

A Dataset Cross-reference Report

by William R. Machose
Pennsylvania Power & Light Company

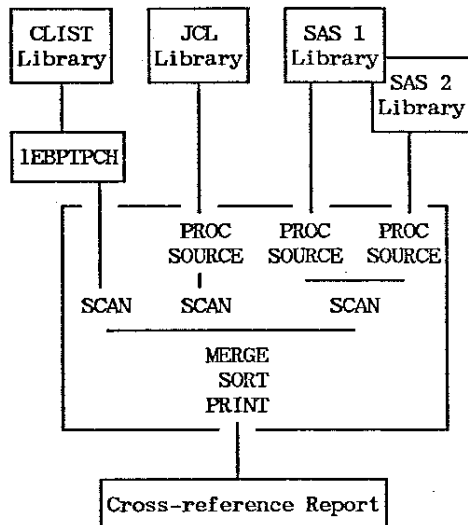
ABSTRACT

Have you ever tried to find all the programs, job control language (JCL), and CLISTS that refer to a specific dataset? Programs, JCL, and CLISTS are often held in partitioned dataset libraries in an OS environment. This poster describes a SAS® program that scans these libraries and prints a master index of dataset names and the library members that reference them.

OVERVIEW

As any department matures in its use of the computer, it accumulates an extensive inventory of programs, JCL, and CLISTS. Frequently, the production versions of these files are placed in partitioned datasets called libraries. Occasionally it is necessary to find all references to a specific dataset in order to perform changes or maintenance. This search can be difficult. This poster describes a SAS program that will build a dataset cross reference list. The list will indicate every production SAS program, JCL, or CLIST that references each dataset.

The sample system demonstrated here consists of one CLIST library, one JCL library, and two SAS libraries.



Two major steps are performed on each library. First, each partitioned library must be converted to a single sequential file. This is easily done with PROC SOURCE for the fixed length 80-column records in our SAS and JCL libraries. Unfortunately, PROC SOURCE will not work with the variable length records in the CLIST library. The IBM utility IEBPTPCH is used to convert the CLIST library to a sequential file. This means that the cross-reference report needs two programs to fulfill its task.

It needs IEBPTPCH and a SAS program to do everything else. After the sequential file is created for each type of data, it is necessary to scan each file for references to datasets. Finally, the results of the three scans are brought together, sorted, and printed.

PROGRAM DETAILS

CLIST - The conversion from a partitioned dataset to a sequential file is done by IEBPTPCH prior to running the SAS program. IEBPTPCH precedes each member output with a record containing the member name. As the SAS program reads this file, it retains the member name while it continues reading the remaining member records. The CLIST scanning routine looks for dataset names among the commands. Each dataset reference generates an observation with three variables (DATABASE, MEMBER, and DSNAME).

JCL PROC SOURCE is used to convert the partitioned dataset into a sequential file. This requires the use of a temporary file. Each member written is preceded by a "./ ADD" record. The sequential file is then scanned for dataset names. Each name found generates an observation with the same three variables (DATABASE, MEMBER, and DSNAME). The name of the JCL library in our sample is RATEMR.CNIL.

SAS Because there are multiple SAS libraries, we utilize a macro to process each one. Our sample uses just two libraries; in reality we have nine. The macro named SOURCE80 requires two arguments: DDNAME and DSNAME. These identify the file reference and actual dataset name of each SAS program library. The sample shows access to two libraries:

RATEMR.AD.SAS (Administrative Library)
RATEMR.CR.SAS (Customer Research Library)

Through the use of DISP=MOD on the allocation of the sequential file, all of the SAS programs are written to a single file. The use of a FIRST statement in PROC SOURCE places a library identification at the beginning of each library's member records. This sequential file is scanned for dataset names which primarily occur in ISO statements. Member identification comes from the "./ ADD" records generated by PROC SOURCE.

REPORT The observations from the various scans are merged, sorted by DSNAME, and finally printed by the DATA _NULL_ step.

```

/*-----*/
/* DSNREF - DOCUMENT DATASETS FOUND IN THE MEMBERS OF THE */
/* PRODUCTION JCL, CLIST, AND SAS LIBRARIES */
/*-----*/

TITLE 'DATASETS REFERENCED IN PRODUCTION LIBRARIES';

/*-----*/
/* FIND DATASETS IN THE CLIST MEMBERS */
/*-----*/

DATA CLIST;
  KEEP MEMBER DSNNAME DATABASE;
  LENGTH MEMBER $ 8 DSNNAME $ 44 DATABASE $ 13 TOKEN $ 60;
  RETAIN MEMBER;
  RETAIN DATABASE 'RATEMR.CLIST';
  INFILE XCLIST;
  INPUT KEYWORDS $ 2-13 RECORD $ 1-124 @;
  IF KEYWORDS='MEMBER NAME' THEN /* MEMBER NAME */
  DO; INPUT MEMBER $ 15-22; DELETE; END;
  IF INDEX(RECORD, '.')=0 THEN DELETE; /* MUST FIND . */
  TOKEN=SCAN(RECORD,1, '.'); /* FIRST TOKEN */
  IF TOKEN='WRITE' THEN DELETE; /* CRT TEXT */
  IF TOKEN='WRITE*' THEN DELETE; /* CRT TEXT */
  IF TOKEN='FIND' THEN DELETE; /* QED SUBCOM */
  IF TOKEN='CHANGE' THEN DELETE; /* QED SUBCOM */
  WORD=1;
  DO WHILE(TOKEN!=' ');
  IF INDEX(TOKEN, '.')=0 THEN GO TO NEXT; /* FIND PERIOD */
  IF INDEX(TOKEN, '&')>0 THEN GO TO NEXT; /* DROP VAR */
  QUOTE=INDEX(TOKEN, '"'); /* FIND QUOTE */
  IF QUOTE=0 THEN GO TO NEXT;
  TOKEN=SUBSTR(TOKEN, QUOTE+1); /* DROP ONE OR MORE QUOTES */
  DO WHILE(TOKEN='"');
  TOKEN=SUBSTR(TOKEN, 2);
  END;
  QUOTE=INDEX(TOKEN, '"'); /* FIND TRAILING QUOTE */
  IF QUOTE=0 THEN GO TO NEXT;
  DSNNAME=SUBSTR(TOKEN, 1, QUOTE-1); /* EXTRACT FINAL DSNNAME */
  OUTPUT CLIST;
NEXT: WORD=WORD+1; /* GET NEXT TOKEN */
  TOKEN=SCAN(RECORD, WORD, '.');
  END;

/*-----*/
/* FIND DATASETS IN THE CNIL MEMBERS */
/*-----*/

PROC SOURCE INDD=CNIL OUTDD=XCNIL NOSUMMARY NOPRINT;

PROC FORMAT; /**** KEYWORDS FOR EXEC JCL STATEMENT ****/
  VALUE SEXECKEY 'ACCT', 'ADDRSPC', 'COND', 'DPRTY', 'DYNAMNBR',
  'PARM', 'PERFORM', 'RD', 'REGION', 'TIME'
  = 'KEYWORD' OTHER = 'NON-KEY';

DATA CNIL;
  KEEP MEMBER DSNNAME DATABASE;
  LENGTH MEMBER $ 8 DSNNAME $ 44 DATABASE $ 13 ;
  LENGTH TOKEN $ 60 PART1 $ 8 ;
  RETAIN DATABASE 'RATEMR.CNIL' SIMT 'J' MEMBER;
  INFILE XCNIL;
  INPUT KEYWORDS $ 1-12 RECORD $ 1-72 @;
  IF KEYWORDS='.' ADD THEN /* MEMBER NAME */
  DO; INPUT MEMBER $ 21-28; DELETE; END;
  IF INDEX(RECORD, '.')=0 THEN DELETE; /* MUST FIND "." */
  IF RECORD='.'; /* KEEP JCL CARDS */
  IF RECORD='.' THEN DELETE; /* DELETE COMMENTS */
  IF SUBSTR(RECORD, 3, 1)=' ' /* SKIP DSNNAME */
  THEN WORD=1; ELSE WORD=2;
  TOKEN=SCAN(RECORD, WORD, ' /,='); /* FIRST TOKEN */
  DO WHILE(TOKEN!=' ');
  IF TOKEN='DD' THEN SIMT='D'; /* SCAN DD SIMT */
  ELSE IF TOKEN='JOB' THEN SIMT='J'; /* DROP JOB SIMT */
  ELSE IF TOKEN='EXEC' THEN SIMT='E'; /* DROP EXEC SIMT */
  IF SIMT = 'J' THEN GO TO NEXT;
  IF INDEX(TOKEN, '.')=0 THEN GO TO NEXT; /* FIND DSNNAME */
  IF INDEX(TOKEN, '&')>0 THEN GO TO NEXT; /* NO BACKWD REF */
  IF SIMT = 'E' THEN /* DROP EXEC KEYWORDS */
  DO;
  PART1=SCAN(TOKEN, 1, '.');
  IF PUT(PART1, SEXECKEY.)='KEYWORD' THEN GO TO NEXT;
  IF INDEX(TOKEN, '(')>0 THEN /* UNBALANCED PARENS */
  IF INDEX(TOKEN, '(')=0 THEN GO TO NEXT;
  END;
  DSNNAME=TOKEN;
  OUTPUT CNIL;
  IF SIMT='D' THEN DELETE; /* 1 DSN PER DD CARD */
NEXT: WORD=WORD+1;
  TOKEN=SCAN(RECORD, WORD, ' /,='); /* GET NEXT TOKEN */
  END;

```

```

/*-----*/
/* CONVERT SAS LIBRARIES TO A SEQUENTIAL FILE */
/*-----*/

%MACRO SOURCE80(DDNAME, DSNNAME);
  PROC SOURCE INDD=&DDNAME OUTDD=XSAS NOSUMMARY NOPRINT;
  FIRST './ LIBRARY=&DSNNAME';
  RUN;
%MEND SOURCE80;

%SOURCE80(AD ,RATEMR.AD.SAS);
%SOURCE80(CR ,RATEMR.CR.SAS);

/*-----*/
/* FIND DATASETS IN THE SAS PROGRAM LIBRARIES */
/*-----*/

DATA SAS;
  KEEP MEMBER DSNNAME DATABASE;
  LENGTH MEMBER $ 8 DSNNAME $ 44 DATABASE $ 13 TOKEN $ 60;
  RETAIN MEMBER;
  RETAIN DATABASE 'DATA';
  INFILE XSAS;
  INPUT KEYWORDS $ 1-12 RECORD $ 1-72 @;

  IF KEYWORDS='.' LIBRARY=' THEN
  DO; INPUT DATABASE $ 13-25; DELETE; END;
  IF KEYWORDS='.' ADD THEN /* MEMBER NAME */
  DO; INPUT MEMBER $ 21-28; DELETE; END;
  TOKEN=SCAN(RECORD,1, '.'); /* FIRST TOKEN */
  IF TOKEN='TSO' OR INDEX(RECORD, ".DATA'")>0; /* TSO COMMAND */
  WORD=1;

  DO WHILE(TOKEN!=' ');
  IF INDEX(TOKEN, '.')=0 THEN GO TO NEXT; /* FIND PERIOD */
  IF INDEX(TOKEN, '&')>0 THEN GO TO NEXT; /* DROP VARIABLE */
  QUOTE=INDEX(TOKEN, '"'); /* FIND QUOTE */
  IF QUOTE=0 THEN GO TO NEXT;
  TOKEN=SUBSTR(TOKEN, QUOTE+1); /* DROP ONE OR MORE QUOTES */
  DO WHILE(TOKEN='"');
  TOKEN=SUBSTR(TOKEN, 2);
  END;
  QUOTE=INDEX(TOKEN, '"'); /* FIND TRAILING QUOTE */
  IF QUOTE=0 THEN GO TO NEXT;
  DSNNAME=SUBSTR(TOKEN, 1, QUOTE-1); /* EXTRACT FINAL DSNNAME */
  OUTPUT;
NEXT: WORD=WORD+1; /* GET NEXT TOKEN */
  TOKEN=SCAN(RECORD, WORD, '.');
  END;

/*-----*/
/* MERGE DSNAMES FROM THE CLIST AND CNIL LIBRARIES */
/*-----*/

DATA MEMBERS;
  SET CLIST CNIL SAS;
  RUN;

PROC SORT DATA=MEMBERS NODUP;
  BY DSNNAME DATABASE MEMBER;
  RUN;

/*-----*/
/* PRINT THE DATASET LOCATION REPORT */
/*-----*/

DATA NULL;
  SET MEMBERS;
  FILE PRINT HEADER=H;
  BY DSNNAME;
  LENGTH LOCATION $ 23; /* LONGEST LIBRARY NAME + 10 */
  COL=26;
  IF FIRST.DSNNAME THEN DO; PUT / DSNNAME $44. @; COL=46; END;
  IF COL>105 THEN DO; PUT ; COL=46; END;
  LOCATION= DATABASE || '(' || TRIM(MEMBER) || ')';
  PUT @COL LOCATION $23. @ ;
  RETURN;

H: PUT / @4 'DATASET NAME'
  @46 '<----->' LIBRARY MEMBERS'
  '<----->';

  IF _N_>1 THEN PUT ;
  RETURN;
  RUN;

/**** END OF PROGRAM - DSNREF ****/

```

APPENDIX B

Sample Report

DATASETS REFERENCED IN PRODUCTION LIBRARIES			
DATASET NAME	LIBRARY MEMBERS		
CINDY.CRDQF86.DATA	RATEMR.CR.SAS(FCDQS86)	RATEMR.CR.SAS(FCDQS00)	RATEMR.CR.SAS(FCDQJ03)
CINDY.CRDQF87.DATA	RATEMR.CR.SAS(FCDQS87)	RATEMR.CNTL (FCDQJ87)	RATEMR.CNTL (FCDQJ99)
CINDY.CRDQF88.DATA	RATEMR.CR.SAS(FCDQS88)	RATEMR.CNTL (FCDQJ88)	
LROTH.CRBUF30.DATA	RATEMR.DATA (CRBUS30)		
LROTH.CRCAT40.DATA	RATEMR.DATA (CRCAS40)		
MACHOSE.FASAG40.DATA	RATEMR.AD.SAS(ADAAS10)		
MACHOSE.FASAW10.DATA	RATEMR.AD.SAS(ADAAS10)		
RATEMR.CR.SAS	RATEMR.CNTL (SAABJ10)	RATEMR.CNTL (SAABJ11)	RATEMR.CNTL (SAABJ12)
RATEMR.CR.SAS(CRAES20)	RATEMR.CNTL (CRAEJ20)		
RATEMR.CR.SAS(CRAES30)	RATEMR.CNTL (CRAEJ30)		
RATEMR.CR.SAS(CRAES40)	RATEMR.CNTL (CRAEJ40)		
RATEMR.CRZZF10.DATA	RATEMR.CNTL (CRAJJ20)	RATEMR.CLIST (CRAJCL0)	RATEMR.CLIST (CRAJCL0)
	RATEMR.CR.SAS(CRZZS10)	RATEMR.CR.SAS(CRZZS11)	RATEMR.CR.SAS(CRZZS12)
	RATEMR.CR.SAS(CRZZS13)	RATEMR.CR.SAS(CRZZS14)	RATEMR.CR.SAS(CRZZS15)
	RATEMR.CR.SAS(CRZZS16)	RATEMR.CR.SAS(CRZZS17)	RATEMR.CR.SAS(CRZZS18)
RATEMR.FMLIB	RATEMR.CNTL (CRAEJ20)	RATEMR.CNTL (SAABJ10)	
RATEMR.FORMAT	RATEMR.CNTL (SAABJ10)		

The report above indicates that references to a permanent SAS format library named RATEMR.FMLIB can be found in production JCL members RATEMR.CNTL(CRAEJ20) and RATEMR.CNTL(SAABJ10).

APPENDIX C

JCL to run Report

```
//MACHOSE1 JOB 999999, 'WM MACHOSE PP&L'
//*
//*****
//*          CONVERT CLIST LIBRARY
//*****
//XCLIST EXEC PGM=IEBPTCH
//SYSUT1 DD DSN=RATEMR.CLIST,DISP=SHR
//SYSUT2 DD DSN=*&XCLIST,DISP=(NEW,PASS),
//          DCB=(LRECL=133,BLKSIZE=8911,
//          RECFM=FBA),UNIT=SYSDA,
//          SPACE=(8911,(200,100))
//SYSIN DD *
PRINT TYPORG=PO,MAXFLDS=1
RECORD FIELD=(124,9,,1)
//SYSPRINT DD SYSOUT=A
//*
//*****
//*          SCAN PRODUCTION LIBRARIES
//*****
//DSNXREF EXEC SAS
//SYSIN DD DSN=RATEMR.RMAAS10.DATA,DISP=SHR
//CNTL DD DISP=SHR,DSN=RATEMR.CNTL
//AD DD DISP=SHR,DSN=RATEMR.AD.SAS
//CR DD DISP=SHR,DSN=RATEMR.CR.SAS
//XCLIST DD DSN=*&XCLIST,DISP=(OLD,DELETE)
//XCNTL DD DISP=(NEW,DELETE),UNIT=SYSDA,
//          DCB=(LRECL=80,BLKSIZE=9040,
//          RECFM=FB),SPACE=(9040,(100,100))
//XSAS DD DISP=(MOD,DELETE),UNIT=SYSDA,
//          DCB=(LRECL=80,BLKSIZE=9040,
//          RECFM=FB),SPACE=(9040,(500,500))
```

SAS is the registered trademark of SAS Institute Inc., Cary NC

The author can be contacted at:

Pennsylvania Power & Light Co.
 2 North Ninth Street TW-5
 Allentown, PA 18101
 (215) 770-5278