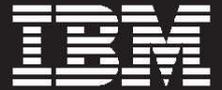


Improves developer productivity and decreases deployment costs by helping analyze and correct application failures quickly

WebSphere. software



## IBM Fault Analyzer for z/OS, V13.1

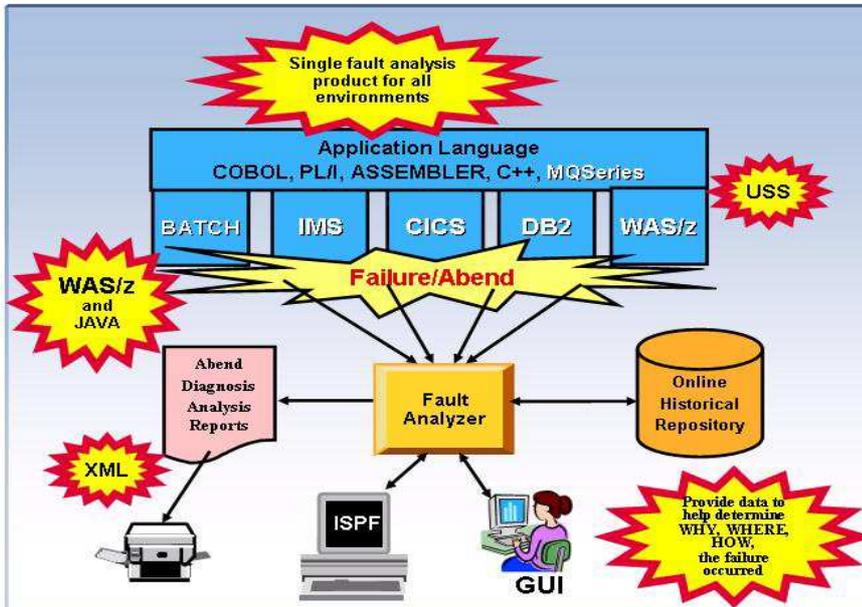


Figure 1: Fault Analyzer environment

### Highlights

- Provides a detailed report about program failures to help resolve them quickly
- Enables you to track and manage application failures and fault reports
- Offers a view of storage contents, trace tables and terminal screen images at the time of failure to help speed corrective action
- Provides the ability to customize message descriptions to be used in application-failure reports
- Helps maximize flexibility through support for CICS, DB2, IMS, Enterprise PL/I, Enterprise COBOL and Java for z/OS systems
- Workstation GUI interface

challenging business demands, application developers must work quickly — with as few errors as possible. The slightest application or system failure can jeopardize goals crucial to business success.

You can meet these challenges more effectively by gaining greater visibility into application events to help analyze and resolve problems before they can pose a serious threat to your business objectives.

IBM® Fault Analyzer for z/OS®, V13.1 is a robust problem determination tool that helps you discover why applications fail.

It helps you repair these failures quickly by gathering information about an application and its environment at the time of failure. You can also develop and test new and existing applications more productively, helping to reduce costs along the way.

When an application abend occurs, Fault Analyzer captures and analyzes real-time information about the application and its environment, then generates an analysis report detailing the cause of the failure. The report describes the failure in terms of the application code, so you no longer lose time reviewing cumbersome, low-level system error messages. Fault Analyzer allows you to choose a report format to locate the information more easily.

Each application abend is recorded by Fault Analyzer in a fault-history file by job name, failure code and other details, along with the analysis report and storage pages referenced during the analysis. This information can later be retrieved to reanalyze the failure, helping to speed future application development. You can logically group fault-history files, and restrict each user's access to particular groups. You can also transfer fault records between history files for analysis on remote systems.

In an increasingly complex and competitive environment with

Designed for usability, Fault Analyzer includes softcopy versions of selected manuals from the IBM z/OS Online Library. Fault Analyzer extracts message and failure-code descriptions from these manuals and inserts them into the analysis report where applicable, freeing your developers from researching the explanation of message and failure details.

And to maximize flexibility, Fault Analyzer allows you to specify criteria to exclude particular jobs from problem determination.

You can also provide your own descriptions for messages. In addition, you can write your own user exits in a variety of languages, including Assembler, C, COBOL, PL/I or Restructured Extended Executor (REXX). For example, you can write user exits to:

- *Override Fault Analyzer options.*
- *Access compiler listings that are compressed or available only*

*through a proprietary access method.*

- *Provide custom explanations for particular message IDs to be included in the analysis report when these messages occur.*
- *Notify specific people about an application or system failure.*
- *Provide your own user section to the analysis report.*

## Maximize your development assets

Because Fault Analyzer starts only after an application abend occurs, you won't waste valuable processing resources during normal application implementation. You can choose a different storage-display concept, based on the total size of eligible storage involved.

You can use this feature to prevent the generation of what can be extremely large reports in situations where, for example, a

COBOL program has a very large working storage area. You can also opt to delete fault-history file entries based on the percentage of space used.

Although Fault Analyzer does not require you to recompile applications, you can choose to produce a side file that identifies the failure source statement. These side files use less DASD space than traditional compiler listings. In the absence of a side file, Fault Analyzer uses the compiler listing. If you choose to keep Fault Analyzer side files instead of compiler listings, you can produce a readable listing file from a side file.

Or you can request prompting for only the point-of-failure side file during interactive reanalysis. You have a wide choice in the source of side-files. For example, you can use COBOL and Enterprise PL/I SYSDEBUG side files and map storage areas in an abend when

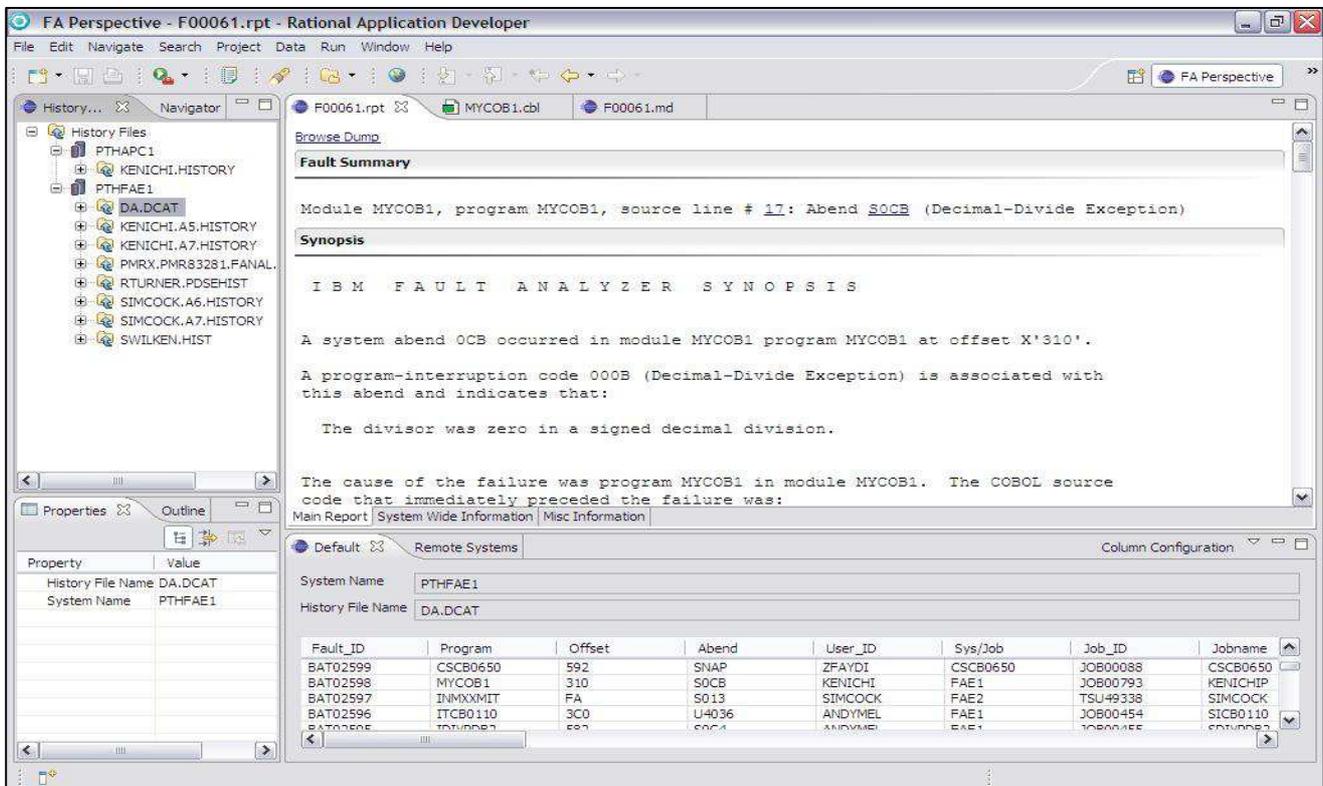


Figure 2: Rational Developer for System z works with Fault Analyzer to make mainframe developers more productive

the Assembler side files are available.

## Support for IBM subsystems

### CICS®

CICS channels and containers data areas are formatted to assist programmers developing CICS Web-services applications.

Fault Analyzer functions are available from within a CICS transaction-based environment in a manner that is consistent with the ISPF usage, permitting you to review and analyze fault entries without the need for a TSO logon.

The ability of Fault Analyzer to detect storage violations is particularly helpful for system programmers who need to debug CICS system problems.

A number of options help optimize the processing of dumps within the CICS environment, which can speed processing of duplicate dumps (the 'NoDup CICSfast' option) and skip analysis of repeated abends in busy systems (the DeferredReport option).

### DB2®

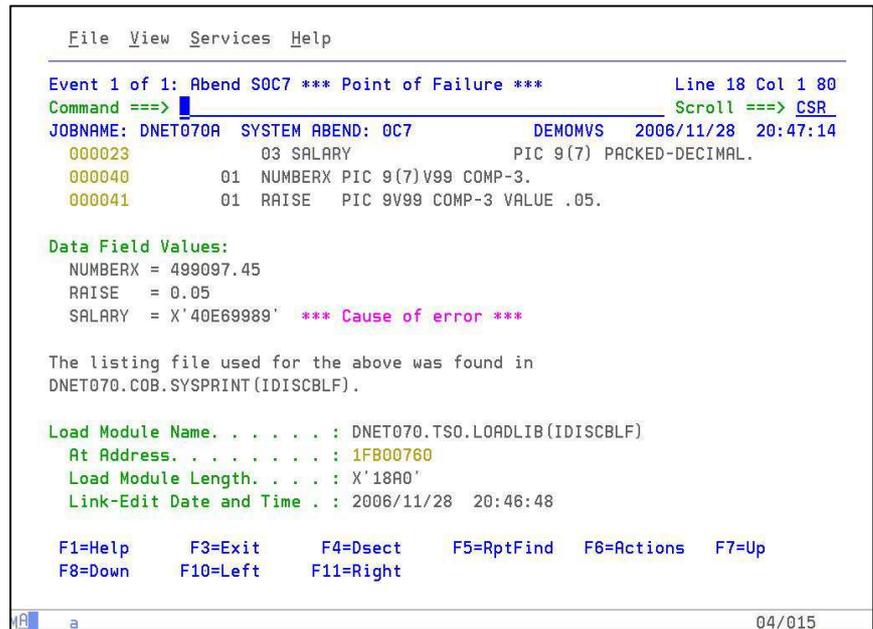
Ongoing support for the new versions of DB2 is provided.

### IMS™

An option also is provided to optimize the processing of dumps within the IMS environment, which can speed processing of duplicate dumps (the NoDup ImageFast IMS option).

### WebSphere® MQ

API information and WebSphere MQ return-code diagnostics are provided to improve problem determination of WebSphere MQ applications.



```
File View Services Help
Event 1 of 1: Abend SOC7 *** Point of Failure *** Line 18 Col 1 80
Command ==> | Scroll ==> CSR
JOBNAME: DNET070A SYSTEM ABEND: 0C7 DEMOMVS 2006/11/28 20:47:14
000023 03 SALARY PIC 9(7) PACKED-DECIMAL.
000040 01 NUMBERX PIC 9(7)V99 COMP-3.
000041 01 RAISE PIC 9V99 COMP-3 VALUE .05.

Data Field Values:
NUMBERX = 499097.45
RAISE = 0.05
SALARY = X'40E69989' *** Cause of error ***

The listing file used for the above was found in
DNET070.COB.SYSPRINT (IDISCBLF).

Load Module Name. . . . . : DNET070.TSO.LOADLIB (IDISCBLF)
At Address. . . . . : 1FB00760
Load Module Length. . . . : X'18A0'
Link-Edit Date and Time . : 2006/11/28 20:46:48

F1=Help F3=Exit F4=Dsect F5=RptFind F6=Actions F7=Up
F8=Down F10=Left F11=Right

04/015
```

Figure 3: Fault Analyzer provides analysis reports detailing the cause of failure including application information

### Java

Ongoing support for the new versions of Java™ are provided.

## Analyze application failures to improve developer productivity

Fault Analyzer provides three modes to help you better track and analyze application and system failure information:

### Real-time analysis

Fault Analyzer includes exit programs (for CICS, IBM Language Environment® and z/OS systems) that it adds to the normal failure processing for these environments. When an application failure occurs in any of these environments, the Fault Analyzer exit starts real-time analysis. After failure processing, you can view the analysis report in the job output or through the Fault Analyzer ISPF interface.

### Batch reanalysis

Batch reanalysis generates a new analysis report. This report is based on the dump and

information gathered in real time, but with potentially different options specified, or with compiler listings or side files made available.

You can submit a Fault Analyzer batch-mode job using either the Fault Analyzer fullscreen ISPF interface or your own job control language (JCL).

### Interactive reanalysis

Interactive reanalysis runs under ISPF and enables you to navigate onscreen through a formatted, structured view of a fully detailed reanalysis. This Fault Analyzer mode enables you to view working storage and control blocks at the time the dump was written.

The ISPF interface has many point-and-shoot fields for easy navigation through the interactive reports. For example, when viewing an analysis report onscreen in interactive mode, you can use the cursor to select a storage address to view the contents of storage at the time of the dump.

You can also associate addresses with notes that might assist with solving a problem. The notes are saved in the fault entry and are automatically displayed whenever the storage for the associated address is displayed. Interactive reanalysis menus are structured to help you access frequently used data and quickly navigate through the analysis information. During interactive reanalysis, you can use the FIND command to locate data anywhere in storage that was saved during real-time analysis. This storage display is not limited to the currently displayed page.

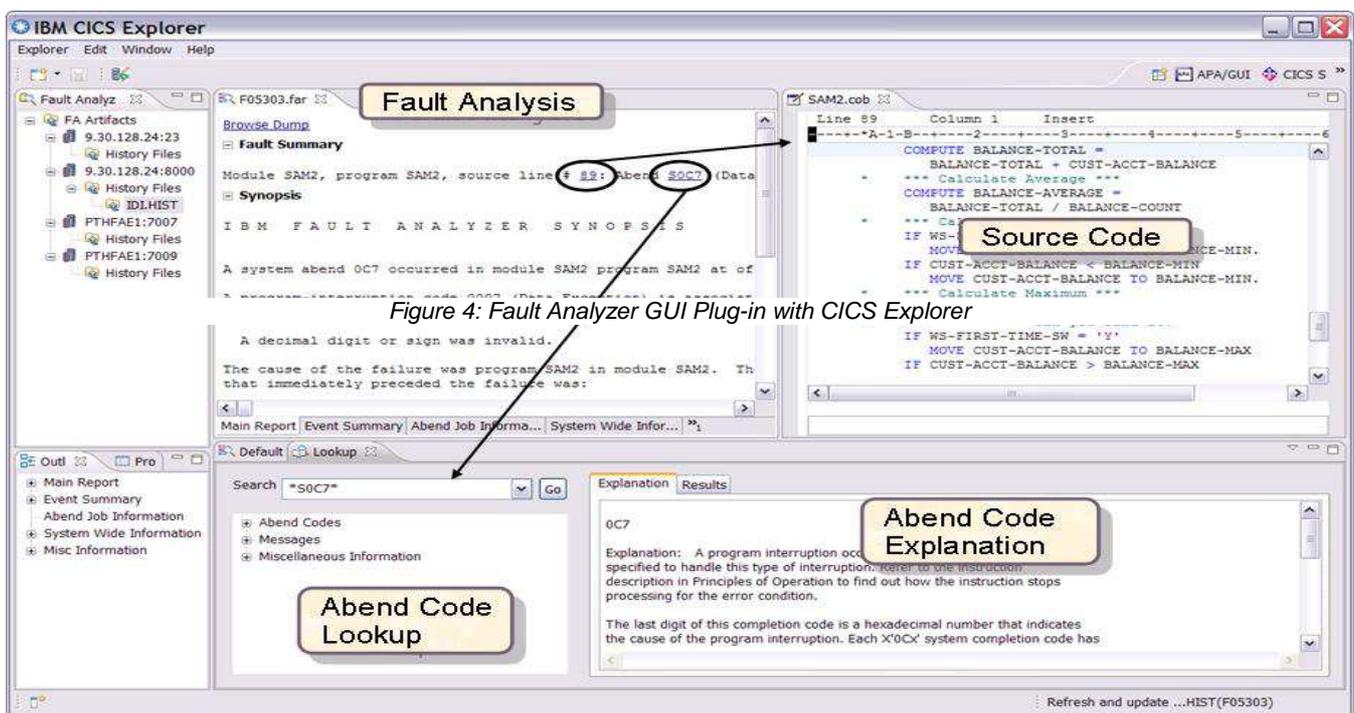
### New in V13.1

- Reporting of a mismatch sidefile is now more consistent with precise details on which sidefile matching tests have passed and failed.
- Users can now provide a list of programs, which include a generic suffix, of those programs for which sidefile processing should be skipped/excluded.
- The new IDIXSFOR exit is added that allows the automatic forced usage of a sidefile

test is the last 12 instructions check.

- Storage usage improvements are made where a single Fault Entry is selected for interactive reanalysis from the ISPF Fault Entry display.
- At the time of a CICS transaction abend analysis, additional information is now gathered about concurrently running CICS tasks and their storage allocations. This information may be useful during storage overlay analysis.
- The Fault Analyzer web interface is redesigned to have a similar look and feel and function as the Eclipse plug-in.
- Users can now provide their own abend code and module descriptions which will be automatically included in the Fault Analyzer report.
- To facilitate easier management of fault entries, the MATCH command now supports a TODAY keyword.
- The option to exclude CICS trace entries as well as the existing include option is provided.
- The SHOWFREE command is added to allow users to determine Fault Analyzer's TSO/ISPF storage usage.

- Enhancements are made to allow improved, more consistent sharing of history file data sets across LPARs in a PDSESHARING(NORMAL) environment.
- Changes are made to allow transaction abend analysis on CICS OPEN TCBS.
- Sidefile processing is enhanced to support DWARF information produced by the Enterprise COBOL V5.1 compiler.
- Currency support is added for Enterprise COBOL V5, CICS TS V5.1, IMS V13, DB2 V11.
- Analysis of abends, which include Java, is improved.
- The new FAISPF option is delivered as consolidation of ISPF specific options - HistCols, InteractiveExitPromptSeconds and InitHist.
- A new utility, IDIDATST, is added that allows for bulk program and sidefile matching. This is particularly useful during initial installation to determine which programs do not have a matching sidefile and hence may require a recompilation in order to generate a matching sidefile.
- Updated and expanded LOOKUP messages now include JES2 and JES3 messages.



where the only failing sidefile

## IBM Fault Analyzer for z/OS, V13.1, at a glance

### Hardware requirements

- Any hardware configuration capable of running IBM z/OS

### Software requirements

- IBM z/OS, V1.12 (5694-A01) or later
- Depending on the functions used in the product, other related products may be required. See the product Web site, listed below, for more detailed software requirements

## Part of a leading-edge family of z/OS tools

Fault Analyzer for z/OS is part of the IBM Problem Determination family of products. These include IBM Application Performance Analyzer for z/OS, IBM Debug Tool for z/OS, IBM Fault Analyzer for z/OS, IBM File Manager for z/OS, and IBM Workload Simulator for OS/390 and z/OS.

Designed to help you maximize your investment in IBM System z products, these products are a robust suite of integrated development tools that enable you to improve IT operational efficiency and transform applications to achieve greater business flexibility.

The IBM Problem Determination tools help application developers to improve application delivery throughout the application life cycle. You can use these tools to help increase productivity and IT effectiveness across source code debugging, application ABEND analysis, data management and application performance analysis.

The IBM Problem Determination tools do much more than support traditional applications. They include capabilities that enable you to build SOA applications. They are tightly integrated with other tools in the IBM problem determination tools portfolio, as well as other tool sets. The tools also continue to support and make the most of the latest subsystem levels. These capabilities help make IBM problem determination tools an excellent choice for your business.

## For more information

To learn more about IBM problem determination tools, contact your IBM representative or IBM Business Partner, or visit the following Web sites:

IBM Problem Determination Tools family:

[ibm.com/software/awdtools/deployment/](http://ibm.com/software/awdtools/deployment/)

IBM Application Performance Analyzer for z/OS:

[ibm.com/software/awdtools/apa](http://ibm.com/software/awdtools/apa)

IBM Debug Tool for z/OS:

[ibm.com/software/awdtools/debugtool/](http://ibm.com/software/awdtools/debugtool/)

IBM Fault Analyzer for z/OS:

[ibm.com/software/awdtools/faultanalyzer](http://ibm.com/software/awdtools/faultanalyzer)

IBM File Manager for z/OS:

[ibm.com/software/awdtools/filemanager](http://ibm.com/software/awdtools/filemanager)

IBM Workload Simulator for z/OS and OS/390:

[ibm.com/software/awdtools/workloadsimulator](http://ibm.com/software/awdtools/workloadsimulator)



© Copyright IBM Corporation 2013

IBM Corporation  
555 Bailey Ave  
San Jose, CA 95141, U.S.A.

Produced in the U.S.A.  
October 2013  
All Rights Reserved

IBM, the IBM logo, [ibm.com](http://ibm.com), and WebSphere are trademarks of IBM Corporation, registered in many jurisdictions worldwide. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at [ibm.com/legal/copytrade.shtml](http://ibm.com/legal/copytrade.shtml).

Java and all Java-based trademarks and logos are trademarks of Sun Microsystems, Inc, in the United States, other countries, or both.

Microsoft and Windows are trademarks of Microsoft Corporation, in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries. Other company, product and service names may be trademarks or service marks of others.

References in this publication to IBM products or services do not imply that IBM intends to make them available in all countries in which IBM operates.

The IBM home page on the internet can be found at [ibm.com](http://ibm.com)

**TAKE BACK CONTROL WITH** 