

# 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

大阪大学大学院 コンピュータサイエンス専攻 楠本研究室

2012年5月29日 (火)



# 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



# Ad hocテストとランダムテスト(p104)

Ad hoc Testing & Random Testing



図：Ad hoc Testing

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



*Development of tests using a black box method, in which test cases are chosen to match the functional cross-section, usually using an algorithm of pseudo-random selection.*

図：Random Testing

# チェックリストに基づくテスト(p104)

Testing with checklists

- 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
- 
- abnormal termination
  - backup and restore
  - communication
  - co-existence
  - file I/O
  - gateway
  - index management
  - installation
  - logging and recovery
  - locking
  - migration
  - stress

図： チェックリスト



# 階層チェックリスト(p105)

Hierarchical Checklists

- 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



- ① ハイレベル項目1
  - ① ローレベル項目1
  - ② ローレベル項目2
  - ③ ローレベル項目3
- ② ハイレベル項目2
  - ① ローレベル項目4
  - ② ローレベル項目5
- ③ ハイレベル項目3
  - ① ローレベル項目6
  - ② ローレベル項目7
  - ③ ローレベル項目8



# 複合チェックリスト(p105)

Combined Checklists

要求1 要求2 要求3

## ① 部品1

- ① 要求1
- ② 要求2
- ③ 要求3

## ② 部品2

- ① 要求1
- ② 要求2
- ③ 要求3

## ③ 部品3

- ① 要求1
- ② 要求2
- ③ 要求3

Component	Standards Items			
	$s_1$	$s_2$	$\cdots$	$s_n$
$c_1$				
$c_2$				
$\vdots$				
$c_m$				

図：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



# チェックリストの問題点(p106)

Problems General Checklists

- 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)

Limitations of General Checklists

- 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



# 分割カバレッジテスト(p107)

Testing for Partition Coverage

- 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



# Motivational Examples(p107)

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

その解を求めるとき

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

全部可能な入力の組み合わせ

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

Test Case	Condition $d = b^2 - 4ac$	Input		
		a	b	c
1	$d > 0$	1	2	-1
2	$d = 0$	1	2	1
3	$d < 0$	1	2	3

図 :  $ax^2 + bx + c = 0$  のテストケース



# 分割 : 概念と定義(p108)

Partition: Concepts and Definitions

- ① 分割した集合は相互に排他的

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

- ② 分割した集合の和集合は全体の集合

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

分割した集合は同値類になる

対称律(symmetric)

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

推移律(transitive)

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

反射律(reflexive)

$R(a, a)$  が常に成り立つ



# 分割テストの網羅範囲の決定(p109)

Testing Decisions for Partition Coverage

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

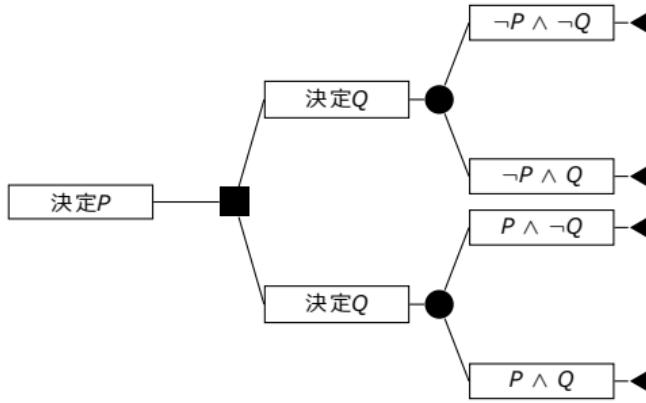
- 論理変数を論理演算子に繋がれる論理述語による
- 数的な変数を比較演算子によって比較する
- 論理述語や比較演算によって入力区間を分割
- 決定木を使う

## ③ 1と2の組み合わせ



# 決定木によって分割(p110)

Testing Decisions based on Decision Tree



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 }
  
```



# 統計的UBT適用する場合(p111)

The Cases for Usage-based Statistical Testing

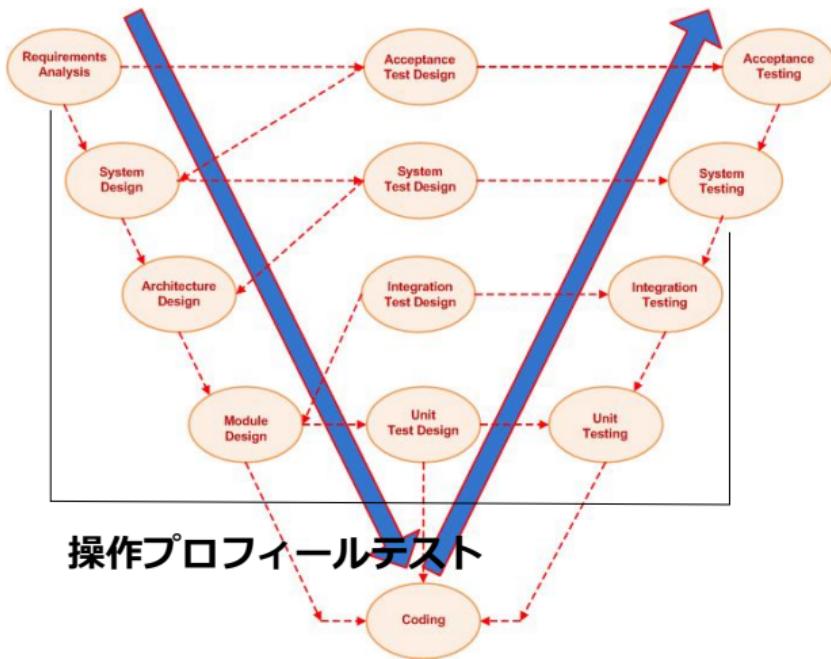
操作	頻度(%)	
スブドメイン1		
-操作1	1%	
-操作2	72%	✓
-操作3	2%	
スブドメイン2		
-操作4	13%	✓
-操作5	2%	
スブドメイン3		
-操作6	3%	
-操作7	7%	✓

- 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, 操作プロファイル (OP, Operational Profile) に変換
- ③ 操作プロファイ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)

## The Period of Development Process that Involves Operational Profiles



図：操作プロフィールを開発プロセスに導入する時期



# MusaによるOPの定義(p112)

Musa OP:Basic Ideas(Defination)

## 定義 (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)]

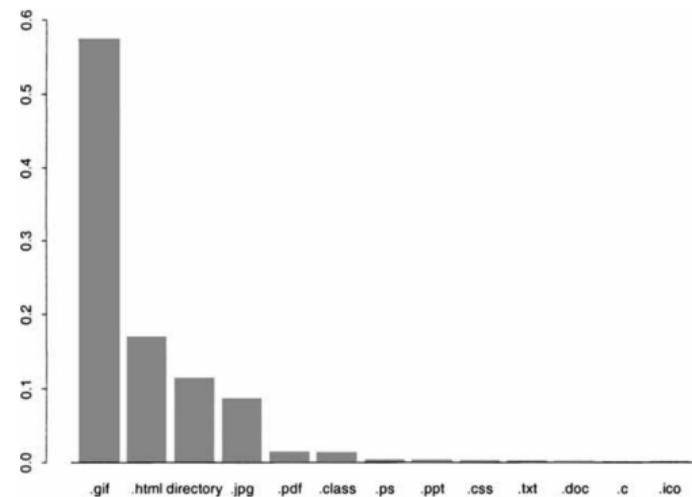
- J. Musa, ``Operational profiles in software-reliability engineering," *Software, IEEE*, vol. 10, no. 2, pp. 14--32, 1993.



# Musa OP: 基本的な考え方(p113)

Musa OP: Basic Ideas(Example)

File type	Hits	% of total
.gif	438536	57.47%
.html	128869	16.89%
directory	87067	11.41%
.jpg	65876	8.63%
.pdf	10784	1.41%
.class	10055	1.32%
.ps	2737	0.36%
.ppt	2510	0.33%
.css	2008	0.26%
.txt	1597	0.21%
.doc	1567	0.21%
.c	1254	0.16%
.ico	849	0.11%
Cumulative	753709	98.78%
Total	763021	100%



図：SMU/SEASの各ファイル型の使用頻度とその確率

図：SMU/SEASの各ファイル型の使用確率



# 操作プロフィールで統計的なテスト(p114)

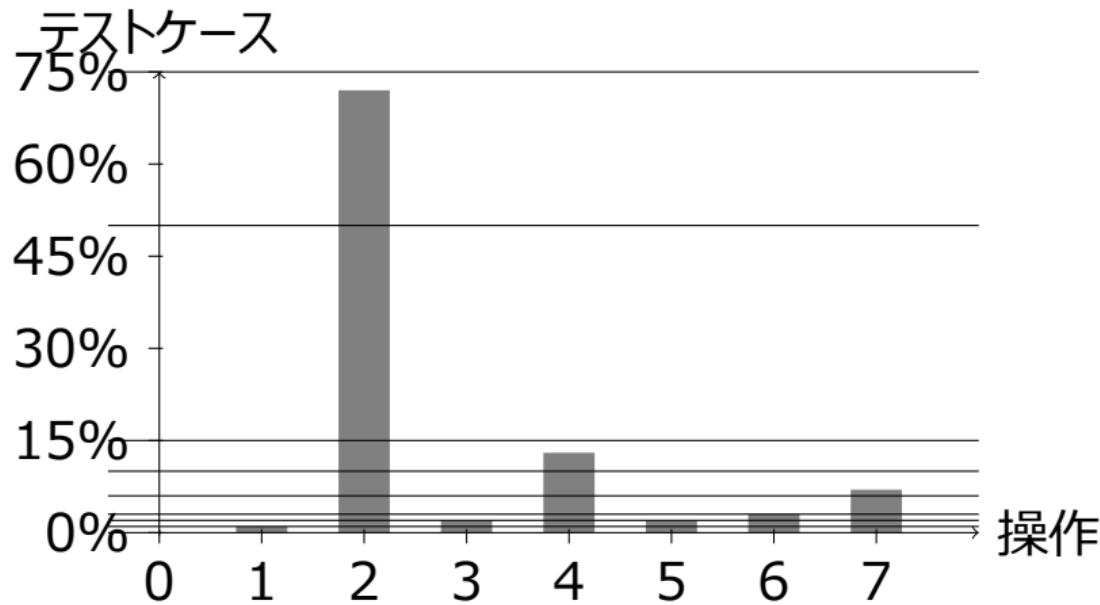
Using OPs for Statistical Testing

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

Using OPs for Progressive Testing



# 操作プロフィール：その他の目的(p115)

OPs Used in Other Situations for Various Purpose

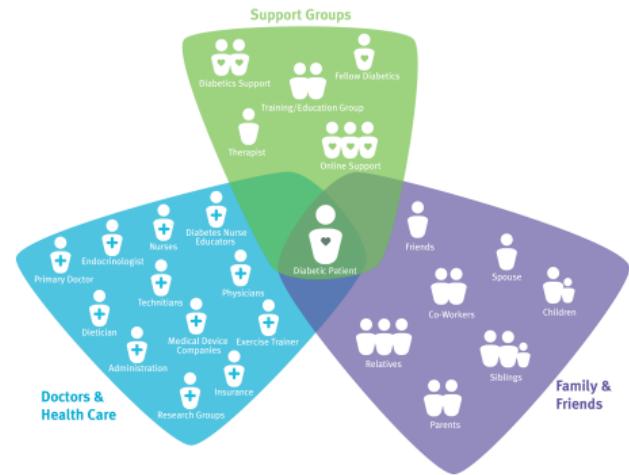
- 進捗テストによって、ソフトウェアの信頼性目標に達成するまで予測
- 開発効率の向上
- よく使われる機能を特定し、それに着目して新しい機能や製品を開発
  - あまり使われていない機能は後で
  - 螺旋(spiral)開発やプロトタイプ開発などのプロセスに使える
- 顧客とより良い交流とより深い関係
  - 顧客から製品の品質や機能に対する意見を把握できる
  - もっと細かい要求分析と仕様が可能になる
  - 顧客トレーニングを実行すべき項目を特定できる
- 高い收益率の投資
  - コストの成長はソフトウェアの規模によって線形に近い



# 単一OP？多数OP？(p115)

Single Operational Profiles? Multiple Operational Profiles?

ユーザーの使用型によって



# OP: 一般的な方法(p116)

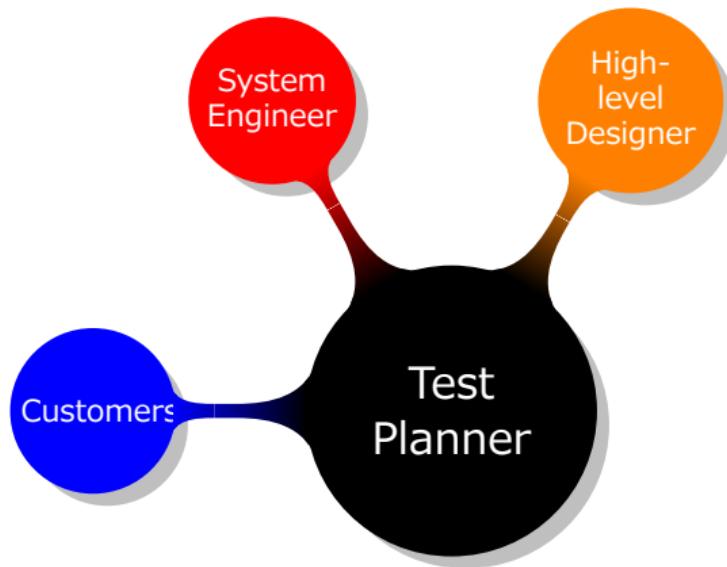
OP: Generic Methods

- 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)

OP: Generic Participants



- 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



# Musa-1 OPの開発過程(p117)

OP development procedure: Musa-1

## 定義 (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*

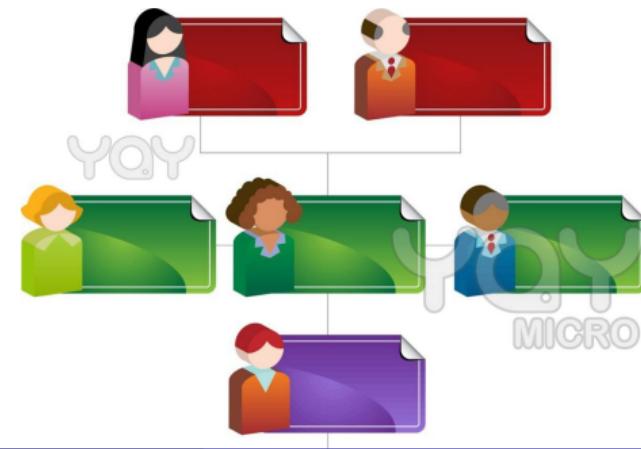
---[Musa(1993)]



# Musa-1の手法でOPを得る手順(p118)

Steps to develop OP, Musa-1

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れの重みを見つけ、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



# Musa-1:例(p118)

Sample for Musa-1

**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	gov	edu	etc	
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：操作プロフィールをMusa-1で開発する例



# 一貫性があるOPの計算(p119)

Calculate Profile for Uniform Operational Stages

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

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

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

操作全体のプロファイル

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



# Musa-2 OPの開発過程(p120)

OP development procedure: Musa-2

## 定義 (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)

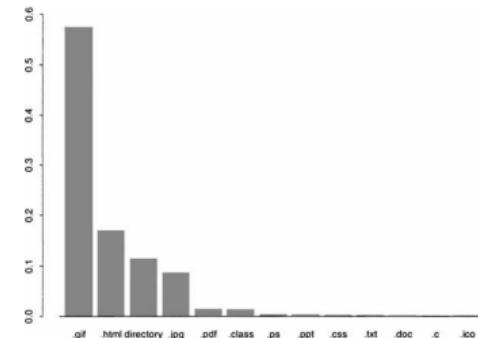
Musa-2 Representation: Tabular

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

8.3.2 Musa操作プロフィール:基本的な考え方(Musa OP:Basic Ideas)

## Musa操作プロフィール: 基本的な考え方(p113) Musa OP:Basic Ideas(Example)

File type	Hits	% of total
.gif	438536	57.47%
.html	128869	16.89%
directory	87067	11.41%
.jpg	65876	8.63%
.pdf	10784	1.41%
.class	10055	1.32%
.ps	2737	0.36%
.ppt	2510	0.33%
.css	2008	0.26%
.txt	1597	0.21%
.doc	1567	0.21%
.c	1254	0.16%
.ico	849	0.11%
Cumulative	753709	98.78%
Total	763021	100%



図：SMU/SEASの各ファイル型の使用頻度とその確率

図：SMU/SEASの各ファイル型の使用確率



# Musa-2 表現の形式：図(p121)

Musa-2 Representation: Graphical

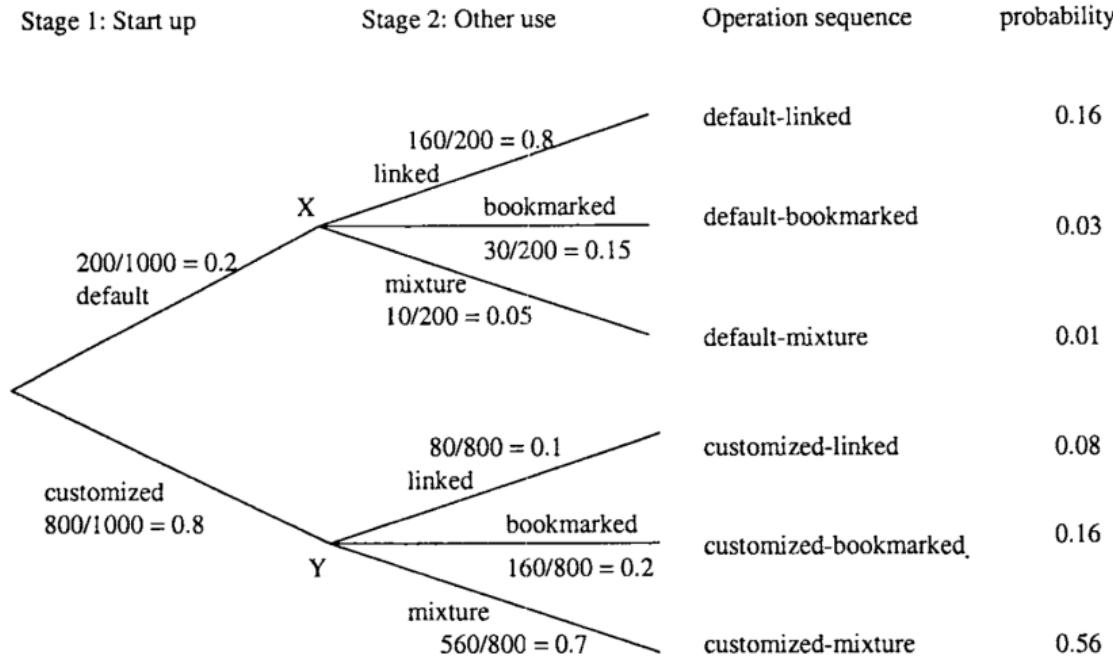


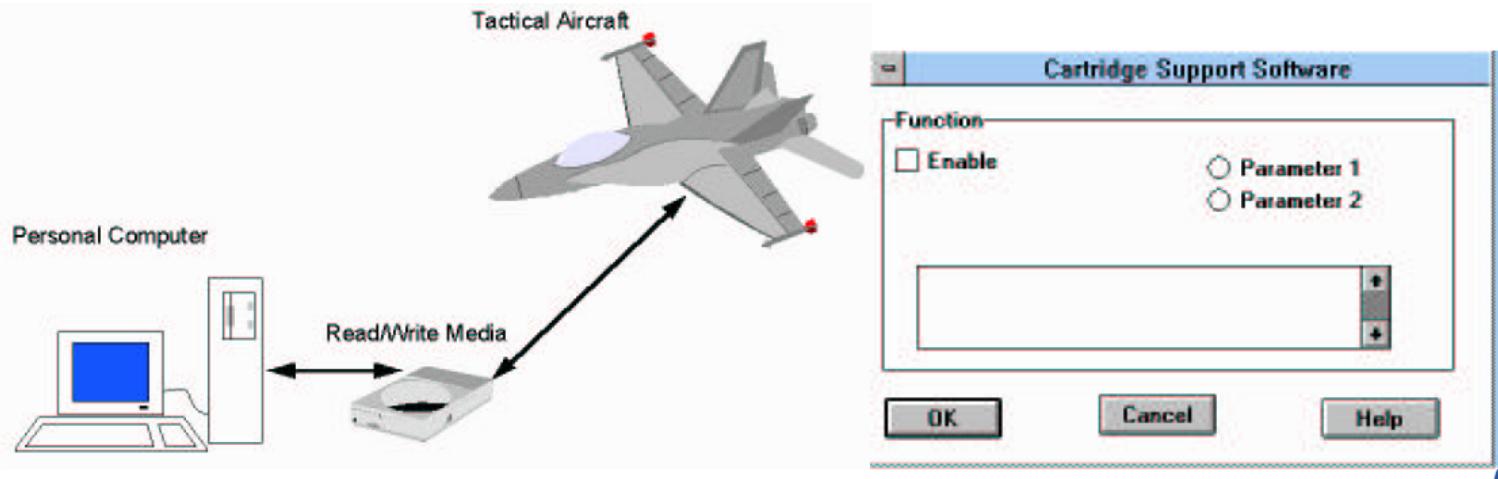
Figure 8.2 A tree-structured or graphical operational profile



# 交換支援システムCSSの背景(p121)

Background for Cartridge Support System

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)

Participants in Developing OP of CSS

参加者[Chruscielski and Tian(1997)]は

- Software Product Manager
- Software Test Engineers
- System Engineers

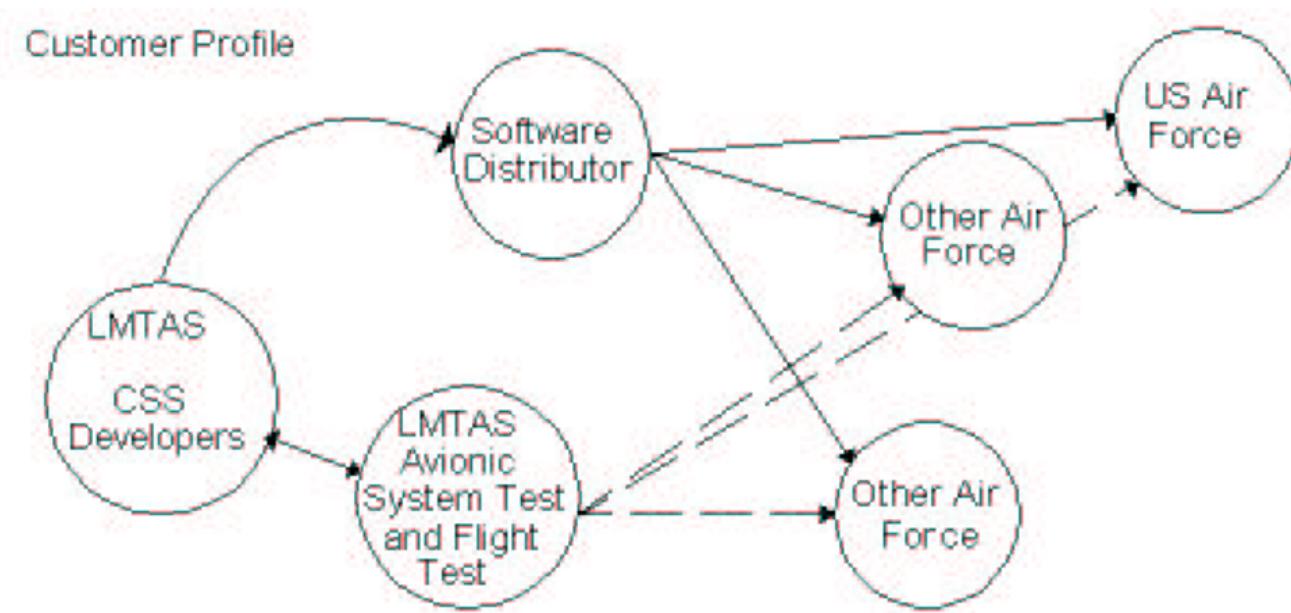
 K. Chruscielski and J. Tian, ``An operational profile for the cartridge support software," in *PROCEEDINGS The Eighth International Symposium On Software Reliability Engineering*. IEEE, 1997, pp. 203--212.



# Step 1&2: 顧客とユーザー(p122)

Step 1&2: Customers & Users

CSSの顧客は空軍



# Step 3: システム・モード(p123)

Step 3: System Modes

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：

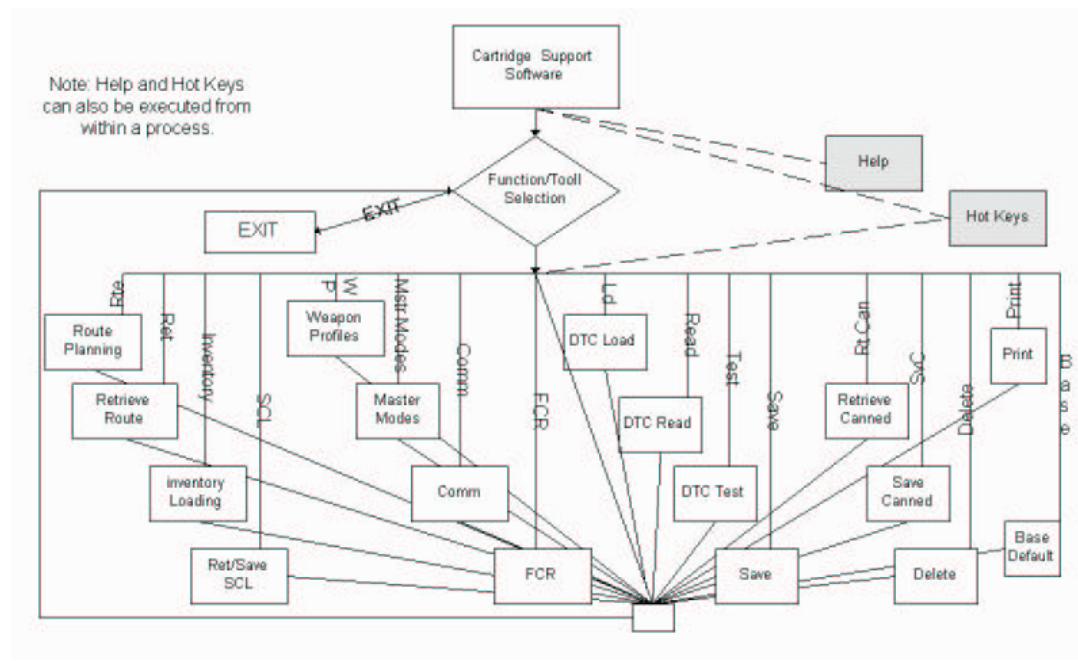
- ① 飛行前の任務計画
- ② 航空機システムテスト
- ③ システム管理

操作分析によると、この三つは区別しない。



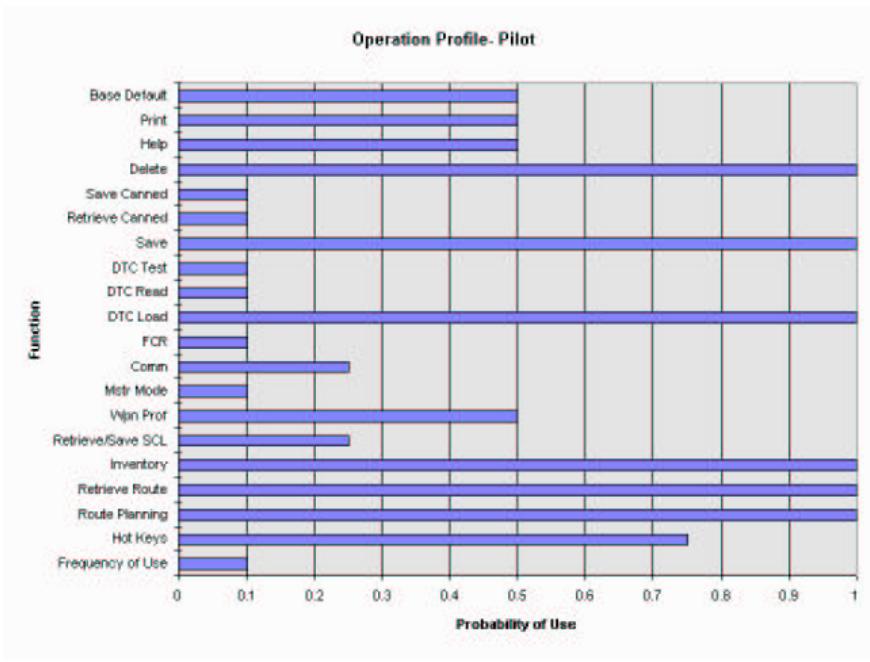
# Step 4&5: 機能と操作(p123)

Step 4&5: High-level functions and low-level operations



# パイロットのOP

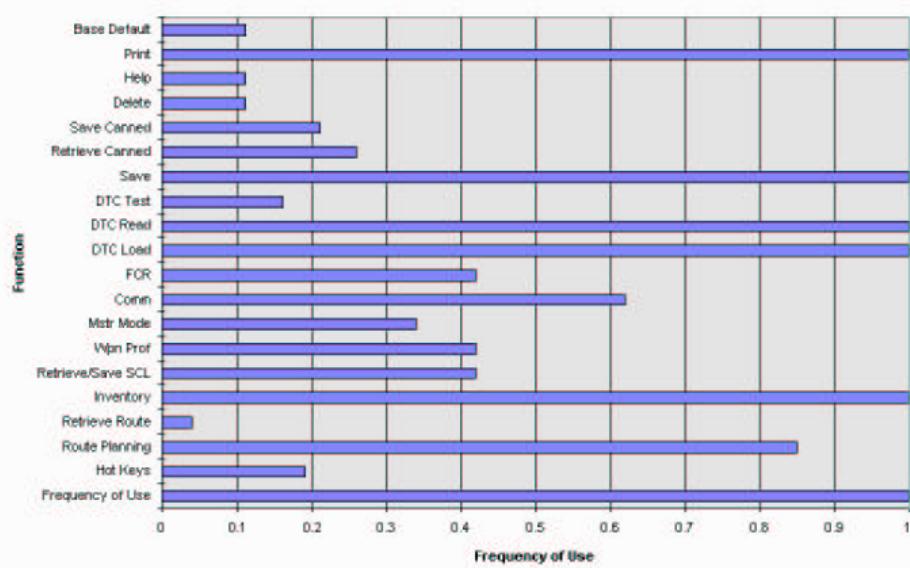
OP: Pilot



# 飛行テスト支援者のOP

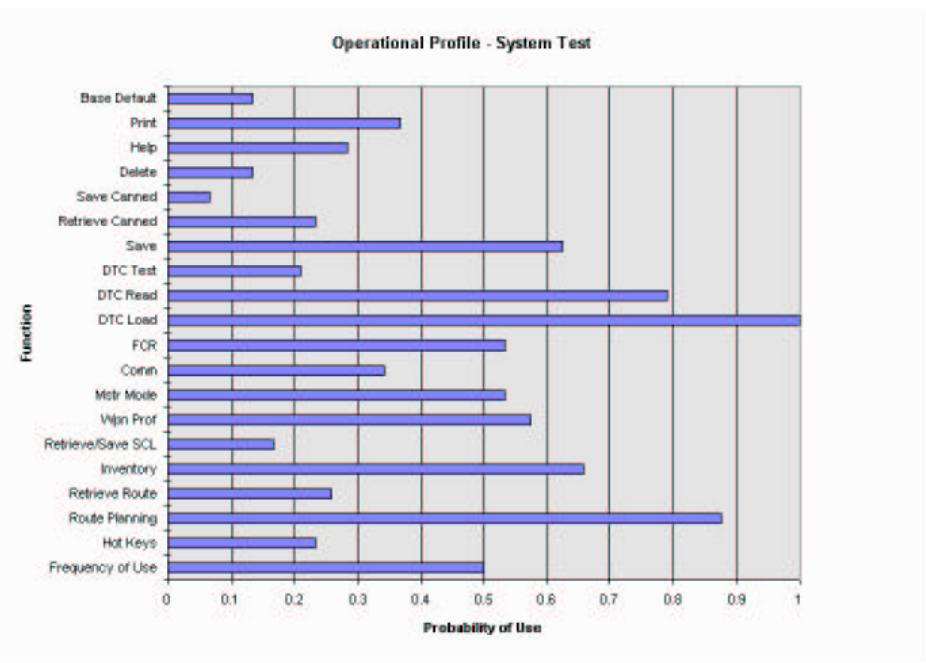
OP: Flight Test Support

Operational Profile



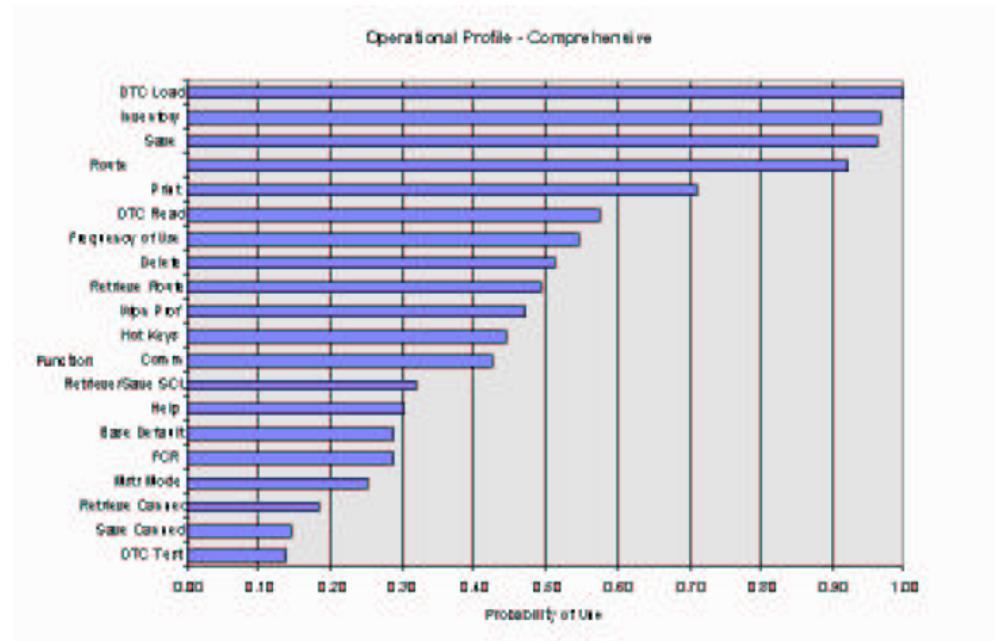
# システム・テスト者のOP

OP: System Test



# 全部のOP

OP: Comprehensive



# CSS使用率による機能の分類(p124)

CSS functions classified according to usage probabilities

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 = 100% - 75%	Medium-high usage = 74.9% - 50%	Medium-low usage = 49.9% - 25%	Low usage = 24.9% - 0%



# メトリックを収集(p124)

Metrics collections

- 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、ユーザー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)

## Result Validation

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



# まどめ(p125)

## Concluding Remarks

- 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

## Reference

- J. Musa, ``Operational profiles in software-reliability engineering," *Software, IEEE*, vol. 10, no. 2, pp. 14--32, 1993.
- 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.

