

Increase productivity and quality of your applications while lowering development costs with IBM Problem Determination Tools and Rational Developer for System z

Dan Brown – Technical Manager

Mitch Funsten – Rational Senior Technical Specialist

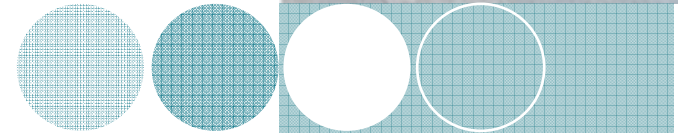
Allan Tortorice – PD Tools Senior Technical Specialist

January Meeting 2010



Agenda

- ▶ RDz overview
- ▶ New Face of PD Tools in RDz
- ▶ Debug Tool Interface
- ▶ Fault Analyzer Interface
- ▶ File Manager Interface
- ▶ Q & A



Why Rational Developer for System z?

A user interface's design affects the amount of **effort** needed to **provide** input and **interpret** output from a system, and the **effort** required to **learn** how.



Developers/System Administrators challenges:

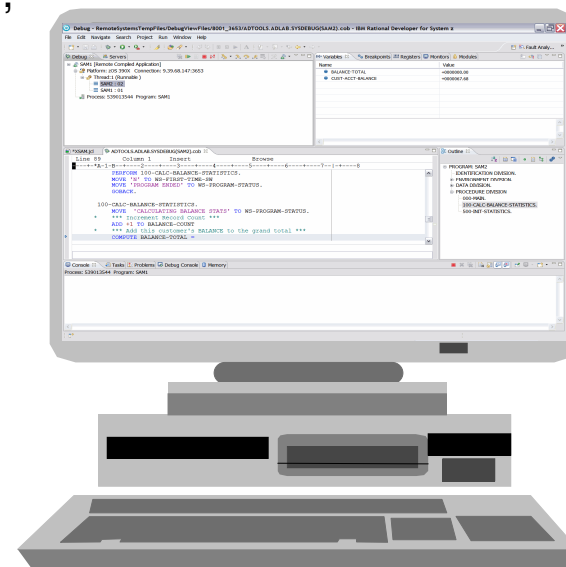
- ▶ “New to mainframe” workloads (Linux, JEE, ERP, z/VM, BI/EPM)
- ▶ Development of a mixture of batch and online applications (Java, Cobol, CICS)
- ▶ Expanding capacity, use, and workloads
- ▶ Experience level deficiencies
- ▶ Employee turnover
- ▶ Retirement of experts

IBM Rational Developer for System z helps companies enhance existing and create new applications in a **more productive** manner and on an **easy to learn** platform .



Rational Developer for System z (RDz) overview

- **What is RDz?**
 - ▶ Eclipse-based Integrated Development Environment (IDE)
- **RDz supports Enterprise Modernization**
 - ▶ Links WAS and core system z processing
 - ▶ Common IDE for COBOL, PL/I, C, C++, HLASM, Java, EGL and web services
 - ▶ Transforms UML to COBOL source code
 - ▶ Interactive access to z/OS for development, job generation, submission, monitoring, debug, command execution
 - ▶ Supports new and existing runtimes (CICS, IMS, Batch, USS, DB2 SP, WAS)
- **RDz supports SOA**
 - ▶ Enables CICS and IMS applications for web services and SOA
 - ▶ Supports for J2EE, JCA, XML, web services



Definition of terms for Rational Developer for System z

■ Workbench

- ▶ Refers to the desktop development environment.
- ▶ Common paradigm for creation, management, and navigation of workspace resources.
- ▶ Each Workbench window contains one or more perspectives.

■ Perspective

- ▶ Controls what appears in certain menus and tool bars.
- ▶ Contains a collection of views and editors.

■ Views

- ▶ Support editors and provide ways to navigate the information in your Workbench.
- ▶ A view might appear by itself, or stacked with other views in a tabbed notebook.

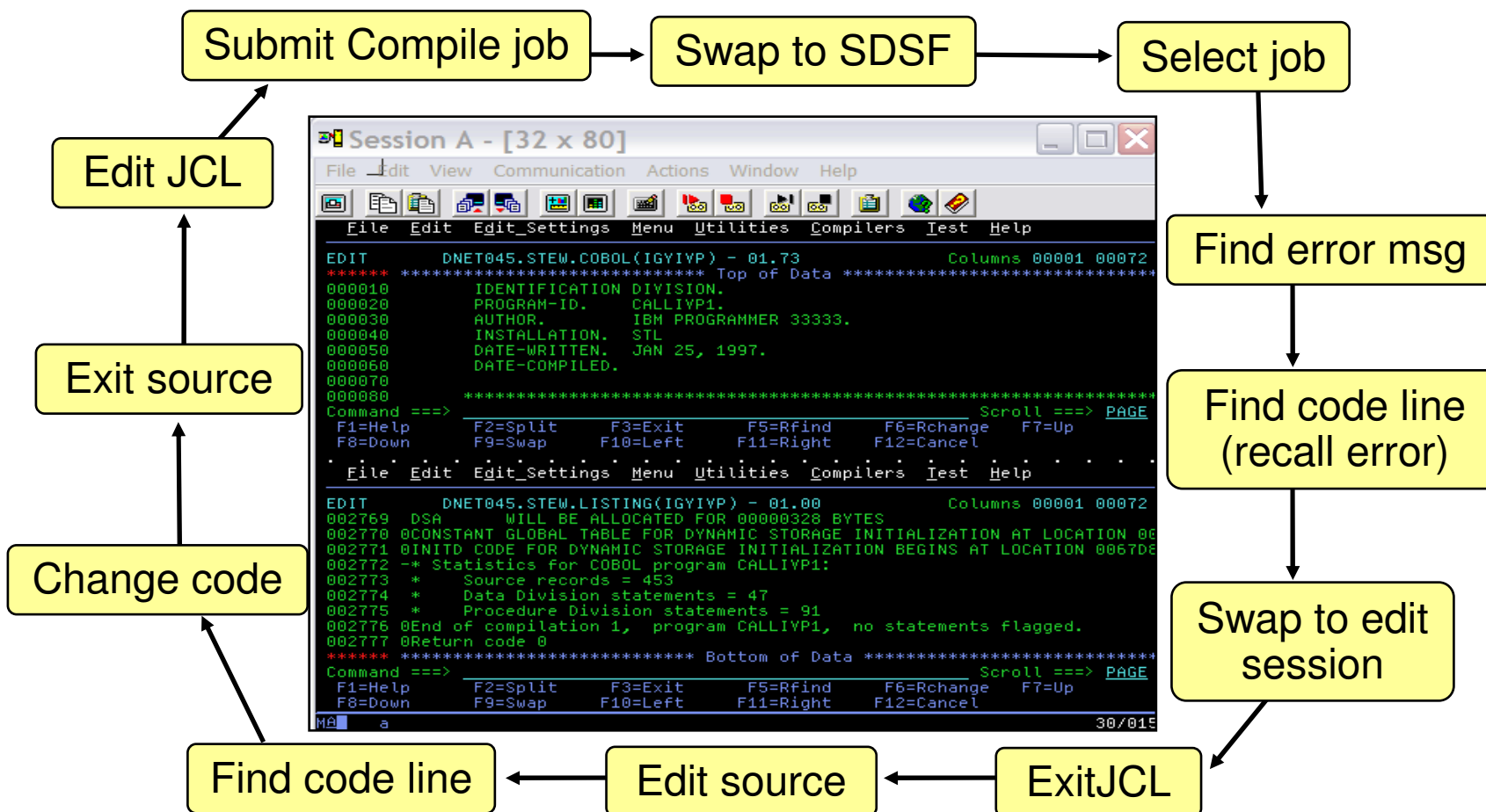
■ Editor

- ▶ Different editors are associated with different types of files.
- ▶ Any number of editors can be open at once.
- ▶ Source Code Editor - Visually configurable editor, assisted statement creation, and syntax checking for COBOL, PL/I, C, C++, High-Level Assembler, and Java languages
- ▶ Data Editor – Formatted driven display of VSAM, PDS Member, and Sequential File data.



ISPF-based development

- Multiple screens/sessions and multiple disparate tools
- 20 x 80 characters of content
- At least 11 steps!



RDz-based development

- Common development environment for COBOL, PL/I, C/C++, and Java
- Simplified development with more information at your fingertips

The screenshot displays the IBM Rational Software Development Platform interface for COBOL development. The main editor window shows the source code for 'REGIOA.cbl' with the following content:

```

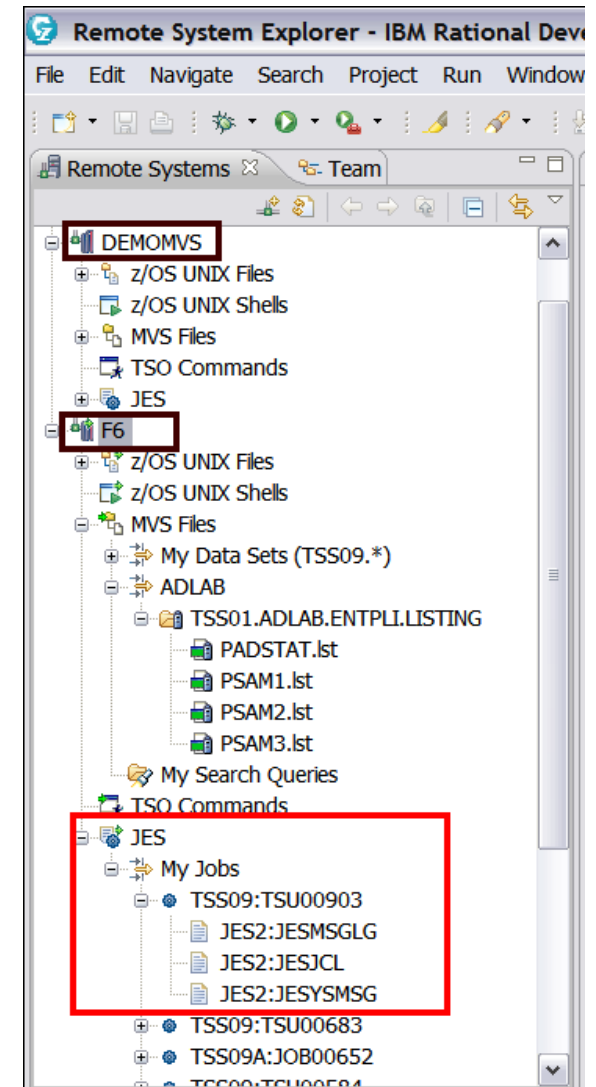
Line 35      Column 1      Insert
-----*A-1-B-----2-----3-----4-----5-----+
000035      DISPLAY "Program REGIOA STARTING "
000036      MOVE 2 TO BRANCHFLAG.
000037      MOVE 'AAAAAA' to FIELD-A.
000038      MOVE 'BBBBBB' to FIELD-B.
000039      MOVE 'CCCCCC' to FIELD-C.
    
```

Callouts highlight the following features:

- Submit jobs, view output, open source members:** Points to the project tree on the left where 'REGIOA.cbl' is selected.
- Open and edit source and JCL simultaneously:** Points to the 'REGIOA.cbl' tab in the editor window.
- Syntax Check:** Points to the 'Tasks' pane at the bottom, which shows '1 error, 0 warnings, 0 infos'.
- Error description:** Points to the error message in the 'Problems' pane: 'IGYPS2072-S "DISPLAI" was invalid. Skipped to the next verb, period or procedure-name d'.
- Outline view:** Points to the 'Outline' pane at the bottom left, showing the structure of the program with '010-INITIALIZATION.' selected.

Navigate datasets or jobs live on zOS

- Connect to multiple hosts concurrently
- Respects existing security configurations and user IDs
- Search, filter, browse, edit, compare, migrate, and allocate new MVS datasets and USS files
- Copy source code, members, or datasets between systems with a few mouse clicks.
- Access JES queues submit jobs, view job state, and open output spools
- Submit TSO or USS commands
- Add datasets and members into projects to group applications and work items together logically
- Open an emulator in the IDE to configured hosts



New Face of PD Tools in RDz

Editing data with File Manager

The screenshot displays the IBM WebSphere Developer for System z interface with several key components:

- Left Panel:** Project Explorer showing a tree structure of files and folders under 'z/OS Projects'.
- Top Center:** Source code editor for 'F01585.far' showing COBOL code with annotations. A red arrow points to the 'Data field values at time of abend:' section, which includes:


```
CUST-ACCT-BALANCE = X'7C7B5B6C50' *** Cau
            WS-UPDATE-NUM = 7.89
```
- Bottom Center:** Event Summary window titled 'Displaying Fault Analyzer report' showing 'IBM FAULT ANALYZER EVENT'.
- Right Panel:** File Manager showing a table of data for 'JPHILDN.ADLAB.SAMFILE'. A red arrow points to the table. Below it, a 'Single Mode' view shows details for a selected record:

Field	Data
REC-TYPE	A
NAME	Turner, Paige
ACCT-BALANCE	*****
ORDERS-YR	4
- Bottom Left:** Properties window showing details for the current file.
- Bottom Right:** Outline window showing the project structure.
- Bottom Center:** Debug window showing assembly code for 'JPHILDN.EPS.SYSDEBUG(SAM2)'. A red arrow points to the 'END-EVALUATE' instruction, with the text 'Debugging application with Debug Tool UAF'.
- Bottom:** Remote Error List and System Details panels. The System Name is 'ZTECMOP' and the Fault History File is 'ADTCFG.IDL.HIST'. Below these is a table of fault records:

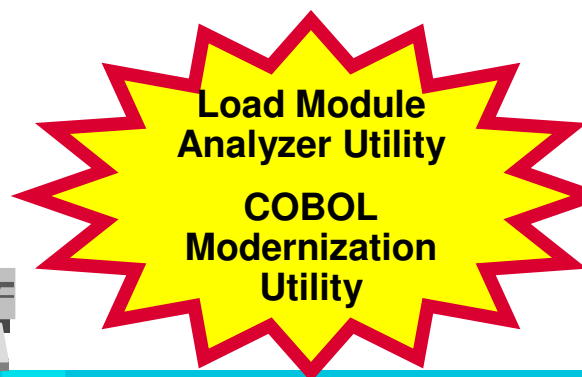
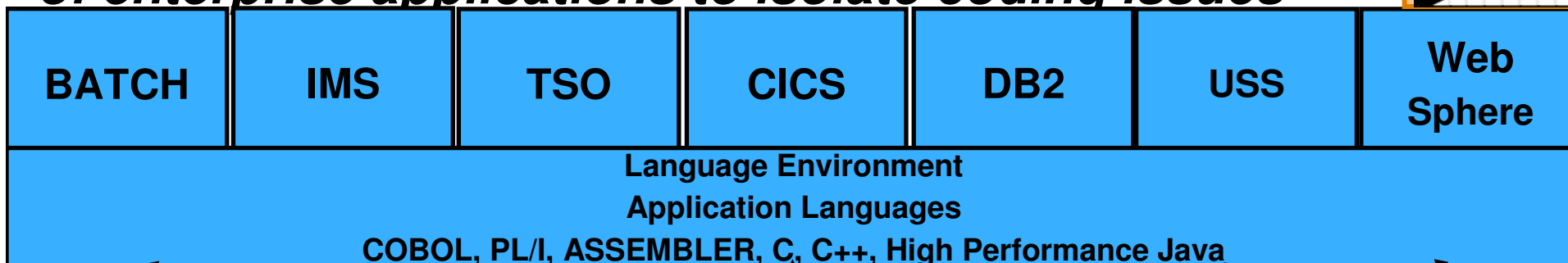
Fault_ID	Job/Tran	User_ID	Sys/Job	Abend	I_Abend	Job_ID	Jobname
F01585	JPHILDNS	JPHILDN	ZT01	S0C7	S0C7	JOB05521	JPHILDNS
F01584	TRAD	CICSUSER	ADTMONO	AED3	AED3	STC02947	ADTMONO
F01583	CEDF	CICSUSER	ADTMONO	ATNI	ATNI	STC02947	ADTMONO

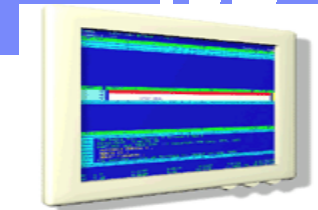
Debug Tool Interface with RDz





Debug Tool – *Debugging of enterprise applications to isolate coding issues*





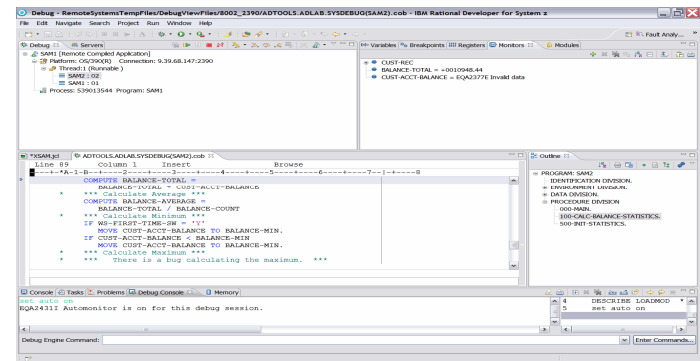
Debug Tool Features

- Source-level debugging
- Multiple breakpoints
 - ▶ Conditional
 - ▶ Unconditional
- Step mode debugging
- Dynamic patching
 - ▶ Modify variables
 - ▶ Insert statements
- Command logging of debugging session
- Frequency sampler
- Programmable command entry
- Utilizes LE services
 - ▶ Breakpoints at LE conditions
 - ▶ Recovery of program abends
- Interactive playback support
- Automonitor support for COBOL, PL/I, and Assembler programs
- Save and restore sessions settings
- An interface to the Fault Analyzer
- Commands to query, allocate, and free files
- Object level disassembly debugging
- CICS
 - ▶ Detect storage violations
 - ▶ Memory editor and browser
 - ▶ Display channel and containers
 - ▶ Pattern-matching breakpoints
- 64-bit register support Assembler



Debug Tool Interface with RDz

- Use the cross-platform debugger to debug end-to-end systems as they execute in the runtime
 - ▶ CICS
 - ▶ IMS
 - ▶ DB2
 - ▶ Batch
 - ▶ WAS
- From the workstation:
 - ▶ View executing source code
 - ▶ Step through host code line-by-line
 - ▶ Set breakpoints at statements, programs, and conditions
 - ▶ View and change the values of variables, storage, and register values
 - ▶ Team Debugging
 - ▶ Etc...
- Debug zOS and distributed code in the same interface even stepping between runtimes and platforms!



The Debug Tool Perspective

The screenshot displays the IBM Rational Developer for System z interface in the debug perspective. The main window shows the source code for `ADTOOLS.ADLAB.SYSDEBUG(SAM2).cob`. The `Program Stack` window on the left shows the current thread and program. The `Monitors and Listings` window on the right shows active monitors. The `JCL Tab` is visible at the top left of the source code editor. The `Active Source Code` window highlights the current line of code. The `Program Outline` window on the right shows the program's structure. The console at the bottom shows the process and program information.

Program Stack

Monitors and Listings

JCL Tab

Active Source Code

Program Outline

The Debug perspective when a program is being debugged

The Debug view

The screenshot displays the IBM Debug view interface. At the top, a yellow callout box contains the text: "See your currently running programs in the Debug view". Below this, the "Debug" pane shows a tree view of the program call chain. The call chain is as follows: Process: 327205648 Program: SAM1 (at the bottom), then SAM1 : 01, then SAM2 : 02 (highlighted with a blue bar), and finally Thread:1 (Runnable) at the top. A yellow callout box with an arrow pointing to the "SAM2 : 02" entry contains the text: "Program call chain (bottom to top)". To the right of this callout is a list of bullet points: "• Click a program to see it in the source view" and "• Right-click a program to run or see program properties". Below the call chain, the source code editor shows the file "*XSAM.jcl" with the following code:

```
Line 89      Column 1      Insert
-----+*A-1-B--+-----2-----+-----3-----+-----4-----+-----5-----+-----6-----
000087          ADD +1 TO BALANCE-COUNT
000088      *      *** Add this customer's BALANCE to the grand total
000089          COMPUTE BALANCE-TOTAL =
```



Action bar buttons perform program actions....

The image shows a screenshot of the IBM Rational Developer for System z IDE in debug mode. The title bar reads "Debug - \\DebugViewFiles\8001_1145\ADTOOLS.ADLAB.SYSDEBUG(SAM2).cob - IBM Rati". The menu bar includes "File", "Edit", "Navigate", "Search", "Project", "Run", "Window", and "Help". The toolbar contains various icons for debugging actions. Six callout boxes with arrows point to specific buttons in the toolbar:

- Resume:** Run the program to the next breakpoint or to the end
- Terminate:** End the program
- Disconnect:** from the debug engine
- Animated Step:** Steps into the program automatically at a rate you specify
- Step:** run one statement
- Step Over:** run one statement, but step over a CALL
- Step Return:** run until return from subprogram

The background shows a code editor with the following text:

```
Line 8  
000087  
000088  
000089  
BALANCE to the grand total
```

The Program Source view

The highlighted line is the current statement

Set a statement breakpoint by double-clicking in the gray area next to a statement

Right click a statement to:

- find text
- create a stmt breakpoint
- jump or run to a stmt
- see other options

Right click a variable to:

- add it to the monitor view
- create a watch breakpoint
- see other options

```

Line 252      Column 1      Insert      Browse
-----+---+*A-1-B-+-----2-----+-----3-----+-----4-----+-----5-----+-----6-----+-----7---|+-----8
000249
000250      000-MAIN.
000251      ACCEPT CURRENT-DATE FROM DATE.
000252      ACCEPT CURRENT-TIME FROM TIME.
000253      DISPLAY 'SAM1 STARTED DATE = ' CURRENT-MONTH '/'
000254      CURRENT-DAY '/' CURRENT-YEAR ' (mm/dd/yy)'.
000255      DISPLAY '          TIME = ' CURRENT-HOUR ':'
000256      CURRENT-MINUTE ':' CURRENT-SECOND.
000257
000258      PERFORM 900-OPEN-TRAN-AND-RPT-FILES.
000259      PERFORM 800-INIT-REPORT .
  
```

Process: 327205648 Program: SAM1

System z LPEX Editor

The Variables View

View and change variables for the program displayed in the source view.

Right click a variable to:

- display in hex
- add to the memory view
- change the value
- see other options

Expand and collapse group level data elements

Overtype a value to change it

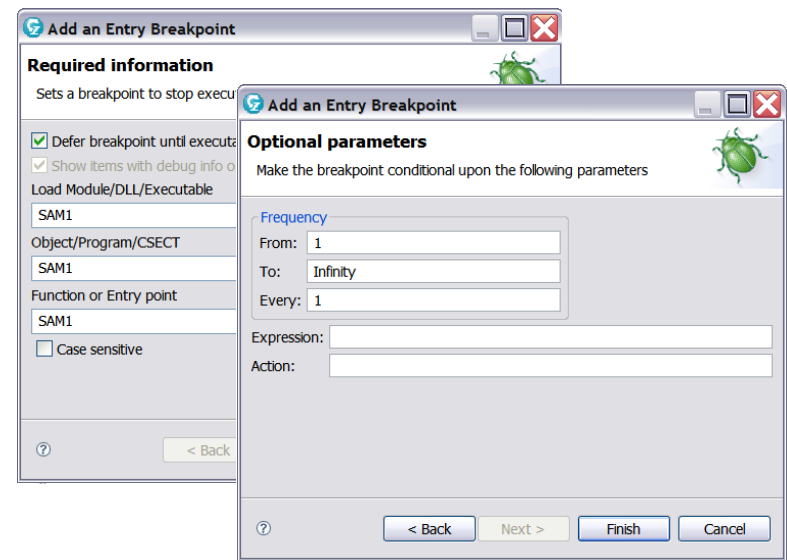
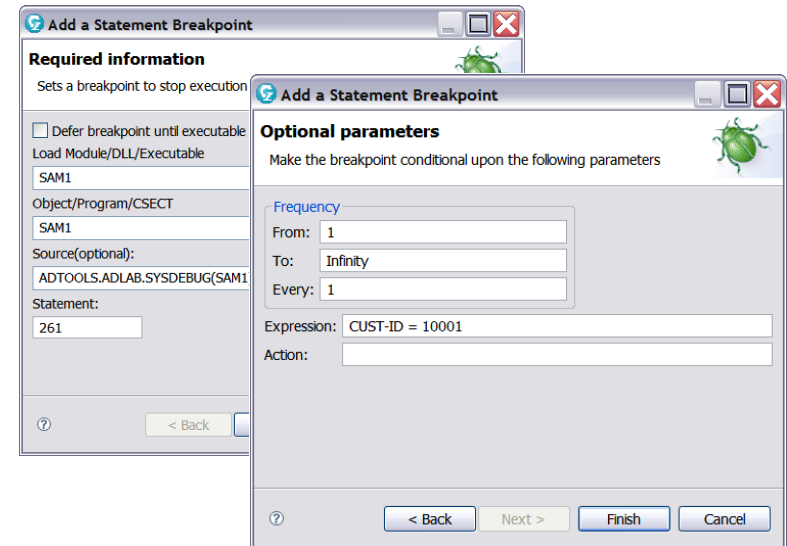
Look at all working storage (show, change screen)

Click on a variable to display it in the expanded area

Name	Value
REPORT-RECORD	'-----'
RPT-DETAIL	
RPT-CUST-ID	'01001'
FILLER	''
RPT-CUST-NAME	'Lynn, Amanda'
FILLER	''
RPT-CUST-OCCUPATION	'Musician'
FILLER	''
RPT-CUST-ACCT-BALANCE	' 67.68'

Menu driven breakpoint building

- Statement Breakpoints
- Watch breakpoint - stops the program when a variable or storage area changes
- Entry Breakpoint – stops the program upon entering a sub-program
- Conditional breakpoints such as: $X > 999$
- Based on a frequency: Stop the Nth time a statement runs
- Also set breakpoints on an address or load module
- Two types (conditional – unconditional)



Additional Debug Tool Interface Items

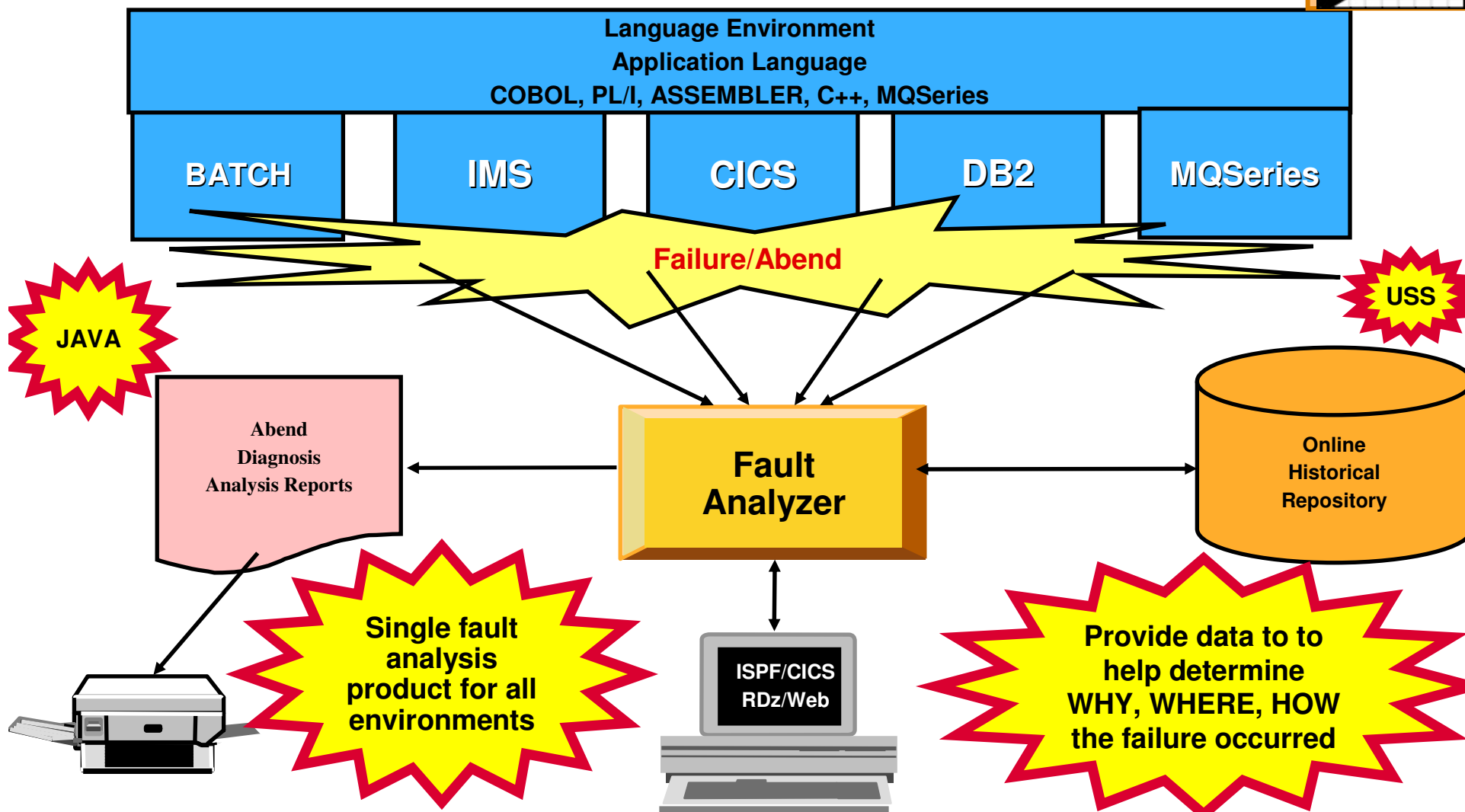
- Integration with the editor facilitates better flow from editing to debugging
- Use of the outline view while debugging
- Content assist while entering debug commands
- Debug Console commands that can also be entered in a command file are added to allow for unattended debugging.
- Team Debugging with IBM Rational Team Concert for System z



Fault Analyzer Interface with RDz



Fault Analyzer – *Abend Analysis*



Fault Analyzer Features

- Analysis at application level
- Information gathered at time of abend
- Translates low-level dump information into application-level information
- Expands abend code and message descriptions
- No recompile of applications
- No JCL changes
- No performance overhead
- Integrated 64-bit DB2 support
- Interface to File Manager for z/OS
- Modes Of Operation
 - ▶ Real-time analysis
 - ▶ Batch dump re-analysis
 - ▶ Interactive dump re-analysis
- Interfaces
 - ▶ ISPF
 - ▶ CICS
 - ▶ RDz
 - ▶ Web
- Support for IBM software
 - ▶ CICS TS 4.1
 - Channels and containers in 64-bit storage
 - ▶ DB2 V9
 - ▶ IMS V10



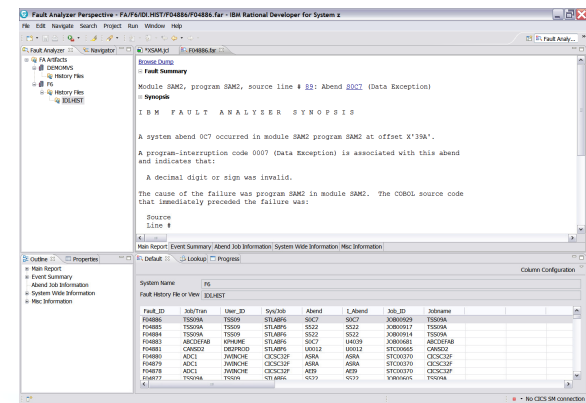
Fault Analyzer Interface with RDz

- Use the cross-platform fault analysis to capture ABEND information captured on the host for various environments.

- ▶ CICS
- ▶ OS/390
- ▶ IMS
- ▶ LE (Language Environment)
- ▶ DB2
- ▶ MQSeries
- ▶ z/OS
- ▶ UNIX System Services

- Viewed from the workstation:

- ▶ Work with multiple fault history files from multiple systems.
- ▶ View dump selections relating to ABENDs.
- ▶ Annotate reports to share comments with other users who browse the same reports.
- ▶ Reference the same editor sessions used for code editing instead of module sidefiles.



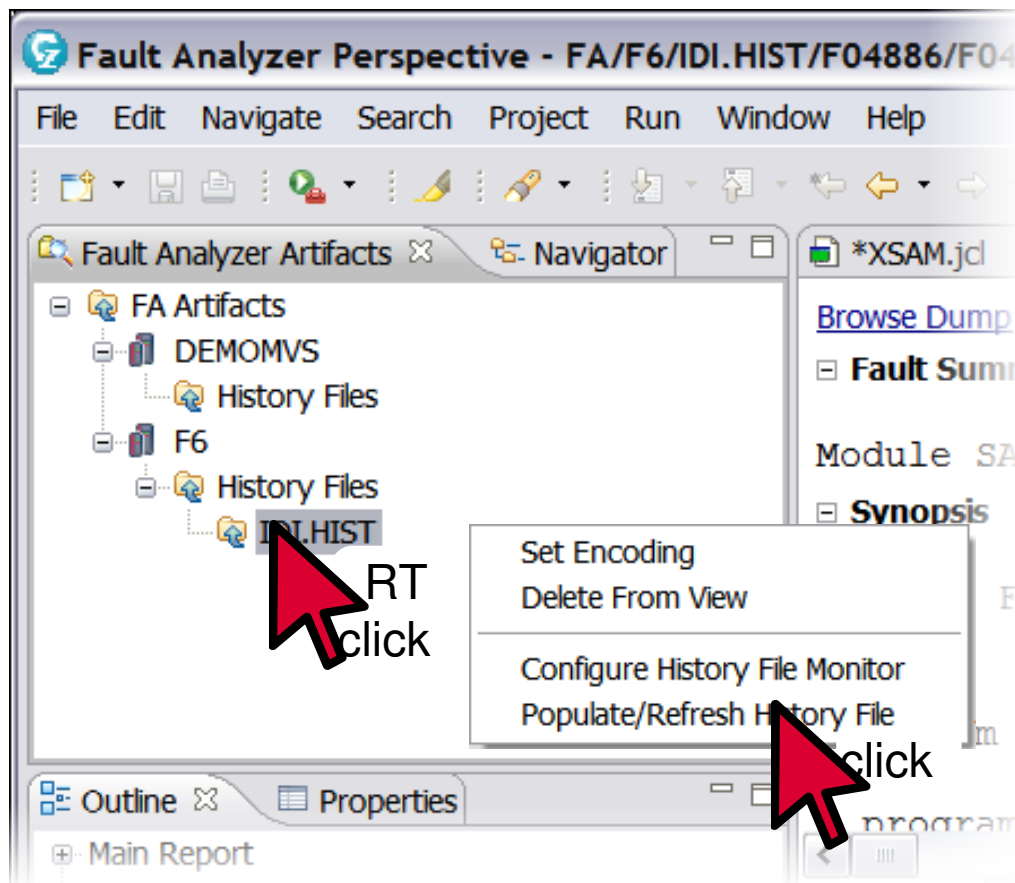
Fault Analyzer Perspective

The screenshot displays the IBM Rational Developer for System z interface for the Fault Analyzer Perspective. The main window title is "Fault Analyzer Perspective - FA/F6/IDI.HIST/F04886/F04886.far". The interface includes a menu bar (File, Edit, Navigate, Search, Project, Run, Window, Help), a toolbar, and several panes:

- FA Artifacts view:** Located on the left, it shows a tree structure with folders for "FA Artifacts", "DEMOMVS", "History Files", "F6", and "IDI.HIST".
- FA Report browser / Dump Storage browser:** The main central pane displays a "Browse Dump" view for "Fault Summary". It shows details for "Module SAM2, program SAM2, source line # 89: Abend S0C7 (Data Exception)". Below this is a "Synopsis" section with the text "I B M FA U L T A N A L Y Z E R S Y N O P S I S" and "system abend 0C7 occurred in module SAM2 program SAM2 at offset X'39A'." It also mentions "A program-interruption code 0007 (Data Exception) is associated with this abend".
- FA Report Outline:** Located at the bottom left, it shows a hierarchical outline of the report sections: "Main Report", "Event Summary", "Abend Job Information", "System Wide Information", and "Misc Information".
- Detail view / Look-up view:** The bottom right pane shows a "Lookup" view with a table of fault data. The table has columns for Fault_ID, Job/Tran, User_ID, Sys/Job, Abend, I_Abend, Job_ID, and Jobname. The data rows are:

Fault_ID	Job/Tran	User_ID	Sys/Job	Abend	I_Abend	Job_ID	Jobname
4886	TSS09A	TSS09	STLABF6	S0C7	S0C7	JOB00929	TSS09A
4885	TSS09A	TSS09	STLABF6	S522	S522	JOB00917	TSS09A
F04884	TSS09A	TSS09	STLABF6	S522	S522	JOB00914	TSS09A

Fault Analyzer Artifacts View Actions



Right Clicking

- ▶ Add new history file
- ▶ Retrieve last accessed history file
- ▶ Retrieve view information
- ▶ Delete from view
- ▶ Populate/refresh history file
- ▶ Configure history file monitor
- ▶ Set encoding



Fault Analyzer Detail View

View and work with Fault entries in the Detail view

Configure columns

Sort Fault entries by clicking on the column

Quick overview of fault entries

Fault_ID	Job/Tran	User_ID	Sys/Job	Abend	I_Abend	Job_ID	Jobname
F04886	TSS09A	TSS09	STLABF6	S0C7	S0C7	JOB00929	TSS09A
F04885	TSS09A	TSS09	STLABF6	S522	S522	JOB00917	TSS09A
F04884	TSS09A	TSS09	STLABF6	S522	S522	JOB00914	TSS09A
F04883	ABCDEFAB	KPHUME	STLABF6	S0C7	U4039	JOB00681	ABCDEFAB
F04881	CANS2	DB2PROD	STLABF6	U0012	U0012	STC00665	CANS2
F04880	ADC1	JWINCH			ASRA	STC00370	CICSC32F
F04879	ADC1	JWINCH			ASRA	STC00370	CICSC32F
F04878	ADC1	JWINCH			AEI9	STC00370	CICSC32F
F04877	TSS09A	TSS09	STLABF6	S522	S522	JOB00605	TSS09A
F04876	TSS09A	TSS09	STLABF6	S522	S522	JOB00496	TSS09A

No CICS SM connection

Fault Analyzer Look-up View

Find definitions to ABEND codes, and system messages with the Look-up view

Enter search term

Code explanation

Manually browse code definitions

Module SAM2, pro (ption)

Synopsis

Main Report | Event Summary | Abend Job Information | System Wide Information | Misc Information

Default | Lookup

Search Go

Abend Codes

- IMS User Abend Codes
- Language Environment User Abend Code
- CICS User Abend Codes
- MVS Abend Codes

Messages

Miscellaneous Information

COBOL File Status

00

02

04

05

Explanation Results

0C7

Explanation: A program interruption occurred, but no routine had been specified to handle this type of interruption. Refer to the instruction description in Principles of Operation to find out how the instruction stc processing for the error condition.

The last digit of this code is the cause of the program interruption. For a reason code, which in most cases equals the last digit. X'0C4' however, several reason codes.

data exception. The reason code is whatever the data exception DWADXC in the SDWA (systems diagnostic work area).

Local Help | No CICS SM connection



Opening a Fault Analyzer Report

The screenshot shows the IBM Fault Analyzer interface. The main window is titled "Synopsis" and has tabs for "Main Report", "Event Summary", "Abend Job Information", "System Wide Information", and "Misc Information". Below the tabs are "Default", "Lookup", and "Progress" buttons. The "Column Configuration" dropdown is visible. The "System Name" is "F6" and the "Fault History File or View" is "IDL.HIST".

Fault_ID	Job/Tran	User_ID	Sys/Job	Abend	I_Abend	Job_ID	Jobname
F04886	TSS09A	TSS09	STLABF6			JOB00929	TSS09A
F04885	TSS09A	TSS09	STLABF6			JOB00917	TSS09A
F04884	TSS09A	TSS09	STLABF6			JOB00914	TSS09A
F04883	ABCDEFAB	TSS09	STLABF6			JOB00901	ABCDEFAB
F04882	CANSD2	DB2PROD	STLABF6			JOB00601	CANSD2
F04881	CANSD2	JWINCHE	CICSC32			STC00370	CICSC32F
F04880	CANSD2	JWINCHE	CICSC32			STC00370	CICSC32F
F04879	CANSD2	JWINCHE	CICSC32			STC00370	CICSC32F
F04877	TSS09A	TSS09	STLABF6			JOB00605	TSS09A
F04876	TSS09A	TSS09	STLABF6			JOB00496	TSS09A

Right click on the entry

RT click

click

click

Column Configuration

Browse Report

Browse Mini-Dump

Refresh Fault Entry

Delete Cached Data

Set Encoding

No CICS SM connection

Fault Analyzer Report Browser View

Read your fault entry report in the multi-tabbed Report Browser View

Hyperlink to dump

Hyperlinks to source line # and ABEND code

Fault Analyzer real-time analysis report

Tabs to other sections

/F6/IDI.HIST/F04886/F0
 Run Window Help
 *XSAM.jcl F04886.far
 Browse Dump
 Fault Summary
 Module SAM2, program SAM2, source line # [89](#): Abend [S0C7](#) (Data Exception)
 Synopsis
 I B M F A U L T A N A L Y Z E R S Y N O P S I S
 A system abend 0C7 occurred in module SAM2 program SAM2 at offset X'39A'.
 A program-interruption code 0007 (Data Exception) occurred and indicates that:
 A decimal digit or sign was invalid
 Main Report | Event Summary | Abend Job Information | System Wide Information | Misc Information
 Default | Lookup | Progress
 Column Configuration

Dump Browser View

Browse the captured mini-dump data stored in the fault entry report in the Dump browser view

Address	Hex Values	Character Values
00000000	+0 00	
- Event 1 Program SAM1 GPR 0 [00000001]		
00000001	+1 0A0000 000130E1 00000000 00000000 00FDE870 00000000 7FFFF000 7FFFF000	*r&.....r.....r.....r.*
00000002	+20 7FFFFFF0 7FFFFFF0 7FFFFFF0 00000000 7FFFFFF0 7FFFFFF0	*+.1. .0.r.....*
00000003	+40 00000000 00000000 00000000 00000000 00000000 000A0000 000140E1	* ..r.....".0.....r...4;&.*
00000004	+60 000A0000 000150E1 000A0000 000160E1 000A0000 000170E1 000A0000 000180E1	*1..08.....*
00000005	+80 00040000 00001004 0002000B 00040016 00000001 00000000 00000000 00000000	
00000006	+A0 0D000001 01499C08 00000000 7F2CF000 00000000 00000000 00012368 00F45E50	
00000007	+C0 28000000 00000000 FBF1FFFB F0F80000 58080000 00000000 00000000 00000000	
00000008	+E0 00000000 00000000 00000000 00000000 00000000 00000000 0	
00000009	+100 00000000 00000000 00000000 00000000 00000000 06A37FBA 0	
0000000A	+120 00000000 00000000 00000000 00000000 00000000 07543300 80000000 0	
0000000B	+140 07850000 80000000 00000000 06A3801C 04041000 80000000 0	
0000000C	+160 00000000 00000000 00000000 00000000 07060000 00000000 0	
0000000D	+180 00000000 00000000 00000000 00000000 00000000 00000000 0	
0000000E	00000000 00000000 00000000 00000000 00000000 00000000 00000000 0	
0000000F	00000000 00000000 00000000 00000000 00000000 00000000 00000000 0	
00000010	00000000 00000000 00000000 00000000 00000000 00000000 00000000 0	
00000011	00000000 00000000 00000000 00000000 00000000 00000000 00000000 0	
00000012	00000000 00000000 00000000 00000000 00000000 00000000 00000000 0	
00000013	00000000 00000000 00000000 00000000 00000000 00000000 00000000 0	
00000014	00000000 00000000 00000000 00000000 00000000 00000000 00000000 0	
00000015	00000000 00000000 00000000 00000000 00000000 00000000 00000000 0	
00000016	00000000 00000000 00000000 00000000 00000000 00000000 00000000 0	
00000017	00000000 00000000 00000000 00000000 00000000 00000000 00000000 0	
00000018	00000000 00000000 00000000 00000000 00000000 00000000 00000000 0	
00000019	00000000 00000000 00000000 00000000 00000000 00000000 00000000 0	
0000001A	00000000 00000000 00000000 00000000 00000000 00000000 00000000 0	
0000001B	00000000 00000000 00000000 00000000 00000000 00000000 00000000 0	
0000001C	00000000 00000000 00000000 00000000 00000000 00000000 00000000 0	
0000001D	00000000 00000000 00000000 00000000 00000000 00000000 00000000 0	
0000001E	00000000 00000000 00000000 00000000 00000000 00000000 00000000 0	
0000001F	00000000 00000000 00000000 00000000 00000000 00000000 00000000 0	
00000020	+220 00FDB600 00F9FB80 00000000 00000000 00000000 00000000 00000000	

Switch between Dump Browser / User Notes

Search for a pattern

Find: CUST-REC

Search Direction: Forward Backward

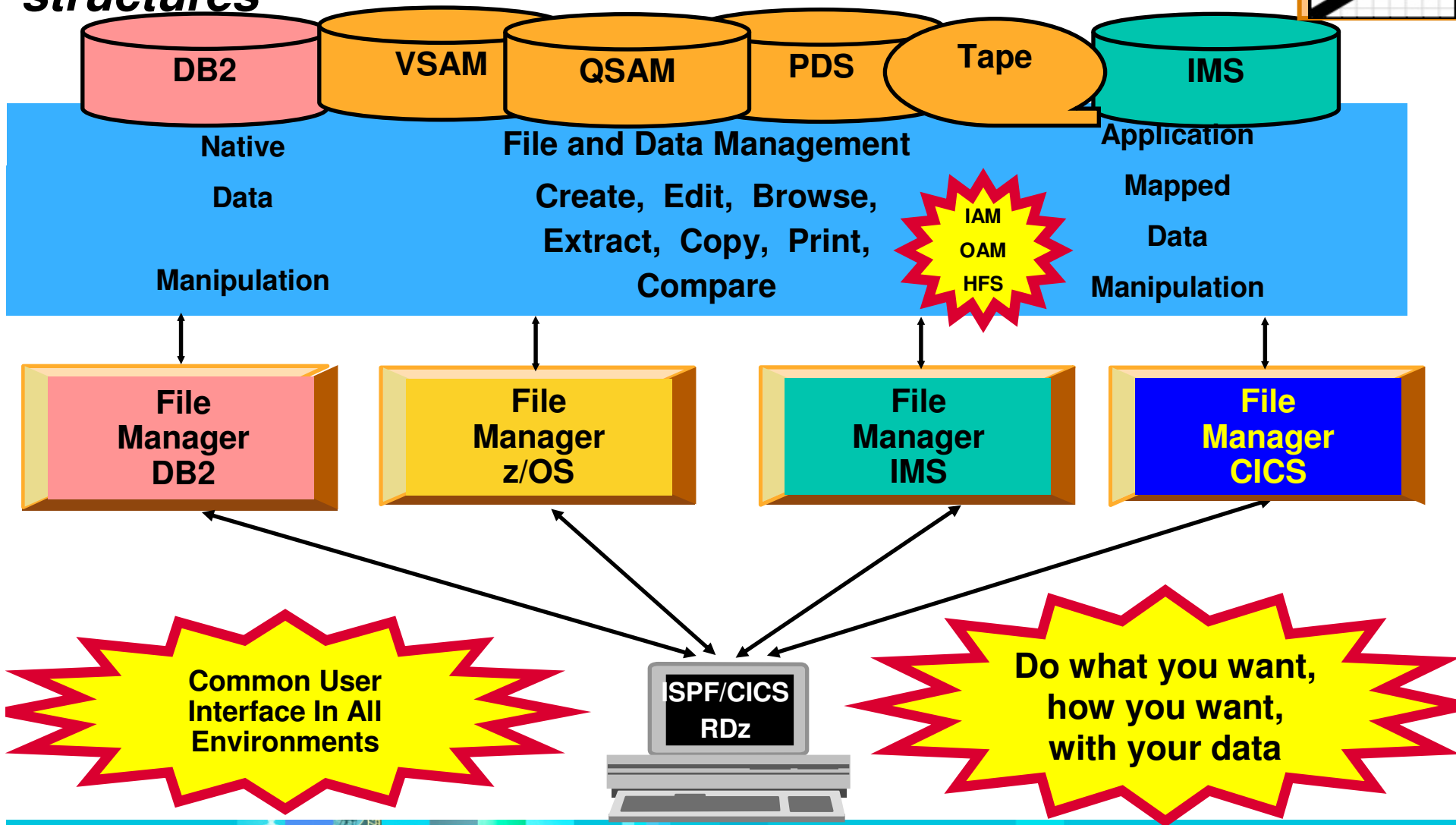
Search Mode: Hexadecimal String

Buttons: Find, Close

File Manager Interface with RDz



File Manager - *Manage a variety of enterprise data file structures*

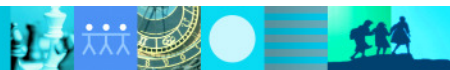


File Manager z/OS Features



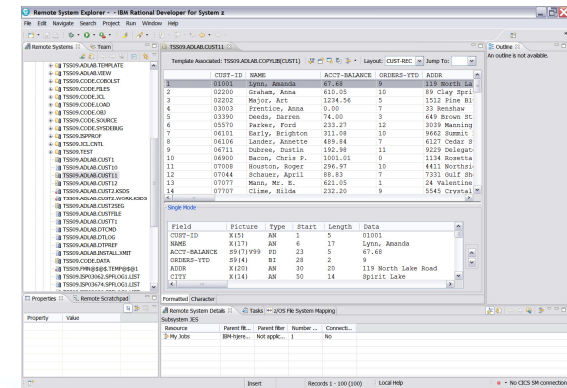
- **Aimed at application developers:**

- ▶ Format VSAM and QSAM data according to definitions in a COBOL or PL1 copybook or assembler **DSECT**, displayed as either:
 - A table, under field column headings
 - A single record per panel, listing field values next to field names
- ▶ Select records with Boolean conditions
- ▶ Select and exclude fields as desired
- ▶ Copy selected records and fields between data sets, **data scrambling** insert/delete/reformat fields and XML creation
- ▶ **Edit files regardless of size**
- ▶ Compare data or **load modules** between data sets according to user-specified patterns



File Manager Interface with RDz

- Facilitates workstation access to data files on the host
- Formatted display of:
 - ▶ VSAM - KSDS, ESDS, etc.
 - ▶ QSAM – PDS, SEQ, etc.
- Menu driven template creation
- Multiple views of the data the formatted editor:
 - ▶ Table
 - ▶ Single
 - ▶ Character
- VSAM data utilities
 - ▶ Copy
 - ▶ Allocate
- Browse very large files quickly
 - ▶ Work with a sub-set of data downloaded to the workstation



Formatted Data Editor

Display and edit PDS members, sequential datasets, and VSAM datasets using the Formatted Data editor

Template Associated: TSS09.ADLAB.COPYLIB(CUST1)

Layout: CUST-REC

CUST-ID	NAME	ACCT-BALANCE	ORDERS-YTD	ADDR
01001	Lynn, Amanda	67.68	9	119 North Lake Road
02200	Graham, Anna	610.05	10	89 Clay Springs Rd
02202	Major, Art	1234.56	5	
03003	Prentice, Anna	0.00	7	
5	03390	Deeds, Darren	74.00	649 Brown Street
6	05570	Parker, Ford	233.27	3039 Manning St.

Associated template

Table Format

Single Mode

Field	Picture	Type	Start	Length	Data
CUST-ID	X(5)	AN	1	5	01001
NAME	X(17)	AN	6	17	Lynn, Amanda
ACCT-BALANCE			23	5	67.68
ORDERS-YTD			28	2	9
ADDR			30	20	119 North Lake Road
CITY	X(14)	AN	50	14	Spirit Lake

Switch to unformatted

Single Format

Formatted Character

Remote System Details | Tasks | z/OS File System Mapping

Subsystem JES

Formatted Data Editor Actions

The screenshot shows the Formatted Data Editor window with several callout boxes pointing to specific controls:

- Hex mode**: Points to the hex icon (010 01).
- Page Up / Down**: Points to the up and down arrow icons.
- Select show setting**: Points to the show settings icon.
- Switch between record layouts**: Points to the 'Layout: CUST-REC' dropdown menu.
- Navigate to "Top", "Bottom", or line number**: Points to the 'Jump To: TOP' dropdown menu.
- Collapse / Expand Single mode**: Points to the collapse/expand icon.

The main data table is as follows:

ACCT-BALANCE	OR	
9		North Lake Road
10		Clay Springs Rd
1234.56	5	1512 Pine Bluff
0.00	7	33 Renshaw

A dropdown menu for 'Select show setting' is open, showing the following options:

- Not selected records
- Suppressed records
- Excluded records
- All records



Copy Utility

The screenshot shows the Copy Utility interface. On the left, a file tree is displayed under the 'Remote Systems' window. The tree structure is as follows:

- Remote Systems
- Team
- F6
 - z/OS UNIX Files
 - z/OS UNIX Shells
 - MVS Files
 - My Data Sets
 - TSS09.ADLAB.COPYLIB
 - TSS09.ADLAB.EQALANGX
 - TSS09.ADLAB.FILES
 - @FILECHK
 - BTSIN
 - CUST2.tdat
 - CUST2FA.tdat (selected)
 - DTPREF
 - DXCOND
 - DXFREQ

A red mouse cursor points to the 'CUST2FA.tdat' file. A text box 'Select, a PDS' is overlaid on the tree. A context menu is open over the selected file, listing various actions:

- Go To
- Open
- Open With
- Refresh (F5)
- Rename... (F2)
- Delete... (Delete)
- Copy
- Search...
- Host Connection Emulator
- Browse
- Add to Subproject...
- Property Group
- File Manager
- Compare With
- Replace With
- Properties (Alt+Enter)

A sub-menu is open for the 'Copy' action, showing:

- Advanced Allocate
- Copy (highlighted)
- Create Template

Two red mouse cursors are present: one pointing to 'CUST2FA.tdat' with the text 'RT click' and another pointing to the 'Copy' option in the sub-menu with the text 'click'.

Copy Utility (continued)

The image displays three sequential screenshots of the IBM Advanced Copy utility interface, illustrating the steps in a copy operation. Each window is titled "Advanced Copy" and contains the following elements:

- First Screenshot:** Shows the "Source Information" section for dataset "TSS09.ADLAB.FILES". The "Number of Records to Skip" is set to 100. A callout box labeled "Skip / copy records" points to these fields. Below, the "Select Source Members" section lists members like @FILECHK, BTSIN, CUST2.tdat, CUST2FA.tdat, DATA01, DFSVSAMP, DTCMD, and DTCMD2. A callout box labeled "Select members to work with" points to this list. The "Next >" button is highlighted with a red mouse cursor and the text "click".
- Second Screenshot:** Shows the "Choose Target" section. The "Target" field is populated with "TSS09.ADLAB.CUST12". A callout box labeled "Indicate target dataset" points to this field. The "Disposition" dropdown menu is open, showing options "MOD", "MOD", and "OLD". A callout box labeled "Choose disposition" points to this menu. The "Next >" button is highlighted with a red mouse cursor and the text "click".
- Third Screenshot:** Shows the "Map Template" section. The "Use Templates" checkbox is checked. The "From" field is "TSS09.ADLAB.TEMPLATE(TEST12)" and the "To" field is "TSS09.ADLAB.TEMPLATE(TEST13)". Callout boxes labeled "Associate 'From' and 'To' templates (optional)" point to these fields. The "Finish" button is highlighted with a red mouse cursor and the text "click".

Summary

- Rational Developer for System z (RDz) is not a replacement for TSO/ISPF, but reduces the **effort** needed to **learn** and **perform** tasks.
 - ✓ Support enterprise modernization and SOA
 - ✓ Develop of a mixture of batch and online applications
 - ✓ Reduce the learning curve
 - ✓ Replace the part of the system that “limits” production

- PD Tool Interfaces with RDz, makes this possible
 - ✓ Add flexibility, while retaining the **proven functionality** of the PD Tools
 - ✓ Use interfaces for Fault Analyzer, File Manager, and Debug Tool
 - ✓ Navigate quickly between multiple product’s functionality
 - ✓ Perform complex tasks with the aid of menus and toolbars

- **Gain maximum leverage of existing IT staff with minimal training expense**



Get more information about these tools at:
www.ibm.com/software/awdtools/deployment

Software > Software Development >

z/OS Problem Determination Tools

z/OS Problem Determination Tools

Library

News

How to buy

Training and certification

Services

Related software

- Application Performance Analyzer for z/OS
- DebugTool for z/OS
- Fault Analyzer for z/OS
- Optim Move for DB2
- File Manager for z/OS
- Workload Simulator for z/OS and OS/390

Related hardware

- System z servers

Related services

- Application Time Facility for z/OS
- Warranties and licenses

z/OS® Problem Determination Tools have powerful functions and features. Organizations that choose to use them improve the health of their application portfolios. To help you to transform your System z environment into a service-oriented architecture (SOA) hub, the IBM Problem Determination Tools deliver support right across the life cycle whether you are building new or reusing existing applications. These tools can help you to modernize and transform existing System z applications whether your goal is to develop and deploy new workloads to leverage the unique performance, availability, security, and cost benefits of System z, increase your responsiveness to business requirements by modernizing your mainframe platform, or optimize management of your IT environment, reducing cost and complexity while improving governance and compliance. These latest versions of the IBM Problem Determination Tools continue the trend of cost-effectively protecting tool investments and maximizing IT productivity.

Select a product



Products

- **Application Performance Analyzer for z/OS**
 A non-intrusive application performance analyzer that aids developers in the design, development and maintenance cycles. Its key function is to measure and report how resources are used by applications running in virtually any z/OS address space.

Solutions

- **Safari of IBM PD Tools: A Live Exploration**
 Join us in this complimentary seminar for hands-on labs that will build your understanding of IBM problem determination tool capabilities, so you can develop applications more efficiently.
- **System z Enterprise Development Tools and Compilers information**

We're here to help



Easy ways to get the answers you need.

Call me now

Request a quote

E-mail IBM

or call us at
 877-426-3774
 Priority code:
 104CBW67

IBM PD Tools win top spot in Software Strategies analyst report



Highlights



IBM Education Assistant

<http://publib.boulder.ibm.com/infocenter/ieduasst/stgv1r0/index.jsp>

Navigation pane to access available materials (Web-based training, Classroom-based training, Maintenance, and Additional Resources)

Multimedia content including voiced-over tutorials

Topic	Duration	Size
About this training	2 min	143K
Introduction	7 min	6.9M
Using the File Manager DB2 editor	23 min	20.5M
Using templates	13 min	11.8M
Your options settings	14 min	11.5M

IBM PD Tools Product Identification

- Fault Analyzer for z/OS (5655-V51)
- File Manager for z/OS (5655-V52)
- Debug Tool for z/OS (5655-V50)
- Application Performance Analyzer for z/OS (5697-P36)
- Application Performance Analyzer Automation Assistant for z/OS (5799-HGC)
- Workload Simulator for z/OS & OS/390 (5655-I39)
- Migration Utility (5697-N44)
- HourGlass (5697-N42)
- ISPF Productivity Tool (5698-R21)
- Rational Developer for System z



Q&A





© Copyright IBM Corporation 2008. 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. Product release dates and/or capabilities referenced in these materials may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way.

IBM, the IBM logo, the on-demand business logo, Rational, the Rational 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.

