PDS - The Swiss Army Knife of Utilities

Session 2837

August 14, 2006

Thomas Conley

Pinnacle Consulting Group, Inc. (PCG)

59 Applewood Drive

Rochester, NY 14612-3501

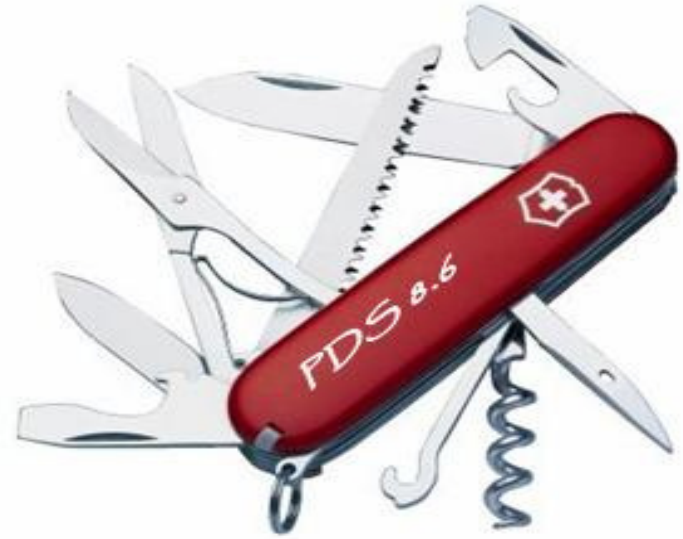
P: (585)720-0012

F: (585)723-3713

pinncons@rochester.rr.com

http://home.rochester.rr.com/pinncons

Wish I Had One!



Legal & Acknowledgement

- Swiss Army™ is the registered trademark of Victorinox AG and subsidiaries, and is used here with their kind permission
- Thanks to John Kalinich at Computer Sciences Corporation; this presentation is largely based on material he prepared

Agenda

- History
- Installing PDS
- Learning to Use PDS
- PDSE Support
- Member Groups
- Getting Started
- PDS Subcommands

Agenda

- Miscellaneous Commands
- Summary
- Finally...

History

- PDS was designed to manipulate partitioned data sets (PDS's)
- Originally written in 1972 at Fireman's Fund Insurance by Tom Springer
- From 1977 to 1990, Bruce Leland and Steve Smith made extensive modifications to PDS
- In 1997, John Kalinich contributed modifications to the PDS command for Year 2000

History

- In 2005, Greg Price and John Kalinich added PDSE support
- The PDS command code base has grown from 3,500 to over 50,000 lines of code
- John Kalinich currently supports PDS; send him an Email at jkalinic@csc.com with your comments, concerns, suggestions, enhancements, etc. for PDS

Installing PDS

- Download File 182 from http://www.cbttape.org to obtain the PDS to install PDS
- Binary FTP or upload to MVS bin file with attributes (DSORG=PS,LRECL=80).
- Issue TSO RECEIVE INDATASET(bin) and respond to prompts to create install PDS
- Review and follow instructions in the $$$$INST member.

Installing PDS

- Edit member #PDSGEN (and #PDSGENB if required) to set PDS global assembler variables according to instructions in each member
- An alternative to updating #PDSGEN is to uncomment the COPY statement for #PDSGENU (at the end of #PDSGEN) and make your changes in that user member (this is the current default)

Installing PDS

- Run job PDSJCLHL to assemble and link edit the PDS load module into a PDS load library
- Run job PDSPJCL to copy the ISPF panels and message members
- Copy member PDSHELP to a SYSHELP dataset. Rename it to PDS86 with aliases of PDS and PDSE

Installing PDS

- To install the PDS related command processors, issue RECEIVE INDS(install.pds(UTILXMIT)) and respond to prompts to create the following load modules in the PDS load library:

  BLKDISK, COMPARE*, DELINKI, DISASM, DSAT, DVOL, HEL, OFFLOAD, RELEASE, REVIEW and VTOC

- While not required, the above commands offer significant extended functions for PDS

Installing PDS

- Tailor PDS LIBDEF CLIST in PDSLIBDF member
- Add Option P to an ISPF selection panel:

```
)BODY
P   PDS Command - Invoke PDS

)PROC
P,'CMD(%PDSLIBDF) SCRNAME(PDS)'
```

- Can run as TSO command processor if placed in TSOLIB/STEPLIB/LINKLIST (e.g. TSO %PDS)

Learning to Use PDS

- Read “Effective use of the PDS command” (member $$$PROSE on the install PDS) by Bruce Leland
- Read “Public MVS Software - The PDS Program” by Sam Golob (members $PDSART1-3)
- Read the TSO Help member for PDS (member PDSHELP)
- This presentation (member $$S2837)

PDSE Support

- PDSE's have been difficult to support due to IBM's proprietary IGWFAMS interface
- In 2005, Greg Price and John Kalinich added PDSE support using standard access methods
- PDS uses QSAM to read PDSE directories and BPAM to read/write PDSE members
- Apply OA13747 to fix S0C4's in PDSE dataspace (thanks to Greg Price for this APAR)

PDSE Support

- Apply OA13224 to allow DISP=SHR for STOW INITIALIZE (thanks to John Kalinich for this APAR)
- STOW INITIALIZE will reset a PDS directory, but it requires exclusive access for a PDS
- PDSE provides its own serialization, so this APAR relaxes the exclusive requirement for PDSE datasets

PDSE Support

- The following subcommands are not supported for PDSE data sets:

Subcommand	PDSE Type	Reason
RESTORE	Data	IBM design
RESTORE	Program Object	IBM design
REPLACE	Program Object	IBM design
ALIAS	Program Object	Binder API
ATTRIB lkedattr	Program Object	Binder API
REPRO	Program Object	Binder API

Member Groups

- Many PDS commands allow lists or groups of members to be specified
- You can specify ranges of members, matching patterns, discrete member names, or any combination of the above

Member Groups

- ":" (colon) represents all members
- memlist :

```
- DSN=SYS2.PDS.INSTALL,VOL=SER=OS39M1   MEM=:   --------------------------------
CMD   NAME      DATA      VER.MOD     CREATED    LAST MODIFIED   SIZE   INIT     ID
      $$$$$$$$            01.00     97/11/24   97/11/24  7:38      6      6    TRIDJK
      $$$$READ
      $$$$REA1            ALIAS
      $$$BUGS             01.05     03/04/08   03/08/04 12:09    531    530    TRIDJK
      $$$CTBL             01.10     92/01/29   05/08/10  8:34    174    174    TRIDJK
      $$$CUSTM            01.40     98/03/16   05/12/28 10:45    384    336    TRIDJK
      $$$CUT              01.03     98/01/27   98/06/17  8:02    105    100    TRIDJK
```

Member Groups

- "*" (asterisk) represents the current sublist
- sublist a*;memlist *

```
- DSN=SYS2.PDS.INSTALL,VOL=SER=OS39M1  MEM=ABEHELP  --------------------------
CMD  NAME     DATA     VER.MOD    CREATED   LAST MODIFIED  SIZE  INIT    ID
     ABEHELP
```

Member Groups

- “mem1:mem2” defines a range starting with mem1, ending with names starting with mem2
- memlist @al:@cl reset

```
- DSN=SYS2.PDS.INSTALL,VOL=SER=OS39M1  MEM=@AL:@CL  --------------------------
CMD  NAME     DATA     VER.MOD    CREATED   LAST MODIFIED  SIZE  INIT    ID
     @ALIAS             01.19    99/07/16  06/01/03 15:20   169   158   TRIDJK
     @ATTRIB            01.79    98/01/15  06/01/04 13:58  1579  1413   TRIDJK
     @BROWSE            01.06    99/07/16  06/01/11  8:14   520   515   TRIDJK
     @CHANGE            01.00    99/07/16  99/07/16  9:34    35    35   TRIDJK
     @CLEAR             01.00    99/07/16  99/07/16  9:34    45    45   TRIDJK
```

Member Groups

- "mem1/mem2" defines a pattern where mem1 and mem2 are anywhere in the member name
- memlist @/pds reset

```
- DSN=SYS2.PDS.INSTALL,VOL=SER=OS39M1  MEM=@/PDS  ----------------------------
CMD  NAME     DATA     VER.MOD    CREATED   LAST MODIFIED  SIZE  INIT    ID
     @FIXPDS           01.34    99/07/16  06/02/22 14:14  1000   906   TRIDJK
     @PDSMAIN          01.99    98/10/13  06/02/22 14:32  5378  4869   TRIDJK
     @PDSMODS          01.99    02/12/18  06/02/22 14:22   212    35   TRIDJK
     PDS@PRIM          01.24    00/03/20  05/09/30  7:38   261   223   TRIDJK
```

Member Groups

- “mem1*mem2” defines a pattern where mem1 precedes mem2 in the member name
- memlist @*pds reset

```
- DSN=SYS2.PDS.INSTALL,VOL=SER=OS39M1  MEM=@*PDS  ----------------------------
CMD  NAME     DATA     VER.MOD    CREATED   LAST MODIFIED  SIZE  INIT    ID
     @FIXPDS           01.34    99/07/16  06/02/22 14:14  1000   906   TRIDJK
     @PDSMAIN          01.99    98/10/13  06/02/22 14:32  5378  4869   TRIDJK
     @PDSMODS          01.99    02/12/18  06/02/22 14:22   212    35   TRIDJK
```

Member Groups

- Combinations can also be specified
- memlist (a:b, tso/, pdshelp) reset

```
- DSN=SYS2.PDS.INSTALL,VOL=SER=OS39M1  MEM=(A:B  ------------------------------
CMD  NAME     DATA     VER.MOD    CREATED   LAST MODIFIED  SIZE  INIT    ID
     @TSO              01.00    99/07/16  99/07/16  9:34   175   175   TRIDJK
     @TSOEDIT          01.00    99/07/16  99/07/16  9:34   156   156   TRIDJK
     ABEHELP
     PDSHELP           01.99    97/09/25  06/01/13 11:59  6359  5972   TRIDJK
     P85VTSO
     P86VTSO
     VTSOPCF
     VTSORACF          01.00    05/11/17  05/11/17  9:51   234   234   TRIDJK
```

Getting Started

- Execute the PDSLIBDF CLIST to start PDS

```
  RefList  RefMode  Options  Help
--------------------------------------------------------------------------------
                           PDS Version 8.6
Option ===>

    I - Enter ISPMODE
    M - Enter MEMLIST with the identified MEMBERS
    L - Enter Line mode
  SET - Set default options prompt
blank - Enter MEMLIST if any member data is entered; ISPMODE otherwise

ISPF Library:
   Project ===> TCONLEY
   Library ===> TSO
   Type    ===> JCL
   Members ===>                    (set to * or a member group to use MEMLIST)

Other Partitioned or Sequential Data Set or FILE(ddname):
   Data Set Name  ===> sys2.pds.install
   Volume Serial  ===>          (If not cataloged)
   Volume Set     ===>          (For a default volume name)
   MEMLIST Prompt ===> NO       (yes/no for a MEMLIST prompt panel)
   PDS PGM Name   ===> PDS86    (latest: PDS86)
```

Getting Started

- PDS log is displayed

```
 ------------------------------------------------- ISPMODE Session Display
 COMMAND ===>
Enter an ISPF command, a PDS subcommand or a special control code:
   LA Lista      1 Suspend ISPF     9 Swap panels    R Recall
   LV Listv      6 MEMLIST all      F Find in table  X Above/Below/All
   ML Memlist    7 Output log       O Options        * Memlist *
 - DSN=SYS2.PDS.INSTALL,VOL=SER=OS39M1  MEM=  --------------------------------
 PDS100I PDS86 -- VERSION 8.6.02  FEBRUARY 22, 2006

 PDS200I DISP UNIT OPT RECFM LRECL BLKSIZE   ALLOCTRK FREETRK SECONDARY FREEDIR
 PDS200I SHR  3390 C   FB       80   27920   5X   401     167    50 TRK      43

 PDS300A ENTER OPTION -- DSN=SYS2.PDS.INSTALL,VOL=SER=OS39M1  MEM=
```

Getting Started

- Enter “6” for MEMLIST all

```
 ------------------------- MEMLIST Source Member List 1 -- Row 1 to 16 of 1,011
 COMMAND ===>                                                  SCROLL ===> CSR
Enter an ISPF command, a PDS subcommand or a special control code:
   8 View log    1 Suspend ISPF    6 MEMLIST all    F Find      R Recall
  LA Lista       4 Sublist =       7 Output table   L Locate    SO Sort
  LV Listv       5 Check aliases   9 Swap panels    O Options   X Above/Below/Al
 - DSN=SYS2.PDS.INSTALL,VOL=SER=OS39M1  MEM=:  -------------------------------
 CMD  NAME      DATA     VER.MOD    CREATED   LAST MODIFIED  SIZE  INIT    ID
      $$$$$$$$           01.00    97/11/24  97/11/24  7:38     6     6   TRIDJK
      $$$$READ
      $$$$REA1           ALIAS
      $$$BUGS            01.05    03/04/08  03/08/04 12:09   531   530   TRIDJK
      . . .
```

PDS Subcommands - ATTRIB

- ATTRIB displays and modifies load module attributes or member statistics
- Attributes for load modules:

```
AMODE24/AMODE31/AMODE64/AMODEANY
AUTH/NOAUTH  or  APF/NOAPF
DC/NODC                        EDIT/NOEDIT
ENTRY(entryname)               EXEC/NOEXEC
LOADONLY/NOLOADONLY            PAGE/NOPAGE
REFR/NOREF                     RENT/NORENT
REUS/NOREUS                    RLDFIX/NORLDFIX
RMODE24/RMODEANY               SSI(hexdata)/NOSSI
UNALIAS                        NONE
ALIASINFO/NOALIASINFO          LKEDDATE/NOLKEDDATE
```

PDS Subcommands - ATTRIB

- Attributes for regular PDS members:

```
ADDSTATS      - create ISPF statistics if none are present
VER(num)      - reset the ISPF version number
MOD(num)      - reset the ISPF modification level
ID(userid)    - reset the ISPF userid
CREATED(mm/dd/yy) - reset the ISPF creation date
LAST(mm/dd/yy)    - reset the ISPF last modification date
TIME(hhmm)        - reset the ISPF last modification time
SECS(ss)          - reset the ISPF last modification time
UNALIAS       - remove any alias attribute
SSI(hexval)   - assign the hexadecimal SSI information
NOSSI         - remove any existing SSI information
NONE          - remove any existing SSI or ISPF attribute information
ALIASINFO     - provide alias name information
NOALIASINFO   - do not provide alias name information
```

PDS Subcommands - ATTRIB

- attrib pds86

```
- DSN=SYS2.PDS.LOAD,VOL=SER=OS39M2  MEM=PDS86  --------------------------------
>attrib pds86

PDS020I PDS86 ATTRIBUTES ARE: RENT, REUS, REFR
PDS024I PAGE ALIGNMENT IS REQUIRED
PDS102I ENTRY POINT AT 000000
PDS104I MODULE LENGTH  040C10  --    260K
PDS064I LAST LINK-EDITED ON  3/04/06  BY LKED 5695PMB01  V01 M03  AT  15:18:54
PDS160I ALIASES FOR THIS MEMBER ARE: PDS
```

PDS Subcommands - ATTRIB

- attrib pds86 norent noreus norefr

```
- DSN=SYS2.PDS.LOAD,VOL=SER=OS39M2  MEM=PDS86
>attrib pds86 norent

PDS020I PDS86 ATTRIBUTES ARE: NONE
PDS024I PAGE ALIGNMENT IS REQUIRED
PDS102I ENTRY POINT AT 000000
PDS104I MODULE LENGTH  040C10  --    260K
PDS064I LAST LINK-EDITED ON  3/04/06  BY LKED 5695PMB01  V01 M03  AT  15:18:54
PDS160I ALIASES FOR THIS MEMBER ARE: PDS
```

PDS Subcommands - ATTRIB

- attrib pdsjclhl

```
- DSN=SYS2.PDS.INSTALL,VOL=SER=OS39M1  MEM=PDSJCLHL  ----------------------------
>attrib pdsjclhl
PDS230I MEMBER      VER.MOD  CREATED   LAST MODIFIED  SIZE  INIT   MOD   ID
PDS230I  PDSJCLHL    01.99  92/01/10  05/08/27  9:02    54    42    41  TRIDJK
```

- attrib pdsjclhl ver(2) mod(02) id(tconley)

```
- DSN=SYS2.PDS.INSTALL,VOL=SER=OS39M1  MEM=PDSJCLHL  ----------------------------
>attrib pdsjclhl
PDS230I MEMBER      VER.MOD  CREATED   LAST MODIFIED  SIZE  INIT   MOD   ID
PDS230I  PDSJCLHL    02.02  92/01/10  05/08/27  9:02    54    42    41  TCONLEY
```

PDS Subcommands - ALIAS

- ALIAS can assign an ALIAS name to a main member in a PDS
- alias $$$$read $$$$rea1

```
- DSN=SYS2.PDS.INSTALL,VOL=SER=OS39M1  MEM=$$$$REA*  -------------------------
CMD  NAME     DATA     VER.MOD    CREATED   LAST MODIFIED  SIZE  INIT    ID
     $$$$READ
     $$$$REA1          ALIAS
```

PDS Subcommands - BROWSE

- BROWSE opens a PDS member in ISPF BROWSE mode
- browse $$$$read

```
         Menu  Utilities  Compilers  Help
 ------------------------------------------------------------------------------
 BROWSE    SYS2.PDS.INSTALL($$$$READ)                 Line 00000000 Col 001 080
 Command ===>                                           Scroll ===> PAGE
******************************** Top of Data ********************************
    The adventure starts here.   Armed with only your imagination and
    the source to PDS you are ready to do battle within the labyrinth
    of the IBM partitioned data set structure.  In the past, the PDS
```

PDS Subcommands - CHANGE

- CHANGE changes the dataset that PDS is working on
- c ‘sys2.pds.load’

```
- DSN=SYS2.PDS.LOAD,VOL=SER=OS39M2  MEM=  ------------------------------------
>c sys2.pds.load
PDS200I DISP UNIT     RECFM LRECL BLKSIZE   ALLOCTRK FREETRK SECONDARY FREEDIR
PDS200I SHR  3390     U         0   32760   2X    45       8     1 CYL      38

PDS300A ENTER OPTION -- DSN=SYS2.PDS.LOAD,VOL=SER=OS39M2  MEM=
```

PDS Subcommands - COMPARE

- COMPARE compares two members using SUPERC
- compare adyset01 adyset02

```
- DSN=ADCD.ZOSV14W.PARMLIB,VOL=SER=W4RES1  MEM=ADYSET02  ------------------------------------------------------------------
>compare adyset01 adyset02
1  ISRSUPC    -   MVS/PDF FILE/LINE/WORD/BYTE/SFOR COMPARE UTILITY- ISPF FOR z/OS         2006/03/08   2.00    PAGE     1
 NEW: ADCD.ZOSV14W.PARMLIB(ADYSET02)                          OLD: ADCD.ZOSV14W.PARMLIB(ADYSET01)

                     LISTING OUTPUT SECTION (LINE COMPARE)

 ID       SOURCE LINES                                                          TYPE  LEN N-LN# O-LN#
     ----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
                                                                          MAT=    1
 I - *    START PARAMETERS.                                          00100000 RPL=    1 00002 00002
 D - *    STOP PARAMETER.                                            00100000
```

PDS Subcommands - COMPRESS

- COMPRESS performs an IEBCOPY compress
- compress shr (undocumented option “SHR” bypasses ENQ, good for hosing LINKLIST)

```
- DSN=SYS2.PDS.INSTALL,VOL=SER=OS39M1  MEM=: ------------------------------
>compress shr
PDS480W COMPRESS IS IN PROGRESS
IEB1135I IEBCOPY  FMID HDZ11G0  SERVICE LEVEL UA05363  DATED 20030923 DFSMS
01.03.00 z/OS    01.04.00 HBB7707  CPU 7490
IEB1035I TCONLEY  DYNISPF  DYNISPF  00:18:50 SUN 05 MAR 2006 PARM=''
DYNISPF  COPY      INDD=SYS00023,OUTDD=SYS00023     GENERATED STATEMENT
IEB1018I COMPRESSING  PDS  OUTDD=SYS00023 VOL=OS39M1 DSN=SYS2.PDS.INSTALL
IEB153I ALL MEMBERS COMPRESSED - ALL WERE ORIGINALLY COMPRESSED
IEB144I THERE ARE 167 UNUSED TRACKS IN OUTPUT DATA SET REFERENCED BY SYS00023
IEB147I END OF JOB - 0 WAS HIGHEST SEVERITY CODE
```

PDS Subcommands - CONTROL

- CONTROL specifies PDS processing options, like sysout class, prompting, etc.
- control noprompt (turns off prompting for PDS)

```
- DSN=ADCD.ZOSV14W.PARMLIB,VOL=SER=W4RES1  MEM=ADYSET02  ------------------
>control noprompt

PDS100I PDS86 -- VERSION 8.6.02  FEBRUARY 22, 2006
PDS106I DATE FORMAT: USA (MM/DD/YY)

PDS030I GLOBAL OPERANDS: NOPROMPT, ALIASINFO, LKEDDATE, RECOVER, TRANSLATOR
PDS030I GLOBAL OPERANDS: NODSNAME, NOSYSOUT, NOFORM, NODEST
PDS031I INPUT BUFFERING: MULTIPLE
PDS036I LARGEST FREE STORAGE AREA IS 7636K
```

PDS Subcommands - CONTROL

- control mods (shows PDS enhancements)

```
- DSN=TCONLEY.TSO.JCL,VOL=SER=XS39M1   MEM=   -------------
>control mods
. . .

PDS070I MODIFICATION HISTORY:
 ---------------------------------------------------------------
                      PDS v8.6
                Update 02, February 22, 2006
 ---------------------------------------------------------------
 New global variable STOWI to specify if APAR OA13224
 "DISP=SHR for STOW Initialize" has been applied.
 ---------------------------------------------------------------
```

PDS Subcommands - COPY

- COPY copies member(s) to another dataset
- Copy ady* tconley.tso.jcl

```
- DSN=ADCD.ZOSV14W.PARMLIB,VOL=SER=W4RES1  MEM=ADY*  ----------------------------------------------------------------
>copy ady* tconley.tso.jcl
PDS484W COPY IS IN PROGRESS
IEB1135I IEBCOPY  FMID HDZ11G0  SERVICE LEVEL UA05363  DATED 20030923 DFSMS 01.03.00 z/OS    01.04.00 HBB7707  CPU 7490
IEB1035I TCONLEY  ISPFPROC ISPFPROC 02:11:56 WED 08 MAR 2006 PARM=''
 COPY    O=SYS00019,I=SYS00006
 S M=ADYSET00,ADYSET01,ADYSET02
IEB1013I COPYING FROM PDS   INDD=SYS00006 VOL=W4RES1 DSN=ADCD.ZOSV14W.PARMLIB
IEB1014I           TO PDS  OUTDD=SYS00019 VOL=XS39M1 DSN=TCONLEY.TSO.JCL
IEB167I FOLLOWING MEMBER(S) COPIED FROM INPUT DATA SET REFERENCED BY SYS00006
IEB154I ADYSET00 HAS BEEN SUCCESSFULLY COPIED
IEB154I ADYSET01 HAS BEEN SUCCESSFULLY COPIED
IEB154I ADYSET02 HAS BEEN SUCCESSFULLY COPIED
IEB1098I 3 OF 3 MEMBERS COPIED FROM INPUT DATA SET REFERENCED BY SYS00006
IEB144I THERE ARE 11 UNUSED TRACKS IN OUTPUT DATA SET REFERENCED BY SYS00019
IEB149I THERE ARE 23 UNUSED DIRECTORY BLOCKS IN OUTPUT DIRECTORY
IEB147I END OF JOB - 0 WAS HIGHEST SEVERITY CODE
```

PDS Subcommands - DCF

- DCF prints a hardcopy using TSO SCRIPT

```
------------------------- O.DCF   SCRIPT Subcommand -------------------------
OPTION  ===>

Enter the member group name and any operands below for DCF:
===>

Operands:  memgroup      (e.g., *; start:end; start:; first*pat; range*; part/)
 BIND(obind {ebind})                  PAGE{({PROMPT}{{FROM} p {TO} q}
 CHARS(font1..font4)                        {{FROM} p FOR n}{{FROM} p ONLY})}
 CONTINUE/NOCONT                      PRINT{(copies,class,fcb,ucs)}
 CTF                                  PROFILE{(fileid)}/NOPROFILE
 DDUT/NODDUT                          QUIET
 DEST(station-id)                     SEARCH(libname)(opnum...)
 DEVICE(devtype)                      SEGLIB/NOSEGLIB
 FILE{(fileid)}                       SPELLCHK
 FONTLIB({filetype}{filemodel})       STOP
 FPASSES n                            SYON/SYOFF
 INDEX                                SYSVAR(n value...)
 LIB(libename...)(opnum...)           TERM
 MESSAGE({DELAY}{D}{TRACE})           TLIB
 NOSPIE                               TWOPASS
 NOWAIT                               UNFORMAT
 NUMBER                               UPCASE
 OPTIONS{(fileid)}                    @user-option
```

PDS Subcommands - DELETE

- DELETE deletes members from a PDS
- del tom*

```
- DSN=TCONLEY.TSO.JCL,VOL=SER=XS39M1   MEM=TOM*   --------------------
>del tom*
PDS162I MEMBERS TO BE DELETED ARE: TOMNEW, TOMOLD

PDS394A SHOULD ALL OF THESE MEMBERS BE DELETED (Y/N) ?
>y
PDS040I TOMOLD HAS BEEN DELETED
PDS040I TOMNEW HAS BEEN DELETED
```

PDS Subcommands - DIRENTRY

- dir pds (for a load module)

```
- DSN=SYS2.PDS.LOAD,VOL=SER=OS39M2  MEM=PDS  ------------------------------
>dir pds
PDS143I PDS DIRECTORY ENTRY, LENGTH=46
         0000  D7C4E240 40404040  001309B1 00140200  *PDS     ........*
         0010  00000000 C2E3040C  10793800 0000A800  *....BT...`....y.*
         0020  02000000 D7C4E2F8  F6404040 0100      *....PDS86   ..*
PDS262I LOC NAME     VALUE     DESCRIPTION
PDS262I --- ----     -----     -----------
PDS262I 00  PDS2NAME PDS       MEMBER NAME
PDS262I 08  PDS2TTRP 001309    TTR OF FIRST BLOCK OF DATA
PDS262I 0B  PDS2INDC B1        ALIAS; 1 TTRS FOLLOW; 17 HALFWORDS OF DATA
PDS262I 0C  PDS2TTRT 001402,00 TTR OF FIRST TEXT BLOCK
```

PDS Subcommands - DIRENTRY

```
PDS262I 10  PDS2TTRN 000000,00 (NOT USED FOR THIS MEMBER)
PDS262I 14  PDS2ATR1 C2        REENTRANT; REUS; NOT OVERLAY; NOT TEST
PDS262I                        NOT ONLY LOAD; NOT SCATTER; EXEC; NOT 1 TEXT
PDS262I 15  PDS2ATR2 E3        NOT DC; TEXT ORG=0; EP=0; HAS RLDS
PDS262I                        EDIT; NOT TEST; LKED F; REFRESHABLE
PDS262I 16  PDS2STOR 260K      TOTAL CONTIGUOUS MAIN STORAGE REQUIRED
PDS262I 19  PDS2FTBL 31,032.   LENGTH OF FIRST BLOCK OF TEXT
PDS262I 1B  PDS2EPA  000000    ENTRY POINT ADDRESS
PDS262I 1E  PDS2FTB1 A8        PROCESSED BY OS/VS LINKAGE EDITOR
PDS262I                        PAGE ALIGNMENT REQUIRED
PDS262I                        APF INFORMATION IS VALID
PDS262I 1F  PDS2FTB2 00        RMODE 24;  ALIAS AMODE 24;  MAIN AMODE 24
PDS262I 20  PDS2FTB3 02        RLD/CONTROL RECORDS AFTER FIRST TEXT BLOCK
PDS262I 21  PDS2EPM  000000    ENTRY POINT OF MAIN MEMBER
PDS262I 24  PDS2MNM  PDS86     MEMBER NAME OF MAIN MEMBER
PDS262I 2C  PDSAPFCT 01        LENGTH OF PROGRAM AUTHORIZATION CODE
PDS262I 2D  PDSAPFAC 00        PROGRAM AUTHORIZATION CODE
```

PDS Subcommands - DIRENTRY

- New for PDS 8.6.04, Large Program Object (LPO) support

```
22  PDSSSIWD 01117795  SSI INFORMATION
26  PDS2LPOL 0D        LENGTH OF LARGE PROGRAM OBJECT SECTION
27  PDS2VSTR 02686040  VIRTUAL STORAGE REQUIREMENT FOR THIS MODULE
2B  PDS2MEPA 00000000  MAIN ENTRY POINT OFFSET
2F  PDS2AEPA 00000000  ALIAS ENTRY POINT OFFSET
```

PDS Subcommands - DIRENTRY

- dir pds (for a source member)

```
- DSN=SYS2.PDS.INSTALL,VOL=SER=OS39M1  MEM=:  ---------------------------
>DIRENTRY $$MXIBAT
PDS143I $$MXIBAT DIRECTORY ENTRY, LENGTH=42
         0000  5B5BD4E7 C9C2C1E3  00A6030F 01090048  *$$MXIBAT.w......*
         0010  0105334F 0105343F  13310009 00070000  *...|............*
         0020  E3D9C9C4 D1D24040  4040              *TRIDJK    *

PDS262I LOC NAME     VALUE     DESCRIPTION
PDS262I --- ----     -----     -----------
PDS262I 00  PDS2NAME $$MXIBAT  MEMBER NAME
PDS262I 08  PDS2TTRP 00A603    TTR OF FIRST BLOCK OF DATA
```

PDS Subcommands - DIRENTRY

```
PDS262I 0B  PDS2INDC 0F        0 TTRS FOLLOW; 15 HALFWORDS OF DATA
PDS262I 0C  DIRSPFV  1.        MEMBER VERSION NUMBER
PDS262I 0D  DIRSPFR  9.        MEMBER REVISION NUMBER
PDS262I 0E  DIRSPFSC 00        SCLM INDICATOR   -- X'80'
PDS262I 0F  DIRSPFCS 48        LAST CHANGE TIME -- FORMAT: SS
PDS262I 10  DIRSPFCR 0105334F  CREATION DATE    -- FORMAT: 0CYYDDDF
PDS262I 14  DIRSPFCD 0105343F  LAST CHANGE DATE -- FORMAT: 0CYYDDDF
PDS262I 18  DIRSPFCT 1331      LAST CHANGE TIME -- FORMAT: HHMM
PDS262I 1A  DIRSPFSI 9.        NUMBER OF LINES CURRENTLY
PDS262I 1C  DIRSPFIN 7.        NUMBER OF LINES INITIALLY
PDS262I 1E  DIRSPFMD 0.        NUMBER OF MODIFIED LINES
PDS262I 20  DIRSPFID TRIDJK    USERID OF LAST PERSON TO UPDATE
```

PDS Subcommands - DISPLAY

- DISPLAY creates a list of members matching a start/end pattern
- DISPLAY is no longer recommended, use MEMBERS instead

PDS Subcommands - DSNAME

- DSNAME displays current dataset stats in format you wish (TSO, JCL, or MSG)
- dsname tso

```
- DSN=SYS2.PDS.INSTALL,VOL=SER=OS39M1  MEM=:  -------------------------------
>dsname tso
PDS210I ALLOC F(SYS00022) DA('SYS2.PDS.INSTALL') SHR UNIT(3390) -
PDS210I   RECFM(F B) LRECL(80) BLKSIZE(27920) OPTCD(C) VOLUME(OS39M1) -
PDS210I   TRA SPACE(401,50) DIR(180)            /*FREE TRK=167,FREE DIR=43*/
```

PDS Subcommands - DSPRINT

- DSPRINT prints a hardcopy listing of a member

```
------------------------- O.DSP    DSPRINT Subcommand -------------------------
OPTION  ===>

Enter the member group name, the printer name and operands for DSPRINT:
===>

Operands:  memgroup      (e.g., *; start:end; start:; first*pat; range*; part/)
           printername
           DDNAME(ddname)  DIRECT(KEEP/DELETE)
           LINES(linenum1:linenum2)
           NUM(loc,length)/SNUM(loc,length)/NONUM
           FCB(fcbname)  FORMS(formname)
           CLASS(c)  HOLD/NOHOLD  COL(col1:col2, ...)  COPIES(number)
           PAGELEN(num)  TMARGIN(num)  BMARGIN(num)
           SINGLE/DOUBLE/CCHAR
           FOLD(width)/TRUNCATE(width)
           EJECT/NOEJECT  NOEJMESS  NOHEADER
           TERMINAL  TRACE  WTR(wtrname)

Required:  printername
Defaults:  NUM, CLASS(A), COPIES(1), NOHOLD, SINGLE, HEADER, NOEJECT, MEMBERS

Note:      The above parameters are actually for the VPSPRINT product.
```

PDS Subcommands - EDIT

- EDIT opens a PDS member in ISPF EDIT mode
- edit $$$$read

```
  File  Edit  Edit_Settings  Menu  Utilities  Compilers  Test  Help
------------------------------------------------------------------------------
EDIT       SYS2.PDS.INSTALL($$$$READ) - 01.00                 Columns 00001 00080
Command ===>                                                    Scroll ===> CSR
****** ******************************** Top of Data ********************************
==MSG> -Warning- The UNDO command is not available until you change
==MSG>           your edit profile using the command RECOVERY ON.
000001     The adventure starts here.   Armed with only your imagination and
000002     the source to PDS you are ready to do battle within the labyrinth
000003     of the IBM partitioned data set structure.  In the past, the PDS
```

PDS Subcommands - FIND

- FIND displays portions of a member which contain a search string
- FIND has a myriad of options, including, but not limited to, the format of the FIND, how many records you want to skip before the FIND, or a THEN/ELSE clause to determine what to do with a member if the string is found or not found within it

PDS Subcommands - FIND

- find : /fixpds/ then(sublist)

```
>mem *
PDS165I MEMBERS ARE: $$$BUGS, $$$CUSTM, $$$MODS, $$$PDSE, $$$PROSE, $$$SECUR,
         $$$UNDOC, $$PDSMAN, $PDSARTH, $PDSART1, $PDSART2, $PDSART3, #MDSECT,
         #PDSGEN, #PDSMODS, #PDSTBL, @DIAFUN, @DIALOG, @DIATBL, @FIXPDS,
         @GLOBAL, @LIST, @PARSE, @PDSMAIN, @PDSMODS, @REPRO, @SUBS, @VERIFY,
         NOTES, PDSHELP, P86ART1, P86ART2B, P86ART3, P86CUST, P86EXPAN,
         P86LOHE, P86OPFX, P86OPHE, P86OPUX, P86OSFX, P86PN5E, P86RLSE,
         P86UNDOC, SMPFIXES, SMPINST, VSUBCMD

PDS193I THIS GROUP CONTAINS 46 MEMBERS
```

PDS Subcommands - FIXPDS

- FIXPDS can perform several utility functions for a PDS to repair or change it
- Expand and free directory blocks
- Modify DCB attributes
- Reset the directory (empty the PDS)
- Release or add space (extents)

PDS Subcommands - FIXPDS

- fixpds expanddir(45)

```
- DSN=TCONLEY.TSO.JCL,VOL=SER=XS39M1  MEM=PDSHELP  --------------------------
>fixpds expanddir(1)
PDS050I DEFDSCB WILL BE MOVED
PDS050I DELUSER WILL BE MOVED
PDS050I IEBGENER WILL BE MOVED
PDS050I PCHJOB1 WILL BE MOVED
PDS050I TAPELBL2 WILL BE MOVED

PDS200I DISP UNIT OPT RECFM LRECL BLKSIZE   ALLOCTRK FREETRK SECONDARY FREEDIR
PDS200I SHR  3390 C   FB       80   27920   6X    90       11     1 CYL      23

PDS392A SHOULD THIS DATA SET BE MODIFIED (Y/N) ?
```

PDS Subcommands - FIXPDS

- fixpds expanddir(45) (reply 'Y')

```
- DSN=TCONLEY.TSO.JCL,VOL=SER=XS39M1  MEM=PDSHELP  ---------------------------
>y
PDS051I DEFDSCB IS BEING MOVED
PDS051I DELUSER IS BEING MOVED
PDS051I IEBGENER IS BEING MOVED
PDS051I PCHJOB1 IS BEING MOVED
PDS051I TAPELBL2 IS BEING MOVED


PDS200I DISP UNIT OPT RECFM LRECL BLKSIZE   ALLOCTRK FREETRK SECONDARY FREEDIR
PDS200I SHR  3390 C   FB       80   27920   6X    90       10     1 CYL      68
```

PDS Subcommands - HISTORY

- The HISTORY subcommand displays the last linkage edit date for a module and lists any CSECT IDR data assigned to a load module
- history pds

```
- DSN=SYS2.PDS.LOAD,VOL=SER=OS39M2  MEM=PDS  ------------------------
>history pds

** HISTORY  PDS
PDS060I TRANSLATOR HISTORY BY CSECT -
PDSMAIN   3/04/06   569623400   V01 M04
PDSINIT   3/04/06   569623400   V01 M04
...
PDS064I LAST LINK-EDITED ON  3/04/06  BY LKED 5695PMB01  V01 M03  AT  15:18:54
```

PDS Subcommands - IF

- IF searches for members meeting certain criteria

```
Enter the member group name and any operands for IF:
===>

Operands:  memgroup      (e.g., *; start:end; start:; first*pat; range*; part/)
           SINCE/BEFORE
           TODAY/YESTERDAY/WEEK/CURRENT/BIWEEK/MONTH/BIMONTH/QUARTER/
                 HALFYEAR/YEAR/BIYEAR/LAST(numdays)/DATE(mm/dd/yy)
           CHANGED(mm/dd/yy:mm/dd/yy) CREATED(mm/dd/yy:mm/dd/yy)
           ABOVE(Count1)  BELOW(Count2)        NULL/NONULL
           ALIAS/NOALIAS  APPARENTALIAS/NOAPP  ORPHAN/NOORPHAN   HASALIAS/NOHAS
           BLOCKERR/NOBL  IOERR/NOIOERR        LRECLERR/NOLRECL  NAMEERR/NONAME
           MAXBLK(size)   SPFEDIT/NOSPFEDIT    TTR(Lttr:Httr)
           ID(Puid)/NOID/NOTID(Puid)           SSI(hx)/SSI/NOSSI/PARTSSI(hx)

         THEN(ATTRIB   / BROWSE   / DIRENTRY / DELETE  / EDIT    / END     /
              FIND     / FSE      / LIST     / MEMBERS / MEMLIST /OUTCOPY  /
              PRINTOFF / REVIEW   / SUBMIT   / SUBLIST / VIEW    /VERIFY)
         ELSE(as above for THEN)

Defaults:  memgroup, THEN(ATTRIB) if neither THEN nor ELSE is entered, SINCE
```

PDS Subcommands - IF

- if : id(tconley) then(sublist)

```
- DSN=TCONLEY.DYNISPF.EXEC,VOL=SER=XS39M1  MEM=($BOOKINS  ----------------------
>if : id(tconley) then(sublist)
>mem *
PDS165I MEMBERS ARE: $BOOKINS, $CHANGES, $DYNMENU, $FILE495, $INSTALL,
         $READWAC, @ACF2, @CAOPT, @CATSS, @CICSVR, @COMPARX, @DB2I, @D2ADMIN,
         @FAULTAN, @ICSF, @IMSCS, @ISM, @MXI, @NPF, @OPC, @OPSBRW, @PLP, @SASC,
         @TABLBAS, @TEST, @VANGARD, @ZEKE, DSNECPRI, TBPRIM

PDS193I THIS GROUP CONTAINS 29 MEMBERS
```

PDS Subcommands - LIST

- LIST displays the data in a member
- list $$$$$$$$

```
- DSN=SYS2.PDS.INSTALL,VOL=SER=OS39M1  MEM=$$$$$$$$  ----------
>l $$$$$$$$

     Ah-ha, you must have come here because of your curiosity,
     thats good.  This member is an attention getter as first
     member in this file, but the starting point is $$$$READ.
     If you have not been there, now is the time.  Have fun.

PDS142I         6  LINES IN THIS MEMBER
```

PDS Subcommands - MAP

- MAP lists CSECT structure of a load module
- map iefbr14

```
- DSN=SYS1.LINKLIB,VOL=SER=W4RES1  MEM=IEFBR14  ----------
>map iefbr14

** MAP       IEFBR14
IEFBR14   000000  000004

PDS103I ENTRY POINT AT 000000  --  IEFBR14
PDS104I MODULE LENGTH  000008  --      1K
```

PDS Subcommands - MEMBERS

- MEMBERS lists member names from the PDS
- members a:b

```
- DSN=SYS2.PDS.INSTALL,VOL=SER=OS39M1  MEM=A:B  -----
>members a:b
PDS165I MEMBERS ARE: ABEHELP

PDS193I THIS GROUP CONTAINS 1 MEMBERS
```

PDS Subcommands - MEMLIST

- MEMLIST creates member selection list

```
---------------- O.MEML     MEMLIST Subcommand (source members) ---------------
OPTION ===>

Enter the member group specification and any operands for MEMLIST:
===>


Operands:  memgroup   - (e.g., *; start:end; start:; first*pat; range*; part/)
           SINCE/BEFORE
           TODAY/YESTERDAY/WEEK/CURRENT/BIWEEK/MONTH/BIMONTH/QUARTER/
                 HALFYEAR/YEAR/BIYEAR/LAST(numdays)/DATE(mm/dd/yy)
           CHANGED(mm/dd/yy:mm/dd/yy)  CREATED(mm/dd/yy:mm/dd/yy)
           ABOVE(count)
           ALIAS/NOALIAS
           BELOW(count)
           HASALIAS/NOHASALIAS
           ID(pname)/NOID/NOTID(pname)
           ORPHAN/NOORPHAN
           SSI(hexdata)/SSI/NOSSI/PARTSSI(hexdata)
           TTR(lttr:httr)
           RESET/NORESET

Defaults:  NORESET, SINCE
```

PDS Subcommands - MEMLIST

- MEMLIST is cumulative, so use RESET to create a new MEMLIST
- memlist a* reset

```
 ------------------------- MEMLIST Source Member List 1 ------- Row 1 to 1 of 1
 COMMAND ===>                                                  SCROLL ===> CSR
Enter an ISPF command, a PDS subcommand or a special control code:
   8 View log    1 Suspend ISPF    6 MEMLIST all    F Find      R Recall
  LA Lista       4 Sublist =       7 Output table   L Locate   SO Sort
  LV Listv       5 Check aliases   9 Swap panels    O Options   X Above/Below/All
 - DSN=SYS2.PDS.INSTALL,VOL=SER=OS39M1  MEM=A*  --------------------------------
 CMD  NAME     DATA     VER.MOD    CREATED   LAST MODIFIED  SIZE  INIT    ID
      ABEHELP
```

PDS Subcommands - OPTIONS

- OPTIONS lists available PDS subcommands
- options

```
- DSN=SYS2.PDS.INSTALL,VOL=SER=OS39M1  MEM=A:B  ----------------------------
>options
PDS071I THE FOLLOWING OPTIONS ARE AVAILABLE:
ATTRIB   - LIST ATTRIBUTES OF A MEMBER
ALIAS    - ASSIGN AN ALIAS TO A MEMBER
BROWSE   - BROWSE USING ISPF
CHANGE   - SELECT A NEW DATA SET
. . .
```

PDS Subcommands - OUTCOPY

- OUTCOPY can create IEBCOPY or IEBUPDTE control cards for selected members
- DD PDSOUT must be allocated to an FB 80 file
- outcopy $$$$$$$$

```
- DSN=SYS2.PDS.INSTALL,VOL=SER=OS39M1  MEM=$$$$$$$$  --------
>outcopy $$$$$$$$ echo
          COPY OUTDD=OUTPUT,INDD=INSTALL
          S M=$$$$$$$$ ,$$$$$$$$))
>outcopy $$$$$$$$ iebupdte echo
./  ADD  NAME=$$$$$$$$
```

PDS Subcommands - PATTERN

- PATTERN lists members matching the patterns
- PATTERN is no longer recommended, use MEMBERS instead

PDS Subcommands - PRINTOFF

- PRINTOFF prints a hardcopy list of a member

```
------------------------- O.PR    PRINTOFF Subcommand -------------------------
OPTION  ===>

Enter the member group name and any operands below for PRINTOFF:
===>


Operands:  memgroup       (e.g., *; start:end; start:; first*pat; range*; part/)
           CLASS(c)            DEST(destname)
           COPIES(num)         HOLD/NOHOLD
           LIST/NOLIST         PRINT/NOPRINT
           FORMS(formname)     UCS(ucsname)
           PROG(progname)      FCB(fcbname)
           HEADING/NOHEADING   SNUM
           CAPS/ASIS           TEXT
           NOMSG               ASA
           VOLUME(volname)

Defaults:  memgroup, CLASS(A), COPIES(1), NOHOLD, LIST, PRINT, ASIS, HEADING
```

PDS Subcommands - RENAME

- RENAME will rename, swap, or replace members or groups of members in a PDS

```
--------------------------- O.REN    RENAME Subcommand -------------------------
OPTION  ===>

Enter the current member name, the new member name and any operands for RENAME:
===>


Operands:  currname - the current member name
           newname  - the new name for the member
           SWAP     - a member name exchange is to be performed
           GROUP    - members from currname:currname are to be renamed
           REPBY(z) - z names an existing member which is to become currname

Defaults:  none
```

PDS Subcommands - RENAME

- Rename tomold tomnew swap

```
- DSN=TCONLEY.TSO.JCL,VOL=SER=XS39M1  MEM=TOMNEW  ----------
>rename tomold tomnew swap
PDS090I TOMOLD HAS BEEN RENAMED TO TOMNEW
PDS090I TOMNEW HAS BEEN RENAMED TO TOMOLD
```

PDS Subcommands - REPLACE

- REPLACE finds and replaces strings in members

```
------------------------- O.REP    REPLACE Subcommand -------------------------
OPTION  ===>

Enter the member group name, search/replace strings and operands for REPLACE:
===>

Operands:  memgroup  (e.g., *; start:end; start:; first*pat; range*; part/)
                     -search-replace-   /  'search' 'replace'
                     NUM / SNUM / NONUM / LBLOCK / LDUMP / BLOCK / DUMP
                     WRITE/NOWRITE
                     CAPS/ASIS/IGNORE
                     STATS/NOSTATS
                     OFFSET(hex)       MODULE(Partname)
                     MAXIN(num)        MAXLEN(num)
                     MAXOUT(num)       MAXFIND(num)
                     SKIPCOL(num)      SKIPREC(num)

Defaults:  memgroup; NUM or previous REPLACE/FIND/LIST format;
           NOWRITE; CAPS; STATS
```

PDS Subcommands - REPLACE

- replace ady* /tom/new/ (write)

```
- DSN=TCONLEY.TSO.JCL,VOL=SER=XS39M1   MEM=ADY*   ------------------------
>replace ady* /tom/new/

** REPLACE   ADYSET00
002000 *    THIS IS EXECUTED AUNEWATICALLY DURING SYSTEM INITIALIZATION
PDS142I         15   LINES IN THIS MEMBER
>replace ady* /tom/new/ write

** REPLACE   ADYSET00
002000 *    THIS IS EXECUTED AUNEWATICALLY DURING SYSTEM INITIALIZATION
PDS142I         15   LINES IN THIS MEMBER
PDS145I          1 BLOCKS UPDATED
```

PDS Subcommands - REPRO

- REPRO can rebuild or add members to a PDS
- Repro iefbr14 to(iefbr15)

```
- DSN=TCONLEY.TSO.JCL,VOL=SER=XS39M1  MEM=IEFBR15  ---------------
>repro iefbr14 to(iefbr15)
PDS050I IEFBR14 WILL BE COPIED TO IEFBR15

PDS396A SHOULD REPRO CONTINUE (Y/N) ?
>y
PDS051I IEFBR15 IS BEING CREATED
```

PDS Subcommands - RESTORE

- RESTORE resurrects deleted members in PDS
- restore zmem repeat noprompt ttr(0)

```
- DSN=TCONLEY.TSO.JCL,VOL=SER=XS39M1  MEM=ZMEM:ZMEM  -------------------------
>restore zmem repeat noprompt ttr(0)


PDS101I DELETED MEMBER FOUND AT TTR: 00020D

PDS144I DATA LINE 1:
//TCONLEYA JOB (TCONLEY),'ADD USER',CLASS=A,                          00010000
PDS144I DATA LINE 2:
//             MSGCLASS=H,NOTIFY=&SYSUID                              00020000
PDS144I DATA LINE 3:
//*****************************************************              00030000
PDS144I DATA LINE 4:
//ADDUSER  EXEC PGM=IKJEFT01                                          00040000
PDS144I DATA LINE 5:
//SYSTSPRT DD SYSOUT=*                                                00050000

PDS091I ZMEM0001 HAS BEEN RESTORED
```

PDS Subcommands - RESTORE

- Options CAPS/ASIS (for FIND) and TEST to simulate RESTORE added with version 8.6.04

```
------------------------- O.RES    RESTORE Subcommand -------------------------
OPTION ===>

Enter the member name desired and any operands below for RESTORE:
===>


Operands:  member     - name to use for the restored member
           TTR(ttr)   - one to six hex digit ttr address or search start address
           REPEAT     - restore multiple members (mbr00001, mbr00002, ...)
           NOREPEAT   - restore only a single member
           DISPLAY    - display data from deleted members
           NODISPLAY  - display no data from deleted members
           PROMPT     - prompt before restoring a member
           NOPROMPT   - do not prompt before restoring a member
           COUNT(nm)  - number of lines to display from a member
           LIKE(xyz)  - take attributes from member XYZ
           FIND('s')  - a character string which must be found before a restore
           MODULE(n)  - a partial CSECT or entry name which must be present
           CAPS/ASIS  - case of the FIND argument
           TEST       - simulate the resurrection of deleted members

Defaults:  TTR(0), DISPLAY, PROMPT, COUNT(5), CAPS
```

PDS Subcommands - SRCHFOR

- SRCHFOR invokes SuperC to search the PDS

```
------------------------- O.SRCH    SRCHFOR Subcommand ------------------------
OPTION  ===>
Enter the member group name, the string and any operands for SCRHFOR:
===>

Operands:  memgroup      (e.g., *; start:end; start:; first*pat; range*; part/)
           'string'
           ALIAS/NOALIAS       - to include associated members
           HEX                 - string is hexadecimal
           PREFIX/SUFFIX/WORD  - type of search
           COL(start)          - start search in this column
           COL(start:end)      - limit search to these columns



Required:  memgroup, string
Defaults:  ALIAS

Process options:  ANYC, LONGLN (IBM SuperC Search-For)
```

PDS Subcommands - SRCHFOR

- srchfor $* ‘pdsjcl’ noslist (undocumented option NOSLIST suppresses SELECT <member> display if not processing ‘:’ for all members)

```
- DSN=SYS2.PDS.INSTALL,VOL=SER=OS39M1  MEM=PDSHELP  ------------------------------------------------------------------------
>srchfor $* 'pdsjcl‘ noslist
  ISRSUPC    -   MVS/PDF FILE/LINE/WORD/BYTE/SFOR COMPARE UTILITY- ISPF FOR z/OS        2006/03/09   2.26    PAGE     1
 LINE-#  SOURCE SECTION                   SRCH DSN: SYS2.PDS.INSTALL


 $$$DOC                      --------- STRING(S) FOUND -------------------

     36    PDSJCL   -- JCL to install PDS
. . .
  ISRSUPC    -   MVS/PDF FILE/LINE/WORD/BYTE/SFOR COMPARE UTILITY- ISPF FOR z/OS        2006/03/09   2.26    PAGE     2
    SEARCH-FOR SUMMARY SECTION           SRCH DSN: SYS2.PDS.INSTALL

LINES-FOUND  LINES-PROC  MEMBERS-W/LNS  MEMBERS-WO/LNS  COMPARE-COLS  LONGEST-LINE
        9       22998           5            37          1:80          80

PROCESS OPTIONS USED: LONGLN ANYC

THE FOLLOWING PROCESS STATEMENTS (USING COLUMNS 1:72) WERE PROCESSED:
   SRCHFOR 'pdsjcl'
   SLIST OFF

PDS171I SRCHFOR HAS COMPLETED; RC=  1
```

PDS Subcommands - SUBLIST

- SUBLIST creates member sublists
- sublist $$*

```
 - DSN=SYS2.PDS.INSTALL,VOL=SER=OS39M1   MEM=($$$$$$$$   --------------
>subl $$*
>mem *
PDS165I MEMBERS ARE: $$$$$$$$, $$$$READ, $$$$REA1, $$$BUGS, $$$CTBL, $$$CUSTM,
         $$$CUT, $$$DOC, $$$EQUAL, $$$HERC, $$$INST, $$$ISODT, $$$MODS,
         $$$PDSE, $$$PROSE, $$$SECUR, $$$TRAP, $$$UCB, $$$UNDOC, $$$Y2K,
         $$DELINK, $$DISASM, $$DRK, $$DSLIST, $$GP, $$MXIBAT, $$PCX, $$PDSMAN,
         $$PDSPPT, $$QUOTES, $$SHARE1, $$S2837, $$USAGE

PDS193I THIS GROUP CONTAINS 33 MEMBERS
```

PDS Subcommands - SUBMIT

- SUBMIT will submit a member or group of members to the internal reader for execution
- sub iefbr14
- sub tconley*
- sub a:b

PDS Subcommands - USAGE

- USAGE displays stats for current dataset
- usage all

```
- DSN=SYS2.PDS.INSTALL,VOL=SER=OS39M1  MEM=PDSHELP  ---------------------------
>USAGE  ALL
PDS200I DISP UNIT OPT RECFM LRECL BLKSIZE   ALLOCTRK FREETRK SECONDARY FREEDIR
PDS200I SHR  3390 C   FB       80   27920   5X   401     167    50 TRK      43

PDS180I DATA SET: CREATED  EXPIRES  LAST USE  UPDATED  SECURITY
PDS180I           5/11/05  0/00/00   3/09/06  YES      NONE

PDS181I EXTENTS IN TRACKS: 201, 50, 50, 50, 50

PDS182I TRACKS: ALLOCATED    USED    FREE  EXTENTS
PDS182I               401     234     167        5

PDS183I DIRECTORY: BLOCKS    USED    FREE   TRACKS  MEMBERS  ALIASES
PDS183I               180     137      43        4     1011        1
```

PDS Subcommands - USAGE

- usage all (cont'd).

```
PDS184I EXTENT  UCB  LO TT-HI TT  TRACKS     LOW CCHH-HIGH CCHH    BOUNDARY
PDS184I ------  ---- ----- -----  ------  ----------- -----------  --------
PDS184I      0  0A82 00.00 00.C8     201  05.FD.00.0C 06.0B.00.02  TRK
PDS184I      1  0A82 00.C9 00.FA      50  06.C3.00.00 06.C6.00.04  TRK
PDS184I      2  0A82 00.FB 01.2C      50  06.C6.00.05 06.C9.00.09  TRK
PDS184I      3  0A82 01.2D 01.5E      50  06.C9.00.0A 06.CC.00.0E  TRK
PDS184I      4  0A82 01.5F 01.90      50  06.CD.00.00 06.D0.00.04  TRK


PDS185I FORMAT 1 DSCB:
PDS185I 0   4   8   C  10  14  18  1C  20  24  28  2C  30  34  38  3C  40  44
PDS185I SYS2.PDS.INSTALL                              1OS39M1....c......IBMOSVS2
PDS185I EEEF4DCE4CDEECDD44444444444444444444444444FDEFFDF00608000000CCDDEEEF
PDS185I 2822B742B952313300000000000000000000000016239410190300050092462522

PDS185I  48  4C  50  54  58  5C  60  64  68  6C  70  74  78  7C  80  84  88
PDS185I      ¦...._....._..&...b&....Z..................C...F.....F...I.......
PDS185I 444446040861009261050008500 60E0B800000F000000000C000C00000C000C0000003
PDS185I 00000A0400D02000D0000002000409220001 05D0C6B02116300660412660569090002 2
```

PDS Subcommands - USAGE

- usage all (cont'd).

```
PDS186I LOC NAME     VALUE        DESCRIPTION
PDS186I --- ----     -----        -----------
PDS186I 00  DS1DSNAM SYS2.PDS.INSTALL
PDS186I 2C  DS1FMTID F1           FORMAT IDENTIFIER
PDS186I 2D  DS1DSSN  OS39M1       DATA SET SERIAL NUMBER
PDS186I 33  DS1VOLSQ 1.           VOLUME SEQUENCE NUMBER
PDS186I 35  DS1CREDT 690083       CREATION DATE
PDS186I 38  DS1EXPDT 000000       EXPIRATION DATE
PDS186I 3B  DS1NOEPV 5.           NUMBER OF EXTENTS ON VOLUME
PDS186I 3C  DS1NOBDB 0.           NUMBER OF BYTES USED IN LAST DIRECTORY BLOCK
PDS186I 3D  DS1FLAG1 00           FLAG 1
PDS186I 3E  DS1SYSCD C9C2D4D6E2E5E2F24040404040
PDS186I 4B  DS1REFD  6A0044       DATE LAST REFERENCED
PDS186I 4E  DS1SMSFG 00           SYSTEM MANAGED STORAGE INDICATORS
PDS186I 4F  DS1SCEXT 806D10       SECONDARY SPACE EXTENSION
PDS186I 52  DS1DSORG 0200         DATA SET ORGANIZATION
PDS186I 54  DS1RECFM 90           RECORD FORMAT
PDS186I 55  DS1OPTCD 20           OPTION CODE
PDS186I 56  DS1BLKL  27,920.      BLOCK LENGTH
```

PDS Subcommands - USAGE

- usage all (cont'd).

```
PDS186I 58  DS1LRECL 80.          LOGICAL RECORD LENGTH
PDS186I 5A  DS1KEYL  0.           KEY LENGTH
PDS186I 5B  DS1RKP   0.           RELATIVE KEY POSITION
PDS186I 5D  DS1DSIND 82           DATA SET INDICATORS
PDS186I 5E  DS1SCALO 50000064     SECONDARY ALLOCATION
PDS186I 62  DS1LSTAR 00E902       TTR OF LAST USED TRACK AND BLOCK ON TRACK
PDS186I 65  DS1TRBAL 45,696.      BYTES REMAINING ON LAST TRACK USED
PDS186I 67           00           RESERVED (ONE BYTE)
PDS186I 68  DS1TTTHI 00           HIGH ORDER BYTE OF TRACK NUMBER IN DS1LSTAR
PDS186I 69  DS1EXT1  0100.05FD000C.060B0002  FIRST EXTENT DESCRIPTION
PDS186I 73  DS1EXT2  0101.06C30000.06C60004  SECOND EXTENT DESCRIPTION
PDS186I 7D  DS1EXT3  0102.06C60005.06C90009  THIRD EXTENT DESCRIPTION
PDS186I 87  DS1PTRDS 0000000232   CCHHR OF ANY ASSOCIATED FORMAT 2 OR 3 DSCB
```

PDS Subcommands - VERIFY

- VERIFY runs a validity check on the PDS
- verify :

```
- DSN=SYS2.PDS.INSTALL,VOL=SER=OS39M1  MEM=:  -------------------------
>verify :

PDS006I END OF DATA SET

PDS110I   131,889 LOGICAL RECORDS WERE INPUT
PDS111I     1,227 PHYSICAL BLOCKS WERE INPUT
PDS112I    27,920 CHARACTERS IN THE LARGEST PHYSICAL BLOCK
PDS113I     8,599 CHARACTERS PER AVERAGE PHYSICAL BLOCK
PDS114I         0 TRACKS COULD BE REGAINED BY COMPRESSING THIS DATA SET
PDS115I     1,011 MEMBERS WERE CHECKED


PDS117I 1,010 MEMBERS COUNTED; CUMULATIVE SIZE IS 131,889 RECORDS
```

PDS Subcommands - VIEW

- VIEW opens a PDS member in ISPF VIEW mode
- view $$$$read

```
  File  Edit  Edit_Settings  Menu  Utilities  Compilers  Test  Help
-------------------------------------------------------------------------------
VIEW       SYS2.PDS.INSTALL($$$$READ) - 01.00                  Columns 00001 00080
Command ===>                                                    Scroll ===> CSR
****** ******************************** Top of Data ********************************
==MSG> -Warning- The UNDO command is not available until you change
==MSG>           your edit profile using the command RECOVERY ON.
000001     The adventure starts here.   Armed with only your imagination and
000002     the source to PDS you are ready to do battle within the labyrinth
000003     of the IBM partitioned data set structure.  In the past, the PDS
```

PDS Subcommands - VPSPRINT

- VPSPRINT prints hardcopy listing of a member

```
------------------------- O.VPS    VPSPRINT Subcommand -----------------------
OPTION  ===>

Enter the member group name, the printer name and operands for VPSPRINT:
===>

Operands:  memgroup      (e.g., *; start:end; start:; first*pat; range*; part/)
           printername
           DDNAME(ddname)  DIRECT(KEEP/DELETE)
           LINES(linenum1:linenum2)
           NUM(loc,length)/SNUM(loc,length)/NONUM
           FCB(fcbname)  FORMS(formname)
           CLASS(c)  HOLD/NOHOLD  COL(col1:col2, ...)  COPIES(number)
           PAGELEN(num)  TMARGIN(num)  BMARGIN(num)
           SINGLE/DOUBLE/CCHAR
           FOLD(width)/TRUNCATE(width)
           EJECT/NOEJECT  NOEJMESS  NOHEADER
           TERMINAL  TRACE  WTR(wtrname)

Required:  printername
Defaults:  NUM, CLASS(A), COPIES(1), NOHOLD, SINGLE, HEADER, NOEJECT, MEMBERS
```

PDS Subcommands - XMIT (new!)

- XMIT uses the TSO TRANSMIT command to act on members in the PDS (new in 8.6.03)

```
------------------------- O.XMIT    XMIT Subcommand -------------------------
OPTION  ===>

Enter the member group name, the addressee and any operands for XMIT:
===>

Operands:  memgroup      (e.g., *; start:end; start:; first*pat; range*; part/)
           addressee           - node.userid, nickname, or distribution list
           OUTDSN(dsname)      - direct output to a file instead of JES
           OUTFILE(ddname)     - direct output to a file instead of JES
           ALIAS/NOALIAS       - to include associated members
           NOTIFY/NONOTIFY     - request notification of delivery
           LOG/NOLOG           - log all transmissions in the log data set
           SYSOUT(class)       - direct utility messages to a sysout class



Required:  memgroup, addressee
Defaults:  ALIAS, NOTIFY, LOG
```

PDS Subcommands - XMIT (new!)

- xmit $$$* a.a outda('tconley.xmit')

```
                                    IEBCOPY MESSAGES AND CONTROL STATEMENTS                          PAGE     1
IEB1135I IEBCOPY  FMID HDZ11G0  SERVICE LEVEL UA05363  DATED 20030923 DFSMS 01.03.00 z/OS    01.04.00 HBB7707  CPU 2066
IEB1035I TCONLEY  ISPFPROC ISPFPROC 18:27:57 SUN 14 MAY 2006 PARM=''
 COPY OUTDD=SYS00008,INDD=((SYS00004,R))
 SELECT MEMBER=($$$$$$$$,$$$$READ,$$$$REA1,$$$BUGS,$$$CTBL,$$$CUSTM)
 SELECT MEMBER=($$$CUT,$$$DOC,$$$EQUAL,$$$HERC,$$$INST,$$$ISODT)
 SELECT MEMBER=($$$MODS,$$$PDSE,$$$PROSE,$$$SECUR,$$$TRAP,$$$UCB)
 SELECT MEMBER=($$$UNDOC,$$$Y2K)
IEB1013I COPYING FROM PDS    INDD=SYS00004 VOL=OS39M1 DSN=SYS2.PDS.INSTALL
IEB1014I           TO PDSU OUTDD=SYS00008 VOL=OS39M1 DSN=SYS06134.T182757.RA000.TCONLEY.R0100257
IEB167I FOLLOWING MEMBER(S) UNLOADED FROM INPUT DATA SET REFERENCED BY SYS00004
IEB154I $$$$$$$$ HAS BEEN SUCCESSFULLY UNLOADED
IEB154I $$$$READ HAS BEEN SUCCESSFULLY UNLOADED
IEB154I $$$$REA1 HAS BEEN SUCCESSFULLY UNLOADED
.
.
.
IEB154I $$$PDSE  HAS BEEN SUCCESSFULLY UNLOADED
IEB154I $$$PROSE HAS BEEN SUCCESSFULLY UNLOADED
IEB154I $$$SECUR HAS BEEN SUCCESSFULLY UNLOADED
IEB154I $$$TRAP  HAS BEEN SUCCESSFULLY UNLOADED
IEB154I $$$UCB   HAS BEEN SUCCESSFULLY UNLOADED
IEB154I $$$UNDOC HAS BEEN SUCCESSFULLY UNLOADED
IEB154I $$$Y2K   HAS BEEN SUCCESSFULLY UNLOADED
IEB1098I 20 OF 20 MEMBERS UNLOADED FROM INPUT DATA SET REFERENCED BY SYS0000
IEB147I END OF JOB - 0 WAS HIGHEST SEVERITY CODE
INMX000I 0 message and 54 data records sent as 7272 records to A.A


INMX001I Transmission occurred on 05/14/2006 at 18:27:56.
```

Miscellaneous Commands

- CAX - Display active catalogs (CAXWA)
- DISASM - Run a disassembler
- DLINK - De-link and breakdown a load module
- FSE - Full Screen Edit
- H - Online help (HEL command)
- LISTA - List TSO allocations
- LISTC/LISTF - List files w/DSAT or VTOC

Miscellaneous Commands

- LISTV - LSPACE list of volumes
- MXIBAT - MXI interface to PDS
- REV - REVIEW file browser

Summary

- Discussed the history of PDS
- Installing and learning to use PDS
- Talked about PDSE support
- Discussed member groups
- Showed how to get started
- Reviewed PDS subcommands
- Covered miscellaneous commands

Finally…

- I'm interested in hearing about your experiences with PDS; if you discover any unique ways to use PDS, please let me know about them
- Feel free to Email me at pinncons@rochester.rr.com if you have any questions or concerns regarding this presentation or PDS in general

ERROR: undefined
OFFENDING COMMAND:

STACK: