ISPF
An Experienced User Shares His Secrets

Bruce Koss
Wells Fargo

Tuesday, August 13, 2015
3:15 PM - 4:15 PM
17412

www.SHARE.org

Introduction

- Provide an overview on how to exploit several ISPF features
  - ISPF configuration
  - Stacking commands
  - MODEL command
  - Work Station Agent (WSA)

- Expectations
  - Familiarity with ISPF configuration and MODEL command
  - Capability to allocate personal ISPF data sets to allow testing

- Other Items
  - Links are provided to IBM manuals and other documents to assist in learning these features in more detail

Agenda

Link	Topic
ISPF Configuration	• Settings
Stacking Commands	• Overview and implementation
MODEL Command	• Obtain ISPF commands • Add JCL utilities, SDSF REXX and REXX commands
Work Station Agent (WSA)	• Overview and installation • Transfer files using ISPF 3.7.2
Links	• ISPF, REXX, TSO and SDSF manuals • SHARE ISPF, SDSF and REXX PPTs • Websites

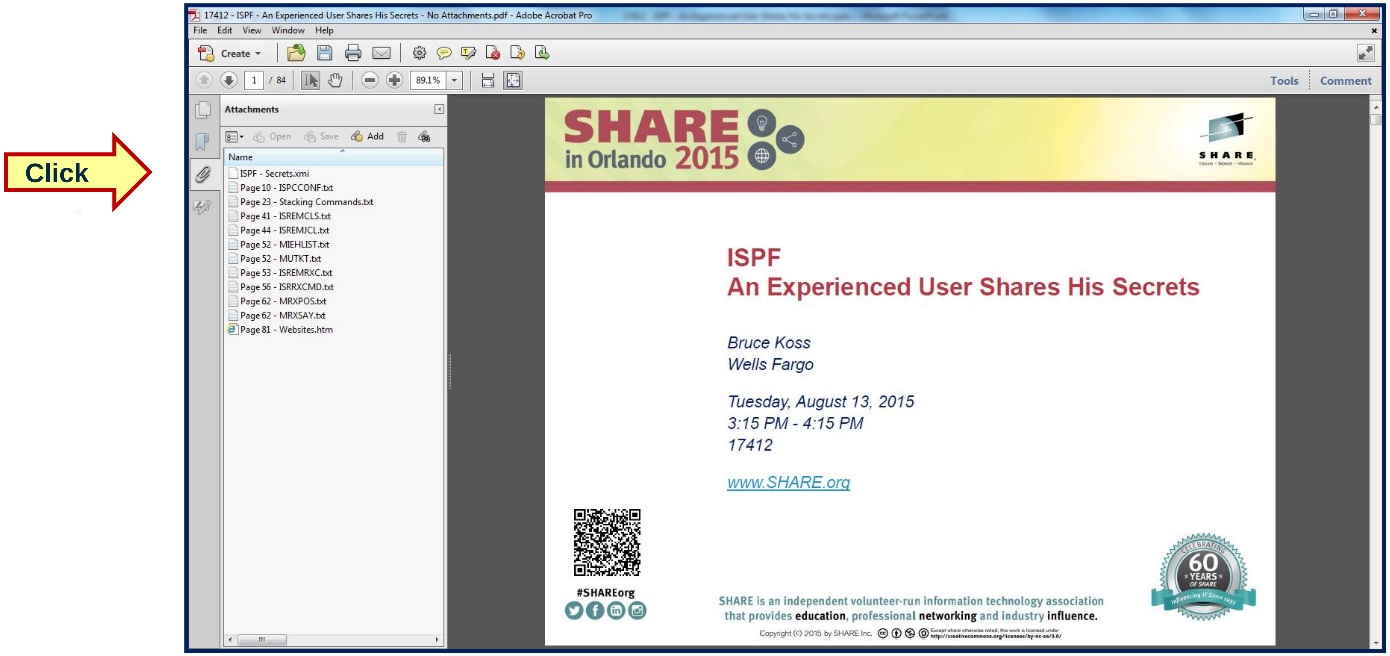
Attachments (📎)

- The following attachments are included in this document:

File	Type
ISPF – Secrets	XMI
Page 10 – ISPCCONF	TXT
Page 23 – Stacking Commands	TXT
Page 41 – ISREMCLS	TXT
Page 44 – ISREMJCL	TXT
Page 52 – MIEHLIST	TXT
Page 52 – MUTKT	TXT
Page 53 – ISREMRXC	TXT
Page 56 – ISRRXCMD	TXT
Page 62 – MRXPOS	TXT
Page 62 – MRXSAY	TXT
Page 81 – Websites	HTML

Attachments ( )

- Invoke Adobe Reader and click the ‘Attachments’ button on the left side of the navigational panel of the main Reader window



Click
17412 - ISPF - An Experienced User Shares His Secrets - No Attachments.pdf - Adobe Acrobat Pro
File Edit View Window Help
Create
1 / 84
89.1%
Tools
Comment
Attachments
Open
Save
Add

Name

- ISPF - Secrets.xmi
- Page 10 - ISPCCONF.txt
- Page 23 - Stacking Commands.txt
- Page 41 - ISREMCLS.txt
- Page 44 - ISREMJCL.txt
- Page 52 - MIEHLIST.txt
- Page 52 - MUTKT.txt
- Page 53 - ISREMRXC.txt
- Page 56 - ISRRXCMD.txt
- Page 62 - MRXPOS.txt
- Page 62 - MRXSAY.txt
- Page 81 - Websites.htm

SHARE in Orlando 2015

ISPF
An Experienced User Shares His Secrets

Bruce Koss
Wells Fargo

Tuesday, August 13, 2015
3:15 PM - 4:15 PM
17412

www.SHARE.org

#SHAREorg

SHARE is an independent volunteer-run information technology association that provides **education**, professional **networking** and industry **influence.**

Copyright (c) 2015 by SHARE Inc. Except where otherwise noted, this work is licensed under http://creativecommons.org/licenses/by-nc-sa/3.0/

ISPF Secrets (📎)

- The ‘ISPF – Secrets.xmi’ file is a TSO TRANSMIT (XMIT) file
  - Contains all members referenced in this slide deck

- Upload this file as binary and LRECL=80 to the mainframe

- Use the JCL below to un-compress the file

```
//TSOBATCH EXEC PGM=IKJEFT01,REGION=0M
//SYSTSPRT DD  SYSOUT=*
//SYSTSIN  DD  *
  RECEIVE INDA('uploaded_xmit_file')
          DA('output_file_to_uncompress_members')
```

Disclaimers

- Must use Adobe Reader to view attachments
  - (📎) signifies attachment
- May need to adjust the Adobe Reader zoom value to view text
- Commands and screens may be slightly different depending on the operating system's version and release level
  - These commands and screen prints were executed on z/OS V2R1
- Security may prevent access to panels or executing commands

ISPF
Configuration

Introduction

- Provide an overview of several ISPF configuration variables
  - Activate new features and enhancements
  - For new users, reduce the time it takes to configure the environment

- ISPF Configuration Dialog
  - Execute ISPCCONF after installing a new z/OS operating system
  - Syntax:

    ➔ TSO ISPCCONF

- Links
  - SHARE, Tom Conley, Configuring ISPF for Fun and Profit
  - IBM Publication, ISPF Planning and Customization, Chapter 2

- Items highlighted in yellow are settings used to implement stacking commands and Work Station Agent (WSA)

ISPCCONF (📎)

Variable	Assigned	Notes
APPLID_FOR_SITE_COMMAND_TABLE	WFS	Allows for system/site wide command tables to be defined/used (WFSCMDS) - WFS stands for Wells Fargo Site commands - The default is NONE
APPLID_FOR_USER_COMMAND_TABLE	WFU, WFU2, WFU3	Allows users to define up to three different user command tables (WFUCMDS, WFU2CMDS and WFU3CMDS) - WFU stands for Wells Fargo User commands - The default is NONE
BLOCK_SIZE_FOR_TEMPORARY_CNTL_DATA_SETS BLOCK_SIZE_FOR_TEMPORARY_LIST_DATA_SETS BLOCK_SIZE_FOR_TEMPORARY_WORK_DATA_SETS	0 0 0	Set the block size to zero to obtain/use half track blocking - The defaults are 800, 3146 and 2560
COMMAND_LINE_PLACEMENT	ASIS	Place the command line at the top - The default is BOTTOM
DEFAULT_LIBDEF_PROCESSING_OPTION	STACK	When LIBDEFing data sets, the default will be STACK - Prevents issues when stacking software on top of one another, specifically, removing previous LIBDEFed data sets - The default is UNCOND
DEFAULT_SYSTEM_NAME	ON	Displays the lpar name in the upper left hand corner of all panels - The default is OFF

ISPCCONF (📎)

Variable	Assigned	Notes
DISALLOW_WILDCARDS_IN_HLQ	YES	Prevents long enqueues on the master catalog - The default is NO
DISPLAY_EMPTY_MEMBER_LIST DISPLAY_EMPTY_MEMBER_LIST_PATTERN	YES YES	Capability to edit an empty PDS - The defaults are NO and NO
DISPLAY_SEQUENCE_NUMBERS	OFF	- The default is ON
DSLIST_RM_ENABLED DSLIST_RM_COMMAND	YES %CA1DS34	Allows executing a REXX program to obtain CA1 information when typing an “I”, “S” or “M” next to a tape data set from ISPF 3.4 - CA1DS34 is an in-house written utility -- “I” invokes CA1, selects option “1 - INQ/UPD” and inserts the VOLSER to obtain the tape information -- “S” invokes CA1's Browse Module (TSOTIQ) and uses the data set name to obtain the tape information -- “M” invokes CA1's Tape Inquiry Program (TMSBINQ) and uses the VOLSER to obtain the tape information - The defaults are NO and %EDGRPD34
EDIT_RECOVERY_BLOCK_SIZE EDIT_RECOVERY_PRIMARY_QUANTITY EDIT_RECOVERY_SECONDARY_QUANTITY	27998 800 800	Allows for a large amount of changes and updates - The defaults are 13680, 40 and 200

ISPCCONF (📎)

Variable	Assigned	Notes
HILITE_DOLOGIC HILITE_IFLOGIC HILITE_PAREN	ON ON ON	Distinguishes lines of code and assists in troubleshooting - The defaults are OFF, OFF and OFF
ISPCTL0_BLOCK_SIZE	0	Set the block size to zero to obtain/use half track blocking - The default is 800
ISPF_TEMPORARY_DATA_SET_QUALIFIER	&SYSNAME.	Inserts the lpar name in the second level qualifier of the ISPF temporary data set names - Allows logging on to multiple, concurrent TSOs within a sysplex - If the lpar name begins with a number, suggest placing the character 'S' in front of &SYSNAME - The default is NONE
LIST_DATA_SET_RECORDS_PER_BLOCK	231	Set the list data set records per block to 231 for a 3390 - LRECL 121 * 231 = 27951, half-track for a 3390 - The default is 26
LIST_SYSOUT_CLASS	*	- The default is NONE
LOCAL_LIST_SYSOUT_CLASS	*	- The default is NONE

ISPCCONF (📎)

Variable	Assigned	Notes
LOG/LIST_JOB_CARD1	JOB Card	Define a standard JOB card in ISPF - Option '0 – Settings', Log/List and '4 – JCL' - Software can obtain and use it by coding "VGET (ZLLGJOB1 ZLLGJOB2 ZLLGJOB3 ZLLGJOB4)" - The default is NONE
LOG_DATA_SET_BLOCK_SIZE	0	Set the block size to zero to obtain/use half track blocking - The default is 129
LOG_DATA_SET_DISPOSITION	D	Delete the log data set - The default is NONE
LOG_PAGES_PRIMARY_QUANTITY LOG_PAGES_SECONDARY_QUANTITY	0 0	Not needed/used, decreases response time - The defaults are 10 and 10
MAXIMUM_NUMBER_OF_SPLIT_SCREENS	32	Allows the capability to have 32 split screens - Set PF2=START, PF9=SWAP NEXT and PF21=SWAP PREV; then, activate SWAPBAR - The default is 8
NUMBER_OF_PFKEYS	24	Typing in KEYS will display "Number of Keys = 24" - The default is 12
OUTLIST_BLOCK_SIZE	0	Set the Outlist Utility block size to zero to obtain half track blocking for OUTLIST data sets - The default is 13566

ISPCCONF (📎)

Variable	Assigned	Notes
PROFILE_SHARING	YES	Allows logging on to multiple, concurrent TSOs within a sysplex - Required to run multiple, concurrent z/OSMF ISPF sessions - The default is NO
RESET_WSA_DOWNLOAD_DATA_SET	YES	Resets the WSA download data set
RETREIVE_COMMAND_STACK_SIZE	4096	Allows saving/retrieving longer primary commands - The default is 512
SCREEN_FORMAT	MAX	Automatically adjusts screen lines/columns to 3270 emulator terminal type (Model 2, 3, 4 and 5) - The default is STD
SCROLL_DEFAULT	CSR	Set the scroll default to cursor - The default is PAGE
SHOW_PFKEYS	OFF	Provides for more screen real estate - The default is ON
SITE_WIDE_INITIAL_MACRO	RESET	Prevents unnecessary messages from appearing when editing a data set or member - The default is NONE
STANDARD_NUMBERS	OFF	- The default is ON

ISPCCONF (📎)

Variable	Assigned	Notes
STATS_EXT_ENABLED	YES	Extended statistics are kept for PDS members - The default is NO
TAB_TO_ACTION_BARS	NO	Pressing the HOME key will position the cursor on the primary command line - The default is YES
UNDO_STORAGE_SIZE	1024	Allocate 1024K of memory to edit UNDO command to be used for keeping history of edit changes - Enables SETUNDO command - The default is 0
USE_KEYLISTS	NO	Makes the PF keys standard throughout ISPF - Assigns PF4=RETURN, PF5=RFIND, PF6=RCHANGE, P7=UP, PF8=DOWN, PF9=SWAP, PF10=LEFT, PF11=RIGHT and PF12=RETRIEVE - The default is YES
VSAM_BROWSE_ENABLED VSAM_BROWSE_COMMAND VSAM_EDIT_ENABLED VSAM_EDIT_COMMAND VSAM_VIEW_ENABLED VSAM_VIEW_COMMAND	YES %ISPF34B YES %ISPF34E YES %ISPF34B	Allows invoking a REXX program when browsing, editing or viewing a VSAM file - ISPF34B/E is an in-house written REXX program to invoke File Manager/MVS (FM1/2) or CBT183 (BR) - Allows the ability to invoke and use Ditto, File-Aid/MVS, File Manager/MVS, Review or CBT Tape software - The defaults are NO and FMNMAIN DSB /

ISPCCONF (📎)

Variable	Assigned	Notes
WSA_DOWNLOAD_DATA_SET	'SYS1.SISPGUI'	The data set name is inserted into the ISPF 3.7.1 screen - The default is 'ISP.SISPGUI'

Stacking Commands

Introduction

- Stacking commands provide the capability to invoke any application from anywhere within ISPF and not have to be located at the user or main menu, reducing keystrokes

- Allows multiple software products to be stacked on one split screen and each split screen is a separate stack

- Stacking commands are implemented using the ISPF site and user command tables (slide 10)

Syntax

- All menu options are stacking commands

- IBM Primary Menu
  - Prefix the character “Z” in front of the option (Z##)
  - Examples:
    - Z1 will invoke ISPF option 1, Browse
    - Z2 will invoke ISPF option 2, Edit
    - Z34 will invoke ISPF option 3.4, Data Set List
    - ZSD will invoke ISPF option SD, SDSF

- Developer and System Menu
  - Type in the entry name
    - CA7, ABENDAID, NDM, OPSMVS, etc.
  - Only need to type in the 1st three or four characters
    - ABE for ABENDAID
    - OPS for OPSMVS

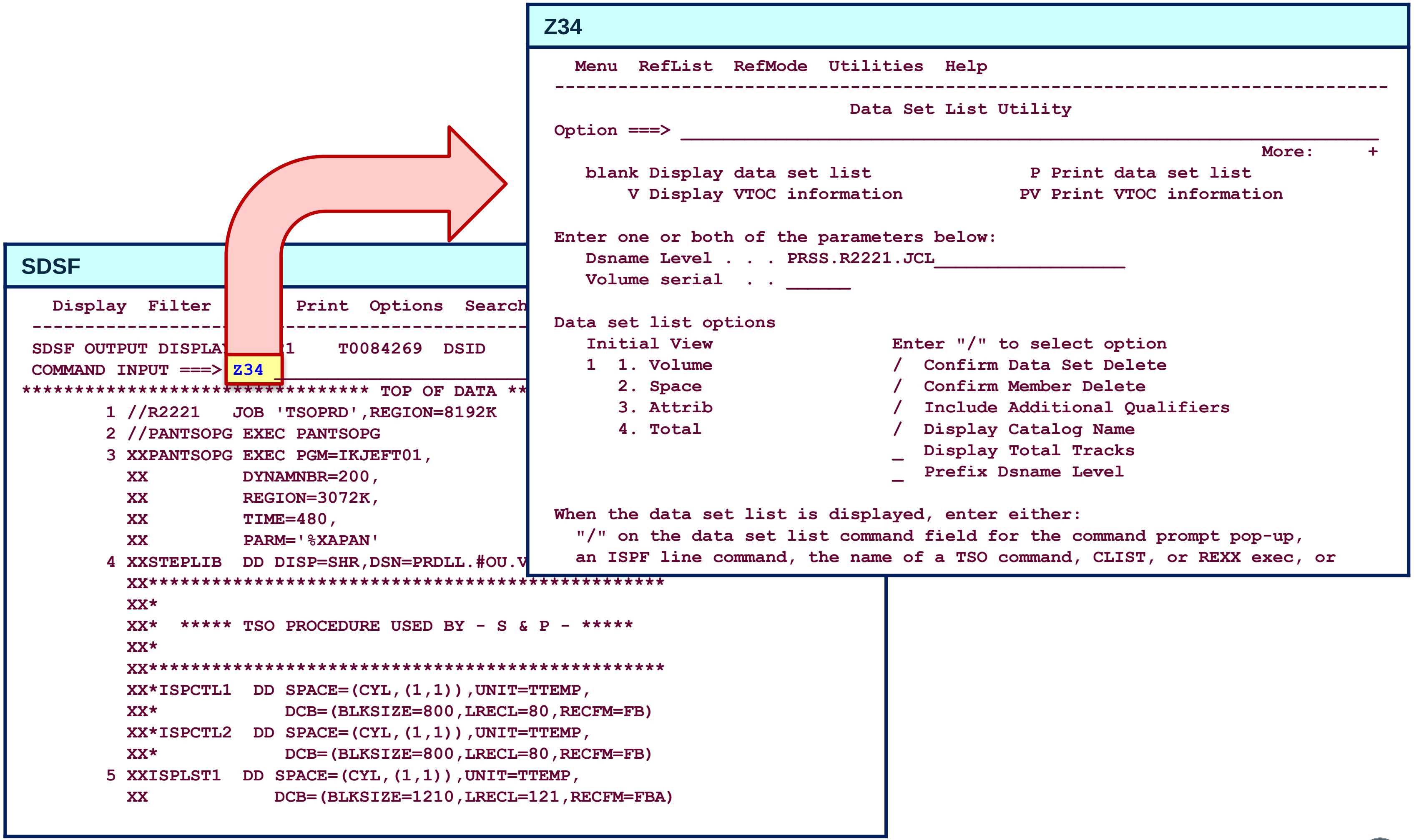
How does it work?

- A user in SDSF may have a need to invoke ISPF 3.4, Data Set List Utility, for a quick look at a data set

- By simply keying in Z34 on the command line, ISPF 3.4 is stacked on top of the current dialog, without interrupting the SDSF session or starting up a new logical split screen

- The END or RETURN PFkeys (PF3/4) will cause an immediate return to the suspended dialog, in this case, SDSF

Example



Z34

```
   Menu  RefList  RefMode  Utilities  Help
 ------------------------------------------------------------------------------
                          Data Set List Utility
 Option ===> __________________________________________________________________
                                                                More:     +
    blank Display data set list                 P Print data set list
        V Display VTOC information             PV Print VTOC information

 Enter one or both of the parameters below:
    Dsname Level . . . PRSS.R2221.JCL__________________
    Volume serial  . . ______

 Data set list options
    Initial View                     Enter "/" to select option
    1  1. Volume                     /  Confirm Data Set Delete
       2. Space                      /  Confirm Member Delete
       3. Attrib                     /  Include Additional Qualifiers
       4. Total                      /  Display Catalog Name
                                     _  Display Total Tracks
                                     _  Prefix Dsname Level

 When the data set list is displayed, enter either:
   "/" on the data set list command field for the command prompt pop-up,
   an ISPF line command, the name of a TSO command, CLIST, or REXX exec, or
```

SDSF

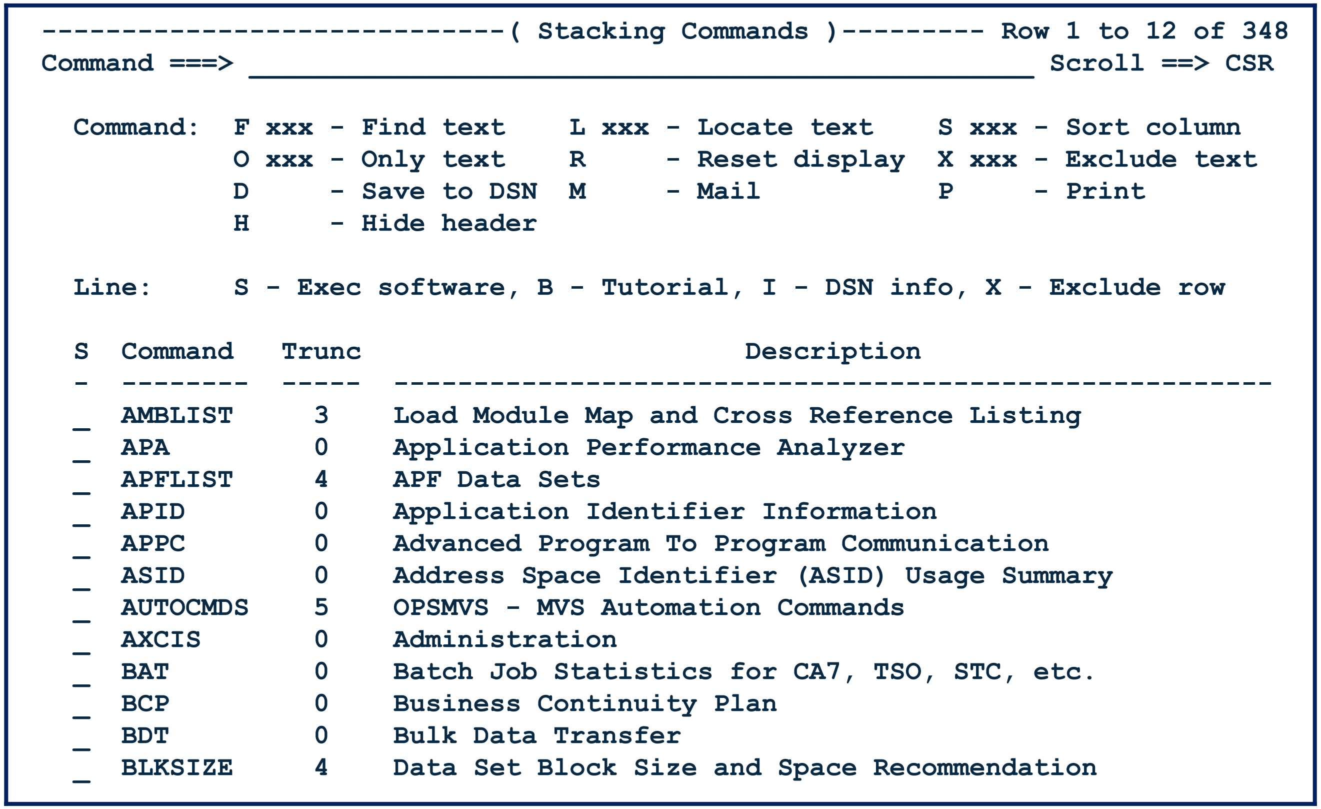
```
    Display  Filter       Print  Options  Search
  ------------------     ------------------------
  SDSF OUTPUT DISPLA     1    T0084269  DSID
  COMMAND INPUT ===> Z34
 ********************************* TOP OF DATA **
         1 //R2221    JOB 'TSOPRD',REGION=8192K
         2 //PANTSOPG EXEC PANTSOPG
         3 XXPANTSOPG EXEC PGM=IKJEFT01,
           XX          DYNAMNBR=200,
           XX          REGION=3072K,
           XX          TIME=480,
           XX          PARM='%XAPAN'
         4 XXSTEPLIB  DD DISP=SHR,DSN=PRDLL.#OU.V
           XX*************************************************
           XX*
           XX*  ***** TSO PROCEDURE USED BY - S & P - *****
           XX*
           XX*************************************************
           XX*ISPCTL1  DD SPACE=(CYL,(1,1)),UNIT=TTEMP,
           XX*            DCB=(BLKSIZE=800,LRECL=80,RECFM=FB)
           XX*ISPCTL2  DD SPACE=(CYL,(1,1)),UNIT=TTEMP,
           XX*            DCB=(BLKSIZE=800,LRECL=80,RECFM=FB)
         5 XXISPLST1  DD SPACE=(CYL,(1,1)),UNIT=TTEMP,
           XX           DCB=(BLKSIZE=1210,LRECL=121,RECFM=FBA)
```

STACK – Utility

- An in-house utility called STACK or STK was developed to display all available stacking commands
- Displays all available commands defined in the site (WFSCMDS) or user (WFU[2|3]CMDS) command tables
- Lists all entries on the Developers, System and ISPF menus, plus TSO commands like PING, NETSTAT, etc.
- An easy way to locate, execute and obtain product information

STACK – Example (📎)

- Typing in STACK or STK from any panel displays all stacking commands



```
-----------------------------( Stacking Commands )--------- Row 1 to 12 of 348
Command ===> _________________________________________________ Scroll ==> CSR

  Command:  F xxx - Find text     L xxx - Locate text     S xxx - Sort column
            O xxx - Only text     R     - Reset display   X xxx - Exclude text
            D     - Save to DSN   M     - Mail            P     - Print
            H     - Hide header

  Line:     S - Exec software, B - Tutorial, I - DSN info, X - Exclude row

  S  Command   Trunc                      Description
  -  --------  -----  -------------------------------------------------------
  _  AMBLIST     3    Load Module Map and Cross Reference Listing
  _  APA         0    Application Performance Analyzer
  _  APFLIST     4    APF Data Sets
  _  APID        0    Application Identifier Information
  _  APPC        0    Advanced Program To Program Communication
  _  ASID        0    Address Space Identifier (ASID) Usage Summary
  _  AUTOCMDS    5    OPSMVS - MVS Automation Commands
  _  AXCIS       0    Administration
  _  BAT         0    Batch Job Statistics for CA7, TSO, STC, etc.
  _  BCP         0    Business Continuity Plan
  _  BDT         0    Bulk Data Transfer
  _  BLKSIZE     4    Data Set Block Size and Space Recommendation
```

Pros & Cons

- Pros
  - Enhances productivity in ISPF
  - Allows invoking multiple applications on one screen and on top of one another
  - The "Stacking" capability is limited only by the amount of TSO virtual storage
  - Saves typing and reduces the amount of time needed to invoke the facility
  - Typing in “=??” will remove the entire stack
    - Where ?? is an ISPF menu entry
    - Examples: =3.4, =D.ABENDAID, =X, etc.

- Cons
  - When two or more applications are stacked on the same split screen
    - Only one, the application on top of the stack, is accessible
    - Applications in the middle of the stack are inaccessible

Implementation – ISPCCONF

- ISPF Configuration Utility
  - Type in
    - ➔ TSO ISPCCONF
  - Define and set
    - APPLID_FOR_SITE_COMMAND_TABLE = WFS
    - APPLID_FOR_USER_COMMAND_TABLE = WFU, WFU2, WFU3
    - DEFAULT_LIBDEF_PROCESSING_OPTION = STACK

- The user command table allows programmers to create their own personal or team stacking commands
  - WFUCMDS, WFU2CMDS and/or WFU3CMDS

- Ensure ISPCFIGU module, created by ISPCCONF, is located in
  - LPA or LNK List
  - or ISPLLIB (STEPLIB) allocated to TSO logon proc

Implementation – ISPF 3.9

- The next few slides will demonstrate how to create a site command table (WFSCMDS) using ISPF 3.9, Commands

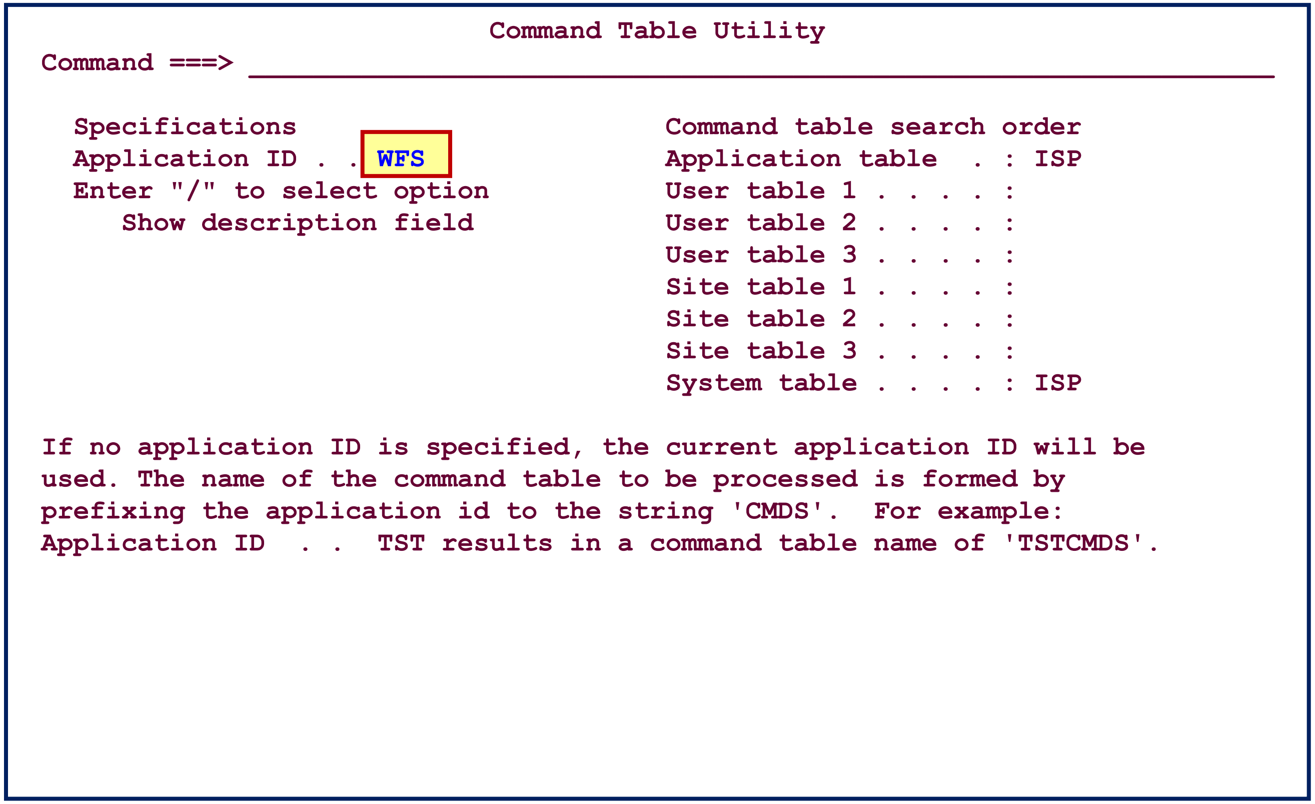
- Use the same process to create user command tables (WFUCMDS, WFU2CMDS and WFU3CMDS)

- Prerequisite
  - Must allocate a PDS or PDSE data set to ISPTABL DD card
    - This data set is where the site command table will be saved (WFSCMDS)
    - Preferably a data set concatenated to the ISPTLIB DD card
      - Allows easy testing and validating of site commands
    - The data set attributes must be RECFM=FB,LRECL=80

- Highlighted in yellow is data typed in on the ISPF 3.9, Commands, panel

Implementation – ISPF 3.9

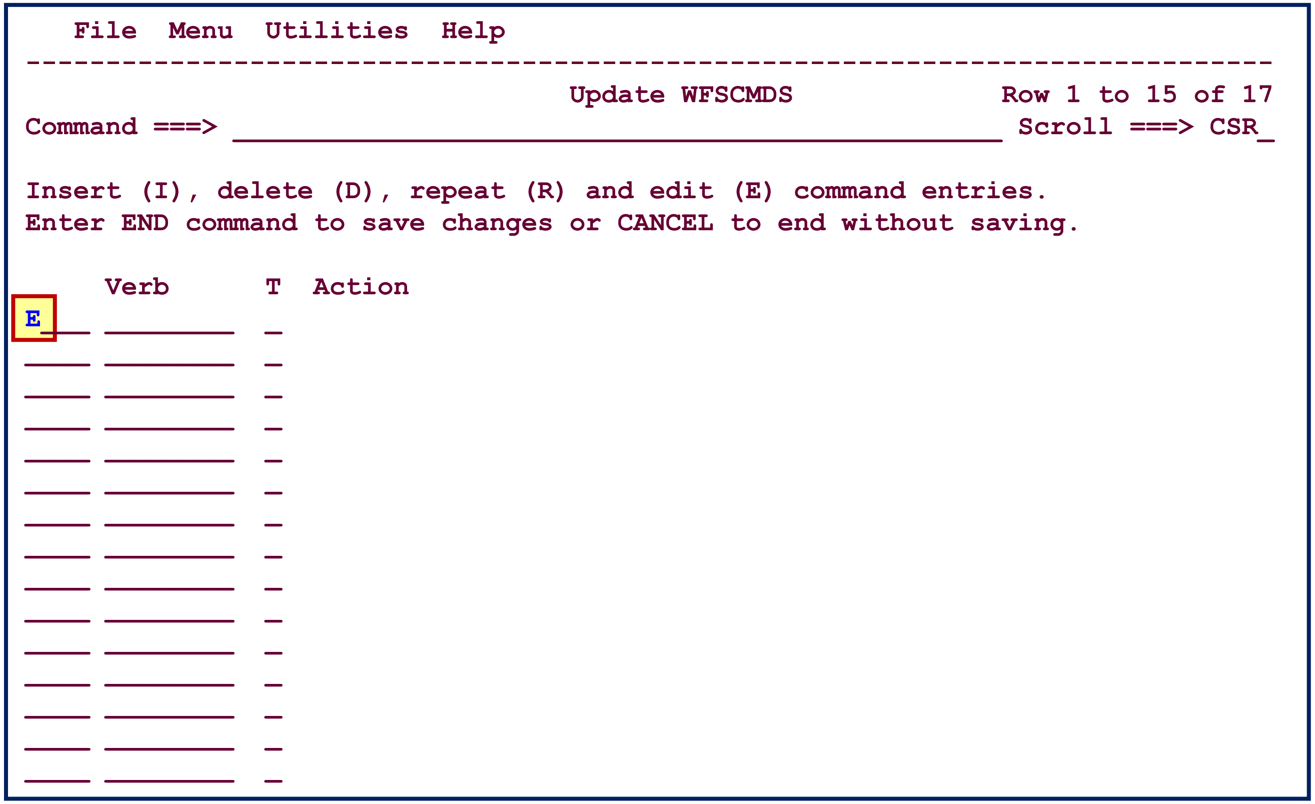
- Invoke ISPF 3.9 and insert the site command table name (WFS)



```
                           Command Table Utility
Command ===> ________________________________________________________________

  Specifications                       Command table search order
  Application ID . . WFS               Application table  . : ISP
  Enter "/" to select option           User table 1 . . . . :
     Show description field            User table 2 . . . . :
                                       User table 3 . . . . :
                                       Site table 1 . . . . :
                                       Site table 2 . . . . :
                                       Site table 3 . . . . :
                                       System table . . . . : ISP

If no application ID is specified, the current application ID will be
used. The name of the command table to be processed is formed by
prefixing the application id to the string 'CMDS'.  For example:
Application ID  . .  TST results in a command table name of 'TSTCMDS'.
```

Implementation – ISPF 3.9

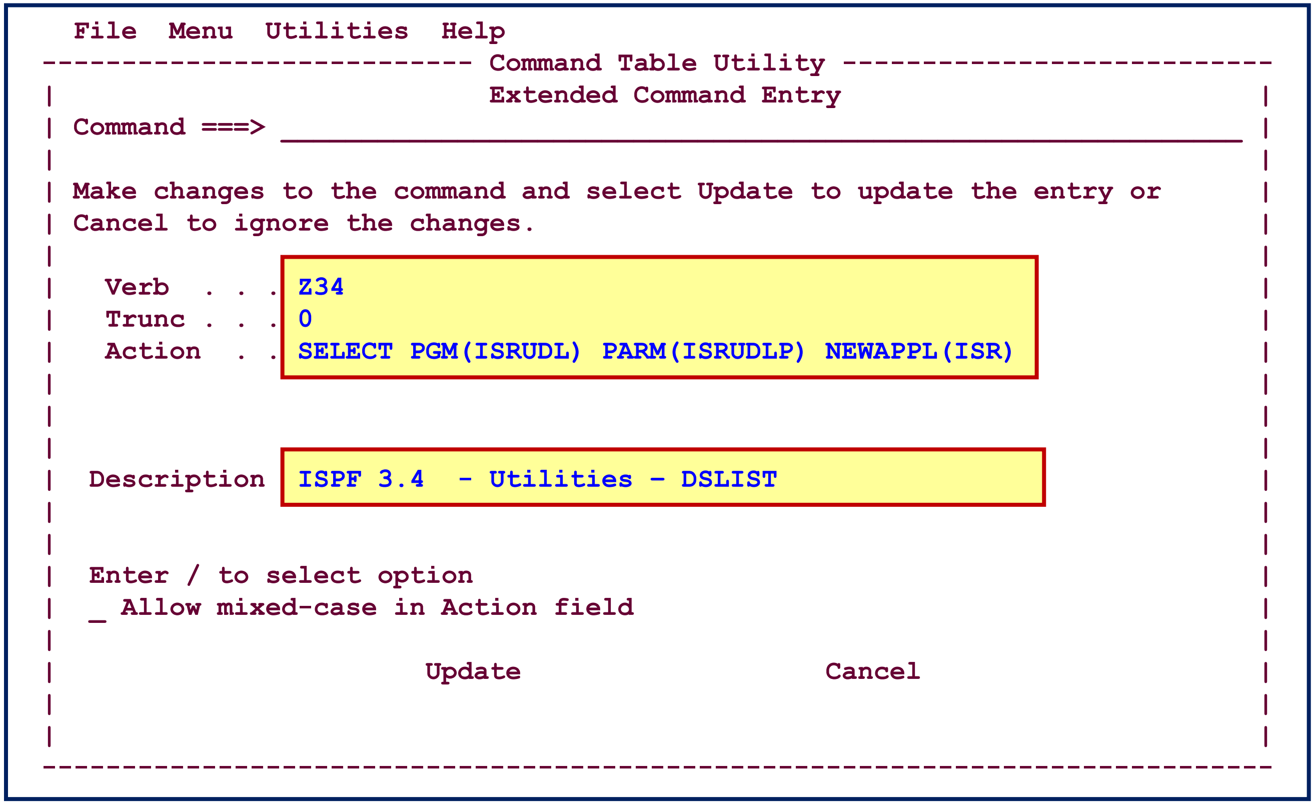
- Tab down to an entry and type in “E” for edit



```
   File  Menu  Utilities  Help
------------------------------------------------------------------------------
                                  Update WFSCMDS              Row 1 to 15 of 17
Command ===> ________________________________________________ Scroll ===> CSR_

Insert (I), delete (D), repeat (R) and edit (E) command entries.
Enter END command to save changes or CANCEL to end without saving.

     Verb      T  Action
E___ ________  _
____ ________  _
____ ________  _
____ ________  _
____ ________  _
____ ________  _
____ ________  _
____ ________  _
____ ________  _
____ ________  _
____ ________  _
____ ________  _
____ ________  _
____ ________  _
____ ________  _
```

Implementation – ISPF 3.9

- Insert the Verb (stacking command), the Trunc (number of characters needed), the Action (what to execute/invoke) and the Description



```
  File  Menu  Utilities  Help
-------------------------- Command Table Utility --------------------------
|                          Extended Command Entry                         |
| Command ===> ___________________________________________________________ |
|                                                                         |
| Make changes to the command and select Update to update the entry or    |
| Cancel to ignore the changes.                                           |
|                                                                         |
|   Verb  . . . Z34                                                       |
|   Trunc . . . 0                                                         |
|   Action  . . SELECT PGM(ISRUDL) PARM(ISRUDLP) NEWAPPL(ISR)             |
|                                                                         |
|                                                                         |
|                                                                         |
|  Description  ISPF 3.4  - Utilities - DSLIST                            |
|                                                                         |
|                                                                         |
|  Enter / to select option                                               |
|  _ Allow mixed-case in Action field                                     |
|                                                                         |
|                     Update                   Cancel                     |
|                                                                         |
|                                                                         |
---------------------------------------------------------------------------
```

Implementation – ISPF 3.9

- VERB
  - Stacking command
  - Must follow IBM's naming convention
    - 1st char = A-Z $#@ and 2nd-8th char = A-Z 0-9 $#@
  - Must be 2 to 8 characters in length
- TRUNC
  - The minimum number of characters the user must type
  - If zero or equal to the verb length, the entire command must be entered
- ACTION
  - SELECT command, program or selection panel to be given control
- Description
  - Brief description/purpose of command
  - 80 characters or less

Implementation – ISPF 3.9

- Repeat the process for all menu entries
- Terminate/re-invoke ISPF and test typing in a stacking command
- Copy the site command table (WFSCMDS) into an ISPTLIB data set accessible to everyone
- To update the active site command table (WFSCMDS)
  - Make a copy (ex: WFSNCMDS)
  - Use ISPF 3.9 to make changes (WFSNCMDS)
  - Copy with replace (WFSNCMDS → WFSCMDS)
  - Terminate/re-invoke ISPF and validate

Suggestions

- Always define site commands with three or more characters
  - Stacking commands take precedence over menu options
  - Prevents menu conflicts

- Prefix the character “Z” in front of all IBM Primary Menu options
  - Very few selection menus use the character Z and a number (Z1, Z2, etc.)

- Ensure software LIBDEFing data sets use sub parameter STACK
  - Change default LIBDEF processing option to STACK in the site command table (Slide 10)
  - Do not use LIBDEF sub parameter UNCOND
    - Prevents removing previous LIBDEFed data sets

Suggestions

- To prevent S878 abends, due to several split screens and stacks
  - Increase the TSO logon size to 8 megabytes (8192K) or above the line
  - Set the REXX/CLIST variable storage above the line

    ➔ TSO PROFILE VARSTORAGE(HIGH)

- Use ISPF 3.9, Commands (Create/change application command table)
  - Specifically designed for updating the site and user command tables
  - Where as ISPF 3.16, Tables (ISPF Table Utility) is a general purpose utility
    - Used to update a wide variety/range of tables

Notes

- Command table search order

Order	Command Table	Member
1	User command table 1	WFUCMDS
2	User command table 2	WFU2CMDS
3	User command table 3	WFU3CMDS
4	Site command table 1	WFSCMDS
5	Site command table 2	
6	Site command table 3	
7	System table	ISPCMDS

IVPs

- Some Independent Vendor Products (IVPs) provide the ability to implement stacking commands

- CA – PDSMAN
  - Initialization parameter
    - Use $EZYCMD to define site commands
  - Pros
    - Refresh the site command table by executing the following MVS command:

      ➔ F PDSMAN,REFRESH
  - Cons
    - To test/validate, must use a different lpar from the one the general TSO population is using
    - Must terminate/re-invoke ISPF to access new commands
    - Does not allow user command tables

CBT Tape

- Several command table utilities can be found on the CBT Tape

- Link
  - CBT Downloads

- Files
  - 183 – Gilbert Saint-flour's collection
    - FASTPATH – Another way to create/use a site command table
  - 312 – Lionel Dyck Collection of Utilities. A thru R
    - ISPFCMDS – Dynamically update ISPF Command Table
  - 349 – REXX to update in core ISPF command table on the fly
    - Prevents having to exit and re-invoke ISPF
  - 578 – General ISPF Table Handling Facility – Roy Gardiner

Docs

- Technical Support, 2003, Lionel Dyck
  - [ISPF – Command Tables for the Average Users]

- Using command tables to define commands
  - [ISPF – V2R1 – Users Guide – Vol 1]

MODEL Command

Introduction

- MODEL command
  - IBM edit macro
  - Quickly/easily obtain skeleton ISPF commands
  - Most useful, but least known/utilized
  - Continually being updated/enhanced by IBM
  - Only usable in EDIT and VIEW mode

- Links
  - SHARE, Session 16628, Aug 2015, Liam Doherty
    - ISPF Editor – Beyond The Basics Hands-on Lab
  - SHARE, Session 2646, August 22, 2002, Lionel B. Dyck
    - ISPF – Design Coding Hints and Tips
  - Chapter 4, Using edit models
    - ISPF – V2R1 – Edit and Edit Macros

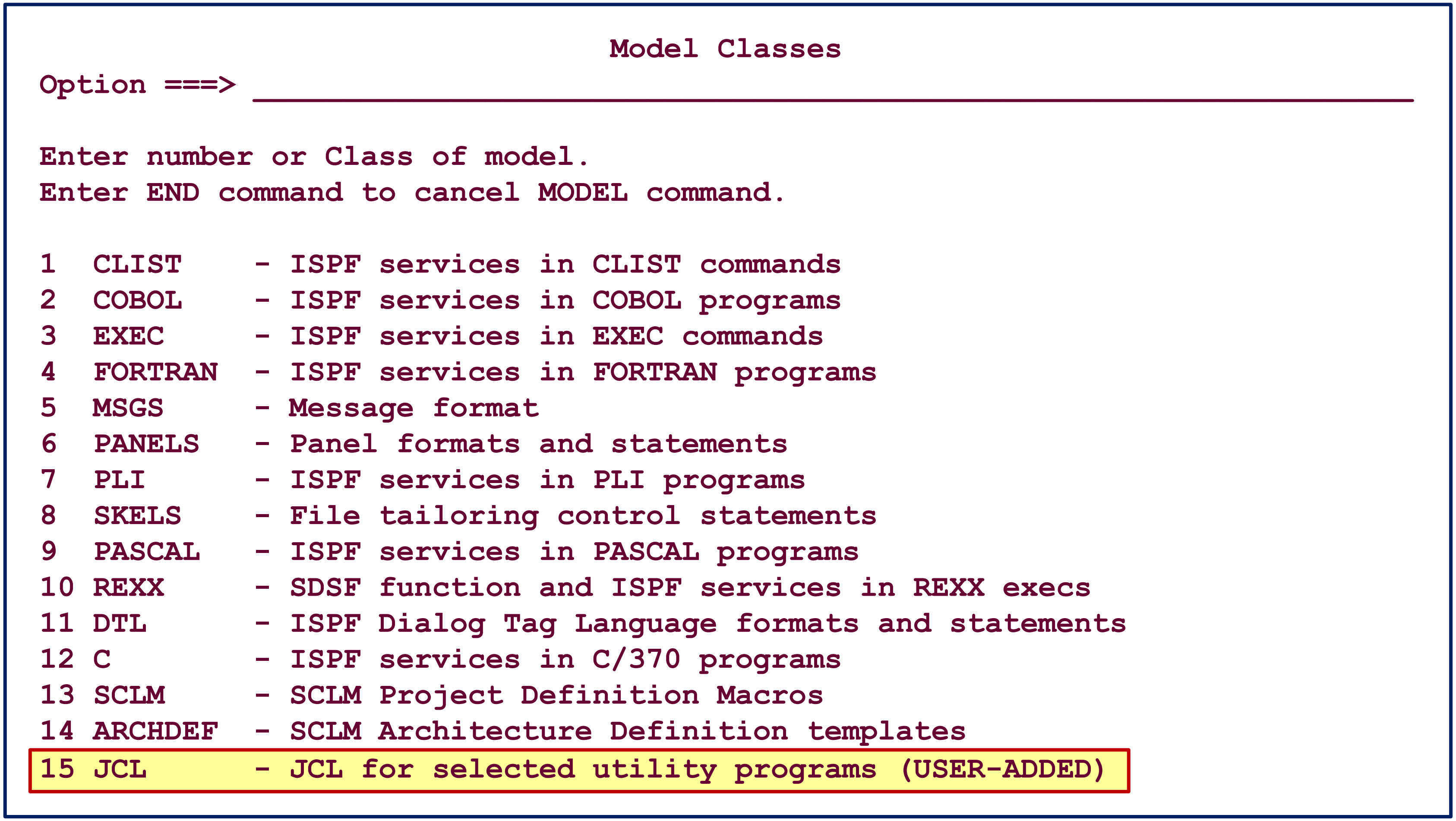
Enhanced

- The MODEL command can be enhanced to provide the following:
  - IBM, OEM and in-house JCL skeletons
  - SDSF REXX examples
  - REXX commands

- The next few slides display the following:
  - Changes highlighted in the IBM ISREMCLS panel
  - The ISREMJCL panel containing table of contents of all JCL utilities
  - Sample JCL skeletons, MIEHLIST and MUTKT
  - Changes highlighted in the IBM ISREMRXC panel
  - Where to obtain SDSF REXX and REXX commands

- Highlighted in yellow are changes made to the IBM panels

JCL – ISREMCLS – Panel (📎)



```
                                   Model Classes
Option ===> ________________________________________________________________

Enter number or Class of model.
Enter END command to cancel MODEL command.

1  CLIST    - ISPF services in CLIST commands
2  COBOL    - ISPF services in COBOL programs
3  EXEC     - ISPF services in EXEC commands
4  FORTRAN  - ISPF services in FORTRAN programs
5  MSGS     - Message format
6  PANELS   - Panel formats and statements
7  PLI      - ISPF services in PLI programs
8  SKELS    - File tailoring control statements
9  PASCAL   - ISPF services in PASCAL programs
10 REXX     - SDSF function and ISPF services in REXX execs
11 DTL      - ISPF Dialog Tag Language formats and statements
12 C        - ISPF services in C/370 programs
13 SCLM     - SCLM Project Definition Macros
14 ARCHDEF  - SCLM Architecture Definition templates
15 JCL      - JCL for selected utility programs (USER-ADDED)
```

JCL – ISREMCLS – Panel (📎)

```
)PANEL KEYLIST(ISRSNAB,ISR)
)ATTR DEFAULT(...) FORMAT(MIX)            /* ISREMCLS - ENGLISH - 7.1 */
 0D TYPE(PS)
 05 TYPE(PT)
 06 TYPE(PIN)
 09 TYPE(FP)
 0A TYPE(NT)
 11 TYPE(SAC)
 10 TYPE(ET)
 27 TYPE(NEF) CAPS(ON) PADC(USER)
 28 AREA(SCRL) EXTEND(ON)
)BODY WINDOW(76,22) EXPAND(..) CMD(ZCMD)
.. .Model Classes. ..
.Option ===>.Z. .                                                        .
.
.Enter number or Class of model.
.Enter.END.command to cancel MODEL command.
.SAREA40                                                                  .
)AREA SAREA40
.1 .CLIST   .- ISPF services in CLIST commands                            .
.2 .COBOL   .- ISPF services in COBOL programs                            .
.3 .EXEC    .- ISPF services in EXEC commands                             .
.4 .FORTRAN .- ISPF services in FORTRAN programs                          .
.5 .MSGS    .- Message format                                             .
.6 .PANELS  .- Panel formats and statements                               .
.7 .PLI     .- ISPF services in PLI programs                              .
.8 .SKELS   .- File tailoring control statements                          .
.9 .PASCAL  .- ISPF services in PASCAL programs                           .
.10.REXX    .- SDSF function and ISPF services in REXX execs              .
.11.DTL     .- ISPF Dialog Tag Language formats and statements            .
.12.C       .- ISPF services in C/370 programs                            .
.13.SCLM    .- SCLM Project Definition Macros                             .
.14.ARCHDEF .- SCLM Architecture Definition templates                     .
.15.JCL     .- JCL for selected utility programs (USER-ADDED)             .
)INIT
.ZVARS = '(ZCMD)'
&ZWINTTL = ' '
.HELP = ISR2P100
.MSG = '        '                  /*   BLANK MSG INITIALLY            */
&ISRMCLAS = '        '             /*   AND THE CLASS HOLDER VARIABLE */
.CURSOR = 'ZCMD'
)PROC
IF (&ISRMONCL = 'Y')               /* ARE WE ONLY OBTAINING A CLASS ? */
    &ISRMCLAS = TRANS(TRUNC(&ZCMD,'.')  1,CLIST 2,COB 3,EXEC 4,FORT
       5,MSGS 6,PANELS 7,PLI 8,SKELS 9,PASCAL 10,REXX 11,DTL 12,C
       13,SCLM 14,ARCHDEF *,*)
                                   /*                        @H1C@H5C*/
    &TMPTRAIL = .TRAIL             /* Save any remaining         @H1A*/
IF (&ISRMONCL ¬= 'Y')       /* ARE WE ONLY OBTAINING A CLASS ? OZ83844*/
```

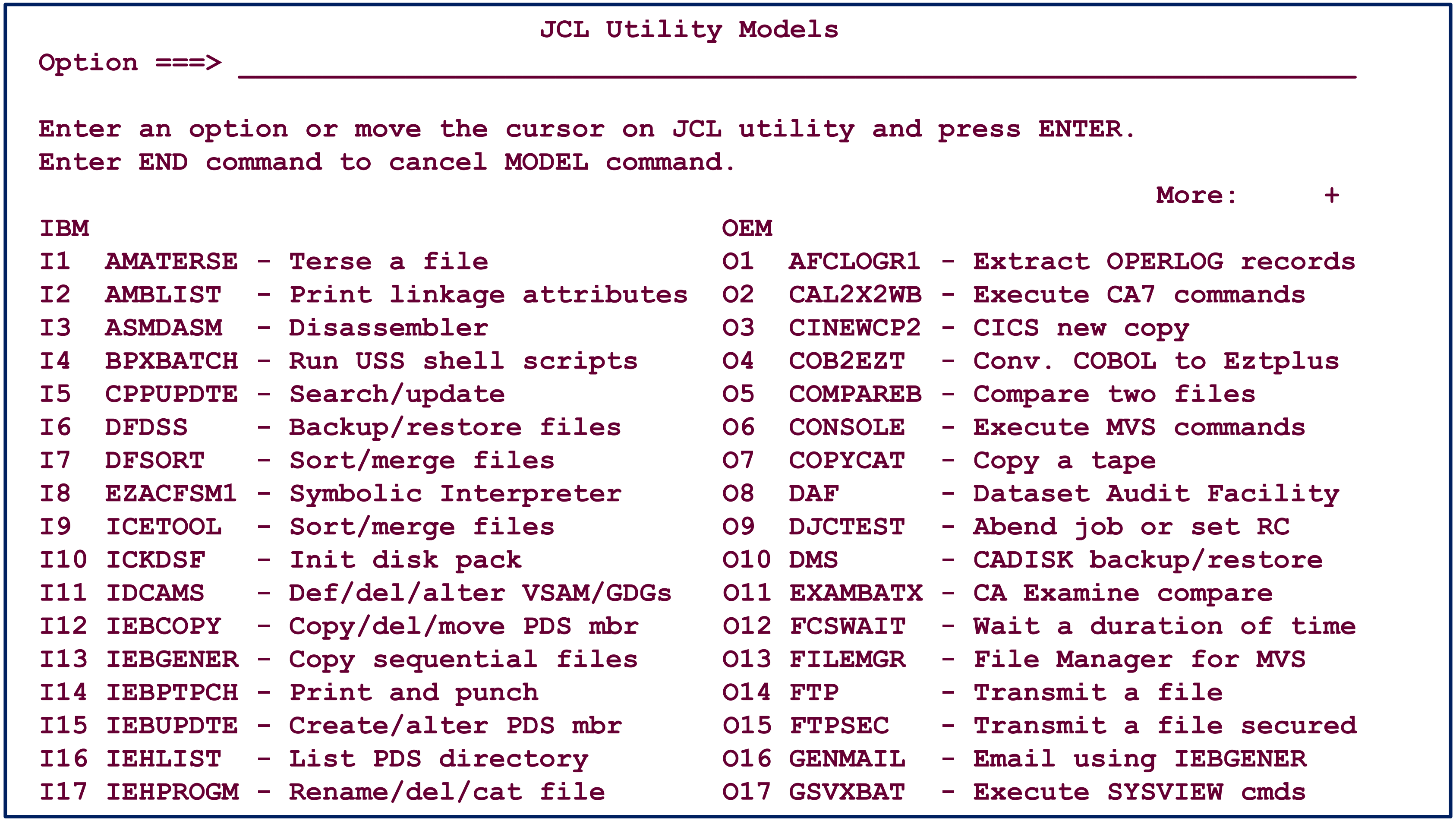
```
     &ISRMDSPL = '        '
&ZCMD = TRUNC (&ZCMD,'.')
&ZTMPTRL = .TRAIL
IF (&ZCMD = 'COBOL') &ZCMD = TRUNC(&ZCMD,3)
IF (&ZCMD = 'EXEC') &ZCMD = TRUNC(&ZCMD,4)
*REXX(ZCMD, JCLQUAL, CNTLQUAL, PROCQUAL, INSTQUAL)            /* @A2 */
   IF (POS("JCL",ZCMD) > 0) THEN                              /* @A2 */
      JCLQUAL = ZCMD                                          /* @A2 */
   ELSE                                                       /* @A2 */
      JCLQUAL = "JCL"                                         /* @A2 */
 IF (POS("CNTL",ZCMD) > 0) THEN                             /* @A2 */
      CNTLQUAL = ZCMD                                         /* @A2 */
   ELSE                                                       /* @A2 */
      CNTLQUAL = "CNTL"                                       /* @A2 */
   IF (POS("PROC",ZCMD) > 0) THEN                             /* @A2 */
      PROCQUAL = ZCMD                                         /* @A2 */
   ELSE                                                       /* @A2 */
      PROCQUAL = "PROC"                                       /* @A2 */
   IF (POS("INST",ZCMD) > 0) THEN                             /* @A2 */
      INSTQUAL = ZCMD                                         /* @A2 */
   ELSE                                                       /* @A2 */
      INSTQUAL = "INST"                                       /* @A2 */
*ENDREXX                                                      /* @A2 */
&ZCMD = TRANS (&ZCMD
CLIST,1 COB,2 EXEC,3 FORTRAN,4 FTN,4A FORT,4B MSGS,5 ISPMLIB,5A PANELS,6
MENUS,6A ISPPLIB,6B PLI,7 PL1,7A PLIOPT,7B PLS,7C SKELS,8 PROCS,8A ISPSLIB,8B
PASCAL,9 REXX,10 DTL,11 C,12 CPP,1C SCLM,13 ARCHDEF,14 JCL,15 ' ',' ' *,*)
IF (&ZCMD ¬= ' ') &ZCMD = '&ZCMD..&ZTMPTRL'
&ZSEL = TRANS (TRUNC (&ZCMD,'.')
  1,'PANEL(ISREMCMF)'
 2,'PANEL(ISREMCOB)'
  3,'PANEL(ISREMEXC)'
  4,'PANEL(ISREMFTN)'
 4A,'PANEL(ISREMFTN)'
 4B,'PANEL(ISREMFTN)'
  5,'PGM(ISRECMBR) PARM(ISREMMSG NOSEQ)'
 5A,'PGM(ISRECMBR) PARM(ISREMMSG NOSEQ)'
  6,'PANEL(ISREMPNL)'
 6A,'PANEL(ISREMPNL)'
 6B,'PANEL(ISREMPNL)'
  7,'PANEL(ISREMPLI)'
 7A,'PANEL(ISREMPLI)'
 7B,'PANEL(ISREMPLI)'
 7C,'PANEL(ISREMPLI)'
  8,'PANEL(ISREMSKL)'
 8A,'PANEL(ISREMSKL)'
 8B,'PANEL(ISREMSKL)'
  9,'PANEL(ISREMPAS)'
 10,'PANEL(ISREMRXC)'
```

JCL – ISREMCLS – Panel (📎)

```
 11,'PANEL(ISREMDTL)'
 12,'PANEL(ISREMWLN)'
 1C,'PANEL(ISREMWLN)'
 13,'PANEL(ISREMFLM)'
 14,'PANEL(ISREMARC)'
 15,'PANEL(ISREMJCL)'
 &JCLQUAL,'PANEL(ISREMJCL)'                                    /* @A2 */
 &CNTLQUAL,'PANEL(ISREMJCL)'                                   /* @A2 */
 &PROCQUAL,'PANEL(ISREMJCL)'                                   /* @A2 */
 &INSTQUAL,'PANEL(ISREMJCL)'                                   /* @A0 */
 ISPCLIST,'PANEL(ISREMCMF)'                                    /* @A1 */
 ISPREXX,'PANEL(ISREMRXC)'                                     /* @A1 */
 ISPEXEC,'PANEL(ISREMRXC)'                                     /* @A1 */
 ' ',' '
   *,'?')
IF (&ZSEL = '?')
                 /* IF (&ISRMDSPL ¬= 'RETURN' ) ** REPLACED BY OY33372*/
  IF (&ZMACRO = 'N' )                /*                          OY33372*/
    &ZEMTEMP = &ZCMD                 /* SET VARIABLE FOR MSG   OZ91267*/
    .MSG = ISRYM012
    &ZCMD = ' '                      /* RESET CMD LINE = BLANK OZ91267*/
 .RESP = 'ENTER'                  /* WANT TO RESET RESP FOR RE-ENT */
                  /* IF (&ISRMDSPL = 'RETURN' ) ** REPLACED BY OY33372*/
  ELSE                               /*                          OY33372*/
    .RESP = 'END'
    &ISRMCLAS = '?'
    .MSG = ISRYM012                  /* INVALID CLASS          OY33372*/
&ISRMMEND = 'N'                      /* SET THE END INDICATOR TO NO   */
IF (.RESP = END)                     /* DID WE GET AN END FROM WHERE  */
    IF (&ISRMDSPL ¬= 'RETURN  ')     /* WAS IT BECAUSE WE HAVE MEMBER */
      &ISRMMEND = 'Y'                /* ITS BECAUSE USER HIT END KEY  */
IF (&ISRMONCL = 'Y')                 /* ARE WE ONLY OBTAINING A CLASS */
   IF (.RESP ¬= 'RETURN  ')          /* Has user hit extended return  */
     IF (&ISRMCLAS ¬= '        ')    /* Make sure we have some class  */
        IF (&TMPTRAIL = ' ')         /* Just class entered        @H1A*/
           .RESP = END              /* ONLY WANT CLASS, NO OTHER DISP.*/
        IF (&TMPTRAIL ¬= ' ')        /* More than class entered   @H1A*/
           &ZEMTEMP = &ZCMD          /* Set variable for message  @H1A*/
           .MSG = ISRYM012           /* Set message number        @H1A*/
           &ZCMD = ' '               /* Reset cmd lint = blank    @H1A*/
           .RESP = 'ENTER'           /* Reset RESP to Re-enter    @H1A*/
&ZMSG = .MSG                         /* SET MESSAGE VARIABLE      @L1A*/
VPUT(ZMSG) PROFILE                   /* SAVE MESSAGE IN PROF POOL @L1A*/
)PNTS
FIELD(ZPS01001) VAR(ZCMD) VAL(1)
FIELD(ZPS01002) VAR(ZCMD) VAL(2)
FIELD(ZPS01003) VAR(ZCMD) VAL(3)
FIELD(ZPS01004) VAR(ZCMD) VAL(4)
FIELD(ZPS01005) VAR(ZCMD) VAL(5)
```

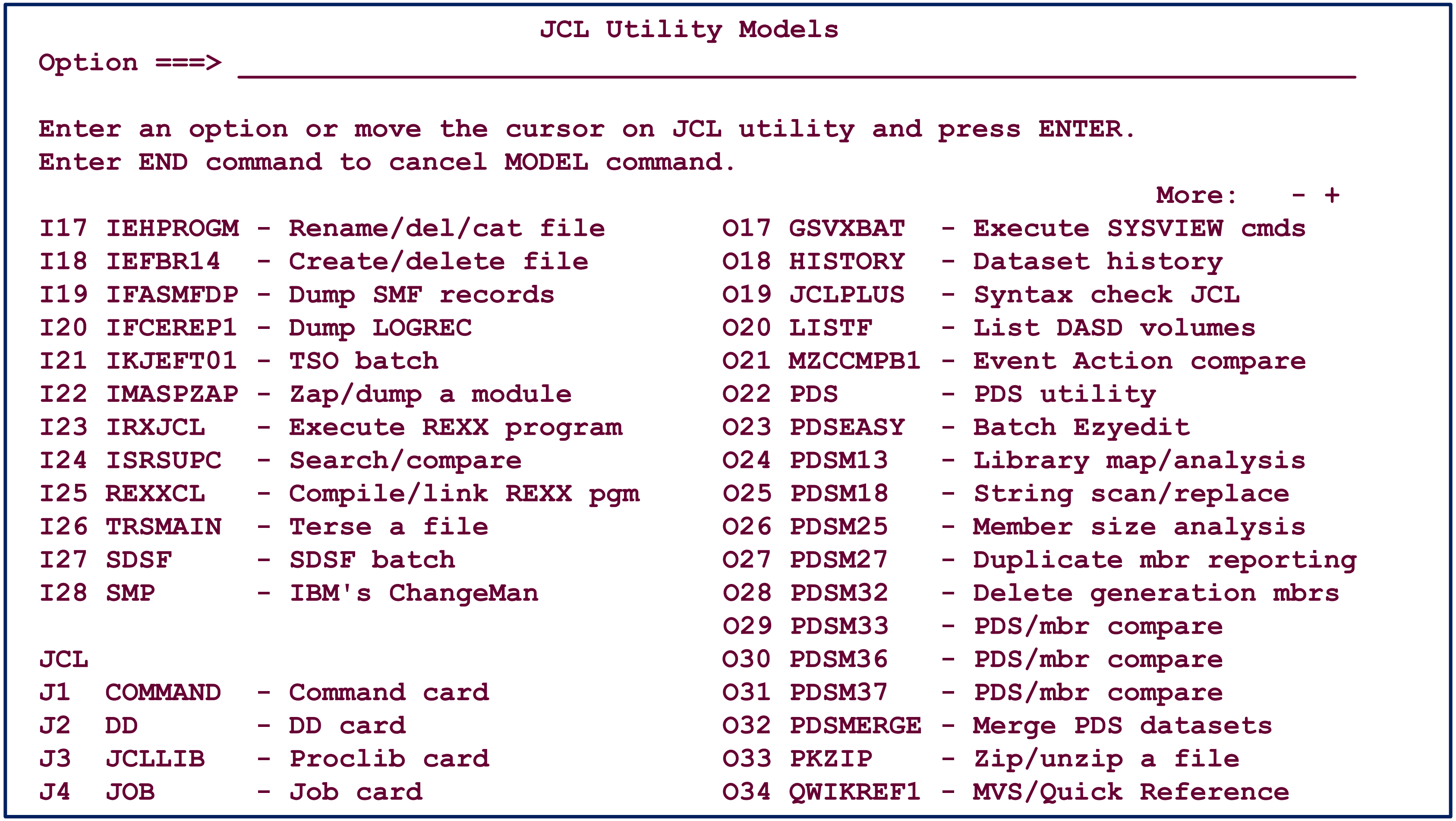
```
FIELD(ZPS01006) VAR(ZCMD) VAL(6)
FIELD(ZPS01007) VAR(ZCMD) VAL(7)
FIELD(ZPS01008) VAR(ZCMD) VAL(8)
FIELD(ZPS01009) VAR(ZCMD) VAL(9)
FIELD(ZPS01010) VAR(ZCMD) VAL(10)
FIELD(ZPS01011) VAR(ZCMD) VAL(11)
FIELD(ZPS01012) VAR(ZCMD) VAL(12)
FIELD(ZPS01013) VAR(ZCMD) VAL(13)
FIELD(ZPS01014) VAR(ZCMD) VAL(14)
FIELD(ZPS01015) VAR(ZCMD) VAL(15)                              /* @A0 */
)END
/* 5694-A01     COPYRIGHT IBM CORP 1980, 2011 */
/* ISPDTLC Release: 7.1.  Level: PID                                    */
/* z/OS 02.01.00.  Created - Date: 27 Aug 2012, Time: 17:08             */
/*--------------------------------------------------------------------*/
/*                         Wells Fargo Changes                        */
/*--------------------------------------------------------------------*/
/* FLG  YYMMDD  USERID   DESCRIPTION                                  */
/* ---  ------  -------  -------------------------------------------- */
/* @A2  100519  R2221    Last level qual. = JCL, CNTL, PROC, INST     */
/* @A1  100203  R2221    Added .ISPCLIST, .ISPREXX, and .ISPEXEC      */
/* @A0  090118  R2221    Added Option Number 15 - JCL utilities       */
/* @A0  090118  R2221    Find everything that was modified execute:   */
/* @A0  090118  R2221       ==> X ALL;F '15' ALL;F "ISREMJCL" ALL     */
/*--------------------------------------------------------------------*/
```

JCL – ISREMJCL – Panel (📎)



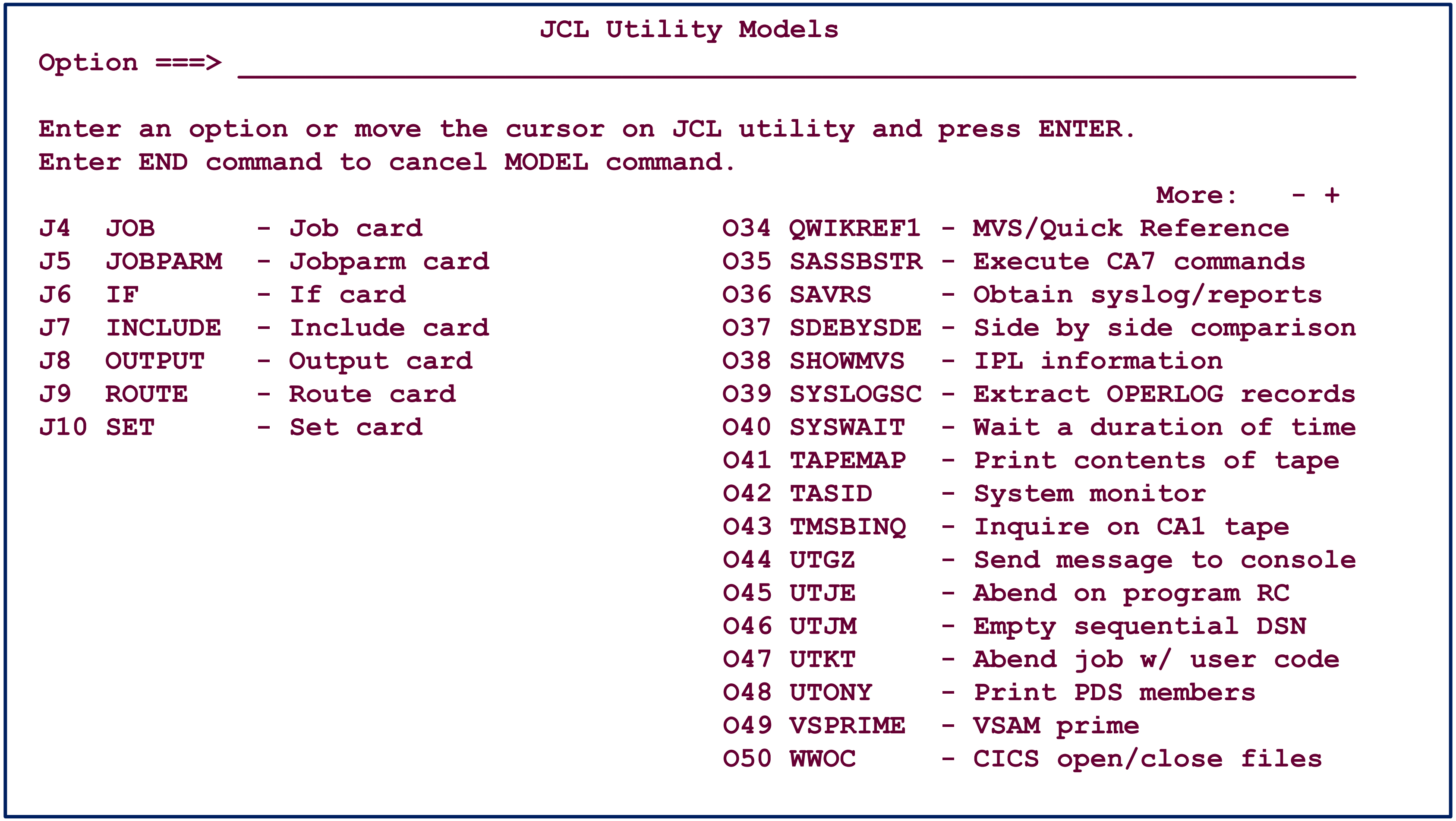
```
                             JCL Utility Models
Option ===> ___________________________________________________________________

Enter an option or move the cursor on JCL utility and press ENTER.
Enter END command to cancel MODEL command.
                                                                  More:     +
IBM                                         OEM
I1  AMATERSE - Terse a file                 O1  AFCLOGR1 - Extract OPERLOG records
I2  AMBLIST  - Print linkage attributes     O2  CAL2X2WB - Execute CA7 commands
I3  ASMDASM  - Disassembler                 O3  CINEWCP2 - CICS new copy
I4  BPXBATCH - Run USS shell scripts        O4  COB2EZT  - Conv. COBOL to Eztplus
I5  CPPUPDTE - Search/update                O5  COMPAREB - Compare two files
I6  DFDSS    - Backup/restore files         O6  CONSOLE  - Execute MVS commands
I7  DFSORT   - Sort/merge files             O7  COPYCAT  - Copy a tape
I8  EZACFSM1 - Symbolic Interpreter         O8  DAF      - Dataset Audit Facility
I9  ICETOOL  - Sort/merge files             O9  DJCTEST  - Abend job or set RC
I10 ICKDSF   - Init disk pack               O10 DMS      - CADISK backup/restore
I11 IDCAMS   - Def/del/alter VSAM/GDGs      O11 EXAMBATX - CA Examine compare
I12 IEBCOPY  - Copy/del/move PDS mbr        O12 FCSWAIT  - Wait a duration of time
I13 IEBGENER - Copy sequential files        O13 FILEMGR  - File Manager for MVS
I14 IEBPTPCH - Print and punch              O14 FTP      - Transmit a file
I15 IEBUPDTE - Create/alter PDS mbr         O15 FTPSEC   - Transmit a file secured
I16 IEHLIST  - List PDS directory           O16 GENMAIL  - Email using IEBGENER
I17 IEHPROGM - Rename/del/cat file          O17 GSVXBAT  - Execute SYSVIEW cmds
```

JCL – ISREMJCL – Panel (📎)



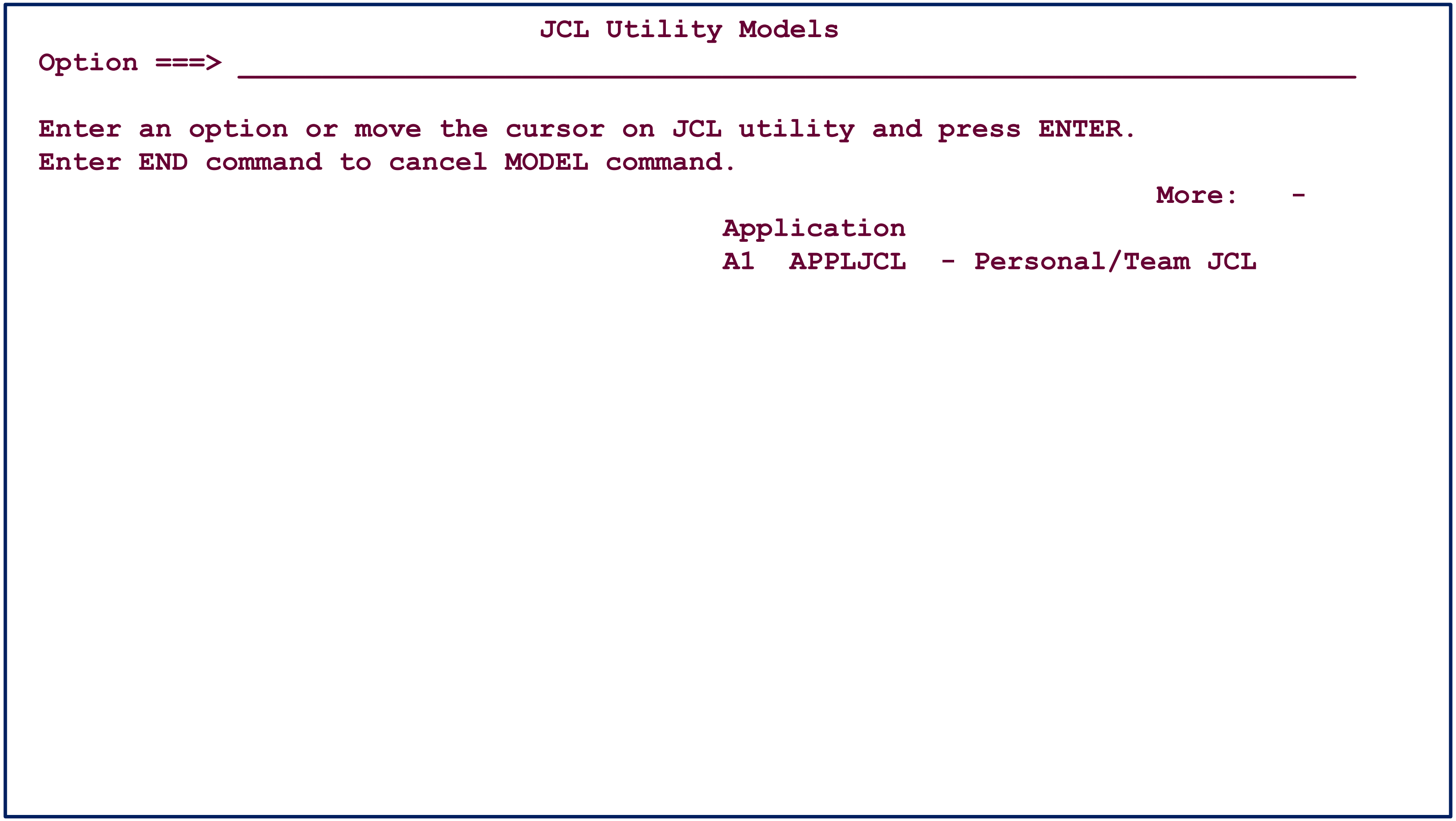
```
                              JCL Utility Models
Option ===> __________________________________________________________________

Enter an option or move the cursor on JCL utility and press ENTER.
Enter END command to cancel MODEL command.
                                                                  More:    - +
I17 IEHPROGM - Rename/del/cat file        O17 GSVXBAT  - Execute SYSVIEW cmds
I18 IEFBR14  - Create/delete file         O18 HISTORY  - Dataset history
I19 IFASMFDP - Dump SMF records           O19 JCLPLUS  - Syntax check JCL
I20 IFCEREP1 - Dump LOGREC                O20 LISTF    - List DASD volumes
I21 IKJEFT01 - TSO batch                  O21 MZCCMPB1 - Event Action compare
I22 IMASPZAP - Zap/dump a module          O22 PDS      - PDS utility
I23 IRXJCL   - Execute REXX program       O23 PDSEASY  - Batch Ezyedit
I24 ISRSUPC  - Search/compare             O24 PDSM13   - Library map/analysis
I25 REXXCL   - Compile/link REXX pgm      O25 PDSM18   - String scan/replace
I26 TRSMAIN  - Terse a file               O26 PDSM25   - Member size analysis
I27 SDSF     - SDSF batch                 O27 PDSM27   - Duplicate mbr reporting
I28 SMP      - IBM's ChangeMan            O28 PDSM32   - Delete generation mbrs
                                          O29 PDSM33   - PDS/mbr compare
JCL                                       O30 PDSM36   - PDS/mbr compare
J1  COMMAND  - Command card               O31 PDSM37   - PDS/mbr compare
J2  DD       - DD card                    O32 PDSMERGE - Merge PDS datasets
J3  JCLLIB   - Proclib card               O33 PKZIP    - Zip/unzip a file
J4  JOB      - Job card                   O34 QWIKREF1 - MVS/Quick Reference
```

JCL – ISREMJCL – Panel (📎)



```
                          JCL Utility Models
Option ===> ___________________________________________________________________

Enter an option or move the cursor on JCL utility and press ENTER.
Enter END command to cancel MODEL command.
                                                                  More:    - +
J4  JOB       - Job card                 O34 QWIKREF1 - MVS/Quick Reference
J5  JOBPARM   - Jobparm card             O35 SASSBSTR - Execute CA7 commands
J6  IF        - If card                  O36 SAVRS    - Obtain syslog/reports
J7  INCLUDE   - Include card             O37 SDEBYSDE - Side by side comparison
J8  OUTPUT    - Output card              O38 SHOWMVS  - IPL information
J9  ROUTE     - Route card               O39 SYSLOGSC - Extract OPERLOG records
J10 SET       - Set card                 O40 SYSWAIT  - Wait a duration of time
                                         O41 TAPEMAP  - Print contents of tape
                                         O42 TASID    - System monitor
                                         O43 TMSBINQ  - Inquire on CA1 tape
                                         O44 UTGZ     - Send message to console
                                         O45 UTJE     - Abend on program RC
                                         O46 UTJM     - Empty sequential DSN
                                         O47 UTKT     - Abend job w/ user code
                                         O48 UTONY    - Print PDS members
                                         O49 VSPRIME  - VSAM prime
                                         O50 WWOC     - CICS open/close files
```

JCL – ISREMJCL – Panel (📎)



```
                              JCL Utility Models
Option ===> ___________________________________________________________________

Enter an option or move the cursor on JCL utility and press ENTER.
Enter END command to cancel MODEL command.
                                                                   More:   -
                                         Application
                                         A1  APPLJCL  - Personal/Team JCL
```

JCL – ISREMJCL – Panel (📎)

```
)PANEL KEYLIST(ISRSNAB,ISR)
)ATTR DEFAULT(%+_) FORMAT(MIX)            /* ISREMPLI - ENGLISH - 5.0 */
 %  TYPE(TEXT) INTENS(HIGH) COLOR(TURQ)
 +  TYPE(TEXT) INTENS(LOW)  COLOR(GREEN)
 [  TYPE(PS)
 ]  TYPE(NT)
 0D TYPE(PS)
 05 TYPE(PT)
 06 TYPE(PIN)
 09 TYPE(FP)
 0A TYPE(NT)
 11 TYPE(SAC)
 22 TYPE(WASL) SKIP(ON) GE(ON)
 10 TYPE(ET)
 27 TYPE(NEF) CAPS(ON) PADC(USER)
 28 AREA(SCRL) EXTEND(ON)
)BODY WINDOW(80,22) EXPAND(..) CMD(ZCMD)
.. .JCL Utility Models . ..
.Option ===>.Z. . .
.
.Enter an option or move the cursor on JCL utility and press ENTER.
.Enter.END.command to cancel MODEL command.
.SAREA40                                                                        .
)AREA SAREA40
%IBM                                     %OEM+
+I1  [AMATERSE]- Terse a file             O1  [AFCLOGR1]- Extract OPERLOG records
+I2  [AMBLIST ]- Print linkage attributes O2  [CAL2X2WB]- Execute CA7 commands
+I3  [ASMDASM ]- Disassembler             O3  [CINEWCP2]- CICS new copy
+I4  [BPXBATCH]- Run USS shell scripts    O4  [COB2EZT ]- Conv. COBOL to Eztplus
+I5  [CPPUPDTE]- Search/update            O5  [COMPAREB]- Compare two files
+I6  [DFDSS   ]- Backup/restore files     O6  [CONSOLE ]- Execute MVS commands
+I7  [DFSORT  ]- Sort/merge files         O7  [COPYCAT ]- Copy a tape
+I8  [EZACFSM1]- Symbolic Interpreter     O8  [DAF     ]- Dataset Audit Facility
+I9  [ICETOOL ]- Sort/merge files         O9  [DJCTEST ]- Abend job or set RC
+I10[ICKDSF  ]- Init disk pack            O10[DMS     ]- CADISK backup/restore
+I11[IDCAMS  ]- Def/del/alter VSAM/GDGs   O11[EXAMBATX]- CA Examine compare
+I12[IEBCOPY ]- Copy/del/move PDS mbr     O12[FCSWAIT ]- Wait a duration of time
+I13[IEBGENER]- Copy sequential files     O13[FILEMGR ]- File Manager for MVS
+I14[IEBPTPCH]- Print and punch           O14[FTP     ]- Transmit a file
+I15[IEBUPDTE]- Create/alter PDS mbr      O15[FTPSEC  ]- Transmit a file secured
+I16[IEHLIST ]- List PDS directory        O16[GENMAIL ]- Email using IEBGENER
+I17[IEHPROGM]- Rename/del/cat file       O17[GSVXBAT ]- Execute SYSVIEW cmds
+I18[IEFBR14 ]- Create/delete file        O18[HISTORY ]- Dataset history
+I19[IFASMFDP]- Dump SMF records          O19[JCLPLUS ]- Syntax check JCL
+I20[IFCEREP1]- Dump LOGREC               O20[LISTF   ]- List DASD volumes
+I21[IKJEFT01]- TSO batch                 O21[MZCCMPB1]- Event Action compare
+I22[IMASPZAP]- Zap/dump a module         O22[PDS     ]- PDS utility
+I23[IRXJCL  ]- Execute REXX program      O23[PDSEASY ]- Batch Ezyedit
+I24[ISRSUPC ]- Search/compare            O24[PDSM13  ]- Library map/analysis
```

```
+I25[REXXCL  ]- Compile/link REXX pgm     O25[PDSM18  ]- String scan/replace
+I26[TRSMAIN ]- Terse a file              O26[PDSM25  ]- Member size analysis
+I27[SDSF    ]- SDSF batch                O27[PDSM27  ]- Duplicate mbr reporting
+I28[SMP     ]- IBM's ChangeMan           O28[PDSM32  ]- Delete generation mbrs
+                                         O29[PDSM33  ]- PDS/mbr compare
%JCL+                                     O30[PDSM36  ]- PDS/mbr compare
+J1  [COMMAND ]- Command card             O31[PDSM37  ]- PDS/mbr compare
+J2  [DD      ]- DD card                  O32[PDSMERGE]- Merge PDS datasets
+J3  [JCLLIB  ]- Proclib card             O33[PKZIP   ]- Zip/unzip a file
+J4  [JOB     ]- Job card                 O34[QWIKREF1]- MVS/Quick Reference
+J5  [JOBPARM ]- Jobparm card             O35[SASSBSTR]- Execute CA7 commands
+J6  [IF      ]- If card                  O36[SAVRS   ]- Obtain syslog/reports
+J7  [INCLUDE ]- Include card             O37[SDEBYSDE]- Side by side comparison
+J8  [OUTPUT  ]- Output card              O38[SHOWMVS ]- IPL information
+J9  [ROUTE   ]- Route card               O39[SYSLOGSC]- Extract OPERLOG records
+J10[SET     ]- Set card                  O40[SYSWAIT ]- Wait a duration of time
+                                         O41[TAPEMAP ]- Print contents of tape
+                                         O42[TASID   ]- System monitor
+                                         O43[TMSBINQ ]- Inquire on CA1 tape
+                                         O44[UTGZ    ]- Send message to console
+                                         O45[UTJE    ]- Abend on program RC
+                                         O46[UTJM    ]- Empty sequential DSN
+                                         O47[UTKT    ]- Abend job w/ user code
+                                         O48[UTONY   ]- Print PDS members
+                                         O49[VSPRIME ]- VSAM prime
+                                         O50[WWOC]    - CICS open/close files
+
+                                        %Application+
+                                         A1  [APPLJCL ]- Personal/Team JCL
)INIT
.ZVARS = '(ZCMD)'
&ZWINTTL = ' '
.HELP = #ISREMJC
IF (&ISRMDSPL = 'RETURN  ')
    .RESP = END
.CURSOR = 'ZCMD'
)REINIT
REFRESH(ZCMD)
)PROC
&ZCMD = TRUNC (&ZCMD,'.')
&ZTMPTRL = .TRAIL
&ZCMD = TRANS (&ZCMD
AMATERSE,I1 AMBLIST,I2 ASMDASM,I3 MBPXBATCH,I4 BPXBATSL,I4
CPPUPDTE,I5 IPOUPDTE,I5
DFDSS,I6 DFSORT,I7 ICEMAN,I7 SORT,I7 EZACFSM1,I8
ICETOOL,I9 ICKDSF,I10 IDCAMS,I11 IEBCOPY,I12
IEBGENER,I13 GENER,I13 IEBPTPCH,I14 IEBUPDTE,I15 IEHLIST,I16
IEHPROGM,I17 IEFBR14,I18 BR14,I18 IFASMFDP,I19 SMFDMP,I19
IFCEREP1,I20 LOGREC,I20 IKJEFT01,I21 TSOBATCH,I21 TSO,I21
```

JCL – ISREMJCL – Panel (📎)

```
IMASPZAP,I22 ZAP,I22 IRXJCL,I23 ISRSUPC,I24 REXXCL,I25
TRSMAIN,I26 SDSF,I27 SMP,I28
AFCLOGR1,O1 CAL2X2WB,O2 CA7,O2 CINEWCPY2,O3 COB2EZT,O4 COMPAREB,O5
CONSOLE,O6 COPYCAT,O7 DAF,O8 DJCTEST,O9 DMS,O10
CADISK,O10 EXAMBATX,O11
FCSWAIT,O12 FILEMGR,O13
FTP,O14 FTPSEC,O15 GENMAIL,O16 GSVXBAT,O17 HISTORY,O18 HIST,O18
JCLPLUS,O19 LISTF,O20 MZCCMPB1,O21 PDS,O22 PDS86,O22
PDSEASY,O23 EZY,O23 PDSM13,O24 PDSM18,O25 PDSM25,O26, PDSM27,O27
PDSM32,O28 PDSM33,O29 PDSM36,O30 PDSM37,O31
PDSMERGE,O32 PKZIP,O33 QWIKREF1,O34
QW,O34 SASSBSTR,O35 SAVRS,O36 SAVERS,O36 SDEBYSDE,O37
SHOWMVS,O38 SYSLOGSC,O39 SYSWAIT,O40 TAPEMAP,O41 TASID,O42
TMSBINQ,O43 CA1,O43 TMS,O43 UTGZ,O44 UTJE,O45 UTJM,O46
UTKT,O47 UTONY,O48 VSPRIME,O49 WWOC,O50
COMMAND,J1 DD,J2 JCLLIB,J3 JOB,J4 JOBPARM,J5 SYSAFF,J5
IF,J6 INCLUDE,J7 OUTPUT,J8 ROUTE,J9 SET,J10
APPLJCL,A1
' ',' ' *,*)
IF (&ZCMD ¬= ' ') &ZCMD = '&ZCMD..&ZTMPTRL'
&ZSEL = TRANS (TRUNC (&ZCMD,'.')
       I1,'PGM(ISRECMBR) PARM(MAMATERS)'
       I2,'PGM(ISRECMBR) PARM(MAMBLIST)'
       I3,'PGM(ISRECMBR) PARM(MASMDASM)'
       I4,'PGM(ISRECMBR) PARM(MBPXBATC)'
       I5,'PGM(ISRECMBR) PARM(MCPPUPDT)'
       I6,'PGM(ISRECMBR) PARM(MDFDSS  )'
       I7,'PGM(ISRECMBR) PARM(MDFSORT )'
       I8,'PGM(ISRECMBR) PARM(MEZACFSM)'
       I9,'PGM(ISRECMBR) PARM(MICETOOL)'
      I10,'PGM(ISRECMBR) PARM(MICKDSF )'
      I11,'PGM(ISRECMBR) PARM(MIDCAMS )'
      I12,'PGM(ISRECMBR) PARM(MIEBCOPY)'
      I13,'PGM(ISRECMBR) PARM(MIEBGENE)'
      I14,'PGM(ISRECMBR) PARM(MIEBPTPC)'
      I15,'PGM(ISRECMBR) PARM(MIEBUPDT)'
      I16,'PGM(ISRECMBR) PARM(MIEHLIST)'
      I17,'PGM(ISRECMBR) PARM(MIEHPROG)'
      I18,'PGM(ISRECMBR) PARM(MIEFBR14)'
      I19,'PGM(ISRECMBR) PARM(MIFASMFD)'
      I20,'PGM(ISRECMBR) PARM(MIFCEREP)'
      I21,'PGM(ISRECMBR) PARM(MIKJEFT0)'
      I22,'PGM(ISRECMBR) PARM(MIMASPZA)'
      I23,'PGM(ISRECMBR) PARM(MIRXJCL )'
      I24,'PGM(ISRECMBR) PARM(MISRSUPC)'
      I25,'PGM(ISRECMBR) PARM(MREXXCL )'
      I26,'PGM(ISRECMBR) PARM(MTRSMAIN)'
      I27,'PGM(ISRECMBR) PARM(MSDSF   )'
      I28,'PGM(ISRECMBR) PARM(MSMP    )'
```

```
       O1,'PGM(ISRECMBR) PARM(MAFCLOGR)'
       O2,'PGM(ISRECMBR) PARM(MCAL2X2W)'
       O3,'PGM(ISRECMBR) PARM(MCINEWCP)'
       O4,'PGM(ISRECMBR) PARM(MCOB2EZT)'
       O5,'PGM(ISRECMBR) PARM(MCOMPARE)'
       O6,'PGM(ISRECMBR) PARM(MCONSOLE)'
       O7,'PGM(ISRECMBR) PARM(MCOPYCAT)'
       O8,'PGM(ISRECMBR) PARM(MDAF    )'
       O9,'PGM(ISRECMBR) PARM(MDJCTEST)'
      O10,'PGM(ISRECMBR) PARM(MDMS    )'
      O11,'PGM(ISRECMBR) PARM(MEXAMBAT)'
      O12,'PGM(ISRECMBR) PARM(MFCSWAIT)'
      O13,'PGM(ISRECMBR) PARM(MFILEMGR)'
      O14,'PGM(ISRECMBR) PARM(MFTP    )'
      O15,'PGM(ISRECMBR) PARM(MFTPSEC )'
      O16,'PGM(ISRECMBR) PARM(MGENMAIL)'
      O17,'PGM(ISRECMBR) PARM(MGSVXBAT)'
      O18,'PGM(ISRECMBR) PARM(MHISTORY)'
      O19,'PGM(ISRECMBR) PARM(MJCLPLUS)'
      O20,'PGM(ISRECMBR) PARM(MLISTF  )'
      O21,'PGM(ISRECMBR) PARM(MMZCCMPB)'
      O22,'PGM(ISRECMBR) PARM(MPDS    )'
      O23,'PGM(ISRECMBR) PARM(MPDSEASY)'
      O24,'PGM(ISRECMBR) PARM(MPDSM13 )'
      O25,'PGM(ISRECMBR) PARM(MPDSM18 )'
      O26,'PGM(ISRECMBR) PARM(MPDSM25 )'
      O27,'PGM(ISRECMBR) PARM(MPDSM27 )'
      O28,'PGM(ISRECMBR) PARM(MPDSM32 )'
      O29,'PGM(ISRECMBR) PARM(MPDSM33 )'
      O30,'PGM(ISRECMBR) PARM(MPDSM36 )'
      O31,'PGM(ISRECMBR) PARM(MPDSM37 )'
      O32,'PGM(ISRECMBR) PARM(MPDSMERG)'
      O33,'PGM(ISRECMBR) PARM(MPKZIP  )'
      O34,'PGM(ISRECMBR) PARM(MQWIKREF)'
      O35,'PGM(ISRECMBR) PARM(MSASSBST)'
      O36,'PGM(ISRECMBR) PARM(MSAVRS  )'
      O37,'PGM(ISRECMBR) PARM(MSDEBYSD)'
      O38,'PGM(ISRECMBR) PARM(MSHOWMVS)'
      O39,'PGM(ISRECMBR) PARM(MSYSLOGS)'
      O40,'PGM(ISRECMBR) PARM(MSYSWAIT)'
      O41,'PGM(ISRECMBR) PARM(MTAPEMAP)'
      O42,'PGM(ISRECMBR) PARM(MTASID  )'
      O43,'PGM(ISRECMBR) PARM(MTMSBINQ)'
      O44,'PGM(ISRECMBR) PARM(MUTGZ   )'
      O45,'PGM(ISRECMBR) PARM(MUTJE   )'
      O46,'PGM(ISRECMBR) PARM(MUTJM   )'
      O47,'PGM(ISRECMBR) PARM(MUTKT   )'
      O48,'PGM(ISRECMBR) PARM(MUTONY  )'
      O49,'PGM(ISRECMBR) PARM(MVSPRIME)'
```

JCL – ISREMJCL – Panel (📎)

```
     O50,'PGM(ISRECMBR) PARM(MWWOC)'
      J1,'PGM(ISRECMBR) PARM(MCOMMAND)'
      J2,'PGM(ISRECMBR) PARM(MDD     )'
      J3,'PGM(ISRECMBR) PARM(MJCLLIB )'
      J4,'PGM(ISRECMBR) PARM(MJOB    )'
      J5,'PGM(ISRECMBR) PARM(MJOBPARM)'
      J6,'PGM(ISRECMBR) PARM(MIF     )'
      J7,'PGM(ISRECMBR) PARM(MINCLUDE)'
      J8,'PGM(ISRECMBR) PARM(MOUTPUT )'
      J9,'PGM(ISRECMBR) PARM(MROUTE  )'
     J10,'PGM(ISRECMBR) PARM(MSET    )'
      A1,'PANEL(ISREMAPP)'
      ' ',' '
        *,'?' )
IF (&ZSEL = '?')
   &ZEMTEMP = &ZCMD
   .MSG = ISRYM013
   &ZCMD = ' '
   IF (&ZMACRO = 'Y')                  /* If a macro command    @OY38423*/
     .RESP = 'END'                     /* don't display error   @OY38423*/
     &ISRMMEMB = '?'                   /* Indicate no member fnd@OY38423*/
&ISRMMEND = 'N'                        /* SET THE END INDICATOR TO NO   */
IF (.RESP = END)                       /* IF ENDING, WHY ... WHO CAUSED */
 IF (&ISRMONCL ¬= 'Y')                /* MAKE SURE ITS NOT A CLASS OP. */
    IF (&ISRMDSPL ¬= 'RETURN  ')       /* MAKE SURE ITS NOT END ON MBR. */
      &ISRMMEND = 'Y'                  /* NO - ITS BECAUSE USER HIT END */
&ZMSG = .MSG                           /* Set message variable          */
VPUT(ZMSG) PROFILE                     /* Save message in prof pool     */
)PNTS
FIELD(ZPS01001) VAR(ZCMD) VAL(I1)
FIELD(ZPS01002) VAR(ZCMD) VAL(O1)
FIELD(ZPS01003) VAR(ZCMD) VAL(I2)
FIELD(ZPS01004) VAR(ZCMD) VAL(O2)
FIELD(ZPS01005) VAR(ZCMD) VAL(I3)
FIELD(ZPS01006) VAR(ZCMD) VAL(O3)
FIELD(ZPS01007) VAR(ZCMD) VAL(I4)
FIELD(ZPS01008) VAR(ZCMD) VAL(O4)
FIELD(ZPS01009) VAR(ZCMD) VAL(I5)
FIELD(ZPS01010) VAR(ZCMD) VAL(O5)
FIELD(ZPS01011) VAR(ZCMD) VAL(I6)
FIELD(ZPS01012) VAR(ZCMD) VAL(O6)
FIELD(ZPS01013) VAR(ZCMD) VAL(I7)
FIELD(ZPS01014) VAR(ZCMD) VAL(O7)
FIELD(ZPS01015) VAR(ZCMD) VAL(I8)
FIELD(ZPS01016) VAR(ZCMD) VAL(O8)
FIELD(ZPS01017) VAR(ZCMD) VAL(I9)
FIELD(ZPS01018) VAR(ZCMD) VAL(O9)
FIELD(ZPS01019) VAR(ZCMD) VAL(I10)
FIELD(ZPS01020) VAR(ZCMD) VAL(O10)
```

```
FIELD(ZPS01021) VAR(ZCMD) VAL(I11)
FIELD(ZPS01022) VAR(ZCMD) VAL(O11)
FIELD(ZPS01023) VAR(ZCMD) VAL(I12)
FIELD(ZPS01024) VAR(ZCMD) VAL(O12)
FIELD(ZPS01025) VAR(ZCMD) VAL(I13)
FIELD(ZPS01026) VAR(ZCMD) VAL(O13)
FIELD(ZPS01027) VAR(ZCMD) VAL(I14)
FIELD(ZPS01028) VAR(ZCMD) VAL(O14)
FIELD(ZPS01029) VAR(ZCMD) VAL(I15)
FIELD(ZPS01030) VAR(ZCMD) VAL(O15)
FIELD(ZPS01031) VAR(ZCMD) VAL(I16)
FIELD(ZPS01032) VAR(ZCMD) VAL(O16)
FIELD(ZPS01033) VAR(ZCMD) VAL(I17)
FIELD(ZPS01034) VAR(ZCMD) VAL(O17)
FIELD(ZPS01035) VAR(ZCMD) VAL(I18)
FIELD(ZPS01036) VAR(ZCMD) VAL(O18)
FIELD(ZPS01037) VAR(ZCMD) VAL(I19)
FIELD(ZPS01038) VAR(ZCMD) VAL(O19)
FIELD(ZPS01039) VAR(ZCMD) VAL(I20)
FIELD(ZPS01040) VAR(ZCMD) VAL(O20)
FIELD(ZPS01041) VAR(ZCMD) VAL(I21)
FIELD(ZPS01042) VAR(ZCMD) VAL(O21)
FIELD(ZPS01043) VAR(ZCMD) VAL(I22)
FIELD(ZPS01044) VAR(ZCMD) VAL(O22)
FIELD(ZPS01045) VAR(ZCMD) VAL(I23)
FIELD(ZPS01046) VAR(ZCMD) VAL(O23)
FIELD(ZPS01047) VAR(ZCMD) VAL(I24)
FIELD(ZPS01048) VAR(ZCMD) VAL(O24)
FIELD(ZPS01049) VAR(ZCMD) VAL(I25)
FIELD(ZPS01050) VAR(ZCMD) VAL(O25)
FIELD(ZPS01051) VAR(ZCMD) VAL(I26)
FIELD(ZPS01052) VAR(ZCMD) VAL(O26)
FIELD(ZPS01053) VAR(ZCMD) VAL(I27)
FIELD(ZPS01054) VAR(ZCMD) VAL(O27)
FIELD(ZPS01055) VAR(ZCMD) VAL(I28)
FIELD(ZPS01056) VAR(ZCMD) VAL(O28)
FIELD(ZPS01057) VAR(ZCMD) VAL(O29)
FIELD(ZPS01058) VAR(ZCMD) VAL(O30)
FIELD(ZPS01059) VAR(ZCMD) VAL(J1)
FIELD(ZPS01060) VAR(ZCMD) VAL(O31)
FIELD(ZPS01061) VAR(ZCMD) VAL(J2)
FIELD(ZPS01062) VAR(ZCMD) VAL(O32)
FIELD(ZPS01063) VAR(ZCMD) VAL(J3)
FIELD(ZPS01064) VAR(ZCMD) VAL(O33)
FIELD(ZPS01065) VAR(ZCMD) VAL(J4)
FIELD(ZPS01066) VAR(ZCMD) VAL(O34)
FIELD(ZPS01067) VAR(ZCMD) VAL(J5)
FIELD(ZPS01068) VAR(ZCMD) VAL(O35)
FIELD(ZPS01069) VAR(ZCMD) VAL(J6)
```

JCL – ISREMJCL – Panel (📎)

```
FIELD(ZPS01070) VAR(ZCMD) VAL(O36)
FIELD(ZPS01071) VAR(ZCMD) VAL(J7)
FIELD(ZPS01072) VAR(ZCMD) VAL(O37)
FIELD(ZPS01073) VAR(ZCMD) VAL(J8)
FIELD(ZPS01074) VAR(ZCMD) VAL(O38)
FIELD(ZPS01075) VAR(ZCMD) VAL(J9)
FIELD(ZPS01076) VAR(ZCMD) VAL(O39)
FIELD(ZPS01077) VAR(ZCMD) VAL(J10)
FIELD(ZPS01078) VAR(ZCMD) VAL(O40)
FIELD(ZPS01079) VAR(ZCMD) VAL(O41)
FIELD(ZPS01080) VAR(ZCMD) VAL(O42)
FIELD(ZPS01081) VAR(ZCMD) VAL(O43)
FIELD(ZPS01082) VAR(ZCMD) VAL(O44)
FIELD(ZPS01083) VAR(ZCMD) VAL(O45)
FIELD(ZPS01084) VAR(ZCMD) VAL(O46)
FIELD(ZPS01085) VAR(ZCMD) VAL(O47)
FIELD(ZPS01086) VAR(ZCMD) VAL(O48)
FIELD(ZPS01087) VAR(ZCMD) VAL(O49)
FIELD(ZPS01088) VAR(ZCMD) VAL(O50)
FIELD(ZPS01089) VAR(ZCMD) VAL(A1)
)END
/*-------------------------------------------------------------------*/
/*                         Wells Fargo Changes                       */
/*-------------------------------------------------------------------*/
/*  This panel is invoked by using the TSO edit command "MODEL",    */
/*-------------------------------------------------------------------*/
/*  which in turn invokes panel ISREMCLS.  Below is the flow-chart: */
/*                                                                   */
/*      MODEL    --> ISREMCLS   --> ISREMJCL   --> M*******          */
/*      IBM Cmd      IBM panel      User Panel     User Skeleton     */
/*-------------------------------------------------------------------*/
/* FLG  YYMMDD  USERID   DESCRIPTION                                 */
/* ---  ------  -------  ------------------------------------------- */
/* @BG  150414  R2221    Added HISTORY                               */
/* @BF  150121  R2221    Removed UTJP, no longer used/retired        */
/* @BE  141125  R2221    Fixed Point and Shoot                       */
/* @BD  141118  R2221    Added ASMDASM (Disassembler)                */
/* @BC  141007  R2221    Removed FileAid                             */
/* @BB  140630  R2221    Added WWOC                                  */
/* @BA  140527  R2221    Added File Manager                          */
/* @B9  140224  R2221    Added JCLPLUS                               */
/* @B8  131216  R2221    Added EZACFSM1                              */
/* @B7  130620  R2221    Added CINEWCP2                              */
/* @B6  130605  R2221    Added DMS  (CADISK)                         */
/* @B5  121206  R2221    Added GSVXBAT                               */
/* @B4  121106  R2221    Added PDSM27 and PDSM37                     */
/* @B3  120928  R2221    Added SYSLOGSC                              */
/* @B2  120729  R2221    Added FTPSEC                                */
/* @B1  120717  R2221    Added ICETOOL                               */
```

```
/* @AZ  120629  R2221    Added PDSMERGE                              */
/* @AY  120523  R2221    Added MZCCMPB1 and EXAMBATX                 */
/* @AX  120113  R2221    Added AFCLOGR1 and FCSWAIT                  */
/* @AW  120113  R2221    Added DAF                                   */
/* @AU  110602  R2221    Added VSPRIME                               */
/* @AT  110518  R2221    Added PDSM36                                */
/* @AS  110206  R2221    Added PDSMAN programs                       */
/* @AR  100818  R2221    Added BPXBATCH (BPXBATSL)                   */
/* @AQ  100812  R2221    Chged SYSAFF to JOBPARM                     */
/* @AP  100812  R2221    Removed COMMAND Utility                     */
/* @AO  100812  R2221    Added JCL COMMAND, IF, INLUDE and SET       */
/* @AN  100809  R2221    Added COMPAREB and SDEBYSDE                 */
/* @AM  100714  R2221    Removed Insync                              */
/* @AL  100702  R2221    Added PDS                                   */
/* @AK  100630  R2221    Fixed Point-n-shoot                         */
/* @AJ  100628  R2221    Added SYSWAIT                               */
/* @AI  100611  R2221    Chged QW to QWIKREF1                        */
/* @AH  100609  R2221    Added CAL2X2WB, CA7 commands using CCI      */
/* @AG  100513  R2221    Added UTGZ, UTJE/JM/JP, UTKT and UTONY      */
/* @AF  100507  R2221    Added FILEAID adn DJCTEST                   */
/* @AE  100504  R2221    Added IRXJCL and REXXCL                     */
/* @AD  100503  R2221    Fixed ICEMAN and QW when passing as parm    */
/* @AC  100503  R2221    Added Cobol to Eztreive Converter           */
/* @AB  100425  R2221    Added capability for application JCL        */
/* @AA  100419  R2221    Removed NDM, suggest using NDMDSN utility   */
/* @A9  100126  R2221    Added CPPUPDTE                              */
/* @A8  091228  R2221    Added ROUTE card                            */
/* @A7  091202  R2221    Added TASID, SHOWMVS, IFCEREP1, IFASMFDP    */
/* @A6  091006  R2221    Consolidated tape entries into OEM / Misc.  */
/* @A5  090922  R2221    Add INSYNC                                  */
/* @A4  090917  R2221    Add OUTPUT, SYSAFF, and JCLLIB              */
/* @A3  090916  R2221    Add AMATERSE, CONSOLE2, GENMAIL, and SAVRS  */
/* @A2  090915  R2221    Removed invalid software                    */
/* @A1  011025  S915460  Added descriptions to programs              */
/* @A0  020225  S915460  Added FDRMAP                                */
/*-------------------------------------------------------------------*/
```

Complete your session evaluations online at www.SHARE.org/Orlando-Eval

JCL – Skeletons (📎)

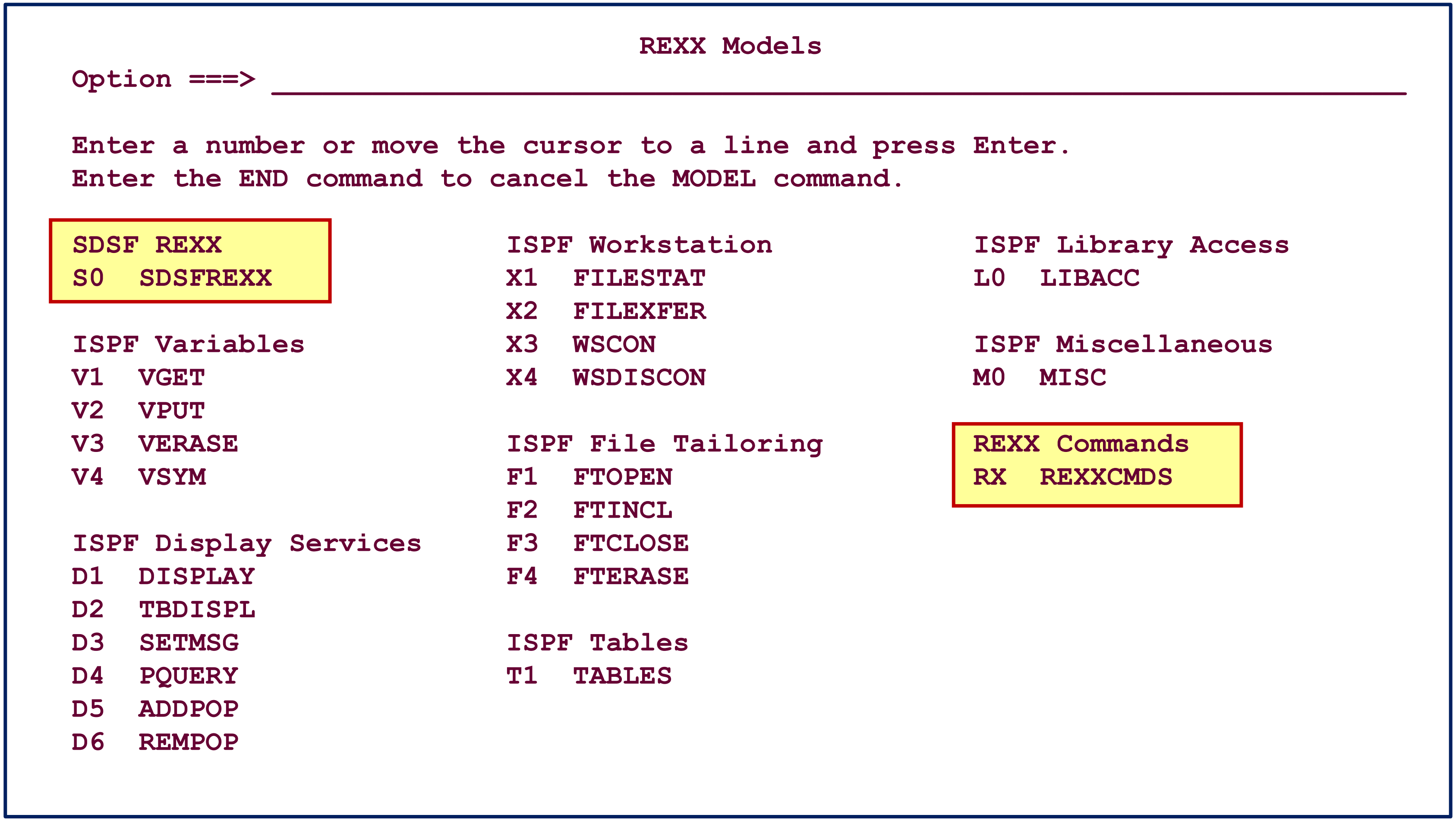
MIEHLIST

```
//*--------------------------------------------------------------------*
//*  PURPOSE:  PDS, VTOC OR CATALOG LISTING
//*--------------------------------------------------------------------*
//*  MANUAL:   DFSMSDFP UTILITIES
//*--------------------------------------------------------------------*
//IEHLIST  EXEC PGM=IEHLIST
//SYSPRINT DD  SYSOUT=*
//DD1      DD  DISP=OLD,UNIT=SYSALLDA,VOL=SER=######
//SYSIN    DD  *
)N*--------------------------------------------------------------------*
)N*  List a PDS directory in hexadecimal format
)N*--------------------------------------------------------------------*
     LISTPDS DSNAME=(******.******.******),                            X
              VOL=SYSDA=######
)N*--------------------------------------------------------------------*
)N*  List a PDS directory attributes (only valid for executables)
)N*--------------------------------------------------------------------*
)N   LISTPDS DSNAME=(******.******.******),FORMAT,                      X
)N            VOL=SYSDA=######
)N
)N*--------------------------------------------------------------------*
)N*  Retrieve a VTOC listing
)N*--------------------------------------------------------------------*
)N   LISTVTOC FORMAT,VOL=DISK=######
)N
)N*--------------------------------------------------------------------*
)N*  Partitioned Dataset (PDS) listing
)N*--------------------------------------------------------------------*
)N   LISTPDS {DSNAME=(dsn1{,dsn2...})}
)N           {,DUMP|FORMAT}
)N           {,VOl=device=serial}
)N
)N*--------------------------------------------------------------------*
)N*  Volume Table of Contents (VTOC) listing
)N*--------------------------------------------------------------------*
)N    LISTVTOC {DUMP|FORMAT}
)N             {,INDEXDSN=SYS1.VTOCIX.xxxxxx}
)N             {,DATE=dddyy}
)N             {,VOl=device=serial}
)N             {,DSNAME=(dsn1{,dsn2...})}
)N
)N*--------------------------------------------------------------------*
```

MUTKT

```
//*--------------------------------------------------------------------*
//*  PURPOSE:  ABEND JOB WITH USER CODE
//*--------------------------------------------------------------------*
//STEP010  EXEC PGM=UTKT,PARM=999,COND=(0,EQ,??????)
)N*--------------------------------------------------------------------*
)N*  Description:
)N*    'UTKT' was written to provide a graceful abend exit
)N*    point for test and production jobs when JCL execution
)N*    needs to be stopped due to prior step execution return
)N*    codes.
)N*
)N*  PARM field:
)N*    The number of characters present in the PARM field
)N*    determines the user abend code to be issued.  In the
)N*    below example, UTKT abends with a user 0003.
)N*
)N*  PARM field limitations:
)N*    - Can contain any combination of alphanumeric and
)N*      national ($,#,@) characters.
)N*    - Up to 100 characters are allowed
)N*--------------------------------------------------------------------*
```

ISREMRXC – Panel (📎)



```
                                  REXX Models
   Option ===> ____________________________________________________

   Enter a number or move the cursor to a line and press Enter.
   Enter the END command to cancel the MODEL command.

   SDSF REXX                  ISPF Workstation           ISPF Library Access
   S0   SDSFREXX              X1   FILESTAT              L0   LIBACC
                              X2   FILEXFER
   ISPF Variables             X3   WSCON                 ISPF Miscellaneous
   V1   VGET                  X4   WSDISCON              M0   MISC
   V2   VPUT
   V3   VERASE                ISPF File Tailoring        REXX Commands
   V4   VSYM                  F1   FTOPEN                RX   REXXCMDS
                              F2   FTINCL
   ISPF Display Services      F3   FTCLOSE
   D1   DISPLAY               F4   FTERASE
   D2   TBDISPL
   D3   SETMSG                ISPF Tables
   D4   PQUERY                T1   TABLES
   D5   ADDPOP
   D6   REMPOP
```

SDSF REXX

- Link
  - IBM System Display and Search Facility Downloads

REXX – Commands – Implementation

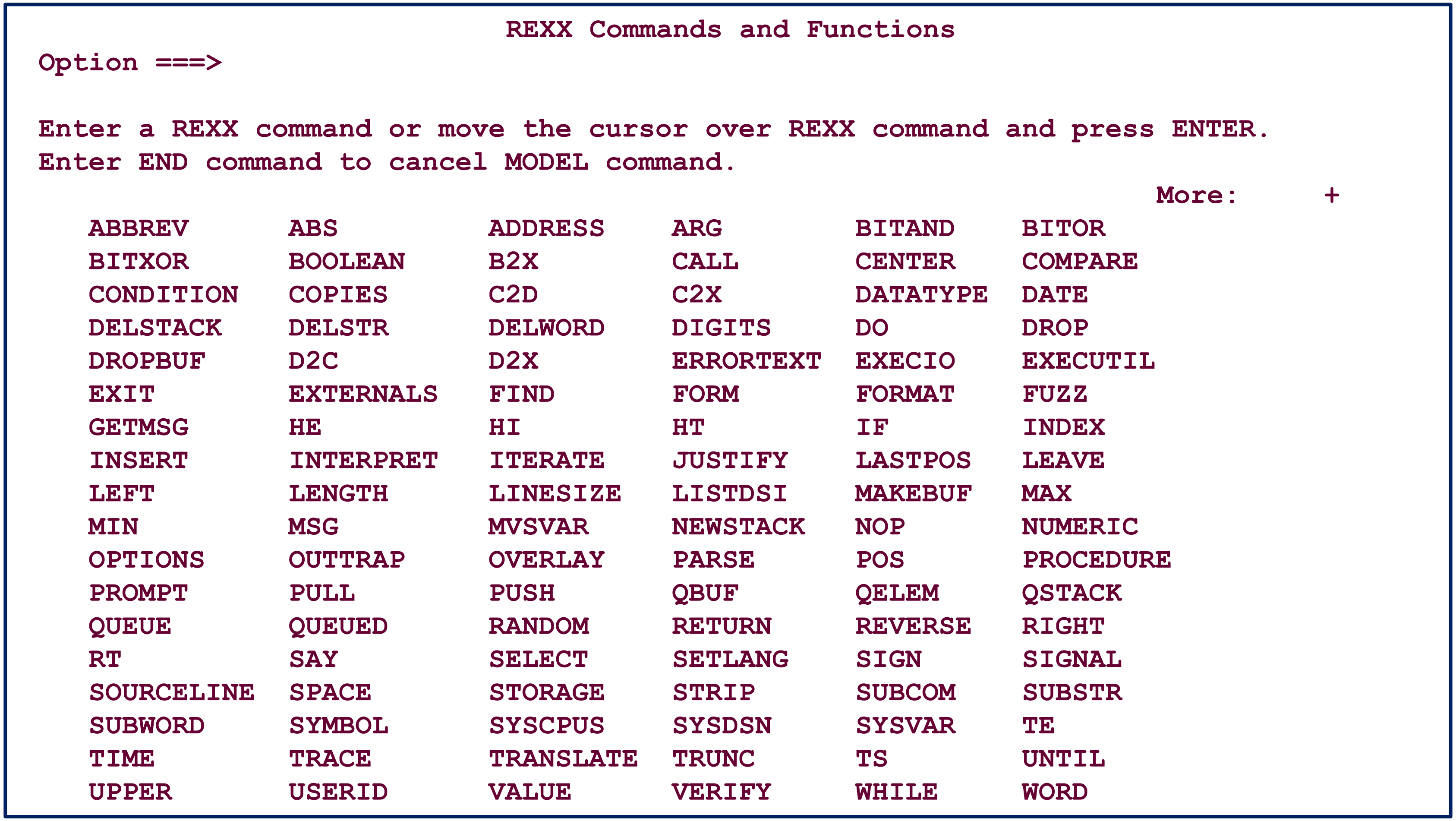
- Create the REXX command panel (ISRRXCMD)

- Invoke MVS/Quick Reference (QW)
  - Select "C", Request Reference Information by Category
  - Select "PROGLANG", Programming Languages Syntax Descriptions
  - Select "Z/OS REXX SYNTAX"
  - Select desired REXX command
  - Type in "CUT ALL"

- Create a new skeleton member
  - Type in QWPASTE
  - Insert ")N" to make all lines notes and adjust text

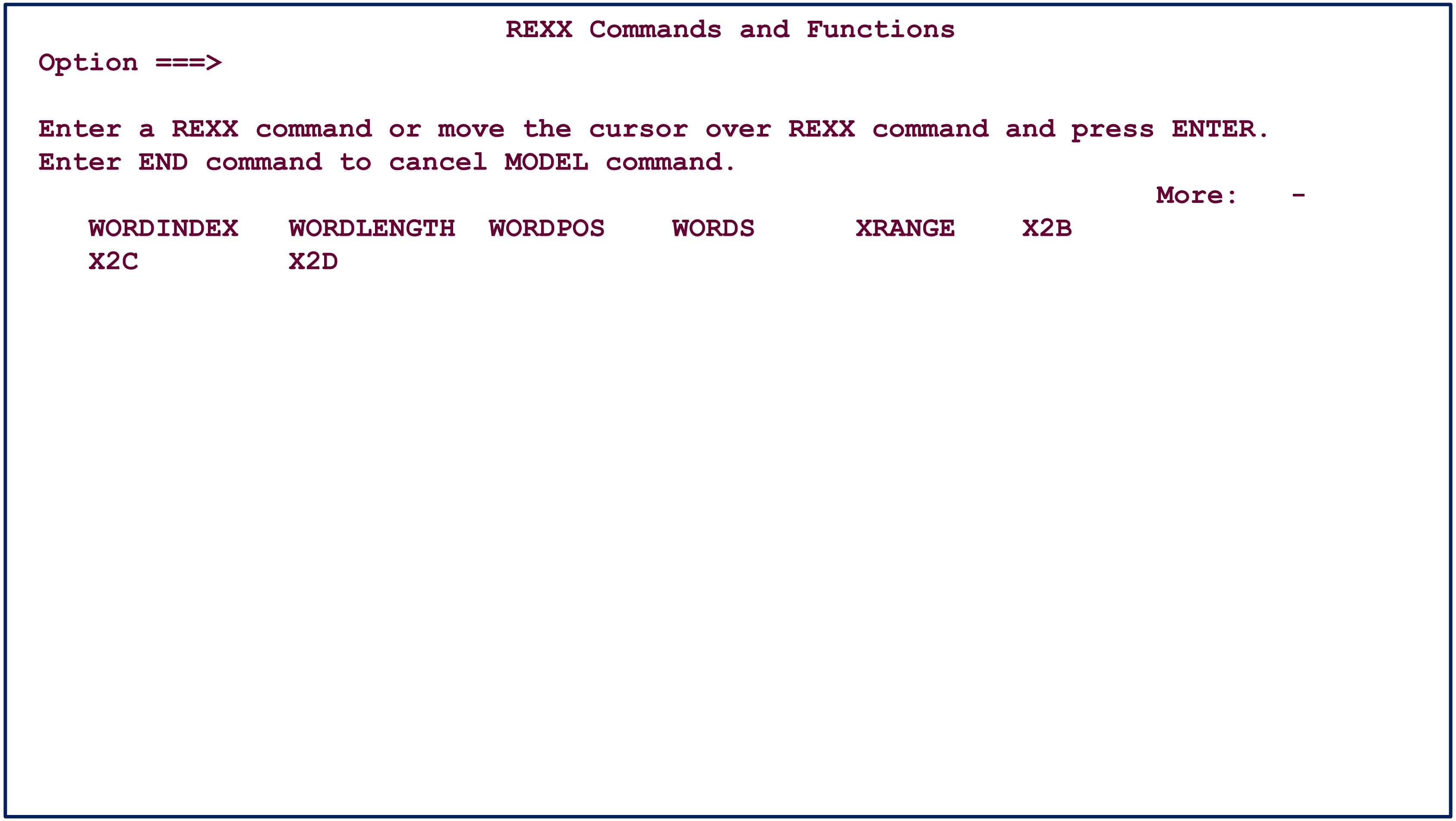
- Add the entry to the REXX command panel (ISRRXCMD)

REXX – ISRRXCMD – Panel (📎)



```
                            REXX Commands and Functions
Option ===>

Enter a REXX command or move the cursor over REXX command and press ENTER.
Enter END command to cancel MODEL command.
                                                                   More:     +
   ABBREV      ABS         ADDRESS    ARG        BITAND    BITOR
   BITXOR      BOOLEAN     B2X        CALL       CENTER    COMPARE
   CONDITION   COPIES      C2D        C2X        DATATYPE  DATE
   DELSTACK    DELSTR      DELWORD    DIGITS     DO        DROP
   DROPBUF     D2C         D2X        ERRORTEXT  EXECIO    EXECUTIL
   EXIT        EXTERNALS   FIND       FORM       FORMAT    FUZZ
   GETMSG      HE          HI         HT         IF        INDEX
   INSERT      INTERPRET   ITERATE    JUSTIFY    LASTPOS   LEAVE
   LEFT        LENGTH      LINESIZE   LISTDSI    MAKEBUF   MAX
   MIN         MSG         MVSVAR     NEWSTACK   NOP       NUMERIC
   OPTIONS     OUTTRAP     OVERLAY    PARSE      POS       PROCEDURE
   PROMPT      PULL        PUSH       QBUF       QELEM     QSTACK
   QUEUE       QUEUED      RANDOM     RETURN     REVERSE   RIGHT
   RT          SAY         SELECT     SETLANG    SIGN      SIGNAL
   SOURCELINE  SPACE       STORAGE    STRIP      SUBCOM    SUBSTR
   SUBWORD     SYMBOL      SYSCPUS    SYSDSN     SYSVAR    TE
   TIME        TRACE       TRANSLATE  TRUNC      TS        UNTIL
   UPPER       USERID      VALUE      VERIFY     WHILE     WORD
```

REXX – ISRRXCMD – Panel (📎)



```
                           REXX Commands and Functions
Option ===>

Enter a REXX command or move the cursor over REXX command and press ENTER.
Enter END command to cancel MODEL command.
                                                                   More:    -
   WORDINDEX    WORDLENGTH  WORDPOS    WORDS      XRANGE    X2B
   X2C          X2D
```

REXX – ISRRXCMD – Panel (📎)

```
)PANEL KEYLIST(ISRSNAB,ISR)
)ATTR DEFAULT(%+_) FORMAT(MIX)            /* ISREMPLI - ENGLISH - 5.0 */
 [  TYPE(PS)
 ]  TYPE(NT)
 0D TYPE(PS)
 05 TYPE(PT)
 06 TYPE(PIN)
 09 TYPE(FP)
 0A TYPE(NT)
 11 TYPE(SAC)
 22 TYPE(WASL) SKIP(ON) GE(ON)
 10 TYPE(ET)
 27 TYPE(NEF) CAPS(ON) PADC(USER)
 28 AREA(SCRL) EXTEND(ON)
)BODY WINDOW(80,22) EXPAND(..) CMD(ZCMD)
.. .REXX Commands and Functions. ..
.Option ===>.Z. .
.
.Enter a REXX command or move the cursor over REXX command and press ENTER.
.Enter.END.command to cancel MODEL command.
.SAREA40                                                                        .
)AREA SAREA40
+  [ABBREV    ][ABS       ][ADDRESS  ][ARG      ][BITAND  ][BITOR   ]
+  [BITXOR    ][BOOLEAN   ][B2X      ][CALL     ][CENTER  ][COMPARE ]
+  [CONDITION ][COPIES    ][C2D      ][C2X      ][DATATYPE][DATE    ]
+  [DELSTACK  ][DELSTR    ][DELWORD  ][DIGITS   ][DO      ][DROP    ]
+  [DROPBUF   ][D2C       ][D2X      ][ERRORTEXT][EXECIO  ][EXECUTIL]
+  [EXIT      ][EXTERNALS ][FIND     ][FORM     ][FORMAT  ][FUZZ    ]
+  [GETMSG    ][HE        ][HI       ][HT       ][IF      ][INDEX   ]
+  [INSERT    ][INTERPRET ][ITERATE  ][JUSTIFY  ][LASTPOS ][LEAVE   ]
+  [LEFT      ][LENGTH    ][LINESIZE ][LISTDSI  ][MAKEBUF ][MAX     ]
+  [MIN       ][MSG       ][MVSVAR   ][NEWSTACK ][NOP     ][NUMERIC ]
+  [OPTIONS   ][OUTTRAP   ][OVERLAY  ][PARSE    ][POS     ][PROCEDURE
+  [PROMPT    ][PULL      ][PUSH     ][QBUF     ][QELEM   ][QSTACK  ]
+  [QUEUE     ][QUEUED    ][RANDOM   ][RETURN   ][REVERSE ][RIGHT   ]
+  [RT        ][SAY       ][SELECT   ][SETLANG  ][SIGN    ][SIGNAL  ]
+  [SOURCELINE][SPACE     ][STORAGE  ][STRIP    ][SUBCOM  ][SUBSTR  ]
+  [SUBWORD   ][SYMBOL    ][SYSCPUS  ][SYSDSN   ][SYSVAR  ][TE      ]
+  [TIME      ][TRACE     ][TRANSLATE][TRUNC    ][TS      ][UNTIL   ]
+  [UPPER     ][USERID    ][VALUE    ][VERIFY   ][WHILE   ][WORD    ]
+  [WORDINDEX ][WORDLENGTH][WORDPOS  ][WORDS    ][XRANGE  ][X2B     ]
+  [X2C       ][X2D       ]
)INIT
.ZVARS = '(ZCMD)'
&ZWINTTL = ' '
.HELP = #ISREMJC
IF (&ISRMDSPL = 'RETURN  ')
    .RESP = END
.CURSOR = 'ZCMD'
```

```
)REINIT
REFRESH(ZCMD)
)PROC
&ZCMD = TRUNC (&ZCMD,'.')
&ZTMPTRL = .TRAIL
IF (&ZCMD ¬= ' ') &ZCMD = '&ZCMD..&ZTMPTRL'
&ZSEL = TRANS (TRUNC (&ZCMD,'.')
      ABBREV        ,'PGM(ISRECMBR) PARM(MRXABBRE)'
      ABS           ,'PGM(ISRECMBR) PARM(MRXABS  )'
      ADDRESS       ,'PGM(ISRECMBR) PARM(MRXADDRE)'
      ARG           ,'PGM(ISRECMBR) PARM(MRXARG  )'
      BITAND        ,'PGM(ISRECMBR) PARM(MRXBITAN)'
      BITOR         ,'PGM(ISRECMBR) PARM(MRXBITOR)'
      BITXOR        ,'PGM(ISRECMBR) PARM(MRXBITXO)'
      BOOLEAN       ,'PGM(ISRECMBR) PARM(MRXBOOLE)'
      B2X           ,'PGM(ISRECMBR) PARM(MRXB2X  )'
      CALL          ,'PGM(ISRECMBR) PARM(MRXCALL )'
      CENTER        ,'PGM(ISRECMBR) PARM(MRXCENTE)'
      COMPARE       ,'PGM(ISRECMBR) PARM(MRXCOMPA)'
      CONDITION     ,'PGM(ISRECMBR) PARM(MRXCONDI)'
      COPIES        ,'PGM(ISRECMBR) PARM(MRXCOPIE)'
      C2D           ,'PGM(ISRECMBR) PARM(MRXC2D  )'
      C2X           ,'PGM(ISRECMBR) PARM(MRXC2X  )'
      DATATYPE      ,'PGM(ISRECMBR) PARM(MRXDATAT)'
      DATE          ,'PGM(ISRECMBR) PARM(MRXDATE )'
      DELSTACK      ,'PGM(ISRECMBR) PARM(MRXDELS )'
      DELSTR        ,'PGM(ISRECMBR) PARM(MRXDELST)'
      DELWORD       ,'PGM(ISRECMBR) PARM(MRXDELWO)'
      DIGITS        ,'PGM(ISRECMBR) PARM(MRXDIGIT)'
      DO            ,'PGM(ISRECMBR) PARM(MRXDO   )'
      DROP          ,'PGM(ISRECMBR) PARM(MRXDROP )'
      DROPBUF       ,'PGM(ISRECMBR) PARM(MRXDROPB)'
      D2C           ,'PGM(ISRECMBR) PARM(MRXD2C  )'
      D2X           ,'PGM(ISRECMBR) PARM(MRXD2X  )'
      ERRORTEXT     ,'PGM(ISRECMBR) PARM(MRXERROR)'
      EXECIO        ,'PGM(ISRECMBR) PARM(MRXEXECI)'
      EXECUTIL      ,'PGM(ISRECMBR) PARM(MRXEXECU)'
      EXIT          ,'PGM(ISRECMBR) PARM(MRXEXIT )'
      EXTERNALS     ,'PGM(ISRECMBR) PARM(MRXEXTER)'
      FIND          ,'PGM(ISRECMBR) PARM(MRXFIND )'
      FORM          ,'PGM(ISRECMBR) PARM(MRXFORM )'
      FORMAT        ,'PGM(ISRECMBR) PARM(MRXFORMA)'
      FUZZ          ,'PGM(ISRECMBR) PARM(MRXFUZZ )'
      GETMSG        ,'PGM(ISRECMBR) PARM(MRXGETMS)'
      HE            ,'PGM(ISRECMBR) PARM(MRXHE   )'
      HI            ,'PGM(ISRECMBR) PARM(MRXHI   )'
      HT            ,'PGM(ISRECMBR) PARM(MRXHT   )'
      IF            ,'PGM(ISRECMBR) PARM(MRXIF   )'
      INDEX         ,'PGM(ISRECMBR) PARM(MRXINDEX)'
```

REXX – ISRRXCMD – Panel (📎)

```
      INSERT        ,'PGM(ISRECMBR) PARM(MRXINSER)'
      INTERPRET     ,'PGM(ISRECMBR) PARM(MRXINTER)'
      ITERATE       ,'PGM(ISRECMBR) PARM(MRXITERA)'
      JUSTIFY       ,'PGM(ISRECMBR) PARM(MRXJUSTI)'
      LASTPOS       ,'PGM(ISRECMBR) PARM(MRXLASTP)'
      LEAVE         ,'PGM(ISRECMBR) PARM(MRXLEAVE)'
      LEFT          ,'PGM(ISRECMBR) PARM(MRXLEFT )'
      LENGTH        ,'PGM(ISRECMBR) PARM(MRXLENGT)'
      LINESIZE      ,'PGM(ISRECMBR) PARM(MRXLINES)'
      LISTDSI       ,'PGM(ISRECMBR) PARM(MRXLISTD)'
      MAKEBUF       ,'PGM(ISRECMBR) PARM(MRXMAKEB)'
      MAX           ,'PGM(ISRECMBR) PARM(MRXMAX  )'
      MIN           ,'PGM(ISRECMBR) PARM(MRXMIN  )'
      MSG           ,'PGM(ISRECMBR) PARM(MRXMSG  )'
      MVSVAR        ,'PGM(ISRECMBR) PARM(MRXMVSVA)'
      NEWSTACK      ,'PGM(ISRECMBR) PARM(MRXNEWST)'
      NOP           ,'PGM(ISRECMBR) PARM(MRXNOP  )'
      NUMERIC       ,'PGM(ISRECMBR) PARM(MRXNUMER)'
      OPTIONS       ,'PGM(ISRECMBR) PARM(MRXOPTIO)'
      OUTTRAP       ,'PGM(ISRECMBR) PARM(MRXOUTTR)'
      OVERLAY       ,'PGM(ISRECMBR) PARM(MRXOVERL)'
      PARSE         ,'PGM(ISRECMBR) PARM(MRXPARSE)'
      POS           ,'PGM(ISRECMBR) PARM(MRXPOS  )'
      PROCEDURE     ,'PGM(ISRECMBR) PARM(MRXPROCE)'
      PROMPT        ,'PGM(ISRECMBR) PARM(MRXPROMP)'
      PULL          ,'PGM(ISRECMBR) PARM(MRXPULL )'
      PUSH          ,'PGM(ISRECMBR) PARM(MRXPUSH )'
      QBUF          ,'PGM(ISRECMBR) PARM(MRXQBUF )'
      QELEM         ,'PGM(ISRECMBR) PARM(MRXQELEM)'
      QSTACK        ,'PGM(ISRECMBR) PARM(MRXQSTAC)'
      QUEUE         ,'PGM(ISRECMBR) PARM(MRXQUEUE)'
      QUEUED        ,'PGM(ISRECMBR) PARM(MRXQUEUD)'
      RANDOM        ,'PGM(ISRECMBR) PARM(MRXRANDO)'
      RETURN        ,'PGM(ISRECMBR) PARM(MRXRETUR)'
      REVERSE       ,'PGM(ISRECMBR) PARM(MRXREVER)'
      RIGHT         ,'PGM(ISRECMBR) PARM(MRXRIGHT)'
      RT            ,'PGM(ISRECMBR) PARM(MRXRT   )'
      SAY           ,'PGM(ISRECMBR) PARM(MRXSAY  )'
      SELECT        ,'PGM(ISRECMBR) PARM(MRXSELEC)'
      SETLANG       ,'PGM(ISRECMBR) PARM(MRXSETLA)'
      SIGN          ,'PGM(ISRECMBR) PARM(MRXSIGN )'
      SIGNAL        ,'PGM(ISRECMBR) PARM(MRXSIGNA)'
      SOURCELINE    ,'PGM(ISRECMBR) PARM(MRXSOURC)'
      SPACE         ,'PGM(ISRECMBR) PARM(MRXSPACE)'
      STORAGE       ,'PGM(ISRECMBR) PARM(MRXSTORA)'
      STRIP         ,'PGM(ISRECMBR) PARM(MRXSTRIP)'
      SUBCOM        ,'PGM(ISRECMBR) PARM(MRXSUBCO)'
      SUBSTR        ,'PGM(ISRECMBR) PARM(MRXSUBST)'
      SUBWORD       ,'PGM(ISRECMBR) PARM(MRXSUBWO)'
```

```
      SYMBOL        ,'PGM(ISRECMBR) PARM(MRXSYMBO)'
      SYSCPUS       ,'PGM(ISRECMBR) PARM(MRXSYSCP)'
      SYSDSN        ,'PGM(ISRECMBR) PARM(MRXSYSDS)'
      SYSVAR        ,'PGM(ISRECMBR) PARM(MRXSYSVA)'
      TE            ,'PGM(ISRECMBR) PARM(MRXTE   )'
      TIME          ,'PGM(ISRECMBR) PARM(MRXTIME )'
      TRACE         ,'PGM(ISRECMBR) PARM(MRXTRACE)'
      TRANSLATE     ,'PGM(ISRECMBR) PARM(MRXTRANS)'
      TRUNC         ,'PGM(ISRECMBR) PARM(MRXTRUNC)'
      TS            ,'PGM(ISRECMBR) PARM(MRXTS   )'
      UNTIL         ,'PGM(ISRECMBR) PARM(MRXUNTIL)'
      UPPER         ,'PGM(ISRECMBR) PARM(MRXUPPER)'
      USERID        ,'PGM(ISRECMBR) PARM(MRXUSERI)'
      VALUE         ,'PGM(ISRECMBR) PARM(MRXVALUE)'
      VERIFY        ,'PGM(ISRECMBR) PARM(MRXVERIF)'
      WHILE         ,'PGM(ISRECMBR) PARM(MRXWHILE)'
      WORD          ,'PGM(ISRECMBR) PARM(MRXWORD )'
      WORDINDEX     ,'PGM(ISRECMBR) PARM(MRXWORDI)'
      WORDLENGTH    ,'PGM(ISRECMBR) PARM(MRXWORDL)'
      WORDPOS       ,'PGM(ISRECMBR) PARM(MRXWORDP)'
      WORDS         ,'PGM(ISRECMBR) PARM(MRXWORDS)'
      XRANGE        ,'PGM(ISRECMBR) PARM(MRXXRANG)'
      X2B           ,'PGM(ISRECMBR) PARM(MRXX2B  )'
      X2C           ,'PGM(ISRECMBR) PARM(MRXX2C  )'
      X2D           ,'PGM(ISRECMBR) PARM(MRXX2D  )'
      ' ',' '
        *,'?' )
IF (&ZSEL = '?')
   &ZEMTEMP = &ZCMD
   .MSG = ISRYM013
   &ZCMD = ' '
   IF (&ZMACRO = 'Y')               /* If a macro command    @OY38423*/
 .RESP = 'END'                  /* don't display error   @OY38423*/
     &ISRMMEMB = '?'                /* Indicate no member fnd@OY38423*/
&ISRMMEND = 'N'                     /* SET THE END INDICATOR TO NO   */
IF (.RESP = END)                    /* IF ENDING, WHY ... WHO CAUSED */
  IF (&ISRMONCL ¬= 'Y')             /* MAKE SURE ITS NOT A CLASS OP. */
    IF (&ISRMDSPL ¬= 'RETURN  ')    /* MAKE SURE ITS NOT END ON MBR. */
      &ISRMMEND = 'Y'               /* NO - ITS BECAUSE USER HIT END */
&ZMSG = .MSG                        /* Set message variable          */
VPUT(ZMSG) PROFILE                  /* Save message in prof pool     */
)PNTS
FIELD(ZPS01001) VAR(ZCMD) VAL(ABBREV    )
FIELD(ZPS01002) VAR(ZCMD) VAL(ABS       )
FIELD(ZPS01003) VAR(ZCMD) VAL(ADDRESS   )
FIELD(ZPS01004) VAR(ZCMD) VAL(ARG       )
FIELD(ZPS01005) VAR(ZCMD) VAL(BITAND    )
FIELD(ZPS01006) VAR(ZCMD) VAL(BITOR     )
FIELD(ZPS01007) VAR(ZCMD) VAL(BITXOR    )
```

REXX – ISRRXCMD – Panel (📎)

```
FIELD(ZPS01008)  VAR(ZCMD)  VAL(BOOLEAN   )
FIELD(ZPS01009)  VAR(ZCMD)  VAL(B2X       )
FIELD(ZPS01010)  VAR(ZCMD)  VAL(CALL      )
FIELD(ZPS01011)  VAR(ZCMD)  VAL(CENTER    )
FIELD(ZPS01012)  VAR(ZCMD)  VAL(COMPARE   )
FIELD(ZPS01013)  VAR(ZCMD)  VAL(CONDITION )
FIELD(ZPS01014)  VAR(ZCMD)  VAL(COPIES    )
FIELD(ZPS01015)  VAR(ZCMD)  VAL(C2D       )
FIELD(ZPS01016)  VAR(ZCMD)  VAL(C2X       )
FIELD(ZPS01017)  VAR(ZCMD)  VAL(DATATYPE  )
FIELD(ZPS01018)  VAR(ZCMD)  VAL(DATE      )
FIELD(ZPS01019)  VAR(ZCMD)  VAL(DELSTACK  )
FIELD(ZPS01020)  VAR(ZCMD)  VAL(DELSTR    )
FIELD(ZPS01021)  VAR(ZCMD)  VAL(DELWORD   )
FIELD(ZPS01022)  VAR(ZCMD)  VAL(DIGITS    )
FIELD(ZPS01023)  VAR(ZCMD)  VAL(DO        )
FIELD(ZPS01024)  VAR(ZCMD)  VAL(DROP      )
FIELD(ZPS01025)  VAR(ZCMD)  VAL(DROPBUF   )
FIELD(ZPS01026)  VAR(ZCMD)  VAL(D2C       )
FIELD(ZPS01027)  VAR(ZCMD)  VAL(D2X       )
FIELD(ZPS01028)  VAR(ZCMD)  VAL(ERRORTEXT )
FIELD(ZPS01029)  VAR(ZCMD)  VAL(EXECIO    )
FIELD(ZPS01030)  VAR(ZCMD)  VAL(EXECUTIL  )
FIELD(ZPS01031)  VAR(ZCMD)  VAL(EXIT      )
FIELD(ZPS01032)  VAR(ZCMD)  VAL(EXTERNALS )
FIELD(ZPS01033)  VAR(ZCMD)  VAL(FIND      )
FIELD(ZPS01034)  VAR(ZCMD)  VAL(FORM      )
FIELD(ZPS01035)  VAR(ZCMD)  VAL(FORMAT    )
FIELD(ZPS01036)  VAR(ZCMD)  VAL(FUZZ      )
FIELD(ZPS01037)  VAR(ZCMD)  VAL(GETMSG    )
FIELD(ZPS01038)  VAR(ZCMD)  VAL(HE        )
FIELD(ZPS01039)  VAR(ZCMD)  VAL(HI        )
FIELD(ZPS01040)  VAR(ZCMD)  VAL(HT        )
FIELD(ZPS01041)  VAR(ZCMD)  VAL(IF        )
FIELD(ZPS01042)  VAR(ZCMD)  VAL(INDEX     )
FIELD(ZPS01043)  VAR(ZCMD)  VAL(INSERT    )
FIELD(ZPS01044)  VAR(ZCMD)  VAL(INTERPRET )
FIELD(ZPS01045)  VAR(ZCMD)  VAL(ITERATE   )
FIELD(ZPS01046)  VAR(ZCMD)  VAL(JUSTIFY   )
FIELD(ZPS01047)  VAR(ZCMD)  VAL(LASTPOS   )
FIELD(ZPS01048)  VAR(ZCMD)  VAL(LEAVE     )
FIELD(ZPS01049)  VAR(ZCMD)  VAL(LEFT      )
FIELD(ZPS01050)  VAR(ZCMD)  VAL(LENGTH    )
FIELD(ZPS01051)  VAR(ZCMD)  VAL(LINESIZE  )
FIELD(ZPS01052)  VAR(ZCMD)  VAL(LISTDSI   )
FIELD(ZPS01053)  VAR(ZCMD)  VAL(MAKEBUF   )
FIELD(ZPS01054)  VAR(ZCMD)  VAL(MAX       )
FIELD(ZPS01055)  VAR(ZCMD)  VAL(MIN       )
FIELD(ZPS01056)  VAR(ZCMD)  VAL(MSG       )
```

```
FIELD(ZPS01057)  VAR(ZCMD)  VAL(MVSVAR    )
FIELD(ZPS01058)  VAR(ZCMD)  VAL(NEWSTACK  )
FIELD(ZPS01059)  VAR(ZCMD)  VAL(NOP       )
FIELD(ZPS01060)  VAR(ZCMD)  VAL(NUMERIC   )
FIELD(ZPS01061)  VAR(ZCMD)  VAL(OPTIONS   )
FIELD(ZPS01062)  VAR(ZCMD)  VAL(OUTTRAP   )
FIELD(ZPS01063)  VAR(ZCMD)  VAL(OVERLAY   )
FIELD(ZPS01064)  VAR(ZCMD)  VAL(PARSE     )
FIELD(ZPS01065)  VAR(ZCMD)  VAL(POS       )
FIELD(ZPS01066)  VAR(ZCMD)  VAL(PROCEDURE )
FIELD(ZPS01067)  VAR(ZCMD)  VAL(PROMPT    )
FIELD(ZPS01068)  VAR(ZCMD)  VAL(PULL      )
FIELD(ZPS01069)  VAR(ZCMD)  VAL(PUSH      )
FIELD(ZPS01070)  VAR(ZCMD)  VAL(QBUF      )
FIELD(ZPS01071)  VAR(ZCMD)  VAL(QELEM     )
FIELD(ZPS01072)  VAR(ZCMD)  VAL(QSTACK    )
FIELD(ZPS01073)  VAR(ZCMD)  VAL(QUEUE     )
FIELD(ZPS01074)  VAR(ZCMD)  VAL(QUEUED    )
FIELD(ZPS01075)  VAR(ZCMD)  VAL(RANDOM    )
FIELD(ZPS01076)  VAR(ZCMD)  VAL(RETURN    )
FIELD(ZPS01077)  VAR(ZCMD)  VAL(REVERSE   )
FIELD(ZPS01078)  VAR(ZCMD)  VAL(RIGHT     )
FIELD(ZPS01079)  VAR(ZCMD)  VAL(RT        )
FIELD(ZPS01080)  VAR(ZCMD)  VAL(SAY       )
FIELD(ZPS01081)  VAR(ZCMD)  VAL(SELECT    )
FIELD(ZPS01082)  VAR(ZCMD)  VAL(SETLANG   )
FIELD(ZPS01083)  VAR(ZCMD)  VAL(SIGN      )
FIELD(ZPS01084)  VAR(ZCMD)  VAL(SIGNAL    )
FIELD(ZPS01085)  VAR(ZCMD)  VAL(SOURCELINE)
FIELD(ZPS01086)  VAR(ZCMD)  VAL(SPACE     )
FIELD(ZPS01087)  VAR(ZCMD)  VAL(STORAGE   )
FIELD(ZPS01088)  VAR(ZCMD)  VAL(STRIP     )
FIELD(ZPS01089)  VAR(ZCMD)  VAL(SUBCOM    )
FIELD(ZPS01090)  VAR(ZCMD)  VAL(SUBSTR    )
FIELD(ZPS01091)  VAR(ZCMD)  VAL(SUBWORD   )
FIELD(ZPS01092)  VAR(ZCMD)  VAL(SYMBOL    )
FIELD(ZPS01093)  VAR(ZCMD)  VAL(SYSCPUS   )
FIELD(ZPS01094)  VAR(ZCMD)  VAL(SYSDSN    )
FIELD(ZPS01095)  VAR(ZCMD)  VAL(SYSVAR    )
FIELD(ZPS01096)  VAR(ZCMD)  VAL(TE        )
FIELD(ZPS01097)  VAR(ZCMD)  VAL(TIME      )
FIELD(ZPS01098)  VAR(ZCMD)  VAL(TRACE     )
FIELD(ZPS01099)  VAR(ZCMD)  VAL(TRANSLATE )
FIELD(ZPS01100)  VAR(ZCMD)  VAL(TRUNC     )
FIELD(ZPS01101)  VAR(ZCMD)  VAL(TS        )
FIELD(ZPS01102)  VAR(ZCMD)  VAL(UNTIL     )
FIELD(ZPS01103)  VAR(ZCMD)  VAL(UPPER     )
FIELD(ZPS01104)  VAR(ZCMD)  VAL(USERID    )
FIELD(ZPS01105)  VAR(ZCMD)  VAL(VALUE     )
```

REXX – ISRRXCMD – Panel ()

```
FIELD(ZPS01106) VAR(ZCMD) VAL(VERIFY    )
FIELD(ZPS01107) VAR(ZCMD) VAL(WHILE     )
FIELD(ZPS01108) VAR(ZCMD) VAL(WORD      )
FIELD(ZPS01109) VAR(ZCMD) VAL(WORDINDEX )
FIELD(ZPS01110) VAR(ZCMD) VAL(WORDLENGTH)
FIELD(ZPS01111) VAR(ZCMD) VAL(WORDPOS   )
FIELD(ZPS01112) VAR(ZCMD) VAL(WORDS     )
FIELD(ZPS01113) VAR(ZCMD) VAL(XRANGE    )
FIELD(ZPS01114) VAR(ZCMD) VAL(X2B       )
FIELD(ZPS01115) VAR(ZCMD) VAL(X2C       )
FIELD(ZPS01116) VAR(ZCMD) VAL(X2D       )
)END
/*-------------------------------------------------------------------*/
/*                           Wells Fargo Changes                     */
/*-------------------------------------------------------------------*/
/*  This panel is invoked by using the TSO edit command "MODEL",     */
/*  which in turn invokes panel ISREMCLS.  Below is the flow-chart:  */
/*                                                                   */
/*      MODEL    --> ISREMRXC   --> ISRRXCMD   --> M*******          */
/*      IBM Cmd      IBM panel      User Panel     User Skeleton     */
/*-------------------------------------------------------------------*/
/* FLG  YYMMDD  USERID   DESCRIPTION                                 */
/* ---  ------  -------  ------------------------------------------- */
/* @A0  100630  R2221    Developed Panel                             */
/*-------------------------------------------------------------------*/
```

REXX – Skeletons (📎)

MRXPOS

```
)NPOS (Position)
)N
)N+--------------------------------------------------------------------------+
)N|                                                                          |
)N| >>--POS(needle,haystack--------------)------------------------------>< |
)N|                            +-,start-+                                    |
)N|                                                                          |
)N+--------------------------------------------------------------------------+
)N
)NReturns the position of one string, needle, in another haystack.
)N(See also the INDEX and LASTPOS functions.) Returns 0 if needle is
)Nthe null string or is not found or if start is greater than the
)Nlength of haystack.  By default the search starts at the first
)Ncharacter of haystack (that is, the value of start is 1).  You can
)Noverride this by specifying start (which must be a positive whole
)Nnumber), the point at which the search starts.
)N
)NHere are some examples:
)N
)N  POS('day','Saturday')       ->    6
)N  POS('x','abc def ghi')      ->    0
)N  POS(' ','abc def ghi')      ->    4
)N  POS(' ','abc def ghi',5)    ->    8
```

MRXSAY

```
)NSAY
)N
)N+--------------------------------------------------------------------------+
)N|                                                                          |
)N| >>--SAY---------------;------------------------------------------->< |
)N|         +-expression-+                                                   |
)N|                                                                          |
)N+--------------------------------------------------------------------------+
)N
)NSAY writes a line to the output stream.  This typically displays it
)Nto the user, but the output destination can depend on the
)Nimplementation.  The result of expression may be of any length.  If
)Nyou omit expression, the null string is written.
)N
)NIf a REXX exec runs in TSO/E foreground, SAY displays the expression
)Non the terminal.  The result from the SAY instruction is formatted
)Nto the current terminal line width (as defined by the TSO/E TERMINAL
)Ncommand) minus 1 character.  In TSO/E background, SAY writes the
)Nexpression to the output stream, which is SYSTSPRT.  In either case,
)Nwhen the length is undefined (LINESIZE() returns 0), SAY uses a
)Ndefault line size of 80.
)N
)NIf an exec runs in a non-TSO/E address space, SAY writes the
)Nexpression to the output stream as defined by the OUTDD field in the
)Nmodule name table (see item -MODNAMET).  The system default is
)NSYSTSPRT.  The ddname may be change on an application basis or on a
)Nsystem basis.
)N
)NExample:
)N
)N  data=100
)N  Say data 'divided by 4 =>' data/4
)N  /* Displays: "100 divided by 4 => 25"  */
```

Work Station Agent

Introduction

- Work Station Agent (WSA) was introduced in ISPF V4 in MVS/ESA (1994)

- WSA is most useful for customers with the following tendencies:
  - Have a lot of screen real estate, several monitors connected to their PC
  - Prefer to use one editor to modify and view mainframe/PC files
  - Frequently download/upload files from their workstation

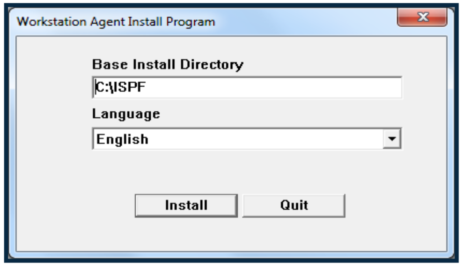
- Links
  - SHARE, Session 16665, March 2015, Bruce Koss
    - ISPF – Workstation Agent (WSA)
  - TSO Times, Spring 2004, Dave Salt
    - Performing File Transfer using ISPF Work Station Agent (WSA)
  - Appendix A, Installing the Client/Server Component
    - ISPF – V2R1 – Users Guide – Vol 1

Pros & Cons

- Pros
  - 32 separate/individual GUI split screens per TSO session
  - Easily transfer all members in a PDS file to a folder
- Cons
  - Work Station Agent must be active
  - Can not initiate the GUI display while in split screen mode
  - Screens are black and white, not colored, and commands like FIND/CHANGE do not highlight or underline the text string
    - Character-level color, intensity and highlighting
    - Field-level intensity and highlighting
  - Maximum fixed window size, can not enlarge to full screen mode
    - Must change the Workstation Agent font size to enlarge the window
    - Window can be resized smaller, but hides the screen
  - The 'Workstation File Name' field is not EXPANDable on ISPF screens
    - Prevents typing in large directory and file names

Install on PC

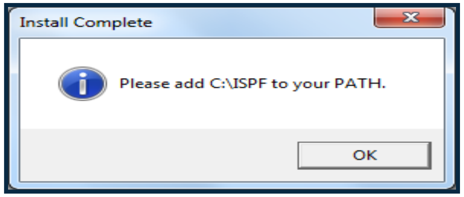
- Download (binary) the following PDS member into a Windows folder called "C:\ISPF" and add the extension '.EXE'
  - ISP.ISPGUI(ISPGUINX) or
  - SYS1.SISPGUI(ISPGUINX)
- Double click on "C:\ISPF\ISPGUINX.EXE"
  - Type in "C:\ISPF" and click the "Install" button
  - Disregard the message and click the "OK" button
  - Click the "QUIT" button



Workstation Agent Install Program

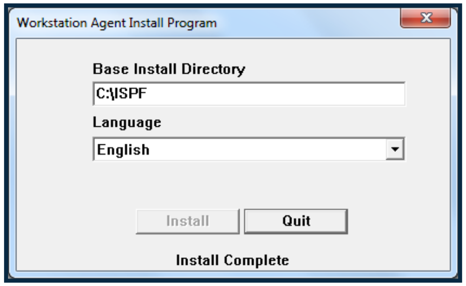
Base Install Directory: C:\ISPF

Language: English



Install Complete

Please add C:\ISPF to your PATH.



Workstation Agent Install Program

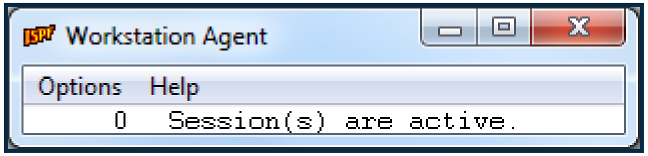
Base Install Directory: C:\ISPF

Language: English

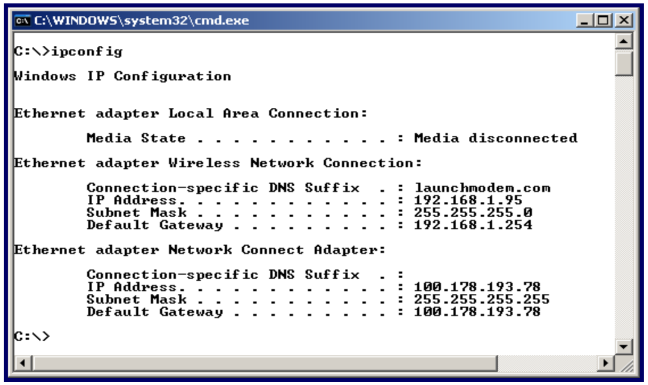
Install Complete

Activation

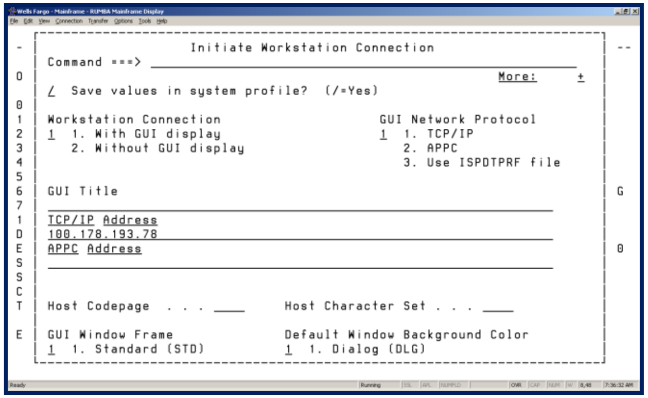
- Once installed, click on “WSA.exe”
- Obtain the workstation IP address
  - From Windows
    - Click “Start”
    - Select “Run”
    - Insert “CMD”
    - Type in “IPCONFIG”
- From ISPF type in “WSCON”
  - Type “1” for “Workstation Connection”
  - Insert “TCP/IP Address”



```
0   Session(s) are active.
```



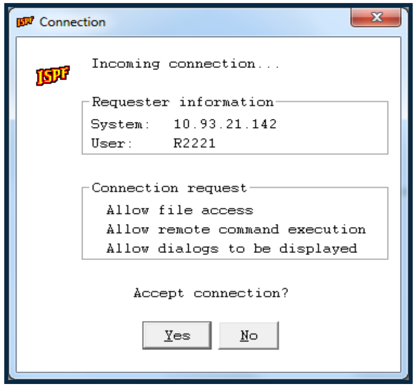
```
C:\>ipconfig

Windows IP Configuration


Ethernet adapter Local Area Connection:

        Media State . . . . . . . . . . . : Media disconnected

Ethernet adapter Wireless Network Connection:

        Connection-specific DNS Suffix  . : launchmodem.com
        IP Address. . . . . . . . . . . . : 192.168.1.95
        Subnet Mask . . . . . . . . . . . : 255.255.255.0
        Default Gateway . . . . . . . . . : 192.168.1.254

Ethernet adapter Network Connect Adapter:

        Connection-specific DNS Suffix  . :
        IP Address. . . . . . . . . . . . : 100.178.193.78
        Subnet Mask . . . . . . . . . . . : 255.255.255.255
        Default Gateway . . . . . . . . . : 100.178.193.78

C:\>
```



```
                 Initiate Workstation Connection
Command ===> ____________________________________________
                                                   More:     +
/  Save values in system profile?  (/=Yes)

Workstation Connection                 GUI Network Protocol
1  1. With GUI display                 1  1. TCP/IP
   2. Without GUI display                 2. APPC
                                          3. Use ISPDTPRF file

GUI Title
__________________________________________________________
TCP/IP Address
100.178.193.78____________________________________________
APPC Address
__________________________________________________________

Host Codepage  . . . ____     Host Character Set . . . ____

GUI Window Frame              Default Window Background Color
1  1. Standard (STD)          1  1. Dialog (DLG)
```

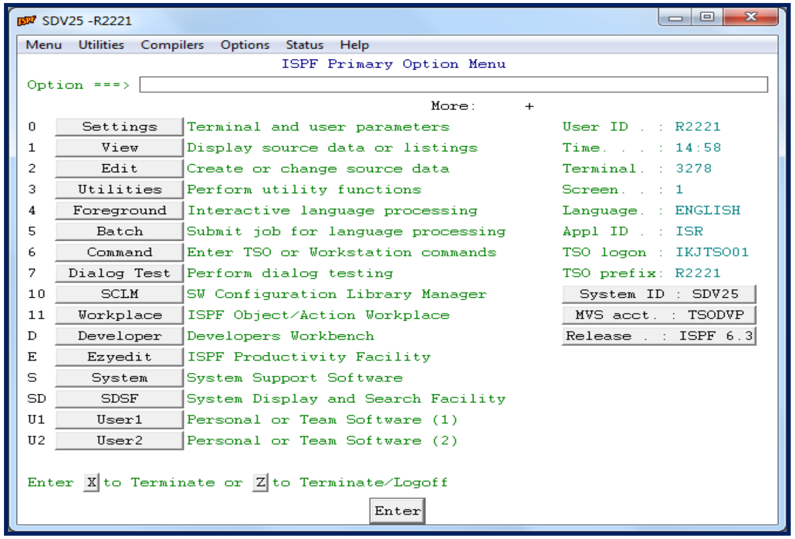
Activation

- Windows “Connection” screen will appear
  - Click the “Yes” button



```
Incoming connection...

Requester information
System:    10.93.21.142
User:      R2221

Connection request
  Allow file access
  Allow remote command execution
  Allow dialogs to be displayed

Accept connection?
```

- ISPF in GUI mode



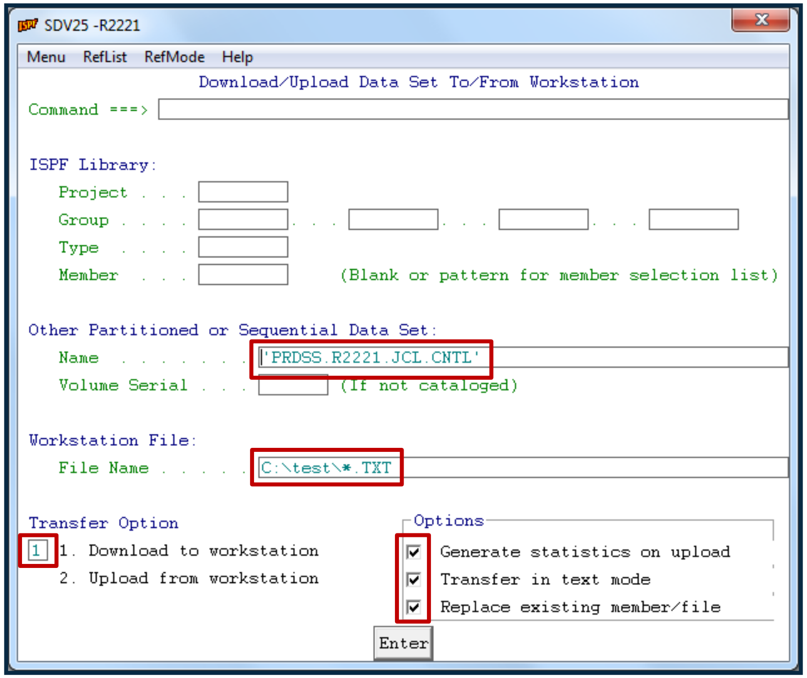
```
                    ISPF Primary Option Menu
Option ===>
                                        More:     +
0  Settings     Terminal and user parameters          User ID . : R2221
1  View         Display source data or listings       Time. . . : 14:58
2  Edit         Create or change source data          Terminal. : 3278
3  Utilities    Perform utility functions             Screen. . : 1
4  Foreground   Interactive language processing       Language. : ENGLISH
5  Batch        Submit job for language processing    Appl ID . : ISR
6  Command      Enter TSO or Workstation commands     TSO logon : IKJTSO01
7  Dialog Test  Perform dialog testing                TSO prefix: R2221
10 SCLM         SW Configuration Library Manager      System ID : SDV25
11 Workplace    ISPF Object/Action Workplace          MVS acct. : TSODVP
D  Developer    Developers Workbench                  Release . : ISPF 6.3
E  Ezyedit      ISPF Productivity Facility
S  System       System Support Software
SD SDSF         System Display and Search Facility
U1 User1        Personal or Team Software (1)
U2 User2        Personal or Team Software (2)

Enter X to Terminate or Z to Terminate/Logoff
```

Commands

- SPLIT or START creates new GUI screen
- SWITCH [3270|GUI] between modes
- WSDISCON terminates GUI session

ISPF 3.7.2 – Transfer Files

- ISPF 3.7.2 is an easy way to transfer PDS members to a PC folder



Download/Upload Data Set To/From Workstation

Command ===>

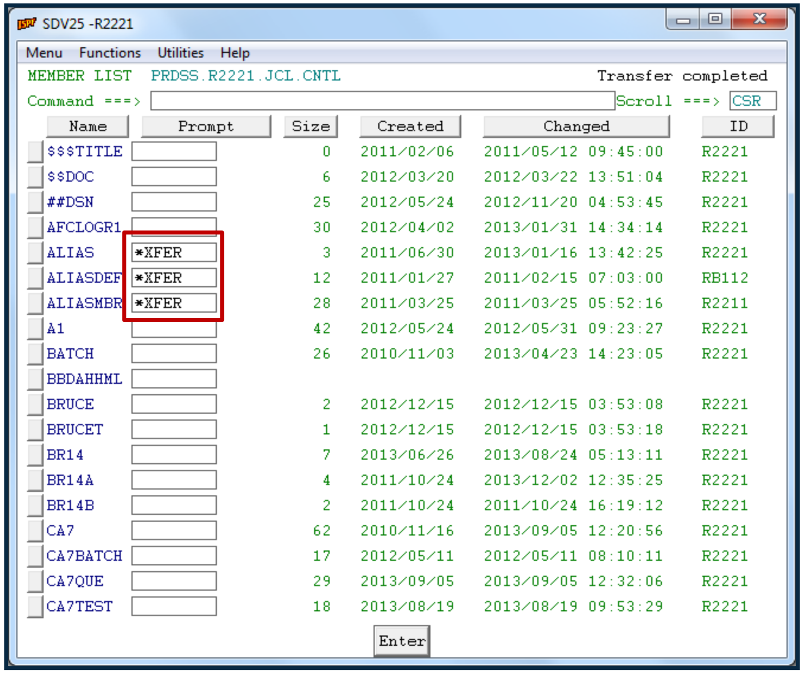
ISPF Library:
Project . . .
Group . . . .
Type . . . .
Member . . . (Blank or pattern for member selection list)

Other Partitioned or Sequential Data Set:
Name . . . . 'PRDSS.R2221.JCL.CNTL'
Volume Serial . . . (If not cataloged)

Workstation File:
File Name . . . . . C:\test\*.TXT

Transfer Option
1 1. Download to workstation
2. Upload from workstation

Options
☑ Generate statistics on upload
☑ Transfer in text mode
☑ Replace existing member/file



MEMBER LIST PRDSS.R2221.JCL.CNTL Transfer completed

Command ===> Scroll ===> CSR

Name	Prompt	Size	Created	Changed	ID
$$$TITLE		0	2011/02/06	2011/05/12 09:45:00	R2221
$$DOC		6	2012/03/20	2012/03/22 13:51:04	R2221
##DSN		25	2012/05/24	2012/11/20 04:53:45	R2221
AFCLOGR1		30	2012/04/02	2013/01/31 14:34:14	R2221
ALIAS	*XFER	3	2011/06/30	2013/01/16 13:42:25	R2221
ALIASDEF	*XFER	12	2011/01/27	2011/02/15 07:03:00	RB112
ALIASMBR	*XFER	28	2011/03/25	2011/03/25 05:52:16	R2211
A1		42	2012/05/24	2012/05/31 09:23:27	R2221
BATCH		26	2010/11/03	2013/04/23 14:23:05	R2221
BBDAHHML					
BRUCE		2	2012/12/15	2012/12/15 03:53:08	R2221
BRUCET		1	2012/12/15	2012/12/15 03:53:18	R2221
BR14		7	2013/06/26	2013/08/24 05:13:11	R2221
BR14A		4	2011/10/24	2013/12/02 12:35:25	R2221
BR14B		2	2011/10/24	2011/10/24 16:19:12	R2221
CA7		62	2010/11/16	2013/09/05 12:20:56	R2221
CA7BATCH		17	2012/05/11	2012/05/11 08:10:11	R2221
CA7QUE		29	2013/09/05	2013/09/05 12:32:06	R2221
CA7TEST		18	2013/08/19	2013/08/19 09:53:29	R2221

Links

Manuals – ISPF

Version	Manual	Link
Website	N/A	ISPF – Manuals and Tools
Red Manual	SG24-6981-02	ABC’s of Z/OS Systems Programming – Vol 1
V2R01	SC19-3619-00	ISPF – Dialog Developer's Guide and Reference
V2R01	SC19-3620-00	ISPF – Dialog Tag Language Guide and Reference
V2R01	SC19-3621-00	ISPF – Edit and Edit Macros
V2R01	SC19-3622-00	ISPF – Messages and Codes
V2R01	GC19-3623-00	ISPF – Planning and Customizing
V2R01	SC19-3624-00	ISPF – Reference Summary
V2R01	SC19-3625-00	ISPF – SCLM Guide and Reference
V2R01	SC19-3626-00	ISPF – Services Guide
V2R01	SC19-3627-00	ISPF – User's Guide – Vol 1
V2R01	SC19-3628-00	ISPF – User's Guide – Vol 2
V1R13	SC34-4821-09	ISPF – Dialog Developer's Guide and Reference
V1R13	SC34-4824-09	ISPF – Dialog Tag Language Guide and Reference
V1R13	SC34-4820-10	ISPF – Edit and Edit Macros

Manuals – ISPF

Version	Manual	Link
V1R13	SC34-4815-10	ISPF – Messages and Codes
V1R13	GC34-4814-09	ISPF – Planning and Customizing
V1R13	SC34-4816-10	ISPF – Reference Summary
V1R13	SC34-4817-11	ISPF – SCLM Guide and Reference
V1R13	SC34-4819-10	ISPF – Services Guide
V1R13	SC34-4822-10	ISPF – User's Guide – Vol 1
V1R13	SC34-4823-10	ISPF – User's Guide – Vol 2

Manuals – REXX

Version	Manual	Link
V2R01	SA32-0972-00	REXX – Reference
V2R01	SA32-0982-00	REXX – User's Guide
V2R01	SA23-2283-00	REXX – UNIX System Services
V2R01	SH19-8160-06	REXX – Compiler/Library – User's Guide and Reference
V2R01	SH19-8179-03	REXX – Compiler/Library – Diagnosis Guide

Manuals – TSO

Version	Manual	Link
V2R01	SA32-0977-00	TSO – Administration
V2R01	SA32-0975-00	TSO – Command Reference
V2R01	SA32-0976-00	TSO – Customization
V2R01	SA32-0979-00	TSO – General Information
V2R01	SA32-0970-00	TSO – Messages
V2R01	SA32-0984-00	TSO – Primer
V2R01	SA32-0981-00	TSO – Programming Guide
V2R01	SA32-0973-00	TSO – Programming Services
V2R01	GA32-0983-00	TSO – System Diagnosis: Data Areas
V2R01	SA32-0974-00	TSO – System Programming Command Reference
V2R01	SA32-0971-00	TSO – User's Guide

Manuals – SDSF

Version	Manual	Link
Website	N/A	SDSF Downloads
Summary Card	N/A	SDSF – Summary Card
Red Manual	SG24-7419-00	Implementing REXX Support in SDSF
V2R01	SA23-2274-00	SDSF – Operation and Customization
V1R13	SA22-7670-15	SDSF – Operations and Customization
V1R12	SA22-7670-14	SDSF – Operations and Customization
V1R11	SA22-7670-12	SDSF – Operations and Customization
V1R10	SA22-7670-11	SDSF – Operations and Customization
V1R09	SA22-7670-10	SDSF – Operations and Customization
V1R08	SA22-7670-09	SDSF – Operations and Customization
V1R07	SA22-7670-08	SDSF – Operations and Customization
V1R06 & V1R05	SA22-7670-07	SDSF – Operations and Customization
V1R04	SA22-7670-04	SDSF – Operations and Customization
V1R03	SA22-7670-02	SDSF – Operations and Customization
V1R02	SA22-7670-01	SDSF – Operations and Customization

SHARE – ISPF

Session	Author	Link
16665	Bruce Koss	ISPF – Workstation Agent (WSA)
16630	Sam Reynolds	ISPF Hidden Treasures and New Features
16628	Liam Doherty	ISPF Editor – Beyond The Basics Hands-on Lab
16605	Bruce Koss	REXX and ISPF – Troubleshooting
14767	Peter Van Dyke	ISPF Behind the Scenes
12718	Thomas Conley	Configuring ISPF for Fun and Profit
11708	Peter Van Dyke	ISPF Panels Advanced
8676	Peter Van Dyke	ISPF User's Boot Camp
8363	Rod Feak	When Worlds Collide – ISPF Tools for the Windows World
2277	Liam Doherty	ISPF Panels – Dialog Tag Language (DTL)

SHARE – SDSF

Session	Author	Link
16604	Bruce Koss	SDSF – Beyond the Basics
15042	Chip Wood	SDSF for New Users Hands-on Lab
14920	Chip Wood	SDSF Product Update for z/OS 2.1
14768	Chip Wood	SDSF Hidden Treasures
11701	Chip Wood	Accessing SDSF data using REXX and JAVA
10644	Chip Wood	What’s new in SDSF z/OS V1.13?
8919	Chip Wood	SDSF: What’s New in z/OS 1.12?
2677	Dave Danner	z/OS 1.9 JES2 and SDSF: User Experiences With Migration and Using the New Features
2672	Tom Wasik	SDSF for Intermediate Users
2671	Bill Keller	SDSF Changes for z/OS V1R7
2664	Chip Wood	SDSF REXX
2663	Chip Wood	What’s new in SDSF z/OS V1.10?
2655	Chip Wood	SDSF z/OS V1.9 Preview
2343	Chip Wood	SDSF for New Users
2341	Rod Freak	SDSF Rexx in a Production Environment – User Experience

SHARE – REXX

Session	Author	Link
16722	Liam Doherty	Using REXX for IBM Mainframe Application Development
16691	Tracy Dean	REXX Language Coding Techniques
16616	Bruce Schaefev	RACF and REXX - A Marriage Made in Heaven!
16605	Bruce Koss	REXX and ISPF – Troubleshooting
16474	John Franciscovich	Introduction to REXX Workshop
15229	Barry Lichtenstein	An Introduction to Using REXX with Language Environment
14019	Brian Marshall	REXX programming for the z/OS programmer
11751	Thomas Conley	Rexx Power Tools – The PARSE Command
11701	Chip Wood	Accessing SDSF Data from Rexx and Java
10425	Eric Rosenfeld	Analyzing Your RACF Database Using REXX
7485	Thomas Conley	Learn to Program in Rexx – Hands-on Lab
5490	Mike Onghena	RACF's R_admin Interface: Now Served with REXX!
2344	Chip Wood	SDSF Rexx API Usage Tutorial
2341	Rod Feak	SDSF Rexx in a Production Environment – User Experience
1579	David Ashley	Using Rexx Complier Hands-on Lab

SHARE – REXX

Session	Author	Link
1569	Ahilan Rajadeva	Rexx and z/OS UNIX Services
2344	Chip Wood	SDSF Rexx API Usage Tutorial
2341	Rod Feak	SDSF Rexx in a Production Environment – User Experience

Websites (📎)

Link
ABCs of Systems Programming
Andy’s MVS Page
Cardett Associates – Database Query Tools
CBT Tape
DFSORT – Documentation and Code
GSF Software – Downloads
ISPF – Manuals and Tools
ISPF – Request For Enhancement (RFE)
Kenneth Tomiak – Stuff
Ken’s Universe
Leonard Woren – Free Stuff
Lionel B. Dyck – Freeware
Mark’s MVS Utilities
Michael Joseph Clearly – Freeware
Parallel Sysplex – Tools and Wizards

Websites (📎)

Link
PDS2PDS – PDS Comparison
Planet MVS
Redbooks – PDFs and Code
Review – Browser and Editor
REXX – Language Association
Schlabb – REXX, TCPIP, etc.
SDSF – Home Page
SDSF – MODEL Examples
SDSF – REXX Examples
SHARE – Conference Proceedings
Simo Time Enterprises – Programming Code Examples
TASID – Downloads
VM – Download Packages
Watson & Walker – Tuning Letter
WLM – Downloads

Websites (📎)

Link
ZOS – V1R13 – PDF Manuals
ZOS – V2R01 – PDF Manuals

Any questions or comments please contact

Bruce Koss

Wells Fargo Mainframe Operating Systems Support

Bruce.Koss@WellsFargo.com

(704) 600-8416