	3
2	a1	1	3	2	1	2
3	a1	2	2	2	3	2
4	a1	3	2	1	2	1
5	a2	1	3	3	2	3
6	a2	2	2	3	1	3
7	a2	2	3	1	3	1
8	a2	3	1	2	3	2
9	a3	1	1	1	3	1
10	a3	1	2	2	3	2
11	a3	2	1	1	2	1
12	a3	3	3	3	1	3

#### 4. 4 使用できる演算子の一覧

関係式で使用できる演算子はパラメータと値、パラメータとパラメータで異なります。表4－10に使用できる演算子を示します。#を除いていずれの演算子もカンマ（,）で区切って複数回記述することができます。条件の欄は1つの制約欄に値またはパラメータを複数記述した場合にOR条件となるのかAND条件となるのかを表します。

表4－10 使用できる演算子

関係式の対象	演算子	条件	説明
パラメータと値	（指定しない）	OR	パラメータに属する値を表します。=と同じ意味です。
	#	AND	記述した値を除いた残りの値を表します。任意の1つの値について先頭に記述しますが値は複数記述することができます。
	>, <, >=, <=	AND	値の大小関係を表します（PICTのみ使用可）。
パラメータとパラメータ	=	OR	パラメータ間の値が等しいことを表します。
	!	OR	パラメータ間の値が異なることを表します。
	&=	AND	パラメータ間の値が等しいことをAND条件で表します。パラメータを複数記述した場合に、2つめ以降のパラメータの前に記述します。
	&!	AND	パラメータ間の値が異なることをAND条件で表します。パラメータを複数記述した場合に、2つめ以降のパラメータの前に記述します。
	+	(OR)	無条件制約で異なる制約でOR条件となることを表します。無条件制約の右側の制約で無条件制約の最初のパラメータの前に記述します。

表中の演算子（+）の条件欄が(OR)となっているのは、異なる制約間の条件を表しているためです。これらの演算子と同じ記号をパラメータおよび値の名称の先頭に使用することはできません。

## 4. 5 制約表の編集方法

制約表の編集方法について説明します。

Excel のメニューからの制約表の行の削除や挿入は行なわないでください。制約表の行数が変化すると次に続く結果表欄の行位置がずれてしまいます。

行の削除や挿入を行ないたい場合はその 1 つの行の 1 列目を右クリックしてください。追加、削除、元へ戻すなどの編集専用のショートカットメニューが表示され、メニューを選択することで行の削除や挿入を行なうことができます。

制約の削除や挿入を行ないたい場合は制約のタイトル欄を右クリックしてください。追加、削除、元へ戻すなどの編集専用のショートカットメニューが表示され、メニューを選択することで制約の削除や挿入を行なうことができます。

制約表への値およびパラメータの記入はショートカットメニューを使ってワンタッチで行なうことができます。結果表への記入も同じようにワンタッチで行なうことができます。

詳細は [6. 4 章](#) を参照してください。

## 4. 6 ワイルドカードの使用

PICT

制約表に記入する値にはワイルドカードを使用することができます。ワイルドカードとは、「任意の文字」を意味する特殊な文字です。「\*」は 0 個以上の任意の文字を、「?」は 1 個の任意の半角文字を意味します。

例えば「A\*」は名称が「A」で始まるすべての値を意味します。「A」だけでも含まれます。「???A」は半角 4 文字で最後が「A」で終わるすべての値を意味します。全角 1 文字は「??」で指定します。

ワイルドカードは、制約表では条件付制約の値の名称のみに使用することができます。値のタイプが数値の場合はワイルドカードを使用することはできません。あるパラメータの値が数値の値と文字の値が混在している場合、値のタイプは文字となります。

ワイルドカードの使用例と PictMaster が生成する制約式を以下に示します。

制約表	
パラメータ	制約 1
A	
B	*B*
C	#C??

図 4-14 ワイルドカードの使用例

### リスト 4-10 ワイルドカードを含む制約式

```
if ([B] LIKE "*B*")  
    then ([C] NOT LIKE "C??") ;
```

値の名称に適切な文字列を付与することで、ワイルドカードを使用して制約指定を簡潔に記述することができます。

## 5. 結果表への記入のしかた

### 5. 1 結果表の構成

テストケースの組み合わせによってテスト結果が異なる場合があります。テストケースの数が多い場合は、期待する結果（結果内容）を記入するのに手間がかかりますが、結果表を使用することでこの手間を省くことができます。テストケースの生成を繰り返す場合は大幅な省力化になります。

未記入の結果表を図 5-1 に示します。

結果表		
結果内容	パラメータ1	パラメータ2

図 5-1 未記入の結果表

結果表の**結果内容欄**には、右側の各パラメータの、値の組み合わせの場合にどのような結果となるべきかを記入します。記入内容が長くなる場合は、結果内容欄に（\*1）（\*2）…などの番号を記入し、生成されたテストケースが書かれたワークシートの欄外で番号ごとに具体的な結果内容を記述すればよいでしょう。結果内容欄は 100 行設けられており、デフォルトでは 15 行分表示されています。不足する場合は残りの行を再表示してください。環境設定フォームを表示させて OK ボタンをクリックすると、結果表のパラメータ欄に実際のパラメータ名が設定されます。

結果内容欄は 1 行目から間を空けずに詰めて記入してください。

結果表の**パラメータ欄**の列には、左側の結果内容欄の結果となる値の組み合わせ（一致条件）を制約表の記述と同じ方法で記入します。ただしパラメータそのものを記入することはできません。

パラメータ欄の列は 50 列設けられています。

### 5. 2 一致条件の指定方法

一致条件は制約表で値を指定する場合と同じ方法で指定します。左側の結果内容欄に書かれている結果となる値の組み合わせを必要なパラメータごとに記入します。1 つの欄には値をカンマ（,）で区切って複数記入することができます。先頭にシャープ（#）をつけると逆条件となり、指定した値以外の値を意味します。色の塗りつぶしは不要です。

一致条件欄の記入は制約表の記述に違反しないようにします。結果内容欄に記入があり、すべてのパラメータの一致条件欄になにも記入がない場合は、他の結果内容欄の一致条件にいずれにも一致しなかったテストケースの結果内容欄にその記述内容が記入されます。

図 5-2 にモデルの例、および図 5-3 に結果表の例を示します。

パラメータ	値の並び
A	a1, a2, a3, a4
B	b1, b2, b3, b4
C	c1, c2, c3

表 5-2 モデルの例

結果表			
結果内容	A	B	C
1111となる	a1, a2	b1, b2	
1111となる		b3	c3
2222となる	#a1, a2		c2
3333となる			

図 5－3 結果表の例

結果表の記入を行なう際は、CTL-e（デフォルト）を押してウインドウの分割を行ない、モデルと結果表がともに見えるようにすると記入しやすくなります。

図 5－3 では結果内容欄には 4 つの確認内容が記入されており、最後の確認内容には一致条件が何も記入されていません。それまでの 3 つの条件に一致しなかったテストケースに最後の結果内容が記入されます。最初の 2 つの結果内容欄は同じ内容です。このような指定も可能です。

一致条件はパラメータ間の AND 条件となります。記入されたすべての条件が一致するテストケースのみに左側の結果内容が記入されます。記入された値がエイリアスの場合はそのすべての値を意味します。

図 5－2 から図 5－3 の条件で生成されたテストケースの例を表 5－1 に示します。

表 5－1 生成されたテストケース

No.	A	B	C	結果内容
1	a1	b1	c3	1111 となる
2	a1	b2	c1	1111 となる
3	a1	b3	c1	3333 となる
4	a1	b4	c2	3333 となる
5	a2	b1	c1	1111 となる
6	a2	b2	c3	1111 となる
7	a2	b3	c2	3333 となる
8	a2	b4	c3	3333 となる
9	a3	b1	c3	3333 となる
10	a3	b2	c3	3333 となる
11	a3	b3	c2	2222 となる
12	a3	b4	c1	3333 となる
13	a4	b1	c2	2222 となる
14	a4	b2	c2	2222 となる
15	a4	b3	c3	1111 となる
16	a4	b4	c1	3333 となる

生成されたテストケースには右端に結果内容の欄が追加され、一致条件に一致したテストケースについて結果表の結果内容欄の記述が記入されています。テストケース数が数十個以上と多い場合は結果表機能があると時間の節約になります。生成を繰り返した場合は時間短縮効果が大変大きくなります。

エイリアスの値を指定する場合は、先頭の値の名称を使用します。

### 5. 3 使用できる演算子の一覧

一致条件で使用できる演算子を表5-2に示します。値はカンマ(,)で区切って複数記述することができます。条件の欄は1つの一致条件欄に複数の値を記述した場合にOR条件となるのかAND条件となるのかを表します。

表5-2 使用できる演算子

関係式の対象	演算子	条件	説明
パラメータと値	(指定しない)	OR	パラメータに属する値を表します。=と同じ意味です。カンマ(,)で区切って複数記述することができます。
	#	AND	記述した値を除いた残りの値を表します。値の先頭に1つだけ記述しますが値は複数記述することができます。

### 5. 4 記入上の注意事項

結果表の記入に関する注意事項を説明します。

#### (1) 一致条件の重複

複数の結果内容を定義した場合、同じ組み合わせに2つの異なる結果内容が一致することがあります。これは一致条件が重複する条件を含んでいるためです。この状態となった場合、PictMasterは図5-4の例のようなエラーメッセージを表示し、そこで処理を中止します。このエラーとなった場合は、条件が重複しないように「#」を使った逆条件を追加するとうまくなります。

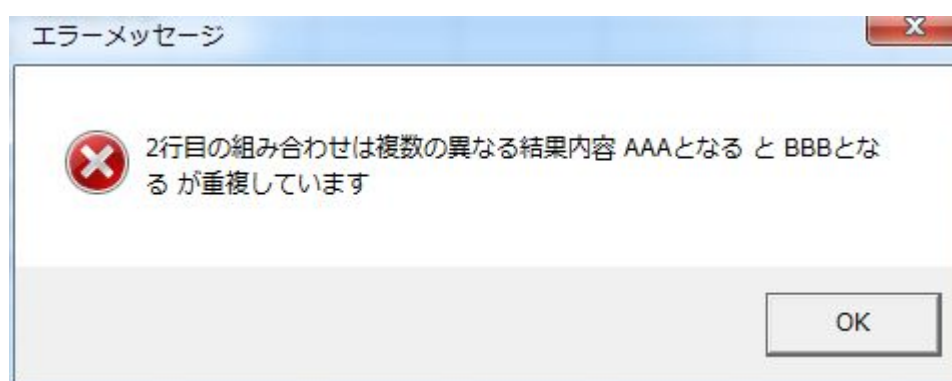


図5-4 一致条件重複のエラーメッセージの例

#### (2) 一致条件の不一致

一致条件に一致する組み合わせが1つもない場合があります。たとえば組み合わせるパラメータ数を2としている場合に一致条件として3つ以上のパラメータの値について指定した場合、起きる可能性が高くなります。また制約表の記述に違反した組み合わせを指定する一致条件の場合もこのエラーとなります。この状態となった場合、PictMasterは図5-5の例のようなエラーメッセージを表示し、そこで処理を中止します。

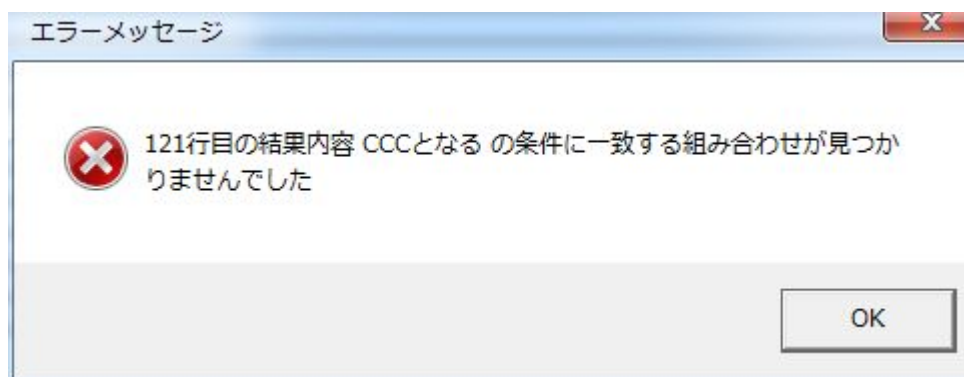


図 5－5 一致条件不一致のエラーメッセージの例

## 5. 5 結果内容の連結

これまで説明した結果表の使い方は、1つのテストケースは1つの結果内容にのみ一致することを条件としていました。しかし、テスト対象によっては、この仕様では結果表への記入が煩雑になることがあります。それは、1つのテストケースを構成する多数のパラメータの値によって結果内容が様々に変化する場合です。

例えば図 5－6 に示すモデルで、結果内容がパラメータ A の値と B の値によって変化する場合、これまでの方法では図 5－6 の結果表のように  $3 \times 3 = 9$  個の結果内容を記入しなければなりません。生成された結果は、表 5－3 のようになります。

パラメータ	値の並び
A	a1, a2, a3
B	b1, b2, b3
C	c1, c2, c3
D	d1, d2, d3

結果表		
結果内容	A	B
A1B1	a1	b1
A1B2	a1	b2
A1B3	a1	b3
A2B1	a2	b1
A2B2	a2	b2
A2B3	a2	b3
A3B1	a3	b1
A3B2	a3	b2
A3B3	a3	b3

図 5－6 モデルと結果表の例

表 5－3 生成結果の例

No.	A	B	C	D	結果内容
1	a1	b1	c3	d3	A1 B1
2	a1	b2	c1	d2	A1 B2
3	a1	b2	c2	d2	A1 B2
4	a1	b3	c2	d1	A1 B3
5	a2	b1	c2	d2	A2 B1
6	a2	b2	c3	d1	A2 B2
7	a2	b3	c1	d3	A2 B3
8	a3	b1	c1	d1	A3 B1
9	a3	b2	c2	d3	A3 B2
10	a3	b3	c3	d2	A3 B3

これまで説明した方法では、この例のようにすべての組み合わせ結果ごとに結果表の結果内容を記述しなければなりません。まだ生成結果に影響を及ぼす値の数が少ないからいいのですが、増えてくると必要となる結果内容の欄が急激に増加してしまいます。

この例のように1つのテストケースを構成する多数のパラメータの値によって結果内容が様々に変化する場合、パラメータの値の組み合わせに対して結果内容を記入するのではなく、個々のパラメータの値ごとに独立して結果内容を記述することができれば、記入しなければならない結果内容欄の数を少なくすることができます。生成されたテストケースの結果内容欄には、一致した各内容欄の記入内容が「連結」されて設定されることになります。

図 5－7 にそうした結果表の例を示します。

結果表		
結果内容	A	B
A1	a1	
A2	a2	
A3	a3	
&B1		b1
&B2		b2
&B3		b3

図 5－7 結果内容を連結する結果表の例

この結果表による生成結果の結果内容欄は、表 5－3 とまったく同一となります。結果内容欄の先頭の文字がアンパサンド (&) で始まっている場合、テストケースの値が結果表の一致条件に一致したときの結果内容が異なると、それぞれの異なる結果内容の文字列が「連結」され、新しい結果内容となります。言い換えると、生成されたテストケースの「結果内容」欄にすでに記入されている内容と、今現在一致条件に一致したときの「結果内容」が異なる場合、それぞれの異なる結果内容の文字列が「連結」され、新しい結果内容となります。

この例では結果内容欄は、これまでの方法の 9 個に対して  $3 + 3 = 6$  個で済んでいます。結果内容に影響を与える値が増えるほど、この差は大きなものとなります。

結果表の一致条件に一致したとき、現在一致した結果内容とすでにテストケースの結果内容欄に記入されている内容が異なる場合、順番が後の結果内容欄の先頭に & がないときは、これまでどおり「一致条件の重複」のエラーメッセージが表示されます。



## 6. より有効な使い方

### 6. 1 PictMaster のカスタマイズ

PictMaster は、Excel の Book であることから使いやすいようにカスタマイズすることが可能です。1～7 行目は自由にレイアウトしてかまいません。PictMaster のファイル名、シート名は任意の名前に変更してかまいません。

PictMaster で編集メニューから「シートの移動またはコピー」を選択し、表示されたフォームの「コピーを作成する」を指定して「OK」ボタンをクリックすることで異なるテストケースを生成するシートを任意の枚数設けることができます。これに対して、挿入メニューからワークシートを選択して新しいシートを作成し、PictMaster のシートをコピーして貼り付けてもそのシートでは正常に動作しませんので注意してください。

テスト対象の大きな機能ごとに PictMaster の Book を設け、そのいくつかの組み合わせテストケースのモデルを複数のシートに分けて管理するという方法がよいかもしれません。またテスト仕様書を Excel で作成している場合は、PictMaster を使用してテストケースを作成したテスト仕様書を PictMaster の別シート上に記述し、**テスト仕様書と PictMaster を 1 つの Book に統合することも可能です**。その場合、「PictMaster」という Book 名はテスト対象を表す機能名などの名称に変更することになるでしょう。このような例でのシート名の並びを図 6-1 に示します。

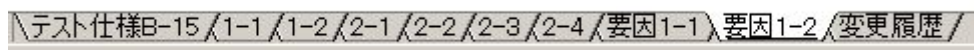


図 6-1 シート名の並びの例

この例ではシート名「テスト仕様 B-15」がテストの記号名であり Book 名です。このシートは 1-1～2-4 のシートに記入されているテストケースについてのテスト方法、確認内容、データ設定内容などを記述した**テスト設計仕様書(\*1)**のシートです。1-1～2-4 のシートは個々の確認内容に応じた**テストケース仕様書(\*1)**です。これらのテストケースのうち、“要因”がシート名についているシートが PictMaster を使用して組み合わせテストケースの作成に使用した PictMaster のシートです。

\*1: IEEE 829 標準規格に準拠したドキュメント。テスト手順書に該当する内容は、テスト設計仕様書またはテストケース仕様書で包含しているので設けていない。

### 6. 2 エラー／警告メッセージが表示された場合

PictMaster では制約表から制約式に変換する過程で多くのチェックを行っており、PICT 自体からエラー／警告メッセージが表示されることはまれだと思います。

エラー／警告メッセージが表示されている間はそのメッセージ内容が記述されているファイル “e.txt” が存在するので、メモ帳などで e.txt のファイルを開くことができます。ファイルを開いた後で、エラー（警告）メッセージの OK ボタンをクリックすれば、メモ帳などでエラー／警告メッセージを見ながら PictMaster のパラメータ定義や制約表の間違いを調べることができます。また警告メッセージの場合は、警告メッセージの OK ボタンをクリックしても、テストケースは生成されているので “a.xls” のファイルを開いて生成結果を確認することができます。

### 6. 2. 1 矛盾した制約をすばやく見つけるには

PICT から表示されるメッセージで多いものとして図 6－2 に示すようなメッセージがあります。



図 6－2 PICT が表示するメッセージの例

生成エンジンが CIT-BACH の場合は PictMaster がチェックを行ない、次に示すようなメッセージを表示します。

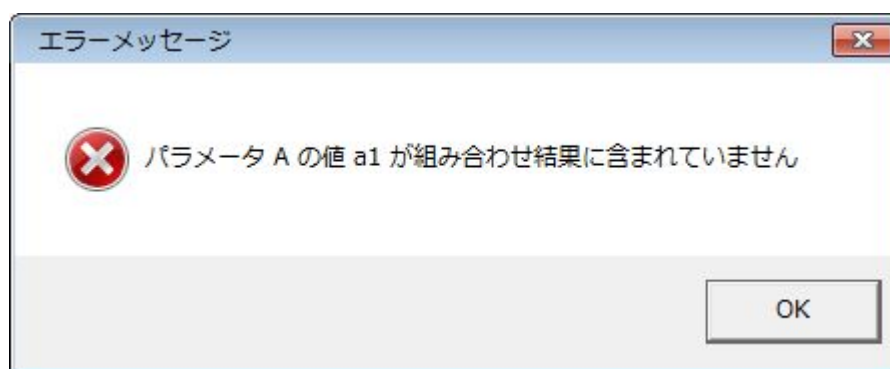


図 6－3 PictMaster が表示するメッセージの例

この例はパラメータ“A”の値“a1”が組み合わせに1つも含まれていないことを警告するメッセージです。この例では1つだけですが場合によっては5～6個（CIT-BACH のときは1個）の指摘がなされることもあります。

こうしたメッセージが出る原因は制約の指定に誤って相互に矛盾する複数の制約を指定したためです。間違った制約を突き止めるには指摘されたパラメータの値が組み合わせに表れないような矛盾した制約の指定を行っていないか制約表で各制約の関係を見直すことです。いくつも指摘された場合はどれか1つに的を絞って調べます。

このような警告が現れる最も単純な制約指定を以下に示します。

制約表		
パラメータ	制約1	制約2
A	a1	a2
B	b1	b1

図 6－4 矛盾した制約指定の例

この制約指定は制約条件の値とそれに対応する制約対象の値が不一致であるため値 a1 が組み合わせに現れません。この制約指定の誤りは一目見れば分かりますが、実際には多くの制約が関係した結果として、こうした制約条件と制約対象との矛盾が起こります。

基本として矛盾した制約かどうかは、同じ行に位置する 2 つのセルの値が異なる場合、値が同じでなくとも矛盾しない関係にあるかどうかを調べることにあります。

### 6. 3 画面を分割し制約表を記入しやすくする

制約の数が多くなると右に横スクロールしなければならないため、各パラメータの値の名称が見えなくなります。そのため左に横スクロールして名称を確認しなければなりません。また結果表の記入を行なう際もパラメータの値が見えず面倒です。これは煩わしいので簡単に記入できる方法を紹介します。

環境設定ボタンを押すと表示される「ウインドウ分割ショートカットキー」に任意の 1 文字（デフォルトは「e」）を入力しておく、コントロールキーを押しながらショートカットキーを押すことで PictMaster のウインドウが 2 つ開かれ、上下に整列され、下側のウインドウはパラメータ欄と制約欄との間で分割されます。PictMaster が複数のワークシートで構成されている場合は、各ワークシートの環境設定フォームで指定されたショートカットキーが有効となります。

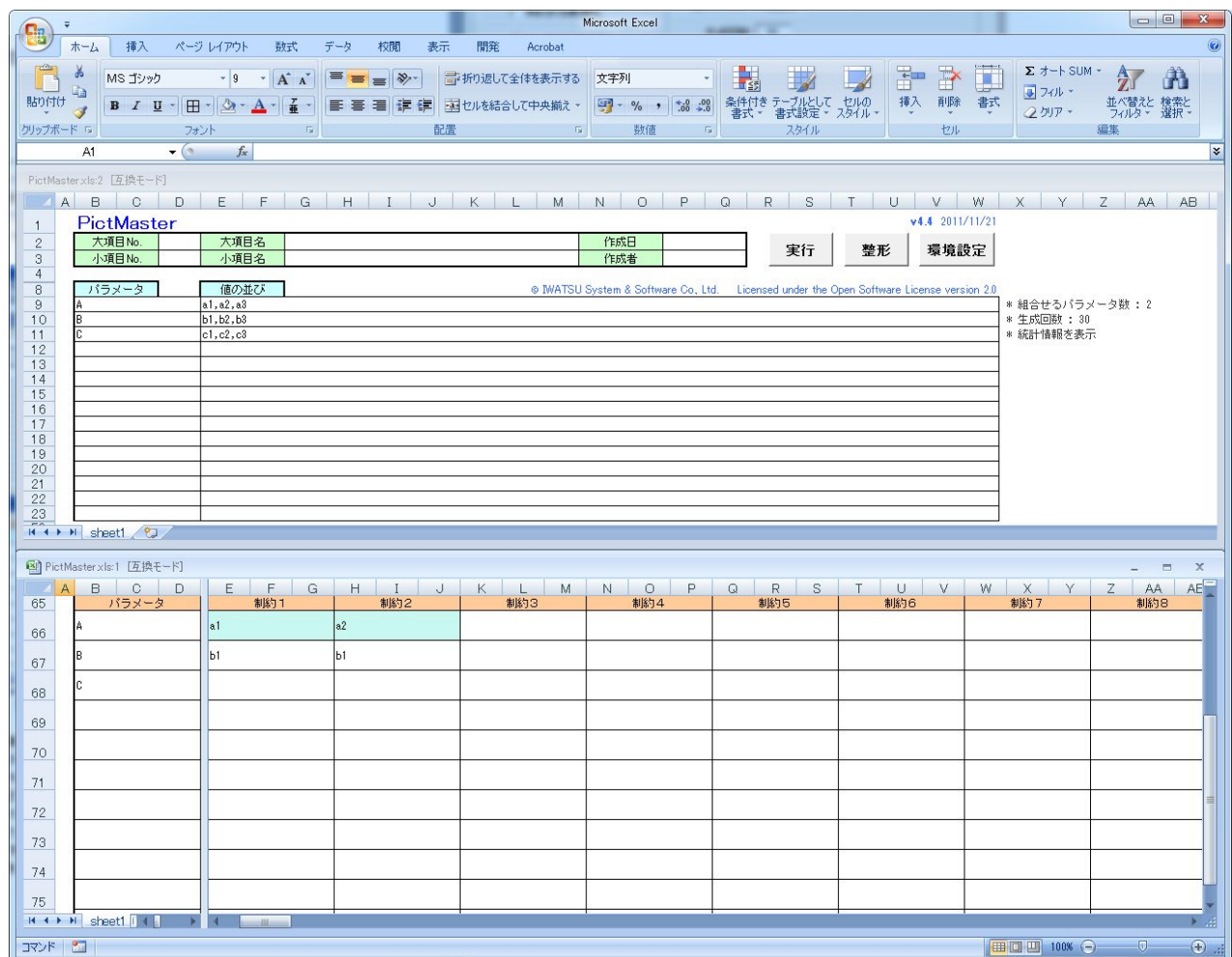


図 6-5 ウインドウ分割の画面例

この状態の画面では制約表を横スクロールしてもパラメータと値が常に見えているので多くの制約を必要とする場合に制約表への記入がしやすくなります。

この状態で再度コントロールキーを押しながらショートカットキーを押すと分割前の画面に戻ります。

なお他のシートや Book に一旦切り替えた場合、元の画面に戻るとウインドウ分割の状態が変化しています。ここで再度ウインドウ分割のショートカットキーを2～3回押すことで正しい分割画面とすることができます。他のシートを表示したい場合は画面の下の方のウインドウからシートを選択することで、選択したシートの本来のズーム（倍率）で表示させることができます。画面の上の方のウインドウから他のシートを選択すると常に100%のズームで表示されます。

#### 6. 4 ショートカットメニューを使って編集する

パラメータ欄と値の並び欄、制約表および結果表は専用のショートカットメニューを使って編集することができます。

第1列目を右クリックすると図6-6に示す編集専用のショートカットメニューが表示されます。ショートカットメニューで上に移動、下に移動、挿入、削除を行なうことができます。

	パラメータ	値の並び
8		
9	A	a1, a2, a3
10	B	b1, b2, b3, b4
11	C	c1, c2, c3
12		d1, d2, d3
13		e1, e2, e3
14		
15		
16		
17		

図6-6 行編集用のショートカットメニュー

制約表の制約タイトル欄を右クリックすると図6-7に示す編集専用のショートカットメニューが表示されます。ショートカットメニューで左に移動、右に移動、挿入、削除を行なうことができます。

制約表	パラメータ	制約1	制約2	制約3
A				
B		b1, b2		
C		c1	c2	
D			d1, d2	

図6-7 列編集用のショートカットメニュー

削除を行なった場合は次のショートカットメニューで「元に戻す」が表示され、削除前の状態に戻すことができます。

一度に編集できる行または列は1つです。挿入と削除は最後の行や列では無効です。

制約表の記入欄を右クリックすると図 6－8 に示す編集専用のショートカットメニューが表示されます。ショートカットメニューで値やパラメータの記入、削除をワンタッチで行なうことができます。

制約表			
パラメータ	制約1	制約2	制約3
OS	Windows XP, Windows 7	Windows 8	
Memory	1GB, 2GB		
HD Interface		USB2.0, USB3.0, eSATA	
HD		500GB, 1TB	

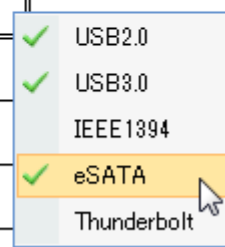


図 6－8 制約表記入用のショートカットメニュー

記入されている値、パラメータを選択すると、制約表の記入欄から削除されます。

パラメータを記入するにはパラメータ用の演算子 (=) または (!) を記入してからショートカットメニューを表示します。

このショートカットメニューは結果表とサブモデル欄でも使用することができます。サブモデル欄にセミコロンが含まれている場合、ショートカットメニューは使用できません。

コントロールキーを押しながら右クリックすると Excel のショートカットメニューが表示されます。

## 付録A 仕様

No.	項 目	値
1	パラメータの最大個数	50
2	パラメータあたり値の最大個数	50
3	制約表の最大制約数	100
4	結果表の最大条件数	100
5	結果表の1つの行に記入可能な値展開後の値の最大個数 (*1)	500
6	結果表で処理可能な生成結果の最大行数	65534
7	組み合わせるパラメータ数の範囲	1～50
8	最小テストケース生成回数の範囲	2～999
9	最小テストケース生成シードの範囲	0～65534
10	整形可能な最大行数	65534
11	原型シートの最大行数	65534
12	値の重み付け指定の範囲	2 倍～10 倍
13	重複した組み合わせを削除可能な最大行数	65534
14	カバレッジを指定可能な組み合わせ数の範囲	2～5
15	指定可能な t-way カバレッジの範囲	30～95 [%]
16	指定カバレッジ生成の繰り返し回数の範囲	3～10
17	カバレッジを表示可能な組み合わせ数の範囲	2～5

\*1: 1つの行のすべてのパラメータの列に記入された値の個数の合計。値がエイリアスを含む場合は、エイリアスで指定した値の数を加算してカウントする。

## 付録B 制限事項

### 【重要な注意点】

多くのパラメータ、値、制約または組み合わせ数を指定した場合など、PICT の処理能力の限界により、テストケースを生成するまできわめて長時間かかる場合があります、こうした場合に処理を途中で止めた場合は、タスクマネージャを起動し、プロセスから pict.exe を選択し、プロセスの終了を行なってください。

CIT-BACH の場合に処理を途中で止めたい場合は、タスクマネージャを起動し、プロセスから java.exe を選択し、プロセスの終了を行なってください。

## 付録 C 生成エンジンによる機能の相違

○ は使用可 ×は使用不可 △ は制限付き使用可

No.	機 能	PICT	CIT-BACH	備 考
1	組み合わせるパラメータ数	○	△	CIT-BACH では2～5、6以上は全数組合せのみ
2	値の重み付け	○	○	
3	エイリアス	○	○	
4	無効値の指定	○	×	
5	制約表	○	○	
6	制約式の最適化	○	—	CIT-BACH では最適化は不要
7	結果表	○	○	
8	サブモデル	○	△	CIT-BACH は全数組み合わせのみ
9	原型シート	○	○	
10	最小テストケース生成	○	○	CIT-BACH ではプログレスバーの表示なし
11	カバレッジの表示	○	○	
12	カバレッジを指定して生成	○	○	
13	矛盾した制約の検出	○	○	CIT-BACH では1つのみ検出
14	値の大小比較	○	×	
15	ワイルドカードの使用	○	×	

### 【PICT と CIT-BACH の使い分けについて】

PICT と CIT-BACH はそれぞれ異なる性質を持っています。テスト対象に合わせて使い分けることができます。

PICT の特徴はとにかく多機能だという点です。まさにかゆい所に手がとどく組み合わせエンジンということができます。その反面、極めて複雑な制約が指定された場合に生成にかかる時間が非常に長くなる場合があります。

CIT-BACH は極めて複雑な制約が指定された場合でも短時間で生成することができます。また最小テストケース生成はエンジン内で行なわれるので PICT に比べて短時間で完了します。一方で機能面ではそれほど多機能ではありません。

通常は CIT-BACH を使用し、PICT 固有の機能が必要な場合だけ PICT を使用する、という使い方がいいかもしれません。



## 付録 D PictMaster でデシジョンテーブルのテストケースを自動生成する

### 1. はじめに

テスト対象が論理関係を持っている場合にデシジョンテーブルを用いたテストを行なうことができます。テスト対象が複雑な論理関係を持っている場合、手作業でデシジョンテーブルを作成しようとすると時間がかかり誤りも起きやすいという問題があります。

動作に対応する条件の組み合わせのルールは制約表として表現することができます。この章では制約表がデシジョンテーブルとほぼ同じ構成を有していることを説明し、制約表を備えた PictMaster を用いることで複雑な論理関係を持つデシジョンテーブルのテストケースを容易に自動生成できることを説明します。

### 2. デシジョンテーブルと制約表の対応

デシジョンテーブルは条件の全数組み合わせであり、その組み合わせに論理関係が存在します。

この論理関係は、条件と条件間の論理関係と、条件と動作間の論理関係の 2 つに分類することができます。

条件と条件間の論理関係とは例えば、中学生と中学生都内在住という条件がある場合、中学生=Yes ならば中学生都内在住は Yes と No がありますが、中学生=No ならば中学生都内在住は No のみとなるような論理関係です。条件同士の論理関係であり、動作との直接的な関係はありません。

条件と動作間の論理関係とは例えば、年齢という条件が中学生以上と小学生以下であるとき、料金という動作が中学生以上と小学生で異なるような論理関係です。条件と条件間の論理関係は動作とは独立して定義することができますが、このことは条件と動作が無関係であることを意味しません。

デシジョンテーブルの各動作に対応して、各条件は一意に決定することができます。このことは、各動作を条件との論理関係を表す制約条件とし、その制約条件と組み合わせることが可能な条件の組み合わせを制約対象として定義した制約表を用いることでデシジョンテーブルを表現することが可能であることを意味しています。

デシジョンテーブルと制約表の対応を図 2 に示します。

条件記述部		独立条件指定部		条件指定部	
制約表					
パラメータ	制約1	制約2	制約3	制約4	
条件A		Yes	No	No	
条件B	No		No	Yes	
条件C	No			Yes	
動作X		Yes	Yes	No	
動作Y		Yes	Yes	No	
動作Z		Yes	No	No	

動作記述部

動作指定部

図 2. デシジョンテーブルと制約表の対応



制約表にはデシジョンテーブルにはない条件と条件間の論理関係を定義する独立条件指定部の存在する場合があります。これは必須ではなくオプションです。独立条件指定部で可能な組み合わせを定義することにより、条件指定部での個々の定義が不要となり、制約表の記述を簡潔にできるという利点があります。

制約表の記入では1つの動作に着目し、それを制約条件とします。異なるルールで同じ動作（制約条件が同じ）となる場合は、同じ制約条件（動作）を隣り合う制約として定義することで、1つの動作で2つ以上のルールをOR条件として定義することができます。図2では制約2と制約3がそれに該当します。

以上のようにPictMasterでデシジョンテーブルの制約表を定義し、組み合わせるパラメータに条件と動作を合わせた数を指定することで、デシジョンテーブルのテストケースを自動生成することができます。

### 3. 適用事例その1

#### 3. 1 遊園地の入場料問題

適用事例のひとつとして、ある遊園地の入場料をテストするデシジョンテーブルを取り上げます。この入場料は次の仕様で決まります。

表3. 1-1 入場料の仕様

	個人	団体
一般	1200円	1000円
小学生	600円	500円
65歳以上	無料	無料
6歳未満	無料	無料

ただし県内在住の小学生は無料

この入場料を決定するデシジョンテーブルを生成するPictMasterのモデルを図3. 1に示します。パラメータには条件と動作の名称を、値には条件と動作が取りうる値を記入します。

パラメータ	値の並び	Licen
区分1	個人, 団体	
区分2	一般, 小学生, 65歳以上, 6歳未満	
県内在住	Yes, No	
入場料	1200円, 1000円, 600円, 500円, 無料	

制約表	制約1	制約2	制約3	制約4
パラメータ				
区分1	個人	団体	個人	団体
区分2	一般	一般	小学生	小学生
県内在住			No	No
入場料	1200円	1000円	600円	500円

パラメータ	制約5	制約6	制約7	制約8
区分1				
区分2	小学生	65歳以上	6歳未満	
県内在住	Yes			
入場料	無料	無料	無料	

図3. 1 入場料を決定するデシジョンテーブルのモデル

制約表で制約 1 から制約 4 は入場料が有料となる条件を指定しています。制約 5 は小学生で県内在住の場合に無料となることを指定しています。

このモデルで生成されたテストケースを次に示します。

表 3. 1-2 自動生成された入場料のテストケース

No.	区分1	区分2	県内在住	入場料
1	個人	65歳以上	No	無料
2	個人	65歳以上	Yes	無料
3	個人	6歳未満	No	無料
4	個人	6歳未満	Yes	無料
5	個人	一般	No	1200円
6	個人	一般	Yes	1200円
7	個人	小学生	No	600円
8	個人	小学生	Yes	無料
9	団体	65歳以上	No	無料
10	団体	65歳以上	Yes	無料
11	団体	6歳未満	No	無料
12	団体	6歳未満	Yes	無料
13	団体	一般	No	1000円
14	団体	一般	Yes	1000円
15	団体	小学生	No	500円
16	団体	小学生	Yes	無料

このテストケースは圧縮していません。動作に影響を与えない条件の集まりをまとめて1つに圧縮するというやり方がありますが、それが可能なのはテスト対象のプログラム構造が分かっている場合です。プログラム構造が分かっていない場合は基本的にテーブルの圧縮はやるべきではありません。

このテストケースでは圧縮を行なうと7件のテストケースとなります。そうした場合に漏れのないテストを行なうためには、テスト対象のプログラムが7か所だけで入場料を決定している必要があります。もし、テスト対象のプログラムが7か所より多い箇所で入場料を決定していた場合、テスト漏れが起きることになります。

デシジョンテーブルを安全に圧縮できるのは、テスト対象のプログラム構造が分かっている場合にに限られるということは覚えておいたほうがよいでしょう。

PictMaster を使用してデシジョンテーブルの圧縮を行なう場合は、任意値を表す「-」などの値を追加し、制約表でそれとの組み合わせを指定することで圧縮されたテストケースを生成することができます。

### 3. 2 ATM 利用手数料問題

少し複雑な適用事例として、ある銀行の ATM 利用手数料をテストするデシジョンテーブルを取り上げます。この手数料は次の仕様で決まります。

- ① 8 時～18 時は無料
- ② その他の時間外は 108 円
- ③ 土日祝日は 108 円
- ④ 預金残高 10 万円以上は平日の時間外は無料
- ⑤ 毎月 25 日と 26 日は無料
- ⑥ 25 日が土日祝日のときは、その前の窓口営業日が終日無料
- ⑦ 26 日が土日祝日のときは、その次の窓口営業日が終日無料
- ⑧ 25 日と 26 日が土日祝日と重なる場合は終日 108 円

この ATM 利用手数料を決定するデシジョンテーブルのテストケースを生成する PictMaster のパラメータと値を図 3. 2-1 に、制約表を図 3. 2-2 に示します。

パラメータ	値の並び
時刻	8時～18時, 時間外
預金残高	10万円以上, 10万円未満
今日の日付	25日, 26日, 他の日
今日が土日祝日	Yes, No
25日が土日祝日で今日がその前の窓口営業日	Yes, No
26日が土日祝日で今日がその次の窓口営業日	Yes, No
ATM手数料	108円, 無料

図 3. 2-1 ATM 利用手数料のパラメータと値

制約表					
パラメータ	制約 1	制約 2	制約 3	制約 4	制約 5
時刻			8時～18時	時間外	時間外
預金残高				10万円以上	
今日の日付	他の日	他の日			25日, 26日
今日が土日祝日	No	No	No	No	No
25日が土日祝日で今日がその前の窓口営業日	Yes	No		No	No
26日が土日祝日で今日がその次の窓口営業日	No	Yes		No	No
ATM手数料			無料	無料	無料

パラメータ	制約 6	制約 7	制約 8	制約 9	制約 10
時刻	時間外	時間外	時間外		
預金残高			10万円未満		
今日の日付	他の日	他の日	他の日		
今日が土日祝日	No	No	No	Yes	
25日が土日祝日で今日がその前の窓口営業日	Yes	No	No	No	
26日が土日祝日で今日がその次の窓口営業日	No	Yes	No	No	
ATM手数料	無料	無料	108円	108円	

図 3. 2-2 ATM 利用手数料の制約表

図 3. 2-2 の制約指定の意味は次の通りです。括弧内は意味している仕様の番号です。

制約 1 と制約 2 (⑥と⑦)：独立条件指定部である。2つの条件がともに Yes となる組み合わせはあり得ないのでこれを除外している。またこの条件が Yes になる日は 25 日でも 26 日でもないなので他の日と組み合わせ、土日祝日とは組み合わせないようにしている。

制約 3 (①)：平日の時間内であればどのような条件でも無料である。

制約 4 (④)：平日の時間外で預金残高が 10 万円以上の場合は無料である。

制約 5 (⑤)：平日の時間外で 25 日と 26 日は無料である。

制約 6 (⑥)：時間外で 25 日が土日祝日で今日がその前の窓口営業日の場合は無料である。

制約 7 (⑦)：時間外で 26 日が土日祝日で今日がその次の窓口営業日の場合は無料である。

制約 8 (②)：平日の時間外で預金残高が 10 万円未満の場合は 108 円である。

制約 9 (③と⑧)：今日が土日祝日の場合は 108 円である。

この制約表では、デシジョンテーブルの圧縮を行なっていません。

PictMaster の組み合わせるパラメータ数に条件と動作を合わせた数である 7 を指定し、生成を行なうと表 3. 2 の 32 件のテストケースが得られます。

表 3. 2 自動生成された ATM 利用手数料のテストケース

No.	時刻	預金残高	今日 の日付	今日が 土日祝日	(*1)	(*2)	ATM 手数料
1	8時～18時	10万円以上	25日	Yes	No	No	108円
2	8時～18時	10万円以上	25日	No	No	No	無料
3	8時～18時	10万円以上	26日	No	No	No	無料
4	8時～18時	10万円以上	26日	Yes	No	No	108円
5	8時～18時	10万円以上	他の日	No	No	No	無料
6	8時～18時	10万円以上	他の日	No	No	Yes	無料
7	8時～18時	10万円以上	他の日	Yes	No	No	108円
8	8時～18時	10万円以上	他の日	No	Yes	No	無料
9	8時～18時	10万円未満	25日	Yes	No	No	108円
10	8時～18時	10万円未満	25日	No	No	No	無料
11	8時～18時	10万円未満	26日	Yes	No	No	108円
12	8時～18時	10万円未満	26日	No	No	No	無料
13	8時～18時	10万円未満	他の日	Yes	No	No	108円
14	8時～18時	10万円未満	他の日	No	No	Yes	無料
15	8時～18時	10万円未満	他の日	No	No	No	無料
16	8時～18時	10万円未満	他の日	No	Yes	No	無料
17	時間外	10万円以上	25日	Yes	No	No	108円
18	時間外	10万円以上	25日	No	No	No	無料
19	時間外	10万円以上	26日	Yes	No	No	108円
20	時間外	10万円以上	26日	No	No	No	無料
21	時間外	10万円以上	他の日	No	No	No	無料
22	時間外	10万円以上	他の日	No	No	Yes	無料
23	時間外	10万円以上	他の日	No	Yes	No	無料
24	時間外	10万円以上	他の日	Yes	No	No	108円
25	時間外	10万円未満	25日	Yes	No	No	108円
26	時間外	10万円未満	25日	No	No	No	無料
27	時間外	10万円未満	26日	Yes	No	No	108円
28	時間外	10万円未満	26日	No	No	No	無料
29	時間外	10万円未満	他の日	No	No	No	108円
30	時間外	10万円未満	他の日	No	No	Yes	無料
31	時間外	10万円未満	他の日	No	Yes	No	無料
32	時間外	10万円未満	他の日	Yes	No	No	108円

\*1: 25 日が土日祝日で今日がその前の窓口営業日

\*2: 26 日が土日祝日で今日がその次の窓口営業日

#### 4. 手作業による作成との比較

手作業でデシジョンテーブルを作成する一般的な方法は次の通りです。

- (1) 各条件のすべての組み合わせを作成する。
- (2) 各条件の組み合わせに対応する動作を記入する。
- (3) 作成した組み合わせのうち、組み合わせ可能な組み合わせを残して他を削除する。

手作業でデシジョンテーブルを作成しようとする場合、テーブルのサイズが大きいと作成が難しくなります。表計算ソフトを使用することでサイズの大きなテーブルを作成することはできますが、ルールが複雑であると作成が難しくなります。

別の方法として動作を最初に記入し、その動作となる組み合わせを後から記入するという方法があります。この方法では複雑なルールの適用は比較的容易に行なうことができますが、テーブルが大きいと必要とされる多くの組み合わせを手作業で網羅しなければならず、誤りの混入する危険が増すことになります。

これに対して制約表による方法では、大きなテーブルの作成はツールが自動的に行ない、複雑なルールの適用は制約表を使用して視覚的に分かりやすく行なうことができます。このことにより、制約表による方法ではサイズが大きく、ルールが複雑なデシジョンテーブルでも容易に作成が可能となっています。