

# チェックリストと分割に基づく 網羅と使用テスト

Coverage and Usage Testing Based on Checklists and Partitions

第 8 章 (p107 ~ p126) B4M1 輪講

修士課程 1 年生 楊 嘉晨

2012 年 5 月 29 日 (火)

## 1 8.1 チェックリストに基づくテスト, とその制限

### 1.1 概要

概要 (p103)

チェックリストや分割 (Partition) 等簡単なモデルで正規テストの手法について紹介

1. 8.1 節, 様々なチェックリストで正規と半正規のテスト
2. 8.2 節, チェックリストを分割に正規化して, 簡単なカバレッジテストを行い
3. 8.3 節, 操作プロフィール (Operation Profile, OP) という, 分割のために簡単な UBT(Usage-based Testing) を紹介
4. 8.4 節, OP を生成する手順
5. 8.5 節, Case Study

第 9 章には, 分割した入力サブドメインの境界条件のテストについて, 似ているモデルを紹介

### 1.2 チェックリストに基づくテスト

Ad hoc テストとランダムテスト (p104)

t

繰り返して Ad hoc テストを実行する時, テスト者は行ったテストを追跡すると, チェックリストテストになる

チェックリストに基づくテスト (p104)

t

- ・ ブラックボックステスト (BBT)
  - ソフトウェア要求チェックリスト
  - 機能チェックリスト
    - \* システム全体からハイレベルの機能
    - \* ローレベルで独立な部品
- ・ ホワイトボックステスト (WBT)
  - プログラムの特徴
  - コーディング標準 (Coding Standard)
- 単体テストのコードの網羅
- 統合テストとシステムテストの部品の網羅
- ・ 構造と特徴など, 実装に関するチェックリスト
  - 関数の呼び出し規則
  - 資源の生産者と消費者
  - モジュール間に共有するデータ

### 1.3 階層及び複合チェックリスト

#### 階層チェックリスト (p105)

1. ハイレベル項目 1
2. ハイレベル項目 2
3. ハイレベル項目 3



- (a) ローレベル項目 1
- (b) ローレベル項目 2
- (c) ローレベル項目 3
2. ハイレベル項目 2
  - (a) ローレベル項目 4
  - (b) ローレベル項目 5
3. ハイレベル項目 3
  - (a) ローレベル項目 6
  - (b) ローレベル項目 7
  - (c) ローレベル項目 8

1. ハイレベル項目 1

#### 複合チェックリスト (p105)

| Component | Standards Items |       |         |       |
|-----------|-----------------|-------|---------|-------|
|           | $s_1$           | $s_2$ | $\dots$ | $s_n$ |
| $c_1$     |                 |       |         |       |
| $c_2$     |                 |       |         |       |
| $\vdots$  |                 |       |         |       |
| $c_m$     |                 |       |         |       |

図 4: 標準チェックリストと部品チェックリストを複合した二次元表

## 1. 部品 1

- (a) 要求 1
- (b) 要求 2
- (c) 要求 3

## 2. 部品 2

- (a) 要求 1
- (b) 要求 2
- (c) 要求 3

## 3. 部品 3

- (a) 要求 1
- (b) 要求 2
- (c) 要求 3

## 1.4 チェックリストの問題点と制限

## チェックリストの問題点 (p106)

- ・ チェックリストの定義が抽象的過ぎて, 具体的にするのが難しい
  - 特にハイレベル
- ・ 具体的なテストケースに変換するのは
  - 経験が必要
  - 特殊な環境や設定等に依存
- ・ 相互接続と相互作用を定義するのも難しい
  - 特に大規模で, 複雑なシステムに

### チェックリストの制限 (p106)

1. 全部の機能（ブラックボックステスト）又は構造部品（ホワイトボックステスト）を, 異なる視点や保証レベルから, 網羅することが難しい
  - ・ 網羅されていない穴（Hole）が残ってしまう
  - ・ 分割したチェックリストを使う
2. より高い網羅率を目標にすると, テストを重複してしまう
  - ・ 無駄なテストを行われてしまう
  - ・ 分割したチェックリストを使う
3. 各システムの部品間の複雑な相互作用を定義するのは難しい
  - ・ 10, 11 章に FSM に基づく体系的な正規モデルを紹介

## 2 8.2 分割カバレッジテスト

### 2.1 紹介

#### 分割カバレッジテスト (p107)

- ・ 分割を基づくテストは一種のチェックリスト・テスト
- ・ 分割は集合全体を徹底的に覆う
  - － より高い網羅率
- ・ 分割はお互いに重複することができない
  - － より高い効率

### 2.2 8.2.1 Some Motivational Examples

#### Motivational Examples(p107)

全部可能な入力の組み合わせ

$$ax^2 + bx + c = 0$$

その解を求めると

$$r = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

$$2^{32} \times 2^{32} \times 2^{32} = 2^{96}$$

### 2.3 8.2.2 分割：概念と定義

#### 分割：概念と定義 (p108)

1. 分割した集合は相互に排他的
2. 分割した集合の和集合は全体の集合

$$\forall i, j, i \neq j \Rightarrow G_i \cap G_j = \emptyset$$

$$\bigcup_{i=1}^n G_i = S$$

分割した集合は同値類になる

$$R(a, b) \wedge R(b, c) \Rightarrow R(a, c)$$

対称律 (symmetric)

$$R(a, b) \Rightarrow R(b, a)$$

反射律 (reflexive)

推移律 (transitive)

$R(a, a)$  が常に成り立つ

## 2.4 8.2.3 分割テストの決定と網羅範囲の予測

### 分割テストの網羅範囲の決定 (p109)

分割テストは一種のチェックリストであり, 分割の種類は 8.1 節に述べたチェックリストの種類に似ている. 但し, 分割の決定は以下のとおりに決められている

#### 1. 製品によって分割

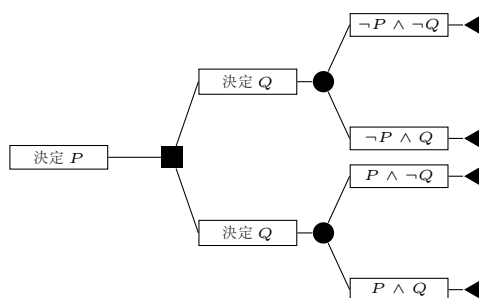
- ・ 例えば外部関数 (BBT 視点) 又はシステム部品 (WBT 視点)
- ・ チェックリストと同じ

#### 2. 性質, 関係, 論理的な条件によって分割, 更に 2 つに分かれている

- ・ 論理変数を論理演算子に繋がれる論理述語による
- ・ 数的な変数を比較演算子によって比較する
- ・ 論理述語や比較演算によって入力区間を分割
- ・ 決定木を使う

#### 3. 1 と 2 の組み合わせ

### 決定木によって分割 (p110)



一貫性を持つ決定木

決定木によって, 一つの結果は一つの分割方法となって, パスを沿って結果を成り立たせる入力の範囲を求める

```

1  if(P){
2      if(Q){
3          P ∧ Q;
4      }else{
5          P ∧ ¬Q;
6      }
7  }else{
8      if(Q){
9          ¬P ∧ Q;
10     }else{
11         ¬P ∧ ¬Q;
12     }
13 }

```

## 3 8.3 Musa 氏の操作プロフィールで使用ベース統計的テスト

### 3.1 8.3.1 Usage-based 統計的テストの場合

統計的 UBT 適用する場合 (p111)

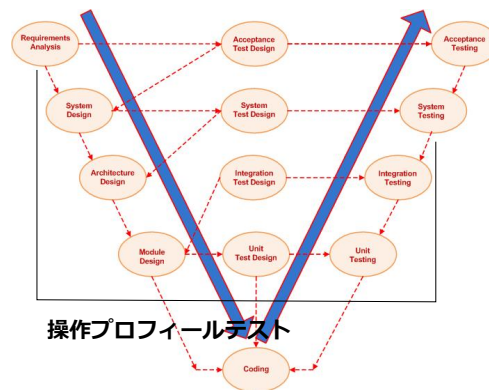


図 6: 操作プロフィールを開発プロセスに導入する時期

| 操作       | 頻度 (%) |
|----------|--------|
| スブドメイン 1 |        |
| -操作 1    | 1%     |
| -操作 2    | 72% ✓  |
| -操作 3    | 2%     |
| スブドメイン 2 |        |
| -操作 4    | 13% ✓  |
| -操作 5    | 2%     |
| スブドメイン 3 |        |
| -操作 6    | 3%     |
| -操作 7    | 7% ✓   |

1. 使用シナリオ, パターン, 関連する使用頻

度を, ターゲットとなる消費者やユーザーから収集

2. 収集した情報を分析し, 操作プロフィール (OP, Operational Profile) に変換

3. 操作プロフィールによってテストを行い

4. テストの結果を分析し, 製品の信頼性を評価し, テストのフィードバックやソフトウェア開発プロセスに役立つ

・製品の信頼性を評価する方法は 22 章に紹介する, 他のテストに関係する活動は 7 章に紹介した

OP を開発プロセスに導入時期 (p111)

### 3.2 8.3.2 Musa 操作プロフィール: 基本的な考え方

Musa による OP の定義 (p112)

**定義 1** (操作プロフィール). 操作プロフィールとは, 一連の操作とそれらの出現に関連する確率の配列

An operational profile is a list of disjoint set of operations and their associated probabilities of occurrence

—[Musa(1993)]

## 参考文献

[Musa(1993)] J. Musa, “Operational profiles in software-reliability engineering,” Software, IEEE, vol. 10, no. 2, pp. 14–32, 1993.

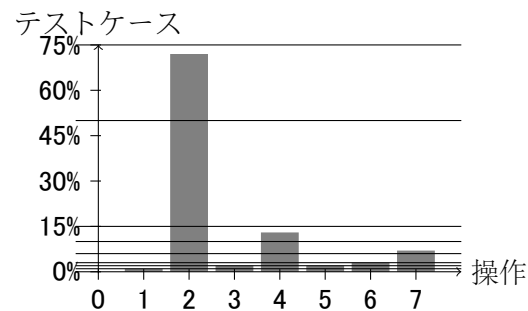
Musa OP: 基本的な考え方 (p113)

### 3.3 8.3.3 操作プロフィールを用いて統計的なテスト等を使う

操作プロフィールで統計的なテスト (p114)

| 操作       | 頻度 (%) | テスト数 (合計 30) |
|----------|--------|--------------|
| サブドメイン 1 |        |              |
| -操作 1    | 1%     | 0            |
| -操作 2    | 72%    | 23           |
| -操作 3    | 2%     | 0            |
| サブドメイン 2 |        |              |
| -操作 4    | 13%    | 4            |
| -操作 5    | 2%     | 0            |
| サブドメイン 3 |        |              |
| -操作 6    | 3%     | 1            |
| -操作 7    | 7%     | 2            |

操作プロフィールで進捗テスト (p114)



操作プロフィール：その他の目的 (p115)

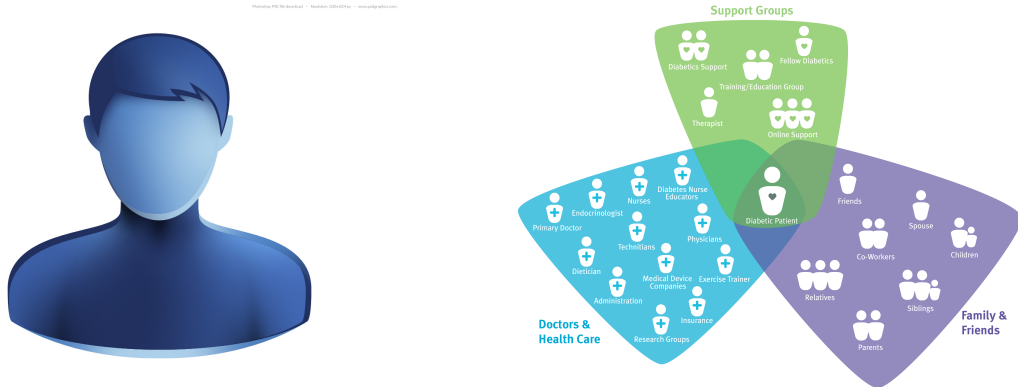
- ・ 進捗テストによって、ソフトウェアの信頼性目標に達成するまで予測
- ・ 開発効率の向上
- ・ よく使われる機能を特定し、それに着目して新しい機能や製品を開発
  - あまり使われていない機能は後で
  - 螺旋 (spiral) 開発やプロトタイプ開発などのプロセスに使える
- ・ 顧客とより良い交流とより深い関係
  - 顧客から製品の品質や機能に対する意見を把握できる
  - もっと細かい要求分析と仕様が可能になる
  - 顧客トレーニングを実行すべき項目を特定できる
- ・ 高い収益率の投資
  - コストの成長はソフトウェアの規模によって線形に近い

## 4 8.4 操作プロフィールを作成

### 4.1 8.4.1 一般的な方法と参加者

単一 OP ? 多数 OP ? (p115)

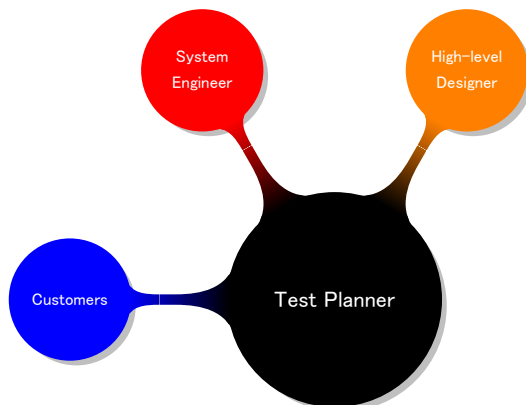
ユーザーの使用型によって



OP: 一般的な方法 (p116)

- ・ 実際に顧客のインストールで使用量の測定
  - 最も精度が高い
  - 新製品に既存のインストールがない
  - 顧客のプライバシーに問題がある
- ・ 顧客を調査
  - 精度は専門家の意見より高い
- ・ 専門家の経験や既存の製品に基づいて使用量を予測
  - コストは最も低い

OP: 一般的な参加者 (p117)



- ・ 計画やマーケティング担当者

- 顧客との主要な接触
- 顧客の懸念とその視点が反映されていることを確認
- ・ システムエンジニア
  - 製品に実装されるハイレベルの機能を含む製品全体の要件と仕様
- ・ ハイレベルデザイナー
  - 仕様に沿うハイレベルの製品設計をデザイン



## 4.2 8.4.2 Musa-1 操作プロフィールの開発過程

### Musa-1 OP の開発過程 (p117)

定義 2 (Musa-1). トップダウン手法, 同じ重要なユーザーグループに対して多数の操作プロフィール

In Musa's top-down approach or Musa-1 procedure, one OP is developed for each homogeneous group of users or operations

—[Musa(1993)]

### Musa-1 の手法で OP を得る手順 (p118)

1. 全種類の顧客に重みを付けて, 顧客のプロフィールを探し 2. 一種の顧客の中関連するユーザーの型と彼らの相対使用率を定義し, ユーザーのプロフィールを提示 3. よく使用される操作とその重みを見つけ, システムモードと関連プロフィールを定義 4. システムモードを分析し, ハイレベルの機能とその関連するプロフィールを決定 5. 細かい機能の使用率を決定

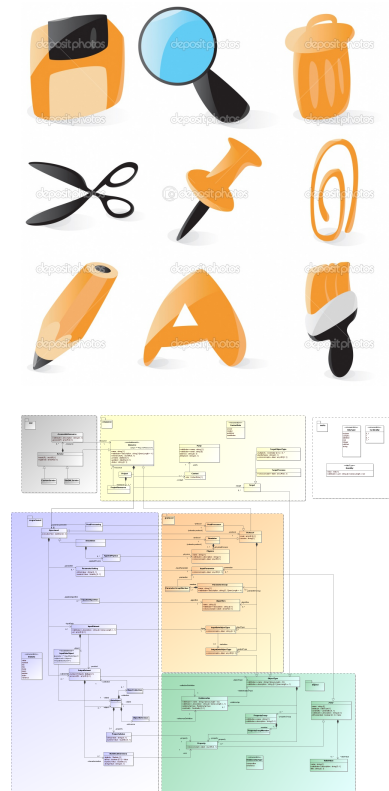
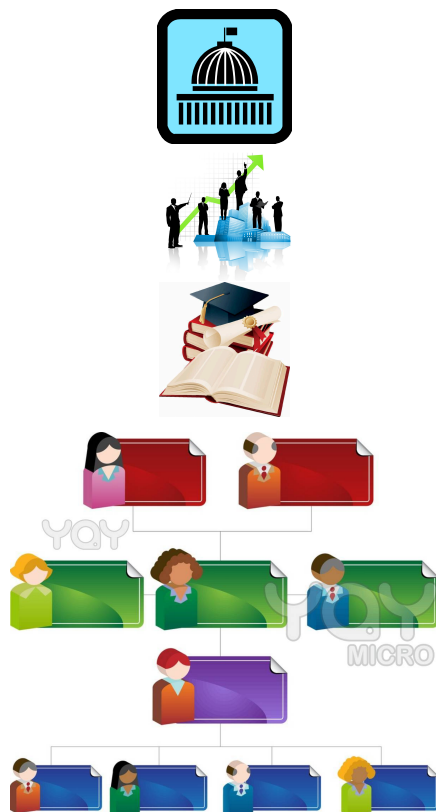


Table 8.5 A sample customer profile

| Customer Type | Weight |
|---------------|--------|
| corporation   | 0.5    |
| government    | 0.4    |
| education     | 0.05   |
| other         | 0.05   |

(a) 顧客操作プロフィールの例

Table 8.6 A sample user profile

| User Type   | User Profile by Customer Type |         |         |          |          | Overall User Profile |
|-------------|-------------------------------|---------|---------|----------|----------|----------------------|
|             | ctype weight                  | com 0.5 | gov 0.4 | edu 0.05 | etc 0.05 |                      |
| end user    |                               | 0.8     | 0.9     | 0.9      | 0.7      | 0.84                 |
| dba         |                               | 0.02    | 0.02    | 0.02     | 0.02     | 0.02                 |
| programmer  |                               | 0.18    | –       | –        | 0.28     | 0.104                |
| third party |                               | –       | 0.08    | 0.08     | –        | 0.036                |

(b) ユーザー操作プロフィールの例

図 9: 操作プロフィールを Musa-1 で開発する例

Musa-1: 例 (p118)

一貫性がある OP の計算 (p119)

もし一つの操作が二つの段階 (A, B) に分かれて、それぞれのプロフィール

$$p_i = \text{prob}(A = A_i)$$

$$p_j = \text{prob}(B = B_j)$$

操作全体のプロフィール

$$p_{ij} = \text{prob}(A = A_i, B = B_j) = p_i \times p_j$$

### 4.3 8.4.3 Musa-2 操作プロフィールの開発過程

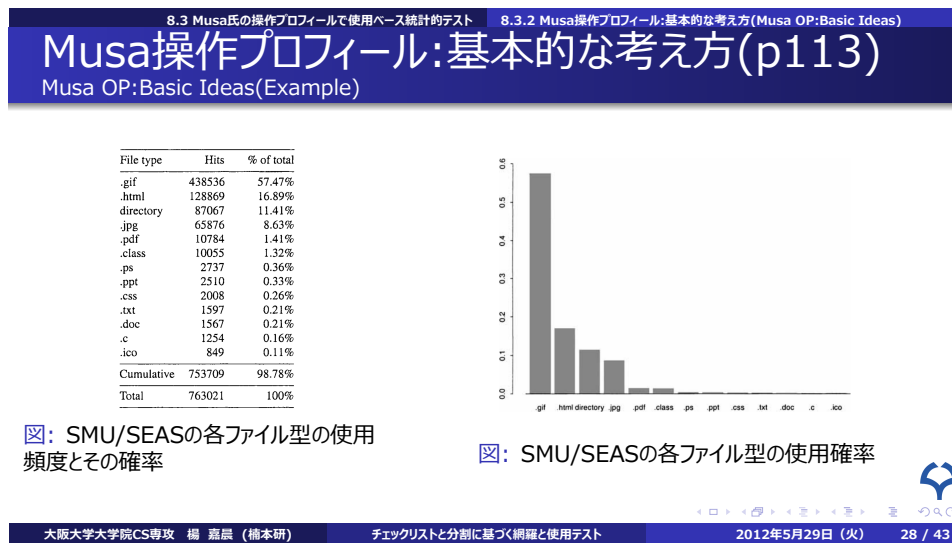
Musa-2 OP の開発過程 (p120)

定義 3 (Musa-2). 一つユーザー型に対して単一な操作プロフィール, もっと小さいデータソースに適用

for smaller products or ones with more homogeneous user population, one profile would probably be enough

1. 操作のイニシエータ (initiator) を決定
2. 表現の形式を決定: 表か図か
3. 操作配列を決定
4. 発生率の測定単位を決定
5. 発生する可能性を決め

## Musa-2 表現の形式：表 (p120)



## Musa-2 表現の形式：図 (p121)

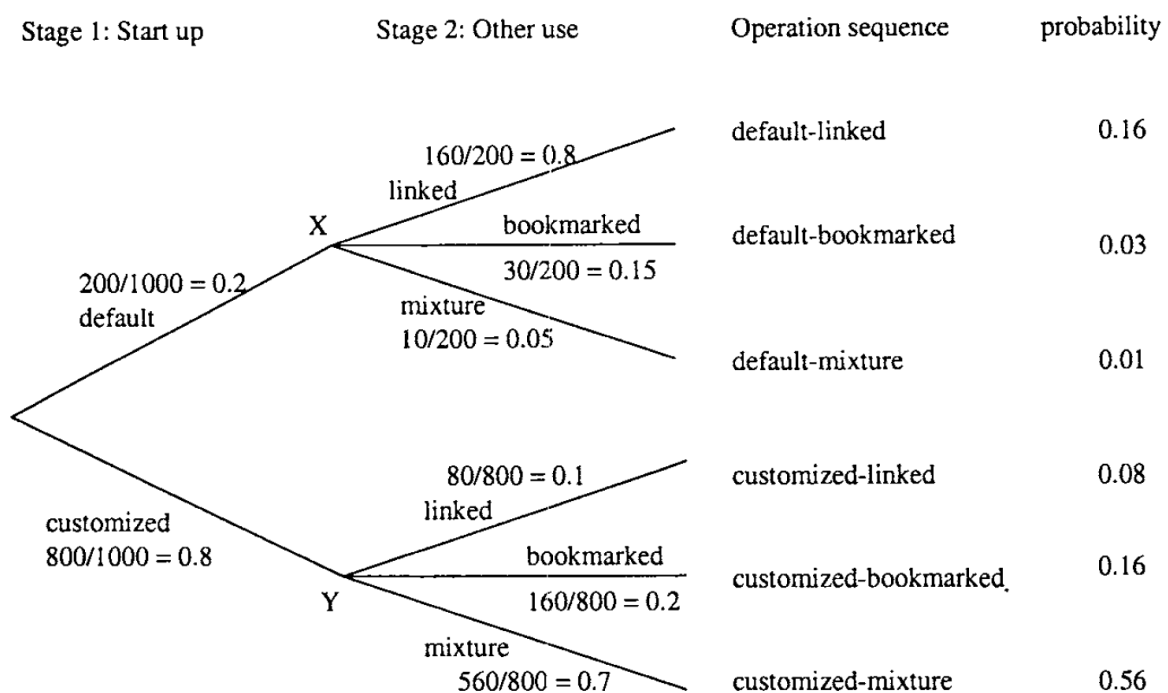


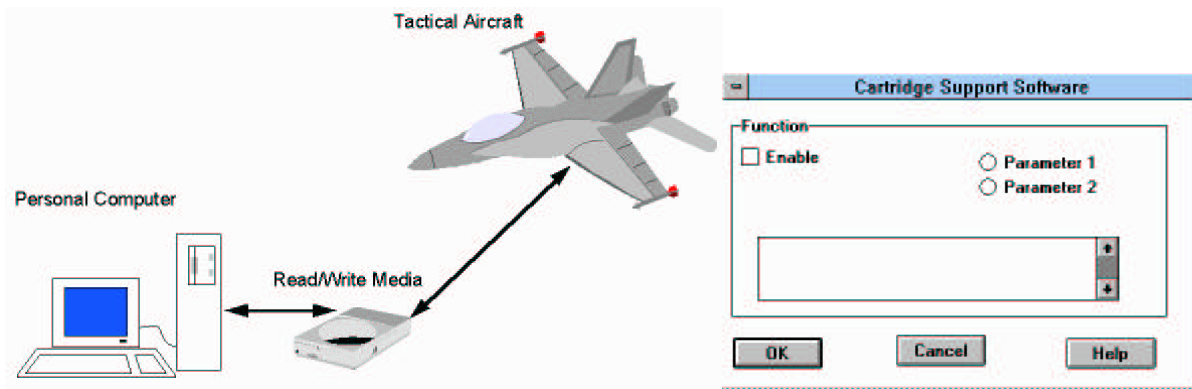
Figure 8.2 A tree-structured or graphical operational profile

## 5 8.5 Case Study: 交換支援ソフトウェアに使う操作プロフィール

## 5.1 8.5.1 背景と参加者

## 交換支援システム CSS の背景 (p121)

Lockheed Martin 戦術航空機 (Tactical Aircraft) 会社 (LMTAS) が開発した, 航空機要員に使われて, 任務の計画の媒介を交換するの支援システム (Cartridge Support Software, CSS)



### CSS の OP を開発する参加者 (p122)

参加者 [Chruscielski and Tian(1997)] は

- ・ Software Product Manager
- ・ Software Test Engineers
- ・ System Engineers

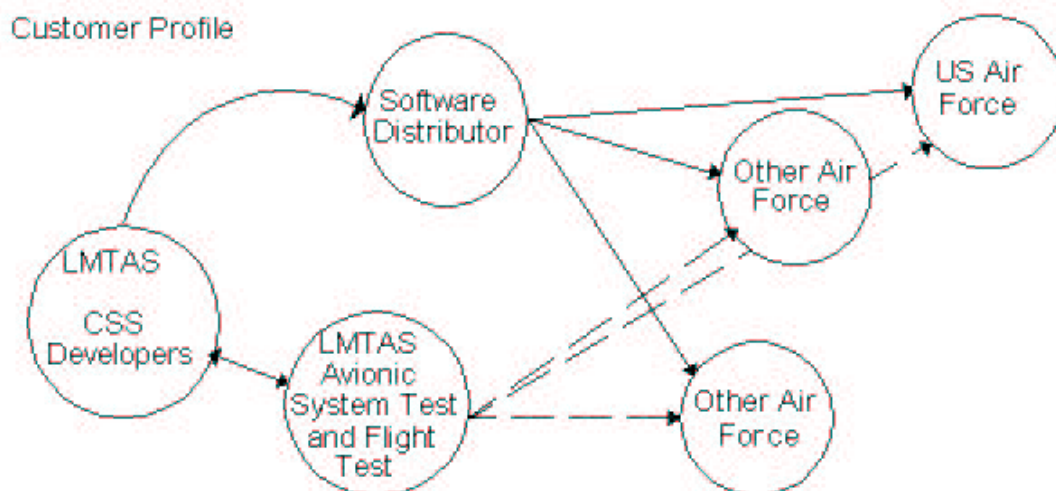
## 参考文献

[Chruscielski and Tian(1997)] K. Chruscielski and J. Tian, "An operational profile for the cartridge support software," in PROCEEDINGS The Eighth International Symposium On Software Reliability Engineering. IEEE, 1997, pp. 203–212.

### 5.2 8.5.2 五つのステップで OP 開発

Step 1&2: 顧客とユーザー(p122)

CSS の顧客は空軍



| User Group           | Marketing Concerns | Frequency of Use | Total Weighting Factor |
|----------------------|--------------------|------------------|------------------------|
| Air Force Pilot      | 0.85               | 0.05             | 0.45                   |
| Flight Test Support  | 0.10               | 0.80             | 0.45                   |
| Avionics System Test | 0.05               | 0.15             | 0.1                    |

図 10: CSS ユーザー・プロフィール

ユーザーの型は

1. 空軍のパイロット
2. 飛行テスト支援者
3. 航空機システムテスト者
4. システム管理者

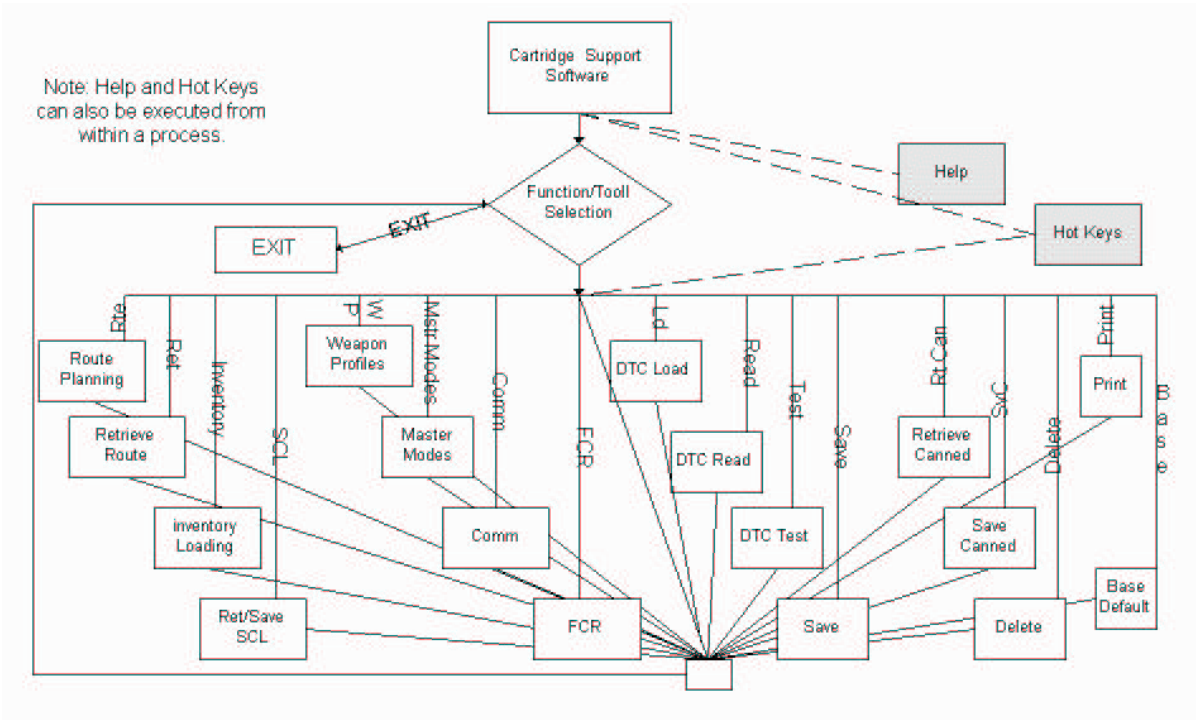
#### Step 3: システム・モード (p123)

CSS に見つけたシステム・モードは三種類に分かれています:

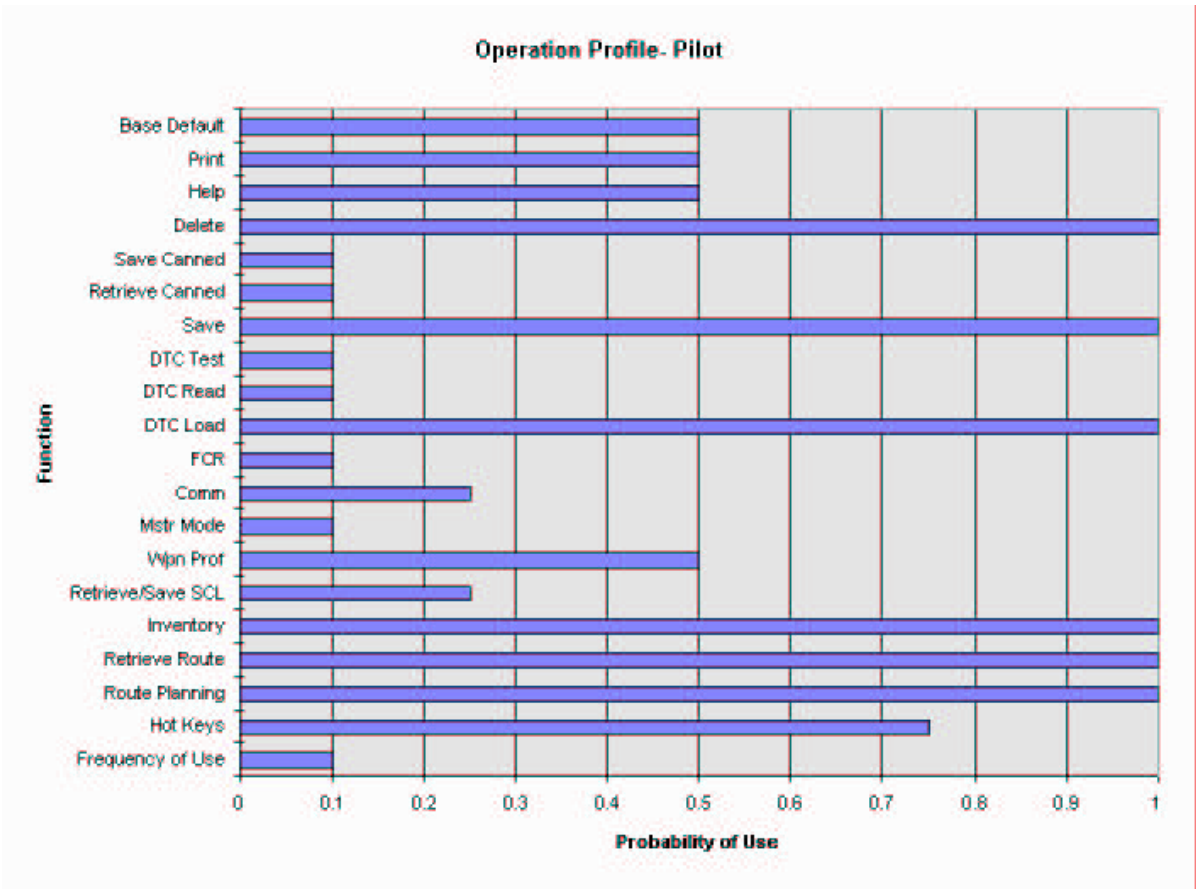
1. 飛行前の任務計画
2. 航空機システムテスト
3. システム管理

操作分析によると, この三つは区別しない。

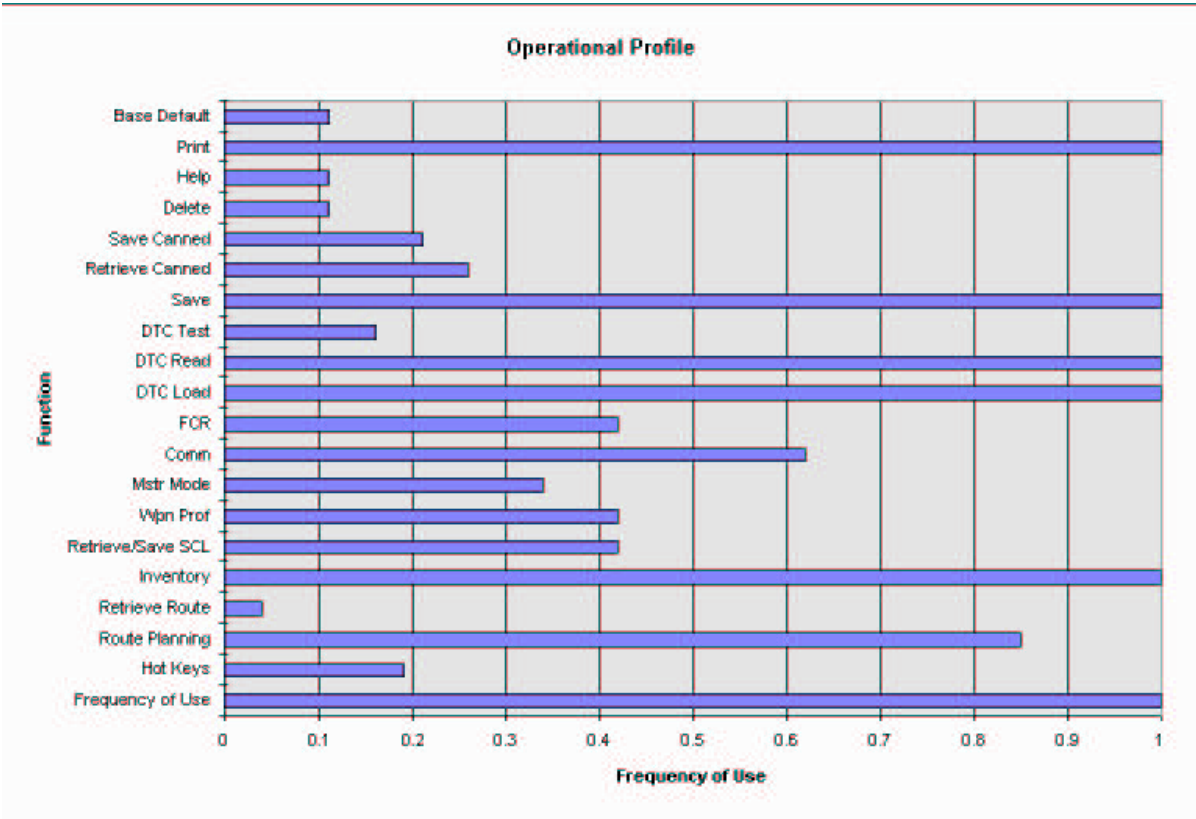
#### Step 4&5: 機能と操作 (p123)



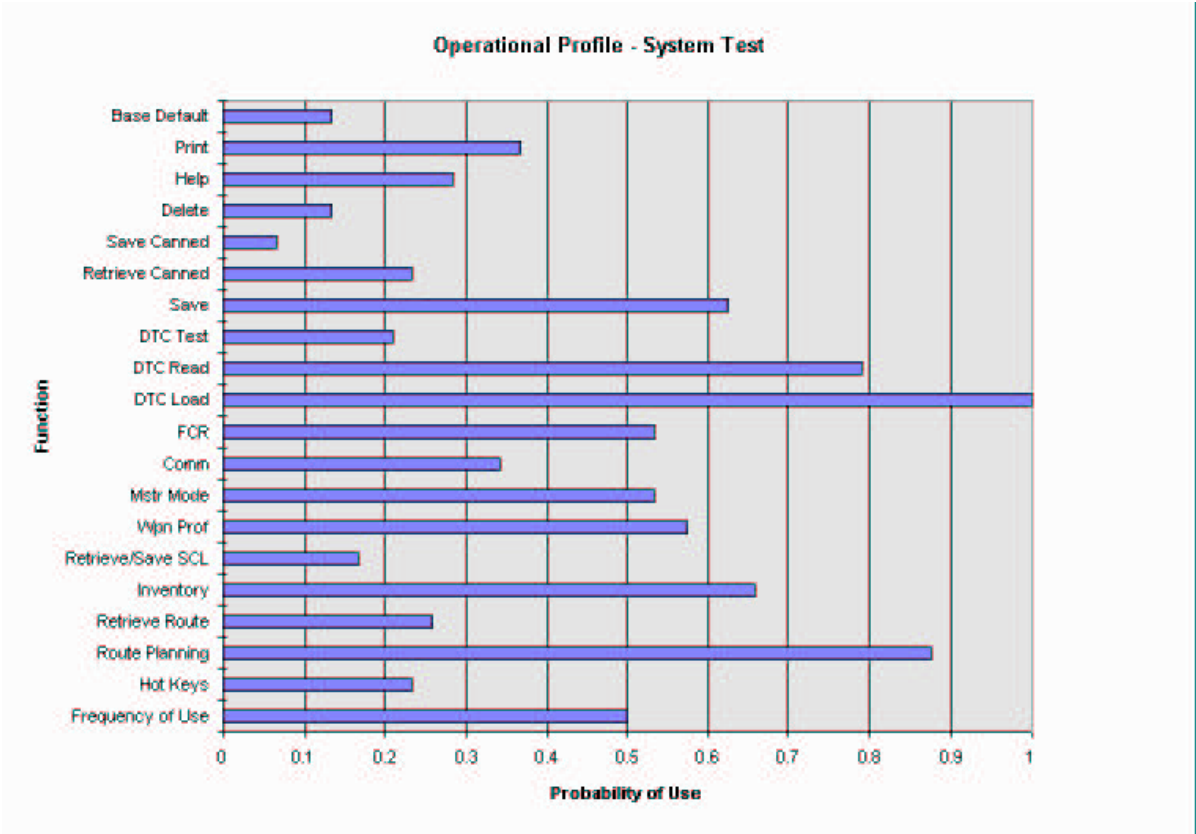
パイロットの OP



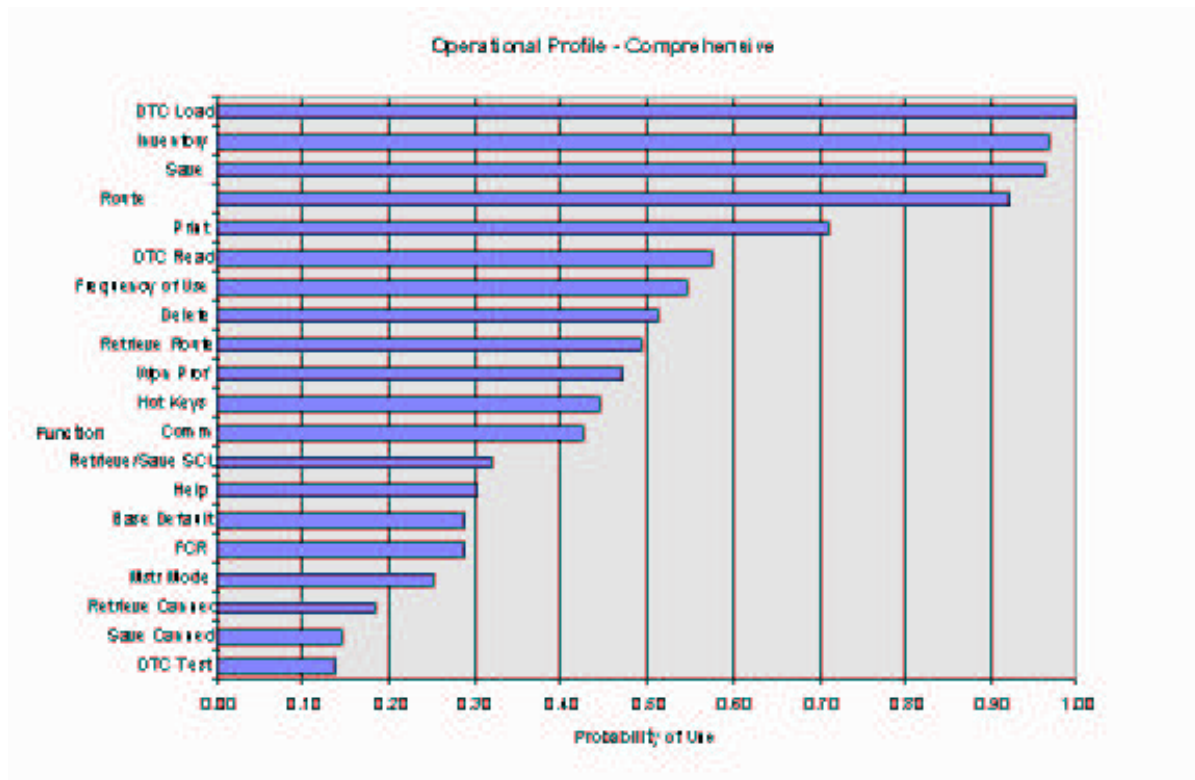
飛行テスト支援者の OP



システム・テスト者の OP



全部の OP



CSS 使用率による機能の分類 (p124)

| High                       | Medium-high                        | Medium-low                        | Low                       |
|----------------------------|------------------------------------|-----------------------------------|---------------------------|
| DTC Load                   | DTC Read                           | Wpn Prof                          | RetrCanned                |
| Inventory                  | Delete                             | Hot Keys                          | Save Canned               |
| Save                       | Retrieve Route                     | Comm                              | DTC Test                  |
| Route Planning             |                                    | Retr/Save SCL                     |                           |
| Print                      |                                    | Help                              |                           |
|                            |                                    | Base Default                      |                           |
|                            |                                    | FCR                               |                           |
|                            |                                    | Mstr Mode                         |                           |
| High usage<br>= 100% – 75% | Medium-high usage<br>= 74.9% – 50% | Medium-low usage<br>= 49.9% – 25% | Low usage<br>= 24.9% – 0% |

### 5.3 8.5 メトリック収集, 結果検証, 経験

#### メトリックを収集 (p124)

##### 1. SPM に製品の市場を把握する

- ・ 何週間に短いインタビュー

##### 2. SPM と議論し、ユーザプロフィールと機能プロフィールの要求を定義



- ・既存の CSS の機能設計は役に立った

### 3. ユーザー調査書を作った

- ・ System Engineers と Test Engineers の経験から
- ・ 二週間で

### 4. メールでのユーザー調査

### 5. 結果操作プロフィールの解釈

- ・ 前の図と表で表す

## 結果を検証 (p125)

意外の結果を解釈中に出た

- ・ ホットキーの使用は Medium-low
  - － ユーザーは既存のホットキーを依存
- ・ ヘルプ機能は Medium-low
  - － ユーザーはシステムにもっと詳しいと思った

## 6 8.6 まどめ

### 6.1 8.6 まどめ

#### まどめ (p125)

1. チェックリスト・テスト, とその制限を紹介した
2. 分割に基づくテストモデルを紹介した
  - ・ 9 章に入力ドメインによる分割と境界テストを紹介する
3. Musa による操作プロフィール (OP) を紹介した
  - ・ Musa-1 多数 OP
  - ・ Musa-2 単一 OP
  - ・ CSS の事例

もっと複雑のプログラムに使う FSM に基づくモデルを 10, 11 章に紹介する

#### 参考文献

## 参考文献

[Musa(1993)] J. Musa, “Operational profiles in software-reliability engineering,” Software, IEEE, vol. 10, no. 2, pp. 14–32, 1993.

[Chruscielski and Tian(1997)] K. Chruscielski and J. Tian, “An operational profile for the cartridge support software,” in PROCEEDINGS The Eighth International Symposium On Software Reliability Engineering. IEEE, 1997, pp. 203–212.