数理解析研究所講究録
第433巻 1981年 120-129

MIRRFシステムによる文献データの作成と利用について

有川節夫（九大理）

1. はじめに

MIRRFシステム(Multistage Information Retrieval System Based upon Researcher Files)は1974年～75年に、次の目的で開発した研究者用の文献情報検索システムである.[1]

1) 研究者の身辺の文献データを蓄積し組織化すること.
2) データの加工を自動化すること.
3) 研究者の独自性を検索に積極的に反映させること.
4) 網羅性のあるデータ・ベースを利用すること.
5) 高次の知的検索を行えること.

こうした目的に従って、研究者ファイルの概念を導入し、文献を2種類の研究者ファイルの形に貯え、それを検索時に積極的に利用する方式にシステム全体をまとめた. ここにいう2種類の研究者ファイルとは、論文末尾にある参考文献の形の文献リストと、主として熟読された論文に対して研究者が自分の考えに従って付与する概念としての木構造をもつキーワードのファイルである. 第2種の研究者ファイルを連関(association)を用いて多段に利用するわけである. データ加工の自動化に関しては、文献のタイトルからキーワードを自動抽出する索引システムREKWEST, 文献コード化, 木構造付きキーワード・ファイルを部分シソーラスとして把えそれを編集するシステム等を開発してきた.

最初のMIRRFシステムはミニコン(FU-200/U-400)で実現して、実験的に使用し、またデータも約20,000件の文献データを作成してきた. このシステムでは上記の目的のうち1)～3)は或る程度満されるものであった. 4)に関しては、最近大型センターのFAIRSによりINSPECのデータを検索出来るようになった. 5)に関しては、現在基礎的な研究を進めている最中であるが、近似的なシステムとして逐字サーチシステムTEXTIRが能力とスピードの両面から有用であるこ

とを見てきた.

一方、1978年までの特定研究「学術情報」M委員会において、学術情報用のデータ・ベース・システムをData Box, PRF (Private Researcher File), UDL (User Data Library), CDL (Center Data Library) という各段階で把える考え方が提案され、その委員を通じて次第に定着しつつある. 更に大型計算機センターの利用形態もTSS端末からの利用が一般的になってきた.

こうした経験や環境・状況の変化を正視して、研究者用の情報システムのあり方を再考してみると、我々が当初掲げた目標は現在でも正当であるように思われる. しかし、それだけでは十分でないことも明白である. このような情報システムの扱うべきデータは文献データに限定されたものではなく、論文作成や書類の整理、数値データの処理まで含めた日常のほとんどの研究活動を支援してくれる作業場としてのシステムであることも要求される. 勿論外部のデータも自由に取り込めて、他の研究者と情報システムを通じて交信でき、更に、こうした情報システムを介した活動を基盤にして、4)でいうところの網羅性のあるデータ・ベースが構築できるものでなければならない.

このような観点に立って、現在九大大型計算機センターのM200に新らたにMIRRFシステムを実現しつつある. 最近、いわゆる研究者の作業場としてのシステムとしては一応の完成をみたので、ここにその概要紹介する. 全体のシステムは完成後センターを通じて公開する予定である. 公開に伴う問題点とその解決法についても併せて述べてみる. なお、こうした研究者の作業場としての情報システムには便利で手ざわりのいい汎用のエディタが不可欠である. 現在のMIRRFシステムにはTEDITシステムがある. これについては篠原[4]を参照されたい.

2. ファイルの構造

記憶域の有効利用とソフトウェアの作成効率を考慮して、2方向のポインタをもった逐次ファイルと各種ファイル名やキーワードの管理のためのB-treeをファイルの物理構造の基本としておいた.

2方向ポインタで連結された逐次ファイルは、主にゴールファイル用に使われるものである. 現在のシステムでは、512文字(バイト)からなるブロック(ページ)を8のセクタに分割して図1のように構成した. このような何個かのセクタからなるファイルの消去は、このファイルをポインタもそのままにして、消去されたファイルの末尾に追加することによって行う. 従って、ファイルの消去は瞬時に行われる. ファイルの移動についても同様である. またファイルの作成に際しては、1) 使いかけのページにあるセクタ, 2) 一度消去されたセクタ, 3) 新しいセクタの順序で必要なセクタが発行されるので、記

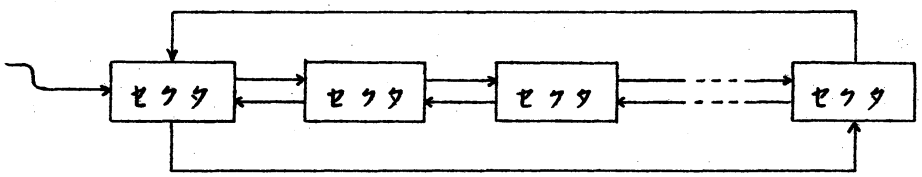

図1. 逐次ファイルの構造

憶域は常に効率よく使用されることになる.

B-treeのレコード構造は図2のようなものである. 図において KLP の部分に実際のキーの長さが記入される. KLP の位置は各B-treeごとに指定できる.

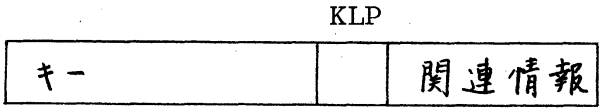

図2. B-treeにおけるレコード

一定の長さの場合には KLP < 0 であり |KLP| でその長さを指定する. このようなレコード構造を採用し, B-treeにおけるキーの検索、挿入、削除のルーチンでは関連情報の部分を全く扱わないようにしたことによって、B-treeをファイル名やキーの管理だけでなく、語の頻度調査などデータの整列化を必要とする作業にもそのまま適用することができる.

3. MIRRF システムにおけるファイルの構成

現在の MIRRF システムのファイル構成は図3に示す通りである. TEDIT ファイルと勿論外部データ・セットを除いて全て512バイトのレコード長のダイレクト・アクセス・ファイル上に構成されている.

MIRRF ファイルは図1, 2で示した逐次ファイルと B-tree からなる. B-tree はファイル名を管理する. 論理的には最大10段までの階層構造をもつものである. ここでのファイルはpass numberによって保護されるが、不用意な消去から護るため及びファイル実体の共有のためにrenameの機能がある. これはファイルの実体に新しい名前を追加するだけであって、一方の名前のもとに変更が生じたときにはじめて実体のコピーが作られる.

作業用ファイルは利用者がMIRRF システムを使って作業をするとき扱われるもので、図1の構造をもつものである. このシステムでは端末機器等も全てファイルと見做される. そうしたファイルや他の各種ファイルへの/からのデータの転送は全て save/load で行われる. 最後に load又は pull up されたファ

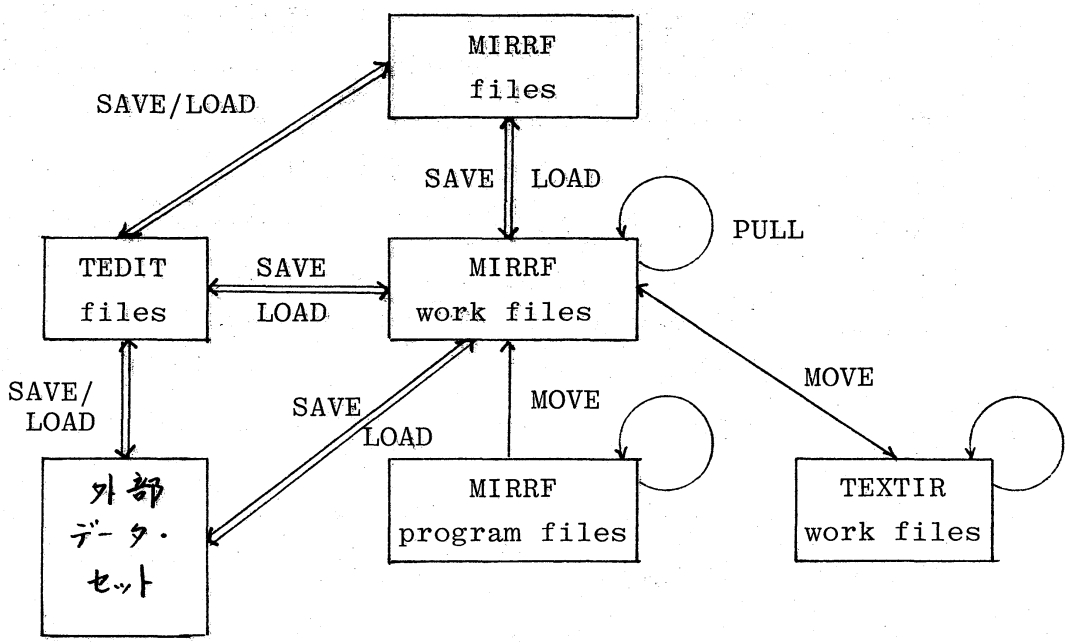

図3. MIRRF システムにおけるファイル

イルがトップにあるようになっている. 従って古いファイルは新しいファイルがもってこられる度に push down されることになる. 作業用ファイルの領域が物理的にいっぱいになったときには、作業中に最近の10個のファイルを残して古いものから順次消去されて作業域が確保される. 古いファイルをトップにもってくる(pull)こともできるので、作業中のファイルを不注意による消滅から守ることができる.

図1に示すように、作業域はTEDITファイル, MIRRFプログラム・ファイル, TEXTIR作業用ファイルだけでなく外部データ・セットと連絡できる.

MIRRFプログラムファイルはMIRRFシステムにおけるキーボードからの全てのコマンド, データを記録するためのものである. 構造は作業用ファイルと同じである.

TEXTIR作業用ファイルはMIRRFシステムのサブシステムTEXTIRの検索結果をファイル機番, レコード番号, セクタ番号, 開始文字位置, 長さ, ヒットした質問番号の組みで記録するためのものである. 構造はプログラム・ファイルと全く同じである.

〈注意〉 図3は、MIRRF作業用ファイル又はTEDITファイルから見たものである. ⟹はファイル実体のコピーを示し, ⟶はポインタのみの移動を示す.

4. MIRRF システムの機能

MIRRF システムの主要な機能について例を挙げて説明する．MIRRF システムには MIRRF 状態と DO 状態とがある．MIRRF 状態はこのシステムの初期状態であり、この状態でコマンドが投入されるとプログラムファイルが新規に作成される．DO 状態ではコマンドやデータの記録は現在使用中のプログラムファイルにとられる．両者はその他の点では全く同一である．状態の遷移は図4に示すようにコマンドによって行われる．図において READY は TSS における READY 状態であり、COMMAND は以下に説明するような END 以外の MIRRF システムのコマンドである．MIRRF 状態で開かれたプログラムファイルは END コマンドで再び MIRRF 状態に戻るときに閉じられて、プログラムファイル領域のトップに置かれる．

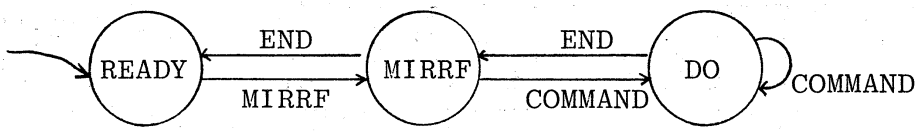

図4. MIRRF システムにおける状態遷移

END　MIRRF システムから離れるためのコマンドである．DO 状態にあるときは、上図に見られるように END を2回投入しなければならない．

KEYIN　キーボードから入力されるデータを作業域のトップに作業用ファイルとして作成する．プロンプトとして ">" が出される．入力の終了は (CR) ;;; で指示する．

LOAD　MIRRF ファイルや TEDIT ファイル，外部データ・セットを MIRRF のトップの作業用ファイルに持ってくるためのコマンドである．FILE:= に続いて必要なファイル名を指定する．指定されたファイルがパス・ナンバーをもっているときには、PUSS NUMBER:= に従ってそれを与えなければならない．なお、キーボードは KB という名前のファイルと見做される．従って、FILE:=KB とすると、KEYIN コマンドと同じ働きをすることになる．

SAVE　作業域のトップにある作業用ファイルの内容を MIRRF ファイルや TEDIT ファイル，外部ファイルへセイブするためのコマンドである．ファイル名やデータ・セットの指定は LOAD の場合と同様である．端末の画面は VD という名前のファイルと見做される．

LOOK　トップにある作業用ファイルの初めの20行の表示を行う．(CR) ごとに次々に20行ずつの表示がくり返される．終了は E (CR) で指定する．

LIST　MIRRF システムがもっているファイル名やファイルの内容をリストするためのコマンドである．ファイル名は $w_1 \cdot w_2 \cdot \cdots \cdot w_n$ の形で階層的に定義できる．完全なファイル名が指定されたときには、そのファイルの内容がリストされる．不完全な指定の場合には、その指定された名前を親とするようなファ

イル名がリストされる．従って，ファイル名を空語（CRだけ）で指定するとMIRRFシステムが現在もっているトップ・レベルのファイル名がリストされる．

RENAME MIRRFファイルの実体に新しい名前を追加するためのコマンドである．ファイル名やパス・ナンバーの指定は上記のコマンドと同様である．以下ファイル名等の指定についての説明は省略する．

APPEND 指定されたMIRRFファイルにトップにある作業用ファイルを追加するためのコマンドである．

PULL 作業用ファイルやプログラム・ファイル，TEXTIRの作業用ファイルの指定されたものをトップにもってくるためのコマンドである．

```
PULL
FILE:=WF
FILE POSITION:=3
```

のように使用する．ファイル名は WF（作業用ファイル）の他にPF（プログラム・ファイル），TF（TEXTIR作業用ファイル）を指定できる．上の例では，MIRRFの作業用ファイルの4番目のものがトップにもってこられる．
FILE POSITION:=0 とすると，トップにあるファイルが末尾におかれる．作業域がいっぱいになったときに先ず最初に消去されることになる．

WFMV 作業用ファイルのトップにあるものを，TEXTIRの作業用ファイルのトップに移動させる．

PRMV プログラム領域のトップにあるファイルを作業域のトップに移動させる．

TXMV TEXTIRの作業域のトップのファイルをMIRRFの作業域のトップに移動させる．

LENGTH 指定されたファイルの長さを表示する．ファイル名WFでトップにある作業用ファイルの長さが分る．

REST MIRRFのファイル領域や作業用ファイルの使用可能なセクタ数を表示する．

REPLACE トップにある作業用ファイルを対象にして，指定された記号列を他の指定された記号列ですべて置き換えたファイルを作り，トップに置く．以下のように使用する：

```
REPLACE
NEW LINE:=!
A1:=x1
B1:=y1
A2:=x2
B2:=y2
  ⋮
  ⋮
```

⋮

$A_n := x_n$

$B_n := y_n$

$A_{n+1} :=$ (CR)

この場合、トップにある作業用ファイルに含まれているすべての x_i が y_i で置き換えられたファイルが作られる。なお $\{x_1, \ldots, x_n\}$ をキーワードをみて、パタン・マッチング・マシンを構成し、それがファイル上を左から右へ走ることになる。

TEXTIR 一方向の逐字サーチシステムである。詳細については、[3]を参照していただくことにして、ここでは使用例だけを挙げておく。下の例において、:= 以下の部分は利用者による入力である。区切語 (delimiters) は文字列であり、キーワード (keyword) は文字列又は x・・・y・・・z のようなトリプルドット・・・ を含む文字列である。論理式 (logical formula) はキーワード変数

```
DO:TEXTIR
<<TEXTIR/E>>

0...RESET, 1...NORESET:=
NEW LINE:=!
0...LIST, 1...NOLIST OF TABLE:=0

 OUTPUT DELIMITERS
D1:=#
D2:=

 INPUT DELIMITERS
D1:=

 ADDITIONAL INPUT DELIMITERS
D1:=

 KEYWORDS
A1:=TURING
A2:=COMPLEXITIES
A3:=ONE-WAY
A4:=

 LOGICAL FORMULAE
Z1:=A1•A2
Z2:=A3
Z3:=
FILE:=MIYANO.PAPERS

 RETRIEVED TEXTS
TOTAL              =      6
QUESTION  1 (Z 1) =       0
QUESTION  2 (Z 2) =       6

 FILE:=
0...LIST, 1...NOLIST OF RESULTS:=0
```

(A1, A2, ...), 式変数(Z1, Z2, ...), 整数, ε-変数(E), かっこ [,], (or), ・(and), ^(not), +, -, *, /, <, >, =, <>(≠)を用いて作られるものである. TEXTIR による検索結果は、先に述べたように、ヒットした質問番号をコード化した整数, ファイルの機番, レコードの番号, セクタの番号, 開始文字位置, 長さの組の形で、TEXTIR作業用域のトップにあるファイルに記録される.

TXEP これはTEXTIRによる検索結果を編集して新しいファイルを作るためのコマンドで、下の例のように使う. QUESTION:= には TEXTIR での質問の番号

```
DO:TXED
 QUESTION:=2
(QUESTION  2=      6 TEXTS)
 NEW LINE:=!
 OUTPUT DELIMITER:=NO.
 NUMBERING (Y OR N):=Y
 SORT BY:=AY
```

を指定する. 次の( )内はその質問に対してヒットしたテキストの個数である. NEW LINE:=は次にある出力区切り語 (output delimiter) の中で return キーの代りに使う文字である. 出力区切り語は任意の文字列である. 番号付けを行いたいときは、NUMBERING (Y OR N):= で Y を選ぶ. 最後の SORT BY:= は、TEXTIRで使う標準的な文献テキストに対して有効なもので、C (文献コード), A (著者), Y (年代), J (誌名) の中から0個以上選び指定する. 上の例では、著者順にソートして、同一著者に関して (論文の) 発行年順にソートされることになる. こうした標準的な文献以外のテキストに対しては何も指定しなければよい. なお、ソーティングにはファイル名等の管理に使った B-tree を使っている.

? Helpである. MIRRF システムが入力要求をした任意の時点で使用できる. このシステムではhelpに対する案内文は1つの MIRRF ファイルとして扱われる. 従って、利用者は必要に応じて内容の追加・変更を行うことができる.

TEDIT MIRRF のテキスト・エディタにはいる. ([4]を参照)

以上が現在の MIRRF システムの基本的なコマンドである. 他に MIRRF ファイルや作業用ファイルを初期化するコマンド等があるが、ここでは省略する.

5. 交信記録とその利用

先に述べたように、MIRRF システムでは、MIRRF状態から次の MIRRF 状態までのキーボードからの入力はコマンドやデータも全てプログラムファイルに記録される. これは通常のファイルと全く同様に扱われるだけでなく、一種のコマンド・プロシジャとして利用できる. MIRRF ファイルにセイブされたこう

した記録は、入力要求が行われた任意の時点で、

/FILE NAME

//FILE NAME

の形で呼び出して使用できる．前者に対しては、全ての作業がキーボードから直接指示した形で実行される．即ち、コマンドやデータも全て表示され、実行される．後者に対しては、こうした表示は行われない．

ファイル名としてキーボード(KB)を指定することもでき，ファイルの内容に再び上記の形の指示を含むことができるので、利用者は簡単に自分用のシステムをMIRRF の中に持つことができよう．

このような交信の記録は当然エディティングの対象になる．したがって、それをプログラムとして使う場合には、一種の構文のチェックが必要である．このシステムでは、こうした記録をプロンプト付きでとっているので、構文のチェックは極めて簡単に（FORTRANの1ステートメント）実行中に行われる．

6. おわりに

エディタを除いて、現在のMIRRFシステムの機能について簡単に説明した．今後、いわゆるPRF マネージメント・システムとして整備し、九大センターを通じて一般の利用者に公開していくつもりである．

安全性の問題　作業用ファイルやTEDITファイルに関しては、ファイルの安全性について或る程度注意を払った．しかし、MIRRFファイルに関しては、直接アクセス・ファイルを使っているために、計算機のcrash等に対して現在は無防備である．この問題もデータの共有の問題と併せて解決したので、本稿で述べた範囲内でのMIRRFシステムをSIGMA（Σ）という名前で近く九大大型計算機センターから公開することにしている。

PRF ⟶ UDL ⟶ CDL の構図　公開されるSIGMAシステムは、少くとも文献データに関しては、PRFからUDLまではカバーできるものと考えている．CDLとしては、現在FAIRSによってINSPECがアクセスできるので、そちらの充実を期待したい．またMIRRFシステムもCDLのデータの蓄積に寄与できるものにしていきたい．

参考文献

1. Arikawa, S. and Kitagawa, T.: Multistage Information Retrieval System Based upon Researcher Files, Res. Rept. Res. Inst. Fund. Inform. Sci., Kyushu Univ., No. 51 (1975), 1-34.

2. Arikawa, S., Kanō, S., Kitagawa, T. and Takeya, S.: Organization and Use of Private Researcher Files in Scientific Research Works, Research on Scientific Information System in Japan (1980).

3. Arikawa, S.: A One-way Sequential Search System and Its Applicability to Medical Information Processing, Proc. HISCC-13 (1980).

4. 篠原武: MIRRF システムにおけるエディタについて,「知識の表現とそれを利用する情報検索システムの研究」報告書 (1981), 247-258.