

操作例の呈示による帳票処理  
コマンドマクロの生成

4H-4

池上良己 深津貞雄 板敷晃弘 菊田ひとみ  
(株)富士通研究所

1. はじめに

従来のエンドユーザ向きの帳票処理簡易言語においては、処理を登録するのに手続きのなプログラムを組む必要があるものが多い。しかし、エンドユーザはループや条件分岐などの制御構造を考えるのが不得意であり、プログラミングするのは容易ではない。

しかしそのようなユーザでも手作業による帳票の処理は問題なく行える。またプログラムは組めないがワープロの操作ができる人は多い。これはプログラムのような抽象的な手段で考えることは難しいが、帳票やディスプレイに書かれた具体的なデータを見ながら順次処理することはそれに比べるとかなり易しいからだと考えられる。

本稿では、ディスプレイに実際の帳票イメージを具体的なデータとともに表示し、ユーザが手作業の感覚でデータの複写や四則演算などを順次行っていくと、入力コマンド列を一般化した形でコマンドマクロを生成するシステムについて述べる。

2. システムの概要

2.1 システム構成

図1に本システムの概略構成図を示す。

手作業の感覚でユーザが帳票処理を指示できるように次の構成になっている。すなわち、ユーザの指示により、システムは処理を行うのに必要な複数の帳票を実際のデータとともにディスプレイに表示する。ユーザはディスプレイに表示されている項目をマウスで指定して、項目の複写や四則演算を指示する。必要ならば表示する帳票の頁を切り換えられる。コマンド実行部はユーザが与えたコマンドをその都度実行する。処理の結果、変更になった項目の内容はデータ表示部がその都度ディスプレイに表示する。

ユーザが入力したコマンドを一般化するために、システム内部に処理モデルを複数持っている。処理モデル選択部はユーザが指定したコマンドをもとに、該当する処理モデルを選択する。該当するモデルが複数ある場合、ユーザがコマンドを新たに入力する度にモデルを限定していく。もし、選択したモデルが処理条件を含んでいるならば、条件質問部はユーザに条件を質問する。

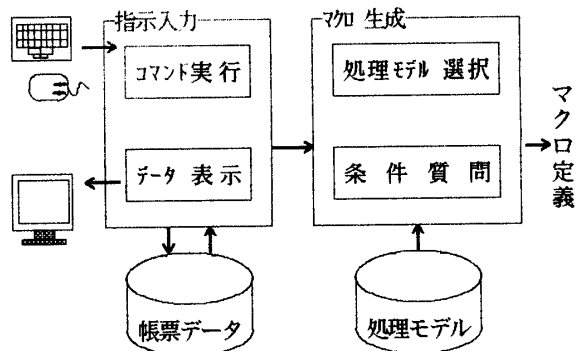


図1 システム構成

2.2 システムで扱える帳票の形式

ファイルは1または複数の頁から構成されている。ひとつの頁には繰り返し項目から構成される表と、非繰り返し項目(これを単項目と呼ぶ)から構成されている。

3. 帳票処理マクロ

3.1 コマンドマクロの形式

マクロは一般化コマンド列により構成される。一般化コマンドとはユーザが指定したコマンドのパラメータのうち、頁番号と表の行番号を変数化したものである。コマンドとしては現在、複写と四則演算がある。同一帳票内、または異なる帳票間の項目同士の処理を行える。一般化コマンドは以下の形式をしている。

(コマンド名 入力項目id・入力項目id 出力項目id)

単項目の項目idは (ファイル名 頁変数 項目名)

表の項目のidは (ファイル名 頁変数 行変数 列番号) で表現する。

3.2 頁変数と行変数の属性

頁変数はファイル毎に、行変数は表ごとに持っている。それぞれの変数ごとに例えば、以下の属性を持っている。

- ①処理を頁(行)単位で繰り返すかどうかの区別
- ②繰り返し開始頁(行)
- ③繰り返し終了頁(行)
- ④繰り返し間隔
- ⑤頁(行)選択条件

### 3.3 処理モデル

頁(行)変数属性の組み合わせを予めシステムが用意している。これを処理モデルと呼ぶ。これ以外の組み合わせは指定できない。モデル番号は可能性の高い順に1から番号が付けられている。処理モデルの例を図2に示す。

モデル番号 属性	1	2	3	4
①繰返有無	有	無	有	無
②開始	先	先	*先	*先
③終了	終		*終	
④間隔	1		*1	
⑤条件	無	無	有	有

「先」は1頁(行)のことである。  
 「\*先」は条件を満たす頁(行)のうち最初の頁(行)。  
 「\*終」は条件を満たす頁(行)のうち最後の頁(行)。  
 「\*1」は条件を満たす次の頁(行)に進むという意味。

図2 処理モデルの例

## 4. マクロ登録の操作例

### 4.1 例題

手紙を出したいので、宛先一覧表に記入されている人の名前と宛先を宛名ラベルに書きたい。宛先は住所録を参照して記入する。図3に各帳票の例を示す。

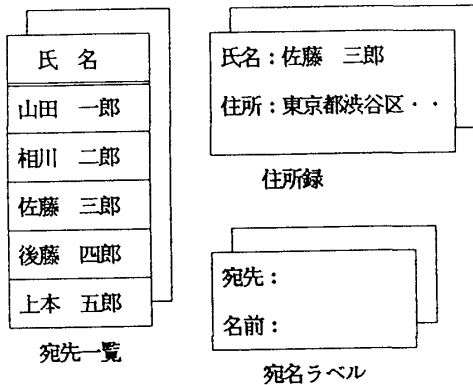


図3 帳票の例

### 4.2 操作例と内部処理

ユーザの操作とシステム内部の処理例を説明する。

- (エ) はユーザの操作、(シ) はシステムの処理である。
- (エ) 一覧表の1頁1行目の氏名を宛名ラベルの1頁の名前欄に複写する
- (シ) 以下の一般化コマンドを生成し、該当する処理モデルを頁/行変数ごとに列挙する。  
 (複写 (一覧表 P1 L1 1)(宛名ラベル P2 名前))  
 該当モデル (P1): (1 2 3 4)  
 該当モデル (L1): (1 2 3 4)  
 該当モデル (P2): (1 2)  
 宛名ラベルは新規に作成するファイルなので、P2は3,4があてはまらない。

- (ユ) 住所録の3頁にある「山田」さんの住所を宛名ラベルに複写する。
- (シ) 以下の一般化コマンドを生成する。  
 (複写 (住所録 P3 住所)(宛名ラベル P2 宛先))  
 該当モデル(P3): (3 4)  
 P3は開始頁が1でないのでモデル1,2は該当しない。
- (シ) 「なぜ3頁を選んだのですか?」と質問する。
- (エ) (一覧表 1頁 1行 1列)  
 = (住所録 3頁 氏名) と答える。
- (シ) P3の選択条件を設定する。  
 条件 (P3): (= (一覧表 P1 L1 1)  
 (住所録 P3 氏名))
- (エ) 一覧表の1頁2行の氏名を宛名ラベルの2頁の名前欄に複写する。
- (シ) L1は次の行に進んだのでモデル2と4は該当しない。P2は次の頁にすすんだのでモデル2は該当しない。  
 該当モデル (L1): (1 3)  
 該当モデル (P2): (1)
- (エ) 住所録の1頁目が「佐藤」さんの頁なので、1頁の住所を宛名ラベルに複写する
- (シ) P3を以下のように限定する。  
 該当モデル (P3): (4)

### 4.3 マクロ定義文の例

以上の結果、次のマクロ文が生成される。

```

(複写 (一覧表 P1 L1 1)(宛名ラベル P2 名前) )
(複写 (住所録 P3 住所)(宛名ラベル P2 宛先) )

処理モデル(P1):(1 2 3 4)
処理モデル(L1):(1 3)
処理モデル(P2):(1 3)
処理モデル(P3):(4)
条件 (P3): (= (一覧表 P1 L1 1)
              (住所録 P3 氏名) )
    
```

図4 生成されたマクロ定義の例

P1, P2, L1 の処理モデルが複数個存在するが、優先順位の高いものすなわち1を取敢えず採用する。次にこのマクロをユーザが実行して誤りを発見した場合、ユーザからの指示により順次あてはまらないものを消去していく。

## 5. おわりに

操作例からコマンドマクロを自動的に生成するシステムについて述べた。システムは現在FMR 60上で稼働中である。

実用化するためにはさらに多くのコマンドを用意する必要がある。この場合、処理モデルは変更する必要はなく、コマンドを増やすだけで対処できる。