U.S. DEPARTMENT OF THE INTERIOR

GEOLOGICAL SURVEY

An Evaluation of the M204 Data-Base Management System

by

George T. Mason, Jr.

Open-File Report 81-1148

1981

CONTENTS

ILLUSTRATIONS

INTRODUCTION

In October of 1980, the Office of Information Resources Management at the Department of the Interior requested that a feasibility study be conducted on the Mineral Data System (MDS) Computerized Resource Information Bank (CRIB) and the effectiveness of the University of Oklahoma information storage and retrieval system (GIPSY--General Information Processing System). The results of the study will be used to determine whether a storage and retrieval system (SRS) or a full data-base management system (DBMS) would be the most efficient type of system to manage the geologic information in CRIB.

The geologic information in CRIB is very complex and has very few control standards. Therefore, the U.S. Geological Survey, which operates the file, decided that each system included in the feasibility study should have an operational data base constructed and tested using actual data from the CRIB file. The selection and testing of these files would provide a true operational picture of the benefits and restrictions associated with each DBMS or SRS.

This report describes the results of the study of M204, which is a DBMS supported by IBM systems 360/370, 303X, and 4300 and Amdahl and ITEL/plug compatible computers running under OS/VS operating systems. M204 has a relationship-type data structure, utilizes the inverted file access method, and supports data independence, flexibility, record security, and field security. Data are read and stored by pages. The page sizes vary depending on physical storage devices at the installation. M204 features hierarchical, network, and relational data-base organization and supports host languages, Cobol, Fortran, PL1, and a user language, as well as telecommunications interface and batch processing. It has an automatic rollback and audit-trail system-accounting facilities and a check point/restart feature.

Objectives

A test file of 623 records containing a representative selection of the records on the Master CRIB file was converted to a M204 file. The objectives of this conversion were: to design and create an operational file to locate and solve problems that would be encountered when doing a conversion; to provide information on the operational benefits and drawbacks associated with a conversion; to obtain the actual cost and storage requirement produced by the conversion; and to devise a procedure for converting to M204 files efficiently.

Information Sources

Information and background relating to M204 were obtained from the vendor demonstrations, classes, publications (Computer Corporation of America, 1979a-d), and representatives, other users, and operational performances. The actual data for creating the M204 file came from the existing CRIB file.

Operational Tasks

The operational tasks performed were designing the file, selecting record/field, assigning field-name descriptions, calculating space, creating the file, adding records and fields, deleting fields or records, and writing

special software. Each task is explained in detail, and examples of the tasks are shown.

File Design

The design of the file was dictated by the current Master file structure. The fields and the records had to maintain their one-to-one relationship, and only one record type could be used. The field name had to remain the same and the file size could not exceed 600 tracks. The common retrievals had to be efficient and inexpensive.

Record/Field Selection

The records selected for the test file had to be examined, and each field had to be checked for data. If a field was found to contain no data within the group of records, that field was deleted from that record group. The group of records that was selected contained data in at least one field of one record in that group.

Field Names and Descriptions

According to the specification, the field names had to remain as they were on the Master CRIB file. Therefore, no field-name change was necessary. However, each field had to be given a description either by assigning it or allowing the system to assume default descriptors for the fields.

The file design specifications ask for an inexpensive and efficient access method for common retrievals; because of this, the selected fields often had to be keyed. If a data element in a field occurred in only one record, it was considered to be unique; because many textual searches are done on the current Master file at word level, it was necessary to make each record keyed and invisible. Further information regarding the descriptor can be found in the M204 File Manager's Technical Reference Manual (Computer Corporation of America, 1979a).

The field names were taken from the MDS operational file, but the descriptions had to be based on design requirements. Therefore, the default descriptor was used for all fields except the ones that influenced the design requirement. The following is a list of the fields/names and their non-default descriptors:

Name and Descriptor	Name and Descriptor
A10	C30
A11	C31
A12	C32
A13	C40
A20 (KEY)	C42
A21 (KEY)	C42E
A22 (KEY)	C42A
A30	C42B

Name and Descriptor
A40 (KEY)
A50 (KEY)
A60
A64 (KEY)
B40
A70
A80
A107
A82
A83
C10
C11
C12
MAJOR
MINOR
COPROD
BYPROD
POTEN
OCCUR
C20
P1D
E1B
G18
G18A
G18B
G18C
G18D
E1C
E1D
H1
H1A
H1B
H1C
H1D
H7
H8
J1
J1A
J1B
J1C
J1D
J8
K1
K1A
K2
K2A
K3
K4
K5
N5
N15
N30

Name and Descriptor
C42C
C42F
C43
C42G
C50
L10
L20
L30
L40
L100
L110
M10
M15
M20
M21
M30
M31
M40
M41
M50
N75
N80
N85
GEN
F1
F2
F3
F4
G2
G1
G3
G4
B10 (KEY, UE)
B20
U
USGS
C44
BTU
SUL
ASH
CARB
C45
C46
C47
KEYWORD (KEY, INV)

Name and Descriptor	Name and Descriptor
N40	D3
N45	D11
N50	D11A
N55	D11B
N60	D11C
N70	D11D
M51	D3A
M60	D3B
M61	D3C
M70	D3D
M80	D4
M90	D4A
M100	D4B
M110	D4C
M120 (KEY)	D4D
M130 (KEY)	D5
M140 (KEY)	D5A
M160	D5B
M161	D5C
M170	D5D
M171	D6
M190	D6A
M191	D6B
M200	D6C
M201	D6D
M210	D7D
M211	G7D
M220	G8D
PROD	G9D
YES	G11D
NO	G13D
LGE	G14D
MED	D7
SML	G15D
D1	P2D
D1A	P3D
D1B	P4D
D1C	P5D
D1D	G12C
D2	G10D
D2A	D10
D2B	G13
D2C	G13A
D2D	G13B
D7A	G13C
G16	G14
G16A	G14A
G16B	G14B
G16C	G14C
G16D	G15
D7B	G15A
G17	G15B

Name and Descriptor	Name and Descriptor
G17A	G15C
G17B	P1
G17C	P1A
G17D	P1B
D7C	P1C
G7	P2
G7A	P2A
G7B	P2B
G7C	P2C
G7D	P3
G8	P3A
G8A	P3B
G8B	P3C
G8C	P4
G9	E1
G9A	P4A
G9B	P4B
G9C	P4C
G10	P5
G10A	P5A
G10B	P5B
G10C	P5C
D9	E1A
G11	G12D
G11A	
G11B	
G11C	
G12	
G12A	
G12B	

Space Calculations

The space calculation procedure will not be discussed in detail because it is highly complex. However, a brief statement may be given on the requirements for performing the space calculations and the areas for which space is calculated.

Three requirements must be fulfilled before the space calculations can begin: 1) The page size must be chosen by the system manager for the installation; 2) the average length for each field must be defined; 3) the descriptors for each field must be assigned or the assumption of default descriptors must be made.

The space calculations were done for:

1) Table A - a dictionary of field names, which are character string values for fields containing few values, many values, or coded values.

2) Table B - the data file of the logical records that contain the values of all fields for which the descriptors had a visible value.

3) Table C - an inverted file divided into six-byte slots. It contains the distinct value of fields having key or numeric range descriptors.

4) Table D - an inverted file that indexes Table C. It stores the text of procedure created by the user language and, for preallocated fields, a record description.

For further information on space calculations see Chapter 3 in the File Manager's Technical Reference Manual (Computer Corporation of America, 1979a).

SPECIAL SOFTWARE

The special software written to perform the conversion included: 1) a program to interface with the GIPSY system. This program reads the GIPSY unformatted file (system storage format), selects the desired records and labels, adds data to the invisible data fields if they are to be part of the converted record, and keeps control totals on all records (see the "GIPSY Interface Program," this report); 2) a program to interface with the M204 system. This program reads M204 files, selects, loads and updates M204 records, and produces an audit trail, an edit list, and control totals (see "Model 204 Interface Program," this report); 3) a program to list the output records from the GIPSY interface program; this program produces images of the converted data. The images are used as a control source in determining a successful or unsuccessful conversion (see "Data-Image Program," this report); 4) two interactive procedures to speed up the conversion and evaluation, one to create, open, and initialize the M204 file and one to retrieve data from the file. The Clist M204N creates a file interactively and allocates the space for the tables. The data elements and their descriptors are also assigned by this Clist (see "TSO Procedures," this report). The Clist TM204 generates printouts and listings interactively. It was designed for general use. It retrieves and manipulates data from the file (see "TSO Procedures," this report).

```
1GIPREC: PROC OPTIONS(MAIN);
         DECLARE
           TAPE FILE RECORD,
           SYSPRINT FILE,
           CARD FILE RECORD,
           WORK CHAR (20000) VARYING,
           WORK1 CHAR (8000) VARYING,
           WORK2 CHAR (80),
           SPACET CHAR (100) INIT (' '),
           TABLE (800) FIXED BINARY (15,0),
           FMARK CHAR (1),
           RMARK CHAR (1),
           FORM CHAR (8),
           POINT FIXED BINARY (15,0),
           DATA FIXED BINARY (15,0),
           FIELD FIXED BINARY (15,0),
           ICODE FIXED BINARY (15,0),
           RLNTH FIXED BINARY (15,0),
           COUNT FIXED BINARY (15,0),
           FLNTH FIXED BINARY (15,0);
         DCL I FIXED BINARY (15,0);
         DCL J FIXED BINARY (15,0);
         DCL K FIXED BINARY (15,0);
         DCL L FIXED BINARY (15,0);
         DCL M FIXED BINARY (15,0);
         DCL N FIXED BINARY (15,0);
       DCL IPRCNT FIXED BINARY (15,0);
       DCL OPRCNT FIXED BINARY (15,0);
       DCL ERRCNT FIXED BINARY (15,0);
       DCL FLDCT FIXED BINARY (15,0);
       DCL FLDNO FIXED BINARY (15,0);
START:   OPEN FILE (TAPE) INPUT;
         OPEN FILE (CARD) INPUT;
           ON ENDFILE (CARD) GO TO A1;
         COUNT = 0;
         IPRCNT = 0;
         OPRCNT = 0;
         FLDNO = 429;
         FMARK = '_';
         RMARK = '@';
         DO I = 1 TO 800;
         READ FILE (CARD) INTO (WORK2);
         TABLE (I) = SUBSTR(WORK2,16,5);
         FLDNO = I;
         END;
         PUT PAGE;
         GO TO A1;
A:       M = 80;
         K = 1;
         DO I = 1 TO J;
         PUT FILE (TAPEOUT) EDIT (SUBSTR(WORK1,K,M)) (A(M));
         K = K+80;
         OPRCNT = OPRCNT+1;
         END;
A1:      READ FILE (TAPE) INTO (WORK);
           ON ENDFILE (TAPE) GO TO ENDIT;
         COUNT = COUNT+1;
         IPRCNT = IPRCNT+1;
         RLNTH = UNSPEC(SUBSTR(WORK,1,2));
         DATA = UNSPEC(SUBSTR(WORK,5,2));
         FORM = SUBSTR(WORK,7,8);
```

```
        POINT = 17;
        L = 1;
        N = 0;
        FLDCT = 0;
B:      ICODE = UNSPEC(SUBSTR(WORK,POINT,2));
        FIELD = UNSPEC(SUBSTR(WORK,POINT+2,2));
        N = N+1;
B1:     IF L >= 8000 THEN GO TO G;
        IF ICODE > TABLE (N) THEN GO TO D;
        IF ICODE < TABLE (N) THEN DO;
        POINT = POINT+4;
        IF POINT >= DATA THEN GO TO E;
        N = N-1;
        GO TO B;
        END;
        IF FIELD < 0 THEN FLNTH = 0;
        ELSE DO;
        FLNTH = UNSPEC(SUBSTR(WORK,FIELD+DATA+1,2));
        END;
        K = 3;
        IF FLNTH = 0 THEN DO;
        SUBSTR(WORK1,L,1) = FMARK;
        L = L+1;
        FLDCT = FLDCT+1;
        IF L > 8000 THEN GO TO G;
        GO TO C;
        END;
        DO I = 0 BY 65 WHILE (FLNTH >= 1);
          IF FLNTH   >= 65 THEN J = 65;
            ELSE J = FLNTH  ;
        IF FLNTH <= J THEN DO;
        SUBSTR(WORK1,L,J) = SUBSTR(WORK,FIELD+DATA+K,J);
        L = L+J;
        SUBSTR(WORK1,L,1) = FMARK;
        L = L+1;
        FLDCT = FLDCT+1;
          GO TO C;
          END;
        SUBSTR(WORK1,L,J) = SUBSTR(WORK,FIELD+DATA+K,J);
        L = L+J;
        FLNTH = FLNTH-J;
        K = K+J;
        END;
C:      POINT = POINT+4;
        IF POINT >= DATA THEN GO TO E;
        ELSE GO TO B;
D:      SUBSTR(WORK1,L,1) = FMARK;
        L = L+1;
        N = N+1;
        FLDCT = FLDCT+1;
        GO TO B1;
E:      DO I = 1 BY 1 WHILE (TABLE (N) < 9999);
        SUBSTR(WORK1,L,1) = FMARK;
        L = L+1;
        N = N+1;
        FLDCT = FLDCT+1;
        END;
EA:     IF FLDCT > FLDNO THEN DO;
          L = L-1;
          N = N-1;
        FLDCT = FLDCT-1;
```

```
            GO TO EA;
          END;
EB:       IF FLDCT < FLDNO THEN DO;
            SUBSTR(WORK1,L,1) = FMARK;
            FLDCT = FLDCT+1;
            L = L+1;
            N = N+1;
            GO TO EB;
          END;
          SUBSTR(WORK1,L,1) = RMARK;
          L = L+1;
          N = L;
          J = 0;
          DO I = 1 TO 100;
          IF N >= 80 THEN J = J+1;
          ELSE GO TO F;
          N = N-80;
          END;
F:        IF N = 80 THEN GO TO A;
          IF N = 0 THEN GO TO A;
          DO N = N BY 1 WHILE (N < 81);
          SUBSTR(WORK1,L,1) = ' ';
          L = L+1;
          END;
          J = J+1;
          GO TO A;
G:        PUT EDIT (L) (F(5));
          PUT EDIT (WORK1,1,80) (A(80));
          GO TO A1;
ENDIT:
          ERRCNT = COUNT-OPRCNT;
          PUT SKIP DATA (IPRCNT,OPRCNT,ERRCNT);
          END GIPREC;
//
```

```
M204A: PROCEDURE OPTIONS(MAIN);
       DECLARE
       SYSPRINT FILE,
       SYSIN FILE,
         IFILE FILE INPUT,
         LFILE FILE INPUT,
       OFILE FILE,
       IFINIT EXTERNAL ENTRY (FIXED BIN (31,0),CHAR (*)),
       IFDFLD EXTERNAL ENTRY (FIXED BIN (31,0),CHAR (*)),
       IFOPEN EXTERNAL ENTRY (FIXED BIN (31,0),CHAR (*)),
       IFCLOSE EXTERNAL ENTRY (FIXED BIN (31,0)),
       IFFNSH EXTERNAL ENTRY (FIXED BIN (31,0)),
       IFBREC EXTERNAL ENTRY (FIXED BIN (31,0),CHAR (*),CHAR (*)),
       IFFIND EXTERNAL ENTRY (FIXED BIN (31,0),CHAR (*),CHAR (*),
                              CHAR (*),CHAR (*)),
       IFSTRT EXTERNAL ENTRY (FIXED BIN (31,0),FIXED BIN (31,0),
                              CHAR (*),FIXED BIN (31,0),
                              FIXED BIN (31,0)),
       IFERR EXTERNAL ENTRY (FIXED BIN (31,0),FIXED BIN (31,0),
                              CHAR (*));
       DECLARE
       IFGET EXTERNAL ENTRY (FIXED BIN (31,0),CHAR (*), CHAR (*),
                              CHAR (*),CHAR (*),CHAR (*)),
       IFPUT EXTERNAL ENTRY (FIXED BIN (31,0),CHAR (*),CHAR (*)),
       IBUF1 CHAR (80) INIT ('     '),
       IBUF2 CHAR (10000) VARYING,
       OBUF1 CHAR (255),
       OBUF2 CHAR (5000) VARYING,
       LBUF2 (500) CHAR (80),
       THRD1 FIXED BIN (31,0) INIT (0),
       THRD2 FIXED BIN (31,0) INIT (0),
       Z1 FIXED BIN (31,0),
       Z2 FIXED BIN (31,0),
       Z3 FIXED BIN (31,0),
       Z4 FIXED BIN (31,0),
       I      FIXED BIN (31,0) INIT (0),
       J      FIXED BIN (31,0) INIT (0),
       K      FIXED BIN (31,0) INIT (0),
       L      FIXED BIN (31,0) INIT (0),
       X0     FIXED BIN (31,0) INIT (0),
       X1     FIXED BIN (31,0) INIT (0),
       X2     FIXED BIN (31,0) INIT (0),
       X3     FIXED BIN (31,0) INIT (0),
       X4     FIXED BIN (31,0) INIT (0),
       X5     FIXED BIN (31,0) INIT (0),
       X6     FIXED BIN (31,0) INIT (0),
       X7     FIXED BIN (31,0) INIT (0),
       X9     FIXED BIN (31,0) INIT (0),
       SUBSTR BUILTIN,
       INDEX BUILTIN,
       STRING BUILTIN,
       STR CHAR (500) INIT ('          '),
       KEYDATA CHAR (50),
       SPACES CHAR (255) INIT ('     ');
       DCL 1 EDTPT2,
             2 ERR FIXED BIN (31,0),
             2 EDT CHAR (6) INIT ('EDIT('),
             2 LBUF1 CHAR (10),
             2 STR2 CHAR (5) INIT ('') (A('),
             2 X8 FIXED BIN (31,0),
             2 STR1 CHAR (3) INIT ('))'),
```

```
          2 PT CHAR (4) INIT ('PUT;');
        DCL 1 EDTPAT,
              2 EPT1 PICTURE '9',
              2 EPT2 PICTURE '99',
              2 EPT3 PICTURE '999',
              2 EPT4 PICTURE '9999';
        DCL SBCT FIXED BIN (31,0) INIT(0);
          DCL KEYWD CHAR (80);
        DCL RECIN FIXED BIN (15,0) INIT (0);
        DCL RECOUT FIXED BIN (15,0) INIT (0);
        DCL GIPRC FIXED BIN (31,0) INIT (0);
P1:
        PUT SKIP LIST ('PROGRAM M204A HAS BEEN ENTERED');
        KEYDATA = SPACES;
        KEYWD = SPACES;
          SUBSTR(STR,251,250) = SPACES;
          SUBSTR(STR,1,250) = SPACES;
        OBUF1 = SPACES;
      LBUF1 = SPACES;
      OPEN FILE (IFILE);
      ON ENDFILE (IFILE) GO TO P9;
      OPEN FILE (LFILE);
      ON ENDFILE (LFILE) I = 501;
        PUT SKIP LIST ('INPUT AND OUTPUT FILES HAVE BEEN OPENED');
        CALL IFSTRT (ERR,3,'VG9225G;GEO;',1,THRD1);
      IF ERR ¬= 0 THEN DO;
        PUT SKIP LIST ('ERROR IN IFSTRT STATEMENT = ',ERR);
        GO TO P9;
      END;
        PUT SKIP LIST ('IFSTART ROUTINE HAS BEEN CALLED');
        CALL IFOPEN (ERR,'MDSMST;;');
      IF ERR ¬= 0 THEN DO;
        PUT SKIP LIST ('ERROR IN IFOPEN STATEMENT = ',ERR);
        IF ERR = 16 THEN GO TO P1A;
        IF ERR = 32 THEN GO TO P1A;
        GO TO P9;
      END;
P1A:
        PUT SKIP LIST ('IFOPEN ROUTINE HAS BEEN COMPLETED');
P2:
        PUT SKIP LIST ('BEGIN READING FILE LFILE');
      DO I = 1 TO 501;
        GET FILE (LFILE) EDIT (IBUF1) (A(80));
        J = J + 1;
        IF I < 501 THEN LBUF2 (J) = IBUF1;
      END;
      J = J - 1;
        PUT SKIP LIST ('FILE LFILE HAS BEEN READ');
        PUT SKIP LIST ('NUMBER OF LFILE RECORDS ARE ',J);
P3:
      X0 = 0;
      X1 = 1;
      X2 = 80;
      X3 = 0;
      X9 = J;
      X7 = 1;
      X5 = 0;
      X8 = 255;
      X4 = 0;
        PUT SKIP LIST ('BEGIN READING FILE IFILE');
P4:
```

```
      GIPRC = GIPRC + 1;
    DO I = 1 TO 200;
      GET FILE (IFILE) EDIT (IBUF1) (A(80));
      RECIN = RECIN +1;
      X0 = INDEX (IBUF1,'@');
      IF X0 > 0 THEN DO;
        I = 200;
        X2 = X0;
      END;
      X3 = X3 + X2;
      SUBSTR(IBUF2,X1,X2) = IBUF1;
      X1 = X1 + X2;
    END;
    IF X0 = 0 THEN GO TO P9;
    X1 = 1;
    X2 = X3;
      PUT SKIP LIST ('RECORDS READ FROM IFILE = ',RECIN);
      PUT SKIP LIST ('GIPSY RECORDS READ = ',GIPRC);
P5:
      RECOUT = RECOUT + 1;
      PUT SKIP LIST ('RECORDS CALLED FROM M204 FILE = ',RECOUT);
      CALL IFBREC (ERR,';','MDSMST');
      IF ERR ¬= 0 THEN DO;
        PUT SKIP LIST ('ERROR IN IFBREC STATEMENT = ',ERR);
        GO TO P9;
      END;
        K = 0;
    DO I = 1 TO 500;
        K = K + 1;
      X0 = INDEX(SUBSTR(IBUF2,X1,X2),'_');
      X5 = X5 + 1;
      IF X0 = 0 THEN GO TO P3;
      IF X0 = 1 THEN GO TO P6;
      IF X5 > X9 THEN GO TO P8;
      X4 = X4 + X0;
      X4 = X4 - 1;
      LBUF1 = SUBSTR(LBUF2 (K),1,10);
      OBUF2 = SUBSTR(IBUF2,X1,X0-1);
      DO WHILE (X4 > 0);
      IF X4 < 256 THEN X8 = X4;
      OBUF1 = SPACES;
     OBUF1 = SUBSTR(OBUF2,X7,X8);
        IF X8 > 0 & X8 < 10 THEN DO;
          EPT1 = X8;
          STR = (EDT || LBUF1 || STR2 || EPT1 || STR1);
        END;
        IF X8 > 9 & X8 < 100 THEN DO;
          EPT2 = X8;
          STR = (EDT || LBUF1 || STR2 || EPT2 || STR1);
        END;
        IF X8 > 99 & X8 < 1000 THEN DO;
          EPT3 = X8;
          STR = (EDT || LBUF1 || STR2 || EPT3 || STR1);
        END;
        IF X8 > 999 & X8 < 10000 THEN DO;
          EPT4 = X8;
          STR = (EDT || LBUF1 || STR2 || EPT4 || STR1);
        END;
        IF X8 > 9999 THEN GO TO P9;
      CALL IFPUT (ERR,OBUF1,STR);
      IF ERR ¬= 0 THEN DO;
```

```
          PUT SKIP LIST ('ERROR IN IFPUT STATEMENT = ',ERR);
          PUT SKIP LIST (OBUF1);
          PUT SKIP LIST (STR);
          GO TO P9;
        END;
        SUBSTR(STR,1,250) = SPACES;
        SUBSTR(STR,251,250) = SPACES;
        IF X4 < 256 THEN SBCT = X4;
        IF X4 > 255 THEN SBCT = 255;
        X4 = X4 - SBCT;
        X7 = X7 + SBCT;
        END;
        Z1 = 1;
        Z2 = X0;
            Z2 = Z2 - 1;
        DO WHILE (Z2 > 0);
        Z4 = INDEX(SUBSTR(OBUF1,Z1,Z2),' ');
          IF Z4 = 1 THEN GO TO P5A;
          IF Z4 = 0 THEN Z4 = Z2 + 1;
          SUBSTR(KEYDATA,1,Z4-1) = SUBSTR(OBUF1,Z1,Z4);
          X8 = Z4 - 1;
          IF X8 > 0 & X8 < 10 THEN DO;
            EPT1 = X8;
          KEYWD = (EDT || 'KEYWORD(*)' || STR2 || EPT1 || STR1);
          END;
          IF X8 > 9 & X8 < 100 THEN DO;
            EPT2 = X8;
          KEYWD = (EDT || 'KEYWORD(*)' || STR2 || EPT2 || STR1);
          END;
          CALL IFPUT (ERR,KEYDATA,KEYWD);
          IF ERR ¬= 0 THEN DO;
          PUT SKIP LIST ('ERROR IN IFPUT STATEMENT = ',ERR);
          PUT SKIP LIST (KEYWD);
          PUT SKIP LIST (KEYDATA);
            GO TO P9;
        END;
          KEYDATA = SPACES;
          KEYWD = SPACES;
P5A:
          Z1 = Z1 + Z4;
          Z2 = Z2 - Z4;
        END;
P6:
        X7 = 1;
        X8 = 255;
        X2 = X2 - X0;
        X1 = X1 + X0;
        END;
P7:
P8:
      PUT SKIP LIST ('ERROR IN COMMAND STATEMENTS');
      CALL IFERR (ERR,'128','ERROR IN COMMAND STATEMENTS');
      GO TO P3;
P9:
        PUT SKIP LIST ('END OF PROGRAM M204A');
      CALL IFFNSH (ERR);
      CLOSE FILE (IFILE);
      CLOSE FILE (LFILE);
        END M204A;
//
```

DATA-IMAGE PROGRAM

EASYTRIEVE CONTROL STATEMENTS

```
INPUT B1 1 80 A
      B2 1 8 A
      B2 17 4 A
9999
IF B1 NQ *
LIST B1
```

TSO(TIME-SHARING-OPTION) PROCEDURES

INTERACTIVE CLIST USED FOR FILE INITIALIZATION

M204N

```
ALLOC F(CCAIN) DA('VG9225g.M204N.DATA') SHR
ALLOC F(SYSPRINT) DA(*)
ALLOC F(SYSAUDIT) DA('VG9225G.M204.AUDIT') SHR
ALLOC F(CCASTAT) DA('VG9225G.M204.CCASTAT') SHR
ALLOC F(CCASNAP) DA('VG9225G.M204.SNAP') SHR
ALLOC F(CCATEMP) DA('VG9225G.M204.CCATEMP') SHR
ALLOC F(MDSMST) DA('RIF.M204.MDF') SHR
ALLOC F(SYSDUMP) DA(*)
CALL 'SYS1.M204.LOAD(BATCH204)'
```

INTERACTIVE CLIST USED FOR RETRIEVING DATA

TM204

```
ALLOC F(CCAIN) DA(*)
ALLOC F(SYSDUMP) DA(*)
ALLOC F(SYSPRINT) DA(*)
ALLOC F(MDSMST) DA('RIF.M204.MDF') SHR
ALLOC F(CCATEMP) DA('VG9225g.M204.CCATEMP) SHR
ALLOC F(CCASTAT) DA('VG9225G.M204.CCASTAT') SHR
ALLOC F(SYSAUDIT) DA('VG9225G.M204.AUDIT') SHR
ALLOC F(CCASNAP) DA('VG9225G.M204.SNAP') SHR
CALL 'SYS1.M204.LOAD(BATCH204)'
```

CREATING THE DATA BASE

The data base is created in five steps:

Step 1. - Execute IEFBR14 on IBM utility, which allocates space.

Step 2. - Set page size to installation specification.

Step 3. - Execute create command to establish the file name and to set the number of pages required for each file control table.

Step 4. - Execute the initialize command to erase all information stored in the file except the file settings and establish the optional sort or hash key files.

Step 5. - Execute the define command to define the field name and the attribute associated with each field.

LOADING THE DATA BASE

The procedure for loading the data base is as follows:

Step 1. - The 623 records selected from the GIPSY unformatted file were divided into segments of 100 records each.

Step 2. - The conversion procedure (fig. 1) was executed to convert the records by segment and load the records into the M204 file.

Step 3. - After each segment was loaded, the audit trail produced was checked, and the loaded file was backed up.

A block diagram of the conversion system is given in figure 2.

FIGURE 1

GEMERALIZED FLOW CHART OF CONVERSION SYSTEM

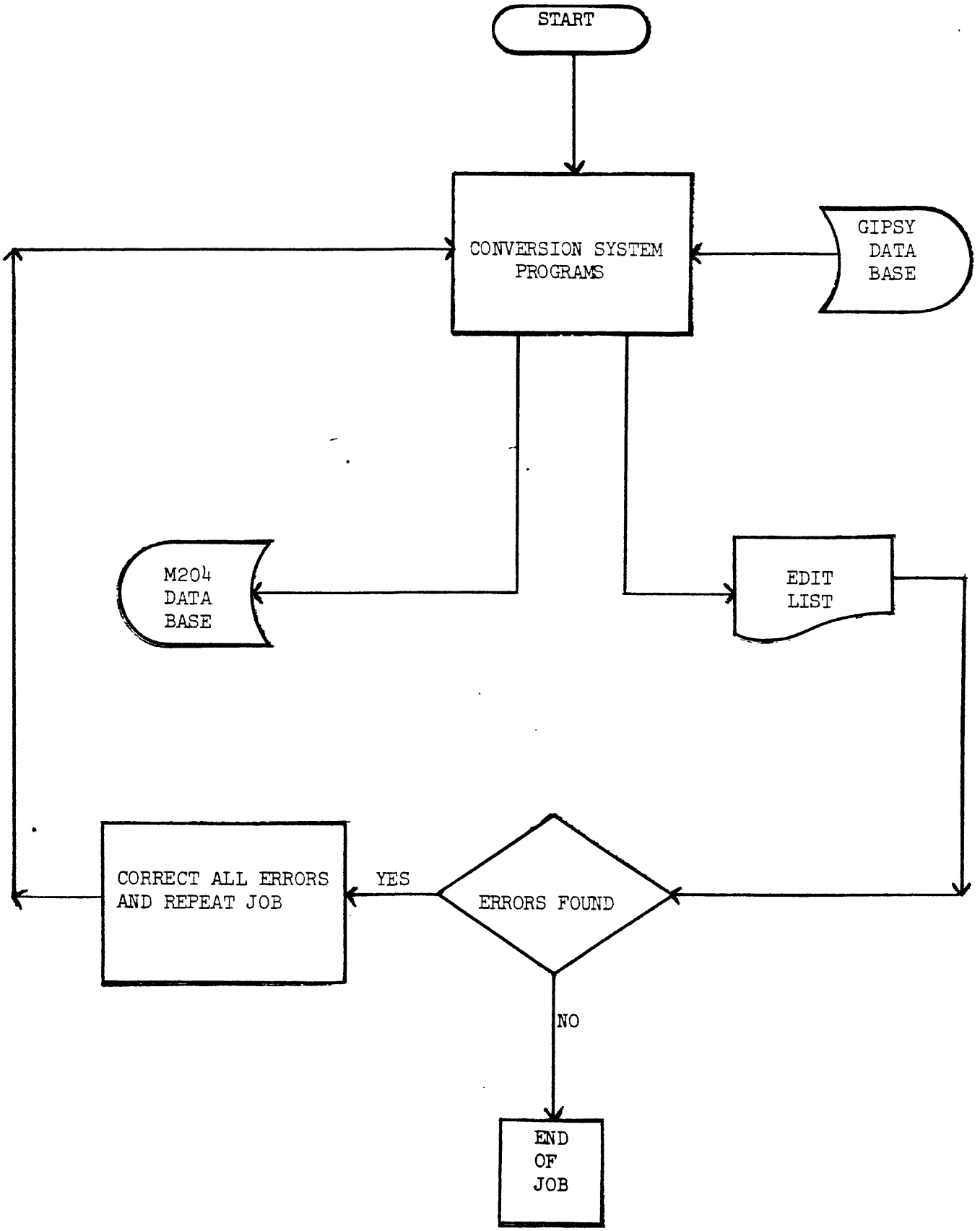
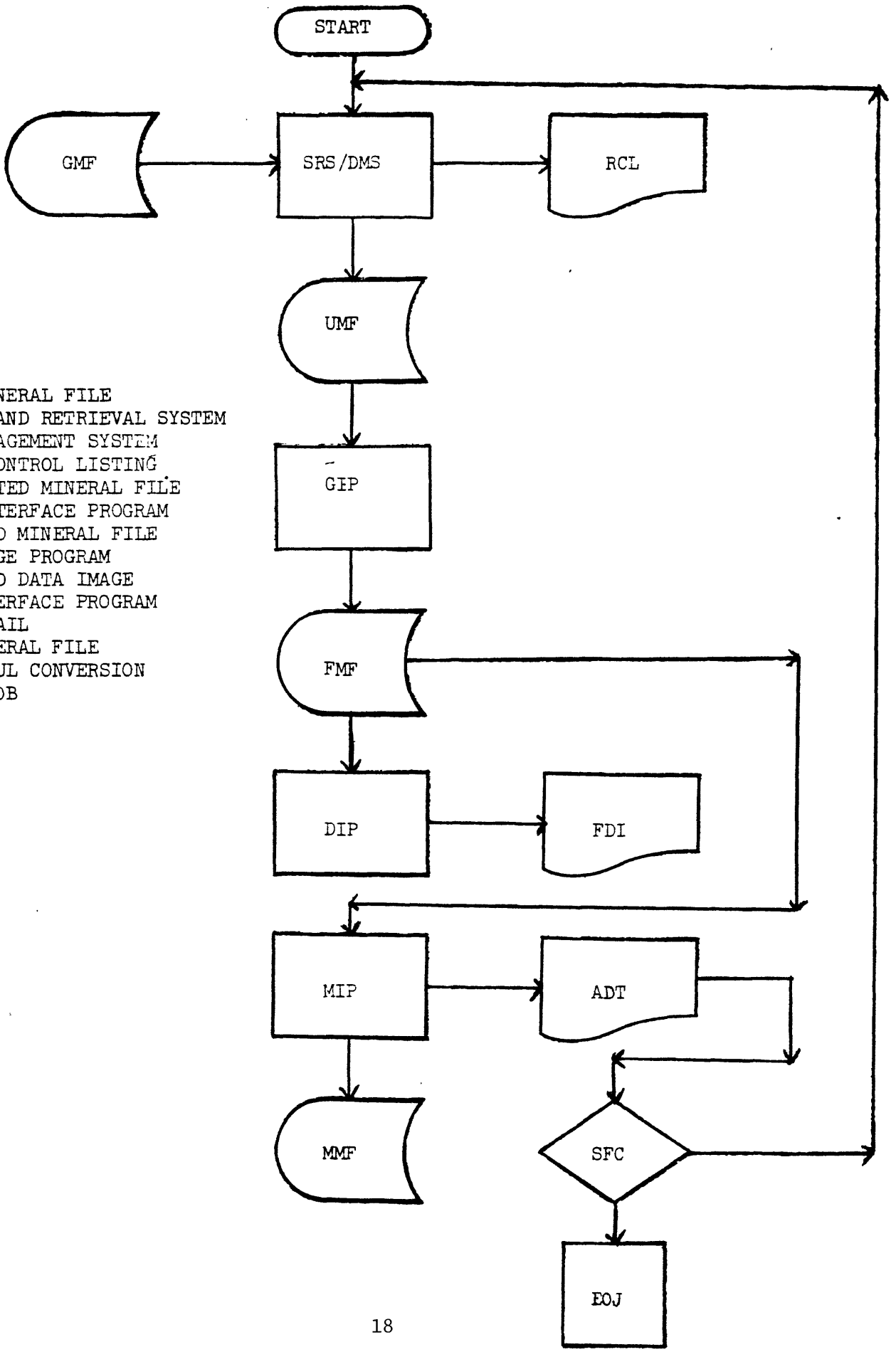


FIGURE 2

DETAILED FLOW CHART OF CONVERSION SYSTEM



GMF = GIPSY MINERAL FILE
SRS = STORAGE AND RETRIEVAL SYSTEM
DMS = DATA MANAGEMENT SYSTEM
RCL = RECORD CONTROL LISTING
UMF = UNFORMATTED MINERAL FILE
GIP = GIPSY INTERFACE PROGRAM
FMF = FORMATTED MINERAL FILE
DIP = DATA IMAGE PROGRAM
FDI = FORMATTED DATA IMAGE
MIP = M204 INTERFACE PROGRAM
ADT = AUDIT TRAIL
MMF = M204 MINERAL FILE
SFC = SUCCESSFUL CONVERSION
EOJ = END OF JOB

COST ANALYSIS

The cost analysis is based on the results of identical jobs running the GISPY and M204 systems. The runs were made against the same data.

For all jobs, the unused core charge is substracted from the total cost because the unused core charge can be eliminated by fine tuning the jobs. For all jobs, the printing and cards read charges would be the same and are not shown.

JOB1:

This run illustrates the charges associated with a Class A run that creates a simple record.

CHARGES	GIPSY	M204
I/O (Input/output)	$ 0.17	$ 0.38
CPU (central processing units)	$ 0.79	$ 0.47
FIXED	$ 3.09	$ 3.09
UNUSED CORE	$ 0.00	$ 4.18
TOTAL	$ 4.05	$ 8.12
MINUS UNUSED CORE	$ 0.00	$ 4.18
CORRECTED TOTAL	$ 4.05	$ 3.94

JOB2:

THIS RUN ILLUSTRATES THE CHARGES ASSOCIATED WITH A CLASS A RUN THAT UPDATES A SIMPLE RECORD.

CHARGES	GIPSY	M204
I/O	$ 1.16	$ 0.36
CPU	$ 0.92	$ 0.51
FIXED	$ 3.09	$ 3.09
UNUSED CORE	$ 4.97	$ 4.20
TOTAL	$ 10.14	$ 8.16
MINUS UNUSED CORE	$ 4.97	$ 4.20
CORRECTED TOTAL	$ 5.17	$ 3.96

JOB3:

THIS RUN ILLUSTRATES THE CHARGES ASSOCIATED WITH A CLASS A RUN THAT RETRIEVES 1 RECORD.

CHARGES	GIPSY	M204
I/O	$ 1.04	$ 0.34
CPU	$ 0.65	$ 0.47
FIXED	$ 3.09	$ 3.09
UNUSED CORE	$ 2.20	$ 4.13
TOTAL	$ 6.98	$ 8.03
MINUS UNUSED CORE	$ 2.20	$ 4.13
CORRECTED TOTAL	$ 4.78	$ 3.90

JOB4:

THIS JOB WAS SET UP BY JIM CALKINS AS A SPECIAL GIPSY RETRIEVAL; THE RUN ILLUSTRATES CHARGES ASSOCIATED WITH A CLASS D RUN THAT RETRIEVES, SORTS, AND LISTS 623 RECORDS.

CHARGES	GIPSY	M204
I/O	$ 6.74	$ 3.35
CPU	$ 1.85	$ 3.38
STORAGE	$ 0.18	$ 1.44
Listing	$ 1.24	$ 1.28
FIXED	$ 0.62	$ 0.62
UNUSED CORE	$ 0.00	$ 9.32
TOTAL	$10.63	$19.39
MINUS UNUSED CORE	$ 0.00	$ 9.32
CORRECTED TOTAL	$10.63	$10.07

Storage:

The illustration below represents the converted file storage charge. The file contains 623 records.

STORAGE	GIPSY	M204
TRACKS USED	83	214
COST/TRACK/MO	$ 0.07	$ 0.07
TOTAL COST/MO	$ 5.81	$ 14.98

SUMMARY

1. The complete project took approximately 240 man-hours including training and documentation.

2. The total cost of the project was approximately $8,000.

3. Model 204 provides an alternative system to GIPSY for managing the MDS.

4. A system to convert GIPSY records to M204 records has been devised.

5. A conversion procedure is established.

6. The advantages of M204 are the:

   a. Flexible user language

   b. Report writer

   c. Security ability

   d. Multifile access capability

7. The disadvantages of M204 are the:

   a. Inability to search NON-KEY fields

   b. Inability to do prefix and suffix searches on fields that contain uncontrolled data.

8. The master file can be converted for $0.75 to $1.50 per record. The maximum cost should not exceed $2.00 per record.

9. Execution costs for making GIPSY runs are somewhat higher than costs for M204 runs; storage costs for M204 are more than double such costs for GIPSY.

CONCLUSION

As presently constituted, MDS is comprised principally of the CRIB, an acronym for computerized resource information bank. This is a library file of mostly descriptive information on mines, deposits, and occurrences and their locations, and is readily amenable to graphic displays of the data. As such there is no immediate incentive to convert it to a DBMS compatible with systems of other Interior agencies.

AN EXAMPLE OF AN INTERACTIVE JOB THAT INITIALIZES A MODEL 204 FILE BY EXECUTING THE CLIST (M204N)

The tasks performed by this job are:

1) allocation of space required for each job-file.

2) The assignment of each data element associated with the M204 file.

3) The assignment of each data element descriptor.

This initialization had to be run before any data could be stored. Therefore, to accomplish this task, the following was executed.

M204N.CLIST

```
00010 ALLOC F(CCAIN) DA('VG9225G.M204N.DATA') SHR
00020 ALLOC F(SYSPRINT) DA(*)
00030 ALLOC F(SYSAUDIT) DA('VG9225G.M204.AUDIT') SHR
00040 ALLOC F(CCASTAT) DA('SYS1.M204.CCASTAT') SHR
00050 ALLOC F(CCASNAP) DA('VG9225G.M204.SNAP') SHR
00060 ALLOC F(SYSDUMP) DA(*)
00070 ALLOC F(CCATEMP) DA('VG9225G.M204.CCATEMP') SHR
00080 ALLOC F(MDSMST) DA('RIF.M204.MDF') SHR
00090 CALL 'SYS1.M204.LOAD(BATCH204)'
```

```
exec m204n
81.051  FEB 20  16.09.46
                                                        PAGE 1

***         ******          MODEL 204 INITIALIZATION.  VERSION = 5B
                                                  **********
***         ******          EXECUTE PARAMETERS:
                                                  **********
***         ******          READING PARAMETERS
PAGESZ=6184
PAGESZ     6184         FILE PAGE SIZE
***         ******          CHKP - NO CHKPOINT DD STATEMENT
***         ******          INITIALIZATION COMPLETED.  BUFFERS = 5
                                                  **********
***         ******          TIMELEFT = 86398.477
CREATE FILE MDSMST
***         ******          BEGIN FILE CREATION:  MDSMST
***         ******          READING FILE PARAMETERS
PARAMETER ASTRPPG=200,ATRPG=2,FVFPG=10,MVFPG=20
ASTRPPG    200          TABLE A STRINGS PER PAGE
ATRPG      2            TABLE A ATTRIBUTE PAGES
FVFPG      10           TABLE A FEW VALUED FIELD PAGES
MVFPG      20           TABLE A MANY VALUED FIELD PAGES
PARAMETER BSIZE=250,BRESERVE=200,BRECPPG=2
BSIZE      250          PAGES IN TABLE B
BRESERVE   200          TABLE B RESERVE SPACE
BRECPPG    2            TABLE B RECORDS PER PAGE
PARAMETER CSIZE=200,DSIZE=200
CSIZE      200          PAGES IN TABLE C
DSIZE      200          PAGES IN TABLE D
END
***         ******          END FILE CREATION:  MDSMST
OPEN MDSMST
***         ******          FILE MDSMST   OPENED
***         ******          FILE NOT INITIALIZED
INITIALIZE
***         ******          BEGIN FILE INITIALIZATION:  MDSMST
                                                  **********
***         ******          END FILE INITIALIZATION:  MDSMST
                                                  **********

DEFINE A10
DEFINE A11
DEFINE A12
DEFINE A13
DEFINE A20 (KEY,FRV)
DEFINE A21 (KEY,FRV)
DEFINE A22 (KEY,FRV)
DEFINE A30
DEFINE A40 (KEY,FRV)
DEFINE A50 (KEY,FRV)
DEFINE A60
DEFINE A64 (KEY,FRV)
DEFINE B40
DEFINE A70
DEFINE A80
DEFINE A107
DEFINE A82
DEFINE A83
DEFINE C10
DEFINE C11
DEFINE C12
DEFINE MAJOR
DEFINE MINOR
```

```
DEFINE COPROD
DEFINE BYPROD
DEFINE POTEN
DEFINE OCCUR
DEFINE C20
DEFINE C30
DEFINE C31
DEFINE C32
DEFINE C40
DEFINE C42
DEFINE C42E
DEFINE C42A
DEFINE C42B
DEFINE C42C
DEFINE C42F
DEFINE C43
DEFINE C426
DEFINE C50
DEFINE L10
DEFINE L20
DEFINE L30
DEFINE L40
DEFINE L100
DEFINE L110
DEFINE M10
DEFINE M15
DEFINE M20
DEFINE M21
DEFINE M30
DEFINE M31
DEFINE M40
DEFINE M41
DEFINE M50
DEFINE M51
DEFINE M60
DEFINE M61
DEFINE M70
DEFINE M80
DEFINE M90
DEFINE M100
DEFINE M110
DEFINE M120 (KEY,FRV)
DEFINE M130 (KEY,FRV)
DEFINE M140 (KEY,FRV)
DEFINE M160
DEFINE M161
DEFINE M170
DEFINE M171
DEFINE M190
DEFINE M191
DEFINE M200
DEFINE M201
DEFINE M210
DEFINE M211
```

```
DEFINE M220
DEFINE PROD
DEFINE YES
DEFINE NO
DEFINE LGE
DEFINE MED
DEFINE SML
DEFINE D1
DEFINE D1A
DEFINE D1B
DEFINE D1C
DEFINE D1D
DEFINE D2
DEFINE D2A
DEFINE D2B
DEFINE D2C
DEFINE D2D
DEFINE D3
DEFINE D11
DEFINE D11A
DEFINE D11B
DEFINE D11C
DEFINE D11D
DEFINE D3A
DEFINE D3B
DEFINE D3C
DEFINE D3D
DEFINE D4
DEFINE D4A
DEFINE D4B
DEFINE D4C
DEFINE D4D
DEFINE D5
DEFINE D5A
DEFINE D5B
DEFINE D5C
DEFINE D5D
DEFINE D6
DEFINE D6A
DEFINE D6B
DEFINE D6C
DEFINE D6D
DEFINE D7D
DEFINE G7D
DEFINE G8D
DEFINE G9D
DEFINE G11D
DEFINE G13D
DEFINE G14D
DEFINE D7
DEFINE G15D
DEFINE P2D
DEFINE P3D
DEFINE P4D
```

```
DEFINE P5D
DEFINE D7A
DEFINE G16
DEFINE G16A
DEFINE G16B
DEFINE G16C
DEFINE G16D
DEFINE D7B
DEFINE G17
DEFINE G17A
DEFINE G17B
DEFINE G17C
DEFINE G17D
DEFINE D7C
DEFINE G7
DEFINE G7
DEFINE G7A
DEFINE G7B
DEFINE G7C
DEFINE G7D
DEFINE G8
DEFINE G8A
DEFINE G8B
DEFINE G8C
DEFINE G9
DEFINE G9A
DEFINE G9B
DEFINE G9C
DEFINE G10
DEFINE G10A
DEFINE G10B
DEFINE G10C
DEFINE D9
DEFINE G11
DEFINE G11A
DEFINE G11B
DEFINE G11C
DEFINE G12
DEFINE G12A
DEFINE G12B
DEFINE G12C
DEFINE G10D
DEFINE D10
DEFINE G13
DEFINE G13A
DEFINE G13B
DEFINE G13C
DEFINE G14
DEFINE G14A
DEFINE G14B
DEFINE G14C
DEFINE G15
DEFINE G15A
DEFINE G15B
```

```
DEFINE G15C
DEFINE P1
DEFINE P1A
DEFINE P1B
DEFINE P1C
DEFINE P2
DEFINE P2A
DEFINE P2B
DEFINE P2C
DEFINE P3
DEFINE P3A
DEFINE P3B
DEFINE P3C
DEFINE P4
DEFINE E1
DEFINE P4A
DEFINE P4B
DEFINE P4C
DEFINE P5
DEFINE P5A
DEFINE P5B
DEFINE P5C
DEFINE E1A
DEFINE G12D
DEFINE P1D
DEFINE E1B
DEFINE G18
DEFINE G18A
DEFINE G18B
DEFINE G18C
DEFINE G18D
DEFINE E1C
DEFINE E1D
DEFINE H1
DEFINE H1A
DEFINE H1B
DEFINE H1C
DEFINE H1D
DEFINE H7
DEFINE H8
DEFINE J1
DEFINE J1A
DEFINE J1B
DEFINE J1C
DEFINE J1D
DEFINE J8
DEFINE K1
DEFINE K1A
DEFINE K2
DEFINE K2A
DEFINE K3
DEFINE K4
DEFINE K5
DEFINE N5
```

```
DEFINE N15
DEFINE N30
DEFINE N40
DEFINE N45
DEFINE N50
DEFINE N55
DEFINE N60
DEFINE N70
DEFINE N75
DEFINE N80
DEFINE N85
DEFINE GEN
DEFINE F1
DEFINE F2
DEFINE F3
DEFINE F4
DEFINE G2
DEFINE G1
DEFINE G3
DEFINE G4
DEFINE B10 (KEY,FRV)
DEFINE B20
DEFINE U
DEFINE USGS
DEFINE C44
DEFINE BTU
DEFINE SUL
DEFINE ASH
DEFINE CARB
DEFINE C45
DEFINE C46
DEFINE C47
DEFINE KEYWORD (KEY,INV)
EOJ
```

AN EXAMPLE OF A JOB THAT CONVERTED GIPSY RECORDS AND LOADED M204 RECORDS

This is an example of a job that performs the task of converting a GIPSY file to a M204 file; all the resources involved in a conversion of a MDS file are displayed in this example. The resources are:

1) JCL (Job Control Language) the communication link between the operating system and the conversion system.

2) Control data - the SRS and DBMS commands, also the literals which may or may not be inserted.

3) Data elements - the labels or field names associated with the SRS and DBMS.

```
//WG9225G4 JOB XXXXXXXXX1,DUMP,50,200,700),'GMASON',CLASS=D
/*PROCLIB  VG9225G.MDB.PROCLIB,SYS002
//JOB1   EXEC  PROC=CONVERT,RGN=250K,RNAME='RIF.W0001.CRIB1',
//             TAPENO=,TAPEDSN=,TBLK=12960,TAPE=SYSDK
//MSORT.SORTIN DD *
B10      4242 1 02450 RECORD NO.............
B20      4242 1 02460 RECORD TYPE...........
U        6565 3 02464    U
USGS     4242 1 02470 COUNTRY/ORGANIZATION.  USGS
B40      4242 1 00170 DEPOSIT NO............
G2       4242 1 02420 NAME..............................
G1       4242 1 02430 DATE..............................
G3       4242 1 02432 UPDATED...........................
G4       4242 1 02434 BY................................
A10      0404 1 00030 DEPOSIT NAME..................
A11      0404 1 00040 SYNONYM NAME..................
A30      0404 1000080 MINING DISTRICT/AREA/SUBDIST.
A40      0404 1000100 COUNTRY CODE..................
A50      0404 1000110 STATE CODE....................
A60      0404 1000120 COUNTY........................
A64      0404 1 00155 LAND CLASSIFICATION...........
A70      0505 2 00210 LATITUDE
A80      2525 2+00220 LONGITUDE
A107     0404 1000230 ALTITUDE.
A82      0404 1000356 POSITION OF NEAREST LOCALITY:
A83      0404 1000357 LOCATION COMMENTS:
C10      0604 1 00370 COMMODITIES PRESENT...........
MAJOR    0617 1 00390 MAJOR PRODUCTS..
MINOR    0717 1 00400 MINOR PRODUCTS..
COPROD   0817 1 00410 COPRODUCTS......
BYPROD   0917 1 00420 BYPRODUCTS......
C11      0617 1 00371 MAIN COMMOD.....
C12      0717 1 00372 MINOR COMMOD....
POTEN    0617 1 00430 POTENTIAL.......
OCCUR    0717 1 00440 OCCURRENCE......
C20      0404 3000450 COMMODITY SPECIALIST INFORMATION:
C30      0404 3000460 ORE MATERIALS (MINERALS,ROCKS,ETC.):
C31      0508 3000461 MAIN ORE MINERALS:
C32      0508 3000462 MINOR ORE MINERALS:
C50      0404 3000500 COMMODITY COMMENTS:
C44      0505 1 04981 SOURCE REFERENCE..
BTU      0505 1 04982 BTU...............
SUL      0505 1 04983 SULFUR............
ASH      0505 1 04984 ASH...............
CARB     0505 1 04985 FIXED CARBON......
C45      0505 1 04986 VOLATILES.........
C46      0505 1 04987 MOISTURE..........
C47      0505 1 04988 THICKNESS OF COAL.
C43      0404 3000498 ANALYTICAL DATA(GENERAL)
C42A     0504 1000492 EXPLORATION M$............
C42B     0504 1 00493 DEVELOPMENT M$............
C42C     0504 1 00495 EXPANSION M$..............
C42E     0504 1000491 TOTAL INVESTMENTS M$.....
C42F     0504 1000496 MILL CAPACITY PER YR.(THOUS. UNITS)..
C426     0510 1 00499 YR APPLICABLE.......
C42      0504 3000490 ECONOMIC COMMENTS:
A20      0404 1 00070 STATUS OF EXPLOR. OR DEV.
A21      0404 1 00072                              PROPERTY IS ACTIVE
A22      0404 1 00073                              PROPERTY IS INACTIVE
L10      0404 1 00520 YEAR OF DISCOVERY........
L20      0404 1 00530 BY WHOM...................
```

```
L30      0404 1 00540 NATURE OF DISCOVERY......
L40      0404 1 00550 YEAR OF FIRST PRODUCTION.
A12      0404 1 00050 PRESENT/LAST OWNER.......
A13      0404 1 00060 PRESENT/LAST OPERATOR....
L100     0404 3000630 REPORTS AVAILABLE:
L110     0404 3000640 EXPLOR. AND DEVELOP. COMMENTS:
C40      0404 3000470 DEPOSIT TYPES:
M10      0404 1 00670 FORM/SHAPE OF DEPOSIT:
M15      0606 1 00685 SIZE OF DEPOSIT......
M20      0606 1 00690 DEPTH TO TOP ........
M21      0636 2+00692 UNITS
M30      0606 1 00700 DEPTH TO BOTTOM......
M31      0636 2+00702 UNITS
M40      0606 1 00710 MAX LENGTH...........
M41      0636 2+00712 UNITS
M50      0606 1 00720 MAX WIDTH............
M51      0636 2+00722 UNITS
M60      0606 1 00730 MAX THICKNESS........
M61      0636 2+00732 UNITS
M70      1106 1 00740 STRIKE OF OREBODY....
M80      1106 1 00750 DIP OF OREBODY.......
M90      1106 1 00760 PLUNGE OF OREBODY....
M100     1106 1 00770 DIRECTION OF PLUNGE..
M110     1104 3 00780 COMMENTS(DESCRIPTION OF DEPOSIT):
M120     0407 1 00800 SURFACE
M130     0407 1 00810 UNDERGROUND
M140     0407 1 00820 SURFACE AND UNDERGROUND
M160     0607 1 00840 DEPTH OF WORKINGS BELOW SURFACE.
M161     0649 2+00842 UNITS
M170     0607 1 00850 LENGTH OF WORKINGS..............
M171     0649 2+00852 UNITS
M190     1007 1 00870 OVERALL LENGTH OF MINED AREA....
M191     1049 2+00872 UNITS
M200     1007 1 00880 OVERALL WIDTH OF MINED AREA.....
M201     1049 2+00882 UNITS
M210     1007 1 00890 OVERALL AREA....................
M211     1049 2+00892 UNITS
M220     1104 3000900 COMMENTS(DESCRIP. OF WORKINGS):
PROD     0102 1-00905 PRODUCTION
YES      0808 1 00920 YES
NO       0808 1 00930 NO PRODUCTION
LGE      0808 1 00931 LARGE PRODUCTION
MED      0808 1 00932 MEDIUM PRODUCTION
SML      0808 1 00933 SMALL PRODUCTION
D1       0404 1 00950 1
D1A      2121 2+00960 AMT
D1B      3030 2+00970 TH UNITS
D1C      4242 2+00980 YEAR
D1D      4848 2+00982 GRADE
D2       0404 1 00990 2
D2A      2121 2+01000 AMT
D2B      3030 2+01010 TH UNITS
D2C      4242 2+01020 YEAR
D2D      4848 2+01022 GRADE
D3       0404 1 01030 3
D3A      2121 2+01040 AMT
D3B      3030 2+01050 TH UNITS
D3C      4242 2+01060 YEAR
D3D      4848 2+01062 GRADE
D4       0404 1 01070 4
D4A      2121 2+01080 AMT
```

```
D4B     3030 2+01090 TH UNITS
D4C     4242 2+01100 YEAR
D4D     4848 2+01102 GRADE
D5      0404 1 01110 5
D5A     2121 2+01120 AMT
D5B     3030 2+01130 TH UNITS
D5C     4242 2+01140 YEAR
D5D     4848 2+01142 GRADE
D6      0404 1 01150 6
D6A     2121 2+01160 AMT
D6B     3030 2+01170 TH UNITS
D6C     4242 2+01180 YEAR
D6D     4848 2+01182 GRADE
D7      0404 1 01190 7
D7A     2121 2+01200 AMT
D7B     3030 2+01210 TH UNITS
D7C     4242 2+01220 YEAR
D7D     4848 2+01183 GRADE
D11     0404 1 01031 8
D11A    2121 2+01032 AMT
D11B    3030 2+01033 TH UNITS
D11C    4242 2+01034 YEAR
D11D    4848 2+01035 GRADE
G7      0404 1 01223 8
G7A     2121 2+01224 AMT
G7B     3030 2+01225 TH UNITS
G7C     4242 2+01226 YEARS
G7D     5656 2+01184 GRADE
G8      0404 1 01227 9
G8A     2121 2+01228 AMT
G8B     3030 2+01229 TH UNITS
G8C     4242 2+01231 YEARS
G8D     5656 2+01185 GRADE
G9      0404 1 01232 10
G9A     2121 2+01233 AMT
G9B     3030 2+01234 TH UNITS
G9C     4242 2+01235 YEARS
G9D     5656 2+01186 GRADE
G10     0404 1 01236 11
G10A    2121 2+01237 AMT
G10B    3030 2+01238 TH UNITS
G10C    4242 2+01239 YEARS
G10D    5656 2+01249 GRADE
G11     0404 1 01241 12
G11A    2121 2+01242 AMT
G11B    3030 2+01243 TH UNITS
G11C    4242 2+01244 YEARS
G11D    5656 2+01187 GRADE
G12     0404 1 01245 13
G12A    2121 2+01246 AMT
G12B    3030 2+01247 TH UNITS
G12C    4242 2+01248 YEARS
G12D    5656 2+01298 GRADE
G13     0404 1 01251 14
G13A    2121 2+01252 AMT
G13B    3030 2+01253 TH UNITS
G13C    4242 2+01254 YEARS
G13D    5656 2+01188 GRADE
G14     0404 1  1255 15
G14A    2121 2+ 1256 AMT
G14B    3030 2+ 1257 TH UNITS
```

```
G14C     4242 2+01258 YEARS
G14D     5656 2+01189 GRADE
G15      0404 1  1259 16
G15A     2121 2+ 1261 AMT
G15B     3030 2+ 1262 TH UNITS
G15C     4242 2+01263 YEARS
G15D     5656 2+01191 GRADE
G16      0404 1 01201 17
G16A     2121 2+01202 AMT
G16B     3030 2+01203 TH UNITS
G16C     4242 2+01204 YEARS
G16D     5656 2+01205 GRADE
G17      0404 1 01211 18
G17A     2121 2+01212 AMT
G17B     3030 2+01213 TH UNITS
G17C     4242 2+01214 YEARS
G17D     5656 2+01215 GRADE
G18      0404 1 01301 19
G18A     2121 2+01302 AMT
G18B     3030 2+01303 TH UNITS
G18C     4242 2+01304 YEARS
G18D     5656 2+01305 GRADE
P1       0404 1  1266 17
P1A      2121 2+ 1267 AMT
P1B      3030 2+ 1268 TH UNITS
P1C      4242 2+01269 YEAR
P1D      4848 2+01299 GRADE
P2       0404 1  1271 18
P2A      2121 2+ 1272 AMT
P2B      3030 2+ 1273 TH UNITS
P2C      4242 2+01274 YEAR
P2D      4848 2+01192 GRADE
P3       0404 1  1275 19
P3A      2121 2+ 1276 AMT
P3B      3030 2+ 1277 TH UNITS
P3C      4242 2+01278 YEAR
P3D      4848 2+01193 GRADE
P4       0404 1  1279 20
P4A      2121 2+ 1281 AMT
P4B      3030 2+ 1282 TH UNITS
P4C      4242 2+01283 YEAR
P4D      4848 2+01194 GRADE
P5       0404 1  1284 21
P5A      2121 2+ 1285 AMT
P5B      3030 2+ 1286 TH UNITS
P5C      4242 2+01287 YEAR
P5D      4848 2+01195 GRADE
D9       0302 1001240 SOURCE OF INFORMATION (PRODUCTION)..
D10      0302 1001250 PRODUCTION COMMENTS....
E1       0404 1 01280 1
E1A      2121 2+01290 AMT
E1B      3030 2+01300 TH UNITS
E1C      4242 2+01310 YEAR
E1D      4848 2+01311 GRADE
H1       0404 1 01560 1
H1A      2121 2+01570 AMT
H1B      3030 2+01580 TH UNITS
H1C      4242 2+01590 YEAR
H1D      4848 2+01591 GRADE
H8       0404 1001805 SOURCE OF INFORMATION (RESERVES)..
H7       0404 1001800 COMMENTS (RESERVES)..
```

```
J1       0404 1 01830 1
J1A      2121 2*01840 AMT
J1B      3030 2*01850 TH UNITS
J1C      4242 2*01860 YEAR
J1D      4848 2*01861 GRADE
J8       0404 1002075 SOURCE OF INFORMATION (POT RESOURCES)..
K1       0404 1002090 AGE OF HOST ROCKS.............
K1A      0404 1 02091 HOST ROCK TYPES...............
K2       0404 1002100 AGE OF ASSOC. IGNEOUS ROCKS..
K2A      0404 1 02101 IGNEOUS ROCK TYPES............
K3       0404 1002110 AGE OF MINERALIZATION.........
K4       0404 1002120 PERTINENT MINERALOGY..........
K5       0404 1002130 IMPORTANT ORE CONTROL/LOCUS..
N5       0606 1 02170 MAJOR REGIONAL STRUCTURES..
N15      0606 1 02190 TECTONIC SETTING.............
N30      0711 1 02240 AGE:
N40      0711 1 02260 AGE:
N45      0711 1 02270 AGE:
N50      0711 1 02290 AGE:
N55      0711 1 02300 AGE:
N60      0711 1 02310 AGE:
N70      0606 3002330 SIGNIFICANT LOCAL STRUCTURES:
N75      0606 3002340 SIGNIFICANT ALTERATION:
N80      0606 3002350 GEOLOGICAL PROCESSES OF CONCENTRATION OR ENRICHMENT:
N85      0603 3002360 COMMENTS (GEOLOGY AND MINERALOGY):
GEN      0202 3002361 GENERAL COMMENTS
F1       0404 1 02380 1)
F2       0404 1 02390 2)
F3       0404 1 02400 3)
F4       0404 1 02401 4)
/*
//MSORT.SYSIN DD *
 SORT FIELDS=(17,4,A),FORMAT=CH,SIZE=E1000
/*
//QUESTRAN.SYSRDR DD *
FORM
 CRIB
SELECT
 A. B10<W028899> THRU <W029600>
 B. G2<SWEETWOOD>
 C. A40< SF >
 LOGIC A*B*C
SORT
 B10 7
DUMP
/*
//CONVERT.CARD1 DD *
007206ACTIVE
007308INACTIVE
080007SURFACE
081011UNDERGROUND
082004BOTH
090504PROD
092003YES
093002NO
093206MEDIUM
093105LARGE
093305SMALL
246401U
247004USGS
/*
```

```
//CONVERT.TAPEOUT DD DSN=&&GEO,SPACE=(TRK,(100)),DISP=(,PASS)
/*
//STEP1  EXEC  ET,REGION=150K
//FILEA  DD  DSN=&&GEO,DISP=(OLD,PASS),UNIT=SYSDK,
//SYSIN DD *
INPT    B1 1 80 A
        B3 1 8 A
        B2 17 4 A
9999
IF B1 NQ *
LIST B1
/*
//RUN     EXEC  PGM=IFAM4,REGION=500K,TIME=15,
//  PARM=('SYSOPT=144,LIBUFF=1024,MAXBUF=10,MINBUF=5,SPCORE=14288',
//      'LAUDIT=0')
//STEPLIB DD DSN=SYS1.M204.LOAD,DISP=SHR
//        DD    DSN=VG9225G.MDB.M24LIB,DISP=SHR
//CCAAUDIT DD SYSOUT=A
//CCASTAT DD DSN=SYS1.M204.CCASTAT,DISP=SHR
//CCATEMP DD   UNIT=SYSDK,SPACE=(TRK,40),
//             DISP=(NEW,DELETE)
//CCASNAP DD   SYSOUT=A
//SYSUDUMP DD DUMMY
//CCAPRINT DD  SYSOUT=A
//CHKPOINT DD DUMMY
//TAPEDEF DD DUMMY
//CCASERVR DD  UNIT=SYSDK,DISP=(NEW,DELETE),SPACE=(CYL,2)
//MDSMST DD    DSN=RIF.M204.MDF,DISP=SHR
//PLIDUMP DD DUMMY
//CCAIN  DD    *
NUSERS=2,NSERVS=2,PAGESZ=6184,NFILES=2,SPCORE=15288
IODEV=23
*SLEEP 3600
/*
//IFAM4IN DD   *
M204A
/*
//IFILE  DD    DSN=&&GEO,DISP=(OLD,DELETE,DELETE)
//LFILE  DD  DSN=VG9225G.FLDNM.DATA,DISP=SHR
//SYSPRINT DD SYSOUT=A
/*
//
//
```

```
---- HASP-II RE1 STATISTICS ----


      349 CARDS READ


      349 SYSOUT PRINT RECORDS
```

AN EXAMPLE OF SOME COMMON TASKS USING THE INTERACTIVE RETRIEVAL CLIST (TM204)

This is an example of a job using TM204.CLIST. This job demostrates the use of the common command functions. The commands that are displayed in this job are:

1) Store record - this command writes records onto a file.

2) ADD - this command adds a field or label to a record.

3) Change - this command changes a field.

4) Print - this command allows one to print any select record or part of a record. Also allows one to format the output.

5) Delete - this command is used to delete a field or record.

Following is an example of each command in use.

TM204.CLIST

```
00010 ALLOC F(CCAIN) DA(*)
00020 ALLOC F(SYSPRINT) DA(*)
00030 ALLOC F(SYSAUDIT) DA('VG9225G.M204.AUDIT') SHR
00040 ALLOC F(CCASTAT) DA('SYS1.M204.CCASTAT') SHR
00050 ALLOC F(CCASNAP) DA('VG9225G.M204.SNAP') SHR
00060 ALLOC F(SYSDUMP) DA(*)
00070 ALLOC F(CCATEMP) DA('VG9225G.M204.CCATEMP') SHR
00080 ALLOC F(MDSMST) DA('RIF.M204.MDF') SHR
00090 CALL 'SYS1.M204.LOAD(BATCH204)'
```

EXAMPLE OF AN EXERCISE USING STORE RECORD,ADD,CHANGE,
PRINT,DELETE FIELD AND RECORD COMMAND OF M204

```
open mdsmst
***        ******            FILE MDSMST   OPENED

b
1. fd b10 = w888888
2. fr 1
2.1 pai                          FILE BEFORE EXECUTION OF STORE RECORD
end                              PRINT COMMAND
```

```
b
1. store record
   b10 = w888888
   a10 = demo1 copper mine
   a40 = us
   a50 = 25                      STORE RECORD COMMAND
   c10 = cu
end
```

```
b
1. fd b10 = w888888
2. fr 1
2.1 pai                          FILE AFTER EXECUTING STORE RECORD COMMAND
end                              PRINT COMMAND

B10 = W888888
A10 = DEMO1 COPPER MINE
A40 = US
A50 = 25
C10 = CU
```

```
b
1. fd b10 = w888888
2. fr 1
2.1 pai
end

B10 = W888888
A10 = DEMO1 COPPER MINE
A40 = US
A50 = 25
C10 = CU
```

RECORD BEFORE USING ADD COMMAND

```
b
1. fd b10 = w888888
2. fr 1
2.1 add g1 = 05 81
end
```

ADD COMMAND

```
b
1. fd b10 = w888888
2. fr 1
2.1 pai
end

B10 = W888888
A10 = DEMO1 COPPER MINE
A40 = US
A50 = 25
C10 = CU
G1 = 05 81
```

RECORD AFTER EXECUTING ADD COMMAND

```
b
1. fd b10 = w888888
2. fr 1
2.1 pai
end

B10 = W888888
A10 = DEMO1 COPPER MINE
A40 = US
A50 = 25
C10 = CU
G1 = 05 81
```

RECORD BEFORE USING CHANGE COMMAND

```
b
1. fd b10 = w888888
2. fr 1
2.1 change g1 to 05/81
end
```

CHANGE COMMAND

```
b
1. fd b10 = w888888
2. fr 1
2.1 pai
end

B10 = W888888
A10 = DEMO1 COPPER MINE
A40 = US
A50 = 25
C10 = CU
G1 = 05/81
```

RECORD AFTER EXECUTING CHANGE COMMAND

```
b
1. fd b10 = w888888
2. fr 1
2.1 pai
end

B10 = W888888
A10 = DEM01 COPPER MINE
A40 = US
A50 = 25
C10 = CU
G1 = 05/81
```

RECORD BEFORE USING DELETE FIELD/LABEL COMMAND

```
b
1. fd b10 = w888888
2. fr 1
2.1 delete g1
end
```

DELETE FIELD/LABEL COMMAND

```
b
1. fd b10 = w888888
2. fr 1
2.1 pai
end

B10 = W888888
A10 = DEM01 COPPER MINE
A40 = US
A50 = 25
C10 = CU
```

RECORD AFTER EXECUTING DELETE FIELD/LABEL COMMAND

```
b
1. fd b10 = w888888
2. fr 1
2.1 pai
end

B10 = W888888
A10 = DEM01 COPPER MINE
A40 = US
A50 = 25
C10 = CU
```

FILE BEFORE USING DELETE RECORD COMMAND

```
b
1. fd b10 = w888888
2. fr 1
2.1 delete record
end
```

DELETE RECORD COMMAND

```
b
1. fd b10 = w888888
2. fr 1
2.1 pai
3. count records in 1
4. print count in 3
end

0
```

FILE AFTER EXECUTING DELETE RECORD COMMAND

```
close mdsmst
***        ******          FILE CLOSED: MDSMST
```

REFERENCES CITED

Computer Corporation of America, 1979a, File manager's manual, model 204, database management system: Cambridge, Mass., 213 p.

_____1979b, Command reference manual, model 204, database mangement system: Cambridge, Mass., 166 p.

_____1979c, User language manual, model 204, database management system: Cambridge, Mass., 184 p.

_____1979d, Terminal user's guide, model 204, database management system: Cambridge, Mass., 75 p.