Achieving Higher Levels of Productivity with IBM ISPF Productivity Tool for z/OS

IBM Redbooks Solution Guide

IBM® ISPF Productivity Tool for z/OS® is an ISPF application that provides significant productivity improvements and time savings to ISPF users on IBM z/OS systems. ISPF Productivity Tool operates as an integrated front end to ISPF, and it enhances and extends ISPF capabilities. This tool also delivers new batch and online functions to manipulate partitioned and sequential data set directories, and it improves performance for copy and compress operations for partitioned data sets. ISPF Productivity Tool is a productivity aid to systems and applications programmers because of the various functions it provides. For example, it includes shortcuts, navigational aids, automatic PDS compression and directory expansion, and search capabilities for volumes, data sets, members, and text within members.

Using ISPF Productivity Tool for z/OS can help your z/OS installations achieve higher levels of productivity.

ISPF Productivity Tool for z/OS helps make any ISPF user more productive.

Figure 1. ISPF Productivity Tool for z/OS

Did you know?

Many IBM z/OS users have experience in working with ISPF and know how to quickly navigate the panels, but other users who are new to the IBM System z® environment, were trained in workstation GUI tools and interfaces. Between these groups of users, one common trait is that, when they work in ISPF, they are looking for tools that offer time-saving advantages, which they now have with IBM ISPF Productivity Tool for z/OS. ISPF Productivity Tool for z/OS increases the productivity of existing ISPF users and attracts new users by enabling them to become efficient quickly. ISPF Productivity Tool boosts productivity for all users, including developers, system programmers, technical support, and operations and control personnel.

Business value

Time is valuable for development organizations, making productivity tools important to the overall bottom line of operations costs. IBM ISPF Productivity Tool for z/OS boosts productivity for all ISPF users who need tools that offer time-saving advantages.

By using ISPF Productivity Tool, you can increase daily performance and productivity by minimizing the amount of typing and screen navigation that are required to perform specific tasks. ISPF Productivity Tool provides shortcuts that help you to make more efficient use of your time when working on a z/OS mainframe. ISPF Productivity Tool standardizes the tools that you need to be productive with ISPF. These aids are great time savers for experienced users and are often of greater value to inexperienced users by providing them a simplified path for performing tasks. In addition to the many usability features, ISPF Productivity Tool supports the double-byte character set (DBCS).

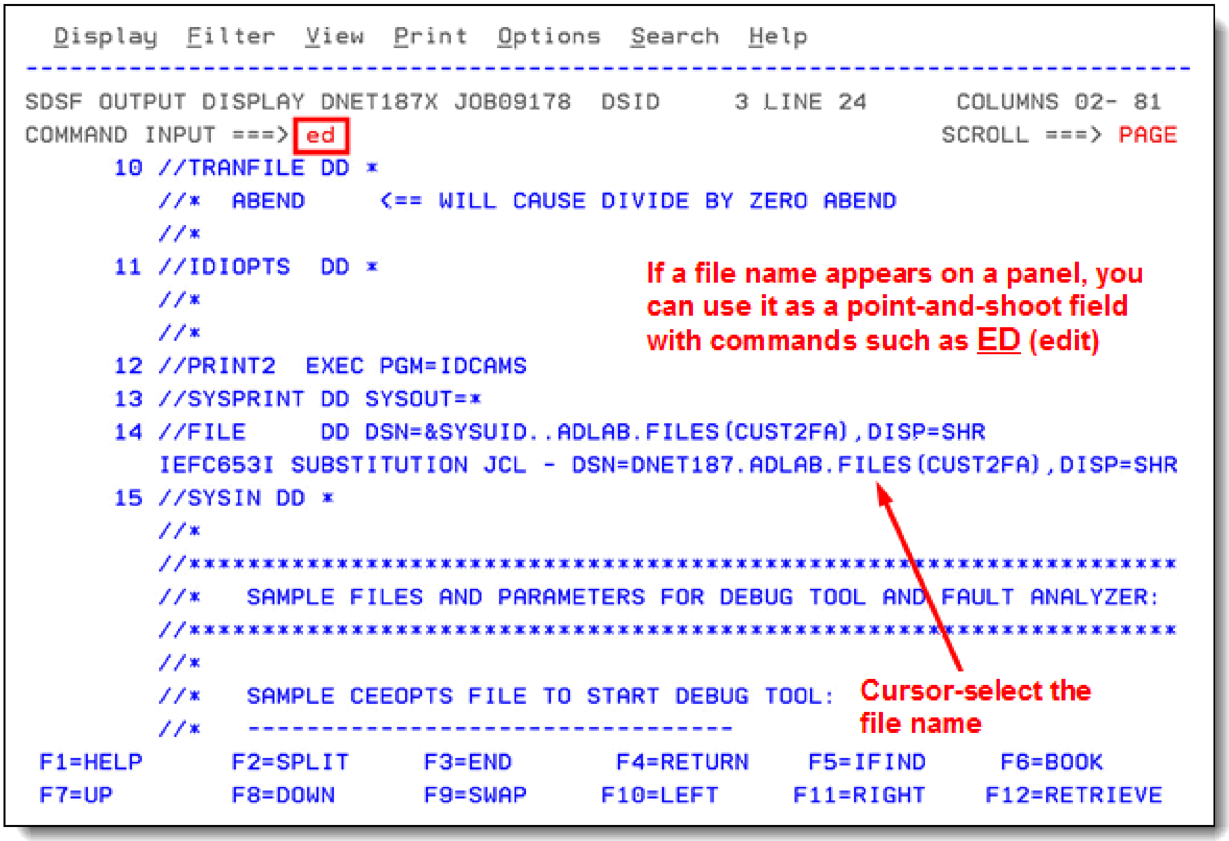
Solution overview

IBM ISPF Productivity Tool combines a range of ISPF utilities and new functions with the standard ISPF functions, such as Browse, Edit, and View. It provides extensive search capabilities and provides drill down capabilities to locate volumes and data sets with extended functions at all levels. Programmers access these extended features by entering commands on the command line or in the selection field for a particular data set or member. They can also use hotbars, field-sensitive areas, and point-and-shoot techniques. Extensive help is available by using a simple command from any panel.

The functions of the ISPF Productivity Tool are added to a user's session through an update of the Time Sharing Option (TSO) logon procedure. These functions enhance productivity through many commands, utilities, and navigational features for commonly used activities. System and application programmers can save a substantial amount of navigation time and can get their work done more efficiently.

Solution architecture

ISPF Productivity Tool is integrated into almost every aspect of ISPF to provide numerous productivity aids and shortcuts. For example, when viewing the output from a job, you want to open a data set that you see in the output. You can enter a command, such as the edit (**ed**) command. Then, you point your cursor to the data set name and press Enter (Figure 2) to open the file for edit.



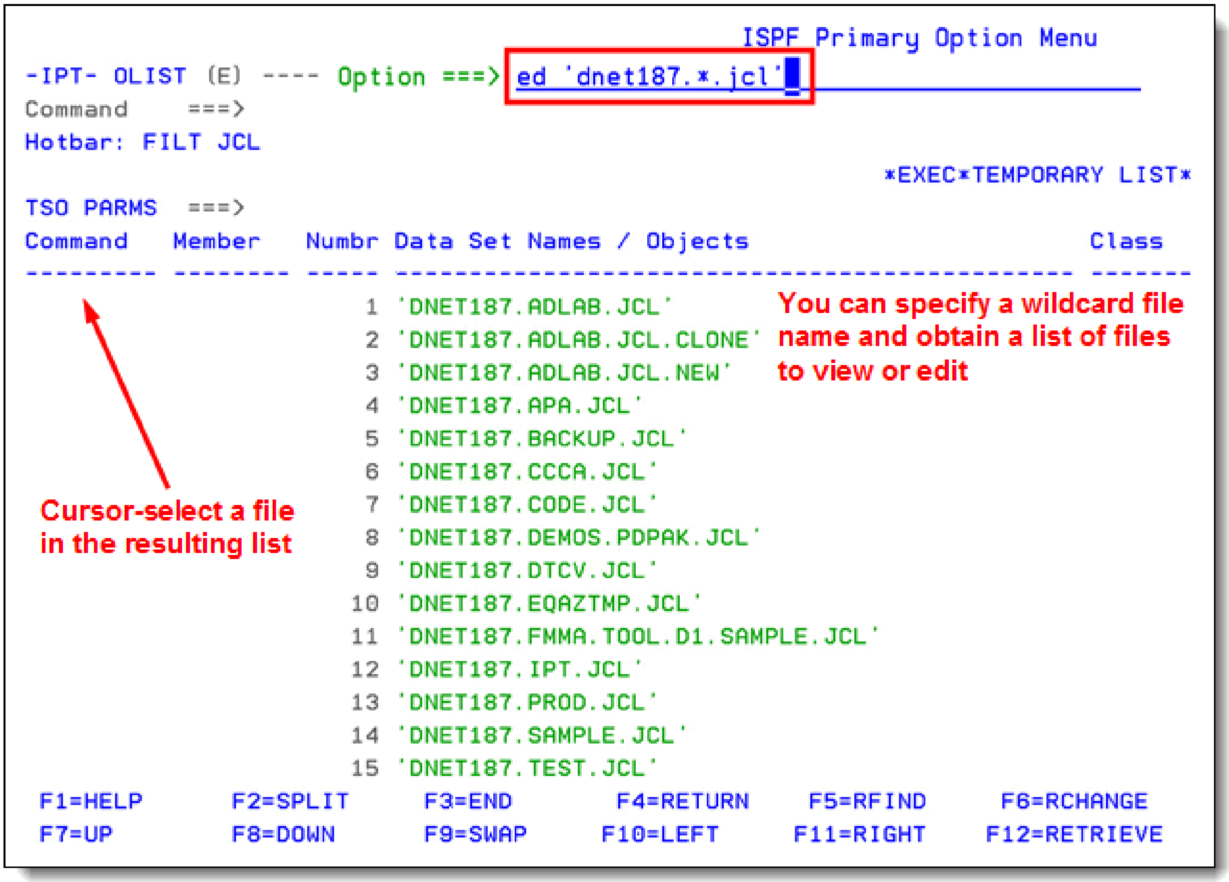
```
  Display  Filter  View  Print  Options  Search  Help
 -------------------------------------------------------------------------------
 SDSF OUTPUT DISPLAY DNET187X JOB09178  DSID     3 LINE 24       COLUMNS 02- 81
 COMMAND INPUT ===> ed                                          SCROLL ===> PAGE
       10 //TRANFILE DD *
          //*  ABEND     <== WILL CAUSE DIVIDE BY ZERO ABEND
          //*
       11 //IDIOPTS  DD *
          //*
          //*
       12 //PRINT2  EXEC PGM=IDCAMS
       13 //SYSPRINT DD SYSOUT=*
       14 //FILE     DD DSN=&SYSUID..ADLAB.FILES(CUST2FA),DISP=SHR
          IEFC653I SUBSTITUTION JCL - DSN=DNET187.ADLAB.FILES(CUST2FA),DISP=SHR
       15 //SYSIN DD *
          //*
          //*********************************************************************
          //*   SAMPLE FILES AND PARAMETERS FOR DEBUG TOOL AND FAULT ANALYZER:
          //*********************************************************************
          //*
          //*   SAMPLE CEEOPTS FILE TO START DEBUG TOOL:
          //*   ---------------------------------
  F1=HELP      F2=SPLIT     F3=END       F4=RETURN    F5=IFIND     F6=BOOK
  F7=UP        F8=DOWN      F9=SWAP     F10=LEFT     F11=RIGHT    F12=RETRIEVE
```

If a file name appears on a panel, you can use it as a point-and-shoot field with commands such as ED (edit)

Cursor-select the file name

Figure 2. Pointing to a data set name in a point-and-shoot text field to open the file for edit

Another shortcut is opening a file for edit. You can enter the edit command (**ed**) by using a pattern to list all data sets that match the pattern and then select a file in the resulting list (Figure 3).



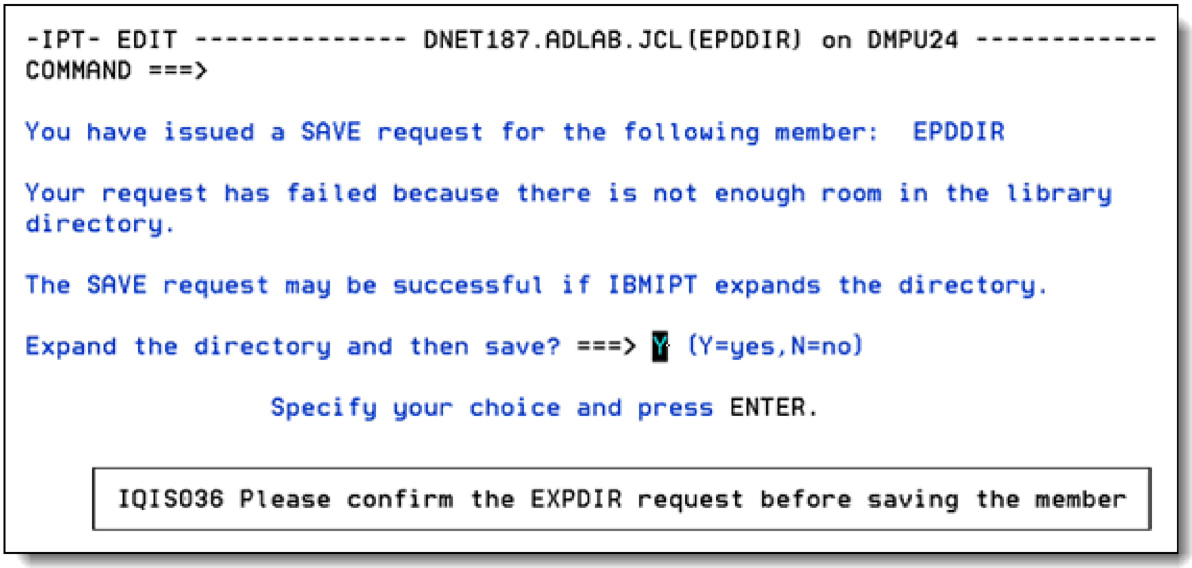
```
                                                ISPF Primary Option Menu
-IPT- OLIST (E) ---- Option ===> ed 'dnet187.*.jcl'
Command    ===>
Hotbar: FILT JCL
                                                          *EXEC*TEMPORARY LIST*
TSO PARMS  ===>
Command   Member   Numbr Data Set Names / Objects                       Class
--------- -------- ----- ---------------------------------------------  -------
                       1 'DNET187.ADLAB.JCL'
                       2 'DNET187.ADLAB.JCL.CLONE'
                       3 'DNET187.ADLAB.JCL.NEW'
                       4 'DNET187.APA.JCL'
                       5 'DNET187.BACKUP.JCL'
                       6 'DNET187.CCCA.JCL'
                       7 'DNET187.CODE.JCL'
                       8 'DNET187.DEMOS.PDPAK.JCL'
                       9 'DNET187.DTCV.JCL'
                      10 'DNET187.EQAZTMP.JCL'
                      11 'DNET187.FMMA.TOOL.D1.SAMPLE.JCL'
                      12 'DNET187.IPT.JCL'
                      13 'DNET187.PROD.JCL'
                      14 'DNET187.SAMPLE.JCL'
                      15 'DNET187.TEST.JCL'
 F1=HELP      F2=SPLIT     F3=END       F4=RETURN    F5=RFIND     F6=RCHANGE
 F7=UP        F8=DOWN      F9=SWAP     F10=LEFT     F11=RIGHT    F12=RETRIEVE
```

You can specify a wildcard file name and obtain a list of files to view or edit

Cursor-select a file in the resulting list

Figure 3. Shortcut for selecting a file for edit

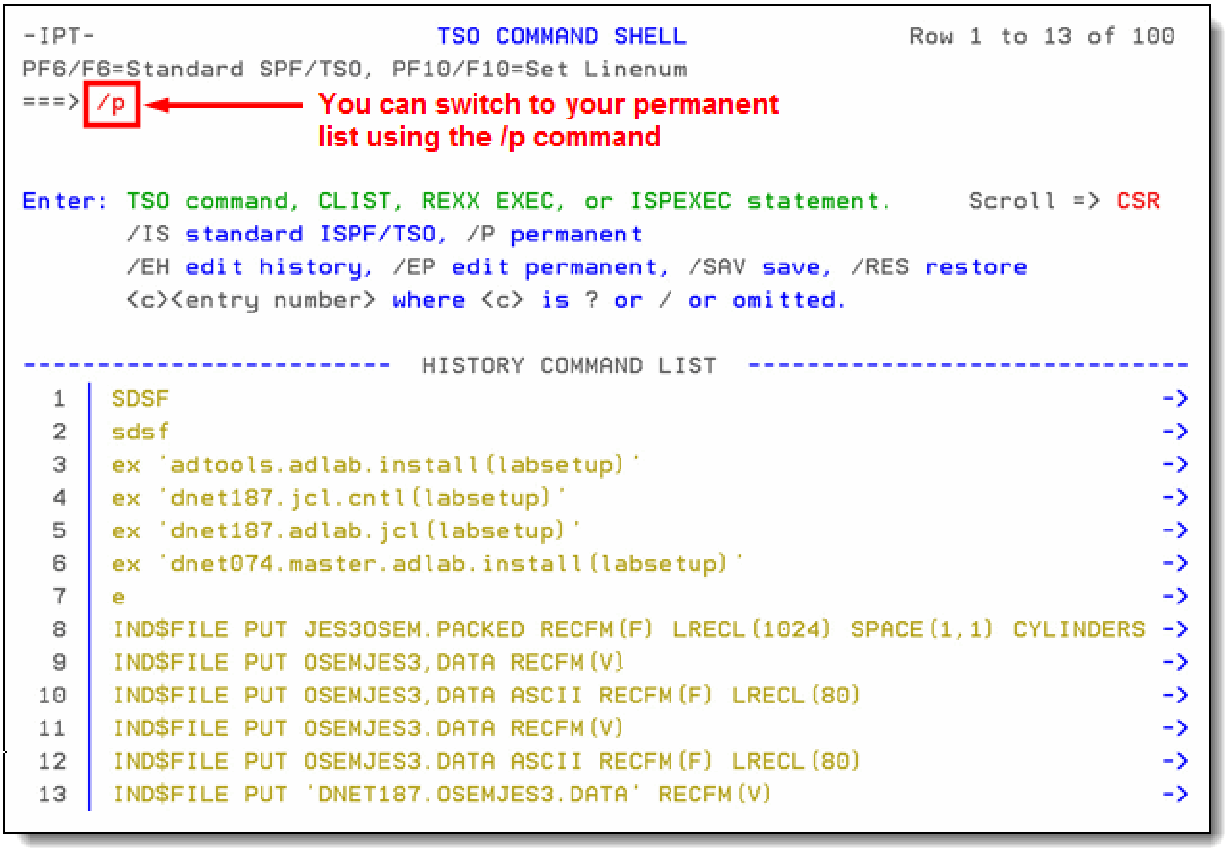
If you have seen the “No space in directory” message when adding members to a partitioned data set (PDS), you might recall the number of panels and keystrokes that ISPF requires when allocating a new PDS, copying the members, and renaming the data set. With ISPF Productivity Tool, you see a confirmation window to expand the PDS directory (Figure 4). When you press Enter, the PDS directory is expanded, without leaving the edit session of your PDS member.



```
-IPT- EDIT -------------- DNET187.ADLAB.JCL(EPDDIR) on DMPU24 ------------
COMMAND ===>

You have issued a SAVE request for the following member:  EPDDIR

Your request has failed because there is not enough room in the library
directory.

The SAVE request may be successful if IBMIPT expands the directory.

Expand the directory and then save? ===> Y (Y=yes,N=no)

                Specify your choice and press ENTER.

     IQIS036 Please confirm the EXPDIR request before saving the member
```

Figure 4. Expanding a directory when needed

Another time saver is the history and permanent command lists that is available in the TSO command shell. The *history command list* is the list of most recently executed commands, with older entries rolling off as new entries are added. The maximum number of entries in the history command list is 999 commands. You can run history commands from TSO Option 6. The *permanent command list* contains the commands that you intend to retain permanently. The maximum capacity of the permanent command list is 999 commands. You can run permanent commands from any ISPF panel. The history command list is displayed when you enter the TSO command shell (Figure 5). You can switch to the permanent list by entering the **/p** command.



```
-IPT-                       TSO COMMAND SHELL                   Row 1 to 13 of 100
PF6/F6=Standard SPF/TSO, PF10/F10=Set Linenum
===> /p

Enter: TSO command, CLIST, REXX EXEC, or ISPEXEC statement.      Scroll => CSR
       /IS standard ISPF/TSO, /P permanent
       /EH edit history, /EP edit permanent, /SAV save, /RES restore
       <c><entry number> where <c> is ? or / or omitted.

-------------------------  HISTORY COMMAND LIST  ------------------------------
   1 | SDSF                                                                   ->
   2 | sdsf                                                                   ->
   3 | ex 'adtools.adlab.install(labsetup)'                                   ->
   4 | ex 'dnet187.jcl.cntl(labsetup)'                                        ->
   5 | ex 'dnet187.adlab.jcl(labsetup)'                                       ->
   6 | ex 'dnet074.master.adlab.install(labsetup)'                            ->
   7 | e                                                                      ->
   8 | IND$FILE PUT JES3OSEM.PACKED RECFM(F) LRECL(1024) SPACE(1,1) CYLINDERS ->
   9 | IND$FILE PUT OSEMJES3,DATA RECFM(V)                                    ->
  10 | IND$FILE PUT OSEMJES3,DATA ASCII RECFM(F) LRECL(80)                    ->
  11 | IND$FILE PUT OSEMJES3.DATA RECFM(V)                                    ->
  12 | IND$FILE PUT OSEMJES3.DATA ASCII RECFM(F) LRECL(80)                    ->
  13 | IND$FILE PUT 'DNET187.OSEMJES3.DATA' RECFM(V)                          ->
```

You can switch to your permanent list using the /p command

Figure 5. History command list in the TSO command shell

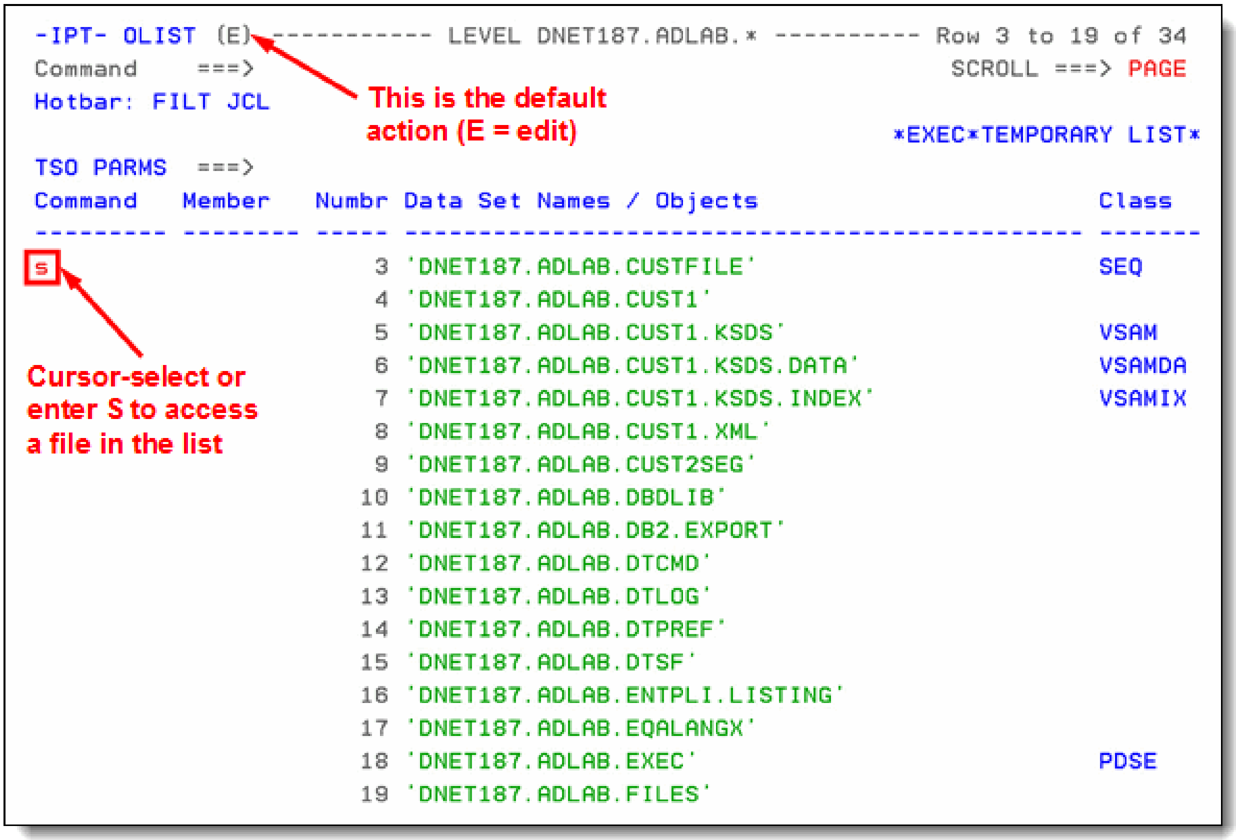
For organizational aids, ISPF Productivity Tool introduces the concept of an object list (OLIST). An *object list* is a data store reference that is used to organize all of the required resources for a project into a single project folder called an OLIST. An OLIST is a list of files and objects that can include the following items and more:

- Sequential files
- PDSs
- PDSs extended (PDSE)
- Virtual Storage Access Method (VSAM) files
- PDS members
- IBM DB2® tables
- z/OS UNIX directories and objects

Many commands are available in ISPF Productivity Tool to work with OLISTs. These commands introduce a various ways to use OLISTs. For example, you can create a temporary OLIST that contains all the files that are allocated to your TSO session. Another example is creating a temporary OLIST of authorized libraries (by the authorized program facility (APF)) and using the VALIDATE function to ensure that the list is current. Alternatively, you can create an OLIST of all files that begin with "IMS" but that do not contain the characters "test." Then you can perform some action on the files as a group, for example **recall** or **delete**.

Although you can create a temporary view of data objects, sometimes you might want to save a set of objects that you can use frequently in a particular operation. The set of objects can be as simple as the set of libraries that you use the most or the set of databases that you monitor or maintain.

After you have your OLIST, you can go through the processes one step at a time by selecting and acting on the files in the list. From the same OLIST, you can also perform such actions as edit, browse, view, copy, rename, allocate, compress, empty, submit, print, and recall. An OLIST is created with a simple command. For example, the **ol dnet187.adlab.*** command creates an OLIST similar to the example in Figure 6 with all the data set names that match that pattern.



```
-IPT- OLIST (E)----------- LEVEL DNET187.ADLAB.* ---------- Row 3 to 19 of 34
Command    ===>                                               SCROLL ===> PAGE
Hotbar: FILT JCL
                                                          *EXEC*TEMPORARY LIST*
TSO PARMS  ===>
Command   Member   Numbr Data Set Names / Objects                        Class
--------- -------- ----- -------------------------------------------- -------
S                      3 'DNET187.ADLAB.CUSTFILE'                       SEQ
                       4 'DNET187.ADLAB.CUST1'
                       5 'DNET187.ADLAB.CUST1.KSDS'                     VSAM
                       6 'DNET187.ADLAB.CUST1.KSDS.DATA'                VSAMDA
                       7 'DNET187.ADLAB.CUST1.KSDS.INDEX'               VSAMIX
                       8 'DNET187.ADLAB.CUST1.XML'
                       9 'DNET187.ADLAB.CUST2SEG'
                      10 'DNET187.ADLAB.DBDLIB'
                      11 'DNET187.ADLAB.DB2.EXPORT'
                      12 'DNET187.ADLAB.DTCMD'
                      13 'DNET187.ADLAB.DTLOG'
                      14 'DNET187.ADLAB.DTPREF'
                      15 'DNET187.ADLAB.DTSF'
                      16 'DNET187.ADLAB.ENTPLI.LISTING'
                      17 'DNET187.ADLAB.EQALANGX'
                      18 'DNET187.ADLAB.EXEC'                           PDSE
                      19 'DNET187.ADLAB.FILES'
```

This is the default action (E = edit)

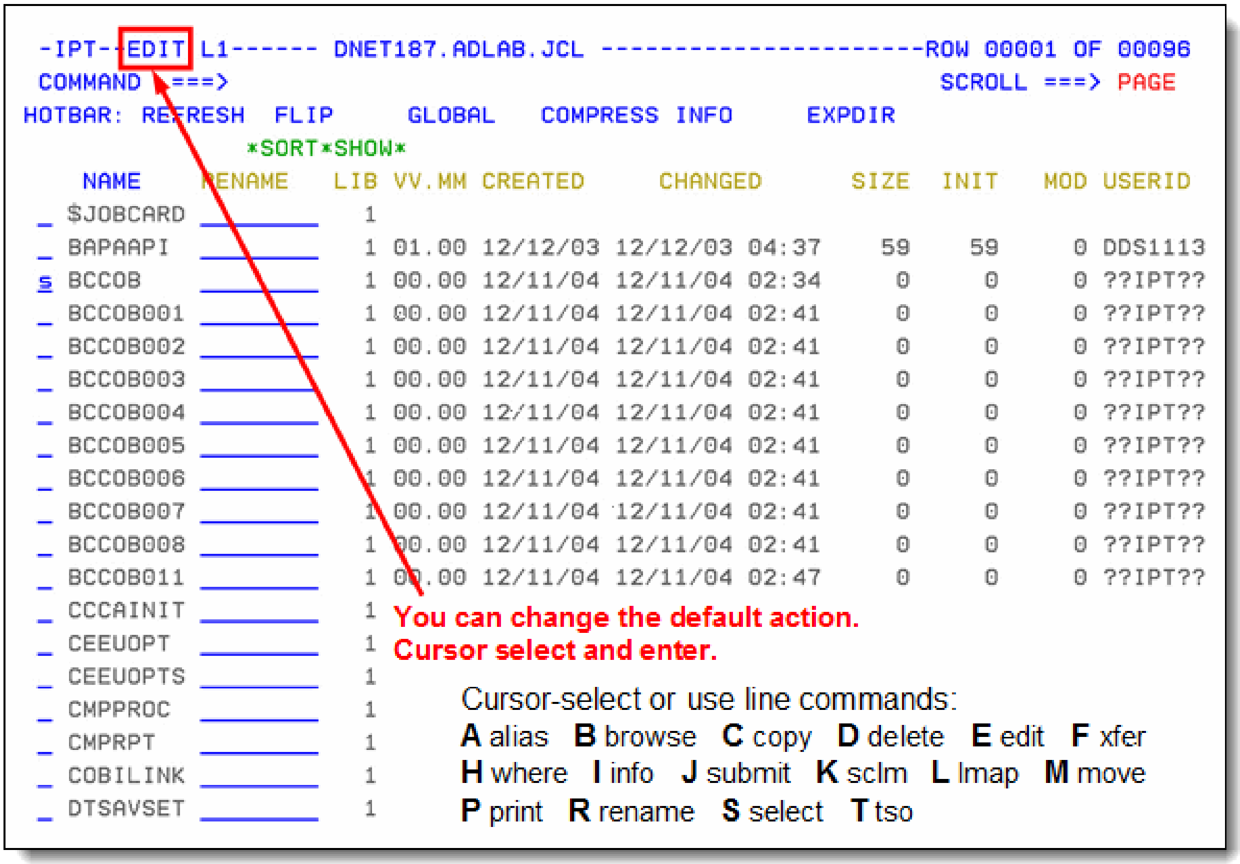
Cursor-select or enter S to access a file in the list

Figure 6. Object list

From the list, you can select an object to perform the default action (in this case edit). Alternatively, you can enter a command, such the browse command (**B**) on the member command line. You can also change the default action. In this example, the list contains VSAM files, which you can open automatically in File Manager if it is installed on the system.

Another key feature is member selection lists. A member selection list is similar to the ISPF member list that is presented when opening a PDS or PDSE. However, a member selection list from the ISPF Productivity Tool provides enhancements that perform complex functions, minimizing the keystrokes and navigation that are required to perform a task.

A member selection list is displayed when you select a PDS or PDSE from an OLIST (Figure 7).



```
 -IPT--EDIT L1------ DNET187.ADLAB.JCL ----------------------ROW 00001 OF 00096
 COMMAND ===>                                                  SCROLL ===> PAGE
HOTBAR: REFRESH  FLIP     GLOBAL   COMPRESS INFO     EXPDIR
               *SORT*SHOW*
    NAME     RENAME    LIB VV.MM CREATED     CHANGED        SIZE  INIT    MOD USERID
 _ $JOBCARD ________    1
 _ BAPAAPI  ________    1 01.00 12/12/03 12/12/03 04:37    59    59      0 DDS1113
 s BCCOB    ________    1 00.00 12/11/04 12/11/04 02:34     0     0      0 ??IPT??
 _ BCCOB001 ________    1 00.00 12/11/04 12/11/04 02:41     0     0      0 ??IPT??
 _ BCCOB002 ________    1 00.00 12/11/04 12/11/04 02:41     0     0      0 ??IPT??
 _ BCCOB003 ________    1 00.00 12/11/04 12/11/04 02:41     0     0      0 ??IPT??
 _ BCCOB004 ________    1 00.00 12/11/04 12/11/04 02:41     0     0      0 ??IPT??
 _ BCCOB005 ________    1 00.00 12/11/04 12/11/04 02:41     0     0      0 ??IPT??
 _ BCCOB006 ________    1 00.00 12/11/04 12/11/04 02:41     0     0      0 ??IPT??
 _ BCCOB007 ________    1 00.00 12/11/04 12/11/04 02:41     0     0      0 ??IPT??
 _ BCCOB008 ________    1 00.00 12/11/04 12/11/04 02:41     0     0      0 ??IPT??
 _ BCCOB011 ________    1 00.00 12/11/04 12/11/04 02:47     0     0      0 ??IPT??
 _ CCCAINIT ________    1
 _ CEEUOPT  ________    1
 _ CEEUOPTS ________    1
 _ CMPPROC  ________    1
 _ CMPRPT   ________    1
 _ COBILINK ________    1
 _ DTSAVSET ________    1
```

You can change the default action.
Cursor select and enter.

Cursor-select or use line commands:
A alias **B** browse **C** copy **D** delete **E** edit **F** xfer
H where **I** info **J** submit **K** sclm **L** lmap **M** move
P print **R** rename **S** select **T** tso

Figure 7. Member selection list

By using a member selection list, you can perform various tasks on the list. For example, you can use filters to select members in the list, use global find and change commands to locate or change data in several or all members of a PDS or PDSE, and perform global edits.

The batch utility is a new component in ISPF Productivity Tool V7. System and application programmers use the batch utility in ISPF Productivity Tool as a productivity aid to manage and manipulate partitioned and sequential data sets. The batch utility in ISPF Productivity Tool is fully compatible with the IBM IEBCOPY utility and provides additional functionality on top of standard IEBCOPY capabilities. For programmers, you can use the batch utility to manage PDS members, including deleting members (and then recovering them), renaming members, and adding alias names to members of a PDS. You can also copy or move members of a PDS and merge them into another PDS or sequential data set. At the data set level for a PDS, you can use the utility to empty and compress data sets and to expand the directory of PDSs. Additional productivity aids are available for sequential data sets. The ISPF Productivity Tool batch utility can also assist in preparing and managing managed lists of member titles in a PDS.

ISPF and ISPF Productivity Tool work together as one product. ISPF Productivity Tool uses hotbars, field-sensitive areas in the OLIST, member selection lists, and 47 shortcut commands to minimize the time to complete the job. By using a single command that you start from an OLIST, you can search multiple files for a member name, data within a member, or data in sequential files from an OLIST. With ISPF Productivity Tool, you can make global changes to PDS or PDSE members. These features are a few of the time-saving features that ISPF Productivity Tool provides.

Usage scenarios

This section highlights the following usage scenarios:

- Organizing your work by using permanent OLISTs
- Managing data sets in a development cycle

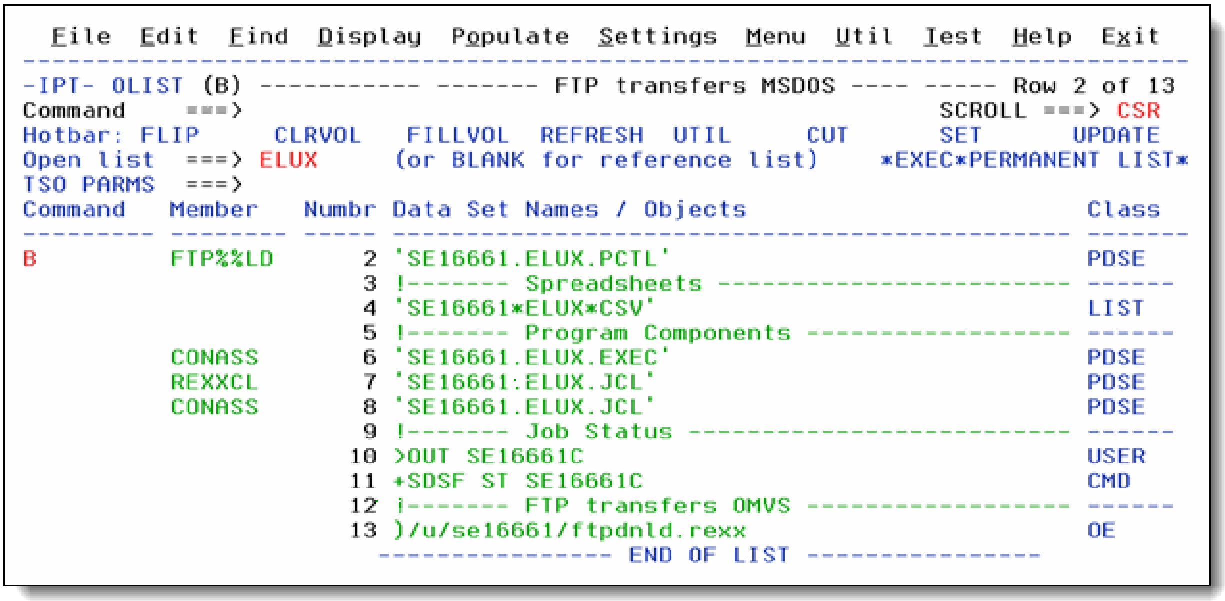
Organizing your work by using permanent OLISTs

This scenario involves an activity that frequently occurs for a development center programmer. The organization's asset manager has several large Microsoft Excel spreadsheets that include such details as assets, contacts, orders, shipments, and deliveries. The asset manager updates and manipulates the spreadsheets to attempt reconciliations. Because the senior auditors were unable to manipulate large volumes by using LOOKUPs VLOOKUPs, they asked the development center programmer for assistance with the complicated cross-matching of data. This request became a periodic on-demand task for this programmer.

The programmer can resolve the problem in several ways. For example, the programmer can use DB2 tables if they are consistent and precise in their use of the spreadsheets, but they are not.

Because this programmer is adept in REXX and has a large processor, the programmer saves and uploads the spreadsheets as comma-separated value (CSV) files and processes them by using compiled REXX programs. Although the file formats, content, and requirements change each time, the activity is similar each time that it is requested. The spreadsheets to be matched (two, three, or sometimes up to six) are sent to the programmer, who uploads them to the host. A new REXX exec is then tailored based on previous examples. Then it is compiled and tested, and any required modifications are applied. The results from the match are downloaded to the PC and then sent by email to the auditors.

The developer creates a new permanent OLIST, shown in Figure 8, that contains the objects and libraries that are needed for the project. The list includes the data sets that are needed for file transfer, the CSV that contains the spreadsheet data, and the data sets that are needed to run the job.



```
  File  Edit  Find  Display  Populate  Settings  Menu  Util  Test  Help  Exit
-------------------------------------------------------------------------------
-IPT- OLIST (B) ------------ ------- FTP transfers MSDOS ---- ----- Row 2 of 13
Command    ===>                                               SCROLL ===> CSR
Hotbar: FLIP     CLRVOL   FILLVOL  REFRESH  UTIL     CUT      SET      UPDATE
Open list  ===> ELUX     (or BLANK for reference list)    *EXEC*PERMANENT LIST*
TSO PARMS  ===>
Command   Member   Numbr Data Set Names / Objects                       Class
--------- -------- ----- ---------------------------------------------- -------
B         FTP%%LD      2 'SE16661.ELUX.PCTL'                            PDSE
                       3 !------- Spreadsheets ------------------------ ------
                       4 'SE16661*ELUX*CSV'                             LIST
                       5 !------- Program Components ------------------ ------
          CONASS       6 'SE16661.ELUX.EXEC'                            PDSE
          REXXCL       7 'SE16661:ELUX.JCL'                             PDSE
          CONASS       8 'SE16661.ELUX.JCL'                             PDSE
                       9 !------- Job Status -------------------------- ------
                      10 >OUT SE16661C                                  USER
                      11 +SDSF ST SE16661C                              CMD
                      12 ¡------- FTP transfers OMVS ------------------ ------
                      13 )/u/se16661/ftpdnld.rexx                       OE
                        ---------------- END OF LIST ----------------
```

Figure 8. Permanent OLIST of the objects used in the project

Now that the process is in place, the asset manager sends the file as an IBM Notes attachment. The following list summarizes the tasks that the programmer completes on the host to process the files:

1. The programmer selects the first data set in the OLIST. The data set contains two members: one with FTP commands to upload the files and one with commands for downloading the files after processing. The programmer selects the member with the upload commands, copies the commands into a command prompt, and then runs the commands.
2. The programmer selects the job control language (JCL) that processes the files from the OLIST and updates the JCL with the names of the uploaded files and new output file names.
3. The programmer selects the compiled JCL from the OLIST and then modifies it to compile the new REXX exec.
4. The programmer selects the EXEC data set from the OLIST and then creates or modifies the code as desired.
5. The programmer selects the JCL from the OLIST that compiles and linkedits the REXX exec and then submits the JCL for execution.
6. The programmer selects the JCL that runs the REXX exec from the OLIST and then submits the JCL for execution.
7. The programmer downloads the output CSV to the PC by using the same process as in step 1.

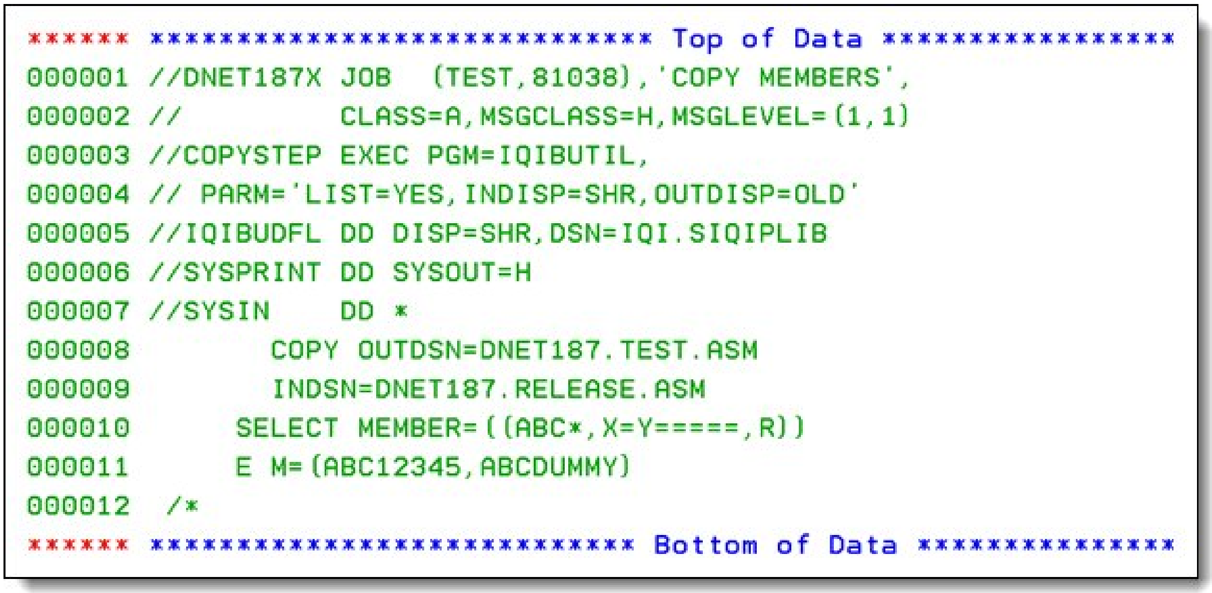
Managing data sets in a development cycle

During the development cycle of a software release, the developer needs to needs to recreate the production environment for testing. This process involves copying many data sets from the production environment to the test environment. Although the developer can use the IEBCOPY utility, the developer recognizes that, over time, unnecessary members were added to the data set and should not be carried forward into the test environment. Fortunately, the ISPF Productivity Tool Batch Utility (IQIBUTIL) extends the IEBCOPY capabilites, allowing a user to copy and replace only selected members of a particular data set. The member name pattern mask support of the utility supports two member name pattern characters:

- A wildcard asterisk (*) represents any string of zero or more characters.
- A wildcard percent sign (%) represents any single character.

With this support, the user can selectively process members in copy and move operations, reducing the steps required to complete this process.

Figure 9 shows the JCL and utility statements to copy one of the data sets to the test environment.

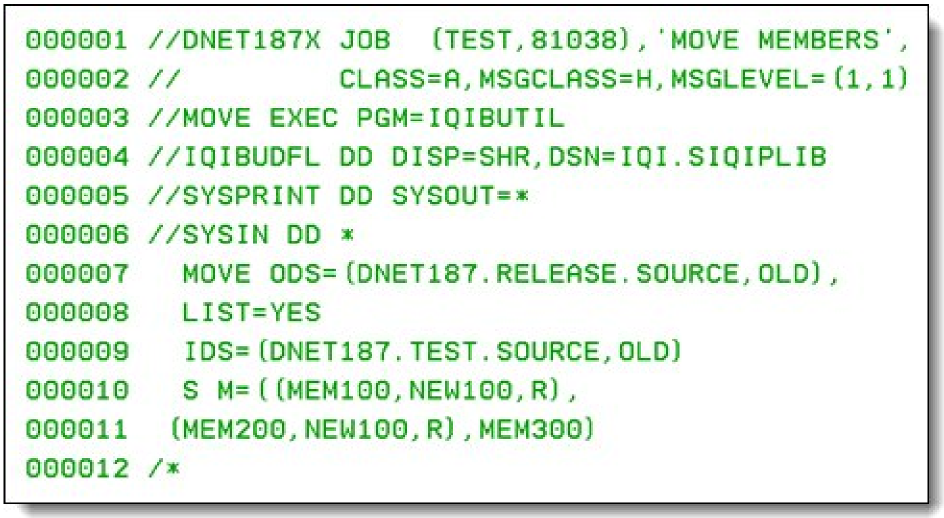


```
***************************** Top of Data *****************
//DNET187X JOB  (TEST,81038),'COPY MEMBERS',
//         CLASS=A,MSGCLASS=H,MSGLEVEL=(1,1)
//COPYSTEP EXEC PGM=IQIBUTIL,
// PARM='LIST=YES,INDISP=SHR,OUTDISP=OLD'
//IQIBUDFL DD DISP=SHR,DSN=IQI.SIQIPLIB
//SYSPRINT DD SYSOUT=H
//SYSIN    DD *
       COPY OUTDSN=DNET187.TEST.ASM
       INDSN=DNET187.RELEASE.ASM
     SELECT MEMBER=((ABC*,X=Y=====,R))
     E M=(ABC12345,ABCDUMMY)
 /*
**************************** Bottom of Data ***************
```

Figure 9. JCL and utility statements to copy one of the data sets to the test environment

In this example, all members with names that begin with ABC, except for members ABC12345 and ABCDUMMY, are copied from the input library, specified by the INDSN parameter to the output library, specified by the OUTDSN parameter. To ensure the data sets in the test environment are kept separate from production, all member names that match the filtering criteria are renamed in the output library, with an X in the first character and a Y in the third character of their original names. The input data set is dynamically allocated with a disposition SHR. The output data set is dynamically allocated with a disposition OLD. The data set disposition is specified in the EXEC parameters. The names of all copied members are reported in the SYSPRINT output data set.

After the test cycle is complete, the development team needs to move the source code for the new release from the test data sets into the production data sets (Figure 10).



```
//DNET187X JOB  (TEST,81038),'MOVE MEMBERS',
//         CLASS=A,MSGCLASS=H,MSGLEVEL=(1,1)
//MOVE EXEC PGM=IQIBUTIL
//IQIBUDFL DD DISP=SHR,DSN=IQI.SIQIPLIB
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  MOVE ODS=(DNET187.RELEASE.SOURCE,OLD),
  LIST=YES
  IDS=(DNET187.TEST.SOURCE,OLD)
  S M=((MEM100,NEW100,R),
 (MEM200,NEW100,R),MEM300)
/*
```

Figure 10. Moving the source code for the new release from the test data sets to the production data sets

As part of the process, the test data sets must be prepared for the test cycle of the next release by deleting the tested members. The move function is not available in the IEBCOPY utility but is supported by the ISPF Productivity Tool Batch Utility. Therefore, the developer uses the utility to move the data set members. Using a move operation instead of a copy eliminates the additional cleanup steps that are required to delete members.

Three members of the source PDS, specified by the IDS keyword, are copied into the target PDS that is specified by the ODS keyword. The same three members are then deleted from the input data set. Both data sets are dynamically allocated when the job is executed. To ensure that the target data set is in the proper state for the move, the utility checks whether member MEM300 is already present in the output data set. If it is present, return code 4 is generated by the MOVE operation, and none of the selected members are deleted.

The process of copying and moving data sets between the test and production environment is repeated many times by the development organization. Using the batch utility ensures that the process is done quickly and efficiently. It also ensures that the test and production environments remain separate.

Integration

The IBM ISPF Productivity Tool extends the functions that are provided by ISPF to other data set objects, including VSAM files, Software Configuration and Library Manager (SCLM) libraries, Librarian or Panvalet libraries, z/OS UNIX System Services, PC files, DB2 tables, and other object classes.

By using the ISPF Productivity Tool customization wizard, you can define the CLIST that is used by the various products. Then, you can access the following files directly from an OLIST:

- VSAM files:
    - IBM File Manager for z/OS
    - Other third-party products
- Software configuration management systems:
    - IBM SCLM
    - Other third-party products
- DB2 tables:
    - IBM File Manager or DB2 for z/OS (with ISPF Productivity Tool V5.10)
    - Other third-party products

ISPF Productivity Tool is also a valuable tool for users who are adopting IBM Rational® Developer for System z. Often, much of a programmer's time is consumed not by actual programming, but by keeping track of data sets, libraries, and members that they work with on a daily or hour-by-hour basis. For these highly specific tasks, users can use the Rational Developer for System z Host Connection Emulator to perform the following tasks:

- Log on to z/OS directly and manipulate JCL
- Compress or empty PDS
- Search across concatenated libraries
- Perform various other tasks in support of the System z environment

By installing ISPF Productivity Tool, users have all the additional capabilities of the tool and can still enjoy all the benefits of the Rational Developer for System z GUI. For more information about Rational Developer for System z, go to:
http://www.ibm.com/software/awdtools/rdz

Supported platforms

IBM ISPF Productivity Tool can be installed in any supported release of z/OS with ISPF installed. For more information about the system requirements, see the ISPF Productivity Tool page at:
http://www.ibm.com/software/awdtools/ispfproductivitytool/prereqs

Ordering information

Table 1 highlights the ordering information.

Table 1. Ordering part numbers and feature codes

Program name	PID number	Charge unit description
ISPF Productivity Tool for z/OS, V7.1	5655-SPF	Value Unit pricing
ISPF Productivity Tool for z/OS Subscription & Support	5698-S89	Value Unit pricing

Related information

For more information, see the following documents:

- *Improving Productivity with IBM ISPF Productivity Tool V7.1*, SG24-8093
  http://www.redbooks.ibm.com/abstracts/sg248093.html
- ISPF Productivity Tool for z/OS product page
  http://www.ibm.com/software/awdtools/ispfproductivitytool/index.html
- IBM Offering Information page (to search on announcement letters, sales manuals, or both):
  http://www.ibm.com/common/ssi/index.wss?request_locale=en

  On this page, enter `ISPF Productivity Tool for z/OS V7.1`, select the information type, and then click **Search**. On the next page, narrow your search results by geography and language.

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service. IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you. This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products. This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

This document was created or updated on February 11, 2013.

Send us your comments in one of the following ways:

- Use the online **Contact us** review form found at:
  ibm.com/redbooks
- Send your comments in an e-mail to:
  redbook@us.ibm.com
- Mail your comments to:
  IBM Corporation, International Technical Support Organization
  Dept. HYTD Mail Station P099
  2455 South Road
  Poughkeepsie, NY 12601-5400 U.S.A.

This document is available online at http://www.ibm.com/redbooks/abstracts/tips0936.html .

Trademarks