

**Lab 17314**

**IBM PD Tools Hands-On Lab: Dive into Increased  
Programmer Productivity**

**IBM Fault Analyzer for z/OS**

**Eclipse interface**

**Hands-on Lab Exercises**

## IBM Fault Analyzer for z/OS V13

### Lab Exercises

© Copyright International Business Machines Corporation, 2015. All rights reserved.

This document is intended for IBM internal use only. The contents of the document may be discussed with existing and/or potential customers. The document itself, however, may not be distributed outside of IBM.

# Contents

---

	<b><u>Page</u></b>
<b>Overview</b>	2
<b>Lab 1:</b> Getting started with the Fault Analyzer perspective	3
<b>Lab 2:</b> Displaying a list of abends in a Fault History File, and displaying an abend report	7
<b>Lab 3:</b> Researching an example abend report	10
<b>Lab 4:</b> Working with the list of fault entries	23

## Overview

### *Fault Analyzer*

IBM® Fault Analyzer for z/OS® is a robust problem determination tool that helps you discover why an application abends. It can report information in terms of application code, which means that developers and maintainers are not forced to interpret a low level system dump or system-level error messages. As a result, the reason for the abend can be determined sooner and with less effort.

When an application abend occurs, Fault Analyzer captures details about the failure and stores the information into a Fault History file. It analyzes the information about the application and its environment in real time, then generates an analysis report detailing the cause of the failure.

Some of the main features of Fault Analyzer are:

- Automatic collection and analysis of application abend information
- Source-level information (abending statement, variable values) for many languages including Enterprise COBOL, Enterprise PL/I, assembler and others.
- Interfaces from either TSO or eclipse

### *This workbook*

This workbook contains instructions for lab exercises that are designed to give you hands-on experience for the eclipse interface of IBM Fault Analyzer for z/OS.

### *Reference*

Product manuals and other information about IBM Fault Analyzer for z/OS, and other IBM problem determination tools, are available on the Web at URL:

<http://www.ibm.com/software/awdtools/deployment>

# Lab Exercise 1

## Getting started with the Fault Analyzer perspective

In this exercise you will:

- Open the Fault Analyzer perspective in the Eclipse workbench.
- Learn how to display help information for Fault Analyzer.

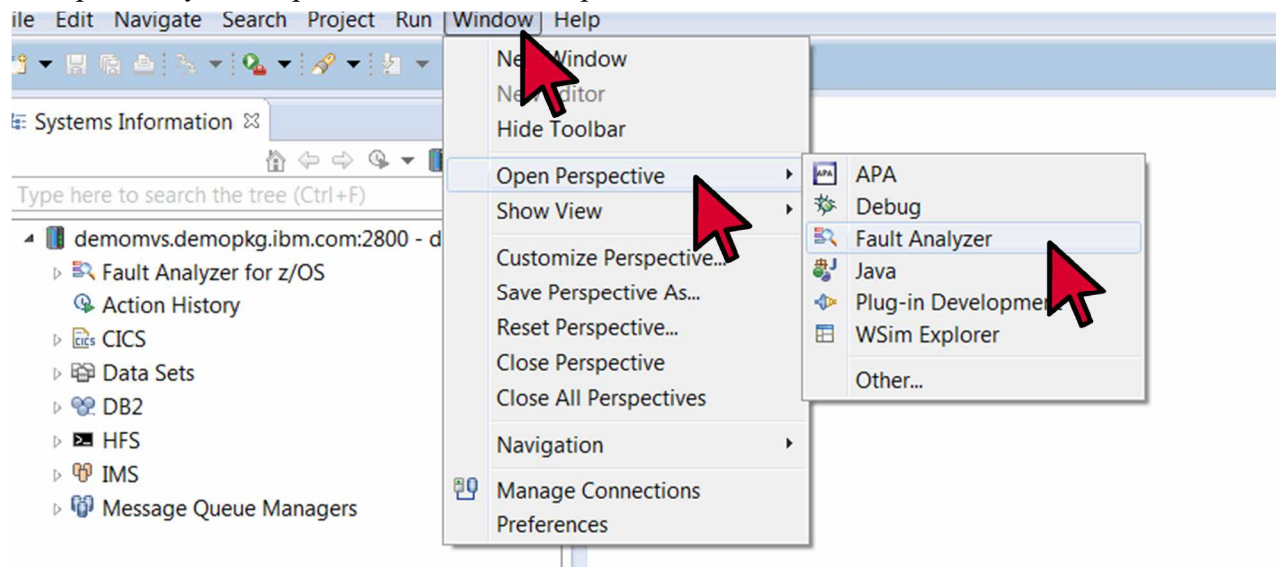
1. Before you begin, you must have the Eclipse interface open on your workstation. If you aren't sure how to open it, please contact your instructor.

In eclipse, a perspective is a set of views (windows), menus, and options that provide a set of functions. The File Manager interface is a perspective. Before you can use File Manager, you need to open it's perspective.

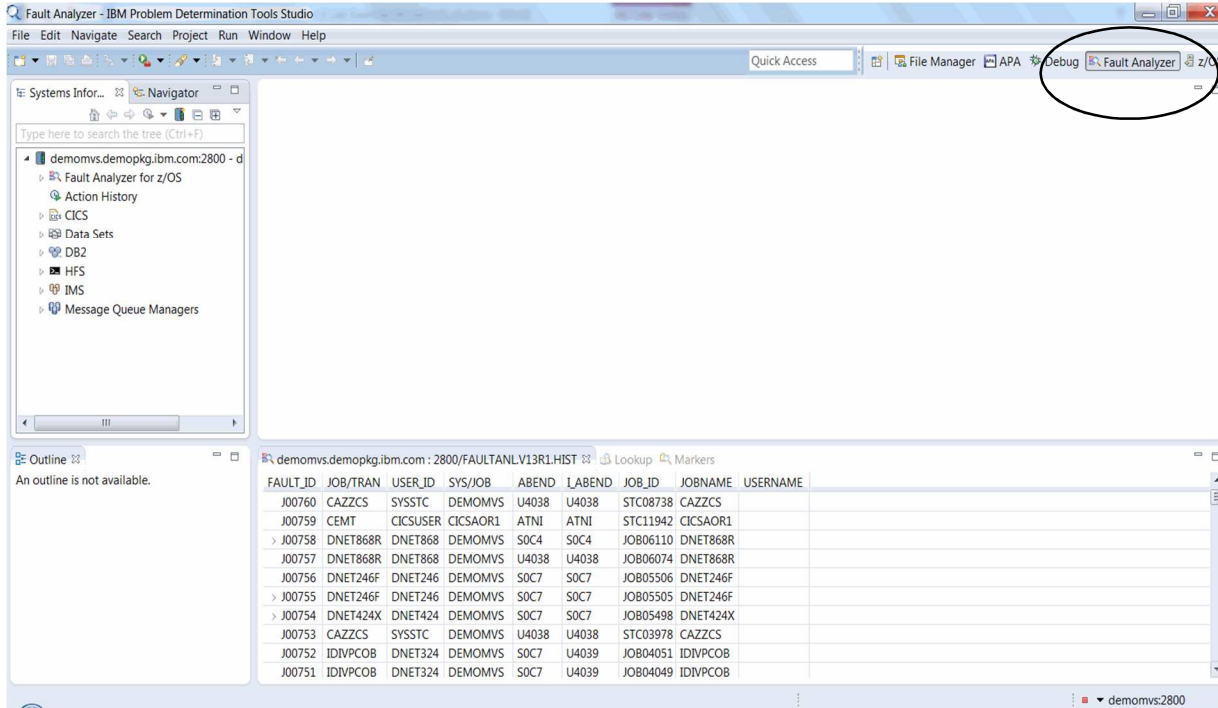
2. Opening the Fault Analyzer perspective.

- a. From the menu bar near the top of the eclipse workbench, select **Window** > **Open Perspective** > **Fault Analyzer**.

- i. Note: If "Fault Analyzer" is not shown as a selection, then the Fault Analyzer perspective is already open and you can proceed to the next step.



3. The Fault Analyzer perspective is displayed. Note: the contents of the Systems Information view may be different or even blank on your system.

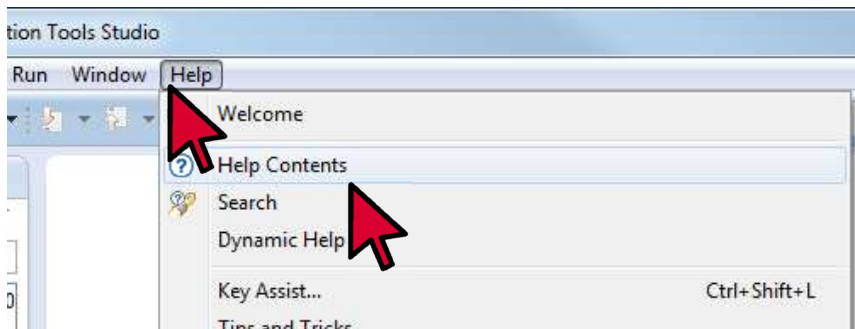


4. Notice that there are several views (windows) in the Fault Analyzer perspective. By default, the Systems Information view is displayed in the upper left. In the next exercise, you will use this view to work with abends that Fault Analyzer has collected on a z/OS system. From the Systems Information view, you can initiate actions including:

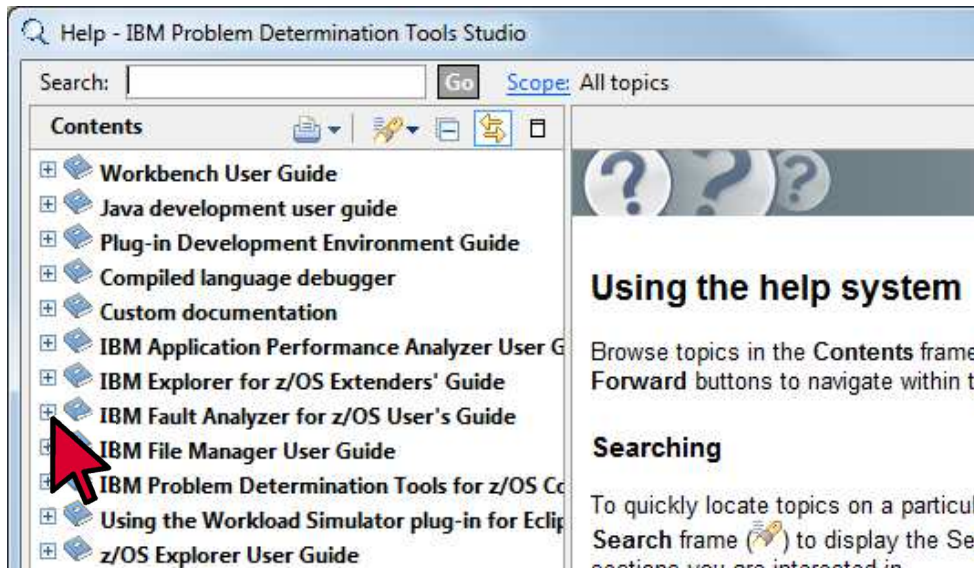
- Select a z/OS system to be accessed
- Specify the name of a fault history file
- Open a fault history file
- and other actions

5. You can display help information to assist with various features of Fault Analyzer.

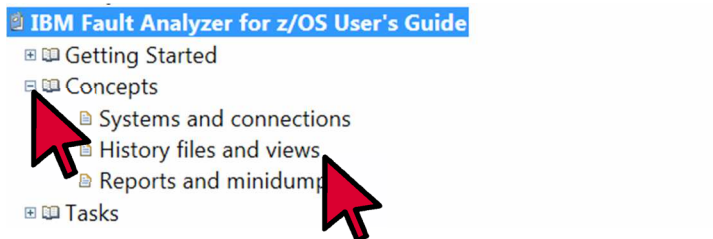
- a. To open the help dialog, select **Help > Help Contents**.



- b. The Help dialog is displayed. To expand the Fault Analyzer help topics, click the plus (  $\pm$  ) next to IBM Fault Analyzer for z/OS User's Guide.



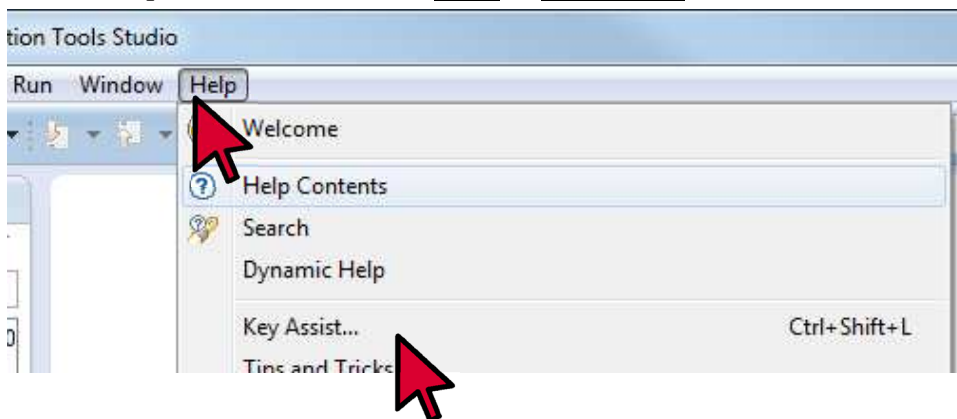
- c. Under **IBM Fault Analyzer for z/OS User's Guide**, select **Concepts** > **History Files and Views**.



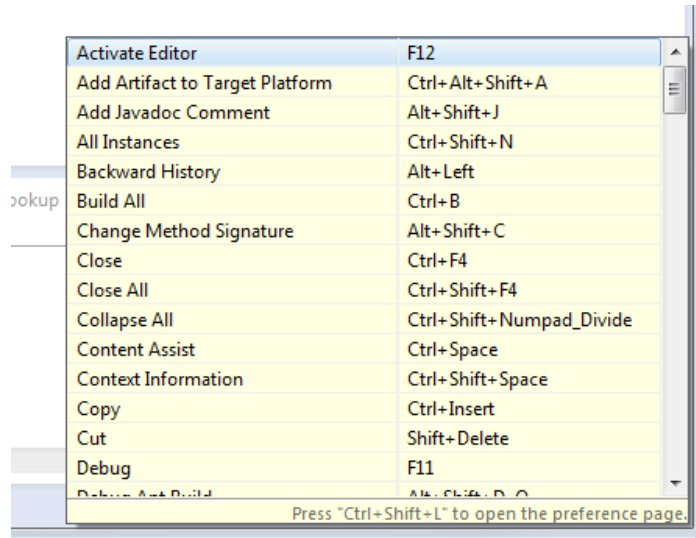
- d. The selected topic is displayed.  
 e. Now you have seen how to display general help information.  
 f. Close the help dialog: click the **X** (close) icon in the upper right corner of the help window.

6. You can also display a list of keyboard commands.

- a. In the eclipse workbench Select **Help** > **Key Assist**.



b. A list of keyboard commands is displayed.



7. Note: As you work through the exercises, you may accidentally close or hide a view (a window) or other parts of the Fault Analyzer perspective. Remember that if that happens, you can easily reset the perspective to its default views. Try it now, so you know how to perform the reset:
- From the menu, select **Window > Reset perspective**.
  - The perspective is reset. The default views are displayed in their default positions and with their default sizes.
8. Now that the Fault Analyzer perspective is open, and you know where to find help, you are ready to start using Fault Analyzer in the following exercises.



# Lab Exercise 2

## Displaying a list of abends in a Fault History File, and displaying an abend report

In this exercise you will:

- Determine if a fault history file has been added, and add a new fault history file if required.
- Become familiar with working with fault history files and fault entries.

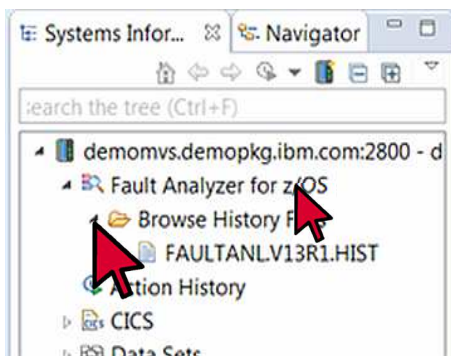
### Introduction to Fault History Files

When a running program abends on a z/OS system, Fault Analyzer can capture information about the abend. The captured information is stored in a special file called ‘fault history’ file. Information for one abend is called a ‘fault entry’. A fault history file can hold more than fault entry.

A system can have one or more fault history files. The name or names of the history files are controlled by the system administrator of Fault Analyzer. If there are multiple fault history file on a system, the administrator configures Fault Analyzer to select the correct fault history file to capture each fault entry. Characteristics such as the job name, user id, job or transaction type, program name and other things can be used to control which fault history file is used to collect each abend.

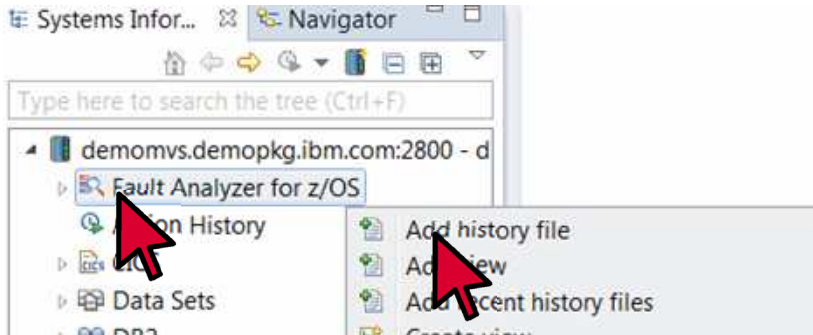
For this exercise, you were provided the name of a history file that you will use. If you are not sure about the name of the history file to use in this exercise, please ask for the name of the file.

1. In the Systems Information Tab, under the z/OS system that you will use during this exercise, select **Fault Analyzer for z/OS > Browse History files**

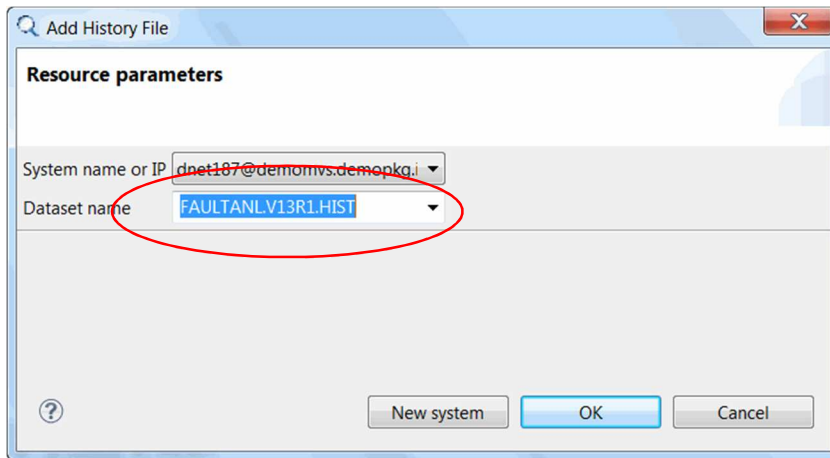


2. See if the history file name that you will use for this exercise is already shown. If it is then skip down to step 4 (you do not need to do step 3).

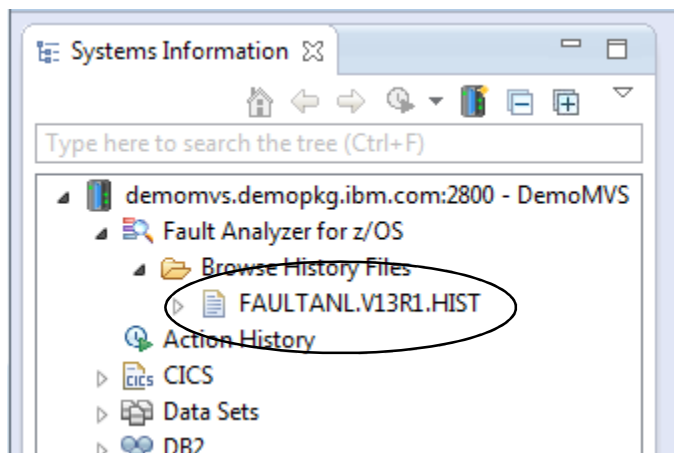
3. You can add a definition for a fault history file.
- Right-click **Fault Analyzer for z/OS**, and select **Add History File**



- Enter the full history file name without quotation marks. Use the history file name that was provided to you for this exercise. DO NOT use the name of the history file shown in this example, unless it is the same as the name provided.

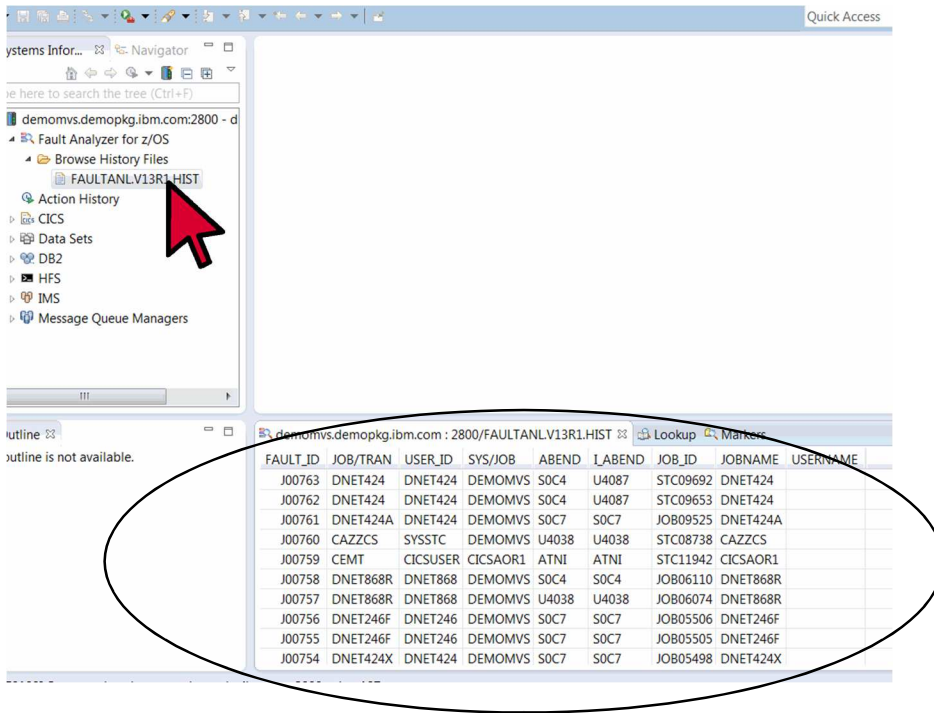


- The history file is displayed in the list. After a history file is in the list, you are ready to work with it. Proceed to the next step.



4. Working with a history file.

- a. To open a history file, **Double-click** its name.
- b. The list of fault entries in the history file is displayed in a separate view.



Tip: If Fault Analyzer has captured new abends since you opened a fault history file, you can refresh it. In the Systems Information view, **right-click** the Fault History file and select **Reload history file from host**.

5. You have completed this exercise.

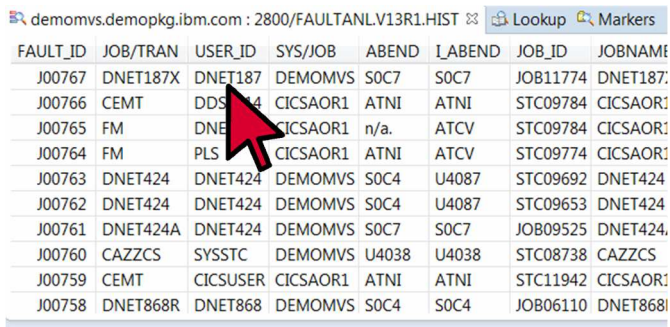
# Lab Exercise 3

## Researching an example abend report

In this exercise you will:

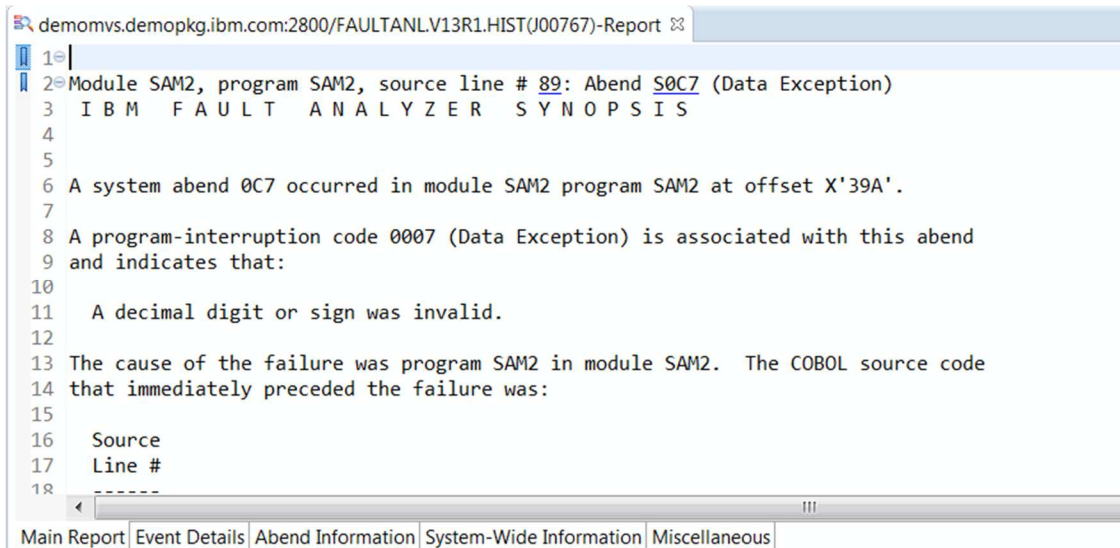
- Learn to navigate an analysis report, and understand what information presented is presented in the various sections of an a report
- Research an example abend to determine why the application abended

1. If the fault history file is already open, and a list of fault entries is displayed, continue with this exercise. If a list of fault entries is not displayed, complete exercise 2 (to open a history file) before continuing.
2. You can open a fault entry from the list.
  - a. In the list of abends, **double-click** the example abend. You were provided the fault id of the sample fault entry that you will use in this exercise. If you are not sure which one to use, please ask.



FAULT_ID	JOB/TRAN	USER_ID	SYS/JOB	ABEND	L_ABEND	JOB_ID	JOBNAME
J00767	DNET187X	DNET187	DEMOMVS	S0C7	S0C7	JOB11774	DNET187:
J00766	CEMT	DDSM14	CICSAOR1	ATNI	ATNI	STC09784	CICSAOR1
J00765	FM	DNET187	CICSAOR1	n/a.	ATCV	STC09784	CICSAOR1
J00764	FM	PLS	CICSAOR1	ATNI	ATCV	STC09774	CICSAOR1
J00763	DNET424	DNET424	DEMOMVS	S0C4	U4087	STC09692	DNET424
J00762	DNET424	DNET424	DEMOMVS	S0C4	U4087	STC09653	DNET424
J00761	DNET424A	DNET424	DEMOMVS	S0C7	S0C7	JOB09525	DNET424:
J00760	CAZZCS	SYSSTC	DEMOMVS	U4038	U4038	STC08738	CAZZCS
J00759	CEMT	CICSUSER	CICSAOR1	ATNI	ATNI	STC11942	CICSAOR1
J00758	DNET868R	DNET868	DEMOMVS	S0C4	S0C4	JOB06110	DNET868:

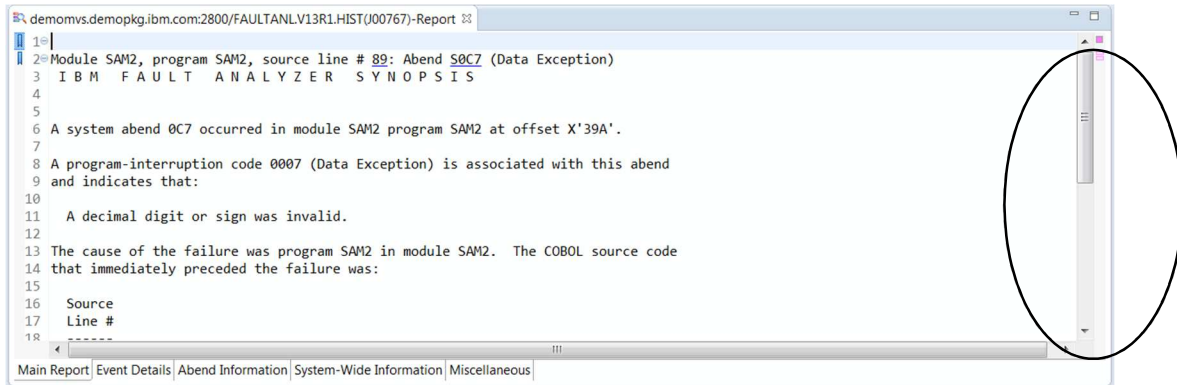
- b. The abend report is displayed



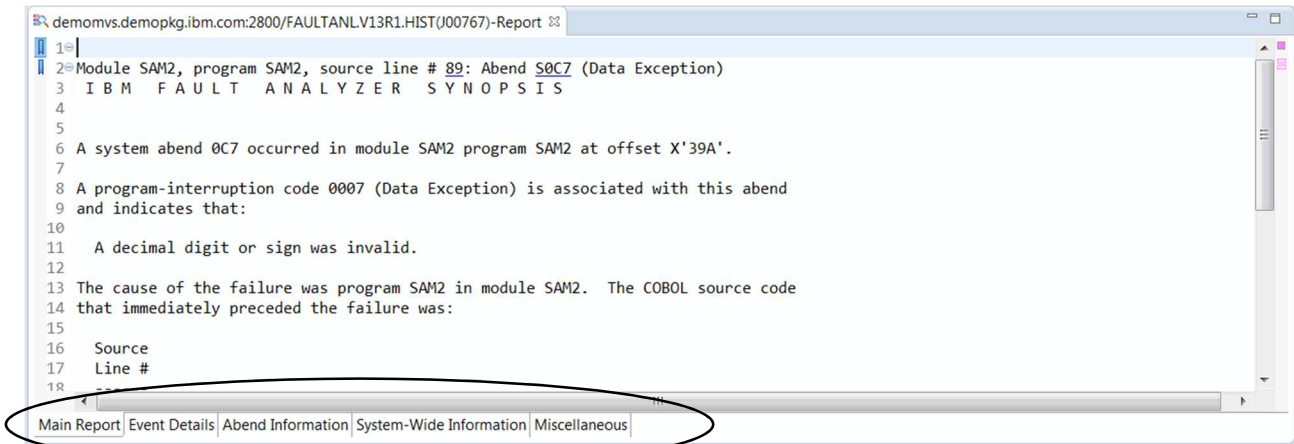
```
demomvs.demopkg.ibm.com:2800/FAULTANLV13R1.HIST(J00767)-Report
1 |
2 | Module SAM2, program SAM2, source line # 89: Abend S0C7 (Data Exception)
3 | I B M   F A U L T   A N A L Y Z E R   S Y N O P S I S
4 |
5 |
6 | A system abend 0C7 occurred in module SAM2 program SAM2 at offset X'39A'.
7 |
8 | A program-interruption code 0007 (Data Exception) is associated with this abend
9 | and indicates that:
10 |
11 |   A decimal digit or sign was invalid.
12 |
13 | The cause of the failure was program SAM2 in module SAM2. The COBOL source code
14 | that immediately preceded the failure was:
15 |
16 |   Source
17 |   Line #
18 |   -----
19 |
```

Main Report | Event Details | Abend Information | System-Wide Information | Miscellaneous

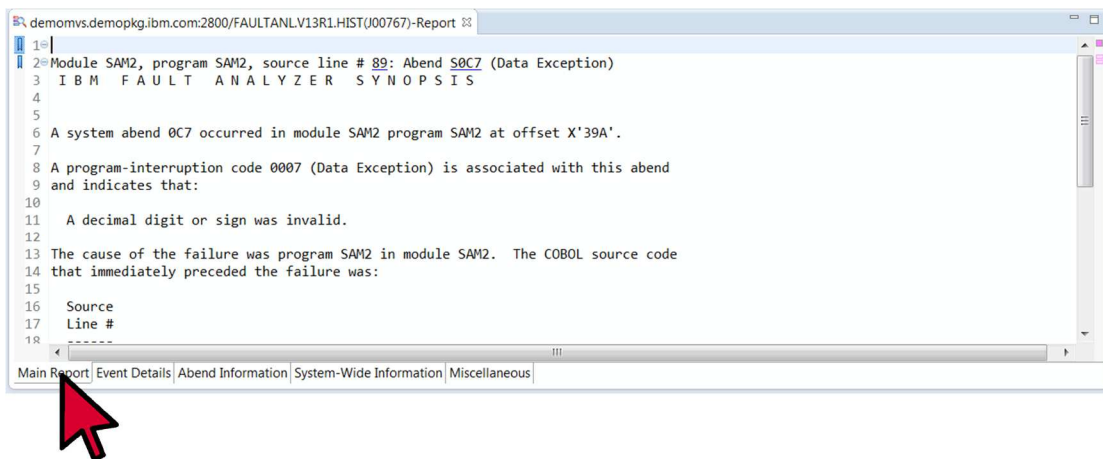
3. Next you will get practice navigating the sections of the abend report.
  - a. Notice and use the scroll bar on the right to scroll the report up and down.



- b. Notice the tabs at the bottom of the report. Each tab can be clicked to display a different part of the abend report.



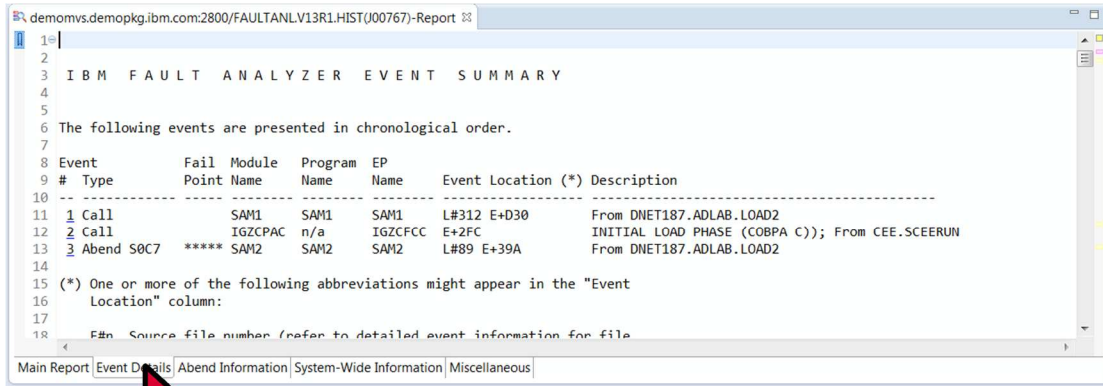
- c. Click the **Main Report** tab.



- The main report shows a synopsis of the abend. It contains information including the abend code and the name of the program that was running when the abend occurred. If source mapping information was available, Fault Analyzer reports the specific statement in the program that was

executing when the abend occurred. Scroll down to see the abending statement and the fields referenced by the abending statement and their values.

d. Click the **Event Details** tab.



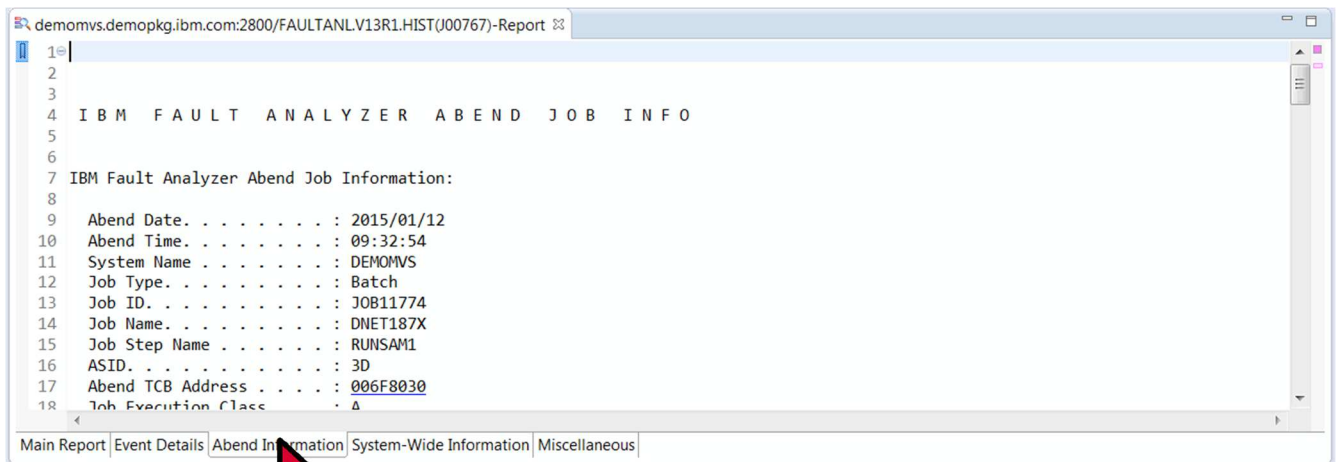
The screenshot shows the 'Event Summary' tab of the IBM Fault Analyzer report. The title bar reads 'demomvs.demopkg.ibm.com:2800/FAULTANLV13R1.HIST(J00767)-Report'. The report content includes a table of events:

Event #	Type	Fail Point	Module Name	Program Name	EP Name	Event Location (*)	Description
11	1	Call	SAM1	SAM1	SAM1	L#312 E+D30	From DNET187.ADLAB.LOAD2
12	2	Call	IGZCPAC	n/a	IGZCFCC	E+2FC	INITIAL LOAD PHASE (COBPA C)); From CEE.SCEERUN
13	3	Abend S0C7	***** SAM2	SAM2	SAM2	L#89 E+39A	From DNET187.ADLAB.LOAD2

Below the table, there is a note: '(\*) One or more of the following abbreviations might appear in the "Event Location" column: E#n, Source file number (refer to detailed event information for file

- The top part of Event Detail report is the Event Summary, which shows the programs in the call chain. You can scroll down to see detailed reports for each program in the call chain.

e. Click the **Abend Information** tab

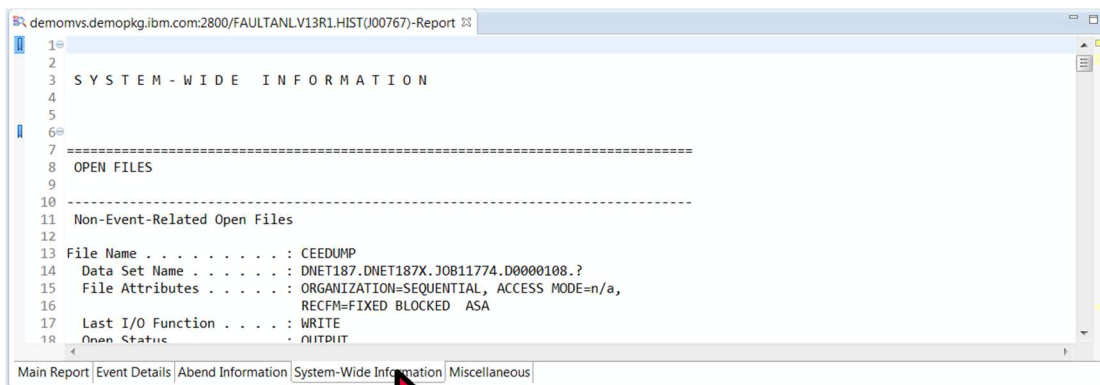


The screenshot shows the 'Abend Information' tab of the IBM Fault Analyzer report. The title bar reads 'demomvs.demopkg.ibm.com:2800/FAULTANLV13R1.HIST(J00767)-Report'. The report content includes the following information:

```
IBM Fault Analyzer Abend Job Information:
Abend Date. . . . . : 2015/01/12
Abend Time. . . . . : 09:32:54
System Name . . . . . : DEMOMVS
Job Type. . . . . : Batch
Job ID. . . . . : JOB11774
Job Name. . . . . : DNET187X
Job Step Name . . . . . : RUNSAM1
ASID. . . . . : 3D
Abend TCB Address . . . . . : 006F8030
Job Execution Class . . . . . : A
```

- The Abend Information report tab displays general information about the abending job and programs.

f. Click the **System-Wide Information** tab



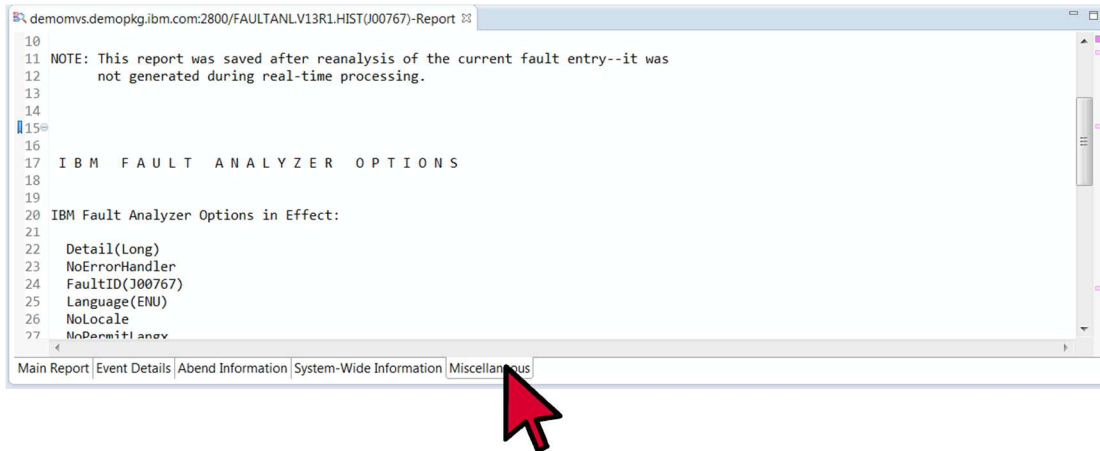
The screenshot shows the 'System-Wide Information' tab of the IBM Fault Analyzer report. The title bar reads 'demomvs.demopkg.ibm.com:2800/FAULTANLV13R1.HIST(J00767)-Report'. The report content includes the following information:

```
SYSTEM - W I D E I N F O R M A T I O N
=====
OPEN FILES
-----
Non-Event-Related Open Files
File Name . . . . . : CEEDUMP
Data Set Name . . . . . : DNET187.DNET187X.JOB11774.D0000108. ?
File Attributes . . . . . : ORGANIZATION=SEQUENTIAL, ACCESS MODE=n/a,
RECFM=FIXED BLOCKED ASA
Last I/O Function . . . . . : WRITE
Open Status . . . . . : OUTPUT
```



- The System-Wide Information Tab displays information captured about the abend that is not associated with a specific program.

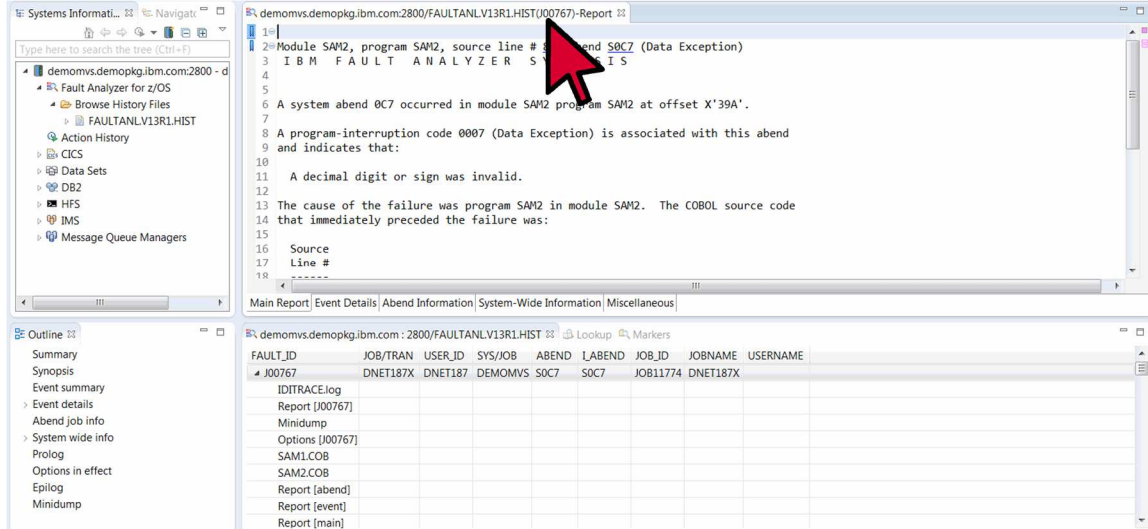
g. Click the **Miscellaneous** tab



- The Miscellaneous Tab displays information about Fault Analyzer options and datasets used by Fault Analyzer as it analyzed and stored the report.

4. A view (a window in the workbench) can be maximized to full size to provide a larger area.

- **Double-click** the tab of the abend report view to expand it to the full size of the workbench.



- Double-click the tab again, and the view is restored to its previous size.
- Remember that you can maximize a view in the workbench if it will be helpful to make it larger.

5. Note: In the following steps, the example abend will be analyzed to determine the reason why the abend occurred.

6. You can determine high-level information about an abend. For example, did the abend occur in a batch job, a CICS transaction, or some other environment? When did it occur? What program abended? If you are not familiar with an application, it can be helpful to use the reports to get information that can help you understand the application and the environment.

- Click the **Abend Information** tab

```

1= |
2 |
3 |
4 | I B M   F A U L T   A N A L Y Z E R   A B E N D   J O B   I N F O
5 |
6 |
7 | IBM Fault Analyzer Abend Job Information:
8 |
9 |   Abend Date. . . . . : 2015/01/12
10 |  Abend Time. . . . . : 09:32:54
11 |  System Name . . . . . : DEMOMVS
12 |  Job Type. . . . . : Batch
13 |  Job ID. . . . . : J0811774
14 |  Job Name. . . . . : DNET187X
15 |  Job Step Name . . . . . : RUNSAM1
16 |  ASID. . . . . : 3D
17 |  Abend TCB Address . . . . . : 006F8030
18 |  Job Execution Class . . . . . : A
19 |  Region Size . . . . . : 4M
20 |  EXEC Program Name . . . . . : SAM1
21 |  User ID . . . . . : DNET187
22 |  Accounting Information. . . : ZACK,81038
23 |
24 | Data Sets:
25 |
26 |   DDname   Data Set or Path Name
27 |   -----
28 |   STEPLIB  DNET187.ADLAB.LOAD2
29 |

```

Main Report | Event Details | **Abend Information** | System-Wide Information | Miscellaneous

- This report shows that the application was running as a batch job. Notice that the job name, the date and time, the name of the system where the job was running, and other high level information is displayed. Scroll down to see the rest of the report.
- For the purposes of analyzing this sample application, all that you need to notice here is that it is a batch job, and the name of the main program (the ‘EXEC Program name’) is SAM1.

7. Click the Main Report tab.

```

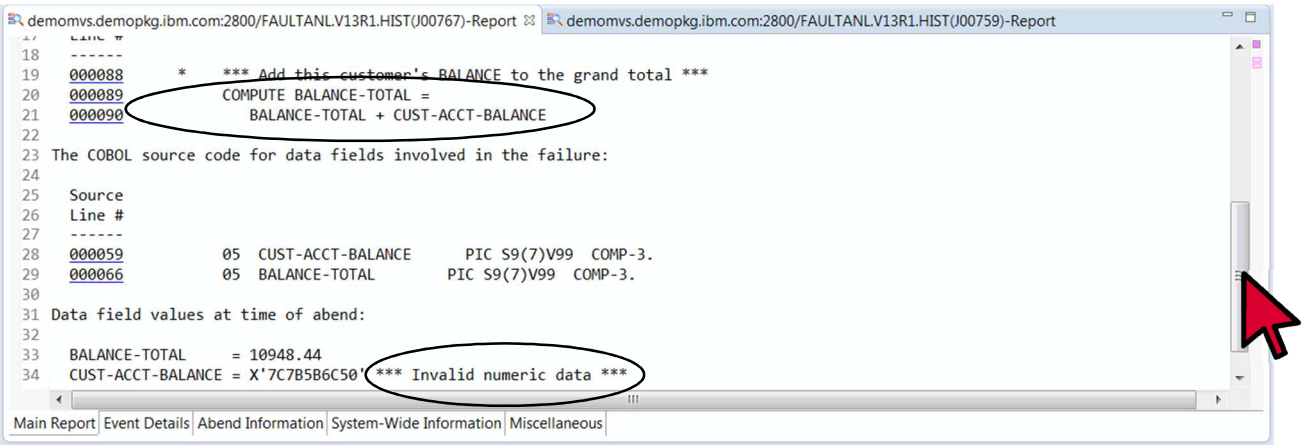
1= |
2= | Module SAM2, program SAM2, source line # 89: Abend S0C7 (Data Exception)
3 | I B M   F A U L T   A N A L Y Z E R   S Y N O P S I S
4 |
5 |
6 | A system abend 0C7 occurred in module SAM2 program SAM2 at offset X'39A'.
7 |
8 | A program-interruption code 0007 (Data Exception) is associated with this abend
9 | and indicates that:
10 |
11 | A decimal digit or sign was invalid.
12 |
13 | The cause of the failure was program SAM2 in module SAM2. The COBOL source code
14 | that immediately preceded the failure was:
15 |
16 | Source
17 | Line #
18 | -----

```

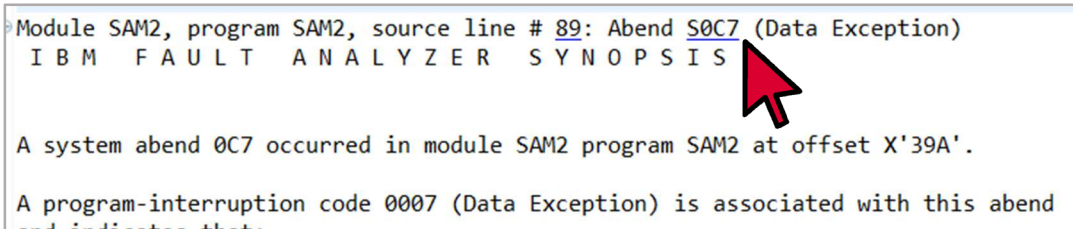
Main Report | Event Details | **Abend Information** | System-Wide Information | Miscellaneous

- In this example the job failed with a S0C7 abend code, while running in program SAM2 at source line 89. **Scroll down** to display the abending statement, variables and variable contents.

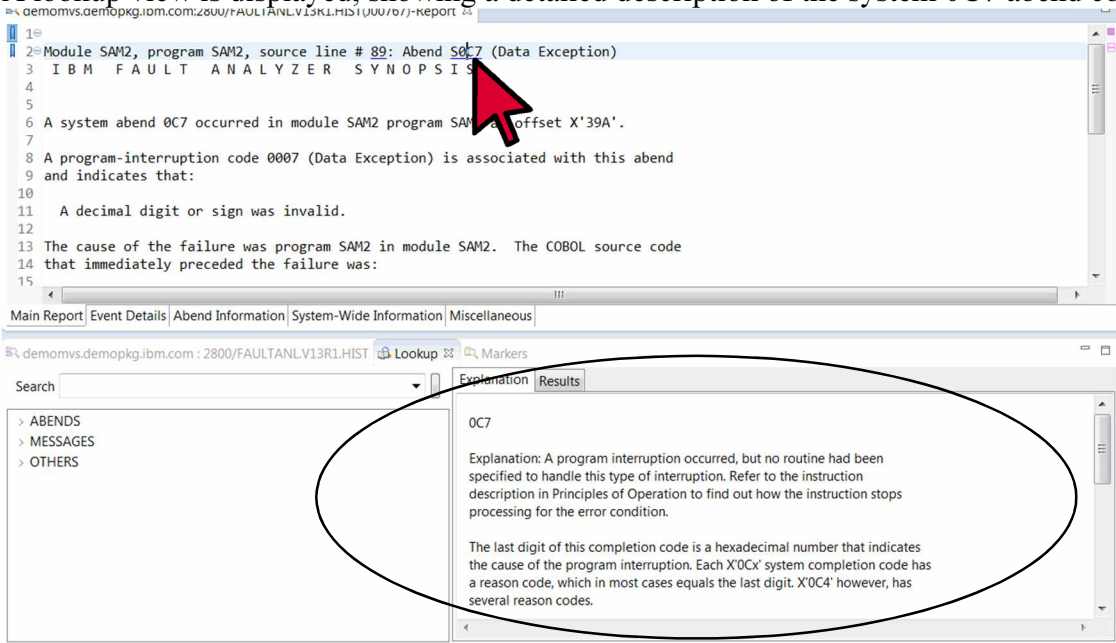




- b. Notice that a brief explanation is given for 'S0C7' abend code. The report describes it as 'A decimal digit or sign was invalid.' The report shows that the program was attempting to execute a COMPUTE statement when it failed. It also shows that variable CUST-ACCT-BALANCE contained invalid numeric data.
- You can determine from the report that the abend occurred because variable CUST-ACCT-BALANCE had invalid data, and the program attempted to use it to perform a computation.
- c. Notice that certain text areas are underlined. These are links, and clicking them will display related information. For example, the '89' and 'S0C7' near the top of the report are links.
- Click the S0C7 link.



- A lookup view is displayed, showing a detailed description of the system 0C7 abend code.



- **Click the 89 link.**
- The source view is displayed, positioned at line 89, which is the abending statement.

```

-----1-----2-----3-----4-----5-----6-----7-----8-----9-----10-----11-----12-----13-----
ADD +1 TO BALANCE-COUNT
* *** Add this customer's BALANCE to the grand total ***
COMPUTE BALANCE-TOTAL = BALANCE-TOTAL + CUST-ACCT-BALANCE
* *** Calculate Average ***
COMPUTE BALANCE-AVERAGE = BALANCE-TOTAL / BALANCE-COUNT
* *** Calculate Minimum ***
IF WS-FIRST-TIME-SW = 'Y'
  MOVE CUST-ACCT-BALANCE TO BALANCE-MIN.
IF CUST-ACCT-BALANCE < BALANCE-MIN
  MOVE CUST-ACCT-BALANCE TO BALANCE-MIN.
* *** Calculate Maximum ***
* *** There is a bug calculating the maximum. ***
* *** Can you find it? ***
IF WS-FIRST-TIME-SW = 'Y'
  MOVE CUST-ACCT-BALANCE TO BALANCE-MAX
IF CUST-ACCT-BALANCE > BALANCE-MAX
  MOVE CUST-ACCT-BALANCE TO BALANCE-MAX.
* *** CALCULATE RANGE ***
COMPUTE BALANCE-RANGE = BALANCE-MAX - BALANCE-MIN.
500-INIT-STATISTICS.
MOVE 'ZEROING STATS VARIABLES' TO WS-PROGRAM-STATUS.

```

8. By default, the source view opens in the same pane as the report. Return to the report by clicking its tab.
9. Next, you will continue to research the abend by reviewing information about the call chain and details about the programs.
  - a. Click the **Event Details** tab.

```

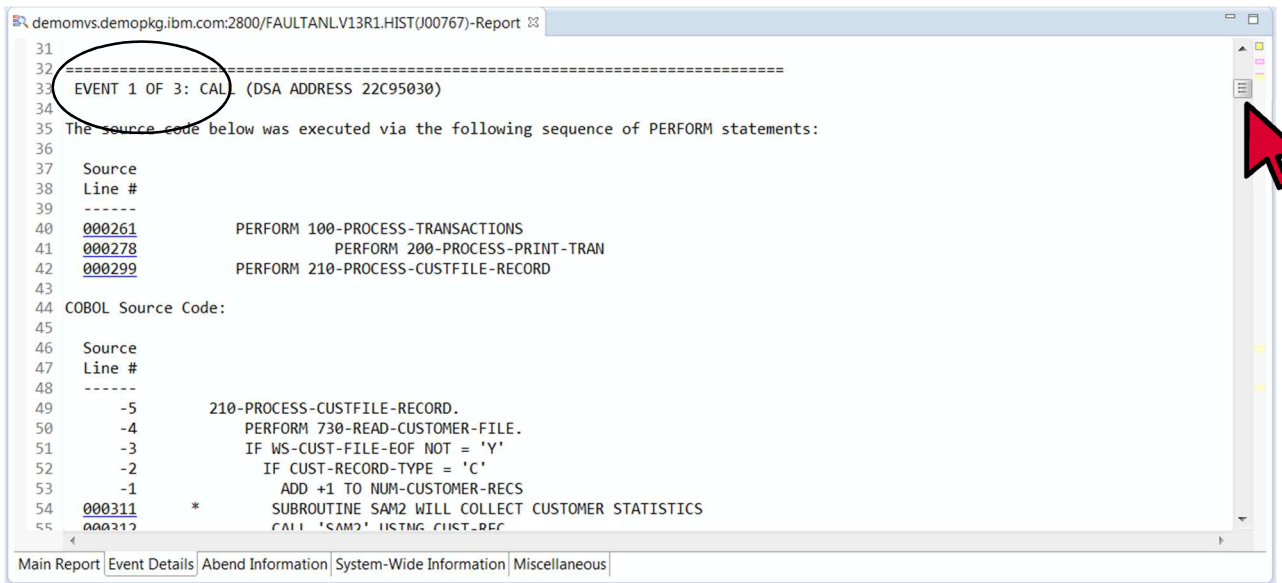
demomvs.demopkg.ibm.com:2800/FAULTANLV13R1.HIST(J00767)-Report
1=|
2
3 IBM FAULT ANALYZER EVENT SUMMARY
4
5
6 The following events are presented in chronological order.
7
8 Event      Fail Module  Program  EP
9 #  Type    Point Name  Name     Name     Event Location (*) Description
10 -----
11 1 Call      SAM1      SAM1     SAM1     L#312 E+D30   From DNET187.ADLAB.LOAD2
12 2 Call      IGZCPAC   n/a      IGZCFCC E+2FC       INITIAL LOAD PHASE (COBPA C)); From CEE.SCEERUN
13 3 Abend S0C7 ***** SAM2     SAM2     SAM2     L#89 E+39A    From DNET187.ADLAB.LOAD2
14
15 (*) One or more of the following abbreviations might appear in the "Event
16 Location" column:
17
18 F#n Source file number (refer to detailed event information for file
19 identification)
20 L#n Source file line number
21 S#n Listing file statement number (refer to detailed event information for
22 file identification)
23 M+x Offset from start of load module
24 P+x Offset from start of program
25 E+y Offset from start of entry point

```

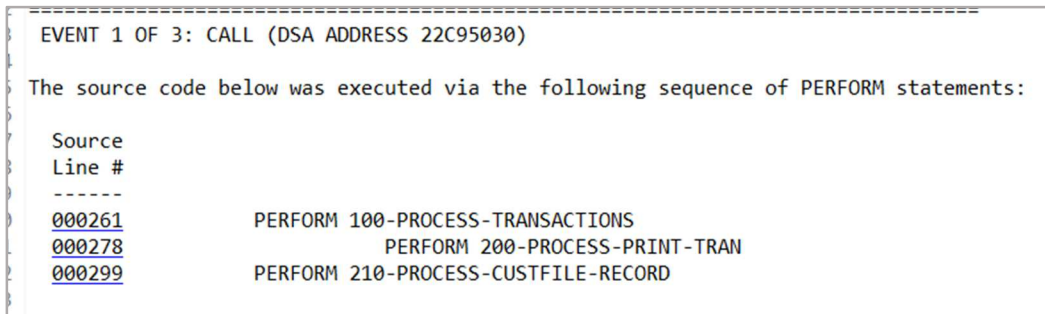
- b. The event summary is displayed near the top. It shows the call chain, which is the list of programs that were active at the time of the abend. The first program in the list is the main program, the second is the program called by the main program, and so on. This report shows three events in the call chain. Program SAM1 is the main program. It called a system routine called IGZCPAC, which in turn called a program called SAM2. The abend occurred in SAM2.
10. Note: The event details section also provides detail reports for each program in the event summary (the call chain). There are two ways to scroll to the detail report for a program. One way is to simply scroll down

using the scroll bar on the right. The other way is to click one of the links in the 'Event #' column (numbered 1, 2 and 3 in this example) of the event summary list.

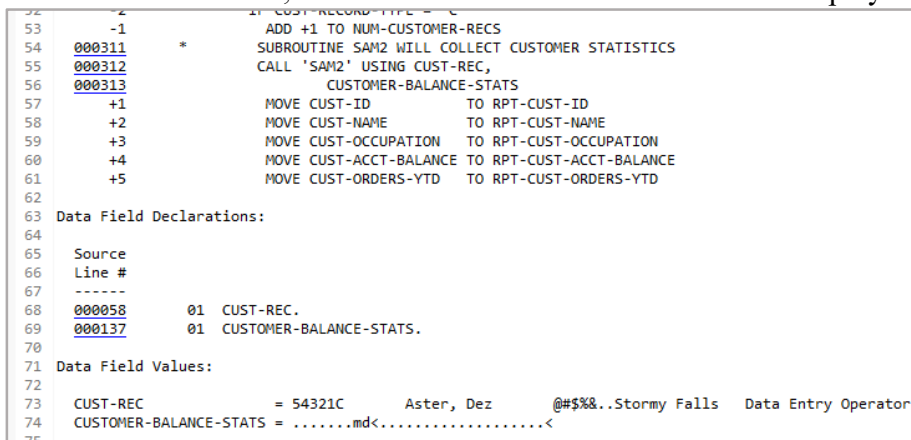
11. **Scroll down** to the detail report for event 1. This is the detail report for the first program in the call chain, SAM1.



- a. Additional information is displayed about the program as you scroll down
  - i. The paragraph trace shows the paragraphs that were executed to get to the current statement. Notice that the line numbers are links.



- ii. The current statement, and the variables that it references are displayed.



- iii. Information about the module is shown, including the name of the load library where the program module resides, the compile date and time, and other details.

```

78 The SYSDEBUG file used for the above was found in DNET187.ADLAB.SYSDEBUG(SAM1).
79
80 Load Module Name . . . . . : DNET187.ADLAB.LOAD2(SAM1)
81 At Address . . . . . : 00007000
82 Load Module Length . . . . . : X'3378'
83 Link-Edit Date and Time . . : 2009/10/06 22:19:22
84
85 Program and Entry Point Name: SAM1
86 At Address . . . . . : 00007000 (Module SAM1
87 offset X'0')
88 Program Length . . . . . : X'2734'
89 Program Language . . . . . : COBOL (Compiled using IBM Enterprise COBOL for
90 z/OS and OS/390 V3 R4 M1 on 2009/10/06 at
91 22:19:21)
92 Compiler Options Used . . . : ADV QUOTE ARITH(COMPAT) NOAWO NOCURRENCY DATA(31)
93 NODATEPROC DBCS NODECK NODLL NODUMP DYNAM
94 NOEXPORTALL NOFASTSRT INTDATE(ANSI) LIB LIST MAP

```

- iv. The machine instruction that was executing when the abend occurred is shown with its address, followed by the values of the general purpose registers.

```

01 Machine Instruction . . . . : 05EF BALR R14,R15
02 At Address . . . . . : 00007D30 (Program
03 SAM1 offset X'D30')
04 AMODE . . . . . : 31
05
06 General Purpose Registers:
07 R0: 22C95278 (589192 bytes of storage addressable)
08 R1: 22C951A8 (589400 bytes of storage addressable)
09 R2: 0000B7FC (235524 bytes of storage addressable)
10 R3: 00024F88 (Module SAM1 program SAM1 FILE SECTION BLF=0000 + X'0', symbol CUST-REC, source line #58)
11 R4: 00008A0C (Module SAM1 program SAM1 + X'1A0C', source line #465)
12 R5: 22C951A0 (589408 bytes of storage addressable)
13 R6: 00000000 (2048 bytes of storage addressable)
14 R7: 00000000 (2048 bytes of storage addressable)
15 R8: 00009070 (Module SAM1 program SAM1 WORKING-STORAGE SECTION BLW=0000 + X'0', symbol SYSTEM-DATE-AND-TIME, source line #95)
16 R9: 000088C8 (Module SAM1 program SAM1 + X'18C8', source line #465)
17 R10: 00007124 (Module SAM1 program SAM1 + X'124')
18 R11: 00007798 (Module SAM1 program SAM1 + X'798')
19 R12: 000070FC (Module SAM1 program SAM1 + X'FC')
20 R13: 22C95030 (589776 bytes of storage addressable)
21 R14: 80007D32 (Module SAM1 program SAM1 + X'D32', source line #312)
22 R15: A2C19E80 (Module IGZCPAC + X'5E80')

```

- v. Further down in the report is the 'Associated Open Files'. Each file that was open to this program at the time of the abend is reported. One of the open files has a DD name of CUSTFILE. It's file name and how it was opened ('INPUT' in this example) are shown. Also notice that the contents of the record buffers are displayed.

```

1 Associated Open Files
2
3 File Name . . . . . : CUSTFILE
4 Data Set Name . . . . . : DNET187.ADLAB.FILES(CUST2FA)
5 File Attributes . . . . . : ORGANIZATION=SEQUENTIAL, ACCESS MODE=SEQUENTIAL,
6 RECFM=FIXED
7
8 Last I/O Function . . . . : READ
9 Open Status . . . . . : INPUT
10 File Status Code . . . . . : 0
11 EXCP (Start I/O) Count . . : 2
12 Channel Connect Time . . . : 0.0003 sec
13
14 Previous Record . . . . . : Record data length 80
15 Address Offset Hex EBCDIC
16 -----
17 00024F38 F2F4F0F9 F0D7D7D6 F0F0F9F4 F5D78981 *24090PP000945Pia*
18 00024F48 +10 95964040 40404040 40404040 40404040 *no *
19 00024F58 +20 40404040 4040F2F0 F0F560F0 F760F0F5 * 2005-07-05 *
20 00024F68 +30 0001F2F0 F0F660F1 F260F2F7 40404040 *.2006-12-27 *
21 00024F78 +40 40404040 40404040 40404040 40404040 * *
22
23 Current Record . . . . . : Record data length 80
24 Address Offset Hex EBCDIC

```

- vi. Scroll down until you reach 'Associated Storage Areas'. Notice that all variables in the program are displayed. Their values are shown both as a value and in hexadecimal.



```

001 Associated Storage Areas
002
003 Task Global Table (TGT) at address 000088C8 for length 00000198
004
005 FILE SECTION
006 Off Hex Value          Data Value          Source (Starting at Line # 000058)
007 -----
008 BLF=0000 at address 00024F88 file CUSTFILE
009
010
011 0 FSF4F3F2 F1          *54321          *          05 CUST-REC.
012 5 C3                  *C              *          10 CUST-KEY.
013 6 40404040 404040      *              *          10 CUST-ID
014 D C1A2A385 996B40C4 85A94040 40404040 *Aster, Dez    *          10 CUST-RECORD-TYPE PIC X(5).
015 1D 40                  *              *          10 FILLER          PIC X.
016 1E 7C7B5B6C 50        *@#%$%&        *          10 CUST-NAME       PIC X(7).
017 23 0002                *              *          05 CUST-ACCT-BALANCE PIC X(17).
018 25 E2A39699 94A840C6 819393A2 404040      2          *          05 CUST-ACCT-BALANCE PIC S9(7)V99 COMP-3.
019 34 C481A381 40C595A3 99A840D6 97859981 *Stormy Falls  *          05 CUST-ORDERS-YTD  PIC S9(4)
020 44 A3969940 40404040 40404040 *Data Entry Opera* *          05 CUST-CITY        PIC S9(4)
021                                     *tor          *          05 CUST-CITY        PIC X(15).
022                                     *          *          05 CUST-OCCUPATION  PIC X(28).
023 Off Hex Value          Data Value          Source (Starting at Line # 000072)

```


12. Next, you will display the detail report for the abending program:

a. **Scroll to the top**, and click event **3** (the abending program).

```

11  1 Call          SAM1      SAM1      SAM1      L#312 E+D30      From DNET187
12  2 Call          IGZCPAC   n/a       IGZCFCC   E+2FC        INITIAL LOAD
13  3 Abend S0C7    ***** SAM2      SAM2      SAM2      L#89 E+39A      From DNET187
14

```

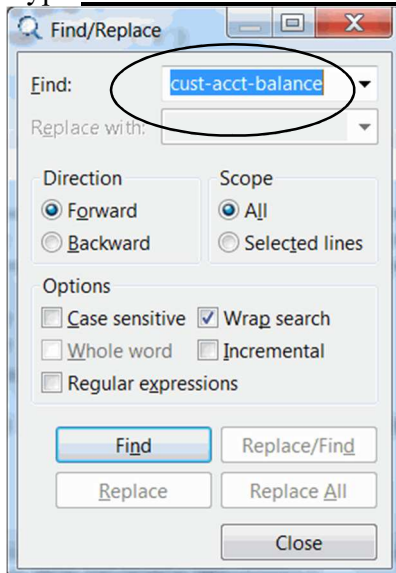


b. The report is positioned to 'Event 3 of 3'. This part of the report is the detail report for the third event, which is program SAM2.

13. Previously, you saw that program SAM2 abended because there was invalid data in a variable named CUST-ACCT-BALANCE. To research this abend, it will be helpful to determine how bad data got into that variable. Next, you will track the data back to its source.

a. When researching an abend caused by invalid data, sometimes it is helpful to examine information about the variable in the 'Associated Storage Areas' section. An easy way to scroll to the variable is to search for it in the report:

- Press the **CTRL+F** keys.
- The Find/Replace dialog is displayed.
  - ◆ Tip: You can also display the Find/Replace dialog by selecting Edit > Find/Replace from the menu.
- Type **CUST-ACCT-BALANCE** in the 'Find' field, and click the **Find** button.



- The report is positioned to the next place that contains the text ‘CUST-ACCT-BALANCE’.

```

-1      INCREMENT RECORD COUNT
-1      ADD +1 TO BALANCE-COUNT
000088 *      *** Add this customer's BALANCE to the grand total ***
000089      COMPUTE BALANCE-TOTAL =
000090      BALANCE-TOTAL + CUST-ACCT-BALANCE]
+1      *      *** Calculate Average ***
+2      COMPUTE BALANCE-AVERAGE =
+3      BALANCE-TOTAL / BALANCE-COUNT
+4      *      *** Calculate Minimum ***
+5      IF WS-FIRST-TIME-SW = 'Y'

Data Field Declarations:

Source
Line #
-----
000059      05  CUST-ACCT-BALANCE      PIC S9(7)V99  COMP-3.
000066      05  BALANCE-TOTAL          PIC S9(7)V99  COMP-3.

Data Field Values:

BALANCE-TOTAL      = 10948.44
CUST-ACCT-BALANCE = X'7C7B5B6C50' *** Invalid numeric data ***

```

- Click the **Find** button a few more times, until you find the variable in the part of the report where all of the variables in the program are displayed (the Associated Storage Areas).

```

demomvs.demopkg.ibm.com:2800/FAULTANL.V13R1.HIST(J00767)-Report  SAM2.COB
670  2F 000000C      0      05 WS-WORK-NUM-5      PIC S9(7)      COMP-3  VALUE +0.
671
672 LINKAGE SECTION
673 BLL=0000 has not been assigned an address
674 Off Hex Value      Data Value      Source (Starting at Line # 000053)
675 -----
676 BLL=0001 at address 00024F88
677      01 CUST-REC.
678      05 CUST-KEY.
679      0 F5F4F3F2 F1      *54321      *      10 CUST-ID      PIC X(5).
680      5 C3      *C      *      10 CUST-RECORD-TYPE      PIC X.
681      6 40404040 404040      *      *      10 FILLER      PIC X(7).
682      D C1A2A385 996B40C4 85A94040 40404040 *Aster, Dez      *      05 CUST-NAME      PIC X(17).
683      1D 40      *      *
684      1E 7C7B5B6C 50      *@#%$%&      *      05 CUST-ACCT-BALANCE      PIC S9(7)V99  COMP-3.
685      23 0002      2      *      05 CUST-ORDERS-YTD      PIC S9(4)      COMP.
686      25 E2A39699 94A840C6 819393A2 404040      *Stormy Falls      *      05 CUST-CITY      PIC X(15).
687      34 C481A381 40C595A3 99A840D6 97859981 *Data Entry Opera*      *      05 CUST-OCCUPATION      PIC X(28).
688      44 A3969940 40404040 40404040      *tor      *
689
690 Off Hex Value      Data Value      Source (Starting at Line # 000064)

```

- Although the variable is defined as numeric, the value ('@#%\$%&') is not numeric.
- Notice that the variable is part of a group variable named CUST-REC. Also notice that it is in linkage section. The abending program, SAM2, is a called program. In COBOL programs, data is passed to called programs in linkage section. The fact that the variable is in linkage section means that the data was possibly passed from the main program SAM1.

14. Next, to track the data back to its source, you will look at the data passed by the main program.

- Scroll to the top, and select event **1** (the main program, called SAM1).

11	1	Call	SAM1	SAM1	SAM1	L#312 E+D30	From DNET187
12	2	Call	IGZCPAC	n/a	IGZCFCC	E+2FC	INITIAL LOAD
13	3	Find S0C7	***** SAM2	SAM2	SAM2	L#89 E+39A	From DNET187
14							

- The report is positioned to event 1 of 3.

```

demomvs.demopkg.ibm.com:2800/FAULTANLV13R1.HIST(J00767)-Report
31
32 =====
33 EVENT 1 OF 3: CALL (DSA ADDRESS 22C95030)
34
35 The source code below was executed via the following sequence of PERFORM statements:
36
37 Source
38 Line #
39 -----
40 000261          PERFORM 100-PROCESS-TRANSACTIONS
41 -----

```

- c. **Scroll down** to the current statement. Notice it is a CALL statement and that it is passing a variable called CUST-REC.
- d. **Scroll down** to the Associated Storage Areas, and locate the CUST-REC variable. Tip: you can press Ctrl+F again to open the Find dialog and search for the variable name.

```

00 -----
01 Associated Storage Areas
02
03 Task Global Table (TGT) at address 000088C8 for length 00000198
04
05 FILE SECTION
06 Off Hex Value          Data Value          Source (Starting at Line # 000058)
07 -----
08 BLF=0000 at address 00024F88 file CUSTFILE
09
10                                01 CUST-REC.
11                                05 CUST-KEY.
12 0 F5F4F3F2 F1          *54321          *          10 CUST-ID          PIC X(5).
13 5 C3                    *C              *          10 CUST-RECORD-TYPE PIC X.
14 6 40404040 404040      *              *          10 FILLER          PIC X(7).
15 D C1A2A395 996840C4 85A94040 40404040 *Aster, Dez    *          05 CUST-NAME      PIC X(17).
16 1D 40
17 1E 7C7B5B6C 50        *@#%$%&       *          05 CUST-ACCT-BALANCE PIC S9(7)V99 COMP-3.
18 23 0002              2              *          05 CUST-ORDERS-YTD  PIC S9(4) COMP.
19 25 E2A39699 94A840C6 819393A2 404040      *Stormy Falls *          05 CUST-CITY      PIC X(15).
20 34 C481A381 40C595A3 99A840D6 97859981 *Data Entry Opera* 05 CUST-OCCUPATION PIC X(28).
21 44 A3969940 40404040 40404040      *tor          *
22
23 Off Hex Value          Data Value          Source (Starting at Line # 000072)
24
25

```

- e. Notice that the bad data ('@#%\$%&') has been traced back to the calling program. Also notice it is in File Section, which indicates that the data may have been read from a file. The DD name of the file is CUSTFILE.

15. Next, look at the file record buffer to see if the invalid data was read from the file.

- a. **Scroll up** to the 'Associated open files' section and notice the information for the CUSTFILE file.

```

126 File Name . . . . . : CUSTFILE
127 Data Set Name . . . . . : DNET187_ADLAB_FILES(CUST2FA)
128 File Attributes . . . . . : ORGANIZATION=SEQUENTIAL, ACCESS MODE=SEQUENTIAL,
129 RECFM=FIXED
130 Last I/O Function . . . . . : READ
131 Open Status . . . . . : INPUT
132 File Status Code . . . . . : 0
133 EXCP (Start I/O) Count. . . : 2
134 Channel Connect Time. . . . : 0.0003 sec
135
136 Previous Record . . . . . : Record data length 80
137 Address  Offset      Hex          EBCDIC
138 -----
139 00024F38          F2F4F0F9 F0D7D7D6 F0F0F9F4 F5D78981 *24090PP000945Pia*
140 00024F48          +10 95964040 40404040 40404040 40404040 *no *
141 00024F58          +20 40404040 4040F2F0 F0F560F0 F760F0F5 * 2005-07-05*
142 00024F68          +30 0001F2F0 F0F660F1 F260F2F7 40404040 *..2006-12-27 *
143 00024F78          +40 40404040 40404040 40404040 40404040 * *
144
145 Current Record. . . . . : Record data length 80
146 Address  Offset      Hex          EBCDIC
147 -----
148 00024F88          F5F4F3F2 F1C34040 40404040 40C1A2A3 *54321C Ast*
149 00024F98          +10 85996B40 C485A940 40404040 40407C7B *er, Dez @#*
150 00024FA8          +20 5B6C5000 02E2A396 9994A840 C6819393 *%$%&..Stormy Fall*
151 00024FB8          +30 A2404040 C481A381 40C595A3 99A840D6 *s Data Entry 0*
152 00024FC8          +40 97859981 A3969940 40404040 40404040 *perator *|
153

```

- b. Notice that CUSTFILE is an input-only file, and that the bad data ('@#\$%&') can be seen in the record buffer.
16. At this point, there can be a conclusion about why this abend occurred. Based on the information in the report, the sequence of events that led up to the abend can be determined:
- a. The main program, SAM1 read a record from the CUSTFILE file. The file contained bad data in the CUST-ACCT-BALANCE field.
  - b. Program SAM1 called program SAM2 and passed the record to it.
  - c. A statement in program SAM2 attempted to perform a calculation using the invalid data, which caused it to fail with a S0C7 abend code.
17. In a real scenario, you might take actions such as fixing the data in the file. Another possible action would be to modify the program so that it would validate numeric data in the file before attempting to perform calculations.
18. You have completed this exercise.



# Lab Exercise 4

## Working with the list of fault entries

In this exercise you will:

- Learn to work with the list of fault entries
- Customize the list of fault entries

1. If the fault history file is already open, and a list of fault entries is displayed, continue with this exercise. If a list of fault entries is not displayed, open the history file. If you are not sure how, complete exercise 2 before continuing.
2. In the previous exercise, you saw that double-clicking a fault entry will open the analysis report. In addition, there are actions you can take such as sorting and filtering the list, and customizing the columns shown.
3. You can sort the list.
  - a. Click the **JOB/TRAN** column heading.
  - b. The entries are sorted based on JOB/TRAN in ascending order.

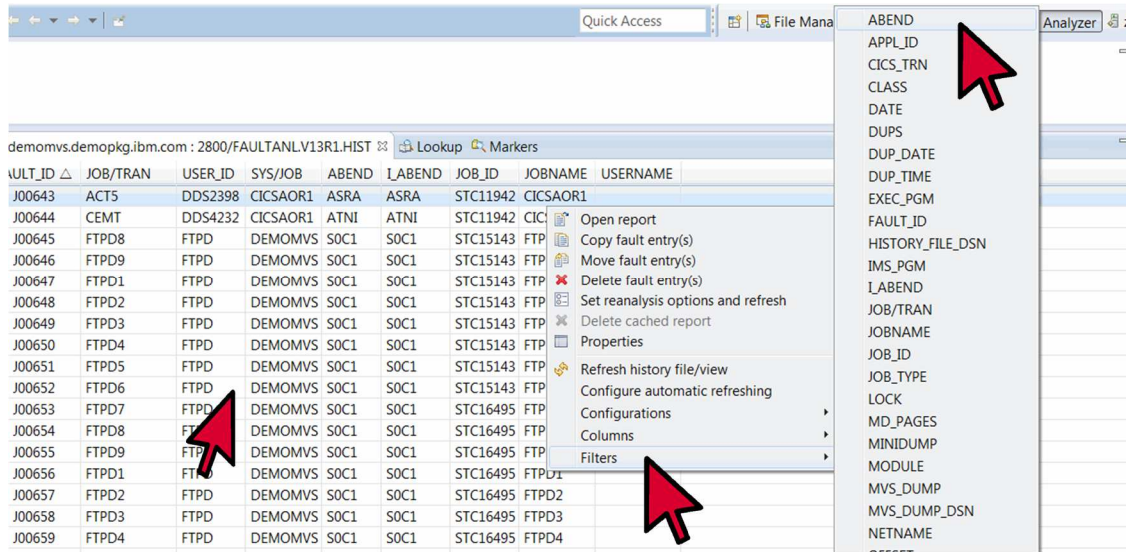
FAULT_ID	JOB/TRAN	USER_ID	SYS/JOB	ABEND	I_ABEND	JOB_ID	JOBNAME	USERNAME
J00766	CEMT	DDS4214	CICSAOR1	ATNI	ATNI	STC09784	CICSAOR1	
> J00759	CEMT	CICSUSER	CICSAOR1	ATNI	ATNI	STC11942	CICSAOR1	
J00723	CEMT	DDS4232	CICSAOR1	ATNI	ATNI	STC11942	CICSAOR1	
J00675	CEMT	DDS2776	CICSAOR1	ATNI	ATNI	STC11942	CICSAOR1	
J00644	CEMT	DDS4232	CICSAOR1	ATNI	ATNI	STC11942	CICSAOR1	
J00741	CFA	DNET324	CICSAOR1	FLT2	FLT2	STC11942	CICSAOR1	
J00740	CFA	DNET324	CICSAOR1	FLT1	FLT1	STC11942	CICSAOR1	
J00739	CFA	DNET324	CICSAOR1	FAD1	FAD1	STC11942	CICSAOR1	
J00738	CFA	DNET324	CICSAOR1	ASRA	ASRA	STC11942	CICSAOR1	
J00713	DB1SWLM	SYSSTC	DEMOMVS	U4038	U4038	STC00956	DB1SWLM	
J00726	DDS04623	DDS0462	DEMOMVS	S0C1	S0C1	STC03334	DDS04623	
J00727	DDS04624	DDS0462	DEMOMVS	S0C1	S0C1	STC03334	DDS04624	
> J00767	DNET187X	DNET187	DEMOMVS	S0C7	S0C7	JOB11774	DNET187X	

- c. Sort the list again on the Fault ID column. Click the **FAULT\_ID** column heading. The entries are sorted based on the fault id.

FAULT_ID	JOB/TRAN	USER_ID	SYS/JOB	ABEND	I_ABEND	JOB_ID	JOBNAME	USERNAME
J00643	ACT5	DDS2398	CICSAOR1	ASRA	ASRA	STC11942	CICSAOR1	
J00644	ACT	DDS4232	CICSAOR1	ATNI	ATNI	STC11942	CICSAOR1	
J00645	FTPD8	FTPD	DEMOMVS	S0C1	S0C1	STC15143	FTPD8	
J00646	FTPD9	FTPD	DEMOMVS	S0C1	S0C1	STC15143	FTPD9	
J00647	FTPD1	FTPD	DEMOMVS	S0C1	S0C1	STC15143	FTPD1	
J00648	FTPD2	FTPD	DEMOMVS	S0C1	S0C1	STC15143	FTPD2	
J00649	FTPD3	FTPD	DEMOMVS	S0C1	S0C1	STC15143	FTPD3	
J00650	FTPD4	FTPD	DEMOMVS	S0C1	S0C1	STC15143	FTPD4	
J00651	FTPD5	FTPD	DEMOMVS	S0C1	S0C1	STC15143	FTPD5	
J00652	FTPD6	FTPD	DEMOMVS	S0C1	S0C1	STC15143	FTPD6	
J00653	FTPD7	FTPD	DEMOMVS	S0C1	S0C1	STC16495	FTPD7	
J00654	FTPD8	FTPD	DEMOMVS	S0C1	S0C1	STC16495	FTPD8	
J00655	FTPD9	FTPD	DEMOMVS	S0C1	S0C1	STC16495	FTPD9	

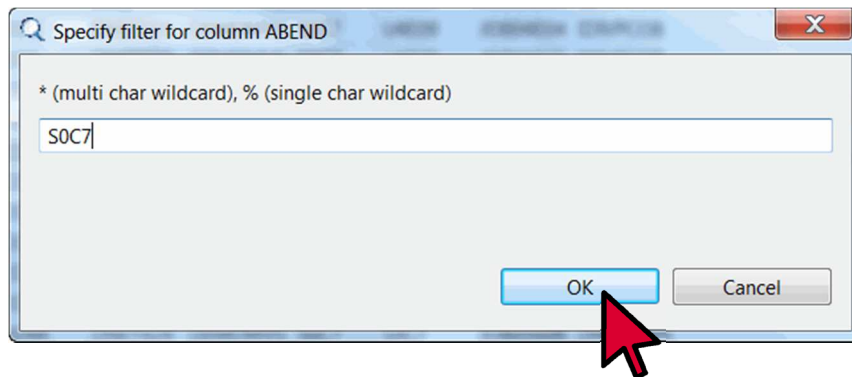
4. You can filter the list to display only selected entries. For example, you can add a filter to display only SOC7 abends:

a. **Right Click** anywhere in the fault list, then select **Filters** > **ABEND**.



b. Enter **S0C7** in the field and click **OK**. You must use capital letters to get a match.

- Tip: you can use wildcard characters \* and % in the filter.



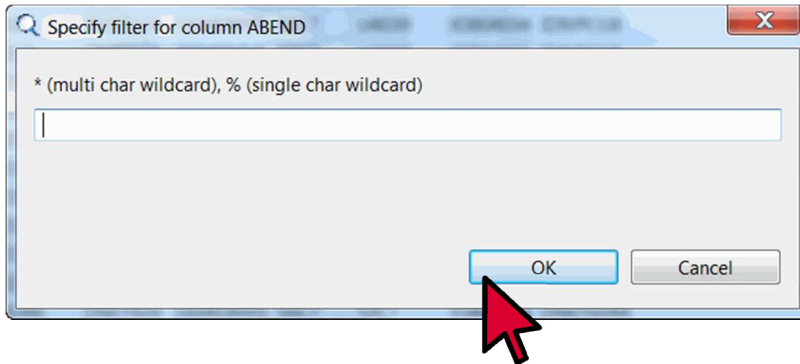
c. Only entries with a SOC7 abend code are displayed.

The screenshot shows the IBM Fault Analyzer interface with the fault list filtered to show only entries with a SOC7 abend code. A red circle highlights the 'ABEND' column. The table data is as follows:

FAULT_ID	JOB/TRAN	USER_ID	SYS/JOB	ABEND	LABEND	JOB_ID	JOBNAME	USERNAME
J00742	IDIVPCOB	DNET324	DEMOMVS	S0C7	U4039	JOB04032	IDIVPCOB	
J00743	IDIVPCOB	DNET324	DEMOMVS	S0C7	U4039	JOB04034	IDIVPCOB	
J00744	IDIVPCOB	DNET324	DEMOMVS	S0C7	U4039	JOB04035	IDIVPCOB	
J00745	IDIVPCOB	DNET324	DEMOMVS	S0C7	U4039	JOB04036	IDIVPCOB	
J00746	IDIVPCOB	DNET324	DEMOMVS	S0C7	U4039	JOB04037	IDIVPCOB	
J00747	IDIVPCOB	DNET324	DEMOMVS	S0C7	U4039	JOB04041	IDIVPCOB	
J00748	IDIVPCOB	DNET324	DEMOMVS	S0C7	U4039	JOB04042	IDIVPCOB	
J00749	IDIVPCOB	DNET324	DEMOMVS	S0C7	U4039	JOB04044	IDIVPCOB	
J00750	IDIVPCOB	DNET324	DEMOMVS	S0C7	U4039	JOB04045	IDIVPCOB	
J00751	IDIVPCOB	DNET324	DEMOMVS	S0C7	U4039	JOB04049	IDIVPCOB	
J00752	IDIVPCOB	DNET324	DEMOMVS	S0C7	U4039	JOB04051	IDIVPCOB	
J00754	DNET424X	DNET424	DEMOMVS	S0C7	S0C7	JOB05498	DNET424X	

d. You can Remove the filter to display all abends again.

- **Right Click** anywhere in the fault list, then select **Filters** > **ABEND**.
- Delete the filter text, then click **OK**.



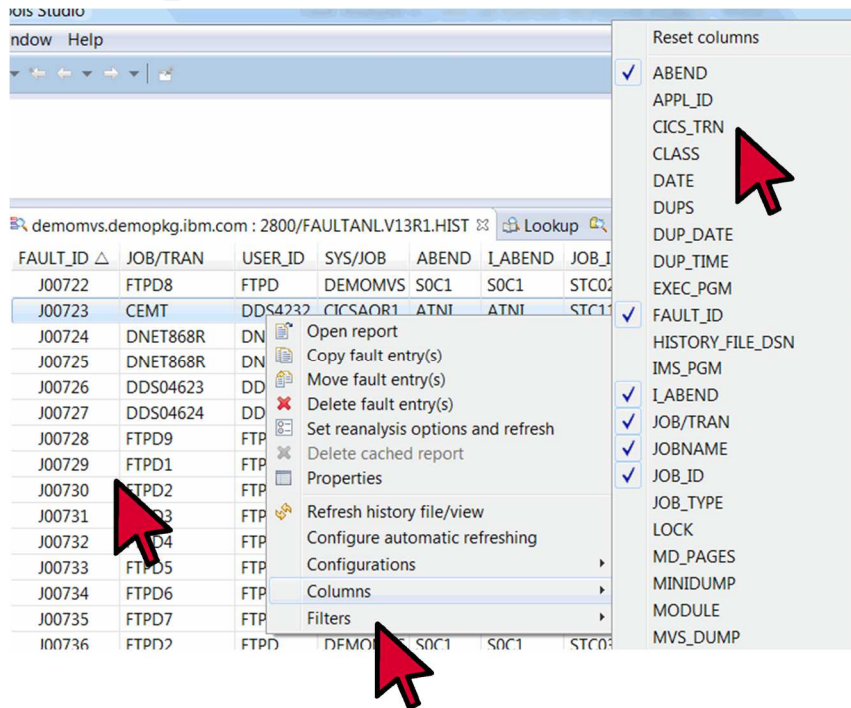
e. The list is displayed showing all entries.

5. You can add and remove columns. Next you will add column 'CICS\_TRN' to the list.

a. **Right Click** anywhere in the fault list, then select **Columns**.

b. Notice that there is a check mark next to each column that is selected.

c. Click **CICS\_TRN**.



d. The CICS\_TRN column now appears in the list of fault entries.

FAULT_ID	JOB/TRAN	USER_ID	SYS/JOB	ABEND	I_ABEND	JOB_ID	JOBNAME	USERNAME	CICS_TRN
J00722	FTPD8	FTPD	DEMOMVS	S0C1	S0C1	STC02692	FTPD8		n/a
J00723	CEMT	DDS4232	CICSAOR1	ATNI	ATNI	STC11942	CICSAOR1	CEMT	
J00724	DNET868R	DNET868	DEMOMVS	S0C4	S0C4	JOB03344	DNET868R		n/a
J00725	DNET868R	DNET868	DEMOMVS	S0C4	S0C4	JOB03346	DNET868R		n/a

e. You can remove a column.

- **Right Click** anywhere in the fault list, select **Columns**.
- Notice that CICS\_TRN is checked now.
- Click **CICS\_TRN**.
- The list is re-displayed, and the CICS\_TRN column is removed.

6. You can drag and drop a column heading to move the column.
- Drag and drop the JOBNAME column header** to make it the second column (after Fault\_ID). (Specifically, click-and-hold the word 'JOBNAME', drag the mouse pointer over until it is just to the right of the word 'FAULT\_ID' in the column heading, then release the mouse button.)

FAULT_ID	JOBNAME	JOB/TRAN	USER_ID	SYS/JOB	ABEND	LABEND	JOB_ID	USERNAME
J00643	ACT5	DDS2398	CICSAOR1	ASRA	ASRA	STC11942	CICSAOR1	
J00644	CEMT	DDS4232	CICSAOR1	ATNI	ATNI	STC11942	CICSAOR1	
J00645	FTPD8	FTPD	DEMOMVS	S0C1	S0C1	STC15143	FTPD8	
J00646	FTPD9	FTPD	DEMOMVS	S0C1	S0C1	STC15143	FTPD9	

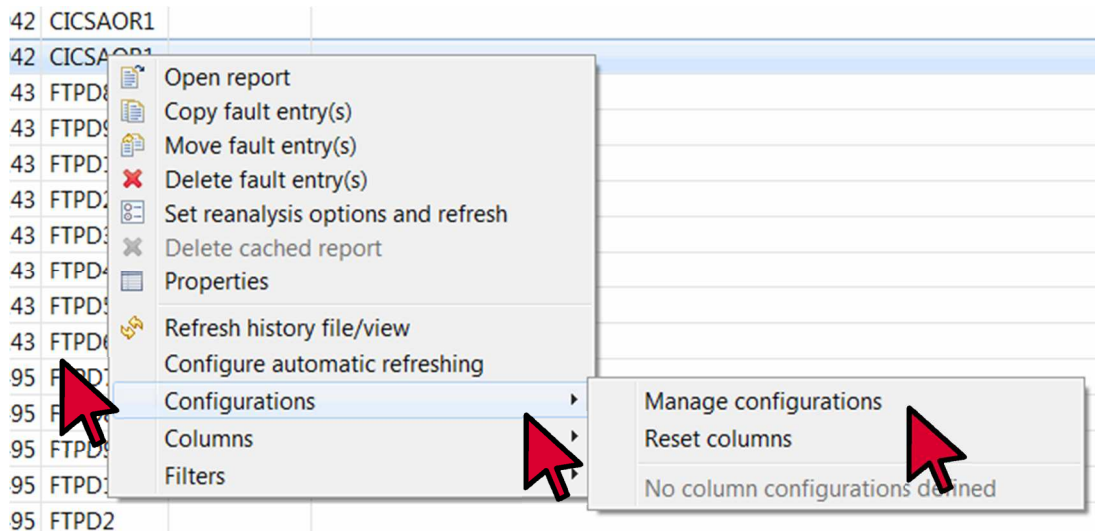
- The new column configuration will be displayed with JOBNAME after FAULT\_ID.

FAULT_ID	JOBNAME	JOB/TRAN	USER_ID	SYS/JOB	ABEND	LABEND	JOB_ID	USERNAME
J00643	CICSAOR1	ACT5	DDS2398	CICSAOR1	ASRA	ASRA	STC11942	
J00644	CICSAOR1	CEMT	DDS4232	CICSAOR1	ATNI	ATNI	STC11942	

- Drag and drop JOBNAME column** back to where it was, to the right of JOB\_ID.

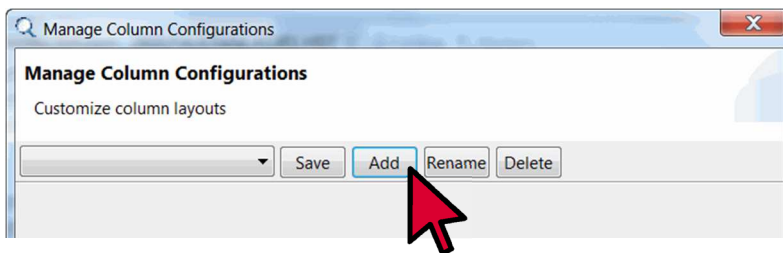
7. You can create a customized column profile.

- Right Click** anywhere in the fault list, and select **Configurations > Manage configurations**.



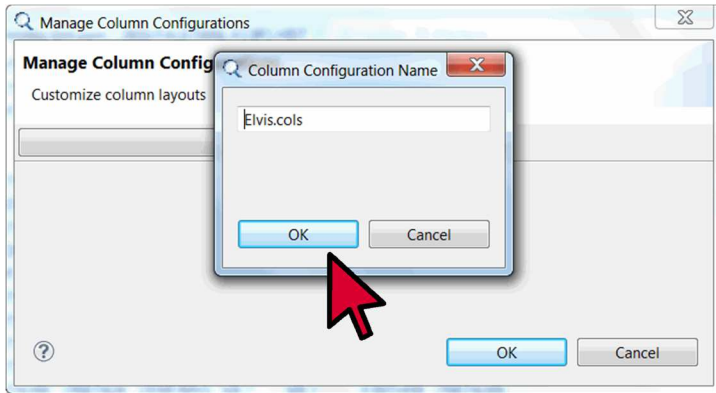
- The 'Manage column configurations' dialog is displayed.

- Click the **Add** button.

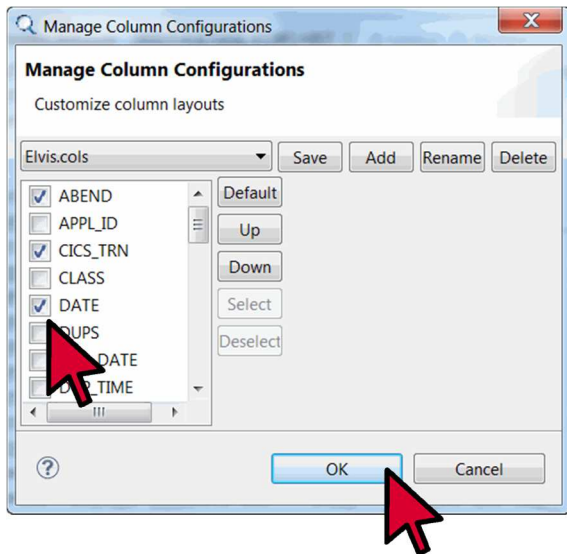


- Enter a name for the new configuration (for example, you can use your own name). Click **OK**.

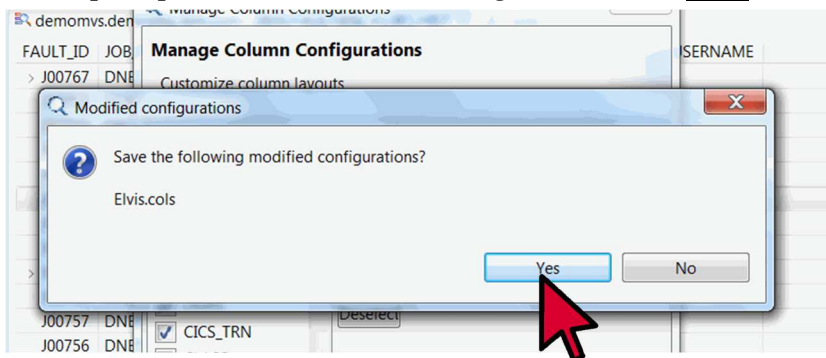




- e. Next you will select the columns that you want in your configuration.
- Click the check box next to all of the column names that you want. (You can select any set of columns).
  - Notice that you can move columns up and down with the Up and Down buttons.
  - When the columns you want are selected and in the order you want, click **OK**.

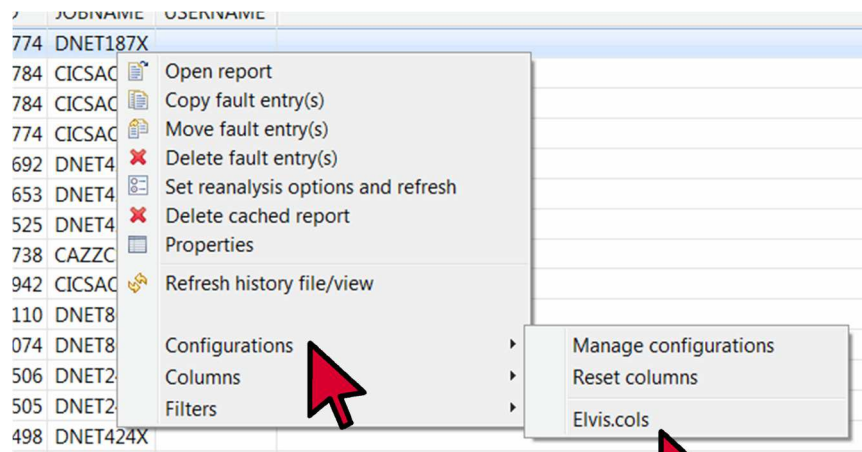


- f. When prompted to save the new configuration, click **YES**.



8. Next you will activate the new configuration.

- a. **Right Click** anywhere in the fault list, then click **Configurations** then click the name of the new configuration.



b. The columns are displayed according to the new configuration. Note that this configuration will be used by default the next time you use Fault Analyzer in this eclipse workbench.

9. You have completed this exercise.

## Summary

Congratulations, you have completed the exercises and have familiarized yourself with the Fault Analyzer eclipse interface.

In this lab:

- You displayed a list of abends in a Fault History file
- You viewed an abend report
- You researched an abend to determine the cause of the failure
- You saw how the abend list can be customized

If you have any questions about Fault Analyzer, please contact the instructor.



© Copyright IBM Corporation 2015. All rights reserved.

The information contained in these materials is provided for informational purposes only, and is provided AS IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, these materials. Nothing contained in these materials is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software. References in these materials to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates.

IBM, the IBM logo, and other IBM products and services are trademarks of the International Business Machines Corporation, in the United States, other countries or both. Other company, product, or service names may be trademarks or service marks of others.