An Experienced ISPF User Shares his Secrets

Bruce Koss
Wells Fargo

August 11, 2012
11559
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 datasets to allow testing

• Other Items
  • Links are provided to IBM manuals and other documents to assist in learning these features in more detail
  • May need to adjust Adobe Reader zoom value to view text
Agenda

• **ISPF Configuration**
  • Settings

• **Stacking Commands**
  • Overview
  • 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
ISPF Configuration
Introduction

• Provide an overview of several ISPF configuration variables to
  • Decrease response time
  • For new users, eliminate time configuring environment
  • Provide new features/enhancements

• ISPF Configuration Dialog
  • Suggest executing this after installing a new z/OS operating system
  • Execution
    ➔ TSO ISPCCONF

• Links
  • SHARE, Session 10948, March 14, 2012, Tom Conley
    • Configuring ISPF for Fun and Profit
  • Chapter 2, The ISPF Configuration Table
    • ISPF – V1R13 – Planning and Customization
<table>
<thead>
<tr>
<th>Variable</th>
<th>Assigned</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>APPLID_FOR_SITE_COMMAND_TABLE</td>
<td>WFS</td>
<td>Allows for system/site wide command tables to be defined/used (WFSCMDS) - WFS stands for Wells Fargo Site commands - Default is NONE</td>
</tr>
<tr>
<td>APPLID_FOR_USER_COMMAND_TABLE</td>
<td>WFU, WFU2, WFU3</td>
<td>Allows users to define up to three different user command tables (WFUCMDS, WFU2CMDS and WFU3CMDS) - WFU stands for Wells Fargo User commands - Default is NONE</td>
</tr>
<tr>
<td>BLOCK_SIZE_FOR_TEMPORARY_CNTL_DATA_SETS</td>
<td>0</td>
<td>Set block size to zero to obtain/use half track blocking - Default is 800, 3146 and 2560</td>
</tr>
<tr>
<td>BLOCK_SIZE_FOR_TEMPORARY_LIST_DATA_SETS</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>BLOCK_SIZE_FOR_TEMPORARY_WORK_DATA_SETS</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>COMMAND_LINE_PLACEMENT</td>
<td>ASIS</td>
<td>Place command line on top - Default is BOTTOM</td>
</tr>
<tr>
<td>DEFAULT_LIBDEF_PROCESSING_OPTION</td>
<td>STACK</td>
<td>When LIBDEFing datasets, the default will be STACK - Prevents issues when stacking software on top one another - Default is UNCOND</td>
</tr>
<tr>
<td>DEFAULT_SYSTEM_NAME</td>
<td>ON</td>
<td>Displays Ipar name in upper left hand corner of all panels - Default is OFF</td>
</tr>
</tbody>
</table>
## ISPCCONF

<table>
<thead>
<tr>
<th>Variable</th>
<th>Assigned</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>DISALLOW_WILDCARDS_IN_HLQ</td>
<td>YES</td>
<td>Prevents long enqueues on the master catalog</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Default is NO</td>
</tr>
<tr>
<td>DLIST_RMENABLED</td>
<td>YES</td>
<td>Allows executing a REXX program to obtain CA1 information when typing</td>
</tr>
<tr>
<td>DLIST_RMCOMMAND</td>
<td>YES %CA1DS34</td>
<td>- &quot;I&quot; invokes CA1, selects option “1 - INQ/UPD” and inserts VOLSER to</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- &quot;S&quot; invokes CA1’s Browse Module (TSOTIQ) and uses dataset name to</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- &quot;M&quot; invokes CA1’s Tape Inquiry Program (TMSBINQ) and uses VOLSER to</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Can use CA1 REXX program TSOTIQ</td>
</tr>
<tr>
<td>DISPLAY_EMPTY_MEMBER_LIST</td>
<td>YES</td>
<td>Capability to edit an empty PDS</td>
</tr>
<tr>
<td>DISPLAY_EMPTY_MEMBER_LIST_PATTERN</td>
<td>YES</td>
<td>- Default is NO and %EDGRPD34</td>
</tr>
<tr>
<td>DISPLAY_SEQUENCE_NUMBERS</td>
<td>OFF</td>
<td>- Default is ON</td>
</tr>
<tr>
<td>EDIT_RECOVERY_BLOCK_SIZE</td>
<td>27998</td>
<td>Allow for large amount of changes and updates</td>
</tr>
<tr>
<td>EDIT_RECOVERY_PRIMARY_QUANTITY</td>
<td>800</td>
<td>- Default is 13680, 40 and 200</td>
</tr>
<tr>
<td>EDIT_RECOVERY_SECONDARY_QUANTITY</td>
<td>800</td>
<td></td>
</tr>
<tr>
<td>Variable</td>
<td>Assigned</td>
<td>Notes</td>
</tr>
<tr>
<td>----------------------------------------------------</td>
<td>----------</td>
<td>------------------------------------------------------------------------</td>
</tr>
<tr>
<td>HILITE_DOLOGIC</td>
<td>ON</td>
<td>Distinguishes lines of code and assists in troubleshooting</td>
</tr>
<tr>
<td>HILITE_IFLOGIC</td>
<td>ON</td>
<td>- Default is OFF, OFF and OFF</td>
</tr>
<tr>
<td>HILITE_PAREN</td>
<td>ON</td>
<td></td>
</tr>
<tr>
<td>ISPCTL0_BLOCK_SIZE</td>
<td>0</td>
<td>Set block size to zero to obtain/use half track blocking</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Default is 800</td>
</tr>
<tr>
<td>ISPF_TEMPORARY_DATA_SET_QUALIFIER</td>
<td>&amp;SYSNAME.</td>
<td>Inserts the lpar name in the second level qualifier of the ISPF</td>
</tr>
<tr>
<td></td>
<td></td>
<td>temporary dataset names</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Allows logging on to multiple, concurrent TSOs within a sysplex</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- If lpar name begins with a number, suggest placing the character ‘S’</td>
</tr>
<tr>
<td></td>
<td></td>
<td>in front of &amp;SYSNAME</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Default is NONE</td>
</tr>
<tr>
<td>LIST_DATA_SET_RECORDS_PER_BLOCK</td>
<td>231</td>
<td>Set list data set records per block to 231 for a 3390</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- LRECL 121 * 231 = 27951, half-track for a 3390</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Default is 26</td>
</tr>
<tr>
<td>LIST_SYSOUT_CLASS</td>
<td>*</td>
<td>- Default is NONE</td>
</tr>
<tr>
<td>LOCAL_LIST_SYSOUT_CLASS</td>
<td>*</td>
<td>- Default is NONE</td>
</tr>
<tr>
<td>LOG_DATA_SET_BLOCK_SIZE</td>
<td>0</td>
<td>Set block size to zero to obtain/use half track blocking</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Default is 129</td>
</tr>
</tbody>
</table>
## ISPCCONF

<table>
<thead>
<tr>
<th>Variable</th>
<th>Assigned</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOG_DATA_SET_DISPOSITION</td>
<td>D</td>
<td>Delete log data set</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Default is NONE</td>
</tr>
<tr>
<td>LOG_LIST_JOB_CARD1</td>
<td>Job Card</td>
<td>Defined a standard JOB card format</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Software can obtain and use it by coding “VGET (ZLLGJOB1 ZLLGJOB2 ZLLGJOB3 ZLLGJOB4)”</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Default is NONE</td>
</tr>
<tr>
<td>LOG_PAGES_PRIMARY_QUANTITY</td>
<td>0</td>
<td>Not needed/used, decreases response time</td>
</tr>
<tr>
<td>LOG_PAGES_SECONDARY_QUANTITY</td>
<td>0</td>
<td>- Default is 10 and 10</td>
</tr>
<tr>
<td>MAXIMUM_NUMBER_OF_SPLIT_Screens</td>
<td>32</td>
<td>Capability to have 32 split screens</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Set PF2=START, PF9=SWAP NEXT and</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PF21=SWAP PREV, then activate SWAPBAR</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Default is 8</td>
</tr>
<tr>
<td>NUMBER_OF_PFKEYS</td>
<td>12</td>
<td>Typing in KEYS will display “Number of Keys = 24”</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Default is 12</td>
</tr>
<tr>
<td>OUTLIST_BLOCK_SIZE</td>
<td>0</td>
<td>Set Outlist Utility block size to zero to</td>
</tr>
<tr>
<td></td>
<td></td>
<td>obtain half track blocking for OUTLIST</td>
</tr>
<tr>
<td></td>
<td></td>
<td>datasets</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Default is 13566</td>
</tr>
<tr>
<td>PROFILE_SHARING</td>
<td>YES</td>
<td>Allows logging on to multiple, concurrent</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TSOs within a sysplex</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Required to run multiple, concurrent z/OSMF</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ISPF sessions</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Default is NO</td>
</tr>
<tr>
<td>Variable</td>
<td>Assigned</td>
<td>Notes</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>----------</td>
<td>--------------------------------------------</td>
</tr>
<tr>
<td>SCROLL_DEFAULT</td>
<td>CSR</td>
<td>Set scroll default to cursor</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Default is PAGE</td>
</tr>
<tr>
<td>SHOW_PFKEYS</td>
<td>OFF</td>
<td>Provides for more screen real-estate</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Default is ON</td>
</tr>
<tr>
<td>SITE_WIDE_INITIAL_MACRO</td>
<td>RESET</td>
<td>Prevents unnecessary messages from appearing when editing a dataset or member</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Default is NONE</td>
</tr>
<tr>
<td>STANDARD_NUMBERS</td>
<td>OFF</td>
<td>- Default is ON</td>
</tr>
<tr>
<td>STATS_EXT_ENABLED</td>
<td>YES</td>
<td>Extended statistics are kept for PDS members</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Default is NO</td>
</tr>
<tr>
<td>TAB_TO_ACTION_BARS</td>
<td>NO</td>
<td>Pressing the HOME key will position cursor on primary command line</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Default is YES</td>
</tr>
<tr>
<td>UNDO_STORAGE_SIZE</td>
<td>1024</td>
<td>Allocate 1024K of memory to edit UNDO command to be used for keeping history of edit changes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Enables SETUNDO command</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Default is 0</td>
</tr>
</tbody>
</table>
### ISPCCONF

<table>
<thead>
<tr>
<th>Variable</th>
<th>Assigned</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>USE_KEYLISTS</td>
<td>NO</td>
<td>Makes PF keys standard throughout ISPF</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Assigns PF4=RETURN, PF5=RFIND, PF6=RCHANGE, PF7=UP, PF8=DOWN,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PF9=SWAP, PF10=LEFT, PF11=RIGHT and PF12=RETRIEVE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Default is YES</td>
</tr>
<tr>
<td>VSAM_BROWSE_ENABLED</td>
<td>YES</td>
<td>Allows invoking a REXX program when browsing, editing or viewing a</td>
</tr>
<tr>
<td>VSAM_BROWSE_COMMAND</td>
<td>%ISPF34B</td>
<td>VSAM file</td>
</tr>
<tr>
<td>VSAM_EDIT_ENABLED</td>
<td>YES</td>
<td>- ISPF34B/E is an in-house written REXX program to invoke File-Aid/</td>
</tr>
<tr>
<td>VSAM_EDIT_COMMAND</td>
<td>%ISPF34E</td>
<td>MVS (F1/2) or CBT183 (BR)</td>
</tr>
<tr>
<td>VSAM_VIEW_ENABLED</td>
<td>YES</td>
<td>- Can use File-Aid/MVS REXX programs F1 (Browse) and F2 (Edit)</td>
</tr>
<tr>
<td>VSAM_VIEW_COMMAND</td>
<td>%ISPF34B</td>
<td>- Can install and invoke Ditto, File Manager or Review from CBT tape</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Default is NO and FMNMAIN DSB /</td>
</tr>
<tr>
<td>WSA_DOWNLOAD_DATA_SET</td>
<td>'SYS1.SISPUI'</td>
<td>Dataset name inserted into ISPF 3.7.1 screen</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Default is 'ISP.SISPUI'</td>
</tr>
</tbody>
</table>
Stacking Commands
Introduction

• Stacking commands provides the capability to invoke any application from anywhere within ISPF and not have to be located at the user or main menu

• 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 6)
Syntax

- All menu options are stacking commands

- IBM Primary Menu
  - Prefix the character “Z” in front of 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
Fast Path

• To fast path directly into a particular menu entry
  • Use the ISPF delimiter character
  • Default is the semi-colon (;)

• Examples of fast pathing into a software product
  • SDSF, Status of jobs and set owner to R2221
    ➔ ZSD;ST;OWNER R2221   -or-   ➔ ZSD ST;OWNER R2221
  • OPSMVS, State Manager, Resource Status, option 4.11.2
    ➔ OPS;4;11;2   -or-   ➔ OPS 4;11;2   -or-   ➔ OPSMVS 4;11;2
  • Endevor option 2.1
    ➔ ENDE;2;1   -or-   ➔ ENDE 2;1   -or-   ➔ ENDEVOR 2;1
  • SPUFI for DB2 subsystem DI32, option 1.1
    ➔ DB2U DI32;1;1
How does it work?

- A user who is 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, they can stack ISPF 3.4 on top, 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

Z34

Menu  RefList  RefMode  Utilities  Help
-------------------------------------------------------------------------------
Data Set List Utility
Option ===>

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

DATA SET LIST UTILITY

R2221    T0084269  DSID     3 LINE 0       COLUMNS 02- 81
COMMAND INPUT ===> Z34                                         SCROLL ===> PAGE

********************************* TOP OF DATA **********************************
1 //R2221   JOB 'TSOPRD',REGION=8192K
2 //PANTSOB EXEC PANTSOB
3 XXPANTSOB EXEC PGM=IKJEFT01,
   XX         DYNAMNBR=200,
   XX         REGION=3072K,
   XX         TIME=480,
   XX         PARM='%XAPAN'
4 XXSTEPLIB DD DISP=SHR,DSN=PRDLL.QU.VPS.LL2.LINKLIB
   XX*       **** TSO PROCEDURE USED BY - S & P - *****
   XX*       ***
   XX**      ***********************************************
   XX*       ISPCTL1 DD SPACE=(CYL,(1,1)),UNIT=TTEMP,
   XX**      DCB=(BLKSIZE=800,RECL=80,RECFM=FB)
   XX*       ISPCTL2 DD SPACE=(CYL,(1,1)),UNIT=TTEMP,
   XX**      DCB=(BLKSIZE=800,RECL=80,RECFM=FB)
5 XXISPLST1 DD SPACE=(CYL,(1,1)),UNIT=TTEMP,
   XX      DCB=(BLKSIZE=1210,RECL=121,RECFM=FBA)
STACK – Utility

- An in-house utility called STACK or STK was developed to display all available stacking commands.
- Users can type in STACK or STK on any primary command line.
- Displays all available commands defined in site (WFSCMDS) or user (WFU[2|3]CMDS) commands table.
- 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 displays all stacking commands

```
-----------------------------( Stacking Commands )--------- Row 1 to 11 of 320
Command ===>                                                   Scroll ==> CSR

Command:  F xxx - Find text    L xxx - Locate text     xxx   - Exec command
          O xxx - Only text    R     - Reset display   S xxx - Sort column

Line:     S - Invoke utility   B - Browse overview

Trunc:    Number of characters needed to invoke Stacking Command.
          If field is zero, must type in full command.

  S  Command   Trunc                        Description
    --------  -----  ------------------------------------------------------
    _  AAVIEW      3    AbendAid Hotkey
    _  ABENDAID    3    Dump Analysis Tool
    _  ACR34       0    Automated Controls and Reconciliation
    _  ACR35       0    Automated Controls and Reconciliation
    _  AMBLIST     3    Load Module Map and Cross Reference Listing
    _  APF			     4    APF Datasets
    _  APPC        0    Advanced Program To Program Communication
    _  ASID        0    Address Space Identifier (ASID) Usage Summary
    _  AUTOCMDS    5    OPSMVS - MVS Automation Cmds
    _  AXCIS       0    Administration
    _  BDT         0    Bulk Data Transfer
```
Pros & Cons

• Pros
  • Enhances productivity in ISPF
  • Allows invoking multiple applications on one screen
    • And on top of one another
  • "Stacking" capability is limited only by amount of TSO virtual storage
  • Reduces amount of time needed to invoke the facility

• Cons
  • When two or more applications are stacked on the same split screen
    • Can only access/use the active one
      • The one on the top of the stack
    • Can not select an application in the middle of the stack
  • Typing in “=??” will remove the entire stack
    • Where ?? is an ISPF menu entry
    • Examples: =3.4, =D.ABENDAID, =X, etc.
Implementation – ISPCCONF

• ISPF Configuration Utility
  • Type in
    ➔ TSO ISPCCONF
  • Define/set
    • APPLID_FOR_SITE_COMMAND_TABLE = WFS
    • APPLID_FOR_USER_COMMAND_TABLE = WFU, WFU2, WFU3

• 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

• Next few slides will demonstrate how to create a site command table (WFSCMDS) using ISPF 3.9, Commands

• This same process can be used to create user commands (WFUCMDS, WFU2CMDS and WFU3CMDS)

• Prerequisite
  • Must allocate a PDS dataset to ISPTABL DD card
    • This is where the site command table will be saved (WFSCMDS)
    • Preferably a dataset concatenated to ISPTLIB DD card
      • Allows easy testing and validating of site commands
    • Dataset attributes: RECFM=FB,LRECL=80
Invoke ISPF 3.9, insert site command table name (WFS)

Command Table Utility

Command ===> 

Specifications
Application ID . . WFS
Enter "/" to select option
Show description field

Command table search order
Application table . : ISP
User table 1 . . . .
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

---

<table>
<thead>
<tr>
<th>Command</th>
<th>Update WFSCMDS</th>
<th>Scroll</th>
</tr>
</thead>
<tbody>
<tr>
<td>==&gt;</td>
<td>Row 1 to 15 of 17</td>
<td>CSR</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Verb</th>
<th>T</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>E</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Implementation – ISPF 3.9

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

<table>
<thead>
<tr>
<th>File</th>
<th>Menu</th>
<th>Utilities</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Command Table Utility</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Extended Command Entry</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Command ==&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Make changes to the command and select Update to update the entry or Cancel to ignore the changes.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Verb . . . Z34</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Trunc . . . 0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Action . . SELECT PGM(ISRUDL) PARM(ISRUDLP) NEWAPPL(ISR)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ISPF 3.4 - Utilities - DLIST</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Enter / to select option</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>_ Allow mixed-case in Action field</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Update</td>
<td>Cancel</td>
</tr>
</tbody>
</table>
Implementation – ISPF 3.9

• VERB
  • Stacking command
  • Must follow IBM naming convention
    • 1st char = A-Z $#@ and 2nd-8th char = A-Z 0-9 $#@
  • Must be 2 to 8 characters in length

• TRUNC
  • Minimum number of characters the user must enter
  • If zero or equal to verb length, user must enter entire command

• 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 dataset accessible to everyone

• To update the active site command table (WFSCMDS)
  • Need to 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 character Z and a number (Z1, Z2, etc.)

• Ensure software LIBDEFing datasets use sub parameter STACK
  • Change default LIBDEF processing option to STACK
  • Do not use LIBDEF sub parameter UNCOND
Suggestions

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

• Use ISPF 3.9, Create/change application command table
  • Specifically designed for updating site and user command tables
  • Where ISPF 3.16, ISPF table utility, is a general purpose utility
    • Used to update a wide variety/range of tables
Notes

• Command table search order

<table>
<thead>
<tr>
<th>Order</th>
<th>Command Table</th>
<th>Member</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>User command table 1</td>
<td>WFUCMDS</td>
</tr>
<tr>
<td>2</td>
<td>User command table 2</td>
<td>WFU2CMDS</td>
</tr>
<tr>
<td>3</td>
<td>User command table 3</td>
<td>WFU3CMDS</td>
</tr>
<tr>
<td>4</td>
<td>Site command table 1</td>
<td>WFSCMDS</td>
</tr>
<tr>
<td>5</td>
<td>Site command table 2</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Site command table 3</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>System table</td>
<td>ISPCMDS</td>
</tr>
</tbody>
</table>
Some Independent Vendor Products (IVPs) provide the ability to implement stacking commands

**CA – PDSMAN**

- Initialization Parameter
  - Use $EZYCMD to define site commands
- Pros
  - Can refresh site command table by executing below MVS command
    \[ F \text{ PDSMAN,REFRESH} \]
- Cons
  - To test/validate, must use a different Lpar from where the general TSO population is logged onto
  - 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 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 – V1R13 – 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
  • Can only be used in EDIT and VIEW mode

• Links
  • SHARE, Session 11567, Aug 2012, Liam Doherty and Peter Van Dyke
    • 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 – V1R13 – Edit and Edit Macros
Enhanced

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

- Next few slides display
  - Changes highlighted in the IBM ISREMCLS panel
  - ISRREMJCL 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
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

```assembler
/* PANEL KEYLIST(ISRSNAB,ISR) */
/* /* 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 */
/* */
/* ATTR DEFAULT() FORMAT(MIX)           /* ISREMCLS */
/* 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
.. .CLIST   .- ISPF services in CLIST commands
.. .COBOL   .- ISPF services in COBOL programs
.. .EXEC    .- ISPF services in EXEC commands
.. .FORTRAN .- ISPF services in FORTRAN programs
.. .MSGSS   .- Message formats
.. .PANELS  .- Panel formats and statements
.. .PLI     .- ISPF services in PLI programs
.. .SHELLS  .- File tailoring control statements
.. .FASCAL  .- ISPF services in FASCAL programs
.. .REXX    .- SDSF function and ISPF services in REXX execs
.. .DTL     .- ISPF Dialog Tag Language formats and statements
.. .C      .- ISPF services in C/370 programs
.. .SCIM    .- SCIM Project Definition Macros
.. .ARCHDEF - SCIM Architecture Definition templates
.. .JCL     .- JCL for selected utility programs (USER-ADDED)

)INIT
.. .EVARS = '(ZCMD)' */
"WINTTL = '&#39;'
.. .HELP = ISR2P100
.. .MSG = ' ' /* BLANK MSG INITIALLY */
.. .ISREMCLASS = ' ' /* AND THE CLASS HOLDER VARIABLE */
.. .CURSOR = 'ZCMD'
```

```assembler
)PROC
IF ("ISRMCLASS = 'Y'\n/* ARE WE ONLY OBTAINING A CLASS ? */
&ISRMCLASS = TRANS(TRUNC("ISREMCL",".'\n1,CLIST,2,COB3,EXEC 4,FORT
5,MSGS,6,PANELS,6
7,PLI,7A,FLOP7B,PLS,7C,SKELS,8,PROCS,8A,ISPSLIB,8B
FASCAL,9,REXX,10,DLL,11,C,12,CPP1,13,SCIM,13,ARCHDEF,14,JCL,15,' ',' ',' '*))
X " "
&ZCMD = TRUNC("ISREMCL",".'
&TMPTRAIL = .TRAIL
/* Save any remaining */
IF ("ISRMCLASS = 'Y'\n/* ARE WE ONLY OBTAINING A CLASS ? */
&ISRMCLASS = ' ' /* ISREMCLAS = ' ' */
&ZCMD = TRUNC("ISREMCL",3)
&ISRMCLASS = ' ' /* ISREMCLAS = ' ' */
&TMPTRAIL = .TRAIL
*/
```

```assembler
IF ("ISRMCLASS = 'Y'\n/* ARE WE ONLY OBTAINING A CLASS ? */
&ISRMCLASS = ' ' /* ISREMCLAS = ' ' */
&ZCMD = TRUNC("ISREMCL",3)
&ISRMCLASS = ' ' /* ISREMCLAS = ' ' */
&TMPTRAIL = .TRAIL
*/
### JCL Utility Models

**Option ====>**

Enter an option or move the cursor on JCL utility and press ENTER. Enter END command to cancel MODEL command.

<table>
<thead>
<tr>
<th>IBM</th>
<th>OEM</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>I1 AMATERSE - Terse a file</td>
<td>O1 AFCLOGR1 - Extract OPERLOG records</td>
<td>IBM</td>
</tr>
<tr>
<td>I2 AMBLIST - Print linkage attributes</td>
<td>O2 CAL2X2WB - Execute CA7 Commands</td>
<td>OEM</td>
</tr>
<tr>
<td>I3 BPXBATCH - Run USS shell scripts</td>
<td>O3 COB2EZT - Conv. COBOL to Eztrpl</td>
<td>IBM</td>
</tr>
<tr>
<td>I4 CPPUPDTE - Search/update</td>
<td>O4 COMPAREB - Compare two files</td>
<td>OEM</td>
</tr>
<tr>
<td>I5 DFDSS - Backup/restore files</td>
<td>O5 CONSOLE - Execute MVS commands</td>
<td>IBM</td>
</tr>
<tr>
<td>I6 DFSORT - Sort/merge files</td>
<td>O6 COPYCAT - Copy a tape</td>
<td>OEM</td>
</tr>
<tr>
<td>I7 ICETOOL - Sort/merge files</td>
<td>O7 DAF - Dataset Audit Facility</td>
<td>IBM</td>
</tr>
<tr>
<td>I8 ICKDSF - Init disk pack</td>
<td>O8 DJCTEST - Abend job or set RC</td>
<td>OEM</td>
</tr>
<tr>
<td>I9 IDCAMS - Def/del/alter VSAM/GDGs</td>
<td>O9 EXAMBATX - CA Examine Compare</td>
<td>IBM</td>
</tr>
<tr>
<td>I10 IEBCOPY - Copy/del/move PDS mbr</td>
<td>O10 FCSWAIT - Wait a duration of time</td>
<td>OEM</td>
</tr>
<tr>
<td>I11 IEBGENER - Copy sequential files</td>
<td>O11 FILEAID - Copy/update/print files</td>
<td>IBM</td>
</tr>
<tr>
<td>I12 IEBPTPCH - Print and punch</td>
<td>O12 FTP - Transmit a file</td>
<td>OEM</td>
</tr>
<tr>
<td>I13 IEBUPDTE - Create/alter PDS mbr</td>
<td>O13 FTPSEC - Transmit a file secured</td>
<td>IBM</td>
</tr>
<tr>
<td>I14 IEHLIST - List PDS directory</td>
<td>O14 GENMAIL - Email using IEBGENER</td>
<td>OEM</td>
</tr>
<tr>
<td>I15 IEHPROGM - Rename/del/cat file</td>
<td>O15 LISTF - List DASD volumes</td>
<td>IBM</td>
</tr>
<tr>
<td>I16 IEFBR14 - Create/delete file</td>
<td>O16 MZCCMPB1 - Event Action Compare</td>
<td>OEM</td>
</tr>
</tbody>
</table>
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: - +

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>I16</td>
<td>IEFBR14 - Create/delete file</td>
</tr>
<tr>
<td>I17</td>
<td>IFASMFDP - Dump SMF records</td>
</tr>
<tr>
<td>I18</td>
<td>IFCEREP1 - Dump LOGREC</td>
</tr>
<tr>
<td>I19</td>
<td>IKJEFT01 - TSO batch</td>
</tr>
<tr>
<td>I20</td>
<td>IMASPZAP - Zap/dump a module</td>
</tr>
<tr>
<td>I21</td>
<td>IRXJCL - Execute REXX program</td>
</tr>
<tr>
<td>I22</td>
<td>IRSUPC - Search/compare</td>
</tr>
<tr>
<td>I23</td>
<td>REXXCL - Compile/link REXX pgm</td>
</tr>
<tr>
<td>I24</td>
<td>TRSMAIN - Terse a file</td>
</tr>
<tr>
<td>I25</td>
<td>SDSF - SDSF batch</td>
</tr>
<tr>
<td>I26</td>
<td>SMP - IBM's ChangeMan</td>
</tr>
<tr>
<td>J1</td>
<td>COMMAND - Command card</td>
</tr>
<tr>
<td>J2</td>
<td>DD - DD card</td>
</tr>
<tr>
<td>J3</td>
<td>JCLLIB - Proclib card</td>
</tr>
<tr>
<td>J4</td>
<td>JOB - Job card</td>
</tr>
</tbody>
</table>

016 MZCCMPB1 - Event Action Compare
017 PDS - PDS Utility
018 PDSEASY - Batch Ezyedit
019 PDSM13 - Library map/analysis
020 PDSM18 - String scan/replace
021 PDSM25 - Member size analysis
022 PDSM32 - Delete generation mbrs
023 PDSM33 - Compare/update files
024 PDSM36 - Compare files
025 PDSMERGE - Merge PDS datasets
026 PKZIP - Zip/unzip a file
027 QWIKREF1 - MVS/Quick Reference
028 SASSBSTR - Execute CA7 commands
029 SAVRS - Obtain syslog/reports
030 SDEBYSDE - Side by side comparison
031 SHOWMVS - IPL information
032 SYSWAIT - Wait a duration of time
## JCL Utility Models

Enter an option or move the cursor on JCL utility and press ENTER. Enter END command to cancel MODEL command.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
<th>More:</th>
</tr>
</thead>
<tbody>
<tr>
<td>J1</td>
<td>COMMAND</td>
<td>O29 SAVRS</td>
</tr>
<tr>
<td>J2</td>
<td>DD</td>
<td>O30 SDEBYSDE</td>
</tr>
<tr>
<td>J3</td>
<td>JCLLIB</td>
<td>O31 SHOWMVS</td>
</tr>
<tr>
<td>J4</td>
<td>JOB</td>
<td>O32 SYSWAIT</td>
</tr>
<tr>
<td>J5</td>
<td>JOBPARM</td>
<td>O33 TAPEMAP</td>
</tr>
<tr>
<td>J6</td>
<td>IF</td>
<td>O34 TASID</td>
</tr>
<tr>
<td>J7</td>
<td>INCLUDE</td>
<td>O35 TMSBINQ</td>
</tr>
<tr>
<td>J8</td>
<td>OUTPUT</td>
<td>O36 UTGZ</td>
</tr>
<tr>
<td>J9</td>
<td>ROUTE</td>
<td>O37 UTJE</td>
</tr>
<tr>
<td>J10</td>
<td>SET</td>
<td>O38 UTJM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>O39 UTJP</td>
</tr>
<tr>
<td></td>
<td></td>
<td>O40 UTKT</td>
</tr>
<tr>
<td></td>
<td></td>
<td>O41 UTONY</td>
</tr>
<tr>
<td></td>
<td></td>
<td>O42 VSPRIME</td>
</tr>
</tbody>
</table>

### Application

| J1     | APPLJCL                      | Personal/Team JCL |
JCL – ISREMJCL – Panel

COMMAND, J1 DD, J2 JCLLIB, J3 JOB, J4 JOBPARM, J5 SYSAFF, J5
IF, J6 INCLUDE, J7 OUTPUT, J8 ROUTE, J9 SET, J10
APPLJCL, A1

+J1 [COMMAND] - Command card
+J2 [DD] - DD card
+J3 [JCLLIB] - Proclib card
+J4 [JOB] - Job card
+J5 [JOBPARM] - Jobparm card
+J6 [IF] - If card
+J7 [INCLUDE] - Include card
+J8 [OUTPUT] - Output card
+J9 [ROUTE] - Route card
+J10 [SET] - Set card

%Application+

A1 [APPLJCL] - Personal/Team JCL

* * *

IF (&ZCMD ¬= ' ') &ZCMD = '&ZCMD..&ZTMPTRL'

&ZSEL = TRANS (TRUNC (&ZCMD,'.')

I1,'PGM(ISRECMBR) PARM(MAMATERS)'
I2,'PGM(ISRECMBR) PARM(MAMBLIST)'
I3,'PGM(ISRECMBR) PARM(MBPXBATCH)'
I4,'PGM(ISRECMBR) PARM(MCPPUPDT)'
I5,'PGM(ISRECMBR) PARM(MDFDSS)'
I6,'PGM(ISRECMBR) PARM(MDFSORT)'
I7,'PGM(ISRECMBR) PARM(MICETOOL)'
I8,'PGM(ISRECMBR) PARM(MICEXTP)'
JCL – ISREMJCL – Panel

```
/* Save message in prof pool */
VPUT(ZMSG) PROFILE

/* Set message variable */
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)
```

Complete your sessions evaluation online at SHARE.org/AnaheimEval
<table>
<thead>
<tr>
<th>Field</th>
<th>Variable</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ZPS01049</td>
<td>ZCMD</td>
<td>I25</td>
</tr>
<tr>
<td>ZPS01050</td>
<td>ZCMD</td>
<td>O25</td>
</tr>
<tr>
<td>ZPS01051</td>
<td>ZCMD</td>
<td>I26</td>
</tr>
<tr>
<td>ZPS01052</td>
<td>ZCMD</td>
<td>O26</td>
</tr>
<tr>
<td>ZPS01053</td>
<td>ZCMD</td>
<td>I27</td>
</tr>
<tr>
<td>ZPS01054</td>
<td>ZCMD</td>
<td>O28</td>
</tr>
<tr>
<td>ZPS01055</td>
<td>ZCMD</td>
<td>I28</td>
</tr>
<tr>
<td>ZPS01056</td>
<td>ZCMD</td>
<td>O29</td>
</tr>
<tr>
<td>ZPS01057</td>
<td>ZCMD</td>
<td>J2</td>
</tr>
<tr>
<td>ZPS01058</td>
<td>ZCMD</td>
<td>O30</td>
</tr>
<tr>
<td>ZPS01059</td>
<td>ZCMD</td>
<td>J3</td>
</tr>
<tr>
<td>ZPS01060</td>
<td>ZCMD</td>
<td>O31</td>
</tr>
<tr>
<td>ZPS01061</td>
<td>ZCMD</td>
<td>J4</td>
</tr>
<tr>
<td>ZPS01062</td>
<td>ZCMD</td>
<td>O32</td>
</tr>
<tr>
<td>ZPS01063</td>
<td>ZCMD</td>
<td>J5</td>
</tr>
<tr>
<td>ZPS01064</td>
<td>ZCMD</td>
<td>O33</td>
</tr>
<tr>
<td>ZPS01065</td>
<td>ZCMD</td>
<td>J6</td>
</tr>
<tr>
<td>ZPS01066</td>
<td>ZCMD</td>
<td>O34</td>
</tr>
<tr>
<td>ZPS01067</td>
<td>ZCMD</td>
<td>J7</td>
</tr>
<tr>
<td>ZPS01068</td>
<td>ZCMD</td>
<td>O35</td>
</tr>
<tr>
<td>ZPS01069</td>
<td>ZCMD</td>
<td>J8</td>
</tr>
<tr>
<td>ZPS01070</td>
<td>ZCMD</td>
<td>O36</td>
</tr>
<tr>
<td>ZPS01071</td>
<td>ZCMD</td>
<td>J9</td>
</tr>
<tr>
<td>ZPS01072</td>
<td>ZCMD</td>
<td>O37</td>
</tr>
<tr>
<td>ZPS01073</td>
<td>ZCMD</td>
<td>J10</td>
</tr>
<tr>
<td>ZPS01074</td>
<td>ZCMD</td>
<td>O38</td>
</tr>
<tr>
<td>ZPS01075</td>
<td>ZCMD</td>
<td>J39</td>
</tr>
<tr>
<td>ZPS01076</td>
<td>ZCMD</td>
<td>O40</td>
</tr>
<tr>
<td>ZPS01077</td>
<td>ZCMD</td>
<td>O41</td>
</tr>
<tr>
<td>ZPS01078</td>
<td>ZCMD</td>
<td>O42</td>
</tr>
<tr>
<td>ZPS01079</td>
<td>ZCMD</td>
<td>A1</td>
</tr>
</tbody>
</table>

) END
JCL – Sample Skeletons

### MIEHLIST

```plaintext
//*-------------------------------------------------------------------*
//*  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=SYSDDA=######
)N*-------------------------------------------------------------------*
)N*  List a PDS directory attributes (only valid for executables)
)N*-------------------------------------------------------------------*
LISTPDS DSNAME=(******.******.******),FORMAT,   X
          VOL=SYSDDA=######
)N*-------------------------------------------------------------------*
)N*  Retrieve a VTOC listing
)N*-------------------------------------------------------------------*
LISTVTOC FORMAT, VOL=DISK=######
)N*-------------------------------------------------------------------*
)N* Partitioned Dataset (PDS) listing
)N*-------------------------------------------------------------------*
LISTPDS [DSNAME=(dsn1{,dsn2...})]
   [,DUMP,FORMAT]
   [,VOL=device=serial]
)N*-------------------------------------------------------------------*
)N* Volume Table of Contents (VTOC) listing
)N*-------------------------------------------------------------------*
LISTVTOC [DUMP,FORMAT]
   [,INDEXDSN=SYS1.VTOCIX.xxxxxx]
   [,DATE=dddyy]
   [,VOL=device=serial]
   [,DSNAME=(dsn1{,dsn2...})]
)N*-------------------------------------------------------------------*
)N* Catalog Listing
)N*-------------------------------------------------------------------*
LISTCTLG [VOL=device=serial]
   [,NODE=name]
)N
```

### MUTKT

```plaintext
//*-------------------------------------------------------------------*
//*  PURPOSE:  ABEND JOB WITH USER CODE
//*-------------------------------------------------------------------*
//*  MANUAL:   DFSMSDFP UTILITIES
//*-------------------------------------------------------------------*
//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*-------------------------------------------------------------------*
```
## REXX Models

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

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SDSF REXX</td>
<td>ISPF Workstation</td>
</tr>
<tr>
<td>S0 SDSFREXX</td>
<td>ISPF Library Access</td>
</tr>
<tr>
<td>X1 FILESTAT</td>
<td>X2 FILEXFER</td>
</tr>
<tr>
<td>X3 WSCON</td>
<td>X4 WSDISCON</td>
</tr>
<tr>
<td>F1 FTOPEN</td>
<td>F2 FTINCL</td>
</tr>
<tr>
<td>F3 FTCLOSE</td>
<td>F4 FTERASE</td>
</tr>
<tr>
<td>D1 DISPLAY</td>
<td>D2 TBDISPL</td>
</tr>
<tr>
<td>D3 SETMSG</td>
<td>D4 PQUERY</td>
</tr>
<tr>
<td>D5 ADDPOP</td>
<td>D6 REMPOP</td>
</tr>
<tr>
<td>M0 MISC</td>
<td>REXX Commands</td>
</tr>
<tr>
<td>RX REXXCMDS</td>
<td>ISPF Miscellaneous</td>
</tr>
</tbody>
</table>

Complete your sessions evaluation online at SHARE.org/AнаheimEval
SDSF REXX

• IBM System Display and Search Facility Downloads
REXX – Commands – Implementation

• Create ISRRXCMD panel

• Invoke MVS/Quick Reference (QW) and select option
  • “C”, Request information by Category
  • “PROGLANG”, Programming languages syntax
  • “z/OS REXX SYNTAX”
  • A REXX command
  • Type in “CUT ALL”

• Create a new skeleton member
  • Type in QWPASTE
  • Shift the text over three characters to the right
  • Insert “)N“ to make all lines notes

• Add/update an entry to REXX model panel (ISRRXCMD)
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: +

<table>
<thead>
<tr>
<th>Command</th>
<th>Command</th>
<th>Command</th>
<th>Command</th>
<th>Command</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABBREV</td>
<td>ABS</td>
<td>ADDRESS</td>
<td>ARG</td>
<td>BITAND</td>
<td>BITOR</td>
</tr>
<tr>
<td>BITXOR</td>
<td>BOOLEAN</td>
<td>B2X</td>
<td>CALL</td>
<td>CENTER</td>
<td>COMPARE</td>
</tr>
<tr>
<td>CONDITION</td>
<td>COPIES</td>
<td>C2D</td>
<td>C2X</td>
<td>DATATYPE</td>
<td>DATE</td>
</tr>
<tr>
<td>DELSTACK</td>
<td>DELSTR</td>
<td>DELWORD</td>
<td>DIGITS</td>
<td>DO</td>
<td>DROP</td>
</tr>
<tr>
<td>DROPBUF</td>
<td>D2C</td>
<td>D2X</td>
<td>ERRORTEXT</td>
<td>EXECIO</td>
<td>EXECUTIL</td>
</tr>
<tr>
<td>EXIT</td>
<td>EXTERNALS</td>
<td>FIND</td>
<td>FORM</td>
<td>FORMAT</td>
<td>FUZZ</td>
</tr>
<tr>
<td>GETMSG</td>
<td>HE</td>
<td>HI</td>
<td>HT</td>
<td>IF</td>
<td>INDEX</td>
</tr>
<tr>
<td>INSERT</td>
<td>INTERPRET</td>
<td>ITERATE</td>
<td>JUSTIFY</td>
<td>LASTPOS</td>
<td>LEAVE</td>
</tr>
<tr>
<td>LEFT</td>
<td>LENGTH</td>
<td>LINESIZE</td>
<td>LISTDSI</td>
<td>MAKEBUF</td>
<td>MAX</td>
</tr>
<tr>
<td>MIN</td>
<td>MSG</td>
<td>MVSVAR</td>
<td>NEWSTACK</td>
<td>NOP</td>
<td>NUMERIC</td>
</tr>
<tr>
<td>OPTIONS</td>
<td>OUTTRAP</td>
<td>OVERLAY</td>
<td>PARSE</td>
<td>POS</td>
<td>PROCEDURE</td>
</tr>
<tr>
<td>PROMPT</td>
<td>PULL</td>
<td>PUSH</td>
<td>QBUF</td>
<td>QELEM</td>
<td>QSTACK</td>
</tr>
<tr>
<td>QUEUE</td>
<td>QUEUED</td>
<td>RANDOM</td>
<td>RETURN</td>
<td>REVERSE</td>
<td>RIGHT</td>
</tr>
<tr>
<td>RT</td>
<td>SAY</td>
<td>SELECT</td>
<td>SETLANG</td>
<td>SIGN</td>
<td>SIGNAL</td>
</tr>
<tr>
<td>SOURCELINE</td>
<td>SPACE</td>
<td>STORAGE</td>
<td>STRIP</td>
<td>SUBCOM</td>
<td>SUBSTR</td>
</tr>
<tr>
<td>SUBWORD</td>
<td>SYMBOL</td>
<td>SYSCPUS</td>
<td>SYSdsn</td>
<td>SYSVAR</td>
<td>TE</td>
</tr>
<tr>
<td>TIME</td>
<td>TRACE</td>
<td>TRANSLATE</td>
<td>TRUNC</td>
<td>TS</td>
<td>UNTIL</td>
</tr>
<tr>
<td>UPPER</td>
<td>USERID</td>
<td>VALUE</td>
<td>VERIFY</td>
<td>WHILE</td>
<td>WORD</td>
</tr>
</tbody>
</table>
REXX – ISRRRXCMD – Panel

/* This panel is invoked by using the TSO edit command "MODEL", which in turn invokes panel ISREMCLS. Below is the flow-chart: */

MODEL --> ISREMREX --> ISRRXCMD --> M*******

IBM Cmd      IBM panel      User Panel      User Skeleton

FLG  YYMMDD  USERID   DESCRIPTION                                 
---  ------  -------  -------------------------------------------
@A0  100630  R2221    Developed Panel                             

APPL DEFAULT(%+_) FORMAT(MIX) /* ISREMPLI - ENGLISH - 5.0 */
] TYPE(PS)  
0D TYPE(NT)  
05 TYPE(PT)  
06 TYPE(PIN)  
09 TYPE(FT)  
10 TYPE(ET)  
27 TYPE(NEF) CAPS(ON) PADC(USER)
11 TYPE(SAC)  
22 TYPE(WASL) SKIP(ON) GE(ON)
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.

AREA SAREA40
+ [ABBREV ] [ABS ] [ADDRESS ] [ARG ] [BITAND ] [BITOR ]
+ [BITXOR ] [BOOLEAN ] [B2X ] [CALL ] [CENTER ] [COMPARE ]
+ [CONDITION ] [COPIES ] [C2D ] [C2X ]
+ [DELS] [DELS] [DELS] [DIGITS ] [DO ] [DROP ]
+ [DROPBUF ] [DZC ] [DZC ] [ERRORTEXT ] [EXECUT ] [EXECLUS ]
+ [EXIT ] [EXECUT ] [EXIT ] [EXIT ] [EXIT ]
+ [MERGE ] [HE ] [HI ] [HT ] [ IF ] [INDEX ]
+ [INSERT ] [INTERPRET ] [ITERATE ] [JUSTIFY ] [LASTPOS ] [LEAVE ]
+ [LEFT ] [LENGTH ] [LINESIZE ] [LISTDSI ] [MAKEBUF ] [MAX ]
+ [MIN ] [MSG ] [MYSVAR ] [NEWSTACK ] [NOP ] [NUMERIC ]
+ [OPTIONS ] [OUTTRAP ] [OVERLAY ] [PARSE ] [POS ] [PROCEDURE ]
+ [FUNCTION ] [FULL ] [FULL ] [QBUF ] [QELEM ] [QSTACK ]
+ [QUEUE ] [QUEUE ] [RANDOM ] [REVERSE ] [REVERSE ] [RIGHT ]
+ [RS ] [RS ] [SELECT ] [SETLANG ] [SIGN ] [SIGNAL ]
+ [SOURCE ] [SPACE ] [STORAGE ] [STRIP ] [SUBCOM ] [SUBSTR ]
+ [SUBWORD ] [SYMBOL ] [SYSCPUS ] [SYSDSN ] [SYSSVR ] [TS ]
+ [TIME ] [TRACE ] [TRANSLATE ] [TRUNC ] [TS ] [UNTIL ]
+ [UPPER ] [USERID ] [VALUE ] [VERIFY ] [WHILE ] [WORD ]
+ [WORDINDEX ] [WORDLENGTH ] [WORDPOS ] [WORDS ] [X RANGE ] [X2B ]
+ [X2C ] [X2D ]

INIT
.ZVARS = '(ZCMD)
&ZWINTTL = ' 
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(MRXABBRE)' 
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)' 
DATA          ,'PGM(ISRECMBR) PARM(MRXDATAT)' 
DATE          ,'PGM(ISRECMBR) PARM(MRXDATE)' 
DELSTACK      ,'PGM(ISRECMBR) PARM(MRXDELS)' 
DELSTR        ,'PGM(ISRECMBR) PARM(MRXDELST)' 
DELWORD       ,'PGM(ISRECMBR) PARM(MRXDELRO)' 
DIGITS        ,'PGM(ISRECMBR) PARM(MRXDIGIT)' 
DROPOUT       ,'PGM(ISRECMBR) PARM(MRXDO)' 
DROP          ,'PGM(ISRECMBR) PARM(MRXDROP)' 
DROPBUF       ,'PGM(ISRECMBR) PARM(MRXDROPB)' 
DZC           ,'PGM(ISRECMBR) PARM(MRXDZC)' 
DZ2           ,'PGM(ISRECMBR) PARM(MRXDZ2)' 
ERRORTEXT     ,'PGM(ISRECMBR) PARM(MRXERROR)' 
EXECUT        ,'PGM(ISRECMBR) PARM(MRXEXECUT)' 
EXECUT        ,'PGM(ISRECMBR) PARM(MRXEXECU)' 
EXIT          ,'PGM(ISRECMBR) PARM(MRXEXIT)' 
EXTERNALS     ,'PGM(ISRECMBR) PARM(MRXEXTERNAL)' 
FIND          ,'PGM(ISRECMBR) PARM(MRXFIND)' 
FIND          ,'PGM(ISRECMBR) PARM(MRXFIND)' 

Complete your sessions evaluation online at SHARE.org/AnaheimEval
FORM          ,'PGM(ISRECMBR) PARM(MRXFORM )'
FORMAT        ,'PGM(ISRECMBR) PARM(MRXFORMA)'
FUEL          ,'PGM(ISRECMBR) PARM(MRXFUEL)'
GETMSG        ,'PGM(ISRECMBR) PARM(MRXGETM)'
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)'
INSERT        ,'PGM(ISRECMBR) PARM(MRXINSER)'
INTERPRET     ,'PGM(ISRECMBR) PARM(MRXINTER)'
ITERATE       ,'PGM(ISRECMBR) PARM(MRXITERA)'
JUSTIFY       ,'PGM(ISRECMBR) PARM(MRXJUSTI)'
LEAVE         ,'PGM(ISRECMBR) PARM(MRXLEAVE)'
LEFT          ,'PGM(ISRECMBR) PARM(MRXLEFT )'
LENGTH        ,'PGM(ISRECMBR) PARM(MRXLENGTH)'
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(MRXMVSV)'
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(MRXPARS)'
POS           ,'PGM(ISRECMBR) PARM(MRXPOS )'
PROCEDURE     ,'PGM(ISRECMBR) PARM(MRXPROC)'
PROMPT        ,'PGM(ISRECMBR) PARM(MRXPROMP)'
READ          ,'PGM(ISRECMBR) PARM(MRXREAD)'
READONLY      ,'PGM(ISRECMBR) PARM(MRXREADO)'
RECO          ,'PGM(ISRECMBR) PARM(MRXRECO)'
RECOV         ,'PGM(ISRECMBR) PARM(MRXRECOV)'
RECORD        ,'PGM(ISRECMBR) PARM(MRXREC)'
REVERSE       ,'PGM(ISRECMBR) PARM(MRXREVER)'
RIGHT         ,'PGM(ISRECMBR) PARM(MRXRIGHT)'
RT             ,'PGM(ISRECMBR) PARM(MRXRT )'
SAY           ,'PGM(ISRECMBR) PARM(MRXSAY )'
SELECT        ,'PGM(ISRECMBR) PARM(MRXSELECT)'
SETLANG       ,'PGM(ISRECMBR) PARM(MRXSETLA)'
SIGN          ,'PGM(ISRECMBR) PARM(MRXSIGN )'
SIGNAL        ,'PGM(ISRECMBR) PARM(MRXSIGNAL)'
SOURCELINE    ,'PGM(ISRECMBR) PARM(MRXSOURCE)'
SPACE         ,'PGM(ISRECMBR) PARM(MRXSPACE)'
STORAGE       ,'PGM(ISRECMBR) PARM(MRXSTOR)'
STRIP         ,'PGM(ISRECMBR) PARM(MRXSTRIP)'
SUBCOM        ,'PGM(ISRECMBR) PARM(MRXSUBCO)'
SUBSTR        ,'PGM(ISRECMBR) PARM(MRXSUBST)'
SUBWORD       ,'PGM(ISRECMBR) PARM(MRXSUBW)'
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(MRXTRANSL)'
TRANSCODE     ,'PGM(ISRECMBR) PARM(MRXTRANSC)'
TRANSID       ,'PGM(ISRECMBR) PARM(MRXTRANSID)'
TRIGGER       ,'PGM(ISRECMBR) PARM(MRXTRIGGER)'
TS             ,'PGM(ISRECMBR) PARM(MRXTS )'
UNTIL         ,'PGM(ISRECMBR) PARM(MRXUNTIL)'
USERID        ,'PGM(ISRECMBR) PARM(MRXUSERI)'
VALUE         ,'PGM(ISRECMBR) PARM(MRXVALUE)'
VERIFY        ,'PGM(ISRECMBR) PARM(MRXVERIFY)'
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)'
XCHANGE       ,'PGM(ISRECMBR) PARM(MRXXCHANGE)'
X2B           ,'PGM(ISRECMBR) PARM(MRXX2B )'
X2C           ,'PGM(ISRECMBR) PARM(MRXX2C )'
X2D           ,'PGM(ISRECMBR) PARM(MRXX2D )'
' ',' ',' ' +', '?')
IF (&ZSEL = '??')
&ZEMTEMP = &ZCMD
&ZMSG = .MSG
VPUT(ZMSG) PROFILE

53
<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(PSO1001)</td>
<td>(CMD)</td>
<td>ABBREV</td>
</tr>
<tr>
<td>(PSO1002)</td>
<td>(CMD)</td>
<td>ABS</td>
</tr>
<tr>
<td>(PSO1003)</td>
<td>(CMD)</td>
<td>ADDRESS</td>
</tr>
<tr>
<td>(PSO1004)</td>
<td>(CMD)</td>
<td>ARG</td>
</tr>
<tr>
<td>(PSO1005)</td>
<td>(CMD)</td>
<td>BITAND</td>
</tr>
<tr>
<td>(PSO1006)</td>
<td>(CMD)</td>
<td>BITOR</td>
</tr>
<tr>
<td>(PSO1007)</td>
<td>(CMD)</td>
<td>BITMOR</td>
</tr>
<tr>
<td>(PSO1008)</td>
<td>(CMD)</td>
<td>BOOLEAN</td>
</tr>
<tr>
<td>(PSO1009)</td>
<td>(CMD)</td>
<td>B2X</td>
</tr>
<tr>
<td>(PSO1010)</td>
<td>(CMD)</td>
<td>CALL</td>
</tr>
<tr>
<td>(PSO1011)</td>
<td>(CMD)</td>
<td>CENTER</td>
</tr>
<tr>
<td>(PSO1012)</td>
<td>(CMD)</td>
<td>COMPARE</td>
</tr>
<tr>
<td>(PSO1013)</td>
<td>(CMD)</td>
<td>CONDITION</td>
</tr>
<tr>
<td>(PSO1014)</td>
<td>(CMD)</td>
<td>COPIES</td>
</tr>
<tr>
<td>(PSO1015)</td>
<td>(CMD)</td>
<td>C2D</td>
</tr>
<tr>
<td>(PSO1016)</td>
<td>(CMD)</td>
<td>C2X</td>
</tr>
<tr>
<td>(PSO1017)</td>
<td>(CMD)</td>
<td>DATETYPE</td>
</tr>
<tr>
<td>(PSO1018)</td>
<td>(CMD)</td>
<td>DATE</td>
</tr>
<tr>
<td>(PSO1019)</td>
<td>(CMD)</td>
<td>DELESTACK</td>
</tr>
<tr>
<td>(PSO1020)</td>
<td>(CMD)</td>
<td>DELSTR</td>
</tr>
<tr>
<td>(PSO1021)</td>
<td>(CMD)</td>
<td>DEDWORD</td>
</tr>
<tr>
<td>(PSO1022)</td>
<td>(CMD)</td>
<td>DIGITS</td>
</tr>
<tr>
<td>(PSO1023)</td>
<td>(CMD)</td>
<td>DO</td>
</tr>
<tr>
<td>(PSO1024)</td>
<td>(CMD)</td>
<td>DROP</td>
</tr>
<tr>
<td>(PSO1025)</td>
<td>(CMD)</td>
<td>DROBUF</td>
</tr>
<tr>
<td>(PSO1026)</td>
<td>(CMD)</td>
<td>DZC</td>
</tr>
<tr>
<td>(PSO1027)</td>
<td>(CMD)</td>
<td>DXZ</td>
</tr>
<tr>
<td>(PSO1028)</td>
<td>(CMD)</td>
<td>ERRORTEXT</td>
</tr>
<tr>
<td>(PSO1029)</td>
<td>(CMD)</td>
<td>EXECIO</td>
</tr>
<tr>
<td>(PSO1030)</td>
<td>(CMD)</td>
<td>EXECUTIL</td>
</tr>
<tr>
<td>(PSO1031)</td>
<td>(CMD)</td>
<td>EXIT</td>
</tr>
<tr>
<td>(PSO1032)</td>
<td>(CMD)</td>
<td>EXTERNALS</td>
</tr>
<tr>
<td>(PSO1033)</td>
<td>(CMD)</td>
<td>FIND</td>
</tr>
<tr>
<td>(PSO1034)</td>
<td>(CMD)</td>
<td>FORM</td>
</tr>
<tr>
<td>(PSO1035)</td>
<td>(CMD)</td>
<td>FORMAT</td>
</tr>
<tr>
<td>(PSO1036)</td>
<td>(CMD)</td>
<td>FUZ</td>
</tr>
<tr>
<td>(PSO1037)</td>
<td>(CMD)</td>
<td>GETMSG</td>
</tr>
<tr>
<td>(PSO1038)</td>
<td>(CMD)</td>
<td>HE</td>
</tr>
<tr>
<td>(PSO1039)</td>
<td>(CMD)</td>
<td>HI</td>
</tr>
<tr>
<td>(PSO1040)</td>
<td>(CMD)</td>
<td>HT</td>
</tr>
<tr>
<td>(PSO1041)</td>
<td>(CMD)</td>
<td>IIF</td>
</tr>
<tr>
<td>(PSO1042)</td>
<td>(CMD)</td>
<td>INDEX</td>
</tr>
<tr>
<td>(PSO1043)</td>
<td>(CMD)</td>
<td>INSERT</td>
</tr>
<tr>
<td>(PSO1044)</td>
<td>(CMD)</td>
<td>INTERPRET</td>
</tr>
<tr>
<td>(PSO1045)</td>
<td>(CMD)</td>
<td>ITERATE</td>
</tr>
<tr>
<td>(PSO1046)</td>
<td>(CMD)</td>
<td>JUSTIFY</td>
</tr>
<tr>
<td>(PSO1047)</td>
<td>(CMD)</td>
<td>LASTPOS</td>
</tr>
<tr>
<td>(PSO1048)</td>
<td>(CMD)</td>
<td>LEAVE</td>
</tr>
<tr>
<td>(PSO1049)</td>
<td>(CMD)</td>
<td>LEFT</td>
</tr>
<tr>
<td>(PSO1050)</td>
<td>(CMD)</td>
<td>LENGTH</td>
</tr>
<tr>
<td>(PSO1051)</td>
<td>(CMD)</td>
<td>LENGTHSIZE</td>
</tr>
<tr>
<td>(PSO1052)</td>
<td>(CMD)</td>
<td>LISTDSI</td>
</tr>
<tr>
<td>(PSO1053)</td>
<td>(CMD)</td>
<td>MAXBUF</td>
</tr>
<tr>
<td>(PSO1054)</td>
<td>(CMD)</td>
<td>MAX</td>
</tr>
<tr>
<td>(PSO1055)</td>
<td>(CMD)</td>
<td>MIN</td>
</tr>
<tr>
<td>(PSO1056)</td>
<td>(CMD)</td>
<td>MSG</td>
</tr>
<tr>
<td>(PSO1057)</td>
<td>(CMD)</td>
<td>MSVVAR</td>
</tr>
<tr>
<td>(PSO1058)</td>
<td>(CMD)</td>
<td>NEWSTACK</td>
</tr>
<tr>
<td>(PSO1059)</td>
<td>(CMD)</td>
<td>NOF</td>
</tr>
<tr>
<td>(PSO1060)</td>
<td>(CMD)</td>
<td>NUMERIC</td>
</tr>
<tr>
<td>(PSO1061)</td>
<td>(CMD)</td>
<td>OPTIONS</td>
</tr>
<tr>
<td>(PSO1062)</td>
<td>(CMD)</td>
<td>OUTTRAP</td>
</tr>
<tr>
<td>(PSO1063)</td>
<td>(CMD)</td>
<td>OVERLAY</td>
</tr>
<tr>
<td>(PSO1064)</td>
<td>(CMD)</td>
<td>PARSER</td>
</tr>
<tr>
<td>(PSO1065)</td>
<td>(CMD)</td>
<td>POS</td>
</tr>
<tr>
<td>(PSO1066)</td>
<td>(CMD)</td>
<td>PROCEDURE</td>
</tr>
<tr>
<td>(PSO1067)</td>
<td>(CMD)</td>
<td>PROMPT</td>
</tr>
<tr>
<td>(PSO1068)</td>
<td>(CMD)</td>
<td>PULL</td>
</tr>
<tr>
<td>(PSO1069)</td>
<td>(CMD)</td>
<td>PUSH</td>
</tr>
<tr>
<td>(PSO1070)</td>
<td>(CMD)</td>
<td>QBUF</td>
</tr>
<tr>
<td>(PSO1071)</td>
<td>(CMD)</td>
<td>QLEM</td>
</tr>
<tr>
<td>(PSO1072)</td>
<td>(CMD)</td>
<td>QSTACK</td>
</tr>
<tr>
<td>(PSO1073)</td>
<td>(CMD)</td>
<td>QUEUE</td>
</tr>
<tr>
<td>(PSO1074)</td>
<td>(CMD)</td>
<td>QUEUED</td>
</tr>
<tr>
<td>(PSO1075)</td>
<td>(CMD)</td>
<td>RANDOM</td>
</tr>
<tr>
<td>(PSO1076)</td>
<td>(CMD)</td>
<td>RETURN</td>
</tr>
<tr>
<td>(PSO1077)</td>
<td>(CMD)</td>
<td>REVERSE</td>
</tr>
<tr>
<td>(PSO1078)</td>
<td>(CMD)</td>
<td>RIGHT</td>
</tr>
<tr>
<td>(PSO1079)</td>
<td>(CMD)</td>
<td>RT</td>
</tr>
<tr>
<td>(PSO1080)</td>
<td>(CMD)</td>
<td>SAY</td>
</tr>
<tr>
<td>(PSO1081)</td>
<td>(CMD)</td>
<td>SELECT</td>
</tr>
<tr>
<td>(PSO1082)</td>
<td>(CMD)</td>
<td>SETLANG</td>
</tr>
<tr>
<td>(PSO1083)</td>
<td>(CMD)</td>
<td>SIGN</td>
</tr>
<tr>
<td>(PSO1084)</td>
<td>(CMD)</td>
<td>SIGNAL</td>
</tr>
<tr>
<td>(PSO1085)</td>
<td>(CMD)</td>
<td>SOURCENAME</td>
</tr>
<tr>
<td>(PSO1086)</td>
<td>(CMD)</td>
<td>SPACE</td>
</tr>
<tr>
<td>(PSO1087)</td>
<td>(CMD)</td>
<td>STORAGE</td>
</tr>
<tr>
<td>(PSO1088)</td>
<td>(CMD)</td>
<td>STRIP</td>
</tr>
<tr>
<td>(PSO1089)</td>
<td>(CMD)</td>
<td>SUBCOM</td>
</tr>
<tr>
<td>(PSO1090)</td>
<td>(CMD)</td>
<td>SUBSTR</td>
</tr>
<tr>
<td>(PSO1091)</td>
<td>(CMD)</td>
<td>SUBWORD</td>
</tr>
<tr>
<td>(PSO1092)</td>
<td>(CMD)</td>
<td>SYMBOL</td>
</tr>
<tr>
<td>(PSO1093)</td>
<td>(CMD)</td>
<td>SYSCPUS</td>
</tr>
<tr>
<td>(PSO1094)</td>
<td>(CMD)</td>
<td>SYSDSN</td>
</tr>
<tr>
<td>(PSO1095)</td>
<td>(CMD)</td>
<td>SYSVAR</td>
</tr>
<tr>
<td>(PSO1096)</td>
<td>(CMD)</td>
<td>TE</td>
</tr>
<tr>
<td>(PSO1097)</td>
<td>(CMD)</td>
<td>TIME</td>
</tr>
<tr>
<td>(PSO1098)</td>
<td>(CMD)</td>
<td>TRAPE</td>
</tr>
<tr>
<td>(PSO1099)</td>
<td>(CMD)</td>
<td>TRANSLATE</td>
</tr>
</tbody>
</table>
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 )
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 (WORDINDEX )
FIELD (ZPS01112) VAR (ZCMD) VAL (WORDS )
FIELD (ZPS01113) VAR (ZCMD) VAL (X2B )
FIELD (ZPS01114) VAR (ZCMD) VAL (X2C )
FIELD (ZPS01115) VAR (ZCMD) VAL (X2D )
FIELD (ZPS01116) VAR (ZCMD) VAL (X2E )
)END
## REXX – Sample Skeletons

### MRXPOS

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NPOS</td>
<td>Returns the position of one string, needle, in another haystack.</td>
</tr>
</tbody>
</table>

**Example:**

```plaintext
POS('day','Saturday') -> 6
POS('x','abc def ghi') -> 0
POS('', 'abc def ghi') -> 4
POS('', 'abc def ghi', 5) -> 8
```

### MRXSAY

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAY</td>
<td>Writes a line to the output stream. This typically displays it to the user, but the output destination can depend on the implementation. The result of expression may be of any length. If you omit expression, the null string is written.</td>
</tr>
</tbody>
</table>

**Example:**

```plaintext
Say data = 100
Say 'divided by 4 => ' data/4
/* Displays: "100 divided by 4 => 25" */
```
Work Station Agent
Introduction

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

• Customers finding WSA useful tend to
  • 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
  • Consistently download/upload files from their workstation

• Links
  • TSO Times, Spring 2004, Dave Salt
    • Performing File Transfer using ISPF Work Station Agent (WSA)
  • Appendix A, Installing the Client/Server Component
    • ISPF – V1R13 – Users Guide – Vol 1
Pros & Cons

• Pros
  • Easily transfer all members in a PDS file to a folder
  • SPLIT adds a new logical screen, acts like START

• Cons
  • 3270 emulator must be configured to use
    • Display Model = Model 2 - 24 rows 80 chrs
  • Maximum fixed window size, can not enlarge
  • Window can be resized smaller, but hides screen
  • Work Station Agent must be active
  • Not supported
    • Character-level color, intensity, and highlighting
    • Field-level intensity and highlighting
  • FIND does not work in browse mode
  • Can not initiate GUI display while in split screen mode
Install on PC

- Download below PDS member into Windows folder called “C:\ISPF”
  - SYS1.SISPGUI(ISPGUINX)

- Double click on “C:\ISPF\ISPGUINX.EXE”
  - Type in “C:\ISPF\” and click “Install”

- Click “OK” button and disregard message

- Click “QUIT” button

- Note
  - ISPGUINX is not Windows 7 (64 bit) compatible
    - Install module on Windows 32 bit system and copy to Windows 7
    - Or open IBM PMR, have IBM uncompress modules and email to you
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”
Activation

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

- ISPF in GUI mode
Commands

- **SPLIT** or **START** creates new GUI screen
- **SWAP LIST** display and select 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.
Links

- ISPF – V1R13 – Dialog Developer’s Guide
- ISPF – V1R13 – Dialog Tag Language Guide
- ISPF – V1R13 – Edit and Edit Macros
- ISPF – V1R13 – Messages and Codes
- ISPF – V1R13 – Planning and Customization
- ISPF – V1R13 – Reference Summary
- ISPF – V1R13 – Services Guide
- ISPF – V1R13 – Users Guide – Vol 1

- REXX – V1R2 – User's Guide
- REXX – V1R13 – Reference
- REXX – V1R13 – UNIX System Services

- SDSF – REXX
- SDSF – V1R13 – Operations and Customization
Any questions or comments please contact

Bruce Koss
Wells Fargo Mainframe Operating Systems Support
Bruce.Koss@WellsFargo.com
(704) 600-8416