

## ユーザマニュアルのデータベース化とその自動引用

松田 孝子

東北大学大型計算機センター

エンドユーザの間では、ソフトウェアのマニュアルの読みにくさ、利用に関連する情報の入手しにくさが大きな問題になっている。これを解決する一つの方法として、著者らは、ユーザが目を読むマニュアルの量を減らし、知的機能によりソフトウェア群を統合提供するエキスパートシステムVCAPを実現した。ここでは、VCAPの中で採用したマニュアル情報のデータベース化とそれを自動引用する方式について報告する。

マニュアルに記載される情報のうちユーザが読む必要のない規約等をデータベース化し、ユーザ個人で所有する利用のための情報を知識ベース化して共有できるようにした。情報の記述・格納は各ソフトウェアの仕様に依存しない方法で行った。これらの格納情報を知識ベースの情報を入子として自動的に引用・関係しユーザに提供できるようにした。事例としてバッチ処理用の統計パッケージSPSS (SPSS Inc.製)とSTATPAC (NEC製)を用いた。パッケージの機能照会、パッケージのジョブデックの自動生成、ジョブの自動遂行を可能にした。ACOS1000上に、論理型言語ShapeUp、データベース管理システムCOODを用いてインプリメントした。

Formation of the Database of User Manuals and Its Automatic Referring

Takako MATSUDA

Computer Center, Tohoku University

2-1-1 Katahira, Sendai 980 Japan

A serious problem of computer uses today is that it has become hard for nonprofessional end-users to understand user manuals of softwares and to acquire knowledge for use of them. The authors have developed an expert system VCAP, which was designed in order to establish an intelligent user-interface of software and relieve end-users of a laborious and troublesome work to refer to the thick manuals.

User protocol described in manuals is stored in the database by COOD, and empirical knowledge obtained through practical use, in the knowledge base by ShapeUp. The specifications of softwares are represented and stored with the same formation. This expert system refers to the built-in information automatically, constructs an actual job deck for end-users and submits this job. For an implementation of this system, we dealt with statistical packages, SPSS and STATPAC, in batch-mode processing.

## 1. はじめに

情報処理の専任要員ではないエンドユーザが、コンピュータを直接操作して各自の仕事を遂行することが一般的になっている。このような状況に呼応して、多種多様なソフトウェアが利用可能である。しかし、この一見高度に普及したコンピュータ利用環境の中で、エンドユーザが実際に利用できるソフトウェアは極めて限られており、システムの発展とは逆にその使いにくさが大きな問題になっている。この第一の原因は、ソフトウェアの設計開発者が真の利用を十分に把握しない(できない)ことによる。この結果、「マニュアルの読みにくさ」、「ソフトウェアの使い勝手の悪さ」、「利用情報の入手しにくさ」が、いたるところで話題に上るようになった。

これに対処するには、ソフトウェアのユーザインタフェースを本質的な面から改善する必要がある。この解決の一つとして、筆者らは、個別に提供されるソフトウェアについて、

- ユーザが目で見つめるマニュアルの量を減らす。
  - 知的機能によりソフトウェア群を統合提供する。
- ことを計画し、プロトタイプシステムVCAP[1-2]を設計開発した。本発表では、この中で実現したユーザマニュアルのデータベース化とその自動引用について報告する。

## 2. ユーザマニュアルの情報

ソフトウェアのマニュアルは、開発時に、「内部仕様」および「外部仕様」として記述される。前者は、後日このソフトウェアを維持管理、改訂するために開発者によって使用されるものである。後者は、ユーザ向けの利用情報の原典になる。外部仕様からさらに解説的なテキストが作成される場合もあり、ユーザに合う言葉(多国語)に翻訳されることもある。

ユーザマニュアルに記載される情報はつぎのものからなる。

- x1. このソフトウェアの全体的な紹介
- x2. 機能別の紹介
- x3. 使用規約
- x4. 運用OSの中での利用規約
- x5. 使用上の制限および注意事項
- x6. エラーメッセージとその処置
- x7. 使用例題

このうち、x4の情報は計算センターなどの運用体ごとに異なるのが一般的である。また、マニュアル記

述者がユーザの利用パターンを予測できない(しない)ことから、x5, x6に関する情報の記述漏れや不適切が実用システムでは日常的である。x7はエンドユーザにとって最も有益な情報であるが、マニュアルに記載される例は不完全かつ不十分である。

この実情を補うために、マニュアルに加えて、つぎのような方法で情報の伝達、交換が行われている。

- y1. ソフトウェア開発者から提供されるマニュアル
- y2. システム運用者による利用の手引き、ニュース
- y3. ユーザによる利用紹介
- y4. 個人レベルの利用事例の情報交換

このうち、y1からy3は印刷物として公開されるが、y4はユーザ個人のもので利用終了後消失してしまう。また、いずれの情報もひとまとまりになっていないので、ユーザは必要な情報(x, y)の中から収集するために手間取ることが多い。

## 3. マニュアルのオンライン化

### 3.1 設計の目標

ソフトウェアの量の増加に比例してマニュアルの量が増え、ソフトウェアの改訂に応じてマニュアルの改版も必要である。従来のような印刷物によるマニュアルの提供には執筆と供給に労力上の限界があるので、マニュアルをオンライン化する方策がとられる。しかし、効率的に作成される大量の部厚いマニュアルをオンラインで提供しても、これを読んで理解する人間の労力について配慮しなければ、問題の解決をはかることはできない。

マニュアル等の利用情報のオンライン化を計画するには、つぎの目標をもつ必要がある。

- ① ユーザが読む必要のある情報と読む必要のない情報を区別する。通読する情報は変更されることの少ないもののみとする。
- ② 読む必要のない情報をオンライン化し、システムが自動解読する。
- ③ ユーザが読む情報は印刷物として提供する。提供方法はオンラインでもオフラインでもよい。
- ④ 個人レベルで所有している利用のための情報を共有できるように蓄積する。
- ⑤ これらの情報の係わりを取り、ユーザが情報収集の手間を省けるようにシステムを設計する。

これにより、印刷物としてのマニュアルの量を減らし、ユーザが読んで理解しなければならない情報量を極力少なくし、情報の入手を容易にする。

### 3.2 方式の設計

ソフトウェア利用のタイプを分類する。

- (1) プログラムを作成するもの
- (2) データを作成するもの
- (3) ソフトウェアを指令で使用するもの

(OSの利用規約, 第4世代言語による処理)

利用のタイプによってシステムの設計方式が異なるので, ここではソフトウェアの種類が多かつシステム依存度が高い(3)のタイプについて考える。第4世代言語のソフトウェアの中には, 特にOAソフトを中心に, マニュアルの引用を殆ど必要としない機能統合型のソフトウェアがあるが, ここでは大部のマニュアルを必要とするソフトウェアを取り上げる。

システムの設計はつぎの方式で行う。ここで, ソフトウェア(個体)をS, Sの開発者をHp, Sの運用提供者をHs, エンドユーザをHuとする。

- Hpは, 開発したソフトウェアのマニュアルのうちユーザに連絡してもらうものを印刷し, 利用時に引用するだけでよいものをデータベースに蓄積する。
- Hsは, ソフトウェアの運用上の利用規約をデータベースに蓄積し, 利用案内簿を管理する。
- Huは, 各自の利用例をシステムに蓄積する。
- Hsは, HuやHpの協力を得て利用事例集を収集し, 知識ベースに蓄積する。
- システムはユーザとの対話の中で, ユーザが要求する処理を推論する機構をもつ。
- 蓄積された利用例を用いてマニュアルを自動引用する処理系を設け, 推論機構と関係させる。

以上より設計するシステムは, 個々のソフトウェアの利用規約を内部生成するため, 異なるソフトウェアを同一のインタフェースで利用可能とするエキスパートシステムとなる。図1にこの構成を示す。

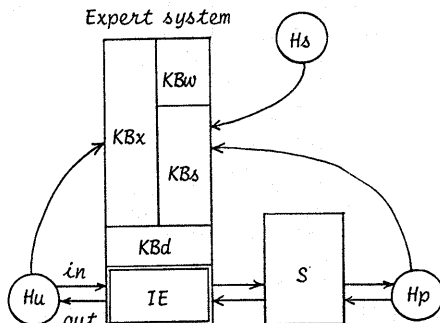


図1 ソフトウェア利用エキスパートシステム

図中の各要素はつぎのものである。

- 知識データベースKB

- KBs.....Sの外部仕様。Hpがトップダウンに作成するデータベース
- KBw.....Sの運用規約。Hsがトップダウンに作成するデータベース
- KBx.....Sの利用例。Hp, Hs, Huがボトムアップに作成する利用例集
- KBd.....ソフトウェアの管理目録。Hsがトップダウンに作成するインデックス

• 推論機構および処理系IE

### 4. 利用情報の蓄積

#### 4.1 マニュアルのデータベース化

マニュアル記載の情報のうちx2, x3, x4をデータベース化する。これらはソフトウェアの開発時点で確定する利用規約で, トップダウンに管理可能なものであり, ユーザは単にこれに従うものである。x5はその表す意味に応じてx1からx4に付随させる。x6は各ソフトウェアのHelp機能として実現するものとし, 本システムとは別に扱う。x7は4.2で述べる。

マニュアルのデータベース化の事例として, バッチ処理用の統計パッケージSPSSとSTATPACを用いた例を示す。格納データはつぎのものである。

- ① ジョブ制御指令(JCL)
- ② SPSSの手続き制御指令
- ③ STATPACの手続き制御指令

これらを別々のテーブルにし, 構成アイテムは自動引用を考慮して同じ名前前で登録した。CODを用いたデータ構造例を図2に示す。テーブルCIJはKBwに, CICSPSSとCICSTATはKBsに対応する。

```
DDL;
DATABASE CI :Control Information of VCAP;
TABLE CIJ
NO :Job Control Language of Application Packages;
(I3) UNIQUE :Serial no. of card set;
CNAME (A8) :Card set name;
TITLE (A30) :Title of card set;
FUNC(3) (A78) :Function of card set;
FORM(10) (A35) :Card form;
PARM(10) (A5) :Parameter;
EXPL(10) (A50) :Explanation of parameter;
IFMT(10) (A5) :Input format of parameter;
TABLE CICSTAT :Control Card of STATPAC;
NO (I3) unique :Serial no. of control card;
CNAME (A6) :Card name;
.....
TABLE CICSPSS :Control Card of SPSS;
NO (I3) unique :Serial no. of control card;
CNAME (A15) :Card name;
TITLE (A70) :Title of card;
FUNC(10) (A70) :Function of card;
FORM(10) (A70) :Card form of parameter list;
PARM(20) (A5) :Parameter;
EXPL(20) (A70) :Explanation of parameter;
IFMT(20) (A5) :Input format of parameter;
END-DDL;
```

図2 マニュアルデータベースのデータ構造例

制御指令の形式を与えるアイテム FORM はシステム

が実際の制御指令を自動生成するために用いるので、機械処理しやすい（人間は読みにくくてよい）ようにコード化した情報を格納した。コード化の記法を各ソフトウェアやJCLに共通に適用できるようにしておけば、これを解読する処理系は一つで済む。このことを考慮して表1に示すコードを設定した。付録Aに、SPSS、STATPACおよびJCLのユーザマニュアル上の記述とこれらをデータベースCIに格納した例を示す。

表1 制御指令のコード化記号

分類	記号	意味
位置付け	!n	第n桁に位置付け（単一カードの場合）
	\n	第n桁に位置付け（複数カードの場合）
	~	空白を一つ挿入
パラメータ	◇a	パラメータa
	[ p ]	p省略可
括弧および繰り返し	( p1   p2 )	p1またはp2
	[ p1   p2 ]	p1またはp2（省略可）
	( p ) ; n	pのn回繰り返し
	[ p ] ; n	pのn回繰り返し（省略可）
	( p ) ; *	pの不定回繰り返し
	[ p ] ; *	pの不定回繰り返し（省略可）

#### 4.2 利用例の知識ベース化

ユーザが目的の仕事をコンピュータで処理するにあたって、入手しようとする情報はつぎのものである。

- 何を使用すればよいか？
- どのような方法をとればよいか？
- どのように遂行すればよいか？

これらの情報は、ユーザアプリケーションに対応して種類が多いばかりでなく、開発時には網羅しきれないものである。そのため、従来マニュアルには記載されることが少なく、ユーザ個人の所有する知識として蓄えられていた。ここでは、これらの情報を知識ベース化することを試みる。

前出の2つの統計パッケージについて、実現例を示す。格納するデータをつぎのように分類した。

- ① パッケージの処理手続きの照会情報AR
- ② パッケージおよび手続きの使用例題AE
- ③ 例題の手続き制御指令の列IC
- ④ 例題のジョブ制御指令の列IJ

これらを論理型言語による宣言述語として蓄積した。

AR ( k, n, p )  
 AE ( n, p, e, r c )  
 IC ( r c, r j, c )  
 IJ ( r j, j )

ここで、nはパッケージ名、pはパッケージ内の手続き名、kは検索用のキーワード（統計量など）、eは使用例題の説明、rcは使用例の手続き制御指令列の

識別名、rjは使用例のジョブ制御指令列の識別名、cは手続き制御指令列の例、jはジョブ制御指令列の例である。n、p、rc、rjは各述語を連結する変数として使用することができる。たとえば、kを入力して、AR、AE、IC、IJをリンクし、必要に応じてユーザと対話しながら情報を選択するように組み立てると、最終的にcとj、すなわち処理を実行するジョブデックを構成することができる。

cとjに与える制御指令列の記法として、表2のように記述対象に依存しない共通の規則を設けた。ここで用いた超記号は、<>は超変数、::=は超等号を示し、[]は括弧内省略可、{}は括弧内繰り返し可を意味する。記述の中の必須の空白は~で表した。

表2 制御指令列のコード化記法

```
<制御指令列> ::= <使用例識別名> : ( <制御指令> ; ) ...
<制御指令> ::= <制御指令名> [ ( <パラメータ> ) ] ...
<パラメータ> ::=
  <パラメータ名> [ ( <代替名> ) ] [= <値> ]
```

<制御指令名> はデータベースCIの中のアイテムCNAMEに登録されている名称、<パラメータ名>はアイテムPARMに格納されている名前とする。<代替名>はこの使用例の記述の中だけで有効なパラメータ名の代替名である。<値>にはパラメータのとり値を引用符で囲んだ文字列で与えるか、または定義済の代替名を書く。図3に記述例を示す。知識ベースへはShapeUpを用いて格納した。この格納データ例を付録Bに示す。cとjはワード列を含むリスト表現にした。

(1) 指定の変数について度数分布を求めヒストグラム付きで出力するSPSSの手続き制御指令 SPEX4 とジョブ制御指令 SPJ1

```
SPEX4: RUN^NAME a; GET^FILE f;
FREQUENCIES vl, op='8'; FINISH;;
```

```
SPJ1: SPSS f0='*userid/....pcc',
f4='*userid/...Afile';;
```

(2) SPSSの実データファイルを作成する手続き制御指令

```
SPEX0: FILE^NAME fn, fb;
VARIABLE^LIST vl;
INPUT^MEDIUM;
N^OF^CASES;
INPUT^FORMAT;
VAR^LABELS vn, vb;
VALUE^LABELS vl, v, b;
PRINT^FORMATS vl, vf;
FREQUENCIES vl='ALL';
READ^INPUT^DATA;
SAVE^FILE;
FINISH;;
```

(3) STATPACの実データファイルを作成する手続き制御指令

```
STEX0: D-CREF d, nv, no, v;
D-SAVE;
STOP;;
```

図3 制御指令列の記述例

### 4.3 ソフトウェアの管理目録

ソフトウェアをサービスする場合には、つぎのような情報を管理しユーザに知らせる必要がある。

- 何でPRされたか？
- サービス開始はいつからか？
- どこで開発されたものか？
- どのようなバージョンを提供しているか？
- どのマニュアルを引用すればよいか？

これらを、つぎの宣言述語として管理した。

AS(n, v, d, w, tp, tj, h)

ここで、nはパッケージ名、vはパッケージのバージョン、dはサービス開始日、wはPRの所在(ニュース番号など)、hは開発者または開発機関、tpはデータベースCIの中のパッケージの説明を格納しているテーブル名、tjはCIの中のJCLの説明を格納しているテーブル名である。付録CにShapeUpによる格納データ例を示す。

### 5. 利用情報の自動引用

4.2の冒頭であげたユーザの要求に対応する処理を設定し、前章で述べた利用知識を自動引用しながら各処理を自動的に連係して実行すれば、IEを実現することができる。処理を記述するために、処理片

in[σ] → [p | d] → out[τ]

を用いた。ここで、pは処理系、dはデータ、σ、τはそれぞれ入力、出力メッセージとする。

ここでは、2つの例について述べる。

#### (1) パッケージ、手続きの照会

この処理は、ユーザに探索用キーワードを入力させて、これに対応するパッケージ、手続きの概要を出力するものである。これを処理片で表す。

in[k] → [ShapeUp | AR(k,n,p)] → out[n,p]

in[n] → [ShapeUp | AS(n,v,d,w,tp,tj,h)]

→ out[tp]

in[n,p,tp] → [COOD | CI]

→ out[n,p,TITLE,FUNC]

ここでは、テーブル名tpによりマニュアルデータベースCIの検索を行い、該当するパッケージnと手続きpの名前、その表題TITLEと機能概要FUNCを出力している。

#### (2) 制御指令の自動生成

統計パッケージを実行するためのジョブデックを生成するには、つぎのステップで行う。

- ① 必要とする解析を行う使用例を検索する。
- ② ①の中から該当する例を選択し、その例題の制

御指令列(手続き制御およびジョブ制御)を求める。

③ 検索したパッケージについてマニュアルデータベースの中のテーブルを求める。

④ 例題で使用されている手続き制御指令の利用規約を参照して、手続き制御指令の列を組み立てる。

⑤ 例題で使用されているジョブ制御指令の利用規約を参照して、ジョブ制御指令の列を組み立てる。

これらのステップを処理片で表現する。

① in[k] → [ShapeUp | AR(k,n,p)] → out[n,p]

in[n,p] → [ShapeUp | AE(n,p,e,rc)]

→ out[e,rc]

② in[rc] → [ShapeUp | Ic(rc,rj,c)]

→ out[rj,c]

in[rj] → [ShapeUp | Ij(rj,j)] → out[j]

③ in[n] → [ShapeUp | AS(n,v,d,w,tp,tj,h)]

→ out[tp,tj]

④ in[tp,c] → [COOD | CI]

→ out[FORM,PARAM,EXPL,IFMT]

in[c,FORM,PARAM,EXPL,IFMT] → [x | -]

→ out[C]

⑤ in[tj,j] → [COOD | CI]

→ out[FORM,PARAM,EXPL,IFMT]

in[j,FORM,PARAM,EXPL,IFMT] → [x | -]

→ out[J]

①の出力eが複数得られる場合はユーザに選択してもらう。④と⑤の出力はそれぞれ手続き制御指令の列Cとジョブ制御指令の列Jであるが、処理片の構成はデータが異なるだけで同じである。これは、前章で述べたように利用情報の記法を制御指令の種類に依存しない形式で行ったためである。COODを用いている最初の処理片は対応するマニュアルを自動参照するもので、2番目の処理片はマニュアルの一般規則と例題の制御列を用いて自動生成するものである。

ここで処理系xについて説明する。xへの入力メッセージは使用例cまたはjとCIから入手した制御指令の形式FORMやパラメータPARAMなどである。図3のSPEX4は4種の制御指令を用いているが、これに対応するマニュアル情報を抽出した例を図4に示す。

cを用いて図5のAC表を以下の手順で作成する。

- ① 制御指令名をcname欄に記入する。
- ② パラメータ名をpname欄に記入する。
- ③ 代替名をつぎにより命名し、名前の前に処理済みマーク◇◇を付してaname欄に記入する。
  - cに定義されていれば、この名前とする。
  - cで未定義なら、AC表の行数を名前とする。

④ cにパラメータの値が指定してあれば *vmk* 欄に ++を記入し、値を *val* 欄に記入する。既定義の代替名が指定してあればその名前を *vmk* 欄に記入し、*val*欄は空のままとする。

⑤ EXPL と IFMT 欄を図4から得て埋める。

⑥ *vmk* 欄が空のパラメータについて、EXPL 欄の値をプロンプトとしてユーザに入力を要求する。入力された値を *val* 欄に書き込む。図5の下線部分が入力で決定した値である。

⑦ *vmk* 欄が代替名であれば、これと同名の *aname* 欄を捜し、その *val* 欄の値を転記する。

実際の制御指令の列は、つぎにより CSET の中に生成していく。図6に例を示す。

① FORM の各行を CSET 上に並べる。

② AC表の *cname* と *pname* を用いて FORM の中の不要綴りを除去し、パラメータを *aname* 欄の名前に書き換える。

③ パラメータをAC表の *val* 欄の値で置き換え、

FORM	:	RUN~NAME !16 ◇a
PARM	:	a
EXPL	:	run label
IFMT	:	A64
FORM	:	GET~FILE !16 ◇f
PARM	:	f
EXPL	:	file name
IFMT	:	A8
FORM	:	FREQUENCIES \16 GENERAL= ◇v1 [OPTIONS \16 ◇op] [STATISTICS \16 ◇st]
PARM	:	v1 op st
EXPL	:	variable list or ALL option list statistics parameter list
IFMT	:	A A A
FORM	:	FINISH
PARM	:	----
EXPL	:	----
IFMT	:	----

図4 例題 SPEX4 に対応するデータベースC Iからの検索データ

<i>cname</i>	<i>pname</i>	<i>aname</i>	<i>vmk</i>	<i>val</i>	<i>EXPL</i>	<i>IFMT</i>
RUN~NAME	a	◇◇1		Frequencies of MBINCM	run label	A64
GET~FILE	f	◇◇2		MB-file	file name	A8
FREQUENCIES	v1	◇◇3		AGE,SEX,ED	variable list or ALL	A
	op	◇◇4	++	8	option list	A

図5 例題 SPEX4 に対するAC表の例

位置付け指示に従って様式を整える。

以上の方法は、jについても全く同様に行う。

```

RUN~NAME !16 ◇a
GET~FILE !16 ◇f
FREQUENCIES \16 GENERAL= ◇v1
[OPTIONS \16 ◇op]
[STATISTICS \16 ◇st]
FINISH

```



```

RUN~NAME !16 ◇◇1
GET~FILE !16 ◇◇2
FREQUENCIES \16 GENERAL= ◇◇3
[ OPTIONS \16 ◇◇4 ]
FINISH

```



```

RUN NAME           Frequencies of MBINCM
GET FILE           MB-file
FREQUENCIES        GENERAL=AGE,SEX,ED
OPTIONS            8
FINISH

```

図6 使用例題 SPEX4 を用いた制御指令列の自動生成の遷移

## 6. おわりに

ソフトウェアのユーザインタフェースの改善をはかるための一方式について述べた。本研究は「大量の読みにくいマニュアルを従来のように印刷し読解する必要があるのだろうか?」という単純な疑問から発した計画であったが、データベースの情報管理技術、OSが提供する各処理機能、これから着実な発展が期待される知識処理の技術を連結することによって、当初の計画である「マニュアルレス」に一步近づくエキスパートシステムを実現することができた。

今後は、この方式の一般化を検討し、情報管理技術の中核としてどのような技術が統合されるべきかについて研究を進めたいと考えている。

### 【参考文献】

- [1] 松田孝子, 鈴木鷹, 田中知行: 知識型統合ソフトウェアVCAP—システム設計および異種統計パッケージ制御列の自動生成, 情報処理学会論文誌, Vol.27, No.8, pp.811-820 (1986).
- [2] 松田孝子, 鈴木鷹, 田中知行: ソフトウェア利用エキスパートシステム—異種ソフトウェアの統合利用, 計測自動制御学会第2回ヒューマンインタフェースシンポジウム論文集, pp.63-68 (1986).

付録A マニュアルデータベースC Iの格納データ例

M 1. SPSSのマニュアル記述例

1	16
VALUE LABELS	$\left\{ \begin{array}{l} \text{variable name} \\ \text{or} \\ \text{variable list} \end{array} \right\} (\text{value}_1)\text{label}_1 (\text{value}_2)\text{label}_2 \dots /$ $\left\{ \begin{array}{l} \text{variable name} \\ \text{or} \\ \text{variable list} \end{array} \right\} (\text{value}_1)\text{label}_1 (\text{value}_2)\text{label}_2 \dots / \dots$

1	16
CROSSTABS	$\text{TABLES} = \left\{ \begin{array}{l} \text{variable name} \\ \text{or} \\ \text{variable list} \end{array} \right\} \text{BY} \left\{ \begin{array}{l} \text{variable name} \\ \text{or} \\ \text{variable list} \end{array} \right\}$ $\text{BY} \dots / \left\{ \begin{array}{l} \text{variable name} \\ \text{or} \\ \text{variable list} \end{array} \right\} \text{BY} \left\{ \begin{array}{l} \text{variable name} \\ \text{or} \\ \text{variable list} \end{array} \right\}$

M 2. STATPACのマニュアル記述例

1	7	13	19	73
手続 き名	デ ータ 名	欠 測 値 の 指 定	コ メ ン ト	空 白

カラム	説 明
1~6	'D-CREF'
7~12	文字型 データ名(これは作成者が決める)
13~18	文字型 'BLANK△': データ値が空白のもの に対して欠測値とみなして特殊な値 10 <sup>75</sup> に変換する。 空白: データ値が空白のとき0に変換さ れる。
19~72	文字型 コメント

D 1. SPSSのマニュアル情報データベースの格納データ例

CNAME	FUNC	FORM	PARM	EXPL
VALUE LABELS	Associates a label of up to 24 characters with each value of any or all of the variables in a file. (Optional. May be prepared whenever the user wishes to declare or alter a label for any value in the file.)	VALUE~LABELS \16 {Ov1 {(Ov) O b};* //};*	v1 v b	variable list value label
CROSSTABS	Crosstabulates variables optionally controlling for other variables and provides various measures of association and significance. (Optional)	CROSSTABS \16 TABLES= {Ov11 {~BY~ Ov12};* //};* [OPTIONS \16 Oop] [STATISTICS \16 O st]	v11 v12 op st	variable list 1 variable list 2 option list statistics parameter list

D 2. STATPACのマニュアル情報データベースの格納データ例

CNAME	FUNC	FORM	PARM	EXPL
D-CREF	Reads a source data from disk file, and catalogs the data to the data file.	D-CREF O d Omis O c PARAMS O nv !16 O ns NAME [!7 O no O v];* VARIABLE (V)	d mis c nv ns no v	data name missing value, BLANK(10**75) or space(zero) comment number of variables number of samples variable number variable name

M 3. JCLのユーザ資料記述例

COL.1	8	16
\$	JOB	kadaibango\$password
\$	APROG	SPSS
\$	LIMITS	t,,,p
	RUN NAME	} (SPSS control card)
	INPUT MEDIUM	
	FINISH	
\$	PRMFL	08,R,S,kadaibango/bdata
\$	ENDJOB	

D 3. JCL情報のデータベースの格納データ例

CNAME	FUNC	FORM
SPSS	SPSS Job control card set on ACOS-6 in Computer Center, Tohoku University	\$ \8 JOB \$ \8 APROG \16 SPSS [ \$ \8 LIMITS \16 O t, O m,, O p] [ \$ \8 PRMFL \16 I*,R,S, O F0] [ \$ \8 PRMFL \16 O8,R,S, O F1] [ \$ \8 PRMFL \16 O9,W,S, O F2] [ \$ \8 PRMFL \16 FV,W,S, O F3] [ \$ \8 PRMFL \16 FR,R,S, O F4] \$ \8 ENDJOB

## 付録B 知識ベースKBの格納データ例

### (1) パッケージの処理手続きの照会情報AR

```
AR("descriptive statistics","SPSS","CONDESCRIPTIVE").
AR("mean, sum","SPSS","CONDESCRIPTIVE").
AR("minimum, maximum, range","SPSS","CONDESCRIPTIVE").
AR("standard deviation, variance","SPSS","CONDESCRIPTIVE").
AR("one-way frequency distribution","SPSS","FREQUENCIES").
AR("mean","SPSS","FREQUENCIES").
AR("minimum, maximum, range","SPSS","FREQUENCIES").
AR("two-way to n-way cross-tabulation table","SPSS","CROSSTABS").

AR("quantification I","STATPAC","COMP1").
```

### (2) パッケージおよび手続きの使用例題AE

```
AE("SPSS","V-A","Data transfer from V-file to A-file","SPEX0").
AE("STATPAC","V-A","Data transfer from V-file to A-file","STEX0").

AE("SPSS","CONDESCRIPTIVE",
   "all conderscriptive statistics of all variables","SPEX1").
AE("SPSS","FREQUENCIES",
   "frequencies of indicated variables with histgrams","SPEX4").
AE("SPSS","CROSSTABS",
   "n-way cross tabulation with indicated variables","SPEX5").

AE("STATPAC","COMP1",
   "quantification I without continuous variable","STEX1").
```

### (3) 例題の手続き制御指令の列IC

```
IC("SPEX0","SPJ0",
   [ <<"FILE~NAME" "fn" "fb">>, <<"VARIABLE~LIST" "v1">>,
     <<"INPUT~MEDIUM">>, <<"N~OF~CASES">>, <<"INPUT~FORMAT">>,
     <<"VAR~LABELS" "vn" "vb">>, <<"VALUE~LABELS" "v1" "v" "b">>,
     <<"PRINT~FORMATS" "v1" "vf">>, <<"FREQUENCIES" "v1" "ALL">>,
     <<"READ~INPUT~DATA">>,
     <<"SAVE~FILE">>, <<"FINISH">>]).
IC("SPEX4","SPJ1",
   [ <<"RUN~NAME" "a">>, <<"GET~FILE" "f">>,
     <<"FREQUENCIES" "v1" "op='8'">>, <<"FINISH">>]).
IC("SPEX5","SPJ1",
   [ <<"RUN~NAME" "a">>, <<"GET~FILE" "f">>,
     <<"CROSSTABS" "v11" "v12">>, <<"FINISH">>]).

IC("STEX0","STJ0",
   [ <<"D~CREE" "d" "nv" "no" "v">>, <<"D~SAVE">>, <<"STOP">>]).
IC("STEX1","STJ1",
   [ <<"PHTL" "t">>, <<"INITF">>, <<"D~LOAD" "d(dname)">>,
     <<"COMP1" "d=dname" "n" "n1=1" "v1">>, <<"STOP">>]).
```

### (4) 例題のジョブ制御指令の列IJ

```
IJ("SPJ0",*UID,
   [ <<"SPSS" "f0='\'^^*UID^"/.....PCC\'" "f1='\'^^*UID^"/....work\'"
     "f3='\'^^*UID^"/...Afile\'">>]).
IJ("SPJ1",*UID,
   [ <<"SPSS" "f0='\'^^*UID^"/.....PCC\'" "f4='\'^^*UID^"/...Afile\'">>]).
IJ("STJ0",*UID,
   [ <<"STATPAC" "f0='\'^^*UID^"/.....PCC\'" "f1='\'^^*UID^"/....work\'"
     "f2='\'^^*UID^"/...Afile\'">>]).
IJ("STJ1",*UID,
   [ <<"STATPAC" "f0='\'^^*UID^"/.....PCC\'" "f3='\'^^*UID^"/...Afile\'">>]).
```

## 付録C ソフトウェアの管理目録の格納データ例

```
AS("SPSS","R9.1",831017,"Center News No.379","CICSPSS","CIJ","SPSS Inc. and NEC").
AS("STATPAC","R4.7",840601,"Center News No.394","CICSTAT","CIJ","NEC").
```