REXX and ISPF – Troubleshooting

Bruce Koss
Wells Fargo

Friday, March 6, 2015
10:00 AM - 11:00 AM
16605

www.SHARE.org

#SHAREorg

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

Introduction

- Demonstrate several commands to assist in troubleshooting REXX/ISPF programs on the z/OS Mainframe Operating System

- Links are provided to REXX/ISPF presentations and manuals
  - To open a link in a new browser window
    - Hold down the SHIFT key and click on the link

- Acronyms
  - Interactive System Productivity Facility (ISPF)
  - Restructured Extended Executor Language (REXX)

Agenda

Topic	Items
Locate Module	• PANELID, MSGID • ISPLIBD, DDLIST (ISRDDN), ISRFIND
REXX Debugging	• HILITE • SAY, TRACE, SIGNAL, CALL • RC, SIGL • CONDITION(C|D), ERRORTEXT(RC), SOURCELINE(SIGL) • TSO EXECUTIL TS
ISPF Debugging	• REXX in panels and skeletons • ISRDTLCV, ISPDPTRC, ISPFTTRC, ISPVCALL
Common Problems	• Common problems
Links	• Manuals, SHARE and Websites

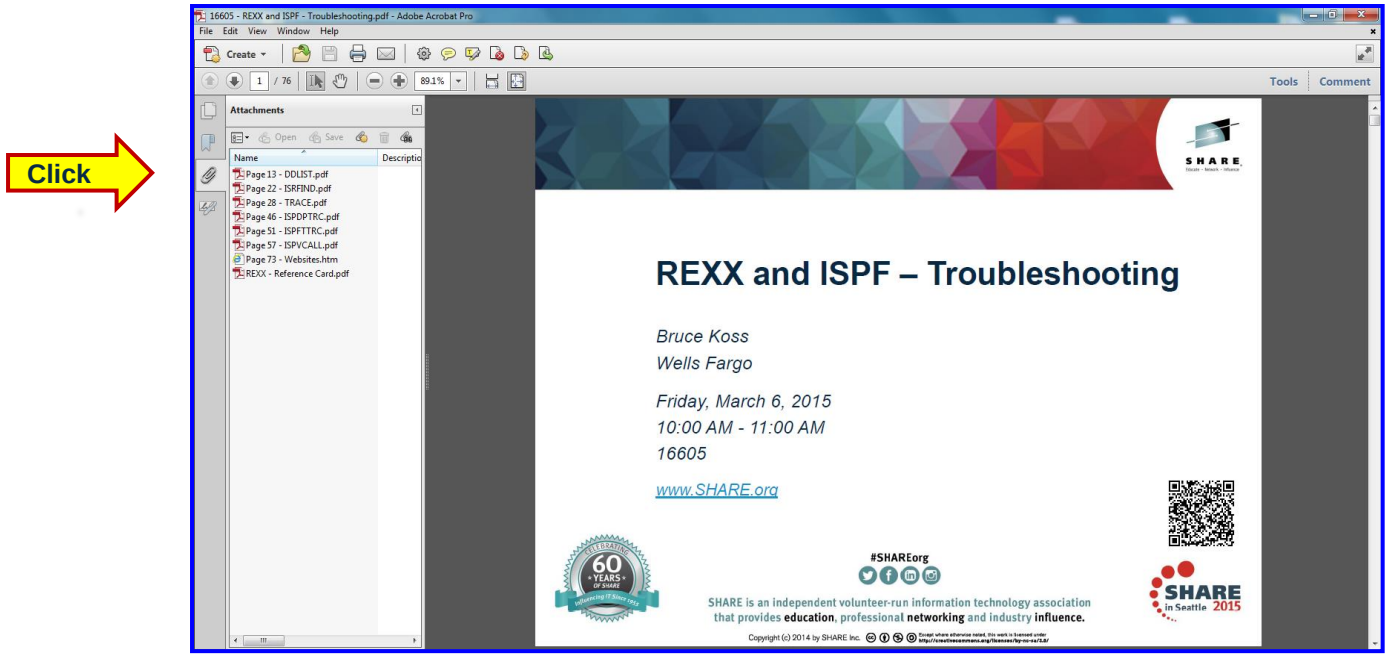
Attachments ( )

- The following attachments are included in this document:

File	Type
Page 13 – DDLIST	PDF
Page 22 – ISRFIND	PDF
Page 28 – TRACE	PDF
Page 46 – ISPDPTRC	PDF
Page 51 – ISPFTTRC	PDF
Page 57 – ISPVCALL	PDF
Page 73 – Websites	HTM
REXX – Reference Card	PDF

Attachments (📎)

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



Disclaimers

- Must use Adobe Reader to view attachments
  - () signifies attachment
- May need to adjust the Adobe Reader zoom value to view text
- Does not cover all REXX and ISPF troubleshooting commands
- Does not cover ISPF Dialog Test, option 7

Locate Module

PANELID [ON | OFF]

- The panel name is displayed below the action bar on the left side
  - ➔ PANELID
- Results:

```
  Menu  Utilities  Compilers  Options  Status  Help
--------------------------------------------------------------------------------
ISR@PRIM                  ISPF Primary Option Menu
Option ===>
                                              More:     +
 0  Settings      Terminal and user parameters          User ID . : R9999
 1  View          Display source data or listings       Time. . . : 05:10
 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 : PANTSOPG
 7  Dialog Test   Perform dialog testing                TSO prefix: R9999
 10 SCLM          SW Configuration Library Manager      System ID : SPRDC
 11 Workplace     ISPF Object/Action Workplace          MVS acct. : TSOPRD
 D  Developer     Developers Workbench                  Release . : ISPF 6.1
 E  Ezyedit       ISPF Productivity Facility
 S  System        System Support Software
 SD SDSF          System Display and Search Facility
 U  User          Personal or Team Software
 C  Changes       Display changes for this release

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

MSGID [ON | OFF]

- The message id is displayed to the left of long messages
  - ➔ MSGID

- Results:

```
  Menu  Utilities  Compilers  Options  Status  Help
--------------------------------------------------------------------------------
                          ISPF Primary Option Menu             Invalid option
Option ===> TEST11
                                              More:     +
0  Settings      Terminal and user parameters            User ID . : R9999
1  View          Display source data or listings         Time. . . : 05:34
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 : PANTSOPG
7  Dialog Test   Perform dialog testing                  TSO prefix: R9999
10 SCLM          SW Configuration Library Manager        System ID : SPRDC
11 Workplace     ISPF Object/Action Workplace            MVS acct. : TSOPRD
D  Developer     Developers Workbench                    Release . : ISPF 6.1
E  Ezyedit       ISPF Productivity Facility
S  System        System Support Software
SD SDSF          System Display and Search Facility
U  User     ------------------------------------------------------
C  Changes | ISPD241 The option that was entered was not valid. |
            ------------------------------------------------------
Enter X to Terminate or Z to Terminate/Logoff
```

ISPLIBD [ddname]

- To obtain a list of LIBDEF data sets
  - ➔ ISPLIBD
- Results:

```
 ---------------------------- LIBDEF Utility ----------------------------
|                        ISPF LIBDEF Display           Row 1 to 9 of 9   |
| Command ===> _____________________________________ Scroll ===> CSR_     |
|                                                                         |
|   Library  Type     USR Identifier                                      |
|   ISPFILE               ** LIBDEF not active **                         |
|   ISPILIB               ** LIBDEF not active **                         |
|   ISPLLIB  LIBRARY      FALLIB                                          |
|                         SYSES.PRD.CWFA.V93051.LOAD                      |
|   ISPMLIB  DATASET      SYSES.PRD.CWFA.V93051.ISPMLIB                   |
|   ISPPLIB  DATASET      SYSES.PRD.CWFA.V93051.ISPPLIB                   |
|   ISPSLIB  DATASET      SYSES.PRD.CWFA.V93051.ISPSLIB                   |
|   ISPTABL               ** LIBDEF not active **                         |
|   ISPTLIB  DATASET      SYSES.PRD.CWFA.V93051.ISPTLIB                   |
| **End**                                                                 |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
```

ISPLIBD – Pros and Cons

- Pros
  - Displays all data sets or a specific LIBDEF data set
  - Can locate DD name

- Cons
  - Does not display ALTLIB data sets
    - To display ALTLIB data sets, type in
      - ➔ DDLIST - or - ➔ TSO ISRDDN
    - To display ALTLIB search order, type in
      - ➔ TSO ALTLIB DISPLAY
  - Can not use Find
  - Can not Browse, Edit or View data sets
  - Can not search for a member
  - Does not display data set attributes (RECFM, LRECL, etc.)

DDLIST [primary command]

- To obtain a list of allocated data sets

  ➔ DDLIST - or - ➔ TSO ISRDDN

- Results:

```
                          Current Data Set Allocations            Row 1 of 77
Command ===> _____________________________________________ Scroll ===> CSR_

 Volume   Disposition Act DDname   Data Set Name   Actions: B E V M F C I Q
 SYS614   SHR,KEEP    > _ FALLIB   SYSES.PRD.CWFA.V93051.LOAD
 PSU320   SHR,KEEP    > _ ISPLLIB  PRDSS.R9999.ISPLLIB
 SYS641   SHR,KEEP    > _          SYSSS.ISPF.ISPLLIB
 SYS609   SHR,KEEP    > _          SYSES.PRD.COMMON.LOADLIB
 PSU312   SHR,KEEP    > _          PRDES.#CSF.LOADLIB
 SYS624   SHR,KEEP    > _          SYSES.PRD.EXAMINE.CAILIB
 SYS648   SHR,KEEP    > _          SYSMJ.#ISPF.VUSR.SCSQAUTH.MNPRDGP2.VERB
 SYS627   SHR,KEEP    > _          SYSDZ.SDSNLOAD
 PWK012   NEW,DEL     > _ ISPLST1  SYS12150.T051051.RA000.R9999.R0192678
 PWK183   NEW,DEL     > _ ISPLST2  SYS12150.T051051.RA000.R9999.R0192679
 PSU331   SHR,KEEP    > _ ISPMLIB  PRDSS.R9999.ISPMLIB
 SYS623   SHR,KEEP    > _          SYSSS.ISPF.ISPMLIB
 SPRSY2   SHR,KEEP    > _          SYS1.ISPF.MLIB
 SYS619   SHR,KEEP    > _          SYSAX.OP.WFP.OPSMLIB
 SYS634   SHR,KEEP    > _          SYSAX.OP.SPRDC.OPSMLIB
 SYS629   SHR,KEEP    > _          SYSES.PRD.COMMON.MSGS
 PSU305   SHR,KEEP    > _ ISPPLIB  PRDSS.R9999.ISPPLIB
 DPS032   SHR,KEEP    > _          PRDSS.ISPF.ISPPLIB
 DSY082   SHR,KEEP    > _          SYSSS.ISPF.MENU.ISPPLIB
```

DDLIST – Primary Commands ()

DDLIST – Primary Commands	
Command	**Description**
APFLIB	Display APF data sets
CLIST or SAVE	Create CLIST to allocate data sets
CON	Display existing ENQ contentions
COUNT	Count and display number of members
CUSTOM	Display ISPF settings
DUPLICATES	Scan and display duplicate members
ENQ	Display ENQs
EXCLUDE [ddname]	Exclude DD name
FIND [string]	Find text string
LONG	Separate data sets from DD name
LNK	Display Linklist data sets
LOCATE [ddname]	Locate DD name

DDLIST – Primary Commands ()

DDLIST – Primary Commands	
Command	**Description**
MEMBER [name|*] [ddname]	Find member
MLIST	Display ISPF modules release / PTF
SHORT	Connect data sets to DD name
ONLY [ddname]	Display only DD names
PARMLIB	Display parmlib data sets
RESET	Display all DD names

- [ddname] can be a partial name, for example
  ➔ M TEST010 ISP - or - ➔ O ISP
- The primary commands above can be passed as a parameter to DDLIST
  ➔ DDLIST ENQ
- Once in DDLIST, press F1 (HELP) and ENTER to display all commands

DDLIST – Line Commands ()

DDLIST – Line Commands	
Command	**Description**
B	Browse the first sixteen data sets or a single data set
C	Compress a PDS using the existing allocation
E	Edit the first sixteen data sets or a single data set
F	Free the entire DDNAME
I	Provide additional data set information
M	Show an enhanced member list for the first 16 data sets or a single data set
Q	Display list of users or jobs using a data set
V	View the first sixteen data sets or a single data set

DDLIST M TEST*

- Invoke DDLIST and search for members using a wild card (*)
  - ➔ DDLIST M TEST*
- Results:

```
                          Current Data Set Allocations          Row 31 of 71
Command ===> _____________________________________________ Scroll ===> CSR_

  Message              Act DDname   Data Set Name    Actions: B E V M F C I Q
Member: TEST*          >   ISPSLIB  SYSAX.OP.WFP.OPSSLIB
                       >           SYSAX.OP.SDV26.OPSSLIB
Member: TEST*          >           SYSES.PRD.COMMON.SKELS
                       > _ ISPTABL  PRDSS.R9999.ISPTLIB
                       > _ ISPTLIB  PRDSS.R9999.ISPTLIB
                       > _          SYSSS.ISPF.MENU.ISPTLIB
                       > _          SYSSS.ISPF.ISPTLIB
                       > _          SYS1.ISPF.TLIB
                       > _          SYSAX.OP.WFP.OPSTLIB
                       > _          SYSAX.OP.SDV26.OPSTLIB
                       > _          SYSAX.SV.SCSYISPF
                       > _ ISP08427 SYSSS.ISPF.$STATS.Y2011
                       > _ ISP08428 SYSSS.ISPF.$STATS.Y2010
                       > _ ISP08431 SYSSS.ISPF.$STATS.Y2011
                       > _ ISP08432 SYSSS.ISPF.$STATS.Y2010
                       > _ OPSCOMP  SYSAX.OP.WFP.OPSEXEC
                       > _ OPSEXEC  SYSAX.OP.WFP.OPSEXEC
```

DDLIST – Pros

- Easily locate a member
  - The member name can contain asterisks (*) to signify a wild card character
  - Displays the members found, not the entire directory
  - Browse, edit or view a specific member
  - Search for a particular DD name
- Large selection of line commands
  - Browse, Edit, View, Member List, Free, Compress, Info, enQue
- Find and display duplicate members
- Counts the number of members in each PDS data set
- Displays data set attributes (RECFM, LRECL, etc.)
- Display APF, LINK, LPA, PARMLIB, ENQ, etc.
- The tutorial is outstanding, no need for a manual

DDLIST – Cons

- LIBDEF data sets are displayed as ISP#####
- ALTLIB data sets are displayed as SYS#####
- Can not use the percent sign (%) as a wild card character in the member name
- Security may prevent access to panels or executing commands

ISRFIND

- Search a particular DD name for member
  - ➔ ISRFIND
- Insert DD name and member name:

```
ISRFIND--ISPF/PDF LEVEL2 DIAGNOSTIC AID                    APPLID   - ISR
                                                           ISPF LVL - ISPF 6.1
                                                           PDF  LVL - PDF 6.1
                                                           TIME     - 19:44
COMMAND ===>

  DD Name          ==> SYSPROC   DD to search (blank for all)

  Dataset info     ==>           (blank) for no dataset info required
                                 B for BASIC info (DSORG RECFM LRECL BLKSIZE
                                 F for FULL info  (BASIC+ALLOC/USAGE+DIRECTORY

  Member Name      ==> TEST010   Member to search for (not required)
  LOADMOD          ==>           Y if the member a LOADMOD.
                                 (adds search of the LPA, LPALIST and LINKLIST)

  Save Data        ==>           (blank) do not save ISRFIND output
                                 S Save data in new dataset
                                   'R9999.ISRFIND.SAVE'
                                 A Append data to existing
                                   'R9999.ISRFIND.SAVE'
```

ISRFIND – Results

- If found, the member will be displayed to the right of the data set name
- Can browse or edit a PDS data set

```
COMMAND INPUT ===>                                            Row 1 to 15 of 15
ENTER   B TO BROWSE    E TO EDIT
S DD/DATASET                                          MEMBER
================================================================================

   >> SYSPROC <<
_  PRDSS.R9999.ISPREXX
_  SYSSS.ISPF.MENU.ISPREXX
_  SYSSS.ISPF.ISPREXX                                 TEST010
_  SYSSS.ISPF.EDIT.MACROS.ISPREXX
_  SYS1.ISRCLIB
_  SYS1.CMDPROC
_  SYSAX.OP.SPRDC.VBCLIST
_  SYSAX.CMDPROC.CLIST
_  SYSES.PRD.JCLPREP.CLIST
_  SYSES.PRD.COMMON.REXX
_  SYSES.PRD.COMMON.CLIST
_  SYSPL.REXX
_  SYSPL.CLIST
****************************** Bottom of data ********************************
```

ISRFIND – Pros and Cons

- Pros
  - Easily locate a member in a specific DD name
  - Displays data set attributes (RECFM, LRECL, etc.)
  - Can search LPA and LINK LIST for modules
  - Can save output to a sequential file
  - Can execute in a REXX program

```
X = OUTTRAP("REC.")
"ISRFIND DD(ISPPLIB) M(ISR@PRIM) NOISPF"
X = OUTTRAP("OFF")
```

- Cons
  - Can not use the asterisk (*) or the percent sign (%) as wild card characters
  - Only Browse and Edit, no View
  - Browsing or Editing displays the PDS directory, not the member
  - Invalid data causes an error message and terminates the utility
  - No tutorial or documentation on how to use it

Links

- ISPLIBD
  - LIBDEF Display Utility
    - ISPF – V2R1 – Services Guide

- DDLIST (ISRDDN)
  - Appendix G, ISRDDN diagnostic utility
    - ISPF – V2R1 – User's Guide – Vol 1

- ISRFIND
  - Information can be obtained by browsing the comments at the beginning of the ISRFIND member (📎)

REXX
Debugging

Debugging

Topic	Command
ISPF Editor	• HILITE
REXX Commands	• SAY • TRACE [##] [?|!] [N|A|C|E|F|I|L|O|R|S] • SIGNAL ON [condition] NAME [label] • CALL ON [condition] NAME [trapname]
REXX Variables	• RC • SIGL
REXX Functions	• CONDITION(C|D) • ERRORTEXT(RC) • SOURCELINE(SIGL)
TSO Command	• TSO EXECUTIL TS

HILITE

- To turn on enhanced color while editing a member, type in HILITE
  - ➔ HI
- Type in 14 for ‘Language’, 3 for ‘Coloring’ and select the three ‘Options’ (/)

```
+-------------------------------------------------------------------------------+
|  File  Languages  Colors  Help                                                |
| ----------------------------------------------------------------------------- |
|                           Edit Color Settings                                 |
| Command ===>                                                                  |
|                                                                               |
|                                                                               |
|                                                                 More:     +   |
| Language: 14   1. Automatic      Coloring: 3  1. Do not color program         |
|                2. Assembler                   2. Color program              |
|                3. BookMaster                  3. Both IF and DO logic       |
|                4. C                           4. DO logic only              |
|                5. COBOL                       5. IF logic only              |
|                6. HTML                                                      |
|                7. IDL             Enter "/" to select option                |
|                8. ISPF DTL        /  Parentheses matching                   |
|                9. ISPF Panel      /  Highlight FIND strings                 |
|               10. ISPF Skeleton  /  Highlight cursor phrase                 |
|               11. JCL                                                       |
|               12. Pascal          Note: Information from this panel is      |
|               13. PL/I            saved in the edit profile.                |
|               14. REXX                                                      |
|               15. SuperC                                                    |
+-------------------------------------------------------------------------------+
```

HILITE – Results

- Host commands, DO-END, IF-ELSE, parentheses and literals will be highlighted

```
000001 /*-------------------------------* REXX *-------------------------------*/
000002 /*  PURPOSE:  Demonstrate HILITE Command                                 */
000003 /*-------------------------------------------------------------------------*/
000004 X = MSG("OFF")
000005 A = 1
000006
000004 IF (A = 1) THEN DO
000005    SAY "A is equal to 1"
000006    SAY "A is equal to 1"
000007 END
000008 ELSE
000009    SAY "A is not equal to 1"
000010
000011 DO J = 1 TO 10
000012    SAY "J = "J
000013 END
000014
000015 ADDRESS TSO
000016   "ALLOC FI(TEST010) SHR DA('TSTSS.R9999.HELP')"
000017 ADDRESS ISPEXEC
000018   "SELECT CMD(%TEST020) NEWAPPL(TEST)"
```

Commands – Pros and Cons

- SAY
  - Most useful, but underutilized
  - Fastest way to debug
- TRACE
  - Great way to learn REXX or how a program is coded
  - Use TRACE L to quickly find a subroutine causing an issue
  - Place TRACE R and TRACE O around problem code
  - Coding TRACE in a procedure will only trace the subroutine
- SIGNAL and CALL
  - Be aware that ERROR takes control when an RC is greater than zero
  - The SIGNAL label must be in the main REXX
    - Whereas CALL can be external
- TSO EXECUTIL TS
  - Turn on interactive tracing without having to edit the program
  - Downside is it traces all lines
  - Use TRACE - ## or TRACE ## to advance through the program

TRACE – Options ()

TRACE [!|?] [N|A|C|E|F|I|L|O|R|S]	
Prefix	**Description**
!	Simulates execution of commands
?	Interactive debugging
Char	**Description**
All	All clauses
Command	All commands and non-zero RCs are displayed
Error	Any executed command resulting in an error or failure
Failure	Any executed command resulting in failure (same as 'N')
Intermediate	All commands and intermediate results
Labels	Only executed labels
Normal	Only commands resulting in a negative RC (default)
Off	Turn tracing off (or no argument)
Results	All commands and (no intermediate) results
Scan	Trace remaining clauses without executing

TRACE – ? (📎)

Interactive Debugging Commands	
Action	**Description**
ENTER	Execute next statement
=	Re-execute last statement
[REXX cmd]	Execute REXX command - For DO-END loops use semicolon (;)
TRACE ##	Execute and display specified number of statements, without pausing - Positive number - TRACE 10 will execute and display the next 10 lines of code
TRACE - ##	Execute specified number of statements, without displaying or pausing - Negative number - TRACE -10 will execute (but not display) the next 10 lines of code
TRACE [option]	Change tracing - TRACE O will turn off tracing
EXIT	End REXX program

TRACE – Output ()

TRACE Output	
Format	**Description**
-	Actual command
+++	Trace message (non zero RC, prompt message, etc.)
>>>	Results from expression, parsed value or value returned from subroutine
>.>	Value “assigned” to a placeholder during parsing
>C>	Name of a compound variable
>F>	Result of a function call
>L>	Literal (string, uninitialized variable or constant symbol)
>O>	Result of an operation on two terms
>P>	Result of a prefix operation
>V>	Contents of a variable

TRACE – Example

Code

```
/*------------------------------* REXX *------------------------------*/
/*  Purpose:  Trace all clauses, intermediate and evaluations        */
/*-------------------------------------------------------------------*/
TRACE I

A = RANDOM()

IF (A < 100) THEN
   SAY "A is less than 100"
ELSE
   SAY "A is greater than 99"
```

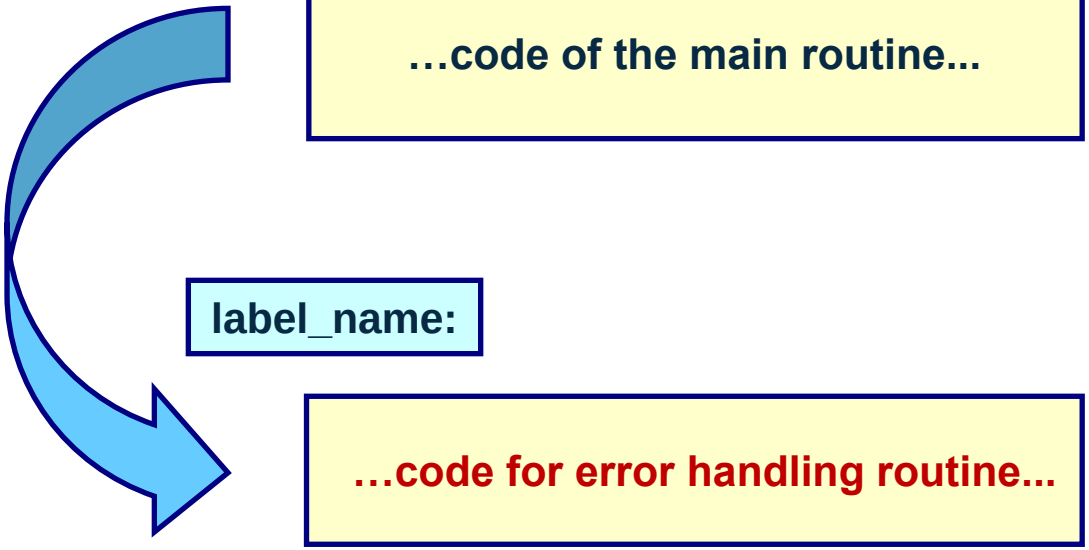
Results

```
      6 *-* A = RANDOM()
        >F>   "603"
      8 *-* IF (A < 100)
        >V>   "603"
        >L>   "100"
        >O>   "0"
     10 *-* ELSE
     11 *-*  SAY "A is greater than 99"
        >L>    "A is greater than 99"

 A is greater than 99
```

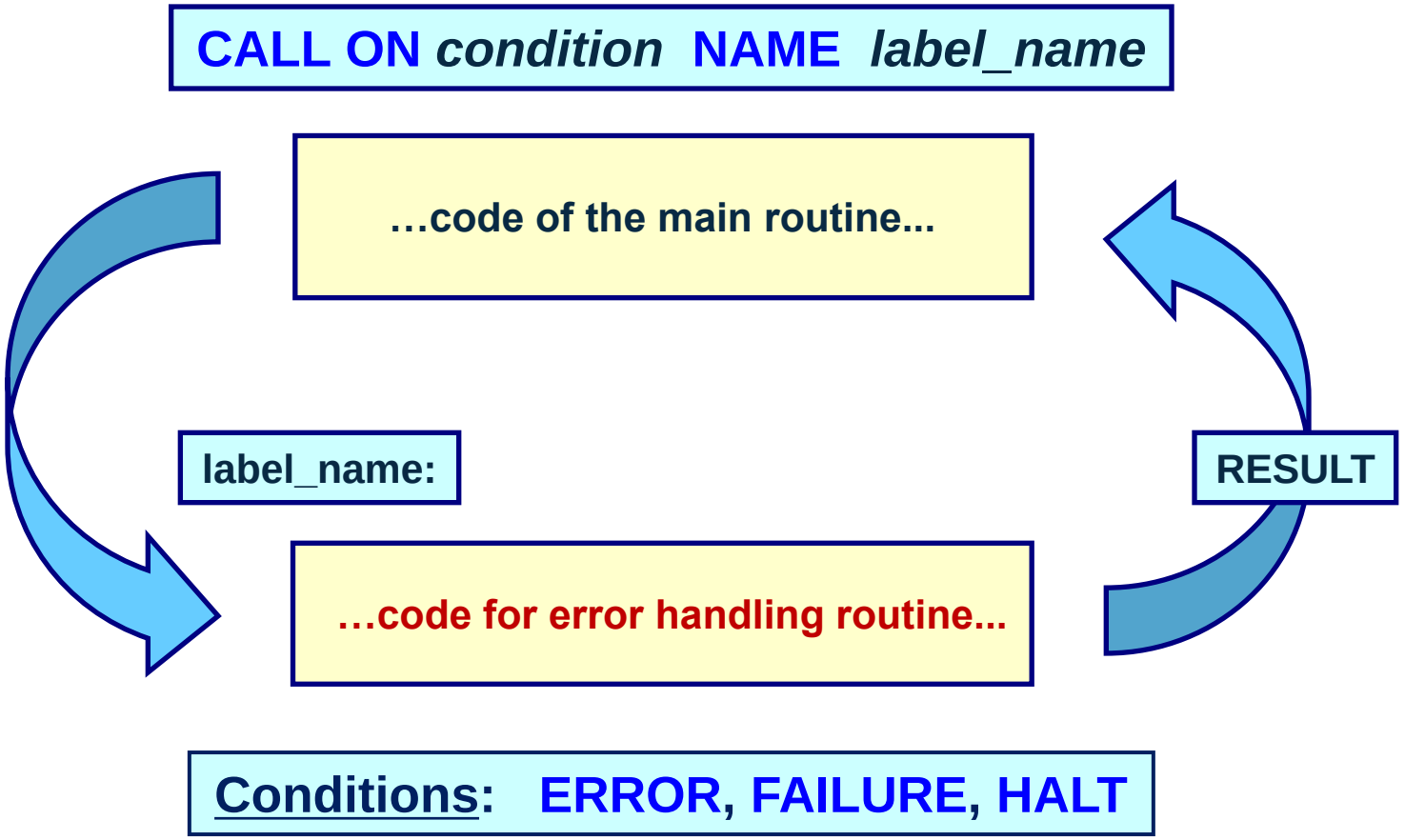
SIGNAL

SIGNAL ON *condition* NAME *label_name*



Conditions: ERROR, FAILURE, HALT, NOVALUE, SYNTAX

CALL


CALL ON condition NAME label_name
…code of the main routine...
label_name:
RESULT
…code for error handling routine...
Conditions: ERROR, FAILURE, HALT

SIGNAL and CALL

SIGNAL ON [condition] NAME [label]	
Condition	**Description**
ERROR	Error upon return (positive return code)
FAILURE	Failure upon return (negative return code)
HALT	An external attempt was made to interrupt and end execution
NOVALUE	Attempt was made to use an uninitialized variable
SYNTAX	Language processing error found during execution
CALL ON [condition] NAME [trapname]	
Condition	**Description**
ERROR	Error upon return (positive return code)
FAILURE	Failure upon return (negative return code)
HALT	An external attempt was made to interrupt and end execution

SIGNAL – Example

Code

```
/*------------------------------* REXX *------------------------------*/
/*    PURPOSE:  Demonstrate SIGNAL command                            */
/*--------------------------------------------------------------------*/
SIGNAL ON ERROR   NAME ERROR          /* RC is positive                */
SIGNAL ON FAILURE NAME ERROR          /* RC is negative                */
SIGNAL ON HALT    NAME ERROR          /* External interrupt            */
SIGNAL ON NOVALUE NAME ERROR          /* Uninitialized variable        */
SIGNAL ON SYNTAX  NAME ERROR          /* Syntax error                  */

VAR10 = VAR20 + 1                     /* VAR20 not initialized         */

EXIT
/*--------------------------------------------------------------------*/
/*                     Display Error Information                      */
/*--------------------------------------------------------------------*/
ERROR:
  SAY "*"COPIES("-",70)"*"
  SAY "*  Condition   :" CONDITION('C') CONDITION('D')  /* Cond/Desc */
  SAY "*  Line Number :" SIGL                          /* Line No.  */
  SAY "*  Code        :" STRIP(SOURCELINE(SIGL))       /* Src code  */
  SAY "*"COPIES("-",70)"*"
EXIT
```

Results

```
*--------------------------------------------------------------------*
*  Condition   : NOVALUE VAR20
*  Line Number : 10
*  Code        : VAR10 = VAR20 + 1     /* VAR20 not initialized */
*--------------------------------------------------------------------*
```

Functions and Variables

- Functions
  - CONDITION('C'|'D')
    - Returns the trapped error condition (C) or description (D)
  - ERRORTEXT(RC)
    - Returns the error message associated with return code
  - SOURCELINE(SIGL)
    - Returns the source code in error

- Variables
  - RC
    - Contains the return code from the host command
  - SIGL
    - Contains the line number of the instruction when SIGNAL or CALL is executed

EXECUTIL

- Provides the capability to turn tracing on without having to edit the program
- Turn interactive trace on
  - ➔ TSO EXECUTIL TS
- Execute REXX program
  - ➔ TSO [rexx program] [parms]
- Terminate tracing and
  - Continue executing the program
    - ➔ TE
  - Or terminate the program
    - ➔ EXIT

Links

- TRACE
  - Chapter 3, Keyword instructions
    - REXX – V2R1 – Reference
  - Chapter 9, Diagnosing Problems Within an Exec
    - REXX – V2R1 – User's Guide
- SIGNAL and CALL
  - Chapter 3, Keyword instructions, and Chapter 7, Conditions and traps
    - REXX – V2R1 – Reference
- RC and SIGL
  - Chapter 7, Conditions and condition traps
    - REXX – V2R1 – Reference
- EXECUTIL
  - Chapter 10, TSO/E REXX commands, and Chapter 11, Debug aids
    - REXX – V2R1 – Reference

ISPF
Debugging

ISPF – Debugging

- Panel
  - Edit macro to convert non-displayable characters
    - ➔ ISRDTLCV
  - Add REXX code containing SAYs
- Skeleton
  - Add REXX code containing SAYs
- IBM Trace Utilities
  - Panels
    - ➔ ISPDPTRC
  - Skeletons (File Tailoring)
    - ➔ ISPFTTRC
  - ISPF
    - ➔ ISPVCALL

ISRDTLCV

- Edit macro that converts panel of non-displayable characters to viewable characters; edit panel and type in
  - ➔ ISRDTLCV

Before

```
000182 )BODY  CMD(ZCMD)
000183    Menu  Utilities  Compilers  Options  Status  Help
000184  ------------------------------------------------------------------
000185                            ISPF Primary Option Menu
000186  Option ===> Z
000187  SAREA39                                                 GRPBOX1
000188                                                          TMPROWS,
000189
000190
000191
000192
000193
000194
000195
000196
000197
000198
000199
000200
000201  ZEXI
000202       Enter Z to Terminate using log/list defaults
000203 )AREA SAREA39
000204  0  Settings      Terminal and user parameters
000205  1  View          Display source data or listings
000206  2  Edit          Create or change source data
000207  3  Utilities     Perform utility functions
000208  4  Foreground    Interactive language processing
000209  5  Batch         Submit job for language processing
000210  6  Command       Enter TSO or Workstation commands
000211  7  Dialog Test   Perform dialog testing
000212  9  IBM Products  IBM program development products
000213  10 SCLM          SW Configuration Library Manager
```

After

```
.BODY  )BODY  CMD(ZCMD)
 ==CHG> #~ Menu~ Utilities~ Compilers~ Options~ Status~ Help#
 ==CHG> `------------------------------------------------------------------
 ==CHG> #                            !ISPF Primary Option Menu#
 ==CHG> @Option ===>:Z
 ==CHG> \SAREA39                                                 \$GRPBOX1
 ==CHG> \                                                        \#'TMPROWS
 ==CHG> \                                                        \#'
 ==CHG> \                                                        \#'
 ==CHG> \                                                        \#'
 ==CHG> \                                                        \#'
 ==CHG> \                                                        \#'
 ==CHG> \                                                        \#'
 ==CHG> \                                                        \#'
 ==CHG> \                                                        \#'
 ==CHG> \                                                        \#'
 ==CHG> \                                                        \#'
 ==CHG> \                                                        \#¬¬ #
 ==CHG> \                                                        \
 ==CHG> {ZEXI¬¬ #
 ==CHG> #    @Enter{Z}to Terminate using log/list defaults#
 000203 )AREA SAREA39
 ==CHG> *0 ?Settings      #Terminal and user parameters          #
 ==CHG> *1 ?View          #Display source data or listings       #
 ==CHG> *2 ?Edit          #Create or change source data          #
 ==CHG> *3 ?Utilities     #Perform utility functions             #
 ==CHG> *4 ?Foreground    #Interactive language processing       #
 ==CHG> *5 ?Batch         #Submit job for language processing    #
 ==CHG> *6 ?Command       #Enter TSO or Workstation commands     #
 ==CHG> *7 ?Dialog Test   #Perform dialog testing                #
 ==CHG> *9 ?IBM Products  #IBM program development products      #
 ==CHG> *10?SCLM          #SW Configuration Library Manager      #
```

Panel – REXX Code

- In the)INIT or)PROC section, code REXX to display variables

- Example:

```
)PROC
     *REXX(*)                    /* Pass all variables  */
        SAY 'VAR1 = 'VAR1        /* Display variable    */
     *ENDREXX                    /* Terminate REXX code */
)END
```

- Note:
  - Do not prefix "&" in front of variable names

Skeleton – REXX Code

- Anywhere within the skeleton, code REXX to display variables

- Example:

```
)REXX VAR1                    /* Pass variable       */
   SAY 'VAR1 = 'VAR1          /* Display variable    */
)ENDREXX                      /* Terminate REXX      */
```

- Note:
  - Do not prefix "&" in front of variable names

ISPDPTRC

- Traces the processing of the Dialog Manager panel
  - Execution of panel service calls
    - DISPLAY, TBDISPL and TBQUERY
  - Processing that occurs within the panels
    -)ABCINIT,)ABCPROC,)INIT,)REINIT and)PROC sections
- Can be invoked from any ISPF panel
- Dynamically allocates ISPDPTRC (RECFM=VB,LRECL=255)
- Execution:

➔ ISPDPTRC — Turn trace on

```
Loading Panel trace.  To end, reinvoke this command
***
```

➔ TSO [rexx program] [parms] — Execute REXX

➔ ISPDPTRC — Turn trace off

```
Unloading Panel trace...
***
```

ISPDPTRC – Results

```
TLD# Type  Panel    Section Cd  RC  Data
---- ---- --------- ------- -- ---  ------------------------------------------------------------------------------
-->
TLD1 Svc                             TBDISPL CICS1942 PANEL(CICSXRE1)
TLD1 Read CICSXRE1 ---------------- DD=ISPPLIB+2
TLD1 Read CICSXRE1                0  Total Records=45
TLD1 PrcR CICSXRE1 INIT           0    .ZVARS='(ZSCROLLD ID PI SD MCT SRT TCT)'
TLD1 PrcR CICSXRE1 INIT           0    .HELP='#CICSXRE'
TLD1 DspO                          0----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
TLD1 DspO CICSXRE1                 |+--------------------------(%CICS - Cross Reference+)--------------------------
TLD1 DspO CICSXRE1                 |%Command ==>&                                                %Scr =>&CSR +
TLD1 DspO CICSXRE1                 |+                            ~                                 @PF11 ==>
TLD1 DspO CICSXRE1                 |+Command: %F xxx+- Find text   %L xxx+- Locate text   %S xxx+- Sort column
TLD1 DspO CICSXRE1                 ++         %O xxx+- Only text   %R    +- Reset display %H    +- Hide header
TLD1 DspO CICSXRE1                 |+         %D    +- Save to DSN %P    +- Print         %T    +- Totals
TLD1 DspO CICSXRE1                 |+
TLD1 DspO CICSXRE1                 |+Line:    %S+- Display parms   %M+- Dump M0 journals%                 M  S  T
TLD1 DspO CICSXRE1                 |%                                                          P  S  C  R  C
TLD1 DspO CICSXRE1                 1%S    Stc      Proc    A Lpar  L  Ver   Applid   Sysid    Id     I  D  T  T  T
TLD1 DspO CICSXRE1                 |%-  -------- --------  - ----- -  ---  --------  -----  -------  -- -- -- -- --
TLD1 DspO CICSXRE1                 |_ +.CICMS09 .C$CMAS   .Y.SDV22.9 .420 .CICMS09  .CP09  .CIX#TST .NO.NO.NO.S2.NO
TLD1 DspO CICSXRE1                 |_ +.CICSCTMP.C$CMAS   .Y.SDV21.2 .420 .CICMS02  .CP02  .CIX#TST .NO.NO.NO.S2.NO
TLD1 DspO CICSXRE1                 |_ +.CICSCVCM.C$CMAS   .Y.SDV25.V .420 .CICSCVCM .CVCM  .CIX#TST .NO.NO.NO.S2.NO
TLD1 DspO CICSXRE1                 +_ +.CICSC2CM.C$CMAS   .Y.SDV21.2 .420 .CICSC2CM .C2CM  .CIX#TST .NO.NO.NO.S2.NO
TLD1 DspO CICSXRE1                 |_ +.CICSC2CW.C$CMAS   .Y.SDV21.2 .420 .CICSC2CW .C2CW  .CIX#TST .NO.NO.CA.S2.NO
TLD1 DspO CICSXRE1                 |_ +.CICSC9CW.C$CMAS   .N.SDV22.9 .420 .CICSC9CW .C9CW  .         .NO.NO.CA.S2.NO
TLD1 DspO CICSXRE1                 |_ +.CICSEVP7.CICSEVP7 .Y.SDV25.V .420 .CICSEVP7 .EVP7  .CIC#TST .YO.61.CF.E@.CC
TLD1 DspO CICSXRE1                 |_ +.CICSE2A2.CICSE2A2 .Y.SDV21.2 .420 .CICSE2A2 .E2A2  .CIC#TST .YA.Y1.CA.E@.CC
TLD1 DspO CICSXRE1                 2_ +.CICSE2A3.CICSE2A3 .Y.SDV21.2 .420 .CICSE2A3 .E2A3  .CIC#TST .YV.Y1.CA.E@.CC
TLD1 DspO CICSXRE1                 |_ +.CICSE2A4.CICSE2A4 .Y.SDV21.2 .420 .CICSE2A4 .E2A4  .CIC#TST .YK.Y1.CA.E@.CC
TLD1 DspO CICSXRE1                 |_ +.CICSE2A5.CICSE2A5 .Y.SDV21.2 .420 .QAAOR2   .AOR2  .CIC#0AI .8A.61.CA.E@.CC
TLD1 DspO CICSXRE1                 |_ +.CICSE2A6.CICSE2A6 .Y.SDV21.2 .420 .QAAOR3   .AOR3  .CIC#0AI .9A.92.CA.E@.CC
TLD1 DspO CICSXRE1                 |_ +.CICSE2A7.CICSE2A7 .Y.SDV21.2 .420 .QAAOR5   .AOR5  .CIC#0AI .FA.61.CA.E@.CC
TLD1 DspI CICSXRE1                 | SDV26----------------------(%CICS - Cross Reference+)------ Row 1 to 12 of 364
```

ISPDPTRC – Options ()

ISPDPTRC [options]	
Option	**Description**
END	Terminates the trace and does not edit the trace data set
VIEW	Terminates the trace and views the trace data set
LIST	Invokes the Data Set List Utility to display the trace data sets
QUIET	Does not display initialization and termination messages
DISPLAY	Controls tracing the panel as displayed on the terminal
	NONE — No trace during panel display processing
	IN — Trace showing panel, including data entered
	OUT — Trace showing panel as is on screen with attributes
	BOTH — Trace both the In and Out display (default)
PANEL	Controls tracing based on the panel name
	* — Trace all panels (default)
	panel_name — Trace only for a specific panel
	panel_mask — Trace panels matching *panel_mask* (% and *)

ISPDPTRC – Options ()

ISPDPTRC [options]		
Option	**Description**	
READ	Controls tracing when a panel is read into memory	
	NONE	No trace during read processing
	SUMMARY	Summary info and location where the panel was loaded (default)
	DETAIL	Same as Summary and panel source
SCREEN	Controls tracing based on the screen ID	
	0	Trace all logical screens (default)
	*****	Trace the current screen id
	screen_id	Trace a specific screen id

ISPDPTRC – Options ()

ISPDPTRC [options]		
Option	**Description**	
SECTION	Controls tracing through different panel logic sections	
	* | ALL	Trace all skeleton sections
	NONE	Do not trace any skeleton sections
	INIT	Trace)ABCINIT and)INIT sections
	REINIT	Trace)REINIT section
	PROC	Trace)ABCPROC and)PROC sections
	NOINIT	Do not trace)ABCINIT and)INIT sections
	NOREINIT	Do not trace)REINIT section
	NOPROC	Do not trace)PROC section
SERVICE	Controls tracing for DISPLAY, TBDISPL and TBQUERY	
	NONE	No trace records are produced
	DETAIL	Traces DISPLAY, TBDISPL and TBQUERY

ISPFTTRC

- Traces the processing of file tailoring (skeletons) services
  - Execution of file tailoring service calls
    - FTOPEN, FTINCL, FTCLOSE and FTERASE
  - Processing that occurs within the file tailoring code and each statement
- Can be invoked from any ISPF panel
- Dynamically allocates ISPFTTRC (RECFM=VB,LRECL=255)
- Execution:

➔ ISPFTTRC — Turn trace on

```
Loading File Tailoring trace.  To end, reinvoke this command
***
```

➔ TSO [rexx program] [parms] — Execute REXX

➔ ISPFTTRC — Turn trace off

```
Unloading File Tailoring trace...
***
```

ISPFTTRC – Results

```
TLD# Type Skeleton  Rec#   IM IF DO TB  Cd RC Data
---- ---- -------- ------  -- -- -- --  -- -- ----------------------------------------->
TLD1 Svc                                     FTOPEN  TEMP
------------------------------------------- DD=ISP07407  DSN=R9999.SDV26.SPFTEMP1.CNTL
TLD1 SvcR                                 0 FTOPEN  TEMP
TLD1 Svc                                     FTINCL  JOBCARD
------------------------------------------- DD=ISPSLIB   DSN=
TLD1 Read JOBCARD       1                    )CM   ************************************************************
TLD1 Read JOBCARD       2                    )CM   *               JOBCARD RETRIEVED FROM ISPF 0.2            *
TLD1 Read JOBCARD       3                    )CM   ************************************************************
TLD1 Read JOBCARD       4                    )SEL &ZLLGJOB1 ¬= &Z
TLD1 Read JOBCARD       5                    &ZLLGJOB1
TLD1 Read JOBCARD       6                    )ENDSEL
TLD1 Read JOBCARD       7                    )SEL &ZLLGJOB2 ¬= &Z
TLD1 Read JOBCARD       8                    &ZLLGJOB2
TLD1 Read JOBCARD       9                    )ENDSEL
TLD1 Read JOBCARD      10                    )SEL &ZLLGJOB3 ¬= &Z
TLD1 Read JOBCARD      11                    &ZLLGJOB3
TLD1 Read JOBCARD      12                    )ENDSEL
TLD1 Read JOBCARD      13                    )SEL &ZLLGJOB4 ¬= &Z
TLD1 Read JOBCARD      14                    &ZLLGJOB4
TLD1 Read JOBCARD      15                    )ENDSEL
TLD1 Read JOBCARD  -------------------------- Total Records=15
TLD1 Src  JOBCARD       1   1                )CM   ************************************************************
TLD1 Src  JOBCARD       2   1                )CM   *               JOBCARD RETRIEVED FROM ISPF 0.2            *
TLD1 Src  JOBCARD       3   1                )CM   ************************************************************
TLD1 Src  JOBCARD       4   1                )SEL &ZLLGJOB1 ¬= &Z
TLD1 CtlR JOBCARD       4   1  1        T   0 )SEL //R9999TST JOB (SST),'KOSS',MSGCLASS=T,CLASS=R,NOTIFY=&SYSUID ¬=
TLD1 Src  JOBCARD       5   1  1             &ZLLGJOB1
TLD1 DatR JOBCARD       1   1  1           0 //R9999TST JOB (SST),'KOSS',MSGCLASS=T,CLASS=R,NOTIFY=&SYSUID
TLD1 Src  JOBCARD       6   1  1             )ENDSEL
TLD1 CtlR JOBCARD       6   1               0 )ENDSEL
TLD1 Src  JOBCARD       7   1                )SEL &ZLLGJOB2 ¬= &Z
TLD1 CtlR JOBCARD       9   1               0 )ENDSEL
```

ISPFTTRC – Options ()

ISPFTTRC [options]	
Option	**Description**
END	Terminates the trace and does not edit the trace data set
VIEW	Terminates the trace and views the trace data set
LIST	Invokes the Data Set List Utility to display the trace data sets
QUIET	Does not display initialization and termination messages
READ	Controls tracing when a panel is read into memory NONE — No trace during read processing SUMMARY — Summary info and location where panel was loaded (default) DETAIL — Same as Summary and panel source

ISPFTTRC – Options (📎)

ISPFTTRC [options]		
Option	**Description**	
RECORDS	Controls trace records during record processing of skeleton member	
	* | ALL	Trace all skeleton records
	NONE	Do not trace any skeleton records
	SOURCE	Trace the source skeleton record before any processing is done
	DATA	Trace for data records after record processing has completed
	CNTL	Trace for control statements after record processing has completed
	NOSOURCE	Turns off trace for the source skeleton records
	NODATA	Turns off trace records for data records
	NOCNTL	Turns off trace records for control statements
SERVICE	Controls tracing for DISPLAY, TBDISPL and TBQUERY	
	NONE	No trace records are produced
	DETAIL	Traces DISPLAY, TBDISPL and TBQUERY

ISPFTTRC – Options ()

ISPFTTRC [options]	
Option	**Description**
SKELETON	Controls trace records based on the skeleton name
	* | ALL — Trace all skeleton records (default)
	skel_name — Trace only for a specific skeleton
	skel_mask — Trace skeletons matching *skel_mask* (% and *)
TBVARS	For)DOT, displays variables on each iteration through the table
	NONE — No trace records are produced during)DOT processing
	DETAIL — Trace)DOT, display variables on each iteration (default)

ISPVCALL

- Traces
  - Service calls
  - Messages
  - Enques
- Displays
  - Operating system and product levels
  - ISPF exits installed
  - ISPF configuration table
  - Cached panel names
  - History of typed commands
  - List of allocated DD names and data sets
  - Active LIBDEFs
  - System SVC table
  - ISPF CSECTs

ISPVCALL

- Can be invoked from any ISPF panel
- Dynamically allocates ISPTRACE (RECFM=FB,LRECL=80)
- Execution:
  - ➔ ISPVCALL — Turn trace on

    Loading CALL trace. To end, reinvoke this command.

    ***
  - ➔ TSO [rexx program] [parms] — Execute REXX
  - ➔ ISPVCALL — Turn trace off

    Unloading CALL trace...

    ***
- Output lines will begin with >ISP, >ISR, >ENQ, >MSG and >CMD
- To view events in context, type in
  - ➔ X ALL; F ">" 1 ALL

ISPVCALL – Results

```
 - - -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  - 1016 Line(s) not Displayed
 001017 >Msg S:                                TLDMMID=
 001018 >Msg L:
 - - -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -   97 Line(s) not Displayed
 001116 >Msg S:   TSO     - Command  -       TLDMMID=
 001117 >Msg L:%PLPMENU /WFS CICSXREF
 - - -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  - 2 Line(s) not Displayed
 001120 >CMD  %PLPMENU /WFS CICSXREF
 - - -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -   10 Line(s) not Displayed
 001131 >ISPCAT   ISPEXEC Control Errors Return
 001132 >ISPEXEC called from IRXSTAM+A02       05198D6A in PLPA  EPA:05198368
 - - -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -   11 Line(s) not Displayed
 001144 >ISPCAT   ISPEXEC Libdef PLPISPF dataset id('SYSSS.ISPF.MENU.ISPTLIB') STACK
 001145 >ISPEXEC called from IRXSTAM+A02       05198D6A in PLPA  EPA:05198368
 - - -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -   29 Line(s) not Displayed
 001175 >ISPCAT   ISPEXEC TBQuery WFSplp
 001176 >ISPEXEC called from IRXSTAM+A02       05198D6A in PLPA  EPA:05198368
 - - -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -   19 Line(s) not Displayed
 001196 >Msg TLD1- TLDMMID:ISPT034  set in ISPCMG  ->ISPCFI
 - - -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -   38 Line(s) not Displayed
 001235 >Msg S:Table is not open               TLDMMID=ISPT034
001236 >Msg L:TBQUERY issued for table WFSPLP that is not open.
- - -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  - 162 Line(s) not Displayed
001399 >ISPCAT   ISPEXEC TBOpen WFSplp nowrite library(PLPISPF)
001400 >ISPEXEC called from IRXSTAM+A02       05198D6A in PLPA  EPA:05198368
- - -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -   38 Line(s) not Displayed
001439 >ENQ CNQ(TLD1,'SPFEDIT ',
```

ISPVCALL – Options ()

ISPVCALL [options]		
Keyword	**Eye Catcher**	**Description**
SVC26 or SVC27	>SVC26 >SVC27	Adds unformatted SVC 26 (LOCATE) and SVC 27 (OBTAIN) data to the trace. For example, would show how option 3.4 builds its list
DAIR	>DAIR	Adds formatted information about ISPF allocations for data sets, such as the list and log data sets, work data sets and edit recovery data sets. These allocations use the TSO Dynamic Allocation Interface Routine
VDEFINE	>VDEF	Shows VDEFINE and VDELETE calls for every variable
MONITOR VAR(abc)	>VARS	Shows a hexadecimal and EBCDIC dump of variable abc every time it changes. Unfortunately, the initial value of the variable isn't shown
PARM abc	>PARM	Dumps parameters passed to an ISPF module. For example, to determine what module processes a particular service in order to check the service parameters. Optionally, takes an L(x) parameter to define the number of parameters to print
STORAGE	>CS	Traces ISPF's GETMAIN and FREEMAIN calls

ISPVCALL – Options ()

ISPVCALL [options]		
Keyword	**Eye Catcher**	**Description**
HISTORY		Keeps 50,000 records of trace in storage and doesn't write the trace until ISPVCALL is invoked again. Used to catch intermittent problems. Since ISPVCALL's overhead is so small, the HISTORY keyword may run enabled for long periods with no performance degradation
STATUS		Shows system information; but, doesn't start the trace
VIEW		ISPVCALL must not be active when the VIEW keyword is specified
END		Ends a running trace without it in the editor

Links

- Panel and Skeleton REXX
  - SHARE, Session 11708, August 2012, Peter Van Dyke
    - ISPF – Panels Advanced
  - Chapter 7, Panel definition statement reference and Chapter 10, Defining file-tailoring skeletons
    - ISPF – V2R1 – Dialog Developer's Guide and Reference
- ISPDPTRC and ISPFTTRC
  - Appendix C, Diagnostic Tools and Information
    - ISPF – V2R1 – Dialog Developer's Guide and Reference
  - Chapter 7, Diagnostic Tools and Information
    - ISPF – V2R1 – Messages and Codes
- ISPVCALL
  - SHARE, Session 14767, Spring 2014, Peter Van Dyke
    - ISPF – Behind the Scenes
  - MainframeZone.com, 1 Oct 04, Doug Nadel
    - ISPVCALL – ISPF Debugging on the Bleeding Edge

Common Problems

Common Problems

- Uninitialized variables
- Not placing quotes around literals or external commands
- Missing DO or END clauses
- Updated panels and messages not picked up
  - Use ISPF 7.2, Dialog Test Panels, to reload modules into memory
  - Or invoke ISPF in TEST mode
- Incorrect environment when executing external commands
  - Resulting in RC = -3
  - Code ADDRESS command

Common Problems

- Missing modules (panels, etc.)
  - When invoking application
    - Add PASSLIB to SELECT statement
  - After exiting application
    - Add STACK to LIBDEF statements
    - And/or make STACK the ISPF default using TSO ISPCCONF

      DEFAULT_LIBDEF_PROCESSING_OPTION = STACK

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

Manuals – ISPF

Version	Manual	Link
V1R13	SC34-4820-10	ISPF – Edit and Edit Macros
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 – 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	SDSF – Implementing REXX Support
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

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

SHARE – ISPF

Session	Author	Link
16630	Sam Reynolds	ISPF Hidden Treasures and New Features
16628	Liam Doherty	ISPF Editor – Beyond The Basics Hands-on Lab
16607	Bruce Koss	ISPF – Workstation Agent (WSA)
15805	Bruce Koss	ISPF – An Experienced User Shares His Secrets
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
16185	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 Migrationand 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

SHARE – SDSF

Session	Author	Link
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!
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
1569	Ahilan Rajadeva	Rexx and z/OS UNIX Services

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

Websites (📎)

Link
Parallel Sysplex – Tools and Wizards
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

Websites (📎)

Link
Watson & Walker – Tuning Letter
WLM – Downloads
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

#SHAREorg

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